OAK RIDGE National Laboratory | LEADERSHIP COMPUTING FACILITY

ADMI Symposium 2026

HPC Crash Course

Togo Odbadrakh, Elijah MacCarthy

U.S. DEPARTMENT OF ENERGY

ORNL IS MANAGED BY UT-BATTELLE LLC
FOR THE US DEPARTMENT OF ENERGY

SCHEDULE

11:00	Intro to OLCF & HPC?
11:05	The Unix Operating System
11:25	The Vim Text Editor
11:45	Slurm Job Submission
11:50	Demo

Oak Ridge Leadership Computing Facility (OLCF)

- One of two Department of Energy LCF's
- Based in Oak Ridge, TN at the Oak Ridge National Laboratory (ORNL)
- Department of Energy-funded research
  - Neutron Science, High-Performance Computing, Advanced Materials, Biology and Environmental Science, Nuclear Science and Engineering, Isotopes, and National Security
- Largest, most modern center for unclassified computing in the US

We are one of the DOE's Office of Science computational user facilities

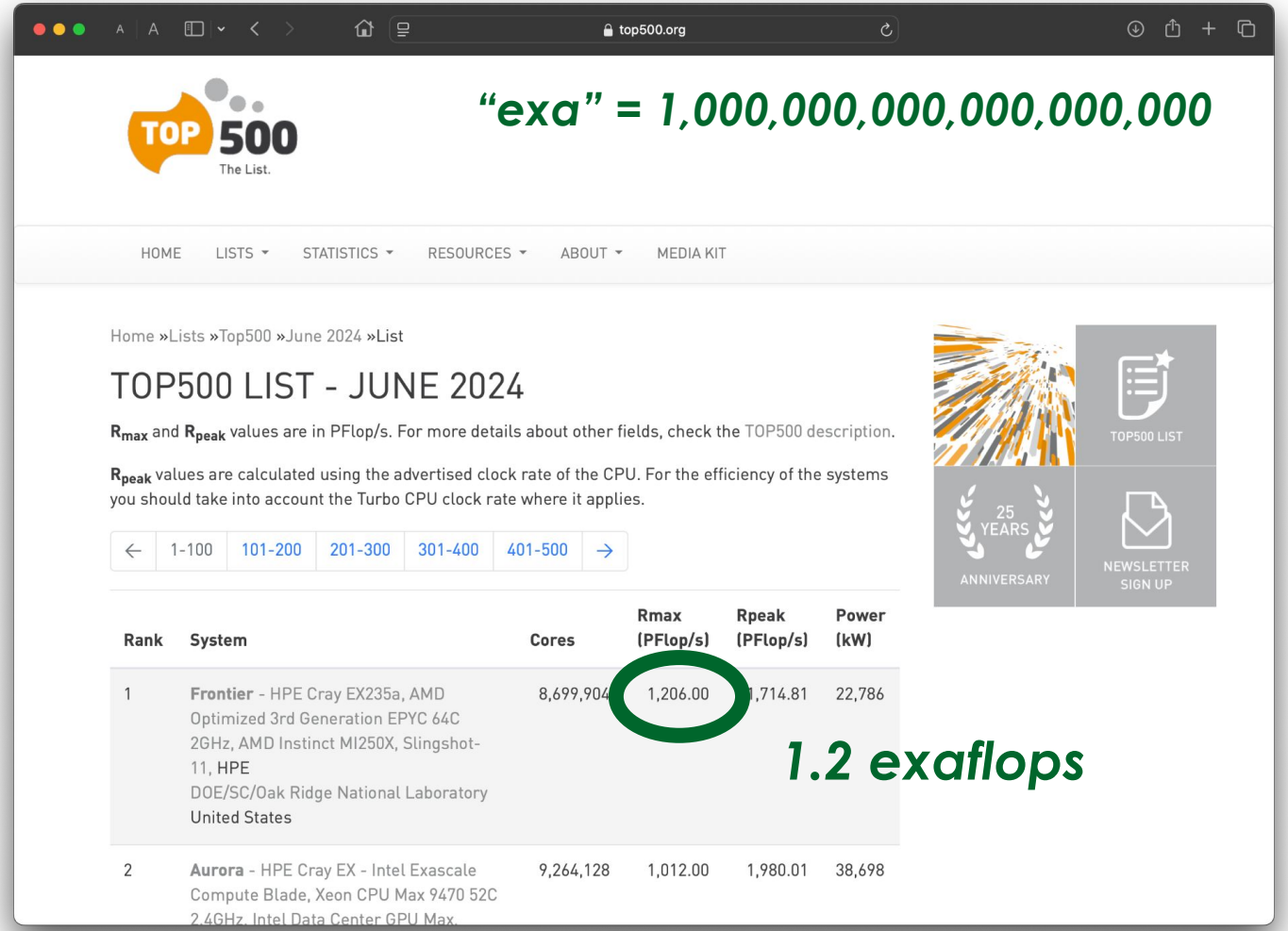
Argonne NATIONAL LABORATORY
Argonne Leadership Computing Center

OLCF
Oak Ridge Leadership Computing Facility
Oak Ridge Leadership Computing Center

NERSC
National Energy Research Scientific Computing Center

- DOE is leader in open high-performance computing
- Provide the world's most powerful computational tools for open science
- Access is free to researchers who publish
- Boost US competitiveness
- Attract the best and brightest researchers

One Big Job


top500.org
TOP 500
The List.
"exa" = 1,000,000,000,000,000,000
HOME
LISTS
STATISTICS
RESOURCES
ABOUT
MEDIA KIT
Home »Lists »Top500 »June 2024 »List
TOP500 LIST - JUNE 2024
Rmax and Rpeak values are in PFlop/s. For more details about other fields, check the TOP500 description.
Rpeak values are calculated using the advertised clock rate of the CPU. For the efficiency of the systems you should take into account the Turbo CPU clock rate where it applies.
1-100
101-200
201-300
301-400
401-500
TOP500 LIST
25 YEARS
ANNIVERSARY
NEWSLETTER SIGN UP
Rank
System
Cores
Rmax (PFlop/s)
Rpeak (PFlop/s)
Power (kW)
1
Frontier - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE
DOE/SC/Oak Ridge National Laboratory
United States
8,699,904
1,206.00
1,714.81
22,786
2
Aurora - HPE Cray EX - Intel Exascale Compute Blade, Xeon CPU Max 9470 52C 2.4GHz, Intel Data Center GPU Max,
9,264,128
1,012.00
1,980.01
38,698
1.2 exaflops

OLCF Frontier Overview

The system includes

- 9,408 nodes
- HPE Slingshot interconnect with 200 GbE interfaces
- 679 PB multi-tier Lustre filesystem “Orion”

System Performance

- Peak of 1.206 ExaFlop/s for modeling & simulation
- Peak of 9.95 ExaOps/s for data analytics and artificial intelligence

Each node has

- 1 AMD Optimized 3rd Gen EPYC 64 core processor
- 4 AMD MI250X Instinct GPUs (8 GCDs)
- 640 GB of fast memory (128 GB HBM2 + 512 GB DDR4)
- 3.48 TB of NV memory (2 X 1.92TB NVMe SSDs)


TOP 500 CERTIFICATE
The List.

