Literature Review: Archiving Archives

22 April 2022

Alex Olivier
University of Cape Town
Rondebosch, Cape Town, South Africa
olvale005@myuct.ac.za

ABSTRACT

Archives are of direct and indirect importance to everyone, especially over the last two decades as technology has allowed digital formats to become useful to society. This has allowed a large amount of data to be transformed and saved in a digital state or more recently born solely in a digital form. It is therefore important to investigate and develop new means of preserving data. This review highlights relevant work in the digital archiving domain, such as the tools, models, standards, and case studies applying these. In particular: the Fedora, DSpace and Greenstone repository model are compared, the Dublin core, METS and PREMIS metadata schemes are reviewed, the Mellon Fedora and Digital Library of India implementations are mentioned, the key characteristics of SimpleDL are included and lastly The Internet Archive is included. This allowed us to identify relevant areas such as: Greentones flat file format, Simple DL's architecture and basic OS components as a basis for archiving archives.

KEYWORDS

digital libraries, archives, data preservation, metadata, repositories

1 Introduction

The work on digital libraries as archives have become of greater importance over the last twenty-five years. Initially, work started out converting traditional library material to a digital form for safekeeping, distribution and searching [1]. Examples include the scanning of files and books and the use of analog to digital converters. More recently, however, items appearing in digital libraries are 'born' digitally. This change, along with the increased scale and development of technology, has seen a tremendous change to the size and importance of digital libraries.

The continuous and rigorous process of the preservation of these digital libraries is therefore mounting. Any improper steps in the process, as archiving is an active state, could lead to a devastating loss to a complete archive [2]. Data that could have been accumulated for years, which is irreplaceable. It is this continual state of archiving, meaning that preservation is a never-ending process, that is of concern to us. While also providing opportunities for us at the same time.

This paper will review relevant work and research in the archiving domain, more specifically a look into digital libraries, their underlying architecture and models currently being used in the space. Along with further tools and case studies to implement and that have been used to implement digital libraries. Finally, an explicit discussion is proposed, a comparison table between three repository models; Greenstone, DSpace and Fedora is drawn up and a summary is provided.

2 Digital Libraries

The main aim of digital libraries is to preserve significant digital data and make it persistently available across networks, for generations to come, for reasons such as education and research [3]. Its integrated set of services include functionality such as capturing, cataloging, storing, searching, protecting, and retrieval of digital information [4].

As more data is inherently 'born' digitally, instead of being print duplicates, it is therefore imperative that the means of archiving the data is robust [5]. Contributors, such as publishers and universities, now assume responsibility for preservation and are therefore active in the creation of archives. To achieve reliability and efficiency, these institutions have often implemented customized models of self-sustaining libraries that adhere to strict formatting standards. This stems from the realization that loosely coupled modular architectures create flexible, extensible, and scalable digital libraries that are configurable to specific use cases [6]. Software packages have made this customization possible by: providing open-source code, incorporating API's (Application Programming Interface) at a layer level or providing plug-in support [7].

2.1 Digital Library Architecture

A common digital library architecture is comprised of 4 systems, called the Generic Digital Library Model, derived from the traditional library components [4]. Conveniently these can be hosted on separate computer systems, creating modularity, over a network such as the Internet. This is beneficial for scaling, access, maintenance, and security. Organizations and institutions also

have multiple choices available to them when considering how they are going to store their digital objects. Big-data-friendly architectures propose hosting a distributed system in the Cloud. This provides the advantage that memory management is distributed, components can be scaled horizontally, and the systems parts are decoupled [7].

Generic Digital Library Model:
Firstly, the repositories are responsible for managing digital objects. This includes the insertion, deletion, and retrieval with restricted access to digital objects [8].
Most modern digital libraries conform to the OAIS (Open Archival Information System) specification, with permanent access to the libraries content using a URI (Uniform Resource Identifier). To simplify matters an interfacing abstraction is applied to the repository called an RAP (Repository Access Protocol) [4]. With the integration of the Internet however, RAP is now used for rich interactions between co-operation repositories, but HTTP-based standards are embraced for web usage [9].

