R on the HPC and parallel computing

Eleanor Zhang

Date: 08/18/2022

Aims

Part I: Parallel computing in R: *parallel* package

Part II: Parallel computing using batch jobs on HPC cluster

Part III: Practice on your own

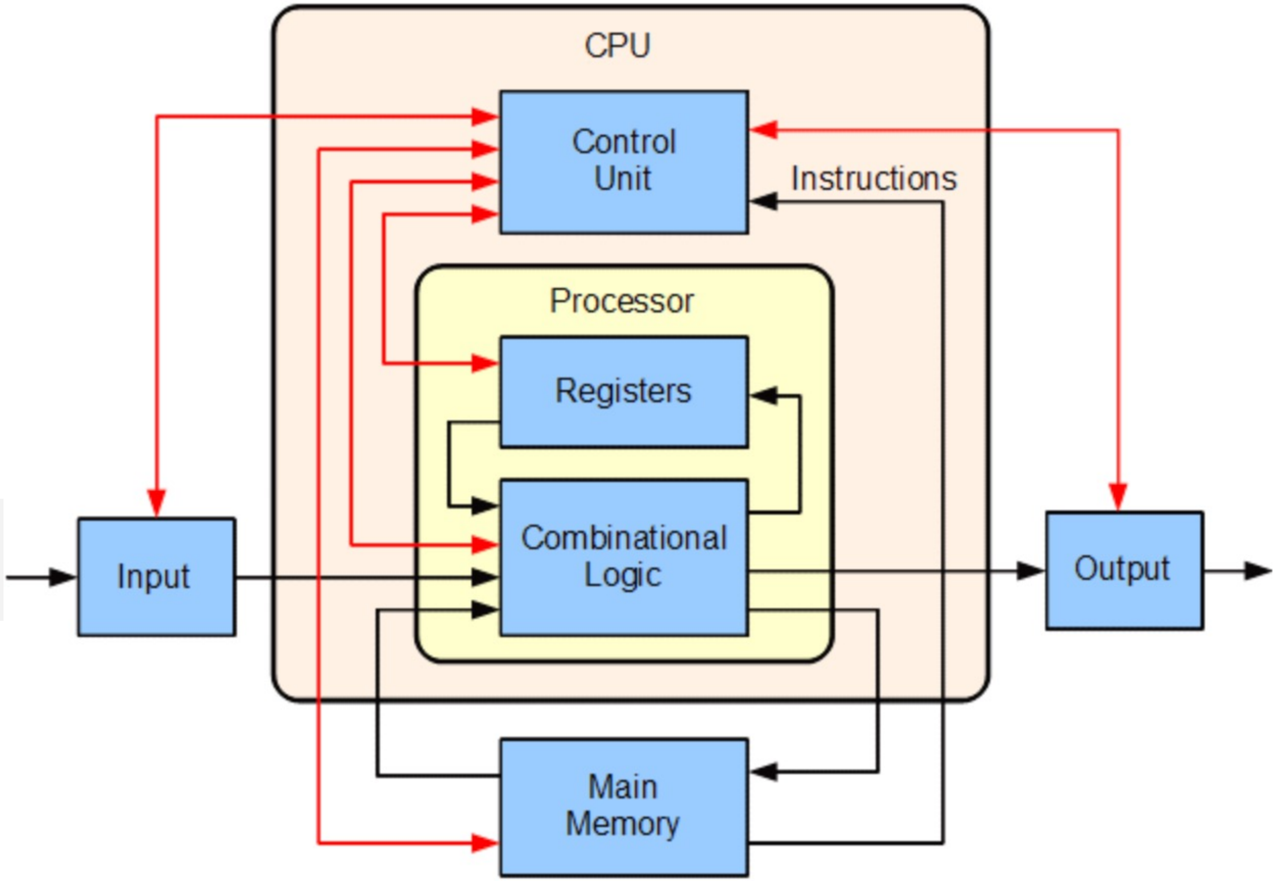
How does R interact with operational system?

```
for (mean_val in c(0, 1)){
  print(rnorm(3, mean=mean_val, sd=1))
}
```

Interaction between R and CPU

- Central processing unit (CPU): the "brain"

