ADVANCES IN GEOREFERENCED DIGITAL LIBRARIES

Greg Janée
(for Terence R. Smith & Michael Freeston)
Alexandria Digital Library Project

www.alexandria.ucsb.edu

Outline

- ADEPT overview
- Core library architecture
  - Metadata interoperability
  - Query translation
  - Collection discovery
- Gazetteers and their application
- Concept modeling & educational applications

Quick digital library overview

- Third layer of Internet
  - "library'" layer
  - persistence, accessibility, and organization
- Digital libraries (DLs) characterized by
  - Collections
  - (Interoperable) Services
- Collections characterized by
  - metadata at collection and item level
- Models of DL organization
  - harvesting/central metadata
  - distributed peer-to-peer DLs (ADEPT model)
- Services for
  - discovering/accessing knowledge
  - using knowledge
  - creating knowledge

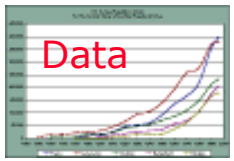
NSF-SUPPORTED DL ACTIVITIES

- DLI-1
  - 94-98
  - 6 projects
- DLI-2
  - 99-04
  - About 30 projects
- NSDL
  - 00-06
  - About 70 projects
- DLESE
  - 99-?
  - 1 project

ADEPT GOALS

- Goals
  - distributed digital library for georeferenced information
  - services supporting DL federation and interoperation
  - personalized research and learning spaces
    - **services for constructing personalized collections**
    - **access based on KB of concepts**
- Scalability
  - many collections
  - collections, very large to very small
  - extreme heterogeneity

Place-based information challenge



Papers

Maps

Books

Metadata

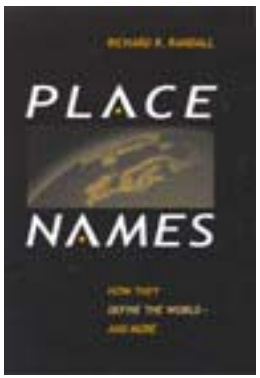
```
<!ENTITY % geographic-coordinate "(#PCDATA)">
<!-- a geographic latitude in degrees north of the equator or
        geographic longitude in degrees east of the Greenwich
        meridian, e.g., "-121.025" -->
<!ELEMENT west_bounding_coor %geographic-coordinate;>
<!ELEMENT east_bounding_coor %geographic-coordinate;>
<!ELEMENT south_bounding_coor %geographic-coordinate;>
<!ELEMENT north_bounding_coor %geographic-coordinate;>
<!ELEMENT measurement_begin_date %calendar-date;>
```

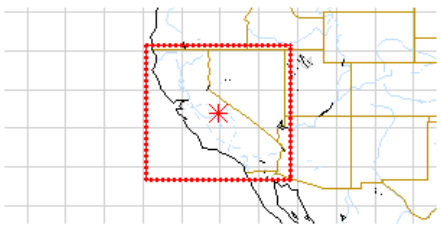
Harvested Webpages

Georeferencing by placename and by spatial footprint

Translation needed between placenames - locations

GIS datasets





Gazetteers

Oral histories

ADEPT Core Library Architecture

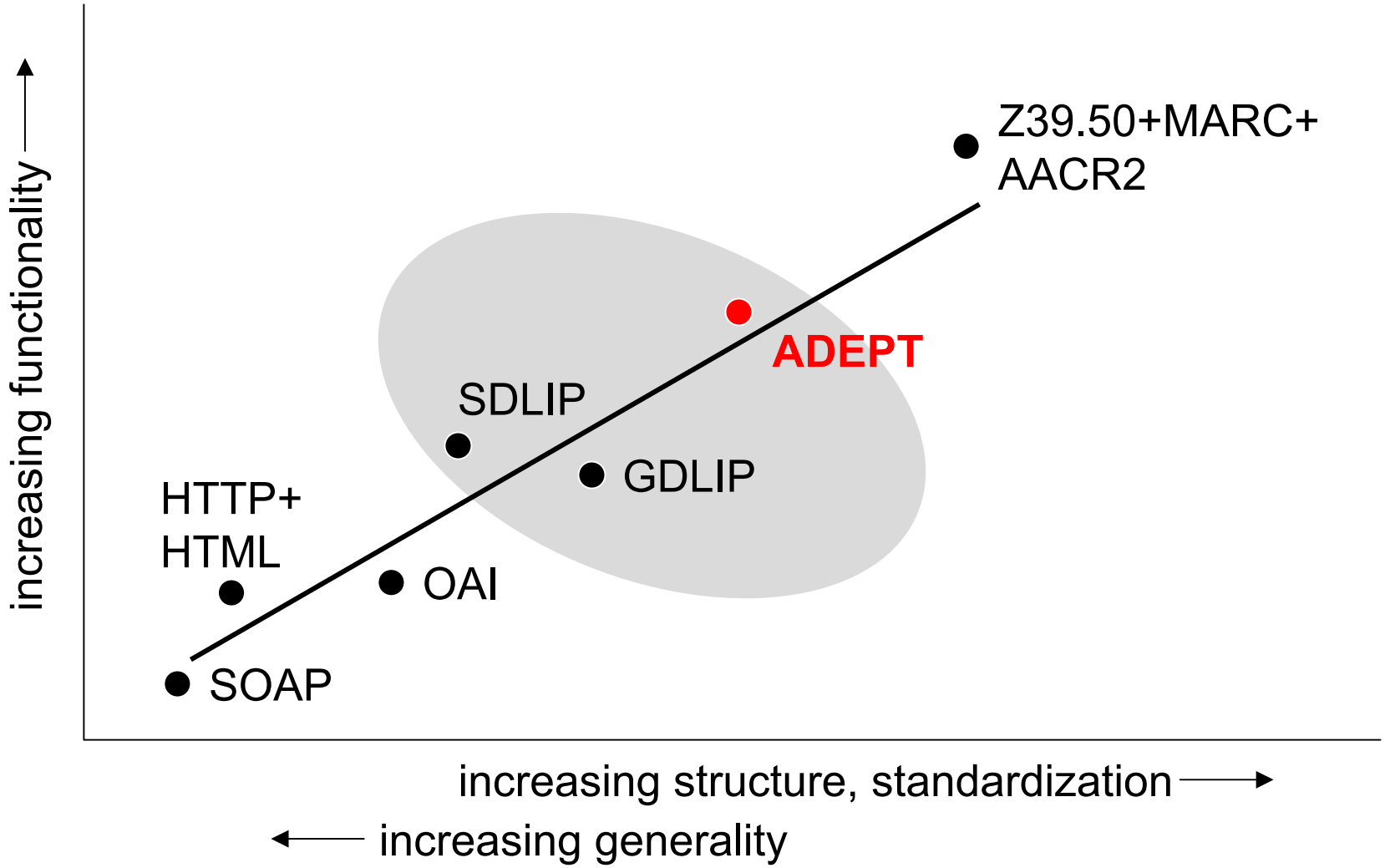
Core ADL/ADEPT architecture goals

- Goals
  - distributed digital library for georeferenced information
  - services supporting DL federation and interoperation
  - "personalized learning spaces"
- Scalability
  - many collections
  - collections, very large to very small
  - extreme heterogeneity

SHOW: Current (thin) ADEPT Client

- Initial view
- Querying detail of initial view
- Result detail of initial view
- Result of query

INTEROPERABILITY LANDSCAPE



Components/services

collection registry

collection-level search

thesaurus

shared vocabularies

library

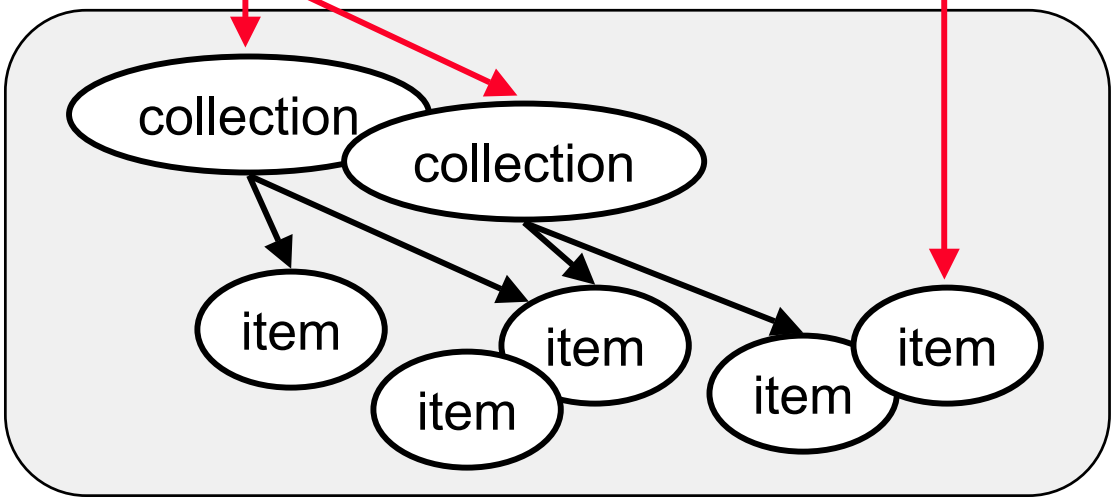
item-level search,
metadata management

content

data access

gazetteer

maps placenames
to locations



map

background imagery,
layering capability

*many interconnections
between services*

Data model

- Collection
  - name
  - static, dynamic metadata
  - set of items
  - functional behaviors
- Item
  - identifier
  - bucket view
    - **searchable metadata mapped to standard, typed buckets**
  - browse view
    - **content abstracts**
- Item, cont'd
  - access view
    - **multiple access points**
      - file-like
      - human interface
      - programmatic service
      - offline
  - other views
    - **collection- and/or item-specific**
    - **FGDC, MARC, etc.**
  - content

Library services

- configuration
- collection-metadata
  - retrieve
- item-metadata
  - retrieve views
- query
  - standard query language
- result-set
  - access server-cached query result sets
- harvest
  - collection
- collection-management
  - {create, delete, replace} static metadata
- item-management
  - {create, delete, replace} views
- reference
  - remote collection

Library server architecture

user interface

metadata mapper

harvest loader

item tracker

client interface (XML / Java,HTTP,RMI)

middleware

access control; query fan-out; query result caching & ranking
collection referencing & registration

collection interface (XML / Java)

