Open Source Web Content Management Systems in a Library Environment

David Gwynn

LIS 631 – Tim Bucknall

7 December 2009

TABLE OF CONTENTS:

INTRODUCTION **1**
CMS BASICS **1**
CONSIDERATIONS WHEN CHOOSING AN OPEN SOURCE CMS **3**
Usabilty/Functionality 3
Cost 4
Platform 5
Support Resources 5
OPEN SOURCE OPTIONS **6**
Drupal 6
Joomla 8
Wordpress 10
CONCLUSION: IMPACT ON LIBRARIES AND ADDITIONAL CONSIDERATIONS **12**
REFERENCES **14**

INTRODUCTION

In recent years, website content management systems (CMS) have become a popular option for producing, organizing, and maintaining online content for businesses, institutions, and even individuals. Libraries have been enthusiastic adopters of the CMS content model, employing open source and commercial products in an effort to freshen up their websites and integrate Web 2.0 features.

Both commercial and open source CMS packages are available. Microsoft's Sharepoint is a widely-used commercial product, while Drupal, Joomla, and Wordpress are three popular open source packages. The term "open source" refers to software code that is often provided free of charge and can also be freely and legally edited and altered by any user. Open source software in libraries has sometimes been a controversial topic; Stephen Abram of SirsiDynix recently published a position paper that was seen by many open source disciples as an indictment of open source in general, although he specifically targeted only open source integrated library systems (ILS) and even admitted that a few open source projects, including the Linux operating system and Apache web server had been very successful (Abram, 2009). Open source has numerous supporters within the practitioner community, however.

This paper is primarily concerned with the open source CMS option, and will discuss considerations for choosing an open source CMS package, some of the options available, and the potential impact on libraries, with a particular focus on Drupal, Joomla, and Wordpress, three of the most popular open source packages.

CMS BASICS

CMS-driven sites are distinct from traditional "static" HTML-based sites in that there are no actual "pages" that reside on a server as HTML files. Instead, there is a collection of templates that format the content of specific database fields into a display that is functionally equivalent to a traditional HTML-based web page.

Therefore, CMS content is referred to as "dynamic" content, because the "page" display is assembled by the CMS anytime the end user requests that particular "page".

