Introduction to HPC and Linux

For deep learning practitioner

Topics to be covered

- Access Massive
- Basic Linux
- Slurm
- software module
- Job usage monitoring
- Visualization
- Transfer data
- Typical deep learning project workflow

Connect to cluster

Login

- You need to setup your password at karaage
- You will be asked for password for the first time

Secure shell (ssh)

replace this with your username, you can find it at
https://hpc.erc.monash.edu.au/karaage/profile/personal/

ssh luhanc@m3.massive.org.au

The hostname/IP address, it can be either:

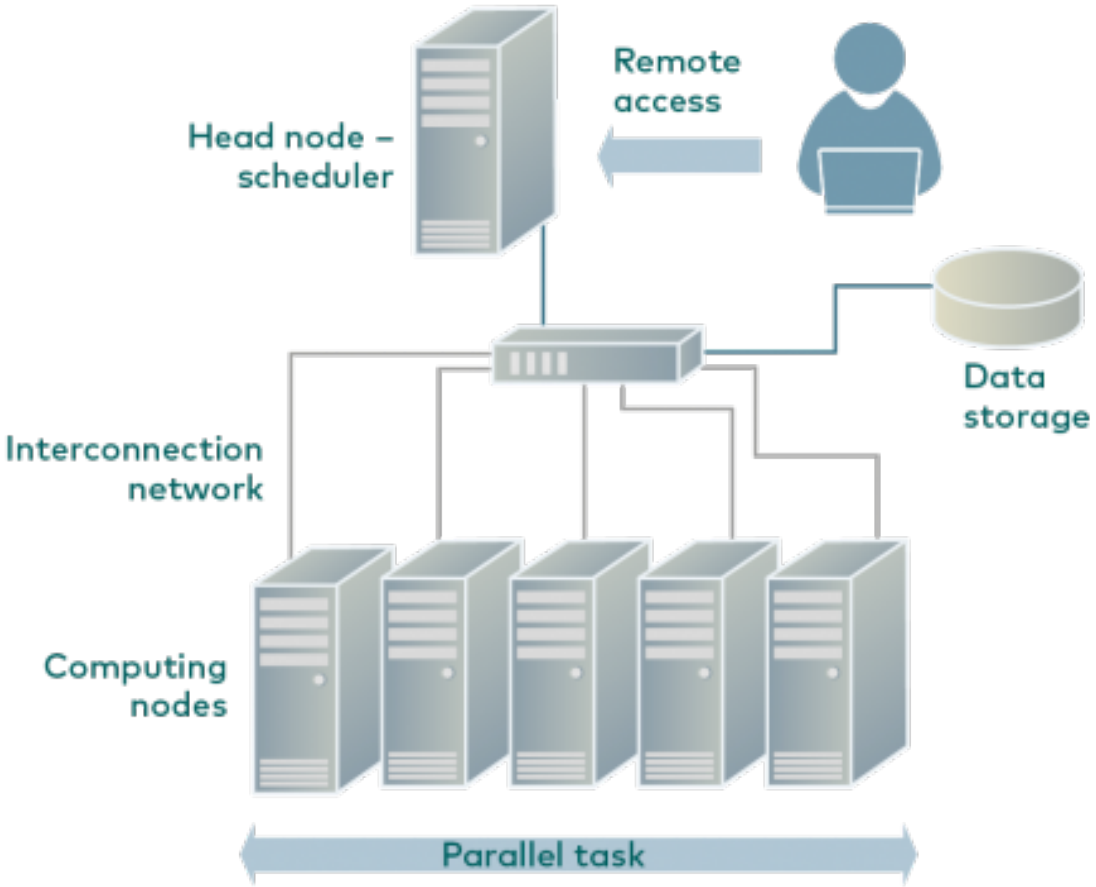
- **m3.massive.org.au**
- **m3-login1.massive.org.au**
- **m3-login2.massive.org.au**
- **m3-dtn.massive.org.au**

Setup quick access (optional)