internal collections

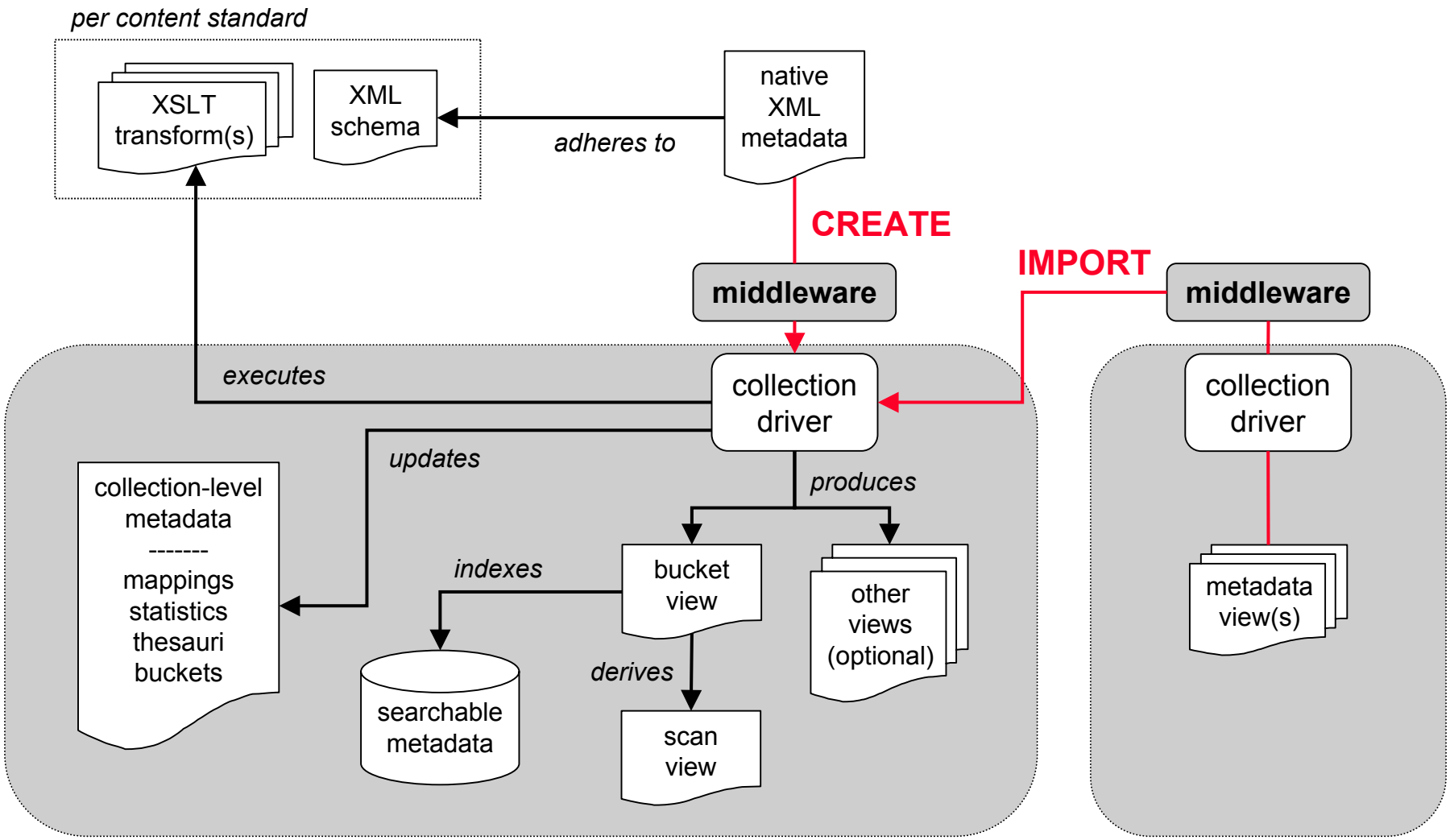
generic database driver

Z39.50 driver

proxy driver

collection aggregator

"Local" collection population



Metadata Interoperability

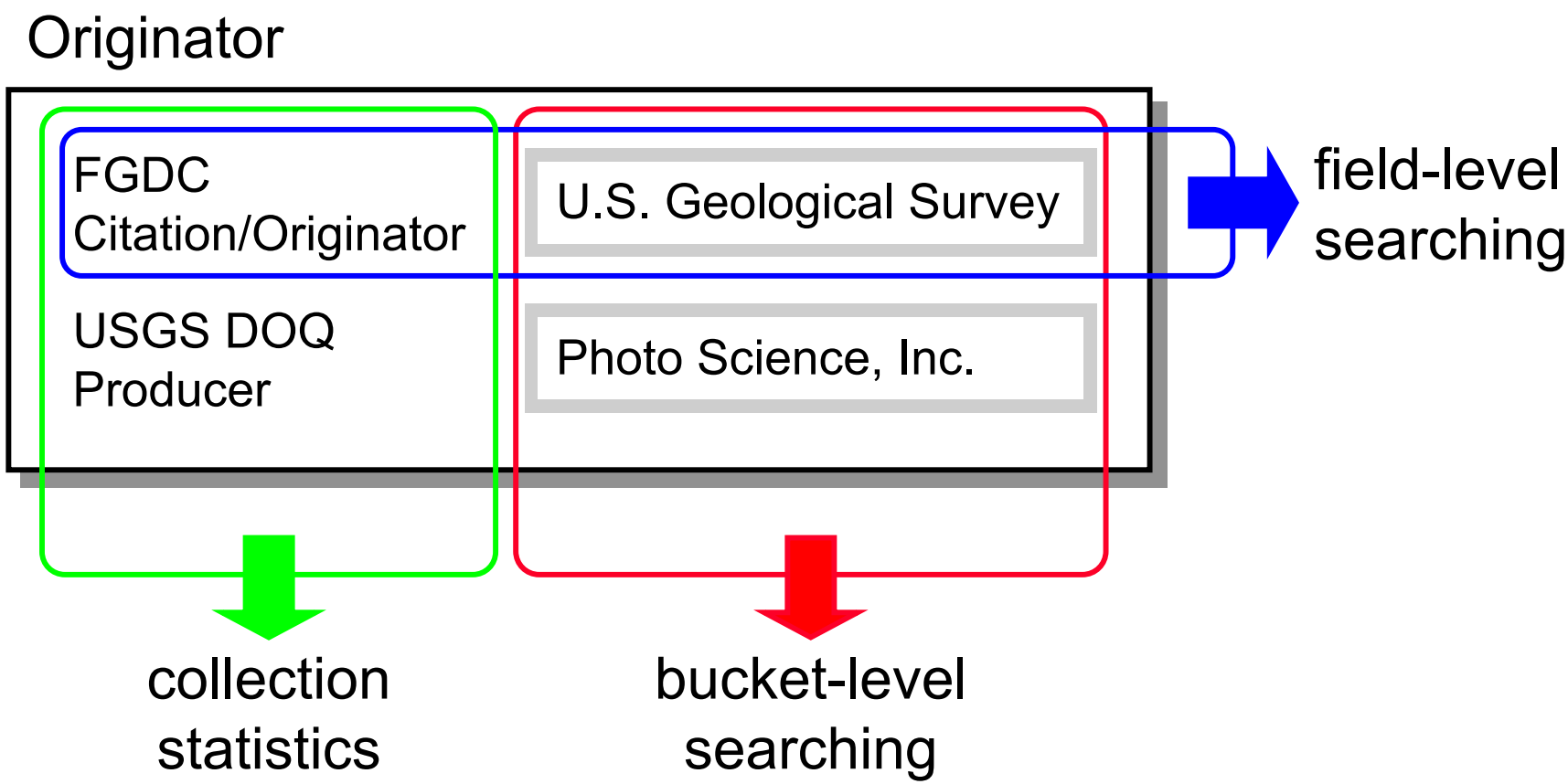
ADEPT's interoperability problem

- Distributed, heterogeneous collections
  - locally, autonomously created and managed
- Minimal requirements on collection providers
  - allow use of native metadata
- Provide uniform client services
  - common high-level interface across collections
  - structured means of discovering and exploiting (possibly collection-specific) lower-level interfaces
- Assumptions
  - items have metadata
  - items have sufficient, "good" metadata
  - i.e., this is a metadata interoperability problem

What is a bucket? (1/2)

- Strongly typed, abstract metadata category with defined search semantics to which source metadata is mapped
- Key properties
  - name
    - **Coverage date**
  - semantic definition
    - **The time period to which the item is relevant.**
  - data type (strictly observed)
    - **calendar date or range of calendar dates**
  - syntactic representation (strictly observed)
    - **ISO 8601**

Bucket mapping



What is a bucket? (2/2)

- Source metadata is mapped to buckets
  - buckets hold *not* just simple values
    - **"2001-09-08"**
  - but rather, explicit representations of mappings
    - **(FGDC, 1.3, "Time period of content", "2001-09-08")**
  - multiple values may be mapped per bucket
- Bucket definition includes search semantics
  - defines query terms
    - **ISO 8601 date range**
  - defines query operators
    - **contains, overlaps, is-contained-in**
  - semantics are slightly fuzzy in certain cases to accommodate multiple implementations

Collection-level aggregation

- Collection-level metadata describes
  - buckets supported by the collection
  - item-level metadata mappings
  - statistical overviews
    - **item counts**
    - **spatiotemporal coverage histograms**
- Example (de-XML-ized)
  - in collection *foo*, the *Originator* bucket is supported and the following item fields are mapped to it:
    - **(FGDC, 1.1/8.1, "Citation/Originator") [973 items]**
    - **(USGS DOQ, PRODUCER, "Producer") [973 items]**
    - **(DC, Creator, "Creator") [1249 items]**
    - **unknown [6 items]**

Searching collections

- Bucket-level
  - uniform across all collections
  - example
    - **search all collections for items whose *Originator* bucket contains the phrase "geological survey"**
- Field-level
  - collection-specific
  - but discovery and invocation mechanisms are uniform
  - example
    - **search collection *foo* for items whose *FGDC 1.1/8.1* field within the *Originator* bucket contains the phrase…**

Bucket types (1/7)

- 6 bucket types: spatial, temporal, hierarchical, textual, identification, numeric
- Type captures the portion of the bucket definition that has functional implications
  - data type & syntactic representation
  - query terms
  - query operators
- Complete bucket definition
  - name
  - semantic definition
  - bucket type

Bucket types (2/7)

- Spatial
  - data type: any of several types of geometric regions defined in WGS84 latitude/longitude coordinates
  - syntax: defined by ADEPT
  - query terms: WGS84 box or polygon
  - operators: contains, overlaps, is-contained-in
  - example query:
    - **<spatial-constraint**
      bucket="geographic-location"
      operator="overlaps">
      <box north="37.5" south="30.0" east="-110"
      west="-140"/>
      </spatial-constraint>

Bucket types (3/7)

- Temporal
  - data type: calendar date or range of calendar dates
  - syntax: ISO 8601
  - query term: range of calendar dates
  - operators: contains, overlaps, is-contained-in
  - example query:
    - **<temporal-constraint**
      bucket="coverage-date"
      operator="contains"
      from="1970-01-01" to="1979-12-31"/>

Bucket types (4/7)

- Hierarchical
  - data type: term drawn from a controlled vocabulary (thesaurus, etc.)
  - one-to-one relationship between hierarchical buckets and vocabularies
  - query term: vocabulary term
  - operator: is-a
  - example query:
    - **<hierarchical-constraint**
      bucket="feature-type"
      operator="is-a"
      vocabulary="ADL Feature Type Thesaurus"
      term="populated place"/>

Bucket types (5/7)

- Textual
  - data type: text
  - query term: text
  - operators: contains-all-words (*special semantics*), contains-any-words, contains-phrase
  - example query:
    - **<textual-constraint**
      bucket="subject-related-text"
      operator="contains-all-words"
      text="orthophotograph"/>

Bucket types (6/7)

- Identification
  - data type: text, optionally namespace-qualified
  - query term: same
  - query operator: matches
  - example query:
    - **<identification-constraint**
      bucket="identifier"
      operator="matches"
      text="90-70002-34-5"
      namespace="ISBN"/>

Bucket types (7/7)

- Numeric
  - data type: real number
  - query term: real number
  - query operators: standard relational operators
  - example query:
    - **<numeric-constraint**
      bucket="minimum-feature-size"
      operator="less-than"
      value="1.0"
      unit="meters"/>

Bucket types vs. buckets

- Bucket *types* are defined architecturally
- Buckets in use are defined by collections and items
  - need standard buckets, defined conventionally, to support cross-collection uniformity
- ADL core buckets
  - simple; universal; easily & broadly populated; useful
- Bucket descriptions in the following slides:
  - bucket type
  - semantic definition
  - comparison to Dublin Core

ADL core buckets (1/6)

- Subject-related text
  - Title
  - Assigned term
- Originator
- Geographic location
- Coverage date
- Object type
- Feature type
- Format
- Identifier

ADL core buckets (2/6)

- Subject-related text
  - type: textual
  - description: text indicative of the subject of the item, not necessarily from controlled vocabularies
  - superset of *Title* and *Assigned term*
  - compare: DC.Subject
- Title
  - type: textual
  - description: the item's title
  - subset of *Subject-related text*
  - compare: DC.Title

ADL core buckets (3/6)

- Assigned term
  - type: textual
  - description: subject-related terms from controlled vocabularies
  - subset of *Subject-related text*
  - compare: qualified DC.Subject
- Originator
  - type: textual
  - description: names of entities related to the origination of the item
  - compare: DC.Creator + DC.Publisher

ADL core buckets (4/6)

- Geographic location
  - type: spatial
  - description: the subset of the Earth's surface to which the item is relevant
  - compare: DC.Coverage.Spatial
- Coverage date
  - type: temporal
  - description: the calendar dates to which the item is relevant
  - compare: DC.Coverage.Temporal

ADL core buckets (5/6)

- Object type
  - type: hierarchical
  - vocabulary: ADL Object Type Thesaurus (image, map, thesis, sound recording, etc.)
  - compare: DC.Type
- Feature type
  - type: hierarchical
  - vocabulary: ADL Feature Type Thesaurus (river, mountain, park, city, etc.)
  - compare: none

ADL core buckets (6/6)

- Format
  - type: hierarchical
  - vocabulary: ADL Object Format Thesaurus (loosely based on MIME)
  - compare: DC.Format
- Identifier
  - type: identification
  - description: names and codes that function as unique identifiers
  - compare: DC.Identifier

