Introduction to UGER

Univa GridEngine Research
compute cluster @ Broad

Peter Ragone
Scientific Computing Services, Research Computing

Today's topics

What is UGER
- a brief introduction

Accessing UGER
- a brief introduction to software and dotkit

Types of UGER sessions
- Interactive vs batch

Examples
- Stating some options
- Monitoring and managing your work

Q&A

Document for cut-and-paste http://broad.io/uger_supp

Job Scheduler - What? Why? and How?

What is a job scheduler?

Finds a free computer to run your linux software.

Why use a job scheduler?

Software requires expanded RAM or cores.

How do I use the job scheduler?

Submit a command from the command line. There are many options!

UGER - What? Why? and How?

What is UGER?

- Univa Grid Engine Research cluster

Why UGER?

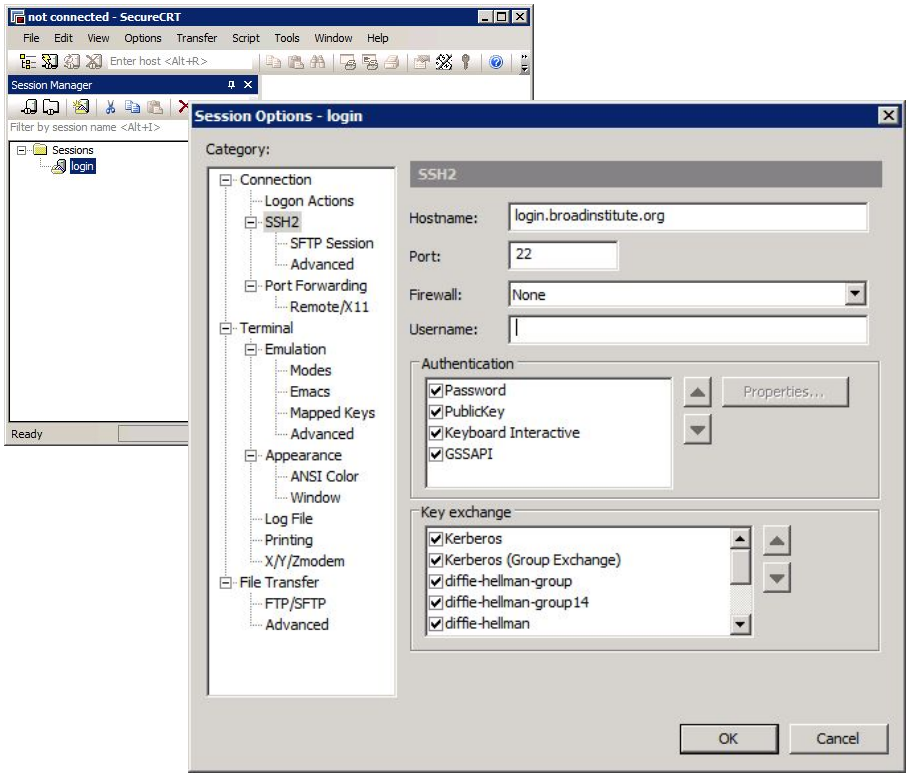
- Open-source GridEngine variants available in cloud environments
- Univa offers enterprise support, more robust job scheduler

How is UGER different (from LSF)?

