

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

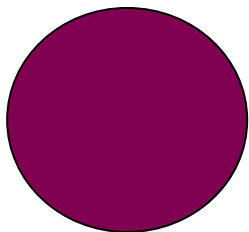
How many colors can you see in rainbow?



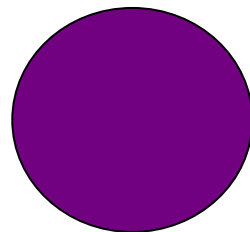
Ontology for Rainbow Colors



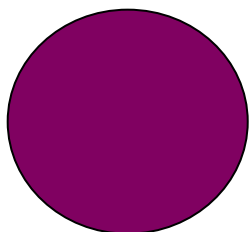
Which are Purple?



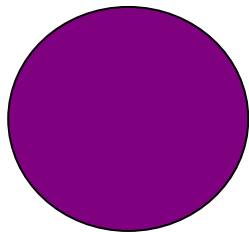
#800050



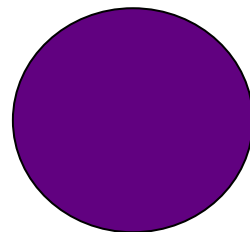
#700080



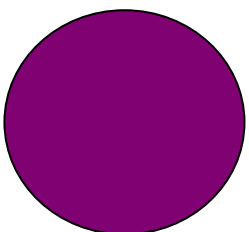
#800060



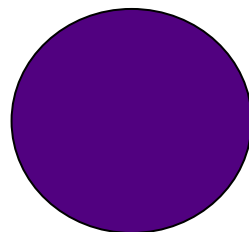
#800080



