

Block

1

LIBRARY AUTOMATION

UNIT 1

Introduction to Library Automation **5**

UNIT 2

Library Automation Processes **50**

UNIT 3

Library Automation – Software Packages **91**

UNIT 4

Library Automation: Application of Open Source Software **141**

Programme Design Committee

Prof. Uma Kanjilal (Chairperson)
Faculty of LIS, SOSS, IGNOU

Prof. B.K.Sen, Retired Scientist
NISCAIR, New Delhi

Prof. K.S. Raghavan, DRTC
Indian Statistical Institute, Bangalore

Prof. Krishan Kumar, Retired Professor
Dept. of LIS, University of Delhi, Delhi

Prof. M.M. Kashyap, Retired Professor
Dept. of LIS, University of Delhi, Delhi

Prof. R.Satyanarayana
Retired Professor, Faculty of LIS, SOSS
IGNOU

Dr. R. Sevukan
(Former Faculty Member) Faculty of LIS
SOSS, IGNOU

Prof. S.B. Ghosh, Retired Professor
Faculty of LIS, SOSS, IGNOU

Prof. T. Viswanathan
Retired Director, NISCAIR, New Delhi

Dr. Zuchamo Yanthan
Faculty of LIS, SOSS, IGNOU

Conveners:

Dr. Jaideep Sharma
Faculty of LIS, SOSS, IGNOU

Prof. Neena Talwar Kanungo
Faculty of LIS, SOSS, IGNOU

Programme Coordinators

Prof. Jaideep Sharma and Prof. Neena Talwar Kanungo

Course Coordinator

Prof. Uma Kanjilal

Course Preparation Team

Unit No(s)
1-4

Unit Writer(s)
Dr. Parthasarathi Mukhopadhyay

Course Editor
Prof. Uma Kanjilal

Print Production

Mr. Manjit Singh
Section Officer (Pub.)
SOSS, IGNOU, New Delhi

Secretarial Assistance

Ms. Sunita Soni
SOSS
IGNOU, New Delhi

Cover Design

Ms. Ruchi Sethi
Web Designer
E Gyankosh, IGNOU

July, 2014 (Second Revised Edition)

© Indira Gandhi National Open University, 2014

ISBN-978-81-266-6776-5

All rights reserved. No part of this work may be reproduced in any form, by mimeograph or any other means, without permission in writing from the Indira Gandhi National Open University.

“The University does not warrant or assume any legal liability or responsibility for the academic content of this course provided by the authors as far as the copyright issues are concerned.”

Further information on Indira Gandhi National Open University courses may be obtained from the University's office at Maidan Garhi, New Delhi-110 068 or visit University's web site <http://www.ignou.ac.in>

Printed and published on behalf of the Indira Gandhi National Open University, New Delhi by the Director, School of Social Sciences.

Laser Typeset by : Tessa Media & Computers, C-206, A.F.E.-II, Okhla, New Delhi

Printed at :

BLOCK 1 LIBRARY AUTOMATION

Introduction

Day to day library functioning covers a series of activities like acquisition and preparation as well as organisation of different types of documents available in varied formats. These activities related to acquisition, technical processing of documents, circulation and maintenance of processed documents are known as housekeeping operations. These activities are time consuming and labour intensive. With the advent of personal computer in 1970s, libraries world-over have been increasingly attempting to automate some of these activities to minimise human interventions and thereby, optimising productivity and creativity of library personnel. Library automation is the generic term that denotes applications of Information Communications Technologies (ICT) for performing manual operations in libraries of any type or size. This block introduces you to the library automation concept and the processes and tools used for the purpose.

This block has four Units.

Unit 1, Introduction to Library Automation covers conceptual views related to library automation and evolution of Integrated Library System (ILS). It also discusses features, advantages, requirements, steps, standards and models of library automation and traces path of progress and future directions in the development of ILS.

Unit 2 on Library Automation Processes discusses typical workflows of library subsystems in the library automation context and analyses housekeeping operations. It also covers requirements, processes and advantages of automating a library and issues related to administration of library automation processes.

Unit 3 on Library Automation –Software Packages deals with evolution and generation of library automation software packages and features and specialties of major commercial and open source software packages. It also covers the trends in library automation software packages.

Unit 4, Library Automation: Application of Open Source Software discusses open source movement and the advantages of using open source software and open standards in library system. It also deals with the emerging concept of open library system.

UNIT 1 INTRODUCTION TO LIBRARY AUTOMATION

Structure

- 1.0 Objectives
- 1.1 Introduction
- 1.2 Evolution of Library Automation
- 1.3 Automated Library Systems
 - 1.3.1 Rationale
 - 1.3.2 Prerequisites and Steps
 - 1.3.3 Procedural Model
 - 1.3.4 Traditional, Automated and Digital: Three Eras of Library Systems
- 1.4 Automated Library System: Standards and Software
 - 1.4.1 Standards
 - 1.4.2 Software
- 1.5 Automated Library System: Global Recommendations
 - 1.5.1 OLE Recommendations
 - 1.5.2 ILS-DI Recommendations
 - 1.5.3 Request for Proposals (RFPs)
- 1.6 Automated Library System: Development of RFP
- 1.7 Automated Library System: Trends and Future
- 1.8 Summary
- 1.9 Answers to Self Check Exercises
- 1.10 Keywords
- 1.11 References and Further Reading

1.0 OBJECTIVES

After going through this Unit, you will be able to:

- understand conceptual views related to library automation and evolution of ILS;
- know features, advantages, requirements, steps, standards and models of library automation; and
- trace the path of progress and future directions in the development of ILS.

1.1 INTRODUCTION

Library services require a series of works like acquiring, preparing and organising documents of different types and available in many formats. The activities related to acquisition of documents, technical processing of acquired documents, circulation and maintenance of processed documents are known as housekeeping operations. In a traditional library system (managed manually) these time consuming, labour intensive activities and routine clerical chores are performed slowly and expensively by library staff. Libraries all over the world, right from 1970s (with the advent of Personal Computer) are increasingly attempting to

automate some of these activities for minimising human clerical routines and thereby optimising productivity and creativity of library staff. Library automation is the generic term that denotes applications of Information Communications Technologies (ICT) for performing manual operations in libraries of any type or size. Library automation process can adopt three routes – i) a piecemeal approach, converting individual operations one at a time (for example installation of Cataloguing module alone to offer OPAC); ii) the process can work towards the integrated system progressively, using a ‘planned installation’ approach (for example installation of Member management module and Circulation modules after the Cataloguing module); and iii) it can go directly for a fully integrated system to cover operations of all subsystems in the library. Therefore, theoretically, a typical library automation may or may not be integrated and may or may not be applied on a Local Area Network (or Intranet). In such automation process, the functions that may be automated are any or all of the followings: acquisition, cataloging, member management, circulation, serials control, inter library lending, and access to online public access catalogue. But the radical development in hardware, software and connectivity along with the reduced costs paved the path for integrated library automation systems (ILS). Presently, library automation processes are integrated systems of a set of interlinked modules responsible for the management of different operational subsystems.

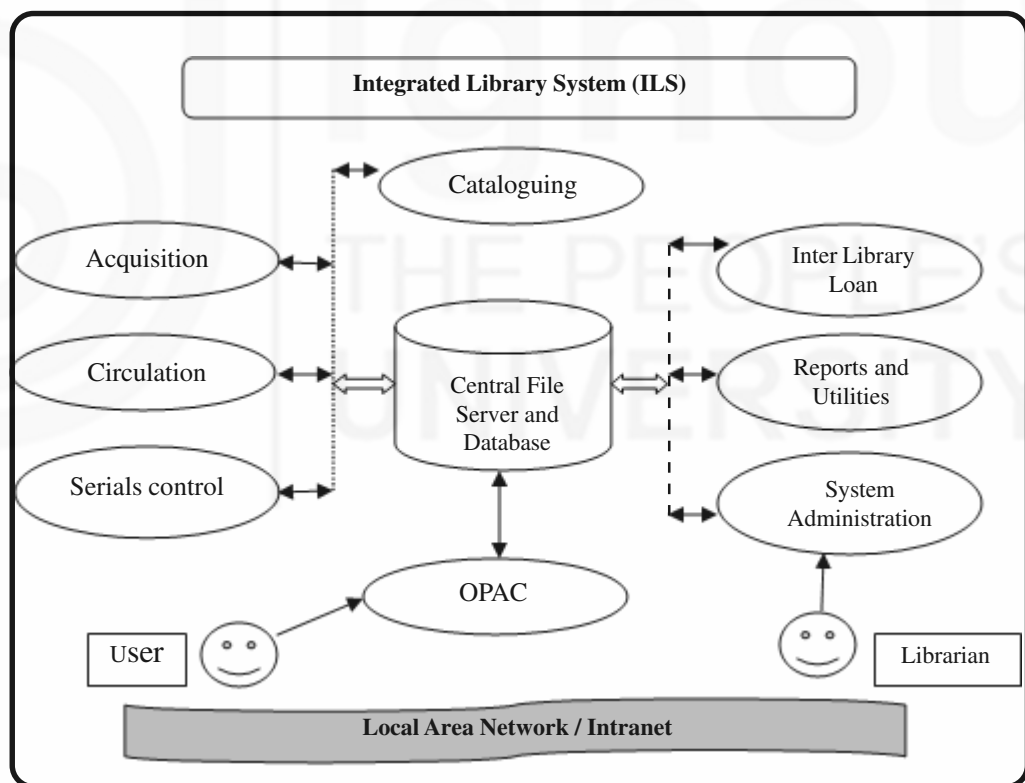


Fig. 1.1: Integrated Library System

Such integrated library automation is also known as Automated Library System. Library Management Software (LMS) forms the core of an automated library system. These LMSs are based on relational database architecture. In such systems files are interlinked so that deletion, addition and other changes in one file automatically activate changes in related files. It means integrated library management system is sharing a common database to perform all the basic functions of a library (see Fig. 1.1). For example, an integrated library system

(ILS) enables the library to link circulation activities with cataloging, serials control, report generation etc. at any given time. It makes use of a file server and clients in a local area network or wide area network (Fig. 1.1). Automated Library Systems now support three broad groups of library activities – i) housekeeping operations; ii) information retrieval; and iii) on-the-fly integration of library materials with open datasets . These are accessible through Local area Network (LAN) or Wide Area Network (WAN) and also over Internet. Modern library automation systems are Web compatible and accessible through Internet, Intranet and Extranet for information retrieval as well as data entry activities. Moreover, automated library systems are now capable to be integrated seamlessly with linked open data (like name authority data, subject access systems etc.), open contents (like book reviews, table-of-contents, cover images etc.) and social networking tools (like Facebook, Twitter etc.) through semantic web technologies and information mashup.

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

1) Define library automation. What are the needs of library automation?

.....
.....
.....
.....
.....

2) What do you mean by integrated library system? Enumerate the features of such systems.

.....
.....
.....
.....
.....
.....

3) Distinguish between library automation and integrated library system.

.....
.....
.....
.....
.....
.....

1.2 EVOLUTION OF LIBRARY AUTOMATION

Library automation has a fascinating history. You will be amazed to know that the whole automation process in our society began with a librarian – Dr. John Shaw Billing (Rayward, 2002). Herman Hollerith, the Census Bureau of USA employee, who invented punched card machinery, attributes the idea to a suggestion by Dr. Billing, the then librarian of Surgeon-General’s Library (now the National Library of Medicine). Hollerith formed the Tabulating Machine Company in 1896, which later became the International Business Machines (IBM) Corporation, one of the largest organisations in computing industry (Mukhopadhyay, 2005). Library professionals initiated application of computers when existing library practices and procedures began to break down under huge bibliographical pressure (also known as information explosion) emerged during late 1950s and early 1960s. Development of low-cost personal computers in 1970s and improved connectivity of 1980s helped establishment of automated library systems mainly in developing blocks of the world. A decade wise analysis of developments in library automation (Mukhopadhyay, 2005) will help you in understanding the rapid upward changes in this domain.

- **Pre-computer era (1950s):** First there was the pre-computer era of unit record equipment.
- **Stand-alone era (1960s):** Then came the off-line computerisation in 1960s and early 1970s.
- **On-line system (1970s):** This was followed by the on-line systems of the 1970s.
- **Micro-computer era (1980s):** The 1980s saw the advent of microcomputers in the form of PCs, emergence of CDROM technology and Local Area Network (LAN).
- **Web era (1990s):** Internet revolution of 1990s paved the path of Web-enabled integrated library systems to support access and operations from anywhere at any time.
- **Open era (2000s):** Emergence of open library systems powered by open source software, open standards and on-the-fly integration with open data and open contents.

Although library automation began in 1930s (1936 to be exact) when punched card equipment was implemented for use in library circulation and acquisitions, the real library automation started in early 1970s with the use of low-cost PCs and locally developed software to automate library house-keeping operations. The whole phase of development i.e., 1970 to till date may be grouped into **five** distinct periods:

The First Automation Age: This era was characterised by computerisation of library operations by utilising either commercial automation package or software developed in-house. The development of shared copy–cataloguing system (also known as distributed cataloguing) was another significant achievement of this phase that utilised computer and communication technologies for collaboration and cooperation within the library community.

The Second Automation Age: This period of library automation was characterised by the rise of public access i.e., the arrival of OPAC as a replacement for the traditional card catalogue. This period also witnessed major developments in online access to abstracting and indexing databases, union catalogues, resource sharing networks and library consortia.

The Third Automation Age: This era was characterised by the full text access to electronic documents over high-speed communication channels. Digital media archiving was an important element of library automation in this period. The advent of Internet as global publishing platform and largest repository of information bearing objects revolutionised the ways and means of delivering library services. As a result, Web-centric library automation was norm of the time.

The Fourth Automation Age: It is also known as ‘networked information revolution’ era. This era supports a vast constellation of digital contents and services that are accessible through the network at anytime, from anyplace, can be used and reused, navigated, integrated and tailored to the needs and objectives of each user. Digital libraries, multimedia databases and virtual libraries are major achievements in this era. Most of the automated library systems in our country are in between the third age and fourth age of library automation.

The Fifth Automation Age: The next generation library automation uses interactive, collaborative and participative platform for developing user-oriented library services with the help of Web 2.0 tools and services. This era of library automation also characterised by the capabilities to on-the-fly integration of Linked Open Data (LOD) with local library resources and operations (for example - utilisation of global dataset VIAF (Virtual Internet Authority File) in managing name authority file of local library catalogue, and integration social networking tool such as Facebook with OPAC to post Like against a library document). Cloud based library management and Web-scale library management are norms of the fifth automation age.

Now you know the phases of development in library automation for almost the last forty-five years. However, a time line for the development of ground-breaking events in library automation can be a handy tool for you to grab the path of development.

1936-59 : Major events of this time period were as follows: Introduction of punched card for circulation control in library; Use of IBM 402, 403 and 407 for manipulating, analysis, sorting and retrieval of data; Vannevar Bush introduced the concept of ‘Memex’ in 1945.

1960-69 : Major breakthroughs of this period were as follows - Use of general-purpose computers that became widely available in the 1960s; H.P. Luhn, in 1961, used a computer to produce the “Keyword in Context” or KWIC index for articles appearing in Chemical Abstracts; Project “MEDLARS” started in 1961 that applied computer in measuring efficiencies of information retrieval systems; Computerised circulation system first appeared in 1962; Project ‘Intrex’ (aimed to provide a design for evolution of a large university library into a new information transfer system) started in 1965;

Project MARC, initiative by Library of Congress to provide a format for machine readable cataloguing data, started in 1965; Introduction of online interactive computer system in place of off-line batch processing systems began in mid 1960s; Initiation of projects like BALLOTS by Stanford University and MAC by M.I.T. These developments deal with the possibility of a new horizon for the library operations and services.

- 1970-79** : Important achievements of this time period – Minicomputers were introduced to automate circulation and books were bar-coded; Computer based acquisition systems were introduced to procure books and serials; ISBDs started appearing from 1971; OCLC established in 1971 to facilitate library cooperation and to reduce costs of processing works; ISO-2709 was developed in 1973 as the standard for data exchange format; OCLC started development of Worldcat in 1975 (Worldcat now contains 8 billion cataloguing records and considered as the largest bibliographic database in the world); Library networks started appearing all over the world.
- 1980-89** : Important events of the decade – Shared copy-cataloguing systems by using computer and communication technologies were established as a norm in 1980s; Remote access to on-line databases became a reality; Appearance of CDROM databases on indexing and abstracting journals started in early 1980s; Library automation packages initiated shifting towards relational architecture; Integrated automation packages began appearing in mid 1980s along with bar-coded circulation system; OPAC became very popular in this decade and made available on campus wide LAN for accessing;
- 1990-99** : Major events were as follows – Library automation packages started upgrading from client server architecture to web architecture; Large scale developments took place in the area of resource sharing, union catalogue and computerised inter library loan. Release of Z39.50 protocol in 1995 to share bibliographical information and to overcome the problems of database searching with many search languages; Formation of collective purchasing consortia started that can negotiate prices for all members of the consortium; Emergence of multimedia databases; Retrieval achieved maturity with an array of search operators; Emergence of Web-based library services; Release of Dublin Core Metadata Standard in 1995; Web-OPAC began appearing for almost all automated libraries; Conversion and digitisation of print contents into electronic format started in a big way; Full text access to information resources over Internet started against IP authentication; Integrated access interface emerged to act as one-stop access interface; IFLA introduced FRBR as a conceptual data model for bibliographical databases in 1998; Introduction and development of Eprint archives and digital libraries; MARC 21 family of standards (Bibliographic format, Authority format, Holdings format, Classification format and Community information format) released in 1999; RFID based inventory management and smart card based user access to on-line library services; OAI/PMH standard developed for metadata

harvesting and initiatives started to make LMSs compatible with this standard;

2000-14 : Remarkable achievements of the present era are – Development of matured and globally competitive open source LMSs; Establishment of open standards like SRW, SRU, MARC-XML and development of standards for different sub-domains of library automation like NCIP (NISO Circulation Interchange Protocol); Applications of Web 2.0 tools and techniques in automated library system; Development of interactive OPAC to support user tagging, rating and comments; Improvements in searching and browsing with a set of newly developed search operators like Fuzzy search, weight-term search etc.; Application of semantic web technologies in LMSs to support integration of Linked Open Data (LOD) with library operations and services.

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

4) What are the five ages of library automation? Explain.

.....
.....
.....
.....
.....

5) Show a decade-wise growth of library automation technologies from 1970 to 2010.

.....
.....
.....
.....
.....

6) Enumerate the major technology breakthroughs in library automation since the introduction of PCs

.....
.....
.....
.....
.....
.....

1.3 AUTOMATED LIBRARY SYSTEMS

The decade-wise development of library automation shows that the effects of ICT on libraries and information centers. The path of developments is characterised by three fundamental factors:

- **Mechanisation** – doing what we are already doing though more efficiently;
- **Innovation** – experimenting with new capabilities i.e., introduction of new services and improvement of existing services through the use of ICT; and
- **Transformation** – fundamentally altering the nature of the library operations and services through capabilities extended by ICT.

This section of the unit discusses how library automation – i) helps in mechanisation of library operations; ii) supports innovation in library operations and user-centric services; and iii) promotes transformation in organisation of information resources and dissemination of services. The discussion covers reasons for library automation, requirements for library automation, steps for developing an effective automated library system, model of library automation and how does an automated library system differs from a digital library system.

1.3.1 Rationale

Society is changing and so are the library users. There are many reasons of the ongoing changes but the most visible one is the impact of ICT on society. As a result libraries need to change to keep pace with these societal changes. It is also required for libraries to get continued support – political and financial from parent organisation as well as from government. However, the rationale for library automation may be summarised as below:

- Automation of library housekeeping operations is considered as an especially critical area from which future benefits will emerge. It means that if a library is not automated it cannot take advantages contributed by ICT such as digitisation, web-enabled library system, use of linked open data, remote management of library, interactive user services etc. ;
- Increased operational efficiencies are achieved through library automation;
- Automation of housekeeping operations relieves professional staff from routine clerical chores and thus make them available for end-users services;
- Betterment of library services in terms of speed, quality and efficiencies;
- Automation may create interactive, collaborative and participative platform for user-centric library services;
- Supports improvement of existing services and introduction of new services;
- Makes library free from two fundamental barriers of information access – time and space. A web-enabled library system allows access at anytime from anywhere and by anyone;
- Automated library system with the capability to generate extensive reports and statistics extends support as decision-making tool for library managers and policy makers;

- An automated library system is able to contribute to resource-sharing networks and on the other hand may take the benefits of resources and services of library networks; and
- Better management of staff, physical resources, financial resources and wider dissemination of information products and services.

But at the same time one should remember that library automation requires huge initial investments in developing network infrastructure, procuring hardware, buying/customising software, retraining of staff or in some cases recruitment of technical staff. It may lead to chaos in resource organisation and dislocations in user services during transformation phase. Initially users and staff may feel uncomfortable, but with the passing of time the benefits of library automation will be realised by all stakeholders. As ICT has spillover effect, an automated library system, after initial teething problem, soon begins to search other areas for extension of bibliographic services.

1.3.2 Prerequisites and Steps

After covering the previous sections, you now know that library automation is a challenging task. We need to know what are the requirements, what are the strength and weakness of the library to be automated, how to prepare the proposal and budget, how to select hardware and software, who requires to be trained, how to plan implementation of software, how to handle retro-conversion (RECON or retro-conversion is transferring old bibliographic resource into machine-readable forms in the software system) and finally how to manage the transformation process. The prerequisites of library automation may be studied under the following heads:

System-level requirements

The system level requirements include hardware, network and storage. These components build the necessary infrastructure for implementation of integrated library system. The infrastructural requirements for library automation may vary from simple (inexpensive) to very complex (expensive) depending on factors like functional requirements, software architecture, support for global domain-specific standards, interoperability requirements, number of library sites or branches, number of records to be managed, number of users to be supported, requirements for managing multi-lingual records, retrieval features, federated search capabilities etc. The infrastructural requirements is very high for an automated library system that aims to serve users through Web-OPAC (requires server, IP address and domain name), to support distributed cataloguing (to serve bibliographic data as Z39.50 server), and to take the advantages of cloud computing. Generally hardware level requirements include Server (a centralised mainframe or minicomputer architecture) and client PCs (low-end computers for data entry and end-user searching). Storage devices are required to store bibliographic data (full-text data in case of digital media archiving). Network is required to link server with storage devices and client PCs.

Software-level requirements

An integrated library system is managed by integrated library management software (LMS). LMS manages different functional modules (for different sub-systems of a library) on the basis of a common database (with different tables for

different modules in relational model). Such a LMS supports seamless exchange of data (bibliographic data, financial data, member data etc.) between the different subsystems of an integrated library system. The essential features that should be supported by an ILS (or LMS) must be known before selection of software. These are applicable to all modules of any modern LMS and should include but not limited to the following features:

- The LMS must be fully integrated, using a single, common database for all operations and a common operator interface across all modules;
- The LMS should have capability of supporting multiple branches or independent libraries, with one central computer configuration sharing a common database;
- The LMS must allow unlimited number of records, users and organisation-specific parameters (e.g. loan period rules, fine calculation criteria, hold parameters etc.);
- The package should include following fully developed and operational facilities at multiple customer sites:
 - Bibliographic and inventory control
 - Authority control
 - Public access catalogue
 - Web catalogue interface
 - Information gateway (telnet, www, Z39.50, proxy server)
 - Acquisition management
 - Serials control
 - Electronic data interchange (EDI)
 - Reservation and materials booking
 - Circulation control
 - Customised generation of reports and usage statistics
 - One step administrative parameters setting
 - Z39.50 sever (minimum version 3 and bath profile level complaint) and Z39.50 client
 - Z39.50 copy cataloguing client
 - Marc 21 bibliographic and authority record import/export utility
 - Outreach services
 - Digital media archive system and Multimedia
 - Fund accounting , Bills and fines
 - Inter library loan
 - Interoperability and crosswalk
 - Web 2.0 supports
- LMS must provide continuous backup in suitable media (as per the choice of libraries) so that all transactions can be recovered to the point of failure;
- LMS must be compliant with the following standards (see section 1.4.1 for a list of standards):
 - Z39.50 information interchange format
 - MARC 21, UNCODE (UTF-8 OR UTF-16)
 - Z39.71 holdings statements
 - Z39.50 information retrieval service (client and server version3)
 - EDIFACT (EDI standard)
 - IEEE 802.2 and 802.3 Ethernet
 - HTTP, TCP/IP, Telnet, FTP, SMTP

- The LMS should be based on web-centric architecture and extend support for a range of multi-user and multitasking operating systems and RDBMSs;
- The LMS must be compliant with UNICODE standard for multilingual support and RFID for inventory management and self-issue/return facility;
- Vendor/Developing group should provide training to enable library staff to become familiar with system functions and operation, should supply full and current system documentation in hard copy and in machine-readable form suitable for online distribution and the LMS should include extensive online help for users and staff;
- LMS must support multiple hardware architecture in terms of server, network infrastructure, PC-workstations and peripheral devices;
- LMS must be supported with regular maintenance and on-call service, periodical software upgrades, continuous R & D, trouble-shooting of third-party software such as database package and the library automation package, distribution of problem fixes/patches and emergency services for system failures and disaster recoveries;
- The package must provide security to prevent accidental or unauthorised modification of records through the establishment of access privileges unique to each user on the system and restriction of specific functions to specific users;
- LMS should provide graphical user interface including, but not limited to extensive online help, user self-service and personalisation features. The system should be supported with PC-based alternative that will allow circulation to continue in the event of system failure, communication failure and downtime required for maintenance;
- LMS must be compliant with web 2.0 features to support interactive, collaborative and participative platform; and
- LMS should be updated regularly to take advantages of cutting-edge technologies like cloud computing, linked open data and semantic web.

Steps of library automation

Library automation is a complex process and should be planned astutely. The complete process of library automation may be divided into following steps:

- Software selection
- Hardware selection
- Site preparation
- General training
- Customisation
- Defining procedures for
 - Bibliographical data entry
 - Administrative data entry
 - Financial data entry
- Commissioning

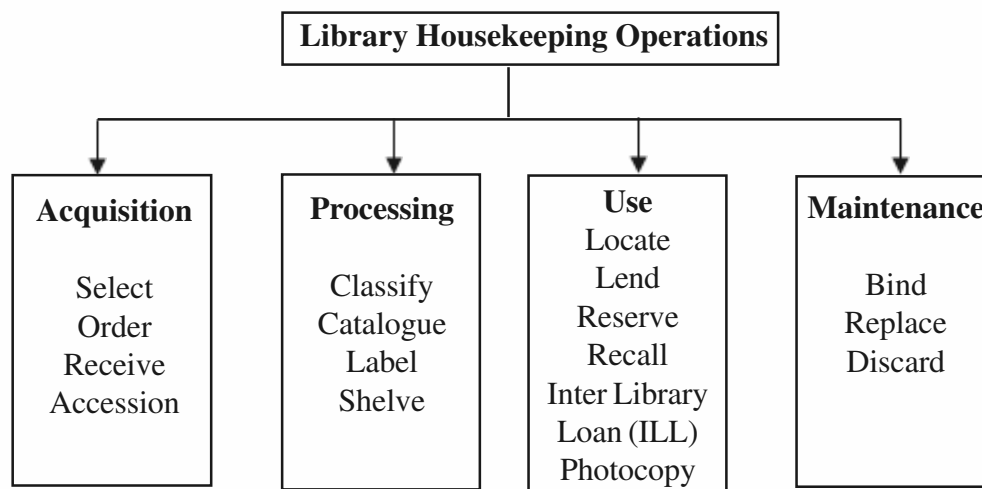
It is quite obvious that implementation of the above steps in library automation requires background study or analysis of the library system (see section 1.3.3 for system analysis process). It is a precondition to utilise library automation package for effective results. A library will not be able to take full advantages of automation until and unless its manual functions are perfect and justified. Therefore, the procedures and tasks followed in different sections should be analysed in terms of :

- Special features of the library system
- Local variations (their validity and usefulness)
- Limitations of the existing system
- Nature and objectives of library
- Total number of collection and nature of collection
- Per year acquisition and procedures followed for acquisition
- Per year subscription of serials and number of back-volumes
- Number of users and their categories
- Per day transactions (issue/return/reservation)
- Availability of multilingual documents
- Need of information services (CAS/SDI etc.)
- Future plan (in terms of networking and consortia, digitisation, cloud computing)
- Available manpower (computer literate staff, retraining of staff, recruitment of technical staff).

This is an illustrative list of factors to be considered during the process of library automation. In reality a library needs to prepare a comprehensive list of such factors for effective utilisation of the automated library system.

1.3.3 Procedural Model

Library automation aims to support workflows of a library in an integrated setup. It means different subsystems of a library (like acquisition, cataloguing, circulation, serials control, OPAC etc.) need to be supported by an ILS. Therefore, to understand library automation we need to understand first the library workflows. In fact an ILS (or LMS) automates the workflows of a library system. Most of the LMSs are based on a model called **procedural model of library automation** (first proposed by P.A. Thomas in an analytical study of library automation conducted by the then ASLIB). The model proposes that a library system has mainly two subsystems – administrative subsystem and operational subsystem. We cannot automate the process of administration but if we can automate operational subsystem, it may help administrative subsystem in taking right decision at the right time. In fact automation of operational subsystem may provide a wholesome MIS (Management Information System) to library managers. Operational subsystem comprises mainly four subsystems for performing housekeeping jobs through eighteen procedures. These procedures under each and every operational subsystem require one or more of six possible activities. There are fifteen basic tasks for performing procedures and activities. In short, procedural model of library automation proposes two basic subsystems, four operational subsystems, three levels, eighteen procedures, six activities and fifteen basic tasks as library workflow irrespective of the type and size of libraries and it advocates automation of the procedures, activities and tasks through different modules of an ILS.



The functions and activities of one division is entirely different from other divisions but they are closely related and the combined efforts lead towards the better library services. It is quite clear now that libraries are complex systems that include subsystems and components. The main two subsystems are operational subsystem and administrative subsystem. Library housekeeping operations are part of the operational subsystem. As per the analytical study of ASLIB (Association of Information Managers, UK), the operational subsystem may be divided into four further subdivisions namely Acquisition, Processing, Use and Maintenance. Within each of these divisions there are a number of procedures and within each procedure there is one or more of six possible activities. The tabular presentation of the place and scope of housekeeping operations related to different subsystems in a library system (as per the procedural model) is given below:

Table 1.1: Procedural model of library automation (Source: Mukhopadhyay, 2005)

System	Subsystems	Operational Subsystems	Procedures	Activities (Common to all Procedures)
Library System	Operational Subsystem	Acquisition	Select Order Receive Accession	Initiate (To commence a procedure)
		Processing	Classify Catalogue Label Shelve	Authorise (To approve a procedure) Activate (To implement a procedure through appropriate action)
		Use	Locate List Lend/Issue Reserve Recall/Return ILL (Inter Library Loan) Photocopy	Record (To record what action has been taken) Report (To notify staff or user about the action taken)
	Administrative Subsystem	Maintenance	Bind Replace Discard	Cancel (To stop a procedure or undoing an action)

In considering libraries from one general organisational point of view, the analysis of housekeeping system is useful for automation of a library. It is a prerequisite to design and use library management software and to communicate with software vendors and programmers. A close analysis of the operations involved in library housekeeping provides us three hierarchical levels – **procedures, activities and tasks**.

Procedures and Activities

The eighteen procedures listed in the previous paragraph are common to libraries of different types. The design and use of an automated library housekeeping system requires the analysis of all these procedures into their atomic structure. It will help to understand and implement mechanised housekeeping operations in an automated environment. The procedures under each and every operational subsystem have been analysed by P.A. Thomas in terms of six possible activities – *initiate, authorise, activate, record, report and cancel*. All of these activities may not be involved in every procedure. There are one or more six possible activities against each procedure. The six common activities are defined as:

- **Initiate** – That which makes it apparent that a procedure should be commenced.
- **Authorise** – In some cases, the decision to carry out a certain procedure must be approved before any further action is taken.
- **Activate** – When a procedure is known to be necessary and in some cases approved, it is usually implemented by taking appropriate actions.
- **Record** – The function that states or records what action has been taken.
- **Report** – To notify library staff or user that an action has been taken.
- **Cancel** – To stop a procedure, in particular the aspect of revoking or undoing an action.

Tasks

The third level in the hierarchy is concerned with ‘tasks’ within an activity under each procedure. Task means a related group of operations carried out to perform a particular kind of job. In an automated library system a task is the collective functions of the elements for the accomplishment of the module at the next higher level. Tasks within each activity, just as the activities themselves, may not all be necessary to each procedure. Most of the works in the operational subsystems of a library include making or using discrete records with bibliographic and administrative information referring to one particular document. In this context, ASLIB defined a set of fifteen tasks for the basic procedures. These are – *pass, receive, discard, place, remove, search, duplicate, attach, separate, move, sort*. Such tasks are supported by other four element tasks namely *read, verify, enter and decide*.

1.3.4 Traditional, Automated and Digital: Three Eras of Library Systems

The application of ICT tools in the form of hardware, software and network changed conventional library system considerably right from 1970s. Now, we have an array of modern information handling systems named as computerised library system, automated library system, electronic library system, digital library

system and virtual library system. However, we are going to restrict discussion to two stable modern library systems – automated library system and digital library system. You already know what an automated library system is. Now question comes what is a digital library system and how does it differ from automated library system? Digital libraries are major application entities of Internet and Web technologies. These are considered as next generation library services. In simple words, Digital libraries are managed collections of digital objects. These entities enable the creation, organisation, maintenance, management, access to, sharing and preservation of digital knowledge bearing objects or document collections. Digital libraries are being created today by many institutes and agencies for different target groups and in diverse fields like agriculture, cultural heritage, education, health, governance, science, social sciences, social development, etc. In its final shape a digital library system will be a single-window federated search interface for a diverse range of information resources collected or optimised by a library system.

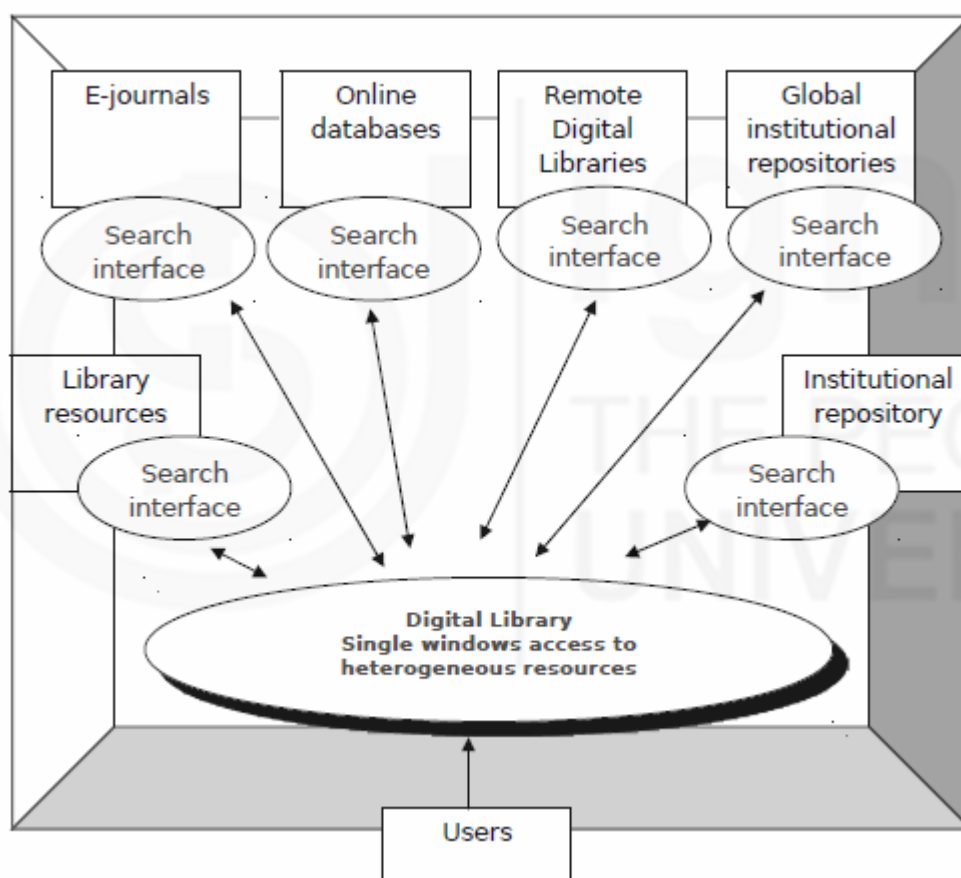


Fig. 1.2: Digital library system

Availability of free/libre open source software (FLOSS) based digital library software packages, application of open standards and sharing of domain knowledge through Wiki, Blogs etc. help in designing Digital libraries even in developing block of the world. Now the question comes that what are the advantages of digital libraries? There are some obvious benefits of Digital libraries over the automated library systems. Some of the key benefits of digital libraries are:

- Traditional libraries are associated with the organisation and provision of access to physical material like print-on-paper publications.

- Automated library systems are providing improved access to their collections but online access facilities are limited to the computerised library catalogue (OPAC).
- Digital libraries differ significantly from such libraries because these entities facilitate online access to and work with digital versions of full text resources in multimedia-driven environment.

Library automation activities address two major issues – library housekeeping operations and access to library resources. An automated library system has cataloguing data in digital format but source documents are mostly available in print formats. In a digital library setup both metadata (document description data) and documents are available in digital format. The other major differences are:

Table 2: Automated Vs. Digital library systems

Automated library system	Digital library system
Only metadata (cataloguing data) is finely searchable	Both metadata set and full-text resources are finely searchable
Provides document description data set, not documents.	Provides document description data set and source documents
Based on Z39.50 standard for cross-system catalogue search/retrieve	Based on OAI/PMH protocol for metadata harvesting
Supports standard bibliographic formats (MARC 21, CCF) for document description	Supports generic and domain-specific metadata schemas (e.g. Dublin Core, LOM, GILS etc.) for resource description
Processes global resources for local users	Processes global and local resources for local and global users
Generally follows centralised processing – distributed access architecture	Generally follows distributed processing – distributed access architecture

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

7) What is the rationale for integrated library system?

.....

.....

.....

.....

.....

.....

.....

.....

8) Discuss the software-level prerequisites for an integrated library system.

.....

.....

.....

.....

.....

9) What is Procedural model of library automation? Illustrate.

.....

.....

.....

.....

.....

10) What is a digital library system? How does it differ from automated library system?

.....

.....

.....

.....

.....

1.4 AUTOMATED LIBRARY SYSTEM: STANDARDS AND SOFTWARE

Integrated library systems depend on two core components – standards and software architecture. Libraries are now operating in a distributed networked environment, where standards are essential for efficiency and interoperability. Order, collaboration and interoperability are three most important prerequisites for effective application of ICT in library operations and services. Library automation is no exception. Therefore, we need to know about standards for developing automated library systems and LMSs should follow strictly different global and national standards prescribed for the domain of library automation.

1.4.1 Standards

Standards are developed by general agreement among stakeholders of an area of human activity. These are used by professional like scientists, engineers, technologists etc. for their respective domain of activities. We often use the terms *standards, guidelines and specifications* synonymously. A “guideline” is a statement of policy by a person or group having authority over an activity. A “standard” is formulated by agreement and applicable to an array of levels – corporate, national, or international. A “specification” is a concise statement of

the requirement for a material, process, method, procedure or service. Standards are frequently updated, modified or revised to keep pace with the technological changes and practical requirements (Withers, 1970). ANSI (American National Standards Institute) defined a standard as a specification accepted by recognised authority as the most practical and appropriate current solution of a recurring problem. IEC Guide 2:2004 of ISO (International Standards Organisation) defines a standard as a document, established by consensus and approved by a recognised body, that provides, for common and repeated use, rules, guidelines or characteristics for activities or their results, aimed at the achievement of the optimum degree of order in a given context. Standards perform important roles in the development of integrated library systems in view of the followings:

- to act as the pattern of an ideal;
- to set a model procedure;
- to achieve interoperability in heterogeneous environment;
- to establish measure for appraisal;
- to act as stimulus for future development and importance; and
- to help as an instrument to assist decision and action.

Standards are mainly developed by Standards Development Organisations (SDOs). An SDO is any entity whose primary activities are developing, coordinating, promulgating, revising, amending, reissuing, interpreting, or otherwise maintaining standards. SDOs are generally grouped by two parameters – geographic designation (e.g. international, regional, national) and organisational authority (e.g. governmental, quasi-governmental or non-governmental entities). Library professionals are generally interested in the library standards developed by their national standard organisations (e.g. BIS – Bureau of Indian Standards in India) and library standards developed by ISO (International Standards Organisations), NISO (National Information Standards Organisation, US) and BSI (British Standards Institute, UK). The library standards developed by NISO are American national standards but in many cases these standards are used by libraries/related organisations across the globe (e.g. Z39.50). These SDOs develop standards in the domain of library services through designated committees and sub-committees. The committee IDT/2 is entrusted by BSI (<http://www.bsi-global.com/>) for Information and Documentation. There are mainly three American National Standards Committees under NISO that develop standards affecting libraries, information services and publishing (www.niso.org). These are X3 (Information Processing Systems); PH5 (Micrographic Reproduction); Z85 (Standardisation of Library Supplies and Equipment); and Z39 (Library and Information Sciences and Related Publishing Practices). Of these, Z39 has developed more standards directly related to LIS fields than others. TC 46 committee of ISO (www.iso.org/iso/) is responsible for standardisation of practices relating to libraries, documentation and information centres, publishing, archives, records management, museum documentation, indexing and abstracting services, and information science. The secretariat of TC 46 is in France (AFNOR - Association française de normalisation). It works through three working groups (WG), four sub committees (SC) and one coordinating group (CG). In BIS, India, MSD 5 (www.bis.org.in) is the Sectional Committee for Documentation and Information.

Although it is difficult to list all the standards related to automated library systems, we may go for listing a set of minimum standards that need to be supported by an ILS/LMS to remain globally competitive and interoperable. These are:

- ISO – 2709 for bibliographic data interoperability;
- Standard bibliographic formats compliant with ISO - 2709 (e.g. MARC 21, UNIMARC, CCF/B);
- Z39.50 protocol standard for distributed cataloguing;
- Z39.71 standard for holdings statements;
- BS ISO 9735-9:2002 Electronic data interchange for administration, commerce and transport (EDIFACT);
- Z39.83-1 (NISO Circulation Interchange Part 1: Protocol (NCIP));
- Z39.83-2 (NISO Circulation Interchange Part 2: Protocol (NCIP));
- ISO/CD 28560-1 (Information and documentation — Data model for use of radio frequency; identifier (RFID) in libraries — Part 1: General requirements and data elements);
- ISO/CD 28560-2 (Information and documentation — Data model for use of radio frequency; identifier (RFID) in libraries — Part 2: Encoding based on ISO/IEC 15962); and
- ISO/CD 28560-3 (Information and documentation — Data model for use of radio frequency identifier (RFID) in libraries — Part 3: Fixed length encoding); and
- ISO/IEC 10646: 2003 (Universal Multiple-Octet Character Set or UCS).

