slurm.schedmd.com

Slurm Workload Manager - sacct

40–51 minutes

Section: Slurm Commands (1)
Updated: Slurm Commands
Index

NAME

sacct - displays accounting data for all jobs and job steps in the Slurm job accounting log or Slurm database

SYNOPSIS

sacct [*OPTIONS*...]

DESCRIPTION

Accounting information for jobs invoked with Slurm are either logged in the job accounting log file or saved to the Slurm database, as configured with the AccountingStorageType parameter.

The **sacct** command displays job accounting data stored in the job accounting log file or Slurm database in a variety of forms for your analysis. The **sacct** command displays information on jobs, job steps, status, and exitcodes by default. You can tailor the output with the use of the **--format=** option to specify the fields to be shown.

Job records consist of a primary entry for the job as a whole as well as entries for job steps. The Job Launch page has a more detailed description of each type of job step. <https://slurm.schedmd.com/job_launch.html#job_record>

For the root user, the **sacct** command displays job accounting data for all users, although there are options to filter the output to report only the jobs from a specified user or group.

For the non-root user, the **sacct** command limits the display of job accounting data to jobs that were launched with their own user identifier (UID) by default. Data for other users can be displayed with the **--allusers**, **--user**, or **--uid** options.

Elapsed time fields are presented as [days-]hours:minutes:seconds[.microseconds]. Only 'CPU' fields will ever have microseconds.

The default input file is the file named in the **AccountingStorageLoc** parameter in slurm.conf.

NOTE: If designated, the slurmdbd.conf option PrivateData may further restrict the accounting data visible to users which are not SlurmUser, root, or a user with

AdminLevel=Admin. See the slurmdbd.conf man page for additional details on restricting access to accounting data.

NOTE: The contents of Slurm's database are maintained in lower case. This may result in some **sacct** output differing from that of other Slurm commands.

NOTE: Much of the data reported by **sacct** has been generated by the *wait3()* and *getrusage()* system calls. Some systems gather and report incomplete information for these calls; **sacct** reports values of 0 for this missing data. See your systems *getrusage (3)* man page for information about which data are actually available on your system.

OPTIONS

-A, **--accounts**=<*account_list*>
Displays jobs when a comma separated list of accounts are given as the argument.

--array
Expand job arrays. Display all array tasks on separate lines instead of displaying groups of array tasks on single lines.

-L, **--allclusters**
Display jobs ran on all clusters. By default, only jobs ran on the cluster from where **sacct** is called are displayed.

-X, **--allocations**
Only show statistics relevant to the job allocation itself, not taking steps into consideration.
NOTE: Without including steps, utilization statistics for job allocation(s) will be reported as zero.

-a, **--allusers**
Displays all users' jobs when run by user root or if **PrivateData** is not configured to **jobs**. Otherwise display the current user's jobs

-x, **--associations**=<*assoc_list*>
Displays the statistics only for the jobs running under the association ids specified by the **assoc_list** operand, which is a comma-separated list of association ids. Space characters are not allowed in the **assoc_list**. Default is all associations.

-B, **--batch-script**
This option will print the batch script of job if the job used one. If the job didn't have a script 'NONE' is output.
NOTE: AccountingStoreFlags=job_script is required for this.
NOTE: Requesting specific job(s) with '-j' is required for this.

-b, **--brief**
Displays a brief listing consisting of JobID, State, and ExitCode.

-M, **--clusters**=<*cluster_list*>
Displays the statistics only for the jobs started on the clusters specified by the *cluster_list* operand, which is a comma-separated list of clusters. Space characters are not allowed in the *cluster_list*. A value of '*all*' will query to run on all clusters. The default is current cluster you are executing the **sacct** command on or all clusters in the federation when executed on a federated cluster. This option implicitly sets the **--local** option.

-c, **--completion**
Use job completion data instead of job accounting. The **JobCompType** parameter in the slurm.conf file must be defined to a non-none option. Does not support federated cluster information (local data only).

-C, **--constraints**=<*constraint_list*>
Comma separated list to filter jobs based on what constraints/features the job requested. Multiple options will be treated as 'and' not 'or', so the job would need all constraints specified to be returned not one or the other.

--delimiter=<*characters*>
ASCII characters used to separate the fields when specifying the **-p** or **-P** options. The default delimiter is a '|'. This option is ignored if **-p** or **-P** options are not specified.

-D, **--duplicates**
If Slurm job ids are reset, some job numbers will probably appear more than once in the accounting log file but refer to different jobs. Such jobs can be distinguished by the "submit" time stamp in the data records.
When data for specific jobs are requested with the --jobs option, **sacct** returns the most recent job with that number. This behavior can be overridden by specifying --duplicates, in which case all records that match the selection criteria will be returned.

NOTE: Revoked federated sibling jobs are hidden unless the **--duplicates** option is specified.

