ARTICLES

D-Lib Magazine
December 2001

Volume 7 Number 12

ISSN 1082-9873

A Framework for Building Open Digital Libraries

Hussein Suleman and Edward A. Fox
Virginia Tech
(hussein, fox)@vt.edu

Abstract

Digital Libraries (DLs) have traditionally been positioned at the intersection of library science, computer science, and networked information systems. The different underlying philosophies of these three fields has had an unsettling influence on the development of DLs. While library science is fairly mature, networked information systems are constantly evolving to keep pace with Internet innovation. DLs are thus expected to demonstrate the careful management of libraries while supporting standards that evolve at an astonishing pace. This architectural moving target is a predicament that all DLs face sooner or later in their lifecycle, and one that few manage to deal with effectively. To exacerbate this problem, there has been a general desire for systems to be interoperable at the levels of data exchange and service collaboration. Such interoperability requirements necessitated the development of standards such as the Dublin Core Metadata Element Set and the Open Archives Initiative's Protocol for Metadata Harvesting (OAI-PMH). These standards have achieved a degree of success in the DL community largely because of their generality and simplicity. Informed by those lessons, this project is an attempt to consistently extend known interoperability standards to form the basis of a framework of components for building extensible DLs.

Preamble

"Open" is a word that conjures up many different connotations depending on the context in which it is used. In this case its use is deemed appropriate since Open Digital Libraries (ODLs) build directly upon the concepts and philosophies of the Open Archives Initiative [OAI, 2001]. Just as Open Archives are data repositories that allow remote access using a simple and well-defined publicly available protocol, so too will ODLs accomplish the same in the context of service components. Extension of standards, such as the OAI-PMH [Lagoze and Van de Sompel, 2001], is another contentious issue since it invariably adds undesirable complexity. This work is based on the premise that if a new standard is needed, it is better derived from an existing and accepted one as long as the two are completely separable.

Introduction

Digital libraries are far from well-defined [Borgman, 1999] Definitional agreement may only extend to notions of accessible collections of information. Because of this, it is hardly surprising to note that the field does not easily converge on standards and technology. Most of the existing systems that are

classified as DLs have resulted from custom-built software development projects -- each the product of intensive design, implementation and testing cycles. There are many reasons why this effort is repeated for each project:

- Many DLs are built in isolation as a response to the needs of a particular community, in most cases not involving personnel with prior experience.
- Most modern DLs have WWW interfaces -- thus the user interfaces and process flows are fashioned to resemble the way people use the WWW, which itself changes with time.
- Each DL is aimed at meeting the needs of a particular community -- so the underlying program logic varies vastly among systems.
- Most DLs are intended to be quick solutions to urgent community needs -- so not much thought goes into planning for future redeployment of the systems.
- DLs, by the very nature of being responses to user needs, can be arbitrarily complex, so new projects sometimes choose to develop from scratch because it is cheaper than adapting what already exists to a different set of scenarios.
- As DL systems get more complex, extensibility becomes more difficult and, as a result, maintainability is compromised. As testimony to this, at the turn of the millenium, Dijkstra wrote that computing's central challenge of "how not to make a mess of it" had not been met [Dijkstra, 2001].
- There are very few software toolkits available to build DLs.

A natural solution would be to create software toolkits. A few institutions have investigated this approach. Dienst [Lagoze and Davis, 1995] is a DL system developed at Cornell University with tasks clearly divided and specified by a protocol based on HTTP and eventually using XML. It was developed to support distributed operation of the NCSTRL project and, while technically sound, required an investment in software, methodology, and support that some prospective users were not willing to make. The Repository-in-a-Box [NHSE, 2001] software from the University of Tennessee is an alternative, as is the E-Prints software from Southampton University [OpCit, 2001]. Both these toolkits avoid many problems related to complexity of DLs by defining workflows that are not easy to change. All of these and other systems have had varying degrees of success among archivists looking for drop-in solutions but they generally suffer from two basic problems:

- The range of possible workflows is restricted by the design of the system.
- The software is either built as a monolithic system or as components that communicate using non-standard protocols -- in both cases making understanding and modification a complex process.

