Computer Setup

Accelerator Code Workshop
Stanford, March 27 – 31, 2023

Computational Electrodynamics Department
SLAC National Accelerator Laboratory

Contents

SLAC

- Internet Connection & Workshop Materials
- Required Software & Introduction to NERSC
- Connecting to NERSC (Linux/iOS/Windows)
- Working on NERSC (Unix commands/editors)
- Transferring Files (SCP)
- Configuring sshproxy for logins
- ACE3P modules and files
- Submitting Jobs on NERSC
- ParaView Setup

Internet Access

- Wi-Fi ‘SLAC-VISITOR’ network is available in the meeting room
- Open your browser to any non-encrypted web page such as http://www.google.com/
- Complete the registration form and press the ‘I agree’ button

Workshop Materials

- **CW23 Website:**
  - https://conf.slac.stanford.edu/cw
- **Tutorials:**
  - https://confluence.slac.stanford.edu/display/AdvComp/Materials+for+CW23
- **ACE3P-CW23 SLACK:**
  - Check your email for signup information

Required Software: Pre- and Post-processing

- **Cubit v. 16.08**

  Pre-processing tool for solid modeling and meshing

  https://cubit.sandia.gov/

  Requires a license

- **ParaView v. 5.11.0**

  Post-processing and data visualization tool

  http://www.paraview.org/

  Free software

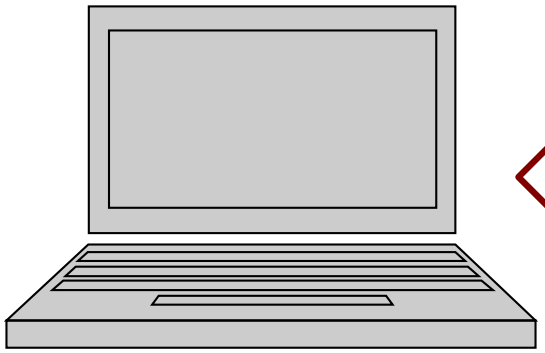
Working at NERSC

The National Energy Research Scientific Computing Center (NERSC) is a High Performance Computing user facility operated by Lawrence Berkeley National Laboratory. Perlmutter is NERSC's primary computing system (Cori will be retired very soon).

- Perlmutter has 1792 GPU nodes and 3072 CPU nodes
  - GPU nodes: 1x AMD EPYC 7763 + 4x NVIDIA A100 processors
  - CPU nodes: 2x AMD EPYC 7763 processors
- ACE3P currently only runs on CPU processors
  - 64 cores per CPU (128 per node)
  - 512GB DDR4 memory per node
  - 39.2 GFlops per core (2.51 TFlops per CPU)
  - Additional information can be found at: https://docs.nersc.gov/systems/perlmutter/architecture/

ACE3P Workflow on NERSC systems

- Most pre- and post-processing will be performed on your own local laptop
- Simulations will be run on National Energy Research Scientific Computing Center (NERSC) systems



Secure SSH Connection

Your Laptop

- Cubit
- ParaView

NERSC Perlmutter

- acdtool
- ACE3P

Secure Connection to NERSC

Linux/iOS: normally installed by default in Unix terminals

- ssh – secure shell client
- scp – secure file transfer tool

Windows 10+: ssh/scp available via cmd or PowerShell

Optional software:

- PuTTY – secure shell client (alternative option)
  http://www.putty.org
- WinSCP – secure file transfer tool (GUI for file transfers)
  http://winscp.net
- NoMachine – virtual desktop client (GUI for NERSC)
  http://www.nomachine.com

Text Editor: gedit, emacs, vi, notepad, notepad++, etc.

