Z39.50 Profile for Access to Digital Collections

Draft Seven
(Final Draft for Review)

May 3, 1996
Library of Congress

Contents

1. Background and Introduction . . . -1-
- 1.1 Document History . . . -1-
- 1.2 Purpose of This Profile . . . -1-
- 1.3 Problems Addressed by this Profile . . . -1-
- 1.4 Scope Limitations . . . -2-
- 1.5 Definitions . . . -3-
- 1.6 References . . . -4-

2. Model, Assumptions, and Principles . . . -4-
- 2.1 Model of an Object . . . -4-
- 2.2 Model of a Collection . . . -4-
- 2.3 Model of a Record . . . -5-
- 2.4 Descriptive Information . . . -5-
  - 2.4.1 Descriptive Information and Content . . . -6-
  - 2.4.2 Modelling Descriptive Information . . . -6-
  - 2.4.3 Collection and Object Descriptive Records . . . -7-
- 2.5 Datastore Model . . . -7-
- 2.6 Enumerating the Members of a Collection . . . -7-

3. Navigational Scenarios . . . -8-

(contents continued)

4. Descriptive Record Schema . . . -9-
4.1 Abstract Record Structure . . . -9-
4.2 Datatype Definitions . . . -10-
4.2.1 CollectionInfo . . . -11-
4.2.2 ObjectInfo . . . -11-
4.2.3 AssociatedDescription . . . -12-
4.2.4 RelatedCollection . . . -13-
4.2.5 CollectionAndDb . . . -13-
4.2.6 ServerAndDb . . . -13-
4.2.7 EnumeratedMember . . . -13-
4.2.8 DigitalObject . . . -14-
4.2.9 Description . . . -15-
4.2.10 RecordPointer . . . -15-
4.2.11 BriefTextDescription . . . -16-
4.3 Schematic Representation of Descriptive Record . . . -17-
4.4 Tag Types . . . -17-
4.5 Descriptive Record TagSet . . . -17-
4.6 Element Set Names Defined for this Schema . . . -18-
4.6.1 Brief Record . . . -19-
4.6.2 Navigation Record . . . -19-
4.6.3 Full Record . . . -19-
4.7 Collection-1 Attribute Set . . . -19-
4.7.1 Collection-1 Use Attributes . . . -20-
4.7.2 Collection-1 Relation Attributes . . . -20-

5. Z39.50 Specific Profile Details . . . -21-
5.1 Protocol version . . . -21-
5.2 Services and Facilities . . . -21-
5.3 Search . . . -21-
5.4 Retrieval . . . -21-
5.4.1 Present Parameters . . . -21-
5.4.2 Segmentation . . . -22-
5.4.3 Element Specification . . . -22-
5.4.4 Variant Set . . . -22-
5.4.5 Fragmentation . . . -22-
5.4.6 Record Syntax . . . -22-
5.4.8 Brief and Navigation Descriptive Records . . . -22-
5.4.9 Digital Objects and their Variants . . . -23-
5.4.10 Element Specification eSpec-1 . . . -23-
5.4.11 GRS-1 . . . -24-
5.4.12 Language . . . -24-
5.4.13 Surrogates . . . -25-
5.4.14 Usage Right . . . -25-
5.5 Explain . . . -25-

6. Conformance . . . -26-
6.1 Basic Conformance . . . -26-
6.2 Basic V3 Conformance . . . -26-
6.3 Enhanced V3 Conformance . . . -26-

1. Background and Introduction

In August, 1995, the Library of Congress convened a team of representatives from several institutions to develop a Z39.50 profile access to digital libraries.

Early in its development the profile was renamed from Z39.50 Profile for the Digital Library Application to Z39.50 Profile for Access to Digital Collections (nicknamed the *Collections* profile) as its scope was narrowed, to apply to *navigation* of digital collections. Other groups were initiating independent efforts to develop profiles aimed at specific types of objects and collections. The intention was to coordinate these efforts and that these latter profiles would be developed as compatible extensions or subsets of the Collections profile. These latter profiles are referred to as *companion profiles*. An example is the CIMI profile: Z39.50 Application Profile Specification for Use in Project CHIO. CIMI, the Consortium for Interchange of Museum Information, is the sponsor of a demonstration project, CHIO (Cultural Heritage Information Online). Another example is the Z39.50 Profile for Access to Digital Libraries being developed at the Library of Congress.

1.1 Document History

The initial draft of this document was developed in early September, 1995; *Draft Two* was issued for discussion at the first meeting of the profile team, September 20. *Draft Three* was issued November 3, for the second meeting, November 14-15. *Draft Four*, issued in late December was prepared for a meeting that was subsequently canceled. *Draft Five* was developed via electronic mail and other discussion, for the meeting February 5-6, and was also distributed to the ZIG for discussion at the ZIG meeting February 7-9. *Draft Six*, issued March 18, reflected extensive discussion at the February 5-6 meeting, the February 7-9 ZIG meeting, as well as subsequent comments.

This draft, *Draft Seven*, reflects comments on Draft Six, and is available for review and comment until June 1. Following comment resolution, the profile will be proposed for approval.

1.2 Purpose of This Profile

This profile specifies a conforming subset of Z39.50-1995 to address problems described in section 1.3, in particular for access to digital collections organized via descriptive information whose structures are described within this profile. It provides semantics for navigating digital collections, to locate and retrieve objects of interest.

1.3 Problems Addressed by this Profile

Libraries and other institutions are creating a growing number of collections, organized thematically, for example, by subject, creator, historical period, etc.; with numerous, diverse objects, both digital and physical. These collections are often organized hierarchically and distributed across servers.

Significant resources may be invested in digitization and in the intellectual efforts of aggregation, organization, and description of the information in a collection. Yet to a remote user or client, the collection may appear to be simply an accumulation of objects and undifferentiated data, because there is no agreed-upon semantics for navigating the collection, to locate and retrieve objects of interest. Coherent organizational structures, imposed on the data, are necessary to provide a view that supports navigation.

A key obstacle to effective navigation is the inability to distinguish content from description. A primary goal of navigation is to locate and retrieve objects of interest; a vital step in that process is to locate relevant descriptive information. Thus it is useful to navigate among descriptive information as well as content, and consequently, to be able to distinguish content from description.

Various forms of descriptive aids have been developed for purposes of describing collections and objects, for example, finding aids, encoded archival descriptions, and exhibition catalogs. Often they do not have a well-defined structure and cannot be used alone for reliable navigation. At many institutions though, descriptive aids of these types have been created, at significant expense. It is therefore imperative that an application relying on descriptive information for collections and objects be able to utilize these available aids, rather than mandate the creation of new redundant structures. In order to exploit these descriptive aids, a higher level enveloping structure is necessary, to allows users and clients to navigate among and select descriptive aids of interest.

Another navigational problem is posed by the various and often complex relationships among and between objects and collections. For a given object, there may be other objects that are duplicates or variations (e.g. different resolutions) or which bear other relationships (e.g. surrogates). For any given collection there may be superior, subordinate, related, and context collections. These relationships among objects and collections must be modelled coherently.

Different digital objects, even within the same collection, may be retrievable by different protocols (e.g. Z39.50, HTTP, FTP). It is therefore important that structures describing these objects, and how they may be accessed, be defined independent of any specific protocol.

