

10

OPEN SOURCE SOFTWARE AND LIBRARIES

Ajit Pyati

OPEN SOURCE AND ITS DEMOCRATIC PROMISE?

Faced with the increased commodification of information resources and services and the encroaching dominant logics of business/corporate culture, today's libraries are in search of innovative technological solutions to protect the free flow of ideas. In response to this situation, a growing segment of the library profession is promoting open source software (OSS). OSS, both as a movement and as a form of software development, offers a challenge to dominant proprietary models of software development. The basic idea behind OSS rests in its inverted logic of property, in which value is derived in the distribution and freely available nature of software code (the building block of software), rather than in exclusive ownership over code (Weber, 2004).

OSS presents a more community-driven model of software development, with distributed developers creating code under various norms, informal and semiformal regulations, and licenses. The success of OSS can be seen in its penetration into various levels of society, through the Linux operating system, the popular Apache Web server application, and the Mozilla/Firefox Web browser, to name just a few prominent open source projects (Weber, 2004). Thus, while providing an alternative to commercial software development, OSS intersects comfortably with the proprietary software world and is also utilized by the for-profit sector. OSS and free software, however, are also associated with a sustained grassroots technology movement with an international following and community, and are often linked to alternative, community-driven visions of an information commons (Bollier, 2003). Open source seems to be implicitly linked to a wider democratic technology movement in

the world, with the idea that enhanced participation in technological decisions challenges dominant societal logics of commodification (Benkler, 2003).

Two large and successful open source projects that provide a glimpse into the power of open source software are Apache and Linux. Apache dominates the Web server market, and Linux is the major open source operating system, with nearly 40 percent of large American companies using Linux in some form (Weber, 2004). The success of these projects depends on a large pool of developers distributed across the world, with many developers contributing code on a volunteer basis. However, the administrative structures of these projects ensure their financial sustainability.

The Apache Software Foundation was incorporated as a nonprofit corporation in 1999, and now serves as an organizational umbrella for a range of Web-relevant open source projects (Weber, 2004). An Apache Software Foundation board of directors is responsible for the overall direction, coordination among the different projects, legal issues, and other kinds of central services that benefit the individual projects (Weber, 2004). The Apache Foundation thus manages and guides the development of the project. On a financial level, Apache development can be described in terms of a cost-sharing mechanism. For instance, Apache development conforms to a model in which competing software users find it to their advantage to cooperatively fund open source development because doing so gets them a better product at a lower cost (Raymond, 2001). A network of Webmasters has been able to pool their resources for a large project with wide benefits, rather than to compete against each other.

Linux, on the other hand, has a more semiformal organization for decision making about code, and the last word on Linux's code management structure rests with Linus Torvalds, the project's founder (Weber, 2004). Raymond (2001) describes Torvalds's style of development as, "release early and often, delegate everything you can, be open to the point of promiscuity" (p. 21). He further argues that the success of Linux, while owed in large part to Torvalds's vision, is sustained through the effective construction of voluntary communities of interest (Raymond, 2001).

The basic premise of the open source movement is the chance for more developers to have access to software code, which will allow for more successful and bug-free software to be developed (Raymond, 2001). This approach to open source development often has a more *practical* and applied ideological orientation. The open source movement, however, is also part of a larger political movement addressing more democratic and noncommercial forms of technology development. In *Free Software, Free Society: Selected Essays of Richard M. Stallman* (Stallman, 2002), the notion of free software is juxtaposed with open source software. Richard Stallman, a computer programmer who began his work at MIT, is the main founder of the free software movement. The main distinction he makes between the free software movement and the open source movement is in the value differences between the

two movements—he states that for the open source movement, the issue of whether software should be open source is a practical question and not an ethical one, whereas for the free software movement, open source is more of an ethical question and is part of a broader social movement (Stallman, 2002).

Given these divergent strands in the objectives of the open source and free software communities, how do libraries fit into this discourse? Libraries around the world have also begun utilizing OSS to help develop some of their services, and some have suggested that the gift culture of the open source programming community complements traditional library service values. Gift cultures are, according to Raymond, “adaptations not to scarcity but to abundance. They arise in populations that do not have significant material-scarcity problems with material goods” (2001, p. 81). He justifies this analogy by stating that there is no serious shortage of disk space, network bandwidth, and computing power, and software is freely shared. In a gift culture, social status is determined not by what you control but by what you give away—in the case of OSS, abundance creates a situation in which the only available measure of competitive success is reputation among one’s peers (Raymond, 2001). In the case of libraries, a gift culture exists in the sense that wide distribution and access to information is at the core of library functions. Also, the collaborative nature of OSS mirrors the many resource-sharing activities of libraries (Clarke, 2000).

