Getting Started at NERSC

Helen He
NERSC User Engagement Group

New User Training, February 23-24, 2017
Outline

• Connecting to NERSC
  – SSH, NX
• Computing Environment
• Compile and Run My First Job
• Common Best Practices
Connecting to NERSC
SSH

• All of the computational systems at NERSC are accessible via SSH
• Each system has a set of load-balanced login nodes which offer SSH service
• Use your NIM username & password
• Addresses for NERSC systems:

  Large-scale Systems
    – edison.nersc.gov
    – cori.nersc.gov

  Mid-range Systems
    – genepool.nersc.gov
    – pdsf.nersc.gov

Data Transfer Nodes
  – dtn[1-4].nersc.gov
Advanced Topic: SSH Keys

• If you choose to set up an SSH key to access NERSC systems, please use a passphrase on the key
  – No unencrypted keys!

• Upload your SSH public keys in NIM
  – Authentication available only to users who have stored their SSH public keys in NIM
  – Public keys stored in user home directories are not honored

• More Details: http://www.nersc.gov/users/network-connections/connecting-to-nersc/
Basic SSH use from Mac/Linux/cygwin

• If you have a UNIX-like computer, you can directly contact NERSC with your built-in SSH client
  1. Open a new terminal
  2. % ssh -l <NIM username> cori.nersc.gov

• Depending on your preferences you might want additional SSH flags:
  • ssh -Y  performs robust X-forwarding over ssh
  • ssh -A forwards ssh-agent information (if you use SSH keys)
SSH from a Windows System

• Many SSH clients exist for Windows
  – A very popular one is **putty**
    • [http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html](http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html)
  – Advanced users might prefer to use SSH directly within **mintty** (from Cygwin distribution)

• **Both of these options support all SSH features (that I’ve ever tried to use)**
  – For X-forwarding to work, you’ll need to find X-server software
    • Cygwin/X
    • Exceed
  – Consider using NX instead of X-forwarding
X-forwarding

• Allows you to access Visualization programs remotely at NERSC

Example:

```
localhost% ssh -l elvis -Y cori.nersc.gov
...
```

e/elvis> module load matlab
e/elvis> matlab

$MATLAB$ starts up

NERSC Recommends using NX instead of X-forwarding.

Next section!
Example Session

localhost:~elvis> ssh -l <NIM username> cori.nersc.gov

*****************************************************************
NOTICE TO USERS

Lawrence Berkeley National Laboratory operates this computer system under contract to the U.S. Department of Energy. This computer system is the property of the United States Government and is for authorized use only. Users (authorized or unauthorized) have no explicit or implicit expectation of privacy.

Any or all uses of this system and all files on this system may be intercepted, monitored, recorded, copied, audited, inspected, and disclosed to site, Department of Energy, and law enforcement personnel, as well as authorized officials of other agencies, both domestic and foreign. By using this system, the user consents to such interception, monitoring, recording, copying, auditing, inspection, and disclosure at the discretion of authorized site or Department of Energy personnel.

Unauthorized or improper use of this system may result in administrative disciplinary action and civil and criminal penalties. By continuing to use this system you indicate your awareness of and consent to these terms and conditions of use. LOG OFF IMMEDIATELY if you do not agree to the conditions stated in this warning.
*****************************************************************

Password: <enter your NIM password here>

Prompt on local system
Notification of acceptable use.
Password prompt
- **MOTD (NERSC Message of the Day)** -

- **After you type the password and logged in to a system, you will see NERSC MOTD before your session prompt appears**

