

Galaxy

Translating workflows into Nextflow with Janis

Galaxy Platform

Online platform for data-analysis

- GUI rather than CLI
- Easy data handling
- Compute provided
- Easy to get started

Functionality is impressive

- Each server has collection of Tools
- History documents your analysis
- Can create a workflow for your analysis

Chose to support Galaxy ingest in janis-translate because..

- Many Galaxy tools & workflows publicly available
- Natural progression for users: Galaxy → CLI
- Workflow editor makes prototyping fast

Galaxy Australia

Galaxy AUSTRALIA

Galaxy Australia invited by GENCODE GALAXY COMMUNITY CONFERENCE 10-16 July 2023 #UseGalaxyEU23

Galaxy Australia is an open, web-based platform for accessible, reproducible and transparent computational research. Galaxy supports thousands of documented and maintained tools that are free to use. We facilitate on-demand training capacities and provision 600GB for Australian institutional (and 100GB for other) users.

AlphaFold 2.0 on Galaxy Australia
Apply now

9000+ Tools & Datasets Ready to install
Request now

Additional storage Available on request
Request now

Features

Each server has collection of Tools. Select a tool, supply inputs, execute.

The screenshot displays the Galaxy Australia interface for the FastQC tool. The left sidebar shows the 'Tools' section with 'fastqc' selected and highlighted by a red dashed box. Below it, the 'WORKFLOWS' section is visible. The central area shows the 'FastQC Read Quality reports (Galaxy Version 0.73+galaxy0)' tool configuration page. The 'Raw read data from your current history' section is highlighted with a red dashed box and contains the input '14: illumina_reads_1.fastq'. Other sections include 'Contaminant list', 'Adapter list', 'Submodule and Limit specifying file', and 'Email notification'. The 'Execute' button at the bottom is also highlighted with a red dashed box. The right sidebar shows the 'History' section with 'Unnamed history' and a list of datasets, including '14: illumina_reads_1.fastq' which is highlighted with a green background and a red dashed box.

Features

History documents your analysis - queued, running, paused, complete

The screenshot shows the Galaxy Australia interface. On the left, the 'Tools' panel is visible with 'fastqc' selected. A green notification box in the center states: 'Executed **FastQC** and successfully added 1 job to the queue. The tool uses this input: • 14: illumina_reads.1.fastq It produces 2 outputs: • 15: FastQC on data 14: Webpage • 16: FastQC on data 14: RawData You can check the status of queued jobs and view the resulting data by refreshing the History panel. When the job has been run the status will change from 'running' to 'finished' if completed successfully or 'error' if problems were encountered.' The 'History' panel on the right shows a list of jobs: '16: FastQC on data 14: RawData', '15: FastQC on data 14: Webpage', and '14: illumina_reads.1.fastq'. The first two jobs are highlighted with a red dashed box.

The screenshot shows the 'History' panel. It includes a search bar for datasets and a 'My History' section. The 'My History' section displays a list of jobs with their status and size:

Status	Count	Size
queued	5	4.15 MB
paused	4	
running	3	
error	2	
ok	1	

Features

Create a workflow for your analysis. From analysis history, or from scratch.

The screenshot displays the Galaxy Australia web interface. At the top, the navigation bar includes 'Workflow Visualize Shared Data Help User' and a 'Using 18%' indicator. The main workspace shows a workflow titled 'Unicycler Assembly' on a grid background. The workflow consists of several interconnected tools:

- 4: fastQC**: Three instances of the fastQC tool, each with a detailed configuration panel. The panels show options for 'Raw read data from your current history', 'Contaminant list', 'Adapter list', 'Submodule and Limit specifying file', and 'FastQC on input datasets: Webpage (html)'. The first instance also includes 'FastQC on input datasets: RawData (txt)'.
- 5: Create assemblies with Unicycler**: A central tool with a configuration panel for 'select first set of reads', 'select second set of reads', 'select long reads', 'FASTA file of genes for start point of riddled regions', and 'FASTA file of known contamination in long reads'. It also includes options for 'Create assemblies with Unicycler on input datasets: Final Assembly Graph (gfa)' and 'Create assemblies with Unicycler on input datasets: Final Assembly (fasta)'.
- 6: Quast**: A tool with a configuration panel for 'Configurable/folds file', 'Quast on input datasets: HTML report (html)', and 'Sequences to analyse'.
- 8: Busco**: A tool with a configuration panel for 'Busco on input datasets: short summary (txt)' and 'Busco on input datasets: full table (tabular)'.

