NESCO HPC Group A End User Training

Dan Schornak (CSC)
Shawn Hoopes (Adaptive Computing)
0800-0810  Training Process
  • Introduce instructors
  • General information
  • Define training roles and responsibilities
  • Review training agenda

0810-0815  Training Objectives
  • Review the objectives of the NESCC HPC Environment overview

0815-0845  Introduction to NESCC HPC Design/Architecture and Environment
  • Describe the overall architecture major components, functions, and features

0845-0900  Available On-line Documentation

0900-0915  BREAK
<table>
<thead>
<tr>
<th>Time</th>
<th>Session</th>
</tr>
</thead>
<tbody>
<tr>
<td>0915-0945</td>
<td>Introduction to User Environment, Policies, Quotas, and Environmental Variables</td>
</tr>
<tr>
<td>0945-1045</td>
<td>Introduction to Programming Environment</td>
</tr>
<tr>
<td></td>
<td>• Introduce tools – shells, applications, compilers, and libraries</td>
</tr>
<tr>
<td>1045-1145</td>
<td>Introduction to NESCC Storage and Data Management</td>
</tr>
<tr>
<td>1145-1245</td>
<td>LUNCH BREAK</td>
</tr>
<tr>
<td>1245-1415</td>
<td>Introduction to Job Scheduling using Moab (Adaptive Computing)</td>
</tr>
<tr>
<td></td>
<td>• Efficient job submission techniques</td>
</tr>
<tr>
<td></td>
<td>• Installed NESCC version compared to other NOAA installations</td>
</tr>
<tr>
<td></td>
<td>• Moab/Torque end-user commands and tools</td>
</tr>
<tr>
<td>1415-1500</td>
<td>Questions/Answers</td>
</tr>
<tr>
<td>1500</td>
<td>Session Wrap up</td>
</tr>
</tbody>
</table>
Need Help? – Where to go...

**User Support and Problem Reporting (Help Desk for NESCC)**
help.nescc@noaa.gov

**NESCC Documentation Wiki Site – “How to…”**
https://nesccdocs.rdhpcs.noaa.gov/

**End User Experience**
EMC Model Transition Team (Provides updates for porting to TDS and Zeus)
http://www.emc.ncep.noaa.gov/mtt

**NESCC Factoid…**
“Additional documentation will be made available on NesccDocs soon.”
Need Help? – Where to go...

NESCC Documentation Wiki Site – “How to…”
https://nesccdocs.rdhpcs.noaa.gov/
Need Help? – Where to go...

End User Experience

EMC Model Transition Team (Provides updates for porting to TDS and Zeus)
http://www.emc.ncep.noaa.gov/mtt
**TRAINING OBJECTIVES (Group A)**

<table>
<thead>
<tr>
<th>What it is...</th>
<th>What it isn’t...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction to NESCC HPC Environment</td>
<td>System Administrator –level Training</td>
</tr>
<tr>
<td>User Environment Overview</td>
<td>Unix scripting Overview</td>
</tr>
<tr>
<td>Programming Environment Overview</td>
<td>Application/System Tuning Overview</td>
</tr>
<tr>
<td>Introduction to Moab Workload Manager/Torque Resource Manager</td>
<td>Moab Administrator-level Training</td>
</tr>
</tbody>
</table>

*Disclaimer: As system integration continues, some training content may become obsolete. For the latest guidance on how to use these NESCC HPC resources, please visit the following site regularly:*

https://nesccdocs.rdhpcs.noaa.gov
# NESCC HPC Design/Architecture and Environment

<table>
<thead>
<tr>
<th>Type</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bastion</td>
<td>All login sessions are routed through these hosts. They provide additional security to protect NOAA assets.</td>
</tr>
<tr>
<td>Service Nodes</td>
<td>Where initial login sessions are created. They are for general interactive use for operations such as editing and compiling code, managing files, initiating actions on external network including data transfer and accessing software repository.</td>
</tr>
<tr>
<td>Data Transfer Nodes (DTN)</td>
<td>Provide high speed inbound and outbound data transfers.</td>
</tr>
<tr>
<td>Compute System</td>
<td>Major compute resources for parallel jobs. Accessible through the batch system.</td>
</tr>
<tr>
<td>Big Memory Nodes</td>
<td>Large memory set for applications that require more memory than a single compute node. Accessible through the batch system.</td>
</tr>
<tr>
<td>High Performance Storage</td>
<td>High speed filesystems</td>
</tr>
<tr>
<td>Home Filesystem</td>
<td>Storage system optimized for editing and compiling code.</td>
</tr>
</tbody>
</table>
NESCC HPC Design/Architecture and Environment
Zeus/Herc System Architecture

**Service Nodes**

- Front End 1
- Front End 2
- Front End 3
- Front End N

**Big Memory Nodes**

- Big Mem 1
- Big Mem 2
- Big Mem 3
- Big Mem N

**Compute System**

- Rack 1
- Rack 2
- Rack 3
- Rack 4
- Rack N

**Networks**

- Ethernet
- Infiniband

**Storage**

- Home FS (BlueArc)
- HPS Filesystem 1 (Lustre)
- HPS Filesystem 2 (Lustre)

**Other Nodes**

- Bastion
- DTN 1
- NWAVE 10Gbe

Get Help! @ help.nescc@noaa.gov
## Herc (TDS) System Configuration

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Nodes</td>
<td>1</td>
</tr>
<tr>
<td>Big Memory Nodes</td>
<td>0</td>
</tr>
<tr>
<td>Compute Nodes</td>
<td>48</td>
</tr>
<tr>
<td>Compute Cores</td>
<td>576</td>
</tr>
<tr>
<td>Total Flops</td>
<td>7.9TF</td>
</tr>
<tr>
<td>Total Data Transfer Nodes</td>
<td>0</td>
</tr>
<tr>
<td>HPS Capacity</td>
<td>????</td>
</tr>
<tr>
<td>HPS Performance</td>
<td>????</td>
</tr>
</tbody>
</table>

- There are no big memory nodes apart of Herc
- Data Transfer is to/from NESCC, so no nodes are assigned to Herc
Zeus System Configuration

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Nodes</td>
<td>8</td>
</tr>
<tr>
<td>Big Memory Nodes</td>
<td>6</td>
</tr>
<tr>
<td>Compute Nodes</td>
<td>2304</td>
</tr>
<tr>
<td>Compute Cores</td>
<td>27648</td>
</tr>
<tr>
<td>Total Flops</td>
<td>383 TF</td>
</tr>
<tr>
<td>Total Data Transfer Nodes</td>
<td>2+</td>
</tr>
<tr>
<td>HPS Capacity</td>
<td>5.8 PB</td>
</tr>
<tr>
<td>HPS Performance</td>
<td>&gt; 50 GB/s</td>
</tr>
</tbody>
</table>

• There will be several DTN types and depending on location of originating connection. Policy is being finalized by NOAA management.
### General System Information

<table>
<thead>
<tr>
<th>Node Type</th>
<th>Total Cores</th>
<th>Core Speed</th>
<th>Memory</th>
<th>Local Storage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service</td>
<td>12</td>
<td>3.46GHz</td>
<td>48 GB</td>
<td>Yes</td>
</tr>
<tr>
<td>Compute</td>
<td>12</td>
<td>3.46GHz</td>
<td>24 GB</td>
<td>No</td>
</tr>
<tr>
<td>Big Memory</td>
<td>12</td>
<td>3.46GHz</td>
<td>96 GB</td>
<td>No</td>
</tr>
</tbody>
</table>

**Notes:**

- There are no CRON Nodes. CRON Services will be provided on Zeus through a NOAA developed system that unifies the crontab across all login nodes.
Test and Development System (TDS)

Test LSC System Components

- **TDS Compute**: SGI Altix ICE 8400 EX
- **TDS Login Node**: SGI/Rackable CH-1103-RK-TY12
- **TDS Administration Node**: SGI/Rackable CH-1103-RK-TY12
- **TDS Batch Node**: SGI/Rackable CH-1103-RK-TY12
- **TDS High Performance Storage (HPS)**: SGI IS16000

**Login Node**: tfe1

---

**TDS (a.k.a Herc) - SGI ICE**

- 48 nodes, 3.46 GHz Intel Westmere
- 12 cores per node
- 2GB per core (24GB per node)
- QDR dual-rail Infiniband (but MPI jobs will only be using one rail)

**Lustre**

The fast parallel filesystem mount points are /mnt/tds_lustre1 and /mnt/tds_lustre2. You can create directories for your use under the `users` directory.