Secondly, the system provides identifiers to where digital objects are stored. This is achieved by assigning 'handles', as general-purpose identifiers, to digital objects in repositories. Users' request these when submitting digital objects to the repository [10]. The repositories are then able to return where the digital object is stored within the repository [4].

Thirdly, digital libraries can accrue large amounts of data. It is therefore likely that many indices and catalogs will be searched during the retrieval of information, these can be independently managed with multiple protocols [4].
Metadata is the core of any information retrieval system and dictates the ability of a digital library to deliver objects in a meaningful way, which greatly affects its long-term preservation ability [11]. Early implementations showed distinctive categories of information that should be captured: descriptive, administrative, technical, rights, digital and structural metadata. It is now understandable why no single schema for metadata collection is prevalent. Libraries are rather implemented with an underlying metadata standard along with the institution's own metadata categories. This allows for a degree of interoperability while fulfilling their own operational requirements.
Preservation metadata is also of concern to ensure the 'fixity' of information as control over the library needs to be maintained. This provides authenticity and validity to its data.

Finally, the user interface integrates the 3 other components, providing a two-part interface. The first part allows users to search and retrieve digital objects, while the second part allows system administrators to manage the collection [4]. The primary interface is usually web browsers that connect to client services, an intermediary service between the browser and the rest of the system. It is common to find HTTP GET or SOAP requests for this interaction [6].

2.1.1 The Dublin Core metadata set, consisting of 15 broad elements, is one of the most common schemas for web content and is widely used as it enables indexing by any metadata search engine [6]. It is recommended to use the schema for general data [8]. Dublin Core is also useful as it can be used on digital and physical resources.

2.1.2 METS (Meta-data Transmission and Encoding Standard), consisting of 7 sections that may contain sub elements and attributes, is commonly used to encode metadata into an XML format which allows objects to be managed and further exchanged between repositories [12]. It acts as a wrapper and encoder around the digital object [8].

2.1.3 PREMIS (PREservation Metadata Implementation Strategies):
PREMIS has been extensively worked on by an authoritative international working group, creating a well-established schema for preservation [11]. Its elements include: the object and its events, agents, and rights associated with it. It is however problematic when trying to incorporate the schema in a framework. For example, it is difficult to fit the agent schema into the framework as it is ambiguous. Attempts have been made to include the schema as a container into the framework, however once again the placement implies some elements are illogically placed.

Further, PREMIS and METS produce many metadata redundancies, especially in the structural element, this produces unnecessary data and causes a storage and priority issue [11]. The use of multiple schemes can also be useful. Big Data sources can be formed from unexpected places such as technical metadata sets to make predictions for example [13].
Other schemas were developed to be used in specific cases: such as the MARC (Machine-Readable Cataloging) schema, originally developed for bibliographic communication and the ISO 19115 standard designed for representing geographical information.

3 Other Architectures and Tools

3.1 Fedora is a commonly used and implements a distributed object paradigm using CORBA (Common Object Request Broker Architecture), which allows the communication between multiple distinct systems [12]. Its real success stems from the Virginia reinterpretation that proved that Fedora could be run as a web application, however sacrificed much of Fedora's advanced interoperability features. This was later fixed by the Mellon Fedora implementation. Key advantages include: its open architecture and data model, the flexible relationships among digital objects and the ease of extending repositories, metadata, relationships, and content types [7].
Other well-known and open-source packages include DSpace and Greenstone.

