Z39.50 Version 3:

Tenth Draft
(V3D10)

April 1994

Status*: This is the tenth draft of Version 3 of Z39.50, prepared by the Z39.50 Maintenance Agency, for review by the Z39.50 Implementors Group at the April 1994 ZIG meeting. This is the last draft of version 3 prepared for ZIG review. A ballot-draft, for ballot by NISO, is planned for June.*

Note to implementors: the Explain facility, as it is described in this draft, may be revised substantially for the ballot draft.

Contents

(Contents continued on reverse)

(Contents continued)

Appendices

1. Introduction

This is one of a set of standards produced to facilitate the interconnection of computer systems. It is positioned with respect to other related standards by the Open Systems Interconnection (OSI) basic reference model (ISO 7498). This standard defines a protocol within the application layer of the reference model, and is concerned in particular with the search and retrieval of information in databases.

1.1 Scope and Field of Application

This standard describes the Information Retrieval application service (section 3) and specifies the Information Retrieval application protocol (section 4). The service definition describes services which provide capabilities within an application. The description neither specifies nor constrains the implementation within a computer system. The purpose of the service definition is to define the capabilities that the protocol must support. The protocol specification includes the definition of the protocol control information, the rules for exchanging this information, and the conformance requirements to be met by implementation of this protocol.

This standard is intended particularly for use by systems supporting information retrieval services for organization such as libraries, information utilities, and union catalogue centers. It addresses connection oriented, program-to-program communication. It does not address the interchange of information with terminals or via other physical media.

1.2 Version

This standard, Z39.50-1994, specifies versions 2 and 3 for the Z39.50 service and protocol.

[Ed. note: "Z39.50-1994" is temporarily used to refer to this version. "1994" will be replaced (if different) by the year of its approval by ANSI.]

Notes:

(1) ANSI Z39.50-1992 specifies version 2 only.

(2) For compatibility with version 1 of the Search and Retrieve Protocol (ISO 10163), Version 2 of Z39.50 is assumed identical to version 1 of Z39.50; thus implementations which support version 2 automatically support version 1. (Version 1 of ANSI Z39.50-1992 should not be confused with ANSI Z39.50-1988).

Certain definitions and procedures specified within the standard apply specifically to version 3 and are noted as such.

1.3 References

ANSI Z39.58 -- *Common Command Language for Online Interactive Information Retrieval* 1991

ISO 2709 -- *Documentation - Format for Bibliographic Information Interchange on Magnetic Tape* 1981.

ISO 4217 -- *Codes for the representation of currencies and funds* 1990.

ISO 7498 -- *Information Processing Systems - Open Systems Interconnection - Basic Reference Model* 1984.

ISO TR 8509 -- *Information Processing Systems - Open Systems Interconnection - Service Conventions* 1987.

ISO 8649 -- *Information Processing Systems - Open Systems Interconnection - Service Definition for the Association Control Service Element* 1987.

ISO 8650 -- *Information processing systems - Open Systems Interconnection - Protocol Specification for the Association Control Service Element* 1987.

ISO 8777 -- *Documentation - Commands for Interactive Text Searching*.

ISO 8822 -- *Information Processing Systems - Open Systems Interconnection - Connection Oriented Presentation Service Definition* 1988.

ISO 8824 -- *Information Processing Systems - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1)* 1990.
ISO 8825 -- *Information Processing Systems - Open Systems Interconnection - Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)* 1990.
ISO 9545 -- *Information processing systems - Open Systems Interconnection - Application Layer Structure* 1989.
ISO 10163 -- *Documentation - Search and Retrieve Protocol Specification* 1991.
Library of Congress -- *USMARC Code List for Languages* 1989.
ISO -- *International Register of Coded Character Sets to be used with Escape Sequences* 1992.

2. Definitions

A-association	See "Application association".
Abstract database record	An abstract representation of the information in a record, formed by the application of a schema to a database record, or an element specification to an abstract database record.
Abstract record structure	The primary component of a database schema. An abstract record structure applied to a database record results in an abstract database record.
Abstract syntax	A description of a particular data type using an abstract syntax notation. It can be referenced by an OID.
Abstract syntax notation	A language that allows the denotation of data types in a representation-independent manner. ASN.1 is an example.
Access point	A unique or non-unique key which can be specified either singly or in combination with other access points in a search for records. An access point may be equivalent to an element (defined by an abstract syntax), derived from a set of one or more elements, or unrelated to any element.
Aggregate present response	Segment requests (if any) together with the Present response, for a Present operation.
APDU	See "Application Protocol Data Unit".
Applied variant	A variant Specification applied to an element by the target, when that element is included in a retrieval record.
Application Association	A communication session between a database user and a database provider. It may consist of one or more (serial) Z-associations. *ISO 9545: A cooperative relationship between two application-entity-invocations for the purpose of communication of information and coordination of their joint operation. This relationship is formed by the exchange of application-protocol-control-information using the Presentation-Service.*
Application Protocol Data Unit	A unit of data passed between an origin and a target. *ISO 7498: A unit of data specified in an application-protocol and consisting of application-protocol-information and possibly application-user-data.*

Application Protocol Control Information	The information conveyed by an application protocol data unit.
Application Protocol	The rules governing the format and exchange of information between an origin and target.
| **ASN.1**	Abstract Syntax Notation One, as specified in ISO 8824 and ISO 8825.
Attribute	A characteristic of a search term, or one of several characteristic components which together form a characteristic of a search term.
Attribute element	An attribute, represented by a pair of integers. The two integers represent an attribute type and a value of that type.
Attribute list	A set of attribute elements, and for each, the attribute set id to which it belongs. An attribute list is combined with a search term to form an operand of a Type-1 query. Usually, one of the attribute elements from the set corresponds to a normalized access point, against which the term (as qualified by the other attribute elements) is matched.
Attribute set	A set of attribute types, and for each, a list of attribute values. Each type is represented by an integer, unique within that set (as identified by its attribute set id), and each value for a given type is represented by an integer, unique within that type.
Attribute set id	An OID which identifies an attribute set, to which an attribute element (within an attribute list) belongs.
| **Attribute type**	A component of an attribute element. An attribute set defines one or more attribute types and assigns an integer to each type (it also defines values specific to each type). For example, bib-1 assigns the integer 1 for the attribute type "Use".
| **Attribute value**	A component of an attribute element. An attribute set defines one or more values for each attribute type that it defines, and assigns to each value an integer (unique within that type). For example, bib-1 assigns the integer 1 to the attribute (within type Use) "personal name".
Client	The application that includes the origin. The database user.
| **Client system**	The system on which the client resides.
Composition specification	A specification which may include a schema name, element specification, and record syntax name.
Confirmed service	A request from the origin or target followed by a response from the peer.
Data element	See "Element".
| **Database**	a collection of information units; each unit is a collection of related information referred to as a database record.
| **Database record**	A local data structure representing an information unit in a database.

Database schema A common understanding shared by the origin and target of the information contained in the records of the database, to allow the subsequent selection of portions of that information via an element specification. A schema applied to a database record results in an abstract database record.

Element A unit of information defined by a schema.

Element request A component of an element specification; a retrieval request for a specific element.

Element set name An element specification which is a primitive name.

Element specification An instance of an element specification format, or an element set name. An element specification transforms an abstract database record into another instance of the abstract database record (this may be a null transformation). The element specification selects elements from the abstract database record, and possibly also specifies different abstract representations for those elements.

Element specification format A structure used to express an element specification.

Element specification name The name of an element specification format or an element set name.

Facility A logical group of Z39.50 services; in some cases, a single service. Alternately, a facility might not consist of services, but instead use services of other facilities.

Final fragment A fragment which ends at the end of a record.

Fragment A proper substring of a record (in the context of level-2 segmentation, a record is treated as a string of bytes).

Initiating request A request that initiates an operation.

Intermediate fragment A fragment which neither starts at the beginning nor ends at the end of a record.

IR Information Retrieval.

Name A linguistic construct, expressed in some language, which corresponds to an object. A name denotes (i.e. identifies) the object to which it is bound.

Non-confirmed service A request from the origin or target (see also "confirmed service").

Object Identifier An unambigous, globally recognized, registered identifier for a data object, assgned by a registration authority.

OID See "Object identifier".

Operation An initiating request and the corresponding terminating response, along with intervening related messages. There may be multiple concurrent operations within a Z-association.

Operation type	The name of an initiating request. For example a Search request initiates an operation whose type is "search".
| **Origin**	The entity that initiates a Z-association, and initiates operations during the Z-association.
OSI	Open Systems Interconnection.
P-context	See "Presentation context".
Presentation context	The pairing of an abstract syntax with a transfer syntax, negotiated by the presentation layer, in order for that abstract syntax to be used during the application association.
Primitive	*ISO TR 8509: An abstract, implementation-independent representation of an interaction between the service-user and the service-provider.*
Primitive Name	A name, whose internal structure is not required to be understood or have significance to users of the name.
Record syntax	When specified by the origin, an OID referring to an abstract syntax paired (or to be paired by the target) with a transfer syntax, that the origin requests the target to use for the retrieval records. When specified by the target, an OID or p-context accompanying retrieval records, referring to an abstract syntax paired with a transfer syntax.
Result set	A local data structure used as a selection mechanism for the transfer of records, identified by a query. Its logical structure is a named, ordered list of entries, each consisting of a database name, a pointer to a record within the database, and possibly, unspecified information associated with each record, which may be used as a surrogate for the search that created the result set.
| **Response record**	A retrieval record or a surrogate diagnostic record, representing a database record, in a Search response or (aggregate) Present response.
Retrieval record	The exportable structure defined by the application of a record syntax to an abstract database record.
RPN	Reverse Polish Notation.
| **RPN query**	A search query represented in RPN format.
Schema	See "Database schema".
Segment	A message which is sent (or is in preparation for transmission) by the target as part of an aggregate Present response, i.e. a Segment request or Present response.
Server	The application that includes the target. The database provider.
| **Server system**	The system on which the server resides.
Service-provider	*ISO TR8509: An abstraction of the totality of those entities which provide a service to peer service-users.* Note: the concept of service-provider is employed to facilitate the specification of protocol procedures. It is not analogous to the

database provider, and it does not refer to providers of telecommunication services. It is used only within 4.2.2, to describe the protocol model.

Service-user	That portion of an application which makes requests upon the open systems environment. Note: The concept of service-user is employed to facilitate the specification of protocol procedures and is not analogous to the database user. It is used only within 4.2.2, to describe the protocol model.
Simple Present response	An aggregate Present response consisting of a single segment, i.e. consisting of a Present response only, and no Segment requests.
Starting fragment	A fragment which starts at the beginning of a record.
| **Surrogate diagnostic Record**	A diagnostic record supplied in place of a retrieval record, representing a database record.
Target	The entity that accepts a Z-association.
| **Supported variant**	A variant specification supported for a particular element. The target might provide a listing for a given element of the supported variants.
Terminating response	A response that terminates an operation.
Transfer syntax	A syntax, paired with an abstract syntax, to form a record syntax.
| **Type-1 query**	See RPN Query.
| **Variant**	One of possibly several forms in which an element is available for retrieval. The origin may request, or the target present, an element according to a specific variant. The target may indicate what variants are available for an element.
| **Variant request**	A variant specification occurring within an element request.
| **Variant Set**	A set of classes, and for each class a set of types, and for each type, a set of values. The triple (class, type, value) is a variantSpecifier.
| **Variant Specification**	A sequence of triples, each of which is a variant specifier. A variant specification may take the form of a variantRequest, appliedVariant, or supportedVariant.
| **Variant Specifier**	A component of a variant specification. It consists of the triple: (class, type, value).
Z-association	See "Z39.50-association".
Z39.50-association	An association explicitly established by the origin and either explicitly terminated by the origin or target, or implicitly terminated by termination of the A-association. Communication between origin and target is via a Z39.50-association within an application association. There may be multiple, serial Z-associations within an A-association.

3. Information Retrieval Service

This section defines the Information Retrieval service, which is supported by the Information Retrieval protocol.

3.1 Model and Characteristics of the Information Retrieval Service

The service definition describes an activity between two applications: an initiating application, the client, and a responding application, the server. The server is associated with one or more databases. Communication between the client and server is carried out by the Z39.50 protocol, which is specified in section 4. The protocol specification is logically divided into procedures pertaining to the client and procedures pertaining to the server. The portions of the client and server which carry out the Z39.50 protocol procedures are referred to respectively as the Z39.50 origin and the Z39.50 target.

Communication between origin and target is via a Z39.50-Association (Z-association) within an application association (A-association; see 4.2.1.2). A Z-association is explicitly established by the origin and may be explicitly terminated by either origin or target, or implicitly terminated by termination of the A-association.

There may be multiple serial Z-associations within an A-association. There may be multiple serial as well as concurrent operations (see 3.1.2) within a Z-association.

The roles of origin and target may not be reversed within a Z-association. A Z-association cannot be restarted, thus no status information is retained once a Z-association is terminated, except information which is explicitly saved.

The service definition describes services and operations; services are grouped by facilities. Services and operations are described in sections 3.1.1. and 3.1.2; facilities are described in section 3.2.

3.1.1 Services

The service definition describes Z39.50 services, each carried out by the exchange of messages between the origin and target. A message is a <u>request</u> or a <u>response</u>. Services are <u>confirmed</u> or <u>non-confirmed</u>. A confirmed service consists of a request from the origin or target followed by a response from the peer. A non-confirmed service consists of a request from the origin or target. For example, the Search service is a confirmed service initiated by the origin; it consists of a Search request from the origin followed by a Search response from the target.

3.1.2 Operations

There are eight <u>operation types</u> described within this standard: Init, Search, Present, Delete, Scan, Sort, Resource-report, and Extended-services. A request from the origin of a particular operation type initiates an operation of that type (for example a Search request initiates a Search operation) which is terminated by the respective response from the target. Only the origin may initiate an operation, and not all origin requests initiate an operation (see 3.4).

A request that initiates an operation is called an <u>initiating request</u> and a response that terminates an operation is called a <u>terminating response</u>. From the origin perspective, an operation begins when it issues the initiating request, and ends when it receives the terminating response. From the target perspective, the operation begins when it receives the initiating request and ends when it sends the terminating response. An operation consists of the initiating request and the terminating response, along with any intervening related messages (see 3.4).

3.1.3 Model of a Database

The objective of this standard is to facilitate the open interconnection of database users with database providers. The ways in which databases are implemented differ considerably; different systems have different styles for describing the storage of data and the means by which it can be accessed. A common, abstract model is therefore used in describing databases, to which an individual system can map its implementation. This enables different systems to communicate in standard, mutually understandable terms, for the purpose of searching and retrieving information from a database. Search and retrieval are described in 3.1.4 and 3.1.5.

The term database as used in this standard refers to a collection of records. Each record is a collection of related information, treated as a unit. The term database record refers to a local data structure representing information in a particular record. Associated with a database are one or more sets of access points which can be specified in a search for database records (see 3.1.4), and one or more sets of elements which may be retrieved from a database record (see 3.1.5). An access point may, but need not, be related to an element: it can be equivalent to an element, derived from a set of one or more elements, or unrelated to any element.

3.1.4 Searching a Database

A search query may be applied to a database, specifying values to be matched against the access points of the database. An access point is a unique or non-unique key which can be specified either singly or in combination with other access points in a search for records.

The subset of records formed by applying a search query is called the result set (see 3.1.6). A result set may itself be referenced in a subsequent query and manipulated to form a new result set.

A generic search refers to one or more databases and includes a query. The Type-1 query defined in this standard consists of either a single access point clause, or several access point clauses linked by logical operators. For example:

> "In the database named 'Books' find all records for which the access point 'title word' contains the value 'evangeline' AND the access point 'author' contains the value 'longfellow'".

Each access point clause consists of an attribute list and a search term. The attribute list contains attribute elements; usually, one of the attribute elements corresponds to a normalized access point, against which the term (as qualified by the other attribute elements within the list) is matched. Each attribute element is a pair of integers representing an attribute type and a value of that type (for example, type might be "access point" and value "author"; or type might be "truncation" and value "left").

Each attribute element is qualified by an attribute set id which identifies the attribute set to which the attribute element belongs. An attribute set specifies a set of attribute types, and for each, a list of attribute values, and integers to represent the types and values.

3.1.5 Retrieving Records from a Database

Following the processing of a search, the result set is made available by the target, to the origin, for subsequent retrieval requests. When requesting the retrieval of a record from a result set, the origin may specify a database schema, an element specification, and a record syntax.

