Digital Libraries

Interoperability
Z39.50

*W. Arms, Cornell*
Digital Library Systems

Users

Identification Systems

Repositories

Collections

Search Systems

Services
Digital Library Systems: Independent Collections and Services
Interoperability in Heterogeneous Distributed Systems

The Computing Challenge

To build large-scale distributed systems where:

• The components are managed by many different organizations
• Every system is a legacy system
Interoperability in Heterogeneous Distributed Systems

The Computing Challenge

To build large-scale distributed systems where:

• The components are managed by many different organizations
• Every system is a legacy system

*Every Technical Decision has an Organizational Context*
Approaches to standardization

The conventional approach

• Technical leaders develop standards: protocols, formats, etc.
  - Everybody implements the standards.
  - This creates an integrated, distributed system.

Unfortunately ...

• Standards are expensive to adopt.
• Concepts are continually changing.
• Systems are continually changing.
Function versus cost of acceptance

Cost of acceptance

Function
Function versus cost of acceptance
Example: text markup

![Function versus cost of acceptance diagram](image-url)
Function versus cost of acceptance
Example: identifiers

Cost of acceptance

- Domain names
- URL
- URN
Federated digital library

Definition

*Federated digital library.* A group of digital libraries that support common standards and services, thus providing interoperability and a coherent service to users.

In a federation, the partners may have different systems, but must agree on:

- **technical standards** (formats, protocols, interfaces, object models, metadata, etc.)
- **policies** (financial agreements, intellectual property, security, privacy, etc.)
The Z 39.50 federation

*Libraries that agree on:*

- Anglo American Cataloging Rules
- MARC format
- Z39.50 protocol
- Bib1 search query

A successful federation.

An important legacy system.
Aims of Z39.50

- Permits one computer, the client, to search and retrieve information on another, the database server
- Important both technically and for its wide use in library systems
- Most development has concentrated on bibliographic data
- Most implementations emphasize searches that use a bibliographic set of attributes to search databases of MARC records
Sample query

In the database named "Books" find all records for which the access point \textit{title} contains the value "evangeline" and the access point \textit{author} contains the value "longfellow."
Z39.50 principles

Abstract view of database searching.

- Server stores a set of databases with searchable indexes
- Interactions are based on a session
- The client opens a connection with the server, carries out a sequence of interactions and then closes the connection.
- During the course of the session, both the server and the client remember the state of their interaction.
State

Z39.50

• The server carries out the search and builds a *results set*

• Server saves the results set.

• Subsequent message from the client can reference the result set.

• Thus the client can modify a large set by increasingly precise requests, or can request a presentation of any record in the set, without searching entire database.
Z39.50 principles

- Client is a computer.
- End-user applications need a user interface for communication with the user.
- The protocol makes no statements about the form of that user interface or how it connects to the Z39.50 client.
**Z 39.50 services**

**init** -- client connects to the server and exchanges initial information, e.g., preferred message size

**explain** -- client inquires of the server what databases are available for searching, the fields that are available, the syntax and formats supported, and other options

**search** -- client presents a query to a database choices of syntax for specifying searches

- only Boolean queries widely implemented
- one or more records may be returned to the client
**Z 39.50 services**

*manipulation of results sets* -- e.g., *sort* or *delete*

*present* -- requests the server to send specified records from the results set to the client in a specified format

- options: for controlling content and formats
  - for managing large records or large results sets
Technical history

**Z39.50**

- Developed for X.25 networks (connection orientation), conversion to run over TCP fitted later
- Original concept in days when repeating a search was expensive computation (about 1980)
- WAIS is a stateless derivative of an early version of Z39.50