

This is a **low resolution, black and white** version of the article you downloaded. To download the whole of Free Software Magazine in *high* resolution and color, please subscribe!

Subscriptions are free, and every subscriber receives our fantastic weekly newsletters — which are in fact fully edited articles about free software.

Please click here to subscribe:

<http://www.freesoftwaremagazine.com/subscribe>

Creating a managed website—Part 2

Focus on the message not the tools. Selecting a CMS, installing it and promoting your site.

Graham Oakes

Free software Content Management Systems (CMS) are capable of running most websites these days. Indeed, low initial costs and strong community-based support mean that many sites which can't afford a proprietary CMS can now benefit from the facilities a CMS provides. In the first part of this article I looked at how a CMS might help and what you need to do to define your site's target audience and structure. Now I'll get down to the nitty gritty of selecting a CMS, installing it and setting up and promoting your site.

How do I choose the CMS?

Does the CMS do what I want it to do?

Start by thinking about content creation. If you have a large team who collaborate extensively to create content, and then subject it to rigorous review before publication, then you probably need a fairly sophisticated CMS such as Plone (<http://www.plone.org>), Apache Lenya (<http://lenya.apache.org>), or Alfresco (<http://www.alfresco.org>). These are more complex to configure and learn, but they're more likely to support the workflows and collaboration tools you need.

On the other hand, if one or two of you are running a small site, then a simpler CMS may let you publish content without so many overheads. In this case, something like Joomla (<http://www.joomla.org/>), Mambo (<http://www.mamboserver.com/>), openCMS (<http://www.opencms.org/opencms/>

Consider the administrative interface. The Joomla CMS gives a simple console



en/), or Drupal (<http://drupal.org/>) might work for you.

While you're thinking about content creation, consider the interface for editors. Will they enter everything via a browser, or will some content be created in OpenOffice.org and then imported? If the latter, then ensure tools are available to do it. If the editors also control the formatting and presentation of the content on the site, then the CMS will need to support their formatting tools. If they simply enter plain text and let templates control the rest, make sure the CMS works well in this mode, and that your designers

What CMSs are available?

There are a wide range of CMSs available. Wikipedia (http://en.wikipedia.org/wiki/List_of_content_management_systems) lists about 130 free software CMSs. CMS Review (<http://www.cmsreview.com/>) lists more. CM Pros (<http://www.cmprofessionals.org/>), a community of content management professionals, is developing a database of CMSs (<https://www.cmprofessionals.org/resources/cmsml/comparator/>) and their features to help narrow down the search, but the best source of up-to-date information about any particular CMS is currently the website for that CMS.

have the right skills to define the templates and style sheets it needs.

Free software is supported by people who care about helping people use the software, rather than about how quickly they can get you off the phone

Think about other aspects of content delivery too. Do you want to publish to multiple websites, or in a format for mobile browsers? Do you want to support RSS? Do you want to generate features like breadcrumbs? Do you need to put embargoes on press releases or other time-sensitive material? Do you want extra security for some pages? Do you need site search? If so, make sure your CMS supports it.

What technical infrastructure does the CMS need?

The CMS will run on a server, and that server needs to sit somewhere. Your options are:

1. **Buy and host your own server.** This gives you complete flexibility, but it means you need a network link with sufficient bandwidth. You also have to manage all the hassle of running the server, dealing with network

outages, etc. If you want to focus on content rather than technology, then this may not be the way to go.

2. **Co-location.** You buy and install the server; your ISP provides space and power for it. This is marginally less complex than (1), but again only really worth considering if you like playing with hardware.
3. **Dedicated, hosted server.** Here the service provider supplies and sets up the server as well as the space. This is less flexible, as you'll be restricted to servers that they're happy to support, but it can be a good way to deliver a high-traffic site without too much technical pain. When choosing a service provider, be clear about what software they support (some will support the database and certain CMSs as well as the operating system). Check what service levels they offer (e.g. what bandwidth they'll provide; what out-of-hours support; what levels of resilience and security they have in their datacentre and networks; what reporting tools they offer). Finally, all of this comes at a price. If you want higher service levels, you'll pay for them.
4. **Shared, hosted server.** In this case you share space on server with other people (although you shouldn't be able to see each others' space if the server is set up correctly). Otherwise, this is similar to a dedicated

To manage Plone, you need to find your way through the mysteries of the ZMI (Zope Management Interface). This is more complex than Joomla, because Plone gives more advanced functionality. Fortunately most users don't need to see this complexity

