Multisite with AxCMS.net® 8.6

Abstract / Summary

Multisite support used to be in AxCMS.net since many years. In version 8.6 it was extended with many useful features, which helps developers easier and faster create complex Enterprise Web applications and editors – to manage them. The most important features in the new version are:

- Extensible site properties
- Filtering content by site
- Cloning pages / Cloning sites
- Better site separation in management system

Business Case / Use Case

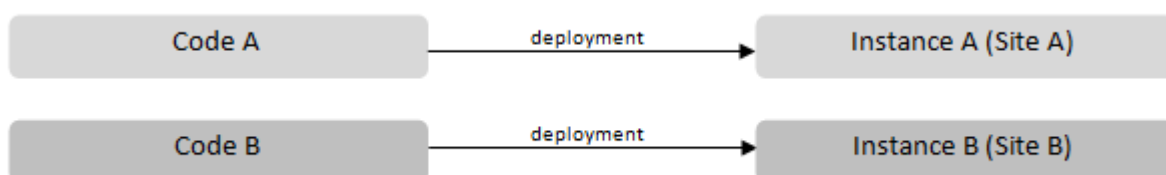In today's global world companies seldom maintain just "the web site". There are usually:

- Main company site
- Subsidiaries, partners, associations, other friendly companies
- Product site(s)
- Microsites for particular occasions (marketing campaigns, SEO actions, products, etc.)
- All listed above can exist many times – for every country and/or language

Depending on the use case more or less strict separation between the sites is needed. Some factors that play a role are:
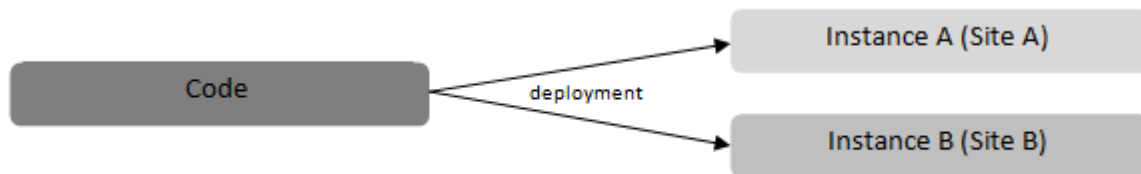
- Feature set – which features should be available on all sites?
- Layout – how close to each other / how different are the sites' layouts?
- Content sharing – do the sites share the same texts, pages, pictures, documents?
- Permissions – is it allowed for editors of different sites to see each others content?
- Users – are the same users editing content of different sites?

Deployment Scenarios

The strictest separation level is: there is a separate application for every site with its own code base (we will refer it here as **separate code**). In this case every site has its own application which has an own development cycle and gets deployed separately. You enjoy the highest separation possible, but do not get any advantages of a multisite application. For each application you need a separate AxCMS.net license (which you get free of charge upon request) and you will set up a separate AxCMS.net instance for every site.
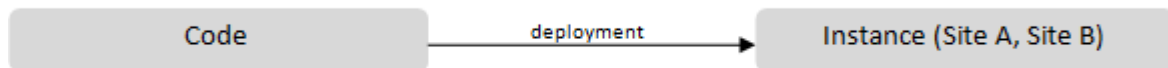
Next level is: there is a single code base, which is deployed to multiple AxCMS.net instances – each instance serves its own site (let's call it **separate instances**). It this model you have only one code base (one application) and one development/release cycle for all sites. After a new release you have to deploy it to every instance which you run (otherwise maintenance gets more and more difficult as the time goes, due to different application versions of the instances). Every instance has its own database and stores only the data (content, users, etc.) relevant for the sites hosted in this instance. For every instance you need a separate AxCMS.net license (which you get free of charge upon request).

In both scenarios (separate instances / separate code), if you still wish some content sharing, you can achieve it through communication between the AxCMS.net instances. There is an open XML format for AxCMS.net content ("Content XML") which can be exchanged via Web Services.

