



Tavaxy

Integrating Taverna and Galaxy with Cloud Computing Support

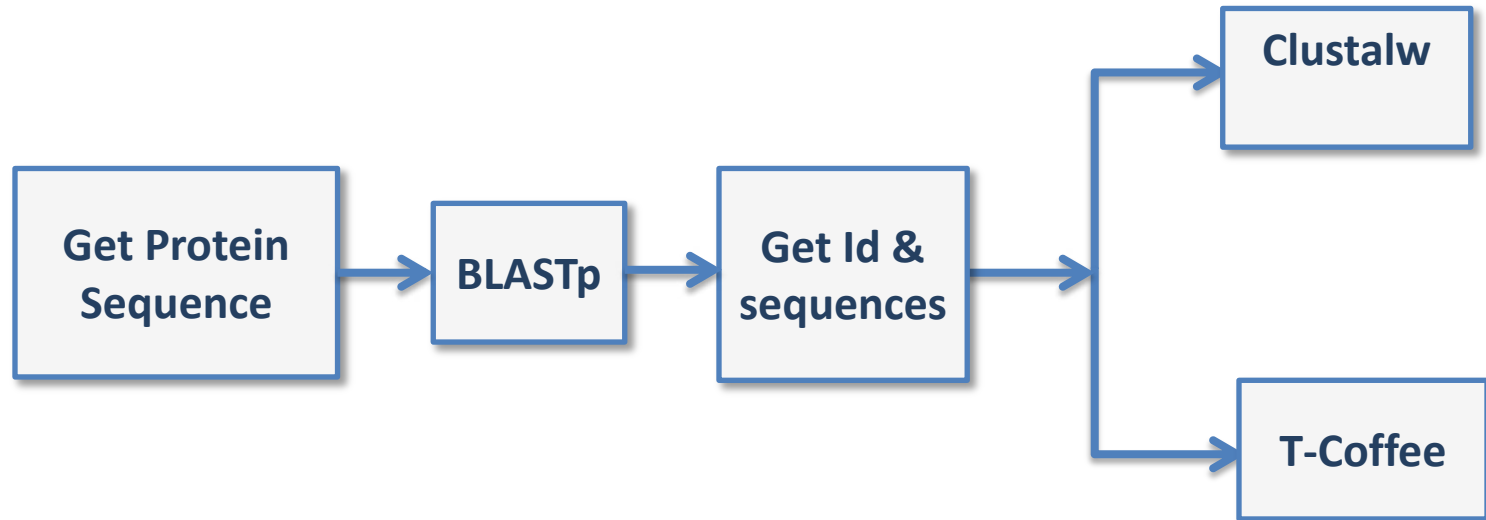
Mohamed Abouelhoda

**Nile University
Egypt**



Workflows in Bioinformatics

Finding Homologous Sequences



Implementing Scientific Workflows

Method 1: Write Python/Perl/Shell script

Advantages

- Reliable and efficient
- Comprehensive programming capabilities (conditionals, loops, etc..)

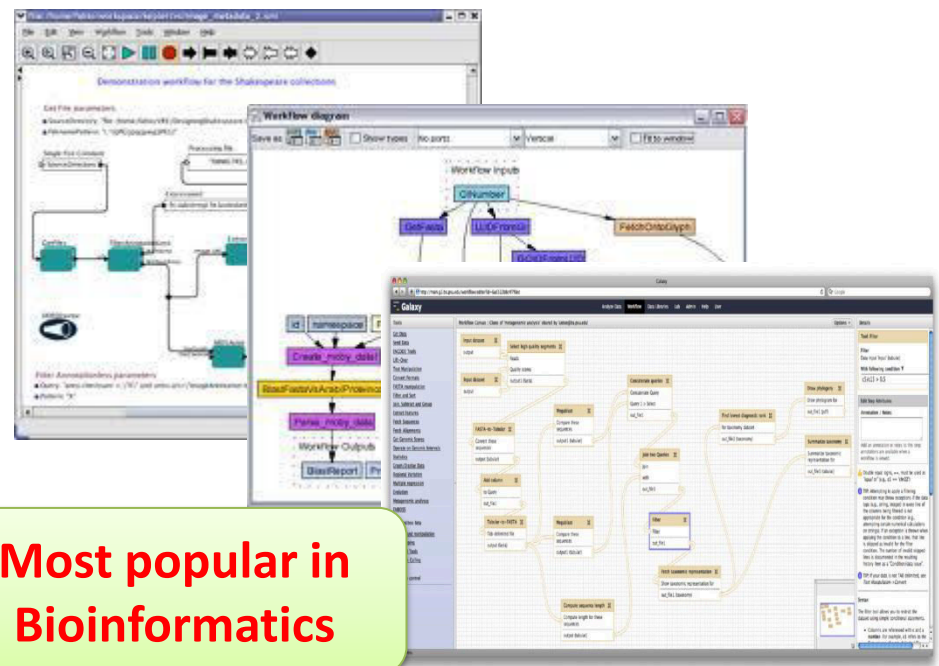
Disadvantages

- Requires programming skills , especially with HPC resources
- Scripts are workflow-specific
- Costly to create, debug and modify
- Requires installing and managing tools

Implementing Scientific Workflows

Methods 2: Use of Workflow Systems

- Kepler
- Triana
- WildFire
- GenePattern
- Pegsus
- Taverna
- Galaxy
- and many more



http://en.wikipedia.org/wiki/Bioinformatics_workflow_management_systems

Composing a Workflow using Galaxy

The screenshot displays the Galaxy web interface. On the left is a 'Tools' sidebar with a scrollable list of categories and tools. The main area features a 'Hello world! It's running...' message at the top, followed by a workflow canvas titled 'WWFSMD?' with the subtitle 'grow noodly appendages...'. The canvas shows a diagram of a workflow with various tool nodes connected by lines. On the right side, there is a vertical list of workflow steps, each with a status icon, a name, and a description. Three green callout boxes with red text are overlaid on the image: one pointing to the top right corner, one pointing to the tool list, and one pointing to the workflow canvas.

Workflow Environment

Tool Library

Canvas for workflow editing

Tools

- Get Data
- Local Tools
- Send Data
- ENCODE Tools
- Lift-Over
- Text Manipulation
- Filter and Sort
- Join, Subtract and Group
- Convert Formats
- Extract Features
- Fetch Sequences
- Fetch Alignments
- Get Genomic Scores
- Operate on Genomic Intervals
- Statistics
- Wavelet Analysis
- Graph/Display Data
- Regional Variation
- Multiple regression
- Multivariate Analysis
- Evolution
- Metagenomic analysis
- FASTA manipulation
- NGS: QC and mapping
- NCR BLAST+
- NGS: Mapping
- NGS: Indel Annot
- NGS: RNA Annot
- NGS: SAM Tools
- NGS: Peak Calling

WWFSMD?
grow noodly appendages...

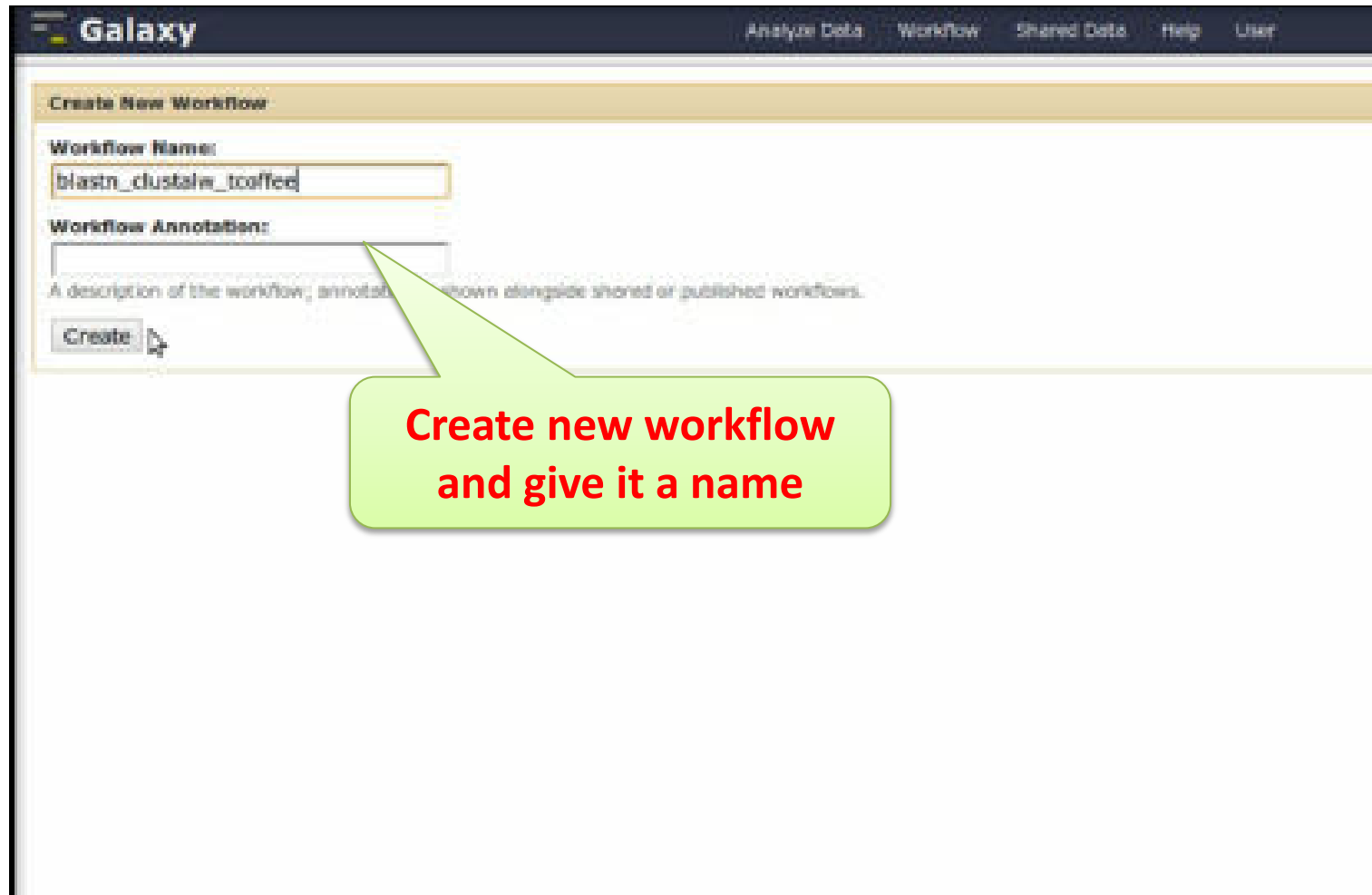
usegalaxy.org

This project is supported in part by NSF, NSERC, and the Huck Institutes of the University of Toronto.

Workflow Steps:

- 6: Local ClustalW2 on data 12 and data 12
- 5: 6
- 5: Local ClustalW2 on data 12 and data 12
- 5: 5
- 4: Local ClustalW2 on data 12 and data 12
- 4: 4
- 3: pvalue on data 11 and data 11
- 3: 3
- 2: pvalue on data 9 and data 9
- 2: 2
- 1: pvalue on data 7 and data 7
- 1: 1

Composing a Workflow using Galaxy



The screenshot shows the Galaxy web interface. At the top is a dark navigation bar with the 'Galaxy' logo and links for 'Analyze Data', 'Workflow', 'Shared Data', 'Help', and 'User'. Below this is a light-colored header for the 'Create New Workflow' section. The form contains two main input fields: 'Workflow Name:' with the text 'blastn_clustalw_tcoffed' and 'Workflow Annotations:' which is currently empty. Below the annotations field is a small text hint: 'A description of the workflow; annotations shown alongside shared or published workflows.' At the bottom left of the form is a 'Create' button with a mouse cursor hovering over it. A green callout box with a pointer to the 'Create' button contains the red text: 'Create new workflow and give it a name'.

Galaxy

Analyze Data Workflow Shared Data Help User

Create New Workflow

Workflow Name:
blastn_clustalw_tcoffed

Workflow Annotations:

A description of the workflow; annotations shown alongside shared or published workflows.

Create

Create new workflow and give it a name

Composing a Workflow using Galaxy

The screenshot displays the Galaxy web interface. At the top, there is a navigation bar with tabs for 'Analyze Data', 'Workflow', 'Shared Data', 'Help', and 'User'. The main area is divided into three panels. On the left is the 'Tools' panel, which contains a list of tool categories such as 'Graph/Display Data', 'Regional Variation', 'Multivariate Analysis', 'Evolution', 'Metagenomic analyses', 'FASTA manipulation', 'NGS: QC and manipulation', 'NCBI BLAST+', 'NGS: Mapping', 'NGS: Indel Analysis', 'NGS: RNA Analysis', 'NGS: SAM Tools', 'NGS: Peak Calling', 'NGS: Simulation', 'SNP/WGA: Data Filters', 'SNP/WGA: QC, LO, Plots', 'SNP/WGA: Statistical Models', 'Human Genome Variation', 'EMBOSS', 'unfolding', 'runners', 'utility tools', 'test tools', 'my tools', 'patterns', and 'sandbox'. Below these categories is a 'Workflow control' section with an 'Inputs' subsection. A green callout bubble points to the 'Inputs' subsection, containing the text 'Click to create input node'. The central panel is the 'Workflow Canvas', which is a large grid area for building workflows. The right panel is the 'Details' panel, which shows 'Edit Workflow Attributes' for a workflow named 'blastn_clustalw_tcoffee'. It includes fields for 'Name', 'Tags', and 'Annotation / Notes'.

Click to create
input node

Composing a Workflow using Galaxy

The screenshot displays the Galaxy web interface for composing a workflow. The main area is the 'Workflow Canvas' titled 'blastn_clustalw_tcoffee'. On the left is a 'Tools' sidebar with a search bar and a list of tool categories including 'Get Data', 'Local Tools', 'Send Data', 'ENCODE Tools', 'Lift-Over', 'Text Manipulation', 'Filter and Sort', 'Join, Subtract and Group', 'Convert Formats', 'Extract Features', 'Fetch Sequences', 'Fetch Alignments', 'Get Genomic Scores', 'Operate on Genomic Intervals', 'Statistics', 'Wavelet Analysis', 'Graph/Discrete Data', 'Regional Variation', 'Multiple regression', 'Multivariate Analysis', 'Evolution', 'Metagenomic analysis', 'FASTA manipulation', 'NGS: QC and manipulation', 'NCBI BLAST+', 'NGS: Mapping', 'NGS: Indel Analysis', 'NGS: RNA Analysis', 'NGS: SAM Tools', and 'NGS: Peak Calling'. In the center of the canvas, an 'Input Dataset' node is placed, with its output labeled 'output'. A green callout bubble points to this node with the text: **Input node created, will read input from user account**. On the right, the 'Details' panel shows the 'Input dataset' section with a 'Name:' field containing 'InputSequence', and an 'Edit Step Attributes' section with an 'Annotation / Notes:' field and a description: 'Add an annotation or notes to this step; annotations are available when a workflow is viewed.'

Composing a Workflow using Galaxy

The screenshot shows the Galaxy web interface. On the left, a 'Tools' panel lists various bioinformatics tools. The 'BLAST' tool is highlighted. In the center, a 'Workflow Canvas' shows a grid with an 'Input Dataset' node and an 'output' label. On the right, a 'Details' panel shows the 'Input dataset' name as 'InputSequence' and a section for 'Edit Step Attributes' with an 'Annotation / Notes' field. A red speech bubble points to the 'BLAST' tool in the left panel, and a green speech bubble points to the 'output' label in the canvas.

Choose BLAST tool and click to create it

BLAST node is custom one added to the system

Composing a Workflow using Galaxy

The screenshot displays the Galaxy web interface. On the left, a 'Tools' sidebar lists various bioinformatics tools. The main 'Workflow Canvas' shows a workflow named 'blastn_clustalw_tcoffee'. An 'Input Dataset' node is connected to a 'Blast' node. A 'Details' panel on the right shows the configuration for the 'Blast' tool, including program, database, type, input sequence, dummy, tool ID, and iteration strategy. Two green callout boxes with red text provide instructions: '1. BLAST node created' and '2. Window opened to set BLAST parameters'.

Tools

- NGS: RNA Analysis
- NGS: SAM Tools
- NGS: Peak Calling
- NGS: Simulation
- SNP/WGA: Data: Filters
- SNP/WGA: QC: LD: Plots
- SNP/WGA: Statistical Models
- Human Genome Variation
- EMBOSS
- unfolding
- runners
- utility tools
- test tools
- my tools
- confind Find conserved regions
- confind_inputs Find conserved regions
- Local Clustalw execute ClustalW program
- GCGtoFASTA convert gde files in to fasta files
- Compute GC content for each sequence in a file
- Blast execute NCBI BLAST web service
- Blast_Fast execute NCBI BLAST web service
- ePrimer3 pick primers
- ePrimer3.Parser execute primer3 parser to get primers in multifasta format
- Get Sequences cat fasta

Workflow Canvas | blastn_clustalw_tcoffee

Input Dataset 20
output

Blast 20
InputSequence
dummy
Blastnits (tabular)

Details

Tool: Blast

program: ▼
blastn

database: ▼
em_rel

type: ▼
dna

InputSequence
Data input 'InputSequence' (fasta)

dummy
Data input 'dummy' (data)

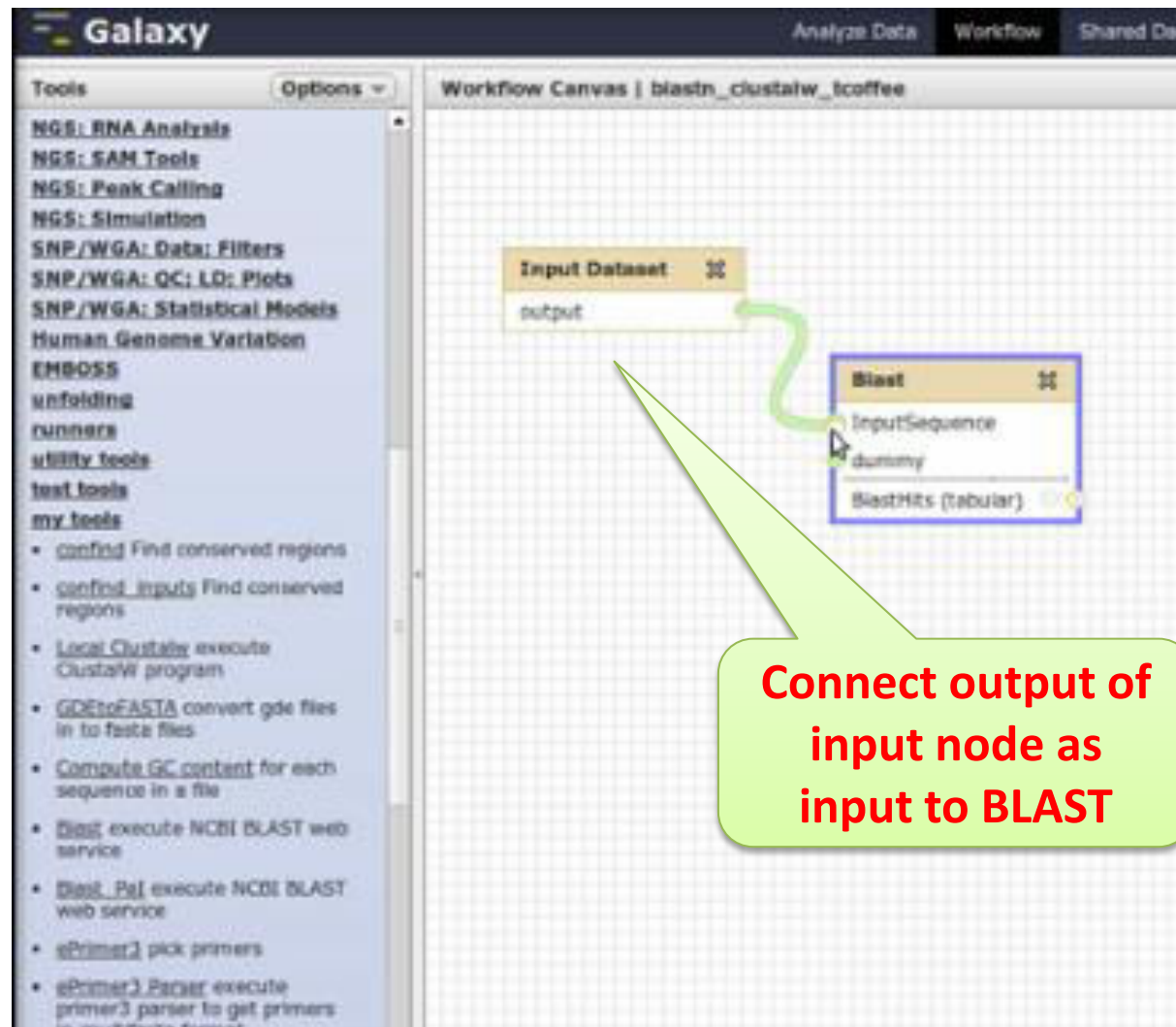
toolID: ▼
[empty]

iteration strategy: ▼
cross ▼

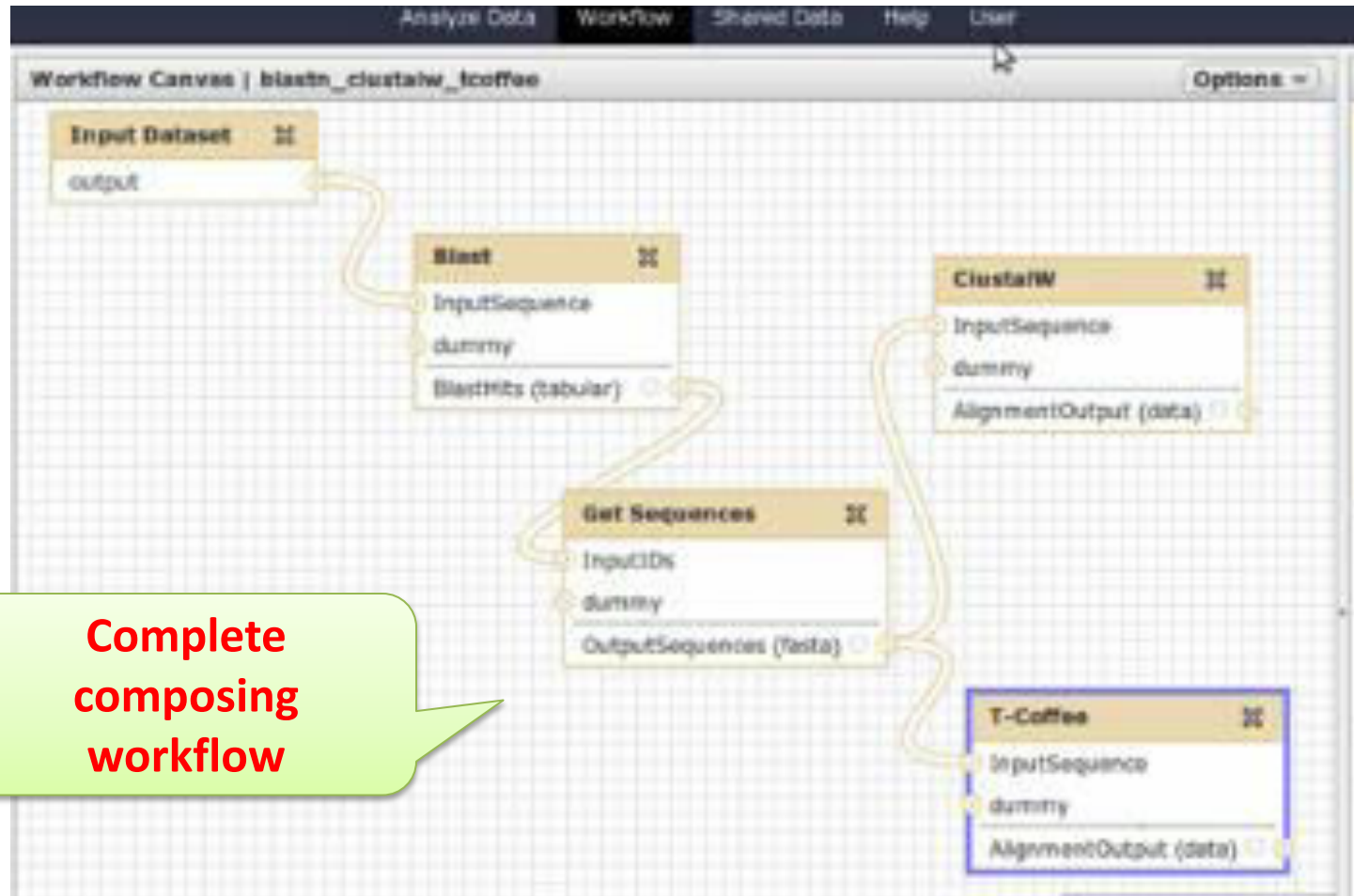
1. BLAST node created

2. Window opened to set BLAST parameters

Composing a Workflow using Galaxy



Composing a Workflow using Galaxy



**Complete
composing
workflow**

Composing a Workflow using Galaxy



The screenshot shows the Galaxy web interface. At the top is a navigation bar with the 'Galaxy' logo and links for 'Analyze Data', 'Workflow', 'Shared Data', 'Help', and 'User'. Below this is a section titled 'Your workflows' with a green 'Go' button. A table lists workflows with columns for 'Name' and '# of Steps'. A context menu is open over the first row, showing options: 'Edit', 'Run' (highlighted with a mouse cursor), 'Share or Publish', 'Download or Export', 'Clone', 'Rename', and 'Delete'. A green speech bubble with red text points to the 'Run' option.