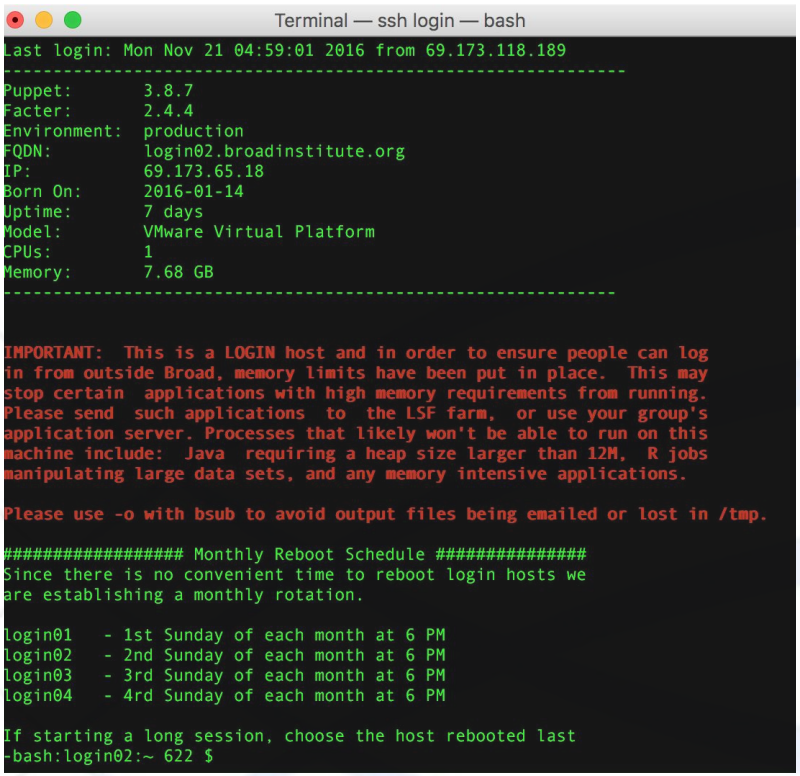
- Assumes job scripts are written for environment setup and job execution

Accessing Unix at the Broad

Windows: SecureCRT



Mac OS X: terminal.app



login.broadinstitute.org

http://broad.io/login

A quick sidebar: dotkit

Dotkit - central software repository for cluster computing

```
use
use -l <text to search>
```
to view visible dotkits

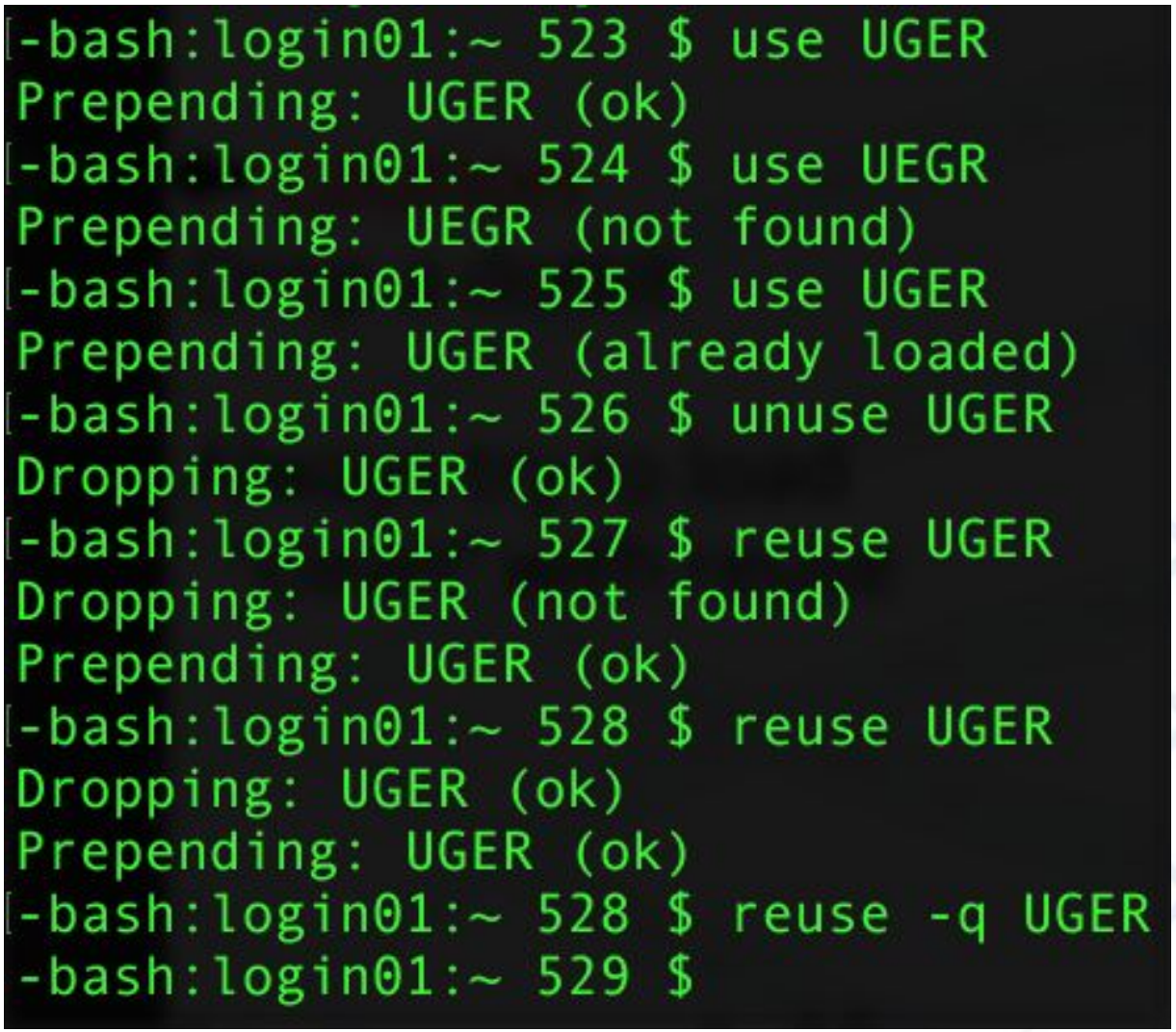
add UGER commands to your environment with dotkit

```
use UGER
```

```
bash:login04:~ 501 $ qstat
bash: qstat: command not found
bash:login04:~ 502 $ use UGER
Prepending: UGER (ok)
bash:login04:~ 503 $ qstat
bash:login04:~ 504 $
```

