

Network Working Group  
Request for Comments: [2056](#)  
Category: Standards Track

R. Denenberg  
Library of Congress  
J. Kunze  
University of California, San Francisco  
D. Lynch  
SilverPlatter Information Ltd.  
Editors  
November 1996

## Uniform Resource Locators for Z39.50

### Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

### 1. Introduction

Z39.50 is an information retrieval protocol that does not fit neatly into a retrieval model designed primarily around the stateless fetch of data. Instead, it models a general user inquiry as a session-oriented, multi-step task, any step of which may be suspended temporarily while the server requests additional parameters from the client before continuing. Some, none, or all of these client/server interactions may require participation of the client user, depending only on the client software (the protocol itself makes no such requirements).

On the other hand, retrieval of "well-known" data may be performed in a single step, that is, with a degenerate Z39.50 session consisting of exactly one protocol search request and response. Besides the basic search sub-service, there are several ancillary sub-services (e.g., Scan, Result Set Delete). Among the functions covered by combinations of the sub-services, two core functions emerge as appropriately handled by two separate URL schemes: the Session URL and the Retrieval URL.

Using two schemes instead of one makes a critical distinction between a Z39.50 Session URL, which opens a client session initialized for interactive use by the user, and a Z39.50 Retrieval URL, which opens and closes a client session to retrieve a specific information item. Making this distinction at the scheme level allows the user interface to reflect it on to the user, without requiring the user interface to

parse otherwise opaque parts of the URL (consistent with current practice).

## **2. Some Basic Concepts**

This section briefly describes the usage of Z39.50-specific terminology within the URL definitions below: specifically, the terms database, elementset, recordsyntax, and docid.

The Z39.50 protocol specifies various information retrieval operations, the two most basic of which are Search and Present. In a Search operation a client provides search criteria and indicates a database (or several databases) on the server to search. The essential result of a Search operation is that a result set is created at the server, consisting of pointers to the selected database records.

Z39.50 models database records, abstract database records, and retrieval records. A database record is a unit of information in a database, represented in a data structure local to the server. An abstract database record is an abstract representation of that information, where the client and server share a common understanding of the representation. This allows logical elements to be addressed and selected for transfer, via an element set specification, or, as used below, an "elementset". A retrieval record is the set of selected elements packaged in an exportable structure, by the application of a "recordsyntax".

Thus a Search operation results in entries pointing to database records; via a Present operation, a client requests a retrieval record, corresponding to a database record, corresponding to an entry in the result set. The client indicates the composition and format of the retrieval record by specifying an elementset and recordsyntax, respectively.

A special case of a Z39.50 search is a "known-item" search, when a client intends that a search identify a single, known database record, or "document" (for purposes of illustration, assume that a database record corresponds to a document), and further, the client knows an identifier for the document that can be used to effect this known-item search. In this case, this identifier is often referred to as a document identifier, or "docid".

### 3. The Z39.50 Session URL

The Z39.50 Session URL may be informally described as providing the mechanism to switch the user to a Z39.50 client application.

- Host is required.
- Port is optional, and defaults to 210.
- All other parameters are optional.
- The Z39.50 client will start a session to the specified host/port (alternatively, it need not explicitly start a session, but may instead utilize an already open session to the same host/port).
- A database must be included if docid is included.
- If docid is included, the client will perform the specified search (in the same manner as for the retrieval URL, specified below).
- If docid is not included, and other parameters (besides host/port) are specified, the client may use those parameters as "hints". Various clients may choose to treat them as requirements, or as preferences, or ignore them.
- In any case (whether a search is performed or not), the client will leave the Z39.50 session open for the user, to do retrievals, new searches, etc. (This is the main distinction from the Retrieval URL which leaves it up to the client whether or not to keep the Z39.50 session open.)

### 4. The Z39.50 Retrieval URL

The Z39.50 Retrieval URL is intended to allow a Z39.50 session to be used as a transparent transfer mechanism to retrieve a specific information object. A Z39.50 client uses information in the URL to formulate a Search Request. The server's Search Response indicates how many records match the Request. If the number of matching records does not equal one, the retrieval is considered unsuccessful, and the client application's behavior is not defined. If the number of matching records equals one, the server may have included the desired record in the Search Response. If not, the client requests transmission of the record with a Present Request. After the client has received the specified record it may close the Z39.50 session immediately, or keep it open for subsequent retrievals.

- Host is required.
- Port is optional, and defaults to 210.
- A database is required.
- The meaning of a retrieval URL with no docid is undefined.
- The docid is placed into a type-1 query, as the single term, in the general format (tag 45), using the Bib-1 attribute set, with a Use attribute value of docid, and a structure attribute of URx. The docid string is server-defined and completely opaque to the client.

- If element set name (esn) is not specified, it is the client's choice. If esn is specified, it should be used either in the Search request for the value of small- and/or medium-set-element-set-names or in a Present request following a Search. These terms and their use are defined within the Z39.50 Standard [2].
- If record syntax (rs) is not specified, it is the client's choice. If one or more record syntaxes are specified, the client should select one (preferably the first in the list that it supports) and use it in a Search or Present request as the value of PreferredRecordSyntax.

