A Literature Review on Institutional Repositories and Legacy Database Migration

CRAIG FELDMAN, University of Cape Town

The long term and effective storage of data is a critical aspect of many organisations and institutions and, as such, much research has been published in this field. From a university's perspective, it is important that their scholarly output is preserved to allow for perpetual access to prior research, data and experiments produced through the university. Thus, they must ensure that it is stored with the intention of providing long term access to this information. This literature review will explore the architecture of previous Institutional Repositories (IRs), as well as the tools used in conjunction with these IRs. Another concern with repositories is that of providing standards for interoperability and effective repository management. Previous implementation projects are discussed with the intention of learning from their experiences. The main purpose of this literature review is to facilitate the development of the tools and knowledge required to perform a database migration for the South African National Research Foundation to a DSpace repository. Thus, prior literature concerning legacy database migration needs to be considered.

• **Applied computing~Digital libraries and archives**

Additional Key Words and Phrases: Institutional repositories, DSpace, preservation, database migration

1. INTRODUCTION

The National Research Foundation (NRF) of South Africa has two legacy databases that store information on NRF funded projects and theses as well as current and complete projects. The NRF feels that the two databases, and the methods of interacting with these databases, are outdated. Interacting with them is a time consuming process that happens through a command-line interface. Currently, universities send the NRF information about their Masters and PhD level research. An NRF employee then inserts this submission into the database.

The NRF would like the databases to be migrated to DSpace[1], so that authorised university staff can insert new submissions directly into the repository, pending the approval of the NRF. The hope is that this will help to reduce the time taken to capture data and will prove to be a more efficient solution than the one currently implemented.

This literature review aims to identify and categorise previous work that has been done in the fields relevant to this project. Prior implementations of similar projects have been researched, with the hope that this information can prove useful in the implementation of this project. The literature will also be analysed to identify potential gaps in research, and help identify a unique and useful contribution that we can make to this field.

This literature review will start off in Section 2 by providing an overview of institutional repositories (IRs) as well as the architecture of various IRs and their protocols. Various repository management tools will then be discussed in Section 3, with a particular emphasis on DSpace. Thereafter, Section 4 will look at various protocols and standards that are used (or have been proposed) to assist with interoperability among meta-archives. Section 5 will look at prior projects that have been implemented. Finally, Section 6 will provide an insight into the tools and techniques used for database migration, with a particular emphasis on migrating from legacy systems.

2. INSTITUTIONAL REPOSITORIES

An institutional repository is a set of services offered by an institution for the management and distribution of intellectual output (such as research), that has been created by that institution [Lynch 2003]. IRs are an important component in reforming scholarly communication as they reduce the monopoly power held by journals and provide control over scholarship to the academic institutions that created them. Furthermore, an IR can serve as a tangible indicator to the quality of the university that is producing the research whilst improving the universities' reputations [Crow 2002].

[1] http://www.dspace.org

Yeates [2003] and Crow [2002] discuss some of the advantages of IRs. Some of the main benefits of using an IR are:

- The ability to share a broad range of knowledge.
- Opportunities to simplify and extend scholarly dissemination.
- Increased visibility that reflects a high quality of scholarship - leading to tangible benefits for the institution such as increased funding.
- The facilitation of interdisciplinary research and discovery.
- Reduced long term costs involved with the storage, access and dissemination of content.

Yeates [2003] however argued that IRs make use of unproven methods for long term digital preservation and may have high initial costs.

Adewumi and Omoregbe [2011] argue that any institution intending to create an IR must consider the following factors:

- The IR software product model (e.g. open source vs proprietary).
- The IR software features (e.g. file format support, interoperability standards compliance, customisability and advanced search capability).
- Technology costs – Servers, staff, preservation and maintenance costs etc.
- Software support – help provided to users of a particular IR platform.

A large number of different repository software platforms exists. The Registry of Open Access Repositories tracks the activity of various repositories and lists over 30 different repository software platforms [ROAR 2015].