Unix Basics

- **Copy files** using cp

  cp <source_file> <destination_file>

  Tip: Use cp -r to recursively copy directories
- **Remove files** using rm

  rm <file_name>

  Tip: use rm -r to recursively delete directories
- **Make directories** using mkdir

  mkdir <directory_name>
- **View current working directory** using pwd
- **Change file permissions** using chmod <code>

  chmod 755 <file_name>

  Tip: use chmod -R 755 to recursively update directories

  Tip: 755 code permissions are typical for most users

Editing files with vi

SLAC

Open 'vi' editor (built-in on NERSC systems) with

vi <file_name>

Basic Operations

- 'i' – enter 'insert' (edit) mode
- 'Esc' – exit 'insert' (edit) mode
- 'dd' – remove the current line
- 'u' – undo previous command
- ':w' – save the file
- ':q!' – quit without saving
- ':qw' – quit and save the file
- 'G' – go to the end of the file
- 'gg' – go to the beginning of the file

Logging into Perlmutter

SLAC

- From a suitable terminal run the command:

  ssh <username>@perlmutter-p1.nersc.gov

  Replace <username> with your NERSC user ID without angle brackets <>

- Type ‘yes’ when asked “Are you sure you want to continue connecting (yes/no)?”

- Enter your password+OTP when prompted
  - You might not see any * when you are typing
  - Your 6-digit one-time-password (OTP) is to be concatenated to your password without spaces

Setting up a working directory

- Once logged in, the base directory is your home directory '~' also linked with $HOME
- Copy ACE3P environment script to your home directory

  cp $CFS/ace3p/perlmutter/CPU/perlmutter-ace3p.sh .
- Create a folder cw23 on your scratch space

  mkdir $SCRATCH/cw23
- Tip: use echo $SCRATCH to view actual path
- Create a symlink 'cw23' for the scratch folder

  ln -s $SCRATCH/cw23 .
- Change directories using cd

  cd ~/cw23
- List the contents of a directory using ls

  ls ~/cw23

Transferring Files to/from NERSC with scp

SLAC

- Files can be transferred between your local computer and NERSC computers using the 'scp' command
- Prefix '<username>@dtn01.nersc.gov:' for remote files and directories ('dtn01' – 'dtn04' are data transfer nodes)
- To copy from your computer to the remote directory, use:

  scp <local_file> <username>@dtn01.nersc.gov:<remote_file>
- To copy from the remote directory to your computer, use:

  scp <username>@dtn01.nersc.gov:<remote_file> <local_file>
- To copy your entire NERSC home directory, use '-r' option:

  scp -r <username>@dtn01.nersc.gov:~/ <local_dir>
- Tip: use '.' for <local_dir> to copy to your current directory

Using sshproxy for repeated logins for Linux/iOS

SLAC

Use sshproxy instead of repeatedly using OTP

- **Linux/iOS:** download sshproxy.sh from NERSC:

  scp <username>@dtn01.nersc.gov:/global/cfs/cdirs/mfa/NERSC-MFA/sshproxy.sh .

- Run sshproxy.sh to generate a key

  ./sshproxy.sh -u <username>

- Enter your password+OTP as before
  - Three files will be created locally in ~/.ssh/nersc
- Login with ssh using the generated key ‘nersc’

  ssh -l <username> -i ~/.ssh/nersc perlmutter-p1.nersc.gov

- Generated keys last 24 hours

Using sshproxy for repeated logins for Windows

SLAC

Use sshproxy instead of repeatedly using OTP

- **Windows 10+:** download sshproxy.exe from NERSC:

  scp <username>@dtn01.nersc.gov:/global/cfs/cdirs/mfa/NERSC-MFA/sshproxy.exe .

- Run sshproxy.exe to generate a key in openssh format

  sshproxy.exe -u <username> --format openssh -o nersc

- Enter your password+OTP as before
  - The key file ‘nersc’ will be in the same directory
- Login with ssh using the generated key ‘nersc’

  ssh -l <username> -i nersc perlmutter-p1.nersc.gov

- Generated keys last 24 hours
- PuTTY-formatted keys (.ppk) can be generated with

  sshproxy.exe -u <username> -o nersc

ACE3P Code Path

The path for ACE3P executables are located at

$CFS/ace3p/perlmutter/CPU/<ace3p_module>

- Note: no need to download these (linking automatic)!

