

***Z39.50***

***Advanced Retrieval Facilities***

***ZIG Tutorial***  
***August 19, 1997***

***Les Wibberley***  
***Chemical Abstracts Service***  
***les.wibb@cas.org***

## Topic Outline

- **Present Service**
  - Model
  - Terminology
- **Element Selection**
  - Element Set Names
  - Composition Specification
  - Element Specification
  - Schema
- **Variants**
- **Record Syntaxes**
- **Complex Present Example**

# Z39.50 Core Protocol

CLIENT

SERVER

INITIALIZE REQUEST .....>>>>

<<<<.....INITIALIZE RESPONSE

- 
- 

SEARCH REQUEST .....>>>>

<<<<.....SEARCH RESPONSE

- 
- 

PRESENT REQUEST .....>>>>

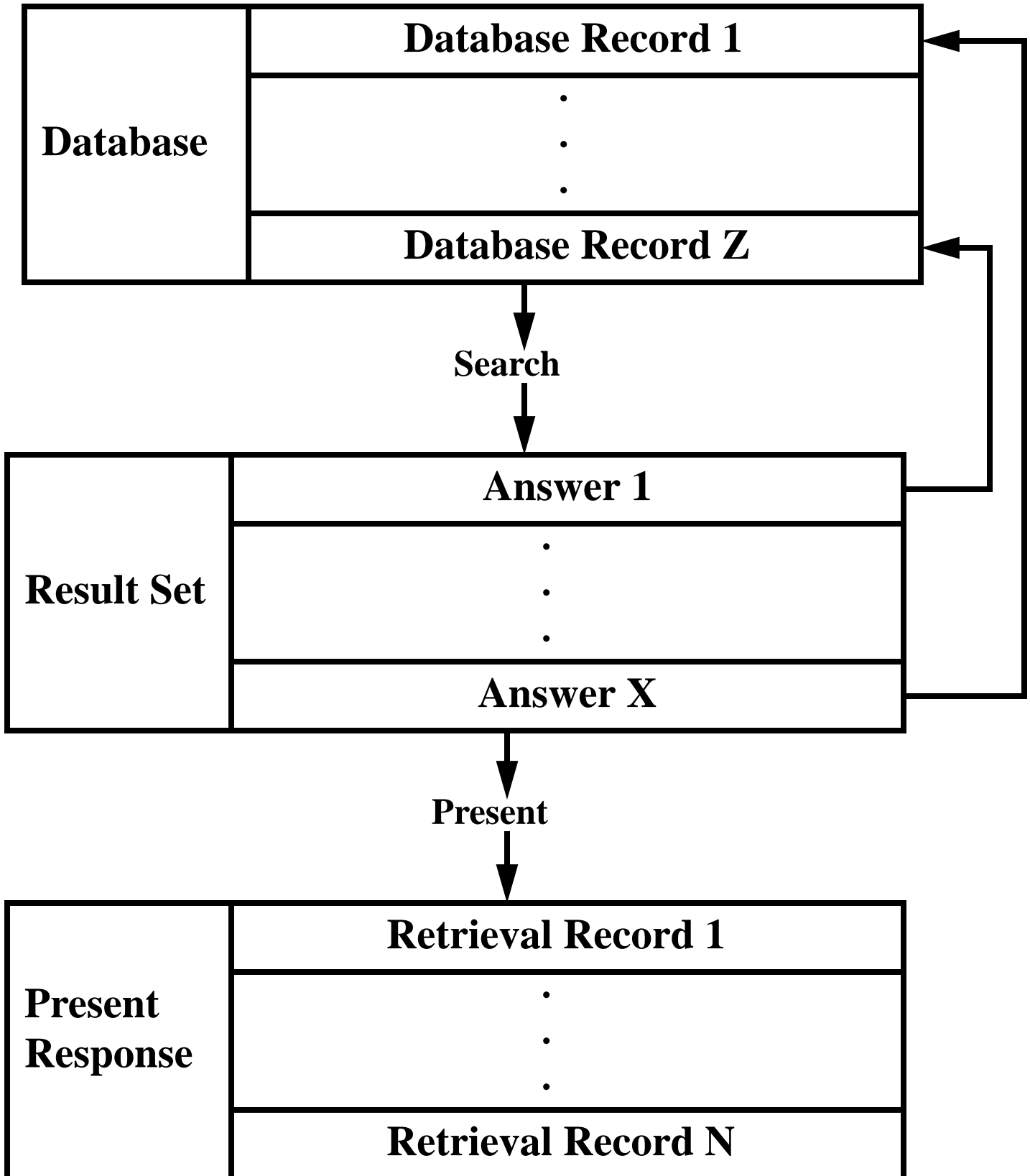
<<<<.....PRESENT RESPONSE

- 
- 

CLOSE REQUEST .....>>>>

<<<<.....CLOSE RESPONSE

# Search and Retrieval Model



## **Z39.50 Present Terminology**

**Database** - collection of Records

**Database Record** - collection of related information, treated as a unit

**Schema** - common understanding of the information contained in a Database

**Element** - retrieval unit of information defined by a Schema

**Variant** - specifies a particular form and representation of an Element

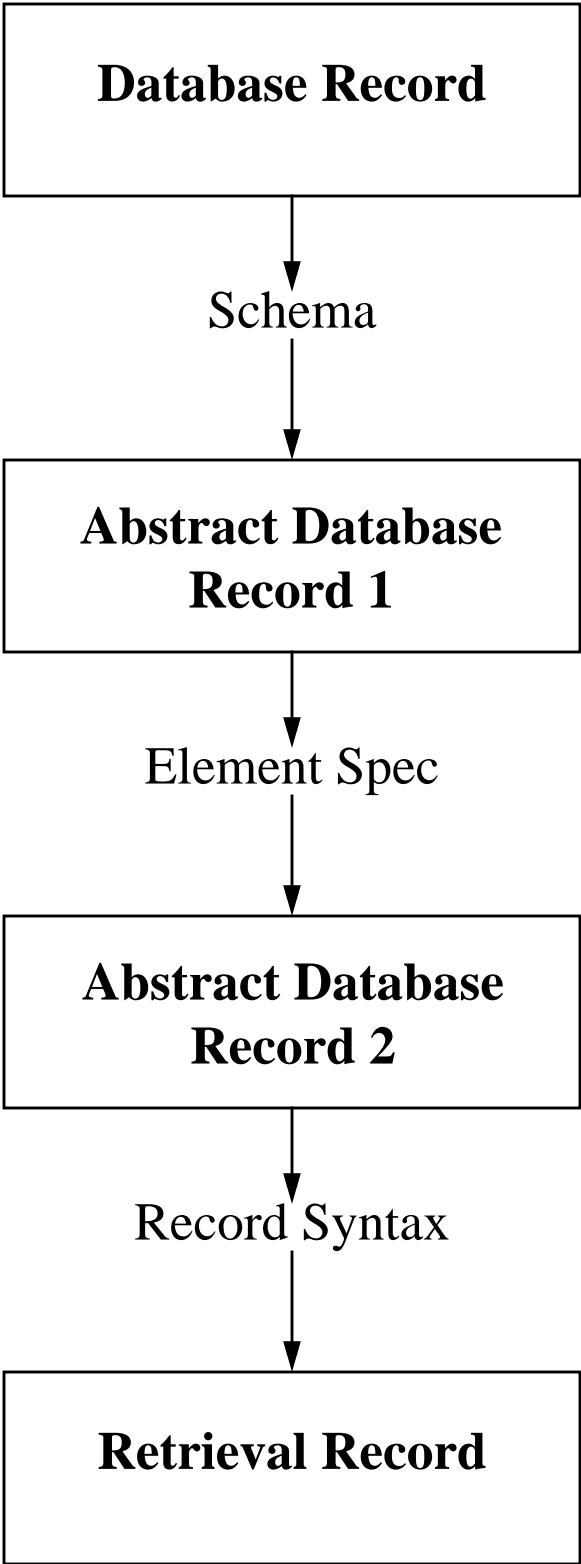