Name	# of Steps
blast	5
work	8
work	5
work	5
work	5
work	5
work	5
workflow =	5
workflow =	5
workflow =	4
workflow =	4
workflow =	4
workflow =	4
workflow =	4
workflow =	3
workflow =	8
workflow =	10
workflow =	5
workflow =	5
workflow =	5
workflow =	3
workflow =	3
workflow =	3

Start running the workflow

Composing a Workflow using Galaxy

The screenshot displays the Galaxy web interface for a workflow titled "blastn_clustalw_tcoffee". The interface is divided into three main steps:

- Step 1: Input dataset**
 - InputSequence**: A dropdown menu showing "773: 773".
- Step 2: Blast**
 - program**: blastn
 - database**: em_nsl
 - type**: dna
 - InputSequence**: Output dataset 'output' from step 1
 - dummy**: A dropdown menu showing "773: 773".
 - toolID**: None
 - iteration strategy**: cross
- Step 3: Get Sequences**
 - InputIDs**: Output dataset 'BlastHits' from step 2
 - dummy**: A dropdown menu showing "773: 773".

A **Log messages** window is open in the top right corner, displaying a list of log entries with timestamps and details about the workflow steps.

Screen showing parts of the output files and provides links to them

Benefits of Workflow Systems

Accelerates computation

- Intuitive abstract means for describing computational experiments
- Requires no programming expertise
- Easy to modify
- Hide execution details: invocation and scheduling
- Direct use of parallel architectures

Benefits of Workflow Systems

Formalize computation

- Come with library of tools or access directories
→ Tool accessibility
- Store experiment details (used tools, their parameters, and used data)
→ Reproducibility
- Share workflows with analysis history including intermediate results
→ Transparency

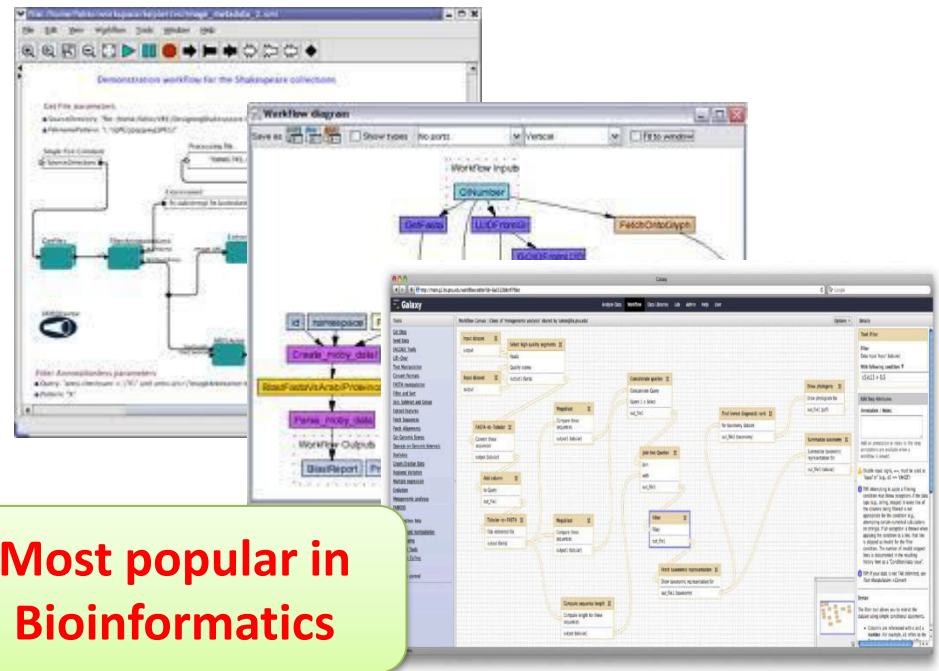
Mesirov. Science, 2010

B. Giardine, et al. Genome Research, 2005.

Implementing Scientific Workflows

Methods 2: Use of Workflow Systems

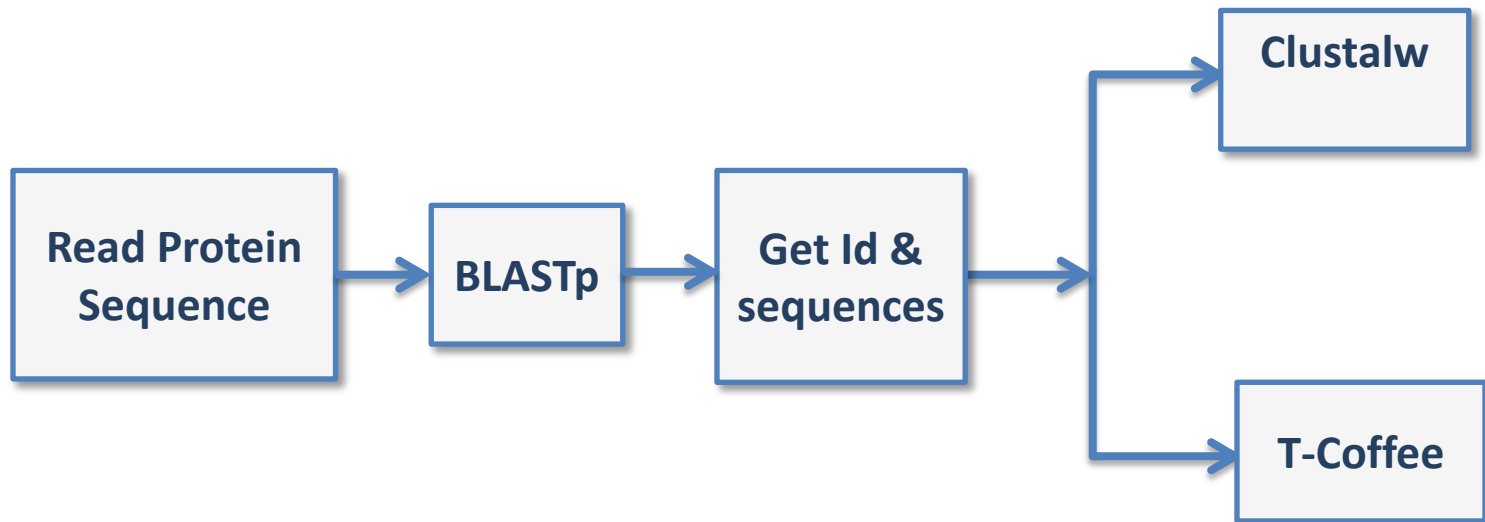
- Kepler
- Triana
- WildFire
- GenePattern
- Pegsus
- Taverna
- Galaxy
- and many more



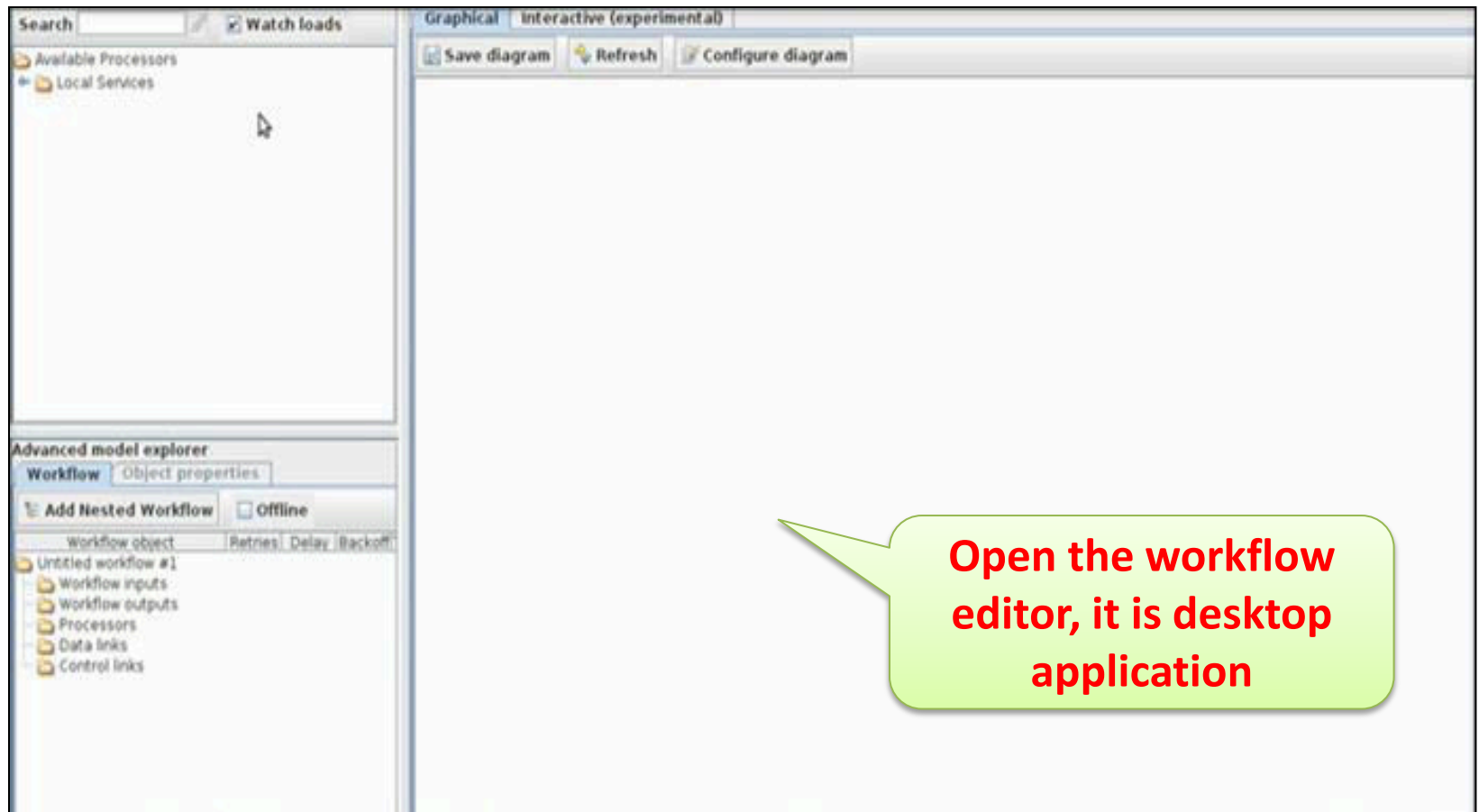
http://en.wikipedia.org/wiki/Bioinformatics_workflow_management_systems

Composing a Workflow using Taverna

Example: Homology Workflow



Composing a Workflow using Taverna



Composing a Workflow using Taverna

EMBL-EBI

Find Help Feedback

Databases Tools Research Training Industry About Us Help Site Index

about clients help services archive msa pfa phylogeny psa sas fasta_rest fasta_soap fastn_rest fastn_soap ncbi_blast_rest ncbi_blast_soap psiblast_rest psiblast_soap psirest_rest psirest_soap wu_blast_rest wu_blast_soap censor delete dbfetch dbfetch_rest

EBI - EBI Web Services - services - sas - ncbi_blast_soap

NCBI BLAST (SOAP)

Important

We kindly ask all users of EMBL-EBI Web Services to submit tool jobs in batches of up to 25 at a time and to not be completed for these. This enables users as well as the service maintainers to deal more easily with local and remote unscheduled downtime.

Service provision happens on a fair-share basis. Overzealous usage of a particular resource will be dealt with in a

Language	Download	Requirements
C#	NET Executable: NcbiBlastClient.exe; Source: AbstractWsClient.cs, NcbiBlastClient.cs, NcbiBlastClient.cs	A .NET runtime environment, if building from source development tools are also required. See the .NET tutorial for details.
Java	Executable jar: NcbiBlast_Axis1.jar; Source: AbstractWsToolClient.java, NcbiBlastClient.java	Axis 1.4, All dependencies, including Axis 1.4 and Commons-CLI, are available in lib-1.4.zip.
Perl	Executable jar: NcbiBlast_JAXWS.jar; Source: AbstractWsToolClient.java, NcbiBlastClient.java	JAX-WS, Various dependencies including Commons-CLI are available in lib-1.4.zip.
PHP	ncbiblast_lib.php_soap.php, ncbiblast_cli.php_soap.php, ncbiblast_web.php_soap.php	
Python	ncbiblast_soapp.py, ncbiblast_soap4r.py	
Ruby	ncbiblast_soap4r.rb	
Taverna	1.x: NCBIBLAST (SOAP), 2.x: NCBIBLAST (SOAP)	

For further details of these tool-kits and workflow platforms...

WSDL

The WSDL for the NCBI BLAST SOAP service: <http://www.ebi.ac.uk/Tools/services/soap/ncbiblast?wsdl>

Operations

getParameters()

Get a list of the parameter names.

Arguments: none

Returns: a list of strings giving the names of the parameters.

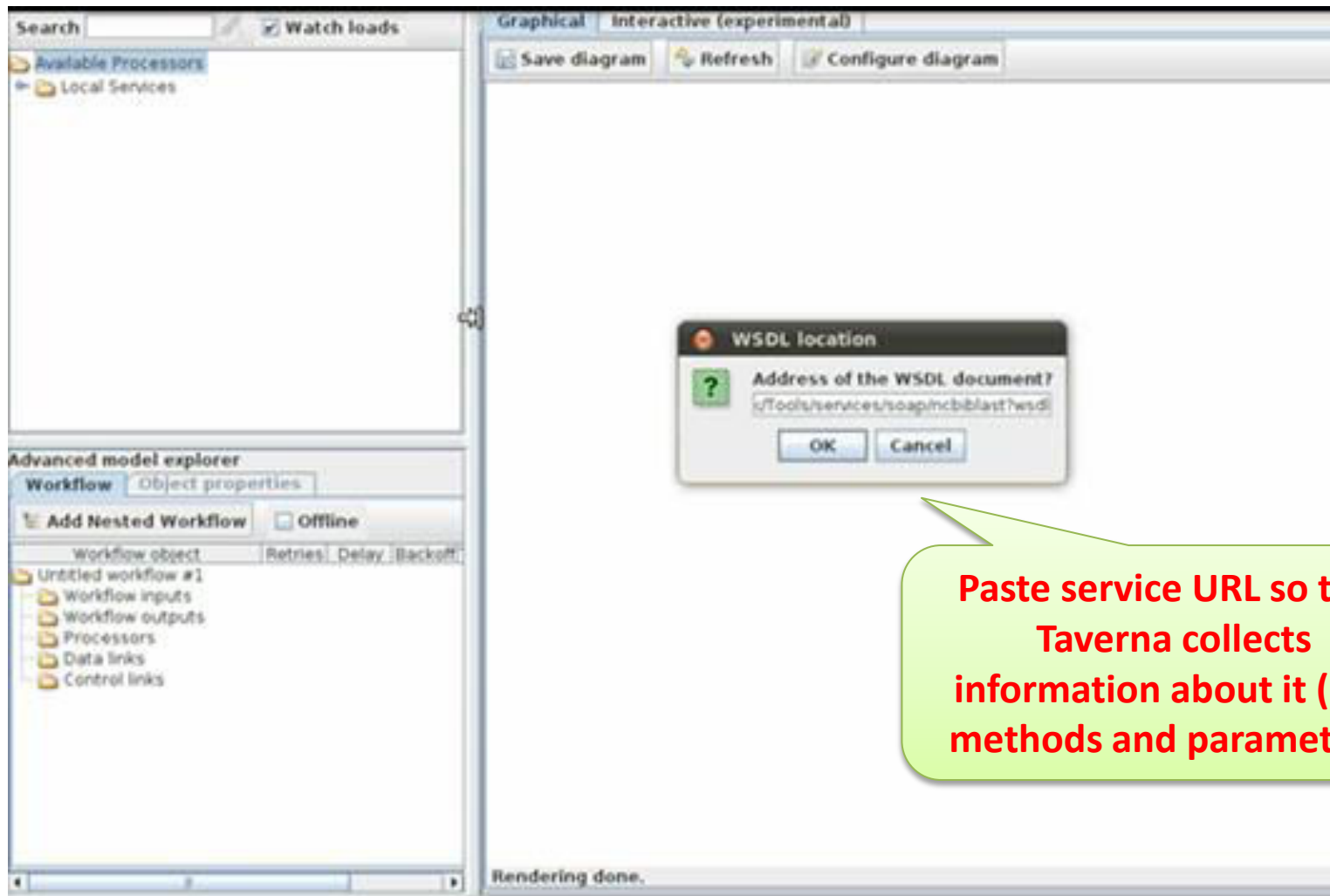
getParameterDetails(parameterId)

Get details of a specific parameter.

