



User Environment on LONI Clusters

Le Yan

Scientific computing consultant
User services group
Louisiana Optical Network Initiative





Outline

- Accounts and allocations
- Hardware overview
- Software overview
- Job management





Outline

- Accounts and allocations
- Hardware overview
- Software overview
- Job management





Account and Allocation Web Interface

- **LONI: allocations.loni.org**
- This is where you
 - Request accounts
 - Request and manage allocations
 - Update contact information
 - Reset password





Who Is Eligible for A LONI account

- Faculty or research staff members from one of the LONI's member institutions
- Students working with a faculty or research staff member from one of the LONI's member institutions
- Researchers who collaborate with a faculty or research staff member from one of the LONI's member institutions
- Faculty members from other Louisiana universities may apply for an account as well
 - Needs approval from the LONI executive director





Account Application Procedure

- Go to the allocation web site and click on “request login”
- Type in your email address and the image code
 - The email needs to be an institutional one
 - The likes of hotmail and gmail will not be accepted
- You will receive an email with a link to the real form
- Fill out the form and submit
 - “LONI Contact/Collaborator”: this is your adviser if you are a student, or yourself if you are a faculty member
- Your account will be activated after your information is confirmed
 - May take up to a week





Account Policy

- No account sharing is allowed
 - An account is for one user only
 - Every person who will use the cluster needs to apply for their own account
- Your account is subject to deletion once your affiliation with LONI member institutions terminates
- Penalties
 - Account suspension
 - Loss of allocation





Password Reset Procedure

- Go to the allocation website and click on “forgot your password?”
- Type in the image code and your email address
 - The email address must be the one you used to apply for the account
- You will receive an email with a link to the reset form
- Type the new password and submit
- A system administrator will approve your password reset
 - May take a short while





Changing Login Shell

- “Profile” menu item
- Supported shells
 - bash
 - tcsh
 - csh
 - ksh
 - sh
- “chsh” and “ypchsh” do not work

LONI
Louisiana Optical Network Initiative

Prospective Researchers | Education | Corporate Visitors

Profile for UID 'lyan1'

Title: []

First Name: [Le]

Last Name: [Yan]

Email: [lyan1@lsu.edu]

LONI Contact/Collaborator: [Unknown UID] [Honggao Liu]

Alternate Email #1: []

Alternate Email #2: []

Alternate Email #3: []

Alternate Email #4: []

Office Phone: [(225) 578-7524]

Mobile Phone: []

Home Phone: [+1-225-336-1726]

Fax Number: []

AOL Instant Messenger ID: []

Office Room Number: []

Office Building Name: []

Office Postal Address: [Frey Computing Center, Baton Rouge, LA 70803 9US]

Department: [Frey Comp Center]

Research Area: []

Personal Web Page: []

Login Shell: [/bin/bash]

Globus DN: []

Old Passwd: []

New Passwd: []

Re-type New Passwd: []

Curriculum Vitae PDF: [] [Browse...]

[Update]

Logged in as lyan1

- Balances
- Activate Teragrid Users
- Activate Users
- Begin Review
- View All
- Request Allocation
- Manage Memberships
- Manage Donations
- Manage Groups
- User Admin
- Profile**
- My Allocations

About
Logout





Allocation

- An allocation is some finite number of service unit (SUs) that allow you to run jobs on a cluster
 - One SU is one cpu-hour
 - Example
 - 40 SUs will be charged for a job that runs 10 hours on 4 cores
- Enforced on all LONI clusters





Types of Allocations

- Startup
 - Less than or equal to 50k SUs
 - Applications reviewed by local allocation committee member
 - Decision will be made within a few weeks after submission
 - Good for one year
- Large
 - Greater than 50k SUs
 - Applications reviewed by LONI allocation committee during the quarterly meeting
 - Decision will be made on January 1, April 1, July 1 and October 1 of each year
 - Good for one year





Requesting A New Allocation

- The principal investigator must be a faculty or research staff member from one of the LONI member institutes
- Procedure
 - Click on “request allocation”
 - Click on “new allocation”
 - Fill out the form and submit
 - You must submit a proposal along with a large allocation request
 - Need to specify how many SUs are needed on each platform
 - IBM AIX clusters
 - Dell Linux clusters





Joining An Existing Allocation

- Any user can join an existing allocation
- Procedure
 - Click on “request allocation”
 - Click on “join allocation”
 - Enter the name, email address or username of the allocation PI to search for the allocation you want to join
 - Click on the "Join Projects" button
 - The allocation PI will receive an email regarding to your request and you can use the allocation after the PI approves the request





Manage An Allocation

- “Manage membership” menu item
 - Add a user to allocation
 - Remove a member from allocation
 - Make a current member allocation administrator
- “Manage donation” menu item
 - Donate the remaining time on any allocation you administer to any other allocation that you are currently able to use





When An Allocation Expires

- Allocations are NOT extensible
- When submitting a new allocation request
 - The content can be copied from the previous requests
 - “My allocations” -> “Clone/edit”
 - The committee is likely to ask for a brief report, especially if it is not the first request for the project
 - “My allocations” -> “Report”
 - If you use up a couple of startup allocations in a short period of time, be prepared to apply for a large one
 - Plan ahead as large allocations are reviewed quarterly





Checking Allocation Balance

- Use the allocation web interface
 - The “Balances” menu item lists the balances of all allocations of which you are currently a member
- Use the “balance” command on a cluster

```
[lyan1@painter2 packages]$ balance
===== Allocation information for lyan1 =====
  Proj. Name|          Alloc|  Balance| Deposited|      %Used| Days Left|      End
-----|-----|-----|-----|-----|-----|-----
loni_loniadmin1|loni_loniadmin1| 37320.35| 100000.00|    62.68|      21|2009-10-01
loni_train09|loni_train09 on @Dell_Cluster| 39658.21| 40000.00|    0.85|     113|
2010-01-01
```

