Basic linux and LSF

Linux and LSF information FAQs

What are the current Cluster queues?
Additional nodes were added to the cluster along with new queues. These changes streamline performance and application integration support. In addition contributed nodes by faculty are listed below. Use of contributed nodes is authorized by node owner. If you are not authorized to use the contributed node(s) you may use the short_all6, normal_all6 or long_all6 queues at lower priority but subject to preemption and other constraints.

For example, note that the queues with the _public suffix will only submit jobs to the public nodes (hosts with the name nodeXX and nodebXX). Queues with the _all suffix can take advantage of user contributed nodes (hosts with the name contribXX) as well as public nodes, however jobs that get dispatched to the user contributed nodes may be preempted without warning by jobs submitted to those nodes by the owner of those nodes.

Node owner	New queue	Old queue	Comments
Public	admin_public	admin	system usage only
	express_public6	express	short term high priority, special needs
	int_public6		interactive jobs, high priority, 4 hour limit
	exbatch_public6		batch jobs, high priority,low capacity, 2 week limit, exclusive
	paralleltest_public6	paralleltest	short term debugging
	short_gpu		2 hour limit
	normal_gpu		72 hour limit
	long_gpu		28 days limit
	short_public6	short	2 hour limit
	parallel_public6	parallel	2 week limit 2-64cpus
	normal_public6	normal	default queue - 3 days limit 1 cpu
	long_public6	long	14 days 1 cpu
	dregs_public6	dregs	364 days 1 cpu
Public shared	short_all6		shared across all nodes, lower priority
	normal_all6		shared across all nodes, lower priority
	long_all6		shared across all nodes, lower priority
ATLAS	atlas_prod_rhel6		Physics Atlas support
	atlas_analysis_rhel6		Physics Atlas support
Khardon	int_khardon		contributed nodes, interactive 4 hour limit
	express_khardon		contributed nodes, 30 minute limit
	short_khardon		contributed nodes, 2 hour limit
	normal_khardon		contributed nodes, 3 day limit
	long_khardon		contributed nodes, 2 weeks limit
Miller	int_miller		contributed nodes, interactive 4 hour limit

	express_miller		contributed nodes, 30 minute limit
	short_miller		contributed nodes, 2 hour limit
	normal_miller		contributed nodes, 3 day limit
	long_miller		contributed nodes, 2 weeks limit
Abriola	**int_abriola**		contributed nodes, interactive 4 hour limit
	express_abriola		contributed nodes, 30 minute limit
	short_abriola		contributed nodes, 2 hour limit
	normal_abriola		contributed nodes, 3 day limit
	long_abriola		contributed nodes, 2 weeks limit
Napier	**int_napier**		contributed nodes, interactive 4 hour limit
	express_napier		contributed nodes, 30 minute limit
	short_napier		contributed nodes, 2 hour limit
	normal_napier		contributed nodes, 3 day limit
	long_napier		contributed nodes, 2 weeks limit
CBI	**int_cbi**		contributed nodes, interactive 4 hour limit
	normal_cbi		contributed nodes, 3 day limit
	long_cbi		contributed nodes, 2 weeks limit

How do I choose between queues:

You may view queue properties with the bqueues command:

-bash-3.2$ bqueues

And extra details by queue name:

-bash-3.2$ bqueues -l normal_public6

What is the default queue?
If you do not specify a queue by name in your bsub arguments, your job goes to the default queue, which is normal_public6.

Where do I find basic unix/linux resources?

There are many web based tutorials and howto's for anything linux oriented. Some sites of interest:

linux-tutorial, Unix info , linux.org

What are some of the basic linux and related commands?

Most usage is centered around a dozen or so commands:

ls, more, less, cat, nano, pwd, cd, man, bsub, bkill, bjobs, ps, scp, ssh, cp, chmod, rm, mkdir, passwd, history, zip, unzip, tar, df, du, head, tail, grep

See the man pages for complete documentation. Here is a short description of some.

Basic Unix Commands

Action Needed	Command	Usage
Display contents of a file	**cat**	cat *filename*
Copy a file	**cp**	cp [-op] *source destination*

Change file protection	**chmod**	chmod *mode filename* **or** chmod *mode directory_name*
Change working directory	**cd**	cd *pathname*
Display file (/w pauses)	**more**	more *filename*
Display first page of text	**less**	less *filename*
Display help	**man**	man *command* **or** man *-k topic*
Rename a file	**mv**	mv [-op] *filename1 filename2* **or** mv [-op] *directory1 directory2* **or** mv [-op] *filename directory*
Compare file	**diff**	diff *file1 file2*
Delete file	**rm**	rm [-op] *filename*
Create a directory	**mkdir**	mkdir *directory_name*
Delete a directory /w files in it	**rmdir -r**	rm -r *directory_name*
Delete a directory	**rmdir**	rmdir *directory_name*
Display a list of files	**ls**	ls [-op] *directory_name*
Change the password	**passwd**	passwd
Display a long list (details)	**ls -l**	ls -l *directory_name*
Display current directory	**pwd**	pwd
Display mounted filesystems	**df**	df

What text editors are available?
nano, nedit, vi, vim, emacs

How do I strip out ^M embedded characters in my files I transferred from my PC?
Use the **dos2unix** command on the file. There is man page documentation available for further info.

