

EPiSERVER CMS 5

Release 2



Administrator's Manual

Copyright

This user guide is protected by the Copyright Act. Changes to the contents, or partial copying of the contents, may not be made without permission from the copyright holder.

The manual may be freely distributed in its entirety, either digitally or in printed format, to all EPiServer CMS users.

EPiSERVER

Administrator's Manual for EPiServer CMS 5 Release 2 Rev B

Created March 24, 2006

Last saved January 29, 2009

© EPiServer AB

Table of Contents

INTRODUCTION TO ADMIN MODE	7
LOGGING IN	7
THE WINDOW	7
WORKING WITH AUTHORIZATION	9
THREE DIFFERENT PROVIDERS	9
START FROM THE STRUCTURE	9
USER GROUPS IN A STANDARD INSTALLATION	10
ACCESS RIGHTS TO DIFFERENT VIEWS	10
All Editors Should Be Members of WebEditors	11
Specify Who Has Access Rights to the Various Modes in web.config	11
CREATE AND SYNCHRONIZE GROUPS	12
Groups in EPiServer CMS	12
Deleting a Group	13
Groups Created in Windows	13
CREATE USERS IN EPISERVER CMS	14
Display Options for Users	14
SETTING ACCESS RIGHTS IN THE TREE STRUCTURE	15
Changing Access Rights for One or More Groups in the Structure	16
Deleting Access Rights for One or More Groups in the Structure	17
Overwrite All Previous Access Rights	17
Setting Access Rights for Different Editors in the Tree Structure	18
SEARCH USER/GROUP	20
Editing Users	20
Deleting Users	21
AUTHORIZATION TO PUBLISH FROM MICROSOFT OFFICE	21
WORKFLOWS IN EPISERVER CMS 5	23
ADMINISTRATION OF WORKFLOWS	24
CREATE NEW WORKFLOWS FROM ADMIN MODE	25
General Tab	26
Security Tab	27
Automatic Start Tab	27
Running Instances Tab	29
START PARAMETERS TAB	29
Sequential Approval Workflow	30
Parallel Approval of Pages	30
Translation of Pages with EPiServer Globalization Support	31
Request for Feedback	32
EDITING DEFINITIONS	33
DELETE DEFINITION	33
FILE MANAGEMENT IN EPISERVER CMS	34
ONE OR MORE STARTING POINTS IN FILE MANAGER	34
Configure Starting Points	34
GLOBAL FILES VS. PAGE FILES	35
Page Files	35
Global Files	36
SETTING AUTHORIZATION FOR FOLDERS	36
DELETING UNUSED FILES	37
AUTHORIZATION FOR ADVANCED UPLOADING OF FILES	37
CUSTOMIZE EDIT MODE	39
PAGE TEMPLATES VS. PAGE TYPES	39
LIST OF PAGE TYPES IN EDIT MODE	39
Page Type Sorting in the List	39

Available Page Types Under a Certain Page Type	40
Page Type Available in Edit Mode	40
DEFAULT VALUE OF FIELD	41
Default Values of Default Field	42
EDITOR BUTTONS	42
Certain Editors Require All Buttons in the Editor	43
Properties for the Editor	43
COMPULSORY FIELDS	44
HIDE FIELDS IN EDIT MODE	44
FIELD SEARCHABILITY IN EPISERVER'S SEARCH ENGINE	44
FIELD ORDER IN THE PAGE TYPE AND TAB LOCATION	44
EDIT TABS	45
WORKING WITH PAGE TYPES	46
Create New Page Type	46
Copy Page Types	47
Convert Pages	47
CREATE PROPERTY	48
Custom Property Types	49
CATEGORIES IN EDIT MODE	50
LANGUAGES IN EDIT AND ADMIN MODE	51
Modify Texts in Edit Mode	51
SET PRESET SIZES IN IMAGE EDITOR	52
HELP FILES	53
GLOBALIZATION IN EPISERVER CMS	54
ENABLE GLOBALIZATION	54
THE WEB SITE'S DEFAULT LANGUAGE	54
MANAGE WEB SITE LANGUAGES	55
Add New Language for Multilingual Web Sites	55
Edit Language for Multilingual Web Sites	56
Delete Language for Multilingual Web Sites	56
Which Language is Shown to Visitors?	56
FIELDS THAT VARY PER LANGUAGE	57
LANGUAGE SETTINGS CAN VARY FOR DIFFERENT PARTS OF THE TREE STRUCTURE	58
SCHEDULED JOBS	59
ARCHIVE FUNCTION	59
MIRRORING SERVICE	59
AUTOMATIC EMPTYING OF THE RECYCLE BIN	60
SUBSCRIPTIONS	60
PUBLISHING DELAYED PAGE VERSIONS	61
FILES FOR EPISERVER CMS	62
DIRECTORY CONTENTS	62
Customizable Directories	62
EPISERVER CMS PAGE TEMPLATES	62
WORKING ON THE DESIGN OF THE WEB SITE	63
Editor Format Template List	63
The Color Selector in EPiServer's Dialog Boxes	64
EXPORTING AND IMPORTING DATA	65
EXPORT DATA	65
IMPORT DATA	66
INTEGRATION WITH EXTERNAL SYSTEMS	67
CONTENT CHANNELS	68
Mapping Page Type Properties	68

CONFIGURATION OF EPISERVER CMS	70
SYSTEM SETTINGS	70
General	70
Editing	71
LICENSING INFORMATION	72
REBUILD NAME FOR WEB ADDRESSES	72
EDIT FRAMES	73
REMOTE WEB SITES	74
ERROR HANDLING IN EPISERVER CMS	74
Access to Detailed Error Messages for Troubleshooting	74
PLUG-IN MANAGEMENT	75
MIRRORING ADMINISTRATION	75
INSTALLING EPISERVER CMS	78
UPGRADE EPISERVER CMS	78
INDEX	79

Introduction to Admin Mode

EPiServer CMS is an information platform system that is not especially demanding on either the editors or the hardware and software of their computers. Editors need basic word processing skills, and computers require a Web browser. The Web site can also be administered relatively easily. The majority of the administration is carried out using simple forms in the Web browser.

Depending on your role and how your organization is structured, you may need more technical knowledge. Knowledge of how the Internet functions and how Web pages are created and managed in various systems is a major advantage. Knowledge of Windows Server and MS Internet Information Server is also necessary for the work to flow smoothly. Knowledge of HTML, CSS and ASP.NET are also a great help if you wish to go further into management in EPiServer CMS.

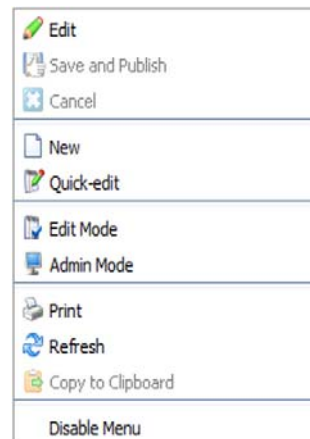
This manual primarily describes the functions available in the Admin mode in EPiServer CMS. Some of the functions in Admin mode require the installation of certain software or special settings of different types and changes to files on the server where EPiServer CMS is installed. In the text, we regularly refer to different documents that describe the different parts in greater detail. You can find all these documents on EPiServer World, which is a meeting place for sharing information with others. You can find EPiServer World at <http://world.episerver.com>.

Logging In

You can access EPiServer's Admin mode through your Web browser. The Web browser must be Internet Explorer 6, Internet Explorer 7 or FireFox 3. Note that *Copy to Clipboard* is only available in Internet Explorer 6 and 7.

If you are automatically logged in on the Web site 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 Web site. Check with your supplier or whoever installed and configured EPiServer CMS for information about this.



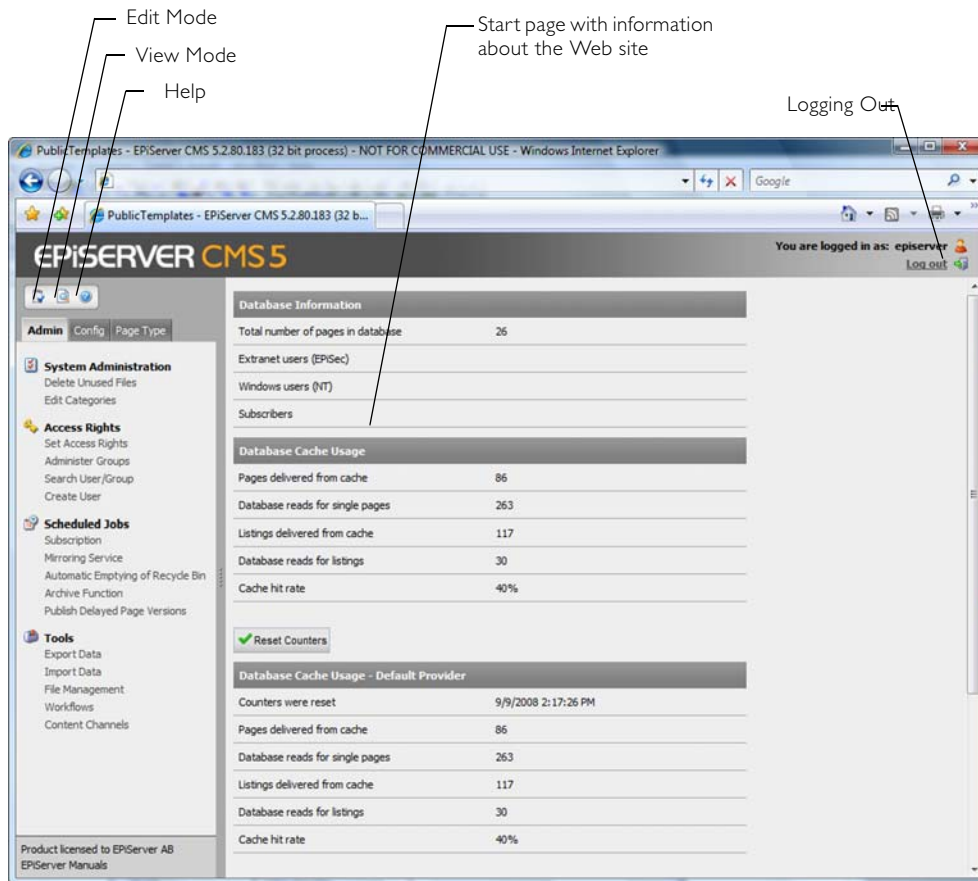
Administering the visual appearance, changing language in Edit and Admin modes and certain other settings, are done directly via changes to files on the Web server. For more information, see *Files for EPiServer CMS* on page 62.

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 Web site. You can see how many pages you have in the Web site, how many subscribers etc.

Three tabs with different functions, organized according to function use, are shown to the left. The different tabs are used in different ways depending on the role you have. The *Admin* tab has functions that often concern the daily work of administering an EPiServer CMS Web site. The *Config* tab contains functions that are often set up at installation; these settings are rarely changed. The *Page Type* tab contains all the page types that the editors create pages in. The page types function is widely used by developers when developing new templates.

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



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 Web site.

Three Different Providers

In EPIserver CMS, it is possible to work with different sources for storing users and groups. The technology used 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 Web site. 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 Web site, Windows applies. This means that you must log in the first time with a Windows user name

Start From the Structure

Access rights in EPIserver CMS have two dimensions. Partially that an individual/group has access rights to different modes in EPIserver CMS and partially that one can have access rights to do different things within the structure of the Web site. 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.

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 for if one wishes 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.

User Groups in a Standard Installation

EPiServer CMS uses two main groups by default. These groups are called WebEditors and WebAdmins. There are two groups in Windows that are also relevant to EPiServer. These are Administrators and Everyone. These four groups have as default different roles and tasks on the Web site. The aim of the groups is primarily to give access to the different modes in EPiServer. The aim is not to give access rights in the structure; that work is carried out for each individual Web site.

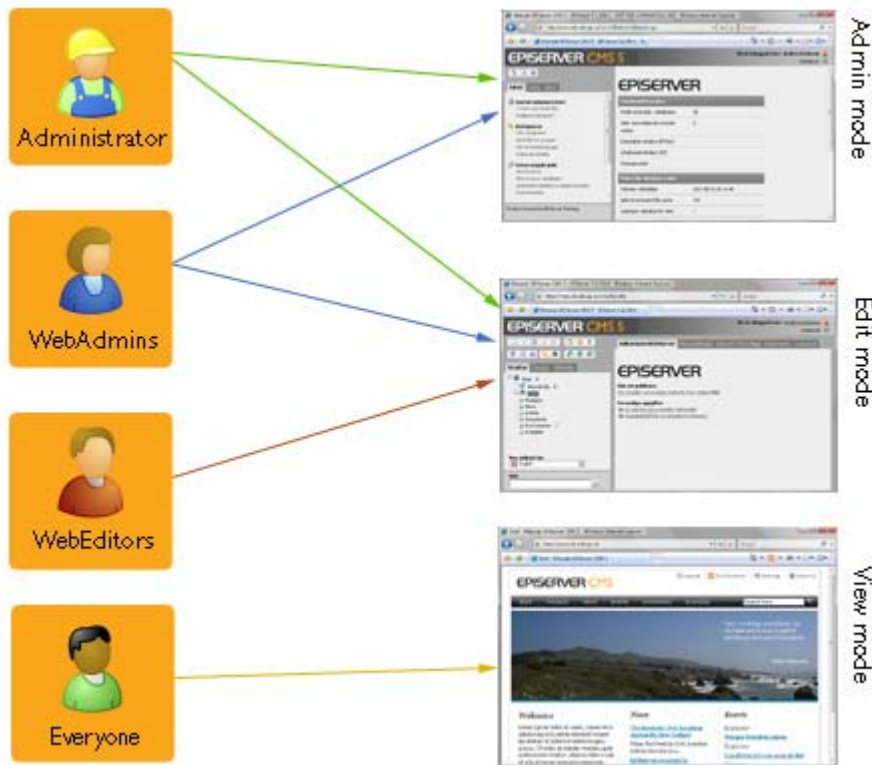
The WebEditors and WebAdmins groups have not been created in a standard installation. When you have selected providers for your Web site, you must begin by creating these two groups there.

Group	Function
Everyone	Anonymous visitors to the Web site. All visitors to a public Web site are essentially anonymous. This means that the system cannot identify the visitor. With an intranet, 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	Web site 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. Web site editors do not have access to Admin mode by default.
WebAdmins	Web site 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. Web site administrators also have access to Edit mode by default.
Administrators	Web site super administrators. To be able to change system settings in EPiServer CMS, you must be a member of this group. Administrators have access rights to both Edit and Admin modes by default.

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 Web site. 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.



All Editors Should Be Members of WebEditors

The file `web.config` contains many settings for the Web site. Changing the wrong thing in this file can have consequences for the entire Web site 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. All editor groups must have access to Edit mode, which is specified in `web.config`.

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.

When you install a Web site on EPiServer CMS the WebEditors and WebAdmins groups have not been created. You will have to do this first in order to be able to work with these groups. Depending on which providers you have selected for storing groups and users, 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.

Specify Who Has Access Rights to the Various Modes in `web.config`

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 Web site.