The benefits of OSS can potentially reduce costs, give users more control, and increase software performance (Courant & Griffiths, 2006). OSS certainly appears to give libraries more control over technological choices and an ability to bring library values to software (Frumkin, 2002); however, the technological, institutional, and social dimensions of this phenomenon need to be explored in further detail. In addition, successful OSS projects often depend on visionary leadership and the development of a large user and codeveloper community (Raymond, 2001), issues that need to be addressed in the context of library open source projects.

A growing and diverse open source development community in libraries exists, but little research has focused on understanding how or whether OSS can enhance library service ethics and goals. It is important to keep in mind, however, that open source products are as commonplace as Web server applications (e.g., Apache), databases, programming languages, and operating systems such as Linux (Weber, 2004). Libraries thus interact and use many of these products and are thus open source users on a Web infrastructure level. In addition, some commercial library vendor products utilize some open source tools and applications (Chudnov, 2006).

This chapter is a first step in contextualizing the terrain of the library open source world within the larger open source/free software debates. I identify some key library open source examples, and explore challenges in the development of library open source projects, along with future lines of action

and best practices research. Articulating a political angle to the library open source debate—particularly given the rise of a techno-corporate culture in the library world (Apostle & Raymond, 1997)—remains a task of growing importance and concern.

LIBRARIES AND OSS: EXPLORING THE TERRAIN

An active open source library community exists, with groups such as OSS-4Lib and Code4Lib serving as clearinghouses for various library open source projects (Oss4Lib, n.d.). OSS is now being considered as a viable alternative to the often expensive proprietary library automation systems. For instance, a major OSS suite available for libraries is Koha. Koha is the first open source integrated library system (Koha, 2005) allowing libraries to have access to library automation software. The software is free and follows the guidelines of the open source general public license (GPL). Developed in New Zealand by a company called Katipo Communications, the software now has been adopted in several libraries around the world. For instance, in New Zealand, the Horowhenua Library Trust has implemented the Koha OSS, as well as other libraries in North America and the rest of the world.

The open source community within libraries is growing, with various applications developed for both academic and public libraries, as well as the presence of library-based groups dedicated to the promotion of OSS (e.g., Code4Lib; Oss4Lib). The size of the library open source community is seen in a growing list of publications focusing on library-based OSS developments (Chawner, 2006), and it appears that the library open source community is approaching a level of critical mass in its development. In fact, some argue that the concept of OSS has become increasingly popular in the library field, with many librarians often discouraged with commercial integrated library systems (Breeding, 2007).

Why would libraries be interested in utilizing OSS? Many of the purported advantages—such as cost, customization, a rapid development cycle, more bug-free software—certainly apply. In addition, the context of the largely commercial library automation and vendor market has played a role in pushing libraries to consider OSS. A consolidation of major library automation vendors over the years has reduced the number of choices libraries have for their automation needs (Breeding, 2006), while a changing information environment dominated by Internet technologies has given libraries new choices in meeting the information needs of their users (Pace, 2004). Thus, different models of development such as OSS can be appealing to libraries, which, like much of the IT world, are moving toward the greater adoption of OSS (Dietz & Grant, 2005). Moreover, on an ideological level, OSS, as a purportedly democratic and grassroots technology movement, has symbolic and practical appeal for libraries trying to wrest control away from commercial vendors. Libraries are taking up OSS as a way to reduce the costs of expensive

commercial products and to take ownership over their own technology development. In addition, it has been argued that the library profession's values line up with those of OSS (Frumkin, 2002).

OSS potentially allows libraries to contribute to software development, which can empower libraries and bring library values to software (Frumkin, 2002). The March 2002 issue of *Information Technology and Libraries*, in fact, is dedicated to examining the possibilities for OSS in libraries. One of the major themes in this issue is that OSS offers opportunities for resource sharing and for libraries to take more control of their technology situations (Frumkin, 2002). In addition, the open source movement allows for libraries to contribute to technology development, pool resources, and save time and money (Frumkin, 2002).

