Using VuFind, XAMPP, and Flash Drives to Build an Offline Library Catalog for Use in a Liberal Arts in Prison Program

When Grinnell College expanded its Liberal Arts in Prison Program to include the First Year of College Program in the Newton Correctional Facility, the Grinnell College Libraries needed to find a way to support the research needs of inmates who had no access to the Internet. The library used VuFind running on XAMPP installed on flash drives to provide access to the Libraries' catalog. Once the student identified a book, it would be delivered from the Libraries to students on request. This article describes the process of getting VuFind operating in an environment with no Internet access and limited control of the computing environment.

by Julia Bauder

Why Build an Offline Library Catalog?

Why VuFind and XAMPP?
Launching and Shutting Down VuFind

The Easy Part: Installing VuFind

The instructions on the VuFind installation wiki can be followed mostly as written, with a few exceptions.

- Instead of installing Apache, MySQL, and PHP, install XAMPP Lite \[1\], which includes all three components and configures them correctly, on the external drive.
- Smarty (the templating system used by VuFind) can be installed as directed on the VuFind installation wiki, following the advice there for installing Smarty with XAMPP.
- The JDK (Java Development Kit) can also be downloaded and installed as normal, with a few exceptions. Be sure not to allow the installation script to install the public JRE (Java Runtime Environment). (Letting the script install the public JRE won’t affect the functioning of the catalog, but it will cause problems on your own computer later, when you’ve removed the external hard drive and the JRE is no longer accessible to your computer.) Also, the JDK should be installed to the flash drive or external hard drive rather than to the internal hard drive, and it is not necessary to set JAVA_HOME as directed in the VuFind installation instructions. (JAVA_HOME and other environment variables will be set dynamically by the scripts used to launch VuFind, since the drive letter for the flash drive may change each time the catalog is run.)
- The final step of the JDK installation instructions on the VuFind wiki is adding the paths to PHP and PEAR to the system PATH variable. Making this change is necessary for the VuFind installation batch file to run as intended, since the batch file installs a number of PEAR packages. These paths should be removed from the system PATH variable after the installation script is run, since PHP and PEAR on the external drive will not be a permanent part of your system.
- The configuration files must be edited as directed on the VuFind installation wiki, except they need to be relative (without drive letters), since the drive letter for the flash drive may change each time the catalog is run.

Launching and Shutting Down VuFind

Since VuFind will be launched by the user, and may be launched from a different location each time, some changes are needed to the scripts that launch it. I moved the script used to start the catalog (vufind.bat, which is created by the VuFind installation script) out of the vufind folder and into the root directory of the flash drive to make it easier for users to find (and to allow us to make the vufind folder hidden) and renamed it “Start VuFind – Run Once to Start VuFind.bat.” I added four lines to this script: two to launch PHP (and none of it is in Ruby), so there was pre-existing staff experience with the language. Also, I was confident that VuFind would run on XAMPP (which was clearly going to be a critical component for carrying out this project with the minimum amount of work), because another VuFind user, Greg Pendlebury, had already added advice on the VuFind installation wiki about how to install VuFind on XAMPP. \[3\] However, for institutions with a reason to prefer using Blacklight, it should presumably be possible to modify that discovery layer to work in an offline environment in roughly the same way that we modified VuFind.

Why Flash Drives?

The original prototype of the offline catalog was created on an external hard drive, but we decided to copy the catalog to flash drives for use in the prison for cost-saving reasons: flash drives are cheaper, plus they are easier to transport and store. Be aware that, depending on security settings, Windows may not allow XAMPP to run from a flash drive, or even in some cases from an external hard drive. The security settings on the computers in the prison lab, luckily, allowed the catalog to run from flash drives.

We originally decided to install the catalog on external drives for procedural rather than technological reasons. Grinnell’s Liberal Arts in Prison Program does not administer the computer lab that it uses in the prison; the lab is managed by IT staff from Des Moines Area Community College (DMACC), which runs other educational programs for prisoners. Installing the offline catalog on external drives required less coordination with the IT staff at DMACC, and it would allow the drives to be brought back to campus periodically for software and data updates. However, we have experimented with copying the catalog onto the desktop, since it seems to run slightly faster after doing so. The Prison Program was recently permitted to place several Grinnell College-owned computers in the prison’s library; the offline catalog will be run from the desktop on these computers in future semesters.

