ISiLS Lecture 12

short introduction to data integration

F.J. Verbeek

Contents

- Genome browsers
- Solutions for integration
- CORBA
- SOAP
- DAS
- Ontology mapping

- 2nd lecture BioASP roadshow

Genome Browsers

- NCBI

http://www.ncbi.nlm.nih.gov/mapview/map_search.cgi

- UCSC

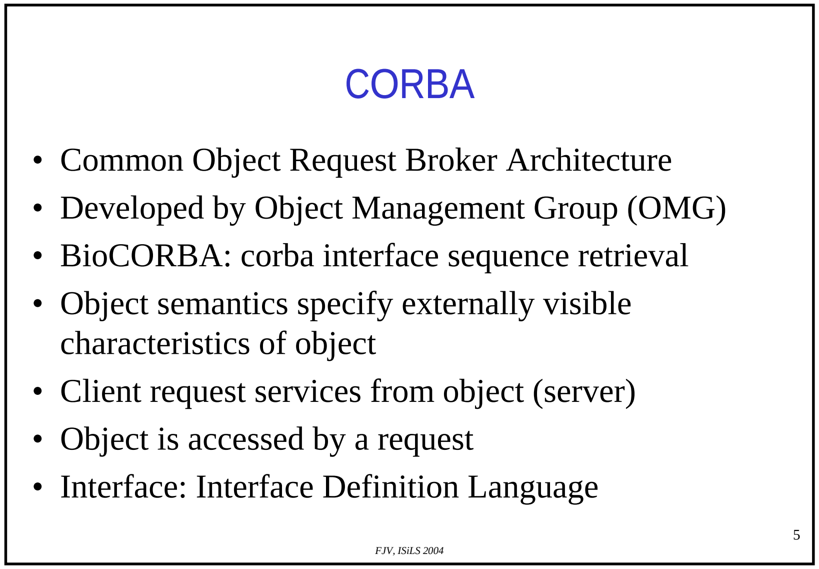
http://genome.ucsc.edu/

- Ensembl

http://www.ensembl.org/

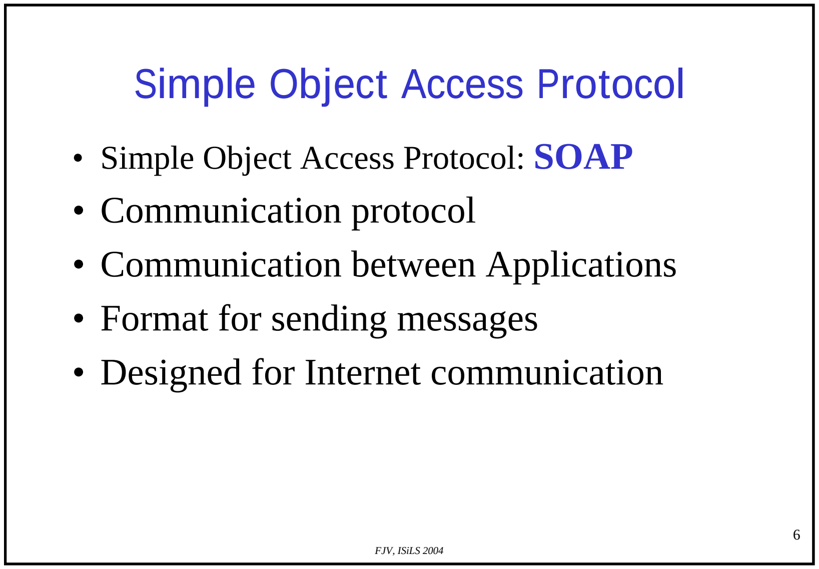
Data Integration

- BioCORBA
  - Communication on the object level
- BioXML
  - Communication on the data level
  - Parameter (data values) transmission: SOAP
- BioDAS
  - Communication on the network level
- BioSQL
  - Consensus data schemas



