Building Digital Libraries from Simple Building Blocks

Authors

Hussein Suleman is a Senior Lecturer in Computer Science at the University of Cape Town, Cape Town, South Africa.

Edward A. Fox is a Professor of Computer Science and Director of the Digital Library Research Laboratory at Virginia Tech, Blacksburg, VA, USA.

Rohit Kelapure and Aaron Krowne are M.S. students and Ming Luo is a Ph.D. candidate in Computer Science, all of whom are members of the Digital Library Research Laboratory at Virginia Tech, Blacksburg, VA, USA.

Keywords

Components, digital libraries, interoperability, open archives, metadata harvesting.

Abstract

Metadata harvesting has been established by the Open Archives Initiative (OAI) as a viable mechanism for connecting a provider of data to a purveyor of services. The Open Digital Library (ODL) model is an emerging framework which attempts to break up the services into appropriate components based also on the basic philosophy of the OAI model. This framework has been applied to various projects and evaluated for its simplicity, extensibility and reusability to support the hypothesis that digital libraries (DLs) should be built from simple Web Service-like components instead of as monolithic software applications.

Introduction

Work in the field of digital libraries was launched in the early 1990s (Fox, 1993). Since then there has been a rapid expansion of research and development, integrating work from related fields and involving hundreds of projects (Fox and Sornil, 1999; Fox and Urs, 2002). Yet, it still is rather difficult to build a digital library. This fact suggests that a new approach, building upon earlier research, and integrating key concepts from modern software engineering practice, is needed. Accordingly we provide background regarding related work on the Open Archives Initiative (OAI), introduce the Open Digital Library (ODL) model and framework, illustrate ODL's applicability by considering a number of case studies, discuss its evaluation with respect to performance and ease of use, and outline future directions.

The Open Archives Initiative

Background and Motivation

The Open Archives Initiative (OAI) was launched in response to a recognised need for low-cost interoperability solutions in the digital library community (Van de Sompel and Lagoze, 2000; Suleman and Fox, 2002a). Besides connecting together systems in distributed digital libraries, the OAI addressed problems arising from collocation of data and services. Acknowledging that the owners of high quality data were not always the best candidates to provide high quality services, the OAI encouraged a multi-layered approach to system development with the data collection distinctly separated from the services provided. To then make the connection between data and services, the OAI developed a Web-based network protocol for simple data transfer – the Protocol for Metadata Harvesting (PMH) (Lagoze, et al., 2002). By just transferring metadata instead of supporting remote searching, this protocol takes the burden off data providers and places it on service providers – making it easier for those who collect to also share their data.

The OAI-PMH enables remote access to collections of metadata, thus enabling the development of interesting services such as the Torii portal (Bertocco, 2001) for resource discovery and annotation, and the Open Citation Project (Hitchcock, et al., 2002) for reference linking of resources. The OAI-PMH is used by many existing popular DL archives to expose their previously opaque collections. In addition, new distributed DL projects – such as the revitalised NCSTRL (Anan, et al., 2002) – use a system model that is based on harvesting metadata from multiple remote sites into one or more central user portals.

Protocol for Metadata Harvesting

The OAI-PMH is a client-server protocol layered over HTTP, using CGI-encoded parameters in requests and XML-encoded data in responses. The aim of the protocol is to support the batch transfer of metadata from a server (data provider) to a client (service provider) using incremental updates whenever a transfer is initiated. This process of obtaining all the (new) metadata from a server, instead of only that which satisfies a search query, is commonly known as harvesting.

The OAI-PMH is made up of 6 requests and associated responses, 3 of which are administrative while the other 3 are for data transfer. These requests, and the semantics of their responses, are as follows:

- Identify – general information about the archive, administrator and policies.
- ListMetadataFormats – a list of all the metadata formats supported by the archive as well their XML namespaces and schema locations.
- ListSets – a list of all the subsections of the archive for selective harvesting.
- ListIdentifiers – a list of identifiers for all records, corresponding to the required metadata format parameter and optional date range and/or set parameters.

- GetRecord – a single record, specified by its unique identifier and metadata format.
- ListRecords – a list of records in the specified metadata format, corresponding to optional date range and/or set parameters.

Open Digital Libraries

Concept

In developing the metadata harvesting protocol, the OAI provided a mechanism to separate data providers from service providers. As part of this process, the OAI established best practices to support their protocol, but which are potentially relevant to digital library design in general. Included among these best practices are the enforcement of identifier uniqueness and the ability to obtain a single record from a source repository based solely on its identifier, metadata format and a network address for the source repository. These are fundamental ideas that were part of Kahn and Wilensky's Repository Access Protocol (Kahn and Wilensky, 1995) and which have now been realised in OAI's broadly-supported DL interoperability protocol.

