Chapter 7

INSTITUTIONAL REPOSITORY SYSTEM OVERVIEWS

The number of institutional repository systems is small but appears poised for tremendous growth in the coming years. This chapter contains brief descriptions of many of the IR systems in use, followed by systems that are in development or not yet available outside their home institution. This list, however, is not exhaustive.

For more information on any of these IR systems, check the system's official website and read some of the articles and documents listed in Appendix A. Speaking directly to current users of the systems also is helpful. Each description is followed by a few examples of live implementations. Appendix B provides more examples.

ARNO

The Academic Research in Netherlands Online project (ARNO) began in September 2000 through a collaboration of Tilburg University, University of Twente, and University of Amsterdam. An outcome of the project is an open-source repository system, also named ARNO.

See "ARNO Deliverables" at www.uba.uva.nl/en/projects/arno.

The software is just one of several of the project's derivatives, which include some interesting research into aspects of digital scholarly communication, including attitudes and perceptions of electronic publishing and peer review. Some of these deliverables have been translated into English, but others are available only in Dutch.

ARNO, www.uba.uva.nl/arno

ARNO downloads, http://arno.uvt.nl/~arno/arnodist

ARNO is written in Perl and requires an Oracle 8i or higher relational database. It runs on Linux or Solaris operating system and uses an Apache Web server. The system does not have an end-user interface but relies on third-party search services, including I-Tor (see below), iPort, and DLXS, to harvest and deliver the content of the ARNO repositories.

iPort, http://oclcpica.org/?id=106&ln=uk

DLXS, www.dlxs.org

ARNO is designed with a flat structure of independent archives that can be combined to form an IR. The system provides the option of depositing the documents themselves into the repository, or depositing or harvesting just a URL along with the associated metadata, that points to the document in another location.

In addition to contributors who make deposits and provide the initial metadata, the system supports metadata editors and provides administrators with the ability to function as archive editors. ARNO can work with an LDAP server to control user authentication.

Dublin Core is the basic metadata schema used by ARNO. By using XSLT style sheets, though, other metadata models can be supported, while still remaining OAI compliant.

XSLT: eXtensible Stylesheet Language Transformation

The system is file-format neutral, but a registry of accepted file types can be created for each archive. Any file types not in the registry are refused submission to the repository. ARNO uses its own native system, called ARNOdocurl, to create persistent URLs for each document file and index page.

Examples of ARNO installations:

- Degree: Dutch Electronic Grey Files on Economics—http://drcwww.uvt.nl/dbi/instructie/degree/ (iPort interface on top of ARNO repositories)
- University of Amsterdam's Digital Academic Repository—http://dare.uva.nl (DLXS interface on top of ARNO repositories)
- University of Tilburg's Master Theses—http://dbiref.uvt.nl:4242/iport?db=scr (iPort interface on top of ARNO repositories)

bepress

bepress, www.bepress.com

Three University of California, Berkeley professors, Robert Cooter, Aaron Edlin, and Benjamin Hermalin founded the Berkeley Electronic Press, or bepress, in 1999. After endeavoring to rescue a struggling economics and law journal, the professors were quite frustrated when a commercial publisher took over the journal and raised the price by 400%.

Cooter and his colleagues decided to build an electronic publishing system that would address the inefficiencies of the current publishing system, including exorbitant subscription costs and significant delays between submission and publication.

The system, EdiKit, automates most of the back-end functions involved in publishing an electronic journal, including submission tracking, communication between author, editor, and reviewers, and journal layout configuration.

The result is a nicely formatted, online journal, hosted entirely by bepress. Bepress launched its first journal, *The B.E. Journals in Macroeconomics*, in November 2001 and now publishes about 25 titles from its website.

Through a partnership with the California Digital Library (CDL), bepress successfully modified EdiKit into an IR system, which was named Digital Commons. As with the journals, each instance of the repository system is entirely hosted by bepress in California.

The cost of the hosted system is determined by an institution's size and type, currently ranging from $5,000 to $35,000 per year. The annual fees include setup costs, technical support, robust backup systems, migration assistance, and a pledge of ongoing development of the system.

bepress repository technology, www.bepress.com/repositories.html

Should an institution decide not to renew its license, the entire content of its IR will be transferred from bepress back to the institution.