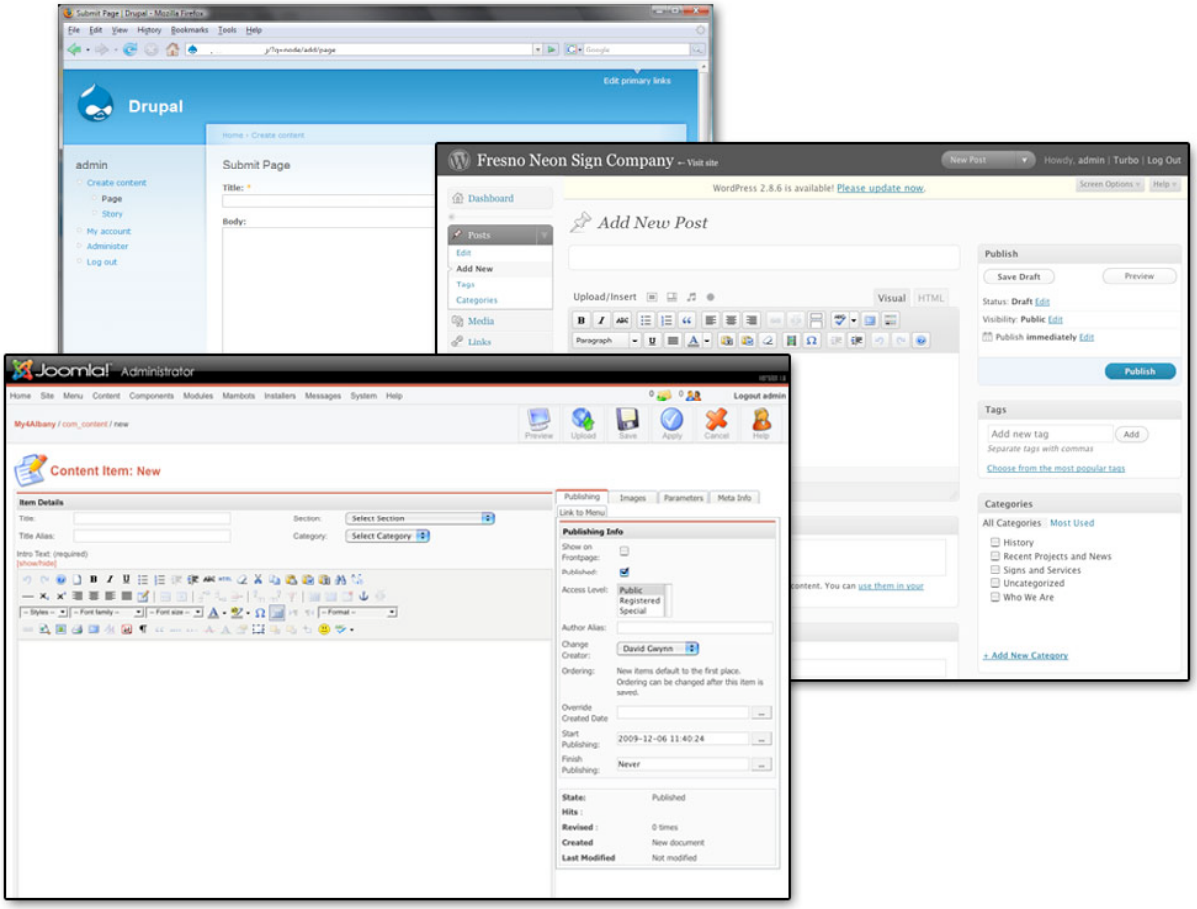

Figure 1: CMS Interfaces

Within a CMS, sites are maintained and content is generated through the administrative interface of the CMS (see Figure 1), which is a series of web forms in which content creators enter new content items (variously called "posts", "articles", or any number of other terms, depending on the software) using an interface that resembles a word processor. Most packages offer some sort of user-based hierarchical workflow that can require content to be approved by an editor or administrator before being displayed live on the site. There are also varying levels of support for content categories or website "sections" and links to new content are generated automatically when the content is created or published. Tags and comments from

end users are supported in a variety of ways, and there may also be a means to integrate other types of content such as blogs, wikis, message boards, etc.

There are several important benefits for generating a site—particularly a large and complex one—in this manner:

1. The process separates content from formatting, allowing content to be recycled and reformatted with minimal effort for additional applications, and also making site redesigns easier.
2. CMS facilitates a multi-user environment. Any number of users can be working on site content simultaneously.
3. CMS frees content creators from having to understand HTML coding and from having to update (and upload) multiple pages in the site manually every time new content is added. This facilitates more frequent content updates, at least in theory, as well as workflow management.
4. Integration of RSS feeds, Web 2.0 applications, and other dynamic content are built into most CMS packages, or are available through plug-in modules.

CONSIDERATIONS WHEN CHOOSING AN OPEN SOURCE CMS

Primary considerations for choosing an open source CMS include usability, cost, operating system/platform, and availability of support resources.

Usabilty/Functionality

A CMS is only useful if it is well-suited to the needs of the organization using it. Jonathan Blackburn, who led Florida State University's transition to the Drupal CMS, cites flexibility, planning, and communication at all levels as essential aspects as lessons learned during the process and warns that if the product does not meet the organization needs, don't use it, no matter how 'cool' it is." (Houlroyd, 2009, par. 6). In the same presentation, Karen Coombs of the blog *Library Web Chic*

(http://www.librarywebchic.net/), cited simplicity, user skillsets, levels of responsibility for content creation, and functionality as key selection criteria. Powel and Gill (2003) also cite planning and communication for content preparation at all levels of the organization as essentials when choosing a CMS. It is essential to consider who will be producing the content over time and to choose an appropriate based on that planning. If more "tech savvy" individuals will be the ones producing content and using the system, a more complex interface with increased options may be appropriate, while the less technologically inclined will need a simpler interface with fewer choices (or more intensive training). Therefore, planning must take into account the variety of users now and in the future. Planning should also consider the likely workflow needs of the site in order to determine the level of moderation required.

Cost

While many open source CMS packages are free downloads, there may still be considerable expense involved—what has been called the total cost of ownership (TCO) for open source software. Coombs has noted that while set-up of free applications may be quick and simple, the customization of those programs can take considerable time and effort. She further states that "Sometimes an effective commercial hosted product is a better solution" and that the relative costs are "judgment calls" (Abreu, 2009, par. 24). Kmetz (2006) notes that in spite of a mandate and preference for open source products, his campus ultimately opted for an ASP-based Microsoft system due to a solid vendor relationship and recommendation.

