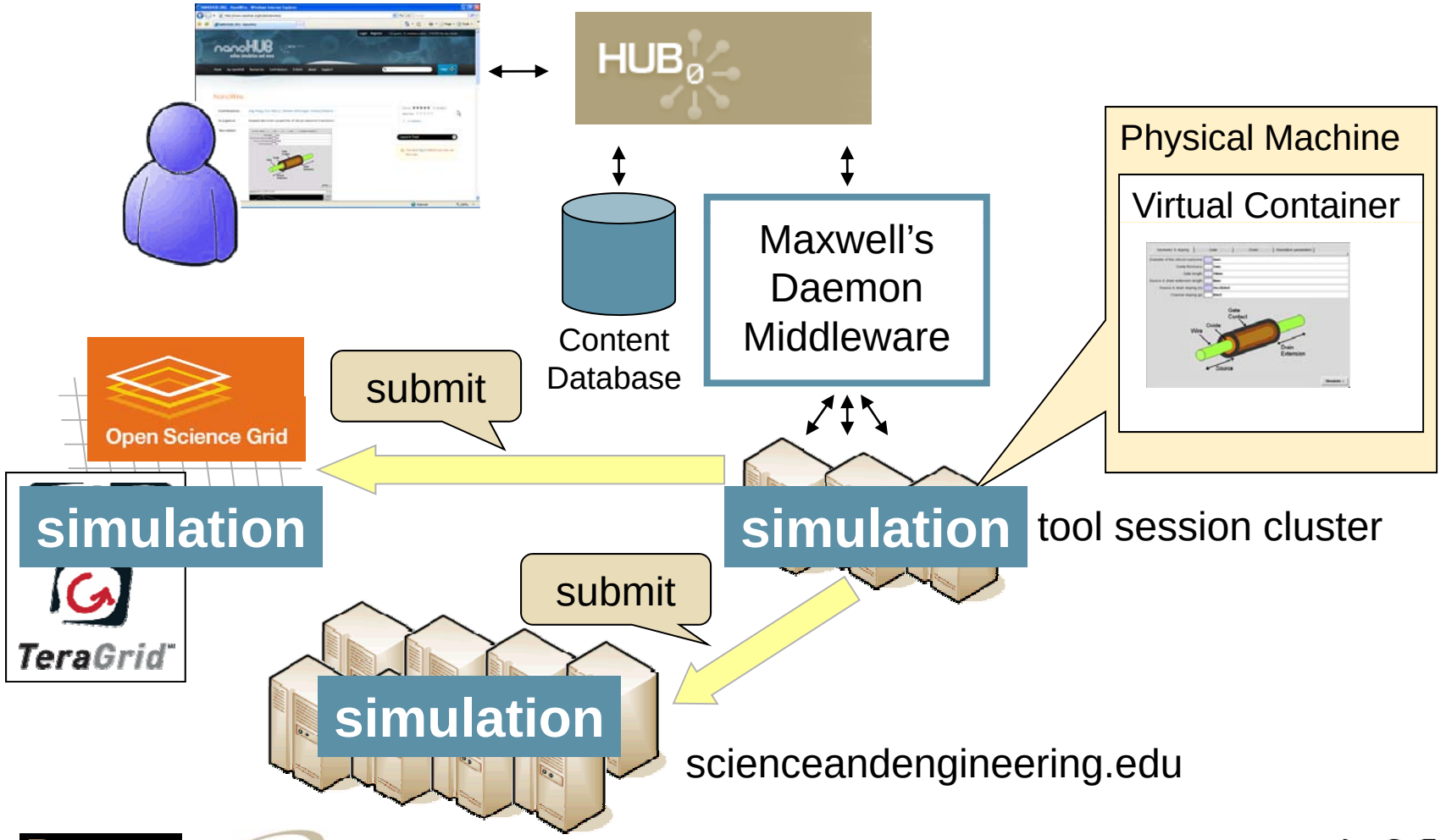
HUBzero Workshop 2010
submit Configuration