The Open Digital Library (ODL) project (Suleman and Fox, 2001) has exploited this conceptual framework provided by the OAI protocol in order to form the base for a general-purpose inter-component interaction protocol for digital libraries. Digital libraries have reached a stage in development where they can be specified in terms of standard suites of services. Discussion on architecture and models has consistently noted the need for flexible component models (Gladney, et al., 1994; DELOS, 2001). The OAI protocol fulfils some of this need by providing the mechanism by which mini- and dumb archives can be set up. It is no longer necessary for such archives to provide user-directed services – these services can be delegated to appropriate service providers. Thus, the data providers form the basic components of a digital library, with the sole requirement being to export data. The layers built upon this fall within the ambit of the ODL project. ODL defines popular services as self-contained components and defines interfaces for these components to interact with upstream data providers and peer components, as well as downstream components and elements of user interfaces. As upstream archives support the OAI-PMH, which already contains many desirable elements of digital library design, it was decided to model the inter-component interaction protocols as extensions of the OAI-PMH. Then each component is an extended Open Archive, and the digital library is made up of a network of extended Open Archives, denoted as ODL service components in Figure 1.

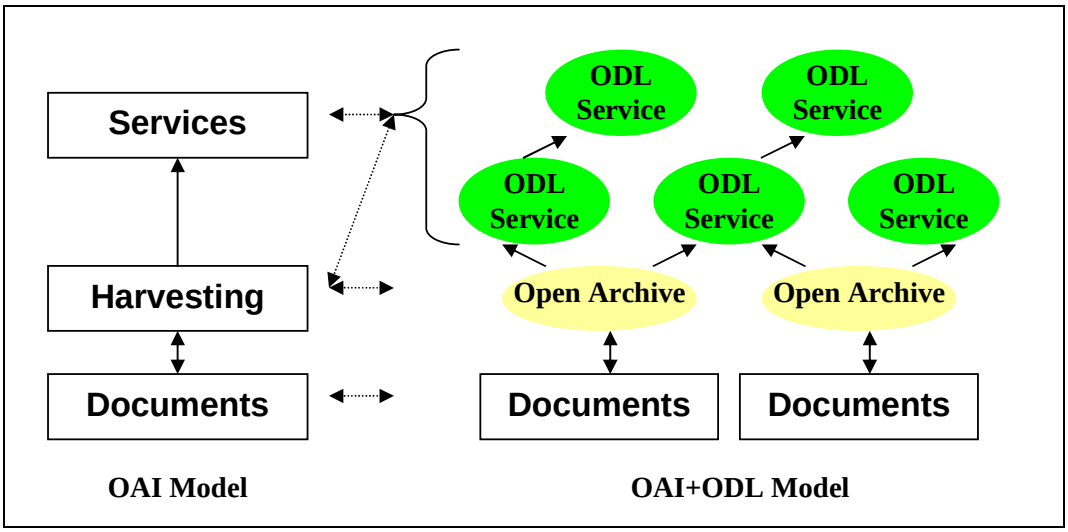

Figure 1. Mapping between OAI model and OAI+ODL component model

Figure 1 illustrates the analogy between the OAI model of interoperability and the ODL component model incorporating both Open Archives Initiative and ODL components. Since any ODL component is fundamentally an Open Archive (albeit with a few extensions), data can be harvested from it by another component. In practice, this means that a search engine can be added to a threaded annotation system as easily as it can be added to a dumb archive. In the above illustration, this capability for inter-component harvesting is indicated by harvesting in the OAI model mapping onto both the Open Archives Initiative and the ODL service components in the OAI+ODL model.

Architecture

An Open Digital Library is thus a network of extended Open Archives, where each of the nodes is a provider of data, services or both. Most of the extensions share common features that were not included in the OAI-PMH because of its focus on metadata harvesting. These common features have been factored out and built into an intermediate layer between OAI and ODL, called the Extended PMH (XPMH). The most important feature of the XPMH is a mechanism to add records to an archive, not necessary for exporting data via the OAI-PMH, but indispensable for inter-component interaction.

Individual ODL protocols for components then were specified as elaborations of the XPMH, by layering specific semantics over the loose frameworks of the underlying protocols. For example, the “set” parameter does not have any specific meaning in the OAI-PMH, so this can be easily used to indicate one more criterion for record selection. In the case of an ODL protocol to support a search engine component, the query is a type of selection (and ranking) criterion and is embedded into the set parameter. Similarly ODL protocols have been specified for other services. These are listed in Table 1 along with the reference ODL components that implement them and a short description of the functionality of each component/protocol.