2.1 Architecture of IRs

Adewumi and Omoregbe [2011] examined a number of IR platforms and found that the architecture of these IR platforms could be classified as either open or closed. Open architectures are those that are modular, extensible and are open to modification and access by the public. Closed architectures on the other hand, are not accessible to the public and cannot be altered by anyone other than the proprietor. The architectures can be further subdivided into either a three tier architecture or a plug-in based architecture. The Dienst protocol and the architecture of Fedora, DSpace, and EPrints are discussed in this section.

2.1.1 Dienst

Dienst is a protocol that provides access to distributed, multi-format, document libraries over the Web through a set of interoperating Dienst servers [Lagoze and Davis 1995]. The protocol makes use of HTTP to make Dienst servers accessible from any WWW client [Lagoze et al. 1995]. Three services are provided through these servers, as described by Lagoze and Davis [1995]:

- A repository service to store digital documents (in multiple formats e.g. GIF, TIFF, HTML etc.).
- Indexes of the document collection, coupled with a search engine to return a list of documents that match either a bibliographic or full-text search.
- A user interface that allows the user to browse, search and access data in the collection.

Each document can exist in multiple file formats and can be broken down into a part-whole relationship (for example, into pages or chapters of a book). The architecture of Dienst provides a number of useful abstractions to the user. First, it allows for all elements of a collection to be searchable and accessible, independent of their location. Second, different representations of a document are logically linked to each other and, finally, documents are structured into a part-whole relationship, allowing users to view in part or as a whole.

2.1.2 Fedora

The Flexible and Extensible Digital Object and Repository Architecture (Fedora) was developed to fulfil the need for a reliable and secure means to store and access digital content [Payette and Lagoze 1998]. The Fedora website [2015] describes Fedora as “a modular, open source repository system for the dissemination and management of digital content”. The transition of Fedora from a research prototype to functional repository software began when the University of Virginia Library experimented with the Fedora architecture as a means to manage their archive of complex digital content [Staples and Wayland 2000]. The experiment proved to be successful [Lagoze et al. 2006]. The key features of the Fedora architecture are that it:

- Supports multiple data types.
- Accommodates new data types as they emerge.
- Aggregates data into complex objects.
- Allows for the user to specify multiple content disseminations of these objects, while allowing rights management schemes to be associated with these objects.

Similar to Dienst, the architecture includes a repository, index and UI service. It makes use of collection services to aggregate sets of objects into meaningful collections, as well as naming services to link persistent names to digital objects. Payette and Lagoze [1998] state that this multi-layered service structure has evolved from the concepts implemented in the Dienst Architecture. The underlying architecture of Fedora is that of a ‘digital object’ (that acts as a container to the actual data along with an interface/behaviour layer to give contextual meaning to the data) and a repository component that provides for the management of and access to digital objects. The repository views the digital object as a generic entity known only by its unique name.

Further work on Fedora aimed to integrate advanced content management with Semantic Web technology, by adding support for the representation of rich information networks, whereby nodes represent complex digital objects that combine data and metadata with Web services and edges represent ontological relationships between these digital objects [Lagoze et al. 2006]. The motivation for integrating content management with the Semantic Web originated from the Fedora community, who felt that it was necessary to be able to express well known management relationships (e.g. the organisation of items into a collection) and structural relationships (e.g. part-whole relationships between articles and a journal) among digital objects. Whilst conventional relational databases can represent these relationships, Fedora makes use of the products of the Semantic Web initiative to provide “extensible open-source solutions for representation, manipulation, and querying these knowledge networks” [Lagoze et al. 2006].

2.1.3 DSpace

DSpace makes use of a three-layer architecture which includes a storage, business and application layer [Smith et al. 2003]. Each layer has a documented API to allow for future customisation and enhancement to be easily incorporated. The storage layer makes use of the file system and is managed by PostgreSQL [2] database tables. The business layer holds all DSpace-specific functionality. The application layer covers the interfaces to the system such as the web UI and batch loader. Smith et al. [2003] provide a graphical representation of the DSpace architecture. This is shown in Figure 1. DSpace is discussed further in section 3.3.

