EPiSERVER CMS 6

Version 6 R2

Administrator's Manual

EPiSERVER

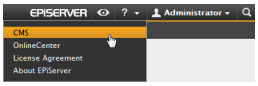
Table of Contents

About this Documentation

This documentation is intended for users of EPiServer CMS and EPiServer OnlineCenter according to role.

How to Access EPiServer Help System

EPiServer CMS features a web help system. The entire web help content can be accessed from the main OnlineCenter menu, which is always visible at the top of the EPiServer CMS website. Click the **?** icon in the top menu and select the product for which you want to view the Help. Browse or search for the topic where you need guidance.



In addition to this, context-sensitive help is also available inside EPiServer CMS. Click the **Help** button where available in Edit or Admin mode, and the Help window will open.

You can also access the web help by browsing to webhelp.episerver.com.

Expected Knowledge

As a reader of this documentation, you are assumed to have good understanding of the following:

- The standard Windows environment
- Basic word processing skills (no specific web publishing or HTML knowledge is needed)

References

The following parts are listed as references and will not be described in this documentation:

- How to set up tabs, zones and gadgets, see EPiServer OnlineCenter Guide.

Online Community on EPiServer World

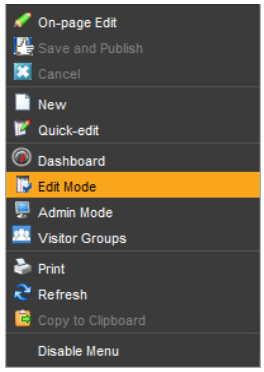
EPiServer World is an online community where you can find the latest product information. It is open to the public, for partners, customers, and everyone working with EPiServer products, such as editors, webmasters, site owners and developers. Here you can download material, participate in discussions, read articles, receive support and much more. Feel free to sign up as a member.

Copyright Notice

Introduction

Logging In

If you are automatically logged in on the website or logged in as a visitor or editor, you can access Admin mode from both View mode and Edit mode. The easiest way to move between the different modes is to use the right mouse button. When you right-click on the page Admin mode is shown in the menu that is displayed, if you have authorization for Admin mode. If you are in Edit Mode, you can also go to Admin mode by clicking the **Admin Mode** button in the toolbar.

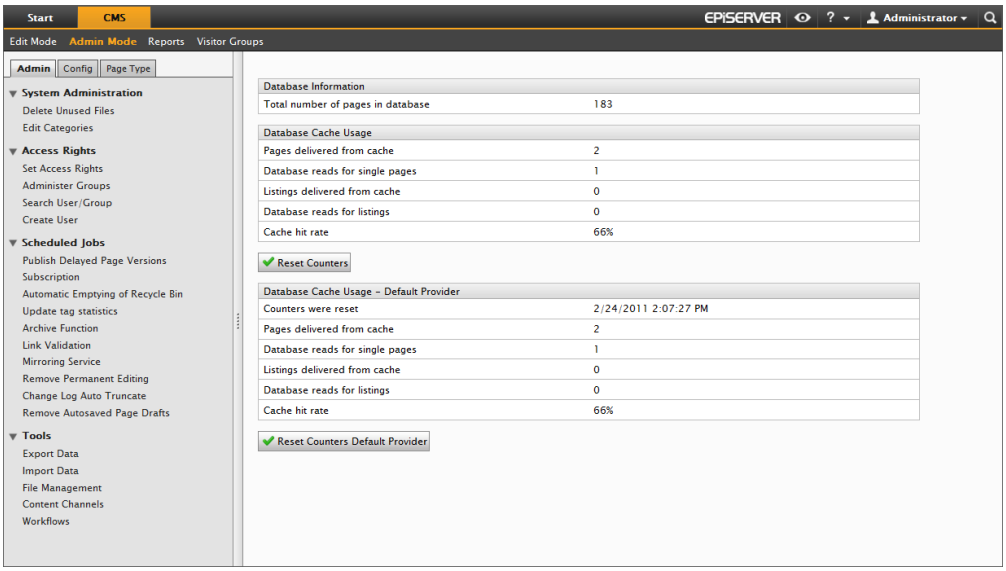


It is also possible to switch to Admin mode by entering a certain address in the address field in your web browser. The address varies for each website. Check with your supplier or whoever installed and configured EPiServer CMS for information about this.

Administering the visual appearance, changing language in Edit and Admin modes and certain other settings, are done directly via changes to files on the web server. "Files for EPiServer CMS" on page 77.

The Window

Once you have logged on to Admin mode, the first page of Admin mode will be displayed. It contains some brief information about the website. You can see how many pages you have in the website, how many subscribers etc.



The following tabs with different functions, organized according to function use, are shown to the left:

- **Admin** has functions that often concern the daily work of administering an EPiServer CMS website.

- **Config** contains functions that are often set up at installation; these settings are rarely changed.
- **Page Type** contains all the page types that the editors create pages in. The page types function is widely used by developers when developing new templates.

The different tabs are used in different ways depending on the role you have.

Above the tabs is a toolbar, which take you to other modes in EPiServer CMS.

The Start Page in Admin Mode

The start page in Admin mode contains the following information:

Database Information

- **Total number of pages in database** shows the total number of pages stored in the database.

Database Cache Usage

- **Pages delivered from cache** shows the number of pages that was delivered from cache by all providers.
- **Database reads for single pages** shows the number of times single pages was fetched from the database by all providers.
- **Listings delivered from cache** shows the number of times page listings was delivered from cache by all providers.
- **Database reads for listings** shows the number of times page listings was fetched from the database by all providers.
- **Cache hit rate** shows in percent how many pages that was fetched from cache by all providers.

You can also see the latest date and time for reset of the cache counting.

Click **Reset Counters** if you want the counting to start over from zero.

Database Cache Usage - Default Provider

See the description as for **Database Cache Usage**, here applicable to the default provider. Click **Reset Counters Deafault Provider** if you want the counting to start over from zero.

If you use several providers, they will be listed below.

Working with Authorization

EPiServer CMS offers opportunities for using access rights to allow visitors access to different pages of information. EPiServer is structured in such a way that a visitor without the right to view a menu or particular object in a menu, will not see that the object exists. Visitors can only see those pages to which they have the access rights. You can also use the authorization system to control which editors can do what on your website.

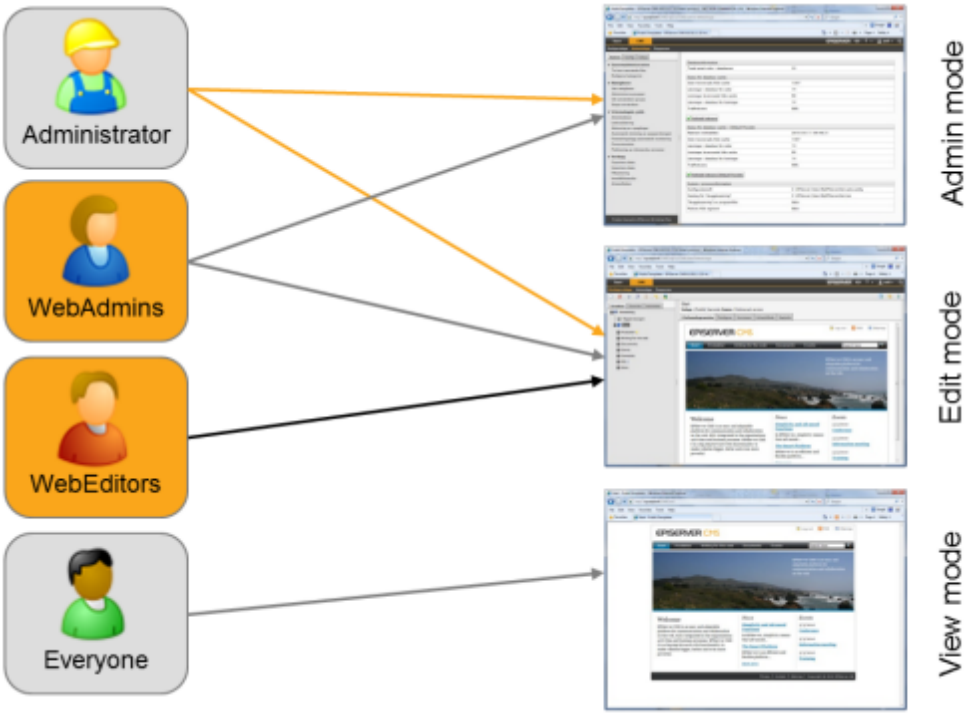
It is usual to build up access rights based on different groups and not on individuals. It is the group that has specific access rights, the individuals are only members of different groups. A group is created for each specific place (page) in the structure where the access rights must differ from other pages. There are also different groups if you wish to have different roles for editors within a certain structure. It is technically possible to give specific access rights to individuals but in practice it is more complicated to administer.

The file **web.config** contains many settings for the website. Changing the wrong thing in this file can have consequences for the entire website and for this reason it is not appropriate to give a lot of people access rights to make changes to this file. Many see being able to change the file as a must if you have a number of editor groups.

Access rights in EPiServer CMS have the following dimensions:

- An individual/group has access rights to different modes in EPiServer CMS (View, Edit, Visitor groups and Admin).
- An individual/group has access rights to do different things within the structure of the website. In general, the contents structure is used as the basis for setting access rights for who may do what. It is a great advantage if you plan your structure before you start to assign access rights to the contents of EPiServer CMS.

Providing Access Rights to Different Views



When you are working with authorizations in EPiServer CMS, you see them in two dimensions. Each individual/group can have access rights to edit content on one or more pages. This can be carried out from two different views in EPiServer, View mode and Edit mode. Special settings are required for an individual/group to have access rights to Edit mode. All visitors normally have access to View mode (the Everyone group). If that group has the access rights to create information in the structure, anyone will be able to create pages in that place. This is used, for example, in public discussion forums..

You specify which groups are to have access to the different modes in EPiServer in the file web.config. The file can be found in the root of the website. The group Administrators and WebAdmins have access rights to Admin mode by default. Administrators, WebAdmins and WebEditors have access rights to Edit mode. All groups normally have access to View mode.

You can also provide stronger access to editors who will work with personalized content on the website and need to administer visitor groups. This is done by creating the default group VisitorGroupAdmins.

Creating User Groups in a Standard Installation

The following groups as default have different roles and tasks on the website:

- EPiServer CMS uses the **WebEditors**, **WebAdmins** and **VisitorGroupAdmins** groups by default.
- Windows uses also the **Administrators** and **Everyone** by default that are relevant to EPiServer.

Group	*Function*
Everyone	Anonymous visitors to the website. All visitors to a public website are essentially anonymous. This means that the system cannot identify the visitor. With an intranet,

Group	*Function*
	it is possible to link the intranet to a login in Windows to allow visitors to be identified. In this case, no visitor is essentially anonymous.
WebEditors	Website editors. They have access to Edit mode by typing the address that has been selected for Edit mode, and then logging on with a user name and password. Website editors do not have access to Admin mode by default.
WebAdmins	Website administrators. They have access to Admin mode by typing the address that has been selected for Admin mode and then logging on with a user name and password. Website administrators also have access to Edit mode by default.
Administrators	Website super administrators. To be able to change system settings in EPiServer CMS, you need to be a member of this group. Administrators have access rights to both Edit and Admin modes by default.
VisitorGroup Admins	Website administrators of the Visitor Group area. For example, editors who are working with personalized content and need to create, change and delete visitor groups. Note that these administrators do not have access rights to Admin mode.

Note that the WebEditors, WebAdmins and VisitorGroupAdmins groups have not been automatically created in a standard installation. When you have selected providers for your website, you must begin by creating these two groups there. If you have not made any changes to the standard installation, Windows is the default provider. This means that you must start by creating the groups in Windows, see Using Different Providers.

Providing Access Rights to Different Modes

If you do not want to use the default groups available in EPiServer CMS for accessing Edit and Admin modes, you can amend this in the **web.config** file found in the root of your website.

1. Open the file in Notepad or a similar program. (It must be a program that supports **UTF-8**[1], otherwise the program can damage the file).
2. Look for the line containing the settings for Edit and Admin modes. This is <location path="/XX/UI/admin"> in the case of Admin mode. This is <location path="/XX/UI/"> in the case of Edit mode. In which XX can vary depending on the path entered during installation.
3. Change the names of the groups in <allow roles=" ">. Enter Computer/Domain name\Names of the groups between " ". Use a comma to separate groups.
4. Save the file.

```
<location path="EPiServer">
<system.web>
...
<authorization>
<allow roles="WebEditors, WebAdmins, Administrators" />
<deny users="*" />
</authorization>
</system.web>
...
</location>
<location path="EPiServer/CMS/admin">
<system.web>
<authorization>
```

1

```
<allow roles="WebAdmins, Administrators" />
<deny users="*" />
</authorization>
</system.web>
</location>
```

Include all editors to be members of the WebEditors Group to provide access to Edit mode. Permit all editors to be members of two groups to avoid too many changes in web.config. One group that gives them access to Edit mode and another group for the place in the tree structure that they are authorized to edit.

See also "Providing Access Rights for Different Functions" on the facing page.

Using Different Providers

In EPiServer CMS it is possible to work with different sources for storing users and groups, which is a standard technology from Microsoft, called **Providers**.

It is possible to store users and groups in a database, which may either be EPiServer's database or another selectable database. It is also possible to use the user groups available in Windows or Active Directory and give access rights to the existing users and groups in Windows. You usually choose one of these options, but it is fully possible to mix the two services.

How you choose to manage your users is a decision taken centrally for the entire website. Configuration is carried out in **web.config** and is normally done by the person helping to develop the entire solution. There are two providers as standard for groups and users. There is an additional provider that allows you to use the other two providers at the same time. It is also fully possible to develop your own provider, which you do for example if you have an existing database of users and passwords.

Providers	*Function*
WindowsRoleProvider	Providers for groups that are managed in Windows or Active Directory.
SqlServerRoleProvider	Providers for groups that are managed in EPiServer's database.
MultiplexingRoleProvider	Providers for groups that are managed both in Windows and EPiServer's database.
WindowsMembershipProvider	Providers for users that are managed in Windows or Active Directory.
SqlServerMembershipProvider	Providers for users that are managed in EPiServer's database.
MultiplexingMembershipProvider	Providers for users that are managed both in Windows and EPiServer's database at the same time.

If you have installed a standard installation of EPiServer CMS and not affected the configuration of which directory service should apply for the website, Windows applies. This means that you must log in the first time with a Windows user name.

Providing Access Rights for Different Functions

Permissions for Functions

Edit mode in EPiServer CMS has several functions that allow you to limit the access for certain people or groups. The following functions can be access protected.

> The **All functions available in the EPiServer CMS 5 HTML Editor** function only applies if you are using the EPiServer CMS 5 Editor and not the TinyMCE HTML Editor. Refer to the user documentation for EPiServer CMS 5 for further information about this.

Detailed error messages for troubleshooting provides one or more groups/individuals with access to detailed error messages. In System Settings, it is possible to activate a function that provides visitors with a form to fill in whenever there is a technical error on the website. By changing the access rights here, you can specify who should receive these forms.

View and publish pages from Microsoft Office Add-in makes it possible for one or more users/groups to publish pages from Microsoft Office. This function requires that an add-in program is installed on the computer being used. Read more about this in the Editor's Manual.

Allow advanced upload of files using for example drag-and-drop makes it possible to allow a group or individual to use advanced uploading. This means, for example, that an editor can upload several files at a time by dragging and dropping files from Windows Explorer to the EPiServer CMS file manager.

Allow the user to act as a web service user allows users to act as a web service user.

Allow users to move data /pages between page providers

Adding/Modifying Permissions to a Function for a Group or User

1. Select Modify for the function you want to modify.
2. Existing groups or users with access appear in a list
3. Select **Add Users/Groups** if you want to give users or groups access to this function.
4. The groups and persons in the system appear in the window that opens. Double-click on the name to add the group or user.
5. Select **OK**.
6. The group or user appears in the list with its check box selected. This means that the group or user has access to this function.
7. Select **Save**.

Delete Permissions to a Function for a Group or User

1. Select **Edit** for the function you want to modify.
2. Existing groups or users with access appear in a list
3. Select the check box of the group or user for which you want to remove access. The check box is then cleared and the tick in the box disappears.
4. Select **Save**.

Access Rights for Page Types

It is also possible to set access rights for page types, making it impossible for certain users or groups to create pages based on a certain page type. This may be useful if, for example, not all groups should be able to create news items on the website, see Working With Page Types.

Access Level for Languages

When new languages are added, they automatically receive an access level so that the group Everyone can work with the language. Users that do not have access levels for a language will not have the language available in the drop-down list in the **Structure** tab and will not be able to edit pages in that language, see Change the Access Level for a Language.

Working with Groups and Users

When you are working with users and groups in EPiServer CMS, you have the opportunity to select which providers should store and manage the groups and users. It is possible to use existing groups from Windows in EPiServer, if you so wish. From an administrative point of view, this can save you a great deal of work. You will also have the opportunity to create your own groups in a database, for example EPiServer CMS's own database. Which of these two services you can use or if you can use both is configured in web.config.

When you select **Administer Groups** in Admin mode, all groups will be shown irrespective of the provider that is used on the website. The provider for the group in question is shown to the right of the group name.

It is common to use SQL Server provider to manage groups and users if part of your website is an extranet, or is operated by an operations supplier. In those cases the server cannot be accessed in the same way, so a simple solution for management via the web browser is required.

Creating Groups in Windows

It is common practice to use groups and users from Windows if EPiServer CMS is used for example in an intranet. All employees usually already have an account in Windows; they also have group affiliations that can be used. Usually all those visiting the intranet are automatically identified using a log in on the computer and the network.

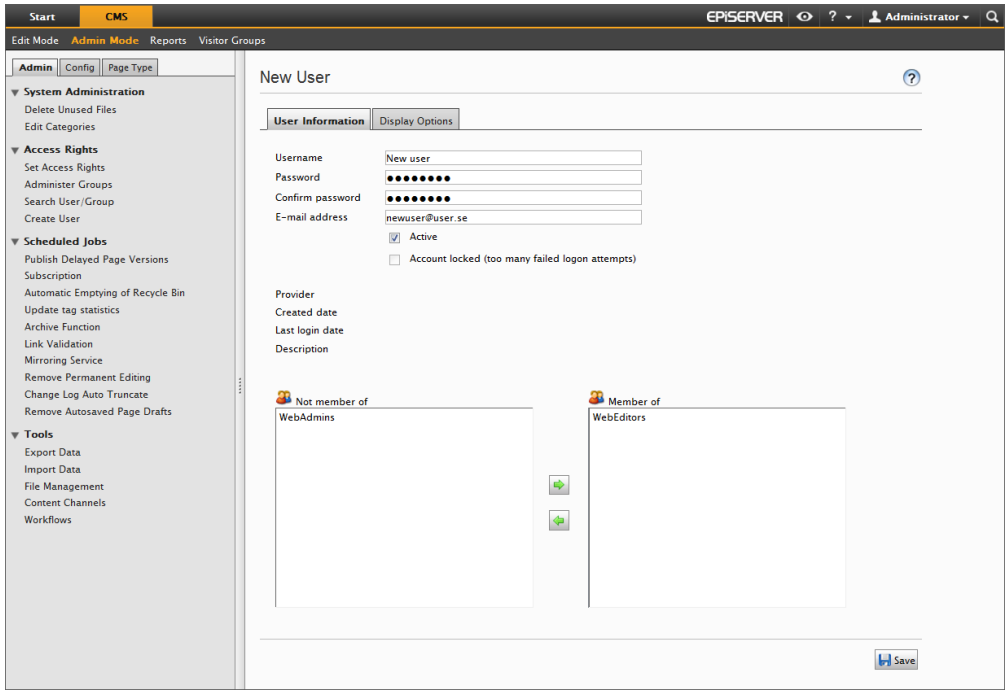
When you use groups and users from Windows in EPiServer CMS, you manage the entire administration process, including user names, passwords and group affiliation in Windows. When you then give them specific access rights in EPiServer CMS, you do this via EPiServer CMS Admin mode.

If a user has logged in to the website it is possible to find the group in EPiServer and it is thus possible to set specific access rights for the group. If you create a new group in Windows, which you want to use in EPiServer CMS you must start by logging in with one of the users. Then log in as an administrator and you will find the group in EPiServer CMS Admin mode.

Creating, Editing and Deleting Users in EPiServer CMS

If you have chosen to work with users and groups in EPiServer CMS, you must create each separate user. If you have developed such a function, it may be possible to allow visitors to create their own user accounts. Using **Create User**, you can set up users in EPiServer CMS.

1. On the **Admin** tab, select **Create User**.
2. For the account to work and the user to be able to log in, a user name, password and e-mail address must be specified and it must be specified that the account is active.
3. Select which group or groups they are to belong to.
4. Specify which system language the user should have and which buttons and icons are to be available to the editors in Edit mode in the **Display Options** tab. Editors can manage this setting themselves from Edit mode.
5. Click **Save**.



Editing Users

All users created in a database via EPiServer's Admin mode can be edited. It can be a case of changing a password or similar. To edit a user's details, search for the user under **Search User/Group** and then edit the user. All individuals with access to Edit mode can modify a number of their details under **My Settings**. You can edit more information in Admin mode, including when the account is to be active, which group the user is a member of, etc.

These properties can only be modified for users that have been created via self-registration or via **Create User**. If you want to edit this type of information when the user is created in another directory service, you must do it with this tool. From Admin mode, you can change the same information on an external user as the user themselves can do via Edit mode.

Edit a user as follows:

1. Select **Search User/Group**.
2. Select the user you want to edit by clicking the username.
3. Make your changes in the form.
4. Click **Save**.

Deleting Users

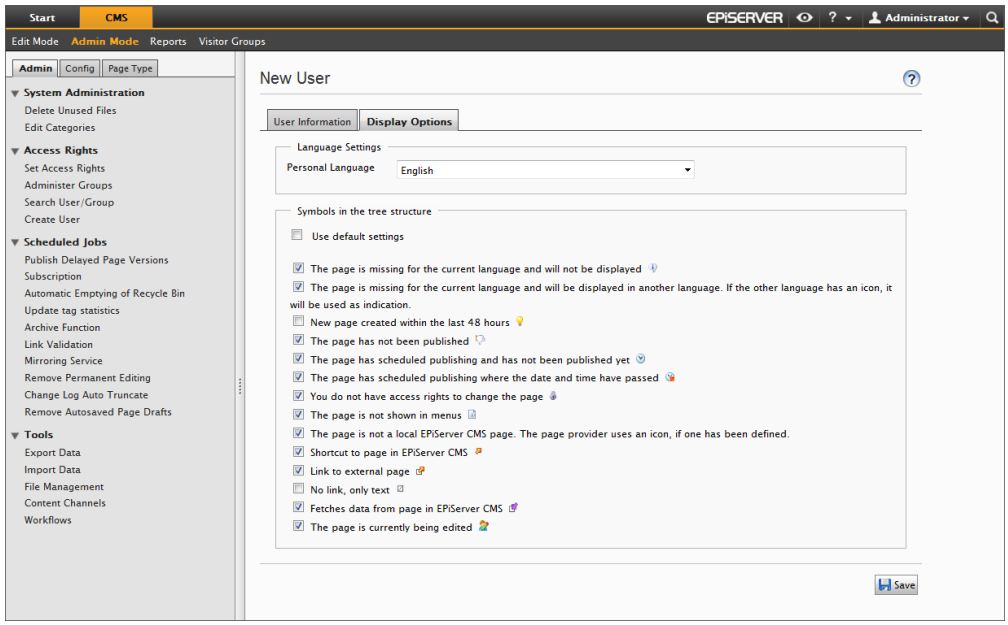
1. In Admin mode click **Search User/Group** in the **Admin** tab and search for the user to be deleted.
2. Select the user you want to delete by clicking the user name.
3. Click **Delete**.
4. Confirm that you want to delete the user.

Display Options for Users

The **Display Options** tab is for each editor to be able to decide what they want to show in the stucture in Edit mode. There are two different settings. On the one hand you can choose system languages for the user. EPiServer CMS has 12 different languages as standard. This means that Edit mode and Admin mode will be in the language that the user has chosen.

It is also possible to choose which symbols will be shown in the tree structure. Sometimes there are a lot of symbols on some pages and it is possible to close them, for those editors that don't need the symbols. Clear the **Use default settings** check box to select which symbols should be visible and which should not.

Editors can change these settings themselves under **My Settings**. These settings can also be made when the account is created.



Creating and Deleting Groups in EPiServer CMS

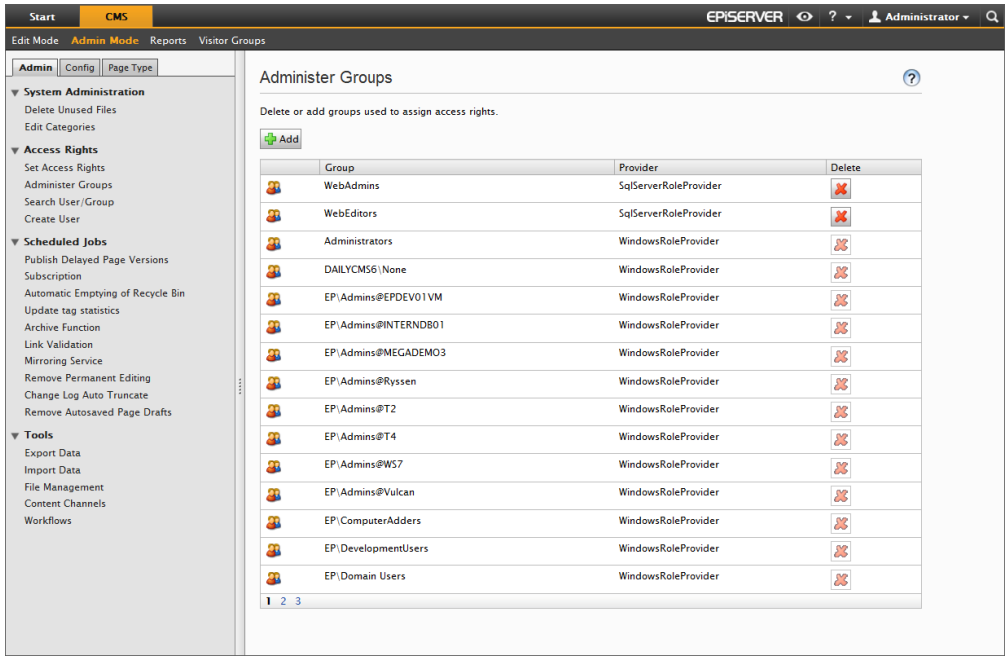
You can create different groups in EPiServer CMS and add users as members to them. Here you also can create the group VisitorGroupAdmins as described in Providing Access Rights to Different Views.

Creating a Group

Add a new group in EPiServer CMS as follows:

1. On the Admin tab, select **Administer Groups**.
2. Click **Add**.
3. Enter the name and description of the group.
4. Click **Save**.

Note that the groups you create in EPiServer CMS are only available here. It is not possible to create groups in EPiServer CMS in such a way that they are also created in Windows. Also note that it is not possible to change the name of an existing group. Delete the existing group and add a new group instead.



Deleting a Group

Delete a group in EPiServer CMS as follows:

1. On the **Admin** tab in Admin mode, select **Search User/Group** and search for the group you want to delete.
2. Select the group you want to delete by clicking the group name.
3. Click **Delete** to the right of the group name. It is also possible to delete members from a group in this way.
4. Confirm that you want to delete the group.

You can also delete a groups by selecting the **Admin** tab in Admin mode > **Administer Groups**, and click the **Delete** button to the right of the group.

Note that it is only possible to delete groups that use the SQL Server provider.

Setting Access Rights in the Tree Structure

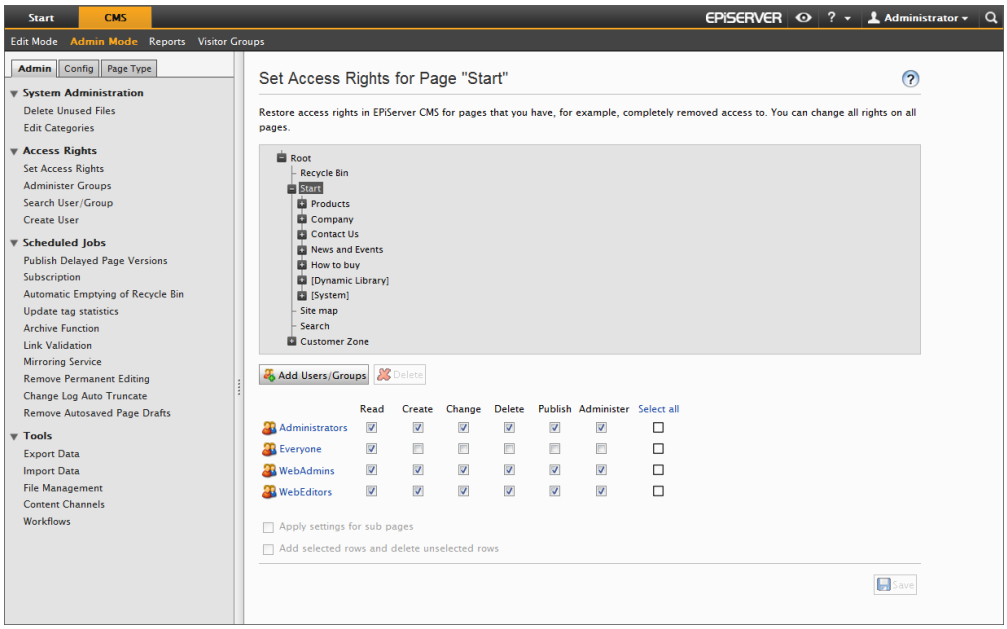
When you have created the group, it is time to enter where in the tree structure the group should have authorization. You then also determine what access rights a specific group is to have in that part of the structure.