<table>
<thead>
<tr>
<th>Last login: Wed Feb 22 16:07:29 2017 from 198.128.212.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact Information</td>
</tr>
<tr>
<td>NERSC Contacts <a href="http://www.nersc.gov/about/contact-us/">http://www.nersc.gov/about/contact-us/</a></td>
</tr>
<tr>
<td>NERSC Status <a href="http://www.nersc.gov/users/live-status/">http://www.nersc.gov/users/live-status/</a></td>
</tr>
<tr>
<td>NERSC: 800-66-NERSC (USA) 510-486-8600 (outside continental USA)</td>
</tr>
<tr>
<td>Current Status as of 2017-02-22 14:35 PST</td>
</tr>
<tr>
<td>Compute Resources:</td>
</tr>
<tr>
<td>Cori: Available.</td>
</tr>
<tr>
<td>Edison: Available.</td>
</tr>
<tr>
<td>Genepool: Available.</td>
</tr>
<tr>
<td>PDSF: Available.</td>
</tr>
<tr>
<td>Global Filesystems:</td>
</tr>
<tr>
<td>DNA: Available.</td>
</tr>
<tr>
<td>Global Common: Available.</td>
</tr>
<tr>
<td>Global Homes: Available.</td>
</tr>
<tr>
<td>Project: Available.</td>
</tr>
<tr>
<td>ProjectA: Available.</td>
</tr>
<tr>
<td>ProjectB: Available.</td>
</tr>
</tbody>
</table>

| Mass Storage Systems:                                   |
| HPSS Backup: Available.                                 |
| HPSS User: Available.                                   |

| Service Status as of 2017-02-22 14:35 PST               |
| All services are available.                             |

| Planned Outages                                         |
| Cori: 02/28/17 6:00-03/01/17 6:00 PST, Scheduled maintenance. Cori will be degraded due to cabinet additions. Datawarp nodes will be reduced during this time. |
| Cori: 03/01/17 6:00-03/03/17 17:00 PST, Scheduled maintenance. Cori will be down for adding cabinets and HSN (high-speed network) maintenance. Logins will not be available. |
| Data Transfer Nodes: 03/01/17 9:00-12:00 PST, Scheduled maintenance. |

| Past Outages                                            |
| Cori: 02/21/17 8:00-21:15 PST, Scheduled maintenance. Cori will be unavailable while updates are applied. Logins will be available, however no jobs will run. |
| Cori: 02/21/17 21:15-22:15 PST, System in degraded mode. The majority of the system’s compute nodes are currently unavailable. Engineers are investigating the issue |

For past outages, see: http://my.nersc.gov/outagelog-cs.php

- 10 -
Login Node Auto-Logout

- Some NERSC systems won’t give you unlimited time on the login nodes
  - After 48 hours idle, Cori & Edison login nodes will terminate your session
  - PDSF and Genepool sessions are unlimited
NX – Accelerated X

• Also uses SSH
• Persistent sessions
• Accelerated Graphics
  – really good for remote access
• KDE Desktop
• What you need for NX
  – Any Desktop/Laptop
    • Windows/Linux/Mac
  – NX Client Software (Free)
Reasons for NX

• **Slow Speeds:** X-Windows is slow over network. Remote windows from emacs can take minutes to open

• **Solution:** NX Buffers/Compresses X messages, giving much better X experience
Reasons for NX

- **Long Lasting Desktop:** NX gives you a desktop, so you can connect to NERSC resources (such as Edison) and start your GUI applications.
Reasons for NX

• **Lost Connections:** If I lose internet connection, I might lose all running processes.

• **Solution:** NX provides sessions. You can suspend the session without terminating the running processes.
  – And get back to the same point when reconnected, *even from a different location or computer.*
NERSC NX Service

• 10 Minute Start-up Guide

Documentation:
Go to www.nersc.gov, search for “NX”

Map of Current Users
Lisa Gerhardt will show a short live NX demo after this talk
Computing Environment
Node Types

• Login nodes
  – Shared with other users
  – Code compilation, job preparation and submission

• Compute nodes
  – Not shared (except in the “shared” partition)
Login Node Configuration

- **Edison**
  - 12 nodes
    - 16 cores, 2.0 GHz Intel Sandy Bridge, 512 GB

- **Cori**
  - 12 nodes
    - 32 cores, 2.3 GHz Intel Haswell, 512 GB
    - Extra login nodes for special purposes (not in load balancer)