[2] An open source object-relational database system - http://www.postgresql.org

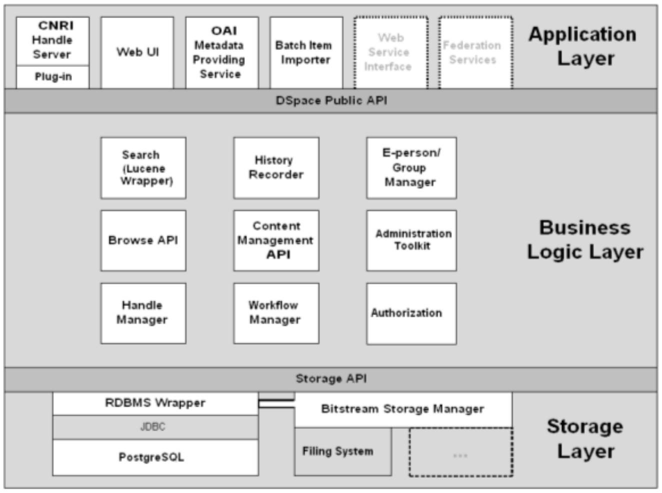

Figure 1 - DSpace technical architecture

2.1.4 EPrints

Unlike DSpace, EPrints[3] makes use of a flexible plug-in architecture for developing extensions [Adewumi and Omoregbe 2011]. It makes use of a generic plugin framework with a set of plug-ins that implement the repository functionality and features such as importing and exporting. The details of EPrints are discussed further in Section 3.2.

3. IR TOOLS

Over the years, the fundamental architectures discussed in the previous section evolved to incorporate the use of more advanced services. Lagoze et al. [2006] argue that technologies for representing digital content should not only be able to match the richness and complexity of their well-established physical counterparts but improve on this as they are freed from the constraints of physical media. Recent papers have focused on theorising how advanced services can improve and connect into these fundamental architectures.

According to Prudlo [2005], LOCKSS, EPrints and DSpace are three of the most widely used and well known repository management tools. As such, these three tools will be discussed further in terms of their cost, functionality, user base and ease of use, with specific emphasis on DSpace as this is the chosen system to be employed by the NRF.

3.1 LOCKSS

Libraries generally purchase subscriptions to electronic journals and not the actual content of a journal, thus access to the content requires the subscription to be renewed (unlike print journals which the library archives for as long as it choses) [Prudlo 2005]. LOCKSS (Lots of Copies Keeps Stuff Safe) is a tool designed for libraries to use to allow them to take indefinite custody of material to which they subscribe [Reich and Rosenthal 2001]. LOCKSS is not an IR, but rather a less specialised content management tool designed to address the need of providing permanent access to Web publications [Yeates 2003]. According to the LOCKSS website, over 530 publishers have selected the Global LOCKSS Network as their e-journal preservation partner [LOCKSS 2015]. Prudlo [2005] argues that the advantage of LOCKSS is that it is cheap and easy to implement and maintain. The LOCKSS website indicates that publishers participate for free but libraries and institutions pay an annual fee of up to $12 216 depending on their size [LOCKSS 2015].

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

3.2 EPrints

EPrints is a research archival tool whose purpose is to give institutions a means to collect, store and provide Web access to electronic material [Prudlo 2005]. Beazley [2011] found that EPrints is a good choice for an institution looking to quickly and easily develop a cost effective IR. Once the IR is established, the user uploads documents and metadata via a simple Web form. One of the major drawbacks found by Beazley [2011] was that EPrints lacked a bulk upload feature and had fairly limited search capabilities.

3.3 DSpace

According to Smith et al. [2003], the development of DSpace's architecture drew on the information provided in Kahn and Wilensky's [2006] Framework for Distributed Digital Object Services (which aimed to describe fundamental aspects of an open architecture infrastructure that supports systems such as digital libraries), as well as Arms [1995] work on Key Concepts in the Architecture of the Digital Library.

