

Smart Libraries Newsletter

News and Analysis in Library Technology Developments



225 N. Michigan Ave., Suite 1300, Chicago, Illinois 60601-7616, USA



Smarter Libraries through Technology

Community Development

By Marshall Breeding

Almost all libraries rely on technology systems to manage their collection resources and provide access to their users. It is hard to imagine libraries carrying out their work without the assistance of some type of computer system. The level of involvement that a library will devote to its key technology products varies.

In the early phase of library automation history, some libraries developed their own systems in-house. This approach gave the library full control of the functionality of the software but came with a high cost, which ultimately proved to be unsustainable. Commercial products eventually replaced systems developed locally at the University of Texas, the Oklahoma City Metropolitan Library System, Claremont Colleges, Duke University, the University of California at Berkeley, the University of Chicago, the University of Georgia, and many others. A few systems developed in-house were commercialized, such as NOTIS from Northwestern University Libraries, LIAS from Pennsylvania State University, and VTLS from the Virginia Tech University Libraries. Both Koha and Evergreen were developed as local library or consortium projects. They were released as open source and have seen widespread adoption.

Today's environment, however, is not favorable to the development of in-house library automation systems. The

requirements are far too complex, and the costs of development are not sustainable unless spread out among large groups of libraries. While it is not practical for a library to gain full control of their technology environment through building it entirely internally, a wide range of options are available for gaining influence in the development of the systems they use.

Most libraries are not able to devote substantial resources toward the advancement of the technology products that they have implemented. They depend on the vendor to take responsibility for maintaining and enhancing the product. This approach does not mean that they are not sophisticated users of the product, but that they do not necessarily have the technical personnel or deep product expertise to advise or participate in the product's ongoing development. Even libraries with ambitious strategies for service and innovation may opt not to invest time in their core automation products, but rather to channel their time and attention to other areas of technology.

Other libraries, especially larger ones, opt to be more actively involved with their core technology products. These libraries have more complex requirements for their technology systems, which may not be entirely fulfilled. This group benefits from engaging with the product developers to help guide future enhancements to address unmet needs or changes in library services or operations. Library engagement with their technology systems and vendors can take a variety of forms.

User Groups and Conferences

User groups have formed around many of the major technology products to advocate the interests of the libraries that have implemented them. Many are governed independently from their respective vendors, ensuring that they represent the interests of the customers rather than the business interests of the company.

IN THIS ISSUE

Ex Libris Introduces Cloud Apps
PAGE 3

Smart Libraries Q&A
PAGE 6

Notes
PAGE 7

- Customers of SirsiDynix Users Group, Inc. <https://www.cosugi.org> (Independent)
- Innovative Users Group <https://www.innovativeusers.org>
- Ex Libris Users of North America <https://el-una.org>
- International Group of Ex Libris Users <https://igelu.org>
- EBSCO User Group <https://www.ebscousergroup.org>
- koha-US <https://koha-us.org>
- Evergreen Community <https://evergreen-ils.org>

Each of the user group organizations depends on financial sponsorships and involvement with their respective vendors, though organizationally they are independent. These user groups convene annual conferences to conduct official business, such as coordinating enhancement requests, advisory panels, or other activities. They also facilitate sharing of information and experiences among users of the products. Vendors and customers facilitate educational sessions on product functionality or other topics of interest. These conferences also provide the vendors with a venue for presenting product roadmaps and to promote new products.

Product Development Partnerships

Many of the current products have emerged out of partnerships between libraries and vendors. It is a well-established practice for vendors to enlist a group of libraries as development partners or as beta test sites when launching a new product. These arrangements can be mutually beneficial in that they give libraries a voice in the early vision and design of a product. Vendors appreciate library input into product development to help ensure that their systems will be well received in the marketplace.

These partnerships require significant commitment of library personnel resources to help formulate functional requirements, validate product concepts, and test the software. Development partner libraries may also have opportunities to be early adopters of the products, often with significant price incentives. These arrangements usually tap individuals in the library with expertise in specific areas of functionality and do not necessarily require the library to perform technical software development.

Development Communities

Some libraries are interested in a deeper level of participation, including technical development. Larger libraries often use a variety of systems that need to exchange data and are interested in developing specialized services for their patrons or staff members. These libraries will often have programmers

and other technologists on staff who are able to work with the APIs of multiple products and to develop scripts or applications beyond core product functions to meet the needs of their institution.