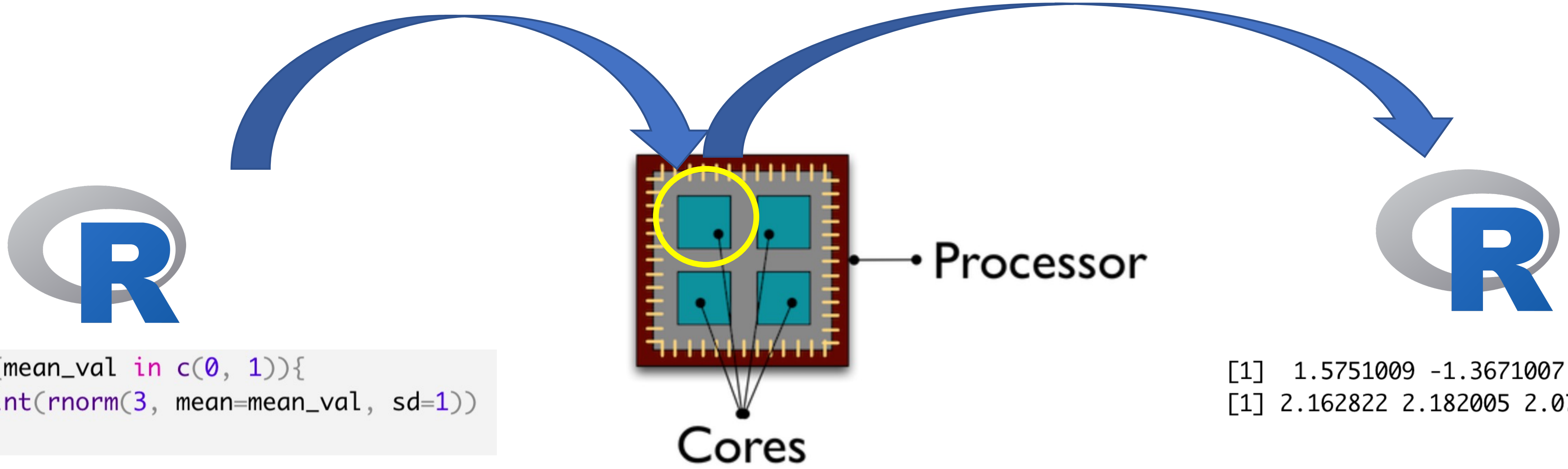
```
for (mean_val in c(0, 1)){
  print(rnorm(3, mean=mean_val, sd=1))
}
```



```
[1]  1.5751009 -1.3671007  0.3757906
[1] 2.162822 2.182005 2.077165
```

Eg. RAM memory

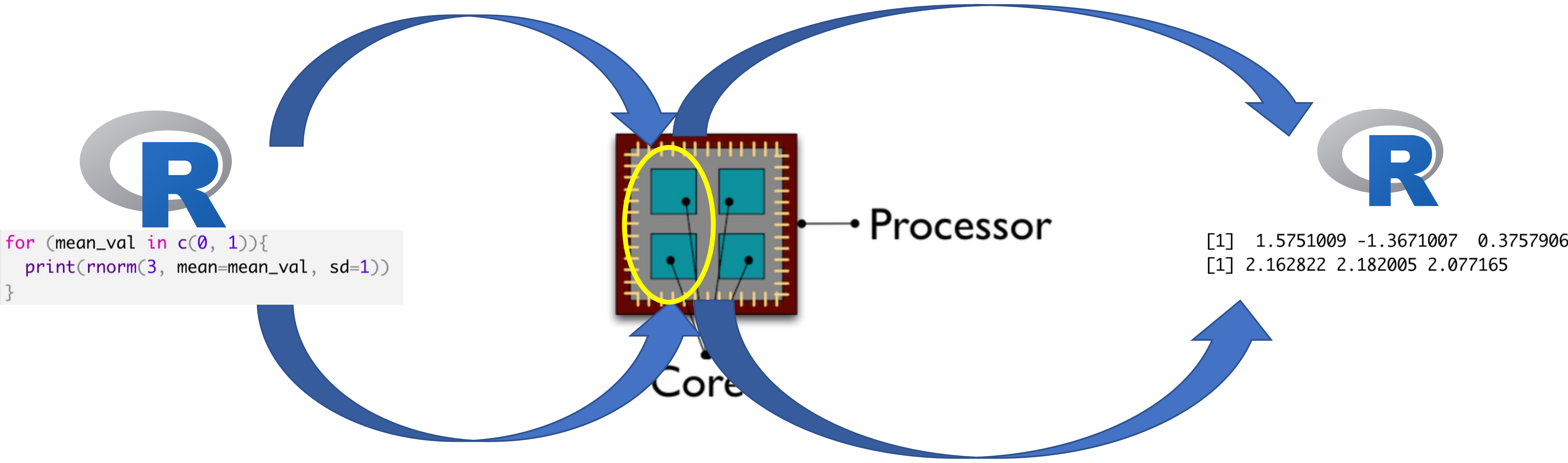
R uses single core by default



```
for (mean_val in c(0, 1)){
  print(rnorm(3, mean=mean_val, sd=1))
}
```

```
[1]  1.5751009 -1.3671007  0.3757906
[1] 2.162822 2.182005 2.077165
```

