LIBRARY À LA CARTE: OPEN SOURCE SOFTWARE FOR LIBRARY RESEARCH GUIDES

Evviva Weinraub[1], Margaret Mellinger[2]

[1]Oregon State University Libraries (USA)
[2]Oregon State University Libraries (USA)
Evviva.Weinraub@oregonstate.edu, Margaret.Mellinger@oregonstate.edu,

Abstract

The core mission of university libraries is to provide access to books, journals, data and primary sources to enhance and enable research and education. Because students are accustomed to gathering information using commercial search engines, they can overlook high-quality library resources. To connect with student researchers when they need library resources, many libraries have built online research guides tied to specific subjects and courses. These guides can address specific assignments where research is required, and course instructors can point students to the guides to assist them in completing their research assignments successfully. In 2006, Oregon State University (OSU) Libraries developed an open source software product called Library à la Carte that librarians can use to construct research guides easily and quickly. The code was made available to other libraries, and a number of institutions have adopted the tool. A successful user community has grown up around Library à la Carte, but development up to this point has been the sole responsibility of OSU Libraries. We are now embarking on building a robust developer community, to contribute code, to build additional features and to take Library à la Carte into the future. As part of our plan to move forward, we are testing a hosted (cloud-based) version of the software. This paper introduces Library à la Carte, presents what was learned from the test and discusses moving forward with building a developer community. In addition, we touch on the benefits and challenges of building and sustaining an open source product from the ground up.

Keywords: open source software, libraries, cloud computing, library guides

1   INTRODUCTION

1.1   Library à la Carte

Library à la Carte is a custom content management system used to publish Web pages that highlight academic library resources and services. To use the software, librarians visit a Web site and log in to their personal accounts to build of three kinds of Web pages: course assignment guides, subject guides, and tutorials. These Web pages are composed of "modules": smaller units of content that can be re-used, shared, and edited. Modules can contain various kinds of content. For example, book titles from the library online catalog, links to subscription article databases, selected Web sites, or instructions on how to conduct research. Other types of modules allow librarians to include instructional videos, specialized search boxes, images, user comments, polls or quizzes. So that students have a way to get additional help, each of the Web pages includes contact information for the librarian who authored the page. Contact modules can also incorporate a chat box. Library à la Carte pages can be organized using tabs, and the content on a tab can be displayed in one or two columns.

1.1.1   Course Assignment Pages

The aim of the course assignment pages is to help students connect with library information to complete particular assignments within a course. Librarians communicate with course instructors to learn about assignments that have a strong research and writing component. With the assignment in mind, librarians select the best library resources for students to use to conduct the research they need to complete the assignment successfully. Librarians share the link to the course assignment with the course instructor, who can embed it in a syllabus or within learning management software. These pages can be used alone or in conjunction with a library instruction session where a librarian is a visiting instructor for the class.
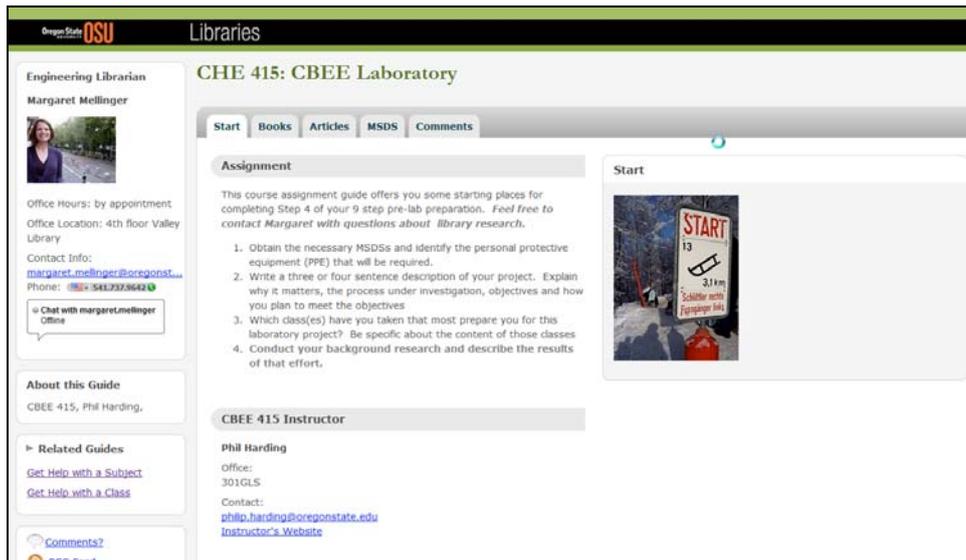
Fig. 1 Course Assignment Guide