Frontier - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11

DOE/SC/Oak Ridge National Laboratory, United States

is ranked

No. 1

among the World's TOP500 Supercomputers

with 1194.00 PFlop/s Linpack Performance

in the 61st TOP500 List published at the ISC23

Conference on June 01, 2023.

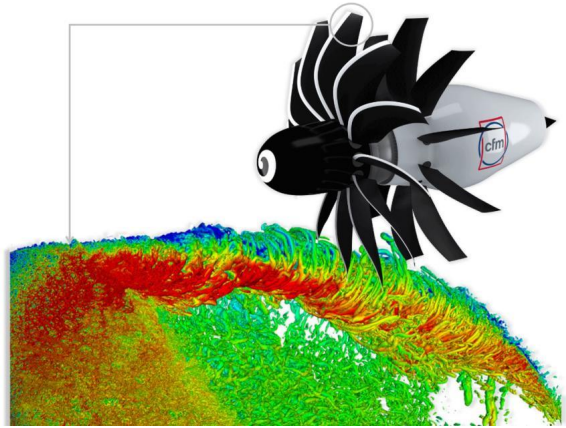
Congratulations from the TOP500 Editors

Erich Strohmaier, NERSC/Berkeley Lab
Jack Dongarra, University of Tennessee
Horst Simon, NERSC/Berkeley Lab
Martin Meuer, Prometeus

First to Exascale!

Example Use Case

- GE Aerospace is designing next-gen commercial aircraft engines
- Frontier enables better evaluation of engine technologies at flight scale
- Software designed to model engine performance and noise levels
- Resulting runs simulated air movement for a full-scale open fan engine design with incredible detail

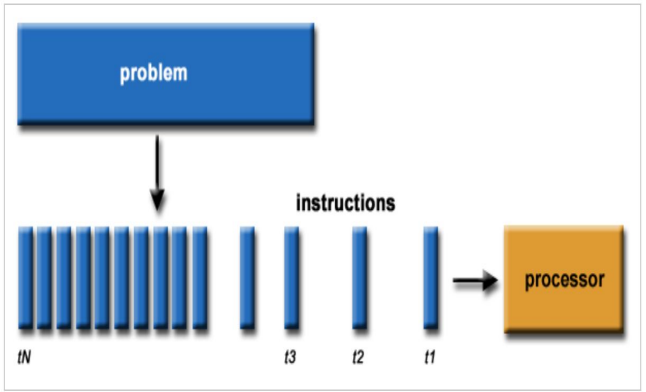


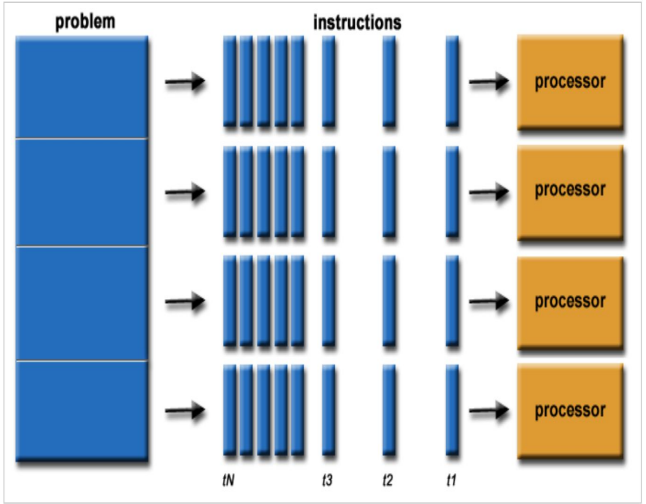
Credit: CFM, GE Research
https://www.ornl.gov/sites/default/files/styles/main_image_style/public/2023-08/GEAerospaceEngine.jpg

7

What is High Performance Computing?

- High Performance Computing (HPC) is about solving the world's largest engineering and science problems with supercomputers.
- That means doing work efficiently in parallel





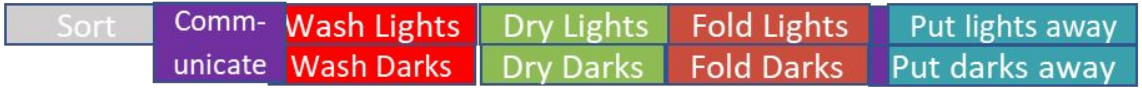
Images courtesy of :
https://hpc.llnl.gov/training/tutorials/introduction-parallel-computing-tutoria

Parallel Algorithms

Just you and one washer and one dryer

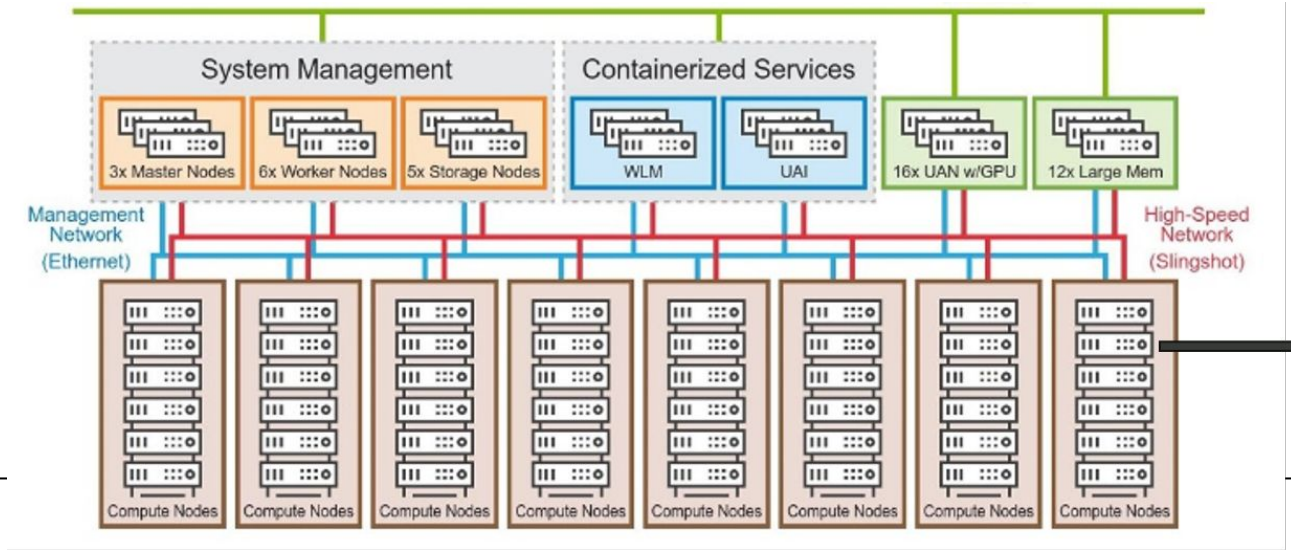
Optimizing just you and one washer and one dryer

