Web services en bioinformática

DAS, BioMOBY y Taverna

Teoría

José Manuel Rodríguez Carrasco
José María Fernández González

Biología Estructural y Bioinformática, CNIO
GN2 (CNIO), INB
Bioinformatic chaos
Bioinformatics integration: State of the Art

A Web Page is the *de facto* standard

**Discovery:**
- Word of mouth
- Web directories
- Web searches
- Paper publications

**Description:**
- Word of mouth
- Web documentation / examples / tutorials / courses
- Paper publications

**Data Transfer and Message Format:**
- Cut & Paste® + Data reformatting

**Automation:**
- CGI & *Bespoke code* (ad hoc)
- APIs (usually only big Bioinformatics Projects/Institutes)

---

Do you have data?
Do you have tools?
Publish a web page

Service Consumers

**Discovery**

Services
(Applications, DBs)

Remote
Programmatic
Access

**Description**

Service Providers

Current (& past) Practices

José Manuel Rodríguez Carrasco (jmrodriguez@cnio.es)
José María Fernández González (jmfernandez@cnio.es)

Escuela Complutense de Verano 2007
Bioinformática y Biología Computacional
We see lots of web pages which provide HTML forms to look up data or perform calculations

- How do we find out where services are being provided?
- What type/kind of services is a provider offering?
- Once I have discovered a service, how do I use it?
- How do I use the service from within a program instead of through a form on a web page?
- Which are the input and output data types/formats?
- How do I take the output of one service and send it to another service?

Why not CGI services?

Why not CGI services?
Web Services can solve these problems

Prerequisite: Web services' authors have to register their services and/or their description with some kind of central repository.

Kinds of Service Repositories

- **“White Pages”**: who is providing services.
- **“Yellow Pages”**: services organized using some classification.
- **“Green Pages”**: how to call each service.
- **“BIO pages”**: given an specific type of biological data (e.g., an UniProt accession, a keyword, or a 3D structure), find services which can take it as input or generate it as output.
The Bioinformatics Curse
(and a way to delay it)

1. A Web page is an interface: Facilitate access of Web pages to scripts.
2. An interface is a contract: Give advance warning of interface changes to minimize inconvenience; use new URLs and provide temporary legacy interface.
3. Choice is good: SOAP/XML is “sweet!” — HTML is discouraged.
4. Allow batch downloads: Break data sets into bite-sized pieces to facilitate downloading.
5. Use existing file formats: Do not reinvent the wheel when multiple workable formats already exist.
6. Design sensible formats: Use common sense and simplicity.
7. Allow ad hoc queries: Make copies of databases available for direct access using native query language.

A web service is an interface which describes a set of operations which are network available using standardized XML messages.*

*Web Services Conceptual Architecture, Heather Kreger, IBM Software Group, 2001
Web Services Status

- Simples, Open, Broad Industry Support
- Futuristic model, although lots of investment has already been made i.e. Microsoft .NET
- Technical infrastructure available (assets): Existing foundation using XML-based technologies
  - **XML** (*eXtensible Markup Language*): Universal Data Transport format
  - **SOAP** (*Simple Object Access Protocol*), **XML-RPC**: Wire protocol, service invocation and interactions.
  - **WSDL** (*Web Services Description Language*), **DAML-S**: Formal service descriptions.
  - **UDDI** (*Universal Description, Discovery and Integration*): Service registry (publication) and discovery (find).
  - **XSD, XSLT** (*Extensible Stylesheet Language Transformations*): Data Types, Description and Conversion.
  - **HTML, SMTP, FTP**: Ubiquitous Communications Protocol
- **Examples**:
  - Google Web APIs ([http://www.google.com/apis/](http://www.google.com/apis/) and Gmail)
  - XEMBL ([http://www.ebi.ac.uk/xembl/](http://www.ebi.ac.uk/xembl/))
Web Service query processing

Bioinformatic Application

Query data

Query Wrapping

Results for the application

Data Translation to the application

Answer object

Answer Wrapping

Application Server

Query

Loading input into memory

Network Connection

Query object

Results for the application

Data Translation to the application

Answer object

Answer Wrapping

Application Server

Query

Loading input into memory

Network Connection

Query object
Web Service technologies related to Bioinformatics

- **I3C** ([http://www.i3c.org](http://www.i3c.org))
  - **LSID**: *Life Sciences Identifier naming standard and data access protocol* (IBM/OMG)

