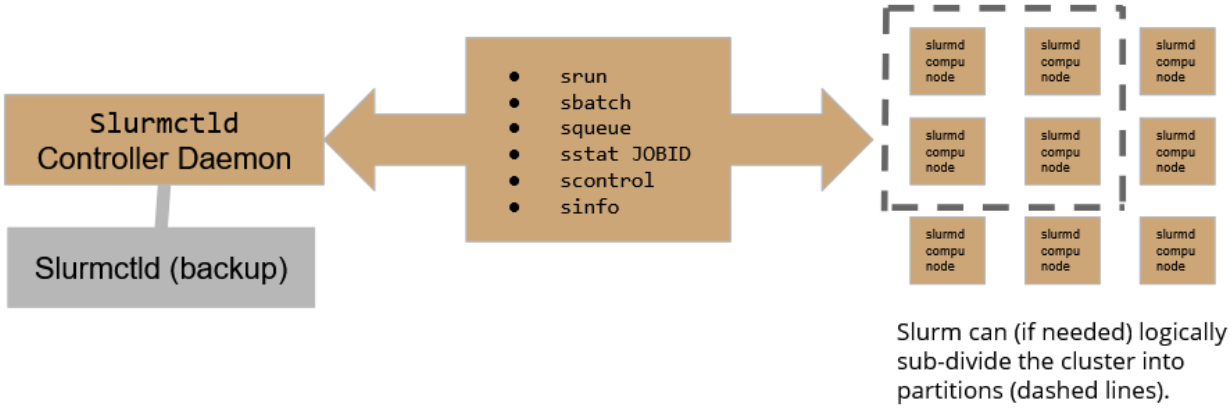
SLURM architecture

Slurm workload manager relies on the following scheme:



Where the Slurmctld daemon plays the role of the controller, allowing the user to submit and follow the execution of a job, while Slurmd daemons are the active part in the execution of jobs over the cluster. To assure high availability, A backup controller daemon has been configured to assure the continuity of service.

On our HPC cluster, there are currently 4 active partitions:

1. **slurmHPC_int** MaxTime allowed for computation = 79h
2. **slurmHPC_inf** MaxTime allowed for computation = 79h
3. **slurmHPC_short** MaxTime allowed for computation = 79h
4. **slurm_GPU** MaxTime allowed for computation = 33h
5. **slurm_hpc_gpuV100** MaxTime allowed for computation = 33h

Please be aware that exceeding the MaxTime enforced will result in the job being held.

If not requested differently at submit time, jobs will be submitted to the _int partition. Users can choose freely what partition to use by configuring properly the batch submit file (see below).

Check the cluster status with SLURM

You can check the cluster status using the **sinfo -N** command which will print a summary table on the standard output.

The table shows 4 columns: NODELIST, NODES, PARTITION and STATE.

1. **NODELIST** shows node names. Multiple occurrences are allowed since a node can belong to more than one partition
2. **NODES** indicates the number of machines available.
3. **PARTITION** which in slurm is a synonym of "queue" indicates to which partition the node belongs. If a partition name comes with an ending asterisk, it means that that partition will be considered the default one to run the job, if not otherwise specified.
4. **STATE** indicates if the node is not running jobs ("idle"), if it is in drain state ("drain") or if it is running some jobs ("allocated").

For instance:

```
-bash-4.2$ sinfo -N
NODELIST       NODES      PARTITION STATE
hpc-200-06-05      1  slurmHPC_int* idle
hpc-200-06-05      1 slurmHPC_short idle
hpc-200-06-05      1   slurmHPC_inf idle
hpc-200-06-05      1      slurm_GPU idle
hpc-200-06-06      1 slurmHPC_short idle
hpc-200-06-06      1  slurmHPC_int* idle
hpc-200-06-06      1   slurmHPC_inf idle
[...]
```