R in parallel using multi-core for independent tasks



```
for (mean_val in c(0, 1)){
  print(rnorm(3, mean=mean_val, sd=1))
}
```

```
[1]  1.5751009 -1.3671007  0.3757906
[1] 2.162822 2.182005 2.077165
```

Parallel package:

Two functions to implement multi-cores for independent tasks:

	Forking method	**Socket method**
R function	mclapply()	parLapply()

→ Go to parallel.Rmd

Parallel package:

Two functions to implement multi-cores for independent tasks:

	Forking method	**Socket method**
R function	mcapply()	parLapply()
Pros and Cons	Simple; Not applicable to Windows users	Structured code; Compatible with any system

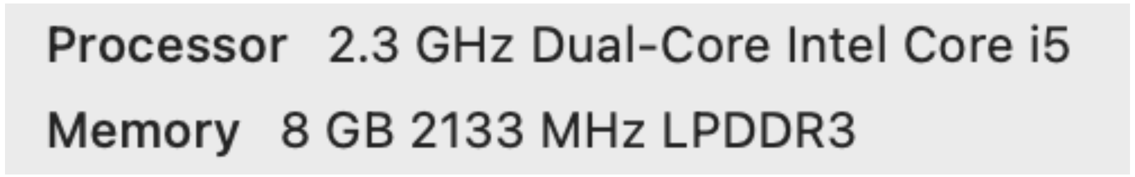
Overhead in parallel computing

- Amdahl’s law: speed up in computing time is NOT proportional to number of cores
- Some computation tasks not worth parallelization
- Try out and see what number of cores to use

Happy with your personal computer?

Hardware on personal computer

- CPU and RAM:



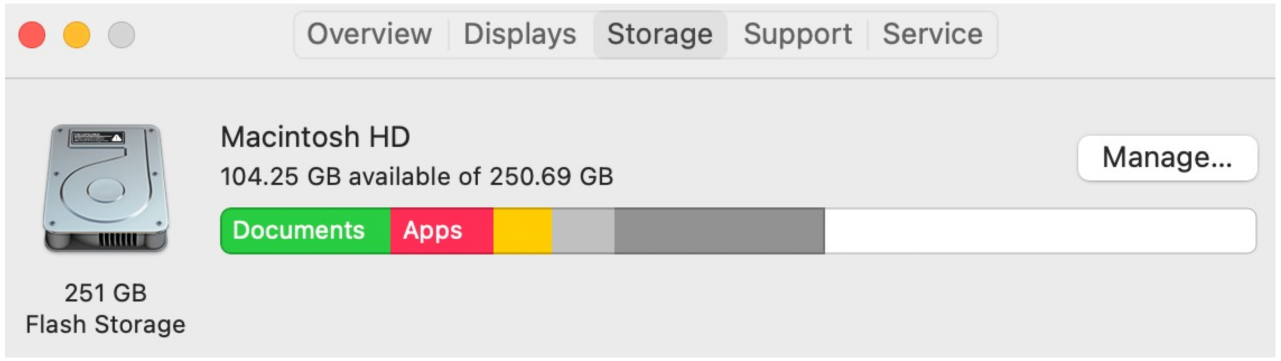
Processor 2.3 GHz Dual-Core Intel Core i5

