INTERSECT

HPC

Introduction to Unix and HPC

What is HPC?

- HPC, or high-performance computing, refers to the application of supercomputers or clusters of computers to computational problems that typically arise through scientific inquiry.
- HPC is useful when a computational problem:
    - **Is too large** to solve on a conventional laptop or workstation (because it requires too much memory or disk space) or
    - **Would take too long** (because the algorithm is complex, the dataset is large, or data access is slow) or
    - **Are too many** – High Throughput Computing

Parallelism on HPC

- HPC systems often derive their computational power by exploiting parallelism
- Programs for HPC systems must be split up into many smaller "sub-programs" which can be executed in parallel on different processors
- HPC systems can offer parallelism at a much larger scale, with 100's or 1000's, or (soon) even millions of tasks running concurrently.
- Writing parallel software can be challenging, and many existing software packages do not already support parallelism & may require development.
- **NOTE: Many tasks cannot be parallelised**

Reasons to use HPC

- You have a program that can be recompiled or reconfigured to use optimized numerical libraries that are available on HPC systems but not on your own system.
- HPC applications are already installed on the HPC machines which is a non-trivial task
- You have a "parallel” problem, e.g. you have a single application that needs to be rerun many times with different parameters.
- You have an application that has already been designed with parallelism
- To make use of the large memory available
- Our facilities are reliable and regularly backed up

When not to use HPC?

- You have a single threaded job which will only run one job at a time (typical of MatLab users)
- You rely on Databases
- You have a lot of data to transfer between your local machine and the HPC on a continuous basis (e.g. per job)
- You need to have a GUI to interact with your program

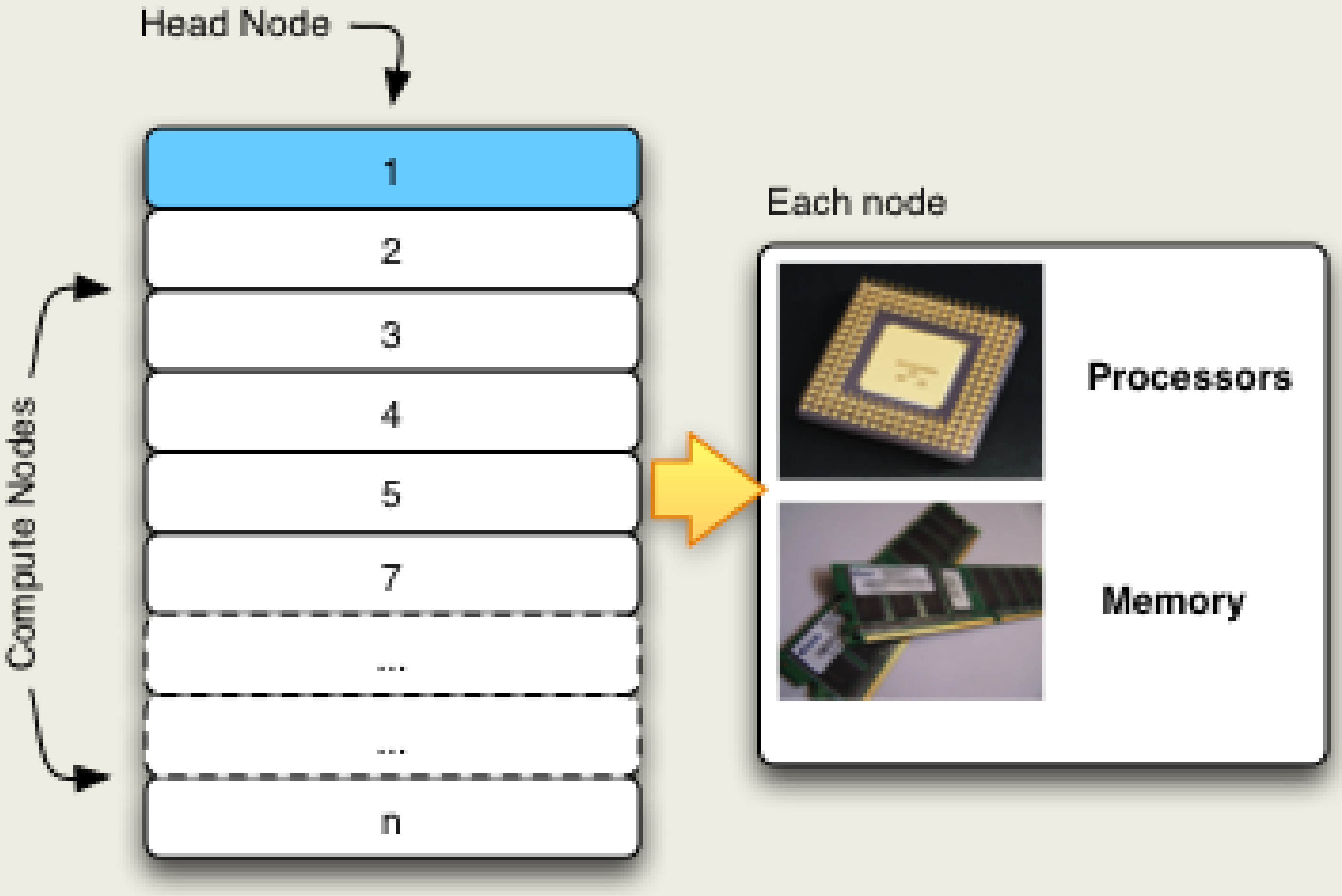
HPC machines

System	Memory Architecture	Cores	Nodes	Memory
Octane (training machine)	Distributed	48	3	48GB
Orange	**Distributed**	**1,600**	**100**	**8TB**
NCI – (Vayu)	**Distributed**	**11,936**	**1492**	**37TB**
NCI – (Name TBA) • **available May 2013**	**Distributed**	**57,472**	**3592**	**158TB**

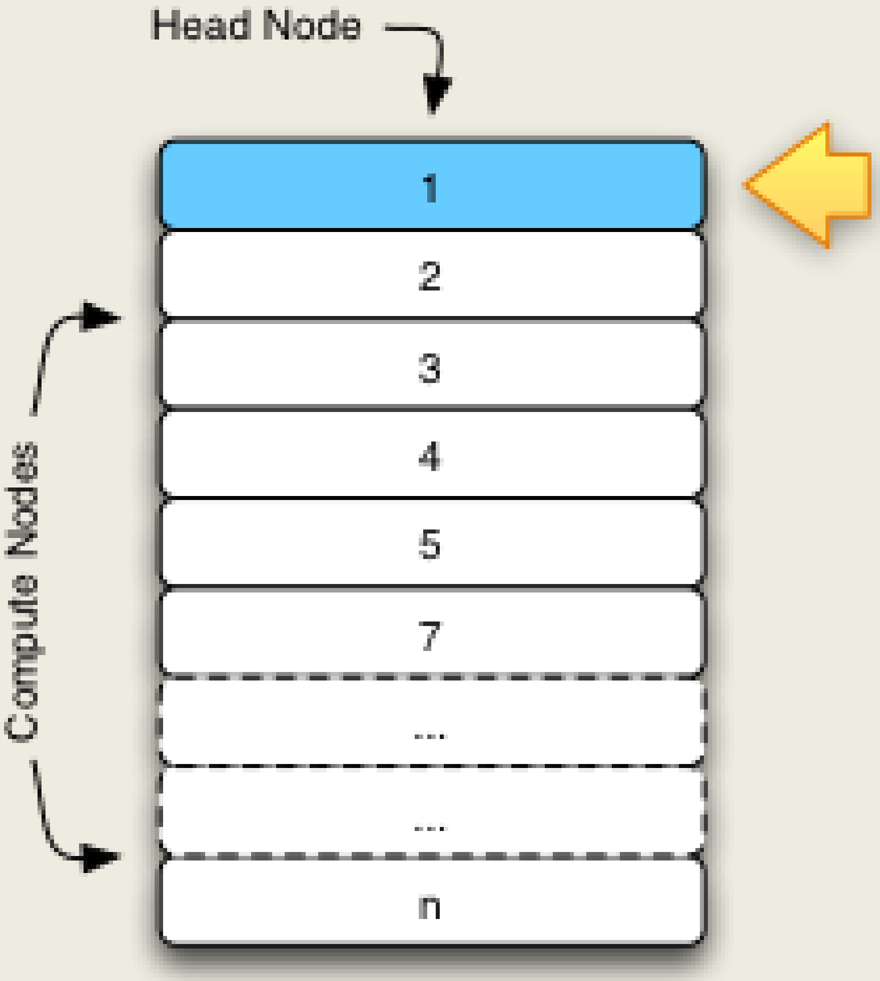
The typical HPC workflow

- In HPC we talk about **jobs**, these are simply commands we wish to run.
- They are generally time consuming and resource intensive.
- Jobs are typically run **non-interactively**, but can also be run interactively
- We add our jobs to a **queue**.
- When the machine has free resources the jobs run.
- Once jobs have completed, we can inspect their output.