Need a new dotkit? email help@broadinstitute.org

How to read dotkit output



← Successful

← Dotkit NOT loaded

← Caution

← Drop a Dotkit

← Always try to load "clean" with reuse

← Recommended for job scripts

Two Types of UGER sessions

If you have software you need to respond to along the way:

“**Interactive**” = similar to a login server, but better.

If you have software you just want to send off to run:

“**Batch submission**” = send a command or script.

Interactive sessions

Log on to a login server

```
use UGER

ish
```
or
```
qrsh -q interactive
```

Remember:

UGER dispatch creates a fresh session on a new host

➜ Dotkits will need to be reloaded
➜ Can add requests for additional RAM, cores

Basic command line batch submission

toy example: submit ls

```
qsub -b y ls
```

By default, GridEngine expects a job script

`-b y`(es) run a **binary** instead of a job script

Job submission defaults

Queue:	“short” (has a two hour runtime limit)
memory request:	1G
Execution directory:	your home directory
output filename:	<jobName>.o<jobID> (in execution directory)
stderr filename:	<jobName>.e<jobID> (in execution directory)
name:	“word” in job command (eg. ls for the toy example above)

What is a “queue”?

Term is used two ways:

“**Job is queued for dispatch**” means:

Each Job is in a line, waiting for dispatch.

“**Job is in the ‘short’ queue**” means:

The job will be running in a resource environment defined by the cluster administrator.

UGER has 3 queues for common use:
“interactive”, “short”, and “long”

Simple job script using dotkit

```
-bash:login03:~ 855 $ more /broad/software/unixdemo/intro2uger/simple.sh
#! /bin/bash

#  "#$" lines indicate flags for qsub
#$ -N FirstScript

source /broad/software/scripts/useuse
reuse -q BLAST

blastall -p blastn -W 28 -v 10 -b 2 -d /broad/data/blastdb/other_genomic/other_genomic
-i /broad/data/blasttest/G889P81FG1.T0.fasta -o blast.out
```

set up dotkit (useuse)

request a dotkit (with quiet mode)

supply your command

qsub /broad/software/unixdemo/intro2uger/simple.sh

Note: Job scripts must be set with permission as '**executable**'! ("chmod +x")