Memory 8 GB 2133 MHz LPDDR3

- By default, R uses single core and single thread
- R stores and manipulate all objects in RAM in R session

Hardware on personal computer

- Storage on disk



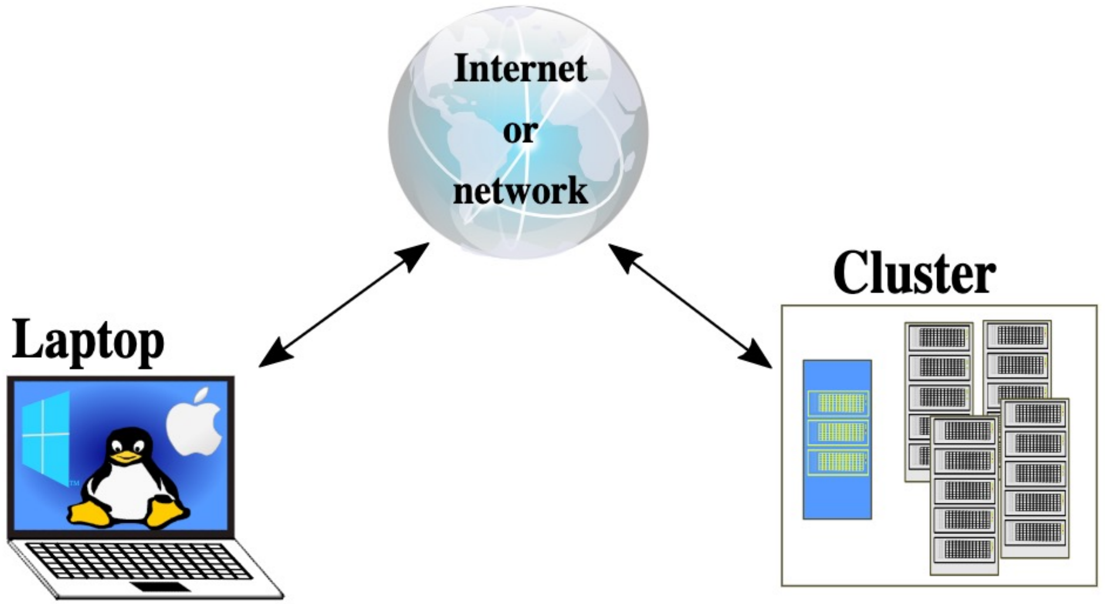
Macintosh HD
104.25 GB available of 250.69 GB

251 GB
Flash Storage

- R write to disk: save(), write_tsv(), etc.

High performance computing (HPC)

- HPC aggregates the resources from individual computers (known as *nodes*) into a cluster that works together to perform advanced, specialized computing jobs.