EPiServer CMS offers six different levels for what a group or user can do on a specific page. Each group or user can have different access rights to each page, since there is no hierarchy within these different levels. This means that a group may have the right to read and publish pages but not to create, edit and delete them.

- **Read** means the group/user has the access rights to read the current page.
- **Create** means the group/user has the access rights to create pages under the current page.
- **Change** means the group/user has the access rights to edit the current page.
- **Delete** means the group/user has the access rights to delete the current page.
- **Publish** means the group/user has the access rights to publish the current page.
- **Administer** means the group/user has the access rights to edit dynamic properties and set access rights and language properties for the current page. This level has nothing to do with Admin mode, except to administer one or several pages in Edit mode.

Specify access rights for one or more pages as follows:

1. Select **Set Access Rights** on the **Admin** tab. The tree structure is displayed in the lower section of the window.
2. Select the page in the tree structure for which you want to set access rights. When the page is highlighted, the name of the page is formatted with a dark gray background.



3. Those groups and individuals that have access rights to the highlighted page are shown with the most recently saved access rights above the structure. If you also want other groups or individuals to have these access rights, click **Add Users/Groups** and then select the ones you

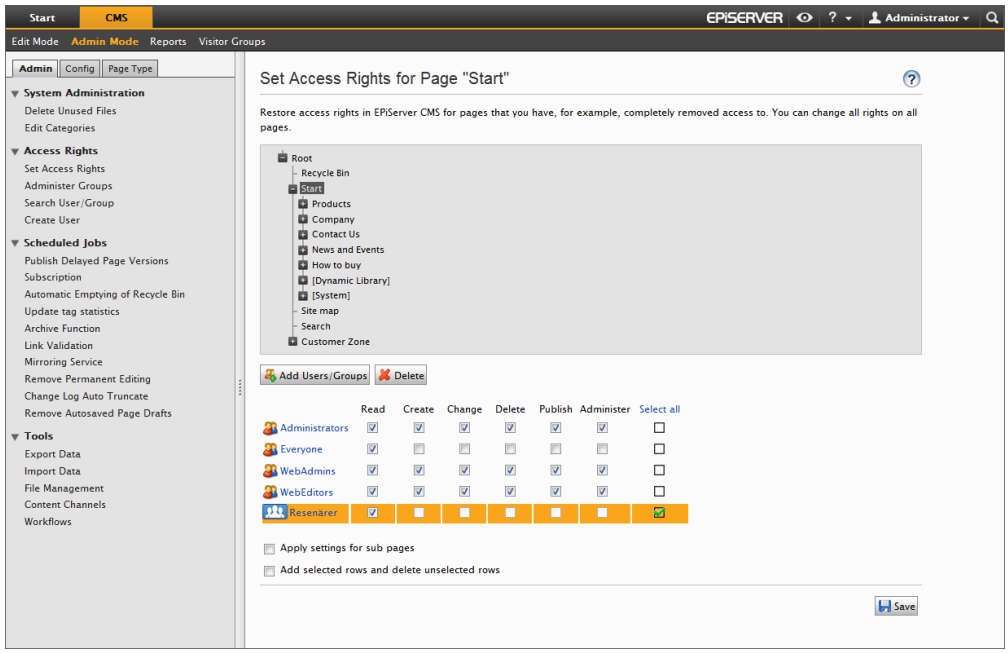
want. When you add a group or user, no settings will be saved; you are only seeing the information in the window.

4. Specify the level of access rights each group or individual is to have.
5. Save the new access rights by clicking **Save** above the groups. If you want the access rights to apply to the current page and its sub pages, select the **Apply settings for sub pages** check box before you save. If you do not select this, you will only save the access rights on the highlighted page.

The rights that you have stored apply to the existing pages in the structure that you have saved. New pages always inherit the same access rights as the main page.

Remember that you only affect the groups that are highlighted. The groups that are not highlighted when you click **Save** will remain unchanged, irrespective of previous settings. It is therefore important that you do not highlight a group by mistake that you do not want to affect.

Changing and Deleting Access Rights for Groups in the Structure



It is possible to make changes to a tree structure with an existing access rights structure. For example, there can be a group of editors who have not previously had access rights to publish their pages, but will now be able to. Or, a visitor group who have not previously had access rights to read a page, but now will be able to.

1. On the **Admin** tab, select **Set Access Rights**. The tree structure is displayed in the lower section of the window.
2. Select the page in the tree structure for which you want to edit the access rights. When the page is selected, the name of the page is formatted with a dark gray background.
3. Those groups and users that have access rights to the selected page are shown with the most recently saved access rights above the structure. Clear or select one or several access rights you want to add or delete for the group or user. The entire group is selected when you click one of the check boxes. EPiServer CMS interprets this as that you want to change this group.

4. Save the new access rights by clicking **Save** above the groups. If you want the access rights to apply to the current page and its sub pages, select the **Apply settings for sub pages** check box before you save. If you do not select this, you will only save the access rights on the selected page.

To set up access rights for visitor groups you generally follow the instruction as described above, but instead you click **Add Users/Groups** and select **Visitor groups**. Search the visitor group you want to add. You can decide whether you want everyone or only a particular visitor group to access the page. Note that visitor groups will only have read access.

Remember that you only affect the groups that are selected. The groups that are not selected when you click **Save** will remain unchanged, irrespective of previous settings. It is therefore important that you do not select a group by mistake that you do not want to change.

Deleting Access Rights for Groups in the Structure

If you have groups or users in the tree structure that you no longer want to have access rights, remove the entire group. There is a great difference between removing the group and removing certain levels from a certain group. If, for example, you want to change the settings so that a group will no longer be able to publish pages, you do this by changing an existing group. We remove the entire group if we have a group that should no longer have any access rights whatsoever.

1. Select **Set Access Rights** on the **Admin** tab. The tree structure is displayed in the lower section of the window.
2. Select the page in the tree structure for which you want to remove the group and its access rights. When the page is selected, the name of the page is formatted with a dark gray background.
3. Those groups and individuals that have access rights to the selected page are shown with the most recently saved access rights above the structure. Select one or several groups you want to delete by clicking on the green tick on the far right.
4. Delete the group by clicking the Delete above the groups. If you want to delete the groups for the current page and its sub pages, select the **Apply settings for sub pages** check box before you click **Delete**. If you do not select this, you will only delete the group from the selected page.

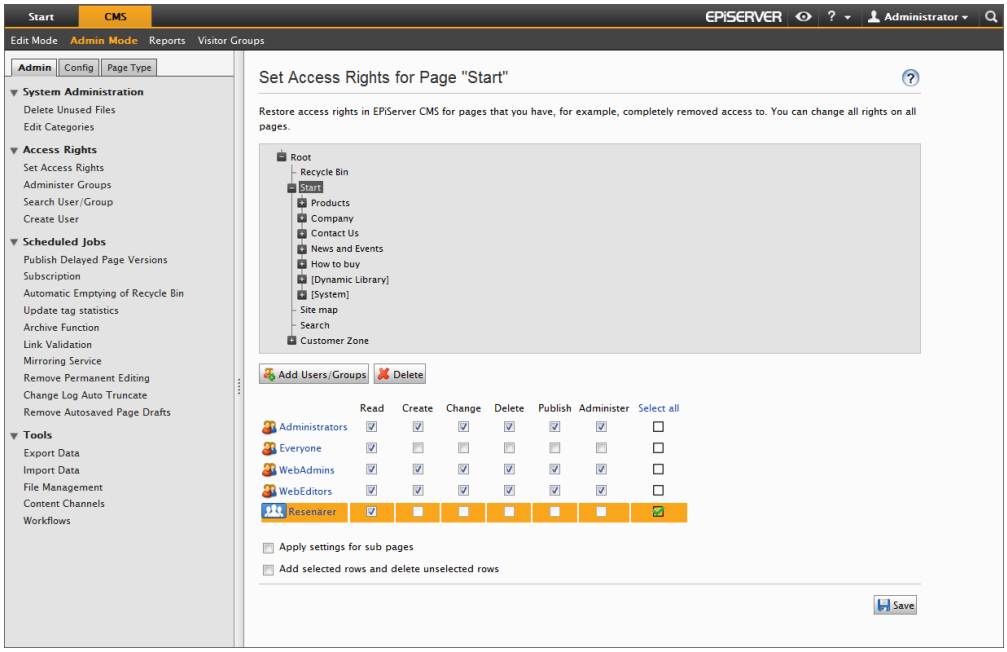
Remember that you only affect the groups that are selected. The groups that are not selected when you click **Delete** will remain unchanged, irrespective of previous settings. It is therefore important that you do not select a group by mistake that you do not want to delete.

Overwriting All Previous Access Rights

Sometimes there are occasions when is might be good to start again from the beginning with the access rights in a structure. Maybe you have changed the names of your groups and want to delete the old information and start with the new instead. To avoid doing this in two stages, by first deleting all the groups and then adding the new ones, there is an option that allows you to do this in one go.

1. Select **Set Access Rights** on the **Admin** tab. The tree structure is displayed in the lower section of the window.

2. Select the page in the tree structure in which you want to delete groups and add new ones. When the page is selected, the name of the page is formatted with a dark gray background.



3. Those groups and individuals that have access rights to the selected page are shown with the most recently saved access rights above the structure. Select the group(s) you want to keep by clicking on the green tick on the far right.
4. Select the **Add selected rows and delete unselected rows** check box.
5. Select the **Apply settings for sub pages** check box if you want to add and delete groups on all sub pages. If you do not select this, you will only add/delete it from the selected page.
6. Click **Save**.

The groups that are selected, with the selected access rights, will be saved on the current page and its sub pages if this check box was selected. All the unselected groups will be deleted. In the example below the Administrators will be saved on the selected page and all its sub pages. Everyone will be removed from the selected page and all its sub-pages.

Setting Access Rights for Different Editors in the Tree Structure

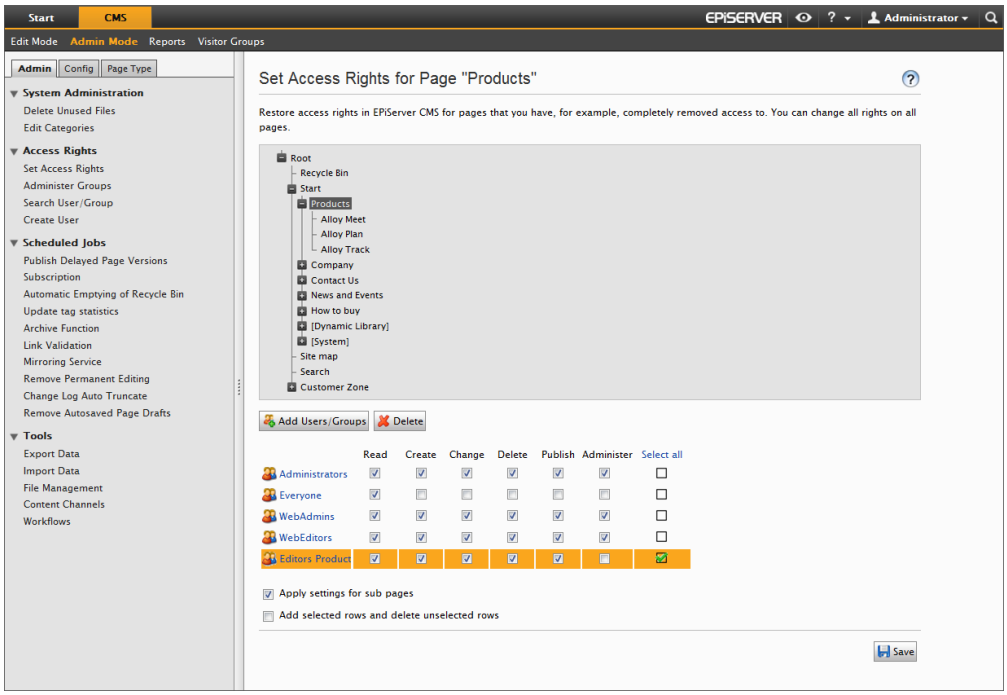
Within a large organization, it is customary to have control over editors, so that they can only create, edit, delete and publish in specific parts of the tree structure. On an extranet, it is also quite usual to grant different customers access to read different pages. To do this, you must start by creating more groups than WebEditors. There must be a group for each part of the tree that can be separated from the other editors.

For example, a website has three different main branches: Products, News and Support. Each section has two or more editors who will help to update information on the website. These editors will only be able to create and edit pages within their own branch of the tree. We start by creating three groups, either in Windows or EPiServer CMS, depending on the normal procedure. In this case, the groups are called Products_Editors, News_Editors and Event_Editors. If we have not yet created the users, we must do so now. Then we make sure that the different editors are members of the group where they are to edit. All editors must be members of WebEditors, otherwise they will not be able to access

Edit mode. This means that all editors must be members of at least two groups: WebEditors and the group that specifies what they can edit.

Once the groups and users have been created, set access rights in the correct locations in the structure.

1. On the **Admin** tab, select **Set Access Rights**.
2. Select the page in the structure for which you want to set access rights. In this case, it is the **Products** page.



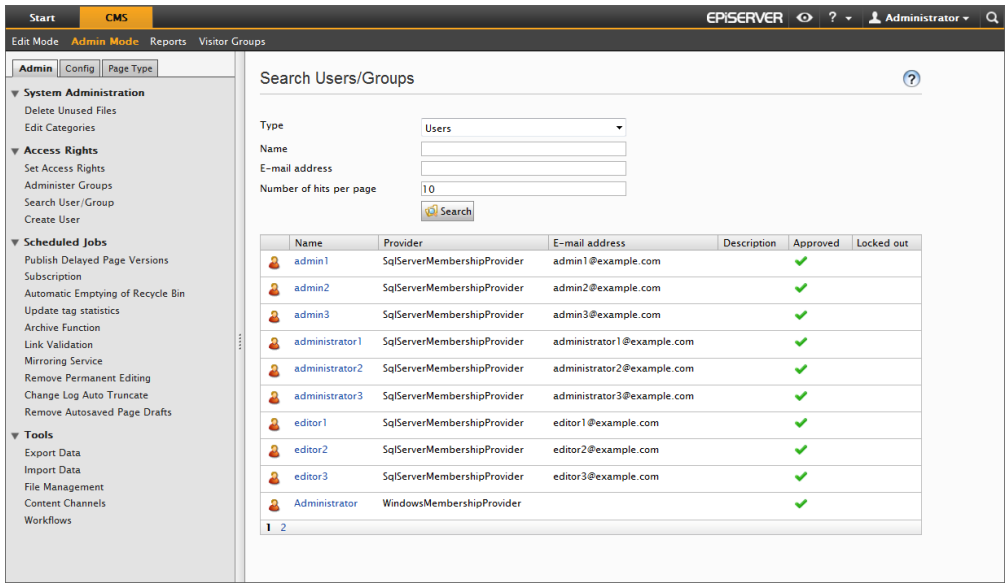
3. You can now see that no editors have access rights to this page, so we must start by adding the products_editors group. Click the **Add Users/Groups** button and add your group to the page.
4. Grant the required access rights to the group by selecting the check boxes beside the group. If you double-click a check box, all the check boxes on the left will be selected.
5. Make sure that none of the other groups are selected.
6. Select the **Apply settings for sub pages** check box.
7. Click **Save**. The current group will be added to the selected page and its sub-pages. The other groups will not be affected.
8. The access rights have been saved.
9. The next time an editor from the products_editors group logs on, he/she will only be able to edit the Products page and its sub pages.

Similarly, it is possible to create different groups in order to grant different visitors access to different pages. On a public website, for example, ensure that “Everyone” is not granted access rights to read those pages you do not want to be seen by the general public. By allowing extranet users to log on, they will have access to different pages, depending on the read rights you have set for them. If a visitor does not have the read rights to a particular page, that page will not appear in their menus. The visitor is simply not aware that there is more information available.

Search User/Group

If you wish to check which groups a particular individual is a member of, change settings or maybe give a user a new password, you must find the user's account. This can be done using **Search User/Group**. Specify the type (group or user) you want to search for; specify search criteria in one or more of the other fields and click **Search**.

The search results are listed and by clicking a group or user name, you can edit the information about that group/user. In the **Number of hits per page field**, enter the number of users that should be displayed on each page. If the search results produce several pages, scroll through the pages in the lower left corner.



Authorization to Publish from Microsoft Office

There is a function in EPiServer that allows an editor to create information in EPiServer CMS by publishing a document from Microsoft Office. This function requires installation on each machine, which will publish directly from Office. This is something that editors can carry out themselves in Edit mode. Each editor must also go in and activate the website they are to work on.

In order for an editor to be able to activate a website to publish from Office they must have authorization to do this.

Authorize a group/individual to publish from Office as follows:

1. Select **Permissions for Functions** on the **Config** tab.
2. Click **Edit for View and publish pages from Microsoft Office add-in**.
3. The group/users with authorization to publish from Office are displayed in a list.
4. Add or delete groups and users, so that only those permitted to publish from Office are displayed.
5. Click **Save**, at the top of the page.

If you have changed the authorization as described above and it is still not possible to publish from MS Office, check your web.config file. In order for the authorizations to work, a setting in web.config is required. The following is included in the web.config file in a standard installation of EPiServer CMS, but must be uncommented before it is possible to publish from Microsoft Office.

To uncomment the text in web.config, do the following:

1. Open the file in Notepad or a similar program. (It must be a program that supports UTF-8, otherwise the program can damage the file.)
2. Remove the initial “<!--” and final “-->” from the code.

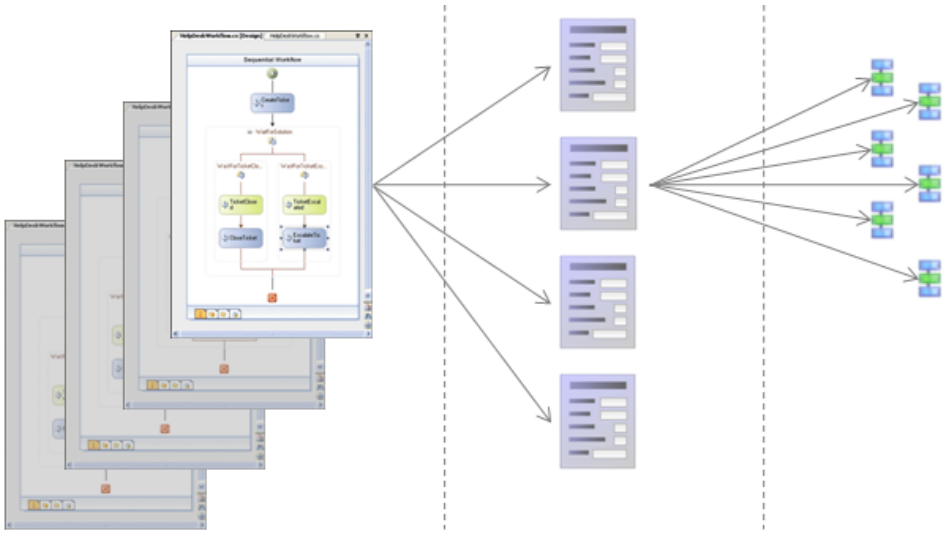
   ```
   <!--<location path="UI/OfficeIntegration">
   <system.web>
   <authorization>
   <allow users="*" />
   </authorization>
   </system.web>
   </location>-->
   ```

   UI is an customized path to Edit mode. This might be different in your installation.
3. If your web.config file does not contain the above code, add it uncommented to the file.
4. Save the file.

Workflows in EPiServer CMS

The workflows in EPiServer CMS are built on standard technology from Microsoft, which is called Windows Workflow Foundation. This gives a developer the opportunity to create all types of workflows and integrate them in EPiServer CMS. A developer must carry out the technical adaptation of the workflow before administrators can manage their daily work directly from Admin mode in EPiServer CMS. The developer creates the functionality to assign tasks, send e-mails, create information or similar. The administrator manages what an instance is triggered by, who will receive tasks and different settings that the developer constructs as variable values.

The basis of a workflow is that there is source code that defines what should happen during different events. Perhaps tasks are to be sent, pages published, etc. This is what we have called Workflow Foundation files in the image below. It is fully possible to have a number of different code files that work in different ways. It is common practice to make the code flexible so that you can use the same code for different scenarios. Sometimes, an editor should perhaps be able to start a workflow manually, another time perhaps it should be started automatically by pressing a certain button in a certain place, but it should still be managed with the same technique. In this case, you create different definitions from Admin mode built on the same technique. The definition contains a large number of different parameters for what should trigger the workflow, authorization to start it and a lot of parameters to facilitate the workflow. When a definition is started, an instance is created. A definition can have several hundred instances underway at the same time and this is often what editors see as the actual workflow.



In a standard installation of EPiServer CMS, four different workflows are included that can be used as they are or as an example of how you could develop them further. If the workflows are not available in Admin mode after installation, remove the comments from the <definitions> section under **<episerver xmlns="http://EPiServer.Configuration.EPiServerSection">**.

Sequential approval workflow

The sequential approval workflow involves one or more people/groups receiving a task to approve an event in a sequential order. If a person in the workflow does not approve the task, information about this will be sent to the person who created the event; at the same time the task is not sent on to the

next person in the workflow. It is, for example, possible to use this for news items that are to be published. Two individuals, one focusing on the content and one with linguistic skills, must approve the content of the page before it is published.

Parallel approval of pages

The workflow for approval in several parallel steps involves several people/groups receiving a task at the same time to approve an event. If one of the people approves or does not approve the event, the task will be removed from the other people. It is, for example, possible to use this for a discussion forum in which we want to moderate the entries in the forum. When a visitor creates an entry, the page is saved and tasks are sent to several people to approve the page. When one of these people has approved or rejected the entry, the task will be deleted from the others in the group.

Translation product pages

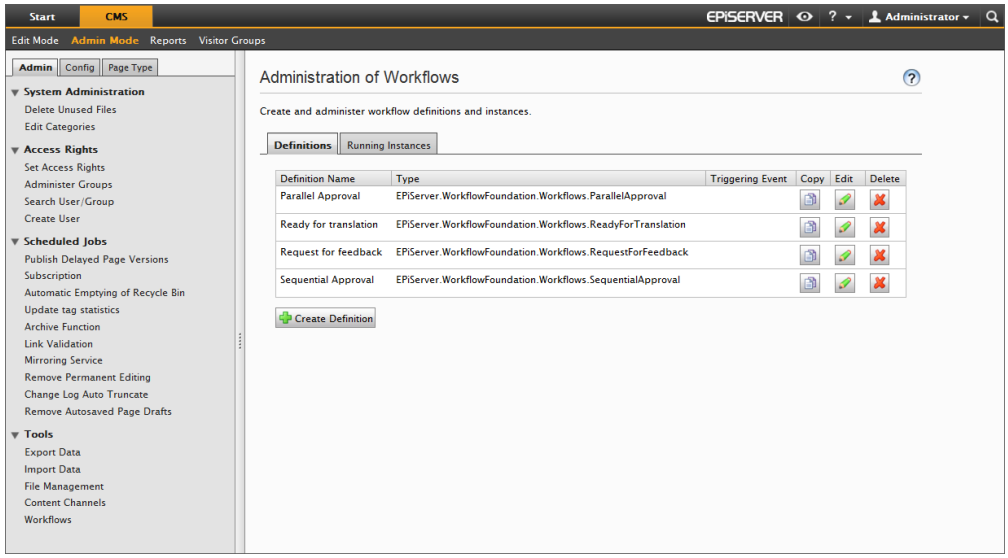
The workflow for managing the translation of pages is based on the activation of globalization in EPiServer CMS. With globalization, you can write the same page in several different languages. Usually one person will create the first language version. The workflow sends a task to another editor saying that it is time to create another language version of the page. If the page has not been translated after a certain length of time, a reminder is sent. If this is not done within a set period of time, the creator of the first page and the process owner receive notification that that page has not been translated. If you have a website that is to be translated to several languages, tasks are sent to all the translators at once.

Request for feedback

The request for feedback workflow means that an editor/group can ask for feedback about something he or she has created in EPiServer. This could be a page that the person in question is unsure that it is being used in the correct way. It might be that someone wants help with linguistic or graphical issues. It is then possible to start a workflow for feedback. The person who starts the workflow selects which page and who he or she wants feedback from. A task is sent to the person/group who then give their feedback. All the managing of messages takes place in EPiServer. Remember that the person who is to give feedback on the page must have authorization to read it, even if it has not been published.

Administration of Workflows

Select **Workflows** in Admin mode to open the Administration of workflows window. A list of all the definitions appears and from here you can create new, copy, change and delete definitions.



In this window you have three different tabs.

Definitions contains all the definitions that have been created. In a definition, you enter what will trigger an instance, which people should be involved, which places it should apply to, etc. It is fully possible to have several definitions that build on the same technical workflow.

Running instances Every definition is triggered by a certain event or a manual start. You can see all the instances that are now running on the Running instances tab.

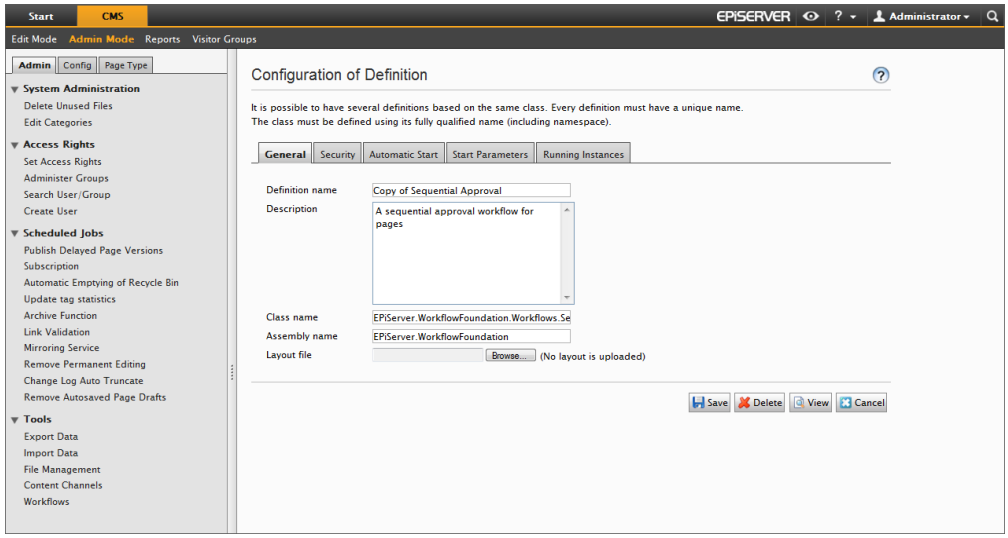
Create New Workflows From Admin Mode

In order to be able to create a new workflow, you will need information from the developer who has created the technology behind the workflow. You can also choose to use one of the workflows included as standard or copy one of these to create a new workflow.

1. Select **Workflows** on the **Admin** tab in Admin mode. Existing workflows on the website are listed on the right side of the page.
2. To configure a new workflow click **Create definition**, or click **Copy workflow** to copy an existing workflow. The form for the settings for the workflow appears.
3. When you create a new workflow, three tabs appear. When you have selected which workflow will manage the technology behind the workflow and saved it, more tabs will appear.
4. Enter your settings for the various tabs and click **Save** at the top, above the tabs.

General Tab

On the **General** tab, specify general information about the workflow and which technology should be used. In order to be able to create a new definition, you will need information from the developer.



Definition name

Specify the name of the workflow. If the workflow is to be started manually, the name should be as descriptive as possible so that the editor can differentiate between the workflows.

Description

A brief description of the workflow; this text will be displayed to the editor when he/she is to activate the workflow.

