CISL Facilities and Support

Getting Started

Rory Kelly

Consulting Services

August 05, 2019

This material is based upon work supported by the National Center for Atmospheric Research, which is a major facility sponsored by the National Science Foundation under Cooperative Agreement No. 1852977.

CISL Mission for User Support

CISL provides services to enable researchers to utilize community resources for computing, data analysis, post-processing and storage. CISL also provides training for effective use of the resources via classes on programming, parallel computing, visualization and analysis, and offers user support via phone, email, and in-person consulting.

CISL Facilities Overview

- Computing Systems - Cheyenne
- Data Analysis and Visualization - Casper
- Data Storage - GLADE
- Allocations Info - SAM
- Additional Training Resources
- Contacting Support

Cheyenne
HPE (née SGI) ICE XA Cluster

Cheyenne
HPE ICE XA Cluster

- Batch Compute Nodes
  - 18-core, 2.3-GHz Intel “Broadwell” CPUs
  - 4,032 dual-socket nodes
  - 145,152 total cores, 5.34 PFLOPs peak
  - 313 TB total memory (3,164 64-GB and 864 128-GB nodes)
- High-Performance Interconnect
  - Mellanox EDR InfiniBand
  - 9-D enhanced hypercube topology
  - 100-Gbps link bandwidth — 0.5 µs latency
  - 36 TB/s bisection bandwidth
- Login Nodes
  - 6 nodes with 2.3-GHz Intel Xeon E5-2697v4 processors
  - 36 cores & 256 GB memory per node
- Service Nodes (scheduler, license servers)
  - 4 nodes, 36 cores & 64 GB memory per node

Accessing Cheyenne

- Access Cheyenne via SSH
  - Linux / macOS : Teminal, Xterm, iTerm2, etc
  - Windows: WSL, Cygwin, PuTTY, etc

- Two factor authentication
  - Yubikey
  - Duo

- Can access a specific login node, or be placed automatically
  - ssh cheyenne3.ucar.edu
  - ssh cheyenne.ucar.edu

Access with Yubikey

- Plug the Yubikey into a USB port on your computer

- Start a connection with ssh

```
      ssh -X <username>@cheyenne.ucar.edu
  Your screen will display a response:
      Token_Response:
```

- Enter your PIN number but do not hit enter.

- Lightly touch the Yubikey button which will automatically insert a one time password

- More info on yubikey authentication at:
  https://www2.cisl.ucar.edu/user-support/authentication-and-security/yubikey

Access with Duo (push notification)

- Start a connection with ssh

```
      ssh -X <username>@cheyenne.ucar.edu
  Your screen will display a response:
      Token_Response:
```

- Enter your CIT password

- Duo will push an authorization request to your phone. After you affirm the authorization request you will be logged in.

Access with Duo (rolling passcode)

- Use the rolling passcode authentication mode for Duo in situations where you can’t receive a push notification.

- Start a connection with ssh

```
      ssh -X <username>@cheyenne.ucar.edu
  Your screen will display a response:
      Token_Response:
```

- Open the UCAR panel in the Duo app to display the rolling passcode

- Enter your CIT password and the rolling code separated by a comma
  - e.g. correcthorsebatterystaple,161803

- More information on Duo authentication at:
  https://www2.cisl.ucar.edu/user-support/authentication-and-security/authenticating-duo

Supported Software

- CISL supports multiple compilers and compiler versions
  - Intel Compilers (C, C++, Fortran)
    - versions 17.0.1, 18.0.5, 19.0.2
  - GNU Compiler Collection
    - versions 7.4.0, 8.3.0, 9.1.0
  - Portland Group Compilers (C, C++, Fortran, CUDA Fortran)
    - version 19.3
  - Nvidia CUDA compilers (not for cheyenne)
- Multiple MPI libraries
  - MPT, Intel MPI, OpenMPI
- For each of those compilers and MPIs we build a corresponding software stack consisting of tools commonly used in HPC and geoscience
  - NetCDF, PNetCDF, PIO, HDF, Grib, etc
  - Numerical Libraries: Lapack, blas, FFTW, etc
- Parallel Debugger and Performance Analyzer
  - ARM Forge (DDT, MAP, Perf Reports)
- We Also support other common tools for scripting and analysis
  - Julia, python, R, NCO

Using Modules to Access Software

- Modules help manage various user software and the dependencies between them

- Helpful module commands

– module avail	(shows available modules)
– module list	(list currently loaded modules)
– module load/unload <module-name>	(load or unload the module ‘module-name’)
– module help	(display help on the modules system)
– module help <module-name>	(display help on a specific module)
– module save <set-name>	(save current modules as ‘set-name’)
– module restore <set-name>	(load module set ‘set-name’)
– module purge	(remove all modules from environment)
– module reset	(reset modules to default environment)

- More info on modules at:

  https://www2.cisl.ucar.edu/resources/computational-systems/cheyenne/user-environment/environment-modules

PBS Batch Jobs

- Jobs on cheyenne run using the PBS batch system, which allocates the requested resources, schedules and charges according to priority, and balances access between users.

