UNIVERSITY of the
WESTERN CAPE

SANBI
South African National
Bioinformatics Institute

High Throughput Computing in bioinformatics

workflows, containers and emerging paradigms

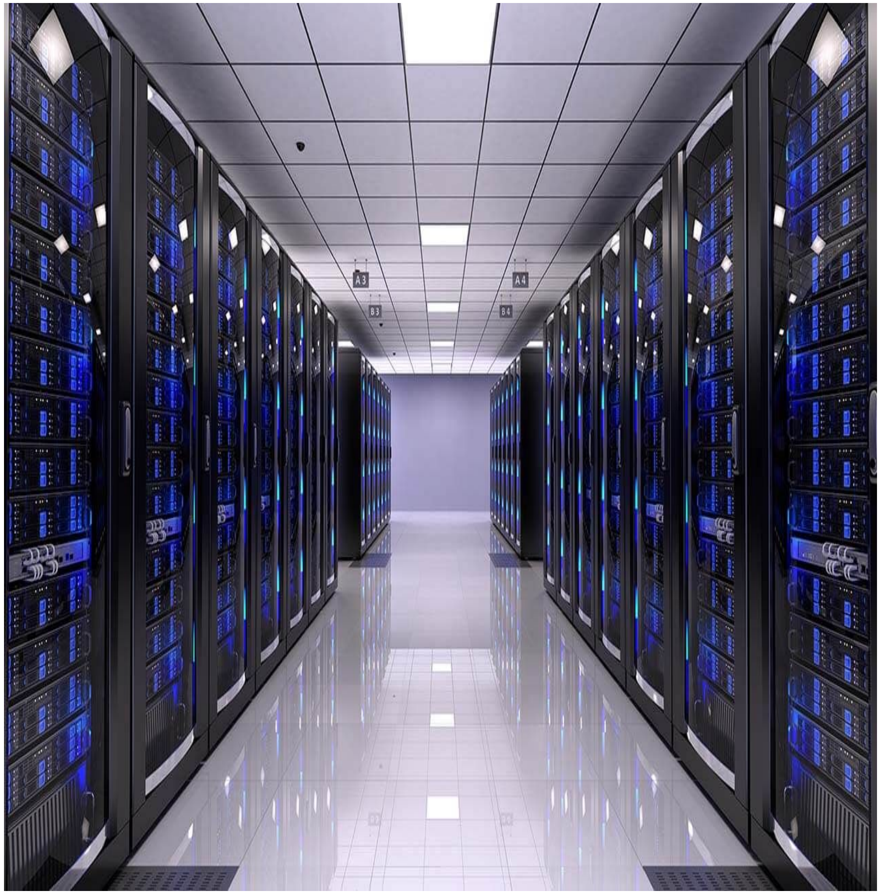
Peter van Heusden pvh@sanbi.ac.za
South African National Bioinformatics Institute

GETTING THE MOST OUT OF COMPUTING (FOR RESEARCH)