DSpace was developed by MIT to address an issue faced by their library of having to collect, preserve, index and distribute an increasing number of scholarly publications and research materials, presented in complex digital formats; this was both time consuming and costly [Smith et al. 2003]. DSpace aims to act as a repository to give digital research and educational material greater visibility and accessibility over time. DSpace was created to address all the basic functionality required in a digital repository service, with the intention of being expanded upon in the future, particularly to address long term data preservation concerns [Tansley et al. 2003]. According to the DSpace website [DSpace 2015], some of the main reasons to use DSpace are that it:

- Can be customised to fit the institution's needs.
- Can easily be installed and configured.
- Can manage and preserve all types of digital content.

3.4 Comparison of EPrints and DSpace

Prudlo [2005] found that unlike EPrints, DSpace was suitable for large institutions that expect to archive materials from a variety of departments and communities. One of the key indicators that can be looked at when deciding between a system such as EPrints and DSpace is the number of institutions that have adopted each product. According to the Registry of Open Access Repositories [2015], 1520 institutions are making use of DSpace, compared to 554 for EPrints. Ultimately the higher adoption of DSpace, coupled with the desire by the NRF to port to this system make it the best choice for this project. A detailed comparison between different open source repository platforms was conducted by the Repository Support Project[4]. A concern arises with how well these tools work in providing easy access to their data. This is discussed in the following section.

4. INTEROPERABILITY

Institutional repository systems must be able to support interoperability to provide for discovery through multiple search engines and other tools [Crow 2002]. According to Suleman [2010], standardization is a key element of interoperability and multiple experiments with protocols have demonstrated what is effective and what is not. It is important that well defined protocols and standards are in place to allow for interoperability among institutions and their repositories. Various protocols and frameworks have been developed to facilitate interoperability among repositories, some of which are discussed below.

[4] http://www.rsp.ac.uk/start/software-survey/results-2010/

4.1 Dissemination

The Open Archive Initiative (OAI) was established as a result of a meeting exploring cooperation among scholarly e-print archives [Van de Sompel and Lagoze 2000]. The OAI aims to facilitate efficient and effective dissemination of content, through the development and promotion of interoperability solutions [Lagoze and Van de Sompel 2001]. One of the initial interoperability protocols was the OAI Protocol for Metadata Harvesting (OAI-PMH), which was based on the Dienst protocol discussed earlier [Lagoze and Van de Sompel 2001]. OAI-PMH is used to harvest the metadata of records in an archive [Breeding 2002]. Metadata describes the nature and content of digital data stored in a repository and OAI-PMH allows third parties to easily gather this metadata from distributed repositories [Crow 2002]. A requirement for OAI-PMH is that OAI compliant providers supply metadata in a common format - the Dublin Core Metadata Element Set[5] [Lagoze and Van de Sompel 2001]. Some of the standard metadata formats supported by IR platforms include: Dublin Core, Qualified Dublin Core, METS[6] and MARC standards[7] [Adewumi and Omoregbe 2011]. DSpace, for example, exposes the Dublin Core metadata for publicly accessible items, so that it can be harvested by search engines [Tansley et al. 2003].

4.2 Structuring and Aggregation

Van de Sompel and Lagoze [2007] argued that existing scholarly communication methods were challenged by the change in the unit of scholarly communication towards one of ‘Compound Information Objects’ whereby these compound objects consist of the results of research that may include datasets, simulations, software, recordings etc. and their aggregations. This led to the development of the OAI Object Reuse and Exchange (OAI-ORE) standard that aims to facilitate the discovery, use and re-use of Compound Information Objects and their components.

4.3 Deposit

The SWORD (Simple Web-service Offering Repository Deposit) interoperability standard was designed to facilitate the interoperable deposit of resources into digital repositories [Lewis et al. 2012]. SWORD can improve the interoperability of deposit by allowing for deposit from multiple locations, deposit to multiple repositories or deposit between repositories [Allinson et al. 2008]. Version 2 of SWORD added support for the whole product life cycle, thus adding support for updating, replacing and deleting resources. Lewis et al. [2012] showed how SWORD facilitated the fulfilment of 9 different use cases. SWORD has implementations that work in DSpace, EPrints and Fedora [Allinson et al. 2008].

5. PREVIOUS PROJECTS

By analysing previous project implementations, we can learn the complexities involved with the implementation of repositories, as well as how obstacles were overcome.