These problems amplify when a collection is distributed across institutions (or otherwise distributed in a manner requiring access to multiple servers). In particular some normalization of query semantics is necessary for coherent navigation of collections. Clients must be able to formulate queries that will be interpreted consistently across servers.

1.4 Scope Limitations

This profile assumes that *companion profiles* (compatible extensions to or subsets of this profile) will be independently developed, extending or limiting the use of this profile to specific applications or classes of information, for example, museum objects, satellite photographs, geospatial data, or chemical compounds. Thus this profile does not directly address all of the problems cited in 1.3, but provides a framework for companion profiles to do so:

- Digital Objects are treated as atomic, that is, their content is opaque and is not addressed by this profile. Thus the profile addresses searching descriptive information rather than searching Digital Objects. Companion profiles may model the content of specific objects (e.g., museum objects).
- Associated Descriptions (e.g., finding aids, exhibition catalogs; see 2.3.2) are treated as opaque (their content is not addressed by this profile), though clients may have at their disposal helper applications that are able to process or display them. Companion profiles may model the content of Associated Descriptions.
- The profile does not model complex relationships among objects of all classes. Companion profiles may do so for specific object classes.

Although this profile emphasizes the logical separation of descriptive information from content, it does not include guidelines or specifications for determining whether information is description or content. The profile does not address organizing principles, i.e, the means by which objects are aggregated into collections.

Although this profile addresses distributed collections, it does not address distributed databases. Different parts of a collection may be managed by different institutions; therefore different databases, corresponding to a single collection, may reside on different servers, but no individual database is distributed across servers.

Only the Z39.50 protocol is addressed by the profile. The profile does not, however, preclude multi-protocol clients, gateways, or digital collections where part of a collection is accessible via Z39.50 while another part is accessible by other protocols.

1.5 Definitions

Associated Description: A unit of descriptive information associated with a collection or object, for example, an encoded archival description or a finding aid (for an archival collections), or a catalog record (for library materials).

Child: Collection or object A is a child of collection B if B is a parent of A. A collection may have objects as well as subcollections as children.

Collection: A group of related objects and/or collections, possibly distributed across locations. A collection is a tree, where leaf nodes are objects and non-leaf nodes are subcollections.

Collection Descriptive Record: A Descriptive Record that describes a Collection.

Companion profile: An independently developed profile which is a compatible extension to or subset of this profile, extending or limiting the use of this profile to specific applications or classes of information, for example, museum objects.

Context Collection: Collection A is a context collection for collection or object B if it is a superior, related collection, and the organization with responsibility for the management of B considers that although collection A may be relevant to a user who is interested in B, any collection superior to collection A is likely not relevant.

Descriptive Record: A unit of descriptive information at a higher level of abstraction than an Associated Description. A Descriptive Record may include one or more Associated Descriptions in addition to other information that describes either a collection (and possibly its contents) or an object within a collection. A Descriptive Record is either a Collection Descriptive Record or an Object Descriptive Record.

Member (of a collection): Child.

Object: A Physical Object or a Digital Object.

Object Descriptive Record: A Descriptive Record that describes an object.

Parent Collection: Collection A is a parent of collection or object B if A is immediately superior to B.

Related Collection: Collection A related to collection or object B if the organization with responsibility for the management of B considers that collection A may be relevant to a user who is interested B.

Subcollection: Child collection.

Subordinate: Collection or object A is subordinate to collection B if A is a node on a tree whose root is B.

Superior Collection: Collection A is superior to collection or object B if B is a node on a tree whose root is A.

1.6 References

ANSI/NISO Z39.50-1995. *Information Retrieval (Z39.50): Application Service Definition and Protocol Specification*. See http://lcweb.loc.gov/z3950/agency.

CIMI Profile. *Z39.50 Application Profile Specifications for Use in Project CHIO.*

Z39.50 Profile for Access to Digital Libraries. Library of Congress.

2. Model, Assumptions, and Principles

The data model and descriptive data described in this section and section 4 are intended for general use, not specific to Z39.50. In particular, the intellectual content of the descriptive data modelled by this profile is distinguished from the format used for its transfer. Z39.50 specifics are described in section 5.

2.1 Model of an Object

An *object* is a *Physical Object* or a *Digital Object*. For any given Physical Object, there may or may not be a Digital Object which is its digital representation. Any given Digital Object may or may not be the digital representation of some Physical Object. A Digital Object may be text or it may be a digitized object; in any case, this profile treats a Digital Object as opaque.

2.2 Model of a Collection

A *collection* is a group of related objects and/or collections, possibly distributed across locations. Thus a collection is a tree, where leaf nodes are objects and non-leaf nodes are subcollections.

Collection A is said to be *related* to collection or object B if the organization with responsibility for the management of B considers that collection A may be relevant to a user who is interested B.

Collection A is said to be *superior* to collection or object B if B is a node on a tree whose root is A; B is said to be *subordinate* to collection A if A is superior to B.

Collection A is said to be a *parent* of collection or object B if A is immediately superior to B; B is said to be a *child* of collection A if A is a parent of B.

Collection A is said to be a *context* collection for collection or object B if it is a superior, related collection, and the organization with responsibility for the management of B considers that although collection A may be relevant to a user who is interested in B, any collection superior to collection A is likely not relevant.

The concept of a context collection is related to, but differs from, the concept of a root collection (which this profile does not employ). A context collection might informally be considered a "relative root" collection, for purposes of navigation. "Root" (as in "root collection") has the connotation of absolute or static. In contrast, "context" (as in "context collection") is relative; furthermore, although a collection might be a root collection from the view of a subordinate collection, a parent collection may subsequently be created for the former, so that it is no longer a root collection, and this new relationships might not be conveyed (nor might it be meaningful) to the subordinate collection.

For any given collection or object there may be zero or more *parent* collections, zero or more *related* collections and zero or more *context* collections. For any given collection there may be zero or more child collections.

A collection is a different logical construct than a database, in several respects:

- A database is an aggregation of records, and a collection is an aggregation of objects, some of which may be physical objects.
- The apportionment of records into databases might be based on very different criteria than the apportionment of objects into collections. For example, a server might provide the following databases: Books, Serials, Maps, Photos, SoundRecordings, and MotionPictures, containing digitized books, serials, maps, photos, sound recordings and motion pictures, respectively. A single collection might contain objects from all of these categories, and therefore records (corresponding to objects in the collection) would be distributed across these databases. The collection might be organized by theme, while the databases are organized, for example, by media type.
- The property of belonging to a collection differs from the property of belonging to a database; section 2.6 describes this difference.
- A database resides on a single server, while a collection may span servers (that is, different databases corresponding to a single collection may reside on different servers).
- A single database may include records corresponding to objects in more than one collection.
- A single object might belong to more than one collection, while the database record for that object may be retrievable from only one database.
- An object might belong to a single collection, while the database record for that object is retrievable from multiple databases, which may be on different servers.

2.3 Model of a Record

Z39.50 models database records, abstract database records, and retrieval records. It employs the concept of an abstract record structure defined by a database schema: a common understanding shared by client and server of the information contained in the records of a database, allowing logical components of that information to be addressed, selected, and represented.