You your housemate, 2 washers, and 2 dryers



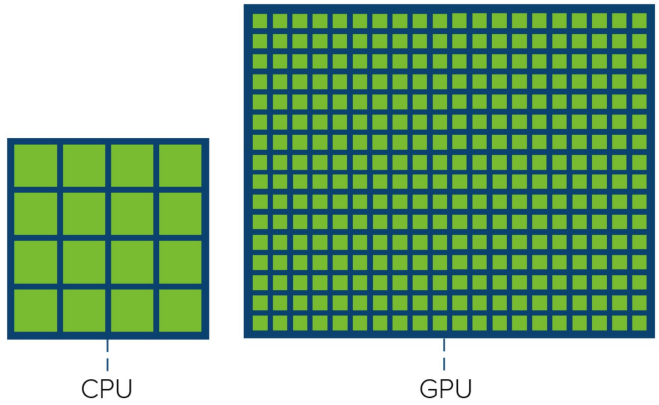
What is High Performance Computing?

- High Performance Computing (HPC) is about solving the world's largest engineering and science problems with supercomputers.
- That means doing work efficiently in parallel



Processors

- Modern HPC nodes combine Central Processing Units CPUS with accelerators such as Graphical Processing Units GPUs.
  - CPUs generally have several compute cores that can run 10s of tasks at a time
  - GPUS have streaming multiprocessor that can run 1000s of tasks at a time.



The Unix Operating System

The Unix Operating System

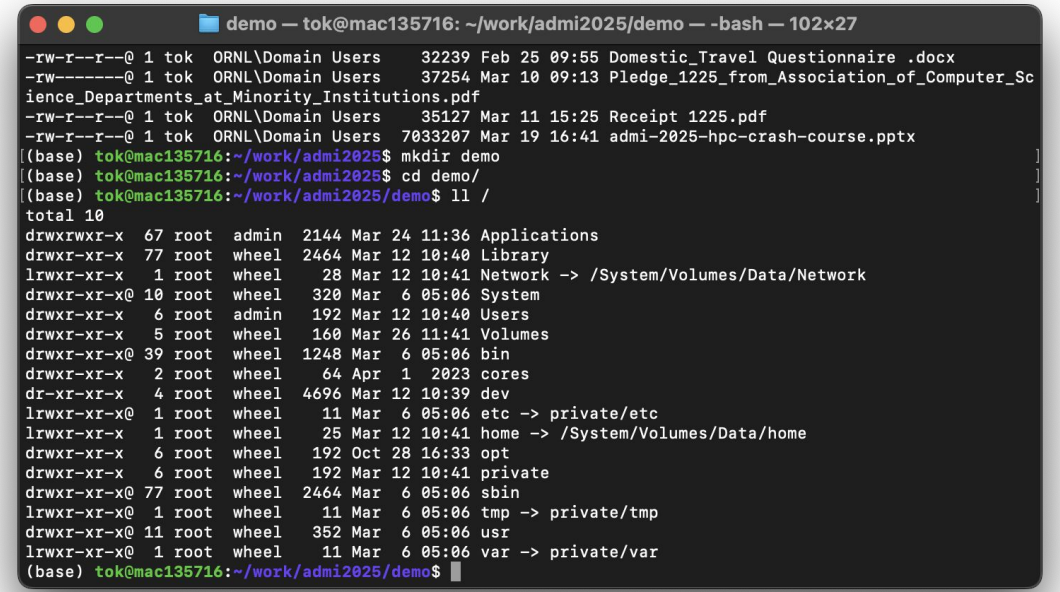
- HPC Clusters operate on the Unix operating system

