Use of Z39.50 for Search and Retrieval of Scientific and Technical Information

Les Wibberley

Chemical Abstracts Service
les.wibb@cas.org

Abstract

A common perception of Z39.50 is that it defines a simple information retrieval standard for bibliographic data. But in its latest version, Z39.50-1995, the standard has evolved into a rich set of interoperable services that can be used for client/server-based search and retrieval within literally any information discipline.

This paper addresses the use of Z39.50 for search and retrieval of scientific and technical information. This topic is explored within the context of an implementation of Z39.50 by Chemical Abstracts Service (CAS). The paper explores some of the practical implementation issues encountered, the solutions applied, and lessons learned.

Introduction

Z39.50 is an American National Standard that specifies an interoperable protocol and services for information search and retrieval. The Z39.50 protocol specifies formats and procedures governing the exchange of messages between a client and server, enabling the client to request that the server search databases for information that meets specified criteria, and to retrieve some or all of the identified information.

This paper addresses the use of Z39.50 for search and retrieval of scientific and technical information. It explores some of the practical implementation issues encountered, the solutions applied, and lessons learned. Specific topics include the use of Attribute Sets, Record Syntaxes, Element specification, Segmentation, OtherInformation, and Extended Services. These topics are explored within the context of an implementation of Z39.50 by Chemical Abstracts Service (CAS).

A brief history of Z39.50

The Z39.50 protocol was originally proposed in 1984 for search and retrieval of bibliographic information. The first version of Z39.50 was prepared by a committee of the National Information Standards Organization (NISO), and was approved as an ANSI standard in 1988 [1]. Early implementations of Z39.50-1988 included WAIS (Wide Area Information Servers) and OCLC systems. Within this paper, Z39.50-1988 will be referred to as "V1."

As interest in Z39.50 broadened, the Z39.50 Implementors Group (ZIG) was established in 1990. The formation of the ZIG was a positive step for Z39.50, since it allowed new versions of the standard to be guided, driven, and defined by the needs and experience of implementors. This lent a practical balance to the academic and theoretical viewpoints that have traditionally influenced standards.

Enhancements proposed by Z39.50 implementors were coupled with changes necessary to align Z39.50 with its international counterpart standard, ISO 10162/10163 [2]. This work led to the second version of Z39.50, approved as ANSI standard Z39.50-1992 [3]. The improved interoperability and functionality of this "V2" standard triggered a large number of successful implementations.

Development of the third version of Z39.50 began in late 1991. Several major enhancements and extensions were proposed by implementors for this third version, to support a wider scope of information retrieval activities. From December 1991 through September 1994, a progression of "V3" drafts was developed by the Z39.50 Maintenance Agency, based on ZIG proposals. Each draft underwent careful scrutiny by implementors, and was discussed at length over the ZIG electronic mail list and at the ZIG meetings. Z39.50-1995 [4] was balloted in the fall of 1994. At the time of this paper's writing, June 1995, all ballot objections and comments had been resolved, and final approval of Z39.50-1995 was underway. Since one of the goals of Z39.50-1995 is to support interoperability with Z39.50-1992, it includes specification of both "V2" and "V3" of the protocol.

CAS interest in Z39.50

Chemical Abstracts Service (CAS) is a world leader in scientific and technical information, with heavy concentration on chemistry-related sciences [5]. In additional to its traditional publishing and CD-ROM products, CAS builds and licenses scientific and technical databases, and provides online access to these and other licensed databases through STN International, the Scientific and Technical Information Network. CAS also develops and licenses search and retrieval software for accessing scientific and technical databases.

As a member of NISO, CAS has tracked and voted on the Z39.50 standard since its inception. CAS interest in

Z39.50 was due to its potential for providing a single robust, standard, interoperable protocol for search and retrieval. Since Z39.50 is based on the client/server model, the user interface, protocol, server, search engine, and database management components can each be treated as independent modules, providing greater architectural flexibility. Like other commercial online services, CAS has traditionally used a proprietary user command language and protocol for accessing its databases and services. Z39.50 appeared to be a potential candidate as the protocol for the next generation of client/server systems, offering both expanded functionality and interoperability.

Z39.50-1988 was too limited to be used for interoperability between sophisticated information retrieval systems, especially in a commercial context. However, by 1991, increasing interest in Z39.50 indicated that it might develop into a widely supported standard within the industry, especially if it could be enhanced to support a broader context of information retrieval. At that point, CAS joined the ZIG and became an active participant in the development of both V2 and V3 of Z39.50.