A database record is a unit of information in a database, represented in a data structure local to the server. An abstract database record is an abstract representation of the information in a database record, formed by the application of an abstract record structure to the database record, where the abstract record structure is known to the client. A retrieval record is the information in an abstract database record represented in an exportable structure (by the application of a record syntax).

This profile employs the concept of a record according to the Z39.50 model, thus the term *record* is used to mean database record, abstract database record, or retrieval record, depending on usage and context. The Descriptive Record (Collection Descriptive Record and Object Descriptive Record) is a fundamental construct in this profile. The term *Descriptive Record* may refer to a database record, in particular when the context is searching. It refers to an abstract database record when the context is the definition of the Descriptive Record. It refers to a retrieval record when the context is retrieval.

2.4 Descriptive Information

This profile exploits organizational structures in order to allow a client to navigate through structured information, possibly hierarchically structured. A coherently defined set of descriptive data is used to manage and navigate collections of otherwise undifferentiated data. These organizational structures allow the data to be viewed as distributed collections, loosely hierarchical. (From a top-down view, a collections is a tree. From a bottom-up view, collections are directed graphs, because any object or collection may have multiple parents.)

2.4.1 Descriptive Information and Content

Collections modelled by this profile are characterized by the logical separation of descriptive information from content. This logical separation is not intended to constrain an implementation. Servers may choose to separate descriptive data from content in retrieval records using the record structure defined in this profile. However, servers are not required to perform this separation. In any case the profile does not prescribe nor address how servers store data.

In general, a collection may include Digital Objects that are completely self-describing, Digital Objects that include some self-description while additional descriptive information is separate, as well as Digital Objects where descriptive information is completely separate.

This profile does not provide guidelines for determining whether information is description or content. It is the responsibility of the institution managing a collection to make that determination, and this profile does not presume that different institutions will make the same determination for similar or even identical information. For example, a Physical Object, for which there is a digitized picture, could be modelled as a Physical Object with no corresponding Digital Object, where the picture is treated as descriptive information; alternatively, the picture could be treated as a Digital Object.

As another example, a Digital Object might be largely self-describing, including, for example, a caption that is an integral part of the Digital Object. A copy of that Digital Object might reside in a different collection, managed by a different institution, that might designate it instead to be description (of a physical object) with no corresponding Digital Object.

2.4.2 Modelling Descriptive Information

Collections and objects often have descriptive information associated with them. For example, archival collections are described in encoded archival descriptions and finding aids; library and other materials are described in catalog records; museum objects and collections are described in wall texts and exhibition catalogs.

An important design objective of this profile is to support the use of such descriptive information, here called an *Associated Description*, for use in navigating collections. A server implementing this profile may have created and stored an existing base of Associated Descriptions; it is the intention of this profile that the server be able to deliver these existing Associated Descriptions without the need to modify or reformat them.

Because of the wide variety of possible descriptive information, this profile does not attempt to enumerate or restrict the types of Associated Description that may be available. It is anticipated that companion profiles will enumerate the types of Associated Description appropriate for particular domains.

Neither does this profile attempt to describe the structure or contents of any Associated Description; for the purpose of this profile an Associated Description is an opaque data element passed from the server to the client along with other information relevant to navigating collections. A client that receives an Associated Description may either display it to the user or pass it to some application that can process it. The nature of such processing is outside the scope of this profile.

This profile supports the retrieval of information needed to navigate within and between collections and to retrieve both Digital Objects and Associated Descriptions. It does so by defining the structure of a Descriptive Record, which is at a higher level of abstraction than either a Digital Object or an Associated Description. A Descriptive Record may include one or more Associated Descriptions in addition to other information that describes either a collection (and possibly its contents) or an object within a collection.

This profile requires that a server be able to provide Descriptive Records (for collections and/or objects). These could be stored as database records, or alternatively, retrieval records may be created upon request by appropriate processing of other information available to the server, including information within digital objects and Associated Descriptions. This is, however, strictly an implementation matter; this profile does not specify how, or if, the Descriptive Records are to be stored. It only specifies the format for transfer from server to client. (Thus, although according to the Z39.50 abstract record model, a retrieval record corresponds to a database record, whether that database record physically exists on the server is transparent to the client.)

2.4.3 Collection and Object Descriptive Records

A Descriptive Record describes a collection or an object and is, respectively a *Collection Descriptive Record* or *Object Descriptive Record*.

A Collection Descriptive Record may provide an overall description of a collection as well as collective or individual descriptions of some or all of the objects in the collection.

An Object Descriptive Record describes an object: a Digital Object (which may or may not be the digital representation of a Physical Object) or a Physical Object.

Both a Collection Descriptive Record and Object Descriptive Record may list parent-, context-, and related- collections. A Collection Descriptive Record may list child collections as well.

For a given Digital Object, there need not necessarily be an Object Descriptive Record. The institution managing a collection to which the object belongs is responsible for making that determination. An object may, for example, be sufficiently described by a Collection Descriptive Record. As another example, a set of digitized photographs might be distinguished as distinct Digital Objects and aggregated into a collection or might be combined as a single Digital Object. If they are aggregated into a collection, there may be a single Collection Descriptive Record and there may or may not be individual Object Descriptive Records for some or all of the individual Digital Objects.

2.5 Datastore Model

A collection is represented by a *datastore*: part or all of one or more databases on one or more servers (distributed databases, however, are not addressed by this profile; see 1.4). To a given collection there corresponds a set of databases that include the descriptive records for that collection: Object Descriptive Records for objects and Collection Descriptive Records for subcollections.

The databases corresponding to a collection are not exclusive to that collection. They may include other records: records pertaining to other collections, or records not related to digital collections.

This model does not prescribe where Digital Objects reside; any given Digital Object may be represented as nested within its Object Descriptive Record, as a distinct record within the same database as its Object Descriptive Record, on a different database within the same server, or on a different server. A Digital Object may be accessible via Z39.50 or via another protocol.

2.6 Enumerating the Members of a Collection

For modelling purposes, *members* of a collection are objects and subcollections.

> Note: thus the property of belonging to a collection differs from the property of belonging to a database. For example, suppose an Object Descriptive Record describes a Physical Object in a collection. The Object Descriptive Record belongs to a database (but is not a member of the collection, because, by definition, only objects and subcollections are members) while the Physical Object is a member of the collection (but does not belong to a database).

A Collection Descriptive Record may provide an enumeration of the members of the collection it describes. Each item in the enumerated list is a pointer that refers to one of the following:

- an Object Descriptive Record (if the member is an object),
- a Digital Object (if the member is a Digital Object), or
- a Collection Descriptive Record (if the member is a subcollection).

3. Navigational Scenarios

This section describes some of the scenarios that sections 4 and 5 are intended to support.

When a client connects to a server providing access to digital collections, on behalf of a user interested in digital collections, the following scenarios are possible.

- The user might know the name of a collection of interest, as well as the database where the Collection Descriptive Record for the collection resides.
- The user may know the name of a collection but not the database where its Collection Descriptive Record resides. In that case, if the client and server both support Explain then the client may attempt to determine that database via Explain (see 5.5.)
- It may be that neither the client nor the user knows any collection names. In that case if the client and server both support Explain, the client might attempt to learn which databases in general correspond to collections and search those databases for desired Collection Descriptive Records. The client may then retrieve Collection Descriptive Records from these databases and display summary information to the user, including brief descriptions.
- The user may select a collection and then navigate to other collections of interest: the client may retrieve a list of related collections, including parent, superior, and context collections, brief descriptions of these collections, and descriptions of their relationship to the subject collection. The user might select one of these collections, determine its parent, superior, and context collections, etc.