Summary

- A *bucket* is a strongly typed, abstract metadata category with defined search semantics to which source metadata is mapped
- Supports discovery/search across distributed, heterogeneous collections that use metadata structures of their choosing
- Uses high-level search buckets for cross-collection searching and supports "drill-down" searching to the item-level metadata elements

Challenges

- Metadata is like life: it refuses to follow the rules
  - unknown semantics; inconsistent typing/syntax; unknown or unidentifiable sources; poor quality; inconsistent quality; proliferation of overlapping vocabularies; ...
- Realities of the marketplace: Dublin Core won
  - adapt approach to qualified Dublin Core
  - incorporate either fallback mechanism or polymorphism
    - **e.g, treat fields as thesauri/controlled vocabularies or as text**

Query Translation

ADEPT query language

- Domain
  - a collection of items
  - each item has unique ID and 1+ fields
  - field = (name, value)
  - bucket = (name, union or concatenation of fields)
- Queries
  - atomic constraint: (attribute name, operator, target)
    - **semantics: return items that have 1+ values for the attribute, for which at least one value matches the target**
  - arbitrary boolean combinations
    - **AND, OR, AND NOT**

The problem

- Algorithmically translate ADEPT queries to SQL
  - ideally, accommodate all possible SQL implementations
  - configuration must be possible by mere mortals
  - must generate "reasonable" SQL
  - e.g., an unacceptable approach:
    - **(A, op, V) -> SELECT id FROM table WHERE cond(V)**
    - **(A1, op1, V1) B (A2, op2 , V2) ->**
      SELECT id FROM table1 where cond1(V1)
      B
      id IN (SELECT id FROM table2 WHERE cond2(V2))
  - ideally, could incorporate optimization considerations

Approach

- Python-based translator
  - ~1500 lines
- Employs extensible system of "paradigms" for describing atomic translation techniques
  - 15 paradigms
  - Each paradigm ~100 lines (50 Python code, 20 assertions, 30 documentation)
- Uses rules (intrinsic & explicit) to combine booleans
  - preferentially unifies; then JOINs; then self-JOINs, etc.
- Configuration file describes:
  - buckets, fields, paradigms, paradigm configuration
  - boolean override rules
  - misc: external identifier table, optimizer clauses

Translation paradigms

- Paradigm:
  - translateBucketAtomic (constraint) -> query
  - optional:
    - **translateBucketBoolean (boolOp, constraintList)**
    - **translateFieldAtomic, translateFieldBoolean**
    - **"adaptors" for standard field techniques**
- Example: Hierarchical_IntegerSet
  - SELECT id FROM table WHERE column IN (codelist)
  - codelist obtained via separate thesaurus interface
  - configuration: table, id, column, thesaurus info, cardinality
- Cardinality: 1, 1?, 1+, 0+
  - row multiplicity (really functional dependence on identifier)
  - optionality

Intermediate query form

- Query:
  - 1+ tables, expression
  - table: name
  - main table: table + id, cardinality
    - **IDs assumed to be equi-joinable**
  - qualified main table: main table + qualification condition
  - aux table: table + join condition
- Structure necessary for
  - analysis of unification, JOIN possibilities
  - translation correctness
    - **SELECT [t] cond1 …AND NOT…**
      SELECT [t, taux] joincond AND cond2
    - **-> SELECT [t, taux] joincond AND cond1 AND NOT cond2**

Combining queries

- Consider T(v) =
  SELECT id FROM t WHERE c IN (codelist(v))
- T(v1) AND T(v2)
  - if cardinality 1 or 1? can unify:
    SELECT id FROM t WHERE c IN (codelist(v1)) AND c IN (codelist(v2))
  - else: self-join or subquery
- In general:
  - Query({tables}, expression) boolOp
    Query({tables}, expresion)

Future work

- Paradigm system works well
- Boolean processing seems amenable to a more formal treatment
- Large, relevant literature
  - Qian & Raschid: algorithmic translation of XSQL to SQL
    - **very complex; not for mere mortals**
  - ADEPT query language is much simpler
    - **and common (Z39.50, WebDAV basicsearch, …)**
- Challenge: generate consistently good SQL
  - stupid things like order of tables & conditions matter
  - make up for DB deficiencies
  - tackle the JOIN problem

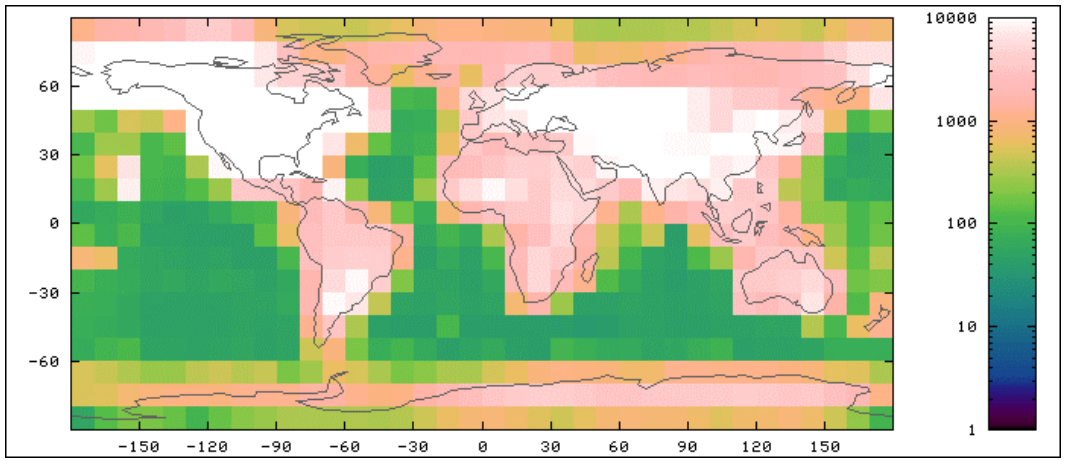
Collection Discovery

The problem

- Distributed queries: necessary evil
  - necessary to achieve scalability
    - **performance**
    - **autonomy**
  - introduce scalability, performance, and reliability problems
- Amelioration strategies
  - increase server performance/reliability
    - **replication, DIENST connectivity regions**
  - turn into offline problem
    - **Web search engines, OAI harvesting model**
  - identify relevant collections to query (ADEPT)
    - **analogous to Web search engine**
- Challenge: identify relevant collections

Approach

- Build on collection-level metadata
  - spatial & temporal density histograms; item counts broken down by collection categorization schemes
    - **more is better**



- Upload periodically to central server
- Replace histograms with Euler histograms to support range queries

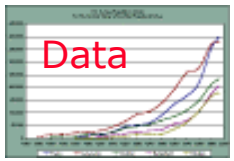
Challenges

- Relevance is not necessarily boolean
  - worldwide, petabyte, 1cm resolution database = world map drawn on napkin?
  - introduce resolution/minimum feature size
    - **but sometimes you want the napkin**
- The problem with JOINs
  - statistics are computed independently
- Integrating text overviews
  - STARTS?

Introduction to digital gazetteers and their development issues

Alexandria Digital Library Project
Gazetteer Development Team

Place-based information challenge



Papers

Maps

Books

Metadata

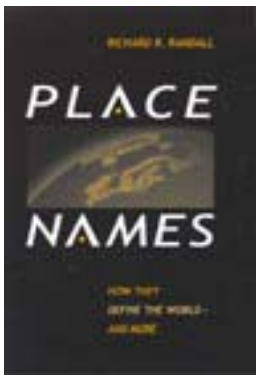
```
<!ENTITY % geographic-coordinate "(#PCDATA)">
<!-- a geographic latitude in degrees north of the equator or
        geographic longitude in degrees east of the Greenwich
        meridian, e.g., "-121.025" -->
<!ELEMENT west_bounding_coor %geographic-coordinate;>
<!ELEMENT east_bounding_coor %geographic-coordinate;>
<!ELEMENT south_bounding_coor %geographic-coordinate;>
<!ELEMENT north_bounding_coor %geographic-coordinate;>
<!ELEMENT measurement_begin_date %calendar-date;>
```

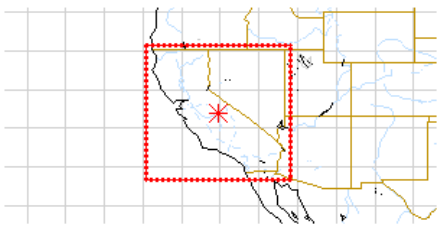
Harvested Webpages

Georeferencing by placename and by spatial footprint

Translation needed between placenames - locations

GIS datasets





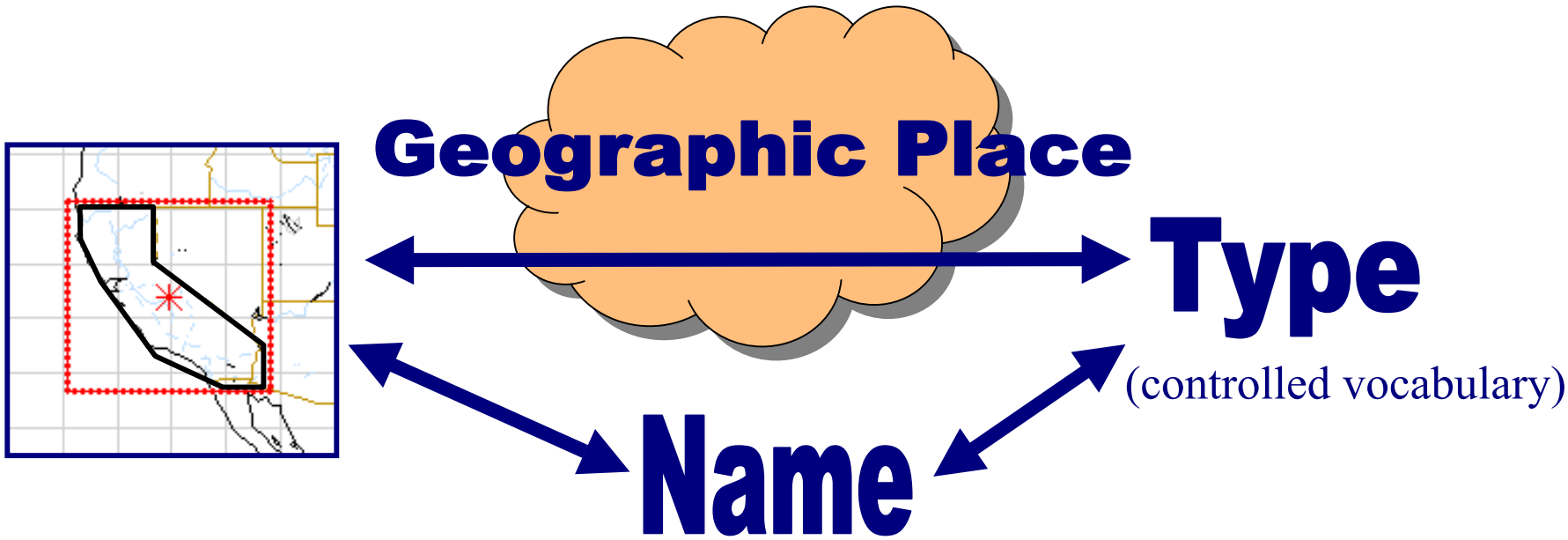
Gazetteers

Oral histories

What's a gazetteer?

