Željko Pajkić, student
(Matematički fakultet, Beograd)

Dejan Jovanović, Zoran Ognjanović
(Matematički institut, Beograd)

A Z39.50 GATEWAY IMPLEMENTATION

Abstract: An implementation of an Z39.50 server and the corresponding gateway is presented. The software enables one to retrieve data from remote databases. Its interface is web-based with user-friendly interface. Our implementation is based on free software.
Key words: Z39.50, web, distributed databases, gateway, server, digitization

1. What is Z39.50 and why do we need a gateway?

1.1. Introduction. This standard specifies a client/server based information retrieval protocol. It was originally proposed (in 1984) for use with bibliographic information. On the server, several data sources may be available. The client is able to search the server and the server returns results to the client. For instance, you can search the Library of Congress for a specific book, using its author as the key word. On the other hand, you can search all the abstracts if you are not sure what you are looking for.

Suppose that you need to search several libraries to find a book. An inconvenient way to do so would be to visit and search each one of them, manually. Even if they all have its own web sites and search engines you still have to deal with every one of them each time you want to run a query. We already have a standard for information searching and retrieval from one server at a time – it is the Z39.50 standard. But, we would like to be able to search many databases at the same time.

A gateway is specialized software that works as an interface between clients and servers. It is a way to connect a single client with many servers. it is not an issue for a server to deal with several clients simultaneously. A gateway enables the client to query many servers with a single search. Through a gateway, we can achieve several goals:
- Interoperability
  - between different databases and user interfaces
  - between different libraries (data formats)
  - libraries in different countries
  - between different communities (libraries, museums, archives)
  - parallel search and information retrieval

Unfortunately, communication between client and end-user is not addressed by the Z39.50 protocol. Therefore, we had to find our own way to implement it.

There are only few consulting companies that work on Z39.50 software implementations in the world. Competent professionals in this software area are very rare. Moreover, a free implementation of such software does not exist. In addition, there are com-

mercial Z39.50 clients that satisfy the above requirements. 30 day trial version of a powerful client Webclarity BookWhere can be downloaded from their web site [10].
In our gateway implementation, we have also built a Z39.50 server [5]. Thorough discussion of both pieces of implemented software is following.

1.2. The gateway. Our gateway implementation [2] was planned to be a so-called web gateway, because it has several advantages. Just to name a few:

- Web site provides an intuitive interface with great ease of use. The users need no previous experience or knowledge.
- End-users do not need special clients, because any web browser will suffice. Hence, there is no need for development of a client application.
- It is supported by almost every system.
- It can be built with free (and free of charge) software.

Here is a brief description what this software actually does and what is happens in the background when a user enters a query.

2. User points his web browser to gateway's web site [2] and fills in the search form. This operation includes selection of either basic or advanced search. If advanced search is selected, data sources to be searched can be chosen.
3. After a query has been submitted, the gateway forwards it to the selected Z39.50 servers. When all the results arrive, they are merged and returned to the web user in a convenient form.

Z39.50 is a very general standard with many different implementations around the world. A list of servers exists [6] where active hosts and their configurations are shown. When a gateway is implemented, the following difficulties must be taken in consideration:

- Many query formulation types have to be implemented
- Many different attribute sets have to be implemented
- Many return-record types have to be implemented

1.3. The server. This is the other peer of the protocol. A Z39.50 server is encapsulating the abstract database model. It is connected to a real database, making it a Z39.50 target. Abstract queries from the client are translated into native queries and run at the database.

Today, the predominant kind of database management systems is relational. Their query language SQL is well known. Relational databases are widely accepted on the market and even their extension – object oriented databases are built on top of them with SQL support. Another good reason to use them is that a number of great free database systems exist on the market.

We were using zSQLgate [3] to build our server. This software makes it possible to connect virtually any database that supports SQL queries, making it a Z39.50 server. Creation of Z39.50 targets becomes just a matter of configuration. The downfalls of this software are the following. First, this software is not free, although licensing it might be free of charge. Next, this software is still beta – it is not in production stage yet. Just a few popular query types, attribute sets and return record syntaxes are supported. However, we have found it to be stable and reliable so far.

1.4. Application. We now have online both gateway and server. It is connected to a MySQL database that holds bibliographic records of the library of mathematical institute. At the present moment which corresponds to testing period, through the software we can access the following databases:

- Library of Mathematical Institute, Belgrade, Serbia
- Library of Congress, Washington DC, USA

There are no limitations for connecting some other data sources to our gateway.

2. Software

2.1. Technical details. We have used Linux operating system to implement the software. Both Z39.50 server and client (encapsulated in gateway) rely fully on YAZ Toolkit [4].

2.1.1. *The gateway*. Luckily, YAZ client is supported by PHP [8]. The PHPYAZ extension [4] needs to be installed on top of standard PHP distribution. Web server is Apache [11]. Presently, the following features have been implemented:

- Two types of searching exist.
  - Basic search enables the user to search a single database for a single attribute. Although it is very simple, this is very common search method.
  - Concurrent basic search enables the user to perform basic search on many databases at the same time. This only makes sense if it's run on attribute compatible databases. For instance, one cannot search two different databases that don't have at least one attribute in common.
- Software has multilingual interface. Translating it to other languages is just the matter of translating one text file.
- Default character encoding is UTF-8. In this way textual information from all over the world can be displayed correctly.
- Configuration file is a XML document. This makes our software faster and easier to customize.

