Web Services in the Library Environment

Web services and the Service-Oriented Architecture have become well established in the broader information-technology industries, yet adoption of Web services within the library arena has been less than aggressive. Although there have been many examples of library-related functions being implemented as Web services, they are not pervasive in the library field—at least not yet.

In this chapter, I will review some of the existing library-related efforts that make use of Web services. First, I'll cover some of the organizational efforts that have taken place to promote Web services in the library field; after that, I'll review some of the emerging new technologies, which have been implemented in this model. Finally, I will examine how some of the major companies involved in producing library-automation software approach Web services.

The VIEWS Initiative

The Vendor Initiative for Enabling Web Services (VIEWS) was founded in June 2004 and was chaired by Carl Grant, president/CEO of VTLS. Recognizing the importance of Web services and the benefits they could provide for libraries, this group—comprised of a consortium of vendors—was formed to facilitate the development of interoperable Web services related to library applications. Although NISO participated in its efforts and served as a liaison to communicate relevant issues with its members, VIEWS was not a NISO-sponsored group. Participating vendors included:

- Auto-Graphics—Oct. 2004
- Dokimas (DS)—Jan. 2005
- Endeavor Information Systems—July 2004
- Ex Libris—Mar. 2005
- Fretwell-Downing Informatics—June 2004
- Index Data—June 2004
- Muse Global—June 2004
- OCLC—June 2004
- Polaris Library Systems—Dec. 2004
- SirsiDynix—Nov. 2004
- Talis—Aug. 2004
- The Library Corporation (TLC)—Sep. 2004
- VTLS—June 2004
- NISO—June 2004

VIEWS was intended to provide a forum in which library-automation vendors could define a set of Web services, which could then be used in a variety of applications in the library field. The thinking was along the lines of: *There are common components of library functionality that need to be expressed as Web services, and it is in the best interest of libraries and those involved in library automation to define a set of Web services that will work across all of their implementations.* In other words, the industry as a whole would benefit from having vendor-neutral definitions for commonly used Web services. The library-automation community benefits from such standards as Z39.50, MARC, SIP2, NCIP, EDI, and the like; the VIEWS initiative was based on the principle that having a shared set of Web-service specifications would offer similar advantages.

Although the various vendors were adopting the model for Web services at different levels, it was clear that momentum was building. There was a great need to start developing some common understandings quickly, because if each vendor wrote its own Web-service specification for common transactions used in library applications, it would be extremely problematic later on—as libraries inevitably would demand them to work together. So those involved in the initiative determined it would be better to establish shared specifications early on, and thus the VIEWS consortium identified two areas in which to begin: *authentication* and various pieces of functionality related to *metasearch*.

The general approach of the VIEWS initiative was to identify transactions that would most benefit from a common approach; define a set of Web services that express the functionality needed; and develop pilot implementations that would test the viability of the Web service.

The VIEWS consortium, however, did not begin without controversy. The relationship between VIEWS and NISO was unclear, and many wondered, "Would the work of a self-selected group of vendors be able to establish specifications followed by the whole industry?" And not all of the vendors in the field chose to participate. Innovative Interfaces—even though it was a supporter of Web services and it had Web-service development efforts underway—was one of the major companies in the industry that chose not to join.

Much of the work, too—related to the potential development of Web services—was intertwined with NISO initiatives.

The organizational status of VIEWS was further complicated by the fact its chair, VTLS's Carl Grant, was to become the chair of NISO in July 2005. At about that time, the VIEWS consortium transitioned into a NISO-sponsored working group. (Information about VIEWS is available at www.views-consortia.org.)

The VIEWS Initiative
www.views-consortia.org

VIEWS Membership Adds Ex Libris (Vendor Initiative for Enabling Web Services)
www.exlibrisgroup.com/newsdetails.htm?nid=377

NISO Web Services and Practices Working Group
www.niso.org/committees/Services/Services_comm.html

NISO Web Services/Practices WG Wiki
http://research.talis.com/2005/wswg/wiki/

NISO Web Services and Practices Working Group

Around August 2005, VIEWS was replaced by the NISO Web Services and Practices Working Group, which has become the primary initiative to develop common ground for Web services in the library arena. The group was cochaired by Candy Zemon of Polaris Library Systems and Ian Davis of Talis. Although similar in scope to the VIEWS effort, the NISO group operates somewhat differently; the VIEWS consortium consisted only of vendors, while the NISO Web Services and Practices Working Group includes both vendors and library representatives.

The VIEWS approach focused on specific implementations of Web services; the NISO group will focus on developing best practices for designing and deploying Web services. According to the group's Web site (www.niso.org/committees/Services/Services_comm.html), the group is charged to:

- produce and maintain a "Web Services Best Practices" document for general use in assessing new and ongoing Web-service applications, not necessarily confined to the library world; and
- provide and maintain "Web Services Interoperability Mechanisms," which can be used to test specific instances of inter-vendor Web-service interoperability.