#600080

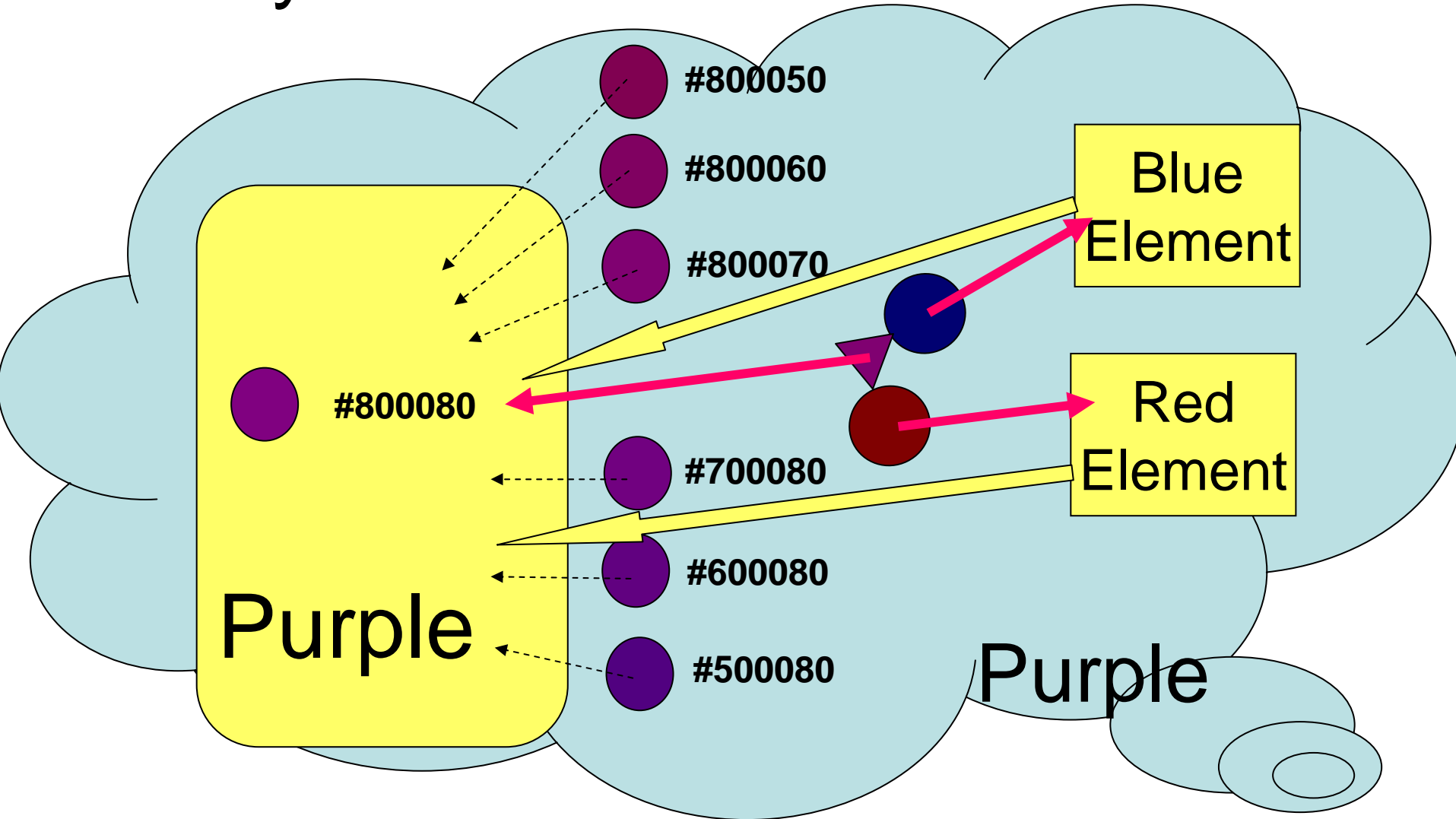


#800070



#500080

Representation by Elements and Constructor



Web Services for Bioinformatics

Formulation of Community



Wakayama University



和歌山大学

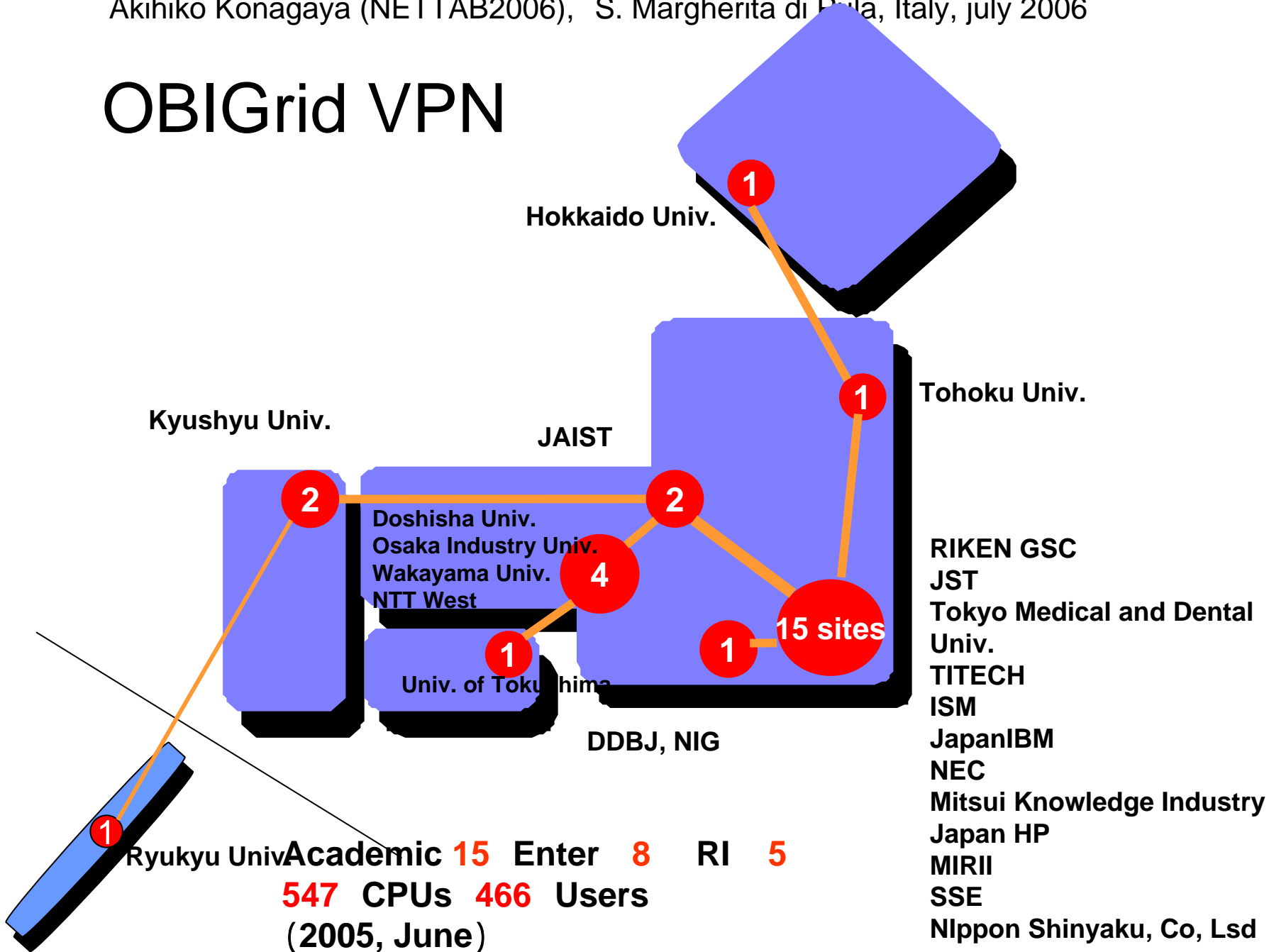


invent

三菱総合研究所



OBIGrid VPN



Bioinformatics Web Services on Grid



GRIDIFIED

BLAST, FASTA, ClustalW,
Glimmer2, InterProScan,

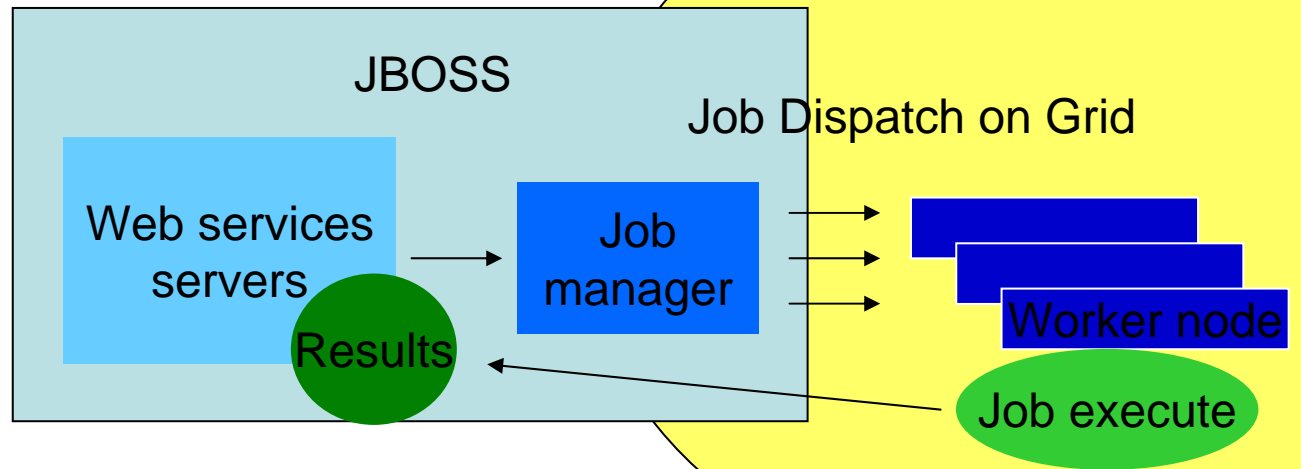
<http://jkt.gsc.riken.jp/sp/spbio/wslist.jsf>