5.1 DSpace

Currently there are about 1520 DSpace repositories tracked by the Registry of Open Access Repositories[8]. The first implementation of DSpace occurred at MIT. In the fall of 2001, a DSpace transition team (consisting of project staff and senior library staff) was given the responsibility of figuring out how DSpace should be released as a new service of MIT Libraries [Smith et al. 2003]. This proved to be useful in helping the library staff become familiar with the system and their feedback was an invaluable contribution to future work on DSpace [Smith et al. 2003].

[5] http://dublincore.org
[6] Metadata Encoding and Transmission Standard - http://www.loc.gov/standards/mets
[7] Machine Readable Cataloging - A set of standards used to identify, store and communicate cataloguing information - http://www.loc.gov/marc
[8] http://roar.eprints.org/

5.2 NETD

In South Africa, a National Electronic Theses and Dissertations (ETD) portal was developed to provide access to a collection of country specific ETDs as well as to assist in the development and management of ETD programmes at various universities [Webley et al. 2011]. The system that was created made use of three separable layers to support expansion and future scalability. The system works by harvesting and storing metadata records for ETDs, which are then made accessible via OAI-PMH. The Lucene[9] open source search engine is used to index the repository and provide a Web interface, which allows the user to browse and search different institution's ETDs via a single Web portal [Webley et al. 2011].

5.3 The United States Library of Congress

The United States Library of Congress (as part of the National Digital Information Infrastructure and Preservation Program) launched a pilot program with the aim of providing perpetual access to digital content by moving its digitised content to the cloud [Allen and Morris 2009]. Other participants in the program included the New York Public Library, who would replicate large collections of digital images from a Fedora repository into DuraCloud[10] - a cloud based data preservation service that stores content with multiple cloud providers.

6. DATABASE MIGRATION

Research has been done into past experiences and practices used for database migration. The complexities of the NRF database migration arise from the need to transition an old legacy system to a new modern system (DSpace). As such, it is important to focus on papers relating to similar projects. According to Bisbal et al. [1997], legacy systems can pose considerable problems, including brittleness, inflexibility, non-extensibility and a lack of openness.

6.1 General Principles of Data Migration

Bisbal et al. [1997] argue that the naïve approach to migrating a legacy system involves redeveloping the system from scratch using modern tools, however, the risk of failure is usually very high when using this approach. Instead, Brodie and Stonebraker [1995] suggest three different approaches. (1) The Forward Migration Method which involves first transferring the legacy data to the new, modern database system and then incrementally migrating the legacy applications. (2) The Reverse Migration method where the legacy applications and interfaces are migrated, followed by the data. (3) The Composite Database approach whereby legacy application are gradually rebuilt and the legacy and target system form a composite system during migration. Wu et al. [1997] proposed the 'Butterfly methodology' as an alternative to the current thinking on legacy system migrations. The Butterfly methodology eliminates the need for users to simultaneously access both the legacy and target systems.

6.2 DSpace Data Migration

The primary means of adding items into DSpace are to either enter each entry via the DSpace Web portal or in batch upload via the DSpace item importer (a command-line tool for batch ingesting items that makes use of a simple archive format) but enhancements to DSpace include new deposit options making use of SWORD, OAI-ORE and DSpace package importers [Walsh 2010]. Many projects have been implemented that make use of scripts to automate the process of creating the archive directory to assist in batch uploading. There is a considerable amount of literature documenting the methods used for batch ingestion to populate institutional repositories. Mishra et al. [2007] and Mundle [2007] developed Perl scripts to create the DSpace archive directory for batch

[9] A high performance, full featured text search engine - https://lucene.apache.org/core/
[10] http://www.duracloud.org

imports of ETDs whereas Brownlee [2009] made use of Python scripts to process CSV files (created using Filemaker[11]). Walsh [2010] describes using Perl scripts to migrate data from spreadsheets and CSV files into the DSpace archive format for the Ohio State University's IR. Ribaric [2009] describes the use of PHP utilities for the automatic preparation of ETDs (from the Internet Archive[12]) for deposit into DSpace. Several other projects are introduced by Walsh [2010].