For the purpose of retrieving records from a result set, associated with each database are one or more schemas, representing a common understanding shared by the origin and target of the information contained in the records of the database, to allow the subsequent selection of portions of that information via an element specification.

A schema defines an abstract record structure which when applied to a database record results in an abstract database record, which is an abstract representation of the information in the record. An element specification applied to an abstract database record results in another instance of the abstract database record (the latter may be a null transformation). The element specification selects elements from the abstract database record, and possibly also specifies representations for those elements.

The target applies a record syntax to an abstract database record, resulting in an exportable structure referred to as a retrieval record.

3.1.6 Model of a Result Set

For generality, it is assumed that query processing does not necessarily require physical access to records; a result set is thus assumed to be the identification of (e.g. pointers to) records, as opposed to the actual set of records, selected by a query. (It is not assumed that the database records are locked. Methods of concurrency control, which would prevent modification or deletion of result set records, are not addressed by this standard.) A result set may be used as a selection mechanism for the transfer of records between systems; the result set itself is considered to be a purely local data structure and is not transferred (that is, records are transferred, but not the local pointers to the records).

For the purpose of retrieving records, the logical structure of a result set is that of a named, ordered list of triples consisting of:

(a) an ordinal number corresponding to the position of the triple in the list,
(b) a database name, and
(c) a unique identifier (of local significance only) of a record within the database named in (b).

A result set item is referenced by its position within the result set, that is, by (a).

For the purpose of searching, when a result set is used as an operand in a query, the logical structure is one of the following:

- Basic model — A set of pairs, each consisting of (b) and (c) of the above model for retrieval.
- Extended model — A set of triples, each consisting of (b) and (c) of the retrieval model; and including unspecified information associated with each record, which may be used as a surrogate for the search that created the result set.

Note: Query specifications may indicate that the basic model applies, or under what condition the extended model applies and the nature of the unspecified information. For the type-1 query, when version 2 is in effect, the basic model applies.

3.1.7 Extended Services

A class of services recognized by this standard, but which are not Z39.50 services, are referred to as extended services. The Extended Services (ES) service, is a Z39.50 service; an ES operation, results in the initiation of an extended services task. The task is not considered part of the Z39.50 ES operation.

An ES operation is initiated by the origin, via an ES request. The ES response, which completes the operation, does not (necessarily) signal completion of the task; it may indicate for example that the task has started or is queued (or it might indicate that the task has completed; in fact the ES request may indicate that the task should complete prior to the ES response). An ES task may have a lifetime beyond the Z-association.

Examples of extended services are:

- SaveResultSet
- SaveQuery
- PeriodicQuerySchedule
- ExportSpecification
- InvokeExportSpecification

For each extended service an abstract data structure is defined. Each ES task is represented by a database record, called a task package, maintained in a special database by the target, whose abstract structure is that defined for that extended service. This "extended services database" may be searched, and records retrieved, by the Z39.50 Search and Retrieval facilities. The origin may search for packages of a particular type, or created by a particular user, or of a particular status (i.e. pending, active, or complete), or according to various other criteria. In particular, the origin may search the database after submitting an ES request (during the same or a subsequent Z-association), for a resulting task package, to determine status information pertaining to the task, for example, to determine if the task has started.

The origin, via an ES request, may 'create', 'delete', or 'modify' a task package. When it creates a task package, it specifies a user-id, to be associated with the package, as well as 'permissions' for other user-ids. Permissions include 'create', 'delete', and 'modify', as well as 'present', and 'invoke'.

For example, when a user creates a task package of type SaveResultSet a (persistent) result set is created, represented by the created task package. When that package is "presented" (either in a the same or a subsequent Z-association), a copy of that persistent result set is made available to that Z-association as a Z39.50 result set (a result set name, for use during the Z-association, is included within the task package). When a user deletes the task package, the persistent result set is deleted.

As another example: when a user creates a task package of type SaveQuery, a (persistent) query is created, represented by the created task package. Another user may subsequently create a PeriodicQuerySchedule task package, which refers to the persistent query task package, only if that user has 'invoke' privilege for that persistent query. The periodicQuerySchedule task package includes an 'active' flag, indicating whether the periodic query is active. The user may subsequently 'modify' the task package (if the user has 'modify' privilege for the package) to turn the 'active' flag 'on' or 'off'. Similarly, a user may create an ExportSpecification (package) and another user (or the same user) may subsequently 'invoke' that ExportSpecification, by creating an InvokeExportSpecification package.

3.1.8 Explain

The origin may obtain details of the target implementation, including databases, attribute sets, diagnostic sets, record syntaxes, and element specifications supported. The origin obtains these details through the Z39.50 Explain facility. The target maintains this information in a database that the origin may access via the Z39.50 Search and Present facilities.

This "explain" database appears to the origin as any other database supported by the target, but it has a well-known name and a pre-defined record syntax. Also, certain search terms, corresponding to information categories, are predefined in order to allow a semantic level of interoperability. Each information category has its own record layout, and all are included in the Explain syntax.

3.2 Facilities of the Information Retrieval Service

Section 3.2.1 through 3.2.11 describe the eleven facilities of the Information Retrieval service. Most consists of logical groups of services; in several cases, a facility consists of a single service. Additional services may be added to any facility in future versions. Following is a summary description of the eleven facilities.

(1) **Initialization Facility**

Init Service: A confirmed service initiated by the origin to initiate an Init operation.

(2) **Search Facility**

Search Service: A confirmed service initiated by the origin to initiate a Search operation.

(3) **Retrieval Facility**

The Retrieval facility consists of two services.

- Present Service: A confirmed service initiated by the origin to initiate a Present operation.
- Segment Service: A non-confirmed service initiated by the target, during a Present operation.

  Note: A Present operation is initiated by the origin with a Present request, which is followed by zero or more Segment requests, and the operation is terminated by the target with a Present response.

(4) **Result-set-delete Facility**

Delete Service: A confirmed service initiated by the origin to initiate a Delete operation.

(5) **Browse Facility**

Scan Service: A confirmed service initiated by the origin to initiate a Scan operation.

(6) **Sort Facility**

Sort Service: A confirmed service initiated by the origin to initiate a Sort operation.

(7) **Access Control Facility**

Access-control service: a confirmed service initiated by the target. It does not initiate an operation, and it might or might not be part of an active operation.

(8) **Accounting/Resource Control Facility**

The Accounting/ Resource Control facility consists of three services.

- Resource-control Service: A conditionally confirmed service initiated by the target. It does not initiate an operation, and it might or might not be part of an active operation.
- Trigger-resource-control Service: A non-confirmed service initiated by the origin. It does not initiate an operation, and it might or might not be part of an active operation.
- Resource-report Service: A confirmed service initiated by the origin to initiate a Resource-report operation.

(9) Explain Facility

The Explain facility allows an origin to obtain details of the nature of the implementation of a target system, including the databases available for searching and features supported. The Explain facility does not consist of any services, but it uses the services of the Search and Retrieval facilities.

(10) Extended Services Facility

Extended-services Service: A confirmed service initiated by the origin to initiate an Extended-services operation.

Note: the Extended-services operation may result in the scheduling, initiation, and possibly completion of an extended services task at the target. The task is not part of the Extended-services operation, and is not part of the Z39.50 service.

(11) Termination Facility

Close Service: A confirmed service initiated by the origin or target. It does not initiate nor is it part of any operation. It allows an origin or target to abruptly terminate all active operations and to initiate termination of the Z-Association. (Following termination of the Z-Association the origin may subsequently attempt to initialize another Z-Association using the Init service.)

3.2.1 Initialization Facility

The Initialization facility allows the origin to establish a Z-association via the Init service.

3.2.1.1 Init Service

The Init service allows an origin to establish a Z-association. The origin proposes values for initialization parameters. The target may respond with alternative values for some of the parameters. If the target responds affirmatively (Result = 'accept'), the Z-association is established. If the origin then does not wish to accept the values in the target response, it may terminate the Z-association, via the Close service (and may subsequently attempt to initialize again). If the target responds negatively, the origin may attempt to initialize again.

Parameter	Origin Request	Target Response
Id/authentication	x (optional)	
Version	x	x
Options	x	x
Preferred-message-size	x	x
Maximum-record-size	x	x
Result		x
Implementation-id	x (optional)	x (optional)
Implementation-name	x (optional)	x (optional)
Implementation-version	x (optional)	x (optional)
User-information-field	x (optional)	x (optional)
Other-information	x (optional)	x (optional)
Reference-id	x (optional)	x (if appl)

3.2.1.1.1 Version Both the origin and target indicate all versions that they support. The highest common version is selected for use, and is said to be 'in force', for the Z-association. If there are no versions in common, the target should indicate 'reject' for the parameter Result.

Notes:

(1) Version numbers higher than the highest known version should be ignored.

(2) Versions 1 and 2 are identical. Systems supporting version 2 should indicate support for version 1 as well, for interoperability with systems that indicate support for version 1 only (e.g. ISO 10163 implementations).

3.2.1.1.2 Id/authentication The origin and target agree, outside the scope of the standard, whether or not this parameter is to be supplied by the origin, and if so, what the value is. This value is used by the target to determine if the origin is authorized to enter into communication with the target.

3.2.1.1.3 Options For each of the capabilities listed below, the origin proposes either 'on' or 'off' (meaning 'in effect' or 'not in effect' respectively) and the target responds correspondingly for each. The response determines whether the capability is in effect.

- search
- present
- delete
- resource-report
- scan
- sort
- extended-services
- trigger-resource-control
- level 1 segmentation
- level 2 segmentation
- concurrent operations
- named result sets
- resource-control
- access-control

Note: the above list of capabilities is subject to expansion in future versions of this protocol.

The following rules, describing how these capabilities are to be negotiated, are intended to allow interoperation even when the origin and target have not necessarily implemented the same capabilities.

The Options parameter consists of a string of boolean flags, each corresponding to an individual capability. The origin might set the flag to 'in effect' for a capability unknown to the target. In that case it is recommended that the target set the corresponding flag to 'not in effect' in the response. However, if the origin sets a flag to 'not in effect' and the target sets the corresponding flag to 'in effect', and if the origin is not aware what capability that flag represents, it is recommended that the origin terminate the Z-association.

<u>search, present, delete, resource-report, scan, sort, and extended-services</u> - for each operation type, the origin indicates whether it wishes to initiate operations of that type; if so, the target indicates whether it is willing to process an operation of that type. If the origin proposes 'not in effect' for a particular operation type, the target must also specify 'not in effect'.

Notes:

(1) The target indication that it is willing to process a Resource-report operation does not imply that it will include a resource report in the response.

(2) Any of the above operation types may be negotiated for any version. In particular, Scan, sort, and Extended-services may be negotiated when version 2 is in effect, even though they are not defined in ANSI Z39.50-1992.

<u>trigger-resource-control</u> - The origin may propose to submit Trigger-resource-control requests; if so, the target indicates whether it will accept Trigger-resource-control requests. If origin proposes 'not in effect', the target must also specify 'not in effect'.

Notes:

(1) If the target specifies 'in effect' for Trigger-resource-control, but 'not in effect' for 'resource-control', than the origin may use only the Cancel function of Trigger-resource-control.

(2) The target may indicate unwillingness to accept Trigger-resource-control requests even if it specifies 'in effect' for 'resource-control'.

(3) An indication by the target of willingness to accept Trigger-resource-control requests does not imply that the target will take any action as a result of a Trigger-resource-control request.

<u>level 1 and level 2 segmentation</u> - The origin proposes one of the following:

- "no segmentation", by specifying 'not in effect' for both level 1 and level 2 segmentation;
- "level 1 segmentation", by specifying 'in effect' for level 1 and 'not in effect' for level 2 segmentation; or
- "level 2 segmentation", by specifying 'in effect' for level 2 segmentation.

Notes:

(1) If the origin proposes 'in effect' for level 2 segmentation then it may also propose 'in effect' for level 1 segmentation to indicate that if the target is unable to support level 2 segmentation, the origin wishes level 1 segmentation to be in effect.

(2) "segmentation" is said to be 'in effect' if either level 1 or level 2 segmentation is in effect.

(3) Segmentation may be in effect only when version 3 is in force.

The target response indicates which, if either, form of segmentation it intends to perform.

- If the target specifies neither level 1 nor level 2 then 'no segmentation' is in effect, regardless or what the origin has proposed.
- If the target specifies level 1 (but not level 2) segmentation, it will not perform level 2 segmentation, and the origin must be prepared to accept level 1 segmentation, regardless of what the origin has proposed.

- If the target specifies level 2 segmentation, the origin must be prepared to accept level 2 segmentation regardless of what it has proposed (the target value for level 1 should be 'not in effect').

When 'no segmentation' is in effect, the target response to a Present request must consist of a single message (or "segment"), containing an integral number of records. When 'Level 1 segmentation' is in effect the target may respond to a Present request with multiple segments, each containing an integral number of records. When 'Level 2 segmentation' is in effect the target may respond to a Present request with multiple segments, and individual records may span segments. Segmentation procedures are detailed in 3.3.

<u>concurrent-operations</u> - The origin may propose to initiate concurrent operations; if so, the target indicates whether it will accept concurrent operations. If the origin proposes 'not in effect', the target must also specify 'not in effect'. If concurrent operations is not in effect, then 'serial operations' is said to be in effect. Concurrent operations may be in effect only when version 3 is in force.

<u>named-result-sets</u> - The origin may propose to use named-result sets (i.e. to specify result set names other than "default" as the value of Result-set-name within a Search request); if so, the target specifies whether it will support named-result-sets. If the origin proposes 'not in effect', the target must also specify 'not in effect'.

<u>resource-control and access-control</u> - The origin indicates whether it wishes the target to invoke Resource-control and/or Access-control (i.e. send Resource-control and/or Access-control requests). The target specifies whether it plans to invoke Resource-control and/or Access-control.

Notes:

(1) If the target specifies 'not in effect' for resource-control (or access-control) then it will not invoke resource-control (or access control) even if the origin has proposed 'in effect'.

(2) If the origin proposes 'not in effect' for resource-control, and the target indicates 'in effect' for resource-control, indicating that it is not willing to suppress Resource-control requests, and if indeed the origin cannot accept Resource-control requests, the origin should terminate the Z-association.

(3) If the origin proposes 'not in effect' for access-control, and if security requirements on the target system mandate that security (other than that which might be provided by the parameter Id/authentication) be invoked at the outset of a Z-association, then the target should reject the Z-association (by setting the parameter Result to 'reject', and specifying 'in effect' for 'access-control').

Security may be invoked at different levels. In addition to user authentication at the outset of a Z-association, security might be invoked to control access to a particular database, record, result-set, or use of an operation (the target might invoke a security challenge during an Init operation, to determine if the origin is authorized to use a capability it has proposed). It might be used to determine whether an origin is authorized to request a resource report.

Thus if the origin proposes 'not in effect' for access-control, and the target invokes security, but not at the Z-association level, then the target may choose to accept the Z-association. In that case, if the origin subsequently initiates an operation that would precipitate an Access-control request, the target agrees to suppress the request and respond with an error status indicating that a security challenge was required but could not be issued.

3.2.1.1.4 Preferred-message-size and Maximum-record-size The Init request contains the origin's proposed values of Preferred-message-size and Maximum-record-size, specified in bytes. The Init response contains the Preferred-message-size and Maximum-Record-size that the target is going to use; these may be different from (and override) the values proposed by the origin. For both the request and response, Preferred-message-size must be less than or equal to Maximum-record-size. The usage of these parameters is specified in 3.3.

3.2.1.1.5 Result The target indicates whether or not it accepts the Z-association by specifying a value of 'accept' or 'reject' in the parameter Result. if 'reject' is indicated, the origin may send another Init request.

3.2.1.1.6 Implementation-id, Implementation-name, and Implementation-version The request or response may optionally include any of these three parameters. They are, respectively, an identifier (unique within the client or server system), descriptive name, and descriptive version, for the origin or target implementation. These three implementation parameters are provided solely for the convenience of implementors, for the purpose of distinguishing implementations.

| **3.2.1.1.7 User-information-field** This parameter may be used by either the origin or target service users for additional information, not specified by this standard.

3.2.1.1.8 Other-Information This parameter may be used by the origin or target for additional information, not specified by the standard. This parameter may be used only if version 3 is in force.

> Note: Care should be taken by the origin when using this parameter; the origin cannot ascertain that version 3 is in force before sending the Init request.