-E, **--endtime**=<*end_time*>
Select jobs in any state before the specified time. If states are given with the -s option return jobs in this state before this period. See the **DEFAULT TIME WINDOW** section (below) for details about how the default values for --starttime and --endtime are determined.
Valid time formats are:
HH:MM[:SS][AM|PM]
MMDD[YY][-HH:MM[:SS]]
MM.DD[.YY][-HH:MM[:SS]]
MM/DD[/YY][-HH:MM[:SS]]
YYYY-MM-DD[THH:MM[:SS]]
today, midnight, noon, elevenses (11 AM), fika (3 PM), teatime (4 PM)
now[{+|-}*count*[seconds(default)|minutes|hours|days|weeks]]

--env-vars
This option will print the running environment of a batch job, otherwise 'NONE' is output.
NOTE: AccountingStoreFlags=job_env is required for this.
NOTE: Requesting specific job(s) with '-j' is required for this.

--expand-patterns
Expand any filename patterns from in **StdOut**, **StdErr** and **StdIn**. Fields that map to a range of values will use the first value of the range. For example "%t" for task id will be replaced by "0".

--federation
Show jobs from the federation if a member of one.

-f, **--file**=<*file*>
Causes the **sacct** command to read job accounting data from the named *file* instead of the current Slurm job accounting log file. Only applicable when running the jobcomp/filetxt plugin. Setting this flag implicitly enables the -c flag.

-F, **--flags**=<*flag_list*>
Comma separated list to filter jobs based on what various ways the jobs were handled. Current flags are SchedSubmit, SchedMain, SchedBackfill and StartReceived. SchedSubmit, SchedMain, SchedBackfill describe the scheduler that started the job.

-o, **--format**
Comma separated list of fields. (use "--helpformat" for a list of available fields).
NOTE: When using the format option for listing various fields you can put a %NUMBER afterwards to specify how many characters should be printed.

e.g. format=name%30 will print 30 characters of field name right justified. A %-30 will print 30 characters left justified.

When set, the SACCT_FORMAT environment variable will override the default format. For example:

SACCT_FORMAT="jobid,user,account,cluster"

-g, **--gid**=, **--group**=<*gid_or_group_list*>
Displays the statistics only for the jobs started with the GID or the GROUP specified by the *gid_list* or the *group_list* operand, which is a comma-separated list. Space characters are not allowed. Default is no restrictions.

-h, **--help**
Displays a general help message.

-e, **--helpformat**
Print a list of fields that can be specified with the **--format** option.

Fields available:

Account AdminComment AllocCPUS AllocNodes AllocTRES AssocID AveCPU AveCPUFreq AveDiskRead AveDiskWrite AvePages AveRSS AveVMSize BlockID Cluster Comment Constraints ConsumedEnergy ConsumedEnergyRaw Container CPUTime CPUTimeRAW DBIndex DerivedExitCode Elapsed ElapsedRaw Eligible End ExitCode Extra FailedNode Flags GID Group JobID JobIDRaw JobName Layout Licenses MaxDiskRead MaxDiskReadNode MaxDiskReadTask MaxDiskWrite MaxDiskWriteNode MaxDiskWriteTask MaxPages MaxPagesNode MaxPagesTask MaxRSS MaxRSSNode MaxRSSTask MaxVMSize MaxVMSizeNode MaxVMSizeTask McsLabel MinCPU MinCPUNode MinCPUTask NCPUS NNodes NodeList NTasks Partition Planned PlannedCPU PlannedCPURAW Priority QOS QOSRAW QOSREQ Reason ReqCPUFreq ReqCPUFreqGov ReqCPUFreqMax ReqCPUFreqMin ReqCPUS ReqMem ReqNodes ReqTRES ReqReservation Reservation ReservationId Restarts SegmentSize SLUID Start State StdErr StdIn StdOut Submit SubmitLine Suspended SystemComment SystemCPU Timelimit TimelimitRaw TotalCPU TRESUsageInAve TRESUsageInMax TRESUsageInMaxNode TRESUsageInMaxTask TRESUsageInMin TRESUsageInMinNode TRESUsageInMinTask TRESUsageInTot TRESUsageOutAve TRESUsageOutMax TRESUsageOutMaxNode TRESUsageOutMaxTask TRESUsageOutMin TRESUsageOutMinNode TRESUsageOutMinTask TRESUsageOutTot UID User UserCPU WCKey WCKeyID WorkDir

NOTE: When using with Ave[RSS|VM]Size or their values in TRESUsageIn[Ave|Tot]. They represent the average/total of the highest watermarks over all ranks in the step. When using sstat they represent the average/

total at the moment the command was run.

NOTE: TRESUsage*Min* values represent the lowest highwater mark in the step.

NOTE: Availability of metrics rely on the **jobacct_gather** plugin used. For example the jobacct_gather/cgroup in combination with cgroup/v2 does not provide Virtual Memory metrics due to limitations in the kernel cgroups interfaces and will show a 0 for the related fields.

The section titled "Job Accounting Fields" describes these fields.

