Sitecore Connect for Content Hub 5.0 Installation Guide

A guide to installing Sitecore Connect for Content Hub 5.0

April 28, 2022

Table of Contents

1. Prerequisites 3
2. Install Sitecore Connect for Content Hub 4
3. Install Sitecore Connect for Content Hub on a CD server 5
4. Upgrade Sitecore Connect for Content Hub to 5.0 on Sitecore 10.2 6
   4.1. Prerequisites 6
   4.2. Prepare for upgrading SCCH 6
   4.3. Upgrade process 6
5. Upgrade a CMP or SCCH connector to SCCH 5.0 on Sitecore 10.1.x or earlier 9
   5.1. Prerequisites 9
   5.2. Upgrade process 9
6. Upgrade an existing DAM connector to SCCH 5.0 10
7. Troubleshoot installation errors 11
8. Configure Sitecore Connect for Content Hub 14
   8.1. Configure the Sitecore CMP part of Sitecore Connect for Content Hub 14
      8.1.1. Add connection strings for CMP to your Sitecore instance 14
      8.1.2. Create a user in Sitecore Content Hub 15
      8.1.3. Create a Sitecore Content Hub action 16
      8.1.4. Create a trigger 17
      8.1.5. Configure content and mappings 18
      8.1.6. Change the taxonomy provider 26
   8.2. Configure the Sitecore DAM part of Sitecore Connect for Content Hub 27
      8.2.1. Configure CORS 27
      8.2.2. Configure redirection settings 28
      8.2.3. Configure the DAM connector in Sitecore 29
9. Problem solving 32
   9.1. General troubleshooting help 32
   9.2. In Azure, browsing a Sitecore DAM link displays a 404 error 32
   9.3. The rich text editor in Sitecore shows an XHTML related validation error 32
   9.4. Azure deployment throws Could not load file or assembly 'Microsoft.Extensions.Caching.Abstractions' error 33
   9.5. On-prem deployment throws Method does not have an implementation error 34

© Copyright 2021, Sitecore® - all rights reserved.

1. Prerequisites

Before installing the Sitecore Connect for Content Hub module, you must have the following:

- Sitecore Experience Platform 9.2, 9.3, 10.0.x, 10.1, or 10.2 installed on your server
- Access to Sitecore Content Hub 3.4.x, 4.0.x, or 4.1.

You must also ensure that the following network ports are available and not blocked by your Windows firewall:

Required port	Role
443	HTTPS
5671	AMQP
9354	Net messaging/SBMP

© Copyright 2021, Sitecore® - all rights reserved.

2. Install Sitecore Connect for Content Hub

To install Sitecore Connect for Content Hub (SCCH) on your Sitecore server:

1. Download the SCCH module installation package and the SCCH Sitecore Installation Framework (SIF) from the Sitecore download page.
2. Unzip the SCCH SIF installation package to your server.
3. Open a PowerShell window with administrator privileges. Use the following command to run the SCCH SIF installation script:

```
<path where you extracted the SCCH SIF installation script>\deploy.ps1 `
-ScchWdpPackage "<path of SCCH module installation package downloaded>" `
-InstanceName "<Sitecore instance name>" `
-InstanceUrl "<Sitecore instance url>" `
-SitecoreAdminUser "<Sitecore admin user name>" `
-SitecoreAdminPass "<Sitecore admin password>" `
-SqlInstanceName "<Sitecore SQL Database prefix name>" `
-SqlUser "<SQL server user name>" `
-SqlPass "<SQL server password>" `
-SqlServerName "<SQL server name>" `
-SkipDatabaseInstallation "<true or false for skip database installation>"
```

NOTE

If you are deploying SCCH to Sitecore 10.2, you must set the `SkipDatabaseInstallation` parameter to `$true`.

For the `SqlInstanceName` parameter, you can find the prefix in the Sitecore database name which is in the form of `<Sitecore SQL Database prefix name>_Core`.

© Copyright 2021, Sitecore® - all rights reserved.

3. Install Sitecore Connect for Content Hub on a CD server

If you have a Sitecore content delivery (CD) server, you can install the Sitecore Connect for Content Hub (SCCH) on it. This enables you to populate images from your DAM to your CD server.

To install SCCH on your CD server:

1. Install the SCCH installation package as described in Install Sitecore Connect for Content Hub.

   > **NOTE**
   > When you run the SCCH SIF installation script, you must use the instance name and instance URL for your CD server in the parameters.

2. To enable DAM images to be populated in CD website, go to the `C:\inetpub\<CD site>\xsl` folder and open the `xslt` file, for example, `sample rendering.xslt`, in a text editor. Add an `sc:image field` line to the `xslt` file. For example:

```
<!-- entry point -->
<xsl:template match="*">
  <div>
    <h1 class="contentTitle">
      <sc:text field="title"/>
    </h1>
    <div class="contentDescription">
      <sc:text field="text" />
      <sc:image field="image" />
    </div>
  </div>
</xsl:template>
```

3. Save the `xslt` file.

© Copyright 2021, Sitecore® - all rights reserved.

4. Upgrade Sitecore Connect for Content Hub to 5.0 on Sitecore 10.2

This section describes how to upgrade your installation from Sitecore Connect for Content Hub (SCCH) to 5.0, if you intend to upgrade your Sitecore Experience Platform (SXP) installation to 10.2.

4.1. Prerequisites

Before you upgrade SCCH and SXP you must have the following installed:

- Sitecore Experience Platform 10.1 or earlier
- Sitecore Connect for Content Hub 4.0 or earlier

4.2. Prepare for upgrading SCCH

To prepare for upgrading the Sitecore Connect for Content Hub (SCCH) connector to version 5.0, you must:

- Create backups of your current Sitecore databases.
- Create a backup copy of the Sitecore folders under *wwwroot* for your current Sitecore instance.

4.3. Upgrade process

To upgrade Sitecore and SCCH:

1. Download the *Upgrade Guide for Sitecore Experience Platform* for SXP 10.2.0 from the Sitecore download page. Follow the Upgrade Guide to prepare the upgrade and set up the `Sitecore.UpdateApp` tool. When you get to the *Clean the content databases* section, perform the following instructions before continuing with the Upgrade Guide.
2. From the Resource files for Modules page, download the Sitecore Connect for Content Hub Upgrade resources package.
3. Extract the `Sitecore Connect for Content Hub Upgrade resources 1.0.0.zip` package to a local directory.
4. Navigate to the `Sitecore Connect for Content Hub Upgrade resources 1.0.0\<version>\Data` folder (for the version you are upgrading from), and copy the `.dat`

© Copyright 2021, Sitecore® - all rights reserved.

files that are included in the `core`, `master`, and `web` folders, to the corresponding folder paths under `\Sitecore.UpdateApp\`. For example:

- Copy the `Data\Items\master\items.master.chub.dat` files and paste it in the `\Sitecore.UpdateApp\Data\Items\master` folder.
- Copy the `Data\Items\core\items.core.chub.dat` file, and paste it in the `\Sitecore.UpdateApp\Data\Items\core` folder.
- Copy the `Data\Items\web\items.web.chub.dat` file, and paste it in the `\Sitecore.UpdateApp\Data\Items\web` folder.

> **NOTE**
> Each folder under the path `\Sitecore.UpdateApp\Data\Items\` now contains two files, that is, the file you just copied and the `items.<master/web/core>.dat` file.

5. For the localization files, navigate to the `Sitecore Connect for Content Hub Upgrade resources 1.0.0\<version>\Data` folder (for the version you are upgrading from), and copy the `.dat` language files that are included in the *localization* subfolders to the corresponding folder paths under `\Sitecore.UpdateApp\`. For example:
   - Navigate to the `Items\core\localization` folder, copy all the `.dat` language files, and paste them in the `\Sitecore.UpdateApp\Data\Items\core\localization` folder.
   - Navigate to the `Items\master\localization` folder, copy all the `.dat` language files, and paste them in the `\Sitecore.UpdateApp\Data\Items\master\localization` folder.
   - Navigate to the `Items\web\localization` folder, copy all the `.dat` language files, and paste them in the `\Sitecore.UpdateApp\Data\Items\web\localization` folder.

   > **NOTE**
   > If you have more than one Web database, you must repeat Step 4 through Step 5 for every Web database.
   >
   > You must update the connection string to each Web database as you run the tool, and not change the connection strings to the other databases.