Monitoring jobs in UGER

qstat lists your UGER jobs (if any)

```
-bash:hw-uger-1000:~ 560 $ qstat
job-ID     prior   name       user         state submit/start at     queue                          jclass                         slots ja-task-ID
-----------------------------------------------------------------------------------------------------------------------------------------------
   1239661 0.46284 test       jlchang      r     03/16/2016 21:13:07 short@hw-uger-1081.broadinstit                                1
   1239987 0.46284 QRLOGIN    jlchang      r     03/16/2016 21:37:16 interactive@hw-uger-1000.broad                                1
```

qstat -j <jobID> list detailed job info (qw or r)

Job ID

```
sge_o_workdir:              /home/unix/jlchang
sge_o_host:                 login02
account:                    sge
stderr_path_list:           NONE:NONE:/dev/null
hard resource_list:         m_mem_free=1g
mail_list:                  jlchang@login02.broadinstitute.org
notify:                     FALSE
job_name:                   QRLOGIN
stdout_path_list:           NONE:NONE:/dev/null
priority:                   0
jobshare:                   0
hard_queue_list:            interactive
```

Job script for our toy example

```
-bash:login02:~ 593 $ more /broad/software/unixdemo/intro2uger/toy.sh
#!/bin/bash

ls
```

```
cd /broad/software/unixdemo/intro2uger
qsub toy.sh
```

Add qsub options to our job script: `qsub toy2.sh`

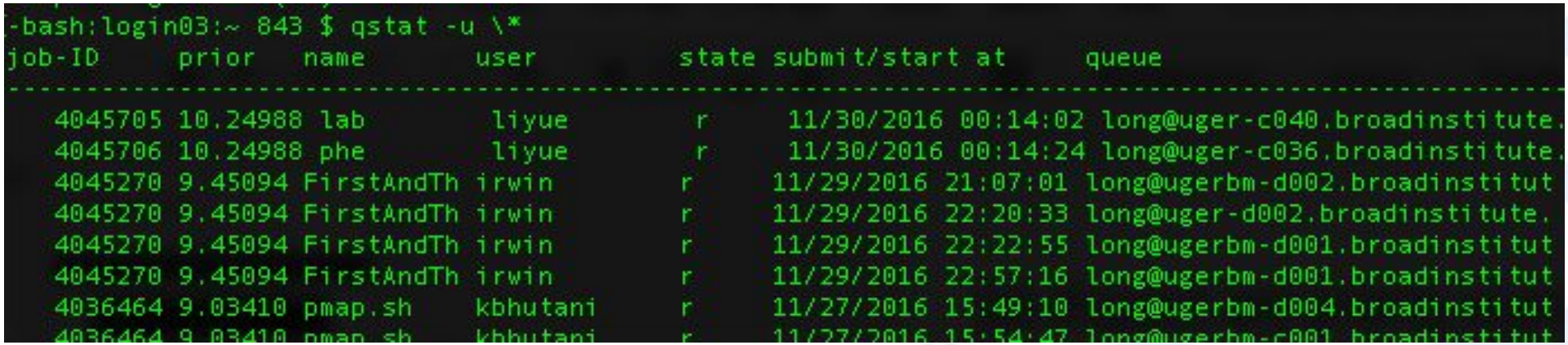
```
-bash:login01:~ 730 $ more /broad/software/unixdemo/intro2uger/toy2.sh
#!/bin/bash

#  "#$" lines indicate flags for qsub
#$ -q long
#$ -N toy2

sleep 300
ls
```

command line flags override job script settings

```
qsub -N toy3 toy2.sh
```

Job dispatch and priority assessment

Compare the priority of your job to others in UGER:
qstat -u “*” cluster jobs for all users



The default is to show all queues.