- Originally (in the simplest case)
  - setof (name, location)
    - **the "index" in an atlas**
    - **a "geographical dictionary"**
- ADL basics
  - setof (name, type, location)
- ADL extended
  - Time-stamped names, extents, and relationships
  - Descriptive information about names and places
  - Merging of information about a place from multiple sources
- Preferred definition
  - Spatial dictionary of named and typed places

Digital gazetteer essentials



Roles of gazetteers in digital libraries

- Collections
  - useful information in their own right
- References
  - canonical (official or preferred) names and locations
- "Finding aids"
  - where's this? location = gaz(name, type)
  - what's here? (name, type) = gaz(location)

Gazetteers as georeferencing services

- Implicit: turn textual references into locations
  - location = gaz(geoparse(text))
  - Textual Geospatial Integration (TGI) project goal
    - **Text-Geo query**
    - **Geo-Text query**
- Indirect: use gazetteer locations as query constraints
  - query(..., gaz(name, type))

Digital libraries and gazetteers

- Standards + Services =
  - Communities >> domain-specific gazetteers
  - Protocols >> search & retrieval for distributed gazetteers
- Federations
  - "middleware" (broker) aggregates access to multiple gazetteers

Spatial representation of place

Footprints (latitude/longitude values)

- Nature and usefulness of spatial generalizations
  - Points **– most common; useful for disambiguating one place from another**
  - Bounding boxes **– simplest footprint for spatial extent; easy to handle in information systems; faithfulness to shape is a problem**
  - Generalized polygons **– needs to be defined for gazetteer information services: how many points; effect of generalization on retrieval**
  - Complex polygons **– computationally intensive to handle**
- Inherent spatial relationships: contains, overlaps, is-contained-by, adjacent
- Explicit statements of relationships
- Documenting spatial accuracy

Temporal aspects of gazetteer data

- Representation of
  - Historical placenames
  - Spatial extents linked to time
  - Historical administrative relationships
  - Historical data values: e.g., population
  - Historical types/roles: e.g., church becomes a school
- Highly important for cultural history collections, specimen collection sites for previous expeditions, …
- Issues
  - Structural design issues for linking time-stamped description elements together
  - User interface design for time-based searching and display

Names for geographic places

- Concept of "the" name versus variant names
  - Authorized naming bodies
  - Preferred name varies with location and use
  - Attribute set for names (see ADL Gazetteer Content Standard online)
- Language and character code set issues
- Name codes: standard codes for postal addresses and other purposes
- "Surnames" as indicators of type of place

Useful

- Perth Airport
- Baldwin County
- Admiralty Oil Seep
- Jar Qudug Gas Field
- Sussex Correctional Institution

Not Useful

- Kindley Field
- The Rock
- Toledo

Typing

- Typing supports queries such as
  - "What schools exists Miami and where are they?"
  - Show wetlands in southern Florida
- Typing schemes
  - List
  - Hierarchical (2-level list)
  - Thesaurus (hierarchy, synonymous terms, associations)
- No shared typing schemes among gazetteers
- ADL Feature Type Thesaurus (online)
  - 1156 terms: 210 preferred terms and 946 non-preferred terms
  - Based on existing typing schemes and placenames themselves
- Goal: community adoption of typing schemes

Merging of data and attribution

- For a named geographic feature, merge information about it
- Allow multiple footprints, names, data, etc. from different sources and for different times
- Document the source of every piece of information
- Tucson example (ADL Gaz ID 600083 if Internet connection available)

Digital gazetteer information exchange

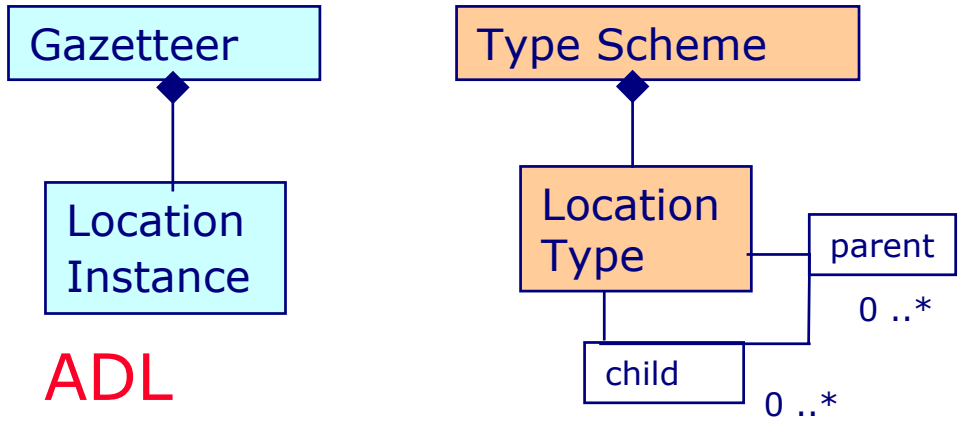
- Gazetteer data comes from many sources
- Being able to share this data would bring great benefits in richness of data
- What's needed for data exchange
  - A content standard – structure for documentation of information
  - An exchange format – XML version of the content standard
  - Shared typing schemes
- What's needed for interoperability among gazetteers
  - Gazetteer service protocol
    - **ADL draft in progress**
    - **OpenGIS protocol in progress**

ADL implementation

- 4.4 million entry global gazetteer – merging of the two federal gazetteers plus other entries
- Internet gazetteer service – worldwide usage
- Published components
  - Gazetteer Content Standard
  - Feature Type Thesaurus
  - XML DTD
- "Content Standard" approach instead of "thesaurus approach"
  - Geographic footprint required
  - Explicit statement of relationships among features optional

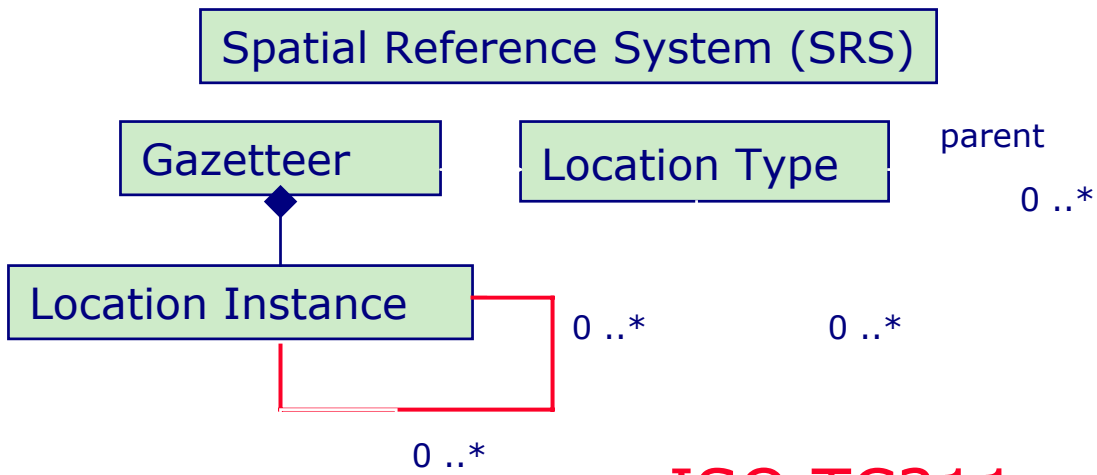
Contrasting structures

1. Uniqueness by ID
2. Gazetteer holds various types
3. Type schemes independent
4. Footprint required
5. Expressive description



ADL

1. Names are unique
2. Gazetteers are typed
3. Type scheme and gazetteer are packaged together
4. Footprint optional
5. Cryptic description
6. Gazetteer structured as a thesaurus



ISO TC211

Contrasting structure examples

Gazetteer Descriptions

Title ADL Gazetteer
Responsible Party ADL Project, UCSB
Scope & Purpose A gazetteer associates geographic names with geographic locations and other descriptive information. A gazetteer can …
Subject Coverage Worldwide
…

ADL

Sample Entries

Feature Name Cambridge (BGN-NIMA-1)
Feature Type populated places (ADL FTT)
Spatial Ref. –2,37,51.73 (BGN-NIMA-1)
Related Entity *IsPartOf* UTM grid WC43
Related Entity *IsPartOf* United Kingdom
Source BGN-NIMA-1: U.S. Board on Geographic Names, U.S. National Imagery and Mapping Agency, …

identifier Towns
scope large population centres
territory of use UK
custodian Ordnance Survey
coord. ref. sys. Nat Grid of Gr Brit
location type town

ISO TC211

geographic identifier Cambridge
temporal extent 19960401
alternative geographic identifier none
geographic extent 5414 2596, 5440 2532, 5493 2545, 5487 2598, 5455 2618
position 5448 2583
administrator Cambridgeshire County Council
parent location instance Cambridgeshire

ADL gazetteer protocol: goals

- Create published standard to support access to distributed gazetteer services
- Capture the essence of...
  - what a gazetteer *is*
  - what a gazetteer *does*
- Balance client needs *vs*. server burden
  - clients want functionality, uniformity, completeness
  - servers want minimal requirements, overhead
  - "non-preclusive simplicity" wins
- Accommodate differing implementations
  - semantics deliberately underspecified

Protocol: abstract gazetteer model

- Gazetteer = gazetteer entries + relationships
- Gazetteer entry
  - describes a single place
  - one entry per place
- Inter-entry relationships
  - Explicit: Sacramento is the “capital of” California
  - Implicit: geospatial relationships

Protocol: gazetteer entry

- Identifier
- Attributes
  - 1+ names
    - **unqualified, e.g., "San Diego"**
  - 1+ footprints
    - **region defined in WGS84 coordinates**
    - **not necessarily contiguous**
  - 0+ classes
    - **term drawn from vocabulary or thesaurus**
    - **city, park, mountain, lake, etc.**
- Attribute qualifiers
  - Primary (e.g., primary name or primary footprint)
  - Historical (e.g., historical name or historical footprint)

Protocol: services

Stateless, independent, synchronous functions

- get-capabilities() → *capabilities description*
  - which protocol features are supported
- query(*query*) → *reports*
  - returns all entries that match a query
- download() → *reports*
  - downloads entire gazetteer
- add-entry(*report*) → *identifier*
- relate-entries(*relationship*, $identifier_1$, $identifier_2$)
- remove-entry(*identifier*)

Protocol: query language

- Five fundamental constraint types...
  - identifier
    - **find gazetteer entry #314159**
  - name
    - **find "San Diego"**
  - footprint
    - **find places that overlap a given region**
  - class
    - **find place by type; e.g., cemeteries**
  - relationship
    - **find the capital of California**
- ...and boolean combinations thereof

Protocol technology

- In current version
  - XML
    - **XML schemas, XML namespaces, XML linking**
  - OpenGIS Geography Markup Language (GML)
  - HTTP
- Newest technologies for later implementation
  - SOAP (**S**imple **O**bject **A**ccess **P**rotocol)
  - WSDL (**W**eb **S**ervices **D**escription **L**anguage)

Protocol: Future directions/outstanding issues

- Seeking broad deployment
  - At least to the "rule of three": i.e., 3 implementations
- Qualification of names in queries
  - "Santa Barbara, CA"
- Relationships
  - codify specific relationships?
  - relationship types?
    - **topological, role, …**
- Extensions
  - if and how to enrich gazetteer protocol model
  - federation of gazetteers

Database implementation issues

- Issues
  - Database Size
  - Loading Issues
  - Indexing Issues
  - Real Query Issues

Gazetteer database size issues

- 4.4 million records
  - 5.9 million names associated with records
- 2 databases
  - Main for report production and data loading
    - **33 tables; generic types and indexing**
  - ADL bucket approach for searching
    - **7 tables**
    - **Uses object-oriented and spatial data types,**
    - **Uses clustered indexes, text indexes, and spatial indexes**

Gazetteer loading issues

- Large data loads can fill logs
  - Backup, split files that are being loaded, make logs larger
  - Turn off logging during loading
  - Turn off indexing during loading
- Know about database extents
  - Unload or copy to new table with extent defined large enough to hold data

Gazetteer indexing issues

- Indexing is the most important issue for performance
  - Corrupt indexes were a big problem, which was solved by reloading the database
- Text indexing
  - Original "blade" required more than 1 gigabyte ram to index gazbucket database
  - Multilingual: How do you handle it?
- Multiple types and custom datatypes complicate indexing
  - We cannot use parallel database features

Gazetteer query issues

- Real queries cause real problems
  - Hand-coded query optimizer being used
  - Generic query translator
    - **In general, much faster than hand-coded queries**
- Query of Death (generic query translator)
  - The query optimizer chooses the wrong path for queries using (text and spatial and type) constraints
  - Solution: submit with optimizer directives

Duplicate detection for gazetteers

- Premise: one entry for one place
- Problem:
  - Places have multiple names, types, and footprints
  - How, then, can duplicate entries for the same place be identified?
- Approach:
  - This is a "textual geospatial integration" problem
  - "Test record" is the query; result set is a ranked list of gazetteer entries, ranked according to their similarity to the "test record"
  - Tests include
    - **Source comparison (Are the records from the same contributor?)**
    - **Name comparison (Same primary names and/or variant names)**
    - **Type comparison (Same scheme? Same type?)**
    - **Spatial comparison (Spatial relationships according to footprint type)**