3.2.1.1.9 Reference-id See 3.4.

3.2.2 Search Facility

The Search facility enables an origin to query databases at a target system, and to receive information about the results of the query.

3.2.2.1 Search Service

The Search service allows an origin to send a query to a target. The basic query concept is: "from the specified collection of databases identify those records with the properties indicated." The target creates a "result set" which represents the set of records identified by the query. The target maintains the result set for subsequent retrieval requests; however, depending on the parameters of the search, one or more items identified by the result set may be immediately returned as part of the search response. The result set is an ordered set; records identified by entries in the result set are referenced by the position of the entry within the result set, beginning with one (1).

Parameter	Origin Request	Target Response
Query-type	x	
Query	x	
Database-names	x	
Result-set-name	x	
Replace-indicator	x	
Small-set-element-set-names	x (opt.)	
Medium-set-element-set-names	x (opt.)	
Preferred-record-syntax	x (opt.)	
Small-set-upper-bound	x	
Large-set-lower-bound	x	
Medium-set-present-number	x	
Response-records		x (if appl.)
Result-count		x
Number-of-records-returned		x
Next-result-set-position		x
Search-status		x
Result-set-status		x (if appl.)
Present-status		x (if appl.)
Additional-search-information	x (opt.)	x (opt.)
Other-information	x (opt.)	x (opt.)
Reference-id	x (opt.)	x (if appl.)

3.2.2.1.1 Query-type and Query The parameter Query-type identifies the syntax of the query. As noted above, the basic query concept is "from the specified set of items, identify those with the properties indicated." The "specified set of items" is a collection of one or more databases, specified by the parameter database-names. The "properties indicated" are specified by the parameter Query.

Five types are defined:

- type-0 may be used only when the origin and target have an a priori agreement outside of the standard;
- type-1 is the Reverse Polish Notation (RPN) query specified in 3.7;
- type-2 is the ISO8777 type query, specified in ISO 8777;
- type-100 is the Z39.58 type query, specified in ANSI Z39.58; and
- type-101 is the extended RPN (ERPN) query, an extension to the Type-1 query to allow proximity searching and restriction of result sets by attributes. It is specified in 3.7.

3.2.2.1.2 Database-names The target designates what databases may be specified on a Search request, and in what combinations they may be specified.

Notes:

(1) The target designates this information through the Explain facility or through some mechanism outside of the standard.

(2) The target designates this information by specifying names for the databases. Each database name is a string of characters, and the string is case-insensitive (that is, for any character which is a letter, the origin may use either upper- or lower-case, regardless of how the target has specified the name).

For example, a target might specify that databases **A**, **B**, and **C**, may be searched individually, and that **A** and **B** may be searched in combination (but not **A** and **C**, nor **B** and **C**). If an origin requests a combination of databases which is not supported, the search will result in a diagnostic such as "combination of specified databases not supported" (see appendix ERR).

3.2.2.1.3 Result-set-name and Replace-indicator The parameter Result-set-name specifies a name to be given to the result set which will be created by the query so that it may be subsequently referenced within the same Z-association.

If a result set with the same name already exists at the target, the action taken depends on the value of the parameter Replace-indicator, as follows:

- If the value of Replace-indicator is 'on', then after processing the query, the existing result set whose name is specified by the parameter Result-set-name will be deleted, and a new result set by that name created. If the search cannot be processed, the content of the result set will be empty.
- If the value of Replace-indicator is 'off', the search is not processed, an error diagnostic is returned by the target, and the existing result set whose name is specified by the parameter Result-set-name is left unchanged.

If a result set does not exist with the name specified by the parameter Result-set-name, then a result set by that name is created by the target, and the parameter Replace-indicator is ignored. The initial content of the result set is empty. If no records are found by the query, the result set remains empty.

A target system need not support, in general, the naming of result sets by the origin. However, a target system must support at least the result set whose name is "default". If the origin specifies "default" as Result-set-name, then Replace-indicator must be 'on'.

A result set which is created by a Search request (that is, specified by the parameter Result-set-name) may be referenced in a subsequent Present request or as an operand in a subsequent Search request (for example, in a Type-1 query). If a result set named "default" is created, it remains available for reference from the time it is created until the end of the Z-association during which it is created, or until either:

- another default result set is created, because the name "default" is specified as Result-set-name in a subsequent Search request, or
- it is unilaterally erased or deleted by the target.

Any result set other than the result set named "default" remains available for reference from the time it is created until it is deleted in one of the following ways:

- by a Delete request;
- implicitly, because a result set was specified by the same name in a Search request, and the value of the parameter Replace-indicator was 'on';
- unilaterally by the target (at any time); or
- by termination of the z-association.

3.2.2.1.4 Small-set-element-set-names and Medium-set-element-set-names These parameters describe the preferred composition of the records expected in the search response. If the query results in a small-set (see 3.2.2.1.6), then Small-set-element-set-names pertains. If the query results in a medium-set, then Medium-set-element-set-names pertains. These two parameters are described in 3.6.2.

3.2.2.1.5 Preferred-record-syntax The origin may specify a preferred record syntax for retrieval records, from the set of abstract syntaxes for which Presentation contexts are currently established for this A-association. If the target cannot supply a particular record according to the Preferred-record-syntax, it supplies the record according to one of the other abstract syntaxes from the established set. If the target cannot supply the record according to any of the abstract syntaxes from the established set, it returns a surrogate diagnostic for that record, unless the established set is empty; in that case, this standard does not prescribe the target action.

3.2.2.1.6 Small-set-upper-bound, Large-set-lower-bound, and Medium-set-present-number The result set is considered a "small-set", "medium-set", or "large-set", depending on the values of parameters Small-set-upper-bound and Large-set-lower-bound of the Search request, and Result-count of the Search response (see 3.2.2.1.8). The result set is a small-set if Result-count is not greater than small-set-upper-bound. The result set is a large-set if Result-count is larger than or equal to Large-set-lower-bound. Otherwise, the result set is a medium-set. If the query results in a small-set, response records corresponding to all database records identified by the result set are to be returned to the origin (subject to possible message size constraints). If the query results in a large-set, no response records are to be returned. If the query results in a medium-set, the maximum number of response records to be returned is specified by Medium-set-present-number.

Notes:

(1) The result set may be a medium-set only when Result-count is greater than small-set-upper-bound and less than Large-set-lower-bound, and this can occur only if Large-set-lower-bound is at least 2 greater than Small-set-lower-bound; i.e., the result set cannot be a medium-set if Large-set-lower-bound exceeds Small-set-upper bound by one. For example, if Large-set-lower-bound is 11 and Small-set-upper-bound is 10, the intent is "if 10 or less database records are found return response records for them all, otherwise do not return any", and medium-set-present-number would not apply.

(2) Small-set-upper-bound may be zero. Large-set-lower-bound must be greater than Small-set-upper-bound.

(3) If the origin does not want any response records returned regardless of the value of Result-count, Large-set-lower-bound should be set to 1 and Small-set-upper-bound to zero.

3.2.2.1.7 Response-records The target processes the search, creating a result set which identifies a set of database records. It cannot be assumed however that search processing requires physical access to the database records. A particular database record might not be accessible but this circumstance might not be recognized until an attempt is made to access the record for the purpose of forming a retrieval record.

After processing the search, the target attempts to create retrieval records to be included in the Search response, corresponding to the first N database records identified by the result set, (N depends on the search parameters and result-count, as described in 3.2.2.1.6). For each database record for which a retrieval record cannot be included, a surrogate diagnostic record is substituted.

The term <u>response record</u> refers to a retrieval record or a surrogate diagnostic record. The parameter Response-records is one of the following:

- N response records;
- a number of response records, which is less than N because of message size constraints (see 3.3);
- one or more non-surrogate diagnostic records (see note) indicating that the search cannot be processed, and why it cannot be processed; or
- one or more non-surrogate diagnostic records (see note) indicating that records cannot be presented, and why not, e.g. "element set name not valid for database".

  Note: If version 2 is in force, the target returns a single non-surrogate diagnostic record. If version 3 is in force, the target returns one or more non-surrogate diagnostic records.

The order of occurrence of response records within the parameter Response-records is according to the order in which they are identified by the result set. Each may optionally be accompanied by the name of the database in which the record resides. However, the database name must accompany the first response record being returned, and must accompany any record from a database different than its immediate predecessor.

3.2.2.1.8 Result-count and Number-of-records-returned The parameter Result-count is the number of database records identified by the result set. If the result set is empty, result-count is zero. The parameter Number-of-records-returned is the total number of response records returned in the Search response.

3.2.2.1.9 Next-result-set-position The value M+1, where M is the position of the result set item which identifies the database record corresponding to the last response record among those returned; or zero if M = Result-count.

3.2.2.1.10 Search-status The parameter Search-status, returned in the response, assumes one of the following two values:

success	-	The search completed successfully.
failure	-	The search did not complete successfully.

A value of "success" does not imply that the expected response records are being returned as part of the response (see Present-status, 3.2.2.1.11). Note also, a value of "success" does not imply that any database records were located by the search. A value of "failure" does imply that none of the expected response records is being returned. In the latter case, the target returns one or more non-surrogate diagnostic records (see note) indicating that the search cannot be processed.

> Note: If version 2 is in force, the target returns a single non-surrogate diagnostic record. If version 3 is in force, the target returns one or more non-surrogate diagnostic records.

3.2.2.1.11 Result-set-status and Present-status These are status descriptors necessary to dissambiguate certain situations that can occur during search and present operations.

Result-set-status occurs if and only if the value of Search-status is "failure", and its value is one of the following:

subset	-	Partial, valid results available.
interim	-	Partial results available, not necessarily valid.
none	-	No results available (result set is empty).

Present-status occurs if and only if the value of Search-status is "success", and its value is one of the following:

success	-	All of the expected response records are available.
partial-1	-	Not all of the expected response records can be returned because the request was terminated by access-control.
partial-2	-	Not all of the expected response records can be returned because they will not fit within the preferred message size.
partial-3	-	Not all of the expected response records can be returned because the request was terminated by resource-control at origin.
partial-4	-	Not all of the expected response records can be returned because the request was terminated by resource-control at target.
failure	-	None of the expected response records can be returned. One or more non-surrogate diagnostic records is returned (see note in 3.2.2.1.7).

3.2.2.1.12 Additional-Search-Information On the response the target may use this parameter to convey information which is a by-product of the search process, including, for example, intermediate result counts, why particular records were returned, or whether a particular attribute was used for searching a database. On the request the origin may use this parameter to indicate the format or content of that information. However, the format and content of the information returned by the target in the response is always determined by the target, whether or not the parameter is included in the request. User Information formats SearchRequest-1 and SearchResponse-1 are defined in appendix USR.

This parameter may be used only when version 3 is in force.

3.2.2.1.13 Other-Information This parameter may be used by either the origin or target for additional information, not specified by the standard. This parameter may be used only when version 3 is in force.

3.2.2.1.14 Reference-id See 3.4

3.2.3 Retrieval Facility

The Retrieval facility enables the origin to send a Present request to the target, to request response records according to position within a result set maintained by the target. The target responds by sending a Present response, containing the requested response records. Alternately, if segmentation is in effect and the requested response records will not fit within the Present response message, the target may segment the response by sending one or more Segment requests (see 3.2.3.2) before the Present response. The procedures for segmentation are described in 3.3.

The Segment requests (if any) together with the Present response are referred to as the "aggregate present response". Each Segment request as well as the Present response is referred to as a "segment" of the Present response. If an aggregate Present response consists of a single segment (i.e. only a Present response) it is called a "simple Present response".

3.2.3.1 Present Service

The Present service allows the origin to request response records corresponding to database records represented by a specified result set. Database records are referenced by relative position within the result set. The origin specifies a range and may follow with subsequent requests specifying different ranges.

Note: If version 3 is in force, a single request may include more than one range.

The origin may request, for example, records one through five and follow with a request for records four through six.

Note: In this section, "record N" means "the response record corresponding to the database record identified by result set entry N".

Parameter	Origin Request	Target Response
Number-of-records-requested	x	
Result-set-start-position	x	
Additional-ranges	x (opt.)	
Result-set-id	x	
Element-set-names	x (opt.)	
Preferred-record-syntax	x (opt.)	
Comp-spec	x (opt.)	
Max-segment-count	x (opt.)	
Max-segment-size	x (opt.)	
Max-record-size	x (opt.)	
Response-records		x (if appl.)
Number-of-records-returned		x
Next-result-set-position		x
Present-status		x
Other-information	x (opt.)	x (opt.)
Reference-id	x (opt.)	x (if appl.)

3.2.3.1.1 Number-of-records-requested and Result-set-start-position The origin requests a range of records: N records beginning at record M. M = Result-set-start-position, N = Number-of-records-requested and N is not greater than (Result-count - M) + 1.

3.2.3.1.2 Additional-ranges The origin may request additional ranges of records by including this parameter, which consists of one or more pairs (M, N) where M and N are as described in 3.2.3.1.1. For the first pair (M, N) M must be greater than or equal the sum of Result-set-start-position and

Number-of-records-requested. For any consecutive pairs (M1, N1) and (M2, N2), M1 + N1 must be less than M2. This parameter may occur only when version 3 is in effect.

3.2.3.1.3 Result-set-id Result-set-id specifies the result set from which records are to be retrieved. It is the result set created by a previous Search request for which the value of the parameter Result-set-name matches the value of Result-set-id.

3.2.3.1.4 Element-set-names See 3.6.2.

3.2.3.1.5 Preferred-record-syntax See 3.2.2.1.5.

3.2.3.1.6 Comp-spec This parameter may occur only if the parameter Element-set-names is omitted, and only if version 3 is in force. If so, it provides an alternative means of specifying the desired composition of retrieved records. See 3.6.

3.2.3.1.7 Max-segment-count, Max-segment-size, and Max-record-size Max-segment-count may be included when segmentation is in effect. Max-segment-size and/or Max-record-size may be included when level 2 segmentation is in effect. The use of these parameters is detailed in 3.3. They may be used only when version 3 is in force.

3.2.3.1.8 Response-records This parameter consists of a sequence of response records, or possibly, if 'level 2 segmentation' is in effect, a final fragment (see 3.3.3) followed by zero or more response records.

Alternately (if the operation included no Segment requests) the parameter consists of one or more non-surrogate diagnostic records indicating that the request cannot be processed, and why not (see note below).

A response record will be returned in the aggregate Present response for each record requested in the request (subject to message size, access-control, and resource-control constraints). Each response record corresponds to a result set entry, and the result set ordinal positions represented by the response records will be ascending and consecutive, unless the request included the parameter Additional-ranges, in which case, the positions will be ascending but may have gaps, but those gaps will correspond exactly to the gaps in the requested ranges.

Each response record may optionally be accompanied by the name of the database to which it corresponds. However, the database name must accompany the first response record (or starting fragment) within the first segment of the aggregate Present response, and must accompany any response record (or starting fragment of a response record) from a database different from its immediate predecessor within the aggregate Present response.

When the origin has received the aggregate Present response, the result (if all of the segments are re-assembled, and segmented response records re-assembled from their fragments) will be one of the following:

- N response records, where N = Number-of-records-requested,
- a number of response records, which is less than N (reason specified by Present-status), or
- one or more diagnostic records (see note) indicating that the request cannot be processed, and why not.

  Note: If version 2 is in force, the target returns a single non-surrogate diagnostic record. If version 3 is in force, the target returns one or more non-surrogate diagnostic records.

3.2.3.1.9 Number-of-records-returned and Next-result-set-position The parameter Number-of-records-returned is the total number of response records in the aggregate Present response. Next-result-set-position is the value M+1, where M is the position of the result set item corresponding to the last record among those included in the response; or zero if M is the position of the last result set item.

3.2.3.1.10 Present-status Present-status is mandatory in a Present response and its values are the same as those listed for Present-status in 3.2.2.1.11. Present-status refers to the aggregate Present response.

3.2.3.1.11 Other-information Additional information, not specified by the standard. This parameter may be used only when version 3 is in force.

3.2.3.1.12 Reference-id See 3.4.

3.2.3.2 Segment Service

If the records requested by a Present request will not fit in a single segment, and if segmentation is in effect, the target returns multiple segments, each of which contains a portion of the records. Each except the last segment is returned as a Segment request (the last segment is returned as a Present response).

Notes:

(1) This service may be used only when version 3 is in force.

