ExCL Cheat Sheet

https://docs.excl.ornl.gov

As of June 2021

Send email to excl-help@ornl.gov to create a ticket for help and support.

Overview

- The Experimental Computing Lab (ExCL) is a laboratory designed for computer science research by offering heterogeneous resources and full configurability of the software stack.
- The computational resources provided by ExCL comprise diverse technologies in terms of chips, memories, and storage. ExCL will also adapt to the ever-changing computing ecosystem and will incorporate the latest technology and make it available to its users.
- The Experimental Computing Lab will offer a mix of exclusive access nodes and shared nodes where users will be able to carry out their research. It follows a novel design that allows a high degree of flexibility for users and administrators to accommodate a wide range of experiments.
- ExCL has been designed and is managed by researchers at the Architectures and Performance Group of Oak Ridge National Laboratory.
- This cheat sheet gives a quick overview, each topic is covered in detail in ExCL documentation.

Software and Job Management

- Several software packages are available on ExCL.
- Can request new software installation via slack or a support ticket.
- Module system for software:

```
module avail              #look for available modules
module whatis             #help on a module
module list               #list loaded modules
module load               #load a module
module unload             #unload a module
module purge              #purge all loaded modules
```

- Additional packages can also be installed by the user with Spack.
- Job management commands for SLURM scheduler:

```
sbatch                    #submit a job
squeue <jobid>            #check job status by job id
squeue -u <userid>        #check job status by user
sinfo                     #queue status summary
scontrol show job <jobid>     #running job info
```

- Use sinfo to see available nodes.
- Start an interactive SLURM job:

```
srun -N 1 -c 32 --mem=0g -t 1:00:00 \
     -A <account> -p <queue> --pty /bin/bash
```

- ExCL also supports GitLab-CI, Docker, and Virtual Machines via KVM.

Storage and Data

- Each user has a home directory on the NFS server: /home/<uid>/
  - Backed up. Good for storing <100 GB.
- Each user has a non-backed-up large file store: /noback/<uid>/
  - Good for storing large files which don't need to be backed up.
- Each system has a local scratch space: /scratch/
  - Good for caching files on a local hard drive.
  - Not shared between nodes.
- Use df -h to see all storage mounted on a node.
- Use du -h <path> to see disk usage.

Access and Connect

- To get access: https://excl.ornl.gov/accessing-excl/
- ssh <id>@login.excl.ornl.gov
  - ID is UCAMS or XCAMS ID
- ThinLinc: https://login.excl.ornl.gov:300/
- To access an internal node, ssh from the login node to the internal node.
  - ssh <internal node>
- Use ThinLinc or X11 forwarding to access GUIs. Using ThinLinc to the login node plus X11 forwarding to internal nodes is the most performant.
  - Login with ThinLinc: https://login.excl.ornl.gov:300/
  - ssh -X <internal node>

Systems

- System list available at https://excl.ornl.gov/excl-systems/.
- Use ssh to connect to the system from the login node.

Spack

- Installation instructions: https://docs.excl.ornl.gov/software/spack
- Detailed Spack documentation: https://spack.readthedocs.io/en/latest/
- Common commands:

```
spack env activate <project> OR spacktivate
spack env create <project> [spack.yml or spack.lock]
spack env status          # Print environment status
spack env list            # List environments
spack install <spec>
spack concretize          # Lock generic spec by concretize.

# Add specific compiler installed by spack to spack
spack compiler add $(spack location -i gcc@8.3.0)

spack list                # What can be installed
spack find                # What is installed

spack versions <package>  # Print all package versions
spack info <package>      # Get package info

spack help -spec          # Print Spack spec help
spack config edit
spack config add
spack config get

spack cd -e <myproject>   # Change to project build directory
spack config blame config # See which config file set config
spack blame <package>     # See commits to a package
```

- Spack environment in a directory:

```
spack env create -d . spack.yaml
spack env activate .
spack install
```

- Load someone else's Spack modules:

```
spack env loads -r        # Create a loads file
module use /noback/<uid>/spack/share/modules/<system-type>
source <generated loads file>
```

Other Cheat Sheets

- Conda
- Matplotlib
- Slurm
- Pandas
- Seaborn

OAK RIDGE National Laboratory

U.S. DEPARTMENT OF ENERGY