Literature Review: Digital Library System Architectures

Lighton Phiri
Department of Computer Science
University of Cape Town, Cape Town, South Africa
lphiri@cs.uct.ac.za

Abstract

A number of Digital Library System architecture designs in existence are complex, effectively making systems difficult to extend, reuse and maintain. As a result, an alternative solution that will make it possible for systems to be easily adopted and managed is required. This literature review discusses design architectures currently employed to implement Digital Library Systems and their underlying frameworks. In particular, the focus of attention is on systems in environments characterised by limited resources, such as those used for cultural heritage collections in developing countries.

1 Introduction

A Digital Library is an informal collection of information, stored in digital formats and accessible over a network, together with associated services [1]. Research on digital libraries flourished in the mid 1990s [2-5] with the advent of the Internet coupled with the need to make information open and easily accessible.

The role of a Digital Library is essentially to collect, manage, preserve and make accessible digital objects [4]. To this effect, at a minimum, the core services expected of a Digital Library System include: a repository service for storing and managing digital objects; a search service to facilitate information discovery; and a user interface through which end users interact with the digital objects.

This literature survey is aimed at investigating existing Digital Library frameworks, reference models and open source Digital Library Systems. The review is focused on design considerations of Digital Library Systems. A case study of cultural heritage collections - a class of Digital Library collections - investigates common approaches used to design and implement cultural heritage collections. The hypothesis is that the formalisms, abstractions and APIs of architectures used to implement Digital Library Systems eventually result in complex software stacks that become difficult to maintain, extend and reuse, thus rendering them difficult to adopt.

The cost involved in the acquisition and initial set up of open source software for managing digital collections might be minimal; the initial costs incurred when setting up digital collections and archives is a mere fraction of the ongoing maintenance costs [6,7]. Lawrence et al. [7] showed, using statistical data collected by the Association of Research Libraries, that the median life cycle costs of maintaining digital media was orders of magnitude higher than the initial costs of setting them up. This is especially high for manuscripts and other historical archives due to their long expected physical life.

The remainder of this paper is divided into the following sections: section 2 is a discussion of the various reference models and software frameworks that have been applied to the implementation of Digital Library software systems; section 3 discusses five commonly used open source Digital Library software systems; and section 4 presents some approaches and techniques that have been used to implement heritage collections.

2 Digital Libraries Reference Models and Frameworks

A reference model is an abstract framework that provides basic concepts used to understand the relationships among items in an environment. The Organization for the Advancement of Structured Information Standards (OASIS) [8] states that a reference model consists of a minimal set of unifying concepts, axioms and relationships within a particular problem domain, and is independent of specific standards, technologies, implementations or other concrete details.

A number of Digital Library reference models have been suggested; a discussion of three popular ones now follows.

2.1 DELOS Digital Library Reference Model

The DELOS[1] Digital Library reference model was initiated on the premise that the Digital Library universe was a complex domain that could not be captured using a single definition [9].

The reference model identified three different systems operating within the Digital Library universe: a Digital Library (DL) representing an organisation that collects, manages and preserves digital content; a Digital Library System (DLS) for implementing DL facilities; and a Digital Library Management System (DLMS) comprising of tools for administering the DLS. Figure 1 below shows the interaction between the three sub-systems that ultimately results in a three-tier framework. The organisation of three distinct systems (Digital Library, Digital Library System and Digital Library Management System) into one framework makes the DELOS model complex in nature.

The DELOS model introduces specialised domains into the Digital Library universe that together help to model a generic information system:

- Content: this domain comprises of the digital objects that are made available to end users of the system.
- User: the user domain represents actors (end users or automated systems) who interact with the system.
- Functionality: this represents the subset of services that are supported by the system.
- Policy: the policy domain comprises of rules and conditions that govern the operation of the system. This may include digital rights associated with the content hosted by the system.
- Quality: represents all aspects needed to assess the quality of the system.
- Architecture: this domain generally represents the software and hardware stack of the system.

