Network Working Group
Request for Comments: 1625
Category: Informational

M. St. Pierre
WAIS, Inc.
J. Fullton
CNIDR
K. Gamiel
CNIDR
J. Goldman
Thinking Machines Corp.
B. Kahle
WAIS, Inc.
J. Kunze
UC Berkeley
H. Morris
WAIS, Inc.
F. Schiettecatte
FS Consulting
June 1994

WAIS over Z39.50-1988

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.

1. Introduction

The network publishing system, Wide Area Information Servers (WAIS), is designed to help users find information over a computer network. The principles guiding WAIS development are:

1. A wide-area networked-based information system for searching, browsing, and publishing.
2. Based on standards.
3. Easy to use.
4. Flexible and growth oriented.

From this basis, a large group of developers, publishers, standards bodies, libraries, government agencies, schools, and users have been helping further the WAIS system.

The WAIS software architecture has four main components: the client, the server, the database, and the protocol. The WAIS client is a user-interface program that sends requests for information to local or remote servers. Clients are available for most popular desktop environments. The WAIS server is a program that services client

requests, and is available on a variety of UNIX platforms. The server generally runs on a machine containing one or more information sources, or WAIS databases. The protocol, Z39.50-1988, is used to connect WAIS clients and servers and is based on the 1988 Version of the NISO Z39.50 Information Retrieval Service and Protocol Standard. The goal of the WAIS network publishing system is to create an open architecture of information clients and servers by using a standard computer-to-computer protocol that enables clients to communicate with servers.

WAIS development began in October 1989 with the first Internet release occurring in April 1991. From the beginning, WAIS committed to use the Z39.50-1988 standard as the information retrieval protocol between WAIS clients and servers. The implementation is still in use today by existing WAIS clients and servers resulting in over 50,000 users of Z39.50-1988 on the Internet.

2. Purpose

The purpose of this memo is to initiate a discussion for a migration path of the WAIS technology from Z39.50-1988 Information Retrieval Service Definitions and Protocol Specification for Library Applications [1] to Z39.50-1992 [2] and then to Z39.50-1994 [3]. The purpose of this memo is not to provide a detailed implementation specification, but rather to describe the high-level design goals and functional assumptions made in the WAIS implementation of Z39.50-1988. WAIS use of Z39.50-1992 and Z39.50-1994 standards will be the subject of future RFCs.

3. Historical Design Goals of WAIS

As an aid to understanding the original WAIS implementation and its use of Z39.50-1988, the historical design goals of WAIS are presented in this section. Included with each goal is a brief description of the assumptions used to meet these design goals.

1. Provide users access to bibliographic and non-bibliographic information, including full-text and images.

Because Z39.50-1988 grew out of the bibliographic community, additional assumptions with the protocol were required to serve non-bibliographic information. They were also necessary to serve documents existing in multiple formats (e.g., rtf, postscript, gif, etc.).

2. Keep the client/server interface simple and independent of changes in the functionality of the server.

To achieve this, the text string entered by the user was transmitted to the server without parsing the string into a Type-1 RPN (reverse-polish notation) query, as is common for bibliographic applications. Instead WAIS defined a new Type-3 query containing the text string. In this way, knowledge of the Z39.50 Attributes supported by the server was no longer required by the client or the user, as is true of many existing Z39.50 implementations. In addition, the client software did not require modification to support the evolving functionality of the server.

3. Provide relevance feedback capability.

Relevance feedback is the ability to select a document, or portion of a document, and find a set of documents similar to the selection. WAIS included documents used in relevance feedback as part of the Type-3 query.

4. Permit the server to operate in a stateless manner.

A WAIS server was designed to be "stateless", meaning that search result sets were not stored by the server. In Z39.50 terms, the server exercised its right to unilaterally delete a result set as soon as it sent the search response. For this reason, the Present Facility of Z39.50 was not used, and retrievals were performed using the Search Facility. Relaxing this constraint in future implementations may prove the most prudent path.

5. Provide the ability for a client to retrieve documents in pieces.

Because retrieval of a portion of a document could be done several ways with Z39.50-1988, specific assumptions were made to implement this functionality. Accessing a portion of a document was required for both retrieval and for relevance feedback.

6. Run over TCP.

The Z39.50-1988 standard was designed to run in the application layer using the presentation services provided by the Open Systems Interconnection (OSI) Reference Model. Due to the popularity of TCP/IP and the Internet, WAIS was designed to run over TCP. Use of Z39.50 over TCP is described in [4].

4. WAIS Implementation of Z39.50-1988

By working with the Z39.50 Implementors Group (ZIG), the WAIS developers used a recommended subset of Z39.50-1988 and specific assumptions to fulfill its requirements. Over time, many of these

requirements have then gone into the definition of subsequent versions of Z39.50. As new requirements become apparent, WAIS will document any additional assumptions and work with the ZIG in developing extensions.

WAIS supported the Init and Search Facilities of Z39.50-1988. Both search and retrieval were implemented using the Search Facility, as described in this section.