The group also maintains a wiki to document its work and interests. The wiki is currently hosted by Talis at http://research.talis.com/2005/wswg/wiki. As of this writing, the transition to the new organization is recent.

Non–Web-Service Library Standards and Protocols

Given that Web services have come on to the scene relatively recently, many of the standards developed for libraries pre-date this approach. These earlier protocols target the same broad goals as Web services—that of providing computer-to-computer communications, which are interoperable and independent of the underlying software application, operating system, and hardware involved. Before examining the technologies that have emerged in the library field based on Web services, I'll briefly discuss earlier technologies.

Z39.50

Z39.50 is a search-and-retrieval protocol that has a long history and continues to find wide use in library software. This protocol performs search operations and returns results in MARC communications format. The protocol is based on Abstract Syntax Notation, or ASN.1 and Basic Encoding Rules (BER). The Library of Congress has been designated as the official maintenance organization for this standard (www.loc.gov/z3950/agency). Z39.50 is considered a very complex protocol and has not been widely adopted outside of the library sphere.

ISO 10160/10161 and ISO ILL

ISO 10160/10161 and ISO ILL provide a set of standard protocols for the interoperable exchange of ILL transactions. This pair of standards, like Z39.50, falls within the Open Systems Interconnect style of communication, based on ASN.1 and BER. The Library and Archives Canada serves as the maintenance agency for these standards and provides authoritative information related to the standard (www.collectionscanada.ca/iso/ill).

NISO Circulation Interchange Protocol, NCIP (Z39.83)

NISO Circulation Interchange Protocol (NCIP), or Z39.83 as it is also called, is a protocol that supports transactions related to library-circulation functions. This protocol finds use in self-service circulation stations, direct consortium borrowing (as a component of ILL systems), and other library applications. NCIP encodes its messages in XML, but the protocol does not follow the architecture of a Web service. The Colorado State Library has been designated as the maintenance agency for this standard (www.cde.state.co.us/ncip).

Non–Web-Service Library Standards and Protocols

Z39.50
www.loc.gov/z3950/agency

ISO 10160/10161 and ISO ILL
www.collectionscanada.ca/iso/ill

NISO Circulation Interchange Protocol (Z39.83)
www.cde.state.co.us/ncip

OpenURL
www.niso.org/committees/committee_ax.html

UN/EDIFACT and ANSI X12

These Electronic Data Interchange (EDI) standards are intended for the exchange of messages related to business transactions. Efforts to implement EDI began as early as the 1970s, and the standards were long considered as the way that libraries could automate some aspects of their business transactions (such as the procurement of books and serials from suppliers). Unlike the other standards mentioned in this section, EDI is an international standard employed across many industries. In the library arena, EDI is the standard used between libraries and suppliers for electronic ordering, making claims for items expected but not received, and for the transfer of invoices. EDI and the Web-service approach are competing approaches for business-to-business e-commerce transactions. EDI is well established and continues to be widely implemented.

MARC 21 Format

Within these computer-to-computer protocols, there are a number of data-format standards employed by libraries. For example, the MARC 21 record format stands as the most widely adapted library standard. Practically all library-automation systems represent bibliographic records in some form of MARC format and use this format when moving bibliographic records between systems. MARC record formats store bibliographic records in a compact format. *MARCXML* provides an XML structure for MARC 21 data. Based on a simplified set of tags, *Metadata Object Description Standard* (*MODS*) provides an XML schema for MARC 21 bibliographic records. *Metadata Authority Description Schema* (*MADS*) provides an XML schema for selected tags of the MARC 21 authority format. The *Encoded Archival Description* (*EAD*) provides a standard for creating finding aids of archival collections in XML.

Dublin Core

Dublin Core is a metadata format expressed in XML, which provides a simplified approach for describing information objects. Dublin Core provides fifteen metadata elements: *Title, Creator, Subject, Description, Publisher, Contributor, Date, Type, Format, Identifier, Source, Language, Relation, Coverage,* and *Rights*. Applications that need more precision than provided by the basic elements can use *Qualified Dublin Core,* which employs the use of qualifiers with any of the elements to provide more information about the meaning of that element. Dublin Core is widely used as a metadata format for describing electronic resources or other applications in which the complexity of the MARC formats may not be needed. Many organizations choose to provide Dublin Core metadata, describing the content of Web pages in the head of the record using *metatags*. Some applications make use of this metadata for automatic cataloging or classification of Web pages. (*Note: MARC and the related MODS and MADS XML schemas, as well as EAD, are examined thoroughly in Brad Eden's two issues of* Library Technology Reports *on metadata—Sep/Oct 2002, 38:5, "Metadata and Its Applications"; and Nov/Dec 2005, 41:6, "Metadata and Its Applications: New Directions and Updates."*)