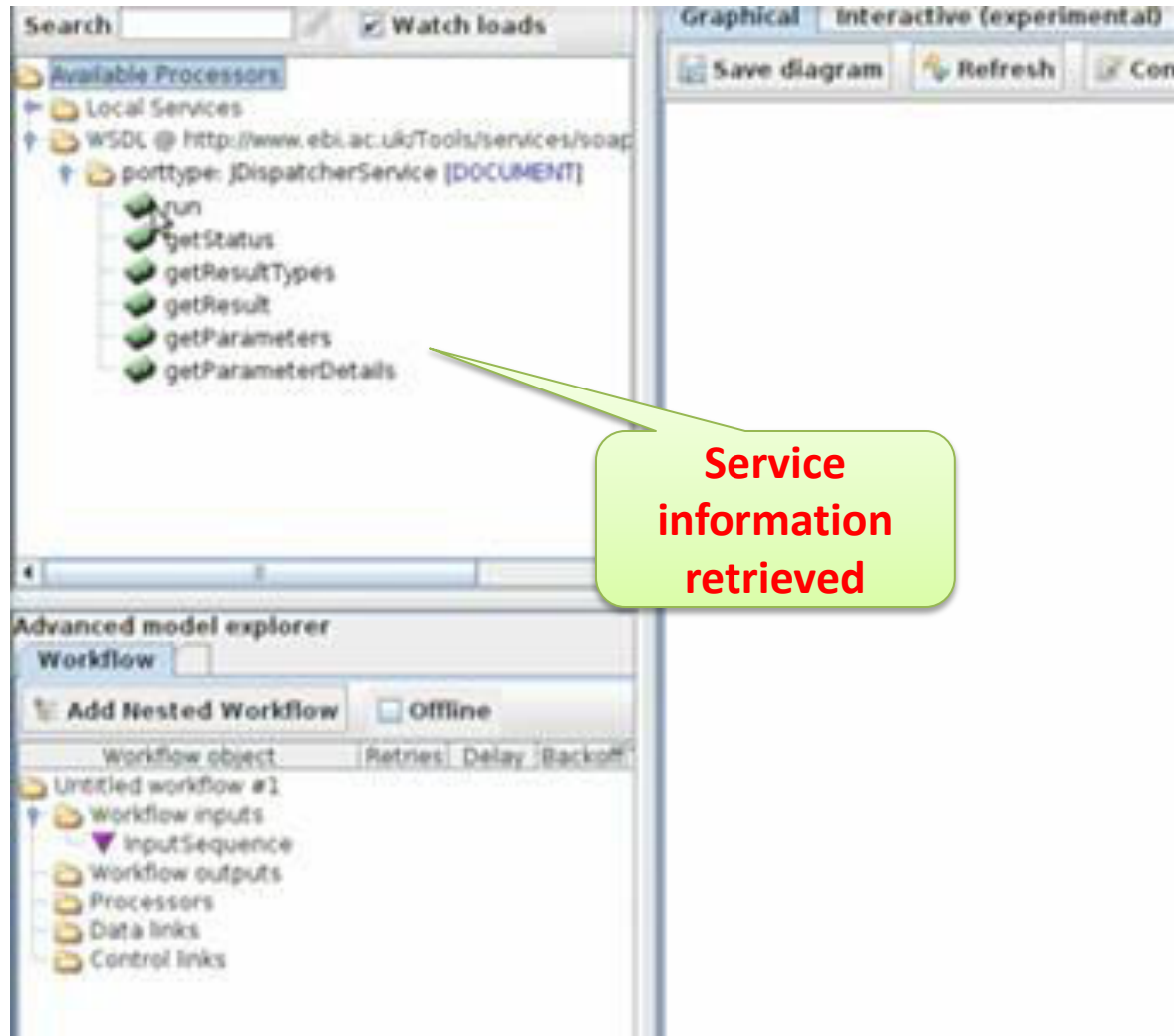
Choose a web-service and copy its URL if not in the service directory,

Here we will use BLAST at EBI

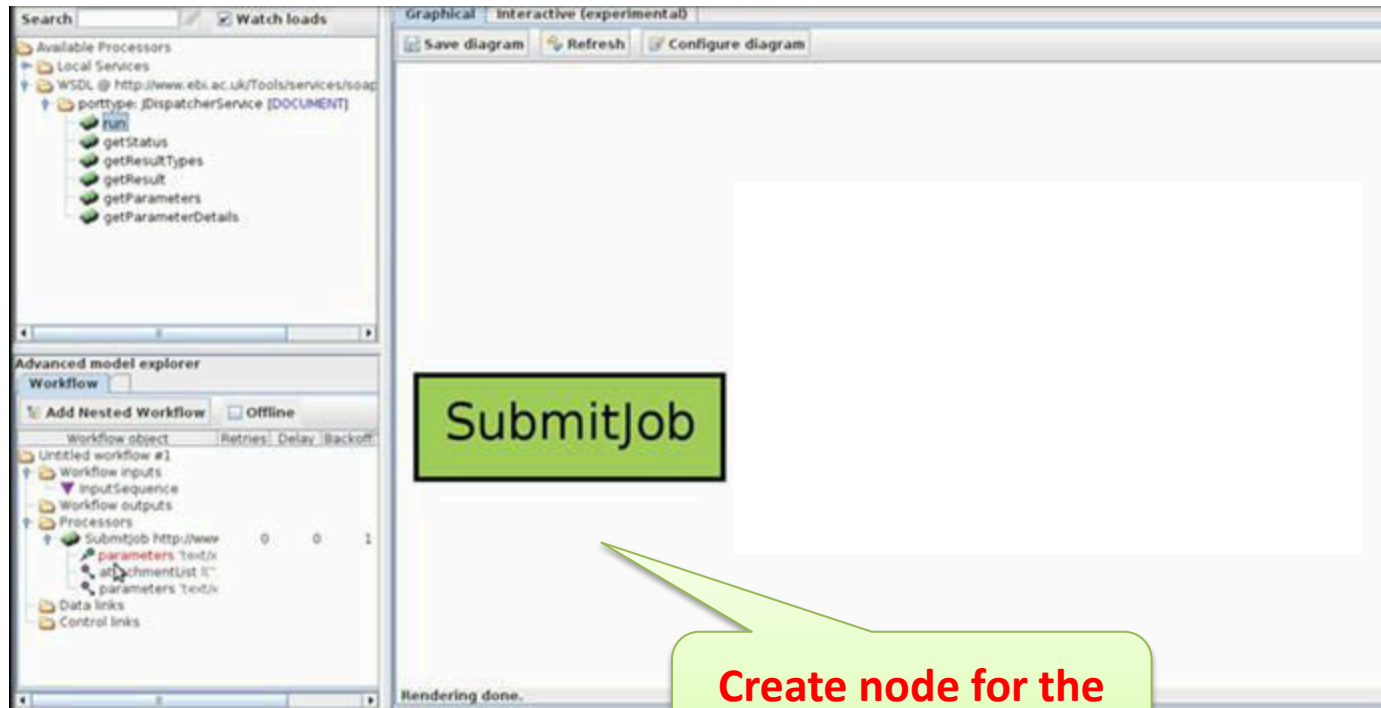
Composing a Workflow using Taverna



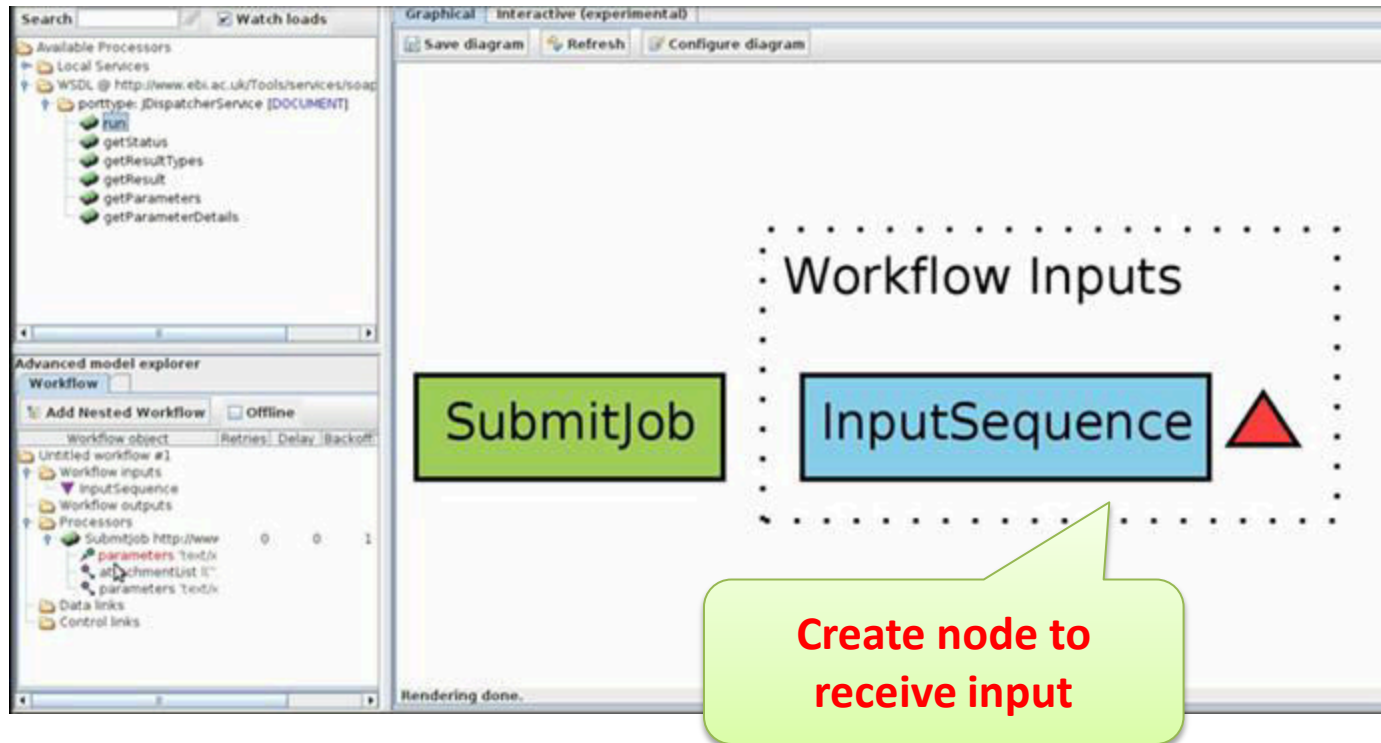
Composing a Workflow using Taverna



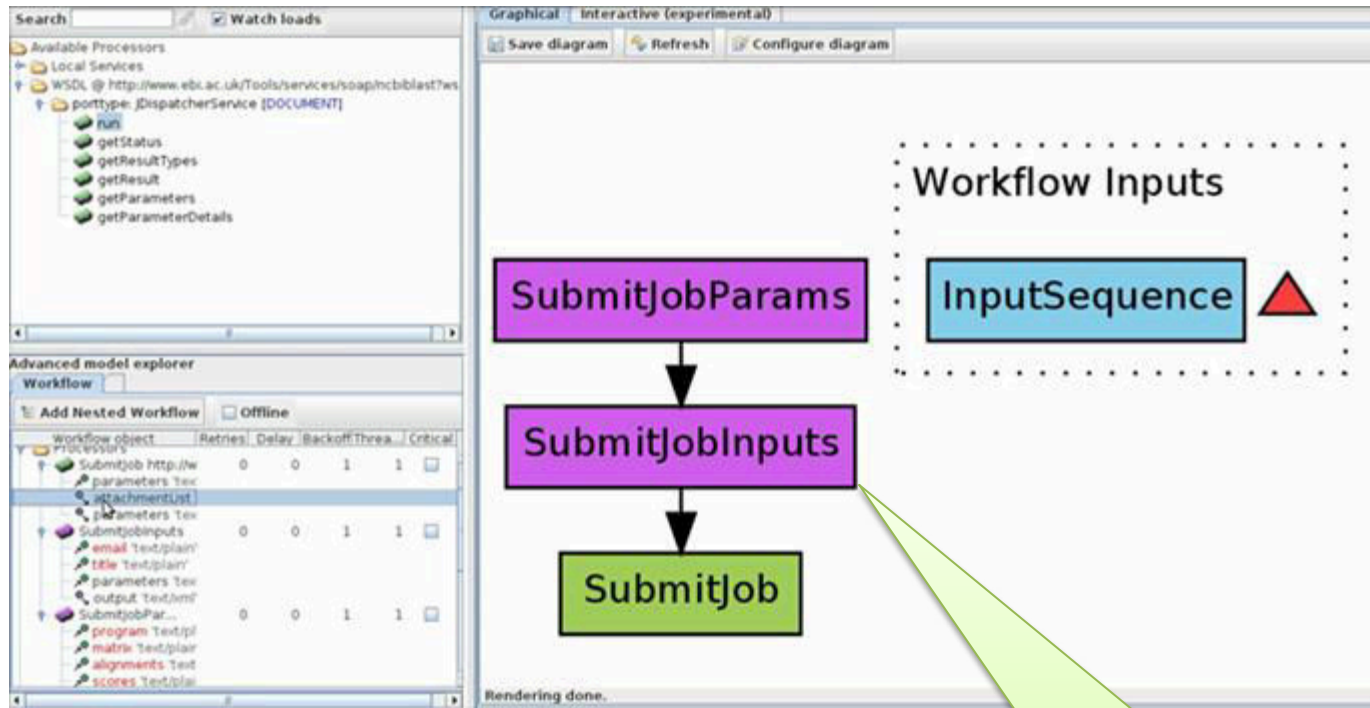
Composing a Workflow using Taverna



Composing a Workflow using Taverna

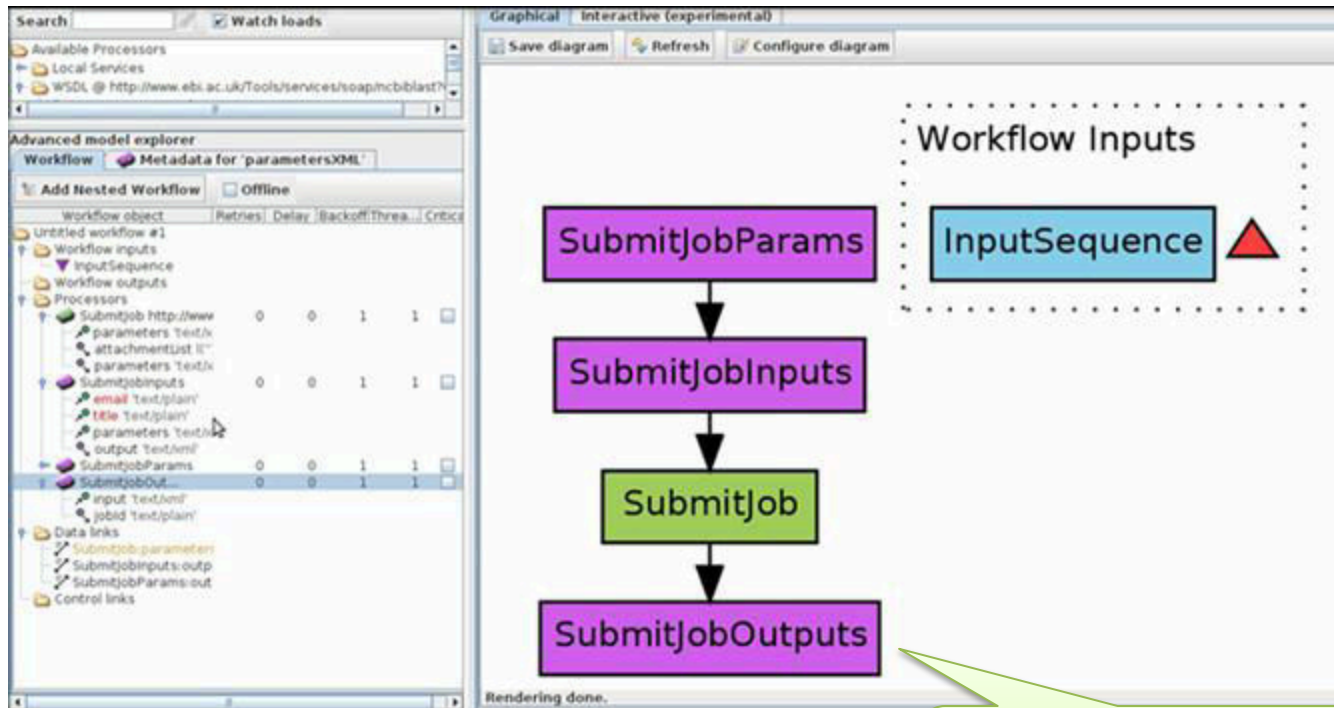


Composing a Workflow using Taverna



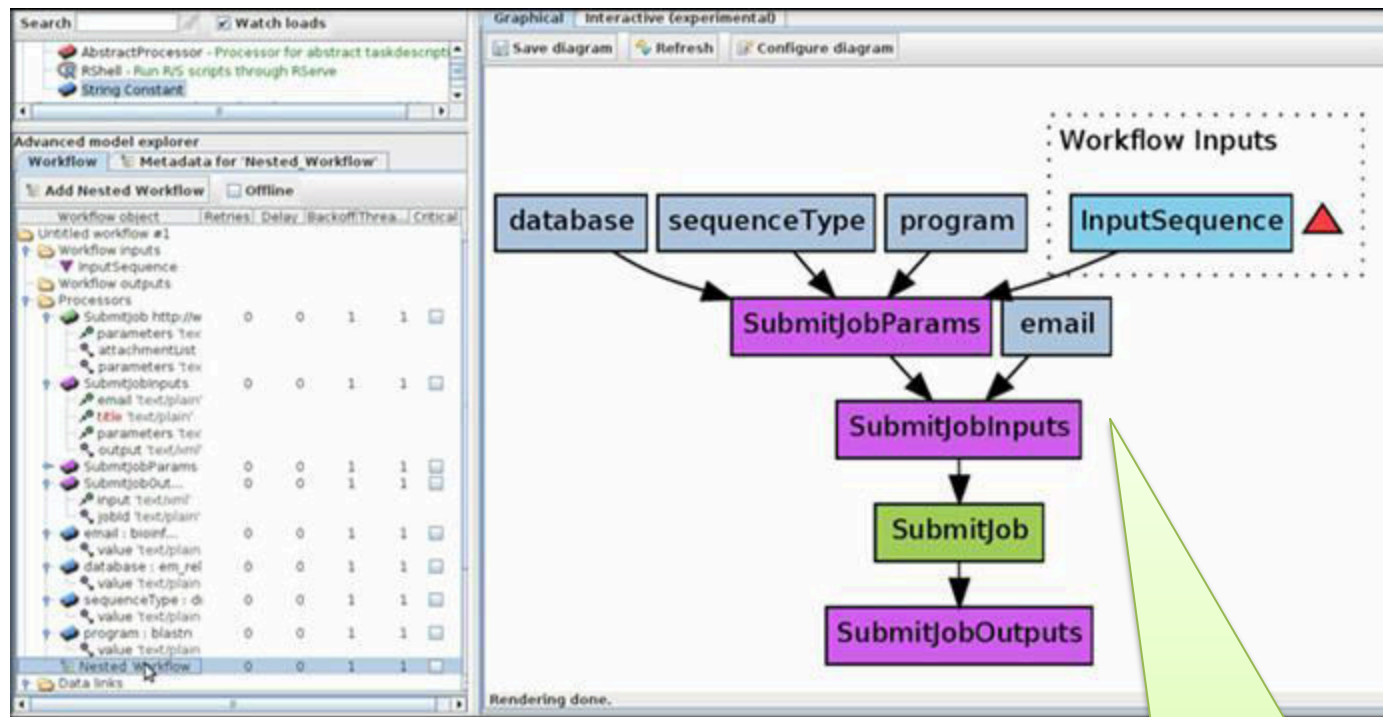
Create nodes to prepare input and parameters in XML

Composing a Workflow using Taverna



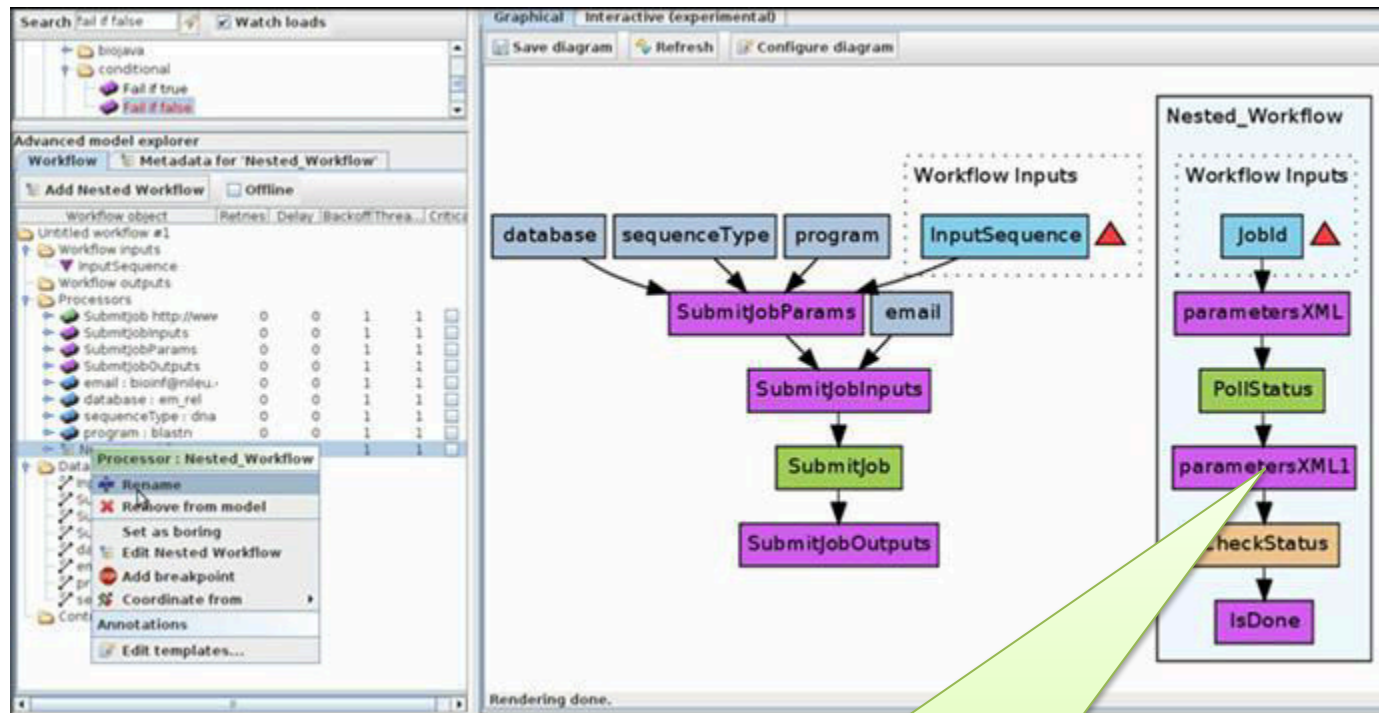
**Create node to
receive Job ID from
service provider**

Composing a Workflow using Taverna



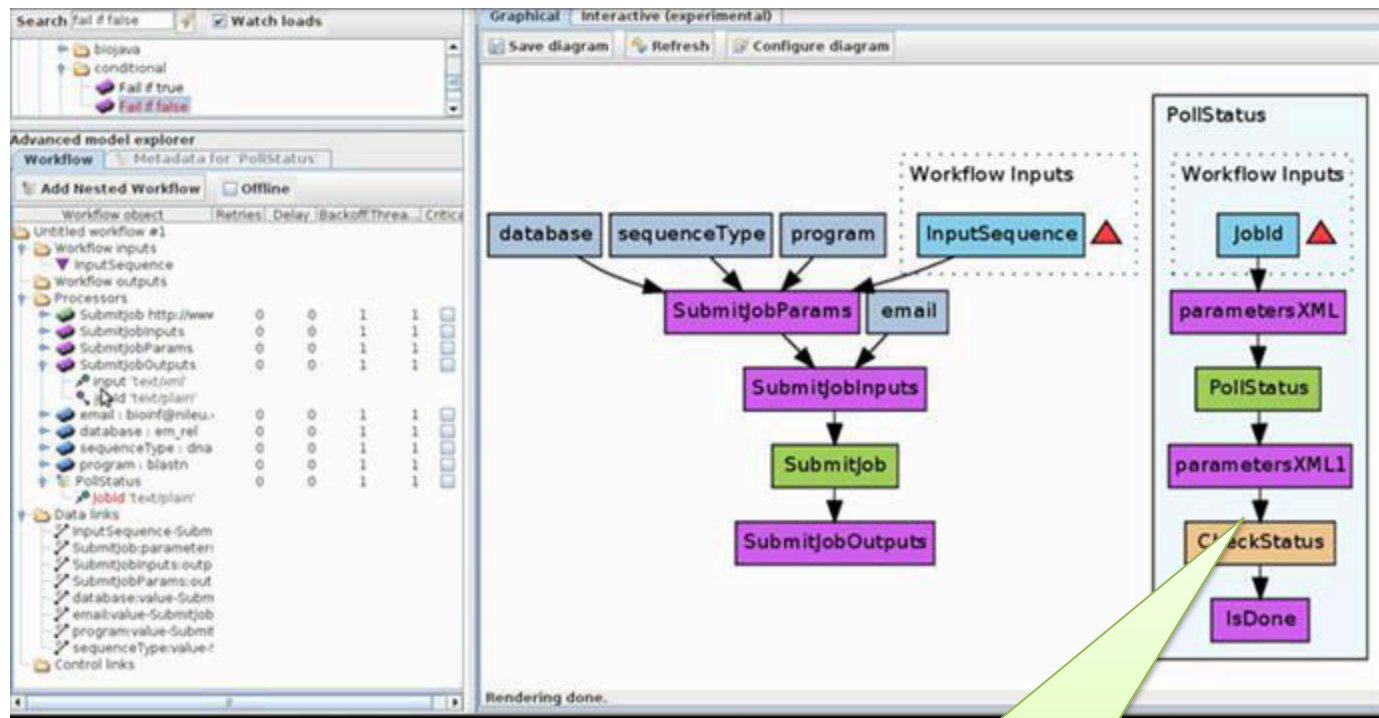
Input data and parameters are merged in XML file

