

Open Source Web Content Management Technologies for Libraries

Dr. M.G. Sreekumar
[UNESCO Coordinator, Greenstone Support, South Asia]
Librarian & Head
Center for Development of Digital Libraries (CDDL)
Indian Institute of Management Kozhikode

The universe (which others call the Web) is the place where society keeps the sum total of human knowledge. It's where we learn and play, shop and do business, keep up with old friends and meet new ones ...

Today we stand at the epicenter of a revolution in how society creates, organizes, locates, presents, and preserves information ...

.... It's all the Web

Ian Witten et al. in "WebDragons"

Abstract

As the Web gets into the centre stage of almost the entire gamut of our activities such as publishing, preservation, access and dissemination of information across the world, it has become imperative as well as a survival issue for library and information professionals to master the Webskills. Thanks to the plethora of open source technologies, in the recent past, they have been substantially enabling and empowering the libraries and information centers for information management. Open source technologies have been gaining increasing attention as well as academic interest owing to the growing demand for improved information and knowledge management solutions in universities, institutions as well as enterprises. Advancements in technology in recent years and its resultant diffusion into society are just the natural offshoots which propelled this movement. A range of Web based solutions are now available in the open source software (OSS) front: from integrated library management systems (ILMS) to library portals, digital libraries, institutional repositories, open archives harvesting, e-learning, content management, knowledge management, open URLs, social softwares for blogs, wikis, RSS, as well as federated searching, and the list goes on and on. In short, open source technologies for improved information management provide for structured storage environments of digital data with a consistent format for index and content abstraction. They do enable the seamless integration of the scholarly electronic information, help in creating and maintaining local digital content, and strengthen the mechanisms and the capacity of the library's information systems and services. Considering the fact that currently the penetration of electronic content into our

libraries and information centers is an unprecedented 70-80 percentage, it unequivocally prompts us to leverage on the latest digital technologies towards building practical digital libraries and in setting up dynamic electronic information systems deploying OSS applications. Without proper understanding of the complexities, procedures and practices involved in content building, content management, collection building, metadata harvesting as well as collaboration, and the long term preservation strategies, these individual applications will remain islands of structure in an unstructured Internet sea. Developing world standard Web based content management applications demand deployment of costly software solutions which are obviously beyond the reach of most of the universities/institutions and this is more so in the Asian and the African region. This tutorial therefore aims at portraying the unlimited potential of a select set of Web based applications for libraries and information centers in a real world perspective by trying out open source solutions. It attempts to unleash and demystify the plethora of features and functionalities of the open source softwares. Special focus is however given to two of the most important applications such as digital libraries and institutional repositories.

1.0. The Web Technology

Recent research and developments in Web technologies offer a sea of opportunities for the library fraternity. Yet the domain of Web based applications today face the unprecedented challenge of managing an array of content spread across a host of publication types comprising a vast multitude of media and in a rapidly proliferating mix of digital formats. Of course technology offer the way forward. **HTML, DHTML, XHTML, XML, CSS, XSLT, Multimedia, MySQL, PostgreSQL, PHP, RDF, DOI, OpenURL, Dublin Core and other metadata standards, Web 2.0...**, just a few to mention them..., all go together to revolutionise the movement. For the benefit of the participants, especially for those who are new to Web technologies, the first 30 minutes will be spent on introducing these technologies, based on online tutorials <http://www.w3schools.com> and <http://xrl.us/6321>, such as :

1. Concept and overview of the Web
2. Client-side technologies
3. Server-side technologies
4. Managing data with SQL
5. Resource discovery and Interoperability

1.1. Open library technologies

Computers and computer networks are being used in libraries for almost three to four decades now. Libraries all over the world are in the constant business of providing their clientele nascent as well as legacy information in an unprecedented array of content categories or publication types, and in a rapidly proliferating mix of formats (digital as well as print). In the current practical library setting there is an amazing penetration of

digital information through a variety of publication forms such as books (published as such or issued as accompaniment), journals, portals, vortals, reports, CBTs, WBTs, cases, databases etc. Undoubtedly it is essential to have a robust and flexible digital collections management and presentation software for creating the numerous indexes the libraries need, and for creating and delivering digital collections. Creation of indexes and the preservation of digital objects are currently intimately tied to software that presents those objects. [Borgman, 1996].

Commercial library software products are indeed capable, comprehensive and extensible enough to support the above requirements, but in many cases they are beyond the reach of most of the libraries in developing countries. The whole lot of associated issues include initial purchase fee, licensing fee, upgrade fee, annual maintenance contracts (AMCs) and so on. The best available choice for the librarian now is to turn to an Open Source Software (OSS). OSS has grown tremendously in scope and popularity over the last several years, and is now in widespread use. The growth of OSS has gained the attention of research librarians and created new opportunities for libraries [Frumkin, 2002]. OSS is close to our hearts primarily for their free (or almost free) availability and the broad rights it awards to the consumer. According to Stallman and others at OSS, 'Free Software' uses the 'free' from 'freedom' or from 'free speech', and not the one from 'free beer' [http://www.opensource.org/docs/definition_plain.html].

1.2. Open Source Software (OSS)

OSS is both a philosophy and a process. As a philosophy it describes the intended use of software and methods for its distribution. OSS is also a process for the creation and maintenance of software [Morgan]. "OSS is software for which the source code is available to the end-user. The source code can be modified by the end-user. The licensing conditions are intended to facilitate continued re-use and wide availability of the software in both commercial and non-commercial contexts. The cost of acquisition to the end-user is often minimal. According to the proponents of OSS, 'Open source is a development methodology; free software is a social movement'. There are number of other notable features to OSS. Firstly, it has no secrets and the innards are available for anyone to inspect. It is not privately controlled and hence likely to promote open rather than proprietary formats. It is typically maintained by communities rather than corporations and hence bug fixes and enhancement are often frequent and free. It is usually distributed free of charge (developers make their money from support, training, and specialist add-ons; not marketing). It is also essential to clear up some of the misunderstandings about OSS. Open source software may or may not cost money. The cost of ownership often bears little relation to the cost of acquiring a piece of software. 'Public domain' is something different. Open source software has a copyright holder and conditions of legal use. Open source software does not mandate exclusivity. One can use open source programs under Windows. Also one should not choose software solely on the basis of open source. Interoperability and open standards for data are equally important" [OSS Watch, 2005].

1.3. OSS and Libraries

According to open source systems for libraries [OSS4Lib], open source systems could improve library services in many ways. "First, when they are licensed in the typical "general license" manner, cost nothing (or next to nothing) to use--whether they have one or one thousand users. Rather than spending thousands on systems, such funds might be reallocated for training, hiring, or support needs, areas where libraries tend toward chronic shortfalls.

Second, open source product support is not locked in to a single vendor. The community of developers for a particular open source product tends to be a powerful support structure for Linux and other products because of the pride in ownership described above. Also, anyone can go into business to provide support for software for which the very source code is freely available. Thus even if a library buys an open source system from one vendor, it might choose down the road to buy technical support from another company--or to arrange for technical support from a third-party at the time of purchase. On top of this flexibility, any library with technical staff capable of understanding source code might find that its own staff might provide better internal support because the staff could have a better understanding of how the systems work.

Third, the entire library community might share the responsibility of solving information systems accessibility issues. Few systems vendors make a profit by focusing their products on serving the needs of users who cannot operate in the windows/icons/menus/pointer world. If developers building systems for the vision impaired and other user groups requiring alternative access environments were to cooperate on creating a shared base of user interfaces, these shared solutions might be freely built into systems around the world far more rapidly and successfully than ever before".

The principles of OSS are very similar to the principles of librarianship. As you will see, there are many shared principles between OSS and librarianship, especially the free and equal access to information [Morgan]. Anybody who works with computers on a daily basis can contribute to OSS because things like information architecture, usability testing, documentation, and staffing are key skills required for successful projects, and these skills are inherent in the people who use computers as a primary tool in their work. The implementation of OSS in libraries represents a method for improving library services and collections. Let's take advantage of these principles and use them to take more control of over our computing environments [Morgan]. According to Altman, for the library fraternity there are other set of reasons too for preferring OSS over commercial software. Long term preservation, assurance of privacy, provision for auditing, facilitating community resources, and conformity to open standards are hallmarks of OSS. Since commercial software is usually distributed only as a binary that will run only on a single hardware platform (and often only under a single version of a particular operating system) commercial software is very difficult to preserve over the long run without developing hardware emulation (and possibly OS 'emulation', as well). OSS, in contrast, can often be recompiled, or at least ported, to new hardware and operating systems [Altman, 2001]. In order to get a picture about the availability of OSS for digital

library applications, it is encouraged to visit the directories of OSS projects, such as GNU [<http://www.gnu.org/>] and Sourceforge [<http://www.sourceforge.net/>] open source directory which lists over 230,000 projects with more than 2 million registered users, and the numbers continue to grow.

Libraries need software solutions for a range of activities in varying degrees of intensity and complexity depending on context. If we can breakdown and categorize them based on the requirements, feature sets, functionalities, utilities and services that we look for each of our needs, we could have a broad classification as given below:

- Integrated Library Management System (ILMS)
- Content Management / Portals
- Electronic (Online) Resources Management
- Digital Libraries
- Open Access Archives / Institutional Repositories
- Open Archives Harvesting
- Federated Searching
- Z39.50 Search/Retrieval
- E-learning
- Open URLs
- Social Computing/Softwares for Blogs, Tags, Wikis, RSS, Feed Aggregation etc.