(Source: www.bepress.com/repositories.html)

Bepress' market includes educational institutions, consortium, and discipline-specific organizations that do no want or cannot afford the staffing, equipment, and support required to establish and maintain an IR. Bepress points out that "MIT, with 9,200 faculty and staff and one campus, will spend more to support its own open source implementation than the University of California, with 10 campuses and 159,000 faculty and staff, pays The Berkeley Electronic Press to operate its repository."

Proquest, www.umi.com/umi/digitalcommons

The marketing and sales of the Digital Commons system itself has been outsourced recently to Proquest Information and Learning Co.

Digital Commons accepts content in any format, including multimedia but has no streaming media support. Any Word or RTF document can be automatically converted by the system on submission into PDF. Moreover, a template-driven

cover page can be added to each PDF document, which includes a full citation and the document's URL.

An associated file feature provides for the inclusion of supplemental materials, such as datasets and previous versions. The repository uses a hierarchical URL system built on a base domain, which is owned by the institution.

bepress demo site, http://umi.bepress.com/dartmouth

Metadata in the Digital Commons system is fully customizable at the collection level. The customized metadata fields are not added to the Dublin Core export format, so that the system can remain OAI compliant.

Controlled vocabularies can be uploaded to create drop-down subject term menus, such as the *Journal of Economic Literature* classification system and the Current Index of Legal Periodicals Subject Headings list.

In addition to OAI harvesting, the metadata can be exported in XML and transferred to third-party indexing services, such as Thomson ISI's Web Citation Index. The system supports browsing and full-text searching, as well as a service that alerts users when future documents match the current search terms.

Thomson ISI, www.isinet.com

Bepress' Digital Commons can sustain potentially unlimited levels of community hierarchies, which was necessitated by the complex organizational structure of the University of California system.

Customization of the user interface is possible throughout all levels of the system, down to even an individual deposit, through the use of templates. Access is controlled by either domain/IP restrictions or login/password system and is set at the collection level. Currently the system cannot authenticate users through an LDAP or other external authentication system.

Examples of bepress installations:

- Boston College—http://dissertations.bc.edu
- New England Law Library Consortium—http://lsr.nellco.org
- University of California—http://repositories.cdlib.org/escholarship

CDSware

The CERN Document Server Software, CDSware, was created at the European Organization for Nuclear Research (CERN), specifically to provide for its growing collection of more than 650,000 bibliographic records and 320,000 full-text documents. CERN has since released the CDSware source code as open-source software, distributed under the GNU General Public License.

CDSware, http://cdsware.cern.ch

GNU is a recursive acronym for "GNU's Not UNIX"; it is pronounced "guh-noo." www.gnu.org.

CDSware acts simultaneously both as a library catalog of bibliographic records, as well as a document repository. For example, at CERN, CDSware received about 1,000 new preprints per week, of which only 5% are full-text documents from the CERN community. About 70% are metadata extracts from arXiv and 25% are metadata extracts from 80 other repositories (Vesely, 2002). The metadata is harvested using CDSware's built-in OAI harvester.

CDSware download, http://cdsware.cern.ch/download

CDSware runs on Linux and Unix and relies on Apache, MySQL, PHP, Python, and WML (Website META Language). Unlike most of the other IR systems, CDSware uses the standard MARC21 metadata schema.

The CDSware system has an easily configurable Web interface. On submission, the content owner has the option of converting the document into different formats, including HTML and XML.

CDSware demo site, http://cdsware.cern.ch/demo

The system retains the native, as well as, converted formats to provide as many avenues for future format migration as possible. Each collection has the potential for any number of designated moderators, reviewers, and approvers.

CDSware contains a powerful and unique search engine. The search interface was designed with "Google-like guidance" (Vesely, 2002). All the MARC metadata fields are searchable and browsable. Full-text searching is possible for any PostScript, PDF, and Microsoft Word, Excel, and PowerPoint files.

In addition, all of a document's references are indexed separately. As a result, not only are the reference citations searchable, but the system creates a reverse search engine feature that can identify any documents that cite the original document.

