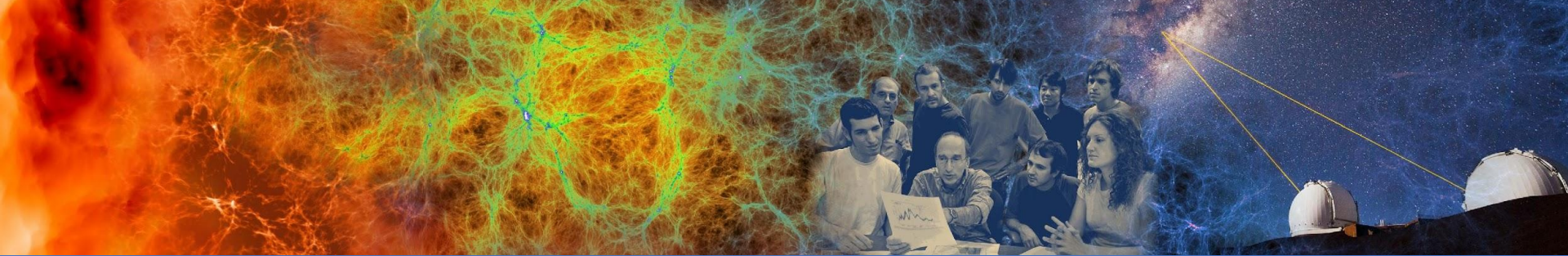


Data Storage and Sharing Best Practices



New User Training
September 28, 2022

Lisa Gerhardt
Data And Analytics Group



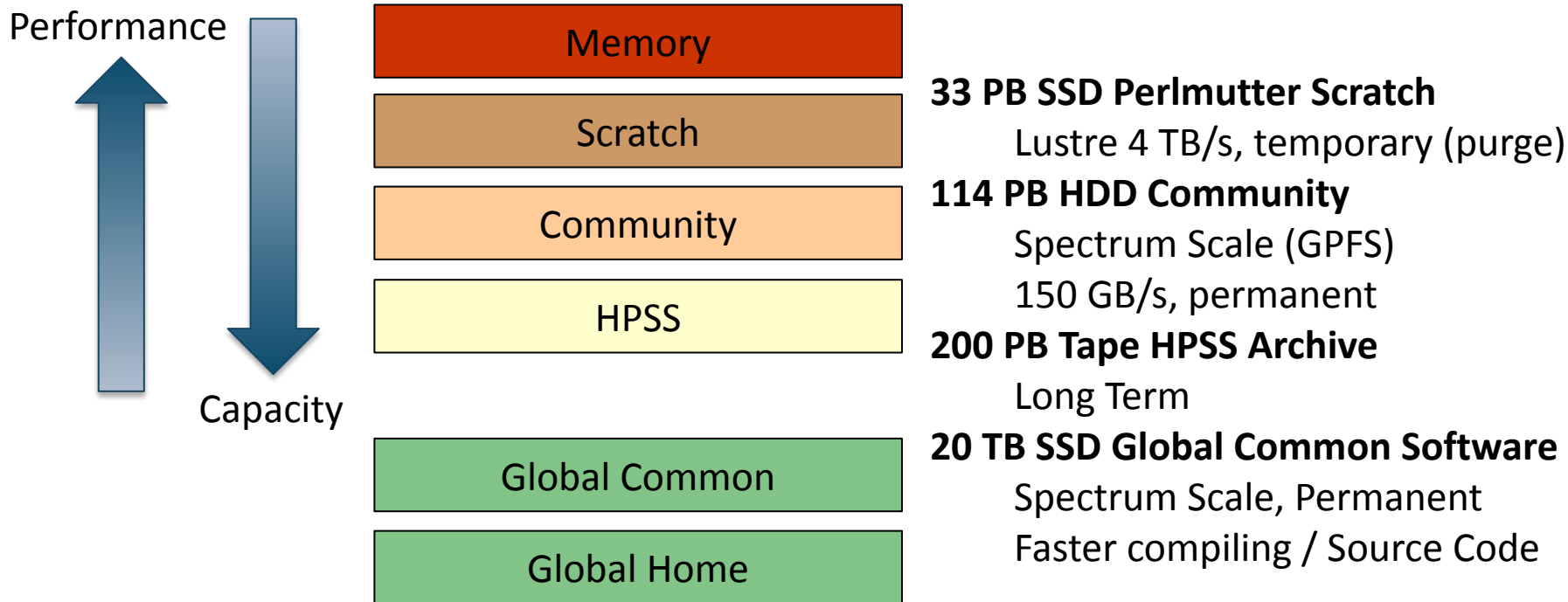
Data Storage Best Practices

NERSC Data Storage Policy

- NERSC provides its users with the means to store, manage, and share their research data products
- NERSC resources are intended for users with *active* allocations. It is strongly recommended that if you no longer have an allocation at NERSC, you transfer your data somewhere that you have access.
- PIs can request the modification, deletion, or transfer to another NERSC file system of data associated with their NERSC award
- Files are protected only using UNIX file permissions based on Iris user and group IDs. It is the user's responsibility to ensure that file permissions and umasks are set to match their needs
- Users have ultimate responsibility for managing their data