Basic job submission options

	flag	options	default
Queue	-q	long, interactive	short
Binary	-b y		n
Job exec location	-cwd, -wd <path>		home directory
Output	-o	Filename or directory	<name>.o<jobID>
Stderr	-e	Filename or directory	<name>.e<jobID>
Output + Stderr	-j y		n (eg. separate files)
Job name	-N <name>		“word” in cmd
Project	-P <project name>		Broad Project
Memory	-l h_vmem=XG		1g (all queues)
Cores	-pe <pe_type> <#>	smp (-binding linear:#), openmpi	1 core, no pe

Short queue = 2 hour runtime (default)
Interactive queue = maximum of two sessions

qsub options examples

Use the current working directory (run job, write output)

```
qsub -cwd toy.sh
```

Specify output directories (in a writable directory)

```
qsub -cwd -j y -o /broad/hptmp/jlchang -N cwd toy.sh
```

Specify a working directory

```
qsub -wd /broad/hptmp/jlchang -N wd toy.sh
```

Reserve memory for your job (example 2G)

```
qsub -l h_vmem=2G toy.sh
```

memory requests are hard limits in UGER

- jobs may not use more memory than requested
- jobs over requested memory will be killed

*** Resource values like h_vmem are cluster-specific. Advice from Google may lead you astray…

Obtaining information on completed jobs

qacct -j <jobID> list completed job details

qacct -f <archived log> -j <jobID>

```
-bash:login02:~ 641 $ qacct -j 4007521
==============================================
qname        short
hostname     hw-uger-1025.broadinstitute.org
group        broad
owner        ragone
project      broad
department   defaultdepartment
jobname      sleep
jobnumber    4007521
```

Job management

Delete an unwanted job

qdel <jobID>

Force delete stuck (dr and dt) jobs:

qdel -f <jobID>

Change job options (queue, priority) for pending jobs:

qalter -q long <jobID>

qalter -p -200 <jobID>

changes (including limits like h_rt and h_vmem) are NOT applied if job is running

Eqw - Error state due to transient failures or job specification errors):

qstat -j <jobID> | grep error

qmod -cj <jobID> #from any login host

qhost

A job script “wrapper”

```
-bash:login02:~ 639 $ more  /broad/software/unixdemo/intro2uger/blast_wrapper.sh
#!/bin/bash

#  "#$" lines indicate flags for qsub
#$ -q short

source /broad/software/scripts/useuse
reuse -q BLAST

# takes rest of qsub command line and runs as a command
"$@"
```

```
qsub /broad/software/unixdemo/intro2uger/blast_wrapper.sh
blastall -p blastn -W 28 -v 10 -b 2 -d
/broad/data/blastdb/other_genomic/other_genomic -i
/broad/data/blasttest/G889P81FG1.T0.fasta -o blast.out
```

FYI, ish is also a "wrapper"...

request more than the default memory reservation:

```
ish -l h_vmem=8G
```

Submit without a job script

```
qsub -b y "source /broad/software/scripts/useuse && reuse -q
BLAST && blastall -p blastn -W 28 -v 10 -b 2 -d
/broad/data/blastdb/other_genomic/other_genomic -i
/broad/data/blasttest/G889P81FG1.T0.fasta -o blast.out"
```

may hit command line character limit

Submitter must manually capture for records (or qstat/qacct)

Advanced Topics

Task arrays — UGER has 100 job limit; each job can be a very large* task array

setting job priority

Job dependencies

X11 forwarding and UGER

UGER tips and tricks (eg. .sge_request, soft resource requests)

dotkits and login scripts

personal dotkits

*was 75,000 task limit, can be >1 million but with the available cores in the cluster, a million-task job array may only be of academic interest

IT Systems on the Broad Intranet

https://iwww.broadinstitute.org/bits

- UNIX @ Broad
  - https://iwww.broadinstitute.org/bits/linux-support
- User Education Sessions
  - https://iwww.broadinstitute.org/bits/help-offerings-trainings-and-staying-informed
- UGER
  - https://iwww.broadinstitute.org/bits/service-catalog/research-support/high-performance-computing-uger
- 'use'
  - https://iwww.broadinstitute.org/bits/dotkit
- BITS Blog:
  - https://it.broadinstitute.org/blog/

Got Questions?

help@broadinstitute.org

Office Hours:

Scientific Computing Services

Monday & Tuesday 1-2pm
75A-6143 (Roosevelt Conference room)

BITS general office hours

Tuesday & Thursday 10 - 11am
75A-6143 (Roosevelt Conference room)

We want your feedback

Help us improve this session!

http://broad.io/uger_feedback