The HPC "Cluster"

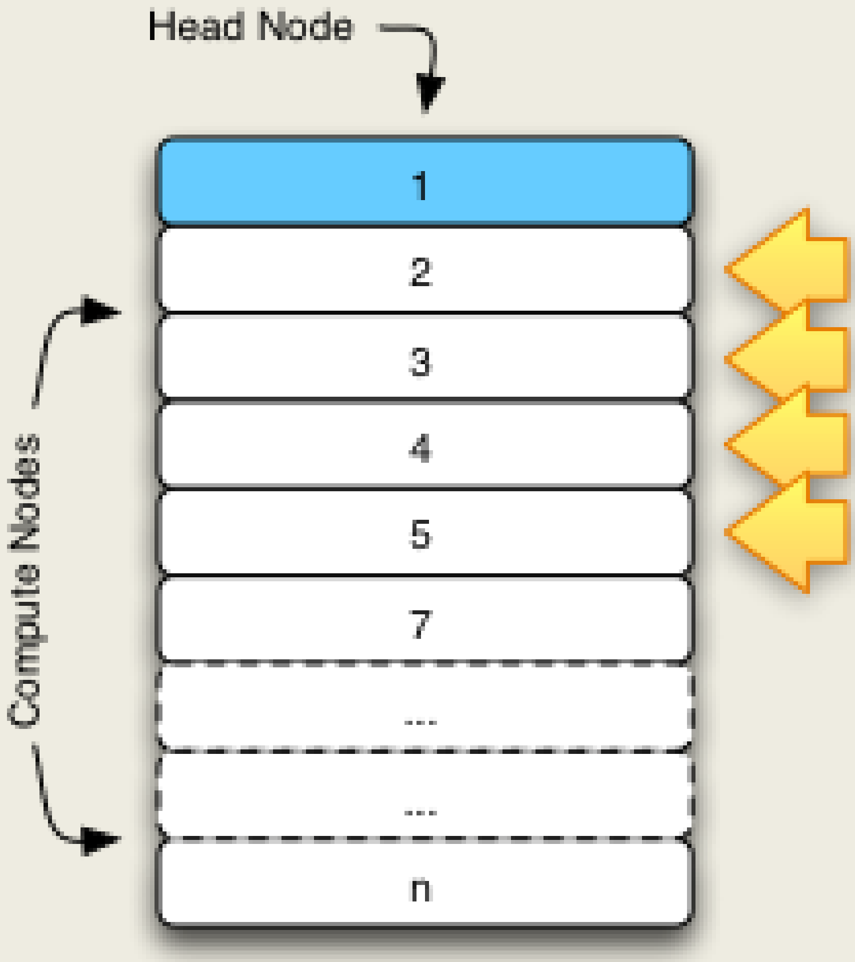


The Head Node



- Interactive programs
- SSH sessions
- Testing
- Compiling
- Queuing jobs

Compute Nodes



- These run your jobs
- Managed by the **scheduler**
- Typically you will not interact with the nodes directly (some users may need to)

Queuing Systems

- **Portable Batch System** (PBS) is the name of computer software that performs job scheduling. Its primary task is to allocate computational tasks, i.e., batch jobs, among the available computing resources.
- The following versions of PBS are currently available:
  - OpenPBS
  - TORQUE
  - PBS Professional (PBS Pro)
  - ANU PBS
- Guide to PBS: http://hpc.sissa.it/pbs/pbs.html

Queuing Systems cont.