Z39.50 Implementation Issues

CAS started an experimental implementation of Z39.50 in 1991, in order to evaluate its potential for providing interoperable access to scientific and technical (scitech) information. In the process of this evaluation, several issues were identified.

1. Protocol scope and extensibility

Initially, Z39.50 would be used to provide access to existing databases using an existing search and retrieval system. It would therefore be necessary to build a gateway to translate between the Z39.50 protocol and the existing search system's protocol.

The first issue encountered was the fact that the functionality defined by Z39.50 V2 was a small subset of the functionality of the search and retrieval system to which it provided access. And the protocol was not extensible to support additional functionality, without sacrificing interoperability.

There was a lack of extensibility at two levels. First, there was no way to add supplementary data to the Protocol Data Units (PDUs) defined for a given service within the protocol. And secondly, there was no interoperable way to add supplementary services to the protocol.

2. Attribute Set limitations

Using the Z39.50 protocol, a client sends its Query in a Search Request Protocol Data Unit (PDU) to the server, which executes the search and returns a Search Response PDU back to the client, indicating the results of the search operation.

Z39.50 supports several query types within its Search Request PDU, but the one required query type is the Type 1, or Reverse Polish Notation query. The Type 1 query allows any number of search terms to be combined with boolean logic. Each search term can be qualified by one or more Attributes, which identify characteristics of the term, such as: how the term is to be used (Use Attribute); whether it is truncated (Truncation Attribute); how it is structured (Structure Attribute); etc. Z39.50-1992 defined a "bib-1" Attribute Set, which included Use Attributes for most of the common bibliographic fields, but did not include Use Attributes for other information disciplines, such as scitech information.

Therefore, the second major issue encountered was how to identify and characterize scitech query terms within a Z39.50 Search Request.

3. Search service limitations

In addition to the search term Attribute limitations, there were also several other implementation issues related to searching.

a. The V2 Search Request only allowed one Attribute Set to be specified. This precluded the possibility of mixing Attributes from two or more Attribute Sets in the same Search Request.
b. There was no mechanism for specifying the datatype of a given term within a Type 1 query. Since bibliographic searches generally required support for textual terms only, this had not been a major problem. However, for scitech searching, terms might be expressed as integers, real numbers, externally defined structures, binary data, or a variety of other datatypes.
c. There was no way to specify the "units" in which a search term was expressed. For example, a search term specifying a boiling point may need to qualify whether the term value is expressed in degree Celsius, Fahrenheit, or Kelvin.
d. There was no way to send or return additional search information. V2 PDUs contained no extensibility features for carrying information not explicitly defined by the ASN.1 structure of the PDU. Thus, information about the "Type" or "Scope" of a search could not be expressed in the Search Request. And additional information about the search results (such as how many "hits" were found for each term in the query) could not be returned in the Search Response.

4. Element Specification

Within the Z39.50 model, once a search is completed, the matching records from the database are represented by a Result Set, logically containing one "record" per "answer." A Z39.50 client may retrieve one or more records from the Result Set, using the Present service. Within V2, the only way for the client to specify the particular data elements to be retrieved from the Result Set records is to specify an ElementSetName in the Present Request PDU. Only two ElementSetNames were pre-defined by the standard: Brief and Full, both essentially defined by the server. Other ElementSetNames also were to be “primitive” in nature, meaning that a name would designate a pre-defined set of retrieval elements.

Database records within scitech databases tend to contain a large number of data elements, and the particular selection of elements to retrieve may vary considerably, depending on a given client’s or user’s needs. Pre-defining unique ElementSetNames for each combination and permutation of the retrievable elements for each database supported by a given server is a tedious and error-prone task. Furthermore, there was no defined mechanism for the client and server to share a common understanding of these ElementSetNames in a way which would ensure general interoperability.

5. Record Syntax limitations

When a Z39.50 client retrieves Result Set records from the server, it may request a particular packaging of the elements by specifying a preferred Record Syntax in the Present Request PDU. The Record Syntax is intended to ensure preservation of the information content and semantics, as information is transferred from the server to the client.