On the left side, there is a 'Tools' sidebar with a search bar and a list of tool categories: FILE AND META TOOLS, Get Data, Send Data, Collection Operations, GENERAL TEXT TOOLS, Text Manipulation, Filter and Sort, Join, Subtract and Group, GENOMIC FILE MANIPULATION, FASTA/FASTQ, FASTQ Quality Control, SAM/BAM, BED, VCF/BCF, Nanopore, Convert Formats, Lift-Over, COMMON GENOMICS TOOLS, Operate on Genomic Intervals, MiModD, Fetch Alignments/Sequences, and GENOMICS ANALYSIS. The bottom status bar shows a zoom level of 67%.

On the right side, there is a metadata panel for the workflow:

- Name**: Unicycler Assembly
- Version**: 7: Jun 1st 2023, 9 steps
- Annotation**: (empty field)
- Notes**: These notes will be visible when this workflow is viewed.
- License**: Specify a license for this workflow.
- Creator**: Add a new creator - either a person or an organization.
- Tags**: Apply tags to make it easy to search for and find items with the same tag.

Users

Biologists, Career bioinformaticians (due to great accessibility)

A number of prominent research ventures use the Galaxy platform to analyse their data:

- The Encyclopedia of DNA Elements (ENCODE)
- The Vertebrate Genomes Project (VGP)
- Assistance Publique–Hôpitaux de Paris (AP-HP)

Accessing Tools

Tools can be accessed 2 ways:

- Via a Galaxy server (<https://usegalaxy.org.au/>)
- Via the Galaxy Toolshed (<https://toolshed.g2.bx.psu.edu/>)

Accessing Tools

Tools can be accessed 2 ways:

- Via a Galaxy server (<https://usegalaxy.org.au/>)
- Via the Galaxy Toolshed (<https://toolshed.g2.bx.psu.edu/>)

Via Galaxy Server

The screenshot shows the Galaxy Australia web interface. On the left, a sidebar titled 'Tools' lists various bioinformatics tools. A red box highlights the 'Tools' sidebar. The main content area displays the 'Samtools idxstats' tool page, which includes a description, an 'Email notification' section, and a 'What it does' section. The 'What it does' section contains the following text: 'Runs the samtools idxstats command. It retrieves and prints stats in the index file. Input is a sorted and indexed BAM file, the output is tabular with four columns (one row per reference sequence plus a final line for unmapped reads):' followed by a table with columns 'Column' and 'Description'. The table lists: 1. Reference sequence identifier, 2. Reference sequence length, 3. Number of mapped reads, and 4. Number of paired but unmapped reads (typically unmapped partners of mapped reads). Below the table, an 'Example output from a de novo assembly:' is shown with a tabular representation of the tool's output.

Via Toolshed

The screenshot shows the Galaxy Tool Shed web interface. The top navigation bar includes 'Repositories', 'Groups', 'Help', and 'User'. The main content area is titled '9425 valid tools on Apr 6' and 'Repositories by Category'. Below this, there is a search bar for 'Search for valid tools' and a section for 'Valid Galaxy Utilities'. The 'Available Actions' section includes 'Login to create a repository'. The 'Repositories by Category' section is a table with the following data:

Name	Description	Repositories
Assembly	Tools for working with assemblies	196
Astronomy	Tools for astronomy	
CHIP-seq	Tools for analyzing and manipulating CHIP-seq data.	77
Climate Analysis	Tools for analyzing climate data	12
CLIP-seq	Tools for CLIP-seq	4
Combinatorial Selections	Tools for combinatorial selection	9
Computational chemistry	Tools for use in computational chemistry	180
Constructive Solid Geometry	Tools for constructing and analyzing 3-dimensional shapes and their properties	11
Convert Formats	Tools for converting data formats	138

Accessing Tools - Tool Wrapper Format

Galaxy Tool Wrappers are written in XML.

They have 2 roles:

- Expose a user interface (UI)
- Execute CLI commands to run software

Accessing Tools - Tool Wrapper Format

Galaxy Tool Wrappers are written in XML.