Composing a Workflow using Taverna



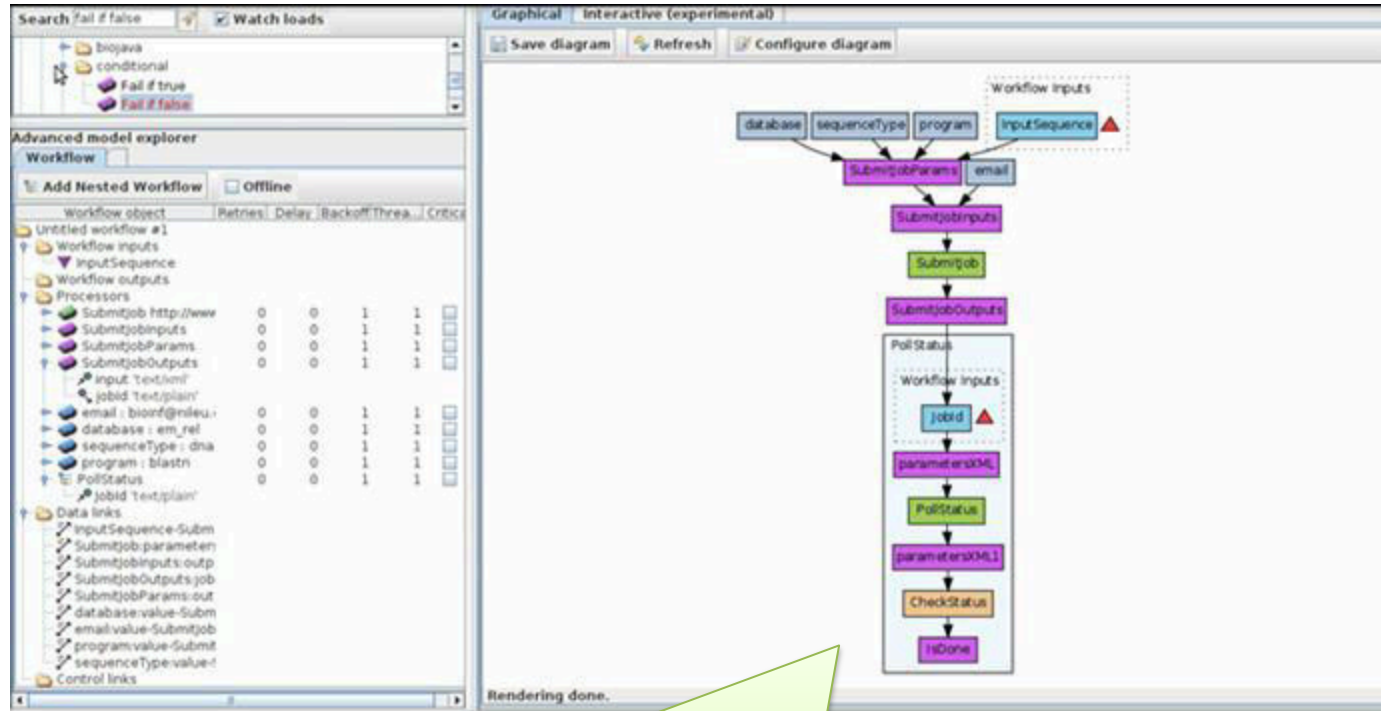
Create sub-workflow to check status of the asynchronous service