OpenURL

OpenURL Framework for Context-Sensitive Services provides a mechanism for linking to resources independently of their physical locations. It emerged out of the need to provide a more intelligent linking mechanism from citations to electronic-journal articles. Rather than hard-coding a URL with a server and document location, URLs carry embedded metadata that can be processed by a link resolver to determine a resource's physical location based on a database of journal holdings and a profile of library subscriptions available to the user. This process addresses the problems that arise when e-journal content exists in multiple resources. Through OpenURL, links can be presented that direct the user to the copy of the article available by virtue of his or her library's subscriptions, avoiding taking the user to copies not available to him or her. OpenURL finds use beyond citation-to-full-text linking too; a link resolver can present other services related to the item described in the OpenURL, such as ILL requests, document-delivery services, or where to obtain print copies. NISO maintains the home page for OpenURL at www.niso.org/committees/committee_ax.html.

Several commercial products available today are based on the OpenURL framework, including SFX from Ex Libris, TOUResolver from TDNet, 1Cate from Openly Informatics (which was acquired by OCLC in January of this year), WebBridge from Innovative Interfaces, and Article Linker from Serials Solutions.

The OpenURL framework does not fully fit into the model of Web Services, which typically involves a service request and a service response. The OpenURL specification describes a specific technique for delivering metadata through a URL, but does not prescribe what services do with that metadata. Although an OpenURL looks somewhat like a REST-style service request, implementations based on OpenURL do not necessarily involve interactions that fit into the model of Web services. In addition, responses to the OpenURL *do not usually* involve a response of an XML data stream or messages transmitted with SOAP. (*Note: In-depth coverage of the OpenURL is available in, "Linking and the OpenURL," by Jill Grogg, the Jan/Feb 2006 issue, 42:1, of* Library Technology Reports.)

Library Protocols Based on Web Services

Relative to the history of library automation, Web services have not been on the scene for that long. Work on Z39.50, for example, began as early as 1988. Work on Web services by the W3C did not begin until 2002, although the members/purveyors of W3C can trace their origins to XML-RPC, which was developed around 1998. As XML and Web services become more widely adopted in the broader IT community, library standards and protocols that emerge are consistent with these prevailing trends. Over time, one of the key issues in library automation will involve retrofitting earlier standards into XML and the Web-service architecture.

Search/Retrieve Web Service

The core function for libraries involves searching for information and retrieving results, thus search technologies have long been a centerpiece of library automation. Almost all library-oriented software involves search and retrieval in some way. Due to its status as the international library standard for search and retrieval, Z39.50—with all its problems and complexities—plays a seminal role in a vast array of library-automaton implementations.

Given the importance of Z39.50, the library community has a strong interest in ensuring that it evolves to stay in step with technology shifts. *Z39.50 International: Next Generation*, or ZING, was established to take the concepts and functionality embodied in Z39.50 forward. In the words of the ZING Web site, this group, "covers a number of initiatives by Z39.50 implementors to make the intellectual/semantic content of Z39.50 more broadly available and to make Z39.50 more attractive to information providers, developers, vendors, and users, by lowering the barriers to implementation while preserving the existing intellectual contributions of Z39.50 that have accumulated over nearly 20 years." (www.loc.gov/z3950/agency/zing)

One of the ZING initiatives involves the development of a version of Z39.50 in the framework of Web services. The initiative resulted in the development of a search-and-retrieval protocol that:

- leverages the concepts of Z39.50;
- mitigates some of its complexities;
- offers a subset of the Z39.50 operations;
- relies on XML data encoding such MARCXML instead of MARC 21; and
- uses a Web-service model for communications.

Two flavors of the new search-and-retrieve protocol emerged:

- *SRU*: Search/Retrieve via URL
- *SRW*: Search/Retrieve Web Service

SRU follows the REST (Representational State Transfer) model for implementing the search-and-retrieval protocol as a Web service. With SRU, service requests are sent to the server encoded in an *HTTP GET* request.

SRW implements the same functionality as SRU but uses *SOAP over HTTP* as the mechanism for transmitting messages. Only minor differences exist between SRU and SRW, and these are due to features helpful in RESTful services, but do not apply to those that rely on SOAP messaging. Although SRW more closely follows the model for Web services, SRU has been much more widely implemented.

Queries in SRU and SRW conform to the Common Query Language (CQL) and are transmitted as text strings rather than using the structured approach as with Z39.50.

