New Users Training
Introduction to FASRC clusters

Learning objectives 1 – FASRC account

o   Learn how to request an FASRC account

o   Activate your new account

o   How to modify your account or add groups

Learning objectives 2 – Intro to HPC

o What is high-performance computing (HPC)? How is it different from a desktop/laptop?

o Laptop vs. Cannon

o Why HPC?

o FASRC clusters

o Cluster architecture

o Job scheduler

o Choose compute resources for jobs

- Memory, cores

- Partitions, file systems

o Storage

o Data Management

o Cluster customs and responsibilities

Learning objectives 3 – Documentation and help

o FASRC docs

o GitHub User_codes

o Office hours

o Tickets

Request FASRC account

https://docs.rc.fas.harvard.edu/kb/quickstart-guide/

1. Request an account using Account Request Tool
   https://portal.rc.fas.harvard.edu/request/account/new

   • Use Harvard Key option

2. Set FASRC password https://portal.rc.fas.harvard.edu/p3/pwreset/

3. Set two-factor authentication https://docs.rc.fas.harvard.edu/kb/openauth/

4. Set FASRC VPN (needed for mounting storage, OOD, level 3 data, license server access)
   https://docs.rc.fas.harvard.edu/kb/vpn-setup/

5. Review intro training

How to modify your account

o Change labs: https://docs.rc.fas.harvard.edu/kb/change-lab-group/

o Add a lab:

   • Portal gives access to lab storage: https://docs.rc.fas.harvard.edu/kb/additional-groups/

   • If you work for more than 1 PI, and need access to lab slurm account (more on slurm later), send a ticket

o Never request a second account!!

o Membership in the FASRC mailing-list is required

o Account needs to be used in the last 12 months to be active

o After 12 months of inactivity

   • Account is disabled, but nothing is deleted

   • Can be reactivated with PI's/admin approval

What is HPC?

o   HPC: High performance computing

o   HPC: biggest and fastest computing machines right now

o   Supercomputers: rule of thumb - at least 100 times as powerful as a PC (personal computer)

o   Jargon: other terms

- Supercomputing

- Cyberinfrastructure (CI)

- Cluster computing

Laptop vs. Cannon

MacBook Pro

o   1 CPU (processor)

o   4-12 cores per CPU

o   Memory: 16-96 GB

Cannon typical nodes

o   2 CPUs

o   24-32 cores **per CPU**

o   Memory: 184-2000 GB

o   **1260+ nodes!!!**

Why HPC?

o Size: problems that can't fit on a desktop/laptop, for example 500+ GB of RAM or 100s of cores

o Speed: problems that take months on a laptop may take a few hours on a supercomputer

o Amount: need 1000s of runs

45 miles/hour

600 miles/hour

What about FASRC clusters?
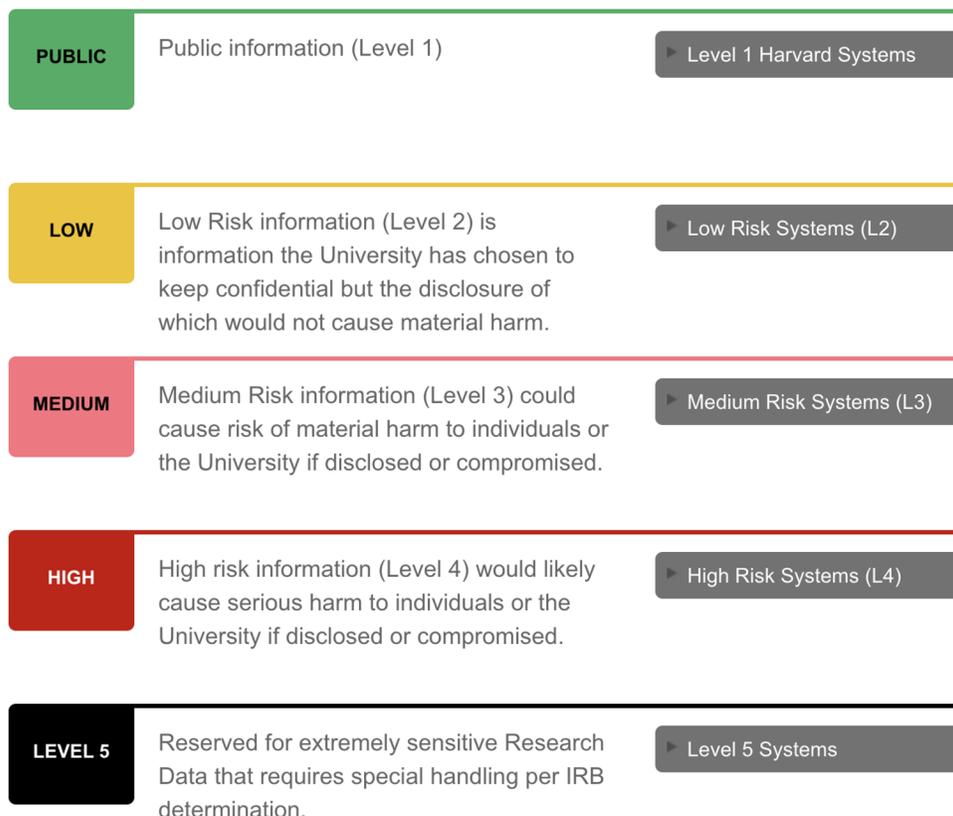
Massachusetts Green HPC Center
(MGHPCC)