https://carpentries-incubator.github.io/hpc-intro/11-connecting/index.html

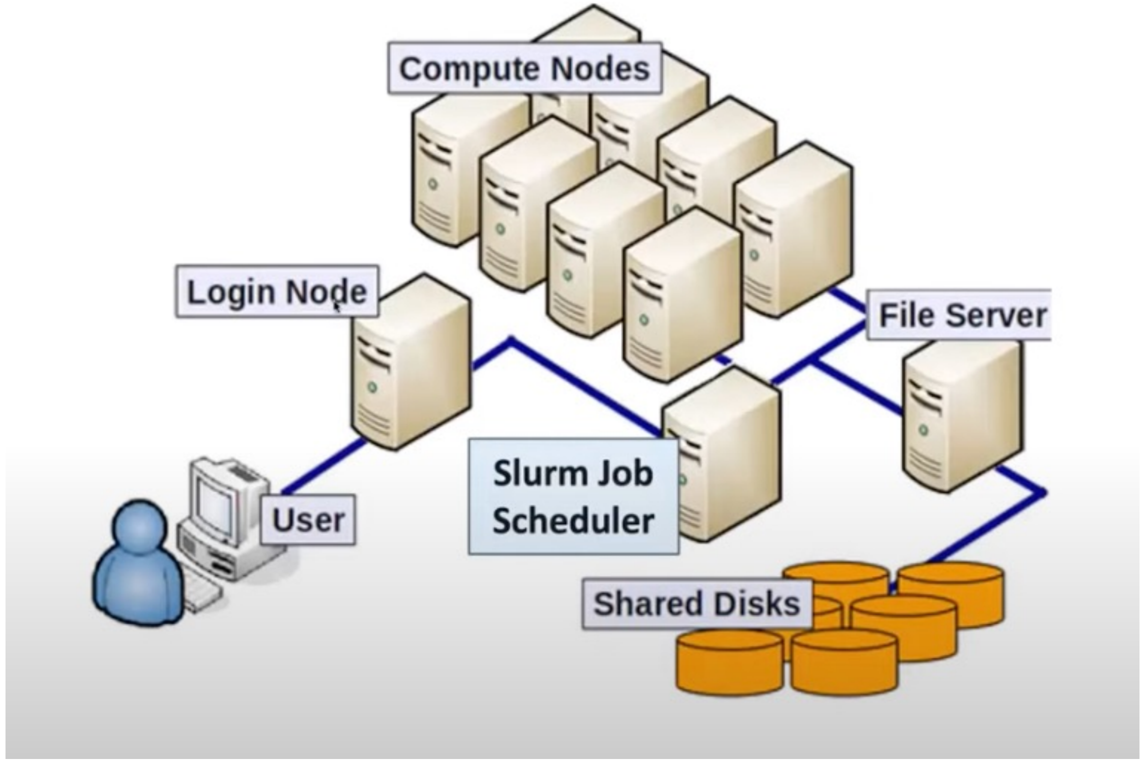
HPC at USC

CARC: Center for Advanced Research Computing

- https://www.carc.usc.edu/
- https://www.carc.usc.edu/user-information/user-guides/hpc-basics/discovery-resources
- Provides workshop and weekly office hours

CARC computing resource

- Login node (Head node)
- Compute node wired together (~400)



https://www.carc.usc.edu/user-information/user-guides/hpc-basics/getting-started-discovery

CARC computing resource

- Login node (Head node)
- Compute node wired together (~400)

One rack in HPC cluster

Multiple racks in HPC cluster

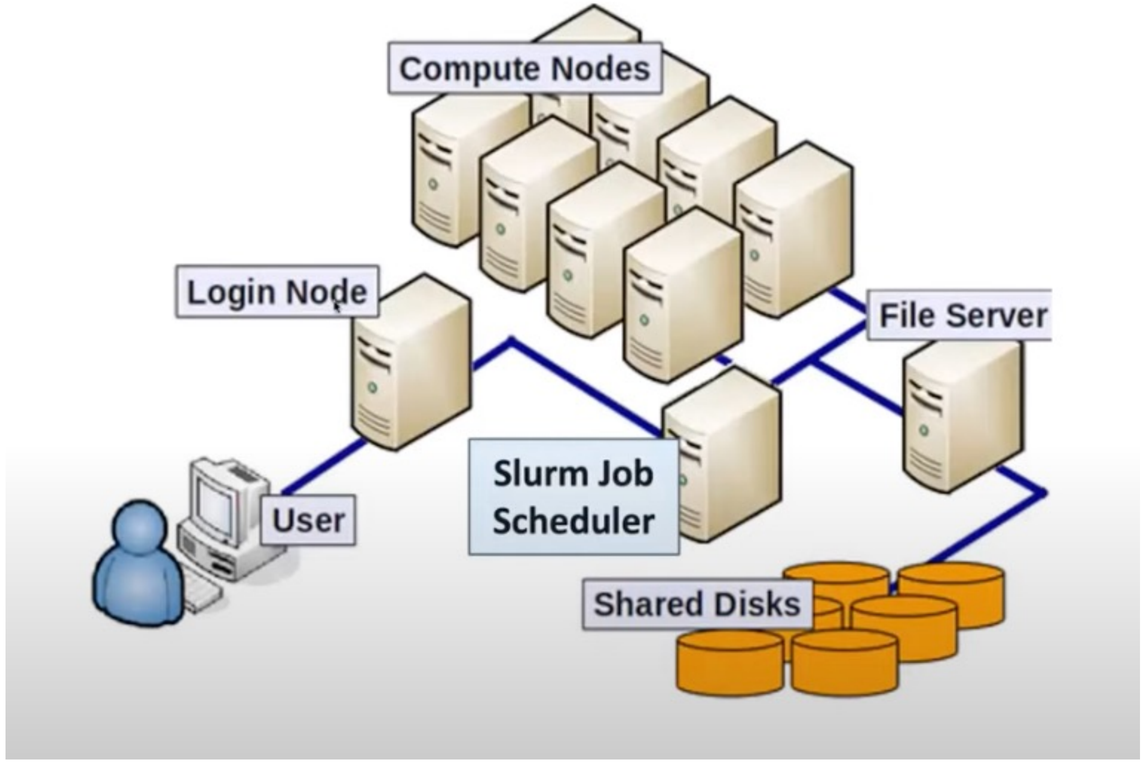
https://www.carc.usc.edu/user-information/user-guides/hpc-basics/getting-started-discovery