- **SOAPLab** ([http://www.ebi.ac.uk/soaplab/](http://www.ebi.ac.uk/soaplab/))
- **SeqHound** ([http://www.blueprint.org/seqhound/](http://www.blueprint.org/seqhound/))
- **myGrid** ([http://www.mygrid.org/](http://www.mygrid.org/))
- **DAS** ([http://www.biodas.org/](http://www.biodas.org/))
Web services en bioinformática

BioDAS

José Manuel Rodríguez Carrasco (jmrodriguez@cnio.es)
José María Fernández González (jmfernandez@cnio.es)
DAS

(Distributed Annotation System)

El sistema DAS de anotación distribuida es un sistema cliente-servidor en el que cualquier cliente puede integrar información de diferentes servidores. Fue diseñado con las siguientes premisas:

- Independencia de esquema o tecnología de BD
- Independencia de tecnologías de cliente
- Servidores de referencia y de anotación no relacionados
- Gestión de inestabilidad en ensamblajes genómicos
- Sencillo de implementar

Como consecuencia de ello, tiene las siguientes características:

- Cualquier cliente DAS puede integrar información de varios sitios web remotos y mostrarla en una vista única al usuario.
- Los distintos proveedores de información necesitan coordinarse entre sí poco o nada.

La especificación 1.0 fue escrita por Lincoln Stein y Robin Dowell
¿Quién está usando DAS?

- Ensembl (muchos organismos, pero sobre todo H. Sapiens)
- WormBase (C. elegans)
- Washington University @ St. Louis (C. elegans)
- Flybase (Drosophila)
- UCSC Genome Center (H. Sapiens)
- TIGR (C. elegans, H. Sapiens)
- Cambridge University Genetics Department (C. elegans, H. Sapiens, Drosophila)
- DOE Joint Genome Institute (H. Sapiens, M. Musculus)
- NCBI (in progress)
- INB (also in progress)
¿Qué es una anotación?

Cualquier información definible entre dos coordenadas genómicas

AC003027

- Genes
- Exons
- CDS
- Restriction Sites
- Ribosome binding sites
- Repeats
- PolyA signals
- Scientific references (papers)

- Sequencing Process Annotations
  - Posiciones de clones, secuencias de fin de BAC
- Análisis *In Silico*
  - Predicción de genes, Homologías
- Datos Experimentales
  - Knockouts, resultados de Microarray
- Otros
  - Grupos de Discusión, referencias PubMed, GO Terms
¿Cómo se anota?

ID: gene:SC01009
Synonym: 2SCG2.22c
Start, Stop: 1063520, 1064140
Orientation: -
Category: Transcription
Type: Gene
Method: GeneFinder v1.3
Group: 2SCG2
Muchos sistemas de coordenadas

cromosoma
contigs
clones
genoves predichos

otras características
¿Qué genomas tienes?

Protocolo DAS
(1) Data Sources

Cliente

Servidor de Referencia
Tengo elegansWS34 y elegansWS37

Protocolo DAS
(2) Respuesta Data Sources

Cliente

Servidor de Referencia
Protocolo DAS

(3) Recuperar mapa de anotaciones

Ahí tienes el mapa

Quiero el Crom. II de elegansWS37
¿Qué servidores de anotación hay?

http://www.tigr.org/cgi-bin/das
http://www.wustl.edu/genome/das
http://www.wormbase.org/db/das

(4) Servidores de anotaciones
Protocolo DAS
(5) Solicitar anotaciones

Quiero anotaciones para:
ChrII:1,2000000, ZK154:300,51993, AH6:1,4000000....
Protocolo DAS

(6) Recuperar anotaciones

Sin problema....