NESCC HPC Design/Architecture and Environment

Production System

Production LSC System (Zeus)
- LSC Compute……..SGI Altix ICE 8400 EX
- LSC Data Movers…..SGI/Rackable CH-1103-RK-TY12
- LSC Login Nodes…..SGI/Rackable CH-1103-RK-TY12

Production High Performance Storage (HPS)...SGI IS16000
- HPS Metadata Server Nodes (MDS)…….SGI/Rackable C3108-TY11
- HPS Metadata Target (MDT)……………..SGI IS6120
- HPS Object Storage Servers (OSS)……...SGI/Rackable CH-1103-RK-TY12
- HPS Object Storage Target (OST)……….SGI Infinite Storage 16K Couplet

Login Nodes: fe1
- fe2
- fe3
- fe4
- fe5
- fe6
- fe7
- fe8

SGI ICE – Compute Nodes
- 2304 1P-105 blades
- 4608 CPUs, X5690 e.46GHz 6-Core
- 24 GB DDR3 1333MHz2 cores per node
  - 4GB per core (48GB per node)
- QDR dual-rail Infiniband (but MPI jobs will only be using one rail)

Lustre
The fast parallel filesystem mount points are /mnt/tds_lustre1 and /mnt/tds_lustre2. You can create directories for your use under the users directory.
NESCC HPC Design/Architecture and Environment

Production Environment Components

Login Nodes
- 8 Small Memory nodes (memory*: 48 GB DDR3 1333MHz)
- 6 Big Memory nodes (memory*: 96 GB DDR3 1333MHz)
  - Twice the memory of Small Memory nodes
  - Managed by batch system and used for special jobs
* Reflect total memory size; User-available memory is lower

Test Storage System – IBM HPSS
* Core Server
* Spare Server
* Data Mover Server
* VFS Servers
* Management Server
* D3512 Storage Arrays
* DS3524 Storage Arrays

Production Storage System – IBM HPSS
* Core Server
* Spare Server
* Data Mover Servers
* VFS Servers

Production Tape Silo – Oracle SL8500 with 32 T10000c tape drives

Migration and Backup Silo – IBM TS3500 Silo with 8 TS-1130 and 8 LTO tape drives

Backup/Restore System – Four Dell R710 servers

NCEP Silos – Three 9310 Powderhorn Silos with 30 TS1130 tape drives
* Installed September 2011
* Temporary component for data migration
* Estimated removal – May 2012
Wide Area Network
Available Online Documentation

NESCC Documentation Wikisite:  https://nesccdocs.rdpcs.noaa.gov/
EMC Model Transition Team Website:  http://www.emc.ncep.noaa.gov/mtt

SGI Documentation
• SGI Altix ICE 8400 Product Information:  http://www.sgi.com/products/servers/ice/8400/

BlueArc Documentation

Adaptive Computing
• Moab Workload Manager User’s Manual:  http://www.adaptivecomputing.com/resources/docs/mwm/6-0/moabusers.php

Direct Data Networks Documentation

IBM Documentation
• HPSS User Guides:  http://www.hpss-collaboration.org/user_doc.shtml
• IBM Total Storage 3584 (now TS3500) Tape Library Information:  http://www-03.ibm.com/systems/storage/tape/ts3500/index.html

Oracle Documentation

Archival Documentation
• HTAR Man Page:  http://www.mgleicher.us/GEL/htar/htar_man_page.html
• HTAR User Guide:  http://www.mgleicher.us/GEL/htar/htar_user_guide.html
<table>
<thead>
<tr>
<th>Time</th>
<th>Session</th>
</tr>
</thead>
<tbody>
<tr>
<td>0800-0810</td>
<td>Training Process</td>
</tr>
<tr>
<td></td>
<td>• Introduce instructors</td>
</tr>
<tr>
<td></td>
<td>• General information</td>
</tr>
<tr>
<td></td>
<td>• Define training roles and responsibilities</td>
</tr>
<tr>
<td></td>
<td>• Review training agenda</td>
</tr>
<tr>
<td>0810-0815</td>
<td>Training Objectives</td>
</tr>
<tr>
<td></td>
<td>• Review the objectives of the NESCC HPC Environment overview</td>
</tr>
<tr>
<td>0815-0845</td>
<td>Introduction to NESCC HPC Design/Architecture and Environment</td>
</tr>
<tr>
<td></td>
<td>• Describe the overall architecture major components, functions, and features</td>
</tr>
<tr>
<td>0845-0900</td>
<td>Available On-line Documentation</td>
</tr>
<tr>
<td>0900-0915</td>
<td><strong>BREAK</strong></td>
</tr>
<tr>
<td>0915-0945</td>
<td>Introduction to User Environment, Policies, Quotas, and Environmental Variables</td>
</tr>
<tr>
<td>0945-1045</td>
<td>Introduction to Programming Environment</td>
</tr>
<tr>
<td></td>
<td>• Introduce tools – shells, applications, compilers, and libraries</td>
</tr>
<tr>
<td>1045-1145</td>
<td>Introduction to NESCC Storage and Data Management</td>
</tr>
<tr>
<td>1145-1245</td>
<td><strong>LUNCH BREAK</strong></td>
</tr>
<tr>
<td>1245-1415</td>
<td>Introduction to Job Scheduling using Moab (Adaptive Computing)</td>
</tr>
<tr>
<td></td>
<td>• Efficient job submission techniques</td>
</tr>
<tr>
<td></td>
<td>• Installed NESCC version compared to other NOAA installations</td>
</tr>
<tr>
<td></td>
<td>• Moab/Torque end-user commands and tools</td>
</tr>
<tr>
<td>1415-1500</td>
<td>Questions/Answers</td>
</tr>
<tr>
<td>1500</td>
<td>Session Wrap up</td>
</tr>
</tbody>
</table>
Introduction to User Environment, Policies, Quotas, and Environmental Variables

Section Objectives
1. Investigate various login approaches to TDS/Zeus
2. Review and explore the User Environment
3. Learn how to establish an SSH Tunnel
4. Knowing the Limitations – Established Policies and Quotas
LOGGING IN – Getting from here to there

PuTTY Example

1. Start PuTTY (Host Name: zeus.rdhpcs.noaa.gov)
2. Log into zeus
3. Hit ^C within 5 seconds
4. Enter “tfe1”
5. You’re in!

If password-less login has not yet been established
1. Enter the default password: Pa$$w0rd  (that’s zero!)
2. Upon entry, please change your password (passwd)

NESCC Factoid…
“Account Names could be in the form of…”
“John.Public” or “John.Q.Public”
(Be safe…stay case sensitive)
1. Start PuTTY (Host Name: zeus.rdhpcs.noaa.gov)
2. Log into zeus
3. Hit ^C within 5 seconds
4. Enter “tfe1”
5. You’re in!