Although the CDSware code is free, CERN offers an optional maintenance contract. How this contract works for institutions outside Europe is not clear.

Example of CDSware installation:

- CERN Document Server—http://cds.cern.ch

Documentum

Documentum,
www.documentum.com

Documentum is a commercial content management software that will be used by OhioLink and University of Southern California to support their institutional repositories. Included among the rather wide array of functionalities within Documentum are the core services of an IR, which are mainly contained in the suite of digital asset management products. In addition to the IR functions, Documentum can supply a suite of useful add-on functions discussed below.

Documentum can run on Sun Solaris, Microsoft Windows, IBM AIX, or HP-UX operating systems. Institutions can opt to use an Oracle, Microsoft SQL, IBM DB/2, or Sybase database.

Many different technical support options exist, ranging from full enterprise support to migration-only support. The system can perform as a single repository on a single server as well as several repositories distributed across several servers.

Depositors have the option of using a Web-based or Microsoft Windows-based client. A distinct advantage of using the Windows-based client is that it can be integrated into common desktop applications, such as Microsoft Office, Windows Explorer, and Acrobat Business Tools.

Section 508 website,
www.section508.gov/index.cfm

National Federation for the Blind certification,
www.nfb.org/seal/intro.htm

The Windows client must first be loaded onto a desktop computer before it can be used, but the Web-based client is available to any computer. Both clients have interfaces that support Unicode, comply with Section 508, and have received certification from the National Federation for the Blind. The interface also is available in French, German, Spanish, Italian, Japanese, and Korean.

Documentum is format neutral, so it can handle the ingest, storage, and distribution of all format types. To provide for multimedia content, Documentum supports the integration of a streaming server and the automatic creation of thumbnails. Multimedia files can be transformed on demand, such as Flash animation to MPEG video.

Documentum automatically converts text documents into PDF on submission, as well as applies optical character recognition (OCR) to scanned documents. Full-text searching is possible across several file types, including PDF, Microsoft Office, TIFF, and ASCII.

In addition to a sophisticated versioning system, Documentum also has check-in and check-out functions to support collaborative authoring. Each collection within the repository also can design its own elaborate workflow process to include any number of people in the editing and approval process for each document submission.

A batch-loading system also is available. Documentum has a range of access control options, including LDAP, SSL, and digital certificates.

SSL: Secure socket layers

Example of Documentum installation:

- OhioLink—http://dmc.ohiolink.edu (several of these databases are built on Documentum)

DSpace

Released in November 2002, DSpace is an end-to-end open-source institutional repository system. The result of a $1.8 million project between MIT Libraries and Hewlett-Packard Co., DSpace is an "electronic system that captures, preserves and communicates the intellectual output of MIT's faculty and researchers."

DSpace, www.dspace.org

(Source of quote: http://libraries.mit.edu/dspace-mit/what/definition.html)

Although designed specifically for MIT's environment, the source code was released as open source to support adoption by other institutions. Through funding from the Andrew W. Mellon Foundation and the Cambridge-MIT Institute, a federation of eight institutions was established in the fall of 2002 to test the adaptability of the system to needs and environments of institutions outside MIT.

The eight institutions are Cambridge University, Columbia University, Cornell University, MIT, Ohio State University, and the Universities of Rochester, Toronto, and Washington.

The source code for the system, available from Sourceforge, has been downloaded more than 8,400 times as of February 2004. Although DSpace has numerous installations around the world, MIT's remains one of the largest. Although owned jointly by MIT and Hewlett-Packard, the source code's BSD distribution license permits any institution or person to download, run, and modify the code.

BSD: Berkeley Software Distribution

Neither MIT Libraries nor Hewlett-Packard provides any formal support for the DSpace system. Users must rely on the system's documentation, technical list serve, and communication with other users of the system for support.

DSpace download site, http://sourceforge.net/projects/dspace

DSpace is written in Java and runs on any Unix-type operating system, including a laptop. It runs on an Apache Web server and uses a PostgresSQL database. MIT Libraries has posted examples of hardware configurations on its DSpace website.