(2) If segmentation is not in effect, the target does not send any Segment requests and the aggregate Present response consists of a simple Present response. If the records requested will not fit in a segment, the procedures described in 3.3.1 apply.

(3) If the records requested will fit in a single segment (whether or not segmentation is in effect) the target does not send any Segment requests and the aggregate Present response consists of a simple Present response.

Parameter	Target Request
Segment-records	x
Number-of-records-returned	x
Other-information	x (optional)
Reference-id	x (if applicable)

3.2.3.2.1 Segment-records If level 1 segmentation is in effect, the parameter Segment-records consists of a sequence of response records.

If level 2 segmentation is in effect, the parameter Segment-records consists of response records and fragments (see 3.3.3) of a retrieval record (note: a diagnostic record may not be segmented). It may be composed of a final fragment (except within the first segment of the aggregate Present response), followed by zero or more response records, followed by a starting fragment. Neither fragment need occur, however if neither occurs there must be at least one response record.

The order of occurrence of a response record or fragment of a retrieval record is according to the order in which the record is identified by the result set. Each response record or starting fragment may optionally be accompanied by the name of the database to which it pertains. However, the database name must accompany the first response record (or starting fragment) within the first segment of the aggregate Present response, and must accompany any response record (or starting fragment of a record) from a database different from its immediate predecessor within the aggregate Present response.

3.2.3.2.2 Number-of-records-returned The parameter Number-of-records-returned is the total number of response records and starting fragments included within the segment.

3.2.3.2.3 Other-information Additional information, not specified by the standard.

3.2.3.2.4 Reference-id See 3.4.

3.2.4 Result-set-delete Facility

The Result-set-delete facility enables an origin to instruct a target to delete one or more result sets known to the target. The target responds with information pertaining to the result of the operation.

3.2.4.1 Delete Service

The Delete service enables an origin to send a Delete request to the target. The origin may request deletion of specified result sets held by the target or all result sets currently on the target created during the Z-association. The target responds by reporting the status of the delete operation.

Parameter	Origin Request	Target Response
Delete-function	x	
Result-set-list	x (if appl.)	
Delete-operation-status		x
Delete-list-statuses		x (if appl.)
Number-not-deleted		x (if appl.)
Bulk-statuses		x (if appl.)
Delete-msg		x (opt.)
Other-information	x (opt.)	x (opt.)
Reference-id	x (opt.)	x (if appl.)

3.2.4.1.1 Delete-function The origin specifies one of the following:

list	-	delete one or more specified result sets (see 3.2.4.1.2), or
bulk-delete	-	delete all result sets currently on the target system created during this Z-association.

3.2.4.1.2 Result-set-list If Delete-function is 'list', then the parameter Result-set-list must be present. It specifies a list of result sets to be deleted, which were created by previous Search operations (in which the value of the parameter Result-set-name matches the value of one of the members of the list) which completed during this Z-association. If Delete-function is 'bulk-delete', then the parameter Result-set-list must not be present.

3.2.4.1.3 Delete-operation-status Delete-operation-status is used by the target to report the status of the delete request. It assumes one of the values "success" or "failure-3" through "failure-9" in the table below.

3.2.4.1.4 Delete-list-statuses Delete-list-statuses is present in a Delete response if the Delete-function in the request was 'list'. It contains the same Result-set- id(s) contained in the Result-set-list parameter of the corresponding Delete request. Each result-set-id in the Delete-list-statuses parameter is paired with a status. Possible status values are 'success,' 'failure-1' through 'failure-6', and 'failure-10'.

Status	**Description**
success	Result set(s) deleted.
failure-1	Result set did not exist.
failure-2	Result set previously unilaterally deleted by target.
failure-3	System problem at target (optional text message may be included in the Delete-msg parameter).
failure-4	Access-control failure: the delete request caused the target to issue an Access-control request which the origin failed to satisfy, or the origin could not accept an Access-control request.
failure-5	Request terminated by origin through resource control.
failure-6	Access terminated by target due to resource constraints.
failure-7	Bulk delete of result sets not supported by target.
failure-8	Not all result sets deleted (on a bulk-delete request) (see 3.2.4.1.5).
Failure-9	Not all requested result sets deleted (on a list request).
Failure-10	Result-set in use.

Notes:

(1) failure-7 and failure-8 can occur only if Delete-operation is Bulk-delete.

(2) failure-10 may be used only when version 3 is in force.

3.2.4.1.5 Number-not-deleted and Bulk-statuses These two parameters occur only if Delete-operation is Bulk-delete and if Delete-operation-status = "failure-8". The parameter Number-not-deleted indicates how many result sets were not deleted, and the parameter Bulk-statuses gives individual statuses for those not deleted.

Note, however, that a target is not obligated to provide statuses for each result set not deleted on a bulk delete. For example, a target may abort a bulk delete when the first failure to delete a result set that is part of the bulk delete fails; in this case only a single status might be included in the parameter Bulk-statuses.

If a bulk delete results in more statuses than can fit into a single Delete-response message, the target may discard those which do not fit.

3.2.4.1.6 Delete-msg Delete-msg, if present, contains an optional text message.

3.2.4.1.7 Other-Information This parameter may be used by either the origin or target for additional information, not specified by the standard. This parameter may be used only when version 3 is in force.

3.2.4.1.8 Reference-id See 3.4.

3.2.5 Access Control Facility

The Access-control facility allows a target to challenge an origin. The challenge might pertain to a specific operation or to the Z-association. The Access-control request/response mechanism can be used to support password challenges, public key cryptosystems, algorithmic authentication, etc.

3.2.5.1 Access-control Service

An origin must be prepared to accept and respond to Access-control requests from the target if access control is in effect. A target may issue an Access-control request which is either part of a specific (active) operation or pertains to the Z-association.

- If concurrent operations is in effect:
  - o If the Access-control request includes a Reference-id: The supplied Reference-id must correspond to an active operation; the Access-control request is part of that operation. The Access-control response must also include that Reference-id.
  - o If the Access-control request does not include a Reference-id: The Access-control request and response are not part of any operation, they pertain to the Z-association.
- If serial operations is in effect: The target may issue an Access-control request only when there is an active operation; the Access-control request and subsequent response are part of that operation and must include the Reference-id of the operation (which is assumed 'null' if not present in the initiating request).

The following procedures pertain to access control as it applies to an operation:

After sending an initiating request, the origin must be prepared to receive an Access-control request (for that operation), respond with an Access-control response, then later receive another Access-control request, etc., before receiving a terminating response. The target might suspend processing of the operation from the time that it sends the Access-control request until it receives the Access-control response. The challenge does not interrupt any other operation. If the origin response is acceptable to the target, the operation proceeds as if the challenge has never taken place. If the origin fails to respond correctly to the challenge then the target's terminating response to the interrupted operation may indicate that the operation was terminated due to an Access-control failure.

Note: During a Search or Present operation, while the target is preparing records for presentation, it might send an Access-control request pertaining to a particular record. If the origin fails to respond correctly to the challenge, the target may simply substitute a surrogate diagnostic: "security challenge failed; record not included".

If the origin fails to respond correctly to the challenge during an Init operation, the target may set the Result parameter to 'reject' and may (optionally) supply such an indication in the User-information-field of the Init response.

The following procedures pertain to access control as it applies to the Z-association:

If concurrent operations is in effect, following initialization, the origin must be prepared at any time during the association, whether or not there are active operations, to receive an Access-control request pertaining to the A-association, respond with an Access-control response, then later receive another Access-control request, etc.

The target might suspend processing of some or all of the active operation from the time that it sends the Access-control request until it receives the Access-control response. If the origin response is acceptable to the target, the suspended operations proceed as if the challenge had never taken place.

If the origin fails to respond correctly to the challenge, the target might decide to terminate one or more operations but leave open the Z-association. In that case, the target's terminating response to any such operations may indicate that the operation was terminated due to an Access-control failure. Alternately the target may close the Z-association.

Parameter	Target Request	Origin Response
Security-challenge	x	
Security-challenge-response		x
Other-information	x (opt.)	x (opt.)
Reference-id	x (if appl.)	x (if appl.)

3.2.5.1.1 Security-challenge and Security-challenge-response Definitions for format and content of the challenge and response are subject to registration; several definitions are defined and registered in appendix ACC of this standard. Alternately the contents of these two parameters may be established by prior agreement between a given target/origin pair.

3.2.5.1.2 Other-information This parameter may be used by either the origin or target for additional information, not specified by the standard. This parameter may be used only when version 3 is in force.

3.2.5.1.3 Reference-id If serial operations is in effect, or if concurrent operations is in effect and the challenge pertains to a particular operation, then the use of Reference-id is governed by section 3.4. If 'concurrent operations' is in effect and the challenge pertains to the Z-association, then the Reference-id is to be omitted from both the request and response.

3.2.6 Accounting/Resource Control Facility

The Accounting/Resource Control facility consists of the Resource-control service (invoked by the target as part of an active operation), the Trigger-resource-control service (invoked by the origin as part of an active operation), and the Resource-report service (invoked by the origin to initiate an operation).

The Resource-control service permits the target to send a Resource-report, notifying the origin when either actual or predicted resource consumption will exceed agreed upon limits (or limits built into the target), and to obtain the origin's consent to continue an operation. In addition, the target can inform the origin about the current status of a result set being generated on the target during a Search operation, and indicate information about the status of the operation.

The Trigger-resource-control service permits the origin to request that the target initiate the Resource-control service, or cancel the operation.

the Resource-report service permits the origin to request that the target send a Resource-report pertaining to a completed operation or to the Z-association.

3.2.6.1 Resource-control Service

An origin must be prepared to accept and respond to Resource-control requests from the target if resource control is in effect. A target may issue a Resource-control request which is either part of a specific (active) operation or pertains to the Z-association.

- If concurrent operations is in effect:
    - o If the Resource-control request includes a Reference-id: The supplied Reference-id must correspond to an active operation; the Resource-control request is part of

that operation. The Resource-control response (if any) must also include that Reference-id.

- o If the Resource-control request does not include a Reference-id: The Resource-control request and response are not part of any operation, they pertain to the Z-association.

- If serial operations is in effect: The target may issue a Resource-control request only when there is an active operation; the Resource-control request and (possible) subsequent response are part of that operation and must include the Reference-id of the operation (which is assumed 'null' if not present in the initiating request).

The Resource-control request indicates whether a response is required:

- If so, the origin must issue a Resource-control response. If the Resource-control request was part of an operation: the response is part of the same operation; the target awaits the Resource-control response, and subsequently issues a terminating response after processing of the operation is concluded.
- If not, the origin must not issue a Resource-control response. If the Resource-control request was part of an operation: the target subsequently issues the terminating response, after processing of the operation is concluded.

An origin should be prepared to receive, and (conditionally) respond to, multiple Resource-control requests as part of an operation (while the operation is active), or pertaining to the Z-association.

If the origin responds to a Resource-control request with a Resource-control response saying to terminate an operation, it can expect to receive a terminating response. This response might indicate that the operation was terminated at origin request. However, the response might alternately indicate that the operation completed, since the operation at the target may continue to execute and subsequently complete before the Resource-control response reaches the target.

Parameter	Target Request	Origin Response
Resource-report	x (opt.)	
Partial-results-available	x (if appl.)	
Suspended-flag	x (if appl.)	
Response-required	x	
Triggered-request-flag	x (opt.)	
Continue-flag		x
Result-set-wanted		x (if appl.)
Other-information	x (opt.)	x (opt.)
Reference-id	x (if appl.)	x (if appl.)

3.2.6.1.1 Resource-report May be used to convey information about the current and estimated resource consumption at the target system. The format of Resource-report resource-1 is defined in Appendix RSC.

3.2.6.1.2 Partial-results-available The target indicates the status of the result set via the flag Partial-results-available, whose value is one of the following:

subset	-	Partial, valid results available.
interim	-	Partial results available, not necessarily valid.
none	-	No results available.

This parameter is meaningful only as part of a search operation. If its value is 'subset' or 'interim', then the target will accept subsequent Present requests against the result set if the origin indicates (via the Continue-flag) that the operation is to be terminated and if the value of the parameter Result-set-wanted is 'on'.

If the value of Partial-results-available is 'none' then the target need not accept subsequent Present requests in the event that the origin indicates (via the Continue-flag) that the operation is to be terminated.

Note that if Suspended-flag is off, the partial results available situation may change since processing of the Search operation may continue. In all cases, the values of Search-status and Result-set-status in the Search response should be treated as the authoritative information.

3.2.6.1.3 Suspended-flag This parameter is valid only when the request pertains to an operation. The target indicates whether or not processing of the operation has been suspended pending the Resource-control response. This flag occurs if and only if the value of Response-required is 'yes'.

3.2.6.1.4 Response-required The target indicates whether or not a response (from the origin) to this request is required.

3.6.2.1.5 Triggered-request-flag This parameter is valid only when the request pertains to an operation. The target may optionally indicate whether or not this request resulted from a Trigger-resource-control request from the origin.

3.2.6.1.6 Continue-flag This parameter is valid only when the request pertained to an operation. The origin indicates to the target whether or not to continue processing the operation.

3.2.6.1.7 Result-set-wanted This flag is meaningful only:

- during a Search operation,
- when the value of Partial-results-available is 'subset' or 'interim', and
- when the value of the parameter Continue-flag is 'do not continue'.

If the value of this flag is 'on', the target will maintain the (possibly partial) result set for subsequent Present operations. If the value of the flag is 'off', the target may delete the result set. A result set status of 'none' on the subsequent Search response indicates that the target has discarded the result set. In all cases, the values of Search-status and Result-set-status in the Search response describe the actual decisions made by the target and the way in which the search terminated.

3.2.6.1.8 Other-information This parameter may be used by either the origin or target for additional information, not specified by the standard. This parameter may be used only when version 3 is in force.

3.2.6.1.9 Reference-id See 3.4.

3.2.6.2 Trigger-resource-control Service

An origin may issue Trigger-resource-control requests during an operation (except during an Init operation), as part of that operation. It serves as a signal to the target that the origin wishes the target to:

a) simply send a Resource-report -- i.e. issue a Resource-control request with Response-required 'off';

b) invoke full resource control -- i.e. issue a Resource-control request with Response-required 'on'; or

c) cancel the operation.

The target is not obligated to take any specific action upon receipt of a Trigger-resource-control request. For the purpose of procedure description, there is no response to the request; if the target wishes to issue a Resource-control request it does so unilaterally. (If the origin issues a Trigger-resource-control request and subsequently receives a Resource-control request as part of the same operation, the origin cannot necessarily determine whether the latter resulted from the Trigger-resource-control request. However, the target may include Triggered-request-flag in the Resource-control-request to so indicate.)

If the origin issues a Trigger-resource-control request saying to cancel the operation, and if the target honors the request, the origin can expect to receive a terminating response indicating that the operation was terminated at origin request.

Although an origin may issue a Trigger-resource-control request as part of an active operation, the target might receive the request after the operation terminates. In that case, the target will ignore the Trigger-resource-control request. Furthermore, the target might receive a Trigger-resource-control request after issuing a Resource-control request for the same operation, while awaiting a Resource-control response. In that case, again, the target should ignore the Trigger-resource-control request. (Note that in general, the target may ignore any Trigger-resource-control request.)

Parameter	Origin Request
Requested-action	x
Preferred-resource-report-format	x (if applicable)
Result-set-wanted	x (if applicable)
Other-information	x (optional)
Reference-id	x (if applicable)

3.2.6.2.1 Requested-action The origin indicates one of the following:

resource-report	-	Issue a Resource-control request and set Response-required to 'off'.
resource-control	-	Issue a Resource-control request and set Response-required to 'on'.
cancel	-	Terminate the operation.

3.2.6.2.2 Preferred-Resource-report-format Identifies a resource report format that the origin prefers.

3.2.6.2.3 Result-set-wanted This flag is meaningful only for a Search operation, and when the value of the parameter Requested-action is 'cancel'. If the value of the flag is 'on', the origin requests that the target maintain the (possibly partial) result set for subsequent Present operations. See 3.2.6.1.7.

3.2.6.2.4 Other-Information This parameter may be used for additional information, not specified by the standard. This parameter may be used only when version 3 is in force.

3.2.6.2.5 Reference-id See 3.4.

3.2.6.3 Resource-report Service

The Resource-report service allows an origin to request a Resource-report, pertaining to a specified operation.