Class name

Specify the class name that the workflow has in the code. The developer of the workflow must provide this information. For the four workflows in a standard installation of EPiServer CMS, the class name is as follows for the different flows:

Workflow	*Class name*
Sequential approval workflow	EPiServer.WorkflowFoundation.Workflows.SequentialApproval
Parallel approval of pages	EPiServer.WorkflowFoundation.Workflows.ParallelApproval
Translation product pages	EPiServer.WorkflowFoundation.Workflows.ReadyForTranslation
Request for feedback	EPiServer.WorkflowFoundation.Workflows.RequestForFeedback

Assembly name

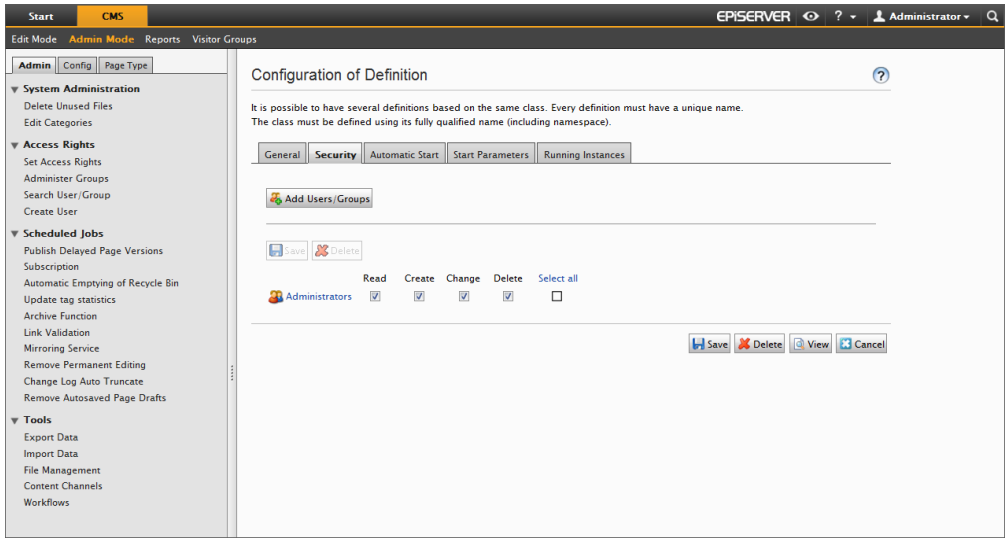
Specify the assembly name for this workflow. The developer of the workflow must provide this information. In the four workflows available in a standard installation of EPiServer CMS, the Assembly name is is always EPiServer.WorkflowFoundation.

Layout file

A layout file is automatically created when you develop the workflow and gives a schematic picture of the different events in the workflow. It is possible to upload your own such picture if you wish.

Security Tab

All the workflows can be started manually from Edit mode if you have authorization to do this. On the **Security** tab you can specify which groups and people should have authorization to read, change, delete and start the current workflow directly from Edit mode.

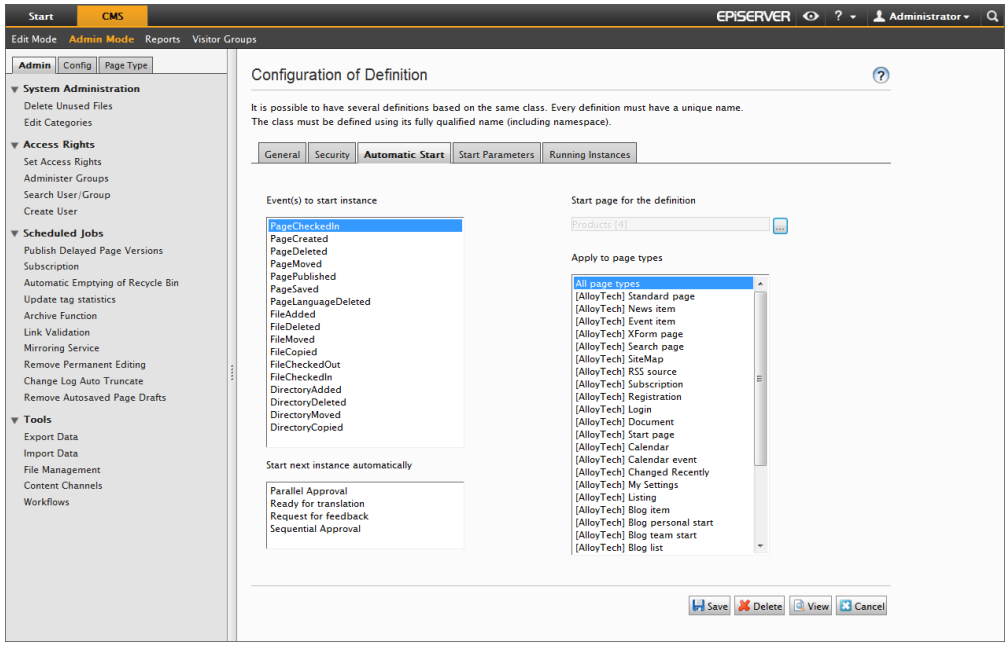


To give different people or groups authorization, you have to add them with the **Add Users/Groups** buttons. By selecting the different check boxes you can give different people/groups different authorization.

Authorization	*What it means*
Read	The group/person can see running instances in the current workflow from Edit mode.
Create	The group/person can start an instance in the current workflow from Edit mode.
Edit	The group/person can change the settings of an instance in the current workflow from Edit mode. This is of course provided that you have also selected the check boxes for certain workflows for this under the Start Parameters tab.
Delete	The group/person can delete an instance that has started in the current workflow.

Automatic Start Tab

On the Automatic start tab you specify which event in EPiServer should trigger the instance to start. You can also choose to start an instance built on another workflow directly after this instance has finished. Depending on which event starts the instance, you will also have the option of selecting a place in the structure and the page type for which it will apply.



Events that will start an instance

All selectable events are events in EPiServer CMS. This involves either pages or files and folders in the File Manager.

Event	*Involves*
PageCheckedIn	A page is checked in by the editor selecting the Ready to publish button.
PageCreated	A page is created.
PageDeleted	A page is deleted.
PageMoved	A page is moved in the tree structure.
PagePublished	A page is published.
PageSaved	A page is saved.
PageLanguageDeleted	A language version of a page is deleted.
FileAdded	A file is uploaded to the File Manager.
FileDeleted	A file is deleted from the File Manager.
FileMoved	A file is moved.
FileCopied	A file is copied.
FileCheckedOut	A file is checked out.
FileCheckedIn	A file is checked in.
DirectoryAdded	A folder is created in the File Manager.
DirectoryDeleted	A folder is deleted from the File Manager.
DirectoryMoved	A folder is moved.
DirectoryCopied	A folder is copied.

Start next instance

All the definitions created are displayed in the list under Start next instance automatically. By selecting another definition, it will start automatically after the one you are creating right now.

Place in the structure

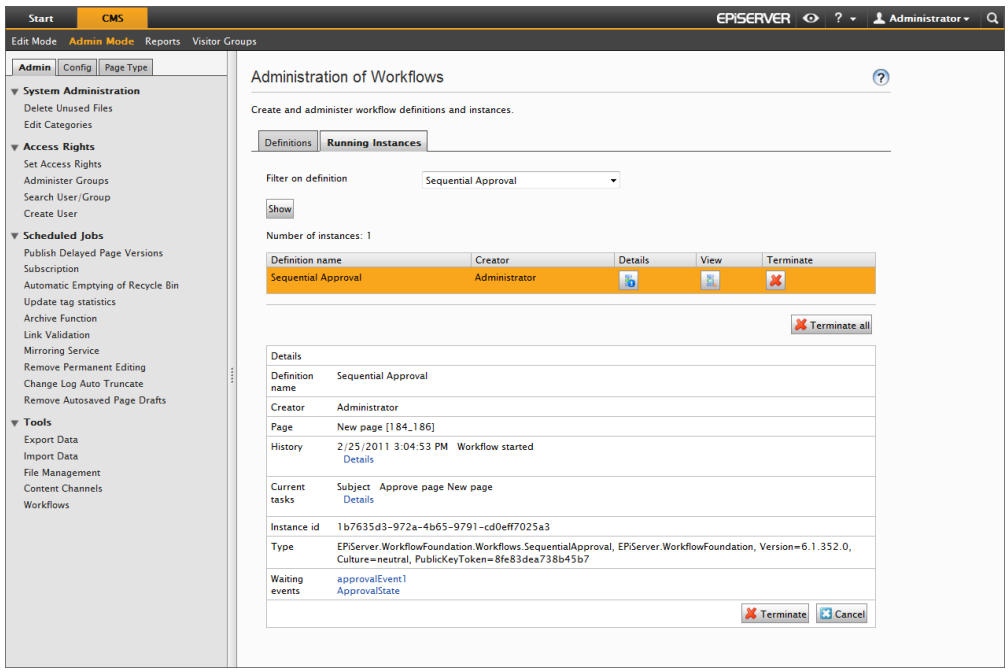
If you have selected a page-related event, you can choose where in the tree structure it will apply. You do not have to choose a place, but if you do not, the event will be triggered by the event you have chosen in the whole structure.

Page type for which the definition will apply

Sometimes you have specific templates for different functions on the website. Using Apply to page types, you can select a certain page type to trigger the instance. You do not have to choose a page type and if you do not, the event will be triggered by the event and place you have chosen irrespective of the page type the editor has used.

Running Instances Tab

As administrator, if you want to view the workflows that are currently in use, you can view a list of all running instances under the Running instances tab. You can reach the tab either from Administration of workflows or when you are looking at a specific definition. Here you can see who has been allocated a task, but as yet has not carried it out. You can also see those tasks that have been carried out and comments that editors have made regarding the tasks. From here, you can also delete the workflow. That is, you delete the active instance, not the entire workflow.



Start Parameters Tab

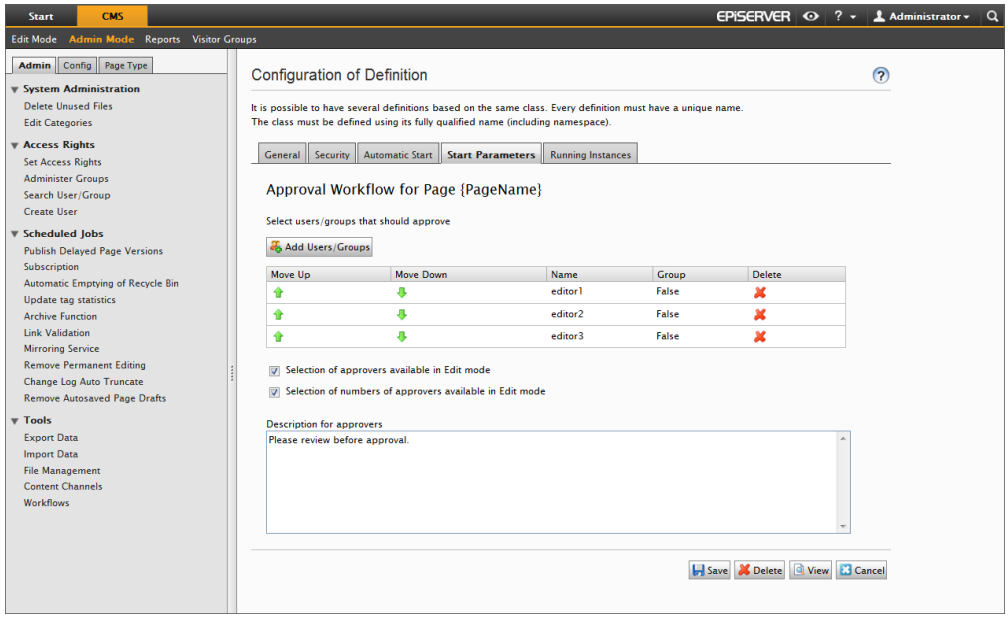
The contents of the **Start Parameters** tab vary for each type of workflow. The type of settings you can enter for each type are decided when you develop the workflow. In general, you try to make the work-

flow as adaptable as possible from Admin mode instead of hard coding in values that will probably be variable.

Sequential Approval Workflow

The Sequential Approval workflow exists to enable a sequential workflow for approval in several steps. Here we have set up a scenario in which two people after each other have to check product pages that editors create. Jane is the one who has to check that the prices are correct and Pelle will check the text and images. In this workflow, we have the option of setting which group or person will approve the pages. When the last person has given their approval, the page is published.

If we give the editors the access rights to edit the workflow, we should also select the **Selection of approvers available in Edit mode** check box here. Then the editor will be able to switch Jane or Pelle for other people or groups. This also requires that we give them the access rights to make changes on the **Security** tab.

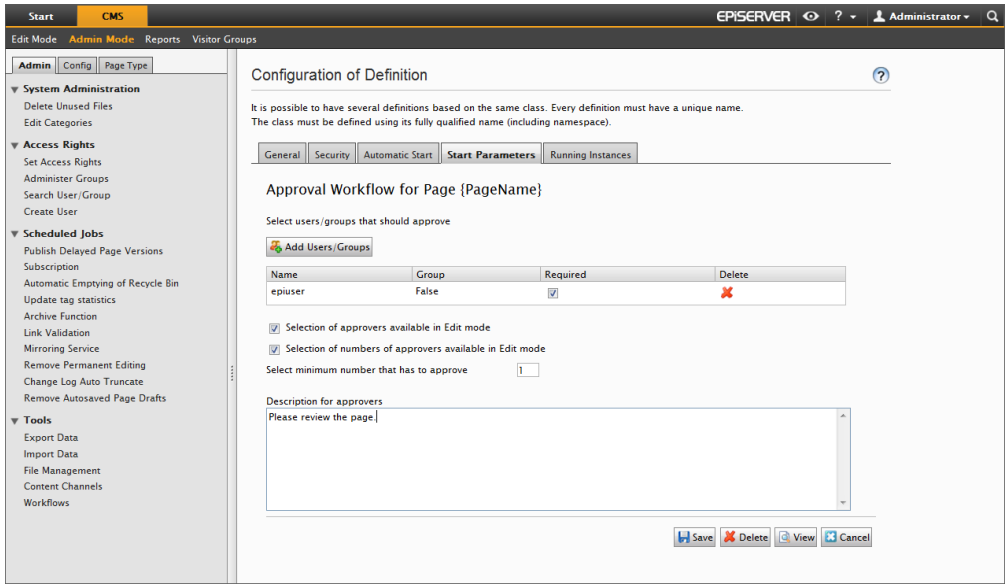


Parallel Approval of Pages Workflow

The workflow for Parallel Approval of Pages exists to allow a parallel workflow for approval in several steps. Here we have set up a scenario in which two people parallel to each other have to check news pages that the editors create. In this case, we have chosen to specify a group to approve the pages. This means that there might be a lot of members of the group and some of them will approve the page. For the **Select minimum number that has to approve** check box, we have entered two. This means that two people from the group must give their opinions of the page. If the second person has approved the page, the page will be published and the task removed from all other group members. If the first person does not approve the page, the page will be deleted from the others and the page will not be published. This is because we have said that two people have to approve the page.

If we give the editors the access rights to edit the workflow, we must also select **Selection of approvers available in Edit mode** and **Selection of number of approvers available in Edit mode** here. Then the editor will be able to switch the group for other people or groups. They will also be able to

choose how many people should give approval. This also requires that we give them the access rights to make changes on the **Security** tab.



Translation Workflow

The workflow for translation of pages with EPiServer globalization support exists to enable the easy division of tasks for websites that are to be available in several languages. As an editor you feel that you only have one website, even though there are in actual fact parallel tree structures with different languages. If there are different people who are to create the different language versions, it is a good idea to set up a workflow that sends tasks to the correct person when it is time to translate pages.

We have set up a scenario in which only the product pages on the website are to be translated into one more language. In our case, an English-speaking editor does the first work on the page. When he or she publishes the page, the sequential approval workflow is triggered first and when the information has been approved by two people, the translation workflow is triggered and someone receives the task of creating a Swedish version of the same page. In this workflow there are a lot of settings to be made:

Start delay (minutes) With this parameter it is possible to delay the start of the instance. It is quite common that the first person publishes the page quickly and then sees the changes that ought to be made, so it is best that tasks are not sent every time the page is published.

Time before reminder is sent (hour) If the person/group that is to translate the page does not do this, a reminder will be sent after the period of time you specify in this field.

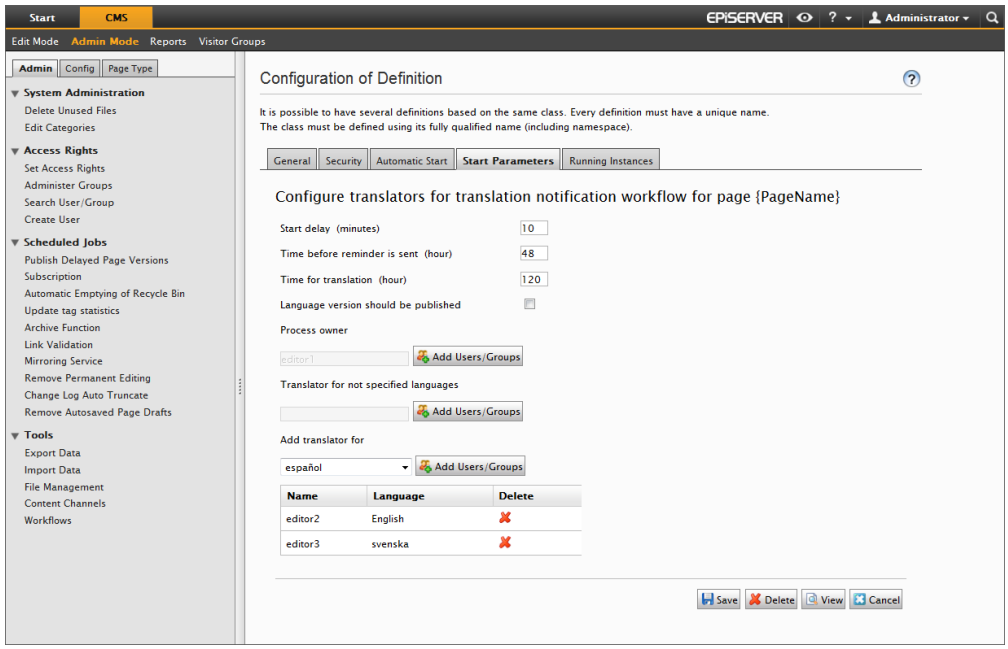
Time for translation (hour) The translation must be done and the page published before the time specified in this field. If it is not done, notification will be sent to the process owner.

Language version should be published The task is not complete before the page is published if this check box is selected. This means that the task will not disappear from the list of tasks until the page is published.

Process owner The person or group that is the process owner will receive information from the workflow. Amongst other things notification will be sent when the page has been translated and published and if the translation has not been done within the specified time.

Translator for not specified languages Usually, an editor is assigned for each language. For the languages that do not have a specific editor, we can assign a person or group to manage the other languages.

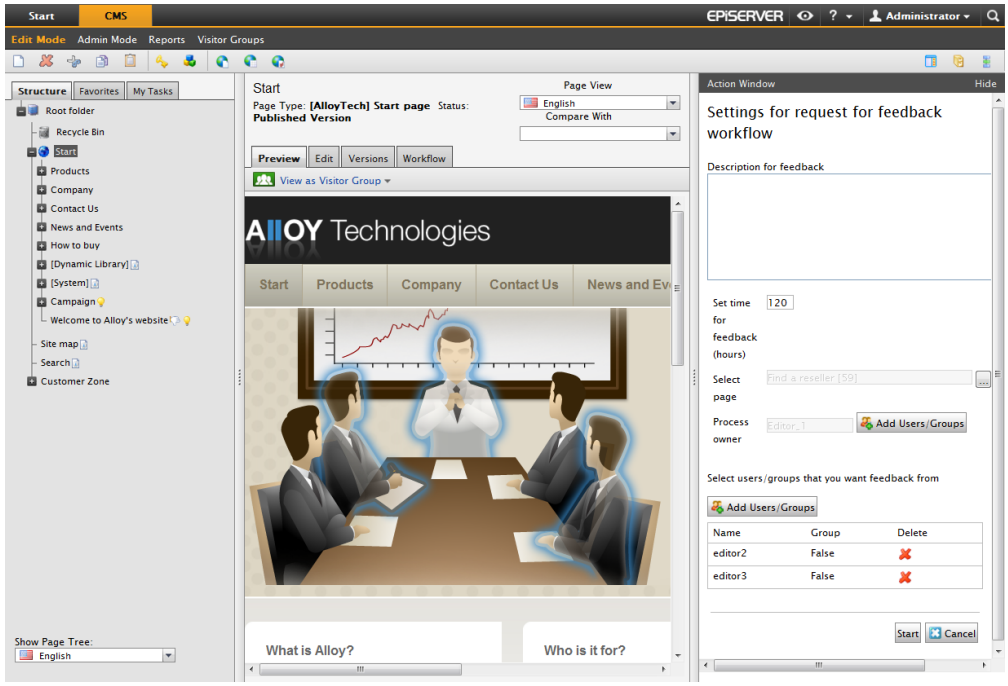
Add translator for Select the language in the drop-down list and then specify which group or user is responsible for the language. Only the languages that are activated on the website are shown in the drop-down list.



Request for Feedback Workflow

The request for feedback workflow exists so that an editor can easily ask for feedback about what has been done from colleagues or those responsible. It could be a question about the selection of images, linguistic issues or how you use a certain page type.

We have set up a scenario in which all the editors have the right to start an instance on this workflow. Anyone should be able to ask for feedback. We have also given them access rights to write their question and decide who will give them feedback. They can also choose to put themselves as process owners, so that they then receive notification if the task is not carried out.



Editing Definitions

If one individual finishes and another takes their place in a workflow, you can edit the existing workflows. Select **Workflows** on the **Admin** tab in Admin mode. Existing workflows on the website are listed on the right side of the page. By clicking the **Edit** button on the workflow in the list, you can edit the workflow. Make your changes to the respective tabs and click **Save** from the top, above the tabs.

Delete Definition

If, for some reason, you want to delete an entire workflow, you can do this from Admin mode. Select **Workflow** on the **Admin** tab in Admin mode. Existing workflows on the website are listed on the right side of the page. By clicking the **Delete** button, you delete the definition of the workflow.

Note that all activated instances within this workflow will be closed if you delete it.

File Management in EPiServer CMS

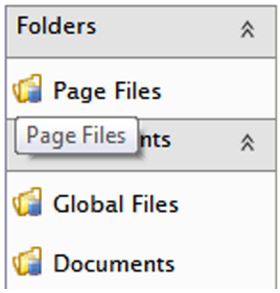
File management in EPiServer CMS is a part of the daily work carried out by editors. They create pages in EPiServer CMS that are stored in the database. The pages are linked to images and documents that are stored in a file structure customized for each website. EPiServer CMS has full support for file management on the server via the browser. From EPiServer's File Manager, the editor can create folders, upload files and manage the files by copying, moving, deleting, etc. To aid editors in working with folders and files it is important that a policy on how to use them be established from the start. Access rights can also be set for different folders for ease of use.

EPiServer CMS also has support for version management of the files uploaded to the server. It is possible to follow all file changes and see who has done what to them. It is also possible for an editor to Check out a file so that nobody else can modify the file before it is checked in again. Read more in the Editor's Manual to find out about file management when working as an editor.

File management in Admin mode has the same functionality as in Edit mode. The difference is that the access rights in Admin mode are greater. Once you have chosen to use the File Manager fully, work starts in Admin mode to determine which groups should do what in various folders.

For the security function for setting access rights on folders and files to work, file management must be configured in web.config. For the file management to work in a folder, this must also be configured. In a standard installation, the starting points available are equipped with document management and access rights management.

One or More Starting Points in File Manager



When you configure your EPiServer CMS, you choose how many starting points should be available in the File Manager. A starting point can point to a specific physical place. The place does not have to be on the same server as the other files for EPiServer CMS. You can use this for various things, but a common scenario is that you want to have access to several different file servers from the website.

It is always the same starting point that is displayed in the link tool when the editor chooses to insert images or documents. The most used folder is usually selected as default folder in the link tool. It is the folder that you want the editors to initially save documents to. You specify this setting in the configuration in episerver.config. The image to the right has two starting points, apart from page files, which is called **Page** files here.

Configure Starting Points

The technology used for managing folders in EPiServer is a standard technology from Microsoft, called Virtual Path Provider. In practice this means that you can build your own solutions for file management and work with them in EPiServer CMS via EPiServer's APIs. In a standard installation of EPiServer CMS 5, two starting points and the page files are configured. During installation a VPP directory is specified and this is the physical place in which the standard folders will be saved. This folder must be saved outside the folder that is the root of the website in order for the management of access rights to work.

The configuration of starting points is done in episerver.config. The image below shows how this looks in a standard installation. By changing the attribute, it is possible to change the configuration of the starting points that have been set up. It is also possible to copy a starting point in order to add more. The starting point that is shown at the top of the list is the starting point that will open when an editor opens the File Manager or inserts an image/document.

```
<virtualPath customFileSummary="~/FileSummary.config">
<providers>
<clear />


<add showInFileManager="false" virtualName="Page Files" vir-
tualPath="~/PageFiles/"
bypassAccessCheck="false" indexingServiceCatalog="Web" phys-
icalPath="C:\EPiServer\VPP\ExampleEPiServerSite4\PageFiles"
name="SitePageFiles" type="EP-
iServer.Web.Hosting.VirtualPathVersioningProvider,EPiServer" />

<add showInFileManager="true" virtualName="Global Files" vir-
tualPath="~/Global/"
bypassAccessCheck="false" indexingServiceCatalog="Web" phys-
icalPath="C:\EPiServer\VPP\ExampleEPiServerSite4\Global"
name="SiteGlobalFiles" type="EP-
iServer.Web.Hosting.VirtualPathVersioningProvider,EPiServer" />

<add showInFileManager="true" virtualName="Documents" vir-
tualPath="~/Documents/"
bypassAccessCheck="false" maxVersions="5" phys-
icalPath="C:\EPiServer\VPP\ExampleEPiServerSite4\Documents"
name="SiteDocuments" type="EP-
iServer.Web.Hosting.VirtualPathVersioningProvider,EPiServer" />

...

</providers>
</virtualPath>
```

Attribute	*Comments*
customFileSummary	Is an XForms file that contains forms for filling in metadata.
showInFileManager	Entered as true/false. "True" means that the starting point will be shown in the File Manager. "False" means that it will not be shown.
virtualName	Is the name shown in the File Manager.
virtualPath	Virtual path to the root of the file system.
bypassAccessCheck	Entered as true/false. "True" means that authorization will not be checked before an editor creates folders and uploads documents. "False" means that authorization is always checked.
indexingServiceCatalog	Enter the folder name for Indexing Service.
physicalPath	The physical path to the folder to be displayed.
Name	A unique name for the instance.
maxVersions	Specifies how many versions of a file can be saved.

Global Files vs. Page Files

In EPiServer CMS, it is possible to work with two different methods when uploading files. The most common is to use both in parallel to each other and the editors select the method that suits them best depending on the page being created.

Page Files

When an editor uploads a file to the page files, a new folder is always created that is linked to that page. The folder is given a name consisting of digits. If you have several pages of files, there will be several folders in your structure. The page folder is called Page Files in a standard installation. It is possible to change this in web.config if you wish.

One advantage of using page files is that editors do not need to keep track of where to put their images. Another is that you can use the **Delete unused files** function from Admin mode. This means that all folders linked to pages that are no longer on the website, are erased. If you use page files, the applicable rights for a page are mirrored in the folder structure. You can use this for various things, but a common scenario is that you want to have access to several different file servers from the website.

The advantage of saving files to page files is that the editors do not need to remove old images themselves. One disadvantage is if you use the same file on several pages, in which case you must upload them to a new folder every time you use the file.

Global Files

When an editor uploads a file to a global file, the editor must locate the place in the folder where the file is to be stored before the upload is started. It is useful to create a file structure that the editor will recognize and comprehend before any work is started. It is also possible to give editors access rights to create their own folders and other file management functions.

One advantage of using the global files is that editors can easily find their way back to images that have already been uploaded on to the web server, because everything is stored in a folder structure that is easy to recognize.

One disadvantage of using the global files is that it is difficult to keep track of which files are being used. There is no automatic clearing for folders. Editors or administrators must clear this folder themselves. If you use EPiServer's File Manager when you move or remove pages, you always receive a warning about whether the file is being used by any page in the system.

Setting Authorization For Folders

When you have configured the File Manager with the starting points you want and have created folders for different editor groups, you must give access rights for the different folders. Here we assume that you already know about access rights and various user groups in EPiServer.

