DESIDOC Journal of Library & Information Technology, Vol. 32, No. 2, March 2012, pp. 101-108

eCataloguer: An Automatic Tool for Developing Bibliographic Database using MARCXML Records

K. Nageswara Rao* and A.L. Moorthy**

**Defence Research & Development Laboratory, Kanchanbagh PO, Hyderabad-500 058*
***Defence Scientific Information & Documentation Centre, Delhi-100 054*
*E-mail: *nageswararao@drdl.drdo.in*

ABSTRACT

The main access point for any library's collections is the online public access catalogue (OPAC) which is the primary source of information in the library. In general, libraries use gateway approach for providing access to e-resources such as e-books. To effectively increase the usage of e-books, it is necessary to develop and maintain library catalogue and provide searchable for e-books full-text collection. eCataloguer– is an automatic tool used for creating library bibliographic database, designed and developed using Java. eCataloguer has a web interface by which library professional can seamlessly and quickly develop bibliographic database of library documents using ISBN number. This tool locates the MARCXML record of the document by matching with ISBN number at the Library of Congress, reads the MARC tags and data values and presents to the user in an interface that develops local database of library documents. In supervised mode of database development, this tool quickly locates and presents bibliographic data to the user for making corrections before submitting to the local database. It also facilitates for the creation of database in a batch mode. eCataloguer helps the library professionals to develop bibliographic database in a multi-user and multi-platform environment as it has been developed using Java technology not only for printed books but also for electronic books.

Keywords: eCataloguer, bibliographic databases, e-books, MARCXML

1. INTRODUCTION

Proliferation of information technology (IT) in the field of library and information science (LIS), printed catalogue cards containing bibliographical details have taken new birth as bibliographic databases. These bibliographic databases, being accessible format in the computer, have played a vital role in searching, browsing, and locating for relevant documents in a massive library collection. Most widely adopted standard for the construction of catalogues and other lists in the libraries is Anglo-American cataloguing rules (AACR2) published jointly by the American Library Association (ALA), the Canadian Library Association, and Chartered Institute of Library and Information Processionals (CILIP) in the UK which provides rules for the description of, and the provision of access points for all library materials.

Internet technologies have created new possibilities and opportunities for librarians to create standard local bibliographic databases of library documents. The most efficient and effective way to create and maintain bibliographic database of library documents is to adopt international bibliographic standards and formats such as common communication format (CCF), Canadian machine readable cataloguing (CANMARC), UK machine readable cataloguing (UKMARC), US machine readable cataloguing (USMARC), Universal MARC format (UNIMARC), AACR2, international standard bibliographic description (ISBD) or MARC21. The fastest approach of developing bibliographic database is to rely on resource files of machine-readable records on internet and CD-ROM products such as the Library of Congress's *BiblioFile*[1], and the National Library of Canada's *Amicus*. Vendors like Yankee Book Peddle and Blackwell North America played an important role in selling books along with the MARC records. This method of developing bibliographic database is called copy cataloguing. To copy the cataloguing details, the resource file is searched for a particular document using international standard book number (ISBN), library of congress control number (LCCN), title or any other field. The complete bibliographic data of a document is retrieved and stored in a local database, when exact match is found. If an exact match is not found, an alternate option needs to be selected to

Received 26 September 2011, revised 14 November 2011, accepted 05 December 2011, online published 17 February 2012

minimise the keying of complete data. In such cases, a derived catalogue record is created by selecting and editing near-matching record that matches the document in hand. When matching record is not found, the last option is to create library database by keying in complete details of the document. Copy and derived cataloguing requires some sort of editing data pertaining to call number, subject headings, etc.

Several problems can arise due to copy or derived cataloguing these include: Errors in MARC coding; problems with punctuation; typographical errors; insufficient call numbers; discrepancies that cause serious problems (i.e., authority control issues); and the separation/integration of series or conferences[2].

2. IMPORTANCE OF BIBLIOGRAPHIC STANDARDS

Bibliographic standards are aimed at achieving consistency and uniformity of practice in the creation of bibliographic records. The full outcome of technological advances in computers, communications, and allied fields will not be realised until the data processing and management communities arrive at uniform understanding about the common information elements and their expression or representation in data systems[3]. So, uniform understanding by the computer systems can only be achieved through standardisation.