Library Open Source Communities and Challenges

Active communities focused on library open source projects also exist—as mentioned earlier, Oss4Lib and Code4Lib are two major Web sites serving as clearinghouses for library-based open source projects. Dan Chudnov, a prominent library open source leader, started Oss4Lib in 1999 and maintains it. The stated mission of Oss4Lib is to build “better and free systems for use in libraries,” and the site maintains a listing of free software designed for libraries and tracks news about related issues of interest (Oss4Lib, n.d.). The Oss4Lib mailing list and Web site, dedicated to OSS in libraries, examines these issues in more detail and is an active community dedicated to finding open source solutions for libraries. In terms of library-specific open source products, over 100 of these have been announced on the Oss4Lib Web site (<http://oss4lib.org>), but all fall into these basic categories:

- metadata tools
- protocols
- OPAC/ILS (integrated library systems)
- repositories
- public services tools (e.g., library reserves applications)
- bibliographic management
- information retrieval (Chudnov, 2006).

Code4Lib is a related Web site, and a Code4Lib annual conference now takes place, described as a “loosely structured conference for library technologists to commune, gather/create/share ideas and software, be inspired, and forge collaborations” (Code4Lib, n.d.).

Despite the promise of OSS in libraries and some of its successes, some important challenges remain. One major challenge is the lack of technical skills among staff members in many libraries (Clarke, 2000). Larger libraries

with skilled systems staff may have the requisite technical expertise; however, many smaller and less financially robust libraries face daunting technical challenges. In addition, the development of a significant library community around open source projects remains a concern, especially since the development of a user and programmer community remains central to the success of open source projects (Clarke, 2000).

While communities such as Oss4Lib and Code4Lib exist, it is still an open question if enough skilled and committed programmers exist in the library community to sustain larger library-specific open source projects. On another note, a main challenge and key to the future success of open source in the library community depends on a shift in budgeting priorities (Clarke, 2000). Specifically, much of the financial resources expended on commercial vendors would have to be reallocated for investment in staff. More technical staff members could be hired, or this investment could take the form of technical competency building for existing staff.

Some of the advantages for libraries in adopting OSS are little or no upfront costs; however, proper expertise is needed to modify the code to meet local practices or requirements (Muir, 2005). In addition, open source products can develop faster because there are multiple sites working on enhancements, developers are usually closer to the end user, and troubleshooting is spread across a large number of sites (Muir, 2005). However, potential drawbacks to OSS include the issue of who actually provides support, needs for technical expertise, and the hidden costs that go into having staff spend time supporting, tailoring, and enhancing software (Muir, 2005). Regardless, OSS can benefit libraries by lowering initial and ongoing costs, eliminating vendor lock-in, and allowing for greater flexibility (Corrado, 2005).

OSS, in conjunction with open access (OA) and open standards movements, can be beneficial to libraries in the long run. For instance, open source and open standards can help libraries provide patrons with easier access to OA materials and other resources, as open standards make it possible to create interoperable systems to access the literature in various OA journals seamlessly (Corrado, 2005). Though there may be no reason to fear open source for libraries, the choice of open source software depends on whether or not the product meets a library's automation needs as well as its support needs (Balas, 2005).

OA can take many forms, and OA electronic publishing is often cited as an example. Internet technologies have allowed the wide dissemination of scholarly research—allowing libraries, scholars, and publishers alike to envision models of scholarly publication. OA literature is digital, online, free of charge, and free of most copyright and licensing restrictions, and what makes it possible is the Internet and the consent of the author or copyright holder (Suber, 2004). OA is compatible with peer review and is not free to produce—it is not focused on whether scholarly literature can be made costless but whether there are better ways to pay the bills than by charging

readers and creating access barriers (Suber, 2004). The two most common forms of OA are OA repositories and OA journals. While it is a topic that has been gaining momentum in recent years, OA represents a growing consciousness around the need to make knowledge and information as widely accessible as possible (Willinsky, 2006). In fact, the very possibility for OA has been greatly enhanced by the presence of digital technologies (Willinsky, 2006). *Open standards* refers to protocols such as those proposed by the Open Archives Initiative (OAI), which provide for greater interoperability in standards to facilitate enhanced retrieval of OA and online materials.