6. In the `Sitecore.UpdateApp\App_config` folder, update the `ConnectionStrings.config` file with the connections to your databases. If you do not have a security database, use the connection to the core database for the security connection string.
7. Follow the steps in the *Upgrade Guide for Sitecore Experience Platform* to run the update tool. Check the logs in the `Data\logs` folder for detailed information about any modified items that were skipped by the clean up. The update tool displays the number of items it removed in the command prompt window. For example:

© Copyright 2021, Sitecore® - all rights reserved.

```
C:\Sitecore.UpdateApp.Content\content>Sitecore.UpdateApp.exe clean
Clean up resource items:
master: 5592 item(s)
web: 5591 item(s)
core: 19448 item(s)
Resource items are cleaned up.

C:\Sitecore.UpdateApp.Content\content>
```

© Copyright 2021, Sitecore® - all rights reserved.

5. Upgrade a CMP or SCCH connector to SCCH 5.0 on Sitecore 10.1.x or earlier

If you have an existing Sitecore Connect for Sitecore CMP 3.0 or Sitecore Connect for Content Hub (SCCH) 4.0, you can upgrade it to SCCH 5.0. The procedure is the same regardless of whether your installation is on-prem or in Azure.

> **NOTE**
> This section assumes you are not upgrading your Sitecore installation to 10.2.

5.1. Prerequisites

Before you upgrade to SCCH 5.0 you must have the following installed:

- Sitecore Experience Platform 10.1.x or earlier
- Sitecore Connect for Sitecore CMP 3.0 or Sitecore Connect for Content Hub 4.0.

5.2. Upgrade process

To upgrade SCCH:

1. Install the SCCH 5.0 installation package as described in Install Sitecore Connect for Content Hub.
2. Configure Sitecore Connect for Content Hub as described in Configure Sitecore Connect for Content Hub.

© Copyright 2021, Sitecore® - all rights reserved.

6. Upgrade an existing DAM connector to SCCH 5.0

If you have an existing Sitecore Connect for Sitecore DAM (SCDAM) 2.0 installation, you can upgrade it to SCCH 5.0. The procedure is the same regardless of whether your installation is on-prem or in Azure.

To upgrade an existing SCDAM installation to SCCH 5.0:

1. Install the SCCH 5.0 installation package as described in the installation instructions.
2. When the installation is done, delete the following files:
   - `<Sitecore instance>\sitecore\shell\Controls\Rich Text Editor\StylelabsMAssets\MAssets Commands.js`
   - `<Sitecore instance>\App_Config\Modules\Stylelabs\Stylelabs.config`
   - `<Sitecore instance>\bin\Stylelabs.M.Sitecore.Integration.Components.dll`.
3. Open the `C:\inetpub\<site>\App_Config\ConnectionStrings.config` file. Update the DAM connection string as follows:

```
<add name="DAM.ContentHub"
connectionString="ClientId={client_id};ClientSecret={client_secret};UserName={username};Pas
sword={password};URI={uri};" />
```

4. In the Sitecore Launchpad, click **Desktop**. Switch from the *Master* database to the *Core* database.
5. Open the Content Editor. Navigate to *sitecore/system/Field types/Simple types/Image*. On the **Content** tab, in the **Data** section, update the following values:

Field name	Default value	Change to
Assembly	`Stylelabs.M.SiteCore.Integration.Components`	`Sitecore.Connector.ContentHub.DAM.Lin`
Class	`Stylelabs.M.SiteCore.Integration.Components.Link.MImageControl`	`Sitecore.Connector.ContentHub.DAM.Lin`
Control	Can be any value	Empty value

6. Save the item. Switch back to the *Master* database.

© Copyright 2021, Sitecore® - all rights reserved.

7. Troubleshoot installation errors

When the installation finishes, one of the following error messages might appear. Follow the instructions to correct the error.

Could not load file or assembly 'System.Diagnostics.DiagnosticSource'

If you get the *Could not load file or assembly 'System.Diagnostics.DiagnosticSource, Version=4.0.2.0'* exception, do the following:

1. In the `web.config` file, add the following *bindingRedirect* tag:

```
<dependentAssembly>
    <assemblyIdentity name="System.Diagnostics.DiagnosticSource"
        publicKeyToken="cc7b13ffcd2ddd51"
        culture="neutral"/>
    <bindingRedirect oldVersion="0.0.0.0-4.0.2.1" newVersion="4.0.2.1"/>
</dependentAssembly>
```

2. Run the installation again.

Could not load file or assembly 'Microsoft.IdentityModel.Clients.ActiveDirectory'

If you get the *Could not load file or assembly 'Microsoft.IdentityModel.Clients.ActiveDirectory'* exception:

1. In the `web.config` file, check that the following bindings exist:

```
<dependentAssembly>
  <assemblyIdentity
    name="Microsoft.IdentityModel.Clients.ActiveDirectory"
    publicKeyToken="31bf3856ad364e35"
    culture="neutral"
    xmlns="urn:schemas-microsoft-com:asm.v1" />
  <bindingRedirect
    oldVersion="0.0.0.0-3.19.3.10102"
    newVersion="3.17.2.31801"
    xmlns="urn:schemas-microsoft-com:asm.v1" />
</dependentAssembly>

<dependentAssembly>
  <assemblyIdentity
    name="Microsoft.IdentityModel.Clients.ActiveDirectory.Platform"
    publicKeyToken="31bf3856ad364e35"
    culture="neutral"
```

© Copyright 2021, Sitecore® - all rights reserved.

```
    xmlns="urn:schemas-microsoft-com:asm.v1" />
  <bindingRedirect
    oldVersion="0.0.0.0-3.19.3.10102"
    newVersion="3.17.2.31801"
    xmlns="urn:schemas-microsoft-com:asm.v1" />
</dependentAssembly>
```

2. Add any missing bindings to the `web.config` file.
3. Run the installation again.

Could not load file or assembly 'Polly'

If you get the *Could not load file or assembly 'Polly'* exception:

- Read this KB article to resolve the issue.

Could not load file or assembly ‘Microsoft.Extensions.Caching.Abstractions' or ‘Microsoft.Extensions.Caching.Memory’

If you get the "*Could not load file or assembly‘Microsoft.Extensions.Caching.Abstractions' or ‘Microsoft.Extensions.Caching.Memory’* exception:

1. In the `web.config` file, verify that the following bindings exist:

```
<dependentAssembly>
    <assemblyIdentity
        name="Microsoft.Extensions.Caching.Abstractions"
        publicKeyToken="adb9793829ddae60"
        culture="neutral"/>
    <bindingRedirect
        oldVersion="0.0.0.0-3.1.5.0"
        newVersion="3.1.5.0"/>
</dependentAssembly>
<dependentAssembly>
    <assemblyIdentity
        name="Microsoft.Extensions.Caching.Memory"
        publicKeyToken="adb9793829ddae60"
        culture="neutral"/>
    <bindingRedirect
        oldVersion="0.0.0.0-3.1.5.0"
        newVersion="3.1.5.0"/>
</dependentAssembly>
```

2. If any of the bindings are missing, add them to the `web.config` file, and run the installation again.

© Copyright 2021, Sitecore® - all rights reserved.

3. Check if the `bin` folder contains the `Microsoft.Extensions.Caching.Abstractions.dll` file. If it does not, download the correct version of the file from `https://www.nuget.org`, and paste it into the `bin` folder.

© Copyright 2021, Sitecore® - all rights reserved.

8. Configure Sitecore Connect for Content Hub

The Sitecore Connect for Content Hub package contains connectors for Sitecore CMP and Sitecore DAM. These connectors must be configured separately.

8.1. Configure the Sitecore CMP part of Sitecore Connect for Content Hub

The Sitecore CMP part of Sitecore Connect for Content Hub enables you to work with content created in Sitecore Content Hub directly in Sitecore. Content is automatically created and updated when changes are made in Sitecore Content Hub.

This section describes the configuration steps you must perform on your Sitecore Content Hub and Sitecore instances:

- Add connection strings for CMP to your Sitecore instance
- Create a Sitecore Content Hub user
- Create a Sitecore Content Hub action
- Create a Sitecore Content Hub trigger
- Add the connection strings
- Configure content and field mappings
- Configure the taxonomy provider

8.1.1. Add connection strings for CMP to your Sitecore instance

After you have installed the package, you must configure the connection strings for CMP.

To add the connection strings:

1. On your Sitecore instance, open the `C:\inetpub\<site>\App_Config\ConnectionStrings.config` file. Add the following connection strings:
   - `<add name="CMP.ContentHub" connectionString="ClientId={client_id};ClientSecret={client_secret};UserName={username};Password={password};URI={uri};" />`
   - `<add name="CMP.ServiceBusEntityPathIn" connectionString="{Azure Service Bus connection string with incoming topic}" />`
   - `<add name="CMP.ServiceBusSubscription" connectionString="{Subscription name}" />`
   - `<add name="CMP.ServiceBusEntityPathOut" connectionString="{Azure Service Bus connection string with outcoming topic}" />`

© Copyright 2021, Sitecore® - all rights reserved.