In libraries, standards are applied to achieve bibliographic control there by identifying two different documents without any confusion and details of a given document can be accessed by any data element. Adoption of standards facilitate better resource sharing between the libraries through maintaining union databases and also reduce the cost of cataloguing by minimising duplication of cataloguing efforts. Reliability and interoperability is only possible if catalogue records are created using consistent application of standards by different libraries. Standards facilitate in exchanging of bibliographic records without any loss of data which has been achieved through the implementation of MARC standard for library documents. MARC is a standard for the representation and communication of bibliographic and related information in machine-readable form, and related documentation. Implementation of MARC standard enables libraries to effectively make use of commercially available library automation systems to manage in-house library operations.

The MARC record structure is an implementation of the exchange format for bibliographic records specified in ANSI/NISO Z39.2 and ISO 2709:1996 standards. As early as 2002, the Library of Congress created the MARCXML (Extensible Markup Language) standard to communicate the same MARC data element set in XML syntax and MARCXML record structure is based on the W3C XML standard. A MARC record consists of three components: the Leader, the Directory, and the Variable fields. The Leader contains data elements that provide information for the automation system processing of the record. The data elements contain numbers or coded values and are identified by relative character position. The Leader is a fixed length field of 24 characters consisting record length, status of the record, type of material being cataloged, and base address of data.

The directory identifies the tags that are present in the record and acts as an index to the fields. For each field, the directory provides an entry containing three digit field identifier or tag, four digit field length, and five digits starting position. The directory is terminated by a field separator.

3. BACKGROUND

Early developed library software packages incorporated cataloguing module for developing bibliographic details of documents. These packages initially allowed inputting data of documents for developing bibliographic database using established cataloguing standards. When the cost of cataloguing increased in terms of money in addition to the trained cataloguing professionals, alternative mechanisms were developed to quick and easy way of developing bibliographic database of library documents. Library automation software packages like LibSys[4] have incorporated mechanism to download MARC records from OCLC. Other tools, which were employed to identify, collect and process MARC records are: Aurora ZMarc Collector[5]-tool for retrieval of MARC records from global Z39.50 resources; BIBLIObase's[6] cataloging module to create and manage records in UNIMARC or other MARC formats such as MARC 21, and facilitate the cooperative exchange of bibliographic data; WebClarity's BookWhere[7]-an easy to use and comprehensive copy cataloging tool for building and maintaining library catalogs' BookSystems eZcat and eZcat Pro[8], which uses Z39.50 technology to obtain MARC 21 records; Mitinet's FastMARC Cataloger[9]-to create original MARC records or clean records from automated library system, book vendors, the web. The MARC of Quality's MARC Report[10]-a quality control software for MARC data against common coding and cataloging errors in MARC database, Ross Johnson's MARC RTP[11] to read files of bibliographic records in MARC format, and convert them to a text-delimited format for importing into a database or spreadsheet. The MARC Template Library[12]–a C++ API for reading, writing and processing MARC records, MARCMaker[13], developed by the Library of Congress (LC) , converts structurally sound MARC records and reformats the information into an ASCII text file format, Terry Reese's MarcEdit[14] used for cross walking with MARC data, Systems Planning's

MARConvert[15] - to handle special problems or unusual requirements in converting records into or out of MARC21, USMARC, CANMARC, UNIMARC, or MARCXML bibliographic/authority formats; MarciveWeb SELECT[16]–a tool to search millions of records from LC, NLM, etc., and obtain customised MARC 21 bibliographic and authority records; MARC/Perl[17]–a Perl 5 library for reading, manipulating, outputting and converting bibliographic records in the MARC format; Systems Planning's MARCView[18]–used to view, search, and print ANSI/ISO standard MARC records, UNIMARC records, and MARCXML records.

Other similar tools are OCLC cataloging services and tools for libraries using MARC 21 bibliographic and authority formats; libraries using OCLC have access to WorldCat[19] (OCLC online union catalog), the world's largest bibliographic database, Surpass Copycat[20]-a Z39.50 copy cataloging tool that allows users to find and download free MARC records from the internet; and Visual MARC Editor[21] provides a user interface that displays fields and sub-fields.

4. OBJECTIVES OF STUDY

The aim of the study was to minimise time required for cataloguing and classification by the library professionals, there by increasing the efficiency and effectiveness of the development of library database pertaining to individual libraries using eCataloguer software.

The specific objectives were to:

- Design and develop eCataloguer software using Java for searching MARC records from Library of Congress and build-in local library database;
- Implement eCataloguer software developed using Java and MARC4J API associated with Jakarta Tomcat web server and MS-Access database;
- Correct the MARC data and develop local database of library documents including e-books by any individual library.

