

A Z39.50 Introduction

Jacob Hallén
LIBRIS Department
The Royal Library
Sweden

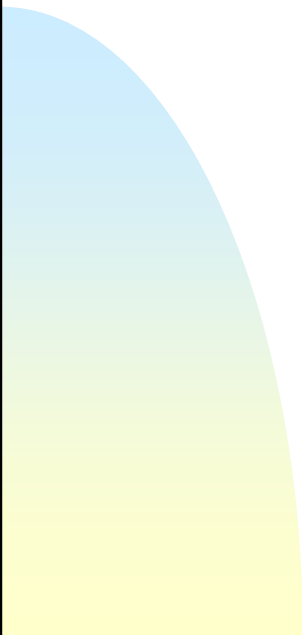
Foreword

- Z39.50 is the rather cryptic code for a standard which is playing an increasingly important role for information distribution, especially in the library world. This standard is rather hard to penetrate. We will try to get you across the first hurdle and make you familiar with some of the most important terminology.



Goals

- Enough knowledge to
 - ◆ have an intelligent conversation with a vendor/programmer
 - ◆ understand the procedure for search and retrieval through Z39.50
- Some knowledge of
 - ◆ different architectures for deployment
 - ◆ profiles and areas of use
 - ◆ the protocol at a cursory level



Overview

- Introduction
- What is Z39.50?
- How is Z39.50 used?
- A small market overview
- How does Z39.50 work?
- Information sources



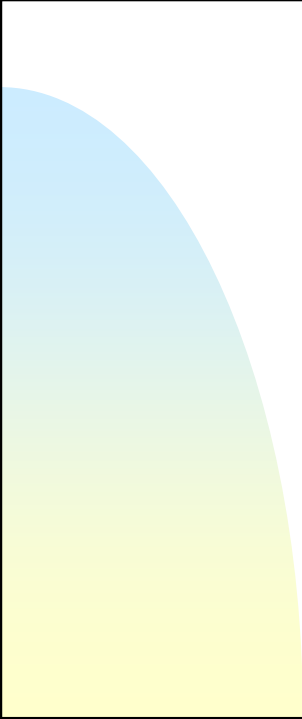
What is Z39.50?

- A standard established by NISO (National Information Standards Organization)
- Accepted by ISO (International Standards Organization) as ISO 23950
- Maintained by: Ray Denenberg, Library of Congress



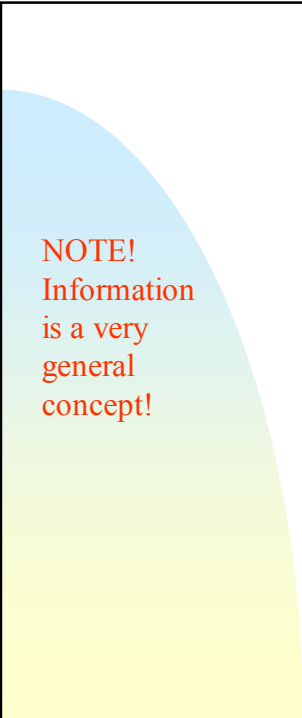
ZIG - Z39.50 Implementors group

- A group of people who develop or run Z39.50 systems
- Discusses amendments, defects and clarification
- Creates implementors agreements
- Meets every 5 months (North America, Europe, Washington DC)
- Works according to the consensus principle



History

- Roots in the WAIS protocol
 - ◆ Simple S/R-protocol from the mid 80-ies
- Supplants ISO 10162/10163 Search & Retrieve (1993)
- Z39.50 - 1988
- Z39.50 - 1992 (version 2)
- Z39.50 - 1995 (version 3)




NOTE!
Information is a very general concept!

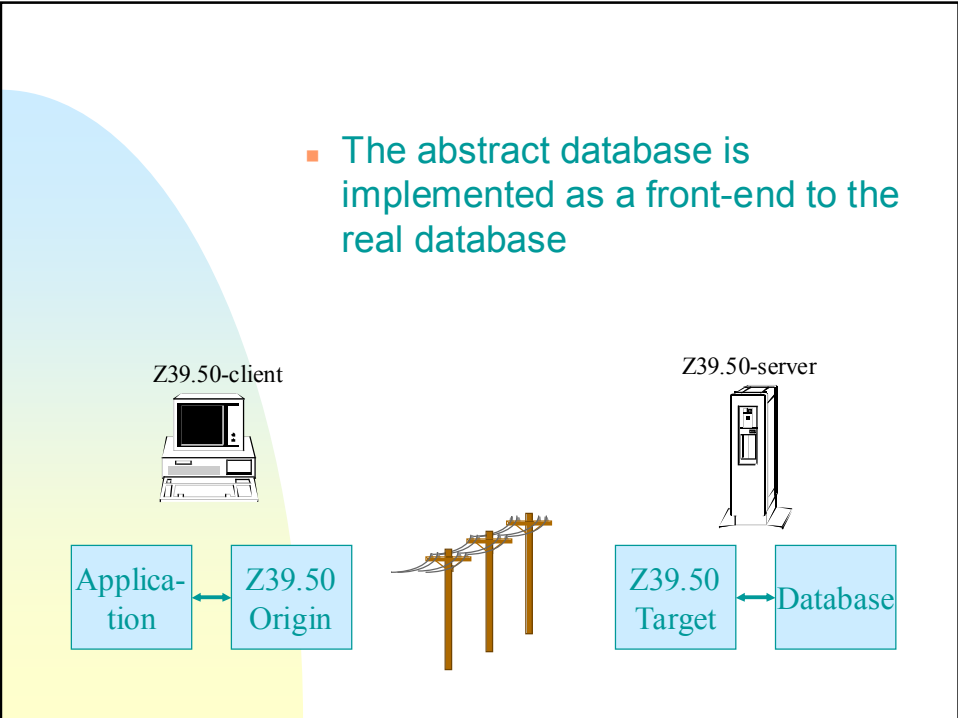
Purpose

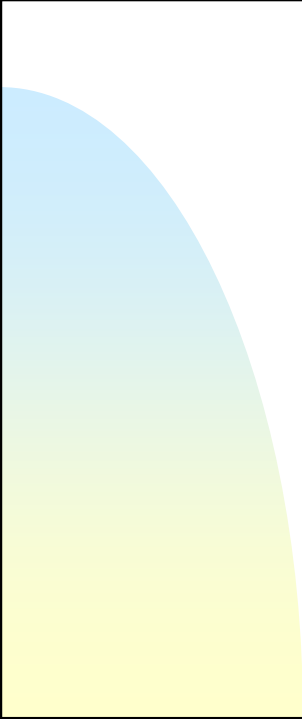
- Interoperability for search and retrieval of information with client/server systems
 - ◆ Interoperability between vendors
 - ☞ Different databases and user interfaces
 - ◆ Interoperability between different organizations
 - ☞ E.g.. using different library formats
 - ◆ Interoperability between groups of users
 - ☞ E.g.. Public libraries/Academic libraries
 - ☞ E.g.. libraries in different countries
 - ◆ Interoperability between communities
 - ☞ E.g.. libraries, publishers, archives, museums

How?