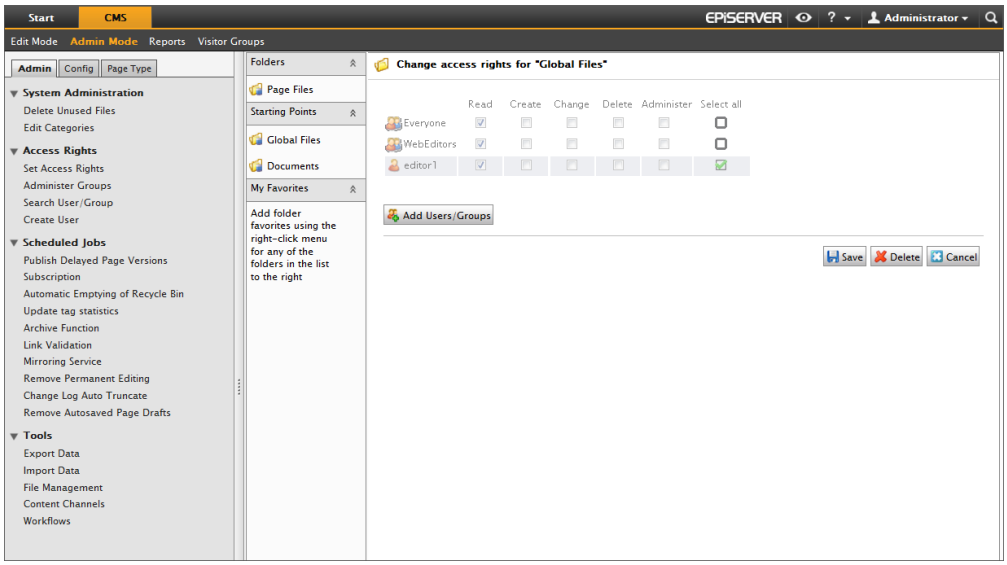
It is normal for Everyone to have read rights in all folders. This is because everyone is usually able to view images and read documents on the page. If you create a folder to load files to an extranet however, it is highly improbable that Everyone should have rights in the folder.

Editors can have five different levels of rights in a folder. The following levels are available:

- **Read** means the group/user has the access rights to read in the folder. Thus also the access right to link to information in the folder.
- **Create** means the group/user has the access rights to create folders and upload files to the folder.
- **Change** means that the group/user has access rights to change names, move and copy the content of the folder.
- **Delete** means the group/user has the access rights to delete information from the folder.
- **Administer** means that the group/user has rights to set access rights to folders from Edit mode. This has nothing to do with Admin mode.

Set access rights to folders as follows:

1. Select **File Management** on the **Admin** tab in Admin mode.
2. Open the folder that you wish to set rights to by double-clicking the folder.



3. Click the **Change access rights** tool. (You can also open this tool by right-clicking directly on the folder, steps 2 and 3 in one.)
4. The groups that have access rights in the folder are now displayed. The **Inherit permissions from parent** check box is usually selected. Clear this by removing the tick in the box.
5. To add more groups or users click **Add Users/Groups**.
6. Specify the level of access rights each group or individual is to have.
7. Click **Save**, above the groups.

All underlying folders automatically inherit the same rights as a main folder if this is not changed manually. If the editor creates new folders, these will also inherit the same rights.

If the Change Access Rights button is not active, it means that in most cases the File Manager is not correctly configured. This can also be because you do not have the authorization required to change access rights in the folders.

Deleting Unused Files

When editors upload files to the web server, they can opt to load to **Page files**. This then creates a folder that is linked to the page on which the editor is currently working. The folder that has been created is given a name comprised of digits. The file uploaded by the editor is stored in the new folder. When the editor deletes the page from the website, the images and documents linked to the page will remain on the web server.

Selecting **Delete Unused Files** removes all the folders in which editors have used **Page files**. For the files to be deleted from the web server, the page must be deleted and the Recycle Bin emptied. It takes a further 30 days after the Recycle Bin is emptied, before you can delete unused files.

Authorization for Advanced Uploading of Files

When an editor uploads files to the web server, it is done using EPiServer's link tool or File Manager. It is possible to allow a group or individual to use advanced uploading. This means, for example, that an editor can upload several files at a time by using a drag-and-drop operation from Window's File Manager to EPiServer's File Manager.

Give certain groups/individuals access rights to advanced uploading as follows:

1. Select **Permissions for Functions** on the **Config** tab.
2. Click **Edit** after **Allow advanced upload of files** using for example drag-and-drop.
3. The groups/users that have access to advanced uploading are displayed in the list.
4. Add or remove the groups and users so that only those who are to have access to advanced uploading are displayed.
5. Click **Save** at the bottom of the page.

Customize Edit Mode

In EPiServer CMS, you can customize Edit mode to best suit your organization and editors and it is possible to modify Edit mode so that it functions differently for different people. One of the main strengths of EPiServer CMS is that it is easy to use. As "easy" means different things to different individuals, Edit mode needs to be customized to be easy for everybody.

It is possible to customize a lot of different functionality in Edit mode. For example:

- available page types
- fields found on a certain page type
- which tabs the fields are placed on
- who can see which tab
- who can create pages based on a certain page type
- default values for fields

It is also possible to customize the Editor and define which buttons will be available to edit content. Finally, we can also work with the language in different sections, so that everybody understands the role of the different fields. A lot of the customizations to Edit mode are done by the solution provider, but it is useful to know what to do if changes are necessary.

The majority of these settings are done in the **Page Type** tab in Admin mode. There are also a number of functions there for creating new page types, copying existing page types, editing properties, etc.

Working With Page Types

EPiServer CMS works with both page templates and page types. Page templates are .aspx pages that are displayed to visitors and contain the functionality required to perform a specific function. Page types, on the other hand, are forms, based on a certain page template, in which the editor can enter information in EPiServer CMS. Work in the Edit and Admin modes primarily involves page types.

Create / Edit Page Types

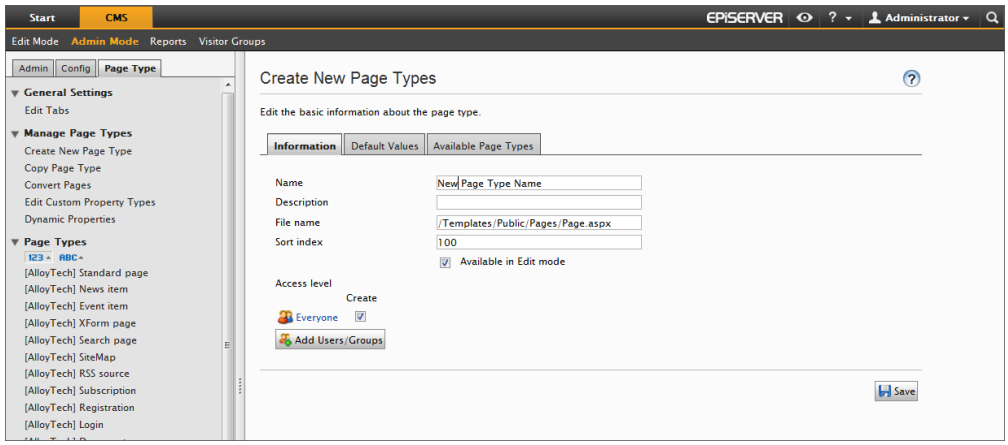
Administrators do not often create new page types as this job is often the task of developers and is done during development of the website. Each page type is linked to an .aspx page (page template), where all the functionality of the page is programmed. The fields in the form that are filled in by the editors are created in Admin mode.

To edit the settings for an existing page type, open the page type in question on the **Page Type** tab, click **Settings** , change the settings according to your requirements and click **Save**.

Create a new page type

1. Select **Create New Page Type** on the **Page Type** tab.
2. Enter a name for the page types in the **Name** field. This text is often determined by the language files.
3. Specify text that describes the page type. The description is the information displayed to editors when they select to create a new page.
4. Specify the path and file name of the page template that is to create the visitor's web page.

5. Specify if the page type can be selected in Edit mode by selecting **Available in Edit mode**.
6. Specify the index amount for the page in the **Sort index** field.
7. If you would like access to the page type to be restricted for certain users and/or groups, specify this by changing the users and groups under **Access level**. The access level is a filter, which defines which users will have access to create pages based on that page type.
8. Click **Save**.



Information Tab

Name

The page type's name is entered here.

Description

The text describing the page type is entered here. The name and description are the information displayed to editors when they decide to create a new page. This text is often managed by the language files in EPiServer CMS.

File name

This field is for entering the path and file name for the page which will create the actual web page shown to the visitors. This name should not be changed, as each template is linked to a particular page to ensure that it always has the right layout and comes with the all the relevant information.

Sort index

The sort index determines how the various page types will be sorted in the list when the editor decides to create a new page. The page type with the lowest sort index comes first. All new page types have 100 as default.

If you want to sort the list of page types in the **Page Type** tab in Admin mode according to the sort index, click the "123" icon under the **Page Types** heading.

Available in Edit mode

When you select the option Available in Edit mode you are providing the editors with the possibility to create pages based on this page type.

In some modes it may be appropriate to hide some of the page types from the editors. This might apply, for instance, to the discussion forum template. It is quite usual to want to have just one discussion forum on the website, as opposed to several in different places. In this case, you create a single forum and then hide the page type. Editors will therefore not be able to access the template, thereby preventing more than one forum being created.

Access level

Access level on a page types is a filter in the interface, which is used to determine which editors can create pages based on certain page types. When a new page type is added, it automatically receives the access level so that Everyone can work with the it.

1. Delete the access level for a group or user that currently has access level to create pages based on a certain page type by clearing the **Change** check box and then clicking **Save**.
2. To grant a user or group access to create pages based on a certain page type by clicking Add users/Groups and selecting the relevant users or groups. Make sure that the **Create** check box is selected and click **Save**.

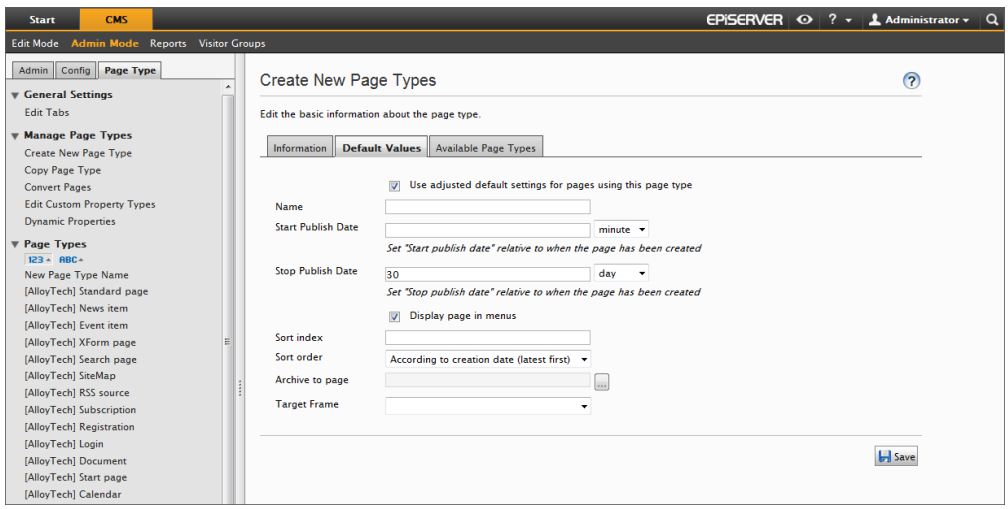
Default Values Tab

A lot of fields (properties) in EPiServer CMS page types cannot be removed. It is, however, possible to set default values for these properties to help editors with their work.

An example of when this may be used is in the "News Item" page type. If there is a policy for news on the website that states that news is "old news" 30 days after publication. After that it must be moved to the News Archive. These two properties could then be set as default on that particular page type.

Change the default values of EPiServer CMS properties

1. Select the page type for which you want to specify default values on the **Page Type** tab.
2. Click **Settings** and then the **Default Values** tab.
3. Specify the default values for the fields you wish to have values for.
4. Click **Save**.



Available Page Types Tab

It is possible to select the page types that should be available for editors when they create a subpage of a page based on a certain page type. An example of when this could be used is in the news listings. Subpages to the "News List" page type should be of the page type "News Item". By only providing the editor with access to this page type, there need never be any mistakes as they have no other page types to choose from.

It is also useful to hide page types that editors don't need; the fewer page types the editor needs to choose from, the easier it will be. Once you have created the page you require, e.g. for the registration page based on the "Registration" page type, you no longer need to have it available, until you want to create a new registration page.

EPiServer CMS also checks if an editor attempts to transfer an unauthorized page type to a location in which you have specified that it cannot be created.

Define available page types

1. Select the page type for which you want to define underlying page types on the **Page Type** tab.
2. Click **Settings** and then the **Available Page Types** tab.
3. The **All page types are available** check box is selected by default. Clear this check box and specify which page type(s) you wish to be available under the page type you are editing.
4. Click **Save**.

Creating Containers

Containers are pages that can be used to store and edit data on the web page. It is possible to use the data through developer APIs, the fetch data functionality as well as dynamic content page properties. Containers are not displayed in View mode and can therefore only be edited in Edit mode. Containers appear dimmed in the tree structure.

Container pages:

- have no preview
- are not searchable, since they are filtered from page searches
- do not have a target frame or simple address
- are not visible in menus
- cannot be linked to
- cannot be compared visually or side by side

You can create a container page type in the same way as other page types without defining a file name.

Copy Page Type

To make it easier to create a new page type, it is common practice to copy an existing page type instead of starting from scratch every time. When you copy a page type, all of the properties are also copied, so you don't need to create them again. You can then edit the information for the page type and its properties.

Copy a page type

1. Select **Copy Page Type** in the **Page Type** tab in Admin mode.
2. Select the page type that you want to copy from the drop-down list and click **Copy**.
3. A window will appear containing the exact same properties and you can then edit the page type information to suit your requirements and click **Save**,

You can create a container page type in the same way as other page types without defining a file name.

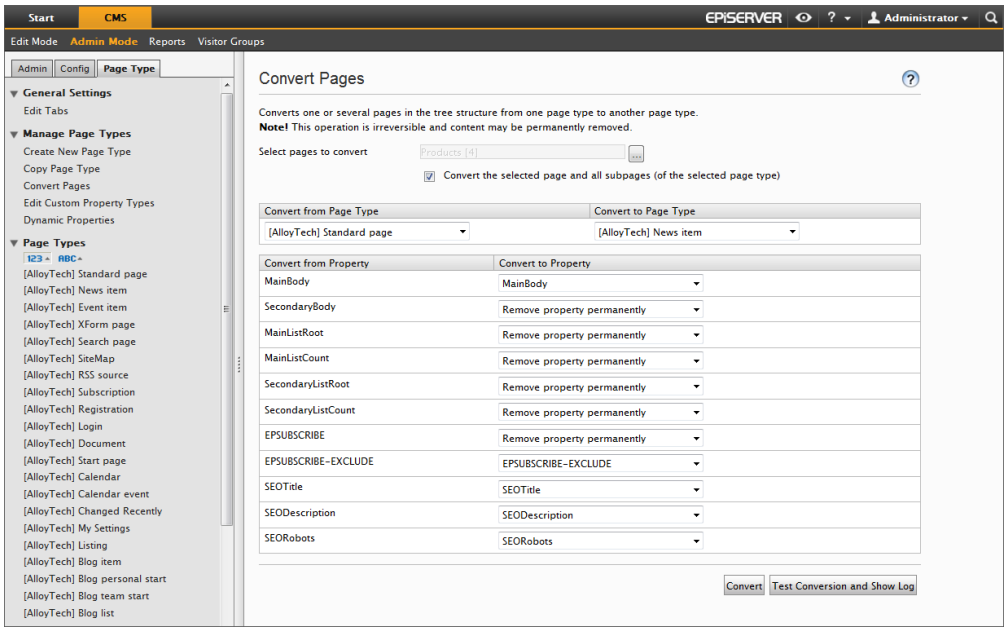
Convert Pages

In EPiServer CMS it is possible to convert one or several pages in the tree structure from one page type to another page type. This may be useful, for example, if you have built up a page structure with all pages based on the page type **Standard page** and you then decide that you would like to use the information on the pages as news items, based on the News item page type. The easiest way to do this would be to convert the pages so that they use another page type.

Each page type contains a number of different properties. These properties generate the fields in the page type. When you convert a page, you in reality convert the content in the page properties in the old page type to corresponding properties in the new page type.

It is important to note that at conversion, all versions of a page are converted, not just the published version. Make sure that no other users are currently editing pages of the page type to be converted on the pages to be converted before you convert the pages.

1. Select **Convert Pages** from the **Page Type** tab in Admin mode.



2. Select a page to be converted in the **Select pages to convert** field and select the **Convert the selected page and all subpages (of the selected page type)** check box if you want all the sub-pages to be included in the conversion.

3. Select the page type that you want to convert from in the **Convert from Page Type** list. When you select a page type to convert from, the list of page properties included in that page type will be updated in the **Convert from Property** list.
4. Select the page type that you want to convert to in the **Convert to Page Type** list. When you select a page type to convert to, the list of page properties included in that page type will be updated in the **Convert to Property** list.
5. Map each page property in the page type that you want to convert to a page property in the page type to which you want to convert the pages.
6. When you have filled in all the properties to be converted, click **Convert**. If you would prefer to first test the conversion first, click **Test Conversion and Show Log**. This will display a detailed log of the pages and page properties to be converted.

Note that a page property in the “new” page type can only be mapped to one “old” page property.

If you select **Remove property permanently**, the page property and its content will not be converted and will instead be deleted. This may damage the content on the website, so think carefully before you decide to permanently remove a property.

Working with Properties

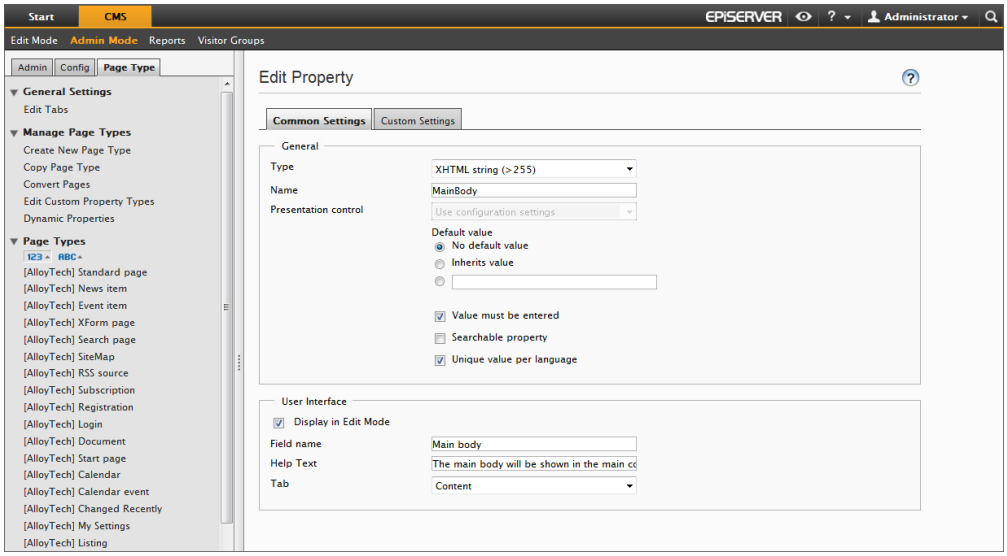
Each page type contains a number of properties and it is these properties that generate the fields in the page type. All properties must be linked with the .aspx page that displays the content on the page and it is for this reason that new properties are not often created.

Add / Edit Properties

The Add / Edit Property dialog is split into two tabs: **Common Settings** and **Custom Settings**.The **Custom Settings** tab is only used for properties that are based on property type XHTML string (>255). Further information about this can be found in the Configuring HTML Editor section.

Add a new property

1. Select the page type where the property is to be added on the **Page Type** tab in Admin mode and click **Add Property**.



2. Enter a name in the **Name** field. This is used when programming and making language settings.
3. Select a property type in the **Type** field. There are a number of different property types to choose from: integer, string, page, date, etc. It is also possible to create your own property types. "Custom Property Types" on page 51.

   If you select type ”XHTML String (>255)”, the **Presentation control** drop-down list will be enabled and you will be able to select whether you wish to use the TinyMCE HTML Editor or the EPiServer CMS 5 HTML Editor.

The **Presentation control** drop-down list will only display the options TinyMCE HTML Editor / EPiServer CMS 5 Editor if the web.config file has been configured to allow multiple HTML Editors. Further information about setting up web.config to allow multiple Editors can be found on EPiServer World.

4. Select whether you want a default value for the property.
5. Specify whether the field should be unique for a certain language if the website has activated support for globalization by selecting the **Unique value per language** check box.
6. Specify if the field is compulsory or not with the **Value must be entered** check box.
7. Specify whether the fields are to be searchable or not with the **Searchable property** check box.
8. Specify whether the field is to be visible in Edit mode with the **Display in Edit mode** check box.
9. Specify a heading for the property in the **Field name** field. The text is displayed for the editor when using the page type, if the field is not language-encoded.
10. Specify help text in the **Help Text** field. The text is displayed for the editor when using the page type, if the field is not language-encoded.
11. Select the tab on which the property is to be displayed, in the **Tab** drop-down list.

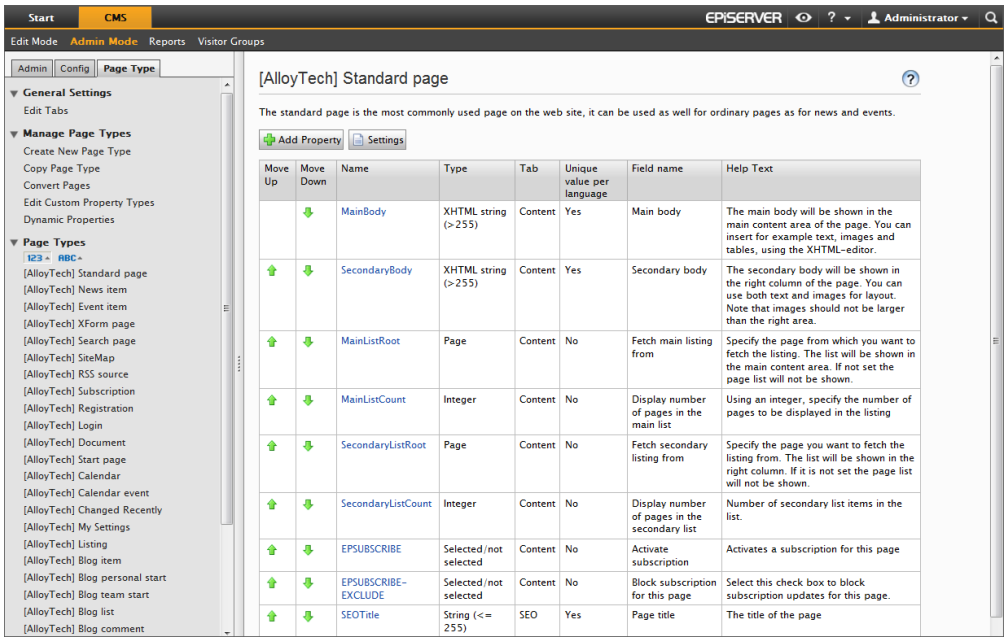
Field Order in the Page Types and Tabs

All fields (properties) that the editor can fill with information have a certain order, which is managed on the page type in Admin mode. All properties can be placed on different tabs and each tab can have different access rights. Using these functions, we can create as simple a form as possible for the editors.

Those who require the advanced sections the least only have access to the tab(s) that the most common fields are placed in.

Change the field order

1. Select the page type for which you want to change the order of the fields on the **Page Type** tab.



2. All properties (fields) are listed on the page in the order that they are in now.
3. Move a property by clicking the up or down arrow depending on the order you want.
4. It is also possible to drag and drop the properties to the appropriate position.

Select which tab a certain property is to be placed on

1. Select the page type with the property you wish to place on a particular tab on the **Page Type** tab.
2. Click the name of the property that you wish to modify.
3. In the **Tab** drop-down list you can select on which tab the relevant property is to be placed.
4. Click **Save**.

Default Value of Field

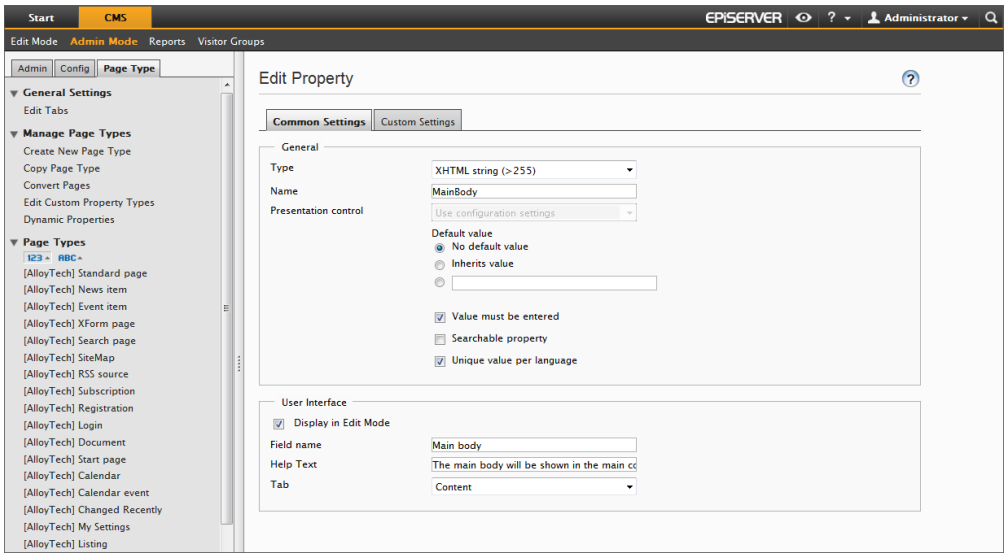
The fields in which an editor enters information are called properties. All page types contain different properties and have different functions. In EPiServer CMS, there are certain properties that create default fields on all templates and these cannot be removed. Each page type also contains a number of properties so that the page can be created according to the template's functionality. Sometimes it is helpful to give the editors default values for the fields. It is possible to specify the default values on both the individual fields and the default fields.

One scenario can be that we use the Writer field where each editor must enter their name when they create a page. Normally, an editor is responsible for one section of the tree and nearly always creates subpages for a main page where they have access rights. We can then apply this property to inherit

the value from the main page as a default value. The Writer field then always has the name that is on the main page as default. It is always possible for an editor to change a default value.

Select the default values for a specific property

1. Select the page type where the property is, on the **Page Type** tab.



2. Click the name of the property that you wish to modify.
3. In the **Default value** section you can choose not to specify default values, for it to inherit values or to specify a custom value.
4. Click **Save**.

Mandatory Fields

It is possible to make fields mandatory so that editors must fill them in before saving the page. The setting is configured per property and page type. Editors that save a page without filling in the mandatory fields will receive a warning that the page cannot be saved until a certain field has been filled in.

Set mandatory fields

1. Select the page type that includes the property that you wish to enforce on the **Page Type** tab.
2. Click the name of the property that you wish to modify.
3. Select the **Value must be entered** check box.
4. Click **Save**.

Field Searchability in the Search Engine

In EPiServer CMS there is a function for searching the website content. The search does not need to be carried out in all fields and it may be of benefit to remove certain fields from the search so that the search engine does not give hits on things that may not affect the content on the page, for example, the Author field. As default, all the fields are searchable.

This only affects the EPiServer CMS search engine.

Exclude a certain field from the search

1. Select the page type where the property to be searched is, on the **Page Type** tab.
2. Click the name of the property that you wish to modify.
3. Clear the **Searchable property** check box.
4. Click **Save**.

Hiding Fields in Edit Mode

You can hide fields in Edit mode so that editors cannot change the value that is saved for the property. The setting is configured per property and page type. As default, all the fields are visible in Edit mode.

Hide a field in Edit mode as follows:

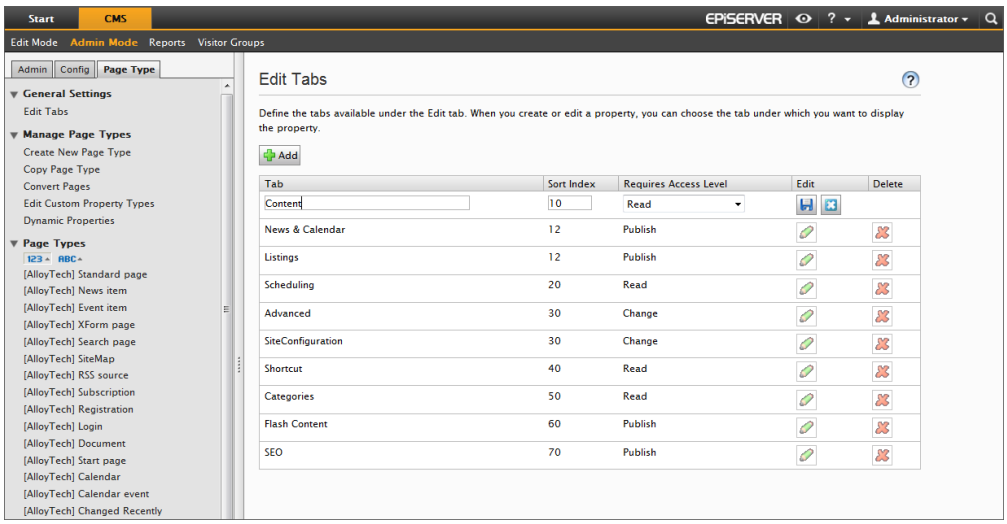
1. On the **Page Type** tab, select the page type and locate the property you want to hide.
2. Click the name of the property and clear the **Display in Edit mode** check box.
3. Click **Save**.