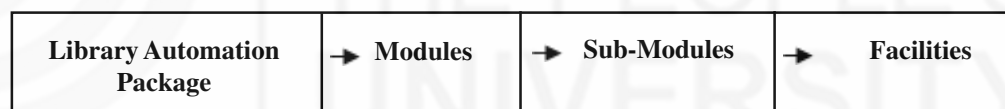
Apart from these formal standards (de jure standards), there are a few specifications (may be considered as de facto standards) in the domain of library services, which are widely in use across different library systems in different countries. Most of these internationally agreed upon informal standards are developed by national libraries (e.g. Library of Congress) and library associations (e.g. ALA, IFLA etc.). Some of these very important non-formal standards are –

- MARCXML – MARC 21 data in an XML structure (developed by Library of Congress - <http://www.loc.gov/standards/marcxml/>) acting as base standard for bibliographic data export/import in place of ISO-2709;
- MODS (Metadata Object Description Standard) – XML markup for selected metadata from existing MARC 21 records as well as original resource description (developed by Library of Congress – <http://www.loc.gov/standards/mods/>);
- MADS (Metadata Authority Description Standard) – XML markup for selected authority data from MARC21 records as well as original authority data (developed by Library of Congress – <http://www.loc.gov/standards/mads/>);
- METS (Metadata Encoding & Transmission Standard) – Structure for encoding descriptive, administrative, and structural metadata (developed by Library of Congress -<http://www.loc.gov/mets/>);

- PREMIS (Preservation Metadata) – A data dictionary and supporting XML schemas for core preservation metadata needed to support the long-term preservation of digital materials. (developed by Library of Congress – <http://www.loc.gov/standards/premis/>);
- SRU/SRW (Search and Retrieve URL/Web Service) – Web services for search and retrieval based on Z39.50 (developed by Library of Congress - semantics <http://www.loc.gov/standards/sru/>); and
- OAI/PMH Version 2.0 – Open Archive Initiative/Protocol for Metadata Harvesting (developed by Open Archive Initiative).

1.4.2 Software

You already know that library management software forms the core part of integrated library automation. You also know what are the prerequisites for an ILS, what are the standards that need to be supported by ILS, and how procedural model of library automation is guiding development of ILS all over the world. The rapid development in utility of hardware, software and connectivity along with the reduced costs paved the path for integrated library automation systems. Current library automation software also known as Library Management Software (LMSs) are integrated systems of a set of related modules responsible for the management of different operational subsystems. These LMSs are based on relational database architecture. Most of the LMSs are presently based on procedural model of library automation and follow a modular approach to perform the tasks related to housekeeping operations. Generally, the whole package is divided in **modules** for each operational subsystem. Modules are divided into **sub modules** and each sub module supports various **facilities** to carry out tasks related to the procedures.



For example, the SOUL package library automation software developed by INFLIBNET, Ahmadabad) includes six modules of which four are for operational subsystems. The other two, namely administration and OPAC are meant for setting up various administrative parameters and searching and retrieving the library resources respectively. Another example may be cited from KOHA – an open source library management software, developed by Horowhenua Library Trust (Katipo team), Newzealand and running at libraries all over the world. It includes one common module for acquisition and cataloguing and other five modules are related with circulation, OPAC, administration etc. A typical LMS supports selection, ordering, acquisition, processing, circulation, serials control, dissemination of information services and also extend help in library administration, planning & decision making process as a management tool. The individual tasks carried out by an ILS under each prime functional subsystems may be identified as below (see Unit 2 in this block for a detail discussion on housekeeping activities):

Ordering and Acquisition

- Ordering
- Receipting

- Claiming
- Vendor database management
- Budgeting and Fund accounting
- Currency conversion
- Suggestions (from users) management
- Enquiries (order status, receiving status)
- Accessioning (in MARC 21 format)
- Bill processing
- Payment
- Reports and Statistics.

Cataloguing

- Standard formats support
- Authority control (in MARC 21 authority format)
- Integration with Linked Open Data (LOD)
- Unicode-compliant multilingual data processing
- Retrieval with sophisticated search operators
- Integration with virtual keyboard for multilingual searching
- Shared cataloguing
- Z39.50 based copy cataloguing
- Output generation and holdings information
- User services (interactive and participative).

Access Services

- Online access
- Public access interface (OPAC)
- Web access and Remote access
- Social-network enabled OPAC
- Gateway services.

Circulation Control

- Setting of user privileges
- Circulation rules
- Issue, return and renewal
- Reservation (user-driven)
- Fine calculation
- User management
- Reminders and recalls
- Enquiries (about item, borrower, reservation)
- Reminders and notices
- Reports and statistics and patron self services.

Serials Control

- Order placement and renewal of subscriptions
- Kardex management
- Receiving and claiming
- Binding control
- Fund accounting
- Cataloguing of serials
- Enquiries (arrival of serials issues)
- Reports and statistics.

MIS

- Reports and statistics
- Analysis of statistics
- Usage statistics (compliant with COUNTER).

Inter Library Loan (ILL)

- ILL protocol
- ILL management.

Outreach Services

- Community information services
- Social-networking support
- Library blog
- Online help for users.

Digital Media Archiving

- 1) Full-text search
- 2) Support for media formats
- 3) Federated search facilities.

System Administration

- Privileges control
- Branch management
- Backup and restoration
- System configuration.

A library may procure commercially available ILS or may opt for implementing an open source ILS. But the above-mentioned basic tasks of an ILS are common to all types of ILSs or LMSs.

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

11) What is a standard? Why an ILS should support global standards? List the standards required for a globally competitive ILS.

.....
.....
.....
.....
.....

12) Discuss the typical tasks performed by an integrated library system.

.....
.....
.....
.....
.....

1.5 AUTOMATED LIBRARY SYSTEM: GLOBAL RECOMMENDATIONS

Libraries of developed countries started taking benefits of ICT through library automation during mid-seventies. Libraries in developing block of the world realised advantages of library automation in early eighties and the process is still going on. But the socio-economic and socio-technical environments within which these libraries operate are changing more rapidly than libraries (in developing block of the world) are changing to meet it. However, in general we can say that present library systems are outgrowing their traditional organisation and discovery tools. Almost all the basic library activities and services are now maintained in an Integrated Library System (ILS) that manages acquisitions, cataloging, circulation, reporting, resource discovery and automatic alerting services. With the advent of socio-technical changes all over the world users expectations have expanded to demand more services in an interactive, quicker and easy way. In many cases, such demands go beyond the scope of a typical ILS. Users now want to find, locate, navigate and obtain resources available in his/her library, at nearby institutions and from open access public domain through a single-window search interface seamlessly. They also want full-text search facility from a single-window federated search interface and when they do find something of interest, they expect to use the library's services for obtaining resources from wherever possible. This situation calls for a set of global recommendations in developing new generation ILS. Such global standards are also required to act as pathfinders for library professionals as well for ILS developers. There are three such sources that can guide us in shaping integrated library systems in view of the future requirements – 1) Open Library Environment (OLE) project recommendations;

2) Digital Library Federation (DLF) - ILS Task Group (ILS-DI) recommendations; and 3) study of Request for Proposals developed by different libraries.

1.5.1 OLE Recommendations

Open Library Environment project (OLE project - <http://oleproject.org>) or the OLE project, funded by Andrew W. Mellon Foundation and participated by more than 300 libraries, started with following objectives – i) to analyse library business processes; ii) to define a next-generation library technology platform; iii) to design Service Oriented Architecture (SOA) for library software; and iv) to frame a community-source model of development and governance. The principal aim of OLE project is cost-effective integration of library management with other institutional systems. The OLE project published the Enterprise Resource Planning (ERP) based Abstract Reference Model (<http://oleproject.org/overview/ole-reference-model>) in 2009. This model shows the relationship between OLE middleware, OLE components, entities, and third-party components, such as Identity Management, Institutional Repositories, and Course Management Systems. As a whole, the OLE framework for future library system is characterised by – 1) Flexibility (Supports for wide range of resources; accessed by a wide range of customers in a variety of contexts); 2) Community ownership (Advocates systems that are designed, built, owned, and governed by and for the library community on an open source licensing basis); 3) Service Orientation (Prescribes technology-neutral service-oriented framework that ensures the interoperability of library systems); 4) Enterprise-Level Integration (Facilitates integration with other enterprise systems such as research support, student information, human resources, identity management, fiscal control, and repository and content management); 5) Efficiency (Provides a modular application infrastructure that integrates with new and existing academic and research technologies); and 6) Sustainability (Creates a reliable and robust framework to identify, document, innovate, develop, maintain, and review the software necessary to further the operation and mission of libraries). See Unit 3 in this Block for a summary of OLE recommendations. The Open Library Environment Project Final Report is available at <http://oleproject.org/final-ole-project-report/>.

1.5.2 ILS-DI Recommendations

In regards to the integrated systems of libraries (automation and digitisation), DLF ILS Discovery Internet Task Group (ILS-DI) Technical Recommendation is playing a pivotal role. These recommendations are framed in view of the variations in user demands and developments in ICT. As per these recommendations library software systems should – i) improve discovery and use of library resources; ii) support a clear set of expectations (framed systematically) for users (end users and power users) and developers; iii) be open and extensible for recommendations applicable to existing and future system requirements; iv) support interoperability, inter-operation and cooperation; and vi) be responsive to the user and developer community. ILS-DI recommendations can be logically related with a set of twenty-five interlinked functions. Each of the twenty-five (25) functions can be grouped into one of four overall categories: 1) Data aggregation (harvesting and distributed searching); 2) Search (simple and advance search operators); 3) Patron services (general and interactive interfaces); and 4) Integrated service framework (on-the-fly integration of open contents, data sets etc.). A summary of ISL-DI recommendations is provided in

Unit 3 of this block. For DLF ILS Discovery Internet Task Group (ILS-DI) Technical Recommendations visit www.diglib.org/architectures/ilstdi/DLF_ILS_Discovery_1.0.pdf and for DLF ILS Discovery Internet Task Group (ILS-DI) Technical Recommendations see www.diglib.org/architectures/ilstdi/DLF_ILS_Discovery_1.1.pdf.

1.5.3 Request for Proposals (RFPs)

RFPs, developed by different libraries, library associations and ILS experts, are good source of information to trace the recent developments in automated library systems. Study of RFPs helps us to determine requirements, prescribing standards and demanding services from ILS vendors and developers. It acts as a guiding document for ILS developers and library automation managers. A request for proposal (RFP) is a formal request for a bid from suppliers of library systems. The RFP provides the ILS vendor with the outline, purpose, scope, description, minimum service requirements, minimum standards requirements, administration and security issues etc. for the automated library system in a comprehensive manner. The RFP process is useful in identifying the needs and priorities of the library including the future plans related with library automation. The RFP prescribes the resources that need to be acquired, the services that need to be offered, the standards that need to supported, the selection criteria for ILS, and the requirements for the software vendor. It also sets the timeframe for the project of automating a library. A RFP for library automation is a critical document in the process of implementing an ILS. L. T. David (2001) advocated consulting following online resources for developing RFP on ILS:

- Cohn, John M. and Kelsey, Ann L. Planning for automation and use of new technology in libraries. Online. URL: <http://web.simmons.edu/~chen/nit/NIT'96/96-065-Cohn.html>
- Integrated Library System Reports. Sample Request for Proposals (RFPs) and Request for Information (RFIs) for library automation projects. Online. URL: <http://www.ilsr.com/sample.htm>
- Kirby, Chris. and Wagner, Anita. The Ideal Procurement Process: The Vendor's Perspective. Online. URL: <http://www.ilsr.com/vendor.htm>
- Planning and Evaluating Library Automation Systems. Online. URL: http://dlis.dos.state.fl.us/bld/Library_Tech/Autoplan.htm
- Sample RFP. Library HQ. Online. URL: <http://www.libraryhq.com/rfp.doc>.

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

13) Discuss how ILS-DI and OLE recommendations may help in shaping futuristic ILSs.

.....

.....

.....

.....

14) What is a RFP? How RFPs may help us in library automation?

.....

.....

.....

.....

.....

1.6 AUTOMATED LIBRARY SYSTEM: DEVELOPMENT OF RFP

You already know what RFP is and how these documents may help us in planning and implementing integrated library system for developing automated library system. It's already clear to you that the first logical step in library automation is to develop RFP. The RFP acts as a base document in developing automated library system, just as a blueprint helps in developing a building. A comprehensive RFP aims to achieve two broad groups of tasks – 1) guides the library in evaluation of integrated library systems; and 2) helps the library to choose and acquire the most appropriate system. Although not all libraries in India (also in abroad) that purchase ILS prepare RFPs, the process of preparing an RFP helps the library identify its needs, priorities, options and also in setting future course-of-action for ICT-enabled library services. Moreover, it may guide a library in customising open source ILS according to goals and requirements set in RFP, if the library decided to use open source software.

Needs for developing RFP

You already know that the widespread use of Integrated Library Systems (ILS), global communications via the Internet, increasing numbers of digital library initiatives, availability of web 2.0 tools, rising of cloud computing, evolving of linked open data have made the need for compliance with standards for a library system more crucial than ever. But which standards are important when considering a library system, what services are necessary for next generation library users, what software architecture is suitable for rapidly changing computing environment, what data formats are the most comprehensive? And how can one determine if a commercially available ILS or an open source ILS really complies with global standards related to functional subsystems of a library? Here lies the importance of developing RFP for library automation. The RFP aims to answer these questions through:

- Setting criteria for evaluating RFP responses and ILS products;
- Prescribing standards compliance needs;
- Identifying the current national, regional and international standards including de facto standards;
- Conforming requirements specific to the library system;
- Assisting in effective and clear communication between library managers and ILS developers; and
- Guiding application of relevant standards for major functional areas of library automation, e.g. Bibliographic Format, Record Structure, Information Retrieval, Serials, etc.

Components of RFP

The RFP requires being a structured document. The components of a typical RFP are as follows:

- 1) Background information about the library
 - What are its mission, vision and goals?
 - What services does it offer?
 - What is the size of its collection, circulation and user community?
- 2) Detailed Statement of needs
 - What are the objectives of the library automation?
 - What are the needs for compliance with standards for a library system?
 - What are the service level requirements?
 - What are the functional requirements?
- 3) Vendor name and contact addresses and numbers
 - Who are the potential ILS vendors that may satisfy library requirements?
 - How these vendors can be contacted?
 - Who are the third-party service providers for potential open source ILSs?
- 4) Time frame
 - What are the steps/activities and when should each be finished?
 - What are the priority-level for required activities?
 - What should be the schedule for completion of tasks?
- 5) Evaluation criteria and method
 - What are the critical factors that must be present?
 - How to frame parameters for evaluating different responses against RFP?
 - What should be the method for evaluating ILS products?
- 6) Systems requirements and specifications
 - What specific features of the system must be present?
 - What are infrastructural requirements?
 - What are the software-level requirements?
- 7) Request for quotation
 - What should be the format for quotation?
 - How much will the system cost?
 - What are the conditions for on-site services and updating of software?
 - How to calculate ROI (Return on Investments)?

Steps in the development of RFP

The above-mentioned components of a typical RFP require to be developed methodically through appropriate steps. David, L. T. (2001) prescribed a set of steps for developing RFP in his guide book entitled *Introduction to integrated library systems* published by Information and Informatics Unit, UNESCO

Bangkok, Thailand. The steps are as follows:

- 1) Needs assessment
- 2) Studying available ILSs (including open source ILSs)
- 3) Listing potential vendors of the ILSs (third-party vendors for open source ILSs)
- 4) Specifying needs and standards compliance
- 5) Specifying criteria for evaluation for ILSs
- 6) Developing a time frame for task completion
- 7) Writing the RFP (with necessary components)
- 8) Submitting to legal office for comment on contract agreements
- 9) Rewriting according to the specifications of the legal office
- 10) Submitting to vendors for requesting proposals
- 11) Receiving proposals from vendors
- 12) Evaluating proposals against a set of parameters
- 13) Preparing a short list of vendors/third-party service providers
- 14) Requesting a demo of the system
- 15) Purchasing/commissioning the system
- 16) Preparing the final contract
- 17) Implementing the system
- 18) Evaluating the implemented system.

Experts recommend that the actual evaluation (both software and responses received from vendors and third-party service providers in case of open source ILS) must be done by a team, and not by an individual.

Time frame for completion of steps needs to be set and followed strictly to achieve targets. David (2001) suggested a time frame for steps to provide standard length of time needed to complete each stage of the process. Table 1.3 is an illustration of the time frame developed by Davis (2001) for the RFP and selection processes.

Table 1.3: Time frame for steps in RFP development (source: David, 2001)

Steps	Month 1	Month 2	Month 3	Month 4	Month 5+
Needs assessment	×				
Studying available ILS	×				
Listing potential vendors of the ILS	×				
Specifying needs	×				
Specifying criteria for evaluation					
Developing a timeframe	×				
Writing the RFP		×			

Submitting to legal office for comment		×			
Rewriting according to the specifications of legal office		×			
Submitting to vendors		×			
Receiving proposals from vendors			×		
Evaluating proposals				×	
Preparing a short list of vendors				×	
Requesting for a demo of the system				×	
Selecting your system				×	
Preparing the contract				×	
Implementing the system					×
Evaluating the implemented system					×

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

15) What is need of a RFP in developing automated library system? Enumerate essential components of a typical RFP.

.....

.....

.....

.....

.....

.....

.....

16) Discuss the steps for developing a RFP as suggested by L. T. Davis.

.....

.....

.....

.....

.....

.....

.....

1.7 AUTOMATED LIBRARY SYSTEM: TRENDS AND FUTURE

This Unit ends with listing a set of ongoing trends and upcoming changes in automated library system. The issues related with changes have been discussed here in full length and linked with global recommendations in Unit 3 of this Block which deals with library management software. This section attempts to introduce you with the cutting-edge technologies that are going to influence the processes, procedures, architectures and platforms for integrated library systems.

1) **Service-oriented Architecture (SoA) in ILS**

Service-Oriented Architecture (SOA) is an ICT architectural style that supports seamless flow of information, which is independent of systems, platforms, software architecture, data structures etc. In short it supports sharing of services and datasets in heterogeneous information infrastructure. The term service-orientation indicates a way of thinking in terms of services, service-based development and the outcomes/deliverables of services. SoA is now established as a mature architectural style and the ILSs have started switching to this promising architectural style to provide end users innovative library services and opportunities to other libraries to utilise resources and services (through application program interface). The SoA is an essential attribute of an ILS to support Cloud Computing. It facilitates the effective use of the Cloud.

2) **Cloud-based library automation**

Cloud computing is network based computing facilities that support on-demand use of hardware and software resources. Libraries can take advantages of cloud computing in the following ways:

- i) using ILS available in remote server through web browser without any installation;
- ii) hosting the Web-OPAC and staff interfaces in remote server without burden of local management of server and arrangement of IP address and domain name;
- iii) setting up own remote file storage and database system (with scheduled backups).

The cloud computing mainly supports three facilities. These are Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). The Cloud based library automation has following advantages:

- i) Resource pooling (cloud computing providers provides a vast network of servers and hard drives for use by client libraries);
- ii) Virtualisation (libraries do not have to care about the physical management of hardware, software, user interface, data backup and hardware compatibility);
- iii) Elasticity (addition of storage space on-demand in hard disk or increasing server bandwidth can be done easily);

- iv) Geographical scalability (cloud computing allows libraries to replicate data to several branch libraries world-wide);
- v) Automatic resource deployment (libraries only needs to choose the types and specifications of the resources required and the cloud will configure it automatically);
- vi) Metered billing (library will be charged for only what they use).

As a whole cloud-based library automation is quite useful and cost effective for small and medium sized libraries. Large-scale libraries may offer datasets on the cloud for use by small libraries (Data as a Service (DaaS)). Some of the well-known cloud-based services are listed in Table 1.4 for your ready reference.

Table 1.4: Cloud platform, systems and services

Cloud platform	Cloud systems	Cloud services
Software as a Service (SaaS)	Server Virtualisation, Open URL resolver, Application software	GoogleDoc, GoogleApps, OpenID, Adobe
Platform as a Service (PaaS)	Cloud based ILS, Inter Library Loan	LibLime, OSSLab, N-LARN project in India, Polaris, Exlibris
Infrastructure as a Service (IaaS)	Discovery services, Digital repository, Web hosting, Storage	Amason Elastic Compute Cloud (EC2), Amazon Simple Storage Solution (S3), Dropbox Cloud storage

The major cloud service providers and related services are listed in Table 1.5.

Table 1.5: Cloud providers and services

Cloud providers	Types of services
Amazon Web Services	IaaS, PaaS, SaaS
EMC	SaaS
Eucalyptus	IaaS open source Software
Google	PaaS (AppEngine), SaaS
IBM	PaaS, SaaS
Lincode	IaaS
Microsoft	PaaS (Asure), SaaS
Rackspace	IaaS, PaaS, SaaS
Salesforce.com	PaaS, SaaS
VMware vCloud	PaaS, IaaS

3) **Linked Open Data (LOD)**

Linked Open Data (LOD) refers to publishing and connecting structured data on the Web for use in public domain. The three Key technologies that support LOD are: URI (Uniform Resource Identifier, a generic means to identify entities or concepts in the web), HTTP (Hypertext Transfer Protocol, a simple yet universal mechanism for retrieving resources, or descriptions of resources over the web), and RDF (Resource Description Framework, a generic graphical data model to structure and link data that describes things in the web). Linked Open Data (LOD) has two basic purposes:

- i) publish and link structured data on the Web; and
- ii) create a single globally connected data space based on the web architecture.

Tim Berners-Lee advocated four rules for converting dataset to LOD. These are:

- 1) Use URIs as names for things;
- 2) Use HTTP URIs so that people can look up those names;
- 3) When someone looks up a URI, provide useful information, using the standards (RDF, SPARQL); and
- 4) Include links to other URIs, so that they can discover more things.

W3C established Library Linked Data Incubator Group in 2011 “to help increase global interoperability of library data on the Web, by bringing together people involved in Semantic Web activities — focusing on Linked Data — in the library community and beyond, building on existing initiatives, and identifying collaboration tracks for the future.” Libraries may utilise bibliographic data, authority data, classification schemes, vocabulary control devices etc. available as LOD for enriching existing library services and for introducing new information services. Some major examples of library LOD are – AGROVOC multilingual structured and controlled vocabulary, British National Bibliography (BNB) published as Linked Data, VIAF, LCSH, LC Name Authority File (NAF) provides authoritative data, MARC country, and language codes, Dewey.info etc. ILSs are taking advantages of integrating LOD available in library domain through appropriate APIs. For example, the cataloguing module of Koha can be linked with VIAF (Virtual Internet Authority File – a linked dataset of authority data from 21 major national libraries of the world) for getting authority data automatically to control name authority in local library catalogue.

4) **Web-scale library management**

Web-scale library management service is essentially, a cloud based solution developed by OCLC. In this service OCLC member libraries are not only getting shared computing infrastructure but also shared data from WorldCat. OCLC is successfully mixing four basic elements of cloud computing i.e. IaaS, PaaS, SaaS and DaaS (see cloud computing section above). There has been a change in trends of library automation. It is no longer about which library provides the largest collection but about which library can provide their community with the best means to access the materials they need, regardless of location (OCLC, 2011). Libraries can increase visibility at the global scale and accessibility to services at the wider scale by using the new Web-scale library management facility.



Fig 1.3: Web-scale Library System

Source: OCLC (2011), Libraries at web-scale, Dublin, p. 23

5) Web 2.0 compliant ILS

The present web (often referred as web 1.0 in blogsphere) is progressing towards a User-centred entity with the support of an advanced set of technological tools that are collaborative, interactive and dynamic in nature. Radfar (2005) identified following characteristics of web 2.0 – i) a platform enabling the utilisation of distributed services; ii) a phenomenon describing the transformation of the web from a publication medium to a platform for distributed services; and iii) a technology that leverages, contributes, or describes the transformation of the web into a platform for services. ILSs are all set to take advantages of participative architecture of the web and introducing new services like user tagging of subject descriptors, ratings of documents by users, RSS feed for search query, integration with web 2.0 services like read/write web, collaborative web, social networking tools and information mashup. This new trend ILS is also termed as ILS 2.0.

6) Information mashup

Information mashups tools allow remixing of data, technologies or services from different online sources to create new hybrid services (O'Reilly, 2005) through lightweight application programming interface (API). ILS uses information mashup in managing and integrating virtual contents distributed globally with local library resources. Information mashups are becoming popular application of Web 2.0 around the world such as KohaZon (integration of Koha OPAC with Amazon services), WikiBios (a mashup where user can create on-line biographies of each other in a Wiki setup), LibraryLookup (integration of Google maps with library directory service in UK) and many more such instances.

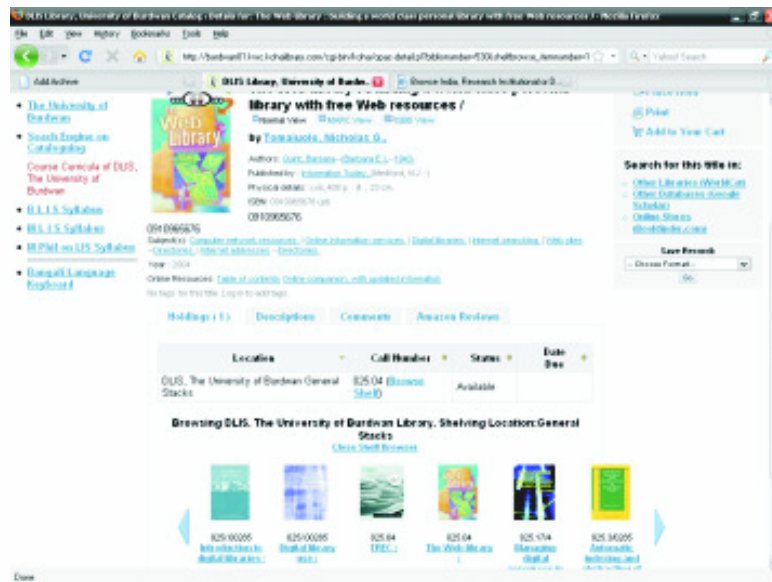


Fig. 1.4: KohaZon: Mashup of Koha with Amazon

7) Interactive user interface: OPAC 2.0

Most of the ILSs now support web-OPACs. OPAC 2.0 is the next generation web-OPAC where users can interact, collaborate and participate in library workflows such as describing resources (folksonomy), tagging subject descriptors, rating of documents, creating personalised information environment, posting on library blog, suggesting new documents, commenting on library services, publishing book reviews, posting likes on facebook for library books and many such facilities. ILSs are increasingly taking advantages of web 2.0 technologies and services to convert static OPAC into dynamic OPAC 2.0.

8) New cataloguing standards

Document description models and standards are changing rapidly. We have now E-R (entity-relationship) based bibliographic data model known as FRBR (Functional Requirements for Bibliographic Records, developed by IFLA in 1998) in place of flat data structure of ISBD. Similarly FRAD ((Functional Requirements for Authority Data, developed by IFLA in 2009), FRSAR (Functional Requirements for Subject Authority Records, developed by IFLA in 2010) are now established data models for managing name authority and subject authority respectively. These changes call upon necessary data structures in ILSs to suite FRBR, FRAD and FRSAD. Both commercial ILS group (e.g. Vitua ILS from VTLS group) and open source ILS group (e.g. Koha) are in the process of implementing the structural changes to address the improvements in cataloguing.

9) Application of discovery tools

Uses of discovery tools are increasing in libraries. Discovery tools, powered by federated search mechanisms, allow library patrons to perform concurrent searching in the library catalogue (metadata level), journal articles (full-text level), electronic theses and dissertations, consortia databases, public web, open access repositories, union catalogues etc. through a single-search interface with a set of feature-rich tools to support users. Discovery tools – i) can be integrated with existing library OPAC; ii) can import metadata into one index; iii) can apply one set of search algorithms to retrieve and rank results. As a result these tools support

rich user experiences in terms of speed, relevance, and ability to interact consistently with results. Moreover, the unified interface is a big boost for users as they no longer need to choose a specific search tool to begin their search. These tools are available commercially (e.g. EBSCO Discovery Service) and also as open source products (such as VuFind, SOPAC, Blacklight, OpenBib etc.).

10) **Digital media archiving module**

The distinction between automated library system and digital library is blurring day-by-day. This is because of the fact that most of the ILSs are integrating digital media archiving module or DMA (e.g. NewGenLib 3.0 onwards) to handle full-text discovery of documents in different formats. This trend of ILS is important in the sense that in future library can handle both automated and digital library systems through a single instance of ILS. Another advantage of DMA is the scope to integrate courseware in multimedia formats in case of academic libraries. Some ILSs are also achieving compatibility with OAI/PMH standard to support metadata harvesting in ILS (e.g. Koha version 3.10.1 onwards).

11) **Community information services as outreach process**

Community information services meant to support community members with the information originated in the community. The service includes three broad groups – survival information such as that related to health, housing, income, legal protection, economic opportunity, political rights etc.; citizen action information required for effective participation as individual or member of a group in the social, political, legal, economic process; and local information i.e. basic information concerning courses, educational facilities, government agencies, local organisations, fractional groups, health professionals etc. including a calendar of local events. ILSs now (e.g. Vitua ILS and Koha are supporting MARC 21 community information format to handle community information resources) are trying to include community information service module to extend the role of ILS to provide outreach services.

12) **Increasing use of open source software**

The domain of library and information science, right from the beginning of the open source movement, is benefitted through structured effort and software philanthropy. We have matured ILSs like Koha (comparable to any global ILS), Evergreen, Emilda, NewGenLib; comprehensive digital library software like DSpace from the MIT, US (with support from HP), Greenstone Digital Library Software (or GSDL) from University of Waikato (presently supported by UNESCO). Use of open source ILSs are increasing all over the world because of the transparent use of library standards and scope of customisation to suite the specific requirements of a library. Moreover commercial ILSs are also utilising open source components like MARCEdit & ISISMARC (MARC cataloguing tools), YAS toolkit (Z39.50 client and server), Lucene & Solr (Text retrieval engines), Unicode-compliant multilingual tools etc. The use of open source software in library automation ensures 3F – fund (as these are free of cost), freedom (as these are free to customise) and fraternity (as these are supported by international communities).

13) Emergence of open standards

Open standards are available in public domain. These are the standards that anyone can incorporate into their software, service and system. MARC record standard is possibly the most visible open standard in the domain of library services. Library systems of any type or size are required to be compatible with global standards to achieve interoperability. Here lies the importance of open standards. These are developed, approved and maintained via collaborative process to facilitate exchange of datasets. These standards are available at no cost, well-documented, transparent and free from any kind of use restriction. ILSs are increasingly depending on open standards such as MARC 21 family of standards (Five standards), OAI/PMH, CCL (Common Command Language), SING, Dublin Core metadata standard, SRU, SRW, OpenURL, MARC-XML, METS, MODS etc.

14) Interoperability capabilities

Interoperability refers to communication between systems (external interaction) or system parts (internal interaction). Libraries are now operating in distributed information environment and many library systems communicate electronically with sources of bibliographic records (publisher or cataloguing agencies), book vendors, and users. They also now interconnect themselves with networked information resources outside of the library and deliver these through library-maintained interfaces (e.g. inter library loan, distributed cataloguing, metadata harvesting etc.). ILS developers are aware of these facts and thereby supporting more and more interoperability standards in different modules of ILSs.

15) Multi-lingual records management through Unicode

Multilingual (including Indic scripts) information processing requires standard text encoding scheme (such as Unicode), which can store, process and retrieve regional language based documents. But creation of multi-script databases requires not only Unicode-compliant operating system (OS) and other application programmes such as Virtual Keyboards to enter multi-script records, Open Type Fonts (OTF) to support extended character sets and layout features, and Rendering Engines to display script specific conjuncts and ligatures properly (Mukhopadhyay, 2006). ILSs are trying to support Unicode (especially UTF-8) to store native character sets, integrated virtual keyboard and supportive text retrieval engines to ensure processing and retrieval of multilingual documents.

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

17) Write in brief the trends in the development of ILSs.

.....

.....

.....

.....

.....

18) What is cloud computing? How is it to going to help libraries?

.....

.....

.....

.....

.....

1.8 SUMMARY

Library automation is an area from where future benefits will emerge. It means that if a library is not automated it won't be in a position to take the advantages of ICT-enabled library services in future. This Unit acts as foundation and aims to introduce you with the concept of integrated library system and the advantages associated with it. It covers historical and theoretical foundations of library automation supported by a timeline of development of related technologies. In this Unit you can find guidance – 1) to identify the requirements for library automation; 2) to follow model for integrated library system; 3) to differentiate automated and digital library system; 4) to understand the typical steps for accomplishment of library automation; 5) to appreciate needs for standards in ILS and to recognise essential standards that need to be ensured; 6) to identify features of ILS in rapidly changing technological environment. This unit also provides knowledge about emerging global recommendations for developing ILS in the context of cutting edge technologies like cloud computing, linked open data and web scale library management. It also covers roles and components of RFP and steps for developing RFP for library automation, and allows you to develop skills in preparing RFP. This unit ends with a brief discussion on forthcoming features and ongoing changes in the arena of ILS against a fifteen-point checklist.

1.9 ANSWERS TO SELF CHECK EXERCISES

- 1) Library automation is the generic term that denotes applications of Information Communications Technologies (ICT) for performing manual operations in libraries of any type or size. It supports three broad groups of library activities – i) housekeeping operations; ii) information retrieval; and iii) on-the-fly integration of library materials with open datasets. Library automation requires for – 1) increased operation efficiencies; 2) betterment of library services; 3) innovative information services; 4) wider user access and 5) more productive use of library staff.

- 2) An ILS is capable of managing the operations of more than one basic library functions by sharing the files in the server to perform them. For example data from the book catalog master file and the patron master file can be retrieved and used in the circulation module to perform the circulation function of the ILS. In such systems files are interlinked so that deletion, addition and other changes in one file automatically activate changes in related files. It means integrated library management system is sharing a common database to perform all the basic functions of a library.

- 3) Library automation is a generic term that refers to the application of computers in libraries to automate operations. It can be standalone system supporting only one module like cataloguing or it can be integrated to link all modules or library subsystems through a common shared database. On the other hand, ILS is an automated library system that utilises shared data and files to provide interoperability of multiple library functions, e.g. cataloging, acquisition, circulation, serials, etc.
- 4) There are five generations of library automation categorised on the basis of technological breakthroughs. Alternatively these are also called five ages of library automation. The first age is characterised by the introduction of PCs in library automation and the second age is dominated by LAN based ILSs. The third age is marked by the Web-enabled ILS and the fourth age is featured by integration of full-text digital objects in ILS. The fifth and the present age is characterised by cutting edge technologies like cloud computing, linked open data and web 2.0 features for interactive user interfaces.
- 5) Library automation initiated in 1930s and applied in large scale in 1970s with the availability of low-cost PCs. The decade of eighties witnessed application of global standards and local area networks in library automation with the advent of campus-wide ILS and relational database architecture. The decade of nineties is dominated by the application of web technologies in library automation. Technologies like CGI architecture, Web-OPAC, digital media archiving are some of the well known features of this decade. The first decade of 21st century is the decade for application open source software, open standards and extended web technologies like web 2.0, cloud computing and linked open data in library automation.
- 6) The chronological order of the technological breakthroughs in the domain of library automation is as follows – i) Low-cost PCs used in 1970s; ii) LAN-based ILS with relational database backend, global exchange format and client-server architecture; iii) use of web technologies to provide time-space independent user services including Web-OPAC; iv) digital media archiving, interoperability standards and open source software; v) interactive user interfaces and seamless integration of linked open data.
- 7) Library automation has manifold advantages. Automation of library housekeeping operations is considered as an especially critical area from which future benefits will emerge. It means that if a library is not automated it can't take advantages contributed by ICT such as digitisation, web-enabled library system, use of linked open data, remote management of library, interactive user services etc. Library automation ensures acceptability of library to new generation users.
- 8) Library management software or ILS forms the core component of library automation. An ILS should support all the basic activities of library, seamless integration in different modules, global and national standards in the domain, suitable software architecture, interoperability standard data formats, multilingual processing and retrieval, and integration with open datasets. ILSs need to be future friendly, user friendly and open for customisation.

- 9) Procedural model of library automation is proposed by ASLIB (Association of Information Managers, UK) as a general model for automating library housekeeping operations. Presently most of the ILSs follow this model for designing different functional modules of ILSs. The model proposes that a library system has mainly two subsystems – administrative subsystem and operational subsystem (amenable for automation). The operational subsystem may be divided into four further subdivisions namely Acquisition, Processing, Use and Maintenance. Within each of these divisions there are a number of procedures (eighteen in total) and within each procedure there are one or more of six possible activities. The procedures and activities are carried out by fifteen basic tasks.
- 10) Digital libraries are managed collection of digital objects that provide full-text access to resources and differ significantly from automated library systems in terms of – 1) search features (metadata only vs. full-text and metadata); 2) document description (MARC 21 vs. Dublin Core); 3) interoperability standards (Z39.50 vs. OAI/PMH); and 4) software architecture (centralised vs. distributed).
- 11) Standard is a specification accepted by recognised authority as the most practical and appropriate current solution of a recurring problem. Establishing order to chaos and building collaboration are two most important prerequisites for effective information services. Both of these requirements depend on shared understanding i.e. on standards. Libraries all over the world are entering into the next wave of development to meet volume and variety of users' information demands. Interoperability and interactive user interface are two buss words in developing global information infrastructure. Libraries are no exceptions. Automated library systems are trying to be compatible with globally agreed upon standards related with information processing such as data formats (MARC 21 data formats, CCF, UNIMARC); interoperability standards (ISO 2709, MARC-XML, Z39.50, SRW, SRU OAI/PMH) and character encoding standards (Unicode).
- 12) ILSs support three broad groups of library activities – i) housekeeping operations; ii) information retrieval; and iii) on-the-fly integration of library materials with open datasets. A typical ILS supports selection, ordering, acquisition, processing, circulation, serials control, dissemination of information services and also extends help in library administration, planning and decision making process as a MIS tool.
- 13) Designing of future friendly ILS requires guidelines. OLE project and ILS-DI recommendations are acting as such guidelines recognised globally. The principal aim of OLE project is cost-effective integration of library management with other institutional systems on the basis of Enterprise Resource Planning (ERP) enabled Abstract Reference Model. On the other hand, ILS-DI guides developers in – 1) Data aggregation (harvesting and distributed searching); 2) Search (simple and advance search operators); 3) Patron services (general and interactive interfaces); and 4) Integrated service framework (on-the-fly integration of open contents, data sets etc.).
- 14) A request for proposal (RFP) is a formal request for a bid from suppliers of library systems or third-party software vendor in case of open source

software. RFPs are aiming to determine library requirements, prescribing standards and demanding services from ILS vendors and developers. The RFP prescribes the resources that need to be acquired, the services that need to be offered, the standards that need to be supported, the selection criteria for ILS, and the requirements for the software vendor including a time schedule for each level of activities. It guides the library in evaluation of integrated library systems and helps the library to choose and acquire the most appropriate system.

- 15) RFP is required to guide us in framing requirements, selecting ILS and implementing ILS. The components of a typical RFP includes: 1) library profile; 2) automation need profile; 3) vendor profiles; 4) time frame; 5) evaluation parameters and method; 6) system requirements and specifications; 7) format for proposal.
- 16) L. T. David in 2001 advocated a set of steps for developing RFP. The process starts with need assessments and ends with evaluation of implemented system. It includes a total of eighteen steps.
- 17) The rapidly changing technological environment leads to corresponding changes in the development of ILS. The influence of technologies leads to the development of ILSs from stand-alone system to web-enabled systems in five decades. The major trends that are influencing ILSs presently are web architecture, Unicode-compliant processing and retrieving environment, supports for interoperability standards, open source movement and cutting edge technologies like cloud computing, web scale platform, web 2.0 and linked open data.
- 18) Cloud-based library automation is quite useful and cost effective for small and medium sized libraries. Cloud computing is network based computing facilities that support on-demand use of hardware and software resources. Libraries can take advantages of cloud computing in the following ways – i) by using ILS available in remote server through web browser; ii) by hosting the Web-OPAC in remote server; iii) by setting up own remote file storage and database system (with scheduled backups).

1.10 KEYWORDS

- Acquisition** : The process of obtaining resources for the library's collection, typically including ordering, receiving and payment.
- API** : Application Programming Interface. A language and message format used by an application program to communicate with the operating system or some other control program such as a database management system (DBMS).
- Authority record** : A record that shows the preferred form of a personal or corporate name, geographic region or subject. It also includes variant forms of the preferred form as cross references.

- Barcode** : A printed code, consisting of lines and spaces that can be read by a bar code scanner (reader), affixed to physical materials in a library collection to identify particular items for tracking and circulation.
- Bibliographic identifier:** A unique identifier which unambiguously identifies a bibliographic record within an ILS catalog and is assumed to be persistent, at least as long as the records are managed within the ILS.
- Bibliographic metadata:** Information about a resource that serves the purpose of discovery, identification and selection of the resource. Includes elements such as title, author, subjects, etc.
- Discovery application:** A computer application designed to simplify, assist and expedite the process of finding information resources.
- Dublin Core** : A fifteen element metadata set for use in resource description intended to facilitate discovery of electronic resources.
- EDI** : Electronic Data Interchange (EDI) is a standard method for exchanging structured data, such as purchase orders and invoices, between computers to enable automated transactions.
- EDIFACT** : EDI For Administrations, Commerce and Transport The concept of utilising a single set of specifications for bibliographic records regardless of the type of material they represent.
- ERMS** : Electronic Resources Management System is used to manage a library's electronic resources, primarily e-journals and databases. Systems can include features to track trials, license terms and conditions, usage, cost, and access.
- FRBR** : Functional Requirement for Bibliographic Records is a conceptual model for the aggregation and display of bibliographic records. FRBR is an entity-relationship model, with four primary entities - work, expression, manifestation, and item - which represent the products of intellectual or artistic endeavor.
- ILL** : Inter Library Loan (ILL) is the process between two libraries of borrowing and lending a physical bibliographic item, or obtaining a copy of it.
- ILS** : An automated library system that utilises shared data and files to provide interoperability of multiple library functions, e.g. cataloging, acquisition, circulation, serials, etc.