## 5. BNF for Z39.50 URLs

The Z39.50 Session and Retrieval URLs follow the Common Internet Scheme Syntax as defined in RFC [1738](#), "Uniform Resource Locators (URL)" [1]. In the definition, literals are quoted with "", optional elements are enclosed in [brackets], "|" is used to designate alternatives, and elements may be preceded with <n>\* to designate n or more repetitions of the following element; n defaults to 0.

```
z39.50url      = zscheme "://" host [":" port]
                 [ "/" [database *["+" database]
                        ["?" docid]]
                 [";esn=" elementset]
                 [";rs=" recordsyntax *["+" recordsyntax]]]

zscheme        = "z39.50r" | "z39.50s"
database       = uchar
docid          = uchar
elementset     = uchar
recordsyntax   = uchar
```

Future extensions to these URLs will be of the form of  
[;keyword=value].

The following definitions are from RFC [1738](#). Between the Internet Draft version and RFC [1738](#) two relevant changes were made: '=' was moved from the <extra> character class to <reserved>, and <national> was removed from the alternatives in <unreserved>. Neither <national> nor <punctuation> is referred to in this document nor in RFC [1738](#).

---

```
lowalpha      = "a" | "b" | "c" | "d" | "e" | "f" | "g" | "h" |
               "i" | "j" | "k" | "l" | "m" | "n" | "o" | "p" |
               "q" | "r" | "s" | "t" | "u" | "v" | "w" | "x" |
               "y" | "z"
hialpha       = "A" | "B" | "C" | "D" | "E" | "F" | "G" | "H" | "I" |
               "J" | "K" | "L" | "M" | "N" | "O" | "P" | "Q" | "R" |
               "S" | "T" | "U" | "V" | "W" | "X" | "Y" | "Z"

alpha         = lowalpha | hialpha
digit         = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" |
               "8" | "9"
safe          = "$" | "-" | "_" | "." | "+"
extra         = "!" | "*" | "'" | "(" | ")" | ","
national      = "{" | "}" | "|" | "\" | "^" | "~" | "[" | "]" | "`"
punctuation   = "<" | ">" | "#" | "%" | "<">

reserved      = ";" | "/" | "?" | ":" | "@" | "&" | "="
hex           = digit | "A" | "B" | "C" | "D" | "E" | "F" |
               "a" | "b" | "c" | "d" | "e" | "f"
escape        = "%" hex hex
unreserved    = alpha | digit | safe | extra
uchar         = unreserved | escape
xchar         = unreserved | reserved | escape
digits        = 1*digit
```

## 6. Security Considerations

The two Z39.50 URL schemes are subject to the same security implications as the general URL scheme [1], so the usual precautions apply. This means, for example, that a locator might no longer point to the object that was originally intended. It also means that it may be possible to construct a URL so that an attempt to perform a harmless idempotent operation such as the retrieval of an object will in fact cause a possibly damaging remote operation to occur.

## 7. Acknowledgements

The Z39.50 Implementors Group contributed the substance of this document.

## 8. References

- [1] Berners-Lee, T., Masinter, L., McCahill, M. (editors), "Uniform Resource Locators (URL)", RFC 1738, December 1994.  
<ftp://ds.internic.net/rfc/rfc1738.txt>
- [2] ANSI/NISO Z39.50-1995, "ANSI Z39.50: Information Retrieval Service and Protocol", 1995. <ftp://ftp.loc.gov/pub/z3950/>

- [3] ANSI/NISO Z39.50-1992, "ANSI Z39.50: Information Retrieval Service and Protocol", 1992.  
<ftp://ftp.cni.org/pub/NISO/docs/Z39.50-1992/www/Z39.50.toc.html>  
(also available in hard copy from Omnicom Information Service, 115 Park St., SE, Vienna, VA 22180).

## 9. Editors' Addresses

Ray Denenberg  
Library of Congress  
Collections Services  
Network Development/MSO  
Washington DC 20540

Phone: (202) 707-5795  
Fax: (202) 707-0115  
EMail: ray@rden.loc.gov

John A. Kunze  
Center for Knowledge Management  
University of California, San Francisco  
530 Parnassus Ave, Box 0840  
San Francisco, CA 94143-0840

Phone: (415) 502-6660  
Fax: (415) 476-4653  
EMail: jak@ckm.ucsf.edu

Denis Lynch  
SilverPlatter Information Ltd.  
10 Barely Mow Passage  
Chiswick, London W4 4PH  
U.K.

Voice: +44 (0)181-995-8242  
Fax: +44 (0)181-995-5159  
EMail: DenisL@SilverPlatter.com

### **Appendix. Examples of Z39.50 URLs**

A basic Z39.50 session URL that a client might use to open a connection to the MELVYL union catalog "cat" at the University of California is

```
z39.50s://melvyl.ucop.edu/cat
```

A URL that would open the MELVYL magazine database just long enough to fetch an article from volume 30, number 19 of a hypothetical periodical might look like

```
z39.50r://melvyl.ucop.edu/mags?elecworld.v30.n19
```

As a final example, here is another retrieval URL that a client could use to request a full record (element set "f") in the MARC syntax from a hypothetical database called TMF at CNIDR:

```
z39.50r://cnidr.org:2100/tmf?bkirch_rules__a1;esn=f;rs=marc
```

As in the previous example, the part of the string after the '?' is determined by the server. In this example, the server is running on non-standard port 2100.

—