7. CONCLUSIONS

In this literature review, papers relating to the various repository platforms and their architectures were studied. Some IR management tools are compared, with particular emphasis on DSpace. Much of the literature talked about the need for not only long term data preservation, but also a set of well-defined interoperability standards. It is clear from the literature that standardisation and the wide acceptance of proposed protocols is the best method of creating interoperable repositories. Similar projects to that required by the NRF are studied. It is clear that database migration is a highly researched and studied topic, with a number of methods and case studies being found that could prove useful in the development of a tool to facilitate the NRF database migration. There appears to be the need for a generic tool that would take a (possibly CSV) file as input, and automatically generate a DSpace repository. Such a tool could also attempt to perform automatic matching of fields from the input data to those of the DSpace metadata fields, thus further simplifying the process of migrating a legacy system to DSpace.

REFERENCES

Clifford A. Lynch. 2003. Institutional repositories: essential infrastructure for scholarship in the digital age. *portal: Libraries and the Academy* 3, 2, 327-336.

Raym Crow. 2002. The case for institutional repositories: a SPARC position paper. *ARL Bimonthly Report 223.*

Robin Yeates. 2003. Institutional repositories. *Vine* 33, 2, 96-101.

Adewole Adewumi and Nicholas Omoregbe. 2011. Institutional repositories: features, architecture, design and implementation technologies. *Journal of Computing* 2, 8.

ROAR. 2015. Registry of Open Access Repositories. Retrieved April 16 2015 from http://roar.eprints.org/.

Carl Lagoze and James R. Davis. 1995. Dienst: an architecture for distributed document libraries. *Communications of the ACM* 38, 4, 47.

Carl Lagoze, Erin Shaw, James R. Davis and Dean B. Krafft. 1995. Dienst: implementation reference manual. Technical Report TR95–1514, Dept. of Computer Science, Cornell University.

Sandra Payette and Carl Lagoze. 1998. Flexible and Extensible Digital Object and Repository Architecture (FEDORA). In *Second European Conference on Research and Advanced Technology for Digital Libraries*, Heraklion, Crete, Lecture Notes in Computer Science, 41-59.

Fedora. 2015. Fedora Repository | Fedora is a general-purpose, open-source digital object repository system. Retrieved April 2015 from http://www.fedora.info.

Thornton Staples and Ross Wayland. 2000. Virginia Dons Fedora: A prototype for a digital object repository. *D-Lib Magazine* 6, 7/8.

Carl Lagoze, Sandy Payette, Edwin Shin and Chris Wilper. 2006. Fedora: an architecture for complex objects and their relationships. *International Journal on Digital Libraries* 6, 2, 124-138.

MacKenzie Smith, Mary Barton, Mick Bass, Margret Branschofsky, Greg McClellan, Dave Stuve, Robert Tansley and Julie H. Walker. 2003. DSpace: An open source dynamic digital repository. *D-Lib Magazine* 9, 1.

Marion Prudlo. 2005. E-archiving: An overview of some repository management software tools. *Ariadne* 43.

Vicky Reich and David S. Rosenthal. 2001. LOCKSS: A permanent web publishing and access system. *D-Lib Magazine* 7, 6, 14.

LOCKSS. 2015. Lots of Copies Keeps Stuff Safe. Retrieved April 16 2015 from http://www.lockss.org.

Mike R. Beazley. 2011. EPrints institutional repository software: A review. *Partnership: the Canadian Journal of Library and Information Practice and Research* 5, 2.

Robert Kahn and Robert Wilensky. 2006. A framework for distributed digital object services. *International Journal on Digital Libraries* 6, 2, 115-123.

William Y. Arms. 1995. Key concepts in the architecture of the digital library. *D-lib Magazine* 1, 1.

Robert Tansley, Mick Bass, David Stuve, Margret Branschofsky, Daniel Chudnov, Greg McClellan and MacKenzie Smith. 2003. The DSpace institutional digital repository system: current functionality. In *Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital libraries (JCDL'03),* Houston, TX, USA, IEEE Computer Society, Washington, DC, USA, 87-97.

