DESIDOC Bulletin of Information Technology, Vol. 21, No. 6, November 2001, pp. 25-39

Z39.50: An Information Retrieval Protocol

A. Aruna

Abstract

The enormous growth of various online bibliographic databases on world wide web (WWW)/internet has led the library professionals to learn the diverse data structures and various search interfaces of these databases for effective information retrieval. Z39.50 is a protocol which reduces the burden of library professionals by providing one search interface for multiple databases. It also makes interlibrary loan totally electronic. This paper discusses this protocol in detail.

1. INTRODUCTION

The development of online bibliographic databases and the rapid growth of online services accessible on the internet and world wide web (WWW) has made it possible to access information in ways not possible before. Libraries, whose primary mandate is to provide information, shifted the emphasis from collecting the information in hard copy to accessing information through electronic resources.

However, the major problem is of variation in software and hardware. Library professionals have to learn the specific features of each system, i.e., command languages, search procedures, etc. More the electronic resources grow; more will be the confusion on how to access the information of so many diverse databases. Another problem is the ability of many diverse database structures and different application designs to exchange information electronically.

ANSI/NISO Z39.50 standard was developed to share the bibliographical information electronically and to overcome the problems of database searching with many search languages.

2. WHAT IS Z39.50?

- ☆ Z30.50 is an American National Standard for information retrieval. The official name of the Z39.50 standard is Information Retrieval (Z39.50): Application Services Definition and Protocol Specification, ANSI/NISO Z39.50-1995.
- ☆ Z39.50 is an open communication protocol.
- ☆ It specifies a standard way of communication between two systems for searching databases and retrieving information.
- ☆ It is platform independent, i.e., it allows communication between two computer systems having different hardware and software. In other words, it allows uniform access to a large number of diverse and heterogeneous information resources.
- ☆ This protocol is session-oriented and stateful in contrast to well-known stateless internet protocols such as HTTP (used by WWW) and gopher.
- ☆ It corresponds to the client/server model of computing. In this model, two computers interact in a peer-to-peer relationship

where each computer has specific tasks for the function being performed.

- ☆ It allows searching of multiple databases using a single, standard interface
- ☆ It allows 'broadcast searching', which means user can perform the same search simultaneously against several databases.

3. WHAT Z39.50 IS NOT?

- ☆ Not a common command language
- ☆ Not a user interface nor an application program interface (API)
- ☆ Not a specification for building a database
- ☆ Not a specification for the telecommunication network
- ☆ Not a bibliographic record structure.

4. HISTORY

The name Z39.50 comes from the fact that the National Information Standards Organisation (NISO), a standard developer for American National Standards Institute (ANSI), was once the Z39 committee of ANSI. NISO standards are numbered sequentially and Z39.50 is the fiftieth standard developed by NISO.

NISO approved the original Z39.50 standard in 1988 (referred to as Z39.50–1988 or version 1). It was developed as an application layer of the OSI (Open System Interconnection) communication model. Z39.50-1988 is now out-dated and incompatible with later Z39.50 versions. It formed the basis of the wide area information server (WAIS) protocol.

Almost at the same time, ISO approved for a 'search retrieve' protocol called 'SR'. This was almost identical to Z39.50 except that it used abstract syntax notation one (ASN.1)/basic encoding rules (BER) as a protocol data unit encoding and omitted a few functions. This work was standardized in 1991 as ISO 10162/10163.

In 1990, Z39.50 Implementors' Group (ZIG) was formed to develop profiles. Later it changed its activity to develop and recommend enhancements to the standards. In addition, in 1990 a Z39.50 Maintenance Agency was also established, at the Library of Congress. In 1991, a second version of Z39.50 was prepared and put out for ballot. This became Z39.50 version 2 or Z39.50-1992. It provided basic capabilities for bibliographic searching and information retrieval, primarily of MARC bibliographic records. The standard is capable of building simple and complex search queries using a wide range of query qualifiers and search terms. This version does not support features currently available in some very sophisticated systems nor can it be conveniently used for non-bibliographic records. This version was strongly influenced by SR and was a compatible superset to it.

Development of Z39.50 version 3 began in late 1991. In 1995, version 3 was accepted. Version 3 was much larger than version 2. In version 2 very little was optional, while the vast majority of the new functionality and changes in version 3 were optional. An additional reason for the bulk and complexity of version 3 was that it included version 2. Version 3 was designed as a superset of version 2, which incorporated the ability to fall back to the older version 2 specification if the parties involved did not support version 3 for the sake of backward compatibility with the existing base of implementations. The major new services developed for version 3 were *Scan* and *Explain*. TCP/IP transport service was also included in this version. In 1994, ISO decided to adopt the text of the Z39.50 version 3 as an ISO standard, instead of amending SR. Recently it was expected as ISO 23950.

5. Z39.50 AND TCP/IP

Z39.50 was developed on open system interconnection (OSI) protocol. This protocol consists of seven logical layers of hardware or software (from top to bottom). These are *Application, Presentation, Session, Transport, Network, Data link,* and *Physical*. Each layer communicates with the layer immediately above it and below it. Though the OSI is a *de jure* standard (developed on paper leaving the proof for after establishment), the internet is ruled by a *de facto* standard (it was proven before its establishment as a standard)

TCP/IP. In order to make Z39.50 suited to internet environment, it was necessary to move it from OSI framework to TCP/IP of internet. TCP/IP is a five layer protocol with its layers closely corresponding to OSI layers. The main difference in layers is that the *Application layer* of TCP/IP must perform the same services of the *Application, Present* and *Session layers* of OSI. Functionally, the similar layers of OSI and TCP/IP are equivalent, but they are not compatible. In 1992, shortly after the Z39.50 version 2 was released, a group of implementors established the Z39.50 interoperability testbed (ZIT) to make Z39.50 implementations work over the internet. Interoperability testing of various Z39.50 implementations through the ZIT made it clear that the transport services of the internet could be used successfully for the protocol. A TCP port number for Z39.50 is registered.

6. HOW Z39.50 WORKS?

6.1 Client and Server Versus Origin and Target

There are following two components to the Z39.50 architecture:

- Client—called an 'origin' in the standard, and
- Server—called a 'target' in the standard

The Z39.50 origin is the search interface software (client software), and Z39.50 target is server software. Here one computer (origin) submits a search request to another computer (target), which then services the request and returns the results. The messages sent between client and server are called protocol data units (PDUs). The structure of the PDU is described in ASN.1. The BER are used to serialize the ASN.1 structure. Generally all PDUs work in pairs; in most cases, the client sends a PDU to the server, which replies with another PDU. The PDU sent from the client is usually a request, such as search request or request to delete search results. Whereas, the PDU sent from the server is usually a response, such as details of the search or confirmation of the deletion of search results. Z39.50 is a program to program the protocol. End user is not aware of it. User interacts with user interface, which is a part of client application. The client application converts the queries accepted at interface into the form required by Z39.50 and communicates to the server. At the server's end, database application takes these and converts them into an appropriate form for running against the database. Result set is communicated back. Fig. 1 shows the working of Z39.50 protocol.

7. FACILITIES AND SERVICES

Z39.50 is divided into eleven basic facilities. Each facility is divided into one or more services as shown in fig. 2. A service facilitates a particular type of operation between the client and the server. out of eleven facilities, three are core facilities and they are *Initialisation, Search* and *Retrieval.*

7.1 Initialisation

It allows the client to negotiate a Z-association. This includes supported Z39.50 features, default character set, default language, protocol version and user authentication.

7.2 Search

It enables a client to query databases at a Server. The search request PDU, sent from the client to the server, contains the user's query, in a well-known search format. The query may contain Boolean operators, fielded search terms, proximity searching, weighted search terms, truncation specification,