- Interoperability** : The ability for two different computer systems to communicate and exchange information in a useful and meaningful manner.
- LAN** : A digital communication system capable of interconnecting a large number of computers, terminals and other peripheral devices within a limited geographical area.
- Library Automation:** Library automation is the mechanisation of housekeeping operations and information handling mainly by using computer and communication technologies.
- MARC 21** : A harmonised MARC format developed by LoC in 1999 for encoding standards related to bibliographic data, authority data, holdings data, classification data and community information. It is used for the communication and exchange of bibliographic information (mentioned earlier) between computer systems.
- MARCXML** : A metadata scheme for working with MARC data in a XML environment.
- Metadata** : Structured information that describes an information resource. “Data about data” for an information bearing object for purposes of description, administration, legal requirements, technical functionality, use and usage, and preservation.
- Metadata harvesting:** A technique for extraction of metadata from individual repositories for collection into a central catalog.
- Module of ILS** : Functions specific to a particular system capability such as the online public access catalog, cataloging, acquisitions, serials, circulation, etc.
- NCIP** : NISO Circulation Interchange Protocol (NCIP) is a standard which defines a protocol for the exchange of messages between and among computer-based application to enable them to perform functions necessary to lend and borrow items, to provide controlled access to electronic resources, and to facilitate co-operative management of these functions.
- Network** : A group of computers and other devices connected together so that they can communicate with each other, share data and resources such as printers, and perhaps share the workload of running complex programs. They may have one or more central servers to coordinate and run things, or all devices may be of equal standing (called “peer-to-peer”). The connections between them may be physical wires and cables, or wireless using infrared or radio frequency.

- OAI-PMH** : OAI - Protocol for Metadata Harvesting. Protocol for application-independent interoperability framework based on metadata harvesting, open standards HTTP (Hypertext Transport Protocol) and XML (Extensible Markup Language).
- OPAC** : On-line Public Access Catalog is a library catalog which can be searched on-line and is a module of the ILS. It is the interface between library resources and users and is designed to be “user friendly.”
- Open Source** : A concept through which programming code is made available through a license that supports the users freely copying the code, making changes it, and sharing the results. Changes are typically submitted to a group managing the open source product for possible incorporation into the official version. Development and support is handled cooperatively by a group of distributed programmers, usually on a volunteer basis.
- Open Search** : A collection of technologies developed by Amason that allow publishing of search results in a format suitable for syndication and aggregation.
- Open URL** : A URL with stored metadata that is user context sensitive in what information or hypertext link is delivered.
- Protocol** : A standard procedure for the message formats and rules that two computer systems must follow to communicate with each other.
- RSS** : Really Simple Syndication is an XML format used for distribution or syndication of frequently updated Web contents.
- SIP2** : Standard Interface Protocol Version 2 is a standard for the exchange of circulation data and transactions between different systems.
- SRU** : Search/Retrieve via URL is a standard search protocol for Internet search queries, utilising CQL (Common Query Language), standard query syntax for representing queries.
- SRW** : Search/Retrieve Web service is web services implementation of the Z39.50 protocol that specifies a client/server-based protocol for searching and retrieving information from remote databases.
- System Analysis** : A powerful technique for the analysis of an organisation and its work.
- Unicode** : A universal character-encoding standard used for representation of text for computer processing. Unicode provides a unique numeric code (a code point)

for every character, no matter what the platform, no matter what the program, no matter what the language. The standard was developed by the Unicode Consortium in 1999.

- WAN** : A computer networking system that operates nationwide or worldwide by utilising telephone line, microwave and satellite links. It is also used to interconnect LANs.
- Web Service** : Software system designed to support interoperable machine to machine exchange of data/information, typically using the XML, SOAP, WSDL and UDDI open standards.
- XML** : eXtensible Markup Language is an open standard for describing data from the World Wide Web Consortium. It is used for defining data elements on a Web page, business-to business documents, and other hierarchically structured text and data.
- Z39.50** : A NISO and ISO standard protocol that specifies a client/server-based protocol for cross-system searching and retrieving information from remote databases. It specifies procedures and structures for a client system to search a database provided by a server.

1.11 REFERENCES AND FURTHER READING

Breeding, M. *Library technology guides: key resources in the field of library automation*. <<http://www.librarytechnology.org>>

Breeding, Marshall. Perceptions 2007: an international survey of library automation. In *Library Technology Guides*, January 9, 2008. <<http://www.librarytechnology.org/perceptions2007.pl>>

Cohn, John M. & Kelsey, Ann L and Fiels, Keith Michael. *Planning for automation: a how-to-do-it manual for librarians*. New York: Neal-Schuman, 1992. Print

David, L. T. *Introduction to integrated library systems*. Bangkok: Information and Informatics Unit, UNESCO Bangkok, Thailand, 2001. Print

Dula, M., Jacobsen, L., Ferguson, T. and Ross, R. Implementing a new cloud computing library management service. In *Computers in Libraries*, 32.1(2012), pp. 6-40.

Duval, B.K. and Main, L. *Automated library systems: a librarian's guide and teaching manual*. Westport, USA: Meckler, 1992. Print

Goldner Matt. Winds of Change: Libraries and Cloud Computing. In *OCLC Online Computer Library Center*, 14.7(2010) <<http://www.oclc.org/multimedia/2011/files/IFLA-winds-of-change-paper.pdf>>

Haravu, L. J. *Library automation: design, principles and practices*. New Delhi: Allied Publishers Private Limited, 2004. Print

Hodgson, Cynthia. *The RFP writer's guide to standards for library systems*. National Information Standards Organisation: Bethesda, Maryland, 2002. <<http://www.niso.org>>

Hopkinson, A. Introduction to library standards and the players in the field. *Digitalia*, (2006). <http://digitalia.sbn.it/upload/documenti/digitalia20062_HOPKINSON.pdf>

Kuali Foundation. *Kuali Open Library Environment: test drive OLE version 0.6*. (2012). <<http://demo.ole.kuali.org/ole-demo/portal.jsp>>

Mukhopadhyay, P. *Library automation packages - introduction* – BLII 003, Block 1, Unit 1 of CICTAL course, IGNOU, 2005.

Mukhopadhyay, P. Library housekeeping operations – BLII- 001, Block 1, Unit 11 of CICTAL course, IGNOU, 2005.

Mukhopadhyay, P. and Asim, A. Multiscript information retrieval system: A FLOSS based prototype for Indic scripts with special reference to Bengali script. *Information management in digital libraries: Proceedings of the National Conference of Indian Institute of Technology, Kharagpur (August 2-4, 2006, Kharagpur.)* (2006), pp. 305-316.

Müller, T. How to choose a free and open source integrated library system. *OCLC Systems & Services*, 27.1(2011), pp 57-78. <doi:10.1108/10650751111106573>

O'Reilly, T. *What is Web 2.0?* (2005). <<http://www.oreilly.com/go/web2>>

OCLC. *Libraries at web-scale*. OCLC, Dublin, 2011. Print

Radfar, H. *On library 2.0*, (2005) <<http://hoo-ville.blogspot.com/>>

Rayward, W.B. A History of Computer Applications in Libraries: Prolegomena. *IEEE Annals of the History of Computing*, April-June, 2002, pp. 4-15.

Swan, James. *Automating Small Libraries*. Ft. Atkinson, Wis.: Highsmith Press, 1996. Print

Wilson, K.. Introducing the next generation of library management systems. *Serials Review*. 38.2 (2012), pp. 110-123.

Withers, F. *Standards for library services*. Paris: UNESCO, 1970. Print

UNIT 2 LIBRARY AUTOMATION PROCESSES

Structure

- 2.0 Objectives
- 2.1 Introduction
- 2.2 Library Workflow: System Approach
 - 2.2.1 Subsystems and Workflows
 - 2.2.2 Analysis of Tasks
 - 2.2.3 Automation of Workflow
- 2.3 Acquisition Subsystem in ILS
 - 2.3.1 Functional Requirements for Acquisition in ILS
 - 2.3.2 Workflow of Automated Acquisition
 - 2.3.3 Products and Advantages
- 2.4 Document Processing Subsystem in ILS
 - 2.4.1 Functional Requirements for Document Processing in ILS
 - 2.4.2 Workflow of Automated Document Processing
 - 2.4.3 Products and Advantages
- 2.5 Serials Control Subsystem in ILS
 - 2.5.1 Functional Requirements for Serials Control in ILS
 - 2.5.2 Workflow of Automated Serials Control
 - 2.5.3 Products and Advantages
- 2.6 Circulation Subsystem in ILS
 - 2.6.1 Functional Requirements for Circulation in ILS
 - 2.6.2 Workflow of Automated Circulation
 - 2.6.3 Products and Advantages
- 2.7 System Administration
- 2.8 Summary
- 2.9 Answers to Self Check Exercises
- 2.10 Keywords
- 2.11 References and Further Reading

2.0 OBJECTIVES

After going through this Unit, you will be able to:

- understand typical workflows of library subsystems amenable for automation;
- know how to analyse housekeeping operations systematically;
- identify the requirements, processes and advantages of automating library workflow; and
- realise issues related to administration of library automation processes.

2.1 INTRODUCTION

You already know what and why of library automation from Unit 1. This Unit aims to introduce you with the processes related to library automation in an integrated environment. You can also see here the application of procedural model of library automation in analysing tasks related to different subsystems of a library. One of the major objectives of library automation is to automate the regular workflow of library system i.e. library housekeeping operations. An ILS performs library housekeeping operation through software modules integrated seamlessly. These modules are also called subsystems under ILS. A typical ILS includes acquisition subsystem, document processing subsystem, serials control subsystems and circulation subsystem as core modules. The other managerial activities like export/import, backup/restoration, parameters setting, configuration settings etc. are performed through administrative module.

2.2 LIBRARY WORKFLOW: SYSTEM APPROACH

Automation of library housekeeping system requires the analysis of workflow and activities into their atomic structure. This process is called system analysis. You already know about Procedural Model of library automation proposed by ASLIB (now Association of Information Managers, UK). The sub-section 1.3.3 of Unit 1 covers procedural model of library automation at length. This model acts as a base for system analysis of library housekeeping operations. The procedural model proposes two basic subsystems, four operational subsystems, three levels, eighteen procedures, six activities and fifteen basic tasks as library workflow irrespective of the type and size of libraries (see Text box 1 and Table 1 in sub-section 1.3.3 of Unit 1). The summary table is given below.

Table 2.1: Library workflow

<p><u>Library System</u></p> <p><u>Four Operational Subsystems</u> (Acquisition, Processing, Use, Maintenance)</p> <p><u>Eighteen procedures</u> (Acquisition: Select, Order, Receive, Accession; <u>Processing</u>: Classify, Catalogue, Label, Shelve; <u>Use</u>: Locate, List, Issue, Reserve, Return, ILL, Photocopy; <u>Maintenance</u>: Bind, Replace, Discard)</p> <p><u>Six activities</u> (Initiate, Authorise, Activate, Record, Report, Cancel)</p> <p>Fifteen tasks (pass, receive, discard, place, remove, search, duplicate, attach, separate, move, sort, read, verify, enter and decide)</p>

2.2.1 Subsystems and Workflows

This section covers the workflow of the subsystems of integrated library system.

A) Acquisition Subsystem

The acquisition of documents is a prerequisite for libraries. A library should acquire and provide all the relevant documents to its users so that the basic

functions of the library are fulfilled. An acquisition subsystem shall perform four basic procedures – Select, Order, Receive and Accession. The scopes of these procedures are as follows:

Procedures in Acquisition Subsystem

Select

Selection of documents for library users is a very responsible job and should be based on definite principles. It is done with the help of selection tools (such as bibliographies, publishers' catalogues, trade catalogues etc.) and requests/suggestions from library users/authority. Selection of documents to be procured in the library is followed by the formal sanction of the competent authority/library committee.

Order

This procedure starts with pre-order searching, especially to avoid duplicate orders. In the next stage purchase orders are generated and placed either directly to the respective publishers or to the listed vendors/book sellers. Additionally, generation of reminders for overdue items and cancellation of orders also comes under the purview of ordering procedure.

Receive

Documents and invoices or bills usually arrive together. Bills are checked with the order list before processing for payment. Newly arrived books are tallied with the bills and the order list to check the author, title, edition, imprints and price before accessioning.

Accession

A stock register is maintained by libraries in which all the documents purchased or received in exchange or as gift are entered. Each document is provided with a consecutive serial number. The register is called Accession register and the serial number of the document is referred as Accession Number.

All the above-mentioned procedures and related activities of the acquisition subsystem can be mechanised through library management software. In such a system these basic activities are linked with the files of publishers, suppliers, budget & fund accounting, currency etc. to achieve the benefit of integrated library system.

B) Processing Subsystem

The processing procedure is the pivot round which all the housekeeping operations revolve in a library. It helps in the transformation of a library collection into serviceable resources. The procedures under this subdivision are classification, cataloguing, labeling and shelving.

Procedures in Processing Subsystem

Classify

The followings are the major classification schemes, which are used in various libraries of the world: Dewey Decimal Classification Scheme (DDC), Universal Decimal Classification Scheme (UDC), Library of Congress

Classification (LC), Colon Classification (CC), and Subject Classification (SC) etc. Classification is a mental process and demands intellectual exercises from classifier. As a result, automatic synthesis of class numbers requires the application of Artificial Intelligence (AI) techniques in development of software. The present edition of DDC is also available in CDROM and known as WebDewey.

Catalogue

Cataloguing is the prime method of providing access to the collection. Cataloguing procedure starts with technical reading of the document to be catalogued by studying title, sub-title, alternate title, author, editor, edition, reprint, imprint, dedication, preface, table of contents, collation, series, bibliographies etc. In case of manual cataloguing, the cataloguer makes separate cards for author, title, subject, cross-references and analytical entries by following any standard catalogue code (such as AACR II, CCC etc.) and file them as per the rules laid down by the library. Computerised cataloguing begins with entering bibliographical data in a pre-designed worksheet. The worksheet or data sheet is very similar to data entry form and is based on any standard content designators scheme (such as MARC 21 Bibliographic Format, CCF/B, UNIMARC etc.). Finally bibliographical data recorded in the worksheets are entered into the computer to produce machine-readable catalogue file and OPAC. Computer-based cataloguing supports importing of bibliographical datasets for the library resources either from centralised cataloguing services or from other libraries and exporting of bibliographical data of its own collection to other library systems. This facility reduces unit cost of cataloguing and ensures standardisation in cataloguing. The recent trend of cataloguing is to utilise Z39.50 protocol to download bibliographical data from other libraries and to provide global access to its own collection through Web-OPAC.

Label

It is the work of pasting various labels on different parts of a document. The following labels are generally pasted in books:

Spine label: This is done to make call number (a combination of class number and book number) properly visible to the users when the book is shelved. The size of the label is in the range of 1.25" × 1.25".

Ownership slip/mark: These are generally pasted on the inner side of the front cover at left hand top most corner. Ownership marks are put at various parts of a document by rubber stamps. The size of slip is 3" × 2.5".

Date slip: It is pasted on the top most portion of the front or back flyleaf of each book. The size of date slip is 5" × 3".

Book pocket: On the bottom of the inner right side of the front or back cardboard cover a book pocket is pasted.

Book card: One printed/hand-written book card of size 5" × 3" is put in the book pocket of each book.

In a computerised environment, various labels are printed by using library management software. In case of barcode based computerised circulation,

accession numbers of documents are converted into barcodes and printouts of barcodes are pasted on the inner back cover of documents.

Shelve

Shelving is the arrangement of documents on the shelves to fulfill the fourth law of library science – Save time of the reader. Generally books are arranged on the shelves in a classified manner as per the call number. Bound periodicals are generally shelved alphabetically by title and then by volume numbers. Although shelving works are generally manual in nature, RFID-enabled ILS helps in identifying misplaced documents in shelves and thereby supports stock rectification.

C) Circulation Subsystem

Circulation service is quite common to libraries of different types. Most libraries lend books and other library materials to be read elsewhere by users. This is convenient for the users, increases the use made of libraries' collection and reduces demand for reading space within library building. This function requires some sort of record keeping arrangement of what has been lent and to whom. There are two good reasons for keeping loan records: i) to reduce the loss of library materials; and ii) to help library staff to answer users' queries about the location of items not on the shelves.

Procedures in Circulation Subsystem

A rich variety of systems of record keeping of loans have arisen out of such needs and these are known as circulation systems. These include some common jobs for successful operations such as enrollment of members, issue and return of library documents, reservation of documents, renewal of documents, maintenance of documents and records, maintenance of statistics, interlibrary loan, issuing of gate pass, calculation and collection of fines for overdue documents etc. In a computer based circulation system, the machine-readable file consists of records for all items on loan from the library is updated periodically with new records. This file is called "transaction file" and it takes required data from other two files – "document file" and "borrower file". Modern library management software support barcode based circulation system. In such a system a barcode reader scans barcoded accession number of a document and the barcode in turn acts as a pointer to the document file. It helps to minimise labour and error in data entry operation. The concept of RFID (Radio Frequency IDentification) based circulation system is emerging rapidly in developed countries. It comprises three components: a tag, a reader and an antenna. The tag contains important bibliographical data. The reader decoded the information stored on the chip after receiving it through the antenna and sent data to the central server to communicate library automation system. RFID technology supports patron self-checkout machines and has the ability to conduct inventory counts without removing a single book from the shelf. As a whole, RFID improves library workflow, staff productivity and customer service with these attributes.

D) Serials Control Subsystem

Serials in general and periodicals in particular are essential for research and development (R & D) activities. These are the primary means of

communication for the exchange of scientific information. The periodicals or journals subscribed by libraries can be grouped into these categories: i) Indexing/Abstracting periodicals; ii) Periodicals containing news items; and iii) Periodicals containing full-text research articles and technical papers. Acquisition of serials/periodicals in a library is different from book ordering system. In contrast to books, the libraries regularly subscribe periodicals against advance payment. The modes of subscription of periodicals in a library are as follows – Through local vendors/subscription agents, Through foreign vendors/subscription agents, Direct from the publishers, As gift or Complementary, Through membership and In exchange.

Procedures in Serials Control Subsystem

The workflow of any serials control system, manual or mechanised, can be listed as below:

- Selection of serials
- Selection of subscription mode
- Formulation of terms of procurement
- Selection of vendors
- Order
- Advance payment
- Receiving and registration of serials issues in kardex
- Sending reminders in case of non-receipted issues
- Adjustment of advance payment for missing issues
- Preparation of list of subscribed journals, new arrivals and serials holdings for consultation by users
- Binding and accessioning of back volumes of serials
- Article indexing (optional).

In an automated system all these tasks are performed by library management software efficiently. It reduces workload of library staff. Automated serials control systems may be predictive or non-predictive. Predictive systems predict the arrival of individual journal issues and can generate reminders in case of non-receipted issues. Prediction means the ability to inform that a named issue of a named journal will arrive in the library within a stated time interval. Modern library management software supports predictive mode of serials control with the facilities of on-line acquisition and access to journals through publishers' portals or library consortia (like UGC Infonet in university libraries in India, N-LIST in colleges under UGC, India and INDEST for IITs, NITs and IIMs). In case of consortia-based access to journals, a library does not perform activities like acquisition, processing and shelving rather optimise user access to the on-line journals. The access interface may be a simple list (by publisher or by journal title) or may be a complex portal with facility for federated searching.

E) Maintenance Subsystem

If we don't take proper care to organise and administer the library documents regularly, these documents would become unserviceable resources immediately. The workflow of the maintenance division/section includes four major jobs.

Procedures in Maintenance Subsystem

- Shelf Rectification : It is to shelve misplaced documents in proper locations.
- Bind : It is to preserve library resources for posterior and present use.
- Replace : It is to replace a lost document by the library.
- Discard/Withdrawn : It is to weed out out-dated and torn & soiled documents from the library for making enough space for usable stock.

The integrated library automation environment requires information on lost, damaged, missing and withdrawn documents as well as documents sent for binding. These datasets are to be entered to generate and display appropriate messages for the library users and staff against specific tasks in different modules. This is also required to generate reports on lost books, missing books, books sent for binding etc. for the library administration.

2.2.2 Analysis of Tasks

The subsystems and the procedures for their managing subsystems require a set of tasks to be performed. In an automated library system a task is the collective functions of the elements for the accomplishment of the module at the next higher level. Tasks within each activity, just as the activities themselves, may not all be necessary to each procedure.

Table 2.2: Task analysis in workflow

<i>SYSTEM SUBSYSTEM PROCEDURE ACTIVITIES</i>	LIBRARY SYSTEM				
	ACQUISITION SUBSYSTEM				
	ORDER				
	INITIATE	AUTHORISE	ACTIVATE	RECORD	CANCEL
What information?	Author, Title, Sub-title, Edition, Place, Publishers, Date, ISBN etc.	Signature of Approval	Library/Branch Library, Date of Order, Order number, Name of Vendor and Bibliographical details etc.	Administrative data, Bibliographic data	Order Number, and Date Vendor, Book details
Where from?	Bibliographies, Index, Requisition, Suggestions	Competent Authority	Book Selection Tools, MIS	Order form/ Order letter	Order File/ Computer Database
When?	After Select Procedure	Before Activation	After Authorisation	After Activation	After Activation
Who?	Library Asst./ Technical Asst.	Librarian/ Section-In-Charge	Library Asst./ Technical Asst	Library Asst./ Library clerk	Library Asst.
How?	Receiving copy of Bibliographies, Suggestion slip	Enter Signature	Enter data/ information on Order form/ Computer Database and Generate Order	Filing the Copy of Order form/ Saving in Computer	Deletion from Database/ Removal from File

The analysis of tasks to perform activities within procedures may be done through a set of five primary questions: What information is needed for the activity? Where is the information obtained? When is it required? Who requires it? How is it used? These five questions should be asked to carry out possible activities under each procedure (see Table 2.2). It provides depth to the framework provided by the procedural model. An example of this approach may be shown (in Table 2.2) in the context of five possible activities of book order procedure in acquisition subsystem.

2.2.3 Automation of Workflow

The subsystems and workflows as discussed in previous two sections are completely amenable to computerisation. An Integrated Library System (ILS) manages all the subsystems of a library such as acquisitions, cataloguing, circulation, serials control and administration. These jobs are done by library professionals through librarian/administrator interface of ILS with proper authentication (login and password). The Fig. 2.1 shows modules in Koha (an open source ILS) for managing acquisition, cataloguing (bibliographic data and authority data), circulation (including member/patron management), serials control, system administration (including report generation, export/import, backup/restoration etc.).

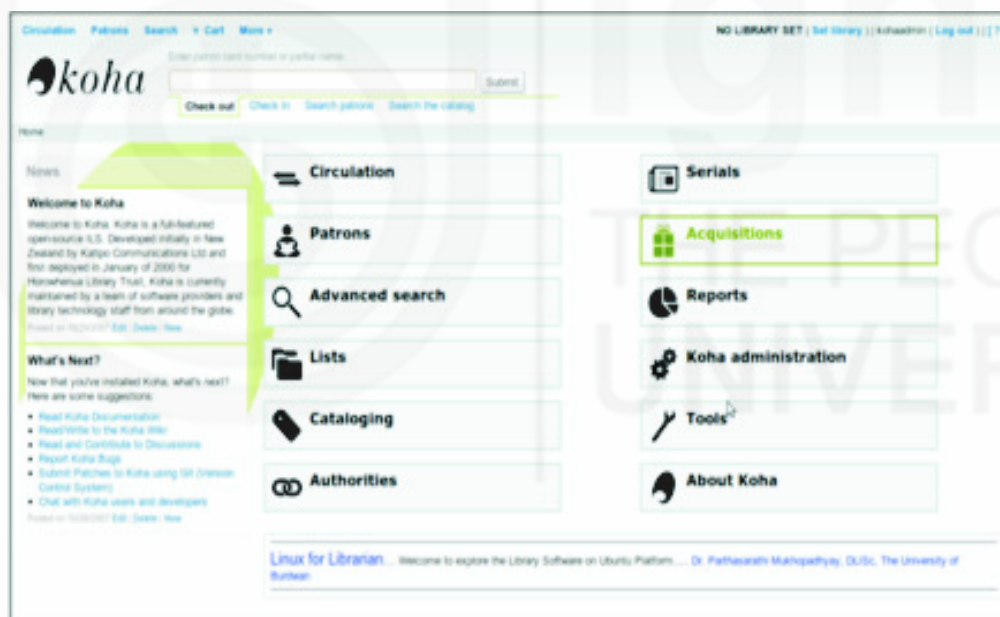


Fig. 2.1: Modules for managing subsystems and workflow in Koha

The ILS also provides a discovery interface (commonly known as the Online Public Access Catalog or “OPAC”) that enables patrons to search for resources. OPAC includes simple and advanced search interfaces with supports for member login (to check reading history, borrowed books, fines, suggestions etc.). Most of the ILSs now provide Web-OPAC (accessible through web browser) and these are now compatible with social networking tools (such as facebook, twitter etc.) and information mashup to integrate external datasets (like book cover image, book reviews etc.) with local library materials. (see Fig. 2.2).

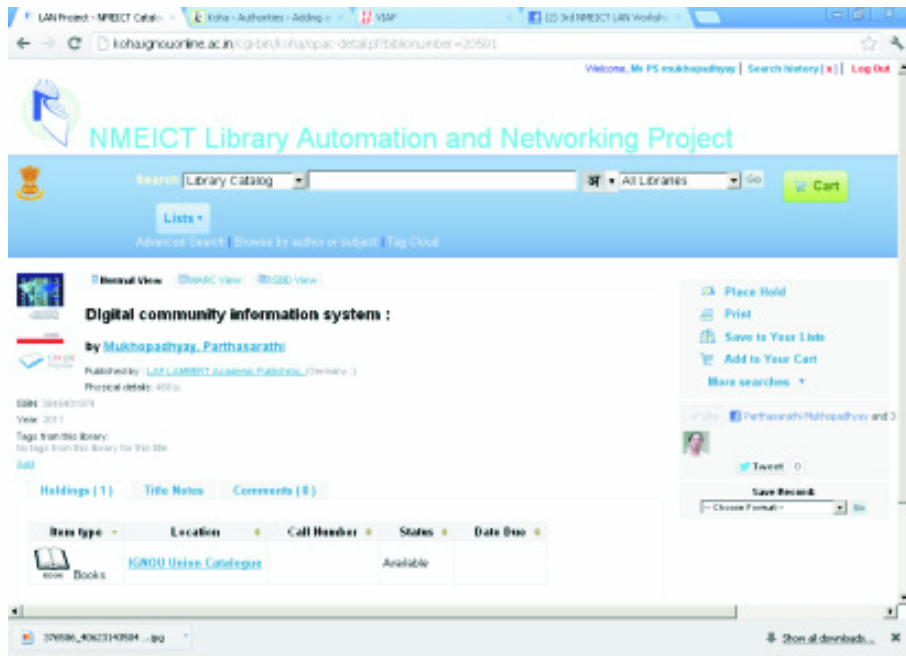


Fig. 2.2: End user interface in Koha with social networking tools

In ILS, system administrator can define privileges (known as privilege control) for each staff of the library. Privilege control ensures responsibility for each staff and also secures integrity of ILS.



Fig. 2.3: Privilege control in Koha

For example only designated circulation staff of the library (with authentication can enter into circulation module for issue, return and collecting overdue charges; similarly one staff (with login and password known only to him/her) can perform acquisition activities. Moreover (see privilege control granularity in Koha in Fig. 2.3) super-user of the ILS can control/enter in every modules. Only chief librarian should know the login/password of super-librarian. The integrated functions of ILS ensure streamlining of library operations, and the data ILS manages gives rich information through information Mashup (the concept discussed in unit 1 of this block).

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

1) Give an overview of library workflow.

.....
.....
.....
.....
.....
.....

2) What is serials control? Enumerate activities in serials control.

.....
.....
.....
.....
.....
.....

3) What is system analysis? Discuss its role in library automation.

.....
.....
.....
.....
.....
.....

2.3 ACQUISITION SUBSYSTEM IN ILS

Acquisition module of an ILS handle administrative, financial and bibliographical data related to the documents to be procured in libraries. An integrated library management system will transfer necessary bibliographical data (such as author, title, ISBN, edition) of newly procured documents to the cataloguing module of the package as and when those are marked received in the acquisition module. Integrated library system thereby avoids unnecessary duplication of data or data redundancy and achieves economy in terms of time, manpower and money. This

section discusses acquisition procedures under three heads – functional requirements, acquisition workflow and advantages of automated acquisition subsystem.

2.3.1 Functional Requirements for Acquisition in ILS

You already know that the ordering and acquisition process involves some basic routine clerical operations (as discussed in Unit 1 of this block), which are applicable to all categories of library. As a result, the procedures related to acquisition subsystem have benefited from computerisation. Generally, acquisition subsystem concentrates on monographs and other documents (available in many formats) excluding periodical publications. The basic activities of automated acquisition subsystem are: 1) To receive records of items to be acquired; 2) To check whether items requested are already in the library or on order; 3) To print orders or dispatch order electronically to supplier/publishers; 4) To check when orders are overdue; 5) To follow up overdue order; 6) To maintain a file of records of items on order; 7) To note the arrival of ordered items; 8) To process for payment; 9) To maintain book fund statistics and accounts; 10) To generate printed and electronic listing of various reports; 11) To control currency conversions; and 12) To maintain vendor performance reports and statistics. Apart from these basic activities, acquisition module of ILS should also provide support to – 1) Accommodate a variety of materials, including but not limited to – monographs, monograph in series, annual and cumulative indexes, loose leaf materials, supplements, reports, musical scores; 2) Accommodate and identify items in a variety of formats, including but not limited to – print, microform, film, videotape, audio cassette, CDROM, magnetic tape etc.; 3) Record, store and display bibliographic information, acquisition type (order, gift, approval etc.), status (reported, received etc.), library/branch/copy/fund information, invoice information, vendor information, accounting information, requester information etc.; 4) Extend facilities for unlimited number of funds/budget head, vendors, orders, claims and transactions; 5) Accommodate different types of order – regular order, membership, approval, blanket order, deposit account etc.; 6) Global standards related to document acquisition such as EDIFACT; and 7) Generate reports and statistics related to acquisition activities.

The next sections discuss three groups of activities related to acquisition. These are – pre-acquisition work, acquisition work and generation of outputs.

2.3.2 Workflow of Automated Acquisition

The acquisition workflow may be studied under two heads – pre-acquisition work and acquisition activities. Acquisition module of an ILS requires some essential works that need to be done before proceeding with actual acquisition work. These are termed as pre-acquisition work and may be identified as:

- **Pre-acquisition Works**

The general activities of this group are:

A.1) **Creation of master file for supplier**

The acquisition module must incorporate a vendor/supplier file supporting an unlimited number of vendor records including at least the following information — vendor name, address, code, phone, fax, e-mail ID, contact person, vendor discount etc.

A.2) Currency conversion

This facility is required to assist in procuring foreign documents priced in various currencies of the world (e.g. US Dollar, Euro, UK Pound etc.). The conversion of foreign currencies into Indian rupees is necessary for fund accounting and payment on the basis of the current exchange rates.

A.3) Budget process control

One of the major functions of library ordering and acquisitions subsystem is to record and to control expenditure from the library's accounts. Funds are committed for spending when orders are placed and are actually spent when the items are received in the library. Fund accounting helps to keep track of library's annual book budget and its allocation. The fund accounting aspect of a typical acquisition module in a library automation package includes four basic steps:

- **Creation of budget heads**

In this step various budget heads are created as per the prevailing practice in the library (e.g. book procurement fund, serial subscription fund, electronic resource procurement fund etc.). Each budget head is described in details and accessed through a code for easy recall as and when required.

- **Main budget allocation**

This is related to allocate the amount to the main budget along with other necessary information such as financial period, budget head, opening balance and total amount allocated or sanctioned amount. This minimum dataset is to be entered before activation of the budget process in the acquisition module.

- **Budget allocation in different heads**

This step is for receiving the amount in different budget heads.

- **Budget division**

Sometimes it is necessary to divide a budget head into several sub-heads (e.g. a book procurement head may further be subdivided into reference books and text books). This step allows a user to divide the budget into sub-heads or even divide the budget sub-heads further.

A.4) Creation of letter formats

An automated acquisition sub system should generate and print various letter formats such as approval letter, purchase order, cancellation of order, reminder letter, intimation letter, payment letter etc. In this step templates of respective letters are created and maintained by the user.

A.5) Creation of member database

This step is to create and maintain a member system. It is required to link and integrate suggestions given by the users (for procuring various materials) with the member database. Creation of member database is based on some master entries. These are – Category and associated privileges, Name of the affiliated institute, Departments/Branches/Divisions/Sections under the institute, Name of member, Member code

etc. New members can be added after these steps. Member codes are either generated automatically or may be entered manually as per the practice of the library.

- **Acquisition Works**

Actual acquisition work starts after completion of pre-acquisition works. The flow of acquisition works for document procurement in computerised libraries irrespective of type or size may be divided into four logically related groups – 1) Document related work; 2) Order processing; 3) Accessioning; and 4) Payment.

Group I tasks

Acquisition work starts with collection of information related to documents to be procured. Library staff initiates acquisition with entering bibliographical information and information about requesters from the suggestion slips and books submitted by the suppliers on approval. Bibliographical data given by the requesters in suggestion slips require to be verified by consulting book selection tools. The online databases of virtual bookstores (like Amazon or BookFinder) may also be utilised for checking bibliographical information of recently published documents. Bibliographical details of documents received by libraries in ex-gratis are also entered into the database. A library normally receives a large number of suggestions and documents for ordering. Library staff shortlist these requests depending on need, availability of fund etc. by clicking the appropriate option(s) available in the package. Finally a report is generated for all the short-listed suggestions and documents indicating number of copies required, budget code, budget head and unit price of the items requested. The library committee approves the list officially and on the basis of the final approval list library staff either select or reject the short listed titles. Books on direct approval and gratis items do not have to go through approval process from library committee or any such authoritative body.

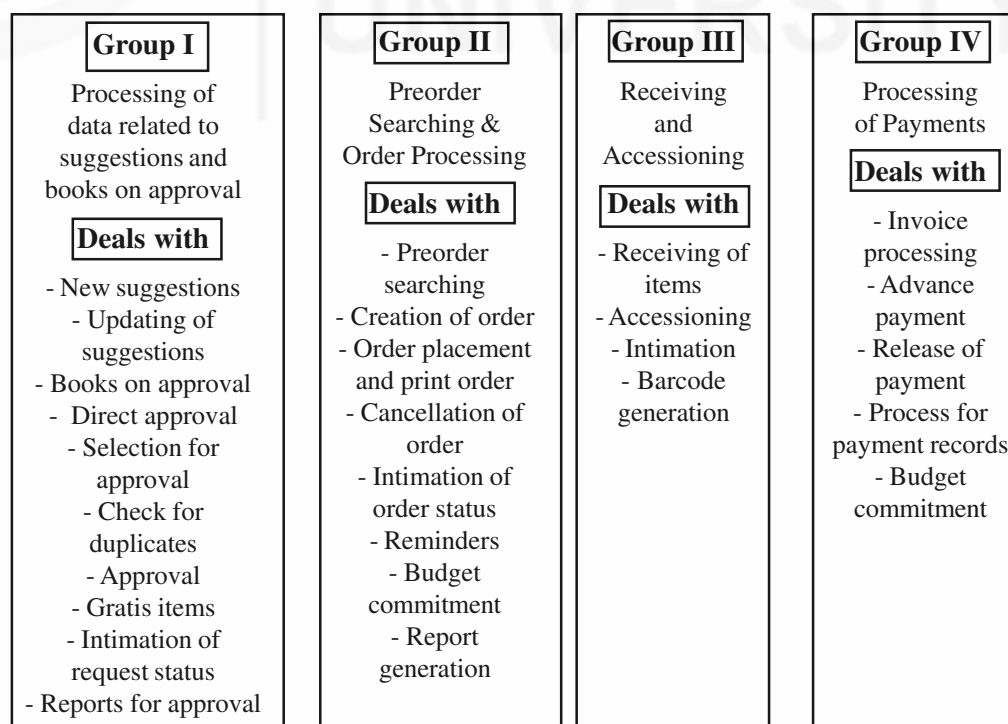


Fig. 2.4: Workflow of acquisition work

Group II tasks

The first step of this group is to select listed vendors (available from master files) for placing orders of approved documents. Order letters are then printed as per the format created in the pre-acquisition stage indicating name of supplier with address, reference number, terms and conditions and expected date of delivery etc. This group also includes the tasks of reordering, reminder generation (for a particular order or to a particular supplier/publisher) and report generation (for ordered items, overdue orders, budget commitment etc.).

Group III tasks

This group includes the works of receiving and accessioning of ordered documents. In case of barcode based circulation system barcode labels for accessioned items are also generated in this sub-module of the package. The requester or department may be informed about the arrival of requested documents in the library through the generation of intimation letter.

Group IV tasks

The work of this group starts with the processing of invoices submitted by the suppliers along with the documents by entering necessary elements into the database. Release of payment is the next step in which letters/reports containing all the necessary administrative and financial details are generated against supplier or order number or invoice number for requesting appropriate authority (generally Finance Section) to release payment to the supplier. After release of payment, the financial details of payment are entered and stored into the database.

2.3.3 Products and Advantages

Computerised acquisition subsystem includes three basic operations – input, processing and output. Data entering and processing tasks in various pre-acquisition and acquisition works are primarily act as input data. The datasets are processed and integrated with other modules of the ILS and finally generated various outputs in the form of list, reports, letters and statistics. Table 3 in the next page lists all the possible reports from acquisition module of ILS. The advantages of computerised acquisition subsystems in an integrated automated environment are manifolds. Such systems can perform following activities:

- Generate financial and statistical reports in the desired format automatically to help planning and management of libraries;
- Ensure quicker and cheaper data processing;
- Contribute in the development of integrated library system by integrating with document processing module (to transfer bibliographic data) and member module (for helping online requisitions/suggestions from members);
- Reduce the workload of processing section by transferring manifestation and item related information related with documents received (modern ILS supports MARC 21 based item processing framework mainly through 9xx series on the basis of FRBR model);
- Minimise routine clerical operations and related paper works;
- Lead towards better management and more productive use of library staff;
- Support real time fund accounting and help to introduce new user services;

- Produce number of reports, letters, statistics and list to support MIS activities of libraries;
- Interact with other library systems/networks to download bibliographical data of items on order on the basis of global standards related to electronic fund transfer; and
- Communicate different outputs of acquisition works electronically to members, suppliers, publishers etc.

Table 2.3: Reports from Computerised acquisition subsystem

<ul style="list-style-type: none"> • List/Report of item(s) requested • List/Report of item(s) from supplier/publisher • Item(s) selected for approval • Item(s) approved by the authority/library committee • Item(s) rejected in the approval process • List of gratis item(s) received by library • Report on request status • Printout or softcopy of letters for approval • Printout or softcopy of order letters & query letters • Printout or softcopy of reminder letters • Printout or softcopy order cancellation letters • Printout or softcopy of reordering • Letters for adjustment of advance payment • Letters to bank for foreign exchange rate • Report on order status • List/Reports of item(s) selected for order 	<ul style="list-style-type: none"> • List/Report of overdue item • List/Report of item(s) actually ordered • Reports of budget commitment • List/Reports of item ordered against advance payment • List/Reports of item(s) received against orders • Letters of intimation (on arrival of documents) • Printout of accession register • Printout of barcode labels • List of supplier/publishers • List of currency and exchange rates • Budget with commitments • Report of detailed annual budget of library • Report on amount received in different budget heads • Report/statistics of vender performance • List of recent additions • Generation of book cards (in case of integrated ordering and cataloguing system)
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

4) What do you mean by Pre-acquisition work?

.....

.....

.....

.....

.....

5) Point out the major advantages of automated acquisition subsystem.

.....

.....

.....

.....

2.4 DOCUMENT PROCESSING SUBSYSTEM IN ILS

In automated document processing environment, resource description or cataloguing is possibly the most important task of library automation work. It requires standardisation and should be supported by carefully crafted decision table(s). The cataloguing module of ILS gives us freedom to choose MARC standards (UNIMARC and MARC 21) or Non-MARC standards (like Common Communication Format or your own standard). However, MARC 21 bibliographic format is now considered as the global de facto standard. MARC 21 family of standards (a family of five coordinated standards such as bibliographic standard, authority standard, community information standard holding format and classification format) are now selected as content designator in most of the ILSs. There are two reasons for it. First, MARC 21 standards are updated continuously, available through Web, and emerging as open standards. Secondly, these are now becoming almost the de facto global standards in the domain of library automation as these are adopted by the national libraries in different parts of the world. Cataloguing module of an ILS should also be supported by an array of internationally agreed upon standards and facilities like – FRBR, FRAD, pickup lists, authorised value lists, standard lists, export-import through ISO-2709 or MARC-XML etc. This section discusses automated document processing subsystem under three major heads – 1) Functional requirements, 2) Workflow, and 3) Advantages and products.

2.4.1 Functional Requirements for Document Processing in ILS

The functional requirements of cataloguing module of an ILS (as suggested by Mukhopadhyay, 2006) include areas like authority data, bibliographical data, distributed cataloguing, OPAC, reports, backup and restoration, export and import, and multilingual data process and retrieval.

Authority Control

The ILS must support following facilities for managing authority data:

- Support for MARC authority format for personal, corporate and topical name headings in a name authority file; title, uniform title and series entries in a title authority file and subject headings in a subject authority file;
- Provision for generation of SEE, SEE ALSO references and NT-BT-RT relationships network from authority records and link these references to matching access points in OPAC;
- Must allow any bibliographic field to be authority controlled (particularly 1xx, 6xx and 7xx groups in MARC 21 bibliographic format) and should include facilities to search, retrieve, and display print and global editing of authority records by authorised operators;
- Must include provision for multiple thesauri with the ability to produce a list of all citations with authority file violations; and
- Provision to link local catalogue data with global linked open authority data like VIAF (a service merging authority data from 25 national libraries available from viaf.org).

Fig. 2.5: MARC 21 authority data entry framework (name authority) in Koha ILS

Bibliographic Control and Interoperability

The bibliographic record management capabilities of an ILS should extend support for –

- MARC 21 bibliographic and authority framework for processing bibliographic data including multilingual data processing support (Unicode character set processing ability);
- MARC record loader that can accept records input from various sources and from various media like tape diskette or over network;
- Global editing utility that find and replace data within specified fields;
- Data format validation during input of bibliographic data;
- MARC 21 format for holding and display of holding on the basis of ANSI Z39.44 serials holdings display format;
- Import of bibliographic data through Z39.50 compliant distributed cataloguing interface; and
- Interoperability and crosswalk through incorporation of XML, RDF and metadata schemas (e.g. Dublin Core Metadata);

Some tags and subfields of bibliographic framework(s) require support for achieving standardisation in data entry activities. For example, the Leader fields (24 character fixed length field) in MARC 21 is necessary for different document types and the process of entering data for different character positions is quite complex. For example, the following tags and subfields of MARC 21 bibliographic format require support of pickup lists, code lists, standard lists etc. during data entry activities:

Field	Description	Type of Support
Leader	24 characters fixed-length field	Pickup list for character positions
005	Date and time of latest transaction	Automated entry of date and time from system
006	Books – (00-17) – Fixed-length field	Pickup list for character positions
007	Text - (00-01)	Pickup list for character positions
008	Fixed-length data elements	Pickup list for character positions
040	Cataloguing Source	Pickup of library code (as per MARC)
041	Language Code	Code list support (as per MARC)

Fig. 2.6: Support to manage Leader field (24 character positions) in Koha ILS

Online Public Access Catalogue (OPAC)

- OPAC must be fully integrated with other modules and accessible through web-based client;
- OPAC should provide browse indexes for author, title, and series and browse index combining all four indexes;
- Should support searching different forms of authorities;
- It should allow combined, specific and field level searching for all formats along with phrase searching, nested searching and truncated searching;
- It must enable searching by using Boolean operators (OR, XOR, NOT, AND), positional operators (SAME, WITH, NEAR, ADJ) and relational operators ('greater than', 'less than', 'equal to', etc.) within and across all fields including provision for Fussy searching;
- It should provide facility to see processing status (fully catalogued, in process, lost, withdrawn etc.) and circulation status (in transit, reserve, recalled, on-hold etc.);
- OPAC should support full, brief, standard and customised display of records including relevancy ranking of search results;

- OPAC should also support bulletin board, information desk and gateway services (to access external databases) along with patron self-service options (e.g. holds, renewals etc.); and
- OPAC must track users' preference and interests, organised into a list of favourites and support interactive, participative and collaborative platform through web 2.0 tools like RSS, social networking tools, user tagging, document rating etc.

