Chapter 2

The Commercial Angle

Open source software does not preclude commercial activity. This software generally uses a different business model than traditionally licensed software. The traditional arrangement for a company involves a license with terms that state the amount to be paid for the use of the software, plus additional fees to be paid annually for ongoing support and maintenance.

Companies involved in open source software focus on a different business model, based more on service and other values they can add to the environment. In an environment where libraries can obtain the software itself for free, the business opportunities are usually in providing services libraries will be willing to pay for.

In most cases, companies involved in open source software compete in an environment that includes no-cost options. Red Hat, for instance, bases its business largely on the Linux operating system, which anyone can download and use for free. It's possible to download the software, compile, and configure it for almost any hardware platform around, and yet it's a technical challenge beyond the everyday computer user. Red Hat bases its business on creating a version of Linux that can be easily installed and optimized for different needs, with strong security features and other appealing tools. More importantly, it comes with support. If something goes wrong, the company will provide any needed assistance. For businesses to rely on Linux for their critical infrastructure, they require a high level of confidence in reliability, performance, and quality of service.

The same issues apply in library automation. Any library can download and install open source ILS products like Koha and Evergreen without paying a penny. No library can be denied use of the software. If a company wants to earn revenue from the software, it must offer services that enhance the value of the software enough to provide an incentive for libraries to pay for them.

The services involved in the support of open source software might include some of these:

- **Conversion services.** Whether the process involves automating for the first time or migrating from an existing system, data must be prepared and loaded into the new ILS. For initial automation, retrospective conversion involves creating a database of bibliographic records that corresponds to the library's collections. For libraries with an existing automation system, the process of migration involves extracting all types of data from the incumbent system so that the data can be loaded into the new one. In the ILS arena, library standards like MARC21 ensure the ability to migrate bibliographic data. The ILS includes many other data components not covered as thoroughly by standards that require complex work for both the extraction and loading process.
- **Installation.** A company involved in the support of an open source ILS will offer the software ready to use and thoroughly optimized for the library's hardware. This saves the library from having to download, compile, and install the software and all its prerequisites.
- **Configuration.** Preparation of an ILS for the use of a library can be a complex and tedious process. An ILS is a highly parameterized system, and it's necessary to provide a complete description of the way that the library will use the system. This information must be properly coded into the configuration and policy tables of the system.
- **Training.** The ILS impacts almost every aspect of the operations of the library and requires almost all

staff members to be familiar with its use. Anytime a library changes to a new system, a large portion of the effort involves teaching library personnel how to use the components of the system necessary for them to carry out their work. One of the key services that can be provided by a company involved in an open source ILS involves training sessions delivered by experts in the use of the system.

- **Ongoing support.** Once the library puts the software into production, operations depend on its reliable performance. Any problems with the software can have a negative impact on the library's services. Support contracts provide the library with technical assistance and expert advice on the use of the system as needed. If the library discovers a bug in the software or if the software fails to perform in some way, the company providing the support is expected to resolve the issue. Without ongoing support, the library incurs a degree of risk that problems could arise that it could not resolve on its own.
- **Hosting.** Many libraries prefer the software-as-a-service (SaaS) approach, where the vendor hosts the application. With SaaS, the vendor provides the hardware and assumes responsibility for the installation, configuration, and maintenance of the software. The business model for SaaS generally involves an annual subscription fee that covers the hosting and maintenance of the software. This approach saves the library from having to purchase its own hardware for the server and obviates the need for technical staff to perform system backups, server administration, security patches, and other tasks related to the technical upkeep of the ILS.
- **Custom development.** If the software does not have a specific feature that a library needs, it might choose to engage a vendor to enhance the software. In the open source model, these enhancements can be contributed back into the software so that other users also benefit.

We should note that these services correspond to those that we would expect from a vendor offering proprietary software. With proprietary software, some of these services may be bundled into the license fee, which authorizes the library to make use of the software.

Total Cost of Ownership

An important question related to the adoption of an open source ILS is whether it results in lower or higher cost to the library. The proponents of open source make claims that their approach results in substantial savings to a library over time. Companies involved with proprietary software dispute those claims. There may be no simple answer to that question. It's up to any library deliberating between open source and proprietary solutions to perform its own analysis of which approach is most cost-effective given the library's circumstances and expectations. Considering just the cost issues, setting aside philosophical preferences, quality, and functionality of software, it's important to work out the total cost of ownership for the solution for as many years as the library plans to use the software. Given that a typical library will operate an ILS for 10 to 15 years (provided that the company stays in business and the product remains viable), the total cost of ownership should be calculated over at least a 5- to 7-year period.

Table 3 describes some of the major categories of cost involved in the implementation and operation of an integrated library system, highlighting some of the factors that differentiate open source and proprietary systems.