Library Open Source Projects

With a sense of the open source terrain in the library community, it is useful to take a more in-depth look at a few prominent projects. This section will highlight the important institutional, technical, and economic factors that maintain these projects. Also, to analyze and compare different library open source projects, several factors will be considered, including: (1) software application, (2) funding/economic structure, and (3) management/development structure.

In terms of software application, the open source movement has the potential to make great changes in libraries if it could produce an integrated library system (ILS) that earns a level of acceptance on the same order that Apache did in the Web server market (Breeding, 2002). The ILS is an essential component of library operations, and a viable open source system will go a long way towards making OSS a more realistic option in library software development. Libraries could potentially play a more active role in the development of technology services to the public if more ILS software is open source. The Koha ILS software (Koha, 2005) is one of the major ILS open source projects currently existing, but is yet to have wide acceptance. I will discuss shortly some of the specifics about another prominent ILS open source project, Evergreen. A company called LibLime, whose stated mission is to make OSS available to libraries (LibLime, 2006), provides support services for libraries planning to use the Koha and Evergreen open source ILS software. This example illustrates the “give away the recipe, open a restaurant” model (Raymond, 2001, p. 136) and shows just one possible adaptation of an open source management style for libraries.

The role of library institutions in the development of software varies from minimal to active. While all projects are nominally open source, it is not always the case that development takes place as part of a larger development community, as it is often limited to a small, localized core of developers. In addition, basic challenges exist surrounding the nature of the technical support structure of library-based OSS projects. For instance, will support take place mainly through library-managed listservs and bug reporting Web sites, or will other spin-off entities provide support? However, despite these challenges,

basic open source philosophical viewpoints and development realities tie many of these projects together.

I will look briefly at three library open source projects in more detail: Evergreen, MyLibrary, and Simon Fraser University Library's reSearcher and PKP projects. These projects have been chosen because of their prominence in the library open source community. In addition, the wide variety of these projects illustrates the different types of applications, economic structures, and management/development structures present in library open source projects. Evergreen is an open source ILS developed by the Georgia Library PINES consortium. MyLibrary developed originally out of the North Carolina State University Libraries and is focused on creating library-specific Web portals. Simon Fraser University Library's open source projects deal with both library-specific applications, as well as one of the world's leading open source journal publishing platforms.

Evergreen—Georgia Library PINES Program

The Evergreen project is one of the more ambitious open source ILS projects, and for this reason it has garnered attention in the library community. The Georgia Public Library Service is developing this open source ILS for use by the Georgia Library Public Information Network for Electronic Services (PINES) Program, a consortium of 251 public libraries (LaJeunesse, 2006). The goal for this project is to have a statewide integrated library system for the wide variety of public and academic libraries in Georgia. According to Brad LaJeunesse (personal communication, September 25, 2006), one of the leading figures in this project and a PINES system administrator, Evergreen was conceived because no product in the marketplace existed that fit the needs of PINES. The major requirement for PINES is having software that not only enforces uniformity but also allows for a certain level of local control and administration (LaJeunesse, 2006). This type of flexibility allows the software to be used across the diverse types of libraries in Georgia.

LaJeunesse (personal communication, September 25, 2006) also comments that the software meets the needs of PINES because it is designed by PINES system administrators and librarians. This statement relates to the issue of enhanced customization of OSS. He adds that the development process has included focus groups and discussions with librarians. In an e-mail correspondence, LaJeunesse explained to the author how the librarians decided on the functionality of the software, and how the entire development process has incorporated feedback from librarians (LaJeunesse, personal communication, September 25, 2006). LaJeunesse (personal communication, September 25, 2006) believes that internal support and management have been key to the project's success. In addition, he points to the presence of a "wonderful and motivated staff" and a "top-notch software development team" as important

factors in sustaining the project (LaJeunesse, 2006). The Evergreen project, while at an early development stage and dealing with a large and complicated application of an ILS, nonetheless has had its successes. LaJeunesse, in fact, believes that no major setbacks have occurred with the project, but the greatest challenge will be adoption on a larger scale. As other users outside the PINES consortium begin using Evergreen, more opportunities for outside code development and feedback can occur.