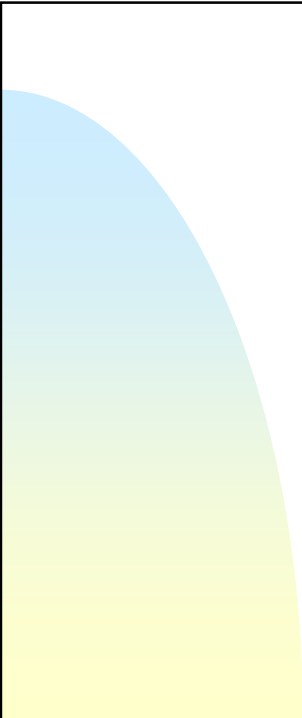
- **Abstract database**
 - ◆ Standardized access points
 - ☞ Attribute sets
 - ◆ Standardized queries
 - ◆ Standardized views
 - ☞ Schemas
 - ☞ Possibilities to select record syntax
 - ☞ Possibilities to select part of record
 - ◆ Searches not tied to record content






Supplementary services

- Scan
- Persistent result sets
- Periodic query
- Item order
- Database update
- Export specification/invocation




Difficulties

- Different databases have different capabilities
 - ◆ Truncation, search indices, implementation of features
- Different databases have different sets of information
 - ◆ US MARC, UNIMARC, LIBRIS MARC, MAB
 - ◆ Embedded holdings or separate holdings



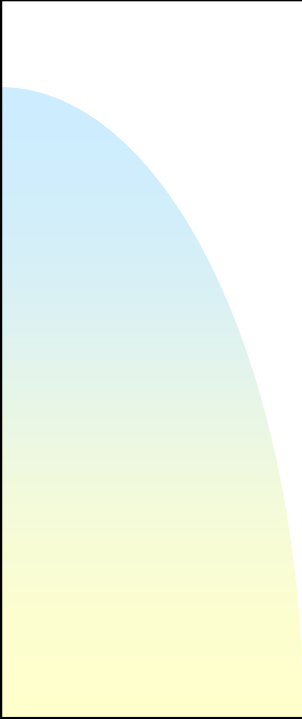
Profiles

- A profile is an agreement about how to use the standard
 - ◆ Which access points are to be used?
 - ◆ Which attributes are applicable?
 - ◆ In what formats should the results be supplied?
 - ◆ What services and supplementary services should be supported?
 - ◆ What options should be supported?
 - ◆ Allowed data for certain fields



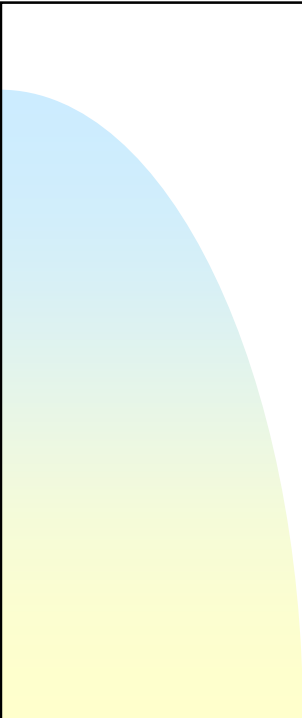
Examples of profiles

- **ATS-1**
 - ◆ Author, Title, Subject
 - ◆ Very basic profile for libraries (obsolete)
- **GILS**
 - ◆ Government Information Locator Service
 - ◆ Profile for document S/R in public administration



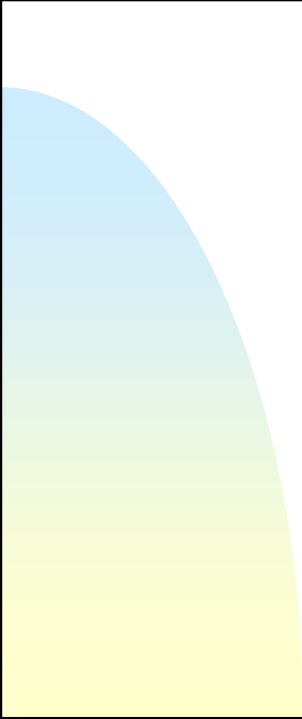
Examples of profiles

- CIMI
 - ◆ Consortium for the Computer Interchange of Museum Information
 - ◆ Not only text. Also specifies how to retrieve images
- CIP
 - ◆ Catalogue Interoperability Protocol
 - ◆ The Committee on Earth Observation Satellites (CEOS)
 - ◆ Search profile for geo-spatial data



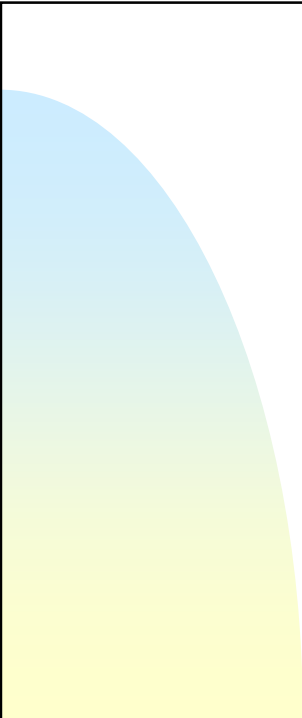
Examples of profiles

- GEO
 - ◆ US government profile for geo-data
- STAS
 - ◆ Scientific and Technical Attribute Set
 - ◆ Not really a profile. More about this later



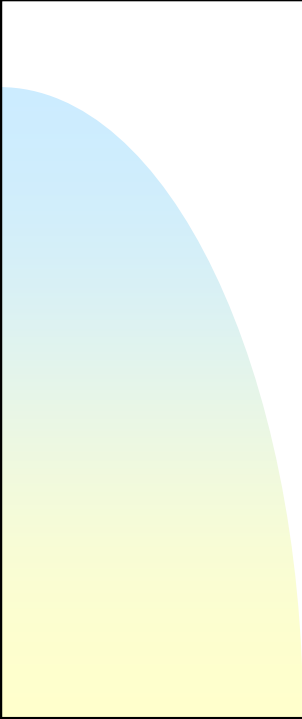
Major library profiles

- ONE
 - ◆ OPAC Network Europe
 - ◆ Developed 1996
 - ◆ Used in the Nordic countries, Germany, UK
 - ◆ Minimum requirements for access points and element sets
- CENL
 - ◆ Conference of European National Librarians
 - ◆ Developed 1997, ratified late 1998
 - ◆ Expands on the ONE profile



Major library profiles

- Finnish Z39.50 profile
- Danish Z39.50 profile
 - ◆ National profiles that add functionality to the international ones
 - ◆ Specify national requirements. E.g.. national classifications
 - ◆ Expand on CENL and ONE respectively



Major library profiles

- Union Catalogue Profile
 - ◆ Defines requirements for cataloguing activity to union catalogue as well as local system through Z39.50
 - ◆ Developed in Australia
 - ◆ Accepted spring 1998



Is Z39.50 any good?

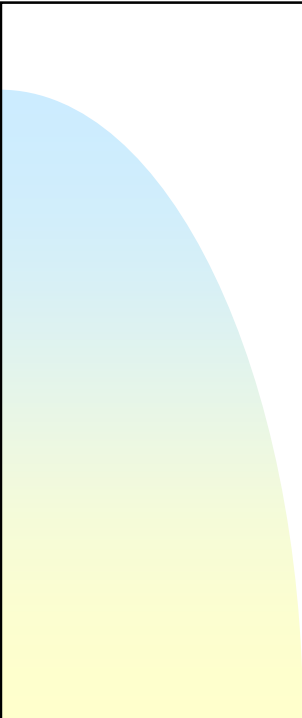
- Very complex
- Difficult terminology
- Originally built on the ISO/OSI protocol
 - ◆ Dominating technology is TCP/IP
 - ◆ Difficult, theory based protocol
 - ◆ Different abstractions
 - ◆ Difficult to re-use existing support services
 - ☞ Authentication
 - ☞ Encryption



Is Z39.50 any good?