Because it is widely accepted as good software engineering practice, most modern programming environments adopt some form of component model. Even in the DL community, as far back as 1994, early discussions on the future of DLs [Gladney, et al., 1994 2001] concluded that components were an integral part of the solution. The University of Michigan Digital Library Project investigated using autonomous agents as the basic components of a DL [Birmingham, 1995]. Stanford's InfoBus project defined a set of services to support distributed digital libraries, each wrapped in an object, communicating through a remote method invocation interface [Baldonado, et al., 1997; Roscheisen, et al., 1998]. Other scientific communities embraced component technology as an aid to rapidly and

correctly solving problems -- for example, the Sieve framework at Virginia Tech [Sieve, 2001] encapsulates scientific functionality into software components. However, in spite of the widespread use of such technology, for the reasons outlined above, the DL community did not in general adopt a single component framework.

In October of 1999 the Open Archives Initiative (OAI) [Van de Sompel and Lagoze, 2000] was launched in an attempt to address interoperability issues among the many existing and independent DLs. The focus was on high-level communication among systems and simplicity of protocols. The OAI has since received much media attention in the DL community and, primarily because of the simplicity of its standards, has attracted many early adopters.

The OAI Protocol for Metadata Harvesting [Lagoze and Van de Sompel, 2001] in essence supports a system of interconnected components, where each component is a DL. Also, since the protocol is simple and is becoming widely accepted, it is far from being a custom solution of a single project. The OAI protocol can be thought of as the glue that binds together components of a larger DL. However, since DLs are themselves defined only loosely, this collaborative system could be composed of individual component DLs, each with different functionality. In the extreme case, each component DL could supply the functionality of exactly one (part of a) service expected by a user. This is the approach taken in this work, where *Digital Libraries are modeled as networks of extended Open Archives, with each extended Open Archive being a source of data and/or a provider of services*. (The "extensions" are necessary since Open Archives are optimized for the provision of data -- but are generalizable to other tasks with a few minor changes.) This network of extended Open Archives, an instance of which is illustrated in Figure 1, is herein called an *Open Digital Library*(*ODL*).

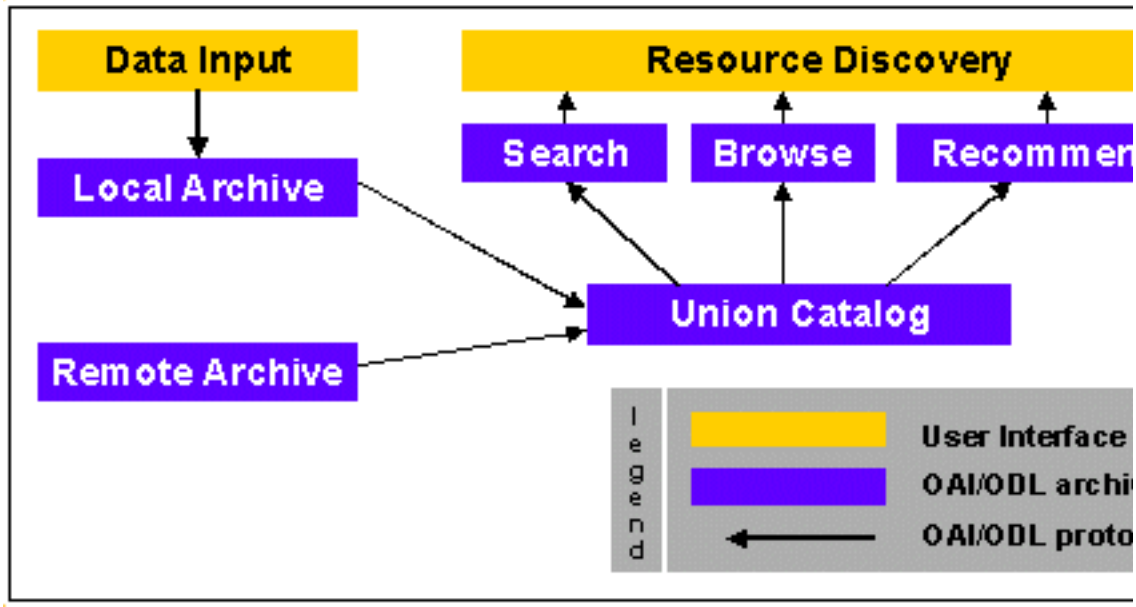

Figure 1. Example networked architecture of an Open Digital Library.