1.1.2 Subject Research Guides

Subject research guides are broader than course assignment pages and give library users a starting place for research in a particular discipline. Librarians highlight the major reference books, electronic resources, journals, and databases for a discipline. Subject guides may also include guidance on using particular format types important within a field, such as standards and technical reports for engineering, or primary sources for history.
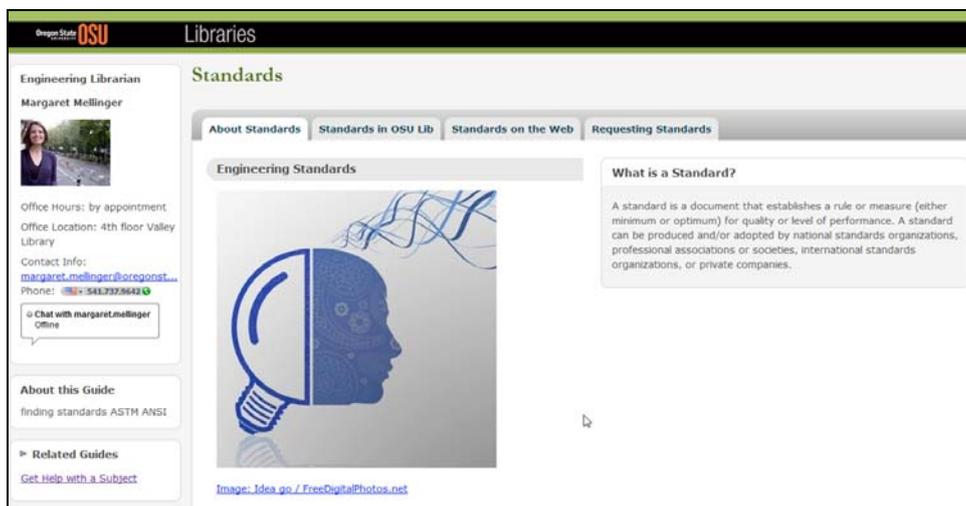

Fig. 2 Subject Research Guide

1.1.3 Tutorials

Tutorials lead students through a particular process or teach them research skills. Using tutorials, students can learn to find books in the library, discover articles in databases, develop search strategies, cite sources properly, or learn to use a particular tool, like Zotero (see Fig.3). Quizzes within the tutorials give students immediate feedback, while librarians and instructors can use quiz results to assess student learning.
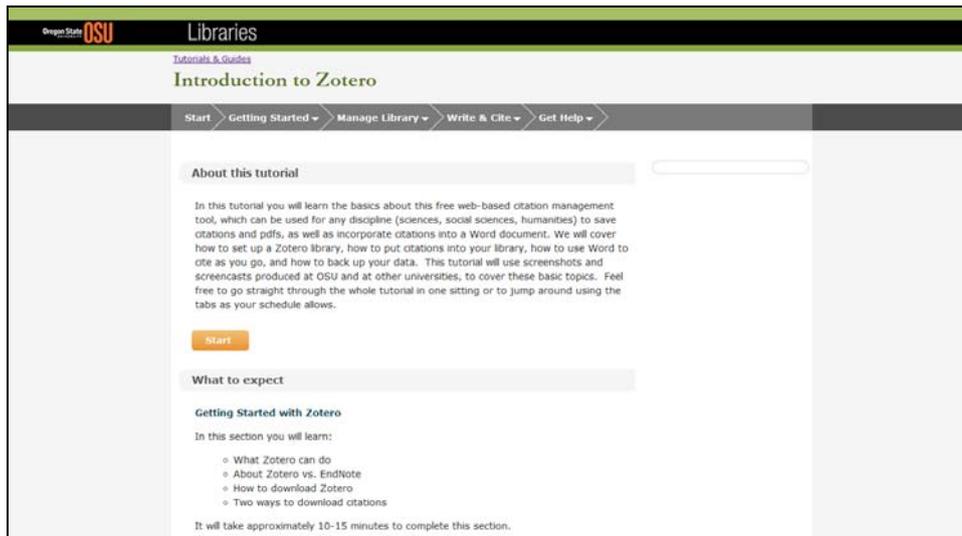
Fig. 3 Tutorial