2. Open the Sitecore Content Hub. On the ribbon, click **Manage**, click **OAuth Clients**, click **LogicApp**, and click **Edit**. Make a note of the **Client id** and **Client Secret** settings.
3. In the connection strings, insert the following values:
    - **Client id** and **Client Secret** - enter the LogicApp settings you obtained in the previous step.
    - **User Name** and **Password** - these fields must correspond to the values in the Sitecore Content Hub for the user that you created.
    - **URI** - set the URI to Sitecore Content Hub.
    - **Incoming topic** - set the connection string for the Azure Service Bus and set the name for the topic of the messages posted by the Sitecore Content Hub, posted after the trigger is fired.
    - **Subscription name** - set the name of the subscription.
    - **Outgoing topic** - set the name of the topic of the messages posted to the Azure Service Bus after content is imported to Sitecore.
4. If you are using the M Azure Service Bus action, insert the following values:
    - **CMP.ServiceBusSubscription** - set the value to `hub_out_subscription`.
    - **CMP.ServiceBusEntityPathOut** - add the `Hub in` value you copied when you created the action.
    - **CMP.ServiceBusEntityPathIn** - add the `Hub out` value you copied when you created the action.
5. After you set or adjust the connection strings, restart the Sitecore website.

8.1.2. Create a user in Sitecore Content Hub

To run synchronization with Sitecore Content Hub, you must have a Sitecore Content Hub user with access to the necessary data.

To give the user permission to run synchronizations, you must assign the correct access rights to the user. The rights you assign depend on whether the user needs access to the Digital Assets Management (DAM) part of Content Hub, the Content Management Platform (CMP) part, or both.

To create a Sitecore Content Hub user:

1. Open Sitecore Content Hub, and on the ribbon, click **Manage**.
2. On the **Users** tab, click **Add User**.
3. In the **Username** field, enter the user name, and click **Save**.
4. To assign user rights for DAM:
    - In the `M.Builtin.SitecoreDAM.Everyone` user group, add a new rule for the `M.Asset` entity, and select the `Read` and `ReadPublicLinks` rights.
    - To assign the user to the group, on the **User groups** tab, click **Add to User group**, select the check box and click **Select**.

© Copyright 2021, Sitecore® - all rights reserved.

5. To assign user rights for CMP, the user must have *read* access to the relevant content hub entity. For example:
    - Access to the `sync blog` entity requires read access to `M.Content`.
    - Access to the `sync product` entity requires read access to `M.PCM.Product`.
6. Optionally, to add the user group to a policy combination, in the **Policy combination** panel, click **Add item**.
7. Click **Save**.
8. On the **Details** tab, click **Edit profile**.
9. In the **Email** field, enter an email address, and click **Save**.
10. Click **Reset Password,** and click **Yes** to confirm. A password reset link is sent to the email address previously provided. If the email message does not appear, check your spam folder.

8.1.3. Create a Sitecore Content Hub action

You use Sitecore Content Hub actions to connect to an Azure Service Bus and post a message when a specified trigger fires.

You can use two methods to create a Sitecore Content Hub action:

- M Azure Service Bus. Use this to connect to an out-of-the-box Azure Service Bus on a Content Hub instance.
- Azure Service Bus. Use this to connect to an Azure Service Bus.

Create an M Azure Service Bus action

To create an M Azure Service Bus action:

1. In the Sitecore Content Hub, click **Manage**.
2. Click **Actions**.
3. Click **New action**.
4. Enter, for example, the following values:
    - **Name**: *Sitecore*
    - **Type**: *M Azure Service Bus*
    - **Label**: Sitecore
5. Copy the `Hub in` and `Hub out` string values. You need these to configure the Sitecore connection string as specified in Configure the Sitecore CMP part of Sitecore Connect for Content Hub
6. To save the new action, click **Save**.

Create an Azure service bus action

To create an Azure Service Bus action:

1. In the Sitecore Content Hub, click **Manage**.

© Copyright 2021, Sitecore® - all rights reserved.

2. Click **Actions**.
3. Click **New action**.
4. Enter, for example, the following values:
   - **Name**: *Sitecore*
   - **Type**: *CMP Azure Service Bus*
   - **Connection string**: enter the connection string for the service bus here. The connection string must be a connection string to the entire Azure Service Bus.
   - **Destination type**: *Topic*
   - **Destination**: *hub_out*
5. To test the connection to the Azure Service Bus, click **Test Connection**.
6. To save the new action, click **Save**.

8.1.4. Create a trigger

The Sitecore Content Hub provides users with a trigger framework that allows them to configure triggers using the user interface. A trigger is fired when a certain event occurs – with certain conditions – and can initiate a set of actions.

To create a trigger:

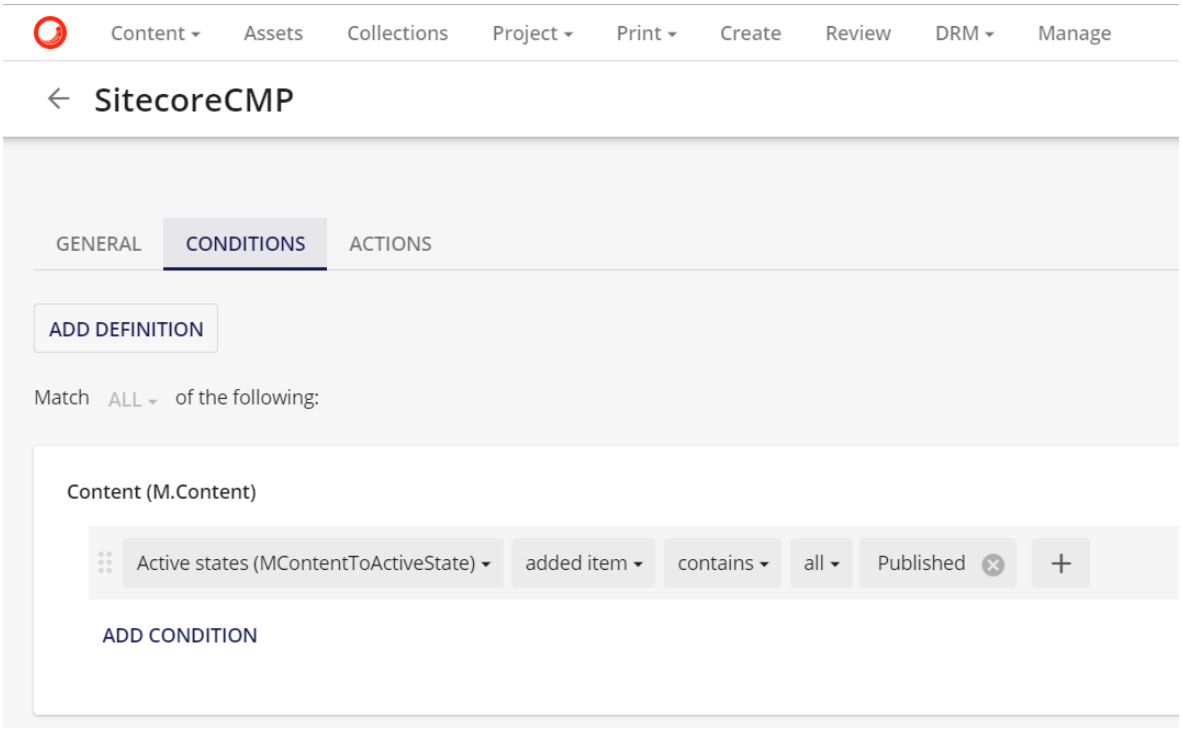
1. Open the Sitecore Content Hub. On the ribbon, click **Manage**.
2. Click **Triggers**, and click **New trigger**.
3. On the **General** tab, enter the following values:

Field	Value
Enter a name	*CMP content modification*
Enter a description	*Post to Azure Service Bus after content change*
Objective	*Entity modification*
Execution type	*In background*

4. On the **Conditions** tab, click **Add definition**, and in the drop-down list, click *Content*. This adds a new item to the list of conditions.

© Copyright 2021, Sitecore® - all rights reserved.

5. Add the following definitions:



This triggers the content import when an item is moved to the *Approved for publication* workflow state.

> **NOTE**
> You can add multiple definition conditions to the trigger. For a complete overview of capabilities, see the Sitecore Content Hub documentation.

