Filesystems and Storage on CU Research Computing Resources

Research Computing
UNIVERSITY OF COLORADO **BOULDER**

Be Boulder.

Formalities

- Andrew Monaghan (Slides developed by Mea Trahan)
- *Email: Andrew.Monaghan@Colorado.edu*
- *RC Homepage: https://www.colorado.edu/rc*
- *RC Email: rc-help@colorado.edu*

- Slides available for download at:
- *https://github.com/ResearchComputing/Filesystems_And_Storage_Fall_2022*

Outline

- Common Terms
- Overview of CURC computing resources
- Overview of CURC storage
- Alpine architecture and filesystems
- Petalibrary
- Data transfers and tools

Common Terms

Term	Meaning
HPC	"High Performance Computing" – infrastructure that can solve complex problems quickly.
Core	A single processing unit on a **CPU** that can execute one task at-a-time.
CPU	"Central Processing Unit": Component of a computer that carries out tasks (data I/O, arithmetic, interpreting instructions); Usually has multiple **cores.**
Memory	"RAM": Short-term memory on a computer, where data is stored while being processed by **CPU**(s).
Node	A single computer within a cluster that has its own **memory** and **CPU**(s).
Fabric	High-speed networking cable that connects **nodes** on a **cluster.**
Filesystem	A framework that defines how files are named, stored and accessed from **storage**.
Storage	Device that stores files persistently. **Filesystems** manage storage operations.
Cluster (Supercomputer)	A specific **HPC** platform such as CURC's Alpine or Summit. Typical features of a *cluster*: lots of **nodes** with multiple **CPUs**, many **cores**, and substantial memory, connected by performant **fabric** and common **filesystems** and **storage.**

Overview of CURC computing resources

System	Description	In service	Types of resources	#CPU cores
Alpine	**CURC's primary supercomputing**	**2022-**	**CPU, GPU, high-mem**	**15,184 + …**
Summit	Supercomputer; predecessor to Alpine	2017-2022	CPU, GPU, high-mem, KNL	17,200
Blanca	"Condo" cluster; groups buy dedicated nodes	2015-	CPU, GPU, high-mem	8,952
Viz	Gpu-accelerated cluster for data visualization	2019-	CPU, GPU	192
CUmulus	Cloud-cluster for databases, web apps, workflow mgt.	2020-	CPU	244