With open source CMS, cost is proportional to the expected level of customization and the technical skills of whomever manages the website. For most major CMS packages, there is also a large community of designers providing open source templates for use with those systems—many of them of higher "commercial grade" quality. Even with these, though, there are issues of implementation and customization. Some libraries have opted to bypass the process completely and outsource installation and site

development to commercial firms such as CraftySpace (http://www.yourlibrarysite.com/) that specialize in producing library websites using open source products such as Drupal.

Platform

Most of the major open source CMS products are deigned to operate in what has become known as the LAMP environment, named for its reliance on the Linux operating system, Apache web server, MySQL database engine, and PHP programming language—all of them free, open source products as well (Bisson, 2007). This environment is common among the commercial providers of website hosting used by some smaller libraries, but it may be problematic for some libraries, particularly those that are part of a larger institution that uses a Microsoft Windows server environment. Although there are comparable products for Windows servers (the WAMP environment), and Microsoft's website even states that Drupal and other PHP-based CMS can be installed (http://www.microsoft.com/video/en/us/details/d9478ec3-df6d-44cd-b102-f27bb44b65df), there may still be problems with Windows installations. There may also be less of a support base.

Support Resources

Some academic libraries have approached the CMS issue by designing their own systems, but a major drawback to this plan is the lack of any sort of support network. The active user community is one of the primary selling points of open source software in general, and message boards, wikis and listervs for open source CMS packages are numerous and active (see specific examples in the next section). Of course in many cases, this support assumes a fairly high level of technological knowledge as a point of entry.

Another concern related to support is the potential stability and longevity of an open source package. It is important to choose a product that will be around for a while,

and a solid support community base may be viewed as evidence of the product's ongoing viability.

OPEN SOURCE OPTIONS

In a 2009 post cited in several library-related publications, web developer Glen Stansberry cited Wordpress, Drupal, and Joomla as the ten most usable content management systems. Wordpress was cited for its simplicity and ease of use and, Drupal for its flexibility and extensibility, and Joomla for its large developer community and ease of installation. Stansberry's evaluation aside, these three packages are easily the most discussed open source packages, although there are numerous other options, as can be seen at the opensourceCMS website (http://php.opensourcecms.com/), which currently allows users to test the interfaces for hundreds of CMS, blogging, and message board packages. For purposes of this paper, though, the discussion will be limited to Drupal, Joomla, and Wordpress.

Drupal

Drupal (http://www.drupal.org/) is probably the most frequently discussed open source CMS, at least from a library perspective. Developed by Dries Buytaert beginning in 1998, the Drupal "community" numbered more than 350,000 ten years later, and the software has been downloaded millions of times (Drupal, 2009a). Commercial users of Drupal include newspapers (http://www.news-record.com/) and large corporate sites (http://warnerbrothersrecords.com/). Drupal has also penetrated the library market. In a 2008 issue of *Library Technology Reports* devoted to Drupal, the authors stated they had selected the package for numerous projects because of its Web 2.0 support, but also found that it worked well for both internal/intranet and public functions (Austin & Harris, 2008). Farkas (2008a) praises its add-on modules that integrate blogs, wikis, and other tolls that previously had been handled by separate applications. There have even been projects taking Drupal to a higher level, including attempts to integrate library catalog systems into the CMS;

one such project involved Millenium OPAC, a commercial product, and another incorporated the open source SOPAC (Graza, 2008; Sheehan, 2009). There is a Drupal user group dedicated to library users (http://groups.drupal.org/libraries) and a listserv for library users, hosted at the University of Illinois at Chicago (http://listserv.uic.edu/archives/drupal4lib.html)

Most criticisms of Drupal are based on its rather steep learning curve, its complex structure, and the fact that it may be more difficult to customize (Stephens, 2009). Page templates may require more work than with some other packages, and the system of data nodes, taxonomies, and blocks may be too abstract for some users. Drupal, by its own assessment, is best suited to a site that includes numerous different types of content, and even its developers warn that there a significant amount of effort may be required to understand it completely (Drupal, 2009b).

Figure 2: Drupal-generated Library Website (http://www.myrcpl.com/)

Numerous libraries use Drupal, including the New York Public Library (http://nypl.org/), Florida State University (http://www.lib.fsu.edu/), and the Stowe Free Library in Vermont (http://stowelibrary.org/). The latter also uses the open source Koha ILS. A list of additional libraries using Drupal can be found at http://groups.drupal.org/libraries/libraries.

Joomla

Like Drupal, Joomla (http://www.joomla.org/) is a full-featured CMS that can incorporate multiple types of content as well as multiple users and user hierarchies, but from a library standpoint, less has been written about Joomla. Evolved in 2005

from an earlier package called Mambo, Joomla works well with shared or commercial web hosting services, is relatively easy to install, and boasts of a large development community that has developed over 3200 for the product (Stansberry, 2009).