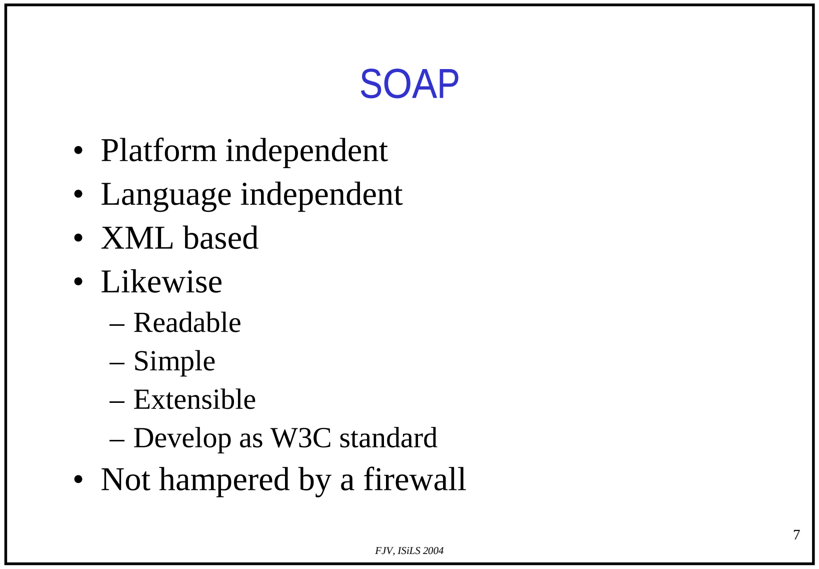
CORBA

- Common Object Request Broker Architecture
- Developed by Object Management Group (OMG)
- BioCORBA: corba interface sequence retrieval
- Object semantics specify externally visible characteristics of object
- Client request services from object (server)
- Object is accessed by a request
- Interface: Interface Definition Language



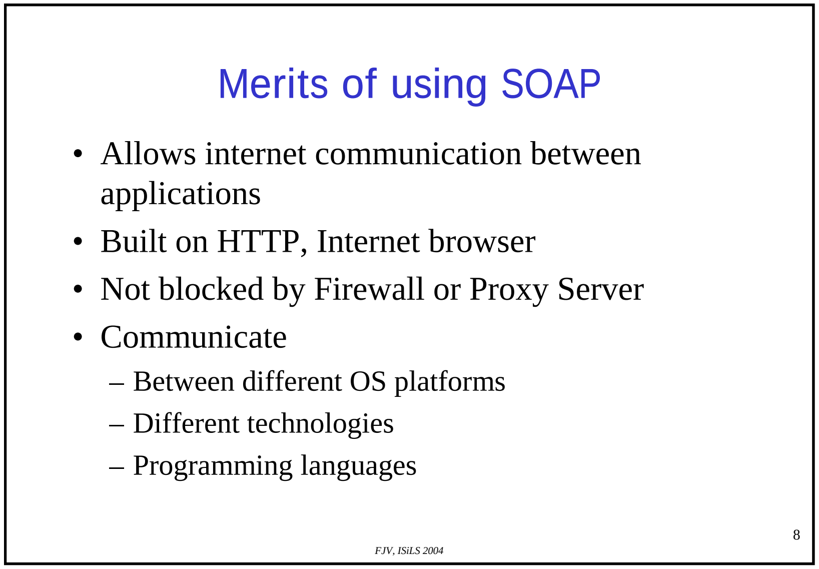
Simple Object Access Protocol

- Simple Object Access Protocol: **SOAP**
- Communication protocol
- Communication between Applications
- Format for sending messages
- Designed for Internet communication



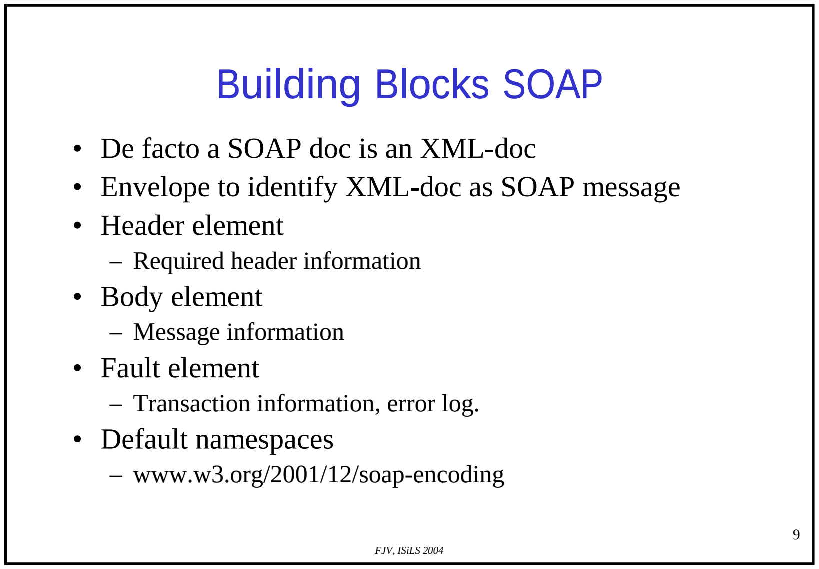
SOAP

- Platform independent
- Language independent
- XML based
- Likewise
  - Readable
  - Simple
  - Extensible
  - Develop as W3C standard
- Not hampered by a firewall



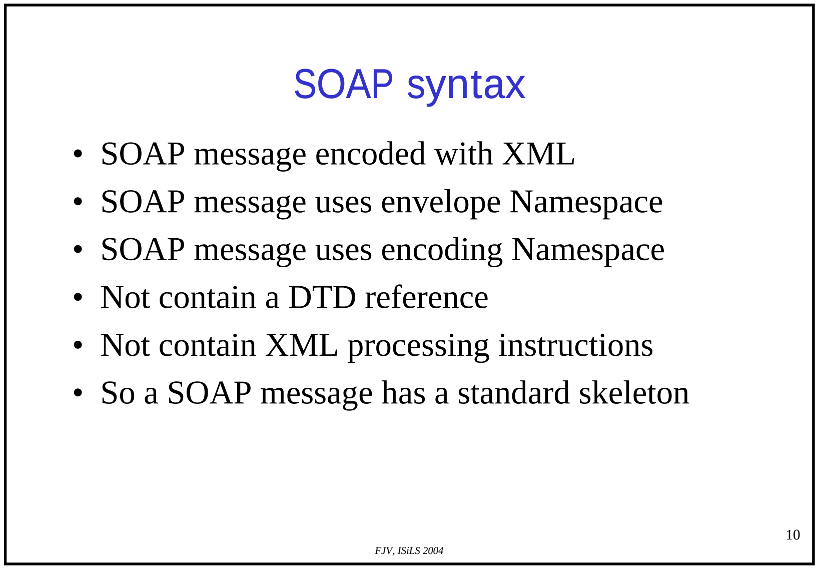
Merits of using SOAP

- Allows internet communication between applications
- Built on HTTP, Internet browser
- Not blocked by Firewall or Proxy Server
- Communicate
  - Between different OS platforms
  - Different technologies
  - Programming languages