In the early implementation stages, there was only one type of Record Syntax defined by the Z39.50 standard: USMARC and the various other national MARC formats. MARC is an old but revered format that was designed to carry bibliographic data in a rather limited tagged field format. The MARC Record Syntax has served the library and bibliographic community quite well, but is completely inadequate for carrying scitech information. However, at the time, the scitech community had not defined any appropriate syntaxes or formats that could be used as a general interoperable Record Syntax within Z39.50.

Retrievable scitech information may be available in a wide variety of forms, including character strings, binary strings, integers, real numbers, tables, images, complex data structures, and others. A given scitech element may be available in various forms and may be expressed in different unit systems. The form in which a client retrieves a given scitech element may depend on what it plans to do with it. For example, a client retrieving scitech information may manipulate it, display it, save it, print it, or feed it into local software to process it. The form in which an element is retrieved for display may be very different than the form useful for local processing.

There may be complex hierarchical relationships between individual elements within a retrieved record. A given logical Result Set “record” may actually consist of a complex hierarchy of records, each containing a complex hierarchy of elements. The amount of data retrieved for a single scitech Result Set record (or even a given element within a record) could be very large, and may need to be retrieved in pieces. It is useful to be able to retrieve metadata (data about data) in addition to the data elements themselves. Finally, numeric information such as integers need to be encoded in a standard manner when transferred between computers with different hardware architectures, to ensure data portability.

Few of these needs were addressed by MARC. Therefore, a new, flexible Record Syntax was needed to support retrieval of scitech information. In addition, a simple Record Syntax was needed to retrieve pre-formatted information which a simple client could simply display for the user, without any particular understanding of its content or semantics.

6. Other Retrieval limitations

In addition to the Element Specification and Record Syntax limitations, there were other limitations to the Present service for retrieving scitech information using Z39.50 V2.

a. There was no way to request retrieval metadata.
b. There was no way to request different elements and Record Syntaxes for different databases, when the Result Set was created by a search against multiple databases.
c. There was no way to request a particular form of a given element.
d. There was no way to request particular sub-trees of a hierarchical data structure.
e. There was no flexible way to return large Result Set records. Since there was no concept of record segmentation, a record larger than the message buffer provided by the client simply could not be retrieved.

Approach to the issues

The CAS approaches to these Z39.50 implementation issues fell into three categories:

- o enhancement of the standard through active participation in the ZIG;
- o development of interoperable formats and conventions to support exchange of scitech information;
- o definition of external data structures for use within the standard.

The primary approach was to actively participate in the evolution of the Z39.50 standard, by working with other implementors within the ZIG to propose and evaluate new functionality and extensibility enhancements that addressed the basic limitations of the protocol. Examples of this work include the addition of the OtherInformation structure, the Scan service, and the Extended Services in V3 of Z39.50.

The second approach was to work with other organizations to develop open, interoperable formats and conventions needed to support the exchange of scitech information.

These formats and conventions were designed to be plugged into the Z39.50 protocol in a standard and interoperable manner. Examples include STAS, the Scientific and Technical Attribute and Element Set [6], and CXF, the Chemical eXchange Format [7].

The third approach was to work with other implementors and partners to develop and propose externally-defined data structures to be used within the protocol to carry additional information needed for commercial scitech information systems. Some of these structures were proposed for public adoption, while others were intended for more limited contexts of interoperability. Examples include the GRS-1 (the Generic Record Syntax), AdditionalSearchInformation, and the SetUserParm Extended Service.

CAS began its implementation of Z39.50 in 1991, and continues to evolve and expand that implementation over time. CAS initially based its implementation on V1, then V2, and finally V3 of the standard. Each of these versions presented its own set of issues, which had to be addressed within its own context. In several cases, the next version of the standard included opportunities for better solutions to a given problem than possible in the previous version. Since most current implementations of Z39.50 are based on V2, interoperability goals dictate that, wherever possible, solutions to issues be applicable to both V2 and V3. Therefore, some approaches to the issues represent compromises between the functionality available in V3 and the need to interoperate with V2 implementations.

Solutions to the issues

1. Protocol scope and extensibility

The first extensibility challenge was that Z39.50 V2 provided no way to carry supplementary data in the Protocol Data Units (PDUs) defined for a given service within the protocol. V3 addresses this problem by adding an optional OtherInformation structure to each Z39.50 PDU. This allows a given Z39.50 service to carry along externally-defined information that augments the core information fields explicitly defined within the PDU. The OtherInformation structure allows the externally-defined information structure to be uniquely identified by an Object IDentifier (OID) to improve interoperability and avoid any ambiguity.