I briefly considered trying to create an offline catalog that could be burned to DVD-ROM, which would have made creating multiple copies of the catalog even cheaper. It also would have been more robust, since we wouldn’t need to worry about students accidentally deleting files or otherwise altering the catalog. However, re-writing the VuFind code to work when operated from read-only media would have required more time than I was able to devote to the project.

How to Do It

The Easy Part: Installing VuFind

The instructions on the VuFind installation wiki can be followed mostly as written, with a few exceptions.

- Instead of installing Apache, MySQL, and PHP, install XAMPP Lite \[1\], which includes all three components and configures them correctly, on the external drive.
- Smarty (the templating system used by VuFind) can be installed as directed on the VuFind installation wiki, following the advice there for installing Smarty with XAMPP.
- The JDK (Java Development Kit) can also be downloaded and installed as normal, with a few exceptions. Be sure not to allow the installation script to install the public JRE (Java Runtime Environment). (Letting the script install the public JRE won’t affect the functioning of the catalog, but it will cause problems on your own computer later, when you’ve removed the external hard drive and the JRE is no longer accessible to your computer.) Also, the JDK should be installed to the flash drive or external hard drive rather than to the internal hard drive, and it is not necessary to set JAVA_HOME as directed in the VuFind installation instructions. (JAVA_HOME and other environment variables will be set dynamically by the scripts used to launch VuFind, since the drive letter for the flash drive may change each time the catalog is run.)
- The final step of the JDK installation instructions on the VuFind wiki is adding the paths to PHP and PEAR to the system PATH variable. Making this change is necessary for the VuFind installation batch file to run as intended, since the batch file installs a number of PEAR packages. These paths should be removed from the system PATH variable after the installation script is run, since PHP and PEAR on the external drive will not be a permanent part of your system.
- The configuration files must be edited as directed on the VuFind installation wiki, except they need to be relative (without drive letters), since the drive letter for the flash drive may change each time the catalog is run.

Launching and Shutting Down VuFind

Since VuFind will be launched by the user, and may be launched from a different location each time, some changes are needed to the scripts that launch it. I moved the script used to start the catalog (vufind.bat, which is created by the VuFind installation script) out of the vufind folder and into the root directory of the flash drive to make it easier for users to find (and to allow us to make the vufind folder hidden) and renamed it “Start VuFind – Run Once to Start VuFind.bat.” I added four lines to this script: two to launch...
Interface Changes

To make the VuFind interface work in a non-networked environment, most features that rely on Internet connectivity to function, such as course reserve lists, new item lists, Ask a Librarian, book covers from external providers, Email This Search, Text This, etc., must either be turned off in configuration files or links to them should be commented out in the Web interface. Other features such as creating an account, adding to favorites, adding tags, saving searches, etc. are also problematic in this environment—they will technically work, but since they only exist on one specific copy of the catalog, they are not very helpful—and should also be commented out or deleted.

The Hard Part: Call Numbers and Locations

VuFind normally pulls call number and location information from the catalog in real time, but simply commenting out the call number and location information was not an ideal solution. We wanted the incarcerated students to be able to copy the call number and location information to the paging slips that the Libraries created so that College staff would not have to look up the call number and location of each book when pulling the items. Also, some of the reference and instruction librarians at Grinnell (myself included) like to use the Library of Congress Classification as a teaching tool, to talk about disciplinary differences in perspectives on a given topic and clues as to which disciplinary perspective a book is coming from. Since we intended to go to the prison from time to time to provide information literacy instruction to the incarcerated students, I wanted the books' call numbers to be visible in the offline catalog for this purpose as well.

Thus, I edited VuFind's code to force it to pull the call number and the location from the MARC record, which is stored in its entirety in the Solr index. In the VuFind 1.0 RC2-based catalog that was actually used in the prison this semester, these changes were made in several different template files; in version 1.2, all of the changes can be made fairly easily in the MARC record driver, although in the long run it would probably be easier to create and maintain a whole new pseudo-ILS driver that connects with the Solr index rather than with an ILS. (The MARC record driver is somewhat more likely to change in new versions of VuFind in ways that would require re-writing the changed code, while the basic functions within ILS drivers should not typically need to be modified when upgrading to new versions of VuFind.)

