Chapter 1

Introduction

This issue of *Library Technology Reports* takes a look at the open source alternatives available to libraries for their strategic systems. Libraries depend on their core technology systems for managing and providing access to their collection resources and other aspects of their operations. These core systems include integrated library systems (ILSs), the longstanding model of library automation originally designed for managing print collections, and the newer genre of library services platforms that has gained favor with academic and research libraries.

These strategic systems have been sold to libraries primarily as proprietary products controlled by a single vendor. Under this proprietary model, libraries that use these products remain dependent on that vendor for the ongoing development of the software, for addressing any problems with the system, and for support or other types of services. Proprietary software will likely have many different configuration options but may have limitations in the ways that it can be customized to address library-specific issues. The costs with proprietary software may also strain library budgets.

Despite these limitations, proprietary software continues as the dominant approach for strategic library systems, and most libraries express relatively high degrees of satisfaction with their systems and their associated vendors.[1]

Open source products have been created that present alternatives to the proprietary systems. These open source products include some that are well established and widely deployed and some that have more recently emerged, as well as ones still in development. These products add an important dynamic to technology products generally dominated by proprietary software. Both open source and proprietary software comes with advantages and disadvantages, which we will explore in the report.

This report aims to provide a current look at the major ILSs and library services platforms and the impact they have made on the broader library technology industry. The intended audience includes those responsible for shaping the technology strategies or involved in evaluating and selecting systems. The discussion is not intended as highly technical and should be accessible to administrators as well as those more directly involved in systems. Persons involved in the development of open source systems will hopefully benefit from the background and perspective provided, but will likely not learn new technical details.

Only a small number of products fall within of the scope of this report. Products discussed include Koha, the original open source ILS, which has come to be one of the most widely deployed library automation products in the world; Evergreen, an open source ILS created primarily for consortia of public libraries; Invenio, an open source platform created by CERN that has recently been made available through its spin-off company TIND; and FOLIO, a new open source initiative to create an open source library services platform.

Other types of open source software have not been addressed in this report. Product categories such as discovery interfaces, content repositories, and a bevy of specialized tools and utilities also have open source options, which deserve their own detailed treatment and are not described here in detail.

This report takes open source software as an objective alternative to proprietary products. In other words, the technology products discussed are not considered more or less desirable based on whether they are open source or proprietary, but on the merits of

their functionality, how well they have served the needs of libraries that use them, and other tangible factors.

The report also does not aim to provide side-by-side comparisons of features and technical characteristics among the various products discussed or with the proprietary options. Some discussion of the general approach of each product is provided to give a general impression of its capabilities and suitability to libraries of different types. Those interested in more in-depth information can consult with the documentation provided for each of the products discussed on their respective websites.

The story of open source in the library technology scene continues to unfold. In some chapters, the scene has been set already. ILSs based on open source software have seen continuous rise in popularity since about 2000 and can now be seen as well within the mainstream. New chapters in the saga are being written even today, with new characters and unexpected turns in plot. The launch of the FOLIO initiative to create a new open source library services platform represents a major twist. In a corner of the industry where Ex Libris has established a solid foothold and seems positioned to extend its dominance even further, a new project launched by its archrival EBSCO has taken aim to disrupt its course. Will this new open source upstart alter the landscape in the long term? The answer lies in the next chapter yet to be written.

Open Source Software Issues

Open source software is not a new aspect to the library technology environment. Its principles are well understood, as well as the relative advantages and disadvantages of its development models and support arrangements. There is a large body of literature and documentation delineating the issues and details of open source versus proprietary software licenses. This section discusses some of the issues relative to the processes typically involved in open source software development and the ecosystem of services surrounding its implementation and support that have been seen in the open source ILS arena.

The tasks associated with the development of software remain much the same regardless of whether the product will be commercially licensed or released as open source. The manner in which those tasks take place or are managed, however, may differ substantially. Open source software tends to be much more oriented to collaboration among a distributed group of developers, often working for different organizations and in different geographic regions. This distributed method fosters a more transparent process since the development takes place more publicly and less behind the firewall of a single development firm.

Proprietary software development can often be conducted with a certain level of efficiency through more direct management of resources and processes. A company creating proprietary software exerts complete control over the software development process. It employs technical architects, software engineers, programmers, quality assurance experts, product managers, domain experts, or other roles needed. Through employment contracts and internal oversight processes, a company can hire an appropriately sized development team and manage the quality and productivity of each step of the process. Companies often make use of outsourced services to reduce personnel costs. It is also possible for such a corporate development shop to produce open source software.

