

HPCC [Preview]

Center for High-Performance Computing

Handy Slurm Commands

Overview

- [scancel – cancel jobs](#)
- [squeue – queuing information](#)
- [sstat – system utilization information \(for running jobs\)](#)
- [sinfo – general information \(by partition\) \(by node characteristic\) \(node counts\)](#)
- [scontrol – configuration information \(licenses\)](#)
- [sacct – resource accounting information \(for finished and running jobs\)](#)
- [sacctmgr – account manager](#)

The examples below show real output. The results will differ when you run the commands. Use **–help** for options for the command.

SCANCEL: Cancel Jobs

The Slurm command [scancel](#) is used to cancel jobs. You can cancel one job, all pending jobs, or all waiting jobs in a partition. Type **man scancel** for details.

```
scancel <job_id>
scancel --user=$USER --state=pending
scancel -u $USER --partition=<partition>
```

SQUEUE: Queuing Information

The Slurm command [squeue](#) is used to view information about queued jobs, per user or per partition. Type **man squeue** for details.

```
squeue -u $USER
squeue --start -u $USER    #include start time
squeue -p <partition>    #try "squeue -p main"
```

SSTAT: Utilization Stats while Running Jobs

The Slurm command [sstat](#) is used to view status information about running jobs. See **sstat -e** for a list of available output fields. Type **man sstat** for details.

```
sstat -j <job_id>
sstat -p -j <job_id> --format=AveCPU,AveRSS,AveVMSize,MaxRSS,MaxVMSize
```

SINFO: Job and Node Information

The Slurm command [sinfo](#) is used to view information about partitions and nodes. You can configure the output by specifying command and format (**-o**) options. Type **man sinfo** for details.

Partition (Queue)

Show all nodes in the general computing partitions — main, large, quick and long — and their attributes.

```
$ sinfo --partition=main,large,quick,long -o "%50N %4c %8m %40f %12G" | uniq
NODELIST                                CPUS  MEMORY  AVAIL_FEATURES                                GRES
hpc[4433-4520]                          20    59392   IB,avx,avx2,xeon,E5-2640v4,r430              (null)
hpc[4570-4573,4623-4632,4657-4674]      32    188888  IB,avx,avx2,avx512,xeon,gold-6130,r740      gpu:v100:2
hpc[0965-0969,0971-0972]                12    19456   myri,xeon,X5650,s1160                        (null)
hpc[0981-1021,1040-1050,1118-1128,1196-1199,1223-1 24  44032   myri,opteron,AMD6176,d1165                  (null)
hpc[1407-1414]                          8     14890   myri,xeon,E5410,dx340                      (null)
hpc[2726-2729,2758-2759,2761]           12    19456   myri,xeon,X5650,dx360                      (null)
hpc[3025-3027,3031-3063,3065-3241,3243-3264] 16    59392   IB,avx,xeon,E5-2665,s1250s                  gpu:k20:2
hpc[3064,3242]                          16    59392   IB,avx,xeon,E5-2665,s1250s                  gpu:k20:1
hpc[3591-3594,3598-3600,3606-3607,3648-3688,3749,3 16    59392+  IB,avx,xeon,E5-2650v2,s1230s                (null)
hpc[3769-3792,3888,3938-3939,3947-3949,3966,3987-4 16    59392   IB,avx,avx2,xeon,E5-2640v3,nx360            (null)
hpc[3817-3834,3852]                     16    59392   IB,avx,avx2,xeon,E5-2640v3,nx360            gpu:k40:2
hpc[4323-4324]                          20    59392   IB,avx,avx2,xeon,E5-2640v4,x1170r          (null)
hpc[4331-4361,4364-4372]                20    123904  IB,avx,avx2,xeon,E5-2640v4,x1190r          gpu:p100:2
hpc[4522-4523]                          24    94444   IB,avx,avx2,xeon,silver-4116,r440          (null)
hpc[4578-4616]                          24    94444   IB,avx,avx2,xeon,silver-4116,c6420         (null)
```

hpc[3520-3527]	16	123904	IB,avx,xeon,E5-2665,sl230s	(null)
hpc[4129-4176]	20	59392	IB,avx,avx2,xeon,E5-2640v4,nx360	gpu:k40:2

Show all nodes in new [scavenge](#) partition.