When using version 1.2, I set the Sample driver as our driver. This driver, which was originally intended for non-production purposes only, supplies hard-coded dummy information for many possible requests. (E.g., it will always inform users that there are no new items and nothing on reserve.) For our intended uses of the offline-catalog, this dummy information was mostly sufficient, except for the above-noted call number and location information. To pull this information from the MARC record, I copied the getOptions function from the Sample driver to the MARC record driver (allowing it to take advantage of the many MARC-specific functions included in that class), and edited it as follows to pull the call number and the location from the MARC record rather than returning hard-coded dummy information. Then I edited the getRealTimeHoldings function in the MARC record driver to use that function, rather than trying to connect to the catalog via a catalog driver and the HoldLogic class, to get the call number and location.

Call Numbers: The Solr index stores call numbers in three ways: a normalized call number field (with spaces removed) that is meant to be used only for behind the scenes purposes such as sorting; a field named callnumber-a that stores the first part of LC-style call numbers with spaces so that they can be used for display; and as part of the full MARC record. The first two functions take the first call number found in fields 099, 090, and 050. The getcallnumber function simply retrieves the callnumber-a field for a given item. The code below then appends the second part of the call number by pulling it from the MARC record. Since SuDoc call numbers are not included in the callnumber-a field, both halves of those call numbers need to be taken directly from the MARC record. (The code below does not check the 099 field because the only campus collections with call numbers in the 099 fields are the media collections and parts of special collections, neither of which were indexed into the offline catalog because we did not intend to deliver items from those collections to the prison. In general, excluding the media collections and special collections, it can safely be assumed that if an item in our system has an LC number in the 099 field it is shelved under that number, if it has a SuDoc number in the 086 field it is shelved under the SuDoc number, and if it has neither it is shelved under the LC number in the 050 field; so the code makes that assumption.)

Locations: The location codes are stored in subfield a (a repeating sub-field) of a non-repeating 998 field in the records exported from our ILS; this may vary depending on your particular set-up. These location codes are not human-readable, but VuFind's built-in translation capability makes it easy to translate these codes into human-readable English (or any other language) locations in the public display by simply adding the codes and their human-readable translations to the appropriate language translation file(s).

Original code from Sample driver:

```java
1. set VUFIND_HOME=%~d0\vufind // if the catalog will only be run from external drives
2. set JAVA_HOME=%~dp0\JAVA_HOME // gives the option of copying the catalog to the desktop
```

The scripts used to import records also must be edited in a similar fashion to set VUFIND_HOME and JAVA_HOME.
Edited code, moved into MARC Record driver:

```php
public function getStatus($id)
{
    $holding[] = array('availability' => 1,
    'status' => 'Available',
    'location' => '3rd Floor Main Library',
    'reserve' => 'No',
    'callnumber' => 'A1234.567',
    'duedate' => '',
    'number' => 1);
    return $holding;
}
```

How It Works from the Students' Perspective

When the student plugs in the flash drive, they see three files: Start VuFind – Run Once to Start VuFind.bat (discussed above), Stop VuFind.bat (a simple three-line script that stops XAMPP gracefully and kills VuFind, allowing the flash drive to be removed safely without shutting down the computer when necessary), and a Web shortcut to VuFind at http://localhost/vufind. (The vufind, xampp and JDK folders are hidden.) (See Figure 1.)

![Figure 1: Shortcuts for running VuFind](image1)

The instructions for the students are relatively easy. First, they need to double-click on “Start VuFind – Run Once to Start VuFind.” This launches XAMPP and Solr in command prompt windows. Solr needs anywhere from a few seconds to a minute before it is ready to handle searches; during this time, text will be scrolling in the command prompt window. (See Figure 2.) Students should be told to wait for the command prompt window to stop scrolling, then to double-click on the Internet shortcut to VuFind. Students should also be instructed that, if they need to remove the flash drive, they should either shut down the computer or run Stop VuFind first.

![Figure 2: DOS screens while Solr initiates](image2)
The students in the one class that has used the offline catalog to date seemed to find it intuitive to search; very little instruction in how to navigate the offline catalog was needed.

