ilifu Online Training

Jeremy Smith

User Training Workshop – Introductory Training

29 March 2022

Topics

- Introduction to the ilifu research facility services
- Directory structure
- Software environment
  - Singularity containers
  - Modules
- Using JupyterHub
- Introduction to Slurm

Getting help

- Support contact

  support@ilifu.ac.za

- User documentation

  http://docs.ilifu.ac.za/#/

- Ilifu System Status

  https://status.ilifu.ac.za/

- Training videos

  https://www.ilifu.ac.za/latest-training/

High Performance Computing

Combining power of distributed computers

- Collection of servers (computers)
- Connected by fast local network

Some terminology

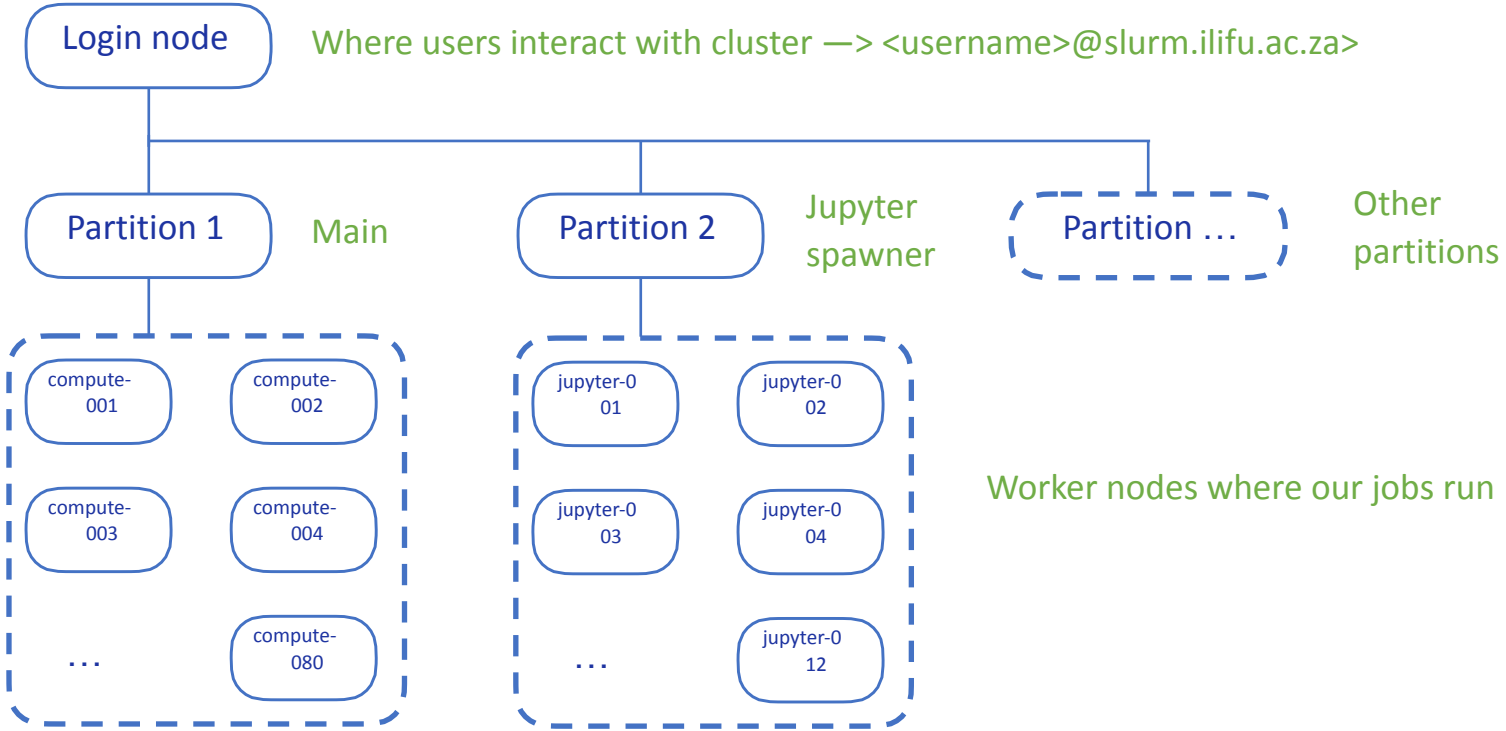
- Servers also referred to as nodes
- Group of nodes is a cluster