Table 1. ODL reference components, descriptions and protocols

Name of Component	Description of Functionality	Interface Protocol
DBUnion	To merge together the metadata from multiple sources	ODL-Union
IRDB	Search engine	ODL-Search
DBBrowse	To browse through metadata based on values of particular fields within the metadata	ODL-Browse
WhatsNew	To track and obtain, upon request, a sample of recent entries	ODL-Recent
Box	Dumb archive supporting submit and retrieve operations	ODL-Submit
Thread	Threaded annotation engine for discussion forums, guestbooks and resource annotation	ODL-Annotate
Suggest	Recommender system to make suggestions based on collaborative filtering	ODL-Recommend
DBRate	To manage the submission and access to ratings for individual resources	ODL-Rate
DBReview	Peer review workflow manager geared towards the review of journal and conference publications	ODL-Review

Each of the components listed in Table 1 was implemented and tested by integration into various digital library systems. Some of these are discussed in the case studies that follow.

Case Studies

AmericanSouth.org

Context

The AmericanSouth.org project [1] is a collaborative endeavor to improve access to scholarly resources concerning Southern history and culture. The OAI-PMH had, until recently, been used extensively in the sciences but not in situations where it can support the research and teaching needs of humanities scholars. AmericanSouth.org provided an opportunity to apply the OAI/ODL protocols in a humanities environment. The project provides facilities for scholars to search and retrieve both the content and the context of electronic resources held at multiple institutions and to compare these resources conveniently with other related materials. The central portal prototype was constituted from ODL components (Suleman, 2002), offering value-added services over OAI-compliant metadata harvested from repositories. These source repositories, containing publicly-accessible digital resources on Southern history and culture, were made OAI-compliant using customisable open source software templates and toolkits. A central metadata server set up at Emory University functioned as a union catalogue of selected

digital resources at cooperating libraries. Technical assistance was afforded by Virginia Tech both in building the componentised central system and in helping remote participants to create OAI-compliant repositories from the existing collections.

Service Provider

The prototype AmericanSouth.org central service provider consisted of loosely connected interoperable ODL components offering high level services to the end-users, such as searching, browsing and annotation. Such loose coupling facilitates modular development through componentisation. The DBUnion component was used to harvest metadata from multiple sources and coalesce it into a single data archive. The IRDB component was employed as a search engine on the OAI-accessible data archive. Use of the DBBrowse ODL component was explored in order to allow the user to scan through the entire contents of a system, sorted and filtered on the basis of configurable fields such as title and author institution. Many questions about protocol syntax and component logic were raised and answered during this prototyping phase. The use of ODL components in a production-quality environment helped to isolate and resolve bugs. Feedback obtained from the deployment of the components suggested the need for more documentation and pre-configured networks of assembled components to avoid configuration of individual components for common scenarios.

The project currently uses the ARC search engine (Liu, et al., 2001), which has better performance than the reference implementation ODL component.

Data Providers

Libraries, as data providers and repositories of collections, provided the content and corresponding metadata for AmericanSouth.org. To keep the project at a manageable size for this initial implementation phase, nine partner libraries were identified for participation in AmericanSouth.org. An assortment of tools were built and customised to transform these libraries into OAI repositories. The metadata for the collections exposed by the collaborating member DLs was in a myriad of formats, including DC (DCMI, 1997), METS (Library of Congress, 2003), TEI (Burnard, 2000), VRA Core (VRA Data Standards Committee, 2002) and MARC (Furrie, 2000). The software toolkits selected to implement data providers for each of these collections was chosen based on the nature of the metadata in the collection.

Components used in Data Providers

The following components were used by AmericanSouth.org in order to implement data providers on the collections hosted by collaborating libraries.

OAI-PMH2 XMLFile File-based Data Provider

This is a standalone software package that exposes a set of XML files containing the metadata as an OAI repository [2]. Ease of use is high and it is an ideal tool for novices as it abstracts away most implementation aspects of the protocol. AmericanSouth.org uses multiple instantiations of the XMLFile component for implementing data providers at the participating DLs, making it the most widely configured data provider template.

This package was used in conjunction with a MARC conversion tool, which generated records in the oai_dc format (Johnston, 2002), in order to support the creation of data providers at Auburn University, University of Georgia and University of North Carolina-Chapel Hill.

OAI-PMH2 Perl Implementation

This toolkit implements the skeleton of the OAI-PMH v2.0 in an object-oriented fashion, thus hiding the details of the protocol from code that is derived from the predefined class [3]. The programmer needs to edit routines in a single module to access the data source/database. Support for multiple metadata formats is provided. Version 1.1 of this Perl template was combined with a tailor-made Web page parser, to create Dublin Core surrogate records from Web pages, in order to expose the metadata of University of Georgia's Russell collection.