Steven Clark

Cyberinfrastructure for running tools



Resource characterization

- Homogeneous clusters
  - » Batch submission system
    - ✓ PBS
    - ✓ LSF
    - ✓ LL
    - ✓ SLURM
  - » High performance parallel computing

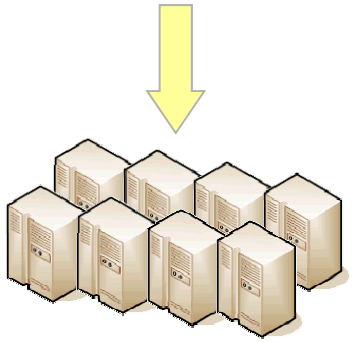

SandE.edu

- Heterogeneous collections
  - » Batch submission systems
    - ✓ Condor
  - » High throughput opportunistic computing

submit command

- Objective – make the details of remote system opaque to the end user. Command should look same as if running on local machine or desktop.

tool session



SandE.edu

```
simulator input1
```

```
submit –v SandE simulator input1
```

site

tool

submit command options

```
$ submit --help

Usage: submit [options]

Options:
     -v, --venue Remote job destination
     -i, --inputfile Input file
     -n NCPUS, --nCpus=NCPUS Number of processors for MPI execution
     -N PPN, --ppn=PPN Number of processors/node for MPI execution
     -w WALLTIME, --wallTime=WALLTIME Estimated walltime hh:mm:ss or minutes
     -e, --env Variable=value
     -m, --manager Multiprocessor job manager
     -M, --metrics Report resource usage on exit
     -W, --wait Wait for reduced job load before submission
     -h, --help Report command usage

Currently available DESTINATIONs are:
     steele

Currently available MANAGERs are:
     mpi
```

submit server configuration

```
$ cat /opt/submit/config

mysql_host = "localhost“
mysql_user = "hubzero“
mysql_password = "XXXXXXX“
mysql_db = "hubzero“

ldap_hosts = [ "hubzero.org" ]
ldap_basedn = "dc=hubzero,dc=org“
ldap_user_dn = "uid=%s,ou=users,dc=hubzero,dc=org“
listen_ports = [ "tcp://:830" ]

load_limit = 600
load_horizon = 86400
load_halflife = 3600
```

MySQL connection – record resource usage statistics

LDAP connection – user authentication

Job throttling (per user)

submit configuration

HUB

- Sites
- Tools
- Job monitors
- Managers
- Environment variables
- SSH tunnels

```
[newton]
venues = newton.SandE.edu
remoteUser = HUBzero
remoteBatchSystem = PBS
venueMechanism = ssh
remoteScratchDirectory = /scratch/HUBzero/Jobs
siteMonitorDesignator = newton
```

SandE.edu

- ssh public key
- Job monitoring
- File transfer scripts
- Batch job scripts

```
[gravity]
destinations = newton
executablePath = /apps/gravity/bin/gravity.x
```

Remote site information

- Access point
- Account
  - » User
  - » Home directory
  - » Scratch directory
- Batch job submission classification


SandE.edu

```
Session  Edit  View  Bookmarks  Settings  Help
$
$ hostname -f
newton.SandE.edu
$ echo $HOME
/home/HUBzero
$ printenv | grep SCRATCH
SCRATCH=/scratch/HUBzero
$
```

```
$ cat /opt/submit/sites.dat
[newton]
venues = newton.SandE.edu
remoteUser = HUBzero
remoteBatchSystem = PBS
venueMechanism = ssh
remoteScratchDirectory = /scratch/HUBzero/Jobs
siteMonitorDesignator = newton
```

Remote tool information

- Site
- Location


```
Session  Edit  View  Bookmarks  Settings  Help
$ cd /home/HUBzero/apps/gravity/bin
$ ls -ls
total 24224
   32 -rwx------ 1 HUBzero public    30358 Aug 24  2009 moon.x
24192 -rwx------ 1 HUBzero public 24740231 Aug 22  2009 earth.x
$
```