ilifu Research Facility

Cloud-based infrastructure for data-intensive research

- Support variety of different scientific projects and requirements
- Flexible compute environment
- Data management: storage, transfer

ilifu Research Facility



Software we use

- Job Scheduler to manage resources - Slurm

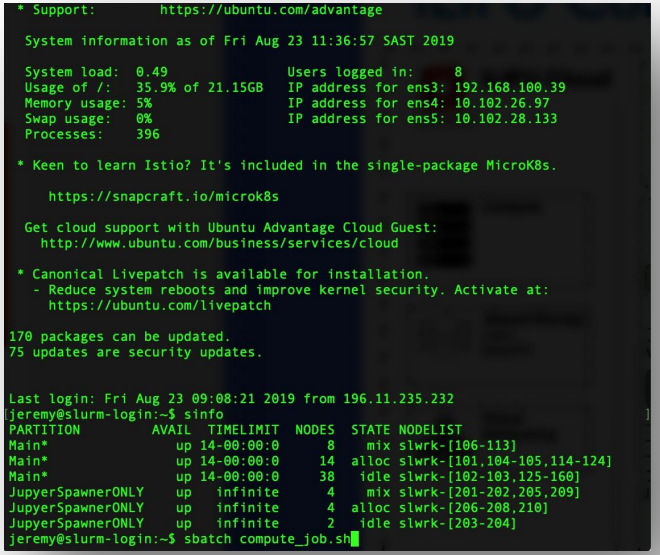
- JupyterHub service - development environment

- Containerised software environment - Singularity

- Other services: data transfer, CARTA

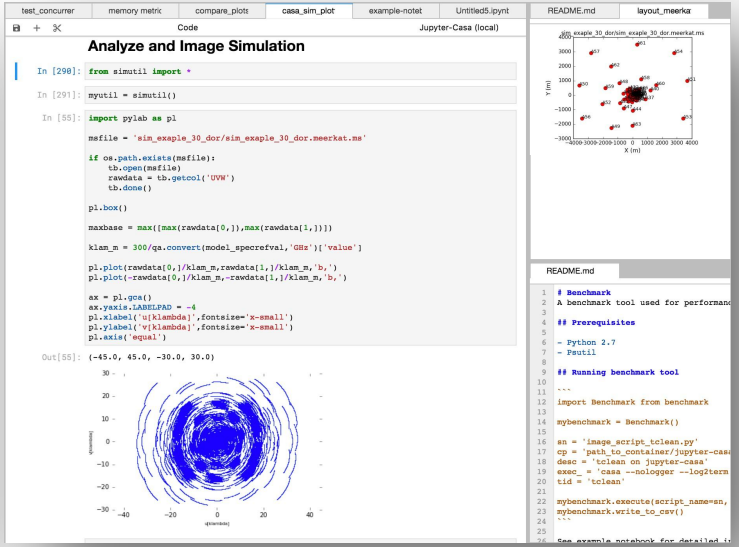
Computing environment - interface

ssh - shell terminal



`ssh <username>@slurm.ilifu.ac.za`

JupyterHub



https://jupyter.ilifu.ac.za

Computing environment - ssh

Your SSH key

- Used in the SSH (Secure Shell) protocol
- Authentication method for gaining access to encrypted connecting between systems
- Use connection to manage system remotely
- We need your SSH public key so our system know to let you in

Compute environment - ssh

Generating SSH key

- If you don’t already have one
- New compute/formatted existing computer

GitHub docs on key generation:

https://docs.github.com/en/github/authenticating-to-github/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent

Directory Structure

Common areas:

- /users
  - only 50 TB shared among all users, for scripts and small files – don't place data here, capping /users storage capacity can prevent access to the cluster for all users.
- /scratch3/users
  - directory space for processing data, temporary storage only, i.e. use this space during processing, and then clear all files immediately after processing. Remove unnecessary data and move data that you want to keep to project folder.