A simple example of the use of this feature within the CAS Z39.50 implementation is to allow a language code to be carried within the OtherInformation field of the Scan Request PDU. This allows the client to specify the preferred national language for the output of a Scan operation against a multi-lingual thesaurus.

The second extensibility challenge was that Z39.50-1992 lacked several necessary services, and provided no interoperable way to add supplementary services to the protocol. Z39.50-1995 addresses this in two ways: (1) addition of new services such as Scan and Sort which are closely related to search and retrieval; and (2) addition of the Extended Services facility.

CAS has leveraged both of these new protocol features to provide better access to its existing databases and services. As an example of the first case, CAS has implemented the Scan facility to allow term expansion within both database indices and thesauri.

In the second case, CAS has leveraged the Extended Services facility extensively to support both “standard” Extended Services (those specified in Z39.50-1995), and “local” Extended Services. An example of a “local” CAS Extended Service is the Analyze Extended Service. Analyze allows the client to perform an analysis of the content of one or more records within a Result Set, based on specified data elements and their values. This information can then be used to help the user select the information of greatest value.

2. Attribute Set limitations

The issue here was how to identify and characterize scitech query terms within a Z39.50 Search Request. The solution to this issue was to define a new Scientific and Technical Attribute and Element Set (STAS).

STAS defines both an Attribute Set and an Element Set. The STAS Attribute Set supports the use of scientific, technical, and related search terms within a standard Type 1 or Type 101 Query carried within a Z39.50 Search Request. The STAS Element Set supports identification and selection of data elements retrievable from scientific, technical, and related databases using a Z39.50 Present Request. The STAS Attributes and Elements are also useful within other Z39.50 services such as Scan and Sort.

CAS originally developed STAS as part of its research project on the use of Z39.50. In September 1994, co-sponsors CNIDR [8], Dialog [9], FIZ Karlsruhe [10], and CAS announced the public availability of STAS as an open, public definition. As such, any interested party may freely use and contribute to STAS. STAS maintenance and registration functions are provided by CNIDR. The Z39.50 Maintenance Agency has assigned the STAS Attribute Set a standard public Object Id, which is listed in Appendix ATR of the Z39.50-1995 standard along with bib-1 and others.

The Search service supported by Z39.50 V2 has a limitation that influenced the approach taken in defining STAS. A V2 Search Request allows Attributes from only a single Attribute Set to be used in a given RPN query. And yet it is a practical requirement to support searches containing both bibliographic and scitech search terms within the same query. This limitation required definition of a single Attribute Set that contained both bibliographic and scitech Attributes.

Therefore, the STAS Attribute Set is defined as a superset of the bib-1 Attribute Set, and implicitly imports all Attributes specified by the bib-1 Attribute Set. Additional

STAS Attribute types and values are assigned identifiers that are outside of the range assigned to bib-1 Attribute types and values. As new Attribute types and values are added to the bib-1 Attribute Set, they automatically become part of the STAS Attribute Set.

The STAS Attribute and Element Set definitions are evolving from the ongoing effort of defining Attribute and Element mappings for existing scitech databases. Wherever a valid mapping can be defined between existing bib-1 Attributes and a database's search fields, bib-1 Attributes will be used. For each database search field that has no equivalent bib-1 Attribute, a new STAS Attribute will be defined.

3. Search service limitations

a. The V2 Search Request allowed specification of only one Attribute Set within an RPN query.

This issue was addressed in two ways: (1) the definition of STAS as a superset of the bib-1 Attribute Set addressed the problem within the V2 context; and (2) expanding the V3 RPN query to allow specification of multiple Attribute Sets provided a long-term robust solution.

Z39.50 V2 supports a single Attribute Set ID field in the RPN query within a Z39.50 Search Request. Search Requests using STAS will specify the STAS Attribute Set Object IDentifier (OID) in this field. This allows use of both bib-1 and other STAS Attributes within the query.

Z39.50 V3 allows optional specification of the Attribute Set Id for each search term, and even for each Attribute. This feature of V3 allows STAS to be used in combination with the bib-1 and/or any other Attribute Set(s). Bib-1 Attributes may be explicitly identified as such, and other STAS Attributes may be identified by the STAS Attribute Set Id.

b. There was no mechanism for specifying the datatype of a given term within a Type 1 query.