This approach to DL architecture is further motivated by the following factors:

- Componentization and standardization are built into the system by design if every service is delivered by an extended Open Archive. This inherently supports reuse and allows for interoperability at the level of individual services within the DL.

- The ODL approach closely resembles the way that physical libraries work. In a physical library the individual systems interoperate within their own communities. For example, the purchasing department interoperates with the booksellers and the inter-library loan department interoperates with peer departments at other libraries. A retiring head of the acquisitions department can be replaced by a peer at another institution because he or she understands a common protocol for all libraries. Interoperability is achieved at the level of individual services rather than at the level of organizations.

- There is currently a significant difference in technology between a research DL and a production DL. The former focuses on experimental concepts and technology while the latter deals with the real issues of meeting the needs of users. Connecting the two is not usually a simple task, but if both systems subscribe to a common protocol, that would greatly simplify matters -- OAI can be the basis for that protocol.

- The Internet is without a doubt the single most effective information dissemination tool of current times. This was primarily possible because of the simplicity of the protocols it relies on and the hierarchical manner in which protocols such as HTTP [Fielding, et al., 1999] build on more fundamental protocols such as IP and TCP. The OAI provides us with a simple protocol to transfer metadata; building simple layered extensions to this protocol would closely follow the proven methodology of the networking community [ISO, 1994].

- While complex system interactions might support complex operations, they also raise the bar on adoption of new technology. A good example would be the hypertext community where the WWW has succeeded well beyond other projects simply because its model was always a simple one [Berners-Lee and Fischetti, 1999]. Modeling DL services as Open Archives would enforce such a degree of simplicity.

- Scholarly communication is a rapidly changing field and many people are slow in making the transition to new forms of communication in spite of a growing number of advocates [Harnad, 1999]. The success of new DL systems in this arena relies on keeping pace with current thinking on how publications are created, processed, and distributed. A simple component model will greatly simplify changes in the workflow of the DL to support the gradual shift to new and improved processes.

- User interface design and workflow management are complex tasks. But common base-level services -- mediators for connecting resources or middleware in three-tier client-server development [Umar, 1997] -- have emerged in practice, for example, supporting searching and browsing. If an arbitrarily complex user interface could access DL components in a standard manner, it would be easier to interchange components and add new services -- the OAI protocol could be the basis of that standard protocol for components.

- Norman advocates that designs should be visible, understandable and natural in their mappings [Norman, 1990]. The OAI protocol is already establishing itself in those areas so it makes an ideal foundation upon which to build.

Open Digital Library Design

ODLs are guided by a set of design principles and operationalized with the aid of a set of OAI-PMH extensions. By analyzing some of the emerging aspects of Internet development, a set of basic design principles have been extracted to guide the construction of new protocols so that they are consistent with proven techniques in networked information systems. The observed factors that influence these include: simplicity of protocols, openness of standards, layering of semantics, independence of components, loose coupling of systems, purposeful orthogonality, and reuse wherever possible. Based on these ideas, a

generalization of the OAI protocol is possible so that it may be used for purposes that go beyond its original intention, namely to provide higher-level DL services. Formally, these principles are stated as follows:

1. All DL services should be encapsulated within components that are extensions of Open Archives.

2. All access to the DL services should be through their extended OAI interfaces.

3. The semantics of the OAI protocol should be extended or overloaded as allowed by the OAI protocol, but without contradicting the essential meaning.