CARC computing resource

- Login node (Head node)
- Compute node wired together (~400)
- Networked storage (~12 PB)

- Disk array and file servers

- Slurm job scheduler
- **Discovery** and Endeavour



https://www.carc.usc.edu/user-information/user-guides/hpc-basics/getting-started-discovery

Discovery account

- PI can add a member to a project
- For this R bootcamp session, Eric added us to a project and we should have access to /project/ekawaguc_906
- Username: USC NetID (eg. zzhang39@usc.edu)

Connecting to Discovery login node

Method 1: Secure shell command line interface

- On Mac: Terminal
- On Windows: OpenSSH, Putty, etc

```
ssh  <username>@discovery.usc.edu
```

Method 2: Use OnDemand interface

https://www.carc.usc.edu/user-information/user-guides/hpc-basics/getting-started-ondemand

Requirement:

- Network connection
- VPN if off-campus
- DUO Two Factor Authentication: see https://itservices.usc.edu/duo/ to set up

Navigate file system

```
myquota
```

- Home directory (private):

/home1/<username>

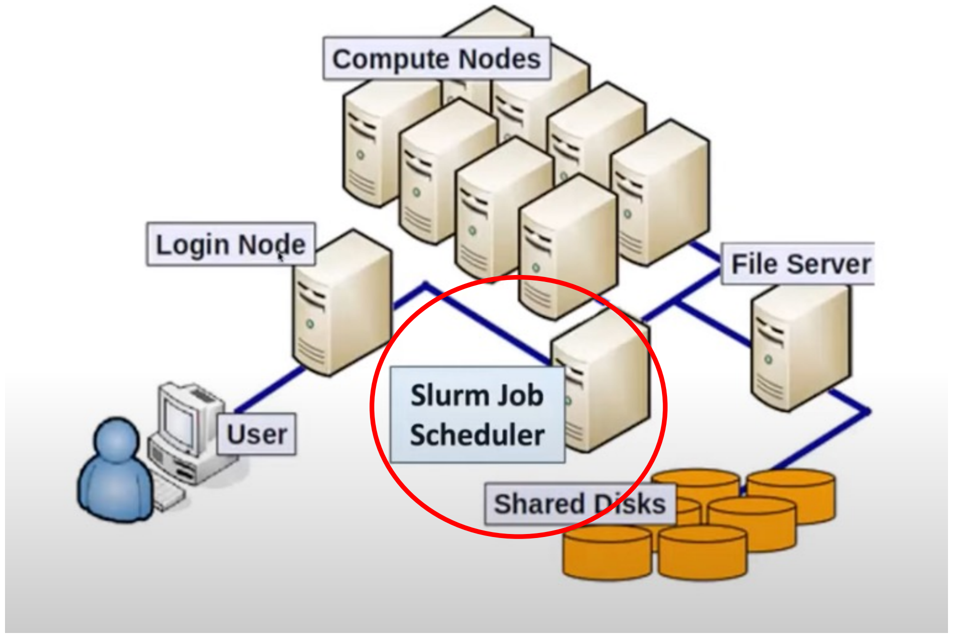
- Two scratch directories:

/scratch1/<username>
/scratch2/<username>

- Project: /project/<pi_username_id>

Slurm: the “manager”

- Slurm is a cluster management system that can allocate and monitor computing resources