[1] http://www.delos.info/

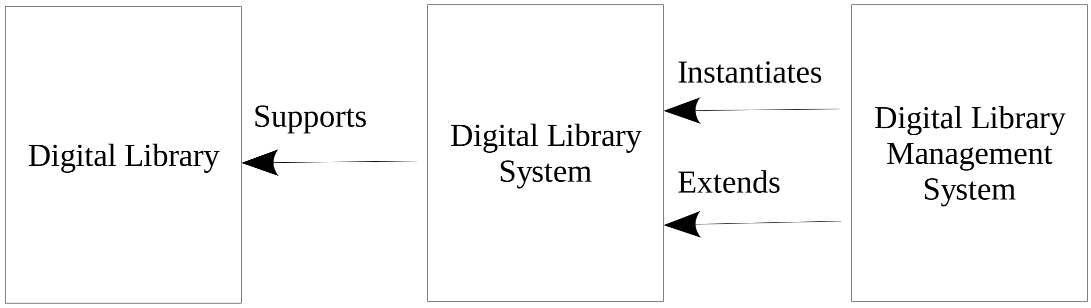

Fig 1. DL, DLS, DLMS: A Three-tier Framework

2.2 Kahn and Wilensky Framework

Kahn and Wilensky [10] outlined a framework for distributed digital object services with digital objects as the main building blocks. The framework is in fact a generic infrastructure with an open architecture that supports large and distributed digital information services.

The Kahn and Wilensky framework has been used as a model for implementation of Digital Library Systems [11]. However, the framework describes an infrastructure that is too general-purpose and does not provide enough detail on how to implement specific digital library concepts such as long term preservation.

The Kahn and Wilensky framework only addresses the Content, Functionality and User main concepts of the DELOS Digital Library reference model. The Quality and Policy main concepts are not dealt with.

2.3 Streams, Structures, Spaces, Scenarios and Societies: 5S Framework

The 5S framework is based on formal definitions and abstraction of five fundamental concepts [12]:

- Streams: sequences of elements of arbitrary type, such as bits, characters or images, that can be used to model static or dynamic content.
- Structures: well-structured information objects, taxonomies, system connections or relationships to facilitate organisation of parts of a whole.
- Spaces: sets of objects, with their associated operations.
- Scenarios: sequence of events that may have parameters.
- Societies: sets of entities and relationships.

Gonçalves et al. [12] showed that the approach is viable by using it to formally define a minimal Digital Library.

2.4 Summary

The 5S framework is too general purpose in nature [9]. The five levels of abstraction and their associated formalisms also render it difficult to adopt due to the complexities involved. Moreover, unlike the DELOS model, the Kahn and Wilensky framework and the 5S framework only covers the Content, Functionality, Quality and User main concepts. The Policy main concepts are not dealt with.

3 Digital Library Architectures

A number of open source Digital Library Systems have been developed over the past two decades. A discussion on design approaches used in the implementation of these systems follows.

3.1 Centralised Component Architectures

In centralised component architecture, information and components interact via major central hubs, unlike decentralised systems where heterogeneous information services are distributed throughout a network.

3.1.1 CDSware/CDS Invenio

CDS Invenio, formally known as CDSware, is an open source repository tool that was developed at CERN[2] and was originally designed to run the CERN document server[3]. CDS Invenio provides an application framework with necessary tools for building and managing a Digital Library server [13,14].

CDS Invenio was designed to run on GNU/Unix systems, with a MySQL database backend server and Apache/Python Web application server. The architecture is made up of modules, with specifically defined functionalities, that interact with one another, the database and the interface layers [13].

CDS Invenio's internal metadata structure uses the MARC 21 [15] format and was specifically chosen because it is a well established standard and can easily be used along side modern mark-up technologies such as XML. The extensible nature of MARC 21 enables CDS Invenio to handle virtually any type of digital object. CDS Invenio complies with OAI-PMH [13][13].