Client



call web services

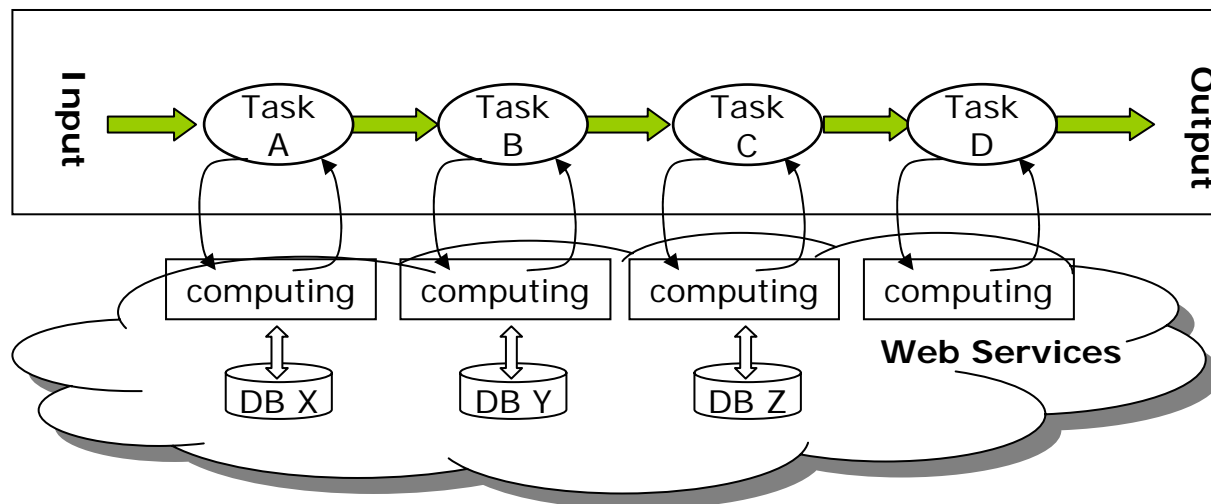
Return



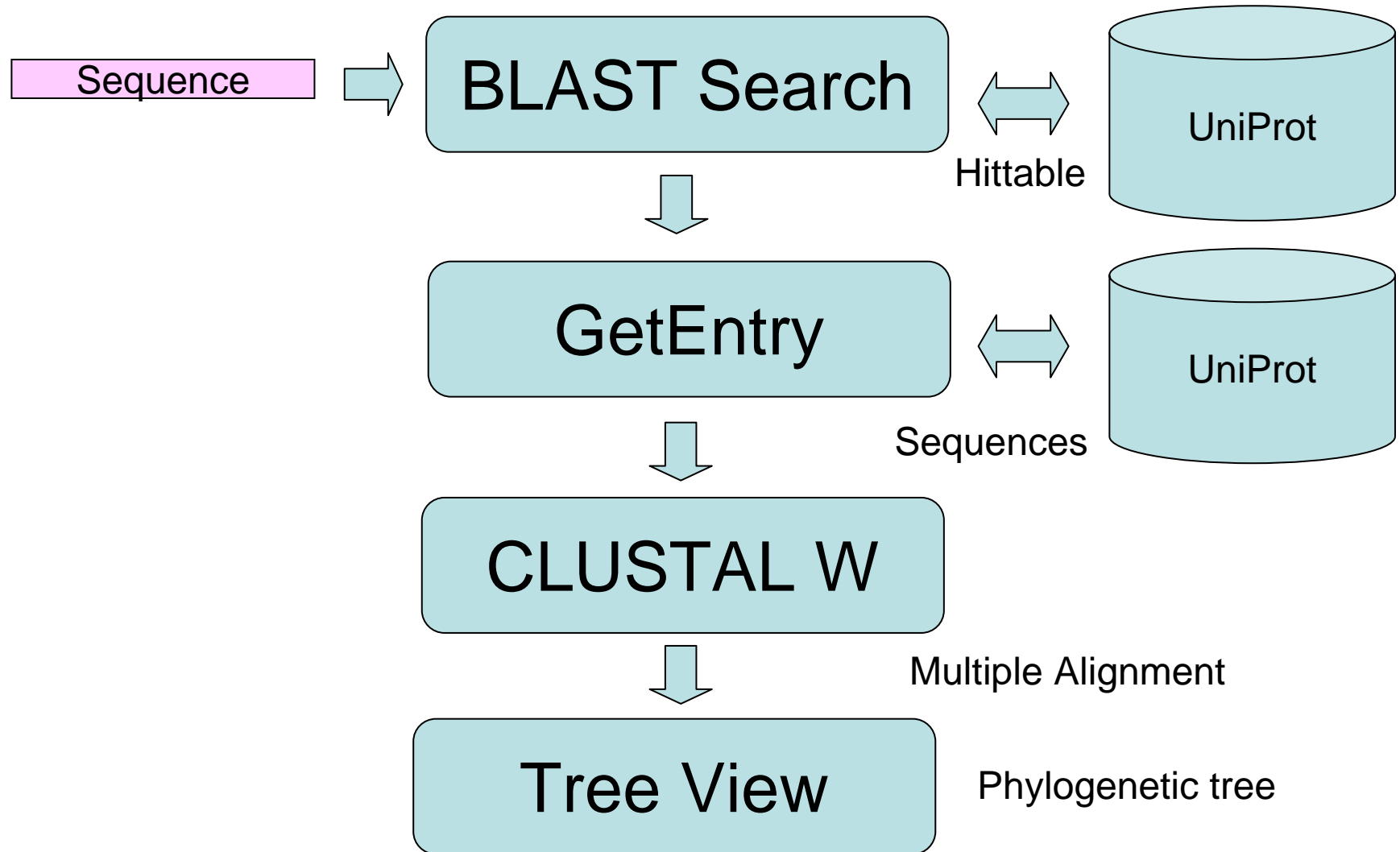
Advantages of Web Services

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

Web Services



Very Simple Work Flow



Manual Workflow on Web Apps

The image illustrates a manual workflow on web applications for sequence analysis. It consists of several overlapping screenshots from a Microsoft Internet Explorer browser:

- BLAST Search Results:** A screenshot of the DDBJ BLAST search results page, showing search parameters and results.
- getentry Tool:** A screenshot of the 'getentry' tool, which is used to retrieve sequence data from the DDBJ database by accession number.
- ClustalW Alignment:** A screenshot of the ClustalW alignment tool interface. It shows various options for alignment, such as 'SHOW ALIGNMENT SCORE' and 'DOTSINOUTPUT'. The 'ALIGN' section is visible, and the 'DOTSINOUTPUT' option is highlighted.
- PHYLIP Tree:** A screenshot of a PHYLIP tree visualization. The tree shows the relationships between several species: Rattus norvegicus, Mus musculus, Homo sapiens, Takifugu rubripes, Xenopus laevis, Gallus gallus, Anopheles gambiae, and Danio rerio. The tree is rooted and shows the evolutionary relationships between these species.