Queries formulated in CQL can be read by humans, yet have enough structure to formulate precise search qualifications. Detailed information on CQL can be found within the Library of Congress' documentation pages of standards (www.loc.gov/standards/sru/cql). SRU and SRW support three operations:

- *SearchRetrieve*—Implements the core functionality of search and retrieval. The SearchRetrieve service request supports ten parameters: *query, startRecord, maximumRecords, recordPacking, recordSchema, resultSetTTL, sortKeys, stylesheet* (SRU only), *extraRequestData, operation* (SRU only), and *version*. The parameters supported in the service response include: *version, records, numberOfRecords, ResultSetId, resultSetIdleTime, nextRecordPosition, diagnostics, extraResponseData,* and *exhoedSearch RetrieveRequest*.
- *Scan*—Provides clients with the ability to browse index entries on a server.
- *Explain*—Describes the capabilities of the SRU server.

Definitive information regarding SRU and SRW can be found on the Library of Congress Web site (www.loc.gov/standards/sru).

Z39.50 International: Next Generation (ZING)
www.loc.gov/z3950/agency/zing

Common Query Language (CQL)
www.loc.gov/standards/sru/cql

A number of implementations of SRU and SRW have been developed including:

- *OCLC Research* (www.oclc.org/research/projects/webservices)—OCLC's SRW/U is developed using Java and the Apache SOAP toolkit. OCLC makes this software available for free use through an open-source license (OCLC Research Public License). OCLC's SRW/U software is included with the DSpace distribution.
- *Index Data*, a company that specializes in open-source development for library applications, includes support for SRW and SRU in its *YAZ toolkit* (www.indexdata.dk/yaz). This toolkit supports the development of search interfaces based on Z39.50, SRW, and SRU. According to Index Data, its YAZ toolkit has been used to develop Z39.50-based applications more than any other alternative. Index Data supports its open-source development with income it receives for its consulting and optional support services.
- *The Cheshire Project* (http://srw.cheshire3.org/downloads) is a collaborative project of the University of California, Berkeley and the University of Liverpool to create, "A next-generation online catalog and full-text information retrieval system." Those involved in the project have created an open-source Python-based SRW server.

A number of libraries, publishers, automation vendors, and other organizations have implemented SRW or SRU, including the British Library, Adlib Information Systems, the Library of Congress, the National Library of the Netherlands, and OCLC PICA. The Library of Congress provides information on SRW implementations on the Web at www.loc.gov/z3950/agency/zing/srw/implementors.html.

OCLC's SRW/U
www.oclc.org/research/projects/webservices

Index Data's YAZ Toolkit
www.indexdata.dk/ya

The Cheshire Project
http://srw.cheshire3.org/downloads

SRW Implementations
www.loc.gov/z3950/agency/zing/srw/implementors.html

The Open Archives Initiative Protocol for Metadata Harvesting

The Open Archives Initiative supports a federated-search model, based on harvesting metadata from multiple information repositories and then creating centralized search services (based on the harvested metadata). This model of federated search provides an alternative to those that are based on simultaneous queries (which are sent to multiple target resources and then collate the results for presentation). The OAI model emerged out of the need to provide search services across multiple repositories.

The Open Archives Initiative for Metadata Harvesting (OAI-PMH) provides a mechanism for systematically obtaining metadata batches from information repositories. OAI-PMH was developed as a REST style Web service, in which requests are sent to repositories as URLs.

The OAI-PMH model relies on data providers (which make metadata available via information repositories) and service providers (which harvest metadata and build new value-added services). The classic OAI-PMH model involves a service provider harvesting metadata from multiple, specialized repositories within a sphere of interest, and then providing a search service for the broader discipline. This model allows researchers to perform searches on broader topics without having to search several different repositories. In most cases, search services harvest only metadata and then point searchers back to the original servers to view the items described by the metadata. OAI-PMH uses unqualified Dublin Core as its default metadata format, but different communities of OAI-PMH implementors can specify alternative metadata formats.

The primary purpose of OAI-PMH is to systematically and efficiently harvest metadata from repositories. The protocol includes a number of flow-control features that prevent a service provider from becoming overloaded during a harvesting operation. At any point, a provider can stop delivering records and issue a resumption token, which gives the service provider information about where the transfer left off and in what time frame the harvesting can resume. OAI-PMH supports requests that are limited to records added or modified, because a given date provides the capability to perform incremental updates (once the initial metadata from a repository has been comprehensively harvested).

OAI-PMH finds use in a wide variety of applications; it has expanded well beyond its original application in e-print servers to all variety of digital-library applications. The protocol, unlike almost all other protocols that emerged out of the library sphere, has been adopted by at least some nonlibrary organizations. Google, for example, supports the OAI-PMH as one of the protocols it uses to harvest Web sites for its index (see www.google.com.tw/webmasters/sitemaps/docs/en/other.html#oai).

OAI-PMH is viewed as a benefit for the process of Web harvesting, because it provides the means to harvest only new and changed pages—unlike the traditional Web-indexing processes, which involve systematically crawling all pages with each pass.