Remaining storage separated by group: IDIA, CBio, ilifu

Directory Structure

IDIA structure:

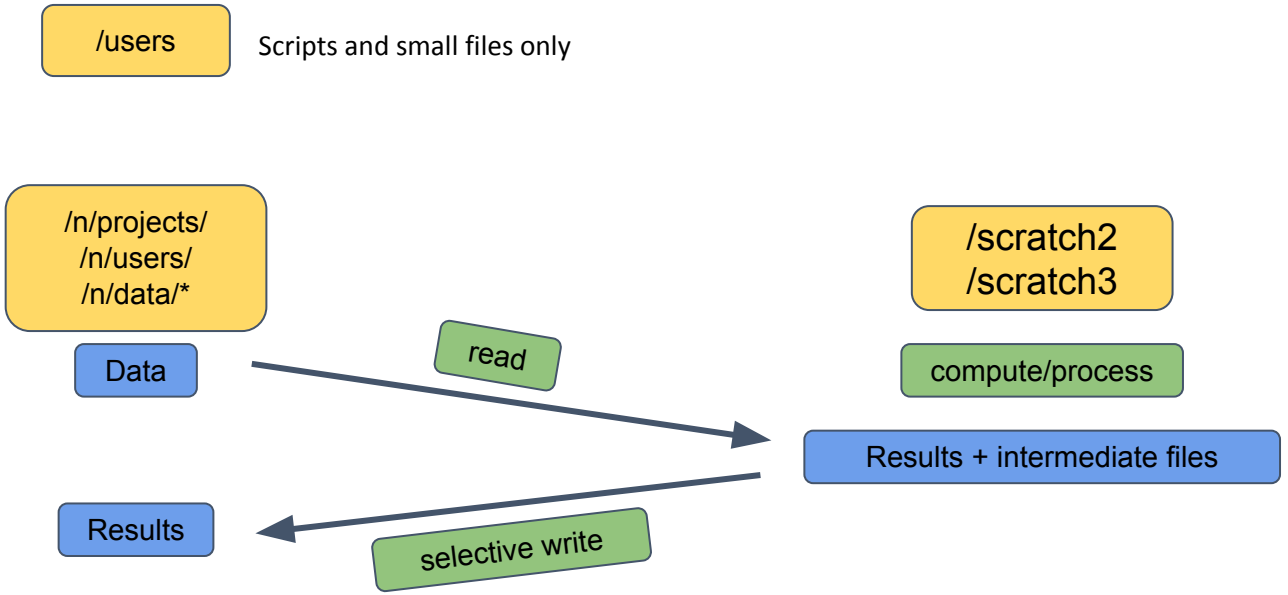
- /idia/users
  - user's private work directory, may store data products that are not ready to move to shared project space
- /idia/projects
  - project specific directories. These directories are for sharing data and resources within project groups. Raw data associated with a project will also be available from the project folder. Raw data should always be read-only.
- /idia/software
  - software containers and the IDIA Pipelines software is stored here

Directory Structure

Similar structure for /cbio and /ilifu groups:

- /cbio/users
- /cbio/projects
- /cbio/soft
- /ilifu/users
- /ilifu/software
- Exception for ilifu projects:
  - /ilifu/astro/projects
  - /ilifu/bio/projects