Vendor/Product Independence

A common argument in favor of open source software is that it gives users more support options and less vulnerability to business transitions. In the proprietary software arena, if a company goes out of business or is acquired by a competitor, then the ongoing viability of that product can be jeopardized. Without the development and support of the original vendor, the product will stagnate, and the libraries that use it may have to migrate to another product. In the case of a business acquisition, the new owners of the software may or may not choose to continue development and support of the products involved. It also may not be possible for another company to step in to take over development and support. If the company that originally developed the software does not provide good service or decides not to continue developing or supporting it, the libraries using that software cannot go somewhere else. With propeitary software, a product remains closely connected with the company that owns and controls it.

In the open source realm, the connection between a software product and a given company is more flexible, at least in theory. Although a company may spearhead the development of a product, once the product is released as open source, a more diffuse body of programmers can become involved. If the company goes out of business, any other interested company, individual, or organization can step in and provide development and support. At any time, multiple companies can provide support for the same product. If a company fails to provide good support, the users of the software can engage another.

We see different examples of multi-vendor support in the open source ILS arena. Koha was originally developed

Cost	Open Source	Proprietary
Technical Infrastructure		
Personnel: server administrator	Neutral cost impact when comparing open source and proprietary options. Under either approach, some support contracts may include full support for server and operating system, obviating the need for a local server administrator. Cost = x FTE * Salary * total years of ownership.	
Personnel: applications programmer	Possible higher need with open source ILS.	Large organizations may employ an applications programmer to work with the ILS. Most small to mid-sized installations do not require an applications programmer to work with the ILS.
Server facilities: personnel, monitoring equipment, data center, racks, etc.	Neutral cost impact. For local installations of either open source or proprietary ILS, the library or its parent institution will need to house the ILS server in its data center. This may involve start-up costs and ongoing annual costs related to the personnel and operational cost of the data center. Total cost of these facilities involves any up-front costs in preparing the data center, plus an allocation for each year of ownership of the system that includes a portion of the cost of operating and staffing the data center.	
Server facilities: cooling, power	Significant costs in maintaining a local installation of either software model. Involves the cost of redundant uninterruptible power, cooling equipment, and energy costs. Must be factored in for each year of ownership.	
Licensing Fees		
License purchase for base system and required modules	Not applicable to open source software.	License fees generally assessed according to the size and complexity of the library.
License fees associated with prerequisite components (Oracle, etc.)	Most open source products do not involve licensed components.	License fees may apply to the operating system, database engines, and other components. These licenses fees may include both up-front costs and annual payments.
Software Support		
Software maintenance for access to upgrades and enhancements	No annual license fees, but no guarantees that new versions of the software will be produced.	A component of annual maintenance supports extensions of the duration of the license and access to new versions of the software. Must be factored for each year of ownership.
Software support for assistance with functional questions or to resolve software or hardware failures	Contracting with a vendor for support services is optional, but a practical necessity for most libraries. Given that most of the income to the vendor concentrates in this category, libraries may expect higher fees in this area to offset lack of licensing fees. Support costs must be factored for each year of ownership.	Annual support for support services bundled into annual fees. Is usually mandatory. Many vendors offer different service levels with corresponding pricing. Must be factored in for each year of ownership, including possible adjustments for inflation.
Systems librarian to manage the ILS: local support and training, policy maintenance, data loading, reports, customization, problem solving	Neutral cost impact. The need for a systems librarian is more a factor of the size and complexity of the library than the license model. Mid-sized to large libraries will employ a systems librarian who devotes significant attention to the ILS, among other responsibilities. Large libraries tend to have multiple systems librarians. Must be factored in for each year of ownership.	

Table 3
Cost factors: open source vs. proprietary

Cost	Open Source	Proprietary
Start-up Costs		
Retrospective conversion	For first-time automation projects, the library will need to build a database that represents its collection. One-time start-up cost.	
Data extraction from legacy system	Libraries converting from a legacy system will need to create routines that extract data of all types from the system: bibliographic, holdings, item, circulation transactions, patrons, orders, vendors, funds, system history, etc. One-time start-up cost.	
Data conversion	Data from the legacy may need to be transformed into a different format for the new system. One-time start-up cost.	
Software installation	The installation of the software onto the library's hardware can be performed by the library itself, or it may include this task as part of a contract of services from a vendor.	While some proprietary vendors allow self-installation, most require the installation to be performed by its authorized representative, which is often covered in the software license fee.
ILS policy configuration	The process of fully configuring the ILS with all of the policies and preferences of the library is proportional to the size and complexity of the library and can represent a large investment of personnel time. Both proprietary and open source vendors offer support options to take on more of this work and reduce the library's time investment.	
Testing / acceptance	Neutral cost impact. The typical installation process involves significant testing of the system by staff throughout the library to ensure that the system functions as expected prior to making the transition to production use.	
Training		
IT and systems staff for technical maintenance	IT personnel and library systems staff must be trained on the technical operation of the software, including configuration and customization details, system tuning options, backup procedures, report creation, diagnostics, troubleshooting, and problem resolution. Mostly a neutral cost impact. Either licensing model may involve options that reduce the library's technical involvement in operating the ILS.	
Library staff members on functional modules appropriate to their job responsibilities	Neutral cost impact. Regardless of licensing option, all staff must receive detailed and extensive training on how to operate the system for the functions within their areas of responsibility. Vendors involved in both licensing models offer training services. Libraries should also plan for training in subsequent years of operation for new staff members and for any necessary refresher courses.	
Software-as-a-service	Vendors involved with both licensing options offer their respective products in a software-as-a-service model, in which the vendor hosts the software. Libraries access the software via the Internet. This model dramatically alters many of the above cost considerations. SaaS generally involves a monthly or annual fee that covers a large portion of the costs mentioned above. For proprietary systems, the monthly SaaS subscription fee will also include the licensing fees, and may obviate the large up-front license payment. The SaaS option involves a higher monthly cost at the trade off of reducing or eliminating many of the up-front costs.	