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. 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="admin">
  <system.web>
    <globalization requestEncoding="utf-8" responseEncoding="utf-8" />
    <authorization>
      <allow roles="WebAdmins, Administrators" />
      <deny users="*" />
    </authorization>
  </system.web>
</location>
<location path="edit">
  <system.web>
    <globalization requestEncoding="utf-8" responseEncoding="utf-8" />
    <authorization>
      <allow roles="WebAdmins, WebEditors, Administrators" />
      <deny users="*" />
    </authorization>
  </system.web>
</location>
```

Create and Synchronize Groups

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. Read more about *Three Different Providers* on page 9

When you select *Administer Groups* in Admin mode, all groups will be shown irrespective of the manager in which they are stored. The manager that manages the group in question is shown to the right of the group name.

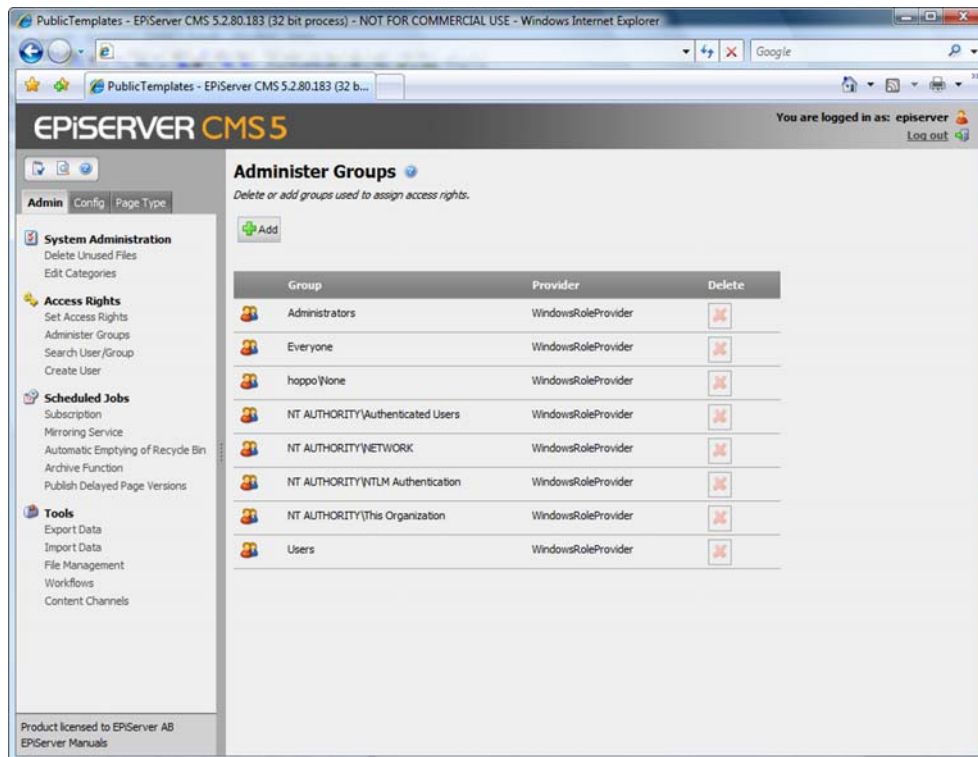
Groups in EPiServer CMS

It is common to use EPiServer CMS's own management of groups and users if part of your Web site is an extranet. You can also make groups and give them specific read access rights to parts of the Web site. It is also common practice to use EPiServer's management if the Web site 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.

Create groups in EPiServer as follows:

1. Select *Administer Groups* on the *Admin* tab.
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.



Deleting a Group

If you have created groups that you no longer require, you can delete them. There are several ways to delete groups. You can delete a group from *Administer Groups* by clicking the *Delete* button displayed to the right of the group.

You can also delete a group in the same way as deleting a user.

1. Select *Search User/Group*.
2. Select the group you want to delete by clicking the group name.
3. Click *Delete* to the right of the group name.
4. Confirm that you want to delete the group in the pop up dialog box.

It is not possible to change the name of an existing group. Delete the existing group and add a new group instead.

Groups Created 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 Web site 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,

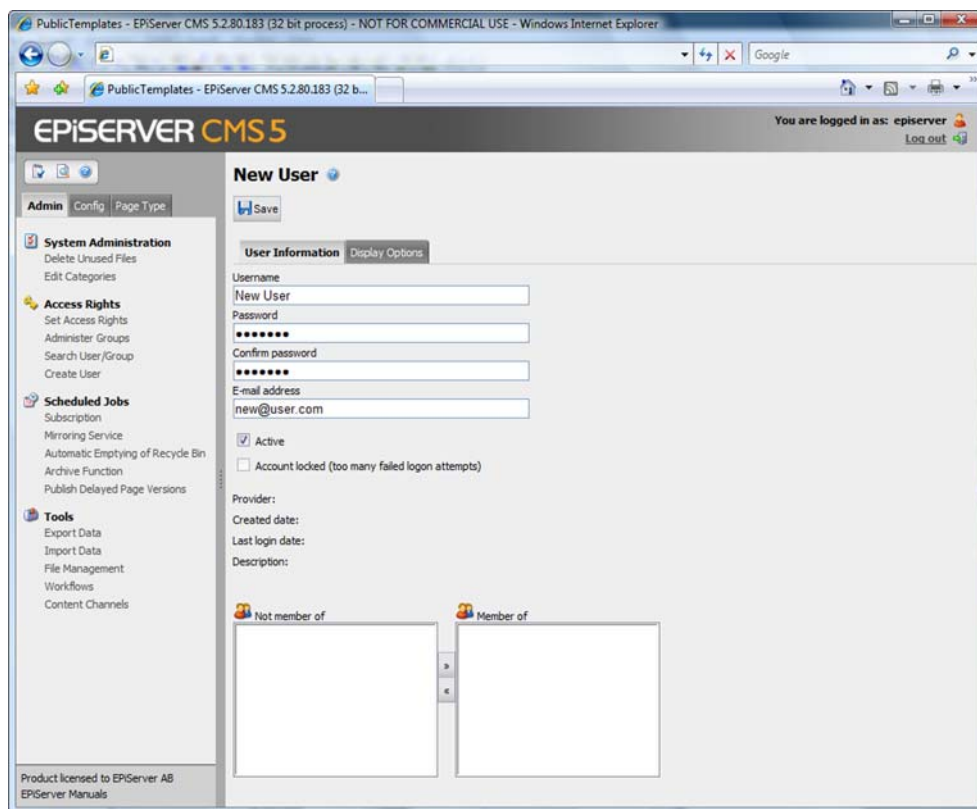
which you want to use in EPiServer CMS you must start by logging in with one of the users. Then log in as an administrator and you will find the group in EPiServer CMS Admin mode.

Create 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.

Create users as follows:

1. Select *Create User* on the *Admin* tab.
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*, at the top of the page.

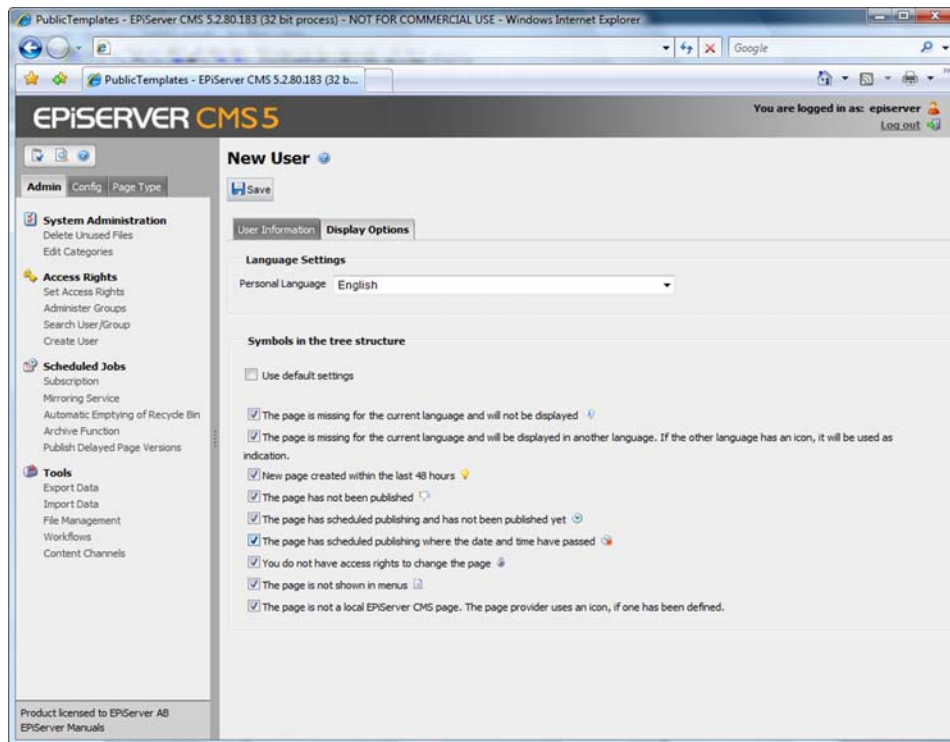


Display Options for Users

The *Display Options* tab is for each editor to be able to decide what they want to show in the structure 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.



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. 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.

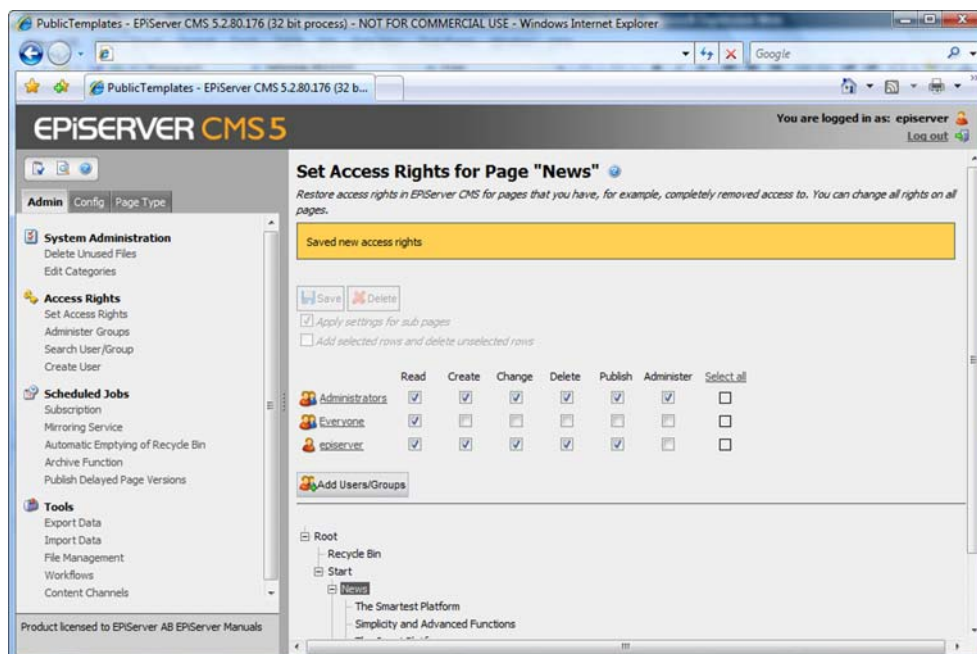
Delete means the group/user has the access rights to delete the current page.

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 Access Rights for One or More Groups in the Structure

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

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 edit the access rights. When the page is highlighted, the name of the page is formatted with a dark gray background.
3. Those groups and users that have access rights to the highlighted page are shown with the most recently saved access rights above the structure. Clear/select the access right(s) you want to add/delete for the group. 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 highlighted 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.

Deleting Access Rights for One or More 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 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. Select the group(s) 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 highlighted page.

Remember that you only affect the groups that are highlighted. The groups that are not highlighted when you click *Delete* 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 delete.

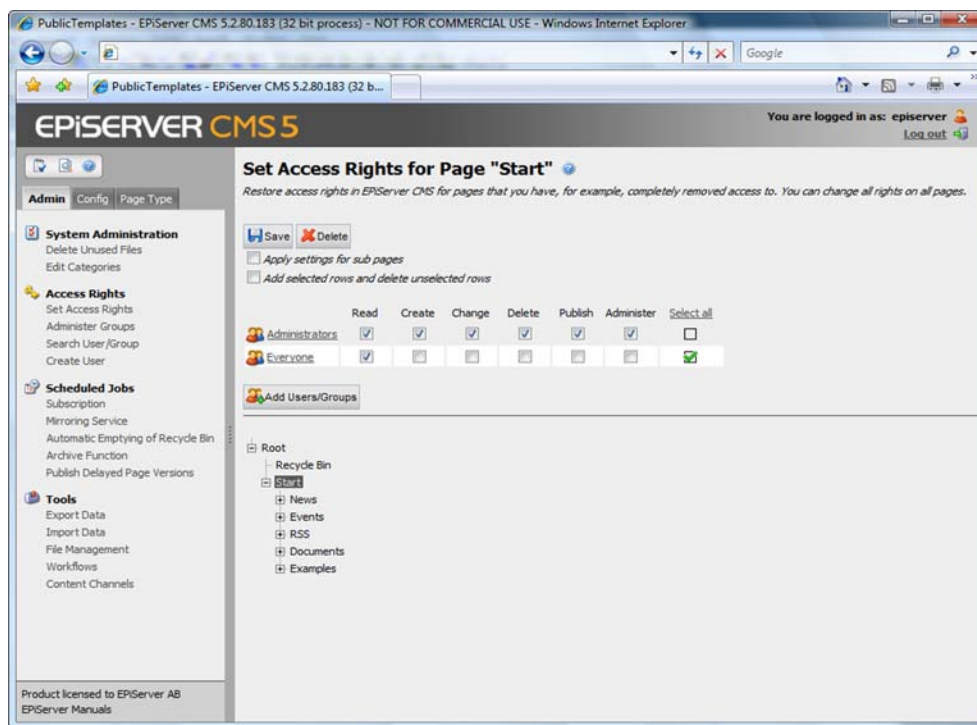
Overwrite All Previous Access Rights

Sometimes there are occasions when it 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 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. 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 highlighted page.
6. Click *Save*.

The groups that are highlighted, 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 highlighted page and all its sub pages. Everyone will be removed from the highlighted page and all its subpages.



Setting Access Rights for Different Editors in the Tree Structure

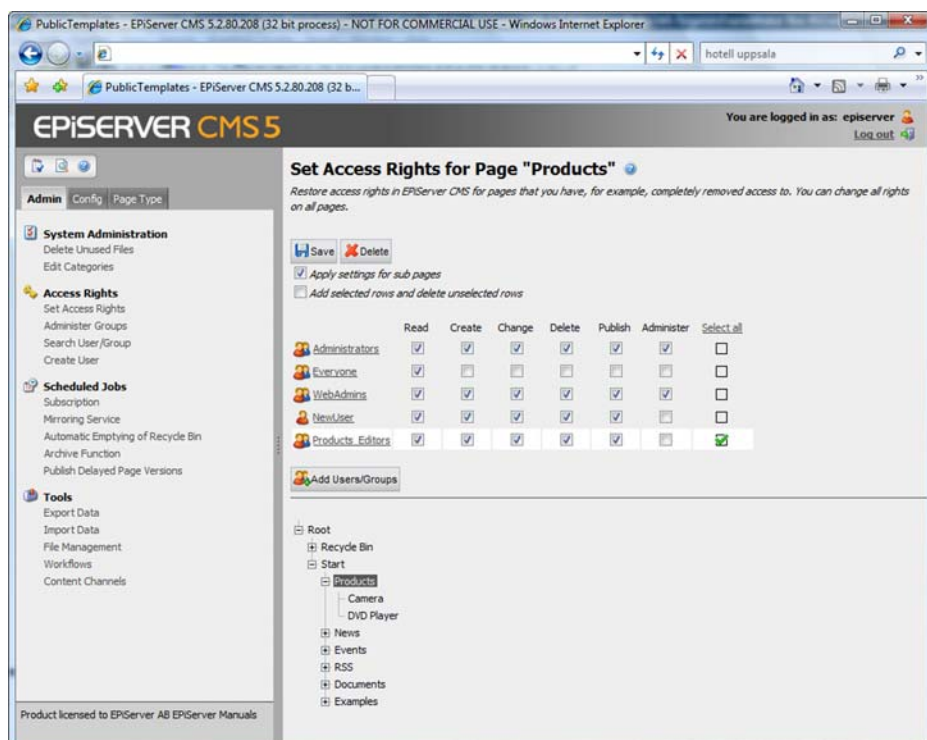
Within a large organization, it is customary to have control over editors, so that they can only create, edit, delete and publish in specific parts of the tree structure. On an extranet, it is also quite usual to grant different customers access to read different pages. To do this, you must start by 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 Web site has three different main branches: Products, News and Support. Each section has two or more editors who will help to update information on the Web site. 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. Select *Set Access Rights* on the *Admin* tab.
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.

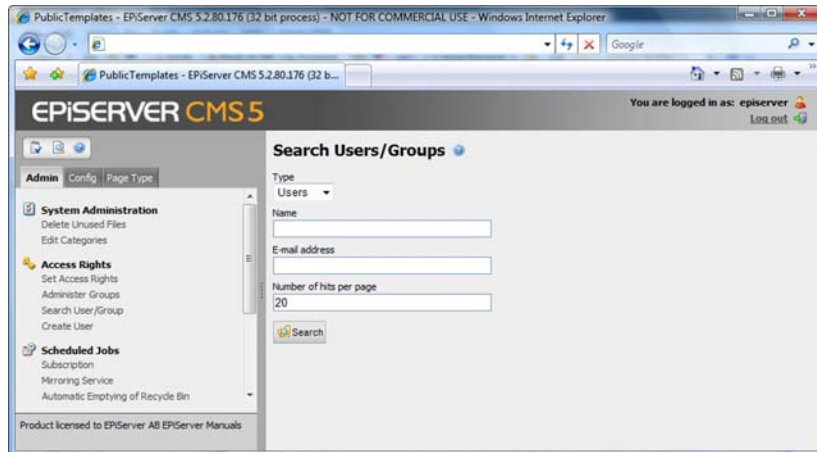


Similarly, it is possible to create different groups in order to grant different visitors access to different pages. On a public Web site, for example, ensure that "Everyone" is not granted access rights to read those pages you do not want to be seen by the general public. By allowing extra-net users 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

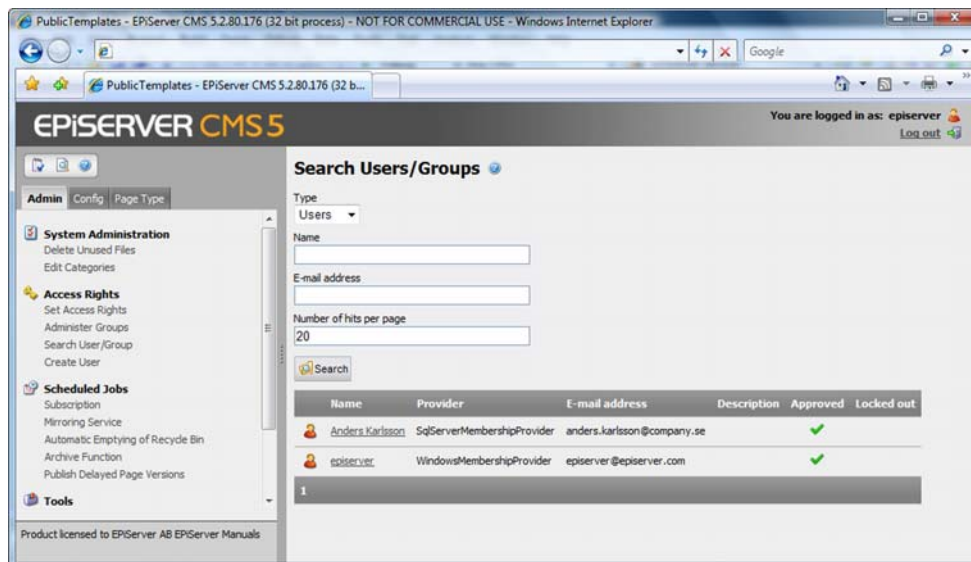
not appear in their menus. The visitor is simply not aware that there is more information available.

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.



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

If there are users in EPiServer CMS that no longer need access to your Web site via login, it is appropriate to delete them.

Delete a user as follows:

1. Select *Search User/Group*.
2. Select the user you want to delete by clicking the username.
3. Click *Delete* at the top of the page.
4. Confirm that you want to delete the user in the pop-up dialog box.

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 Web site they are to work on. Read more about this in the *Editor's Manual*.

In order for an editor to be able to activate a Web site 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>-->
```

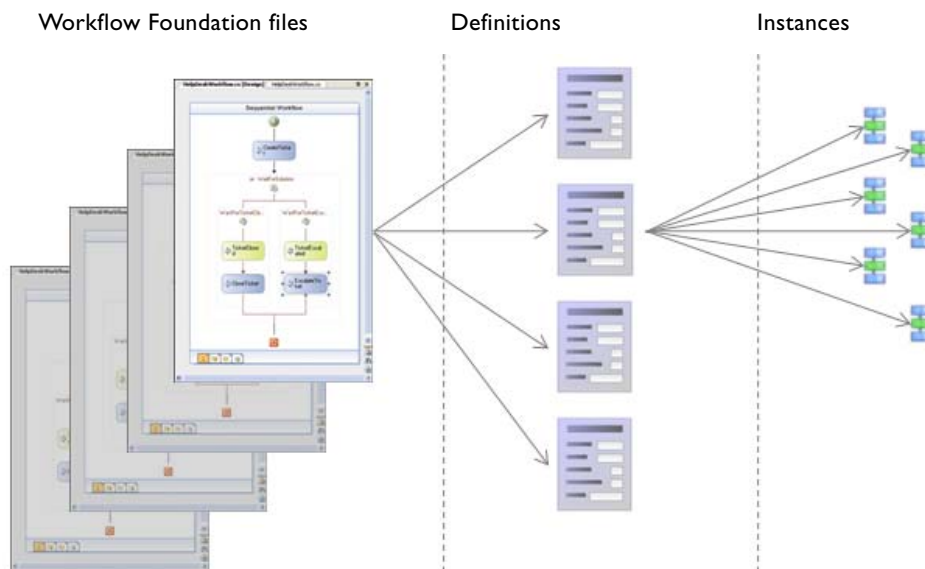
UI is an customized path to Edit mode. This might be different in your installation.