The most integrated scenario is **single instance**. As the name suggests you have one code base and a single AxCMS.net instance and you run multiple sites inside it. For this scenario you need only one AxCMS.net license. All the sites can share the same features, content, and user base. The rest of this white paper targets this single instance scenario, because it is the most appropriate one in most cases and delivers the most flexibility.

Of course it is possible to mix the scenarios – e.g. one code base deployed to two instances each hosting multiple sites.

Single Instance Scenario

Like IIS can host multiple websites in the same server instance, so AxCMS.net can host multiple sites (sometimes referred as AxSites) in the same IIS website.

You manage the sites simply through navigation (Edit / Navigation). The first level of the navigation is reserved per convention for sites. So first you create a site-node and under it you create the navigation tree for this site. In the versions before 8.6 this convention was not mandatory, so you could have changed it for a single-site solution. Starting with 8.6 you have to follow it. It means for a single site solution please create one site node representing the site itself and move the site navigation below it.

Site properties

If you edit the site node you can set many site properties (starting with 8.6; before 8.6 some custom ways were necessary to store the site parameters, like using ExternID property or additional categories):

- Site Name
- Site Country (country list according to ISO-3166 is provided; or none)

- Site Language (language list according to ISO-639-1 is provided; or none)
- Site URLs (Collection) – corresponds to the host headers in IIS
- Page naming convention for this site – a pattern like "en_*"
- Error pages used for: Not Found, Server Error, No Access, Login. If not specified, defaults from web.config are used
- Page handling configuration (error, redirect, switch, ignore)

Fig. 1 Site properties

So if you create a German site for your company *Company*, you can assign it to "Germany", set the language to "de" (German), and add hosts "company.de" and "www.company.de". As naming convention for the pages you can select a prefix ("de_*") or a suffix ("*_de") or maybe "*-de", if you prefer dashes over underscores[1]. You can customize the error pages for this site – at least they should provide error messages in German. For this create an error page as a usual AxCMS.net-page, e.g. "de_error" and assign it as a page for "Server error". Of course you can select to use a custom template for the error page.

Page handling configuration requires more attention. If a user accesses some page in your website, it is loaded and the site of this page is determined. To determine a site, the page's assignments to navigation are considered, but also the activeID parameter, if present. On the other hand, the URL requested contains also the server part, and this is compared with URL collections of particular sites. If

---

[1] If you wish your page URLs also to use dashes instead of underscores, you need to override the method responsible for URL-normalization (see AxCMS.net Developer Documentation)

the site implied by the page does not match with the URL provided (e.g. http://www.company.it/de_products.html), four options are available:

- Error – it will return 404-not-found error to the user and redirect to a custom not-found-page. The rationale behind this option is – the page requested belongs to a different site, and on the requested site there is no such a page (company.it does not contain "de_products").
- Redirect – it will redirect the user to the same page, using the first URL from the current site's URL collection (redirect to company.de/de_products occurs)
- Switch – current page will be changed to the matching page from the site matching the host header. For example, if user requests page "de_products" on site company.it, page "it_products" will be served.
- Ignore – simulates the old behaviour (AxCMS.net 8.5 and before) – it will not change the site and just deliver the page. This is also default behaviour in MS.

As a developer you can easily extend your site object with new properties.

Assigning Content to Sites

Any object (Page, Document, etc.) which you assign to navigation is automatically assigned to a site (remember – a site is defined with the 1st level of navigation). It means also, an object can be assigned to multiple sites (through multiple navigation nodes) or to no site at all (if no navigation is selected). Every overview pages features a site filter (starting with AxCMS.net 8.6), which allows selecting only objects from this particular site:

Fig. 2 Site filter

If a user adds a link in the content and selects a page as a link target and this page is assigned to more than one site, a particular site with its URL can be selected, allowing exact control over the link behavior. You can also leave it empty, meaning the link will be relative (use current host) – but if the page is from another site, but the current site has page handling set to "Error", you will have error 404 (not found).