Cannon cluster

From https://www.servethehome.com/the-harvard-cannon-powered-by-lenovo-neptune/
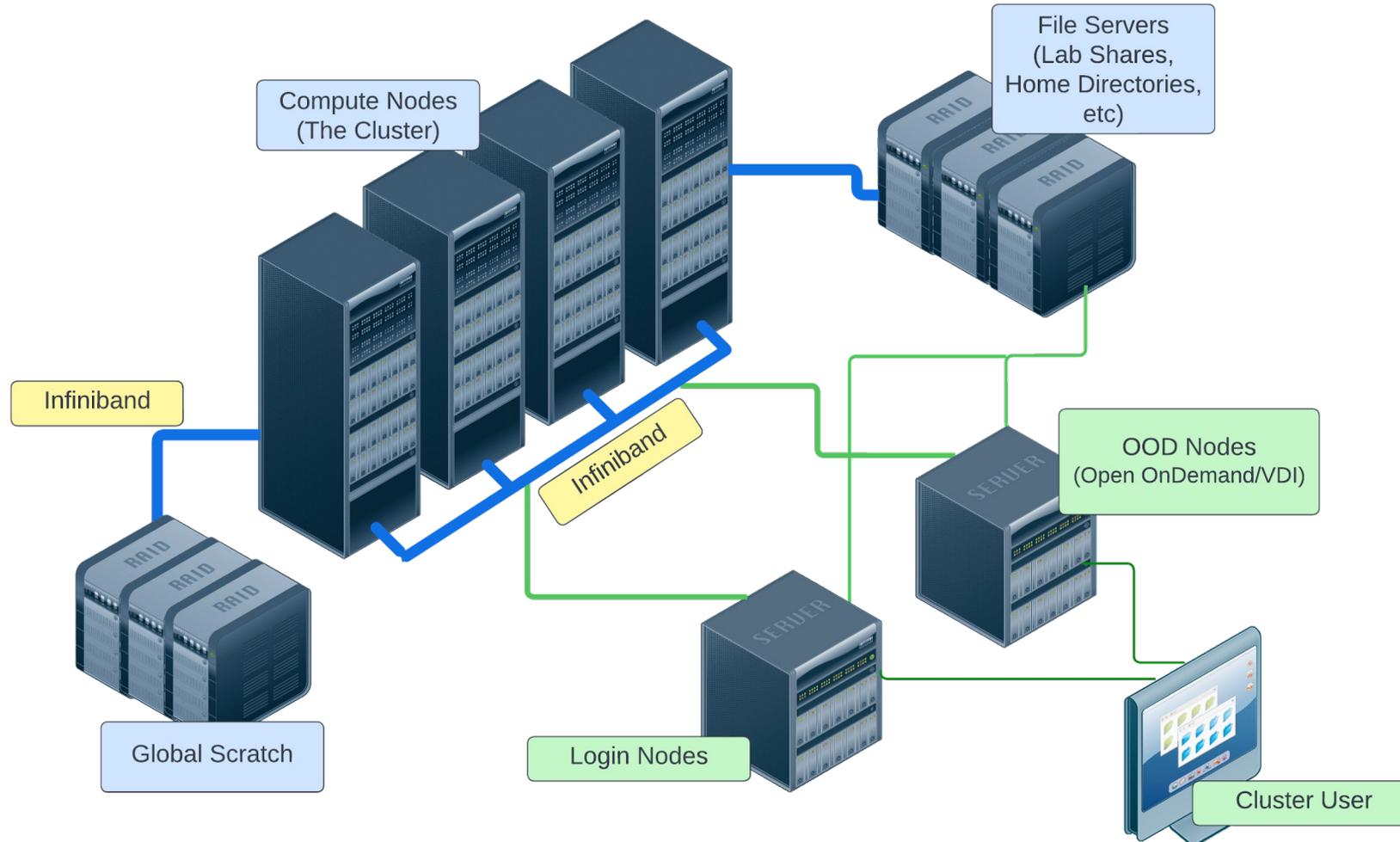
FASRC clusters: Cannon and FASSE