If password-less login has not yet to be established
1. Enter the default password: Pa$$w0rd (that’s zero!)
2. Upon entry, please change your password (passwd)
NESCC Factoid…

“Avoid connection timeouts by sending ‘keepalive’ packets”
TUNNELLING – Getting data from here to there

Linux Example

1. Get Your Personal Port Number:
   a. Log in through Zeus (zeus.rdhpcs.noaa.gov)
      
      ssh Daniel.Schornak@zeus.rdhpcs.noaa.gov
   
   b. Enter password and PIN
   
   c. Hit ^C within 5 seconds
   
   d. Enter “tfe1” (for TDS) or “fe1” (For Zeus)
   
   e. Find port tunnel ID in Welcome Message
   
   f. Log out of the session

2. On Linux station, log in with tunnel attributes:

   ssh -L21796:localhost:21796 Daniel.Schornak@zeus.rdhpcs.noaa.gov

3. Verify tunnel is established in Welcome Message

NESCC Factoid…
“Port number is unique for each user.”
1. Get Your Personal Port Number:
   a. Log in through Zeus (zeus.rdhpcs.noaa.gov)
   b. Hit ^C within 5 seconds
   c. Enter “tfe1” (for TDS) or “fe1” (For Zeus)
   d. Find port tunnel ID in Welcome Message
   e. Log out of the session

2. On Linux station, log in with tunnel attributes:
   ssh -L21796:localhost:21796 Daniel.Schornak@zeus.rdhpcs.noaa.gov

3. Verify tunnel is established in Welcome Message
TUNNELLING – Getting data from here to there

Linux Example (continued)

Create or edit ~/.ssh/config
Add the following lines on the client:

HOST *
  ServerAliveInterval 60
^

Note: Add space in front of “ServerAliveInterval”

NESCC Factoid…
“Avoid connection timeouts by sending ‘keepalive’ packets”
Getting data from here to there

**WinSCP Example**

1. Start WinSCP
2. In Session Window type: `zeus.rdhpcs.noaa.gov`
3. Select File Protocol: **SCP**
4. Select: "**Advanced options**" button
5. Select "**Tunnel**"
6. Select “**Connect through SSH tunnel**” button
7. Enter Host name: `zeus.rdhpcs.noaa.gov`
8. Enter your user name (case sensitive)
9. Enter your NOAA-provided Local tunnel port number
10. Save the session (if you choose) or select “**Login**”

**Prerequisite:** Active PuTTY session must be running
11. Server prompt window appears
12. Enter your <NOAA User Name><RSA PIN>
13. Authentication Banner window appears
14. Select: “Continue”
15. Password Window appears
16. Enter local NESCC password
TUNNELLING – Getting data from here to there

WinSCP Example (continue)

17. WinSCP Window appears
TUNNELLING – *Getting data from here to there*

Copying Data

To use the port tunnel for SCP from your host:

----------------------------------------------

**Template:**

- `scp -P 21796 /local/path/to/file Daniel.Schornak@localhost:/remote/path/to/file`
- `scp -P 21796 Daniel.Schornak@localhost:/remote/path/to/file /local/path/to/file`

(Windows PUTTY users will use `pscp` instead of `scp`)

**IMPORTANT NOTE:**

=*=*=*=*=*=*=*=*=*=*=*=*=*

ALWAYS USE 'localhost' for the SCP commands so the port tunnel is used.
Data Transfer Nodes are being established to permit large data transfers from/to locations external to NESCC, but still within the NOAA WAN.

Use policies and implementation schemes are currently being developed until finalized, it is recommended that SCP or PSCP be used to transfer data.

Jet account users
You can use the jetscp hosts for 2 party transfers. While that requires an additional hop, transfers are much faster (100+ MB/s). You need to establish access as you would from any remote host.

https://jetdocs.rdpcs.noaa.gov/wiki/index.php/Transferring_Files#Option_2:_jetscp
## User Environment – *Environment Variables*

<table>
<thead>
<tr>
<th>Environment Variable</th>
<th>Example Output (echo &lt;Environment Variable&gt;)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$HOME</td>
<td>/home/daniel.schornak</td>
<td>Typically, the initial directory after logging in</td>
</tr>
<tr>
<td>$PATH</td>
<td>/usr/local/bin:/usr/local/X11/bin:/etc:/usr:/etc:/usr/local/etc:.</td>
<td>Provides search path for shell commands. (User modifiable)</td>
</tr>
</tbody>
</table>
| $SHELL               | /bin/ tcsh                                  | Current active UNIX shell for session (User modifiable) Available shells include:  
|                      |                                             | sh (bourne SHell); bash (Bourne-Again SHell);  
|                      |                                             | csh (C SHell); ksh (Korn SHell);  
|                      |                                             | tcsh (TC SHell); zsh – Z SHell |
| $PWD                 | /home/daniel.schornak                       | Print Working Directory. Changing directory locations updates this environment variable |
| $LOGNAME             | daniel.schornak                             | Name of the user |
| $MAIL                | /var/spool/mail/daniel.schornak             | Location of incoming mail |
| $USER                |                                             | Name of the user |
### User Environment - Environment Variables

#### NESCC
- **LANG** = en_US.UTF-8
- **USER** = Daniel.Schornak
- **LOGNAME** = Daniel.Schornak
- **HOME** = /home/Daniel.Schornak
- **SHELL** = /bin/bash
- **PATH** = /local/sbin:/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/X11R6/bin
- **LOAD_MODULES** = /local

#### GFDL
- **TERM** = xterm
- **HOME** = /home/Philip.Hollembeak
- **SHELL** = /bin/sh
- **USER** = Philip.Hollembeak
- **LOGNAME** = Philip.Hollembeak
- **PATH** = /local/sbin:/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/X11R6/bin

#### NCEP
- **AUTHSTATE** = files
- **MANPATH** = /opt/sr/sbin/rcst/man
- **TERM** = xterm
- **SHELL** = /usr/bin/ksh
- **SSH_CLIENT** = bastion
- **REMOTEHOST** = bastion
- **GROUP** = Zeusgrp18100
- **HOME** = /home/Philip.Hollembeak
- **USER** = Philip.Hollembeak
- **LOGNAME** = Philip.Hollembeak
- **PATH** = /local/sbin:/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/X11R6/bin
- **LOAD_MODULES** = /local

#### ESRL
- **LOGNAME** = daniel.schornak
- **HOME** = /home/daniel.schornak
- **PATH** = /usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
- **LOAD_MODULES** = /local

#### XCLARITY
- **TERM** = xterm
- **HOME** = /home/Daniel.Schornak
- **SHELL** = /bin/bash
- **USER** = Daniel.Schornak
- **LOGNAME** = Daniel.Schornak
- **PATH** = /local/sbin:/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/X11R6/bin
- **LOAD_MODULES** = /local

#### 3.3
- **TERM** = xterm
- **HOME** = /home/Daniel.Schornak
- **SHELL** = /bin/bash
- **USER** = Daniel.Schornak
- **LOGNAME** = Daniel.Schornak
- **PATH** = /local/sbin:/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/X11R6/bin
- **LOAD_MODULES** = /local

#### XOP
- **TERM** = xterm
- **HOME** = /home/Daniel.Schornak
- **SHELL** = /bin/bash
- **USER** = Daniel.Schornak
- **LOGNAME** = Daniel.Schornak
- **PATH** = /local/sbin:/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/X11R6/bin
- **LOAD_MODULES** = /local

#### 13269
- **TERM** = xterm
- **HOME** = /home/Daniel.Schornak
- **SHELL** = /bin/bash
- **USER** = Daniel.Schornak
- **LOGNAME** = Daniel.Schornak
- **PATH** = /local/sbin:/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/X11R6/bin
- **LOAD_MODULES** = /local

#### XOP
- **TERM** = xterm
- **HOME** = /home/Daniel.Schornak
- **SHELL** = /bin/bash
- **USER** = Daniel.Schornak
- **LOGNAME** = Daniel.Schornak
- **PATH** = /local/sbin:/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/X11R6/bin
- **LOAD_MODULES** = /local

#### TOTALVIEW
- **TERM** = xterm
- **HOME** = /home/Daniel.Schornak
- **SHELL** = /bin/bash
- **USER** = Daniel.Schornak
- **LOGNAME** = Daniel.Schornak
- **PATH** = /local/sbin:/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/X11R6/bin
- **LOAD_MODULES** = /local

#### XOP
- **TERM** = xterm
- **HOME** = /home/Daniel.Schornak
- **SHELL** = /bin/bash
- **USER** = Daniel.Schornak
- **LOGNAME** = Daniel.Schornak
- **PATH** = /local/sbin:/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/X11R6/bin
- **LOAD_MODULES** = /local
The Zeus login ("frontend") nodes provide access to the rest of the HPC cluster. They are intended for code and batch job management tasks, not for computation. Login nodes should ONLY be used for tasks similar to the following:

- Editing and compiling code
- Organizing data on project and home directories
- Copying data on and off the system
- Performing other operations that require external access
- Running very short (less than 1 minute) “glue” jobs.