- No shrink-wrap products
- Hard to find competent professionals
- Long development cycle for products
- Subject not fully explored before standardization

- Only widespread solution to a difficult problem!

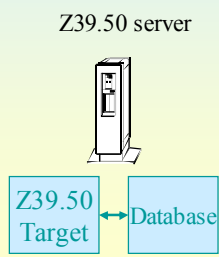


How to apply Z39.50?

- Target
- Gateway
- Origin

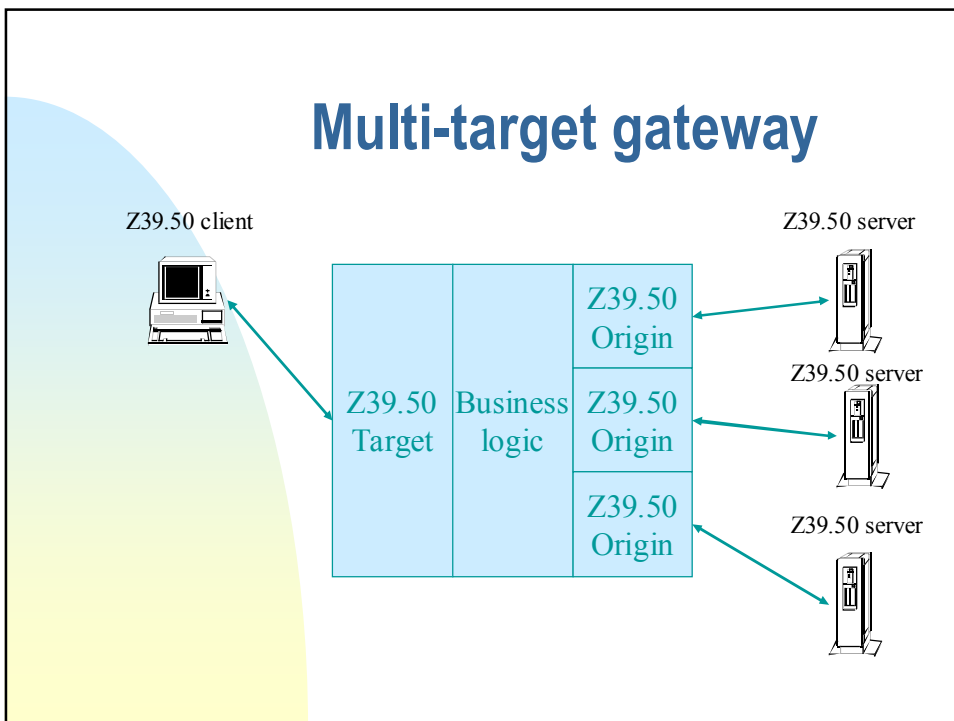
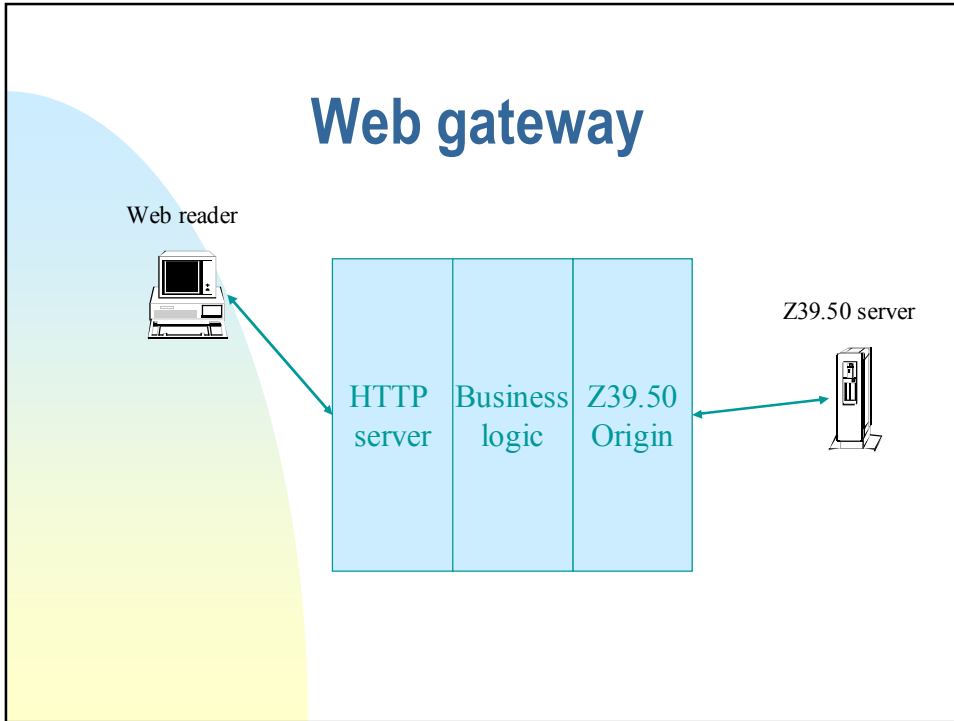
Target

- Implements the abstract database
- Special development
- Customization of toolkit
- Ready made server module
- Often requires advanced configuration
 - ◆ How shall the real database be represented as an abstract one?



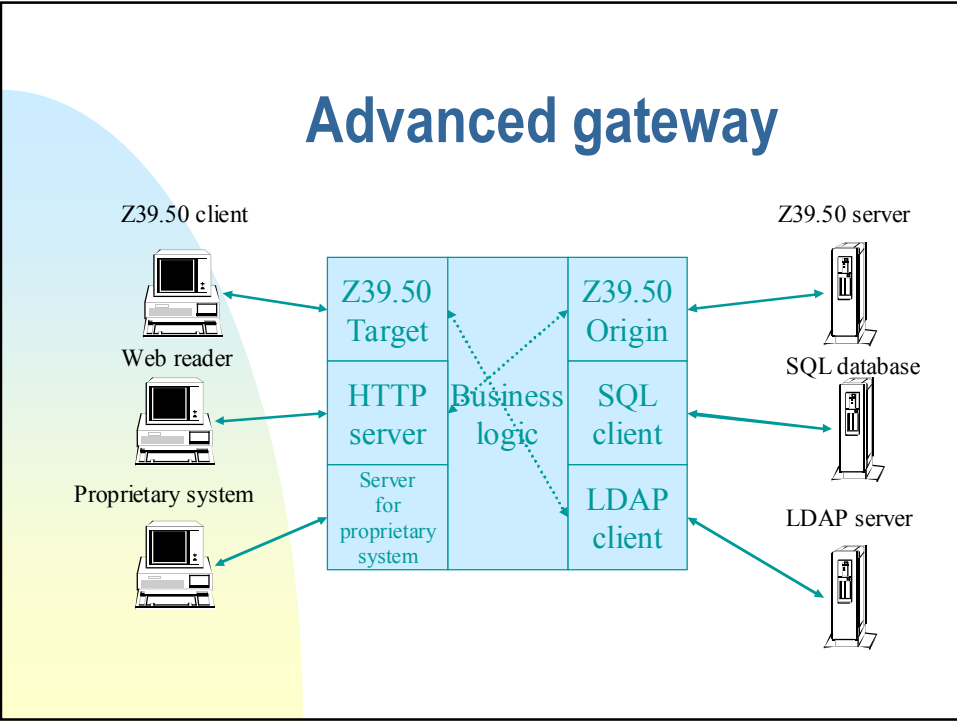
Gateway

- A program that has 2 interfaces
- One where it acts as Origin to a Z39.50 Target
- One where it handles communication with a client application
 - ◆ Client protocol may be HTML, Telnet, Z39.50, etc.