Type Name	Size	Last Modified
ATCache		2005-11-02 18:26
MailHost		2005-11-02 18:26
Members (Members)	1 KB	2005-11-02 19:32
AMCache		2005-11-02 18:26
act_users (Group-aware User Folder)		2005-11-02 18:26
archetype_tool		2005-11-02 18:26
caching_policy_manager		2005-11-02 18:26
company-repository (Company Repository)	1 KB	2005-12-06 12:19
content_type_registry		2005-11-02 18:26
cookie_authentication		2005-11-02 18:26
documentation (Documentation)	1 KB	2005-12-12 17:56
error_log		2005-11-02 18:26
events (Events)	1 KB	2005-11-02 19:32
find-page (OFB Internet Home)	1 KB	2006-02-25 17:53
kupu_library_tool (Kupu visual editor)		2005-11-02 19:30
immetypes_registry (IMRE types recognized by Plone)		2005-11-02 18:26
news (News)	1 KB	2005-11-02 19:32
open-source-academy-portal (Open Source Academy Portal)	1 KB	2004-01-09 14:17
sia (SIA)	1 KB	2005-12-06 13:31
permissions-test (Permissions Test)	1 KB	2005-12-06 17:03
plone_utils (Various utility methods)		2005-11-02 18:26
portal_actions (Associates actions with icons)		2005-11-02 18:26
portal_actions (Contains custom tabs and buttons)		2005-11-02 19:47
portal_calendar (Controls how events are shown)		2005-11-28 13:20
portal_catalog (Indexes all content in the site)		2005-11-02 18:26
portal_controlpanel (Control Panel)		2005-11-02 18:26
portal_css (CSS Registry)		2005-11-02 18:26

server—slightly less flexible and with lower performance, but a good way to run a low-traffic site. It's generally a lot cheaper than a dedicated server.

If you want to stick with your current ISP, check what operating systems and applications they support. You may still have a good choice of CMS.

One final technical consideration: if you want to integrate the CMS with other applications, or if you want to customise it, check it's written in a language that you can work with. The most common languages are PHP, Java and Perl. As Rick Brenner (<http://www.chacocanyon.com/>) observes: "Most CMSs have a limited dynamic range. They make some modifications easy, but you still need technical interventions from time to time". If you're not happy to work within your CMS's capabilities, make sure you have the skills to extend them. (Of course, one of the benefits of free software is that such extension is at least possible.)

What support is available?

That brings us onto support. If it's really important to have someone to beat up when there's a problem, then there are many very good proprietary CMSs. Free software is supported by people who care about helping people use the software, rather than about how quickly they can get you off the phone. If that's the type of support you want, check out the community behind the CMS. How rapidly do questions get answered on the discussion boards? Does there seem to be a regular pattern of releases, with multiple developers contributing to each new version? Look for signs of a sizeable and active community.

How do I set up the CMS?

Once you've identified the CMS, you can start to set it up. The specific steps to do this differ according to the CMS, but the general stages will be:

1. **Set up an account with your hosting provider.** Depending on how you've chosen to host the CMS, you need to set up your account and ensure you have all the necessary passwords to access your server.
2. **Set up the server.** You need to install the server that will run the CMS. (For a site that's handling very large

traffic volumes, you may use separate servers for database and CMS. For such a site, you probably have a dedicated technical team...) You also need to install and configure the operating system, and set up network connections, firewalls, etc.

3. **Install any software needed to run the CMS.** Most CMS need supporting software. For example, Joomla needs the Apache webserver (<http://httpd.apache.org/>), MySQL database (<http://www.mysql.com/>) and PHP script processor (<http://www.php.net/>). Plone needs the Python language processor (<http://www.python.org/>) and Zope application server (<http://www.zope.org/>). Read the installation instructions on your CMS's website to find out exactly what you need. (If you have trouble understanding the instructions, that's a sign that either this isn't the right CMS for you, or that you should use a hosting service with the CMS pre-installed.)
4. **Install and configure the CMS.** Download the CMS and install it, as per the instructions on the website. You may need to configure a number of parameters such as port assignments, directory settings and site names as you do this. This may mean contacting your hosting provider to get information or to set up the necessary structures.

A dedicated server is a good option for a high-traffic site. A shared server is cheaper and good enough for a low-traffic site. Be clear about the software the service provider supports and the service levels they offer

If this seems like a lot of work, that's because it is. On the positive side, it's a one-off effort. Once the CMS is installed, you can make site changes (or at least ones the CMS supports) quickly and easily. For a site that doesn't change very often, the initial effort may not be worth it. For a dynamic site with lots of fast-breaking content, the long term benefits of investing some effort upfront can be substantial.

Buying a full hosted service can remove a lot of this initial hassle. You still need to set up and configure your site,

but the more technical stuff of setting up servers and installing software is done before you begin. Again, you'll need to pay for this service.

How do I set up my site?

Now the CMS is installed, you can set up your site. The stages will look something like:

1. **Create templates to deliver the site design.** The pages of your site will probably have common design elements (header and footer, navigation bars, etc) plus space to display the text, images and other content elements. Templates tell the CMS how to assemble all these elements into pages. You'll also need to define the hierarchy that pulls these pages together into a coherent website.
2. **Create forms and workflows for content entry.** You may need to define the forms that editors will use to enter content, and the workflows they'll use to publish this content. (These may be predefined within the CMS. Most CMSs assume you will want a lot of flexibility in the way you present the content to the world, but that your internal editors will be prepared to invest effort in learning the CMS's ways. If this isn't true for you, then allocate time for this stage.)
3. **Set up user accounts and other administration.** The CMS will need accounts for editors and other such users. It's also a good idea to set up processes like backups at this point, so that you can recover everything if there are glitches while you set up the rest of the site.
4. **Migrate existing content.** You can now migrate content from your existing site into the CMS. There may be automated tools for this but in many cases, especially for smaller sites, it's easier to do it manually. Be pragmatic here: if no-one is viewing the content on the current site, is it suddenly going to become more interesting on the new site? If not, don't invest effort in migrating it.
5. **Create new content.** While you're migrating content, you can also start to create new content. It's important that the site looks full of interesting goodies—nothing turns away people more than an empty-looking site. So make sure you've got enough to say to justify the size

of the site. And remember, this is just the beginning of an ongoing process of content creation.

