Network Working Group
Request for Comments: 1729
Category: Informational

C. Lynch
University of California
Office of the President
December 1994

Using the Z39.50 Information Retrieval Protocol in the Internet Environment

Status of this Memo

This memo provides information for the Internet community. This memo does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Summary

This memo describes an approach to the implementation of the ANSI/NISO Z39.50-1992 Standard for Information Retrieval in the TCP/IP environment which is currently in wide use by the Z39.50 implementor community.

Introduction

Z39.50 is a US national standard defining a protocol for computer-to-computer information retrieval that was first adopted in 1988 [1] and extensively revised in 1992 [2]. It was developed by the National Information Standards Organization (NISO), an ANSI-accredited standards development body that serves the publishing, library, and information services communities. The closely related international standard, ISO 10162 (service definition) [3] and 10163 (protocol) [4], colloquially known as Search and Retrieve or SR, reached full International Standard (IS) status in 1991. Work is ongoing within ISO Technical Committee 46 Working Group 4 Subgroup 4 to progress various extensions to SR through the international standards process. The international standard is essentially a compatible subset of the current US Z39.50-1992 standard. Z39.50 is an applications layer protocol within the OSI reference model, which assumes the presence of lower-level OSI services (in particular, the presentation layer [5]) and of the OSI Association Control Service Element (ACSE) [6] within the application layer.

Many institutions implementing this protocol chose, for various reasons, to layer the protocol directly over TCP/IP rather than to implement it in an OSI environment or to use the existing techniques that provide full OSI services at and above the OSI Transport layer on top of TCP connections (as defined in RFC 1006 [7] and implemented, for example, in the ISO Development Environment

software). These reasons included concerns about the size and complexity of OSI implementations, the lack of availability of mature OSI software for the full range of computing environments in use at these institutions, and the perception of relative instability of the architectural structures within the OSI applications layer (as opposed to specific application layer protocols such as Z39.50 itself). Most importantly, some of these institutions were concerned that the complexity introduced by the OSI upper layers would outweigh the relatively meager return in functionality that they were likely to gain. Thus, for better or worse, the decision was taken to implement the Z39.50 protocol directly on top of TCP (with the understanding that this decision might be revisited at some point in the future).

During 1991-1993, a group of implementing institutions agreed to participate in the Z39.50 Interoperability Testbed project (sometimes referred to by the acronym "ZIT") under the auspices of the Coalition for Networked Information (CNI). Their primary objective was to encourage the development of many interoperable Z39.50 implementations running over TCP/IP on the Internet. By mid-1993, a number of independent Z39.50 implementations were operational and able to interoperate across the Internet.

The Library of Congress, in its role as the Z39.50 Maintenance Agency for NISO, maintains a registry of the implementors [8], which includes members of the Z39.50 interoperability testbed.

This document describes implementation decisions by current implementors of Z39.50 in the Internet environment. These have been proven within the ZIT project and are being used by most of the members of the Z39.50 Implementors' Group (ZIG), an informal group that meets quarterly to discuss implementation and interoperability issues and to develop extensions to the Z39.50 protocol targeted for inclusion in future versions of the standard. Intended as a guide for other implementors who seek to develop interoperable Z39.50 implementations running over TCP/IP, this document focuses on issues related to TCP/IP, and it does not address other potential interoperability problems or agreements that have been reached among the implementors to address these problems. It does include a few notes about extensions to the existing Version 2 protocol that are being used in the implementor community which have interoperability implications. Potential implementors of Z39.50 should subscribe to the Z3950IW LISTSERV [9] to obtain information specific to the Z39.50 protocol and extensions under development as well as details of current implementations.

Except where otherwise noted, the version of Z39.50 discussed here is ANSI/NISO Z39.50-1992, sometimes called Z39.50 Version 2 (the obsolete original version is referred to as Z39.50-1988 or Z39.50 Version 1). The approach defined should also be applicable, perhaps with some minor changes, to future versions of the Z39.50 protocol, and specifically to Version 3 which is currently under development. This document will probably be updated to address new versions of the base Z39.50 protocol as they become stable.

Encoding

The Z39.50 standard specifies its application protocol data units (APDUs) in Abstract Syntax Notation One (ASN.1) [10]. These APDUs include EXTERNAL references to other ASN.1 and non-ASN.1 objects such as those defining record transfer syntaxes to be used in a given application association.

The standard Basic Encoding Rules (BER) [11] are applied to the ASN.1 structures defined by the Z39.50 protocol to produce a byte stream that can be transmitted across a TCP/IP connection. The only restriction on the use of BER to produce this byte stream is that direct, rather than indirect, references must be used for EXTERNAL objects. This is necessary because there is no presentation context in the TCP/IP environment to support indirect reference. A Z39.50 implementation developed according to this specification and running over TCP/IP should produce a valid byte stream according to the Z39.50 standard, in the sense that the same byte stream could be passed to an OSI implementation. However, not all byte streams that can be produced by applying BER to the APDUs specified in the Z39.50 standard in an OSI environment will be legitimate under this specification for the TCP/IP environment; this specification defines a subset of the possible byte streams valid in a pure OSI environment which excludes those using indirect reference for EXTERNAL objects.