Many of the library technology products used by larger libraries offer APIs available to local programmers to establish interoperability with external systems and to create custom functionality. Vendors generally encourage this kind of technical work because it adds value to their products and facilitates the formation of developer communities. Prompted by pressures to facilitate open systems for programmatic access to the data and functionality of their systems, vendors work hard to create comprehensive APIs and to provide a collaborative environment for sharing customer-driven development. Many have created developer portals, which offer documentation for APIs, sandbox environments for testing code against the APIs, and code repositories for sharing contributions.

SirsiDynix was one of the earliest companies to offer an API to the libraries using its products. This command-line API has been available since the mid-1980s and provides read and write access to all data in the system and was made available to a customer site only after its system administrator completed a training course. The company has continued its emphasis on API access to its products via the Web Services layer, which offers modern RESTful API for its ILS products. SirsiDynix provides a Developer Community portal with access restricted to its customers.¹

Ex Libris has a longstanding practice of encouraging technical development among its customer libraries. The company worked collaboratively with its customers to encourage local technical work with its Aleph ILS. This approach has accelerated with Alma, Primo, Leganto, and other recently launched products. Developers have convened annually for the Technical Seminar, usually in conjunction with the International Group of Ex Libris Users conference. A wide range of resources and a code repository in support of the technical development by customers is provided through the Ex Libris Developer Network portal.²

OCLC also works closely with its users in support of APIs and other technical work. It released the WorldCat Search API in 2008 and has since created APIs for its other products, including those based on the WorldShare platform. The OCLC Developer Network portal provides collaborative tools, documentation, and code repositories of interest to those working with its APIs and other technical components.³

Development communities have formed around each of the major open source products in the library technology arena. While the technical activity of proprietary products centers on the APIs, open source communities develop the application itself. Open source products may also produce APIs,

which usually require a lower threshold of technical expertise to exploit compared to the internal code of the application.

The open source Koha ILS has an extensive global community that takes responsibility for ongoing development, quality assurance, documentation, translation, and other activities. This community comprises developers representing dozens of support organizations as well as programmers in libraries using the product.⁴

Evergreen, implemented mostly by libraries in the United States and Canada, has a development community of similar scope, though of somewhat smaller scale than Koha.⁵

The FOLIO project has cultivated an extensive global development community that has been engaged for the past four years in the creation of this new open source library services platform. This newsletter has given extensive coverage of the FOLIO project, including recent implementation. The community maintains a wiki for resources supporting the FOLIO developers.⁶

As shown by this broad range of possibilities, libraries can

Prompted by pressures to facilitate open systems for programmatic access to the data and functionality of their systems, vendors work hard to create comprehensive APIs and to provide a collaborative environment for sharing customer-driven development.

choose the level of engagement with their core technology products and providers that best aligns with their priorities. Given the prevailing trend of fewer in-house technical personnel in libraries, these products need to work well without local technical intervention. Fortunately, technologists with many libraries have the expertise and interest in working with these strategic technology applications at a deeper level. Library-based developers have made a substantial number of contributions surrounding each of the major products, benefiting the broader library community.

The opportunities for library-based development continue to expand. The practice of developing applications and scripts based on APIs exposed by core library applications has become well established. This issue of *Smart Libraries Newsletter* features a recent advancement in this space where library developers can create apps that run

directly within the Ex Libris Alma platform, fully integrated within its native interface. An important innovation, this approach resonates with the concept of user-installable apps, the basis of mobile computing.

Ex Libris Introduces Cloud Apps

The concept of user-installable apps has become a basic expectation of modern computing, especially on mobile devices. Both the Apple iOS and Android mobile environments make it easy for persons to find, select, and install apps made available through their respective app stores. Installing these apps does not require technical expertise and safeguards are in place to ensure that the apps will work correctly and not crash the device.

Ex Libris has recently released a new framework on its software platform that emulates the app paradigm experienced in the mobile arena. Ex Libris Cloud Apps brings installable apps into the environment of its strategic products. Apps can be created and deployed for any of the major products running on the “Ex Libris higher-ed platform,” which includes Esploro, Primo VE, Leganto, Rialto, and Rapido, in addition to Alma. The new framework enables Alma users to find, select, and enable apps without technical intervention. These apps,

created by Ex Libris customers, give Alma users an extended set of tools to work more efficiently or perform specialized tasks. These apps run within the Alma platform and are used directly within the Alma staff-use interface, inheriting context such as authorizations and preferences.

This app framework enables the creation of functionality beyond what is built into the core products by the company’s developers. Enhancements and new features for core products such as Alma must be suitable for the entire customer base. Even though the ongoing development relies on customer recommendations, many bits of functionality may not be appropriate for all sites.

