**Subject Guides & More: Creatively Transforming an Open Source Management System**

This article describes the implementation of SubjectsPlus to manage the subject guides at the Wichita State University Libraries. The decision to implement an open source solution, the implementation process, and customizations to the software are discussed. In addition to the subject guides, SubjectsPlus is also used to manage course specific and miscellaneous topic guides, the library staff directory, and database links. The article also covers the reception of SubjectsPlus by the librarians and teaching faculty.

by Gemma Blackburn and Mary Walker

**Introduction**

During the Summer of 2010 the library at Wichita State University, a medium sized academic library, implemented the SubjectsPlus subject guide management system during a complete overhaul of the library’s web presence. SubjectsPlus is an open source content management system developed by Andrew Darby at Ithaca College, and is based on the Pirate Source system developed at East Carolina University [1]. SubjectsPlus was chosen over other systems in part because its open source nature allowed us to modify the system to best fit the needs of our library. SubjectsPlus is also able to manage other online library resources, including course specific and miscellaneous topic guides, the library staff directory, and database links. It has proven to be a valuable time-saver in all three areas, but most especially in the management of database links. Implementation of the system took approximately three months of part time work, with an additional two months for librarians to enter content and build the guides.

**A New System**

A new way of delivering subject guide material was long overdue. Librarians were maintaining subject web pages and paper subject guides. The Collection Development department kept track of the user names and passwords for those pages so that if a Librarian should leave, the next person could access and edit the pages. Staffing in Collection Development has been reduced and the responsibility of tracking the access information had become an added burden. The database listings, both alphabetic and subject-oriented, needed to be kept up-to-date but we needed a more efficient way of doing this. Since this was done manually, and database information changes frequently, it was incredibly time consuming for the Electronic Resources Librarian to stay on top of these updates.

Because each librarian has his or her own style and HTML/CSS skill level, the look of the subject pages and the type of information found on the pages varied greatly. In addition, some librarians were not comfortable adding multimedia content to their pages. The Engineering subject page, which was created in Adobe Flash by a graduate student, created a further complication. After the student left, the expertise and support files were no longer available, making updating the guide nearly impossible.

We also needed a better way to connect the electronic databases to their subject areas and a better way of maintaining the lists of database links on the library website. Previously, the library maintained two lists of links (an alphabetical list of database by title, and a list of database by subject area) in static HTML pages, as well as 49 subject guides with sections for primary and additional database listings. If a database was added, removed, or had updated information, the two lists required modification, along with all the pertinent subject pages.

We needed a better way to handle all of this information. We desired a solution that had a uniform appearance, was easy to maintain, and was discoverable by our students. We knew we wanted a system that would be more user-friendly...
for librarians, and be Web 2.0 compatible, while also bringing consistency to our current environment for a smoother transition for both the students and the librarians.

**SubjectsPlus**

Budget concerns at WSU made the cost of an open source solution more attractive than a commercial solution. The University Computing and Telecommunications department (where the library’s servers are maintained) already managed several LAMP (Linux Apache MySQL PHP) applications for the library, so additional hardware and expertise was not required for the MySQL/PHP driven SubjectsPlus. The open source *Library a la Carte* system from Oregon State University [2] was also considered, but was not chosen because there are no other Ruby on Rails applications currently used at WSU, and therefore no available expertise to install and maintain the system. The implementation of *Library a la Carte* would have taken more resources (especially time) to achieve the library’s goals. The popular LibGuides system [3] was also considered, but the cost of the system, and the additional addons that we would have required from Springshare, would be an additional budget concern for the library. We also found the appearance of LibGuides to be cluttered. Because visual customizations to this hosted system are limited it would be impossible to make the guides consistent with the rest of the library’s website.

The ease of using SubjectsPlus was beneficial to librarians with busy schedules and varying skill levels. Multimedia elements are especially easy to embed. Minimal training was required, and librarians felt confident in using SubjectsPlus with the majority of guides being finished within 90 days.

The biggest advantage of SubjectsPlus is the ease with which it can be extended to meet the needs of any library. Expanding SubjectsPlus is only limited by the expertise available at WSU. Also, the SubjectsPlus community, organized through a Google Group, has proven to be a helpful source of support and information in implementing and expanding the system [4].

**Database Access**

Before SubjectsPlus, the Electronic Resources Librarian maintained static web pages with links to all library databases, organized by title and by subject on the library website. Included were fifty databases by subject pages which displayed the database name, access type (off campus, open, library only, or limited access), a link to the database description page, the major subject area, the vendor (some with links to the Vendor’s site), and indicators for “full text”, “some full-text”, and/or a “link to the WSU catalog”.