> Note: The Resource-report service differs from the Trigger-resource-control service in several respects. Trigger-resource-control is a non-confirmed service; it is part of, but does not initiate, an operation; it requests a report pertaining to that active operation. Resource-report is a confirmed service; the request and response initiate and terminate an operation respectively; the request identifies a particular completed operation and solicits a report pertaining to that operation (or it may solicit a report pertaining to the entire Z-association). But even though the Resource-report request requires a response, the target is not obliged to include a resource report in that response.

Parameter	**Origin Request**	**Target Response**
Preferred-resource-report-format	x (opt.)	
Op-id	x (opt.)	
Resource-report-status		x
Resource-report		x (opt.)
Other-information	x (opt.)	x (opt.)
Reference-id	x (opt.)	x (if appl.)

3.2.6.3.1 Preferred-resource-report-format Identifies a resource report format that the origin prefers.

3.2.6.3.2 Op-id This parameter may be supplied by the origin to identify a completed operation for which the origin wants a resource report. This parameter may be used only when version 3 is in force.

- If Op-id is present, it consists of a Reference-id, and refers to the most recently completed operation which used that Reference-id.

  Note:

  (1) When an operation terminates, if the origin anticipates that it will subsequently issue a Resource-report request pertaining to that operation, it is the origin's responsibility to ensure that the Reference-id is not re-used before doing so.

  (2) The origin may (but need not) use the same reference-id for the Resource-report operation as that specified in Op-id, and if so, Op-id will nevertheless pertain to a completed operation only. However, it is recommended that the origin not specify a value of Op-id equal to any reference-id being used by any active operation other than this Resource-report operation. If the origin does so, the target may (but need not) consider the request in error (see failure-6 of Resource-report-status).

  (3) If the origin wants resource information about an active operation, it should use the Trigger-resource-control service, as part of that operation. If the operation terminates before the target receives the Trigger-resource-control request, the origin will receive a terminating response and may then subsequently issue a Resource-report request pertaining to that (completed) operation.

- If Op-id is not present, the origin requests a resource report pertaining to all active and completed operations for the Z-association.

3.2.6.3.3 Resource-report-status The target supplies one of following status values:

success	-	A resource report is included (and in the preferred format, if the parameter Preferred-resource-report-format was included in the request).
partial	-	A resource report is included, but not in the preferred format (applies only if the parameter Preferred-resource-report-format was included in the request).
failure-1	-	Target unable to supply resource report.
failure-2	-	Access terminated by target due to resource constraints.
failure-3	-	Access-control failure.
failure-4	-	Unspecified failure.
failure-5	-	There is no known operation with specified id.
failure-6	-	There is an active operation with specified id.

Note: Failure-5 and failure-6 apply only when version 3 is in force.

3.2.6.3.4 Resource-report See 3.2.6.1.1.

3.2.6.3.5 Other-information This parameter may be used by either the origin or target for additional information, not specified by the standard. This parameter may be used only when version 3 is in force.

3.2.6.3.7 Reference-id See 3.4.

3.2.7 Sort Facility

The Sort facility enables the origin to request an ordering of a result set by the target. The origin specifies a sequence of sort elements. The result set is to be ordered according to the specified sequence, and subsequent positional requests against the result set will be construed by the target to apply to the result set as so ordered.

3.2.7.1 Sort Service

| The Sort service allows an origin to request that the target sort a result set (or merge multiple
| result sets and then sort) according to a specified sort sequence.

Parameter	Origin Request	Target Response
Input-result-sets	x	
Sorted-result-set	x	
Sort-sequence	x	
Sort-status		x
Result-set-status		x (if appl.)
Diagnostics		x (if appl.)
Other-information	x (opt.)	x (opt.)
Reference-id	x (opt.)	x (if appl.)

3.2.7.1.1 Input-result-sets The name of a result set to be sorted, or the names of result sets to be merged and the result sorted.

3.2.7.1.2 Sorted-result-set The name of the sorted result set. It may be the name of an existing result set (including one of the names included in Input-result-set); if so, then if the sort is processed, the existing result set is deleted, and a new result set by that name created, whose content is the sorted results. If Sorted-result-set is not the name of an existing result set: if the sort is processed, a result set by the specified name is created by the target, whose content is the sorted results; the content of the Input-result-sets is left unchanged. In any case, if the sort is not processed, the final content of Sorted-result-set is indicated by the parameter "result-set-status".

3.2.7.1.3 Sort-sequence The elements that are to be used for sorting, together with the direction of the sort (ascending or descending), case sensitivity (if applicable), and target action if an element is missing from a record in the result set to be sorted. Each of the sort elements is a set of attributes, a sort-field-designator, or an element specification, that the target has designated for use as a sort key.

Note: The target designates this information either via the Explain facility, or through some mechanism outside of the standard.

3.2.7.1.4 Sort-status The parameter Sort-status, returned by the target, assumes one of the following values:

success	-	The sort was performed successfully.
partial-1	-	The sort was performed but the target encountered records with missing values in one or more sort elements.
failure	-	The sort was not performed. The target supplies a diagnostic in the parameter Diagnostics.

3.2.7.1.5 Result-set-status Occurs if and only if the value of Sort-status is "failure". It refers to the contents of Sorted-result-set, and its value is one of the following:

empty	-	The result set is empty.
interim	-	Partial results available, not necessarily valid.
unchanged	-	The content of the result set is unchanged.
none	-	Result set not created.

3.2.7.1.6 Diagnostics The target includes this parameter if the value of Sort-status is 'failure'. It includes one or more diagnostic records.

3.2.7.1.7 Other-information This parameter may be used by either the origin or target for additional information, not specified by the standard.

3.2.2.7.1.8 Reference-id See 3.4

3.2.8 Browse Facility

The Browse facility currently consists of a single service, the Scan service.

3.2.8.1 Scan Service

The Scan service is used to scan an ordered term-list (subject terms, names, titles, etc.). The ordering of the term-list is target defined. The origin specifies a term-list to scan and starting term (implicitly, by specifying an attribute/term combination and a database-id), the size of the scanning steps, and the desired number of entries and position of the starting term in the response.

Parameter	Origin Request	Target Response
Database-names	x	
Term-list-and-start-point	x	
Step-size	x (optional)	x (if applicable)
Number-of-entries	x	x
Position-in-response	x (optional)	x (optional)
Scan-status		x
Entries		x (optional)
Other-information	x (optional)	x (optional)
Reference-id	x (optional)	x (if appl.)

3.2.8.1.1 Database-names Identifies a set of databases to which the term-list (specified by Term-list-and-start-point) pertains.

3.2.8.1.2 Term-list-and-start-point An attribute list and term. The attribute list contains attributes indicating which term-list to scan. The term, as qualified by those attributes, indicates where scanning begins; this will be a presumed entry in the term-list. If there is no matching entry, the first entry with higher value is to be the starting point.

As an example, to scan a list of personal names: the attribute set would consist of a single attribute (from the bib-1 attribute set definition) whose type is 'use' and whose value is 'personal name'; the term would specify a personal name; the database-id would identify one or more databases to which the list of personal names pertains.

3.2.8.1.3 Step-size The origin may specify the desired number of entries in the term-list between two adjacent entries in the response. A value of zero means "do not skip any entries". If the target cannot support the requested step size, it sets Scan-status to 'failure' and includes a non-surrogate diagnostic, such as "only step size of zero supported" or "requested step size not supported". If the origin omits this parameter, the step size is selected by the target, and the target includes the selected step size in the response.

3.2.8.1.4 Number-of-entries The origin indicates the proposed number of entries to be returned. The target indicates the actual number of entries returned. If the actual number is less than the proposed number, the reason is indicated in Scan-status.

3.2.8.1.5 Position-in-response The origin may optionally indicate the preferred position, within the returned entries, of the specified starting point value. A value of 1 refers to the first of the returned entries. The target indicates the actual position of the chosen starting point within the returned entries. Example: If the values of the request parameters Number-of-entries and Position-in-response are 10 and 3 respectively, then the origin requests two terms immediately preceding the starting point value, followed by the starting point value, followed by the immediately-following seven terms.

Note: If response parameter Position-in-response is less than the value proposed in the request, the origin may conclude that there were fewer terms than expected in the low end of the term-list. However, if Position-in-response is the same value in the response as proposed in the request, but Number-of-entries in the response is less than the value proposed in the request, the origin may not conclude that there were fewer terms than expected at the high end of the term-list, unless Scan-status is Partial-5. The reason that fewer terms than expected are returned is indicated in the Scan-status.

3.2.8.1.6 Scan-status The target indicates the result of the operation. The defined values are:

- success - The response contains the number of entries (term-list-entries or surrogate diagnostics) requested.
- partial-1 - Not all of the expected entries can be returned because the operation was terminated by access-control.
- partial-2 - Not all of the expected entries will fit in the response message.
- partial-3 - Not all of the expected entries can be returned because the operation was terminated by resource-control at origin.
- partial-4 - Not all of the expected entries can be returned because the operation was terminated by resource-control at target.
- partial-5 - Not all of the expected entries can be returned because the term-list contains fewer entries (from either the low end, high end, or both ends of the term-list) than the number of terms requested.
- failure - None of the expected entries can be returned. A single non-surrogate diagnostic is returned.

3.2.8.1.7 Entries The parameter Entries returned by the target consists of one of the following:

- N entries, where each entry is a term-list-entry or surrogate diagnostic, where N = Number-of-entries in the request.
- a number of entries, which is less than N (reason specified by Scan-status), or
- a single diagnostic record indicating that the operation cannot be processed, and why it cannot be processed.

Each term-list-entry includes a term (occurring in one of the databases specified in the parameter Database-names), and optionally the following:

- A list of suggested attributes for use in subsequent Scan requests (useful for scanning multiple indices, e.g. author and title, at the same time).
- A suggested alternative term.
- Occurrence-information: this might include a count of records in which the term occurs. It may also list counts for specific attributes, possibly further broken down by database. Alternately, it might list databases in which the term occurs and for which attributes (with no counts).
- Other information: additional information concerning the entry.

3.2.8.1.8 Other-information This parameter may be used by either the origin or target for additional information, not specified by the standard.

3.2.8.1.9 Reference-id See 3.4.

3.2.9 Extended Services Facility

The Extended Services facility allows an origin to create, modify, or delete a task package at the target. The target maintains task packages in a special database, described in section 3.2.9.2. A task package pertains to an Extended Services (ES) task.

An extended service is a task type, related to information retrieval, but not defined as a Z39.50 service. Execution of a task by the target is outside the scope of Z39.50. The extended services defined by this standard are listed in section 3.2.9.1.2. Definitions of those services are included in appendix EXT.

3.2.9.1 Extended Services Service

The Extended Services (ES) service allows an origin to send an ES Request to the target requesting execution of a task. The request includes parameters which the target uses to construct the task package. The target checks the request for validity, for consistency with the user's access privileges, and possibly for other target-dependent limitations. The target sends an ES response indicating that the request was accepted or supplying an indication of the reason the request was rejected.

The ES service is a confirmed service, initiated by the origin. The ES operation consists of a request from the origin and a response from the target, possibly with intervening Access-control and/or Resource-control messages. However, although the request may result in the initiation of a task, the task is not considered part of the Z39.50 ES operation. The target response, which completes the ES operation, does not necessarily signal completion of the task. A task may have a lifetime that exceeds a single Z-association.

Execution of the ES Operation results in the creation of a task package, represented by a database record in the ES database. A task package contains parameters, some of which are common to all task packages regardless of package type, and others which are specific to the particular extended service. Among the common parameters (indicated in the table below, listed under "task package parameter" in the right column), some are supplied by the origin as parameters in the ES request, and used by the target to form the task package; some of those supplied by the origin may be overiden by the target. Others are supplied by the target. The specific parameters are derived from the parameter Task-specific-parameters of the ES request (see Appendix EXT).

Note: The response parameter Task-package below refers to the actual task package, and if it occurs (see 3.2.9.1.16), it includes some or all (depending on the parameter Elements) of the parameters listed under "task package parameter".

Parameter	**Origin request**	**Target Response**	**Task Package Parameter**
Function	x		
Package-type	x		
User-id	x (opt)		x
Package-name	x (opt)		x (opt)
Retention-time	x (opt)		x
Permissions	x (opt)		x
Target-reference			x (opt)
Creation-date-time			x
Description	x (opt)		x (opt)
Task-status			x
Task-specific-parameters	x		(see note)
Wait-action	x		
Elements	x (if appl)		
Operation-status		x	
Diagnostics		x (if appl)	
Task-package		x (if appl)	
Reference-id	x (opt)	x (if appl)	

Note: Task-specific-parameters are defined for each extended services. For each task-specific parameter, the definition states whether or not the parameter occurs in the task package.

3.2.9.1.1 Function The origin specifies Create, Delete, or Modify.

If the function is Create, the target is to create a task package, and assign to it the name specified by the parameter Package-name, if any.

If the function is Delete or Modify, the target is to delete or modify the task package specified by the parameter Package-name. A target that supports deletion or modification may nonetheless deny the request, for example because the task is already in progress.

If the function is Delete, then if the specified task has not been acted on, it will not be started. If the task is active, the target will either terminate the task or refuse the request.

If the function is Modify, the origin requests that parameter values in Request-parameter-package, as well as values for Package-name, Retention-time, and Permissions replace the values in the task package. If an optional parameter is omitted, the target does not modify that parameter within the task package, to return a parameter to its default value, an origin must explicitly provide the default value.

3.2.9.1.2 Package-type The extended service requested. The extended services defined by this standard (see Appendix EXT) are:

- Save a result set for later use
- Save a Query for later use
- Define a periodic search schedule
- Order an item
- Update a database

- Create an export specification
- Invoke a previously created export specification
- Send a usage accounting report

3.2.9.1.3 User-id The Id of the user to be associated with the task package. If not supplied, defaults to the Id of the current user. A target may allow an origin to supply a user Id different from its own (although this might not be permitted by some targets).

3.2.9.1.4 Package-name Optional origin-supplied name for the task package to be created. If it is included, the triple (Package Type, User Id, Package Name) must be unique (otherwise the request is in error) and identifies the task package for subsequent reference. Package-name should be included if the origin intends to reference the task package.

3.2.9.1.5 Retention-time A number of days, after which the target may delete the retained task package. If the value is zero, the task package will not be retained after the task is completed. If not specified, it will default to the target-specified retention (which may be infinite).

3.2.9.1.6 Permissions Indicates who may access the task package. If not supplied only the creating user may do so. See 3.2.9.3.

3.2.9.1.7 Target-reference A target-unique identifier for the task package.

3.2.9.1.8 Creation-date-time The date and time that the task package was created.

3.2.9.1.9 Description The origin may specify a description, for example, of the result set, for a Persistent Result Set task; or of a query, for a Persistent Query task.

3.2.9.1.10 Task-status The status of the task. Values are 'pending', 'active', 'complete', and 'aborted'.

3.2.9.1.11 Task-specific-parameters Contains information specific to the particular extended service to which the request pertains.

3.2.9.1.12 Wait-action Indicates whether the target should include the result of the ES task in the ES response. This parameter has three possible values: 'wait', 'wait-if-possible', and 'do not wait'.

If the value is 'wait', the target must perform the task before issuing the ES response (unless the operation aborts; see section 3.2.9.4). If the target is not willing to perform the task before issuing the response it must refuse the request by responding with a status of 'failure' and an appropriate diagnostic. If the target accepts the request, it includes the parameter Task-package in the response.

If the value is 'wait-if-possible' the origin requests that the target perform the task before issuing the ES response, if possible, but if not possible, proceed as though the value were 'do not wait'.

If the origin does not wish to request that the target attempt to perform the task before issuing the ES response, it should set this parameter to 'do not wait'.

This immediate response mechanism may avoid the need for follow-up Search and Present operations, or in general, for making the task package available through the extended services database (see section 3.2.9.2).

3.2.9.1.13 Elements May optionally be included if Wait-action is 'wait' or 'wait-if-possible'. It is an element set name for the task package to be returned in the response parameter Task-package. See section 3.2.9.2.

3.2.9.1.14 Operation-status The status of the ES operation. One of the following:

done -	The request was accepted, the task is complete and results are included in Task-package.
queued -	The request was accepted and the task is queued for processing.
failure -	The request was refused. A diagnostic is supplied.

3.2.9.1.15 Diagnostic Additional diagnostic information supplied if Status is 'failure'. See Appendix ERR.

3.2.9.1.16 Task-package Included if status is 'done'. Contains the actual task package, or information from the task package as determined by the parameter Elements.