4. All DL services should get access to other data sources using the extended OAI protocol.

5. Digital Libraries should be constructed as networks of extended Open Archives.

Each DL service is then designed as a self-contained component that communicates with other services using a protocol that is an extension of the OAI-PMH. A typical example of how this works in practice is illustrated in Figure 2, which shows a search engine's relationships to its data source and the interface that uses it as a service component.

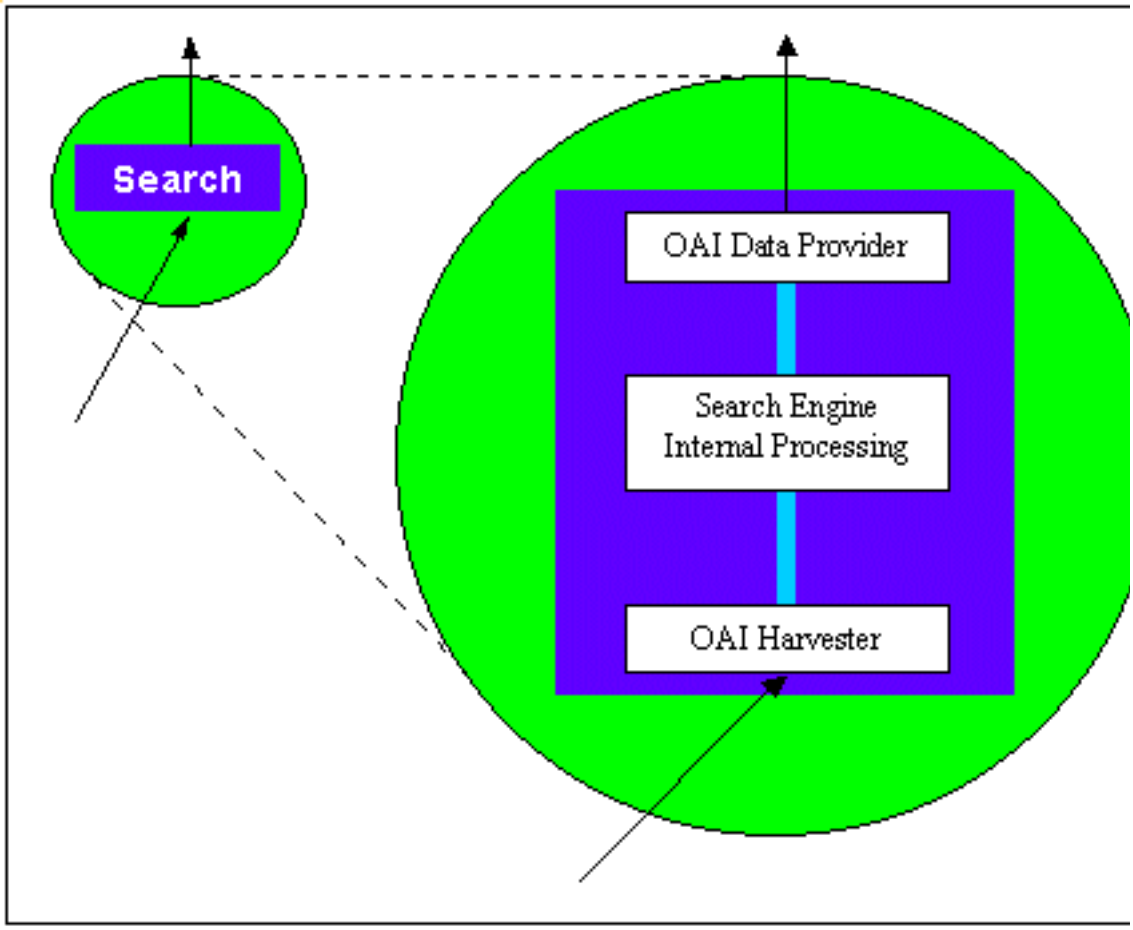

Figure 2. Internal structure of a typical ODL search component.