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 5

The workflows in EPiServer CMS 5 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 5, 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">.

In this manual, we describe how they can be configured and used. For developing new workflows, please see the documentation available in EPiServer World and EPiServer CMS SDK.

1. 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.

2. 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.

3. Translation product pages

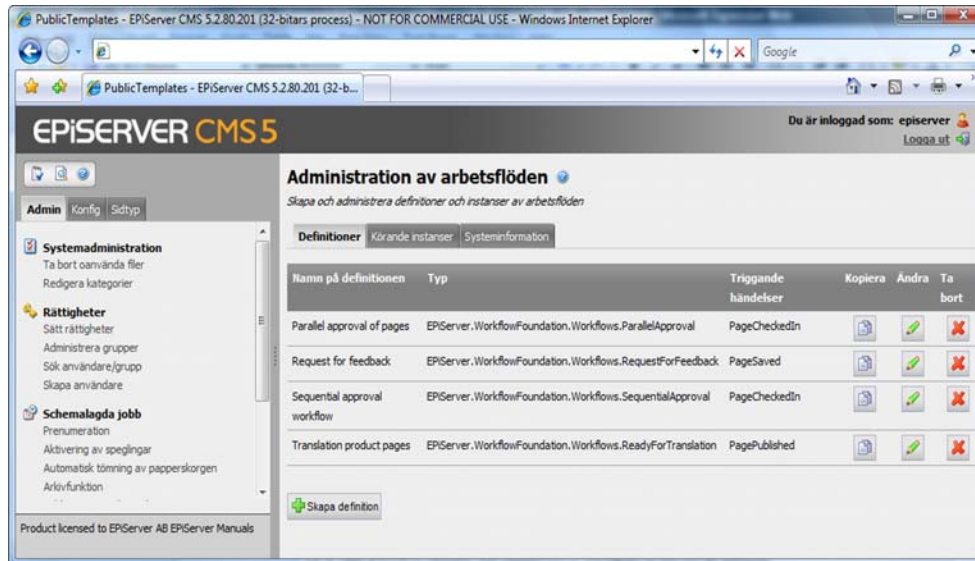
The workflow for managing the translation of pages is based on the activation of globalization in EPiServer CMS. With globalization, you can write the same page in several different languages. Usually one person will create the first language version. The workflow sends a task to another editor saying 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 Web site that is to be translated to several languages, tasks are sent to all the translators at once.

4. 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.

System Information Every definition is triggered by different events in EPiServer CMS. On the *System information* tab you can retrieve data on how many running instances are waiting for a certain event, for example how many instances are waiting right now for a page to be created. This information is primarily an aid for developers/administrators for verifying/troubleshooting in the workflows they are running to see if they are behaving as expected.

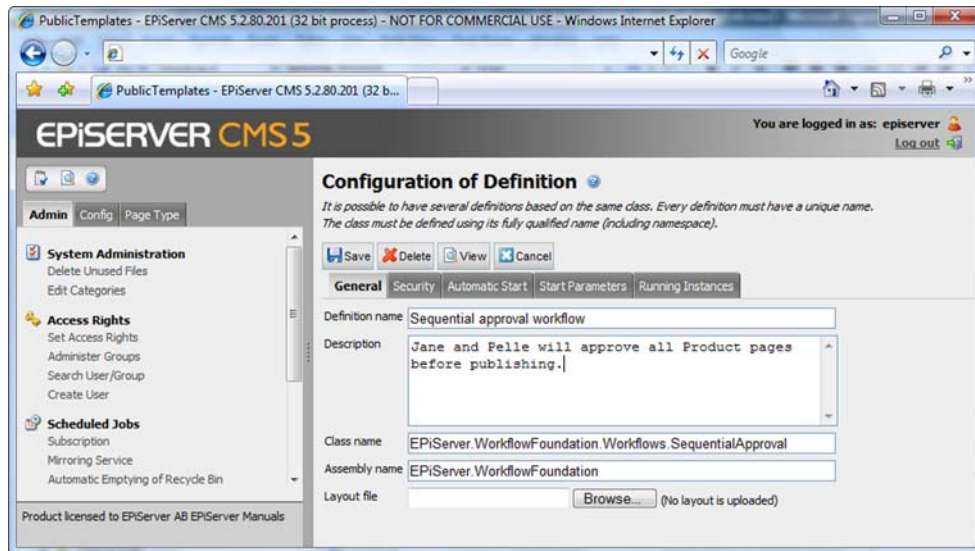
Create New Workflows from Admin Mode

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 Web site 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

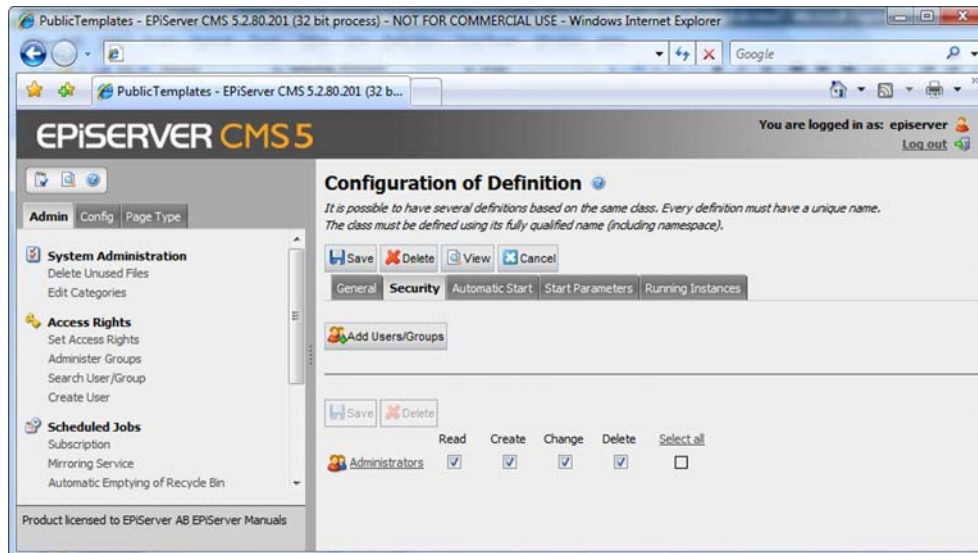
Specify the assembly name for this workflow. The developer of the workflow must provide this information. In the four workflows available in a standard installation of EPiServer CMS, the *Assembly name* is always EPiServer.WorkflowFoundation.

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.

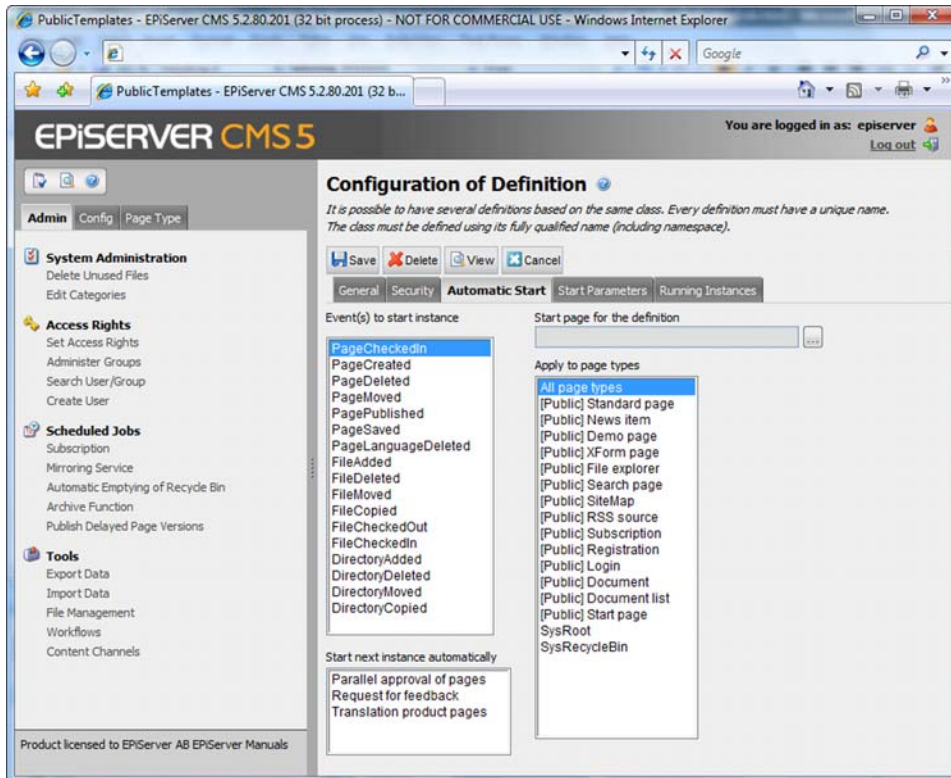


To give different people or groups authorization, you have to add them with the *Add Users/Groups* buttons. By selecting the different check boxes you can give different people/groups different authorization.

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 <i>Start Parameters</i> 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 <i>Ready to publish</i> 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 Web site. 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.

The screenshot shows the EPiServer CMS 5 Administration of Workflows interface. The main content area displays a table of running instances for the 'Sequential approval workflow'. The table has columns for Definition name, Creator, Details, View, and Terminate. One instance is listed with the creator 'epiuser' and a 'Terminate all' button. A 'Show' button is also present. The right-hand pane shows details for the selected instance, including Definition name, Creator, Page, History, Current tasks, Instance id, Type, and Waiting events.

Definition name	Creator	Details	View	Terminate
Sequential approval workflow	epiuser			Terminate

Number of instances: 1

Details:

- Definition name: Sequential approval workflow
- Creator: epiuser
- Page: New page [28_30]
- History:
 - 9/17/2008 3:45:04 PM: Workflow started
 - 9/17/2008 3:45:07 PM: Task created, subject:
- Current tasks: Subject: Approve page New page
- Instance id: a5f21107-ec30-46d2-aa71-4094a22e447a
- Type: EPiServer.WorkflowFoundation.Workflows.SequentialApproval, EPiServer.WorkflowFoundation, Version=5.2.80.201, Culture=neutral, PublicKeyToken=8fe83dea738b45b7
- Waiting events: approveEvent1, ApproveData

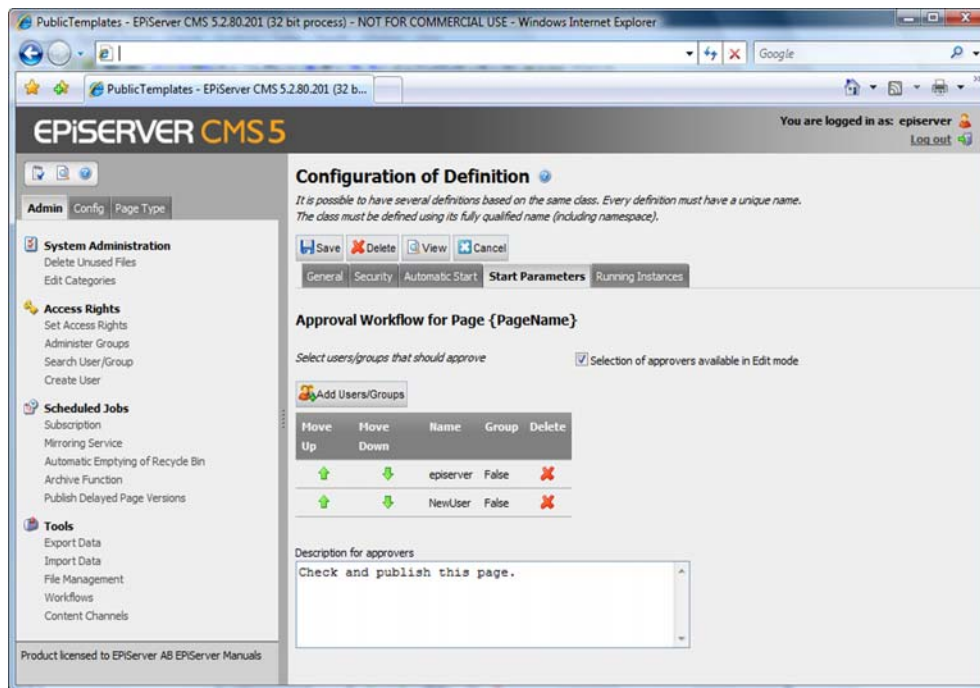
Start Parameters Tab

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 workflow as adaptable as possible from Admin mode instead of hard coding in values that will probably be variable. Below is a brief description of the different settings that can be made on the workflows included in a standard installation.

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.

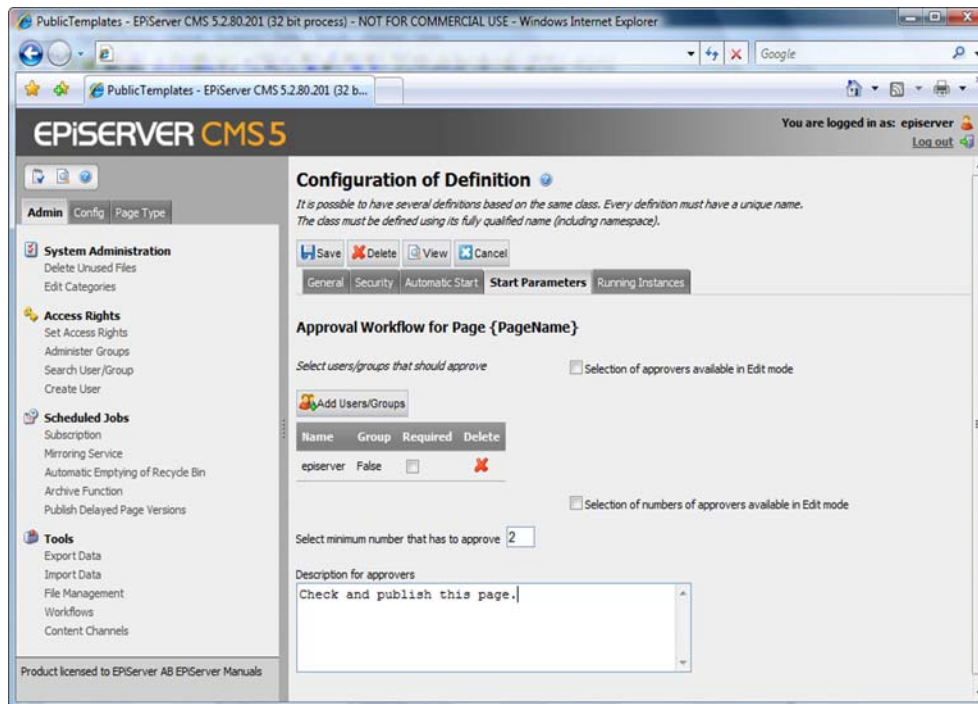
If we give the editors the access rights to edit the workflow, we should also select the *Selection of approvers available in Edit mode* check box here. Then the editor will be able to switch Jane or Pelle for other people or groups. This also requires that we give them the access rights to make changes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



Parallel Approval of Pages

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 of Pages with EPiServer Globalization Support

The workflow for translation of pages with EPiServer globalization support exists to enable the easy division of tasks for Web sites that are to be available in several languages. As an editor you feel that you only have one Web site, 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 Web site 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.

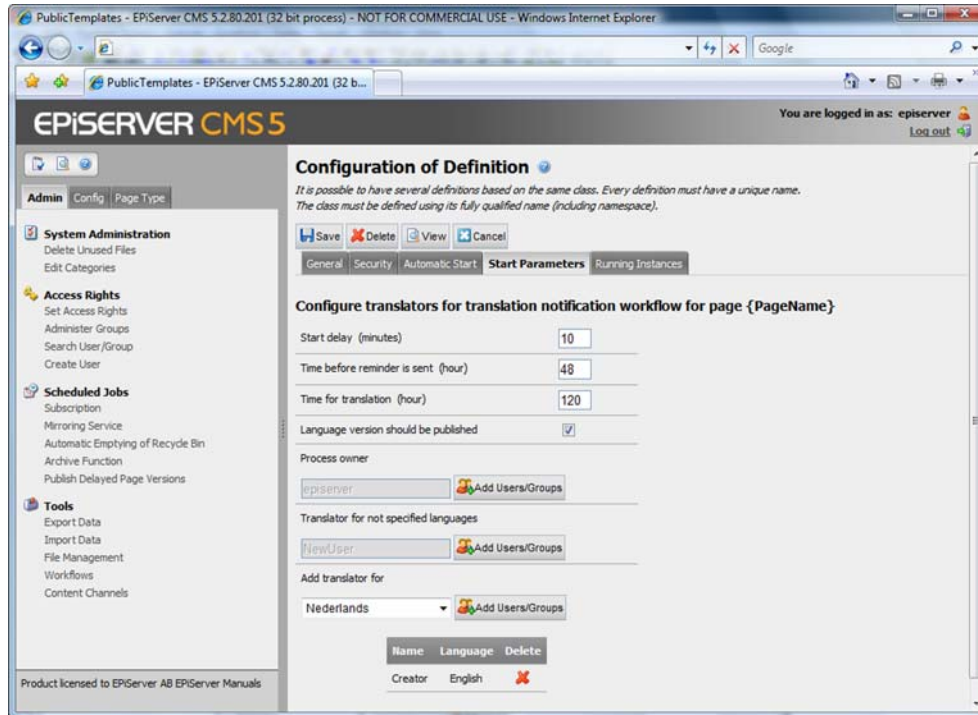
Time for translation (hour) The translation must be done and the page published before the time specified in this field. If it is not done, notification will be sent to the process owner.