The Electronic Resources Librarian maintained these pages using Microsoft Frontpage. Using the find and Replace feature to update URL changes or delete databases from the lists is not too difficult. However, care must be taken to update only relevant URLs found on the current database pages. Archive copies of the alphabetic list were made before major changes were implemented.

Adding a new database was more time consuming. Not only did the database and all related information need to be added to the alphabetical page, but it also needed to be added to each appropriate subject page either as a primary database or as an “Other Useful Database.” An email would be sent out to the Subject Librarians informing them that the database was now available and soliciting their input on what subject guides to include it on either as a primary or secondary resource. Updates would be made as responses were received.

Using SubjectsPlus has streamlined this process. Now we have only one record to add, delete, or modify and have reduced the number of pages requiring updates by at least 50%. Web of Science, a popular general database, appeared in 32 of the static HTML database pages, requiring 32 files to be updated if the information about this database changed. Now it is changed in only one place.

Adding a new database is done by filling out a form (Figure 1); knowledge of HTML is no longer necessary. Updating a record is as simple as logging in, retrieving the database record, making the necessary updates, and clicking on the ‘Save’ button. Deleting is just as easy; once the record is returned, just click the ‘Delete’ button.
When adding a record, an email is still sent by the Electronic Resources Librarian to the Subject Librarians but this is only to inform them that the record is now available for them to add to their own subject pages.

Subject Librarians add the resource to individual subject pages by using the newly created record. They associate their subject guide with that record by selecting their subject from the list of subjects which allows the record to display on their subject page. They also select a source type to define that record in their guide (choosing either Databases or Databases – Supplemental), and can also choose to override the default description of the database with their own.

The benefit of entering, updating, or removing information from one place was long overdue. A process that took 45 minutes per record to add, now only takes about twenty. Updating takes only a few minutes and deleting takes seconds.

The main downside to using SubjectsPlus instead of a static website is that archiving older versions of our database listings for historical or practical purposes is not a built-in feature. However, it is possible to keep expired records but not have them display publicly.

**Customizing SubjectsPlus**

While SubjectsPlus will work fine out-of-the-box, its openness allows libraries to build upon and customize the system as much or as little as desired. Some alterations made to SubjectsPlus at WSU were based on the need of the library, and others were intended to add value to already existing library services by integrating them into the subject guides. Some of the customizations were straightforward to implement and are covered in depth in the SubjectsPlus documentation or had previously been implemented by other SubjectsPlus libraries, and some modifications required changes to the database structure and additional PHP files.

**Adding Pluslets**

The guides within SubjectsPlus are built by dragging and dropping boxes, called pluslets, onto a guide canvas (Figure 2). There are numerous Pluslets readily available in SubjectsPlus, including Twitter, Flickr, del.icio.us, and generic RSS feed displays, a base pluslet containing the FCKeditor for custom web content, and many others.
Because of the open source nature of SubjectsPlus there are limitless possibilities for additional pluslets. The SubjectsPlus wiki [5] provides an example of an added pluslet to embed a Meebo chat box along with instructions to create it. This pluslet allows a subject librarian to add live chat to any guide, giving point-of-need service to students. The top priority custom pluslet for the library was an online catalog search box. The catalog search and the Meebo box pluslets were both achieved by creating a new PHP file incorporating static HTML, and linking the new pluslet to the guide editor so it becomes available to drop onto the canvas. New pluslets are made available in the guide editor by adding them to the get_pluslets.php file.

Another custom pluslet created was a search box for the Libraries' federated search engine. This concept proved to be more complex than the catalog search box as we desired more than a simple static form. The WSU libraries use the Serials Solutions 360 Search federated search engine, which was locally branded Harvester Article Finder. Harvester gives patrons the option to search most of our databases (104 of our 110 databases) all at one time, to search a specific subject grouping of our databases as defined by our subject librarians, or to select specific databases to cross-search. In order to make the Harvester search box pluslet as relevant to each subject guide as possible, it was preferred that each guide would contain a box to search only those databases relevant to that guide. For example, the Harvester search box on the Mathematics & Statistics subject guide would search only those databases defined as relevant to Mathematics & Statistics. The subject-specific Harvester search boxes can be created using the 360 Search catID – a five digit code used to define each subject group.

In order to create the Harvester search box pluslet, first we added a new field (harvester_id) to the SubjectsPlus database to hold the catID. Then we created a new pluslet PHP file to build the search box and make an SQL call to the database to retrieve the correct catID for the appropriate subject guide. The final step was to add input fields to the staff interface for the Creating New Guide form (Figure 3). Now the librarian may choose to add the Harvester catID to each guide as it is created in the Create New Guide form. A lookup screen with each subject and the corresponding catID is linked to the form for reference. Once the Harvester pluslet is added to the guide it will use the selected catID to build the subject-specific search box (Figure 4).