In this case, the OAI Harvester is used to obtain a stream of data which in turn is used to create indices for searching. Queries are then submitted through the OAI Data Provider interface. These queries overload the semantics of the OAI-PMH, by using the OAI notion of sets to correspond to the dynamically generated result sets of a search engine. Analogously, when such a request is submitted, the query is mapped to the name of a set. Thus, without making any changes, the OAI-PMH can be used to serve as the interface to a search engine. With a few minor additions to the OAI-PMH, information such as cardinality of result sets also can be returned.

An example of such an OAI request, with overloaded semantics for a search component is:

```
verb=ListIdentifiers&set=odlsearch1/computer%20science/1/10
```

This query specifies that the response should contain the identifiers of the first 10 most relevant documents with respect to the query "computer science". The response generated by the component is in the standard format returned by OAI-compliant archives.

To use this component in a typical ODL network, it must be connected to a source of metadata and a user interface, much in the same manner as UNIX pipes and filters are used to connect cooperating processes together. This is illustrated in Figure 3.

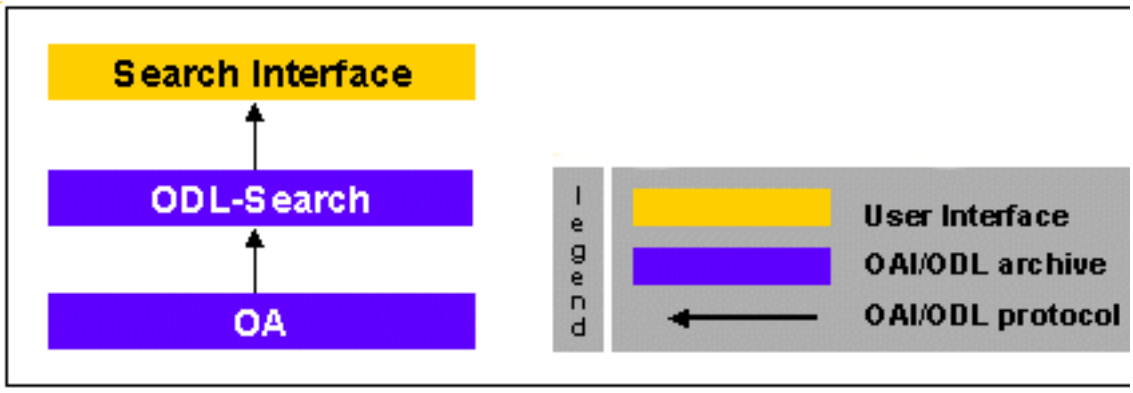

Figure 3. Simple ODL network using Search component.

Implementation and Analysis

To test the feasibility of ODL design, a suite of components was specified, implemented, integrated into a network, and assessed for their ability to replace existing systems. These components and their functionality are specified in Table 1.

Component	Function
ODL-Union	Combine metadata from multiple sources
ODL-Filter	Reformat metadata from (non-OAI-Conforming) data sources
ODL-Search	Provide search engine functionality

ODL-Browse	Provide category-driven browsing functionality
ODL-Recent	Provide a sample of recently-added items

Table 1. Components used in prototype systems

The components were implemented individually, tested using the Repository Explorer [Suleman, 2001], and combined into an ODL network to provide DL services over the union archive of metadata maintained by the Networked Digital Library of Theses and Dissertations (NDLTD) [Suleman, et al., 2001]. Then, to illustrate reusability, some components were integrated into the production server for the Computer Science Teaching Center [CSTC, 2001]. A system architecture for the former of these systems is depicted in Figure 4.

