

OPEN SOURCE INTEGRATED LIBRARY MANAGEMENT SYSTEMS: COMPARATIVE ANALYSIS OF KOHA AND NEWGENLIB

Mani Bhusan Roy

Ex-Student of Department of Library and Information Science
North Bengal University

Email - manibhusanroy84@gmail.com

Dr. Naresh Kumar

Librarian, Galaxy Global Group of Institutions
Dinarpur, Ambala

Email: reach4naresh@gmail.com

Abstract

This paper aims to study the open source integrated library management systems, i.e. Koha and NewGenLib, to inform librarians about what considerations to make when choosing an open source integrated library management system (ILMS) for their library. The paper provides a detailed comparative analysis of both types of software, i.e. Koha and NewGenLib which are undertaken in the study. Both types of software are web-enabled and support library automation. Koha has more specific characteristics of open source ILMS. Koha needs to upgrade, modify and improve its features. Koha requires very little hardware and is easy to install. Koha has advanced database features. NewGenLib has better functionality of modules than Koha. More formats and standards are supported by Koha. Digital library functionality of NewGenLib is more specific in terms of technology, data structure and programming. Also NewGenLib provides more user help and support whereas Koha provides more user-friendly downloads and a documentation facility. NewGenLib has more enhanced features which are significant for ILMS while selecting software for automation. The comparative analysis of the open source ILMS done in the study will help librarians in making necessary considerations while choosing open source software for the library.

Keywords: Integrated Library Management System (ILMS), Comparative Analysis, Koha and NewGenLib, Open source software.

1.0 Introduction:

Due to the advancement of technology the libraries are forced to adopt new technology. The libraries adopted the technologies in Library automation, digital library, content management systems, journal publishing, information management, electronic resource management etc. The Free/open source movement has brought the revolution in the field of library science, the developers, contributors and open source software users have turned more and more to free/open source software (FOSS). The FOSS is hosting on cloud computing to overcome the challenges of technical support and other aspects of the security. For open source software's the source code is free to edit, modify and redistribute to others but for Free software's the source code may not available for modify but software can be used for free and it can redistribute to any numbers. These FOSS are helping the libraries with financial crunches and the librarians to get the wider scope for their carrier. Here the detailed study of Free/open source integrated library management systems for library automation, it helps the librarians to select the best open source software for library automation.

Libraries are always concerned about the cost as they often have more demands than resources and they are funded by their parent organisations with a responsibility to manage public funds wisely. If libraries can serve their users' needs at a lower cost by using open source software rather than proprietary software, it would be in their interest to adopt it. Minimally funded libraries, such as small libraries and libraries in rural areas and developing countries, may find OSS affordable. Library decision makers are concerned about the total cost of ownership of an integrated library management system (ILMS). This includes costs of installation, documentation, training, support, and

maintenance. Therefore, the purpose of the present study is to inform librarians about what considerations to make when choosing an open source ILMs for their library. The study investigates why some libraries have chosen a particular open source ILMs and what are the different features and benefits of the select open source ILMs. The study will also analyse ILMs features in each module of the select ILMs, to insight the librarians to know technical aspects to each ILMs. Examples of well-known Open source ILMs include Avanti MicroLCS, Emilda, Evergreen, Gnuteca, Koha, NewGenLib, OpenBiblio, PhpMyLibrary, andPhpMyBiblio.

2.0 Review of the literature

The review of related literature is done on different aspects of Koha and NewGenLib, open source integrated library management systems including various studies on open source software, open source software and libraries, integrated library management systems (ILMS/ILS), current research findings on open source, studies on different open source software with addition to evaluation and comparison of open source software.

Bissels, Gerhard (2008) describes the selection process and criteria of implementation of Koha 3.0 at the Complementary and Alternative Medicine Library and Information Service (CAMLIS), Royal London Homoeopathic Hospital. Koha 3.0 was selected because of the GNU license was considered more future-proof than proprietary products, and more open to customization to meet the special needs of the library.

Cargile, Cathleen (2005) examines the potential of open source integrated library systems (OSILS) and their applications. He pays particular attention to four OSILS software's like Koha, Emilda, MyPhpLibrary, and Learning Access ILS. Brief notes are included relating to OS operating systems and programming languages.

Dalziel, Karin (2008) reveals the OSS solutions to meet the needs of library patrons and disadvantages of OSS in respect of support by vendors and volunteers. Today, libraries can choose open source and enjoy the benefits of full support and turnkey hosting for open source ILSs. In the US, libraries currently have two options: LibLime, which supports the open source products Koha and Evergreen, and Equinox supports Evergreen only. This paper takes a look at LibLime, primarily their Koha and Koha ZOOM offerings.

De Smet (2009) in his paper presents the new ABCD software for free and open library automation with ISIS with its technological and practical characteristics. As a web-based integrated solution it combines most (if not all) functions of other systems such as Koha with the flexibility of the (Win) ISIS software to create and handle databases of any structure. The main technical characteristics as well as some managerial issues are briefly presented. The planning on the further work is discussed along with some challenges related to the specific nature of the ISIS users' community.

Helling, John (2010) compared the two leading open source library management system (LMS) packages and highlights the reasons behind a switch from one open source provider to another. It is based on practical experience. Koha and Evergreen ILMs are presented in terms of their current use as well as their future directions of development and opportunities. He compared the history of Koha and Evergreen. The former is criticized because of the recent "fork," i.e. commercial use of open source code. Problems with a local implementation of Koha are also discussed. Evergreen however, since it was developed by a library consortium, appeared the better choice for the Indiana Shared Library Catalog (ISLC), a consortium of different libraries in Indiana.

Kapoor and Goyal (2007) in their paper seeks to provide a comparative analysis of the functionality of five web based OPAC's which includes the web-based OPACs of Libsys, VTLS's iPortal, NewGenLib, Troodon, and Alice for Windows, implemented in five academic libraries in India. It was found after the comparison of functionality of OPAC that the web-based OPACs investigated offered a range of facilities for searching by author, title and control number and by keywords which was a useful source for librarians who are planning to introduce web based OPAC and also important for vendor who wants to improve their products. Yang and Hofmann (2010) present a comparative study of the next generation features and services of the OPACs of two open-source ILSs (Koha and Evergreen) and one proprietary ILS (Voyager's WebVoyage). The study results reveal that open-source OPACs is more favourably to the ideal next-generation catalogue than the proprietary OPAC. However, none of them is capable of federated searching. Only Koha offers faceted navigation. WebVoyage does not even provide a spell checker.