-j, **--jobs**=<*job*[.*step*]>
Displays information about the specified *job*[.*step*] or list of *job*[.*step*]s. The *job*[.*step*] parameter is a comma-separated list of jobs. Space characters are not permitted in this list.
NOTE: A step id of 'batch' will display the information about the batch step.
By default sacct shows only jobs with Eligible time, but with this option the non-eligible will be also shown.
NOTE: If --state is also specified, as non-eligible are not PD, then non-eligible jobs will not be displayed. See the **DEFAULT TIME WINDOW** section (below) for details about how this option changes the default values for --starttime and --endtime.

--json, **--json**=*list*, **--json**=<*data_parser*>
Dump job information as JSON using the default data_parser plugin or explicit data_parser with parameters. Sorting and formatting arguments will be ignored.

--local
Show only jobs local to this cluster. Ignore other clusters in this federation (if any). Overrides --federation.

-l, **--long**
Equivalent to specifying:
--format=jobid,jobidraw,jobname,partition,maxvmsize,maxvmsizenode,
maxvmsizetask,avevmsize,maxrss,maxrssnode,maxrsstask,averss,maxpages,
maxpagesnode,maxpagestask,avepages,mincpu,mincpunode,mincputask,avecpu,ntasks,
alloccpus,elapsed,state,exitcode,avecpufreq,reqcpufreqmin,reqcpufreqmax,
reqcpufreqgov,reqmem,consumedenergy,maxdiskread,maxdiskreadnode,maxdiskreadtask,
avediskread,maxdiskwrite,maxdiskwritenode,maxdiskwritetask,avediskwrite,
reqtres,alloctres,tresusageinave,tresusageinmax,
tresusageinmaxn,tresusageinmaxt,tresusageinmin,tresusageinminn,tresusageinmint,
tresusageintot,tresusageoutmax,tresusageoutmaxn,
tresusageoutmaxt,tresusageoutave,tresusageouttot

--name=<*jobname_list*>
Display jobs that have any of these name(s).

-i, **--nnodes**=<*min*[-*max*]>
Return jobs that ran on the specified number of nodes.

-I, **--ncpus**=<*min*[-*max*]>
Return jobs that ran on the specified number of cpus.

--noconvert
Don't convert units from their original type (e.g. 2048M won't be converted to 2G).

-N, **--nodelist**=<*node_list*>
Display jobs that ran on any of these node(s). *node_list* can be a ranged string.
NOTE: This is not reliable when nodes are added or removed to Slurm while jobs are running. Only jobs that started in the specified time range (-S, -E) will be returned.

-n, **--noheader**
No heading will be added to the output. The default action is to display a header.

-p, **--parsable**
Output will be '|' delimited with a '|' at the end. See also the **--delimiter** option.

-P, **--parsable2**
Output will be '|' delimited without a '|' at the end. See also the **--delimiter** option.

-r, **--partition**
Comma separated list of partitions to select jobs and job steps from. The default is all partitions.

-q, **--qos**
Only send data about jobs using these qos. Default is all.

-R, **--reason**=<*reason_list*>
Comma separated list to filter jobs based on what reason the job wasn't scheduled outside resources/priority.

-S, **--starttime**
Select jobs in any state after the specified time. Default is 00:00:00 of the current day, unless the '-s' or '-j' options are used. If the '-s' option is used, then the default is 'now'. If states are given with the '-s' option then only jobs in this state at this time will be returned. If the '-j' option is used, then the default time is Unix Epoch 0. See the **DEFAULT TIME WINDOW** section (below) for details about how default values for --starttime and --endtime are determined.
Valid time formats are:
HH:MM[:SS][AM|PM]
MMDD[YY][-HH:MM[:SS]]
MM.DD[.YY][-HH:MM[:SS]]
MM/DD[/YY][-HH:MM[:SS]]
YYYY-MM-DD[THH:MM[:SS]]
today, midnight, noon, elevenses (11 AM), fika (3 PM), teatime (4 PM)
now[{+|-}*count*[seconds(default)|minutes|hours|days|weeks]]

-s, **--state**=<*state_list*>
Selects jobs based on their state during the time period given. Unless otherwise specified, the start and end time will be the current time when the **--state** option is specified and only currently running jobs can be displayed. A start and/or end time must be specified to view information about jobs not currently running. See the **JOB STATE CODES** section below for a list of state designators. Multiple state names may be specified using comma separators. Either the short or long form of the state name may be used (e.g. **CA** or **CANCELLED**) and the name is case insensitive (i.e. ca and **CA** both work).
NOTE: Note for a job to be selected in the PENDING state it must have "EligibleTime" in the requested time interval or different from "Unknown". The "EligibleTime" is displayed by the "scontrol show job" command. For example jobs submitted with the "--hold" option will have "EligibleTime=Unknown" as they are pending indefinitely.

NOTE: When specifying states and no start time is given the default start time is 'now'. This is only when -j is not used. If -j is used the start time will default to 'Epoch'. In both cases if no end time is given it will default to 'now'. See the **DEFAULT TIME WINDOW** section (below) for details about how this option changes the default values for --starttime and --endtime.