Returning to our three categories for analyzing library open source projects—software application, funding/economic structure, and management/development structure—Evergreen presents an interesting case. The ILS application of *Evergreen* is ambitious—many in the library community remain skeptical over the development of an open source ILS. This skepticism is fueled by the fact that the complexity of library automation systems often exceeds the pool of programmers, and many volunteer programmers often do not have the time allotment, project management infrastructure, and other resources needed for the concerted development efforts required to build and maintain an ILS (Breeding, 2002). However, Georgia PINES has been able to release Evergreen, and this points to two major factors in its apparent success—its funding structure and management structure. A consortium is funding its development, and it enjoys the full support of management. While outside developers are not contributing much code yet (LaJeunesse, personal communication, September 25, 2006), the project appears to have a viable management and economic structure for meeting the needs of its member libraries.

MyLibrary

The MyLibrary project is a Web portal designed especially for libraries. MyLibrary is a user-driven, customizable interface for collections of Internet resources, and its purpose is to reduce information overload by allowing patrons to select as little or as much information as they desire for their personal pages (MyLibrary, 2005). The idea of customization for the user drives this project, and the project was first conceived in 1997 at the North Carolina State University (NCSU) Libraries. A driving force behind this project is Eric Lease Morgan, with whom I corresponded via e-mail. As a librarian and also the lead programmer in this project, Morgan and two other librarians at NCSU saw the emergence of personalized services such as MyYahoo during the peak of the dot-com boom and decided to extend this idea to the library realm (E. L. Morgan, personal communication, October 3, 2006). Morgan was part of the Digital Library Initiatives Department at the library, and believes the “forward thinking” nature of the library allowed them to develop MyLibrary primarily as a set of services for users (Morgan, personal communication, October 3, 2006). Thus, rather than focusing on developing collections, this project is primarily geared towards user services.

The success of the project is seen in a number of libraries that are using the software—some notable examples include Cornell University, Los Alamos National Laboratory, and Open University. Morgan (personal communication, October 3, 2006) also discusses that a fair number of MyLibrary “imitators” exists, and that imitation is the “sincerest form of flattery.” The development of the product is continuing apace, with new versions being released on a regular basis. However, Morgan no longer works at NCSU Libraries and is now based at the University of Notre Dame. When he left NCSU Libraries, it was decided that the copyright for the software would remain with NCSU (Morgan, personal communication, October 3, 2006). Morgan remains the lead programmer on the project, and primary development is now based at the University Libraries of Notre Dame.

Despite the successes of the MyLibrary project, certain challenges are notable and ongoing. According to Morgan (personal communication, October 3, 2006), the lack of computer programming expertise in the library community is a pressing concern. MyLibrary is not an easy piece of software to download and install without adequate programming skills, and technical support is maintained through a mailing list Morgan oversees (Morgan, personal communication, October 3, 2006). Morgan believes MyLibrary allows librarians to take greater control over their computing environments, but he does not think enough librarians understand these technologies and are thus unable to fully take advantage of them. The time needed for ongoing development of the software remains a challenge—Morgan, as the main programmer, has to divide his time spent on MyLibrary with his other responsibilities at the University Libraries of Notre Dame.

In my correspondence with Morgan, he did not specifically mention growing the outside developer community as a major challenge, but it appears that a wider development community could result in a faster development cycle. The MyLibrary experience also points to an important fact about much of OSS development—programming time is often given on a volunteer basis, and challenges exist in balancing this volunteer work with other job demands. The initial management structure of MyLibrary at NCSU Libraries, however, gave the project an official status beyond a volunteer project, but sustaining this project beyond the original development team would require other library-based programmers to dedicate their time and expertise to the project. Although, as Morgan discusses, the level of programming skills within the library community remains fairly low.

Simon Fraser University Library: reSearcher and PKP

Simon Fraser University (SFU) Library in British Columbia, Canada has been developing OSS for several years. SFU Library has been an active developer and advocate for OSS solutions for libraries since the mid-1990s, and the library develops, supports, and coordinates the development of two software

suites, the reSearcher suite and PKP suite (SFU, 2005). Central to both of these projects has been the support and leadership of key administrators. SFU Library developed and implemented the reSearcher suite, an award-winning integrated set of open source tools for locating and managing electronic information resources, designed for use by students and researchers in academic libraries (SFU, 2005). The components of reSearcher are Citation Manager, CUFTS, GODOT, dbWiz, and the CUFTS Knowledgebase, and they were developed with the support of the Council of Prairie and Pacific University Libraries (COPPUL) and the British Columbia Electronic Library Network (BC ELN; SFU, 2005).