**Element Set Name** - primitive name for a collection of one or more Elements

**Element Specification** - specification for a set of Elements to be retrieved

**Record Syntax** - specification for the packaging of a Retrieval Record

**Composition Specification** - a retrieval specification, which may include a Schema, Element Specification, and Record Syntax

# Retrieval Model



# Present Request

**Z39.50 Present Request specifies:**

- **Result Set**
- **selection of Resultset Records**
- **selection of Elements**
- **form of Elements**
- **packaging of Elements**
- **segmentation of Present Response**

# Present Request

```
PresentRequest ::= SEQUENCE{
  referenceld          Referenceld OPTIONAL,
  resultSetId         ResultsetId,
  resultSetStartPoint [30] IMPLICIT INTEGER,
  numberOfRecordsRequested [29] IMPLICIT INTEGER,
  additionalRanges    [212] IMPLICIT SEQUENCE OF Range
                      OPTIONAL,
  recordComposition  CHOICE{
                      simple [19] ElementSetNames,
                      complex [209] IMPLICIT CompSpec}
                      OPTIONAL,
  preferredRecordSyntax [104] IMPLICIT OBJECT IDENTIFIER
                      OPTIONAL,
  maxSegmentCount     [204] IMPLICIT INTEGER OPTIONAL,
  maxRecordSize       [206] IMPLICIT INTEGER OPTIONAL,
  maxSegmentSize      [207] IMPLICIT INTEGER OPTIONAL,
  otherInfo           OtherInformation OPTIONAL
}
```