ACE3P modules (replace <ace3p_module> with a name):

- acdtool – pre- and post-processing tool
- omega3p – electromagnetic eigenmodes solver
- s3p – s-parameters solver
- t3p – time domain electromagnetic solver
- tem3p – multiphysics analysis
- track3p – multipacting and dark current solver
- gun3p – space-charge and beam optics solver

ACE3P Input Files and Results

SLAC

ACE3P environment must be loaded once per session

source ~/perlmutter-ace3p.sh

ACE3P example input files and results at NERSC

- Input Files: $CFS/ace3p/cw23/exercises
- Examples: $CFS/ace3p/cw23/examples

For CW23 you will need a local copy of the input files

- To download them visit the workshop materials page: https://confluence.slac.stanford.edu/display/AdvComp/Materials+for+CW23
- The bottom of the page has links to compressed file archives

ACE3P Input File Structure

ACE3P input files use the format: <project>.<ace3p_module>

For example: cw23/examples/omega3p/pillbox/**pillbox.omega3p**

```
ModelInfo : {
  File: ./pillbox4.ncdf

  BoundaryCondition : {
    Magnetic: 1, 2
    Exterior: 6
  }

  SurfaceMaterial : {
    ReferenceNumber: 6
    Sigma: 5.8e7
  }
}

FiniteElement: {
  Order:          2
  CurvedSurfaces: on
}
…
```

```
…

EigenSolver : {
  NumEigenvalues: 2
  FrequencyShift:  1.0e9
}

PostProcess : {
 Toggle: on
 ModeFile: mode
}
```

NERSC Batch Script File Structure

SLAC

NERSC uses the SLURM batch system

- ACE3P batch scripts are typically named: run-<ace3p_module>.batch

For example: cw23/examples/omega3p/pillbox/**run-omega3p.batch**

```
#!/bin/bash -l                    Run bash
#SBATCH -C cpu                    Specify CPU nodes for Perlmutter
#SBATCH -q debug                  Use 'debug' queue for testing, 'regular' for larger jobs
#SBATCH -A m349                   Use 'm349' allocation
#SBATCH -N 1                      Request 1 computing node (128 cores per node)
#SBATCH -t 00:10:00               Request 10 minutes of computing time
#SBATCH -J myjob                  Set the job name
#SBATCH -e ace3p.e%j              Write errors to .err file
#SBATCH -o ace3p.o%j              Write output to .out file
                                  Run solver using 16 tasks across 128 cores
srun -n 16 -c 16 --cpu-bind=cores /global/cfs/cdirs/ace3p/perlmutter/CPU/omega3p PillboxCoax.omega3p
```

See https://docs.nersc.gov/jobs/#commonly-used-options for additional information