- **Genepool**
  - 2 nodes
    - 32 cores, 2.3 GHz Intel Haswell, 128 GB

- **PDSF**
  - 3 nodes
    - 32 cores, 2.6 GHz Intel Haswell, 128 GB
Login Node Access

• Connect (via SSH) to load balancer
  % ssh edison.nersc.gov
  % ssh cori.nersc.gov
  % ssh genepool.nersc.gov
  % ssh pdsf.nersc.gov

• Load balancer selects login node based on:
  – Number of connections
  – Memory of previous connections from same IP
Login Node Usage

• Login nodes are shared by many users, all the time
• Edit files, compile programs, submit batch jobs
• *Some light* post-processing/data analysis
  – IDL, MATLAB, NCL, python, etc.
  – All can run on compute nodes (so you have dedicated resources)
• *Some* file transfers
  – Use data transfer nodes for large/long-running transfers
• Please use discretion
  – *All* users get frustrated by sluggish interactive response
Login Node Guidelines

• Use *no more* than 50% of available cores
• Use *no more* than 25% of available memory
• Limit use of parallel “make”
  \[% make -j 4 all\]
• NERSC will kill user processes if login nodes become unacceptably slow or unresponsive
• Terminate idle sessions of licensed software
  – IDL
  – MATLAB
  – Mathematica
Shell Initialization Files

• Standard dot files are maintained by NERSC
  – .bashrc, .profile, .cshrc, .login, etc.
  – Symbolic links to read-only files

• Personal dot files
  – Aliases, environment variables, modules, etc.
  – Use .ext suffix (“.ext files”) .bashrc.ext, etc.

• Broken? Use “fixdots” to start over
  – Creates $HOME/KeepDots.<timestamp>
  – Restores all dot files to default state
  – If PATH corrupted:
    /usr/common/software/bin/fixdots

• Use NIM to change default login shell
Software is Managed by Modules

• Identify the software you need
  
  http://www.nersc.gov/users/software/
  
  – Use `module avail package_name`
    
    • *Lots* of output
      
      – All module output goes to stderr, not stdout
    
    • Each system has different modules!

• Load the module

  % which idl
  
  idl: Command not found.
  
  % module load idl
  
  % which idl
  
  /global/common/cori/software/idl/idl83/bin/idl
Other Useful Module Commands

`module unload <modulename>`

– Remove the module from your environment

`module swap <module1> <module2>`

– Unload one module and replace it with another

```
% module swap intel intel/16.0.3.210
```
(replace current default to a specific version)

`module list`

– See what modules you have loaded right now

`module show <modulename>`

– See what the module actually does

`module help <modulename>`

– Get more information about the software
NERSC Supported Software

- NERSC provides a wide range of software
- [http://www.nersc.gov/users/software/](http://www.nersc.gov/users/software/)
  - Scientific Applications
    - VASP, Amber, NAMD, Quantum Espresso, ...
  - Compilers
    - Intel, GCC, Cray
  - Scripting Languages
    - perl, python, R - including common packages for each
  - Software Libraries (some maintained by Cray)
    - blas/lapack (MKL), boost, hdf5, netcdf, ...
  - Development utilities
    - git, mercurial, cmake, shifter, ...
  - Debuggers and Profilers
    - DDT, TotalView, gdb, Perftools, MAP, Darshan, IPM, Vtune
  - Grid Software
    - Globus
  - Visualization and Analytics packages
    - Visit, ParaView, Jupyter, Rstudio, ...
  - Development Environment
    - Shifter
Cray Programming Environment

• Meta-modules
  PrgEnv-intel, PrgEnv-cray, PrgEnv-gnu
  – Organize a set of modules
    • Compiler (intel, cray, gnu)
    • Libraries (including MPI) tuned for compiler
  – Intel is default on Edison and Cori

• Swapping Programming Environments
  % module swap PrgEnv-intel PrgEnv-cray
  – swaps compiler
  – no need to swap libraries!
Compiler Wrappers