Distributed cataloguing

- Must be Z39.50 compliant cataloguing system [ANSI/NISO Z39.50 (1995) or ISO 239.50 (1998)];
- Should enable to capture bibliographic and authority records from any Z39.50 server through Z39.50 client; and
- Should allow local manipulation (change of call number etc) of captured data.

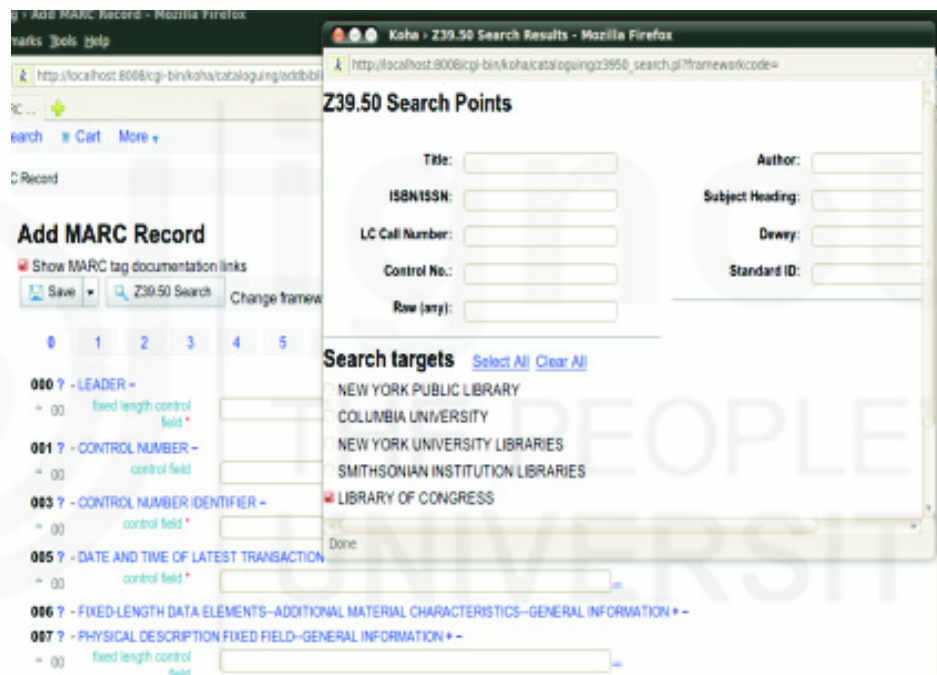


Fig. 2.7: Z39.50 client to support distributed cataloguing in Koha ILS

Reports and backup requirements

- Must produce a count of all records added, edited by a specific operator or over a specified time period;
- Must generate lists, statistics and counts of items added or tabulated by call number, item categories, item location, holding library etc.;
- Must produce a list of all citations with authority file violations; and
- Must support backup of all cataloguing records in suitable media (magnetic, optical etc.) and easy recovery of records at the time of need.

2.4.2 Workflow of Automated Document Processing

The workflow of document processing subsystem involves two major jobs – bibliographic data management and authority data management. Bibliographic data are managed in two basic modes – 1) cataloguing data entry for newly

acquired library materials processed in acquisition module; and 2) cataloguing data entry for existing library materials not processed through acquisition module (also known as Retrospective Conversion or ReCon). The works of cataloguing module of an ILS are –

- Authority data management
 - 1) Authority data entry
 - Name authority
 - Subject authority
 - Title authority
 - 2) Authority data linking
- Bibliographic data management
 - For newly acquired document
 - For existing old stock

Bibliographic Data Entry for Cataloguing

This facility of the catalogue module of automation packages is utilised for updating and standardisation of bibliographical data elements of newly procured documents and entering bibliographical data of existing old stock of the library. Easy and structured data entry form design on the basis of standard content designator scheme is important for local creation of records. An integrated automation package use the same record for cataloguing function as is used in the acquisition module. In the catalogue module the record is standardised through entering additional data elements and rendering of access points with the help of authority file. The transformation of bibliographical data elements of existing stock of any library into machine-readable form is called Retrospective Conversion or simply RECON. The work of RECON starts with recording of bibliographical data elements on a worksheet. The worksheet is designed as per the internal data format of the automation package. These internal bibliographic data formats are based on internationally adopted standard content designator schemes such as MARC 21, UNIMARC or CCF. Finally bibliographical data of each document as recorded on the worksheet is entered into the catalogue database. The data entry work may be done by the library staff or the job may be done through outsourcing. In some cases library may procure validated MARC 21 bibliographic data from the following sources –

1) Existing library catalogue in machine readable form

Bibliographic data in standard formats (MARC, UNIMARC, USMARC, CCF, MARC 21) are available in many libraries for merging into the catalogue database of newly installed LMS through import (ISO-2709 based exchange of bibliographic data).

2) Union catalogue

Library networks at the global level (like OCLC, RLN) and national level (like INFLIBNET and DELNET in India) provides union catalogue of member libraries in machine readable form. Union files of the stock of several libraries, or another shared database may be imported, converted into local standard format and finally merged into the catalogue database.

3) Commercially available files of MARC records

In this process records from external databases may be added from tape, or by downloading directly from the files through network. A further option is to acquire records on CDROM or DVDROM and to download records from optical media. For example Harvard University, US recently uploaded all bibliographic records in MARC 21 format (2 million book records) for other libraries.

4) Z39.50 server

Computerised cataloguing provides a unique advantage of loading and merging of bibliographic and authority records from external databases. There are thousands of Z39.50 servers from where selective downloading of validated bibliographic data may be done at the local level (see Fig. 7). This feature of an automated system leads to a reduction in cataloguing effort and a consequent saving in the unit cost of cataloguing. This mode of shared cataloguing is popularly termed as copy cataloguing and implemented in ILSs through Z39.50 standard developed by ANSI/NISO.

Authority Data Entry for Cataloguing

A library catalogue supports two basic functions – finding function and collocation function. Bibliographic datasets support finding function and authority datasets support collocation function. Therefore, authority file is essential to control from of index terms or headings, such as author headings, or subject index terms for better retrieval efficiency. Authority data management has two basic routes – internal dataset creation and external dataset application. Records in this file may be created locally by using a standard authority data framework standard like MARC 21 authority data format (see Fig. 2.5) or drawn from externally available files such as the name and subject authority files of the Library of Congress or other agencies. Library automation packages provide facility to create and maintain authority file in the catalogue module. This file is acting as a master database, where entry is to be made once. This gets reflected in various modules of the package. The master file containing authority entries can be consulted



Fig. 2.8: Authority data types in Koha ILS

during cataloguing, possibly by display in a separate window and new headings are immediately added to the authority file with an opportunity to review or authorised locally or remotely. For example, Fig. 2.8 shows the authority data entry options in Koha ILS. Selection of authority data type will display corresponding authority data entry framework (as Fig. 2.5 shows name authority data entry format) for processing work.

Alternatively libraries may take advantages of cooperative authority datasets like LoC authority data, NACO, SACO and VIAF –

Name Authority Cooperative Program (NACO)

It is one of the components of the Program for Cooperative Cataloging (PCC) that was initiated in 1995 by the Cooperative Cataloging Council (CCC) in the USA (PCC, 1998). The NACO program enables participants to add name authority records to the national name authority file, which is hosted at the Library of Congress and downloading of authority data from the server.

Subject Authority Cooperative (SACO)

The SACO program allows cataloguers to propose new and updated authority records for inclusion in Library of Congress Subject Headings (LCSH) and the LC/SACO Authority File. SACO is also working under Program for Cooperative Cataloging (PCC).

LoC Authority Data Service

Library of Congress Authority datasets allows to browse and view authority headings for subject, name, title and name/title combinations for bibliographic and other materials available in LoC. It also facilitates downloading authority records in USMARC/MARC 21 format for use in a local library system. This service is offered by LoC free of charge.

Virtual International Authority File (VIAF)

VIAF is a new, international service designed to provide convenient access to the world's major name authority files from 25 national libraries under the leadership of OCLC (limited in the initial stages of the service to names for persons). Its creators envision the VIAF as a Linked Open Data (LOD) for linking in local services like ILSs. An ILS can link VIAF automatically from authority data entry interface through application program interface.

2.4.3 Products and Advantages

OPAC is possibly the most visible product of document processing subsystem of an ILS. But it is not the only one. This subsystem produces different other forms of library catalogue like Card catalogue (main entry and added entries), Printed book catalogue, Microform and Computer output on microform. ILS supports the generation of various reports, lists and labels that are required for the management of catalogue section such as Reports with a count of all records added, modified or edited by a specific operator or over a specific period of time; Reports that produce statistical account of items added and tabulated by call number, item categories, item location etc.; Lists of items catalogued by class number, subject heading, collection type, language etc.; Spine labels, shelf catalogue, book cards etc.. This module of ILS also generates information products

that form the basis of a number of user services such as bibliographic service, current awareness service etc. These are typically – List of books received in the library (during a particular period, on a particular subject, by a particular author or by a particular author on a particular subject in a particular period) and Bibliographies of documents received by the library in standard format or as per the format specified by users. Modern OPACs are changing from monologue to dialogue based service by the applications of Web 2.0 tools, federated search mechanism and discovery services (see section 1.7 of Unit 1 in this block).

The application of advance level ICT in the management of library processes leads to a significant change in the nature and role of catalogue records. The impact of these changes has contributed towards standardisation of entry format, resource sharing and efficient access to documents and their contents. For example Web-OPAC overcomes two fundamental barriers of access to information – time and space (anyone can search from anywhere at any time). In an integrated set up circulation module and acquisition control programs utilise cataloguing records. Similarly catalogue module uses bibliographical data elements of records created in acquisition procedure and also utilises transaction records from circulation control to notify users about the availability of a selected document. The other advantages of automated document processing (as identified by Mukhopadhyay, 2006) are –

- Computerised cataloguing ensures greater standardisation in catalogue records;
- It reduces routine clerical operations required for maintenance of catalogue;
- It supports interchange of catalogue records and thereby ensures reduction in unit cost of cataloguing;
- It supports seamless access to not only library resources but also web resources, OPACs of other libraries, online databases and a variety of information services including subject gateways through federated search mechanism and thereby ensues a single-window access interface for users;
- It provides opportunities to take output in a number of forms and formats;
- It enables users to retrieve relevant records through the application of variety of search techniques and search operators and to display the retrieved records in desired formats; and
- It helps library staff to generate variety of information services.

Self Check Exercises

- Note:** i) Write your answers in the space given below.
ii) Check your answers with the answers given at the end of this Unit.
- 6) What is distributed cataloguing? How can it help libraries?

.....
.....
.....
.....
.....

- 7) Discuss the MARC 21 family of standards.

.....

.....

.....

.....

.....

2.5 SERIALS CONTROL SUBSYSTEM IN ILS

International Serials Data System (ISDS) defined *serial* as a publication issued in successive parts and intended to be continued indefinitely. Serials include periodicals, newspapers, annuals, proceedings, transactions etc. and are differentiated from monographs by their ongoing or continuing nature. Serials management subsystem of an ILS has to deal with the features unique in serials control such as – Periodicals are procured through various subscription modes and by gift or exchange; Successive issues are received at regular or irregular intervals and it is necessary to ensure that successive issues arrive when they have been published; Subscriptions to periodicals must be renewed recurrently; Catalogue data that describe serials must be extensive and should be supported by formats exclusively designed for serials; Serials change their titles, are published under variant titles and may change their frequency of publication, therefore, references must be inserted to link associated periodical titles; Precise control over the binding of successive issues is very important (alternatively called as backvolume management); Indexes, special issues and supplements must be controlled for effective retrieval; and Article-indexing is an added advantage for serials control module.

2.5.1 Functional Requirements for Serials Control in ILS

In view of the foregoing, you can now understand that the serials control subsystem of ILS which attempts to provide mechanical means for checking in serials issues, issuing claims, handling binding and other such functions has to be designed very carefully because of the complex nature of serials management. The serials control module of ILS should meet following functional requirements (Mukhopadhyay, 2006):

- New subscription
- Renewal of subscription
- Cancellation of subscription
- Budget control
 - Department/unit-wise budget
- Invoice processing
 - Invoice for individual issues, or for annual (or other period) subscription
- Recording the receipt of journal issues
 - Formula for generating expected issues (predictive mode of serials control)
- Managing (sending claims for) missing issues
 - Sending reminders

- Support for domain-specific bibliographic format like MARC 21
- Needs to be able to cope with “special editions”, supplements, and indexes
- Should also be able to cope intelligently with name changes (of publication, publisher) and merges or splits (i.e., one journal becomes two, or two join together)
- Binding control
- Accessioning bound volumes
 - Barcoding of accession numbers
- Complete holding information for individual title
- Report generation
- Listing the periodical for browsing
 - Hyper linking the e-journals from publisher’s sites or consortia sites
- Editing and updating of records
- Searching in OPAC
 - By title
 - By publisher
 - By distributor
 - Sorting by date or volume/issue number
- Printing of holdings of periodicals and supporting Routing of periodicals
- Options for display holdings and receiving of serials in Web-OPAC
- Table of contents and other personalised information services
- Article indexing (The serials control module should support indexing of journal articles by author, title, and subject keywords)
- Union list and union catalogue (In union catalog the complete holdings information is given along with all its missing issues, discontinuation in subscription, changes in title etc.).

2.5.2 Workflow of Automated Serials Control

The basic workflow of serials control subsystem in ILS may be grouped into four subdivisions – 1) Creation and maintenance of the master database; 2) Subscription and acquisition; 3) Cataloguing and article indexing; and 4) Circulation and binding. These four basic groups of activities include series of tasks. Obviously, the procedures, activities and tasks related to serials control requires frequent and repetitive record addition or amendment. Computerisation is an attractive proposition for serials control because of this reason.

Group I: Creation and Maintenance of Master Database

In serials control module of an ILS, master databases play important role. Any number of addition, modification and deletion is possible in the master database and these changes are automatically reflected in all the sub-modules under that module. It reduces data entry work and ensures standardisation. A typical serials control module includes:

Title master

In this file bibliographical details of new serials are entered (on the basis of standard comprehensive data format like MARC 21 bibliographic format) after the selection and approval process.

Country master

This file contains name of countries and their corresponding codes for entering country of publication data in sub-modules of serials control. Country code is generally based on ISO-3166 where each country is represented by two unique characters e.g. the code of India is **in** as per ISO-3166.

Language master

Now in most of the cases MARC 21 geographic area code (GAC) is used for the purpose. But this file may also contain entries for languages and their three digit codes as per the ISDS manual and CCF manual.

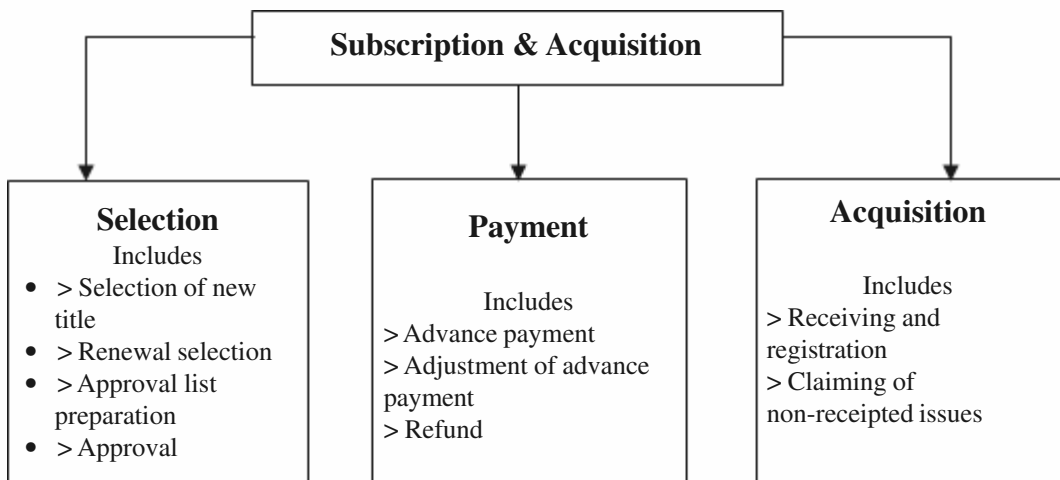
Supplier/Publisher/Binder master

This master file contains details of all local and foreign subscription agents, publisher of serials and binders along with their corresponding codes. These codes are generally created locally.

The above mentioned master files are essential and the other important master tables are – 1) Subject master (holds lists of subject descriptors); 2) Frequency master (holds codes for serials frequencies); 3) Budget master (holds financial data necessary for serials acquisition); 4) Currency master (contains currency description, codes and exchange rate for foreign currencies); 5) Delivering mode master (contains different modes of delivery of serials by publishers and vendors); 6) Physical media master (holds forms, formats and media for serials in coded form); 7) Binding type master (contains different modes of binding (e.g. standard, lather binding, cloth and rexin binding etc.) and their corresponding codes); 8) Letter master (includes formats for every type of letters required for the generation of outputs such as order letter, cancellation of order letter, reminder letters etc.).

Group II: Subscription and Acquisition

The tasks of this group may be organised into three groups and may be represented diagrammatically as below:



All together, there are 12 basic works in this group of works related to serials control given in the sequence – 1) Selection of serials for new subscription; 2) Renewal or discontinuation of existing journals/serials; 3) Selection of delivery mode; 4) Selection of subscription mode; 5) Formulation of terms of procurement; 6) Selection of vendors; 7) Approval from authority; 8) Ordering and renewal; 9) Payment; 10) Receiving and registration; 11) Reminder generation; and 12) Adjustment of advance payment for non-receipted issues.

Group III: Cataloguing and Article Indexing

The major jobs of this group are –

Cataloguing

Cataloguing formats for serials are fundamentally similar to those of monographs. But the content and format of serials bibliographic records varies considerably between systems. Some catalogues are based on ISBD(s) and others on ISDS formats. Some cataloguing systems use local formats and some use standard format like MARC 21, CCF/B, UNIMARC etc. You may consult the Table 4 in next page for a set of minimum essential tags and subfields related to serials from MARC 21 bibliographic format.

Article indexing

Article indexing option is generally requires by libraries in research institutes. Indexing of articles (also called papers) from journal issues is an optional facility of serials control subsystem. Generally, publishers of primary periodicals produce annual and other sorts of indexes regularly. Apart from such products, libraries also subscribe to number of indexing and abstracting journals related to the areas of their interest. As a result, article indexing is only necessary when available indexing and abstracting services do not cover the core journals on discipline of interest.

Leader 24 characters fixed-length field

00X group

	Control Fields
005	Date and time of latest transaction (NR)
006	Serials – (00-17) – Fixed-length field (R)
008	Fixed-length data elements – General information (NR)

0X0 group

	Number and Code Fields
022	ISSN (R) [##; \$a (NR)]
040	Cataloguing Source (NR) [##; \$a (NR)]
041	Language Code (NR) [0/1_; \$a (NR)]
042	Authentication Code (NR) [##; \$a (R)]
043	Geographic Code (NR) [##; \$a (R)]
082	DDC (R) [0#; \$a (R), \$b (NR), \$2 (NR)]

2XX group

	Title Related Fields
210	Abbreviated Title (R) [0#; \$a (NR)]
222	Key Title (R) [#0; \$a (NR)]
245	Title Statement (NR) [00; \$a (NR), \$c (NR)]
246	Varying Form of Title [14; \$a (NR)]
260	Publication, Distribution etc. [##; \$a (R), \$b (R)]

3XX group	Physical Description etc. Fields
300 Physical Description (R)	[##; \$a (R), \$b (NR), \$c (R)]
310 Current Publication Frequency	[##; \$a (NR)]
362 Dates of Publication etc.	[1#; \$a (NR)]
5XX group	Note Fields
500 General Note (R)	[##; \$a (NR)]
6XX group	Subject Access Fields
650 Subject Added Entry-Topical Term (R)	[#0; \$a (NR), \$v (R), \$s (R)]
653 Index Term – Uncontrolled (R)	[##; \$a (R)]
7XX group	Added Entry Fields
710 Added Entry – Corporate Name (R)	[1#; \$a (NR), \$b (R)]
770 Supplement/Special Issue Entry (R)	[0#; \$a (NR), \$t (NR), \$x (NR), \$w (R)]
780 Preceding Entry (R)	[0-0/7; \$a (NR), \$t (NR), \$x (NR), \$w (R)]
780 Succeeding Entry (R)	[0-0/8; \$a (NR), \$t (NR), \$x (NR), \$w (R)]
841-88X group	Holdings, Location, etc. Fields
850 Holding Institution (R)	[##; \$a (R)]
852 Location/Call Number (R)	[##; \$a (NR), \$b (R), \$c (R)]
856 Electronic Location and Access (R)	[##; \$u (NR), \$s (R)]

Table 4: Data elements (minimum) for serials on the basis of MARC 21 bibliographic format (R=Repeatable field and NR= Non-repeatable fields)

Group IV: Circulation and Binding

This group includes following jobs –

Circulation

Circulation of serials is often referred as Routing of journals. Circulation pattern of serials differs largely from that of books. But if serials are available for ordinary loan, then the same circulation control system will suffice as for monographs. However, serials are generally reserved for reference use only. In special libraries, the short time loan options for journals are common because of the specific need of users. If the number of transactions per day is large enough then such transaction system may be computerised. Such computerised facility must have a list of serials taken, a list of users and their addresses, and transaction interface with options for the generation of required output.

Binding

Back volume management is an important job in serials control. It is a valuable feature of computer based serials control subsystems to inform the library staff of volumes that have been completed and are now ready for binding. It is a very helpful feature to assist in work scheduling and to spread the binding load to give an even distribution of work in the binding throughout the year. After binding of back volume of a journal, accessioning is done for the bounded volume and then holding information for the concerned journal is changed / modified in the bibliographic database of journals.

2.5.3 Products and Advantages

The output of products of an automated serials control subsystem may be grouped into three basic categories – OPAC (gives search option for journals, journal articles and journal holdings), Reports and lists (provides status reports and MIS reports for decision making) and information products (such as table-of-contents and other altering services including SDI). OPAC of an ILS allows searching serials by Title (Current title, Complete holdings, Key title, Linked title, Variant title), Subject (Broad subject heading, Subject divisions, descriptors and class number), Publisher, Title history (Title split, Title merge, Title change, Title holdings), ISSN and Free text. Several reports, letters and statistics can be generated by the automated serials control system such as List of suggestions, List of approved titles, List of titles ordered, List of issues received, List of non-receipted issues, List of missing issues etc. In serials control module of an ILS information products are originated either from article indexing activities or serials catalogue database and produced on demand such as List of recent arrival for issues of a group of journals (as selected by users), List of journal available on a particular discipline, Discipline-wise holding list of serials, Table of contents service of a group of journals (as per user selection), Compilation of on demand subject bibliographies, CAS and SDI services in online and offline mode etc.

Serials management is a complex process. This subsystem involves frequent and repetitive record addition or amendment. Computerisation is an attractive proposition for serials control because of this reason and it leads to following advantages –

- Generates various reports in required formats for MIS activities as decision support tool for serials control (requires for addition, deletion and continuation of journals);
- Ensures timely reminders generation for missing issues and better binding control for completed volumes;
- Offers easy and simple solutions for fund accounting, payment management and budget control, a critical requirement for serials control;
- Facilitates creation and maintenance of article indexing database and thereby generates number of user services on demand;
- It helps library staff in quick production of serials holdings and list of recent arrivals in many forms;
- Facilitates online access to the serials database from anywhere at any time in any format;
- Predicts the arrival of journal issues and generates schedules for receiving journal issues;

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

8) Discuss Kardex management in serials control module of an ILS.

.....

- 9) What is a predictive mode of serials control? Discuss its advantages in library automation?

.....

.....

.....

.....

.....

2.6 CIRCULATION SUBSYSTEM IN ILS

Circulation module of ILSs are effective tool for managing issue, return, renew, reservation and fine calculation easily and quickly. A circulation subsystem in ILS records loan transactions to specify – What material is in the library stock or readily accessible on ILL; Which material is in loan, and from whom or where it can be retrieved and When materials on loan will next be available in library for other users. In ILS, the transaction or loan database is the core of circulation subsystem. This database comprises a series of records, one for each transaction. Each record includes a brief dataset that specifies details of the document (through document number such as accession number), details of the user (through membership code) and transaction details (e.g. date of issue & date of return are extracted from the system date, and due date is calculated automatically). In an integrated setup, the bibliographical details (e.g. author, title edition, place and year of publication) of documents on loan are extracted from the catalogue database and the membership database is utilised for collecting user information. Accession numbers of documents are used as the key data elements in first case, whereas membership codes act as pointer to the member database in the second instance. Data-capturing is generally based on barcodes (to encode/decode both accession number for books and member ID from member card) but the use of RFID technologies in circulation are increasing significantly even in libraries of developing countries.

2.6.1 Functional Requirements for Circulation in ILS

Computerised circulation subsystems generally perform a group of functions utilising three basic categories of information – Information about the borrower; Information about the resources being borrowed; and Information about the loan transaction. An automated circulation system should provide facilities for managing the above mentioned three categories of information including following support services – 1) To locate circulating items (on loan, reserved by user, at binding, being reprocessed); 2) To identify items on loan (to a particular borrower, to a specific class of borrowers; 3) To record ‘personal reserves’ for items on loan but desired by another borrower and to issue alerting notice to the library staff on return of the reserved item by a borrower; 4) To print recall notices (for returning overdue items, for renewing of items); 5) To arrange renewal of loan; 6) To notify to the library staff of overdue items and printing of overdue notices; 7) To calculate fines or overdue charges for generating (printout of fine notices, receipts of fines records, printout of fine receipts); 8) To generate statistical reports (document related, user related, top ten items by popularity, top ten user by circulation activity etc); 9) To extend provision for handling special categories

of borrowers and special types of materials; 10) To generate and print gate pass and due date slips; 11) To act as decision support system for better circulation management; 12) To support various data capturing devices e.g. barcode readers, smart card and RFID equipments; and 13) To extend facilities for ILL and maintenance activities.

2.6.2 Workflow of Automated Circulation

The workflow of automated circulation subsystem starts with defining library circulation rules. Modern ILSs supports branch management system in circulation. It means if a library has branches, each branch may have their own circulation rules and one circulation module will serve all the branches on the basis of circulation rules of that branch. Circulation rules match patron category with item types by defining number of checkouts, loan days, fine amount, grace period, number of renewals, number of reservations etc.

The screenshot shows the Koha ILS interface for defining circulation rules. The main heading is "Defining circulation and fine rules for 'Central Library, BU'". Below this, there is a list of bullet points representing different rule categories. A table is provided for defining specific rules, with columns for Patron Category, Item Type, Current Checkouts Allowed, Loan Period (day), Hard Due Date, Fine Amount, Fine Changing Interval, Fine Grace period (day), Suspension in Days (day), Renewals Allowed (count), Holds Allowed (count), and Rental Discount (%). Below the table, there is a section for "Default checkout and hold policy for Central Library, BU" with input fields for "Total Current Checkouts Allowed" and "Hold Policy".

Fig. 2.9: Circulation rules setting option in Koha ILS

The other broad groups of activities for the workflow of automated circulation are:

Membership Management

This sub-module is basically meant to create and update membership records in a library. The works of this sub-module are – 1) Master database creation and maintenance facility; 2) Member category and privileges management; 3) Institute

profile and profiles of Departments/Divisions under the institute; 4) Calendar to record weekdays and closed days for library; 5) Member enrollment facility including modification/deletion/renewal of membership; 6) Output generation facility.

Transaction Management

Transaction sub-module includes all the day-to-day activities of circulation section of a library vis. issue, return, renewal, reservation, reminders for overdue books, searching document availability and listing of items issued to a member.

Reminder Generation

This facility is meant for generating reminders for overdue documents – To a group of members, To individual members, For a particular due date, To all members. The format and text of reminder letter may be modified by using this facility or by using the master database.

Fiscal Management

It provides option to manage outstanding dues against a member. It also includes generation of payment receipt. Fine amount may be waived by authorised staff. This facility should also allow printing of fine statement if a member wants to have a statement of fines.

Inter Library Loan (ILL)

Inter library loan method simply means that documents of a library can be issued to the members of other libraries. ILL activities of an ILS are - ILL membership management; ILL transactions management; and ILL supervision.

Maintenance

Maintenance is generally attached with circulation module for recording information about lost documents, documents sent for binding, damaged documents, missing documents and documents withdrawn from library.

2.6.3 Products and Advantages

The typical products or outputs from automated circulation subsystem in an ILS are –

- List of library members (list of members can be printed either by name or by member code and can be sorted on any required sequence or order);
- Items issued over a period (list of documents issued on a particular date or date range);
- Items returned over a period (list of documents returned on a particular date or date range);
- Items reserved over a period (list of documents reserved on a particular date or date range);
- Member ID card (Member ID card with name of the member, membership code, department, institute, category, branch and year may be printed by utilising appropriate facility); Fig. 2.10 shows the member card generation utility in Koha ILS. You can observe the ability of the ILS to convert member ID into corresponding barcode.

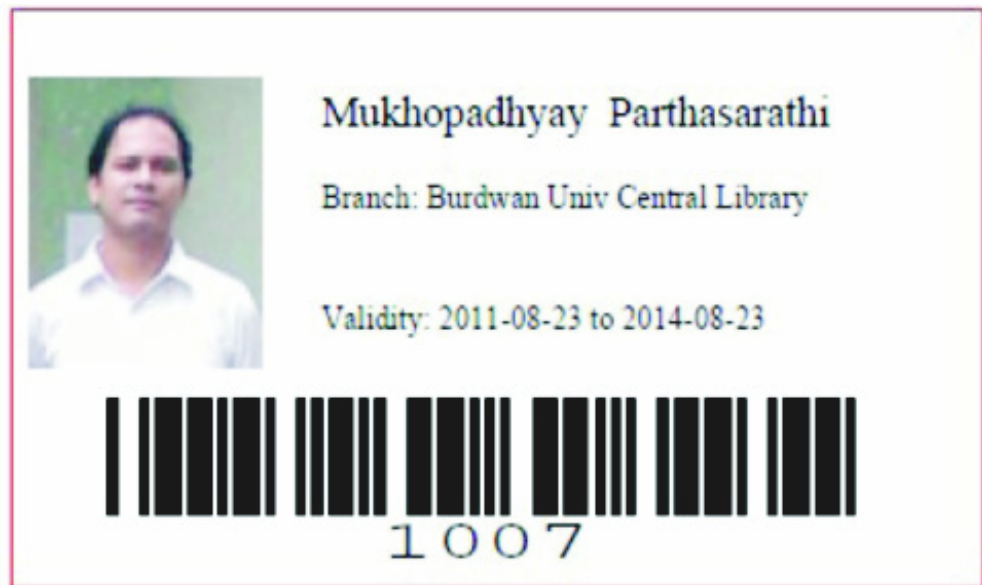


Fig. 2.10: Bar-coded member card generation in Koha ILS

- Reminder letters and notifications (preformatted reminder letters for overdue document(s) is a regular task of circulation section);
- Item's transaction history (transaction history of any particular document);
- Membership expiry list (list of memberships expiring on a particular date or date range);
- Member history (list of documents issued and returned by a member during his/her membership period);
- Fiscal report (details of the fines collected by the library on a particular date or date range);
- Library usage (usage by deferent category of library members or by usage of different types of library materials);
- Most frequently issued items (list of most frequently issued documents);
- Most frequent member (list of most frequent users by circulation activities).

The other important products are –

- List of items issued to a member;
- 'No dues' certificate;
- ILL reports (arrival intimation, reminder, list of items on ILL, overdue charges and payment receipts);
- Transaction details undertaken by a staff working at circulation;
- List of lost, missing or damaged documents;
- List of lost documents for which amount recovered;
- List of documents sent to binding;
- Order letter for binding;
- List of withdrawn items.

The main advantage of automated circulation subsystem is the ability of library staff in extensive control of stock. Transaction records can be entered and saved

into the main database through a terminal. The central transaction database is updated immediately and subsequent consultation of the database will communicate the current situation. Some of the important issues may be enumerated as – Fines can be calculated on demand; Reservation and other modification to document records can be made instantly; Automatic identification of over borrowing and problem borrowers; Error-free data capturing through barcode, RFID and smart card technology; Provision of self-checking or self-issue option through web interface; Back up provision and exchange of circulation records on the basis of NCIP (NISO Circulation Interchange Protocol) standard.

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

10) “Automated circulation is fairly successful right from the eighties” – elucidate.

.....
.....
.....
.....
.....

11) Explain the use of RFID in automated circulation.

.....
.....
.....
.....
.....

2.7 SYSTEM ADMINISTRATION

System administration of ILS is not regular and repetitive in nature but the working of each modules of ILS activated after configuration of each module as per the requirements of library through system administration interface. System administration involves two sets of works – 1) Setting of initial configuration for each module; and 2) Adjustment of configuration settings from time-to-time to match requirements of library. Post-installation configuration of an ILS is required to make the default installation of ILS library specific. Only super user of ILS can set the administrative parameters. The typical system administration jobs are listed herewith –

General parameters

- Date format: Selection of “metric,” “us,” or “iso” date format for entire ILS (“us” = mm/dd/yyyy; “metric” = dd/mm/yyyy; “iso” = yyyy/mm/dd);
- Tax parameter: Setting of tax (generally in percentage) for acquisition of documents;

- Parameters for Authorities: Involves decisions regarding Authority Display Hierarchy and Authority separator;
- Default character encoding: Selection of character encoding standard for whole ILS, usually Unicode for multilingual data;
- Theme selection: Selection of themes for appearance for both librarian and user interfaces;
- Branch management: Option for setting managing parameters for library branches.

Cataloguing parameters

- Allows settings of the following parameters for cataloguing activities – default display format for retrieved documents, default data format (MARC, UNIMARC etc.), Auto/manual barcode generation, Filing rules etc.

Circulation parameters

- Allows parameters setting related to maximum outstanding fine amount, maximum reservations allowed, patron image display, notification for borrower expiry, generation of gate pass etc.

OPAC parameters

- Supports setting for the following parameters related to OPAC – enhanced content linking (like Amazon etc), suggestions by users from OPAC, virtual shelf management.

Library Branches: Options for setting library code, name, address, IP address, domain name etc.

Library Funds: Setting of budget heads for different library materials as per the decision of the authority;

Currencies: Define the currencies library deal with exchange rates.

Item Types: Setting “categories” into which library items are divided.

Borrower Categories: Setting definition for the types of users of library and how they will be given privileges.

Issuing Rules: Controls aspects related to the circulation of library materials.

Authorised values for bibliographic format: Options for setting list of authorised values for different tags and sub-fields of selected bibliographic format.

Bibliographic framework: Scope for customising of data entry framework by selecting require tags and sub-fields.

Printers: Setting of printers (or several printers) that is attached to ILS server.

Stop words: Provision to list all of the words library staff wish to ignore by ILS when performing catalogue searches or building the keyword index.

Z39.50 Servers: Adding Z39.50 servers library want ILS to search.

Export/Import: Settings for performing export/import activities by following standards like ISO-2709 and MARC-XML.

Backup/Restoration: Regular backing up databases and restoration at the time of emergency.

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

12) What do you mean by system administration? List some major jobs covered by this module.

.....
.....
.....
.....
.....

2.8 SUMMARY

This Unit starts with a theoretical discussion on system analysis and shows the application of procedural model to analyse tasks related to housekeeping operations under different sections of a library. It discusses library automation processes in integrated setup under four major subsystems namely acquisition subsystem, document processing subsystem, serials control subsystem and circulation subsystem. Each subsystem includes three major heads of discussion uniformly. The heads of discussion are functional requirements for the subsystem, workflow of the subsystem and advantages of automating the subsystem including typical products of the automated subsystem. Functional requirements section argues what an ILS should support and workflow section discusses how an ILS may be utilised for automating the subsystem. This unit ends with a discussion on system administration jobs related to library automation.

2.9 ANSWERS TO SELF CHECK EXERCISES

- 1) Library workflow or housekeeping operations are basic functions of any type or size of library. The works include acquiring, processing and preserving of library documents. The circulation of documents and maintenance of library stack is other important works of library housekeeping. These works are done through various divisions/sections of a library namely acquisition, processing, circulation, serials control and maintenance. These are basically routine and recurring works. Mechanisation of such works may be done through the application of ICT tools e.g. computer hardware and software (called ILS).
- 2) Serials control is concerned with the management of operations of journal section of a library. These are subscription, renewal, order, payment, check-in or receiving, reminder, binding and accessioning of bound volumes. Such activities lead to various information products and user services.
- 3) System analysis is technique for the analysis of components of a organisation and its works into atomic structure. Library is a complex system and consists of various subsystems and components. ASLIB, on the basis of system analysis techniques, identified a set of eighteen procedures related with

different subsystems. The same study also identified six common activities for all the eighteen procedures. These are – initiate, authorise, activate, record, report and cancel. All of these activities may not be applicable for each procedure. These procedures and activities are common to each type or size of library. An ILS should cover procedures, activities and tasks related to each subsystem of a library. Therefore, system analysis is a powerful tool for implementing an ILS.

- 4) Acquisition module of any ILS requires some essential works that need to be done before proceeding with actual acquisition work. These are termed as pre-acquisition works. This set of activities include – creation of master file for vendors/publishers/suppliers, creation and maintenance of currency conversion table, budget allocations under different heads, setting pre-defined letters for ordering etc, member creation and privilege setting.
- 5) Acquisition module of an ILS reduces a great deal of routine clerical chores in acquisition, supports online data entry and Electronic Data Interchange (EDI), generates reminders for overdue orders and sends them automatically over communication channel, provides real-time fund accounting, transfers bibliographical data of newly acquired items entered in the acquisition module to catalogue module for necessary modifications and up-gradation. Such a system helps to introduce new user services and cheaper data processing. It generates reports, statistics and lists required for the better library management and planning of efficient library services. Another advantage of automated acquisition system is to provide ready answers against queries related to the status of requests or orders.
- 6) Distributed cataloguing is a form of shared cataloguing and cooperative cataloguing. It allows online capturing of bibliographic data from remote library servers over the Internet. It reduces unit cost of cataloguing and saves lot of time for individual libraries. However, the major problem is of variation in data formats, software and hardware. ANSI/NISO Z39.50 standard was developed to support distributed cataloguing and to overcome the problems of database searching with different search languages. Z39.50 is a session oriented program-to-program open communication protocol based on client-server computing model. ILS incorporated with Z39.50 copy-cataloguing client (called origin in the standard) submits a search request to any Z39.50 server (called target), which then process the request and returns the result in desired standard. ILS will then place the captured record in the catalogue editor for changing and modifying bibliographic data in local library.
- 7) MARC 21 is a family of five coordinated formats developed 1999 through conciliation of major national MARC formats like USMARC, UKMARC, CANMARC etc. The five standards are namely - MARC 21 format for authority data, bibliographic data, classification data, community information and holdings data. MARC 21 is mainly a development over USMARC, and has become the de facto bibliographic standard in the area of computerised cataloguing since the beginning of 21st century.
- 8) Kardex management basically deals with loose issue management of journals in a library. It is also known as Cheek-in operation. It involves works related

to the receiving and registering of individual parts or issues of serials in library. It is necessary to make a careful note of the arrival of every issue of all periodicals along with special issues, indexes or other accompanying materials. Reminder generation for non-receipted issues depends largely upon this function.

- 9) Predictive mode of serials control means the ability of the ILS to predict the arrival of individual issue of a journal and to generate reminders automatically in case of non-receipted issues or parts within a stated time interval. An automated serials control subsystem may be predictive or non-predictive. A predictive serials control system saves labour, energy, time and money and ensures timely delivery/release of reminders for due issues of journals.
- 10) Circulation work of a library involves a group of operations that are specific, repetitive and systematic. As a result automated circulation systems have been fairly successful from the early days of library automation. Such systems require minimum set of essential data for carrying out circulation activities and data may be captured in a variety of ways. In an academic library, where users are generally large in number, this automated subsystem saves time of the users in great way.
- 11) Automated circulation subsystems are now-a-days RFID-enabled for many reasons. Libraries apply RFID (Radio Frequency IDentification) technology to manage un-manned self-service counters for issue and return of documents. An RFID system comprises three components: a tag, a reader and an antenna. The tag is paper-thin chip, which stores necessary bibliographic data. The tag is to be placed on the inside cover of the corresponding document. RFID reader and antenna are often integrated into patron self-checkout machines or inventory readers. The reader powers the antenna to generate RF field to decode information stored on the chip. Reader sent information to the central server, which in turn communicates with the ILS. RFID, apart from self-issue facility, also supports stock verification, theft detection (through EAS gate), and identification of misplaced books and inventory counts.
- 12) The administrator or super user should control the overall administration of ILS through a highly secured module for managing access control - for individual user, for each module and for each function; system security to prevent unauthorised access to databases; standard implementation and setting of system parameters and keep a log of each transaction, which alters the database. The other important jobs of system administration are privileges control, branch management, backup and restoration and System configuration.

2.10 KEYWORDS

- Backup** : Storage of records in magnetic or optical media for recovery of data at the time of need.
- Barcode** : A barcode is simply a computer readable tag that is used to identify individual items and patrons that are related to a specific library database.

- Boolean Operators** : The words AND, OR, and NOT used to combine concepts or search terms when searching a database for information.
- Budget Allocation** : It is the distribution of total library budget into various budget heads and subheads.
- Charging** : It is the act of 'issuing' a document and to record the loan transaction.
- Check-in** : The act of receiving and recording arrival of individual parts of serials.
- Common Communication Format (CCF)** : The CCF was developed by the General Information Programme (PGI) of UNESCO in order to facilitate exchange of bibliographic data between organisations, and first published in 1984. It is a highly compatible format that provides a structure in which records may be entered to the system; a format best suited to long-term storage; a format to facilitate retrieval and a format for display.
- Data field** : In a record, a meaningful collection of one or more related characters treated as a unit. In bibliographic records, these are variable length portion containing a particular category of data.
- Directory** : A table of entries, each of which gives the tag, length, and location within the record, segment identifier and occurrence identifier of one data field.
- Discharging** : The act of cancelling the records of documents on loan after their return.
- Indicator digit** : The first two characters of each data field, supplying further information about the contents of the field.
- Intranet** : The network that uses Internet technologies (TCP/IP and others) for local connectivity and is available only to the members of the network.
- ISDS** : An acronym for International Serials Data System. An international network of operational centers (established in 1973 within the framework of UNISIST programme), which are jointly responsible for the creation and maintenance of computer-based databank, and facilitates retrieval of scientific and technical information in serials.
- ISO-2709** : An international standard for bibliographic information interchange on magnetic tape, developed in 1981. Most of the content designator schemes constitute a specific implementation of this standard.
- ISSN** : Acronym for International Standard Serial Number – an internationally accepted code for the identification of serials publications. It consists of seven Arabic digits with an eighth that serves to verify the number in computer processing.