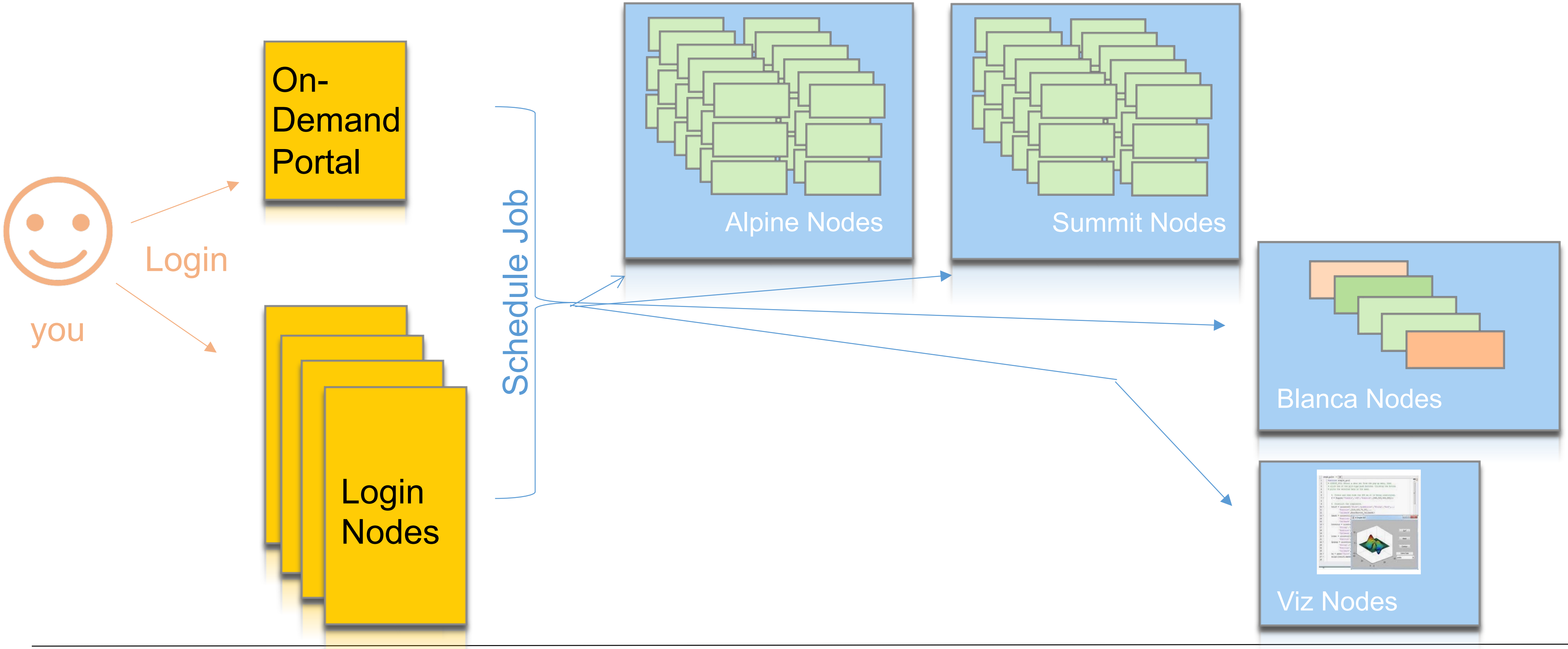
Overview of your CURC directories

- 3 major user directories
  - Home – Used for reusable job scripts, setting files, and other important small files.
  - Projects – Used for application and small datasets.
  - Scratch – Work directory. Used with jobs for highspeed access to data or output.
- Table:

	Directory	Capacity	Snapshots	Purge
Home	/home/$USER	2 GB	2 hours for 7 days	Never
Projects	/projects/$USER	250 GB	6 hours for 7 days	Never
Scratch*	/scratch/alpine/$USER	10 TB	(none)	90 days