6. **Launch.** When the site is ready, you can launch it. In the simplest case, this just involves switching your URL to point at the new site. You'll probably need to work with your hosting provider to do this. If you have more complex editorial workflows, then you may have set up a staging area (an area where editors can view content before it goes live, so they can see what it looks like to external browsers). In this case, your launch may also entail moving content from the staging area to the live site.

Promoting the site

Your site is up. You've had the launch party. Don't leave it there—you need to promote the site and keep it fresh so that people will keep coming to it.

"You've made a significant investment in your website. That investment will only bring results if people visit it. Ensure the site is prominent on search engines such as Google, Yahoo and MSN. Consider using other internet marketing channels such as pay per click, email, pod casts and on-line communities to extend your reach." Jo Green, Pazang (<http://www.pazang.co.uk>)

The key is: promotion only works if there's something to promote. A campaign might drive a one-off surge in traffic, but if your site doesn't say anything interesting then people will leave quickly, and they won't come back. Before you invest in promotions, invest in creating valuable and regularly updated content. Once you've got content, think about things like:

- **Search engine optimisation (SEO).** There are plenty of people who sell ways to improve your search engine rankings. Be wary. Search engines try very hard to work around any tricks you might use to bump up your ranking. You might gain a temporary advantage but it's unlikely to last. What should you do? Focus on substance: if you have something distinctive to say, other people will link to it and search engines will rank it accordingly. At the same time, don't make it hard for the search engines. Create meaningful URLs

for key pages and include sensible keywords. Register your site with search engines using their (free) facilities, e.g. at Google (<http://www.google.com/addurl/?continue=/addurl>).

“In the past, search engines didn’t track dynamic pages, for a number of technical reasons. Now they do some, but friendly URLs can help. (They’re also easier for people to use!) Create topical pages that aren’t dynamic, use essential keywords on those pages, and then link to local dynamic pages. Then at least the topical pages are likely to be seen.” Nynke Fokma, Moebius (<http://www.moebius.nl>)

For a dynamic site with lots of fast-breaking content, the long term benefits of investing some effort upfront can be substantial

- **Linking to (and from) other sites.** Link trading (networks of sites that exchange links with each other to increase search engine rankings) is like SEO. Be wary. Think of it this way: if someone comes to your site from a totally irrelevant site, how likely are they to be interested in what you’re saying? Invest your energy in building relationships with people who are interested in something that’s relevant to you. When you respect someone, link to their site. Ask them to reciprocate. People (and search engines) will follow these links to see what you’ve got to say.
- **Other media.** Don’t ignore other media. When you send emails, include your URL so people can easily come to your site. Include your URL in any advertising you may be doing. Include it in your letterhead. (Some people still spend time offline.) These things build off each other. Rick Brenner (<http://www.chacocanyon.com/>) notes “there’s a lot of synergy across channels: if people find your site in a search engine and also see you mentioned in a magazine, then that makes you seem a lot more credible.”
- **Measure and refine.** Monitor who is coming to your site and where they’re coming from. Note which content gets the most viewers. (Your site reports should

help here. Make sure your CMS supports the reports you need.) Use this information to refine your content and promotion strategy.

“A client of mine compared a website to a shop: you have to re-arrange the window every once in a while.” Willem van den Ende, Living Software BV (<http://www.willemvandenende.com/>)

Conclusion

Setting up a CMS isn’t easy. For a complex site, it can take months of work to clarify requirements, choose a CMS, and set up the site. Even for a small organisation with a simple site, this can take several days of work. But the core message is clear: sites with interesting and regularly updated content are the ones which attract readers. If this matters to you, then a CMS could make the whole content creation process a lot easier. In that case, your initial investment will be repaid well.

Over the last few years, free software, with its reduced licence costs and active support community, has significantly reduced the size of that initial investment. CMSs have become a viable option for even very small sites.

Copyright information

Copyright © 2006 Graham Oakes

This article is made available under the “Attribution” Creative Commons License 2.5 available from <http://creativecommons.org/licenses/by/2.5/>.

About the author

Graham Oakes: Dr Graham Oakes is the principal of Graham Oakes Ltd, a consultancy formed in 2003 to help organisations untangle the complexity within their systems, processes and governance. He helps define business and technology strategies that people will adopt, and then to assure implementation of those strategies. He can be contacted via his web site (<http://www.grahamoakes.co.uk>).