Slurm User Cheat Sheet

by Guillermo (guilleaf) via cheatography.com/19177/cs/42873/

Job Options

Account to be charged for resources used

```
-A, --acco unt =<a cco unt>
```

Job array specification (sbatch only)

```
-a, --arra y=< ind ex>
```

Initiate job after specified time

```
-b, --begi n=< tim e>
```

Required node features

```
-C, --cons tra int =<f eat ure s>
```

Bind tasks to specific CPUs (srun only)

```
--cpu-bind=<type>
```

Number of CPUs required per task

```
-c, --cpus -pe r-t ask =<c oun t>
```

Defer job until specified jobs reach specified state

```
-d, --depe nde ncy =<s tat e:j obi d
>
```

Specify distribution methods for remote processes

```
-m, --dist rib uti on= <me tho d[: m
et hod ]>
```

File in which to store job error messages (sbatch and srun only)

```
-e, --erro r=< fil ena me>
```

Specify host names to exclude from job allocation

```
-x, --excl ude =<n ame>
```

Reserve all CPUs and GPUs on allocated nodes

```
--excl usive
```

Export specified environment variables (e.g., all, none)

```
--expo rt= <na me= val ue>
```

Number of GPUs required per task

```
--gpus -pe r-t ask =<l ist>
```

Job name

```
-J, --job- nam e=< nam e>
```

Prepend task ID to output (srun only)

```
-l, --label
```

Job Options (cont)

E-mail notification type (e.g., begin, end, fail, requeue, all)

```
--mail -ty pe= <ty pe>
```

E-mail address

```
--mail -us er= <ad dre ss>
```

Memory required per allocated node (e.g., 16GB)

```
--mem= <si ze> [units]
```

Memory required per allocated CPU (e.g., 2GB)

```
--mem- per -cp u=< siz e>[ uni
ts]
```

Specify host names to include in job allocation

```
-w, --node lis t=< hos tna mes
>
```

Number of nodes required for the job

```
-N, --node s=< cou nt>
```

Number of tasks to be launched

```
-n, --ntas ks= <co unt>
```

Number of tasks to be launched per node

```
--ntas ks- per -no de= <co unt
>
```

File in which to store job output (sbatch and srun only)

```
-o, --outp ut= <fi len ame>
```

Partition in which to run the job

```
-p, --part iti on= <na mes>
```

Signal job when approaching time limit

```
--sign al= [B: ]<n um> [@time]
```

Limit for job run time

```
-t, --time =<t ime>
```

The options can be used on the command line or in the case of a submission script they must be preceed by `#SBATCH`

Job submission

`sbatch`	Submit a batch script
`salloc`	Request allocation for interactive job
`srun`	Request allocation and run an application

sbatch and salloc examples

```
# Request interactive job on
debug node with 4 CPUs
salloc -p standby -c 4
# Request intera ctive job with
V100 GPU
salloc -p comm_g pu_ inter --
ntasks=1 --gpus=3
# Submit batch job
sbatch runjob.slurm
```

sprio options

Output format to display

```
-o, --form at= <op tio ns>
```

Filter by job IDs (csl)

```
`-j, --jobs=<job_id_list>`
```

Show more available information

```
-l, --long
```

Show the normalized priority factors

```
-n, --norm
```

Filter by partitions (csl)

```
-p, --part iti on= <pa rti tio n
_l ist>
```

Filter by users (csl)

```
-u, --user =<u ser _li st>
```

csl = comma-separated list

By **Guillermo** (guilleaf)
cheatography.com/guilleaf/
bit.ly/guilleaf

Not published yet.
Last updated 28th March, 2024.

Cheatography

Slurm User Cheat Sheet

by Guillermo (guilleaf) via cheatography.com/19177/cs/42873/

sprio examples

```
# View normalized job priorities for your own jobs
sprio -nu $USER
# View normalized job priorities for specified partition
sprio -nlp standby
```

scancel examples

```
# Cancel specific job
scancel 314159
# Cancel all your own jobs
scancel -u $USER
# Cancel your own jobs on specified partition
scancel -u $USER -p standby
# Cancel your own jobs in specified state
scancel -u $USER -t pending
```

scancel options

Restrict to the specified account

```
-A, --acco unt =<a cco unt>
```

Restrict to jobs with specified name

```
-n, --name =<j ob_ nam e>
```

Restrict to jobs using the specified host names (csl)

```
-w, --node lis t=< hos tna mes >
```