In any case, eventually the user may identify a collection of interest. A number of possible scenarios might ensue:

- The client might retrieve an enumeration of the Associated Descriptions for the collection. For each Associated Description, the client may retrieve the Associated Description type, format, size, and a brief text description. The client might use the Associated Description types and format to determine which Associated Descriptions it can process or if not process, display to the user. The client might then display the brief descriptions (and size) of those Associated Descriptions to the user, who then selects an Associated Description of interest for the client to process or display.
- The client might retrieve from the Collection Descriptive Record an enumeration of the objects in the collection. For each object, the client retrieves a brief description for display to the user (if the number of enumerated objects is large, the client can retrieve a few at a time). For a given object:
    - •• The client might retrieve part or all of the Object Descriptive Record for the object, in particular an enumeration of its Associated Descriptions, and, as in the Collection Descriptive Record scenario above, the client displays brief descriptions of Associated Descriptions to the user, who selects an Associated Description of interest for the client to process or display.
    - •• Alternatively, the client might retrieve and display object metadata including a list of available formats for the object, and for each, its size, and level of integrity (see 5.4.9). The user might select a format and the client may then retrieve the object in that format.
- The client may search the collection based on user supplied search criteria. (Here, "search the collection" means search the databases corresponding to the collection, where the search is restricted to Object Descriptive Records which list the collection as a parent.) User supplied search criteria are not addressed by this profile, but may be addressed by companion profiles. If the client and server both support Explain the client may retrieve information about the databases, for example, attributes supported, in order to assist the user in formulating meaningful searches. (Note that this profile does not require support for Explain.)

- The client might expand the search to apply to superior collections and/or subordinate collections.

The user might identify an object of interest and wish to locate other objects, related in a specific way. The client may assist the user by listing the various collections to which the object belongs, including descriptions of criteria used to aggregate objects in each collection.

For example, consider a collection of digitized theater costumes, with various subcollections. One of the objects is a digitization of an eighteenth century, yellow, Shakespearean costume. That single object might belong to a number of collections, including "18th century costumes", "yellow costumes" and "shakespearean costumes". A user interested in that object might select one of these collections to search, based on the specific aspect of interest.

Note that the scenarios above suggest that a collection may know of subordinate collections and objects, and a collection or object may know of superior collections; correspondingly, a client might build a collection-map bottom-up or top-down. It must be emphasized that this profile does not mandate nor does it intend to favor either approach. A companion profile may choose to favor or mandate either approach.

4. Descriptive Record Schema

This section defines a schema for the Descriptive Record. The object identifier for this schema is {iso (1) member-body (2) US (840) ANSI-standard-Z39.50 (10003) schema (13) descriptive record (3)}.

4.1 Abstract Record Structure

The abstract record structure for a Descriptive Record is defined as follows:

Element	Occurrence	Repeat-able	Datatype
typeOfDescriptiveRecord	mandatory	no	INTEGER
briefDescription	optional	no	BriefTextDescription
collectionInfo	optional; occurs only if typeOfDescriptiveRecord is 1	no	CollectionInfo
objectInfo	optional; occurs only if typeOfDescriptiveRecord is 2	no	ObjectInfo
associatedDescription	optional	yes	AssociatedDescription
relatedCollection	optional	yes	relatedCollection

Element *typeOfDescriptiveRecord* distinguishes whether the Descriptive Record describes a collection or an object, i.e., it distinguishes a Collection Descriptive Record from an Object Descriptive Record. It is one of the following:

1 = Describes a collection
2 = Describes an object
3 = Unspecified

Note: 'Unspecified' in included cases when there is a unit of information, possibly an Associated Description, where it is unclear whether that information is a collection or object, yet the information is interesting enough to warrant a descriptive record.

Element *briefDescription* is intended for use in a brief record, for display to the user. It is a brief description of the collection or object that the Descriptive Record describes.

Element *collectionInfo* consolidates elements that occur only in a Collection Descriptive Record (as opposed to an Object Descriptive Record). It includes the name of the collection, the databases that pertain to the collection, and an enumeration (possibly partial) of the members of the collection.

Element *objectInfo* consolidates elements that occur only in an Object Descriptive Record (as opposed to a Collection Descriptive Record). It indicates the type and category of the object, as well as the Digital Object itself (or a pointer to the Digital Object).

Each occurrence of element *associatedDescription* represents an Associated Description. Multiple occurrences of this element represent different Associated Descriptions (to represent multiple representations of the same Associated Description, see 4.2.3).

Each occurrence of element *relatedCollection* identifies a parent-, superior-, related-, or context collection. The name of the collection and the database where Collection Descriptive Record resides are indicated, as well as the nature of the relationship.
When the Descriptive Record is an Object Descriptive Record (typeOfDescriptiveRecord = 2), this element may indicate the collections to which the object (described by this Descriptive Record) belongs.

4.2 Datatype Definitions

This section defines the following datatypes:

- CollectionInfo
- ObjectInfo
- AssociatedDescription
- CollectionAndDb
- RelatedCollection
- ServerAndDb
- EnumeratedMember
- DigitalObject
- Description
- RecordPointer
- BriefTextDescription

4.2.1 CollectionInfo

Datatype *CollectionInfo* is structured as follows:

Element	Occurrence	Repeatable	Datatype
collectionName	mandatory	no	InternationalString
database	optional	yes	ServerAndDb
enumeratedMember	optional	yes	EnumeratedMember
fullyEnumerated	optional	no	BOOLEAN
childrenKnowThisParent	optional	no	BOOLEAN

Element *collectionName* is the name of the collection.

Each occurrence of element *database* identifies a database. The collective set listed is the set of databases that pertain to the collection. The set of databases may span servers.

When this element is omitted, the client should infer that it was omitted either because the server did not know, or did not wish to indicate, the pertinent databases. When one or more occurrences are included, the client should not infer that all of the pertinent databases are necessarily included.

The occurrences of element *enumeratedMember* enumerate the members of the collection. Each occurrence corresponds to an object or subcollection. The enumeration may or may not be inclusive, as indicated by the element fullyEnumerated. Each occurrence may include a pointer to an Object Descriptive Record, to a Digital Object, or to a Collection Descriptive Record.

Element *fullyEnumerated* indicates whether or not the enumeration of members, as enumerated by the occurrences of element enumeratedMember, is a complete enumeration of the members of the collection. If this element is not supplied no default should be inferred; it should be omitted only if the target does not know or does not wish to indicate if the enumeration is complete.

Element *childrenKnowthisParent*, if 'true', means that every member of this collection lists this collection as a parent. If this element is not supplied no default should be inferred; it should be omitted only if the target does not know or does not wish to indicate its value.

Note: A value of 'true' for element fullyEnumerated indicates that the client may discover the members of the collection via retrieval (by retrieving occurrences of element enumeratedMember). A value of 'true' for element childrenKnowThisParent indicates that the client may discover the members via search (see 4.7.1, Collection-1 Use attribute Parent-collection). If both elements have value 'false', this indicates that there is no straightforward way to discover the members of the collection.