-K, **--timelimit-max**
Ignored by itself, but if timelimit_min is set this will be the maximum timelimit of the range. Default is no restriction.

-k, **--timelimit-min**
Only send data about jobs with this timelimit. If used with timelimit_max this will

be the minimum timelimit of the range. Default is no restriction.

-T, **--truncate**
Truncate time. So if a job started before --starttime the start time would be truncated to --starttime. The same for end time and --endtime.

-u, --uid=, **--user**=<*uid_or_user_list*>
Use this comma separated list of UIDs or user names to select jobs to display. By default, the running user's UID is used.

--units=[**KMGTP**]
Display values in specified unit type. Takes precedence over **--noconvert** option.

--usage
Display a command usage summary.

--use-local-uid
When displaying UID, sacct uses the UID stored in Slurm's accounting database by default. Use this command to make Slurm use a system call to get the UID from the username. This option may be useful in an environment with multiple clusters and one database where the UIDs aren't the same on all clusters.

-v, **--verbose**
Primarily for debugging purposes, report the state of various variables during processing.

-V, --version
Print version.

-W, **--wckeys**=<*wckey_list*>
Displays the statistics only for the jobs started on the wckeys specified by the *wckey_list* operand, which is a comma-separated list of wckey names. Space characters are not allowed in the *wckey_list*. Default is all wckeys.

--whole-hetjob[=yes|no]
When querying and filtering heterogeneous jobs with **--jobs**, Slurm will default to retrieving information about all the components of the job if the het_job_id (leader id) is selected. If a non-leader heterogeneous job component id is selected then only that component is retrieved by default. This behavior can be changed by using this option. If set to 'yes' (or no argument), then information about all the components will be retrieved no matter which component is selected in the job filter. If set to 'no' then only the selected heterogeneous job component(s) will be retrieved, even when selecting the leader.

--yaml, **--yaml**=*list*, **--yaml**=<*data_parser*>
Dump job information as YAML using the default data_parser plugin or explicit data_parser with parameters. Sorting and formatting arguments will be ignored.

Job Accounting Fields

Descriptions of each field option can be found below. Note that the Ave*, Max* and Min* accounting fields look at the values for all the tasks of each step in a job and return the average, maximum or minimum values of the task for that job step. For example, for MaxRSS, the returned value is the maximum memory consumption seen by one of the tasks of the step, and MaxRSSTask shows which task it is.

ALL
Print all fields listed below.

Account
Account the job ran under.

AdminComment
A comment string on a job that must be set by an administrator, the SlurmUser or root.

AllocCPUs

Count of allocated CPUs. Equivalent to **NCPUS**.

AllocNodes
Number of nodes allocated to the job/step. 0 if the job is pending.

AllocTres
Trackable resources. These are the resources allocated to the job/step after the job started running. For pending jobs this should be blank. For more details see AccountingStorageTRES in slurm.conf.
NOTE: When a generic resource is configured with the no_consume flag, the allocation will be printed with a zero.

AssocID
Reference to the association of user, account and cluster.

AveCPU
Average (system + user) CPU time of all tasks in job.

AveCPUFreq
Average weighted CPU frequency of all tasks in job, in kHz.

AveDiskRead
Average number of bytes read by all tasks in job.

AveDiskWrite
Average number of bytes written by all tasks in job.

AvePages
Average number of page faults of all tasks in job.

AveRSS
Average resident set size of all tasks in job.

AveVMSize
Average Virtual Memory size of all tasks in job.

BlockID
The name of the block to be used (used with Blue Gene systems).

Cluster
Cluster name.

Comment
The job's comment string when the AccountingStoreFlags parameter in the slurm.conf file contains 'job_comment'. The Comment string can be modified by invoking **sacctmgr modify job** or the specialized **sjobexitmod** command.

Constraints
Feature(s) the job requested as a constraint.

ConsumedEnergy
Total energy consumed by all tasks in a job, in joules. Value may include a unit prefix (K,M,G,T,P). Note: Only in the case of an exclusive job allocation does this value reflect the job's real energy consumption.

ConsumedEnergyRaw
Total energy consumed by all tasks in a job, in joules. Note: Only in the case of an exclusive job allocation does this value reflect the job's real energy consumption.

Container
Path to OCI Container Bundle requested.

CPUTime
Time used (Elapsed time * CPU count) by a job or step in HH:MM:SS format.

CPUTimeRAW
Time used (Elapsed time * CPU count) by a job or step in cpu-seconds.

DBIndex
Unique database index for entries in the job table.

DerivedExitCode
The highest exit code returned by the job's job steps (srun invocations).

Following the colon is the signal that caused the process to terminate if it was terminated by a signal. The DerivedExitCode can be modified by invoking **sacctmgr modify job** or the specialized **sjobexitmod** command.

Elapsed
The job's elapsed time.
The format of this field's output is as follows:

[DD-[HH:]]MM:SS

as defined by the following:

DD
days

hh
hours

mm
minutes

ss
seconds

ElapsedRaw
The job's elapsed time in seconds.

