

Genetic Cluster Computer



The Genetic Cluster Computer (GCC) is used by researchers in genetic epidemiology, molecular genetics, statistical genetics or behavioral genetics. It consists of 128 processors (3.4 GHz) on 64 nodes. The GCC is financed by an NWO Medium Investment grant (480-05-003 PI Posthuma) and by the Faculty of Psychology and Education of the Free University, Amsterdam, The Netherlands. It is hosted by the Dutch National Computing and Networking Services (SARA).

Getting started

The GCC is a separate part of the LISA cluster at the Dutch National Computing and Networking Services (SARA) and works in a similar way, except that the queue name is *qq-genetics*. For more information on working with the batch queuing system (BQS) on LISA, see below for a short introduction or check:

<http://www.sara.nl/userinfo/lisa/usage/index.html>

- *How to obtain a login-name?*
Send an e-mail to danielle@psy.vu.nl, (include full address details)
You will login to the interactive node where you can upload files and scripts and tryout short runs. From here you need to submit jobs to the jobscheduler who will then distribute your scripts to the nodes.
- *How do I login to the system?*
Use SSH2 (e.g. putty) to `lisa.sara.nl`. For FTP use secureFTP (e.g. winscp or Filezilla)
- *What software is installed?*
Genehunter - MeV - Mendel - Mx - R - SimWalk - Merlin - QTDT - Solar - Twinsim - Allegro - GASP - GASSOC - LOKI - Mapmaker - MEGA2 - PEDIG - SAGE - TRANSMIT - UNPHASED - Vitesse
You may need to type `$ module load vitesse`, once [if you want to use e.g. vitesse]
Please direct requests for other software to dr D Posthuma at: danielle@psy.vu.nl.
Software can also locally be installed.

Submitting one serial analysis to the cluster

You need:

1. Your normal script for data analysis (i.e. an R script, shell script with commands for promptline software, or mx script) + datafile
2. BQS submission job <job>

An example <job> looks as follows:

```
#PBS -qq_genetics
#PBS -lnodes=1
#PBS -lwalltime=1:00:00
cd $HOME/MyAnalysis || exit
mx script.mx script.mxo
```

This will ask the cluster to run one script.mx on one node in the genetics queue. You need to say the maximum time the analysis will run, and need to specify the directory where the script is.

Using the command

```
> qsub job submits 'job'
```

The command

```
> qstat -u [username] will show the status of all jobs, including job-id  
you have submitted
```

The command

```
> qdel [jobid] deletes a job from the queue
```

After analysis, the output appears in your working directory on the interactive node.

Do I need extensive knowledge of UNIX or programming languages?

No, fortunately not. Extensive knowledge of UNIX/LINUX commands is not needed and lack of it should not obstruct your access to super computing power. A short manual to

UNIX commands can e.g. be found here <http://www.ee.surrey.ac.uk/Teaching/Unix/>. Also, the SARA helpdesk is always available by e-mail hic@sara.nl.

Using two processors per node

Every node has two processors. If you ask for one node, you occupy two processors, even though most software is not parallel. It is therefore more efficient to start two processes per node, using the following example job:

```
#PBS -qq_genetics
#PBS -lnodes=1
#PBS -lwalltime=1:00:00
cd $HOME/MyAnalysis || exit
mx script1.mx script1.mxo &
cd $HOME/MyAnalysis || exit
mx script2.mx script2.mxo &
```

Using the command

```
> qsub job submits 'job'
```

The command

```
> qstat -u [username] will show the status of all jobs, including job-id
you have submitted
```

The command

```
> qdel [jobid] deletes a job from the queue
```

Running multiple serial analyses

The cluster is especially equipped to run multiple serial analyses, e.g. 100, 500, or 1000. You do not want to write 100, 500 or 1000 submission jobs, and therefore need another short script that will generate submission jobs for you.

You need 3 documents:

1. Your normal **script** for data analysis (i.e. an R script, shell script or mx script) + datafile if needed. The script needs to include some parameters that need to be changed for every analysis. In this example we use an R script called sjabloon.R, with xxnpermaxx and xxncaxx in the script that need to be replaced by parameters 2 and 3 that are generated by the shell script
2. **Shell script** that loops through your intended series of analysis <runloop>.
Example:

```
#!/bin/bash
for nrep in 5000 ; do
  for nperma in 100 500 1000 ; do
    for nca in 100 200 500 1000 5000 ; do
      echo $nrep $nperma $nca
    done
  done
done | \
while read nrep1 nperma1 nca1 ; do
  read nrep2 nperma2 nca2
  ./subjobg $nrep1 $nperma1 $nca1 $nrep2 $nperma2 $nca2
done
```

3. **Submission-job generating script** <subjob>. Example:

```
#!/bin/bash
cat <<ej > tmpjob
#PBS -qq_genetics
#PBS -lwalltime=60:00:00
ej
while [ "$1" ]; do
echo creating partial job for $1 $2 $3
cat <<ej >> tmpjob
cd \${HOME}/RPERMS_G || exit
mkdir -p d$1.$2.$3
cd d$1.$2.$3 || exit
cat ../sjabloonG.R | sed 's/xxnpermaxx/$2;/s/xxncaxx/$3/' > finalG.$1.$2.$3
R CMD BATCH finalG.$1.$2.$3 &
ej
shift 3
done
echo wait >> tmpjob
echo submitting job
qsub tmpjob
```

The command

```
> ./runloop &
```

generates a series of submission jobs that each submit a unique R script, using the parameters specified in the shell script from 2.

Example session

Example session: running hundreds of analyses using the program R

We have

- an R script (sjabloon.R) (ascii format)
- runloop (ascii format)
- subjob (ascii format)

The three files above can be made with notepad.

1. Open Filezilla (or any other secure FTP program)
2. Upload the three files, using ASCII transfer mode
3. Open secure FTP program (e.g. Putty)
4. Log on to lisa.sara.nl
5. Go the directory where your files are (e.g. > cd myfiles)
6. Make runloop and subjob executable by typing

```
> chmod +x runloop
> chmod +x subjob
```
7. Call runloop by typing

```
> ./runloop &
```
- (8. Type

```
> qstat -u [username]
```

to check the status of your jobs)

Your jobs will now be submitted. Wait until they are finished and find the output in your working directory.

Questions or remarks: SARA helpdesk hic@sara.nl or D Posthuma danielle@psy.vu.nl