## Topic Outline

- **Present Service**
  - Model
  - Terminology
- • **Element Selection**
  - Element Set Names
  - Composition Specification
  - Element Specification
  - Schema
- **Variants**
- **Record Syntaxes**
- **Complex Present Example**

# **Present Element Selection**

- **Element Set Names**
- **Composition Specification**
- **Element Specification**
- **Schema**
- **Element Selection Example**

## Element Set Names

**ElementSetNames ::= CHOICE {**  
    **genericElementSetName**           **[0] IMPLICIT InternationalString,**  
    **databaseSpecific**               **[1] IMPLICIT SEQUENCE OF SEQUENCE{**  
                                    **dbName     DatabaseName,**  
                                    **esn         ElementSetName}}**

**DatabaseName    ::=           [105] IMPLICIT InternationalString**  
**ElementSetName ::=           [103] IMPLICIT InternationalString**

# Element Set Name Examples

- **Standard Z39.50 ESNs**

**F (Full) - all available elements**

**B (Brief) - target defined subset of elements**

- **Primitive ESNs**

**AU - Author only**

**AUTI - Author and Title**

**BIB - all bibliographic Elements**

- **List-style ESNs**

**AU, TI, AB - Author, Title and Abstract**

**62,4,1003 - Element Numbers**

# **Composition Specification**

**Generalized retrieval specification including:**

- **Element Specification - selects elements**
  - **Element Set Name**
  - **External Structure (Espec-1)**
- **Schema - supports element identification**
- **Record Syntax - specifies element packaging**

# Composition Specification

```
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
}
```

# **Element Specification Format Espec-1**

- **Generalized element specification**
- **selects elements for retrieval**
- **selects forms of elements: Variants**
- **specifies packaging of elements**
- **requests element meta-data**
- **supports discovery of elements**
- **supports element segmentation**
- **handles complex data structures**

# Element Specification Format Espec-1

**Espec-1 ::= SEQUENCE{**  
    **elementSetNames**           **[1] IMPLICIT SEQUENCE OF**  
                                    **InternationalString OPTIONAL,**  
    **defaultVariantSetId**       **[2] IMPLICIT OBJECT IDENTIFIER**  
                                    **OPTIONAL,**  
    **defaultVariantRequest**   **[3] IMPLICIT Variant OPTIONAL,**  
    **defaultTagType**           **[4] IMPLICIT INTEGER OPTIONAL,**  
    **elements**                 **[5] IMPLICIT SEQUENCE OF**  
                                    **ElementRequest OPTIONAL}**

**ElementRequest ::= CHOICE{**  
    **simpleElement**           **[1] IMPLICIT SimpleElement,**  
    **compositeElement**       **[2] IMPLICIT SEQUENCE{**  
        **elementList**       **[1] CHOICE{**  
            **primitives**       **[1] IMPLICIT SEQUENCE OF**  
                                    **InternationalString,**  
            **specs**           **[2] IMPLICIT SEQUENCE OF**  
                                    **SimpleElement},**  
        **deliveryTag**       **[2] IMPLICIT TagPath,**  
        **variantRequest**     **[3] IMPLICIT Variant OPTIONAL}}**



## Elements within Espec-1

```
SimpleElement ::= SEQUENCE{
    path          [1] IMPLICIT TagPath,
    variantRequest [2] IMPLICIT Variant OPTIONAL}
```

```
TagPath ::= SEQUENCE OF CHOICE{
    specificTag [1] IMPLICIT SEQUENCE{
        tagType      [1] IMPLICIT INTEGER OPTIONAL,
        tagValue     [2] StringOrNumeric,
        occurrence   [3] Occurrences OPTIONAL
    },
    wildThing      [2] Occurrences,
    wildPath       [3] IMPLICIT NULL
}
```

```
Occurrences ::= CHOICE{
    all          [1] IMPLICIT NULL,
    last        [2] IMPLICIT NULL,
    values      [3] IMPLICIT SEQUENCE{
        start      [1] IMPLICIT INTEGER,
        howMany    [2] IMPLICIT INTEGER OPTIONAL
    }
}
```

## Element Retrieval Terminology

Element - a retrievable unit of information

TagSet - a set of Elements, and their assigned Tags

Tag - identifier of an Element, consisting of a TagType and a TagValue

TagType - short-hand identifier for a TagSet

TagValue - identifier of an Element within a TagSet

TagSetId - a unique Object Identifier for a Tag Set

TagPath - a sequence of tags representing the hierarchical path to an Element (or subtree of Elements)

“Base” TagSets:

- TagSet-M defines Metadata Elements
- TagSet-G defines Generic Elements

# TagSet-M

## Meta-data Elements

TagSetId: 1.2.840.10003.14.1

<u>Element</u>	<u>tag</u>	<u>ASN.1 datatype</u>
schemaidentifier	1	OBJECT IDENTIFIER
elementsOrdered	2	BOOLEAN
elementOrdering	3	INTEGER
defaultTagType	4	INTEGER
defaultVariantSetId	5	OBJECT IDENTIFIER
defaultVariantSpec	6	VariantSpec
processingInstructions	7	InternationalString
recordUsage	8	INTEGER
restriction	9	InternationalString
rank	10	INTEGER
userMessage	11	InternationalString
url	12	InternationalString
record	13	structured
local control number	14	InternationalString
creation date	15	GeneralizedTime
dateOfLastModification	16	GeneralizedTime
dateOfLastReview	17	GeneralizedTime
score	18	INTEGER
wellKnown	19	InternationalString
recordWrapper	20	structured
defaultTagSetId	21	OBJECT IDENTIFIER

# TagSet-STAS: Scientific & Technical Element Set

TagSetId: 1.2.840.10003.14.3

<u>Element</u>	<u>tag</u>	<u>ASN.1 datatype</u>
Boiling Point	2494	structured
Chemical Name	2052	InternationalString
Molecular Formula	2085	InternationalString
Structure Diagram	2159	OCTET STRING
Patent Number	2033	InternationalString
Patent Application Date	2028	GeneralizedTime
Int. Patent Class.	2016	InternationalString
Abstract	62	InternationalString
Accession Number	2010	InternationalString
Author or Inventor	1003	InternationalString
Title	4	InternationalString

.	.	.
.	.	.
.	.	.

# Schema

- **Schema is an evolving concept within Z39.50 V3**
- **Provides common understanding of the information contained in a Database**
- **Defines the abstract structure of a database record**
- **Specifies retrievable Elements and their identifiers (Tags)**
- **Specifies TagSets supported and maps them to Tag-Types**
- **Supports Element selection in Present Request**
- **Supports Element identification in Present Response**

## Schema Example: “ChemResearch”

- Identifies Elements in Database “ChemResearch”
- TagSets supported:

<u>TagType</u>	<u>TagSet</u>	<u>TagSet OID</u>
1	TagSet-M	1.2.840.10003.14.1
2	TagSet-G	1.2.840.10003.14.2
3	TagSet-STAS	1.2.840.10003.14.3

- Abstract Record Structure definition
- “ChemResearch” Schema Object Id:  
1.2.840.10003.13.1000.6.1

This OID is composed of the following components:

1.2.840.10003	ISO object ID for ANSI Z39.50
1.2.840.10003.13	Object ID for Z39.50 Schemas
.1000	identifier for Z39.50 implementors
.6	implementor ID assigned to CAS
.1	“ChemResearch” Schema identifier

## Abstract Record Structure for Schema “ChemResearch”

<u>Tag Path (Type,Value)</u>	<u>Element</u>	<u>Repeatable?</u>	<u>Mandatory?</u>
(1,1)	schemaidentifier	no	no
(1,4)	defaultTagType	no	no
(1,5)	defaultVariantSetId	no	no
(1,8)	recordUsage	no	no
(1,9)	restriction	no	no
(1,11)	userMessage	no	no
(1,21)	defaultTagSetId	no	no
(3,62)	Abstract	yes	no
(3,2010)	Accession Number	no	yes
(3,1003)	Author or Inventor	no	no
(3,2494)	Boiling Point	yes	no
(3,2052)	Chemical Name	no	no
(3,2016)	Int. Patent Class.	no	no
(3,2085)	Molecular Formula	no	no
(3,2028)	Patent Application Date	no	no
(3,2033)	Patent Number	no	no
(3,2159)	Structure Diagram	no	no
(3,4)	Title	no	no

## Simple CompSpec Example

- Retrieve AUTHOR (1003), TITLE (4), and ALL ABSTRACTS (62)
- Prefer GRS-1 Record Syntax, but will accept SUTRS.

```
CompSpec {
  selectAlternativeSyntax FALSE, -- only GRS-1 or SUTRS
  generic
    Specification {
      schema 1.2.840.10003.13.1000.6.1, -- ChemResearch
      elementSpec {
        externalEspec, -- eSpec-1
      }
    }
  recordSyntax {
    1.2.840.10003.5.105, -- GRS-1 Record Syntax
    1.2.840.10003.5.101 -- SUTRS Record Syntax
  }
}
```



## Simple Espec-1 Example

- Retrieve AUTHOR (1003), TITLE (4), and ALL ABSTRACTS (62)

```
Espec-1 {
  elements {
    ElementRequest {
      SimpleElement {
        TagPath {
          tagType 3, -- STAS
          tagValue {
            numeric 1003} -- AUTHOR
          }},
    ElementRequest {
      SimpleElement {
        TagPath {
          tagType 3, -- STAS
          tagValue {
            string "TITLE"} -- TITLE
          }},
    ElementRequest {
      SimpleElement {
        TagPath {
          tagType 3, -- STAS
          tagValue {
            numeric 62}, -- ABSTRACT
        Occurrences ALL
      }
    }
  }
}
```