Tasks that do not fit into the above categories should be run via the batch system. Processes that use a significant amount of CPU cycles or use a large amount of memory, particularly if they execute regularly, are better suited to be run on a compute node than a frontend. There are two ways the user can do this:

- Create a batch script and submit it as a compute or serial job
- Use a dedicated node or nodes in an interactive job session

For further details visit:

https://nesccdocs.rdhpcs.noaa.gov/wiki/index.php/Main_Page
# Proposed Zeus Frontend Limits

<table>
<thead>
<tr>
<th>Proposed Zeus Frontend Limits</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>memlock(soft)</td>
<td>4194304 (KB) [0.5GB]</td>
</tr>
<tr>
<td>memlock(hard)</td>
<td>8388608 (KB) [1GB]</td>
</tr>
<tr>
<td>nofile (soft/hard)</td>
<td>6400/64000</td>
</tr>
<tr>
<td>locks (soft/hard)</td>
<td>64/1024</td>
</tr>
<tr>
<td>nproc (soft/hard)</td>
<td>64/512</td>
</tr>
<tr>
<td>cpu (soft/hard):</td>
<td>100/2000</td>
</tr>
<tr>
<td>maxlogins (soft/hard):</td>
<td>20/20</td>
</tr>
</tbody>
</table>
Introduction to User Environment, Policies, Quotas, and Environmental Variables

In Practice – Home Filespace Usage

- Home directories shall have quota limits (Default quota size is 10GB).
- Home directories and subdirectories WILL BE backed up regularly through full and daily incremental backups.

NESCC Factoid…
“Quotas can be increased but only for users that require additional space for source code. Model data should not be stored on the home filesystem (for size and performance reasons.”
Introduction to User Environment, Policies, Quotas, and Environmental Variables

In Practice - High Performance Storage (HPS)

Zeus Facts
- Two (2) HPS filesystems (/scratch1 and /scratch2)
- Filesystem is Lustre
  - Lustre is the most widely used FS in high performance computing
  - Great for large block read/write (like all HPC filesystems)

• Quotas will be enforced on a per project basis
• There are no user quotas
• Scratch space IS NOT be backed up.
• The filesystem will not be purged
  - If projects require purging capability, please email help.nescc@noaa.gov for guidance
Introduction to User Environment, Policies, Quotas, and Environmental Variables

In Practice – Herc (TDS) High Performance Storage (HPS)

Herc Facts
- Not operated the same as Zeus
- Little warning given when the filesystems are reformatted or reconfigured
- Filesystem is Lustre
  - (/mnt/tds_lustre1 (/scratch1) and /mnt/tds_lustre2 (/scratch2))

- Project quotas are not enforced due to test and develop aspect of the system
- Scratch space IS NOT be backed up

NESCC Factoid…
“We will keep the Herc filesystems as stable as possible before Zeus goes live as this time is needed by you for code porting and bug fixing prior to production availability of Zeus.”
In Practice – Tape Storage Access

Tape Storage

The tape storage assets will not be directly mounted to the login or compute nodes. Access is provided through custom scripts that will permit the transfer of data to and from the tape storage server.

NESCC Factoid…
“Current methods to use the tape storage are hsi and htar. Other methods will be provided prior to Zeus going into production.”
Introduction to User Environment, Policies, Quotas, and Environmental Variables

Recommended Exercises

1. Log into TDS or Zeus
   a. Try using PuTTY (Windows Environment)
   b. Try using SSH (Linux Environment)

2. Get to know your session environmental variables ("printenv")

3. Review and Explore the User Environment

4. Set up an SSH Tunnel between your system and TDS or Zeus

5. Knowing the Limitations – Established Policies and Quotas
   a. Determine available files systems ("df")
   b. Locate posted policies and quotas
## Environment Variables

<table>
<thead>
<tr>
<th>Environment Variable</th>
<th>Example Output (echo &lt;Environment Variable&gt;)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$LD_LIBRARY_PATH</td>
<td>/opt/java/jdk1.6.0_04/lib:/opt/udunits/2.1.15/lib:/opt/papi/4.1.0/lib:/opt/ncl/5.2.0_bin/lib:/opt/hdf5/1.8.3-intel/lib:/opt/netcdf/3.6.3-intel/lib:/opt/intel/Compiler/11.1/072/stdlib/lib/intel64:/opt/intel/Compiler/11.1/072/mkl/lib/em64t:/opt/intel/Compiler/11.1/072/lib/intel64:/lib64/usr/lib64/usr/X11R6/lib64:/usr/local/lib64:/opt/sge/default/lib/1x26-amd64:/usr/local/lib64</td>
<td>Base path for resolving library paths for applications using Shared Objects and linkers</td>
</tr>
<tr>
<td>$LIBRARY_PATH</td>
<td>/lib64:/usr/lib64:/usr/X11R6/lib64:/usr/local/lib64</td>
<td>Provides search locations for resolving interactive and script command entries. (User modifiable)</td>
</tr>
<tr>
<td>$SHELL</td>
<td>/bin/tcsh</td>
<td>Current active UNIX shell for session (User modifiable) Available shells include: sh (bourne SHell); bash (Bourne-Again SHell); csh (C SHell); ksh (Korn SHell); tcsh (TC SHell); zsh (Z SHell)</td>
</tr>
<tr>
<td>$PWD</td>
<td>/home/Daniel.Schornak</td>
<td>Print Working Directory. Changing directory locations updates this environment variable</td>
</tr>
<tr>
<td>$LOGNAME</td>
<td>daniel.schornak</td>
<td>Name of the user</td>
</tr>
<tr>
<td>$MAIL</td>
<td>/var/spool/mail/daniel.schornak</td>
<td>Location of incoming mail</td>
</tr>
<tr>
<td>$USER</td>
<td>Daniel.Schornak</td>
<td>Name of the user</td>
</tr>
</tbody>
</table>
Alias command provides a simplified way of performing compound commands

Example:
At command line type: alias / history
Now display shell history by typing: /

Note:
Add the above entry to .cshrc and it will be available at shell instantiation
<table>
<thead>
<tr>
<th>Available Editors</th>
<th>Learn More</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ed (a.k.a. red)</td>
<td>ed --help</td>
<td>Line editor</td>
</tr>
<tr>
<td>vi</td>
<td>vi --help</td>
<td>Visual Interface (Note: vi is aliased to vim at NESCC)</td>
</tr>
<tr>
<td>emacs</td>
<td>emacs --help</td>
<td>Emacs is the extensible, customizable, self-documenting real-time display editor. This Info file describes how to edit with Emacs and some of how to customize it; it corresponds to GNU Emacs version 23.1.</td>
</tr>
<tr>
<td>rvi (a.k.a. vim; rview)</td>
<td>rvi --help</td>
<td>Vi IMproved, a programmer's editor</td>
</tr>
<tr>
<td>ex</td>
<td>ex --help</td>
<td>Line editor or VIM editor capability</td>
</tr>
</tbody>
</table>
## General Compiler Discussion

<table>
<thead>
<tr>
<th>Compilers</th>
<th>Intel * (module load intel)</th>
<th>Portland Group (module load pgi)</th>
<th>Gnu Compilers</th>
</tr>
</thead>
<tbody>
<tr>
<td>CC</td>
<td>icc</td>
<td>pgcc</td>
<td>gcc</td>
</tr>
<tr>
<td>F77</td>
<td>ifort</td>
<td>pgf77</td>
<td>gfortran</td>
</tr>
<tr>
<td>F90</td>
<td>ifort</td>
<td>pgf90</td>
<td>gfortran</td>
</tr>
<tr>
<td>C++</td>
<td>icpc</td>
<td>pgCC</td>
<td>g++</td>
</tr>
</tbody>
</table>