CDS Invenio's modular approach has its own shortcomings as dedicated effort is required to ensure that vital components of the system are running. For instance, BibSched, the task scheduler daemon, needs to be checked on periodically to ensure that it is running as it is the central module that allows other modules to access the database in a controlled manner. Third-party software components, such as the MySQL RDBMS, normally require to dedicated human effort once the system is deployed.

3.1.2 DSpace

DSpace is an open source digital repository system that was developed as a collaborative partnership between Massachusetts Institute of Technology (MIT) and Hewlett-Packard (HP) [16] to help ease the problems at MIT with regard to the collection, preservation, indexing and distribution of research materials and scholarly publications. DSpace's informational model is based on an organisational structure. It is open and freely available. It can handle various digital formats and it supports OAI-PMH. It allows two levels of digital preservation: bit preservation and functional preservation.

The system was implemented in Java and runs within a servlet engine container (e.g. tomcat) with a relational database management system (PostgreSQL and Oracle are currently supported) as a backend [16,17]. As shown in Figure 2, DSpace is organised into a three-tier architecture, comprising of: an application layer with components that make it possible for users to access information stored in the repository; the business

[2] http://www.cern.ch/

[3] http://cdsweb.cern.ch/

logic layer that handles the overall management of dynamic content in the archive and the workflow process; and the storage layer that is responsible for the physical storage of metadata and digital objects.

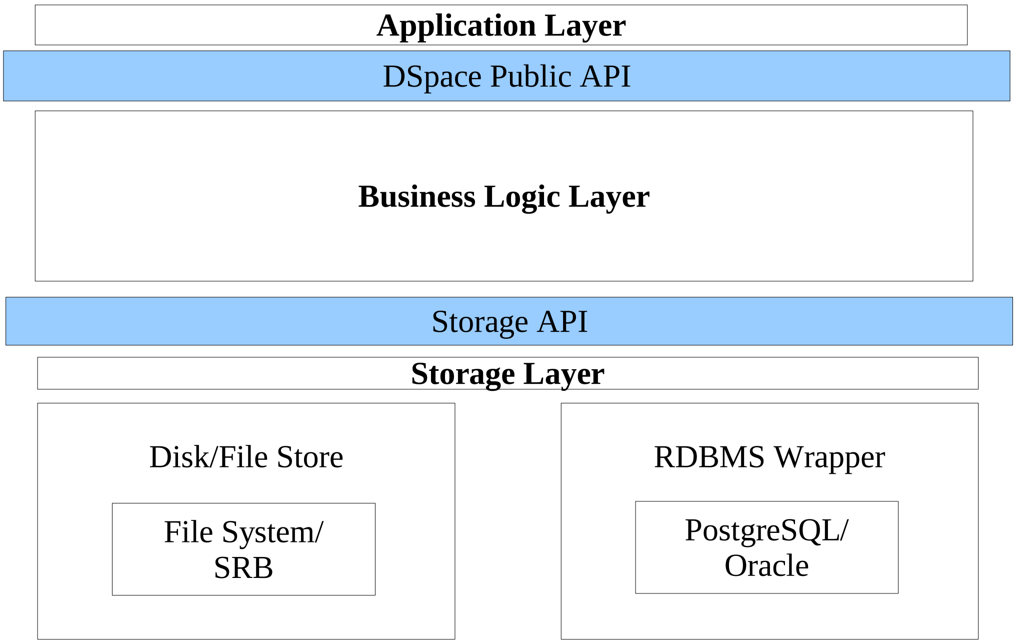

Fig 2. DSpace Three-tier Architecture

At a very high level, DSpace's data model consists of a hierarchy of communities, collections, items, bundles and bitstreams. The bitstreams are stored on the file system whilst their corresponding metadata are stored in a database.