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 Web site are shown in the drop-down list.

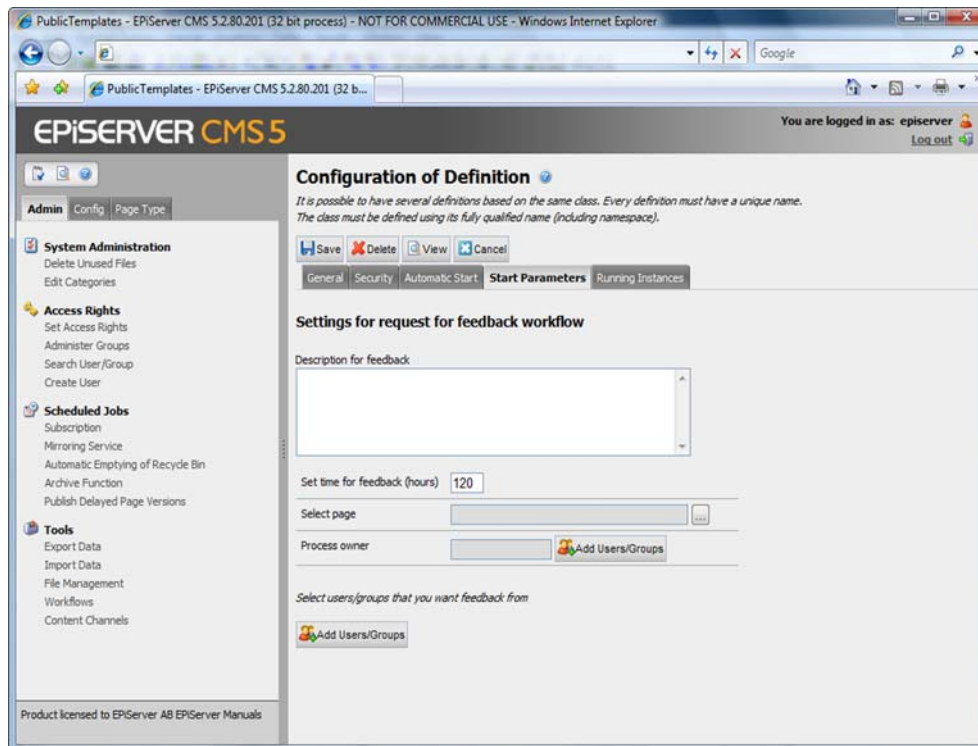


Request for Feedback

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.

Read about how, as an editor, you can start an instance on the workflow and the different settings that can be made in the Editor's Manual for EPiServer CMS 5 Release 2.



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 Web site 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 Web site 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 Web site. 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.

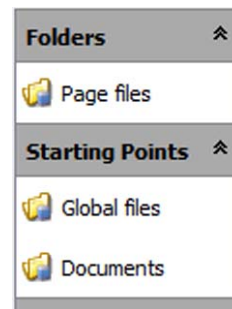
File management in Admin mode has the same functionality as in Edit mode. The difference is that the access rights in Admin mode are greater. Once you have chosen to use the File Manager fully, work starts in Admin mode to determine which groups should do what in various folders.

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 Web site.

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 `web.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 Web site in order for the management of access rights to work.

The configuration of starting points is done in `web.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" virtualPath="~/PageFiles/"
        bypassAccessCheck="false" name="SitePageFiles" type="EPiServer.Web.Hosting.VirtualPathVersioningP;
        indexingServiceCatalog="Web" physicalPath="C:\EPiServer\VFP\ExampleEPiServerSite2\PageFiles" />
    <add showInFileManager="true" virtualName="Global Files" virtualPath="~/Global/"
        bypassAccessCheck="false" name="SiteGlobalFiles" type="EPiServer.Web.Hosting.VirtualPathVersioning;
        indexingServiceCatalog="Web" physicalPath="C:\EPiServer\VFP\ExampleEPiServerSite2\Global" />
    <add showInFileManager="true" virtualName="Documents" virtualPath="~/Documents/"
        bypassAccessCheck="false" maxVersions="5" name="SiteDocuments" type="EPiServer.Web.Hosting.Virtual;
        physicalPath="C:\EPiServer\VFP\ExampleEPiServerSite2\Documents" />
    <add name="App_Themes_Default" virtualPath="~/App_Themes/Default/"
        physicalPath="C:\Program Files\EPiServer\CMS\5.2.375.7\application\App_Themes\Default"
        type="EPiServer.Web.Hosting.VirtualPathNonUnifiedProvider,EPiServer" />
    <add name="UI" virtualPath="~/EPiServer/UI/"
        physicalPath="C:\Program Files\EPiServer\CMS\5.2.375.7\application\UI"
        type="EPiServer.Web.Hosting.VirtualPathNonUnifiedProvider,EPiServer" />
    <add name="UtilFiles" virtualPath="~/Util/"
        physicalPath="C:\Program Files\EPiServer\CMS\5.2.375.7\application\util"
        type="EPiServer.Web.Hosting.VirtualPathNonUnifiedProvider,EPiServer" />
    <add name="WebServiceFiles" virtualPath="~/WebServices/"
        physicalPath="C:\Program Files\EPiServer\CMS\5.2.375.7\application\webservicess"
        type="EPiServer.Web.Hosting.VirtualPathNonUnifiedProvider,EPiServer" />
  </providers>
</filters />
</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 Web site, 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 Web site.

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. If not, start by reading *Working with Authorization* on page 9.

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.

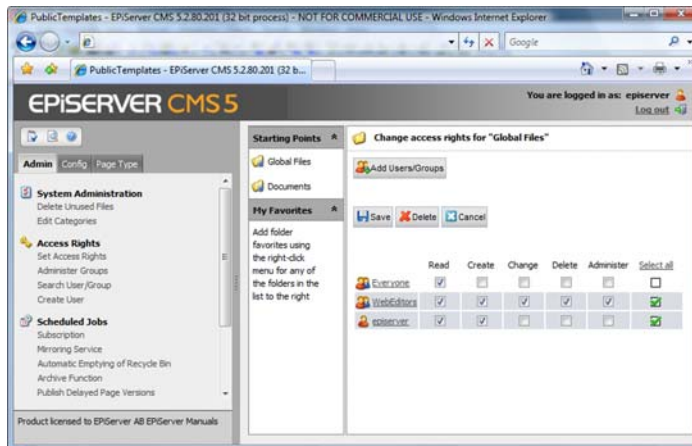
Delete means the group/user has the access rights to delete information from the folder.

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. Then 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 Web site, 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 have the opportunity of customizing the Edit mode to best suit your organization and your editors. It is therefore possible to use the editor's previous knowledge and modify Edit mode so that it is functions differently for different people. The idea behind EPiServer CMS is that it should be easy to use. "Easy" means different things to different individuals, therefore customization is required for it to be easy for everybody.

We can split the customization up into many different sections. It is possible to customize the page types to be used, the fields found in a certain page type, the tab on which the fields appear and who will see which tab. We can also work with various buttons in the Editor; those that will and will not be available. We can also help the editors with the default values in different fields. Finally, we can also work with the language in different sections, so that everybody understands the role of the different fields.

Most of this work is carried out by the solution supplier. In the majority of cases, there are ideas on how to update the Web site in a suitable way, as well as a good plan for how visitors should use the Web site. Even if this work has been carried out, it is useful to know what to do if changes are necessary.

The majority of these settings are made in the *Page Type* tab in Admin mode. There are also a number of functions there for creating new, copying page types, editing properties, etc. This part is covered briefly in this manual, as it primarily covers development work.

Page Templates vs. Page Types

EPiServer CMS features 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 in which the editor can enter information in EPiServer. Work in the EPiServer Editor and Admin modes is defined as primarily involving page types. In the Editor's Manual and in day-to-day discussions page types are often called page templates. Here, however, we will use the correct term, page types. Read more about page templates under *EPiServer CMS Page Templates* on page 62.

List of Page Types in Edit Mode

When an editor creates a new page, a list of page types that can be selected is displayed. The list can be customized to make it as easy as possible to create new pages and select the correct template. In Edit mode, it is the page type's sorting index that controls the list. In Admin mode you can choose if you wish to sort the list by sorting index or alphabetically, by clicking on the buttons to the left in the menu.

Page Type Sorting in the List

The sort order of the pages in Edit mode is created from the sort index that each page type has. The page type with the lowest index number will be first. All new page types have 100 as default.

Change the sort index of the page type as follows:

1. Select the page type you wish to change the sort index for, on the *Page Type* tab.
2. Click *Settings*.
3. Select the *Information* tab.

4. Enter a number greater or less than the other page types in the *Sort index* field.
5. Click *Save*.

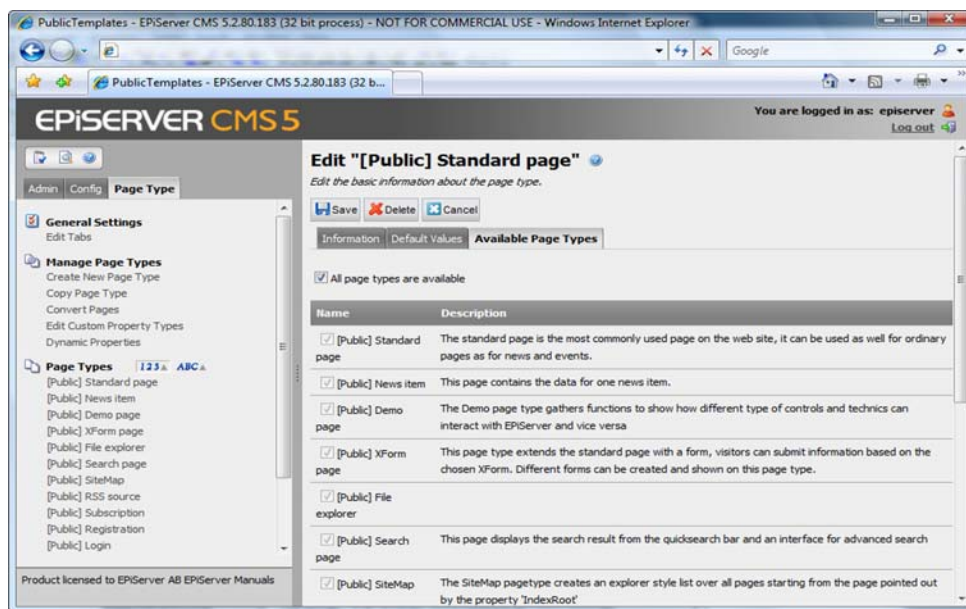
Available Page Types Under a Certain Page Type

It is possible to select the type of pages that are to be available for the editor when he/she creates a sub page for a certain page type. This function is used for calendars or news lists for example. Sub pages to the calendar page type should be of the page type Calendar Event. By only providing the editor with access to this page type, there need never be any mistakes. They have no other page types to choose from. It is also useful to hide page types that are not to be used with this function. The fewer page types the editor needs to choose from, the simpler it will be. Once you have created the page you require, e.g. for the search engine, you do not need to have it available any longer, as long as you do not plan to have several search pages.

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.

Select which page types will be available under another page type as follows:

1. Select the page type you wish to specify underlying page types for, on the *Page Type* tab.
2. Click *Settings*.
3. Select the *Available Page Types* tab.
4. The *All page types are available* check box is often selected. In order to be able to select specific page types, you must clear this check box.
5. Specify which page type(s) you wish to be available under the page type that you are now editing by selecting the check boxes in front of the page type's name.
6. Click *Save*.



Page Type Available in Edit Mode

It is possible to completely hide a page type in Edit mode. It is not associated with a particular page type. One scenario can be with the page type *Search page*. It is quite common to have only one search page on the Web site and not several in different places. In this instance, create

the search page and then hide the page type. Editors will not then be able to access the template, which will prevent more than one from being created.

Hide a page type in Edit mode as follows:

1. Select the page type you wish to hide from Edit mode, on the *Page Type* tab.
2. Click *Settings*.
3. Select the *Information* tab.
4. Clear the *Available in Edit mode* check box.
5. Click *Save*.

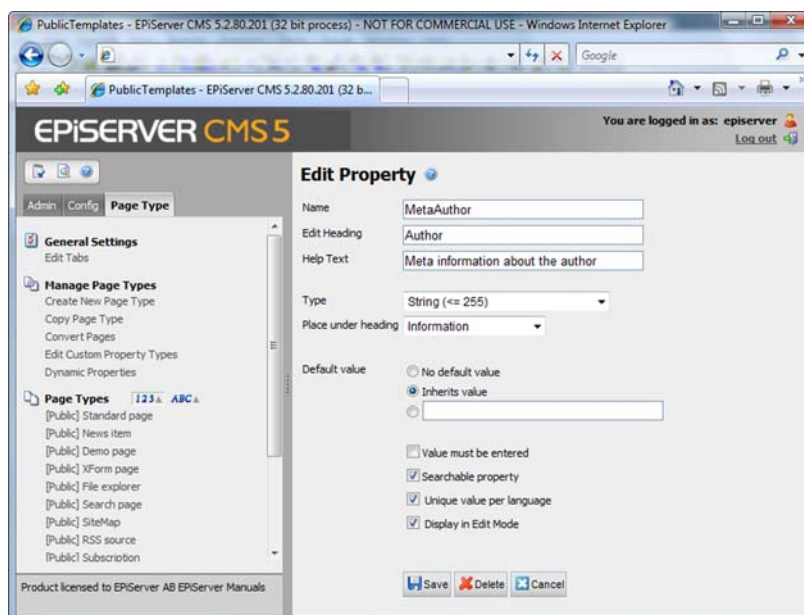
Default Value of Field

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 are default fields on all templates, fields that cannot be removed. Each page type also contains a number of properties that mean 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 sub-pages 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 as follows:

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*.

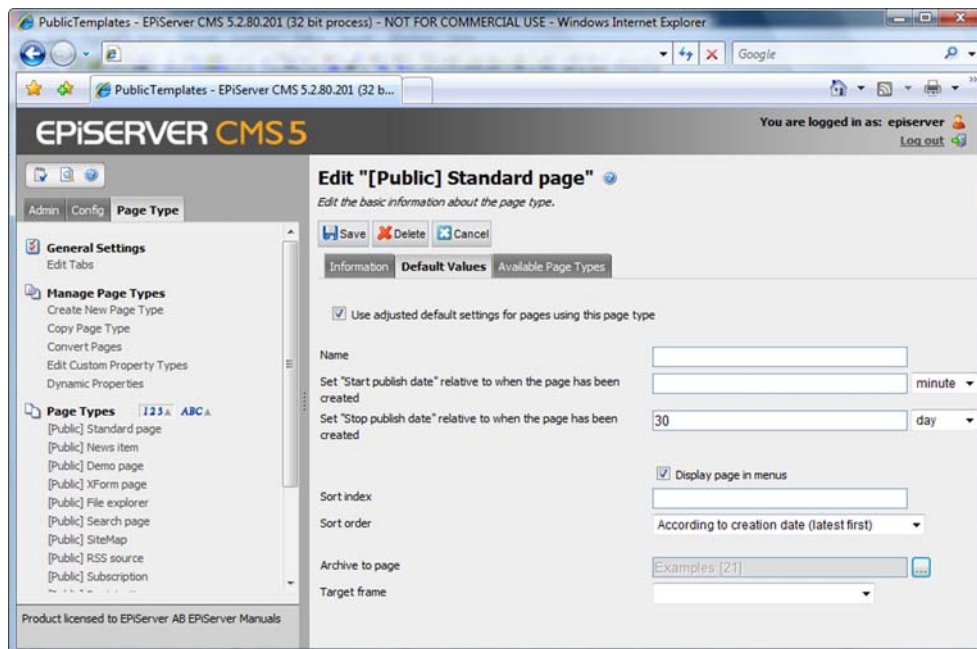


Default Values of Default Field

A good number of properties (fields) in all EPiServer CMS templates cannot be selected for removal. It is, however, possible to set default values for these properties to help the editors with their work. One scenario can be used in the News Page template. We have a policy for news on the Web site that states that news is news for 30 days after which it must be moved to the News Archive. In order to make creating news easier, we specify these two properties as defaults on the News Page template.

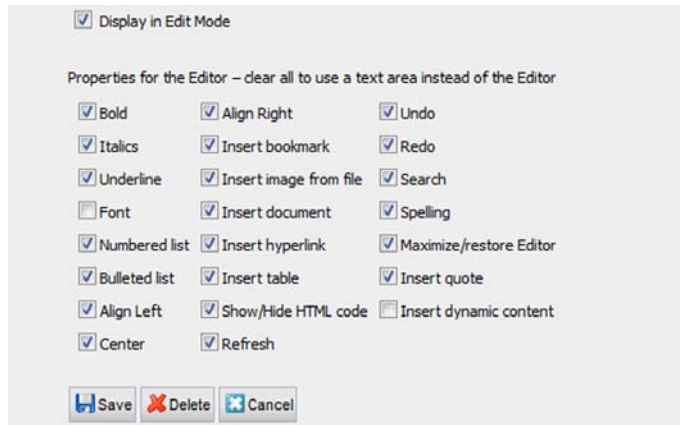
Change the default values of EPiServer's properties as follows:

1. Select the page type you wish to specify underlying page types for, on the *Page Type* tab.
2. Click *Settings*.
3. Select the *Default Values* tab.
4. Specify the default values for the fields you wish to have values for.
5. Click *Save*.