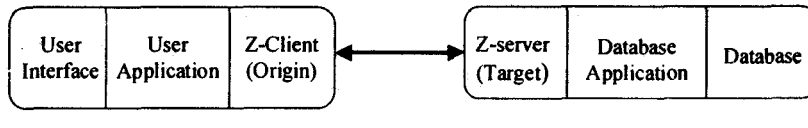

Figure 1. Working of Z39.50 Protocol

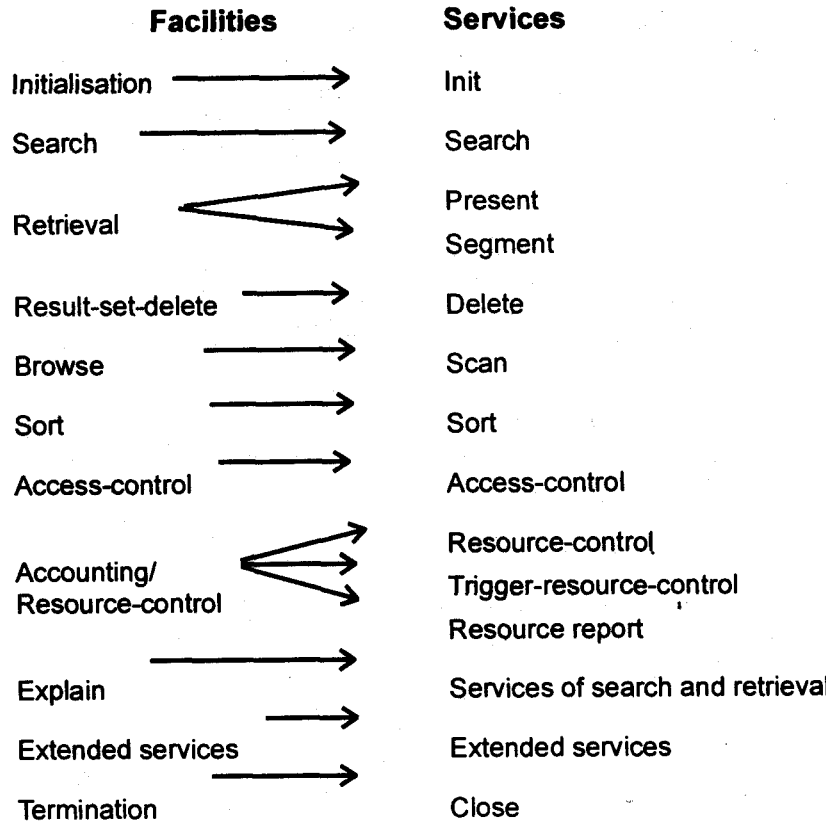

Figure 2. Facilities and Services of Z39.50

generalised pattern-matching, etc. The server takes this query, carries out a search on the database, and stores the results. It then sends a search response PDU back to the client, giving details of the search results. It should be noted that the search results themselves are not transmitted to the client. Only the details about the search results are transmitted. Some servers can store the results of many searches at any given time. The Bib-1 attribute set is used to define search parameters for bibliographic records.

7.3 Retrieval

This facility offers following two services:

- **Present**: allows the client to request one or more records from a specified result set. This includes requesting specific ranges of search results (e.g., records 10 through 20), specific elements in a record (e.g., title and author), specific variants of the record (e.g., MS-Word and HTML, or English and French), search term highlighting, etc. Present service is more useful to the librarians as it defines how records can be requested, i.e., MARC, OPAC, etc.
- **Segment** *(added in version 3):* if the records requested by a present request do not fit in a single segment, segment breaks the large number of records into smaller numbers.

7.4 Result-set-delete

Here, the client request the server to delete one or more sets of results by use of the delete request PDU. Server's delete response PDU contains operation status.

7.5 Browse *(added in version 3)*

This service is called 'scan' and allows an ordered list, e.g. alphabetical list of subject headings or authors to be scanned or browsed. Scan is a request to bring back a list

of headings in order from a specific start point and allow browsing. Browsing is an important feature when searching remote catalogues.

7.6 Sort *(added in version 3)*

Sort facility is a single service carried out on the server. It allows a client to request the server to sort a result-set (or merge multiple result sets and then sort). This way of working is easier, faster and more flexible.

7.7 Access-Control

This facility covers security measures. The server as part of a specific operation or the whole association initiates this. The access-control request/response mechanism can be used to support access-control challenges or authentication, including password challenges, public key cryptosystems, and algorithmic authentication.

