Sun Grid Engine - A Batch System for DESY

Wolfgang Friebel,
Peter Wegner
28.8.2001
DESY Zeuthen

Introduction

- Motivations for using a batch system
  - more effective usage of available computers (e.g. more uniform load)
  - usage of resources 24h/day
  - assignment of resources according to policies (who gets how much CPU when)
  - quicker execution of tasks (system knows most powerful least loaded nodes)
- Our goal:

  You tell the batch system a script name and what you need in terms of disk space, memory, CPU time

  The batch system guarantees fastest possible turnaround
- Could even be used to get xterm windows on least loaded machines for interactive use (later)

Popular batch systems

- **Condor** targeted at using idle workstations
- **NQS** public domain and commercial versions, basic functionality
- **Loadleveler** mostly found on IBM machines, used at DESY
- **LSF** popular, rich set of features, licensed software, used at DESY
- **PBS** public domain and commercial versions, origin: NASA
  rich set of features, became more popular recently, used in H1
- **Codine/GRD** batch system similar to LSF in functionality, used at DESY
- **SGE/SGEEE** Sun Grid Engine (Enterprise Edition), open source successors of Codine/GRD. Will be the only batch system at Zeuthen (9/01)

3

Benefits using the SGEEE Batch System

- For users:
  - jobs get executed on the most suitable (least loaded, fastest) machine
  - fair scheduling according to defined sharing policies
  - no one else can overuse the system and provoke system degradation
  - users need no knowledge of host names where their jobs can run
  - quick access to load parameters of all managed hosts
- For administrators:
  - one time allocation of resources to users, projects, groups
  - no manual intervention to guarantee policies
  - reconfiguration of the running system (to adapt to changing usage pattern)
  - easy monitoring of hosts and jobs

4

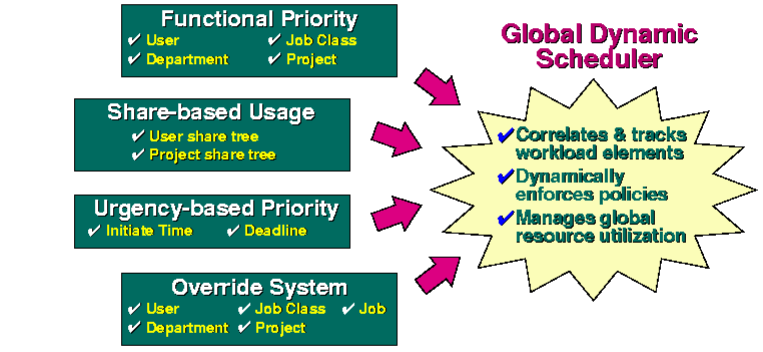
The Sun Grid Engine Batch System

Components of the system

- **Queues** contain information on number of jobs and job characteristics that are allowed on a given host. Jobs need to fit into a queue to get executed.
- **Resources** Features of hosts or queues that are known to SGE. Resource attributes are defined in so called (host queue and user defined) complexes
- **Projects** contain lists of users (usersets) that are working together. The relative importance to other projects may be defined using shares.
- **Policies** Algorithms that define, which jobs are scheduled to which queues and how the priority of running jobs has to be set. SGEEE knows functional, share based, urgency based and override policies
- **Shares** SGEEE can use a pool of tickets to determine the importance of jobs. The pool of tickets owned by a project/job etc. is called share

5

GRD Policy Capabilities



Raytheon E-Systems GENIAS Instrumental

Hosts and Users

- **Submit Host** — node that is allowed to submit jobs (qsub) and query its status
- **Exec Host** — node that is allowed to run (and submit) jobs
- **Admin Host** — node from which admin commands may be issued
- **Master Host** — node controlling all SGE activity, collecting status information, keeping access control lists etc.

A certain host can have any mixture of the roles above

- **Administrator** — user that is allowed to fully control SGE
- **Operator** — user with admin privileges, who is not allowed to change the queue configuration
- **Owner** — user that is allowed to suspend jobs in queues he owns or disable owned queues
- **User** — can manipulate only his own jobs

7

Batch Systems at Zeuthen

Present status:

- Codine
  - cell herab: beauty farm
  - cell h1: elan farm
  - cell l3: coyote farm
  - cell default: HP computers
- GRD (Global Resource Director)
  - cell default: bear, husky, ice farms

Planned configuration (9/01):

- SGEEE
  - default cell: all linux farm computers
  - cell hp: all HP computers
  - all other public linux machines become submit hosts for the default cell, further machines on request

At present: 95 Linux nodes in default SGEEE cell
17 HP nodes in cell hp

A cell is a separate pool of nodes controlled by a master node
Setting the ENV variable SGE_CELL in SGEEE selects a cell (not default!)