Adding, Editing and Deleting Tabs

You can divide your properties onto several tabs by selecting the Edit Tabs function. Here you can also decide in which order the tabs should be in, and apply access level to a certain tab.

Adding and Editing a Tab

1. On the **Page Type** tab, select **Edit Tabs**.



2. Click **Add** to create a new tab. Click the **Edit** icon to edit a tab.
3. In **Tab**, name the tab.
4. In **Sort Index**Specify the index amount for the tab. The lower the value the further to the left the tab is placed.
5. In **Requires Access Level**, you can select which access level should apply for an editor to see the tab. It is linked to the access the editor has for the page.
6. Click **Save**.

Deleting a Tab

Delete a tab by clicking the **Delete** icon to the right.

Custom Property Types

A property is defined by its name and data type and the page type it is connected to. EPiServer CMS has a number of standard data types for properties (built-in) and a whole infrastructure in place to make it easy for editors to work with the properties and their values as they create and maintain web pages.

There are two types of page property in EPiServer CMS:

- **Standard properties** are always present and are supplied by the EPiServer CMS runtime system. For example, PageName (name of the web page) and PageStartPublish (start publish date for the page), see EPiServer CMS Online SDK.
- **Custom properties** are handled in Admin mode by developers and administrators and can be divided into two types: static properties and dynamic properties.

Static properties are created, named and given a data type and other settings in Admin mode. The editors only have to provide a value for the property in Edit mode.

Dynamic properties are used to store a property on one or more pages simultaneously. The different dynamic properties that are available vary and are set up under **Dynamic Properties** in Admin mode. Dynamic properties are always stored on the page that you have selected and all its sub-pages. If you have other values stored on one of the sub-pages, you can choose to retain them or overwrite the values. Dynamic properties are added in the same was as other properties, although from the **Dynamic Properties** menu option on the Page Type tab in Admin mode, see Working with Properties.

You can always create new types by inheriting from the existing ones. The following property data types are a selection of available types in EPiServer CMS:

Data Type Name	*Data Type Name in Admin Mode*	*Description*
Boolean	Selected/not selected	True or false.
Number	Integer	An integer.
FloatNumber	Floating point number	Number with decimals, for example, 3.14.
PageType	Page Type	EPiServer CMS page type defined in Admin mode.
PageReference	Page	Link to a web page.
Date	Date/Time	Date and time.
String	String (<=255)	Short text, fewer than 256 characters in length.
LongString	Long String (>255)	Used for text that can exceed 255 characters in length, for example the main Editor field in Edit mode.
Category	Category selection	List of categories.
LinkCollection	Link collection	A collection of links.

Configuring the HTML Editor

The Editor in EPiServer CMS is configured from **Admin** mode under the **Page Type** tab, by selecting a page type and either adding or updating a property of the type ”XHTML String (>255)”.

Properties based on property type ”XHTML String (>255)” will result in an HTML Editor, which can be configured so that you can choose which buttons will be available. The Editor in a standard installation of EPiServer CMS, is a customized version of the open source editor **TinyMCE**. This Editor has many functions, is easy for developers to customize and supports all the browsers supported by EPiServer CMS.

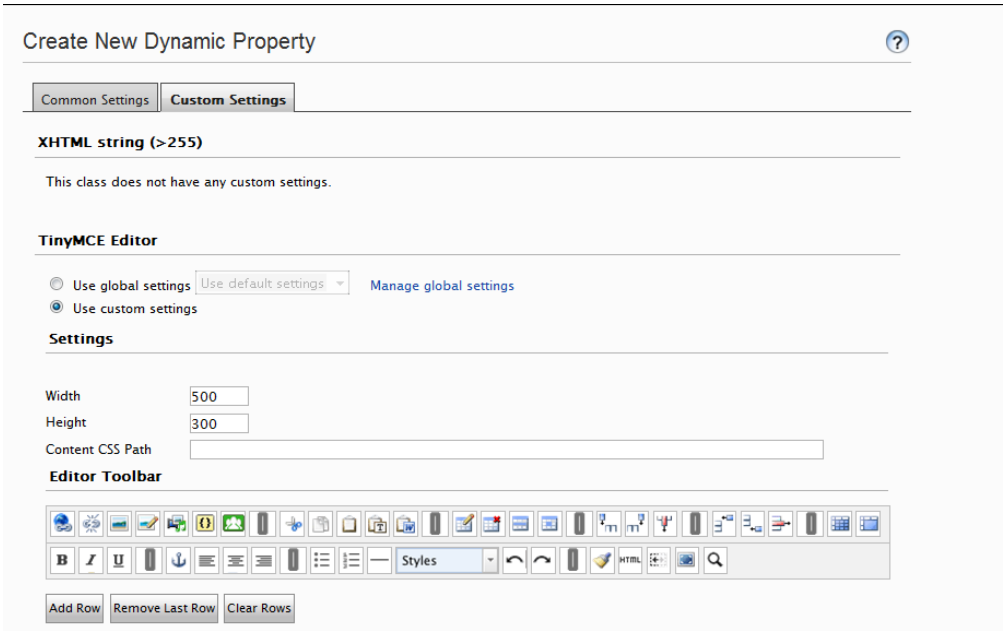
The documentation for EPiServer CMS describes the integrated TinyMCE Editor as it appears in a standard installation, with a selection of functions being available. There are however additional functions, which can be incorporated into the Editor from Admin mode. Refer to the official TinyMCE website for further information about these functions.

The Editor can easily be adapted to suit different groups of editors making different buttons available depending on the functionality required. It is also possible to change the size of the Editor dialog. To change the settings of the Editor, you must firstly have configured the property to use property type ”XHTML String (>255)”. This will enable the fields in the **Custom Settings** tab.

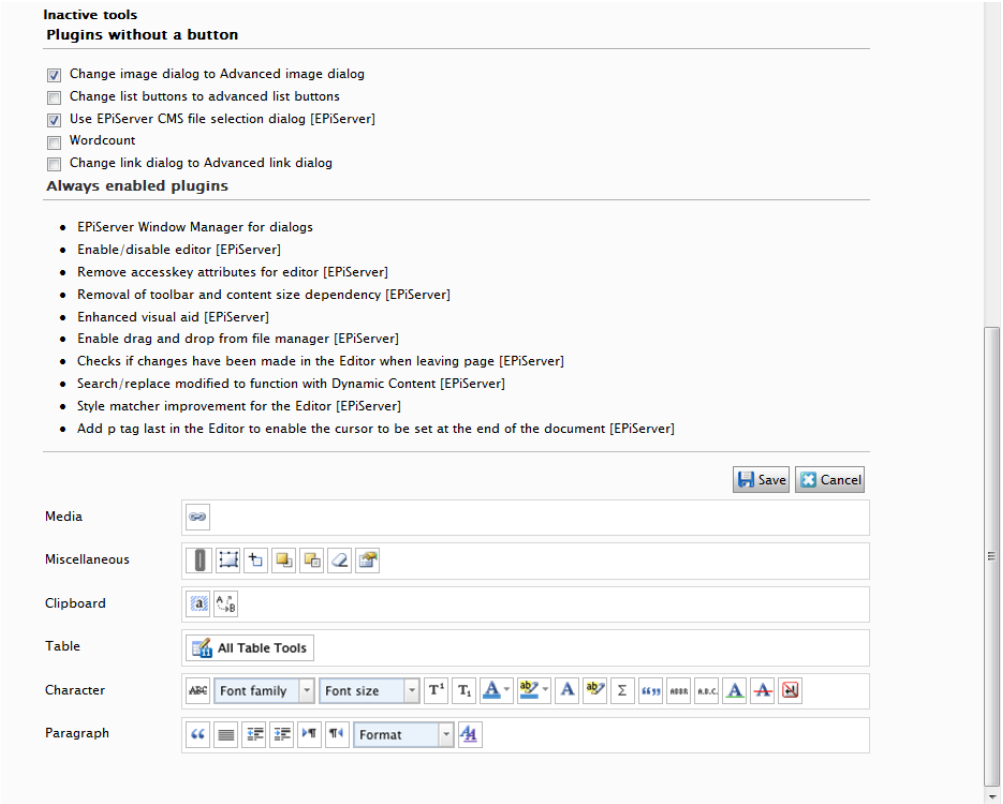
There are two kinds of settings used to change the layout and buttons available in the Editor: global settings and custom settings. The layout of the Editor toolbar is configured in the same way regardless of the type of setting.

Change the layout of the Editor

1. Enter the required width and height of the Editor (in pixels) in the **Height** and **Width** fields.
2. The **Content CSS Path** field contains the path to the cascading style sheet that is to be used in the Editor.
3. Configure the layout of the Editor by dragging and dropping the icons that you want to be available from the Editor Toolbar designer to and from the **Inactive tools** section.
4. Remove an icon by dragging it from the toolbar designer and dropping it in the **Inactive tools** section. The icon will automatically be placed in the category to which it belongs.
5. Add an icon to the Editor by dragging it from the **Inactive tools** section to the desired position in the toolbar designer.
6. Add and remove rows from the Editor by clicking **Add Row** and **Remove Last Row**. The easiest way to clear all the icons from the toolbar designer and start from scratch is by clicking **Clear Rows**.
7. Configure the plug-ins to be used in the Editor and click **Save**.



In the lower part of the **Create New/Edit Property** dialog, you can chose whether to use an advanced image or link dialog. You can also turn on the wordcount in the Editor, and you can specify if you want to use the EPiServer CMS file selection dialog. By default the advanced image dialog is used as well as the EPiServer CMS file selection dialog.



Global Settings

Global settings are used to change the layout and the buttons in the Editor toolbar for a property and can either be used on all page types as a default or on only one page type. You can have as many global settings as you like on your website, making it possible to create an Editor suitable for all editors working with the website.

Configure a global setting for the XHTML String (>255) property

When you apply a global setting to all properties based on the XHTML String (>255) property type, all the Editors on the website using a global setting will be based on this, unless stated that they should be based on another global setting or a custom setting.

1. On the **Page Type** tab in Admin mode, select **Edit Custom Property Types** and click **Add Setting**.
2. Enter a name for the global setting, generally a descriptive name so that you know what the global setting refers to.
3. Change the layout of the Editor, configure the plug-ins and click **Save**.
4. The global setting appears in a list and you can select to set one of the settings as default by clicking **Set as Default**. This means that it will be used for all the Editor on the website unless another setting has been chosen for the property in a certain page type.

If you do not configure a global setting as default, the properties will use the standard toolbar set at installation.

Configure global settings for a property on a page type

It is also possible to configure the global settings for a property on a page type so that one of the global settings is used, but only on this property and on this page type.

1. Open the page type for which you want to change the global settings on the **Page Type** tab in Admin mode.
2. Click the property you want to configure and select the **Custom Settings** tab.
3. Select the **Use global settings** check box and select **Use default settings** if you want to use the default settings for the property.
4. Create a new global setting for the property by clicking **Manage global settings**. Add a global setting by following the instructions on how to configure a global setting for the XHTML String (>255).
5. Change the layout of the Editor, configure the plug-ins and click **Save**.
6. Change the global setting in the drop-down list. Click **Save**.

Delete a global setting

Delete a global setting by opening the setting and clicking **Delete**.

Custom Settings

Custom settings are used to change the layout and the buttons in the Editor toolbar **for this property on this page type only**. Refer to the documentation for global settings if you wish to change the settings globally for the property.

Configure a custom setting

1. On the Custom Settings tab, select the Use custom settings radio button.
2. Change the layout of the Editor, configure the plug-ins and click **Save**.
3. The custom setting will now be used for this property on this page type alone.

Editor Plug-ins

There are two kinds of plug-ins used to configure the Editor toolbar: plug-ins without a button and plug-ins that are always enabled.

Plug-ins Without a Button

The plug-ins that do not have a button enable functionality in the Editor that is not triggered by a toolbar button.

- **Change image dialog to Advanced image dialog** enables an advanced Add/Edit Image dialog box. Select this check box to make it possible to configure advanced properties for your images.
- **Change link dialog to Advanced link dialog** enables an advanced link dialog box.
- **Wordcount** enables the word count functionality. Select this check box to add a **Words** field to the bottom right of the Editor. This displays the amount of words included in the Editor.
- **Use EPiServer CMS file selection dialog**

Always Enabled Plug-ins

There are some Editor plug-ins that are always enabled in a standard installation of EPiServer CMS. These are configured in the Plug-in Manager on the **Config** tab in Admin mode. The following plug-ins are always enabled in a standard installation of EPiServer CMS. "Plug-in Management" on page 94.

- Enhanced visual aid
- Search/replace modified to function with dynamic content
- Automatic resize of dialogs
- Remove access key attributed for Editor
- Enable/disable Editor
- Checks if changes have been made in the Editor when leaving page
- Enable drag and drop from the File Manager
- Removal of toolbar and content size dependency

Non-Standard Editor Buttons

The following buttons are not defined as available in the standard toolbar, but can easily be added:

- **Abbreviation**

The Abbreviation tool in the TinyMCE HTML Editor does not work in Internet Explorer (IE) due to IE not recognizing the element. The tool is fully compliant with Firefox and other browsers supported by EPiServer CMS. This is a known issue for the TinyMCE

development team at MoxieCode. The Abbreviation tool is not enabled in the HTML Editor by default.

- **Acronym**
- **Align full**. After marking a text and pressing this button, the entire paragraph will be stretched over the full page.
- **Citation**
- **Deletion**
- **Direction left to right**. Changes the text direction to “left to right”.
- **Direction right to left**. Changes the text direction to “right to left”.
- **Edit CSS Style**
- **Font family (drop-down)**. Change the font family of the marked text using styles.
- **Font size (drop-down)**. Change the font size of the marked text.
- **Indent**. Indents the marked text using style attribute.
- **Insert column after**. Functionality for inserting a new column after the current table column in the editor.
- **Insert column before**. Functionality for inserting a new column before the current table column in the editor.
- **Insert custom character**. Functionality for inserting custom characters/symbols.
- **Insert new layer**
- **Insert row after**. Functionality for inserting a new row after the current table row in the editor.
- **Insert row before**. Functionality for inserting a new row after the current table row in the editor.
- **Insert/edit attribute**
- **Insert/edit image (simple)**. When this button is clicked and the advanced image dialog is not checked we get a simple image dialog.
- **Insert/edit link (Simple)**. With the advanced link dialog cleared this will show the simple link dialog where you can enter a URL, target and title for the link.
- **Insert/edit link (Advanced)**. With the advanced link dialog checked this will show the advanced link dialog where you can enter a URL and title for the link.
- **Insertion**. Opens a insertion dialog, for setting properties for the <ins> tag that makes a text as inserted.
- **Inserts a new table (table)**
  Inserts a new table in the editor. The dialog that opens makes it possible to change the appearance and layout of the table.
- **Merge table cells**
  Opens a dialog for merging table cells. Cells with be merged within the number of cells/rows specified from the origin of the selected table cell.
- **Move backward**
- **Move forward**
- **Outdent**. Outdent the selected element using style attribute.
- **Remove column**. Remove the column the current column from the table.
- **Remove formatting**. Removes any formatting of the selected element.
- Replace

- **Select All**. Selects all content in the editor.
- **Select background color**. Selection of the background color of the element.
- **Select text color**. Selection of the text color of the element.
- **Split merged table cells**. Splits table cells that have previously been merged by the merge functionality.
- **Strikethrough**. Sets a strikethrough style on the marked element
- **Subscript**. Makes the market text into subscript.
- **Superscript**. Makes the market text into superscript.
- **Table cell properties**. Button that opens the properties dialog for a table cell.
- **Table row properties**. Button that opens a dialog for setting properties for table row.
- **Toggle absolute positioning**. Toggles absolute positioning for a layer.
- **Underline**. Underlines the selected text using a span with style attribute set to text-decoration:underline.
- **Unlink**. Removes any links within the selection.
- **Unordered list**. Inserts an unordered at the current selection.
- **Word count**

Languages in Edit and Admin Mode

EPiServer CMS is available in a variety of European and Asian languages as standard and more languages can easily be implemented. Each editor can select the language that he or she wants to see in Edit and Admin mode. The settings are done on the user's account and are changed by clicking the user name in the top right corner of the EPiServer OnlineCenter window and selecting **My Settings** and then **EPiServer CMS - My Settings**.

Modify Texts in Edit Mode

In order to make EPiServer CMS as simple to use as possible, the language can be modified in Edit mode so that it suits the language that the editors are used to. All text found in Edit and Admin mode is language encoded. The language is controlled several files for each language. LanguageXX.xml is the main file that contains text that applies to Edit and Admin mode, while langXX.xml contains text that applies to the supplied and modified page types and page templates.

You should make the changes in a new XML file which you save in the \root\lang\ directory for the site on your server. EPiServer CMS imports all XML files in alphabetical order; the XML file that is last in this order is the properties that the system imports.

Change existing texts as follows:

Many of our templates have an Author field, in which the editor enters his/her name as being responsible for the page contents. Our organization now wants to replace the word Author with Writer and also wants to change the help text displayed for this field.

1. Start by opening the langEN.xml file in Notepad or other similar Editor. (The Editor must support UTF-8)
2. Copy the file contents to a new file.
3. Edit the properties you want to change, in this case the following:

```
<property name="MetaAuthor">
<caption>Writer</caption>
<help>Specify the author of text that will be used by search engines. If
not set, the value in the dynamic property will be used instead.</help>
</property>
```

4. Change the text you want to amend and save the file with a name that will be listed after langEN.xml when sorted alphabetically.
5. The results can look like the following example. Always make a copy of the file and carry out modifications in the copy, to prevent problems when upgrading EPiServer CMS

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<languages>
<pagetypes>
<common>
<property name="MetaAuthor">
<caption>Writer</caption>
<help>Specify the author of text that will be used by search engines. If
not set, the value in the dynamic property will be used instead.</help>
</property>
</common>
</pagetypes>
</language>
</languages>
```

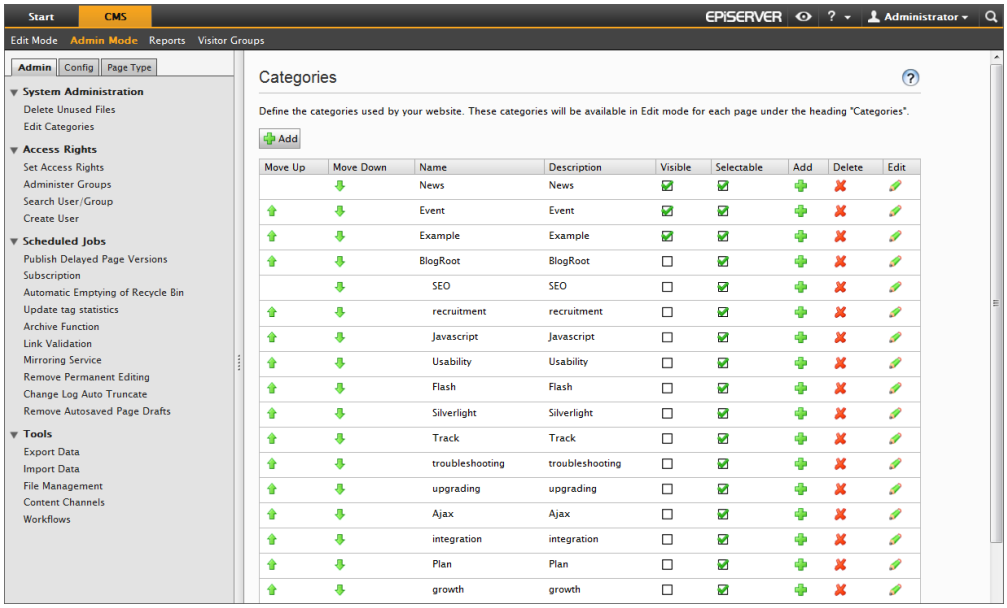
It is not always easy to know what is what in the language files. All properties (fields) are easy to recognize. They always start with <property name="">. The property name is always entered within quotation marks as it is entered in Admin mode. "Working with Properties" on page 46. If you still find it difficult to find, you can use the search function in the program to search for text and replace it with other text.

Categories in Edit Mode

Categories in EPiServer CMS can be used in various ways. It is common to categorize all information in order to achieve an advanced search function or to achieve a personalized list of the pages that interest the visitor most. Not all websites use categories but for those that do, the categories that the editor may use are defined using the Edit categories function. It is possible to specify categories that are to be visible to, and selectable by, the editor here. If the category cannot be selected, it will only appear as a heading.

Add a new category as follows:

1. Select **Edit Categories** on the **Admin** tab.



2. Click **Add** to add new main categories.
3. Specify a name in the **Name** field. This name is used when programming.
4. Enter a description in the **Description** field. This name is seen by the editors when selecting a category. These fields can also be language encoded.
5. Click **Save**.

Click the Edit button to edit information about a category.

Add a category under the selected category. Note that no changes will be made until you click Save after the category.

Move category up the list.

Move category down the list.

Delete a category from the list.

EPiServer CMS Web Help

EPiServer CMS features a web help system, which can be access from the **?** icon at the top of the OnlineCenter window.

The pages are linked to the relevant page in EPiServer CMS and the context-sensitive web help is opened by clicking the Help icon (?) on the appropriate page. All functions in EPiServer CMS have their own .aspx page. By creating an HTML page with the same name and storing it in the Help directory, a question mark will appear on the page. This means that the editor can click the question mark and receive support when working.

Working with Globalization

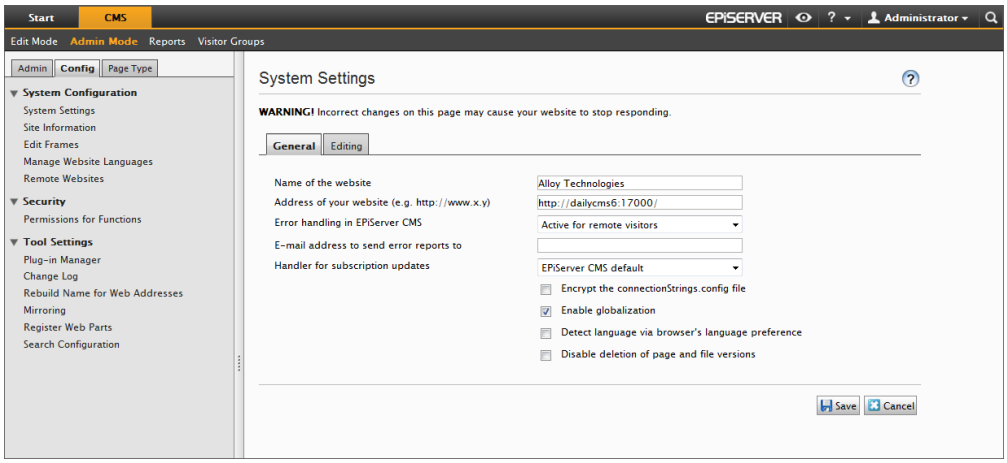
For globalization to work, it is important to set up the support from the beginning. Most of this work is done from Admin mode. It is important to activate the globalization and choose the languages to be available. You should also decide which properties in each template should be translated.

Enable Globalization

Globalization of the website is enabled in Admin mode.

How to activate globalization

1. Select **System settings** on the **Config** tab.



2. Select the **General** tab.
3. Select the **Enable globalization** check box.
4. Click **Save**. The setting will then be changed in web.config.

Default Language for the Website

The default language is the language that applies first when a visitor navigates to the website. Normally you specify the language that you believe the majority of your target group wish to read as the default language. If the visitor wants another language, an active action is required from them. "Which Language is Shown to Visitors?" on the facing page.

The default language for the entire website can be changed in web.config. You can do this by changing the language encoding in the **uiCulture** attribute. If **uiCulture** is not defined in web.config, the website will use the first enabled language in the list of website languages in Admin mode as default.

```
<!-- GLOBALIZATION

This section sets the globalization settings of the application.

-->

<globalization culture="en-US" uiCulture="en" requestEncoding="utf-8" respon-
seEncoding="utf-8" resource-
ProviderFactoryType="EPiServer.Resources.XmlResourceProviderFactory,
```