Example of duplicate detection

- **New record (incoming)**
  - Name: Paris
  - IsPartOf: Texas
  - Type scheme: Local
  - Type: PPL
  - Coords: -95.55,33.66
- **Existing record**
  - Name: Paris (county seat)
  - IsPartOf: Lamar County, Texas
  - Type scheme: ADL FTT
  - Type: populated places
  - Coords: -94,32 –96,34

Example test results (hypothetical scores)

Source comparison: 0.0 (sources are not the same)
Name comparison: 0.8 (partial but close match of primary names)
Type comparison: 0.8 (different schemes; types are similar)
Spatial comparison: 1.0 (point is contained within the box)
Rank value: 2.6

Duplicate detection technologies

- Text
  - Syntactic normalization of placenames (e.g., removing parenthetical phrases)
  - Information retrieval techniques for text similarity
  - Thesaurus techniques for related types
- Spatial
  - Spatial match types
    - **Polygon-to-polygon match (contains, overlaps)**
    - **Point-in-polygon match (contained within)**
      - Edge buffers where point near the edge of polygon
    - **Point-to-point match (nearness)**
  - Accuracy weighting (confidence in the coordinate values)
  - Visual checking (evaluating footprints displayed on a map)

ADL Gazetteer development

- Web page for all ADL Gazetteer developments is at www.alexandria.ucsb.edu/gazetteer

Concept Modeling and Education Applications

Terence R. Smith
Marcia Zeng

Concept-Based (CB) Learning Spaces: I

- Basic ideas:
  - **Scientific understanding based on large body of concept**
    - ``objective'' representations/interpretations
    - represent phenomena in terms of concepts/relations
    - example of mathematics
  - **Implicit treatment of concepts in learning environments**
    - No explicit model of a concept
      - Glossaries of terms at end of textbook chapters
      - Difficult for students to attain global conceptual views
  - **Structured representations of concepts**
    - Gardenfors models of concepts
      - Connectionist -> concept-level (MD spaces) -> symbolic
    - Concept level representations
      - structured model of concepts
      - inference-rich representations
  - **knowledge base (KB) of concepts as basis for teaching**

Scientific Semantics

- Key idea: scientific concepts **MUST** have:
  - **Informative representations**
  - **Semantics linking representations/phenomena**
- Scientific semantics derived from:
  - **Operations linking concepts and phenomena**
  - **Operations relating to symbolic representations**

Concepts: Classification

- Basis
  - **Classification of concepts by role in scientific activities**
  - **Different operational semantics for each class**
- Concept classes
  - **Abstract (interpretable in terms of syntax)**
    - First order predicate logic concepts
    - Arithmetic concepts
  - **Concrete (interpretable in terms of world phenomena)**
    - measurable concepts
    - recognizable concepts
    - Concretely interpreted abstract concepts
    - Interpreted abstract concepts
  - **Methodological (interpretable in terms of scientific activities)**
    - Observation concepts
    - modeling concepts
    - Communication concepts
    - hypothesis testing concepts
    - Interpreted abstract

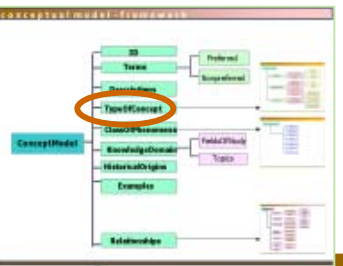
classification of concepts

- **type of concept**
  - **abstract**
    - syntactic (linguistic)
    - logical
    - mathematical
  - **methodological**
    - identification/ characterization
    - representation
    - understanding
    - application
    - communication
  - **concrete**
    - measurable
    - recognizable
    - interpreted abstract

- observations
- measures
- analysis
- examples

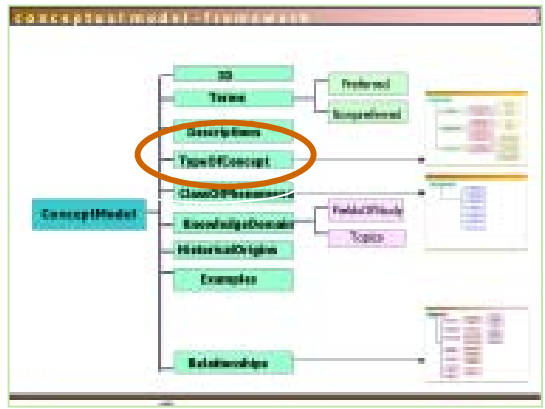
- topics
- concepts
- models

- questions/answers
- problems/solutions
- hypotheses/evaluations
- predictions/tests
- statements/deviations
- applications/evaluations
- facts/validations



classification of concepts

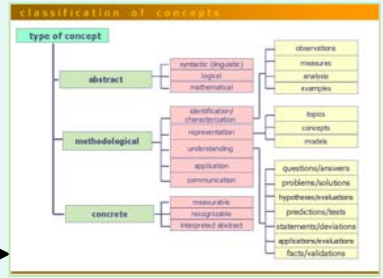
- class of phenomena
  - object
  - material
  - process
  - form
  - event
  - state
  - … …



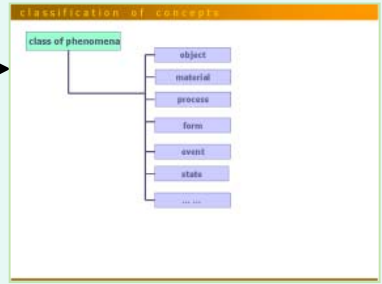
Concepts: structured representations

- ID
- TERM(S)
- DESCRIPTION(S)
- TYPE
- CLASS OF PHENOMENA
- KNOWLEDGE DOMAINS
- HISTORICAL ORIGIN(S)
- EXAMPLES
- RELATIONS TO OTHER CONCEPTS
  - **HIERARCHIES**
  - **SCIENTIFIC USES**
  - **REPRESENTATIONS**
    - Explicit full/Explicit partial/Implicit full/Implicit partial
  - **DEFINFINING OPERATIONS**
  - **PROPERTIES**
  - **CO-RELATIONS**
  - **CAUSALITY**
  - **OTHER**

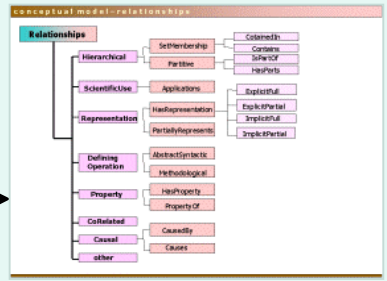
- **ConceptModel**
  - **ID**
  - **Terms**
    - Preferred
    - Nonpreferred
  - **Descriptions**
  - **TypeOfConcept**



  - **ClassOfPhenomena**

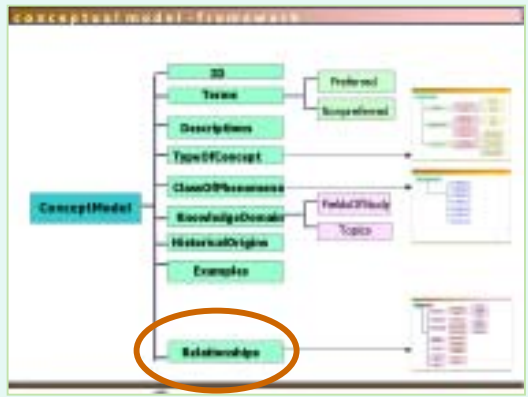


  - **KnowledgeDomain**
    - FieldsOfStudy
    - Topics
  - **HistoricalOrigins**
  - **Examples**
  - **Relationships**



conceptual model–relationships

- **Relationships**
  - **Hierarchical**
    - SetMembership
      - CotainedIn
      - Contains
    - Partitive
      - IsPartOf
      - HasParts
  - **ScientificUse**
    - Applications
  - **Representation**
    - HasRepresentation
      - ExplicitFull
      - ExplicitPartial
      - ImplicitFull
      - ImplicitPartial
    - PartiallyRepresents
  - **Defining Operation**
    - AbstractSyntactic
    - Methodological
  - **Property**
    - HasProperty
    - PropertyOf
  - **CoRelated**
  - **Causal**
    - CausedBy
    - Causes
  - **other**



Partial Concept Example: *Plane polygon*

- TERM(S): *plane polygon, (polygon, n-gon)*
- TYPE: *mathematical (abstract) concept*
- RELATIONS TO OTHER CONCEPTS
  - **HIERARCHIES:**
    - *Contains Triangles; ContainedIn Closed Bounded Connected Pointsets*
  - **REPRESENTATIONS**
    - Explicit full: *Tuple of Plane Points [P1,…,Pn]*
      Tuple of Line segments [Li,…,Ln]
    - *Implicit full: Intersection of Set of Half-Planes {HP1,…,HPn}*
  - **DEFINING OPERATIONS: *construction, intersection,…***
  - **PROPERTIES: *Area (+ equation/algorithm for area in terms of some representation), Circumference (+ equation/algorithm for its computation in terms of some representation)***

Concepts: structured representations

- ID
- TERM(S)
- DESCRIPTION(S)
- TYPE
- CLASS OF PHENOMENA
- KNOWLEDGE DOMAINS
- HISTORICAL ORIGIN(S)
- EXAMPLES
- RELATIONS TO OTHER CONCEPTS
  - **HIERARCHIES**
  - **SCIENTIFIC USES**
  - **REPRESENTATIONS**
    - Explicit full/Explicit partial/Implicit full/Implicit partial
  - **DEFINFINING OPERATIONS**
  - **PROPERTIES**
  - **CO-RELATIONS**
  - **CAUSALITY**
  - **OTHER**

Support for CB Learning Spaces

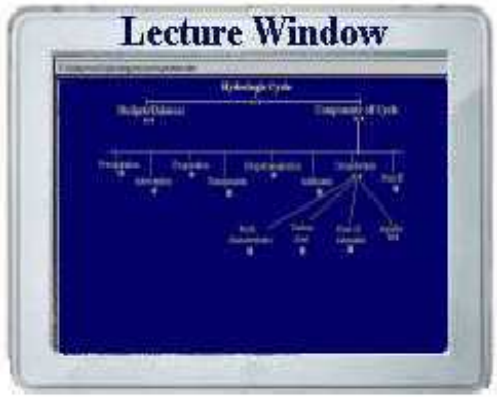
- Organized collections
  - **KBs of structured concept representations**
  - **Collections of illustrative materials**
    - Accessible by concept/property
  - **Collections of lectures/self-learning materials**
    - Lecture as trajectory through concept space
- Services
  - **Creation/editing**
    - KBs/collections/lectures
  - **Search over KB/collections/lectures**
    - Pre-stored lectures
    - Real-time access to KBs/collections
  - **Graphic/textural views of KB/collections/lectures**
    - ‘’concept map” views of topics
    - Views of illustrative material
    - Views of lectures

Application to learning environments

- Course as a trajectory (personalized collection)
  - **through space of concepts**
  - **through space of illustrative materials**
- Instructor provides framework
  - **Motivations with topics**
  - **Set of topic related issues**
  - **Representing/manipulating representations of issues**
    - Using concepts/illustrative material
- Three concurrent hierarchically-organized views
  - **KB items**
  - **Collection items**
  - **Course/lecture structure**
- Static (trajectory) and dynamic (real-time access) views


http://alexandria.ucsb.edu/~ushakov/A...
File Edit View Favorites Tools Help
Done
Internet

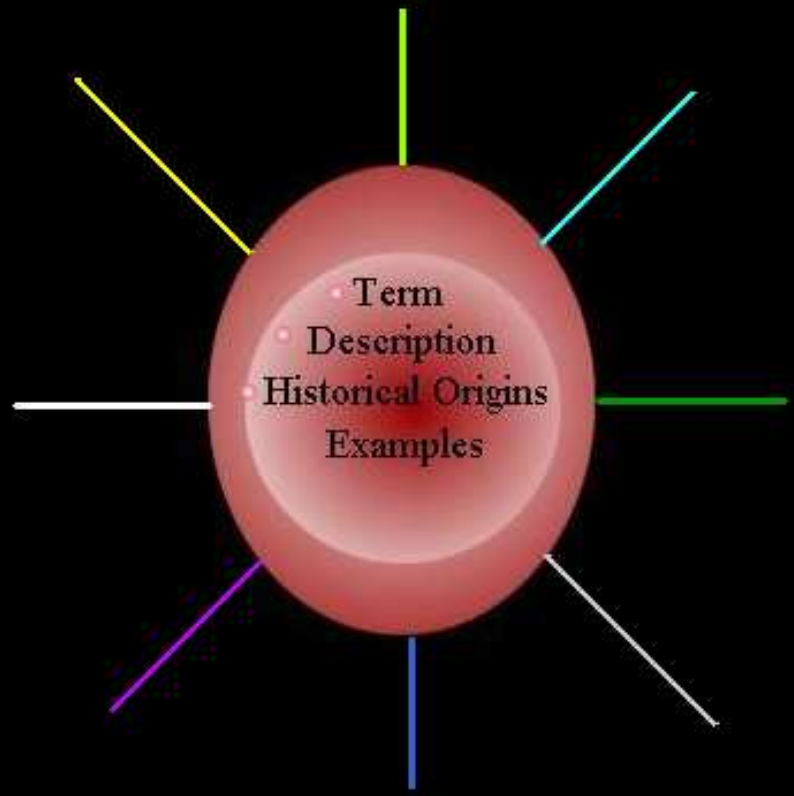
Address http://alexandria.ucsb.edu/~ushakov/Adeptvisudemo/present13.htm