Eligible
When the job became eligible to run. In the same format as **End**.

End
Termination time of the job. The output is of the format YYYY-MM-DDTHH:MM:SS, unless changed through the SLURM_TIME_FORMAT environment variable.

ExitCode
The exit code returned by the job script or salloc, typically as set by the exit() function. Following the colon is the signal that caused the process to terminate if it was terminated by a signal.

Extra
The job's extra string when the AccountingStoreFlags parameter in the slurm.conf file contains 'job_extra'. The Extra string can be modified by invoking **sacctmgr modify job** command.

FailedNode
The name of the node whose failure caused the job to be killed.

Flags
Job flags. Current flags are SchedSubmit, SchedMain, SchedBackfill.

GID
The group identifier of the user who ran the job.

Group
The group name of the user who ran the job.

JobID
The identification number of the job or job step.
Regular jobs are in the form:

JobID[.JobStep]

Array jobs are in the form:

ArrayJobID_ArrayTaskID

Heterogeneous jobs are in the form:

HetJobID+HetJobOffset

When printing job arrays, performance of the command can be measurably improved for systems with large numbers of jobs when a single job ID is specified. By default, this field size will be limited to 64 bytes. Use the

environment variable SLURM_BITSTR_LEN to specify larger field sizes.

JobIDRaw
The identification number of the job or job step. Prints the JobID in the form *JobID[.JobStep]* for regular, heterogeneous and array jobs.

JobName
The name of the job or job step. The **slurm_accounting.log** file is a space delimited file. Because of this if a space is used in the jobname an underscore is substituted for the space before the record is written to the accounting file. So when the jobname is displayed by **sacct** the jobname that had a space in it will now have an underscore in place of the space.

Layout
What the layout of a step was when it was running. This can be used to give you an idea of which node ran which rank in your job.

MaxDiskRead
Maximum number of bytes read by all tasks in job.

MaxDiskReadNode
The node on which the maxdiskread occurred.

MaxDiskReadTask
The task ID where the maxdiskread occurred.

MaxDiskWrite
Maximum number of bytes written by all tasks in job.

MaxDiskWriteNode
The node on which the maxdiskwrite occurred.

MaxDiskWriteTask
The task ID where the maxdiskwrite occurred.

MaxPages
Maximum number of page faults of all tasks in job.

MaxPagesNode
The node on which the maxpages occurred.

MaxPagesTask
The task ID where the maxpages occurred.

MaxRSS
Maximum resident set size of all tasks in job.

MaxRSSNode
The node on which the maxrss occurred.

MaxRSSTask
The task ID where the maxrss occurred.

MaxVMSize
Maximum Virtual Memory size of all tasks in job.

MaxVMSizeNode
The node on which the maxvmsize occurred.

MaxVMSizeTask
The task ID where the maxvmsize occurred.

MCSLabel
Multi-Category Security (MCS) label associated with the job. Added to a job when the MCSPlugin is enabled in the slurm.conf.

MinCPU
Minimum (system + user) CPU time of all tasks in job.

MinCPUNode
The node on which the mincpu occurred.

MinCPUTask
The task ID where the mincpu occurred.

NCPUS
Total number of CPUs allocated to the job. Equivalent to **AllocCPUS**.

NNodes
Number of nodes in a job or step. If the job is running, or ran, this count will be the number allocated, else the number will be the number requested.

NodeList
List of nodes in job/step.

NTasks
Total number of tasks in a job or step.

Partition
Identifies the partition on which the job ran.

Planned
How much wall clock time was used as planned time for this job. This is derived from how long a job was waiting from eligible time to when it started or was cancelled. Format is the same as **Elapsed**.

PlannedCPU
How many CPU seconds were used as planned time for this job. Format is the same as **Elapsed**.

PlannedCPURAW
How many CPU seconds were used as planned time for this job. Format is in processor seconds.

Priority
Slurm priority.

QOS
Name of Quality of Service.

QOSRAW
Numeric id of Quality of Service.

QOSREQ
List of Quality of Services requested by the job.

Reason
The last reason a job was blocked from running for something other than Priority or Resources. This will be saved in the database even if the job ran to completion.

ReqCPUFreq
Requested CPU frequency for the step, in kHz. Note: This value applies only to a job step. No value is reported for the job.

ReqCPUFreqGov
Requested CPU frequency governor for the step, in kHz. Note: This value applies only to a job step. No value is reported for the job.

ReqCPUFreqMax
Maximum requested CPU frequency for the step, in kHz. Note: This value applies only to a job step. No value is reported for the job.

ReqCPUFreqMin
Minimum requested CPU frequency for the step, in kHz. Note: This value applies only to a job step. No value is reported for the job.

ReqCPUS
Number of requested CPUs.

ReqMem
Minimum required memory for the job. It may have a letter appended to it indicating units (M for megabytes, G for gigabytes, etc.). Note: This value is only from the job allocation, not the step.

ReqNodes
Requested minimum Node count for the job/step.