All other BER features should be tolerated by Z39.50 implementations running over TCP/IP, including the ability to accept indefinite length encodings, although it is preferable that implementations do not generate such encodings since they have caused problems for some ASN.1/BER parsers. It should also be noted that at least to the best of the author’s knowledge, there are no implementations at present that use ASN.1/BER representations of floating point numbers; instead, integers with scaling factors have been used for these purposes. It should also be noted that Z39.50 version 2 does not really address character set encoding issues; these questions, and their interactions with ASN.1/BER support for multiple character sets, are under active discussion as part of the effort to develop Z39.50 version 3.

Connection

In the Internet environment, TCP Port 210 has been assigned to Z39.50 by the Internet Assigned Numbers Authority [12]. To initiate a Z39.50 connection to a server in the TCP/IP environment, a client simply opens a TCP connection to port 210 on the server and then, as soon as the TCP connection is established, transmits a Z39.50 INIT APDU using the BER encoding of that INIT APDU as described above.

Implementors should be aware that there is a substantial installed base of implementations of the Wide Area Information Server (WAIS) system. The original versions of this software employed Z39.50 Version 1 with some extensions. Z39.50 Version 1 did not use BER encoding and Z39.50 Version 1 INIT APDUs look very different from the INIT APDUs of Z39.50 Version 2. Implementations of Z39.50 should at least be prepared to reject gracefully WAIS-type INIT APDUs. Some implementations recognize such INIT APDUs and revert to the Z39.50 Version 1 variant used in WAIS upon encountering them, thus providing backwards compatibility with the existing base of WAIS clients and; the usual means of checking for a WAIS, as opposed to Z39.50 Version 2, client is to see if the first byte sent on the connection is an ASCII zero, which indicates a WAIS client. (In version 1 of WAIS, bytes 0-9 of the first PDU contain an ASCII packet length; the lower case ASCII string "wais" appears starting at byte 12.) Work is currently underway to specify a WAIS profile for use with Z39.50 version 2 [13]; it is expected that this will be issued as a Z39.50 Applications Profile through the NIST OIW Library Automation Special Interest Group. This profile is expected to be compatible with the layering defined in this RFC.

Service Mappings

The Z39.50 standard maps Z39.50 services onto a variety of association control and presentation layer services. Connection establishment has already been discussed. The other two association control services that are relevant to Z39.50 are ABORT and RELEASE. The mapping of the RELEASE service to a standard TCP CLOSE is straightforward. The Z39.50 protocol itself does not, in the current version, include a Z39.50 CLOSE APDU. When the client has completed its interaction with the server, it calls the IR-RELEASE service, which is directly mapped to association control’s orderly association release. In the TCP/IP environment, the client should simply initiate a TCP CLOSE. The mapping for association abort is more complex, partially because some TCP/IP implementations cannot distinguish a TCP reset from the other side of the connection from other events. To accomplish an abort (that is, a mapping of the IR-ABORT service in the Z39.50 protocol) in the TCP/IP environment, client or server need only terminate the TCP connection either via TCP ABORT or TCP CLOSE.

Real-world implementations need to be prepared to deal with both TCP ABORT and CLOSE anyway, so this approach presents no additional problems, other than the somewhat ambiguous nature of the type of association termination.

It is expected that Z39.50 Version 3 will include a termination service which will involve an exchange of Z39.50 CLOSE APDUs, followed by an association RELEASE (which would presumably, in the Internet environment, be mapped to a TCP CLOSE). This new termination service is expected to support both graceful and abrupt termination. Of course, robust implementations will still need to be prepared to encounter TCP CLOSE or ABORT.

Service mappings for the transmission of data by client and server (to the presentation layer P-DATA service) are trivial: They are simply mapped to TCP transmit and receive operations. TCP facilities such as expedited data are not used by Z39.50 in a TCP environment.

Contexts

At the point when the TCP connection is established on TCP port 210, client and server should both assume that the application context given in Appendices A and B of the Z39.50-1992 standard are in place. These are the ASN.1 definitions of the Z39.50 APDUs and the transfer syntax defined by applying the BER to these APDUs.

Implementations can reasonably expect that the diagnostic set BIB-1 is supported, and, if resource control is being used, the resource report format BIB-1 is supported as well.

In the absence of a presentation negotiation mechanism, clients and servers should be cautious about using alternative attribute sets, diagnostic record formats, resource report formats, or other objects defined by optional EXTERNALs within the Z39.50 ASN.1, such as authentication parameters, unless there is known to be prior agreement to support them. Of course, either participant in an association can reference such an object by object ID in an APDU, but there is no guarantee that the other partner in the association will be able to understand it. Robust implementations should be prepared to encounter unknown or unsupported object IDs and generate appropriate diagnostics. Over time, the default, commonly known pool of object IDs may be expanded (for example, to support authentication parameters).

Implementors should refer to the document [14] issued by the Z39.50 maintenance agency in June 1992 for more details on the assumed contexts and object identifiers.