BIND LINKS

You can enter your link directly or choose a **page**, **document** from the system. With the button 'Reset' on the left side you can delete your entries.

URL: Home.Sample

Pages: Select page...

Page URL: Home

Site URL: [▼]
                www.axinom.com
Navigation Item: www.axinom.de
                www.axinom.ee   tion item belonging to this page.

Target: ⦿ Current Window
                ○ Main Window
                ○ New Window

Track clicks?   ⦿ no   ○ yes

Fig. 3 Selecting a target site for a link

Permissions

If you run a multisite system with many editors it is very important to give everyone only those rights he/she really needs.

AxCMS.net uses Categories (i.e. also Navigation) to distribute permissions. So you can give users rights (like Edit or Publish) on particular navigation nodes. As Sites are defined as the 1st navigation level, you can easily manage user rights for the whole sites. So you can give someone Edit- and Publish-rights for a Site-node, making him virtually responsible for the content of this site.

AxCMS.net 8.6 goes one step further and introduces a **Read**-right. Without a read-right a user cannot see an object at all. Even more – such objects are not listed on the overview pages, preventing a user to discover the existence of such an object. It allows very clear separation of responsibilities.

As filtering out objects for a user without a read-right has some performance implications, we did our best to optimize this check and to move as much as possible into the database server.

Upgrading to 8.6 please consider the Read-right – ensure you give it to everybody who needs it. Previous AxCMS.net-versions run in the assumption, that every user has a read right on every object in Management System. For Live System the Read-right exists since the very beginning and is the base of the security model.

Cloning a Page

Only a few people would create new pages from scratch. Most of us prefer taking something as a template and change gradually until we are satisfied with the result.

Similar feature does exist in AxCMS.net for a long time – **Copy Content**. It works like this:

- Create a new page
- Fill out the properties, save the page
- Click "Copy content…", select the source page
- Content from the source page is copied into the new page

- Click "Edit Content"
- Edit the content of the new page

More natural seems to be **Save As**- approach, which is implemented in AxCMS.net 8.6:

- Select a source page
- Click "Clone"
- Specify cloning parameters
- Give a name to the new page
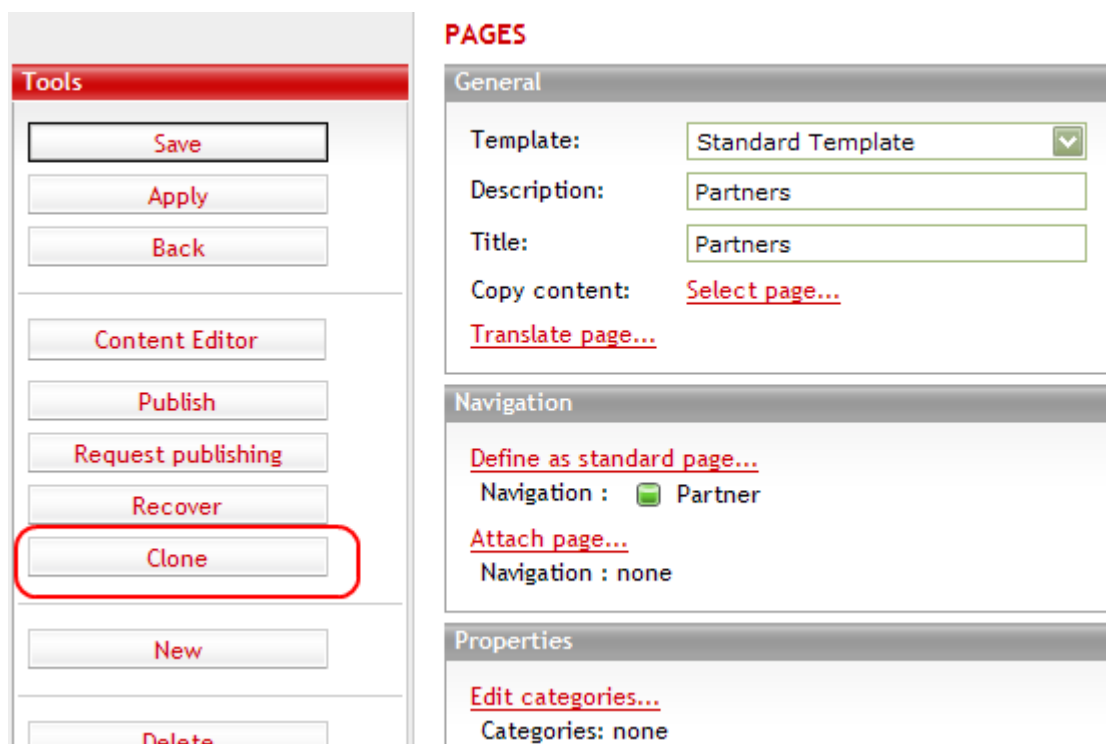- Page is copied and necessary parts are copied
- Edit new page