** ”scratch” storage is system-specific – example above is for Alpine*

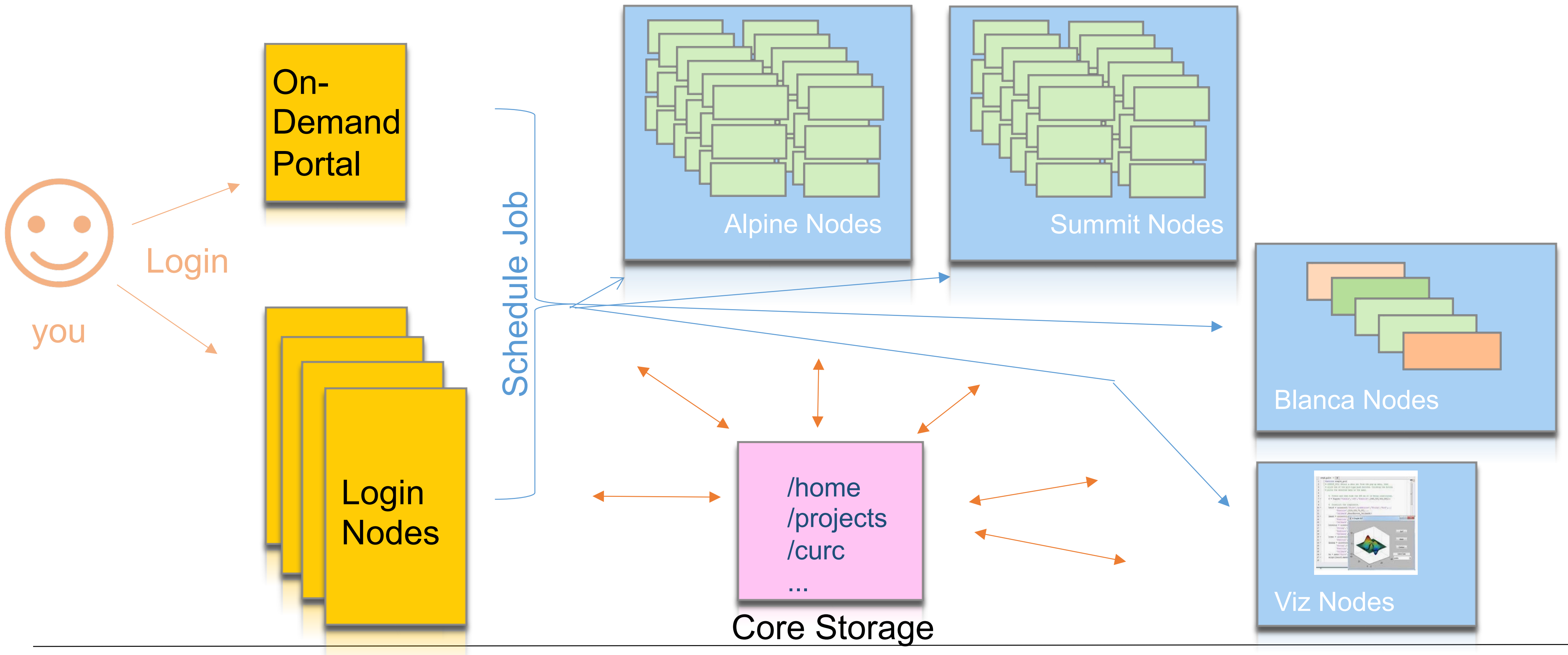
RC's HPC network



RC's filesystems

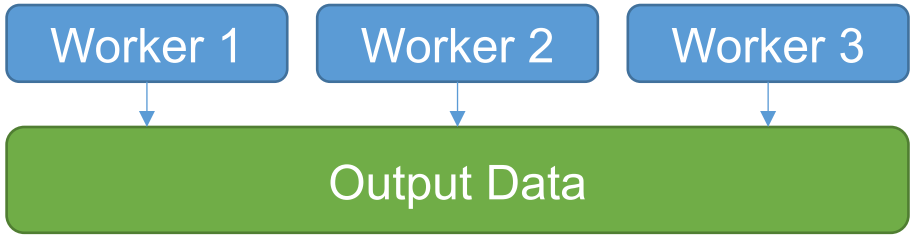
- To reduce the amount of complexity for an end users, CURC uses a shared file server – "core storage" -- to manage user related storage.
  - Contains `/home, /projects`
  - Contains all shared software and the module stack (`/curc`).
  - Every node or login VM is connected to this resource allow user to easily manage their files.
  - Non-Parallel IO – not designed for performant read/write access

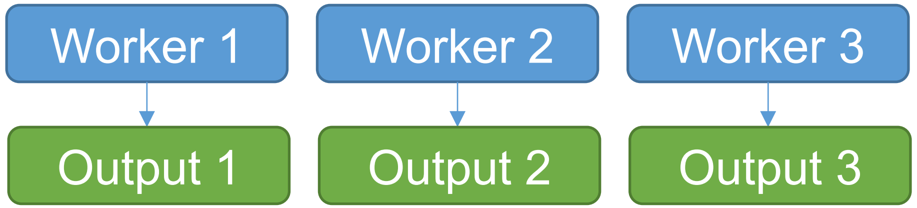
RC's HPC network *(incl. core storage)*



Problems with I/O and threads

- Suppose someone is computing with 120 threads and needs to write their data to a file system…
- Single File:
  - Many threads means that applications may idle waiting for free resources.
  - Nonlocking I/O may cause corruption of data.
- Many Files:
  - Separate file writes may lead to issues with the filesystem's metadata service.
- So what do we do?