Born 1969

At Bell Labs

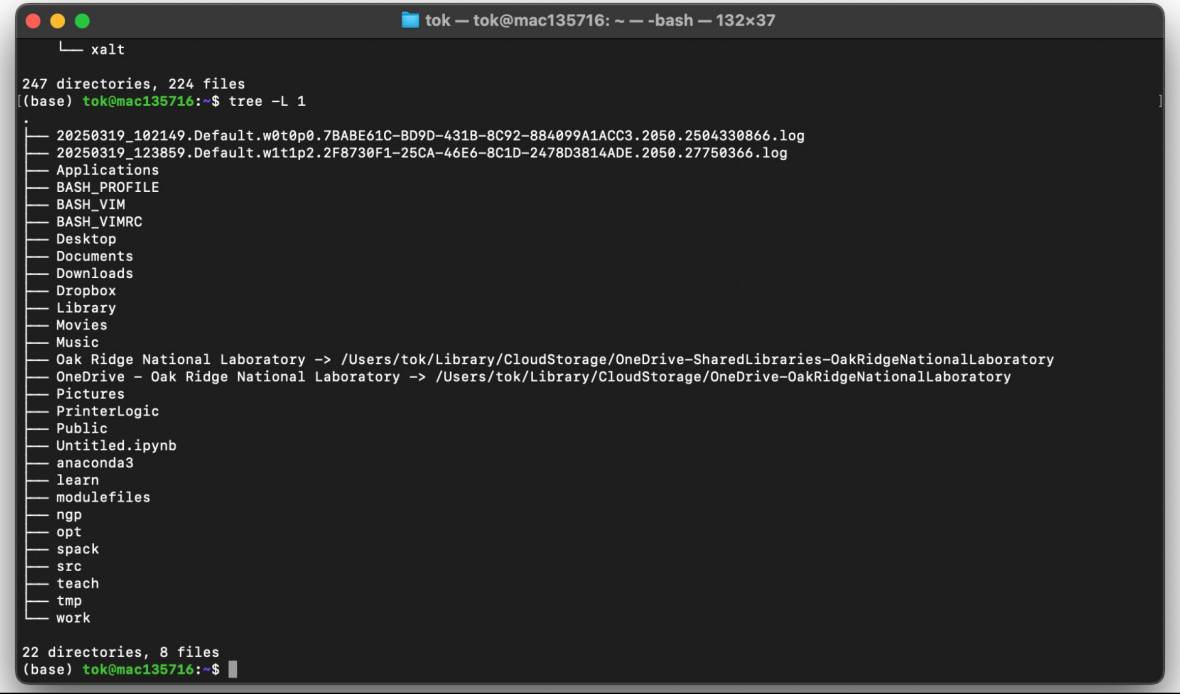
The Unix Operating System

- The Unix OS operates via the command line interface



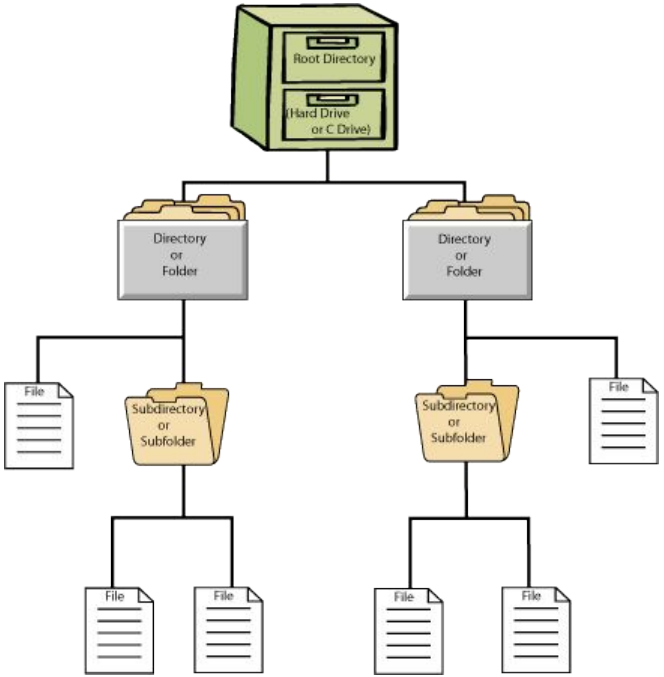
The Unix Operating System

- The Unix OS is file-based



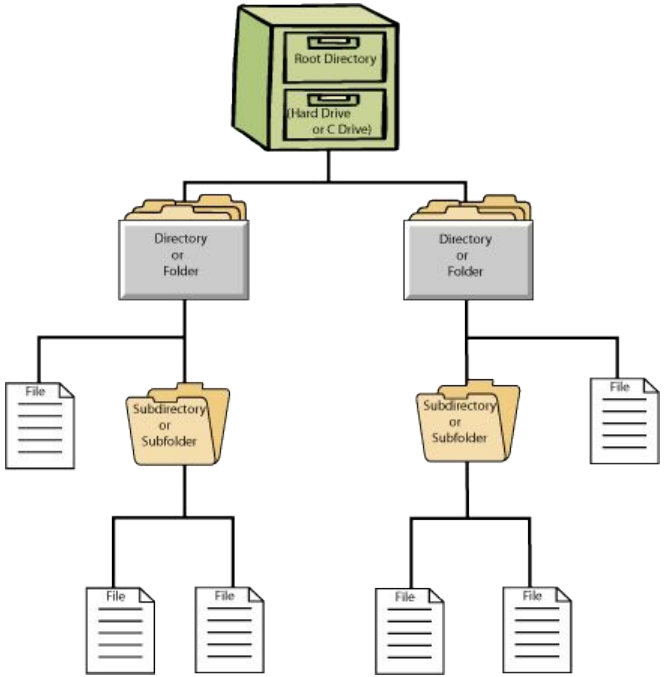
The Filesystem 1

- **File** – A collection of data
  - Plain text, Input/output, a program (executable), an image, etc.
- **Directory** – A logical structure to help organize files (think "folder")
- **Filesystem** – A collection of files and directories
- As a user, you land in your "home" directory when you log in, and can create directories and files there, or move elsewhere to do that.



The Filesystem 2

- Think of the filesystem as a tree. Instead of branches, there are “directories. Instead of leaves, there are “files”
- It starts at /, which is called the “root” directory
- The slash (/) is the directory separator; thus when we move into subdirectories it’s used to separate things (i.e. /ccsopen/home/your_username)



If you are ever “lost”, use the pwd command. It prints your location.

```
$ pwd
/ccsopen/home/your_username
```

Basic Commands

- Commands are usually abbreviations of words (or a series of words)
  - cp for “Copy”, rm for “Remove”, cd for “change directory”
- Commands tend to follow the following syntax:

```
[command] [option flags] [file/directory/object to act upon]
```

- Almost all commands take various options to control what they do.
- Some help is available via an online manual
  - The `man` command
  - Example: Want info about `ls`? Type the following:

```
man ls
```

Basic File/Directory Commands

Command	Description
pwd	**P**rint **w**orking (i.e. current) **d**irectory
man	Display the **Man**ual for a command
whoami	Display the current user's username
mkdir	Create a directory (**M**a**K**e **DIR**ectory)
rmdir	Delete a directory (**R**e**M**ove **DIR**ectory)
cd	**C**hange (into a) **d**irectory
ls	**Ls**t files
cp	**Cp**y a file
mv	**M**o**v**e a file (also used to rename a file)
rm	**Rm**ove (delete) a file
cat	Display the contents (con**cat**enate) a (hopefully text) file
less	Display the contents of a text file in a viewing mode
find	**Find** files by filename
grep	Search for text within files
diff	Identify **diff**erences in the contents of files

Directories

Use `pwd`, `cd`, `mkdir`, and `rmdir` commands to navigate the filesystem and manipulate directories

```
$ pwd
/home/user1

$ mkdir dir1

$ ls
dir1

$ cd dir1

$ pwd
/home/user1/dir1

$ cd ..

$ rmdir dir1
```

Typing just `cd` will always take you back to home no matter where you are.

Directories must be empty in order to delete them with `rmdir`

Special Directories

- Every directory contains two special directory entries: ”.” and “..”
- . is a reference to the current directory
- .. is a reference to the parent directory (so we can do things like `cd ..`)
- ~ can be a reference to home directories
  - ~/ is yours
  - ~user1/ is user1’s
- You’ll see how these are useful later

Manipulating Files 1

- The `rm` command is used to delete a file.
- The `mv` command is used to move and rename files
- There are multiple ways to create and view text files. In the challenge, we will look at various ways to use `cat` and `less` commands to do this.
- Utilities such as `less` and `cat` are intended only for text files. The system will not stop you from running them on a non-text file
  - If you do, you'll get a screenful of unintelligible characters
  - You might get a recognizable prompt (you might not)
  - There's no shame in closing that session's window & re-connecting

Manipulating Files 2

- The `cp` command is used to copy a file, and the `mv` command is used to move and rename files.

```
$ ls
dir1    filea   fileb

$ cat filea
This is a file that
contains three lines
of text.

$ cp filea filea1

$ ls
dir1    filea   filea1  fileb

$ mv filea1 filec

$ ls
dir1    filea   fileb   filec
```