Note: Balance and Deposit are measured in CPU-hours





Outline

- Accounts and allocations
- Hardware overview
- Software overview
- Job management





Architectures of LONI Clusters

- Two architectures
 - Linux clusters
 - Vendor: Dell
 - Operating System: Linux (Red hat)
 - Processor: Intel
 - AIX clusters
 - Vendor: IBM
 - Operating System: AIX
 - Processor: IBM





Current deployment status - Dell Linux clusters

	Name	Peak TeraFLOPS/s	Location	Status	Login
LONI	Queen Bee	50.7	ISB	Available	LONI
	Eric	4.7	LSU	Available	LONI
	Oliver	4.7	ULL	Available	LONI
	Louie	4.7	Tulane	Available	LONI
	Poseidon	4.7	UNO	Available	LONI
	Painter	4.7	LaTech	Available	LONI





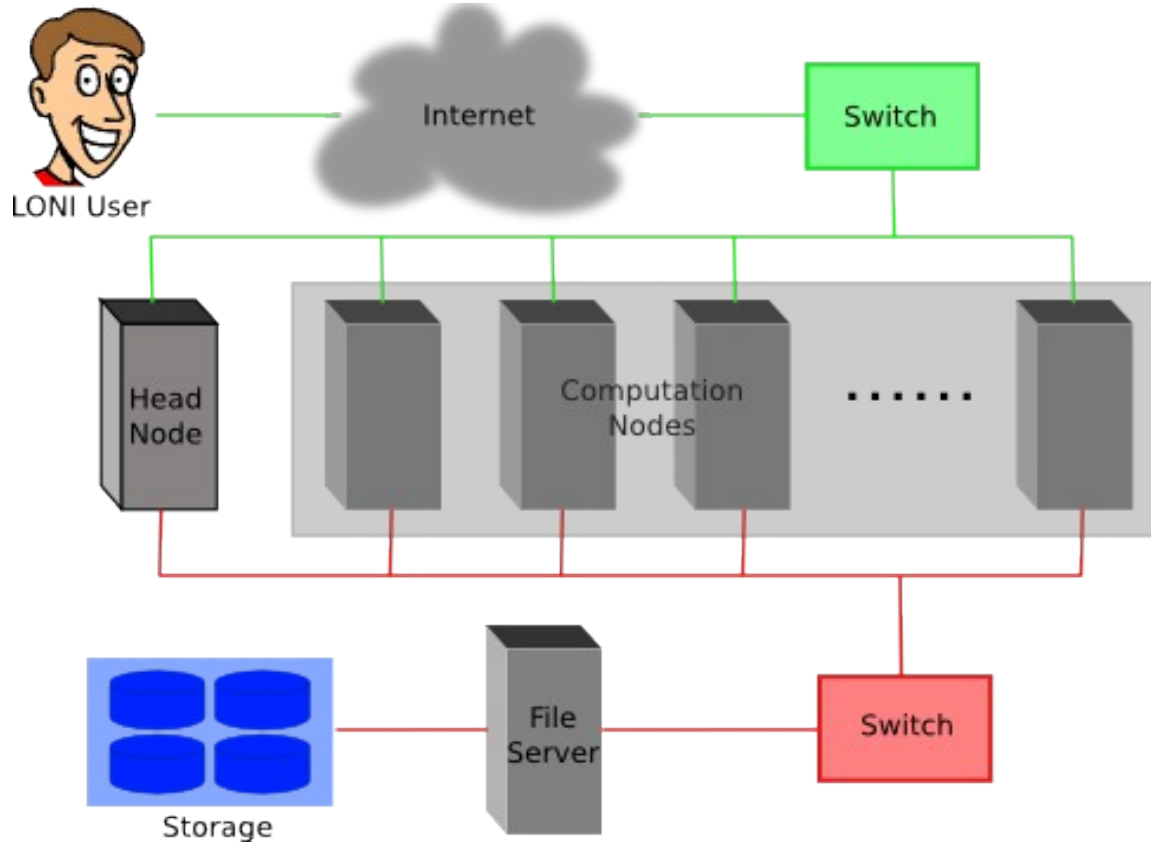
Current deployment status - IBM AIX clusters

	Name	Peak TeraFLOPS/s	Location	Status	Login
LONI	Bluedawg	0.85	LaTech	Available	LONI
	Ducky	0.85	Tulane	Available	LONI
	Zeke	0.85	ULL	Available	LONI
	Neptune	0.85	UNO	Available	LONI
	Lacumba	0.85	Southern	Available	LONI





Cluster Architecture



- A cluster is a group of computers (nodes) that works together closely
- Type of nodes
 - Head node
 - Compute node





Hardware Specification

- Queen Bee
 - 668 nodes
 - **8** Intel Xeon cores @ 2.33 GHz, **8** GB RAM, 36 GB HD
 - 192 TB storage
- Other LONI Linux clusters
 - 128 nodes
 - **4** Intel “Woodcrest” Xeons cores @ 2.33 Ghz, **4** GB RAM, 80 GB HD
 - 9 TB storage
- LONI AIX clusters
 - 14 power5 nodes with each node having: **8** IBM Power5 processors @ 1.9 GHz, **16** GB RAM
 - 280 GB storage





How Much Memory Your Program Can Use

- The amount of installed memory less the amount that is used by the operating system and other utilities
- Max amount per node
 - Linux clusters: ~**6** GB for Queen Bee, ~**3** GB for others
 - AIX clusters: ~**13** GB





Choose The Correct Architecture/Platform

- There are numerous different architectures in the HPC world
- You need to choose the correct one when installing or using software
 - Linux clusters
 - EM64T, AMD64, X86_64
 - AIX clusters:
 - PowerPC, Power5

Software Downloads

Download NAMD:

NAMD is a parallel, object-oriented molecular dynamics code designed for high-p complete information and documentation.