Servidores de Anotación

Cliente

Anotación

ID
Version
Ref sequence
Start, stop
Type, method, category
URL to link to
Protocolo DAS
(7, opcional) Recuperar hojas de estilo

Quiero las hojas de estilo

Cliente

Servidores de Anotación

<table>
<thead>
<tr>
<th>Tipo de Anotación</th>
<th>Representación</th>
</tr>
</thead>
<tbody>
<tr>
<td>transcript</td>
<td>arrow</td>
</tr>
<tr>
<td>SNP</td>
<td>dot</td>
</tr>
<tr>
<td>similarity</td>
<td>box</td>
</tr>
<tr>
<td>knockout</td>
<td>wavyline</td>
</tr>
</tbody>
</table>
Protocolo DAS

(8) Integrar y representar
Ejemplo Cliente DAS
XML para el intercambio de datos

```xml
<DASGFF>
  <GFF gffversion="0.80" summary="no">
    <SEGMENT id="ZK154" start="1" stop="49000" version="2.0">
      <FEATURE id="exon:ZK154.3" version="4.0">
        <TYPE id="exon" category="transcription">exon</TYPE>
        <METHOD id="gf1.3">GeneFinder 1.3</METHOD>
        <START>2229</START>
        <END>2620</END>
        <ORIENTATION>+</ORIENTATION>
        <GROUP id="ZK154">
          <LINK href="http://www.wormbase.org/db/get?ZK154.3">WormBase</LINK>
        </GROUP>
      </FEATURE>
    </SEGMENT>
  </GFF>
</DASGFF>
```
Web services en bioinformática

moby
BioMOBY - scope and definition

BioMOBY is a project which aims to develop a web services architecture for bioinformatics.

BioMOBY is an international research project which involves biological data servers, biology-related service providers, and programmers whose target is exploring the available methodologies for biological data representation, distribution and finding.

TARGETS

- Study how to address interoperability problems that are actually being faced by bioinformatics users of web-accessible resources today, and what are the factors that promote the adoption of new approaches.

- How to balance between increasing potential for interoperability and the likelihood of widespread adoption? I.e. focus upon minimizing the barriers to entry into the system, or insist upon a set of constraints that will guarantee usefulness of components of the system.

**1. Common Syntax**

**2. Common Semantics**

**3. Dynamic Discovery**
MOBY-S system

Components

MOBY Objects
- data itself (XML)

MOBY Central
- centralized service registry

MOBY Service
- any standard service

MOBY Client
- the bit that you see!
MOBY-S overview (II)

MOBY services and servers

MOBY Central

Sequence

Align

Phylogeny

Primers

…
A BioMOBY session

Registration Phase

Query Phase

Transaction Phase

DATA

Kinds of Services

Available Services

Selected Service

Service Definition

Query

WSDL

Input Data Object

Output Data Object

MOBY Client

MOBY Central

MOBY Service

DATA
BioMOBY Service
Query Processing

MOBY Central
- Data Translation to Application
  - Results for the application

Bioinformatic Application
- Query Data
- MOBY Query Envelope

MOBY Query
- MOBY Object with the answer
- MOBY Object with the Query

Data Answer Envelope
- Answer
- Application Server
- Network Connection (SOAP)
- MOBY Object Validation & Memory Load
- Data Answer
- Envelope

BioMOBY Service
- Query Processing
- Data
- Results for the application
- Application
- Translation to Application
- Server
MOBY messages dispatching

MOBY Service Internal Skeleton

Application

Data to the Application

Data from the Application

Translation from MOBY Object to Application Data

MOBY Object Extraction

MOBY Object

MOBY Object

MOBY Object

Translation to MOBY Object from Application Data

SOAP Message Building

SOAP Message

From/To SOAP Server

External Network Connection

MOBY messages dispatching

Application

Data to the Application

Data from the Application

Translation from MOBY Object to Application Data

MOBY Object Extraction

MOBY Object

MOBY Object

MOBY Object

Translation to MOBY Object from Application Data

SOAP Message Building

SOAP Message

From/To SOAP Server

External Network Connection

Where is new information from?

“SERVICE AMPLIFICATION”

What can I do with Medline, EMBL, GO ...???