5. MARC TAGS AND MARCXML

Bibliographic record of a document is divided logically into fields such as title, author, publisher and so on and represented by 3-digit tags. The tag identifies the field and the kind of data it contains. Most frequently used tags by the library community are :

010	Library of Congress Control Number (LCCN)
020	International Standard Book Number (ISBN)
040	Cataloguing source
043	Geographical code
050	Library of congress call nmber
082	Dewey decimal classification number
100	Main entry - personal name
110	Main entry - corporate name
111	Main entry - conference name
245	Title statement
250	Edition statement
260	Publication, distribution information
300	Physical description
440	Series statement
490	Series statement
500	General note
504	Bibliography note
505	Contents note
520	Annotation/summary note
600	Subject add entry - personal name
610	Subject added entry - corporate name
650	Subject added entry - topical term
651	Subject added entry - geographic term
700	Added entry - personal name
710	Added Entry - corporate name
800	Series added entry

The 9XX section is used for recording local information about specific copy's condition, who donated the item to the library, the copy number, purchase history, processing history, etc. Some fields are further defined by indicators, which are followed by tag number having two character positions with values from 1 to 9. Most fields contain related pieces of data which are called subfields. Subfields are identified by sub-field codes and delimiter.

6. eCATALOGUER SYSTEM DESIGN

The aim of developing eCataloguer is to create bibliographic database of library documents in a library without keying the data. eCataloguer helps the library professionals to create large library databases quickly and easily with minimum human intervention.

The technologies and components of eCataloguer include Java Servlet technology-Java server pages (JSP), MS-Access database, Apache-Tomcat web server, MARC4J application programming interface (API) for working with MARC and MARCXML in Java.

6.1 Programming Language-Java Servlets and Java Server Pages

Java servlets are server side components that provide a powerful mechanism for developing server side programs. Servlets provide component-based, platform-independent methods for building web-based applications, without the performance limitations of CGI (common gateway interface) programs. Servlets enable web developers to create fast and efficient server side application which can run on any servlet-enabled web server. Servlets run entirely inside the Java virtual machine and accesses the entire family of Java APIs, including the JDBC API to access enterprise databases. Today servlets are the popular choice for building interactive web applications.

The JSP is a server-side programming technology that enables the creation of dynamic web pages and applications which is accomplished by embedding Java code into HTML, XML, DHTML, or other document types. When a web browser makes a request to a web server, the static page is converted behind the scenes, and displayed as dynamic content to the viewer. The JSP technology allows Java code and selects predefined actions to be inserted into static web page content. This code is compiled at runtime for each request is made to the page.

6.2 Web Server–Apache Tomcat

Apache Tomcat (Jakarta Tomcat or simply Tomcat) is an open source servlet container developed by the Apache Software Foundation (ASF). Tomcat is the best choice for use as a web server for many applications because it implements both Java servlet and the JSP specifications from Sun Microsystems/ Oracle Corporation to provide the platform to run Java code on a web server. Tomcat receive request from client, dynamically compile a container-managed Java class to handle the request, and return the result to the client.

6.3 MARC4J Application Programming Interface

MARC4J is an open source software library for working with MARC, MARCXML and related bibliographic standards in a popular platform independent language known as Java. The MARC4J software library has built-in support for reading and writing MARC and MARCXML data and also facilitates transformation of MARC records to MODS (metadata object description schema) and Dublin Core using XSL (eXtensible Stylesheet Language). The main aim of MARC4J is to provide an API to develop Java program or servlet that helps reading or writing MARC data. XML is implemented using Java XML interface specified in Sun's Java API for XML Processing (JAXP).

MARC4J provides implementation of an interface called MarcReader for reading MARC data and two methods known as hasNext() and next() to iteratively read MARC data from an input source. MARC4J provides two classes that implement MarcReader such as MarcStreamReader for ISO 2709 format and MarcXmlReader for MARCXML format. Record object model in implemented in org.marc4j.marck package and consists of core interfaces such as Record, Leader, VariableField, ControlField, DataField, and Subfield.

7. eCATALOGUER IMPLEMENTATION

7.1 Installation of Java (JVM and JRE)

Java platform or Java is a widely used platform for programming in the Java language and to deploy portable applications for general use. Java consists of a virtual machine, which must be used to run Java programs, together with a set of libraries (or 'packages') needed to allow the use of file systems, networks, graphical interfaces, and so on. The version of java development kit (JDK) chosen for developing eCataloguer was JDK1.5.0 and java runtime environment (JRE) is JRE1.5.0. Both of them were installed on the system for running the java programs.

7.2 Installation of Apache Tomcat Web Server

Apache tomcat web server version 6.0 was installed on the system for searching and downloading MARC records in a multi-user environment. Configuration screen for the purpose of starting and stopping Tomcat server is shown in Fig. 1.