*Intel compilers are the primary compilers on Zeus

Note: To use PGI compilers, you must “module unload intel” first
Numerical Libraries Discussion

<table>
<thead>
<tr>
<th>Source</th>
<th>Package</th>
<th>Associated Libraries</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intel</td>
<td>Integrated Performance Primitives (IPP)</td>
<td>Intel® Integrated Performance Primitives (Intel® IPP) is an extensive library of multicore-ready, highly optimized software functions for multimedia, data processing, and communications applications. Intel IPP offers thousands of optimized functions covering frequently used fundamental algorithms</td>
</tr>
<tr>
<td>Intel</td>
<td>Math Kernel Library (MKL)</td>
<td>BLAS, LAPACK, PARDISO, FFT, FFTW, ScaLAPACK, Vector Math Library (VML), Vector Statistical Library (VSL), Summary Statistics Library (SSL)</td>
</tr>
<tr>
<td>PGI</td>
<td>PGI Math libraries</td>
<td>LAPACK, BLAS, FFT, Cluster FFT</td>
</tr>
<tr>
<td>SGI</td>
<td>Message Passing Toolkit (MPT)</td>
<td>Message Passing Toolkit (MPT) is a software package that supports interprocess data exchange for applications that use concurrent, cooperating processes on a single host or on multiple hosts. Data exchange is done through message passing, which is the use of library calls to request data delivery from one process to another or between groups of processes.</td>
</tr>
</tbody>
</table>

Many codes may use IBM’s ESSL and, in particular, FFT routines. IBM has chosen NOT to port ESSL to the x86_64 platform.
Module Commands

- `module help`
- `module avail`
- `module whatis`
- `module load`
- `module list`
- `module unload`
- `module switch`
- `module show`

NESCC Factoid…
“Module definitions will continue to be updated and changed as Zeus is prepared for production.”
Module Command - “module --help”

[daniel.schornak@fe2 ~]$ module --help

Modules Release 3.2.6 2007-02-14 <Copyright GNU GPL v2 1991>:

Usage: module [ switches ] [ subcommand ] [subcommand-args ]

Switches:

-H|--help       this usage info
-V|--version    modules version & configuration options
-f|--force      force active dependency resolution
-t|--terse      terse format avail and list format
-l|--long       long format avail and list format
-h|--human      readable format avail and list format
-v|--verbose    enable verbose messages
-s|--silent     disable verbose messages
-c|--create     create caches for avail and apropos
-i|--icase      case insensitive
-u|--userlvl <lvl> set user level to <nov[ice],exp[ert],adv[anced]>

Available SubCommands and Args:

+ add[load] modulefile [modulefile ...]
+ rm[unload] [modulefile1] modulefile2
+ switch[swap] modulefile [modulefile ...]
+ display[show] [modulefile [modulefile ...]]
+ avail [modulefile [modulefile ...]]
+ use [-a|--append] dir [dir ...]
+ unused dir [dir ...]
+ update
+ refresh
+ purge
+ list
+ clear
+ help [modulefile [modulefile ...]]
+ whatis [modulefile [modulefile ...]]
+ apropos [keyword] string
+ initadd modulefile [modulefile ...]
+ initprepend modulefile [modulefile ...]
+ initrm modulefile [modulefile ...]
+ initswitch modulefile1 modulefile2
+ initlist
+ initclear

Sample Output “module --help”
Module Commands – “module avail”

Sample Output “module avail”

Sample Output “module -l avail”
### Modules – Establishing the Programming Environment

**Sample modules at various NOAA HPC sites**

<table>
<thead>
<tr>
<th>NCEP Modules</th>
<th>ESRL Modules Standard Modules (module list)</th>
<th>ESRL Default 3.2.6 Modules (module avail)</th>
<th>GFDL Modules (module avail)</th>
<th>NESCC Modules (module avail)</th>
</tr>
</thead>
<tbody>
<tr>
<td>N/A</td>
<td>1) icc/11.1.072</td>
<td>1) R/2.0.9(default)</td>
<td>1) adaptive/6.1.0</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>2) mvapich2/1.4.1-intel-11.1</td>
<td>2) acroread/0.4.2-1(default)</td>
<td>2) adaptive/6.1.3(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>3) netcdf/3.6.3</td>
<td>3) acroread/0.4.2-1(default)</td>
<td>3) bffp/client/3.2.0</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>4) hdf4/4.2.27-intel</td>
<td>4) analysis_dujour/0.0.0(default)</td>
<td>4) bffp/client/3.2.0</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>5) ncl/5.2.0-intel</td>
<td>5) gempak/5.11.4</td>
<td>5) esmf/4.0.0p2(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>6) netcdf/3.6.3</td>
<td>6) gmpsc/1.3c</td>
<td>6) g2clib/1.2.2(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>7) ifort/11.1.073</td>
<td>7) hdf/4.2.6</td>
<td>7) g2clib/1.2.2(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>8) gempak/5.11.4</td>
<td>8) hdf4/4.2</td>
<td>8) g2clib/1.2.2(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>9) jet-tools</td>
<td>9) hdf5/1.8.3</td>
<td>9) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>10) udunits/2.1.15</td>
<td>10) hdf5/1.8.3</td>
<td>10) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>11) mathlibs/ftw</td>
<td>11) hdf5/1.8.7</td>
<td>11) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>12) mpi/1.2.1</td>
<td>12) hdf5/1.8.7</td>
<td>12) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>13) jet-tools</td>
<td>13) mpi/1.2.1</td>
<td>13) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>14) ifort/11.1.073</td>
<td>14) mpi/1.2.1</td>
<td>14) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>15) netcdf/3.6.3</td>
<td>15) mpi/1.2.1</td>
<td>15) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>16) intel/12.0.19</td>
<td>16) mpi/1.2.1</td>
<td>16) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>17) intel/12.0.19</td>
<td>17) mpi/1.2.1</td>
<td>17) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>18) intel/12.0.19</td>
<td>18) mpi/1.2.1</td>
<td>18) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>19) intel/12.0.19</td>
<td>19) mpi/1.2.1</td>
<td>19) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>20) intel/12.0.19</td>
<td>20) mpi/1.2.1</td>
<td>20) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>21) intel/12.0.19</td>
<td>21) mpi/1.2.1</td>
<td>21) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>22) intel/12.0.19</td>
<td>22) mpi/1.2.1</td>
<td>22) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>23) intel/12.0.19</td>
<td>23) mpi/1.2.1</td>
<td>23) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>24) intel/12.0.19</td>
<td>24) mpi/1.2.1</td>
<td>24) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>25) intel/12.0.19</td>
<td>25) mpi/1.2.1</td>
<td>25) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>26) intel/12.0.19</td>
<td>26) mpi/1.2.1</td>
<td>26) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>27) intel/12.0.19</td>
<td>27) mpi/1.2.1</td>
<td>27) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>28) intel/12.0.19</td>
<td>28) mpi/1.2.1</td>
<td>28) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>29) intel/12.0.19</td>
<td>29) mpi/1.2.1</td>
<td>29) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>30) intel/12.0.19</td>
<td>30) mpi/1.2.1</td>
<td>30) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>31) intel/12.0.19</td>
<td>31) mpi/1.2.1</td>
<td>31) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>32) intel/12.0.19</td>
<td>32) mpi/1.2.1</td>
<td>32) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>33) intel/12.0.19</td>
<td>33) mpi/1.2.1</td>
<td>33) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>34) intel/12.0.19</td>
<td>34) mpi/1.2.1</td>
<td>34) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>35) intel/12.0.19</td>
<td>35) mpi/1.2.1</td>
<td>35) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>36) intel/12.0.19</td>
<td>36) mpi/1.2.1</td>
<td>36) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>37) intel/12.0.19</td>
<td>37) mpi/1.2.1</td>
<td>37) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>38) intel/12.0.19</td>
<td>38) mpi/1.2.1</td>
<td>38) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>39) intel/12.0.19</td>
<td>39) mpi/1.2.1</td>
<td>39) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A</td>
<td>40) intel/12.0.19</td>
<td>40) mpi/1.2.1</td>
<td>40) g2clib/1.2.0(default)</td>
<td>N/A</td>
</tr>
</tbody>
</table>
Module Commands – "module whatis <module>"