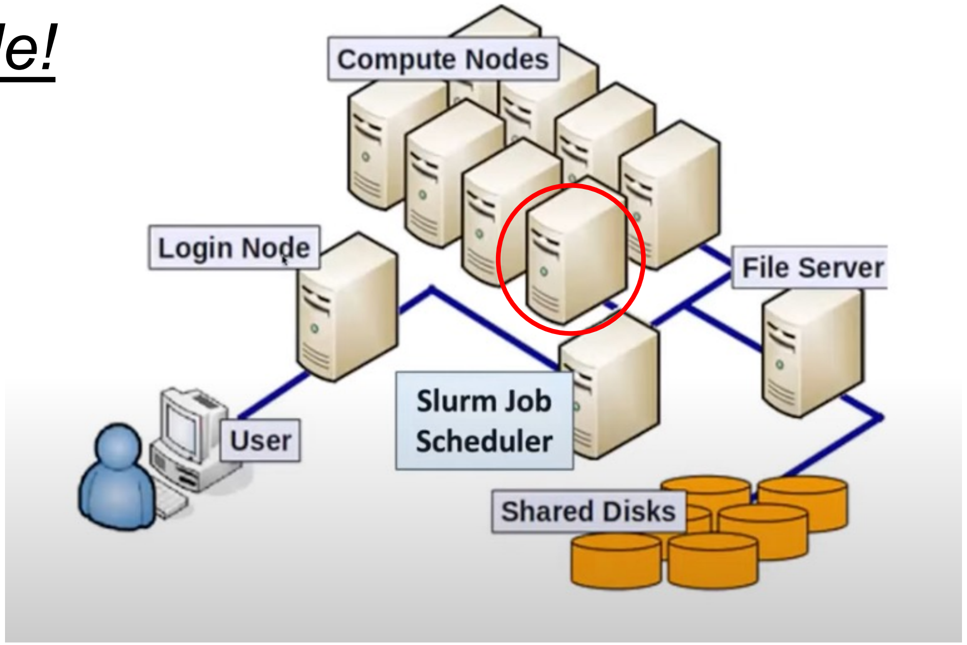
https://www.carc.usc.edu/user-information/user-guides/hpc-basics/getting-started-discovery

Request for interactive compute node

In shell:

```
salloc --time=1:00:00 --mem=8Gb
```

<u>Please do not run heavy task on login node!</u>



https://www.carc.usc.edu/user-information/user-guides/hpc-basics/getting-started-discovery

Using R on HPC

Method 1: Pre-built R module by CARC

```
module purge                      # remove all currently loaded modules
module load gcc/11.3.0            # load complier
module load openblas/0.3.20       # load linear algebra subprogram
module load r/4.2.1
```

Method 2: Install and manage R using other distributions: eg. Anaconda (Practice in the end)