## Topic Outline

- **Present Service**
  - Model
  - Terminology
- **Element Selection**
  - Element Set Names
  - Composition Specification
  - Element Specification
  - Schema
- • **Variants**
- **Record Syntaxes**
- **Complex Present Example**

## Variants

- **Variants specify particular form, representation, and/or piece of an Element**
- **Variant is a new concept within Z39.50 V3**
- **Qualifies Element selection in Present Request**
- **Qualifies Element identification in Retrieval Records**
- **Variants specified by Class, Type, and Value**
- **A Variant Set is uniquely identified by an OID**
- **Variant-1 is a “standard” Variant Set**
- **Variant-1 defines “common” Variants used in retrieval**

## Variants ASN.1

```
Variant ::= SEQUENCE{
  globalVariantSetId [1] IMPLICIT OBJECT IDENTIFIER
    OPTIONAL,
  triples [2] IMPLICIT SEQUENCE OF SEQUENCE{
    variantSetId [0] IMPLICIT OBJECT IDENTIFIER
      OPTIONAL,
    class [1] IMPLICIT INTEGER,
    type [2] IMPLICIT INTEGER,
    value [3] CHOICE{
      INTEGER,
      InternationalString,
      OCTET STRING,
      OBJECT IDENTIFIER,
      BOOLEAN,
      NULL,
      unit [1] IMPLICIT Unit,
      valueAndUnit [2] IMPLICIT IntUnit
    }
  }
}
```

```
IntUnit ::= SEQUENCE{
  value [1] IMPLICIT INTEGER,
  unitUsed [2] IMPLICIT Unit}
```

```
Unit ::= SEQUENCE{
  unitSystem [1] InternationalString OPTIONAL, -- e.g. 'SI'
  unitType [2] StringOrNumeric OPTIONAL, -- e.g. 'mass'
  unit [3] StringOrNumeric OPTIONAL, -- e.g. 'kilograms'
  scaleFactor [4] IMPLICIT INTEGER OPTIONAL-- e.g. 9 means 10**9
}
```

# Variant Set Variant-1

<u>Class</u>	<u>Type</u>	<u>Example Value(s)</u>
1	= variant identifier	
2	= BodyPartType	
1	= ianaType/subType	“application/postscript”
2	= Z39.50Type/subType	“SGML/TEI”
	:	
3	= formatting/presentation	
1	= characters per line	72
2	= line length	100 characters
3	= lines per page	50
	:	
4	= Language/CharacterSet	
1	= language	Z39.53 language code
2	= registered character set	ISO CharacterSet Number
	:	
5	= Piece	
1	= what fragment wanted	2 = next
2	= what fragment returned	3 = last
3	= start	page 5
4	= end	page 10
	:	

## Variant Set Variant-1 Cont'd.

<u>Class</u>	<u>Type</u>	<u>Example Value(s)</u>
<b>6 = meta-data requested</b>		
	1 = cost	in dollars
	2 = size	in bytes
	3 = hits, variant-specific	NULL
	4 = hits, non-variant-specific	NULL
	5 = variant list	NULL
	:	
<b>7 = meta-data returned</b>		
	1 = cost	\$1.25
	2 = size	5 KBytes
	:	
<b>8 = Highlighting</b>		
	1 = prefix	OCTET STRING
	2 = postfix	OCTET STRING
	3 = server default	NULL
<b>9 = miscellaneous</b>		
	1 = NoData	NULL (variantRequest only)
	2 = Unit	Unit (variantRequest only)
	3 = Version	InternationalString

## Variant Example: Espec-1

Retrieve ABSTRACT (62) in Plaintext, English, 72 characters per line.

```
Espec-1 {
  defaultVariantRequest {
    globalVariantSetId 1.2.840.10003.12.1, -- Variant-1
    triples {
      {class 2 -- bodyPartType
        type 1 -- ianaType/subType
        value "text/plain"
      },
      {class 3 -- formatting
        type 1 -- characters per line
        value 72
      },
      {class 4 -- language/character set
        type 1 -- USMARC language code
        value "English"
      },
    },
  },
  elements {
    ElementRequest {
      SimpleElement {
        TagPath {
          tagType 3, -- STAS
          tagValue {
            numeric 62}, -- ABSTRACT
          }}}
      }
    }
  }
}
```

## Topic Outline

- **Present Service**
  - Model
  - Terminology
- **Element Selection**
  - Element Set Names
  - Composition Specification
  - Element Specification
  - Schema
- **Variants**
- • **Record Syntaxes**
- **Complex Present Example**