```
EPiServer" />
```

Which Language is Shown to Visitors?

The language for visitors can be selected and managed in different ways in EPiServer CMS. There are primarily three basic factors that control which language is shown: the default language, the address or the browser's language.

When you manage the languages, you can enter the prefix a language will have. This means that you add the prefix to the address to be able to see the relevant language. If you do not provide a prefix, the language's language code (key) will work as a prefix. If, for example, you want to view the Swedish page of a website that is normally English, you can enter the address www.company.com/sv. The Swedish homepage then appears, provided there is a homepage available.

It is possible to show languages based upon the visitor's browser settings. For this to work, the function must be activated.

Activate detection via the browser's language

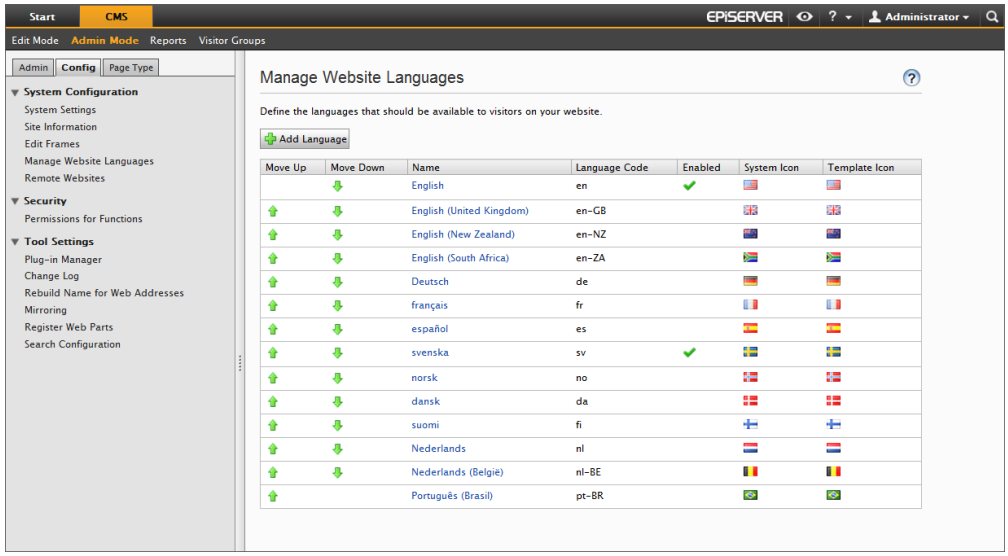
1. Select **System Settings** on the **Config** tab.
2. Select the **General** tab.
3. Select the **Detect language via browser's language preference** check box.
4. Click **Save**.

With this function, the languages that are chosen as the default languages in the visitor's browser will be shown first. This is a setting that every visitor can manage themselves. There are also different programmable properties that can control which language is displayed, for example, a cookie. It is normal for templates to contain a functionality where the visitor can choose the language and for this choice to be stored as a cookie so that the visitor doesn't need to choose the language on their next visit.

Manage Website Languages

If you want your editors to be able to write content in a particular language, the language must be added and activated for use. This is done under **Manage Website Languages**.

When you open **Manage Website Languages**, all the languages added to the website are shown in a long list and you can immediately see which languages are available and whether they are active. Change the settings for a language by clicking the language's name. You can adjust the order of the various languages using the arrows to the left of the language.

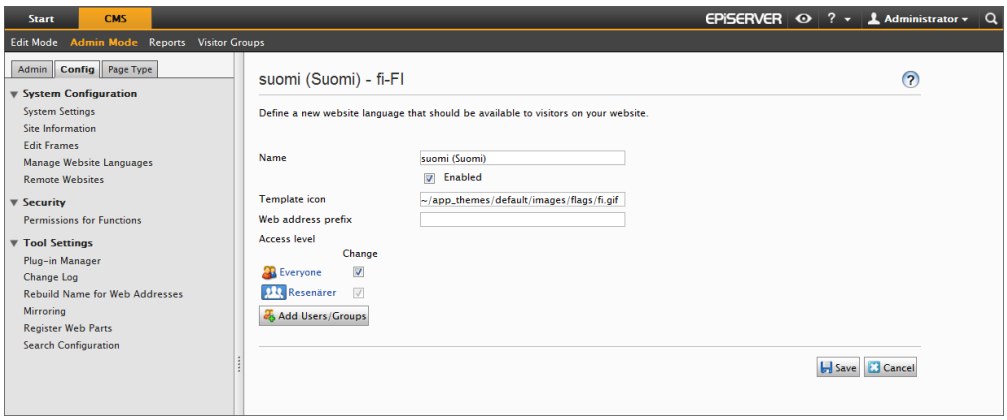


Languages on a Globalized Website

For an editor to be able to create content in several different languages, the language must be added and enabled under **Manage Website Languages**. It is possible to change the access level for a language, so that certain editors cannot create or edit pages in a certain language.

Add / Edit a Language

1. Select **Manage Website Languages** on the **Config** tab.
2. Click **Add Language**.



3. A list of available language codes appears. Select the language code you want your language to use by clicking the name.
4. In the **Name** field, enter a name for that language. This is this name that will be shown to the editor. The field is filled in with the name of the language encoding, but you can change this if you wish.
5. Select **Enabled** if you want the language to be active for editing in Edit mode. The Enabled check box also affects whether the language will be available to website visitors and accessible in View mode. When a language is disabled, it will not be visible in Edit mode. Existing content is however still accessible, but cannot be edited.

Setting a language to disabled is the recommended way of removing a language from a website.

6. In the **Template icon** field you can enter the path to an icon that will symbolize the language to visitors. There are a number of flags stored in the C:\Program Files\EPi-Server\CMS\<release>\Application\App_Themes\Default\Images folder. If you are going to make your own icons, these should measure 15x15 pixels so that they fit properly in the Edit mode.

The images defined in the **Template icon** field are dependent on the image functionality being programmed in the templates on the website.

7. In the **Web address prefix** field, you can provide a specific prefix to be able to easily show the contents of the relevant language. If you do not provide a prefix, the language's language code will be used as a prefix, for example www.company.com/nl. "Which Language is Shown to Visitors?" on page 61.
8. If you would like access to the language to be restricted for certain users and/or groups, specify this by changing the users and groups under **Access level**. The access level is a filter, which defines which users will have access to create and edit pages in a certain language.
9. Click **Save**.

Change the Access Level for a Language

When a new language is added, it automatically receives the access level so that Everyone can work with the language. Users that do not have access levels for a language will not have the language available in the drop-down list in the **Structure** tab and will not be able to edit pages in that language.

1. Delete the access level for a group or user that currently has access level to a certain language by clearing the **Change** check box and then clicking **Save**.
2. To grant a user or group access to create and edit pages in a certain language, click Add users/Groups and select the relevant users or groups. Make sure that the **Change** check box is selected and click **Save**.

Delete an Existing Language

1. Select **Manage Website Languages** on the **Config** tab.
2. Click the language's name.
3. Click **Delete**.

This is not the recommended way of removing a language from a website. This should preferably be done by disabling the language.

Fields that Vary per Language

When you work with globalization, you define in every template which fields will vary depending upon the language by setting whether the property for that field should be "locked" for globalization or "open". This is done by using the **unique value per language** setting.

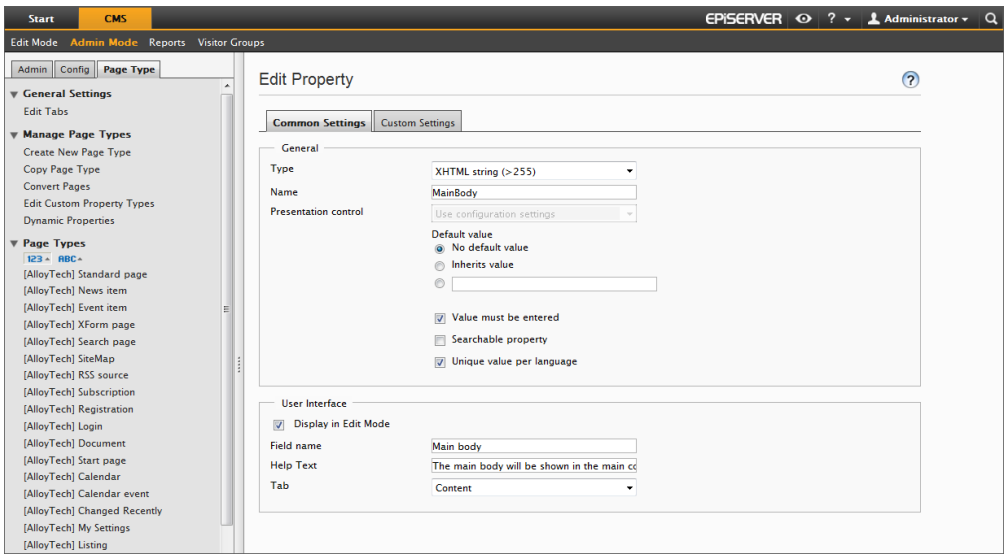
Properties that are set as a unique value per language are editable in all languages enabled on the website. This is normally the case, for example, in the property forming the HTML Editor, so that editors can enter content for the pages in the different languages. Properties that are not set as a unique value per language can only be edited in the language in which the page was created (the original page language). These fields will be grayed out in Edit mode with an icon indicating which language is the original language.

Imagine that the property defining the sort order field is not set as being a unique value per language, i.e. the **Unique value per language** check box is not selected. When you create a new page, you will be able to set the sort order in the original page language, but when you create a version of the page in another language, the sort order field will not be editable. This results in the sort order being the same for all the languages on the website. If you want the sort order to be different for each language, select the **Unique value per language** check box.

When a property is changed from being a unique value per language to not being a unique value, all the existing values for that property will be deleted. This means that if the property for the Editor is changed to not have a unique language, all the text that has been entered in the Editor for all the languages on the website will be permanently deleted.

Set a property to be editable per language

1. Select the page type that contains the property to be set on the **Page Type** tab.



2. Click the name of the property that you wish to modify.
3. Select the **Unique value per language** check box.
4. Click **Save**.

Administering Visitor Groups

EPiServer CMS has a feature called **Personalization**, which is a way to adapt the content on your website to particular groups of your audience, also called visitor groups. A visitor group is a virtual group. As an administrator you can design the visitor groups used for personalized content on your website. You need to be a member of VisitorGroupAdmins to have the access rights to do this as described in "Working with Authorization" on page 9.

You define the expected audience on your site by dragging and dropping various criteria to a visitor group you are creating, for example, “Competitors”, “Potential customers”, “People from Sweden”, “Potential employees”. In EPiServer CMS a lot of the basic criteria are provided out-of-the-box, for example, a geographic location criteria, number of visits, referring search word, and more. But you can also develop your own criteria.

When creating a new visitor group the administrator selects one or more criteria and sets appropriate settings for them. Those criteria will be used to determine if a user visiting the website is a part of that visitor group or not.

For examples about how you can combine the criteria, see "Examples of Adding Visitor Groups" on page 69.

Creating Personalized Content Work Procedure

Create personalized content in the following steps:

1. The administrator creates visitor groups. Also the administrator can set up access rights in Admin mode for a visitor group to a page and its files.
2. The editor selects the content (such as text, images and dynamic content) on a web page to personalize and selects visitor group. Previews the personalized content, publishes or continues to work with the personalized content.
3. If statistics was enabled when the visitor group was being added, the editor can add the **Visitor Group Statistics** gadget in EPiServer OnlineCenter to see how many visitors have visited a page with personalized content.

Available Visitor Group Criteria

In a standard installation the following set of criteria are available:

Site Criteria

- **Number of Visits** will match the number of times the visitor has visited the website.
- **User Profile** will match a value stored in a user’s profile. When adding this criterion you can specify that to be part of this group the user has to have a specific value for a specific profile setting. You can decide if the user profile property should match on equal, contain, start with, or end with a specified value.
- **Visited Category** lets you select one of the page categories on the website. You can also select how many different pages that use the specified category the visitors have visited. The visitor must have visited the specified number of pages that has the specified category set on it.

- **Visited Page** will let you select one specific page on the website. The visitor must have visited the specified page during the current session.

Time and Place Criteria

- **Geographic Coordinate** will look up the visitor's approximate location in a geolocation database (based on the visitor's IP address). Drag the marker and drop it on the map (Google Maps), and select a radius around that point to match the visitors location to the selected location.
- **Geographic Location** will look up the visitor's approximate location in a geolocation database (based on the visitor's IP address) and match their location to the selected location. You can match the visitor to a specific continent, country and region. It is also possible to specify a wildcard for country and region.
- **Time of Day** will match the visitors time period with the start time, end time, and weekdays you have specified.

URL Criteria

- **Landing URL** will store the URL that the user enters the site with. You can decide if the landing URL should match on equal, contain, start with, or end with a specified value.
- **Referrer** will store the URL of the referred page that was clicked before entering the site, for example, a **SERP**[1]. You can decide if the referrer criteria should match on equal, contain, start with, or end with a specified value for the whole URL or parts of it (for example, host name).
- **Search Keyword** will store the URL of the referred page that was clicked before entering the site, for example, a SERP. You need to specify the search word to match against the URL by a regular expression. The regular expression finds the search words in the URL from the most common search engines on the market, for example Google, Yahoo, and Bing.

Visitor Groups

- **Visitor Group Membership** let you select members from one or several existing visitor groups.

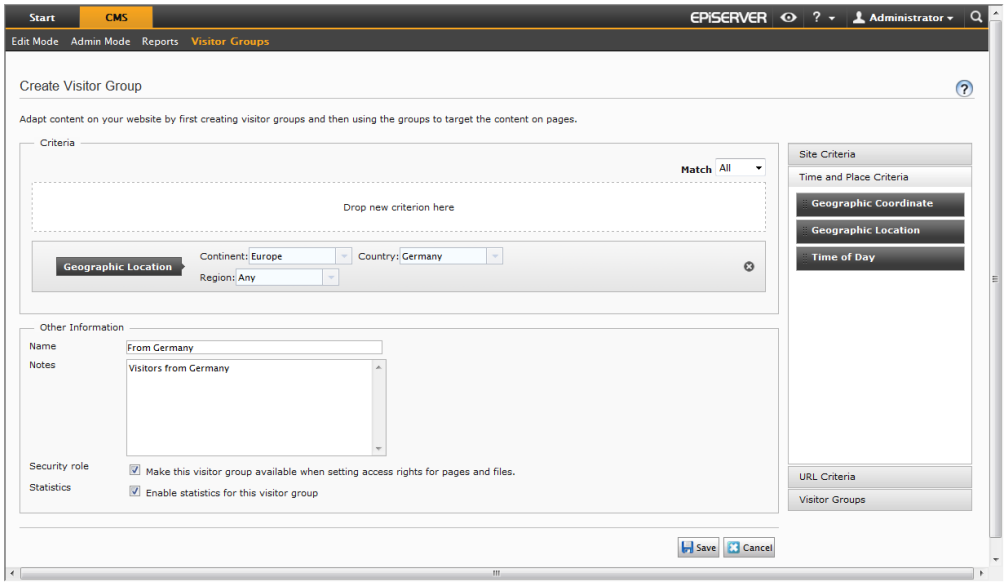
Creating a Visitor Group

Add a visitor group to be available for personalized content as follows:

1. Log in to EPiServer CMS and select the **Visitor Groups** tab. Or, you can right-click and select **Visitor Groups** instead.

[1]Search Engine Result Page. The search engines crawl the internet and show the results on a page. The Title tag of the web page, H1 heading and the lead will be displayed as the text search engine result page.

2. Click **Create**.

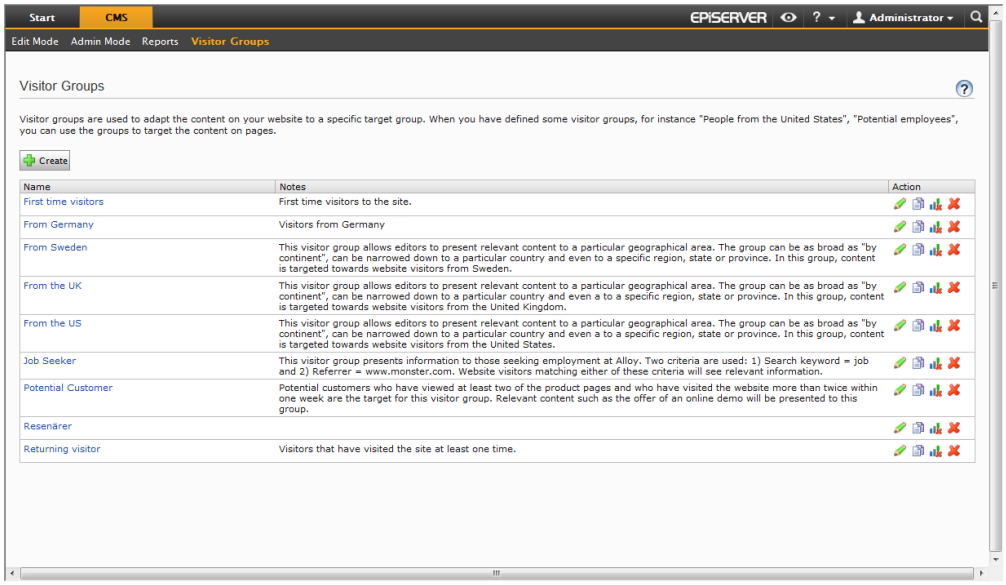


3. In the **Criteria** section, click to add one or several criteria for each visitor group as follows:
   a. In the **Match** drop-down list, select the criteria to match **All**, **Any** or **Point**. What you select here will affect all criteria for the visitor group. Using points is a way to set a value for what an desired action on the website is worth.
   b. Drag the criteria you want from the pane on the right and drop it into the **Drop new criterion here** area.
4. Make the settings for the criteria as described in "Examples of Adding Visitor Groups" on page 69.
5. In **Name**, name the visitor group you have created. This name will be displayed in the personalized content box when you select the content on a page.
6. In **Notes**, type a descriptive text about the visitor group you have created, for example, its purpose. This description will be displayed as a ToolTip when the editor is adding a visitor group to the content on a page.
7. In **Security Role**, select the check box if you want this visitor group to be available when setting access rights for pages and files in Admin mode. Note that visitor groups will only have read access.
8. In **Statistics**, keep the check box selected to enable statistics for the visitor group (this check box is selected by default).
9. When you are done, click **Save**.

Changing a Visitor Group

Change a visitor group as follows:

1. Select the **Visitor Group** tab.



2. Click the **Edit** button for the visitor group you want to change.
3. Add new criteria for the visitor group by drag-and-drop, change the value for an existing criteria, or click ⊗ to delete an existing criteria.
4. When you are done, click **Save**.

Note that if you change the name of a visitor group available in the list for access rights, the settings for this visitor group will no longer work.

Copying a Visitor Group

Copy a visitor group as follows:

1. Select the **Visitor Group** tab.
2. Click the **Copy** button for the visitor group you want to copy. The new copy will have the same name as the original but with the extension “- Copy”.
3. Rename and change criteria for the new visitor group you have copied.

Deleting a Visitor Group

Delete a visitor group as follows:

1. Select the **Visitor Group** tab.
2. Click the **Delete** button for the visitor group you want to delete.
3. Confirm the deletion.

Clearing Statistics

The visitor group statistics are shown by a gadget in EPiServer OnlineCenter. Clear the statistics from the database as follows:

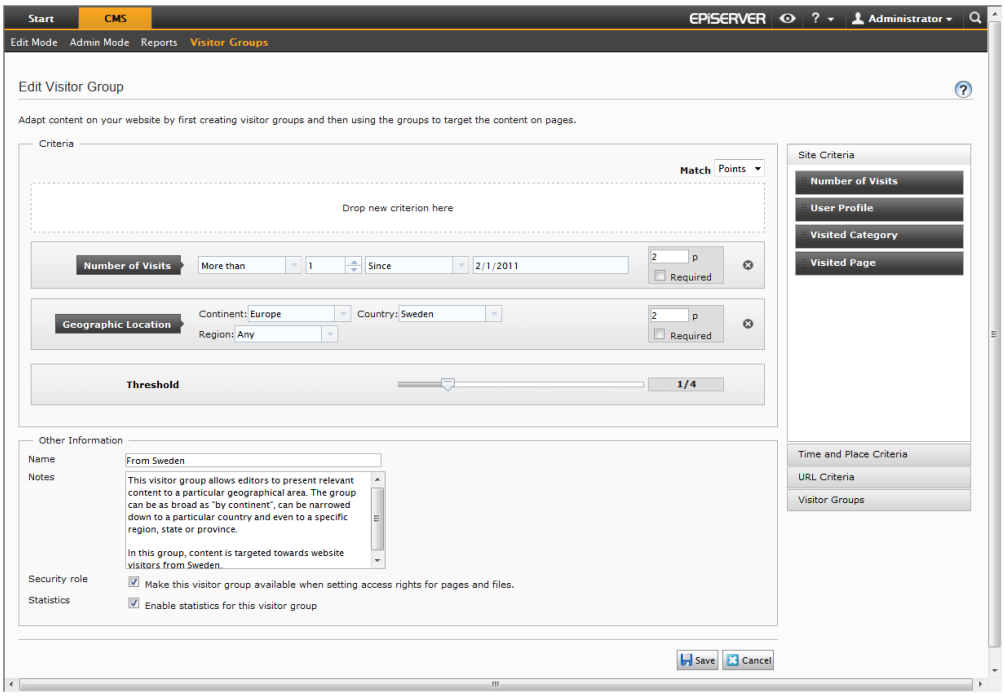
1. Select the **Visitor Group** tab.
2. Click the **Clear statistics** button .
3. Confirm the deletion.

> Note that the clear statistics action takes effect immediately, and actually deletes it from the database. It cannot be undone.

Examples of Adding Visitor Groups

This section provides the following examples on how you can define the visitor groups by combining different criteria:

Points



By using **Points** you can set a value for how much an action is worth, for example, a visited campaign page. In this example, visitors who never has visited the page before will match the criteria for this visitor group.

1. In the **Match** drop-down list, select the criteria to match **Point**.
2. Drag and drop the **Visited Page** criteria, and select page. Use drag-and-drop of the criteria again to add several pages.
3. Drag and drop the **Number of Visits** criteria, and select **Equal** > **1** > **Since [Date]**.
4. Select **Threshold** for the criteria you have added in your visitor group, for example, that the visitor must fulfill 1 of 3 criteria to be included in the visitor group.
5. Save the visitor group.

Geographic Location

You can direct your content to visitors from a specific country and specific days, for example, “People

from Sweden" visiting your website weekends. For example, you can show all Swedish people a clickable banner to sign up for a conference.

1. Drag and drop the **Geographic Location** criteria, and select **Continent**, **Country** and/or **Region**. Use drag-and-drop of the criteria again to add several countries.
2. Drag and drop the **Time of Day** criteria, and select **[Weekday]**. You can also select the personalization to start and end at a specific time.
3. Save the visitor group.

Geographic Coordinate

You can direct your content to visitors from a specific part in the city, for example, "People from Upplandsgatan, Stockholm".

1. Drag and drop the **Geographic Coordinate** criteria, and click **Select Location**.
2. Select location by clicking the map you want to include in the geographic coordinate and click **OK**.
3. Select the **Radius [Number of Kilometers or Miles]**.
4. Save the visitor group.

Referrer

The HTTP Referrers is based on pages, for example, used in a campaign. If the visitors search on "episerver cms" on Google.com and come to the website from the SERP, for example, http://www.google.se/#hl=sv&source=hp&biw=1338&bih=790&q=episerver+cms&aq=f&aqi=g10&aql=&oq=&gs_rfai=&fp=c37e43602e02c461.

Visitors searching "episerver cms", clicking a link to EPiServer's website through the search engine result page on Google.com.

1. Drag and drop the **Referrer** criteria, and select **URL** > **Equals** > **[SERP[1], for example, http://www.google.se/#hl]**
2. Save the visitor group.

[1]Search Engine Result Page. The search engines crawl the internet and show the results on a page. The Title tag of the web page, H1 heading and the lead will be displayed as the text search engine result page.

Administering Scheduled Jobs

Within EPiServer CMS, there are a number of functions that perform their tasks within a specified time interval. One example of this is the subscription function, which sends e-mail messages to all those who have signed up for subscription. You can decide when scheduled functions should be carried out, and how often. In the standard installation there are five different scheduled jobs. It is however possible to develop customized scheduled jobs and these can be found under the same heading in Admin mode.

Go to the **Admin** tab > **Scheduled Jobs**.

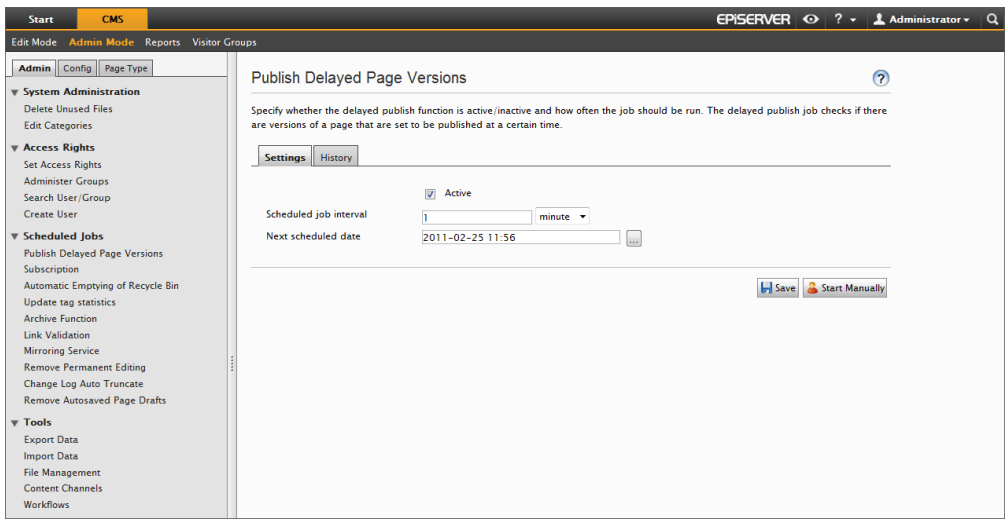
All scheduled jobs has the following tabs:

- **Settings**, where you can make the settings for scheduled job. You have the following options:
  - **Scheduled job interval**. Change the interval for the scheduled job according to the configuration and click **Save**. For example, this will remove page drafts older than 3 days if the default configuration is used.
  - **Next scheduled date**. Change the date for the scheduled job according to the configuration by selecting a date from the calendar and click **Save**.
  - **Start Manually**. You can immediately remove drafts older than 3 days by clicking **Start Manually**.
- **History**, where you can see the log of performed scheduled jobs.

Enabling scheduled jobs and defining the time interval are done in the **episerver.config** file.

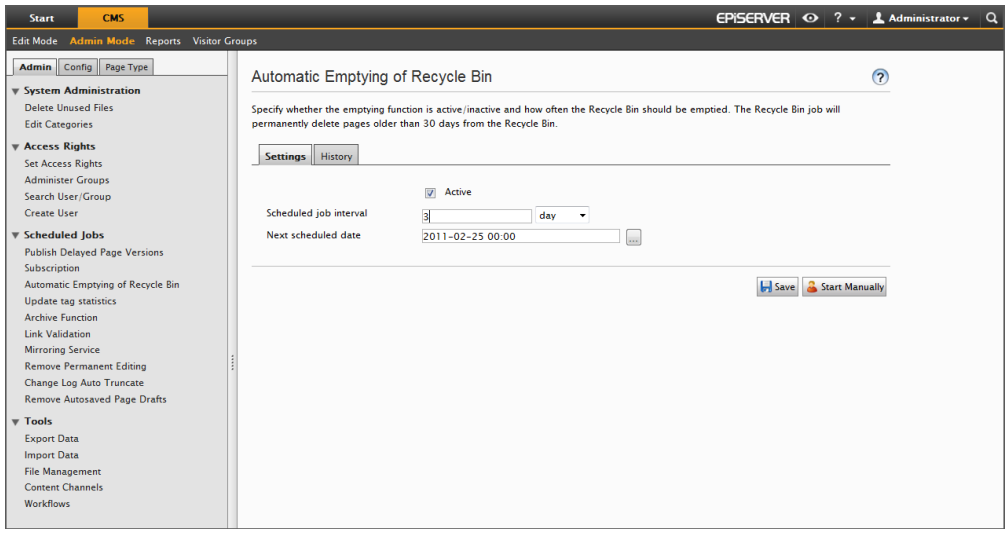
Publishing Delayed Page Versions

Using **Publishing Delayed Page Versions**, you can activate and set up how often the system is to check if there are versions of a page for publication at a set time. Delayed publish of a whole page is not controlled by this; it only takes place when an editor has selected Start Publish for a single version of a page.



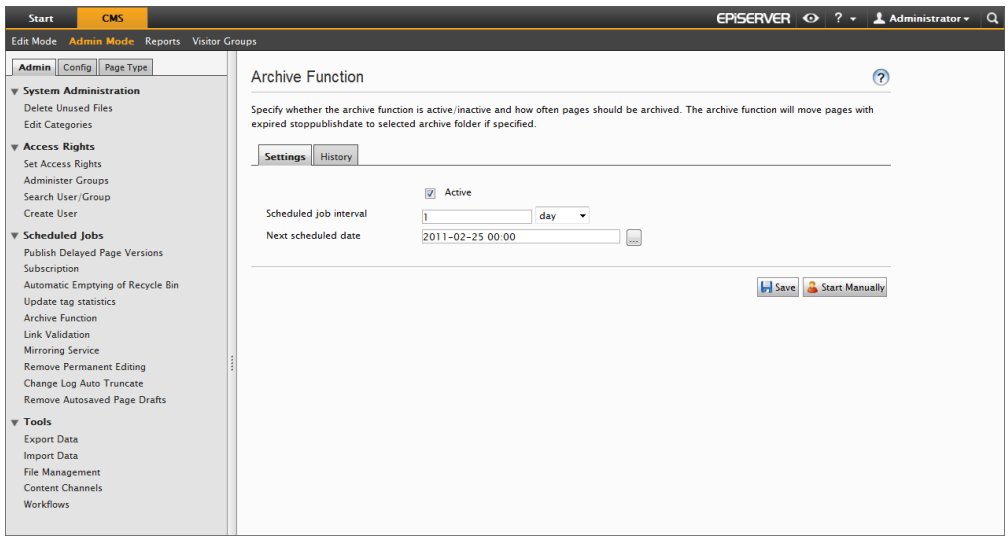
Automatic Emptying of the Recycle Bin

With **Automatic Emptying of Recycle Bin**, you can set how often your Recycle Bin should be emptied. The aim of this function is to stop old information from being left in the Recycle Bin for a long period of time. With automatic emptying, all information that is older than 30 days will be deleted from the Recycle Bin. If you disable the automatic emptying function, you must empty the Recycle Bin manually, by clicking **Empty Recycle Bin** from the toolbar in Edit mode or by deleting individual pages from the Recycle Bin. When you empty the Recycle Bin manually, all information in the Recycle Bin will be deleted.



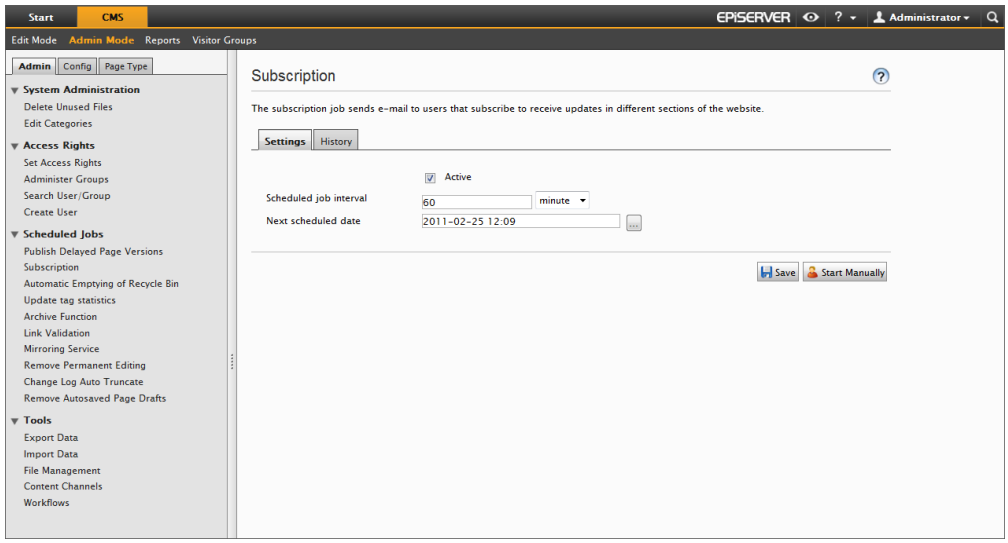
Archive Function

Using **Archive Function**, you can activate and set how often the system is to archive information after the publication period has expired. Note, that there can be a delay between the time when the information stops being published and when it appears in the archive. This may be because archiving only take place once a day. If this is the case, it can take up to 23 hours and 59 minutes before the information appears in the archive, depending on when the page ceased being published.



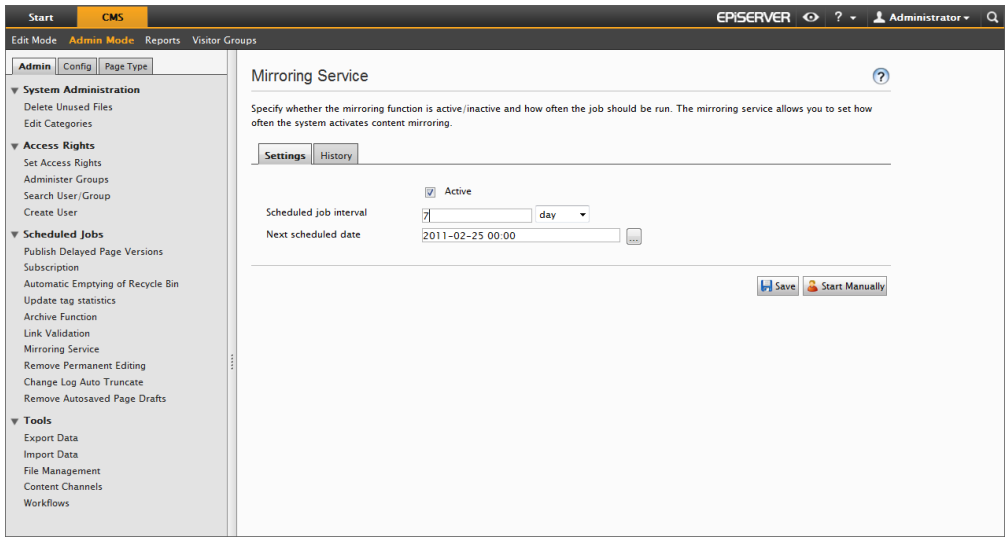
Subscriptions

With **Subscription**, you can activate and set up how often the system is to send out e-mail messages to all visitors who have signed up for subscription. Each visitor selects to time interval to receive e-mail messages from the subscription function. You decide here how often the system is to check for new and updated pages to distribute.



Mirroring Service

Using **Mirroring Service**, you can set how often the system is to activate the mirroring of contents for other websites. If your website is set up to mirror content from one website to another it can be carried out manually by a single editor or automatically by the system. If you have chosen to let the system mirror the information use the **Mirroring Service** function to set when and how often the system is to activate mirroring.



1. To start the job manually click **Start Manually**.
2. Select the **Active** check box to set up deletion of change log items.
3. Select how often the job should be run and click **Save**.

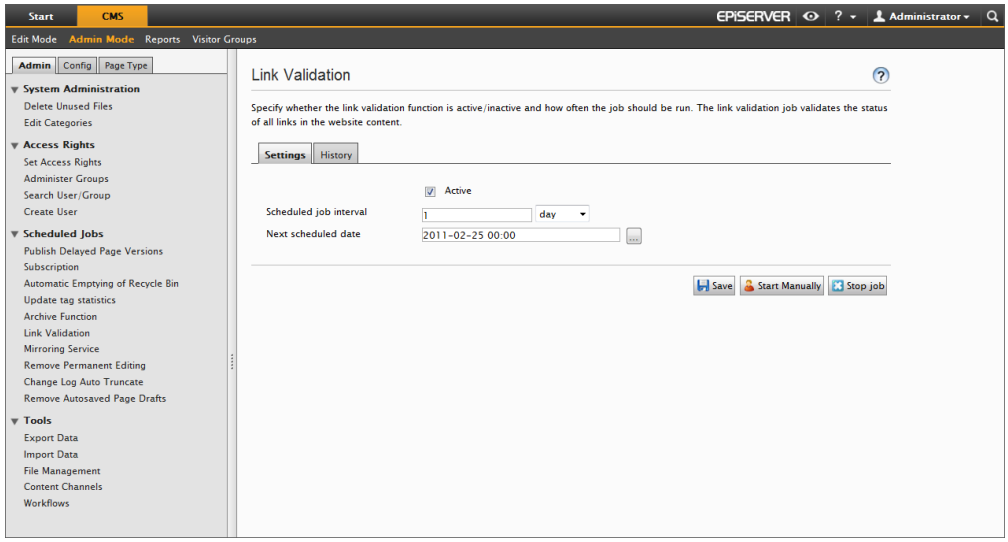
The **History** tab displays the result of previous jobs that have been run. If a job has failed, information will be displayed under **Message** stating why the job failed.

Link Validation

Using **Link Validation**, you can activate a scheduled job which will go through and check all the links on your website to look for broken links. EPiServer CMS will try to contact the target for the link to verify that it is responding.

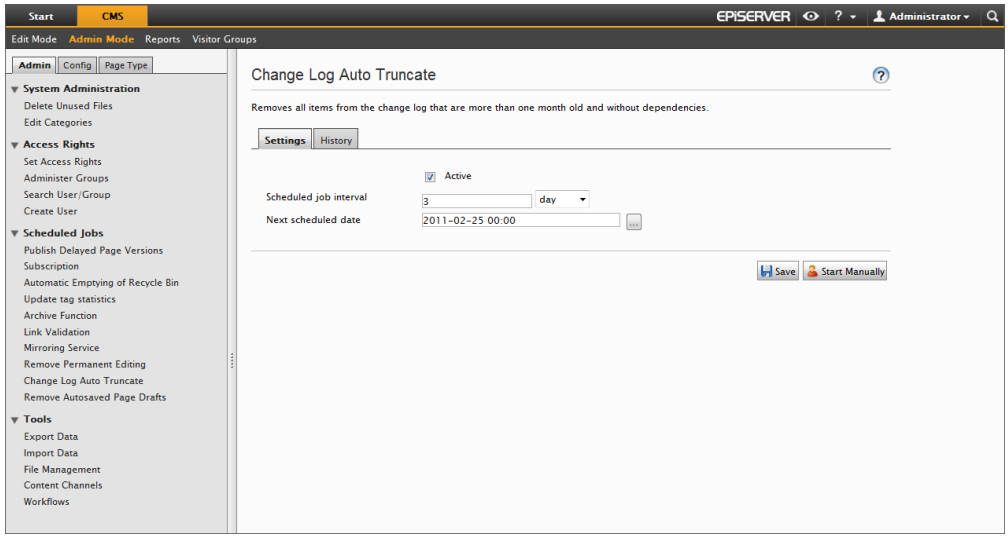
Only links that are unchecked or have been checked earlier than the time when the job started will be returned. The job will continue until no more unchecked links are received from the database. If a large number of consecutive errors are found for external links, in case of a general network problem with the server running the site, the job will stop.

The result of the link validation job is made available as a report called **Link Status**, in the **EPiServer CMS Report Center**.



Change Log Auto Truncate

The Change Log Auto Truncate scheduled job deletes items from the change log that are over one month old and do not have any dependencies registered against them by another part of EPiServer CMS (e.g. Mirroring).

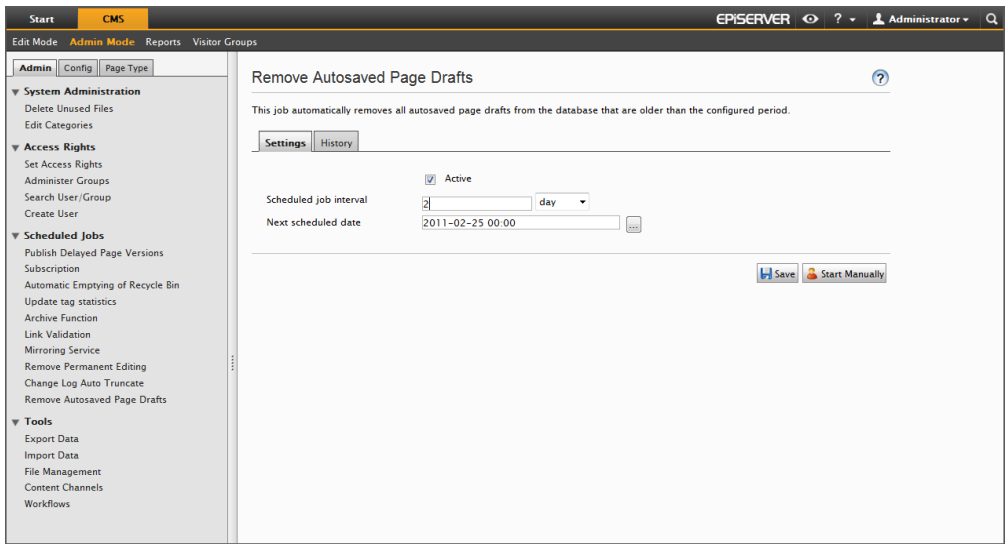


1. To start the job manually click **Start Manually**.
2. Select the **Active** check box to set up deletion of change log items.
3. Select how often the job should be run and click **Save**.

The **History** tab displays the result of previous jobs that have been run. If a job has failed, information will be displayed under **Message** stating why the job failed.

Removing Autosaved Drafts

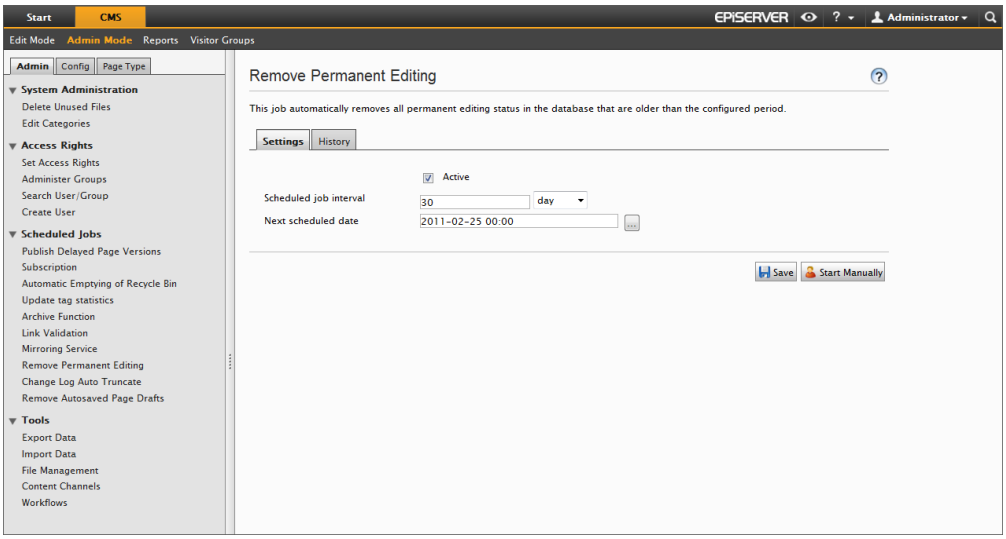
When an editor is working on a page in Edit mode, the **Autosave** function ensures that the work is not going to disappear in case of a sudden crash of the system or browser. It is enabled by default, and any changes will be automatically saved according to the configured time interval.



To prevent the database from being overloaded with page drafts, you can set up when you want to run the scheduled jobs or remove the drafts manually in Admin mode. In the EPiServer CMS configuration the time interval to remove the drafts from the database is set to 3 days by default, which can be changed in the **episerver.config** file.

Removing Permanent Editing

When an editor is working on a page in Edit mode, the **Permanent Editing** function displays if there is someone else working on a page to avoid the risk of conflicts. It is enabled by default.

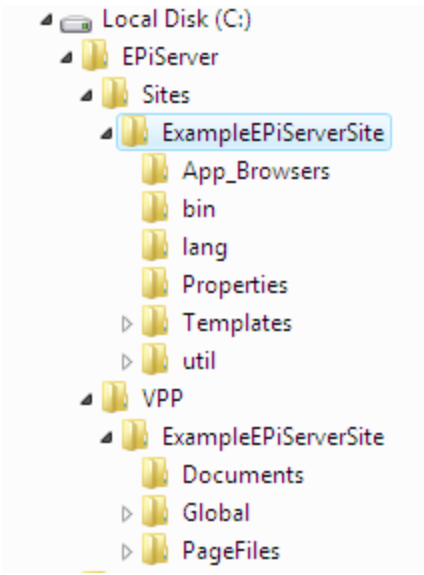


If the editors forget to remove the setting, you can set up when you want to run the scheduled jobs or remove the drafts manually in Admin mode. In the EPiServer CMS configuration the time interval to remove the setting from the database is set to 30 days by default, which can be changed in the **episerver.config** file.

Files for EPiServer CMS

EPiServer CMS websites are installed on the C:/ disk unless otherwise configured at installation. By knowing what is in the various directories, you will be able to customize your website so that your editors' work is made easier.

All files regarding the EPiServer CMS user interface are stored under the "/Program Files" folder on the web server in question. Read more about these files in the Installation Instructions.



Directory Contents

Directory name	*Content*
App_Browsers	Definition files for different web browsers. Used to get the same function to work in different web browsers.
bin	Compilation code libraries for the EPiServer CMS core and website.
lang	Files that control the language in Admin and Edit modes.
Properties	Information about the project.
Templates	Files for the different page types and functions on the website. This folder also contains images and style files for the website.
Util	System files used by EPiServer CMS.

Customizable Directories

When a supplier adapts EPiServer CMS to meet customer requirements, there will be some files under "/Sites" that the developer needs to modify. The main directory that will be changed is **Templates and its subdirectories**.

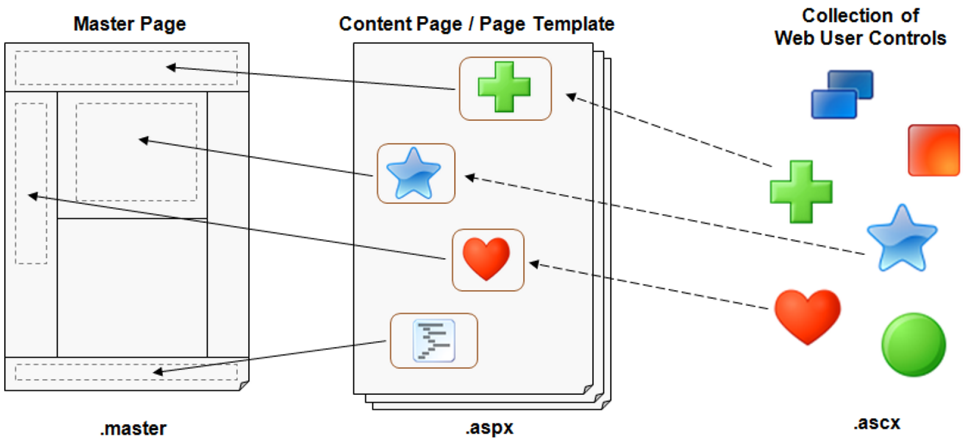
Files associated with the EPiServer CMS system are not located in the same directory and cannot be changed so that the system can be upgraded at a later date and, thereby, acquire new functionality for editors and administrators. If these directories are to be used, it is important not to make any changes to existing files but instead to add new ones. These files are located under the "Program Files" folder on the the web server.

It is fully possible to change the design of the Edit and Admin modes if you wish. In the Application/App_Themes folder there is a directory called Default. Copy the Default folder and change the images and settings in the CSS files to create a new appearance for the Edit and Admin modes.

EPiServer CMS Page Templates

Each website built on EPiServer CMS is customized for each unique organization. A large part of the customization is the design of the website. Another part is the functionality for the visitors. On EPiServer's sample website, there are some ten complete functions that can be used as they are. However, they are usually customized from a technical and design point of view. Certain functions are built completely from scratch.

Each page type is linked to an .aspx file where there is programming for the requested function. However, functions are not always in the template you might think. This is because the entire template setup is built on a working method called **Master Pages**. This is based on always being able to reuse functionality that has been developed. The entire development work means that you build small building blocks (so-called user controls, Units) that are finally placed in a Framework, which the page template then uses.



Design of the Website

The design of the website is primarily controlled by several CSS's. CSS stands for Cascading Style Sheets. In the CSS templates, you define the appearance of various markers in the HTML code. It is possible to give a specific appearance to ready-made HTML markers such as tables, links etc. You will also have the opportunity to create your own classes in which you specify how a certain class should look. The HTML code then invokes the specific class to achieve the desired appearance.

In the standard version of EPiServer CMS, the appearance is controlled by the **Styles.css** and **Editor.css** files which are saved in the **Templates/Public/Styles/Glossy** directory. By editing these files, you can change the font size, colors etc.

Editor Format Template List

In the Editor, editors can choose to format text with different formats from a predefined list. The list is stored in the CSS file that controls appearance in the Editor. It is possible to have different Editors with different appearances on different parts of the website. This is common in, amongst other things, an Enterprise solution, which has a number of websites in the same Edit mode. You define which CSS file will apply for a certain place through a dynamic property in Edit mode. It is possible to add more formats or change the existing ones in a CSS file for the Editor. By adding the text **EditMenuName**

and a name, this format will also be present in the list of various formats in the Editor from which editors can select.