Directory Structure - Typical workflow



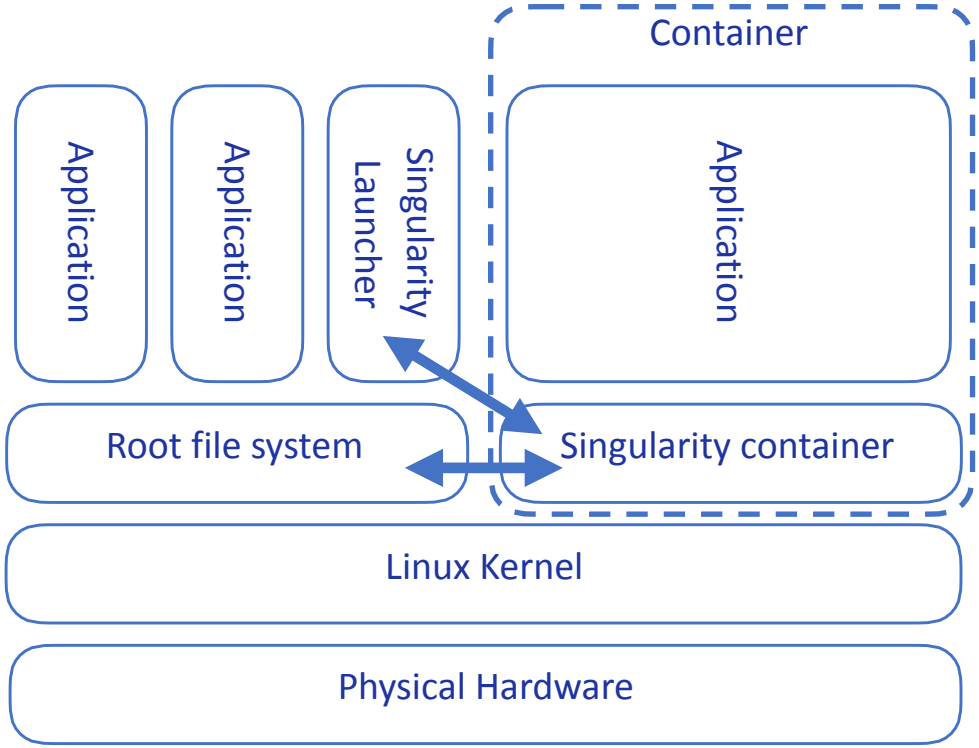
*/n/data generally read-only

Software environment - Singularity containers

- Encapsulated software environments
- A software stack that contains everything required to run an application/workflow, including files, environments variables, libraries and dependencies
- Containers accessible across platforms and services, allowing sharing of applications environments

https://sylabs.io/singularity

Software environment - Singularity containers



Software environment - Singularity containers

Supported Containers:

- CASA
- Astronomy container (ASTRO-PY3)
- KERN suite
- Python 2.7, Python 3.6, R
- GPU container
- Project containers:
  - MeerLICHT, LADUMA, HI Intensity mapping

Directories:

- /software
- /idia/software/containers
- /ilifu/software/containers

Software environment - Singularity containers

Open container as an interactive shell:

```
singularity shell /path/to/container

Example:

$ singularity shell /idia/software/containers/ASTRO-PY3.simg
```

Run a script/workflow using a container environment:

```
singularity exec /path/to/container <software> <script/input_parameters>

$ singularity exec /idia/software/containers/casa-stable.img casa -c myscript.py
```

Software environment - modules

module avail

```
$ module avail

------------------------------------------------------------------ /software/modules/common -------------------------------------------------------------------
   LAPACK/3.9.0                anaconda3/2020.07             githubcli/2.0.0            mono/6.8.0.123        perlbrew/perlbrew    python/3.10.0
   R/RStudio1.2.5042-R4.0.0    anaconda3/2021.05             go/1.16.3                  mpich/3.3a2           python/2.7.18        python/3.10.1     (D)
   R/RStudio1.2.5042-R4.0.4    anaconda3/2021.11      (D)    go/1.17.3           (D)    openBLAS/0.3.9        python/3.6.15        ruby/2.6.6
   R/3.6.3                     cuda/10.0.130_410.48          graphviz/2.49.1            openmpi/2.1.1         python/3.7.7         singularity/2.6.1
   R/4.0.0                     cuda/10.1.243_418.87.00       homebrew/2.4.13            openmpi/2.1.6         python/3.8.2         singularity/3.7.3
   R/4.0.2                     cuda/10.2.89_440.33.01        hwloc/1.11.13              openmpi/3.1.6         python/3.8.3         singularity/3.8.3
   R/4.0.3                     cuda/11.0.2_450.51.05         java/jre-1.8.0_261         openmpi/4.0.3         python/3.8.6         singularity/3.9.0
   R/4.1.1              (D)    cuda/11.4.2_470.57.02  (D)    java/openjdk-14.0.1 (D)    openmpi/4.0.5         python/3.9.0         singularity/3.9.1 (L,D)
   anaconda3/login.old         dotnet/5.0.301                julia/1.5.3                openmpi/4.1.0  (D)    python/3.9.4         user_tools
   anaconda3/login             drmaa/1.1.1                   maven/3.6.3                perl/5.33.0           python/3.9.7

------------------------------------------------------------------ /software/modules/astro --------------------------------------------------------------------
   casa/5.7.0      casa/5.8.0                  casa/6.1.2.7-pipeline    casa/6.2    casa/6.4     (D)
   casa/5.7.2-4    casa/6.1.0-118-monolithic   casa/6.1.2.7-modular     casa/6.3    pybdsf/1.9.2

------------------------------------------------------------------- /software/modules/bio ---------------------------------------------------------------------
   bcbio/bcbio_container       biobambam2/2.0.183    genomestrip/2.00.1958    plink/2.00a2.3     samtools/1.13          vep/singularity
   bcbio/1.2.3                 canvas/1.40.0.1613    htslib/1.10.2            popgen/0.1         samtools/1.14   (D)    vep/101.0       (D)
   bcbio/1.2.9          (D)    cd-hit/4.8.2          mafft/7.490              prsice-2/2.3.1d    treePL/homebrew
   bcftools/1.10.2             gemini/gemini         mash/2.3                 samtools/1.10      vcftools/0.1.16

------------------------------------------------------------- /usr/share/lmod/lmod/modulefiles --------------------------------------------------------------
   Core/lmod/6.6    Core/settarg/6.6

  Where:
   L:  Module is loaded
   D:  Default Module
```