Composing a Workflow using Taverna



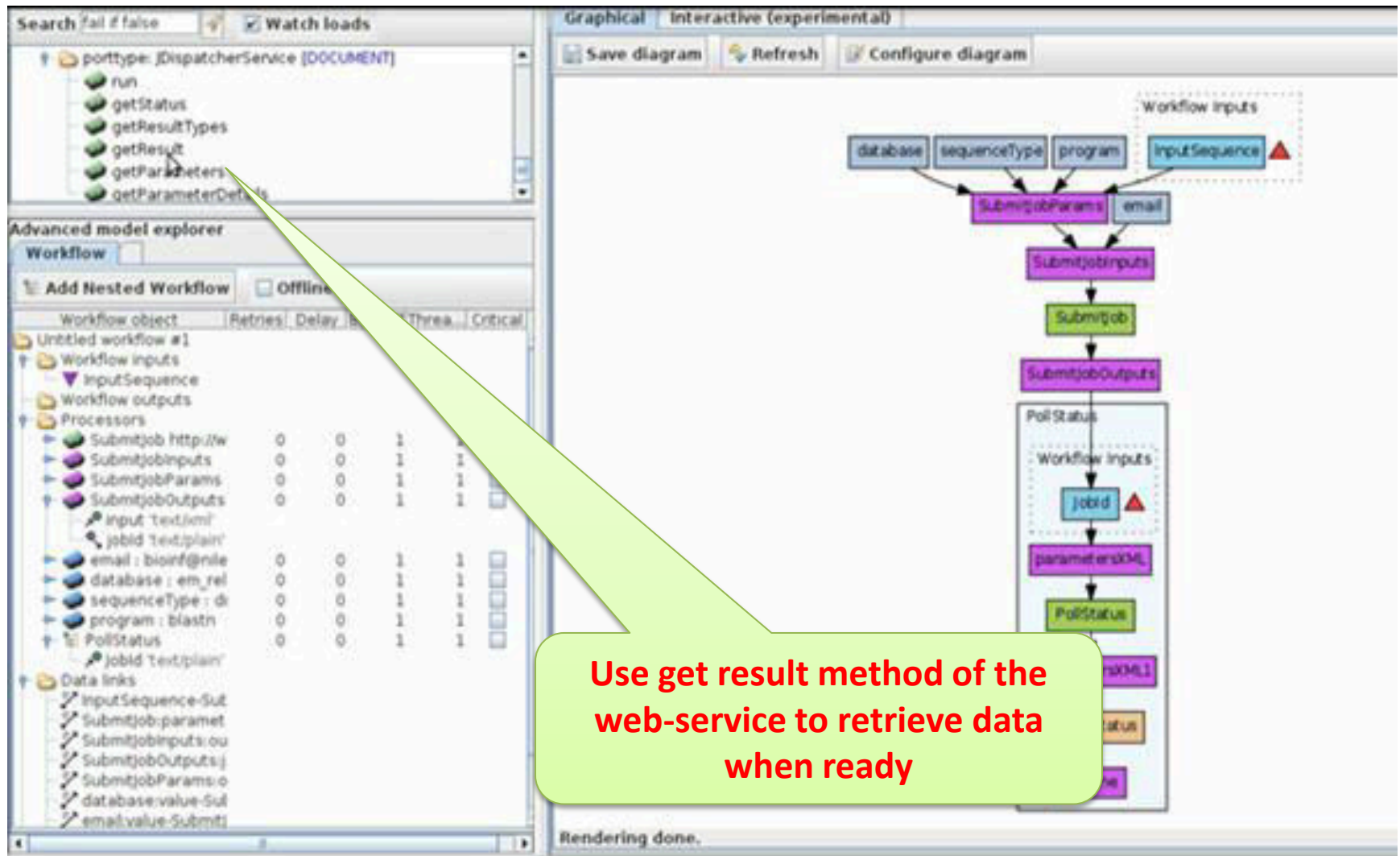
Rename it Poll status

Composing a Workflow using Taverna

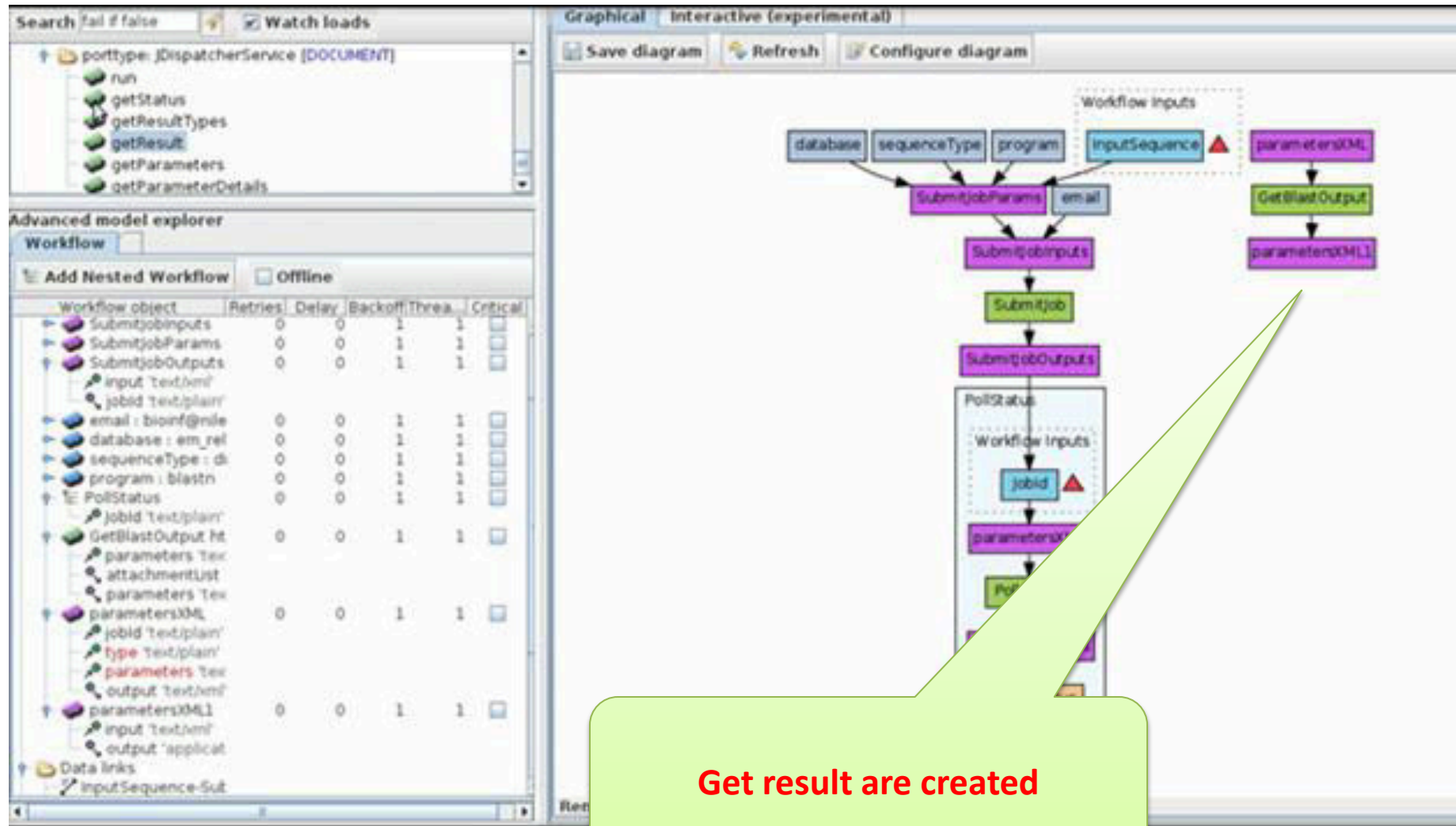


Put in place after job submission

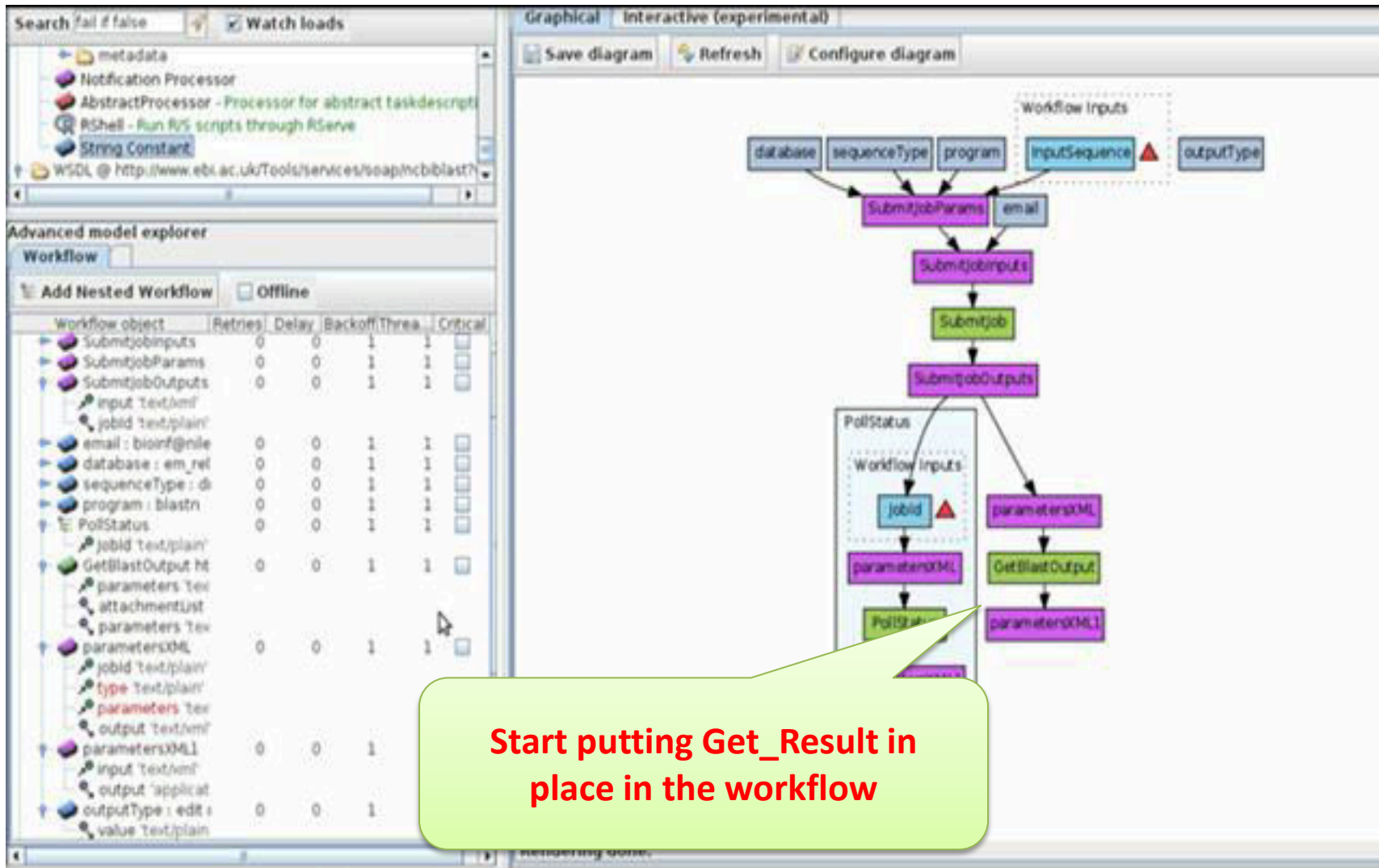
Composing a Workflow using Taverna



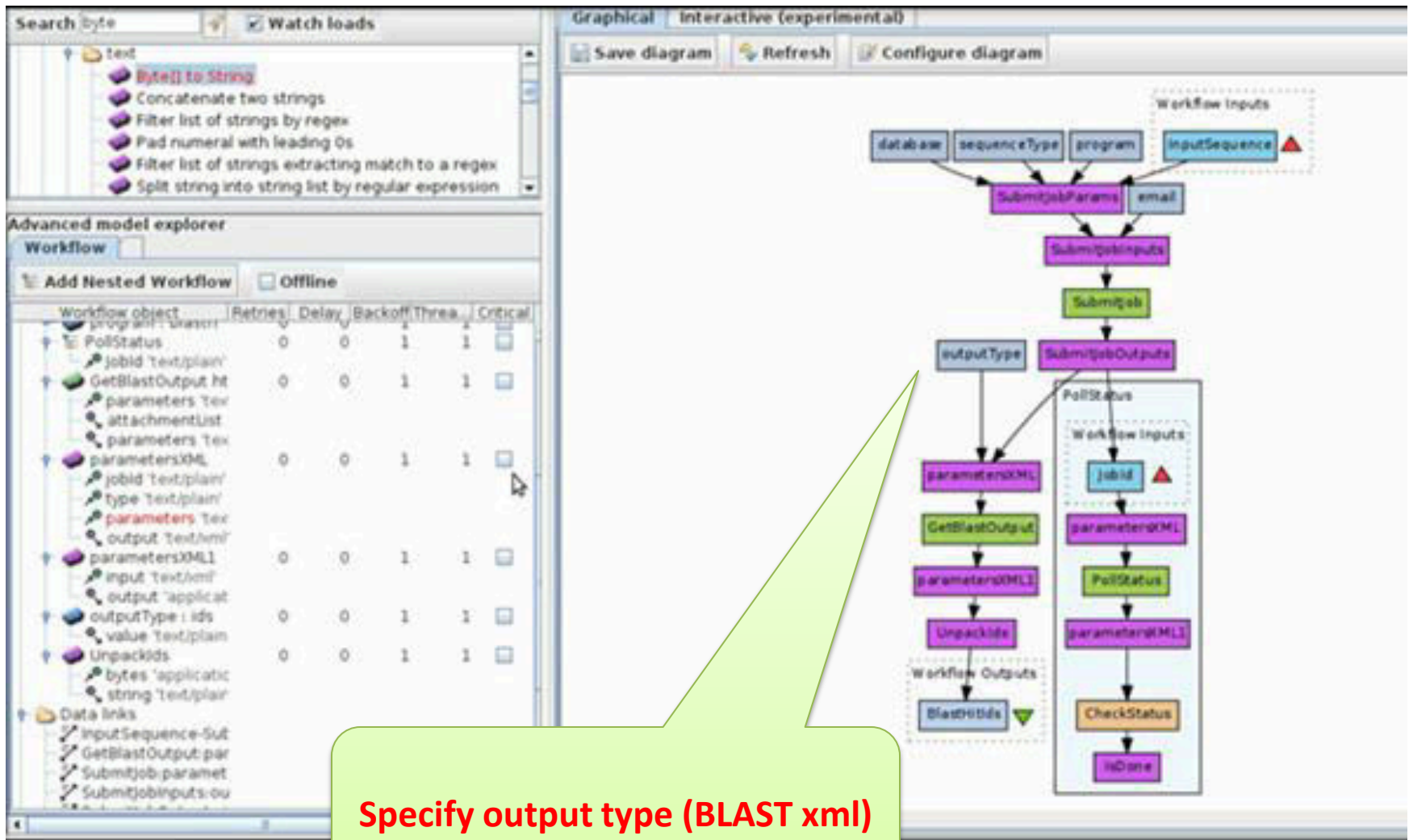
Composing a Workflow using Taverna



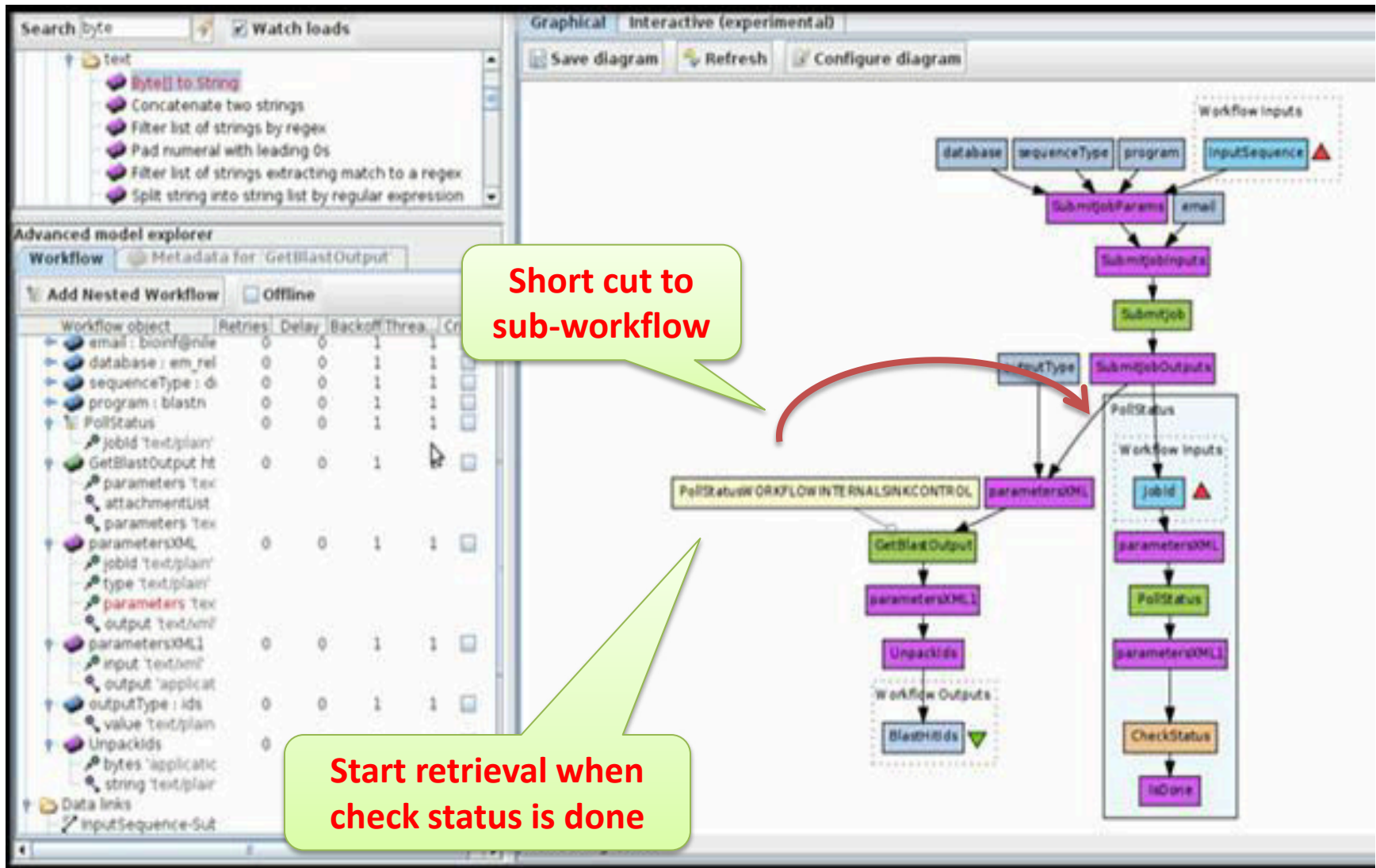
Composing a Workflow using Taverna



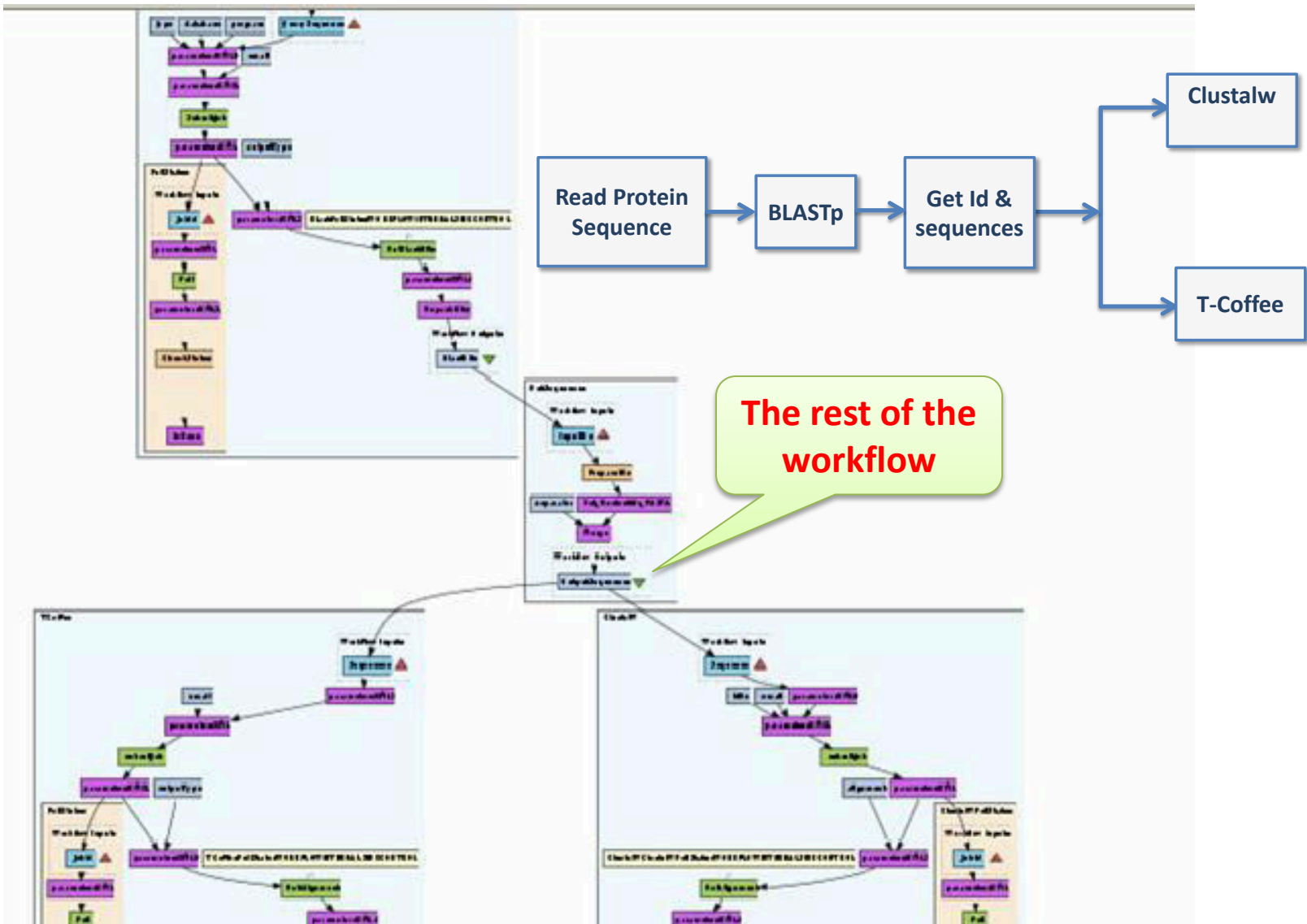
Composing a Workflow using Taverna



Composing a Workflow using Taverna



Composing a Workflow using Taverna



Taverna vs. Galaxy

	Taverna	Galaxy
Purpose	General	Bioinformatics
Interface	Desktop	Web-interface
Usability	More difficult	Easy
Engine	Control flow	Data flow
Control constructs	Yes	No
Jobs	Web-service	Local invocation
Programs	Service directory	Library of tools
Use of Local HPC	Threads only	Threads/Cluster

Taverna

D. Hull, et al. Nucleic Acids Research, 2006.

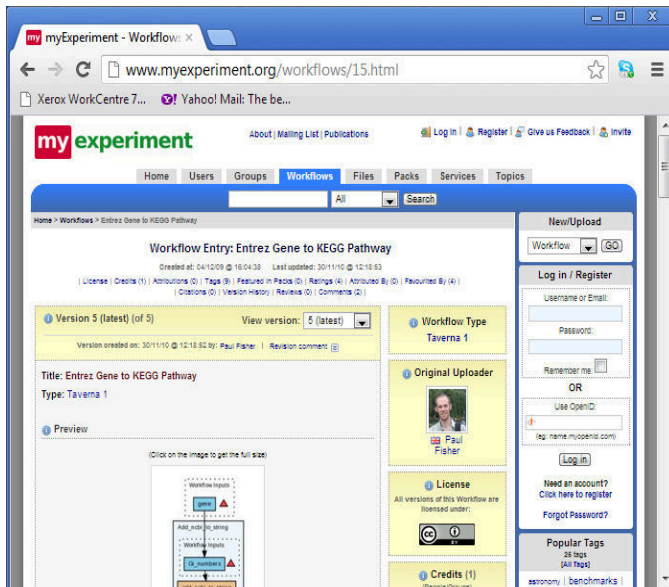
T. Oinn, et al. Bioinformatics, 2004.

Galaxy

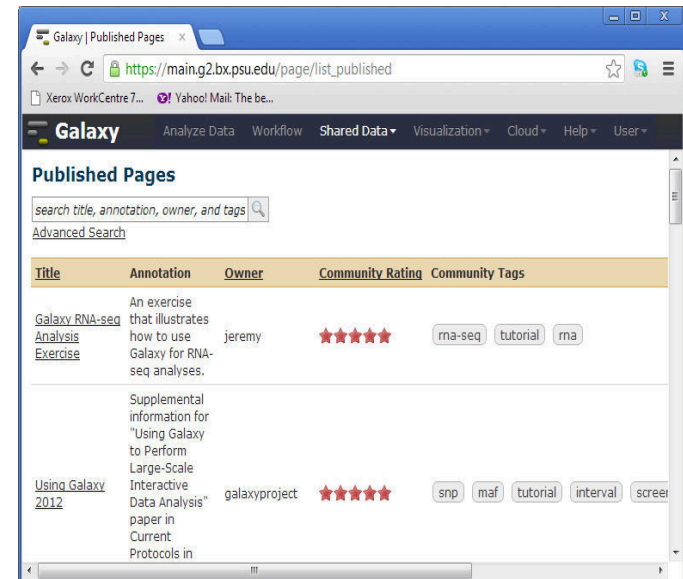
B. Giardine, et al. Genome Research, 2005.

Communities

Taverna MyExperiment

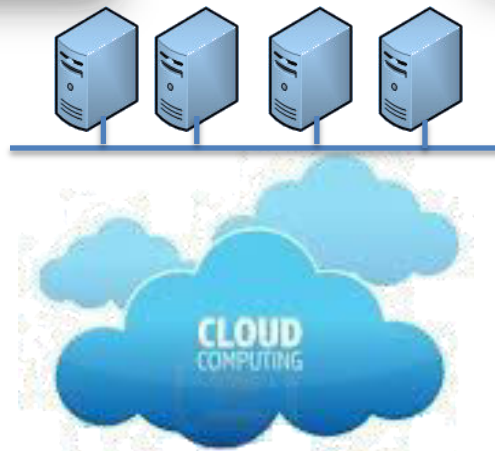
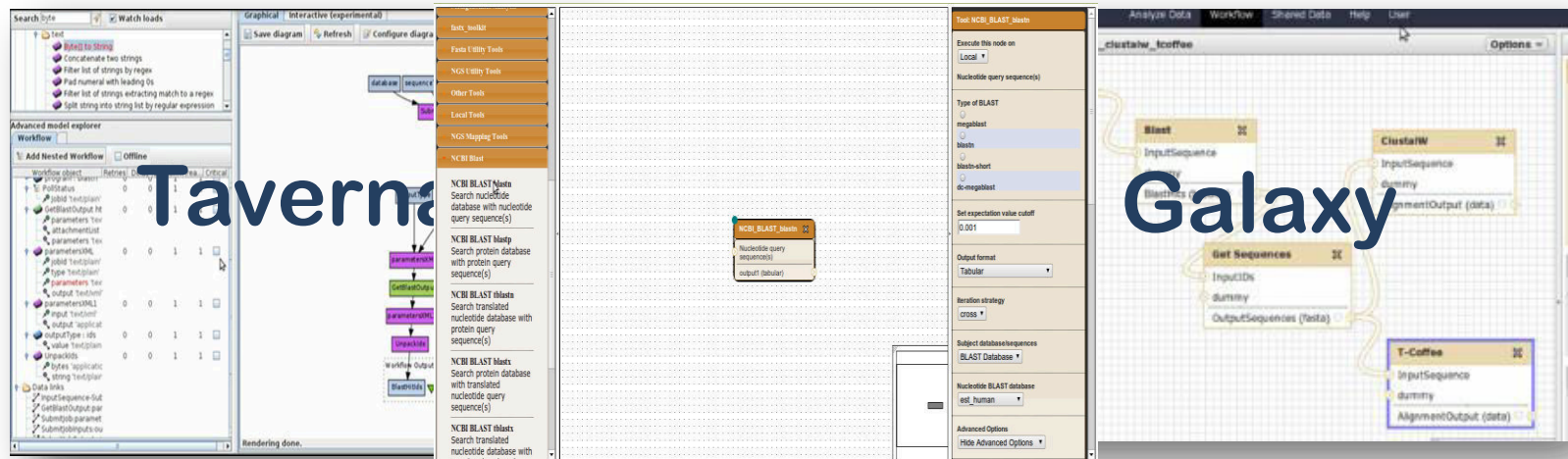


Galaxy Pages



What if we can make use of both repositories and what if we can have advantages of both systems??

Integrating Taverna and Galaxy Workflows



Integrating Taverna and Galaxy Workflows

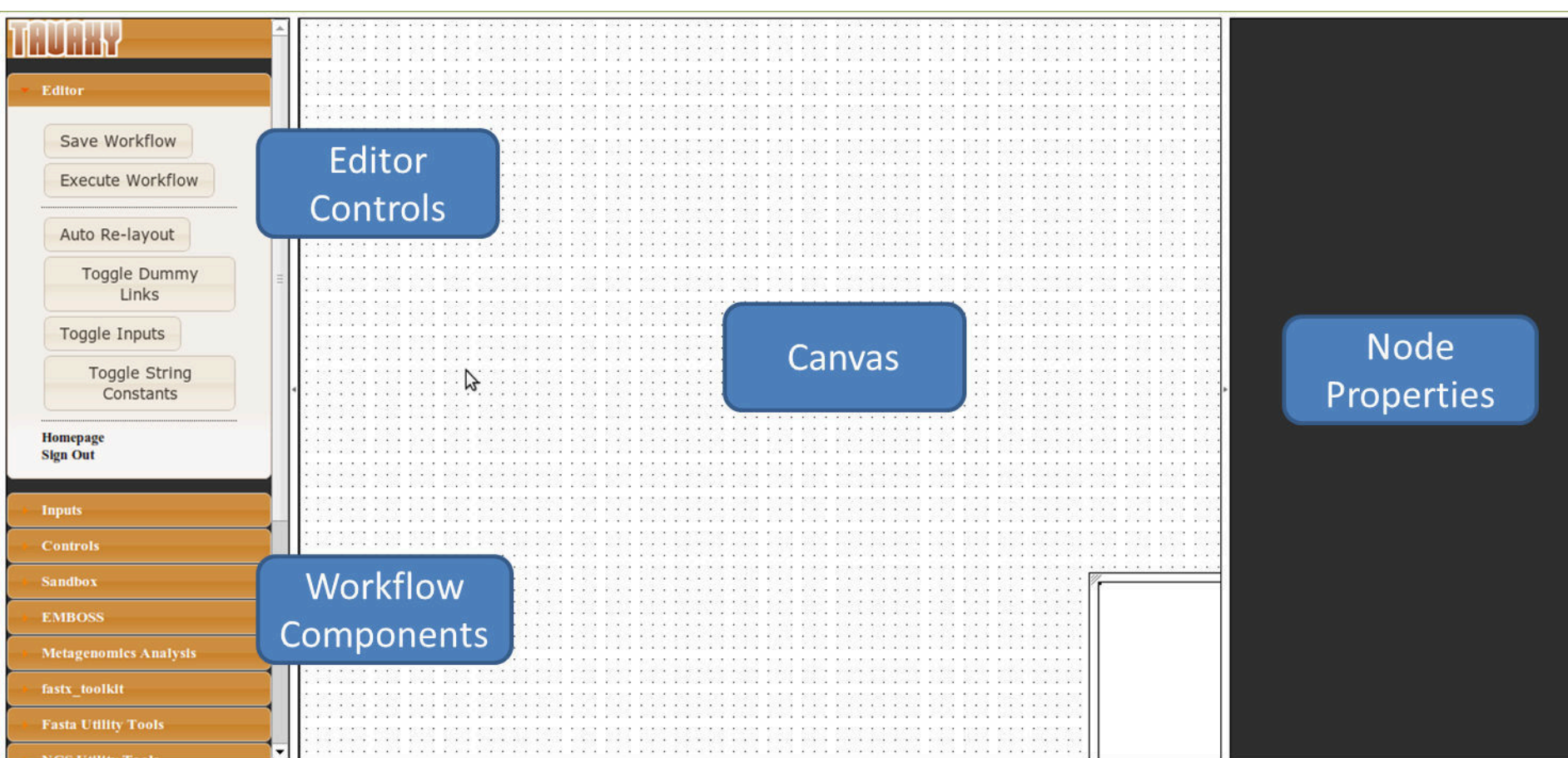
Tavaxy



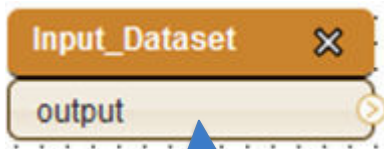
Tavaxy

- A standalone workflow system based on workflow patterns
- Integrates both Taverna and Galaxy workflows and improve their performance
- Easy to use and includes a large library of software tools
- Exploits high performance computing resources (parallel infrastructure) with all details being hidden
- Runs on local or cloud computing infrastructure
- Optimized to handle large datasets

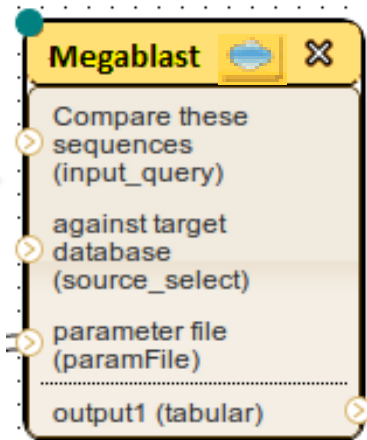
Tavaxy Environment



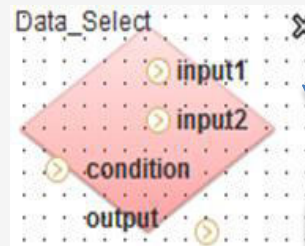
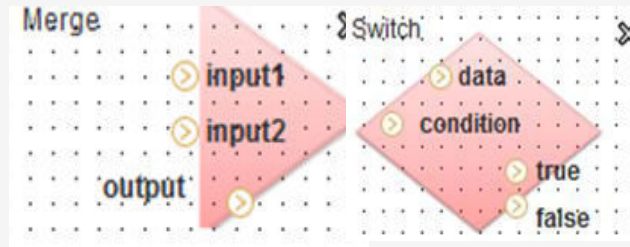
Tavaxy Nodes (Processing Units)



Input node

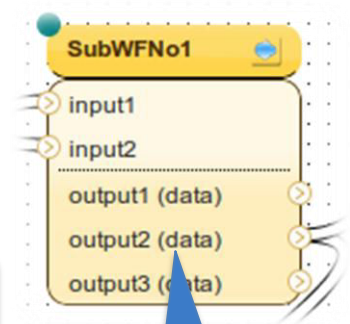


Tool node



Pattern nodes

- If-else conditional
- Iteration
- Data Merge
- Data Select



Sub-workflow nodes

New in Tavaxy and not in Galaxy

- Pattern nodes, Sub-workflow nodes
- Remote execution

New in Tavaxy and not in Taverna

- Easier to use interface
- Direct use of local HPC

Tools in Tavaxy

220 Tools organized in the following categories

- **EMBOSS:** Package of sequence analysis tools
- **SAMtools:** Package of scripts and programs to handle NGS data
- **Fastx:** Package for manipulating fasta files
- **Galaxy tools:** A set of data utilities and tools developed by the galaxy team
- **Taverna tools:** A set of tools based on web-services based on Taverna. collection
This section includes as well a set of data manipulation utilities developed by the taverna team.
- **Tavaxy tools:** This section includes extra utilities and tools for sequence analysis and genome comparison developed/added by the Tavaxy team.
- **Cloud utilities:** A set of data manipulation and configuration of computing infrastructure on the Amazon cloud.

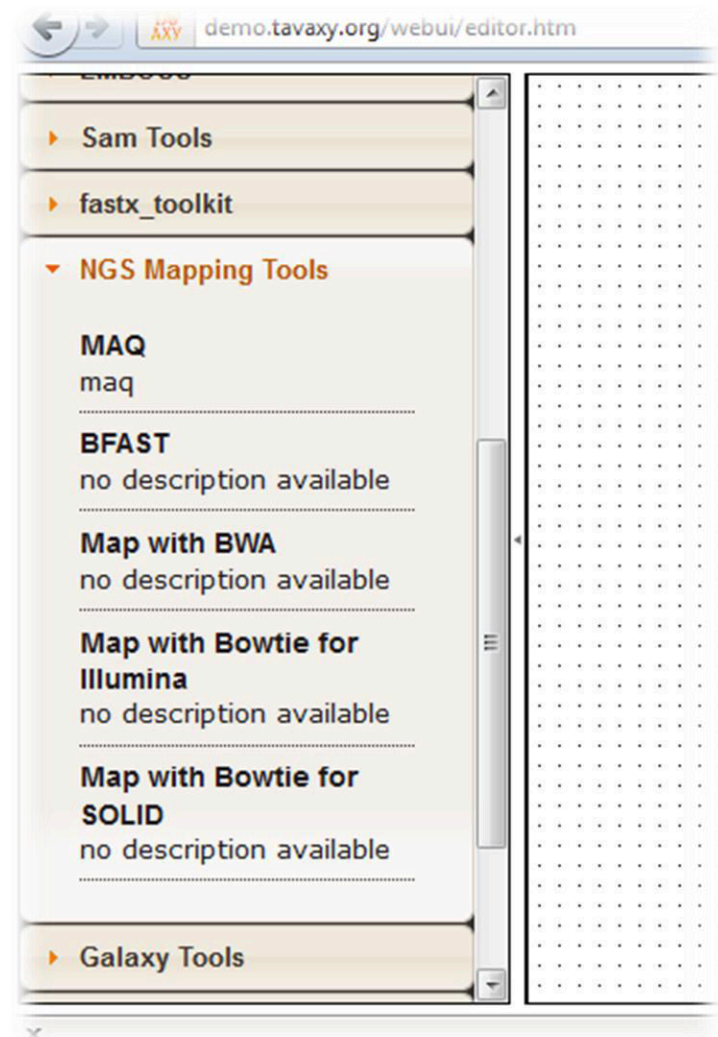
NGS Mapping Tools and Databases in Tavaxy

Tools

- MAQ
- BFAST
- BWA
- Bowtie
- ...