```
h1
{
font-weight: normal;
font-size: 1.7em;
line-height: 1.4em;
padding: 0;
margin: 0.2em 0;
EditMenuName: Heading 1;
}
h2
{
font-weight: normal;
font-style: italic;
font-size: 1.4em;
padding: 0;
margin: 0.2em 0em;
EditMenuName: Heading 2;
}
h3
{
font-weight: bold;
font-size: 1em;
padding: 0;
margin: 0.4em 0em;
EditMenuName: Heading 3;}
```

The Color Selector in EPiServer's Dialog Boxes

In several of EPiServer's dialog boxes it is possible for the editor to choose colors. In the dialog boxes there are font, table and cell properties, amongst others. The scroll list in the color selector has sixteen standard colors as default. It is possible to switch out these colors to your own colors if you wish. This is done by adding a string of the various colors to the **web.config** file.

Exporting and Importing Data

It is possible to export and import data between one EPiServer CMS and another. This function is widely used by developers who build new functionality in a test/development environment. When work is complete and the information is to be put into operation in the live environment, it is useful to be able to export from the test environment and import into the live environment.

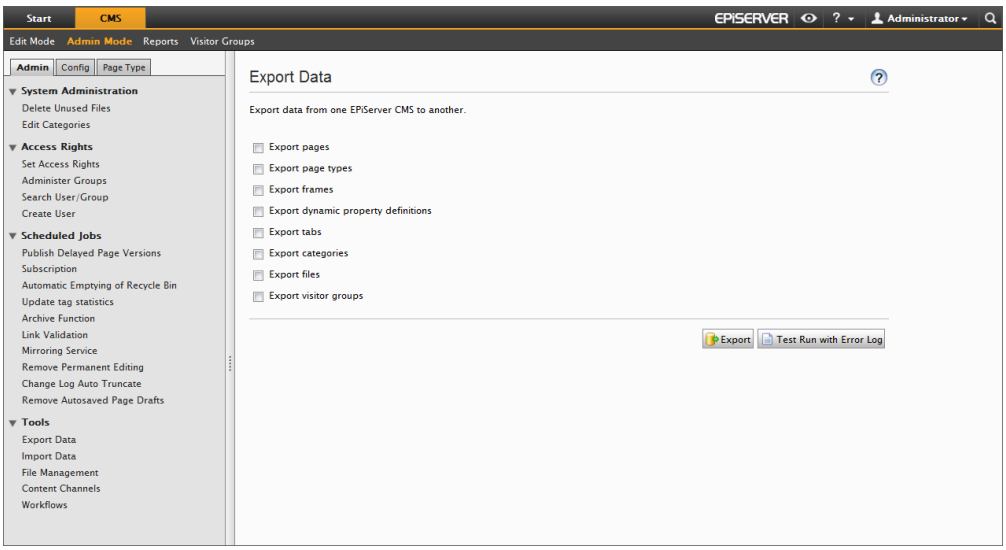
Export Data

The **Export Data** tool is used to export data from the database and/or files to a file that can be imported into another website. This function can be used to transfer information from one EPiServer CMS website to another.

You can export the following:

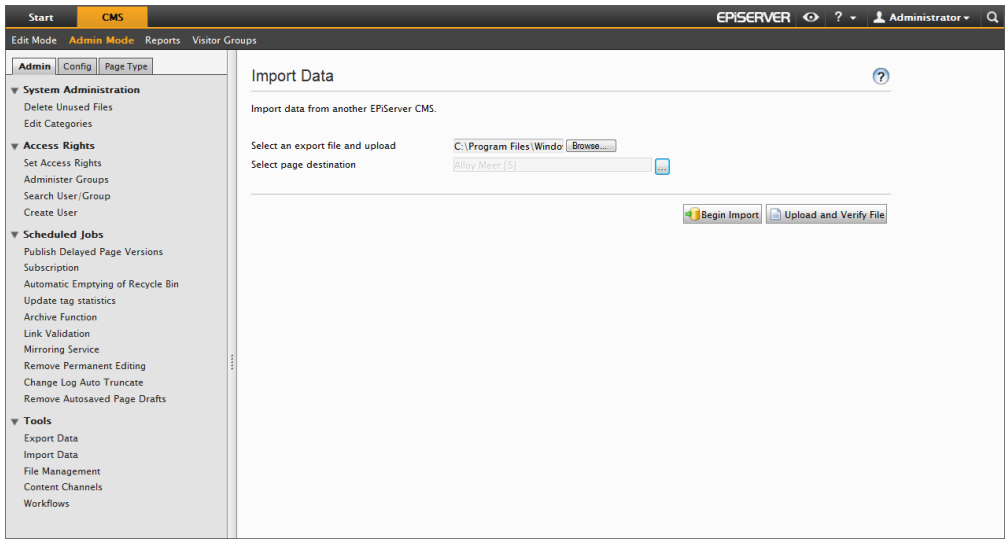
- Pages with (if you so choose) associated page types
- Page types
- Frames
- Dynamic property definitions
- Tabs
- Categories
- Files
- Visitor groups

When you select to export page types, for example, you will see a list of all accessible page types, from which you can select those you want to export. When you have made your selection, click **Export** and indicate where the files are to be saved.



Import Data

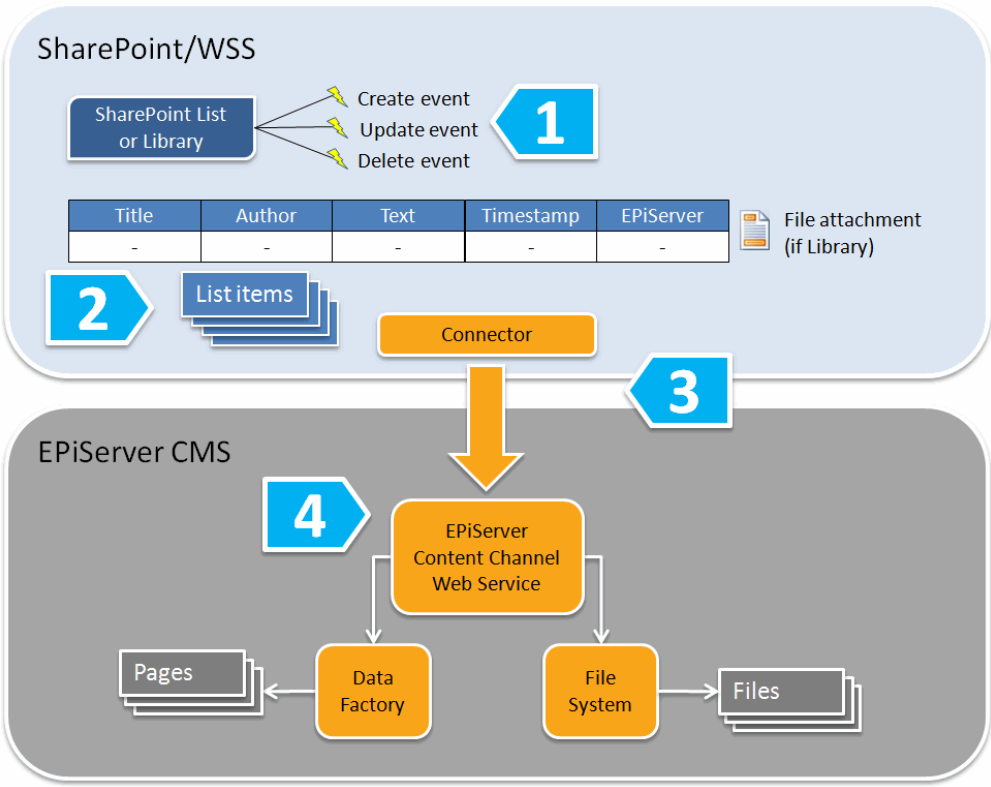
With the **Import Data** function, you can retrieve information that was previously exported from another EPiServer CMS website. Start by indicating the files you want to import. Files must end with .episerverdata in order for the import to function. The files are read and checked. Information on their content is displayed in a list, and you can check that these are the correct parts to import. You must also specify any settings for your new pages, etc. Complete the procedure be clicking **Begin Import**.



Integration with External Systems

The open architecture in EPiServer CMS makes it easy to integrate with other external systems. EPiServer Connect for Microsoft SharePoint Server 2007 allows SharePoint users to have their information published on a website running EPiServer CMS. After installing and configuring the connector, SharePoint users can mark content and documents that should be shipped to the EPiServer CMS website in order to be created and/or published there.

Developers can attach their custom business logic on both the SharePoint side as well on the EPiServer CMS part of such a solution by using built-in events and integration points. For example, when a document is transferred from SharePoint to EPiServer CMS, it should be moved to a specific folder and a page should be created to refer to the newly created file.



Process description

1. Event triggers listening to user actions.
2. User adds or modifies content in SharePoint lists and/or libraries.
3. The connector transforms and transports modified content by invoking the EPiServer Content Channel Web Service.
4. The EPiServer Content Channel Web Service transforms content into EPiServer CMS pages and/or files

SharePoint is an example of an external system that can be integrated with EPiServer CMS. The instructions below can also be followed when using EPiServer CMS together with other external system.

In order for the connector to work, you must first configure the settings in web.config and set up content channels in Admin mode (described below). Further information about setting up the connector can be found on EPiServer World.

Content Channels

Content is retrieved from external systems through channels on the website that define which content should be retrieved, where it should be placed and how it should be displayed. Each channel consists of a number of settings that define the pages, page types and properties to be retrieved. These settings are defined in Admin mode.

Create a content channel in Admin mode as follows:

1. Select **Content Channels** on the **Admin** tab.
2. Click **Add**.
3. Enter a name for the channel in the **Channel name** field. **Note** The name entered in this field is important and is also used when configuring the external system.
4. In the **Page root** field, select where in the tree structure that you want the content from the external system to be displayed.
5. In the **Virtual root** for files field, enter the path in the File Manager where you want any files from the external system to be stored, e.g. ~/Global Files/SharePoint Files.
6. The Save action field defines the status that the pages will receive when they are transported to EPiServer CMS.
   None - Set the save action to None if you don't want any pages to be retrieved from the external system. This can be seen as a way of switching the channel off.
   Save - The pages will be imported so that they have status "Not ready".
   CheckIn - The pages will be imported so that they have status "Ready to publish".
   Publish - The pages will be imported and published.
7. In the **Default page type** field, state which page type you want to be used as default when retrieving content from the external system.
8. Map the page type properties according to the instructions below and click **Add**.

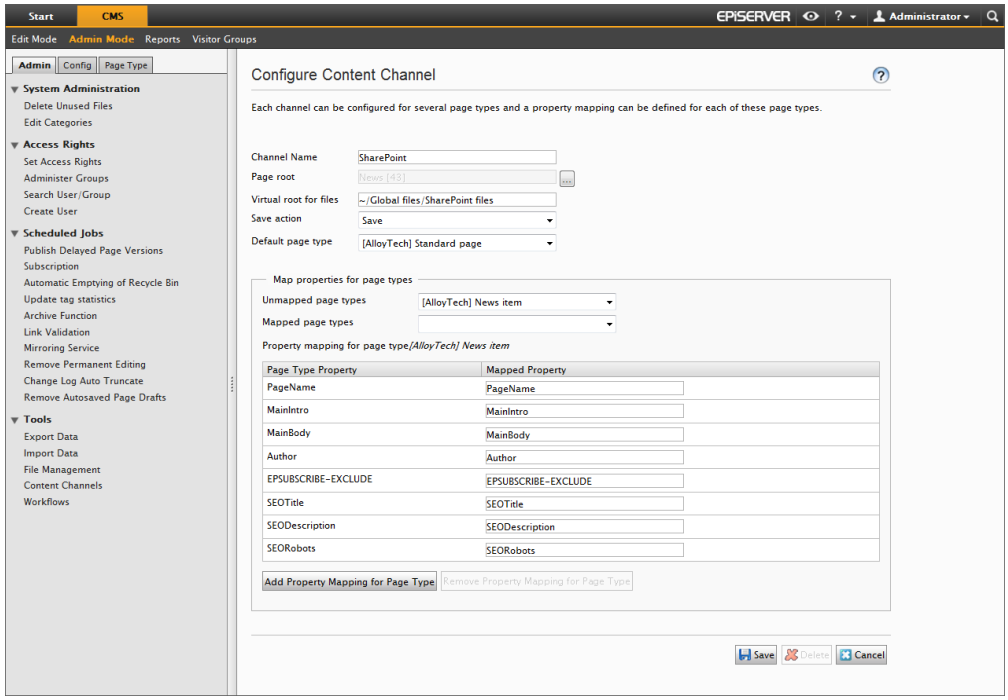
Mapping Page Type Properties

All page type properties in EPiServer CMS have a corresponding property in the external system. It is necessary to map these properties so that EPiServer CMS knows which content should be transported to which property in the page types.

To map the properties for a page type, select the unmapped page type from the Unmapped page types drop-down list. Change the property names in the Mapped Property column so that they correspond to the properties in the external system and click **Add Property Mapping for Page Type**. The page type with the mapped properties will now appear in the Mapped page types drop-down list.

To change the properties for a mapped page type, select the page type from the Mapped page type drop-down list, change the properties and click **Update Propery Mapping for Page Type**.

To remove the mapped properties for a certain page type, select the page type from the Mapped page type drop-down list, change the properties and click **Remove Propery Mapping for Page Type**.



Configuring of EPiServer CMS

Installation of EPiServer CMS is performed through EPiServer Deployment Center, which is encapsulated within a standard MSI-package. The program is downloaded and installed on the machine on which the website is to be stored. Each EPiServer CMS installation is upgraded manually. For further information about installing EPiServer CMS, see the Installation Instructions on EPiServer World.

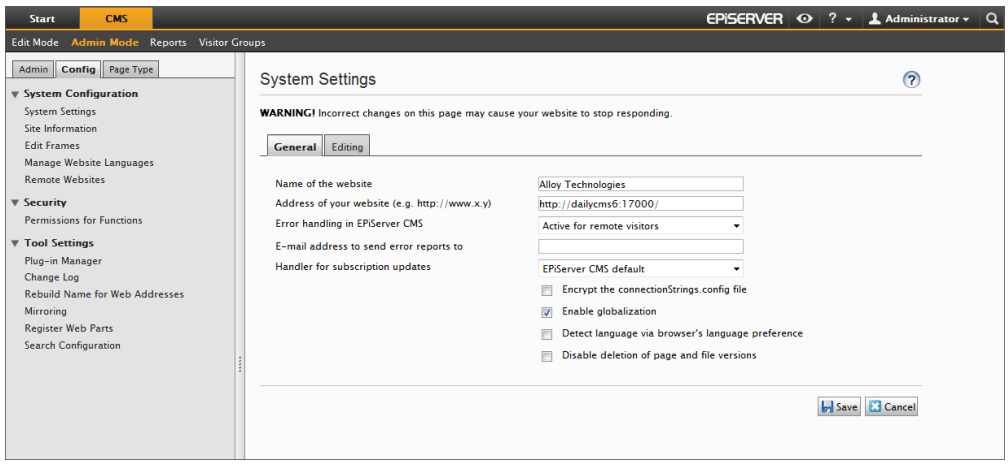
When installing EPiServer CMS in live operational environments, many settings are configured once and then seldom modified. They can include various settings for visitors and editors, as well as purely technical fixes.

The majority of these settings are made in the **Config** tab in Admin mode. Some settings are also done in **Internet Information Services (IIS) Manager**. Other settings are made in files on the server.

System Settings

In system settings, you apply settings for the whole EPiServer CMS installation. You can change error handling, paths to other parts of the web server etc. These settings are usually set during installation and seldom modified.

General Tab



Name of the website

The name that will be displayed in the title bar, at the top of the web browser's window.

Address of your website

The address of a website is used on several occasions in EPiServer CMS, such as in subscriptions. Enter the address of your website in this field, so that the correct information is sent to the users.

Error handling in EPiServer CMS

Select how you would like the errors to be handled, whether it should be active for all visitors, remote visitors or should be disabled.

E-mail address to send error reports to

Enter an e-mail address to which you would like the error reports to be sent.

Handler for subscription updates

There is a subscription function in EPiServer CMS that allows visitors to the website to receive information about new and updated pages. Depending on whether you have a website with EPiServer multi-language support, you can select how the subscription dispatch is to be managed. This list can also include your own solutions for the subscription function.

Encrypt the connectionStrings.config file

The connectionStrings.config file contains sensitive information regarding the system that must not fall into unauthorized hands. One example is information to and from the database. To ensure that nobody can read the information, you must encrypt the file. Select the check box and save the system settings. Information on database connections, etc. is then encrypted.

Enable globalization

EPiServer CMS supports management of contents in several languages/variations of the same page. The way of working with contents management then changes in Edit mode. To gain access to this support, globalization must be activated. This is done by selecting the Enable Globalization check box.

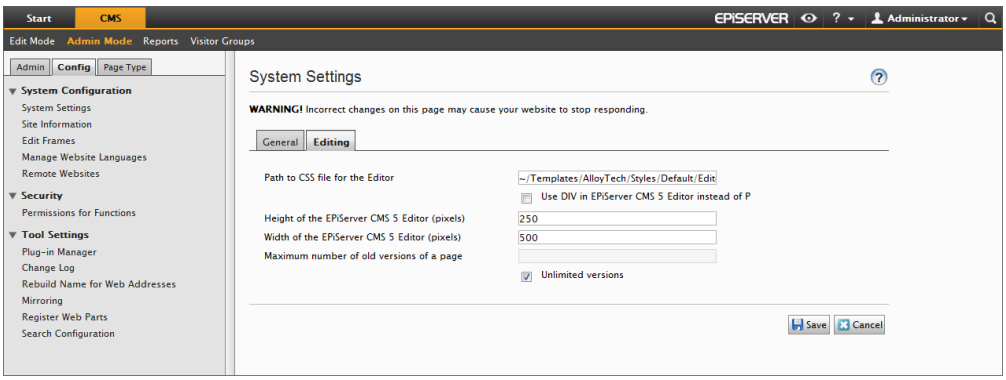
Detect language via browser's language preference

It is possible to show languages based upon the visitor's browser settings. For this to work, the function must be activated. Do this by selecting the **Detect language via browser's language** settings check box. Then the language that each visitor has chosen as their default language comes up in their own browser.

Disable deletion of a page and file versions

Select this check box if you do not want old versions of pages and files to be able to be deleted. When you select this check box, any values in the **Maximum number of old versions of a page** and **Unlimited versions** field in the **Editing** tab will be ignored.

Editing Tab



Path to CSS file for the Editor

The path to the CSS file that controls the appearance of the Editor. This should be the same as the one that controls the appearance of the website, in order to make life easier for the editors. It is also

possible to set other CSSs for different Editors on the website. This is a dynamic property that can be changed in Edit mode.

> This applies to both the TinyMCE Editor and the EPiServer CMS 5 Editor.

Use DIV in Editor instead of P

In HTML, there are different ways of indicating a paragraph break. When you use the tag <P>, there will always be a gap between the rows to indicate the beginning of a new paragraph. If you do not want the extra gap to be visible when pressing ENTER, you can use the tag <DIV> instead. If you select the check box, <DIV> will be used instead of <P>. Note that the change will only apply to those pages created after this change has been made.

> This function only applies if you are using the EPiServer CMS 5 Editor and not the TinyMCE HTML Editor.

Height of the Editor (pixels)

The height of the Editor expressed in pixels. This value applies for all Editor fields, irrespective of page type.

> This function only applies if you are using the EPiServer CMS 5 Editor and not the TinyMCE HTML Editor.

Width of the Editor (pixels)

The width of the Editor expressed in pixels. If the value is zero, the width of the Editor is changed proportionate to the window.

> This function only applies if you are using the EPiServer CMS 5 Editor and not the TinyMCE HTML Editor.

Maximum number of old versions of a page

In this field, you can specify how many old versions of each page the editor can store. If you enter a value here, such as 3, three old versions will be stored in addition to the published version, i.e. four versions in all. When a fifth version is created, the first version of the stored page will disappear.

Unlimited versions

Select this check box if you want an unlimited amount of versions to be displayed in the version list.

Configured Sites

The Configured Sites page lists all the site configurations you have defined. The list may contain one or more site configurations depending on your license model.

- If you are using a single site license model, the list can only contain one site configuration.
- If you are using the Enterprise licence model, the list can contain several site configurations, this also gives you access to the **Add Site** button.

There are one or two buttons available in this view; **Restart Sites**, which is always available and **Add Site**, which is only available if you are using an Enterprise license. Clicking **Restart Sites** will restart all the sites in the list and clicking **Add Site** allows you to add additional site configurations to the list.

Clicking on any of the Site ID links in the list sends you to the Site/License Information page which displays more detailed information about that site's configuration.

Site/License Information

This view shows the configuration for a specific site in more detail. The settings are split between two tabs; the **Site Information** tab and the **License Information** tab.

Site Information Tab

On this tab you can view and modify settings for a specific site configuration.

Site Section

- Site ID - a word or short text that identifies the site.
- Site URL - the URL that is used to construct links when no HTTP context is available, e.g. in subscription mail.
- Start page - the page to which the visitor is sent if only a host name is specified, e.g. when the visitor navigates to http://world.episerver.com.

Host Name Section

The settings in the **Host Name** section enables you to define which host names are associated with this site and specify default languages to be used when a visitor uses a specific host name. Each host name binding in the list consists of two fields:

- Host name - the name of the host.
- Culture - the default language to be used when a visitor accesses the website using the host name.

One of the site configurations must be bound to the * host name. That site will be used as a fallback when an exact match for the host name used by the visitor cannot be found. This setting is less important in the single site scenario as you are only allowed to have one site configuration. However, in the multi-site (enterprise) scenario you must be very careful to make sure that all host bindings that are active in IIS are mirrored in the corresponding site configuration in EPiServer CMS.

License Information Tab

The **License Information** tab lists details about the current license as well as some license management features.

Automatic License Management Section

In the **Automatic License Management** section you can set different options for what action the system should take when it detects that a license is about to expire (less than five days left). The different options are:

- Send an e-mail - An email is sent to the specified e-mail address saying that the license file is about to expire.
- Auto update - The system will make a call to the EPiServer license service (hosted at EPiServer), passing in the current license information such as license number, IP address and MAC address. The license service will use the provided information and the stored information to generate a new license file which will replace the existing file.
- Do nothing - The system will take no action.

Download License Section

In the **Download License** section there are some features that involve communications with the EPiServer license service. These features are:

- Activate license - When an activation key has been received from the EPiServer license system (e.g. by an e-mail) this feature can be used to download the license file. So in this case you type in the received activation key and click **Download License**. EPiServer CMS will then make a call to the license service with the activation key and - given that is a valid activation key - receive the license file and update the system to use the received file.
- Update license - This feature can be used for example when a site is moved to another server (and hence has a new IP/Mac address). In this case EPiServer CMS will send the current license file together with information about current IP and MAC address. The license service will update the information in the license system and generate an updated license.
- Demo license - EPiServer CMS will in this case send the current IP/Mac address to the license service, which will generate a demo license for the site.

Upload License Section

The **Upload License** section lets the user upload a license file from disk and update the site to use it.

Restore License Section

Clicking the **Restore License** button brings you to the Restore Licenses page.

Restore Licenses

Before a license is updated, a backup of the current license file is made automatically. The **Restore License** functionality allows you to browse and restore license file backups.

There are two views available on the **Restore Licenses** page; **All License Backups** and **Backups for Current License Number** which is the default view. You can switch between the views by clicking the buttons at the top of the page. The only difference between the two views is how the license file backup list is filtered.

- The **All License Backups** view lists all backups regardless of which site the license was originally used for.
- The **Backups for Current License Number** view lists only backups that have the same license ID as the license currently in use by the site you selected in the Configured Sites list.

Apart from the different filters used on the license file backup list the functionality is identical for both views. Clicking on any of the file names in the list displays the license information in the **License Information** section and enables the **Restore License** button.

Below the License Information section are two buttons; **Restore License** and **Cancel**. Clicking the **Restore License** button will prompt you with a dialog asking you to confirm the action and, if the action is confirmed, restore the selected backup making it the active license. Clicking the **Cancel** button will bring you back to the Configured Sites list.

Change Log

By default all changes to pages, files and directories are currently logged in the Change Log system. It is possible to filter the information contained in the Change Log making it easier to find relevant information, see below for further information.

Changing the Change Log State

1. In Admin mode, go to the **Config** tab and select **Change Log** under **Tool Settings**.
2. To change the change log, select the desired state from the drop-down list. The available states are Auto, Enabled and Disabled.

- **Enabled** means that the change log tracker will start automatically and items can be written to and read from the change log
- **Disabled** means that the change log tracker will not start, items written to the change log will be ignored but items may still be read from the change log
- **Auto** means that the change log tracker will start as soon as any dependencies have been registered against the change log. If no dependencies exist, then the tracker will not start or will stop if already running. Items can be written to and read from the change log at any time.

Filtering and View the Change Log

1. Filter and view the change log items in the **View** tab. Enter values in the fields according to the table.

Field Name	*Description*
Change date from	The query will be run from the change log from this date.
Change date to	The query shall be run from the change log to this date.
Category	By not selecting an item from the drop-down list Page, File and Directory changes will be read from the Change Log when the query is run. Select **Page** to run a query on pages only. Select **File** to run a query on Files only. Select **Directory** to run a query on pages directories only.
Action	The actions that can be filtered in the Change Log are: Check In, Create, Delete, Delete language, Move, Publish, Save and Delete Children
Changed By	To filter for a specific user enter the EPiServer CMS username.
Maximum number of items per page	Limits the number of items displayed. Click the next and previous arrows to browse through the list of items.
Start with sequence number	Enter a specific sequence number to start listing the items in the change log from, either in ascending or descending order.
Read direction	List change log items in either ascending or descending order.

2. After entering the values in the fields click **Read** to run the query. A list of the matching change log items will then be displayed.

It is possible to remove all items from the change log that are more than one month old and without dependencies with the change log auto truncate scheduled job.

Website developers can easily customize and extend the change log functionality. Consult your website developer for further assistance.

Preset Sizes for Images

The Image Editor is used when working with images in EPiServer CMS. Images can be cropped, resized and transformed with the Image Editor. When resizing images it is possible to configure preset sizes so that editors can easily select the correct size for their image on a web page. The preset values are configured in the episerver.config file.

You can choose to change the preset sizes or add more sizes in the episerver.config file. The easiest way to add more sizes is to copy an already existing row, for example <present width="64" height="64" /> and place it before the end tag </sizePresets>. Change the sizes for width and height and save the episerver.config file.

The following excerpt from episerver.config is the settings from a standard installation of EPiServer CMS. These settings result in the