- While you are waiting, please have a look at the setup https://docs.github.com/en/github/authenticating-to-github/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent
- Successful setup should enable you to login by simply type ‘ssh massive’

Overview of cluster

High-performance Cluster



Basic Linux commands

Please open the Linux cheat sheet and follow the activity
Feel free to shout out any question

The following command copy training resource to your home directory

```
/projects/vf38/introHPC/
```

Data transfer

```
scp <files> <username>@<hostname>:<target location>
```

```
scp *.py luhanc@m3.massive.org.au:/home/luhanc/workshop/
```

Script

- Scripting is basically executing a list of commands line by line
- For example

```
#!/bin/bash
echo current date is `date`
echo my hostname is `hostname`
```

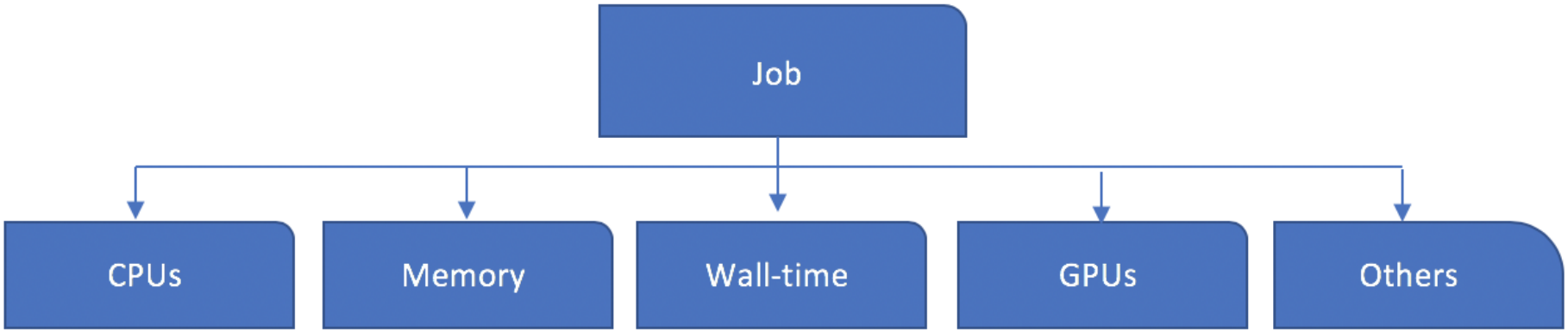
Work remotely (optional)

```
sshfs -o allow_other,defer_permissions,follow_symlinks <username>@<hostname>:<remote
direcotry> <local directory>
```

```
sshfs -o
allow_other,defer_permissions,follow_symlinks,auto_cache,reconnect,defer_permissions,noappledo
uble,nolocalcaches,no_readahead luhanc@m3:/home/luhanc /Users/ChengLuhan/mount-remote/m3/home
```

Job scheduling

Component of Job



Example slurm script

Options available at https://slurm.schedmd.com/sbatch.html

```
#!/bin/bash
#SBATCH --time=00:00:10
#SBATCH --mem=1MB
#SBATCH --cpus-per-task=1
#SBATCH --ntasks=1
echo ‘echoing ...’
echo ‘current date is ’
date
echo ‘my hostname is ’
hostname
```


Resource is specified by #SBATCH header

Tips

- Running `show_cluster` print the status of machine
- You can check your job through `show_job`
- If your job takes forever to schedule, check if you misspecified some config (e.g. time is 1 day by default)
- specify mail-user/mail-type can often be quite useful

Software Modules

Module Commands

Command	**Description**
`module avail`	Show available modules
`module list`	List loaded modules
`module load` ***`modulename`***	Load a module into the current environment
`module rm|unload` ***`modulename`***	Unload a module from the environment
`module purge`	Unload all loaded modules
`module swap` ***`module1 module2`***	Swap a loaded module with another
`module show` ***`modulename`***	Give help for a particular module
`module help`	Show module specific help

Software Installation

- Existing packages may not meet your requirement
  - Option 1: Email help@massive.org.au
  - Option 2: Install required software by yourself
    - Issue: you cannot run `sudo` on cluster
    - solution: python virtual environment