DSpace hardware configs, http://libraries.mit.edu/dspace-mit/what/faq.html

DSpace is format neutral, meaning that it accepts submissions of any format type. Its designers recognize that making a preservation commitment to all formats is not feasible, so DSpace provides for the designation of three levels of format support; supported, known, and unknown. MIT defines the three categories as follows:

- Supported—fully support the format
- Known—recognizes the format, but cannot guarantee full support
- Unsupported—cannot recognize the format

(Source of quote: MIT's DSpace Format Support policy, http://libraries.mit.edu/dspace-mit/mit/policies/format.html)

Support is defined by MIT as to "make usable in the future, using whatever combination of techniques (such as migration, emulation, and so on) is appropriate given the context of need." Each DSpace instance is not required to follow the same route as MIT.

What constitutes support can and should be locally defined. Also, the determination of what formats receive which format support designation is completely customizable through a Web interface in the administration system.

Dublin Core, http://dublincore.org/documents/library-application-profile

DSpace's metadata schema is a qualified Dublin Core, based on the Dublin Core Libraries Working Group Application Profile. Although the metadata schema is locally customizable through an administrative Web interface, any changes to the metadata schema affect all current and potential collections.

DSpace is built on the concept of communities and their collections. The communities represent organizational units, such as academic and administrative departments, research centers, laboratories, or even single individuals. Currently, the organizational hierarchy has only two levels, so representing organizational units as subgroups of larger units is not possible.

Version 1.2, due out in May 2004, will support expanded hierarchies. A community can create any number of collections, into which deposits are made. A deposit can consists of multiple files in any format.

DSpace organizes users into e-people and groups. Once an e-person is created, the account must then be identified as a member of a group, because only at the group level are the permissions of read, write, add, and remove assigned. Authorization of a group can be done at a community or collection level.

The out-of-box DSpace system creates a minimally customizable homepage for each community and collection. However, with some HTML and Java skills, the user interface can be completely redesigned. For a comparison, see the out-of-box interface at MIT and the DSpace interface re-designed by the Rochester Institute of Technology (RIT).

DSpace at RIT, https://ritdml.rit.edu/index.jsp

DSpace uses the Lucene freeware search engine to support searching across several of the metadata fields. Full-text searching of the documents themselves will be possible in Version 1.2.

CNRI Handle System, www.handle.net

The advanced search feature supports Boolean searches in a single collection, a community, or across the entire repository. DSpace is OAI-compliant and uses CNRI Handles to create persistent URLS.

Examples of DSpace installations:

- Cambridge University—https://www.dspace.cam.ac.uk
- MIT—https://dspace.mit.edu/index.jsp
- University of Toronto—https://tspace.library.utoronto.ca/index.jsp

EPrints

EPrints, http://software.eprints.org

In 2000, a professor at the University of Southampton, U.K., Stevan Harnad, began a project that resulted in the open-source EPrints software. Harnad's graduate student, Rob Tansely, wrote the first version of the Perl-based software before joining the Hewlett-Packard development team that created DSpace. Christopher Gutteridge has continued to develop the EPrints software, which was at Version 2.3.3 as of March 2004.

With the release of Version 2.0, the software changed its official name from EPrints to GNU EPrints to reflect that it is now part of the GNU Project. The requirements of GNU include that the system be free and dependent only on other free software.

GNU Project, www.gnu.org/gnu/thegnuproject.html

Stevan Harnad, a vocal and long-time advocate of open access, created EPrints specifically to support self-archiving. EPrints can be used for a single digital collection, such as an electronic preprint archive, or several EPrint installations can be combined to make up an institutional repository, such as the more than 18 EPrint instances that make up the California Institute of Technology's repository, CODA. More repositories in full production run on EPrints than any other open-source or commercial system.

CODA, http://library.caltech.edu/digital

EPrints is written in Perl and uses an Apache Web server with a MySQL database. It can run on a Linux, GNU, or Solaris operating system. The out-of-box metadata schema used by EPrints is Dublin Core, but the metadata can be customized for each type or genre of document, such as book chapter, thesis, and preprint.

EPrints download site, http://software.eprints.org/download.php

Authors can submit documents in any format, but the repository administrator can restrict the formats that the repository will accept. Once a document has been submitted, it remains in a submission buffer until someone such as an editor decides to approve, reject, or return the submission to the author for modifications.