Russo Gallo, Patricia (2007) describes the characteristics of the open source library management software, PMB, including licensing, requirements and functionality, then they analyses the features of the different modules and the OPAC. The author considers PMB to be appropriate for both large and small libraries and document centres, enumerating examples of existing applications in Catalan libraries.

Wan (2007) notes that the “Linux” has become a major operating system and “Apache” as the most popular web server today which leads to the familiarisation of open source software (OSS) among many people. Also many software developers from all over the world have been contributing their efforts to build a huge OSS community that will exercise a great impact on various disciplines and helps librarians and information professionals to believe that relatively new model of OSS would bring lotsof benefits to libraries. Wray and Mathieu (2008) developed and tested a model of therelative performance of open source software (OSS) projects by evaluating multiple project inputs and multiple project outputs by using a data envelopment analysis (DEA) model. This model can be used by OSS project developers to evaluate the relative performance of their projects and make resource decisions.

Selection of ILMs

The open source ILMs Koha and NewGenLib are targeted in this survey. Other open source ILMs exist, but Koha claims to have been the first Open Source LMS and NewGenLib, an Indian Integrated Library Management System, is now freely available as open source under the most widely used free software license, GNU General Public License (GNU GPL) v3 since 2007. The software selected for the study is based on the following criteria:

The software must be freely downloadable and available for installation with number of reasonable installed bases. The software should be standard based, well known and commonly used such as in universities and public libraries etc. The software must be compatible with most common platforms of Linux, Windows and Mac OS. On the basis of given criteria, Koha and NewGenLib were selected as described above.

3.0 Overview of Integrated Library Management Systems:

Koha: It is developed in the year 1999 by Kaptio Communications for the Horowhenua Library/ Trust of New Zealand. Koha is web based open source integrated library system used world-wide by public libraries, special libraries and educational institutions. The catalogue data is stored in MARC format and accessible through Z39.50 servers. It has using Web 2.0 technology for tagging and to feed RSS. It also the Union cataloguing to unity the group of catalogues and also no vendor lock-in.

Newgenlib: It is an open source integrated Library Management system developed by Verus Solutions Pvt. Ltd. with Kesavan Institute of Information and Knowledge management, Hyderabad, India and released in the year 2005. It is web based software using JAVA and has multi-tier architecture. It is a compliant of MARC format and OAI-PMH and Z39.50. It has RFID integration and supports multi-user and multiple security levels. The latest versions are supporting the Android Mobiles and tablets to find out the transactions of users of the Library.

Technology

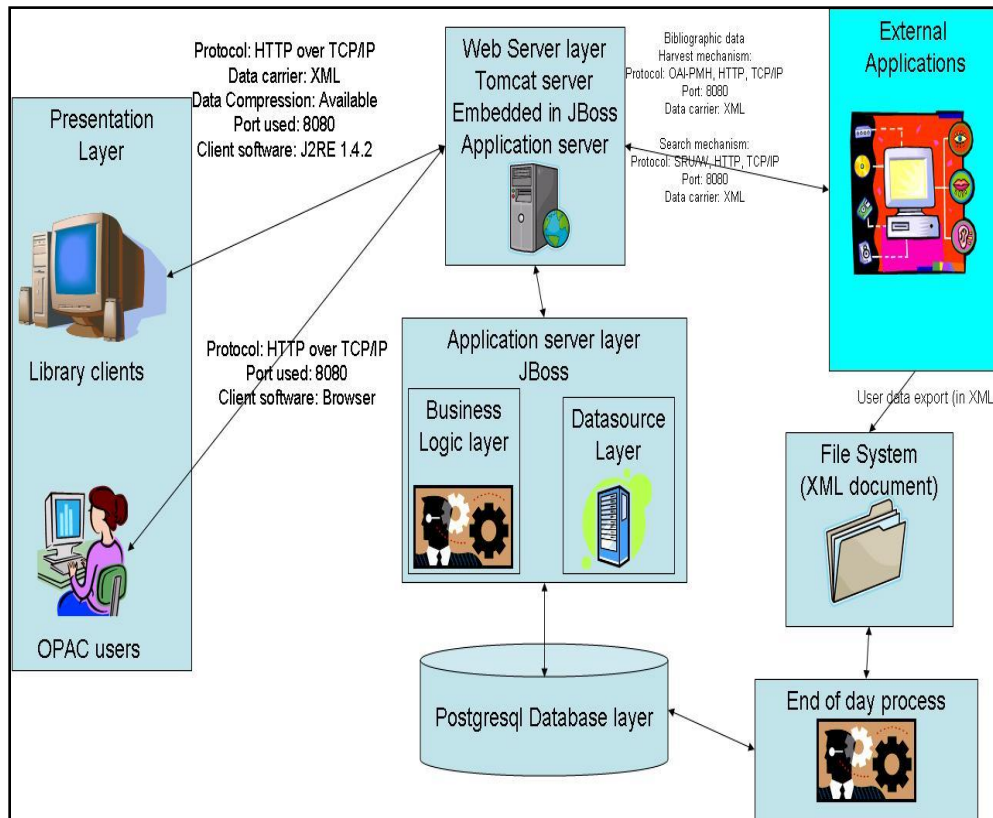
Both Koha and NewGenLib are web-based solutions. By this is meant that they are distributed applications; programs that run on more than one computer and communicate through a network or server. Specifically, web applications are accessed with a web browser and are popular because of the ease of using the browser as a user client. For the enterprise, the ability to update and maintain web applications without deploying and installing software on potentially thousands of client computers is a key reason for their popularity. Both Koha and NewGenLib use programming languages which are designed for web applications and therefore are platform-neutral. Both are possible to be used on a single computer (localhost), an institutional LAN or WAN, as well as in consortium mode across several networked libraries via the worldwide web (Web). (Reddy)

Architecture:

Koha is based on the classical two-tier architecture where a client computer connects directly to the server without any mediating application. The querying of the Koha database and transaction processing is done via CGI scripts written in Perl; all processing happens on the client computers, and the connection to the server is used only for data retrieval. Koha uses the Apache web server and the MySQL database server, two of the most popular open source tools in use worldwide. It was originally developed for the Linux platform but has recently been ported also to run

under Windows although the developers and service providers clearly state that the Windows version is not as effective as the Linux version.