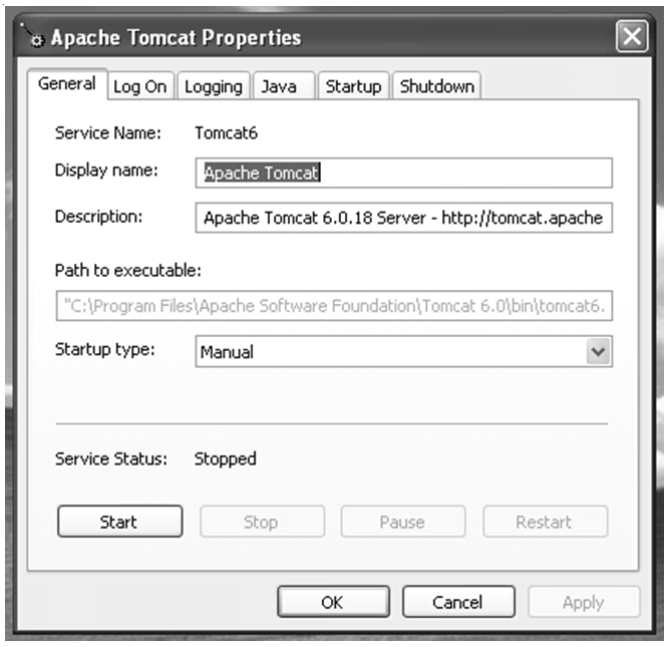

Figure 1. Apache Tomcat configuration set-up.

7.3 Installation of MS-Access

To develop the database easily and quickly, it is advised to use MS Access database for creating and developing library database when the library collection is less than one lakh records. In the experimental set-up, e-books database was created with e-books table structure as shown in Table 1.

Table 1. Table structure of e-books database

Field name	Data type	Field size
ISBN	Text	20
TITLE	Text	255
LCC	Text	100
DDC	Text	100
AUTHORS	Text	255
KEYWORDS	Text	255
EDITION	Text	20
PLACE	Text	100
PUBLISHER	Text	100
YEAR	Text	4
FILENAME	Text	20

7.4 Configuring ODBC Connectivity

A JDBC driver is a software component which enables Java application to interact with database. To connect with individual database, JDBC API requires drivers for each database. In the case of MS Access, JDBC-ODBC, bridge driver which is a database driver implementation, employs the ODBC driver to connect to the database. The bridge driver converts JDBC method call into ODBC function calls. For posting of data using Java into MS-Access database, it is necessary to configure ODBC connectivity through ODBC data source administrator (DSN). ODBC administrator facilitates to create system DSN name with MS Access driver for connecting to Access database. System DSN name is used for connecting to Java. The configuration of ODBC data source is illustrated in Fig. 2 and 3.

8. METHODOLOGY

8.1 Supervised Database Development

eCataloguer is an automatic software tool to develop local database of bibliographic records of library documents developed using JSP technology. The tool takes the ISBN number as input and locates the MARC record existed in the database of Library of Congress. Once MARC record is located, the tool reads the selected fields data from the MARC record and display to the user for making corrections, if any. Upon submission of data, the tool posts the data into appropriate fields in the local database. The local database is created depending on identified data fields using any RDBMS or MS-Access.

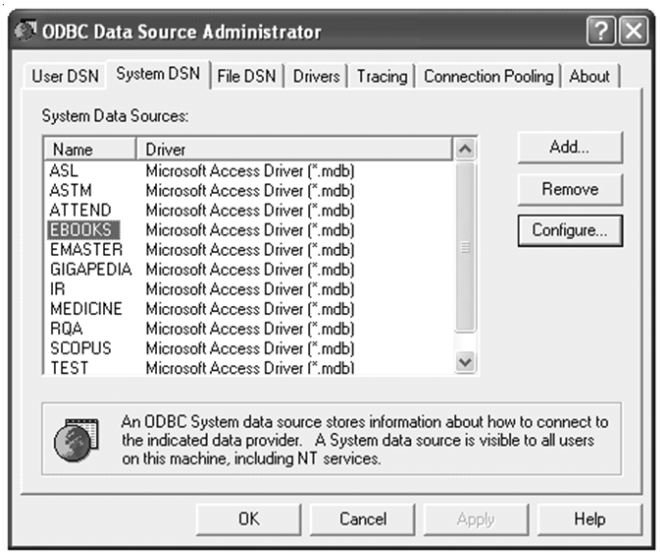

Figure 2. Configuration of system DSN name with microsoft access driver.