Search was initiated by the client with a Search Request APDU (Application Protocol Data Unit) using a Type-3 query. The query contained two main fields:

1. The "seed words", or text, typed by the user.
2. A list of document objects, where a document object is a full document, or portion thereof, to be used in relevance feedback. Each document object contains a document identifier (Doc-ID) [5], type, chunk-code, and start and end locations. The Doc-ID and type specify the location and format, respectively, of the document. The chuck-code determines the unit of measure for the start and end locations. Examples of chunk-codes used include byte, line, paragraph, and full document. If the chunk code is a full document, the start and end locations are ignored.

A Search Response APDU returned by the server contained a relevance ranked list of records, or WAIS Citations. A WAIS Citation refers to a document on the server. Each WAIS Citation contains the following fields:

1. Headline - a set of words that convey the main idea of the document.
2. Rank - the numerical score of the document based on its relevance to the query, normalized to a top score of 1000.
3. List of available formats - e.g. text, postscript, tiff, etc.
4. Doc-ID - the location of the document.
5. Length - the length of the document in bytes.

The number of WAIS Citations returned was limited by the preferred message size negotiated during the Init.

Retrieval of a document was initiated by the client with a Search Request APDU using a Type-1 query. The query contained up to four terms:

1. Term: Doc-ID
   Use Attribute: system-control-number code = "un"
   Relation Attribute: equal code = "re"

```
      2. Term: the requested document format
         Use Attribute: data-type                  code = "wt"
         Relation Attribute: equal                 code = "re"
      3. Term: the start location
         Use Attribute: paragraph, line, byte      code = "wp", "wl",
                                                          "wb"
         Relation Attribute: greater-than-or-equal code = "ro"
      4. Term: the end location
         Use Attribute: paragraph, line, byte      code = "wp", "wl",
                                                          "wb"
         Relation Attribute: less-than             code = "rl"
```

Because full-text and images were often larger in size than the receive buffer of the client, clients were designed to optionally retrieve documents in chunks, specifying the start and end positions of the chunk in the query. An example of a fully-specified retrieval query is:

```
query = ( ( use = "un", relation = "re", term = <Doc-ID> )
          AND
          ( use = "wt", relation = "re", term = postscript )
          AND
          ( use = "wb", relation = "ro", term = 0 )
          AND
          ( use = "wb", relation = "ro", term = 2000 )
        )
```

A retrieval response was issued by the server with a Search Response APDU. In this case a single record corresponding to the requested document, or portion thereof, was returned in the specified format.

5. Security Considerations

Security issues are not discussed in this memo.

6. 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 Z30.50-1992 (version 2) Information Retrieval Service and Protocol: American National Standard, Information Retrieval Application Service Definition and Protocol Specification for Open Systems Interconnection, 1992.

[3] Z39.50 Version 3: Draft 8", October 1993. Maintenance Agency Reference: Z39.50MA-034.

[4] Lynch, C., "Using the Z39.50 Information Retrieval Protocol in the Internet Environment", Work in Progress, November 1993.

[5] "Document Identifiers, or International Standard Book Numbers for the Electronic Age", Brewster Kahle, Thinking Machines Corporation, see URL=<ftp://wais.com/pub/protocol/doc-ids.txt>, September 1991.

7. Authors' Addresses

Margaret St. Pierre
WAIS Incorporated
1040 Noel Drive
Menlo Park, California 94025

Phone: (415) 327-WAIS
Fax: (415) 327-6513
EMail: saint@wais.com

Jim Fullton
Clearinghouse for Networked Information
Discovery & Retrieval
3021 Cornwallis Road
Research Triangle Park, North Carolina 27709-2889

Phone: (919)-248-9247
Fax: (919)-248-1101
EMail: jim.fullton@cnidr.org

Kevin Gamiel
Clearinghouse for Networked Information
Discovery & Retrieval
3021 Cornwallis Road
Research Triangle Park, North Carolina 27709-2889

Phone: (919)-248-9247
Fax: (919)-248-1101
EMail: kevin.gamiel@cnidr.org

Jonathan Goldman
Thinking Machines Corporation
1010 El Camino Real, Suite 310
Menlo Park, California 94025

Phone: (415) 329-9300 x229
Fax: (415) 329-9329
EMail: jonathan@think.com

Brewster Kahle
WAIS Incorporated
1040 Noel Drive
Menlo Park, California 94025

Phone: (415) 327-WAIS
Fax: (415) 327-6513
EMail: brewster@wais.com

John A. Kunze
UC Berkeley
289 Evans Hall
Berkeley, California 94720

Phone: (510) 642-1530
Fax: (510) 643-5385
EMail: jak@violet.berkeley.edu

Harry Morris
WAIS Incorporated
1040 Noel Drive
Menlo Park, California 94025

Phone: (415) 327-WAIS
Fax: (415) 327-6513
EMail: morris@wais.com

Francois Schiettecatte
FS Consulting
435 Highland Avenue
Rochester, New York 14620

Phone: (716) 256-2850
EMail: francois@wais.com