NewGenLib is based on the so called n-tier architecture with an Application Server that mediates between the client machines and the database server. The diagram below shows the NewGenLib architecture.



NewGenLib Architecture

The concept of Application Servers [3] followed the success of the Java platform known as J2EE (Java-2 Enterprise Edition). The advantages of using a middle tier application server resulted in many enterprises level application servers and among these, IBMs WebSphere, WebLogic, JBoss, JonAs, Glassfish are well known. In an application server, Web modules are servlets and Java Server Pages (JSP), while business logic (processing algorithms specific to the business for which the application is designed) is built into the Enterprise JavaBeans (EJB-3). A Java Server Page executes in a Web container—the Java equivalent of CGI scripts. JSPs create HTML pages by embedding references to the server logic within the page. The Hibernate Project (also used in NewGenLib) provides an EJB container implementation for JBoss. NewGenLib’s OPAC uses JSPs. The JBoss application server (open source from RedHat) embeds the Apache Tomcat web server another open source offering.

4.0 Several advantages of using a middle-tier application server are cited and some of these are:

- ❖ The integrity of data and code is better ensured
- ❖ Centralized business logic on server machines, updates and upgrades to the application for all users can be guaranteed.
- ❖ Configurations within an installation can be centralized.
- ❖ Changes to the application configuration, such as a move of database server, or system settings, can be done centrally.
- ❖ A central point through which access to data and portions of the application itself can be managed is considered a security benefit

- ❖ Connection pooling to ensure that response times are good.
- ❖ Better scalability of the application via new versions and/or alternative database servers without too much reprogramming effort as long as the application server conforms to the J2EE specifications.

5.0 Software

Koha is entirely written in Perl, an established and powerful scripting language with a long history of successful use in web applications. JavaScript is also used within pages. It is not clear if this includes the use of Ajax. Koha also uses style sheets to control how data is displayed.

NewGenLib on the other hand is entirely Java-based and uses the following related software technologies in its presentation, application and database server layers.

- ✚ **Java Web Start**– based Applets. Java web Start (JWS) is a framework developed by Sun Microsystems, and allows users to start application software for the Java Platform directly from the Internet using a web browser. Unlike Java applets, Web Start applications do not run inside the browser, and the sandbox (A tightly controlled set of resources) in which they run need not have as many restrictions, although this can be configured. Web Start has an advantage over applets in that it overcomes many compatibility problems with browsers' Java plugins and different Java Virtual Machine (JVM) versions. NewGenLib's librarian's interface is entirely written as JWS applets.

Some of the features associated with Java Web Start [4] include:

A richer user interface is possible as compared to using a browser-based one.

- ✓ **Portability:** Java Web Start is available on Windows, Solaris, and Linux, and is expected to be ported to other platforms.
- ✓ **Caching:** Applications launched with Java Web Start are cached locally. Thus, an already-downloaded application is launched on par with a traditionally installed application.
- ✓ **Maintainability:** If the remote application is updated, Java Web Start updates the locally cached version at the application's next invocation.
- ✓ **Easy launching:** Java Web Start allows applications to be launched independently of a Web browser. The application can also be launched through desktop shortcuts, making launching the Web-deployed application similar to launching a native application.
- ✓ **Ability to work offline:** An application can be used in situations where launching through the browser is inconvenient or impossible.

Other software technologies used in NewGenLib's presentation layer are:

- ❖ Java SWING based GUI client - SWING provides a sophisticated set of graphical user interface (GUI) tools for Java programs.
- ❖ Java Runtime Environment (JRE) (freeware). JRE is the only software required at the client side to run NewGenLib's java programs downloaded from the server
- ❖ JDOM (open source) , a parser for XML documents used as the messaging platform between clients and server in NewGenLib. In NewGenLib XML messages between clients and server are compressed before transmission over the Internet. This is a feature that reduces network traffic.
- ❖ Java Help 2.0 for help (freeware)
- ❖ Jasper Reports (open source)
- ❖ Hibernate (open source)
- ❖ Java Mail (Freeware)
- ❖ Jakarta POI (open source)
- ❖ openOffice 2.0 – to generate and print form letters and reports (open source) .

The use of a distinct software technology for the presentation layer in NewGenLib does make it possible, theoretically at least, of moving the front-end to a richer Internet application using new tools such as Adobe's Flex-3. A recent example of an application that exposes a rich internet interface is Biblios (<http://biblios.org/>), an open source cataloguing application. How difficult such a move is for NewGenLib or Koha and the extent to which code can be reused or repurposed is an issue that will come into play.

Technologies used by NewGenLib in the Application server layer are:

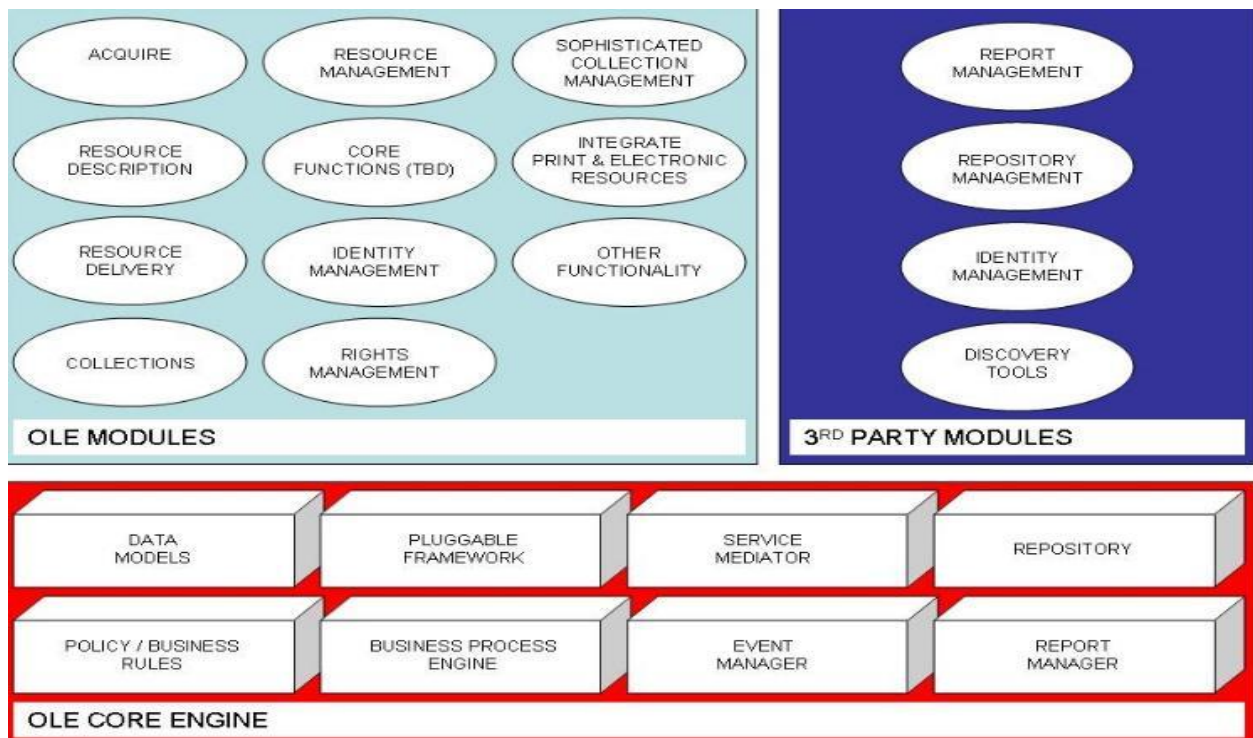
- ✦ Servlets and JSPs
- ✦ Tomcat web server embedded in the JBoss application server (both open source)
- ✦ Struts, an open source framework that has as its goal the separation of model (business logic) from the view (html pages presented to the user) and the Controller (instance that passes information between model and view). A central configuration file binds the model, view and controller.
- ✦ Enterprise Java Beans - JBoss Application Server 3.2.1 (open source)
- ✦ JDOM
- ✦ XCQL – CQL (common query language version 1.1) parser (open source). CQL (Common Query Language now renamed Contextual Query Language) is a formal language for representing queries to information retrieval systems such as web indexes, bibliographic catalogs and museum collection information. The design objective is that queries be human readable and writable, and that the language be intuitive while maintaining the expressiveness of more complex languages. (<http://www.loc.gov/standards/sru/specs/cql.html>). NewGenLib supports only the basic version of CQL and the Bath and Dublin Core profiles. The SRU/W federated search protocol searches use CQL queries.

Given the differences in the technological base of the software, it would appear that NewGenLib, because of its more modular architecture, use of object-oriented analysis and design, its use of middle-tier technologies, XML Streams and EJB containers that encapsulate business logic, is probably better suited to be broken into modules with a service-orientation than Koha. NewGenLib may be in a better position to take advantage of the OLE Project's findings: for modular components dealing with the core processes and for these to connect to one or more open services based on well defined w3c standards.

A distinction between the terms: service-oriented architectures (SOA) and web services is in order. SOA is an approach [<http://www.slideshare.net/bikesandbooks/open-library-environment-samm-08>] where:

- ✦ software pieces are built independently,
- ✦ they can be interchanged or repurposed, and
- ✦ can be combined to create new systems and services.

The diagram below taken from the OLE Project shows a high level reference model of how an open library environment may be. As can be seen, the core functionality comprises the OLE modules. A service bus is an important concept and is shown as the OLE core engine with other 3rd party applications constituting the third leg. (Haravu)



The broad meaning of web-services, on the other hand, is that of providing computer-to-computer communications which are interoperable and independent of the underlying software application, operating system and hardware. Although earlier protocols such as z39.50, ISO-ILL-10160/61, openURL had similar objectives, they are not strictly web-services based which use W3C standards-based compliance with well-defined service requests and service responses. The new protocols: SRU/W (search and retrieve via URL, Search and Retrieve Web), Z39.50 International Next Generation (ZING) and CQL are the new web services protocols. These use XML streams transmitted using one or other web services protocols such as SOAP, WSDL, etc. These are not library-specific and therefore have wider applicability, including for instance between a library and vendor applications and services (e.g., Amazon, Google). Importantly NISO in 2005 created a working group of both libraries and vendors to produce and maintain a Web Services Best Practices document [10] for general use in assessing new and ongoing web service applications, not necessarily confined to the library world. The recommendations and findings of NISO, no doubt, will have a bearing on new versions of ILS, both commercial and open source.

NewGenLib already uses XML streams for exchange of data between clients and servers in its different modules. Conceivably, it should be easy for NewGenLib to expand into using web services protocols both at its client as well as server sides, although it does not use any so far. It is not clear to what extent Koha modules use XML streams to communicate between Koha clients and servers and with other servers. But the fact that Koha modules are all written in Perl and that it already invokes web services means that Koha too should have the capability to expand into being compliant with current and new web service applications and standards.

6.0 System administration

An important criterion that will determine how well software can be used in different application environments is how parameterized the software is. Various parameters (preferences) that are configurable are also important in ensuring that these are used automatically within the functional modules, e.g., budgets required to be charged for acquiring materials, and

Both Koha and NewGenLib provide for the setting-up of parameters. Some of the unique features of each of these, the similarities and differences are pointed out in the table below.

Parameter	Koha	NewGenLib	Remarks/Differences
System preferences	Provides links to set-up global parameters as well as those specific to functional modules.	Provides tabs in the librarians interface to set up global (general system) parameters and those specific to functional modules	NewGenLib allows only system administrators to set-up parameters. It is not clear if Koha uses the concept of system administrators. (See also remarks under different modules)
Global system parameters	Defines these under several tabs which includes both the functional modules as well as specific headings such as Patrons, OAI-PMH, I18/L1ON (internationalization), z39.50 targets	Defines these under several tabs which include both the functional modules as well as specific headings, patrons, z39.50 targets.	NewGenLib allows the setup of some unique parameters: Acquisitions order time (the time in days that a firm order should wait before claims/reminders are sent), Reservation (or hold) queue factor, This is to ensure a maximum number for patron holds based on number of copies of an item held by the library, customizable data to be added to holdings records, customized fields that should occur in patron records, the maximum duration in days until which holds can be claimed by patrons after which the item passes to the next in the queue, etc. Koha allows the set up of enhanced content preferences which includes the use of Amazon Web Services, e.g., Amazon content in the OPAC, querying of FRBRized web services, Google jackets, Amazon similar items. In