3.2 GSDL (The Greenstone Digital Library):

Greenstone provides an alternative approach to providing its created collections on the Internet, it also allows for publishing on a DVD or USB Flash Drive [14]. Greenstone is made up of two components: the Receptionist and Collection Server. In a networked implementation, the two components communicate via a server over a user chosen protocol. The Receptionist interfaces with the user and makes requests to the Collection server or servers. The client is Java-based and can use CORBA. The Collection server/s provide an abstraction to the Receptionist for managing the collection. It makes use of two databases: the MG (Managing Gigabytes) for full-text search and retrieval and GDBM (Gnu Database Manager) for collection information. Interestingly, the two components can be combined into a single executable in a single server configuration; the protocol between the components are now direct function calls referred to as a null protocol. Extensibility is supported by plug-ins and the software is freely available via the Gnu public licence [15].

3.3 DSpace:

DSpace, jointly developed by MIT Libraries and Hewlett-Packard Labs, comprises of a three-layer architecture – the application, business logic and storage layer [14]. It was developed to encompass research functionality, while maintaining simplicity [16]. All three provide an API interface for user customization and future enhancement [14]. The storage layer uses PostgreSQL database tables and offers two ways of storing data. It can either use the file system on the server or use SRB (Storage Resource Broker). The application layer supports OAI (Open Archive Interface) for persistent access to its items.

3.4 Simple DL:

Simple DL takes an unconventional approach to building a digital library. The most successful implementations arise from organizations and universities that were well funded either by companies such as HP or the Mellon Foundation [17]. However, many poorer countries and unfunded universities and institutions lack the resources to build, implement or manage a digital library. While some implementations have been made, improper model use and a software failure (middleware) can result in the loss of a complete archive. Simple DL tries to address this gap by providing a practical toolkit enabling long term access to digital libraries even when active preservation is no longer applied. It is able to do this because Simple DL does not implement the traditional backend database and database management system, it stores unstructured data as flat files and structured data as XML [18]. The data is therefore easily distributed and viewed on many devices, however, still provides a basic web application to display collections, which can be customized using CSS and XSLT. The collection is able to keep this static form due to Simple DL requiring data to be pre-processed.

Information retrieval is supported by a tf.idf (term frequency-inverse document frequency) search system in the web application with JavaScript. Results indicated an adequate response time for less than one hundred thousand items [17].

4 Case Studies

4.1 The Fedora System:

The Fedora system is one of a few digital libraries that supports versioning and is documented. It preserves former instantiations by versioning within digital objects [12]. This preserves content and services by creating multiple datastreams and disseminators. However, this type of versioning provides a means to track changes in digital objects over time, from a management perspective, than provide a view of the system at a particular time. This choice seems to stem from the idea that the management subsystem would rather have to deal with versions of disseminators and datastreams than multiple XML files, which stored on the same system would be redundant and further introduce system capacity concerns.

Behavioral service changes are more difficult to represent in a version of a system since the disseminators themselves can be altered. The Fedora System dealt with this by using a versioning strategy that records changes to methods and releases these as upgrades to the behavior service implementation.

4.2 DLI (Digital Library of India):

The DLI uses a data farm with servers implementing a hardware-based RAID (Redundant Array of Independent Disks) in an effort to boost reliability [3]. This will ensure data is not lost when singular occurrences of disk failures occur. For example, The Internet Archive has reported disk replacement rates as high as 6% [19]. The cluster is setup with Linux enhanced by LTSP (Linux Terminal Server Project) [3]. This allows for diskless network booting without devoting storage for OS files, this saves space and allows for easy management of nodes in the cluster, as no configurations or installations are required. The DLI further saves a redundant copy of the data in the case of a server crash, however it does not appear to do the same with the metadata server. This is advantageous if a failure occurs in the cluster which is arguably more likely as it comprises of multiple servers. However, a failure on the metadata server would leave the data in an unusable state.

5 The Internet Archive

Traditional institutions such as national libraries have been focused on preserving our cultural heritage. However, it is estimated that 27% of interesting and important resources shared on social media are lost within 2.5 years [20]. This problem has been recognized by institutions and web preservation has been started. The Internet Archive has the largest collection containing 2.5 Petabytes spanning from 1996. Another notable collection is the Internet Memory Foundation, which focuses on specific topics, domain, and projects. A common drawback of these initiatives is trying to access and explore their data. While modern search engines provide a means via an interface, the Internet Archive's Wayback Machine only allows retrieval of past web pages. This provides an ineffective, manual search technique.

Query logs have been identified as a major component needed to understand users' information needs, as they help rank search results [20]. These however do not exist in the context of archived websites. Kanhabua et al. [20] implemented their own search system leveraging Bing's search engine and entity-orientated searches on the Wayback Machine. To compensate for the lack of query logs, users are only allowed to query entities described in Wikipedia. To help users formulate these queries, query auto-completion and related entity suggestion functionality was included. This is used to return a ranked list of results colour coding the difference between current websites and archived ones.

Drawbacks and improvements to the system include the inability to process complex entity-based queries, which would support exploratory search, and refining searches to time periods or events containing two entities [20]. Recall data incorporation for better ranking of results and the related entity suggestion component can be improved by considering entity relationship evolution over time.

6 Discussion

The closest implementation to our aim is the Internet Archive's Wayback Machine [20]. The Wayback Machine, however, does not include a range of functionality that is needed to classify it as a digital library [4]. Notably, digital libraries require a browse and search functionality. Where the Wayback Machine only supports the direct retrieval of a website [20]. Kanhabua et al.'s implementation does however propose a means of providing a ranked search engine, however, it is not optimized due to the reliance of search engines on query logs. It is also not possible to formulate complex queries. Providing at best a primitive search functionality.

The three architectures compared in table 1, all have open-source licencing. This provides the advantage that they are not only customizable and secure, but also aids widespread adoption, which helps with the integration and standardization of multiple digital libraries [14]. While all three architectures can use relational databases, Greenstone is able to use a flat file system. The flat file system could be beneficial to ensure long term support [15]. Flat formats are easy to duplicate, store and view using basic OS tools. Where repositories binary encode digital entities and accompanying metadata [8]. Functionality could be built to any extent in the future if needed. Instead of relying on possibly outdated middleware or formats with dependencies that are no longer supported, which can inhibit access to an archive. [2]. Essentially, a tool would draw all data from an archive, pre-process the data and save the files to storage. Making the tool and original archive middleware unnecessary in accessing the data in the future if needed. Simple DL implementations, prove that it is possible to preprocess data, store it in a flat format and still provide a search and browse user interface with common tools, such as a web browser [18]. Scalability is however of concern in this regard, as Simple DL's JavaScript web search approach produced acceptable search results, up to one hundred thousand items [17]. The accumulation of multiple archives could quickly surpass this.

The three architectures, by default support the Dublin core metadata schema. Greenstone and DSpace however, are better suited for custom formats [14]. This is advantageous for new types of data, that might be restricted, when classified by current metadata fields or are insufficient for particular institutions. Metadata fields are plagued with inconsistencies and ambiguity. Work on containerizing schemes has also fallen short as a solution, work on PREMIS noted this [11].

Lastly, Greenstone provides an uncommon means to distribution. The ability of a portable distribution, such as on a USB Flash Drive or DVD provides a unique 'air gap' level of security, complementing the digital libraries authenticity [15]. However, complicating a scheduled backup and introducing concerns with software dependency mentioned above.

Table 1: Comparison Table Between Greenstone, DSpace and Fedora

	Greenstone	**DSpace**	**Fedora**
Licence	Open-Source, GNU General Public Licence	BSD Licence	Open-Source
Database / RDBMS	Flat file database engine - GDBM/JDBM or relational database systems - Microsoft SQL	Relational databases - PostgreSQL/Oracle	Relational databases - MySQL, Oracle, PostgreSQL & Microsoft SQL Server
User interface	Two interfaces – Greenstone Librarian Interface for library management. Greenstone user interface, website application user interface.	Two interfaces – JSPUI and XMLUI for user information searching and administration tools.	Website application user interface and administration tool.
Metadata	Default - Dublin core. RFC 1807, NZGLS (New Zealand Government Locator Service), AGLIS (Australian Government Locator Service)	Default - Dublin core. OAI-PMH (Open Archive Initiative Protocol for Metadata Harvesting) compliant. METS and	Default - Dublin core. OAI-PMH (Open Archive Initiative Protocol for Metadata Harvesting) compliant.

	and METS support. New format support with Greenstone's Metadata Set Editor.	PREMIS support. New format support with XML.	METS and PREMIS support. RDF (Resource Description Framework) support.
Distribution	Server or distributable via a USB Flash Drive or DVD.	Server	Server

Versioning is only touched on by a few papers. The well-known Mellon Fedora System particularly uses distinct datastreams and disseminators to portray versioning [12]. The full extent to which versioning functionality is preserved by an archive, will unlikely be able to be ported over to a pre-processed flat file system. This would usually be managed at a middleware layer in an archive [4]. Archive logs could possibly be transferred but would not portray the system at particular points in time, like a snapshot [12]. Authenticity would rely on the fact that the original archive has protocols and strategies in place and that any redundant backup of the archive would safely stored.

7 Summary

Many digital library architectures are available. The Mellon Fedora case study provides an example of how these architectures can be modified to suit current requirements. If resources allow, institutions prefer to build these components in-house because off the shelf implementations cannot provide for their requirements. Most of the bought solutions offered are expensive, require training, and need regular maintenance. They therefore present a risk to low resource institutions as preservation is an ongoing concern. Years of careful well archived work can be lost due to the current circumstances.

Solutions can be born out of common pre-existing tools, a few popular ones, namely: Greenstone, DSpace and Fedora, are conveniently open-source, or provide functionality in order to customize them to suit their institution or use-case. This is still typically only attempted by bigger institutions and organizations, that have the required resources and funding to attempt an implementation. Institutions also try keep their digital libraries as integral as possible, by first implementing well known commonly accepted metadata standards such as Dublin core. They do however customize metadata sets or implement multiple schemes to satisfy internal needs. This often leads to redundant data capture and illogical data placement. This is a downfall of metadata schemes.

Greenstone provides a unique distribution format of digital libraries suited to low resource environments, with limited networks or Internet access. It also, along with SimpleDL, provides an alternative approach to saving data. While most digital libraries use a relational database, Greenstone has the option of, and Simple DL uses a flat file system as a repository. Complementing a low resource environment that might not be able to install additional software, to access data and provide long term access.

This led to a discussion around the possibility of extending Simple DL, like other institutions have - to other architectures in the past, to build a custom implementation that would be able to archive archives. This could be beneficial due to the use of flat files, that are widely supported and accessible for persistence concerns. The possibility of later configuration and use of already existing OS tools.

References

[1] S. Sugimoto, S. Kiryakos and C. Wijesundara, "Metadata Models for Organizing Digital Archives on the Web: Metadata-Centric Projects at Tsukuba and Lessons Learned," Proc. Int'l Conf. on Dublin Core and Metadata Applications, 2018

[2] T. Owens, "The Theory and Craft of Digital Preservation," Johns Hopkins University Press,
2018.

[3] V. Ambati, N. Balakrishnan, R. Reddy, L. Pratha and C. V. Jawahar, "The Digital Library of India Project". DOI:10.5860/rbm.20.2.119.
2006

[4] R. Pandey, "Digital Library Architecture," Indian Statistical Institute. Available: http://dlissu.pbworks.com/w/file/fetch/44829234/B_architecture.pdf.
March 2003

[5] L. Solla, "Building Digital Archives for Scientific Information," Cornell University. DOI: 10.5062/F43X84M3. Available: http://www.istl.org/02-fall/article2.html.
2002

[6] A. Kumar, R. Saigal, R. Chavez and N. Schwertner, "Architecting an Extensible Digital Repository," Proceedings of the 2004 Joint ACM/IEEE Conference on Digital Libraries. DOI: 10.1109/JCDL.2004.239994.
September 2004

[7] D. Krafft, A. Birkland and E. Cramer, "NCore: Architecture and Implementation of a Flexible, Collaborative Digital Library". DOI: 10.1145/1378889.1378943.
April 2008

[8] D. Koutsomitropoulos, A. Tsakou, D. Tsolis and T. Papatheodorou, "TOWARDS THE DEVELOPMENT OF A GENERAL-PURPOSE DIGITAL REPOSITORY," High Performance Information Systems Laboratory, Department of Computer Engineering and Informatics. Available: https://www.scitepress.org/papers/2004/26374/26374.pdf.
January 2004

[9] M. Nelson and H. Van de Sompel, "A 25 Year Retrospective on D-Lib Magazine". Available: https://www.researchgate.net/publication/343903718_A_25_Year_Retrospective_on_D-Lib_Magazine.

2020
[10] R. Kahn and R. Wilensky, "A framework for distributed digital object services," DOI: 10.1007/s00799-005-0128-x.
2006
[11] R. Gartner, "Metadata for digital libraries: state of the art and future directions," JISC. Available: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.183.3762&rep=rep1&type=pdf.
2008
[12] S. Payette and T. Staples, "The Mellon Fedora Project," Digital Library Architecture Meets XML and Web Services. Available: https://arxiv.org/ftp/arxiv/papers/1312/1312.1385.pdf.
2002
[13] D. Gerrard, J. Mooney and D. Thompson, "Digital Preservation at Big Data Scales: Proposing a step-change in preservation system architectures," DOI: 10.1108/LHT-06-2017-0122. Available: https://doi.org/10.1108/LHT-06-2017-0122.
2017
[14] H. Sastry and L. Reddy, "Digital Repository Software Packages: An extended architecture for image handling in open source packages". Available: https://www.researchgate.net/publication/279466681.
2010
[15] I. Witten, R. McNab, S. Boddie and D. Bainbridge, "Greenstone: A Comprehensive Open-Source Digital Library Software System". DOI: DOI:10.1145/336597.336650.
2002
[16] M. Smith, M. Bass, G. McClellan, R. Tansley, M. Barton, M. Branschofsky, D. Stuve and J. Walker, "DSpace: An Open Source Dynamic Digital Repository," D-Lib Magazine. DOI:10.1045/january2003-smith.
2003
[17] H. Suleman, "Reflections on Design Principles for a Digital Repository in a Low Resource Environment," University of Cape Town, South Africa. Available: https://pubs.cs.uct.ac.za/id/eprint/1331/1/ho_2019_lowresource.pdf.
2019
[18] H. Suleman, "Simple DL: A toolkit to create simple digital libraries," University of Cape Town, South Africa, 2021. Available: https://pubs.cs.uct.ac.za/id/eprint/1512/1/paper_88.pdf.
2021
[19] T. Schwarz, M. Baker, S. Bassi, B. Baumgart, W. Flagg, C. van Ingen, K. Joste, M. Manasse and M. Shah, "Disk Failure Investigations at the Internet Archive". Available: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.64.1324&rep=rep1&type=pdf.
2006
[20] N. Kanhabua, P. Kemkes, W. Nejdl, T. N. Nguyen, F. Reis and N. K. Tran, "How to Search the Internet Archive Without Indexing It," Department of Computer Science, Aalborg University, Denmark, L3S Research Center / Leibniz Universitat Hannover, Germany. DOI:10.1007/978-3-319-43997-6_12.
2016
[21] Y. AlNoamany, A. AlSum, M. Weigle and M. Nelson, "Who and What Links to the Internet Archive," Old Dominion University, Department of Computer Science. DOI:10.1007/s00799-014-0111-5.
2014
[22] M. Kuźma and A. Mościcka, "Metadata evaluation criteria in respect to archival maps description: A systematic literature review," DOI: 10.1108/EL-07-2019-0161.
2020