Python virtual environment

- Two dominate approach
  - conda: You gain lots of speed advantage, but generally harder to manage
  - pip: Simple, widely supported

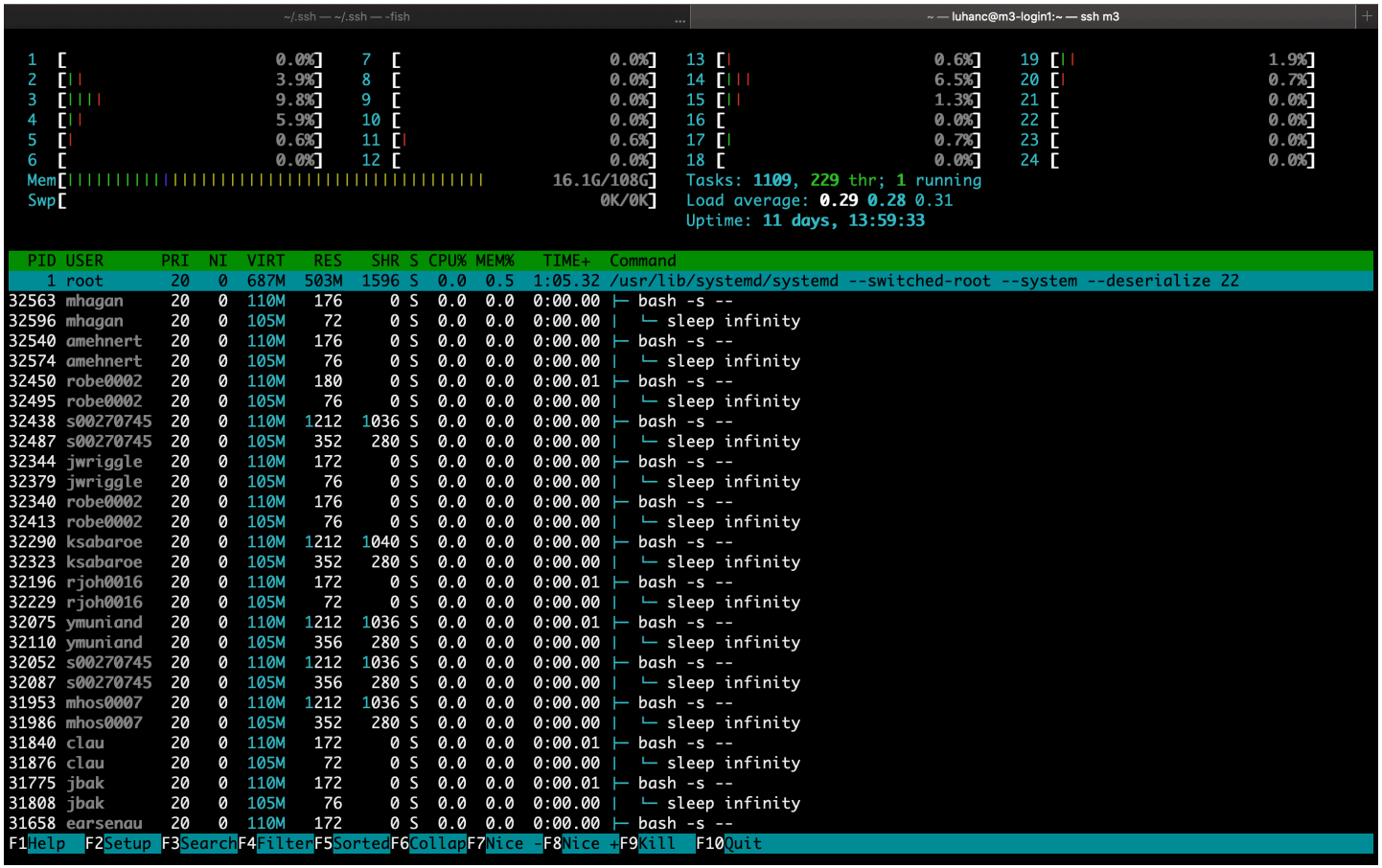
Python virtual environment

- Create venv
  - python3 -m venv <path to env>
- Enter environment
  - source activate <path to env>/bin/activate
- Install packages
  - pip install <package name>
- Exit environment
  - deactivate