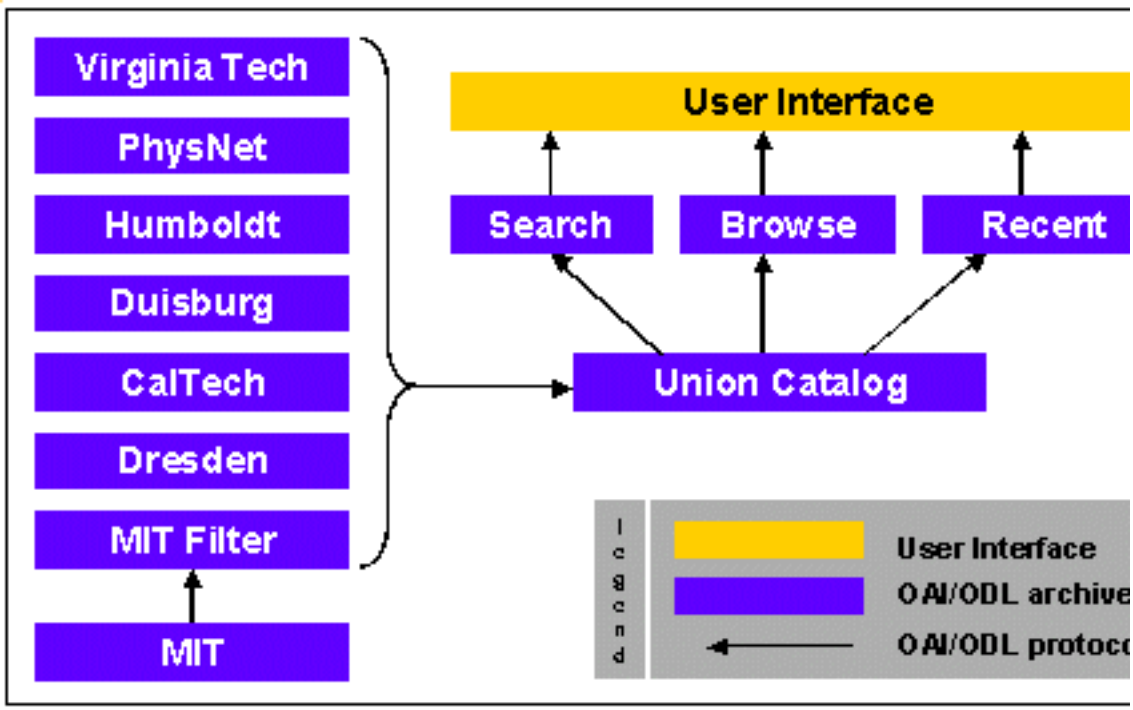

Figure 4. Architecture of NDLTD ODL system.

There were seven source archives from which data was harvested through OAI-PMH interfaces, one of which required a filter because of minor differences in implementation from the others. The data was aggregated into a central archive for use by local services. Three high-level services were provided using this data: Search, Browse, and Recent. Search indexed the data and exposed an OAI-like interface for specifying keyword queries. Browse sorted the data and exposed a slightly different OAI-like interface for accessing items by controlled vocabulary elements. Recent stored recent items and upon request returned a random sample of those.

The user interfaces were created using minimal scripts to translate HTML form elements into ODL requests and then convert the XML responses into human-readable displays using XSL stylesheets. The user interface for this NDLTD ODL system (http://purl.org/net/etdunion) is shown in Figure 5.