7.8 Accounting/Resource Control

It has following three service:

- **Resource-Control**: permits the server to send a resource-control request, e.g. notifying the client, that either actual or predicted resource consumption will exceed the agreed upon limits. A consent to continue an operation is sought from the client—'Can I spend some more of your money?'
- **Trigger Resource Control**: The client triggers this service and it asks the Server 'how much money is left in the account?' The response is a resource-control message with a resource report.
- **Resource Report**: The resource report which is passed to the client can contain a lot of detailed information including figures on:
  - Numbers of records in current set so far
  - Estimated complete set size
  - Processing time so far
  - Estimated cost so far for this operation
  - Estimated cost if completed
  - Cost for the client/server association.

7.9 Explain Facility *(added in version 3)*

Explain facility is designed to maintain a searchable database of all the features of a Z39.50 implementation on a server. The client can interrogate this database and find out exactly what services and their basic characteristics are available at the server. These include general features (description, contact information, hours of operation, restrictions, usage cost, etc.), databases available for searching, indexes, attribute sets, attribute details, schemas, record syntaxes, element specifications, sort capabilities and extended services supported. It uses the services of search and retrieval.

7.10 Extended Services *(added in version 3)*

The extended service facility is a major enhancement to the standard's original SR objectives. Extended service allows the client to start-up specific 'task packages' on the server and to control how they operate. It provides access to the service outside the protocol. Presently, the defined features of extended service are:

- **Database update**: A client can insert, modify, and delete database records, or elements of a database record.
- **Persistent result sets**: A client can save one or more search result lists on a server for access and refinement later.
- **Persistent queries**: A client can save one or more favourite queries on a server for access at later time.
- **Periodic queries**: A client can set up a set periodic query schedule with the server for executing the client's query on a regular basis.
- **Document ordering (ILL request)**: A client can request delivery of a specific document.

When extended service task package starts, the details are kept in an extended service database. These may be searched (via Z39.50 of course) in order to see what is running on the server.

7.11 Termination *(added in version 3)*

The termination facility allows the client or server to close down a Z-Association. The reasons for the termination can be system problem, cost limits, security violation, etc. The client can also ask for a resource report as part of the close request, i.e., how much money have I got left in the account?

Appendix - 1 covers the examples of envisaged order of 'use of facilities within a session' and for a detailed facilities diagram refer to the article Z39.50 Pt 2–Technical details, *Biblio Tech Review*, October, 1998.

8. OBJECT REGISTRIES

Z39.50 uses registries for various types of objects, used for interoperability with the protocol. Theses are referred to via object identifiers (OID). OIDs are used as parameters in the various protocol requests and responses that take place between Client and server. The three major registries are *query sets, attribute sets and schemas* and *record syntax.*

8.1 Query Sets

A query is a string of data and associated parameters that define the records being sought, the database to search and so on. The query message sent to the server is hidden from the user with the client reformatting normal OPAC style search statement. The standard at present specifies six query types:

(1) **Type 0**: designated 'private', allowing two systems to use a private mutually agreed upon query format.

(2) **Type 1**: queries are expressed by individual search terms, each with a set of attributes. Terms may be combined/linked by Boolean operators. Terms and operators are expressed in Reverse Polish Notation (RPN), i.e., nierarchical search expression, sub expressions connected by AND, OR and AND-NOT. This is used for bibliographic queries.

(3) **Type**: commands for interactive text searching as specified by ISO 8777.

(4) **Type 100**: Common command language for online interactive information retrieval as specified by ANSI Z39.58.

(5) **Type 101**: (extension of type-1) query for proximity searching

(6) **Type 102**: ranked list query

(7) **Type SQL**: SQL query *(Currently under review)*

8.2 Attribute Sets and Schemas

The attributes (or access points) associated with search queries are presented as an attribute set. An attribute set can have one of several values and these values govern the manner in which a search proceeds.

Attribute sets can be logically divided into two types, domain specific and facility specific—domain specific being directly related to attributes associated with individual communities, whilst facility specific being essential for implementing certain Z39.50 facilities.

8.2.1 Domain Specific Attributes

Following are domain specific attributes:

- ★ Bib-1 - Bibliographic
- ★ GILS - Government information locator service
- ★ STAS - Scientific and technical
- ★ DL - Digital library collections
- ★ CIMI - Museum collection information
- ★ GEO - Digital geospatial metadata, etc.