The Harvester Pluslet:
<?php

// Harvester info
// Grabs 360 Search catID and throws it into search box

global $proxyURL;
global $check_this;

if (isset($_GET['subject_id'])) {
    if (is_int($_GET['subject_id'])) {
        $subject_id = $_GET['subject_id'];
    } else {
        $subject_id = 1;
    }
} elseif (isset($check_this)) {
    $q1 = "SELECT harvester_id FROM subject WHERE shortform = " . $check_this . ";";
    $r1 = mysql_query($q1);
    $subject_id = mysql_fetch_row($r1);
    $subject_id = $subject_id[0];
} else {
    $pluslet .= "<p>You will see results when you return to this guide later.</p>";
    return;
}

<input type="text" name="term" id="articleBox" size="28" value="Enter search ...
</form>";

$pluslet .= $harvester;
?>

Figure 3. Create New Guide Form with Harvester ID Field

Figure 4. The Harvester Search Box
WSU uses several user-built extensions for the Voyager ILS, including the New Books List feature created by Michael Doran at the University of Texas at Arlington Library which allows users to search only the most recent books added to our collection [6]. Jim Robinson, of the Tarrant County College Library, later extended this feature with RSS feeds of the items within the New Books List, sorted by Library of Congress class [7].

A New Books List RSS feed pluslet was built for SubjectsPlus to integrate these RSS feeds into the subject guides. This pluslet allows the librarian to enter LC call number classes and choose the number of items to display. The pluslet will retrieve the latest items from the new books list that fall into those classes, and display them in the subject guide as links to the catalog records.

The New Books pluslet was created differently than the other custom pluslets, as a new PHP file was not needed. There are several files in SubjectsPlus that handle the preexisting RSS pluslets (general RSS, Twitter, del.icio.us, and Flickr). Additions were made to these files by copying and modifying the code for the other RSS pluslets for the New Books pluslet (Figure 5).

Additions made to guide_functions.php for New Books pluslet:

```php
case "newstuff":
    print $pluslet;
    print "<br />
    print "<input type="text" id="$this_instance" name="new-$itemtype-body-$idnum"
    print "<p style="font-size: 11px;padding-top: 3px;"> $vRss1 <input type="text" id="
    print $close_pluslet;
    return;
    break;
```

Additions made to guide.php for New Books pluslet:

```javascript
$(".find_newstuff").each(function(n) {
    var feed = $(this).attr("name").split("|");
    $(this).load("includes/feedme.php", {type: 'rss', feed: feed[0], count: feed[1], show_des });
});
```

Additions made to local.js for New Books pluslet:

```javascript
$(".find_newstuff").each(function(n) {
    var feed = $(this).attr("name").split("|");
    $(this).load("..../subjects/includes/feedme.php", {type: 'rss', feed: feed[0], count: fe });
});
```
One of the benefits of a locally hosted, open source system is the ability to make that system look exactly like the library website and conform to campus-wide visual standards. This was easy to accomplish in SubjectsPlus. Using the Firebug addon for Firefox to test new styles and compare our library website with SubjectsPlus side-by-side, the visual customizations were uncomplicated and quick to achieve.

The most significant visual alteration was creating a visual separation of the right-hand sidebar column from the main column content (Figure 6). The pluslets in each column contain the same CSS classes, so separating them was achieved by using descendant selector rules [8]. The sidebar pluslets exist within <div id="sidebar">, so a second set of rules was added by defining all classes of the pluslets within the sidebar separately (Figure 7).

Example of CSS variations:

```
titlebar_text {
    background: transparent;
    color: #fff;
    font-weight: 700;
}
```

```
#sidebar .titlebar_text {
    background: #eee;
    color: #000;
    font-weight: bold;
}
```
Since WSU decided to use SubjectsPlus to manage our electronic database links, it was essential to maintain an experience in SubjectsPlus similar to the end user’s experience using the database web pages. SubjectsPlus produces an alphabetical list of all databases, including their descriptions and a search box for locating a database. However, out-of-the-box SubjectsPlus does not contain a way to browse databases by subject – something the patrons at WSU are used to.

A database by subject page was created using a three step process:

1. An array of parent subject areas (humanities, business, social sciences, etc.) was defined and added to the config.php file.
2. The values in the array were added as a drop down box option to the Create New Guide form (with minor changes made to the database structure to handle the new field).
3. A new page was created to display a direct link to the databases section of each subject guide (using an anchor link common to each guide), sorted by the array values.