There are many open source solutions in place for almost all of the above applications and in fact the library community at times feel embarrassed and utterly confused seeing the multitude and features list of many of them. However, the choice of a software for a particular application is crucial as well as critical for the reason that the application has to serve for a long term.

Some of the significant and important applications include, but not limited to, the following. These are primarily based on the experience and user feed back seen from literature and e-lists subscribed to by librarians/information professionals and hence cannot be claimed as comprehensive.

General Platforms/Applications

i. Operating Systems

Linux, Free / Open BSD, Open Solaris...

ii. Web Servers

Apache

Lots in Java! see at...

<http://java-source.net/open-source/web-servers>

iii. Web Server-side Scripting

- PHP
 - Architecture
 - Linux, Apache, MySQL, PHP (LAMP)
 - Windows, Apache, MySQL, PHP (WAMP),
eg. xampp (<http://www.apachefriends.org/en/xampp.html>)

iv. Web Services

- **Apache Tomcat Web Container/Service**
- **Apache Cocoon Content Framework/Service**
- **Apache Ant Build Tool**

v. Programming Languages

Perl, PHP, Python...

vi. Database Management Systems

MySQL, PostgreSQL, mSQL ...

vii. Applications

Apache, Tomcat, emacs, grep, MySQL, sendmail, ssh

viii. Image processing

ImageMagick, tiffinfo/tiffdump

ix. Server Log Analysis

- Webalizer
 - <http://www.webalizer.org/>

Library Specific Platforms/Applications

i. Integrated Library Management Systems (ILMS)

- **KOHA**
 - <http://www.koha.org/>
- **Evergreen**
 - <http://wiki.code4lib.org/index.php/Evergreen>
- **Emilda**
 - <http://wiki.code4lib.org/index.php/Emilda>
- **OpenBiblio**
 - <http://wiki.code4lib.org/index.php/OpenBiblio>
- **phpMyLibrary**
 - <http://wiki.code4lib.org/index.php/PhpMyLibrary>
- **NewGenLib**
 - <http://www.verussolutions.biz/>

ii. Z39.50 Protocol for online search/retrieval (<http://www.loc.gov/z3950/>)

- YAZ Z39.50 Client
 - <http://indexdata.com/yaz/>
- 'Mercury' Z39.50 Client
 - <http://www.basedowinfosys.com/projects/mzc>

iii. MARC Parsers / Editors / Tools

- MarcEdit <http://oregonstate.edu/~reaset/marcedit/html/index.php>
- MARC.pm (Perl), MARC4J (Java)

iv. Library Oriented Search Engines

- Cheshire (<http://cheshire.berkeley.edu/>)
- Pears
- dbWiz (<http://researcher.sfu.ca/dbwiz>)...

v. Portals

MyLibrary, Wordpress

vi. OAI service providers and data providers

PHP OAI Data Provider

vii. Database Management Systems

CDS/ISIS, Win/ISIS, GenISIS etc.

viii. Serials Manager

- **CUFTS**
 - <http://researcher.sfu.ca/cufts>

ix. Citation Manager (from PKP, Simon Fraser University, Canada)

- Bibliographic Management (<http://researcher.sfu.ca/cm>)

x. Link Resolving

- **GODOT** - Electronic (Online) Resources Management
 - <http://researcher.sfu.ca/godot>

xi. OJS (Open Journal Publishing)

- <http://pkp.sfu.ca/ojs>

xii. OCS (Open Conference workflow automation)

- <http://pkp.sfu.ca/ocs>

xiii. Open URL 1.0

- <http://www.oclc.org/research/software/openurl/default.htm>

xiv. Open Digital Libraries

- Greenstone
- DSpace
- Eprints
- FEDORA etc.

xv. Open Access Archives / Institutional Repositories

- DSpace
- Eprints
- FEDORA

- CDSWare
- Greenstone etc.

xvi. Open Archives Harvester

- **Harvester**
 - <http://pkp.sfu.ca/harvester>

xvii. Learning Management Systems (LMS)

- Moodle
- Manhattan etc.

xviii. Content Management Systems (CMS)

- Joomla
- Drupal
- MediaWiki

xix. XML Tools and Systems

- **Utilities**
 - **Xalan, Xerces, libxml, libxslt, saxon**
- **Editors**
 - **emacs / nxml-mode**
- **Database / Search Engines**
 - **Apache Xindice**
 - **Berkeley DB XML**
 - **eXist**
- **Publishing/Web Application Frameworks**
 - **AxKit**
 - **Cocoon**

This tutorial intends to focus on some of the most popular library requirements such as integrated library automation, digital library, open access archives (institutional repositories), open archives harvesting etc., and in particular, highlights the major features and functionalities of KOHA, Greenstone, Dspace and the PKP OAI Harvester.

2.0. Integrated Library Automation Systems (ILS)

Library Automation is believed to be the foundation to all steps in the modernization of the library and information systems. A well planned and executed library automation process can make a great deal of difference in the efficiency of the systems as well as image among the users as well as the management.

An **Integrated Library System**, or ILS, is an enterprise resource planning system for a library, used to track items owned, orders made, bills paid, and patrons who have borrowed. An ILS is usually comprised of a relational database, software to act on that database, and two graphical user interfaces (one for patrons, one for staff). Most ILSes separate software functions into discrete programs called modules, which are then integrated into a unified interface.

Examples of modules include:

1. Acquisitions (ordering, receiving, and invoicing materials),
2. Cataloging (classifying and indexing materials),
3. Circulation (lending materials to patrons and receiving them back),
4. Serials (tracking magazine and newspaper holdings), and the
5. OPAC (public search/retrieval interface for users). Each patron and item has a unique ID in the database that allows the ILS to track its activity.

Larger libraries use ILSes to order and acquire, receive and invoice, catalog, circulate, track and shelf materials. Smaller libraries, such as private homes or non-profit organizations (e.g. churches and synagogues), often forego the expense and maintenance required to run an ILS, and instead use a library computer system. Most sizable First World libraries use an ILS. ILSes are sometimes referred to as Library Management Systems (LMS) or Integrated Library Management Systems (ILMS).

There are a number of open source ILSes available for libraries to make use of. Among them, **KOHA** (<http://www.koha.org>) is an ILMS fast being accepted by most of the leading libraries worldwide.

Features of Koha

1. Koha is an open-source Integrated Library System (ILS).
2. It supports global standards including MARC 21 bibliographic format and
3. Z 39.50 information retrieval protocol.

4. Web-centric architecture (no additional software/utility is required at the client side).
5. Provides tremendous freedom for customization.
6. All the modules of LMS including Acquisition, Cataloguing, Circulation, OPAC, Membership Management, System Administration, Serial Control, etc is available.
7. Web based OPAC system (allows the public to search the catalogue in the library and at home).
8. The software is UNICODE compliant. The creation and retrieval of Indic script based documents is possible.
9. Export/Import and backup/restoration facilities are available.
10. Includes features of fourth generation Library Management Software (LMS).
11. Runs on Linux, Unix, Windows and MacOS platform.
12. Koha uses Apache Web server, MySQL as backend RDBMS and Perl (for server-side scripting). All these softwares are open source.

Koha's site gives the installation manuals for different OSs. A comprehensive manual on Koha 3.01.x is available at <https://sites.google.com/a/liblime.com/koha-manual/>. For Windows it is available at http://www.koha.rwjr.com/Koha_on_Windows.html.

3.0. Digital Libraries

Digital Libraries (DL) are now emerging as a crucial component of global information infrastructure, adopting the latest information and communication technology. Digital Libraries are networked collections of digital texts, documents, images, sounds, data, software, and many more that are the core of today's Internet and tomorrow's universally accessible digital repositories of all human knowledge. According to the Digital Library Federation (DLF, USA - <http://www.dlf.org>), "Digital libraries are organizations that provide the resources, including the specialized staff, to select, structure, offer intellectual access to, interpret, distribute, preserve the integrity of, and ensure the persistence over time of collections of digital works so that they are readily and economically available for use by a defined community or set of communities".

Currently in vast majority of instances, the concept 'Digital Library' is being practiced by and large loosely or even confused by many information systems. It is therefore imperative that the concept is properly understood so that there is no ambiguity while we progress with the work of designing or developing a digital library which is fully justified in the technical sense of the word. It is important that embarking on a digital library project is something which will take away substantial amount of time, energy, manpower and of course the hard earned money being pumped into it – be it for system development or towards development and maintenance of the collection, in a meaningful way. There is consensus all over that there exists a very large quantum of digital information, scholarly as well as trade, which are scattered and distributed throughout the Net and also being stored in numerous other databases and repositories spread across the world. Also there is an unprecedented technology support and availability of infrastructure for digital libraries.

3.1. DL Features

Digital libraries offer new levels of access to broader audiences of users and new opportunities for library and information science field to advance both theory and practice [Marchionini, 1998]. They contain information collections predominantly in digital or electronic form. Electronic publications have some special management requirements as compared to printed document. They include infrastructure, acceptability, access restrictions, readability, standardization, authentication, preservation, copyright, user interface etc.