Open source projects generally follow a more collaborative and democratic approach. The open source projects related to library systems exemplify the processes of community development. A diverse group of companies involved in providing services and libraries that make use of the software collaborate out of mutual interest to create and improve the product. This spirit of collaboration and the general philosophy of open source software results in a much more public process. It also demands robust communications processes and tools for collaborative development. Most open source software development projects will make use of several types of collaborative tools, including the following:

- A public repository for storing source code, documentation, and other project resources. Ideally, these repositories will include advanced version control features. Many projects use GitHub.
- An e-mail distribution list. Most projects have one or more Listservs to distribute news and other types of communications appropriate for e-mail.
- Real-time messaging. It is common for open source projects to have an IRC or Slack channel for conversations among developers and other project participants.
- A website. The project website will provide general information, including links to all pertinent resources.

GitHub
https://github.com

Contrary to the stereotype where open source software is created by volunteers in their spare time, most projects are carried out by professional programmers employed by companies with a direct or indirect financial interest in the product. Open source library system projects will often include a mix of developers from commercial support firms and individuals

Table 1.1. Common open source infrastructure components. Source: Marshall Breeding, "Open Source Integrated Library Systems," *Library Technology Reports* 44, no. 8 (November–December 2008): 7.

Category	Open Source	Proprietary
Source code	Must be made available.	Usually held privately by developers.
Intellectual property	Subject to copyright.	Subject to copyright and commercial licenses.
License options	Must include terms consistent with open source principles.	Commercial license will specify conditions of use, fees, and responsibility of provider.
Software development	Collaborative community.	Directly managed development team.
Source code management	Stored in a public repository such as GitHub.	Stored in private repository.
Support	Internal, community, or commercially provided.	Commercially provided.

working in libraries that use the software. An open source software development project will rely on a variety of technical and nontechnical roles. These projects greatly benefit from persons who work in libraries lending their expertise in functional areas in overall product design, workflow optimization, specifications for feature requirements, testing, and documentation. Some libraries may also have programmers who work directly with the source code for the product.

The realm of technology increasingly is based more on revenues derived for services than from direct sales of software licenses. Categories of service fees include the following:

- custom software development
- implementation
- hosting
- help desk and support

The use of any software product involves costs, regardless of whether it is open source or proprietary. Open source software does not involve license fees but will require other types of investments or expenditures (see table 1.1). In some cases, an organization can be entirely self-sufficient and may be able to operate a product based on open source software on existing hardware and through the efforts of its own personnel. Such a situation is relatively rare. In most cases, especially in the realm of enterprise-class business software, organizations will contract with specialized firms for the services needed to operate the software with the levels of reliability and performance expected for critical business functions.

Open Source Resource Management Systems

The following chapters provide information on some of the major products used in libraries available as open source software. These products fall within the category of what can be termed resource management systems, which serve as the primary business process automation environment to support the core activities in a library. ILSs and library services platforms are two different types of resource management systems. ILSs are well established as automation systems that were originally created when library collections were primarily composed of print and other physical media. Although ILSs have evolved substantially, they continue to retain an orientation to physical materials. Examples of ILSs include the following:

- Symphony, a proprietary product from SirsiDynix
- Horizon, a proprietary product from SirsiDynix
- Millennium, a proprietary product from Innovative Interfaces
- Polaris, a proprietary product from Innovative Interfaces
- Library·Solution, a proprietary product from the Library Corporation
- Carl·X, a proprietary product from the Library Corporation
- Apollo, a proprietary product from Biblionix
- Voyager, a proprietary product from Ex Libris (A ProQuest Company)
- Koha, an open source product
- Evergreen, an open source product

Library services platforms embody a different overall model of resource management, addressing workflows for complex, multiformat collections, including electronic, digital, and print materials. These products are deployed on multitenant platforms, have web-based interfaces for all staff and public functions, and include built-in knowledge bases. Examples of library services platforms include the following:

- WorldShare Management Services from OCLC
- Alma from Ex Libris (A ProQuest Company)
- FOLIO, a new product still under development as an open source initiative

The dividing lines between ILSs and library services platforms are not precise. There are some

products with hybrid characteristics, including the following:

- SirsiDynix BLUEcloud, a multitenant platform providing web-based interfaces and extended functionality for libraries using Symphony or Horizon.
- Sierra, a product evolved from Innovative's Millennium product that includes some of the functional and technology characteristics of a library services platform though retaining much of the character of an ILS.

The next sections take a closer look at the open source resource management systems Koha, Evergreen, TIND, and FOLIO.

Note

1. Marshall Breeding, "Perceptions 2016: An International Survey of Library Automation," Library Technology Guides, January 25, 2017, https://librarytechnology.org/perceptions/2016.