Search and Retrieval Web Services

John Weatherley <jweather@ucar.edu>
Shelley Olds <seolds@ucar.edu>
University Corporation for Atmospheric Research

Workshop presented at the NSDL Annual Meeting, 2004
Overview of Topics

- Web services defined
- Example applications that have been built using a search and retrieval Web service
- Detailed look at the DLESE search and retrieval Web service
- Example creation of a search Web page
What are Web services?

- Web services are building blocks for creating open distributed systems, and allow organizations and individuals to quickly and cheaply make their digital assets available worldwide.

- A web service is a collection of functions that are packaged as a single entity and published to the network for use by other programs.
Features of web service technology

• A web service is a software application identified by a URI, whose interfaces and bindings are capable of being defined, described, and discovered as XML.

• A web service supports direct interactions with other software applications using XML based messages exchanged via Internet-based protocols.
Features of Search and Retrieve Web Services

- Perform textual searches over library collections
- Limit searches by controlled vocabularies
- Display the search results
- Page through the results
- Discover the available controlled vocabularies (grade range, subject, resource type, content standard and collection)
Features of a Search and Retrieval Web Service Client

- Display resources by pre-defined category
- Search by entering text
- Select controlled vocabularies
- Display resource metadata
- Page through search results
Benefits of Web Service Architectures

- Provides instructional designers and content developers with global access to metadata and search functionality
- Enables the creation of custom learning environments that incorporate library resources
- Enables external partners to provide a custom view of the library for their constituents
- Enables others to access library data for analysis
- Decouples the search functionality from UI designs and behaviors
- Enables a clean way to build prototypes for testing UI designs and Information Retrieval algorithms
Types of Web Services

- REST
- SOAP/WSDL
REST-Based Web services

- REST – REpresentational State Transfer
- In REST style Web service architectures, requests are typically encoded as a URL and responses are returned as XML
  - All state is encapsulated in the HTTP GET or POST parameters found in the request URL
  - Relatively simple for humans to interpret and explore using available tools like Web browsers and text editors
- Examples: OAI, RSS, SRU (Z39.50), ODL, DDSWS, Dublin Core Registry
- NSDL – REST search service scheduled for release in January, 2005 (to replace current SDLIP and WebDAV)
SOAP/WSDL Web services

- SOAP - Simple Object Access Protocol
- WSDL - Web Services Description Language
- SOAP/WSDL are typically used together
- SOAP/WSDL architectures feature:
  - Requests and responses are supplied in XML
  - Strict data binding and data definitions
  - Robust machine-to-machine communication and data transfer
  - More difficult for humans to explore and interpret directly
- Example: SRW (Z39.50), Dublin Core Registry
Example: DDSWebService at DLESE

- DDSWebService – DLESE’s search and retrieval Web service
- REST-based
- Shares design elements found in OAI and the ODL search protocol
DLESE Web Service Architecture

Library user search
- Web browser

Discovery Engine
- Conducts search over the index
- Ranks and returns records

Index

Web service search
- Custom User Interfaces
- Custom applications

Index is accessed for resource discovery and metadata retrieval

XML is read to generate the index

Metadata about resources

HTML

XML
What is Available from DDSWebService?

- Metadata about
  - Earth science educational resources
  - Annotations for resources
  - Earth science collections by theme and institution
  - News and opportunities in the Earth sciences
- Vocabularies that describe grade range, Earth science subjects, educational resource types, and content standards
- Information about relationships among metadata
  - Which annotations are associated with a resource
  - Which collections catalog the same resources
- Information Retrieval (IR) search and discovery functions
  - Textual and field-based retrieval using Boolean operators, support for advanced IR features
  - Full control over IR search algorithms
Examples: Three Projects that use the DLESE Web Services
The service is used to...

FCOSEE – contextualized resource discovery for ocean science education

- Perform textual search
- Return resources by subject
- Display the results
- Page through the results
- View expanded metadata
- Limit the domain to the ocean sciences and boost Florida-related resources automatically
IdeaKeeper – a tool to create scaffolding around resources

- Students articulate a research question
- Perform a search to find relevant resources
- Attach notes to the resource (not shown)
- Publish their project for others to view
- Choose resources of interest
- Formulate a formal argument
- Publish their project for others to view
The service is used to...

IdeaKeeper – a tool to create scaffolding around resources

- Perform textual search
- Display the results
- Limit search by grade range
The service is used to...

**Strand Map Browser** – browse earth science concepts to find resources

Return resources aligned to concepts

Users navigate using an interactive concept map browser
DDSWebService Protocol Overview

• DDSWebService has eleven requests, known as *verbs*
• Requests and their arguments are sent as HTTP parameters
• Responses are returned in a DDSWebService XML response envelope
Example request

- The service URI is the **BASE URL**: http://www.dlese.org/dds/services/ddsws1-0
- The request verb (required) and additional arguments are added to the BASE URL:
  http://www.dlese.org/dds/services/ddsws1-0
  ?verb=GetRecord&id=DLESE-000-000-000-001

verb argument indicates the request:
  GetRecord

Additional arguments:
  id
Example response

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<DDSWebService>
  <GetRecord>
    <record>
      <head>
        <id>DLESE-000-000-000-001</id>
        <collection recordId="DLESE-COLLECTION-000-000-000-015">DLESE Community Collection (DCC)</collection>
        <xmlFormat>adn</xmlFormat>
        <fileLastModified>2004-06-24T19:06:07Z</fileLastModified>
        <whatsNewDate type="itemnew">2003-07-10</whatsNewDate>
      </head>
      <additionalMetadata realm="adn">
        <accessionStatus>accessioneddiscoverable</accessionStatus>
        <partOfDrc>false</partOfDrc>
      </additionalMetadata>
    </record>
  </GetRecord>
</DDSWebService>
```

The XML elements that appear in the response vary by request.

The goal of this course is to introduce participants to the basic concepts of meteorology and the quality necessary to understand meteorological computer models. This course, along with companion courses in Computational Atmospheric Science and Emissions, anticipates the Models3 environment under development by the U.S. Environmental Protection Agency (EPA). Upon completion of the course,
# List of requests

Requests related to search and retrieval

<table>
<thead>
<tr>
<th>Request</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search</td>
<td>Supports searching over all metadata and content. Allows arbitrary textual and field-based Boolean queries and full control over IR search algorithms</td>
</tr>
<tr>
<td>UserSearch</td>
<td>Supports end-user style search over educational resources in the library. Incorporates the default search algorithm (stemming, boosting, etc.) that is applied to user’s searches in the DLESE library Web site</td>
</tr>
<tr>
<td>GetRecord</td>
<td>Retrieves a single metadata record by ID</td>
</tr>
<tr>
<td>ListXmlFormats</td>
<td>Lists the XML formats that may be returned by the UserSearch, Search and GetRecord requests</td>
</tr>
</tbody>
</table>
List of requests

Vocabulary List requests

- Used to discover available library vocabularies, their search field and keys, UI labels and rendering guidelines
- Each share the same response format

<table>
<thead>
<tr>
<th>Request</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ListCollections</td>
<td>Lists the metadata collections that are available in the library</td>
</tr>
<tr>
<td>ListGradeRanges</td>
<td>Lists the grade ranges (audiences) that are available in the library</td>
</tr>
<tr>
<td>ListResourceTypes</td>
<td>Lists the resource types that are available in the library</td>
</tr>
<tr>
<td>ListSubjects</td>
<td>Lists the Earth science subject areas that are available in the library</td>
</tr>
<tr>
<td>ListContentStandards</td>
<td>Lists the content standards that are available in the library</td>
</tr>
</tbody>
</table>
## List of requests

### Additional requests

<table>
<thead>
<tr>
<th>Request</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>UrlCheck</td>
<td>Determines whether a resource URL is cataloged in the library</td>
</tr>
<tr>
<td>ServiceInfo</td>
<td>Provides general information about the web service</td>
</tr>
</tbody>
</table>
Use Case Scenario: A Simple Search

- **Actions:**
  1. User performs a search for the term *oceans*
  2. User pages through to the second page of results
- **Client implementation:** Uses the Search or UserSearch request
  - The request argument `q` indicates the user’s query *oceans*
  - The `s` and `n` arguments indicate the client state as it pages through the results – `s` indicates the starting position within the set of matching results, `n` indicates the number of results the service will return
  - The `xmlFormat` argument indicates the realm in which the service will search and the format it will return
- Example continued on the next two pages...
Use Case Scenario: Simple Search – First page of Results

http://www.dlese.org/dds/services/ddsws1-0?
verb=Search&q=oceans&s=0&n=10&xmlFormat=adn-localized

Request sent to the service: Search
Argument: q = query
Argument: s = start offset
Argument: n = num results
Argument: xmlFormat = format to return

XML used to extract metadata for display

Visible Earth
http://visibleearth.nasa.gov/

NASA's Visible Earth is a searchable directory of images, visualizations, and animations of the Earth. The images are also listed under the following categories: agriculture, atmosphere, biosphere, cryosphere, human dimensions, hydrosphere, land surface, oceans, radiance or imagery, solid earth, locations, and satellites. Accompanying each image are credits, data about the image, the satellite... Full description

Submit a comment or teaching tip for this resource.
Use Case Scenario: Simple Search – Second page of Results

http://www.dlese.org/dds/services/ddsws1-0?
verb=Search&q=oceans&s=10&n=10&xmlFormat=adn-localized

\( s = \text{start offset} \)
Incremented to 10 for the second page of results

All other arguments remain the same

Search the Digital Library for Earth System Education (DLESE) - Microsoft Internet Explorer

Search results
Results 11 - 20 out of 232 for 'oceans'

Oceans of Kansas
http://www.oceansofkansas.com/index.html

Oceans of Kansas is the unofficial, but highly useful, web page of the Sternberg Museum of Natural History. To find content, scroll to the middle of the page. The website contains information on exhibits, articles with photographs of fossils and paintings about marine reptiles and fish who flourished in the Western Interior Sea. The site also contains an online collection of full-text paleontology ... Full description

Submit a comment or teaching tip for this resource.
Use Case Scenario: Simple Search with Query Augmentation

- **Actions:**
  1. User performs a search for the term *oceans*
- **Client implementation:** Uses the Search request to apply query augmentation. Query augmentation is implemented using the IR functionality that is available in the Search request
  - The Search request argument ‘q’ is used to indicate the user’s query with query augmentation applied
  - Example: Augment the user’s query to apply word stemming and assign greater weight to records with matching terms in their title field:
    \[q=(stems:oceans \text{ OR } titlestems:oceans)\]

  Indicate to search for oceans in the *stems* field. This will match records that contain *ocean*, *oceans* or *oceanic* in their title, description, etc.

  Apply a second clause to search in the *titlestems* field. This adds additional weight for records that contain *ocean*, *oceans* or *oceanic* in their title.
Use Case Scenario: Simple Search with Query Augmentation

http://www.dlese.org/dds/services/ddsws1-0?
verb=Search&q=(stems:oceans OR titlestems:oceans)&s=10&n=10...

The user’s query is augmented to search using stemming. This results in greater search recall.

A second clause adds weight for records with matching terms in their title, which boosts these records to the top of the search results.

Oceans Alive
http://www.mos.org/oceans/index.html

Oceans Alive covers basic information about Earth’s oceans, including sections such as: The Water Planet, Oceans in Motion, Life in the Sea, Scientists at Sea and Resources. Topics include physical features of oceans, how the oceans formed, the water cycle, currents and waves, ebbs and tides, ocean plants and animals, and ocean research. The resources section contains links for more information ...
Full description
ª Submit a comment, teaching tip or review for this resource.
Tools for Developers

• DLESE has created tools to aid developers in their use of the Web services and promote adoption of the service

• Tools are available for two developer audiences:
  • Web site designers
  • Application developers and software engineers
Tools for Web Site Designers

- Search Web page template
  - Easily modified and customized using a text editor or tools like Dreamweaver
  - Implements the common features found in a library search Web page
  - Includes customizable “smart link” menus that allow developers to map their own vocabularies to arbitrary search queries
- Implemented with Java Server Pages (JSP)
  - The JSP search pages can be edited, developed and deployed on Windows, Mac OS, Linux or other platforms
- Available for browsing and download at: http://www.dlese.org/dds/services/examples/dds/ws/
Tools for Application Developers

- DDSWebService Explorer
  - A Web page that facilitates issuing requests to the service and viewing the Web service requests and responses in a Web browser
  - Located at: http://www.dlese.org/dds/services/

- Code examples
  - Illustrates how the Web service is used to implement a variety of functions
  - Implementation and examples provided as JSP
  - Available for browsing and download at: http://www.dlese.org/dds/services/examples/ddsws/

- Web service specification
  - Provides the formal specification for the Web service
  - Includes example requests and responses
  - Details the search fields and Information Retrieval features available from the service
  - Located at: http://www.dlese.org/dds/services/ddsws1-0/service_specification.html
References

- OAI – Open Archives Initiative
  http://www.openarchives.org/
- ODL – Open Digital Libraries
  http://oai.dlib.vt.edu/odl/
- DDSWebService – The DLESE Discovery System Web Service
  http://www.dlese.org/dds/services/
- SRW/SRU – Z39.50 International: Next Generation
  http://www.loc.gov/z3950/agency/zing/
- REST – REpresentational State Transfer
  http://rest.blueoxen.net/cgi-bin/wiki.pl
- SOAP – Simple Object Access Protocol
  http://www.w3.org/TR/soap/
- WSDL – Web Services Description Language
  http://www.w3.org/TR/wsdl
- Lucene – Information Retrieval engine
  http://jakarta.apache.org/lucene/
- JSP – Java Server Pages
  http://java.sun.com/products/jsp/
Contact Information

- For information about DDSWebService, see: http://www.dlese.org/dds/services/

or send e-mail to: support@dlese.org