The short term (V2) approach to this issue was to define STAS Attributes explicitly enough to provide strong hints about the datatype of the term. In retrospect, the disadvantage of this approach was the proliferation of similar Use Attributes for a term with the same semantics but different datatypes, forms, or formats.

The long term solution was to expand the RPN search term definition within V3 to support explicit data typing of the term contents. With this capability, the current STAS philosophy is to move away from datatype-specific Use Attributes.

c. V2 provided no way to specify the "units" in which a search term was expressed.

The short-term (V2) approach to this issue was to carry the units indicators along with the term value within the RPN query term. Although this works adequately within a limited context, the lack of publicly defined conventions for expressing units in this manner limits the interoperability of this approach.

The long-term solution to this problem is the explicit support for units within V3. CAS defined and proposed an IntUnit structure for specifying values with units. IntUnit allows specification of an Integer value, qualified by a scale factor, Units System, Unit Type, and Unit. After some discussion and modifications, this IntUnit structure has been incorporated into several parts of the V3 standard. In particular, the IntUnit is one of the supported datatypes for RPN search terms, thus allowing a search term to be expressed in explicit units.

d. There was no way to send or return additional search information.

Since V2 Search PDUs lacked the capability for carrying additional search information, indicators about the "Type" or "Scope" of a search could not be expressed in the Search Request.

The V2 approach to this issue was to define new STAS Attribute Types to express Search Type and Search Scope. For example, currently defined STAS Search Types include Substructure, Closed Substructure, Family, and Exact Searches. And currently defined Search Scope values include Full File, Sample File, Range, and Subset Searches.

Although these indicators are generally expressed globally for an entire query, this approach allowed the flexibility of specifying Search Type and Scope at the subtree or even the term level of a given query.

The V3 approach to this issue was the addition of a new optional AdditionalSearchInfo field in the Search Request PDU. Although this V3 feature has not been leveraged yet, it will eventually provide a more robust solution.

A second issue was that additional information about the search results, such as how many "hits" were found for each term in the query, could not be returned in the Search Response.

The V3 solution to this problem is a new optional AdditionalSearchInfo field in the Search Response PDU. CAS leveraged this V3 feature by defining an external structure for carrying various types of information about the search results, including how the server interpreted the query and how many "hits" were found for each term in the query. This structure was proposed to the ZIG, and following discussion and modification, was added to the V3 standard.

4. Element Specification

The CAS solution to the Element Specification issue within the V2 context was to define a simple syntax for expressing any combination of elements to be retrieved. This syntax is called STETSEN (the Scientific and Technical Element Set Names) [11]. STETSEN draws on STAS, by using the STAS Element Numbers to identify individual elements to be retrieved. Just as a unique STAS Use Attribute Value can be defined for each database search field, a unique STAS Element Number can be assigned to each database retrieval field (retrievable element).

A given combination of retrieval elements is expressed using the STETSEN syntax by a character string containing a list of the corresponding STAS Element Numbers, separated by commas or spaces. These STAS Element Numbers may be combined with other ElementSetNames such as Full (F), Brief (B), or target-defined names. Since a STETSEN ElementSetName is simply a character string, it can be legally carried in the ElementSetName field within either the V2 or V3 Present Request PDU.

The general Element Specification issue was addressed in a more robust manner in V3. CAS worked with other implementors within the ZIG to define several new mechanisms within the V3 Present service to support more flexible and powerful mechanisms for element retrieval. These include the CompSpec, eSpec-1, Variant, and Schema features. Although CAS has not yet fully leveraged these new features, they provide very powerful retrieval capabilities, and will be implemented in the future.

5. Record Syntax limitations

Since externally-defined Record Syntaxes can be flexibly "plugged into" both V2 and V3 of Z39.50, the basic issue was addressed by defining new Record Syntaxes that met the requirements for scitech information. Related issues were addressed within the context of the V3 Present service via mechanisms such as the Schema concept and the CompSpec structure.

One fundamental issue that was clarified during the development of these V3 mechanisms was the fact that a Record Syntax consists of both an Abstract Syntax and a Transfer Syntax. The Abstract Syntax (often expressed in a formal notation, such as ASN.1) specifies the content, semantics, and structure of the record [12]. The Transfer Syntax (usually defined by a set of encoding rules such as BER) ensures that the information in the record is successfully conveyed over a network in a portable and unambiguous manner [13]. When USMARC and the other MARC formats were the only supported Z39.50 Record Syntaxes, these distinctions were less critical, and not well articulated. But development of new Record Syntaxes to support non-textual information forced developers to better articulate this concept within the Z39.50 standard.