DSpace requires a number of version-specific pre-requisite third-party components and tools before it can be installed and run. The current version (1.7.1), for instance, requires Oracle Java JDK 6, Apache Maven 2.2.x, Apache Ant 1.7 or later, a relational database management system and a servlet engine.

Further more, the storage layer's RDBMS wrapper stores the metadata in a vendor-specific binary format (depending on which RDBMS is used as a data store), making it difficult to migrate information to alternate platforms when the need arises.

As a result, in as much as it is a useful tool, it is mostly suitable for large organisations with the necessary personnel to ensure the effective operation of the system.

3.1.3 EPrints

EPrints[4] is a generic archive software tool designed to create highly configurable Web-based archives. It was developed to help foster open access to research publications and provides a flexible Digital Library platform for building repositories, making it particularly suited to building open access repositories that are compliant with the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) [18].

EPrints runs within an Apache HTTP server and uses a MySQL database server to store metadata about records and users. The actual files in the archive-e-prints-are stored on the file system. EPrints 3 - the current version - was implemented using a

[4] http://www.eprints.org/

modular design with plug-in support, which can be configured for import and export options, changes to the interface and new ways for users to enter data [19].

An EPrints installation can host more than one archive at any given instance. A document within an EPrints archive can be composed of more than one file; an HTML page for instance, can be composed of image files. Each document is stored on the file system while the corresponding metadata records are stored in a MySQL database.

3.2 Distributed Component Architectures

Distributed component architectures are characterised by heterogeneous information services, distributed across a network. The ultimate goal of such architectures, however, is to provide end users with a consistent, transparent and integrated view of the different sources of information [20].

3.2.1 Greenstone

Greenstone is an open source software tool for building and distributing Digital Library collections [21]. The main aim for the development of the software was to empower universities, libraries and other public institutions to build their own Digital Libraries. Greenstone's ability to redistribute collections on a self-installing CD-ROM has particularly made it a popular tool in regions with very slow Internet collections.