8

Batch Farms

- Linux Farms, current situation

ice(~50)	2 x PIII 800 MHz, 512 MByte
husky(10)	2 x PIII 600 MHz, 256 MByte
bear(4)	1 x PII 400 MHz, 128 MByte
elan(10)	2 x PIII 600 MHz, 256 MByte
beauty(12)	1 x PII 300 MHz, 128 MByte
coyote(6)	2 x PIII 450 MHz, 512 MByte

Dedicated queues (hosts) for projects amanda, h1

Common queues for projects l3, tesla, theorie, herab

9

Submitting Jobs

- Requirements for submitting jobs
  - have a valid token (verify with `tokens`), otherwise obtain a new one (`klog`)
  - ensure that in your .[t]cshrc or .zshrc no commands are executed that need a terminal (tty) (users have often a stty command in their startup scripts)
  - you are within batch if the env variable JOB_NAME is set or if the env variable ENVIRONMENT is set to BATCH
- Submitting a job
  - specify what resources you need (-l option) and what script should be executed

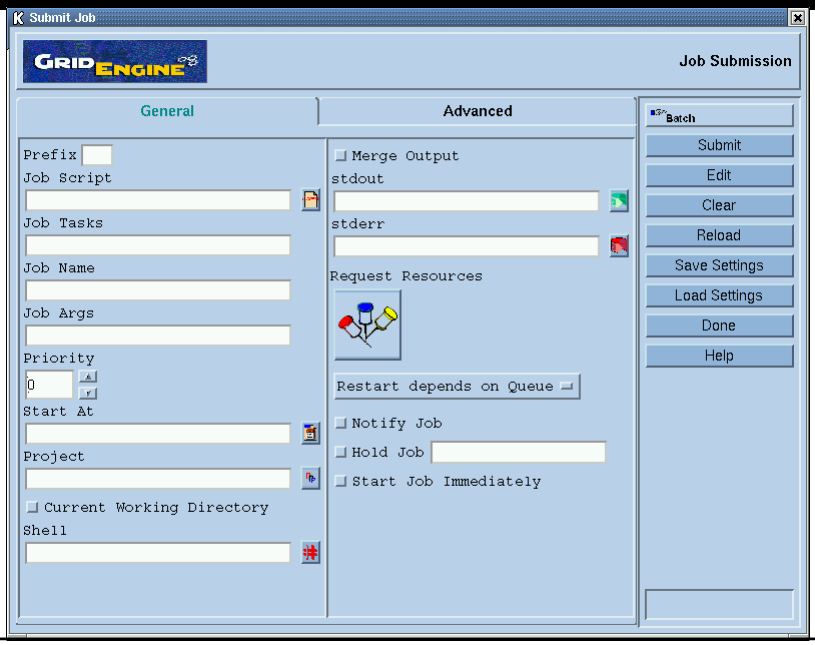
```
qsub -l t=1:00:00 job_script
```

  - in the simplest case the job script contains 1 line, the name of the executable
  - many more options availabe
  - alternatively use the graphical interface to submit jobs

```
qmon &
```

10

The Submit Window of qmon



11

Job Submission and File Systems

- Current working directory
  - the directory from where the qsub command was called. STDOUT and STDERR of a job go into files that are created in $HOME. Because of quota limits and archiving policies that is not recommended.
  - With the -cwd option to qsub the files get created in the current working directory. For performance reasons that should be on a local file system
  - If cwd is in NFS space, the batch system must not use the real mount point but be translated according to /usr/SGE/default/common/sge_aliases. As every job stores the full info from sge_aliases, we want to get rid of that file and discourage the use of NFS as current working directory
  - If required, create your own `$HOME/.sge_aliases` file
- Local file space
  - /usr1/tmp is guaranteed to exist on all linux nodes and has typically > 10GB
  - /data exists on some linux nodes and has typically > 15GB capacity. A job can request the existence of /data by -l datadir
  - $TMP[DIR] is a unique directory below /usr1/tmp, that gets erased at the end of the job. Normal jobs should make use of that mechanism if possible

12

A simple Job Script

```
#!/bin/zsh
#$ -S /bin/zsh      otherwise the default shell would be used
#
#$ -l t=0:30:00     the cpu time limit for this job (t - alias for s_cpu)
#$ -j y
WORKDIR=/usr1/tmp/$LOGNAME/$JOB_ID
DATADIR=/net/hydra/h1data7
echo using working directory $WORKDIR
mkdir -p $WORKDIR
cp $DATADIR/large_input $WORKDIR
cd $WORKDIR
h1_reco
cp large_out $DATADIR
if [ -s large_out = -s $DATADIR/large_out ]; then
  cd; rm -r $WORKDIR
fi
```