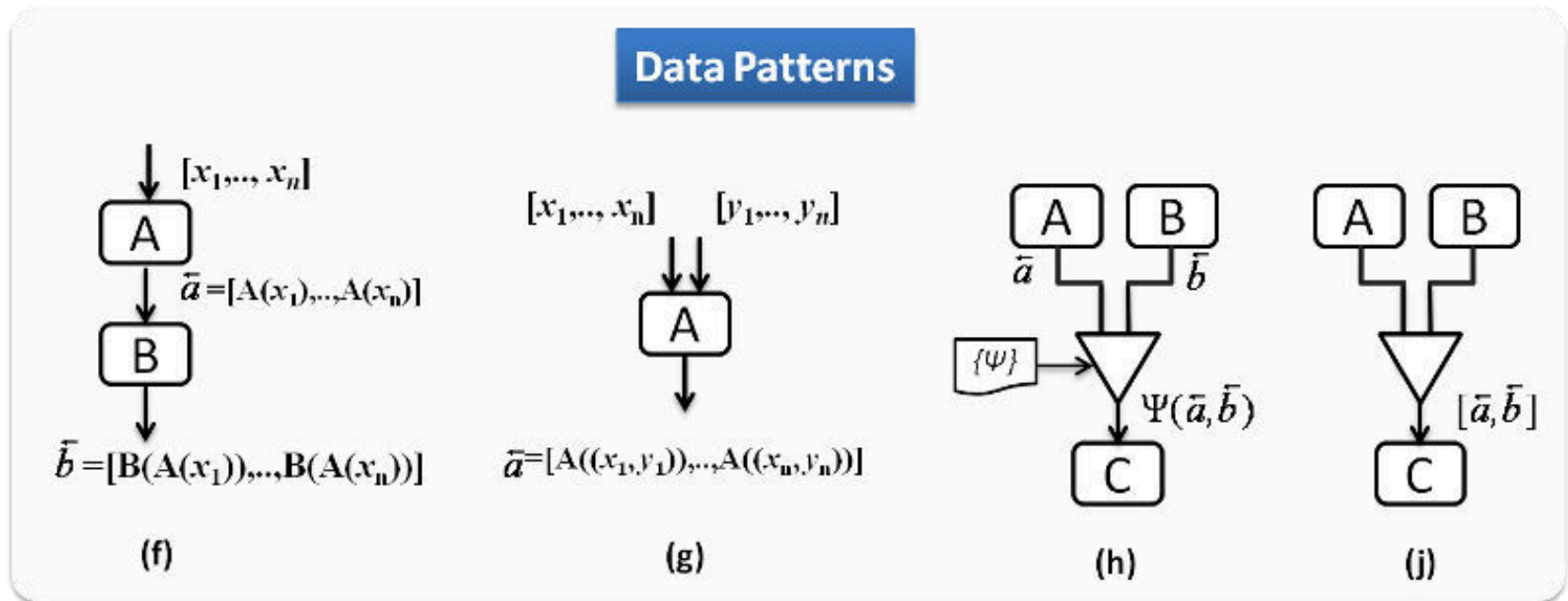
Databases

- Reference human genome hg36
- Indexed genome for MAQ, Bowtie
- NCBI nucleotide/protein DBs



Data Patterns of Tavaxy

To facilitate execution of data-intensive tasks in cloud computing cluster



Domenstration 1

Importing Taverna Workflow

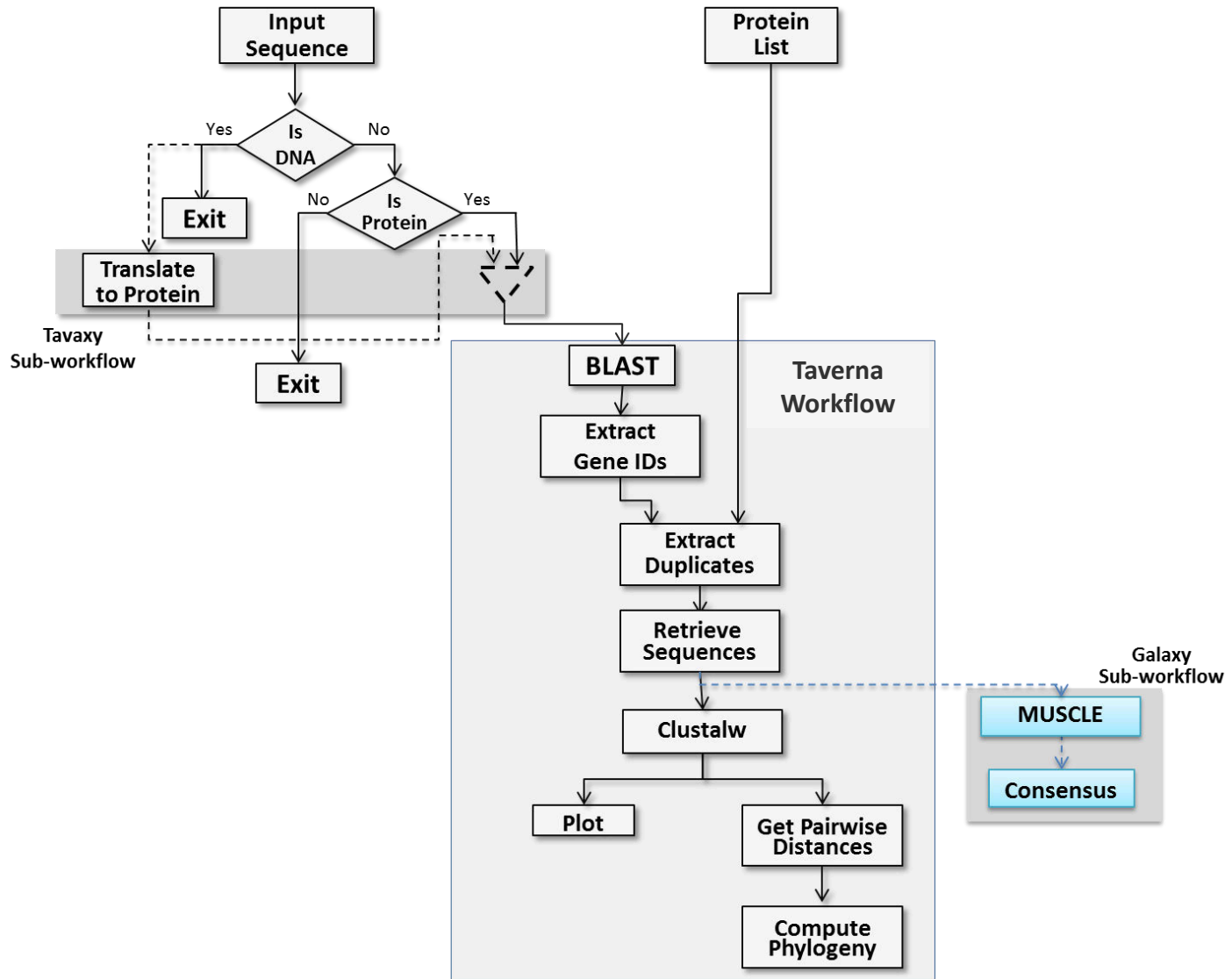
The *Tavaxy* Project

Single environment for integrating Galaxy and Taverna workflows:

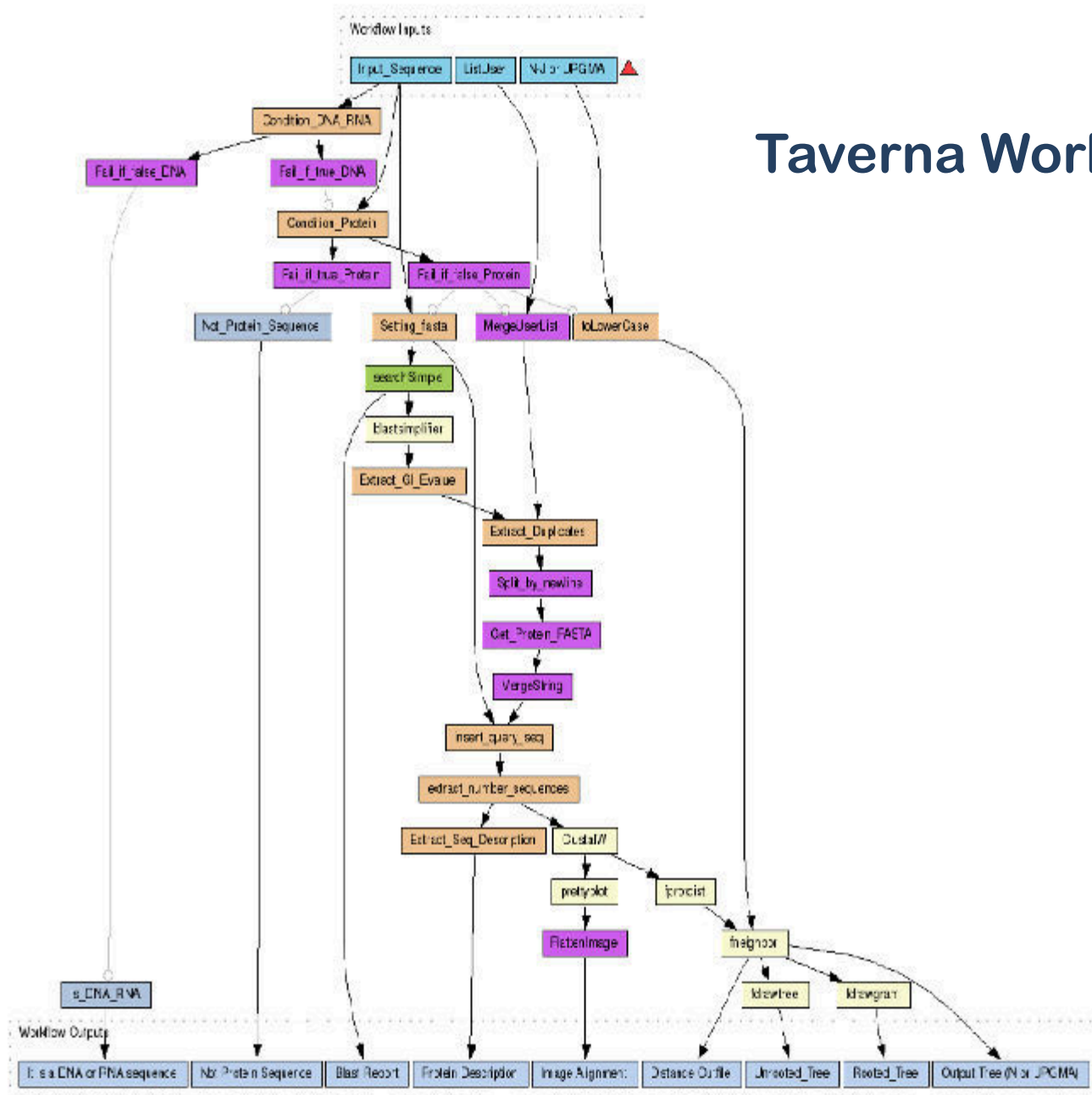
Taverna or Galaxy workflows can be

- imported , → open and re-draw in Tavaxy environment
- re-designed, → delete or re-order workflow nodes
- enhanced, → add new nodes and sub-workflows
- optimized,
 - Tavaxy constructs can be used to exploit parallelization
 - remote Taverna calls can be replaced with local tools
- and executed in Tavaxy
 - on local (HPC) infrastructure
 - on cloud

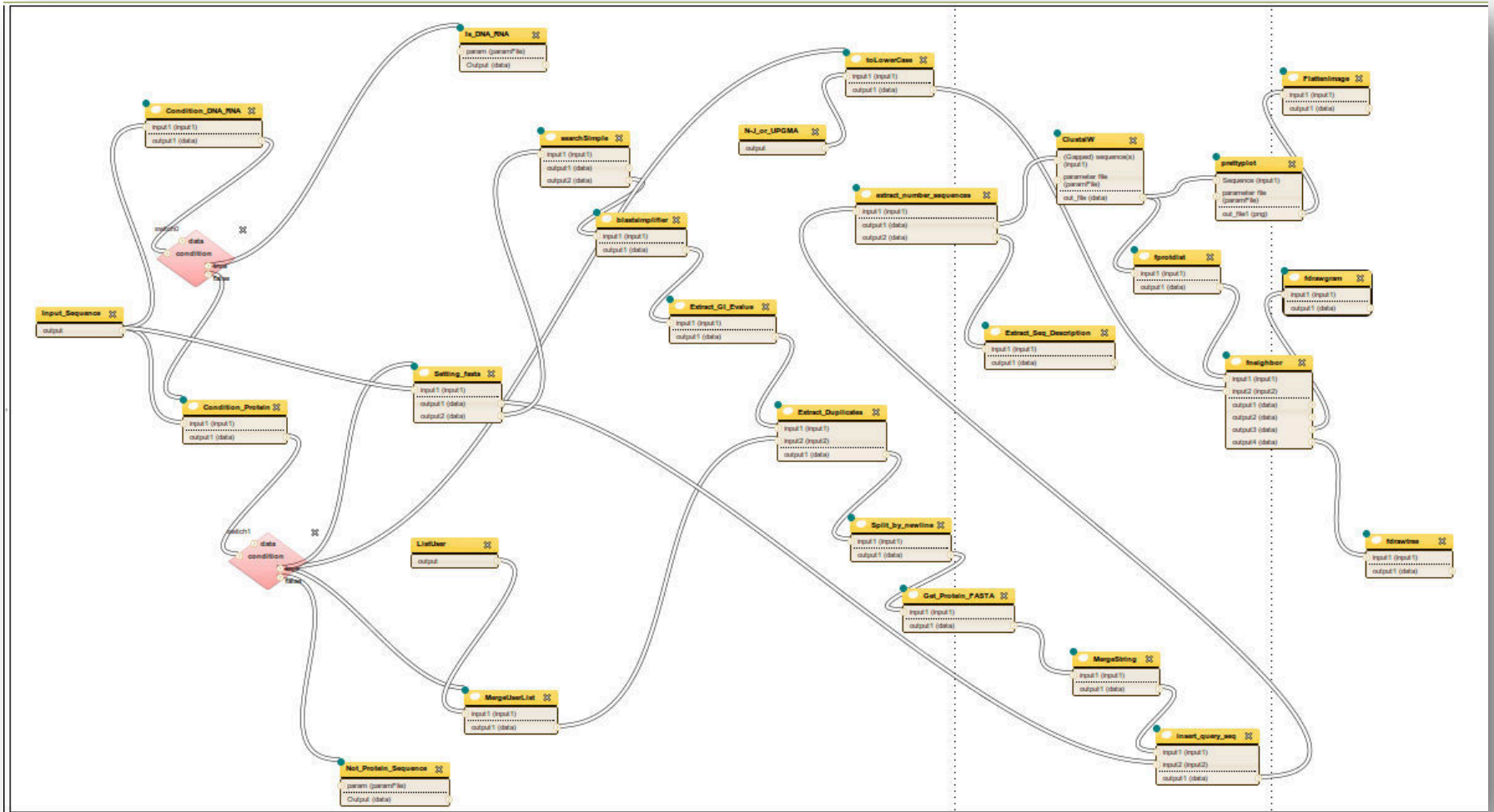
Importing Taverna Workflow



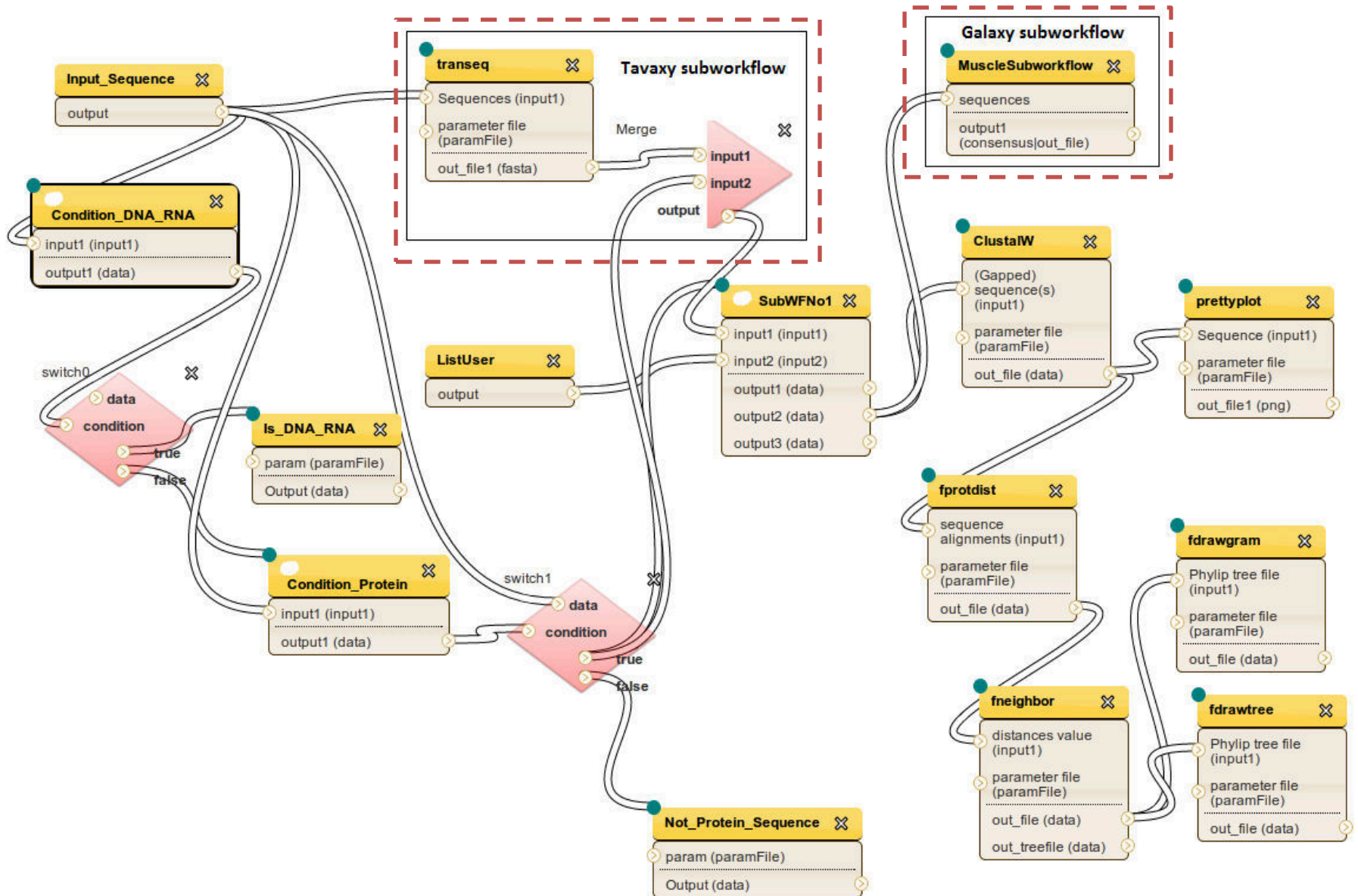
Taverna Workflow



Imported Taverna Workflow

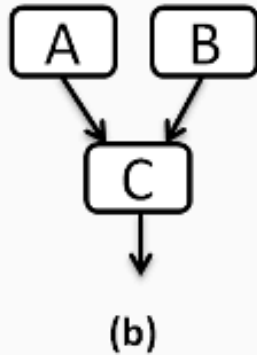


Optimized Imported Taverna Workflow

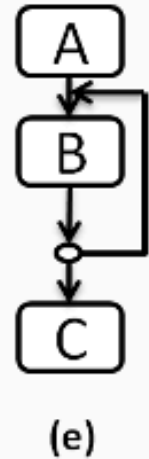
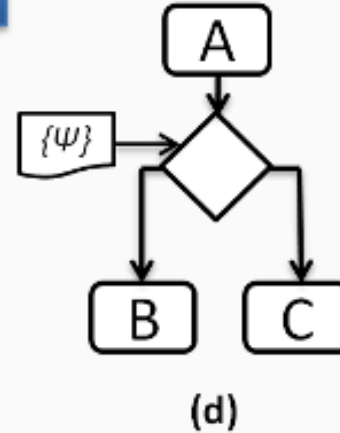
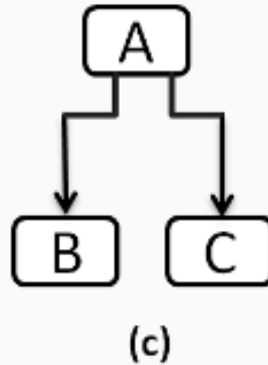


Idea of Integration

Workflow Patterns



Control Patterns



Bag of techniques

- Taverna is used as a secondary engine to execute Taverna sub-workflows
- The use of patterns as special nodes in the data-flow oriented engine
- Simulating control constructs over the data flow digraph representing the workflow
- Use of sub-workflow to enable iteration pattern

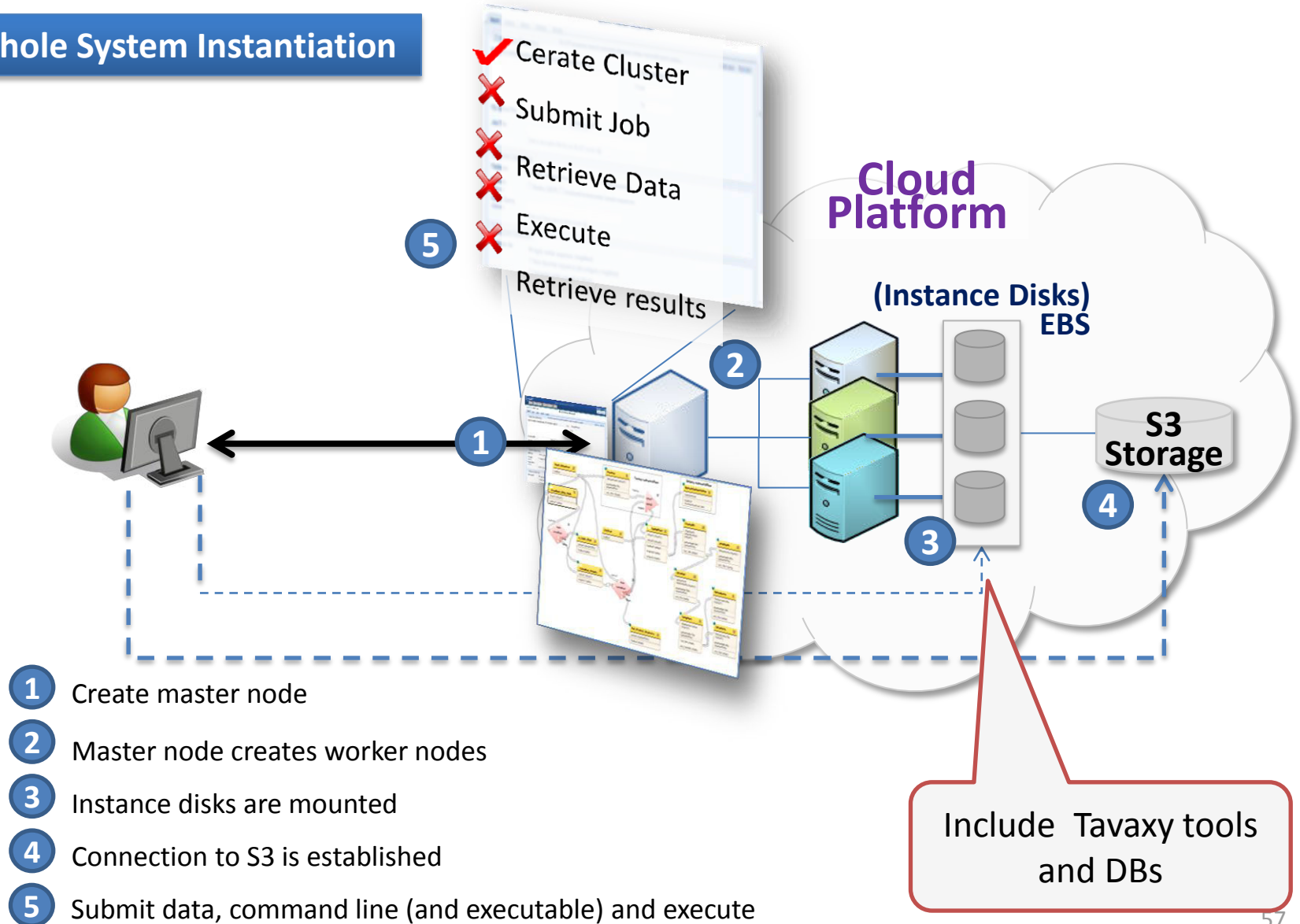
High Performance Cloud Computing Support