CAS initially worked with a small group of Z39.50 implementors, led by John Kunze (University of California Berkeley) to develop a new "info-1" Record Syntax. The goal of info-1 was to flexibly carry tagged elements of multiple datatypes and formats as well as metadata about those elements. CAS implemented at least three generations of this concept, starting with info-1, then GRS-0, and finally GRS-1. After several years of discussions, implementations, and refinement, this work has evolved into the Generic Record Syntax-1 (GRS-1), as specified in V3. GRS-1 is a very powerful Record Syntax that supports flexible delivery of literally any type of information of essentially arbitrary complexity. GRS-1 supports tagged elements, metadata, hierarchical data structures, unit specification, and information about the particular form (variant) of individual elements. Use of the BER standard to encode GRS-1 records ensures strong data portability across networks and computing platforms.

CAS has upgraded its Z39.50 implementations to use GRS-1 extensively in delivering scitech information via the Present service. When STAS is used in combination with GRS-1, the Tag Numbers used to identify elements carried in a GRS-1 record are the STAS Element Numbers. The STAS Element Numbers therefore constitute a standard Z39.50 TagSet that has been registered in the V3 standard.

Use of STAS Element Numbers within GRS-1 leverages a STAS convention, wherein the same number space is used to assign STAS Use Attribute Values, STAS Element Numbers, and STAS Tag Numbers. Within many databases, there are often retrieval fields (elements) that correspond to search fields (Attributes). It is often useful for a client to be able to relate a retrieval field with a corresponding search field. A database field that can be both searched and retrieved is assigned the same value for its STAS Use Attribute, its STAS Element Number, and its STAS GRS-1 Tag number.

In addition to GRS-1, there was also a need for a simple Record Syntax for delivering pre-formatted textual information for display. CAS worked with a small group of other implementors to propose and refine SUTRS (the Simple Unstructured Text Record Syntax), which is now a registered Record Syntax defined within V3. SUTRS is especially useful as a "lowest common denominator" Record Syntax between clients and servers that have minimal knowledge of each others' data or conventions.

Finally, CAS needed a standard interoperable format for exchanging detailed chemical information, and worked with other organizations to develop the Chemical eXchange Format (CXF). CXF is defined in ASN.1, encoded using BER, and may be used either as a Record Syntax or as an Element Syntax for a tagged element within GRS-1. In the interest of maximum scitech interoperability, CAS has submitted CXF to the industry as an open definition, available for use by any interested organization. CAS uses CXF extensively in the search and retrieval of chemical information via Z39.50.

6. Other Retrieval limitations

There were other limitations to the Present service for retrieving scitech information using Z39.50 V2.

a. There was no way to request retrieval metadata.

In conjunction with the development of the GRS-1 Record Syntax, CAS also worked with other implementors to develop a complementary element specification mechanism for use within V3. This work resulted in the V3 eSpec-

1 structure, which may be used within the CompSpec structure of the V3 Present Request.

eSpec-1 allows the client to request retrieval of element metadata, with or without the corresponding data. GRS-1 provides the complementary ability to deliver the metadata, with or without the corresponding data. And the V3 TagSet-M defines a set of Tags which can be used to identify metadata carried within GRS-1 in an interoperable manner. The combination of these new mechanisms allows a client to dynamically discover characteristics such as the size, cost, and copyright restrictions of information, prior to retrieving it.

b. There was no way to request different elements and Record Syntaxes for different databases, when the Result Set was created by a search against multiple databases.

This need has been addressed by development of the new CompSpec structure of the V3 Present Request. This new structure allows the client to specify a particular combination of Record Syntax and element specification for each database from which the Result Set was created.

c. There was no way to request a particular form of a given element.

The V3 eSpec-1 structure allows the client to request a particular form of a given element for retrieval via a concept called "Variants." V3 defines Variant-1, a standard Variant Set which identifies a number of classes and types of variants such as national language, body type, size, etc. GRS-1 provides the complementary ability to deliver the requested variant of the element as well as the corresponding "applied variant" identifiers. And the V3 metadata capabilities already mentioned allow dynamic discovery of the available variants of a given element before retrieving it. The combination of these new V3 mechanisms allows a client to dynamically negotiate and retrieve the best form of information for its needs.

d. There was no way to request particular sub-trees of a hierarchical data structure.