ReqTres
Trackable resources. These are the minimum resource counts requested by the job/step at submission time. For more details see AccountingStorageTRES in slurm.conf.

ReqReservation
Comma separated list of reservation names requested by the job.

Reservation
Reservation Name.

ReservationId
Reservation Id.

Restarts
How many times this job has been requeued/restarted.

SegmentSize
When a block topology is used, this is the size of the segments that will be used to create the job allocation.

Start
Initiation time of the job. In the same format as **End**.

State
Displays the job status, or state. See the **JOB STATE CODES** section below for a list of possible states.
If more information is available on the job state than will fit into the current field width (for example, the UID that CANCELLED a job) the state will be followed by a "+". You can increase the size of the displayed state using the "%NUMBER" format modifier described earlier.

NOTE: The RUNNING state will return suspended jobs as well. In order to print suspended jobs you must request SUSPENDED at a different call from RUNNING.

NOTE: The RUNNING state will return any jobs completed (cancelled or otherwise) in the time period requested as the job was also RUNNING during that time. If you are only looking for jobs that finished, please choose the appropriate state(s) without the RUNNING state.

StdErr
Display the "*filename pattern*" for stderr redirection specified in a job or in job steps. Path wildcards will not be substituted and will be shown as defined in the original batch submission.

StdIn
Display the "*filename pattern*" for stdin redirection specified in a job or in job steps. Path wildcards will not be substituted and will be shown as defined in the original batch submission.

StdOut
Display the "*filename pattern*" for stdout redirection specified in a job or in job steps. Path wildcards will not be substituted and will be shown as defined in the original batch submission.

Submit
The time the job was submitted. In the same format as **End**.
NOTE: If a job is requeued, the submit time is reset. To obtain the original submit time it is necessary to use the -D or --duplicate option to display all duplicate entries for a job.

SubmitLine
The full command issued to submit the job.

Suspended
The amount of time a job or job step was suspended. Format is the same as *Elapsed*.

SystemComment
The job's comment string that is typically set by a plugin. Can only be modified by a Slurm administrator.

SystemCPU

The amount of system CPU time used by the job or job step. Format is the same as **Elapsed**.
NOTE: See the note for TotalCPU for information about how canceled jobs are handled.

Timelimit
What the timelimit was/is for the job or job step. Format is the same as **Elapsed**, but two additional special values can be displayed:

Partition_limit
Indicates that the job did not have its time limit set and was not yet subjected to a partition MaxTime (i.e. job is pending). You can define the **DefaultTime** on the partition to avoid seeing this value.

UNLIMITED
Indicates the job did not have a time limit defined.

TimelimitRaw
What the timelimit was/is for the job or job step. Format is in number of minutes. **NOTE**: See **TimeLimit** description.

TotalCPU
The sum of the SystemCPU and UserCPU time used by the job or job step. The total CPU time of the job may exceed the job's elapsed time for jobs that include multiple job steps. Format is the same as **Elapsed**.
NOTE: For the steps interrupted by signal (e.g. scancel, job timeout) TotalCPU provides a measure of the task's parent process and may not include CPU time of child processes. This is a result of **wait3** resource usage (**getrusage**) internals. For processes completing in regular way all the descendant processes (forks and execs) resources are included. However, if the processes are killed the result may differ between proctrack plugins and end-user applications.

TresUsageInAve
Tres average usage in by all tasks in job. **NOTE**: If corresponding TresUsageInMaxTask is -1 the metric is node centric instead of task.

TresUsageInMax
Tres maximum usage in by all tasks in job. **NOTE**: If corresponding TresUsageInMaxTask is -1 the metric is node centric instead of task.

TresUsageInMaxNode
Node for which each maximum TRES usage out occurred.

TresUsageInMaxTask
Task for which each maximum TRES usage out occurred.

TresUsageInMin
Tres minimum usage in by all tasks in job. **NOTE**: If corresponding TresUsageInMinTask is -1 the metric is node centric instead of task.

TresUsageInMinNode
Node for which each minimum TRES usage out occurred.

TresUsageInMinTask
Task for which each minimum TRES usage out occurred.

TresUsageInTot
Tres total usage in by all tasks in job.

TresUsageOutAve
Tres average usage out by all tasks in job. **NOTE**: If corresponding TresUsageOutMaxTask is -1 the metric is node centric instead of task.

TresUsageOutMax
Tres maximum usage out by all tasks in job. **NOTE**: If corresponding TresUsageOutMaxTask is -1 the metric is node centric instead of task.

TresUsageOutMaxNode
Node for which each maximum TRES usage out occurred.

TresUsageOutMaxTask
Task for which each maximum TRES usage out occurred.
TresUsageOutMin
Tres minimum usage out by all tasks in job.
TresUsageOutMinNode
Node for which each minimum TRES usage out occurred.
TresUsageOutMinTask
Task for which each minimum TRES usage out occurred.
TresUsageOutTot
Tres total usage out by all tasks in job.
UID
The user identifier of the user who ran the job.
User
The user name of the user who ran the job.
UserCPU
The amount of user CPU time used by the job or job step. Format is the same as **Elapsed**.
NOTE: See the note for TotalCPU for information about how canceled jobs are handled.