The PKP software suite, on the other hand, is OSS that supports scholarly publishing and communication. In contrast to the reSearcher suite, the PKP suite is a development partnership with the Public Knowledge Project (PKP) at the University of British Columbia (SFU-UBC). The PKP suite components are Open Journal Systems, Open Conference Systems, and the PKP Open Archives Harvester. Open Journal Systems (OJS) has been adopted worldwide as an online publishing platform by hundreds of scholarly online journals (SFU, 2005). OJS has been recently recognized as a Scholarly Publishing and Academic Resources Coalition (SPARC) Leading Edge Project, and has received funding from Canada's Social Sciences and Humanities Research Council (SSHRC), the Max Bell Foundation, the Soros Foundation, the International Network for the Advancement of Scientific Publishing (INASP), and the MacArthur Foundation (Synergies Project, 2006).

The Public Knowledge Project (PKP) at the University of British Columbia is the original developer of this software package, but a memorandum of understanding signed on January 14, 2005 made SFU Library the home for future PKP software development. This memorandum of understanding affirms the "SFU-UBC Partnership for Open Source Publishing Software Development." In this agreement, the SFU Library and SFU's Canadian Centre for Studies in Publishing (CCSP) agreed to enter into a partnership with UBC's PKP to support the maintenance and ongoing development of Open Journal Systems (OJS), Open Conference Systems (OCS), and the PKP Harvester (PKPH; SFU-UBC, 2005). This joint venture will involve providing a permanent home for this suite of OSS in the SFU Library (SFU-UBC, 2005). A major portion of SFU Library's activities is in managing the development of the OJS, OCS, and PKPH software, with SFU Library Systems staff taking on this responsibility. SFU Library is committing Can \$21,000 annually to provide ongoing systems management and support for the software suite and will work both with PKP and CCSP to coordinate and support efforts to apply for research and development grants to ensure continuing support and development of PKP software (SFU-UBC, 2005).

The PKP is a leading voice in open source and OA models in scholarly publishing. The PKP is a project funded by the Canadian federal government,

which is committed to expanding the realm of public education by improving social science's contribution to public knowledge (PKP, 2007a). PKP is involved with many activities, including major grant-funded activities related to increasing the research capacities of developing nations, as well as the development of prototype Web sites in collaboration with partners, with a focus on integrating research resources with more public information sources and more interactive environments (PKP, 2007a).

The OJS software has the potential to reduce the time and energy devoted to the clerical and managerial tasks associated with editing a journal while improving the record-keeping and efficiency of editorial processes (Willinsky, 2005). In addition, while not necessarily promoting OA publishing, OJS has the ability to facilitate OA to scholarly information. The community of journals deploying OJS continues to grow, with over 140 registered users on the PKP Support Forum worldwide (Willinsky, 2005).

OJS has been a successful open source product, with several hundred journals using this software (PKP, n.d.). Much of the user base for OJS, in fact, comes from the developing world, with over 200 journals in Africa using the OJS software through the African Journals Online program (PKP, n.d.). This large amount of uptake in the developing world is not surprising given the economic challenges of accessing commercially controlled scholarly information in that part of the world. The open source nature of the product (free to download) certainly makes it an attractive product for users, as traditional corporate models of scholarly publishing can be bypassed.

The management of the reSearcher and PKP software projects differ—most reSearcher products rely on some level of support from consortia. PKP, on the other hand, depends on research grants and will benefit from a nationwide grant of \$5.8 million (Can) for the Synergies Project, funded by the Canada Foundation for Innovation (PKP, 2007b). SFU Library presents the case of two successful open source projects with quite different applications—one a regional, library-based set of applications, the other being an internationally well-regarded open source journal publishing application. The continued growth of these projects will depend on the growth of the developer communities for these projects and the long-term sustainability of funding models. However, the OJS project is starting to develop wider acceptance and growth as the major open source journal publishing software. For instance, the First International PKP Scholarly Publishing Conference took place in July 2007 in Vancouver, bringing together a range of international participants with an interest in OJS, as well as a concern for improving access to research and scholarship on a global scale (PKP, 2007c).

BEST PRACTICES IN LIBRARY OPEN SOURCE PROJECTS?