OAICAT

OAICAT is a Java Servlet Web application developed by OCLC, providing an OAI-PMH v2.0 repository framework [4]. The framework can be customised to work with arbitrary data repositories by implementing some Java interfaces. All the implementation aspects of the data provider that deal with the database are captured in a properties file. Support for metadata formats is provided by coding crosswalks for the desired metadata formats. Virginia Tech's Imagebase collection, consisting of 70000 images with metadata in the VRA format stored in a mySQL relational database, uses OAICAT to implement a data provider.

Lessons Learnt

AmericanSouth.org provided a platform to apply and test the concepts of OAI and ODL, for serving a humanities audience. This has given rise to an open forum for scholars to distribute and review their work, building a knowledge base around a distributed, online repository of documents. The central benefits of the demonstration phase of AmericanSouth.org were the harvesting and integrating of metadata using OAI-PMH and the application of ODL for creation of a componentised DL that benefits libraries, scholars and students directly, and provides a foundation for the development of future services.

Computing and Information Technology Interactive Digital Educational Library (CITIDEL)

Context

The Computing and Information Technology Interactive Digital Educational Library (CITIDEL) [5] aims to be a comprehensive, pedagogically-focused digital library in Computer Science and Information Technology. CITIDEL is being built by a consortium of institutions: Virginia Tech, Hofstra University, Pennsylvania State University, Villanova University and The College of New Jersey, and is one of the largest components of the US NSF's National Science Digital Library (NSDL).

The implementation of CITIDEL began in 2001, when ODL was still taking shape both conceptually and in terms of availability of components. As such, CITIDEL does not represent a from-scratch case of building a digital library from ODL components, but instead is a project which is evolving towards this architecture.

In addition, the exploratory nature of the CITIDEL project has contributed to the fact that much of the current system does not utilise ODL. However, ODL representations are being designed and developed for many parts of CITIDEL, and vice versa.

Current OAI and ODL use in CITIDEL

Currently CITIDEL makes use of the following ODL and OAI components:

- IRDB. IRDB is used as the basic search service provider for CITIDEL. IRDB is a general-purpose search engine, with support for indexing and querying structured documents (XML), as well as standard modifiers such as force (+) and forbid (-) in queries.
- OAI Harvester. CITIDEL has made use of the Perl OAI harvester template to implement its harvester subsystem. This allows for a standard interface to acquiring records from most of CITIDEL's member collections. Where some member collections do not yet use OAI-PMH, support is being provided to assist.
- OAI data provider template. CITIDEL itself is an Open Archives Initiative data provider, using the Perl data provider template. First and foremost this allows for integration with NSDL, but CITIDEL also can be a valuable source for other digital libraries through its OAI-PMH interface.
- OAI XMLfile data provider. As part of the CITIDEL effort, the CS History collection was turned into an Open Archive so that it can more easily contribute to the main metadata repository. The most convenient way to do this was found to be turning the source set of XML files for the collection almost directly into a data provider. This was tractable because of the relatively small size of the collection, and in fact was the original impetus for developing the XMLfile component.
- ODL-Thread component. Threaded annotations for resources in CITIDEL were part of the design from a very early stage. This is an important feature for value-added digital library services, re-use of resources, and improved pedagogy. ODL-Thread provides an abstract system for attaching discussion areas to any CITIDEL resource that is currently a part of the system. The latest annotations posted through this system can be accessed through the CITIDEL home page.

Planned Integration of Existing ODL Components

There are plans to eventually integrate the following existing ODL components into the CITIDEL system:

- The DBUnion union catalog component. CITIDEL currently rests atop a union catalog, which contains all metadata harvested from all of its member collections.

Currently this is custom code. However, in the future this could be implemented using the standard union catalog component with minimal modifications.

- ODL-DBBrowse. Applying this component would expand CITIDEL's browsing capabilities, including generic sorting and filtering controls based on various metadata fields.
- MDEdit. MDEdit is a schema-driven forms-based XML metadata editor used to support the creation of user interfaces for ODL components (as discussed next in the context of the DL-in-a-box project). There are a number of places in the CITIDEL user interface where this can be deployed, for example, for facilitating user-supplied translations of resource metadata into various languages.

Digital Library in a Box (DL-in-a-box)

Context

This project is a collaborative venture between the University of Florida, University of Illinois at Urbana-Champaign and Virginia Tech, with the aim of supporting new NSDL projects which do not have experience in building digital libraries. By providing them with basic software building blocks, it would greatly reduce the cost of building new digital library software (Fox, et al., 2002).

The NDLTD member profile management system

At Virginia Tech ODL components were used to help build an NDLTD (Networked Digital Library of Theses and Dissertations) member profile management system. There are 4 components in this system, as shown in Figure 2.