Web Service Programming

```
#!/usr/bin/perl

use SOAP::Lite;

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

# 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?

The screenshot displays the Taverna Workbench interface with several key components:

- Enactor invocation window:** Contains a table of processor statuses.
- Processor status table:**

Type	Name	Last event	Event timestamp	Event detail
	Blast2_program	ProcessComplete	28-Jul-2004 11:37...	
	comparer	ProcessComplete	28-Jul-2004 11:39...	
	Fasta_to_numbered	ProcessComplete	28-Jul-2004 11:39...	
	simplifier	ProcessComplete	28-Jul-2004 11:39...	
	ncbiblast	ProcessComplete	28-Jul-2004 11:39...	
	repeatmasker	ProcessComplete	28-Jul-2004 11:38...	
	retrieve	ProcessComplete	28-Jul-2004 11:39...	
	copyright	ProcessComplete	28-Jul-2004 11:37...	
	blast2	ProcessComplete	28-Jul-2004 11:39...	
	lister	ProcessComplete	28-Jul-2004 11:39...	

- Workflow graph:** A network diagram showing the flow of data between various workflow objects and processors.
- Advanced model explorer:** A tree view of the workflow model, including inputs, outputs, and processors.
- Available services:** A list of external services and processors available to the workflow.
- Run Workflow window:** A dialog for managing workflow inputs and outputs, with a 'Run Workflow' button.

Needs Automatic Workflow Generate Tool from Very High Level Specification

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

GetEntry from **UniProt**

apply **CLUSTALW**

apply **TreeView**

Automatics
Generation

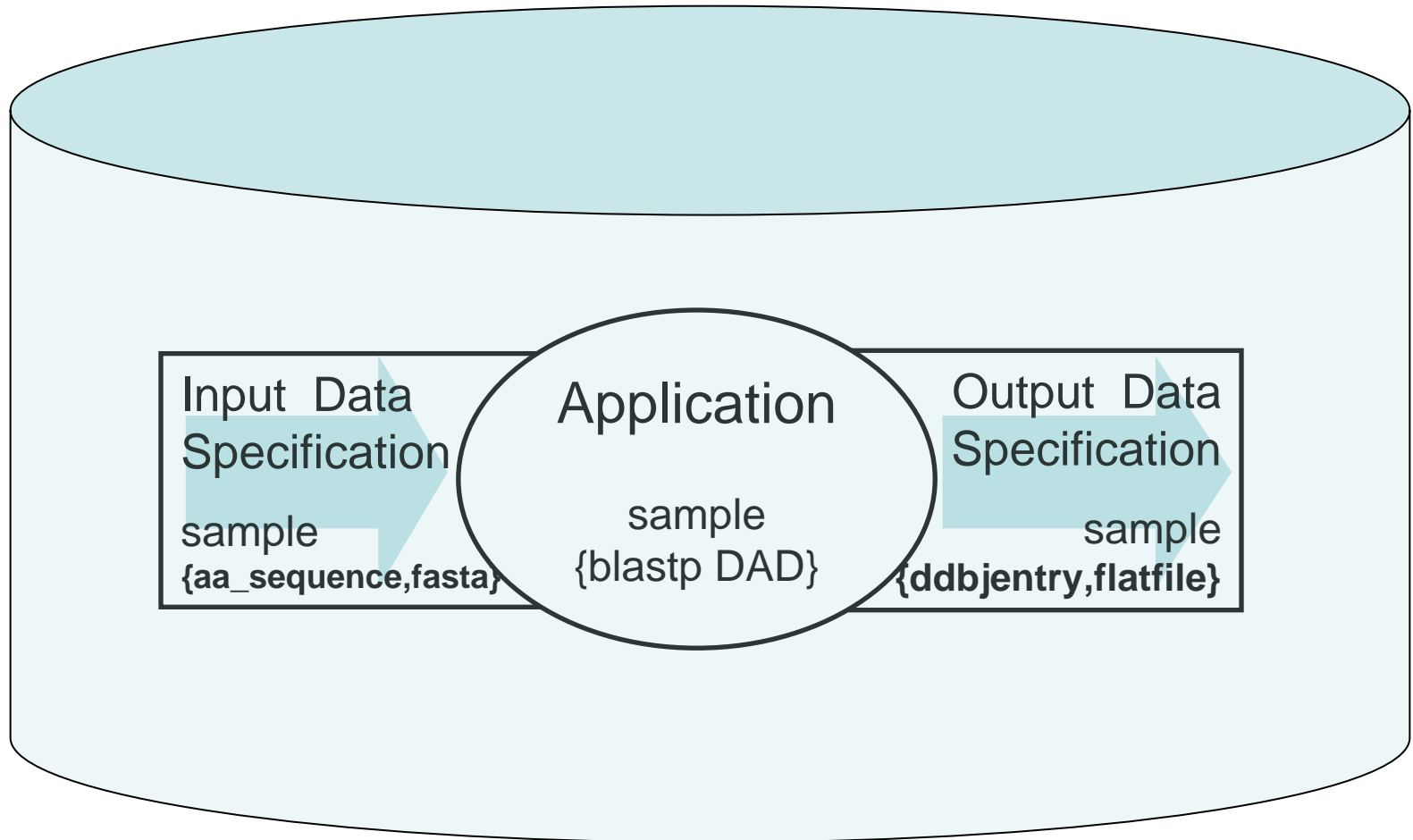


?

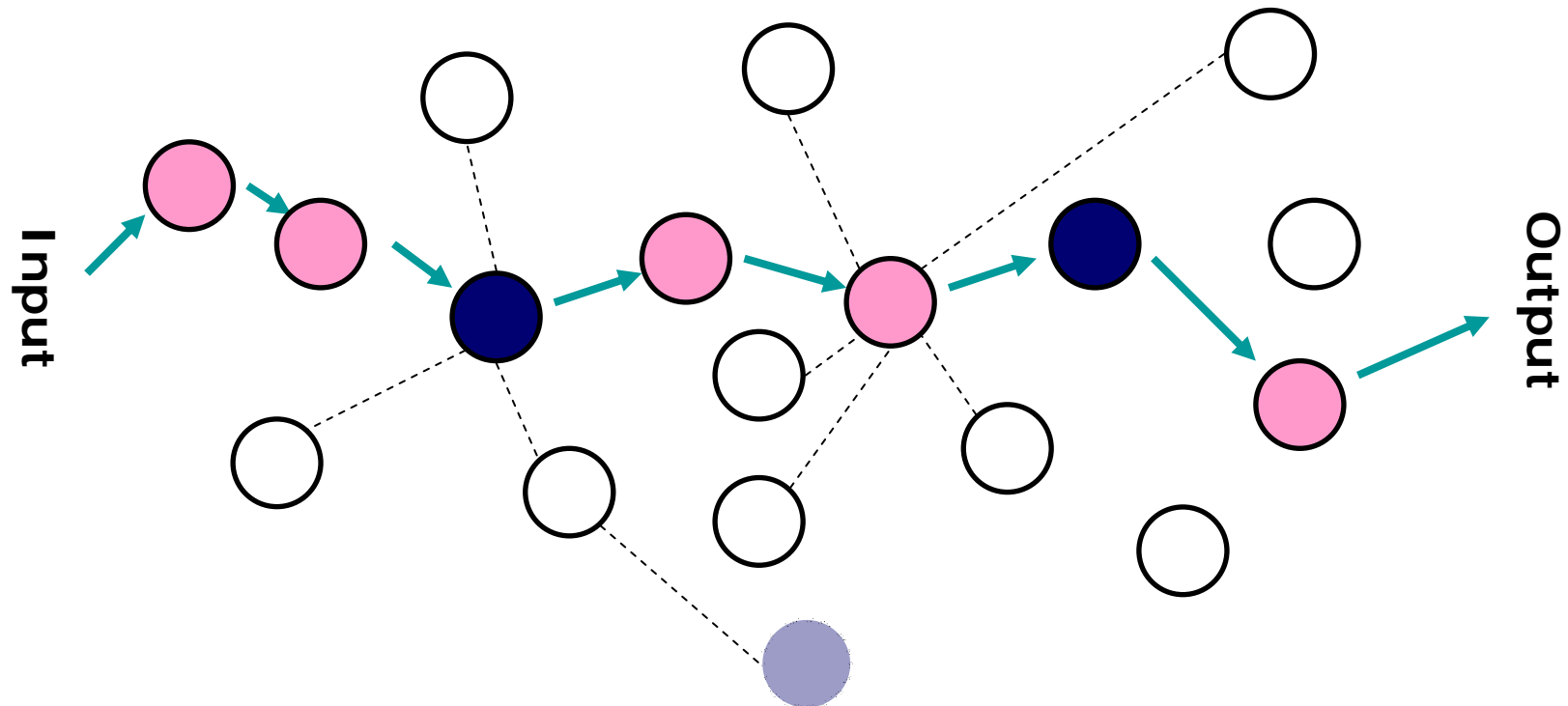
**Workflow
for
Bioinformatics Web Services**

Automatic Generation of Bioinformatics Workflow

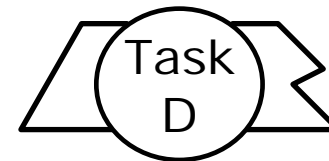
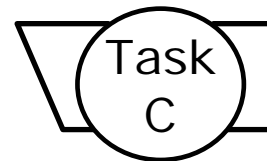
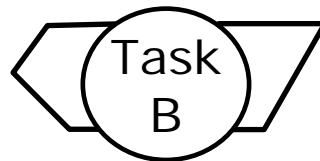
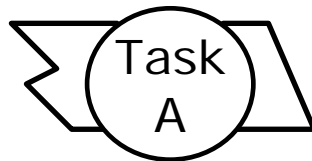
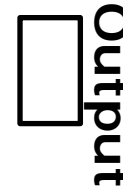
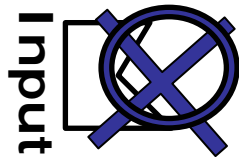
Task as Atomic Component of Workflow



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 Information
for Command and
Options

{blastn}
{getentry}

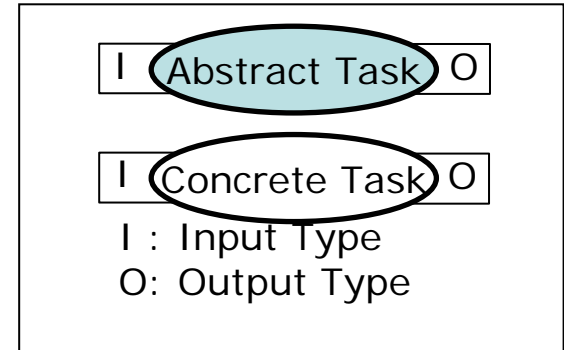
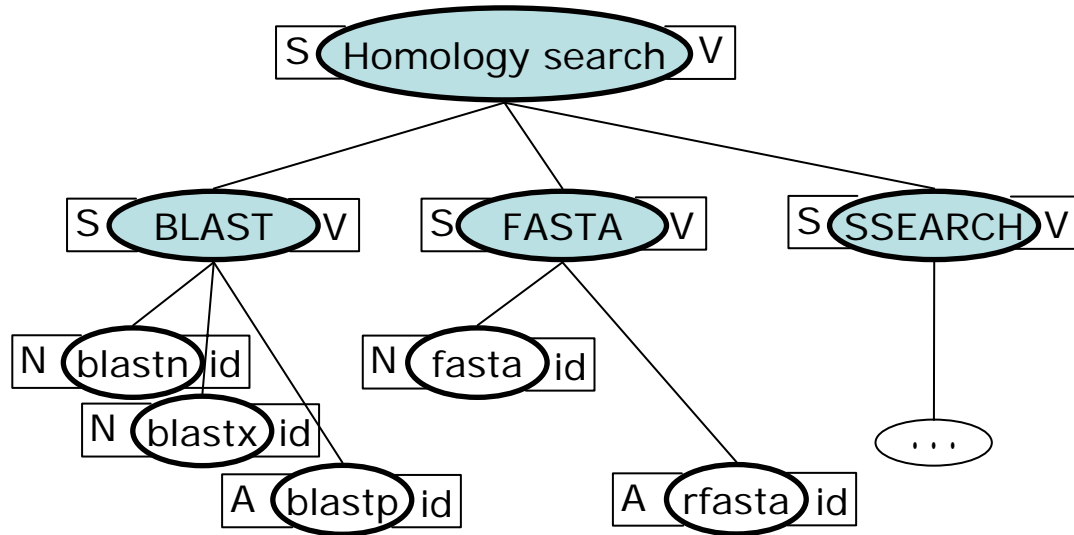
Meta Data
for Input

samples
{na_sequence,fasta}
{aa_sequence,fast}

Meta Data
for Output

sample
{ddbjentry,flatfile}
{ablastentry,hittable}

Task Hierarchy (is_a)