This review of a few prominent library open source projects has outlined some of the challenges these projects have in common, and some important

differences. Are there lessons we can begin to draw out in order to understand library open source best practices? For instance, a common theme that arises in these projects is leadership, both on an individual and organizational level. As Eric Raymond (2001) correctly observes, the initial development of an open source project often depends on the vision of an individual, but it will only be successful through sustained effort at community development. Community building is thus an important theme in developing a set of best practices for library open source projects.

Building community includes both users and code contributors—thus far, it appears for the projects I have discussed that code contribution is not occurring beyond the initial development group. However, user communities appear to be growing, especially in the case of SFU Library's OJS project. It remains an open question if, as user communities continue to grow, developer communities will grow along with them. Developing a strong community of users and developers can help ensure the technical sustainability of open source projects.

On the theme of sustainability, the financial viability of projects remains important. Different models of sustainability are presented in these examples, from consortium-funding models to foundation support, individual library support, and international agency support. Financial sustainability is also linked to the management structure of the projects. Developing financially sustainable library open source projects may require one of these models or a combination of them. Finally, establishing the means for technical support is also an important issue. For instance, will an outside company (as in the case of LibLime) provide technical support, or will project coders do it? In addition, it needs to be determined which services will be free and which services will have a fee associated with them.

Any set of best practices will thus need to include the various issues of funding and management structure, leadership, code development, community building, and technical and financial sustainability. These issues are not necessarily particular to libraries and affect other service institutions seeking to utilize OSS, but nonetheless they need to be resolved if libraries are to be successful in developing and maintaining OSS projects.

OSS IN LIBRARIES: A POLITICAL AND ACTIVIST MOVEMENT?

With a discussion of a range of technical, management, and economic issues surrounding OSS in libraries, I end with some reflections about the political and activist nature of this movement in libraries. As I mentioned earlier in this chapter, it is useful to ask how the library open source movement is tied into larger grassroots struggles in favor of an information commons and against the increasing commodification of information. A useful framework to begin addressing these questions is to look at the various levels at which advocacy

and activism can take place, including: (1) policy, (2) individuals and communities, and (3) systems and institutions (Pyati, 2007).

On the policy level, it is important to understand the progressive and democratic orientations of the open source movement. For instance, is the open source movement in libraries linked to larger advocacy agendas for greater library-based control of technology? Does a consciousness of the political and grassroots democratic aspects of the open source movement exist in libraries, or are more practical reasons driving the movement? Are policies being enacted at institution- and profession-wide levels to promote open source in core functionalities, such as integrated library systems?

On the individual and community level, one has to question the participatory and community-oriented aspects of the movement. Specifically, does open source create a more participatory technology development process in libraries? Does technical expertise become more democratized, or do new technical hierarchies develop? Finally, on the systems and institution level, it is useful to explore how open and collaborative the open source process in libraries is. Does open source build on existing library strengths of resource sharing and cooperation? Does it challenge hierarchical models of technology development?

Applying these questions to the library open source context remains crucial to moving beyond the discourse of open source as a merely practical and cost-effective alternative to proprietary software development. The examples I have shown in this chapter offer a range of successes for libraries in terms of reduced cost, and increased customization, and control, but the ideological underpinnings for many of these projects, with the exception of OJS, do not appear to be activist or political in character. Strong links to *radical*, grassroots conceptions of open source and free software are generally lacking in these examples and in the wider library open source community as well. Is this fact important? Perhaps not. OSS, in fact, may not be quite the democratizing technology that many of its fervent advocates claim it is. However, OSS and the free software movement provide an opportunity and opening for libraries to re-envision alternatives to the dominance of corporate, capitalist modes of software development. Libraries are taking part in these movements, but more work needs to be done in understanding how to sustain these projects and in theorizing a larger political voice for libraries in debates about technological democratization.

REFERENCES

- Apostle, R., & Raymond, B. (1997). *Librarianship and the information paradigm*. Lanham, MD: Scarecrow Press.
- Balas, J. L. (2005). There's no need to fear open source. *Information Today*. Retrieved October 5, 2007, from <http://www.infotoday.com/cilmag/may05/balas.shtml>.