The most popular attribute set is Bib-1. It was developed for the bibliographic resources (to view different library systems for searching and retrieving information in standard and mutually understandable terms) but now commonly used for a wide range of applications.

Bib-1 comprises six groupings of attributes, or attribute types. These are:

- ♦ **Use attributes**: defines the access points for a search (title, author, subject, etc.).
- ♦ **Relation attributes**: determines how the search term entered by the user relates to values stored in the database index (less

than, greater than, equal to, phonetically matched, etc.).

- **Truncation attributes**: defines which part of the value stored in an index is to be searched (the beginning of any word in a field, the end of any word in a field, etc.).
- **Completeness attributes**: specifies whether a search term can be the only value in an index.
- **Position attributes**: specifies where in an index field the search term should occur (at the beginning, anywhere, etc.).
- **Structure attributes**: specifies the form to be searched for a word/phrase/date, etc.

8.2.2 Facility Specific Attributes

Facility specific attributes are as follows:

- ★ CCL-1 - Common command language
- ★ Exp-1 - For use with an explain database
- ★ Ext-1 - for use with an extended service database

8.3 Response Record Syntaxes

Z39.50 uses record syntax concept to transmit the records/data. After the search is completed with some search results, the next decision is—how a record should be displayed. The client can ask server to send the data in the display format of its choice. If the server can not send the data in the requested format, it is possible for the server to send the record back in some other format that it is capable of. The protocol distinguishes two types of response records that may occur from the server: database and diagnostic records. Both record types may be returned in several formats.

8.3.1 Database Records

MARC formats: Z39.50 supports most of the main MARC formats—assigning them an 'object identifier number' so that the Client and Server can easily understand which flavour of MARC they are dealing with. These formats are exchanged in ISO2709 format. USMARC, UNIMARC, UKMARC, CANMARC, etc. are various MARC formats.

OPAC (Online Public Access Catalogue): the OPAC syntax is designed to simulate a catalogue card or similar bibliographic reference format. The OPAC syntax includes additional fields for the display of holdings and circulation data e.g. date due if on loan.

SUTRS (Simple Unstructured Text Record Syntax) (text only): designed to allow ASCII text records to be transferred during a Z39.50 session. The syntax does not have any field structures.

GRS1 (Generic Record Syntax 1): a complex tree like structure designed to represent a database record. Although complex, it can be used to represent any type of database records.

Summary: the summary record syntax is used to present brief records to the user as the first response during the search operation. Fields include, title, author, call number place, date.

Explain: Server Information syntax. This syntax is used for querying the explain database where information about available services is stored; e.g. databases available, record syntaxes supported (versions of MARC), lists of terms available for the Browse function, payment terms, languages, etc.

Extended: when a user starts an extended service, the details of the 'task package' are held in the extended services database. These details may be queried via Z39.50 and information is presented using the extended services record syntax. Information held is brief (package name, user ID, retention time permissions, etc.).

SQL-RS: SQL record syntax supporting SQL3 datatypes (under review)

8.3.2 Diagnostic Records

- ★ Error formats: bib-1, diag-1
- ★ Resource report: resource-1, resource-2
- ★ Access-control: prompt-1, des-1, krb-1

9. Z39.50 PROFILES

A Z39.50 profile is a set of implementor agreements, which specifies the use of Z39.50 to support a particular application (e.g. GILS or WAIS), function (e.g. author/title/subject searching), community

(e.g. library community, museum community, chemists); or environment (e.g. Europe, Canada).

Z39.50–1995 has many optional features and is an open standard. It is up to the implementers to decide upon the more specific set of Z39.50 requirements. The group of implementers with common interests agrees upon the set of Z39.50 features to be followed, and form a Z39.50 profile.

The advantages of forming a Z39.50 profile are listed below:

- ☆ Vendor can be provided with specification to build the software product, or these specifications can be used in selecting the software.
- ☆ It increases the reliability of results as a profile will define the attributes and attribute combinations, which are supported by server.
- ☆ It will reduce the instances of varying attribute definitions between client and server.
- ☆ Profile will help in standardising the server configuration i.e., the configuration will be the same for all servers which confirm to the profile. This will also help in configuring client easily as server configuration will be known which in turn will improve interoperability and search results.