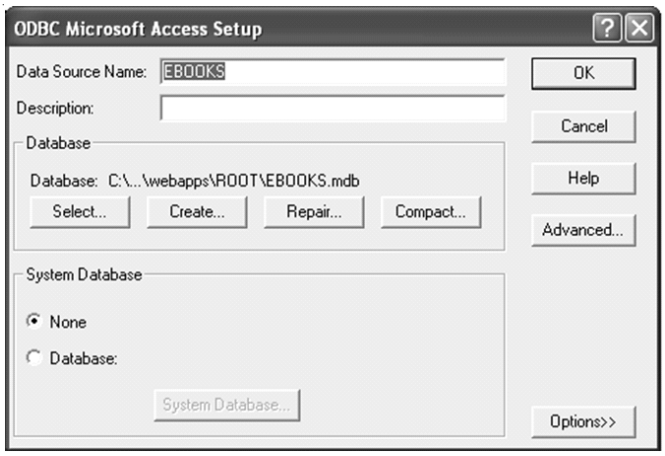

Figure 3. Configuration of data source name with access database.

Finally, an interface is provided to the users to search on various fields for documents of interest on intranet. The purpose of developing eCataloguer tool is to quickly develop the bibliographic database of library documents and publish on intranet. The tool alleviates the efforts of library professionals, manually entering bibliographic data and reduces the typing error while creating the bibliographic database of library documents. Steps involved in the development of eCataloguer system are explained in detailed in the succeeding sections.

Step 1. Searching LC database to locate MARC record using ISBN

The search interface developed in Java aimed at accepting 10 or 13 digit ISBN number with or without a hyphen between prefix identifier, group identifier, publisher identifier, title identifier and check digit. The search interface for MARC record based on ISBN is shown in Fig. 4.

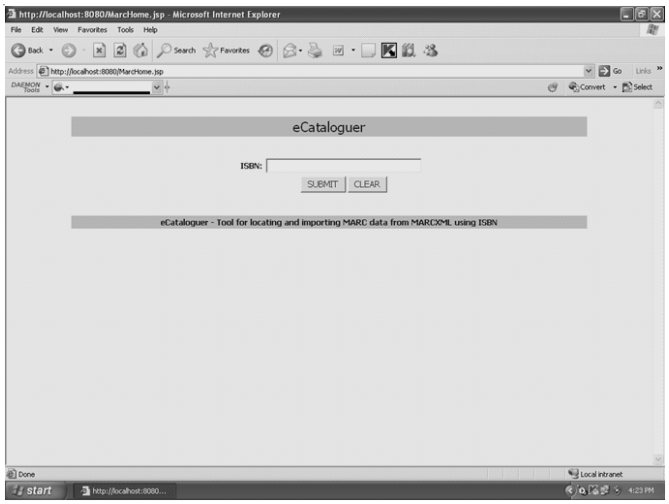

Figure 4. Search interface for MARC record-based on ISBN.

Step 2. Locating exact bibliographic record for identified ISBN

eCataloguer locates the uniform resource locator (URL) pertaining to bibliographic details of the document and matches with the given ISBN number. For example, retrieval of bibliographic details of the document from LC with 0471618942 as ISBN is shown in Fig. 5.

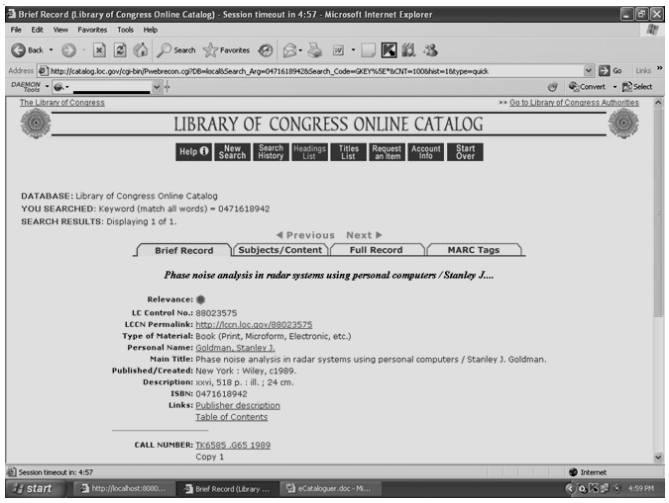

Figure 5. Bibliographic details of a document from LC with 0471618942 as ISBN.