- Mandatory field** : A data field, which should appear in the record when the relevant information appears on the item.
- MARC 21** : MARC 21 is a family of five coordinated formats namely MARC 21 format for authority data, bibliographic data, classification data, community information and holdings data. MARC 21 is a development over USMARC, and has become the de facto bibliographic standard in the area of computerised cataloguing.
- Merging of Title** : It refers to combine two or more journals into a single journal under one title.
- Record** : A collection of information, in one or more fields, about an entity.
- Repeatable field** : A data field, which may appear more than once in the same segment.
- Repeatable sub-field** : A subfield, which may appear more than once in a single occurrence of the data field to which it belongs.
- Reservation** : A request for a specific book or other circulating items to be reserved for a member as soon as it becomes available on completion of processing, or on its return from the binder or another member.
- Routing** : The systematic circulation of periodicals or other printed material among the staff or members of a library in accordance with their interests in order to keep them informed of new developments.
- SDI** : Abbreviation for Selective Dissemination of Information Systems. It is an automated system of information retrieval utilising a computer for disseminating relevant information to users. An interest profile depicting and defining each area of interest is compiled for each user; it consists of terms, which are likely to appear in relevant documents.
- Splitting of Title** : The breaking of a single journal into two or more different journal titles.
- Standing Order** : An order to supply each succeeding issue of a serial publication or subsequent volumes of a work published in a number of volumes issued intermittently.
- Sub-field** : A separately identified part of a data field containing a data element.
- Sub-field identifier** : Two characters immediately preceding and identifying a subfield. First character is called subfield flag and the second character is termed as subfield code.
- System Analysis** : A powerful technique for the analysis of an organisation and its work.
- Tag** : A three characters code appearing in the directory, associated with a data field and used to identify it.

- Union Catalogue** : A catalogue of the various departments of a library, or a number of libraries, indicating their locations. Union catalogue of serials includes the complete holding of serials available in member libraries.
- Withdrawal** : The process of cancelling records in respect of documents that have been withdrawn from the stock of a library.

2.11 REFERENCES AND FURTHER READING

- Cohn, John, M., Kelsey, Ann L and Fiels, Keith Michael. *Planning for automation: a how to-do-it manual for librarians*. New York: Neal-Schuman, 1992. Print
- David, L. T. *Introduction to integrated library systems*. Bangkok: Information and Informatics Unit, UNESCO Bangkok, Thailand, 2001. Print
- Dempsey, L. Distributed library and information systems: the significance of Z39.50. *Managing Information* 1.6, (1994), pp. 41-42.
- Haravu, L. J. *Library automation: design, principles and practices*. New Delhi: Allied Publishers Private Limited, 2004. Print
- Hodgson, Cynthia. *The RFP writer's guide to standards for library systems*. Bethesda, Maryland: National Information Standards Organisation, 2002. <<http://www.niso.org>>
- Morgan, E. L. *Open Source Software in Libraries* (2002). <<http://dewey.library.nd.edu/morgan/musings/ossnlibraries.php>>
- Mukhopadhyay, P. The progress of Library Management Software: an Indian scenario. *Vidyasagar University Journal of Library Science*. 6(2001), pp.51-69.
- Mukhopadhyay, P. Library housekeeping operations – BLII- 001, Block 1, Unit 11 of CICTAL course, IGNOU (2005).
- Mukhopadhyay, P. Library automation – housekeeping operations (pp.85-117), Unit 5, MLII-104 (ICT Applications – Part I), IGNOU, (2006).
- Mukhopadhyay, P. *Library automation through Koha*. Kolkata: Prova Prakashani, 2008. Print
- Rayward, W.B. A History of Computer Applications in Libraries: Prolegomena. *IEEE Annals of the History of Computing*, April-June (2002), pp. 4-15.
- Reynold, D. *Library automation: issues and applications*. London: Bowker, 1985. Print
- Rowley, J. *The electronic library*. London: Library Association Publishing, 1998. Print
- Swan, James. *Automating small libraries*. Ft. Atkinson, Wis.: Highsmith Press, 1996. Print
- Wilson, K. Introducing the next generation of library management systems. *Serials Review*, 38.2 (2012).pp. 110-123.

UNIT 3 LIBRARY AUTOMATION – SOFTWARE PACKAGES

Structure

- 3.0 Objectives
- 3.1 Introduction
- 3.2 History, Evolution and Generations
 - 3.2.1 Historical Foundation
 - 3.2.2 Evolution
 - 3.2.3 Generation of Packages
- 3.3 Categorisation of ILS
 - 3.3.1 Categorisation by Distribution Policy
 - 3.3.2 Categorisation by Place of Origin
- 3.4 Open Source Software Packages
 - 3.4.1 Evergreen
 - 3.4.2 Koha
 - 3.4.3 NewGenLib
 - 3.4.4 PMB
- 3.5 Commercial Software Packages
 - 3.5.1 LibSys
 - 3.5.2 SLIM
 - 3.5.3 SOUL
 - 3.5.4 Virtua ILS
- 3.6 Freeware ILS
 - 3.6.1 ABCD
 - 3.6.2 E-Granthalaya
 - 3.6.3 WEBLIS
- 3.7 Evaluation of Software Packages
 - 3.7.1 Generic Parameters of Evaluation
 - 3.7.2 Specific Parameters of Evaluation for Commercial ILSs
 - 3.7.3 Specific Parameters of Evaluation for Freeware and Open Source ILSs
- 3.8 Global Recommendations
- 3.9 Summary
- 3.10 Answers to Self Check Exercises
- 3.11 Keywords
- 3.12 References and Further Reading

3.0 OBJECTIVES

After going through this Unit, you will be able to:

- understand historical background, evolution and generation of library automation software packages;
- categorise library automation software as per origin and distribution policies;

- identify features and specialties of major commercial and open source software packages in the domain of library automation; and
- know the processes for evaluating library automation packages and understand the trends in developing library automation software packages.

3.1 INTRODUCTION

In this Unit we are going to study the library automation packages. We have already covered different aspects of library automation in Unit 1 and processes and workflows of library systems in Unit 2. This Unit aims to introduce you to the applications of library automation software for different workflows in a library system and its roles in providing information services to users and MIS services to library staff. Mukhopadhyay (2006) outlined the role of typical library automation software for two major subsystems of a library – operational subsystem and administrative subsystem (see Fig. 3.1).

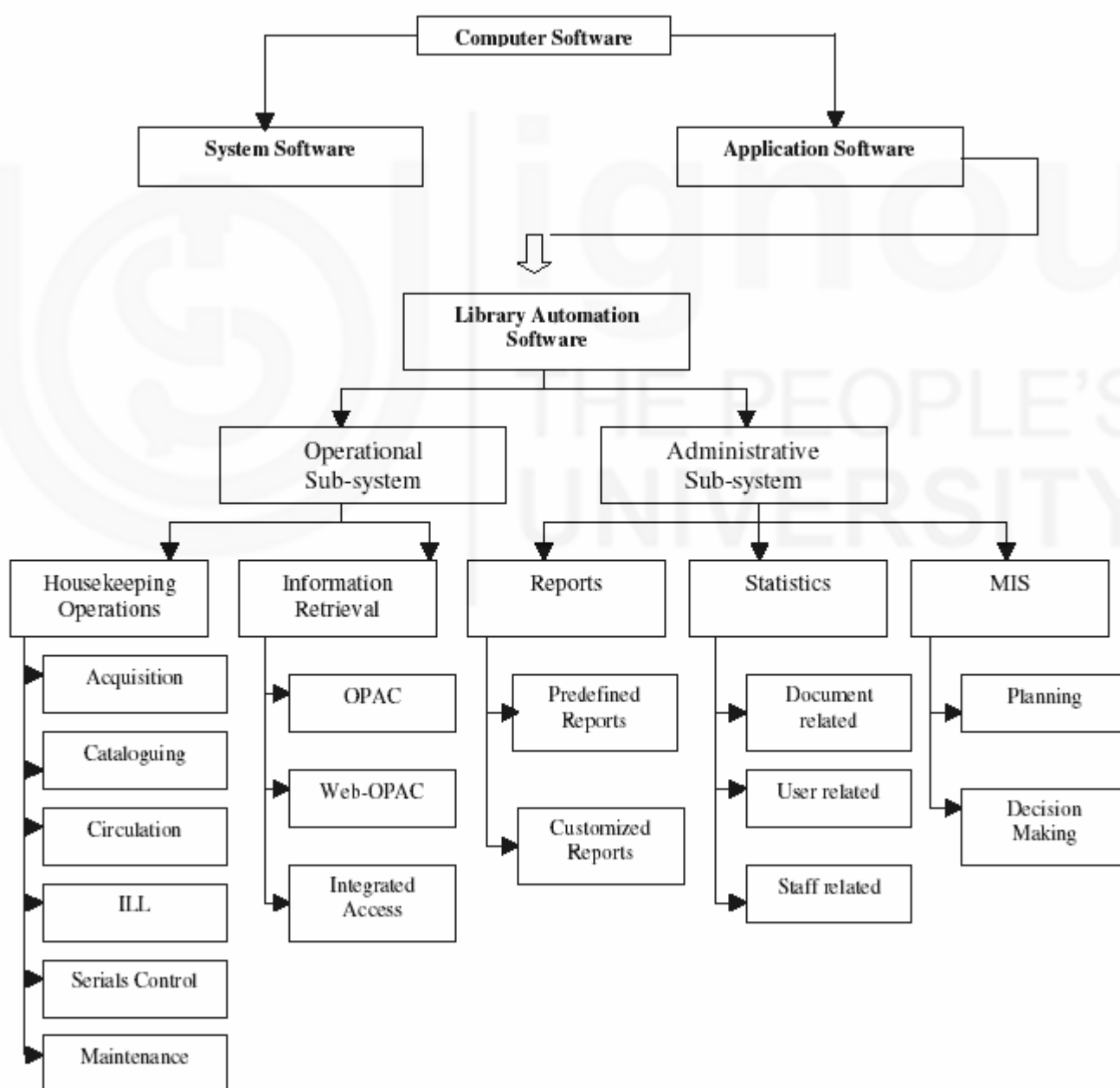


Fig. 3.1: Role of library automation software in integrated setup

Source: Mukhopadhyay, 2006

The above-mentioned roles of an ILS are supplemented by many other value-added features like online acquisition, FRBRised cataloguing, RFID-enabled circulation, member card printing, bar-coding of accession number and member ID, predictive mode of serials control, interactive OPAC, federated searching, extensive reports and statistics in different formats for supporting decision making process etc. Obviously, these enhanced features added into basic core modules over the time, with the improvements in technologies particularly relational data model, web architecture, multilingual technologies, linked open data and with the development of global open standards in the domain of library automation. Presently library automation software are maturing rapidly with the advent of the above technologies.

3.2 HISTORY, EVOLUTION AND GENERATIONS

We already covered the progress of library automation for the last fifty years in Unit 1. This section is trying to associate the development of library automation software with the fundamental improvements in library automation itself.

3.2.1 Historical Foundation

Library automation began in 1930's with the use of punched card equipments in circulation and acquisitions processes in developed countries like US. But you already know from unit 1 that the computer systems applied in automating libraries in late 1960s with the use of low-cost PCs as hardware support and with the development of in-house software for managing processes related to acquisition, cataloguing and circulation. It may safely be said that right from the beginning of library automation, software played the most important role. However, software by definition is the representation of human knowledge in the forms of bits and bytes. In this sense software may be viewed as digital version of human knowledge not just as a set of related programs. Similarly, library automation software are based on knowledge and experiences acquired by library professionals over centuries. These software tools are helping in easy and effective management of housekeeping operations. Such software is also supporting dissemination of information services and helping library staff in administrative activities. Presently almost all library automation software are integrated systems, based on relational database architecture. In such systems files are interlinked so that deletion, additions and other changes in one file automatically activate appropriate changes in related files. The use of library automation software is rapidly increasing in India right from 1995. Almost all special libraries and large academic libraries in India adopted integrated library system. Recently public libraries and college libraries all over the country are either adopting automation software or planning actively to go for library automation with the advent of globally competitive open source ILSs (available free of cost and can be customised extensively). There are also supports from governments in adopting open source ILS, for example, National Library Mission (Ministry of Culture, Govt. of India) advocated to adopt Koha (an open source globally reputed ILS) for automating public libraries, Kerala State Government declared Koha as the official ILS for the public libraries in the state and almost 250 public libraries have already been automated by using Koha in West Bengal. A network of public libraries in Konkan area is automated through Koha (see granthalaya.org). Ministry of HRD, Government of India through it N-LARN project under NMEICT (see n-larn.ac.in) is helping college libraries under UGC and AICTE in adopting Koha for library

automation. Overall, libraries in India are moving towards a large-scale implementation of library automation in different parts of the country.

3.2.2 Evolution

You already know after covering the Unit 1 that the library automation process underwent five eras on the basis of technological improvements in computer programming, database management system, network capabilities and web integration. To respond these changes, library automation software also improved considerably through five different generations. Mukhopadhyay in 2006 reported a comparative account of four generations of ILSs. Use of cloud computing, web-scale management, linked open data and web 2.0 technologies initiated the fifth generation of ILSs. This section points out major technological features of five different generations of ILSs and next section (3.2.3) gives a comparative account of five generations of library automation software against the features earmarked by Mukhopadhyay (2006).

- The first generations ILS packages were piecemeal, non-integrated and non-portable across hardware architectures and software platforms. These packages were module-based systems with no or very little integration between modules. Circulation module and cataloguing module were the priority issues for these systems and were developed to run on specific hardware platform and proprietary operating systems;
- The most important achievements in second generation of packages were hardware and platform independence. The second generations ILSs become portable between various platforms with the introduction of UNIX and DOS based systems. The ILSs of this generation offer links between systems for specific functions and are command driven or menu driven systems;
- The most important features in third generation of packages were GUI, seamless integration of modules and relational model based client-server architecture. The third generations ILS packages are fully integrated systems based upon relational database structures and client-server architecture. They embodied a range of standards, which were a significant step towards open system interconnection. Colour and GUI features, such as windows, icons, menus and direct manipulation have become standards and norms in this generation;
- Web architecture, Unicode and digital media archiving were the major attributes of the fourth generation ILSs. The fourth generations ILSs were based on web-centric architecture and facilitate access to other servers over the Internet. These systems were Unicode complaint and allow accessing multiple sources from one multimedia graphical user interface; and
- The present of the fifth generation ILSs are adopting rapidly cutting edge technologies like web-scale management, cloud computing, web 2.0 features on the basis of AJAX (Asynchronous Java and XML) technology, Application Program Interface (API), and linked open data. Rising of open source ILSs and implementation of open standards are also remarkable features of this generation.

The progress of ILSs through five different generations improved functionalities, enhanced user access to library resources in 24X7 mode, facilitated new generation

3.2.3 Generation of Packages

Library automation software are categorised into four different generations on the basis of core attributes of the packages like software architecture, programming language, internal DBMS, module integration capabilities etc. (Mukhopadhyay, 2006). This categorisation adopted by many researchers in the domain of library automation (see <http://shodhganga.inflibnet.ac.in/jspui/handle/10603/9406>). Table 3.1 provides a comparative study of five different generations of ILSs in the same line with bit modifications in parameters.

Table 3.1: Five generations of ILSs

Sl. No.	Features	1 st Generation	2 nd Generation	3 rd Generation	4 th Generation	5 th Generation
1	Programming Language	Low level language	COBOL, PASCAL, C	4 GL	OOPS	AJAX
2	Operating System	In house	Vendor Specific	UNIX, MSDOS	UNIX, Windows and Linux	Mainly Linux distributions
3	Data model	Non-standard	Hierarchical and Network model	Entity-Relation model	Object oriented model	Support for FRBR, FRAD and FRSD
4	Import/Export	None	Limited	Standard	Fully integrated and seamless	Distributed across formats through XML
5	Communication	Limited	Some interface	Standard	Full connectivity across Internet	Support for Linked Open Data
6	Standards support	Limited and proprietary	Improved for bibliographic data	Bibliographic and authority data	Standards for all modules	Emphasis on open interoperability standards
7	Portability	Machine dependent and hardware specific	Machine independent but Platform dependent	Multi-vendor	Multi-vendor and Platform independent	Complete portability
8	Reports and statistics	Fixed format, limited fields and statistics	Fixed format, unlimited fields and moderate statistics	Customised report generation and wide statistical range	Customised report generation with e mail interface and statistics in different formats	Complete control over report elements and comprehensive statistics generation
9	Media	None	None	Available in limited way	Fully available with Multimedia	All formats for digital objects

Library Automation

10	Capacity of record holding	Limited	Improved	Unlimited	Unlimited	Unlimited
11	Module Integration	None	Bridges	Seamless	Seamless and object oriented	Seamless with API for new modules
12	Architecture	Stand-alone	Shared	Client-Server	Web-centric/ Distributed	Cloud and Web-scale
13	Interface	Command driven (CUI)	Menu driven (CUI)	Icon driven (GUI)	Icon driven with Web and Multimedia (GUI)	Web 2.0-enabled interfaces
14	User Support	Single user	Limited number of users	Unlimited number of users	Unlimited number of users	Unlimited concurrent users
15	Multi-lingual support/ UNICODE	None	Limited (through Hardware support)	Standard	UNICODE based	UNICODE with embedded virtual keyboard for languages
16	External resource integration	None	None	Limited	Improved	Full integration with external datasets
17	Discovery and Federated searching	None	None	None	Limited	Support for federated search
18	Distribution mode	Close and in-house	Close and proprietary	Close and proprietary	Both close and open source	Mainly open source

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

1) Mention typical role of an ILS in library automation.

.....

.....

.....

.....

.....

2) Make a comparison between 3rd and 4th generation ILSs.

.....

.....

.....

.....

- 3) Enumerate features of 5th generation ILS.

.....

.....

.....

.....

.....

3.3 CATEGORISATION OF ILS

CDS/ISIS, a textual database management software developed by UNESCO in 1985, played an important role of forerunner for library automation in India. This package is not an ILS but provides an excellent framework for managing bibliographic databases such as library catalogue. It is specifically meant for the structured non-numerical databases, powered by a very comprehensive formatting language to control display of records and also provides many advanced level retrieval features. In India, erstwhile NISSAT (national distribution agency for CDS/ISIS) with the help of other professional bodies organised a number of training courses on application of CDS/ISIS (DOS and Windows version) in information organisation activities. As a result, a large pool of trained manpower developed all over the country. Some organisations from the experience of use of CDS/ISIS, MINISIS etc. developed their own ILSs e.g. DESIDOC developed DLMS (Deference Library Management System), INSDOC came with CATMAN (Catalogue Management) and SANJAY was developed by DESIDOC under NISSAT project by augmenting CDS/ISIS (Version 2.3) for library management activities. So we may say that first era of ILS in India dominated by ILSs developed in house such as DLMS, CATMAN and SANJAY. This trend is followed by commercial software firms in developing comprehensive full-featured ILSs in India. The era of commercial ILS is dominated by ILSs of foreign origin (such as Virtua ILS), ILSs developed in India by using foreign ILS (such as BASISPlus and TECHLIBPlus) and ILSs of purely India origin (such as LibSys, E-Granthalaya). However, the scenario of library automation in India has changed from 2001 onwards with the availability of open source ILSs which are available freely, customisable and based on global open standards in the domain of library automation. In this section we are for categorising ILSs available in India on the basis of two different train of characteristics – distribution policy (close source and open source) and place of origin (foreign origin, Indian origin and hybrid).

3.3.1 Categorisation by Distribution Policy

You know that software of any kind can be grouped into two fundamental categories – system software and application software. This grouping is based on the application levels of software. System software (such as operating system) is related with the management of resources in a computer system whereas an application software are designed to perform certain tasks such as database management (DBMS software), word processing (Word processing software), image processing (Graphic software) etc. Library automation software is an application software and manages library automation activities. On the other hand, as per the distribution policy (conditions for availability of software), software may be grouped into two broad divisions – close source software and

open source software (OSS). Close source ILSs are available against license fees (one time capital expenditure and recurring annual maintenance fees) or freely (a few close source ILS are available freely e.g. e-Granthalaya) without source codes. It means users cannot customise or modify the source code of ILS. Close source software therefore, may again be placed in two groups – commercial software and freeware. Open source software, on the other other hand, available freely with full freedom to customise the source code as per the requirements of the library. So, as per the distribution policy, the whole array of ILS may be categorised into three groups – Close source commercial ILS, Close source freely available ILS, and Open source ILS (see Table 3.2 with illustrative examples).

Table 3.2: Categorisation of ILSs by distribution policy

<i>Types of Library</i>			
Distribution policy	Large Library Systems	Medium Range Library Systems	Small Library System
Close source ILSs (Commercial)	<ul style="list-style-type: none"> • VIRTUA ILS • LibSys 	<ul style="list-style-type: none"> • SLIM 21 • SOUL 	<ul style="list-style-type: none"> • AUTOLIB • NIRMALS
Close source ILSs (Freeware)	<ul style="list-style-type: none"> • ABCD 	<ul style="list-style-type: none"> • e-Granthalaya • WEBLIS 	<ul style="list-style-type: none"> • LAMP • Librarian
Open source ILSs (Freely available)	<ul style="list-style-type: none"> • Evergreen ILS • Koha (version 3.x) 	<ul style="list-style-type: none"> • Koha (version 2.x) • NewGenLib 	<ul style="list-style-type: none"> ○ Emilda ○ PHPMyLibrary

Please remember the examples are only illustrative not comprehensive. There are several ILSs in use in Indian libraries both from commercial and open source domains. In the close source group the LibSys and SOUL are dominating ILSs, and in the open source group Koha and NewGenLib are the most popular ILSs. Some libraries in India are using WEBLIS which is based on CDS/ISIS. It has already been mentioned that the availability of open source ILSs helped in large-scale library automation in India as far as school libraries, college libraries and public libraries are concerned. Till date around fifteen open source ILSs are available for use. However, we may go for categorising open source ILSs as per the maturity level in terms of architecture, data model, core modules, support for standards, multilingual data processing ability, user services and interoperability. The KualI ILS is an experimental open source library automation software as it is trying to implement the OLE and ILS-DI recommendations for developing the next generation automated library system.

Table 3.3: Categorisation of open source ILSs by maturity level

<i>Categorisation of Open source ILS by Maturity Level</i>			
Fairly matured	Moderately matured	Infancy	Experimental
<ul style="list-style-type: none"> • Emilda • Evergreen • Koha (version 3.x onwards) • NewGenLib 	<ul style="list-style-type: none"> • MiniSOPULI • OPALS • OpenBiblio • OtomiGenX • phpMyLibrary 	<ul style="list-style-type: none"> • Avanti • e-library • PHPMyBibli • PMB • PYTHEAS 	<ul style="list-style-type: none"> • KualI ILS

3.3.2 Categorisation by Place of Origin

Mukhopadhyay (2001, 2005) grouped ILSs available in India on the basis of place of origin. This grouping later on was adopted by many researchers in the field. It includes three fundamental categories – ILSs of foreign origin, ILSs developed over ILSs (or textual database management systems) of foreign origin and ILSs of Indian origin. This grouping may again be sharpened by dividing the packages on the basis of size of library systems i.e. large library system, medium range library system and small range library system.

Table 3.4: Categorisation of ILSs by place of origin

<i>Application Domain</i>			
Origin	Large System System	Medium Range	Small System
ILSs of foreign origin	<ul style="list-style-type: none"> • Alice for WINDOWS • Evergreen • Koha (ver 3.x) • Virtua ILS 	<ul style="list-style-type: none"> • Koha (ver 2.x) • Emilda 	<ul style="list-style-type: none"> • phpMyLibrary • OpenBiblio • PMB
ILSs developed over ILS of foreign origin	<ul style="list-style-type: none"> • NG-TLMS.NET (over TLMS package) 	<ul style="list-style-type: none"> • WINSANJAY • ABCD (Over CDS/ISIS) 	<ul style="list-style-type: none"> • LAMP • WEBLIS (Over CDS/ISIS)
ILSs of Indian origin	<ul style="list-style-type: none"> • LIBSUITE • LIBSYS • MECSYS • NEWGENLIB • NEXLIB • SLIM 21 • SOUL • SUCHIKA • TULIPS • ULYSIS • WILISYS 	<ul style="list-style-type: none"> • AUTOLIB • DLMS • GRANTHALAYA • LIBRA • LIBRARIAN • LISTPLUS • NETLIB • NIRMALS • SLIM ++ 	<ul style="list-style-type: none"> • ARCHIVES • CATMAN • E-GRANTHALAYA • GOLDEN LIBRA • LIBMAN • Library- Manager • LIBRIS • LIBSOFT • LOAN-SOFT • SALIM

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

4) What is an open source ILS? List some major open source ILSs.

.....

.....

.....

.....

.....

- 5) Categorise ILSs available in India with example.

.....

.....

.....

.....

.....

3.4 OPEN SOURCE SOFTWARE PACKAGES

Free/Libre Open Source (FLOSS) or simply Open source ILSs are maturing day-by-day and increasingly considered as viable alternatives to commercially available ILSs. Some of the open source ILSs are taking technological lead in cutting edge technologies, such as Koha is considered as leader in developing the model OPAC 2.0 (through integration of Web 2.0 tools like RSS, virtual shelf browsing, user-driven tagging, provision of book reviews by users, information mashp with Amason, Syndicate, LibraryThing, Open Library etc.) and in developing Z39.50 server facility for distributed cataloguing (most of the commercial ILSs only include Z39.50 client). Apart from these technological advantages, open source ILSs provide many other benefits such as –

- **Community ownership:** Users are considered as co-developers and there is no single owner of the ILS, rather user libraries are considered as stakeholders of the product;
- **Vendor independence:** Open source ILSs are free from vendor-lock in. It means libraries are free to hire expertise at the time of requirements;
- **Smooth migration:** If user library decides to switch over from one open source ILS to another ILS (commercial or open) the data migration is quite smooth and loose-less. But the migration from commercial ILS to open source ILS is not always an easy task due to problematic data transmission for obvious commercial reasons;
- **Use of open standards:** Open source ILSs use open standards for most of the work-flows and activities and thereby ensure transparent library operations;
- **Customisation:** No two libraries under the Sun run in the same way. Commercial ILSs provide a fit-to-all-size solution for libraries of any type or size. And these software cannot be customised as source codes are not available. Open source ILSs allow libraries to customise the source code to meet the requirements of individual libraries;
- **Fund savings:** As open source ILSs are available at no cost or at nominal cost, the library budget for software procurement and annual maintenance of the ILS may be utilised in other areas of library development;
- **Freedom:** Open source ILS allows librarians to operate at the system level whereas in commercial ILSs the role of librarians reduced to mere data entry operators. Apart from this benefit, open source ILSs provide freedom to use, modify and distribute the software on the basis of GPL (GNU General Public License); and

- **Fraternity:** Open source ILS supports fraternity in library community at the international level through cooperation, sharing of expertise and experiences.

A detail account of philosophies and principles of open source software is available in the next Unit i.e. Unit 4 in this block. However, in this section we are going to study the features of some matured open source ILSs that are globally reputed for their features, architecture and respectable user base (number of active users of the ILS). Presently fourteen ILSs are available against licensing agreements and these are Emilda, Evergreen, Gnuteca, InfoCid, Jayuya, Koha, NewGenLib, oBiblio, OPALS, OpenAmapthèque, OpenBiblio, PhpMyLibrary, PMB and Senayan. Müller (2011) in his study categorised the open source into two levels – i) Maturity of ILS Community; and ii) Maturity of ILS Functionality. Each of these two categories have divisions. For example, Müller divided the category ILS community into four divisions namely Inactive community, Just released community, Emerging community and Sustainable community against weight based decision matrices. The result is given below:

Category	FOSS ILS name
Sustainable	Evergreen, Koha
Emerging	PMB
Just Released	Gnuteca, InfoCID, NewGenLib, oBiblio, OPALS, Open Amaptheque, Senayan
Inactive	Emilda, EspaBiblio, Jayuya, OpenBiblio, PhpMyLibrary

Source: Müller, T. (2011). How to choose a free and open source integrated library system. *OCLC Systems & Services*, 27(1), 57-78.)

Similarly, rating by maturity of functionalities of open source ILSs in the above research study shows the following result:

Categories	FOSS ILS name
Mature	Koha
Improving	Evergreen, PMB

Source: Müller, T. (2011). How to choose a free and open source integrated library system. *OCLC Systems & Services*, 27(1), 57-78.)

The research study of Müller (2011) identified three matured open source ILS namely Evergreen, Koha and PMB. We are going to study these three open source ILSs along with NewGenLib as a special case as it is originated from India.

3.4.1 Evergreen

Evergreen (<http://evergreen-ils.org/>) is originated from public library domain in 2006 like Koha (released in 2000 as open source ILS). The Evergreen Project was started in 2006 by the Georgia Public Library System to support 275 public libraries in the state of Georgia, US. This Client-Server open source ILS is based on a robust, scalable, message-passing framework – OpenSRF, available under GNU GPL, version 2, and currently used by over 1000 libraries around the world.

It has modules for circulation (with sophisticated fiscal management), cataloging (with comprehensive MARC 21 based catalogue editor), Web catalog, and statistical reporting, acquisition and serials control. It also supports the SIP2 protocol for self-check. The current release is version 2.6 (released in April 2014) and the next release (version 2.7) is due in September 2014. It has comprehensive documentation (<http://docs.evergreen-ils.org/>), wiki (<http://evergreen-ils.org/dokuwiki/doku.php>), and feature request facility.

System requirements

Evergreen is based on client-server architecture. It means that at server level we need to install server version of Evergreen and in client machines client version of Evergreen need to be installed and configured. The minimum hardware requirements of server and client machines are as follows:

Server level

- A high-end desktop or entry-level server.
- 1GB RAM, or more (if server runs a graphical desktop).
- Architecture to run Unix-like Operating System (any flavour of Linux).
- Ports 80 and 443 should be opened in for TCP/IP connections to allow OPAC and staff client connections to the Evergreen server.
- Network to establish server-client connections.

Client machines

- Low-end desktop with Windows (XP, Vista, or 7/8), Mac OS X, or Linux operating system.
- A reliable high speed Internet connection.
- 512MB of RAM.
- TCP protocol to connect Evergreen server at ports 80 and 443.
- Barcode scanner and printer (optional).

Companion software

Apart from Evergreen server and client software, the server machine requires following companion software to run server version of Evergreen:

- 4) Unix-like Operating System.
- 5) PostgreSQL as RDBMS (version 9 or later).
- 6) Apache as Web server (version 2.x).
- 7) OpenSRF (version 2.3.0 or later).
- 8) libdbi-libdbd libraries.

Major Features

The general features of evergreen ensure stability (even under extreme server load), capability (robust handling of high volume of transactions and concurrent users), flexibility (to accommodate the varied needs of libraries), security (to protect our patrons' privacy and data) and interactivity (to facilitate patron and staff in using the system). Apart from these features, it supports all sorts of core activities like:

- System administration (privilege control, user and group management, cataloguing editor control, log records management, system parameters settings, report generation, granular access control, search enhancing, Z39.50 server and client settings, module administration, SMS gateway management, federated search control, EDI based acquisition control, theme and skin control for fine tuning user interface, data migration, backup and restoration etc.);
- Acquisitions (acquisitions settings, cancel/suspend reasons, claiming, currency types, distribution formulas, EDI (electronic data interchange), exchange rates, fund tags, funding sources, funds management, invoice menus, line item features [alerts appear in a pop-up box when the line item, or any of its copies, are marked as received], providers [vendor/supplier based profile that includes contact information for the provider, holdings information, invoices, and other information.]);
- Cataloguing (comprehensive MARC editor, authority data control, model data entry worksheet, authority lists support, multilingual data entry, integration of external resources, authority control through MARC 21 authority format, thesaurus integration (eleven number of thesauri are available and cataloguer can create new thesauri), creation of browsing categories, record display control, link checker (helps to verify the validity of URLs stored in MARC records), cross-linking of items (facility to link items to multiple bibliographic records), distributed cataloguing through Z39.50 client, bibliographic data export/import, bibliographic search enhancements – supports for advanced search operators);

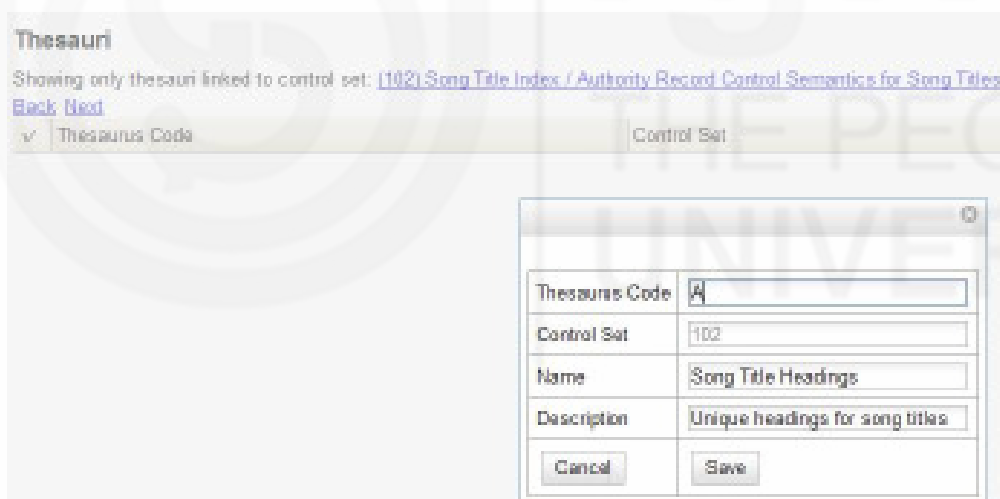


Fig. 3.2: Thesaurus creation in Evergreen

- Circulation (Member management, member data migration, RFID integration, in-built support for bar-coded circulation, smooth issue/return, self-checkout facility through SIP2, circulation parameters settings, a separate facility for holds/reservation management, auto calculation of fines and overdue, SMS alert for overdue materials, facility to manage long overdue, member card generation, off-line circulation etc.);
- Serials control (MARC Format for Holdings Display (MFHD) display in the OPAC, two views of serials control – small number of issues and large number of issues (both views help to create subscriptions, add distributions, define captions, predict future issues, and receive items), loose issue management, holdings management through MFHD, special issues management, template toolkit for OPAC views for serials etc.);

- Report generation (separate report daemon, comprehensive report generation, facility to run recurring reports, reports organisation in folders, facility to select fields for report generation, sorting and filtering facilities, interface to generate report from back-end RDBMS (PostgreSQL), creation of report templates, exporting reports in different formats, report dump feature etc.); and
- OPAC (searching and browsing, availability of sophisticated search operators, separate OPAC for kids, user-driven skin control for OPAC, search results in many formats, including HTML, MARCXML, MODS and binary MARC21 format, facility to store favourite books in “My List”, third party content support (such as reader reviews) in Kids OPAC, user-driven holds/reservation etc.).

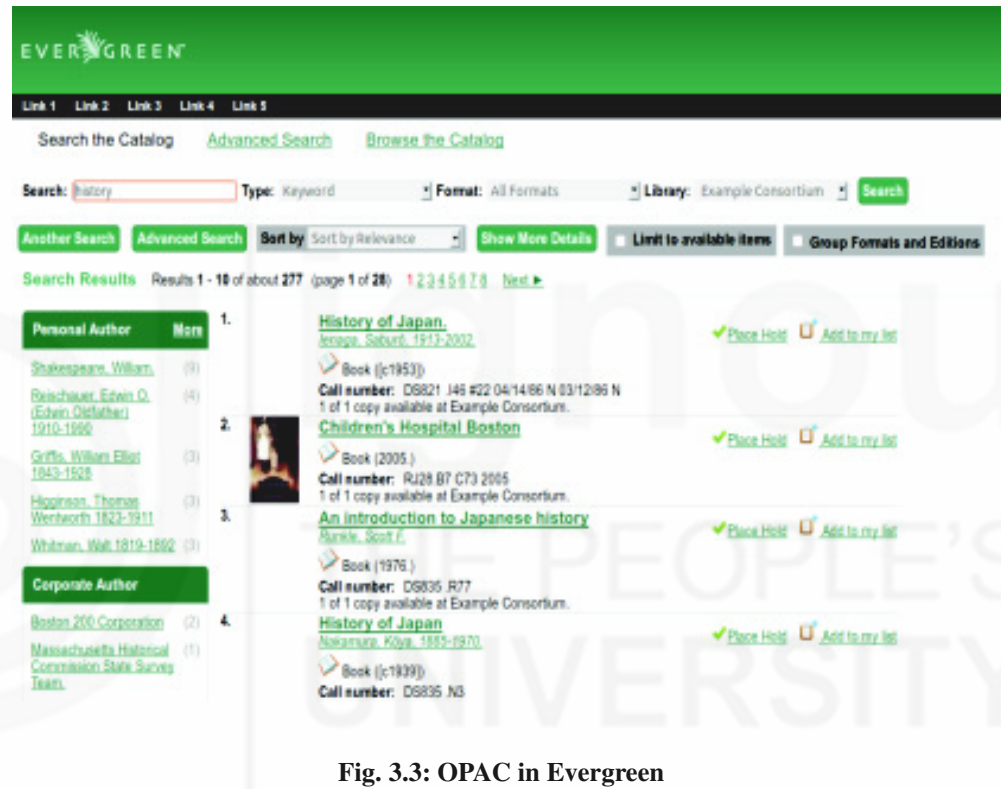


Fig. 3.3: OPAC in Evergreen

Special features

The Evergreen open source ILS originated as ILS for library consortia and has the credit of many special or unique features such as:

- Use of Open SRF (a message routing network that offers scalability and failover support for individual services and entire servers with minimal development and deployment overhead);
- TPAC support to associate a web page with a library (useful to link library information page, library rules, journal portals etc.);
- Auto-suggest option during OPAC searching (the facility may be enabled/disabled by users);
- OPAC is Web Content Accessibility Guidelines (WCAG) 2.0 compatible to support access by physically challenged users;
- Meta-record search facility to access group formats and editions and for listing multiple constituent records;

- Support for MARC format for holdings display and its integration with OPAC for journal holdings;
- EDI support for acquisition of library materials and SIP2 support for self checkout; and
- Support for template creation by administrator and skin selection by users.

Important URLs

- Downloading (<http://evergreen-ils.org/egdownloads/>);
- Documentation (<http://evergreen-ils.org/eg-documentation/>);
- Users list (http://evergreen-ils.org/dokuwiki/doku.php?id=evergreen_libraries);
- Wiki (<http://wiki.evergreen-ils.org/doku.php>);
- Mailing list (<http://evergreen-ils.org/communicate/mailling-lists/>);
- Blog (<http://evergreen-ils.org/communicate/blog/>);
- IRC (<http://evergreen-ils.org/communicate/irc/>); and
- Book (http://en.flossmanuals.net/_booki/evergreen-in-action/evergreen-in-action.pdf).

Remark

Evergreen open source ILS has improved a lot in recent years and presently considered as the model ILS for managing library consortia and library networks. However, the above mentioned features of Evergreen suggest that the ILS can be deployed in any type or size of individual library to support core automation workflow as well as many value-added features.

3.4.2 Koha

As you know already, there are now almost fourteen open source ILS in the domain of library automation. But Koha is the first open source ILS (released in 2000 as open source) and possibly it is now the most feature rich open source ILS. Koha changed the rule of game in the ILS market and set trends in many ongoing changes in the area of library automation. Koha was originated in public library system of New Zealand. In Maori language Koha means an *unconditional gift*. The first version (1.0) of Koha made available for downloading as open source software in July 2000. The current stable version is 3.14.06 (released in April 30, 2014). The Koha ILS community is very active and in every month the developer community provides a bugfix release. Koha versions with new features are released in every six months (for example the next stable version 3.16 is expected to be released in June 2014). Koha is an integrated library management system that was originally developed by Katipo Communications Limited of Wellington, New Zealand for the Horowhenua Library Trust (HLT), a regional library system located in Levin near Wellington. In 1999, Katipo proposed developing a new system for HLT using open source tools (PERL, MySQL, and Apache) that would run under Linux and use Telnet to communicate with the branches. The software was in production on 3rd January 2000, and released under the GPL for other people to use in July 2000. Koha 1.01 was released on August 9, 2000. Koha is essentially based on LAMP architecture. Here L is Unix-like OS (different flavours of Linux); A is Apache Web server; M is MySQL

RDBMS and P is PERL programming environment. Koha is pioneer in a number of technological achievements such as use of Web 2.0 tools, integration of authority format and bibliographic data format, availability of OPAC interface in 25 different languages, implementation of Z39.50 server and OAI/PMH compatibility, in built support for social networking tools, independent branch management, Web-based self issue, use of open standards for different modules and granular system administration facilities.

System requirements

Koha is based on Web architecture. Both staff interface for professional activities and public access interface for retrieval are available through Web browser. This Web-enabled open source ILS supports 24×7 mode of access for both for staff and users. Another important advantage of the Web architecture is no requirement of installation of client software in the end-user terminals. A web browser (like Firefox, Chrome etc.) may act as client software at end user terminal. This feature of Koha reduces maintenance works to a great extent in a large campus library (for example we need to install, configure and maintenance Koha only at the server; at client level no Koha specific maintenance is required as client machines access Koha through a preloaded Web browser). In short, at server level we need to install Koha and client machines can access Koha server through Web browser (most of desktops and laptops are preloaded with web browser). The minimum hardware requirements of server and client machines are as follows:

Server level

- A high-end desktop or entry-level server
- 1GB RAM, or more (if server runs a graphical desktop)
- Architecture to run Unix-like Operating System (any flavour of Linux but Debian and its derivatives like Ubuntu are mostly in use)
- Ports 80 and 8080 should be opened in for TCP/IP connections to allow OPAC and staff client connections to the Koha server. These two ports are default ports for OPAC and staff interfaces respectively but the ports can be changed as per the network settings of the library
- Network to establish TCP/IP connections.

Client machines

- Low-end desktop with Windows (XP, Vista, or 7/8), Mac OS X, or Linux operating system
- A reliable high speed Internet connection (optional)
- 512 MB of RAM
- TCP/IP protocol to connect Koha server at ports 80 and 8080 (or other ports as desired)
- Barcode scanner and printer (optional).