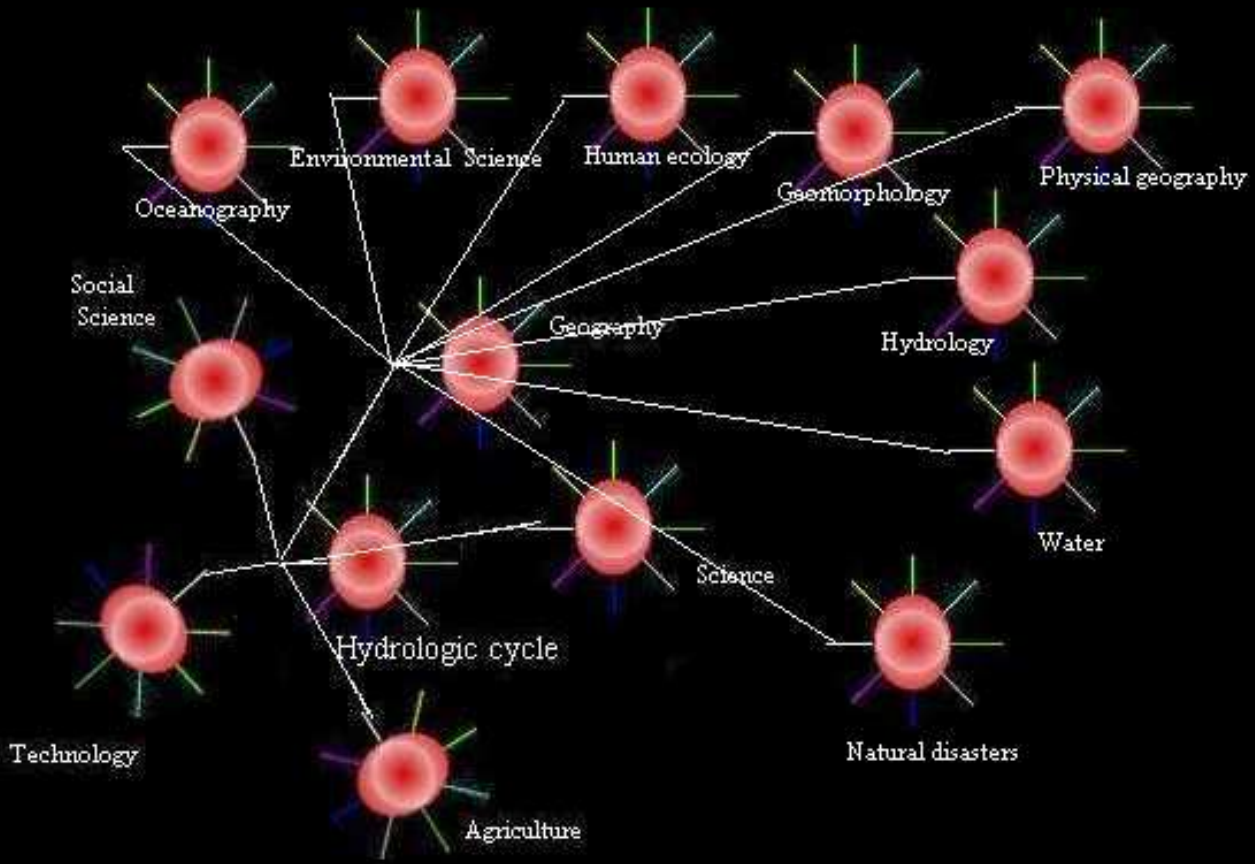


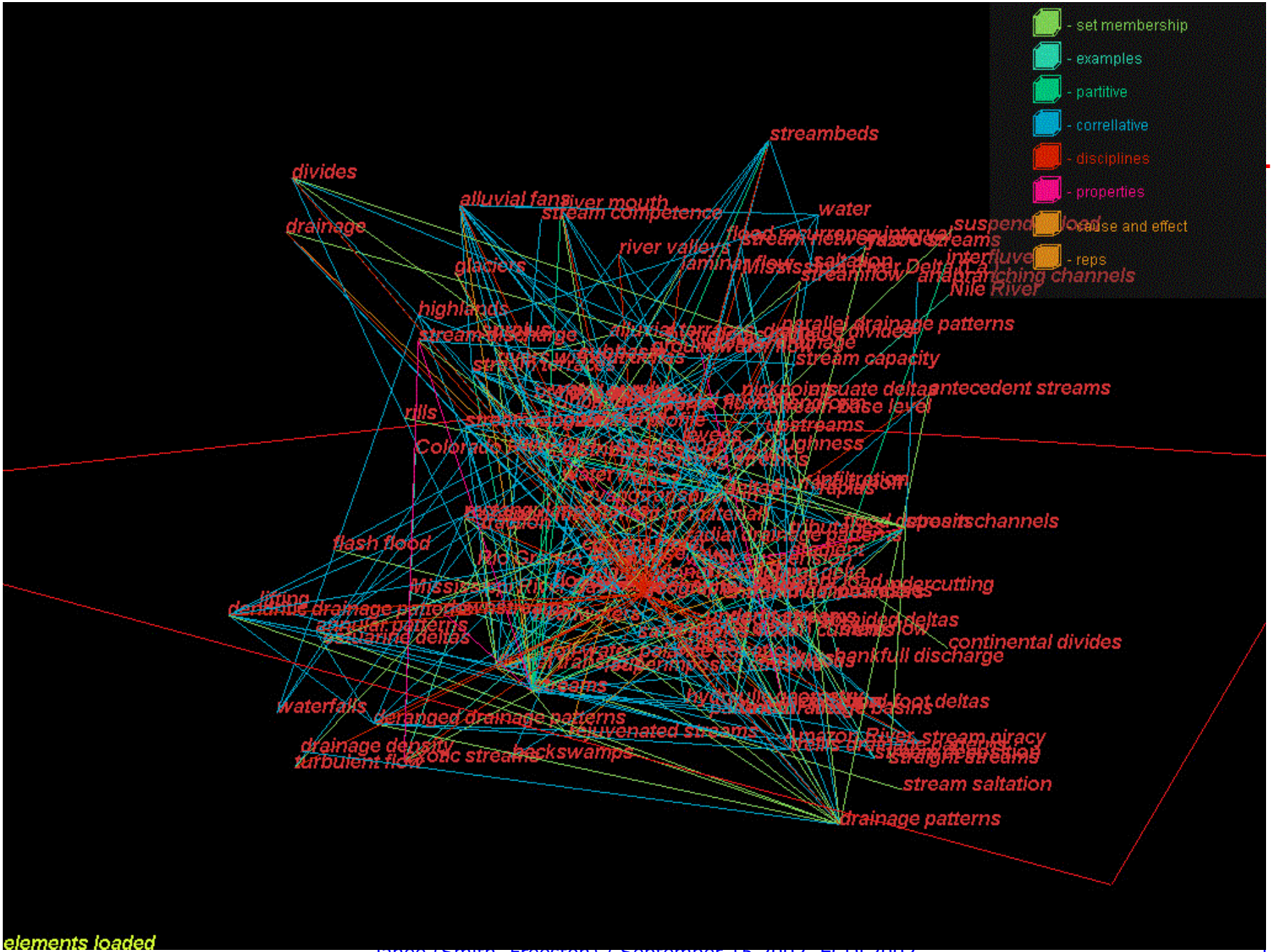
Next

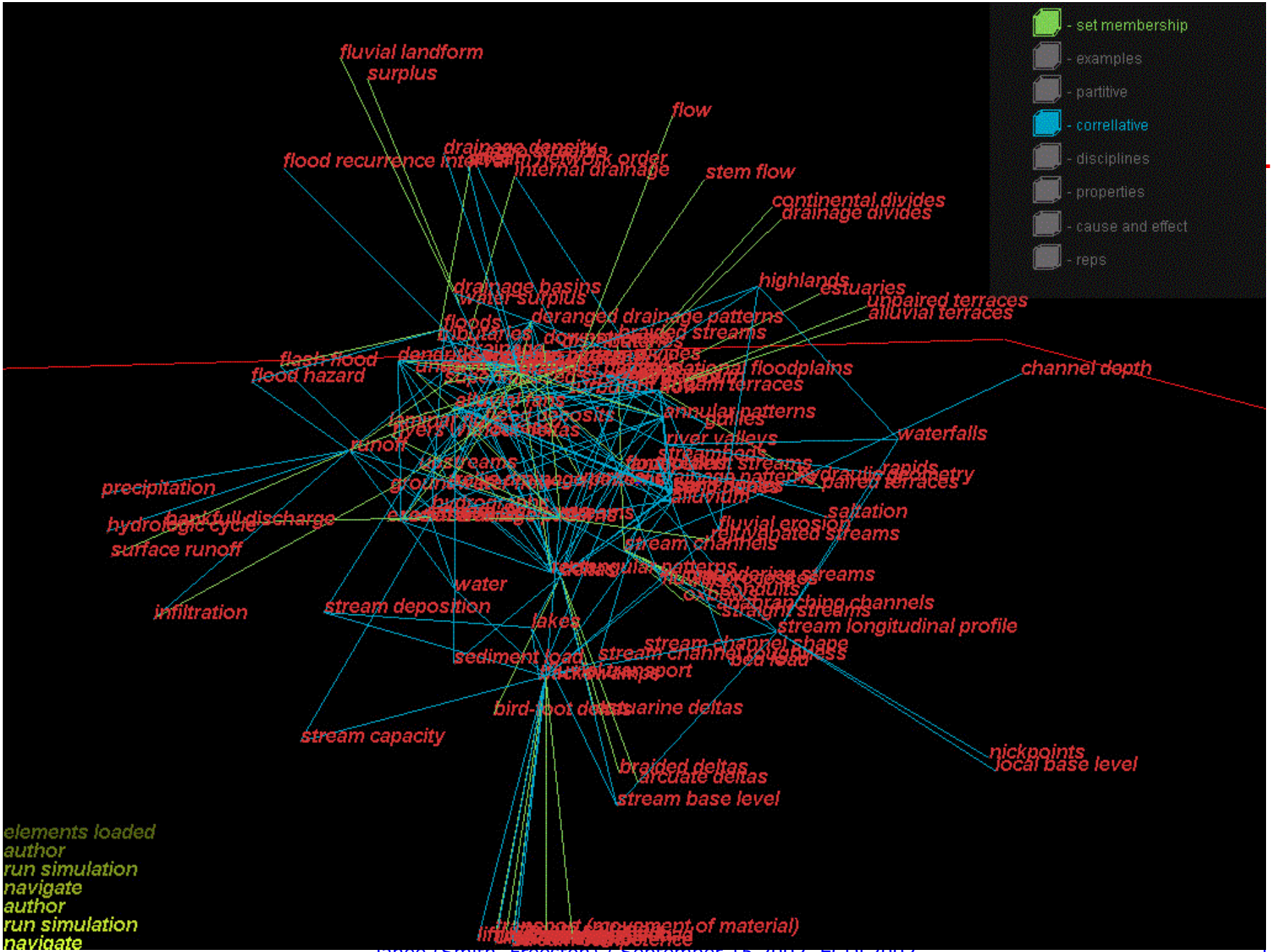
First Component: KB of Concepts

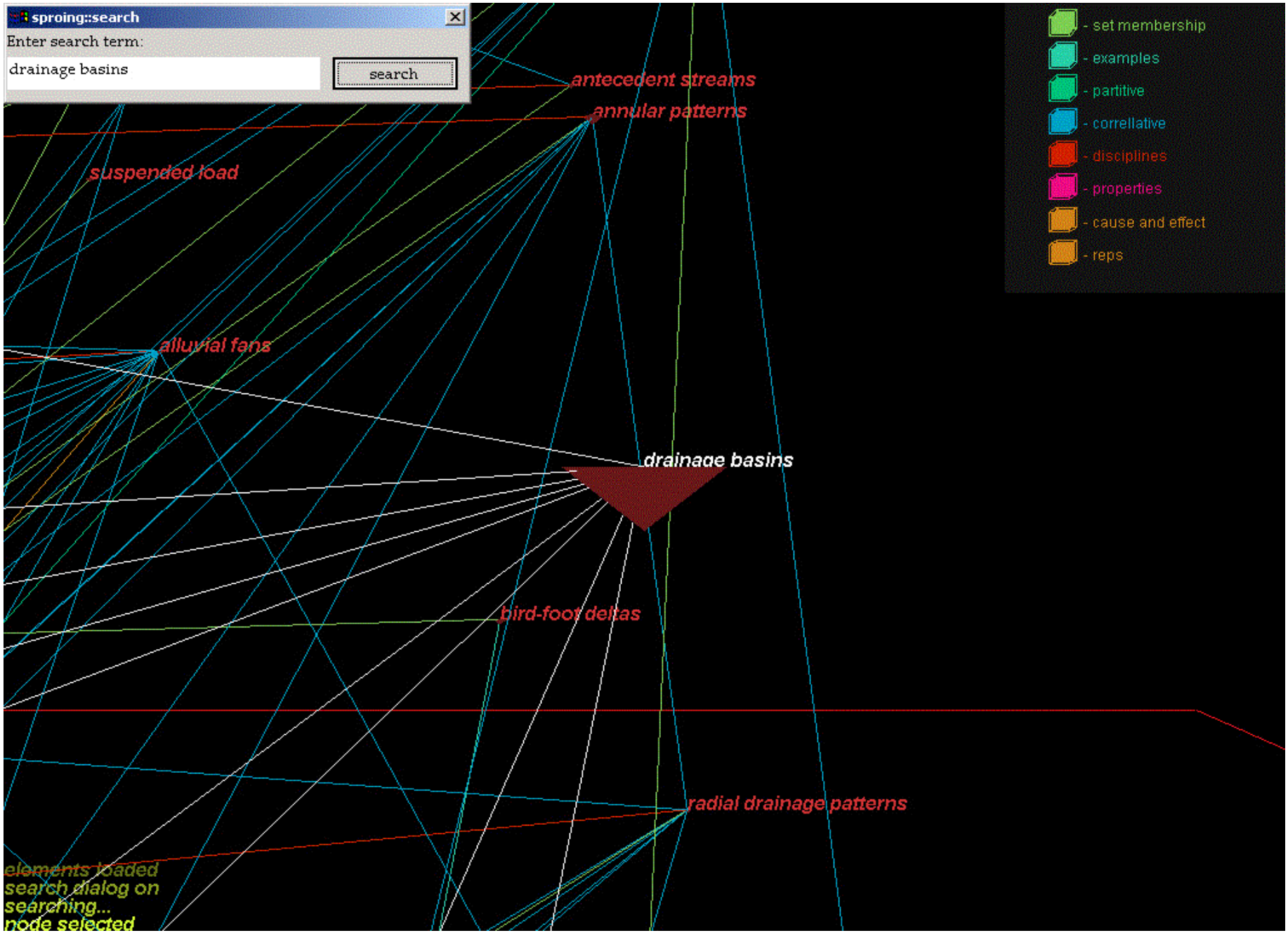
- Structured model(s) of concepts
  - **Abstract model**
  - **XML schema**
  - **DB schema**
- Organization of concepts
  - **Files of XML records of concept representations**
    - Access using commercial support
  - **RDBMS**
  - **Logical (deductive) databases**
- Services
  - **Concept entry/modification/…**
  - **Search/access**
  - **Concept space visualization**
  - **Automated KOS generation (e.g., thesauri)**

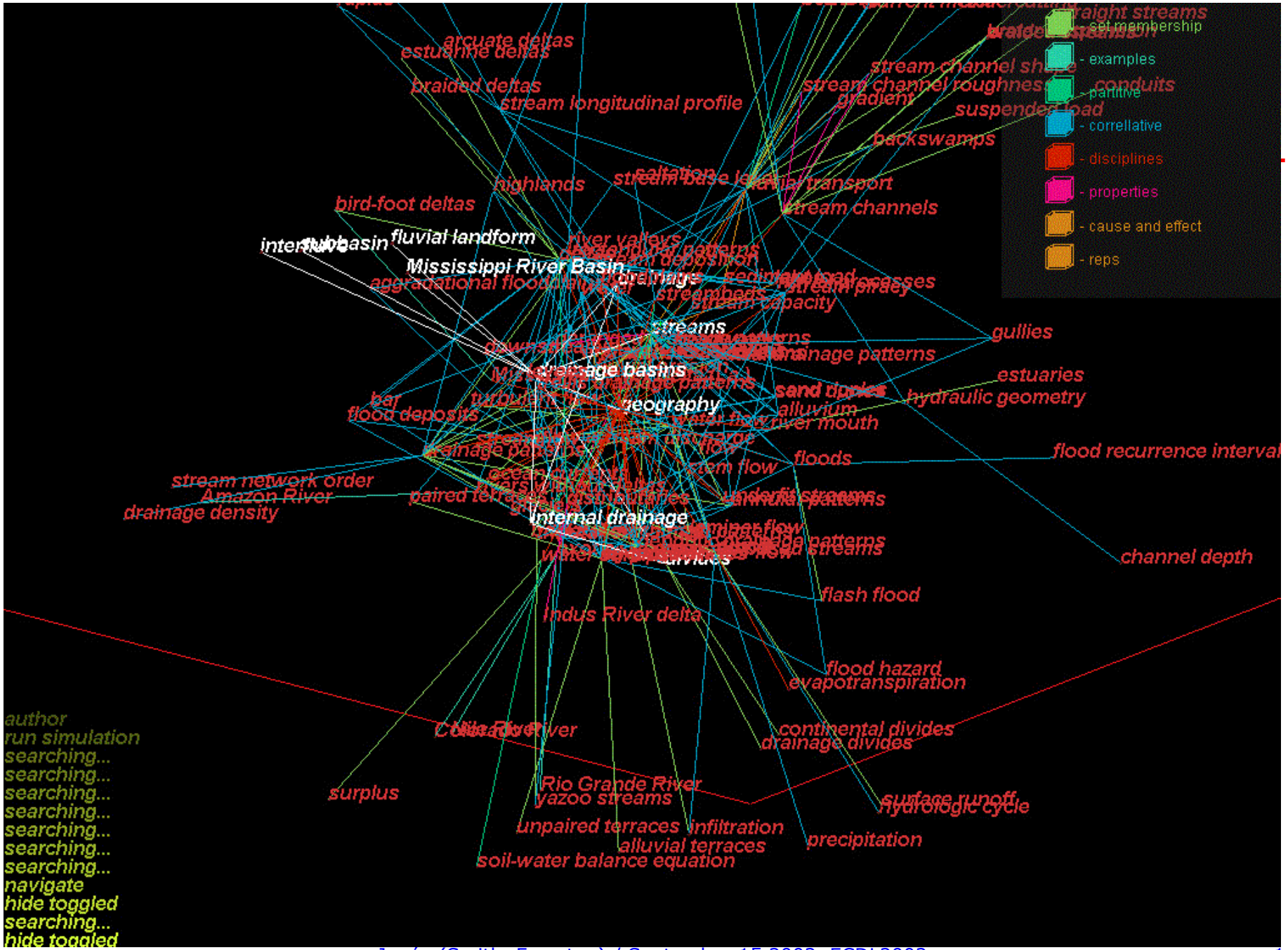

Term
Description
Historical Origins
Examples