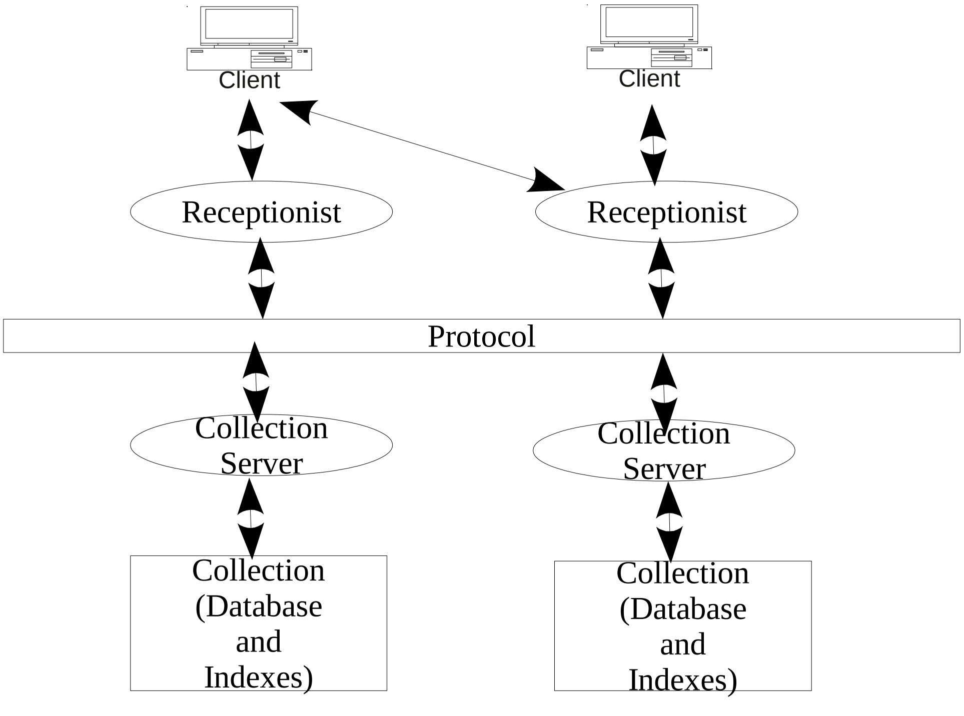

Fig 3. Greenstone Distributed Architecture

The current version, Greenstone3, was implemented as a Java servlet and runs within a servlet container. The architectural design, shown in Figure 3, is agent-based and logically composed of a back end, known as the digital repository site, and a front end user interface, known as the receptionist. This agent-based architectural design is decentralized, making the system scalable, flexible and extensible [22]. The flexibility of the architectural design enables Greenstone to support distributed collections served from different machines, but at the same time maintaining a consistent presentation view for the end users.

The Collection building process is a sequence of distinct phases characterised by expansion, recognition, encoding, extraction, classification and indexing. The Data in the Digital Library System is classified into documents and resources. Documents are expressed in XML. A document that undergoes the building process is encoded, using a specific plugin, into a METS document framework representation. The METS encoded representation is then stored into an SQL database for rapid retrieval, and eventual access from the live Digital Library System.

3.2.2 Fedora

Flexible Extensible Digital Object Repository Architecture (Fedora) is an open source digital object management system. The Fedora repository is capable of storing any type of digital content due to its flexibility, and provides a basis for ensuring long-term preservation of digital content [11].

Fedora was implemented using Java and was designed to run as a Web service within a servlet container. The distributed architecture, shown in Figure 4, consists of three layers: the Web services layer, used for managing and accessing the repository; the core subsystem layer, that implements operations necessary for access, management and storage of digital content; and the storage layer for storing digital content, metadata and indexes. The core Fedora service is the repository service that facilitates interaction with digital objects via a Web service Application Programming Interface (API).