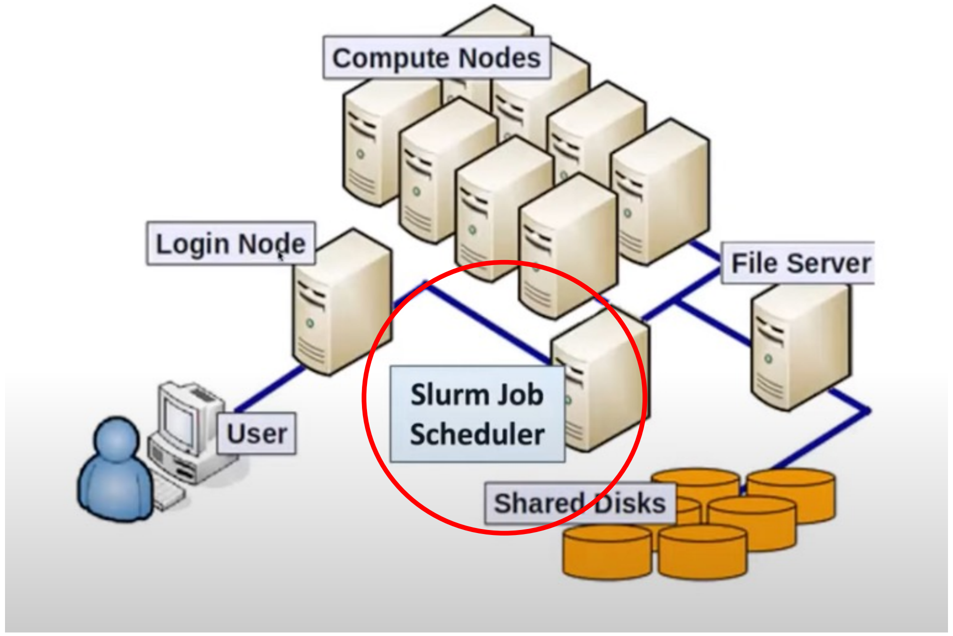
Run R script on command line

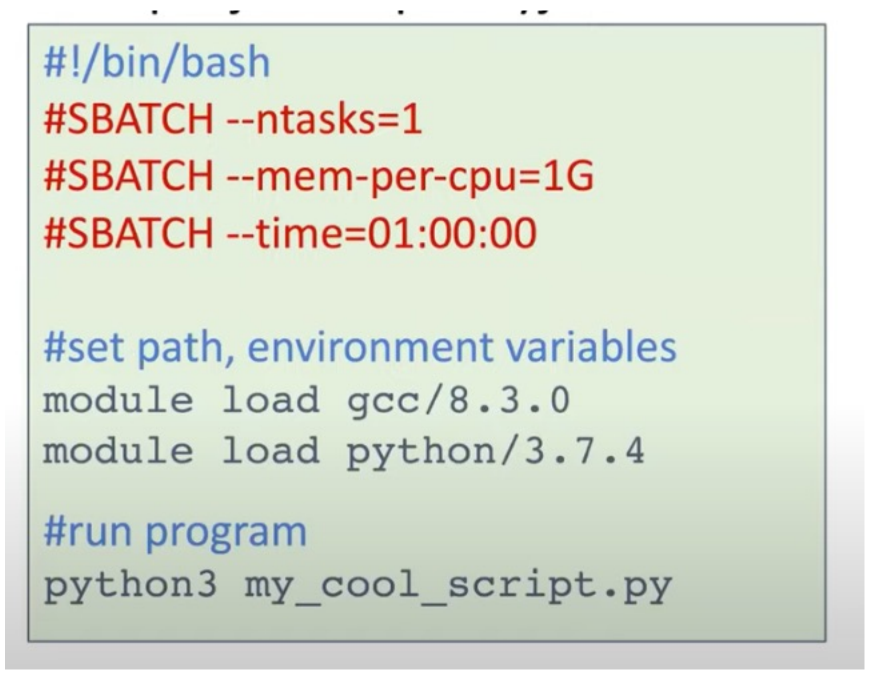
- Create R script (with or without arguments)
- Text editor: vim, Emacs, nano, etc
- Run command in shell:

```
Rscript <scriptname> arg1 arg2 …
```

Submit batch jobs using Slurm

- Batch job: commands that will be executed with little interaction between user and system
- Use job script to submit batch jobs





```
#!/bin/bash
#SBATCH --ntasks=1
#SBATCH --mem-per-cpu=1G
#SBATCH --time=01:00:00

#set path, environment variables
module load gcc/8.3.0
module load python/3.7.4

#run program
python3 my_cool_script.py
```

https://www.carc.usc.edu/user-information/user-guides/hpc-basics/getting-started-discovery

Example of using batch jobs

- Previous example

Calculate bootstrap standard errors for slope in logistic regression:

Species ~ Sepal Length

- Now let’s calculate bootstrap standard errors of slope for each feature variable:

Species ~ Sepal.Width

Species ~ Sepal.Length

Species ~ Petal.Width

Species ~ Pepal.Length

Practice

- Set up DUO two factor: https://itservices.usc.edu/duo/
- Go to practice_instruction.html file and follow the steps to practice on your own
- Let me know if you have questions

Reference

R parallel computing:

https://nceas.github.io/oss-lessons/parallel-computing-in-r/parallel-computing-in-r.html

USC HPC workshops:

- Discovery cluster overview: https://www.youtube.com/watch?v=gtX6VM2UkTo
- Using R: https://www.youtube.com/watch?v=3sRy2Lx8HIc&t=658s