Workshop on
Information Resource Management
13th-15th March, 2002
DRTC, Bangalore

Paper: EA

A Z39.50 Client for Retrieving MARC21 Records in Batch Mode

ARD Prasad

Documentation Research and Training Centre

Indian Statistical Institute

Bangalore

email: *ard@isibang.ac.in*

Abstract

The paper presents a Perl program for downloading MARC21 records from a Z3950 compliant target in a batch mode. Also discusses some of the open source software for Z39.50 protocol required for running the Perl program.

1 Introduction

One of the advantages of resources sharing is that not only the information, but also the bibliographic information can be shared, provided the libraries in the network follow common bibliographic standards. With the advent of Internet many libraries are providing access to their Online Public Access Catalogs (OPAC). For example, one can access the Library of Congress OPAC and retrieve data in MARC21 format. If you use the ISBN of a document recently acquired for your library as the search key on LC site, and if the document is found, you can retrieve the data and export it to a local database of your library. In a way, you can avoid cataloguing and data entry.

However, the disadvantage with this approach is that as LC site allows interactivity, you should enter the ISBNs one by one and download each one of them in a separate file and finally you should merge all the files before you go for exporting the data to your local database. Ideally, we should be able to put all the ISBNs of the documents acquired by a library in a file and the system should somehow access the LC site and retrieve all the data that is found. In other words, the access and retrieval should be carried out in batch mode.

In order to solve this problem, the Z39.50 protocol can be used as many of the library OPACs are Z39.50 protocol compliant targets (servers). We only need to develop a Z39.50 protocol compliant client. Fortunately, many of the tools required to build such a client are open source software. This paper presents a Z39.50 client using a Perl program, which runs on Intel based RedHat Linux 7.1.

2 Requirements

2.1 PERL

A Perl compiler is needed and RedHat Linux 7.1 comes with Perl 5.6.0 version. If one wishes to get the latest compiler it can be found at ma ny sites on the Internet.

2.2YAZ TOOLKIT

Yaz toolkit is meant for C and C++ programmers to develop Z39.50 clients and servers and also ILL (Inter Library Loan Protocol). The toolkit is available for Linux, many Unix platforms and MS-windows operating systems. In fact, the Bookwhere 2000 (a Z39.50 client) software was developed using Yaz toolkit. Yaz was developed by Index Data for Europagate project. Yaz toolkit can be downloaded from *www.indexdata.dk/yaz/.* It is better to download the 'yaz-1.8.5.tar.gz' file rather than the 'rpm' file.

Steps in loading Yaz Toolkit

1. Copy the above file into a directory of you choice.
2. Unzip the file with the command ‘gunzip yaz-1.8.5.tar.gz’
3. Untar the file with the command ‘tar xvf yaz-1.8.5.tar’
4. Enter the directory with the command ‘cd yaz-1.8.5’
5. Enter the command ‘./configure’
6. Enter the command ‘make’
7. Enter the command ‘make install’ (requires root password)

2.3 EVENT.PM

This Perl module is essential for event handling and the file ‘Event-0.85.tar.gz’ can be downloaded from the site *http://www.cpan.org/modules/by-module/NetServer/JPRIT/.*

After unzipping and untarring the file, issue the following commands.

1. cd Event-0.85
2. perl Makefile.PL
3. make
4. make install (requires root password)

2.4 NET::Z3950

These are a set of Perl modules essential for the Z39.50 protocol developed by Mark Taylor and the required file ‘*Net-Z3950-0.27.tar.gz*’ can be obtained from ‘perl.z3950.org/’. After unzipping and untarring the file follow a similar procedure as explained under the section “*Event.pm*”.

2.5 MARC.PM

The perl module that can handle MARC format files is ‘MARC-1.14.tar.gz’ and can be obtained from the site ‘*marcpm.sourceforge.net/*’. After unzipping and untarring the file follow a similar procedure as explained under the section “*Event.pm*”.

3 Z39.50 Client

Following is the Perl code which can retrieve MARC21 records from a given Z39.50 target. The program uses the Library of Congress site i.e. 128.151.244.116 on the port 7090 and the database ‘*voyager*’. The ISBNs are entered in the file ‘*isbns*’. The program downloads the MARC21 records in ISO2709 format in a file called ‘out’ and writes the ISBNs not found in a file called ‘err’.

3.1 LISTING OF THE PERL PROGRAM

```
#!/usr/bin/perl -w
use Net::Z3950;

open(INFILE, "isbns");
@query = <INFILE>;
close INFILE;
$totalisbns = @query;
$totalfound = 0;

open(OUTFILE, ">out");
open(ERRFILE, ">err");

# Establish connection

$conn =  new Net::Z3950::Connection('128.151.244.116', 7090,
databaseName => 'voyager')
   or die "Can not connect $!";
$conn->option(querytype => 'ccl2rpn');
$conn->option(elementSetName => "f");
$conn->option(preferredRecordSyntax =>
Net::Z3950::RecordSyntax::USMARC);

# Take each query from ISBN file

foreach $q (@query)
  {
  chomp($q);
#  $q =~ s/^/isbn=/;
  $rs = $conn->search($q);

  if($rs->size()==0)
    {
    print "Could not find ISBN: ", $q, "\n";
    print ERRFILE $q, "\n";
    $conn->close();

# You can change the IP address, Port number and database name
# in the following line

    $conn =  new Net::Z3950::Connection('128.151.244.116', 7090,
databaseName => 'voyager')
       or die "Can not connect  $!" ;
    $conn->option(querytype => 'ccl2rpn');
    $conn->option(elementSetName => "f");
    $conn->option(preferredRecordSyntax =>
Net::Z3950::RecordSyntax::USMARC);
    }
  else
    {
     print "Getting Record for ISBN: ", $q, "\n";
     $rec = $rs->record(1);
     $record = $rec->rawdata();
     if($record =~ /^Failed/)
```

```
        {
        print "Found ", $rs->size(), "records, BUT ", $record, "\n";
      print "adding ", $q, " to not found list\n";
      print ERRFILE $q, "\n";
        next;
        }
  $totalfound++;
#  Uncomment the following line, if you do not want CDS/ISIS export
format
#    print OUTFILE $record;

#  Comment the following upto 'end of for' if you do not want CDS/ISIS
format

     $len = length($record);
     for($i=0; $i < $len; $i=$i+80)
        {
      $s = substr($record, $i, 80);
      $recsep = chr(29);
      $fldsep = chr(30);
      $subsep = chr(31);
      $s =~ s/$recsep/#/g;
      $s =~ s/$fldsep/#/g;
      $s =~ s/$subsep/^/g;
      print OUTFILE "$s\n";
      }  # end of for
    }      #end of else
  }        # end of foreach

print "Total $totalfound / $totalisbns reocrds printed\n";
close OUTFILE;
close ERRFILE;
```

4 Conclusion

The program can generate the output in ISO2709 format i.e. all the records in one line, alternatively, it can generate output in CDS/ISIS format which contains a new line character after every 80 characters and the subfiled identifier as '^' instead of ASCII 31. Once the data is obtained, one can search another OPAC using the list of ISBNs not found (stored in file 'err'). The program has been tested by importing the data into WINISIS. A list of Z39.50 compliant servers can be obtained by searching 'google' or some other search engine by entering the key as 'list of Z3950 targets'. One should get the IP number or IP address, Port number and the database name for the program to work.