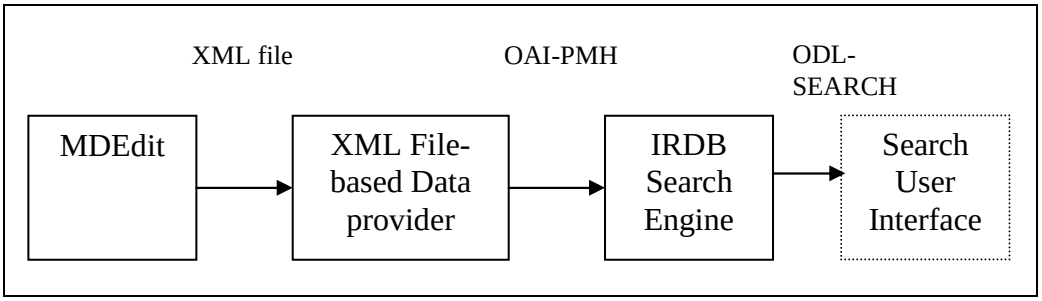

Figure 2. NDLTD member profile management system

1. The MDEdit module provides a convenient tool for entering data. MDEdit is a metadata editor driven by XML Schema descriptions. It supports drop-down menus, checking of mandatory fields and multiplicity of fields at all levels of embedding within the metadata. A new member application page was constructed using MDEdit. After the administrator approves an application, an XML file is created to hold the member's profile information.
2. With XML files as the primary data source, the XML file-based data provider module is used to establish an OAI data provider. Any OAI harvester on the Internet can

harvest data from this data provider. This fact represents another aspect of openness for the ODL/OAI approach to building systems.

3. Then the IRDB search engine is used to provide a service to search among the profiles. IRDB contains a harvester that harvests data from the above mentioned data provider using the standard OAI-PMH protocol.
4. The design philosophy of ODL is to separate the service component logic from the user interface. Thus, the user interface is not technically part of the ODL framework. Any interface conforming to the ODL-Search protocol should be able to act as a user interface. To simplify that process, current experiments are investigating the use of XSLT to componentise the user interface aspects of building digital libraries.

The steps listed above, as indicated in Figure 2, were followed to create a simple digital library, which is both extensible and flexible. Browsing and rating components can easily be added if there is a demand for such services. The components also make testing easier - if there is doubt as to the correctness of any component, it can be replaced by a newly-installed version or a different component that adheres to the same ODL protocol. Thus, any errors in the individual components do not have a catastrophic effect on the whole system.

Documentation

The DL-in-a-box project also maintains a repository of ODL components and a collection of publications and tutorials to assist new adopters in understanding and using the technology for their own projects [6].

Other Systems

The case studies presented illustrate how the OAI and ODL philosophies have been applied to different problem areas, with differing requirements and user communities. In addition to these, the following systems also were assembled during the development of the ODL component suite:

The NDLTD Union Catalog (Suleman, et al., 2001) is a central portal for students and researchers to locate electronic theses and dissertations. The system is built in a layered fashion, with metadata collection separated from resource discovery. For the former aspect, metadata is collected from 14 remote sites (as of March 2003) using a combination of ODL and OAI components, and then exposed as a single data provider known as the Union Archive (Suleman and Fox, 2002b). A simple service provider built exclusively of ODL components then supports rudimentary access to the resources. In addition, VTLS has built a portal for resource discovery based on metadata harvested from the Union Archive. In this project, the use of ODL and OAI components allowed a factoring of common functionality from separate systems, thus improving data coherence and reducing the network bandwidth used during harvesting from internationally-distributed archives.

The Computer Science Teaching Center (CSTC) [7] is a digital library of peer reviewed teaching resources in computer-related disciplines. ODL components were used to replace part of the production system to support additional functionality (viz., a more

advanced browse function based on the DBBrowse component). Further experiments then were conducted to model the services with a network of ODL components. This system was used for testing of the framework for building complete systems and can, in the future, replace the existing CSTC system or function as a standalone system for similar archives. Unlike the existing system, an ODL-based CSTC system lends itself more easily to extension and replacement of individual components.

Evaluation

While the ODL framework was designed to promote better software engineering practices in building digital libraries, this cannot be at the expense of performance. To test for this, experiments were conducted (Suleman, 2002) to assess the impact of the componentised architecture on the overall system performance. These tests have illustrated that:

- execution time is affected to a larger degree by the spawning of server-side processes than by additional processing for the ODL/OAI layer and indirection through the Web server,
- using persistent Web applications - e.g., SpeedyCGI [8], Java servlets (Zeiger, 1999) - greatly reduces the effect of layering and indirection, and
- user interactions that require internal requests to multiple components can be accomplished in a reasonable amount of time (<1 sec for up to 9 internal requests in tests).

