Contents

- System Description
- Accessing Resources
- Working Areas
- Submitting Jobs
- Compiling and working on the nodes
- User workflow at a glance
The system is composed of 18 computing nodes, configured as a unique Cluster and one management node is centralizing all the services. With this setup, the batch system (Slurm) is in charge of deciding which computing node will execute the jobs, depending on the demanded resources. Therefore, jobs can be submitted from the login node, and will be dispatched at the proper computing node as soon as resources will be available.

The system shares a parallel file system based in BeeGFS software.

**Compute nodes:**
- 18 nodes - node[001-018]
- 2 x Processors 2650 v4
- 96GB memory

**Accessing Resources**

Accessing to the system can be achieved by using ssh protocol, to the login node:

```
ssh username@hpc.s.upf.edu
```

For security reasons the connection can only be established using VPN connection.

If you are experiencing trouble logging into the system or in case you do not remember the password, please contact support staff.

In order to transfer files to the systems (and not from it, as outgoing ssh connections are not allowed) you can use either scp or sftp command:

```
scp local_file username@hpc.s.upf.edu
sftp username@hpc.s.upf.edu
```

If you work in a Windows machine you can work with Putty or MobaXterm.
Working Areas

The working areas available at UPF-DTIC facilities are summarized below:

<table>
<thead>
<tr>
<th>Path</th>
<th>Size</th>
<th>Availability</th>
<th>Quota</th>
<th>Backup</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>/homedtic</td>
<td>376TB</td>
<td>Global</td>
<td>NO</td>
<td>NO</td>
<td>HPC</td>
</tr>
<tr>
<td>/datasets</td>
<td>376TB</td>
<td>Global</td>
<td>NO</td>
<td>NO</td>
<td>HPC</td>
</tr>
<tr>
<td>/tmp</td>
<td>240GB</td>
<td>Local</td>
<td>NO</td>
<td>NO</td>
<td>HPC</td>
</tr>
</tbody>
</table>

/homedtic & /datasets area is globally available shared by all the machines by sNow! server.

/tmp is the local disc of the nodes and must be used for I/O intensive jobs.
Submitting Jobs

In order to improve the scheduling and reduce job’s waiting time, a structure of queues has been defined, used in the framework of the Slurm batch manager. The following table summarizes all the queues features, with the cpu time, number of cores, memory and disk limits assigned to each of them.

<table>
<thead>
<tr>
<th>Partition</th>
<th>Walltime(h)</th>
<th>Priority</th>
<th>Job Preemption</th>
</tr>
</thead>
<tbody>
<tr>
<td>short</td>
<td>2:00:00</td>
<td>100</td>
<td>none</td>
</tr>
<tr>
<td>medium</td>
<td>8:00:00</td>
<td>75</td>
<td>none</td>
</tr>
<tr>
<td>high</td>
<td>infinite</td>
<td>50</td>
<td>none</td>
</tr>
</tbody>
</table>

Running a batch job:

❖ Build the batch script example.sl
❖ Send the job to the batch queue
  ➢ sbatch example.sl
❖ Monitor the execution
  ➢ squeue
❖ Controlling jobs
  ➢ scontrol
❖ Cancel job
  ➢ scancel jobid

Most commun Slurm commands:

● sbatch - submits a job script
● scancel - cancels a running or pending job
● srun - runs a command across nodes
● sbcast - transfers file(s) to the compute nodes allocated for the job
● interactive - opens an interactive job session
● sattach - connect stdin/out/err for an existing job or job step
● squeue - displays the job queue
● sq - squeue wrapper which displays all the jobs
● squ - squeue wrapper which only displays your jobs
Compiling and Working on the nodes

In order to work interactively in the job you must use the slurm system (users will not allowed to enter via ssh to the node).

Most of the libraries and applications useful environment can be loaded using lmod module system.

```
module avail
module load whatever
```

The `interactive` command is the easiest way to submit and run an interactive job. Once the job starts, the user is automatically logged into the compute node. All variables set by SLURM are available on the compute node.

Example: if you want to compile a program using 2 cores you should run the command:

```
interactive -c 2
```

More info:

```
interactive -help
```

User workflow at a glance:

**Login**

```
ssh username@hpc.s.upf.edu -X
```

**Interactive job session**

```
interactive
```

**Check software available in the cluster**

```
ml av
ml spider whatever
```

**Load a software package and its dependencies**

```
ml Python/2.7.12-foss-2017a
```
Example

[username@node002 ~]$ cp /datasets/.../python.sl /home/username
[username@node002 ~]$ cat python.sl
#!/bin/bash
#SBATCH -J PYTHON
#SBATCH --time=00:15:00
#SBATCH --ntasks=32
#SBATCH --mem-per-cpu=4G
ml Python/2.7.12-foss-2017a
script.py

Submit the job

[username@node002 ~]$ sbatch python.sl
Submitted batch job 15
[username@node002 ~]$ squeue

Review the results

tail -f slurm-15.out