Companion software

Apart from Koha, the server machine requires following companion software to run server version of Evergreen:

- 9) Unix-like Operating System (Koha users prefer Debian, Ubuntu and CentOS)

- 10) MySQL as RDBMS (version 5.5 or later)
- 11) Apache as Web server (version 2.x)
- 12) YAS toolkit
- 13) PERL programming environment (version 5.10 or later) and PERL modules (version 3.14 of Koha requires a total of 139 PERL modules).

Major Features

Koha is considered as the first and the best ILS from open source domain. It is a global The Koha developer team explored many emerging possibilities to redefine the scope of ILS such as OAI/PMH server, Z39.50 server, OPAC in 25 languages (the list is growing everyday), options for two text retrieval engines (Sebra and Apache-Solr), and options for two cataloguing interfaces (default cataloguing template and Biblios template). However, the major features are as follows:

- System administration (global parameters settings for each module, basic parameters settings for library, enhanced contents for integrating cataloguing data with global resources through information mashup, comprehensive report generation, granular access control, independent branch management option, log records supervision, fine tuning of privilege control MARC bibliographic framework set, Z39.50 client settings etc.);

Koha administration

Global system preferences

Manage global system preferences like MARC flavor, date format, administrator email, and templates.

Basic parameters

Libraries and groups
Define libraries and groups.

Item types
Define item types used for circulation rules.

Authorized values
Define categories and authorized values for them.

Patrons and circulation

Patron categories
Define patron categories.

Circulation and fines rules
Define circulation and fines rules for combinations of libraries, patron categories, and item types

Patron attribute types
Define extended attributes (identifiers and statistical categories) for patron records

Library transfer limits
Limit the ability to transfer items between libraries based on the library sending, the library receiving, and the item type involved. These rules only go into effect if the preference UseBranchTransferLimits is set to ON.

Transport cost matrix
Define transport costs between branches

Item circulation alerts
Define rules for check-in and checkout notifications for combinations of libraries, patron categories, and item types

Cities and towns
Define cities and towns that your patrons live in.

Road types
Define road types (street, avenue, way, etc.). Road types display as authorized values when adding/editing patrons and can be used in geographic statistics.

[g-rules.pl](#)

Catalog

MARC bibliographic framework
Create and manage Bibliographic frameworks that define the characteristics of your MARC Records (field and subfield definitions) as well as templates for the MARC editor.

Koha to MARC mapping
Define the mapping between the Koha transactional database (SQL) and the MARC Bibliographic records. Note that the mapping can be defined through MARC Bibliographic Framework. This tool is just a shortcut to speed up linkage.

Keywords to MARC mapping
Define the mapping between keywords and MARC fields, those keywords are used to find some datas independently of the framework.

MARC Bibliographic framework test
Checks the MARC structure. If you change your MARC Bibliographic framework it's recommended that you run this tool to test for errors in your definition.

Authority types
Create and manage Authorities frameworks that define the characteristics of your MARC Records (field and subfield definitions).

Classification sources
Define classification sources (i.e., call number schemes) used by your collection. Also define filing rules used for sorting call numbers.

Record matching rules
Manage rules for automatically matching MARC records during record imports.

OAI sets configuration
Manage OAI Sets

Acquisition parameters

Currencies and exchange rates
Define currencies and exchange rates used for acquisitions.

Budgets
Define your budgets

Funds
Define funds within your budgets

Additional parameters

Z39.50 client targets
Define which servers to query for MARC data in the integrated Z39.50 client.

[Did you mean?](#)

Fig. 3.4: Koha administration

- Acquisitions (basic parameters for acquisition, budget head and fund allocation, real time fund accounting, vendor management, different types of order handling, order through Z39.50 searching, exclusive data entry framework in acquisition module, provision for item related information etc.);
- Cataloguing (comprehensive MARC editor, inclusion and integration of MARC 21 bibliographic and authority framework, integration of thesaurus and authority lists, multilingual data entry, sub module for authority data management, Z39.50 client search for both bibliographic and authority data, implementation of FRBR model in providing item related information, integration of catalogue data with global related resources through title-ISBN matching rule, help to manage leader, control (00X) and number and code fields (0XX) in MARC 21 etc.);

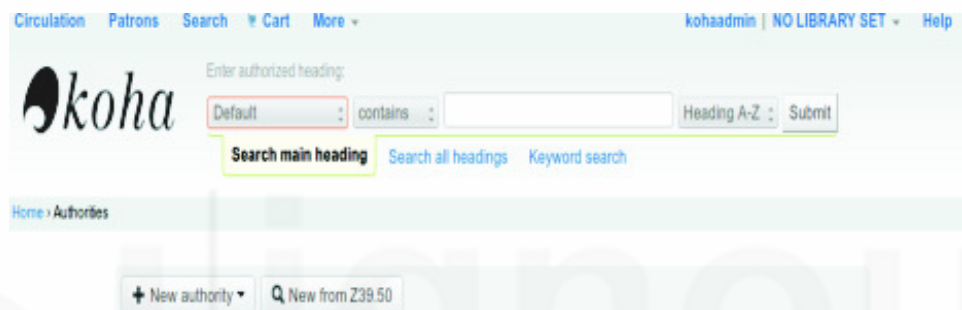


Fig. 3.5: Authority cataloguing in Koha

- Circulation (all required activities support, off-line circulation, granular circulation rules, fine calculation through cron job, RFID integration facility, member photo management, fast cataloguing in circulation module, renew, holds management, user-driven reservation etc.);
- Serials control (predictive mode of serials control, easy management of Kardex of loose issues of journals, holdings management, separate display for back volumes and current issues, provision for routing, easy renewals, creation of frequency master and numbering patterns, vendor-wise claim management, links with cataloguing module and budget head under acquisition module etc.);
- Report generation (predefined reports, custom report format, provision for pick-and-choose fields, auto scheduling of reports, sorting and filtering provision, statistical reports, top lists, format exchange provision); and
- OPAC (searching and browsing, enhanced content integration through information mashup, simple and advanced search interfaces, OPAC language change option, user login for personal information environment, authority searching, tag cloud, subject cloud, purchase suggestion, filter by language, item types and library, different sorting options – title, author, relevance, dates, popularity, call number, range search and sophisticated search operators, cart for listing favourite documents, private and public lists, filtering by subtype – by audience, by content type, by format, and by content type, by availability, purchase suggestions etc.).



Fig. 3.6: OPAC in Koha

Special features

The Koha open source ILS originated as ILS has many special or unique features. Some of the important special features are:

Enhanced features

- Can be integrated with free bibliographic data services (XISBN, Amazon, ThingISBN)
- Full authority control
- Compliant fully with Unicode 5.1
- Can be used as CMS (Integration of ILS and CMS)
- Easy control of contents/news/running text
- Can easily be integrated with wiki, blogs etc.
- Supports emerging standards like NCIP, MARC-XML, DCMES, METS
- Supports sophisticated search features – Boolean, Relational and Positional operators
- Any report generation.

Standard supports

- SRU/W, Z39.50, UnAPI (<http://unapi.info/>) , COinS/OpenURL
- OpenSearch (<http://opensearch.a9.com/>)
- Records are stored internally in an SGML-like format and can be retrieved in MARCXML, Dublin Core, MODS, RSS, Atom, RDF-DC, SRW-DC, OAI-DC, and EndNote;
- OPAC can be used by citation tools such as Zotero
- Koha 3.x includes support for 3M's Standard Interchange Protocol (SIP2), using the OpenNCIP libraries (<http://openncip.org>)

- Cross-platform, multi-RDBMS architecture
- News writer, label creator, calendar, OPAC comments, MARC staging and overlay, notices, transaction logs, guided reports with a data dictionary and task scheduler, classification sources/filing rules etc.

Web 2.0 features

- Can generate RSS (including ATOM) feed for search query
- Supports information mashup (OPAC can be linked with book jacket service, book rating/review from Amazon, Google books, Syndicate LibraryThing, Open Library etc.)
- Users can submit comments/rating/tags for any item from any device (mobile OPAC)
- Can be integrated easily with many Web 2.0 tools like zoreto, delicious, etc.

Important URLs

- Downloading (<http://koha-community.org/download-koha/>);
- Documentation (<http://koha-community.org/documentation/>);
- Users list (http://wiki.koha-community.org/wiki/Category:Koha_Users)
- Wiki (<http://wiki.koha-community.org/>);
- Mailing list (<http://koha-community.org/support/koha-mailing-lists/>);
- Free support (<http://koha-community.org/support/free-support/>);
- IRC (<http://koha-community.org/get-involved/irc/>); and
- Calendar of events (<http://koha-community.org/calendar/>).

Remark

Koha has already established itself as a global trend setter in the domain of ILS. Many libraries in India are using Koha ILS such as Delhi Public Library system, Konkan Public Library system etc. There are almost 2500 installations of Koha. The inspiring examples are the National Library of Venezuela (7.5 million volumes), Delhi Public Library (1.4 million volumes), and the United Nations Food and Agriculture Library (1 million volumes). Koha provides mature support for all major library standards including MARC21 (a family of five standards), UNIMARC, Z39.50 (server and client), SRU/SRW, SIP2, OAI/PMH, Unicode etc. Koha presently serves the needs of a wide range of libraries from academic to public and from special and research libraries to corporate libraries.

3.4.3 NewGenLib

NewGenLib or NGL started as commercial ILS in 2005 and made available as open source ILS under GNU GPL in 2008. NewGenLib is the result of collaboration between a charitable trust called Kesavan Institute of Information and Knowledge Management (KIIKM), Hyderabad and Verus Solutions Pvt. Ltd. It is a platform independent ILS that can be installed in both Windows and Unix-like OS. NGL has five functional modules – technical Processing (Cataloging), circulation, acquisitions, serials management and web OPAC including administration for parameters settings and report generation. The features of the ILS are:

- Architecture (completely web based and adheres to International standards, supports web services and allows networking of unlimited number of libraries, database and operating system independent and uses open-source, n-tier, and Java based technologies for scalability, reliability and efficiency);
- Companion software requirements (JAVA SDK as programming environment, PostgreSQL as RDBMS, Apache Ant as Java installer, Lucene and Solr text retrieval engine, Apache Tomcat as web server);
- Standards support (NGL adheres to international standards like MARC21 (bibliographic, authority and holdings formats), ISO 2709, and AACR-2R. Cataloguing database design is based on well proven database design to adhere to MARC and also supports Unicode 4,0 and UTF-16 encoding format, by which it can support all the possible languages);
- Enhanced services (Import of MARC data from sources such as OCLC and freely available web-based resources, Extensive use of setup parameters in configuring the software to suit specific needs, e.g., in management of fines, Multi-user and multiple security levels, Automated email facility integrated into different functions of the software to ensure efficient communication between library and users, vendors, Module-specific querying in all modules);
- Acquisition (Online requests by users, Firm orders, On-approval purchases, Standing orders, Solicited gifts, Unsolicited gifts, Exchange-triggered acquisitions, Web service interfaces to supply sources such as *amazon.com*, Management information reporting to enable better decisions in acquisitions management);
- Cataloguing (supports data-entry using MARC tags, fields, sub-fields, etc., or Simple, label and form based data-entry, Import of MARC records from sources such as OCLC or from free MARC download sites on the web, Access to authority files during data entry and catalogue database searching, Catalogue record attachments enabling access to related data, e.g., multimedia, web-based resources, scanned images, and full text digital documents, Provision of a search engine to search full text documents, Plugins for specialised thesauri, Automatic validation etc.);
- Additional utilities (Network functionalities supports sharing of hardware, server and application software between the host and one or more associate libraries. It helps users of branch libraries - To download metadata or the full text of records, where records are available, into their desktops, In acquisition of new publications from the host library, To access their circulation records, To access electronic journals across all the libraries in the network, To improve services to both the end user and the library staff);
- Circulation (apart from traditional functions supports - Setting of a wide range of circulation options, fines, user privileges, etc., needed in different library environments, Rapid charging, discharging, renewal and reservation operations, Built-in traps for delinquent users, reservations, etc., On-the-fly circulation, Interlibrary transactions, Binding management, Management Information Reporting for better management of collection and Assistance in stock verification);

- Serials control (includes facilities like – Integrated management of serials subscriptions, registration, cataloguing and binding, Rapid registration of incoming serials using a kardex-like interface, Batch and on-demand claiming for missing issues, Support for Union catalogues, ?MIS reporting for better serials management); and
- OPAC (supports - Browser-based access to the library’s catalogue database, Extensive search, retrieval, display, print, download and formatting options for patrons (Customised, text format (brief), Text format (Full), MARC tagging, ISO 2709, MARC-XML, Dublin core), Patrons can request new additions, access their circulation data, make reservations and go to the web via the OPAC, Patrons can trigger interlibrary loans, interact with library staff via instant messages/email).

Special features

Functional modules are completely web based. Uses Java Web Start™ Technology

- Compliant with international metadata and interoperability standards: MARC-21, MARC-XML, Z39.50, SRU/W, OAI-PMH
- Runs on open source components like Java SE, PostgreSQL
- A high degree of scalability
- OS independent - Windows and Linux flavours available
- Z39.50 Client for distributed searching
- Multilingual supports (Unicode 4.0 complaint, easily extensible to support Indic scripts, storage, processing and retrieval of multilingual data)
- Provision for RFID integration
- Alerting and messaging services integrated into different modules of the ILS
- Templates for generation of form letters and applies XML-based OpenOffice templates
- Scope for extensive cutomisation like other open source ILS
- Supports digital media archiving and Android compatible.

Important URLs

- Downloading (<http://www.verussolutions.bis/web/content/download>);
- Documentation (<http://www.verussolutions.bis/web/content/documentation>);
- Users list (http://wiki.koha-community.org/wiki/Category:Koha_Users);
- Help from experts (<http://www.verussolutions.bis/web/content/do-you-need-urgent-help-newgenlib-get-expert-help-free-cost>);
- Forum (<http://www.verussolutions.bis/web/content/forum>); and
- Free support (<http://www.verussolutions.bis/web/content/get-help-librarians-my-region>).

Remark

NGL is the first open source ILS released from India. It is now a matured open source ILS and many libraries are using NGL. It is under continuous development,

for example recently NGL Touch developed as a library kiosk application. The features of NGL ILS are quite suitable for Indian libraries for obvious reasons. Both free and paid supports are available for this ILS along side discussion forum, blog and documentation services.

3.4.4 PMB

Müller (2011) reported that PMB (PhpMyBibli) is improving rapidly and coming up as a fully featured open source integrated library system. The PMB ILS project was started by François Lemarchand in October 2002, the then Director of the Public Library of Agneaux, France. Presently it is managed by PMB Services, an initiative to support open source software. PMB is Web-enabled ILS and is using XAMP architecture (X – any OS; Apache as Web server, PHP as programming environment and MySQL as RDBMS). It is also using AJAX to support interactive and collaborative framework. This software is easy to install in compare with other ILSs from open source domain. It supports both Windows and Linux platform with XAMP architecture. This open source ILS is available in four languages interfaces (English, French, Spanish, Italian). The first version was released in the year 2003 and the current version is 4.1 (released in March 2014). PMB, as open source ILS was initially available through GNU GPL licensing but presently it is available against CeCILL free software license. This platform independent open source ILS supports all basic library automation workflow alongside some advanced features like OPAC 2.0 and electronic SDI service.

System requirements

PMB is based on Web architecture. It means that only server version is required to be installed and in client machines Web browsers (like Firefox, Google Chrome, IE etc) may act as client software to access PMB server. The minimum hardware requirements of server and client machines are as follows:

Server level

- A high-end desktop or entry-level server
- 1GB RAM
- Architecture to run Windows or Unix-like Operating System
- Ports 80 should be opened in firewall for TCP/IP connections to access OPAC and staff client of PMB ILS
- Network to establish TCP/IP connections.

Client machines

- Low-end desktop with any operating system
- A reliable high speed Internet connection for enabling AJAX based services
- 256 MB of RAM
- TCP/IP protocol to connect PMB server at ports 80.

Companion software

Apart from Evergreen server and client software, the server machine requires following companion software to run server version of Evergreen:

- 14) Any Operating System

- 15) MySQL as RDBMS (version 9 or later)
- 16) Apache as Web server (version 2.x)
- 17) PHP programming environment (version 5.x or later).

Major Features

Apart from supporting basic activities and automation operations, PMB is supporting authority file management, linking of subject headings with UNESCO thesaurus in cataloguing interface, Web 2.0 features (such as RSS feed, user tagging), SDI service module, facility to search formula (mathematical and chemical formulae), links to search external sources (Amazon, US books etc), shelf management, basic cataloguing of different document forms, on-line help etc. The regular features are as follows:

- System administration (configuration, parameters settings, security, thesaurus linking, SDI setup, external resource management etc.);
- Acquisitions (purchase management – invoice, order, delivery, invoice, payment, accounting etc, budget control, suggestions management, vendor management, budget control etc.);
- Cataloguing (comprehensive UNIMARC editor, authority data control, Z39.50 client search, in built support of UNESCO thesaurus for subject access fields and authority search, predefined data entry format for different document forms, analytical entry etc.);



Fig. 3.7: Cataloguing in PMB

- Circulation (Member management, easy issue/return, calculation of fines and overdue, facility to manage overdue, hold/reservation management etc.);
- Serials control (new serials management, renewals, loose issue management, holdings management, bindings of back volumes etc.);
- Report generation (basic reports, statistical reports, report groups – borrower related, document related loan related); and

- OPAC (Web OPAC, basic and advanced searching, linking of UNESCO thesaurus in OPAC, search filter by document types, search filter by fields, all field search option, search for external resources, search help, basic content management utility in OPAC, language selection facility in OPAC etc.).

Special features

The Evergreen open source ILS originated as ILS for library consortia and has the credit of many special or unique features such as:

- OPAC and Staff interfaces in four different languages and facilities to switch over language by selecting target language;
- A module to manage alerting service in SDI mode;
- UNIMARC bibliographic format for different document forms;
- Web-OPAC with Web 2.0 features like RSS, user tagging, book review linking etc.;
- Support for OAI/PMH, FRBR, RDF and RDA;
- E-book management options for different formats including e-Pub;
- RFID integration option; and
- XML based export/import.

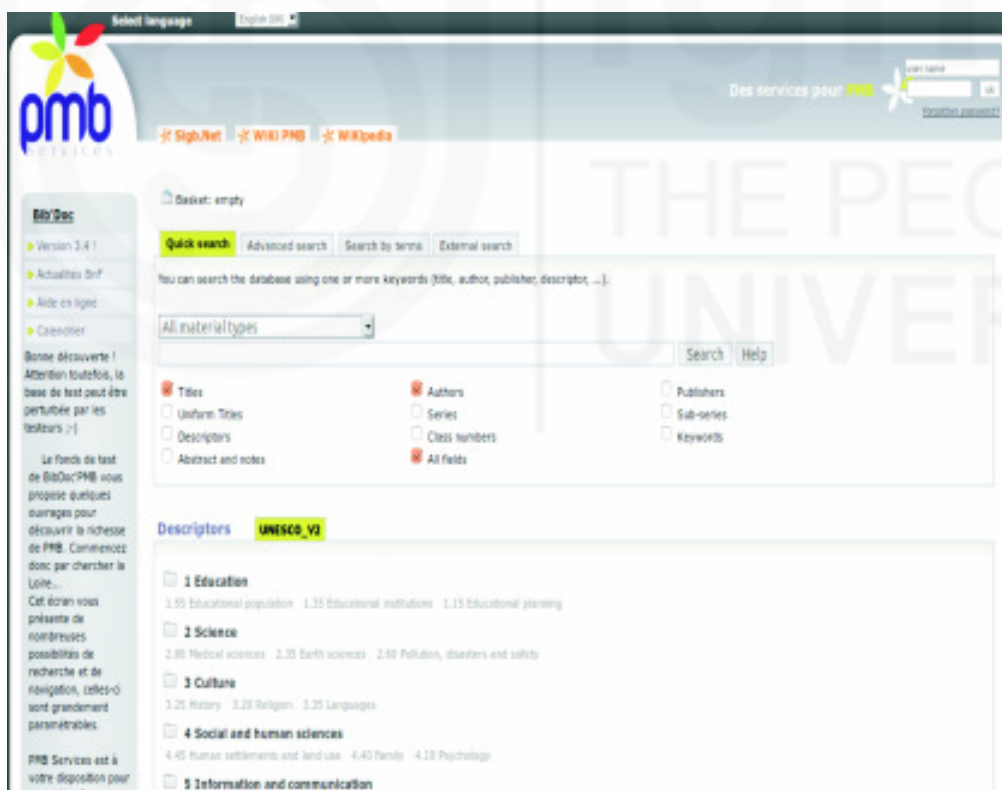


Fig. 3.8: OPAC of PMB

Important URLs

- Downloading (<http://forge.sigb.net/redmine/projects/pmb/files>);
- Documentation (<http://www.sigb.net/index.php?lvl=cmspage&pageid=20>);
- User community (<http://www.sigb.net/index.php?lvl=cmspage&pageid=18>); and

- Technical support (<http://www.sigb.net/index.php?lvl=cmspage&pageid=17>).

Remark

PMB is quite suitable for small and medium scale libraries. The ease of installation and configuration makes it a suitable candidate for public libraries in India. It can be customised to a great extent to incorporate Indian languages. The only problem of this open source ILS is that the PMB portal is available in French language only and this ILS supports only UNIMARC format.

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

- 6) Point out the salient features of any one open source ILS known to you.

.....

.....

.....

.....

.....

- 7) Make a comparison between any two open source ILSs of your choice.

.....

.....

.....

.....

.....

3.5 COMMERCIAL SOFTWARE PACKAGES

Most of the large Indian libraries including elite institutes like IITs, IIMs, NITs, IISc, Universities and big college libraries, corporate libraries have adopted commercial ILSs for automating workflows of the libraries. There are two reasons for it – i) most of these institutes started library automation projects in early 1990s when open source ILSs were not available (remember that Koha, the first open source ILS released in July 2000); and ii) the institutes which started automation projects in early 2000 could not rely on open source ILSs because of the lack of on call support. However, situation in India is changing quickly. Many newly established institutes (such as West Bengal University of Technology, Kolkata, MG University, Kerala) are adopting open source ILSs (mainly Koha and NewGenLib) because of the availability/inclusion of features on regular basis, fund savings opportunities, active discussion forum/ mailing list/software wiki etc and growing user base of open source ILSs. Some of the large scale libraries like British Council libraries (all centres in India) Indian Statistical Institute, Kolkata switched over from commercial ILS (LibSys) to open source ILS (Koha). This unit already categorised and listed commercial ILSs in sub-section 3.3.1

(see Table 2). There are many commercial ILSs in India that are in use. There is a pattern in adopting ILSs in India. The software LibSys, one of the early initiatives in library automation in India, is utilised by most of the large-scale academic libraries all over India but other commercial ILSs are region specific. For example, SLIM ILS (SLIM 21 and SLIM++) is popular in West India (Maharashtra, Gujarat), AutoLib and NIRMALS are popular in South India. As it is not possible to cover all of the commercial ILSs listed in table 2 because of the space limitation, this section discusses only four commercial ILSs on the basis of their huge user base. These are LibSys, SLIM, SOUL and Virtua ILS.

3.5.1 LIBSYS

LibSys (<http://www.libsys.co.in/>) is an indigenous ILS designed and developed by LibSys Corporation, New Delhi in 1984. LibSys is presently available in six different editions/versions to suite requirements of different types of libraries. These are:

LIBSYS 7: This version of LibSys has features like Unicode Support, Federated Searching, Customisable look and feel, User notification through E-mail and SMS, RSS feeds and integration with Google Books, BookFinder, etc. and interactive features like online reviews, ratings, renewals, reservations etc. The modules are – Acquisition, Cataloguing, Circulation, Serials, Article Indexing, Web OPAC, Customisable Reports. LibSys 7 supports following standards – MARC21, Unicode, SRU/SRW, Z39.50, NCIP (NISO), SICI Barcode.

LSEase: The basic features of this version of LibSys are – independent of Operating System, support for digital media archiving, user-friendly workflow, user-defined security, may be extended to Web architecture.

LSAcademia: It is an ERP Solution to integrate administration of academic institutions and ILS. Apart from library management, it supports Admissions, Student Management, Academic Administration, Examination/ Results, Fee Management, Learning Triggers, Time Table, Student/ Parent Portal, Faculty/ Director Portal, Bus Use, Hostel, Staff Management, Payroll, Alumni etc.

LSmart: It integrates RFID and EM hardware from world renowned manufacturers with LIBSYS and thereby offers following add-on services - RFID Tags on Books/Documents and CD/DVDs, Multiple item processing simultaneously, Self-use Kiosk for check-out/check-in, Book Drops for quick check-in of items, Hand held RFID readers for Shelf Management, EAS Security Gates, Books Sorters to reduce items replacement times on shelves.

LSNet: This version of LibSys evolves around a virtual library that includes the collection of books, CD/DVDs, reference material, etc through a single Web-enabled search interface. It may be integrated with LIBSYS 7 to provide platform for sharing e-content, promotion of library materials, value added services like book updates, reviews, upcoming titles etc.

LSDigital: It is a complete Digital Resource Management System (DRMS) which can be integrated with LIBSYS 7 for value-added digital contents dissemination. The integration provides Implicit interaction with LIBSYS database, Full-text and bibliographic searching through LIBSYS OPAC, Converts different data into format of choice (PDF, Doc, etc.), Define & organises library data structure / flow according to needs and Supports various image manipulations

3.5.2 SLIM

SLIM (System for Library Information Management) a client-server architecture based ILS developed by Algorithms consultants Pvt. Ltd., Pune (<http://slimpp.com>). It is a module-based LMS that offers wide range of functionality for library management. Presently there are two versions of SLIM – SLIM 21 and SLIM++.

SLIM 21: The are three levels of SLIM 21 version – Basic Level (Acquisition, Cataloguing, Serials control, Circulation and OPAC); Enterprise Level (Basic Level integrated with Web based OPAC, Selective Dissemination Information (SDI), Inter Library Loan (ILL), Current Awareness Service (CAS), Web Proposals, Statistical Analysis); and L2L Level (Basic level + Enterprise level integrated with Z39.50 client, Z39.50 server, MARC-XML). All of these three levels are supported by additional utilities like Colon classification shelving order, Touch Chip Interface (Biometrics), Newspaper monthly billing, Smart Card / RFID interface, Library Map and News clipping publishing, Multilingual data processing and retrieval, Support for standards like NCIP, SIP2, ISO-2709 etc.

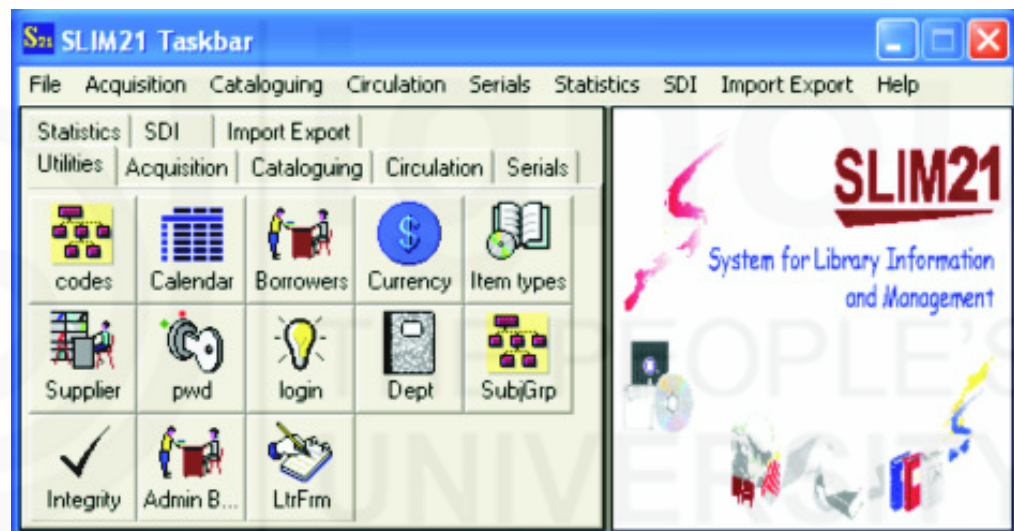


Fig 3.9: SLIM 21 control panel

SLIM ++ is a stripped down version of SLIM 21. It supports export/import through MARC/CCF/ISO-2709 standards and downloading of bibliographic data from online databases through DB Bridge module and Z39.50, generates customised reports on screen/printers/RTF or as text/PDF/HTML files with auto e-mailing facility, supports unicode based LMS that supports multi-script sequencing for Indian scripts, generates shelving order for documents as per colon classification, supports smart card/ RFID based circulation and touch chip (biometric) interface for user authenticity, creates library map for easy location of items and provides user-friendly online help and reference manual.

3.5.3 SOUL

SOUL (<http://www.inflibnet.ac.in/soul/>) is one of the oldest ILS initiative in India. The story of SOUL (Software for University Libraries) started with the development of ILMS (Integrated Library Management Software) by INFLIBNET in collaboration with DESIDOC. INFLIBNET later decided to develop a state-of-the-art, user friendly, Window based system which will contain all the features/

facilities available with other ILSs in the market. As a result, the first version (version 1.0) of SOUL (Software for University Library) released in February 1999 during CALIBER-99 at Nagpur. SOUL uses RDBMS on Windows NT operating system as backend to store & retrieve data. The SOUL has six modules – Acquisition; Cataloguing; Circulation; Serials Control; OPAC and Administration. The modules have further been divided into sub-modules to take care of various functions normally handled by the university libraries. The features of SOUL version 1.0 are: Window based user friendly system with extensive help messages at affordable cost, Client-server architecture based system allowing scalability to users, Uses RDBMS MSSQL to organise data, Multi-user software with no limitation for simultaneous access, User friendly OPAC with web access facility, Supports bibliographic standards like CCF & AACR II and ISO 2709 for export & import facility, Provides facility to create, view & print records in regional languages, Supports LAN & WAN environment and Available in two versions – university library version and college library version. The second version of SOUL, named as SOUL 2.0 was released in January 2009.



Fig 3.10: SOUL 2.0 OPAC

SOUL 2.0 provides two options for back end DBMS - MS-SQL and MySQL. SOUL 2.0 is compliant to international standards such as MARC 21 bibliographic format, Unicode based Universal Character Sets for multilingual bibliographic records and NCIP 2.0 and SIP 2 based protocols for electronic surveillance and control. MARC-XML as standard for export/import, Supports cataloguing of electronic resources such as e-journals, e-books, virtually any type of material, Supports requirements of digital library and facilitate link to full-text articles and other digital objects, Supports ground-level practical requirements of the libraries such as stock verification, book bank, vigorous maintenance functions, transaction level enhanced security, etc.

3.5.4 Virtua ILS

Virtua ILS (<http://www.vtls.com/products/virtua>) is a globally reputed ILS product that offers the full spectrum of library activities. This ILS is designed and

developed by VTLIS Inc., Virginia, US. It uses off-the-Shelf UNIX hardware and the Oracle RDBMS to guarantee continued availability and support. Apart from providing facilities to manage circulation, cataloguing, serials, acquisitions, it also ensures integration with course reserves and managed information environment (integration with student database, institutional repository and so on). All functions are fully integrated, allowing any staff user to access any function at any time according to their library-assigned permissions. The important features of this world-class software are enumerated here in the form of a list.

- System administration (It is fully parameterised software i.e. libraries can configure the setting to achieve maximum flexibility, Basic system includes modules for OPAC, circulation, reserves, cataloguing, acquisition, serials control and reporting,); Provides support for excellent security options at different levels of access, Provides comprehensive customisation parameters (over 1000) for global settings and each subsystem (OPAC, cataloguing, circulation, acquisition, serials control etc, Provides extensive and precise control over user activities and helps creation of rich and customised web interface for various collection components for each patron class;
- Ensures management of multiple libraries or branches across a library);
- Cataloguing (Supports national and international standards for data interchange, Full support for FRBR, FRAD and RDA, Basic system may be supplemented by companion products like RFID, MARC data processing suite, ILL manager and patron self check system, Supports multilingual authority control, and networked multimedia database management and seamless access to multiple databases through Z39.50 client, Supports UNICODE and thereby enables the input and display of different languages in their native scripts. In fact Virtua ILS ensures true multi-lingual catalogue database);
- Acquisition (Comprehensive support for all acquisition activities, Integration with institutional financial system, EDI support);
- Additional utilities (Syndetics content enrichment, OverDrive e-books, Comprise PC reservation and print management, iTiva automated telephone notification as well as most self-check and RFID circulation solutions, Allows data exchange with your student information system or financial management system);
- User interface (Helps designing web-enabled digital media archiving and supports development of digital library database (delivery options include CDROM, DLT, DVD and DAT), Provides ‘security bit’ enabled RFID solution to serve both inventory and theft deterrence functions.

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

8) Point the advantages and disadvantages of using commercial ILSs.

.....
.....
.....

- 9) Discuss the features of any commercial ILS known to you.

.....

.....

.....

.....

.....

3.6 FREEWARE ILSS

Freeware by definition are software that are available free of cost but without the availability of source code. There are some ILSs which are available for downloading and use freely but either they are using companion software which are not open source products (e.g. e-Granthalaya is based on Microsoft products like Windows OS, MSSQL RDBMS and ASP.NET programming environment) or based on non-open source textual database management system (e.g. ABCD and WEBLIS are based on CDS/ISIS). These ILSs are generally used by small-scale libraries like school libraries and rural public libraries. A total of three ILSs are most visible in the freeware ILS domain. e-Granthalaya in India is developed and supported by a reputed government institute National Informatics Centre (NIC), WEBLIS is now supported by UNESCO and ABCD is the product of BIREME (an organisation based in Brazil that develops and maintains information resources for health science in Latin America and the Caribbean).

ABCD

ABCD (Automation of Libraries and Documentation Centers) is a comprehensive Web-enabled integrated library automation system developed by BIREME, Brazil. It is based on CDS/ISIS as back end databases and WWWISIS as middle-ware. The web interface of CDS/ISIS, called WWWISIS was developed by BIREME in 2005. BIRME in 2010 developed ABCD by using CDS/ISIS as database and WWWISIS as CGI script for designing Web-enabled ILS. It includes all major activities generally expected from a third-generation ILS. Core modules are – Cataloging, Circulation, Acquisitions, Statistics and Reports and OPAC. It also includes a facility called “Adds a Site”. This facility is a built-in feature in ABCD to support content management system (CMS). It allows easy production of a library website with integrated meta-search option. In ABCD, cataloguers may use predefined bibliographic formats (like MARC21, UNIMARC, CEPAL) or they may create custom format by using FDT (Field Definition Table) utility of CDS/ISIS. As a whole, ABCD is a very flexible and versatile ILS for use in libraries and information centres where non-standard database-structure create non-bibliographical applications like experts databases, data bank and technology directory. ABCD (present version is 1.0) includes two circulation interfaces – i) standard loans-module; and ii) advanced loans module. The advanced circulation module provides external links with SQL-databases. The upcoming version 2.0 of ABCD will include digital media archiving module. This module will provide facility to handle textual objects and multimedia objects with full-text indexing facilities. The problem of ABCD is that it is not Unicode-compliant (the problem is inherited from CDS/ISIS) and therefore, cannot handle Indic scripts based documents. ABCD is available under GPL (version 3) and independent of

Operating System (browser based cross-platform system) with standards support like MARC 21, MODS, OAI, XSLT. The programming environments are open source components like Java, JavaScript and PHP. As a whole ABCD is based on an array of technologies like ISIS database, ISIS formatting language, CISIS, ISIS Script, ISIS NBP, Java Script, Groovy and Jetty, PHP, MySQL, Apache and YAS

Resources:

- Technological features (<http://reddes.bvsaude.org/projects/abcd/wiki/Features>);
- Wiki (<http://wiki.bireme.org/en/index.php/ABCD>);
- Download (http://bvsmodelo.bvsalud.org/download/abcd/ABCD_1.0_wis_full.exe);
- Project homepage (<http://reddes.bvsaude.org/projects/abcd>).

e-Granthalaya

e-Granthalaya has improved a lot recently through continuous up-gradation. The current release (version 3.0) supports almost all core activities of an ILS alongside advanced features like e-book management, Web-OPAC, predictive serials control, Unicode-compliant multilingual support, easy data migration and MARC 21 support for both bibliographic and authority data. This ILS is a product of National Informatics Centre (NIC), Department of Electronics & Information Technology, Ministry of Communications and Information Technology, Government of India. The only problem of e-Granthalayas is its dependency on Microsoft products (commercial close source software) like VB.NET or ASP.NET and MSSQL server 2005. The software can be implemented either in stand-alone or in client-server mode. In client-server mode database and WebOPAC are installed on the server PC while the data entry program is installed on client PCs. The version 3.0 of e-Granthalaya supports union catalog output. The major features of this freeware ILS are as follows:

- Technological features (runs on Windows Platform Only (Win XP/vista/7/8/Server 2003/2008) on LAN/WAN environment, UNICODE Compliant, supports data entry in local language);
- Administration (Module - Wise Permission to the software Users, Workflow as per Indian Libraries and Retro-Conversion as well as Full Cataloguing Modes of Data Entry, Library Statistics Reports);
- Cataloguing (Authority Files/ Master tables for Authors, Publishers, Subjects, etc, Multi-Vol, Multi-Copy and Child-Parent Relationship pattern, Z39.50 Client Search Built-in, Export Records in CSV/Text File/MARC 21/MARC XML/ISO:2709/MS ACCESS/EXCEL formats, Centralised Database for member libraries, Import Data from any structured Source (MARC21/EXCEL), Generate Bibliography in AACR2, Data Entry Statistics Built-In, e-Books management with digital files in pdf or other formats);
- Acquisition (Main/Branch Libraries Acquisition/Cataloguing, Print Accession Register, Bulk accessioning in single click, Budget and account control, Budget Modules with Bill Register Generation, Manages multi-budget heads, Exchange rates, Report generation, Printing accession register etc.);

- Circulation (Issue/return, Membership module, Bar-coding support, comprehensive circulation reports);
- Serials control (Subscription/renewal with auto-generate schedule, CAS/SDI Services and Documentation Bulletin, Micro-Documents Manger (Articles/Chapter Indexing));
- OPAC and Utilities (Search Module built-in with basic/advance/boolean parameters, Full Text News Clipping Services, Digital media integration with uploading / downloading of pdf/html, etc documents, Web Based OPAC Interface, Photo Gallery available for uploading photo and pictures of the organisations - published on the Library Web site).

Resources

- Portal (<http://egranthalaya.nic.in/>);
- Forum (https://lsmgr.nic.in/mailman/listinfo/egranthalaya_forum);
- Software request (<http://egranthalaya.nic.in/Request%20Form.pdf>);
- Documentation (http://egranthalaya.nic.in/eG3_UserManual.pdf).

WEBLIS

WEBLIS stands for Web based Library and Information System. This Web based ILS is based on CDS/ISIS. It has been developed by the Institute for Computer and Information Engineering (ICIE), Poland by combining CDS/ISIS and WWW-ISIS engine (also developed by ICIE). It is freeware ILS and provides basic library workflow support through four modules – Cataloguing system, OPAC (search), LOAN module, Statistical module. WEBLIS is presently supported by UNESCO. The features of these four components of WEBLIS are:

- 1) Cataloguing system (module is supported by WWW-ISIS data entry facilities and allows management of different document types with support for powerful validation tools, Provision of integrated on-line thesaurus, Availability of model data entry worksheet etc.);
- 2) Circulation (Issue/return, Hold/reserve management, Auto generation of claiming (by e-mail or a traditional mail in word form), Task schedule, Authorised circulation (through password authentication), Member management, Member management, Loan statistics etc.);
- 3) OPAC (Simple and advanced search, Search history, Saving queries function, and ISIS Query language facilities, Thesaurus based search support, ISO-2709 based export/import);
- 4) Statistics (Generate statistical data aggregated from the CDS/ISIS databases, Statistical analysis may be defined in a spreadsheet, Statistical data can be stored in given database).

Resources

- UNESCO Portal (http://portal.unesco.org/ci/en/ev.php-URL_ID=16841&URL_DO=DO_TOPIC&URL_SECTION=201.html);
- Download (<http://www.unesco.org/webworld/weblis/Weblis070826.sip>);
- Documentation (<http://www.unesco.org/webworld/weblis/WEBLIS-DOC.sip>);

Self Check Exercises

- Note:** i) Write your answers in the space given below.
ii) Check your answers with the answers given at the end of this Unit.
- 10) What is freeware ILS? List major freeware ILSs.

.....
.....
.....
.....
.....

- 11) Discuss the features of e-Granthalaya. What are the problems associated with this ILS?

.....
.....
.....
.....
.....

3.7 EVALUATION OF SOFTWARE PACKAGES

Evaluation of ILS is an important task for library professional in selecting an ILS for procurement and for migration from one ILS to another. Evaluation criteria must be framed on the basis of factors like: i) type and size of the library system; ii) nature of library services; iii) requirement of technical skills to handle the ILS; iv) use of ILS in neighbouring libraries; v) time needed to perform migration as well as regular maintenance; vi) compliance of ILS with global standards in the domain of library services and interoperability; and vii) fund requirements for capital and recurring expenditure (remember procurement of ILS is not one time capital expenditure, it also involves recurring cost for annual maintenance and regular updation). This section discusses the issues related with ILS evaluation in three heads – generic parameters, specific parameters for commercial ILS and parameters for open source and freeware ILS.

3.7.1 Generic Parameters of Evaluation

Experts differ in clustering the factors or parameters for ILS evaluation. This section attempts to group evaluation parameters into three broad groups – generic parameters, specific parameters for evaluation of commercial ILSs and parameters applicable for open source and freeware ILS. The generic parameters of evaluation for an ILS are applicable to all sorts of ILS irrespective of the origin of these products. The generic parameters (as devised by Mukhopadhyay in 2006) that should be taken into consideration are as follows:

Services availability checklist: An ILS is ranked by the services it provides. Evaluation of a typical third generation ILS should be based on the following core, enhanced and value-added services (Mukhopadhyay, 2006)–

- *Core services:* Acquisition, Cataloguing, Circulation, OPAC, Serials control, Bibliographic format support, Data exchange format support, Article indexing, Retro conversion, Standard report and System administration.
- *Enhanced services:* Customised report generation, GUI based user interface, Reservation facility, Interlibrary loan module, Multi-lingual support, Union catalogue, Authority file support and controlled vocabulary, Online help, Online tutorial, Power search facility, Internet support, Intranet support, Web access OPAC, Multimedia interface, Barcode support and Backup utility.
- *Value-added services:* Patron self service through RFID & Smart card (self circulation, self reservation etc.), Online user training/orientation, Stock verification facility, Members photo ID card generation, Barcode generation, Fine calculation & receipt generation, Gate pass generation, Bulletin board services & e-mail reports, Electronic SDI, CAS support, Digital media archiving support.