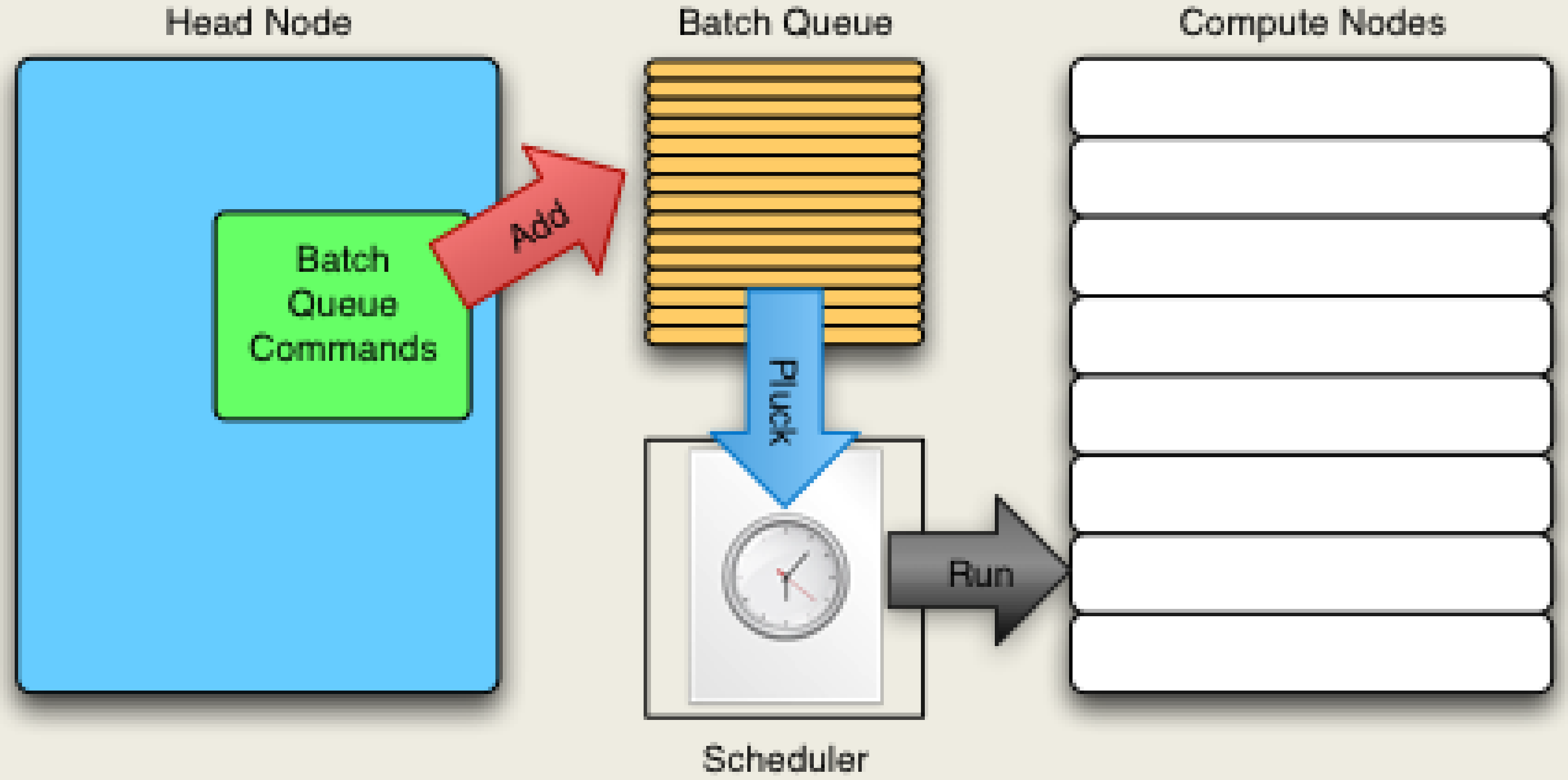
- Another popular batch system is **SLURM** (Simple Linux Utility for Resource Management)
  - Open source, fault-tolerant, and highly scalable cluster management and job scheduling system for large and small Linux clusters.
  - Very useful for use on clusters
  - Platform Tools used by IBM
  - Used by many supercomputers, e.g. TERA 100 at CEA (Europe’s most powerful supercomp.)
- Many banks and commercial entities using batch systems