Going Further

An offline library catalog is a niche product today, when the vast majority of library patrons in developed countries either have Internet access at home or can go to a physical library to access the Internet. Yet those potential patrons who cannot easily access an online catalog—a category that is not limited to incarcerated students—are likely to be some of the most under-served patrons. For example, consider rural patrons without Internet access who receive their library services through a non-Internet-equipped bookmobile. An offline library catalog might be the only feasible way to provide such patrons with equitable access to resources that these patrons could request through their library system if they were aware that the resources existed.

Ideally, an offline library catalog should not just replicate the basic search and display functions of the online catalog, but should also contain extra features that would only make sense in the offline environment. For example, in our case, instead of having the incarcerated students fill out request forms by hand, it would be helpful to allow students to mark books of interest and then print out a request form with their list of desired books at the end of their search session, without having to create an account. To date, due to time constraints I have mostly been concerned with removing features that did not work in the offline environment and ensuring that the most critical functions worked, but in the future I hope to be able to add features that would improve the experiences of those patrons who must use an offline catalog but would not make sense in a typical online library catalog.

Notes:

1. EBSCO does offer an offline periodical database on CD-ROM that is intended for use in prisons and other institutions that lack Internet access, but for various reasons it was decided that that database was not a good choice for this program at this time.


3. The VuFind installation wiki: http://vufind.org/wiki/installation_windows


5. For more information on batch parameters, see http://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/batch.mspx?mfr=true.

Acknowledgements

The author thanks Emily Guenther and Micah Bot-Miller for all of their help in making the prison catalog a reality, as well as the Code4Lib Journal editors and everyone else who provided feedback on drafts of this article.

About the Author

Julia Bauder is the Data Services Librarian at the Grinnell College Libraries. She graduated with a BA from Bard College at Simon’s Rock and an MLIS from Wayne State University, and can be reached at bauderj@grinnell.edu.

Subscribe to comments: For this article | For all articles

2 Responses to “Using VuFind, XAMPP, and Flash Drives to Build an Offline Library Catalog for Use in a Liberal Arts in Prison Program”

Please leave a response below, or trackback from your own site.

1. Nada O’Neal, 2012-02-06

   Great work! I hope you get time in the future to work on some of the offline-specific features you described, and that you release your code changes to github or some place similar. :)

2. Adalid Ortiz, 2012-06-22

   Hey, this is great. Why not make a portable ready to use version to download like this ILS http://framakey.org/WebApp/PMBPortable

   Saludos,
Leave a Reply

Name (required)

Mail (will not be published) (required)

Website

Sorry, something went wrong. Please try reloading the page.

Submit Comment

ISSN 1940-5758

Current Issue
Issue 43, 2019-02-14

Previous Issues
Issue 42, 2018-11-08
Issue 41, 2018-08-09
Issue 40, 2018-05-04
Issue 39, 2018-02-05
Older Issues

For Authors
Call for Submissions
Article Guidelines
This site uses cookies for analytics, personalized content and ads. By continuing to browse this site, you agree to this use. Learn more.

TechNet. Today when we checked in to DPM it showed the whole library as being offline and we can not connect to either the drive or slots. However, a new drive appeared as stand alone, but without any slots. This drive is the one inside the library but there haven been any recent changes that should make it appear separate. We tried rebooting the DPM server. We turned off the library and rescanned it in DPM, that made the stand alone drive disappear but when we turned it back on the actual Overland library still said it was offline and when we rescanned that it re-added the stand alone drive. Bauder, Julia. “Using VuFind, XAMPP, and Flash Drives to Build an Offline Library Catalog for Use in a Liberal Arts in Prison Program,” Code4Lib Journal, no. 16 (2012). http://journal.code4lib.org/articles/6225. Julia Bauder. 1807 6th Ave. Graduate Student Assistant, Library and Information Science Program, Wayne State University, Detroit, MI. Freelance researcher and writer, Wixom, MI. Publications: Bauder, Julia. “Visualizations in Discovery Systems.” In Top Technologies Every Librarian Needs to Know, 2nd ed., edited by Ken Varnum. Chicago: American Library Association Editions, 2018. Bauder, Julia, and Catherine Rod.