3.2.9.1.17 Reference-id See section 3.4.

3.2.9.2 The Extended Services Database

Targets that support the Extended Services facility provide access to a database with the name IR-Extend-1 (referred to as the "extended services database" or "ES database"). Records in the extended services database are task packages constructed from the Request-parameter-package parameter in ES requests (the target may begin execution of the task at any time after it accepts the request, which may be before the task package has been stored in the database). The target may (but need not) retain a task package until the requested task has completed; it may retain the task package until the origin requests that it be deleted. A target may unilaterally delete a task package from the ES Database at any time.

When the target receives an ES request it may immediately create a task package, with status 'pending', before completely validating the request. The origin may thus search the database anytime after submitting a request (during the same or a subsequent Z-association), for a resulting task package. In particular, if an ES operation is aborted (see 3.2.9.4) the origin may be able to determine that the request for that operation was received.

An ES database may be listed in the target Explain database, with a list of extended services the target supports, allowable export destinations, options that an origin may supply for an export task, etc.

An extended services database will appear to the origin as any other database supported by the target (records may be searched and retrieved by the Z39.50 Search and Retrieval facilities; search processing is defined locally by the target; the target may impose access control or exclude records to which the origin is not authorized access). However, certain search terms are predefined in order to allow a semantic level of interoperability. The attribute set(s) used to search the database are defined and registered in Appendix ATR. The syntaxes of the task packages (for retrieval, and also used on the ES request and response) are defined and registered in Appendix EXT.

The ES database may provide the following special element sets (in addition to "F"):

- Identification: includes the creating user's identification, the origin-supplied name of the task package and possible permissions for other users to access the request. Other identifying information such as time of creation may be included.
- UniqueName: the creating user's identification and the name of the task package.
- Permissions: the contents of the UniqueName element set, and in addition, the granted permissions for the task package. A target might present the full permissions list only to a task package's creator, presenting to other users only the permissions applicable to them.
- Status: a short summary of the current status of the request, perhaps including cost and other resource usage.
- Brief: Identification element set plus the most important elements of the Status element set.

3.2.9.3 Owners and Permissions

The creating user of a task package may apply any extended service function to the package, as well as retrieve the full package (via the Retrieval facility) and invoke the package via other extended services. (Invocation occurs, for example, when a Periodic Query task references a saved Query.)

Using the Modify function of the ES request, an origin can change the access permissions of a task package by supplying a new permissions list, which is a sequence of items, each containing a user Id and a sequence of allowed operations from the following set:

- Delete
- Modify-Contents
- Modify-Permissions
- Present
- Invoke

Targets may provide group names for use in permission lists, but a group name would be syntactically the same as a user Id. The composition of groups could be reported through a dedicated database.

3.2.9.4. Aborted Operations

An origin may receive a response to an ES request only during the Z-association in which it issues the request (as for any other Z39.50 operation). If an ES operation is aborted (explicitly, or because the Z-association is closed or the A-association terminated), the origin will not receive a terminating response. This has no effect on the disposition or processing of the task, regardless of the value of wait-action. However, if an ES request specifies "wait" (or "wait-if-possible"), and the operation subsequently aborts, the "wait-action" no longer is meaningful.

If an ES operation is aborted, the origin may search the ES database (possibly in a subsequent Z-association) to obtain (if the package still exists in the ES database) information that would otherwise have been returned in the response.

3.2.10 Explain Facility

The Explain facility allows an origin to obtain details of the nature of the implementation of the target system, including the databases available for searching, attribute sets and diagnostic sets used by the target, and record syntaxes and element sets supported for retrieval. Targets that support the Explain facility:

- provide access (via the Z39.50 Search and Present services) to a database with the name IR-Explain-1 (referred to as the "Explain database");
- support the explain attributes, described in Appendix ATR, and
- support the explain syntax, described in Appendix REC.

A record within the Explain database, or within a result set which represents records in the Explain database, is referred to as an "Explain record".

3.2.10.1 Searching the Explain Database

The Explain database appears to the origin as any other database supported by the target. However, certain search terms, corresponding to information categories, are predefined in order to allow a semantic level of interoperability. Predefined information categories are listed in section 3.2.10.1.1. Records corresponding to a particular category are retrieved by a search operand where the term is the name of that category; for example, the record corresponding to general-target-info is retrieved by a search on the term "general-target-info". Categories are searchable using the Explain attribute Explain-category (see Appendix ATR). Terms are searched case insensitive.

3.2.10.1.1 Predefined Information Categories

A list and brief description of the Explain information categories (and thus search terms) is given below. In Section 3.2.10.3 each category is described in detail.

category	Brief Description
General-Target-Info	General description of the target; for example, hours of operation, contact name.
List-of-Databases	Enumeration of the databases available through this target and summary information about each database.
Target-IR-Parameters	Search constraints imposed by the target; for example, number of result sets allowed, whether multiple database searching is supported, maximum result set size allowed.
General-Database-Info	General information about each database; for example, number of records, last update time, costing information, provider.
Database-IR-Parameters	Z39.50 related aspects about each database; for example, information about supported attribute set Ids, record composition names, and record syntaxes.
Database-Attributes-Info	Information about the attributes sets supported by a particular database; for example, information about use attributes, their values and "meaning".
Attribute-details	A different view of an attribute, and its relationship to the other attributes, than that provided by the "Database-Attributes-Info" category.
Record-Syntax-Info	Information about the record syntaxes for a particular database; for example, OIDs defined elsewhere that are supported, and ASN.1 descriptions that can be used to directly describe the record.
Schema-info	Information about the Schemas defined for a particular database.
Record-Element-Info	Information about the elements making up the records of a particular database; for example, the elements, their relationship to different Use attributes (i.e. how they are indexed), and how they are contained in different record compositions.
Element-Details	A different view of an element than that provided by the "Record-element-Info" category; for example, its content and its relationship to the other data in a record.
Display-Language	Advice from the target about how the content of a record might be presented to a user.
Term-lists	Information about term lists used with Scan.
Category-list	Explain categories that the target supports.

3.2.10.1.2 Searching for Information in a Particular Language

Elements intended to be presented to the user by the origin are said to consist of "human readable text". Each record includes a language element indicating the language of the human readable text within the record. The explain database might contain several records with identical information, in different languages. To retrieve records in a certain language, the 'language' attribute may be used (in conjunction with the three-character language code as the term; see the USMARC Code List for Languages). For example, to retrieve a list of databases in english, the query might be of the form:

(Category = "list-of-databases") AND (Language = "eng")

3.2.10.2 Retrieval of Explain Records

A Present request for records from a result set representing records from the Explain database should specify a preferred-record-syntax of IR-Exp-syntax, referred to as the "Explain syntax", which is defined and registered in Appendix REC. Each explain information category has its own record layout, and all are included in the Explain syntax.

3.2.10.2.1 Retrieval and Human Readable Text

As noted in section 3.2.10.1.2 the explain database might contain several records with identical information, in different languages. The Explain database might also provide alternative representations of identical (human readable) information according to variants other than language; for example, a text element might be retrievable in ASCII, SGML, or Postscript. An explain record with such elements should be viewed logically to contain multiple elements for each form that the information is available in, for each variant. A Present request may indicate a particular form by specifying an appropriate Element-set-name.

3.2.10.3 Detailed Descriptions of the Information Categories

This sections includes complete descriptions of each information category. In addition to the information enumerated, each record:

- contains information about the record itself, e.g. date of creation and expiration date of the record; and
- includes an element indicating the language of the "human readable text" elements of the record.

These are logical descriptions, which do not reflect the possibility that there might be language variants of a record or syntax variants of a element.

3.2.10.3.1 General-Target-Info General information about the target. There is one such Explain record, containing the following information:

- A name for the target (only one), in human readable text.
- A set of nicknames or alternate names by which the target is known.
- A description of the target, in human readable text.
- Restrictions pertaining to this target, in human readable text.
- Contact information for the organization supporting this target.
- A payment address (e.g. business office) for the organization supporting this target.
- Recent news of interest to people using this target, in human readable text.
- Hours of operation that the target is available.
- An icon used to represent this target (in machine presentable form).

3.2.10.3.2 List-of-Databases Brief summary data about all the databases served by the target. There is one Explain record, containing the following information for each database supported by the target:

- Full database name (only one), in human readable text.
- A list of short (or alternate) names for the database, in human readable text.

- A description, in human readable text.
- An icon used to represent this database (in machine presentable form).
- A boolean flag indicating whether there is charge for this database.
- A boolean flag indicating whether this database is currently available for access.

In addition to the above list (which occurs for each database), the record also includes a single list of:

- Sets of databases which may be searched in combination.

3.2.10.3.3 Target-IR-Parameters Features supported and limitations imposed by the target. There is one such Explain record, containing:

- A boolean flag indicating whether named results sets are supported.
- A boolean flag indicating whether multiple databases can be searched in one search request.
- The maximum number of concurrent result sets supported.
- The maximum size (in records) of a result set.
- The maximum number of terms allowed in one search request.
- A timeout interval (in seconds), after which the target will trigger an event if no activity has occurred.
- Which query-types (e.g. Type-2) are supported.
- Which services are supported.
- Content-structure description; global definitions of how the target builds documents, with respect to the specific structure, e.g. SGML, ODIF.
- The Page formats supported by this target.

  Note: Content-structure description and Page format also listed per database in section 3.2.10.3.5.
- A boolean flag indicating whether the proximity operator is supported, and if so, what units.
- Diagnostic sets supported (general, non-database-specific).
- Resource reports supported (general, non-database-specific).
- Access challenges supported (general, non-database-specific).

3.2.10.3.4 General-Database-Info Describes in detail the database and database related restrictions and parameters. There is one such Explain record, for each database supported by the target, containing all the information in the "List-Of-Databases" record for the database, and then more detailed information. For completeness, the summary information is repeated.

- Full database name (only one), in human readable text.
- A list of short (or alternate) names for the database, in human readable text.
- A description of the database, in human readable text.
- An icon used to represent this database (in machine presentable form).
- A boolean flag indicating whether there is charge for this database.
- A boolean flag indicating whether this database is currently available for access.
- If it is a group (or cluster) of databases: a list of databases making up this database group.
- Any disclaimers concerning this database, in human readable text.
- Any news about this database, in human readable text.
- A record count for the database (and whether the count is accurate or an estimate).
- A description of the default order in which records are presented, in human readable text.
- An estimate of the average record size (in bytes).
- A maximum record size (in bytes).
- Hours of operation that this database is available.

- Best time to access this database, in human readable text.
- Time of last update of this database.
- Update cycle/interval for this database.
- Coverage dates of this database, in human readable text.
- A boolean flag indicating whether this database contains proprietary information.
- A description of copyright issues relating to this database, in human readable text.
- A notice concerning copyright which the target expects the origin to display to the user if possible, in human readable text.
- A boolean flag indicating that the origin should expect an access control challenge if this database is used.
- Text describing to users access control on this database, in human readable text.
- Costing information, in both machine readable format, and in human readable text, for connect, present, and search.
- Description and contact information for the database producer, database supplier, and for how to submit material for inclusion in this database, in human readable text.
- Databases that the target allows/encourages to be searched in combination with this database.

    Note: however, the most complete information about valid combinations is in the "List-of-databases" record.
- Sub-databases that make up this conceptual single database. (Oriented towards what the target believes users will be interested in. For example, if the target allows searching of "newspapers", which is really a combination search of databases of different newspapers, this would be indicated here.)

3.2.10.3.5 Database-IR-Parameters Specifics about Z39.50 characteristics supported by the database. There is one such Explain record for each database supported by the target (however a single explain record may represent several databases if the information is identical for each database) containing:

- Full name of the database to which this record pertains (or may include more than one name if the information is identical for each database named) in human readable text.
- Attribute set Ids supported by this database.
- Schemas defined for this database.
- Record syntaxes supported by this database.
- Element set names supported, with names and descriptions given in human readable text.
- Relationships among element set names, indication which are subsets/supersets of others.
- Diagnostic sets supported by this database.
- Resource reports supported by this database.
- Access challenges supported by this database.
- Content-structure description.
- The Page formats supported by this database.
- Proximity parameters supported.
- Term-lists associated with this database. This is a list of names, pointing to records in 3.2.10.3.12 (below). This database should occur in the "combination of databases" for each name listed here.

3.2.10.3.6 Database-Attributes-Info Descriptive information about how a given attribute set is implemented for a given database. For a given database/attribute-set pair, there may be several such records, each pertaining to a supported attribute combination, containing:

- Full name of the database to which this record pertains (or may include more than one name if the information is identical for each database named) in human readable text.

- The attribute set Id for this attribute set.
- A list of all Use attributes with the following "Attribute Details" per attribute:
  (Note: this information is repeated in the records described in section 3.2.10.3.7.)
  - o Attribute name, in human readable text.
  - o Attribute number.
  - o Attribute description, explaining what aspect of records this attribute accesses, in human readable text.
  - o Sub-attributes: other Use attributes that allow access to the same aspect of the record, but in greater detail.
  - o Super-attributes: other Type-or-term attributes that allow access to the same aspect of the record at a coarser level.
  - o Non-Use Attributes. This would give values of other attributes that acceptably combine with this Use attribute.
- Information about other (Non-Use) attributes in this attribute set.

3.2.10.3.7 Attribute-Details Contains the same information, per attribute, as provided in section 3.2.10.3.6, but allows retrieval on a per-attribute basis. There is one Explain record for each database/ abstract record syntax pairing, for each element for each Use attribute supported by a target, containing:

- Full names of the databases to which this attribute applies, in human readable text.
- The attribute set Id to which this attribute belongs.
- Attribute Details as listed in section 3.2.10.3.6.

3.2.10.3.8 Record-Syntax-Info Descriptive information about a given record syntax. (Note: this is not specific to a database.) There is one Explain record for each abstract record syntax, containing:

- The OID of the abstract record syntax to which this record pertains.
- Other names by which this syntax is known, in human readable text.
- Transfer syntaxes that can carry this abstract syntax.
- A description of this abstract record syntax, in human readable text.
- Databases in which this syntax is supported.
- An ASN.1 module describing the syntax.
- Element set names supported by this syntax.

3.2.10.3.9 Schema-Info Descriptive information about a given database schema. (Note: this is not specific to a database.) There is one Explain record for each schema, containing:

- The OID of the schema definition.
- Names for this schema.
- A description of this schema, in human readable text.
- Databases in which this schema is supported.
- Elements defined by this schema.
- Other schemas whose definitions (or parts of) are imported by this schema.
- Record syntaxes that may be used with this schema.

3.2.10.3.10 Record-Element-Info Descriptive information about the elements making up a record. Note that an element is relative to a database schema. There is one such Explain record for each database/schema pairing, containing:

- Full name of the database to which this record pertains (or may include more than one name if the information is identical for each database named) in human readable text.
- Schema in which these elements are defined.

- A description of this element breakdown, in human readable text.
- Element set names supported, with names and descriptions in human readable text; and the elements they contain.
- Attribut set(s), if any, that correspond to this syntax (i.e. Use values correspond to elements).
- A list of all the elements, with the following "Element Details" per element:
  (Note: this information is repeated in the records described in section 3.2.10.3.11.)
  - o Boolean flag indicating whether the target requires the origin to display this element.
  - o element name, in human readable text.
  - o Alternate names for this element in human readable text.
  - o Generic names for this element in human readable text (e.g. a "geographicSubject" element might also be under the generic name "subject").
  - o Element number.
  - o Size information, in bytes: max, min, average, and fixed (if applicable).
  - o Whether more than one occurrence of the element is allowable.
  - o Boolean flag indicating whether the element is required in all records.
  - o Information about whether the element is indexed and with which attribute values.
  - o Element sets containing this element, in human readable text.
  - o A description of the element, in human readable text.
  - o A datatype specifier for the element.
  - o Charging/billing issues related to this element, in human readable text.
  - o Copyright and/or proprietary restrictions pertaining to use and access to this element, in human readable text.

3.2.10.3.11 Element-Details Contains the same information, per element, as provided in section 3.2.10.3.10, but allows retrieval on a per-element basis. There is one such Explain record for each database/schema pairing, for each element defined by the schema, containing:

- Full name of the database to which this record pertains (or may include more than one name if the information is identical for each database named) in human readable text.
- Schema in which this element is defined.
- Element Details (as listed in section 3.2.10.3.10).