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cit</td>
<td>loads the default CIT management environment</td>
</tr>
<tr>
<td>cuda/2.0</td>
<td>loads the CUDA development environment</td>
</tr>
<tr>
<td>cuda/2.1</td>
<td>loads the CUDA development environment</td>
</tr>
<tr>
<td>cuda/2.1beta</td>
<td>loads the CUDA development environment</td>
</tr>
<tr>
<td>cuda/2.2</td>
<td>loads the CUDA development environment</td>
</tr>
<tr>
<td>cuda/3.1</td>
<td>loads the CUDA development environment</td>
</tr>
<tr>
<td>cuda/3.2.16</td>
<td>loads the CUDA development environment</td>
</tr>
<tr>
<td>cuda/4.0.17</td>
<td>loads the CUDA development environment</td>
</tr>
<tr>
<td>dot</td>
<td>adds <code>.</code> to your PATH</td>
</tr>
<tr>
<td>gams/23.7.1</td>
<td>loads the path to GAMS</td>
</tr>
<tr>
<td>gempak/5.11.4</td>
<td>loads the Gempak environment</td>
</tr>
<tr>
<td>gempak/6.2.0</td>
<td>loads the Gempak environment</td>
</tr>
<tr>
<td>hdf4/4.2.5-intel</td>
<td>loads the HDF4 4.2.5-intel</td>
</tr>
<tr>
<td>hdf4/4.2.5-pgi</td>
<td>loads the HDF4 4.2.5-pgi</td>
</tr>
<tr>
<td>hdf5/1.8.3</td>
<td>loads the HDF5 library</td>
</tr>
<tr>
<td>hmmp/2.4.2</td>
<td>loads the HMPP Compiler</td>
</tr>
<tr>
<td>icc/11.1.038</td>
<td>loads the Intel C/C++ compiler</td>
</tr>
<tr>
<td>icc/11.1.072</td>
<td>loads the Intel C/C++ compiler</td>
</tr>
<tr>
<td>icc/11.1.073</td>
<td>loads the Intel C/C++ compiler</td>
</tr>
<tr>
<td>icc/2011.2.137</td>
<td>loads the Intel C/C++ compiler</td>
</tr>
<tr>
<td>ifort/11.1.038</td>
<td>loads the Intel C/C++ compiler</td>
</tr>
<tr>
<td>impi/4.0.0.017</td>
<td>loads the Intel MPI</td>
</tr>
<tr>
<td>intel/11.1.038</td>
<td>loads the Intel Compiler</td>
</tr>
<tr>
<td>intel/11.1.072</td>
<td>loads the Intel Compiler</td>
</tr>
<tr>
<td>intel/11.1.073</td>
<td>loads the Intel Compiler</td>
</tr>
<tr>
<td>intel/2011.2.137</td>
<td>loads the Intel Compiler</td>
</tr>
<tr>
<td>jet-tools</td>
<td>loads the jet tools</td>
</tr>
<tr>
<td>lahey/8.0.0a</td>
<td>loads the Lahey Compiler</td>
</tr>
<tr>
<td>lahey/8.1.0b</td>
<td>loads the Lahey Compiler</td>
</tr>
<tr>
<td>lf95/8.0.0a</td>
<td>loads the LF95 compiler</td>
</tr>
<tr>
<td>lf95/8.1.0b</td>
<td>loads the LF95 compiler</td>
</tr>
<tr>
<td>mathlibs/fftw</td>
<td>get most recent modules</td>
</tr>
<tr>
<td>module-cus</td>
<td>returns all various modules</td>
</tr>
<tr>
<td>module-info</td>
<td>loads the modules environment</td>
</tr>
<tr>
<td>modules</td>
<td>loads the modules environment</td>
</tr>
<tr>
<td>mpip/4.1.0</td>
<td>loads the PGI compiler environment</td>
</tr>
<tr>
<td>pgi/4.1.0</td>
<td>loads the PGI compiler environment</td>
</tr>
<tr>
<td>pgcc/10.8</td>
<td>loads the PGCC compiler</td>
</tr>
<tr>
<td>pgcc/10.9</td>
<td>loads the PGCC compiler</td>
</tr>
<tr>
<td>pgcc/11.7</td>
<td>loads the PGCC compiler</td>
</tr>
<tr>
<td>pgcc/9.0-2</td>
<td>loads the PGCC compiler</td>
</tr>
<tr>
<td>pg/10.6</td>
<td>loads the PGI compiler</td>
</tr>
<tr>
<td>pg/10.9</td>
<td>loads the PGI compiler</td>
</tr>
<tr>
<td>pg/11.1</td>
<td>loads the PGI compiler</td>
</tr>
<tr>
<td>pg/11.17</td>
<td>loads the PGI compiler</td>
</tr>
<tr>
<td>pg/12.0-2</td>
<td>loads the PGI compiler</td>
</tr>
<tr>
<td>sge</td>
<td>loads the SGE environment</td>
</tr>
<tr>
<td>sms/2.9.0-intel/9.1</td>
<td>loads the SMS environment for Intel 9.1/MUAPICH2-1.0</td>
</tr>
<tr>
<td>sms/2.9.0-pgi/7.1-3-muapich2-1.0</td>
<td>loads the SMS environment for Intel 9.1/MUAPICH2-1.0</td>
</tr>
<tr>
<td>torque</td>
<td>loads the Torque environment</td>
</tr>
<tr>
<td>torque callers/8.8.0-1</td>
<td>loads the TotalView debugging environment</td>
</tr>
<tr>
<td>ucinic/11.2.15</td>
<td>loads the Udins environment</td>
</tr>
<tr>
<td>ucinic/11.2.15</td>
<td>loads the Udins environment</td>
</tr>
<tr>
<td>use-nun</td>
<td>adds your own modules file to MODULEPATH</td>
</tr>
<tr>
<td>wkrh2/1.8.1.0a</td>
<td>loads the path to wkrh2</td>
</tr>
<tr>
<td>wkrh2/1.9.2c</td>
<td>loads the path to wkrh2</td>
</tr>
<tr>
<td>wkrh2/1.9.2c</td>
<td>loads the default wkrh2 environment</td>
</tr>
</tbody>
</table>

Sample Output – “module whatis <module>”
Module Commands

At Log in…

At Job Execution…

Related Environmental Variables:

MODULESHOME - The location of the master Modules package file directory containing module command initialization scripts, the executable program modulecmd, and a directory containing a collection of master modulefiles.

MODULEPATH - The path that the module command searches when looking for modulefiles.

LOADEDMODULES - A colon separated list of all loaded modulefiles.

_LMFILES_ - A colon separated list of the full pathname for all loaded modulefiles.