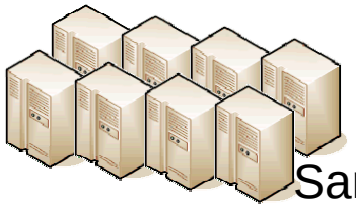
```
$ cat /opt/submit/tools.dat
[earth]
destinations = newton
executablePath = ${HOME}/apps/gravity/bin/earth.x

[moon]
destinations = newton
executablePath = ${HOME}/apps/gravity/bin/moon.x
```

Job monitor configuration

HUB

```
$ cat /opt/submit/monitors.dat
[newton]
venue = newton.SandE.edu
remoteUser = HUBzero
remoteMonitorCommand = ${HOME}/Submit/monitorPBS.py
```



SandE.edu

```
Session  Edit  View  Bookmarks  Settings  Help
$ cd ${HOME}/Submit
$ ls -lsR
total 10048
  32 drwxr-xr-x 2 HUBzero public     2048 Sep 11  2008 logs
9984 -rw-r--r-- 1 HUBzero public 10215351 Apr  8 17:36 monitorPBS.history
  32 -rwx------ 1 HUBzero public     8023 May  9  2008 monitorPBS.py
./logs:
total 1440
1440 -rw-r--r-- 1 HUBzero public 1446120 Apr  8 12:56 monitorPBS.log
```

```
$ cat /home/HUBzero/Submit/monitorPBS.py
#!/usr/bin/env python
siteDesignator     = "newton"
monitorRoot        = os.path.join(os.sep,'home','HUBzero','Submit')
qstatCommand       = "qstat -u HUBzero"
monitorLogLocation = "logs"
monitorLogFileName = "monitorPBS.log"
historyFileName    = "monitorPBS.history"
```

Multi-processor configuration


HUB

```
$ cat /opt/submit/managers.dat
[mpich]
managerCommand = mpirun -machinefile ${PBS_NODEFILE} -np NPROCESSORS

[mpich2-pgi64-mpd]
preManagerCommands = . ${MODULESHOME}/init/sh, module load mpich2-pgi64, \
                                               mpdboot -f ${PBS_NODEFILE}
managerCommand = mpirun -np NPROCESSORS
postManagerCommands = mpdallexit
mpiRankVariable = PMI_RANK

[mpich2-pgi64-manual]
preManagerCommands = . ${MODULESHOME}/init/sh, module load mpich2-pgi64, \
                        export MPD_CON_EXT=${PBS_JOBID}, \
                        mpdboot -n NNODES -f ${PBS _NODEFILE} –remcons
managerCommand = mpiexec.py -np NPROCESSORS
postManagerCommands = mpdallexit
mpiRankVariable = PMI_RANK
environment = OMP_NUM_THREADS=1
```