EPrints demo site, http://demoprints.eprints.org

EPrints supports discovery through browsing and searching. Users can browse either by date or by subjects arranged hierarchically. The difference between basic and advanced searching is only the range of potential fields to search on. EPrints is OAI-compliant, and the system assigns all submissions a persistent URL using CNRI Handles.

CNRI Handle System, www.handle.net

The user interfaces as well as the administrative interfaces are Web-based and easily customized through the use of templates. EPrints is designed to use a user name and password system for user authentication; however, LDAP and other external authentication methods can be used in its place.

The design for Release 2.0 of EPrints included an international focus. All metadata is stored using UTF-8, so it can handle non-Latin characters. Phrases displayed in the user interface are stored in XML, where they can be easily located and translated. A single instance of EPrints can host multiple versions of the interface in different languages, such as the University of Montréal's Papyrus, which has both an English and French version.

Papyrus, http://papyrus.bib.umontreal.ca

Examples of EPrints installations:

- Lund University, Sweden—http://eprints.lub.lu.se
- University of Bologna, Italy—http://amsacta.cib.unibo.it
- Weizmann Institute, Israel—http://wisdomarchive.wisdom.weizmann.ac.il:81

Fedora

Flexible Extensible Digital Object and Repository Architecture (Fedora) has origins that extend back to the work of Sandra Payette and Carl Lagoze at Cornell University in 1998. After the University of Virginia succeeded in implementing the Fedora architecture into a working prototype, Virginia obtained a

Fedora, www.fedora.info

$1 million grant from the Andrew W. Mellon Foundation to collaborate with Cornell University to build an open-source, robust Fedora implementation.

The Fedora repository system is a work-in-progress. Although still in the first of three project phases, source code is available that provides the base modules of what will be a robust, sophisticated open-source digital object repository.

Version 1.2 of the source code, which is distributed under the Mozilla Public License, is not an out-of-box repository solution. It supplies the inner workings but lacks many of the out-of-box features of other IR systems, such as an end-user interface and complex access controls.

VTLS Inc., www.vtls.com

Institutions with strong IT support can build on the current Fedora code to create a fully functional IR system. Alternatively, VTLS Inc. has created Vital, a commercial system that provides a Windows-based graphical interface, and search and content management services to the Fedora repository architecture. VTLS also offers fee-based installation, training, support, and hosting services for Fedora users.

The present Fedora source code contains three application program interfaces (APIs) exposed as Web services, which are described on the Fedora website as:

- Management API (API-M)—defines an interface for administering the repository. It includes operations necessary for clients to create and maintain digital objects and their components. API-M is implemented as a SOAP-enabled Web service.
- Access API (API-A)—defines an interface for accessing digital objects stored in the repository. It includes operations necessary for clients to perform disseminations on objects in the repository and to discover information about an object using object reflection. API-A is implemented as a SOAP-enabled Web service.
- Access-Lite API (API-A-Lite)—defines a streamlined version of the Fedora Access Service that is implemented as an HTTP-enabled Web service.

Fedora download site, www.fedora.info

The system is written in Java and built only on software available as open source, including Apache and MySQL. Fedora can be deployed on Windows 2000, NT, and XP, Solaris, and Linux.

METS, www.loc.gov/standards/mets

Fedora's content, referred to as digital objects, is stored using an extended version of the Metadata Encoding and Transmission Standard (METS). Every object has a Dublin Core record, which is harvestable by OAI harvesting services. Fedora also has a batch utility that supports bulk loading of objects.

Administrative functions of a Fedora repository, such as creating and modifying digital objects, are handled through a Java Swing (a graphical user interface Java toolkit). Searching can be performed across about two dozen indexed metadata fields. Some fields are directly from the Dublin Core record (such as title, creator, and contributor) and others are Fedora-specific fields (such as Fedora object type and content model).

The Virginia/Cornell Fedora project is *not* the same as the Red-Hat sponsored Fedora project, which aims to build an operating system entirely from free software.

Fedora uses its own persistent identifier system (Persistent ID or PID), which is assigned to each digital object.