Three Modes for Supporting Cloud

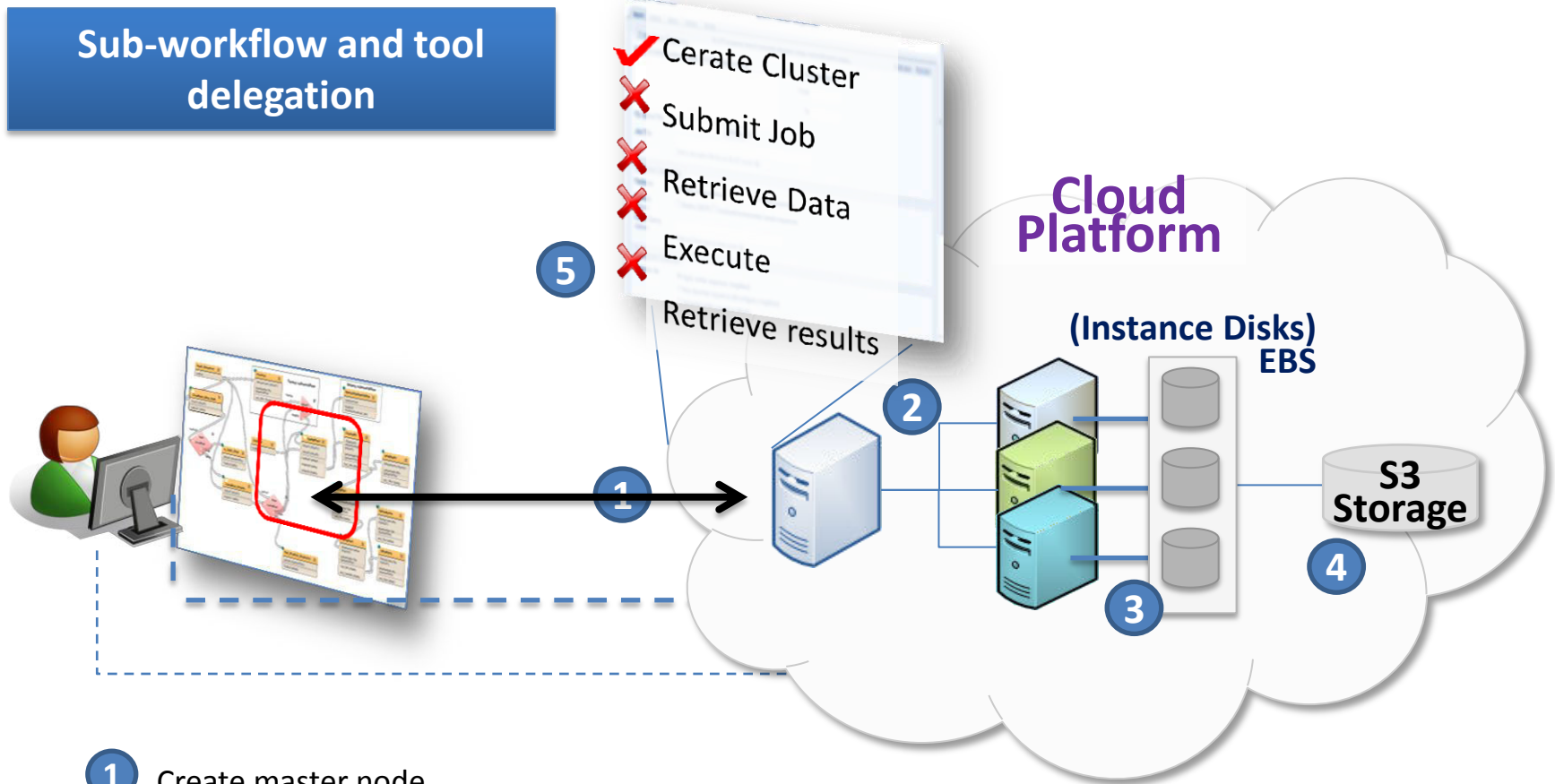
- Whole system instantiation
- Delegating execution of a sub-workflow to the cloud
- Delegating execution of one tool to the cloud

User case scenario

Whole System Instantiation

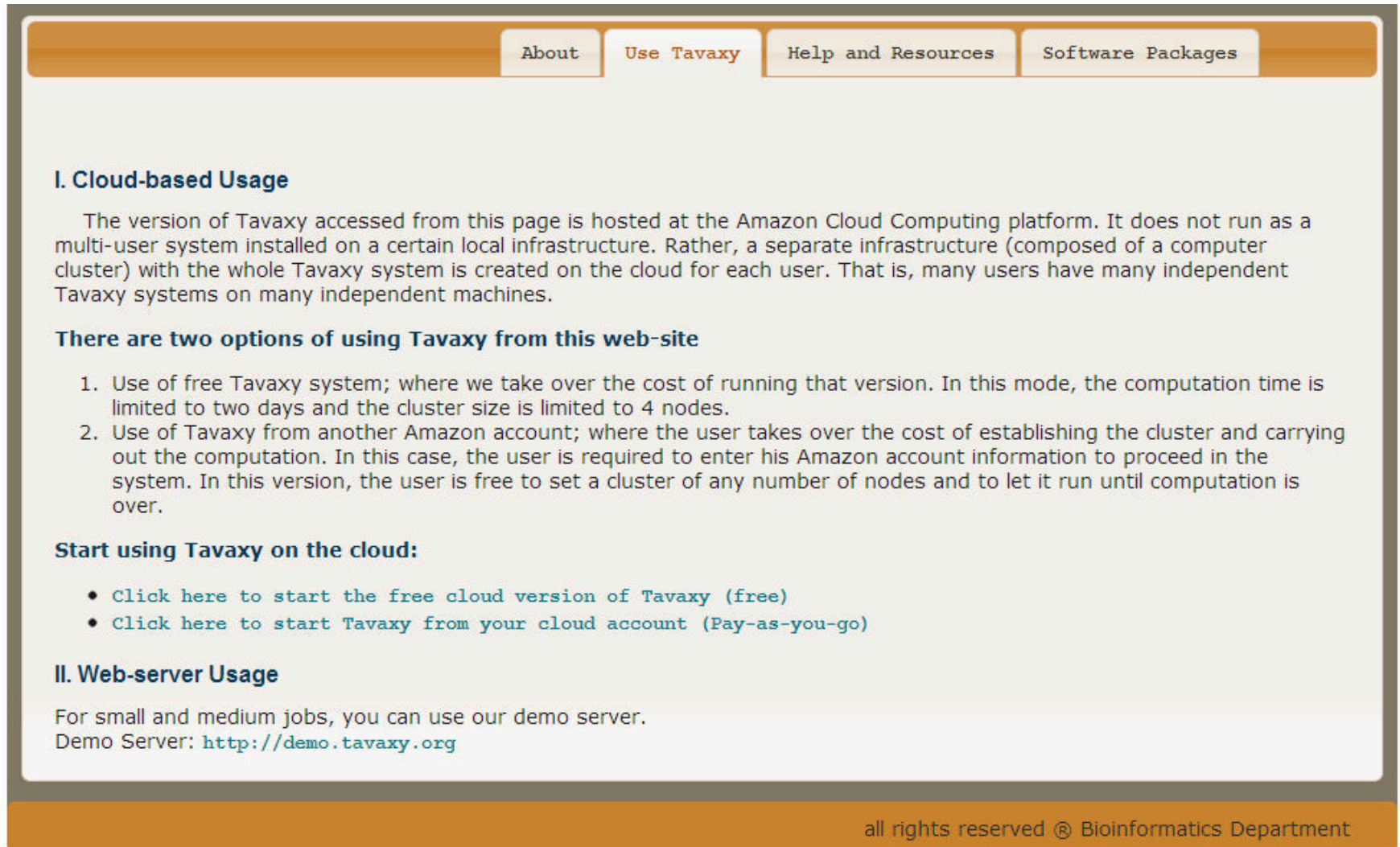


User case scenario



- 1 Create master node
- 2 Master node creates worker nodes
- 3 Instance disks are mounted
- 4 Connection to S3 is established
- 5 Submit data, command line (and executable) and execute

Whole System Instantiation



The screenshot shows a web browser window with a navigation bar at the top containing four tabs: "About", "Use Tavaxy", "Help and Resources", and "Software Packages". The "Use Tavaxy" tab is selected. Below the navigation bar, the page content is organized into sections. The first section is titled "I. Cloud-based Usage" and contains a paragraph explaining that the version of Tavaxy accessed from this page is hosted on the Amazon Cloud Computing platform. It states that it does not run as a multi-user system on local infrastructure but rather as a separate infrastructure created on the cloud for each user. Below this paragraph, there is a sub-section titled "There are two options of using Tavaxy from this web-site" followed by a numbered list of two options. The first option describes using the free Tavaxy system with a two-day limit and a four-node cluster size. The second option describes using Tavaxy from another Amazon account, where the user takes over the cost of establishing the cluster. Below the list, there is a sub-section titled "Start using Tavaxy on the cloud:" followed by two bullet points with links to start the free cloud version or the pay-as-you-go version. The second section is titled "II. Web-server Usage" and contains a paragraph stating that for small and medium jobs, the demo server can be used, with the URL <http://demo.tavaxy.org> provided. At the bottom of the page, there is a footer with the text "all rights reserved © Bioinformatics Department".

About Use Tavaxy Help and Resources Software Packages

I. Cloud-based Usage

The version of Tavaxy accessed from this page is hosted at the Amazon Cloud Computing platform. It does not run as a multi-user system installed on a certain local infrastructure. Rather, a separate infrastructure (composed of a computer cluster) with the whole Tavaxy system is created on the cloud for each user. That is, many users have many independent Tavaxy systems on many independent machines.

There are two options of using Tavaxy from this web-site

1. Use of free Tavaxy system; where we take over the cost of running that version. In this mode, the computation time is limited to two days and the cluster size is limited to 4 nodes.
2. Use of Tavaxy from another Amazon account; where the user takes over the cost of establishing the cluster and carrying out the computation. In this case, the user is required to enter his Amazon account information to proceed in the system. In this version, the user is free to set a cluster of any number of nodes and to let it run until computation is over.

Start using Tavaxy on the cloud:

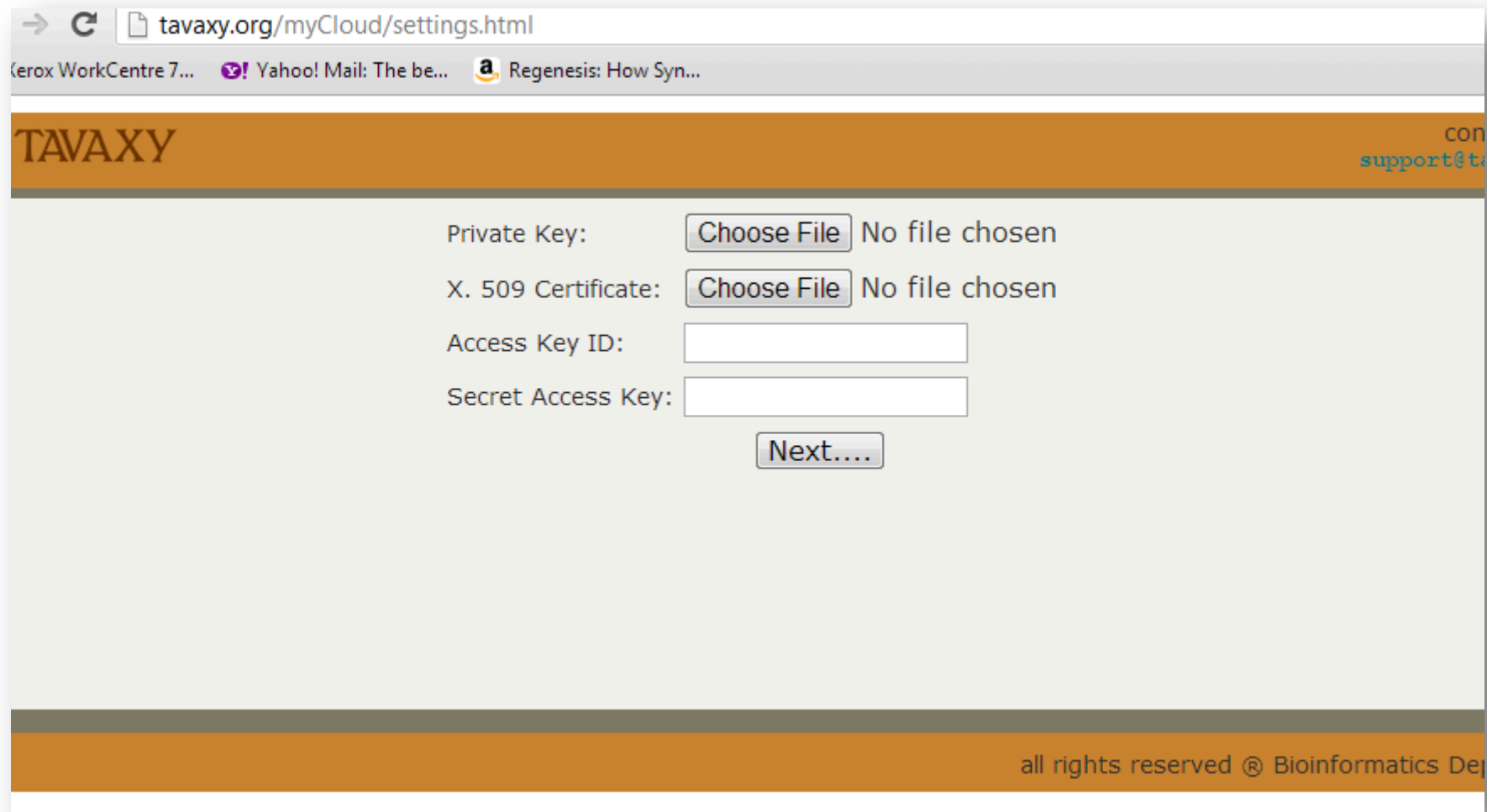
- [Click here to start the free cloud version of Tavaxy \(free\)](#)
- [Click here to start Tavaxy from your cloud account \(Pay-as-you-go\)](#)

II. Web-server Usage

For small and medium jobs, you can use our demo server.
Demo Server: <http://demo.tavaxy.org>

all rights reserved © Bioinformatics Department

Whole System Instantiation



The screenshot shows a web browser window with the address bar displaying `tavaxy.org/myCloud/settings.html`. The browser's tab bar shows several open tabs, including "Gerox WorkCentre 7...", "Yahoo! Mail: The be...", and "Regenesis: How Syn...". The TAVAXY website has an orange header with the logo on the left and "support@ta" on the right. The main content area is light gray and contains four configuration fields: "Private Key:" with a "Choose File" button and "No file chosen" text; "X. 509 Certificate:" with a "Choose File" button and "No file chosen" text; "Access Key ID:" with an empty text input field; and "Secret Access Key:" with an empty text input field. Below these fields is a "Next...." button. The footer is orange and contains the text "all rights reserved © Bioinformatics Dep".

→ ↻ `tavaxy.org/myCloud/settings.html`

Gerox WorkCentre 7... Yahoo! Mail: The be... Regenesis: How Syn...

TAVAXY support@ta

Private Key: No file chosen

X. 509 Certificate: No file chosen

Access Key ID:

Secret Access Key:

all rights reserved © Bioinformatics Dep

Sub-workflow/Tool Instantiation

The image displays a workflow editor interface with three main components:

- MySubWF (Sub-workflow):** A yellow node on the left containing inputs `db`, `query`, and `paramFile`, and outputs `output1 (Subworkflow|output1)`, `output2 (Subworkflow|output2)`, and `output3 (Subworkflow|output3)`.
- Tool: MySubWF:** A configuration panel for the sub-workflow node, showing:
 - Execute this node on: **Cloud**
 - Select cluster:
 - [Start New Cluster](#)
 - Select a workflow: **TestsubWF**
 - [Edit Workflow](#)
 - Help icon and text: "Promote any of the following outputs as output ports for this workflow. Promote a subworkflow in other workflow. For more information, check the 'Promote Outputs' section in the manual."
 - ☐ **output1**
- Megablast:** A configuration panel for the Megablast tool node, showing:
 - Execute this node on: **Local** (dropdown menu with options: Local, Cloud)
 - Compare these sequences (input_query)
 - against target database (source_select)
 - parameter file (paramFile)
 - output1 (tabular)
- Tool: Megablast:** A detailed configuration panel for the Megablast tool, showing:
 - Execute this node on: **Local** (dropdown menu with options: Local, Cloud)
 - Compare these sequences (input_query)
 - against target database (source_select)
 - using word size (word_size): **28**
 - report hits above this identity (-p) (iden_cutoff): **90.0**
 - set expectation value cutoff (-e) (evalue_cutoff): **0.001**
 - Filter out low complexity regions? (-F) (filter_query)

Exploiting Cloud Computing

Two steps:

- 1- Enter AWS credentials
- 2- Define your cluster

☐ Enable AWS Cloud

Private Key: No file chosen

X. 509 Certificate: No file chosen

Access Key ID:

Secret Access Key:

Region: ☐ US (Virginia) ☒ Ireland

S3 Bucket: [Create New Bucket](#)

Security Group:

Key Pair:

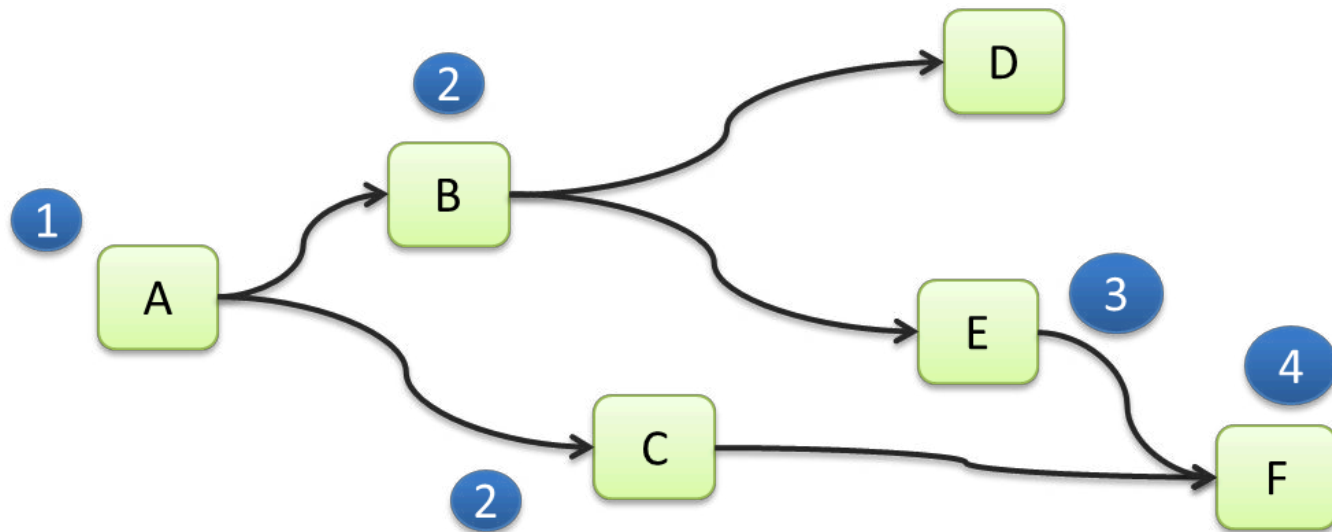
Cluster Size: Count: [What are these types?](#)

Cluster Name:

Supporting Task Parallelism

I. Parallelism due to branching

- Branching in the DAG implies independent execution
- Independent jobs can run in parallel