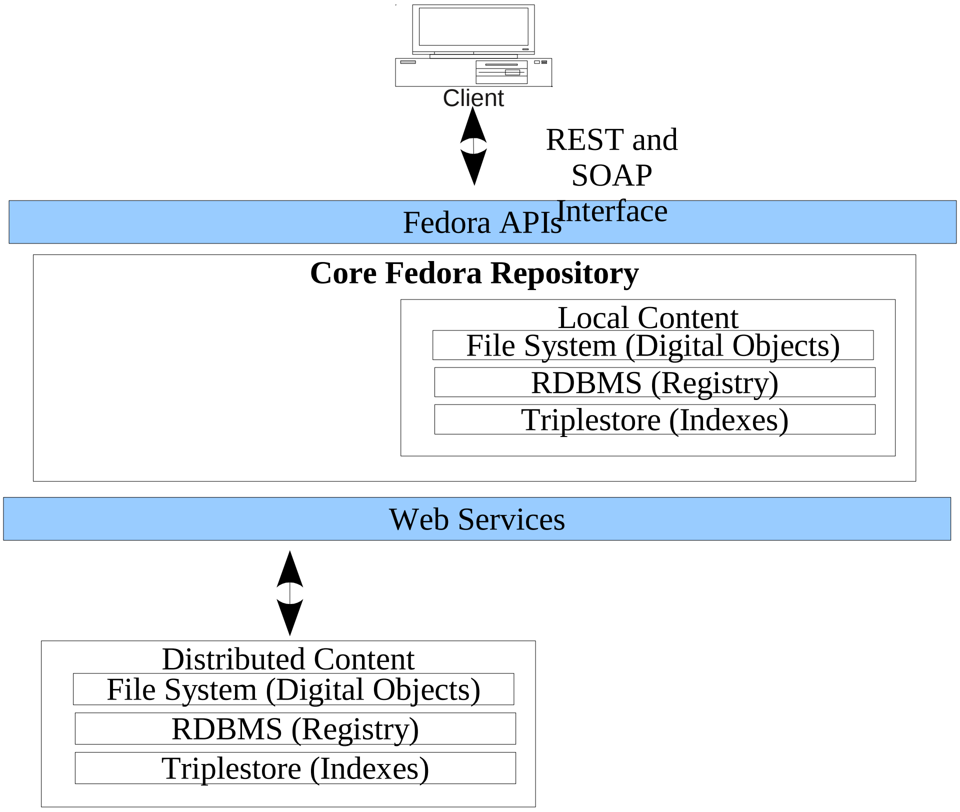

Fig 4. Fedora Distributed Architecture

The Fedora architectural design is based on the Kahn and Wilensky framework [10] and is based on a digital object model that manages objects within an internal service framework composed of loosely-coupled services [11]. Data objects are represented by a FOXML file - an XML file - that contains the necessary information required to find all the components of the object, as well as information required for long-term preservation. This object model feature particularly makes it possible for complex digital objects to make reference to content stored on remote storage systems.

3.3 Summary

Digital content of any type is support by all the systems. However, some of them require experts to customise. Greenstone, for instance, requires development of specialised plugins for each type of media supported.

The standard Dublin Core[5] set of elements is supported by all the tools for furnishing metadata and describing items. This presents a number of advantages since it makes it possible for information to be represented in a standard format. However, this is not used by default by some systems; Fedora uses FOXML while CDS Invenio uses METS.

All the Digital Library Systems are interoperable, as they support OAI-PMH at a bare minimum. This is desirable since it makes it possible for digital content to be accessible by other external systems.

The high level architectural design of all the Digital Library Systems conforms to the traditional three-tier architecture, comprising of; a client layer for interacting with the user interface; a business logic layer with the necessary operations for managing the digital objects; and a storage layer, for storing digital content with corresponding metadata in a database management system. The use of such designs complicates the architecture of the system through the use of third-party components such as database management systems. The metadata stored in databases might also hinder preservation since information is normally stored in proprietary formats. Fedora is an exception, since the metadata is stored on the file system, however, migration of a Fedora repository requires a complete rebuild of the entire system, which is a process that takes a considerable amount of time.

4 Cultural Heritage Collections

An increasing number of Digital Library Systems have been implemented to help preserve cultural heritage. A number of architectures have been proposed to help facilitate the implementation of these systems, each unique to specific environments [23,24].

Greenstone, possibly due to its association with United Nations Educational, Scientific and Cultural Organization (UNESCO), has been extensively used to implement cultural and heritage collections [25].

Most literature on implementation of heritage collections points to the fact that custom-built software tools are preferred to generic Digital Library Systems. Suleman et al. [26] suggested a novel approach, using the latest Web technologies, to leverage online and offline digital collections that is particularly useful in regions with a very slow Internet connection. There are other approaches that make use of open source Digital Library Systems by customising them to suite their needs. The Tufts Digital Library (TDL) [23] is one such example; the core of the architecture - the repository system - uses an extensively customized and enhanced version of Fedora to integrate a number of heterogeneous digital collections.

[5] http://dublincore.org/

There are also other novel techniques that have been suggested for heritage collections: Suleman [27] developed an XML-centric solution for heritage collections that addresses preservation, scalability and efficiency concerns.

Some approaches have proposed the use of enterprise architecture frameworks: Kurniawan et al. [28] developed a cultural heritage portal using an e-cultural heritage architectural framework that is based on the Zachman [29] enterprise architecture framework.

The use of different approaches to solve the digital cultural heritage preservation problem indicates that most of the generic solutions are not particularly suitable for most cultural heritage organizations. This could be partly because most of the generic solutions require experts to install, configure and maintain software. Additionally, most of the generic solutions (with the exception of Greenstone) were specifically implemented to solve problems that institutional repositories were experiencing. A lightweight, cost effective generic solution is thus required for such environments.