2.1.2. *The server*. Software that encapsulates Z39.50 server with connection to a relational database is called zSQLgate [3]. It uses DBI library (universal SQL-accessible database interface) from Perl. Database management system is MySQL [9]. Here are its key features:

- Each physical relational database is associated with one server. Logical tables inside a database however, can be joined to form many Z39.50 databases.
- Attribute set bib1 is supported.
- Record type XML is supported.

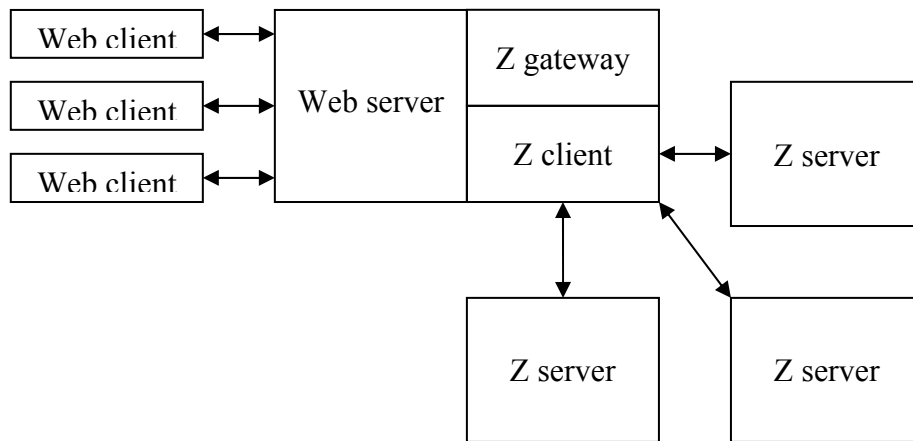Secure connections are supported via SSL. This is implemented in YAZ toolkit.
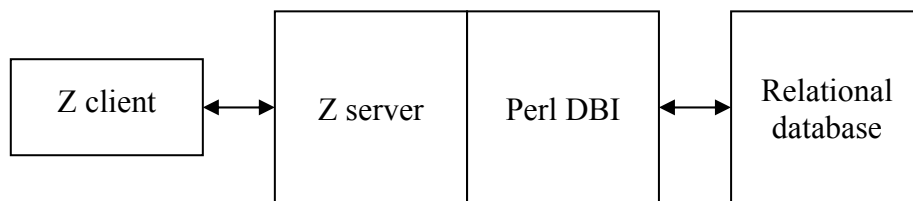
Figure 1. Web gateway

Figure 2. zSQLgate server

2.2. Legal. All the software is completely free of charge except zSQLgate. This software has a special license. At this time, our intention is not to distribute this software. We only intend to use it internally.

Software	License	Free of charge
Linux, including PHP and Perl with DBI	GNU GPL1	✓
YAZ Toolkit, including PHPYAZ	YAZ	✓
MySQL	GNU GPL	✓
zSQLgate	?	

Table 1. Software licenses

2.3. Conclusion. This software consists of many software packages. Most of them are free of charge. We now have an operative gateway for searching indefinite number of data sources. There are also many ways to improve this software, just to name a few:

---

[1] Actually, PHP and Perl have their own licenses which are not GNU GPL but they have very few restrictions which don't concern our work

- Searches can be improved. Logical operations can be applied on search terms to create complex queries.
- Return-records have all to be standardized. At this moment, each syntax is processed and displayed separately; their representations differ.
- There are means of representing and searching multimedia information (sound, pictures) using specific Z39.50 profiles. Such features have yet to be tested. Good example for it can be found at American Library of Congress [12].

Although we didn't have any legal consulting, we find no legal obstacles for running of this software.

Important note: references [2] and [5] were only temporarily available.

References

1. Željko Pajkić, *On The Standard Z39.50*, NCD Review 3 (2003), 26–35, (in Sebian)
2. Library of Mathematical Institute, Belgrade Z39.50 Gateway – http://toledo.mi.sanu.ac.yu/zgate
3. zSQLgate – http://sql.z3950.org
4. YAZ Z39.50 Toolkit and PHPYAZ – http://www.indexdata.dk
5. Library of Mathematical Institute, Belgrade Z39.50 Server – host:toledo.mi.sanu.ac.yu port:3950
6. Infodata Z39.50 Target Directory– http://www.indexdata.dk/targettest/
7. PHPYAZ reference – http://www.php.net/manual/en/ref.yaz.php
8. PHP Hypertext Preprocesor – http://www.php.net/
9. MySQL AB – http://www.mysql.com/
10. Webclarity Software Inc. – http://www.webclarity.info/
11. Apache Software – http://www.apache.org/
12. Library of Congress, Washington, http://www.loc.gov/

JEDNA IMPLEMENTACIJA Z39.50 SISTEMA

Sažetak: U radu je predstavljena jedna implementacija Z39.50 servera i odgovarajućeg gateway-a. Ovaj programski sistem omogućava pretraživanje distribuiranih i udaljenih baza podataka. Njegov interfejs je web-baziran. Implementacija je bazirana na slobodno raspoloživom softveru.
Ključne reći: Z39.50, web, gateway, server, digitalizacija, distribuirane baze podataka

zeljko_pajkic@web.de
dejan@mi.sanu.ac.yu
zorano@mi.sanu.ac.yu