Unlike many IR systems, Version 1.2 of Fedora can only restrict access to the repository content by IP address. More robust access controls, however, are planned for the second phase of the project.

Fedora does contain a fully functional versioning system, which is lacking in many other IR systems. Moreover, Fedora was designed to easily integrate tools and services that can handle and manipulate the digital objects.

The three services that come bundled with the Fedora software are an XSLT (eXtensible Stylesheet Language Transformation) service, a PDF conversion service, and an image manipulation service.

Examples of Fedora installations (not IRs):

- Northwestern University: Andreas Versalius' *De Humani Corporus Fabrica*—http://versalius.northwestern.edu (customized interface on top of a Fedora repository)
- Rutgers University: Eagleton Center for Public Interest Polling—www.scc.rutgers.edu/eagleton (customized interface on top of a Fedora repository)
- University of Virginia—http://icarus.lib.virginia.edu:8080/fedora/search

I-Tor

I-Tor is an open-source repository system created by The Netherlands Institute for Scientific Information Services (NIWI), an institute of the Royal Netherlands Academy of Arts and Sciences. I-Tor functions differently from other IR systems, such as GNU Eprints and DSpace, in that it can be used simultaneously as a repository, information portal, and content management system.

I-Tor, www.i-tor.org/en/toon

Essentially, I-Tor is a Web server that pulls data from existing distributed sources, including databases, OAI-compliant archives, and local file systems, and then renders the data in a consistent-looking website.

The documents themselves can remain in their native locations, or, in the absence of an existing archive, I-Tor also can function as a repository. Explained another way, "I-Tor is a medium by which various types of information can be presented through a Web interface, no matter where the displayed information comes from or what its format is." It allows an institution to combine existing, distributed repositories into a single information portal.

(Source of quote: I-Tor flyer, dated August 2003 www.i-tor.org/en/flyer_i_tor_august_2003/toonplaatje]

I-Tor is written in Java and relies only on open-source components, including MySQL. The code is distributed under the GNU General Public License. The repository is format neutral but can be set to accept submissions of only certain file types.

I-Tor download site, http://sourceforge.net/projects/i-tor

All the text content linked by I-Tor is automatically full-text searchable through the I-Tor interface. Relationships between objects can be manually added or automatically derived from pre-existing relationships, such as objects in the same database. I-Tor does not impose any particular metadata schema, allowing the pre-existing repositories to retain their native metadata models.

The interface is customizable through the use of templates. The administrative functions, such as adding or deleting objects, is accomplished through a Web interface. Access, submission, and editing rights are controlled through a user login and password system.

I-Tor imposes no hierarchical structure to the IRs nor enforces any particular workflows. Users can continue to build their pre-existing repositories as they have done in the past, and I-Tor acts as an OAI service provider to harvest and expose the repositories' metadata, as well as make them full-text searchable.

Examples of I-Tor installations:

- European Visual Archive—http://evamp.itor.org/en/toon

- Dutch Digital Academic Repositories (DARE) website is an I-Tor installation—www.daretodemonstrate.nl/en/toon
- I-Tor's website is an I-Tor installation—www.i-tor.org/en/toon

Systems in development

In addition to the IR systems described above, several are either in development or are not yet available outside their home institutions. Within the next year or so, the following systems will likely be available for public use.

eDoc-Server

The Max Planck Society (MPS) launched its homegrown IR system, eDoc-Server, in October 2002. Development of eDoc-Server was done specifically to support MPS' strong open-access commitment. Due to many unspecified "pragmatic reasons" the eDoc-Server code is not yet available as open source (Beier, 2004).

MPS eDoc-Server, http://edoc.mpg.de

eDoc-Server offers the 80 institutes that make up the MPS with an IR for both their full-text documents, as well as bibliographic metadata for content stored outside the system. A wide variety of document types can be submitted into eDoc-Server, the access to which can be controlled at a refined level. MPS is in the exploratory phase of a project to adapt eDoc-Server to provide for the storage and distribution of primary source materials.

MPS' eDoc-Server is written in Perl and built on Apache and a PostgreSQL database. The system runs on Linux. It uses both Dublin Core and a custom eDoc XML exchange metadata schemas. The system is OAI-complaint, and because it is based on OpenURL, it can work with context-sensitive linking services, such as ExLibris' SFX.