ANU PBS vs PBS Pro

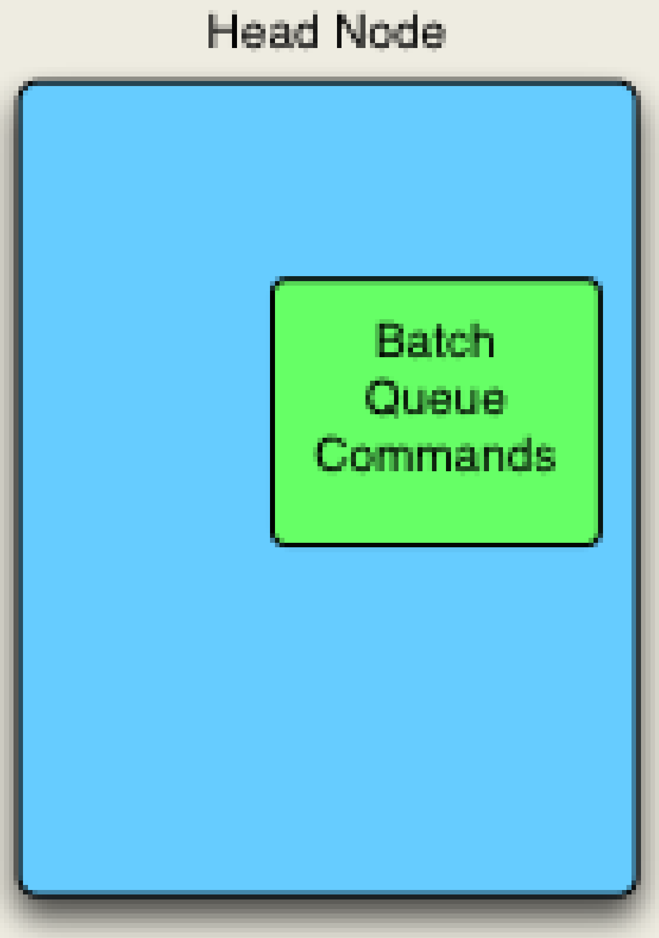
- ANU PBS is a customised version of PBS based on OpenPBS 2.3 maintained by ANU
- Details of ANU PBS modifications are found here:
  http://anusf.anu.edu.au/~dbs900/PBS/local_modifications.html

Batch System	ANU PBS	PBS PRO
Machines using	Vayu & NCI New Facility	Orange & Octane
Code Base	OpenPBS 2.3	PBS Professional
Licence	ANU Licence	Altair Licence