Apps created by the community of developers from customer sites address specialized scenarios. Ex Libris has cultivated, from its customer base, programmers and technologists interested in working with APIs and other technical aspects of its products. The company launched its Developer Network

portal in April 2014 to bring together documentation and other collaborative tools to support technical collaboration with its customers.

Assistance with apps comes through the Ex Libris Developer Network. This model of community support is well established among the users of Ex Libris products. While Ex Libris personnel participate in the Developer Network, the model of collaborative support differs from the standard channels of support associated with the company's core products.

The concept of apps is not new to Alma. In May 2019 Ex Libris introduced an environment, branded as the App Center, in which apps ran on external computing environments, such as local server or cloud infrastructure operated by customer sites. Cloud Apps brings a significant change in that its apps run completely within the Ex Libris platform, enabling their activation and use to take place entirely within Alma's interface.

Cloud Apps are accessed through a new icon in the Alma toolbar. Selecting the Cloud Apps icon presents the apps currently installed. Users can browse or search for additional apps that can be activated. Each app listed includes its name, a brief description, its developer, any applicable terms of use, and a button to install it.

The new Cloud Apps framework will be deployed in the August 2020 release of Alma. More than 60 apps are currently available through the App Center. A subset of these apps has been developed as Cloud Apps; others are extensions that operate outside the Ex Libris cloud. With the Cloud Apps framework, the apps run within the Ex Libris platform infrastructure and are fully integrated into its user interfaces. Apps operate according to the context of the current display.

Some apps automate routine tasks to improve efficiency and reduce manual steps. These are similar to macros or keyboard shortcuts used with other systems. Combining multiple steps of a task reduces times waiting for a system response. Others interact with external resources to bring data into Alma.

Some of the Cloud Apps in the initial deployment include:

- **Hathitrust Availability:** a tool for checking the availability of items in a list of search results in the Alma staff interface.
- **Alma Refine:** integrates references into bibliographic records from selected resource compatible with the OpenRefine standard, including the Getty vocabularies, WikiData, and GeoNames.
- **Alma Manager.**
- **Role Toggler:** a tool for activating or deactivating all the roles of a user record, which saves time by avoiding the need to submit multiple changes.

- **CSV User Loader:** creates a batch of Alma users from a file.

The Cloud Apps environment includes a layer of institutional management. Not all apps will be appropriate for all sites. Managers at each institution can select which apps will be available to their users. Individual users will then be able to select and use the apps without administrative intervention.

Enabling apps to execute within a platform such as Alma requires multiple layers of validation to protect the broader environment from any possible disruption or interference. These layers would include technical measures, such as coding structure and standards to ensure that apps properly integrate with the host environment. Manual or automated validation verify that any code or included libraries are free from known security vulnerabilities. Apps undergo rigorous app validation. Manual testing by the platform administrator, in this case Ex Libris, verifies that the app performs as expected and conforms with user interface conventions.

Beyond APIs

Ex Libris has a longstanding strategy of enabling APIs for its products based on proprietary software. Cloud Apps and its predecessor App Center leverage these APIs into a new development and deployment framework, enabling the creation and use of new components beyond those delivered on the core product.

APIs and Apps provide openness and extensibility for major software platforms, including both open source and proprietary products. App developers do not need access to the underlying source code of the core application itself to create related services or interfaces if suitable APIs are provided and well documented.

The Cloud Apps framework is based on the open API Ex Libris exposes on its platform, documented via its Developer Network portal. Ex Libris enables open access to this portal and the extensive documentation of its APIs. Live use of the APIs requires applying for a developer's key, which does require an authorized login to the site.

Developing Cloud Apps

Programmers in the library can develop apps for use by others in their institution or to be published for the entire Alma community.

The development process of a Cloud App involves five steps:

1. Develop the app in HTML and Angular according to the templates and styles documented.

2. Create a release of the code on GitHub.
3. Submit the app to the App Center on the Ex Libris Developer Network.
4. Wait for manual review from Ex Libris.
5. Once approved, the Cloud App will appear in Alma and can be activated by users.

The Ex Libris Developer Network portal provides documentation and other resources for creating apps and a process for publishing them for global deployment.

Cloud Apps are created by following the templates, structure, and components described in the Developer Network. Developers create apps using the Angular programming platform, HTML, and JavaScript. Angular is an open source web application framework and is the latest version of AngularJS, developed by Google and released in 2010. Ex Libris provides a style guide to help developers create apps with interfaces consistent with Alma. The CSS bundled with the Cloud Apps project templates helps achieve consistency and compatibility. Cloud Apps can address the Ex Libris Cloud Apps API to interact with Alma data or functionality.