S : Sequence or
Sequence Name

V : Various Type

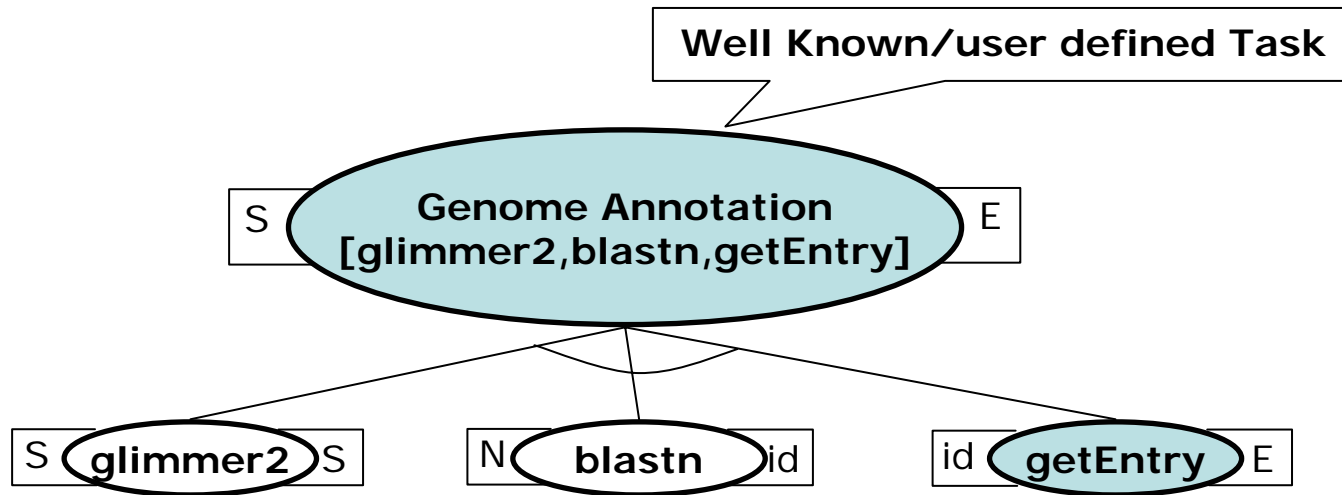
N : Nucleoside
Sequence

A : Amino acid
Sequence

id : Accession ID

E : Database Entry

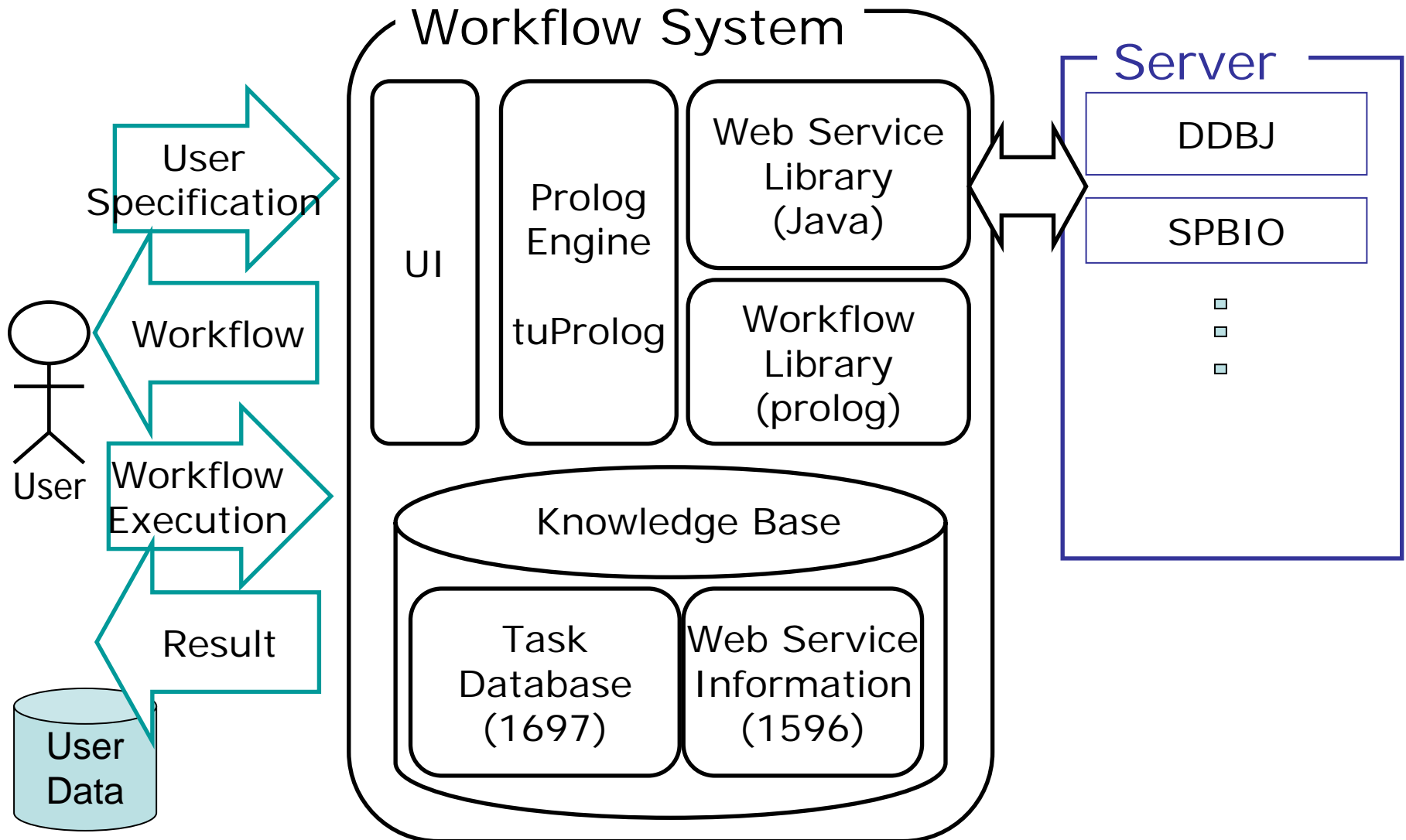
Task Hierarchy (has_a)



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



Screen Snapshot (Workflow Generation Phase)

The screenshot displays the KONALOG software interface during the workflow generation phase. The main window is titled "KONALOG" and has a menu bar with "file" and "configure".

There are two main panels:

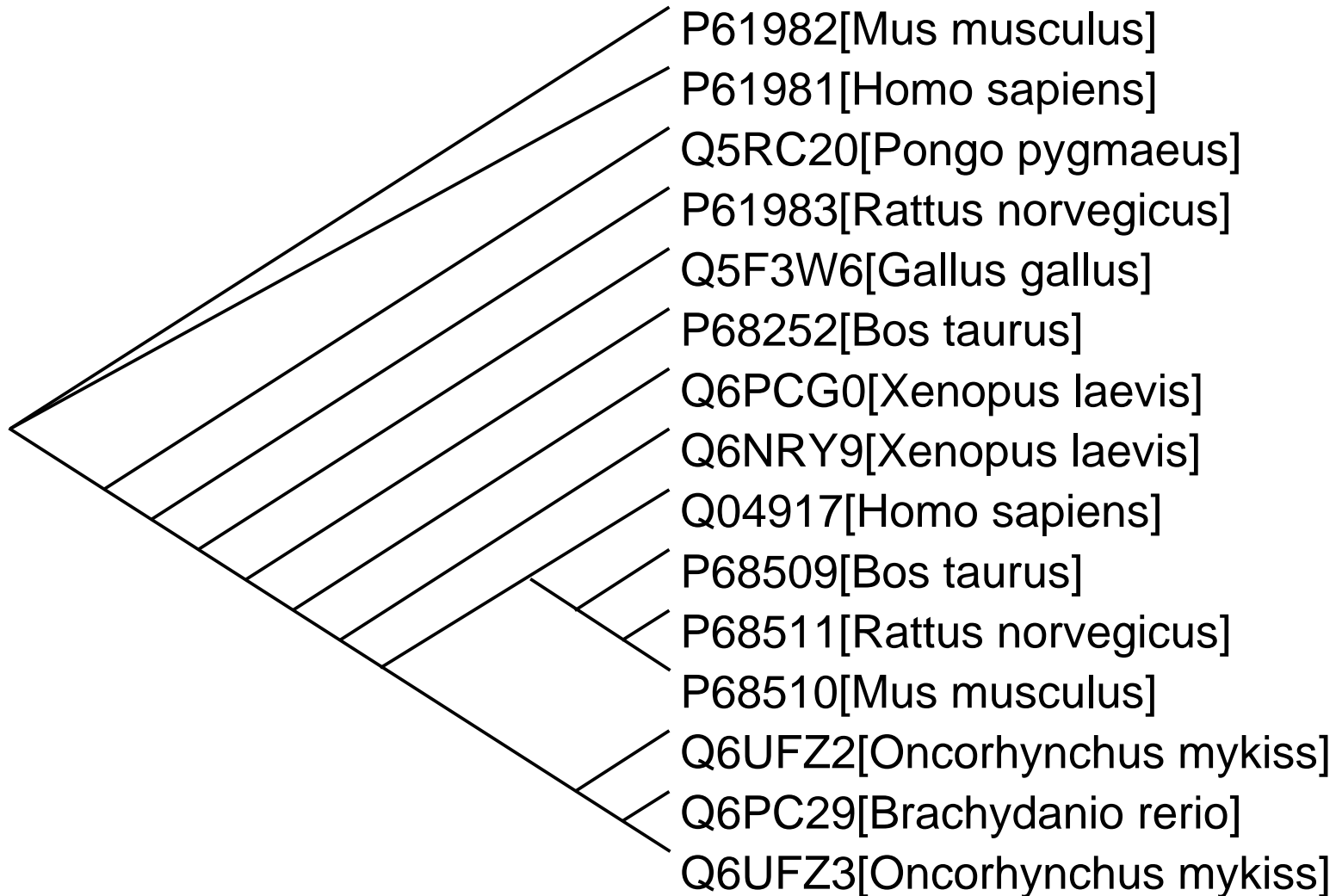
- Workflow Search:** This panel is used to define search criteria. It has a "file" menu and two input sections:
 - Input:** Type is "aasequence" and Format is "fasta".
 - Output:** Type is "phylogenetictree" and Format is "dnd".
- Workflow Search Result:** This panel shows the details of a search result for a specific node. It includes:
 - Node ID:** [1052,102,104,109,10005,3107,129,4001,200]
 - Input Format:** [aasequence,fasta] and **Output Format:** [phylogenetictree,dnd]
 - Description:** alignment by blastp from UNI identify data format and conte extract SeqIdentifier from ddb extract Swiss-plot ACNumbe idlist[25] from idlist[??] Get SWISSPROT entry of FA multi fasta format from fasta l multiplealignment by clustalw get dnd from multiple alignme
 - Operation:** webService(ddbj,'Blast',searchParam,_20),webServiceCall(_20,[splitString(_33,'\\n',_46) map(',','splitString(_49,'\\t\\s',_48),element(2,_48,_47)),_49,_47, map(',','splitString(_65,'[]',_64),get_next_by_key(_64,sp,_63)),_ list_reduce(_78,25,_91) map(',','webService(ddbj,'GetEntry',getFASTA_SWISSEntry,_94 list_concat_with_dlmt(_107,',',_120) webService(ddbj,'ClustalW',analyzeParam,_121),webServiceCal splitString(_135,'-----',_134),element(4,_134,_136)

At the bottom of the main window, there is a "Search Workflow" button and a "Ready." status indicator.

Screen Snapshot (Workflow Execution Phase)

The screenshot displays the KONALOG Workflow Execution Console interface. The main window is titled "KONALOG" and contains a "Workflow Execution Console" pane. The "Input Data" section shows a protein sequence: `>sp|Q13541|4EBP1_HUMAN Eukaryotic translation initiation factor 4E-binding protein 1 (4E-SGGSSCSQTPSRAIPATRRVVLGDGVQLPPGDYSTTPGGTLFSTTPGGTRIYDRKFLME CRNSPVTKTPPRDLPTIPGVTPSSDEPPMEASQSHLRNSPEDKRAGGEESQFEMDI`. Below