There are many well developed profiles and some are under development. Some of them are:

- ♦ **GILS profile**: 'Government Information Locator Service' profile is the outcome of the need for users to identify and locate publicly available government information resources.
- ♦ **ATS profile**: The 'Author-Title-Subject' profile aims to improve the reliability of Z39.50 search results.
- ♦ **WAIS profile**: The 'Wide Area Information Server' profile specifies rules for access to WAIS server supporting Z39.50 version 2.

It is not possible to discuss all the profiles. A few popular profiles are listed here:

- ♦ **CIMI profile**: Computer Interchange of Museum Information profile
- ♦ **DL profile**: Digital Library profile
- ♦ **CIP profile**: Catalogue Interoperability Protocol profile
- ♦ **ZSTARTS profile**: a Z39.50 profile for STARTS (Stanford Protocol for Internet Search and Retrieval).

10. Z39.50 AND WWW

The point to ponder is when Web is giving all the solutions to access information available on internet, what is the need for Z39.50. It is true that Web provides, a unified terminal interface to navigate through and ability to transfer files. However, for most current library applications, the user is typically dropped into a telnet session. A number of bibliographic services are being made available through common gateway interface (CGI) interfaces. A gateway is written to a server, and the user is offered forms-based access. This provides access to a particular bibliographic resource. This work has to be repeated for each resource. Z39.50 comes into picture at this point. An http-Z39.50 gateway (Z39.50 client hosted on a Web server, referred to as a Web/Z39.50 gateway) allows a number of resources to be made available through the same interface, and allows a user to access the bibliographic resources using their familiar Web browsers. The browser accesses Z39.50 client software on a Web server. This software presents the searcher with HTML forms to build the query and provides other search support functionality. The gateway uses the information entered in the form to create a Z39.50 message that is sent to one or more Z39.50 server systems selected by the user. The server sends the Z39.50 formatted response to the client software on the gateway.

The gateway program formats the structured output from the bibliographic database in HTML so that it can be returned via the Web to be presented in a human-readable format. By providing a user interface, only the Web/Z39.50 gateway uses Z39.50 to communicate to the remote server

and not the software on the searcher's computer. The searcher only needs browser software to access a gateway service in order to query Z39.50 servers. An application built on Z39.50 is smart to understand the returned data structure. It tags it for input into a personal bibliography; takes a record and searches in another database for works by the same author; passes the record to a document request service; and so on. Z39.50 allows the creation of interworking bibliographic applications, which share structured data. The final output of such a service is delivered to the user through a Web interface. The Web begins where the need for smartness ends. Thus the Web is used to provide a user interface to particular resources or applications and Z39.50 is used for interworking bibliographic applications.

According to Hammer Sebastien, "there is a strong potential for a profitable and synergetic relationship between the WWW and Z39.50. The two worlds are merging together, with each one growing stronger by using the best elements of the other-hyperlinks between systems and document types from the WWW—structured searching and document discovery from Z39.50". There are a number of Z39.50 interfaces on WWW available for a variety of platforms. These interfaces allow users to search and retrieve directly from the Web browser (e.g. Internet Explorer, Netscape Navigator) without the need for new software. A short list of the same is given at Appendix 2. A schematic view of Z39.50 and WWW is shown in fig. 3.

11. IMPLEMENTATION

In the International scene, most of the library system vendors have incorporated Z39.50 protocol into their products. The Z39.50 source code, client and server