Digital libraries do enable the seamless integration of the scholarly electronic information, help in creating and maintaining local digital content, and strengthen the mechanisms and capacity of the library's information systems and services. They increase the portability, efficiency of access, flexibility, availability and preservation of digital objects. Digital Libraries can help move the nation towards realizing the enormously powerful vision of 'anytime, anywhere' access to the best and the latest of human thought and culture, so that no classroom, individual or a society is isolated from knowledge resources. Digital library brings the library to the user, overcoming all geographical barriers [ICDL, 2004].

3.2. DL Software

Undoubtedly it is essential to have a robust and flexible digital collections management and presentation software for creating and delivering digital collections. The preservation of digital objects is currently intimately tied to software that presents those objects. Complete preservation of complex digital objects, especially, is likely to require preservation of the software needed to use those objects. [Borgman, 1996]. The complexity of the situation is that digital library technologies and contents are not static. Continual evolution and investment are required to maintain the digital library. Commercial digital library products are comprehensive and extensible enough to support this evolution, but in many cases they are beyond the reach of most of the libraries in India. Some of the popular commercial DL software in the Indian libraries are VTLS (<http://www.vtls.com>) from the international market and ACADO (<http://www.transversalnet.com/acado/index.htm>) as an Indian initiative. The latter is definitely less costlier when compared but still striving its best to get a critical mass of users. The whole lot of associated issues include initial purchase fee, licensing fee, upgrade fee, annual maintenance contracts (AMCs) and so on. The best available choice for the librarian now is to turn to an Open Source Software (OSS). OSS has grown tremendously in scope and popularity over the last several years, and is now in widespread use. The growth of OSS has gained the attention of research librarians and created new opportunities for libraries [Frumkin, 2002]. OSS is close to our hearts primarily for their free (or almost free) availability and the broad rights it awards to the consumer.

3.3. DL Objectives and Workflow

The primary objective of a digital library is to enhance the digital collection in a substantial way, by strategically sourcing digital materials, conforming to copyright permissions, in all possible standards/formats so that scalability and flexibility is guaranteed for the future and advanced information services are assured to the user community right from beginning. The digital library should also be able to integrate and aggregate the existing collections and services mentioned above with an outstanding client interface. This implies that the digital library system should also have a strong collection interface capable of embracing almost all the popular digital standards and formats and software platforms, in line with the underlying digital library technologies in vogue. This is crucial in the case of multimedia integration, which is again important as we planned to also host a digital audio and video library as part of the core library collection. Emphasis should also be given to maximise the efficiency and effectiveness of the information access and retrieval capabilities of the system by deploying Resource Description Framework [RDF] supplemented with popular descriptive metadata standards. The Internet also possesses, in addition to its mammoth proprietary information base, an invaluable wealth and a vast collection of public domain information products such as databases, books, journals, theses, technical reports, cases, standards, newsletters etc., scattered and distributed across the world. This treasure should also be explored to its maximum for collection building, based on the source and

quality. Standard workflow patterns are to be identified for the system which include 'content selection', 'content acquisition', 'content publishing', 'content indexing and storage', and 'content accessing and delivery'. The system should also concern about such related issues, viz., preservation, usage monitoring, access management, interoperability, administration and management etc.

It is always desirable to have crosswalks between the digital catalogue of the library (OPAC) and the digital library, as the OPAC in most cases, acts as a stepping stone for effective information discovery in the library. It also facilitates a healthy bridging between the traditional and the digital library. MARC or any of its variant forms is the desired bibliographic standard recommended for the OPAC, for want of interoperability. Dublin Core [DCMI], MODS (Metadata Object Description Schema) or METS (Metadata Encoding and Transmission) are the recommended metadata format for the digital collection, and XML is the desired encoding scheme [XML]. The XML encoding schemas and the related DTDs (Document Type Definition) strengthen the digital library on strong footing and the XSL (Extensible Stylesheet Language) transformations acts as dynamic gateways between the diverse data streams and the HTML front-end.

3.4. Selection of the DL Software

The software selection based on set parameters is an uphill task, as the technology itself was still emerging only. In general, what is desirable is a system that is flexible enough to fit the current digital information system as above and to accommodate future migration. It should be robust in technical architecture as well as the content architecture. The system should address all major digital library related issues such as 'design criteria', 'collection building', 'content organisation', 'access', 'evaluation', 'policy and legal issues' including 'intellectual property rights'. That the system should be in a position to embrace almost all predominant and emerging digital object formats and capable of supporting the standard library technology platforms, should be the major focus. It should provide two important user interfaces: a public user interface for presentation and a metadata creation interface for administration. The system should also provide a powerful search engine and the interface should be easy to navigate and there should be provision for customisation.

There are many digital library softwares available, proprietary as well as open source, and most of them conform to international standards. As mentioned earlier, VTLS and ACADO are the commercial ones available and popular in the Indian market. Some of the popular Open Source Softwares for digital libraries, which are in use internationally, are 'DSpace', 'Dienst', 'Eprints', 'Fedora', 'Greenstone' etc. In line with the subject thrust of this paper, the Greenstone features are discussed in this paper.

3.5. Developing Digital Libraries using Open Source Software

Digital libraries do enable the creation of local content, strengthen the mechanisms and capacity of the library's information systems and services. They increase the portability, efficiency of access, flexibility, availability and preservation of content. A state-of-art

Digital Library shall give a real boost to the library's modernization activities and its endeavours to launch innovative digital information services to the user community. Once the information is made digital, it could be stored, retrieved, shared, copied and transmitted across distances without having to invest any additional expenditure. Value added and pinpointed information at the click of the mouse will become a reality if there is a Library Portal to provide access to the invaluable collection hosted by the Digital Library.

World over there is increasing appreciation of the Open Access movement and the Open Source Software philosophies and for many libraries it is a chosen decision, be it technical or financial reasons, not to go for a proprietary digital library software. One needs to evaluate some of the popular Open Source Software for digital libraries, which are in use internationally. 'Dienst', 'Eprints', 'Fedora', 'Greenstone' etc. are among the candidates for the preferred software. Obviously **Greenstone** outscores the group as a general purpose digital library software from the point of view of a multi-publication type, multi-format, multi-media and a multi-lingual practical digital library [Greenstone]. And once finalized, it could be formally adopted as the software for creating the digital library.

The **Greenstone Digital Library Software (GSDL)** is a top of the line and internationally renowned Open Source Software system for developing digital libraries, promoted by the New Zealand Digital Library project research group at the University of Waikato, led by Dr. Ian H. Witten, and is sponsored by the UNESCO. Greenstone software uses three more additional associated softwares namely, Java Run Time Environment (JRE), ImageMagick and Ghostscript. The software suite is available at the open source directory 'Sourceforge.Net'.

3.6. Greenstone Fact Sheet (www.greenstone.org)

Greenstone is a suite of software for building and distributing digital library collections. It is not a digital library but a tool for building digital libraries. It provides a new way of organizing information and publishing it on the Internet in the form of a fully-searchable, metadata-driven digital library. It has been developed and distributed in cooperation with UNESCO and the Human Info NGO in Belgium. It is open-source, multilingual software, issued under the terms of the GNU General Public License. Its developers received the 2004 IFIP Namur award for "contributions to the awareness of social implications of information technology, and the need for an holistic approach in the use of information technology that takes account of social implications."

There are presently two versions of Greenstone going around, Version 2 and 3, and they are generally represented as Greenstone2 and Greenstone3. The latest in Greenstone2 as on January 2008 is V.2.80 and that of Greenstone3 is V.03. Greenstone2 will be there for some more years, but ultimately Waikato/Greenstone see that Greenstone3 will replace it.

3.7. Technical Features

3.7.1 Platforms. Greenstone runs on all versions of Windows, and Unix, and Mac OS-X. It is very easy to install. For the default Windows installation absolutely no configuration is necessary, and end users routinely install Greenstone on their personal laptops or workstations. Institutional users run it on their main web server, where it interoperates with standard web server software (e.g. Apache).

3.7.2 Interoperability. Greenstone is highly interoperable using contemporary standards. It incorporates a server that can serve any collection over the Open Archives Protocol for Metadata Harvesting (OAI-PMH), and Greenstone can harvest documents over OAI-PMH and include them in a collection. Any collection can be exported to METS (in the Greenstone METS Profile, approved by the METS Editorial Board and published at <http://www.loc.gov/standards/mets/mets-profiles.html>), and Greenstone can ingest documents in METS form. Any collection can be exported to DSpace ready for DSpace's batch import program, and any DSpace collection can be imported into Greenstone.

3.7.3 Interfaces. Greenstone has two separate interactive interfaces, the Reader interface and the Librarian interface. End users access the digital library through the Reader interface, which operates within a web browser. The Librarian interface is a Java-based graphical user interface (also available as an applet) that makes it easy to gather material for a collection (downloading it from the web where necessary), enrich it by adding metadata, design the searching and browsing facilities that the collection will offer the user, and build and serve the collection.

3.7.4 Metadata formats. Users define metadata interactively within the Librarian interface.

These metadata sets are predefined: Dublin Core (qualified and unqualified) , RFC 1807, NZGLS (New Zealand Government Locator Service), AGLS (Australian Government Locator Service). New metadata sets can be defined using Greenstone's Metadata Set Editor. "Plug-ins" are used to ingest externally-prepared metadata in different forms, and plug-ins exist for XML, MARC, CDS/ISIS, ProCite, BibTex, Refer, OAI, DSpace, METS.