WCKey
Workload Characterization Key. Arbitrary string for grouping orthogonal accounts together.
WCKeyID
Reference to the wckey.
WorkDir
The directory used by the job to execute commands.

JOB STATE CODES

The following states are recognized by sacct. A full list of possible states is available at <https://slurm.schedmd.com/job_state_codes.html>.

BF BOOT_FAIL
Job terminated due to launch failure, typically due to a hardware failure (e.g. unable to boot the node or block and the job can not be requeued).
CA CANCELLED
Job was explicitly cancelled by the user or system administrator. The job may or may not have been initiated.
CD COMPLETED
Job has terminated all processes on all nodes with an exit code of zero.
DL DEADLINE
Job terminated on deadline.
F FAILED
Job terminated with non-zero exit code or other failure condition.
NF NODE_FAIL
Job terminated due to failure of one or more allocated nodes.
OOM OUT_OF_MEMORY
Job experienced out of memory error.
PD PENDING
Job is awaiting resource allocation.
PR PREEMPTED
Job terminated due to preemption.
R RUNNING
Job currently has an allocation.

RQ REQUEUED
Job was requeued.
RS RESIZING
Job is about to change size.
RV REVOKED
Sibling was removed from cluster due to other cluster starting the job.
S SUSPENDED
Job has an allocation, but execution has been suspended and CPUs have been released for other jobs.
TO TIMEOUT
Job terminated upon reaching its time limit.

DEFAULT TIME WINDOW

The options --starttime and --endtime define the time window between which **sacct** is going to search. For historical and practical reasons their default values (i.e. the default time window) depends on other options: --jobs and --state.

Depending on if --jobs and/or --state are specified, the default values of **--starttime** and **--endtime** options are:

WITHOUT EITHER **--jobs** NOR **--state** specified:
--starttime defaults to Midnight.
--endtime defaults to Now.

WITH **--jobs** AND WITHOUT **--state** specified:
--starttime defaults to Epoch 0.
--endtime defaults to Now.

WITHOUT **--jobs** AND WITH **--state** specified:
--starttime defaults to Now.
--endtime defaults to --starttime and to Now if --starttime is not specified.

WITH BOTH **--jobs** AND **--state** specified:
--starttime defaults to Epoch 0.
--endtime defaults to --starttime or to Now if --starttime is not specified.

NOTE: With **-v/--verbose** a message about the actual time window in use is shown.

PERFORMANCE

Executing **sacct** sends a remote procedure call to **slurmdbd**. If enough calls from **sacct** or other Slurm client commands that send remote procedure calls to the **slurmdbd** daemon come in at once, it can result in a degradation of performance of the **slurmdbd** daemon, possibly resulting in a denial of service.

Do not run **sacct** or other Slurm client commands that send remote procedure calls to **slurmdbd** from loops in shell scripts or other programs. Ensure that programs limit calls to **sacct** to the minimum necessary for the information you are trying to gather.

ENVIRONMENT VARIABLES

Some **sacct** options may be set via environment variables. These environment variables, along with their corresponding options, are listed below. (Note: Command line options will always override these settings.)

SACCT_FEDERATION
Same as **--federation**

SACCT_FORMAT
Allows you to define the columns to display in the output. Same as **--format**

SACCT_LOCAL
Same as **--local**

SLURM_BITSTR_LEN
Specifies the string length to be used for holding a job array's task ID expression. The default value is 64 bytes. A value of 0 will print the full expression with any length required. Larger values may adversely impact the application performance.

SLURM_CONF
The location of the Slurm configuration file.

SLURM_DEBUG_FLAGS
Specify debug flags for sacct to use. See DebugFlags in the slurm.conf(5) man page for a full list of flags. The environment variable takes precedence over the setting in the slurm.conf.

SLURM_JSON
Control JSON serialization:

compact
Output JSON as compact as possible.

pretty
Output JSON in pretty format to make it more readable.

SLURM_TIME_FORMAT
Specify the format used to report time stamps. A value of *standard*, the default value, generates output in the form "year-month-dateThour:minute:second". A value of *relative* returns only "hour:minute:second" if the current day. For other dates in the current year it prints the "hour:minute" preceded by "Tomorr" (tomorrow), "Ystday" (yesterday), the name of the day for the coming week (e.g. "Mon", "Tue", etc.), otherwise the date (e.g. "25 Apr"). For other years it returns a date month and year without a time (e.g. "6 Jun 2012"). All of the time stamps use a 24 hour format.
A valid strftime() format can also be specified. For example, a value of "%a %T" will report the day of the week and a time stamp (e.g. "Mon 12:34:56").

SLURM_YAML
Control YAML serialization:
compact Output YAML as compact as possible.
pretty Output YAML in pretty format to make it more readable.