the input data is an "Execute" button. The "Progress" section shows a list of steps: "alignment by blastp from UNIPROT", "identify data format and content", "extract SeqIdentifier from ddbj BLAS...", "extract Swiss-plot ACNumber from ...", "idlist[25] from idlist[??]", "Get SWISSPROT entry of FASTA Fo...", "multi fasta format from fasta list", "multiplealignment by clustalw with bl...", and "get dnd from multiple alignment result". The "Intermediate Data" section shows a list of 1052 entries, each with a species identifier and a protein accession number, such as `'sp|Q13541|4EBP1_HUMAN sp|Q13541|4EBP1_HUM`, `sp|Q62622|4EBP1_RAT`, `sp|Q60876|4EBP1_MOU`, `tr|Q3TII9|Q3TII9_MOUSE`, `tr|Q9BG57|Q9BG57_PIG`, `tr|Q6PFS8|Q6PFS8_BR/`, `tr|Q3UFP6|Q3UFP6_MOI`, `sp|P70445|4EBP2_MOU:`, `tr|Q497A9|Q497A9_RAT`, `sp|Q13542|4EBP2_HUM`, `tr|Q3UZD4|Q3UZD4_MOU`, and `tr|Q6FG68|Q6FG68_HUM`. The "Result Data" section is currently empty.

Obtained Phylogenetic Tree by a generated workflow when applying to a Human Insulin Sequence



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

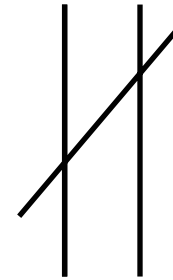
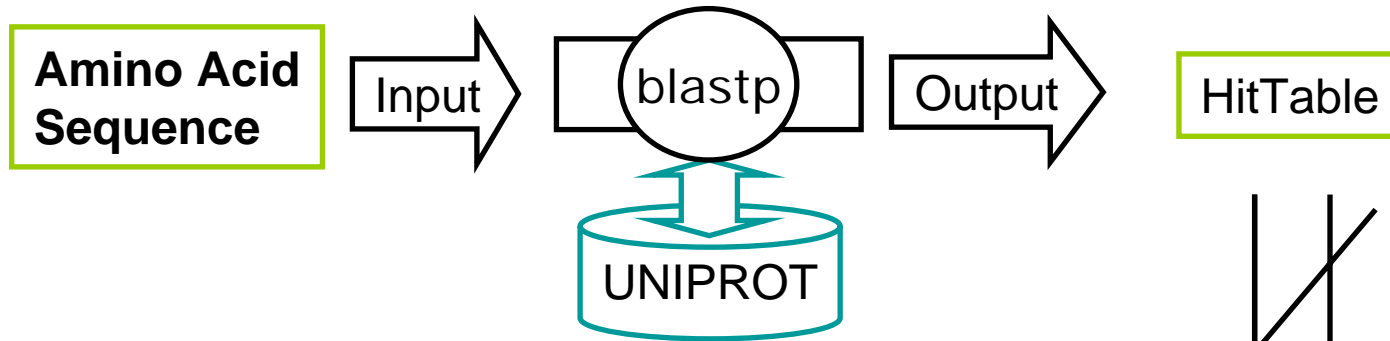
Test Set of Specification

No	Workflow				Applications
	Input		Output		
	Format	Type	Format	Type	
1	fasta	aasequence	gde	aamultiplealignment	blastp uniprot
					filter num25
					getfasta_swissentry
					multiplealignment
2					blastp uniprot
					filter num25
					getfasta_swissentry
					multiplealignment
3	fasta	aasequence	gde	aamultiplealignment	alignmentsearch
					filter
					getentry
					multiplealignment
4	fasta	aasequence			filter
					getentry
					multiplealignment

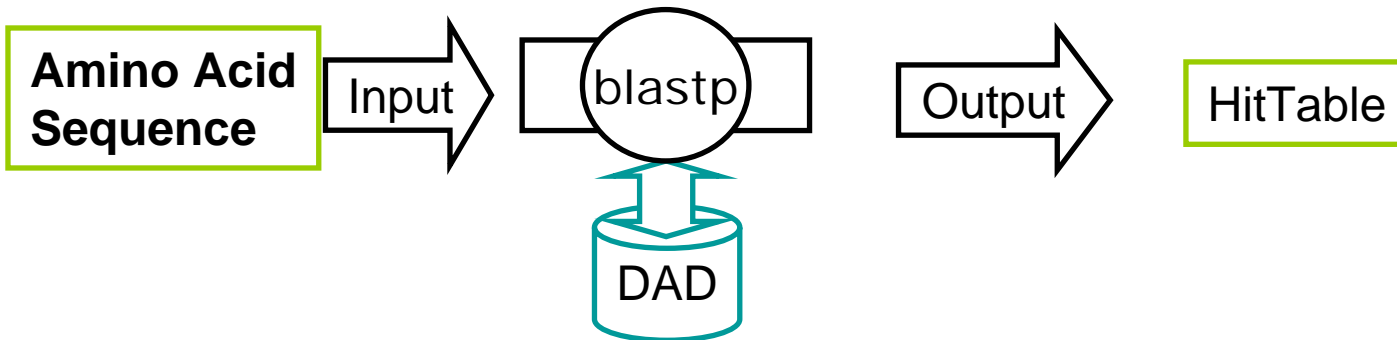
Differences of Generated Workflow

Meta Data	No.	Solution Num	time(ms)	First Match WebserviceCall		
				ID	Description	
Input Database Output Full Cmds	1	8	11266	1052	alignment by blastp from UNIPROT	○
				102	identify data format and content.	
				104	extract SeqIdentifier from ddbj BLAST result record	
				109	extract Swiss-plot ACNumber from SequenceIdentifier	
				10005	idlist[25] from idlist[??]	
				3107	Get SWISSPROT entry of FASTA Format by Accession Number.	
				129	multi fasta format from fasta list	
				4018	multiplealignment by clustalw with blosum	
No input Database No output Full Cmds	2	41	46704	1052	alignment by blastp from UNIPROT	○