Joomla in Libraries, a library-specific Joomla support site (http://www.joomlainlibrary.com/) provides articles, a technical support forum, and links to resources for Joomla users in libraries. Mc Dermott (2008) discusses her public library's implementation of Joomla with resources and installation tips, while Sharma et al. (2009) present a case study and step-by-step description of a Joomla installation at an academic library in India; both found the software acceptable for their needs.

Figure 3: Joomla-generated Library Website (http://www.wcls.org/)

One caveat expressed by Stansberry and others is that the large Joomla development community is more apt to charge fees for its plug-ins and enhancements than developers for some other open source platforms, although free add-ons are still plentiful as well. In addition, the default installation of Joomla lacks some of Drupal's support for other content types, such as message boards and wikis, there are, however, third party plug-ins that can add some of this functionality.

Wordpress

There has been considerable discussion recently on the use of the very popular Wordpress (http://www. wordpress.org/) as a full-blown CMS. Originally created in 2003 as a simple blogging platform, Wordpress offers a significantly simpler codebase than Joomla or Drupal. It also features a simple and user-friendly interface and a modular structure that facilitates add-ons and "widgets" to customize and extend page content, including the integration of flickr albums, RSS feeds, chat, and other applications. Karen Coombs has been a big proponent of Wordpress as a content management system, speaking and writing numerous articles and blog posts on the subject in the past year, and also producing a step-by-step user guide for libraries using Wordpress as a CMS (2009). Coombs notes that use of the Wordpress page structure, combined with template design can easily produce a standard website (Houlroyd, 2009), and Farkas (2008b) stresses that with increasingly sophisticated templates, a Wordpress-generated site need not look like a blog.

Use of Wordpress as a complete CMS is increasing. The Northeast Kansas Library System uses it to create sites for member libraries as part of the My Kansas Library program (http://www.mykansaslibrary.org/). Many individual institutions have begin using Wordpress as well, including Stevens Memorial in Massachusetts (http://www.stevensmemlib.org/), Lake Washington Technical College near Seattle (http://lwtclearningcommons.com/), and the Park County Library System in Wyoming (http://parkcountylibrary.org/).

Figure 4: Wordpress-generated Library Website (http://lwtclearningcommons.com/)

Many libraries already use the Wordpress platform (or its free hosted version at Wordpress.com) for blogging, as do many others, so the support community is large and the interface familar. There is even an open source OPAC tool in development based on Wordpress, Scrblio (http://about.scriblio.net/), which is currently being used by Plymouth State Universty (http://library.plymouth.edu/) and the Cook Memorial library (http://www.tamworthlibrary.org/) among others. Scrblio integrates the OPAC and CMS fuctions of their websites. One user has also devised a scheme to use the software to create dynamically-generated subject guides (Dodson, 2008).

One issue with Wordpress, however, is that its the "famous 5- minute installation" promoted on its website produces only a default version of the program suitable for simple blogging. Upgrading to a more CMS-like version can requires a significant amount of effort researching and modifying appropriate page templates and plug-in modules, and may also require some technical prowess. Once this set-up is complete, however, the interface for everyday content creators remains nearly as simple as the standard blogging interface. Wordpress also currently lacks the extensibility of Drupal with respect to integration of other types of content such as wikis and message boards.

CONCLUSION: IMPACT ON LIBRARIES AND ADDITIONAL CONSIDERATIONS

Moving to a CMS, whether open source or commercial, can go a long way toward promoting new and fresh content on the library's website, this making the site (and hopefully the library itself) a more inviting destination. Using a CMS can make adding and managing content easier, and can also facilitate the addition and integration of Web 2.0 applications including chat, flickr albums, RSS news feeds, and social networking tools like facebook. What a CMS *cannot* do, however, is create actual original content. This is a purely human function that requires planning and dedication of resources, which is not an easy task for already over-extended library staffs. Any plan for implementing a CMS must also recognize that someone will need to research the articles, write the reviews, and compile the event listings that will make up the new library website.

Libraries must also engage in a process of ongoing evaluation and assessment. The library's physical collection grows and evolves over time, and its web presence must do the same; a good web site is never "finished." One of the most important aspects of the Web 2.0 phenomenon is the concept of "constant Beta"—the idea that web applications are constantly being tweaked and improved as the community demands. In this environment, it will not be enough to simply provide a library website with no provision for content updates nor interactivity. If the site is not providing what

patrons want (or is not doing so in a manner that allows them to *find* what they want), the site will probably not be used to its potential, no matter how attractive the design nor how advanced the back end may be.