To test the simplicity and understandability of the framework, a controlled user experiment was conducted with 54 computer science graduate students. The subjects first were given an introductory lecture on OAI and ODL. This was followed by an exercise to build a simple digital library using components. Detailed instructions were provided and the students' ability to understand the process was assessed. Their responses supported the hypothesis that the ODL/OAI approach to building DLs was easily understood and that installation, configuration and composition of components was a reasonably simple and understandable process.

Between this experiment and the many case studies performed, the ODL framework is being assessed on an ongoing basis for the ease with which it can be applied to a project. All performance measurements are aimed at measuring inter-component interactions instead of absolute component performance. This approach was adopted to ensure that ODL could be assessed in terms of its model rather than the real-world performance of reference implementations of components.

Future Work

There is a wealth of possible ODL components which can be developed as part of the various case studies presented, in particular the CITIDEL and DL-in-a-box projects. Componentising and standardising these services would allow others to easily implement them, and allow the digital library community as a whole to benefit from work already done. The following is a brief introduction to possible future research avenues.

ODL-XMLLog

Currently in alpha stages, investigations have been initiated into the componentisation of XML Logging for Digital Libraries, using a standard developed at Virginia Tech (Gonçalves, et al., 2002). This component will allow logging of higher-level information than traditional Web server logs using the common log format (Luotonen, 1995), and enables detailed logging about services and users, including click stream analysis. The output of this could in turn be used by other digital library services, for example query refinement or bug reporting.

Profile Filtering

CITIDEL accommodates multiple user sub-communities by means of filtering profiles that tell the filtering system what view of the content should be provided to specific users. This system could be generalised to any result set which can be produced in the digital library by an ODL component, and could therefore be embodied in an ODL component of its own.

Multi-classification browsing

CITIDEL provides a subject-hierarchical browsing interface, which is built upon categorisation metadata of the DL resources, and is invariant with respect to which supported scheme is being used. This means that users can browse the same set of content organised by schemes like ACM's Computing Classification System (ACM, 1998), Computing Curricula 2001 (Joint Task Force on Computing Curricula, 2001) and others, according to their preferences. Resources need only be classified under one scheme, with the invariant behaviour provided by inter-scheme mappings. As a side effect, this system removes almost all of the work in migrating between subsequent revisions of classification schemes. This robust and useful functionality can be abstracted into a component and deployed in many other settings where multiple alternate classification schemes exist or multiple revisions of schemes are used.

Active Lists component

CITIDEL currently has an Active Lists service, which evolved from the Instructional Architect software (Dorward, et al., 2002). This service allows sequencing and application of views to lists of resources gleaned from CITIDEL, and re-publishing of the results back to CITIDEL. The modular nature of this service with respect to the core functionality of CITIDEL makes it a prime candidate for ODL componentisation.

Instructional Activities component

CITIDEL also has an instructional activities service provided through the VIADUCT subsystem. This service facilitates the development of lesson plans using resources from CITIDEL as attachments of various sorts (required reading, background, etc.). Like Active Lists, this service will fit well into the ODL paradigm.

High-performance Search Component

A search engine is currently being developed in C++ with an emphasis on performance and scalability for large collections. Special features include the ability to run entirely in-memory, which is becoming more useful due to the falling cost of RAM. An ODL interface is being built for this search component which will essentially allow it to be used as a drop-in replacement for the prototype IRDB component.

User Interface support

ODL is based on HTTP requests and responses, but system designers need quick and easy user interface setup mechanisms. One approach to address this is to integrate the components with existing portal software, such as uPortal (JASIG, 2002), to provide an integrated interface for users and a complete solution for developers.

Impact and Conclusions

Digital libraries are gradually moving to deployment beyond the experimental arena. As more systems are produced, design patterns are emerging and the OAI and ODL projects exploit these patterns to make system building a repeatable process. Other projects in the sphere of digital library architecture and interoperability, such as the OpenDLib project (Castelli and Pagano, 2002), have shown similar results and strengthen the case for building all systems as networks of components instead of as software monoliths.

The ODL/OAI component approach has important implications for Web Services technology, such as SOAP (Box, et al., 2000), and vice versa. The anticipated ubiquity of Web Services provides a natural evolutionary path for component models and this inevitability has already influenced the development of the current version of the OAI protocol. At the same time, the ODL experiments have shown how Web-based services can be used to effectively and efficiently create distributed systems to meet the needs of a particular community, thus vindicating the Web Services approach.

It is important, however, to note that ODL is an experimental framework to test the simplicity of creating and using components for a controlled set of digital library use cases, as discussed in this article. It is hoped that once such an approach is widely accepted, a standardisation process will be initiated to develop a production-quality framework, which will subsequently lead to production quality components being made available for integration into new and existing systems.