Editor Buttons

If you edit/create a property that has more than 225 characters, you get an Editor as standard. You can then always choose which buttons will be available for each Editor. Under *Properties for the Editor*, you can specify which buttons are to be available. If you clear all check boxes, the properties will reflect a normal text area input field. In a standard installation the *Font* and *Insert dynamic content* buttons are not selected and do not therefore appear in the Editor by default.

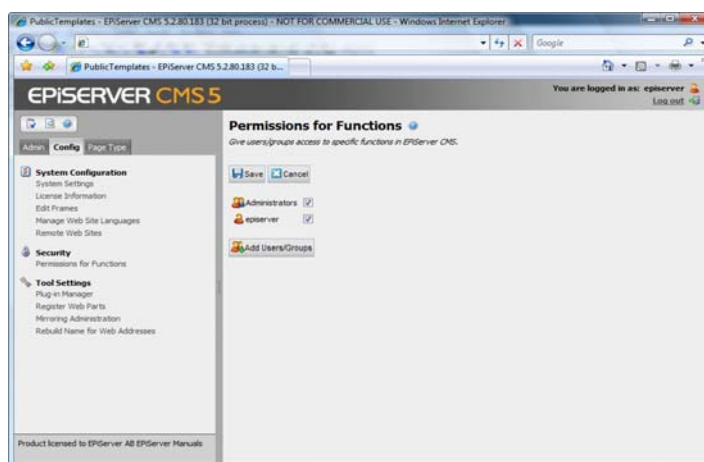


Certain Editors Require All Buttons in the Editor

In most cases, it is a good idea to hide the buttons that an editor does not need, or that might cause more damage than help. It is common to remove the *Underline* and *Show/Hide HTML code* buttons for example. Some editors and administrators feel that they need all the buttons to be able to correct any faults that may have occurred. In such cases, it is possible to give these groups and/or individuals access to all buttons.

You can give certain groups/individuals access to all buttons in the Editor as follows:

1. Select *Permissions for Functions* on the *Config* tab.
2. Click *Edit* after *All functions available in the Editor*.
3. The groups/users that have all buttons available are displayed in the list.
4. Add or remove groups and users so that only those who are to have access to all buttons are displayed.
5. Click *Save*.



Properties for the Editor

It is possible to define certain settings for the Editor in EPiServer CMS. It is possible to determine the size of the Editor and how pressing ENTER should be interpreted. For further information, see *Editing* on page 71.

It is also possible to extend the list with various format templates that can be selected in the Editor. Read more under *Editor Format Template List* on page 63.

Compulsory Fields

In EPiServer CMS, it is possible to force the editors to fill in certain fields. The setting is made on each property for every page type.

This is how you set compulsory fields:

1. Select the page type where the property you wish to be enforced is, 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*, at the bottom of the page.

Hide Fields in Edit Mode

In EPiServer CMS, it is possible not to show all the fields in Edit mode. This of course means that the editor cannot change the value that is saved for the property. This can sometimes be the purpose. The setting is made on each property for every page type. As default, all the fields are visible in Edit mode.

Hide a field in Edit mode as follows:

1. Select the page type where the property you wish to be enforced is, on the *Page Type* tab.
2. Click the name of the property that you wish to modify.
3. Clear the *Show in Edit mode* check box.
4. Click *Save*, at the bottom of the page.

Field Searchability in EPiServer's Search Engine

In EPiServer CMS, there is a function for searching the Web site content. The search does not need to be carried out in all fields. 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.

Exclude a certain field from the search as follows:

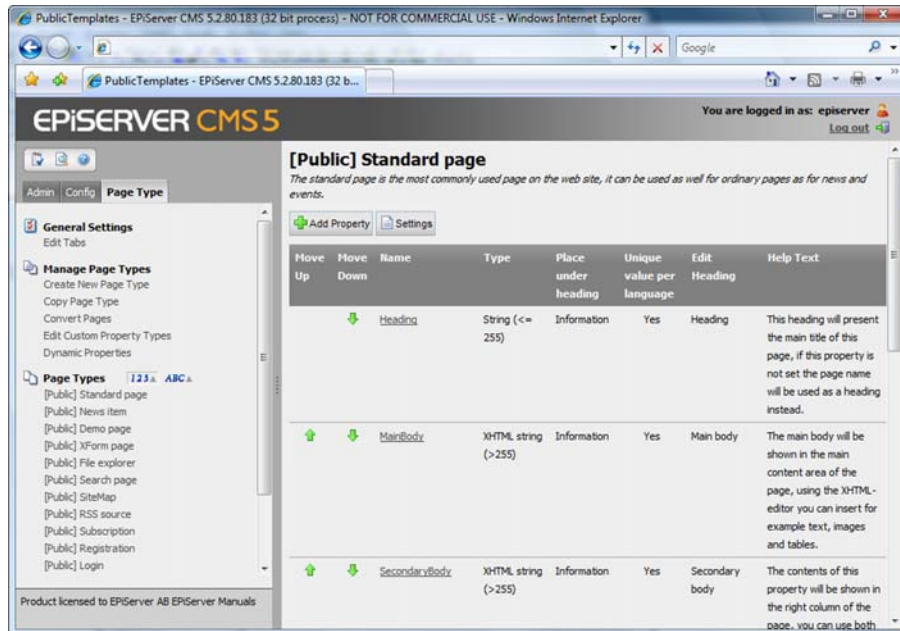
1. Select the page type where the property to be searched is, on the *Page Type* tab.
2. Click the name of the property that you wish to modify.
3. Clear the *Searchable property* check box.
4. Click *Save*, at the bottom of the page.

Field Order in the Page Type and Tab Location

All fields 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 as follows:

1. Select the page type you wish to change the field order for, 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.

**Select which tab a certain property is to be placed on as follows:**

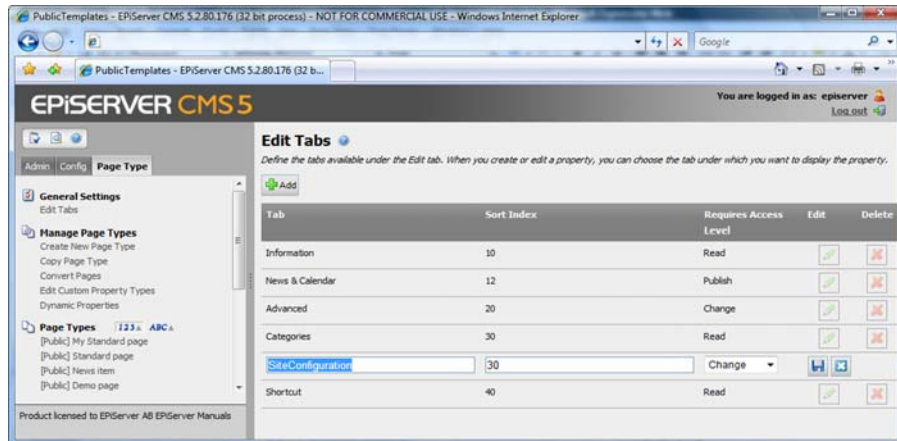
1. Select the page type with the property you wish to place on a particular tab, on the *Page Type* tab.
2. Click the name of the property that you wish to modify.
3. In the *Place under heading* scroll bar you can select which tab the relevant property is to be placed on.
4. Click *Save*.

Edit Tabs

It is possible to divide your properties on more tabs by selecting the *Edit Tabs* function. Here you can also decide in which order the tabs should be and which access rights should apply to a certain tab.

Add/edit tabs as follows:

1. Select *Edit Tabs* on the *Page Type* tab.
2. Click *Add* to create a new tab. Click the tab name to edit a tab.
3. Specify the tab name in the *Tab* field.
4. Specify the index amount for the tab in the *Sort Index* field. The lower the value the further to the left the tab is placed.
5. In the *Requires Access Level* field, 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. Read more under *Setting Access Rights in the Tree Structure* on page 15.
6. Click *Save*, to the right.



Working with Page Types

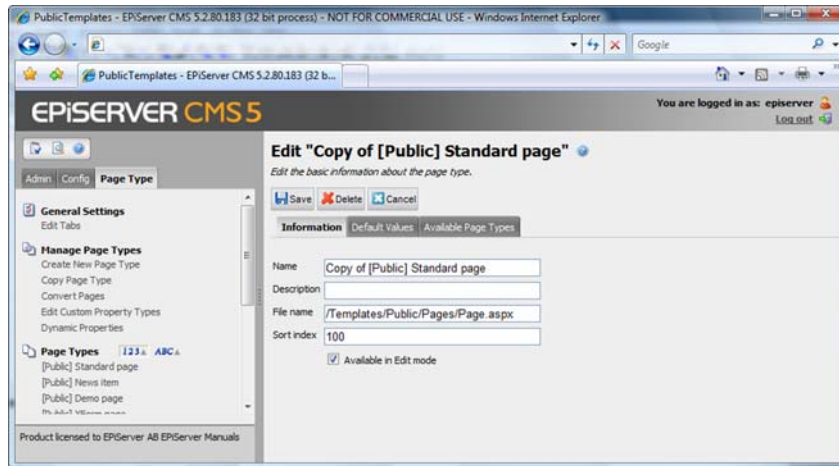
EPiServer CMS features both page templates and page types. Page templates are .aspx pages that are displayed to visitors and contain the functionality required to perform a specific function. Page types, on the other hand, are forms in which the editor can enter information in EPiServer. Read more about page templates under *EPiServer CMS Page Templates* on page 62.

Create New Page Type

Administrators do not normally create new page types; this is primarily the task of developers. Each page type is linked to an .aspx page (page template), where all the functionality of the page is programmed. If you do not have any knowledge of programming in ASP.NET, you should not create new page types. The fields in the form that are filled in by the editors are created in Admin mode.

Create a new page type as follows:

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*. Read more under *Page Type Available in Edit Mode* on page 40.
6. Specify the index amount for the page in the *Sort index* field. Read more under *Page Type Sorting in the List* on page 39.
7. Click *Save*.



Copy Page Types

To facilitate the creation of 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 its properties follow, so you need not create them again. You can then edit the information for the page type and its properties.

Convert Pages

The Convert Pages function allows you 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 Web 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.

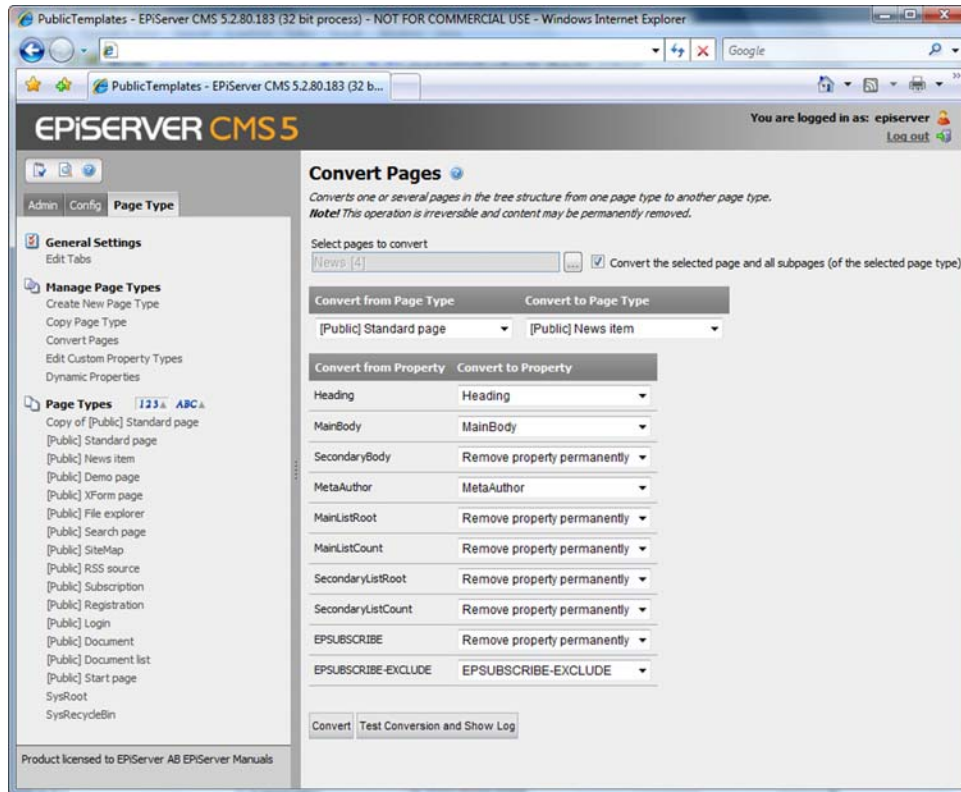
Each page type contains a number of different properties. These properties generate the fields in the page type. When you convert a page, you in reality convert the content in the page properties in the old page type to corresponding properties in the new page type.

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 subpages 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. 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 Web site, so think carefully before you decide to permanently remove a property.

- 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.



Create Property

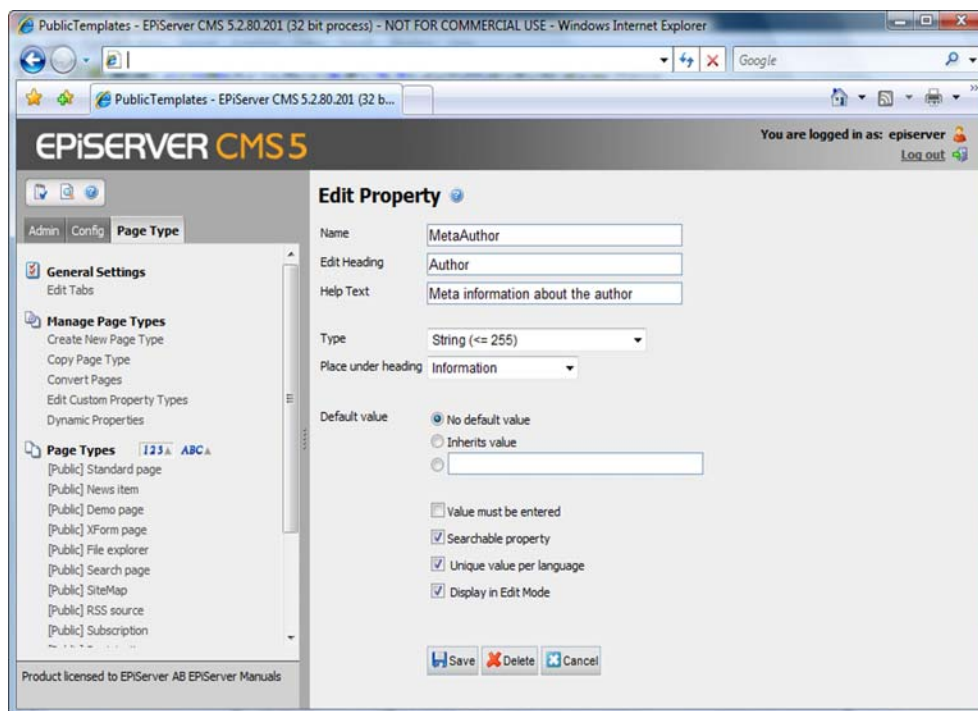
Each page type contains a number of different properties. These properties generate the fields in the page type. It is not common practice to create new properties, because these must always be linked with the .aspx page that displays the content on the page.

Create a property as follows:

- Select the page type where the property is to be created, on the *Page Type* tab.
- Click *Add Property*.
- Enter a name for the property in the *Name* field. This name is used when programming and when making language settings.
- Specify a heading for the property in the *Edit Heading* field. The text is displayed for the editor when he/she uses the page type, if the field is not language encoded.
- Specify help text in the *Help Text* field. The text is displayed for the editor when he/she uses the page type, if the field is not language encoded.
- Select a field type in the *Type* field. Read more about custom property types under *Custom Property Types* on page 49.
- Select the type of field that is to be in the *Type* field. There are a number of different property types to choose between in EPiServer CMS. You can choose integer, string, page,

date etc. When you select Long String, you always have the option of using the HTML Editor. It is also possible to create your own property types.

8. Select the tab under which the properties are to be displayed, under the *Place under heading* field. Read more under *Field Order in the Page Type and Tab Location* on page 44.
9. Select whether to have a default value on the tab. Read more under *Default Value of Field* on page 41.
10. Specify if the field is compulsory or not with the *Value must be entered* check box. Read more under *Compulsory Fields* on page 44.
11. Specify whether the properties are to be searchable or not with the *Searchable property* check box. Read more under *Field Searchability in EPiServer's Search Engine* on page 44.
12. Specify whether the field should be unique for a certain language if the Web site has activated support for globalization by selecting the *Unique value per language* check box. Read more under *Globalization in EPiServer CMS* on page 54.
13. Specify whether the property is to be visible in Edit mode with the *Display in Edit mode* check box. Read more under *Hide Fields in Edit Mode* on page 44.



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 built-in data types for properties 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:

- Built-in properties are always present and are supplied by the EPiServer CMS runtime system. Examples of built-in properties are PageName (name of the Web page) and PageStartPublish (start publish date for the page). Read more about built-in properties in the EPiServer CMS Online SDK, <http://sdk.episerver.com>.

- Custom properties are handled in Admin mode by developers and administrators and can be divided into two types: static properties and dynamic properties. Read more about dynamic properties in the Editor's Manual.

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

It is always possible to create new types by inheriting from the existing ones. There are 10 property data types available in EPiServer CMS. (The names of property data types are slightly different in Admin mode; keep this in mind when you add properties in Admin mode.)