X-REFS:
- Medline
- SwissProt
- PubMed
- EMBL
- GO_Annotation

“I can get Expression data From EMBL ID”

“I can find functionally similar genes From GO_Ids”

Citation
- AA
- Locus
- DNA
- GO_Annotation
- Homologues
- Expression

Sequence
- attccg
- ggtcac

MOBY Central
MOBY-S uses ontologies

- By classifying data and services we make searching more powerful
- **Object ontology** types of input/output objects
- **Service type ontology** types of services (currently very poor)
  - Analysis - Service analyzes data
    - NCBI-BLAST - Analysis Blast sequences using NCBI Blast program
  - Parsing - Service parses data
  - Registration - Service registers data
  - Retrieval – Service resolves data
    - Resolution - The base retrieval type
- **Service ontology** available services
- **Namespace ontology** scope of the identifiers
A piece of the MOBY Object Ontology
MOBY and Data Namespaces

- **Examples**: AGI_LocusCode or NCBI_gi
- MOBY adopted Gene Ontology Cross-reference Abbreviations List as its namespace list.
- Namespace list is not closed to the previous ones: you can add the ones you need.
- Each input parameter of the services which are registered at MOBY Central is usually restricted to one or more namespaces.
- E.g. If a service takes as input parameter a GenericSequence object from any namespace, then it is registered with no restriction about the namespaces that input parameter can take.

<table>
<thead>
<tr>
<th>Namespace</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EC</td>
<td>The Enzyme Commission.</td>
</tr>
<tr>
<td>EMBL</td>
<td>EMBL-EBI International Nucleotide Sequence Data Library/DBJ/GenBank.</td>
</tr>
<tr>
<td>ENSEMBL</td>
<td>Database of automatically annotated genomic data.</td>
</tr>
<tr>
<td>ENZYME</td>
<td>The Swiss Institute of Bioinformatics database of Enzymes.</td>
</tr>
<tr>
<td>Flybase</td>
<td>Human Genome Database.</td>
</tr>
<tr>
<td>Global</td>
<td>A somewhat generic prefix for namespaces that are not identifiers for any given database. This could be a namespace for keywords, for example.</td>
</tr>
<tr>
<td>GO</td>
<td>Gene Ontology Database.</td>
</tr>
<tr>
<td>GO_REND</td>
<td>Gene Ontology Database references.</td>
</tr>
<tr>
<td>HGNC</td>
<td>HUGO Gene Nomenclature Committee.</td>
</tr>
<tr>
<td>HUGO</td>
<td>Human Genome Organisation.</td>
</tr>
<tr>
<td>INTERPRO</td>
<td>The Interpro database of protein domains and motifs.</td>
</tr>
<tr>
<td>IPI</td>
<td>International Protein Index.</td>
</tr>
<tr>
<td>ISBNI</td>
<td>International Standard Book Number.</td>
</tr>
<tr>
<td>KEGG</td>
<td>Kyoto Encyclopedia of Genes and Genomes.</td>
</tr>
<tr>
<td>keyword</td>
<td>Any keyword, e.g. &quot;wuschel&quot;.</td>
</tr>
<tr>
<td>LIGAND</td>
<td>KEGG LIGAND database.</td>
</tr>
<tr>
<td>LocusID</td>
<td>NCBI LocusLink ID.</td>
</tr>
<tr>
<td>MEDLINE</td>
<td>The Medline literature database.</td>
</tr>
<tr>
<td>MGD</td>
<td>Mouse Genome Database.</td>
</tr>
<tr>
<td>NCBIAcc</td>
<td>An identifier representing an NCBI (GenBank) Accession number, without a version (e.g. 300322)</td>
</tr>
<tr>
<td>NCBIAccVer</td>
<td>An identifier representing an NCBI (GenBank) Accession number, including the version suffix (e.g. 300322:1)</td>
</tr>
<tr>
<td>NCBI_gi</td>
<td>NCBI databases.</td>
</tr>
<tr>
<td>NCBI_NM</td>
<td>NCBI RefSeq.</td>
</tr>
<tr>
<td>NCBI_NP</td>
<td>NCBI RefSeq.</td>
</tr>
</tbody>
</table>
MOBY Objects