Restrict to the specified partition

```
-p, --part iti on= <pa rti tio n>
```

Restrict to the specified user

```
-u, --user =<u ser nam e>
```

csl = comma-separated list

squeue examples

```
# View your own job queue with estimated start times
squeue --me
# View own job queue with estimated start times for pending jobs
squeue --me --start
# View job queue on specified partition in long format
squeue -lp epyc-64
```

squeue options

Filter by accounts (csl)

`-A, --account=<account_list>`

Output format to display

`-o, --format=<options>`

Filter by job IDs (csl)

`-j, --jobs=<job_id_list>`

Show more available information

`-l, --long`

Filter by your own jobs

`--me`

Filter by job names (csl)

`-n, --name=<job_name_list>`

Filter by partitions (csl)

`-p, --partition=<partition_list>`

Sort jobs by priority

`-P, --priority`

Show the expected start time and resources to be allocated for pending jobs

`--start`

Filter by states (csl)

`-t, --states=<state_list>`

Filter by users (csl)

`-u, --user=<user_list>`

csl = comma-separated list

Job Management

`squeue`	View information about jobs in queue
`scancel`	Signal or cancel jobs, job arrays, or job steps
`sprio`	View job scheduling priorities

Partition and node information

`sinfo`	View information about nodes and partitions
`scontrol`	View or modify configuration and state

sinfo options

Output format to display

```
-o, --form at= <op tio ns>
```

Show more available information

```
-l, --long
```

Show information in a node-oriented format

```
-N, --Node
```

Filter by host names (comma-separated list)

```
-n, --node s=< hos tna mes>
```

Filter by partitions (comma-separated list)

```
-p, --part iti on= <pa rti tio n _l ist>
```

Filter by node states (comma-separated list)

```
-t, --stat es= <st ate _li st>
```

Show summary information

```
-s, --summ arize
```

sinfo examples

```
# View all partitions and nodes by state
sinfo
# Summarize node states by partition
sinfo -s
# View nodes in idle state
sinfo --stat es=idle
```

Slurm User Cheat Sheet

by Guillermo (guilleaf) via cheatography.com/19177/cs/42873/

sinfo examples (cont)

> # View nodes for specified partition in long, node-oriented format
sinfo -lNp standby

scontrol actions and options

Show more details

```
-d, --details
```

Show information on one line

```
-o, --oneliner
```

Show partition

```
show partition <pa rti tio n>
```

Show node

```
show node <ho stn ame>
```

Show job

```
show job <jo b_i d>
```

Hold Jobs

```
hold <jo b_l ist>
```

Release Jobs

```
release <jo b_l ist>
```

Show Hostnames

```
show hostnames
```

scontrol examples

```
# View information for specified partition
scontrol show partition standby
# View inform ation for specified node
scontrol show node tcocs002
# View detailed inform ation for running job
scontrol show job 314159 -d
# View hostnames for job (one name per line)
scontrol show hostnames
```

Slurm environment variables

Number of tasks in job array

```
SLURM_ ARR AY_ TAS K_COUNT
```

Job array task ID

```
SLURM_ ARR AY_ TASK_ID
```

Number of CPUs requested per task

```
SLURM_ CPU S_P ER_TASK
```

Account used for job

```
SLURM_ JOB _AC COUNT
```

Job ID

```
SLURM_ JOB_ID
```

Job Name

```
SLURM_ JOB _NAME
```

List of nodes allocated to job

```
SLURM_ JOB _NO DELIST
```

Number of nodes allocated to job

```
SLURM_ JOB _NU M_NODES
```

Partition used for job

```
SLURM_ JOB _PA RTITION
```

Number of job tasks

```
SLURM_ NTASKS
```

MPI rank of current process

```
SLURM_ PROCID
```

Directory from which job was submitted

```
SLURM_ SUB MIT_DIR
```

Number of job tasks per node

```
SLURM_ TAS KS_ PER _NODE
```

Slurm Environment Variables examples

```
# Specify OpenMP threads
export OMP_NU M_T HRE ADS =$S -
LUR M_C PUS _PE R_TASK
# Specify MPI tasks
srun -n $SLURM _NTASKS ./mpi_ -
program
```

By **Guillermo** (guilleaf)
cheatography.com/guilleaf/
bit.ly/guilleaf

Not published yet.
Last updated 28th March, 2024.