Software environment - modules

- module avail
- module help <module>

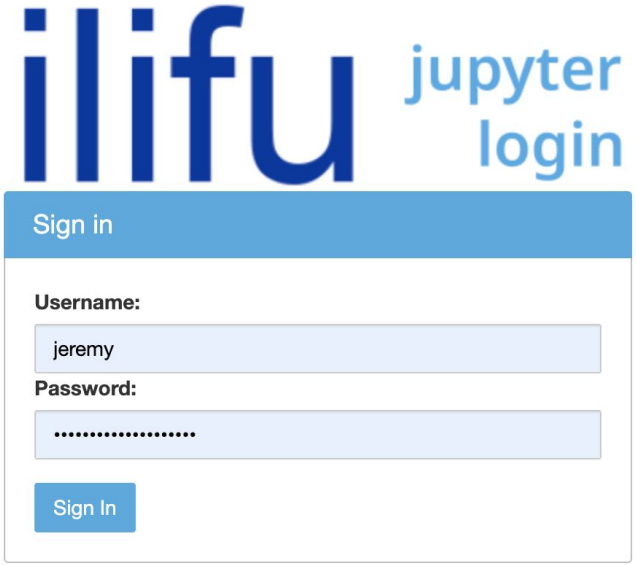
```
$ module help python

------------------------------ Module Specific Help for "python/3.10.1" ---------------------------------
This module configures Python 3.10.1 for use
```