	etc.		<p>this respect Koha does provide for mash-ups that are typical of web 2.0 (library 2.0) applications. Internationalization (date format, OPAC language, client-side language is also easier with Koha as compared to NewGenLib. NewGenLib allows the configuration of form letters typically used in libraries.</p>
Cataloguing	Authority values, types, MARC templates	Authority values, types, MARC templates	<p>Koha allows the definition of record matching rules when MARC records are imported. NewGenLib does not have this feature.</p> <p>Classification filing rules based on standard schemes is allowed by Koha. NewGenLib does not provide this feature.</p> <p>Both allow the creation of cataloguing templates using MARC fields and subfields.</p> <p>Koha allows <i>ab initio</i> definition of stop words. NewGenLib builds stop words progressively and only if the keyword index is generated for entered and/or imported records. Stop words have to be specifically marked as such by the cataloger.</p> <p>NewGenLib allows an item to be designated to all material types and physical/presentation forms permitted by the MARC standard.</p> <p>Shelving locations and sub-locations (e.g., General reading, Reference) can be defined in Koha and NewGenLib. Shelving locations show up in the OPAC and a library floor map if configured.</p> <p>NewGenLib allows custom indexes to be built. The fields to be added to the index are identified by their MARC tag.</p> <p>RSS feeds can be configured in NewGenLib. The feeds are then visible via the OPAC</p>
Patrons and Circulation	Patron categories and types, circulation privileges and fine rules with respect to item types.	Patron categories and types, circulation privileges and fine rules with respect to item types, departments and courses	<p>NewGenLib and Koha both allow the setting up of maximum fines (over dues) and also a default check-in date for long-term loans, e.g., to faculty members.</p> <p>NewGenLib allows module and sub -module specific privileges to patrons. It is possible to ensure a high level of security in the use of functional modules, based on privileges for library staff.</p> <p>NewGenLib allows the definition of patron categories and types to as fine a granularity as may be required by a library. This was not seen in Koha.</p> <p>NewGenLib allows also the definition of current and permanent addresses of patrons.</p> <p>NewGenLib also allows patron category-based renewal privileges</p> <p>NewGenLib allows the setting of communication options for a patron. The options are: as an instant message when the patron logs-in to the OPAC, as printed output, as email, none of the above. More than one of these can also be setup for a patron.</p> <p>NewGenLib also allows the setting up of binders, binding types, costs and binding specifications for serial bound volumes.</p> <p>Departments and Courses can be defined in</p>

			NewGenLib. Such a feature would be needed in academic environments. Koha does not have this feature.
Acquisitions (books and serials)	Funds, budgets, vendors, currencies and exchange rates	Funds, budgets, vendors, currencies and exchange rates, fiscal year, fund allocations.	NewGenLib distinguishes between funding source and funds (or budget heads). More than one funding source may be defined and each of these may fund the same or different budget heads. NewGenLib allows a library to define a fiscal or financial year in which budgets are allocated and spent. Koha does not seem to have this feature. Allocated funds in NewGenLib could be defined as possible to carry forward (to the next fiscal year) or restricted to a fiscal year. NewGenLib allows the set up customized shelving locations. A unique parameter called Accession Series is possible to be set in NewGenLib. An accession series makes it possible to uniquely accession different kinds of materials, e.g., theses, CDs, DVDs, etc. NewGenLib uses the unique accession number also as the bar code (or RFID tag) for an item.

It is clear that both NewGenLib and Koha are well suited for use in different application environments including language environments. Koha is better able to be quickly customized to be used in different languages as compared to NewGenLib. However, NewGenLib's cataloging module has features which are required in academic and special libraries, e.g., the entry of analytics. This feature is not available in Koha. (Haravu)

6.1 Functional modules

Both Koha and NewGenLib provide full support to the MARC-21 bibliographic format. Both provide crosswalks to other metadata standards. The extent to which Koha supports the Authorities and Holdings format is not clear. NewGenLib provides partial support to the Authorities and Holdings formats. The table below lists the unique features of each of these, the similarities and differences.(Haravu)

Module	Functionality/Feature	Koha	NewGenLib	Remarks
Cataloging	Data Entry and editing of original records	Exposes a Marc template (framework) for different types of bibliographic items	Exposes three templates: MARC, General and Simple.	In Koha the templates (frameworks) show different MARC tags with indicators and sub-fields. Koha assumes a good understanding of MARC-21 cataloging. This may not be true of developing countries in S Asia. Catalogers need to navigate through 9 tabbed pages. Navigation between tabs is fast. In NewGenLib, the MARC template is a long scrollable page. Tags and subfields can be added on demand. The General template has main tabs and several sub-tabs. Navigation between tabs is fast. Item information is integral part of Koha templates unlike in NewGenLib where item data needs to be entered after bibliographic details are entered and validated. Both Koha and NewGenLib allow full MARC

				<p>cataloging. Koha allows the definition of one or more MARC frameworks in which the library chooses the tags, subfields, etc., it wishes to use for different types of items. In NewGenLib, the General template hides the complexity of MARC and allows entry of data into most of the widely used tags/subfields of the MARC format. The Simple template is meant for a minimal level of detail in catalogue records, suited for small libraries.</p> <p>Since item data (bar code, class number, etc.) are entered separately in NewGenLib there is a risk of creating orphan bibliographic records – records not linked to item data.</p> <p>Location, sub-location and shelving location data is supported in the software.</p> <p>Koha allows entry of details such as date acquired, cost of item, replacement cost. In NewGenLib the information of cost, date of acquisition, etc., needs to be configured as additional fields in holdings records before they can be added.</p> <p>An important feature of Koha, missing in NewGenLib, is the possibility of entering public notes.</p>
	Bibliographic levels and Material types	Koha allows selection of item type: books, Audio cassettes, CD, DVD, etc., and the template displays the fields as appropriate for the chosen type.	Selection of bibliographic level and material type is allowed in each template and follows the MARC standard. All possible combinations of bibliographic level and material type items can be entered.	Koha does not allow the bibliographic description of analytics, e.g., chapters in monographs. NewGenLib allows analytic entries. It is possible to define various types of relationships between an item and its host via the Host/Related terms data as per the MARC format.
	Authorities	Koha allows the search of Authorities (main entry elements and uniform title, topical term, geographic name, form genre) and Headings files.	NewGenLib allows the search of Authorities (main entry elements and uniform title, topical term, geographic name, form genre) and Headings files	It was not possible to test if Koha permits import of authority records from a source such as the Library of Congress. NewGenLib allows the import of authority data into local headings files.
	Data validation	Koha validates that	NewGenLib validates that	