Record syntaxes present a serious practical problem. In the OSI environment, the partners in a Z39.50 association are assumed to agree, either through presentation negotiation as part of association establishment, or later, dynamically, as part of the PRESENT process (through the use of the alter presentation context function at the presentation layer), on which record syntaxes the two entities commonly know. There is a preferred record syntax parameter that can be supplied by the client to guide this negotiation. A number of registered record syntaxes exist; some are based on ASN.1 and others use formats such as the MARC standard for the interchange of machine readable cataloging records which predate ASN.1, but are widely implemented. In the TCP/IP environment, if the server cannot supply the record in the preferred syntax, it has no guarantee that the client will understand any other syntax in which it might transmit the record back to the client, and has no means of negotiating such syntaxes.

Several proposals have been suggested to solve this problem. One, which will likely be part of Z39.50 Version 3, is to replace the preferred record syntax parameter with a list of prioritized preferred syntaxes supplied by the client, plus a flag indicating whether the server is allowed to substitute a record syntax not on the list provided by the client. The currently proposed ASN.1 for this extension is upwards compatible with Z39.50 Version 2, although the details are still under discussion within the Z39.50 Implementor's Group. As the Version 3 ASN.1 becomes stable in this area, Z39.50 servers are encouraged to accept the extended ASN.1 for generalized preferred record syntax. The extensibility rules for Z39.50 negotiation let clients and servers negotiate the use of Z39.50 Version 2 plus the generalized preferred syntax feature from Version 3. Thus, a client could support the generalized preferred record syntax, propose its use to any server, and, if the server rejects the proposal, revert to the Version 2 preferred syntax feature.

A second alternative (not incompatible with the Version 3 extension) would be to adopt a convention for TCP/IP implementations that the server not return a record in a syntax not on the preferred record syntax list provided by the client. Instead, it would return a diagnostic record indicating that a suitable record transfer syntax was not available. This strategy could be viewed as simply implementing a subset of the Version 3 solution, and should be considered by implementors of servers as a possible interim measure.

Other Interoperability Issues

Version 3 will include an "other" data field in each APDU, which can be used to carry implementation-specific extensions to the protocol. A number of implementations are already employing this field, and interoperable implementations might be wise to include code which at least ignores the presence of such fields rather than considering their presence an error (in contravention of the standard).

References

[1] National Information Standards Organization (NISO). American National Standard Z39.50, Information Retrieval Service Definition and Protocol Specifications for Library Applications (New Brunswick, NJ: Transaction Publishers; 1988).

[2] ANSI/NISO Z39.50-1992 (version 2) Information Retrieval Service and Protocol: American National Standard, Information Retrieval Application Service Definition and Protocol Specification for Open Systems Interconnection, 1992.

[3] ISO 10162 International Organization for Standardization (ISO). Documentation -- Search and Retrieve Service Definition, 1992.

[4] ISO 10163 International Organization for Standardization (ISO). Documentation -- Search and Retrieve Protocol Definition. 1992.

[5] ISO 8822 - Information Processing Systems - Open Systems Interconnection - Connection Oriented Presentation Service Definition, 1988.

[6] ISO 8649 - Information Processing Systems - Open Systems Interconnection - Service Definition for the Association Control Service Element, 1987. See also ISO 8650 - Information Processing Systems - Open Systems Interconnection - Protocol Specification for the Association Control Service Element, 1987.

[7] Rose, M., and D. Cass, "ISO Transport Layer Services on Top of the TCP, Version 3", STD 35, RFC 1006, Northrop Research and Technology Center, May 1987.

[8] Registry of Z39.50 Implementors, available from the Z39.50 Maintenance Agency (Ray Denenberg, ray@rden.loc.gov)

[9] To subscribe to the Z39.50 Implementor's Workshop list send the message: SUB Z3950IW yourname to: LISTSERV@NERVM.NERDC.UFL.EDU (or NERVM.BITNET). Current drafts of the Version 3 Protocol document are available through the Library of Congress GOPHER server, MARVEL.LOC.GOV.

[10] ISO 8824 - Information Processing Systems - Open Systems Interconnection - Specifications for Abstract Syntax Notation One (ASN.1), 1987

[11] ISO 8825 Information Processing Systems - Open Systems Interconnection - Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1) 1987

[12] Reynolds, J., and J. Postel, "Assigned Numbers", STD 2, RFC 1700, USC/Information Sciences Institute, October 1994.

[13] WAIS Profile of Z39.50 Version 2, Revision 1.4, April 26, 1994, available from WAIS Inc.

[14] Registration of Z39.50 OSI Object Identifiers (Z39.50-MA-024), available from the Z39.50 Maintenance Agency (Ray Denenberg, ray@rden.loc.gov).

Security Considerations

This document does not discuss security considerations. However, it should be noted that the Z39.50 protocol includes mechanisms for authentication and security that implementors should review.

Author's Address

Clifford A. Lynch
University of California, Office of the President
300 Lakeside Drive, 8th Floor
Oakland, CA 94612-3550

Phone: (510) 987-0522
EMail: clifford.lynch@ucop.edu