\$ sinfo --partition=main,large,quick,long -o "%50N %4c %8m %40f %12G" uniq				
NODELIST	CPUS	MEMORY	AVAIL_FEATURES	GRES
hpc[4433-4520]	20	59392	IB,avx,avx2,xeon,E5-2640v4,r430	(null)
hpc[4570-4573,4623-4632,4657-4674]	32	188888	IB,avx,avx2,avx512,xeon,gold-6130,r740	gpu:v100:2
hpc[0965-0969,0971-0972]	12	19456	myri,xeon,X5650,sl160	(null)
hpc[0981-1021,1040-1050,1118-1128,1196-1199,1223-1	24	44032	myri,opteron,AMD6176,dl165	(null)
hpc[1407-1414]	8	14890	myri,xeon,E5410,dx340	(null)
hpc[2726-2729,2758-2759,2761]	12	19456	myri,xeon,X5650,dx360	(null)
hpc[3025-3027,3031-3063,3065-3241,3243-3264]	16	59392	IB,avx,xeon,E5-2665,sl250s	gpu:k20:2
hpc[3064,3242]	16	59392	IB,avx,xeon,E5-2665,sl250s	gpu:k20:1
hpc[3591-3594,3598-3600,3606-3607,3648-3688,3749,3	16	59392+	IB,avx,xeon,E5-2650v2,sl230s	(null)
hpc[3769-3792,3888,3938-3939,3947-3949,3966,3987-4	16	59392	IB,avx,avx2,xeon,E5-2640v3,nx360	(null)
hpc[3817-3834,3852]	16	59392	IB,avx,avx2,xeon,E5-2640v3,nx360	gpu:k40:2
hpc[4323-4324]	20	59392	IB,avx,avx2,xeon,E5-2640v4,xl170r	(null)
hpc[4331-4361,4364-4372]	20	123904	IB,avx,avx2,xeon,E5-2640v4,xl190r	gpu:p100:2
hpc[4522-4523]	24	94444	IB,avx,avx2,xeon,silver-4116,r440	(null)
hpc[4578-4616]	24	94444	IB,avx,avx2,xeon,silver-4116,c6420	(null)
hpc[3520-3527]	16	123904	IB,avx,xeon,E5-2665,sl230s	(null)
hpc[4129-4176]	20	59392	IB,avx,avx2,xeon,E5-2640v4,nx360	gpu:k40:2

Node State

To display the state of a node, add the format option **%6t**.

You can [pipe](#) the output above to the [grep](#) utility to find nodes of special interest. To find idle GPU nodes, grep both ‘idle’ and ‘gpu’.

\$ sinfo --partition=main,large,quick,long -o "%6t %50N %4c %8m %40f %12G" uniq grep idle grep gpu				
idle hpc[3158-3167,3173-3209,3212-3231,3234,3237,3239-3	16	59392	IB,avx,xeon,E5-2665,sl250s	gpu:k20:2
idle hpc[3832-3833]	16	59392	IB,avx,avx2,xeon,E5-2640v3,nx360	gpu:k40:2
idle hpc[4129-4134,4136-4176]	20	59392	IB,avx,avx2,xeon,E5-2640v4,nx360	gpu:k40:2

GPU Nodes

To find GPU nodes in the scavenge partition, change the partition request. (Note that there were no idle GPU nodes in the scavenge partition when this was run.)