		mandatory MARC control fields, e.g. 008, 004 are entered before a record can be saved. However, entered or imported data in authority controlled fields are not validated against local headings file.	data in mandatory fields of the General Template is entered. The mandatory MARC fields are not as well defined in either the General or MARC templates of NewGenLib. Validation is also done for authority controlled fields with the headings files. Data validation for entered in the MARC template is not as thorough as in the case of Koha.	
	on-line help			The cataloger does not have on-line help In Koha; on-line help is available for each of the templates in NewGenLib.
	Copy cataloging	Koha allows import of records via connection to z39.50 servers.	NewGenLib allows to: Connect to a MARC-21 source that exposes its records as ISO-2709 or MARC/XML and this can then be imported into any of the templates. Connect to z39.50 or SRU/W servers	See separate table for differences in the z39.50 functionality in both the software.

7.0 Z39.50 functionality

The z39.50 functionality in Koha and NewGenLib is provided under the respective cataloging module to assist catalogers to locate one or more records in MARC-21 for a known item record.

Feature	Koha	NewGenLib	Remarks
Searchable fields	Title, author, ISSN/ISBN, Subject Heading, LC Call Number, Dewey class number	Title, author, conference name, corporate name, title series	ISBN and ISSN are important search elements and are missing from NewGenLib. Koha does not allow Boolean combinations in searching the servers. NewGenLib allows one Boolean operator between two search fields. NewGenLib distinguishes between author and personal name fields, presumably to search for data in 100 and 700 fields as well as in personal name authority headings. NewGenLib's search options seem more useful for research and academic libraries, while Koha's functionality seems to be targeted primarily to public libraries.
Search function	Title and author field searches are left and right truncated. When the search element has both surname and first name of an author, the search uses both parts of the name as a single unit and hence the results are specific.	Title and author (personal name) field searches are left and right truncated. When the search element has both surname and first name, both parts of the name are searched independently (as an OR combination); there is thus too much noise and is unlikely to find favor with catalogers	Title searches do not produce predictable results in NewGenLib.
Server types and configuration	Allows only z39.50 servers to be configured. By default only 5 servers are configured. It is possible to configure other servers but this has to be done under the Koha Administration module. Configuration of	Allows both z39.50 and SRU/W servers to be configured. Configuration (adding, editing, deleting and selecting) servers are possible directly from the z39.50 window. Allows also the possibility to select servers by type of library and country.	Koha and NewGenLib both use the Yaz toolkit to implement the z39.50 client function. Koha supports the z39.50 Server functions via the Zebra sever (this was not tested). NewGenLib does not support the z39.50 server functionality. All NewGenLib servers are SRU/W compliant and can be searched via a SRU/W client.
Results display	Shows Title, author, ISBN, server from which retrieved and LCCN of	Shows Main entry, title and servers from which record was retrieved.	

	<p>hits Allows viewing of the record as MARC tagged or as a catalog card. The record can be imported into the Koha cataloging template Does not dedupe records retrieved. If the same record is retrieved from two or more servers these are shown as separate hits</p>	<p>Records are deduped before display. If a record is retrieved from two or more servers, it is possible to select the server from which to perform copy cataloging. Allows viewing of the record as MARC tagged. Records can be imported into templates.</p>	
--	---	---	--

7.1 Circulation

Both Koha and NewGenLib support the work flows typical in circulation management. However, the following similarities and differences are noteworthy. NewGenLib and Koha allow detailed Holds administration including the setting of maximum holds permitted for a title based on the number of copies held by the library, the period for which a hold is valid before it passes to another user or back to circulation. The Holds queue for an item is possible to be modified by authorized library staff. Users can cancel their holds from the OPAC. Koha allows the setting of Holds for available items as well.

Koha allows an item to be transferred to another library configured to be in the network. This is useful and required in public library networks.

The management of repair and re-binding of items is a function that is supported by NewGenLib. This is typically a requirement of libraries in the developing world. Binding management is an integral part of NewGenLib. Koha does not have any binding management functionality.

NewGenLib allows a patron to be declared as 'delinquent' under certain conditions. Staff is alerted when a delinquent patron attempts to check-out items.

In Koha, it is possible to define a maximum fine which when reached disables further issues to a patron.

Koha allows due date for check-outs to be reckoned to include or exclude library holidays. By default NewGenLib excludes holidays in due date calculations.

Koha allows web-based check-outs. This feature is not clearly explained and could mean that a patron could renew loans on-line. NewGenLib allows on-line renewals based on patron privileges.

7.2 Acquisition

The two have some commonalities but many differences in the way acquisitions and related work flows are managed. Koha's functionality, compared to NewGenLib, is pretty straight forward and focuses mainly on managing suggestions from patrons, adding, receiving and modifying orders. Koha allows also the addition of vendors under its acquisitions module and searching for orders by vendors. A very friendly feature of Koha is immediate access to the budgets and figures for expenditures and commitments. In Koha, new acquisitions are also directly cataloged unlike in NewGenLib where cataloging is considered as a distinct activity.

NewGenLib's acquisitions functionality is much wider in scope and reflects the work flows and practices that are typical of academic and public libraries in India and probably other developing countries. These include:

- Management of user suggestions
- On-approval purchases
- Firm orders including the search for orders by fund, vendor, or order number.
- Advance payments and the application of credit notes
- Gift acquisitions
- Receipt of orders
- Accessioning of received items
- Payment processing
- Tendering for supplies of items

Since NewGenLib is targeted to different types of libraries, it is known that some of the above functions are not used (e.g., on-approval purchases) by some libraries. Importantly, in NewGenLib acquisitions each budget needs to be associated with one or more authority (e.g., head of a department) that can sanction expenditures from the budget. On-approval purchase is a method that many public and academic libraries follow and NewGenLib supports this mode. Other major differences include:

- ✚ handling of invoices pending the receipt of an item
- ✚ handling of items pending the receipt of an invoice
- ✚ Rolling back of payments

In view of the complications in the work flows in acquisitions typical of Indian libraries and their practices, the intuitiveness of the module is poor.