				102	identify data format and content.	
				104	extract SeqIdentifier from ddbj BLAST result record	
				109	extract Swiss-plot ACNumber from SequenceIdentifier	
				1005	idlist[25] from idlist[??]	
				3107	Get SWISSPROT entry of FASTA Format by Accession Number.	
				129	multi fasta format from fasta list	
				4001	multiplealignment by clustalw with blosum	
Input No DB Output Partial Cmds	3	100 over	249906	1043	alignment by blastp from DAD	X?
				102	identify data format and content.	
				104	extract SeqIdentifier from ddbj BLAST result record	
				109	extract Swiss-plot ACNumber from SequenceIdentifier	
				10001	idlist[25] from idlist[??]	
				3107	Get SWISSPROT entry of FASTA Format by Accession Number.	
				129	multi fasta format from fasta list	
				4018	multiplealignment by clustalw with blosum	
input No DB No output Partial Cmds	4	100 over	25297	1043	alignment by blastp from DAD	X?
				102	identify data format and content.	
				104	extract SeqIdentifier from ddbj BLAST result record	
				109	extract Swiss-plot ACNumber from SequenceIdentifier	
				10001	idlist[5] from idlist[??]	
				3107	Get SWISSPROT entry of FASTA Format by Accession Number.	
				129	multi fasta format from fasta list	
				4001	multiplealignment by clustalw with blosum	

Why Failed?



Lack of Interoperability Between the Web Services



Very Similar but not the Same Format

Blastp for DAD

[CLUSTALW SETUP (Graphical View<= 100 sequences) | Text View(any number of sequences))]

BLASTP 2.2.12 [Aug-07-2005]

Reference: Altschul, Stephen F., et al. "Gapped BLAST and PSI-BLAST: a new generation of protein database search programs", *Nucleic Acids Res.* 25:3389-3402.

Query= gi|307072|gb|AAA59179.1 (107 letters)

Database: DAD: DAD sequence taken from the May/15/2006
3,084,498 sequences; 935,706,57

Searching.....

Sequences producing significant alignments

L15440-1	AAA59179.1	107 Homo sapiens insulin protein.	<u>177</u>	1e-43
BC005255-1	AAH05255.1	110 Homo sapiens INS		
X70508-1	CAA49913.1	110 Homo sapiens pre		
V00565-1	CAA23828.1	110 Homo sapiens pre		
M10039-1	AAA59173.1	110 Homo sapiens ins		
J00265-1	AAA59172.1	110 Homo sapiens ins		
X61092-1	CAA43405.1	110 Cercopithecus ae		
X61089-1	CAA43403.1	110 Pan troglodytes		
J00336-1	AAA36849.1	110 Macaca fasciula		
AY137503-1	AAN06937.1	110 Pongo pygmaeus		
AY137500-1	AAN06935.1	110 Gorilla gorilla		
AY137497-1	AAN06933.1	110 Pan troglodyte		