Cannon
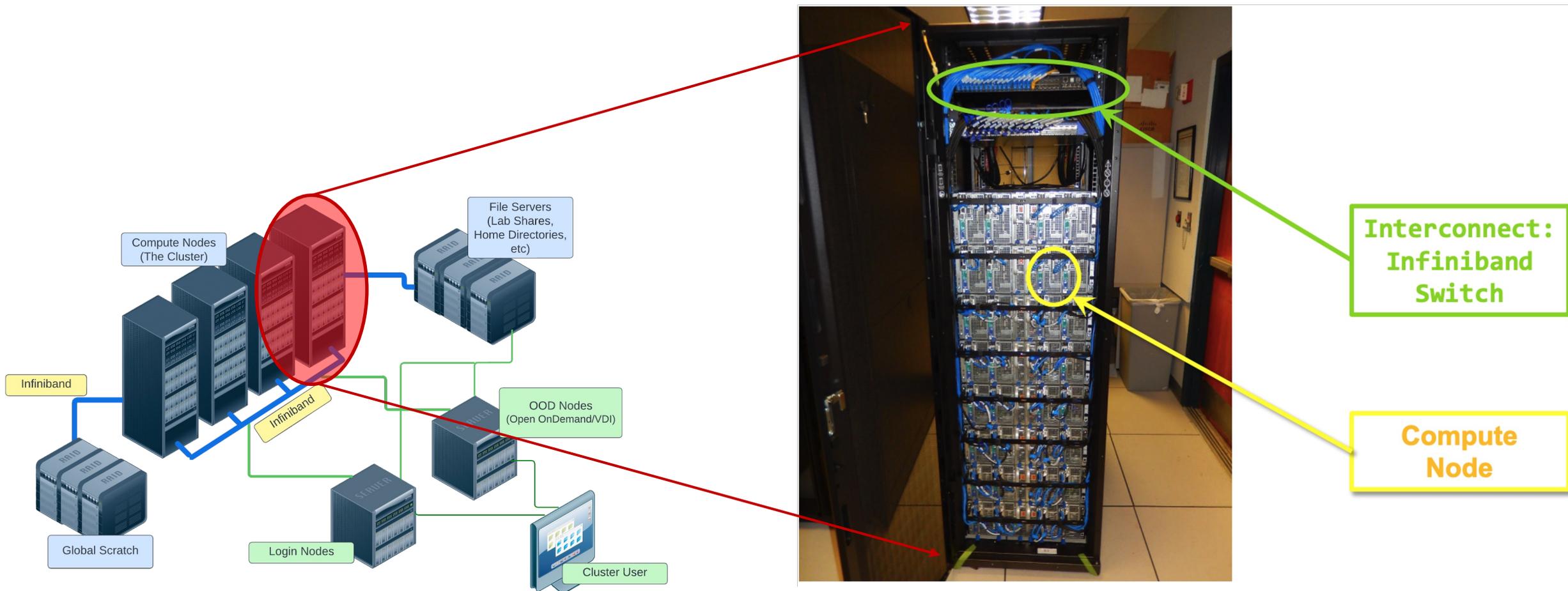
○ General purpose

○ Only level 1 and 2 data

FASSE

○ FAS Secure Environment

○ Secure multi-tenant environment

○ Analysis of sensitive datasets with DUAs and IRBs

○ Level 3 data, no level 4 data
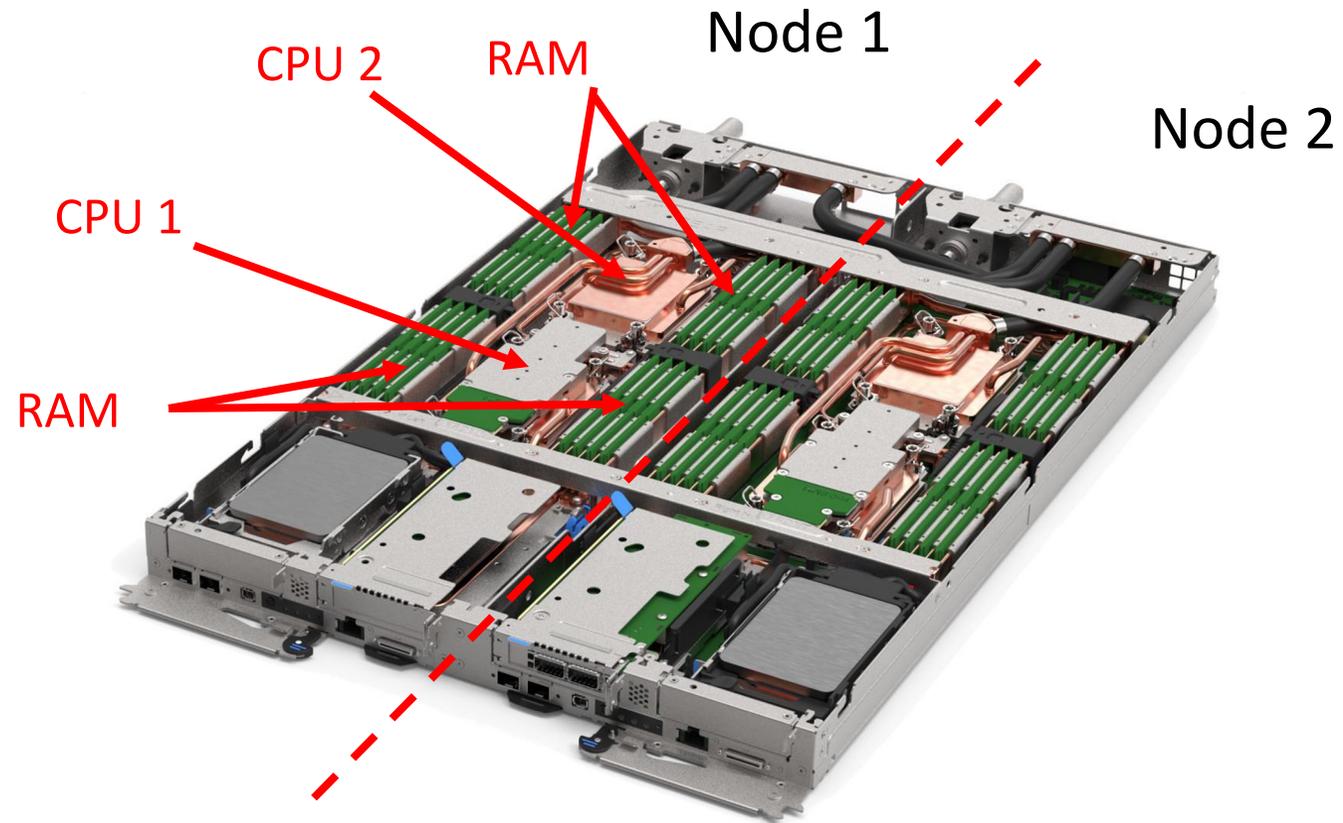
○ PI/lab responsibility to know their data

○ https://policy.security.harvard.edu/
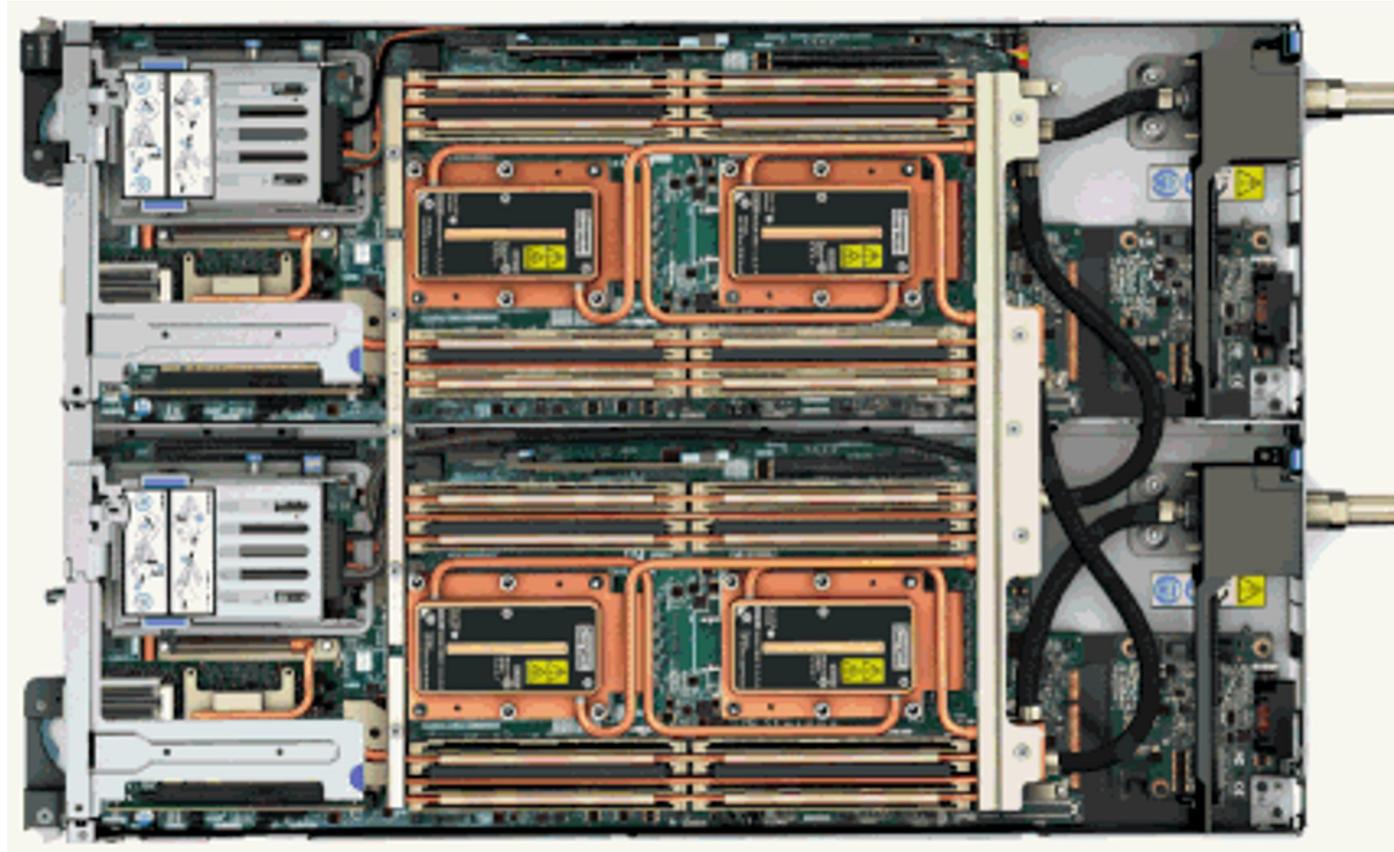
○ https://docs.rc.fas.harvard.edu/kb/data-use-agreements/

○ https://security.harvard.edu/

○ https://docs.rc.fas.harvard.edu/kb/fasse/

PUBLIC — Public information (Level 1) — ▸ Level 1 Harvard Systems

LOW — Low Risk information (Level 2) is information the University has chosen to keep confidential but the disclosure of which would not cause material harm. — ▸ Low Risk Systems (L2)

MEDIUM — Medium Risk information (Level 3) could cause risk of material harm to individuals or the University if disclosed or compromised. — ▸ Medium Risk Systems (L3)

HIGH — High risk information (Level 4) would likely cause serious harm to individuals or the University if disclosed or compromised. — ▸ High Risk Systems (L4)

LEVEL 5 — Reserved for extremely sensitive Research Data that requires special handling per IRB determination. — ▸ Level 5 Systems

Cluster architecture

Rack

Compute Nodes (The Cluster)

Infiniband

Global Scratch

File Servers (Lab Shares, Home Directories, etc)

Infiniband

OOD Nodes (Open OnDemand/VDI)

Login Nodes

Cluster User

Interconnect: Infiniband Switch

Compute Node

From HPC@LSU training (http://www.hpc.lsu.edu/training/weekly-materials/2022-Fall/HPC_UserEnv1_Fall2022.pdf)
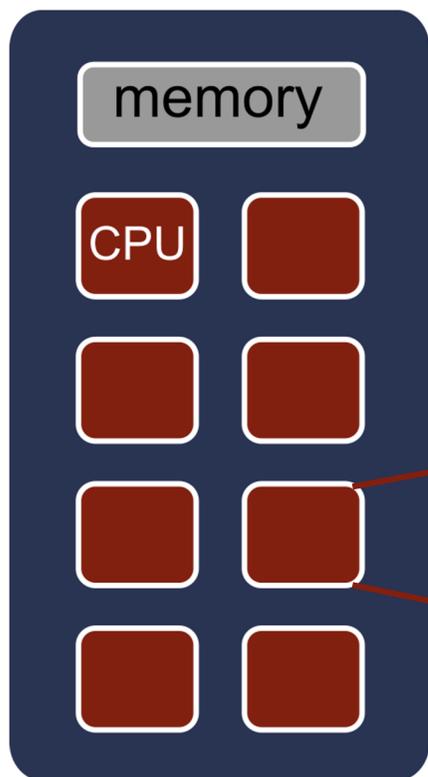
Node

From https://lenovopress.lenovo.com/lp1603-thinksystem-sd650-v3-server

Node water cooling

From https://www.rc.fas.harvard.edu/blog/cannon-makes-top-500-list/

Node, processors, core

Node: a computer in the cluster
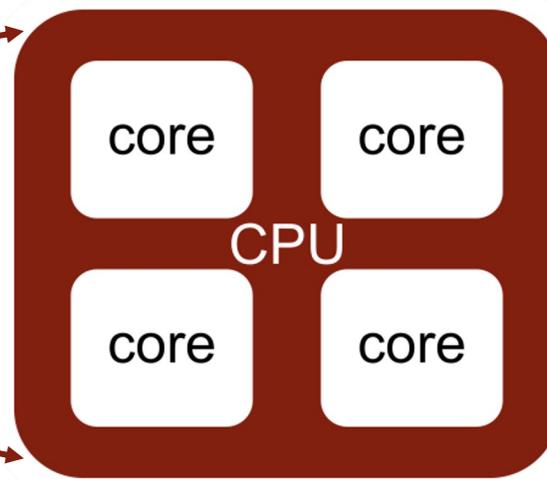
CPU
o   Central processing unit, processor
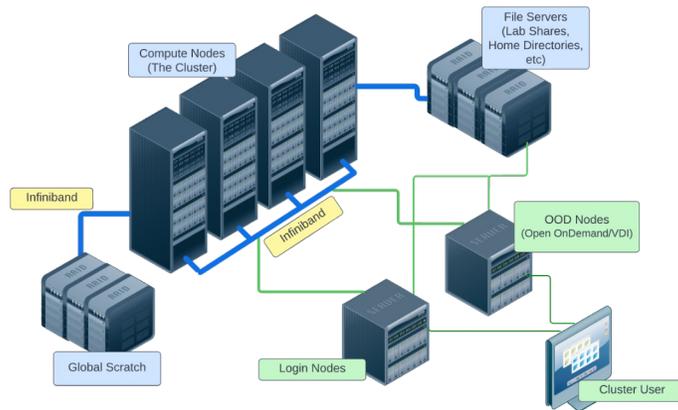o   Can have many cores

Cores
o   Basic unit of compute
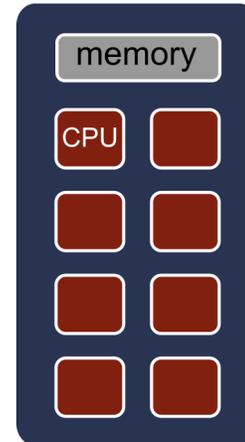o   Runs a single instruction of code

Nomenclature summary
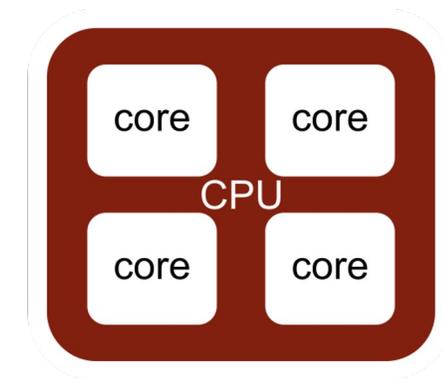
Cluster
Top level unit of a supercomputer

Node
One host in the cluster
(i.e., one computer)

Core
Basic unit of computer

New term: Job

A user's request to use a certain amount of resources for a specific amount of time

Glossary: https://docs.rc.fas.harvard.edu/kb/glossary/

Adapted from HPC@LSU training (http://www.hpc.lsu.edu/training/weekly-materials/2021-Summer/HPC_UserEnv_2021_Summer_session_2.pdf)
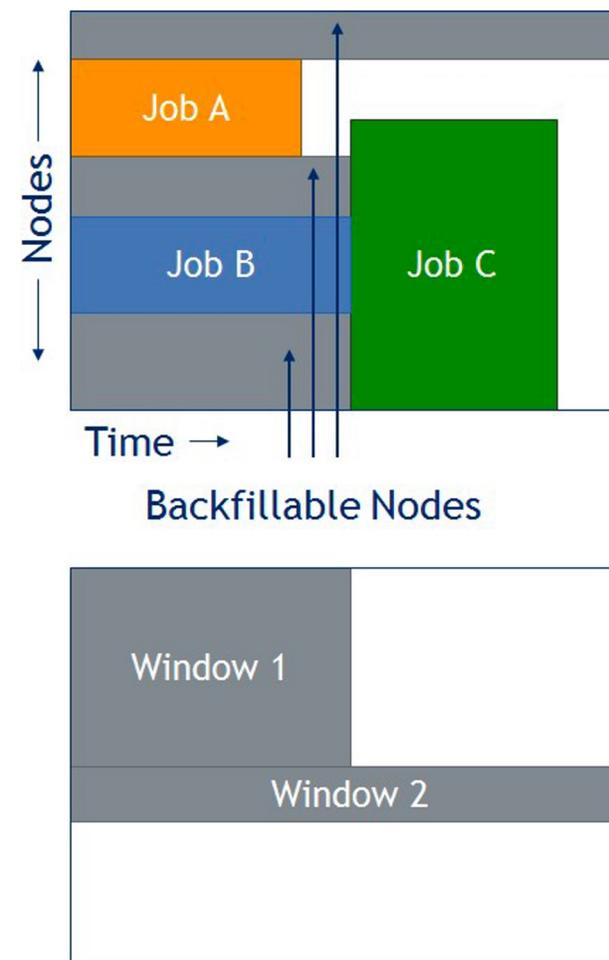
Job scheduler

o   The Cluster is a multi-tenant environment, so how can everyone use it fairly?

o   Job scheduler!

o   Slurm: Simple Linux Utility for Resource Management

- Manages job queue for a cluster of resources
- Prioritizes jobs
- Provides status of running, queue, completed and failed jobs
- Determines the order jobs are executed
- On which node(s) jobs are executed

Job management philosophy

o Prioritize workload
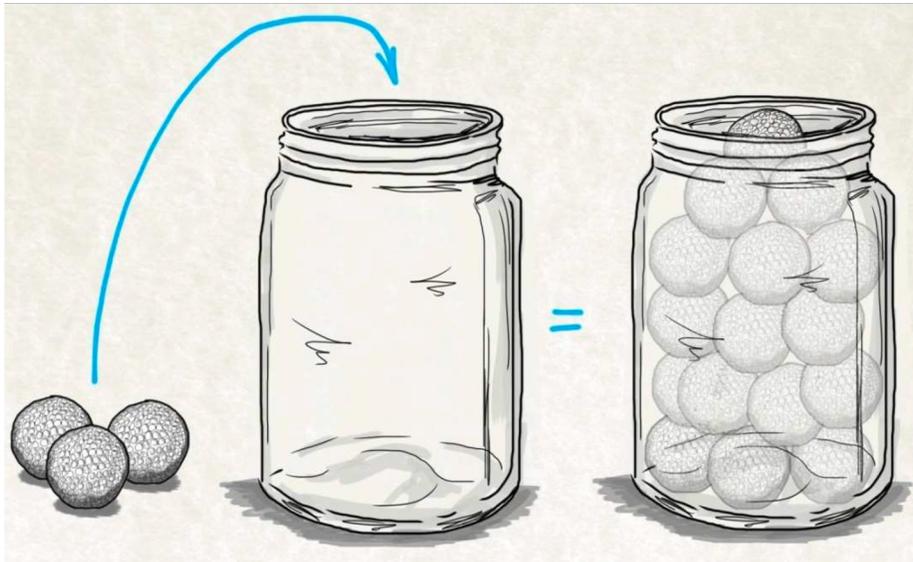
o Backfill idle node to maximize cluster use

Job Priority

o **Not** first come, first served

o Job with higher priority scheduled ahead
of jobs with lower priority

o Priority depends on

- Fairshare

- Amount of time pending

- Group priority
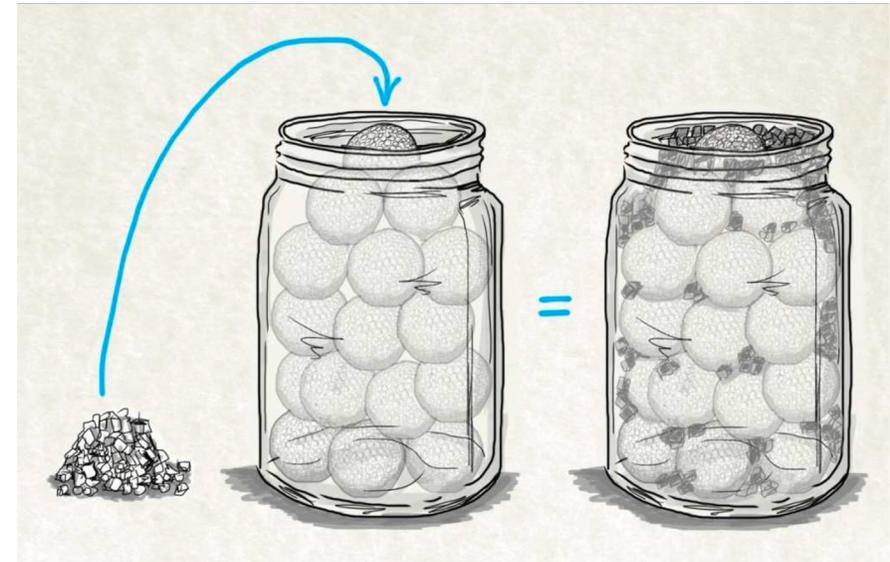
How to maximize cluster usage?

1. Fill in high-priority jobs

2. Backfill with low-priority jobs

Adapted from HPC@LSU training (http://www.hpc.lsu.edu/training/weekly-materials/2021-Summer/HPC_UserEnv_2021_Summer_session_2.pdf)

Choosing computational resources

o How do we choose memory, cores, partitions, and file systems?

o First time ever running on a cluster?

- Run a test case choosing similar resources as the machine you are currently using

- Check how efficient your job was and adjust it accordingly

o Increasing a job/analysis/simulation?

- Run for a small test case

- Increase size by 1.5, 2.0, 2.5x and check how job scaled

- Then you can have a rough estimation of how much a first trial production job of ~10x would require

Cannon partitions

Documentation: https://docs.rc.fas.harvard.edu/kb/running-jobs/

Partitions	shared	gpu	test	gpu_test	serial_requeue	gpu_requeue	bigmem	ultramem	intermediate	bigmem_intermediate	unrestricted	pi_lab
Time Limit	3 days	3 days	12 h	12 h	3 days	3 days	3 days	3 days	3-14 days	3-14 days	no limit	**varies**
# Nodes	264	25	27	10	1264	138	30	3	12	4	8	**varies**
# Cores / Node	48	64 + 4 A100	48	32 + 4 V100	varies	varies	64	64	48	64	64	**varies**
Memory / Node (GB)	196	375	196	375	varies	varies	499	2000	184	499	256	**varies**

FASSE partitions

Documentation: https://docs.rc.fas.harvard.edu/kb/fasse/

Partitions	fasse	fasse_gpu	test	serial_requeue	fasse_bigmem	ultramem	remoteviz	pi_lab
Time Limit	7 days	7 days	12 h	7 days	7 days	7 days	7 days	varies
# Nodes	42	4	5	65	6	1	1	varies
# Cores / Node	48	32 + 4 V100	48	varies	64	64	32	varies
Memory / Node (GB)	184	373	5	varies	499	2000	373	varies

Storage

Tier storage documentation: https://www.rc.fas.harvard.edu/services/data-storage/

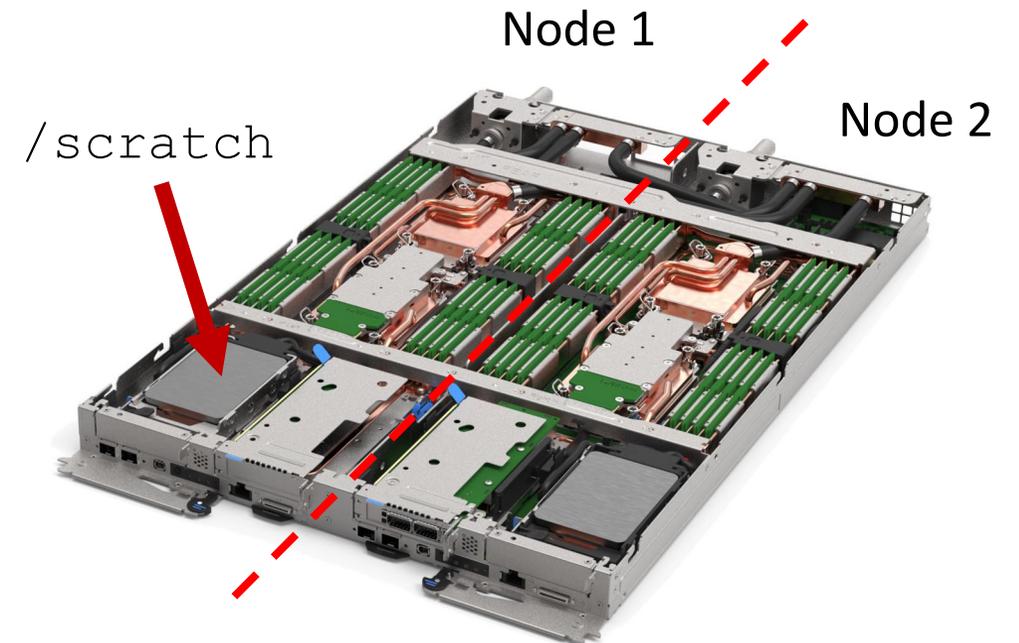|	Home Directories	Lab Directory (Startup)	Local Scratch	Global Scratch	Tier Storage
Mount Point	$HOME /n/home#/$USER /n/home_fasse/$USER	/n/holylabs/pi_lab	/scratch	$SCRATCH /n/holyscratch01/pi_lab	/n/pi_lab
Size Limit	100GB	1- 4TB	70GB/node	2.4PB total	Based on Tier
Availability	All cluster nodes + Desktop/laptop	All cluster nodes	Local compute node only	All cluster nodes	All cluster nodes/ mountable
Retention Policy	Indefinite	Indefinite	Job duration	90 days	Indefinite
Backup	Hourly snapshot + Daily Offsite	No backup	No backup	No backup	Depending on Tier
Performance	Moderate. Not suitable for high I/O	Moderate. Not suitable for high I/O	Suited for small file I/O intensive jobs	Appropriate for large file I/O intensive jobs	Depending on Tier
Cost	Free	Free max of 4TB	Free	Free	Paid

Storage schematics

Global Scratch

o   Networked scratch

o   Global variable: `$SCRATCH`

o   Path: `/n/holyscratch01/pi_lab`

Local Scratch

o   Storage on the node

o   Path: `/scratch`

Node 1

Node 2

`/scratch`

From https://lenovopress.lenovo.com/lp1603-thinksystem-sd650-v3-server
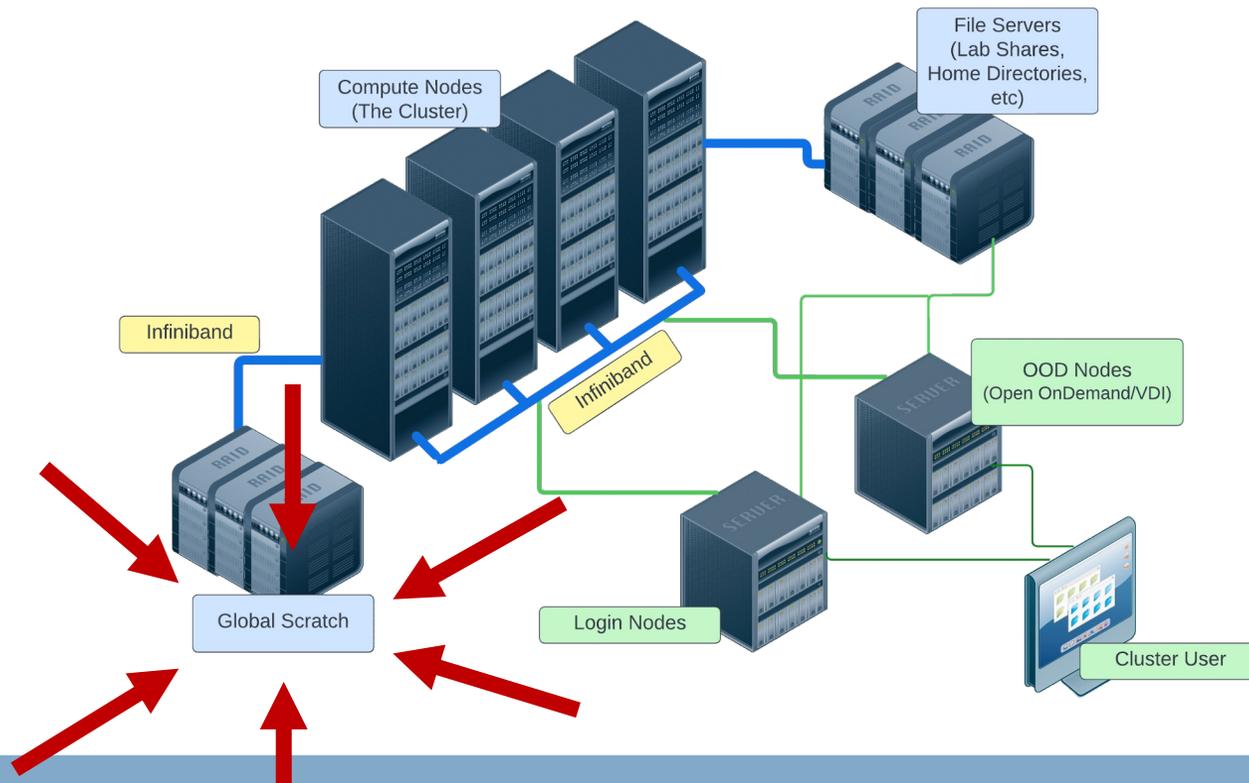
Data management

o **Home**
  - Backed up with daily snapshots (up to 2 weeks)
  - "Valuable" and small code

o **Global scratch**
  - Temporary storage
  - Copy job scripts and executables for jobs
  - Input data
  - Output results

