Data Integration Using SOAP in the VSO

Karen Q. Tian, Richard S. Bogart
Stanford University

Frank Hill, Stephen Wampler
National Solar Observatory

Piet Martens, Alisdair Davey
Montana State University

Joseph B. Gurman, George Dimitoglou
Solar Data Analysis Center
VSO Overview

Perspectives

☞ From a user
  ➢ Search all participating data sources with a common interface

☞ From a data provider
  ➢ Register data services with VSO
  ➢ Utilize existing internal query and export facilities

Challenges

☞ Distributed locations

☞ Heterogeneous data services
  ➢ DBs or file systems
  ➢ Search capabilities

☞ Integration
  ➢ Transparency
  ➢ Metadata as “glue”

☞ Extensibility
  ➢ Add future data providers
  ➢ Session logging
XML (eXtensible Markup Language)

A mechanism to identify structure in a document

- “keyword=value” + structure
- user defined arbitrary tags, no semantics
- text-based and platform-independent

Applications

- format for data exchange — widely accepted
- format for data storage — ???, native XML databases exist
- mid-ground — relational DB that provides XML view
  - XML query
  - Mapping between XML view and relational DB

And tons more X-concepts: XML schema, XLink, XPointer, XPath, XSL, XSLT, XQuery, DOM& SAX, XHTML, ...
Resource Registry

 Dataset description — “What”
➢ Observable
➢ Time coverage
➢ Operation status
➢ Contact
➢ etc

 Interface description — “How”
➢ Searchable
➢ Retrievable
➢ Format
➢ etc

A sample entry:

<DataProvider>
   <Name>S01</Name>
   <Organization>Stanford University</Organization>
   <Facility>Instrument</Facility>
   <Contact>R. Bogart</Contact>
   <Dataset Name="MDI">  
     <Dopplergram>
       <Polarization>Linear</Polarization>
     </Dopplergram>
   ...
   <TimeCoverage>
     <Start>1996-01-01</Start>
     <End>2003-02-01</End>
   </TimeCoverage>
   <URL>http://soi.stanford.edu</URL>
   <OperationStatus>On-line</OperationStatus>
   <Distribution>HTTP</Distribution>
   </Dataset>
</DataProvider>
Web Services

Characteristics

➢ Service available over the network
➢ Standardized XML messaging
➢ Independent of platform and programming language

Application-centric replacing human-centric (POST/GET)

Automation of the Web: service description, service registry

Protocol stack

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery</td>
<td>UDDI</td>
</tr>
<tr>
<td>Description</td>
<td>WSDL</td>
</tr>
<tr>
<td>XML messaging</td>
<td>XML-RPC, SOAP</td>
</tr>
<tr>
<td>Transport</td>
<td>HTTP, SMTP, FTP</td>
</tr>
</tbody>
</table>
Simple Object Access Protocol

Characteristics

- RPC (Remote Procedure Call) mechanism
- HTTP as transport
- Client-server messaging encoded in XML documents.
  - Independent of platform and programming language

Implementation available for Java, Perl, Python, etc.

Three major parts

- Data encapsulation specs: XML envelope
- Data encoding rules: agreed-upon data types
- RPC conventions: one- or two-way messaging
Perl SOAP::Lite Module

Written by Paul Kulchenko

Interfaces

Client

use SOAP::Lite;
$soap = SOAP::Lite
  -> uri('http://vso.stanford.edu/MDI')
  -> proxy('http://vso.stanford.edu/mdi.cgi');

$result = $soap->Query();

Server

use SOAP::Transport::HTTP;
SOAP::Transport::HTTP::CGI
  -> dispatch_to('MDI')
  -> handle;
package MDI;
sub Query { ... }

Error handling mechanism

➢ Timeout
➢ Reason of failure: standard and custom-defined
Functionalities

➢ Data Model Mapping
➢ Query Construction

Implementation points

Diagram showing Provider and VSO with connecting arrows.
Data Provider Issues

- Diversity
  - Bookkeeping mechanisms
    - Databases
      - PostgreSQL
      - Oracle
      - MySQL
    - File systems
  - Existing search/export capability
- Work load concern
  - Limit maximum number of results
- Cost of data delivery
- Duplicated datasets
## Time Interval Search: Registry

### Sample entries

<table>
<thead>
<tr>
<th>Data provider</th>
<th>Time coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stanford University</td>
<td></td>
</tr>
<tr>
<td>National Solar Observatory</td>
<td></td>
</tr>
<tr>
<td>Solar Data Analysis Center</td>
<td></td>
</tr>
<tr>
<td>Montana State University</td>
<td></td>
</tr>
</tbody>
</table>

- For a given a time interval, resource registry tells VSO which provider might have the matching datasets.

- It is these providers that VSO queries.
Search for all datasets for October 30, 2001

WHERE UNIX_TIMESTAMP(obstart) >=
    UNIX_TIMESTAMP("2001.10.30 00:00:00")
    AND UNIX_TIMESTAMP(obsend) <=
    UNIX_TIMESTAMP("2001.10.30 23:59:00")

WHERE date_obs >= 10-Oct-2001 00:00:00
    and date_end <= 10-Oct-2001 23:59:00

WHERE '2001.10.30' <= Date_Obs
    AND Date_Obs <= '2001.10.30'
    AND '00:00:00' <= Time_Obs_Start
    AND Time_Obs_End <= '23:59:00'

WHERE series_num >= 77376
    AND series_num <= 77399
Reference

- Official VSO webpage http://virtualsolar.org
- Time Search http://vso.stanford.edu/ti.html
- Perl::Lite http://soaplite.com
- E. Cerami, Web service essentials, O’Reilly, 2001