Definitive information regarding the OAI-PMH is available at the Open Archives Initiative Web site (www.openarchives.org/OAI/openarchivesprotocol.html). Software tools that implement the OAI-PMH protocol are available from several sources including:

- OCLC Research created *OAICat*, an open-source Java implementation of OAI-PMH (www.oclc.org/research/software/oai/cat.htm); and
- Virginia Tech offers its *VTOAI OAI-PMH2 PERL Implementation* toolkit for implementing the OAH-PMH protocol (www.dlib.vt.edu/projects/OAI/software/vtoai/vtoai.html).

A more complete list is available on the Open Archive's Web site (www.openarchives.org/tools/tools.html).

Standardized Usage Statistics Harvesting Initiative (SUSHI)

The Standard Usage Statistics Harvesting Initiative (SUSHI) deals with the issue of transferring statistics that assess an institution's use of an electronic resource (from a publisher) to which a library subscribes. Libraries spend a significant portion of their budgets on electronic resources, and given the limitations of their budgets, libraries must make decisions on which e-resource subscriptions they will continue each year and which they will cancel. In order to make informed decisions regarding the allocation of their budgets on subscriptions, libraries need detailed information on how frequently their patrons/users access each resource. It can be difficult, however, to produce these use statistics; libraries can devise ways to count use of each resource from their networks but such measurements are often incomplete. The e-resource publishers are in a much better position to produce these measurements, because each user that accesses their systems must be identified by institution and authenticated before gaining access.

The Open Archives Initiative
www.openarchives.org

Google and the OAI-PMH
www.google.com.tw/webmasters/sitemaps/docs/en/other.html#oai

OAI-PMH Protocol
www.openarchives.org/OAI/openarchivesprotocol.html

OCLC Research's OAICat
www.oclc.org/research/software/oai/cat.htm

VTOAI OAI-PMH2 PERL
www.dlib.vt.edu/projects/OAI/software/vtoai/vtoai.html

Standardized Usage Statistics Harvesting Initiative (SUSHI)
www.niso.org/committees/SUSHI/SUSHI_comm.html

Project COUNTER is an initiative to develop a set of standard practices regarding how publishers will produce use statistics related to library subscriptions. The *COUNTER Code of Practice for Journals and Databases* details the recommendations on: information that should be included in use reports; the format in which those statistics will be presented; the methods through which these reports will be made available; and the frequency and methods in which libraries will be notified that use reports are available. A publisher may, for example, need to produce reports of the total searches and sessions performed by users (associated with a subscriber's institution) by month and database title.

Typically, libraries gain access to these use-statistic reports by accessing a password-protected Web site to view the statistical tables. Statistics are also made available in CSV (Comma Separated Values) format for downloading and importing into spreadsheets. Because libraries often subscribe to e-resources from dozens (if not hundreds) of publishers, retrieving COUNTER statistics from each publisher can be a tedious time-consuming process. So NISO charged SUSHI to develop an automated process for transferring these use reports from publishers to libraries. One of the features expected in a library's electronic-resource management system includes the functionality to record the use statistics for each of the library's e-resource subscriptions. The mechanism that SUSHI developed was a Web-service interface in which the ERM acts as a service consumer, requesting statistical data from the publisher as a service provider. Definitive information on SUSHI can be found on NISO's Web site at www.niso.org/committees/SUSHI/SUSHI_comm.html.

Adoption of Web Services by the Primary Library-Automation Vendors

In this section, I'll examine the degree to which the current library-automation software vendors embrace Web services. Most of the data in this section summarizes responses provided by each company to a survey about Web services I distributed to them. The questions asked in the survey were:

- Does your company offer products that use Web services to communicate with external applications? To what extent do your products use Web services internally?

- How important are Web services to your company's development strategy? To what extent have you received inquiries or requests from clients or potential clients that involve Web services or SOA?

- How would you characterize your company's Web-services efforts and/or products in:
  □ Production use by clients?
  □ Development?
  □ Planning stage?
  □ Not interested.

- Describe how your software components make use of the following:
  SOAP (Simple Object Access Protocol)
  UDDI (Universal Description, Discovery, and Integration)
  WSDL (Web Services Description Language)

- Do your applications offer an API (application programming interface) that interoperates with other components using conventions other than Web services? Does this API use XML for service and message exchange?

- What development environment does your company use to implement Web Services?
  □ Microsoft .NET?
  □ Java?
  □ Perl?
  □ Others?

- What business-to-business applications do you enable for libraries that rely on Web services?

- Please provide any other information that reflects your company's strategy regarding Web services and SOA.

DS

DS Ltd., a software developer in the United Kingdom, offers the *OpenGalaxy* library-automation system as well as a variety of other products for archives and museums. The company did not provide detailed information on how it makes use of Web services throughout its product line. In broad terms, the company indicates, "Web Services [are] extremely important for the development strategy of DS products. It will be the key method of integrating DS library applications with other government and e-commerce applications, which is increasingly being demanded by government initiatives and our customers."