What is a man page?

man pages are linux/unix style text based documentation. For example, to obtain documentation on the command cat:

> man cat

> xman
is the command for the x-based interface to man.

> man man
is the man documentation.

> man -k graphics
finds all related commands concerning graphics.

Are the compute nodes named differently from the old cluster compute nodes?

Yes. You should not hard code the names anywhere. The convention is node01, node02, ... To see the current list use the bhosts command.

How can I verify that my requested storage is mounted?
Use the **df** command.

What is LSF?
Load Sharing Facility

Why do I have to submit jobs to compute nodes via LSF?

The cluster has been configured to allocate work to compute nodes in a manner that provides efficient and fair use of resources. A job queueing system called **LSF** is provided as the work interface to the compute nodes. Your work is then distributed to queues that provide compute node resources. Login to compute nodes via ssh is not suggested and you will be asked to refrain from using the resources in that manner; let **LSF** do it!

How do I find LSF documentation?
The vendor website has documentation, the cluster man pages and this local link.

How to request memory resources on the cluster?
Memory usage is often hard to estimate in a new context or program. Generally speaking the larger the input data the more memory and perhaps other resources are used. Since the cluster has compute nodes with different amounts of memory, we have created an LSF resource so that you may request in a **bsub** job submission the upper limit of memory required. This is helpful in many ways in preventing resource collisions and to other jobs sharing the same compute node.

The LSF defined resources are: **Mem8**, **Mem16**, **Mem24**, **Mem32**, **Mem48**, **Mem64**, **Mem80**, **Mem96**, **Mem112**, and **Mem128** . These correspond to gigabytes of ram.

What happens if I don't explicitly use a defined ram memory resource when I submit a job?
LSF will place your job(s) on a node(s) that is considered available without understanding your needs for ram. If your job starts to request more ram than is available on that node given it's current load, then your job(s) may be at risk for taking too long to run or may put the node in an unresponsive state. This can affect other users jobs.

Suppose my jobs have a very small memory requirement, say 100meg, do I have to use the defined memory resources?
Not usually. There are many cases like this, and experience tells us that this is usually not a problem.

My program needs access to more than 16 gig or ram, what are my options?
An LSF resource has been defined to identify those nodes with 24 gig of ram. You access this through a **bsub** command line option, -R, when you submit your job.

-bash-3.2$ bsub -R Mem24 -queue normal_public6 ./myprogram

I sometime notice that my job duration can vary when I rerun a program with exactly the same inputs, condition, etc... Why?
The cluster has a mix of several different Intel Cpus and motherboard combinations. The absolute performance potential is different among them but with the choice of LSF queue and mix of running jobs your job competes with, your results will vary. This is not something that is predictable when the cluster is well loaded.

I see that there are some nodes with more than 32gig ram, such as 48 and 96 gig. How do I access them in exclusive mode since I need almost all the ram and my expected job duration is just minutes?

-bash-3.2$ bsub -q express_public6 -x -R Mem48 ./myprogram

I have a program that is threaded. Is it possible to submit a job that will "take over" a node, or at least the 4 cores on a single chip?

You can force your job to run exclusively on a compute node (the only job running) by using bsub -x when you submit the job. It may take a bit longer to run though as it will have to wait in the queue for a compute node to become fully available.

You should also be able to use a combination of -n and -R to request a specific number of CPU's on one host. The following example should reserve 4 CPU's for your job on one compute node:

-bash_3.2$ bsub -n 4 -R "spanhosts=1" ./yourprogram

How does one use a node exclusively?
Currently the only queues that allow exclusive use is the **express_public6** and **exbatch_public6** queues. However, not all jobs are suitable, so please inquire with an email to cluster support and describe what you intend to do.

How does one actually invoke a job exclusively?
LSF bsub command has the -x option. To send your job to a node that has extra memory and runs exclusively for hours.
-bash-3.2$ bsub -q exbatch_public6 -x -R Mem16 ./myprogram

How does one make use of nodes with /scratch2 storage?
Note that is is disk storage and not ram memory.
Access to this storage is by request. Please make this request via cluster-support@tufts.edu.

If you submit a job with the following, LSF will place a job on nodes with /scratch2 partitions.
For example, to request at least 40gig of storage for a job to run in the long_public queue try:

-bash_3.2$ bsub -q long_public6 -R "scratch2 > 40000" ./your_jobname

Other queues are possible as well. Note, the storage argument is in megabytes.