Functional checklist: The following general features are part of software module testing, and each functional activity must be tested or conducted during the evaluation process:

- Searching Capabilities (All modules)
- Data Entry and Editing (All modules)
- Bibliographic/item File and Maintenance
- Cataloguing editor (Cataloguing)
- Authority Control (Cataloguing)
- Inventory (Circulation)
- Check-out (Circulation)
- Renewal (Circulation)
- Circulation/Management Reports (Circulation)
- Check-in (Circulation)
- Fines and Fees (Circulation)
- Notice Production (Circulation)
- Holds (Circulation)
- Recalls (Circulation)
- Patron File (Circulation)
- Reserves (Circulation)
- Portable Back-up Units
- Report Writer
- Acquisitions
- Serials
- Electronic Databases
- Gateways
- Network Operations
- Z39.50 Client and Server

- Inter-Library Loan
- Web Accessibility
- Integrated Archiving
- Self Registration
- Statistics Generation
- Export and Import
- Fund Accounting
- Digital media archiving.

Data conversion and backup utility: The ability of the ILS in terms of support for data conversion from other library systems and adherence to the international bibliographic data standards and protocols should be checked extensively. In this age of shared cataloguing systems and web integration, the ILS should also support metadata schemas and interoperability issues like XML, RDF and OAI/PMH. Backup facility in suitable media is also to be checked in view of data recovery at the time of need.

Standards compliance: In Unit 1 (sub-section 1.4.1) of this block, we already discussed the standards that need to be supported by a typical ILS. The minimum essential standards are – ISO–2709 for bibliographic data interoperability; Standard bibliographic formats compliant with ISO - 2709 (e.g. MARC 21, UNIMARC, CCF/B); Z39.50 protocol standard for distributed cataloguing;

Z39.71 standard for holdings statements; BS ISO 9735-9:2002 Electronic data interchange for administration, commerce and transport (EDIFACT); Z39.83-1 (NISO Circulation Interchange Part 1: Protocol (NCIP)); Z39.83-2 (NISO Circulation Interchange Part 2: Protocol (NCIP)); ISO/CD 28560-1 (Information and documentation — Data model for use of radio frequency; identifier (RFID) in libraries — Part 1: General requirements and data elements); ISO/CD 28560-2 (Information and documentation — Data model for use of radio frequency; identifier (RFID) in libraries — Part 2: Encoding based on ISO/IEC 15962); ISO/CD 28560-3 (Information and documentation — Data model for use of radio frequency identifier (RFID) in libraries — Part 3: Fixed length encoding); and ISO/IEC 10646: 2003 (Universal Multiple-Octet Character Set or UCS).

The global de facto standards for interoperability that should be supported by an ILS are – MARCXML - MARC 21 data in an XML structure (developed by Library of Congress - <http://www.loc.gov/standards/marcxml/>) acting as base standard for bibliographic data export/import in place of ISO-2709; MODS (Metadata Object Description Standard) - XML markup for selected metadata from existing MARC 21 records as well as original resource description (developed by Library of Congress – <http://www.loc.gov/standards/mods/>); MADS (Metadata Authority Description Standard) - XML markup for selected authority data from MARC21 records as well as original authority data (developed by Library of Congress – <http://www.loc.gov/standards/mads/>); METS (Metadata Encoding & Transmission Standard) - Structure for encoding descriptive, administrative, and structural metadata (developed by Library of Congress -<http://www.loc.gov/mets/>); PREMIS (Preservation Metadata) - A data dictionary and supporting XML schemas for core preservation metadata needed to support the long-term preservation of digital materials (developed by Library of Congress –

<http://www.loc.gov/standards/premis>); SRU/SRW (Search and Retrieve URL/ Web Service) - Web services for search and retrieval based on Z39.50 (developed by Library of Congress - semantics <http://www.loc.gov/standards/sru/>); and OAI/PMH Version 2.0 - Open Archive Initiative/Protocol for Metadata Harvesting (developed by Open Archive Initiative).

Hardware and third party software requirements: The ILS should provide a complete list of hardware requirements (processor type and RAM) for server and client machines, operating system requirements and back end RDBMS (with version) requirements. Evaluation should be based on total cost for minimum hardware and third party software requirements of the package.

Performance testing: Any ILS should be evaluated by checking some performance testing like transaction throughput capacity and response time, hardware functionality, module functionality, conversion testing, database loading, index building etc.

3.7.2 Specific Parameters of Evaluation for Commercial ILSs

Vendor validity: The reputation of software development group or the vendor is extremely valuable. The following questions should be raised to judge the validity –

- Is the vendor also the software developer, or is the vendor a distributor or agent for the software developer?
- Is there an international presence or is the company localised?
- How long has the software developer been in the library systems industry?
- How long has the library system you are interested in been on the market?
- Who use their products? (Look for someone in close proximity and contact him or her with questions. If possible, make an on-site visit to see the product in action.)

Training, Documentation and Customer support: The vendor must provide:

- Adequate training facilities without fees for supervisor and operators
 - To manage and operate the system on a day-to-day basic
 - To run file backup operations, software utilities and cataloguing utilities
 - To troubleshoot and solve simple problems and load software enhancement received from the vendor.
- Complete documentation (in hard copy and machine-readable form) must be available with the package along with regular documentation updates and release notes available for local printing or downloading via www including online help for modules and OPAC search.
- The package must have support from the software vendor for hardware and software maintenance, data conversion, emergency and on-call support and disaster management.

3.7.3 Specific Parameters of Evaluation for Freeware and Open Source ILSs

Public Library Association (PLA) working under ALA recommended a set of criteria in selecting open source ILS for library (see <http://www.ala.org/pla/tools/technotes/opensourceils>). These criteria apart from the general criteria discussed above must be kept in mind in selecting open source ILS. The minimum essential criteria specifically meant for open source ILSs are as follows –

- **Currency and regular releases:** The open source ILS under consideration must have at least two substantial releases a year along with a road map for future development activities.
- **Core modules:** All core activities of a library like acquisition, cataloging, circulation, serials control, systems administration and patron access catalog modules must be available. Value-added services that require to run library operations smoothly (like barcode generation, fine calculation, gate pass printing, member card printing, web-OPAC etc.) must be included in road map of development.
- **Standard Data Formats:** MARC 21 family of standards (at least MARC 21 bibliographic format and Authority format) should be supported alongside export/import facilities (based on ISO-2709/MARC-XML). Availability of UNIMARC format in addition to MARC 21 standards is an added advantage.
- **IPR and Licensing:** Current source code and technical documentation are available for downloading under the GNU General Public License.
- **User base:** The product is currently in use in a significant number of libraries.
- **Scalability:** Scalability should not be an issue; it means there should be no risk of database size or activity levels exceeding the capacity of the software.
- **Developer group:** A dedicated group of developers ensures the progress of open source ILS under consideration such as adopting cutting edge technologies in developing new features and facilities.

Of course, the main OSS ILS in the U.S., Evergreen and Koha, meet all of these criteria. Libraries that have already decided to choose one of these systems will need to consider other factors. The Massachusetts Library Network Cooperative has released a useful list of points comparing these systems (<http://masslnc.cwmars.org/node/1892>).

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

12) Why do we need a framework for ILS evaluation? Enumerate the factors to be considered in selecting ILS.

.....

.....

.....

.....

13) What are specific factors to be considered in selecting open source ILS?

.....

.....

.....

.....

.....

3.8 GLOBAL RECOMMENDATIONS

ILSs are changing fundamentally to meet the challenges of network era and as a direct result of this transformation the difference between automated library system and digital library system is blurring day-by-day. We already covered the role of global recommendations in shaping ILSs and basic recommendations as proposed by ILS-DI and OLE in sub-sections 1.5.1 and 1.5.2 of Unit 1 (block 1 of course 9) respectively. Here we are going to study major technical recommendations advocated by these two global agencies.

DLF ILS Discovery Internet Task Group (**ILS-DI**) Technical Recommendations are acting as pathfinders for advancement of ILSs or Library Management Systems (LMSs) globally. These recommendations were developed in 2008 by Digital Library Federation (DLF) to guide inter-operation between integrated library systems and external discovery applications (DLF, 2008). These recommendations are under continuous revision. The major ILS-DI recommendations may be grouped as follows:

General

- Improve discovery and use of library resources via an open-ended variety of external applications that build on the data and services of the ILS;
- Articulate a clear set of expectations;
- Make recommendations applicable to both existing and future systems and technologies;
- Support interoperability and cooperation with applications outside the traditional library domain;
- Ensure that the recommendations will be feasible to implement; and
- Be responsive to the user and developer community.

Interoperability, Functionality and Standard Compatibility

- Basic Discovery Interface (BDI) should support applications that provide discovery outside the ILS;
- BDI should include a broad range of practical discovery tools that operate in tandem with the OPAC;
- BDI may be linked with domain-specific discovery platforms (e.g. courseware repository in case of academic libraries and community information resources in case of public libraries);

- BDI should facilitate metadata harvesting, availability checking for resources (within and outside of library system) and bibliographic request functionality;
- Data aggregation, Real Time search, Patron functionality, and OPAC interaction;
- Compatibility with the established and emerging standards like OAI/PMH, SRU/SRW, METS, MODS, DCMES, MARC-XML, NCIP etc.;
- Facilities to expose bibliographic records to different external discovery tools (such as SOPAC, Vufind, etc.).

Data aggregation

- Many external discovery applications need to maintain external copies of ILS data and thereby supports should be provided for extracting, or harvesting, ILS data (bibliographic, authority, holdings, and other item metadata (such as circulation information) in bulk;
- Facilities must be provided for – selective harvesting for external metadata transformation, cleanup, relationship (FRBRising), vocabulary mapping and other processing services;
- Bibliographic records should be in a well-specified format and each record should have a unique persistent identifier;
- Bibliographic records must be available in interchangeable native format (for example, a MARC record stored as relational table elements could be returned as native marc21, or as MARC-XML schema, or DCMES or MODS and METS; and
- Support for compatibility with different text retrieval engines (for example, a Lucene index of bibliographic records that can be searched with facets using Solr).

Search and retrieval

- Integration of ILS with digital library system or other application requires the capacity to perform rich, real time searches as a mission-critical feature;
- ILS should provide XML-based protocol like SRU/W (SRU and SRW) for distributed search apart from traditional library-centric search protocol like Z39.50;
- Enabling the ILS as a target for meta-searching via a standard federated search product or other discovery tool (with inclusion of features like result paging, sorting, and query filtering);
- Search system should display real time availability of results (both at the bibliographic level and at the item level), rather than availability data;
- Search system should be able to storing, processing and retrieving of Unicode-compliant multilingual documents;
- Full authority records should be available for Real Time Search. Like bibliographic and holdings information, authority information can be expressed using the MARC 21 authority format (<http://www.loc.gov/marc/authority/>).

Patron Functionality

- Library system should note that patrons use the OPAC for more than just discovery – they also use it to manage their account and request delivery of discovered materials;
- System should ensure patron authentication, patron account retrieval, and circulation/delivery transactions;
- System should support standard protocols like NCIP and SIP2;
- Patrons must be able to retrieve all the personal information (like fine information, hold request information, loan information, messages etc.);
- System must support privilege control facilities to provide selective functionalities to patrons.

User Interaction

- Interface should have provision for adding links to external resources from within the OPAC;
- Availability of federated search mechanism is desirable;
- System should support standard protocols openURL;
- System must support interactive user interface for user-driven tags, comments, reviews and ratings.

The abstract reference model of **OLE project** centres on seven fundamental functions of library systems. The major recommendations are as follows –

Select Entity

This function describes the processes of acquisition of an entity and includes workflow like Obtain Metadata and Create Metadata. The resources may be gifts, approval plan items, firm orders, interlibrary loan requests, reserve requests, remote location requests, publication references, trial databases. Metadata can be obtained (if available) or created for descriptive, holdings (e.g. what is available and being considered for acquisition), authority, financial, or other types. The metadata may be harvested from or deposited by another system.

Acquire Entity

Associated license/registry terms are managed and documented within the system through this function. The workflow includes – selection of entity, assigning supplier/vendor, fund management, determine claiming cycle etc. The invoice process and payment activity may be executed manually or electronically (by using protocols such as: EDIFACT; ANSI X12, XML EDI.).

Describe Entity

This function is associated with description of physical or digital entities (resources, collections, people, organisations, services, events, courses, facilities, finances, relationships, etc.). It includes process to obtain, create, modify, delete, or expose metadata for an entity.

Deliver Entity

This function describes the process where a user submits a request for a service or resource and entity supplied to him/her to satisfy information demand. Entities cover a wide range like physical/digital, returnable/consumable, free/fee based, local/trans-local, and ownership/external.

Manage Entity

This function covers processes that track the life-cycle of an entity including preservation, conservation, evaluation, retention, relocation, duplication, version preference, rights management, binding, repair, reformat, replacement, and withdraw. The workflow includes Preserve/Conserve Resource, Manage Inventory, Configure Metadata, Manage Rights, and Reformat Resource.

OLE recommendations are very promising in developing futuristic ILSs. One example of such application is KualI ILS, an extensible service-driven library management system. KualI is an enterprise-ready, community-source software package developed on the basis of OLE recommendations. It manages and provides access not only to items in library collection but also to licensed and local digital contents. KualI ILS has four major OLE components –

Select and Acquire Module

- This module of KualI developed on the basis of Open Library Environment (OLE) recommendations and includes Financial, Selection, Acquisitions, Receiving, Payment/Invoicing, Licensing, and Electronic Resource Management (ERM), a component that supports operational processes for demand-driven acquisitions of library resources.

Describe and Manage Module

- This module is based on OLE's user-friendly interface that allows library staff to create and manage core metadata relating to library resources such as bibliographic data, localised holdings, and electronic resources access information.

Deliver Module

- This module covers the interactions between the library, its collection, patrons and discovery systems and provides the basic features/functions to manage patron records, item records, circulation tasks, holds management, fine calculation, NCIP standards compliance with local parameters e.g., patron-related blocks, item-related blocks, loan periods, notice types and notice frequency, etc.

System Integration

- Systems Integration is the link between the three modules: Select and Acquire, Describe & Manage, and Deliver. KualI uses a common middleware suite called KualI Rice to achieve service oriented architecture (SOA). The SOA supports interoperability related with identity management, acquisitions/financial accounting, course and learning management, and student information systems.

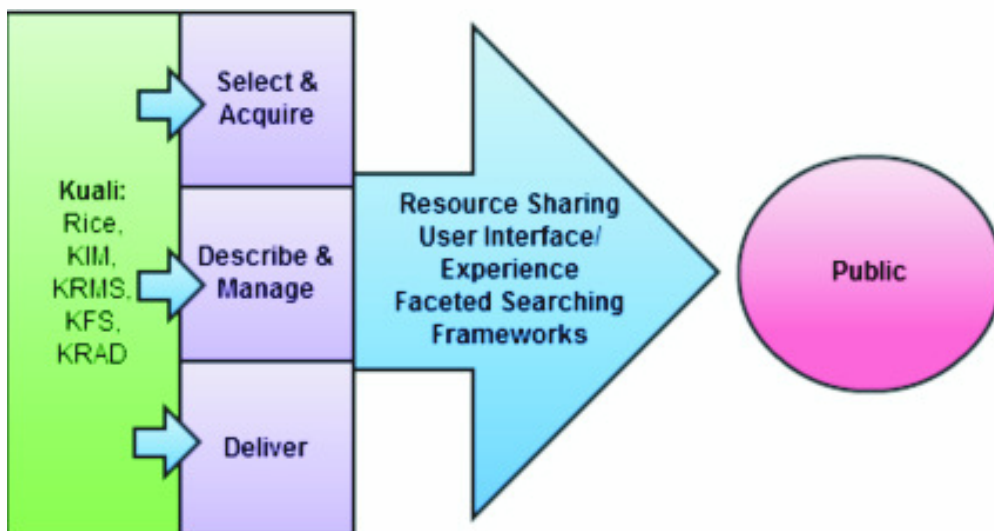


Fig. 3.10: SOA of Kualii ILS based on OLE recommendations

Source: http://www.kualii.org/sites/default/files/ole/system_integration.png

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

14) Draw a summary of OLE recommendations.

.....
.....
.....
.....
.....

15) Discuss how Kualii ILS is applying OLE recommendations.

.....
.....
.....
.....
.....

3.9 SUMMARY

This Unit covered ILS available in India in depth. It provided a historical and theoretical foundation of library automation software development spanning last sixty years and under five different generations. Five generations of ILSs against a set of parameters framed in view of the technologies in use and services expected to be available have been compared. After discussing features of different generations of ILS, comparison of ILSs available in India on the basis two trains of characteristics – distribution policy (commercially available ILS, open source ILS and freeware ILS) and place of origin (foreign, Indian and originated in

foreign and developed in India) has been done. This Unit discussed features of four most promising open source ILSs, four commercial ILSs (selected on the basis of their user base in India) and three visible freeware ILSs. As evaluating exercise is considered as one of the most important tasks in library automation process, this Unit discussed evaluation parameters under three heads – generic (applicable to all kinds of ILS irrespective of distribution policy and place or origin), specific parameters to be considered for evaluating commercial ILSs and parameters important for evaluating open source ILSs. This Unit ends with a brief discussion on two sets of global recommendations in the domain of library automation namely ILS-DI recommendations and OLE recommendations. It also throws light on the impact of these recommendations in future development of ILS.

3.10 ANSWERS TO SELF CHECK EXERCISES

- 1) The role of typical library automation software is to manage two major subsystems of a library – operational subsystem and administrative subsystem. Apart from the core activities like acquisition, cataloguing, serials control, circulation and public access interface, an ILS provides many value-added services like online acquisition, FRBRised cataloguing, RFID-enabled circulation, member card printing, bar-coding of accession number and member ID, predictive mode of serials control, interactive OPAC, federated searching, extensive reports and statistics in different formats for supporting decision making process etc.
- 2) Third and fourth generation ILSs mainly differ in the context of – i) architecture (client-server vs. Web-enabled); ii) database technology (entity-relationship vs. object-oriented); iii) standardisation (bibliographic vs. all round); media support (limited support vs. extensive support); and distribution mode (mainly commercial vs. both commercial and open source).
- 3) The major features of the fifth generation ILSs are – AJAX support, Support for FRBR, FRAD and FRSAD. Support for Linked Open Data, Use of open interoperability standards, provision of Cloud and Web-scale resource discovery, and Support for federated search.
- 4) Open source ILSs are available freely under GNU GPL license, extensively customisable (as source codes are available) and based on global open standards in the domain of library automation. The major open source ILSs are Koha, Evergreen, PMB, Avanti, NewGenLib and so on.
- 5) ILSs available in India may be grouped on the basis of two trains of characteristics – distribution policy (close source and open source) and place of origin (foreign origin, Indian origin and hybrid). as per the distribution policy (conditions for availability of software), software may be grouped into two broad divisions – close source software and open source software (OSS). Close source software therefore, may again be placed in two groups – commercial software and freeware. As per the place of origin, ILSs may be grouped under three fundamental categories – ILSs of foreign origin, ILSs developed over ILSs (or textual database management systems) of foreign origin and ILSs of Indian origin. This grouping may again be sharpened by dividing the packages on the basis of size of library systems

i.e. large library system, medium range library system and small range library system.

- 6) There are many open source ILSs of which Koha appeared first in the year 2000. It is now considered as the most feature rich open source ILS in the world. The user base of Koha is increasing rapidly all over the world. Many libraries are switching from commercial ILS to Koha because of the following features – i) Web-centric architecture; ii) compliant with all major standards in the domain of library automation; iii) OPAC 2.0; iv) use of open source companion software; v) multi-lingual and Unicode-compliant; vi) supports all core and value-added features expected from fourth generation ILS packages; and vii) OPAC available in 25 languages.

- 7) A comparative study of Koha and Evergreen may be represented as below:

Koha	Evergreen
Web-centric architecture	Client-server architecture
Meant for individual library but may be extended to manage library network or library consortia	Meant for library network or library consortia but may be deployed in individual library
Uses MySQL as back end RDBMS	Uses PostgreSQL as back end RDBMS
Applies PERL modules	Applies OpenSRF

- 8) The advantages of using a commercial ILS are – i) less responsibility on the part of the librarian; ii) on call support service; iii) arrangement of training by vendor; iv) up gradation is responsibility of vendor; v) customisation is fee based vendor activity; and vi) light learning curve.

The disadvantages are – i) no customisation of workflow; ii) non transparent use of standards; iii) huge capital and recurring expenditure; iv) problem in data transfer and migration; v) vendor dependency in every step; and vi) slow release cycle.

- 9) Virtua ILS, a product of VTLS Inc, US, is one of the most comprehensive ILSs at the global scale. The real advantages of this ILS are – i) compliance with all global standards of library automation, ii) full support for bibliographic data models like FRBD, FRAD, FRASD; iii) provision for RDA based cataloguing along side MARC 21 and AACR 2; v) full support for Web 2.0 architecture to generate interactive user interface; vi) very sophisticated search mechanisms; viii) facility to create customise workflow for library and many more such facilities. Virtua ILS is used by many national libraries including National Library of India.

- 10) Freeware ILSs are available for downloading and use freely but either they are using companion software which are not open source products (e.g. e-Granthalaya is based on Microsoft products like Windows OS, MSSQL RDBMS and ASP.NET programming environment) or based on non-open source textual database management system (e.g. ABCD and WEBLIS are based on CDS/ISIS). The visible freeware ILSs are e-Granthalaya, ABCD and WEBLIS.

- 11) The current version of e-Granthalaya (version 3.0) is client-server mode integrated library automation package that supports almost all core activities of an ILS along side some value-added services like news clippings, CAS/SDI, article indexing, digital media archiving etc. It also supports many library standards like MARC 21, MARC-XML, ISO-2709 and S39.50 protocol. The main disadvantage of this ILS lies on it's heavy dependency on Microsoft products (Windows OS, MSSQL, VB.NET/ASP.NET) which are not open source software product. As a result a library is getting this freeware ILS at no cost but companion software procurement places huge financial burden on the library budget.
- 12) A framework for evaluation of ILS is required for three major purposes –
 - i) selection of an ILS for procurement from a short-listed group of ILS; and
 - ii) selection of an ILS for migration from one ILS to another; and
 - iii) development of RFP for seeking expression of interest (EOI). The parameters of selection must be based on following factors –
 -) service availability checklist and standards support checklist;
 - ii) functional features;
 - iii) companion software requirement;
 - iv) hardware support required;
 - v) vendor reputation (in case of commercial ILS),
 - vi) project duration and release cycle (in case of open source ILS);
 - vii) data conversion and transfer support;
 - viii) software architecture;
 - ix) support for cutting edge technologies (like AJAX, Web 2.0, Linked Open Data) and
 - x) support for training, documentation, on-call service (availability of forum, wiki and mailing list in case of open source ILS).
- 13) The following specific parameters, apart from the generic parameters should be checked in selecting an open source ILS – Currency and regular releases, Core modules support, Standard Data Formats, IPR and Licensing, User base, Scalability, and reputaion and duration of Developer group.
- 14) Open Library Environment project (OLE project - <http://oleproject.org>) or the OLE project, funded by Andrew W. Mellon Foundation has started in early 2000. As a whole, the OLEproject report for future ILSs may be summarised under following heads –
 - 1) Flexibility (Supports for wide range of resources; accessed by a wide range of customers in a variety of contexts);
 - 2) Community ownership (Advocates systems that are designed, built, owned, and governed by and for the library community on an open source licensing basis);
 - 3) Service Orientation (Prescribes technology-neutral service-oriented framework that ensures the interoperability of library systems);
 - 4) Enterprise-Level Integration (Facilitates integration with other enterprise systems such as research support, student information, human resources, identity management, fiscal control, and repository and content management);
 - 5) Efficiency (Provides a modular application infrastructure that integrates with new and existing academic and research technologies);
 - and 6) Sustainability (Creates a reliable and robust framework to identify, document, innovate, develop, maintain, and review the software necessary to further the operation and mission of libraries).
- 15) Kualī – Open Library Environment or simply Kualī-OLE is an experimental ILS, developed by Kualī Foundation Inc and funded by Andrew W. Mellon Foundation right from January 2010, to achieve the goals of OLE project. The final product is due in late 2014. It is based on six fundamental criteria

as set by OLE project for future ILSs. It is trying to implement following OLE features in the ILS product – Built, owned, governed by the academic and research library community; Supports a wide range of resources and formats of scholarly information; Interoperates and integrates with other enterprise and network-based systems, Supports federation across projects, partners, consortia, and institutions, Provides workflow design and management capabilities and Offers information management capabilities to non- library efforts.

3.11 KEYWORDS

- Bibliographic metadata:** Information about a resource that serves the purpose of discovery, identification and selection of the resource. Includes elements such as title, author, subjects, etc.
- EDI** : Electronic Data Interchange (EDI) is a standard method for exchanging structured data, such as purchase orders and invoices, between computers to enable automated transactions.
- EDIFACT** : EDI For Administrations, Commerce and Transport
The concept of utilising a single set of specifications for bibliographic records regardless of the type of material they represent.
- ERMS** : Electronic Resources Management System is used to manage a library’s electronic resources, primarily e-journals and databases. Systems can include features to track trials, license terms and conditions, usage, cost, and access.
- Evergreen** : The first open source ILS designed to handle the processing of geographically dispersed, resource-sharing library networks and library consortia.
- GPL** : The GNU General Public License is an open source license that is used by Evergreen and Koha.
- ILS** : An automated library system that utilises shared data and files to provide interoperability of multiple library functions, e.g. cataloging, acquisition, circulation, serials, etc.
- Interoperability** : The ability for two different computer systems to communicate and exchange information in a useful and meaningful manner.
- MARCXML** : A metadata scheme for working with MARC data in a XML environment.
- Metadata** : Structured information that describes an information resource. “Data about data” for an information bearing object for purposes of description, administration, legal requirements,

- technical functionality, use and usage, and preservation.
- Metadata harvesting** : A technique for extraction of metadata from individual repositories for collection into a central catalog.
- Module of ILS** : Functions specific to a particular system capability such as the online public access catalog, cataloging, acquisitions, serials, circulation, etc.
- NCIP** : NISO Circulation Interchange Protocol (NCIP) is a standard which defines a protocol for the exchange of messages between and among computer-based application to enable them to perform functions necessary to lend and borrow items, to provide controlled access to electronic resources, and to facilitate co-operative management of these functions.
- Open Source** : A concept through which programming code is made available through a license that supports the users freely copying the code, making changes it, and sharing the results. Changes are typically submitted to a group managing the open source product for possible incorporation into the official version. Development and support is handled cooperatively by a group of distributed programmers, usually on a volunteer basis.
- OpenSRF** : Open Service Request Framework is developed by Evergreen ILS team to achieve load balancing and service availability.
- SIP2** : Standard Interface Protocol Version 2 is a standard for the exchange of circulation data and transactions between different systems.
- SOA** : Service-Oriented Architecture (SOA) is a software framework for managing loosely-coupled, distributed services which communicate and interoperate via agreed standards.
- SRU** : Search/Retrieve via URL is a standard search protocol for Internet search queries, utilising CQL (Common Query Language), standard query syntax for representing queries.
- SRW** : Search/Retrieve Webservice is web services implementation of the Z39.50 protocol that specifies a client/server-based protocol for searching and retrieving information from remote databases.

- Unicode** : A universal character-encoding standard used for representation of text for computer processing. Unicode provides a unique numeric code (a code point) for every character, no matter what the platform, no matter what the program, no matter what the language. The standard was developed by the Unicode Consortium in 1999.
- Z39.50** : A NISO and ISO standard protocol that specifies a client/server-based protocol for cross-system searching and retrieving information from remote databases. It specifies procedures and structures for a client system to search a database provided by a server.
- Zebra** : A high performance open source text retrieval engine for indexing and retrieval, used by Koha as its primary search system for bibliographic and authority data.

3.12 REFERENCES AND FURTHER READING

Breeding, M. Chapter 7: Next-Generation Flavor in Integrated Online Catalogs. *Library technology reports*, 434 (2007), pp.38-41.

Breeding, M. The viability of open source ILS. *Bulletin of the American Society for Information Science and Technology*, 35.2(2009), pp. 20-25.

Breeding, Marshall . Perceptions 2007: an international survey of library automation. *Library Technology Guides*, January, (2008) <<http://www.librarytechnology.org/perceptions2007.pl>>

David, L. T. *Introduction to integrated library systems*. Bangkok: Information and Informatics Unit, UNESCO Bangkok, Thailand, 2001. Print

Digital Library Federation. *DLF ILS Discovery Internet Task Group (ILS-DI) Technical Recommendation* (2008). <www.diglib.org/architectures/ilssi/ILS_Discovery_1.1.pdf>

Hodgson, Cynthia. *The RFP writer's guide to standards for library systems*. Bethesda, Maryland: National Information Standards Organisation, 2002. <<http://www.niso.org>>

Hopkinson, A. Introduction to library standards and the players in the field. *Digitalia* (2006). <http://digitalia.sbn.it/upload/documenti/digitalia20062_HOPKINSON.pdf>

Kuali Open Library Environment (2013). <<http://www.kuali.org/ole>>

Mukhopadhyay, P. The progress of Library Management Software: an Indian scenario. *Vidyasagar University Journal of Library Science*. 6 (2001), pp.51-69.

Mukhopadhyay, P. Library automation packages - introduction – BLII 003, Block 1, Unit 1 of CICTAL course, IGNOU, 2005.

Mukhopadhyay, P. Library automation – software packages – MLII 104 (ICT applications – Part 1), MLIS, IGNOU, 2006.

Müller, T. How to choose a free and open source integrated library system. *OCLC Systems & Services*, 27.1 (2011), pp.57-78. <<http://eprints.rclis.org/15387/1/How%20to%20choose%20an%20open%20source%20ILS.pdf>>

Open Library Environment: The Open Library Environment Project Final Report (2009). <<http://oleproject.org/final-ole-project-report/>>

Rayward, W.B. A History of Computer Applications in Libraries: Prolegomena. *IEEE Annals of the History of Computing*, April-June (2002), pp. 4-15.

Singh, V. Why migrate to an open source ILS? Librarians with adoption experience share their reasons and experiences. *Libri*, 63.3 (2013), pp.206-219.

Wang, S. Integrated library system (ILS) challenges and opportunities: a survey of US academic libraries with migration projects. *The Journal of Academic Librarianship*, 35.3 (2009), pp. 207-220.

Yang, S. Q., & Hofmann, M. A. The next generation library catalog: A comparative study of the OPACs of Koha, Evergreen, and Voyager. *Information Technology and Libraries*, 29.3 (2013), pp.141-150.

UNIT 4 LIBRARY AUTOMATION: APPLICATIONS OF OPEN SOURCE SOFTWARE

Structure

- 4.0 Objectives
- 4.1 Introduction
- 4.2 Open Source Movement
 - 4.2.1 Open Source Software
 - 4.2.2 Open Source Software: Development Path
 - 4.2.3 Open Source Software vs. Commercial Software
- 4.3 Open Source Software: Philosophy, Principles and Licensing
 - 4.3.1 Philosophy of Open Source Software
 - 4.3.2 Principles of Open Source Software
 - 4.3.3 Licensing of Open Source Software
 - 4.3.4 Open Source and Open Standards
- 4.4 Open Source Software and Libraries
 - 4.4.1 Use of Open Source Software
 - 4.4.2 Prospects and Problems
 - 4.4.3 Use of Open Standards
- 4.5 Open Source Software in Libraries: System Level
 - 4.5.1 Open Source Operating System
 - 4.5.2 LAMP Architecture
 - 4.5.3 LAMP Components
- 4.6 Open Source Software in Libraries: Domain Level
 - 4.6.1 Automated Library System
 - 4.6.2 Digital Library System
 - 4.6.3 Cataloguing Tools
 - 4.6.4 Other Library Activity Tools
- 4.7 Towards Open Library System
- 4.8 Summary
- 4.9 Answers to Self Check Exercises
- 4.10 Keywords
- 4.11 References and Further Reading

4.0 OBJECTIVES

After going through this Unit, you will be able to:

- know what is open source movement and how is it improving computing infrastructure;
- understand differences between commercial and open source software;
- identify advantages of using open source software and open standards in library system; and
- understand the emerging concept of open library system.

4.1 INTRODUCTION

Present library services are software-centric. As per the availability and distribution policy, software products are divided into two groups – closed source commercial products and open source free to use products. Commercial software in the domain of library activities are available against huge license fees along with separate annual maintenance contracts, updating fees and many other hidden costs. As a result, adaptation of a commercial LMS in library (for example) is not one-time capital expenditure but it leads to considerable recurring expenditure on already strained library budget. Moreover, these commercial LMSs are basically available in a generic or fit-to-all size model and provide no scope for customisation to suite the need of a particular library (Mukhopadhyay, 2008). This is an alarming situation for libraries in India. Libraries are paying huge sum of money to procure commercial LMS but unfortunately not in a position to even change the colour of the user interface. Another serious lacuna is the non-transparent nature of these software in the use of global *de jury* or *de facto* standards.

Application of open source software in different library activities may be a viable alternative solution to get rid of the problems related with the application of commercial software. The tradition of open source software started with the advent of ARPANET (now Internet) in 1969 and boosted with the development of open source operating systems like GNU Linux. Naturally, one question is coming to your mind – what is open source software and how is it different. According to OSI (Open Source Initiative, 2003) – “*Open source promotes software reliability and quality by supporting independent peer review and rapid evaluation of source code. To be certified as open source, the license of a program must guarantee the right to read, redistribute, modify, and use it freely*”. Open source software are available freely to end users. Here the term *Free* has dual meaning – users are given *freedom* to customise the source code and these software are available *free of cost*. An open source software is attached with four freedoms – read (source code is available for verification), use (binary code is available for application), modify (source code is available for modification and customisation), redistribute (source code in original or in modified form is available for redistribution).

In the area of library services, the greatest benefit of open source software is the opportunity for library professionals to work at the system level and to participate in software development process as co-developers. Fortunately, the domain of library and information science, right from the beginning of the open source movement, is benefited through structured effort and software philanthropy. We have matured ILS like Koha (comparable to any global ILS) from HLT, New Zealand, comprehensive digital library software like DSpace from the MIT, US (with support from HP), Greenstone Digital Library Software (or GSDL) from University of Waikato (presently supported by UNESCO). Apart from these very popular open source software, the arena is presently fielded with an array of promising software like MARCEdit and ISISMARC (MARC cataloguing tools), WEBLIS (ILS based on CDS/ISIS), YAS toolkit (Z39.50 client and server), Lucene and Solr (Text retrieval engines), Unicode-compliant multilingual tools etc. Most of these open source software in the domain of LIS are very transparent in the use of standards and generally deploy open standards for achieving interoperability.

This brief introduction gives you idea on open source software and the possibilities for applications of open source software in enhancing library systems and services. Now we are all set to discuss open source software in depth. The discussion mainly cover six areas – 1) history, development, features and advantages of open source; 2) philosophy, principles and IPR issues related with open source; 3) use and advantages of open source software in libraries in general; 4) application of open source software in library activities at the system level; 5) application of open source software in library activities at the domain level; and 6) the emerging concept of open library systems that manages open contents and supported by open standards and open source software.

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

1) Enumerate the problems for application of commercial software in libraries.

.....
.....
.....
.....
.....

2) What do you mean by open source? Enumerate the freedoms associated with open source.

.....
.....
.....
.....
.....

3) List a few open source software in the domain of library services.

.....
.....
.....
.....
.....

4.2 OPEN SOURCE MOVEMENT

This section covers systematically the definition, scope and origin of open source software including the fundamental differences between open source and close source software.

4.2.1 Open Source Software

Open Source Software (OSS) is not a new idea. You already know that the open source movement started with the Internet. Recently, technical and market forces joined together to draw a niche role of open source movement. Open source movement has all the potentials to define computing infrastructure of the next century (Marco & Lister, 1987). Open source is a software development model as well as a software distribution model. OSS development follows Linus Torvalds's (Linus Torvalds is the developer of Linux operating system – an open source system software) style of development – *release early and often, delegate everything and be open to the point of promiscuity*. Raymond (2001a; 2001b) termed this type of software development as bazaar style of development in comparison with traditional software development process (termed by Raymond as cathedral model), which is carefully crafted by individual wizards or small group of experts working in splendid isolation. The Open Source Initiative (2004), a forum to promote open source software movement as a viable alternative to commercial software claims –

“This rapid evolutionary process produces better software than the traditional closed model, in which only a very few programmers can see the source and everybody else must blindly use an opaque block of bits.”

OSS is also considerably different from shareware, public-domain software, freeware, or software viewers and readers that are made freely available without access to source code. Shareware, whether or not one registers it and pays the registration fee, typically provides no access to the original source code. Unlike freeware and public domain software, OSS is copyrighted and distributed with license terms designed to ensure that the source code will always be available. Sometimes small amount of fee may be charged for the software's packaging, distribution, or support.

Definition

The open source movement has been in conscious development for nearly two decades but the term “open source” itself has been a relative latecomer. Christine Peterson of the Foresight Institute proposed the term *open source* in late 1997 during a meeting of small group of open source movement key persons (Raymond, 2001c). This group registered the domain name *opensource.org*, defined “open source,” developed Open Source Initiative (OSI) group, designed OSI certification, and created a list of licenses that meet the standards for open source certification. In the open source software development model the source code of software is made freely available along with the binary version so that anyone can see, change, and distribute it subject to the condition he/she abide by the accompanying license. According to OSI (Open Source Initiative, 2003a) –

“Open source promotes software reliability and quality by supporting independent peer review and rapid evaluation of source code. To be certified as open source, the license of a program must guarantee the right to read, redistribute, modify, and use it freely”.

Analysis of definitions given by Chudnov (1999), Raymond (1996), Moody (2001), and Morgan (2002), identifies following attributes of OSS –

- OSS is typically created and maintained by developers crossing institutional and national boundaries, collaborating by using Internet based communications and development tools;
- OSS development process follows the famous Linus's law – “Release early, release often and listen to users”;
- Quality, not profit, drives open source developers who take personal pride in seeing their working solutions adopted; and
- Intellectual property rights to open source software belong to anyone who helps to build it or simply use it and is not locked to any single vendor or institutions.

4.2.2 Open Source Software: Development Path

Computing community started realising the advantages of sharing of source codes in the late 1970s by using Internet as platform. Early 1980s witnessed a big conflict between OSS and proprietary software. For example, MIT Artificial Intelligence Lab established an agency called Symbolics in early 1980s and made all the freely available software proprietary under its name. This conversion process eventually killed the culture of code-sharing at MIT Lab. This destruction is important in the history of OSS because it initiated the free software movement through the formation of Free Software Foundation (FSF). Richard Stallman, one of the MIT lab members at the time, started The GNU (recursive acronym for GNU is Not Unix) project (a free operating system) in January 1984 and established FSF in 1985 to promote Free Software and the GNU project. The next big contribution in free software movement came from a student in 1991. Linus Torvalds, who at the time was a second year graduate student at the University of Helsinki, wrote a Unix-like kernel (Kernel is core part of operating system) and named it as Linux. He distributed Linux widely, considered users as co-developers and improved it considerably in a short span of time. Linux kernel soon adapted to become the core of the GNU/Linux operating system and many other parallel projects (like BIND, Perl etc.) merged with it. In 1997 GNU/Linux became the bussword in computing community because within 5 years it owned 25 per cent of the server market and growing at the rate of 25 per cent per annum. It's now clear that the code sharing and free software culture has been in conscious development for nearly three decades since the beginning of Internet. But the term “open source” has been a relative latecomer. Christine Peterson of the Foresight Institute proposed the term *open source* in late 1997 during a meeting of small group of open source movement key persons (Raymond, 2001a). This group registered the domain name *opensource.org*, defined “open source,” developed Open Source Initiative (OSI) group, designed OSI certification, and created a list of licenses that meet the standards for open source certification.

4.2.3 Open Source Software vs. Commercial Software

The whole array of software can be grouped into two fundamental categories – system software and application software. System software (such as operating system) is responsible for the overall management of computer resources whereas application software are designed to perform certain tasks and thereby make computers able to perform different predefined jobs. This division is based on the application domain of software. As per the distribution policy, software may be grouped into two broad divisions – close source software and open source

software (OSS). Open source software is also known as Free/Open Source Software (FOSS) or Free/Libre Open Source Software (FLOSS). Close source software may again be placed in two groups – commercial software and freeware. So, as per the distribution policy (as mentioned in the beginning), the whole array of software may be categorised into three groups – Commercial software, Freeware, and Open source software.

Table 4.1: Software as per the distribution policy

Commercial software	Freeware	Open source software
Only binary code is available against fees	Only binary code is available at no cost	Both source code and binary codes are available at no cost
As source code is not available, customisation is not possible	As source code is not available, customisation is not possible	As source code is available, extensive customisation is possible and allowed
License agreement allows only the use of software for a definite period and it is mandatory	License agreement allows to use for indefinite period and it is optional	License agreement allows to use, change, modify and distribution of software for indefinite period and it is mandatory

You can easily understand from table 4.1 that the fundamental difference is the opportunity for customisation. Open source also provides freedom to redistribute the customised version of the software.

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

4) Explain the term “open source”?

.....

.....

.....

.....

5) Write a brief history of open source movement.

.....

.....

.....

.....

6) Differentiate Close source, Freeware and Open source.

.....

.....

.....

.....

4.3 OPEN SOURCE SOFTWARE: PHILOSOPHY, PRINCIPLES AND LICENSING

You already know from the previous section what open source is and how it is different from other software distribution including a brief history of open source movement. In this section we are going to study philosophies and principles of open source software, IPR issues related with open source and application of open standards in open source software development.

4.3.1 Philosophy of Open Source Software

Open source software world is dominated by two major philosophies namely the Free Software Foundation (FSF) philosophy and the Open Source Initiative (OSI) philosophy. The philosophy of FSF centres around four user-driven freedoms –

- the freedom to run a program, for any lawful purpose;
- the freedom to study how a program works and adjust it to specific needs (obviously access to the source code is a precondition for this);
- the freedom to redistribute software; and
- the freedom to improve a program and distribute modified program (again access to the source code is a prerequisite for this).

Therefore, we may say that Freedom is at the core of FSF philosophy – the freedom to use, study and customise, the freedom to redistribute, the freedom to cooperate. FSF philosophy is against software patents and additional restrictions as included in existing copyright laws. On the other hand, the OSI philosophy is slightly different from FSF philosophy. The philosophy of OSI gives less emphasis on the ethical issues as proposed by FSF and is directed towards the practical rewards of the distributed development process of open source software. It targets on the technical values of participatory software development model for developing software, and is more business-friendly than the FSF. But there are many common issues in these two philosophies of open source software development such as efforts against proliferation of commercial software, software patenting and efforts in making software development process easy and user friendly. Richard Stallman, the father of FSF, rightly said that the Free Software Movement and the Open Source Movement are two political parties in the same community (Wong and Sayo, 2004).

4.3.2 Principles of Open Source Software

Development of open source software is governed by ten principles. OSI proposed a set of ten criteria (Open Source Initiative, 2006) for a software product to be called open source software. OSI provides *OSI Certified License* to a software product if it satisfies following ten criteria (popularly known as *Ten Commandments* of open source):

- **Free redistribution:** The license must allow end users to redistribute the software, even as part of a larger software package and may not charge royalties for this right.
- **Source code:** The distribution must make the source code freely available to developers.