```
<imageEditor windowWidth="900" windowHeight="600" enabled="true">
  <sizePresets>
    <preset width="320" height="240" />
    <preset width="640" height="480" />
  </sizePresets>
</imageEditor>
```

Rebuild Name for Web Addresses

The **Rebuild Name for Web Addresses** function is used for getting and changing addresses in the address field. When a visitor looks at a certain page on a website based on EPiServer CMS, a path to the page is shown in the address field, as in the example below. The address reflects the page's place in the structure.

The names in the address field are created automatically from the name an editor has specified for the page. If an editor changes the name of the page, the name in the address field will not change. This can however be changed manually by changing the field **Page name in web address** on the **Settings** tab.

Some pages have no value in the field for names in web address. This could, for example, be pages imported from other EPiServer solutions. With the **Rebuild Name for Web Addresses** function you can create all the web addresses for the website at the same time. It is also possible to change all the existing addresses and overwrite them with new ones.

> Remember that this function can affect links to the website. All internal links are managed and updated automatically, so you do not have to worry about these. There can however be other websites that have linked to a certain page and this can mean that such a link will be broken. It can also affect the favorites that visitors have in their web browsers. So you should therefore think twice before using this function.

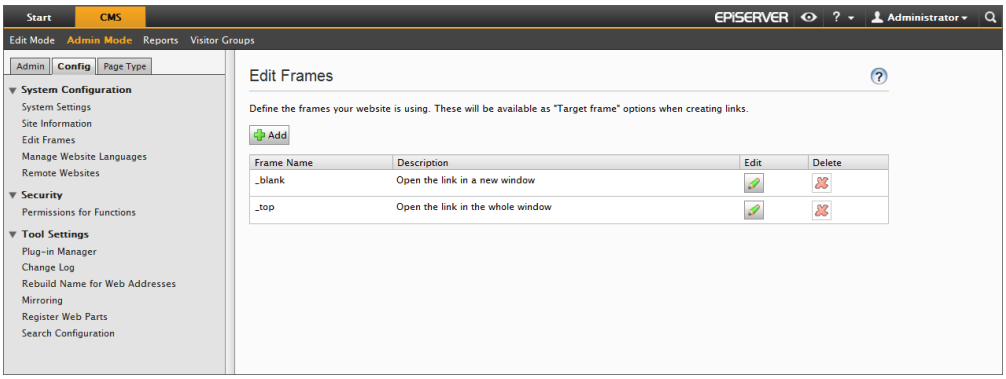
Rebuild names for web addresses as follows:

1. Select **Rebuild Name for Web Addresses** on the **Config** tab.

2. If you want to overwrite existing addresses, select **Rebuild all page names in web addresses (overwriting existing links)**.
3. Click **Rebuild links**.

Edit Frames

If you have chosen to use frames on your website, there may be times when an editor must be able to open a link in a particular area of the frame. The names of any frames used by the system are defined here. These designations are accessible in Edit mode, whenever the editor must designate a target frame for his/her links.



Remote Websites

The remote websites function enables to you to retrieve information between the various EPiServer websites. This is used for load balanced websites and content mirroring, amongst other things. To be able to take information from one website to another, is a function that is often requested by customers

with several websites on EPiServer CMS on several different machines, such as a public website and an intranet. Editors need only create information in one place, but it can be displayed in all.

With the **Remote Websites** function, you create the connections to the websites that you want to be able to send information to and receive information from. In Admin mode it is possible to set the function completely. A number of changes in web.config are also required. Which changes are required depends on which functions you are going to use it for.

Error Handling in EPiServer CMS

EPiServer CMS has functions for showing error messages to visitors and editors when there is a problem with the website. All sorts of errors are possible. Some are as simple as a broken link or an error in the program code. If a technical error occurs in EPiServer CMS, an error message will normally be displayed from the .NET environment. The message can have an alarming effect and is not helpful to a visitor or editor. However, this is useful to a developer in solving the problem. The function **Error handling in EPiServer CMS** lets you select which groups should receive friendly error messages and which should see error messages with all the code.

Activate error handling as follows:

1. Select **System Settings** on the **Config** tab.
2. In the **Error handling in EPiServer** field, select one of the three options.
   If error handling is **Switched Off**, everyone receives the .NET environment's error messages. If it is **Activated for visitors from outside**, everyone who is not logged on to the website receives friendly error messages. Others receive error messages with accompanying help from the .NET environment. If error handling is **activated for all**, everyone receives friendly error messages, regardless of role.
3. Specify an e-mail address in the **E-mail address to send error reports to** field.
   If technical problems occur, the visitor or editor will receive a form to fill in.
4. Click **Save**.

Access to Detailed Error Message for Troubleshooting

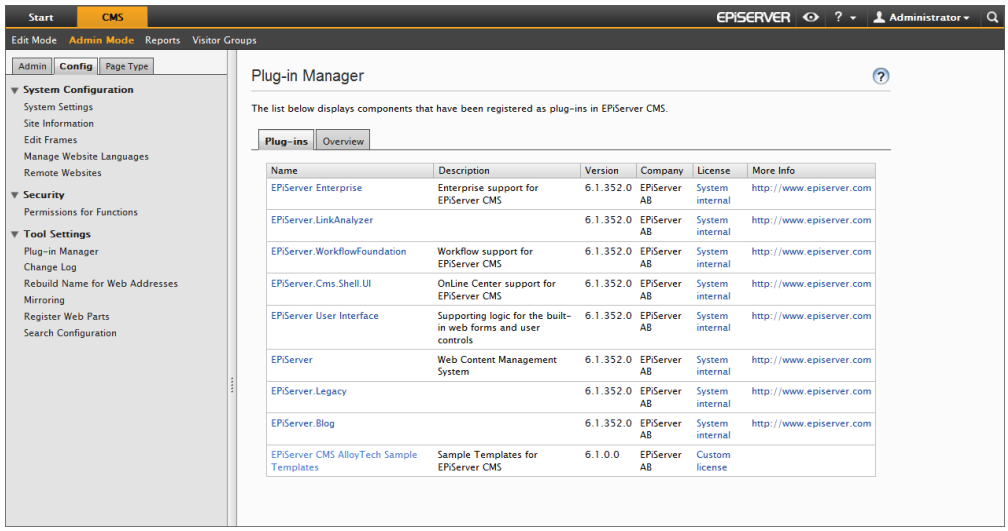
If you want most people to receive friendly error messages from EPiServer CMS, activate them for everyone. It can however be good for developers and administrators to see detailed error messages. By changing the permissions for this function, you can specify who should receive a detailed error message in the event of a technical error.

Give certain groups/individuals access to detailed error messages as follows:

1. Select **Permissions for functions**, on the **Config** tab.
2. Click **Edit** after **Detailed error messages for troubleshooting**.
3. The groups/users that have access to detailed error messages are displayed.
4. Add or delete groups and users, so that only those who are to receive detailed error messages are displayed.
5. Click **Save**, at the bottom of the page.

Plug-in Management

Many of the functions in EPiServer CMS are created as plug-ins. Which can be managed from the **Plug-in manager**. Selected parts can be activated and deactivated. If your organization has invested in any additional plug-ins, these can also be found in the Plug-in Manager. When you have chosen a certain plug-in, you can choose which parts of the plug-in are to be accessible in the **Overview** tab.



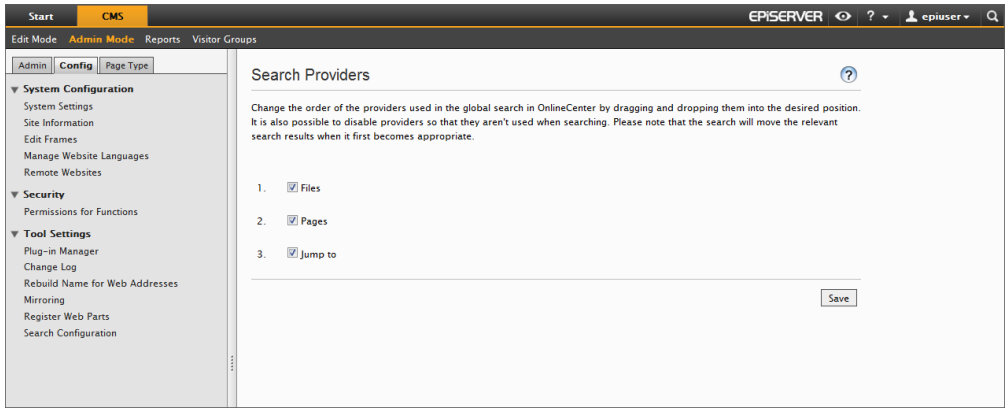
Search Configuration

It is possible to configure different **search providers** for the modules in your website implementation. A search provider can be for instance pages, files, categories, forums or page types. The standard installation of EPiServer CMS comes with two search providers, **Pages** and **Files**. To configure additional search providers for your website, check with your system administrator and refer to the technical documentation for OnlineCenter on world.episerver.com.

You can decide which search providers you want to enable, and the order in which they will appear in the search hit list. To access these settings, switch to the EPiServer CMS Admin mode and select **Search Configuration** under the **Config** tab. The following options are available for a standard installation of EPiServer CMS:

- **Jump to** -selecting this option will make it possible to jump from the search hit list directly to **menu alternatives** matching your search criteria.
- **Pages** - selecting this option will allow for search in pages on the website.
- **Files** - selecting this option will allow for search in files on the website.

You can drag and drop the search provider options to change the order between them. This will control the order in which the results will be displayed in the hit list. Clearing a checkbox will disable this search provider option. Select **Save** to save your changes.



Mirroring Administration

Mirroring is a function that can be used to copy duplicate content from one website to another. EPiServer CMS can be used to mirror selected parts or all of a website and can either run automatically or manually. You can also mirror from one EPiServer CMS to another EPiServer CMS. This can be very useful if you wish to create large sections in a test environment and then publish all the information at once in the public section. An enterprise license is required in order for mirroring in EPiServer CMS.

A mirror from one website to another is built up through channels on the website that transmit information. It is possible to create several channels on the source website. A mirroring job is an execution of channel – transmitting information from one website to another.

The mirroring jobs are carried out at certain time intervals; state how often and when the mirroring shall occur under **Scheduled jobs** and **Mirroring Service** on the **Admin** tab.

In order for mirroring to work in an EPiServer CMS website, a mirroring application needs to be installed and running. The mirroring application handles the data transfer between the websites and is run separately to the EPiServer CMS source site and target sites. Source and target websites can be configured to use separate mirroring applications, it is also possible to install and configure a single mirroring application. The mirroring application is installed with the EPiServer Deployment Center. For further information regarding configuring mirroring, see the mirroring tech notes on EPiServer World.

The mirroring application can even be hosted on a totally separate machine to the source and target sites. This allows for much better performance, scalability and greatly reduced impact on the EPiServer CMS site. The default configuration creates an application hosted in IIS, however this can be changed by partners and customers to be hosted and any type in Windows application.

Mirroring Settings

To set up mirroring from one website to another channels need to be created in Admin mode. Channels define the mirroring jobs from one EPiServer CMS to another, or even another external recipient.

Create a Channel

Go to the Admin mode on the source site. In the configuration tab, Under Tool Settings, click **Mirroring** to create and edit channels for Mirroring.

1. Click the **Config** tab and then **Mirroring**.
2. Click **Create**. Mirror Settings will be displayed.
3. Enter values for the channel. See the table below for information regarding the respective fields.
4. After entering the values in the fields, click **Save**.

Field Name	*Description*
Name	Enter a name for the channel in the **Name** field.
Parameters	This is an optional field that can be used by providers.
Use default URI	Select this check box to use the URI defined in the configuration file (web.config) for the mirroring application. When you select this check box, the **URI** field will be disabled.

URI	The URI to the destination mirroring application's target service, for example http://localhost/R3Mirroring/Mirroring/MirroringTransferServer.svc. The service for the default provider is “MirroringTransferServer.svc”.
Start page	The root page on the source site to be mirrored.
Root page on destination	The page number on the target site where the pages will be mirrored.
Include the start page	This field determines if the start page will be mirrored or if only its children will be mirrored.
Import as anonymous user	Run as anonymous user, determines if a special user will do the export and import of pages and files when the mirroring job is run.
Import content as user	Enter the username of a user that will perform the export and import of pages and files when the mirroring job is run.
Enable reporting	Select this check box to enable reporting to an e-mail. When you select this, the **E-mail address** field will become editable.
E-mail address	Enter the e-mail address where a report will be sent.
Continue on error	Select this check box to enable a mirroring Job to continue mirroring subsequent items if a specific item cannot be mirrored. If this box is not selected, a mirroring Job will terminate as soon as an error occurs.
Enabled	This field determines if the channel is active or not. If it is not enabled, nothing will be mirrored for the channel.
Enable validation	Select this check box to enable validation before items are mirrored. The mirroring Job will validate that all page types for pages being mirrored are present in the target CMS site and that the page type has the same number and type of CMS page properties as the source CMS site.

Edit a Channel

To edit a channel, click the name of the channel, update the fields as appropriate and click **Save**.

Delete a Channel

To delete a channel, click the name of the channel and then click **Delete**.

- **Reset Channel**. Select **Reset State** and the mirroring application will re-mirror everything from the root page of the mirroring channel to the source site next time it is run. Click the name of the mirroring channel and then click **Reset**.
- **Check System**. To validate the created channel, click**Check System**. The result of this will be displayed under the **Messages** tab.
- **Messages**. The result of a check system is listed in the **Messages** tab. Click the arrow icon to view the result and other information regarding previous mirroring jobs that have been run.
- **Monitoring**. The status of a mirroring job is displayed in the **Monitoring** tab.
- **Target site configuration**. To use a site as a mirroring target, some configuration is required. See the mirroring tech notes for further information regarding mirroring configuration.

- **Mirror Pages to a Target Site**. To create a target page, go to Edit mode on the target site and create a page that will act as the root page for the mirrored pages. It is also possible to use an existing page as the mirroring root page. In order to publish pages, the mirroring application needs to be installed and configuration needs to be implemented. Read the mirroring tech notes for information regarding this.
- **Mirroring in other Formats**. Mirroring in formats other than EPiServer CMS Import/Export format can be created by partners and customers themselves and plugged-in using the provider model. See the mirroring tech note for further information.
- **Using EPiServer CMS 5 Mirroring**. The mirroring plug-in from EPiServer CMS 5 is disabled by default in later versions of EPiServer CMS. To enable the plug-in, go to the **Plug-In Manager** in the **Config** tab on the source website, click **EPiServer User Interface** and then **Overview**. You will now see that EPiServer CMS 5 Mirroring is unselected. Select this check box and save. Reload the page and go to the **Config** tab again and you will see **EPiServer CMS 5 Mirroring** in the list under **Tool Settings**.