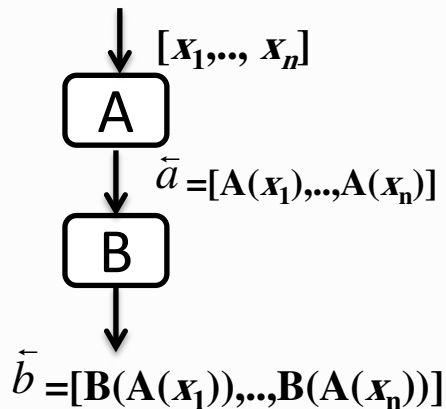


Supporting Data Intensive Tasks

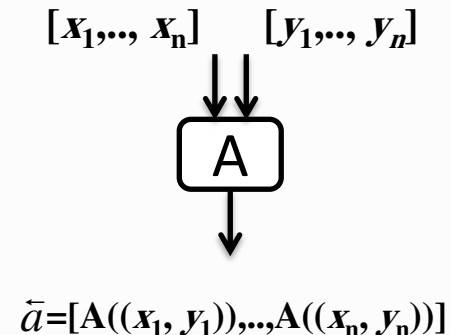
Data parallelism

- For an input as a list, node A can process the list items in parallel

Tavaxy Data Patterns



Single List



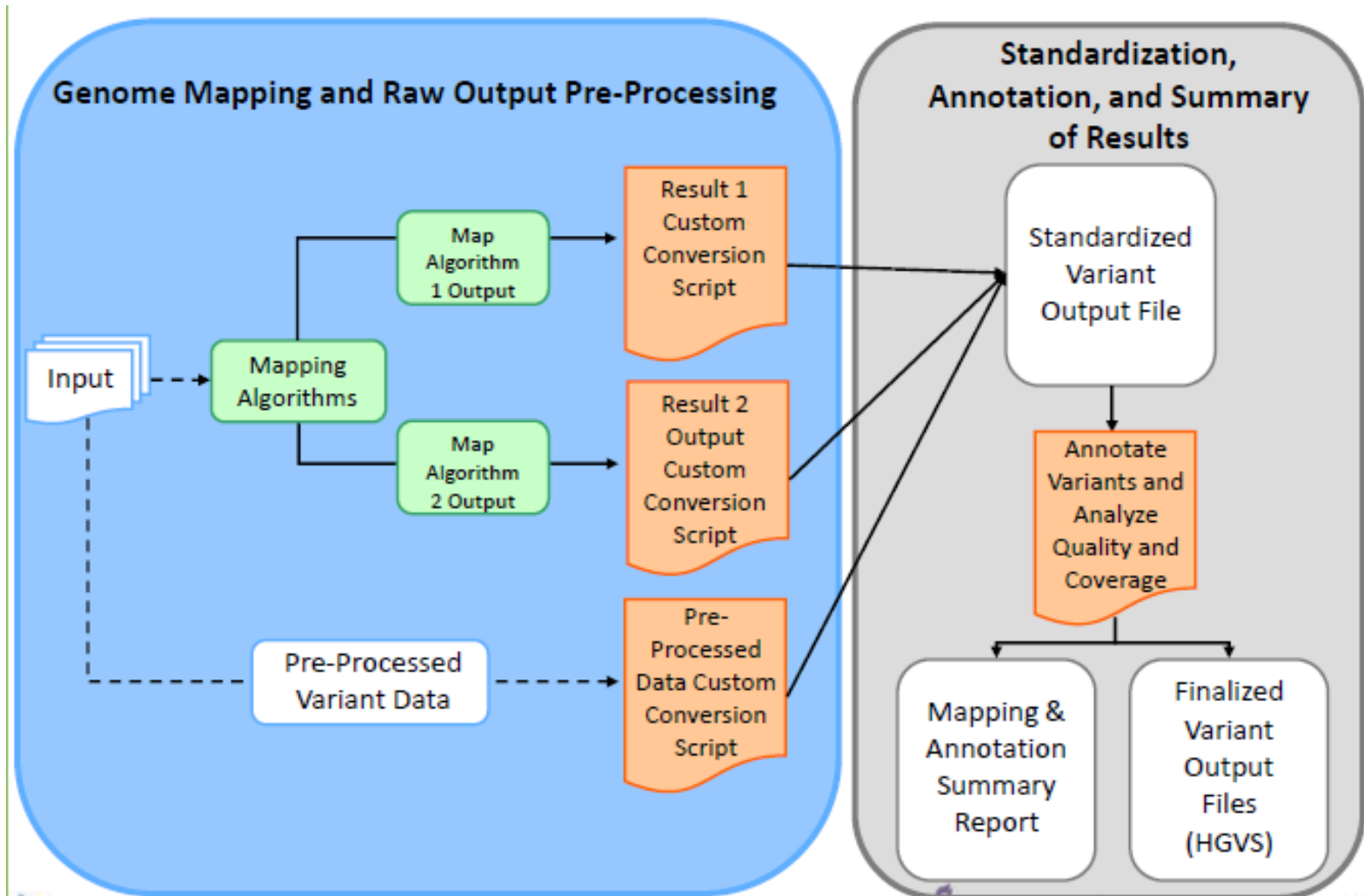
Multiple List

Personalized Medicine Workflow

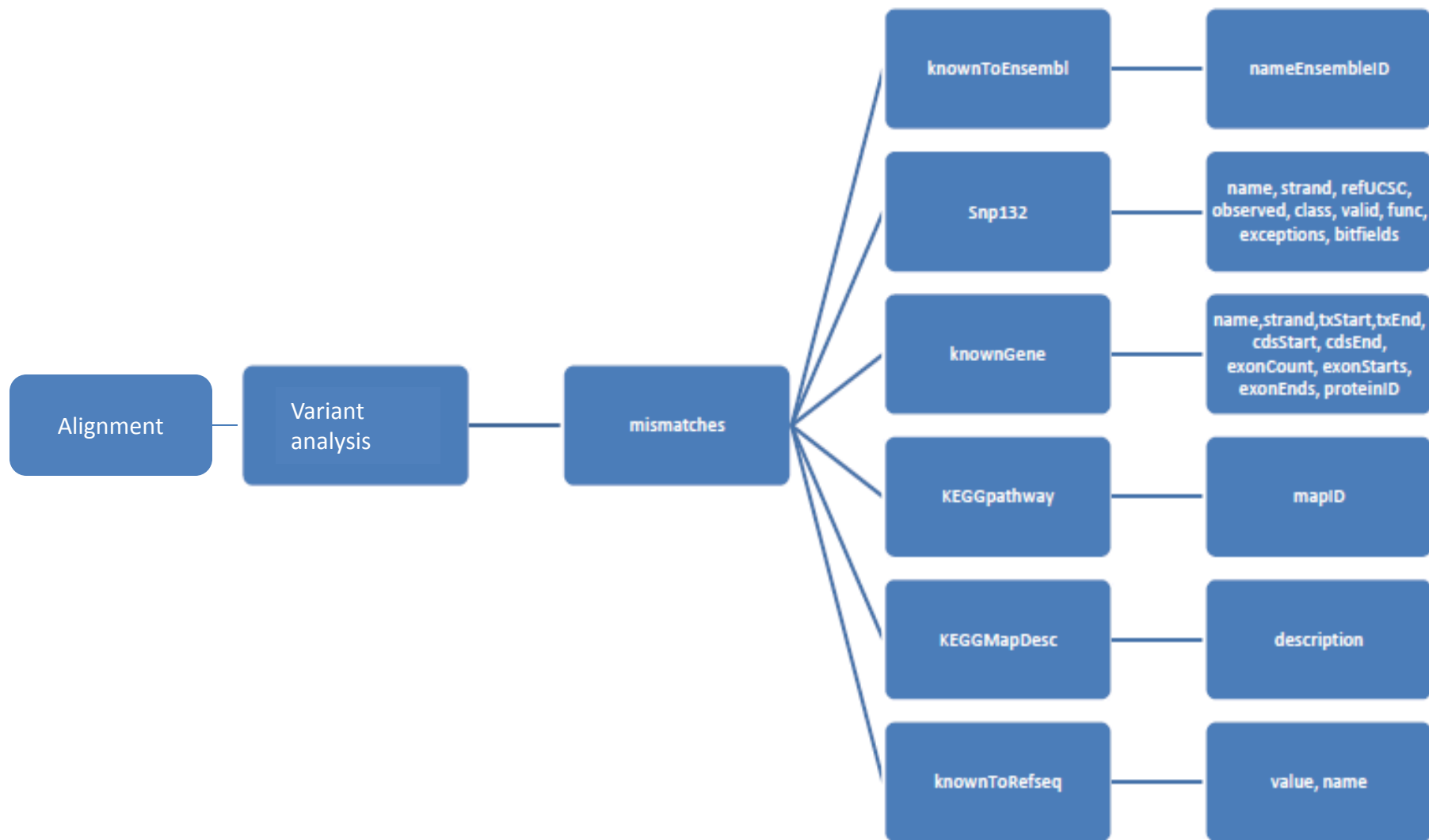
Personalized Medicine Workflow

Individual Whole Genome Mapping and Variant Annotation

Tonellato-Wall



Workflow Sktech



Personalized Medicine Workflow on Tavaxy

Read Mapping

index
output

Map_with_Bowtie_for_Illumina

Select the reference genome or index from history (ownFile)
parameter file (paramFile)
FASTQ file (singlePaired.slnput1)
output (sam)
outputIndex (data)

reads
output

Formatting

Sam_to_Bam

Convert SAM file (input)
Using reference file (ref_file)
parameter file (paramFile)
output1 (data)

ref_genome
output

Variant Calling

BioAnalysis

sorted alignment (alignment)
reference file (ref_file)
parameter file (paramFile)
output (data)

Annotation_from_Ensembl

output analysis (analysis)
reference file (ref_file)
parameter file (paramFile)
sift (data)
polyphen (data)
maf (data)
table (data)
html (data)

Annotation_from_KEGG

output analysis (analysis)
reference file (ref_file)
parameter file (paramFile)
sift (data)
polyphen (data)
maf (data)
table (data)
html (data)

Annotation_from_UCSC

output analysis (analysis)
reference file (ref_file)
parameter file (paramFile)
sift (data)
polyphen (data)
maf (data)
table (data)
html (data)

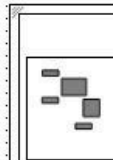
Annotation_from_SnpDB

output analysis (analysis)
reference file (ref_file)
parameter file (paramFile)
sift (data)
polyphen (data)
maf (data)
table (data)
html (data)

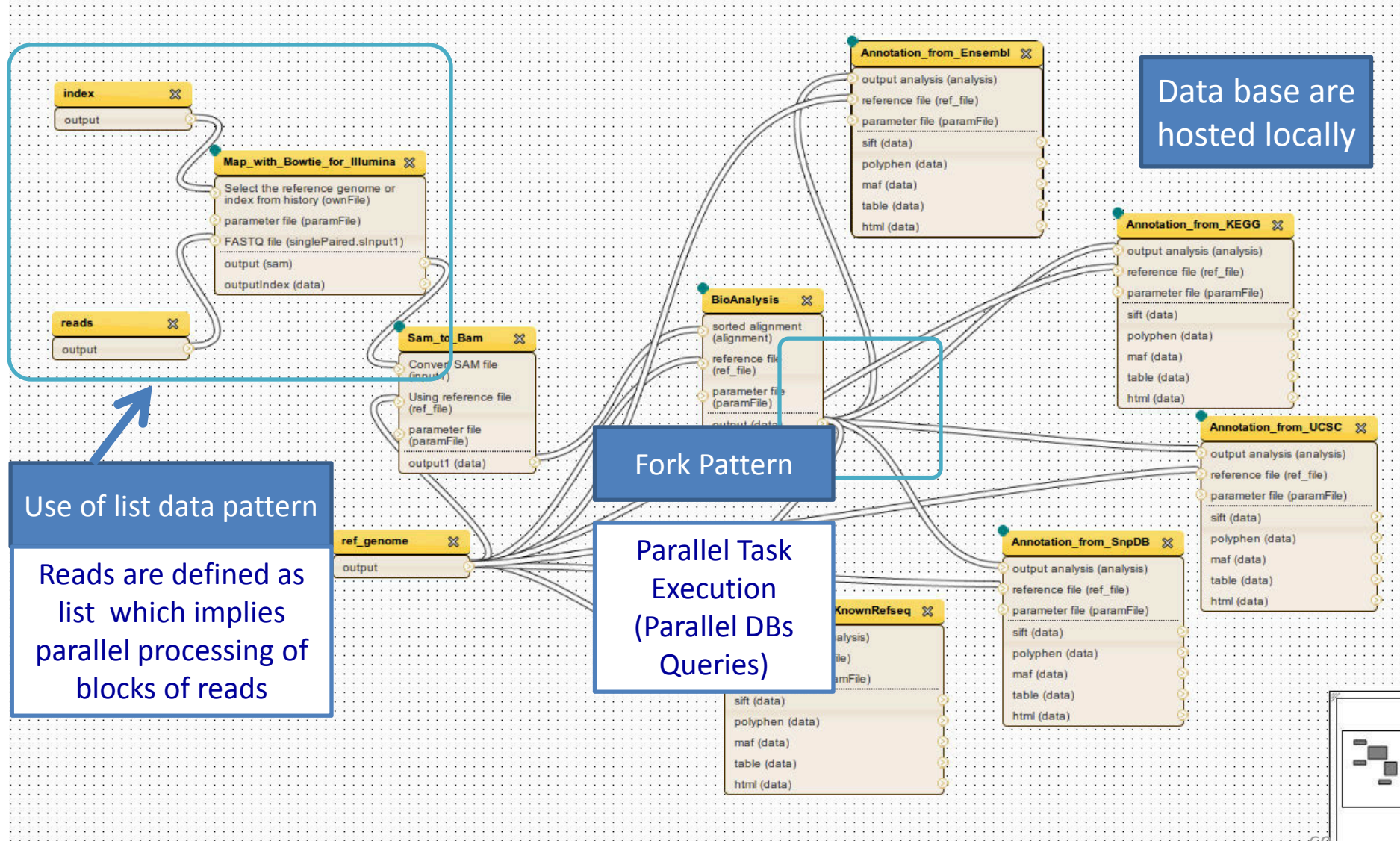
Annotation_from_KnownRefseq

output analysis (analysis)
reference file (ref_file)
parameter file (paramFile)
sift (data)
polyphen (data)
maf (data)
table (data)
html (data)

SNP/Disease
Database
Queries



Exploiting Parallelization and Locality of Data



Read-Mapping with Crossbow

- Crossbow (based on EMR/Hadoop) is used to map set of human reads to human genome.
- With this cluster, we mounted EBS volumes including the reference human genome files (each including one chromosome)
- Read Datasets:
 - illumina reads of around 13 Gbp (47 GB) from from the African genome,
 - the human genome version hg18, build 36

Read-Mapping with Crossbow

Table 4 - Running times of Crossbow on EMR using *elasticHPC*.

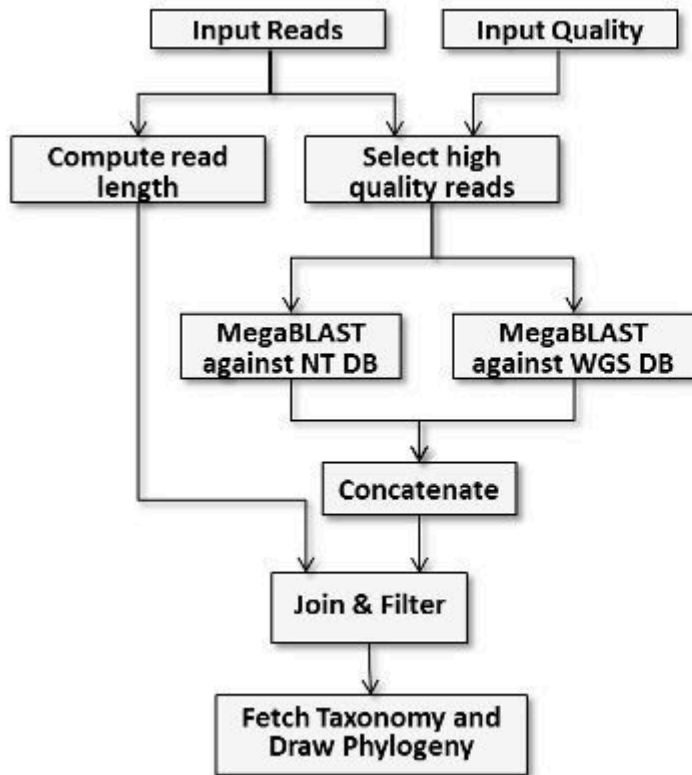
Num Nodes	Num Cores	Processing Time	Mapping Time	Total Time	Cost
Using c1.xlarge					
1	8	66m	769m	835.1m	\$1.68
4	32	39.5m	258.6m	298.4m	\$2.4
8	64	35.25m	121.5m	156.8m	\$2.88
16	128	34.1m	62.6m	96.9m	\$3.84
24	192	33m	46.6m	79.8m	\$5.76
32	256	33.0m	39.95m	73.5m	\$7.68
64	512	32.65m	23.6m	56.1m	\$7.68
Using m1.xlarge					
1	4	72.2m	1675.6m	1748m	\$2.7
4	16	40.6m	431.4m	472.6m	\$2.88
8	32	37.3m	263.8m	301.1m	\$4.32
16	64	33.6m	95.6m	129.6m	\$4.32
24	96	32.9m	54.2m	87.1m	\$4.32
32	128	32.6m	51.5m	84.3m	\$5.76
64	256	32.8m	33.3m	66.1m	\$11.52

The average running times in minutes for EMR clusters of different sizes and machine types in the cloud. The machine types are c1.xlarge and m1.xlarge. A number in the column titled 'Total Time' is the summation of the pre-processing and alignment times.

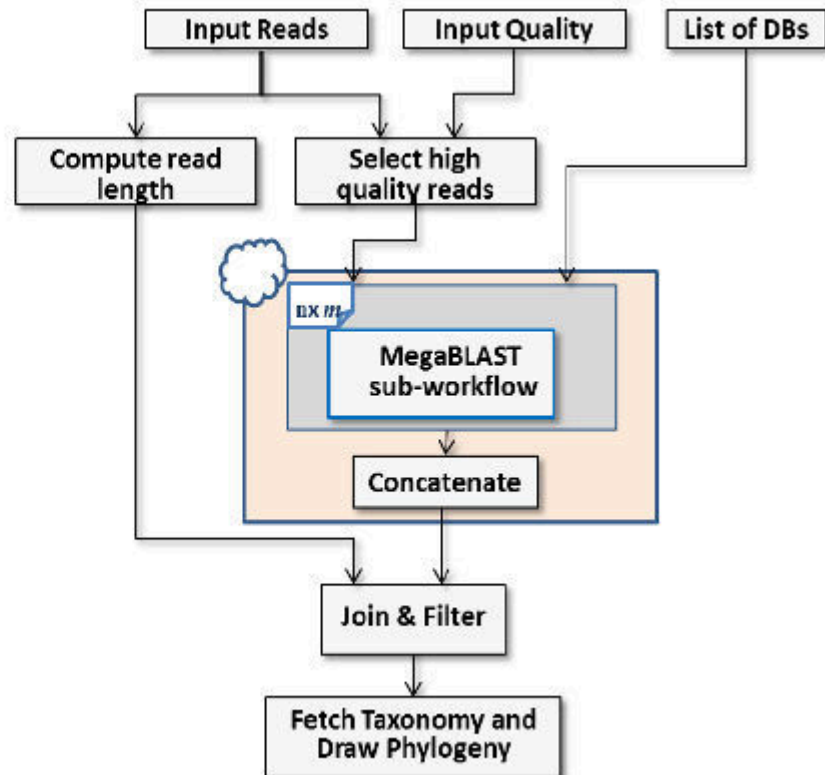
Domenstration 3

Metagenomics Workflow

NGS-based Metagenomics Study

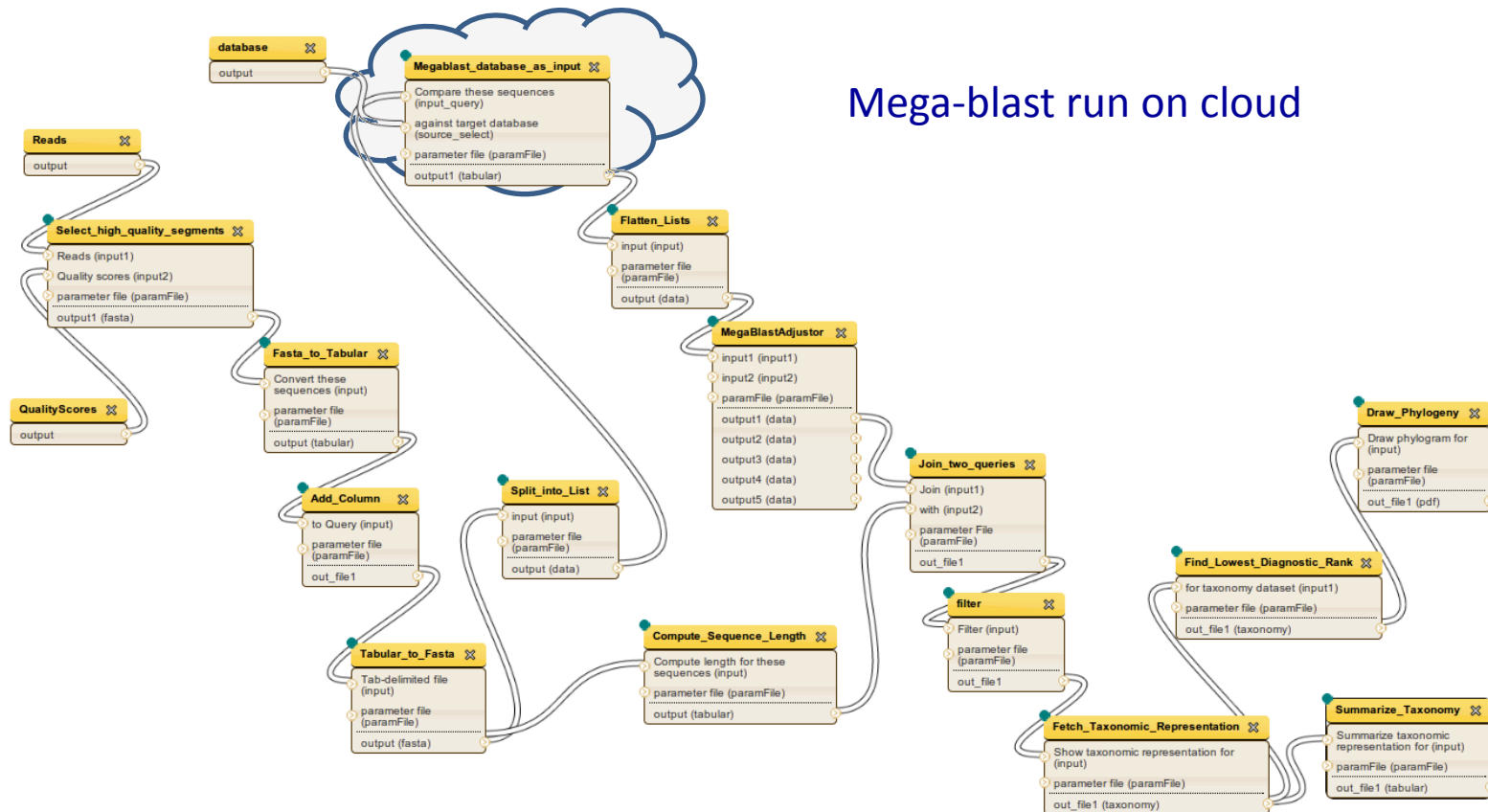


Galaxy Workflow



Optimized Tavaxy Workflow

NGS-based Metagenomics Study



NGS-based Metagenomics Study

- The MegaBLAST program is used to annotate a set of sequences coming from a metagenomics experiment.
- With this cluster, we mounted EBS volumes including the NCBI NT database
- Datasets: Windshield dataset composed of two collections of 454 FLX reads:
 - Trip A: 138575 (25.3 Mbp) Mb104283 (18.8 Mbp)
 - Trip B:) 151000 (30.2 Mbp) 79460 (12.7 Mbp)

Bioinformatics Experiment (2)

Dataset	AWS Cores				
	1	8	16	32	64
c1.xlarge (8 cores)					
	1 node	1 node	2 nodes	4 nodes	8 nodes
Trip A Left	93 (\$1.32)	27 (\$0.66)	20 (\$1.32)	13 (\$2.64)	9(\$5.28)
Trip A Right	127 (\$1.98)	33 (\$0.66)	21 (\$1.32)	13 (\$2.64)	7(\$5.28)
Trip B Left	80 (\$1.32)	25 (\$0.66)	17 (\$1.32)	13 (\$2.64)	7(\$5.28)
Trip B Right	65 (\$1.32)	23 (\$0.66)	13 (\$1.32)	8 (\$2.64)	6(\$5.28)
Total	365 (\$4.62)	108 (\$1.19)	71 (\$2.64)	47 (\$2.64)	29(\$5.28)
m1.xlarge (4 cores)					
	1 node	2 nodes	4 nodes	8 nodes	16 nodes
Trip A Left	77 (\$1.28)	18 (\$0.64)	13 (\$2.64)	9 (\$5.12)	7(\$10.24)
Trip A Right	119 (\$1.28)	34 (\$0.64)	25 (\$2.64)	16 (\$5.12)	10(\$10.24)
Trip B Left	70 (\$1.28)	31 (\$0.64)	23 (\$2.64)	15 (\$5.12)	9(\$10.24)
Trip B Right	65 (\$1.28)	27 (\$0.64)	13 (\$2.64)	9 (\$5.12)	6(\$10.24)
Total	331 (\$2.56)	110 (\$1.28)	74 (\$5.12)	49 (\$5.12)	32(\$10.24)

The average running times in minutes for traditional computer clusters of different sizes and machine types in the cloud. The machine types are c1.xlarge and m1.xlarge. The numbers in brackets are the computation costs in US Dollar for the US-East site with \$0.66 per hour for c1.xlarge and \$0.64 per hour for m1.xlarge. (Note that partial computing hour of an instance is billed on Amazon as a full hour) The cost in the column titled "Total Time" is not the summation of the above rows, but it is the cost of the total running time if the four datasets in the respective column were processed altogether in the cluster.

Conclusions and Future Work

- Use of workflow systems provides flexibility and efficiency
- High performance and cloud computing resources are exploited with technical details being hidden
- Future work include
 - Further performance optimization for execution on local and cloud infrastructures.
 - Supporting multiple cloud providers
 - Handling larger data sizes in multi-use environment

Thanks for attention