5 Conclusion

In conclusion, it is evident from the widespread use of various Digital Library Systems that no one specific software tool exists that is suited for cultural heritage collections. The complex nature of the abstractions used in the underling frameworks used to implement Digital Library Systems further make it difficult to extend and reuse components once these systems are deployed. In a usability study [30] of digital repository software, Nils and Suleman identified problems associated with administrative usability of digital repository tools and, while the study was focused on DSpace, it can be argued that the results are applicable to other systems.

It is also interesting to note that the design and implementation of Digital Library System pays very little attention to the eventual impact of the architectures used on Digital Library maintenance costs.

Greenstone appears to be the most widely used system for implementation of cultural heritage collections. However, the very fact that specialised software libraries are needed for it to function appropriately makes the software stack complex, requiring specialised human resources to operate the system.

6 Bibliography

[1] W.Y. Arms, *Digital Libraries*, Cambridge, Massachusetts: The MIT Press, 2000.

[2] C. Lagoze, “The Warwick Framework,” *D-Lib Magazine*, vol. 2, Jul. 1996.

[3] W.Y. Arms, “Key Concepts in the Architecture of the Digital Library,” *D-Lib Magazine*, vol. 1, Jul. 1995.

[4] W.Y. Arms, C. Blanchi, and E.A. Overly, “An Architecture for Information in Digital Libraries,” *D-Lib Magazine*, vol. 3, Feb. 1997.

[5] C. Lagoze and D. Fielding, “Defining Collections in Distributed Digital Libraries,” *D-Lib Magazine*, vol. 4, Nov. 1998.

[6] B. Hole, P. Wheatley, L. Lin, P. McCann, and B. Aitken, “The Life3 Predictive Costing Tool for Digital Collections,” *New Review of Information Networking*, vol. 15, Nov. 2010, pp. 81-93.

[7] S.R. Lawrence, L.S. Connaway, and K.H. Brigham, “Life cycle costs of library collections: Creation of effective performance and cost metrics for library resources,” *College & Research Libraries*, vol. 62, 2001, p. 541.

[8] M. MacKenzie, K. Laskey, F. Mccabe, P.F. Brown, R. Metz, and B.A. Hamilton, *Reference Model for Service Oriented Architecture 1.0*, 2006.

[9] L. Candela, D. Castelli, N. Ferro, Y. Ioannidis, G. Koutrika, C. Meghini, P. Pagano, S. Ross, D. Soergel, M. Agosti, M. Dobreva, V. Katifori, and H. Schuldt, *The DELOS Digital Library Reference Model (Version 0.98) - Foundations for Digital Libraries*, 2007.

[10] R. Kahn and R. Wilensky, “A framework for distributed digital object services,” *International Journal on Digital Libraries*, vol. 6, Mar. 2006, pp. 115-123.

[11] C. Lagoze, S. Payette, E. Shin, and C. Wilper, “Fedora: an architecture for complex objects and their relationships,” *International Journal on Digital Libraries*, vol. 6, Dec. 2005, pp. 124-138.

[12] M.A. Gonçalves, E.A. Fox, L.T. Watson, and N.A. Kipp, “Streams, structures, spaces, scenarios, societies (5s),” *ACM Transactions on Information Systems*, vol. 22, Apr. 2004, pp. 270-312.

[13] A. Pepe, T. Baron, M. Gracco, J.Y. Le Meur, N. Robinson, T. Simko, and M. Vesely, “CERN Document Server Software: the integrated digital library,” *ELPUB 2005 conference, Heverlee (Belgium)*, Citeseer, 2005, p. 8–10.

[14] M. Vesely, T. Baron, J.Y. Le Meur, and T. Simko, “Creating open digital library using XML: implementation of OAi-PMH protocol at CERN,” *Presented at Elpub*, 2002.

[15] The Library of Congress, “MARC Standards,” *Library of Congress - Network Development and MARC Standards Office*, 1960.