Systems built from components also have far-reaching consequences for encouraging the establishment of digital libraries. Traditionally, any such endeavour required installation and configuration of an off-the-shelf package or custom software development. The former solution would not always fit the requirements of the problem while the latter solution is time-consuming and costly. Component architectures make it possible for a bevy of components to be assembled in a multitude of configurations, each supporting further customisation as needed. Ideally, there no longer would be a need for specialised and complex software catering to the needs of particular communities or use cases. Ultimately, the suite of available components can be used as a basis on which to build more interesting and advanced services to better meet the needs of users.

Acknowledgements

Thanks are given for the support of NSF through its grants CCR-9627922, IIS-0002935, DUE-0121679 and DUE-0136690. Thanks also are given to the AmericanSouth.org project funded by the Mellon Foundation.

Websites

[1] AmericanSouth.org. http://www.americansouth.org/

[2] OAI-PMH2 XMLFile File-based Data Provider. http://www.dlib.vt.edu/projects/OAI/software/xmlfile/xmlfile.html

[3] VTOAI OAI-PMH2 Perl Implementation. http://www.dlib.vt.edu/projects/OAI/software/vtoai/vtoai.html

[4] OAICat. http://www.oclc.org/research/software/oai/cat.shtm

[5] Computing and Information Technology Interactive Digital Educational Library (CITIDEL). http://www.citidel.org/

[6] Digital Library-in-a-Box. http://dlbox.nudl.org/

[7] Computer Science Teaching Center. http://www.cstc.org/

[8] SpeedyCGI. http://daemoninc.com/speedycgi/

References

ACM (1998), The ACM Computing Classification System (1998), ACM. Available http://www.acm.org/class/1998/ccs98.html

Anan, H., Liu, X., Maly, K., Nelson, M. L., Zubair, M., French, J. C., Fox, E. A. and Shivakumar, P. (2002), "Preservation and transition of NCSTRL using an OAI-based architecture", in Proceedings of Second ACM/IEEE-CS Joint Conference on Digital Libraries, Portland, OR, USA, 14-18 July 2002, pp. 181-182.

Bertocco, S. (2001), "Torii, an Open Portal over Open Archives", in High Energy Physics Libraries Webzine, Issue 4, June 2001. Available http://library.cern.ch/HEPLW/4/papers/4/

Box, D., Ehnebuske, D., Kakivaya, G., Layman, A., Mendelsohn, N., Nielsen, H. F., Thatte, S. and Winer, D. (2000), Simple Object Access Protocol (SOAP) v1.1, W3C, 8 May 2000. Available http://www.w3.org/TR/SOAP/

Burnard L. (2000), "Text Encoding for Interchange: A New Consortium", in Ariadne, Issue 24. Available http://www.ariadne.ac.uk/issue24/tei/

Castelli, D. and Pagano, P. (2002), "OpenDLib: A Digital Library Service System", in Research and Advanced Technology for Digital Libraries, Proceedings of the 6th European Conference, ECDL 2002, Rome, Italy, September 2002, pp. 292-308.

DCMI (1997), Dublin Core Metadata Element Set Version 1.1: Reference Description. Available http://www.dublincore.org/documents/dces/

DELOS (2001), Digital Libraries: Future Directions for a European Research Programme, San Cassiano, Alta Badia, Italy, 13-15 June 2001. Available http://delos-noe.iei.pi.cnr.it/activities/researchforum/Brainstorming/brainstorming-report.pdf

Dorward, J., Reinke, D. and Recker, M. (2002), "An evaluation model for a digital library services tool", in Proceedings of Second ACM/IEEE-CS Joint Conference on Digital Libraries, Portland, OR, USA, 14-18 July 2002, pp. 322-323.

Fox, E. A. (ed.) (1993), Sourcebook on Digital Libraries: Report for the National Science Foundation, TR-93-35, VPI&SU Computer Science Dept., Dec. 1993, Blacksburg, VA, 400+ pages. Available ftp://fox.cs.vt.edu/pub/DigitalLibrary/

Fox, E. A. and Sornil, O. (1999) "Digital Libraries", in Modern Information Retrieval, eds. Ricardo Baeza-Yates and Berthier Ribeiro-Neto, Ch. 15, ACM Press / Addison-Wesley-Longman England, pp. 415-432.

Fox, E. A., Suleman, H. and Luo, M. (2002), "Building Digital Libraries Made Easy: Toward Open Digital Libraries", in Proceedings of the 5th International Conference of Asian Digital Libraries, 11-14 December 2002, Singapore.

Fox, E. A. and Urs, S. R. (2002), "Digital Libraries", in Annual Review of Information Science and Technology, ed. Blaise Cronin, Vol. 36, Ch. 12, pp. 503-589.