The Batch Queuing System



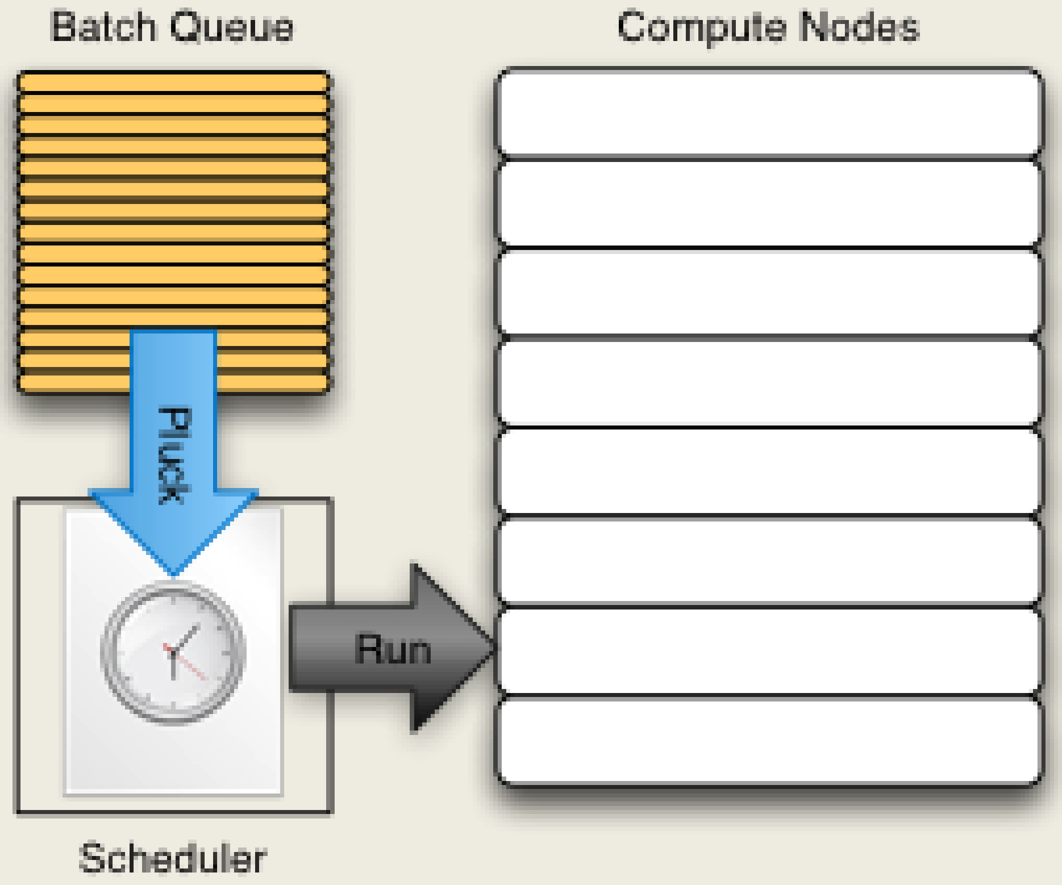
Batch Queuing component



- The batch system is a normal program
- Lets you add and remove jobs from the queue and monitor the queue
- Script/command line driven

The Scheduler component

- Allocates jobs to compute nodes
- Optimizes usage of resources
- "Optimize" can mean many things
- Non-trivial
- Never interact with directly



PBSPro Commands

In order to use the batch system productively, we need to know how to perform three actions:

- Add a job to the queue
- Remove a job from the queue
- See where our job is in the queue

Command	Description
`qsub <job-script>`	Submit a job (add to queue) Returns a <job-number>
`qdel <job-number>`	Delete job (remove from the queue)
`qstat <job-number>`	Monitor jobs
`qalter <job-number>`	Modifies the attributes of the job or jobs

Module Package

In order to use the available modules more productively, we need to know how to perform four actions:

- Add a job to the queue
- Load or unload modules for use
- List all available modules

Command	Description
`module avail`	Will list all available module files in the current MODULEPATH
`module load/unload`	Will load/unload a modulefile into the shell environment
`module list`	List all loaded modules
`module show [modulefile]`	Will show information about the modulefile

Monitoring the queue

Command	Description
`qstat -a`	List all jobs in the queue
`qstat -u <username>`	List all jobs of a particular user
`qstat -f <job-number>`	Show detailed information about a job

Exercise 1: Monitoring the queue with qstat

Add a job to the queue

- To add a job to the queue, we write a **job script**.
- The job script is a simply a script.
- It has some special comments that pass information to PBSPro.
- When we want to queue the job, we pass its filename as a parameter to **qsub**, e.g.
  - **qsub <job-script>**
- The batch queuing system will return a number that uniquely identifies the job.

A sample PBS job script

NOTE: This script will not NCI facilities

```
#!/bin/bash
# Request resources
# * 10 minutes wall time to run
#PBS -l walltime=00:10:00
# * 1 node, 1 processor
#PBS -l nodes=1:ppn=1
# * 100 megabytes physical memory allocated to job
#PBS -l mem=100mb
# Specify a project code (for accounting)
#PBS -P a40
cd $PBS_O_WORKDIR
# Specify the job to be done
date
sleep 10
date
```

You’ve got mail!

```
# Set email address
#PBS -M fred@intersect.org.au
# Send an email when jobs
# begins (b), gets aborted (a)
# and ends (e)
#PBS -m abe
```

Exercise 2: Submitting a sample job

Useful Environment Variables

- These are available in the context of your job script.

Command	Description
PBS_O_WORKDIR	The directory the job was submitted from
PBS_JOBID	The job number given when the job was submitted

Job limits on Orange

- 200 hours of **walltime**
- 64GB of **memory** per standard node. e.g. 128GB for 2 nodes etc.
- 256GB of **memory** per large memory nodes
- **NOTE**: If you grab a node with 64GB, you can effectively use about 60GB as the OS uses memory

Priorities of Jobs

In order of importance, jobs are prioritised in this order:

1. Resources available to the project
2. Walltime
3. Number of jobs (fair share)

Best strategy

- Submit jobs constantly/daily
- Have about 10-20 jobs in the machine
- Be realistic with walltime
- Don’t ask for resources you don’t need!

NCI Facilities

Disk Type	Disk Usage
VAYU	• Sun Constellation Cluster with 1492 nodes, each containing 2 quad core Nehalem processors summing up to 11,936 cores. 37TB RAM and 800 TB disk space. Commissioned in 2010. • The unit of **shared memory parallelism is the node**, which comprises dual 8-core processors, i.e., 16 cores.
NCI Upcoming System (available in May)	• 57,472 cores in the compute nodes; • Approximately 160 TBytes of main memory; • Infiniband FDR interconnect; and • Approximately 10 PBytes of usable fast file system (for short-term scratch space). • Will be commissioned in its entirety in early 2013.