They have 2 roles:

- Expose a user interface (UI)
- Execute CLI commands to run software

```
<command detect_errors="exit_code"><![CDATA[
#import re
#set $sample_name = re.sub('[^\w\-\.\_]', '_', $file_input.
element_identifier)

ln -sf '${file_input}' ${sample_name} &&

abricate ${sample_name}
$adv.no_header
#if $adv.min_dna_id
| --minid=$adv.min_dna_id
#end if
#if $adv.min_cov
| --mincov=$adv.min_cov
#end if
2>&1
--db=$adv.db
> '$report'
]]></command>
```

abricate.xml command

```
<inputs>
<param name="file_input" type="data" format="fasta,genbank,embl"
label="Input file (Fasta, Genbank or EMBL file)" help="To screen for
antibiotic resistant genes, can be a fasta file, a genbank file or an
EMBL file." />
<section name="adv" title="Advanced options" expanded="False">
<param argument="--db" type="select" label="Database to use -
default is 'resfinder'" help="Option to switch to other AMR/other
database">
<option value="argannot">ARG-ANNOT</option>
<option value="card">CARD</option>
<option value="ecoh">EcoH</option>
<option value="ncbi">NCBI Bacterial Antimicrobial Resistance
Reference Gene Database</option>
<option value="resfinder" selected="true">Resfinder</option>
<option value="plasmidfinder">PlasmidFinder</option>
<option value="vfdb">VFDB</option>
<option value="megares">megares</option>
<option value="ecoli_vf">Ecoli_VF</option>
</param>
<param name="no_header" argument="--noheader" type="boolean"
truevalue="--noheader" falsevalue="" label="Suppress header"
help="Suppress output file's column headings" />
<param name="min_dna_id" argument="--minid" type="float"
value="80" min="0" max="100" optional="true" label="Minimum DNA
%identity" />
<param name="min_cov" argument="--mincov" type="float" value="80"
min="0" max="100" optional="true" label="Minimum DNA %coverage" />
</section>
</inputs>

<outputs>
<data name="report" format="tabular" label="${tool.name} on $
{on_string} report file" />
</outputs>
```

abricate.xml I/O

Accessing Tools - Tool Wrapper Format

Can get super complicated!

Translation becomes very challenging.

- Runtime values can affect the structure of the shell command
- Won't know what the command will look like till it actually runs
- Sometimes impossible to parse correctly.

Accessing Tools - Tool Wrapper Format

Can get super complicated!

Translation becomes very challenging.

- Runtime values can affect the structure of the shell command
- Won't know what the command will look like till it actually runs
- Sometimes impossible to parse correctly.

```
1  <command detect_errors="exit_code"><CDATA[
2  ## Link in the input and output files, so Cutadapt can tell their type
3
4  #import re
5  #set read1 = "input.f"
6  #set read2 = "input.r"
7  #set paired = False
8  #set library_type = str($library.type)
9  if $library_type = "paired":
10 #set paired = True
11 #set read1 = re.sub("[^\w-]", "_", str($library.input_1.element_identifier))
12 #set read2 = re.sub("[^\w-]", "_", str($library.input_2.element_identifier))
13 #set input_1 = $library.input_1
14 #set input_2 = $library.input_2
15 #else if $library_type == "paired_collection"
16 #set paired = True
17 #set input_1 = $library.input_1.Forward
18 #set input_2 = $library.input_1.Reverse
19 #set read1 = re.sub("[^\w-]", "_", str($library.input_1.name)) + "_1"
20 #set read2 = re.sub("[^\w-]", "_", str($library.input_1.name)) + "_2"
21 #else
22 #set input_1 = $library.input_1
23 #set read1 = re.sub("[^\w-]", "_", str($library.input_1.element_identifier))
24 #end if
25
26 if $input_1.is_of_type("fastq.gz", "fastqsanger.gz"):
27 #set read1 = $read1 + ".fq.gz"
28 #set out1 = "out1.gz"
29 #else if $input_1.is_of_type("fastq.bz2", "fastqsanger.bz2"):
30 #set read1 = $read1 + ".fq.bz2"
31 #set out1 = "out1.bz2"
32 #else if $input_1.is_of_type('fasta'):
33 #set read1 = $read1 + ".fa"
34 #set out1 = "out1.fa"
35 #else:
36 #set read1 = $read1 + ".fq"
37 #set out1 = "out1.fq"
38 #end if
39 ln -f -s '$input_1' '$read1' @
40
41 if $paired:
42 #if $input_2.is_of_type("fastq.gz", "fastqsanger.gz"):
43 #set read2 = $read2 + ".fq.gz"
44 #set out2 = "out2.gz"
45 #else if $input_2.is_of_type("fastq.bz2", "fastqsanger.bz2"):
46 #set read2 = $read2 + ".fq.bz2"
47 #set out2 = "out2.bz2"
48 #else if $input_2.is_of_type('fasta'):
49 #set read2 = $read2 + ".fa"
50 #set out2 = "out2.fa"
51 #else:
52 #set read2 = $read2 + ".fq"
53 #set out2 = "out2.fq"
54 #end if
55 ln -f -s '$input_2' '$read2' @
56 #end if
57
58 ## Run Cutadapt
59
60 cutadapt
61
62 # cutadapt (up to version 1.16) can't be run in multicore mode with these options
63 #if not any($output_options.info_file, $output_options.rest_file, $output_options.wildcard_file, $output_options.too_short_file, $output_options.too_long_file, $output_options.untrimmed_file)
64 # -j ${GALAXY_SLOTS:-1}
65 #end if
66
67 #if str($library.type) = "single":
68 @read1_options@
69 --output="$out1"
70 #else:
71 @read1_options@
72 @read2_options@
```

1/3rd of cutadapt.xml command

Handling Galaxy Tool Wrapper Requirements

Galaxy uses conda to handle tool requirements.