o **Lab storage**
  - Permanent storage
  - If you have code here and not backed up, use version control (git)!!
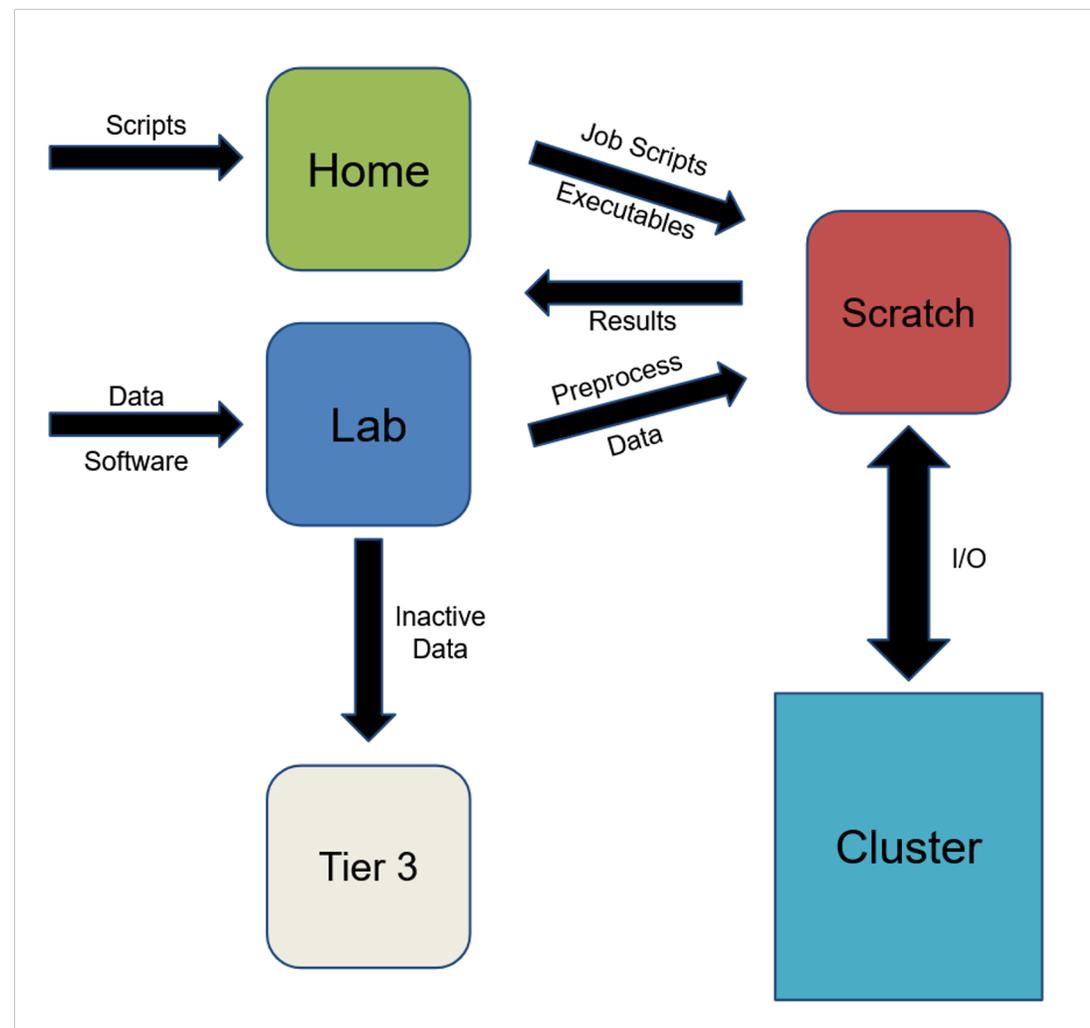
Cluster customs and responsibilities (1)

Documentation: https://docs.rc.fas.harvard.edu/kb/responsibilities/

o   Don't run anything on the login nodes

o   Be as accurate as possible for memory requests

o   Keep job counts reasonable: 10,000 job limit per user (scheduled or running)

o   Request at least 10 minutes

o   Don't overwhelm scheduler: wait 0.5 to 1 sec for `sbatch` and `sacct` commands

Cluster customs and responsibilities (2)

Documentation: https://docs.rc.fas.harvard.edu/kb/responsibilities/

o   Use appropriate partition

o   Use `serial_requeue` and `gpu_requeue` when possible

o   Heavy I/O should be done on `/scratch` and `$SCRATH`

o   Keep at most 1000 files per directory (i.e., folder)

o   No production work on test partitions

o   Poorly behaved jobs will be terminated

o   Don't mine digital currency or misuse Harvard resources

Publications Acknowledging the FASRC Cluster

Documentation: https://docs.rc.fas.harvard.edu/kb/attribution/

o   If you publish work performed on FASRC clusters, acknowledge it:

"The computations in this paper were run on the FASRC Odyssey cluster supported by the FAS Division of Science Research Computing Group at Harvard University."

FASRC documentation

o FASRC docs: https://docs.rc.fas.harvard.edu/

- Search

o GitHub User_codes: https://github.com/fasrc/User_Codes/

o Getting help

- Office hours: https://www.rc.fas.harvard.edu/training/office-hours/

- Ticket

    o Portal: http://portal.rc.fas.harvard.edu/rcrt/submit_ticket (requires login)

    o Email: rchelp@rc.fas.harvard.edu

Upcoming trainings

Training calendar: https://www.rc.fas.harvard.edu/upcoming-training/

Getting started on the FASRC clusters with Open OnDemand

- new users not familiar with command-line interface
- prefers to use a GUI
- Single-node jobs
- Working FASRC account with cluster access
- Content
  - Access Open OnDemand
  - Launch Jupyter, Rstudio Server, Remote Desktop
  - Install Rstudio Server packages
  - Install python packages for Jupyter
  - Launch software from Remote Desktop

Getting started on the FASRC clusters with CLI

- Users familiar with command-line interface
- New to Cannon and FASSE, but familiar with HPC systems
- Working FASRC account with cluster access
- Content
  - Submit interactive job
  - Submit batch job
  - Monitor jobs
  - Cluster software (modules, spack)

Thank you :)

FAS Research Computing