Running Jobs on Perlmutter

SLAC

- Submit the job to queue using sbatch

  sbatch <your_batch_script>

- Check job status using squeue or sqs

  squeue -u <username>

  sqs

- Cancel jobs using scancel

  scancel <job_id>

- Check the job log/output as it runs using tail

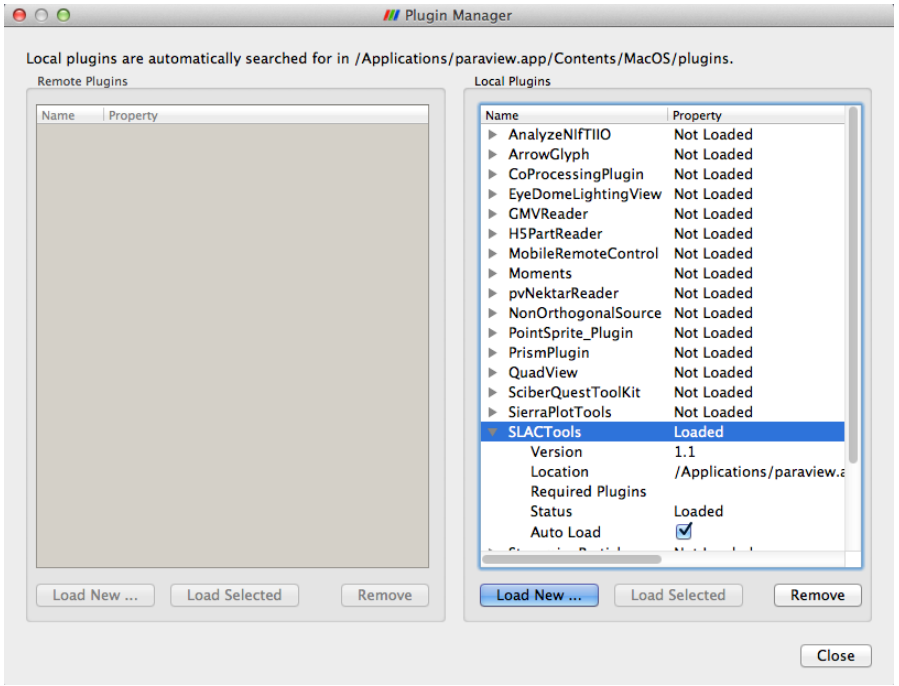
  tail -f <job_log_file>

- Tip: the job log/output files use the .log/.out suffixes

Configuring ParaView

Enable SLAC toolbar

- Tools > Manage Plugins…
- Select SLACTools
- SLACTools > Auto Load

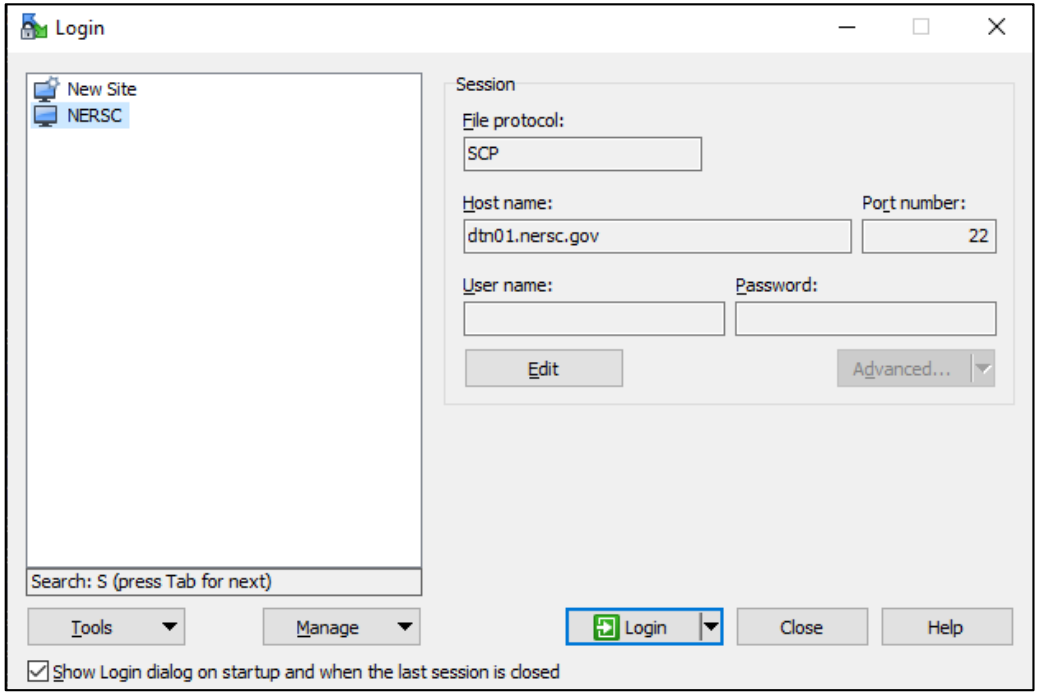


Load macros

- Macros > Add new macro…
- Go to '**cw23/macros**'
- Load the following macros one by one:
  - Track3P: **enhancement.py, resonant_lineplot.py, resonant_location.py, trajectory.py**
  - T3P: **wakeplot.py, timeFFT.py, fft.py**