• **On Cori / Edison:**
  – Defined by `PrgEnv-*` modules
  – `ftn` (fortran), `cc` (C), `CC` (C++)
  – Provides include header and library search paths for MPI, common math libraries (e.g., Cray Libsci), Cray system software
  – Provides consistent level of optimization across compilers

• **Use compiler wrappers to build applications**
• **Seldom need native compilers!**
• **More details in a Compiling Codes talk later today**
CHOS Environment

• Provides different OS environments
  – Often different third-party software
    • Some software packages have specific OS requirements
      – Possibly due to validation requirements

• Used on PDSF and Genepool

• Transparent
  – Default configuration for most users
  – Alternate configurations for some users

• More Details
  http://www.nersc.gov/users/computational-systems/pdsf/software-and-tools/chos/
Compile and Run My First Job (Cori Haswell example)
My Hello World Program

elvis@cori04> cat mpi-hello.f90

IMPLICIT NONE
INCLUDE 'mpif.h'

INTEGER :: myPE, totPEs, ierr
CALL MPI_INIT( ierr )

CALL MPI_COMM_RANK( MPI_COMM_WORLD, myPE, ierr )
CALL MPI_COMM_SIZE( MPI_COMM_WORLD, totPEs, ierr )
PRINT *, "myCPU is ", myPE, " of total ", totPEs

CALL MPI_FINALIZE(ierr)
STOP
END
• Use compiler wrappers which links MPI libraries automatically.

elvis@cori04> ftn -o mpi-hello mpi-hello.f90
elvis@cori04> ls -al mpi-hello
-rwxr-x--- 1 elvis elvis 9241160 Feb 22 10:14 mpi-hello
Submit Batch Job

• Prepare a Slurm batch script

```
elvis@cori04> cat run-hello.sl
#!/bin/bash

#SBATCH -N 2       # Use 2 compute node
#SBATCH -t 00:10:00 # Set 10 minute time limit
#SBATCH -p debug   # Submit to the "debug" partition
#SBATCH -L SCRATCH # Job requires $SCRATCH file system
#SBATCH -C haswell # Request Haswell nodes

srun -n 64 ./mpi-hello
```

• Submit it to the batch queue

```
elvis@cori04> sbatch run-hello.sl
Submitted batch job 3838675
```
Check Results

- Check job in batch queue

```bash
elvis> sqs
JOBID PARTITION  ST  USER  NAME          NODES  USED    REQUESTED   SUBMIT
3838675 debug    PD  Priority  elvis        2      0:00    10:00       2017-02-22T10:24:32
```

- Once it is completed, check results

```bash
elvis> cat slurm-3838675.out
myCPU is 0 of total 64
myCPU is 1 of total 64
myCPU is 2 of total 64
...
myCPU is 61 of total 64
myCPU is 62 of total 64
myCPU is 63 of total 64
```

- More details on running jobs in a later talk today
Common Best Practices
Selected Best Practices (1)

• Check MOTD messages for current system status, past outages, and planned maintenances
  – From SSH login prompt
  – http://www.nersc.gov/live-status/motd/

• Be nice to others regarding the shared resources
  – Limit CPU and memory usage on login nodes
  – Do production work on compute nodes

• Release licenses
  – Limited number of certain software packages are available, such as Matlab, IDL, etc.
Selected Best Practices (2)

• Don’t use “watch” with default 2 seconds interval
  – Check every 10 min or more
  – Send emails when batch job starts and ends
    • #SBATCH –mail-type=<events>
      – Valid events: BEGIN, END, FAIL, etc.
    • #SBATCH –mail-user=<email_address>

• Run applications from Lustre scratch or /project instead of global homes directory, to get
  – Larger space
  – Optimal IO performance

• Back up your important files frequently
  – /scratch files are purged
Further Information

- [http://www.nersc.gov/users/getting-started/](http://www.nersc.gov/users/getting-started/)
- [https://www.nersc.gov/users/computational-systems/cori/getting-started/your-first-program-on-cori/](https://www.nersc.gov/users/computational-systems/cori/getting-started/your-first-program-on-cori/)
Thank you.