Each MOBY Object is made by a triplet:
- object name, identificator and namespace.
- But it also has:
  - Cross-Reference Information Block (CRIB)
  - Payload (it brings additional data about the object)

```
<Sequence namespace="GenBank/Acc" id=D21125.1>
  <CrossReference>
    <Object namespace="PubMed/ID" id="7948893"/>
    <Object namespace="SwissProt/ID" id="BAA65.1"/>
    <Object namespace="TAIR/Locus" id="AP3"/>
    <Object namespace="GO/Acc" id="GO:0001835"/>
    <Object namespace="EMBL/ID" id="AF056541"/>
  </CrossReference>
  <Length>876</Length>
  <SequenceString>
    gatcaatcca tgttagtttc taactgtggc caacttagtt ....
  </SequenceString>
</Sequence>
```

"Light" object inheritance
- based on simpler objects
- It depends on the instance
MOBY Queries

Both Query and Answer are wrapped in a “MOBY Envelope”

<MOBY Authority="ncbi.nlm.nih.gov" log="Query/ID" id="1334543">
  <mobyData queryId='1'>
    <Simple>
      <Sequence namespace="GenBank/Acc" id=D21125.1>}
        <CrossReference>
          <Object namespace="PubMed/ID" id="7948893"/>
          <Object namespace="SwissProt/ID" id="BAA65.1"/>
          <Object namespace="TAIR/Locus" id="AP3"/>
          <Object namespace="GO/Acc" id="GO:0001835"/>
          <Object namespace="EMBL/ID" id="AF056541"/>
        </CrossReference>
        <Length>876</Length>
        <SequenceString>
          gatcaatcca tgttagtttc
          taactgtggc caacttagtt
          ...
        </SequenceString>
      </Sequence>
    </Simple>
  </mobyData>
</MOBY>
MOBY-S Programming API's

- MOBY-S backbone is a public protocol, so anyone can create their own libraries and programs.
- There are both client and service libraries for Perl, Java and Python programming languages.
- Unlike DAS, currently there is only one implementation of MOBY Central. The Central core is written in Perl. The additional Central services (RDF ontologies, service monitoring) are written in Java.
MOBY Service Clients

These clients will allow you to discover and access data and analysis services provided through the BioMOBY framework. Some also allow the construction and execution of re-usable workflows and analytical pipelines.

<table>
<thead>
<tr>
<th>Client Title</th>
<th>Source</th>
<th>Platform &amp; Language</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gbrowse moby</td>
<td>Mark Wilkinson</td>
<td>Web-based, cross-platform</td>
<td>The first Moby client ever written; very simplistic and limited in power.</td>
</tr>
<tr>
<td>Taverna</td>
<td>Tom Oinn, Martin Senger; myGrid</td>
<td>Java, cross-platform</td>
<td>Taverna allows linking of inputs and outputs from both Moby and non-Moby services into extensive and complex workflows.</td>
</tr>
<tr>
<td>MOWserv</td>
<td>Instituto Nacional de Bioinformatica</td>
<td>Web based, cross-platform</td>
<td>MOWserv allows construction and execution of workflows through a web interface</td>
</tr>
<tr>
<td>Remora</td>
<td>Genopole Toulouse</td>
<td>Web based, cross-platform</td>
<td>Remora allows design of Moby workflows through a web interface</td>
</tr>
<tr>
<td>Ahab</td>
<td>Benjamin Good, Clarence Kwan, Wilkinson Laboratory, UBC</td>
<td>Web based, cross-platform</td>
<td>Ahab allows parallel execution of multiple services simultaneously.</td>
</tr>
<tr>
<td>Seahawk</td>
<td>Paul Gordon, Sun Centre of Excellence, University of Calgary, Alberta, Canada</td>
<td>Java applet, cross-platform</td>
<td>Allows a user to load text or HTML data sources, then discover and execute MOBY Services through hyperlinks and text highlighting.</td>
</tr>
</tbody>
</table>
Clients with embedded MOBY functionality

These are applications that provide a variety of bioinformatics functionality and take advantage of BioMOBY to extend that functionality.