A48810-1	CAA03148.1	86 unidentified pro		
A11939-1	CAA00997.1	90 synthetic const		

Blastp for UniProt

[CLUSTALW SETUP (Graphical View<= 100 sequences) | Text View(any number of sequences))]

Reference: Altschul, Stephen F., et al. "Gapped BLAST and PSI-BLAST: a new generation of protein database search programs", *Nucleic Acids Res.* 25:3389-3402.

Query= gi|307072|gb|AAA59179.1 (L15440) insulin [Homo sapiens] (107 letters)

Database: /db/SP/216,3

Searching.....

Sequences producing significant alignments:

	Score (bits)	E Value
sp Q8HXV2 INS_PONPY Insulin precursor [Contains: Insulin B chain...	<u>171</u>	4e-43
sp P30410 INS_PANTR Insulin precursor [Contains: Insulin B chain...	<u>171</u>	4e-43
sp P30406 INS_MACFA Insulin precursor [Contains: Insulin B chain...	<u>171</u>	4e-43
sp P01308 INS_HUMAN Insulin precursor [Contains: Insulin B chain...	<u>171</u>	4e-43
sp Q6YK33 INS_GORGO Insulin precursor [Contains: Insulin B chain...	<u>171</u>	4e-43
sp P30407 INS_CERAE Insulin precursor [Contains: Insulin B chain...	<u>171</u>	4e-43
sp P01311 INS_RABIT Insulin precursor [Contains: Insulin B chain...	<u>158</u>	4e-39
sp Q91X13 INS_SPETR Insulin precursor [Contains: Insulin B chain...	<u>156</u>	1e-38
sp P01321 INS_CANFA Insulin precursor [Contains: Insulin B chain...	<u>154</u>	8e-38
sp P01310 INS_HORSE Insulin precursor [Contains: Insulin B chain...	<u>147</u>	6e-36
sp P01323 INS2_RAT Insulin-2 precursor [Contains: Insulin-2 B ch...	<u>147</u>	6e-36
sp P01326 INS2_MOUSE Insulin-2 precursor [Contains: Insulin-2 B ...	<u>147</u>	6e-36
sp P01313 INS_CRIL0 Insulin precursor [Contains: Insulin B chain...	<u>147</u>	1e-35
sp P01315 INS_PIG Insulin precursor [Contains: Insulin B chain; ...	<u>144</u>	5e-35

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

Akihiko Konagaya: "Bioinformatics Ontology: Towards the Automatics Generation of Bioinformatics Workflow for Web Services," in Proc. of Distributed Applications, Web Services, Tools and GRID Infrastructures for Bioinformatics (NETTAB2006), S. Margherita di Pula, Italy (<http://www.nettab.org/2006/>), pp.75-82 (2006)

Akihiko Konagaya: "OBIGrid: Towards the 'Ba' for Sharing Resources, Services and Knowledge for Bioinformatics", in Proc. of Fourth International Workshop on Biomedical Computations on the Grid (BioGrid), Singapore ([CCGRID 2006](#)), 37 (2006)

Thank You for Listening