This page, which was created by copying and modifying the main SubjectsPlus index page, filters out the topic and course guides, displaying only links to the databases by subject area.

‘Area’ Array added to config.php:

```php
```

Creating the links by subject area:

```php
foreach ($area as $value) {
    //Display parent subject areas
    $options = "";
    $leftcol .= "
    <div class="pluslet">
        <div class="titlebar">
            <div class="titlebar_text">$value</div>
        </div>
        <div class="pluslet_body">
            // Call all subject guides (not topic or course guides) that are live for each parent subject
            $q = "select subject, subject_id, shortform from subject where active = '1' and Area = '$val";
            $r = mysql_query($q);
            while($myrow = mysql_fetch_array($r)) {
                //Create a link to the databases section (anchor #3) for each subject guide
                if ($mod_rewrite == 0) {
                    $link = "subjects/" . $myrow[shortform] . '#3';
                } else {
                    $link = "guide.php?subject=" . $myrow[shortform] . '#3';
                }

                //Display the links with the subject guide name
                $leftcol .= "
                    <a href="$link">$myrow[0]</a>&nbsp;
                ";
            }
        
```
Reactions to the Change

The librarians were excited about the potential that SubjectsPlus pages held. They were apprehensive about the time it would take to learn the new software and complete the task at hand. Once they received training and began using SubjectsPlus, they realized it was not as daunting as they had first perceived it to be. They also realized that teaching bibliographic instruction sessions would be easier because all the information for a particular subject would be located on one page rather than in two places on the library website; on a subject page and on a databases by subject page.

When the idea of course guides was introduced to a group of non-library faculty members at a Blackboard training session, an Education professor was very eager to work with a librarian on developing a guide for her course. Since then eight course guides have been created: BIO 210, HP 330, CHEM 532, PSY 328, ME 251, ENGL 102, COMM 111, and LASI 170. These guides have eliminated the need for lengthy handouts and paper copies. The librarians created each course guide for its own purpose. Some teaching faculty have a specific need that they want addressed by the library resulting in highly structured guides that complement a class and contain class assignments. Other course guides are more general and contain only a list of resources to aid students with research projects.

For example, the BIO 210 (General Biology 1) course guide was created to help with a specific library assignment created by the instructor for that class. There were nine sections of the course being taught in Fall 2010, with a total of 213 students. The guide was created to reflect the previous paper copy of the assignment with links to resources that could be used to answer the questions. It also included additional help and tips. Having the assignment in a course guide instead of on paper was more convenient for the students.

Librarians have created several Topic Guides, including Children's Literature, Reader’s Advisory & Leisure Reading, SAC & SAG Movie Collection, and Style Manuals, to name a few. The librarian for Juvenile Literature has received positive feedback from a Curriculum and Instruction Professor who said, “I love it!” The Professor is also recommending her class bookmark the course guide and is planning to make assignments from this guide in a future course. The SAC & SAG Movie Collection and Reader’s Advisory & Leisure Reading guides have also received positive feedback.

Plans for the future

Overall the WSU Libraries is happy with the outcome of SubjectsPlus, but there is still room to grow. Many of the future plans hinge on the addition of new features in upcoming releases of SubjectsPlus. A recent survey of the SubjectsPlus community suggested the following enhancements: a link checker, a mobile interface, and better integration with course management systems such as Blackboard [9]. All of these would be useful in meeting the current initiatives of our library, including expanding the ‘Embedded Librarian’ concept by creating more library content that can be easily embedded into Blackboard, and developing more mobile-friendly services.

Other plans include the creation of more custom pluslets to enhance services, such as a ‘Resource of the Month’ pluslet to promote specific databases, and additional collaboration with teaching faculty to create more course-specific guides. Library Administration desires that a course guide be created for each class with a high DWIF (drop/withdraw/incomplete/fail) rate to aid the university in raising success in these courses.

You can view WSU’s new library guides at http://libraries.wichita.edu/subsplus/

Acknowledgments

We wish to acknowledge Kathy Downes, Nan Myers, and Stephanie Sauls who were also members of the SubjectsPlus Implementation Team at WSU Libraries, as well as Andy Speagle, Systems & Storage Administrator in the University Computing & Telecommunications department.

References


About the Authors

Gemma Blackburn is Assistant Professor & Library Systems Developer at Wichita State University (gemma.blackburn@wichita.edu)

Mary Walker is Assistant Professor & Electronic Resources Librarian at Wichita State University (mary.walker@wichita.edu)