## Record Syntaxes

- **specify how to package retrieval data**
- **V2: specify via PreferredRecordSyntax**
- **V3: Composition Specification added**
- **MARC record syntax for bibliographic data**
- **SUTRS record syntax for text data**
- **Generic Record Syntax for mix of data types**
- **Custom Record Syntaxes for specialized data**

## **Generic Record Syntax: GRS-1**

- **general purpose Record Syntax**
- **used to deliver all kinds of data**
- **elements are identified using tags**
- **supports both data and meta-data**
- **supports hierarchical data structures**
- **identifies Element Variants**
- **supports any & all data representations**
- **GRS-1 identified by unique Object Id**

## Generic Record Syntax: GRS-1

**GenericRecord ::= SEQUENCE OF TaggedElement**

**TaggedElement ::= SEQUENCE {**  
    **tagType**           **[1] IMPLICIT INTEGER OPTIONAL,**  
    **tagValue**         **[2] StringOrNumeric,**  
    **tagOccurrence**   **[3] IMPLICIT INTEGER OPTIONAL,**  
    **content**           **[4] ElementData,**  
    **metaData**         **[5] IMPLICIT ElementMetaData**  
                          **OPTIONAL,**  
    **appliedVariant**   **[6] IMPLICIT Variant OPTIONAL**  
                          **}**

## Element Value within GRS-1

```
ElementData ::= CHOICE{
    octets          OCTET STRING,
    numeric         INTEGER,
    date           GeneralizedTime,
    ext            EXTERNAL,
    string         InternationalString,
    trueOrFalse    BOOLEAN,
    oid            OBJECT IDENTIFIER,
    intUnit        [1] IMPLICIT IntUnit,
    elementNotThere [2] IMPLICIT NULL,
    elementEmpty   [3] IMPLICIT NULL,
    noDataRequested [4] IMPLICIT NULL,
    diagnostic     [5] IMPLICIT EXTERNAL,
    subtree        [6] SEQUENCE OF TaggedElement
}
```

## Metadata within GRS-1

**ElementMetaData ::= SEQUENCE{**

<b>seriesOrder</b>	<b>[1]</b>	<b>IMPLICIT Order OPTIONAL,</b>
<b>usageRight</b>	<b>[2]</b>	<b>IMPLICIT Usage OPTIONAL,</b>
<b>hits</b>	<b>[3]</b>	<b>IMPLICIT SEQUENCE OF HitVector OPTIONAL,</b>
<b>displayName</b>	<b>[4]</b>	<b>IMPLICIT InternationalString OPTIONAL,</b>
<b>supportedVariants</b>	<b>[5]</b>	<b>IMPLICIT SEQUENCE OF Variant OPTIONAL,</b>
<b>message</b>	<b>[6]</b>	<b>IMPLICIT InternationalString OPTIONAL,</b>
<b>elementDescriptor</b>	<b>[7]</b>	<b>IMPLICIT OCTET STRING OPTIONAL,</b>
<b>surrogateFor</b>	<b>[8]</b>	<b>IMPLICIT TagPath OPTIONAL,</b>
<b>surrogateElement</b>	<b>[9]</b>	<b>IMPLICIT TagPath OPTIONAL,</b>
<b>other</b>	<b>[99]</b>	<b>IMPLICIT EXTERNAL OPTIONAL}</b>

## Metadata within GRS-1

**Order ::= SEQUENCE{**  
    **ascending**                   **[1] IMPLICIT BOOLEAN,**  
    **order**                       **[2] IMPLICIT INTEGER}**

**Usage ::= SEQUENCE{**  
    **type**                       **[1] IMPLICIT INTEGER{**  
                                  **redistributable (1),**  
                                  **restricted (2),**  
                                  **licensePointer (3)},**  
    **restriction**               **[2] IMPLICIT InternationalString**  
                                  **OPTIONAL}**

**HitVector ::= SEQUENCE{**  
    **satisfier**                   **Term OPTIONAL,**  
    **offsetIntoElement**       **[1] IMPLICIT IntUnit OPTIONAL,**  
    **length**                     **[2] IMPLICIT IntUnit OPTIONAL,**  
    **hitRank**                   **[3] IMPLICIT INTEGER OPTIONAL,**  
    **targetToken**               **[4] IMPLICIT OCTET STRING**  
                                  **OPTIONAL}**

## Simple GRS-1 Example

Retrieve AUTHOR (1003), TITLE (4), and ALL ABSTRACTS (62)

```
GenericRecord {
  TaggedElement {
    tagType 3,      -- STAS
    tagValue {
      numeric 1003}, -- Author
    content {
      string "Marshall T. Rose"}
  },
  TaggedElement {
    tagType 3,
    tagValue {
      numeric 4},  -- Title
    content {
      string "The Open Book"}
  },
  TaggedElement {
    tagType 3,
    tagValue {
      numeric 62}, -- Abstract
    content {
      elementNotThere}
  }
}
```