Manipulating Files 3

- The `rm` command is used to delete a file. `cat` prints contents of a file to the screen. `less` displays the contents of a file in a viewing mode. (Press “q” to exit).

```
$  cat filec
This is a file that
contains three lines
of text.

$ less filec

$ mv filec dir1

$ ls dir1
filec

$ rm dir1/filec

$ ls
dir1    filea   fileb
```

Many uses of cat

Command	Explanation
cat file1.txt	Display contents of file
cat file1.txt file2.txt	Concatenate two text files and display the result in the terminal
cat file1.txt file2.txt > newcombinedfile.txt	Concatenate two text files and write them to a new file
cat >newfile.txt	Create a file called newfile.txt. Type the desired input and press CTRL+D to finish. The text will be in file newfile.txt.
cat -n file1.txt file2.txt > newnumberedfile.txt	Some implementations of cat, with option -n, can also number lines
cat file1.txt > file2.txt	Copy the contents of file1.txt into file2.txt
cat file1.txt >> file2.txt	Append the contents of file1.txt to file2.txt
cat file1.txt file2.txt | less	Run the program "less" with the concatenation of file1 and file2 as its input

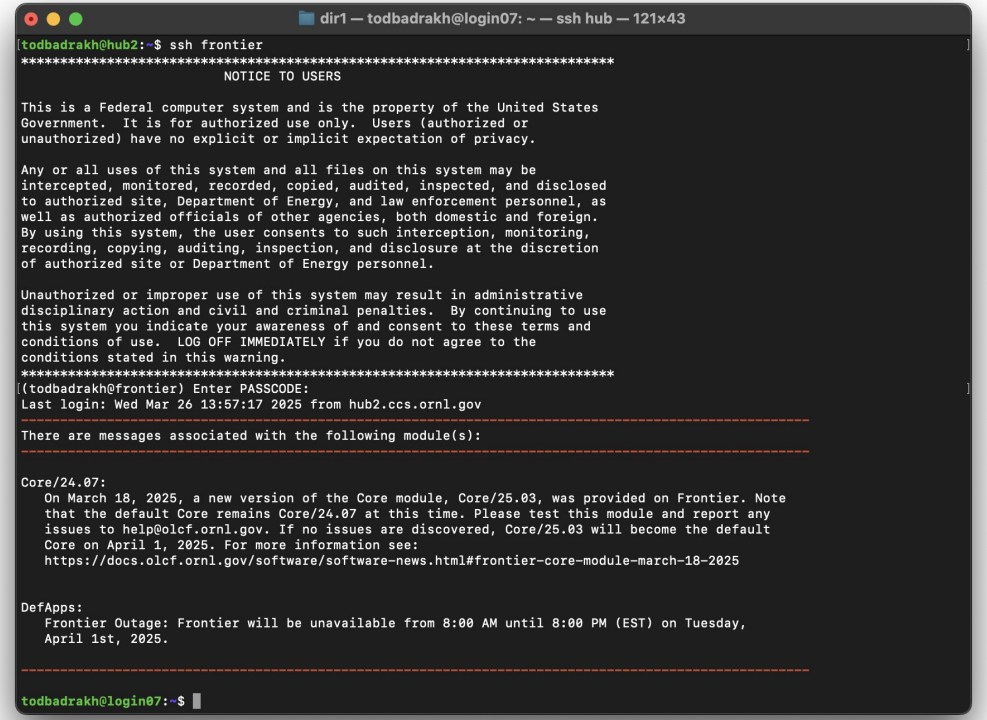
Searching Within Files – grep 1

- Sometimes you want to search for patterns/strings in a file. As with other commands, `grep` takes many options. The grep command searches for "regular expressions"…strings that contain characters with special meaning
- Simple case: find lines with the string 'user' in file1
  ```
  grep "user" file1
  ```
- More complex: show lines ending with 'user' in file1
  ```
  grep "user$" file1
  ```
- …or perhaps lines beginning with 'user'
  ```
  grep "^user" file1
  ```
- Search all files in a dir1 for the string 'user'

```
grep -r "user" dir1/
```

The Unix Operating System

- Frontier command line environment



Login

SSH client

- Open Terminal or Powershell

```
ssh <username>@frontier.olcf.ornl.gov
```

- Enter your password when prompted.

Browser method

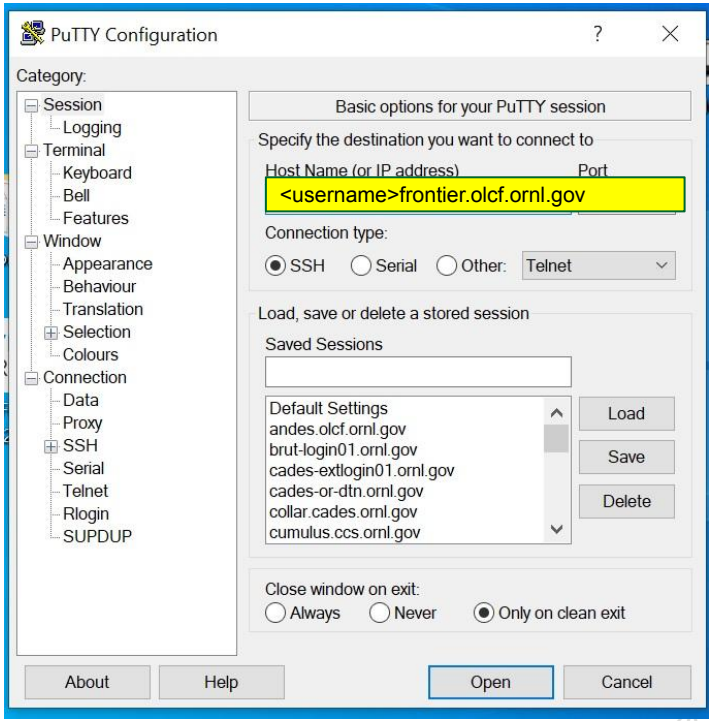
- Open browser
- Go to https://jupyter.olcf.ornl.gov/
- Login with your csep### Username and PIN followed by your Token Passcode
- Choose “HPC and AI Crash Course lab”
- Navigate to the terminal icon and click it.
  - Then type

```
ssh <username>@frontier.olcf.ornl.gov
```

- Enter your PIN followed by the token code

Windows Putty

- Click Open Putty
- Enter<username>@frontier.olcf.ornl.gov
- Click Open
- Enter your password when prompted.