Transferring Files to/from NERSC with WinSCP

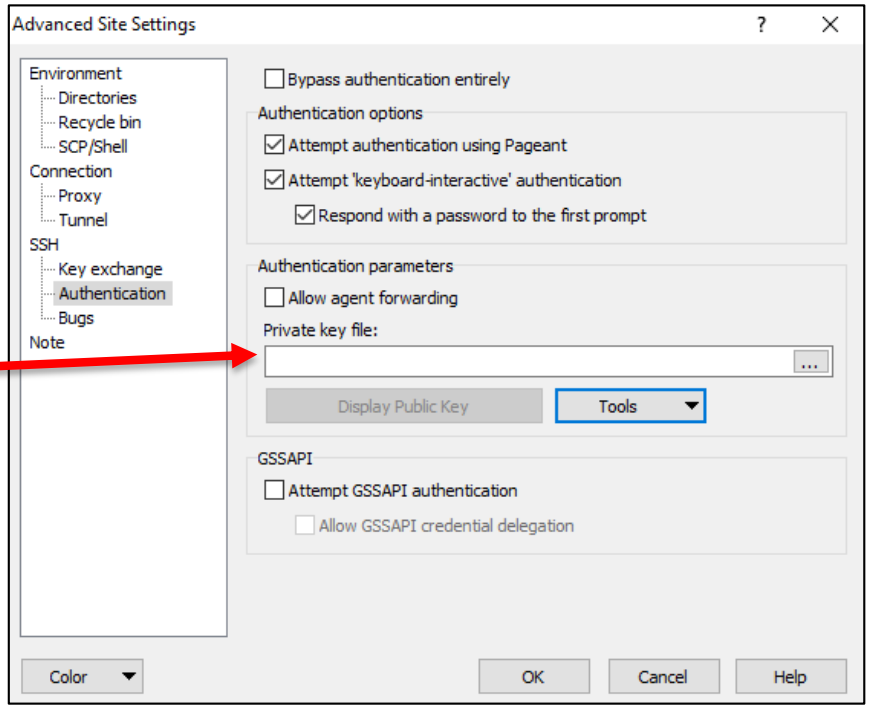
- Run WinSCP
- Create a new connection
- Set file protocol: ‘SCP’
- Set host name and port:
  ‘dtn01.nersc.gov’ and ‘22’
- Set your username
- Leave password blank
- Click save and login

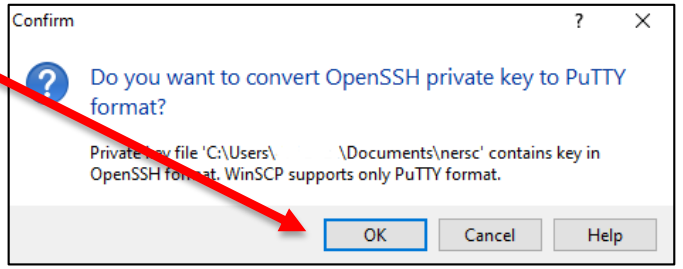


- Enter your NERSC password+OTP when prompted
- Once connected, copy files from one panel to another
- Tip: a built-in text editor is included in WinSCP

Using sshproxy with WinSCP

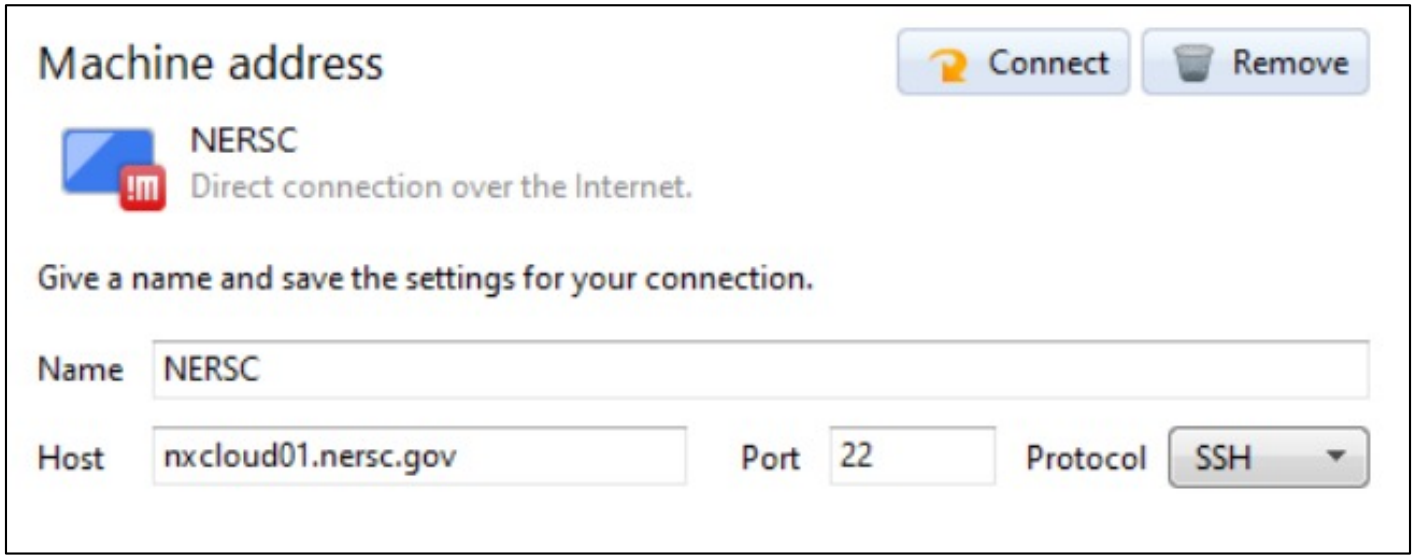
- Select NERSC connection
- Click ‘Edit’ > ‘Advanced’
- Select SSH > Authentication
- Select private key file
  - Click ‘…’ to browse
- Enter the ‘nersc’ key file path
  - If browsing, view all files (*.*)
- Agree to convert to PuTTY format
- With ‘nersc.ppk’ file created/set
- Click ‘OK’ > Click ‘Save’
- Note: this process must be redone daily, alternatively a new .ppk key can be created with sshproxy.exe daily