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

SFX, www.exlibrisgroup.com/sfx.htm

The system organizes materials into collections. Each collection has at least one moderator and one authority, who check the quality and appropriateness of the deposit. The role of a metadata editor also is included.

The user and administrative interfaces are all Web-based. The system includes full-text, simple, and advanced searching, as well as browsing by institute, collections, and authors.

To guarantee the integrity of the citation for any documents in eDoc-Server, no version can replace a previous one. Instead all versions are stored together, with a document ID suffix that indicates the relationship between versions.

Example of an eDoc-Server installation:

- Max Planck Society—http://edoc.mpg.de

MyCoRe

MyCoRe is an open-source system in development through a collaboration of several universities and institutes in Germany and Sweden. The MyCoRe project is still in the design phase and no expected date of completion is provided on the project's website. Once complete, the open source code will be distributed under the GNU General Public License from Sourceforge.

MyCoRe, www.mycore.de/engl

MyCoRe download site, http://sourceforge.net/projects/mycore

The MyCoRe project's roots go back to the University of Essen's MILESS project, which created an infrastructure to store and distribute collections of multimedia teaching and learning materials.

Since MILESS was developed specifically for the local needs of Essen, adapting the source code for other environments was not easy. The MyCoRe project builds on the experiences of the MILESS project, but with the specific goal to create a digital library and archive solution that can be locally adapted easily.

MyCoRe is being written in Java and currently uses an IBM Content Manager and IBM DB2 as its database backend. It can run on Windows 95, 98, and 2000, AIX, Linux, and SunOS/Solaris. The system will be OAI and Z39.50 compliant. Substantial technical documentation is available in German from the project's website.

MILESS, http://miless.uni-essen.de

OPUS

The University of Stuttgart created Online Publications for the University of Stuttgart (OPUS) in 1998 with funding from Deutsches Forschungsnetz and Bundesministerium für Bildung und Forschung.

The goal of the project was to create a system "to enable every university member (staff or student) to publish, retrieve and use electronic documents via the World Wide Web."

OPUS (and source of quote), http://elib.uni-stuttgart.de/opus/doku/english/index_english.php

The University of Stuttgart continues to expand the OPUS system, with development emphasis on digital signatures, electronic payment systems, multimedia content support, and ETDs.

All documents in OPUS are stored as PDFs with Dublin Core metadata. The system is available for use at other institutions and has been or is being adopted by several universities in Germany.

Examples of OPUS installations:

- University of Stuttgart—http://elib.uni-stuttgart.de/opus/doku/english/index_english.php
- University of Constance—www.ub.uni-konstanz.de/kops

Function-specific systems

A word of caution: if you have a single, precise use, implementing a full-blown IR system may not be necessary. You may be able to find a focused system that can fit your precise needs in a way that a general-purpose IR system cannot.

For instance, if you need a system only to support electronic portfolios, take a look at ePortfolio, an open-source system developed at the University of Minnesota, Duluth. Or investigate some of the other projects of the Open Source Portfolio Initiative (OSPI) community.

ePortfolio, http://eportfolio.d.umn.edu

OSPI, www.theospi.org

If you only want to create an ETD collection, then examine Virginia Tech's ETD database (ETD-db) or some of the work of other members of the Networked Digital Library of Theses and Dissertations (NDLTD).

Virginia Tech's EDT-db, http://scholar.lib.vt.edu/ETD-db

NDLTD, www.ndltd.org

If you do not want to use an open-source system, commercial ETD services are available, such as the one provided by bepress.

bepress, www.bepress.com

Do not settle too quickly for a general-purpose IR system when you have precise needs. You may find just what you are looking for from the open-source community or a vendor who is willing to customize their product line to fit your needs.

Future IR system vendors

In the next few years the marketplace of institutional repository systems will greatly expand, particularly in the number of commercially available products. Likely entrants into the IR marketplace will probably come from three existing sectors: integrated library system (ILS) vendors, digital asset management (DAM) system vendors, and online electronic book system vendors.