LCCN Permalink is the service offered by LC for citing and linking persistent URLs of the bibliographic records in the LC Online Catalog. LCCN Permalinks are constructed using the syntax: http://lccn.loc.gov/ followed by the normalised LC control number. eCataloguer identifies the LCCN Permalink of the document from LC site. In this case, the Permalink for the document shown in Fig. 5 is http://lccn.loc.gov/88023575 and contents are shown in Fig. 6.

Permalink uses the LC Online Catalog's Z39.50/SRU gateway to retrieve and display records in MARCXML,

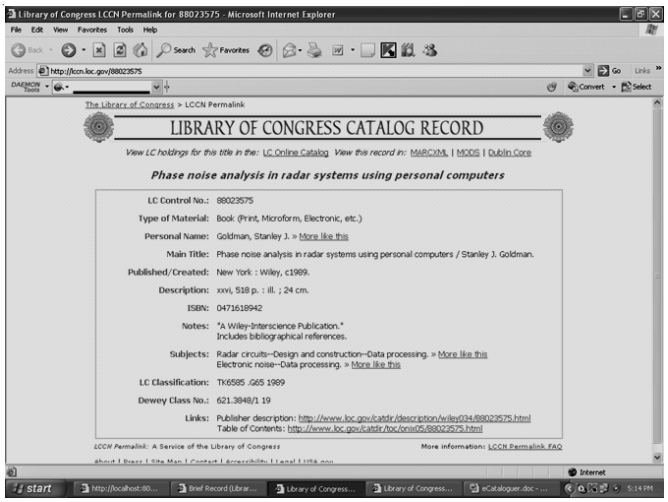

Figure 6. Permalink and contents for the document with ISBN=0471618942 is http://lccn.loc.gov/88023575.

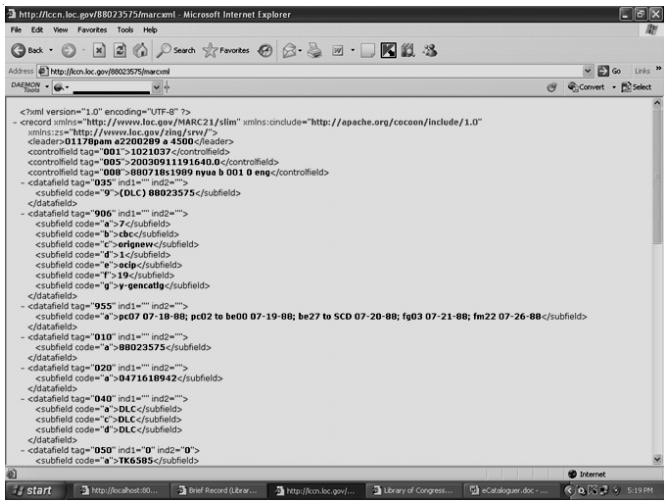

Figure 7. MARCXML display for the document with ISBN=0471618942.

MODS, and Dublin Core format. MARCXML display for the above document as shown in Fig. 7.

Step 3. Retrieval of identified MARC field's data

eCataloguer is designed to extract predefined set of data fields from MARCXML format to meet the local needs of the database development. MARCXML has control field tag, datafield tag, and subfield codes to describe bibliographic details of documents. To meet the local needs, the title, authors, keywords, edition, place of publication, publisher, year of publication, LCC number, and DDC number are extracted and displayed (Fig. 8)

Step 4. Edit and correct MARC data fields, if any

eCataloguer allows to correct spelling mistakes, if any, with the help of editable text fields shown in the Fig. 8. The tool also facilitates to add filename of the eBooks for providing online access to the full-text documents.

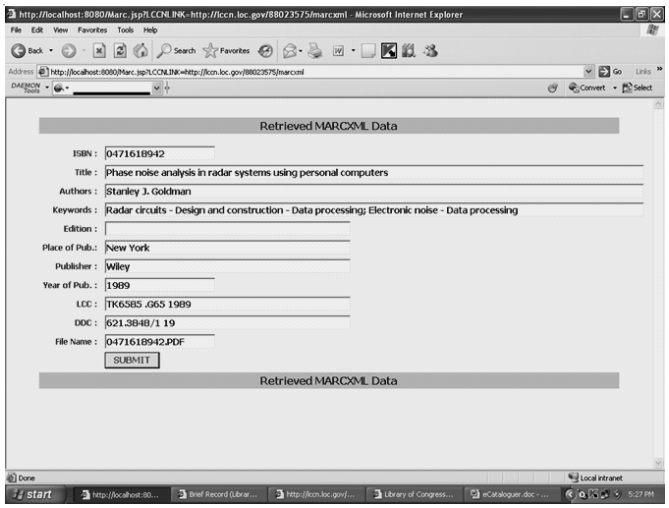

Figure 8. Extracted bibliographic details of the document with ISBN=0471618942.