6. On the **Actions** tab, add the action you created previously (*CMP content modification*).
7. Click **Save and close**. Confirm to activate the trigger.

> **NOTE**
> You can also activate the trigger after you have completed all the configuration settings in Sitecore Content Hub and in Sitecore. At this point, the trigger might start running before the configuration is completed.

8.1.5. Configure content and mappings

To import content, you must first create these items:

- Bucket(s) - specify where to store imported content.
- Template(s) - define the structure of imported content items.
- Entity mapping item(s) and Field mapping item(s) - determine how imported content is imported and where it is stored.

You can optionally add these items to control how relation content from Content Hub is processed:

© Copyright 2021, Sitecore® - all rights reserved.

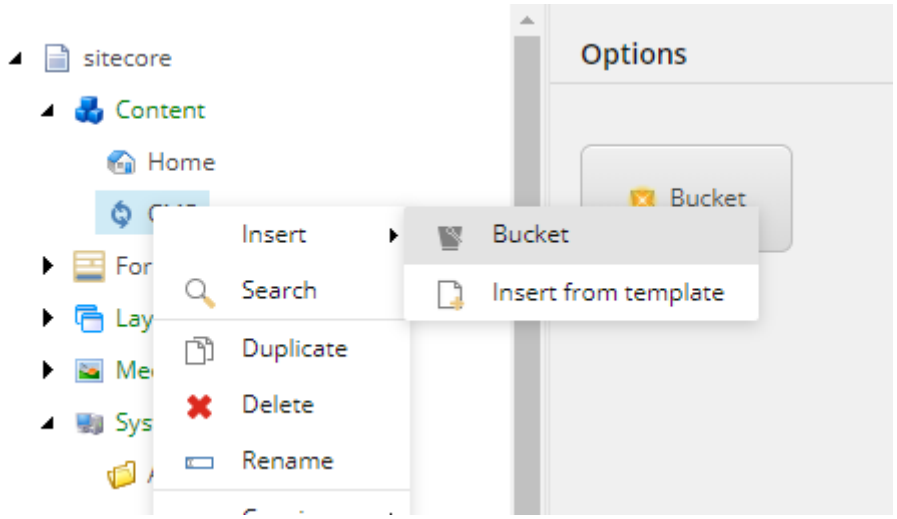
- Relation field mapping item(s) - map relation content to Sitecore fields.
- Tag field mapping item(s) - map relation content to Sitecore tags.
- Custom taxonomy provider - by default, tag field mapping stores the relation content value as a tag in Sitecore tag repositories. You can change this by specifying a custom taxonomy provider.
- Related entity mapping item(s) - link different Content Hub asset types in Sitecore.

Create a bucket

All the imported content is stored in Sitecore buckets. For example, all blog posts are stored in a bucket called *Blogs*.

To create a bucket:

1. Right-click the **CMP** folder, click **Insert**, and click **Bucket**.



2. Name the bucket, for example, *Article*, and click **OK**.

Create a template

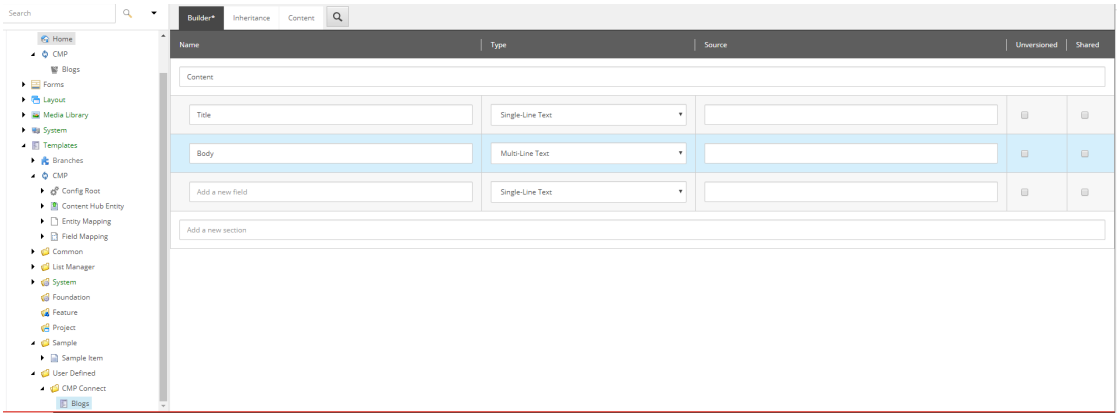
The structure of the imported content is defined by the Sitecore templates. You must make sure that each template inherits the *Content Hub Entity* template. This template contains the **EntityIdentifier** field.

To create a template:

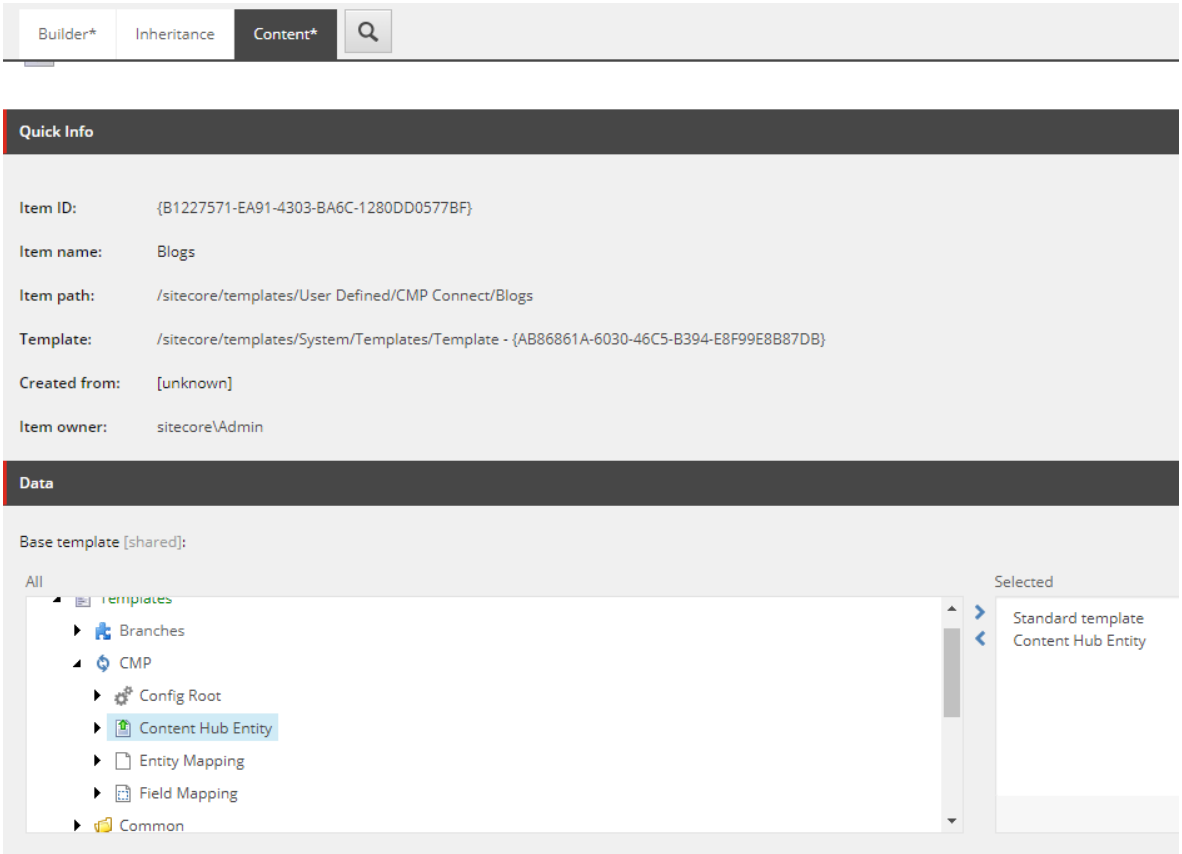
1. Navigate to a *CMP* folder, for example, `templates\User Defined\CMP Connect`. Create a template and name it, for example, *Blog*.

© Copyright 2021, Sitecore® - all rights reserved.

2. Add all the necessary fields. For example, for a blog post, add a **Title** Single-Line Text field and a **Body** Multi-Line Text field.



3. In the **Data** section, in the **Base template** field, click **Content Hub Entity** and then click the right arrow to add it to the **Selected** list.



4. In the menu bar, on the **View** tab, select **Standard Fields** and **Buckets**.
5. In the menu bar, on the **Builder Options** tab, select **Standard Values**.
6. On the **Content** tab, in the **Item Buckets** section, select **Bucketable**. Save the item.