The V3 eSpec-1 structure allows the client to request retrieval of specific elements or subtrees within a hierarchical data structure, using the concept of TagPaths. A TagPath specifies the path through a data structure, where each node in the path is identified by a tag. ESpec-1 allows the client to specify a given element or subtree for retrieval by specifying its TagPath. GRS-1 provides the complementary ability to deliver the requested element or subtree as well as the corresponding Tags representing the Path. And the V3 Schema concept provides the client and server with a common understanding of the hierarchical database structure by documenting it using TagPaths.

e. There was no flexible way to return large Result Set records.

The new Segmentation features of the V3 Present Facility address this problem. Records larger than the message buffer provided by the client can now be retrieved, by breaking them up into pieces, which are delivered in Segment Request PDUs. CAS is one of the first implementors of Segmentation, including segmentation of GRS-1 records, using a recently defined Fragmentation Syntax.

A second V3 capability was added to support the retrieval of "pieces" of individual elements, using the eSpec-1, GRS-1, and Variant mechanisms. A client may retrieve a single large element, such as a large image, in pieces by specifying its retrieval using eSpec-1. The particular piece and its size can be specified using Variants. And GRS-1 identifies the specific element, its piece and its size upon delivery.

Lessons learned

In the course of implementing Z39.50, several lessons were learned.

1. Standards can be enhanced and expanded, but it is not easy.

A given standard rarely meets all the needs of a given implementor. In some cases, the best way to address the shortcomings of a key standard is to actively participate in its development. However, influencing the scope and functionality of an evolving standard such as Z39.50 is neither easy nor inexpensive. It takes time, effort, resources, patience, persistence, and commitment. For some implementors with advanced requirements, it may actually be simpler to design and implement a proprietary protocol which does exactly what is needed. However, we have previously learned that a world of proprietary protocols does not promote the flow of information. The interoperability provided by Z39.50 promotes the unencumbered flow of information, and opens up many technical and business opportunities which would not be possible with the use of proprietary protocols. In the case of Z39.50, our investment in active standards participation was successful and will be leveraged.

2. Implementation experience leads to better standards.

Having participated in several other standards development efforts, it is the author's opinion that the process for developing the Z39.50 standard was an unusually successful one. A lot of the credit for this goes to the Z39.50 Maintenance Agency and the ZIG. The administrative and political hurdles were kept at a manageable level, allowing technical needs to be addressed in a timely manner. An active role by Z39.50 implementors in defining and expanding the standard within the ZIG added a practical influence to the process. In several cases, implementors such as CAS designed and implemented new features and services before proposing them to the ZIG for inclusion in the Z39.50 standard. In other cases, early implementation of features proposed within a working draft of the standard helped refine and improve their definition, leading to a better specification. This active participation by implementors coupled with early implementation experience resulted in a better Z39.50 standard.

3. The Attribute, Element, and metadata problems are difficult, and cannot be solved by a protocol alone.

Some of the major remaining interoperability challenges for implementors of Z39.50 revolve around the need for unique and unambiguous identification of information, both in search queries and retrieved answers. The ambiguities and inconsistencies in the use of the bib-1 Attribute Set and the MARC record syntax are the most visible aspects of the problem. However, an underlying source of the problem lies in the original indexing policies used to identify information. Different organizations index and identify information in different ways. Mappings of the bib-1 Attributes into search elements and mappings of retrieval elements into MARC records are not consistent across databases or organizations. This results in reduced interoperability of information. Definition of a protocol such as Z39.50 cannot alone solve this problem. Definition of metadata standards, improved indexing standards, and unambiguous Attribute and Element Sets are also needed.

CAS has attempted to avoid many of the MARC and bib-1 Attribute Set interoperability problems by defining and using the STAS Attribute, Element, and Tag Sets in conjunction with the SUTRS and GRS-1 Record Syntaxes. However, for protocol consistency, STAS also inherits some of the characteristics of the bib-1 Attribute Set. It is therefore expected that STAS will continue to evolve, as additional experience is gained with its use. Other Attribute, Element, and Tag Sets will probably be defined to address other information disciplines. A new bibliographic Attribute Set may eventually evolve to replace bib-1. In summary, this particular area of information interoperability is a challenging one, and will require additional work in the future.