<https://docs.nersc.gov/policies/data-policy/policy/>

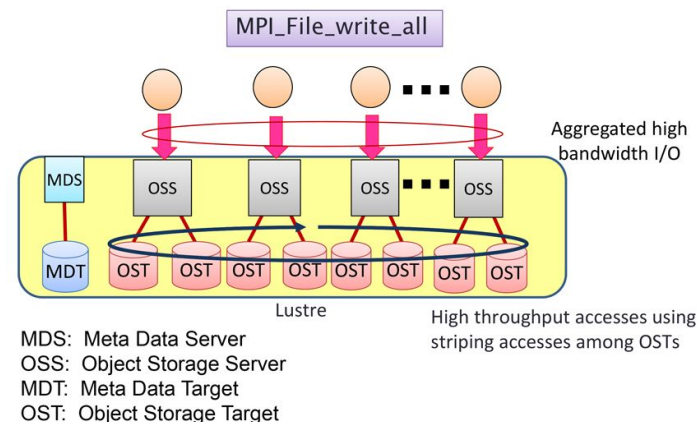
Simplified NERSC File Systems



<https://docs.nersc.gov/filesystems/>

Perlmutter Scratch

- **Lustre**, one of the most successful/mature HPC FS. This is where to store data being actively read or written by jobs on computes
- Directories are user-readable and writable by default
- Purged! Back up any important data
- Quotas are 20TB (soft) and 30TB (hard). After you exceed the hard quota, you will not be able to write any more data to the file system



Using MPI-I/O on Lustre[1]

“Scratch”: Optimize Performance with Striping

Scratch Striping Recommendations

- By default data on 1 OST, ideal for small files and file-per-process IO
- Single shared-file I/O should be striped according to its size
- Helper scripts

`stripe_small, stripe_medium, stripe_large`

- Manually query with
`lfs getstripe <path>`
- Set striping on a directory
 - New files will automatically pick it up
 - Copy files in to inherit the striping

	Single Shared-File I/O	File per Process
File size (GB)	command	
< 1	keep default striping	keep default striping
1 - 10	<code>stripe_small</code>	keep default striping
10 - 100	<code>stripe_medium</code>	keep default striping
> 100	<code>stripe_large</code>	keep default striping
> 1000	<code>stripe_large</code>	<code>stripe_large</code>

Community File System

- For: large datasets that you need for a longer period
- Set up for sharing with group read permissions by default
- Not for intensive I/O - use Scratch instead
- Data is never purged. Snapshots. Usage managed by quotas
- Projects can split their space allocations between multiple directories and give **separate** working groups **separate** quotas
 - Environment variable \$CFS points to /global/cfs/cdirs

<https://docs.nersc.gov/filesystems/community/>

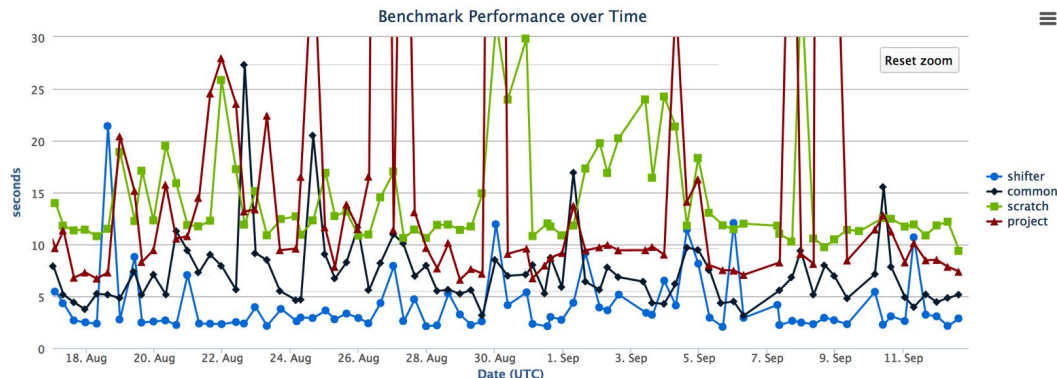
HPSS

- For: data from your finished paper, raw data you might need in case of emergency, really hard to generate data
- HPSS is tape!
 - Data first hits a spinning disk cache and gets migrated to tapes
 - Files can end up spread all over, so use `htar` to aggregate into bundles of 100 GB - 2 TB
 - Archive the way you intend to retrieve the data
- Quotas are controlled in Iris. If you're a member of multiple projects you can adjust the percentage you want charged to each