- module load <module>
- module list
- module purge
- module --help

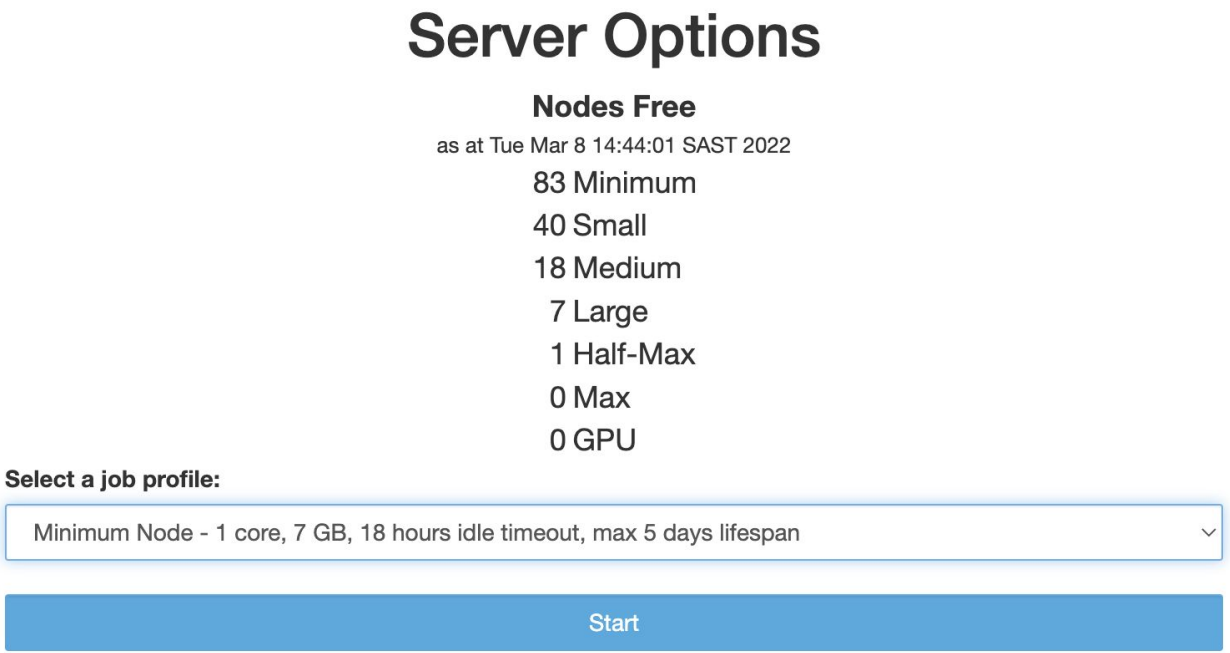
JupyterHub

https://jupyter.ilifu.ac.za



JupyterHub

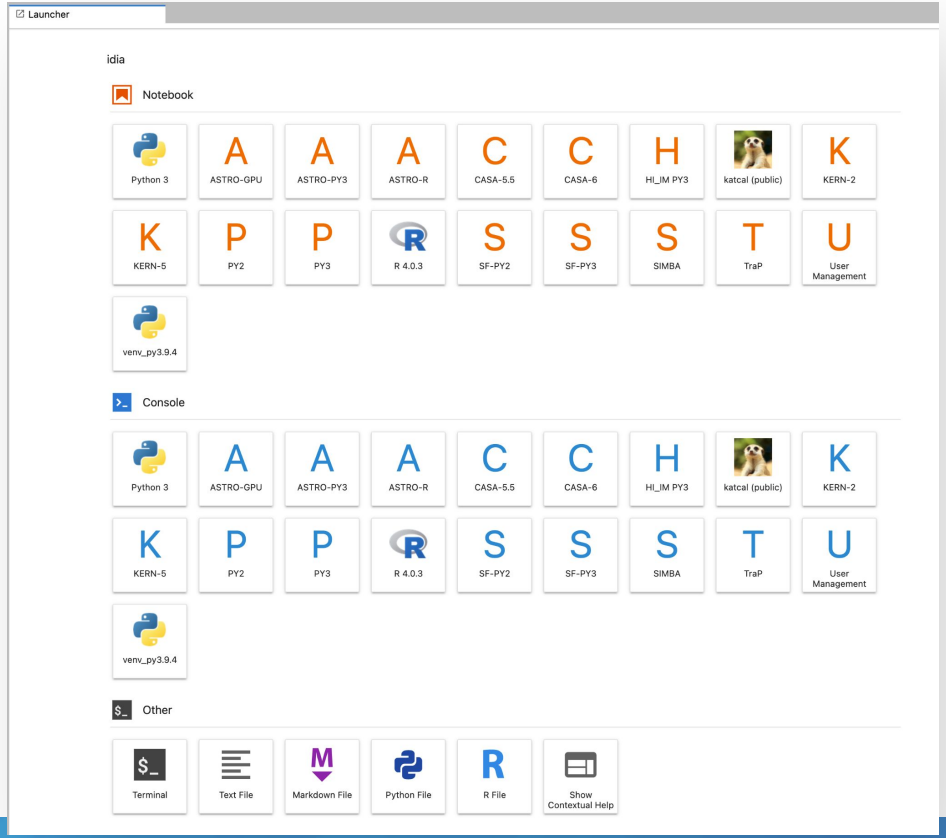
Session size



JupyterHub

Choose kernel

in launcher



Demo time