Companies from all three sectors already market products and services that are similar to an IR system that, with a few modifications, could be retooled to fit the needs of the IR market.

Integrated library system (ILS) vendors

In 2000, integrated library system (ILS) vendors began to build and market digital library systems. This expansion of their product lines was due in large part to the saturation of the ILS market since most of the libraries in need of an online catalog had already purchased one.

ExLibris DigiTool, www.exlibrisgroup.com/digitool.htm

Endeavor EnCompass, http://encompass.endinfosys.com

These digital library systems, such as ExLibris' DigiTool and Endeavor's EnCompass, offer many of the same features and functionalities required in an institutional repository system. They support the creation and distribution of collections of a variety of digital files, with a strong emphasis on robust metadata and access controls.

The major difference between a digital library system and an institutional repository system is the intended content provider. Systems such as DigiTool and EnCompass were built with the assumption that librarians would populate the collections. Consequently, both DigiTool and EnCompass require a client to be installed on the computers of all potential depositors.

This setup is certainly possible when working with a finite number of librarians, but it can be quite an undertaking when the potential depositors include hundreds of faculty members or thousands of students.

The submission interface is another aspect of a digital library system that would require retooling before it can be used to support an IR. The submission interface of both DigiTool and EnCompass were designed specifically for librarians, and without a solid understanding and expertise of metadata schemas, particularly Dublin Core, the submission process could significantly deter the average user.

(Source: Information taken from presentation of March 4, 2004 to the Minnesota Digital Library Consortium by Michael Kaplan and Stan Smith of ExLibris.)

ExLibris has already begun to retool its DigiTool product for IR use. Version 2.4 will include a Web-based professorial interface through which faculty can deposit their documents. OAI compliance and SSL and LDAP for authentication also are planned.

Digital asset management (DAM) system vendors

Digital asset management (DAM) systems are generally broad, expensive systems to manage the storage and sharing of digital files at large organizations. Most have been designed specifically for use in corporate settings to realize efficiencies in the management of digital documents.

IDC: http://itresearch.com/home.jhtml

In 2001, IDC, a firm that analyzes and predicts technology trends, estimated that "an enterprise employing 1,000 knowledge workers wastes at least $2.5 to

$3.5 million per year searching for nonexistent information, failing to find existing information, or recreating information that can't be found" (Feldman & Sherman). These types of findings have spawned a market for DAM systems that can recoup some of this lost time and money.

The differences between an IR and a DAM system are subtle. The main distinction is the intended audience of the material. Organizations establish a DAM to facilitate the secure storage, retrieval, and sharing of documents for people internal to the organization—in other words, a sophisticated intranet.

Although a document in a DAM system may be accessible to anyone over the Internet, the vast majority of the documents are for authorized, known users. As a result, the pricing of DAM systems is often based on the number of authorized or simultaneous users.

On the other hand, large portions, if not all, of an IR are accessible potentially to the world. Although the administrators or editors of IR systems carefully control who is depositing documents *into* the repository, the IR system facilitates distribution of those documents to any number of unknown people.

DAM systems' focus on internal users leads to a user interface that is inappropriate for external users or lacks the elegance and sophistication that most organizations would want the outside world to see.

The boundaries between DAM systems and IRs are already beginning to blur, and the two technologies may become synonymous in the next few years. Documentum, described earlier in this chapter, is in fact a DAM system. Another example is Xerox's DocuShare, which is being used by the University of South Australia to create online collections for the Bob Hawke Prime Ministerial Library.

DocuShare, http://docushare.xerox.com

Bob Hawke Prime Ministerial Library, http://docushare.unisa.edu.au/docushare/dsweb/HomePage

Online electronic book vendors

Online electronic book vendors also have a product that can be retooled to serve the IR marketplace. Online electronic book systems, such as ebrary and netLibrary, already have many of the core IR functions, including access control and a robust discover mechanism.

Consequently, the vendors can retool their e-book systems to fit the needs of an IR system. Ebrary announced an institutional repository pilot program in April 2003. Participants receive 500 MB of free storage, co-branding of the interface, and simultaneous, multiuser access.

ebrary, http://corp.ebrary.com/libraries/ir.jsp

netLibrary: www.netlibrary.com