- Benkler, Y. (2003). The political economy of commons. [Electronic version]. *Upgrade*, 4(3), 6.
- Bollier, D. (2003). The rediscovery of the commons. [Electronic version]. *Upgrade*, 4(3), 10.
- Breeding, M. (2002). The open source ILS: Still only a distant possibility. *Information Technology and Libraries*, 21(1), 16–18.
- Breeding, M. (2006). Reshuffling the deck. *Library Journal*, 131(6), 40–54.
- Breeding, M. (2007). An update on open source ILS. *Computers in Libraries*, 27(3), 27–29.
- Chawner, B. (2006). *Open source software and libraries bibliography*. Retrieved December 18, 2006, from http://www.vuw.ac.nz/staff/brenda_chawner/biblio.html.
- Chudnov, D. (2006). *The future of FLOSS in libraries*. Unpublished book chapter manuscript.
- Clarke, K. S. (2000). *Open source software and the library community*. Unpublished master's thesis, University of North Carolina at Chapel Hill.
- Corrado, E. M. (2005). The importance of open access, open source, and open standards for libraries. *Issues in Science and Technology Librarianship* (Spring 2005). Retrieved May 10, 2006, from <http://www.istl.org/05-spring/article2.html>.
- Courant, P. N., & Griffiths, R. J. (2006). *Software and collaboration in higher education: A study of open source software*. Ithaca, NY: Organization for Open Source Software Study. Retrieved July 24, 2008, from http://www.ithaka.org/strategic-services/oss/OOSS_Report_FINAL.pdf.
- Dietz, R., & Grant, C. (2005). The dis-integrating world of library automation. *Library Journal*, 130(11), 38–40.
- Frumkin, J. (2002). Guest editorial: Balancing the playing field. *Information Technology and Libraries*, 21(1), 2.
- Koha. (2005). *About Koha*. Retrieved December 5, 2005, from <http://www.koha.org>.
- LaJeunesse, B. (2006). Evergreen: Built for a consortium. *Library Journal.com*. Retrieved October 5, 2007, from <http://www.libraryjournal.com/article/CA515806.html>.
- LibLime. (2006). *Our mission—LibLime*. Retrieved December 27, 2006, from <http://liblime.com>.
- Muir, S. P. (2005). An introduction to the open source software issue. *Library Hi Tech*, 23(4), 465–468.
- MyLibrary. (2005). *MyLibrary@NCState*. Retrieved December 21, 2006, from <http://my.lib.ncsu.edu/>.
- Oss4Lib. (n.d.). *Oss4Lib: Open source systems for libraries*. Retrieved November 10, 2006, from <http://www.oss4lib.org>.
- Pace, A. K. (2004). Dismantling integrated library systems. *Library Journal*, 129(2), 34–36.
- Public Knowledge Project. (2007a). *About the Public Knowledge Project*. Retrieved October 5, 2007, from <http://pkp.sfu.ca/node/4>.
- Public Knowledge Project. (2007b). *Major Canadian initiative for PKP*. Retrieved October 5, 2007, from <http://pkp.sfu.ca/node/695>.
- Public Knowledge Project. (2007c). *PKP Scholarly Publishing Conference 2007*. Retrieved July 23, 2008, from <http://ocs.sfu.ca/pkp2007/index.php>.

- Public Knowledge Project. (n.d.). *Open Journal Systems—publications*. Retrieved October 5, 2007, from http://pkp.sfu.ca/pkp_publications.
- Pyati, A. K. (2007). *Re-envisioning libraries in the information society: A critical theory of library technology*. Unpublished dissertation, University of California, Los Angeles.
- Raymond, E. S. (2001). *The cathedral and the bazaar*. Sebastopol, CA: O'Reilly.
- Simon Fraser University. (2005). *Software@SFU library*. Retrieved November 30, 2005, from <http://www.software.lib.sfu.ca>.
- Simon Fraser University, University of British Columbia. (2005). *Memorandum of understanding: SFU-UBC partnership for open source publishing software development*. Unpublished manuscript.
- Stallman, R. (2002). *Free software, free society: Selected essays of Richard M. Stallman*. Boston: GNU Press.
- Suber, P. (2004). *A very brief introduction to open access*. Retrieved September 29, 2006, from <http://www.earlham.edu/~peters/fos/brief.htm>.
- Synergies Project. (2006). *Synergies—CFI application—project 12112*. Unpublished report.
- Weber, S. (2004). *The success of open source*. Cambridge, MA: Harvard University Press.
- Willinsky, J. (2005). Open journal systems: An example of open source software for journal management and publishing. *Library Hi Tech*, 23(4), 504–519.
- Willinsky, J. (2006). *The access principle*. Cambridge, MA: MIT Press.