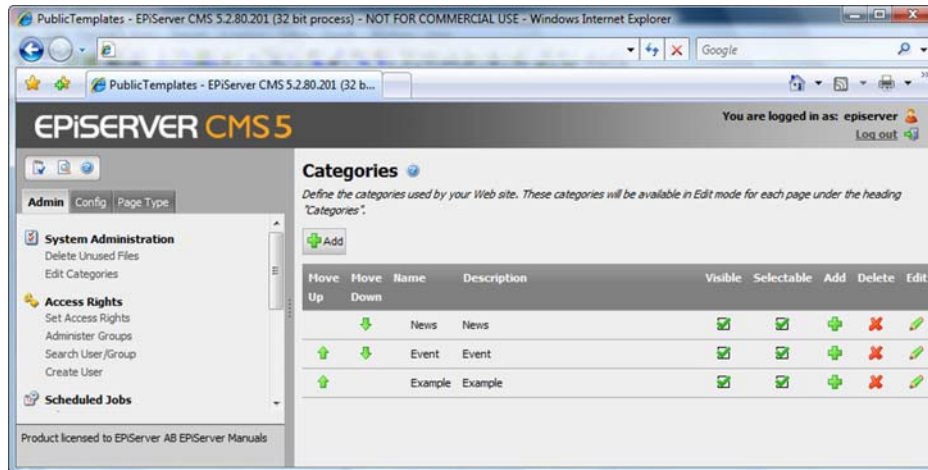
Data Type Name	Data Type Name in Admin Mode	Description
Boolean	Selected/not selected	True or false.
Number	Integer	An integer.
FloatNumber	Floating point number	Number with decimals, e.g. 3.14.
PageType	Page Type	EPiServer CMS page type defined in Admin mode.
PageReference	Page	Link to a Web page.
Date	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.






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 Web sites 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.

Languages in Edit and Admin Mode

EPIserver CMS is available in twelve languages as standard: English, Swedish, Norwegian, Danish, Finnish, German, French, Spanish, Portuguese, Dutch, Chinese and Japanese. More languages can be implemented, if required. 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. He or she can change the language under *My Settings*.

The system language for the entire Web site can be changed in `web.config`. You specify the system language during installation; the value will be written in `web.config`. This can be changed if you switch the language prefix on "uiCulture". See the image below.

```
<!-- GLOBALIZATION
      This section sets the globalization settings of the application. |      -->
<globalization requestEncoding="utf-8" responseEncoding="utf-8"
  culture="sv-SE" uiCulture="sv" resourceProviderFactoryType="EPIserver.Resources.XmlResourceProviderFactory, EPIserver" />
```

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>
  <language name="English" id="en">
    <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. Read more under *Create Property* on page 48. 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.

Set Preset Sizes in Image Editor

Image Editor is used when working with images in EPiServer CMS. Images can be cropped, resized and transformed with *Image Editor*, and it is also used to set the image alignment, border widths and spacing. Refer to the Editor's Manual for further information on how to insert and work with images in *Edit mode*.

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 web.config file.

You can choose to change the preset sizes or add more sizes in the web.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 web.config file.

The image below displays how web.config should be configured to achieve the required preset values in the Image Editor.



Help Files

The help files available in EPiServer CMS are HTML pages and are stored in the same folder as the other EPiServer CMS application files, i.e. EPiServer\CMS\<version>\Application\UI\Help. The pages are linked to the relevant page in EPiServer CMS via their file names. All functions in EPiServer 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.

Globalization in EPiServer CMS

EPiServer CMS supports being able to work with several page languages/variations simultaneously. This means that a tree structure can consist of several parallel trees, one for each language. This function is not only useful on Web sites that have several languages. Globalization can also be used if you, for example, want an easily read part of a Web site. It is another variation of a page even if it is written in the same language.

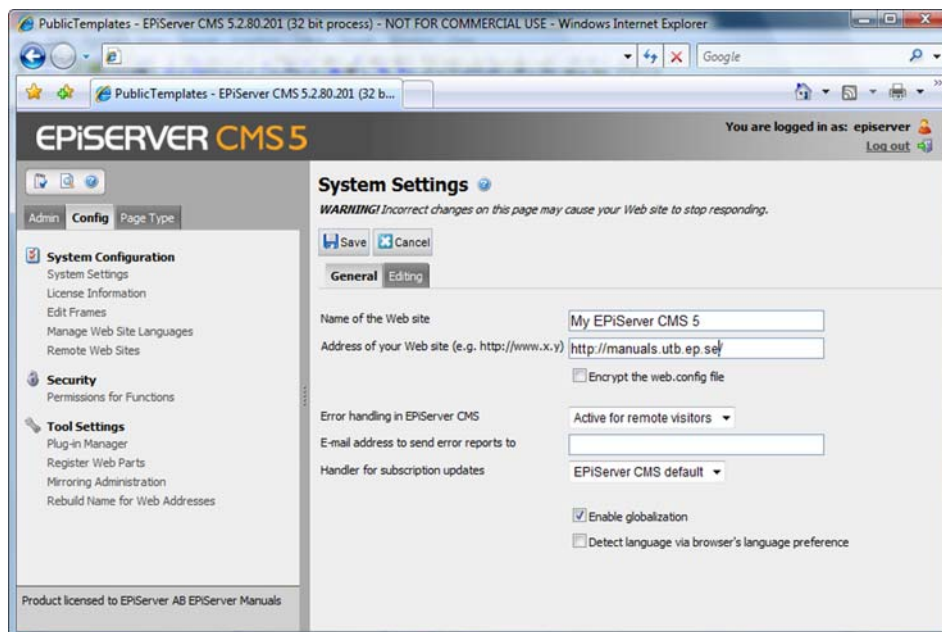
In the *Editor's manual* there are descriptions of how to work with the globalization support in your daily work. For globalization to work, it is important to set up the support from the beginning. Most of this work is done from Admin mode. It is important to activate the globalization and choose the languages to be available. You should also decide which properties in each template should be translated.

Enable Globalization

If the Web site is going to use several languages/variations of the same page, the globalization must be enabled.

Activate globalization as follows:

1. Select *System settings* on the *Config* tab.
2. Select the *General* tab.
3. Select the *Enable globalization* check box.
4. Click *Save*.



The Web Site's Default Language

The default language is the language that shall apply first when a visitor arrives at the Web site. 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. Read more under *Which Language is Shown to Visitors?* on page 56.

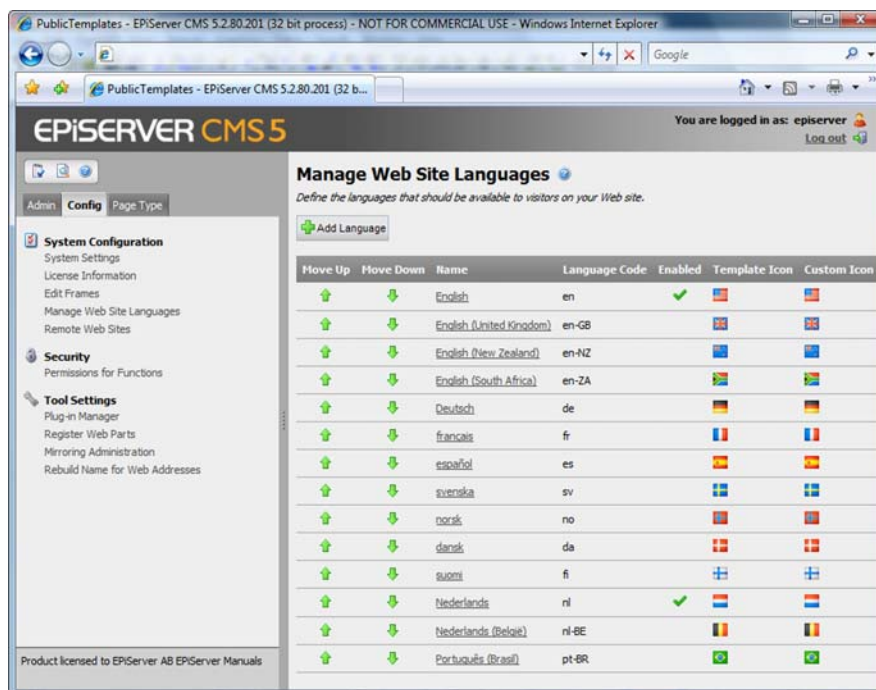
The default language for the entire Web site can be changed in web.config. You can do this by changing the language encoding in the "Culture" attribute. See the image below.

```
<!-- GLOBALIZATION
      This section sets the globalization settings of the application. |      -->
<globalization requestEncoding="utf-8" responseEncoding="utf-8"
      culture="sv-SE" uiCulture="sv" resourceProviderFactoryType="EPiServer.Resources.XmlResourceProviderFactory, EPiServer" />
```

Manage Web Site Languages

If you want your editors to be able to write contents in a particular language, the language must be added, and activated for use. You do this under *Manage Web Site Languages*.

When you open *Manage Web Site Languages*, all added languages are shown in a long list. Straight away you can see which languages are available and if they are active or not. If you wish to change the settings for a language, click on the language's name. You can adjust the order of the various languages using the arrows to the left of the language.



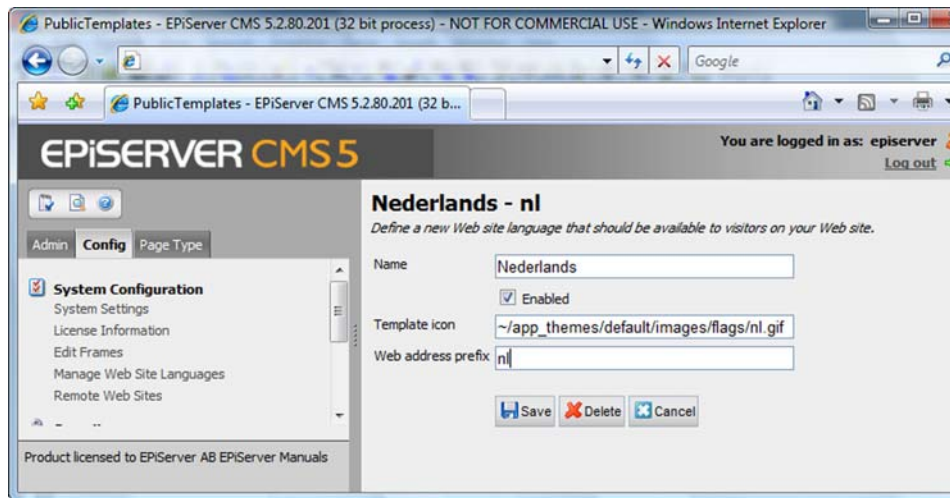
Add New Language for Multilingual Web Sites

For an editor to be able to create content in several different languages, the language must be added under *Manage Web Site Languages* and be active.

To add a new language that will be accessible for contents management, do the following:

1. Select *Manage Web Site 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. It 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.
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 `MyEPiServer\App_Themes\Default\Images\Flags` directory; you are free to use these. If you are going to make your own icons, these should measure 15x15 pixels so that they fit properly in the Edit mode.
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.foretaget.se/is`. Read more under *Which Language is Shown to Visitors?* on page 56.
8. Click *Save*.



Edit Language for Multilingual Web Sites

To edit an existing language that will be accessible for contents management, do the following:

1. Select *Manage Web Site Languages* on the *Config* tab.
2. Click the language's name.
3. Make your changes.
4. Click *Save*.

Delete Language for Multilingual Web Sites

To delete an existing language that will not be accessible for content management, do the following:

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

Which Language is Shown to Visitors?

The language for visitors can be selected and managed in different ways in EPiServer. 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 key will work as a prefix. If, for example, I want to view an English page of a Web site that is normally Swedish, I can enter the address `www.foretaget.se/en`. The English start page then comes up, provided it is available in English.

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

To activate detection via the browser's language do the following:

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*.

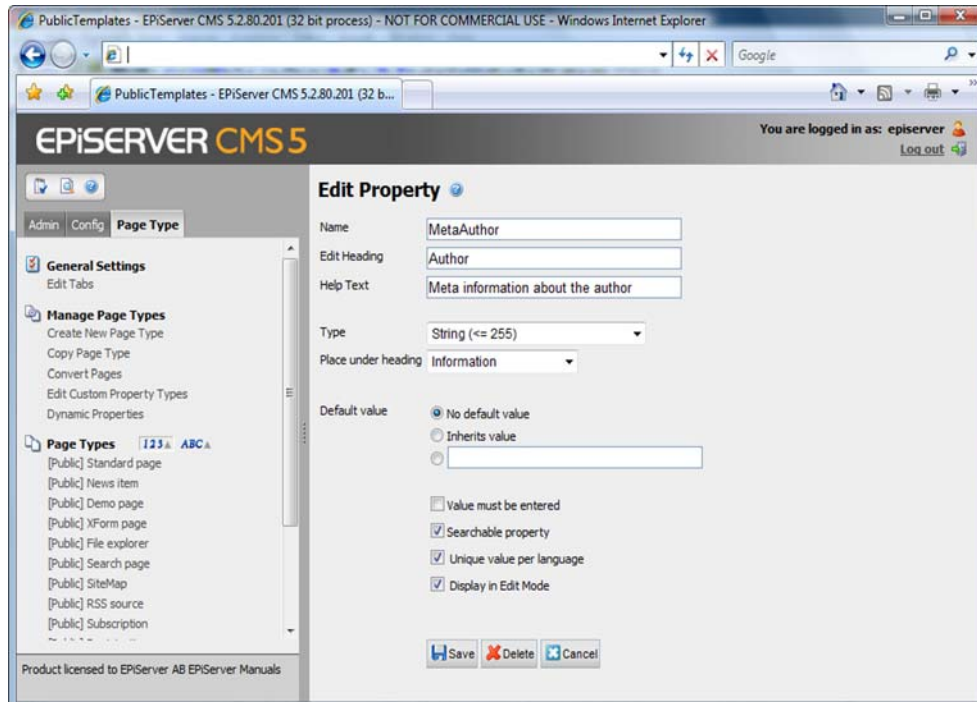
With this function, the languages that are chosen as the default languages in the visitor's browser will be shown first. This is a setting that every visitor can manage themselves. There are also different programmable properties that can control which language is displayed, for example, a cookie. It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

Fields that Vary per Language

When you work with globalization, you decide in every template which fields will vary depending upon the language. It is normal that not all fields vary. The values of those fields that won't vary are always provided in the original language of that page. The fields that will vary are provided on each template and property.

To language vary a property/field do the following:

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. Select the *Unique value per language* check box.
4. Click *Save*.



Language Settings Can Vary for Different Parts of the Tree Structure

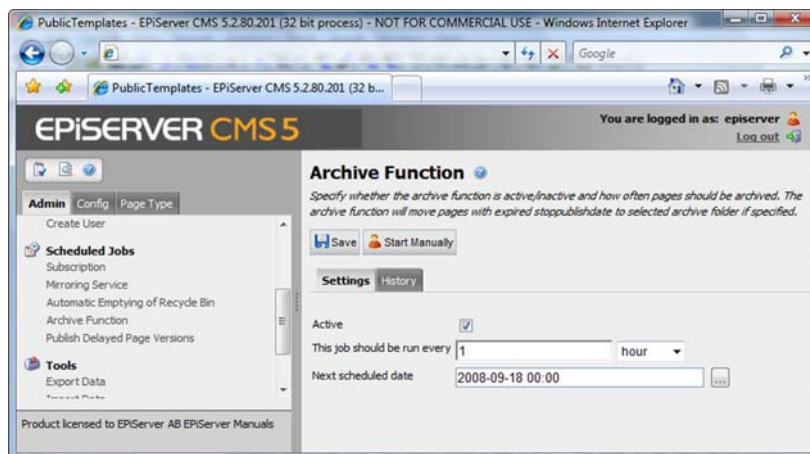
The languages that will be accessible can vary in various places in the tree structure. It is possible to language code a small part of the Web site, instead of doing it on the entire structure at once. It is also possible to have so-called replacement languages. This means that you can define, in advance, what applies if a page is not available in all languages, which language would then be shown to the visitor. These settings are made in Edit mode and are therefore described in the *Editor's Manual*.

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.

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.



Mirroring Service

Using *Mirroring Service*, you can set how often the system is to activate the mirroring of contents for other Web sites. If your Web site is set up to mirror content from one Web site 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. Read more about this under *Mirroring Administration* on page 75.



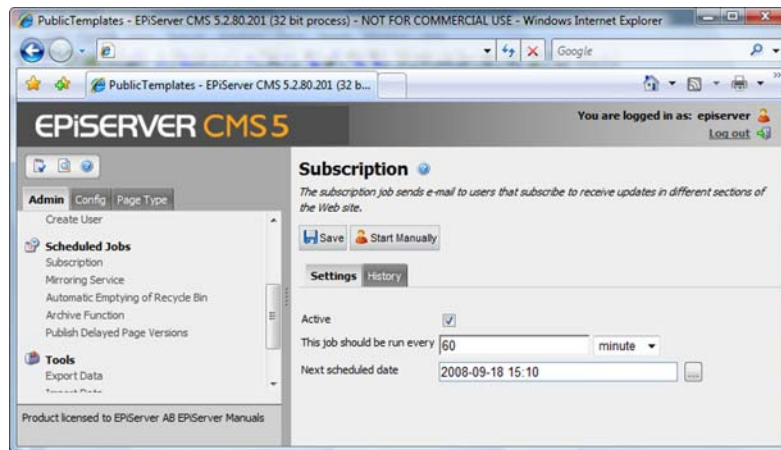
Automatic Emptying of the Recycle Bin

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.