4.2.2 ObjectInfo

Datatype *ObjectInfo* is structured as follows:

Element	Occurrence	Repeatable	Datatype
typeOfObject	mandatory	no	INTEGER
categoryOfObject	optional	no	InternationalString
digitalObject	optional; may occur only if typeOfObject less than 4	yes	DigitalObject

Element *typeOfObject* is one of the following:

- 0 = Unspecified.
- 1 = Object is a Digital Object.
- 2 = Object is a Digital Object for which there is a Physical Object.
- 3 = Object is a Digital Object for which there is no Physical Object (for example when an object is originally created electronically).
- 4 = Object is a Physical Object.
- 5 = There is no (separate) object.

Element *categoryOfObject* may be included to indicate that a Digital Object is, for example, an exhibition catalog or cataloging record (these examples are purely illustrative and not intended as inclusive). This profile does not define any categories. It is anticipated that companion profiles will do so.

Note: See also categoryOfAD within AssociatedDescription. A given object in one collection might be treated as an Associated Description in a different collection (perhaps managed by a different institution). Therefore, there may be categories of objects in common with categories of Associated Descriptions.

Each occurrence of element *digitalObject* includes the actual Digital Object, or a pointer to the Digital Object. Multiple occurrences of this element identify different representations of the Digital Object, such as variant resolutions, color depths or encodings (they do not represent different Digital Objects; datatype ObjectInfo represents a single, logical Digital Object).

4.2.3 AssociatedDescription

Datatype *AssociatedDescription* is structured as follows:

Element	Occurrence	Repeatable	Datatype
briefDescriptionOfAD	optional	no	BriefTextDescription
categoryOfAD	optional	no	InternationalString
description	mandatory	yes	Description

Element *briefDescriptionOfAD* is a brief text description of the Associated Description, for display purposes.

Element *categoryOfAD* may be included to indicate that the Associated Description is, for example, an encoded archival description, exhibition catalog, or cataloging record (these examples are purely illustrative and not intended as inclusive). This profile does not define any categories. It is anticipated that companion profiles will do so.

Note: See also categoryOfObject (and its note) within ObjectInfo.

Each occurrence of element *Description* includes the actual Associated Description or a pointer to the Associated Description. Multiple occurrences of this element represent different representations of the Associated Description (they do not represent different Associated Descriptions; multiple occurrences of associatedDescription in the main structure may be used to represent different Associated Descriptions).

4.2.4 RelatedCollection

Datatype *RelatedCollection* is structured as follows:

Element	Occurrence	Repeatable	Datatype
collection	mandatory	no	CollectionAndDb
relationship	optional	yes	INTEGER
relativeLevel	optional	no	INTEGER
descriptionOfRelationship	optional	no	InternationalString

Element *collection* identifies the related collection.

Element *relationship* is one of the following:

1 = superior collection
2 = context collection
3 = collection may be of interest (no familial relationship implied)

Notes:

(1) The subordinate relationship is not included here, because subordinate collections are enumerated within element enumeratedMember of collectionInfo.

(2) Relationship is repeatable, and the valid combinations are 1; 1,2, and 3; and 3.

Element *relativeLevel* indicates the level of this collection relative to the (subject) object or collection. A value 0 means "sibling" (or "same level"), 1 means "parent", and a value greater than 1 means "superior but not parent".

Element *descriptionOfRelationship* provides a text description of the nature of the aggregation, relationship, or context, and may indicate why the collection might be of interest.

4.2.5 CollectionAndDb

Datatype *CollectionAndDb* is structured as follows:

Element	Occurrence	Repeatable	Datatype
collectionName	mandatory	no	InternationalString
database	mandatory	no	ServerAndDb

Element *collectionName* is the name of a collection.
Element *database* identifies a server and database.

4.2.6 ServerAndDb

Datatype *ServerAndDb* is structured as follows:

Element	Occurrence	Repeatable	Datatype
server	optional; if omitted, the indicated database is assumed to be on this server; must be supplied if indicated database is not on this server	no	InternationalString
db	mandatory	no	InternationalString

Element *server* identifies a host and port.
Element *db* is a database on the indicated server.

4.2.7 EnumeratedMember

Datatype *EnumeratedMember* has the following structure:

Element	Occurrence	Repeatable	Datatype
briefDescriptionOf Member	optional	no	BriefText Description
pointer	optional	no	RecordPointer
whatPointerPointsTo	occurs if and only if pointer occurs	no	INTEGER

Element *briefDescriptionOfMember* is a brief text description of the object or subcollection.

Element *pointer* is a pointer to an Object Descriptive Record or Digital Object, if the member is an object, or to a Collection Descriptive Record if the member is a subcollection.

Element *whatPointerPointsTo* is one of the following:
1 = element *pointer* points to an Object Descriptive Record
2 = element *pointer* points to a Digital Object
3 = element *pointer* points to a Collection Descriptive Record
4 = type of pointer is Unspecified

4.2.8 DigitalObject

Datatype *DigitalObject* is structured as follows:

Element	Occurrence	Repeatable	Datatype
briefDescriptionOfDOVariant	optional	no	BriefTextDescription
actualDO	occurs if and only if pointerToDO does not occur	no	variant or category dependent; default is OCTET STRING
pointerToDO	occurs if and only if actualDO does not occur	no	RecordPointer
authoritative	optional	no	BOOLEAN

Element *briefDescriptionOfDOVariant* is a brief text description of this particular variant or representation of the Digital Object. It is intended to allow the user to discriminate among variants or representations of the Digital Object. It is not intended as a description of the Digital Object itself; that description is provided either from element briefDescription in the Descriptive Record (when the object is obtained via an Object Descriptive Record) or from element briefDescriptionOfMember in enumeratedMember (when the Digital Object is discovered via a Collection Descriptive Record).

Element *actualDO* is the actual Digital Object.

Element *pointerToDO* identifies a retrievable information object that is (or contains) the Digital Object. It may be retrievable via Z39.50 or via other mechanisms outside the scope of this profile.

Element *authoritative* indicates whether the server deems this representation of the digital object to be an authoritative representation. The server may provide several representations of the Digital

Object (represented by multiple occurrences of element digitalObject within ObjectInfo) some considered authoritative and others not considered authoritative, as reflected by the value of this element.

4.2.9 Description

Datatype *Description* is structured as follow:

Element	**Occurrence**	**Repeatable**	**Datatype**
briefDescriptionOfADVariant	optional	no	BriefTextDescription
actualAD	occurs if and only if pointerToAD does not occur	no	variant or category dependent; default is InternationalString
pointerToAD	occurs if and only if actualAD does not occur	no	RecordPointer
authoritative	optional	no	BOOLEAN

Element *briefDescriptionOfADVariant* is a brief text description of this particular variant or representation of the Associated Description, intended to allow the user to discriminate among variants or representations of the Associated Description. This element is not intended as a description of the Associated Description itself; that description is provided by element briefDescriptionOfAD within associatedDescription.

Element *actualAD* is the actual Associated Description.

Element *pointerToAD* identifies a retrievable information object that is (or contains) the Associated Description. It may be retrievable via Z39.50 or via other mechanisms outside the scope of this profile.