<https://docs.nersc.gov/filesystems/archive/>

“Global Common”: Software Filesystem

- For: software stacks - Why? Library load performance



- Group writable directories similar to community, but with a smaller quota, `/global/common/software/<projectname>`
 - Write from login node; read-only on compute node
- Smaller block size for faster compiles than project

Home Directories

- For: source files, compiling, configuration files
- 20G quota
- Not intended for intensive I/O (e.g. application I/O) - use Scratch instead
- Backed up monthly by HPSS
- Snapshots are also available e.g. my homedir is at `/global/homes/.snapshots/2022-06-14/e/elvis`

Data Dashboard in my.nersc.gov

Data Dashboard

Showing disk space and inode usage for project directories at NERSC to which you have access as PI, PI proxy, or user (includes /project, /p

CAL directory in /project



carver directory in /project



coe directory in /project



cosmo directory in /project



crcns directory in /project



das directory in /project



Data Dashboard: Usage Reports

Data Dashboard

Showing disk space and inode usage for project directories at NERSC to which you have access as PI, PI proxy, or user (includes /project, /projecta, and /projectb/sandbox)

CAL directory in /project

Toggle Usage Details

My Files and Dirs

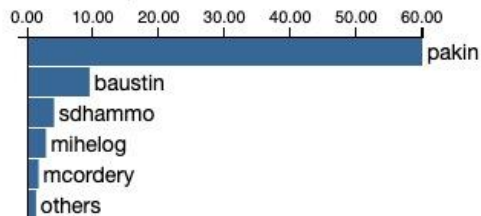
Browse



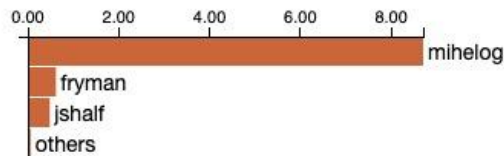
Data as of Thu Jun 20 2019 23:59:59 GMT-0700 (Pacific Daylight Time)

Breakdown of allocation usage:

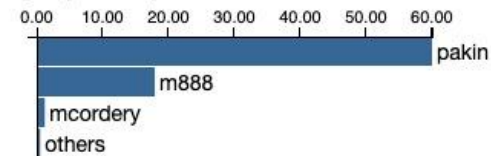
user % of space allocation



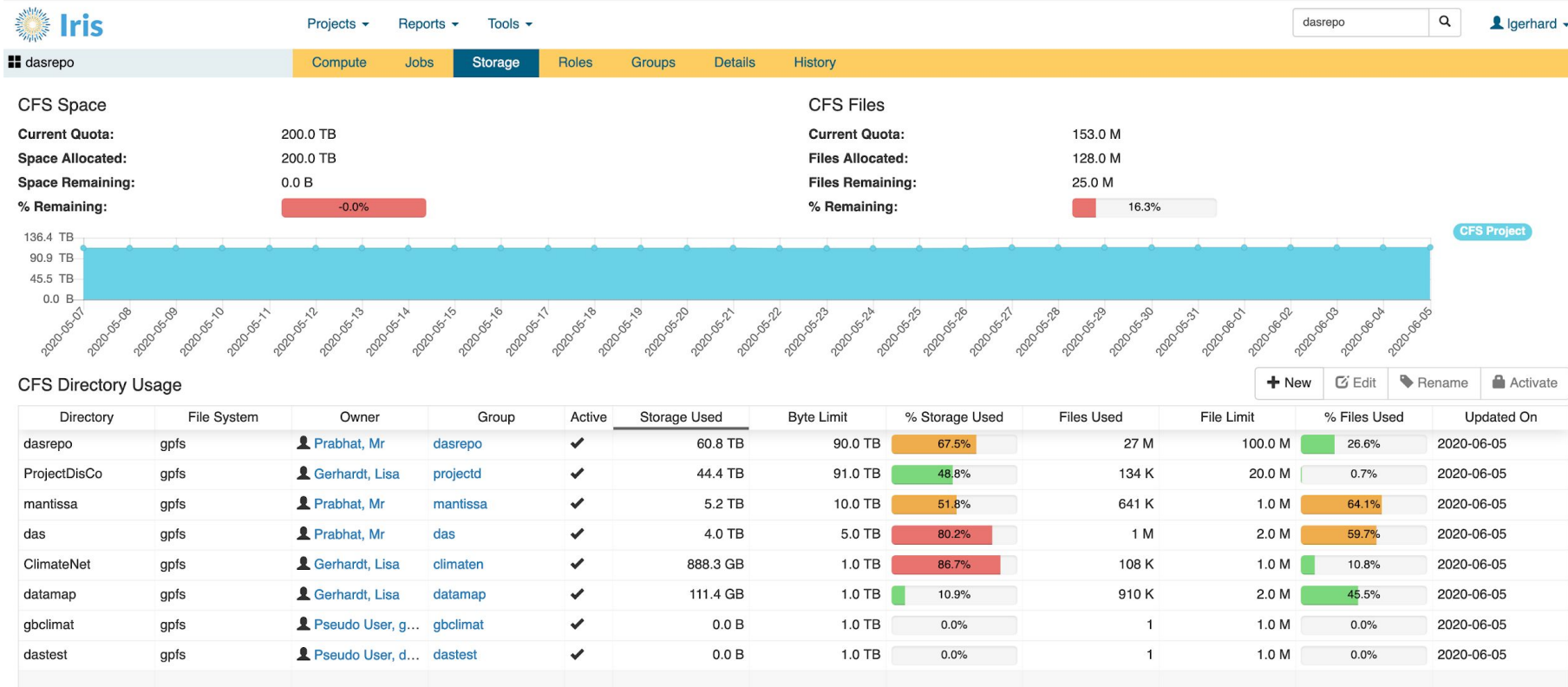
user % of inode allocation



group % of space allocation



Adjusting Quotas in IRIS



PI Toolbox: my.nersc.gov/pitools/

PI Toolbox

Jump to: das directory in cfs Make entire project dir group-readable

Path: /global/cfs/cd Show

☐ Select All (But sele

☐ Refresh file list

Select **Name** **Size** **Date** **Permissions**

<input type="checkbox"/>	Parent Direct					
<input type="checkbox"/>	.ipynb					
<input type="checkbox"/>	DaskE					
<input checked="" type="checkbox"/>	MODS					
<input type="checkbox"/>	agrein					
<input type="checkbox"/>	backu					
<input type="checkbox"/>	canon	canon	das	4096	Aug 21 10:32	drwxrwx---
<input type="checkbox"/>	certs.nersc.gov	dastest	das	4096	Sep 9 16:03	drwxrwxr-x
<input type="checkbox"/>	dfulton	dfulton	das	4096	Aug 21 11:14	drwxrwxr-x

Owning group can:

☐ Read (r)

☐ Write (w)

☐ Execute file, enter directory

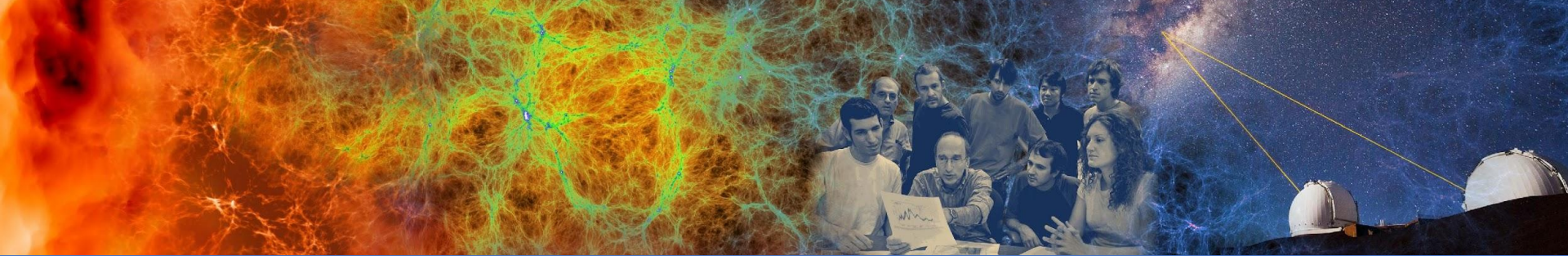
☒ Make directory group openable and executable files group executable (X)

☐ Execute binary file as member of owning group and force new items in directory to be owned by the group (s)