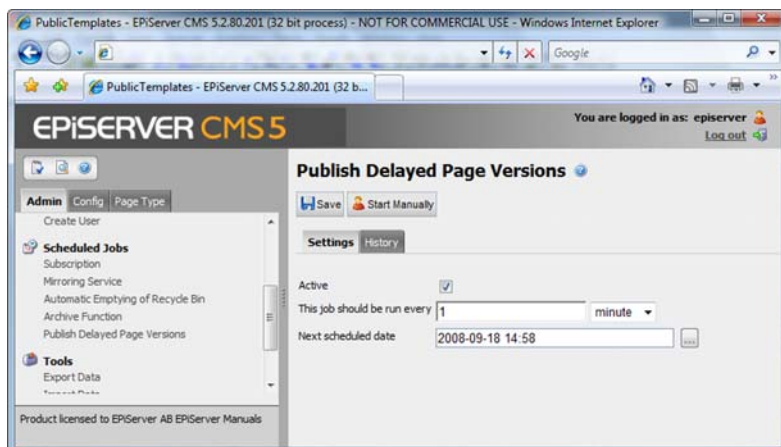
Subscriptions

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 decides how often he/she wants 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.



Publishing Delayed Page Versions

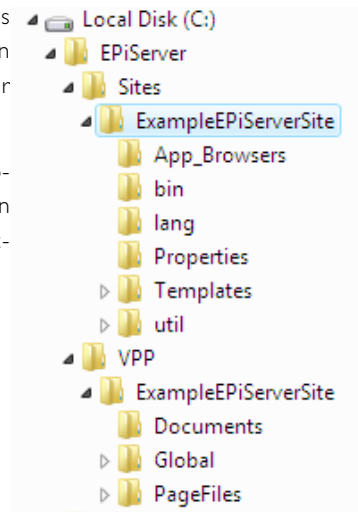
Using *Publishing Delayed Page Versions*, you can activate and set up how often the system is to check if there are versions of a page for publication at a set time. Delayed publish of a whole page is not controlled by this; it only takes place when an editor has selected Start Publish for a single version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



Files for EPiServer CMS

EPiServer CMS Web sites 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 Web site 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 Installation Instructions on EPiServer World.



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 Web site.
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 Web site. This folder also contains images and style files for the Web site.
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.

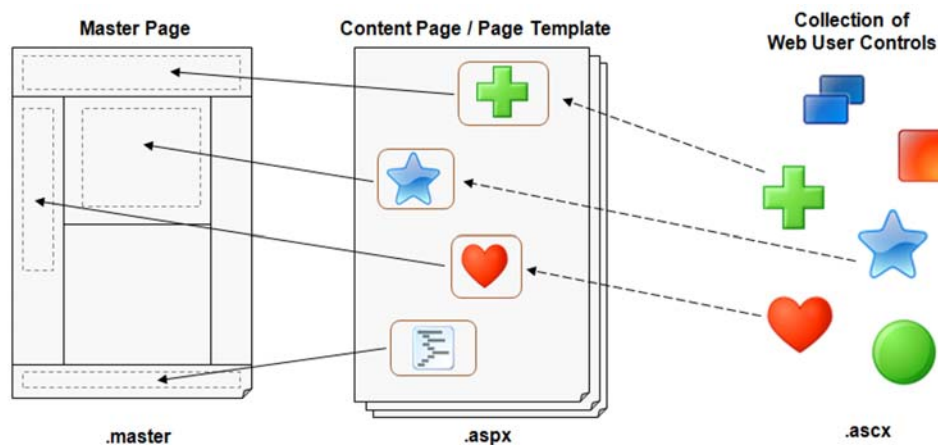
It is fully possible to change the design of the Edit and Admin modes if you wish. In the Application/App_Themes folder there is a directory called Default. Copy the Default folder and change the images and settings in the CSS files to create a new appearance for the Edit and Admin modes.

EPiServer CMS Page Templates

Each Web site built on EPiServer CMS is customized for each unique organization. A large part of the customization is the design of the Web site. Another part is the functionality for the visitors. On EPiServer's sample Web site, 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 set-up 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.



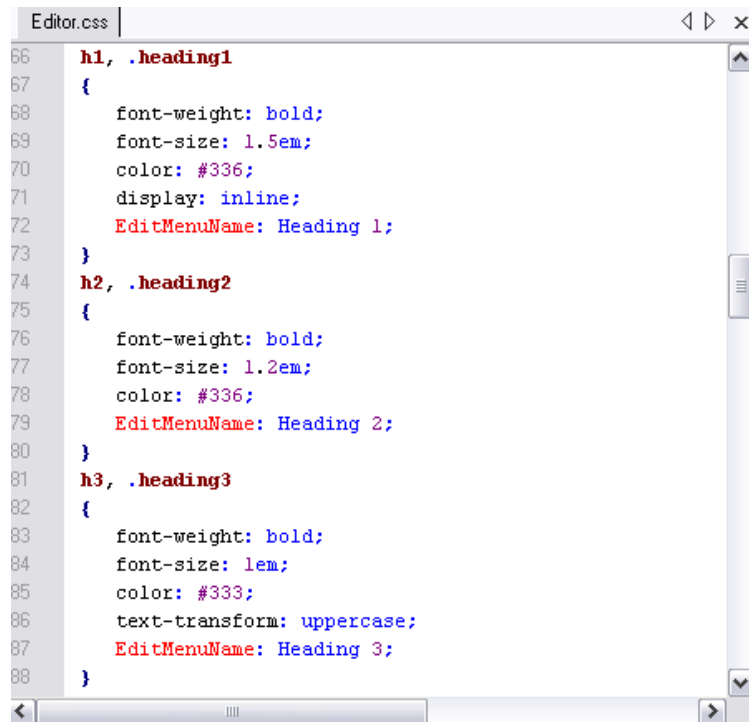
Working on the Design of the Web Site

The design of the Web site 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 Web site. This is common in, amongst other things, an Enterprise solution, which has a number of Web sites 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.



```
66  h1, .heading1
67  {
68      font-weight: bold;
69      font-size: 1.5em;
70      color: #336;
71      display: inline;
72      EditMenuName: Heading 1;
73  }
74  h2, .heading2
75  {
76      font-weight: bold;
77      font-size: 1.2em;
78      color: #336;
79      EditMenuName: Heading 2;
80  }
81  h3, .heading3
82  {
83      font-weight: bold;
84      font-size: 1em;
85      color: #333;
86      text-transform: uppercase;
87      EditMenuName: Heading 3;
88  }
```

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.

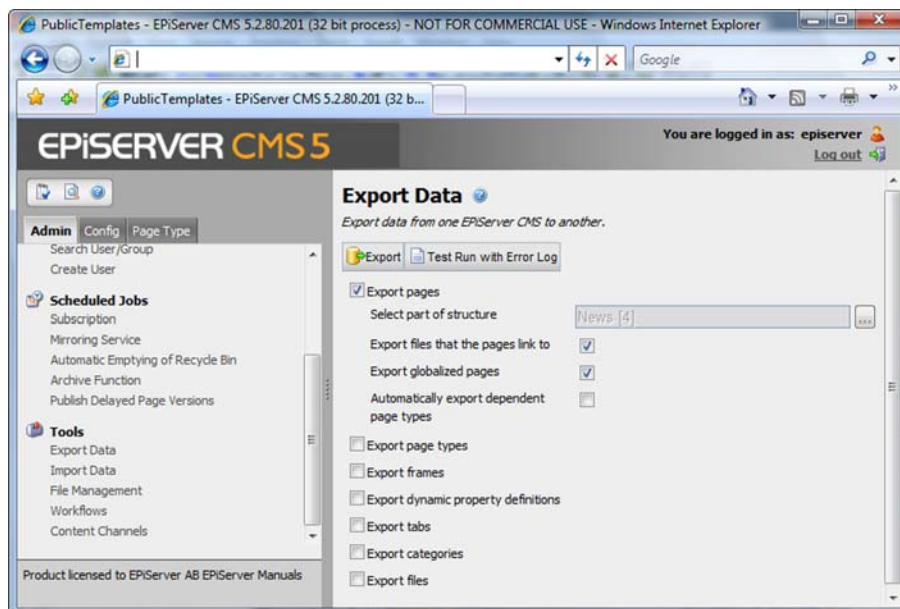
Export Data

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

The following can be exported:

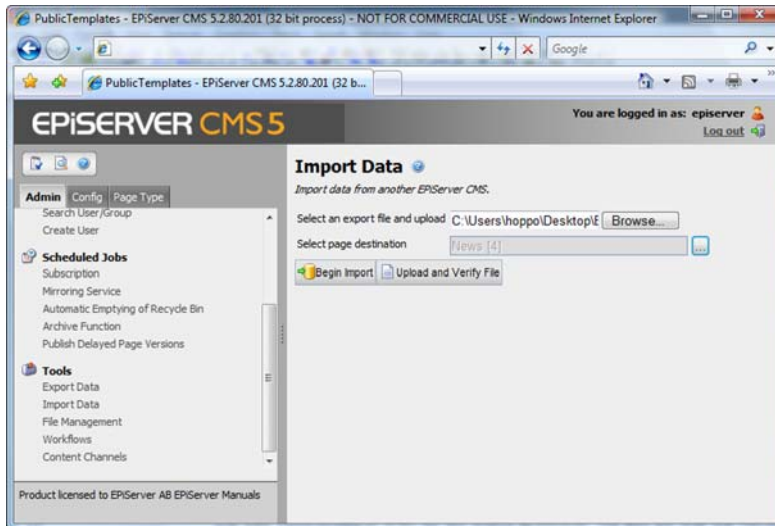
- Pages with (if you so choose) associated page types
- Page types
- Frames
- Dynamic property definitions
- Tabs
- Categories
- Files

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 Web site. 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 by clicking *Begin Import*.

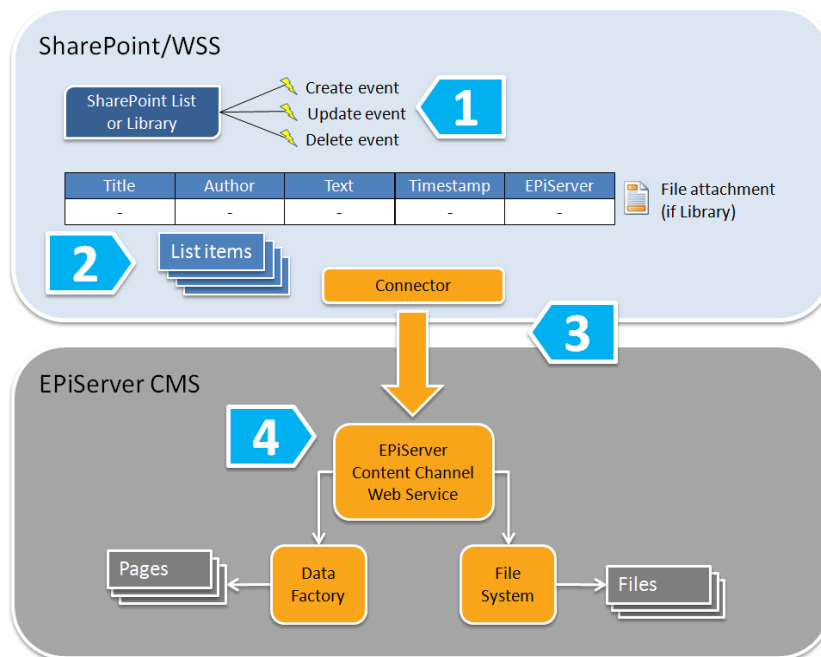


Integration with External Systems

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 Web site running EPiServer CMS. After installing and configuring the connector, SharePoint users can mark content and documents that should be shipped to the EPiServer CMS Web site 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.

For more specific information about the SharePoint connector, see <http://www.episerver.com/connector2007>.



Process description (see image above)

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

Note 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 in the technical note, "EPiServer Connect for Microsoft SharePoint", on EPiServer World.

Content Channels

Content is retrieved from external systems through channels on the Web site 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 **Save**.

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 Property 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 Property Mapping for Page Type**.

EPISERVER CMS 5 You are logged in as: episerver

Configure Content Channel

Each channel can be configured for several page types and a property mapping can be defined for each of these page types.

Save Delete Cancel

Channel Name:

Page root:

Virtual root for files:

Save action:

Default page type:

Map properties for page types

Unmapped page types:

Mapped page types:

Property mapping for page type *[Public] News item*

Page Type Property	Mapped Property
PageName	<input type="text" value="Title"/>
Heading	<input type="text" value="Title"/>
MainIntro	<input type="text"/>
MainBody	<input type="text" value="Body"/>
MetaAuthor	<input type="text"/>
EPSUBSCRIBE-EXCLUDE	<input type="text"/>

Add Property Mapping for Page Type Remove Property Mapping for Page Type

Product licensed to EPiServer AB EPiServer Manuals

Configuration of EPiServer CMS

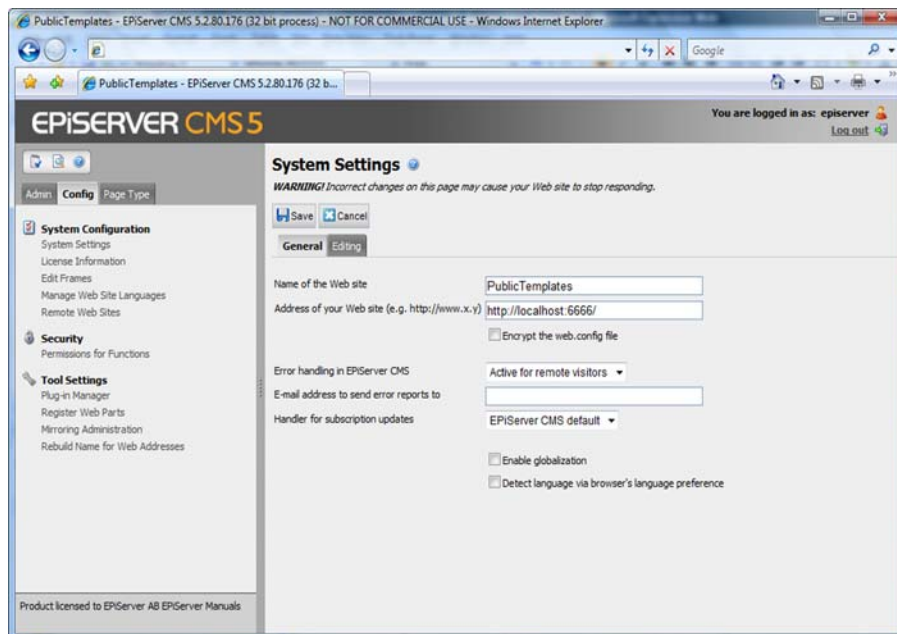
When installing EPiServer CMS in live operational environments, many settings are configured once and then seldom modified. They can include various settings for visitors and editors, as well as purely technical fixes.

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



Name of the Web site

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

Address of your Web site

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

Encrypt the web.config file

The web.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.

Handler for subscription updates

There is a subscription function in EPiServer CMS that allows visitors to the Web site to receive information about new and updated pages. Depending on whether you have a Web site

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.

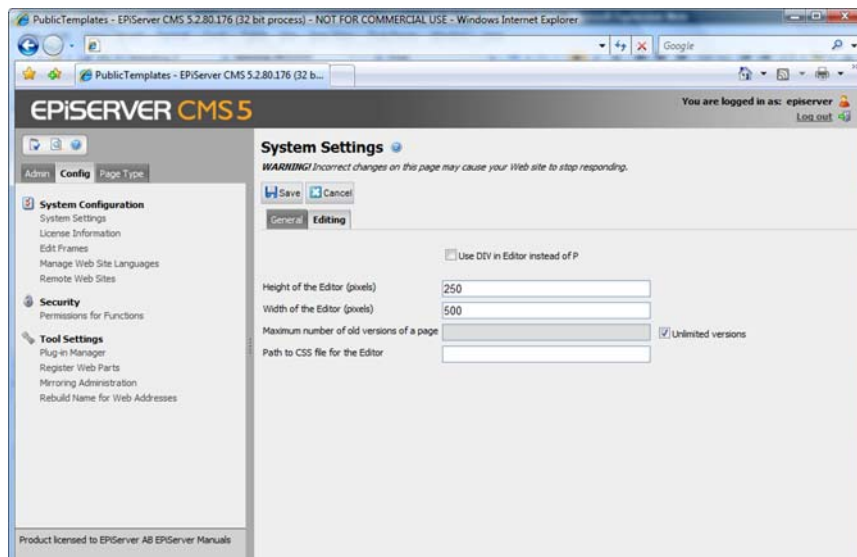
Enable globalization

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.

Editing



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.

Height of the Editor

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

Width of the Editor

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

Maximum number of old versions of a page

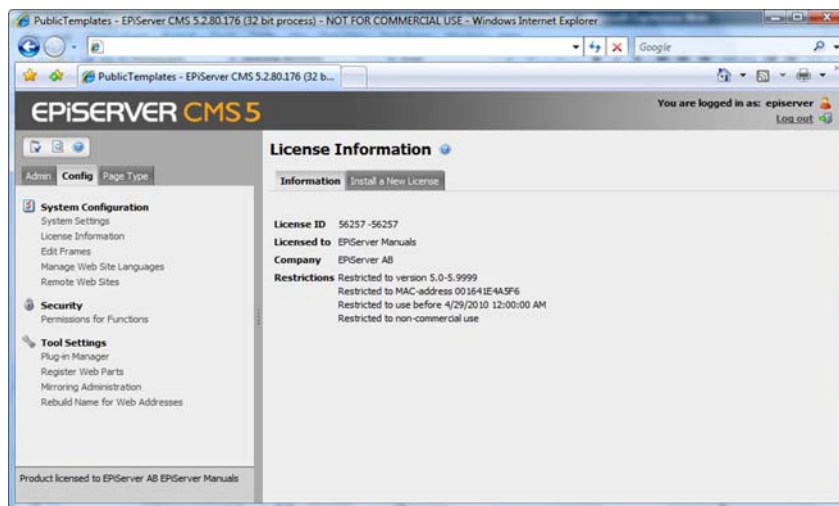
In this field, you can specify how many old versions of each page the editor can store. If you enter a value here, such as 3, three old versions will be stored in addition to the published version, i.e. four versions in all. When a fifth version is created, the first version of the stored page will disappear.