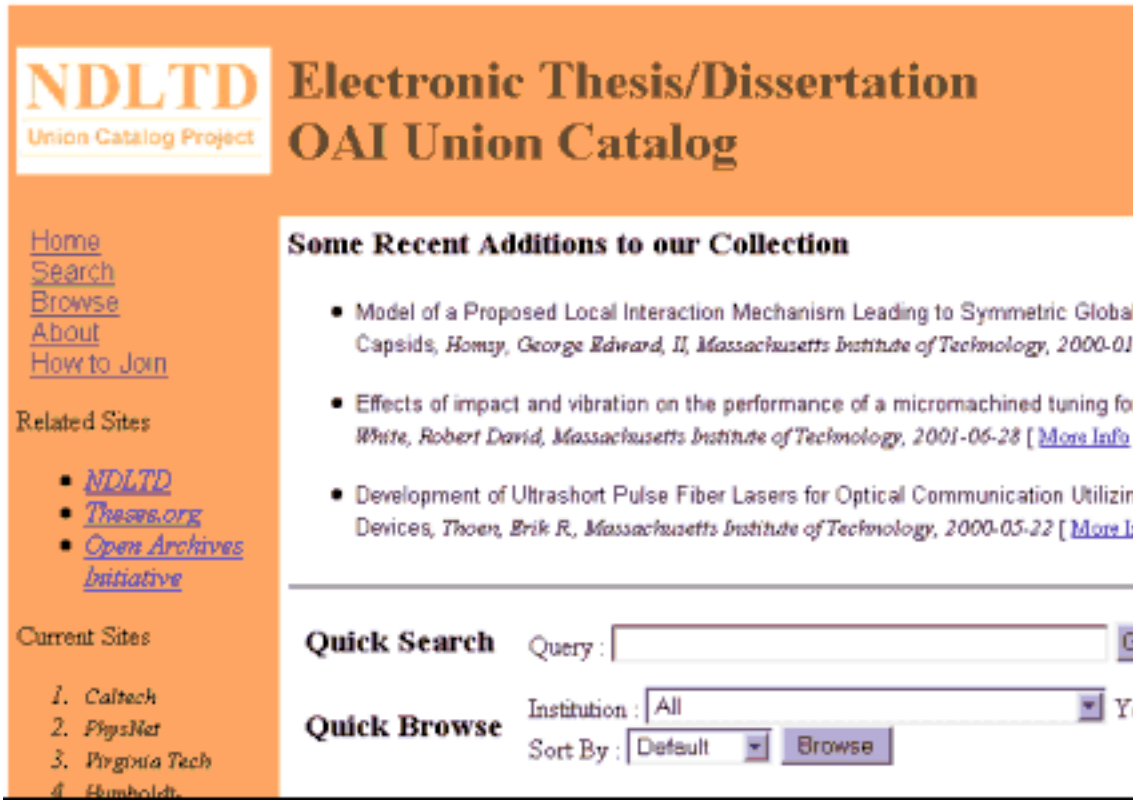

Figure 5. User Interface for NDLTD ODL system, depicting browsing operation.

The design and implementation of the NDLTD ODL system conformed to the design principles stated earlier and thus provide an extensible framework for future expansion. Initial reactions to the design have been positive, with simplicity cited by some users as a motivation for using such a system.

Equivalence of the ODL network to a monolithic system is an important criterion for acceptance. (In this case we can compare the network with VTLS Inc.'s Virtua system union catalog service for NDLTD, see <http://www.vtls.com/ndltd>.) Some work has already been done on techniques to improve speed and reduce redundancy in storage. Substantial speed improvements were realized by using caching and persistent CGI script facilities provided by FastCGI and SpeedyCGI. Storage requirements were reduced by obtaining records on demand from co-located metadata archives. Network performance was tuned through some initial studies of the complexity of the OAI approach to harvesting. The net result of designing by principle and implementing for real-world scenarios is that the resulting system provides the expected level of service while still abiding by the principles of software engineering as applied to networked information systems, and the design and development of Digital Libraries in particular.

Future Work

It is hoped that the results of this ongoing work will change the way people build Digital Libraries. The evaluations and feedback received from users and colleagues strengthen the case for building on the OAI protocol to support high-level services, and composition of those services into complete Digital Library systems.

Building upon a foundation of extensibility, it then will be possible to work on providing more interesting services to users, thus bridging the wide gap between current research and production systems, and

ultimately making information more accessible to more people.

References

Baldonado, Michelle, Chen-Chuan K. Chang, Luis Gravano, and Andreas Paepcke. (1997). "The Stanford Digital Library Metadata Architecture", in *International Journal on Digital Libraries*, Vol 1, No 2, pp. 108-121. Available <http://www-diglib.stanford.edu/cgi-bin/get/SIDL-WP-1996-0051>.

Berners-Lee, Tim and Mark Fischetti. (1999). *Weaving the Web*, Harper, San Francisco.