For best-practises pipelines, we need a single container image.

New feature added:

- `--build-galaxy-tool-images`
- Builds a new container image on the fly (during translation)
- Contains all software listed in tool `<requirements>`
- Image will appear as the container requirement in translations

```
<tool id="limma_voom" name="limma" version="@TOOL_VERSION@+galaxy0">
  <description>
    | Perform differential expression with limma-voom or limma-trend
  </description>

  <macros>
    | <token name="@TOOL_VERSION@">3.50.1</token>
  </macros>

  <requirements>
    <requirement type="package" version="@TOOL_VERSION@">bioconductor-limma</requirement>
    <requirement type="package" version="3.36.0">bioconductor-edger</requirement>
    <requirement type="package" version="1.4.36">r-statmod</requirement>
    <requirement type="package" version="1.1.1">r-scales</requirement>
    <requirement type="package" version="0.2.21">r-rjson</requirement>
    <requirement type="package" version="1.20.3">r-getopt</requirement>
    <requirement type="package" version="3.1.1">r-gplots</requirement>
    <requirement type="package" version="2.4.0">bioconductor-glimma</requirement>
  </requirements>
</tool>
```

Handling Galaxy Tool Wrapper Requirements

Some notes about this feature:

- Requires docker
- Can be slow (2-30 mins)

In the interest of time, we will not use this feature today.

- Images for relevant tools (limma-voom, hisat2, featurecounts) available on a quay.io repo
- We will replace the container requirement for affected tools

When this feature is turned off...

- janis-translate picks a suitable container based on the main software requirement.

```
process HISAT2 {  
  
    container "quay.io/biocontainers/hisat2:2.2.1--h87f3376_5"  
    publishDir "${params.outdir}/hisat2"  
  
    input:  
    path library_input_1  
    path index_path  
  
    output:  
    path "${library_input_1.simpleName}.alignment_summary.txt", emit: out_summary_file  
    path "${library_input_1.simpleName}.bam", emit: output_alignments  
  
    script:  
    """  
    hisat2 \  
    -U ${library_input_1} \  
    -x ${index_path[0].simpleName} \  
    --summary-file ${library_input_1.simpleName}.alignment_summary.txt \  
    -S out.sam  
    """  
}
```



Accessing Workflows

Workflows can be accessed 3 ways

- Via Shared Workflows
- Via User Workflows
- Via Link

Accessing Workflows

Workflows can be accessed 3 ways

- Via Shared Workflows
- Via User Workflows
- Via Link

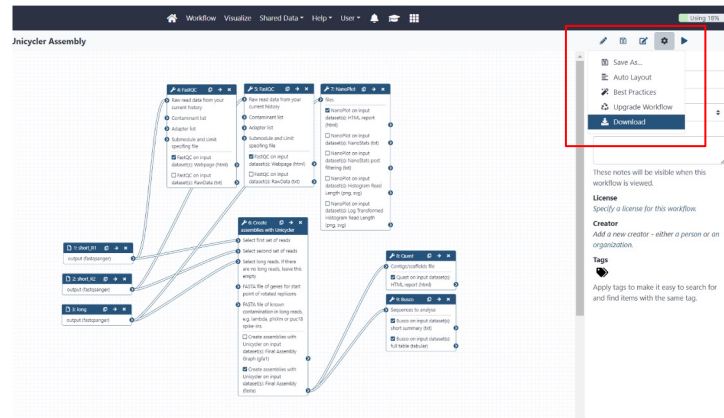
Published Workflows

search name, annotation, owner, or Advanced Search

Name	Annotation	Owner	Community Rating	Community Tags	Last Updated
Phylogenetic Tree Building		graceh1024	★★★★★		Apr 14, 2021
COVID-19: variator	Call variants from WGS (non-ampliconic) paired-end reads.	galaxy-australia	★★★★★	COVID-19 covid19.galaxyproject.org	Mar 03, 2021
COVID-19: variation analysis on WGS SE data	Call variants from WGS (non-ampliconic) single-end reads.	galaxy-australia	★★★★★	COVID-19 covid19.galaxyproject.org	Mar 03, 2021
COVID-19: variation analysis on ARTIC PE data	Call variants from ampliconic paired-end reads	galaxy-australia	★★★★★	COVID-19 covid19.galaxyproject.org	Mar 03, 2021
COVID-19: variation analysis of ARTIC ONT data	A Galaxy workflow that replaces the ARTIC minion shell command	galaxy-australia	★★★★★	ONT covid-19 covid19.galaxyproject.org	Mar 03, 2021
COVID-19: variation analysis reporting	Generate variant reports for the output of SARS-CoV-2 variation analysis workflows	galaxy-australia	★★★★★	COVID-19 covid19.galaxyproject.org	Mar 03, 2021
COVID-19: consensus construction	Build a consensus sequence from a list of variants. Hard-mask regions with low coverage and sites with called, but filtered variants. Note: Sites with...	galaxy-australia	★★★★★	COVID-19 covid19.galaxyproject.org	Mar 03, 2021
SARS-CoV-2: downsample ONT reads assigned to transients	Downsamples ONT reads assigned to regions and then runs the sub workflows: SARS-CoV-2: coverage	galaxy-australia	★★★★★	ONT covid-19 covid19.galaxyproject.org	Mar 03, 2021

Shared Workflows

User Workflows



Via Link



Workflows

These workflows are associated with 3: RNA-seq genes to pathways

To use these workflows in Galaxy you can either click the links to download the workflows, or you can right-click and copy the link to the workflow which can be used in the Galaxy form to import workflows.

Workflow	Updated	Import	Has Tests	Tested	License	Creators
ma-seq-genes-to-pathways.ga	Feb 25, 2022	Launch in Tutorial Mode ? Import to UseGalaxy.eu Import to UseGalaxy.org Import to another server (≥23.8+ only!)	✗	✗	None Specified, defaults to CC-BY-4.0	

Importing into Galaxy

Below are the instructions for importing these workflows directly into your Galaxy server of choice to start using them!

Accessing Workflows - Workflow Format

Galaxy Workflows can be downloaded as .ga files.

These are json files which describe the metadata a workflow needs to run on a Galaxy server.

- Workflow metadata
- Step metadata
- Step input parameters
- Step tool info (tool name, revision etc)

Accessing Workflows - Workflow Format

Galaxy Workflows can be downloaded as .ga files.

These are json files which describe the metadata a workflow needs to run on a Galaxy server.

- Workflow metadata
- Step metadata
- Step input parameters
- Step tool info (tool name, revision etc)

```
"7": {
  "input_connections": {
    "inputFile": {
      "id": 5,
      "output_name": "output_alignments"
    }
  },
  "name": "MarkDuplicates",
  "outputs": [
    {
      "name": "metrics_file",
      "type": "txt"
    },
    {
      "name": "outFile",
      "type": "bam"
    }
  ],
  "tool_id": "toolshed.g2.bx.psu.edu/repos/devteam/picard/picard_MarkDuplicates/2.18.2.1",
  "tool_shed_repository": {
    "changeset_revision": "f6ced08779c4",
    "name": "picard",
    "owner": "devteam",
    "tool_shed": "toolshed.g2.bx.psu.edu"
  },
  "tool_state": "{\"assume_sorted\": \"true\", \"barcode_tag\": \"\", \"comments\": [], \"duplicate_scoring_strategy\": \"SUM_OF_BASE_QUALITIES\", \"inputFile\": {\"_class\": \"ConnectedValue\", \"optical_duplicate_pixel_distance\": \"100\", \"read_name_regex\": \"\", \"remove_duplicates\": \"false\", \"validation_stringency\": \"LENIENT\", \"_page\": null, \"_rerun_remap_job_id\": null}}",
  "tool_version": "2.18.2.1",
  "type": "tool",
  "uuid": "d68e12ea-a68a-49b0-a4cd-736e4dbd2178",
```

For Today

Will translate 2 Galaxy Tool Wrappers to Nextflow

- https://usegalaxy.org.au/root?tool_id=toolshed.g2.bx.psu.edu/repos/devteam/samtools_flagstat/samtools_flagstat/2.0.4
- https://usegalaxy.org.au/root?tool_id=toolshed.g2.bx.psu.edu/repos/iuc/limma_voom/limma_voom/3.50.1+galaxy0

Will translate 1 Galaxy Workflow to Nextflow

- <https://training.galaxyproject.org/training-material/topics/transcriptomics/tutorials/rna-seq-reads-to-counts/workflows/rna-seq-reads-to-counts.ga>

Let's Begin