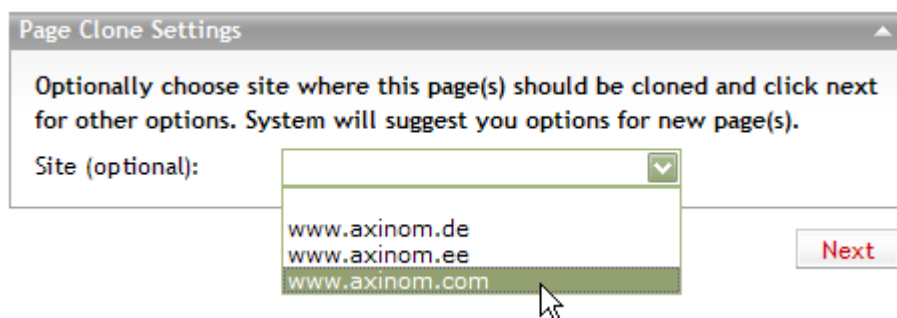*Fig.4 Cloning – Select a source page and click "Clone"*

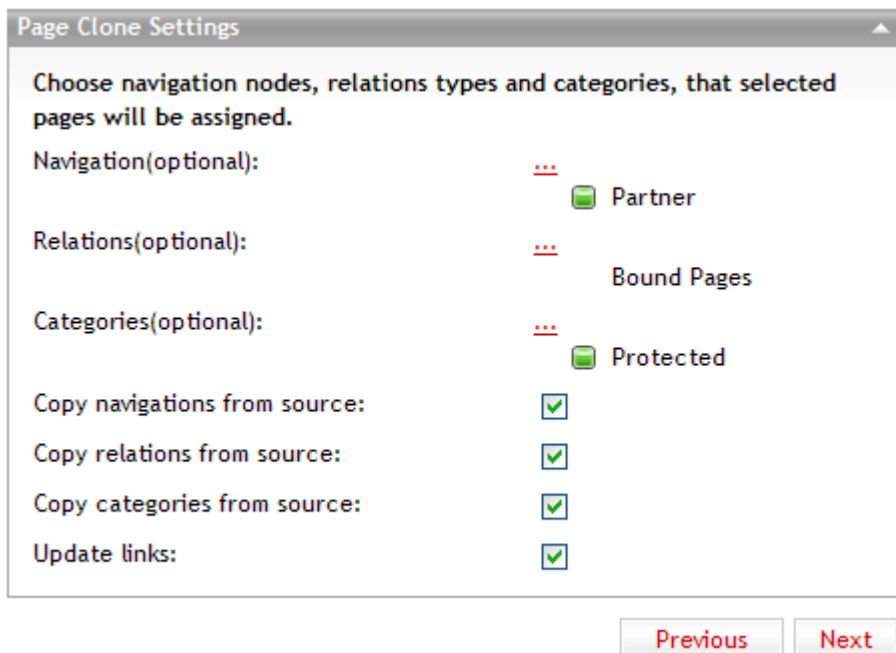*Fig. 5 Cloning – Select a new target site (optionally)*