NCI Facilities

VAYU	New NCI Facilities
66% of Nodes have 32Gb (2Gb/core)	96.5% of Nodes have 24Gb/node
33% of Nodes have 64Gb (4Gb/core)	3.2% of Nodes have 48Gb/node
2% of Nodes have 128Gb (8Gb/core)	0.3% of Nodes have 96Gb/node

Software on NCI

Area	Software
Computational Chemistry	ABINIT, Amber, CPMD*, GULP*, NAMD*, Molpro etc.
Bioinformatics	AbySS, BEAST, BIOPERL, Cufflinks, MAW, etc.
Math Libraries	ARPACK, BLACS, Boost, FFTW, GSL, MKL, Tao
Statistics & Maths Env's	Maple*, Mathematica*, MatLab*, Octave*, R, Stata*

- Asterisked items indicates that discussion with NCI facility staff is required before use (Licensing issues)
- http://nf.nci.org.au/facilities/software/index.php

Orange Physical Disks -

- Which of the 3 disks to use and when?

Disk Type	Disk Type	• Disk Usage
Panasas	59Tb	• Parallel global file system • All nodes see the Panasas disk • Very fast for large files • Very slow for small files • System director blade creates metadata for each file
SGI	50Tb	• An NFS mounted file system • Uses old technology, therefore very robust • Scales nicely for clusters up to 100 Nodes (very good for Orange)
Local Scratch	200Tb	• Exist in each node • No network is necessary making these the fastest disk • If you have a lot of I/O, you should copy your data to here and work here

Disk Partitions - Orange

/home	
Mounted under:	/home/username
Disk Type	SGI Disks
Size:	60GB default
Backed up:	Yes
Speed:	Intermediate disk (SGI Disks)
Life time:	Permanent

Disk Partitions - Orange

/projects/project-name	
Mounted under:	/projects/project-name
Disk Type	Panasas Disk
Size:	no default size
Backed up:	Yes
Speed:	High speed
Life time:	Till end of the running year - merit allocation period

- There will also be some “repository space” for large datasets, such as bioinformatics databases

Disk Partitions - Orange

/data2	
Mounted under:	/data2 on each node
Disk Type	xx
Size:	Limit of disk - 2TB
Backed up:	No
Speed:	Fastest
Life time:	Job duration

Warning: This partition is shared among users, so can be “filled up” (with other jobs) while your job is running!

Disk Partitions

You can find out more about the partitions on the HPC machine using the **df** command.

Command	Description
df -h	Show disk free space for all partitions in human readable format

You can find out more about current disk usage, using the **du** command.

Command	Description
du -hs .	Show disk usage of current directory in human readable format

Quotas

- There is no quota on scratch disks for performance reasons.

- The quota on /projects/project-name depends on your allocation.

- 60 GB soft limit for /home.
- 80 GB hard limit for /home (30 days).

More Info on NCI & Orange

- Read more about Orange and NCI Facilities

  - http://www.intersect.org.au/hpc-news
  - http://www.intersect.org.au/orange
  - http://www.intersect.org.au/nci_next
  - http://www.intersect.org.au/orange-handbook

Resource Allocation Round

- Merit-based system by which Intersect members can gain access to our HPC facilities
- Applications reviewed by HPC staff (for technical complexity and track record) and the Intersect Resource Allocation Committee (for research merit)
- Applications to Intersect's HPC systems will be made through NCI's forms in October each year
- Applications must be made by Academic Staff at an Intersect member institutions. PhD students can make use of the facilities, the lead CI must be an academic staff member.
- Questions to: hpc_support@intersect.org.au

Register with NCI (step 1)

Register **a new Id with NCI**:

http://nf.nci.org.au/accounts/forms/user_registration.php

This will provide your details to NCI

Register with NCI (step 2)

Apply for **a project from NCI**:

https://nf.nci.org.au/accounts/projects_new/APP_form.php

This will provide link your Id to your Project

Project Registration Form

- Pick INTERSECT under partner/scheme on the first page of the project registration form or else you won't get access to Orange
- If you're unsure about which machine to get access to, email hpc_support@intersect.org.au who can advise you
- You can add accounts to an existing project also!

Conclusion

- In this course we have covered the basics of the Unix command line, transferring data, and the specifics of our HPC machine
- As different machines have different PBS systems, scripts that work on Octane may not work on NCI facilities
- Please complete our survey!
- Any questions?