- **Derived works:** The license must allow modifications and derived works and must allow them to be distributed under the same terms as the license of the original software.
- **Integrity of the author's source code:** The license may require that modified distributions be renamed, or that modifications be made via patch files rather than modifying the source code.
- **No discrimination against persons or groups:** The license must not discriminate against any person or group of persons.
- **No discrimination against fields of endeavour:** The license must not restrict anyone from making use of the program in a specific field of endeavour.
- **Distribution of license:** The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.
- **License must not be specific to a product:** A program may be extracted from a larger distribution and used under the same license.
- **The license must not restrict other software:** The license must not contaminate other software by placing restrictions on any software distributed along with the licensed software.
- **The license must be technology-neutral:** The license should not be framed on the basis of any individual technology or style of interface.

4.3.3 Licensing of Open Source Software

Licensing issues related with open source software are complex in nature. Open source software may be released under a variety of different licenses. Open Source Initiative (OSI) reported availability of more than 60 licenses and categorised these licenses under eight categories (<http://www.opensource.org/licenses/index.html>). However, an in-depth analysis shows that there are only two primary types of licenses and countless variants are based on these two widely adopted licenses. These two main licenses are the GNU (recursive acronym for GNU's not Unix) General Public License (GPL) and the BSD-style licenses.

The GNU General Public License (GPL)

The key features of GPL are – i) user freedoms is ensured and protected; ii) source code is always available; iii) users are allowed to copy, distribute and modify original code; iv) any changes made to a GPL program by the distributor must also be licensed under the GPL; v) distributors may not place any non-GPL restrictions upon the users; vi) recipients of GPL software are granted the same rights as the original distributor; and vii) a commercial software company cannot take a GPL program, modify it and then sell it under a different, proprietary license.

BSD-style Licenses

BSD-style (Berkeley System Distribution) licenses are identical to the original license issued by the University of California, Berkeley. These are among the most permissive licenses and include key features like – i) attribution is given to the original license holder by including the original copyright notice in source code files; ii) no attempt is made to sue or hold the original licensor liable for

damages; iii) software code available under BSD-style license can easily be incorporated into commercial applications; and iv) BSD-style licenses do not require the distribution of source code (after modification of original code). These two major licenses may be compared against the following features in the context of distributing open source software –

	GPL Licensed	BSD Licensed
Must distribute original source code	Yes	No
Must distribute user-created source code	Yes	No
User-created source code must be available under GPL	Yes	No
Proprietary Software linking possible	No	Yes
Compatible with GNU GPL	Yes	No*

*The original BSD license is not GPL compatible but the modified BSD license is compatible with GPL.

4.3.4 Open Source and Open Standards

Library services have long depended on shared standards. Recently, one question has been attracting our attention: whether a specific standard is of an open or a proprietary in nature. A proprietary standard is characterised by the fact that it is owned by someone (individual or organisation) who puts restrictions on - or can put restrictions on - users' access and use. On the other hand, a completely open standard has the following properties:

- It is accessible and free of charge to all (i.e. there is no inequity between users, and no payment or other considerations are required as a clause of use of the standard);
- It remains accessible (i.e. owners will not limit access to the standard later on i.e. afterwards); and
- All aspects of the standard are translucent, well documented, and freely available.

The W3C (2006) provides a set of six pack criteria in defining Open Standards:

- transparency (due process is public, and all technical discussions, meeting minutes, are archived and citable in decision making);
- relevance (new standardisation is started upon due analysis of the market needs, including requirements phase, e.g. accessibility, multilinguism);
- openness (anybody can participate, and everybody does: industry, individual, public, government bodies, academia, on a worldwide scale);
- impartiality and consensus (guaranteed fairness by the process and the neutral hosting of the W3C organisation, with equal weight for each participant);
- availability (free access to the standard text, both during development and at final stage, translations, and clear IPR rules for implementation, allowing open source development in the case of Web technologies); and
- maintenance (ongoing process for testing, errata, revision, permanent access).

Software development, as a process, depends on standards (de jury/de facto or proprietary/open) in each step. Open standards provide following advantages – 1) free to apply for any lawful purposes; 2) open and collaborative process of development; 3) well documented and no chance of data loss due to technical obsolescence. The visible disadvantages of open standards are – 1) availability of only a few major players (e.g. Loc, IFLA etc.); 2) lack of coordination between open standard initiatives and open source software developers; and 3) non-availability of open standards in many important facets of library activities (e.g. exchange of bibliographic and authority data). Some of the well known open standards that are in use in different library related software are – MARC 21 family of standards for resource description, MARC-XML as exchange format, OAI/PMH as metadata harvesting standard, SRU/SRW as standards for web based distributed searching etc.

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

7) What are Ten Commandments of open source software?

.....
.....
.....
.....
.....

8) Discuss the features of open standards.

.....
.....
.....
.....
.....

9) Comment on IPR issues related to FLOSS.

.....
.....
.....
.....
.....

4.4 OPEN SOURCE SOFTWARE AND LIBRARIES

Libraries and open source software are a natural fit on the basis of the philosophy and practices. The spirit of Five Laws of Library Science (as proposed by Ranganathan) and philosophy of Ten Commandments of Open Source Software

(as specified by OSI) are directed towards the open knowledge movement. Both promote learning and understanding through the dissemination of information. One of the Keystone Principles of Association of Research Libraries (2004) states, “Libraries will create interoperability in the systems they develop and create open source software for the access, dissemination, and management of information”.

4.4.1 Use of Open Source Software

Use of open source software in libraries is increasing all over the world. This trend you have also observed in section 1.7 of Unit 1. Daniel Chudnov, a professional evangelist in the area of OSS application in library services (1999) identified three factors – *fund, freedom and fraternity*, which are advancing the use of OSS in libraries:

- OSS licenses allow libraries to use budget in an optimum way. Budget on software can be reduced and that fund can be utilised in other areas that require more funds;
- OSS product is not locked into a single vendor or software developer. It means library can hire services from computer programmers for customising OSS; and
- Use of OSS can increase fraternity i.e. the entire library community might share the responsibility of solving information systems accessibility issues.

Digital Library Federation (2004) of USA considers and advocates use of OSS in libraries in its draft report on the basis of following reasons –

- OSS is an economical alternative to libraries’ reliance upon commercially supplied software. It means that the real costs involved in the development, maintenance, and use of OSS software are lower than those associated with commercial software (license, upgrading and maintenance fees);
- With OSS, the IT infrastructure for library operations and services can be:
 - Open, that is, built according to open standards and as such potentially interoperable with other software and systems;
 - Ubiquitously available to libraries and can be tailored to suit the needs and circumstances of individual libraries;
 - Documented (and documentation is accessible to all); and
 - Modified and corrected more effectively (“many eyeballs make bugs shallow”).

The above factors and advantages as identified by experts are responsible for increasing use of open source software in different libraries. Open source is a boon for libraries in developing countries like India. Now small libraries, which cannot afford costly ILS can opt for library automation with the availability of open source software.

4.4.2 Prospects and Problems

OSS democratises the use of software applications in libraries irrespective of the type or size of the library. OSS ensures that library systems and on-line services

will be more functional for patrons because libraries, through OSS movement –

- Are interested in experimenting new possibilities that results in new systems and software;
- Can take part in software development process and thereby have greater influence over the functional and performance requirements associated with particular software tools and systems;
- Can motivate and empower library staff to work at the system level; and
- Are able to collaborate more easily with experts of other similar domains engaged in common research and development activities.

The major advantages of open source software are –

- freedom to incorporate changes as required by an individual library;
- no vendor lock-in and freedom to hire technical expertise from outside; and
- better software development model (continuous upgrading, scope to contribute as co-developer and global professional fraternity).

The disadvantages associated with open source applications are – 1) steep learning curve; 2) non-availability of in-house technical expertise; 3) no on-call and on-site technical support.

Certainly OSS provides new opportunities in the development of library system and services in an economic way. But at this point it is too early to say that OSS is all set to replace proprietary software. In fact the issue is more whether OSS can provide a viable alternative and obviously there remain a number of obstacles to its wider adoption. First of all, OSS generally demands higher level of technical knowledge to install and maintain it. Users who migrate to open source applications face a steep learning curve and owing to this reason, the implementation of open source solutions today tends to be restricted to infrastructure and other “invisible” applications such as servers, where technical personnel are responsible for their installation and management. Obviously, open source offers new opportunities but also raises a number of challenges for the library and information community. Many library automation software vendors say (Poynder, 2001) that open source isn’t an easy option for libraries as it requires them to take more personal responsibility for their system and they have to carry the burden of development themselves, or to turn to a commercial vendor to customise the product to their needs. True, but one should not forget that OSS is not only a software development and delivery model, it is also a software solution model that helps users through discussion forums, FAQ (Frequently Asked Questions), online chats, manuals and developer’s guides. In this context we may quote Andy Powell (2002), assistant director of the U.K. Office for Library and Information Networking (UKOLN) – “*You might well need a higher level of technical understanding, but with good open source solutions help is often just an e-mail message away*”.

4.4.3 Use of Open Standards

You already know what an open standard is and how is it different from proprietary standards in sub-section 4.3.4 of this Unit. Software development, as a process, depends on standards (de jure/de facto or proprietary/open) in each step. Open standards provide following advantages – 1) free to apply for any lawful purposes;

2) open and collaborative process of development; 3) well documented and no chance of data loss due to technical obsolescence. The visible disadvantages of open standards are – 1) availability of only a few major players (e.g. Loc, IFLA etc.); 2) lack of coordination between open standard initiatives and open source software developers; and 3) non-availability of open standards in many important facets of library activities (e.g. exchange of bibliographic and authority data).

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

10) What is the view of Digital Library Federation (DLF) in the matter of using OSS in libraries?

.....
.....
.....
.....
.....

11) Point out advantages and disadvantages of open source applications in libraries.

.....
.....
.....
.....
.....

4.5 OPEN SOURCE SOFTWARE IN LIBRARIES: SYSTEM LEVEL

You already know the use and advantages of open source software applications in libraries. This section covers the essential system level open sources that are commonly in use in developing library software.

4.5.1 Open Source Operating System

The Linux kernel was initially conceived, created and uploaded in public domain by Finnish computer science student Linus Torvalds in 1991. The Linux kernel is released under the GNU General Public License version 2 (GPLv2) and is developed by contributors across the globe. Development activities like patches, software, debugging etc. takes place on the Linux kernel mailing list. Many Linux distributions (Redhat/Fedora, Ubuntu, Debian etc.) have been released based upon the Linux kernel. The development of X window system allows programmers to develop GUI based Linux distributions for end users. GNOME is the most popular deployment for different Linux distributions but other X Window programs (such as KDE) are gaining strength over the years. The kernel

of Linux performs different tasks through different layers. There are six major functions of Linux kernel – system, processing, memory, storage, networking and human interaction. The kernel is divided into six logical layers – hardware, device, functional, bridges, virtual and user. A Linux distribution (also referred as GNU/Linux distribution) is a member of the family of Unix-like OS (Unices) built on top of the Linux kernel. Such distributions (often called *distros* for short) include a large collection of software applications such as word processors, spreadsheets, media players and database applications. There are commercial agency initiated/backed free distributions, such as Fedora (Red Hat), openSUSE (Novell), Ubuntu (Canonical Ltd.), and Mandriva Linux (Mandriva) and community driven distributions such as Debian and Gentoo. Slackware is an example of corporate house-driven Linux distro.

4.5.2 LAMP Architecture

LAMP stands for **Linux-Apache-MySQL-PERL/PHP**. It refers to a combination of Linux (any distribution of Linux mentioned in previous section) as Operating System, Apache as Web Server, MySQL as Backend RDBMS and PERL or PHP as Programming Environment. Most of the open source software are based on LAMP architecture. LIS domain is no exception. The open source software we commonly use (generally application software) for designing and developing library systems and services are based on LAMP architecture. For example, Koha, E-Print Archive, Joomla, Emilda all are based on LAMP framework.

4.5.3 LAMP Components

Apart from Linux-based operations systems as mentioned above, the LAMP architecture includes Apache, MySQL, PERL and PHP. This section gives you a very brief introduction to each of these components.

Apache Web Server

- **Description:** The Apache httpd server is a powerful, popular and flexible Web server. It is compliant with HTTP/1.1 and available as open source under GPL. Apache is highly customisable and extensible. It can be customised by writing ‘modules’ using the available API. Although originated in Unix domain, Apache runs on almost every operating systems including Windows OS and different distributions of Linux.
- **Availability:** Available from <http://httpd.apache.org/> against GNU General Public License (GPL)
- **Dependencies:** None
- **Remark, if any:** Presently almost 90% of Internet host computers use Apache Web server.

MySQL Database Management System

- **Description:** MySQL, is possibly the most popular Open Source SQL database. It is created, distributed, maintained and supported by MySQL A B. It can handle large databases effectively and much faster than existing solutions. MySQL has been successfully used in production environments for last several years. The features like connectivity, speed, and security make MySQL Server highly suitable for accessing bibliographic databases on the Internet.

- **Availability:** Available from <http://www.mysql.com/> against GNU Public License (GPL).
- **Dependencies:** Generally requires no additional software but OpenSSL library is required to run secure connections.
- **Remark, if any:** MySQL is completely compatible with ANSI SQL standard. It has a large user base and is generally much faster than other RDBMSs. MySQL provides API (Application Programme Environment) to an array of programming languages.

PERL Programming Environment

- **Description:** PERL (Practical Extraction Report Language) was originally created to extract information from text files and then use that information to prepare reports. It is an open source scripting language, which means that the programmer does not have to compile and link a PERL script. Instead, a PERL interpreter executes the PERL script. It is widely used for CGI programming. It is originated in the UNIX community and has a strong user-base in UNIX community, but usage on Windows is on the rise.
- **Availability:** Available from <http://www.activestate.com/> against GNU Public License (GPL) and PERL modules are available from <http://www.cpan.org/>
- **Dependencies:** Generally requires no additional software but PERL modules necessary for running other software are available from CPAN archive.
- **Remark, if any:** ActivePerl is a quality-assured binary build of PERL, available for Windows, Linux and Solaris. It supports Unicode and large file operations on different platforms.

PHP Programming Environment

- **Description:** PHP is an open source server side scripting language. PHP is a parsed language. It means that there will be no compiled binaries. Every time a client browser requests a page with PHP code, the parser executes PHP-statements in the code.
- **Availability:** Available from <http://www.php.net/> against GNU Public License (GPL).
- **Dependencies:** Generally requires no additional software but Web server (e.g. Apache) is required to run PHP programmes in Web environment.
- **Remark, if any:** PHP supports many databases (MySQL, PostGreSQL and other commercial RDBMSs), generic ODBC's and almost all Web servers (Apache, IIS etc.) and it runs on different platforms (Unices, Windows, Solaris etc.).

Self Check Exercises

Note: i) Write your answers in the space given below.

ii) Check your answers with the answers given at the end of this Unit.

12) What is LAMP? Explain.

.....
.....
.....

13) What is MySQL? List some of the library software that are using MySQL.

.....

.....

.....

.....

.....

4.6 OPEN SOURCE SOFTWARE IN LIBRARIES: DOMAIN LEVEL

This section deals with the available open source application in different domains of library activities namely library automation, digital library system, cataloguing tools etc.

4.6.1 Automated Library System

Libraries are now operating in a distributed global networked environment. On the other hand, the volumes and varieties of user demands are increasing day-by-day. As a result, libraries reliance upon open standards and open source software is also increasing to satisfy growing multidimensional need of users and systems because open source software are adapting new technologies and architecture rapidly in compare with commercial software. Moreover, the age-old software development model followed by most of the commercial ILSs is not adequate for modern library activities. The serious lacunae of the commercial ILS model are –

- no scope to customise source code to incorporate new features;
- old and inefficient workflows built into the ILS for managing digital information resources;
- inability to integrate ILS with other institutional information systems like personnel database, course management system, institutional repositories, social networking facilities etc.

In short, we may safely say that proprietary ILS systems (acting as base for library automation and digitisation in many libraries) with firmly interlaced components make it difficult to respond to the ongoing changes (particularly the opportunities initiated by Web 2.0 technologies) and force library professionals to adjust library activities (mainly workflows and operations related with organisation and retrieval of digital information resources) to work within older systems. There are many open source software in the domain of library automation now but Koha appeared first in the year 1999. The list is given below –

- *ABCD**: ABCD is a fully integrated library automation system based on ISIS-technology as the underlying database with support for standards like MARC 21, UNIMARC, MODS and OAI. URL: <http://bvsmodelo.bvsalud.org/php/index.php>
- *Avanti*: Avanti is an open source ILS for small scale libraries with an emphasis on simplicity, usability and careful design (FLOSS based Dependencies:

Java Run Time Environment (JRE), Any Web server, PicoDB); URL: <http://www.avantibrarysystems.com/>

- *Emilda*: Emilda consists full featured Web-OPAC , template based layout , MARC compatibility & full customisation of the system with Emilda configurator. (FLOSS based Dependencies: Apache, PERL, MySQL, PHP, Sebra server, YAS toolkit); URL: <http://www.emilda.org/>
- *Evergreen*: The open source software Evergreen is stable, robust, **flexible**, secure and user-friendly to their patrons. URL: <http://www.open-ils.org/>
- *FireFly*: FireFly facilitates public libraries a Free-Software set to run and maintain library systems. (FLOSS based Dependencies: Any Web server, Any SQL, Python & PHP); URL: <http://savannah.nongnu.org/projects/firefly/>
- *GNU Library Management System (GLIBMS)*: A Library can be automated its various activities through GNU Library management System. (FLOSS based Dependencies: Any Web server, PostGreSQL, PHP, PERL); URL: <http://sourceforge.net/projects/glibs/>
- *GNUTeca*: (FLOSS based Dependencies: Apache, PostGreSQL PHP); URL: <http://www.solis.org.br/index.php/projetos/gnuteca>
- *Koha*: Koha is fully-featured ILS with **Dual Database Design** , interoperable with **Library Standards and protocols having Web-based Interfaces without vendor lock-in** (FLOSS based Dependencies: Apache, MySQL, PERL); URL: <http://www.koha.org/>
- *LearningAccess ILS*: The LearningAccess ILS is a standards-based, fully integrated, flexible, Open and powerful system. It provides smaller libraries access to state-of-the-art library automation in affordable pricing (FLOSS based Dependencies: Apache, MySQL, PHP, YAS); URL: <http://www.learningaccess.org/tools/ils.php>
- *NewGenLib*: NewGenLib is a scalable, manageable and efficient open source software with federated search facilities and RFID integration (FLOSS based Dependencies: JBoss, Java SDK, PostGreSQL Ant; URL: <http://www.verussolutions.bis/>
- *OpenBiblio*: In OpenBiblio library system one can edit almost everything i.e., wiki like interface (FLOSS based Dependencies: Apache/Any Web server, MySQL, PHP); URL: <http://obiblio.sourceforge.net/>
- *PHPMyBibli*: PhpMyBibli is a web-based library automation for French libraries. (FLOSS based Dependencies: Any Web server, MySQL, PHP); URL: <http://phpmybibli.sourceforge.net/>
- *PHPMyLibrary*: (FLOSS based Dependencies: Apache, MySQL, PHP); URL: <http://phpmylibrary.sourceforge.net/>
- *PYTHEAS*: PYTHEAS as Library Application Framework providing server-based metadata (MARC) and information retrieval capabilities (RDF). (FLOSS based Dependencies: JDK version 1.4 and above, MySQL, Apache-Tomcat Web server); URL: <http://seus.uwindsor.ca/library/leddy/people/art/pytheas/index.html>

- **WEBLIS***: (FLOSS based Dependencies: CDS/ISIS, Any Web server, ISIS.DLL); URL: <http://www.unesco.org/isis/files/weblis.sip>

(* ABCD and WEBLIS are based on CDS/ISIS which is a close source textual DBMS developed by UNESCO and available free of cost)

Most of the LMSs listed above are in their infancy. The mature LMS block includes Koha, Emilda, Evergreen, NewGenLib, WEBLIS and PHPMyLibrary. Koha, the first open source library management software, has created a high level of interest in library profession for open source movement internationally. Koha (in Maori language Koha means an unconditional gift) is a full-featured open-source ILS. Developed initially in New Zealand by Katipo Communications Ltd and first deployed in January of 2000 for Horowhenua Library Trust, Koha is currently maintained by a team of software developers and library technology staff from around the globe.

4.6.2 Digital Library System

Some of the well known open source digital library software are –

- **Dspace**: Dspace is a popular OAI/PMH compatible institutional repository software (FLOSS based Dependencies: Jakarta-Tomcat, PostGreSQL, Java SDK, Apache Ant); URL: <http://www.dspace.org/>
- **E-print Archive**: It is a platform which builds repositories of research literature, scientific data, student theses, project reports, multimedia artefacts, teaching materials, scholarly collections, digitised records, exhibitions and performances. (FLOSS based Dependencies: Apache, MySQL, PERL and PERL modules); URL: <http://www.eprints.org/>
- **Fedora**: Fedora facilitates management, preservation or linking of any digital contents. (FLOSS based Dependencies: Java SDK, Jakarta-Tomcat, Any RDBMS (MySQL, Oracle, McKoi); URL: <http://www.fedora.info/>
- **Greenstone Digital Library Software**: The digital library software Greenstone organise information and publish on the Internet or on CD-ROM.(FLOSS based Dependencies: Apache, PERL and Java Runtime Environment, ImageMagik); URL: <http://greenstone.org/>

4.6.3 Cataloguing Tools

- **ISISMARC**: ISISMarc is a MARC 21-enabled multi-lingual and independent data entry interface which supports record validation through CDS/ISIS format and cross-data base copy/paste of records. (FLOSS based Dependencies: WINISIS DBMS, YAS DLL file); URL: <http://portal.unesco.org/>
- **MarcEdit**: A comprehensive and user-friendly utility suite for MARC records (FLOSS based Dependencies: YAS toolkit); URL: <http://oregonstate.edu/~reset/marcedit/html/>
- **MARC Template Library**: The MARC Template Library is collection of source code libraries and software for reading, writing and processing of MARC records (FLOSS based Dependencies: GCC); URL: <http://mtl.sourceforge.net/>

- *MARC/PERL*: MARC/Perl is for reading, manipulating, outputting and converting bibliographic records in the MARC format (FLOSS based Dependencies: PERL); URL: <http://marcpm.sourceforge.net/>
- *MARC2OPAC* (FLOSS based Dependencies: Apache, PHP, Grep); URL: <http://www.bundaberg.qld.gov.au/library/catalog/about.php4>
- *YAS Toolkit*: YAS toolkit implements Z39.50 standard and protocol to both the origin and target (FLOSS based Dependencies: None); URL: <http://www.indexdata.dk/yas/>
- *Scontent*: S Content is a perl based module facilitates a Z39.50 target (FLOSS based Dependencies: Perl, YAS Toolkit, SimpleServer); URL: <http://www.lib.utah.edu/portal/site/marriottlibrary/>

4.6.4 Other Library Activity Tools

The other useful open source software for different library activities are –

OAI/PMH Tools

- *ARC* (FLOSS based Dependencies: Java Servlet Engine, Tomcat, RDBMS (Oracle/MySQL); RL: <http://physnet.uni-oldenburg.de/oai/>
- *OAI Harvester* (FLOSS based Dependencies: Java, Apache Ant); URL: <http://www.oclc.org/research/software/oai/harvester.shtm>
- *OAICat* (FLOSS based Dependencies: Java Servlet Engine, RDBMS (tested with MySQL); URL: <http://www.oclc.org/research/software/oai/cat.shtm>
- *PKP Harvester* (FLOSS based Dependencies: PHP, Apache, MySQL); URL: <http://www.pkp.ubc.ca>

Inter Library Loan

- *ILL Wizard*: ISO compliant ILL can run from desktop or from the library website server directory. (<http://library.olivet.edu/iso-ill.html>)
- *Biblio::ILL::ISO - ISO-protocol-based Interlibrary Loan*: Biblio::ILL::ISO - ISO-protocol-based Interlibrary Loan is a perl language based ILL (<http://maplin.gov.mb.ca/pub/TEST/>)

Subject Gateways

- *ROADS*: Resource Organisation And Discovery in Subject-based Services (<http://roads.sourceforge.net/>)
- *IMesh Toolkit*: Imesh Toolkit is a set of tools and standards used by subject gateway software developers. (<http://clark.cs.wisc.edu/cgi-bin/cvsweb.cgi>)

Text Retrieval Tools

- *HTDig*: HTDig is indexing and searching system for public domain resources (<http://www.htdig.org/>)
- *SWISH-E*: SWISH-E is fast, flexible, and open source system for indexing collections of Web pages or other files. (<http://swish-e.org/>)
- *ASPSeek*: ASPSeek is an Internet search engine software consists of an indexing robot, a search daemon, and a CGI search frontend (<http://www.aspseek.org/>)

- *Harvest*: Harvest system collecting information and make them searchable using a web interface (<http://harvest.sourceforge.net/>)
- *Sebra Server*: Sebra is a high-performance, general-purpose structured text indexing and retrieval engine. (<http://indexdata.dk/sebra/>)
- *Site Search*: Site Search facilitates some tools to integrate electronic resources under web and make them flexible. (<http://www.sitesearch.oclc.org/>)

Self Check Exercises

Note: i) Write your answers in the space given below.
ii) Check your answers with the answers given at the end of this Unit.

14) “Library automation is gradually taking the open way”. Elucidate.

.....
.....
.....
.....
.....

15) Comment on any three open source ILSs.

.....
.....
.....
.....
.....

16) Mention name of any two open source digital library software.

.....
.....
.....
.....
.....

4.7 TOWARDS OPEN LIBRARY SYSTEM

Open library system or popularly called the O₃ library is gaining strength from three well-coordinated movements namely open access, open source and open standards. The OLE recommendations also promoted the concept of open library system. The open library system is based on four pillars – i) open and distributed information system (the Internet); ii) open contents; iii) open standards and iv) open source software. Libraries all over the world are entering into the next wave of development to meet volume and variety of users’ information demands. The O₃ library targets to develop a world-wide retrieval system for open

knowledge objects. It has three strands – open contents (as library resources), open source software (as tools for building mechanisms for resource organisation and resource dissemination), and open standards (as means to achieve interoperability). Open knowledge movement is considered as an alternative path to fight against exorbitant price rise in commercial publication systems. The scholarly world and library professionals are developing forums (e.g. SPARC - Scholarly Publishing and Academic Resources Coalition) to promote the philosophy that *publicly funded research should be available in public domain* (see <http://www.arl.org/sparc/>). On the other hand, large-scale research repositories require mechanisms and systems for resource organisation, maintenance and dissemination. As opined by E. M. Corrado “*Open source software can benefit libraries by lowering initial and ongoing costs, eliminating vendor lock-in, and allowing for greater flexibility*” (see <http://www.istl.org/05-spring/article2.html>). No physical libraries are self-sufficient. Similarly no digital retrieval systems can hold all information resources in one place. Open standards can be of great help to achieve interoperability between different library resources and to solve the problems of data migration between systems. In view of these facts and discussions, we can predict the increasing importance of open source software and open standards in designing future library systems. The OLE report rightly suggested a set of six characteristics for software framework of future library systems –

- **Flexibility:** Accommodating wide range of resources accessed by users globally for different purposes;
- **Community ownership:** Library software frameworks are designed, developed, owned, and maintained by and for the library community on the basis of open source license;
- **Service oriented architecture:** Technology-neutral service-oriented frameworks to assure interoperability of library systems;
- **Enterprise-level integration:** Provision of integration with other enterprise systems such as research support systems, student information systems, human resources, identity management, fiscal control, institutional repository and content management;
- **Efficiency:** Suitability for modular application infrastructure that integrates with new and existing academic and research technologies; and
- **Sustainability:** Creates reliable and robust frameworks to identify record, innovate, develop, maintain, and review the software necessary to further the operation and mission of libraries.

4.8 SUMMARY

This Unit covered what and why of open source software in general. It also discusses history of open source movement including philosophy, principles and licensing of open source software. Most of the library experts are in opinion that open source software has all the potential to change the way libraries deal with the software. Library automation process is greatly influenced by the applications of open source software and open standards. OSS can provide a viable alternative to commercial ILSs. This unit examined the use of open source software in libraries at two different levels – system level and task level. In system level

LAMP architecture is prevailing in many libraries. At the task level libraries are fortunate to have open source ILSs, open source digital library software, open source cataloguing tools and many more. This unit also discusses problems of open source software in general and issues related with the use of open standards in developing OSS. Finally, it predicts the emergence of open library systems with three interrelated components – open source, open standards and open contents.

4.9 ANSWERS TO SELF CHECK EXERCISES

- 1) Commercial ILSs have following problems for applications – 1) Huge license fees; 2) Recurring payment cycle in the name of annual maintenance contract; 3) No scope for customisation to suite needs for individual library; 4) Non-transparent use of standards in the domain of library services; and 5) Delay in adaptation of new technologies.

- 2) In simple words open source software means a software development and distribution model (often referred as Basar style of software development) where software are available with source code to support extensive customisation and to provide four freedoms (in place of restrictions imposed by commercial close source software). Generally open source software are available free of cost. A typical open source software is attached with four freedoms – read (source code is available for verification), use (binary code is available for application), modify (source code is available for modification and customisation), redistribute (source code in original or in modified form is available for redistribution).

- 3) Library professionals took a great interest in open source movement, possibly because of the fact that the movement is promoting the concept of access to knowledge for all. As a result the domain of LIS is benefited by the movement with lots of open source software for different library activities such as ILS (Koha, Emilda, Evergreen, NewGenLib); Digital library (Dspace, Greenstone, Eprint archive); Cataloguing editor and protocols (MARCEdit, Yas Toolkit); Library portal (MyLibrary, Joomla, Drupal) and many more.

- 4) The culture of open source software started with the Internet in 1969 in the name of shareware or free software. The movement gained momentum with the establishment of Free Software Foundation (FSF) by Richard Stallman in 1985. But the term *open source* itself has been a relative latecomer. Christine Peterson of the Foresight Institute proposed the term *open source* in late 1997. Open source software are fundamentally different from shareware, public-domain software, freeware that are made freely available without access to source code.

- 5) The history of open source movement includes a series of groundbreaking events, contribution from individual as well as groups, encouragement from philanthropists and thinkers and support from different national governments and inter-governmental agencies like UNESCO. The distributed network platform i.e. Internet helped in growth of open source software by performing as platform for distribution of programs developed within the academic community. The sharing of source codes was a prevalent culture in universities and research laboratories during 1969 to 1982. This code sharing

culture of 1970s is considered as origin of open source movement. But this code sharing culture got a setback when MIT Artificial Intelligence Lab agency known as Symbolics made all shareware as proprietary software in the name of the agency. This unfortunate event led the development of GNU project and foundation of Free Software Foundation during 1984-85 by Richard Stallman (one of the member of MIT Lab). Next important event was the release of Unix-like kernel (named as Linux) by Linus Torvalds in 1991. Linux kernel played a significant role in developing open source software infrastructure. Lots of Linux-based open source operating systems were released over the last twenty years. The open source architecture LAMP (Linux as operating system, Apache as web server, MySQL as RDBMS and PERL, PHP as programming environment) acted as framework for developing open source software for different human activities including library services. The major events during 1997-2001 were - formation of open source group, registration of the domain name *opensource.org*, establishment of Open Source Initiative (OSI) group, design of OSI certification, and creation of a list of licenses that meet the standards for open source certification.

- 6) As per the distribution policy, the whole array of software may be categorised into three groups – Commercial software, Freeware, and Open source software. In case of commercial software only binary code (or executable code) is available against fees. Whereas freeware are available at no cost with binary code. In both of these cases source codes are not available with software and therefore customisation activities are not possible. But open source software includes both source code and binary codes at no cost. It supports modification of source code and distribution of source code against license.
- 7) Open Source Initiative (OSI) set aside ten criteria in 2006 for a software product to be called open source software. These ten criteria are popularly known as Ten Commandments of open source. These are – 1) Free redistribution of software; 2) Availability of Source code; 3) Derived works also available as open source; 4) Integrity of the author's source code; 5) No discrimination against persons or groups; 6) No discrimination against fields of endeavor; 7) Distribution of license; 8) License must not be specific to a product; 9) The license must not restrict other software; and 10) The license must be technology-neutral.
- 8) A proprietary standard is characterised by the fact that it is owned by someone (individual or organisation), who puts restrictions on - or can put restrictions on - users' access and use. On the other hand, a completely open standard is accessible at free of charge to all. It remains accessible and all aspects of the standard are translucent, well documented, and freely available. In library domain most of the global standards are open in nature such as MARC 21 family of standards, SRU/SRW, ISBDs and many more.
- 9) Open source software are available with attached licenses. The licenses provide freedom to study, customise and redistribute open source software. Licensing issues related with open source software are complex in nature. Open source software are released under a variety of different licenses. Study shows that there are more than 60 licenses. These licenses are grouped under

eight categories by OSI. However, an in-depth analysis shows that there are only two primary types of licenses and countless variants are based on these two widely adopted licenses. These two main licenses are the GNU (recursive acronym for GNU's not Unix) General Public License (GPL) and the BSD-style licenses.

- 10) Digital Library Federation in US is a platform for libraries from different places for developing principles and policies for automated and digital library systems. This forum published a draft report in 2004 and suggested use of OSS in libraries on the basis of following reasons – 1) OSS is an economical alternative to libraries' reliance upon commercially supplied software. Libraries can save fund require for license, upgrading and maintenance fees; 2) OSS ensures development of open and interoperable library systems by using open standards; 3) OSS allows extensive customisation and thereby can be tailored to suit the needs and circumstances of individual libraries; 4) OSS source codes, program logics, software architecture, data structure are well documented and documentation is accessible to all); 5) OSS can be modified and corrected more effectively because of large-scale participations by library professionals.
- 11) The major advantages of open source software are – 1) freedom to incorporate changes as required by an individual library; 2) no vendor lock-in and freedom to hire technical expertise from outside; and 3) better software development model (continuous upgrading, scope to contribute as co-developer and global professional fraternity). The disadvantages associated with open source applications are – 1) steep learning curve; 2) non-availability of in-house technical expertise; 3) no on-call and on-site technical support.
- 12) LAMP stands for Linux-Apache-MySQL-PERL/PHP. It means an open source based software architecture for development of web-enable open-source application software. In this software framework Linux kernel based operating systems such as Fedora, CentOS, Ubuntu etc are acting as platforms. Apache web server and MySQL relational database management systems are two major components of the framework. The programming languages like PERL, PHP etc are used for developing source codes for application programs.
- 13) MySQL is an open source relational database management system (RDBMS). It is a major component of LAMP architecture. MySQL is a popular RDBMS in the open source domain. The features like connectivity, speed, and security make MySQL Server highly suitable for accessing bibliographic databases on the Internet.
- 14) Libraries all over the world are passing through a rapid phase of development. Sometimes technologies demand fundamental changes in library operations and services. Moreover, libraries are now operating in a distributed global networked environment. It's no more possible for a library to serve in stand-alone mode. On the other hand, the volumes and varieties of user demands are increasing day-by-day. As a result, libraries reliance upon open standards and open source software is also increasing to satisfy growing multidimensional need of users and systems because open source software

are adapting new technologies and architecture rapidly in compare with commercial software. Moreover, the age-old software development model followed by most of the commercial ILSs is not adequate for modern library activities. As a result library automation and digitisation programs are increasingly using open source software for different library activities.

- 15) There are many open source software for different library activities. This is another facility in the open source domain, one particular area of activity includes many open source software. For example, the domain of library automation includes a total of 14 open source software. Koha is web-enabled open source ILS based on LAMP architecture meant for library automation activities. Evergreen is client-server architecture based open source ILS meant for automation of a group of libraries and useful for developing union catalogues in a library network setup. Another major open source ILS is NewgenLib developed in India. It uses open source companion software like PostGreSQL as RDBMS, Apache-Tomcat as java servlet engine and Java SDK as programming environment.
- 16) In the open source domain, like open source ILSs, there are many open source digital media archiving software. This domain of open source digital library software can be categorised into two basic groups – 1) Centralised processing – Distributed access architecture; and 2) Distributed process and distributed access architecture. In the first group, the most comprehensive one is Greenstone Digital Library Software and Dspace is the most popular software in the second group. Greenstone is written in PERL programming language and supports archiving many digital formats. Dspace is using PostGreSQL RDBMS, Apache-Tomcat and Java SDK.

4.10 KEYWORDS

- API** : Application Programming Interface. A language and message format used by an application program to communicate with the operating system or some other control program such as a database management system (DBMS).
- Discovery application** : A computer application designed to simplify, assist and expedite the process of finding information resources.
- DNS** : Domain Name Server, a service that resolves symbolic host names into numeric IP addresses, and vice versa.
- Encoding** : A character encoding scheme is a set of rules for representing a sequence of character codes with byte sequence.
- ERMS** : Electronic Resources Management System is used to manage a library's electronic resources, primarily e-journals and databases. Systems can include features to track trials, license terms and conditions, usage, cost, and access.

- FOSS** : Free/Open Source Software.
- GNOME** : GNU Network Object Modeling Environment, a desktop environment based on GTK+ toolkit and other desktop components.
- GNU** : A recursive acronym standing for “GNU’s system based on Unix architecture.
- I18N** : Abbreviation for Internationalisation.
- IIMF** : Internet/Intranet Input Method Framework, a new framework for cross-platform input method developed by OpenI18N.org. IIMF bridges different IM protocols by using wrappers that communicate with a common protocol.
- Interoperability** : The ability for two different computer systems to communicate and exchange information in a useful and meaningful manner.
- Kernel** : A very low-level software that manages computer hardware, multi-tasks the many programs that are running at any given time, and other such essential things.
- L10N** : Abbreviation for Localisation.
- Localisation** : Implementation of cultural conventions defined by the internationalisation process according to different languages and cultures.
- Metadata harvesting** : A technique for extraction of metadata from individual repositories for collection into a central catalog.
- Multilingual** : Supporting more than one language simultaneously. Often implies the ability to handle more than one script and character set.
- Open Source** : A concept through which programming code is made available through a license that supports the users freely copying the code, making changes it, and sharing the results. Changes are typically submitted to a group managing the open source product for possible incorporation into the official version. Development and support is handled cooperatively by a group of distributed programmers, usually on a volunteer basis.
- OpenSearch** : A collection of technologies developed by Amason that allow publishing of search results in a format suitable for syndication and aggregation.
- OpenURL** : A URL with stored metadata that is user context sensitive in what information or hypertext link is delivered.

- Pango** : A Unicode-based multi-lingual text rendering engine used by GTK+ 2. Like GTK+, Pango is written in C and licensed under LGPL.
- PHP** : A server-side scripting language for creating dynamic web pages.
- POSIX** : Portable Operating System Interface Specification is the minimum specification of system calls for operating systems based on Unix, defined by IEEE so that applications based on it are guaranteed to be portable across OSs. Although based on Unix, POSIX is also supported by some non-Unix OSs.
- Protocol** : A standard procedure for the message formats and rules that two computer systems must follow to communicate with each other
- RSS** : Really Simple Syndication is an XML format used for distribution or syndication of frequently updated Web contents.
- Script** : A system of characters used to write one or several languages.
- SSH** : Secure Shell is used for remote login using an encrypted connection to prevent sniffing by third parties.
- System Analysis** : A powerful technique for the analysis of an organisation and its work.
- UCS** : Universal Multi-octet coded character set, as defined by ISO/IEC 10646 to represent the world's writing systems. It is maintained by ISO/IEC JTC1/SC2/WG2, with contributions from the Unicode Consortium.
- Unicode** : A universal character-encoding standard used for representation of text for computer processing. Unicode provides a unique numeric code (a code point) for every character, no matter what the platform, no matter what the program, no matter what the language. The standard was developed by the Unicode Consortium in 1999.
- UTF-8** : Unicode (UCS) Transformation Format, using 8-bit multibyte encoding scheme.
- X Window** : A graphical environment initially developed by the Athena project at MIT with support from some vendors, and later maintained by the X consortium. X Window is the major graphical environment for most Unix variants nowadays.
- XML** : EXtensible Markup Language is an open standard for describing data from the World Wide Web Consortium. It is used for defining data elements on a Web page, business-to business documents, and other hierarchically structured text and data.

4.11 REFERENCES AND FURTHER READING

A Brief History of Free/Open Source Software Movement <<http://www.openknowledge.org/writing/open-source/scb/brief-opensource-history.html>>

Chudnov, Daniel. *Open source library systems: Getting started* (1999). <www.oss4lib.org/readings/oss4lib-gettingstarted.php>

Digital Library Federation. *The future is open: Digital libraries through open source software* (2004). <<http://www.dlf.org/Dlinitiatives/archiv/open.htm>>

Marco, D., and Lister, S. *Peopleware: Productive projects and teams*. New York: Dorset House Publishing, 1987. Print

Moody, T. *Open source and libraries: A natural fit* (2001). <<http://www.oss4lib.org/readings/moody.htm>>

Morgan, E. L. *Open Source Software in Libraries* (2002). <<http://dewey.library.nd.edu/morgan/musings/ossnlibraries.php>>

Mukhopadhyay, P. Progress of library management software: an Indian scenario. *Vidyasagar University Journal of Library and Information Science*, 6 (2001), pp. 51-69.

Mukhopadhyay, P. Comparative study of library management software. *Automation and networking of the college libraries*. Kolkata: Moulana Asad College, 2005. Print

Mukhopadhyay, P. Five laws and ten commandments: the open road of library automation in India. *Proceedings of the National Seminar on Open Source Movement – Asian Perspective, XXII*, Roorkee, 2006. Kolkata: IASLIC, 2006. pp. 27-36. Print

Mukhopadhyay, P. *Library automation through Koha*. Kolkata: Prova Prakashani, 2008. Print

Open Source Initiative. *Open source software certification process* (2003). <<http://www.opensource.org/osslicense.htm>>

Open Source Initiative. *OSI certified license: The ten basic criteria* (2003). <<http://www.opensource.org/tencom.htm>>

Open Source Initiative. *Open source software and future of computing* (2004). <<http://www.opensource.org/future.htm>>

Powell, A. *Open source movement: News and views* (2002). <<http://www.ukoln.ac.uk/powell.htm>>

Raymond, E. S. *The new hacker's dictionary*. Cambridge: MIT Press, 1996. Print

Raymond, E. S. *A brief history of hackerdom* (2001). <<http://tuxedo.org/~esr/writings/cathedral-basaar/hacker-history>>

Raymond, E. S. *Homesteading the noosphere* (2001). <http://tuxedo.org/~esr/writings/cathedral-bazaar/hacker-history.htm>

**Library Automation:
Application of Open Source
Software**

Raymond, E. S. *The cathedral and the bazaar: Musings on Linux and open source by an accidental revolutionary* .(Rev. ed). Cambridge: O'reilly and Associates, 2001. Print

Wong, K. and Sayo, P. *FOSS: a general introduction*. Kuala Lumpur, Malaysia: UNDP-APDIP, 2004. < http://www.iosn.net/downloads/foss_primer_current.pdf>

