Moab

How do I use checkjob to diagnose job issues?

Issue: I’m trying to understand why my job is not running. The checkjob output can be confusing.

Affected Versions: All

Solution: The best way to analyze why a job will not run is to run “checkjob -v -v -v job-ID”. This will provide a wealth of job and system information, but it can be hard to understand. The remainder of this article will try to explain portions of the output that are not obvious. Many of the fields in the output simply reflect input values and submission options.

Note: if one only wishes to check the state of the job, running “checkjob” without the “-v”s will show that.

General Job Information:

The first section of the output will contain information about the job. An example of this section, for a running job, is:

job 123456 (RM job '123456.moabsvr')
AName: rabbits
State: Running
Creds: user:bugsbunny group:characters class:queue-1 qos:normal
WallTime: 1:01:07:52 of 1:00:00
SubmitTime: Sat Apr 24 15:16:44
(Time Queued Total: 00:00:25 Eligible: 00:00:00)

StartTime: Sat Apr 24 15:17:09
TemplateSets: DEFAULT
NodeMatchPolicy: EXACTNODE
Total Requested Tasks: 15
Total Requested Nodes: 1

Req[0] TaskCount: 15 Partition: pbs
Available Memory >= 0 Available Swap >= 0
Dedicated Resources Per Task: PROCS: 1
Utilized Resources Per Task: PROCS: 0.80 MEM: 3185M SWAP: 14G
Avg Util Resources Per Task: PROCS: 0.80
Max Util Resources Per Task: PROCS: 0.80 MEM: 3185M SWAP: 14G
Average Utilized Memory: 2001.13 MB
Average Utilized Procs: 12.11
TasksPerNode: 15 NodeCount: 1

Allocated Nodes:
[node045:15]

SystemID: Moab
SystemJID: Moab.12345
Notification Events: JobFail
Task Distribution:
node045,node045,node045,node045,node045,node045,node045,node045,node045,node045,node045,...
IWD: /home/bugsbunny
UMask: 0000
Executable: /opt/moab/spool/moab.job.rPn45q

OutputFile: /dev/null (moabsvr:/home/bugsbunny/test.out)
ErrorFile: /dev/null (moabsvr:/dev/null)

In the output above, the AName field is the account. Note that for this job, the WallTime limit has been exceeded, but options can be set in the system to allow for some amount of overrun.

(c) 2025 Shawn Hoopes <shoopes@adaptivecomputing.com> | 2025-05-11 11:21
URL: /phpmyfaq/index.php?action=artikel&cat=1&id=236&artlang=en

There are several fields that can either reflect the system defaults, or special options used on the job submission. One example is the “NodeMatchPolicy” field. In this example, it’s “EXACTNODE”, which is telling Moab to select as many nodes as requested even if it could pack multiple tasks onto the same node. If the job matched one or more job templates (configured using JOBCFG), the template(s) would show up in the “TemplateSets” field.

For this job, the State field is “Running”. Additional valid job states that might be seen are, in alphabetical order: BatchHold, Canceling, Completed, Deferred, Hold Idle, Running, Starting, Suspended, and UserHold. The Moab documentation will describe all of these states.

If a job has not yet started, it will not have the “StartTime” and utilization fields. It also will probably will not have the “Allocated Nodes” field.

Also worth noting is the “Task Distribution” list will be truncated for display purposes. If that information is needed, “checkjob --xml job-ID” will show all of the nodes in the “AllocNodeList” field. The CML output can be hard to read.

Additional useful information for jobs that are not yet running:

If a job is not yet running, it will have additional information displayed about the job. Here’s an example of that information for one such job:

Priority Analysis:
Job PRIORITY* Serv(QTime:Uprio)
Weights -------- 1(1: 1)

424503 9 100.0(9.0: 0.0)
PE: 1.00
Holds: Defer (DeferTime: 00:00:56 DeferCount: 0)

Node Availability for Partition pbs --------
node001 rejected: State (Busy) allocationpriority=0.00
<skipping most nodes>
node199 available: 20 tasks supported allocationpriority=831.56
node200 rejected: Class allocationpriority=1046.47
NOTE: job req cannot run in partition pbs (available procs do not meet
requirements: 48 procs needed, 42 procs found)
idle procs: 2939 feasible procs: 42

Node Rejection Summary: [Class: 86][State: 353]

BLOCK MSG: job hold active - Defer (recorded at last scheduling iteration)

Job Messages -----
Label CreateTime ExpireTime Owner Prio Num Message
0 -00:06:47 23:45:56 N/A 0 1 1 nodes unavailable to start
reserved job after 29 seconds (job Moab.120000[142] has exceeded wallclock
limit on node hpc-n012 - check job)

The ***Priority Analysis*** information may be useful, but running “mdiag -p” and “mdiag -f” will provide a better analysis of this information.

One of the most useful sections in this output is the ***Node Availability*** section. With the tripple “-v”s, this section will show each node in the system, along with either the available processors on that node, or information about why the node has been rejected from consideration. The reasons why a node may be rejected include:

Class: The node does not support the job’s class/queue

(c) 2025 Shawn Hoopes <shoopes@adaptivecomputing.com> | 2025-05-11 11:21
URL: /phpmyfaq/index.php?action=artikel&cat=1&id=236&artlang=en

CPU: The CPU on the node supports too few tasks per node
Features: The job requires features not configured on the node
Gres: There is a generic resource request in the job that the node cannot satisfy
HostList: A hostlist was specified for the job that did not include the node
Memory: Not enough free memory on the node to support the job
Reserved: The node has a reservation that will prevent the job from using it
State: The node is in a state the prevents it from being used
Swap: The job requires more swap space than the node has available

When a node is rejected for “Reserved”, it will show the reservation name. When a job is rejected for “State”, it will show the state of the node, which will be Busy, Down, Drained, Draining, or Running.

Also, the “NOTE” information, and “Node Rejection Summary” sections are useful for quickly seeing the resources that are available.

The “BLOCK MSG:” field will show the exact reason why Moab could not start the job. There may be additional messages in the “Job Messages” section, which may be information from Moab, or from a resource manager.

In the job used for this example, the reason it can’t run is actually quite interesting, and often missed by administrators. Moab had a reservation for this job, but a job already running on the target node has exceeded it’s wallclock limit, forcing Moab to push out the execution of this job.

The job messages are usually fairly clear. Giving a complete list of examples is not reasonable. Most of the messages generated by Moab will tell you what’s wrong. The messages that Moab report “RM Failure”, however, will require looking at the resource manager. For Torque, there will be an error code (for example, “rc: 15064, msg: 'Unknown node '”), and the Torque documentation has a complete list of those return-codes (“rc” field) and what they mean.

The final piece of the output will be the job script that was submitted.

Unique solution ID: #1236
Author: Rob Greenbank
Last update: 2022-09-13 16:56

(c) 2025 Shawn Hoopes <shoopes@adaptivecomputing.com> | 2025-05-11 11:21
URL: /phpmyfaq/index.php?action=artikel&cat=1&id=236&artlang=en