☐ Execute binary file normally, as member of user's default group (x)

☐ I want to apply these permissions recursively

Cancel Submit request



Data Sharing Best Practices

Sharing Inside of NERSC

- **Community File System: CFS**

- Every project has at least one directory that has permissions set up to be group writable and readable
- PI Toolbox (my.nersc.gov/pitools/) can manage permissions

- **HPSS Project Directories**

- Directories in HPSS with group writable and readable permissions

- **Collaboration Accounts**

- Account tied to a group instead of an individual user. Access is controlled by the project's PI. Useful for managing shared datasets, running shared workloads

- **Scratch**

- User who desire to share data on scratch can do it by adjusting Linux permissions
 - Only share read access. If you want to allow writes, we recommend using a collaboration account instead
 - `chgrp -R <project_name> $SCRATCH; chmod g+rX $SCRATCH` (read only)

- **give / take**

- Mechanism to give single files to any other NERSC user

Sharing with External Collaborators

- **Public HTML access**
 - Project specific area can be created:
 - `/global/cfs/cdirs/<yourproject>/www`
 - These are available for public access under the URL:
 - `https://portal.nersc.gov/project/<yourproject>/`
- **Science Gateways** (docs.nersc.gov/services/science-gateways/)
 - Web portals allow you to interface with your data and computation at NERSC
 - For more sophisticated web applications: **Spin** (docs.nersc.gov/services/spin/)
- **Globus Sharing** (docs.nersc.gov/services/globus/#globus-sharing)
 - Projects can set up read-only endpoints for sharing data with certain Globus users
 - Excellent way to share large volumes of data, can be incorporated into web pages

NERSC's Dedicated Data Transfer Nodes

- **Data Transfer Nodes** (DTNs, <https://docs.nersc.gov/systems/dtn/>)
 - Dedicated servers for moving data at NERSC (dtnXX.nersc.gov)
 - Servers include high-bandwidth network interfaces & are tuned for efficient data transfers
 - Monitored bandwidth capacity between NERSC & other major facilities such as ORNL, ANL, BNL, SLAC...
 - Direct access to Community, HPSS Archive, and Cori Scratch
- Use NERSC DTNs to move large volumes of data in and out of NERSC or between NERSC systems
- User Perlmutter Login nodes for data transfers to Perlmutter Scratch

Globus

The **recommended** tool for moving data in & out of NERSC

- Reliable & easy-to-use web-based service:
 - Automatic retries
 - Email notification of success or failure
- Accessible to all NERSC users
- NERSC-managed endpoints on DTNs for optimized data transfers
- Web based GUI for drag-and-drop transfers
- NERSC Globus scripts for command line transfers
- REST/API for scripted interactions with service
- Globus Connect Personal for setting up endpoints on your laptop



<https://docs.nersc.gov/services/globus/>

General Tips for Transferring Data

- Use Globus Online for **large transfers**
 - Also for internal NERSC transfers e.g. between CFS and scratch
- **scp** is fine for **smaller, one-time transfers** (<100MB)
 - But note that Globus is also fine for small transfers
- **Don't use DTN nodes for non-data transfer purposes**
 - Use system login nodes for more general routine tasks

Performance Considerations

- Performance is often **limited by the remote endpoint**
 - Not tuned for WAN transfers or have limited network link
 - These can lower performance <100 MB/sec.
- File system contention may be an issue
 - Try the transfer at a different time or on a different FS.
- **Don't use your \$HOME directory for I/O!**
 - Instead use CFS, \$SCRATCH ...
- If you think you are not getting the performance you expect, let us know: help.nersc.gov

Transferring with NERSC HPSS

- HPSS tape archive is recommended for **archiving large amounts** of data for **long periods of time**
 - See: <https://docs.nersc.gov/filesystems/archive/>
- Use interactive DTNs or xfer queue to transfer to / from HPSS
 - HSI for individual files and conditional access
 - HTAR for aggregation & optimization of storage/archival of large numbers of files. Aim for bundle sizes of 200GB - 2 TB
- NERSC Globus Command line tools for external Globus transfers
 - Will automatically sort files in tape order
 - However the Globus web interface does not directly support aggregation with htar or tape-ordering
 - Preferred use: small number of large files

<https://docs.nersc.gov/services/globus/#command-line-globus-transfers-at-nersc>

Conclusion

- NERSC has multiple file systems to fulfill different performance and capacity needs
- Many different ways to share and transfer data
- Further reading: <https://docs.nersc.gov/filesystems/>

Thank You and
Welcome to
NERSC!