Gateway

- A more advanced Gateway can connect to several Z39.50 Targets
 - ◆ Parallel search
 - ◆ Serial search
 - ◆ Merging of results
- Even more advanced Gateways handle several different protocols on both interfaces
 - ◆ SQL, LDAP, HTML, DNS...



Origin

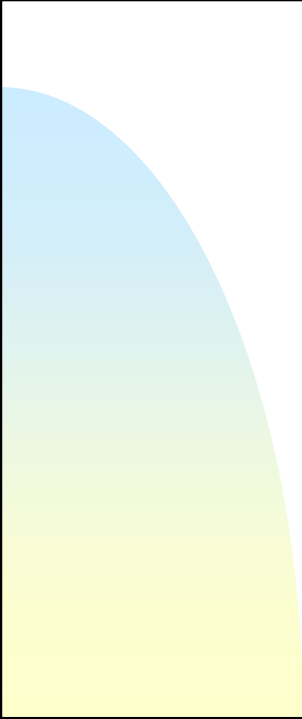
- An Origin is normally part of a graphical client
 - ◆ Hides complexity from the user
 - ◆ Often needs extensive configuring
 - ◆ Can sometimes access several targets simultaneously
 - ◆ There are clients with a “raw” Origin interface

The diagram illustrates the relationship between a Z39.50 client and an application. A computer icon labeled 'Z39.50 client' is positioned above two boxes: 'Application' and 'Z39.50 Origin'. A double-headed arrow connects the 'Application' and 'Z39.50 Origin' boxes, indicating bidirectional communication. The background of the slide features a large, light blue curved shape on the left side.

Market overview

- Integrated systems
 - ◆ Library systems
 - ☞ All large systems support Z39.50
 - ☞ Most have a dedicated client or a web gateway
 - ☞ Some smaller systems use (or rely fully on) Z39.50
 - ☞ Many systems are still version 2, though sometimes with features from version 3
 - Especially American systems

The diagram illustrates the relationship between a Z39.50 client and an application. A computer icon labeled 'Z39.50 client' is positioned above two boxes: 'Application' and 'Z39.50 Origin'. A double-headed arrow connects the 'Application' and 'Z39.50 Origin' boxes, indicating bidirectional communication. The background of the slide features a large, light blue curved shape on the left side.




Market overview

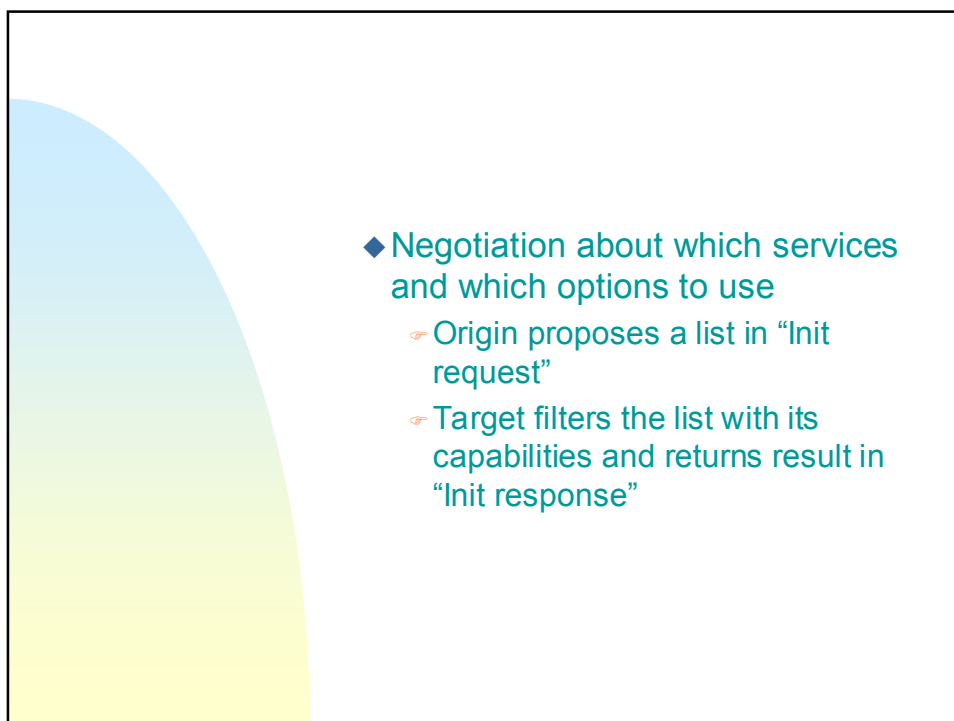
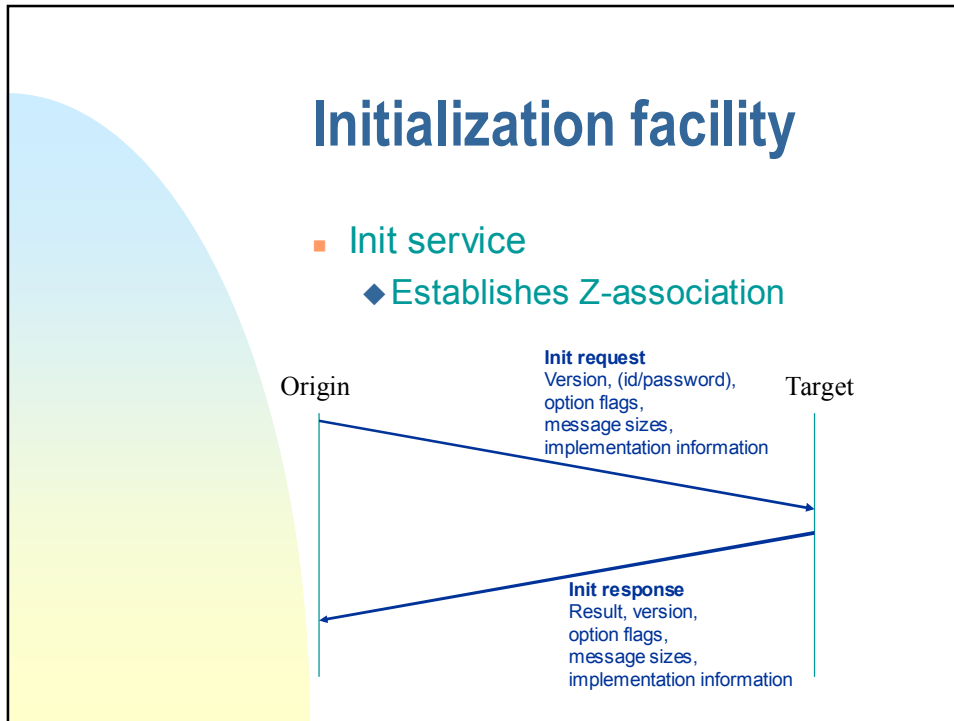
- Standalone products
- Toolkits
- Consultants
 - ◆ Crossnet (UK)
 - ◆ Fretwell-Downing (UK)
 - ◆ Indexdata (Denmark)
 - ◆ Sunstone (Sweden)
 - ◆ Blueangel Technologies (US)
 - ◆ Finsiel (Italy)