Step 5. Posting data into library database using ODBC connectivity

eCataloguer is configured to post bibliographic data pertaining to identified document using MS Access ODBC driver. It also displays the status of the record whether it is added to the database or not. Developed MS-Access database of documents as shown in Fig. 9.

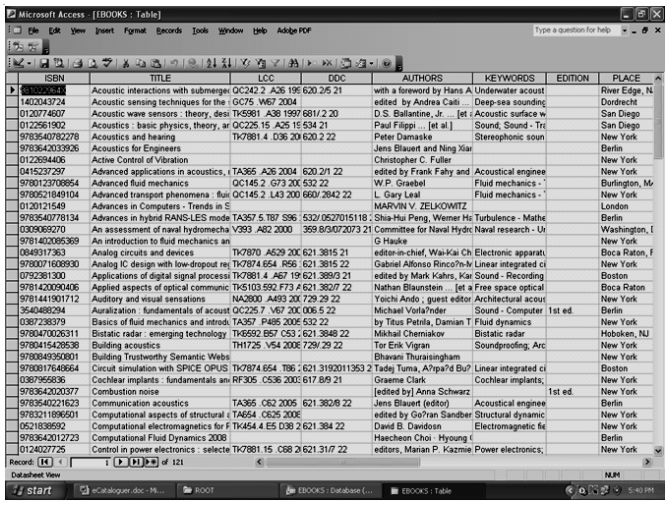

Figure 9. Record structure and data of a documents in MS-Access database.

8.2 Un-supervised Database Development

eCataloguer was designed to develop library database in batch mode, i.e., submission of text file which consists of several ISBNs. In this mode of operation the eCataloguer reads the first ISBN, searches, locates and downloads MARC data if records are found at Library of Congress. The tool will display the status of imported data of available MARC records and also the status of unavailable MARC records. For example, the interface for the batch mode of database development as shown in Fig.10.

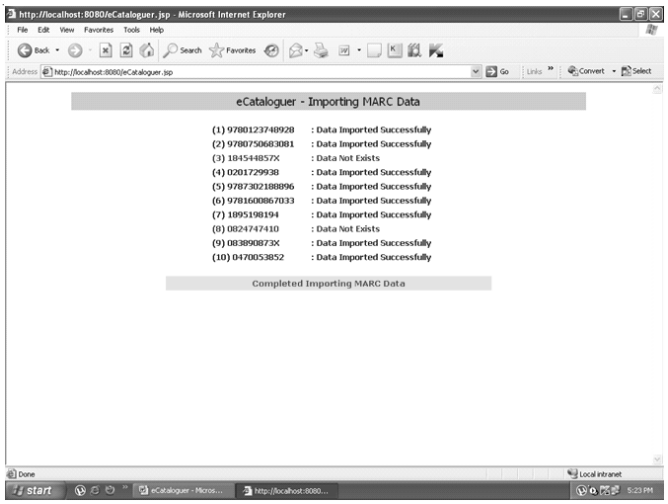

Figure 10. Batch mode interface for the development of database.

9. SUMMARY AND CONCLUSIONS

Recently, libraries have started procuring e-books and investing more budgets due to the advantages with the features of e-books. However, it is not so transparent about the bibliographic implications for managing electronic collections. It is apparent that the main access point for any library's collections is the online public access catalogue (OPAC). In general, libraries use gateway approach for providing access to e-resources such as e-journals, e-books, etc., by isolating library's own collection. To effectively increase the usage of eBooks, it is necessary to develop and maintain library catalogue, which is the primary source of information for any library, and provide searchable collection for e-books integrated with OPAC.

Efforts were made to find out various alternatives for developing database of library documents using software tools without entering bibliographic data. The main purpose of designing and developing eCataloguer software is to develop the database of library documents without physically entering the bibliographic details. After design, development and implementation of eCataloguer software, the researcher has carried out evaluation of test database of more than few hundred records using ISBN. It is found that the supervised mode of downloading and development of MARC data took 5-10 s and unsupervised mode, i.e., batch process took 4-5 s for downloading and posting MARC records into local database, if MARC records exists at LC. From the experimental results, it may be concluded that the eCataloguer software is suitable for easily and quickly developing library database simply using ISBN. The tool is limited to search, locate and download MARC data of printed books as well as e-books from LC only. Further research in the development of OPAC software or configured with any other OPAC will facilitate the retrieval of information in web environment when integrated with database developed using eCataloguer.

REFERENCES

1. Beiser, K. Windows on a catalog database: WLN LaserCat. *Online,* 1995, **19**(1), 82-84.

2. Taylor, A. Cataloguing with copy: A decision maker's handbook. Libraries Unlimited, Englewood, 1988

3. National Bureau of Standards. Guide for the Development, Implementation and Maintenance of Standards for the Representation of Computer Processed Data Elements (FIPS Pub. 45). NBS, Washington DC, 1976, p.1.

4. LibSys Limited. http://libsys.co.in (accessed on 19/09/2011).

5. Aurora Information Technology. Aurora ZMarc Collector http://www.auroraunion.com.au/ait/azmarc1.htm (accessed on 19/09/2011).

6. BiblioSoft. BIBLIObase. http://www.bibliosoft.pt (accessed on 19/09/2011).

7. WebClarity Software Inc. BookWhere. http://www.webclarity.info (accessed on 19/09/2011).

8. BookSystems, Inc. eZcat and eZcat Pro. http://www.booksys.com (accessed on 19/09/2011).

9. Mitinet, Inc. FastMARC Cataloger. http://www.mitinet.com/Products/FastMARCCataloger.aspx (accessed on 19/09/2011).

10. The MARC of quality. MARC Report. http://www.marcofquality.com/soft/softindex.html (accessed on 19/09/2011).

11. Johnson, S. MARC RTP. http://rossjohnson.homemail.com.au/MARCRTP (accessed on 19/09/2011).

12. MARC toolkit for libraries. The MARC template library. http://mtl.sourceforge.net (accessed on 19/09/2011).

13. Library of Congress. MARCMaker. http://www.loc.gov/marc/makrbrkr.html#download (accessed on 19/09/2011).

14. Reese, T. MarcEdit http://people.oregonstate.edu/~reeset/marcedit/html/index.php (accessed on 19/09/2011).

15. Systems planning. MARConvert. http://www.systemsplanning.com/marconvert (accessed on 19/09/2011).

16. MARCIVE, Inc. MarciveWeb Select. http://home.marcive.com (accessed on 19/09/2011).

17. SourceForge.MARC/Perl.http://marcpm.sourceforge.net (accessed on 19/09/2011).

18. Sytems Planning. MARCView. http://www.systemsplanning.com/marc (accessed on 19/09/2011).

19. OCLC. WorldCat. http://www.oclc.org (accessed on 19/09/2011).

20. Surpass Software. Surpass copycat. http://www.surpasssoftware.com (accessed on 19/09/2011).

21. Algorhythms Consultants Pvt Ltd. Visual MARC Editor. http://www.slimpp.com/SlimPPsite/vismarc (accessed on 19/09/2011).

About the Authors

Dr K. Nageswara Rao received Masters degree in Physics in 1988 from Sri Venkateswara University, Tirupati, and Masters degree in Library and Information Science in 1992 from Annamalai University. From 1993 to 1995, he worked at the National Informatics Centre, Hyderabad. During 1995 to 1999, he worked as scientist at the Naval Physical and Oceanographic Laboratory, Kochi. Presently, he is Head, Technical Information Centre, Defence Reseach and Development Laboratory, Hyderabad. His research interests encompass internet technologies, web mining, design and development of digital libraries, and recommender systems. He has published/presented a number of technical papers in national/international journals/conferences.

Dr A.L. Moorthy is Director, Defence Scientific Information and Documentation Centre (DESIDOC), DRDO, Ministry of Defence. He obtained his MSc in Physics in 1976 and BLISc in 1978, both from Sri Venkateswara University, Tirupati; Associateship in Documentation and Information Science from Documentation Research and Training Centre, ISI, Bengaluru in 1984; and PhD from Karnatak University, Dharwad in 2000. His professional career started at the Satish Dhawan Space Centre, Srihari Kota, ISRO, in January 1979. He joined the Defence Research and Development Oganisation at the Defence Electronics Research Laboratory (DLRL), Hyderabad as Scientist in March 1984. Later he moved to DESIDOC, Delhi in 1987. Dr Moorthy is a Life Member of several professional associations. He has published about 45 papers and edited over 30 books/conference proceedings, special publications, and is also on the Editorial Board of many professional journals. He is recepient of commendation letter from Scientific Adviser to *Raksha Mantri* (1989), ILA-Kaula Best Librarian Award-2008 and ILA-CD Sharma Best Paper Award (2008) from Indian Library Association, SR Ranganathan-Kaula Award-2009 Medal and Citation, SIS Fellowship-2009 from the Society for Information Science, New Delhi. His current research interests include open access, intellectual property rights in the digital era, and information retrieval.