\$ sinfo --partition=scavenge -o "%6t %50N %4c %8m %40f %12G" uniq grep gpu				
down* hpc3014	16	59392	IB,avx,xeon,E5-2665,sl250s	gpu:k20:2
mix hpc[3028-3030]	16	59392	IB,avx,xeon,E5-2665,sl250s	gpu:k20:2
alloc hpc[3001-3013,3015-3020,3022-3024]	16	59392	IB,avx,xeon,E5-2665,sl250s	gpu:k20:2
alloc hpc3021	16	59392	IB,avx,xeon,E5-2665,sl250s	gpu:k20:1

Node Memory

You can use [awk](#) to restrict the output numerically. To view nodes in the general partition with more than 128GB of memory, use awk to compare the value in column 4 (\$4) to 128000.

\$ sinfo --partition=main,large,quick,long -o "%6t %50N %4c %8m %40f %12G" uniq awk '{ if (\$4 > 128000) print }'				
STATE NODELIST	CPUS	MEMORY	AVAIL_FEATURES	GRES
drain* hpc4659	32	188888	IB,avx,avx2,avx512,xeon,gold-6130,r740	gpu:v100:2
drain hpc[4570,4661,4665]	32	188888	IB,avx,avx2,avx512,xeon,gold-6130,r740	gpu:v100:2
alloc hpc[3591-3594,3598-3600,3606-3607,3648-3688,3749,3	16	59392+	IB,avx,xeon,E5-2650v2,sl230s	(null)
alloc hpc[4571-4573,4623-4632,4657-4658,4660,4662-4664,4	32	188888	IB,avx,avx2,avx512,xeon,gold-6130,r740	gpu:v100:2

Node CPU Cores

Similarly, to view nodes in the scavenge partition with more than 16 cores, use awk to compare the value in column 3 (\$3) with 16.

\$ sinfo --partition=scavenge -o "%6t %50N %4c %8m %40f %12G" awk '{ if (\$3 > 16) print }'				
STATE NODELIST	CPUS	MEMORY	AVAIL_FEATURES	GRES
mix hpc[1028-1039,1129-1132]	24	44032	myri,opteron,AMD6176,dl165	(null)

Counting Nodes

If you want a count of the nodes with a particular characteristic, like those with P100 GPUS or RAM > 128GB, you can add the command option **-N**, which will list individual nodes, line-by-line, and then use the word count utility [wc -l](#) to count the number of lines printed.

For example, there are 14 nodes in the scavenge partition, 342 of which are idle, 27 of which are GPU nodes, 17 of which have more than 16 cores, and 14 of which have more than 100GB.

\$ sinfo -N --partition=scavenge -o "%6t %8N %2c %7m %40f %12G" wc -l	
507	

\$ sinfo -N --partition=scavenge -o "%6t %8N %2c %7m %40f %12G" grep idle wc -l	
342	

\$ sinfo -N --partition=scavenge -o "%6t %8N %2c %7m %40f %12G" grep gpu wc -l	
27	

\$ sinfo -N --partition=scavenge -o "%6t %8N %2c %7m %40f %12G" awk '{ if (\$3 > 16) print }' wc -l	
17	

```
$ sinfo -N --partition=scavenge -o "%6t %8N %2c %7m %40f %12G" | awk '{ if ($4 > 128000) print }' | wc -l
1
```

There are 6 idle GPU nodes in the scavenge partition. To display the properties of these nodes, remove the `-N` option and the word count command.

```
$ sinfo -N --partition=scavenge -o "%6t %8N %2c %8m %40f %12G" | grep gpu | grep idle | wc -l
6
```

```
$ sinfo --partition=scavenge -o "%6t %8N %2c %8m %40f %12G" | grep gpu | grep idle
idle    hpc[3017 16 59392    IB,avx,xeon,E5-2665,s1250s                gpu:k20:2 6
```

To condense its output, Slurm will combine similar nodes as much as possible when displaying. if you want to see the names of all the nodes, you must widen the print width. Here we changed `%8N` to `%50N`. We also added `%P` to display the partition.

```
$ sinfo --partition=scavenge -o "%10P %6t %50N %2c %8m %40f %12G" | grep gpu | grep idle
scavenge  idle    hpc[3017,3020,3022-3023,3028-3030]                16 59392    IB,avx,xeon,E5-2665,s1250s                gpu:k20:2
```

SCONTROL: Job, Node and Partition Information

The Slurm command [control show](#) is used to view information about resource configuration and state. Type **man scontrol** for details.