3.7.5 Document formats. Greenstone basically supports all popular file formats and media. Plug-ins are also used to ingest documents. For textual documents, there are plug-ins for PDF, PostScript, Word, RTF, HTML, Plain text, Latex, ZIP archives, Excel, PPT, Email (various formats), source code. For multimedia documents, there are plug-ins for Images (any format, including GIF, JIF, JPEG, TIFF), MP3 audio, Ogg Vorbis audio, and a generic plug-in that can be configured for audio formats, MPEG, MIDI, etc.

3.8. User base

3.8.1 Distribution. As with all open source projects, the user base for Greenstone is unknown. Tens of thousands of installations of Greenstone are estimated across world as is evidenced by the increasing volume of messages being exchanged in the various fora,

especially the Greenstone E-lists. It is distributed on SourceForge, a leading distribution centre for open source software.

3.8.2 Greenstone Example Collections: Examples of public Greenstone collections (see <http://www.greenstone.org> for URLs) can be found at:

- Association of Indian Labour Historians, Delhi
- Auburn University, Alabama
- California University at Riverside
- Chicago University Library
- Detroit Public Library
- Gresham College, London
- Hawaiian Electronic Library
- Illinois Wesleyan University
- Indian Institute of Management Kozhikode (IIMK)
- Kyrgyz Republic National Library
- LeHigh University, Pennsylvania
- Mari El Republic, Russia
- National Centre for Science Information, Bangalore, India
- Netherlands Institute for Scientific Information Services

- New York Botanical Garden
- Peking University Digital Library
- Philippine Research Education and Government Information Network
- Secretary of Human Rights of Argentina
- Slavonski Brod Public Library, Slovenia
- State Library of Tasmania
- Stuttgart University of Applied Sciences
- Texas A&M University Center for the Study of Digital Libraries
- University of Illinois
- University of North Carolina Biblio project
- Vietnam National University
- Vimercate Public Library, Milan, Italy
- Washington Research Library Consortium
- Welsh Books Council

One of Greenstone's unique strengths is its multilingual nature. The reader's interface is available in the following languages: Arabic, Armenian, Bengali, Catalan, Croatian, Czech, Chinese (both simplified and traditional), Dutch, English, Farsi, Finnish, French, Galician, Georgian, German, Greek, Hebrew, Hindi, Indonesian, Italian, Japanese, Kannada, Kazakh, Kyrgyz, Latvian, Maori, Mongolian, Portuguese (BR and PT versions), Russian, Serbian, Spanish, Thai, Turkish, Ukrainian, Vietnamese.

The Librarian interface and the full Greenstone documentation (which is extensive) is in: English, French, Spanish, and Russian.

3.9. Training

Training is a bottleneck for widespread adoption of any digital library software. Greenstone's Waikato site <http://www.greenstone.org>; the Greenstone Wiki <http://greenstone.sourceforge.net/wiki/index.php/GreenstoneWiki>, and the Greenstone Support for South Asia <http://greenstonesupport.iimk.ac.in> give many training materials and guidance on the software. It is observed that Greenstone training and workshops are quite common in digital library conferences and seminar all over the world and this itself speaks volumes the importance of Greenstone.

3.10. E-mail support

There are many E-Lists and E-Groups available for Greenstone support. For subscribing to the main Greenstone lists, visit <https://list.scms.waikato.ac.nz/mailman/listinfo/greenstone-users> for User's List (greenstone-users-request@list.scms.waikato.ac.nz) and <https://list.scms.waikato.ac.nz/mailman/listinfo/greenstone-devel> for Developer's list. There is also an E-List for supporting the South Asian Greenstone users greenstonesupport@iimk.ac.in.

3.11. Greenstone : Features

The salient features of Greenstone are basically taken from two of the official publications of the software development team appeared in D-Lib Magazine during the year 2001 [Witten, 2001] and 2003 [Witten, 2003]. Greenstone builds collections using almost popular and standard digital formats such as HTML, XML, Word, Post Script, PDF, RTF, JPG, GIF, JPEG, MPEG etc. and many other formats which include audio as well as video. It is provided with effective full-text searching and metadata-based browsing facilities that are attractive and easy to use. Moreover, they are easily maintained and can be augmented and rebuilt entirely automatically. The system is extensible: software "plug-ins" accommodate different document and metadata types. Greenstone incorporates an interface that makes it easy for people to create their own library collections. Collections may be built and served locally from the user's own web server, or (given appropriate permissions) remotely on a shared digital library host. End users can easily build new collections styled after existing ones from material on the Web or from their local files (or both), and collections can be updated and new ones brought on-line at any time. The Greenstone Librarian Interface (GLI) is a Java based GUI interface for easy collection building. Greenstone software runs on a wide variety of platforms such as Windows, Unix / Linux, Apple Mac etc. and provides full-text mirroring, indexing, searching, browsing and metadata extraction. It incorporates an interface that makes it easy for institutions to create their own library collections. Collections could be built and served locally from the user's own web server, or (given appropriate permissions) remotely on a shared digital library host. The other set of features include OAI plug-in (introduced since the 2.40 version) and DCMI compliance, UNICODE based multi-lingual capabilities and a user-friendly multimedia interfacing [Unicode]. Further more, it has a powerful search engine 'Managing Gigabyte' Plus-Plus ('MG' PP) and metadata-based browsing facilities. A very interesting feature of Greenstone is its exhaustive set of well documented and articulated manuals (<http://www.greenstone.org/cgi-bin/library?e=p-en-docs-utfZz-8&a=p&p=docs>) such as 'Installer's Guide', 'User's Guide', 'Developer's Guide', and 'From Paper to Collection' a document describing the entire process of creating a digital library collection from paper documents. This includes the scanning and OCR process and the use of the "Organizer". There is one more interesting documentation 'Inside Greenstone Collections' which clarifies most of the trickier parts of using Greenstone, especially dealing with configuration file for the collection in question.

The primary objective of any digital library will be to enhance the digital collection in a substantial way, by strategically sourcing digital materials, conforming to copyright permissions, in all possible standards / formats so that scalability and flexibility is guaranteed for the future and advanced information services and are assured to the user community right from beginning. The digital library has to be planned in such a way that it will integrate and aggregate the existing collections and services with an outstanding user interface. Accordingly, necessary strategies are to be adopted towards working out the digital library system. This implies that the digital library system should have a strong collection interface capable of embracing almost all the popular digital standards, digital formats and software platforms, in line with the underlying digital library technologies in vogue. This is crucial in the case of multimedia integration, which is again important as it is planned to host digital audio and video library as part of the core library collection.

3.12. Greenstone Installation

The GNU Public License version Greenstone can be downloaded from 'http://www.greenstone.org' or 'http://sourceforge.net/index.php'. You can download the binaries for Linux or Windows. The associated softwares such as Java Runtime Environment (JRE) and the Imagemagick also to be downloaded. A graphical tool is used for collection building and configurations and customization. This is called the Greenstone Librarian Interface (GLI) and it requires the Java Runtime Environment (JRE). The latest version pertaining to Volume 2 release of Greenstone as on January 2009 is V.2.81.

Click on "gsdl-2.81-win32.exe". The Install Shield Wizard will begin the installation. Accept all the term of license agreement by clicking on <Yes> button. Click on <next> to install GSDL in the default folder, which is C:\program files\greenstone2. Choose the type 'Local Library'. By default, Local Library is highlighted. Set the Admin Password as "admin" (you can later change it). Installation wizard now starts copying the required files from the GSDL folder. Click on the Finish button to finish GSDL installation. To check whether your installation is proper, Click on 'Start→Programs→Greenstone Digital Library→Greenstone Digital Library'. Click on Enter Library in the 'Dialog Box' and Your Browser should display The GSDL Homepage.

The 2.81 version comes with Imagemagick and Ghostscript softwares bundled with it and hence we need not install them separately.

3.13. Collection Building and Configuration

Greenstone used to have three modes for collection building, viz., Command Line, Web Interface and the Greenstone Librarian Interface (GLI). Among these GLI is the one getting more prominence as far as the librarian / information professionals are concerned.

3.14. Greenstone Librarian Interface (GLI)

The GLI (Greenstone Librarian Interface) was introduced recently, progressing with the version 2.4x. Soon the GLI got strengthened as well as popularized, and the Web Interface mode has been withdrawn temporarily, while it could also be reinforced if one wishes so. The GLI based collection building is quite easy and simple a method. Collection developers can activate the GLI software and use the '**Gather**', '**Enrich**', '**Design**', '**Create**', and the '**Format**' panels for making, configuring, customizing and managing collections.

i. The '**Gather**' Panel facilitates putting the relevant files from the 'workspace' to the 'collection building' area. The 'Enrich' Panel explains how metadata is created, edited, assigned and retrieved, and how to use external metadata sources. Help for this is provided in the GLI Interface. The 'Design' Panel facilitates customising your interface, once your files are marked up with metadata. Using the Gather Panel, you can specify the fields that are searchable, allow browsing through the document, facilitate the languages that are supported, and provide the buttons that are to appear on the page. Help for this is provided in the GLI Interface. The Create Panel facilitates creation of your collection.

To build a typical collection, say 'MyTest' collection, first go to 'File' section, select 'New' and then give the collection name as 'MyTest'. Select OK from the panel and then you will get another panel popped up where you will select the appropriate Metadata Set. You may also give the description about the collection here. By default, the system will prompt Dublin Core metadata set. Click on OK button and you will get the collection create panel made ready for accepting the file(s).