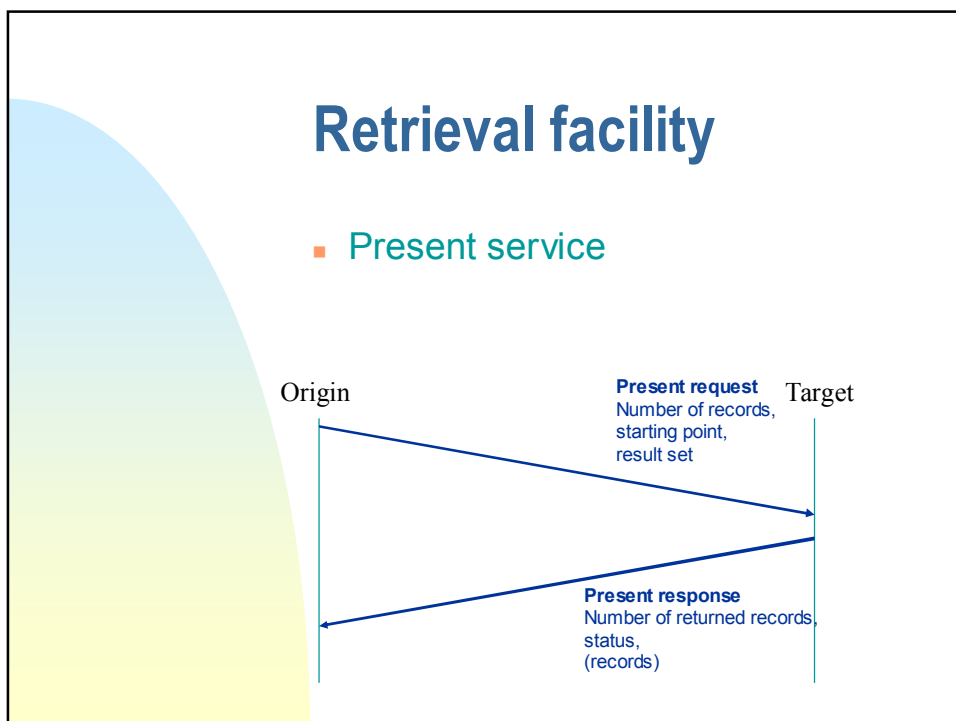
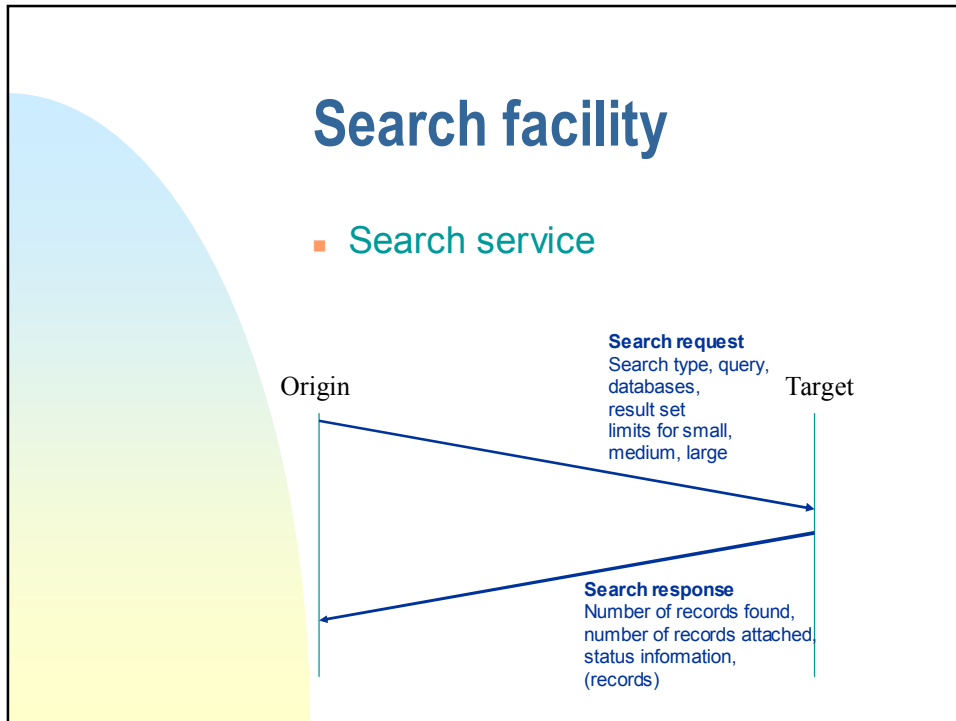


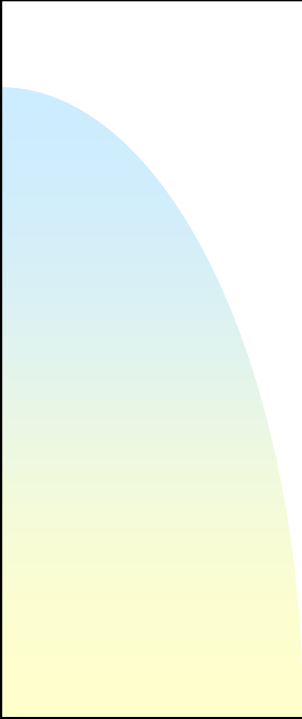
How does Z39.50 work?

- Facilities and Services
 - ◆ A Facility consists of one or more Services










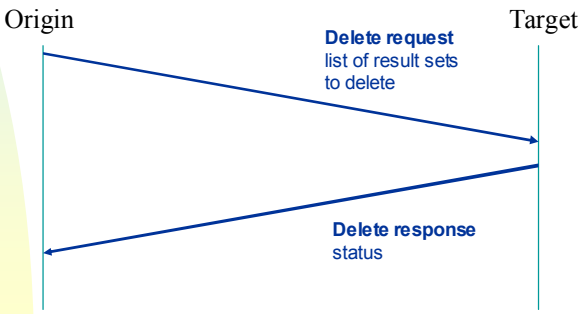
Retrieval facility

- Segment service
 - ◆ Allows a “Present response” that is larger than max size to be split in segments
 - ◆ Two levels
 - ☞ Level 1: only whole records in a segment
 - ☞ Level 2: records can be fragmented