[16] R. Tansley, M. Bass, D. Stuve, M. Branschofsky, D. Chudnov, G. McClellan, and M. Smith, “The DSpace institutional digital repository system: current functionality,” *2003 Joint Conference on Digital Libraries, 2003. Proceedings.*, IEEE Comput. Soc, 2003, pp. 87-97.

[17] M. Smith, M. Barton, M. Branschofsky, G. McClellan, J.H. Walker, M. Bass, D. Stuve, and R. Tansley, “DSpace - An Open Source Dynamic Digital Repository,” *D-Lib Magazine*, vol. 9, Jan. 2003.

[18] C. Gutteridge, “GNU EPrints 2 Overview,” *11th Panhellenic Academic Libraries Conference*, 2002.

[19] P. Millington and W.J. Nixon, “EPrints 3 Pre-Launch Briefing,” *Ariadne*, Jan. 2006.

[20] M.A. Gonçalves, R.K. France, E.A. Fox, and V. Tech, “MARIAN : Flexible Interoperability for Federated Digital Libraries,” *Proceedings of Research and Advanced Technology for Digital Libraries, 5th European Conference (ECDL 2001)*, 2001.

[21] I.H. Witten, S.J. Boddie, D. Bainbridge, and R.J. McNab, “Greenstone: a comprehensive open-source digital library software system,” *Proceedings of the fifth ACM conference on Digital libraries - DL ’00*, 2000, pp. 113-121.

[22] P.C. Weinstein, W.P. Birmingham, and E.H. Durfee, “Agent based digital libraries: decentralization and coordination,” *IEEE Communications Magazine*, vol. 37, 1999, pp. 110-115.

[23] A. Kumar, R. Saigal, R. Chavez, and N. Schwertner, “Architecting an extensible digital repository,” *Proceedings of the 4th ACM/IEEE-CS joint conference on Digital libraries*, New York, New York, USA: ACM, 2004, p. 2–10.

[24] P.B. Viterbo and D. Gourley, “Digital humanities and digital repositories,” *Proceedings of the 28th ACM International Conference on Design of*

Communication - SIGDOC ’10, New York, New York, USA: ACM Press, 2010, p. 109.

[25] U. of W. New Zealand Digital Library Project, “Greenstone Example Collections,” *http://www.greenstone.org/examples*, 2000.

[26] H. Suleman, M. Bowes, M. Hirst, and S. Subrun, “Hybrid online-offline digital collections,” *Proceedings of the 2010 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on - SAICSIT ’10*, New York, New York, USA: ACM Press, 2010, pp. 421-425.

[27] H. Suleman, “Digital Libraries Without Databases: The Bleek and Lloyd Collection,” *Proceedings of Research and Advanced Technology for Digital Libraries, 11th European Conference (ECDL 2007)*, L. Kovács, N. Fuhr, and C. Meghini, eds., Budapest, Hungary: Springer Berlin / Heidelberg, 2007, pp. 392-403.

[28] H. Kurniawan, A. Salim, H. Suhartanto, and Z.A. Hasibuan, “E-CULTURAL HERITAGE AND NATURAL HISTORY FRAMEWORK : AN INTEGRATED APPROACH TO DIGITAL,” *2011 International Conference on Telecommunication Technology and Applications*, Singapore: IACSIT Press, 2011, pp. 177-182.

[29] J.A. Zachman, “A framework for information systems architecture,” *IBM Systems Journal*, vol. 26, 1987, pp. 276-292.

[30] N. Körber and H. Suleman, “Usability of Digital Repository Software : A Study of DSpace Installation and Configuration,” *ICADL 08 Proceedings of the 11th International Conference on Asian Digital Libraries: Universal and Ubiquitous Access to Information*, G. Buchanan, M. Masoodian, and S. Cunningham, eds., Berlin / Heidelberg: Springer, 2008, pp. 31-40.