Scaling up computing

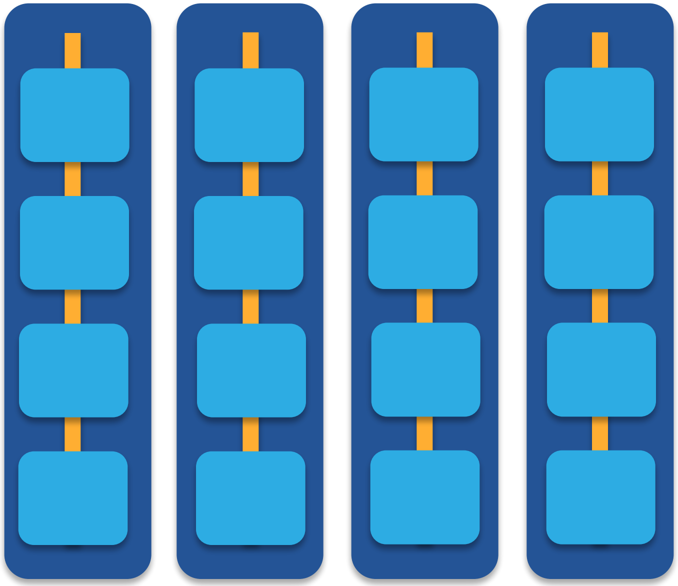
From

TO

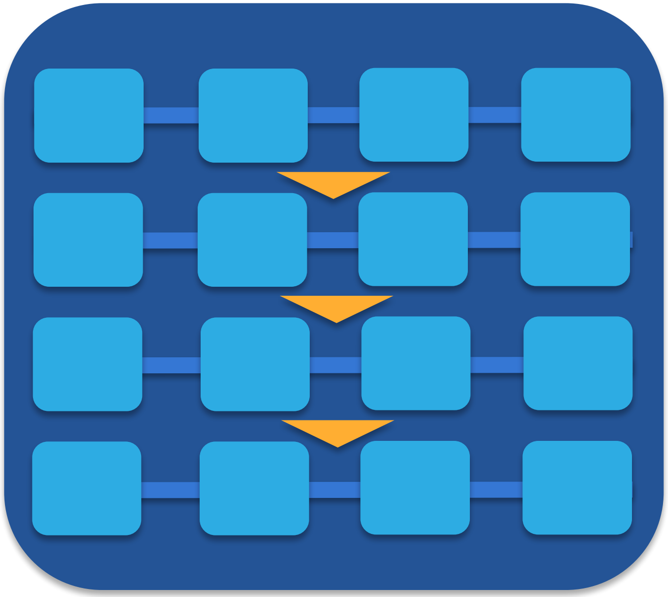


Computing types

- Our challenge: how to make use of computers working together to tackle large compute tasks...



high-throughput



high-performance (e.g.MPI)

Two Strategies

Cloud

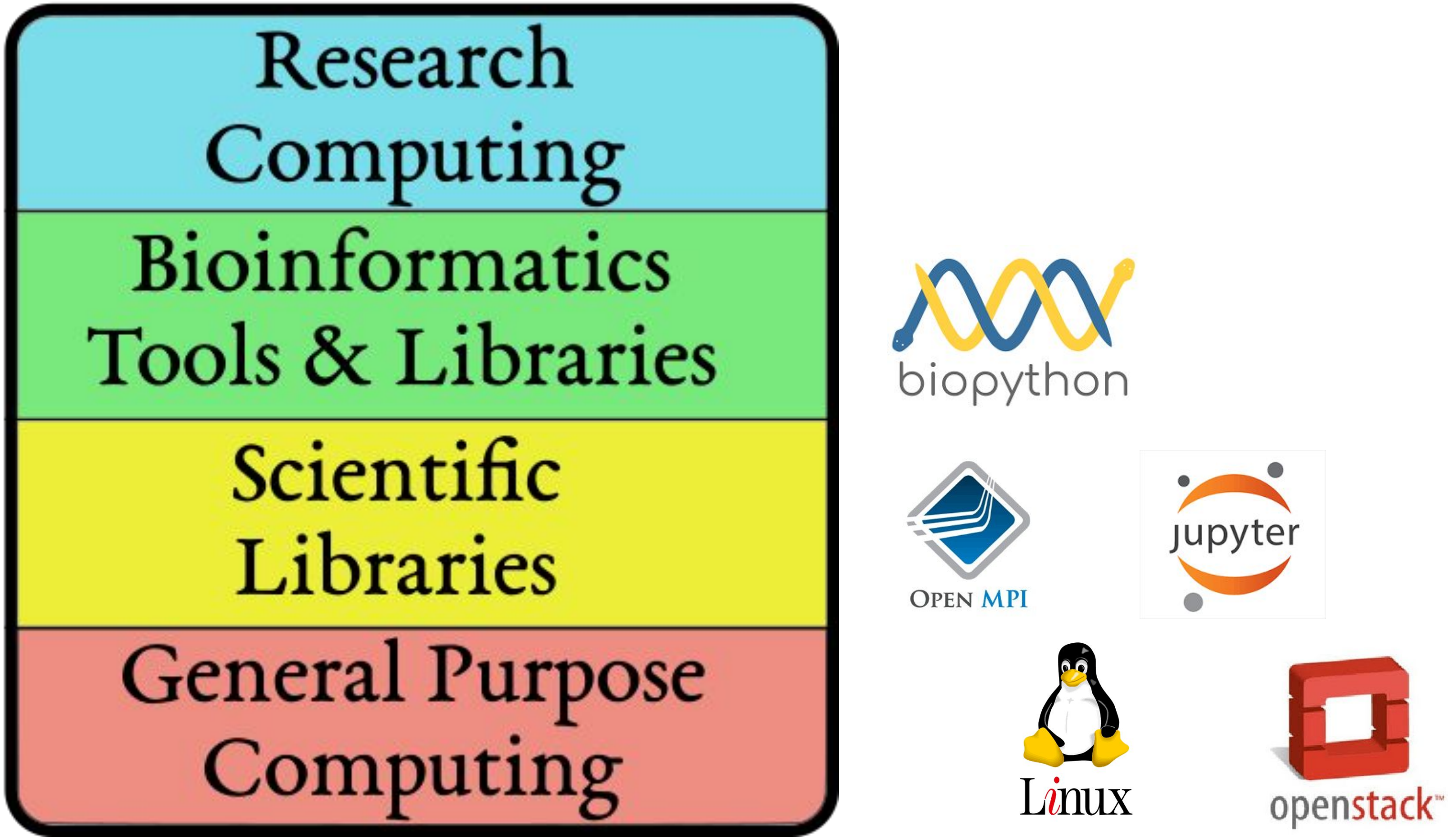
Focus: Service *many user groups* by providing *generic* computing

Skills focus: *systems* engineering to create *virtual infrastructure*, compose *multiple component* services

HPC Cluster

- Focus: Service *specialised* computing groups working on *computationally challenging* problems
- Skills focus: *research software* engineering and *parallel algorithms*

The Research Computing Stack



Research Computing

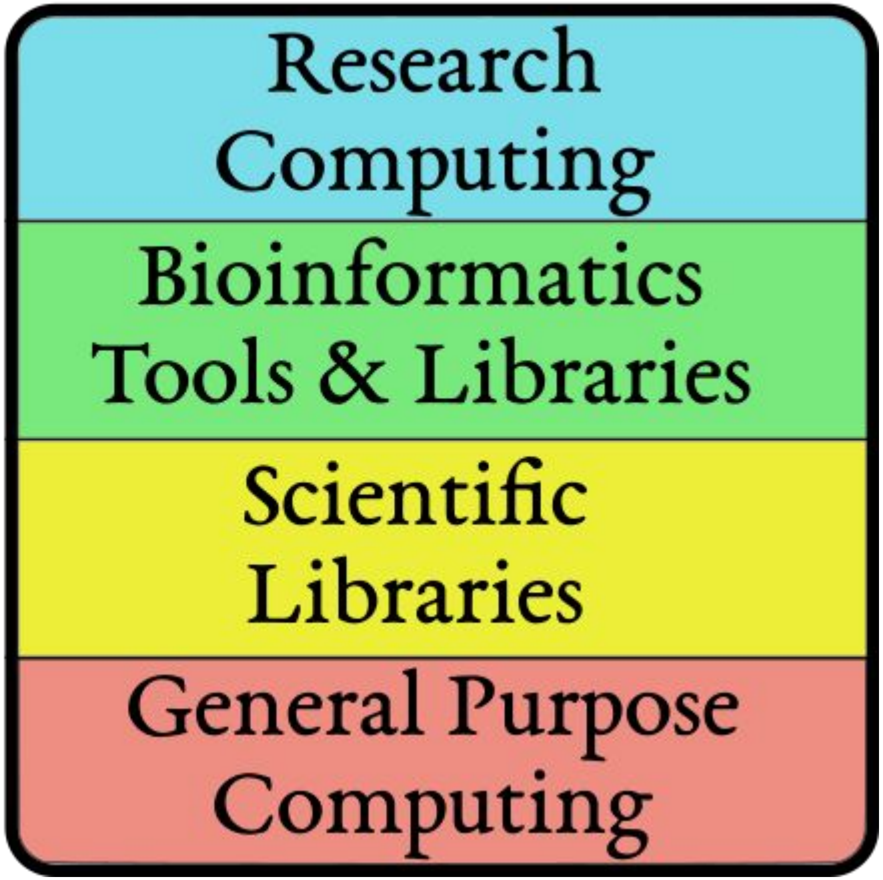
Bioinformatics Tools & Libraries

Scientific Libraries

General Purpose Computing

Stack idea taken from [Konrad Hinsen's blog]

Research Computing Roles

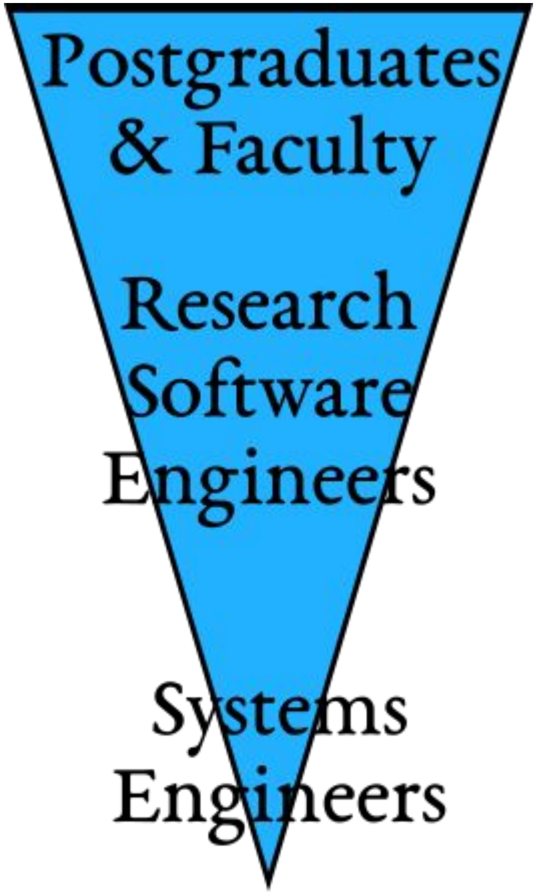


Research Computing

Bioinformatics Tools & Libraries

Scientific Libraries

General Purpose Computing



Postgraduates & Faculty

Research Software Engineers

Systems Engineers

Two Architectures

High Throughput

Focus: Workflows with many *small*, *largely independent* compute tasks

Optimize: *throughput*, or

time from *submission* to *overall completion*

High Performance

- Focus: Workflows with *large, highly coupled* tasks
- Optimize: *individual tasks*, software, communication between processes

Making Good Choices

- How do you choose the best approach?
- Guiding question:

Is your problem “HTC-able”?

Typical HTC Problems

- batches of similar program runs (>10)
- “loops” over independent tasks
- others you might not think of …
  - programs/functions that
    - process files that are already separate
    - process columns or rows, separately
    - iterate over a parameter space
  - *a lot* of programs/functions that use multiple CPUs on the same server

Ultimately: Can you break it up?

What is not HTC?

- fewer numbers of jobs
- jobs individually requiring significant resources
  - RAM, Data/Disk, # CPUs, time

  (though, “significant” depends on the HTC compute system you use)
- restrictive licensing

HTC Bioinformatics

Steps to HTC:

Automate on each layer

- Software defined infrastructure
- Software dependency management
- Workflows
- Reference data collections
- End user usability

HTC on a Cluster (1)

- Software provisioning:
  - Conda / bioconda: consistent installation
    - Environment modules - lacks naming consistency
  - Containerisation:
    - Singularity (biocontainers)

HTC on a Cluster (1)

- Software provisioning:
  - Conda / bioconda: consistent installation
    - Environment modules - lacks naming consistency
  - Containerisation:
    - Singularity (biocontainers)

- Workflows
  - “Custom scripts” limit re-use and waste effort
  - Many command line options:
    - Nextflow
    - Snakemake
    - Common Workflow Language (CWL) with Toil
  - Galaxy front-end, cluster back-end
    - Works well, **not** tuned for multi-tenant clusters
    - No shared filesystem? Galaxy-Pulsar