Slurm configuration

```
scontrol show config
```

Job information

Use **scontrol show partition *partition_name*** to see information about completed jobs.

```
scontrol show job <job_id>
```

Node Information

```
scontrol show nodes $nodename
```

Partition information

Use **control show partition *partition_name*** to see information about the configuration of a partition.

```
scontrol show partition largemem
PartitionName=largemem
  AllowGroups=ALL AllowAccounts=ALL AllowQos=ALL
  AllocNodes=ALL Default=NO QoS=largemem_partitionLimits
  DefaultTime=00:30:00 DisableRootJobs=NO ExclusiveUser=NO GraceTime=0 Hidden=NO
  MaxNodes=1 MaxTime=14-00:00:00 MinNodes=1 LLN=NO MaxCPUsPerNode=UNLIMITED
  Nodes=hpc-1t-[1-4]
  PriorityJobFactor=1 PriorityTier=1 RootOnly=NO ReqResv=NO OverSubscribe=EXCLUSIVE
  OverTimeLimit=NONE PreemptMode=OFF
  State=UP TotalCPUs=160 TotalNodes=4 SelectTypeParameters=NONE
  DefMemPerNode=UNLIMITED MaxMemPerNode=UNLIMITED
  TRESBillingWeights=CPU=1.0,Mem=1.0
```

For the largemem partition, we see the following. Try this with the *large*, *quick* and *long* partitions.

- **AllowAccounts=ALL** – Anyone may request nodes for this partition. For condo partitions, the value will be a list of all accounts with access to the partition; only members of those accounts can request the partition.
- **MaxNodes=1** – Only one node at a time may be requested for this partition. For condo partitions, the value will be the total number of nodes available.
- **Nodes=hpc-1t-[1-4]** – The nodes hpc-1t-1, hpc-1t-2, hpc-1t-3, and hpc-1t-4 are available for this partition. For condo partitions, the value will be a list of nodes available.
- **Maxtime=14-00:00:00** – Up to 14 hours of wall time may be requested for this partition.
- **TotalCPUs=160 (TotalNodes=4)** – There are 160 cores in the partition (there are 4 nodes in the partition)

License information

Currently, this displays information about HPC’s MATLAB licenses.

```
scontrol show licenses
```

SACCT: Resource Accounting Information

The Slurm command [sacct](#) is used to view information about resources. Type **man sacct** for details.

Job information

Show running or completed job information.

```
sacct -j <job_id>          #basic
sacct -p -j <job_id>       #parsable
sacct -X -j <job_id>       #omit steps
sacct -X -j <job_id> -l    #default long format
sacct -X -j <job_id> --format=account,user,jobid,jobname,partition,state,elapsed,elapseddraw,start,end,maxrss,maxvmsize,consumedenergy
```

Show jobs run on a particular day.

```
sacct -X --starttime 2019-06-01 -u $USER
```

Partition information

SACCTMGR: Account Manager

Show your default account and account associations.

```
sacctmgr show user $USER
sacctmgr list association where users=$USER
```

References:

- <https://schedmd.com>
- https://hpc.uni.lu/users/docs/slurm_examples.html
- https://ecs.rutgers.edu/slurm_commands.html

Getting help: If you have any questions or concerns, please send an email to hpc@usc.edu.

[Edit](#)

USC

3434 South Grand Ave
Los Angeles, CA 90089-0706

hpc@usc.edu

Website by: [USC Web Services](#)