7.3 Serials management

It is generally acknowledged that the serials management module of an ILS is its most complicated one from the point of view of developers because of the many idiosyncrasies of serials as a form of publication. Building an intuitive interface for librarians to perform the work of serials cataloging, subscription management and serials registration is indeed a challenge to developers of open source systems. Little wonder then that Evergreen developers are still to develop the serials module. Current systems must deal not only with print serials but also e-serials. Serials cataloging present issues which are different from that of monographs. The display of serials holdings too presents different kinds of problems from that of non-serial items. In fact current ILS are expected to conform to the rather esoteric concepts such as 'patterns and captions', and 'enumerations and chronologies' which the MARC-21 format exposes. In addition, standards such as the z39.71, specific to serials displays too require to be considered. (Haravu)

Koha has taken a direct and simpler approach to serials management as compared to NewGenLib. Koha exposes serials functionality under its cataloging module. NewGenLib has a separate module for serials management. Koha does not have the functionality and features that NewGenLib has and required in academic and special libraries. The table below highlights the features of the software. However, neither of them have full functionality required for the management of e-serials and electronic resource management (ERM).

Feature	Koha	NewGenLib	Remarks
Subscription management	Adding new subscriptions	Adding new subscriptions	Koha requires that a bibliographic record for the new serial is first created before a subscription for it is added. The catalog record can be created directly or via z39.50 import. Public and private notes are allowed in catalog records. The subscription record created includes subscription related data and also data on patterns and captions, e.g., Numbering pattern (e.g., volume, number, issue or seasonal). Koha also asks that a prediction pattern be generated. However, this

Feature	Koha	NewGenLib	Remarks
			feature is not intuitive and does not seem to result in a prediction algorithm to generate receipt dates, claims. NewGenLib follows a more traditional work flow in the adding of subscriptions. Initially only basic information about the serial is entered. Subscription-related data and enumeration and chronology data is added when a subscription order with a vendor is raised. Only after the first issue of the new subscription is received, is cataloging of the serial done. At this stage, detailed information about the patterns and captions is entered. Importantly, NewGenLib exposes a wizard which makes it easy to enter levels of enumerations pattern data. The same data is automatically used to generate a prediction algorithm for the serial. This is altered based on actual receipts of issues.
Serials registration	Koha does not expose serials registration functionality. This may not be required for e-serial subscriptions	NewGenLib has a full-featured registration functionality using a Kalamazoo -or kardex-based interface.	NewGenLib's registration interface allows single-click registration of issues followed by prediction of when the next issue is expected. Out -of-turn, combined, special and supplementary issues and index and title page receipts are also handled. Receipt displays are clearly seen in grid format.
Binding management	Koha does not have a binding function	NewGenLib has an elaborate binding management function for serials.	

7.4 On-line public access catalog (OPAC)

The conversion of the card and print catalog to the on-line catalog has been the basic design for many current ILS. Patron expectations in the new millennium are increasingly shaped by their experiences with Goggle and Amazon. Many current library OPACs tied to the ILS vendor provided offering has constrained libraries from improving their presence on the web. There is a growing trend for libraries to use third-party OPACs that have new functionality that have become accepted as part of the social web. Vendors too have re-vamped their offering or they offer Application Programmer Interfaces (APIs) to allow external OPAC applications to connect with their library databases. Also, they have begun to use web services to draw on other libraries and organizations (e.g., Amazon, Google) to enhance patron experience.

An important development that will influence the future of OPACs is the technical recommendations of the Digital Library Foundation's (DLF) ILS Discovery Interface Task Group (ILS-DI). The group's report (June 2008) makes valuable recommendations for effective interoperation between integrated library systems and external discovery applications and includes a basic level of ILS-application interoperability that was endorsed by most ILS vendors, as well as more advanced levels that may allow even richer interactions in the future.

Given that both Koha and NewGenLib are relatively recent offerings, it seems appropriate to compare the OPACs of these using criteria that are generically considered to be important in the context of library 2.0 applications and technologies. In what follows, OPACs of both are compared across features that have been found to be useful by library users. It is probably true that none of the current offerings have all the features shown below and as Casey [9] points out none of these is cutting-edge technology and is already in use by search engines and e-commerce sites.

The comparison has used Koha's LibLime Academic Library Demo OPAC and NewGenLib's (version 2.2) OPAC. (Haravu)

Feature	Koha	NewGenLib	Remarks
Relevancy ranking	Koha's default search shows records in relevance-ranked order	NewGenLib's free-text search produces records in relevance-ranked order	Koha's relevance ranking method is not known. NewGenLib uses a sophisticated algorithm using a PostgreSQL contributed module called Tsearch2 to produce ranked output.
Clean interface. (By this is meant that there is no need for too many search field options. Instead, all MARC tags and user tags should be searched by default and results relevancy-ranked. Results pages should be well laid out and help users to refine searches).	Koha provides a clean interface but does provide a few options for title, author, subject, ISBN. All searches result in relevance-ranked order. The brief results display pages are well organized and show 3 columns with the bibliographic details in the middle. A lot of Item information is seen including availability. Users also have a link to place the item on hold. The left column of brief displays allows the search to be refined by availability, library, authors, related topics, etc. The item details display pages of chosen record are also well laid out. Item details pages have links to user comments, description, Amazon	NewGenLib's interface looks cluttered with links in a box on the left to user-initiated actions, e.g., see circulation data, request new addition of an item. The links appear on all search option and result pages and this does not make for a clean interface. The free-text search is the only one which is simple for users. Other options: Basic and Advanced search show too many options, which most users may never need or use, e.g., ISBN. Relevancy-ranking is limited only to the free-text search results. Brief results pages are column based. This limits the amount of information that can be shown for hits. The item details display pages are also column-based. These pages too have	In Koha, results pages, sorted by popularity, are noisy. Koha uses APIs or web services from Amazon, Google, Baker and Taylor among other providers. These web services can be enabled or disabled by a library depending on their need. NewGenLib does not use APIs/web services to mashup or enhance displays. Koha displays links to search for a title in the detailed display page in WorldCat, Google Book Scholar and on-line book stores.