The Vim Text Editor

Questions to Answer

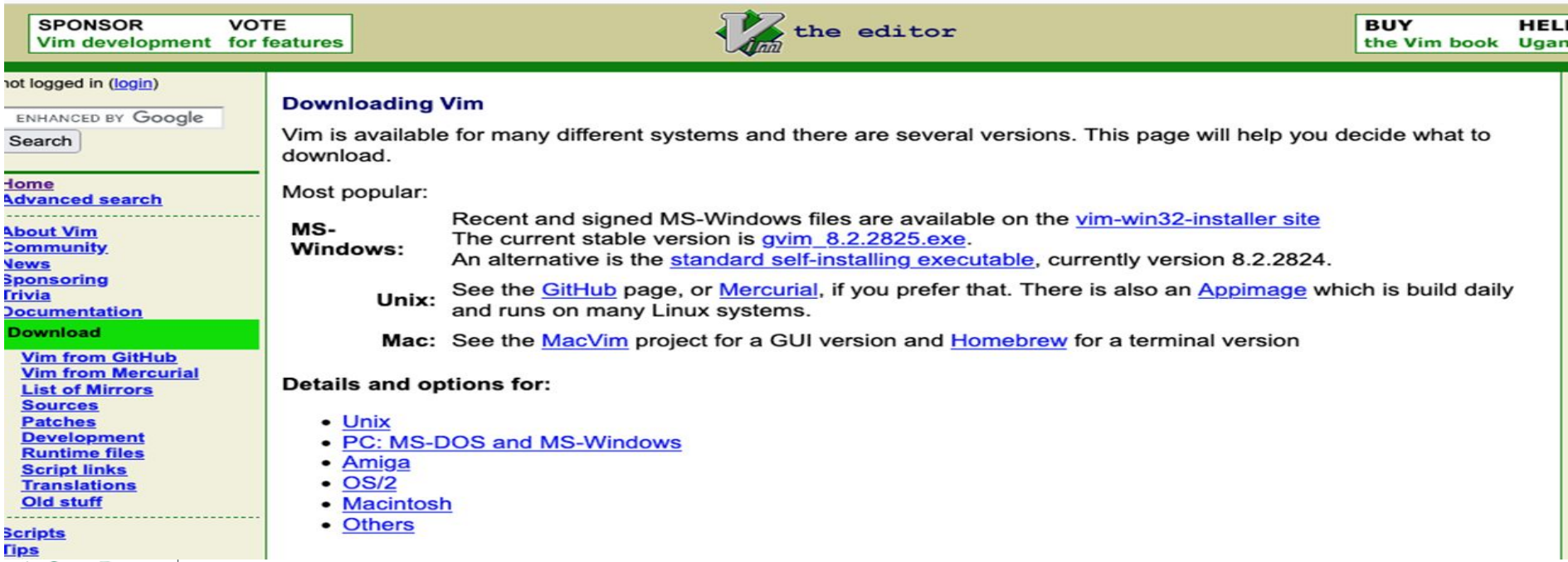
- What is Vim?
- How do I use Vim?
- Where can I learn more about Vim?

What is Vim?

- "Vim is a highly configurable text editor built to enable efficient text editing. It is an improved version of the vi editor distributed with most UNIX systems"
- Modal command-line text editor
- Common in HPC environments
  - Edit source code, job submission scripts, config files, & others!
- Vi and Vim are often used synonymously
  - Vim is a superset of Vi

Installing Vim

- Available by default on all OLCF systems
- Easily installable to your own laptop



Where can I learn more about Vim?

- In command line: $ vimtutor
- vim-adventures.com
- www.vim.org/docs
- Help reference in Vim: `:help`

How do I use Vim?