Endeavor Information

Endeavor, an early participant in the VIEWS Initiative (see page 36), offers a suite of library-automation products, including the *Voyager Library Management System, Meridian* electronic-resource management system, the *Discovery:Finder* federated-search environment, and *Journals Onsite.* Endeavor points out that it has been involved in developing applications involving Web services for more than five years. The company uses Web services with several of its products:

- Discovery:Finder (formerly known as ENCompass for Resource Access) relies on Web services as protocols used for searching remote resources in this federated-search environment. Discovery:Finder includes an API based on Web services that libraries can use to create customized XML gateways for access to additional search targets.
- Meridian makes extensive use of Web services internally and for communicating with external applications, such as an ILS. Meridian communicates with the acquisitions module of Voyager and other automation systems using Web services. According to Endeavor, "The ILS Web service contacts Endeavor Meridian, requesting specific acquisitions data for a

particular resource (be it product information, interface terms, license terms or resource information), and Endeavor Meridian responds with the appropriate records."

- Journals Onsite (Formerly ENCompass for Journals Onsite) makes use of Web services.
- Curator, Endeavor's product for creating digital collections, supports both OAI-PMH and SRU/SRW.

To emphasize its strategy, Endeavor states: "Today, any application created or enhanced by Endeavor will incorporate some level of Web services: internally, externally or both."

EOS International

EOS International provides its *EOS.Web* library-automation systems to special libraries, primarily those in corporate libraries. Given that EOS products operate on enterprise networks in corporations, many of which have extensive implementations of SOA, this company makes extensive use of Web services. EOS does not necessarily make great use of library-specific Web services, such as SRU/SRW and OAI-PMH, but rather focuses on custom Web services that operate in conjunction with nonlibrary applications within corporate-enterprise networks. EOS has created interfaces based on Web services for internal human-resource systems, order-processing systems, and Web portals that exchange data relating to library patrons, bibliographic records, and search results. The company creates these customized Web services as part of its consulting and custom-development services.

EOS products are based on Microsoft .NET technologies and make extensive use of Web services, both internally and for communication with external applications, and use SOAP extensively and exclusively for system communication. According to EOS, "EOS.Web Services operate in the background of EOS.Web Enterprise, providing for the sharing of business-process applications from server to server, enabling libraries to retrieve and share search results delivered from the search interface. EOS.Web Services also allows Authority, Bibliographic, and Patron records to be retrieved, added, updated, and deleted via XML."

Ex Libris

Ex Libris offers the *ALEPH 500* library-automation system, the *SFX* link server, the *MetaLib* federated-search environment, the *DigiTool* digital-library platform, and the *Verde* electronic-resource management system.

States the company: "Ex Libris is, and has always been, dedicated to providing its customers with tools to enable integration of its products with other existing library and institutional systems and applications—even before these were known as Web services."

Web services find use throughout the vendor's product line:

- DigiTool employs Web services delivered through SOAP and described through WSDL;
- Verde uses Web services with SOAP for communications with external applications;
- Verde and SFX communicate with each other using Web services transmitted via SOAP;
- both ALEPH 500 and MetaLib offer an X-Server as an API for customers to programmatically access internal functionality of the systems; and
- MetaLib implements SRU/SRW for searching remote resources.

In summary, Ex Libris states: "As can be seen by [the company's] use of Web services and their integration into existing and future products, this is a major factor in the [c]ompany's product-development strategy and the beginning of an industry-wide trend. Verde and Primo were designed and built on [principles] of Service-Oriented Architecture to provide interoperability. Web services [are] the method that was chosen to realize this."

DS
www.ds.co.uk

Endeavor Information Systems
www.endinfosys.com

EOS International
www.eosintl.com

Index Data
www.indexdata.dk

Innovative Interfaces Inc.
www.iii.com

OCLC XSLTProc
http://alcme.oclc.org/xsltproc

OCLC xISBNservice
www.oclc.org/research/projects/xisbn/technical.htm

Index Data

Index Data specializes in developing open-source toolkits that support library applications. The company's products include the *Keystone* digital-library platform, the *YAZ* proxy, the *YAZ* toolkit, and the *Zebra* indexing application and search engine. I noted previously that Index Data's YAZ toolkit is the dominant platform for the creation of search interfaces that operate through Z39.50. The support for SRU/SRW in the YAZ toolkit makes it easy to add this interface to existing Z39.50 applications. The company makes its products freely available as open source but offers optional fee-based support and consulting services.

Innovative Interfaces Inc.

Innovative Interfaces offers the *Millennium* integrated-library system, *Electronic Resource Management*, the *Symposium* institutional-repository platform, and a number of related library-automation products and services. The company makes use of Web services in several aspects of its product line.