NESCC Factoid…

“At a minimum, you will want to:

• module load intel
• module load mpt

NESCC Factoid…

“Modules are **not** automatically loaded at user log in or job execution.”
Module Commands – “module load”

Sample Outputs from “printenv” ...Before and After “module load intel”
Module Commands – “module list”

Sample Output
“module list”

```
[daniel.schornak@fe2 ~]$ module list
Currently Loaded Modulefiles:
  1> icc/11.1.072
  2> mvapich2/1.4.1-intel-11.1
  3> netcdf/3.6.3
  4> hdf4/4.2r3-intel
  5> hdf5/1.8.3
  6> ncl/5.2.0-intel
  7> intel/11.1.072
  8> papi/4.1.0
  9> nco/4.0.1
10> udunits/2.1.15
11> mathlibs/fftw
12> uberftp/2.7
13> gempak/5.11.4
14> jet-tools
15> sge
16> totalview/8.8.0-1
17> wjet
```

Sample Output
“module -l list”

```
[daniel.schornak@fe2 ~]$ module -l list
- Package -----------------------------+ Versions --+ Last mod. -------
- Currently Loaded Modulefiles:
  icc/11.1.072
  mvapich2/1.4.1-intel-11.1
  netcdf/3.6.3
  hdf4/4.2r3-intel
  hdf5/1.8.3
  ncl/5.2.0-intel
  intel/11.1.072
  papi/4.1.0
  nco/4.0.1
  udunits/2.1.15
  mathlibs/fftw
  uberftp/2.7
  gempak/5.11.4
  jet-tools
  sge
  totalview/8.8.0-1
  wjet
[daniel.schornak@fe2 ~]$ [daniel.schornak@fe2 ~]$ 
```
Module Commands – “module switch <this> <that>”
“module swap <this> <that>”

Change to a newer C compiler version (current: intel/intel-cc-12-12.0.4.191; want: intel/intel-cc-12-12.1.0_sp1.6.233)

Before...

```
$ module list
Currently Loaded Modulefiles:
1) intel/intel-cc-12-12.0.4.191
2) intel/intel-icg-12-12.0.4.191
3) intel/intel-cmkp-12-12.0.4.191
4) intel/tools
```

Make the “Switch”...
```
module switch intel/intel-cc-12-12.0.4.191 intel/intel-cc-12-12.1.0_sp1.6.233
(same as: module swap intel/intel-cc-12-12.0.4.191 intel/intel-cc-12-12.1.0_sp1.6.233)
```

After...
```
$ module list
Currently Loaded Modulefiles:
1) intel/intel-cc-12-12.1.0_sp1.6.233
2) intel/intel-icg-12-12.0.4.191
3) intel/intel-cmkp-12-12.0.4.191
4) intel/tools
```
Module Commands – “module show”

Command shows what environmental variables are set and any associated modules.
Module Commands – What to watch out for!

Changing modules does not guarantee success!
Version incompatibilities among modules can occur

**Solution:**

1. Remove all installed modules and build from scratch (*module clear*)

   ```bash
   [Daniel.Schornak@tfe1 ~]$ module clear
   Are you sure you want to clear all loaded modules!? [n] y
   [Daniel.Schornak@tfe1 ~]$ module list
   No Modulefiles Currently Loaded.
   [Daniel.Schornak@tfe1 ~]$   
   ```

2. Load only the modules that you require *

   ```bash
   [Daniel.Schornak@tfe1 ~]$ module load intel
   [Daniel.Schornak@tfe1 ~]$ module list
   Currently Loaded Modulefiles:
   1) intelprod/cc/12-12.0.4.191  3) intelprod/cmk1/12-12.0.4.191
   2) intelprod/fc/12-12.0.4.191  4) intel/12-12.0.4.191
   [Daniel.Schornak@tfe1 ~]$   
   ```

   • Note: Interdependent/compatible modules will automatically load

3. Logging out and back in will provide a new session with default modules loaded
### Recommended Exercises

1. Module appreciation exercise…
   a. Unload all modules ("module clear")
   b. Verify Intel C compilers not available ("icc --v; if77 --v")
   c. Load and verify Intel compiler modules ("icc --v; if77 -v")
   d. Load and verify PGI compilers ("pgcc --version; pgf77 --version")

2. Modules for libraries
   a. Check library paths

3. Ensure modules are loaded at login
   a. Edit .profile file and add appropriate “module load”

4. Ensure modules are loaded at job execution
   a. Include “module load <modulename>” in batch script
### General Compiler Discussion

<table>
<thead>
<tr>
<th>Compilers</th>
<th>Intel (module load intel)</th>
<th>Portland Group (module load pgi)</th>
<th>Gnu Compilers</th>
</tr>
</thead>
<tbody>
<tr>
<td>CC</td>
<td>icc</td>
<td>pgcc</td>
<td>gcc</td>
</tr>
<tr>
<td>F77</td>
<td>ifort</td>
<td>pgf77</td>
<td>gfortran</td>
</tr>
<tr>
<td>F90</td>
<td>ifort</td>
<td>pgf90</td>
<td>gfortran</td>
</tr>
<tr>
<td>C++</td>
<td>icpc</td>
<td>pgCC</td>
<td>g++</td>
</tr>
</tbody>
</table>

Note: To use PGI compilers, you must “module unload intel” first.
**Introduction to Programming Environment**

**NESCC Software Stack**

<table>
<thead>
<tr>
<th>Operating System</th>
<th>bbcp</th>
<th>gcc</th>
<th>GrADS</th>
<th>libtool</th>
<th>netCDF v4</th>
<th>Python netcdf module</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch System</td>
<td>bbFTP</td>
<td>gdb</td>
<td>GRIB libraries and utilities</td>
<td>libxml</td>
<td>OpenMP</td>
<td>Python-numpy module</td>
</tr>
<tr>
<td>Batch System</td>
<td>Bison</td>
<td>Ghostview</td>
<td>gzip</td>
<td>Linux/UNIX debugger</td>
<td>Optimized BLAS</td>
<td>R (statistical package)</td>
</tr>
<tr>
<td>System Monitoring and Accounting</td>
<td>C/C++</td>
<td>Globus Toolkit</td>
<td>HDF4</td>
<td>McIDAS</td>
<td>Optimized FFTW</td>
<td>Ruby</td>
</tr>
<tr>
<td>ANTLR</td>
<td>Environment Modules</td>
<td>GNU binutils</td>
<td>HDF5</td>
<td>MP Toolbox, MPI,MP-21/0</td>
<td>Perl</td>
<td>SVN/CVS</td>
</tr>
<tr>
<td>autoconf</td>
<td>ESMF</td>
<td>GNU coreutils</td>
<td>Heirloom cpio</td>
<td>NCAR Graphics</td>
<td>PETSc</td>
<td>Udunits v1.X</td>
</tr>
<tr>
<td>automake</td>
<td>FORTRAN 77/90/95</td>
<td>GNU make</td>
<td>ImageMagick</td>
<td>Ncview</td>
<td>Python</td>
<td>UNPACK</td>
</tr>
<tr>
<td>bash</td>
<td>gawk</td>
<td>GNU tar</td>
<td>Java</td>
<td>netCDF Operators</td>
<td>Python scipy module</td>
<td>X-11</td>
</tr>
</tbody>
</table>
Introduction to Programming Environment

Preparing for Job Submission – Batch Script Considerations

Use msub. Here is an example batch script:

```sh
#!/bin/sh
#
#PBS -l procs=12
#PBS -l walltime=8:00:00
pwd
cd /home/ctierney/hpl/bin/Linux_em64t_mkl
module load intel
module load mpt
mpiexec_mpt -np 12 ./xhpl
```

In the example above, it uses "procs" to set the total number of cores.

This is the way all users should set the number of procs (or cores) needed.

NESCC Factoid…

“One Herc (TDS), you do not need to specify an account (-A) for running jobs. This will be a requirement on Zeus.”
Introduction to Programming Environment

Preparing for Job Submission – Batch Script Considerations (continued)

If you need to change the processor layout, there is a way. Use the nodes option.

#PBS –l nodes=4:ppn=6
Each node has 12 cores. The above syntax will allocate 4 nodes, and schedule 6 tasks per node. This is what you would want to do for an OpenMP job where each MPI process uses 2 threads.

Other examples:

#PBS –l nodes=12:ppn=1
Start job on 12 nodes, with one thread per node (For 12 MPI tasks, OMP_NUM_THREADS=12)

#PBS –l nodes=2:ppn=1+4:ppn=12
Start job with one process on each of the first two nodes, then fill up the last 4 nodes with 48 tasks - This would be for a job that has I/O tasks for the first two MPI ranks.
Optimizing performance for MPI applications using MPT

The default buffer settings used by the MPT MPI library are inadequate to provide decent performance for most applications. It is recommended that you set the following variables prior to calling mpiexec_mpt.

```
setenv MPI_BUFS_PER_PROC=128
setenv MPI_BUFS_PER_HOST=128
setenv MPI_GROUP_MAX=128
```

For more information: run “man mpi” after loading the mpt module.
Interactive Job performance for MPI applications using MPT

If you need to do other interactive work, including running, development, or debugging of jobs, an interactive job is a great way to do this. To create an interactive job, use the `-l` option to msub:

```bash
# msub -l nodes=1
```

![Interactive Job Performance](image.png)

- **10-Dec-2011**
  -> Prologue/Epilogue
  frees memory

- **07-Sep-2011**
  -> Reverted back to Prologue - Epilogue v 1.2
due to errors in script.