DSpace. 2015. Top Reasons to Use DSpace. Retrieved April 17 2015 from http://www.dspace.org/why-use.

[11] http://www.filemaker.com

[12] http://www.archive.org

Hussein Suleman (2010). Interoperability in Digital Libraries. In *E-Publishing and Digital Libraries: Legal and Organizational Issues*, Ioannis Iglezakis, Tatiana-Eleni Synodinou and Sarantos Kapidakis, Eds. IGI Global, Hershey, PA, 31-47.

Herbert Van de Sompel and Carl Lagoze. 2000. The Santa Fe convention of the open archives initiative. *D-Lib magazine* 6, 2.

Carl Lagoze and Herbert Van de Sompel. 2001. The Open Archives Initiative: Building a low-barrier interoperability framework. In *Proceedings of the 1st ACM/IEEE-CS joint conference on Digital libraries (JCDL'01),* Roanoke, VA, USA, ACM, 54-62.

Marshall Breeding. 2002. Understanding the Protocol for Metadata Harvesting of the Open Archives Initiative. *Computers in Libraries* 22, 8, 24-29.

Herbert Van de Sompel and Carl Lagoze. 2007. Interoperability for the discovery, use, and re-use of units of scholarly communication. *CTWatch Quarterly* 3, 3, 32-41.

Stuart Lewis, Pablo de Castro and Richard Jones. 2012. SWORD: Facilitating deposit scenarios. *D-Lib Magazine* 18, 1, 4.

Julie Allinson, Sebastien François and Stuart Lewis. 2008. SWORD: Simple Web-service Offering Repository Deposit. In *JISC CETIS EC and MDR SIG meeting,* Strathclyde University, Glasgow, Ariadne.

Lawrence Webley, Tatenda Chipeperekwa and Hussein Suleman. 2011. Creating a National Electronic Thesis and Dissertation Portal in South Africa. In *Proceedings of the 14th International Symposium on Electronic Theses and Dissertations (ETD2011),* Cape Town, South Africa, National Research Foundation.

Erin Allen and Carol M. Morris. 2009. Library of Congress and DuraCloud Launch Pilot Program Using Cloud Technologies to Test Perpetual Access to Digital Content. In *Library of Congress, News Release*.

Jesus Bisbal, Deirdre Lawless, Bing Wu, Jane Grimson, Vincent Wade, Ray Richardson and Donie O'Sullivan. 1997. An overview of legacy information system migration. In *Proceedings of the Fourth Asia-Pacific Software Engineering and International Computer Science Conference (APSEC '97 / ICSC '97),* Clear Water Bay, Hong Kong, IEEE Computer Society, Washington, DC, USA, 529-530.

Michael L. Brodie and Michael Stonebraker. 1995. Migrating legacy systems: gateways, interfaces & the incremental approach. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Bing Wu, Deirdre Lawless, Jesus Bisbal, Jane Grimson, Vincent Wade, Donie O'Sullivan and Ray Richardson. 1997. Legacy system migration: A legacy data migration engine. In *Proceedings of the 17th International Database Conference (DATASEM'97),* Brno, Czech Republic, 129-138.

Maureen P. Walsh. 2010. Batch Loading Collections into DSpace: Using Perl Scripts for Automation and Quality Control. *Information Technology and Libraries* 29, 3, 117-127.

R. Mishra, S. Vijayanand, KPP Noufal and Gaurav Shukla. 2007. Development of ETD Repository at IITK Library using DSpace. In *International Conference on Semantic Web and Digital Libraries (ICSD-2007),* Bangalore, India, Indian Statistical Institute, 249-259.

Todd Mundle. 2007. Digital retrospective conversion of theses and dissertations: an in house project. In *8th International symposium on electronic theses and dissertations (ETD 2005),* Sydney, Australia, NDLTD.

Rowan Brownlee. 2009. Research data and repository metadata: policy and technical issues at the University of Sydney Library. *Cataloging & Classification Quarterly* 47, 3/4, 370-379.

Tim Ribaric. 2009. Automatic Preparation of ETD Material from the Internet Archive for the DSpace Repository Platform. *Code4Lib Journal* 8.