Table 3
Cost factors: open source vs. proprietary *(cont.)*

by a company in New Zealand, Katipo Communications. In the United States and Canada, LibLime markets and provides services for Koha. In France, a company called BibLibre provides Koha-related services. Other companies are emerging in other geographic regions. To date, there are no major examples of multiple companies competing within the same region for the support of an open source ILS.

In practice, each of the companies involved in open source ILS focuses its attention on a single product and generally has fairly exclusive involvement relative to that product. It remains possible that we will see additional competition for support for open source ILS products as this sector of the market expands and matures.

In some ways, the business dynamics of open source vendors is similar to that of proprietary vendors. The geographic distribution of companies offering support for Koha is not unlike the arrangement seen with many vendors of proprietary software, who partner with other companies to market and sell their products in specific countries or regions.

Where regional distributors for a proprietary ILS are arranged through contracts that that assign a company

specific rights regarding a product—like marketing and support within a defined area—the proliferation of companies supporting an open source product occurs more informally. If a company intends to become involved in providing services for a given open source product, it does not need to obtain permission to do so.

We have seen some examples where support for a product has shifted from one company to another as the result of business transitions. In February 2007, LibLime made an agreement with Katipo Communications to assume the portion of its business related to Koha, including the support arrangements for the libraries in that region. In July 2008, CARE Affiliates was acquired by LibLime. The contracts made by CARE Affiliates, primarily involving federated search implementations using open source components from Index Data, were assumed by LibLime.

So far, there are no major examples of libraries using an open source ILS and demonstrating vendor independence by moving from one company to another for support. There have been libraries using open source products whose support arrangement was transferred from one company to another as a result of a business transaction. These transfers of support happen regularly in the proprietary ILS arena.

An important part of this issue involves the support and development options that exist with open source software that are not possible with proprietary software. A library can use an open source ILS without direct involvement from any commercial company, and if it wants to contract for support, the library is not forced to work with any specific vendor.

Contracting for services and support for an open source ILS is optional. Many libraries have implemented Koha independently, both within the United States and internationally, including the Delhi Public Library in India, the Nelson Memorial Public Library in Samoa, the Bering Strait and Chinook school districts in Alaska, a group of schools associated with the Southwest Educational Development Center in Utah, the Washington County School District in Utah, and many others.

If the library needs additional assistance beyond what is available within its existing staff, it has many options. It could simply hire its own programmer, or it could contract with a commercial company. Although each of the products has ties to a specific company that specializes in its development and support, a library can hire or contract with anyone it chooses for any services that it might require.

Collaborative Development

With a proprietary ILS, the company that owns the product controls its development. In most cases, development is performed by direct employees of the company. There have been some examples where an ILS company has engaged services from a third-party offshore firm for development, but this has been fairly rare. In general terms, development of a proprietary software product is a closed process.

Open source software allows and even encourages wide participation in ongoing development. Since anyone can gain access to the source code, other programmers can inspect the code, fix bugs, or extend its functionality.

In the current open source sector of the ILS industry, there are companies specializing in services for a given ILS that employ programmers to actively develop their product. Of course, not all development takes place in these companies. Programmers who work for libraries, for other companies, or out of their own interest also contribute to the development of open source ILS products. The open source movement encourages voluntary development efforts, where organizations contribute the equity of their efforts to improve software initially for their own benefit, but ultimately for all users of the software.

Sponsored Development

In the open source ILS arena, the majority of the development of products is performed by their principal support company. One of the key strategies that leads to the improvement of the software involves "sponsored development." As a general principle, each new feature that gets added to the product needs to be paid for only once.

The model of sponsored development has driven the advancement of the open source ILS products. The basic premise holds that open source software components can be paid for only once for the benefit of many. Each library that subsequently adopts the software benefits from the accumulation of features that were sponsored previously. Early adopters bear a larger portion of the costs and assume higher risks, but to the advantage of those that implement the software in a more finished and complete form.

Notes

1. See LibLime press release dated July 28, 2008: http://www.librarytechnology.org/ltg-displaytext.pl?RC=13424. Also based on communications with company officials of CARE Affiliates and LibLime.