Setting up NoMachine

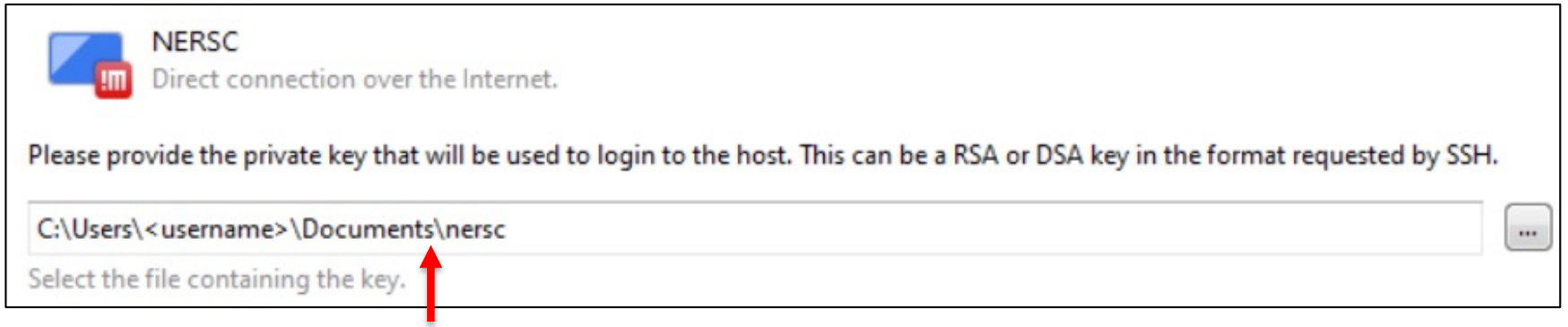
- **Download and install the NoMachine Enterprise Client:** https://www.nomachine.com/download-enterprise
- **Run NoMachine and add a connection with these details:**



- **Under configuration select to use a user-provided key:**

Setting up NoMachine

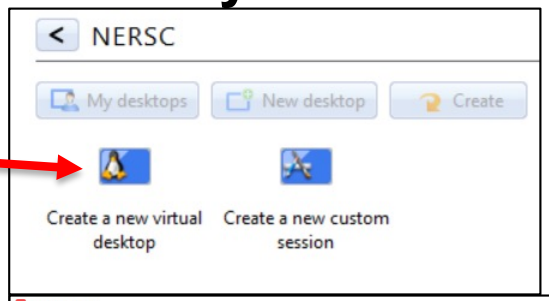
- **Locate the NERSC sshproxy key (openssh format):**



- Replace this path with where sshproxy.exe is installed!
- **Save these settings and close NoMachine**
- **Locate the "player.cfg" file and open it for editing:**
  - C:\Users\<username>\.nx\config\player.cfg
- **Edit the "SSH client mode" key from "library" to "native":**
  - <option key="SSH client mode" value="native" />
- **Save and close the "player.cfg" file**

Setting up NoMachine

- Reopen NoMachine: connect to the saved NERSC connection
- Enter your NERSC username in the popup window
- Select to “Yes” on the verify host authenticity window
- Once connected create a new virtual desktop
- Tutorial popups can be skipped



- These buttons open terminals to Perlmutter, Cori, DTN nodes

- For more information visit: https://docs.nersc.gov/connect/nx/