13

Advanced usage of qsub

- Option files
  - instead of giving qsub options on the command line, users may store those in `.sge_requests` files in their $HOME or current working directories
  - content of a sample `.sge_requests` file:

```
-cwd -S /usr/local/bin/perl -j y -l t=24:00:00
```

- Array jobs
  - SGE allows to schedule n identical jobs with one qsub call using the -t option:

```
qsub -t 1-10 array_job_script
```

  - within the script use the variable SGE_TASK_ID to select different inputs and write to distinct output files (SGE_TASK_ID is 1...10 in the example above)
- Conditional job execution
  - jobs can be scheduled to wait for dependent jobs to successfully finish (rc=0)
  - jobs can be submitted in hold state (needs to be released by user or operator)
  - jobs can be told not to start before a given date
  - start dependent jobs on the same host (using qalter -q $QUEUE ... within script)

14

Abnormal Job Termination

- Termination because of CPU limit exceeded
  - jobs get an XCPU signal that can be catched by the job. In that case termination procedures can be executed, before the SIGKILL signal is sent
  - SIGKILL will be sent a few minutes after XCPU was sent. It cannot be catched.
- Restart after the ececution host has crashed
  - if a host crashes when a given job is running, the job will be restarted. In that case the variable RESTARTED is set to 1
  - The job will be reexecuted from the beginning on any free host. If the job can be restarted using results achieved so far, then check for the variable RESTARTED and force the job to be executed on the same host by inserting

    ```
    qalter -q $QUEUE $JOB_ID
    ```

    in your job script
- Signalling the end of the job
  - with the qsub option -notify a SIGUSR1/SIGUSR2 signal is sent to the job one minute before the job is suspended/killed (configurable queue attribute notify)

(see: http://www-zeuthen.desy.de/www_users/rz/maillists/linux/msg00005.html)

15

Queues

- Current situation
  - on computers that did previously run CODINE and on the husky farm:

    same queues as before
  - on ice farm:

    queues hostname_timelim, where timelim is 1h, 10h, 1d, 14d (e.g. ice1_1d)
- In future
  - one queue per host with maximum time limit and low priority
  - optionally a second queue that gets suspended as soon as there are jobs in the first queue (idle queue)
  - interactive use is possible because of low priority
  - relation between jobs is respected because of sharing policies

16

Complexes

- Currently defined complexes
  - host
    - architecture (a), mem_free (mf), mem_total (mt). slots (s), s_vmem (s_vmem), h_vmem (h_vmem), s_fsize (s_fsize), h_fsize (h_fsize)
  - queue
    - qname (q), hostname(h), s_cpu (t), h_cpu (h_cpu)
  - farm
    - farm (f) - value ice is set for all queues on ice hosts
  - datadir
    - datadir - will be set for all hosts which contain /data
  - qgroup
    - group - for historical reasons

Usage:
qsub -l *complex_attribute_1[=value_1]* ... -l *complex_attribute_n[=value_n] jobscript*
e.g.
qsub -l mem_free=512M -l t=30:00:00 -P theorie *jobscript*

17

Useful SGEEE commands

- qstat - Job status

qstat -f -r (output all queues, see most everything)

```
...
---------------------------------------------------------------------------------------
ice12_10h.q        B    1/2      1.00    glinux
 43408    0 sim2000.au mkowalsk    r    08/24/2001 12:34:16 MASTER
      Full jobname:       sim2000.auto.script
      Hard Resources:   farm=ice
                          s_cpu=10:0:0
---------------------------------------------------------------------------------------
```

queue name
BCPIT (Batch/Checkpoint/Parallel/Interactive/Transfer)
used/slots total
load average
architecture
state (E=error, d=disabled a=alarmed, u=unavailable)

job number
job name
User
state(r=running,S/s/T=suspend, R=restarted,qw=queued and waiting)
submit date and time

18

Useful SGEEE commands (cont.)

- qstat - Job status (cont.)

qstat (basic output)
qstat -u user_id (show jobs for one user)
qstat -ext (show project assigned)
qstat -j (information on dropped queues)

- qdel - deletes job

qdel *jobnumber*

- qalter - change of qsub resources
- qselect - show queues which can run with specified resources

qselect -l t=20:0:0

qhold, qrls - hold and release job

19

Useful SGEEE commands (cont.)

