



Alpine: New User Seminar

CURC Alpine: New User Seminar

Instructor:

- Website: www.rc.colorado.edu
- Helpdesk: rc-help@colorado.edu
- Slides: https://github.com/ResearchComputing/New_User_Seminar
- Survey: <http://tinyurl.com/curc-survey18>

Account Check

Does anyone ***not*** have a **CU Research Computing account** who would like to use a temporary account*?

**only available during seminar*

Learning Goals

1. What is CU Research Computing (CURC)?
2. Understand Basic Resources (Alpine cluster)
3. Getting an account & logging in
4. Navigate the RC system
5. Running a job
6. Help!

Things to take note of:

- Confusing, ambiguous, highly nuanced concepts
- Our goal is to help you avoid common mistakes, pitfalls, and frustrations



Ask Questions!

What is CU Research Computing?

What is CU Research Computing?

Large-scale, beyond-the-desktop computing resources for the University of Colorado and more!

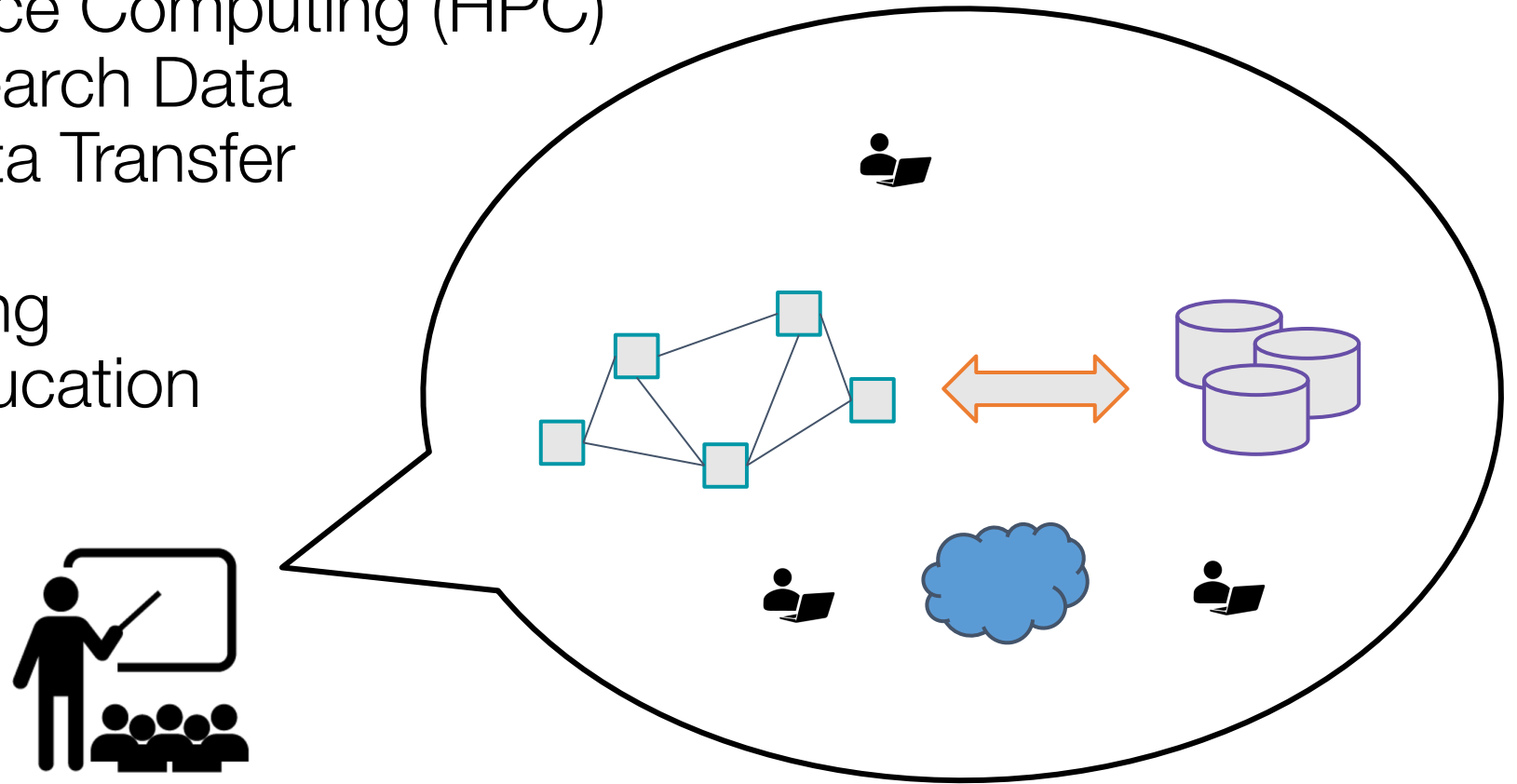


Colorado
State
University



Resources Include:

- High Performance Computing (HPC)
- Storage of Research Data
- High-Speed Data Transfer
- Data Sharing
- Cloud Computing
- Training and Education



Primarily known for: High Performance Computing (HPC)



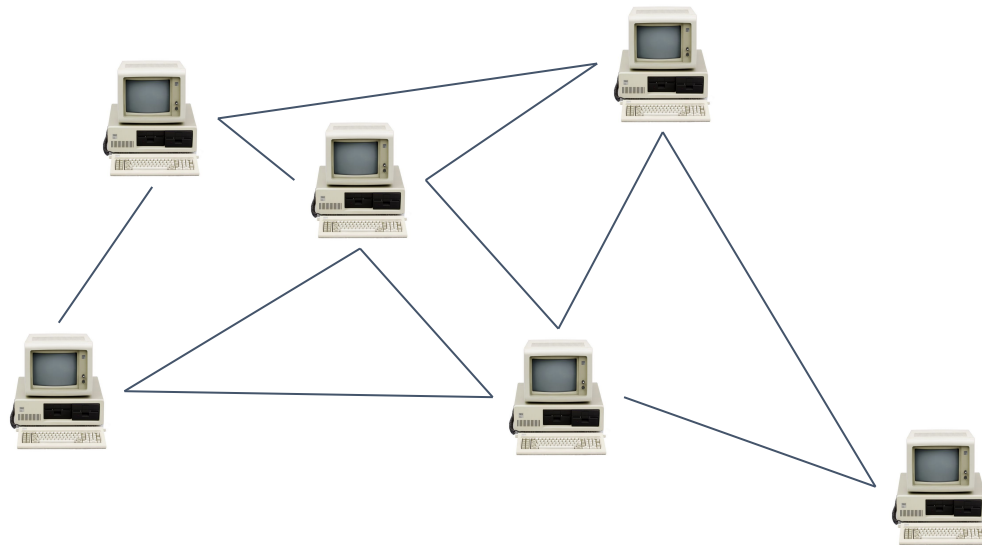
High Performance Computing (HPC) vs. Traditional Computing

- Traditional computing *generally* has access to a single processor (perhaps multiple cores)



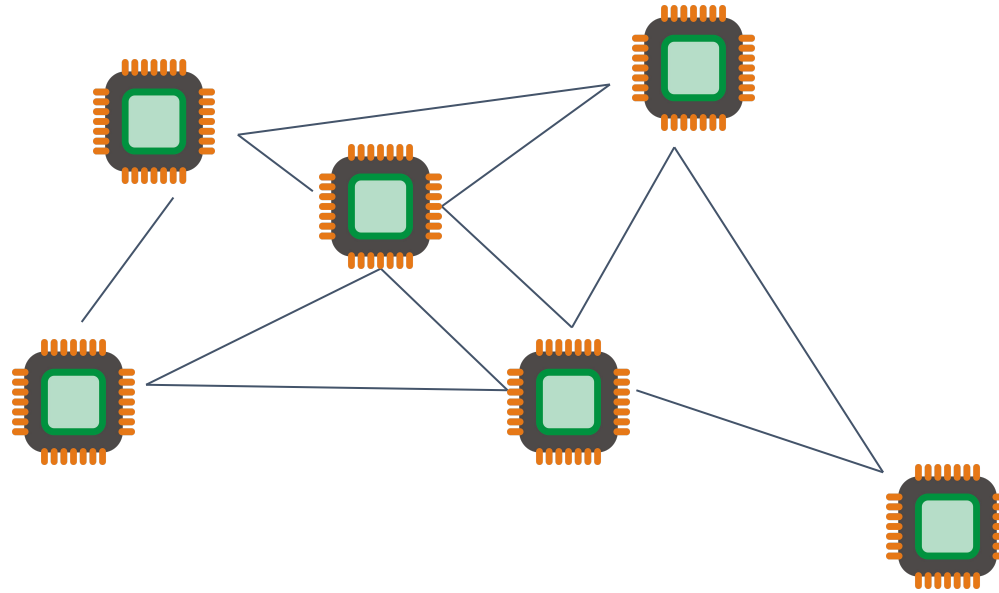
High Performance Computing (HPC) vs. Traditional Computing

- HPC is a network (or cluster) of hardware we call “nodes” linked together with high-speed interconnects



High Performance Computing (HPC) vs. Traditional Computing

- HPC is a multiprocessor environment that allows users to run jobs on several processors at once (also called parallel processing).



What can / use HPC for?

- Solving large problems that require more:
 - Memory than you have on your PC
 - cores/nodes/power than you have on your PC
- Jobs that require hardware you may not have:
 - High Performance GPU computing
 - Specific Operating System
- Visualization rendering

Puzzle Analogy

- Working on a puzzle by yourself (although enjoyable!) will take time, even if you're fast!



Puzzle Analogy

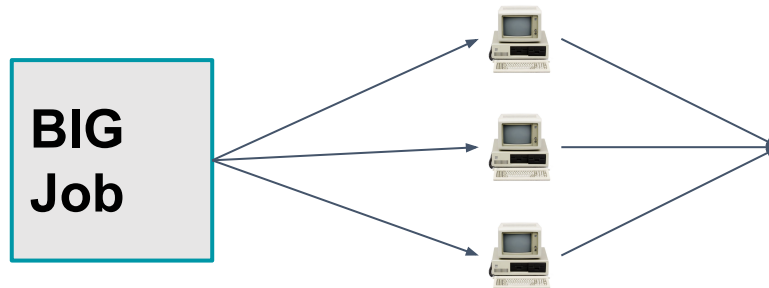
- You can speed it up by inviting friends over and enticing them with food



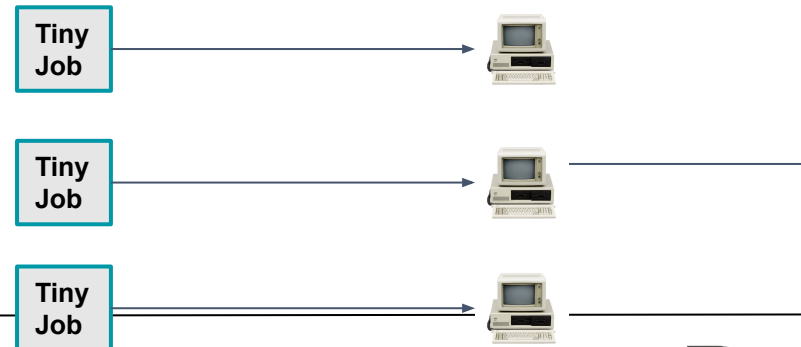
- What happens if there are too many people?

What can / use HPC for?

- Jobs that would take a long time on local machines can instead be distributed over hardware:
 - Parallelized to split up then joined (if software enabled)



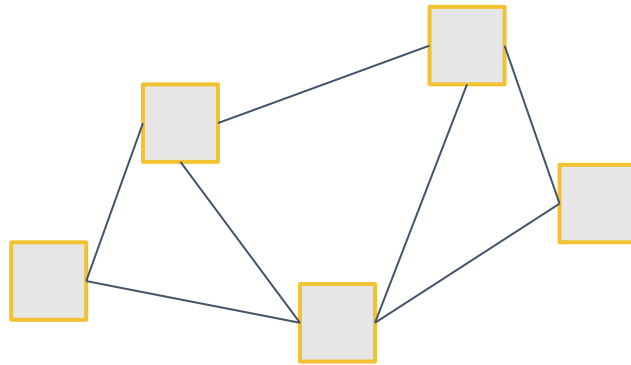
- Broken up into many serial jobs



Research Computing Resources

HPC Cluster: Alpine

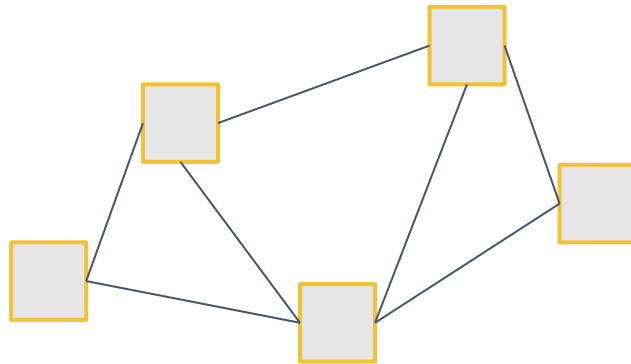
Alpine



- Alpine is the 3rd-generation HPC cluster at CURC, following:
 - Janus
 - RMACC Summit.
- Alpine is a heterogeneous cluster with pooled resources from the following institutions:
 - CU Boulder
 - Colorado State University
 - Anschutz Medical Campus.

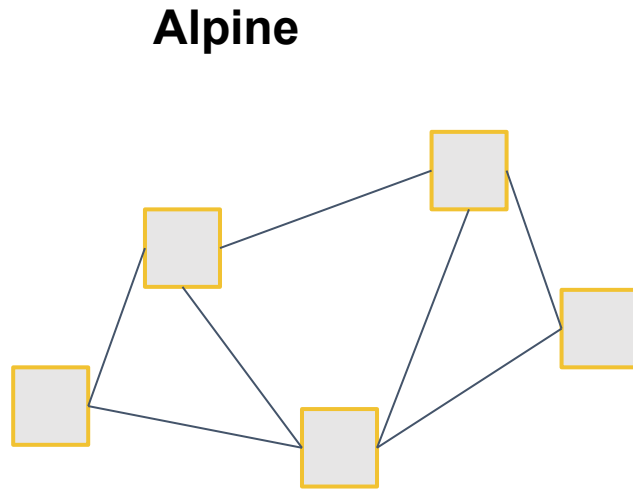
HPC Cluster: Alpine

Alpine



- Hardware on Alpine will be purchased and released in stages:
- Alpine (stage 1):
 - 64 General CPU Nodes
 - *AMD Milan, 64 Core, 3.83G RAM/Core*
 - 8 NVIDIA GPU Nodes
 - *3x NVIDIA A100 (atop General CPU node)*
 - 8 AMD GPU Nodes
 - *3x AMD MI100 (atop General CPU node)*

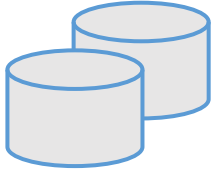
HPC Cluster: Alpine



- **Interconnect**
 - **CPU nodes:** HDR-100 InfiniBand (200Gb inter-node fabric)
 - **GPU nodes:** 2x25 Gb Ethernet +RoCE
 - nvlink compatibility in progress
 - **Scratch Storage:** 25Gb Ethernet +RoCE
- **Operating System**
 - RedHat Enterprise Linux version 8 operating system

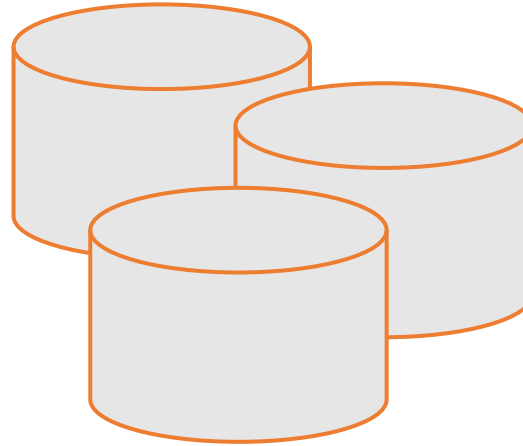
Storage at CURC

Core



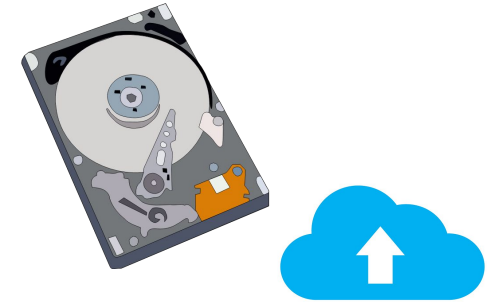
- Included with RC account
 - /home
 - /projects
 - scratch space

PetaLibrary



- Paid Service for:
 - Storage
 - Archive
 - Sharing of research data

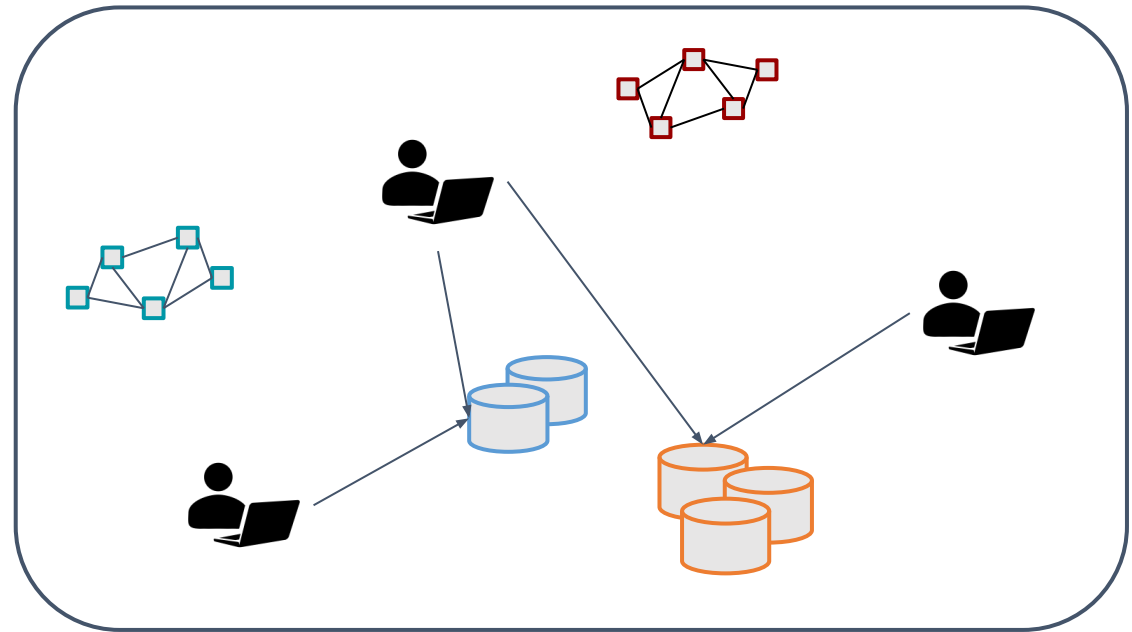
Local or Cloud



- You can download your data locally or to a variety of other cloud resources

Data Sharing: Within RC

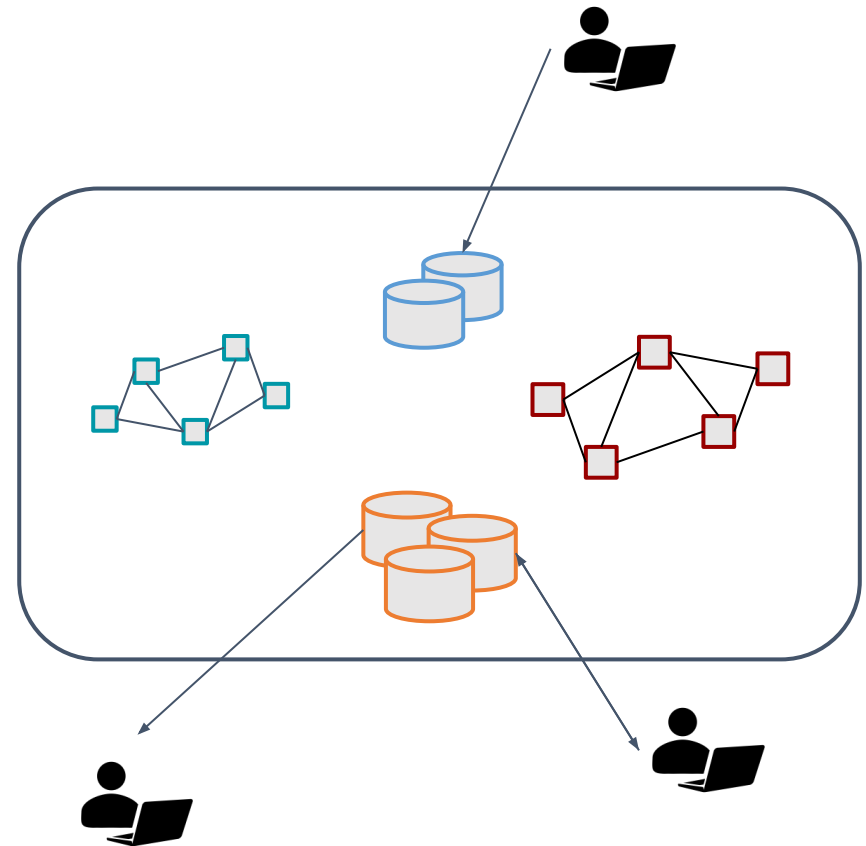
- Sharing workspaces
 - Project space
 - Scratch Space
 - PetaLibrary Space*



*If you have purchased PetaLibrary space

Data Sharing: Outside RC

- Globus (recommended):
 - GUI Web Application
 - Automates large transfers
 - Resumes failed transfers
 - Distributes large transfers across DTNs
 - Endpoints that can shared
- Data Transfer Nodes (DTN)
 - Internal CU network needed
- Command line tools
 - scp, sftp, rsync, rclone



Accessing Research Computing

How to Access RC Resources?

1. Get an RC account
2. Set up two-factor authentication with Duo
3. (Inform us of any specific needs)
4. Log in
5. Create greatness! (responsibly)

Getting an RC Account

- **University of Colorado, Boulder users and affiliates:**
 - Request an account through the RC Account request portal
 - <https://rcamp.rc.colorado.edu/accounts/account-request/create/organization>
- **Colorado State University users:**
 - Request an CSU eID if you don't have one
 - Fill out account application form
 - Duo authentication
 - Then get an RC user account
 - <https://www.acns.colostate.edu/hpc/summit-get-started/>
- **RMACC Users:**
 - Contact your local representative, if known. Email rc-help@colorado.edu
 - We'll guide you through the process

Demo: Getting an Account

- CU Boulder users and affiliates:
 - Request an account through the RC Account request portal
 - <https://rcamp.rc.colorado.edu/accounts/account-request/create/organization>

Your RC Account

Access to:

1. Alpine Cluster (also Summit)
2. Core Storage
3. PetaLibary Storage*
4. Gateways (CU Affiliates)

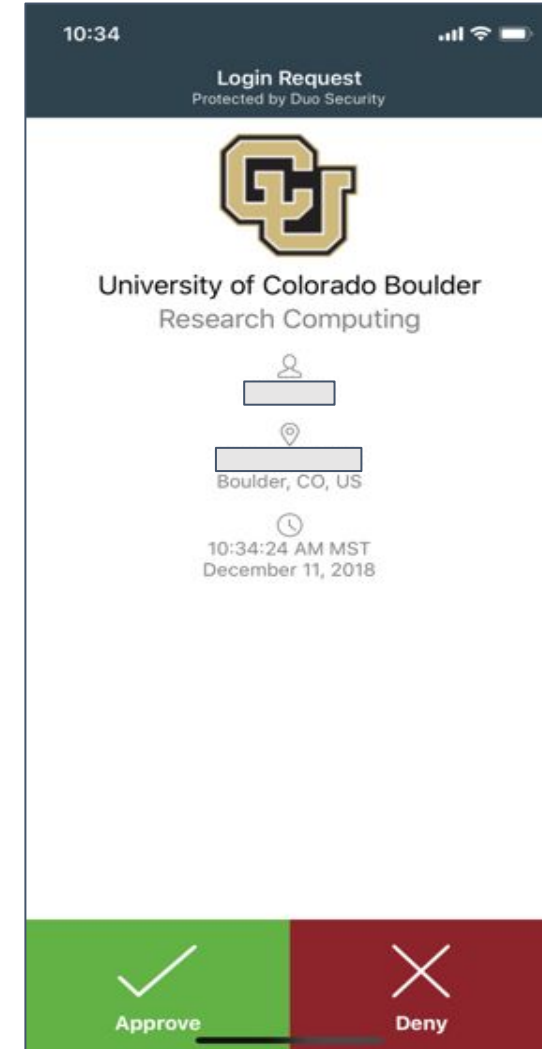
*If purchased

Two Factor Authentication (Duo)

- Provides an extra level of authentication
 - We are outside the firewall!
 - Valuable resources
 - Inviting, high-profile target
 - Lost time investigating/fixing
- Duo
 - You will receive a Duo invitation when your RC account is created.

Duo Authentication

1. **Duo smartphone app (recommended)**
2. Phone Call/Text is an alternatives
3. Physical code generator “token” available for \$20



Linux comfort level check

- On a scale from (1-10) **how familiar/comfortable are you with Linux?**
 - The command line
 - Basic commands
 - Linux filesystem
 - Navigating the filesystem

Terminal

- Mac or Linux
 - Terminal application
- Windows
 - PuTTY (recommended)
 - Powershell
- Open OnDemand (*alternative for CU affiliates*)
 - For those less familiar with Linux (ondemand.rc.colorado.edu/)

```
[user0083@tlogin1 ~]$ pwd  
/home/user0083  
[user0083@tlogin1 ~]$ █
```

Demo: Logging in via Terminal

- To login to an RC login node:

```
$ ssh <username>@login.rc.colorado.edu
```

Supply your IdentiKey password and your Duo app will alert you to confirm the login

If you're using a tutorial account (we provide password):

```
$ ssh <tutorial_user>@tlogin1.rc.colorado.edu
```

Demo: logging in with OnDemand

CURC Open OnDemand is a browser based, integrated, single access point for all of your HPC resources at CU Research Computing.

- Visit <https://ondemand.rc.colorado.edu>.
- CU Boulder *only* at the moment

Logging In

- It's important to note that you are ***NOT*** logging into any specific resource, Summit, etc.
- When you log in, you land on our ***login nodes***
- From ***there***, you can access our other resources:
 - Alpine
 - Summit
 - Blanca

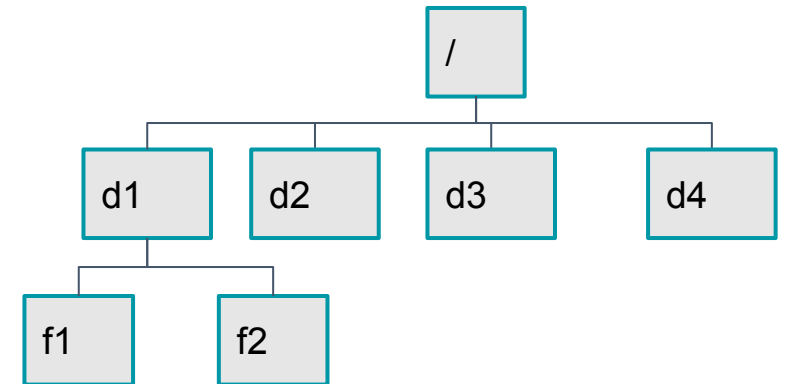
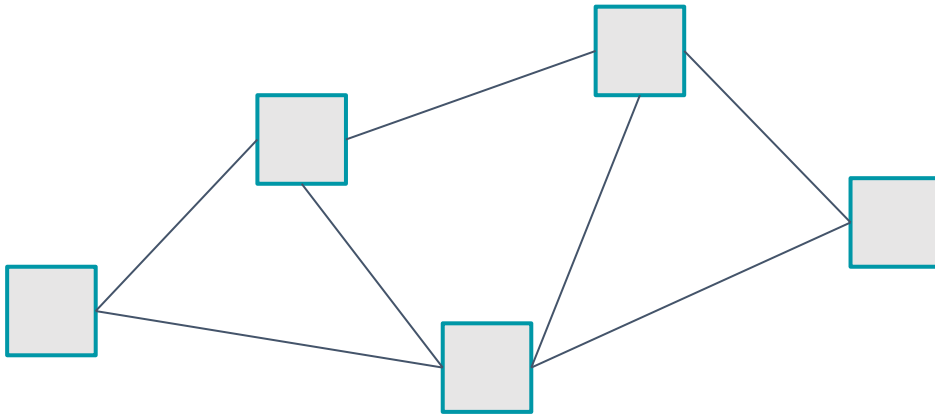
Navigating Research Computing

Node

- One computing server
- Physical hardware
- Work together in parallel

File System

- The basic tree-like layout
- From most nodes* you have access to the entire file system

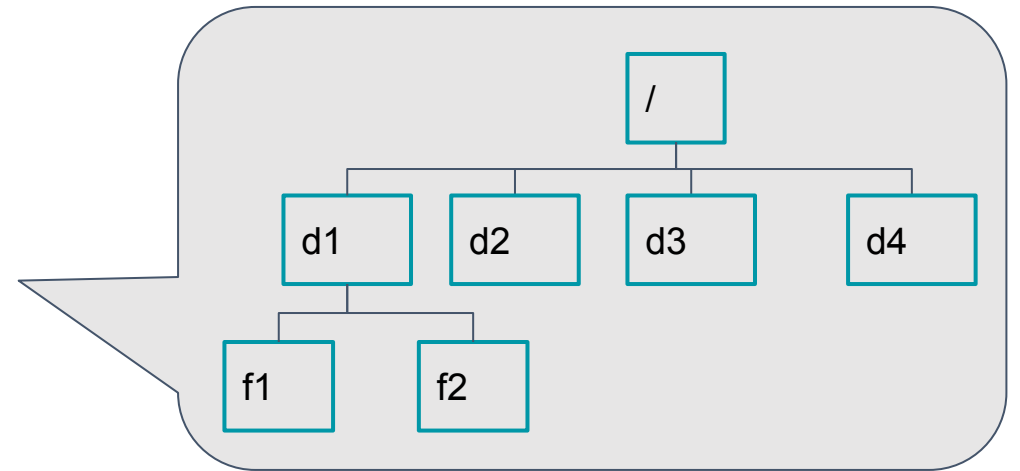
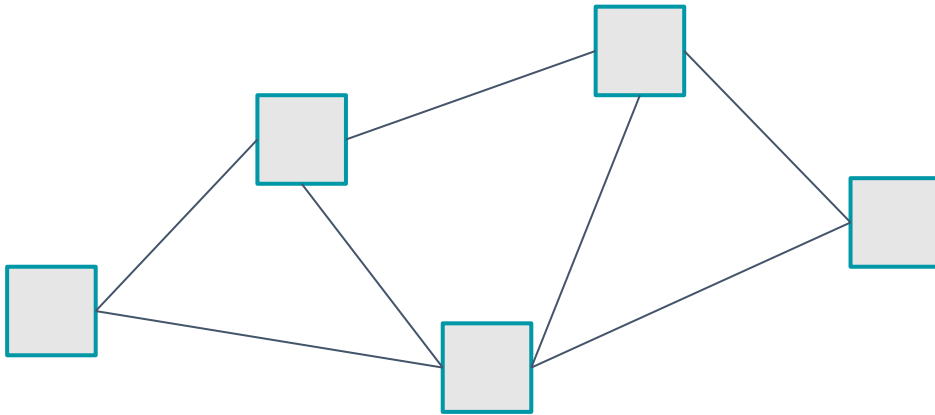


Node

- One computing server
- Physical hardware
- Work together in parallel

File System

- The basic tree-like layout
- From most nodes* you have access to the entire file system



Node Types

Login

- Where you log in to
- For editing code, job submission
- No heavy computation

Ex. edit job script

Compile

- Where you compile code, install packages
- Explore the Summit software environment
- Edit code, submit jobs
- No heavy computation

Ex. Install python libs

Compute

- Where scheduled jobs run
- Intended for heavy computation

Ex. Running Matlab

A note on Alpine Compile Nodes

- If you have used Summit in the past, compile nodes work *slightly* differently:
 - Instead of having dedicated hardware (2 nodes) which are oversubscribed for users to ssh into
 - Alpine's **acompile** command starts an interactive job which users can compile in which provides the following benefits:
 - Users can request specific resources (i.e. more cores to compile with)
 - Limits dedicated hardware set aside
 - Can't accidentally run full workflows
- Because of this you *may* wait longer for an acompile job

Demo: Exploring Nodes

- Once logged in, type:

\$ compile

To log in to an Alpine compile (or head) node.

- Once on a compile node, type:

\$ module avail

To list currently available software

Filesystem Structure

/home (2GB)

- Scripts, Code, Small, important files/directories
- Not for sharing files or job output

Ex .bashrc

/projects (250GB)

- Code/files/libraries
- Software you are installing
- Sharing files
- Not for job output

Ex. Shared job scripts

Scratch (10TB)

- Output from running jobs
- Large files/datasets
- Sharing files
- Cluster specific
- Not for long term storage

Ex. Data

Demo: Exploring the Filesystem

- Once logged in use the following commands to navigate to your different workspaces

```
$ cd /home/<user>
```

```
$ cd /projects/<user>
```

```
$ cd /scratch/alpine/<user>
```

Using RC Resources

- We have:
 - Logged in
 - Explored nodes
 - Explored filesystem
- How do we actually *use* the computing resources?

Running a Job

Jobs

What is a “***job***”?

- Work for the cluster to perform on
- Has a unique ID

1. Batch jobs

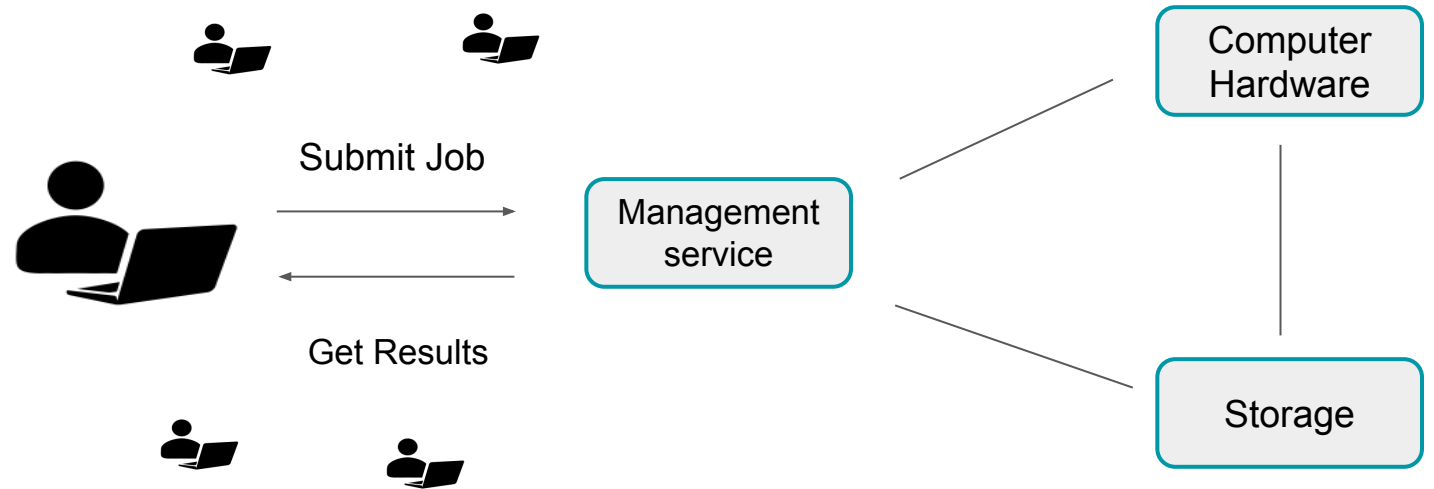
- Submit job script which will be executed when resources are available
 - Create script containing information about the job
 - Submit the job file to a queue

2. Interactive jobs

- Work interactively at the command line of a compute node

Job Scheduling

- CURC Clusters are shared resources, jobs are:
 - Submitted to a queue
 - When the required resources become available, the scheduler determines which set of nodes to use
 - Executes your job



SLURM

- **S**imple **L**inux **U**tility for **R**esource **M**anagement
- Through SLURM users can:
 - Schedule jobs on specific compute resources
 - Run jobs interactively or hands off
 - Query job statistics

Job Script: 3 main parts

1. Directives

- Specify resource requirements

2. Software

- Because jobs run on a different node than from where you submitted...
- ...software that is needed must be loaded via the job script

3. User scripting

- the actual user scripting that will execute when the job runs

```
#!/bin/bash

## Directives
#SBATCH --<resource>=<amount>

## Software
module purge

## User Scripting
hostname # example bash command
```

Slurm Options (directives)

```
#SBATCH <options>          sbatch <options>
```

- Allocation: `--account=<account_no>`
 - Partition: `--partition=<partition_name>`
 - Sending emails: `--mail-type=<type>`
 - Output file: `--output=<file name>` (%j gives you job id)
 - Number of nodes: `--nodes=<nodes>`
 - Number of tasks: `--ntasks=<processes>`
 - Quality of service: `--qos=<qos>`
 - Reservation: `--reservation=<name>`
 - Wall time: `--time=<wall time>`
 - Job Name: `--job-name=<jobname>` ...etc...
- FYI: You do NOT actually type <> above – this designates something specific you as a user must enter about your job

Slurm Options (directives)

- There are **MANY** slurm directives, most of which are not required
 - See all options at <http://slurm.schedmd.com/sbatch.html>
- We will focus on the most important ones:
 - **Partition:** Nodes with the same hardware configuration
 - **Wall time:** max time your job will run for
 - **Node count:** # of nodes requested
 - **Core count:** # of cores requested
 - **Output file:** name of output file (variable %j holds job id)

Alpine Partitions

Partition	Description	# of nodes	RAM/core (GB)	cores/node	GPUs/node
amilan-<institution>	General Compute Node: AMD Milan	64	3.83	64	0
ami100-<institution>	GPU Node: 3x AMD MI100	8	3.83	64	3
aa100-<institution>	GPU Node: 3x Nvidia A100	8	3.83	64	3

Institutions:

- **ucb**: University of Colorado, Boulder
- **csu**: Colorado State University
- **amc**: Anschutz Medical Campus

Demo: Writing a simple job `hostname.sh`

- Submit a slurm job with the following instructions:
 1. The job will be submitted from a bash script named **hostname.sh**
 2. The job will run on **1 node**
 3. We will request **1 minute of wall time**
 4. Run on the **amilan-ucb partition**
 5. Output should **contain job id**
 6. The job should run the **Unix “hostname” command**

Demo: Writing a simple job (hostname)

- Set up batch job boilerplate
- Using text editor (vim or nano), create a file called **hostname.sh** with the following text:

```
#!/bin/bash

## Directives
#SBATCH --<resource>=<amount>

## Software
module purge

## User Scripting
```


Demo: Writing a simple job (hostname)

- Directives:
 - 1 node
 - 1 minute wall time
 - “amilan-ucb” partition
 - Output should contain job id
- Number of nodes:
`--nodes=<nodes>`
- Wall time:
`--time=<wall time>`
- Partition:
`--partition=<partition_name>`
- Output file:
`--output=<file name>`
(%j gives you job id)

Demo: Writing a simple job (hostname)

- Software:
 - Do we need any software? (modules)
- User script:
 - What command do we want to run?

Demo: Writing a simple job (hostname)

```
#!/bin/bash
#SBATCH --nodes=1                # Number of requested nodes
#SBATCH --time=0:01:00          # Max wall time
#SBATCH --partition=amilan-ucb  # Specify Summit haswell nodes
#SBATCH --output=hostname_%j.out # Rename standard output file

# purge all existing modules
module purge

hostname
```

Submitting a Job

1. Load up the slurm Alpine module (default is still Summit)

```
$ module load slurm/alpine
```

2. Submit the job:

```
$ sbatch <script-name>.sh
```

3. Check output

```
$ cat <output-name>.out
```

Review: Learning Goals

1. What is CU Research Computing (CURC)?
2. Understand Basic Resources (Alpine cluster)
3. Getting an account & logging in
4. Navigate the RC system
5. Running a job
6. Help!

Help! I'm stuck, where do I go?

- **Documentation:** curc.readthedocs.io/
- **Trainings with Center for Research Data and Digital Scholarship (CRDDS):**
<https://www.colorado.edu/crdds/>
- **Helpdesk:** rc-help@colorado.edu

Questions

Survey and feedback

<http://tinyurl.com/curc-survey18>