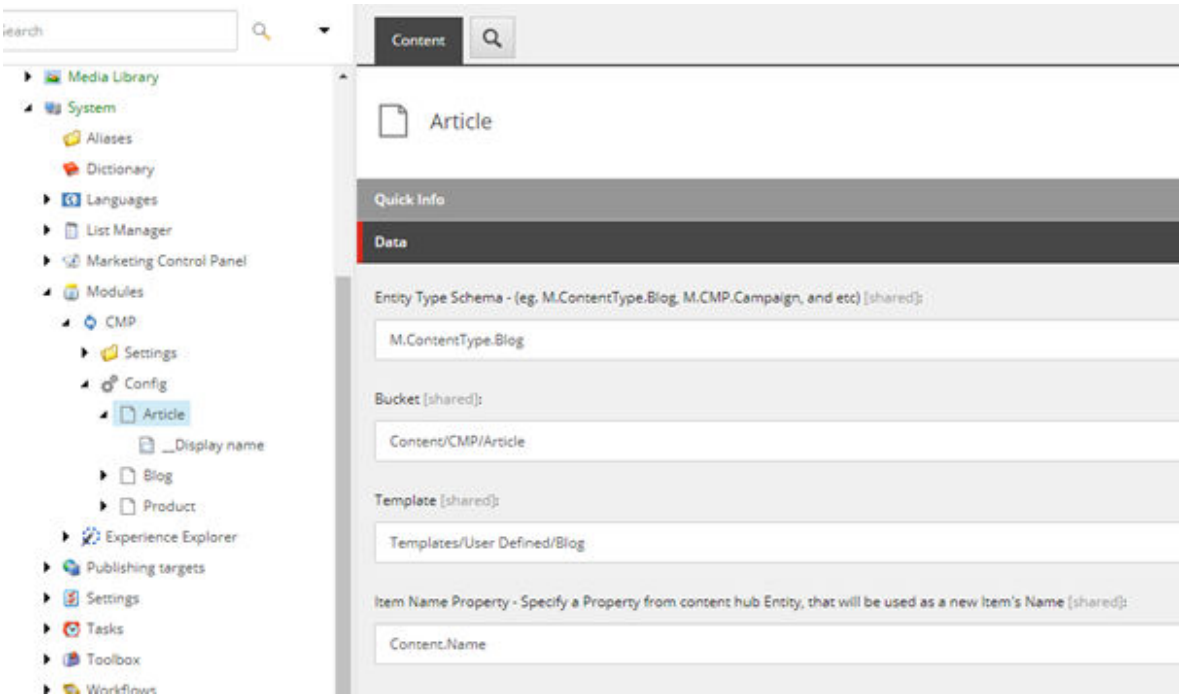
Create an entity mapping

Entity Mapping items and *Field Mapping* items define how content is imported and in what bucket it is stored.

© Copyright 2021, Sitecore® - all rights reserved.

To create an entity mapping item:

1. Navigate to `/sitecore/system/Modules/CMP`, right-click **Config**, click **Insert**, and click **Entity Mapping**. Enter the name for the new item, for example *Article*. By default, every entity mapping comes with a default field mapper for the display name.
2. Under the Article entity mapping, select `/__Display name`. On the **Content** tab, in the **CMP Field Name** field, enter a valid **Property** field name from the entity type, or schema, in Sitecore Content Hub. Sitecore uses this name as the display name.
3. Save the item.
4. In the content tree, navigate to the `Article` entity mapping.



5. On the **Content** tab, in the **Data** section, fill in the following fields:

Field	Content
Entity Type Schema	Enter the entity type used in Sitecore Content Hub
Bucket	Enter the item bucket to keep Sitecore items created from M-Entities of a given content type ID
Template	Enter a template to be used to create the Sitecore items. The template must contain the fields that are used to store the imported content
Item Name Property	Enter a valid **Property** field value of the entity type as Sitecore item name

6. Save the item.

Create a field mapping

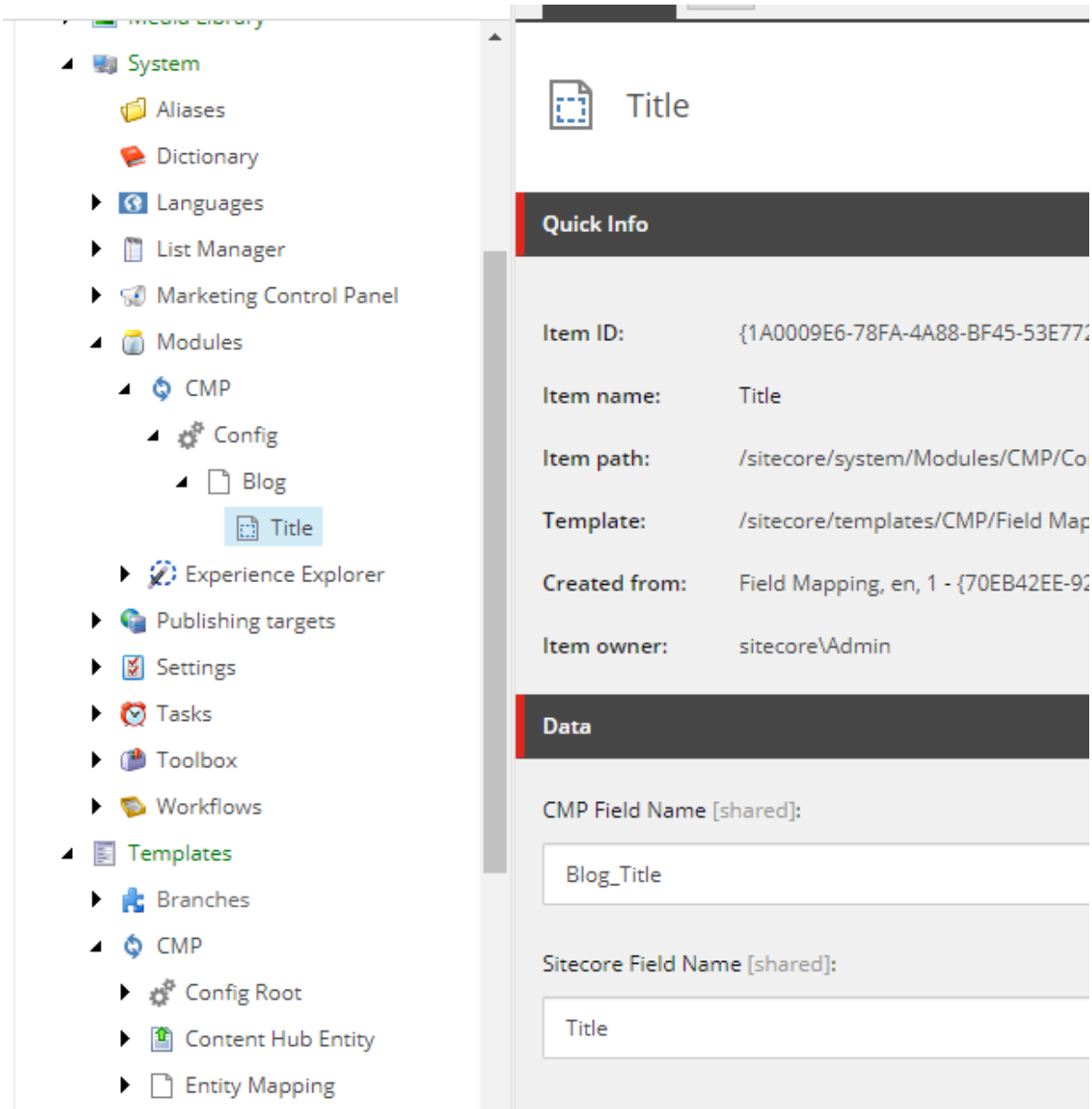
To map imported content to the Sitecore item structure, you must add *Field Mapping* items to each *Entity Mapping* item. You must create a *Field Mapping* item for each field that you want to import.

© Copyright 2021, Sitecore® - all rights reserved.

To create a *Field Mapping* item:

1. Right-click the *Entity Mapping* item that you created earlier, click **Insert**, and then click **Field Mapping**.
2. Enter a name, for example, *Title*, and click **OK**.
3. On the **Content** tab, in the **Data** section, fill in the following fields:

Field	Content
CMP Field Name	The name of the field in the Sitecore Content Hub
Sitecore Field Name	The field name as defined in the template you are using



You can add as many *Field Mapping* items as you require.

To verify the entity type and structure in the Sitecore Content Hub:

© Copyright 2021, Sitecore® - all rights reserved.

