Simple DL: A toolkit to create simple digital libraries

Hussein Suleman[0000−0002−4196−1444]

University of Cape Town, South Africa
hussein@cs.uct.ac.za
http://dl.cs.uct.ac.za/
https://github.com/slumou/simpledl

Abstract. Digital library systems are not always successfully implemented and sustainable in low resource environments, such as in poor countries and in organisations without resources. As a result, some archives with important collections are short-lived while others never materialise. This paper presents a new toolkit for the creation of simple digital libraries, based on a long trajectory of research into architectural styles. It is hoped that this system and approach will lower the barrier for the creation of digital libraries and provide an alternative architecture for experiments and the exploration of new design ideas.

Keywords: simple · low-resource · digital library architecture.

1 Introduction

The earliest examples of digital libraries (e.g., Project Gutenberg [8], arXiv.org) were based on custom software systems developed to meet what was at the time a very specific goal. Over time, however, it was increasingly recognised that the needs of a specific project could be generalised to a larger community. An example of this was when the open-source EPrints software was created to support increasing interest in self-archiving in the research comunity [7]. This shift from custom solutions to general toolkits resulted in a proliferation of digital libraries around the world to meet the needs of various communities. Variations of the same underlying architecture have been used to create repositories based on tools such as DSpace [1], AtoM [2] and Omeka [9]. This does not, however, meet the needs of all archivists.

In 2006, the Digital Bleek and Lloyd Collection was created, based on a custom-developed software system [13]. Given that the project was based in a country with relatively poor Internet connectivity, the collection was packaged onto a DVD-ROM so that it could be distributed as part of a related book and in keeping with the LOCKSS [12] principle that many copies keeps information safe. The system was designed on the basis of an atypical set of principles: that the network could not be assumed; that mediation via a software system should be avoided; and that the pre-processing of data to create static representations was always preferable.

It can be hypothesized that pre-processed collections in digital libraries that are largely offline are better suited to low-resource environments. They require less ongoing technical maintenance as there are fewer things to break. They also require fewer computational resources for access as there is no software middleware layer. Finally, it can be argued that they are more rescuable, as the digital objects are already in a familiar hierarchical file organisation and do not require APIs to extract data.

This paper presents a software toolkit designed according to these principles - the Simple DL toolkit. The toolkit is designed to be as simple as possible, to enable long-term access to digital libraries even when there is no active preservation and when there is computer system or network failure. It is designed for disaster or, if there is no disaster, to enable easy migration to the next generation of solutions.

Some would argue that digital library software systems are a solved problem. The lack of sustainable digital libraries in poor countries, and failures with current systems, suggest that there is still scope for experimental systems that test alternative design ideas. Simple DL is exactly that - an experimental system with a radically different design meant for interrogation by researchers and practitioners.

2 Related Work

Digital library architecture has evolved to encompass both the granular level of individual systems, as well as collections and systems and how these systems are interconnected and made interoperable at national and international levels [14].

DSpace [18] and EPrints [6] remain among the most popular toolkits for creating repositories. Both are Web-based systems, with Web applications and databases as back-end services. In contrast, Greenstone [19] was designed to function both online and offline. Greenstone collections could be distributed on CDROM, but required installation of software in order to access collections.

Some attempts have been made to avoid software installation altogether. OpenDlib [4] and OpenDL [5] were early efforts to define digital library systems as collections of components, thus reducing the problem to component assembly rather than monolithic software installation. Diligent [3] was a generalisation of the component model of digital library systems to arbitrary instantiations on a high performance grid system. In contrast, Lumpa [10] demonstrated that entire instances of DSpace could be managed within a private cloud on-demand. These grid and cloud solutions attempt to make it easier for end-users by the use of sophisticated high performance computing frameworks.

As an alternative, the Digital Bleek and Lloyd [13] was designed to be easy for end users to use by changing the fundamental architecture and removing the need for network access and computation at the time of access. This simplification was still a custom solution, though variations of the idea showed promise for institutional repositories [11] and systems with non-static collections [17].

While many aspects of offline collections have been investigated, no previous attempts have been made to create a reusable toolkit for simple offline digital libraries to support experimentation and explore different models and principles for digital libraries [16].

3 System Design

This paper reports on the initial design of Simple DL, a toolkit for creating simple pre-generated digital libraries.

3.1 Features

The major features of Simple DL (in its 2021 release) are as follows:

- Metadata is stored in spreadsheets and in XML files. The system can support any metadata format but the default configuration is based on either Dublin Core or ICA-AtoM [2].
- There is no database management system and no database. All unstructured data is stored as flat files, and all structured data is stored as XML.
- There is minimal use of Web applications.
- Sites can be generated and then served locally or via a Web server or shared drive. This allows access from a mobile device (phone or tablet) with no Internet connectivity.
- The site's appearance can be customised using standard XSLT and CSS.
- User profiles are stored for users who make contributions online. Entities are also extracted.
- Submitted items, comments and new user registrations can be moderated.
- Search and browse is implemented as a faceted search that is in-browser.

3.2 Storage Layout

Given that Simple DL relies on file-based stores, the storage of data and software is a key part of the design. This default arrangement can also be customised, as the configuration can specify different locations for the different components. The default locations, and their purposes, are as follows:

- **simpledl** is the core software toolkit. This is meant to be stable across systems and configurations. It contains bin and template. **bin** is the location for the applications/scripts that are core to Simple DL, while **template** contains a template for a new collection's website.
- **public_html** is the self-contained offline website that can be served to users through a Web server or opened directly in a browser. Its contents look like a typical website and the figure does not indicate standard directories for styles, thumbnails, etc. Some directories are, however, specific to Simple DL. **metadata** contains the metadata, in both XML and HTML format, for all items in the collection. **collection** contains all the digital objects. **indices** contains the XML indices for the faceted search. Finally, **cgi-bin** is the default space for Web applications.

- **data** stores all data needed to configure the system and generate the website. **config** stores the core configuration information as well as the XSLT templates to transform XML to HTML pages. **website** is a supplementary template for a collection's website, in order to override and add onto the default template. **comments** and **uploads** store all contributions from users, whose profiles are stored in **users**. Finally, **spreadsheets** stores all spreadsheets containing metadata for the collection. In principle, this **data** directory completely defines the configuration and all data for the system such that its website can be regenerated from scratch, if given only the digital objects in the **collection** directory.
- **db** is a temporary store for working and cached versions of files. **entities**, **comments** and **fulltext** are all caches to speed up the processing. **counter** keep track of identifiers. **moderation** is a set of directories where submissions are stored temporarily before/after being moderated; when a comment or item is accepted, it is added to the relevant **data** directory.

3.3 Import, Index and Generate

The main operations of Simple DL centre around ingesting a metadata collection and creating a website representation of the collection. This is done using 3 steps that corresponds to applications/scripts in the system. This process is similar to that used in Greenstone [19] but is different in that the target is: in the first instance, a file-based store; and in the second instance, a static website.

Step 1 - import. Metadata is read in from source spreadsheets and source XML files and individual target XML files are created for metadata. A hierarchy of directories is constructed to correspond to the original structure of the source directories, and according to nesting rules defined in the AtoM metadata entries. Entities are extracted if necessary and used to generate user profiles automatically, also as XML files.

Step 2 - index. All XML metadata and user files are indexed for an information retrieval engine that supports faceted search. Fulltext is extracted from PDF files as needed (and cached).

Step 3 - generate. All XML files (metadata, users, website pages, etc.) are converted to HTML by applying the XSLT stylesheet. In addition, the template website is copied over and thumbnails are created for granular objects and subcollections.

Each time a new metadata sub-collection is added to the system, these 3 steps need to be invoked. The applications have parameters to control which processing occurs, for greater efficiency. They will also do automatic dependency management, so if a spreadsheet has not changed it will not be imported again and if an HTML file is up-to-date, it will not be regenerated from its XML source.

3.4 Web management interface

While all the core applications are meant to be used off-line in the first instance, some archivists may prefer not to use a command-line to interact with

the system. As such, there is a rudimentary Web interface that serves mostly as a front-end to the command-line applications.

An administrator is able to log into the system using Google credentials for authentication and authorisation handled by a specification of authorised administrators in the configuration.

Administrators are also able to manage the files through a Web interface and authorise new items, comments and requests for user accounts. Figure 1 shows the default administrator interface.

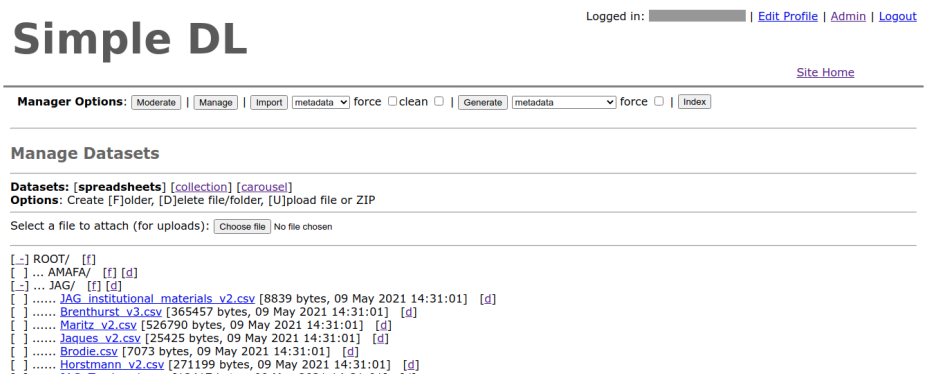

Fig. 1. Simple DL Web administrator interface

3.5 Users and entities

The system has 2 types of user profiles.

Automatically-generated profiles are extracted from the metadata for named entities in defined fields (creator in Dublin Core, eventActor in ICA-AtoM). These are then linked to all items where the entity has been mentioned.

Users can also request permission to make contributions to the digital library (if it is online). Once this is approved by an administrator, a user profile is created and handled similarly to the automatically-generated profiles. The key difference is that a contributor can log into the system while this is not possible for an extracted entity. One open philosophical question is how to link these, or if linking of these should even be allowed.

3.6 Comments and Submissions

Contributors are able to add comments to any metadata item (which are posted after approval by an administrator). As part of a comment, it is possible to attach a new digital object. If approved, this then becomes a part of the collection in its own right, and it can be commented on as well, etc.

New digital objects can also be uploaded as standalone items, without a link to an existing item. These too, must first be approved by an administrator.

The configuration of the system defines: where new items will be placed; and what metadata will be required of new submissions and comments.

3.7 Information Retrieval System

The Information Retrieval subsystem comprises 3 key components: an offline indexer; a Javascript query engine; and a Javascript user interface. Together these implement a classical tf.idf search system, with faceted searching and multiple indices for different data subsets.

The performance of this faceted search system was tested extensively in prior work [15], where it was demonstrated that sub-seconds responses were possible in typical cases for up to 100000 items, which is sufficient for many smaller archives. Given that the goal of this project is not scalability but support for smaller low-resource archives, this was deemed more than sufficient.

4 Case Studies

Three case studies are presented in the next section to illustrate how Simple DL is being applied to different scenarios, where all the projects have a common need for solutions that do not require large amounts of resources for sustainability.

4.1 Emandulo

Emandulo is a project from the Archives and Public Culture (APC) Initiative to gather digital material from related to pre-colonial Southern African history and organise and assemble this as a tool for researchers. Thus, there are sub-collections from different institutions, each with its own system of hierarchical organisation.

APC staff assembled all the metadata in spreadsheets, and painstakingly edited this to recontextualise items within the various collections. Given the focus on archives, and archival culture, metadata was considered most important and was therefore highlighted throughout the site.

User contributions are expected and there is a strong requirement for entities and entity management through authority files.

Many additional features were desired within the website. A professional design team was hired to design the look and feel. A carousel was used on the front page and on item pages to show multiple items with different views. Some tables of items on the website (such as contributions made by one contributor) are sortable by columns - this was implemented in Javascript.

Simple DL was effectively used as a replacement for AtoM, which was the previous system used by the project. Figure 2 shows a typical item listing with metadata and composite item thumbnails.

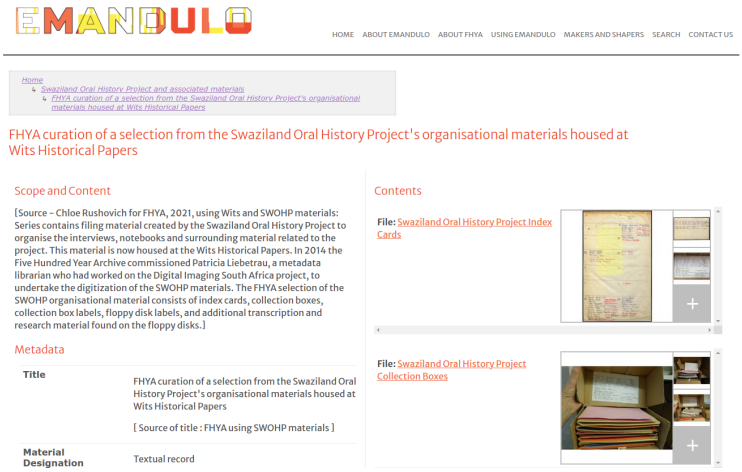

Fig. 2. Emandulo item listing

4.2 Digital Bleek and Lloyd

The Digital Bleek and Lloyd is a project of the Centre for Curating the Archive, to digitise and make available the Bleek and Lloyd Collection of books, drawings, and other historical documents on the language, culture and history of the |Xam and !kun speakers and other early South Africans.

The emphasis on this project was to a larger degree on the visual rather than the metadata, as the visual elements (such as pages of books and annotated drawings) contained most of the information desired by researchers. Also, the older form of text in the books is arguably not representable in Unicode and not understood by any living person so the original form is needed for ongoing study.

Figure 3 shows the faceted search interface used to drill down to images on a particular topic created by a particular contributor.

4.3 NDLTD Document Archive

The Networked Digital Library of Theses and Dissertations (NDLTD) hosts a series of annual symposia around the world. The papers and presentations serve to document the evolution of the community over a long period of time.

Simple DL was therefore configured to serve the content in a minimal manner, giving access to metadata and digital objects for symposia as sub-collections. The metadata was originally in spreadsheets (for ingest into other systems) so could easily be re-purposed for Simple DL. The advantage is that this can be copied and archived offline and will never fail as long as the current variation of HTML is supported.

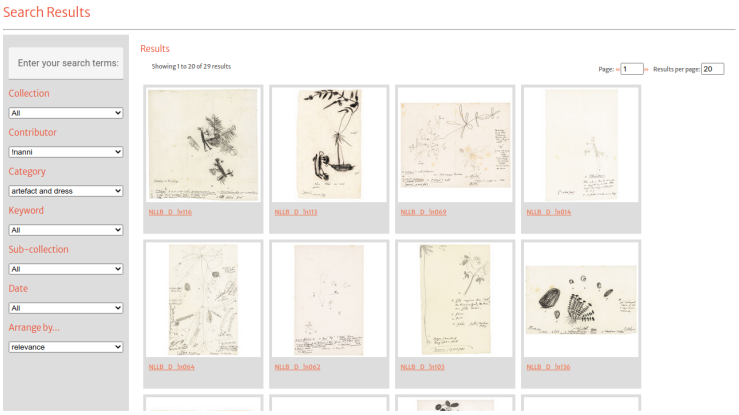

Fig. 3. Bleek and Lloyd faceted search

5 Reflections

Not all archivists need the functionality of popular digital library toolkits. Arguably some archivists need sustainability and the ability to recover from disaster as a paramount requirement. Archivists in low resources environments need systems that will work without much computational power and without much maintenance.

Simple DL has been proposed in this paper as an exemplar of an alternative model of digital library system to meet these objectives. One size does not fit all. This is a solution for those who do not need to store millions of digital objects, but who need a system that is as simple as possible. This is a solution for when the network fails or the operating system upgrades fail, that allows entire digital libraries to be copied as easily as individual items.

Ongoing developments with the toolkit include: making it easier to install and test; improving the performance and stability of various scripts; authoring of complex objects; and archiving of entire digital libraries at a higher level. Ultimately, by keeping the core technology simple, it may even be possible to do a lot more.

Acknowledgements

This research was partially funded by the National Research Foundation of South Africa (Grant numbers: 105862, 119121 and 129253) and University of Cape Town. The authors acknowledge that opinions, findings and conclusions or recommendations expressed in this publication are that of the authors, and that the NRF accepts no liability whatsoever in this regard.

References

1. Bollini, A., Cortese, C., Groppo, E., Mornati, S.: Extending dspace to fulfil the requirements of digital libraries for cultural heritage management. In: IRCDL 2017 Conference. Modena: IRCDL (2017)
2. Bushey, J.: International council on archives (ica)âĂIJaccess to memoryâĂİ(atom): Open-source software for archival description. Archivi & Computer **1** (2012)
3. Candela, L., Castelli, D., Pagano, P., Simi, M.: The evolution of digital library systems: from opendlib to diligent. In: Post-proceedings of the First Italian Research Conference on Digital Library Management Systems (IRCDL 2005). p. 23 (2005)
4. Castelli, D., Pagano, P.: Opendlib: A digital library service system. In: International Conference on Theory and Practice of Digital Libraries. pp. 292–308. Springer (2002)
5. Fox, E.A., Suleman, H., Luo, M.: Building digital libraries made easy: Toward open digital libraries. In: International Conference on Asian Digital Libraries. pp. 14–24. Springer (2002)
6. Gutteridge, C.: Gnu eprints 2 overview (2002)
7. Harnad, S.: The self-archiving initiative. Nature **410**(6832), 1024–1025 (2001)
8. Hart, M.: The history and philosophy of project gutenberg. Project Gutenberg **3**, 1–11 (1992)
9. Kucsma, J., Reiss, K., Sidman, A.: Using omeka to build digital collections: The metro case study. D-Lib magazine **16**(3/4), 1–11 (2010)
10. Lumpa, M., Suleman, H.: Investigating the feasibility of digital repositories in private clouds. In: International Conference on Asian Digital Libraries. pp. 151–164. Springer (2019)
11. Phiri, L., Williams, K., Robinson, M., Hammar, S., Suleman, H.: Bonolo: A general digital library system for file-based collections. In: International Conference on Asian Digital Libraries. pp. 49–58. Springer (2012)
12. Reich, V., Rosenthal, D.S.: Lockss: A permanent web publishing and access system. D-Lib Magazine **7**(6), 1082–9873 (2001)
13. Suleman, H.: Digital libraries without databases: The bleek and lloyd collection. In: International Conference on Theory and Practice of Digital Libraries. pp. 392–403. Springer (2007)
14. Suleman, H.: Design and architecture of digital libraries. In: Digital Libraries and Information Access: Research Perspectives. Facet Publishing (2012)
15. Suleman, H.: Investigating the effectiveness of client-side search/browse without a network connection. In: International Conference on Asian Digital Libraries. pp. 227–238. Springer (2019)
16. Suleman, H.: Reflections on design principles for a digital repository in a low resource environment. In: Proceedings of HistoInformatics Workshop 2019. CEUR (2019), http://pubs.cs.uct.ac.za/id/eprint/1331/
17. Suleman, H., Bowes, M., Hirst, M., Subrun, S.: Hybrid online-offline digital collections. In: Proceedings of the 2010 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists. pp. 421–425. SAICSIT '10, ACM, New York, NY, USA (2010). https://doi.org/10.1145/1899503.1899558, http://doi.acm.org/10.1145/1899503.1899558
18. Tansley, R., Bass, M., Stuve, D., Branschofsky, M., Chudnov, D., McClellan, G., Smith, M.: The dspace institutional digital repository system: current functionality. In: 2003 Joint Conference on Digital Libraries, 2003. Proceedings. pp. 87–97. IEEE (2003)

19. Witten, I.H., Boddie, S.J., Bainbridge, D., McNab, R.J.: Greenstone: a comprehensive open-source digital library software system. In: Proceedings of the fifth ACM conference on Digital libraries. pp. 113–121 (2000)