EXAMPLES

This example illustrates the default invocation of the **sacct** command:

```
# sacct
Jobid    Jobname    Partition    Account AllocCPUS State    ExitCode
---------- ---------- ---------- ---------- ---------- ---------- --------
2         script01   srun       acct1          1 RUNNING          0
3         script02   srun       acct1          1 RUNNING          0
4         endscript  srun        acct1           1 RUNNING          0
4.0                  srun       acct1          1 COMPLETED        0
```

This example shows the same job accounting information with the **brief** option.

```
# sacct --brief
```

```
   Jobid    State  ExitCode
---------- ---------- --------
2         RUNNING         0
3         RUNNING         0
4         RUNNING         0
4.0       COMPLETED         0

# sacct --allocations
Jobid     Jobname    Partition Account    AllocCPUS  State     ExitCode
---------- ---------- ---------- ---------- ------- ---------- --------
3         sja_init   andy      acct1          1 COMPLETED         0
4         sjaload    andy      acct1          2 COMPLETED         0
5         sja_scr1   andy      acct1          1 COMPLETED         0
6         sja_scr2   andy      acct1         18 COMPLETED         2
7         sja_scr3   andy      acct1         18 COMPLETED         0
8         sja_scr5   andy      acct1          2 COMPLETED         0
9         sja_scr7   andy      acct1         90 COMPLETED         1
10        endscript  andy      acct1        186 COMPLETED         0
```

This example demonstrates the ability to customize the output of the **sacct** command. The fields are displayed in the order designated on the command line.

```
# sacct --format=jobid,elapsed,ncpus,ntasks,state
   Jobid    Elapsed     Ncpus   Ntasks     State
---------- ---------- ---------- -------- ----------
3         00:01:30         2        1 COMPLETED
3.0        00:01:30         2        1 COMPLETED
4         00:00:00         2        2 COMPLETED
4.0        00:00:01         2        2 COMPLETED
5         00:01:23         2        1 COMPLETED
5.0        00:01:31         2        1 COMPLETED
```

This example demonstrates the use of the -T (--truncate) option when used with -S (--starttime) and -E (--endtime). When the -T option is used, the start time of the job will be the specified -S value if the job was started before the specified time, otherwise the time will be the job's start time. The end time will be the specified -E option if the job ends after the specified time, otherwise it will be the jobs end time.

Without -T (normal operation) sacct output would be like this.

```
# sacct -S2014-07-03-11:40 -E2014-07-03-12:00 -X -ojobid,start,end,state
  JobID              Start                End       State
--------- ------------------- ------------------- ------------
2         2014-07-03T11:33:16  2014-07-03T11:59:01   COMPLETED
3         2014-07-03T11:35:21  Unknown             RUNNING
4         2014-07-03T11:35:21  2014-07-03T11:45:21   COMPLETED
5         2014-07-03T11:41:01  Unknown             RUNNING
```

By adding the -T option the job's start and end times are truncated to reflect only the time requested. If a job started after the start time requested or finished before the end time requested those times are not altered. The -T option is useful when determining exact run times during any given period.

```
# sacct -T -S2014-07-03-11:40 -E2014-07-03-12:00 -X -
ojobid,jobname,user,start,end,state
   JobID             Start               End      State
--------- ------------------- ------------------- ------------
2         2014-07-03T11:40:00 2014-07-03T11:59:01  COMPLETED
3         2014-07-03T11:40:00 2014-07-03T12:00:00    RUNNING
4         2014-07-03T11:40:00 2014-07-03T11:45:21  COMPLETED
5         2014-07-03T11:41:01 2014-07-03T12:00:00    RUNNING
```

NOTE: If no **-s** (**--state**) option is given sacct will display eligible jobs during the specified period of time, otherwise it will return jobs that were in the state requested during that period of time.

This example demonstrates the differences running sacct with and without the **--state** flag for the same time period. Without the **--state** option, all eligible jobs in that time period are shown.

```
# sacct -S11:20:00 -E11:25:00 -X -ojobid,start,end,state
      JobID             Start               End      State
------------ ------------------- ------------------- ----------
2955                  11:15:12            11:20:12  COMPLETED
2956                  11:20:13            11:25:13  COMPLETED
```

With the **--state=pending** option, only job 2956 will be shown because it had a dependency on 2955 and was still PENDING from 11:20:00 until it started at 11:21:13. Note that even though we requested PENDING jobs, the State shows as COMPLETED because that is the current State of the job.

```
# sacct --state=pending -S11:20:00 -E11:25:00 -X -ojobid,start,end,state
      JobID             Start               End      State
------------ ------------------- ------------------- ----------
2956                  11:20:13            11:25:13  COMPLETED
```

COPYING

FILES

/etc/slurm.conf
Entries to this file enable job accounting and designate the job accounting log file that collects system job accounting.

/var/log/slurm_accounting.log
The default job accounting log file. By default, this file is set to read and write permission for root only.

SEE ALSO

sstat(1), **ps** (1), srun(1), squeue(1), **getrusage** (2), **time** (2)

Index

This document was created by *man2html* using the manual pages.
Time: 18:54:23 GMT, September 04, 2025