- Commands executed by /bin/sh
- multi-line entries for display only


SandE.edu

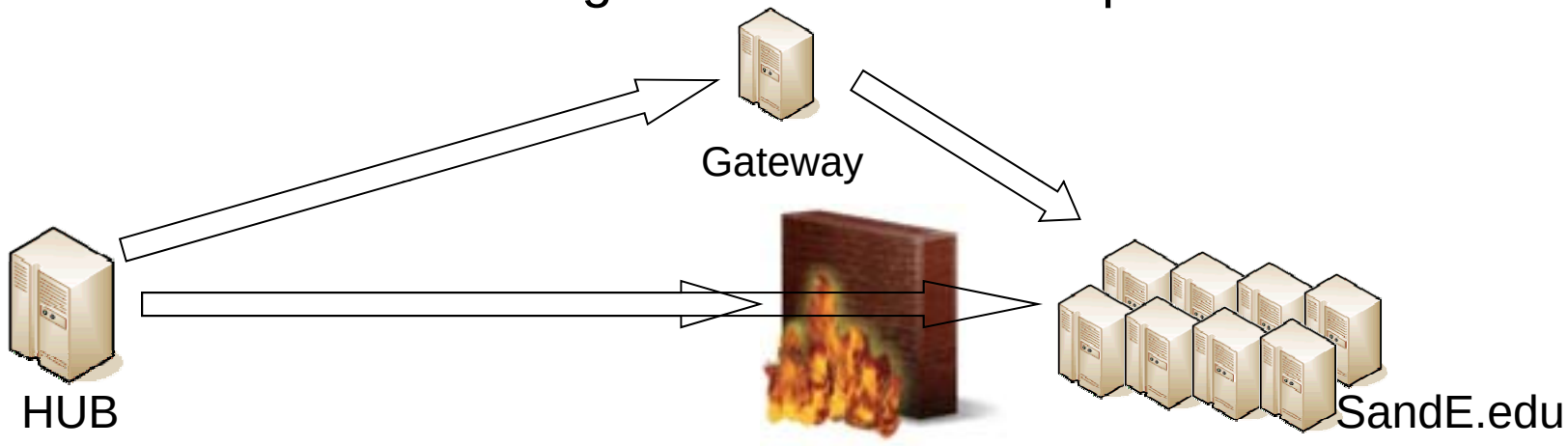
Environment variable control

- Security – prevent passage of environment that could be used maliciously on remote machine. An example of this is `LD_PRELOAD`
- Multi-processor execution often requires use of environment variables. An example of this is `OMP_NUM_THREADS`
- Tools may require environment variables to define parameters or data locations

```
$ cat /opt/submit/environmentwhitelist.dat
REBO_SPLINE_DATA
MOCA_DIR
OMP_NUM_THREADS
MPICH_HOME
```

SSH tunnel definition

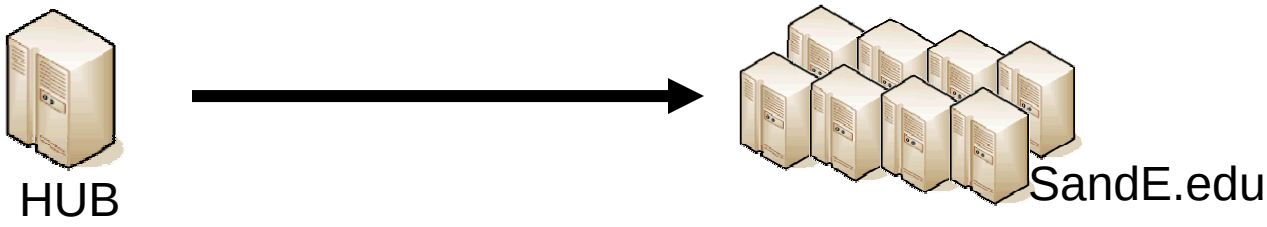
- SSH tunnels are used when required by remote sites to gain access to resources through restricted access points




```
$ cat /opt/submit/tunnels.dat
[newton]
venue = newton.SandE.edu
venuePort = 22
gatewayHost = gateway.SandE.edu
gatewayUser = HUBzero
localPortOffset = 1
```

ssh public key

- The ssh public key must be added to the authorized_keys file for the remote site user.



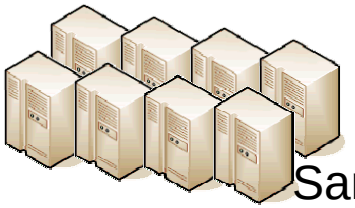

```
Session  Edit  View  Bookmarks  Settings  Help
$ cd /opt/submit
$ ls -ls .ssh
.ssh:
total 8
4 -rw------- 1 apps apps 1675 Nov 10 19:33 submit_rsa
4 -rw-r--r-- 1 apps apps  398 Nov 10 19:33 submit_rsa.pub
```

```
Session  Edit  View  Bookmarks  Settings  Help
$ cat ~/.ssh/authorized_keys
ssh-rsa AAAAB3N.....RWJHQQ== submit@hubzero.org
```