In September 2004, Innovative launched its *Inventory Express* product for linking Millennium with vendor applications through an interface of Web services to expedite the process of ordering and transfer of data. Through Inventory Express, libraries can use Web services to conduct business-to-business transactions with vendors, such as Baker & Taylor, BWI, Couts, and Amazon. The company's Electronic Resource Management product supports SUSHI for automatic harvesting of electronic-resource use statistics.

Innovative was not a member of the VIEWS consortium, but this should not be taken as a lack of commitment to Web services. The company has been involved in exploiting Web-service technologies as aggressively as any of its competitors. It characterizes its approach to Web services as, "Our successful implementations show that Innovative is focused on practical applications of Web services that have a real impact on library operations and services."

OCLC

OCLC has been one of the major players in exploring Web-service technologies for library applications. I explained previously that OCLC has developed implementations of OAI-PMH and SRW/SRU, major technologies (for libraries) based on Web services. In addition to these efforts, OCLC has been involved in developing other products that involve Web services. Some of these efforts include:

- *OCLC XSLTProc*, a Web service that performs XSLT processing as a Web service. The focus of this service lies in addressing some of the security risks associated with running XSL stylesheets from unknown sources. (http://alcme.oclc.org/xsltproc)
- *OCLC xISBNservice*, a Web service that receives an ISBN and returns a list of ISBNs associated with related materials, such as other editions of the same work. (www.oclc.org/research/projects/xisbn/technical.htm)

Polaris Library Systems

Polaris Library Systems specializes in library-automation products for public libraries. The company's flagship *Polaris* library-automation system is based on Microsoft .NET technologies. Some of the specific uses that Polaris makes of Web services include its *SiteScan* product for monitoring the health of its customers' systems, which uses Web services to transfer system-status data. The *Polaris Inventory Manager* uses Web services to communicate with the Polaris ILS as it performs inventory tasks. The company also uses Web services to display enriched data content from Baker & Taylor and Syndetic Solutions in its Web-based online catalog products.

According to Polaris, "Web services are important internally as a useful tool for rapid development and deployment of functional content. Web services are important externally to the extent that other vendors with whom we want to interoperate use them. That number is expected to increase."

Sagebrush Corporation

With a focus primarily on K–12 school libraries, Sagebrush Corporation offers a variety of library-automaton products, including its new *InfoCentre* library-automation system as well as older products, such as *Winnebago Spectrum* and *Sagebrush Athena*.

Sagebrush relies on Web-service technologies in the development of its new products, performing its development in Java. According to the company, "SOAP is integral to our client/server application. Both Web clients (via Java servlets) and rich clients (Java Swing) use SOAP for all communications with our server. A Business Service layer exists that abstracts the SOAP calls from upper layers of the client software. The SOAP calls then initiate actions on the server. The results are returned via the SOAP call."

SirsiDynix

The SirsiDynix umbrella covers a wide range of technologies, including those from the former Sirsi and Dynix companies that merged in 2005. Some of the major products include the *Unicorn* and *Horizon* library-automation systems, the *URSA* resource-sharing environment, the *Horizon Information Portal*, the *SirsiDynix Enterprise Portal Solution*, and many other library services. Between the two companies, there are many products that make use of Web services.

One of the products with specific support for Web services is the *Vendor Interface Protocol* (*VIP*) developed by Dynix before Sirsi acquired the company in June 2005. VIP is a Web service in the Horizon acquisitions module that communicates with library suppliers to receive price data and to transfer bibliographic records related to items purchased. VIP is based on the Java and Apache Axis environment.

In summary, SirsiDynix says, ". . . our strategy on Web services is to provide an interface for us to gather information from other vendors, as well as for vendors to gather information from us. Both our current offerings and our future plans support this strategy."

Polaris Library Systems
www.polarislibrary.com

Sagebrush Corp.
www.sagebrushcorp.com

SirsiDynix
www.sirsidynix.com

*Sydney*PLUS
www.sydneyplus.com

Talis
www.talis.com

The Library Corp. (TLC)
www.tlcdelivers.com

VTLS
www.vtls.com

Sydney*PLUS*

Sydney*PLUS* develops library-automation products primarily for special libraries. The company makes some use of Web services, especially in the *Sydney*PLUS *Information Pathfinder Module.* The product uses Web services for language-translation services and for generating RSS news feeds. The company's strategy for Web services is evolving. Sydney*PLUS* indicates its, "Current implementation utilizes XML data transfer to enable integration. Future development will incorporate this into Web services."

Talis

Talis offers the *PRISM* automation system to libraries primarily located in the U.K. This company has been one of the most outspoken organizations regarding the importance of Web services and other "Web 2.0" technologies. Talis points to its *Keystone* product that uses Web services to deliver library functionality to other external systems. The company is working to expose components of library functionality to external systems via Web services, especially for student portals. Talis has performed this integration for multiple universities in the U.K., each using different portal-software products. In the public-library environment, Talis uses Web services for integration with CRM systems and other e-government systems of local councils. In higher-education libraries, Talis has employed Web services for interfaces with student-registry systems with the Talis LMS.