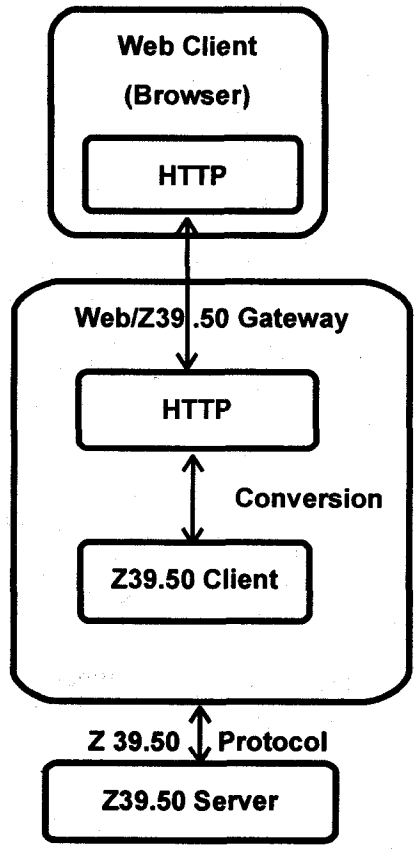

Figure 3. A Schematic View of Z39.50 and WWW

software and application programming interfaces (APIs) are available in the public domain for use by noncommercial and commercial implementors.

A Z39.50 can be implemented in libraries in various ways. If a library wishes to make its databases and catalogues accessible to searching by Z39.50 clients, it needs Z39.50 server software. Installing this system can be complicated and expensive. Libraries that do not have to have a Z39.50 server to be able to search other libraries' catalogues and databases, can go for a Z39.50 client. Client software can be installed in a few different ways. A copy of the software can be installed on each computer. It can be installed on a single computer, with other computers having access to the software via a network. Another option is a Z39.50 client hosted on a Web server. When using the services of a gateway, the searcher only requires a Web browser and not a Z39.50 client system to search a Z39.50 server. The Library of Congress, USA has a Z39.50 gateway. It provides access to the catalogues of many institutions, including the University of Alberta, USA (which is listed alphabetically under NEOS). Europgate has links to a number of European and American databases. The National Library of Canada's vCuc gateway provides Z39.50 access to the catalogues of several Canadian libraries. A list of gateways is available at :

http://www.lib.ncsu.edu/staff/morgan/aleuin/wwwedcate logs-z39.50.html

A number of groups of libraries have recognized the potential of Z39.50. Projects have been undertaken in some American libraries to determine the feasibility of offering Z39.50 services and implementing them. Some of them are: SILO (State of Iowa Libraries Online) project, Virtual Canadian Union Catalogue project, Report to the CIC, DALI (Document and Library Integration) project, ONE (OPAC Network in Europe) project.

12. ADVANTAGES OF Z39.50

- ☆ Any database that is Z39.50-enabled may be searched by using a single search interface, a single set of search commands and search strategies.
- ☆ Allows intersystem searching and record transfer irrespective of different software and hardware.
- ☆ Multi-database searching using single search interface.
- ☆ Virtual union catalogue—a combined catalogue of several libraries on diverse systems (without merging the catalogues) is possible. Z39.50 clients will permit searching of members' collections with a single interface.
- ☆ Interlibrary loans (ILL) process will be made easier. Extended services for ordering documents, updating databases and sorting searches can be defined and controlled via Z39.50.
- ☆ Records can be displayed in any flavour of MARC. In addition, work is going on to use generic set of data element called Dublin Core to use as attribute set and record syntax.
- ☆ CD-ROM databases from various vendors could be searched with a single interface. The time and expense of training users on multiple interfaces could be reduced.
- ☆ The SDI or selective dissemination of information service could be set to run a saved search in the catalogues of many different libraries regularly and automatically with results being sent to the user by fax or e-mail.

13. DISADVANTAGES OF Z39.50

- ☆ Very complex.
- ☆ Levels of services supported within and between the particular implementation depend on Z-client/Z-server and host library management system.
- ☆ The differences between the facilities on Z-client and Z-server lead to poor output.
- ☆ Multi-database searching leads to slow response time and information overload.

14. CONCLUSION

The Z39.50 protocol not only rescues a librarian from learning different search interfaces for different databases, it also helps the librarian to improve resource sharing among libraries by virtual union catalogue and ILL. Virtual union catalogue does not require to be merged at one place. The catalogue is retained distributed. The extended facility of this protocol makes the ILL totally electronic i.e., it allows system to arrange for delivery, including account verification and billing, of an item to the enquirer. It provides 'broadcast searching' (i.e., same search is performed simultaneously against several databases). Z39.50 also allows users to specify search statement to be saved and run at intervals. This greatly helps the librarian in providing the SDI service.