- **08-Sep-2011**
  -> Upgraded Moab to Moab 6.1.1 w/o ODBC

- **04-Sep-2011**
  -> New Prologue and Epilogue are running on the compute nodes.
  -> Updated MoAB to use NODCPO Features.

- **31-Aug-2011**
  -> Enabled dynamic overclocking (aka Intel TurboBoost)
  and set the scaling governor in Linux to “performance”
  -> Added kernel boot options recommended by SGI benchmarking

- **Before 31-Aug-2011**
  -> MoAB is now running 6.1
  -> Compiler licenses are restored
## Scheduler Commands Cross Reference Table

The table below reflects similar functions across the various schedulers within the NOAA HPC environments.

<table>
<thead>
<tr>
<th>NCEP LoadLeveler Commands</th>
<th>ESRL SGE Commands</th>
<th>NESCC (and GFDL) Moab Commands</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>llcancel</td>
<td>qdel</td>
<td>mjobctl -c</td>
<td>Cancels a submitted job</td>
</tr>
<tr>
<td>llclass</td>
<td>TBD</td>
<td>TBD</td>
<td>Returns information about job classes</td>
</tr>
<tr>
<td>llcc</td>
<td>TBD</td>
<td>TBD</td>
<td>Returns free cpus about job classes</td>
</tr>
<tr>
<td>llhold</td>
<td>qhold qrls</td>
<td>mjobctl -h mjobctl -u</td>
<td>Holds or releases a hold on a job</td>
</tr>
<tr>
<td>llq</td>
<td>sgestat qstat</td>
<td>checkjob showq</td>
<td>Queries the status of jobs</td>
</tr>
<tr>
<td>llstatus</td>
<td>qhost</td>
<td>showstate</td>
<td>Queries the status of machines</td>
</tr>
<tr>
<td>llsubmit</td>
<td>Qsub</td>
<td>msub</td>
<td>Submits a job</td>
</tr>
<tr>
<td>share_reporter.ksh</td>
<td>TBD</td>
<td>showstats</td>
<td>Reports the total share and used share for each group and user</td>
</tr>
<tr>
<td>TBD</td>
<td>userstat</td>
<td>TBD</td>
<td>Displays node statistics and batch queue for cluster</td>
</tr>
</tbody>
</table>
Running an OpenMP/MPI hybrid job

The previous instructions tell you how to request nodes and placement in a way to support OpenMP/MPI hybrid applications. However, it is necessary to include an additional command to get the processes of your job to be allocated correctly. To do this, use the `omplace` program. Example:

```
# mpiexec_mpt -np $NP omplace ./f.applic.x
```

By default, `omplace` will use the value set by `$OMP_NUM_THREADS`. If that value is not set, it will use 1. You can explicitly set the number of threads to be used by each MPI rank with the `-nt` option. Ex:

```
# mpiexec_mpt -np $NP omplace -nt 4 ./f.applic.x
```

In this example, each `f.applic.x` executable could use 4 threads. You still need to set `OMP_NUM_THREADS` to the correct value for your model run.
Online Documentation—Where to go...

Website
https://nesccdocs.rdhpcs.noaa.gov

Access and Use of TDS

PGI Compilers User Guide

Optimization with Intel Compilers

NESCC Factoid…
“Documentation, such as listed above, will start to be made available on NesscDocs soon.”
Need Help? – Where to go...

User Support and Problem Reporting – Help Desk for NESCC
help.nescc@noaa.gov

NESCC Documentation Wiki Site – “How to…”
https://nesccdocs.rdhpcs.noaa.gov/

End User Experience
EMC Model Transition Team (Provides updates for porting to TDS and Zeus)
http://www.emc.ncep.noaa.gov/mtt

NESCC Factoid…
“Additional documentation will be made available on NesccDocs soon.”
Group A End User Training

0800-0810 Training Process
  • Introduce instructors
  • General information
  • Define training roles and responsibilities
  • Review training agenda

0810-0815 Training Objectives
  • Review the objectives of the NESCC HPC Environment overview

0815-0845 Introduction to NESCC HPC Design/Architecture and Environment
  • Describe the overall architecture major components, functions, and features

0845-0900 Available On-line Documentation

0900-0915 BREAK

0915-0945 Introduction to User Environment, Policies, Quotas, and Environmental Variables

0945-1045 Introduction to Programming Environment
  • Introduce tools – shells, applications, compilers, and libraries

1045-1145 Introduction to NESCC Storage and Data Management

1145-1245 LUNCH BREAK

1245-1415 Introduction to Job Scheduling using Moab (Adaptive Computing)
  • Efficient job submission techniques
  • Installed NESCC version compared to other NOAA installations
  • Moab/Torque end-user commands and tools

1415-1500 Questions/Answers

1500 Session Wrap up
Shawn Hoopes – Adaptive Computing
<table>
<thead>
<tr>
<th>Time</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>0800-0810</td>
<td>Training Process</td>
</tr>
<tr>
<td></td>
<td>• Introduce instructors</td>
</tr>
<tr>
<td></td>
<td>• General information</td>
</tr>
<tr>
<td></td>
<td>• Define training roles and responsibilities</td>
</tr>
<tr>
<td></td>
<td>• Review training agenda</td>
</tr>
<tr>
<td>0810-0815</td>
<td>Training Objectives</td>
</tr>
<tr>
<td></td>
<td>• Review the objectives of the NESCC HPC Environment overview</td>
</tr>
<tr>
<td>0815-0845</td>
<td>Introduction to NESCC HPC Design/Architecture and Environment</td>
</tr>
<tr>
<td></td>
<td>• Describe the overall architecture major components, functions, and features</td>
</tr>
<tr>
<td>0845-0900</td>
<td>Available On-line Documentation</td>
</tr>
<tr>
<td>0900-0915</td>
<td><strong>BREAK</strong></td>
</tr>
<tr>
<td>0915-0945</td>
<td>Introduction to User Environment, Policies, Quotas, and Environmental Variables</td>
</tr>
<tr>
<td>0945-1045</td>
<td>Introduction to Programming Environment</td>
</tr>
<tr>
<td></td>
<td>• Introduce tools – shells, applications, compilers, and libraries</td>
</tr>
<tr>
<td>1045-1145</td>
<td>Introduction to NESCC Storage and Data Management</td>
</tr>
<tr>
<td>1145-1245</td>
<td><strong>LUNCH BREAK</strong></td>
</tr>
<tr>
<td>1245-1415</td>
<td>Introduction to Job Scheduling using Moab (Adaptive Computing)</td>
</tr>
<tr>
<td></td>
<td>• Efficient job submission techniques</td>
</tr>
<tr>
<td></td>
<td>• Installed NESCC version compared to other NOAA installations</td>
</tr>
<tr>
<td></td>
<td>• Moab/Torque end-user commands and tools</td>
</tr>
<tr>
<td>1415-1500</td>
<td>Questions/Answers</td>
</tr>
<tr>
<td>1500</td>
<td>Session Wrap up</td>
</tr>
</tbody>
</table>
Group A End User Training – Wrap Up

Reminder: Please complete and return the Post-training Questionnaire

Additional Questions?

Dan Schornak: dschorn@cscl.com
Need Help? – Where to go...

Help Desk for NESCC

help.nescc@noaa.gov

NESCO Documentation Website
https://nesccdocs.rdhpcs.noaa.gov/wiki/index.php/Main_Page

EMC Model Transition Team – Providing updates for porting to TDS and Zeus
http://www.emc.ncep.noaa.gov/mtt