File transfer scripts

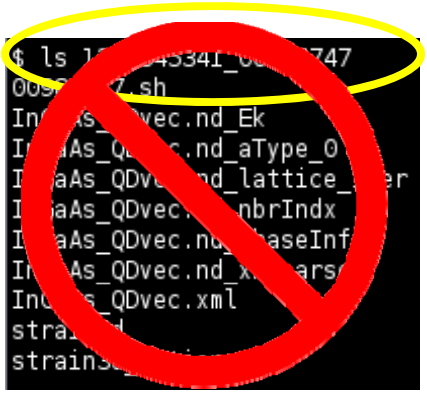
receiveinput.sh

Two arguments – job working directory
file to record completion time
Standard input – zipped tar stream



SandE.edu

HUB



transmitresults.sh

One argument – job working directory
Standard output – zipped tar stream of all files modified by job execution

cleanupjob.sh

One argument – job working directory

Batch job scripts


```
Session  Edit  View  Bookmarks  Settings  Help
$ qstat -u HUBzero

newton.SandE.edu:
                                                            Req'd  Req'd   Elap
Job ID          Username Queue    Jobname     NDS TSK  Time  S Time
--------------- -------- -------- ----------  --- ---  ----- - -----
4364798.newton HUBzero  standby  JOB_009920   32 256  04:00 Q   --
4364871.newton HUBzero  standby  JOB_009920    1   1  00:10 Q   --
4364873.newton HUBzero  standby  JOB_009920    6  48  01:00 Q   --
4364886.newton HUBzero  standby  JOB_009921    2  16  02:00 Q   --
4364887.newton HUBzero  standby  JOB_009921    6  48  01:00 Q   --
```

Q+

submitbatchjob.sh

Two required arguments – job working directory,
job description file
Optional input arguments – used to log job id and user
Standard output – job identifier

Q-

killbatchjob.sh

Two arguments – job identifier,
batch system type

* Samples are located in /opt/submit/Scripts

Access control

- Staged tools – located on remote sites
  - » Apply user based restriction to either tool or site by use of `restrictedToUsers` keyword in `tools.dat` and `sites.dat`
  - » Apply group based restriction to either tool or site by use of `restrictedToGroups` keywords in `tools.dat` and `sites.dat`
- Published applications – located in /apps/…
- Tools under development – located in /home/…
  - » Requires membership in group submit
  - » Permission from site – submission can be declined by setting `arbitraryExecutableAllowed = False` in `sites.dat`

* Tool location is based on the submit command arguments

Online documentation

https://hubzero.org/documentation/0.9.0/installation/Setup.submit

submit lab exercise

- Objective – configure submit to enable execution of sample programs on provided remote resource. You will be provided with a HUB and a matching account on a remote resource.
- Application – compute pi
  - » A single input file must be provided
    - ✓ Each line in the file contains a single parameter that determines the accuracy of the calculation
    - ✓ The file name should be given as the single command line argument
  - » Serial version of the program is located at `${HOME}/apps/pi/bin/pi_seq`
  - » Parallel version of the program is located at `${HOME}/apps/pi/bin/pi` and should be run with the command `mpirun –machinefile ${PBS_NODEFILE} –np NPROCESSORS`

submit lab exercise

- A site has been set up as `pbs-test.hubzero.org` running PBS. It is suggested that the number of cores used for program execution be kept to a limit of 8. Connection between your HUB and the remote should be via ssh with probe checking turned off. For purposes of this exercise the scratch directory has been located in the home directory as `${HOME}/scratch/hubbub2010`
- The monitoring script has been setup in `${HOME}/Submit/monitorPBS.py`

submit lab exercise

- To configure the various aspects of submit edit files located in `/opt/submit` as the user `apps`.
- The submit server daemon and the main job monitoring will be running on your HUB's. The job monitoring daemon should be stopped and started after you edit the file `monitors.dat`. This should be done with the commands
  - `sudo /etc/init.d/submon stop`
  - `sudo /etc/init.d/submon start`
- Various log files are located in `/var/log/submit`
- You should use the workspace application to run tests of your `submit` configuration.