	reviews and ratings. Users can add their reviews. Cover art is shown for some of the titles in the brief as well as details display. Serial records have a link to the full text item.	the user-initiated action links box which does not result in a clean look to pages. The user has to scroll down to see availability details. This is not very friendly. Cover art is not part of results display pages. Enhanced content mashups are not available in NewGenLib.	
Spell checking (did you mean option)	Not available	Not available	
Faceted searching (including the display of Tag clouds based on the search term)	Not available	Not available	
Advanced searching	Koha allows Boolean operators between selected search fields. As many search fields can be selected as needed. It is possible to show all the search field options or only the most important ones.	NewGenLib allows selection of a maximum of two fields in a Boolean combination. The searchable fields are pre-configured.	
User tagging	Koha allows users to tag items and search tags	Not available	
Full-text searching of holdings (not only citations)	Not available	Not available	
Reviews	Koha allows access to reviews on Amazon. This feature too is by virtue of using an API exposed by Amazon.	Not available	
Blogs	Not available	Not available	
Option to allow reputation ranking	Not available directly but users can see Amazon ratings.		
Aggregated rating system	Not available	Not available	
Suggest to friends (email)	Not available	Not available	
RSS feeds for the catalog and library web site	Not available	Not available	

8.0 Conclusions

The analysis of Koha and NewGenLib software (both are web-enabled) shows that library automation is a necessary and easy process. In the study, the actual working and functioning of the software with all their prominent aspects and other features and functions are reported. Each software has its own design and architect for the work flow. In each software almost all modules, necessary features, technology are available but after their evaluation and comparative study, it is concluded that Koha has more specific characteristics of open source ILMS. NewGenLib

has more features than Koha, which needs to be upgraded, modified and improved. Koha is easy to install as it requires very less of hardware in comparison to NewGenLib. Koha has a dual database design which makes it more flexible to work and use its database. It is further found that NewGenLib has better functionality of module than Koha. But few of the modules are also better than NewGenLib as Koha has simpler and user-friendly acquisition module and advance OPAC system. It is clear from the study that NewGenLib has a better cataloguing module, circulation module and serial management than Koha where it is lacking in many of the prominent features and functions required as ILS by a library. Reporting in NewGenLib meets the requirements in better way than Koha. Koha supports more number of different types of formats and standards in comparison to NewGenLib. Digital library functionality of NewGenLib is more specific in terms of technology, data structure and programming. Also NewGenLib provides much more user help and support whereas Koha provides more user-friendly downloads and documentation facility. NewGenLib has more enhanced features which are significant for ILS while selecting software for automation. The study concludes that both software packages are more or less equally important in different aspects, so it is the choice for the client library/librarian to select the software on the basis of their requirements to automatetheir entire library system. The enhancements and improvements could be done on the basis of the conclusion drawn after the evaluation and comparison of the both ILS.

9.0 References:

1. Bissels, G. (2008). Implementation of an open source library management system: experiences with Koha 3.0 at the Royal London Homoeopathic Hospital. Program: Electronic Library and Information Systems, 42 (3), 303-314. Accessed 1 April 2017.
2. Cargile, C. (2005). Open source ILS for the non-systems librarian: a reality? PNL Quarterly , 69 (3), 15-17. Accessed 2 April 2017
3. Dalziel, K. (2008). Open source meets turnkey: Koha for software, LibLime for support. PNL Quarterly , 72 (3), 15-16. Accessed 1 April 2017
4. De Smet, E. (2009), "ABCD: a new FOSS library automation solution based on ISIS", Information Development, Vol. 25 No. 1, pp. 61-7. Accessed 4 April 2017
5. Haravu, L. (n.d.). Comparison of two open source integrated Library Systems (ILS): Koha (version. 3.0) and NewGenLib (version. 2.2 beta). 1-21. Retrieved April 2, 2017, from [https://www.google.co.in/#q=Comparison of two open source integrated Library Systems \(ILS\): Koha \(version. 3.0\) and NewGenLib \(version. 2.2 beta\)&](https://www.google.co.in/#q=Comparison%20of%20two%20open%20source%20integrated%20Library%20Systems%20(ILS)%3A%20Koha%20(version.%203.0)%20and%20NewGenLib%20(version.%202.2%20beta)&)
6. Helling, J. (2010). Cutting the Proprietary Cord: A Case Study of One Library's Decision to Migrate to an Open Source ILS. Library Review, 59 (9), 702-707. Accessed 1 April 2017A
7. https://www.google.co.in/search?hl=en&site=img&tbm=isch&source=hp&biw=1600&bih=789&q=NewGenLib+Architecture&oq=NewGenLib+Architecture&gs_l=img.1 Accessed 3 April 2017
8. Koha 3.0 at the Royal London Homoeopathic Hospital. Program: Electronic Library and Information Systems, 42 (3), 303-314. Accessed 1 April 2017
9. Koha (2011a), "Home", available at: www.koha.org (Accessed 1 April 2017).
10. Koha (2011b), "About Koha", Koha Library Software Community, available at: <http://kohacommunity.org/plabout> Accessed 03 April 2017.
11. Kushwah, S.S., Gautam, J.N. and Singh, R. (2009), "Migration from CDS/ISIS to KOHA: a case study of data conversion from CCF to MARC 21", International Cataloguing and Bibliographic Control, Vol. 38 No. 1, pp. 6-10. (Accessed 03 April 2017).
12. NewGenLib (2011a), "About us", NewGenLib, available at: www.newgenlib.com
13. (Accessed 01 April 2017).
14. NewGenLib (2011b), "Technology, NewGenLib", available at: www.verussolutions.biz/technology.php (Accessed 03 April 2017).
15. Singh, M., &Sanaman, G. (2012). Open source integrated library management systems Comparative analysis of Koha and NewGenLib. The Electronic Library, 30(6), 809-832. Retrieved April 2, 2017, from https://courseweb.pitt.edu/bbcswebdav/institution/Pitt%20Online/MLIS_Pitt_Online/LIS%202610/Module%202004/msingh_17065317.pdf
16. VENKATARAMA REDDY, C. (2013). COMPARATIVE STUDY OF FREE/OPEN SOURCE INTEGRATED LIBRARY MANAGEMENT SYSTEMS (FOSILMS) WITH REFERENCE TO KOHA, NEWGENLIB AND E-GRANTHALAYA. E-Library Science Research Journal, 1(12), 1-10. Retrieved April 4, 2017, from <http://egranthalaya.nic.in/129.pdf>