Element *authoritative* indicates whether the server deems this representation of the Associated Description to be an authoritative representation. The server may provide several representations of the AssociatedDescription (represented by multiple occurrences of element description within associatedDescription) some considered authoritative and others not considered authoritative, as reflected by the value of this element.

4.2.10 RecordPointer

Datatype *RecordPointer* has the following structure:

Element	Occurrence	Repeatable	Datatype
database	occurs (in conjunction with recordId) if and only if alternativeIdentifier does not occur	no	ServerAndDb
recordId	occurs (in conjunction with database) if and only if alternativeIdentifier does not occur	no	OCTET STRING
alternative Identifier	occurs if and only if Database does not occur	no	OCTET STRING
typeOf Identifier	optional; occurs only if alternativeIdentifier occurs	no	OBJECT IDENTIFIER or InternationalString

RecordPointer is used in the following element: pointer within EnumeratedMember, pointerToDO within DigitalObject, and pointerToAD within Description. It points to a database record or other retrievable information object corresponding respectively to: the Descriptive Record for an object or collection enumerated by EnumeratedMember, a Digital Object, or an Associated Description. It takes one of the following two forms:

(a) a database (serverAndDb) and unique identifier of a record within the database (recordId),
(b) some alternative identifier.

In case (a) the format of recordId is local to the server. In case (b), the alternative identifier might be, for instance, a URL. It might self-identify its type, but in any case its type may optionally be indicated by the element typeOfIdentifier.

Elements *database* and *recordId* occur together and identify a database and record.

Alternatively, element *alternativeIdentifier* occurs if elements database and recordId do not, and it identifies a database and record.

Element *typeOfIdentifier* indicates the type of identifier, for example a URL or URN, for element alternativeIdentifer.

4.2.11 BriefTextDescription

Datatype *BriefTextDescription* has type InternationalString.

4.3 Schematic Representation of Descriptive Record

```
typeOfDescriptiveRecord
briefDescription
collectionInfo
        collectionName
        database (repeatable)   (server and db)
        enumeratedMember (repeatable)
            briefDescriptionOfMember
            pointer (database and recordId) XOR (alternativeIdentifier and typeOfIdentifier)
            whatPointerPointsTo
            fullyEnumerated (BOOLEAN)
            childrenKnowThisParent (BOOLEAN)
objectInfo
        typeOfObject
        categoryOfObject
        digitalObject  (repeatable)
            actualDO  XOR pointerToDO
                    (database and recordId) XOR (alternativeIdentifier and typeOfIdentifier)
            authoritative (BOOLEAN)
associatedDescription
        briefDescriptionOfAD
        categoryOfAD
        description (repeatable)
            actualAD XOR pointerToAD
                    (database and recordId) XOR (alternativeIdentifier and typeOfIdentifier)
            authoritative (BOOLEAN)
relatedCollection (repeatable)
        collection  (collectionName, server, and database)
        relationship  (repeatable)
        relativeLevel
        descriptionOfRelationship
```

4.4 Tag Types

For this schema a GRS-1 record will use the following tagTypes:

TagType	Usage
1	Elements from tagSet-M defined in Z39.50-1995, Appendix TAG, TAG.2.1. A target may include an element from tagSet-M at its discretion, and an origin may ignore it.
2	Elements from tagSet-G defined in Z39.50-1995, Appendix TAG, TAG.2.2. A target may include an element from tagSet-G at its discretion, and an origin may ignore it.
3	Reserved for tags locally defined by a target. This profile does not define the usage of tagType 3, but companion profiles may do so.
4	The tagSet defined in 4.5.

4.5 Descriptive Record TagSet

This section defines a tagSet for the Abstract Record Structure defined in 4.1. The object identifier for this tagSet is {iso (1) member-body (2) US (840) ANSI-standard-Z39.50 (10003) tagSet (14) digital collections (5)}

Tag	Element	DataType
1	typeOfDescriptiveRecord	INTEGER
2	briefDescription	BriefTextDescription
3	collectionInfo	CollectionInfo
4	objectInfo	ObjectInfo
5	associatedDescription	AssociatedDescription
6	relatedCollection	relatedCollection
7	collectionName	InternationalString
8	database	ServerAndDb
9	enumeratedMember	EnumeratedMember
10	fullyEnumerated	BOOLEAN
11	childrenKnowThisParent	BOOLEAN
12	typeOfObject	INTEGER
13	categoryOfObject	InternationalString
14	digitalObject	DigitalObject
15	briefDescriptionOfAD	BriefTextDescription
16	categoryOfAD	InternationalString
17	description	Description
18	collection	CollectionAndDb
19	relationship	INTEGER
20	relativeLevel	INTEGER
21	descriptionOfRelationship	InternationalString
22	collectionName	InternationalString
23	database	ServerAndDb
24	server	InternationalString
25	db	InternationalString
26	briefDescriptionOfMember	BriefTextDescription
27	pointer	RecordPointer
28	whatPointerPointsTo	INTEGER
29	actualDO	variant dependent; default is OCTET STRING
30	briefDescriptionOfDOVariant	BriefTextDescription
31	pointerToDO	RecordPointer
32	authoritative	BOOLEAN
33	briefDescriptionOfADVariant	BriefTextDescription
34	actualAD	variant dependent; default is InternationalString
35	pointerToAD	RecordPointer
36	database	ServerAndDb
37	recordId	OCTET STRING
38	alternativeIdentifier	OCTET STRING
39	typeOfIdentifier	OBJECT IDENTIFIER or InternationalString

4.6 Element Set Names Defined for this Schema

The following element set names are defined for this schema:

- 'b' (brief)
- 'navigation'
- 'full'

4.6.1 Brief Record

The brief record is defined as follows:

Element	Occurrence in Brief Record
typeOfDescriptive Record	Always included.
briefDescription	Included if available.
collectionInfo	(When typeOfDescriptiveRecord is 1) include the element collectionName only.
objectInfo	(When typeOfDescriptiveRecord is 2) include elements typeOfObject and categoryOfObject. Also, optionally, digitalObject: include only actualDO (of DigitalObject), if it is available, and include only the variant information (no content).
associated Description	Optional. Some or all occurrences may be included at target discretion. Include the elements briefDescriptionOfAD and categoryOfAD only.
relatedCollection	Excluded.

Companion profiles may specify additional elements that are to be included in a brief record, including elements defined by tagSets other than the tagSet used in this profile.

4.6.2 Navigation Record

The Navigation record is defined as follows:

Element	Occurrence in a navigation record
typeOfDescriptiveRecord	Always included.
briefDescription	Excluded.
collectionInfo	Occurs if available. All available elements of CollectionInfo occur. All available elements and all occurrences of EnumeratedMember (within CollectionInfo) occur.
objectInfo	Excluded.
associatedDescription	Excluded.
relatedCollection	All occurrences included.

4.6.3 Full Record

In a full record, all available elements for all available occurrences should be included.

4.7 Collection-1 Attribute Set

This section defines attribute set collection-1, for searching databases with records whose structure is as defined by the Descriptive Record schema. The object identifier for this attribute set is {iso (1) member-body (2) US (840) ANSI-standard-Z39.50 (10003) attributes (3) digital collections (7)}.

The Collection-1 attribute set defines the two attribute types: Use (type 1) and Relation (type 2). Semantics for combining attributes from this set with attributes from a different set, within a single operand, are not provided by this attribute set definition.