## Variant Example: GRS-1

Retrieve ABSTRACT (62) in Plaintext, English, 72 characters per line.

```
GenericRecord {
  TaggedElement {
    tagType 3,      -- STAS
    tagValue {
      numeric 62}, -- Abstract
    content {
      string "This is the plaintext,
              english language
              Abstract of Marshall
              Rose's Open Book..."}
    }
  appliedVariant {
    globalVariantSetId 1.2.840.10003.12.1,
    triples {
      {class 2,      -- bodyPartType
        type 1,      -- IANAtype/subType
        value "text/plain"}
    },
    {class 3,      -- formatting
      type 1,      -- characters per line
      value 72}
    },
    {class 4,      -- language/character set
      type 1,      -- Z39.53 language code
      value "eng" -- English}
    }
  }
}
```



## Topic Outline

- **Present Service**
  - Model
  - Terminology
- **Element Selection**
  - Element Set Names
  - Composition Specification
  - Element Specification
  - Schema
- **Variants**
- **Record Syntaxes**
- • **Complex Present Example**

## Complex Retrieval Example

1. Search for all chemical substances with:

- molecular formula = c18h23no3

AND

- boiling point between 220 - 222 degrees C

2. Retrieve 4 Elements from Answer 1:

- chemical name of the substance

- boiling point table for the substance

- molecular formula of the substance

- structure diagram for the substance

3. Discover available Variants of Structure Diagram Element

4. Retrieve Postscript Variant of Structure Diagram Element

# STAS Attribute/Element/Tag Set

ELEMENT	USE Attribute	Element Number	GRS-1 Tag Number
Patent Number	2033	2033	2033
Int. Patent Classification	2016	2016	2016
Patent Application Date	2028	2028	2028

•  
•  
•

Molecular Formula	2085	2085	2085
Boiling Point	2493	2494	2494
Chemical Name	2052	2052	2052
Structure Diagram	--	2159	2159

•  
•  
•

Abstract	62	62	62
Author or Inventor	1003	1003	1003
Title	4	4	4

## Example of Non-Bibliographic RPN Query

- **Query Type = 1 (Query)**
- **Attribute Set Id = STAS**
- **RPNStructure 1: Operand 1: AttributesPlusTerm**
  - **Term: c18h23no3**
  - **Use Attribute: molecular formula**
- **RPNStructure 2: RPN Structure**
  - **Operand 1: AttributesPlusTerm**
    - **Term:220**
    - **Use Attribute: boiling point**
    - **Relation Attribute: GE**
  - **Operand 2: AttributesPlusTerm**
    - **Term: 222**
    - **Use Attribute: boiling point**
    - **Relation Attribute: LE**
  - **Operator: AND**
- **Operator: AND**

# Complex Retrieval Example: RPN Query

```
rpn {  
  rpn1 {  
    rpn1 {  
      AttributeElement {  
        attributeType 1,          -- USE Attribute  
        attributeValue {  
          numeric 2085}},        -- Molecular Formula  
      term { general "c18h23no3" },  
    rpn2 {  
      AttributeElement {  
        attributeType 1,          -- USE Attribute  
        attributeValue {  
          numeric 2493}},        -- Boiling Point  
      AttributeElement {  
        attributeType 2,          -- Relation Attribute  
        attributeValue {  
          numeric 4 }},          -- Greater/Equal  
      term { general "220" },  
    op { and NULL } },  
  rpn2 {  
    op {  
      AttributeElement {  
        attributeType 1,          -- USE Attribute  
        attributeValue {  
          numeric 2493 }},        -- Boiling Point  
      AttributeElement {  
        attributeType 2,          -- Relation Attribute  
        attributeValue {  
          numeric 2 }},          -- Less/Equal  
      term { general "222" } },  
    },  
  op { and NULL }  
}
```

# Retrieve Answer Record 1: PresentRequest

Prefer GRS-1 Record Syntax, but will accept SUTRS.

```
PresentRequest {
  referenceld 103,
  resultSetId "ResultSet1",
  resultSetStartPoint 1,
  numberOfRecordsRequested 1,
  recordComposition {
    CompSpec {
      selectAlternativeFlag FALSE,
      generic
      Specification {
        schema 1.2.840.10003.13.1000.6.1,
          -- ChemResearch
        elementSpec {
          externalEspec -- Espec-1
        }
      }
    }
  }
}
```