The 'Gather' Panel is activated now. From the 'Workspace' provided, identify the document to be put in the collection by locating it in the local folder. Drag and drop the file to the Collection Area using the mouse. The necessary 'plugin' for the creation of the collection is to be tick marked and enabled in the 'Design' panel, which is the next step in the collection building process. If the collection has objects for which 'plugins' are not provided in the default set, a new dialog box for adding the required plugin will appear and it has to be added to the default set.

ii. Go to the '**Enrich**' panel and give necessary values for the Dublin Core element sets.

Manage Metadata Sets - This feature allows you to add, configure and remove the Metadata Sets in your collection and what Elements they contain.

iii. Design Panel

The next step is to give necessary values and arguments for the '**Design**' panel which include [Note: GLI Design Panel's own language is used below i. to x., for want of clarity and to avoid any ambiguity in usage]:

i. *General Options* - In this section, give the e-mail address of the 'collection creator', 'collection maintainer', 'collection title' (will be supplied by the system), collection folder (will be supplied by the system), Image file location for the Collection icon and the Image file location for the Document icon. Click on the Tick mark for making this collection publicly available.

ii. *Document Plugins* - This section facilitates adding, configuring or removing plugins from your collection. To add one, choose it from the combobox and click 'Add Plugin'. To configure or remove one, select it from the list of assigned plugins and then: i) Change its position in the plugin order by clicking on the arrow buttons. (Note: The position of RecPlug and ArcPlug are fixed). ii) Configure it by clicking 'Configure Plugin', iii) Remove it by clicking 'Remove Plugin'. Plugins are configured using a pop-up design area with a scrollable list of arguments. Enable arguments and enter or select values as necessary.

iii. *Search Types* - Defining the search type is an advanced feature, only available when enabled (by checking the 'Enable Advanced Searches' box). Once enabled, further controls for selecting and changing the order of search types become available. See the 'Search Type Selection and Ordering' section of the 'Design' Panel for more information on this.

iv. *Search Indexes* - The required number of searchable indexes the collection must have, is to be selected here. To add a new index, enter a unique name for the index, select material/metadata is to be indexed, and click 'Add Index'. If you wish to add all of the available sources so as to have indexes built on them, then click 'Add All'.

v. *Partition Indexes* - This feature help to refine index creation. This facility is disabled in the GLI mode.

vi. *Cross-Collection Search* - This feature facilitates cross-collection searching, where a single search is performed over several collections, as if all the collections were one. Specify (Tick Mark) the collections to include in a search by clicking on the appropriate collection's name in the list below. The current collection will automatically be included. [Note : If the individual collections do not have the same indexes (including sub collection partitions and language partitions) as each other, cross-collection searching will not work properly. The user will only be able to search using indexes common to all collections].

vii. *Browsing Classifiers* - This feature allows the AtoZ browsing of the collection and by default it takes the 'Dublin Core . Title'. You can more data elements in the AtoZ classify list as deem fit for the collection using this feature.

viii. *Format Features* - The web pages you see when using Greenstone are not pre-stored, but are generated 'on the fly' as they are needed. Format commands are used to change the appearance of these generated pages. Some are switches that control the display of documents or parts of documents; others are more complex and require html

code as an argument. To add a format command, choose it from the 'feature' list. If a True/False option panel appears, select the state by clicking on the appropriate button.

For example, to get the Cover Image displayed in the document while building the collection, go to the 'Choose Features' dropdown box and enable the 'DocumentIMages', i.e., make its value to True.

ix. *Translate Text* - Use this feature to review and assign translations of text fragments in your collection. The translated text will appear in a different box in the browser.

x. *Metadata Sets* - This feature allows you to add, configure and remove the Metadata Sets in your collection and what Elements they contain.

iv. Now go to the 'Create' panel and click on the 'Build Collection'. Greenstone will start creating the collection. You can see the built collection by clicking on the 'Preview Collection'.

Please remember you have to save your collection development process from time to time. It is not mandatory that you need to comply with the entire set of formalities for a building a collection in a single stretch. You can do it in different sessions too. What is important is saving the sessions from time to time. In the GLI mode of collection building, the various panels to be used are illustrated in Figure 1.

v. Format Panel

i. *General* - This section explains how to review and alter the general settings associated with your collection. First, under the "Format" tab, click "General". Here some collection wide metadata can be set or modified, including the title and description entered when starting a new collection. First are the contact email addresses of the collection's creator and maintainer. The following field allows you to change the collection title. The folder that the collection is stored in is shown next, but this cannot be altered. Then comes the icon to show at the top left of the collection's "About" page (in the form of a URL), followed by the icon used in the Greenstone library page to link to the collection. Next is a checkbox that controls whether the collection should be publicly accessible. Finally comes the "Collection Description" text area as described in "Creating a New Collection".

ii. *Search* - This section explains how to set the display text for the drop down lists on the search page. Under the "Format" tab, click "Search". This pane contains a table listing each search index, index level (for MGPP or Lucene collections), and index or language partition. Here you can enter the text to be used for each item in the various drop-down lists on the search page. This pane only allows you to set the text for one language, the current language used by GLI. To translate these names for other languages, use the Translate Text part of the Format view (see "Translate Text" feature in the Format panel).

iii. *Format Features* - The web pages you see when using Greenstone are not pre-stored, but are generated 'on the fly' as they are needed. Format commands are used to

change the appearance of these generated pages. Some are switches that control the display of documents or parts of documents; others are more complex and require html code as an argument. To add a format command, choose it from the 'feature' list. If a True/False option panel appears, select the state by clicking on the appropriate button.

For example, to get the Cover Image displayed in the document while building the collection, go to the 'Choose Features' dropdown box and enable the 'DocumentIMages', i.e., set its value to True.

iv. ***Translate Text*** - Use this feature to review and assign translations of text fragments in your collection. The translated text will appear in a different box in the browser.

v. ***Cross-Collection Search*** - This feature facilitates cross-collection searching, where a single search is performed over several collections, as if all the collections were one. Specify (Tick Mark) the collections to include in a search by clicking on the appropriate collection's name in the list below. The current collection will automatically be included. [Note : If the individual collections do not have the same indexes (including sub collection partitions and language partitions) as each other, cross-collection searching will not work properly. The user will only be able to search using indexes common to all collections].

vi. ***Collection Specific Macros*** - Under the "Format" tab, click "Collection Specific Macros". This view shows the contents of the collection's extra.dm macro file. This is where collection specific macros can be defined. To learn more about macros, see Chapter 3 of the Greenstone Developer's Guide.

3.15. Hierarchy Structure

To create indexes for section and sub-section, the pre-requisite is that the document should be in HTML format. Therefore your collection files in other formats like PDF, Word, etc. are first to be converted into HTML format. Also in the Collection Configuration file (for GLI, in the Design Panel, in the Document Plugin section, while configuring the Arguments in the HTML Plugin, click and enable the 'description_tags'), the HTML plugin has to be modified to 'plugin HTMLPlug -description_tags'. Corresponding changes have to be made in the 'indexes' and the 'collectionmeta' lines. Obviously now the Source File has to be edited as a HTML file structure. For the section and sub sections, you need to edit the source file as follows, giving XML tags as comments in the body of the HTML file. Fig.1 below shows a hierarchy structured EBook.

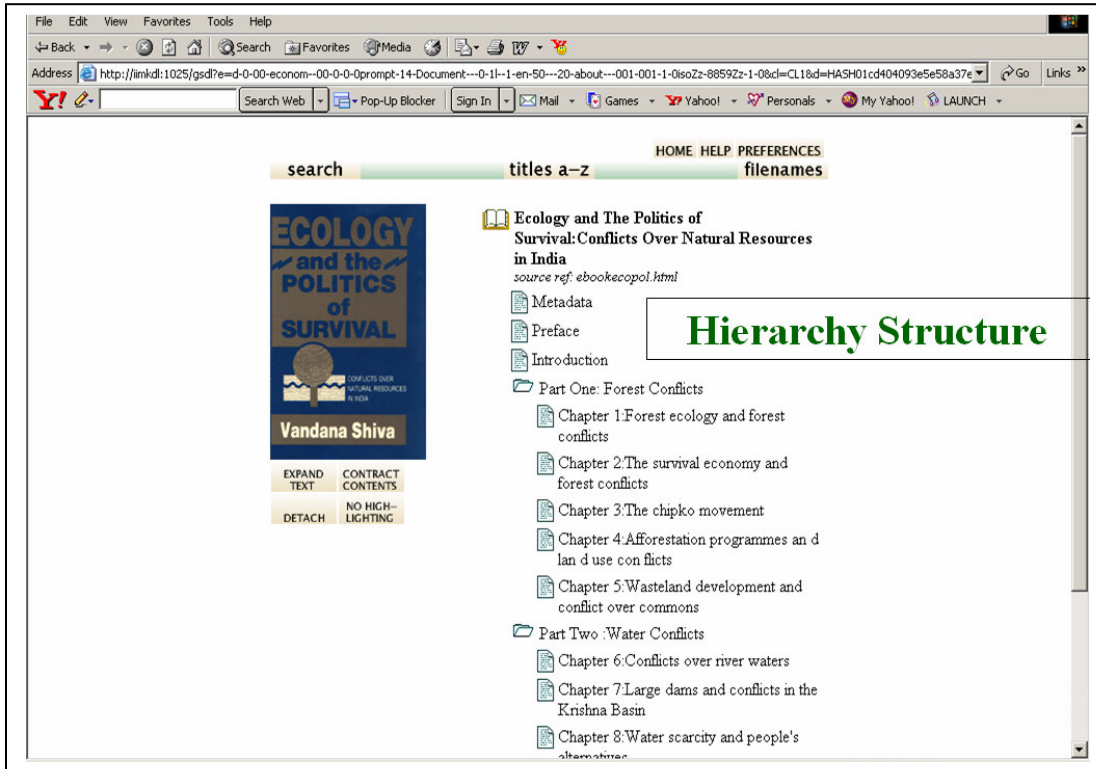


Fig. 1. An EBook with the Hierarchy Structure

3.16. Customization of User Interface (MyLibrary)

In order to change the look and feel of the Greenstone user interface, you need to work on the Collection Configuration (Collect.cfg) files. Customising the User Interface requires a certain degree of knowledge on HTML and some level of Web Designing skills are prerequisites for this.