Fig. 6 Cloning – Select clone settings. This cloning copies not only page content, but also page properties, assigned categories, relations (if selected), navigation (if selected). A clone is optionally related with its source
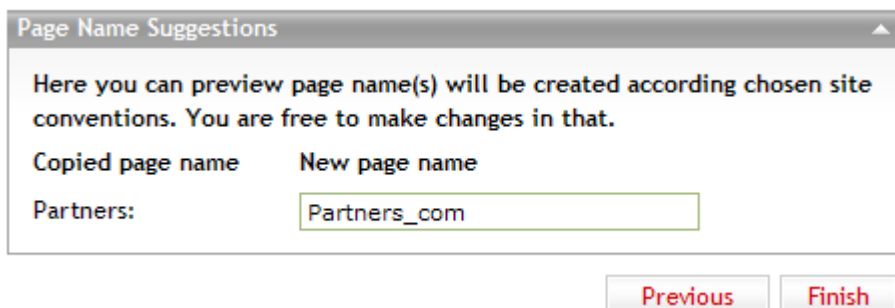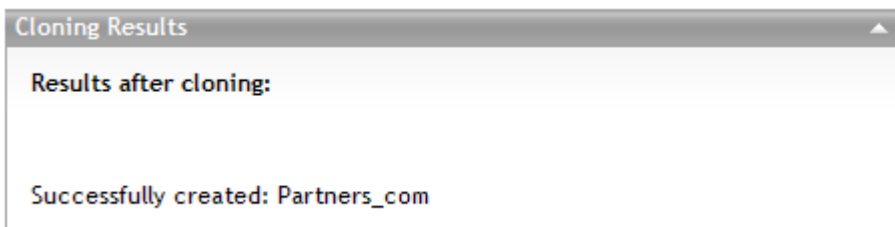

Fig. 7 Cloning – Select a new page name

Fig. 8 Cloning – a clone is created successfully

We describe this feature in this multisite-paper, because it makes it especially easy to copy pages between the sites. Cloning a page the first thing a user can do is to select a new site. If it is a different site, we speak about cloning to a different site.

If this is the case, AxCMS.net would suggest a name for the new page. For this AxCMS.net analyzes naming conventions specified to the source and destination site. E.g. "de_products" copied to an Italian site would suggest "it_products" as a new name.

