Max-Planck-Institut für Polymerforschung
Max Planck Institute for Polymer Research

Retired: The GE Cluster

Robert Klein*

February 24, 2025

The GE cluster has been retired!

Introduction

- Operating System:
  - openSUSE 11.2
  - Debian Linux 9
- 160 Dual QuadCore Xeon, 16 Dual Hexa Core Xeon, 120 Dual Octa-Core Xeon, 88 dual DecaCore Xeon

Preparation before using the cluster

The cluster software uses ssh to execute your jobs on the compute nodes. You will have to set up a private/public key pair to be able to submit jobs to the cluster.

If you currently have to enter your password when you ssh to another Linux computer in the group, execute the following commands to create your public/private key pair.

```
# first, create an rsa key
ssh-keygen -t rsa

# copy your public key  to the public keys file
ssh-copy-id -i ~/.ssh/id_rsa.pub localhost
```

*Robert.Klein@mpip-mainz.mpg.de

Compiling for the cluster

```
# initialize Intel Compiler Suite
source /sw/linux/intel/XE15u3/parallel_studio_xe_2015/psxevars.sh
export PATH="/sw/linux/mpi/intel/openmpi/bin/:$PATH"
export LD_LIBRARY_PATH="/sw/linux/mpi/intel/openmpi/lib/:$LD_LIBRARY_PATH"
# I'm not sure you really need to set the following environment variables.
export CC="icc";
export F77="ifort";
export CXX="icpc";
export MPIEXEC="/sw/linux/mpi/intel/openmpi/bin/mpirun"
```

Keep in mind

- You can submit jobs from any of our Linux workstation.
- Write the result data to /usr/scratch. This directory is not backed up. Please remember to copy the result of your finished jobs away (e.g. using the `rsync` command).

Best Practices

- jobs should write to `/usr/scratch/$LOGNAME`.
- copy the output to your personal disk space after the job has finished.

Cluster Queues

Name	Cores / Threads	Maximum CPU time	Parallel Environment	Who
Test.q	1..20	15 min	`PE_20_T`	all users
Parallel20.q	n*20 $\leq$ 700	36 h	`PE_Multi20_IB`	all users
Standard20.q, Parallel20.p, Debian20.q	20	36 h	`PE_20`	all users
Parallel16.q	n*16 $\leq$ 432	36 h	`PE_Multi16_IB`	
Standard16.q, Parallel16.p	16	36 h	`PE_16`	all users
Parallel.q	n*8 $\leq$ 216	36 h	`PE_Multi_IB`	all users
Standard8.q, Parallel.q	8	36 h	`PE_8`	all users
Base12.q	n*12	36 h	`PE_B12`	all users
Legacy8.q	8	72 h	`PE_8_Legacy`	all users
Single.q	1	72 h		all users
SingleLong.q	1	350 h		register
NMR64.q	1-64	unlimited	`NMR64`	NMR group
Weidner16.q	n*16 $\leq$ 128	unlimited	`PE_Weidner16`	Weidner group

Single core queue systems have at least 2 GB of RAM and at least 2 cores.

8-core queue systems have at least 12 GB of RAM and at least 8 cores.

Note, the Legacy8.q will be removed in the future; only the old nodes thop63 up to thop102 are in this queue.

Important commands

qsub submit a job
qstat check jobs and queues
qdel cancel a job

example command usage

qsub -w v "jobscript" test your submit script for shell syntax and availability of requested resources.
qsub [-pe <parallel environment>] "jobscript"
qalter -w -v "jobid" returns extented list of errors, including resource-related problems which are omitted otherwise.
qstat -j "jobid" returns all parameters for a submitted job: check for reason for not being scheduled.
qstat -explain E -j "jobid" returns reason, why a submitted job is in error state "Eqw".
qstat -cj "jobid" clears the error state of the specified job.

Submit Script examples

Simple Grid Engine script

```
#!/bin/sh
a.out
```

more complex Grid Engine script

```
#!/bin/sh

# to run I want 8 parallel processes under the PE openmpi:
#$ -pe openmpi 8

# Send mail to me at beginning/end/on abort:
# (Put your username in place of 'username'.)
#$ -m bea -M username@mpip-mainz.mpg.de
```

```
# The job is located in the current working directory:
#$ -cwd

/sw/linux/mpi/gcc/openmpi/bin/mpirun -np 8 a.out
```

More about submit scripts

A grid engine submit script is a shell script. The script can have embedded options for the qsub command. All those options are on lines beginning with #$.

-M specify an email address to send mails to.
-m when to send mails:

- b send mail at beginning of job
- e send mail at end of job
- s send mail on suspension of job

-cwd The job is located in the current working directory

SGE tips

Building Software with cmake

- parallelization for CMAKE is set with *CMAKE_BUILD_PARALLEL_LEVEL*. You can set this to *$NSLOTS* in the SGE script which is the number of cores assigned to the SGE job.

SLURM

- The shell in the first line of your job script should be the same as your login shell. If you use any other shell, your job will be limited to the interactive time limit (15 min).
- The system's openmpi is not compiled with SLURM support, so you cant start jobs using "srun". (use "mpirun" instead
- If you're using/initializing the intel compiler, your job script must be a bash script. First line should be:

```
#!/bin/bash
```