Bioinformatics Ontology: Towards the Automatics Generation of Bioinformatics Workflow for Web Services

Konagaya Akihiko

Project Director
Advanced Genome Information Technology Research Group
RIKEN Genomic Sciences Center
Contents

• Introduction of Ontology
• Web Services for Bioinformatics
• Automatics Workflow Generation
• Lessons from our First Experience
Introduction of Ontology
Tacit and Explicit Knowledge

We should start from the fact that 'we can know more than we can tell'.

Michael Polanyi, “The Tacit Dimension” 1967

Michael Polanyi (1891-1976)
Rainbow Color

How many colors can you see in rainbow?
Ontology for Rainbow Colors

From 360 nm～400 nm

Purple
RGB Value #800080

Indigo
#000080

Blue
#0000FF

Green
#008000

Yellow
#FFFF00

Orange
#FF8000

Red
#FF0000

to 760 nm～830 nm

All the colors you can see with your own eyes!
Which are Purple?

- #800050
- #700080
- #800060
- #800080
- #600080
- #800070
- #500080
Representation by Elements and Constructor

Purple

Blue Element

Red Element
Web Services for Bioinformatics
Formulation of Community
Bioinformatics Web Services on Grid

GRIDIFIED
BLAST, FASTA, ClustalW, Glimmer2, InterProScan,

http://jkt.gsc.riken.jp/sp/spbio/wslist.jsf
Advantages of Web Services

- Liberating from the maintenance of biological databases and tools
- Scalability of computational resources
- High-level application programming interface

Web Services

```
Task A        Task B        Task C        Task D
\arrow{input} \arrow{computing} \arrow{computing} \arrow{computing} \arrow{output}
   DB X   \arrow{input} \arrow{computing} \arrow{computing} \arrow{computing} \arrow{output}
   DB Y   \arrow{input} \arrow{computing} \arrow{computing} \arrow{output}
   DB Z   \arrow{input} \arrow{computing} \arrow{output}
```
Very Simple Work Flow

1. Sequence
2. BLAST Search
3. GetEntry
4. CLUSTAL W
5. Tree View

UniProt

Hittable

Sequences

Multiple Alignment

Phylogenetic tree
Manual Workflow on Web Apps
Web Service Programming

#!/usr/bin/perl

use SOAP::Lite;

# SOAP API
# specify WSDL
my $service = SOAP::Lite-> service('http://xml.nig.ac.jp/wsdI/GetEntry.wsdI');

# call web service
$result = $service->getXML_DDBJEntry("AB000003");

# print result
print $result;

http://www.xml.nig.ac.jp/perl.txt
Why don’t we use workflow tools?

http://www.cyclonic.org/Taverna_and_myGrid.ppt
Needs Automatic Workflow
Generate Tool from
Very High Level Specification

apply **Blastp** to **UniProt**

**GetEntry** from **UniProt**

apply **CLUSTALW**

apply **TreeView**

Automatics Generation

Workflow for Bioinformatics Web Services
Automatic Generation of Bioinformatics Workflow
Task as Atomic Component of Workflow

Input Data Specification
sample
{aa_sequence, fasta}

Application
sample
{blastp DAD}

Output Data Specification
sample
{djjentry, flatfile}
Workflow as a Sequence of Tasks
Automatic Generation of Workflow from Given Input and Output Data Specification and Tasks

• Path Finding using Meta Information
Meta Information to Specify the Functionality of Task

**TASK**

- **Meta Data for Database**
  - samples
    - {uniprot}
    - {nt}

- **Meta Data for Input**
  - samples
    - {na_sequence,fasta}
    - {aa_sequence,fast}

- **Meta Information for Command and Options**
  - {blastn}
  - {getentry}

- **Meta Data for Output**
  - sample
    - {ddbjentry,flatfile}
    - {aablastentry,hittable}
Task Hierarchy (is_a)

- **Abstract Task**
  - Homology search

- **Concrete Task**
  - BLAST
  - FASTA
  - SSEARCH

- **Input Type**:
  - S: Sequence or Sequence Name
  - N: Nucleoside Sequence
  - A: Amino acid Sequence
  - id: Accession ID

- **Output Type**:
  - O: Various Type
  - E: Database Entry
Task Hierarchy (has_a)
Prototype for ‘Proof of Concept’

• **Language** tuProlog
  – Java to Prolog
  – Prolog to Java
    • Web Service Interface through JAVA API

• **Task Database**
  – Prolog Clause Database

• **Optimal Path Finding**
  – Bidirectional Breadth First Search Algorithm
System Overview

Workflow System

- User Specification
- Workflow
- Workflow Execution
- Result

User Data

Knowledge Base

- Task Database (1697)
- Web Service Information (1596)

Server

- DDBJ
- SPBIO

UI

Prolog Engine
  - tuProlog

Web Service Library
  - Java

Workflow Library
  - prolog
Screen Snapshot
(Workflow Generation Phase)
Screen Snapshot
(Workflow Execution Phase)
Obtained Phylogenetic Tree by a generated workflow when applying to a Human Insulin Sequence

- P61982 [Mus musculus]
- P61981 [Homo sapiens]
- Q5RC20 [Pongo pygmaeus]
- P61983 [Rattus norvegicus]
- Q5F3W6 [Gallus gallus]
- P68252 [Bos taurus]
- Q6PCG0 [Xenopus laevis]
- Q6NRY9 [Xenopus laevis]
- Q04917 [Homo sapiens]
- P68509 [Bos taurus]
- P68511 [Rattus norvegicus]
- P68510 [Mus musculus]
- Q6UFZ2 [Oncorhynchus mykiss]
- Q6PC29 [Brachydanio rerio]
- Q6UFZ3 [Oncorhynchus mykiss]
Lessons from our First Experience
Task Database (prototype)

Web Service Call

DDBJ Blast  453
DDBJ SRS  638
DDBJ GetEntry  38
DDBJ ClustalW  62
SPBIO Blast  405

Format Transformation  56
Data Selection  45

In Total  1697
# Test Set of Specification

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

Akihiko Konagaya (NETTAB2006), S. Margherita di Pula, Italy, July 2006
### Differences of Generated Workflow

<table>
<thead>
<tr>
<th>Meta Data</th>
<th>Input</th>
<th>Database</th>
<th>Output</th>
<th>Full Cmds</th>
<th>No input</th>
<th>Database</th>
<th>No output</th>
<th>Full Cmds</th>
<th>Input</th>
<th>No DB</th>
<th>Output</th>
<th>Partial Cmds</th>
<th>input</th>
<th>No DB</th>
<th>No output</th>
<th>Partial Cmds</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Why Failed?

Amino Acid Sequence → blastp → HitTable

Lack of Interoperability Between the Web Services

Amino Acid Sequence → blastp → HitTable

DAD

UNIPROT
Very Similar but not the Same Format

Blastp for UniProt

L15440-1|AAA59179.1| 107|Homo sapiens insulin protein. 177 1e-43

Blastp for DAD

sp|Q8HXV2|INS_PONPY Insulin precursor [Contains: Insulin B chain... 171 4e-43
Conclusion

• **Web Services** have great potential to share Bioinformatics Data and Tools in all over the world

• Needs **Automatic Workflow Generation Tools** to make full use of Web Services

• **Bioinformatics Ontology** is a key to establish Interoperability among Bioinformatics Web Services
Acknowledgement

• Daisuke Shinbara  Tokyo Institute of Technology (Hitachi, ltd.)
• Sumi Yoshikawa  RIKEN GSC, TITECH

References


Thank You for Listening