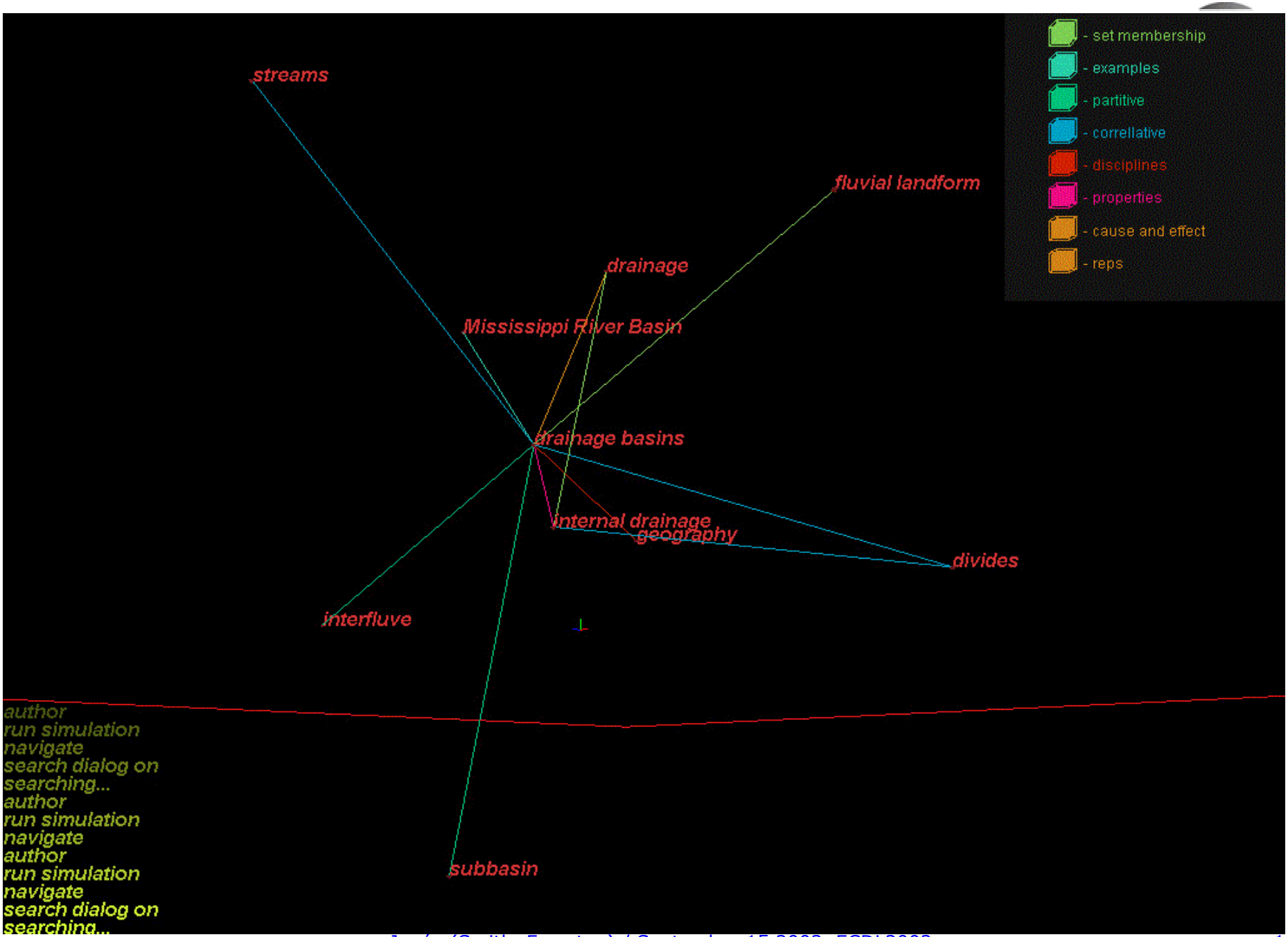

Oceanography
Environmental Science
Human ecology
Geomorphology
Physical geography
Social Science
Geography
Hydrology
Water
Science
Hydrologic cycle
Technology
Natural disasters
Agriculture


- set membership
- examples
- partitive
- correllative
- disciplines
- properties
- cause and effect
- reps
streambeds
divides
alluvial fans
river mouth
stream competence
water
drainage
suspended load
river valleys
glaciers
Nile River
highlands
parallel drainage patterns
stream capacity
antecedent streams
rills
base level
flash flood
continental divides
bankfull discharge
waterfalls
deranged drainage patterns
drainage density
straight streams
stream saltation
drainage patterns
elements loaded