Furrie, B. (2000), Understanding MARC Bibliographic: Machine-Readable Cataloging, Cataloguing Distribution Service, Library of Congress. Available http://lcweb.loc.gov/marc/umb/

Gonçalves, M. A., Luo, M., Shen, R., Ali, M. F. and Fox, E. A. (2002), "An XML Log Standard and Tool for Digital Library Logging Analysis", in Research and Advanced Technology for Digital Libraries, Proceedings of the 6th European Conference, ECDL 2002, Rome, Italy, September 2002, pp. 129-143.

Gladney, H., Ahmed, Z., Ashany, R., Belkin, N. J., Fox, E. A. and Zemankova, M. (1994), "Digital Library: Gross Structure and Requirements", Workshop on On-line Access to Digital Libraries, June 1994.

Hitchcock, S., Bergmark, D., Brody, T., Gutteridge, C., Carr, L., Hall, W., Lagoze, C. and Harnad, S. (2002), "Open Citation Linking: The Way Forward", in D-Lib Magazine, Vol. 8, No. 10, October 2002. Available http://www.dlib.org/dlib/october02/hitchcock/10hitchcock.html

JASIG (2002), uPortal 2.0 Architecture Overview. Available http://mis105.mis.udel.edu/ja-sig/uportal/architecture/uPortal_architecture_overview.pdf

Johnston, P. (2002), XML Schema for oai_dc. Available http://www.openarchives.org/OAI/2.0/oai_dc.xsd

Joint Task Force on Computing Curricula (2001), "Computing Curricula 2001", in Journal of Educational Resources in Computing, Vol. 1, Issue 3es, September 2001.

Kahn, R. and Wilensky, R. (1995), A Framework for Distributed Digital Object Services. Available http://www.cnri.reston.va.us/k-w.html.

Lagoze, C., Van de Sompel, H., Nelson, M. L. and Warner, S. (2002), The Open Archives Initiative Protocol for Metadata Harvesting – Version 2.0, Open Archives Initiative, June 2002. Available http://www.openarchives.org/OAI/2.0/openarchivesprotocol.htm

Liu X., Maly, K., Zubair, M. and Nelson, M. L. (2001), "Arc - An OAI Service Provider for Digital Library Federation", in D-Lib Magazine, Vol. 7, No. 4, April 2001. Available http://www.dlib.org/dlib/april01/liu/04liu.html

Library of Congress (2003). METS: An Overview and Tutorial. Available http://www.loc.gov/standards/mets/METSOverview.html

Luotonen, A. (1995), The Common Logfile Format. Available http://www.w3.org/pub/WWW/Daemon/User/Config/Logging.html

Suleman, H. (2002), Open Digital Libraries, Virginia Tech Department of Computer Science, Blacksburg, Ph.D. Disseration, 2002. Available http://scholar.lib.vt.edu/theses/available/etd-11222002-55624/

Suleman, H. and Fox, E. A. (2001), "A Framework for Building Open Digital Libraries", in D-Lib Magazine, Vol. 7, No. 12, December 2001. Available http://www.dlib.org/dlib/december01/suleman/12suleman.html

Suleman, H. and Fox, E. A. (2002), "The Open Archives Initiative: Realizing Simple and Effective Digital Library Interoperability", in special issue on "Libraries and Electronic Resources: New Partnerships, New Practices, New Perspectives" of Journal of Library Automation, Vol. 35, No. 1/2, pp. 122-145.

Suleman, H. and Fox, E. A. (2002), "Towards Universal Accessibility of ETDs: Building the NDLTD Union Archive", presented at The Fifth International Symposium on Electronic Theses and Dissertations (ETD 2002), Provo, Utah, USA, 30 May – 1 June 2002. Available http://www.wvu.edu/~thesis/proceedings.html

Suleman, H., Atkins, A., Gonçalves, M. A., France, R. K., Fox, E. A., Chachra, V., Crowder, M. and Young, J. (2001), "Networked Digital Library of Theses and

Dissertations: Bridging the Gaps for Global Access – Parts 1 and 2", in D-Lib Magazine, Vol. 7, No. 9, September 2001. Available http://www.dlib.org/dlib/september01/suleman/09suleman-pt1.html and http://www.dlib.org/dlib/september01/suleman/09suleman-pt2.html

Van de Sompel, H. and Lagoze, C. (2000), "The Santa Fe Convention of the Open Archives Initiative", in D-Lib Magazine, Vol. 6, No. 2, February 2000. Available http://www.dlib.org/dlib/february00/vandesompel-oai/02vandesompel-oai.html

VRA Data Standards Committee (2002), VRA Core Categories Version 3.0. Available http://www.vraweb.org/vracore3.htm

Zeiger, S. (1999), Servlet Essentials. Available http://www.novocode.com/doc/servlet-essentials/