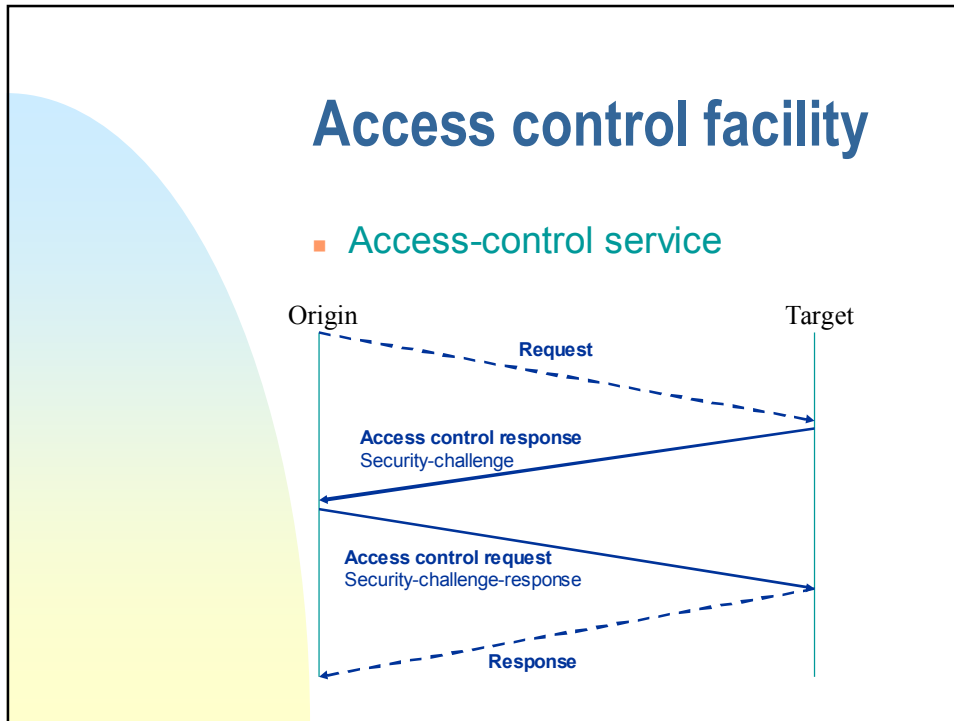


Result-set-delete facility

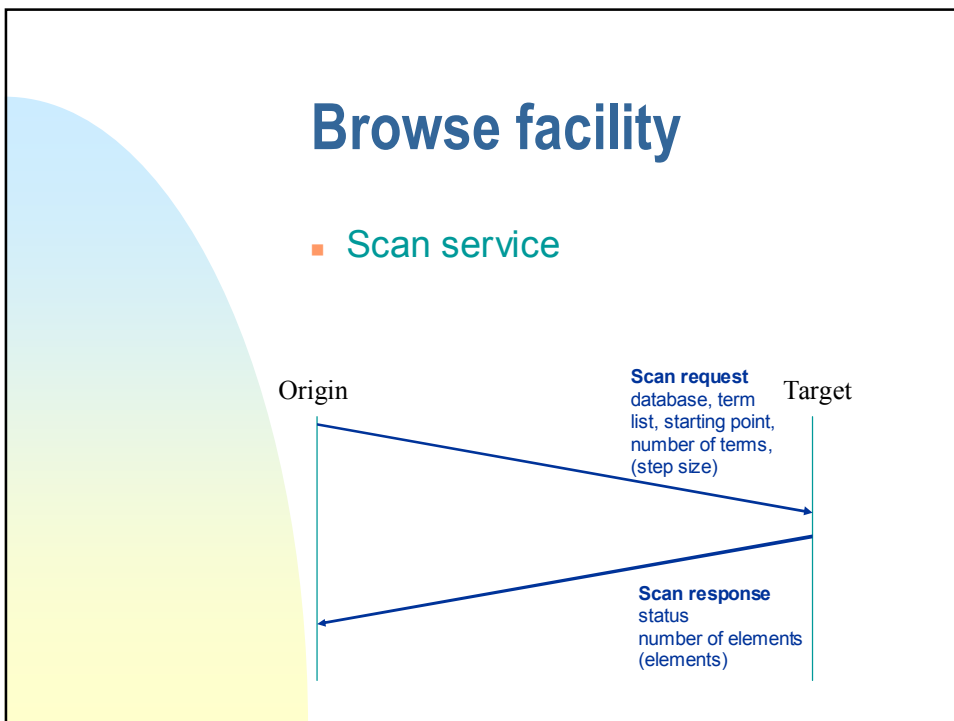
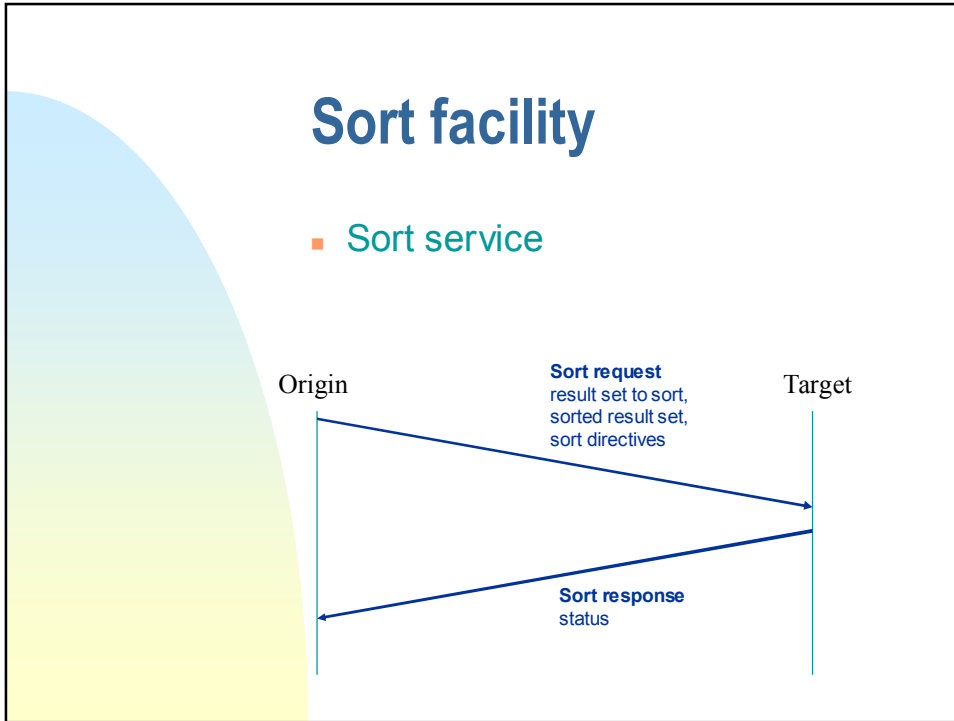
- Delete service

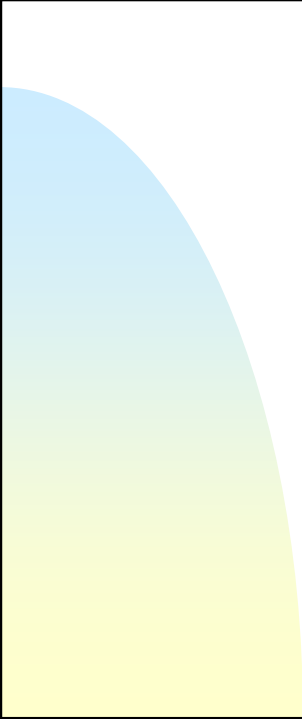


```
sequenceDiagram
    participant Origin
    participant Target
    Origin->>Target: Delete request  
list of result sets  
to delete
    Target-->>Origin: Delete response  
status
```



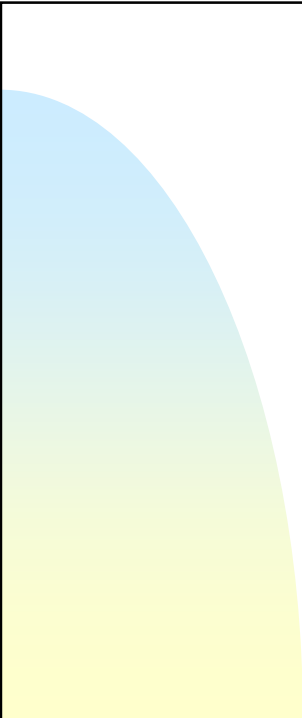
- ## Accounting/Resource control facility
- Resource-control service
 - Trigger-resource-control service
 - Resource-report service
 - ◆ Complex functionality to control and report resource usage
 - ◆ Mostly used for fee based operation





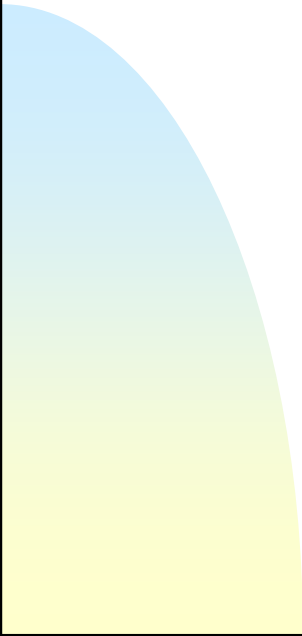
Extended Service facility

- Extended services service
 - ◆ Persistent Result Set Extended Service
 - ◆ Persistent Query Extended Service
 - ◆ Periodic Query Schedule Extended Service
 - ◆ Item Order Extended Service
 - ◆ Database Update Extended Service
 - ◆ Export Specification Extended Service
- Task package
 - ◆ Used to create, modify or delete an Extended Service Request



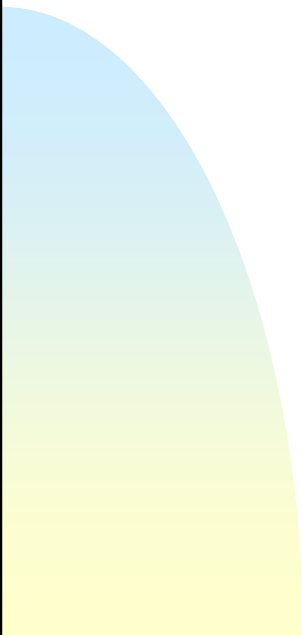
Explain facility

- Explain service
 - ◆ Gives access to information about the Z39.50 target
 - ☞ Databases
 - ☞ Access points
 - ☞ Query languages
 - ☞ Element sets
 - ☞ ...



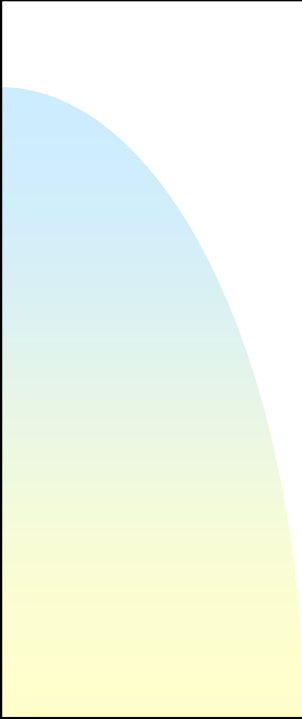
Termination facility

- Close service
 - ◆ Terminates a Z-association



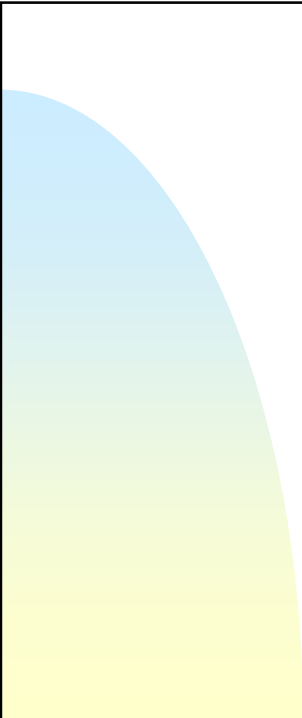
Attribute sets

- The abstract access points that are available, plus domain specific search qualifiers
- BIB-1
- STAS



Carrier protocols

- TCP/IP (usually)
 - ◆ TCP Port 210
- ISO OSI



BER

- Basic encoding rules
 - ◆ A way of coding data for transmission
 - ◆ Coded form not human readable
- Identifier
- Length
- Content

ASN.1

- Abstract Syntax Notation 1
- An implementation independent way of describing data

```

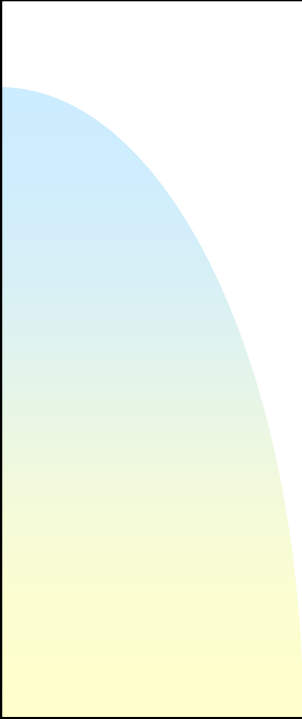
Permissions ::= SEQUENCE OF SEQUENCE{
  userId          [1] IMPLICIT InternationalString,
  allowableFunctions [2] IMPLICIT SEQUENCE OF INTEGER{
    delete          (1),
    modifyContents  (2),
    modifyPermissions (3),
    present         (4),
    invoke          (5) }}
    
```

APDU

- Application Protocol Data Unit
 - ◆ The packages that contain requests and responses

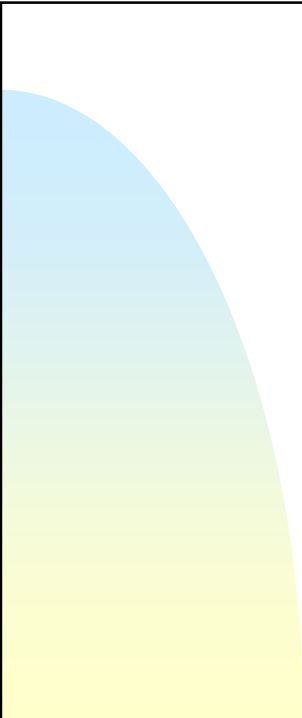
```

InitializeRequest ::= SEQUENCE{
  referenceId          ReferenceId OPTIONAL,
  protocolVersion      ProtocolVersion,
  options              Options,
  preferredMessageSize [5] IMPLICIT INTEGER,
  exceptionalRecordSize [6] IMPLICIT INTEGER,
  idAuthentication     [7] ANY OPTIONAL, -- see note below
  implementationId     [110] IMPLICIT InternationalString OPTIONAL,
  implementationName   [111] IMPLICIT InternationalString OPTIONAL,
  implementationVersion [112] IMPLICIT InternationalString OPTIONAL,
  userInformationField [11] EXTERNAL OPTIONAL,
  otherInfo            OtherInformation OPTIONAL}
--Note:
-- For idAuthentication, the type ANY is retained
-- for compatibility with earlier versions.
-- For interoperability, the following is recommended:
-- IdAuthentication [7] CHOICE{
--   open VisibleString,
--   idPass SEQUENCE {
--     groupId [0] IMPLICIT InternationalString OPTIONAL,
--     userId [1] IMPLICIT InternationalString OPTIONAL,
--     password [2] IMPLICIT InternationalString OPTIONAL,
--     anonymous NULL,
--     other EXTERNAL
--   }
-- May use access control formats for 'other'. See Appendix 7 ACC.
    
```



Queries

- Query types
 - ◆ Type-0: proprietary between 2 parties
 - ◆ Type-1: RPN (standard)
 - ◆ Type-2: ISO 8777
 - ◆ Type-100: Z39.58
 - ◆ Type-101: Extended RPN (v 2)
 - ◆ Type 102: Ranked List query



Type-1 Query

- Consists of
 - ◆ One or more operands, linked with Boolean operators (AND, OR, AND_NOT)
 - ◆ Every operand is a search expression consisting of 7 parts