Besides this, the library community has much to offer in the way of providing structure to information resources on the internet. The Z39.50 standard is a concrete representation of this fact.

In India, much attention has not been paid on this. Considering the endless advantages Z39.50 protocol to the library community in specific, it is necessary that we insist for Z39.50 support from software vendors and developers and also start thinking to implement this.

REFERENCES

1. Behjat Al-Yousuf. A review of Z39.50 FAQ. September 1998. http://lcweb.loc.gov/z3950/agency/papers/zfag.doc
2. Z39.50: Pt. 1: An overview. *Biblio Tech Review.* April 1999. http://www.bibliotech.com/html/z39_50.html
3. *Biblio Tech Review.* Z39.50: Pt 2: Technical Details. April 1999. http://www.bibliotech.com/html/z39_50_part_2.html
4. An evaluation of Z39.50 within the SILO project. Blue Angel Technologies, Inc. http://www.silo.lib.ia.us/bluang.html
5. Dempsey, Lorcan, *et al.* Towards distributed library systems: Z39.50 in a European context. *Program*, 1996, **30**(1).
6. Denenberg, Ray. Z39.50 recent development and future prospects. http://lcweb.loc.gov/z3950/agency/papers/kbr.html
7. Hammer, Sebastian & Favaro, John. Z39.50 and the World Wide Web. *D-Lib Magazine,* March 1996. http://www.dlib.org/dlib/march96/briefings/03indexdata.html
8. Iltis, Susannah. Z39.50: Overview of development and future. March 1995. http://www.cqs.washington.edu/~camel/z/z.html
9. Kelly, Annette & Alton, Bryan. IRIS: A Z39.50 based service for database searching and document ordering. *Journal of Information Networking*, 1995, **2**(3).
10. Kunze, John A. & Rodgers, R.P.C. Z39.50 in a nutshell: An Introduction to Z39.50. http://www.informatik.tu-darmstadt.de/VS/Infos/Protocol/Z39.50/z39.50-nutshell.htm
11. List of Z39.50 compliant library systems. http://www.bookwhere.com/hosts.htm
12. Lunau, Carrol D. The need for an international Z39.50 profile for searching virtual catalogues. October, 1998. http://nlc-bnc.ca/resource/vcuc/profiss.
13. Lynch, Clifford A. The Z39.50 information retrieval standard. Pt 1: A strategic view of its past, present and future. *D-Lib Magazine,* April 1997. http://www.dlib.org/dlib/april97/04lynch.html
14. St. Pierre, Margaret. Z39.50 explained. http://www.blueangeltech.com/NewsAndInfo/FAQs/ Z3950.htm
15. Miller, Paul. Z39.50 for All. *Ariadne*, **21**, 20 September 1999. http.//www.ariadne.ac.uk/ issue21/23950
16. Payette, Sandra D. & Oya Y Rieger. Z39.50: The user's perspective. *D-Lib Magazine* April 1997.

17. Finnigan, Sonya & Ward, Nigel. Z39.50 Made Simple: http://www.dstc.edu.au/DDU/ projects/ Z3950/zsimple.htm
18. Thompson, Valerie. New technology for library information systems bolsters e-commerce applications. *Byte,* April 1998.
19. Turner, Fay. An overview of the Z39.50 information retrieval standard. *UDT Ocassional Paper* No. 3; July 1995. (revised January 1997) http://www.ifla.org/VI/5/op/udtop3/udtop3.htm
20. Turner, Fay. Document ordering standards: The ILL protocol and Z39.50 item order. *Library Hi Tech*, 1995, **13**(3).
21. Turner, Fay. Selecting a Z39.50 client or web gateway. 1998. http://www.lib.msu.edu/hi-tech/ Turner.html
22. Z39.50 Maintenance Agency. Information retrieval (Z39.50): Application service definition and protocol specification. July 1995. http://lcweb.loc.gov/z3950/agency/1995 doce.html
23. Z39.50 client and web gateway survey http://www.dstc.edu.au/RDU/reports/zrevi ews/

Contributor: **Ms A. Aruna** is scientist 'E' at Gas Turbine Research Establishment (GTRE), DRDO, Bangalore.