4.7.1 Collection-1 Use Attributes

The following Collection-1 Use attributes are defined:

Value **Use**

1 Record-id

2 Record-type

This attribute is used to distinguish the types of records of interest, as follows:

Term	**Meaning**
0	record is a Descriptive Record
1	record is a Collection Descriptive Record
2	record is a Collection Descriptive Record for a context collection
3	record is an Object Descriptive Record
4	record is a Digital Object

3 Object-type

This attribute is used to distinguish types of objects, as follows:

Term	**Meaning**
1	object is a Digital Object
2	object is a Digital Object for which there is a Physical Object
3	object is a Digital Object for which there no Physical Object
4	object is a Physical Object

This attribute may be combined with Use attribute Record-type, for example, to search for Object Descriptor Records describing Physical Objects only.

4 Parent-collection

This attribute is used to search for Descriptive Records that list the collection (specified by the term) as a parent.

Note: There is no requirement in this profile that a collection or object know all of its parents, but when it does not, full functionality might not be achieved.

This attribute may be combined with Use attribute Record-type to restrict the search to Collection Descriptive Records or Object Descriptive Records.

5 Superior-collection

This attribute is similar to Parent-collection, used to search for subordinate, but not necessarily immediately subordinate, Descriptive Records.

6 Related-collection

Used to search for Descriptive Records for objects or collections that list the collection (indicated by the term) as a related-collection.

7 Collection-name

Used to search for the Collection Descriptive Record for the collection (indicated by the term).

4.7.2 Collection-1 Relation Attributes

The following Collection-1 Relation attributes are defined:

Value	**Relation**
1	equal
2	always-matches

5. Z39.50 Specific Profile Details

Section 6 below defines three levels of conformance. These are briefly summarized here:

Basic Conformance	-	Version 2; support required for the DR schema and the GRS-1 record syntax.
Basic V3 Conformance	-	Basic conformance, plus support required for version 3.
Enhanced Conformance	-	Basic V3 conformance, plus support required for selected additional features.

When a feature is stated to be mandatory, then unless otherwise specified, support is mandatory both for origin and target, and for any conformance level.

5.1 Protocol version

Both version 2 and version 3 of Z39.50 are addressed by this profile. For Basic V3 Conformance or Enhanced Conformance version 3 must be supported.

5.2 Services and Facilities

The Init, Search, and Present services are mandatory. The Segment service is optional for Basic V3 and Enhanced Conformance (and excluded for Basic Conformance). The Explain facility is optional; see 5.5.

5.3 Search

Support for the Collection-1 attribute set (see 4.7) is mandatory. The use and semantics of Type-1 query operands that combine attributes from the Collection-1 set with attributes from a different set is not within the scope of this profile. Companion profiles may specify such usage and semantics.

5.4 Retrieval

5.4.1 Present Parameters

Support for Present parameter CompSpec is mandatory for Enhanced Conformance; it is optional for Basic V3 Conformance and excluded for Basic Conformance.

CompSpec is defined in Z39.50 as follows:

```
CompSpec ::= SEQUENCE{
   selectAlternativeSyntax     [1] IMPLICIT BOOLEAN,
   generic                     [2] IMPLICIT Specification OPTIONAL,
   dbspecific                  [3] IMPLICIT SEQUENCE OF SEQUENCE{
                                          db      [1] DatabaseName,
                                          spec    [2] IMPLICIT Specification} OPTIONAL,
   recordSyntax                [4] IMPLICIT SEQUENCE OF OBJECT IDENTIFIER
                                                OPTIONAL
      }
Specification ::= SEQUENCE{
   schema              [1] IMPLICIT OBJECT IDENTIFIER OPTIONAL,
   elementSpec         [2] CHOICE{
                            elementSetName    [1] IMPLICIT InternationalString,
                            externalEspec     [2] IMPLICIT EXTERNAL} OPTIONAL}
```

When the CompSpec parameter is used for the purpose of retrieving a Descriptive Record, it should be used as follows:

- A value of 'false' for 'selectAlternativeSyntax' should be accepted by the target.
- 'generic' should occur.
- 'dbSpecific' should not occur.
- 'recordSyntax', which is a SEQUENCE OF, should be the single syntax, GRS-1.
- 'schema' should be the Descriptive Record schema defined in section 4.
- For 'elementSpec', when 'elementSetName' is chosen it should be 'b', 'navigation', or 'f' (see 4.6).

5.4.2 Segmentation

Support for level-2 segmentation is optional for Basic V3 and Enhanced Conformance (and excluded for Basic Conformance). Support for level-1 segmentation (without level-2 segmentation) is outside the scope of this profile.

5.4.3 Element Specification

Support for element specification eSpec-1 is mandatory for Enhanced V3 Conformance. See 5.4.10.

5.4.4 Variant Set

Support for variant set variant-1 is mandatory for Enhanced V3 Conformance. A variant specification may occur:

- as a variant request in an element specification using eSpec-1, and as such the usage of variant-1 is detailed in 5.4.10;
- as an appliedVariant in a GRS-1 record, and as such the usage of variant-1 is detailed in 5.4.11; or
- as a supportedVariant in a GRS-1 record, and as such the usage of variant-1 is detailed in 5.4.11.

5.4.5 Fragmentation

Support for fragmentation is mandatory for Enhanced V3 Conformance. See 5.4.10.

The term *fragmentation*, as used in this profile, refers to the use of Z39.50 variants to retrieve a fragment of an element (see 5.4.10). A fragment is a logically contiguous piece of the element not necessarily distinguished as a subelement (i.e. if the entire element were to be retrieved, considering that element as a string of bits, a fragment is any substring of that string). Criteria for defining fragments are determined by the target. For example the target could define fragments based on number of bytes, or could define a page or screen to be a fragment.

For the purpose of this profile, fragmentation applies to either an Associated Description or Digital Object, and is used to retrieve either the "first" or "next" fragment.

5.4.6 Record Syntax

Support for GRS-1 is mandatory. See 5.4.11.

5.4.7 Descriptive Record Schema

Support for the Descriptive Record schema (section 4) is mandatory.

5.4.8 Brief and Navigation Descriptive Records

Support for element set names 'b' and 'navigation' (4.6) are mandatory.

5.4.9 Digital Objects and their Variants

A server providing access to different versions, instances, or representations of a given Digital Object may represent these as:

- distinct Digital Objects, using separate instances of ObjectInfo;
- a single logical Digital Object, using a single instance of ObjectInfo with multiple occurrences of digitalObject; or
- a single logical Digital Object, using a single instance of ObjectInfo and a single instance of digitalObject, with Z39.50 variant information provided for element actualDO.

The server may also use a combination of these. For example, the server might provide several variants of a Digital Object (using Z39.50 variant capabilities) and also provide a pointer to another representation of the Digital Object (perhaps on a different server). For the latter representation, the server would have to provide the pointer in a separate instance of digitalObject.

In any case, whenever a server provides a Digital Object via element actualDO (rather than pointer), whether only a single variant or multiple variants are provided, the server should use the Z39.50 variant capabilities including variant-1 and (for Enhanced V3 conformance) should support a request to list the available variants, including body part type. In addition a server may choose to indicate a level of *integrity* for a variant. (The variant may be a derived object, created on demand, and as such may be a conversion or reduction of the genuine object. See 5.4.11, ElementMetadata, supported variants MetaData -- integrity.)