2   OPEN SOURCE SOFTWARE

Library à la Carte is free and open source software that can be redistributed and/or modified under the terms of the GNU Affero General Public License. The software was developed using Ruby on Rails with a MySQL database. The code can be downloaded from the Library à la Carte Web site (http://alacarte.library.oregonstate.edu/), or from the RubyForge code repository (http://rubyforge.org/projects/alacarte/). In addition to being open source, it is also cross-platform. It can be installed on Windows, Linux or Mac.

Library à la Carte originated as an in-house solution to help Oregon State University (OSU) Libraries connect with students at their point of need – when they have a research assignment. Although it was released as open source software, interest in Library à la Carte from other libraries looking for a similar solution exceeded expectations. There have been hundreds of downloads of Library à la Carte, and over a thousand accounts registered on the demonstration version of the software. Library à la Carte employs standard open source applications (Ruby on Rails, MySQL). It became apparent that, while many librarians wanted to use such a tool, there were significant barriers preventing institutions from choosing the software. Reported barriers included insufficient access to Web servers, lack of technical support staff to install the software, and inability of library technical personnel to take on and maintain yet another software tool. OSU Libraries is aware of more than twenty libraries worldwide that have implemented Library à la Carte in the last few years. There may be more, as there is no mechanism in place to track absolute numbers. It is likely many more libraries would adopt the software if a turnkey, pre-installed version was offered at low cost.

Libraries are not primarily in the business of software development. While the tenets of open source software align with the sharing mindset of librarianship, most libraries do not have the resources in place to support a software development project from cradle to grave. Over the lifetime of this project, OSU Libraries has come to realize the level of staffing devoted to this project is not enough to provide the support and effort needed to expand Library à la Carte's capabilities. When developing the software, the programmer spent 100 percent of her time on it, with varying numbers of hours from student programmers, librarians and the head of the library technology department. As the project matured, and the priorities of the library changed, 30 percent of the programmer's time was spent on Library à la Carte with help from one student programmer and occasional hours from librarians and administration. It is time to widen the group of people involved with Library à la Carte.

OSU Libraries is approaching this challenge in two ways. First, to set in place a way to broaden the pool of adopters, it will be necessary to offer the software as a hosted or cloud-based solution. To determine the feasibility of doing so, tests running Library à la Carte on the cloud have begun. The second way to approach the challenges of adoption and staffing is for OSU Libraries to leverage support from the existing group of users to build a robust and supportive developer community.

3 CLOUD BASED IMPLEMENTATIONS

OSU Libraries has conducted two of three planned tests of Library à la Carte on cloud platforms thus far. The first cloud-based test was with a partner institution using the Heroku platform [1]. The second consists of mounting a copy of OSU Libraries' Library à la Carte production environment on Amazon Web Services [2]. The primary reasons for conducting these tests was to ascertain if there would be any issues with configuration, installation, or functionality in those environments and to develop cost models based on likely usage.