Selecting an archive below will lead to a user registration and login page. Your dc

Version Nightly Build (2009-06-29) Platforms:

- [Linux-x86_64](#) (Opteron, Athlon64, Intel EMT64)
- [Source Code](#)

Version 2.7b1 (2009-03-23) Platforms:

- [AIX-POWER](#) (formerly AIX-RS6000)
- [AIX-POWER-MPI](#) (also p690, formerly IBM-SP)
- [BlueGeneP](#) (Blue Gene/P)
- [Linux-x86](#)
- [Linux-x86-TCP](#) (TCP may be better on gigabit)
- [Linux-x86_64](#) (Opteron, Athlon64, Intel EMT64)
- [Linux-x86_64-TCP](#) (TCP may be better on gigabit)
- [Linux-Itanium-Altix](#) (SGI Altix)
- [MacOSX-x86](#) (Mac OS X for Intel processors)
- [MacOSX-PPC](#) (Mac OS X, 2.6b1 and newer need **IBM libraries**)
- [Solaris-Sparc](#)
- [Solaris-x86_64](#)
- [Win32](#) (Windows XP, etc.)
- [Source Code](#)





File Systems

	Distributed file system	Throughput	File life time	Best used for
Home	Yes	Low	Unlimited	Code in development, compiled executables
Work	Yes	High	30 days	Job input/output
Local Scratch	No		Job duration	Temporary files needed by running jobs

- Tips

- Never let you job write output to your home directory
- Do not write temporary files to /tmp
 - Write to the local scratch or work space
- The work space is not for long-term storage
 - Files are purged periodically
- Use “rmpurge” to delete large amount of files





Disk Quota

Cluster	Home		Work		Local scratch
	Access point	Quota	Access point	Quota	Access point
LONI Linux	/home/\$USER	5 GB	/work/\$USER	100 GB	/var/scratch
LONI AIX	/home/\$USER	500 MB	/work/default/\$USER	20 GB	/scratch/local

- No quota is enforced on the work space on Queen Bee
- On Linux clusters, the work directory is created within an hour after the first login
- Check current disk usage
 - Linux: showquota
 - AIX: quota





Accessing LONI Clusters

- Host name: <cluster name>.loni.org
 - Queen Bee: qb.loni.org
- Use ssh to connect
 - *nix and Mac: type “ssh <host name>” in a terminal
 - Windows: use Putty
- Only accessible via Internet 2 at the moment
- The default Login shell is bash
 - Supported shells: bash, tcsh, ksh, csh & sh
 - Change the login shell at the profile page
 - Log in at allocations.loni.org and click on “profile”





Exercise 1: Now it's time to log in

- Log in any cluster
- Check your disk quota
 - Linux clusters: use “showq uota” command
 - Your scratch directory will be created within an hour of the first login
 - AIX clusters: use “q uota” command
- Locate the directory `/home/lyan1/traininglab/environment`
 - There are files that you will need for following exercises





Outline

- Accounts and allocations
- Hardware overview
- **Software overview**
- Job management





Software Available on LONI Clusters

- Installed Software
 - Compilers
 - Mathematical and utility libraries
 - FFTW, HDF5, NetCDF, PETSc...
 - Applications
 - Amber, CPMD, NWChem, NAMD, Gromacs, R, LAMMPS...
 - Programming Tools
 - Totalview, TAU...
- List of software
 - Linux clusters: https://docs.loni.org/wiki/Linux_Software
 - AIX clusters: https://docs.loni.org/wiki/AIX_Software
- Installed under `/usr/local/packages`





Using SOFTENV

- Environment variables
 - PATH: where to look for executables
 - LD_LIBRARY_PATH: where to look for shared libraries
 - Other custom environment variables needed by various software
- **SOFTENV**
 - Is a software that helps users set up environment variables properly to use other software packages
 - More convenient than setting environment variables in .bashrc or .cshrc





Listing All Packages

- Command “`softenv`” lists all packages that are managed by SOFTENV

```
[lyan1@tezpur2 ~]$ softenv
```

```
...
```

These are the macros available:

```
* @default
* @globus-4.0          globus client
* @intel-compilers    compiler: 'Intel Compilers', version: Latest.
                      A pointer to the latest installed intel
                      compilers.
```

These are the keywords explicitly available:

```
+Mesa-6.4.2           No description yet for Mesa-6.4.2.
+R-2.8.0-gcc-3.4.6    application: 'R', version 2.8.0
+ansys-lsdyna-11.0    application: 'ANSYS LS-DYNA', version: 11.0
                      ANSYS LS-DYNA is a premier software package
                      for explicit nonlinear structural
                      simulation with finite element pre- and
                      post-processor. docs =>
                      http://www1.ansys.com/customer/
```

Softenv key

```
...
```





Searching A Specific Package

- Use “-k” option with “softenv”

```
[lyan1@tezpur2 ~]$ softenv -k fftw
SoftEnv version 1.6.4
```

...

```
Search Regexp: fftw
```

These are the macros available:

These are the keywords explicitly available:

+fftw-3.1.2-gnu	application: FFTW, version 3.1.2, binded with GNU compiler.
+fftw-3.1.2-intel10.1	application: FFTW, version 3.1.2, binded with Intel compiler v10.1.
+fftw-3.1.2-intel9.1	application: FFTW, version 3.1.2, binded with Intel compiler v9.1.

...





Setting up Environment via Softenv — permanent change

- Set up the environment variables to use a certain software package
 - First add the key to `$HOME/.soft`
 - Then execute `resoft` at the command line
 - The environment will be the same next time you log in

```
[lyan1@tezpur2 ~]$ cat .soft
#
# This is the .soft file.
...
+matlab-r2007b
@default
[lyan1@tezpur2 ~]$ resoft
```





Setting up Environment via Softenv – one time change

- Set up the environment variables to use a certain software package **in the current login session only**
 - Add a package: `soft add <key>`
 - Remove a package: `soft delete <key>`

```
[lyan1@tezpur2 ~]$ which gcc
/usr/local/compilers/GNU/gcc-4.2.0/bin/gcc
[lyan1@tezpur2 ~]$ soft add +gcc-4.3.0
[lyan1@tezpur2 ~]$ which gcc
/usr/local/compilers/GNU/gcc-4.3.0/bin/gcc
[lyan1@tezpur2 ~]$ soft delete +gcc-4.3.0
[lyan1@tezpur2 ~]$ which gcc
/usr/local/compilers/GNU/gcc-4.2.0/bin/gcc
```





Querying a Softenv key

- Command “`soft-dbq`” shows which variables are set by a SOFTENV key

```
[lyan1@tezpur2 ~]$ soft-dbq +gcc-4.3.0
This is all the information associated with
the key or macro +gcc-4.3.0.
```

```
-----
Name: +gcc-4.3.0
Description: GNU gcc compiler, version 4.3.0
Flags: none
Groups: none
Exists on: Linux
-----
```

On the Linux architecture,
the following will be done to the environment:

The following environment changes will be made:

```
LD_LIBRARY_PATH = ${LD_LIBRARY_PATH}:/usr/local/compilers/GNU/gcc-4.3.0/lib64
PATH = ${PATH}:/usr/local/compilers/GNU/gcc-4.3.0/bin
-----
```





Exercise 2: Use Softenv

- Find the key for VISIT (a visualization package)
- Check what variables are set through the key
- Set up your environment to use VISIT
- Check if the variables are correctly set by “`which visit`”





Exercise 2: Use Softenv

- Find the key for VISIT (a visualization package)
 - Use `softenv -k visit`
- Check what variables are set through the key
 - Use `soft-dbq +visit`
- Set up your environment to use VISIT
 - Use `soft add +visit`
 - Or add “+visit” to your `.soft` file and `resoft`
- Check if the variables are correctly set by “`which visit`”
 - The output should be the path to the executable `visit`





Compilers

Language	Linux clusters			AIX clusters
	Intel	GNU	PGI	XL compilers
Fortran	ifort	g77	pgf77,pgf95	xlf,xlf_r,xlf90,xlf90_r
C	icc	gcc	pgcc	xlc,xlc_r
C++	icpc	g++	pgCC	xlC,xlC_r

- Usage: `<compiler> <options> <your_code>`
 - Example: `icc -O3 -o myexec mycode.c`
- Some compilers options are **architecture** specific
 - Linux: EM64T, AMD64 or X86_64
 - AIX: power5 or powerpc





MPI Compilers (1)

Language	Linux clusters	AIX clusters
Fortran	mpif77,mpif90	mpxlf,mpxlf_r,mpxlf90,mpxlf90_r
C	mpicc	mpcc,mpcc_r
C++	mpiCC	mpCC,mpCC_r

- Usage: similar to what we have seen
 - Example: `mpif90 -O2 -o myexec mycode.f90`
- On Linux clusters
 - Only one compiler for each language
 - There is no `intel_mpicc` or `pg_mpicc`





MPI Compilers (2)

- These MPI compilers are actually **wrappers**
 - They still use the compilers we've seen on the previous slide
 - Intel, PGI or GNU
 - They take care of everything we need to build MPI codes
 - Head files, libraries etc.
 - What they actually do can be reveal by the `-show` option

```
[lyan1@tezpur2 ~]$ mpicc -show
icc -DUSE_STDARG -DHAVE_STDLIB_H=1 -DHAVE_STRING_H=1 -DHAVE_UNISTD_H=1
-DHAVE_STDARG_H=1 -DUSE_STDARG=1 -DMALLOC_RET_VOID=1
-L/usr/local/packages/mvapich-1.0-intel10.1/lib -lmpich
-L/usr/local/ofed/lib64 -Wl,-rpath=/usr/local/ofed/lib64 -libverbs
-libumad -lpthread -lpthread -lrt
```





MPI Compilers (3)

```
[lyan1@qb2 ~]$ ls -ld /usr/local/packages/mvapich*
drwxr-xr-x 12 root root 4096 Oct 18 13:25 /usr/local/packages/mvapich-0.98-gcc
drwxr-xr-x 12 root root 4096 Jan 23 11:35 /usr/local/packages/mvapich-0.98-intel10.1
drwxr-xr-x 12 root root 4096 Oct 18 13:25 /usr/local/packages/mvapich-0.98-intel9.1
drwxr-xr-x 12 root root 4096 Oct 18 13:25 /usr/local/packages/mvapich-0.98-intel9.1-LM
drwxr-xr-x 12 root root 4096 Feb 12 10:27 /usr/local/packages/mvapich-0.98-pgi6.1
drwxr-xr-x 12 root root 4096 Oct 18 13:25 /usr/local/packages/mvapich-0.98-pgi6.1-eric
...
drwxr-xr-x 10 root root 4096 Oct 18 13:25 /usr/local/packages/mvapich2-0.98-intel9.1
drwxr-xr-x 11 root root 4096 Nov 9 16:31 /usr/local/packages/mvapich2-1.01-intel10.0
drwxr-xr-x 9 root root 4096 Jan 25 09:54 /usr/local/packages/mvapich2-1.0.1-intel10.1
drwxr-xr-x 11 root root 4096 Nov 8 13:10 /usr/local/packages/mvapich2-1.01-intel9.1
```

- There are many different versions of MPI compilers on Linux clusters
- Each of them is built around a specific compiler
 - Intel, PGI or GNU
- **It is extremely important to compile and run you code with the same version!!!**





Exercise 3: Compile a program

- Serial code
 - Copy hello.f90
from `/home/lyan1/traininglab/environment`
 - Compile it with a compiler of your choice
 - Run the executable from the command line
- MPI code
 - Copy hello_mpi.f90
from `/home/lyan1/traininglab/environement`
 - Compile it with a serial compiler and see what happens
 - Compile it with an MPI compiler





Exercise 3: Compile a code

- Linux

```
cp /home/lyan1/traininglab/environment/*.f90 .
icc -o hello_ser hello.f90
./hello_ser
mpif90 -o hello hello_mpi.f90
```

- Mac

AIX

```
cp /home/lyan1/traininglab/environment/*.f90 .
xlf90_r -o hello_ser hello.f90
./hello_ser
mpxlf90_r -o hello hello_mpi.f90
```





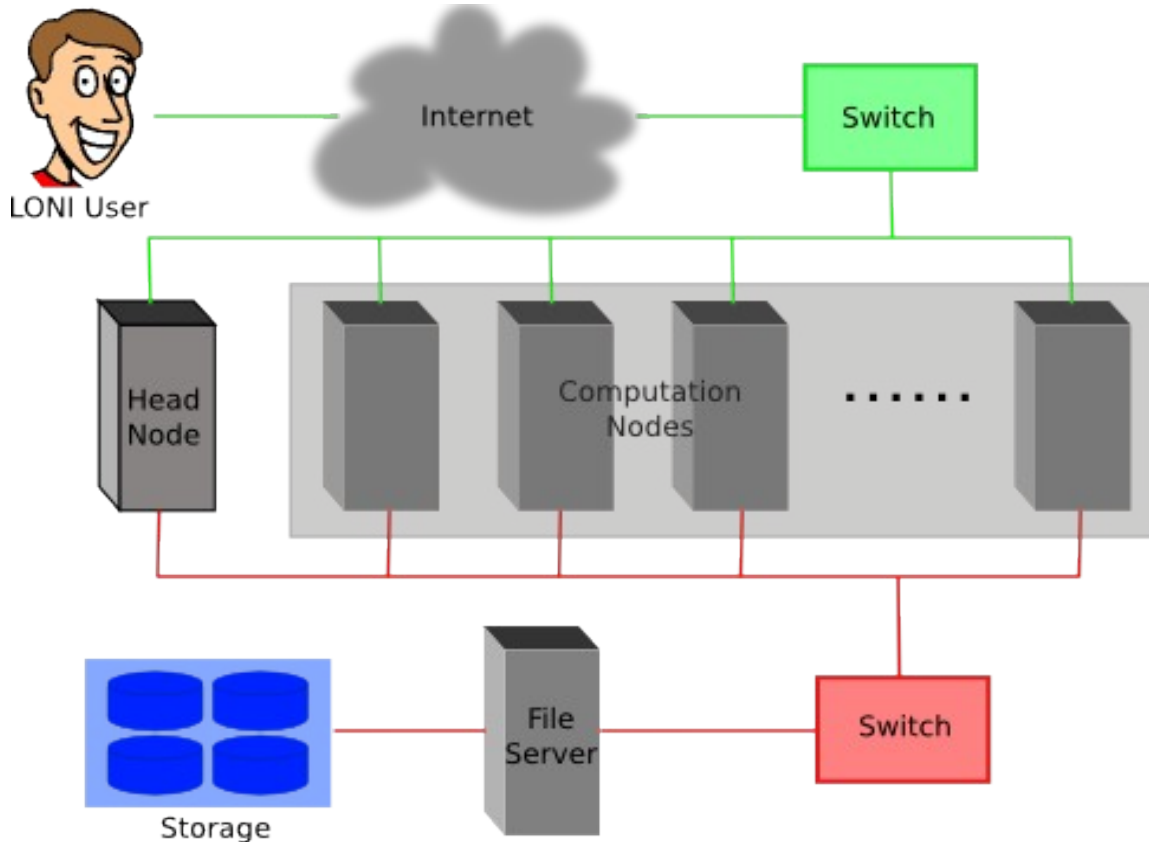
Outline

- Accounts and allocations
- Hardware overview
- Software overview
- Job management





Cluster Environment



- Multiple compute nodes
- Multiple users
- Each user may have multiple jobs running simultaneously





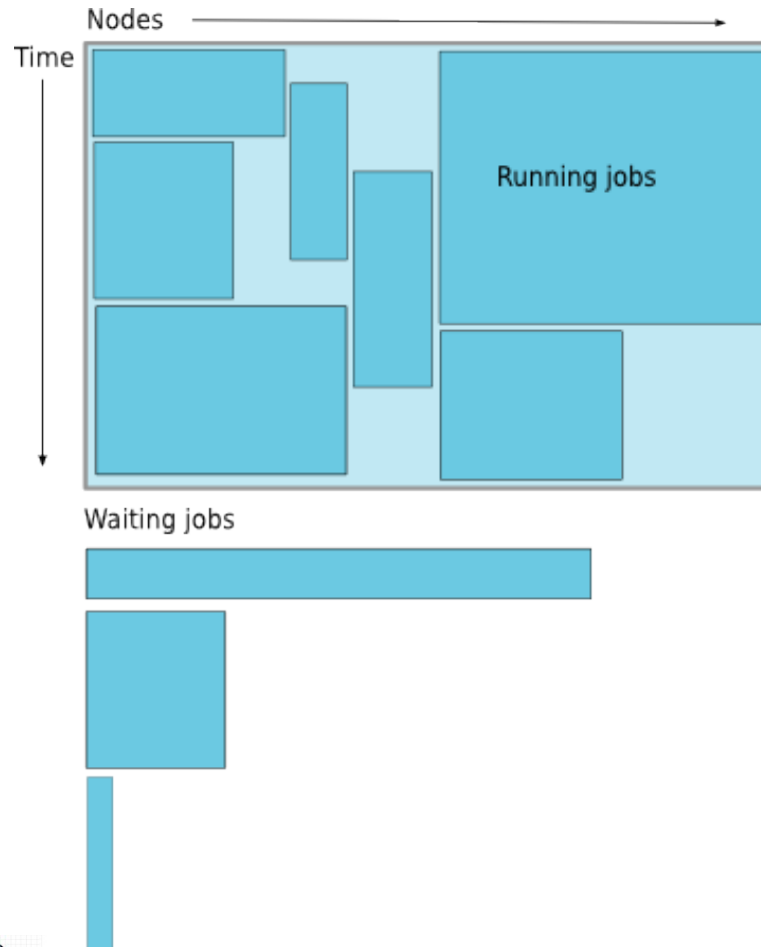
Batch Queuing System

- A software that manages resources (CPU time, memory etc.) and schedules job execution
 - Linux clusters: Portable Batch System(PBS)
 - AIX clusters: Loadleveler
- What is a job
 - A user's request to use a certain amount of resources for a certain amount of time
- The batch queuing system determines
 - The order jobs are executed
 - On which node(s) jobs are executed





A Simplified View of Job Scheduling

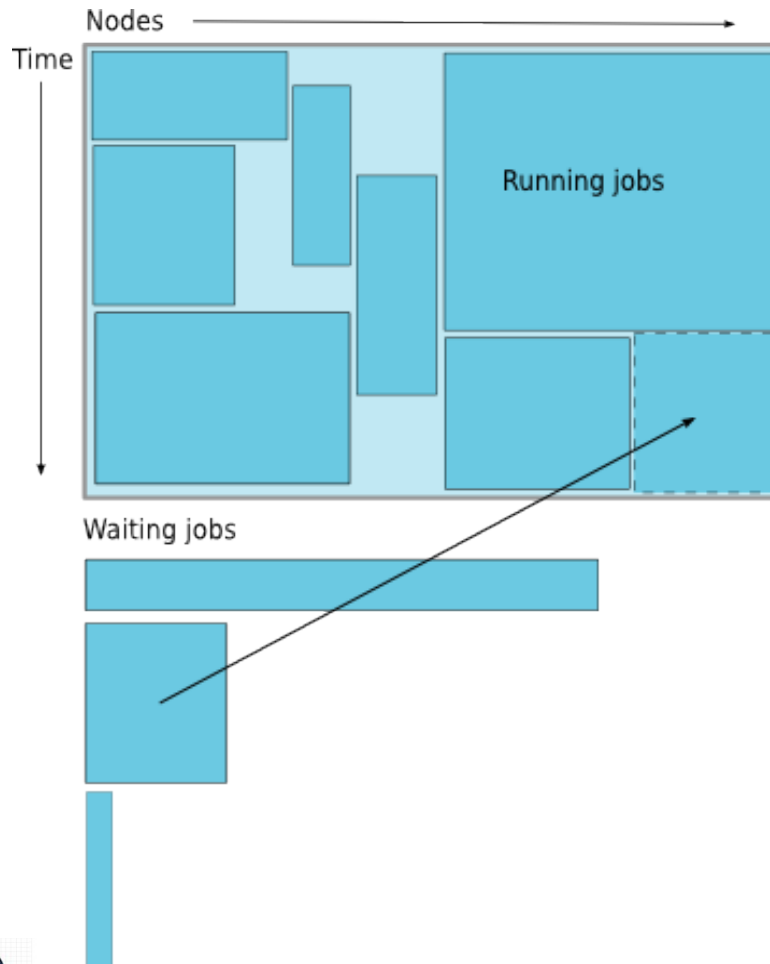


- Map jobs onto the node-time space
 - Assuming CPU time is the only resource
- Need to
 - Honor the order in which jobs are received
 - Maximize resource utilization





Backfilling

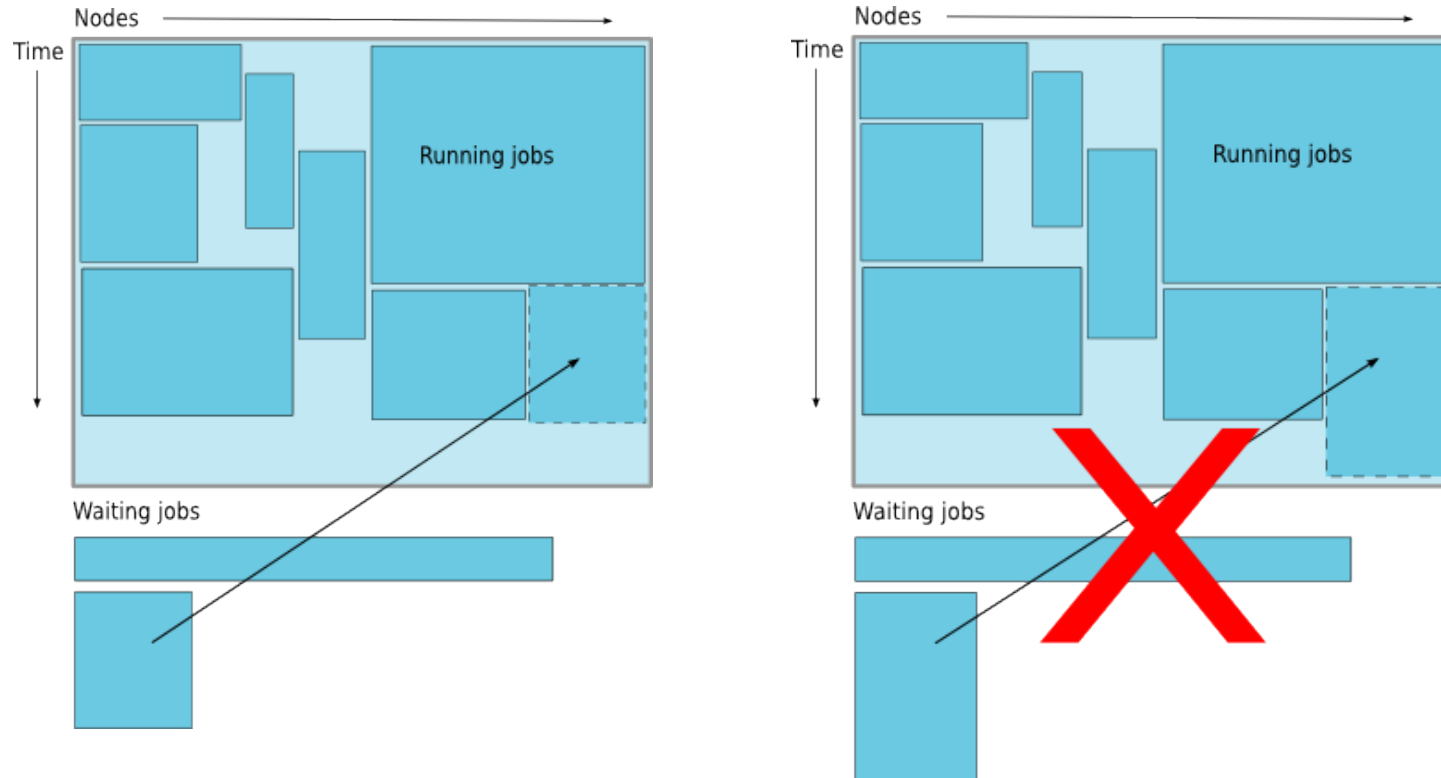


- A strategy to improve utilization
 - Allow a job to jump ahead of others when there are enough idle nodes
 - Must not affect the estimated start time of the job with the highest priority
- Enabled on all LONI clusters





How Much Time Should I Ask for?



- Ask for an amount of time that is
 - Long enough for your job to complete
 - As short as possible to increase the chance of backfilling





Job Queues

- There are more than one job queue
- Each job queue differs in
 - Number of available nodes
 - Max run time
 - Max running jobs per user
 - ...
- The main purpose is to maximize utilization





Queue Characteristics – Queen Bee

Queue	Max Runtime	Total number of available nodes	Max running jobs per user	Max nodes per job	Use
Workq	2 days	530	8	128	Unpreemptable (default)
Checkpt		668		256	Preemptable jobs
Preempt		668	NA		Require permission
Priority		668	NA		Require permission





Queue Characteristics – Other LONI Linux Clusters

Queue	Max Runtime	Total number of available nodes	Max running jobs per user	Max nodes per job	Use
Single	14 days	16	64	1	Single processor jobs
Workq	3 days	96	8	40	Unpreemptable (default)
Checkpt		128		64	Preemptable jobs
Preempt		64	NA	Require permission	
Priority		64	NA	Require permission	





Queue Characteristics – LONI AIX Clusters

Queue	Max Runtime	Total number of available nodes	Max running jobs per user	Max nodes per job	Use
Single	14 days	1	8	1	Single processor jobs
Workq	5 days	8		8	Unpreemptable (default)
Checkpt		14		14	Preemptable jobs
Preempt		6	NA	Require permission	
Priority		6	NA	Require permission	





Basic Commands

- Queue querying
 - Check how busy the cluster is
- Job submission
 - Submit a job to run
- Job monitoring
 - Check job status (estimated start time, remaining run time etc.)
- Job manipulation
 - Cancel/hold jobs





Queue Querying – Linux Clusters

- Command: `qfree`
 - Show the number of free, busy and queued nodes
- Command: `qfreeloni`
 - Equivalent to run `qfree` on all LONI Linux clusters

```
[lyan1@louie2 ~]$ qfree
PBS total nodes: 128, free: 81, busy: 44, down: 3, use: 34%
PBS checkpt nodes: 128, free: 81, busy: 28
PBS workq nodes: 32, free: 16, busy: 16
```





Queue Querying – AIX Clusters

- Command: `llclass`

lyan1@l2f1n03\$ llclass

Name	MaxJobCPU d+hh:mm:ss	MaxProcCPU d+hh:mm:ss	Free Slots	Max Slots	Description
interactive	undefined	undefined	8	8	Interactive Parallel jobs running on interactive node
single	unlimited	unlimited	4	8	One node queue (14 days) for serial and up to 8-processor parallel jobs
workq	unlimited	unlimited	51	56	Default queue (5 days), up to 56 processors
priority	unlimited	unlimited	40	40	priority queue reserved for on-demand jobs (5 days), up to 48 processors
preempt	unlimited	unlimited	40	40	preemption queue reserved for on-demand jobs (5 days), up to 48 processors
checkpoint	unlimited	unlimited	91	96	queue for checkpointing jobs (5 days), up to 104 processors, Job running on this queue can be preempted for on-demand job
















Checking Loads on All LONI Clusters

- Check Loads on all LONI clusters at docs.loni.org
- Updated every 15 minutes

Dell Linux Clusters

System Name	Nodes	SMP Size	Total CPUs	Memory/Node	TFLOPS	Work Disk	Location	Load	Running jobs	Queued jobs
Queen Bee	680	8	5440	8 GB	50.7	58 TB	LSU		0	422
Eric	128	4	512	4 GB	4.772	9 TB	LSU		70	111
Oliver	128	4	512	4 GB	4.772	9 TB	ULL		16	13
Louie	128	4	512	4 GB	4.772	9 TB	Tulane		27	56
Poseidon	128	4	512	4 GB	4.772	9 TB	UNO		17	3
Painter	128	4	512	4 GB	4.772	9 TB	LaTech		23	28

IBM P5 Clusters

System Name	Nodes	SMP Size	Total CPUs	Memory/Node	TFLOPS	Work Disk	Location	Load	Running jobs	Queued jobs
Bluedawg	14	8	104	16 GB	0.851	270 GB	LaTech		16	3
Ducky	14	8	104	16 GB	0.851	270 GB	Tulane		7	0
Zeke	14	8	104	16 GB	0.851	270 GB	ULL		1	0
Neptune	14	8	104	16 GB	0.851	270 GB	UNO		9	8
LaCumba	14	8	104	16 GB	0.851	270 GB	SU		8	8





Job Types

- Interactive job
 - Set up an interactive environment on compute nodes for users
 - Advantage: can run programs interactively
 - Disadvantage: must be present when the job starts
 - Purpose: testing and debugging
 - Disabled on AIX clusters because of limited number of nodes
- Batch job
 - Executed without user intervention using a job script
 - Advantage: the system takes care of everything
 - Disadvantage: can only execute one sequence of commands which cannot be changed after submission
 - Purpose: production run





Submitting Jobs – Linux Clusters

- Interactive job
 - `qsub -I -V -l walltime=<hh:mm:ss>,nodes=<# of nodes>:ppn=4 -A <your allocation> -q <queue name>`
- Batch job
 - `qsub <job script>`
- `ppn` must be either 4 (all Linux clusters except Queen Bee) or 8 (Queen Bee) except for serial jobs





PBS Job Script – Parallel Jobs

<code>#!/bin/bash</code>	
<code>#PBS -l nodes=4:ppn=4</code>	Number of nodes and processor
<code>#PBS -l walltime=24:00:00</code>	Maximum wall time
<code>#PBS -N myjob</code>	Job name
<code>#PBS -o <file name></code>	File name for standard output
<code>#PBS -e <file name></code>	File name for standard error
<code>#PBS -q checkpt</code>	Queue name
<code>#PBS -A <loni_allocation></code>	Allocation name
<code>#PBS -m e</code>	Send mail when job ends
<code>#PBS -M <email address></code>	Send mail to this address
<code><shell commands></code>	
<code>mpirun -machinefile \$PBS_NODEFILE -np 16 <path_to_executable> <options></code>	
<code><shell commands></code>	





PBS Job Script – Serial Jobs

#!/bin/bash

#PBS -l **nodes=1:ppn=1**

#PBS -l walltime=24:00:00

#PBS -N myjob

#PBS -o <file name>

#PBS -e <file name>

#PBS -q **single**

#PBS -A <loni_allocation>

#PBS -m e

#PBS -M <email address>

Number of nodes and processor

Maximum wall time

Job name

File name for standard output

File name for standard error

The only queue that accepts serial jobs

Allocation name

Send mail when job ends

Send mail to this address

<shell commands>

<path_to_executable> <options>

<shell commands>





Submitting Batch Jobs - AIX Clusters

- Batch job

- llsubmit <job
script>

```
#!/bin/sh
#@ job_type = parallel
#@ output = /work/default/username/${jobid}.out
#@ error = /work/default/username/${jobid}.err
#@ notify_user = youremail@domain
#@ notification = error
#@ class = checkpt
#@ wall_clock_limit = 24:00:00
#@ node_usage = shared
#@ node = 2,2
#@ total_tasks = 16
#@ initialdir = /work/default/username
#@ environment = COPY_ALL
#@ queue

<shell commands>

poe <path_to_executable> <options>
<shell commands>
```





Loadleveler Job Script – Serial Jobs

```
#!/bin/sh
#@ job_type = serial
#@ output = /work/default/username/${jobid}.out
#@ error = /work/default/username/${jobid}.err
#@ notify_user = youremail@domain
#@ notification = error
#@ class = checkpt
#@ wall_clock_limit = 24:00:00
#@ initialdir = /work/default/username
#@ environment = COPY_ALL
#@ queue
```

```
<shell commands>
<path_to_executable> <options>
<shell commands>
```





Job Monitoring – Linux Clusters

- **Command:** `showstart <job_id>`
 - Check when a job is estimated to start
- Things that can change the estimated start time
 - Higher priority job gets submitted
 - Other jobs terminate earlier than the system expects
 - The system has trouble starting your job





Job Monitoring – Linux Clusters cont'd

- **Command:** `qstat <options> <job_id>`
 - Show information on job status
 - All jobs are displayed if `<job_id>` is omitted
 - Show jobs submitted by a specific user: `qstat -u <username>`
 - Display in the alternative format: `qstat -a <job_id>`
- **Command:** `qshow <job_id>`
 - Show information on a running job
 - On which node(s) the job is running
 - CPU load





Job Monitoring – AIX Clusters

- **Command:** `llq <options> <job_id>`
 - All jobs are displayed if `<job_id>` is omitted
 - Display detailed information: `llq -l <job_id>`
 - Check the estimated start time: `llq -s <job_id>`
 - Show jobs from a specific user: `llq -u <username>`

```
lyan1@l2f1n03$ llq
```

Id	Owner	Submitted	ST	PRI	Class	Running On
12f1n03.3697.0	collin	1/22 16:59	R	50	single	12f1n14
12f1n03.3730.0	jheiko	1/28 13:30	R	50	workq	12f1n10
12f1n03.3726.0	collin	1/26 08:21	R	50	single	12f1n14
12f1n03.3698.0	collin	1/22 17:00	R	50	single	12f1n14
12f1n03.3727.0	collin	1/26 08:21	R	50	single	12f1n14

5 job step(s) in queue, 0 waiting, 0 pending, 5 running, 0 held, 0 preempted





Job Manipulation – Linux Clusters

- Command: `qdel <job_id>`
 - Cancel a running or queued job
 - May take some time depending on the size of the job
- Command: `qhold <job_id>`
 - Put a queued job on hold
- Command: `qrls <job_id>`
 - Resume a held job





Job Manipulation – AIX Clusters

- **Command:** `llcancel <job_id>`
 - Cancel a running or queued job
- **Command:** `llhold <job_id>`
 - Put a queued job on hold
- **Command:** `llhold -r <job_id>`
 - Resume a held job





Exercise 4

- Compile the parallel program `hello_mpi.f90`
 - Located under `/home/lyan1/traininglab/environment`
 - To compile
 - Linux clusters: `mpif90 -o <name of executable> hello_mpi.f90`
 - AIX clusters: `mpxlf90 -o <name of executable> hello_mpi.f90`
- Run it within an interactive job session
 - Submit an interactive job
 - Run on the command line
 - Linux clusters: `mpirun -np <# of cpus> <name of executable>`





Exercise 5

- Run the same program as a batch job
 - Sample submission scripts can be found under the same directory
 - Linux clusters: `submit.aix`
 - AIX clusters: `submit.linux`