3.2.10.3.12 Display-Language A description of how the target believes the data should be presented to the user. There is one Explain record for each database supported by the target (however a single explain record may represent several databases if the information is identical for each database) containing:

- Full name of the database to which this record pertains (or may include more than one name if the information is identical for each database named) in human readable text.
- A list of displays, with the following for each display:
  - o The name of the display, in human readable text.
  - o The record composition name needed to get a record with sufficient information/elements to produce this display.
  - o A description of the display, in human readable text.
  - o The display itself, in a specified display description language.
- A list of tables used in the displays. These tables basically allow the displays to modify and access sub-pieces of the elements. Each table consists of:
  - o The name of the table, in human readable text.
  - o The table itself, in a specified table description language.

3.2.10.3.13 Term-lists Description of the term-lists supported by the target. There is one record for each term-list.

- A name for the term-list.
- A description (optional).
- Combination of databases that this term-list pertains to.
- A list of alternative database combinations (optional).
- Attribute-list (including attribute-set-id) used by Scan to access the term-list.
- A list of alternative attribute-lists (optional).
- Maximum step-size supported.
- Collating sequence (e.g. ASCII, EBCDIC; this element is human-readable; optional).
- Order (ascending or descending; optional).
- Estimated number of terms (optional).
- A list of sample terms (not guaranteed to be valid; optimally would represent a uniformly distributed sampling of the list; optional).

| **3.2.10.3.14 Category-list** A list of the Explain categories supported by the target. There is one such record
| for the Explain database. It consists of the information below, for each supported category.

- The search term used in conjunction with Use attribute of InformationCategory to search for records of this category.

    Note: the following items need occur only if the target is supporting a category not defined in this standard.

- The original search term. (This is for information categories where the target is supporting a revision of the original definition of a category.)
- A description.
- An ASN.1 definition of the records in this category.

3.2.11 Termination Facility

The Termination Facility allows either an origin or target to abruptly terminate all active operations and initiate termination of the Z-association.

3.2.11.1 Close Service

The Close service may be used only when version 3 is in force. If so, following initialization, at any time until a Close request is either issued or received, either the origin or target:

- may issue a Close request, consider all active operations to be abruptly terminated, await a Close response (discarding any intervening messages), and consider the Z-association closed; and
- should be prepared to receive a Close request, consider all active operations to be abruptly terminated, issue a Close response, and consider the Z-association closed.

Parameter	Request	Response
Close-reason	x	x
Diagnostic-information	x (optional)	x (optional)
Resource-report	x (if applicable)	x (if applicable)
Other-information	x (optional)	x (optional)

3.2.11.1.1 Close-reason The reason why the origin or target is closing the Z-association. Reasons are:

finished
shutdown
system problem
cost limits
resources
security violation
protocol error
lack of activity
unspecified
response to Close request

Note: Both the Close request and Close response map to the same protocol message (Close PDU). If both systems issue a Close request at the same time, each will receive the peer message as a Close response (even though the message was not sent as such). This potential ambiguity will not effect the correct operation of the protocol. However, the last of the above listed statuses, "response to Close request" may (but need not) be used to signal that the message was indeed issued as a response.

3.2.11.1.2 Diagnostic-information An optional text message, providing additional diagnostic information.

3.2.11.1.3 Resource-report The target (only) may optionally include a resource report. See 3.2.6.1.1.

3.2.11.1.4 Other-information Additional information, not specified by the standard.

3.3 Message/Record Size and Segmentation

A "segment" is a message which is sent (or is in preparation for transmission) by the target as part of an aggregate Present response, i.e. a Segment request or Present response.

Throughout 3.3, "record":

- unless otherwise qualified, means "response record", i.e. retrieval record or surrogate diagnostic;
- is considered to be a string of bytes; and
- except within 3.3.3, is a surrogate diagnostic record if the record size exceeds preferred-message-size.

"Record N" means "the response record corresponding to the database record identified by result set entry N".

Except within 3.3, a set of records is said to "fit in a segment" if the sum of their sizes (in bytes), not including any protocol control information, does not exceed Preferred-message-size. For the Present operation, the target might be unable to fit the requested records in a single segment, because of record or message size limitations. In that case, the target may perform segmentation of the Present response (if segmentation is in effect) by sending multiple segments (Segment requests followed by Present response).

Two levels of segmentation are subject to negotiation, level 1 and level 2. If neither level is in effect, the target response to a Present request consists of a simple Present response (a single segment) which contains an integral number of records. If level 1 segmentation is in effect, the target response to a Present request may consist of multiple segments (Segment requests followed by a Present response), and each segment must contain an integral number of records, i.e. records may not span segments. If level

2 segmentation is in effect, the target response to a Present request may consist of multiple segments, and records may span segments.

3.3.1 Procedures Which Apply When No Segmentation is in Effect

The procedures in this section (3.3.1) apply when no segmentation is in effect. (They apply not only to a Present operation when no segmentation is in effect, but they also apply in general to a Search operation, whether or not segmentation is in effect; a Search response is not subject to segmentation.)

The target responds to a Present request with a simple Present response (or to a Search request with a Search response), which contains an integral number of records. If the target is not able to return all of the records requested, because of message size limitations, the target should fit as many records as possible.

Assume that the target is attempting to return records M through N. If records M through N fit in the response, then the target returns those records. Otherwise, the target returns records M through P, where P is chosen such that records M through P fit in the response, but records M through P+1 do not.

<u>Illustration</u>

Assume that the target is attempting to return records 1 through 10; records 1 through 6 fit in the response, but retrieval records 1 through 7 will not fit:

The size of retrieval record 7, itself:

- (a) does not exceed Preferred-message-size, or
- (b) exceeds Preferred-message-size, but does not exceed Maximum-record-size, or
- (c) exceeds Maximum-record-size.

In case (a), the target returns records 1 through 6. In case (b), except as noted below (see "Exception"), the target substitutes a diagnostic record for retrieval record 7, indicating that the record exceeds Preferred-message-size. In case (c) the target substitutes a diagnostic record for retrieval record 7, indicating that the record exceeds Maximum-record-size. (If Maximum-record-size equals Preferred-message-size then there is no distinction between the meaning of the two diagnostics.)

In case (b) or (c):

- If the diagnostic record will not fit along with records 1 through 6, the target returns records 1 through 6. (Preferred-message-size must always be large enough to contain any diagnostic record; thus a subsequent present request beginning with record 7 will retrieve the diagnostic.)
- Otherwise, the target inserts the diagnostic record and proceeds to attempt to fit records 8 through 10.

<u>Exception</u>

If a Present request specifies a single record (i.e. Number-of-records-requested equals 1) then if the size of that retrieval record exceeds Preferred-message-size, but does not exceed Maximum-record-size,

the target will return that single retrieval record. Note that this exception applies only to a Present operation and not to a Search operation.

Thus in case (b), the origin may subsequently retrieve retrieval record 7, by issuing a Present request in which that record is the only record requested.

Note that the purpose of this distinction between Preferred-message-size and Maximum-record-size is to allow the transfer of normal length records to proceed in a routine fashion, with convenient buffer sizes, but to also provide for the transfer of an occasional exceptionally large retrieval record without requiring the origin to continually allocate and hold local buffer space for worst-case records. Note also that this intended purpose is defeated if the origin routinely requests a single record.

3.3.2 Level 1 Segmentation

When level 1 segmentation is in effect, the target may segment the aggregate Present response into multiple segments (zero or more Segment requests followed by a Present response), each consisting of integral records (i.e. records may not span segments). The procedures described in this section (3.3.2) apply if level 1 segmentation is in effect.

Beginning with the first record requested (see note) and continuing with adjacent higher number records, the target forms segments to contain the requested records, but not more segments than specified by the parameter Max-segment-count, if supplied on the Present request. Each segment is sent as a Segment request, except the last, which is sent as a Present response.

If the value of Max-segment-count is 1, then the procedures of 3.3.1 apply. Also, the same exception as cited in 3.3.1 applies if a Present request has requested a single record.

Assume that the origin requests result set records M through N.

Case A. M<N (i.e. more than one record requested).

1. Set P=M.
2. If records P through N fit in a segment:
   - o Fit records P through N in the segment.
   - o Go to step 3.

   Otherwise,
   - o Fit records P through Q, where Q (which is less than N) is such that records P through Q fit in a segment, but records P through Q+1 do not.
   - o If Max-segment-count is reached, go to step 3.
   - o Send a the segment as a Segment request.
   - o Set P=Q+1.
   - o Repeat step 2.
3. Send the segment as a Present response.

Case B. M=N (i.e. a single record requested).

The target sends a simple Present response (a single segment). The size of the segment may exceed Preferred-message-size. The segment contains the single requested retrieval record, or a surrogate diagnostic record if the size of the record exceeds Maximum-record-size.

<u>Illustration</u>

Assume the origin has requested records 1 through 10.

(1) If all ten records fit in a segment, the aggregate Present response consists of a Present response including the requested records.

(2) Suppose records 1 through 4 fit in a segment, but records 1 through 5 do not; records 5 through 9 fit in a segment but records 5 through 10 do not. (Assume the Present request has specified a value of 3 or greater for the parameter Max-segment-count.) Then the aggregate Present response consists of:
   - o a segment request including records 1 through 4,
   - o a segment request including records 5 through 9,
   - o a Present response including record 10.

   Note that the target is expected to pack as many records into a segment as will fit; thus for example, the first Segment would not consist of records 1 through 3, because records 1 through 4 will fit.

(3) Assume the conditions in (2) are true, except that the Present request has specified a value of 2 for the parameter Max-segment-count. Then the aggregate Present response consists of:

- o a segment request including records 1 through 4,
- o a Present response including records 5 through 9.

Present-status would be Partial-2.

3.3.3 Level 2 Segmentation

When level 2 segmentation is in effect, the target may segment the aggregate Present response into multiple segments (as is the case for level 1 segmentation) and in addition, records may span segments. The procedures described in this section (3.3.3) apply if level 2 segmentation is in effect.

If a retrieval record will not fit in a segment (along with records already packed into the segment) it may be segmented into multiple contiguous fragments (see 3.3.3.1) to be packed into consecutive segments according to the procedures detailed in 3.3.3.2 and 3.3.3.3.

3.3.3.1 Fragments

A fragment is defined to be a proper substring of a record (as noted above, within section 3.3.3 a record is treated as a string of bytes). A particular instance of segmentation of a record results in a sequence of two or more fragments whose concatenation (not including protocol control information) is identical to the record. However, there may be many different instances of segmentation of a particular record, and the origin cannot necessarily predict how a record will be segmented by the target in a particular instance.

For the purpose of procedure description (3.3.3.3) a starting fragment is defined to be a fragment which starts at the beginning of a record. An intermediate fragment is a fragment which neither starts at the beginning nor ends at the end of a record. A final fragment is a fragment which ends at the end of a record. An integral record (not segmented) is not a fragment.

The sum of the sizes in bytes of the records and record fragments in a segment, not including protocol control information, must not exceed Maximum-segment-size (see 3.3.3.2).

3.3.3.2 Segment Size, Record Size, and Segment Count

If level 2 segmentation is in effect, the Present request may optionally include these three parameters:

Max-segment-size	The largest allowable segment. If included, overrides Preferred-message-size (for this Present operation only). If not included, Max-segment-size assumes the value Preferred-message-size.
Max-record-size	The largest allowable retrieval record within the aggregate Present response. If included, it must equal or exceed Max-segment-size. (If level 2 segmentation is in effect, the parameter Maximum-record-size that was negotiated during initialization does not apply, whether or not Max-record-size is included.)
Max-segment-count	The maximum number of segments the target may include in the aggregate Present response. If its value is 1, no segmentation is applied for the operation, the procedures of 3.3.1 apply, and Max-record-size should not be included.

If the latter two parameters are both included, Max-record-size must not exceed the product Max-segment-size times Max-segment-count.

If Max-record-size but not Max-segment-count is included, this standard does not impose an upper bound on the value that Max-segment-count may assume.

If Max-segment-count is included (and its value is greater than 1), but Max-record-size is not, the product Max-segment-size times Max-segment-count is the maximum record size for the operation.

If the latter two parameters are both omitted the origin should be prepared to receive arbitrarily large records and an arbitrary number of segments.

3.3.3.3 Segmentation Procedures

The following procedures apply for level 2 segmentation. The target fits as many integral records as possible into the first segment. If all will fit, the segment is sent as a simple Present response. Otherwise, in the space remaining within that segment the target fits a starting fragment of the following record (if possible), and the segment is sent as a Segment request. The target then fits the remainder of that record into the next segment (if possible; and if not possible, sends Segment requests as necessary with intermediate fragments, and fits the final fragment, if any, into the beginning of the next segment) and fills as many integral records as possible within the space remaining within that segment. If the last of the requested records is placed in the segment (or Max-segment-count is reached) the segment is sent as a Present response. Otherwise the target continues to fill segments in this manner until the last of the requested records is placed in a segment or Max-segment-count is reached, and sends each segment as a Segment request except the last, which it sends as a Present response.

These procedures are re-iterated more formally as follows:

Assume that the origin requests records M through N.

> Note:"Record" means "surrogate diagnostic record" if the size of the record exceeds Max-record-size, or if the target is unable to segment the record such that each fragment fits within a segment.

1. Set R=M (begin preparation of first segment).
2. If record R fits in the current segment:
   - Fit integral records R through P, where P is the largest number (not exceeding N) such that records R through P fit.
   - If P equals N, or if Max-segment-count is reached, go to step 8.
   - R=P+1
3. (Note: having reached this step, record R will not fit in the current segment).
   If the Present request has included Max-segment-count and the target is unable to determine whether record R will fit in the remainder of the aggregate response:
   - Insert a surrogate diagnostic record, which in effect suggests that the origin might again attempt to retrieve the record, but without specifying a Max-segment-count.
   - Go to step 7.
4. If record R will not fit in the remainder of the aggregate response, go to step 8.
5. If record R will fit in the remainder of the aggregate response, but no starting fragment will fit in the current segment (Note: this condition precludes the possibility that the segment is empty; see the note preceding step 1):
   - Transmit a Segment request (begin preparation of the next segment).
   - go to step 2.
6. (Note: having reached this step, Record R will fit in the remaining segments; it will not fit within the current segment, but a starting fragment will fit in the current segment.)
   - Fit the largest possible starting fragment of record R and transmit a Segment request.
   - Fill as many complete segments as necessary (which may be zero) with intermediate fragments of record R and send Segment requests.
   - Begin preparation of the next segment, first inserting the final fragment of record R.
7. - Set R = R+1.
   - If R is less than or equal to N, go to step 2.
8. Send a Present response.

<u>Illustration</u>

Assume the origin has requested records 1 through 12. All records are 500 bytes, except record 5, which is 10,000 bytes. Max-segment-size is 3200.

(1) Suppose record 5 consists of 10 elements, each 1000 bytes. The target is able to segment record 5, but only at element boundaries; the target will not let the elements span fragments.

Note: this means that the target may segment the record such that a fragment consists of bytes M+1 through M+N, where M and N are multiples of 1000; M= 0,1, ...9; N = 1, 2, ..., 10-M.

Suppose further that the target cannot segment any other records. The aggregate Present response is as follows:

segment 1: Segment request consisting of records 1 through 4, and the first 1000 bytes of record 5 as a starting fragment.

Note: the size of the segment is 3000 bytes, which is less than the Max-segment-size of 3200; the target cannot fit another fragment in the segment because the minimum fragment size is 1000 bytes which would cause the segment size to exceed 3200 bytes.

segment 2: Segment request consisting of bytes 1001 through 4000 of record 5 as an intermediate fragment.

segment 3: Segment request consisting of bytes 4001 through 7000 of record 5 as an intermediate fragment.

segment 4: Segment request consisting of bytes 7001 through 10,000 of record 5 as a final fragment.

segment 5: Segment request consisting of records 6 through 11.

segment 6: Present response consisting of record 12.

(2) Suppose further that the target can segment the smaller records into 100 byte fragments (or multiples).

Segments 1 through 3 are as in case (1).

segment 4: Segment request consisting of bytes 7001 through 10,000 of record 5 as a final fragment, and bytes 1 through 200 of record 6 as a starting fragment.

segment 5: Segment request consisting of bytes 201 through 500 of record 6 as a final fragment, records 7 through 11, and the first 400 bytes of record 12 as a starting fragment.

segment 6: Present response consisting of bytes 401 through 500 of record 12 as a final fragment.