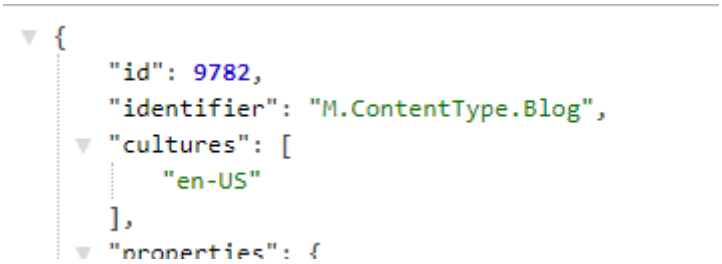
1. Create a new item in the Sitecore Content Hub or edit an existing one, for example, a blog post.
2. Copy the URL, for example, `https://yourdomain/en-us/content/detail/9782`, and paste it in a new browser tab.

> **NOTE**
> In the example URLs, replace *9782* with the ID of the entity item you have selected.

3. Change the URL to `https://yourdomain/api/entities/9782` and press **Enter**.
   The entity structure is now shown in JSON.
4. To find the entity's content type, find the **ContentTypeToContent** relation and follow the link (`"href": "https://yourdomain/api/entities/9782"`).



   Now the content type is shown, in this example `M.ContentType.Blog`.



Create a relation field mapping

Relation field mapping lets you map the taxonomy fields from any content in Sitecore Content Hub to a Sitecore field. If the content belongs to multiple taxonomy values, the mapping separates the values using the configured delimiter. The default delimiter is a comma (,).

To create a relation field mapping item:

1. Right-click the *Entity Mapping* item that you created earlier, click **Insert**, and then click **Relation Field Mapping**.
2. Enter a name, for example, *Audiences*, and click **OK** .

© Copyright 2021, Sitecore® - all rights reserved.

3. In the **Data** section, fill in the following fields:

Field	Content
CMP Relation	The relation name of the Sitecore Content Hub entity.
CMP Field Name	The name of the property in the Sitecore Content Hub relation entity.
Sitecore Field Name	The name of the content field where Sitecore stores the taxonomy values, as defined in the template you are using.

Change the relation field mapping delimiter

You can change the delimiter Sitecore uses to separate values when it stores multiple taxonomy values.

To change the delimiter:

1. On your Sitecore instance, open the `App_Config\Modules\Sitecore.Connector.CMP.config` file.
2. In the `configuration/sitecore/settings` node, set the value of `Sitecore.RelationFieldMappingSeparator` to the delimiter you want to use. For example, `<setting name="Sitecore.RelationFieldMappingSeparator" value="," />`.

Create a tag field mapping

Tag field mapping lets you map the taxonomy fields from any content in Content Hub to the tag repositories in Sitecore. If the content belongs to multiple taxonomy values, the mapping separates the values using the configured delimiter.

To create a tag field mapping:

1. Right-click the *Entity Mapping* item that you created earlier, click **Insert**, and then click **Tag Field Mapping**.
2. Enter a name, for example, *Tag Taxonomy*. Click **OK**.
3. In the **Data** section, fill in the following fields:

Field	Content
CMP Relation	The relation name of the Sitecore Content Hub entity.
CMP Field Name	The name of the property in the Sitecore Content Hub relation entity.
Enable / Disable to store Non-Leaf Node	Choose one of the following: • Enable Non-Leaf Node (Default) • Disable Non-Leaf Node
Store Tag Name Option As	Choose one of the following: • Full Path Name (Default) • Node Name

© Copyright 2021, Sitecore® - all rights reserved.

Field	Content
Custom Taxonomy Provider Name	The name of a Sitecore custom taxonomy provider. If you select a custom taxonomy provider, Sitecore stores the taxonomy in the custom provider instead of in the default taxonomy provider. Leave this field blank to use the default taxonomy provider. **NOTE** See Build a custom taxonomy provider in the Sitecore developer documentation for information on how to create custom taxonomy providers.

For more information on how the tag field mapping settings work, refer to the Sitecore developer documentation.

Change the tag field mapping delimiter

You can change the delimiter Sitecore uses to separate values when it stores multiple values in a tag.

To change the tag field mapping delimiter:

1. On your Sitecore instance, open the `App_Config\Modules\Sitecore.Connector.CMP.config` file.
2. In the `configuration/sitecore/settings` node, change the setting `Sitecore.NodeNameSeparator` to the delimiter you want to use. For example, `<setting name="Sitecore.NodeNameSeparator" value="." \>`.
3. If you use special characters as a separator, you must allow those characters on the Sitecore level. You do this by using a patch file to change the `ItemNameValidation` setting. For example:

```
<configuration xmlns:patch="http://www.sitecore.net/xmlconfig/">
    <sitecore>
        <settings>
            <setting name="ItemNameValidation">
                <patch:attribute name="value">^.*$</patch:attribute>
            </setting>
        </settings>
    </sitecore>
</configuration>
```

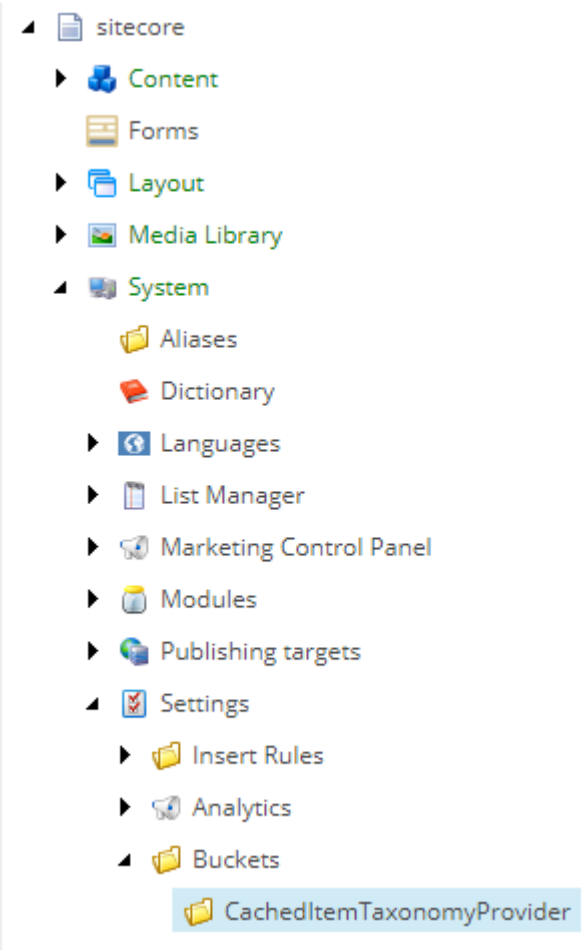
IMPORTANT

This example allows *all* characters to be used. It is best practice to only allow the characters you need.

© Copyright 2021, Sitecore® - all rights reserved.

8.1.6. Change the taxonomy provider

When you set up the retrieval of taxonomy fields from any content in Sitecore Content Hub, and synchronize the field values to the tag repositories in Sitecore, by default, Sitecore stores the taxonomy in `/sitecore/System/Settings/Buckets/CachedItemTaxonomyProvider`:



If you have created a custom taxonomy provider, you can store the taxonomy there instead of in the default provider.

To use a custom taxonomy provider:

1. On your Sitecore instance, open the `[your Sitecore Instance] \App_Config\Modules\Sitecore.Connector.CMP\ Sitecore.Connector.CMP.config` file.
2. Change the `Sitecore.DefaultTaxonomyProvider` setting to the name of your custom taxonomy provider. For example:

```
<setting name="Sitecore.DefaultTaxonomyProvider" value="MyCustomTaxonomyProvider" />
```

> **NOTE**
> Do not leave the `Sitecore.DefaultTaxonomyProvider` setting empty.

© Copyright 2021, Sitecore® - all rights reserved.

3. Save the config file.

8.2. Configure the Sitecore DAM part of Sitecore Connect for Content Hub

The Sitecore DAM part of Sitecore Connect for Content Hub enables you to browse, search, and insert digital assets from Sitecore DAM directly from within Sitecore's Content and Experience editing.

> **NOTE**
> The Sitecore DAM part of the connector does not support *multilanguage* fields when inserting assets in Sitecore.