- Submitting a job (qsub)
  - qsub <script-name>   (submits a script ‘script-name’ to run in batch)

- Monitoring jobs
  - qstat   (lists all the jobs currently running or waiting to run)
  - qstat –u <user-name>   (lists status of all jobs belonging to user ‘user-name’)
  - qstat –xf <job-id>   (gives an unreasonable amount of detail on job ‘job-id’)

- Removing jobs
  - qdel <job-id>   (remove job ‘job-id’ from the queue, kill if necessary)

Example PBS Batch Script

```
#!/bin/bash
#PBS -N job_name
#PBS -A project_code
#PBS -l walltime=01:00:00
#PBS -q queue_name
#PBS -j oe
#PBS -m abe
#PBS -M your_email_address
#PBS -l select=2:ncpus=36:mpiprocs=36

### Run the executable
mpiexec_mpt ./executable_name.exe > output_file
```

PBS Batch Jobs

- Running a command in batch with qcmd

  qcmd -- <command>

  will execute 'command' on a batch node

- Getting an interactive node

  qinteractive

  will launch an interactive session on a batch node

- More information on PBS at:

  https://www2.cisl.ucar.edu/resources/computational-systems/cheyenne/running-jobs/submitting-jobs-pbs

Casper Cluster

Data Analysis, Visualization, GPU compute

- 20 Supermicro SuperWorkstation Nodes (data analysis and visualization)
  - Up to 384 GB DDR4-2666 memory per node
  - 2 18-core 2.3-GHz Intel Xeon Gold 6140 (Skylake) processors per node
  - 2 TB local NVMe Solid State Disk
  - Mellanox VPI EDR InfiniBand dual-port interconnect
  - 1 NVIDIA QuadroGP100 GPU on each of 8 nodes
- 2 Supermicro SuperServer Nodes (GPU compute / ML)
  - 768 GB DDR4-2666 memory per node
  - 2 18-core 2.3-GHz Intel Xeon Gold 6140 (Skylake) processors per node
  - 2 TB local NVMe Solid State Disk
  - Mellanox VPI EDR InfiniBand dual-port interconnect
  - 4 NVIDIA Tesla V100 SXM2 GPUs with NVLink
- 4 Supermicro SuperServer Nodes (GPU compute / ML)
  - 1152 GB DDR4-2666 memory per node
  - 2 18-core 2.3-GHz Intel Xeon Gold 6140 (Skylake) processors per node
  - 2 TB local NVMe Solid State Disk
  - Mellanox VPI EDR InfiniBand dual-port interconnect
  - 8 NVIDIA Tesla V100 SXM2 GPUs with NVLink
- Access via ssh to ‘casper.ucar.edu’ and login with 2FA (Duo / Yubikey)

Slurm Batch Jobs

- Jobs on casper run using the Slurm Workload Manager, which allocates the requested resources, schedules and charges according to priority, and balances access between users.

- Submitting a job (sbatch)
  - sbatch <script-name>   (submits a script ‘script-name’ to run in batch)

- Monitoring jobs
  - squeue   (lists all the jobs currently running or waiting to run)
  - squeue –u <user-name>   (lists status of all jobs belonging to user ‘user-name’)
  - scontrol show job <job-id>   (shows a reasonable amount detail on job ‘job-id’)

- Removing jobs
  - qcancel <job-id>   (remove job ‘job-id’ from the queue, kill if necessary)

- Slurm jobs to run on casper can be submitted from cheyenne or casper

Example Slurm Batch Script

```
#!/bin/bash -l
#SBATCH --job-name=mpi_job
#SBATCH --account=project_code
#SBATCH --ntasks=8
#SBATCH --ntasks-per-node=4
#SBATCH --time=00:10:00
#SBATCH --partition=dav
#SBATCH --output=mpi_job.out.%j

### Run program
srun ./executable_name
```

Slurm Batch Jobs

- Starting and interactive job with execdav
  - execdav command starts an interactive job on casper

```
--account=project_code
--time=00:00:00
--ntasks=number_of_tasks
--mem=nG
--constraint=<skylake, gpu, gp100, v100, 4xv100, 8xv100, x11>
```

- Not as necessary since direct-login is now available

GLADE Shared File System

- GLADE is a high performance GPFS file system shared across all CISL HPC and DAV computers

File space	Quota	Back Up	Purge Period	Description
/glade/u/home/username	25 GB	Yes	None	Home directories
/glade/scratch/username	10 TB	No	90 days	Temporary space for short term use
/glade/work/username	1 TB	No	None	Work space for longer term storage
/glade/p/entity/project_code	N/A	No	1 year	Shared space for project allocations

System Account Manager (SAM)

- SAM provides several services related to your computing allocation on CISL machines

- Reports on your available projects and allocations, along with recent charges by project and resource

- Change your default shell on CISL resources. We support two: bash and tcsh

- Change your default project number (HPSS only)

- List of exemptions

- Can access SAM with your CIT password at
  https://sam.ucar.edu

Additional Training

- The Consulting Services Group offers additional training throughout the year
  - Fortran, C, C++
  - Python
  - Parallel computing (MPI and OpenMP)
  - Linux / Unix commands
  - Visualization

- Classes are archived and available for viewing on the web
  https://www2.cisl.ucar.edu/user-support/training

Contacting User Support

- CISL Support Homepage:
  https://www2.cisl.ucar.edu/user-support
- In-person consulting
  - Located at Mesa Lab, 1B level, area 55
  - Enter via tree plaza, make a left and walk to the end of the hall
  - Consultants on duty 8 AM – 5 PM
- Help via web and phone
  - 303.497.2400
  - email to cislhelp@ucar.edu

Questions?