That said, a website that is in "constant Beta" will never be perfect either. And that may not be a bad thing. Laura Cohen of the University of Albany, SUNY, recommends that in addition to using CMS, looking at commercial sites for inspiration, and involving the most creative people in web planning regardless of departmental affiliation, librarians should:

> Lighten up about your site. This may seem like an odd thing to say, but think about it. Sure, our sites are a branch library, a front doorway, our public face to the world. But we take our sites too seriously. *When we do that, major changes to the site become too important. When something becomes too important, positive change can get discussed to death, compromised, delayed, or nixed.* We've got to relax. Most of our users are comfortable with sites that undergo frequent change, are informal, and tolerate - if not feature - experimentation. Why should our sites be different? You can tell me that library sites are about serious business, and I'll reply that a hands-off, spiritless Web site does nothing to enhance use of the library. (2006, par. 8, emphasis added)

Enhancing the use of the library, and by extension its website, is the ultimate goal, after all. Again, a major benefit of CMS is that it separates content from foratting and design, and theoretically allows contributors to concentrate on their own area of expertise. Open source CMS will likely be an major starting point for increasing numbers of large and small libraries who want to move to the next generation of websites.

REFERENCES

Abram, S. (2009). Integrated library system platforms on open source. Retrieved 13 November 2009 from http://stephenslighthouse.sirsidynix.com/Open Source Position Paper 20091014 FINAL[1].pdf

Abreu, A. (2009). Q&A: Karen Coombs of Library Web Chic. Retrieved 4 December 2009 from http://www.libgig.com/karencoombsinterview

Austin, A. & Harris, C. (2008). Drupal in libraries. *Library Technology Reports 44*(4).

Bisson, C. (2007). Free at last: taking advantage of open source software. *American Libraries 38*(7), 44.

Cohen, L. (2006). Transforming our library web sites. Retrieved 29 November 2009 from http://liblogs.albany.edu/library20/2006/12/transforming_our_library_web_s.html

Coombs, K. (2009). *Wordpress as a content management system*. Retrieved 22 November 2009 from http://lepton.wils.wisc.edu/wordpress_cms.pdf

Dodson, J. (2008). WordPress as a content management system for a library web site: How to create a dynamically generated subject guide. *Code4Lib Journal 3*. Retrieved 29 November 2009 from http://journal.code4lib.org/articles/76

Drupal (2009a). *Drupal history as seen by Dries* Retrieved 22 November 2009 from http://drupal.org/node/297669

Drupal (2009b). *Is Drupal the right tool for the job?* Retrieved 22 November 2009 from http://drupal.org/node/346217

Farkas, M. (2008a). CMS for next-gen websites. *American Libraries 39*(10), 26.

_____ (2008b). Our new website is a blog. *American Libraries 39*(9), 45.

Garza, A. (2008). From OPAC to CMS: Drupal as an extensible library platform. Library Hi-Tech 27(2), 252-267.

Houlroyd, L. (2009). Content management systems in libraries: Opporunities and lessons learned. Retrieved 1 November 2009 from http://litablog.org/2009/07/content-management-systems-in-libraries-opportunities-and-lessons-learned/

Kmetz, T. & Bailey, R. (2006). Migrating a library's web site to a commercial CMS within a campus-wide implementation. *Library Hi Tech 24*(1), 102-114.

McDermott, I. (2008). Joomla! Looms: Can open source CMS save a library website? *Searcher 16*(5), 8-13.

Powel, W. & Gill, C. (2003). Web content management systems in higher education. *Educause Quarterly 26*(2), 43-50.

Sharma, D., Sirtaj, R., Ahluwalia, P., & Singh, V. (2009). *Web content management in universities using Joomla!: Freedom all together*. Paper presented at the 7th Annual Convention on Automation of Libraries in Education and Research Institutions (CALIBER), Pondichery, Tamil Nadu, India.

Sheehan, K. (2009). Creating open source conversation. *Computers in Libraries 29*(2), 8-11.

Stansberry, G. (2009). Top 19 most usable content management systems. Retrieved 29 November 2009 from http://net.tutsplus.com/articles/web-roundups/top-10-most-usable-content-management-systems/

Stephens, M. (2009). Drupal vs. WordPress — Which CMS is better for libraries? Retrieved 29 November 2009 from http://classes.tametheweb.com/kasiag/2009/04/12/drupal-vs-wordpress-which-cms-is-better-for-libraries/