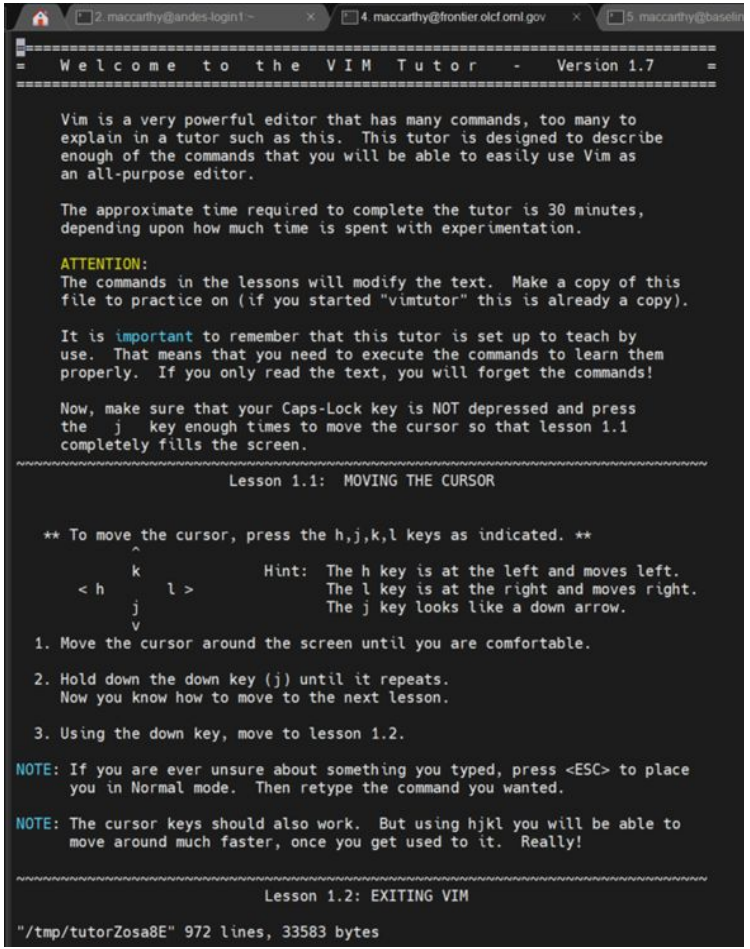
Let's go to the command line

[maccarthy@login05.frontier ~]$ vimtutor

$git clone https://github.com/olcf/foundational_hpc_skills.git

$cd foundational_hpc_skills/intro_to_vim

$ vim intro_vim_with_exercises.txt

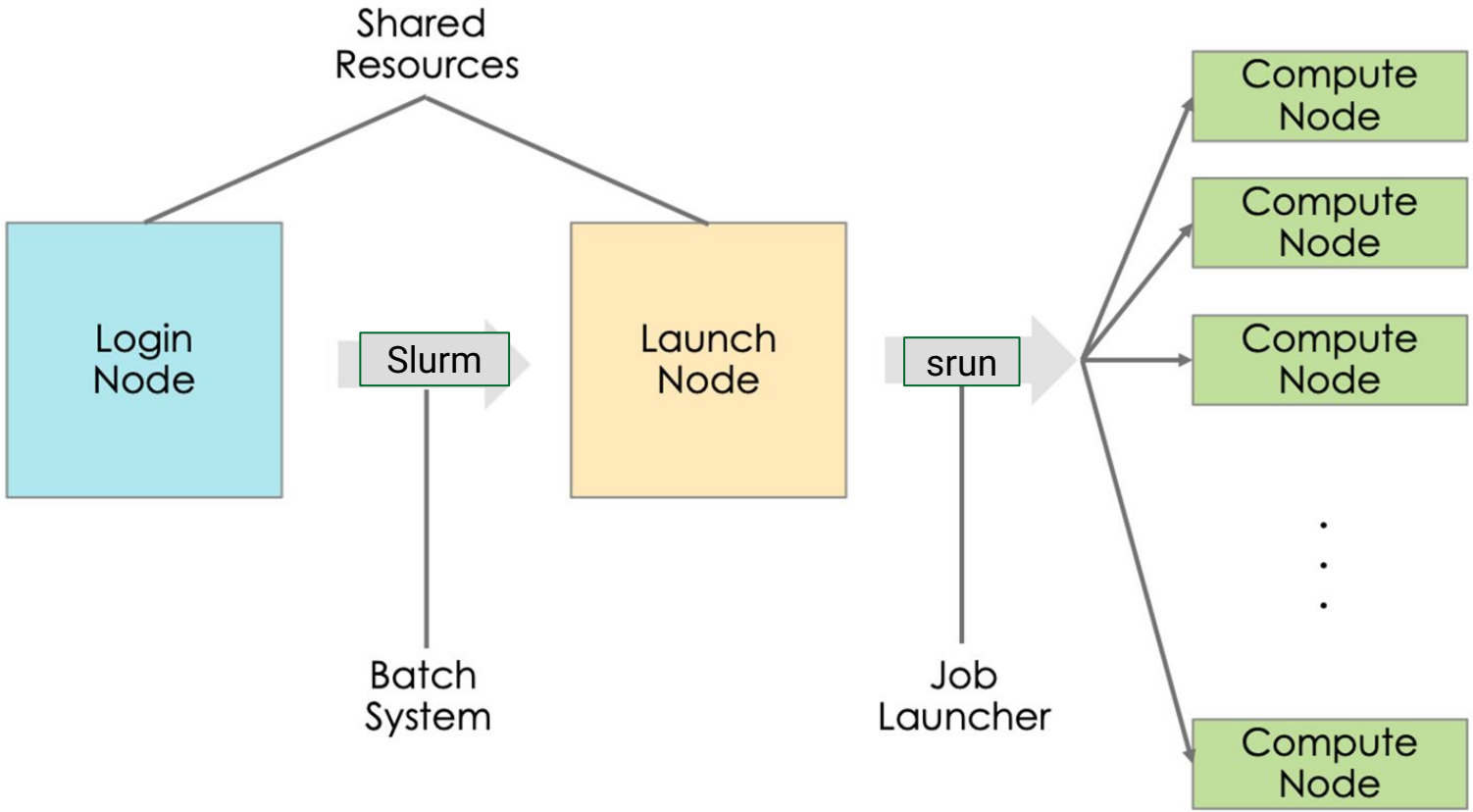


The Unix Operating System

Resource Schedulers

- There are two node categories on most HPC systems
  - Login node
  - Compute Node
- On most systems, you land directly on a login node once you sign-on.
- Login nodes are shared and not for intensive compute or IO operations.
- Thus, jobs are supposed to be run on compute nodes.
- HPC Resource schedulers are needed to be able to run jobs on compute nodes.
- Schedulers manage and distribute computational resources across a cluster, ensuring fair and efficient access for multiple users.
- There are several types of schedulers
  - Slurm Scheduler
  - PBS/Torque scheduler
- Frontier has a slurm scheduler

Login and Compute Nodes



Frontier Slurm Scheduler

Slurm provides 3 ways of submitting and launching jobs on Frontier compute nodes: **batch scripts, interactive, and single-command.**

NOTE: Jobs launch on the allocated compute nodes, with the first compute node in the allocation serving as head-node.

Slurm commands associated with the 3 methods:

`sbatch`	Used to submit a batch script to allocate a Slurm job allocation. The script contains options preceded with `#SBATCH`.
`salloc`	Used to allocate an interactive Slurm job allocation, where one or more job steps (i.e., `srun` commands) can then be launched on the allocated resources (i.e., nodes).
`srun`	Used to run a parallel job (job step) on the resources allocated with `sbatch` or `salloc`, or used to create a resource allocation and run a job step in a single command.

Slurm Scheduler

```
#!/bin/bash
#SBATCH -A ABC123
#SBATCH -J RunSim123
#SBATCH -o %x-%j.out
#SBATCH -t 1:00:00
#SBATCH -p batch
#SBATCH -N 1024

cd $MEMBERWORK/abc123/Run.456
cp $PROJWORK/abc123/RunData/Input.456 ./Input.456
srun ...
cp my_output_file $PROJWORK/abc123/RunData/Output.456
```

You submit with **sbatch <script-name>**

Option	Example Usage	Description
`-A`	`#SBATCH -A ABC123`	Specifies the project to which the job should be charged
`-N`	`#SBATCH -N 1024`	Request 1024 nodes for the job
`-t`	`#SBATCH -t 4:00:00`	Request a walltime of 4 hours. Walltime requests can be specified as minutes, hours:minutes, hours:minuts:seconds days-hours, days-hours:minutes, or days-hours:minutes:seconds
`--threads-per-core`	`#SBATCH --threads-per-core=2`	Number of active hardware threads per core. Can be 1 or 2 (1 is default) **Must** be used if using `--threads-per-core=2` in your `srun` command.
`-d`	`#SBATCH -d afterok:12345`	Specify job dependency (in this example, this job cannot start until job 12345 exits with an exit code of 0. See the Job Dependency section for more information
`-C`	`#SBATCH -C nvme`	Request the burst buffer/NVMe on each node be made available for your job. See the Burst Buffers section for more information on using them.
`-J`	`#SBATCH -J MyJob123`	Specify the job name (this will show up in queue listings)
`-o`	`#SBATCH -o jobout.%j`	File where job STDOUT will be directed (%j will be replaced with the job ID). If no *-e* option is specified, job STDERR will be placed in this file, too.
`-e`	`#SBATCH -e joberr.%j`	File where job STDERR will be directed (%j will be replaced with the job ID). If no *-o* option is specified, job STDOUT will be placed in this file, too.
`--mail-type`	`#SBATCH --mail-type=END`	Send email for certain job actions. Can be a comma-separated list. Actions include BEGIN, END, FAIL, REQUEUE, INVALID_DEPEND, STAGE_OUT, ALL, and more.
`--mail-user`	`#SBATCH --mail-user=user@somewhere.com`	Email address to be used for notifications.
`--reservation`	`#SBATCH --reservation=MyReservation.1`	Instructs Slurm to run a job on nodes that are part of the specified reservation.