Futures

The CAS implementation of Z39.50 is an ongoing project. The specific issues and solutions identified in this paper reflect the functionality implemented to date. However, Z39.50-1995 defines a very rich set of services and features, several of which CAS has not yet implemented. In the future, CAS will continue to implement additional Z39.50 features and services, as required by its users and projects. Some representative examples are noted here.

The STETSEN syntax for ElementSetNames has provided an adequate means for specifying element selection within both the V2 and V3 contexts. However, the future direction will be to implement support for the eSpec-1 definition to enhance interoperability and support more granular retrieval specifications. For example, eSpec-1 will allow the client to:

- retrieve and leverage metadata
- discover variants of retrieval elements
- retrieve the optimal form of information
- request the server to package information in an optimal manner

The Explain service supports the discovery of information useful to both the users and clients of Z39.50 services. Explain holds great potential for expanding the intelligence and reach of Z39.50 clients, without building specific database knowledge into the client software. Use of Explain will be especially useful when the client and server have been independently developed or are operated by different organizations. However, the Explain specification has just recently been finalized, and is probably one of the least mature portions of Z39.50-1995. As Explain matures and gains wider acceptance, CAS will add support for Explain in the future.

There are several standard Extended Services defined in Z39.50-1995, which CAS will implement in the future to provide access to advanced features of the CAS search and retrieval systems. For example, the ItemOrder Extended Service defines a mechanism for initiating document orders from a Z39.50 client. CAS will initially use Item Order to convey user requests to the CAS Document Delivery System.

Due to project requirements and timeframes, CAS implemented some V3 Extended Services and data structures prior to their finalization in Z39.50-1995. Since Z39.50 explicitly supports the identification and use of such "local" data structures and Extended Services, they are currently being used between CAS clients and servers, with no compromise in protocol compliance. However, over time, these CAS-defined conventions and Extended Services will be migrated to adhere to the "standard" structures and Extended Services defined by Z39.50-1995. This will improve interoperability with Z39.50 clients and servers implemented and operated by other organizations.

Conclusion

Z39.50 is a very positive example of the value of interoperable standards. It supports a rich set of interoperable services between separately developed clients and servers. Z39.50 has rapidly evolved to address the needs of implementors operating within the context of various information disciplines and commercial search systems. The model of the ZIG working in concert with NISO, ANSI and ISO has proven to be very successful in meeting the needs of the information retrieval community. In recognition of the demonstrated value of Z39.50, it is being used as one of the key access protocols for current and future CAS software projects and products.

References

[1] ANSI/NISO Z39.50-1988, “ANSI Z39.50: Information Retrieval Service and Protocol”, 1988.

[2] ISO 10163—Information and Documentation - Search and Retrieve Application Protocol Specification for Open Systems Interconnection, 1991.

[3] ANSI/NISO Z39.50-1992, “ANSI Z39.50: Information Retrieval Service and Protocol”, 1992.
<URL:ftp://ftp.cni.org/pub/NISO/docs/Z39.50-1992/www/Z39.50.toc.html>

[4] ANSI/NISO Z39.50-1995, “ANSI Z39.50: Information Retrieval Service and Protocol”, 1995.
<URL:http://lcweb.loc.gov/z3950/agency/1995doc.html>

[5] Chemical Abstracts Service (CAS), a division of the American Chemical Society, is the world’s leading provider of chemical information.
<URL:http://info.cas.org/welcome.html>

[6] STAS, the Scientific and Technical Attribute and Element Set.
<URL:http://stas.cnidr.org/STAS.html>

[7] CXF, the Chemical eXchange Format, Version 1.0, September 1994, CAS.
<URL:ftp://info.cas.org/pub/cxf/specification/cxf.1.0/>

[8] CNIDR is The Clearinghouse for Networked Information Discovery and Retrieval, located at MCNC in Research Triangle Park, N.C.
<URL:http://cnidr.org/welcome.html>

[9] Knight-Ridder Information Inc., formerly Dialog Information Services, Inc., a Knight-Ridder company.
<URL:http://www.dialog.com/>

[10] FIZ-Karlsruhe is the FachInformationsZentrum Karlsruhe, operator of STN International in Europe.
<URL:http://www.fiz-karlsruhe.de/>

[11] STETSEN—The Scientific and Technical Element Set Names, an internal CAS technical paper.

[12] ISO 8824—Information Processing Systems - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1), 1990.

[13] ISO 8825—Information Processing Systems - Open Systems Interconnection - Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1), 1990.