Operands in Type-1

- 0. Term
 - ◆ What you are looking for
- 1. Use Attributes
 - ◆ Which abstract access point to use
- 2. Relation Attributes
 - ◆ Relation between the term and the data in the access point
 - ◆ E.g.. less than, equals, phonetic equals

Operands in Type-1

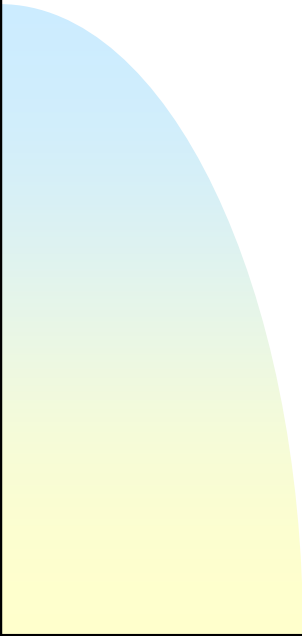
- 3. Position Attributes
 - ◆ Where in the access point should the term be?
 - ◆ E.g.. first in field, first in subfield
- 4. Structure Attributes
 - ◆ How is the term to be treated?
 - ◆ E.g.. as phrase, as words, as date, as normalized name

Operands in Type-1

- 5. Truncation Attributes
 - ◆ Should truncation be applied on the match?
 - ◆ E.g.. left truncation, right and left truncation, no truncation, regular expression
- 6. Completeness Attributes
 - ◆ What is the term to be matched against?
 - ◆ E.g.. part of subfield, whole subfield, whole field

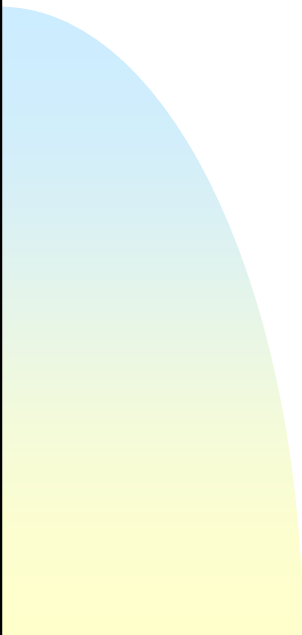
Example of query

- (“Mark Twain”, 1:1003, 2:3, 3:1, 4:1, 5:100, 6:1)
 (“Clemence, Samuel”, 1:1003, 2:3, 3:3, 4:101, 5:100, 6:2)
 AND-NOT



Result sets

- Default result set
- Named result sets
- Persistent result sets
- All contain Result Set Items



Database schema

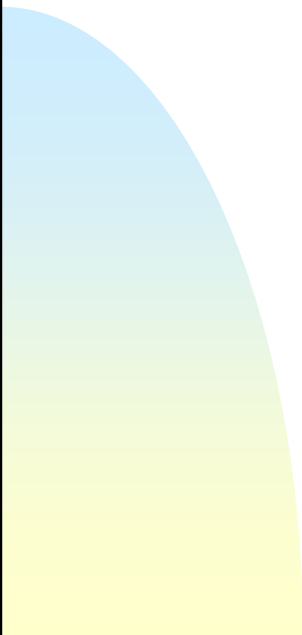
- Definition of the layout of the abstract database
- Contains Elements
 - ◆ Element specification
 - ◆ Element set name



Tags

- Identifiers that uniquely label an element or a substructure

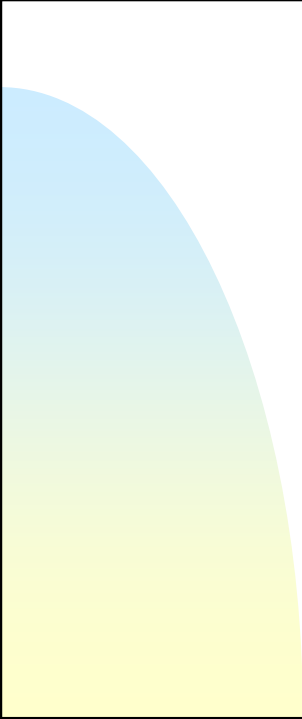
```
schemaIdentifier  
  datatype: OBJECT IDENTIFIER
```



Tag sets

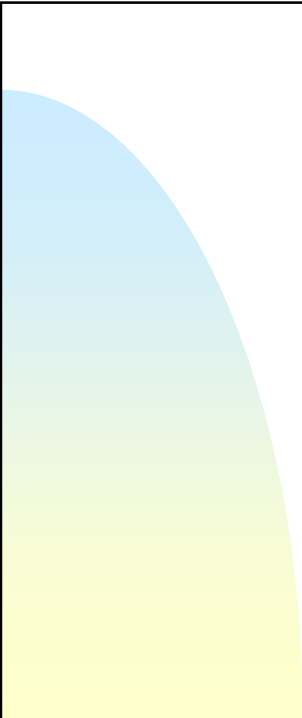
- Sets of identifiers for specific data structures

- 1.schemaIdentifier
 datatype: OBJECT IDENTIFIER
- 2.elementsOrdered
 datatype: BOOLEAN
- 3.elementOrdering
 datatype: INTEGER
- 4.defaultTagType
 datatype: INTEGER



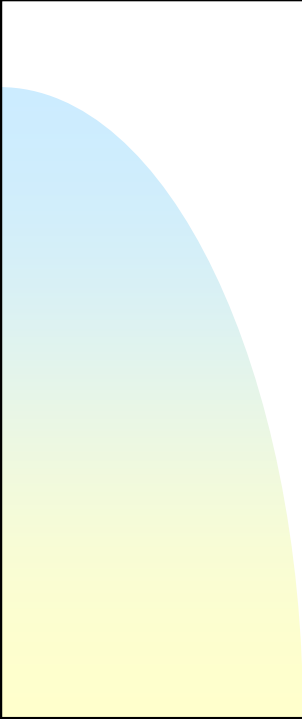
Skipped details

- Composition Specification
 - ◆ A way of indicating which subpart of a data structure you want to retrieve



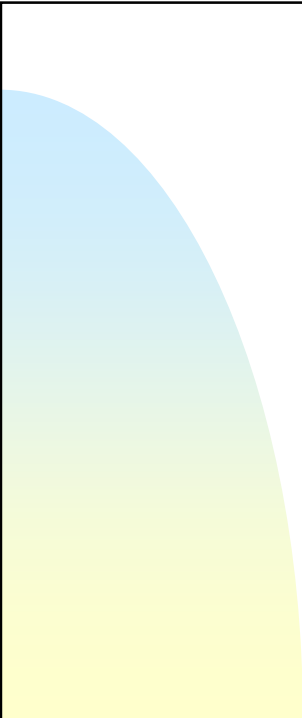
Summary

- Z39.50 is a complex standard that allows interoperability at several levels
- However, interoperability is not for free. It takes knowledge and a lot of hard work to make systems truly interoperable



More information

- The standards text
- Z39.50 Maintenance agency
<http://lcweb.loc.gov/z3950/agency/>
 - ◆ The standards text
 - ◆ Links to profiles
 - ◆ Information about implementors
 - ◆ Amendments, defects, clarifications, ZIG commentaries
 - ◆ Information about upcoming meetings, minutes from previous



More information

- Indexdata AS
 - ◆ YAZ toolkit (written in C)
<http://www.indexdata.dk>
- OCLC
 - ◆ BER Utilities (C, C++ and Java)
ftp://ftp.rsch.oclc.org/pub/BER_utilities/
 - ◆ Toolkit (Java)