i. *Collect.cfg* - This is the collection configuration file. You can find this file in the “Program Files\Greenstone2\collect\etc” directory. Details on how to create this file can be found in the Developer’s Guide, “1.5 Collection configuration file” and “2.3 Formatting Greenstone output”.

ii. *Macro files* - Macro files have an extension ‘.dm’. All macro files are stored in the “macros” directory. Details on how to create macros and macro files can be found in the Developer’s Guide “2.4 controlling the Greenstone user interface”.

iii. *Image files* - All images files can be found in the ‘Program Files\Greenstone2\images’ directory.

iv. *Main.cfg* - This file contains a list of all macro files used for the User Interface. If you created a new ‘.dm’ file, you need to add it to this file. The main.cfg file is stored in the “Program Files\Greenstone2\etc” directory.

v. *Getting the Cover Image* - For you to get the Cover Image of your input document, you need to put the image file and the source file (document) into a single folder. They both should bear the same name also. While building the collection, Greenstone will take both the files to “Program Files\Greenstone2\collect\<collection name>\archives\Hash”. The collection thus built will display the Cover Image along with the document. Also in the Design Panel, in the Document Plugin section, while configuring the Arguments for the HTML Plugin, give the custom argument as ‘cover_image’.

vi. *Getting the Collection Icon* - Click on Design panel ->General Option -> URL to home page icon (Browse for image and locate it).

vii. *Getting Header Image for the Digital Library* - To get the header image which says MyLibrary banner in the DL head, create the graphic file (preferably a GIF file), name it as ‘gsdlhead.xxx’ and then replace it with the file available in ‘Program Files\Greenstone2\images.’

viii. *Deep Level Customization* - By default, Greenstone’s collection icon area is a matrix grid (the N X 3 format). You can change the collection icon area by editing the ‘_content_macro’ in ‘home.dm’. You will need to remove the ‘_homeextra_macro’ (this is the N x 3 table that the Greenstone C++ code automatically creates for you) and can then put whatever design customization you want into this area. You will need to put the icons and links to the collection yourself.

You can also achieve high end customization by replacing the ‘home.dm’ with ‘yourhome.dm’ in the \greenstone2\etc folder.

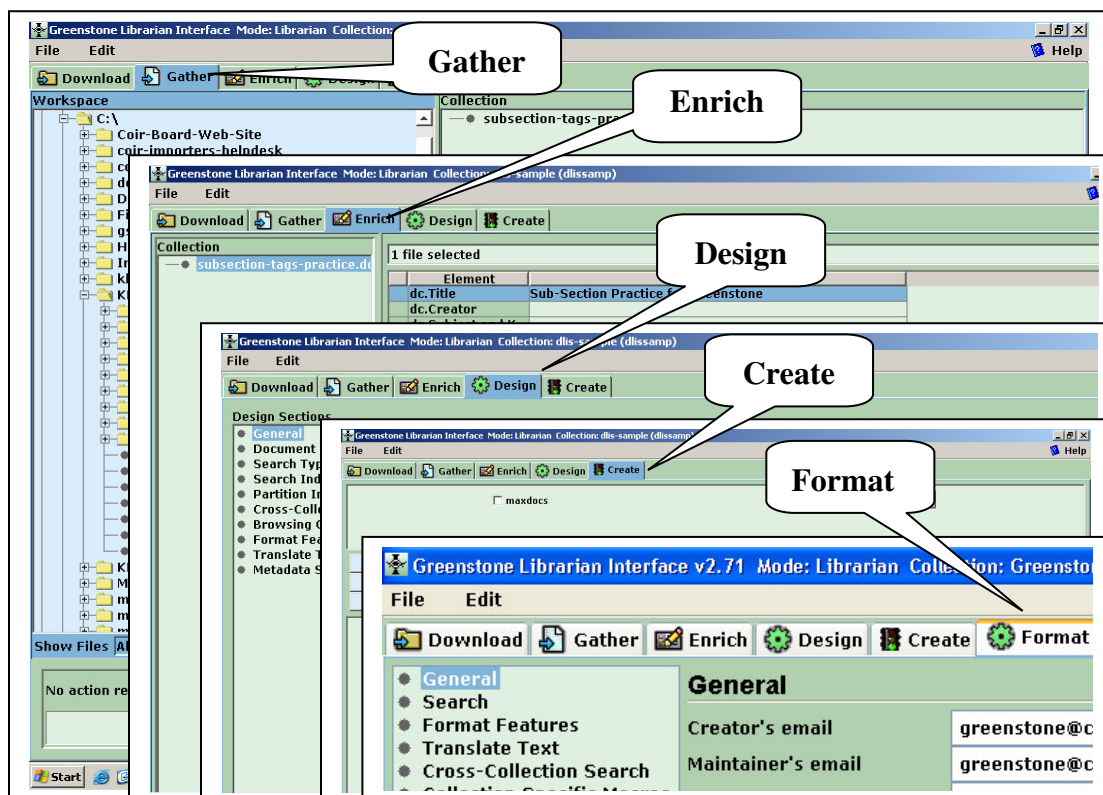


Fig. 2. GLI Panels for Gather, Enrich, Design, Create and Format

3.17. GSDL : Helpline, Archives

Greenstone's E-Mail list is a very useful and active listserv which shares and clarifies user experiences and stories dealing with real life situations. To subscribe or unsubscribe to the list via the World Wide Web, visit "<https://list.scms.waikato.ac.nz/mailman/listinfo/greenstone-users>" or, via email, send a message with subject or body 'help' to "greenstone-users-request@list.scms.waikato.ac.nz". Greenstone has started one more List recently, for the Greenstone 3 Version (the latest Beta version) user group, and the details are available at "<https://list.scms.waikato.ac.nz/mailman/listinfo/greenstone3>".

UNESCO has initiated a Greenstone support organization for South Asia in 2006, supported by a group of experts in the region, and it is coordinated by IIM Kozhikode <http://greenstonesupport.iimk.ac.in>. The site is rich with many of the Greenstone support materials. In addition, an E-list greenstonesupport@iimk.ac.in offers online support to professionals on Greenstone.

For those looking for quick solutions for their real-time or on-the-job trouble shooting while using the software, 'Greenstone Archives' is a treasure house. It is a database of the email messages circulated in the List, and is searchable. The mails generated from the List and its threads are archived and made available for the user community. The archive is available at "<http://www.sadl.uleth.ca/nz/cgi-bin/library?a=p&p=about&c=gsarch-e>". This is the major list used worldwide for Greenstone and the content of the messages is usually global in nature. Developers and Greenstone users can avoid a great deal of unwanted labour by carefully going through the archive before they start working on problem solving, or before shooting a mail to the List.

4.0. Open Access Archives(OAA) and Institutional Repositories(IR)

An archive is a generally accepted synonym for a repository. A repository is a network accessible server that holds scholarly digital content or eprints. Scholarly Archives or Institutional Repositories are established medium to communicate peer reviewed (post-prints) and non-peer reviewed scholarly literature (pre-prints). There are basically three types of scholarly archives in vogue, viz., author archives, institutional archives and subject archives. Subject archives are also called as central archives. According to Stevan Harnad, open archiving is just self-archiving the articles the author has published in (peer-reviewed) non-OA journals. Hence it neither bypasses nor replaces peer-review. It has nothing to do with changing peer review. Self-archiving is a way of supplementing non-OA journal access with an OA version for those would-be users whose institutions cannot afford the non-OA journal.

There are numerous advantages that OA boasts while they campaign worldwide. Authors as well as Institutions can derive a number of benefits out of Archives. For authors, instant dissemination of the fruits of their long years of rigorous research to a global audience is the first and foremost. OA papers get increased visibility through novel models of harvesting done by search engines such as the Google, CiteSeer etc. and the interoperability among similar archives achieved through the Open Archives Initiative (OAI) Protocol for Metadata Harvesting (PMH) are unparalleled value additions to OA Archives. While more visibility leads to more citations, one's research impact naturally gets scaled up. Authors are therefore attracted to come to OA Archives. Additional benefits to self archiving include the assurance of the long term preservation of their articles and the facility to have a proper control as well as meticulous monitoring of one's own Publications.