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 Web site, in order to make life easier for the editors. It is also possible to set other CSS's for different Editors on the Web site. This is a dynamic property that can be changed in Edit mode.

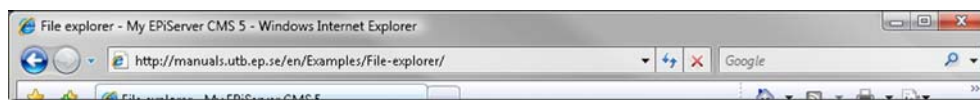
Licensing Information

The license for your EPiServer CMS is connected to an IP address or MAC address on the machine on which it is installed. The license information for your EPiServer CMS Web site is stored in an encrypted file on the Web server. This information can tell you to whom the license belongs, what type of license it is and if there are any relevant restrictions. If for any reason you need to change your license, for example if you change IP address or type of license or similar, you can upload and install the new license file from the *Install a new license* tab.



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 Web site based on EPiServer, 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.



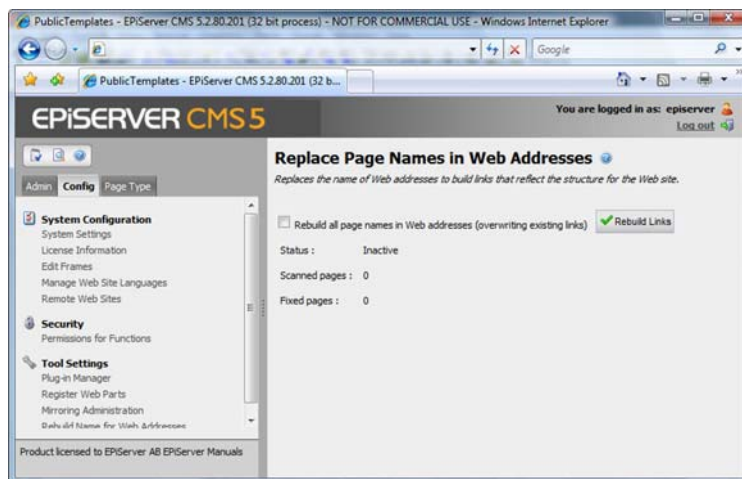
The names in the address field are created automatically from the name an editor has specified for the page. If an editor changes the name of the page, the name in the address field will not change. This can however be changed manually by changing the field *Page name in Web address* on *Advanced Information*.

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

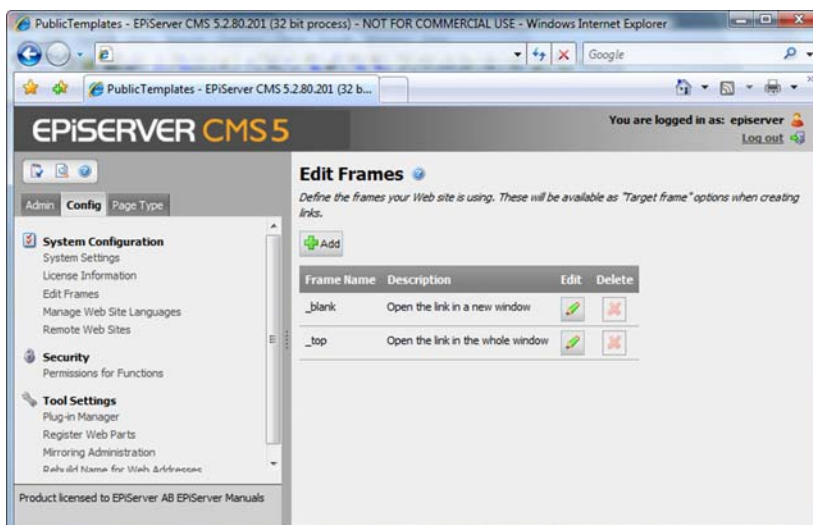
Rebuild names for Web addresses as follows:

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 Web site, 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 Web Sites

The remote Web sites function enables to you to retrieve information between the various EPiServer Web sites. This is used for load balanced Web sites and content mirroring, amongst other things. To be able to take information from one Web site to another, is a function that is often requested by customers with several Web sites on EPiServer CMS on several different machines, such as a public Web site and an intranet. Editors need only create information in one place, but it can be displayed in all.

With the *Remote Web Sites* function, you create the connections to the Web sites 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. Read more about this in the different Technotes.

Error Handling in EPiServer CMS

EPiServer CMS has functions for showing error messages to visitors and editors when there is a problem with the Web site. 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 Web site 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 Messages for Troubleshooting

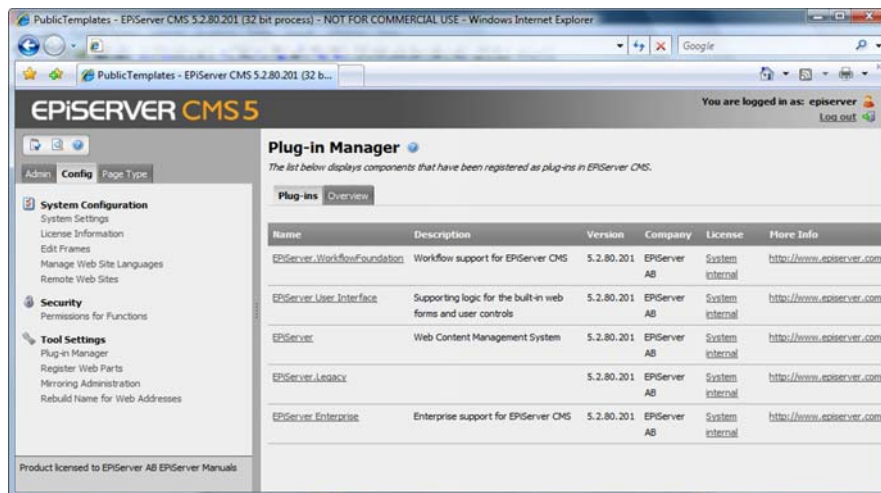
If you want most people to receive friendly error messages from EPiServer CMS, activate them for everyone. It can however be good for developers and administrators to see detailed error messages. By changing the permissions for this function, you can specify who should receive a detailed error message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

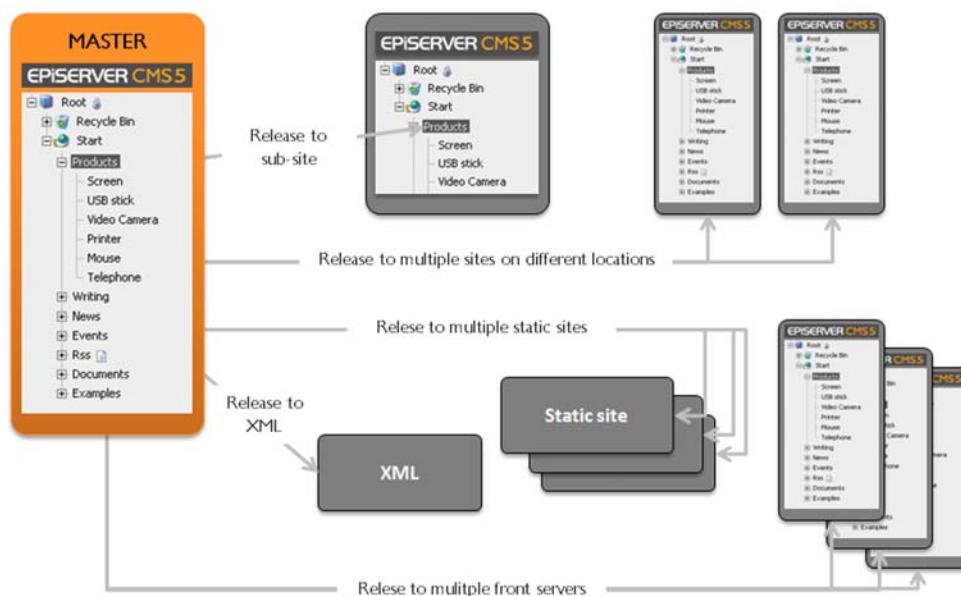
Plug-in Management

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.



Mirroring Administration

Mirroring is a function that can be used to mirror content from one Web site to another. EPiServer CMS can be used to mirror selected parts or all of a Web site. Mirroring can run automatically, or you can start it manually. You can also mirror from one EPiServer CMS to another EPiServer. 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. Another useful function is that you can create an HTML-based Web site using your dynamic EPiServer, which can be extremely useful for archiving.



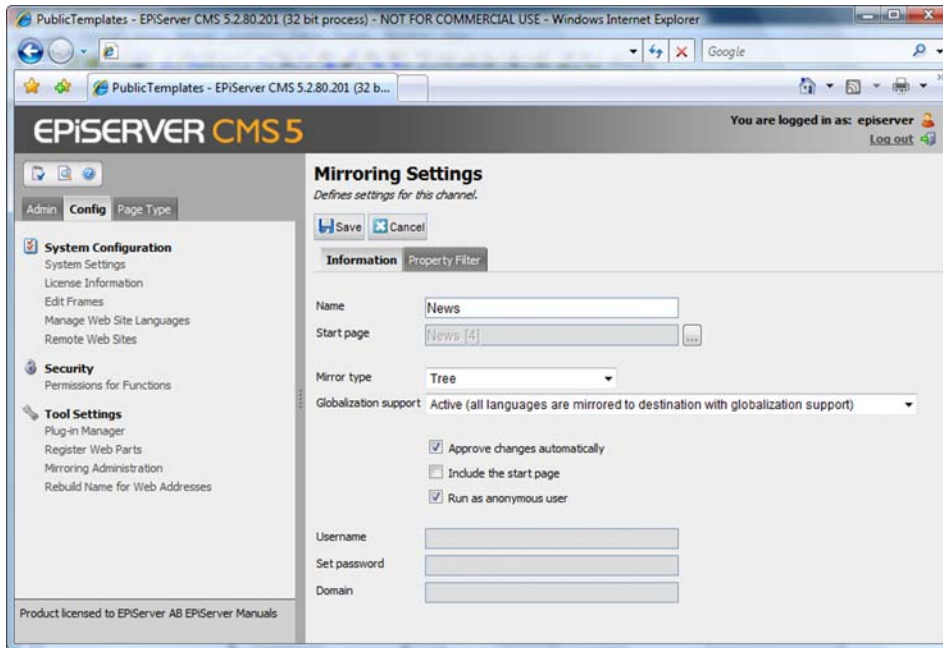
A mirror from one Web site to another is built up through channels on the Web site that transmits information. It is also possible to define several channels. Each channel consists of a number of settings for the pages that are to be mirrored and, of these, the types of page to be included. One of a channel's most important properties is the kind of a channel it is:

- **Tree** - This function finds all changes made in the tree. It finds pages that have been moved, erased, updated or created. The function finds all the changes from the root page down in the structure. This function is primarily used to mirror an exact tree structure. In this case, a page deleted on one Web site must also be deleted on the other.
- **List** - This function works like Tree, except it does not include changes below the first level.
- **Search** - This function finds changed and new pages. This function is useful when you wish to mirror news from one Web site to another.

To set up a mirroring from one Web site to another, channels are created in Admin mode. For the mirroring to work properly, you must change some settings in `web.config` as well as setting up the receiving Web site in *Remote Web sites*.

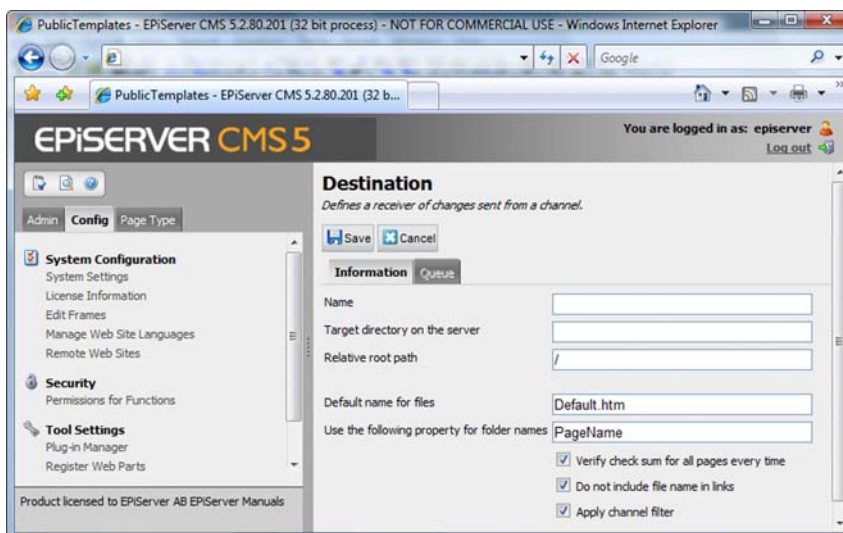
Create a channel in Admin mode as follows:

1. Select *Mirror Administration* on the *Config* tab.
2. Click *Create*.
3. Enter a name for the channel in the *Name* field.
4. Select where in the tree structure mirroring should occur, in the *Start page* field.
5. Select the type of mirroring in the *Mirror type* field. *Tree*, *List* or *Search*.
6. If you have a Web site where globalization is enabled, you must decide how you want the various language versions to be mirrored. Choose to mirror all languages or individual languages in the *Globalization support* list.
7. If you want the changes to be automatically approved, select the *Approve changes automatically* check box.
8. Select the *Include start page* check box if the changes probably also affect the start page.
9. Select either moving information as an anonymous user or specify a user name, password and domain.
10. Click *Save*.



When you have saved the settings for the channel, you must create the destination; that is, the Web site to receive the data transmitted.

11. Click *Create Destination*.
12. Select format for the destination: EPiServer, HTML or XML.
13. Fill in the information about which Web site and location in the structure the information is to be stored in.
14. Click *Save*.



The mirroring is 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.

Installing EPiServer CMS

EPiServer CMS is installed with the Installation Manager, which is encapsulated within a standard MSI-package. The program is downloaded and installed on the machine on which the Web site is to be stored. For further information about installing EPiServer CMS, see the Installation Instructions on EPiServer World.

Upgrade EPiServer CMS

On average one large version of EPiServer CMS is released each year. In between, there might be maintenance releases with smaller changes. All larger versions contain new functionality for different roles in EPiServer CMS. Sometimes there is a great focus on editors, sometimes on administrators and sometimes great focus on improving the technical solutions.

Each EPiServer CMS installation is upgraded manually. When purchasing your license for EPiServer CMS you also purchase Software Subscription. This means that you have access to all new versions of EPiServer CMS via the annual fee you pay for the agreement. Each organization decides if and when they should upgrade their tool. EPiServer AB recommends that the Web site is always upgraded in cooperation with the supplier who has developed the solution.

In slightly simplified terms, EPiServer CMS is divided into two different parts: A platform and a customized section. When installing EPiServer CMS you can choose to install only the actual platform or to include examples of a customized Web site. In broad terms, it can be said that the example Web site contains different functions for the visitor. There is also some content with examples of how to use the functions.

When upgrading EPiServer CMS, only the actual platform is upgraded. The customized part is not affected at all. In a new version of EPiServer CMS, the new example Web site may contain more functions. However, these are not installed automatically when upgrading. It is however possible to download and install them manually.

Index

A

Access Rights to Different Views	10
Access rights	9
Address of your Web Site	70
Administer Groups	12
Advanced Uploading of Files	37
Archive Function	59
Authorization to Publish from Microsoft Office	21
Authorizations	9
Automatic Emptying of the Recycle Bin	60
Available Page Types	40
Available in Edit Mode	40

C

Categories	50
Change Access Rights	16
Color Selector	64
Compulsory Fields	44
Convert Pages	47
Copy Page Types	47
Create Groups from Windows	13
Create Users in EPiServer CMS	14
Creating Groups in EPiServer CMS	12
Custom Property Types	49

D

Default Language	54
Default Values of Default Field	42
Default value of field	41
Delete Access Rights	17
Deleting	17
Deleting Unused Files	37
Deleting Users	21
Deleting a Group	13
Design of the Web Site	63
Display Options for Users	14

E

Edit Tabs	45
Editing Users	20
Editor Buttons	
Hide some	42
Editor Format Template List	63
Editor buttons	
All visible	43
Enable Globalization	54
Encrypting the web.config file	70
Error Handling in EPiServer	74
Export Data	65

F

Field Order	44
-------------	----

Fields that Vary	57
Files for EPiServer	62

G

Global Files	36
--------------	----

H

Handler for subscription updates	70
Height of the Editor	71
Hide Fields in Edit Mode	44

I

Image Editor	
Preset sizes	52
Importing Data	66
Installation	78
Integration with External Systems	67

L

Licensing Information	72
-----------------------	----

M

Manage Web Site Languages	55
Add Language	55
Delete language	56
Edit Language	56
Maximum number of old versions of a page	71
Mirroring Service	59
Modify Texts in Edit Mode	51

N

Name of the Web Site	70
----------------------	----

O

Overwrite Access Rights	17
-------------------------	----

P

Page Files	35
Page template	39
Page type	39
Create new	46
Edit	46
Path to CSS file for the Editor	72
Plug-in Management	75
Publishing Timed Versions	61

R

Rebuild Name for Web Addresses	72
Remote Web Sites	74

S

Scheduled Jobs	59
----------------	----

Search User/Group	20
Searchability in Fields	44
Setting Access Rights	15
Setting Authorization for Folders	36
Sort Index of Page Types	39
Starting Points in File Manager	34
Subscriptions	60
System Language	51
System Settings	70

T

The Window	7
------------------	---

U

Upgrade	78
Use DIV in Editor instead of P	71
User Groups in a Standard Installation	10

W

Web.config	11
Width of the Editor	71
Workflows	23
Administration	24
Automatic start tab	27
Create definition	25
Definitions	25
Delete definition	33
Editing definitions	33
General Tab	26
Running Instances Tab	29
Running instances	25
Security tab	27
Start Parameters Tab	29
System Information	25