The company summarizes its approach as follows: "Web services, SOA, and Web 2.0 are core to the Talis development strategy of making our sub-system components available via Web services for the purpose of integration and for use by developers (both Talis and third-party) for remixing into new and innovative applications. This forms the heart of the Talis Library platform."

The Library Corporation (TLC)

TLC offers the *Library.Solution* and the *Carl.X* library-automation systems, *Online Selection & Acquisitions*, as well as a number of other library-automation products. The company indicates it uses Web services within its *AquaBrowser Library* and *Endica* search platforms and interfaces. The Online Selection & Acquisitions product—an ASP-hosted application that performs collection-development and acquisitions functions—uses Web services for real-time communication with library suppliers' business systems for pre-order searching, electronic ordering, and harvesting data for price lists. TLC is also developing interfaces for Web services between the budget and finance functions of Online Selection & Acquisitions with external financial-management information systems.

The Library Corporation sees Web services and SOA as important technologies in support of its product strategy to create suites of stand-alone components built from the ground up for interoperability. The company sees a future where more ILS implementations are hosted remotely with browser-based applications. TLC expresses its view toward this technology as, "Web Services [are] a key, if not the key, to TLC's development strategy. Our corporate direction is focused on developing strong stand-alone components that are wired for interoperability from the outset."

VTLS

VTLS offers the *VIRTUA* library-automation system, the *VITAL* institutional-repository product based on *FEDORA*, and will soon launch its *VERIFY* electronic-resource management system.

The company makes extensive use of Web services via its products. VITAL is based on FEDORA, which makes extensive use of Web services, both internally and as it communicates with external applications. VITAL supports the OAI-PMH for metadata exchange and SRU/SRW. VIRTUA offers an XML gateway based on Web services and supports NCIP, which is XML based. VERIFY, scheduled for release in 2006, will employ SOAP for the transfer of ONIX for Serials messages and will include support for SUSHI and SRU/SRW.

VTLS President Carl Grant expresses his company's view on this technology: "The flexibility that the loose-coupling of Web services gives our products only furthers our intentions to continue delivering products that allow our customers to use and extend our solutions in new and creative ways. The profession is yielding a far more technically minded generation of information architects—we hope to give them tools to realize their ideas." As further evidence of the company's commitment to Web services, Grant points out that VTLS founded the VIEWS consortium.

The Potential of Web Services for Libraries

As Web services mature in the broader IT industry, it is clear the library field has increased adoption in library-specific applications. The nonscientific survey of library-automation vendors conducted for this report reveals that many library-automation companies have, at least to some extent, begun introducing some aspect of Web-service technologies in their products. Some have already embraced Web services as the center of their future development strategy.

Libraries must also consider how the pervasive trend toward Web-service-based technologies impacts their own strategies. As with any technology, libraries need to consider whether and how Web services can be used to help them further their strategic goals. As libraries become increasingly more involved in digital-information delivery and reliant on computer technologies, decisions regarding these technologies rise to a level of utmost strategic importance. Information in this report should help library decision makers realize that Web services need to be considered as part of a library's broad technology strategies.

The capabilities that Web services provide—toward helping diverse computing platforms interoperate—parallels organizational interests in partnerships and cooperative arrangements. If a library aims to become more involved with its surrounding organizations, then Web-service technologies might be used to enable that goal. Examples of applications based on Web services that might further an organization's goals could include:

- real-time interaction among library-automation systems and business systems of a library's parent organization;
- real-time interaction among library-automation systems and library suppliers or other business partners;
- blending of library services into campus or municipal portal environments;
- insertion of library services and content into courseware-management systems or other learning environments;
- blending of content from external sources into library interfaces; and
- delivery of library services and content to library users through nontraditional channels.

Libraries that consider Web services of increasing importance to their organizations may want to:

- assess their current automation environments to learn the extent to which Web services are supported;
- assess the abilities of the systems of partner organizations to support Web services in areas of mutual interest;
- focus new projects that involve local development on leveraging XML and Web-service technologies;
- convey the library's interest in Web services to automation vendors;
- include support for Web services as a requirement for future software acquisitions as appropriate; and
- assess awareness of library technical staff on Web-service technologies and provide education and training opportunities as needed.

Standards and interoperability have long been valued within the library-automation community. In previous eras, the focus was on interoperability among the various applications within a library or with the systems of other libraries. Now that interest often expands to interoperability with nonlibrary applications. Web services provide the opportunity to move beyond library-specific standards and protocols into a technical arena shared among a very broad range of industries and organizations.