- qhost - show status of SGEEE hosts

qhost

HOSTNAME	ARCH	NPROC	LOAD	MEMTOT	MEMUSE	SWAPTO	SWAPUS
global	-	-	-	-	-	-	-
linos.ifh.de	glinux	2	0.33	251.4M	171.4M	525.5M	11.5M
bear1.ifh.de	glinux	1	0.02	124.6M	28.1M	266.7M	10.7M
bear2.ifh.de	glinux	1	0.01	124.6M	20.1M	266.7M	632.0K
bear3.ifh.de	glinux	1	0.00	124.6M	19.0M	266.7M	1.1M
bear4.ifh.de	glinux	1	0.00	124.6M	19.8M	266.7M	720.0K
psyche.ifh.de	glinux	1	0.00	124.6M	52.7M	282.4M	14.6M
husky4.ifh.de	glinux	2	3.20	251.4M	46.0M	266.7M	1.7M
husky2.ifh.de	glinux	2	2.02	251.4M	36.4M	266.7M	2.0M
...							
ice1.ifh.de	glinux	2	0.04	504.8M	54.6M	1.0G	3.6M
ice3.ifh.de	glinux	2	1.00	504.8M	124.1M	1.0G	8.7M
ice4.ifh.de	glinux	2	0.07	504.8M	47.6M	1.0G	6.6M
...							

0

Useful SGEEE commands (cont.)

- qconf - show (-s...) or modify (-m...) SGEEE configuration

qconf -sq *queue_name* (show all queues)
qconf -sql (show queue parameters)
qconf -sprjl (show list of all projects)
qconf -scl (show complex list)
qconf -sul (show all usersets)
qconf -su *userset* (show user list)

```
qconf -su l3-user
name       l3-user
type       ACL
oticket    0
fshare     0
entries
akrueger,boos,fatima,friebel,funnell3,gruenew,gut,hebbeker,hvogt,iashvili,klabuhn,l3cos,l3dbsm,
l3mc,l3www,lcwww,leiste,nowakh,pohl,rasp,riemanns,sachwitz,schoene,schreibe,serge,shanidze,
shumeiko,sushkov,truetz,utecht,wegnerp,wlo,zchamber,ziegler
```

SGEEE log and accounting information

- SGEEE message file

/usr/SGE/default/spool/qmaster/messages

- SGEEE accounting file

/usr/GRD/default/common/accounting

- Statistics for the amanda project from the accounting file

Project amanda:
CPU time : 4 year(s) 51 week(s) 3 day(s) 21 hour(s) 51 minute(s) 17 second(s)
(total of 156981077 seconds)
SYSTEM time: 0 year(s) 2 week(s) 0 day(s) 13 hour(s) 33 minute(s) 12 second(s)
(total of 1258392 seconds)
Total number of amanda jobs : 26915

Projects

Jobs can be submitted to projects in SGEE and a project can be assigned with a level of importance via the a certain SGEEE policy (functional, override)

qconf -sprj amanda

name amanda	Project name
oticket 0	Override tickets
fshare 0	Functional shares
acl amanda-user	Userset access list
xacl NONE	Referring to Usersets being not allowed to submit jobs to the project

Current projects:

amanda, l3, herab, h1, theorie, tesla, vhdl, vhdl_low, (hermes)

23

Projects (cont.)

- Project assignment to queues, hosts (Administrator)

qconf -mqattr projects amanda ice10_1d.q allow access to queue ice10_1d.q for project amanda

qconf -mqattr xprojects l3,tesla,theorie,herab ice10_1d.q deny access to queue ice10_1d.q for the projects l3,tesla,theorie,herab

- Project definition in queue submission

qsub -l t=30:00:00 -l farm=ice -P theorie *jobscript*

-Because all queues will be assigned to projects and xprojects the definition of a project was (is) mandatory.

-SGEE : For every user a default project will be defined

24

Smooth migration to SGEEE

- CODINE & GRD expiration time

/usr/GRD/bin/glinux/grd_qmaster -show-license
Expiration time = Fri Aug 31 23:59:59 2001
...

- Setting SGEEE environment (only up to Sep 1st)

ini sge
qsub ...
...

On September, 1st SGEEE will be the one and only batch system at DESY Zeuthen

25

Advanced use of SGEEE

- Using the perl API
  - every aspect of the batch system is accessible through the perl API
  - the perl API is accessible after `use SGE;` in perl scripts
  - there is almost no documentation but a few sample scripts in /afs/ifh.de/user/f/friebel/public and in /afs/ifh.de/products/source/gridengine/source/experimental/perlgui
- Using the load information reported by SGEEE
  - each host reports a number of load values to the master host (qmaster)
  - there is a default set of load parameters that are always reported
  - further parameters can be reported by writing load sensors
  - qhost is a simple interface to display that information
  - a powerful monitoring system could be built around that feature, which is based on the "Performance Data Collection" (PDC) software from Instrumental Inc.

26