HTC on a Cluster (2)

- Data provisioning:
  - Mostly an unsolved problem
  - Galaxy solution: reference data via CVMFS

HTC on a Cluster (2)

- Data provisioning:
  - Mostly an unsolved problem
  - Galaxy solution: reference data via CVMFS

- End-user usability
  - Commandline high cost of entry for users
  - Training is largely specific to single cluster
    - Trying to address that with “modular” training materials at HPC Carpentry (Thursday)

HTC on a Cluster: Demo

- Demonstrated at CHPC Conf 2016
  - CommonWL with cwltool - SANBI / CHPC
- H3ABionet Hackathon (Aug 2016)
  - 4 workflows in 1 week
  - Nextflow & CWL
  - http://bit.ly/h3a_wf_scidataconf18
  - http://bit.ly/h3ahack_paper

HTC on a Cloud (1)

- Infrastructure provisioning
  - Software defined infrastructure
  - System software deployed via images
  - System configuration

Terraform

HashiCorp Packer

ANSIBLE

HTC on a Cloud (1)

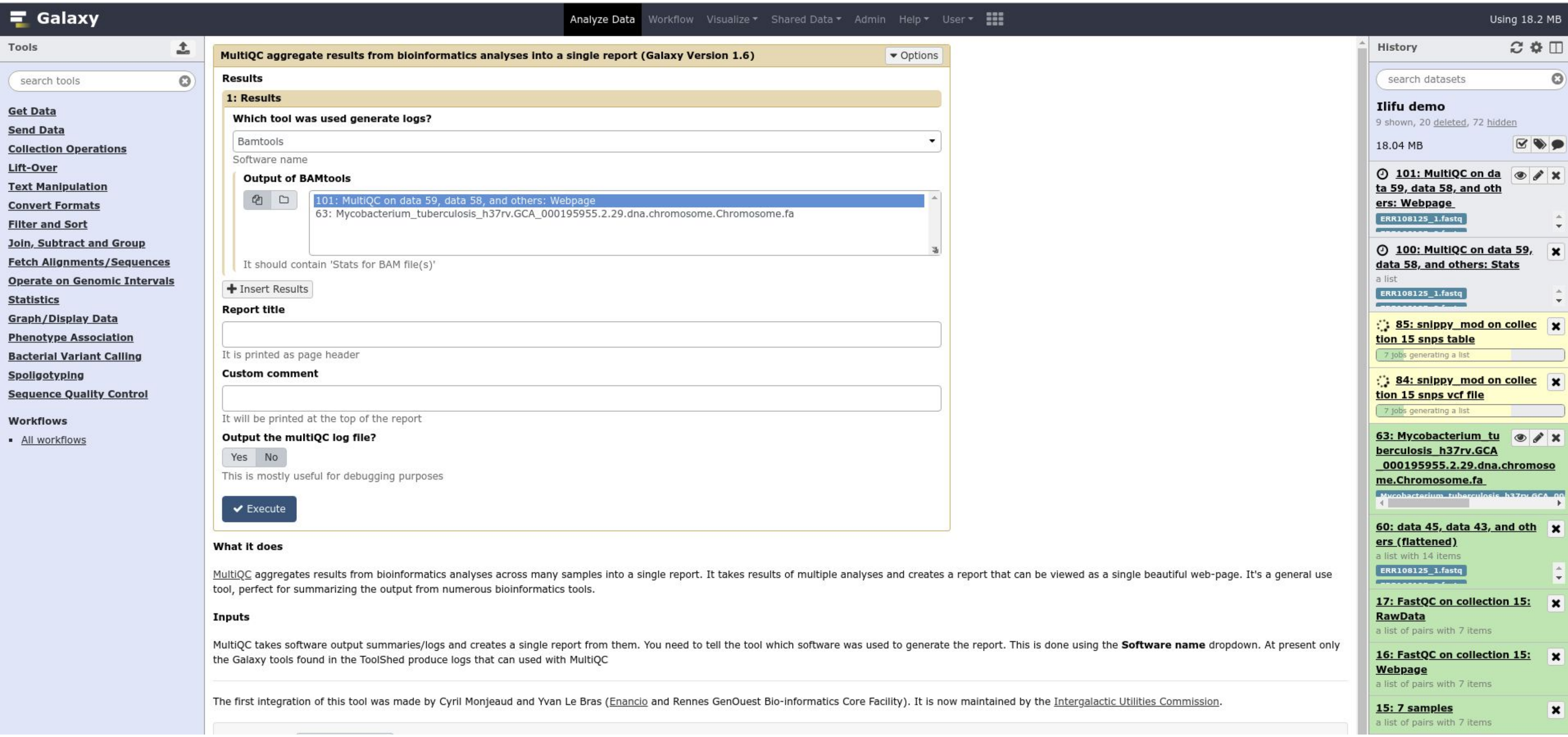
- Infrastructure provisioning
  - Software defined infrastructure
  - System software deployed via images
  - System configuration
- Software provisioning
  - Conda / bioconda
  - Containerisation
- Workflows
  - “Virtual cluster”
  - Galaxy front-end, cluster back-end (HTCondor)
    - experimental: Kubernetes back-end

HTC on a Cloud (2)

- Data provisioning
    - Still a mostly unsolved problem
    - Use S3 semantics: objects + metadata
    - Inherently multi-tenant: move analysis to data

- End-user usability
    - Provide a higher level of abstraction
    - Build on consistent training materials / community

HTC on a Cloud (Ilifu) Demo



MultiQC aggregate results from bioinformatics analyses into a single report (Galaxy Version 1.6)

Results

1: Results

Which tool was used generate logs?

Bamtools

Software name

Output of BAMtools

101: MultiQC on data 59, data 58, and others: Webpage
63: Mycobacterium_tuberculosis_h37rv.GCA_000195955.2.29.dna.chromosome.Chromosome.fa

It should contain 'Stats for BAM file(s)'

Report title

It is printed as page header

Custom comment

It will be printed at the top of the report

Output the multiQC log file?

Yes No

This is mostly useful for debugging purposes

What it does

MultiQC aggregates results from bioinformatics analyses across many samples into a single report. It takes results of multiple analyses and creates a report that can be viewed as a single beautiful web-page. It's a general use tool, perfect for summarizing the output from numerous bioinformatics tools.

Inputs

MultiQC takes software output summaries/logs and creates a single report from them. You need to tell the tool which software was used to generate the report. This is done using the **Software name** dropdown. At present only the Galaxy tools found in the ToolShed produce logs that can used with MultiQC

The first integration of this tool was made by Cyril Monjeaud and Yvan Le Bras (Enancio and Rennes GenOuest Bio-informatics Core Facility). It is now maintained by the Intergalactic Utilities Commission.

Ilifu demo

9 shown, 20 deleted, 72 hidden

18.04 MB

101: MultiQC on data 59, data 58, and others: Webpage
ERR108125_1.fastq

100: MultiQC on data 59, data 58, and others: Stats
a list
ERR108125_1.fastq

85: snippy_mod on collection 15 snps table
7 jobs generating a list

84: snippy_mod on collection 15 snps vcf file
7 jobs generating a list

63: Mycobacterium_tuberculosis_h37rv.GCA_000195955.2.29.dna.chromosome.Chromosome.fa

60: data 45, data 43, and others (flattened)
a list with 14 items
ERR108125_1.fastq

17: FastQC on collection 15: RawData
a list of pairs with 7 items

16: FastQC on collection 15: Webpage
a list of pairs with 7 items

15: 7 samples
a list of pairs with 7 items

The Future

- Convergence of technologies for:
  - Software deployment
  - Data management and deployment
  - Workflows

The Future

- Convergence of technologies for:
  - Software deployment
  - Data management and deployment
  - Workflows
- Challenges of convergence
  - Who will engineer cross-institutional projects?
  - Where and how will users learn / be trained?

Thanks

- SANBI: Prof Alan Christoffels & our IT office
    - University of the Western Cape
    - National Research Foundation
    - Medical Research Council
- H3ABionet: workflow hackathon
- Ilifu & IDIA (our astronomy fellow-travellers)
- The Galaxy, Carpentries & open source bioinformatics communities