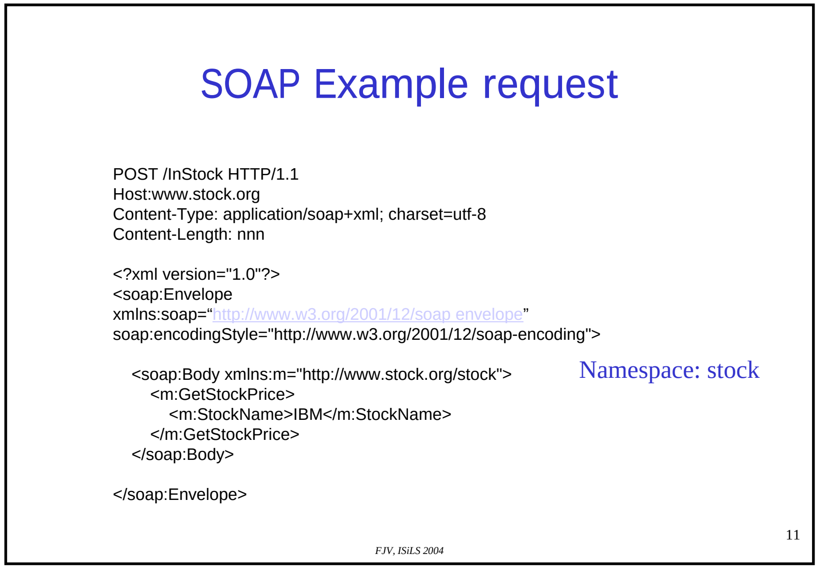
Building Blocks SOAP

- De facto a SOAP doc is an XML-doc
- Envelope to identify XML-doc as SOAP message
- Header element
  - Required header information
- Body element
  - Message information
- Fault element
  - Transaction information, error log.
- Default namespaces
  - www.w3.org/2001/12/soap-encoding



SOAP syntax

- SOAP message encoded with XML
- SOAP message uses envelope Namespace
- SOAP message uses encoding Namespace
- Not contain a DTD reference
- Not contain XML processing instructions
- So a SOAP message has a standard skeleton



SOAP Example request

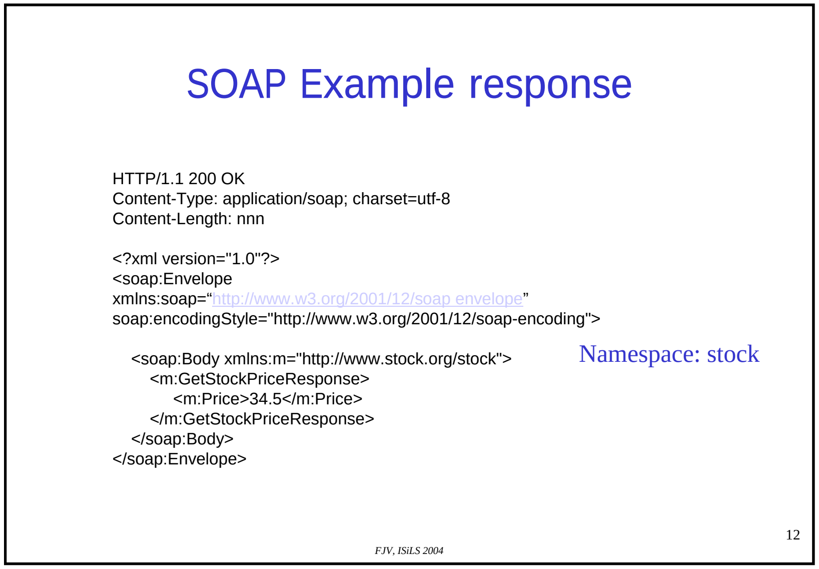
```
POST /InStock HTTP/1.1
Host:www.stock.org
Content-Type: application/soap+xml; charset=utf-8
Content-Length: nnn

<?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 xmlns:m="http://www.stock.org/stock">
    <m:GetStockPrice>
      <m:StockName>IBM</m:StockName>
    </m:GetStockPrice>
  </soap:Body>

</soap:Envelope>
```

Namespace: stock



SOAP Example response

```
HTTP/1.1 200 OK
Content-Type: application/soap; charset=utf-8
Content-Length: nnn

<?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 xmlns:m="http://www.stock.org/stock">
    <m:GetStockPriceResponse>
      <m:Price>34.5</m:Price>
    </m:GetStockPriceResponse>
  </soap:Body>
</soap:Envelope>
```

Namespace: stock

Distributed Annotation System (DAS)

- DAS is a client-server system
- Client integrates information from multiple servers.
- Single machine
  - gathers genome annotation info from multiple web sites,
  - collates the information, and
  - displays it to the user in a single view.
- Requires little coordination among the information providers.

- http://www.biodas.org/documents/rationale.html

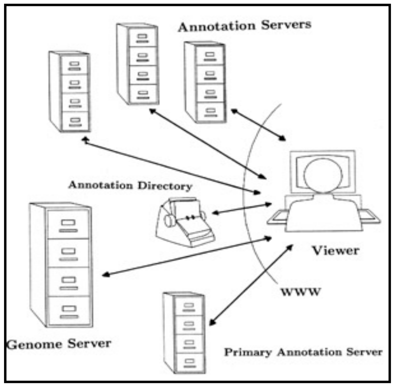
- http://www.biodas.org/documents/msproposal.html

DAS

- Distributed Annotation System
  - – Lincoln Stein (CSHL), Robin Dowell (Wash U)
- The "genome annotation napster"
- Consensus communication protocol
- On top of HTTP (DAS/1)
- DAS/1 – stable; DAS/2 – RFC process

- http://www.biodas.org/

Schematic view of DAS



DAS request

DAS Request

Form of a URL.

- URL has a site-specific prefix.
- DAS: followed by a standardized path and query string.

- Standardized path begins with the string **/das** .
- Followed by URL components containing
  - data source name
  - a command.
- Example:

```
http://www.wormbase.org/db/das/elegans/features?segment=CHROMOSOME_I:1000,2000
^^^^^^^^^^^^^^^^^^^^^^^^^^ ^^^ ^^^^^^^ ^^^^^^^^ ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
     site-specific prefix    das  data   command  arguments
```

DAS response

DAS Response

Response from the server to client consists of

- standard HTTP header
- with DAS status info within that header
- followed optionally by an XML file
- XML contains the answer to the query

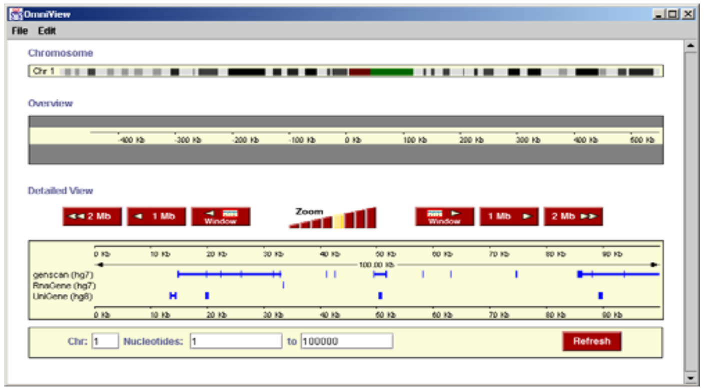
•DAS status the header

- consists of two lines.
- 1: X-DAS-Version, current protocol version number: DAS/1.0.
- 2: X-DAS-Status and contains a three digit status code
- indicates the outcome of the request.

`Example HTTP header`:

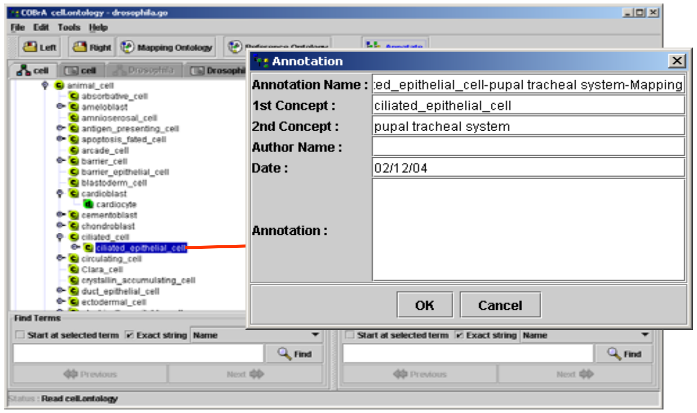
```
HTTP/1.1 200 OK     Date: Sun, 12 Mar 2000 16:13:51 GMT
Server: Apache/1.3.6 (Unix) mod_perl/1.19
Last-Modified: Fri, 18 Feb 2000 20:57:52 GMT Connection: close
Content-Type: text/plain
X-DAS-Version: DAS/1.5X-DAS-Status: 200 X-DAS-Capabilities: error-segment/1.0;
 unknown-segment/1.0; unknown-feature/1.0; ... data follows...
```

Java DasViewer



Chr: 1 Nucleotides: 1 to 100000

Ontologies, mapping & integrating



Annotation Name : ed_epithelial_cell-pupal tracheal system-Mapping
1st Concept : ciliated_epithelial_cell
2nd Concept : pupal tracheal system
Author Name :
Date : 02/12/04
Annotation :