3.1.1 Heroku Cloud Application Platform

We chose Heroku for the first test implementation for two reasons. First, Heroku is a platform as a service (PAAS) for applications using Ruby. Library à la Carte was developed using Ruby on Rails, so this seemed a good match. Second, Heroku offered a very low cost (free in Fall 2010 – Spring 2011) shared database environment for testing of low-scale production applications. Library à la Carte is not a large program and it was not anticipated to exceed the maximum size allowed by Heroku's shared database (20 GB). If it had, Heroku provided a seamless way to move into its tiered pricing for dedicated databases.

Initial evaluation from the pilot test users was quite positive. The software seemed to be working correctly and one librarian created several pages and guides. Problems arose when the administrator tried to add new users to the system. Other problems were not easy to troubleshoot, such as unpredicted downtime of the Heroku platform. Ultimately, it was not possible to determine from the Heroku test if Library à la Carte worked well as a cloud application. Too few pages were created and too little traffic went to those pages to provide enough feedback to set a cost structure.

In addition, although Heroku works specifically with Ruby applications, it uses PostgreSQL for its relational database engine. Unfortunately, Library à la Carte uses MySQL. In order for the search function to work properly on Heroku, the database would need to have been migrated to PostgresSQL. This was not done for the pilot cloud test on Heroku, but had the test implementation been more successful, the issue could have been resolved.

3.1.2 Amazon Web Services

The second cloud test was conducted in two phases. Phase I tested the installation, configuration and functionality of Library à la Carte on Amazon Web Services (AWS). Phase I was designed to outline what it would take to set up a copy of OSU Libraries' production installation of Library à la Carte on the cloud, testing all major functions and making sure everything worked exactly the way it would in a local installation.

A copy of OSU Libraries' production version of Library à la Carte was installed on AWS Free Usage Tier, with the following specifications:

- 750 hours of Elastic Compute Cloud (EC2)
- Linux/Unix Micro Instance
- 613MB Ram
- 1-2 EC2 compute units (1 CPU, burstable to 2)
- 30GB Amazon Elastic Block Storage plus 2 million IOs
- 15GB outbound bandwidth aggregated across all AWS services
- 1GB of Regional Data Transfer

The AWS test ran for two weeks beginning in early January 2012. Ten OSU librarians participated in the test, all of whom were familiar with the software and knew how it should work. Each participant performed twelve tasks that they would normally do when creating, editing and sharing Library à la Carte pages. Testers found that the software performed as well on the cloud as it does on the local server.

Using the installation and configuration documentation, Library à la Carte was set up on the EC2 server. Because Amazon EC2, unlike some other, more restrictive services, is close enough to a

"regular" server, there were no issues with installation, and only one minor issue with configuration related to modifying a security group. One small problem occurred with opening port 80, which is closed by default in EC2. Amazon firewalling happens at a higher level and Amazon's security groups must be set to allow access to port 80. The first time Library à la Carte was brought online, it did not connect. Once this issue was fixed, testing began. Other issues may arise in Phase II testing, but for Phase I, Amazon EC2 is working well.

For Phase II testing, OSU Libraries' production version of Library à la Carte will be installed on an AWS On-Demand Instance for one term (10 weeks). On-Demand Instances are recommended for applications being developed or tested on Amazon EC2 for the first time [3]. Real time information will be gathered on how many EC2 compute units and how much storage it will take to run the Library à la Carte program at a medium-sized university library. From this information, a cost structure for running Library à la Carte on AWS will be derived. One issue to address in the second phase of this test is that of user authentication. Phase I of the test used local, rather than LDAP or SSO authentication. In Phase II, LDAP authentication will be set up. Library à la Carte is not yet Shibboleth enabled, but because of the modular nature of the program, it should not be difficult to build an appropriate module. Phase II of this test will begin in March 2012. While completing Phase II of AWS testing, OSU Libraries will be soliciting members for a core developer community.

4   BUILDING A COMMUNITY

In *The Cathedral and the Bazaar*, Eric Raymond describes stages of open source software development, one of which is the "incubation period" where the initial creators of the software have worked to get something off the ground [4]. In some ways, Library à la Carte has gone beyond incubation (many successful installations, several major updates of the code), but the project has stalled in nurturing a community of programmers and users.

4.1.1   Governance Model

Building a robust and effective community around an open source tool takes resources and commitment. Oregon State University Libraries has been supporting the programming and development of Library à la Carte since its inception. Enlisting the community of users is essential for continuing maintenance, development and support. The original development plan for Library à la Carte did not include a governance model, but to bring contributors into development, one is needed. Potential contributors will want to know how to contribute to the code, what will happen to their code, and contributions will need to be handled in such a way as to retain the value of the original code [5].

The plan for governance is to put in place a Benevolent Dictator model of open source development [6]. A core group of code and content contributors will comprise the governance group and they will collaborate to set the direction of the project. The group will likely be small, so a top-down approach to governance will be the best initial course. In the short term, OSU Libraries' programmer will serve as leader of this project. After the core group agrees on a direction, the benevolent dictator role would evolve into a coordinator, which could be a rotating position within the group. Alternatively, the project could move to a meritocratic model [7] where contributors making the most positive contributions make consensus decisions about development.

4.1.2   Development

Agile programming will continue to be used for development. Agile programming works by breaking tasks into small pieces that do not directly involve long-term planning [8]. Programming iterations occur over a short period of time, which allows the project to adapt to changes quickly (like dealing with a major change in a browser), and to produce documentation as required. Minor changes (or iterations) happen continuously without having to wait for the next major release of the software. In agile development, bug fixes also happen continuously, so that fewer modifications are necessary with each major release.

There are many tasks for the group to do, such as writing a roadmap for the project, defining the infrastructure for contributing code and patches, setting up mechanisms to track bugs and collect suggestions for new features, and deciding on communication tools. The community will also make decisions about where to house the code repository and whether to continue to use Ruby Forge and/or GitHub. Not all the requirements of the project are in programming. User documentation needs to be written or revised. Group members could also monitor the discussion forums, update the project Web site, or conduct usability testing. Translating Library à la Carte into multiple languages (with

relevant updates as needed) would facilitate adoption worldwide. Currently Library à la Carte is available in English and French-Canadian; Université de Montréal in Québec Canada provided the translation.

5   CONCLUSION

Library à la Carte is an open source product at the tipping point. Its future depends on building a community interested in enhancing the existing tool and expanding its capabilities to meet user needs. When Library à la Carte was first conceptualized, the idea was to create something to improve access, and push library content out to users.

In the intervening years, the information landscape has become increasingly diversified and the demand for tools such as Library a la Carte has continued to grow. Unfortunately, the things that we want it to do far outstrip our programming staff's ability to make it happen. To answer this need – and to ensure that Library a la Carte can continue to meet the challenges and opportunities that emerging Web technologies provide – OSU Libraries is looking to cultivate partners who not only share our vision for its future, but who are willing and able to assist in making that vision a reality.

ACKNOWLEDGEMENTS

REFERENCES

[1]   Heroku. [online] Available: http://www.heroku.com/ [accessed 23 January 2012]

[2]   Amazon Web Services. [online] Available: http://aws.amazon.com/ [accessed 23 January 2012]

[3]   Amazon EC2 Instance Purchasing Options. http://aws.amazon.com/ec2/purchasing-options/.

[accessed 23 January 2012]

[4]   Raymond, E. (1999). The Cathedral and the Bazaar. Cambridge, MA: O'Reilly.

[5]   OSS Watch. Governance Models. [online] Available:

http://www.oss-watch.ac.uk/resources/governanceModels.xml [accessed 23 January 2012]

[6]   OSS Watch. Benevolent Dictator Governance Model. [online] Available:

http://www.oss-watch.ac.uk/resources/benevolentdictatorgovernancemodel.xml [accessed 23 January 2012]

[7]   OSS Watch. Meritocratic Governance Model. [online] Available:

http://www.oss-watch.ac.uk/resources/meritocraticGovernanceModel.xml [accessed 23 January 2012]

[8]   Cohen, D., Lindvall, M. and Costa, P. (2003). Agile Software Development. Data & Analysis Center for Software (January), p. 17.