5.4.10 Element Specification eSpec-1

For Enhanced V3 Conformance, usage of element specification eSpec-1 is defined as follows:

In the main structure, the parameter 'elements' must be supported, and none of the other parameters should be included (elementSetNames, defaultVariantSetId, defaultVariantRequest, or defaultTagType).

ElementRequest may select 'simpleElement' or 'compositeElement':

- 'simpleElement' must be supported, and within SimpleElement the variantRequest must be supported. A variant request will use the variant-1 definition and consist of one or more of:
  - a variantId;
  - Piece, where Piece is 'start' or 'next' of 'what-fragment wanted'.

  Within TagPath, specificTag will always include tagType and tagValue. 'occurrence' is mandatory, for retrieval of the following repeatable elements:
  - associatedDescription in the main structure of the descriptive record,
  - enumeratedMember of CollectionInfo,
  - digitalObject of ObjectInfo, and
  - description of AssociatedDescription.

  Within Occurrences, 'values' should be supported, and within 'values', 'start' and 'howMany' should be supported. This is useful when the origin wishes to scroll through enumerated Associated Descriptions or collection members, or multiple versions of a given Associated Description or Digital Object.
- Support for 'compositeElement' is optional. Companion profiles may mandate its support, for use with Associated Descriptions or Digital Objects which, though opaque to this profile, might not be opaque to a companion profile. For example, an origin might request specific elements of an Associated Description to be packaged as a single SGML element within a GRS-1 transfer record.

5.4.11 GRS-1

Usage of record syntax GRS-1 within this profile is defined as follows: In the GRS-1 main structure, the following parameters must be supported: tagType, tagValue, tagOccurrence, and content. In addition, for Enhanced V3 Conformance, metaData and appliedVariant must be supported.

For ElementData:

octets	Support mandatory.
numeric	Support mandatory.
date	Optional.
ext	Optional.
string	Support mandatory.
trueOrFalse	Support mandatory.
oid	Optional.
intUnit	Optional.
elementNotThere	Support mandatory for origin, optional for target.
elementEmpty	Support mandatory for origin, optional for target.
noDataRequested	Support mandatory for Enhanced V3 Conformance.
diagnostic	Support mandatory for origin, optional for target.
subtree	Support mandatory.

For ElementMetaData:

Series order	Optional. (Useful for enumerated collection members or Associated Descriptions when they are arranged in a meaningful order that the target wishes to convey.)	
usage right	Support mandatory for origin, optional for target. See 5.4.14.	
hits	Optional. (Not within the scope of this profile, however, companion profiles that model the content of Digital Objects might specify its use.)	
display name	Optional.	
supported variants	Support mandatory for Enhanced V3 Conformance.	
	variantId	Always supplied.
	bodyPartType	Always supplied.
	formatting	Optional.
	language	Optional (if applicable; see 5.4.12).
	character set	Optional (if applicable).
	piece	Not supplied.
	metaData -- cost	Optional.
	metaData -- size	Optional.
	metaData -- integrity	Optional. If supplied, zero represents highest integrity; higher values indicate progressively less integrity.
	metadata -- (all other)	Not supplied.
	(other)	Not supplied.
message	Support mandatory for origin.	
element descriptor	Optional. (Not within scope of this profile; however companion profiles might specify its use, for example to include an SGML DTD.)	
surrogate for	Optional. See 5.4.13.	
surrogate element	Optional. See 5.4.13.	

For appliedVariant:

variantId	Not supplied.
bodyPartType	Always supplied.
language	Optional. See 5.4.12.
character set	Optional (if applicable).
piece	For Enhanced V3 Conformance, must be supplied if applicable: values 'start', 'middle', 'last', and (optionally) 'whole' of 'what fragment returned'. If not supplied, 'whole' is assumed.
metaData	Optional. (Not within scope of this profile. Companion profiles may specify additional metadata.)
highlighting	Optional. (Not within scope of this profile. Companion profiles may specify highlighting.)

5.4.12 Language

'Language' (type 4 class 1 of variant-1) is applicable for text elements only, when there is a predominant language.

For Digital Objects, 'language' is not within the scope of this profile, however, companion profiles may specify its use.

For elements of type BriefTextDescription, 'language', within an applied variant, is the prescribed mechanism for a target to indicate the language of the description.

For an Associated Description, 'language', within a supported or an applied variant, is the prescribed mechanism for a target to indicate the language.

5.4.13 Surrogates

When a Digital Object is considered to be a surrogate for another Digital Object, and both objects are members of the same collection and thus are both represented as enumerated members within the same Collection Descriptive Record, the surrogate relationships may be expressed via 'surrogate for' and 'surrogate element' of GRS-1. (Note, however, this profile does not define what is meant by a surrogate relationship).

The enumerated member for a surrogate object may indicate via a tagPath the enumerated member representing the Digital Object for which it is a surrogate, and vice versa. The two elements will share the same tagType and tagValue, and are distinguished by tagOccurrence.

5.4.14 Usage Right

Via the 'usage right' parameter of ElementMetaData of GRS (see 5.4.11) a target may either (a) indicate whether a Digital Object is re-distributable or restricted, or (b) supply a license pointer.

When a client product claims to support this parameter, the claim means that the product is implemented with the intention that whenever the client receives what it perceives to be a usage right statement or pointer, it will make a good-faith effort to inform the user. It does not mean that it will follow the pointer, attempt to enforce the usage right statement, or obtain acknowledgement or concurrence from the user.

5.5 Explain

The use of the Z39.50 Explain facility is optional for this profile. Targets that support Explain may wish to distinguish one or more databases as those which provide access to digital collections (i.e. databases which include Descriptive Records). It is recommended that targets assign to those databases the searchable keyword 'collection-descriptive-records', searchable via the exp-1 Use attribute Keyword.

A Target using this feature may apply whatever criteria it deems appropriate in selecting databases to assign this keyword. For example a target may select only databases that include Collection Descriptive Records (as opposed to Object Descriptive Records), or only databases that include Collection Descriptive Records for context collections.

6. Conformance

An implementation claiming conformance to this profile shall indicate:

- that it conforms as an origin or target (or both); and
- that it conforms at one of the following levels:
  - Basic Conformance,
  - Basic V3 Conformance, or
  - Enhanced V3 Conformance.

6.1 Basic Conformance

An implementation claiming conformance at the Basic Conformance level must support version 2 of Z39.50-1995 (see 4.4 of that standard for specific conformance requirements), the DR schema specified in section 4, and the GRS-1 record syntax.

6.2 Basic V3 Conformance

An implementation claiming conformance at the Basic V3 Conformance level must conform at the Basic Conformance level (see 6.1) and in addition must support version 3 of Z39.50-1995.

6.3 Enhanced V3 Conformance

An implementation claiming conformance at the Enhanced V3 Conformance level must conform at the Basic V3 Conformance level (see 6.2) and in addition must support the following features:

- Present parameter CompSpec See 5.4.1.
- Element specification eSpec. See 5.4.3.
- Variant set variant-1. See 5.4.4.
- Fragmentation. See 5.4.5 (and 5.4.9).
- Metadata and appliedVariant of GRS-1. See 5.4.11.