DEMO

1. Srun Job Launcher
2. Basic Unix Vim
3. Basic Workflow
4. Password in a Haystack

What is the Programming Environment?

At the highest level, the PE is your shell's build- and run-time environment.

- Compilers, compiler wrappers, tools, scientific libraries, runtimes, etc.
- See output of running **env**

OLCF offers many software packages for users

Managed by users through session environment variables.

- Search paths
  - `PATH, LD_LIBRARY_PATH, LIBRARY_PATH, PKG_CONFIG_PATH, …`

Much of the available software cannot coexist simultaneously in your environment.

LMOD Environment Modules

Frontier runs LMOD to manage environment complexity

Build- and runtime-environment software managed with LMOD

- https://lmod.readthedocs.io

Usage:

```
$ module -t list                        # list loaded modules
$ module avail                          # show modules that can be loaded given current env
$ module help <package>                 # help info for package (if provided)
$ module load <package> <package>…      # add package(s) to environment
$ module unload <package> <package>…    # remove package(s) from environment
$ module reset                          # restore system defaults
$ module restore <collection>           # load a saved collection
$ module spider <package>               # deep search for modules
$ module purge                          # clear all modules from env
```

LMOD Environment Modules – `module avail`

- The **`module avail`** command shows *only what can be loaded given currently loaded packages.*
- Full or partial package names limit output to matches

```
$ module avail
…
---------------------- [ Cray Programming Environment ] -----------------------
   PrgEnv-amd/8.3.3                     cray-mpixlate/1.0.5
   PrgEnv-amd/8.4.0                     cray-mpixlate/1.0.6
   PrgEnv-amd/8.5.0                     cray-mrnet/5.0.4
   PrgEnv-amd/8.6.0             (D)     cray-mrnet/5.1.0
   PrgEnv-cray-amd/8.3.3                cray-mrnet/5.1.1
   PrgEnv-cray-amd/8.6.0        (D)     cray-mrnet/5.1.4               (D)
   PrgEnv-cray/8.3.3                    cray-openshmemx/11.5.7
PrgEnv-cray/8.6.0            (L,D)    cray-openshmemx/11.6.1
…
  Where:
   L:  Module is loaded
   D:  Default Module

Use "module spider" to find all possible modules and extensions.
Use "module keyword key1 key2 ..." to search for all possible modules matching any of the "keys".
```

LMOD Environment Modules – `module spider`

- Use **`module spider`** (instead of **`module avail`**) to search for modules
  - Finds packages that cannot be loaded given current environment
  - Shows requirements needed to make package available

```
$ module spider hdf5
----------------------------------------------------------------------------
   hdf5:
----------------------------------------------------------------------------
    Versions:
      hdf5/1.12.1-mpi
      hdf5/1.14.3-mpi
      hdf5/1.14.5
Other possible modules matches:
      cray-hdf5  hdf5-vol-cache  hdf5-vol-log  hdf5-vol-async  cray-hdf5-parallel  cray-netcdf-hdf5parallel

----------------------------------------------------------------------------
For detailed information about a specific module use the module's full name.
For example:

    $ module spider Foo/1.2.3----------------------------------------------------------------------------------------
  For detailed information about a specific "hdf5" package (including how to load the modules) use the module's full name.
  Note that names that have a trailing (E) are extensions provided by other modules.
  For example:

     $ module spider hdf5/1.10.4
```

Compiler Environments

CCE
cce/18.0.1(D)
cce/18.0.0

(OpenMP Offload)
(OpenACC)*
(HIP)*

AMD
amd/6.2.4(D)
amd/6.3.1

(OpenMP Offload)
(OpenACC)*
(HIP)

GCC
gcc/12.2.0(D)
gcc/11.2.0

(OpenMP Offload)*
(OpenACC)*

Programming Environments (PrgEnv)

PrgEnv-cray
PrgEnv-cray/8.6.0(D)
PrgEnv-cray/8.5.0

PrgEnv-amd
PrgEnv-amd/8.6.0(D)
PrgEnv-amd/8.5.0

PrgEnv-gnu
PrgEnv-gnu/8.6.0(D)
PrgEnv-gnu/8.5.0

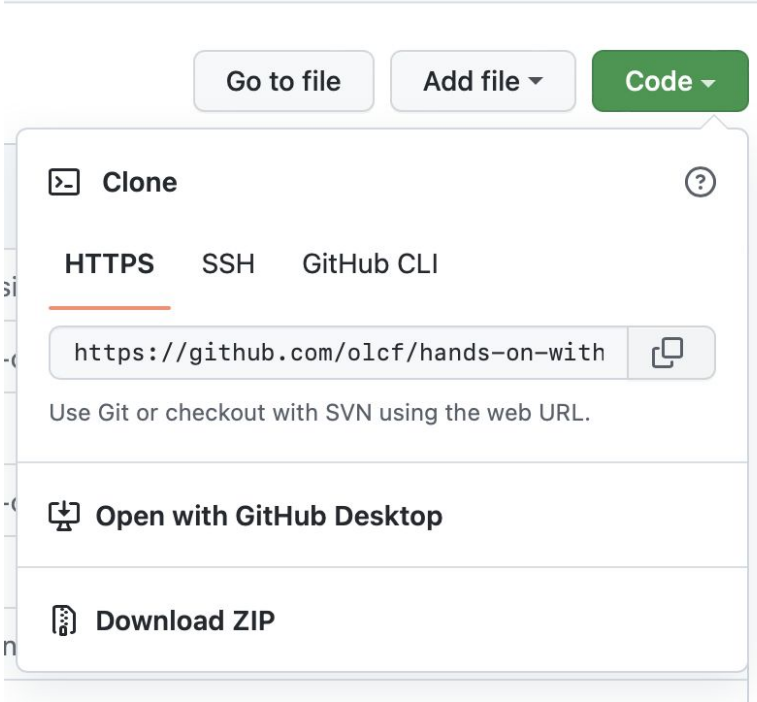
How to get the repo

Direct your browser to

https://github.com/olcf/hands-on-with-frontier

Click on the  to copy the address.

Go to your command line and type
“git clone”
then paste in the address.




```
$ git clone https://github.com/olcf/hands-on-with-frontier
Cloning into 'hands-on-with-frontier'...
remote: Enumerating objects: 3242, done.
remote: Counting objects: 100% (695/695), done.
remote: Compressing objects: 100% (272/272), done.
```

Join the Course SLACK Workspace

https://bit.ly/4hyqLHh

Want to Work at Oak Ridge National Laboratory?

Fill out our Talent pool form --->

Jobs: https://jobs.ornl.gov

Internships: https://education.ornl.gov

Oak Ridge National Laboratory

OAK RIDGE National Laboratory | LEADERSHIP COMPUTING FACILITY