<table>
<thead>
<tr>
<th>Clients with embedded MOBY functionality</th>
<th>Web based, cross-platform</th>
<th>Explore protein interaction networks by surfing MOBY!</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BioTrawler</strong></td>
<td>Frank Gibbons, Harvard</td>
<td></td>
</tr>
<tr>
<td><strong>BlueJay</strong></td>
<td>Paul Gordon, University of Calgary (Genome Canada/Genome Alberta)</td>
<td>Web based, cross-platform</td>
</tr>
<tr>
<td><strong>BioFloWeb</strong></td>
<td>Sophie Durand, INRA, France &amp; PlaNet Consortium</td>
<td>Web based, cross-platform</td>
</tr>
<tr>
<td><strong>AtiDB Client</strong></td>
<td>Sean Walsh, John Innes Centre, UK</td>
<td>Web based, cross-platform</td>
</tr>
</tbody>
</table>

These clients allow exploration of the Moby Central registry, e.g. by graph traversal.

**Moby Graphs** - Martin Senger @ EBI/myGrid
**Moby Service Encyclopedia** - Ken Stuebe
**Registry Browser** - Mark Wilkinson

MOBY Constructor Clients

These clients enable exploration, manipulation, and/or editing of the various Moby Ontologies, and entering new service instances into the MOBY Central registry.

**Object Ontology**

**Object Creation Applet** - Eddie Kawas, UBC
**Object Ontology Browser** for Unix and Windows - Yan Wong (un: Invite; pw: Invite)
**Object Ontology Browser** for OS-X - Yan Wong (un: Invite; pw: Invite)

**Service Ontology**

**Service Type Creation Applet** - Eddie Kawas, UBC

**Namespace Ontology**

**Namespace Creation Tool** - Ambrose Ng, UBC

**MOBY Central - New Service Instance**

**Service Type Creation Applet** - Eddie Kawas, UBC
“Holy Grail” service integration using web services

Retrieve and align 2000nt 5' from every serine/threonine kinase in Fabaceae expressed exclusively in the root cortex whose expression increases 5X or more upon infection by Rhizobium but is not affected by osmotic or heavy-metal stresses and is <40% homologous in the active site to kinases known to be involved in cell-cycle regulation in any other species.
Workflow which solves the query!

GO
BG
SMD
Prosite
prints

Get Serine/Threonine Kinase

Keyword search
Serine/Threonine Kinase

GO Database

MOBY Central

Workflow which solves the query!
Another perspective: S-MOBY

Semantic MOBY

- Semantic MOBY ([www.semanticmoby.org](http://www.semanticmoby.org)) is a Semantic Web-based architecture and open source software project for integrating web-hosted resources in the field of bioinformatics.
- Although this project was frozen more than one year, now it has been invigorated by VPIN (Virtual Plant Information Network) partners.
- Both the MOBY-S and S-MOBY projects are hoping to merge into a common framework within the next year or so.
The Virtual Plant Information Network is a consortium of information resources using emerging semantic web services technologies, composed by:

- **DragonDB** - The *Antirrhinum majus* (Snapdragon) Database
- **Gramene** - A Comparative Mapping Resource for Grains
- **IRIS** - The International Rice Information System
- **IWIS** - The International Wheat Information System
- **LIS** - The Legume Information System
- **TAIR** - The Arabidopsis Information Resource
- **TIGR** Gene Indices - Gene Indices for Medicago
Semantic MOBY Phylosophy (I)

- S-MOBY architecture mimics much of what works for the web – in particular, a document-based architecture with explicit delineations of data and its contextual relationships. Technologies used for that task are RDF and OWL DL (both W3C standards).
- Instead of specifying a syntax and messaging layer to connect clients and providers via a registry lookup (like MOBY-S), it provide clients and providers a way to describe their data and identify data relevant to them.
Semantic MOBY Phylosophy (II)

- **Deploy a common syntax** — enable clients and providers to engage each other under shared syntactic rules of engagement. The decision was made to use RDF graphs serialized as RDF/XML.

- **Develop a common semantic** — enable machine-discernible meaning so clients can request the same conceptual object or service from different providers. It has been done developing a set of RDF classes and properties to describe a canonical graph structure that providers and clients can agree upon, and add OWL DL constraints on the classes and properties to enforce the canonical structure.

- **Implement a discovery server** — enable clients to find providers based on the semantics of their data and services. The decision was made to store RDF graphs that describe mapping operations performed by providers, and provide an HTTP interface for searching the repository.