Monitor Job Usage

htop

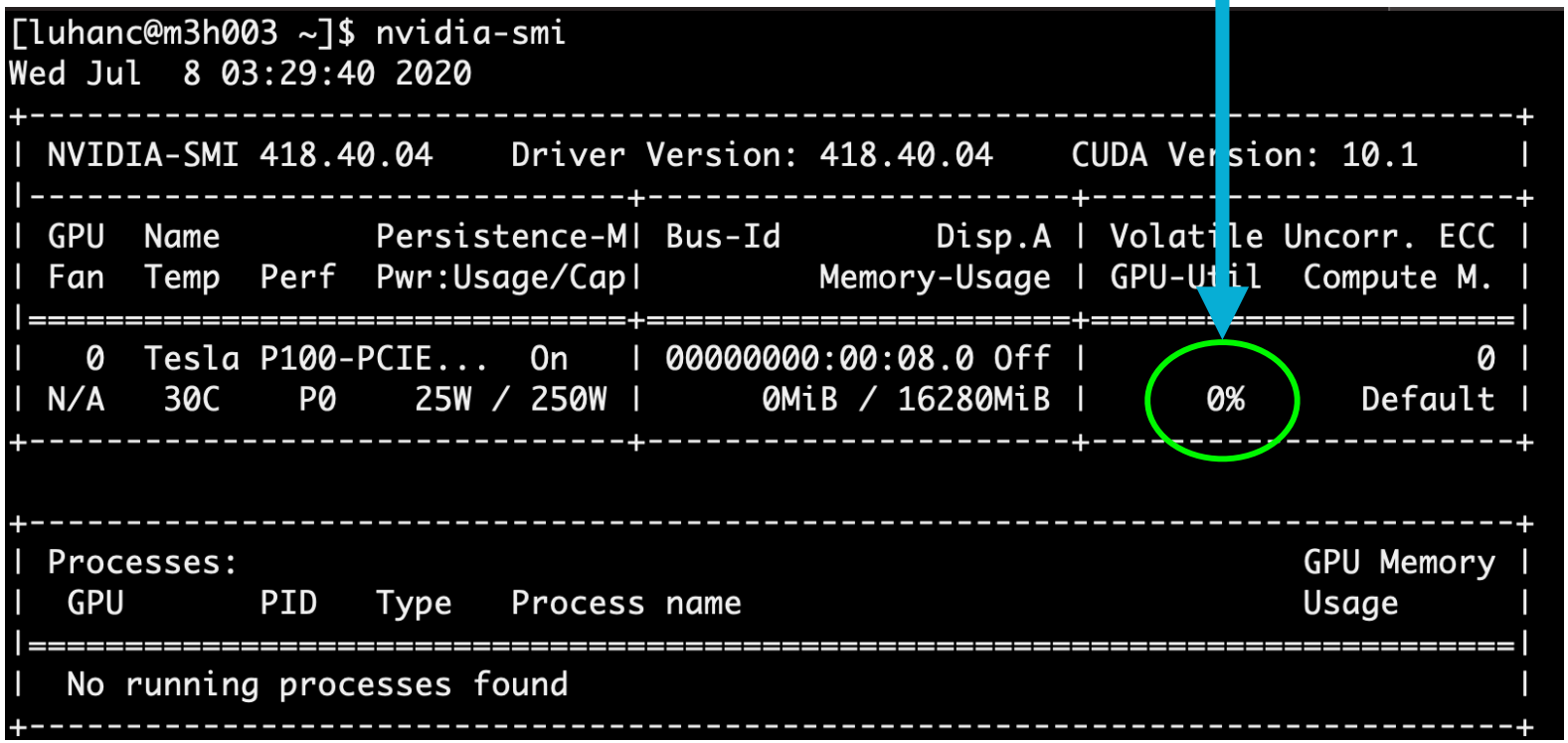
For CPU and memory



nvidia-smi

For GPU

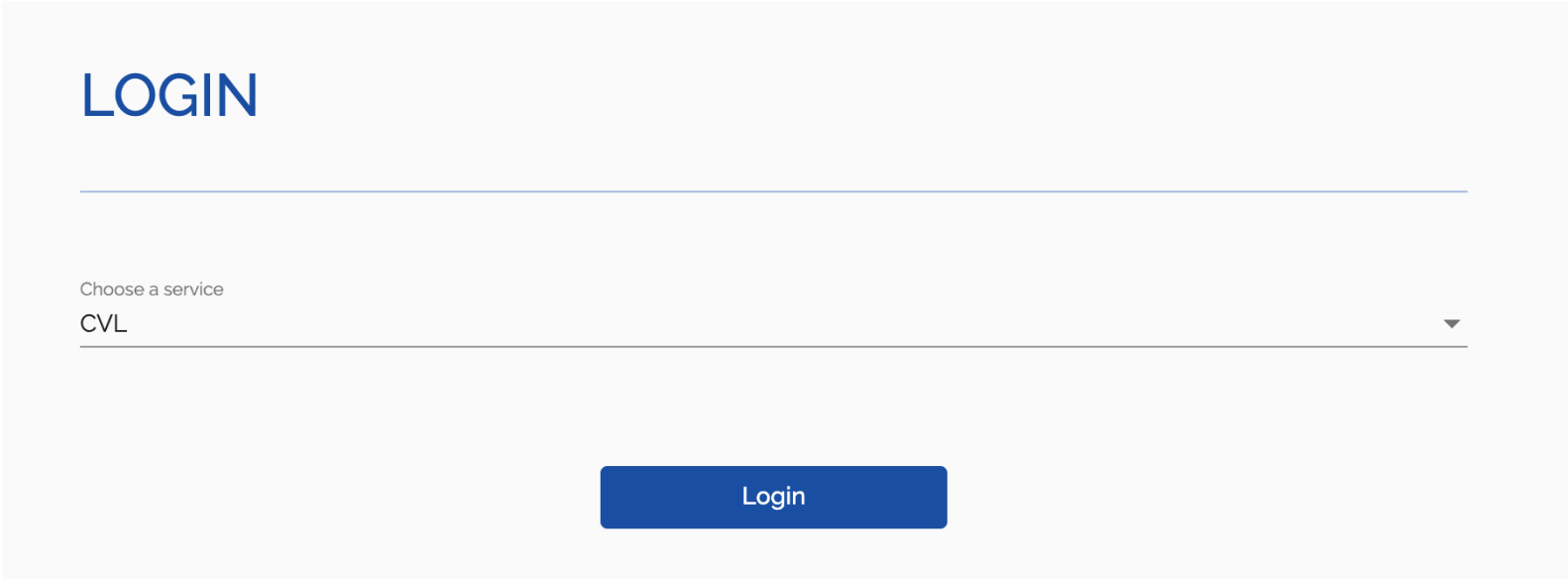
Maximizing GPU utility by pushing this value to its max



Visualization and Strudel2

Remote Desktop for Graphics

- Go to https://beta.desktop.cvl.org.au/login
- Select CVL as the service then login using AFF with your monash account



conda environment

- Creating your conda environment

```
module load conda-install
conda-install <absolute path of installation directory>
source <absolute path of installation directory>/bin/activate
```

- Install necessary packages for desktop

```
conda install -c conda-forge -y jupyter jupyterlab
```

A MNIST Example

Deep Learning + GPU + Multi-node

- Existing tools generally take care of multi-processing (OpenMP) and GPU (CUDA) well. But Multi-node (MPI) computing is more convoluted.
- Copy data is expensive, optimisation means reduce $\frac{communication}{computation}$
- Data parallel vs model parallel

Question