What are some of the most common LSF commands:

Action Needed	Command	Usage
System verification	**lsid**	lsid
Display load levels	**lsmon**	lsmon
Display hosts	**lshosts**	lshosts
Summarize past usage	**bacct**	bacct **or** bacct *job ID* #
Display hosts	**bhosts**	bhosts

View current jobs	**bjobs**	bjobs **or** bjobs *job ID #*
Run LSF batch job	**bsub**	bsub [-op] *filename*
Kill a job	**bkill**	bkill *job id #*
Review/select queue	**bqueues**	bqueues **or** bqueues *queue_name*
Suspend a job	**bstop**	bstop *job ID #*
Changes job order (new or pending)	**btop**	btop *job ID* | *"job_ID"(index_list)"*
Resume suspended jobs	**bresume**	bresume *job ID #*
View job history	**bhist**	bhist *job ID #*
Modifying or Migrating jobs	**bmod**	see man page

How can I get notified when my lsf submitted jobs finish?

By default no mail is generated. You need to add the -u option to bsub. As an example:
-bash-3.2$ bsub ... -u firstname.lastname@tufts.edu sleep 10

This will cause an e-mail to be sent when the job finishes, containing a summary of the job, the output, CPU & memory utilization, etc.

Also note that this action might send an amount of output to your email account that it may put you over your email quota, thus preventing receipt of mail!

I need to submit 100s of jobs using the same program but with different input data, what is the best practice?

LSF provides a structure called a job array that allows a sequence of jobs that share the same executable and resource requirements, but have different input files, to be submitted, controlled, and monitored as a single unit. Using the standard LSF commands, you can also control and monitor individual jobs and groups of jobs submitted from a job array.

Now that I submitted 100s of jobs, I realized that I don't want them to run after all, how do I kill them all?
All the jobs listed when you use the **bjobs** command can be removed by doing the following:
-bash-3.2$ bkill 0

I have many jobs running and pending, how do I kill off only the pending jobs?
> bjobs | awk '$3=="PEND" {print $1}' | xargs bkill

Here is another approach:
> bkill ` bjobs -u username |grep PEND |cut -f1 -d" "`

I have subitted many jobs and don't recall which queues I used, how do I find out the status?
> qstat -u your_tufts_utln

I have a job in one queue, but would rather have it in another. How do I migrate the job?
Use the lsf command, bmod. For example:
-bash-3.2$ bmod -q express_public6 <job_number>

This will migrate your job with <job_number> to the express_public6 queue or some other queue.

The contributed nodes often seem idle. How do I use them if I am not in a particular contributed node queue user group?
There are three queues that will make use of all compute nodes. The Public Shared queues allow job placement to all nodes via LSF. When contributed nodes are idle and there are many jobs already in the normal_public6 or long_public6 queue, use of the Public Shared queues will likely land your jobs on idle contributed nodes. See above table for corresponding Public Shared queue names and properties. For more detail on a particular queue:

-bash-3.2$ bqueues -l short_all6

How can I submit jobs to LSF on the cluster from my workstation without actually logging into the cluster?

If you have ssh on your workstation, try the following:
> ssh cluster6.uit.tufts.edu ". /etc/profile.d/lsf.sh && bsub -q queuename ./yourprogram"
where queuename is one of the above queues.

Suppose I want to copy data via scp from my bash script that is running on a compute node to the /scratch/utln storage area of the login node. How do I reference it?

scp filename tunic6.uit.tufts.edu:/scratch/utln
Note, your utln username is needed.

How does a program or shell on one compute node reference data from another compute node's local scratch storage?

The local scratch directory on each compute node is automounted when requested. For example, to access file abcd.data on compute node 07 from compute node 19; the path is:

/cluster/scratch/node07/utln/abcd.data

This will give you access to your scratch directory and file on node07.

What is the path to reference from a job on a compute node to the storage on the login node?
/cluster/scratch/tunic6/utln/ ....

How do I convert mixed case file names in a directory to lower case?

Issue the following in the directory of interest:

find . -name "**A-Z**" | cut -c 3- - | awk '{print $1,tolower($1)}' | xargs -i echo "mv {}" | csh

This will find everything with uppercase letters and rename it to the
same thing with all lowercase.

How do I uncompress and extract many .gz and corresponding tar files?
> ls -1 *.gz | xargs -n 1 -r -I {} tar zxf {}

Sometimes I get a cryptic message about too many open files, what is that?
There are several resource settings associated with a default account. To see the settings:

-bash-3.2$ ulimit -a

The default setting is 2048 for the open files parameter. A user may increase it up to 10K in 1024 increments. To set it to 4096:

-bash-3.2$ ulimit -n 4096

However, this shows what has happened on the headnode only.

Since jobs execute on compute nodes you should include this in a simple shell script along with your bsub command, so that the new setting takes effect on the compute node as well.

Additional User contributed Cluster Documentation

The following has been contributed by Rachel Lomasky. **Click Here** for the web version or under Wiki Page Operations a Pdf version is available as an attachment.