# Retrieve Answer Record 1: Espec-1

Retrieve the Chemical Name (2052), Boiling Point (2494),  
Molecular Formula (2085), and Structure Diagram (2159).

```
Espec-1 {
  elements {
    ElementRequest {
      SimpleElement {
        TagPath {
          tagType 3, -- STAS
          tagValue {
            numeric 2052} -- Chemical Name
          }},
    ElementRequest {
      SimpleElement {
        TagPath {
          tagType 3, -- STAS
          tagValue {
            numeric 2494} -- Boiling Point
          }},
    ElementRequest {
      SimpleElement {
        TagPath {
          tagType 3, -- STAS
          tagValue {
            numeric 2085} -- Molecular Formula
          }},
    ElementRequest {
      SimpleElement {
        TagPath {
          tagType 3, -- STAS
          tagValue {
            numeric 2159} -- Structure Diagram
          }},
  }
}
```

# Deliver Answer Record 1

Deliver the Chemical Name (2052), Boiling Point (2494), Molecular Formula (2085), and Structure Diagram (2159).

```
GenericRecord {
  TaggedElement {
    tagType 3,      -- STAS
    tagValue {
      numeric 2052}, -- Chemical Name
    content {
      string "1,3,4,5,6,7-Hexadydro..."
    },
  }
  TaggedElement {
    tagType 3,
    tagValue {
      numeric 2494}, -- Boiling Point
    content {
      string "Boiling Point Table..."
    },
  }
  TaggedElement {
    tagType 3,
    tagValue {
      numeric 2085}, -- Molecular Formula
    content {
      string "C18H23NO3"},
    },
  }
  TaggedElement {
    tagType 3,
    tagValue {
      numeric 2159}, -- Structure Diagram
    content {
      octets "Structure Diagram..."
    }
  }
}
```



# Generic Record Syntax Example

- tagged elements

- Element1: Chemical Name

tag: 2052

content: 1,3,4,5,6,7-Hexahydro-1-<3,4-dimethoxy-phenethyl>-2H-1-pyrindin-2-on

variant: ascii text

- Element2: Boiling Point

tag: 2494

content:

**Boiling Point:**

Value (BP) (deg C)	Press. (.P) (bar)	Ref
220 - 230	0.000999	1
220 - 230	0.000999	2
220 - 230	0.000999	3

variant: ascii table

- Element3: Molecular Formula

tag: 2085

content: C18 H23 N O3

variant: ascii

- Element4: Structure Diagram

tag: 2159

content: (see next page)

variant: ascii

## Generic Record Syntax Example (cont'd).

- **Element4: Structure Diagram**

tag: 2159

variant: ascii

value:

## Discover available Variants

Discover what Variants of the Structure Diagram Element (2159) are available

```
Espec-1 {
  defaultVariantRequest {
    globalVariantSetId 1.2.840.10003.12.1, -- Variant-1
    triples {
      {class 6,      -- meta-data requested
       type 5,      -- variant list
       value NULL -- return variant info
      },
      {class 9,      -- miscellaneous
       type 1,      -- NoData
       value NULL -- return no data
      },
    },
  },
  elements {
    ElementRequest {
      SimpleElement {
        TagPath {
          tagType 3, -- STAS
          tagValue {
            numeric 2159} -- Structure Diagram
          }}}
      }
    }
  }
}
```

## Deliver list of available Variants

Deliver list of Variants available for the Structure Diagram Element (2159).

```
GenericRecord {
  TaggedElement {
    tagType 3,
    tagValue {
      numeric 2159}, -- Structure Diagram
    content {
      noDataRequested} -- no data returned
    }
  metaData {
    supportedVariants {
      globalVariantSetId 1.2.840.10003.12.1,
      triples {
        {class 2, -- bodyPartType
          type 1, -- IANAtype/subType
          value "text/plain"},
        triples {
          {class 2, -- bodyPartType
            type 1, -- IANAtype/subType
            value "application/postscript"}
          }
        }}
      }
    }
  }
```

## Request the Postscript Variant

Retrieve the Postscript Variant of the Structure Diagram (2159).

```
Espec-1 {
  defaultVariantRequest {
    globalVariantSetId 1.2.840.10003.12.1, -- Variant-1
    triples {
      {class 2, -- bodyPartType
       type 1, -- IANAtype/subType
       value "application/postscript"
      },
    },
  },
  elements {
    ElementRequest {
      SimpleElement {
        TagPath {
          tagType 3, -- STAS
          tagValue {
            numeric 2159} -- Structure Diagram
          }}}
      }
    }
  }
}
```

## Deliver the Postscript Variant

Deliver the Postscript Variant of the Structure Diagram (2159).

```
GenericRecord {
  TaggedElement {
    tagType 3,
    tagValue {
      numeric 2159}, -- Structure Diagram
    content {
      octets "Postscript Structure Diagram..."}
  }
  appliedVariant {
    globalVariantSetId 1.2.840.10003.12.1,
    triples {
      {class 2, -- bodyPartType
        type 1, -- IANAtype/subType
        value "application/postscript"}
    }
  }
}
```

# Variant Example

- **Element1: Structure Diagram**

tag: 2159

variant: postscript

value: