

Web Services

What is a Web Service?

- Piece of software available over Internet
- Uses standardized (i.e., XML) messaging system
- More general definition: collection of protocols and standards used for exchanging data between applications or systems

Web Services Architecture

- Technologies capable of
 - Exchanging messages
 - Describing Web services
 - Publishing and discovering Web service descriptions

An Example: EMBL

- EMBL nucleotide sequence database
- <http://www.ebi.ac.uk/cgi-bin/dbfetch>
 - X12399

Another Example: NOAA

- Oceanographic data
- Water level, currents data, etc.
- <http://opendap.co-ops.nos.noaa.gov/axis/>

A service example

- Previous two examples provide **data**
- Web services can also perform analysis
- Example: sequence similarity using ClustalW:
 - <http://www.ebi.ac.uk/clustalw/>

Observations

- Web services useful for both retrieval and analysis
 - Leverage existing programs/data
 - “Service-oriented science”
- What if I want to analyze 100s of objects?

Service-Oriented Science

- Idea: need standards and interfaces to encapsulate information tools as services
- No knowledge of inner workings required
- Service oriented architecture: systems as networks of loosely-coupled, communicating services
- Service oriented science: scientific research enabled by networks of interoperating services

Advantages for Scientists

- Data on the web, not in the lab
- Automate time-consuming activities
- Infrastructure issues
 - Share compute resources
- Automation
 - Enables programs to process large volumes of data quickly

Creating and Sharing Services

- Describe operations service supports
- Define protocol to invoke services over Internet
- Operate server to process incoming requests

Web Service Challenges

- Interoperability
- Scale
- Management
- Quality Control
- Incentives

Interoperability

- Need standards
- Why is this useful?
 - Automate processing of requests
 - Common interface to a variety of services
 - Need common protocols, data formats, and semantics

Scale

- Services must deal with:
 - Large volumes of data
 - High computational demands
 - Many users
- May require access to high-performance resources

Management and Quality Control

- Management – who uses services and for what purpose?
 - Prevent overloading
- Quality control – metadata and lineage info
 - Allow users to determine quality of data
- Incentives
 - Sharing data
 - Overhead of developing web service

Infrastructure

- Idea: domain independent functions and resources can be handed off to specialized providers
- Scientists can focus on domain-specific data and software
- Example:
 - Open Science Grid (OSG)
 - “SourceForge for scientists”

Approaches to Scaling

- 1. Cookie-cutter approach
 - Researchers create dedicated domain-specific infrastructures
 - Standardized domain specific software
 - Examples:
 - Bioinformatics Research Network
 - Network for Earthquake Engineering Simulation
 - PlanetLab
 - High degree of central control and uniformity
 - But high cost of expansion, requires new hardware

Approaches to Scaling

- 2. Bottom-up approach
 - Researchers develop service ecologies – agreements on interfaces
 - Participants provide content and function as they see fit
 - Each site responsible for its own equipment
 - Examples:
 - Earth System Grid
 - myGrid – biological workflows
 - Low cost of entry, but may not scale

Approaches to Scaling

- 3. General-purpose infrastructure
 - Discipline independent resources or functions
 - Access to monitoring functions, simulation codes, etc.
 - Examples:
 - TeraGrid
 - Campus Grids

Summary

- Benefits of service-oriented science:
 - Access to data
 - Access to programs to analyze data
 - Automatic processing
- Challenges:
 - Interoperability, scalability, quality
 - Individual communities need to outsource infrastructure functions and resources

Some Web Service Standards

- REST (Representational State Transfer)
 - Architecture style of networked systems
- SOAP (Simple Object Access Protocol)
 - Standard for sending messages between applications
- WSDL (Web Service Description Language)
 - Standard for describing web services and their capabilities

REST

- **Representational State Transfer**
- Architectural style (technically not a standard)
- Idea: a network of web pages where the client progresses through an application by selecting links
- When client traverses link, accesses new resource (i.e., transfers state)
- Uses existing standards, e.g., HTTP

REST Characteristics

- Client-Server: Clients pull representations
- Stateless: each request from client to server must contain all needed information.
- Uniform interface: all resources are accessed with a generic interface (HTTP-based)
- Interconnected resource representations
- Layered components - intermediaries, such as proxy servers, cache servers, to improve performance, security

REST examples

- The Web is RESTful (a bunch of HTTP links)
- EBI (European Bioinformatics Institute)
 - <http://www.ebi.ac.uk/xembl>
- Dbfetch – retrieve entries from EMBL nucleotide sequence database
 - <http://www.ebi.ac.uk/cgi-bin/dbfetch?db=embl&id=x12399&format=default>

SOAP

- **S**imple **O**bject **A**ccess **P**rotocol
- Format for sending messages over Internet between programs
- XML-based
- Platform and language independent
- Simple and extensible
- Stateless, one-way
 - But applications can create more complex interaction patterns

SOAP Building Blocks

- Envelope (required) – identifies XML document as SOAP message
- Header (optional) – contains header information
- Body (required) – call and response information
- Fault (optional) – errors that occurred while processing message

Simple Example

“Get the price of apples”

```
<?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>
```

```
  <m:GetPrice xmlns:m="http://www.w3schools.com/prices">
```

```
    <m:Item>Apples</m:Item>
```

```
  </m:GetPrice>
```

```
</soap:Body>
```

```
</soap:Envelope>
```

Note:**GetPrice** and **Item** are application-specific (not part of SOAP)

Summary: SOAP vs. REST

- REST – clients submit requests to Web services as HTTP requests
- SOAP – clients submit request in form of XML document

SOAP vs. REST

- SOAP advantages:
 - Widely used and supported
 - Supports variety of protocols (network transfer, authentication, encryption)
- REST advantages:
 - Simple, relies only on HTTP protocol
 - Performance and reliability
- Many sites support both SOAP and RESTful interfaces
 - Amazon.com, XEMBL from EBI

WSDL (Web Service Description Language)

- Standard method of describing Web Services and their capabilities
- Idea: Automate details involved in applications communication
- Operations and messages are described abstractly
- Bound to a concrete network protocol and message format to define an endpoint
- Provide documentation for distributed systems

WSDL Details

- A WSDL document defines **services**
- Services are collection of network endpoints (**ports**)
- Abstract definition of endpoints and messages is separated from their concrete network deployment or data format bindings
- Allows the reuse of abstract definitions:
 - **messages** -abstract descriptions of data being exchanged
 - **port types** -abstract collections of **operations**
 - concrete protocol and data format specifications for a particular port type constitutes a reusable **binding**

An example

- From BLAST service at DDBJ
<http://xml.nig.ac.jp/wsdl/Blast.wsdl>
- SearchSimple – function takes as input:
 - Program – which BLAST program to use
 - Database – which BLAST database to query
 - Query – sequence to query
- SearchSimple output:
 - Result – string containing matches

An example

```
<message name="searchSimple1In">  
  <part name="program" type="xsd:string" />  
  <part name="database" type="xsd:string" />  
  <part name="query" type="xsd:string" />  
</message>  
<message name="searchSimple1Out">  
  <part name="Result" type="xsd:string" />  
</message>
```

Example (continued)

```
<portType name="Blast">  
  <operation name="searchSimple"  
    parameterOrder="program database query">  
    <documentation>Execute Blast</documentation>  
    <input name="searchSimple1In"  
      message="tns:searchSimple1In" />  
    <output name="searchSimple1Out"  
      message="tns:searchSimple1Out" />  
    </operation>  
    ....[other operations]  
  </portType>
```

Summary

- WSDL document lists functions supported by each web service, inputs and outputs
- To actually call a web service, need interface or tools:
 - SOAP:lite (Perl)
 - Apache Axis (Java)
 - Many others...

A WSDL example: DDBJ

- <http://xml.ddbj.nig.ac.jp/wsdl/index.jsp>
- Look at the SearchSimple WSDL example
- SearchSimple(program,database,query) in BLAST web service
- Many other web services available

Using SOAP:Lite

```
sub Blast_SOAP{  
    local $query_file = @_ [0];  
    $service = SOAP::Lite ->  
service('http://xml.nig.ac.jp/wsdl/Blast.wsdl');  
    $service->proxy('http://localhost/', timeout =>  
60*60);  
    return $service->searchSimple("blastn",  
"ddbjhum", $query_file);  
}
```

\$service connects to Blast web service at DDBJ site

blastn is program, **ddbjhum** is database, **\$query_file**
contains sequence

Using Apache Axis

```
public class BlastSoap extends SoapMaster{

    public BlastSoap(){

super("http://xml.nig.ac.jp/wsd/Blast.wsdl", "http://www.themi
ndelectric.com/wsd/Blast/", "Blast", "searchSimple");
    }

    public String runBlastSoap(Vector pQuery) throws Exception{
        Vector query = new Vector();
        query.add("blastn");
        query.add("ddbjhum");
        for(int i=0;i< pQuery.size();i++){
            query.add(pQuery.elementAt(i));
        }
        return this.runSoap(query);
    }
}
```

How are these used?

Many SOAP and REST based bioinformatics web services, including:

- EBI
 - www.ebi.ac.uk/Tools/webservices/
- NCBI
 - http://eutils.ncbi.nlm.nih.gov/entrez/query/static/eutils_help.html .
- DNA Databank of Japan (DDBJ)
 - <http://xml.ddbj.nig.ac.jp/soapp.html>

Describing Web Services - UDDI

- Universal Description, Discovery, and Integration
- Directory for locating Web services
- Can be interrogated by SOAP messages and provides access to WSDL documents describing web services in its directory

Some example web services

- SDSS-based web services
- NOAA
- Bioinformatics
 - PubMed
 - GenBank
 - BLAST
 - SwissProt
 - ClustalW

SDSS

- <http://cas.sdss.org/dr5/en/skyserver/ws/>
- Image Cutout – get jpg images of parts of the sky:
<http://casjobs.sdss.org/ImgCutoutDR5/Imgcutout.aspx>
- CAS access – submission of SQL queries
– http://voservices.net/CasService/ws_v1_0/CasService.aspx
- Many others...

NOAA

- <http://opendap.co-ops.nos.noaa.gov/axis/>

PubMed/Medline

- <http://www.ncbi.nlm.nih.gov/entrez/>
- Comprehensive database of scientific literature in biomedical area
- Useful for finding literature on a given topic



All Databases

PubMed

Nucleotide

Protein

Genome

Structure

OMIM

PMC

Journals

Books

Search PubMed for dUTPase [Save Search](#) Display Summary Show 20 Sort by Send to All: 322

Items 1 - 20 of 322

Page 1 of 17

- 1:** [Jiang YL, Chung S, Krosky DJ, Stivers JT.](#) [Related Articles, L](#)
- Synthesis and high-throughput evaluation of triskelion uracil libraries for inhibition of human dUTPase and UNG2.**
Bioorg Med Chem. 2006 May 4; [Epub ahead of print]
PMID: 16678429 [PubMed - as supplied by publisher]
- 2:** [McCarthy OK, Schipani A, Buendia AM, Ruiz-Perez LM, Kaiser M, Brun R, Pacanowska DG, Gilbert IH.](#) [Related Articles, L](#)
- Design, synthesis and evaluation of novel uracil amino acid conjugates for the inhibition of Trypanosoma cruzi dUTPase.**
Bioorg Med Chem Lett. 2006 May 2; [Epub ahead of print]
PMID: 16677813 [PubMed - as supplied by publisher]
- 3:** [Wu M, Shen J, Zhan J, Yu Y.](#) [Related Articles, L](#)
- dUTP Pyrophosphatase, its appearance in extracellular compartment may serve as a potential biomarker for N-methyl-N'-nitro nitrosoguanidine exposure in mammalian cells.**
Proteomics. 2006 May;6(10):3001-7.
PMID: 16619294 [PubMed - in process]
- 4:** [Guillet M, Van Der Kemp PA, Boiteux S.](#) [Related Articles, L](#)
- dUTPase activity is critical to maintain genetic stability in Saccharomyces cerevisiae.**
Nucleic Acids Res. 2006 Apr 14;34(7):2056-66. Print 2006.

[About Entrez](#)
[NCBI Toolbar](#)[Text Version](#)[Entrez PubMed](#)
[Overview](#)
[Help | FAQ](#)
[Tutorials](#)
[New/Noteworthy](#)
[E-Utilities](#)[PubMed Services](#)
[Journals Database](#)
[MeSH Database](#)
[Single Citation](#)
[Matcher](#)
[Batch Citation Matcher](#)
[Clinical Queries](#)
[Special Queries](#)
[LinkOut](#)
[My NCBI](#)[Related Resources](#)
[Order Documents](#)
[NIH Mobile](#)

PubMed Example

- <http://www.ncbi.nlm.nih.gov/entrez/>
- Find out about “dUTPase”

GenBank

- National Institutes of Health (NIH) genetic sequence database
- Collection of all known sequences
- Each GenBank entry consists of:
 - Locus name and accession number
 - Reference section – link to relevant articles
 - Features
 - Sequence

[Nucleotide](#) [Protein](#) [Genome](#) [Structure](#) [PMC](#) [Taxonomy](#) [DMIM](#) [Books](#)
[Sign In](#) [Register](#)

Reverse complemented strand

1: [X01714](#). Reports *E. coli* dut gene ...[gi:41296]

[Comment](#) [Features](#) [Sequence](#) [Links](#)

ACCESSION X01714
VERSION X01714.1 GI:41296
DEFINITION *E. coli* dut gene for dUTPase (EC 3.6.1.23) (deoxyuridine 5'-triphosphate nucleotidohydrolase).
KEYWORDS dUTPase; unidentified reading frame.
ORGNAMISM *Escherichia coli*
Escherichia coli
 Bacteria; Proteobacteria; Gammaproteobacteria; Enterobacteriales; Enterobacteriaceae; *Escherichia*.
REFERENCE 1 (bases 1 to 1609)
AUTHORS Lundberg,L.G., Thoreson,H.O., Karlstrom,O.H. and Nyman,P.O.
TITLE Nucleotide sequence of the structural gene for dUTPase of *Escherichia coli* K-12
JOURNAL EMBO J. 2 (6), 967-971 (1983)
PUBMED [6139280](#)
COMMENT Data kindly reviewed (25-NOV-1985) by L. Lundberg.
FEATURES Location/Qualifiers
 source 1..1609
 /organism="*Escherichia coli*"

Other scientific domains:

- Astronomy:
 - <http://www.atnf.csiro.au/vo/rvs/thumbnail.html>
 - Converts astronomy image to thumbnail
- Chemistry:
 - <http://www.liv.ac.uk/Chemistry/Links/webservices.html>
- GIS
 - <http://terraservice.net/webservices.aspx>

Summary

- Web services provide standardized interface to data and functionality on web
 - Access to data
 - Perform computations
- Accessible by machines/programs
- Stateless – websites don't keep track of requests
- Preview of things to come: scientific workflows

Other Web Services

- Web services exist in many domains
 - Stock quotes, conversions, weather...
- Web service search engines (e.g., Woogole):

<http://haydn.cs.washington.edu:8080/won/wonServlet>