CURC’s Parallel filesystems

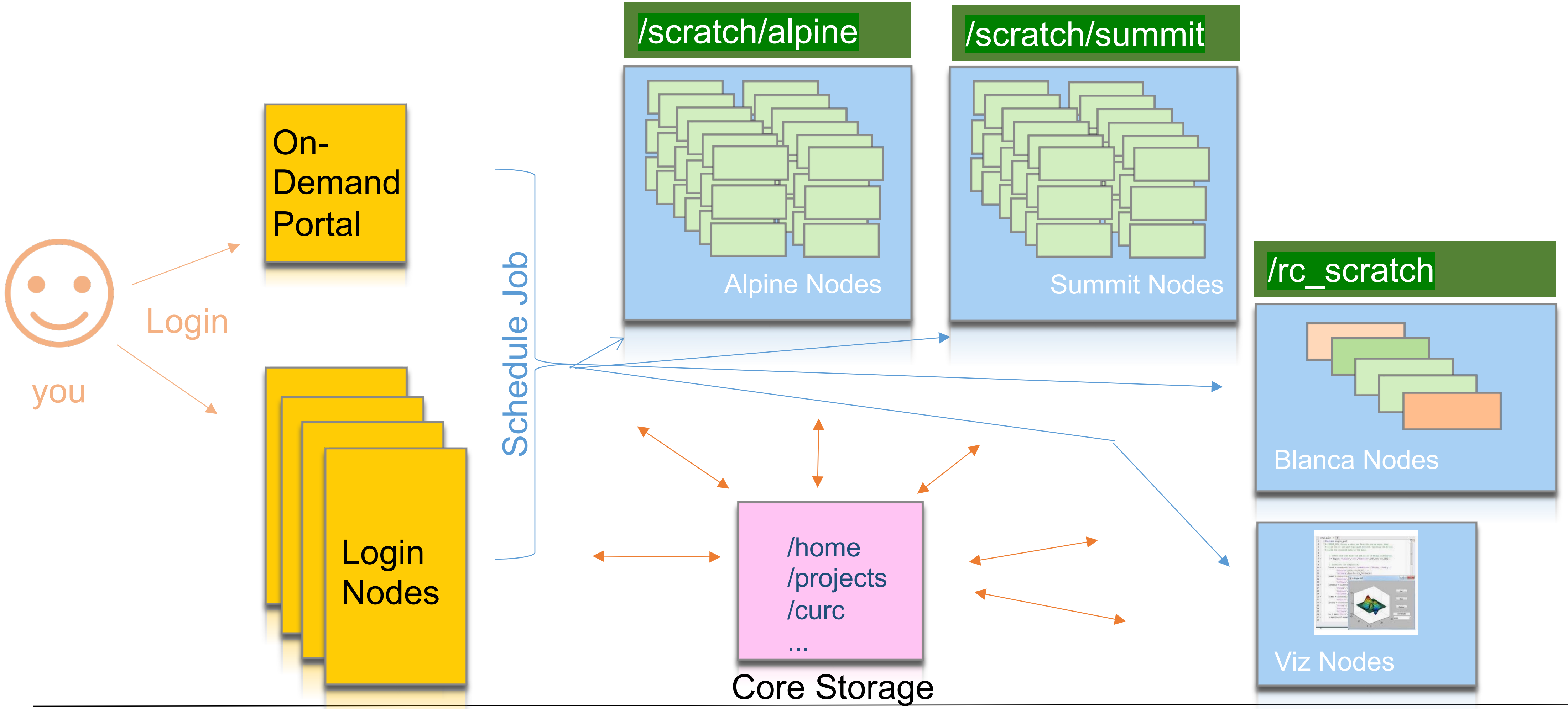
There is an additional parallel file system available on Blanca, Summit and Alpine! :

