BNL-95294-2011-CP

RESTful Web Services
at Brookhaven Lab
NLIT 2011

Rich Casella
rac@bnl.gov
Brookhaven National Laboratory
IT Division, Application Services

BROOKHAVEN NATIONAL LABORATORY
a passion for discovery

U.S. DEPARTMENT OF ENERGY | Office of Science

Disclaimer

Caveat

- This is not a SOAP vs REST talk
- Choose the tool that is right for you based on your own design considerations

About this talk

- Brief description of RESTful
- Design Considerations
- Implementation
- Outstanding Issues
- Future
- Questions

What is REST

- REpresentational State Transfer
- Requests/responses relate to representations of states of a resource
- Client/Server WWW Transfer Protocol
- Architecture Introduced by Roy Fielding in 2000
- Resource Oriented Architecture

RESTful Constraints

- Client-Server
- Stateless (no cookies)
- Cacheable (where possible)
- Uniform Interface
- Layered System
- Code On-Demand (optional)

Using RESTful Services

- Uses simple HTTP protocol in practice, rather than SOAP or XMLRPC
- Using HTTP methods:
  - GET: Return data, nothing is changed on server
  - POST: Create, update, or delete data on server
  - PUT: Replace referenced resource(s)
  - DELETE: Delete referenced resource(s)

SOAP/REST request comparison

- SOAP

```
<?xml version="1.0"?>
<soap:Envelope
  xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
  soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
  <soap:body pb="http://www.acme.com/phonebook">
    <pb:GetUserDetails>
      <pb:UserID>12345</pb:UserID>
    </pb:GetUserDetails>
  </soap:Body>
</soap:Envelope>
```

- REST

```
http://www.acme.com/phonebook/UserDetails/12345
```

RESTful Pros

- Simple interface (URI based)
- Uses HTTP service methods (GET, POST, ...)
- Cacheing can be leveraged for performance
- Small learning curve
- Simple to test (browser compatible)
- Less reliance on tools
- No standard

RESTful Cons

- Not yet well integrated into IDE's (but getting better)
- Security relies on HTTP authentication
- * Less reliance on tools
- * No standard

RESTful Players

- Amazon Web Services
- Google Maps
- Yahoo

Design Considerations

- Data
  - *NIX configuration data
    - Configuration files
    - NIC data
    - OS signature
    - Process data
  - Cyber Security compliance data
    - Disk encryption type
    - Accounts/User logins
    - Package/software versions

Design Considerations

- Diverse customer base drives different output requirements
  - Human readable (reports)
    - Cyber Security
    - Management
  - Non-human readable
    - System Administrators
    - Web Designers
- RESTful architecture allows for different output formats

URI Construction

- Resource Oriented
  - URI to resource returns API/Usage
  - URI to resource details returns data
- Examples
  - API/Usage
    - http://webservices.bnl.gov/ordo
    - http://webservices.bnl.gov/ordo/packages
    - http://webservices.bnl.gov/ordo/status
  - Details
    - http://webservices.bnl.gov/packages/130.199.130.70
    - http://webservices.bnl.gov/packages/package/openssl
    - http://webservices.bnl.gov/ordo/setOSStatus/supported

Output Format

- Gather requirements from all users
- Support both human and non-human readable output
- Report Formats
  - (X)HTML (human readable)
    - Straightforward design using existing tools and templates for standard formatting.
  - XML (non-human readable)

XML

- Design XML schema for output data to ensure well-formed response to data requests
  - Report Data
    - Include XML schema location (show XML)
    - Use tools for standard formatting (Hash-based)
  - Error Data
    - Use standard HTTP errors where applicable
      - (404) Not Found
      - (503) Service Unavailable
    - Create library of application specific errors
    - Return both error code and error text from library call (XML)

Implementation

Perl CGI on Apache web server

Dual homed to handle different output formats

```
[apps@toolbox5 cgi-bin]$ ls -l
total 64
-rw-r----- 1 apps app-www    0 Mar 18 09:20 index.html
-rwxr-x--- 1 apps app-www 6745 May 13 12:50 ordo
-rwxr-x--- 1 apps app-www 9182 May 16 15:11 setOSStatus
drwxr-s--- 2 apps app-www 4096 May 17 11:08 WS
[apps@toolbox5 cgi-bin]$ ls -l WS
total 40
-rw-r----- 1 apps app-www    0 Apr 28 10:14 index.html
-rwxr-x--- 1 apps app-www 6116 May 16 11:09 ordo
-rwxr-x--- 1 apps app-www 8370 May 17 11:05 setOSStatus

URI: http://webservices.bnl.gov/cgi-bin/ordo
URI: http://webservices.bnl.gov/cgi-bin/WS/ordo
```

Outstanding Issues

- Authentication
- Not completely there yet
  - non-cacheing
- XML to (X)HTML

Questions?

Resources

- Rich Casella - rac@bnl.gov
- Learn REST: A Tutorial:
  - http://rest.elkstein.org
- Fielding Dissertation:
  - http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm