Saguaro User Environment

Gil Speyer speyer@asu.edu January 5, 2012

Outline

- Linux Clusters
- Initial login
- Modules
- Batch System
- Job Monitoring
- System & Memory Information
- Interactive mode
- Improving performance

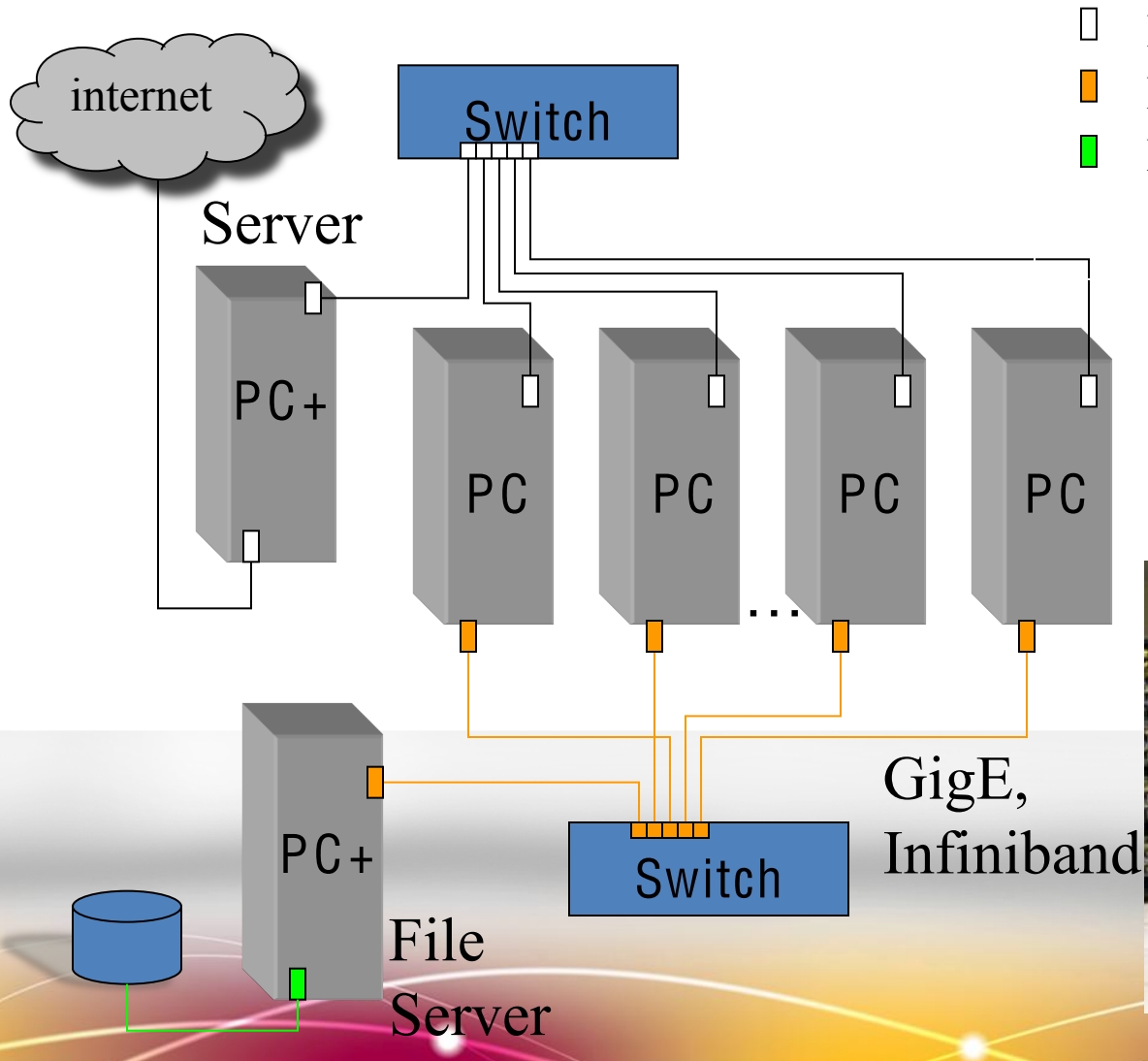
Anti-outline

- Shell scripting
- Available software
- Installing software
- Compiling
- Debugging (parallel or otherwise)

For any assistance with these – please contact us at any time:

`support@hpchelp.asu.edu`

Generic Cluster Architecture



(Adv. HPC System)

Initial Login

- Login with SSH

  ```
  ssh saguaro.fulton.asu.edu
  ```

- Connects you to a login node
- Don’t overwrite ~/.ssh/authorized_keys
  - Feel free to add to it if you know how to use it
  - SSH is used for job start up on the compute nodes. Mistakes can prevent your jobs from running
- For X forwarding, use `ssh -X`

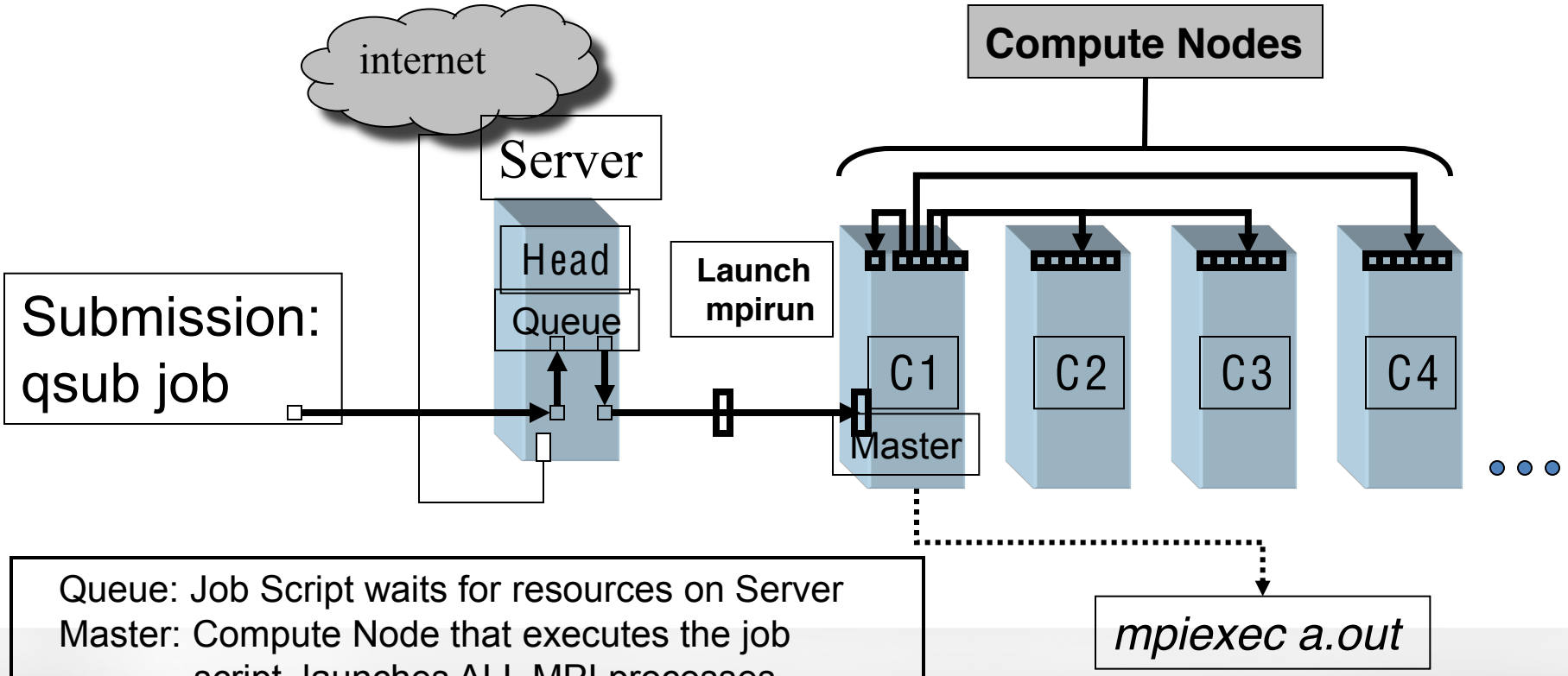
Packages (saguaro)

- Modules are used to setup your PATH and other environment variables
- They are used to setup environments for packages & compilers

```
saguaro% module                          {lists options}
saguaro% module avail                    {lists available packages}
saguaro% module load <package> <…>       {add one or more packages}
saguaro% module unload <package>         {unload a package}
saguaro% module list      {lists loaded packages}
saguaro% module purge     {unloads all packages}
```

- Multiple compiler families and MPI implementations available, so make sure you are consistent between libraries, source and run script!

Batch Submission Process



Queue: Job Script waits for resources on Server
Master: Compute Node that executes the job script, launches ALL MPI processes

Launch: contact each compute node to start executable (e.g. a.out)

Torque and Moab and Gold

- Torque – The resource manager (`qstat`)
- Moab – The scheduler (`checkjob`)
- Gold – The accounting system (`mybalance -h`)

Images.fanpop.com

Batch Systems

- *Saguaro* systems use Torque for batch queuing and Moab for scheduling
- Batch jobs are submitted on the front end and are subsequently executed on compute nodes as resources become available
- Order of job execution depends on a variety of parameters:
  - Submission Time
  - Backfill Opportunities: small jobs may be back-filled while waiting for bigger jobs to complete
  - Fairshare Priority: users who have recently used a lot of compute resources will have a lower priority than those who are submitting new jobs
  - Advanced Reservations: jobs my be blocked in order to accommodate advanced reservations (for example, during maintenance windows)
  - Number of Actively Scheduled Jobs: there are limits on the maximum number of concurrent processors used by each user

Commonly Used TORQUE Commands

qsub	Submit a job
qstat	Check on the status of jobs
qdel	Delete running and queued jobs
qhold	Suspend jobs
qrls	Resume jobs
qalter	Change job information

man pages for all of these commands

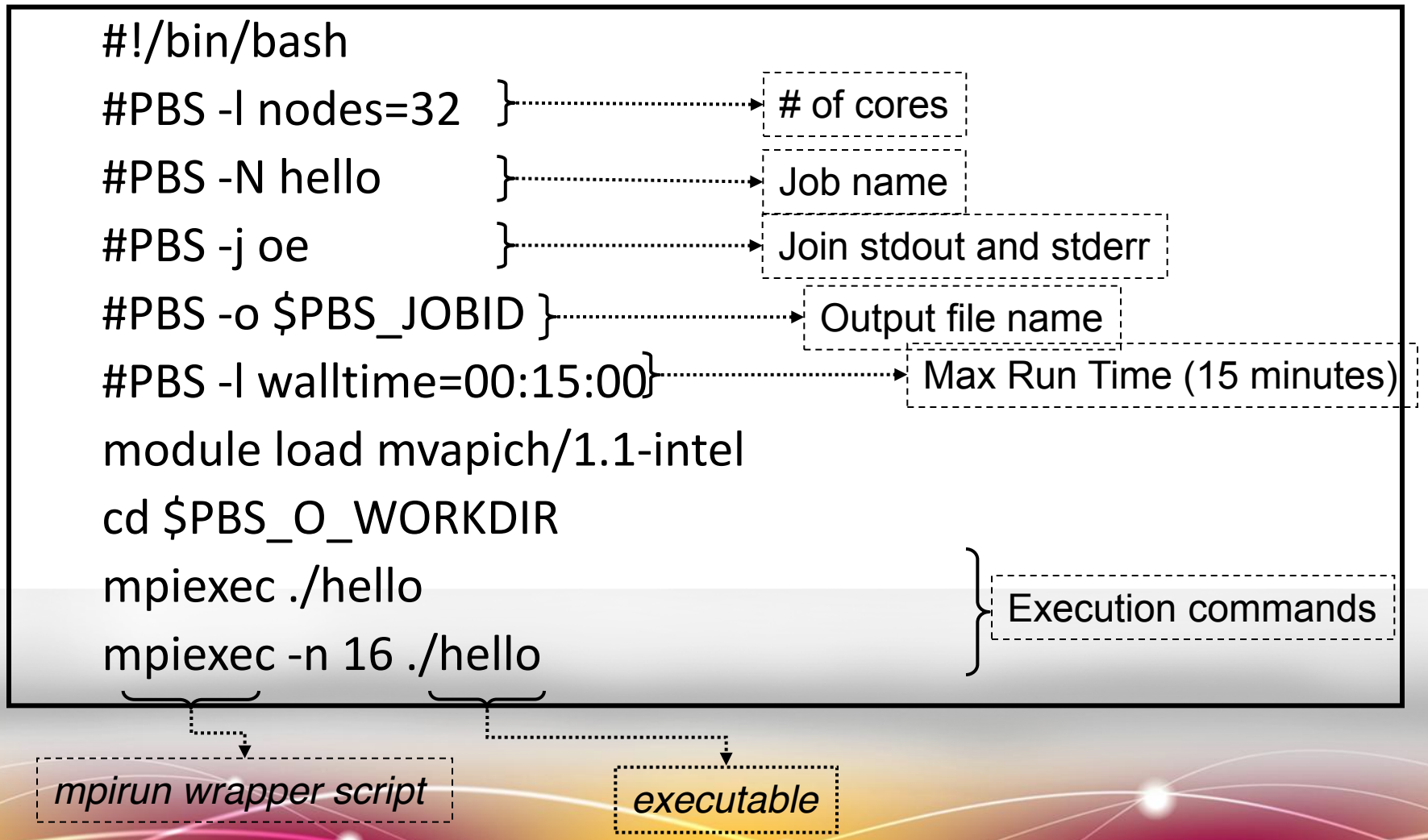
TORQUE Batch System

Variable	Purpose
PBS_JOBID	Batch job id
PBS_JOBNAME	User-assigned (-J) name of the job
PBS_TASKNUM	Number of slots/processes for a parallel job
PBS_QUEUE	Name of the queue the job is running in
PBS_NODEFILE	Filename containing list of nodes
PBS_O_WORKDIR	**Job working directory**

Batch System Concerns

- Submission (need to know)
  - Required Resources
  - Run-time Environment
  - Directory of Submission
  - Directory of Execution
  - Files for stdout/stderr Return
  - Email Notification
- Job Monitoring
- Job Deletion
  - Queued Jobs
  - Running Jobs

TORQUE: MPI Job Script



Batch Script Debugging Suggestions

- Echo issuing commands
  - (“set -x” or “set echo” for bash or csh).
- Avoid absolute pathnames
  - Use relative path names or environment variables ($HOME, $PBS_O_WORKDIR)
- Abort job when a critical command fails.
- Print environment
  - Include the "env" command if your batch job doesn't execute the same as in an interactive execution.
- Use “./” prefix for executing commands in the current directory
  - The dot means to look for commands in the present working directory. Not all systems include "." in your $PATH variable. (usage: ./a.out).
- Track your CPU time

Job Monitoring (*showq* utility)

```
saguaro% showq
ACTIVE JOBS--------------------
JOBID     JOBNAME    USERNAME      STATE   PROC   REMAINING        STARTTIME

11318  1024_90_96x6    vmcalo    Running     64    18:09:19  Fri Jan  9 10:43:53
11352           naf   phaa406    Running     16    17:51:15  Fri Jan  9 10:25:49
11357           24N   phaa406    Running     16    18:19:12  Fri Jan  9 10:53:46
 23 Active jobs     504 of 556 Processors Active (90.65%)

IDLE JOBS----------------------
JOBID     JOBNAME    USERNAME      STATE   PROC    WCLIMIT        QUEUETIME

11169        poroe8      xgai       Idle   128    10:00:00  Thu Jan  8 10:17:06
11645   meshconv019    bbarth       Idle    16    24:00:00  Fri Jan  9 16:24:18
 3 Idle jobs

BLOCKED JOBS-------------------
JOBID     JOBNAME    USERNAME      STATE   PROC    WCLIMIT        QUEUETIME

11319  1024_90_96x6    vmcalo   Deferred     64    24:00:00  Thu Jan  8 18:09:11
11320  1024_90_96x6    vmcalo   Deferred     64    24:00:00  Thu Jan  8 18:09:11
 17 Blocked jobs

Total Jobs: 43    Active Jobs: 23    Idle Jobs: 3     Blocked Jobs: 17
```

Job Monitoring (*qstat* command)

```
saguaro$ qstat -s a
job-ID  prior   name       user         state submit/start at     queue                 slots
-----------------------------------------------------------------------------------------------
  16414 0.12347 NAMD       xxxxxxxx     r     01/09/2008 15:13:58 normal@i101-302...    512
  15907 0.13287 tf7M.8     xxxxxxxx     r     01/09/2008 13:36:20 normal@i105-410...    512
  15906 0.13288 f7aM.7     xxxxxxxx     r     01/09/2008 13:33:47 normal@i171-401...    512
  16293 0.06248 ch.r32     xxxxxxxx     r     01/09/2008 14:56:58 normal@i175-309...    256
  16407 0.12352 NAMD       xxxxxxxx     qw    01/09/2008 12:23:21                       512
  16171 0.00000 f7aM.8     xxxxxxxx     hqw   01/09/2008 10:03:43                       512
  16192 0.00000 tf7M.9     xxxxxxxx     hqw   01/09/2008 10:06:17                       512
```

Basic *qstat* options:

-u username Display jobs belonging to specified user (* or ‘*’ for all)

-r/-f Display extended job information

TORQUE Job Manipulation/Monitoring

- To kill a running or queued job (takes ~30 seconds to complete):
  `qdel <jobID>`
  `qdel -p <jobID>` (Use when qdel alone won't delete the job)
- To suspend a queued job:
  `qhold <jobId>`
- To resume a suspended job:
  `qrls <jobID>`
- To alter job information in queue:
  `qalter <jobID>`
- To see more information on why a job is pending:
  `checkjob -v <jobID> (Moab)`
- To see a historical summary of a job:
  `qstat -f <jobID> (TORQUE)`
- To see available resources:
  `showbf (Moab)`

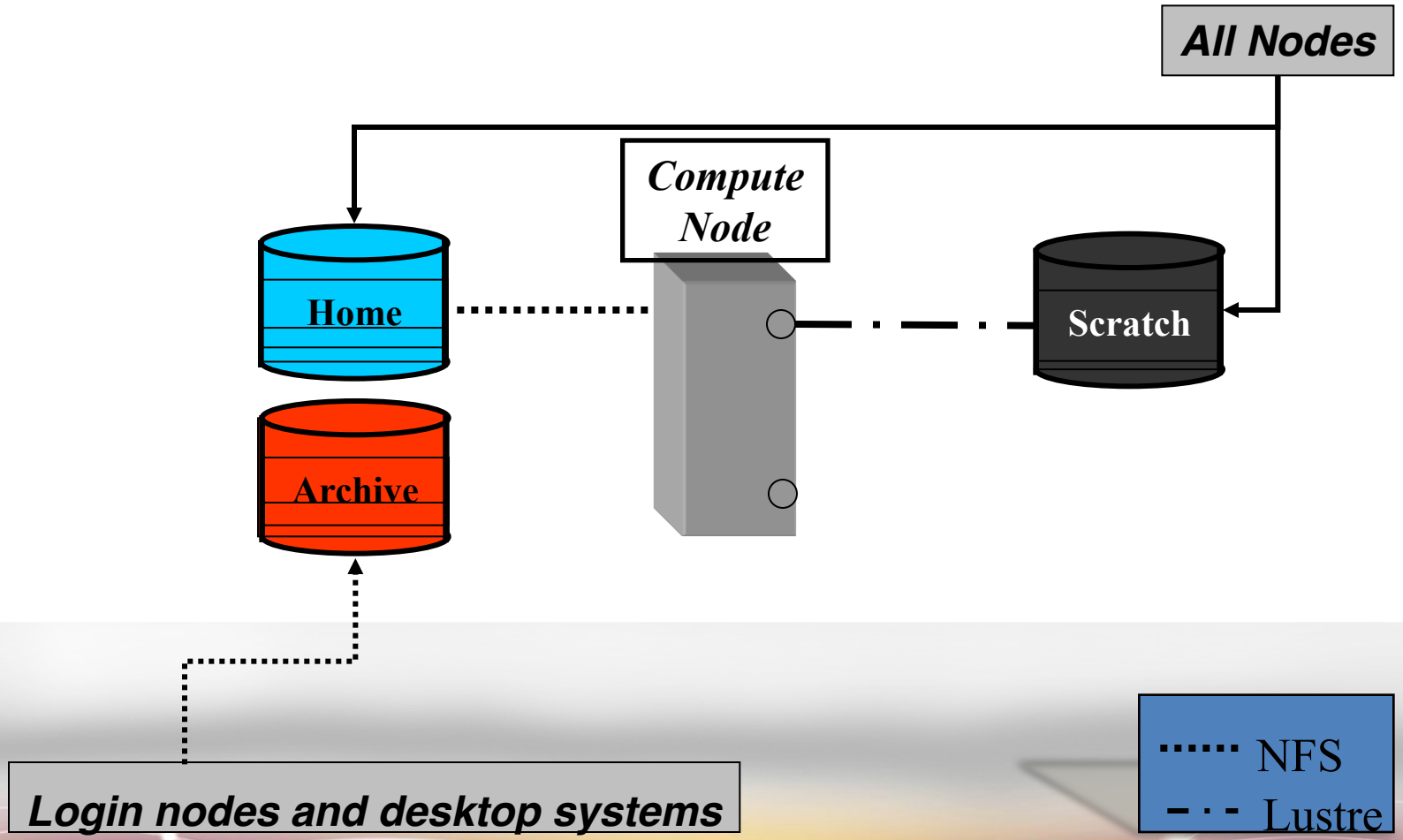
System Information

- **Saguaro is a ~5K processor Dell Linux cluster,**
  - **>45 trillion floating point operations per second**
  - **>9TB aggregate RAM**
  - **>400TB aggregate disk**
  - **Infiniband Interconnect**
  - **Located in GWC 167**
- **Saguaro is a true *cluster* architecture**
  - **Each node has 8 processors and 16GB of RAM.**
  - **Programs needing more resources *must* use parallel programming**
  - **Normal, single processor applications *do not go faster* on Saguaro**

Saguaro Summary

Hardware	Components	Characteristics
Compute Nodes Dell 1955	336 “Harpertown” Nodes 220 “Clovertown” Nodes 200 “Nocona” Nodes 32 “Nehalem” Nodes 32 “Westmere” nodes *Additional special purpose nodes	2.66/2.83 GHz 8MB/Cache 16 GB Mem/node* 8 cores/node* Over 5300 CPUs total
SCRATCH File System I/O Nodes Dell Filers - DataDirect	4 I/O Nodes Lustre File System	300 TB
Login and Service Nodes	3 login & 4 service blades (scheduler, monitor, logs, allocations, etc).	3.2Ghz, 2GB mem
Interconnect (MPI) InfiniBand (TopSpin)	24-port leafs 144-port cores	1 or 2 GB/sec P-2-P Fat Tree Topology
Ethernet (GigE)		128 MB/sec P-2-P

Available File Systems



File System Access & Lifetime Table

Mount point	User Access Limit	Lifetime
/home	50GB quota	Project
/scratch	no quota	30 days
/archive	Extendable	5 years

Interactive Mode

- Useful commands
  - `qsub -I` Interactive mode
  - `qsub -I -X` Interactive mode with X forwarding
  - `screen` Detach interactive jobs (reattach with –r)
  - `watch qstat` Watch job status

Improving Performance

- Saguaro is a heterogeneous system – know what hardware you are using: `qstat -f`

on node {

```
more /proc/cpuinfo /proc/meminfo
echo $PBS NODELIST
```

CPU	Part #	IB data rate	CPUs per node	GB per node	Node ids
Nocona	“Xeon”	No IB	2	4-6	s44-s53
Clovertown	53xx	SDR	8	16	s1-s22
Harpertown	54xx	DDR	8	16	s23-s43
Tigerton	73xx	DDR	16	64	fn1-fn12
Nehalem	55xx	DDR	8	24	s54-s57
Westmere	56xx	DDR	12	48	s58-s59

TORQUE: MPI Job Script II

```
#!/bin/bash
#PBS -l nodes=2:ib,walltime=00:15:00
#PBS -l pmem=3072mb
#PBS -l nodeset=ANYOF:FEATURE:harpertown:nehalem
#PBS -m abe
#PBS -M speyer@asu.edu

module load mvapich/1.1-intel
cd $PBS_O_WORKDIR
mpiexec ./hello
```

- # of cores, walltime
- physical memory
- Heterogeneous arch.
- Send mail when job aborts/begins/ends
- Email address
- Execution command
- *mpirun wrapper script*
- *executable*

Improving Performance

- Scratch space not ideal if
  - You are writing many small files
  - It is more than 75% full
  - Program is not running at DDR (clovertown)
- Many codes can run on multiple cores
  - Sometimes simply finding out which arguments are needed for running on multiple cores (and requesting more cores) does the trick.
- Monitor your code
  - `qstat -f` will give memory usage for your job
  - Try different compilers, libraries

Intel Math Libraries

- MKL (Math Kernel Library)
  - LAPACK, BLAS, and extended BLAS (sparse), FFTs (single- and double-precision, real and complex data types).
  - APIs for both Fortran and C
  - www.intel.com/software/products/mkl/
- VML (Vector Math Library) [equivalent to libmfastv]
  - Vectorized transcendental functions.
  - Optimized for Pentium III, 4, Xeon, and Itanium processors.