Development of Cloud Apps is performed in a local environment. This environment would include an installation of the Ex Libris Cloud App Command Line Interface (CLI), which includes the tools needed to develop, test, and build Cloud Apps.

After the app has been successfully built with CLI, with no errors, the developer will publish a release on Github in a public repository. The next step is to submit the release to the App Center on the Developer Network as a Cloud App along with the link for the Github repository. Ex Libris will then begin its review process, and it will appear as an available Cloud App in Alma once approved.

Industry Context

The Ex Libris Cloud App framework of user-installable apps represents a significant advancement in the realm of major library software applications. This framework provides an elegant model of extensibility and interoperability so that institutions implementing the system can gain capabilities beyond the delivered product. While the Cloud App environment does not obviate the need for Ex Libris to continually enhance the product for capabilities useful to all its customers, it enables

the creation of custom features, interfaces, or extensions via its customer development community.

Other examples of extending core products via apps can be seen in the broader educational technology sector. The Cloud App framework, for example, is similar to the Building Blocks concept associated with the learning management systems from Blackboard. According to the product documentation: “A building block is a Java web app that is deployed as a plug-in to the Blackboard Learn application.”⁷ These Building Blocks can be created by external developers for integration with external systems or to create customized features. They run within the Blackboard platform but must be installed by Blackboard Support personnel.

Almost all the major integrated library systems and library services platforms offer an API available to local programmers and systems integrators. These APIs provide essential capabilities for batch data management and interoperability with external business applications. They can be the basis for creating new interfaces or software components. Many vendors also offer portals for collaborative development and sharing of code. These environments do not match the sophistication of a framework where apps can be developed, deployed, and run natively within the platform of the core application.

Library services platforms with a single global codebase are well positioned to offer user-installable apps. OCLC’s WorldShare Management Services and Ex Libris Alma both fall into this product genre. Integrated library systems, mostly developed as single tenant server-based applications, are less well suited for user-installable apps, given the multiplicity of versions and other variations among installations.

OCLC offers a full API for WorldShare Management Services and likewise offers capabilities to create custom apps and scripts that can be created and shared. The OCLC Developer Network includes an API Explorer, which offers documentation and tools for developing and testing code against the APIs. More than 100 apps and services based on the APIs across multiple OCLC products and developed by customers or other vendors are made available through the OCLC Developer Network Gallery. These apps run externally to the products involved and are not deployed within the OCLC platforms. While this gallery of apps offer important capabilities in extending functionality of OCLC products and services, it is not quite the same as an integrated app framework.

Cloud Apps brings a significant change in that its apps run completely within the Ex Libris platform, enabling their activation and use to take place entirely within Alma’s interface.

The open source FOLIO project follows the app metaphor in its overall architecture. Developers can create modules that operate within the Okapi API gateway and are based on the Stripes user interface toolkit. This approach to module development lends great flexibility in how FOLIO instances can be assembled and deployed. An important part of the vision for FOLIO involves enabling libraries, vendors, or other stakeholders to create modules according to their specific requirements. Any given deployment of FOLIO can assemble the modules supporting its needs. It does not, at least at this point in its development, provide a framework where individual FOLIO users can select and deploy modules and apps without technical intervention.

FOLIO encompasses most of the characteristics of the library services platform category. It differs from WorldShare Management Services and Alma in that there will be different instances of the software deployed. Some libraries plan to host FOLIO locally, and multiple vendors will deploy FOLIO on their respective hosting infrastructure.

Entirely created through community-based development, FOLIO has been developed and licensed as open source software. Ex Libris products, in contrast, are based on proprietary software licenses, with open APIs made available to customer and non-customer developers. With Alma and WorldShare Management Services, the core product is based on proprietary software, with community-based development of extensions and apps. Both models depend on robust and documented APIs

Cloud Apps build on the Alma platform, which has seen well over a decade of development, including a four-year initial development effort and monthly updates issued since its initial production deployment in July 2012. The unified codebase underlying all instances of Alma and a robust API framework provide a reliable basis for custom app development. A mature product plus community-based apps and extensions further strengthens Ex Libris's position as a technology provider for academic libraries.

Smart Libraries Q&A

Each issue Marshall Breeding responds to questions submitted by readers. Email questions to Patrick Hogan, Managing Editor, at phogan@ala.org.

As libraries prepare for service post-pandemic, self-checkout is looking more appealing. What are functionalities or factors we should be looking at in choosing a self-checkout station? Who are the major vendors in the marketplace? What are recommended steps for sanitizing self-checkout stations in operation?