For institutions too, a long list of advantages and benefits invite them to OA. Firstly, the institute's archive, popularly known as Institutional Repository helps in pooling the organization's Intellectual Capital into once central place which is otherwise scattered, distributed and unnoticed. The archive therefore serves as a one-stop-source or a single access point for the research output of an institution. It provides ample scope for introspection as to whether the institute is going in the right direction on its research activities. Necessary strategies and meticulously designed action plans could be charted out based on the feedback. Institutional repositories facilitates instant generation of research reports and thereby saves a valuable amount of time otherwise spent unwanted. Most importantly the archives ensures long term preservation of its scholarly materials with the help of Open Source softwares and Open Standards of data models and data structures.

4.1. Open Archive Directories and Search Engines

There are many value added services which index OA archives spread globally, as well as harvest metadata records for search and retrieval. OpenDOAR, the Directory of Open Access Repositories lists 502 OA archives situated worldwide [OpenDOAR]. OpenDOAR is a joint effort led by the Open Society Institute (OSI), along with the Joint

Information Systems Committee (JISC), the Consortium of Research Libraries (CURL) and SPARCEurope [JISC],[CURL]. DMOZ, the largest open directory of the Web, lists 59 free access online archives [DMOZ]. The Registry of Open Access Repositories (ROAR) hosted by Eprints.Org lists 607 plus open access archives [ROAR]. OAIster, one of world's outstanding OA repository registry services offered by the University of Michigan, indexes over 663 OAI-compliant open repositories worldwide with an overwhelming 8,593,164 records [OAIster]. Arc, developed by the Old Dominion University, is among the early federated search services based on OAI-PMH protocol [Arc].

4.2. Institutional Repository (IR) Softwares

There are many world renowned free open source Institutional Repository (IR) softwares available such as EPrints, DSpace, FEDORA, ARNO, i-TOR, CDSware etc. They are issued either under GNU public license or the BSD license and can be downloaded from their own sites or open source software directories such as SourceForge [Sourceforge]. Each of the software has a host of features, unique facilities and excellent capabilities, which the users could explore and experiment.

4.3. DSpace

DSpace is a digital asset management software jointly developed by Hewlett-Packard and MIT Libraries, and it is arguably one of the appreciated open source software deployed worldwide for building digital institutional repositories that captures, stores, indexes, preserves, and redistributes content in digital formats. DSpace provides the institutions and universities operate an open access and interoperable institutional repository at the local level. It is also intended to serve as a repository back up for future development to address long term preservation and remote/online access issues. The system was launched during late 2002 as a live service hosted by MIT Libraries, and the source code made publicly available according to the terms of the BSD open source license, with the intention of encouraging the formation of an open source community around DSpace [DSpace Wiki].

4.4. DSpace: Features & Functions

DSpace is a 100% open source software and is freely available for download from the open source software directory SourceForge (<http://sourceforge.net>). The software has been built on a strong architecture supported by state-of-art digital library technologies and embracing almost all latest trends in information sciences. It provides the users, especially the librarians and system administrators, every freedom for building, managing, customizing, administering and Internet publishing world class institutional repositories and digital libraries. Its major features include the ability to accept all forms of digital materials including text, images, video, and audio files. Possible content includes scholarly articles and preprints, technical reports, working papers, conference papers, books, e-theses, multimedia publications, Datasets: statistical, geospatial, matlab,

etc. Images: visual, scientific, etc.; audio files, video files, learning objects, bibliographic datasets, reformatted digital library, collections, Web pages etc.

For enhancing the resource discovery features, DSpace supports Dublin Core metadata unqualified element set as well as provisions for the qualified Dublin Core metadata registry. The software allows the communities and users to publish their articles remotely on the archives. It has CNRI 'Handles' support for Persistent URLs (PURL) which assigns and resolves persistent identifiers for each digital item. Interoperability is another salient feature of DSpace, and it supports the Open Archives Initiative's Protocol for metadata harvesting (OAI-PMH) V2.0 as a data provider. OAI support was implemented using OCLC's OAICat open-source software to make DSpace item records available for harvesting. DSpace uses the versatile Lucene search engine for full text searches. Lucene search engine is a part of Apache Jakarta project, and brings along laudable search features like 'fielded', 'boolean', 'exact term', 'proximity', 'wild cards', 'fuzzy', 'range', 'boosting terms' etc. DSpace supports unlimited exporting/importing of digital content, along with its metadata in a simple XML-encoded file format. The database management system used is PostgreSQL which supports transactions between Oracle as well as MySQL. DSpace enjoys international acceptance across the world and it provides a customizable Web interface. The workflow process for content submission, the decentralized submission process, the remote publishing facility are regarded the unique features of DSpace. Most importantly it is Open URL compliant also [DSpace Wiki].

4.5. DSpace Installation

Mainly six prerequisite softwares are essential for running a DSpace server, viz., i). Java SDK, ii). Apache, iii). Tomcat, iv). Apache Ant v). PostgreSQL and vi). the DSpace software itself. These softwares are to be installed in sequence also. It is very important to setup corresponding HOME variables and modify the PATH variables after the installation of Java SDK and Apache Ant respectively. After the installation of PostgreSQL, we need to create a database named 'dspace' owned by user 'dspace' with UNICODE encoding. To load DSpace you have to start the three services, namely, Apache, Apache Tomcat and the PostgreSQL Database server.

4.6. DSpace Configuration

Primarily configuration of DSpace Software is done by editing the file 'dspace.cfg' located at \dspace\config, which contains basic information about a DSpace installation, including system path information, network host information, SMTP mail server address and other things like site name etc. We configured 'dspace.url' line in the 'dspace.cfg' file with the desired DSpace site address, i.e., the URL 'dspace.iimk.ac.in'.

For mailing purposes we need to modify the 'mail.server' configuration item in 'dspace.cfg' file on a case to case basis. There could be two instances here – i). the Dspace server itself has got a mail server configured and running (say, sendmail) or ii). the mail server is running elsewhere. You need to furnish the SMTP mail server address as per the situation, in the 'dspace.cfg' file, located at c:\dspace\config\ . If the Institute has a separate mail server, this server will have to relay the DSpace server IP.

4.7. Communities and Collections

In DSpace, a digital repository is organized in terms of communities, sub-communities and collections. In other words, communities, sub-communities and collections can be arranged hierarchically. In our repository we followed the subject approach for creating communities and publication type classification for creating collections within each community.

4.8. Collection Building

Adding content to DSpace is quite easy and straight forward. As DSpace has a workflow based remote publishing facility, authorized users can submit their items from their own client machines. You will need to be logged in to DSpace before you can submit. Most collections will also require specific authorization for you before you can submit items to it.

4.9. Workflow:

After starting a submission, you will be led through a seven-step workflow process. These include some basic metadata descriptions about the materials first, then several screens where you describe the details, then file uploads, a verification screen, a license granting screen and finally a submission complete screen. The following figure (Fig. 3) shows the workflow process of item submission in DSpace.

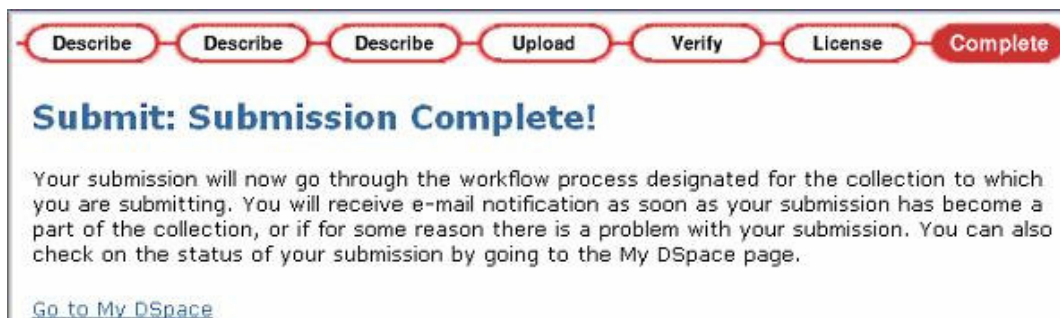


Fig. 3. DSpace Collection Submission Workflow

After submission completion, the submitted item will go through some formalities like review, edit, or approve according to the collection's policies. This means that the submitted item might not go directly into the main archive, before the validation process.

4.10. DSpace Customization

DSpace is implemented by using Java Servlets and JavaServer Pages which produce the HTML pages for DSpace. As JSP coding is similar to HTML, changing the look and feel of DSpace page is very easy. You can make your own header image by replacing the existing 'dspace-blue.gif' located at \tomcat5\webapps\dspace\image\ with customized image. We have edited the news-top.html file located at \dspace\config\ for giving an

introductory note about the repository in the DSpace home page. By editing the news-side.html located at \dSPACE\config\, we added information about our library related programmes/seminars/workshops etc. We also configured the item count against communities and collections by setting the 'webui.strengths.show' configuration item's value to 'true' in the 'dSPACE.cfg' file. DSpace uses a 'styles.css.jsp' file which is located at \Tomcat5\webapps\dSPACE\, which we modified for altering font type, size and colours from default style. The home page of the IIM Kozhikode's DSpace repository is shown in Figure 4.



Fig. 4. DSpace @ IIM Kozhikode

4.11. DSpace Administration

In DSpace administration the administrator has to do a wide range of tasks for the successful maintenance of a digital repository. When we first configure a digital repository using DSpace, we begin with creating Communities and Collections. Policy setting is the most important step in the administration of a digital repository. DSpace administration provides a number of archive control features. We can literally control the access, usage and preferences of each and every collection as well as user(s) or community through this versatile tool. We have to take a decision with regard to whom or which group who can submit digital items to each collection. In addition, we also have to

take a decision as to who or which groups of members (E-people) are authorized to review, approve and modify metadata while submission / collection building.