With an optional setting it is possible to update links in the page content. Relative links which pointed to another pages inside the source site are updated to point to respective pages in the target site (naming conventions are used also here). For example, a link "/de_homepage.AxCMS" will be automatically updated to "/it_homepage.AxCMS" if you clone a German page to an Italian site.
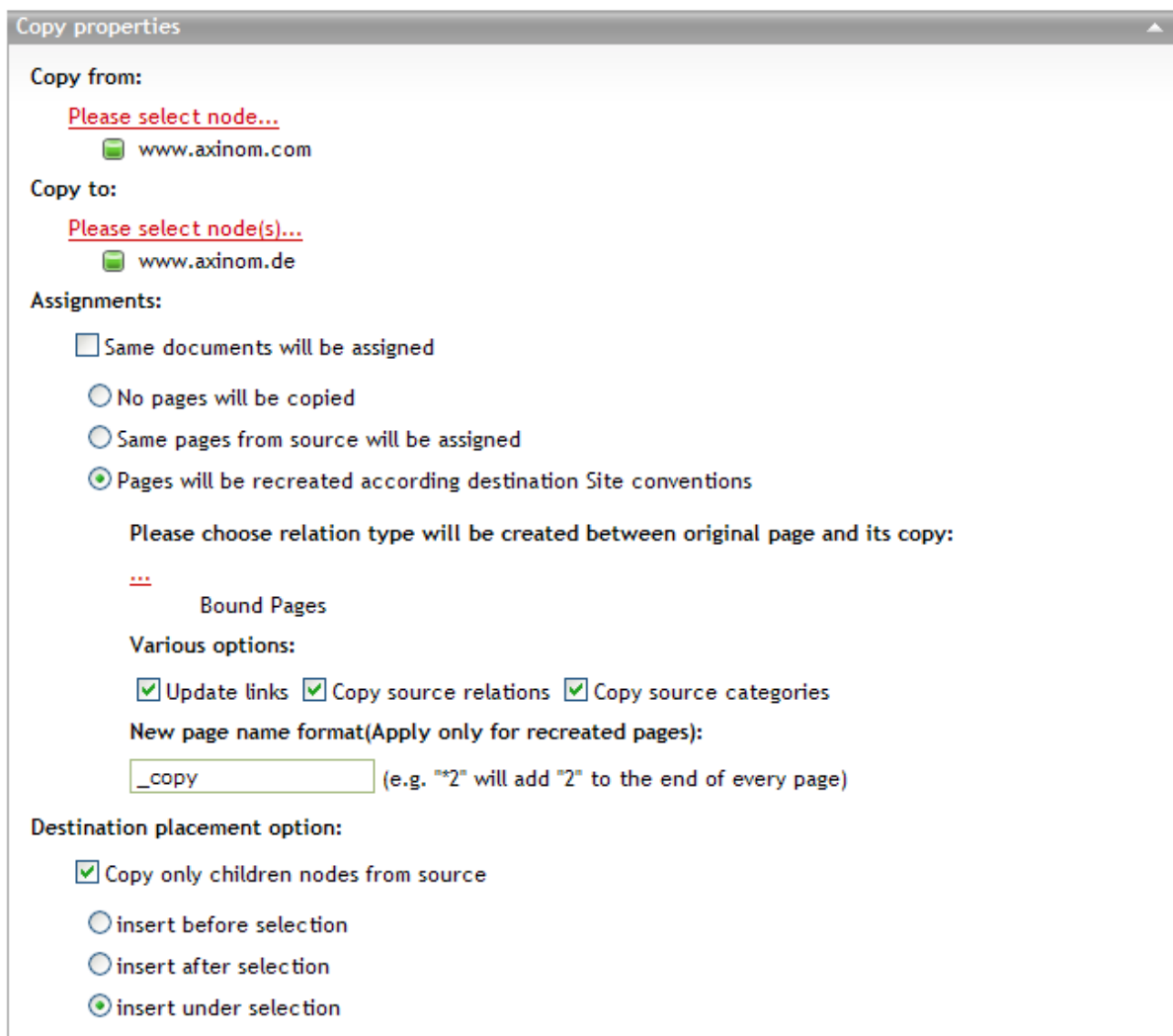
Cloning a Site

You can clone a page, but you can also clone a part of a navigation tree or even clone a whole site.
Even before 8.6 it was possible to copy navigation tree. Now you can let the system copy the whole sites, including attached pages, relations and assignments to documents. If you select to copy pages, they are copied (cloned) according to the rules described above under "Cloning".
If you copy inside one site, you can specify a pattern, how to name the cloned pages (e.g. "*_new" on a site with naming convention "*_de" would produce "products_new_de" out of "products_de".
There is still an option not to copy the pages, but link to the same pages. Assignments to the documents can be also copied.
And as a developer you can override the cloning logic to better fit your needs.

Fig. 9 Cloning a site (or generally – cloning a navigation subtree)

To copy the whole site, you should first create a new empty site and set its properties, then copy navigation, select the root node of the source and destination site and set a checkbox "Copy only children from the source" and radio button "insert under selection".

Axinom International Holding is the leading European technology integrator specializing in the Microsoft enterprise platform and Enterprise Content Management solutions. Axinom creates value for its customers by leveraging Microsoft enterprise technology to design, build, and deploy customized, reliable architectures and scalable infrastructures. With AxCMS.net® Axinom provides a completely .NET based Enterprise Content Management System completely free of license costs. AxCMS.net® won the "Microsoft .NET Server Innovation Cup„, is "Certified for Microsoft Windows Server 2003" and is the base for the worldwide AxCMS.net® Solution Provider Program.