Birmingham, William P. (1995). "An Agent-Based Architecture for Digital Libraries", in *D-Lib Magazine*, July 1995. Available <http://www.dlib.org/dlib/July95/07birmingham.html>.

Borgman, C. L. (1999). "What are digital libraries? Competing visions.", in *Information Processing and Management*, Vol 35, No 3, pp. 227-243.

CSTC. (2001). *Computer Science Teaching Center*. Website <http://www.cstc.org/>.

Dijkstra, Edsger. (2001). "The End of Computing Science", in *Communications of the ACM*, Vol 44, No 3, March 2001, p. 92.

Fielding, R., J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee. (1999). *RFC2616: Hypertext Transfer Protocol - HTTP 1.1*, Network Working Group, June 1999. Available <ftp://ftp.isi.edu/in-notes/rfc2616.txt>.

Gladney, H., Z. Ahmed, R. Ashany, N. J. Belkin, E. A. Fox, and M. Zemankova. (1994). *Digital Library: Gross Structure and Requirements (Report from a Workshop)*, IBM Almaden Research Center, Virginia Tech Dept. of Computer Science, June 1994.

Harnad, S. (1999). "Free at Last: The Future of Peer-Reviewed Journals", in *D-Lib Magazine*, Vol 5, No 12, December 1999. Available <http://www.dlib.org/dlib/december99/12harnad.html>.

ISO. (1994). *ISO/IEC 7498-1:1994, Open Systems Interconnection Basic Reference Model: The Basic Model*, International Organization for Standardization.

Lagoze, C. and J. R. Davis. (1995). "Dienst - An Architecture for Distributed Document Libraries", in*Communications of the ACM*, Vol 38, No 4, p. 47.

Lagoze, Carl and Herbert Van de Sompel. (2001). *The Open Archives Initiative Protocol for Metadata Harvesting*, Open Archives Initiative, January 2001. Available <http://www.openarchives.org/OAI/openarchivesprotocol.htm>.

NHSE. (2001). *Repository-in-a-Box*. Website <http://www.nhse.org/RIB/>.

Norman, Donald. (1990). *The Design of Everyday Things*, Currency/Doubleday, New York.

OAI. (2001). *Open Archives Initiative*. Website <http://www.openarchives.org/>.

OpCit. (2001). *E-Prints*. Website <http://www.eprints.org/>.

Roscheisen, M., M. Baldonado, C. Chang, L. Gravano, S. Ketchpel, and A. Paepcke. (1998). "The

Stanford InfoBus and Its Service Layers: Augmenting the Internet with Higher-Level Information Management Protocols", in *Digital Libraries in Computer Science: The MeDoc Approach, Lecture Notes in Computer Science*, No. 1392, Springer, 8 August 1998. Available <http://dbpubs.stanford.edu:8090/pub/1998-25>.

Sieve. (2001). Sieve. Website <http://simon.cs.vt.edu/sieve/>.

Suleman, Hussein. (2001). "Enforcing Interoperability with the Open Archives Initiative Repository Explorer", in *Proceedings of the ACM-IEEE Joint Conference on Digital Libraries*, Roanoke, VA, 24-28 June 2001, pp. 63-64.

Suleman, Hussein, Anthony Atkins, Marcos A. GonÁalves, Robert K. France, Edward A. Fox, Vinod Chachra, Murray Crowder, and Jeff Young. (2001). "Networked Digital Library of Theses and Dissertations: Bridging the Gaps for Global Access - Part 1: Mssion and Progress, and Part 2: Services and Research", in *D-Lib Magazine*, Vol 7, No 9, September 2001. Part 1 is available at <http://www.dlib.org/dlib/september01/suleman/09suleman-pt1.html>
Part 2 is available at <http://www.dlib.org/dlib/september01/suleman/09suleman-pt2.html>.

Umar, Amjad. (1997). *Object-Oriented Client/Server Internet Environments*, Prentice Hall, New Jersey.

Van de Sompel, Herbert and Carl Lagoze. (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>.

DOI: 10.1045/december2001-suleman