- **“Scratch”** filesystem. Typical setup:
  - Spinning disk platters rated at 12 Gb/s
  - GPFS File System for parallel I/O w/ 32 Clients and 4 Servers
    - Distributed metadata to avoid bottlenecking
    - Consistent chunking allows for parallel I/O
  - Locally mounted to each node on a specific cluster
- Default is 10 TB of scratch storage/user; can be expanded upon with request.
  - Files purged 90 days from creation date.
  - Technically shared among all users

Parallel Filesystem

- Normal application I/O is usually lacking the ability to leverage a parallel file system for performance
  - On Alpine you will naturally get an I/O performance boost when using scratch.
- Need to utilize specialized software libraries
- MPIIO
  - Middle wear, requires modification of code for efficient usage.
- HDF5
  - High level, use a HDF5 dataset
- NETCDF
  - High level use a Netcdf dataset

RC's HPC network *(incl. parallel f.s.)*



Other fast storage: Local Node SSDs

- Alpine/Summit/Blanca nodes also have100-800 GB of local node SSD storage.
- These SSDs are not shared among nodes so must move files over within job.
- No Cooperative Parallel I/O, but fast because solid state
- Located on each node at `/scratch/local`
- Can point I/O to `$SLURM_SCRATCH` `(/scratch/local/<jobid>) during job (directory purged at end of job)`

Permanent large-scale storage: Petalibrary

- Research Computing offers a subsidized but **paid**, long-term storage solution closely coupled with RC resources.
- Petalibrary
  - Large-scale subsidized storage solution
  - Enterprise Grade
  - RC Staff supported with assistance on transfer strategies
  - Available in several flavors:
    - Active – Disk
    - Archival – Tape
    - Active Storage with Archive copy

Hardware Specifications

- Active Storage
  - Spinning disk platters for frequent reads and writes
  - ZFS filesystem
  - RAID-6 file protection
  - Allocations located at: `/pl/active/`
  - Mounted on all clusters + login nodes + data transfer nodes (DTNs)
- Archive Storage
  - Presently tape storage for infrequent reads and writes
  - Currently being replaced with more cost-effective spinning disk storage
  - Allocations located at: `/pl/archive/`
  - Mounted on login nodes + data transfer nodes (DTNs)

Checking your storage limits:

- *curc-quota* – Research computing tool to monitor disk usage.
  - Provides detailed summary of your core storage
  - Provides detailed summary of scratch space on compile and compute nodes
  - Also lists current capacity of all Petalibrary allocations you have access to

```
[userXXXX@login12 ~]$ module load curc-quota
[userXXXX@login12 ~]$ curc-quota
```

Data Transfers

- Data transfers are usually handled by one of 2 methods:
- Globus
  - By far the most stable and recommended way for data transfers
  - Fast transfers
  - Transfers continue if a user disconnects
  - Web GUI option or Globus Connect Personal
- SCP/SFTP/RSYNC
  - Secure Copy and Secure File Transfer Protocol
  - Straightforward method of transferring data
  - Can transfer through login nodes _or_ through data transfer nodes (recommended)
- https://curc.readthedocs.io/en/latest/compute/data-transfer.html

globus

Data Transfers (2)

- Less common methods of transferring data…
- sshfs and SMB
  - Mounting the RC filesystem to your drive remotely!
  - Single sign in for multiple data transfers
  - Great when needing to repeatedly access files on RC Resources
  - Less Performant
- rclone
  - Useful for file transfers across very heterogenious systems (e.g., Google Drive to CURC)

Questions?

Thank you!

- Please fill out the survey: http://tinyurl.com/curc-survey18
- Contact information: rc-help@Colorado.edu
- Slides: *https://github.com/ResearchComputing/Filesystems_And_Storage_Fall_2022*