During this time of the COVID-19 pandemic crisis, libraries are investigating many different ways in which they can reopen and provide services in the safest way possible. It is essential to ensure the safety of library workers and library patrons. Libraries should look to health care experts as they shape their policies and procedures that can be considered reasonably safe and to avoid spread of the coronavirus.

Self-service stations can minimize the interactions between library workers and library patrons. This equipment along with Plexiglas barriers can provide at least some protection. Self-service kiosks do not address the possibility of contamination via surfaces, such as the keypads and other surfaces of the kiosk or the materials themselves. Even if the virus spreads more easily through other means, transmission via

surfaces remains a concern. Research from the REALM Project has shown that the virus does not persist on library materials beyond three days.⁸ In a context where libraries might assume that patrons could be infected and transmit virus by handling materials, it would be challenging to design processes where patrons freely browse library stacks and select their own materials. Many libraries are implementing processes where patrons request materials online, which can then be picked up via a curbside service. In this scenario enabling patrons to use self-service to charge materials may provide less safety than having library workers charge the material.

If there are procedures where patrons can charge materials to themselves, procedures would need to be in place to prevent contamination via the equipment. Cleaning exposed components between patrons would be onerous, and libraries would need to seek expert advice on whether that would be necessary.

Another self-service option can be accomplished through mobile check-out where patrons use their own mobile phones or tablets to charge materials. Many ILS or self-check vendors offer apps with this capability. Mobile checkout still involves patrons' handling of materials, but avoids the risks related to patron use of self-check stations.

Libraries can also use sorters for returned items or other automated material handling equipment to reduce the contact

with materials by library workers. Libraries can develop processes for returned materials that automatically sort into bins that can be quarantined for a prescribed duration.

Many of the companies offering self-check and related equipment have developed resources to assist libraries in planning for equipment to deal with current and future events. Some of the ones I have come across include:

- **Bibliotheca:** www.bibliotheca.com/reopening-libraries
- **Tech Logic:**
 - totalAMH Quarantine Mode & selfCIRC Touchless Checkout <https://www.tech-logic.com/covid/quarantine-touchless>
 - Disinfecting and cleaning procedure for Tech Logic

equipment. <https://www.tech-logic.com/covid/cleaning>

- **Lyngsoe Systems:** Library Units updates on Coronavirus (COVID-19) <https://lyngsoesystems.com/library-coronavirus>

Acquiring and installing self-check stations or automated material handling equipment will likely take longer than would be helpful for addressing the short-term needs for library re-opening. Absorbing the expense of these items may also be difficult when library budgets may be constrained or frozen due to funding shortfalls. But as the crisis abates, libraries will want to consider technology components in longer term planning to address concerns that will inevitably remain even after the current crisis subsides.

Notes

1. SirsiDynix. “Developer Community.” <https://www.sirsi-dynix.com/developer-community>
2. ExLibris. “Developer Network.” <https://developers.exlibrisgroup.com>.
3. OCLC. “Developer Network.” <https://www.oclc.org/developer/home.en.html>.
4. Koha. <https://koha-community.org>.
5. Evergreen. <https://evergreen-ils.org>.
6. FOLIO Project. “FOLIO Wiki.” <https://wiki.folio.org>.
7. Blackboard. “Building Blocks in Learn SaaS.” https://help.blackboard.com/Learn/Administrator/SaaS/Building_Blocks.
8. OCLC Web Junction. “REALM Project: Reopening Archives, Libraries, and Museums.” <https://oclc.org/realm-project>.

Questions or suggestions
for topics in future issues?



Contact Patrick Hogan at
phogan@ala.org



Smart Libraries Newsletter
American Library Association
225 N. Michigan Ave., Suite 1300
Chicago, IL 60601-7616 USA
Address Service Requested

NON PROFIT
US POSTAGE
PAID
PERMIT 4
HANOVER, PA

September 2020 Smarter Libraries through Technology

Smart Libraries Newsletter

Marshall Breeding's expert coverage of the library automation industry.

Editor

Marshall Breeding
marshall.breeding@librarytechnology.org
Twitter: @mbreeding

Managing Editor

Patrick Hogan
312-280-3240
phogan@ala.org

Digital Access for Subscribers

journals.ala.org/sln

TO SUBSCRIBE

To reserve your subscription, contact the Customer Service Center at 800-545-2433, press 5 for assistance, or visit alatechsource.org.

ALA TechSource purchases fund advocacy, awareness, and accreditation programs for library professionals worldwide.

Production and design by the American Library Association
Production Services Unit.

Smart Libraries Newsletter is published monthly by ALA TechSource, a publishing imprint of the American Library Association.

alatechsource.org

Copyright © American Library Association 2020. All rights reserved.