(3) Suppose the target can segment any of the records at arbitrary byte boundaries.

segment 1: Segment request consisting of records 1 through 4 and the first 1200 bytes of record 5 as a starting fragment.

segment 2: Segment request consisting of bytes 1201 through 4200 of record 5 as an intermediate fragment.

segment 3: Segment request consisting of bytes 4201 through 7400 of record 5 as an intermediate fragment.

segment 4: Segment request consisting of bytes 7401 through 10,000 of record 5 as a final fragment, record 6, and the first 100 bytes of record 7 as a starting fragment.

segment 5: Present response consisting of bytes 101 through 500 of record 7 as a final fragment, and records 8 through 12.

3.4 Operations and Reference-id

A request from the origin of a particular operation type initiates an "operation", which is terminated by the respective response from the target. The following operation types are defined: Init, Search, Present, Delete, Resource-report, Sort, Scan, and Extended-services. (Thus each origin request type corresponds to an operation type with the exception of the following request types: Trigger-resource-control and Close).

An operation consists of the initiating request and the terminating response, along with any intervening Access-control and Resource-control requests and responses, Trigger-resource-control requests, and Segment requests. An operation is assigned a Reference-id by the origin, the origin includes the Reference-id within the initiating request, and it must be included within each message of the operation. If serial operations is in effect, the Reference-id parameter may be omitted in the initiating request; in that case the reference-id is considered null for that operation, and all other messages of that operation must also omit the Reference-id parameter.

Any message sent from origin to target or vice versa (i.e. any request or response defined by this service definition) is part of an operation (identified by its Reference-id), with the following exceptions:

- a Close request or response is not part of any operation;
- if concurrent operations is in effect any Resource-control or Access-control request or response which does not include a Reference-id is not part of an operation.

This standard does not assume any relationship between a given operation and any subsequent operation even if the latter operation uses the same reference-id. This standard does not specify the contents of the Reference-id parameter, nor its meaning, except to the extent that it is used to refer to an operation. Reference-ids are always assigned by the origin and have meaning only within the origin system. Since there are no semantics attributed to the Reference-id, it has no implied datatype and can only be described as transparent binary data. It is therefore described as ASN.1 type OCTETSTRING.

3.5 Concurrent Operations

If 'concurrent operations' is in effect, the Reference-id parameter is mandatory in an initiating request (however, see note), and the origin may initiate multiple concurrent operations, each identified by a different reference-id.

Note: The Reference-id parameter is always optional in an Init request; 'concurrent operations' does not take effect until negotiation is complete, and is thus not in effect during an Init operation.

Once an operation is initiated, until that operation is terminated, another operation may not be initiated with the same reference-id. This standard does not specify the order in which concurrent operations are processed at the target; the target may process concurrent operations in any manner it chooses.

> Example: the origin may issue a Search request using Reference-id "100", and then issue a second Search request using Reference-id "101" before receiving the Search response from the first Search request. There would then be two concurrent operations. Receipt by the origin of the response corresponding to the second Search request (identified by Reference-id "101") would terminate the second operation, and that might occur before termination of the first operation (identified by Reference-id "100"). The origin might then issue a Present request (against the result set created by the second operation), initiating another operation. In that case, the origin must supply a Reference-id other than "100" (because there is an active operation with that Reference-id). The new Reference-id could (but need not) be "101"; if it is, the target may not assume that there is any implied relationship between this new operation and the previous operation which used Reference-id "101".

No operation may be initiated while an Init operation is in progress. No operation may be initiated within a Z-association after a Close request has been sent or received.

All result sets are, in principle, available to any operation. It is possible that two or more concurrent operations will attempt to reference the same result set. This standard does not specify what happens in that circumstance. The origin should not initiate concurrent Search operations with the same value of Result-set-id.

There are no restrictions on the re-use or management of Reference-ids by the origin (other than the restriction cited above that when the origin uses a Reference-id to initiate an operation, until that operation is terminated it may not use that Reference-id to initiate another operation). The origin might re-cycle Reference-ids randomly among users, or it may manage local threads by assigning different Reference-ids to end-users. The target is not required to know how the origin manages Reference-ids, or in particular, that the origin is using Reference-ids to distinguish different users. There is no requirement for the target to have any knowledge of multiple end users at the origin, the target only interacts with the single origin.

3.6 Composition Specification

For each database supported the target defines one or more schemas (see 3.1.5), and designates one as the default schema. For each schema, the target designates one or more element specification names.

An <u>element specification name</u> is the name of an <u>element specification format</u> (a structure used to express an element specification) or an <u>element set name</u>. The latter is a primitive name. An <u>element specification</u> is an instance of an element specification format, or an element set name.

For the default schema, at least one of the element specification names must be an element set name, and the target designates one as the default element set name for the database.

> Note: the target designates this information either via the Explain facility, or through some mechanism outside of the standard.

For each record to be returned in a Search or aggregate Present response, the target applies a schema (for the database to which that record belongs) to form an abstract database record, to which the target applies an element specification to form another instance of the abstract database record (the latter might be a null transformation), to which the target applies a record syntax, to form a retrieval record.

If the origin includes the parameter comp-spec (in a Present request) the procedures of 3.6.1 apply. For a Search operation, or a Present operation when the parameter Comp-spec is omitted, the default schema is assumed for each record, and the procedures of 3.6.2 apply.

3.6.1 Comp-spec Specified

The Present request parameter Comp-spec includes a set of one or more pairs of a database name and associated composition specification. Each composition specification may include a schema name (or if not, the default schema for the database is assumed) and an element specification. For each record to be returned in the aggregate Present response:

- If the database to which the record belongs is specified (as a component of one of the pairs) then the target forms an abstract database record by applying the corresponding composition specification (i.e. by first applying the schema to the database record to form an abstract database record, and then applying the element specification; where the schema and element specification are from the composition specification), if it is able to do so.
- Otherwise, the target forms an abstract database record by applying the default schema and default element set name for the database to which the record belongs.

The parameter Comp-spec may alternatively consist of a single composition specification with no database specified. In that case, for each record to be returned, if the target is able to form an abstract database record according to that composition specification, it does so. If not, an abstract database record is composed according to the default schema and default element set name for the database to which the record belongs.

The target applies a record syntax (which may be included in the composition specification or within the parameter Preferred-record-syntax) to the resulting abstract database record, to form a retrieval record.

3.6.2 Comp-spec Omitted

When requesting the retrieval of a set of records from a result set, if the parameter Comp-spec is omitted, the procedures of this section apply.

Notes:

(1) This is always the case on a Search request, because the parameter Comp-spec is not included in the definition of the Search request.

(2) This is always the case when version 2 is in force, because the parameter Comp-spec is not defined in version 2.

The Search request parameters Small-set-element-set-names and Medium-set-element-set-names, and the Present request parameter Element-set-names, take the form of a set of one or more pairs of a database name and associated element set name. For each record to be returned in the Search or aggregate Present response, the target first applies the default schema for the database to which the record belongs, to form an abstract database record, and then applies an element set name, as follows:

- If the database to which the record belongs is specified (as a component of one of the pairs), and if the corresponding element set name is valid for the default schema for the database, then the target applies that element set name.
- If not, the target applies the default element set name for the database.

Each of these parameters may alternatively consist of a single element set name with no database specified. In that case, for each record to be returned, if the element set name is valid for the default schema for the database to which the record belongs, the target applies that element set name; if not, the target applies the default element set name for the database.

A target must always recognize the character string "F" as an element set name to mean "full" and which when applied to an abstract database record results in the same record.

A target must always recognize the character string "B" as an element set name to mean "brief" record. This standard does not define the meaning of "brief". Unless the origin knows the target's definition of "brief" for a given schema, it should not assume that any particular elements are included.

The origin may specify a "preferred-record-syntax", which the target applies (to the abstract database record formed by the application of the element set name) to form a retrieval record. If the origin does not specify a preferred-record-syntax the target will select one.

3.6.3 Record Syntax

For each record to be returned in a Search or aggregate Present response, the element set name, or the schema and element specification from the composition specification, results in an abstract database record, as described above, to which the target applies a record syntax, indicated as described above. The term "record syntax" has the following meaning:

- When specified by the origin (either as the value of Preferred-record-syntax or within a composition specification), it takes the form of an OID and refers to an abstract syntax, paired, or to be paired (see note), by the target, with a transfer syntax, that the origin requests the target use for the retrieval records.

- When specified by the target, it refers to an abstract syntax paired with a transfer syntax. It takes the form of an OID or p-context accompanying retrieval records in a Search or Present response.

3.7 Type-1 and Type-101 Queries

This section specifies procedures when Query-type is 1 (or Type-101; see Note 2 below). Type-1 is the "reverse polish notation" (RPN) query. It has the following structure:

```
RPN-Query ::= argument | argument + argument + operator
    argument       ::=    operand | RPN-Query
    operand        ::=    Attribute-set+Term | Result-set-id | Restriction
    Restriction    ::=    result-set-id + attribute-set
    operator       ::=    AND | OR | AND-NOT | Prox
```

Notes:

(1) For Type-1, the Prox operator and the Restriction operand are defined for version 3 only. When version 2 is in effect, a Type-1 query which includes either the Prox operator or Restriction operand is in error.

(2) The Type-101 query is defined as identical to the Type-1 query, with the exception that the Prox operator and Restriction operand are defined not only for version 3, but for version 2 as well. Thus the definition of the Type-101 query is independent of version.

The notation used above is as follows:

```
::=    means "is defined as",
|      means "or",
+      means "followed by", and + has precedence over | (i.e., + is evaluated before |).
```

A Z39.50-conforming target must support the Type-1 query, but support of the Type-1 query does not imply support of any of the defined operators or operands.

The target designates what query types it supports, and which operators and operands.

> Note: The target designates this information either through the Explain facility or through some mechanism outside of the standard.

If the target claims support for the Prox operator, the target should also designate whether it supports the extended result set model for proximity (the extended result set model for searching as described in 1.2 and its specialization for proximity as described in 3.7.2.2). If the target claims support for the Restriction operand, then it must also support the extended result set model for restriction (the extended result set model for searching and its specialization for restriction as described in 3.7.3).

> Note: Only in certain circumstances (detailed below) does support of the Prox operator require support of the extended result set model for proximity. However, support of the Restriction operand always requires support of the extended result set model for Restriction.

3.7.1 Representation and Evaluation of the Type-1 and Type-101 Queries

At the origin, the query is represented by a tree. Each subtree represents an operand, either a simple operand or a complex operand. Each leaf node represents a simple operand: Result-set-id, Attribute-set+Term, or Restriction. Each non-leaf node represents a complex operand: a subtree whose root is an operator, and which contains two subtrees, a left operand and a right operand.

The origin traverses the tree according to a left post-order traversal, to produce a sequence of (simple) operands and operators, which is transmitted to the target.

At the target, evaluation of the sequence of operands and operators is illustrated by the use of a stack. Whenever an operand is encountered, it is put on the stack. Whenever an operator is encountered, the last two items that have been put on the stack are pulled off and the operator is applied as follows.

Each operand represents a set of database records. Each is one of the following:

(a) attribute-set + term -- in which case it represents the set of database records obtained by evaluating the specified attribute-set and term against the collection of databases specified in the Search request.

(b) Result-set-id -- in which case it represents the set of database records represented by the result set identified by Result-set-id.

(c) Restriction operand (Result-set-id + attribute-set): in which case it represents the set of database records represented by the result set identified by Result-set-id, restricted by the specified attribute set (see 3.7.3.1).

    Note: if the Restriction operand occurs the target must support the extended result set model for restriction; otherwise the query is in error.

(d) an intermediate result set (resulting from a previous evaluation placed on the stack) -- in which case it represents the records identified by that result set.

Let S1 and S2 be the sets represented by the left and right operand respectively.

- if the operator is AND, let S be the intersection of S1 and S2;
- if the operator is OR, let S be the union of S1 and S2;
- if the operator is AND-NOT, let S be the set of elements in S1 which are not in S2.
- If the operator is Prox:
    - o if both operands are of form (a) let S be the subset of records in the set (S1 AND S2) for which A ProxTest B is true (see 3.7.2.1) where A and B are the two operands.
    - o Otherwise:
        - -- The target must support the extended result set model for proximity; otherwise the query is in error.
        - -- let R1 and R2 be result sets representing the sets S1 and S2.

            (I.e. Each is either:

            - --- the result set specified by the corresponding operand, if it was of form (b), or
            - --- the hypothetical result set representing the set of records represented by that operand, otherwise.

            In either case, both R1 and R2 are assumed to conform to the extended result set model for proximity.)

        Each entry in R1 and R2 contain positional information, in the form of position vectors. For each record represented by both R1 and R2, consider every ordered pair consisting of a position vector associated with the record as represented in R1 and a position vector associated with the record as represented in R2. For each pair that qualifies according to the ProxTest:

        - --- the record is qualified into the set S; and
        - --- a position vector is created for that record as represented in the resultant set, composed from that ordered pair.

An intermediate result set is then created, which represents the records in the set S, and is put on the stack.

When evaluation of the query is complete (i.e. all query-terms have been processed) there will be one item remaining on the stack (otherwise the query is in error), representing a set of database records, which is the result of the query.

3.7.2 Proximity

3.7.2.1 The Proximity Test

The proximity test, ProxTest, includes a Distance, Relation, Unit,and two boolean flags: Ordered and Exclusion.

- Distance: Difference between the ordinal positional values of the two operands. (E.g. if unit is 'paragraph', distance of zero means "same paragraph".) Distance is never negative.

- Relation: LessThan, LessThanOrEqual, Equal, GreaterOrEqual, Greater, or NotEqual.
- Unit: Character, Word, Sentence, Paragraph, Section, Chapter, Document, Element, Subelement, ElementType, Byte, or a privately defined unit.
- Ordered flag: if set, the test is for "right" proximity only (the left ordinal must not exceed the right ordinal and Distance is compared with the difference between the right and left ordinals); otherwise, the test is for "right" or "left" proximity (Distance is compared with the absolute value of the difference between the left and right ordinals).
- Exclusion flag: if set, "not" is to applied to the operation (for example if the test with Exclusion flag 'off' is "'cat' within 5 words of 'hat'", then the same test with Exclusion flag 'on' is "'cat' not within 5 words of 'hat'").

Example: suppose A and B respectively specify "personal name = 'McGraw,J.' " and "personal name= 'Stengel, C.' ", and:

- Distance is 0,
- Relation is 'equal',
- Proximity-unit is 'paragraph',
- Ordered flag is 'false',
- Exclusion flag is 'false'.

Then the result is the set of records in which both of the personal names occur within the same paragraph. Using the same example, if the Exclusion flag is set to 'true', the result is the set of records in which the two personal names never both occur within the same paragraph.

If the Ordered flag is set to 'true' (and Exclusion flag to 'false') then the result is the set of records in which the personal name 'McGraw, J.' occurs within the same paragraph as, but before, the personal name 'Stengel, C.'.

If distance is instead 1 ('ordered' and 'exclusion' flag 'false') the result is the set of records in which the two personal names occur in adjacent paragraphs. If, in addition, Relation-type is 'less-than-or-equal' the result is the set of records in which the two names occur within the same or adjacent paragraphs.

3.7.2.2 Extended Result Set Model for Proximity

In the extended result set model for proximity, the target maintains positional information, in the form of one or more position vectors, associated with each record represented by the result set, which may be used in a proximity operation as a surrogate for the search that created the result set.

Example:

Let R1 and R2 be result sets produced by Type-1 query searches on the terms 'cat' and 'hat'. In the extended result set model for proximity, the target maintains sufficient information associated with each entry in R1 and with each entry in R2 so that the proximity operation "R1 near R2" would be a result set equivalent to the result set produced by the proximity operation "cat near hat" ("near" is used here informally to refer to a proximity test).

The manner in which the target maintains this information is not prescribed by the standard. Appendix ERS (non-normative) provides examples.

3.7.3 Restriction and the Extended Result Set Model

The Restriction operand specifies a result-set-id and a set of attributes, and it represents the set of database records identified by the specified result set, restricted by the specified attributes.

Example:

Let R be the result set produced by a search on the term 'cat', representing three records:

1 where 'cat' occurs in the title,

2 where 'cat' occurs in the title and as an author, and

3 where 'cat' occurs in the title, as an author, and subject.

Then "R restricted to 'author'" might produce the result set consisting of the entries 2 and 3 of R.

In the extended result set model for restriction, the target maintains information associated with each record represented by the result set, that may be used in the evaluation of a restriction operand as a surrogate for the search that created the result set. The manner in which the target maintains this information is not prescribed by the standard. Appendix ERS (non-normative) provides examples.