- set membership
- examples
- partitive
- correllative
- disciplines
- properties
- cause and effect
- reps
fluvial landform
surplus
flow
flood recurrence interval
internal drainage
stem flow
continental divides
drainage divides
drainage basins
water surplus
highlands
estuaries
unpaired terraces
alluvial terraces
floods
deranged drainage patterns
braided streams
flash flood
flood hazard
channel depth
floodplains
terraces
annular patterns
gullies
waterfalls
runoff
river valleys
streamheads
rapids
precipitation
paired terraces
saltation
hydrologic cycle
bankfull discharge
fluvial erosion
rejuvenated streams
surface runoff
stream channels
water
infiltration
stream deposition
lakes
straight streams
stream longitudinal profile
stream channel shape
sediment load
bed load
bird-foot deltas
estuarine deltas
stream capacity
nickpoints
local base level
braided deltas
arcuate deltas
stream base level
elements loaded
author
run simulation
navigate
author
run simulation
navigate


sproing::search
Enter search term:
drainage basins
search
antecedent streams
annular patterns
suspended load
alluvial fans
drainage basins
bird-foot deltas
radial drainage patterns
- set membership
- examples
- partitive
- correllative
- disciplines
- properties
- cause and effect
- reps
elements loaded
search dialog on
searching...
node selected


set membership
- examples
- partitive
- correllative
- disciplines
- properties
- cause and effect
- reps
arcuate deltas
estuarine deltas
braided deltas
stream longitudinal profile
stream channel roughness
gradient
conduits
suspended load
backswamps
highlands
saltation
stream base level
fluvial transport
stream channels
bird-foot deltas
fluvial landform
river valleys
Mississippi River Basin
aggradational floodplain
streambeds
stream capacity
streams
gullies
drainage basins
geography
estuaries
hydraulic geometry
bar
flood deposits
alluvium
floods
flood recurrence interval
stream network order
Amazon River
drainage density
paired terraces
internal drainage
channel depth
flash flood
Indus River delta
flood hazard
evapotranspiration
continental divides
drainage divides
surplus
Rio Grande River
yazoo streams
surface runoff
hydrologic cycle
unpaired terraces
infiltration
alluvial terraces
precipitation
soil-water balance equation
author
run simulation
searching...
searching...
searching...
searching...
searching...
searching...
searching...
navigate
hide toggled
searching...
hide toggled


- set membership
- examples
- partitive
- correllative
- disciplines
- properties
- cause and effect
- reps
streams
fluvial landform
drainage
Mississippi River Basin
drainage basins
internal drainage
geography
divides
interfluve
subbasin
author
run simulation
navigate
search dialog on
searching...
author
run simulation
navigate
author
run simulation
navigate
search dialog on
searching...

Second Component: Collection of Docs

- Multi-media documents illustrating concepts/elements
- Metadata for objects forms ADEPT collection
  - **Alexandria/DLESE/NASA metadata content standard for learning objects**
  - **Additional element for concepts/elements**
  - **Current hand entry**
- Initial collections
  - **Illustrations from electronic version of text book**
  - **Materials from web**
- Services
  - **Entry tools for document metadata**
  - **Search services over document metadata**
    - Search/retrieval over docs by concept information
  - **Visualization of doc contents**
  - **Visualization of document relationships**

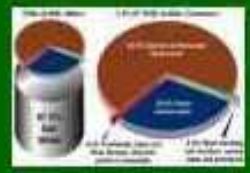
Concept Name: *Components of cycle*

ADL Search on/of

Next

Learning Objects

Zoom



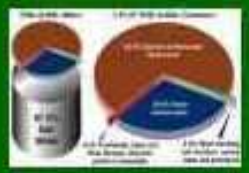
Concept Name: **Components of cycle**

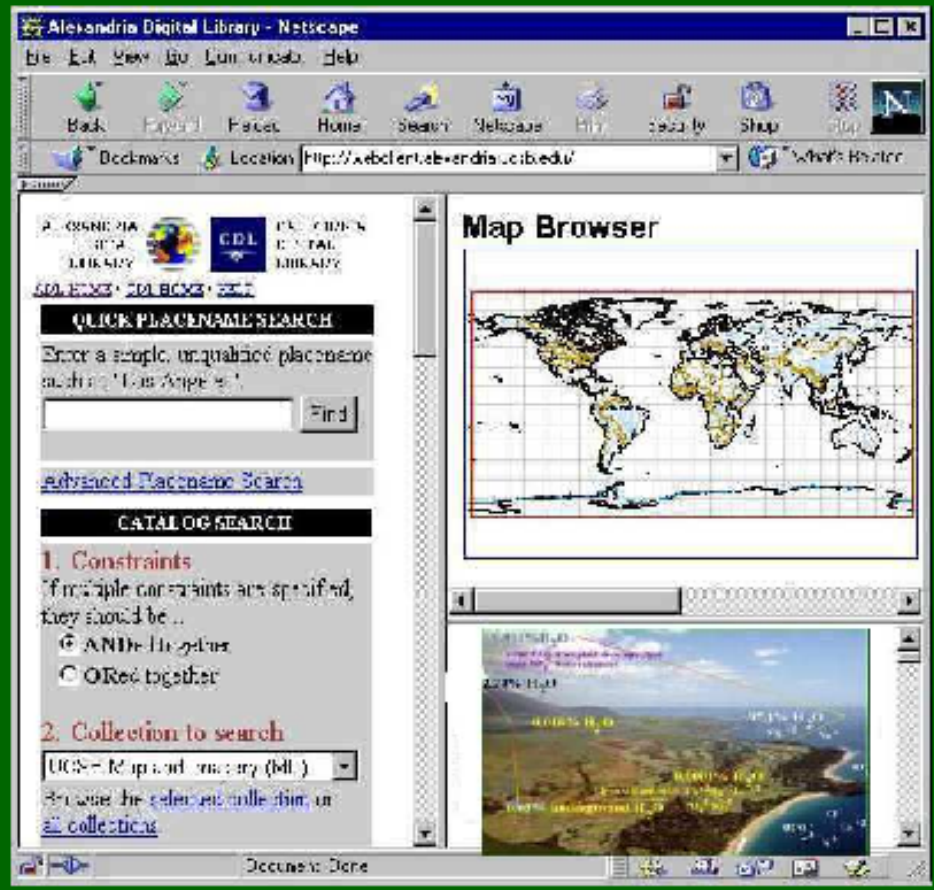
ADL Search on/of

Next

Learning Objects

Zoom





Third Component: Lecture Collection

- Structured model(s) of lectures
  - **Generic**
    - Arbitrarily indented lists
    - Addition of links to items
  - **Personalized**
    - e.g., using aspects of previous concept classification
- Services
  - **Lecture creation/modification/…**
    - Inclusion of KB/collections items
  - **Storing lectures as items in DL collection**
  - **Search/access**
  - **Lecture visualization**
    - Individual lectures
    - High level view of lectures

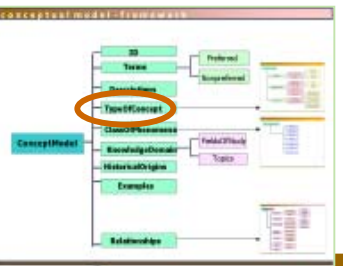
classification of concepts

- **type of concept**
  - **abstract**
    - syntactic (linguistic)
    - logical
    - mathematical
  - **methodological**
    - identification/ characterization
    - representation
    - understanding
    - application
    - communication
  - **concrete**
    - measurable
    - recognizable
    - interpreted abstract

- observations
- measures
- analysis
- examples

- topics
- concepts
- models

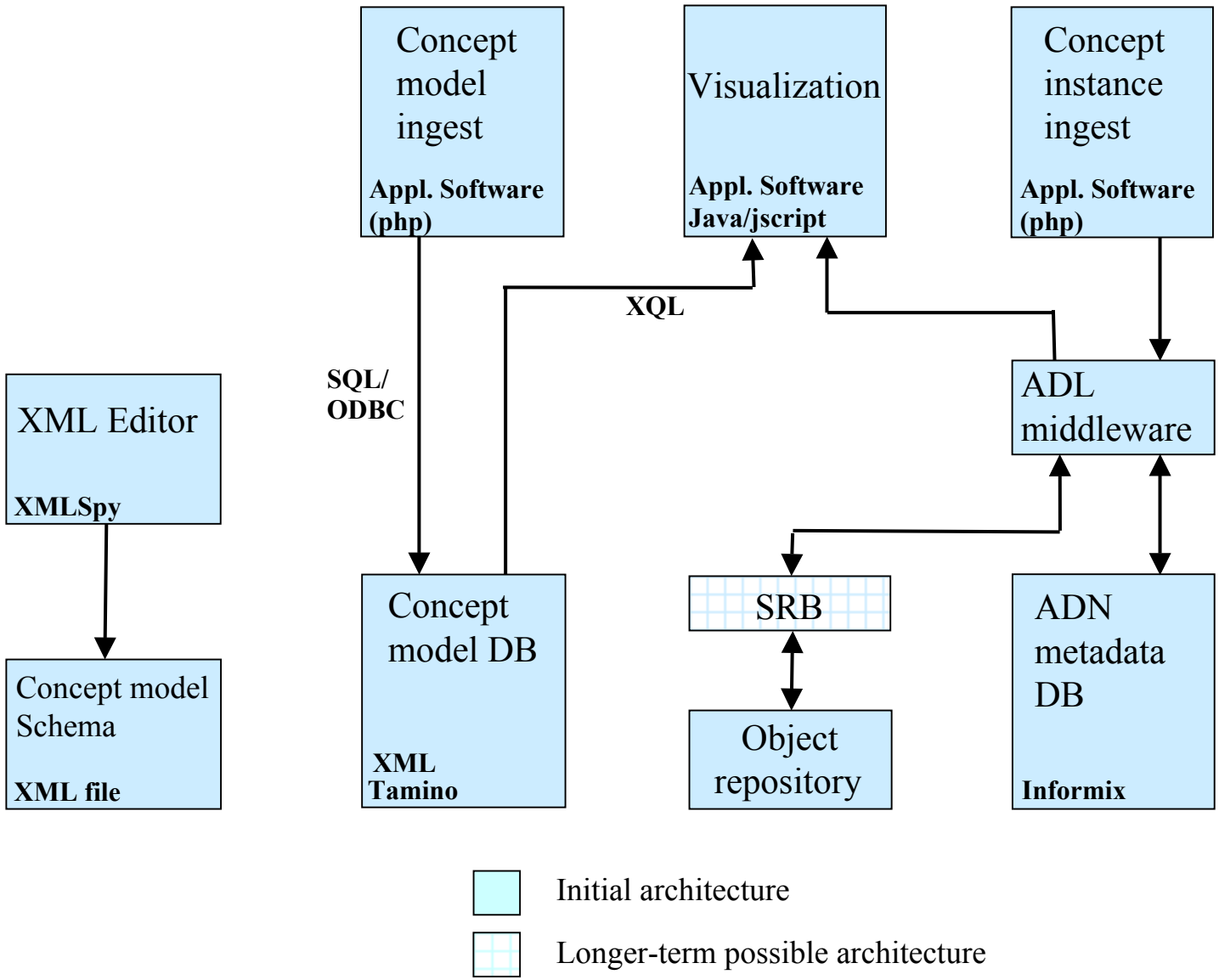
- questions/answers
- problems/solutions
- hypotheses/evaluations
- predictions/tests
- statements/deviations
- applications/evaluations
- facts/validations



SHOW: Concept, item, lecture inputs

- Concept input form
- Item metadata input form
- Cataloged record
- Lecture structure example

ADEPT Concept Base Architecture version 3: objective for March 31 2002



M Freeston 14 Feb 2002

Application and evaluation

- Application in teaching introductory physical geography
  - **Earlier ``prototype'' lectures**
  - **Fall 2003 Geography 3B (100 students)**
  - **Full electronic versions of text**
- Evaluation of efficacy wrt student learning
  - **Do students attain ``deeper levels'' of understanding?**
  - **Evaluation team activities**
    - Comparison approach
- Evaluation of value to instructors/Tas
  - **UCLA evaluation team**

SUMMARY

- ADEPT overview
- Core library architecture
  - Metadata interoperability
  - Query translation
  - Collection discovery
- Gazetteers and their application
- Concept modeling & educational applications