This section describes the configuration steps you must perform on your Sitecore DAM and Sitecore instances:

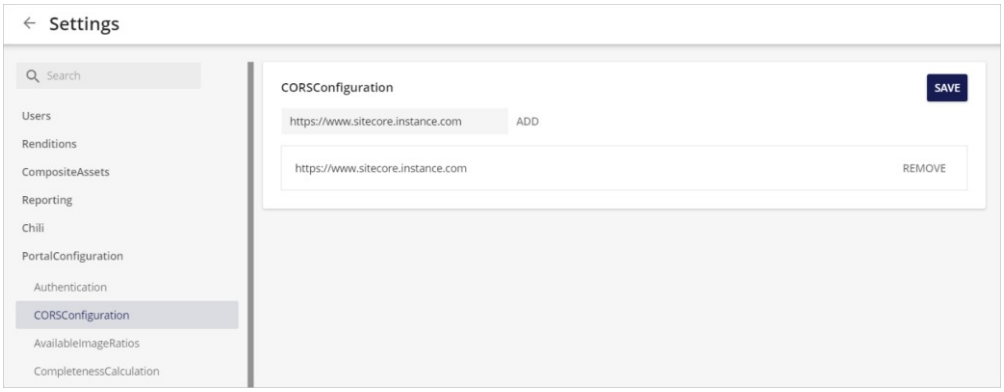
- Configure CORS
- Configure redirection settings
- Configure the DAM connector in Sitecore

8.2.1. Configure CORS

You must configure the CORS for Sitecore DAM to allow incoming traffic from the Sitecore instance.

To configure the CORS:

1. On the Sitecore DAM main menu, click **Manage**.
2. Click **Settings**.
3. On the **Settings** page, click **PortalConfiguration**, then click **CORSConfiguration**.
4. Click **Add**, and add the URL for your Sitecore instance, for example, `https://www.sitecore.instance.com`.



5. Save the changes.

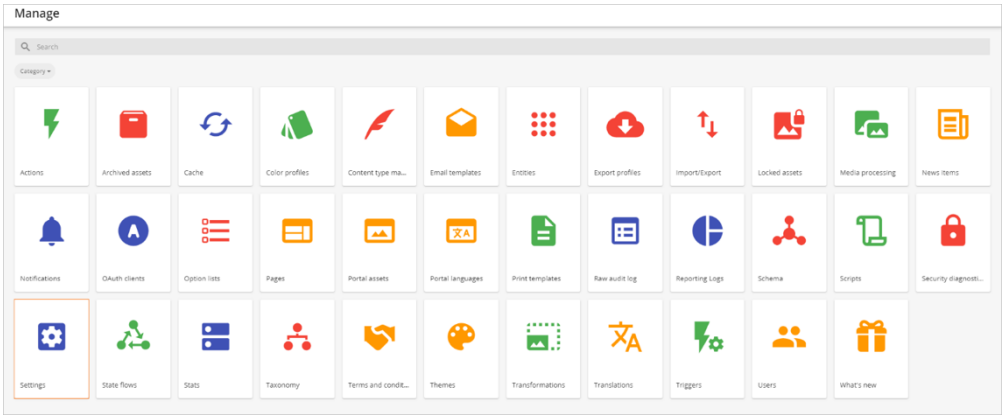
© Copyright 2021, Sitecore® - all rights reserved.

8.2.2. Configure redirection settings

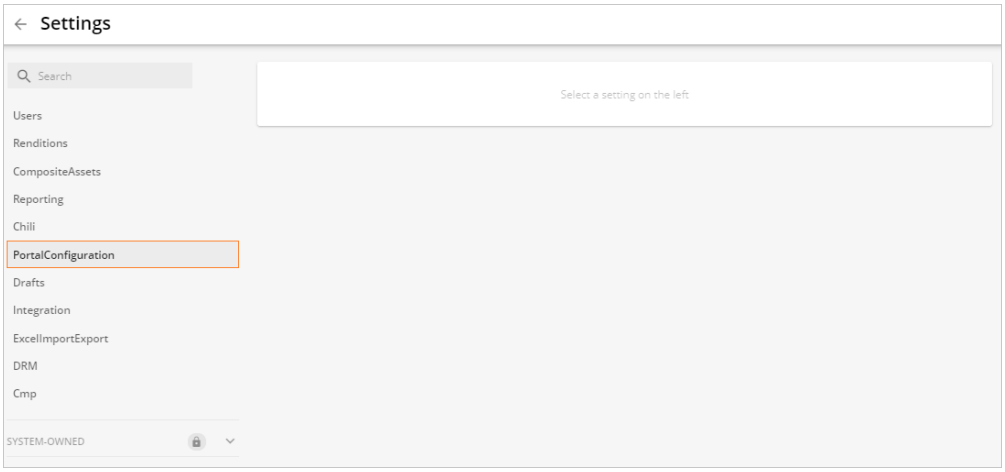
You must configure the redirection setting within the **PortalConfiguration** of Sitecore DAM to allow the Sitecore instance to access/display the Sitecore DAM.

To configure the redirection settings:

1. In the Sitecore DAM main menu, click **Manage**.
2. Click **Settings**.



3. On the **Settings** page, click **PortalConfiguration**, then click **Authentication**.



4. In the drop-down menu, click **Text**.
5. In the `Providers` node, for all SSP providers, set the `authentication_mode` value to *Passive*.

> **NOTE**
> If you have not configured SSO, you can skip this step, as `authentication_mode` in that case is *Passive* by default.

For example:

```
"sitecore":
[
    {
        "authentication_mode": "Passive",
        "client_id": "<ClientId>",
```

© Copyright 2021, Sitecore® - all rights reserved.

```
        "client_secret": "<ClientSecret>",
        "email_claim_type": "<ClaimTypeOverride>",
        "external_user_creation_url": "https://www.registerme.com",
        "identity_server_url": "<Url>",
        "is_enabled": true,
        "messages": {
            "signIn": "translationKey",
            "signInDescription": "translationKey",
            "signInTitle": "translationKey"
        },

        "provider_name": "Sitecore,",
        "username_claim_type": "<ClaimTypeOverride>"
    }
]
```

NOTE

For Content Hub 3.4.0 and higher, the translation keys provided in the `messages` configuration for the login screen must be provided without the `T_` prefix that was used in previous versions. For example, `"signIn": "T_translationKey"` must be `"signIn": "translationKey"`.

6. Save the changes.

8.2.3. Configure the DAM connector in Sitecore

In your Sitecore instance,

The Sitecore Connect for Content Hub (SCCH) installation adds the *Insert from Sitecore DAM* item to the Rich Text Default profile for the Rich Text Editor. In order to enable the **Insert from Sitecore DAM** function, you must configure a connection to the Sitecore DAM part of the SCCH connector.

To configure the connection to Sitecore DAM in Sitecore:

1. Open the `C:\inetpub\<site>\App_Config\ConnectionStrings.config` file, and add the following connection strings:

```
<add name="DAM.ContentHub"
connectionString="ClientId={client_id};ClientSecret={client_secret};UserName={username};Pas
sword={password};URI={uri};" />
<add name="DAM.SearchPage" connectionString="<Sitecore Content Hub search page URI>" />
<add name="DAM.ExternalRedirectKey" connectionString="Sitecore" />
```

Substitute the placeholders in the example with the values from this table:

Placeholder	Value
Sitecore Content Hub URI	the URI for your Sitecore Content Hub DAM instance, for example, `https://dam-connector.stylelabs.qa.com`.
Sitecore Content Hub search page URI	the URI for the page that opens the DAM in an IFrame. For example, `<Sitecore Content Hub DAM URI>/en-us/sitecore-dam-connect/approved-assets`.

NOTE

The user account using the connector must have read access to the connector pages in Sitecore DAM.

© Copyright 2021, Sitecore® - all rights reserved.

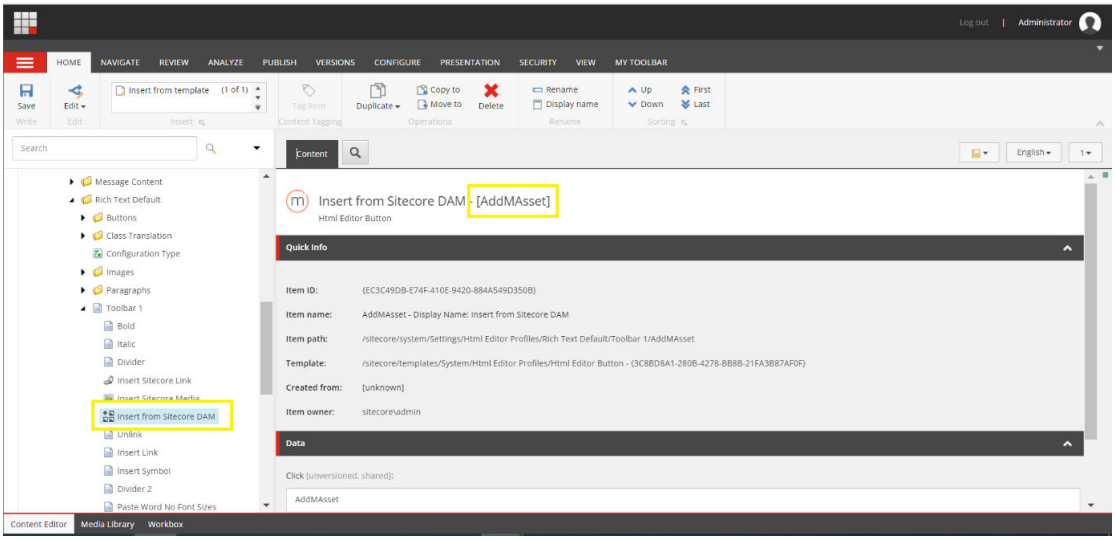
2. Save the changes.
3. In the root folder (by default, `wwwroot`), open the `web.config` file. Make sure the URL of the Content Hub instance (for example, `content-hub-url.stylelabs.com`), other delivery hostnames (for example, `content-hub-url-delivery.stylelabs.cloud`), and generated public links (for example, `content-hub-url.stylelabs.com:8686`) are added as sources to the `Content-Security-Policy` tag. For example:

```
<add name="Content-Security-Policy" value="default-src 'self' 'unsafe-inline' 'unsafe-
eval' https://apps.sitecore.net;
    img-src 'self' data https://content-hub-url.stylelabs.com/ https://content-hub-url-
delivery.stylelabs.cloud/;
    style-src 'self' 'unsafe-inline' https://fonts.googleapis.com;
    font-src 'self' 'unsafe-inline' https://fonts.gstatic.com;
    upgrade-insecure-requests; block-all-mixed-content;
    child-src 'self' https://content-hub-url.stylelabs.com/ https://content-hub-url-
delivery.stylelabs.cloud/;
    connect-src 'self' https://content-hub-url.stylelabs.com/ https://content-hub-url-
delivery.stylelabs.cloud/;
    media-src https://content-hub-url.stylelabs.com/ https://content-hub-url-
delivery.stylelabs.cloud/;" />
```

> **NOTE**
> In Content Hub, you can configure different delivery hostnames for public links by using the `AssetExtensionDataConfigurationSetting` setting.
>
> If you have configured different hostnames, you must add all the hostnames as sources. You do this in the `Content-Security-Policy` parameter, in the `img-src`, `child-src` and `connect-src` properties.

4. The **Insert from Sitecore DAM** icon  is now available in the Rich Text Editor for fields that use the Rich Text Default profile.
5. If you need to access DAM assets in other profiles than Rich Text Default:
   - In the Core database, navigate to */sitecore/system/Settings/Html Editor Profiles/Rich Text Default/Toolbar 1*.
   - Copy the *Insert from Sitecore DAM* item.

© Copyright 2021, Sitecore® - all rights reserved.



- Navigate to the Rich Text Editor profile you intend to use, and paste the *Insert from Sitecore DAM* item.

© Copyright 2021, Sitecore® - all rights reserved.

9. Problem solving

This section contains ways to solve problems you might encounter in the Content Hub connector.

9.1. General troubleshooting help

To help find solutions for errors or problems in general, you can do the following:

- Check the log in the Sitecore Content Hub for the created trigger. Trigger the event that fires the trigger and then check the log as follows:
  - On the menu ribbon, click **Manage** .
  - Click **Actions**.
  - Click the **Auditing** tab.
- Check posted messages in the Azure Service Bus.
- Check the Sitecore log file and scan for any CMP or DAM related errors.

9.2. In Azure, browsing a Sitecore DAM link displays a 404 error

You browse a Sitecore DAM image field type link, and you get a *404 not found* error.

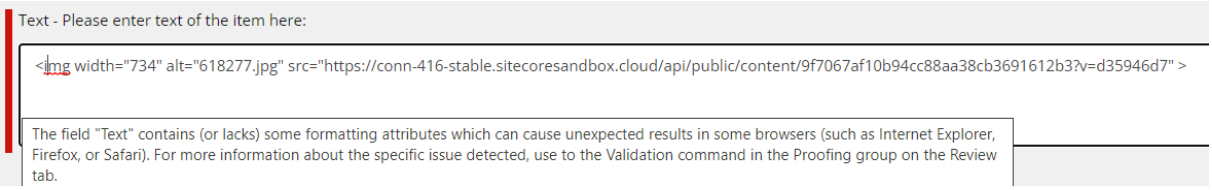
To solve this problem:

1. In the Sitecore Launchpad, click **Desktop**, and switch from *Master* database to *Core* database.
2. In the Content editor, navigate to *sitecore/system/Field types/Simple types/Image*.
3. On the **Content** tab, in the **Data** section, clear the **Control** field. Save the item.

9.3. The rich text editor in Sitecore shows an XHTML related validation error

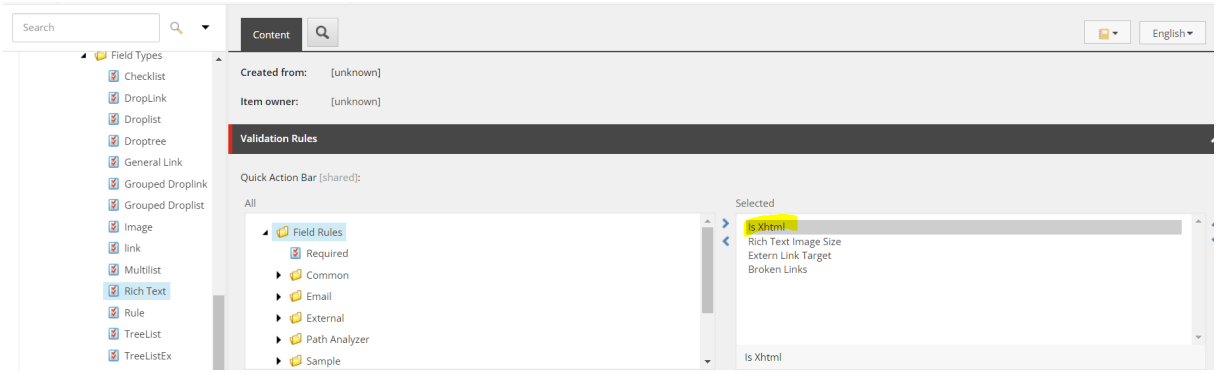
The connector does not change any data values or formats when it syncs data from the Content Hub rich text editor field to the Sitecore rich text editor. The XHTML validation in the Sitecore rich text editor may highlight the error:

© Copyright 2021, Sitecore® - all rights reserved.



To solve this problem:

1. In the Content editor, look for the following items:
   - */sitecore/system/Settings/Validation Rules/Field Types/Rich Text*
   - */sitecore/system/Settings/Validation Rules/Global Rules*
2. For all validation rules under these two items, in the **Selected** list, click **Is Xhtml** to move it to the **All** list.



3. Save the item.

9.4. Azure deployment throws Could not load file or assembly 'Microsoft.Extensions.Caching.Abstractions' error

The Sitecore Azure ToolKit (SAT) requires minimum dependency to perform the config transformation. After you have applied the SCCH package to the instance, the `bin/scch` folder has all the necessary assemblies.

To solve the problem:

1. In the Azure App Service Editor, open the `Web.config` file.
2. In the `assemblyBinding` node, add the following code:

```
<dependentAssembly>
    <assemblyIdentity name="Microsoft.Extensions.Caching.Abstractions"
publicKeyToken="adb9793829ddae60" />
    <bindingRedirect oldVersion="0.0.0.0-3.1.14.0" newVersion="3.1.14.0" />
    <codeBase version="2.1.2.0" href="bin/Microsoft.Extensions.Caching.Abstractions.dll" />
    <codeBase version="3.1.5.0" href="bin/Microsoft.Extensions.Caching.Abstractions.dll" />
    <codeBase version="3.1.14.0" href="bin/scch/
```

© Copyright 2021, Sitecore® - all rights reserved.

```
Microsoft.Extensions.Caching.Abstractions.dll" />
</dependentAssembly>
```

3. Save and refresh the Azure App Service Editor.

9.5. On-prem deployment throws Method does not have an implementation error

The Content Hub SDK in SCCH 5.0 requires `Microsoft.Extensions.DependencyInjection` version 3.1.14.0. For deployments on Sitecore 10.1.x and earlier that reference earlier versions, you must do one of the following:

- Update and rebuild any custom assemblies to refer to version 3.1.14.0.
- Add a redirect binding configuration in the `web.config` file.

© Copyright 2021, Sitecore® - all rights reserved.