4.12. E-People

Collection items can be accessed by everyone (anonymous group), but users must be authenticated to perform functions such as submission, email alert or administration. DSpace calls its registered members as *e-people*. DSpace holds the details about each e-person such as their E-mail address, first and last names etc. E-people can be members of 'groups' to make administrator's tasks easier when manipulating authorization policies.

4.13. Publishing the Archive on the Internet

This is one of the simplest yet most interesting and enjoying step in the development of your IR. You need to identify a suitable domain name, configure the same, and give the value at 'dspace.url' in the 'dspace.cfg' file, say 'dspace.url = https://dspace.iimk.ac.in:8080/'. You will then need to register the domain with a public IP. By default, DSpace is running on 8080 port, i.e anybody accessing 'dspace.mydomain' will have to give http://dspace.mydomain:8080. If you want to avoid 8080, you have to configure port forwarding 8080 to 80 (the default port) using a firewall. We can now access the archive as http://dspace.mydomain.

4.14. DSpace Lifeline

DSpace's mailing lists (eLists) are very useful and powerful, and there are three active listservs maintained by DSpace which shares and clarifies user experiences and stories dealing with real life DSpace situations. The 'dspace-general' list is reachable through 'dspace-general@mit.edu', or subscribed to by visiting 'http://mailman.mit.edu/mailman/listinfo/dspace-general' and following the instructions to subscribe. Beginners are advised to first check the 'FAQ' (<http://wiki.dspace.org/EndUserFaq>) or the 'archive' (<http://mailman.mit.edu/pipermail/dspace-general/>) to see if the question in hand has been answered before. For systems professionals and developers there are two more Lists, viz., the 'DSpace-Tech' (technology discussion list) and the 'DSpace-Devel' (developers' list), and both of them are very informative and supportive to post questions or contribute one's expertise to other developers working with DSpace, and to share ideas and discuss code changes to the open source platform. To subscribe to these, you may visit 'http://lists.sourceforge.net/lists/listinfo/dspace-tech' and 'http://lists.sourceforge.net/lists/listinfo/dspace-devel' respectively. Also, there are separate archives for these two lists for getting access to the old postings.

From India, the eList managed by DRTC is very active and useful. You can subscribe to the List via the Web at <http://drtc.isibang.ac.in/mailman/listinfo/dlrg> or by sending a message with subject or body 'help' to dlrg-request@drtc.isibang.ac.in.

DSpace Wiki (<http://wiki.dspace.org/>) is also a very useful reference tool which is worth looking at before we send out messages to eLists, as there could be possible solutions already provided in the Wiki.

4.15. Institutional Policy on Open Access

Setting up an institutional repository is not a big deal now a days. But arriving at a suitable and feasible open access policy at the institutional level is a Herculean task and this need the active participation of the information professionals and the scholarly community of the institution. We need to do a bit of scouting, and if necessary, little lobbying also towards this. A reasonable amount of guidance on this and also on submission guidelines, author benefits, copyright issues etc. could be well seen at the IR at the Indian Institute of Science (IISc) set up by NCSI (<http://eprints.iisc.ernet.in>).

5.0. Open Archives Harvester

Two of the major value adding features of OA archives, among many others, are their Internet presence (omnipresence) and interoperability. The interoperability feature keeps all OA archives virtually as a single digital library system wherein they share their metadata through some common services called metadata harvesters or service providers using the OAI-PMH protocol.

A number of tools are now available for starting such services. The OA harvester service PKP harvester software (<http://pkp.sfu.ca/?q=harvester>) developed by the Public Knowledge Project of the Simon Fraser University, Canada is an excellent application software which can be easily downloaded, configured and customized.

The PKP OAI Harvester allows you to create a searchable index of the metadata from Open Archives Initiative (OAI)-compliant archives, such as sites using Open Journal Systems (OJS) or Open Conference Systems (OCS).

Harvester version 2.x includes the following features:

- Fully functional harvesting and search engine without any coding required
- Built-in support for OAI Protocol for Metadata Harvesting (v1.1 and v2.0)
- Built-in support for Dublin Core, MODS, and MARC metadata formats
- Additional support for harvesting protocols and metadata formats may be added via plugins
- Reading Tools for content, based on administrator's choice
- Context-sensitive online help
- Flexible search interface that allows simple searching and advanced searching using crosswalked fields from all harvested archives. Advanced searching of archives that share the same schema will be possible using fields as defined in the schema. When creating crosswalks for searching, admins can define elements are text, date, or HTML multiple select interface widgets.
- Ability to perform post-harvest and pre-indexing filtering/normalization on metadata.
- User Interface with CSS and template-based HTML for easy customization.
- Searching is highly scalable (creates an inverted index for searching).

A discussion forum (<http://pkp.sfu.ca/support/forum/>) is also available.

Installing OAI Harvester

You will need to install the following in the reverse order:

- PKP OAI Harvester 2.0.1 (<http://pkp.sfu.ca/harvester2/download/harvester-2.0.1.tar.gz>)
- PHP support (4.2.x or later)
- MySQL (3.23.23 or later)
- Apache (1.3.2x or later) or Apache 2.0 (2.0.4x or later) or Microsoft IIS 6

Operating system: Any OS that supports the above software, including Linux, BSD, Solaris, Mac OS X, Windows.

For examples from India, DRTC's SDL (Search Digital Libraries) harvester service has indexed about 38614 records from 24 archives spread globally (<http://drtc.isibang.ac.in/sdl/>). CASSIR (Cross Archive Search Service for Indian Repositories) is a DSIR sponsored cross-repository indexing and search service recently launched by NCSI (National Centre for Science Information), IISc. (<http://casin.ncsi.iisc.ernet.in/oai>). CASSIR, a PKP archive server based service, harvests metadata from country's OAI-PMH compliant institutional repositories, and provides search and browse functionality over the Web.

A manual on installation, configuration and system administration is available at <http://pkp.sfu.ca/harvester2/AdminGuide.pdf>.

Conclusion

The ever changing landscape of the information paradigm poses a host of new IT and information challenges not only to the library and information professionals, but to the users, patrons and scholars and the publishing community as well. Indeed the new environment throws up a host of unprecedented features and avenues, and interestingly enough, if we know how to tap them well, we find there is a plethora of opportunities, and most of them even for free. The 'free things', so belovedly called world wide the 'Open Source Softwares', many of them could even be compared against their commercial counterparts in terms of their strength, efficiency, power and the ever increasing user base. Among the major challenges include the information professionals' emergent need to acquire the necessary skill sets and working knowledge on the cutting-edge information science and information technology areas and in leveraging them in a contextually relevant manner.

References

1. Morgan, EL. <http://infomotions.com/musings/>

2. Orsdel, Lee Van; Born, Kathleen. 2002
Doing the Digital Flip.
Library Journal, 127 (7): 51-55.
3. OCLC Report on Five-Year information format trends. 2003
<<http://www.oclc.org/reports/2003format.htm>>
4. Marchionini, G. 1998
Research and development in digital libraries. Allen Kent (Ed.)
Encyclopedia of Library and Information Science, 63: 259-279.
5. ICDL 2004. <www.teriin.org/events/icdl/background.htm>
6. Sreekumar M.G. and Sunitha T. 2005
Essential Strategies and Skill Sets Towards Creating Digital Libraries Using Open Source Software.
[Proceedings of NAACLIN 2005, DELNET, Bangalore, India].
7. Borgman, Christine L. 1996
Social Aspects of Digital Libraries, pp170
[Proceedings of the first ACM international conference on Digital libraries
Bethesda, Maryland, United State, March 20-23, Organised by Association of Computing Machinery]
8. Frumkin, Jeremy (ED). 2002
Special Issue: Open Source Software
Information Technology and Libraries 21(1)
9. Stallman, Richard
<http://www.opensource.org/docs/definition_plain.html>
10. OSS Watch <http://www.oss-watch.ac.uk/talks/2003-09-24-csg/index.xml.ID=body.1_div.37>
11. Altman, M. 2001
Open Source Software for Libraries: from Greenstone to the Virtual Data Center and Beyond.
IASSIST Quarterly. Winter : 1-7.
12. GNU (GNU's Not Unix!)
<<http://www.gnu.org/> (13 June, 2005)>
13. SourceForge.Net (world's largest Open Source software development website)
<<http://www.sourceforge.net/>>
14. RDF (Resource Description Framework)

<<http://www.w3c.org/RDF>>

15. DCMI (Dublin Core Metadata Initiative)

<<http://dublincore.org>>

16. Greenstone Digital Library Software

<<http://www.greenstone.org>>

17. Witten, Ian H. et al. 2001

Greenstone : Open-Source Digital Library Software

D-Lib Magazine, 7 (10): 1-16.

18. Witten, Ian H. 2003

Examples of Practical Digital Libraries : Collections Built Internationally Using
Greenstone

D-Lib Magazine 9 (3): 1-15.

19. Unicode Consortium

<<http://unicode.org>>

20. IIMK Digital Library

<<http://iimk.ac.in/gsdll/cgi-bin/library>>

21. Greenstone Support for South Asia

<<http://greenstonesupport.iimk.ac.in> >