

# Moab

## Job over wall clock despite it not having started

**Description:** A job has a priority reservation and is scheduled to start, but gets deferred. There are log messages about the job not being able to start because another job has exceeded it's wallclock limit:

```
Jul 15 18:29:09 moab01 moab[64116]: reserved job 12345 cannot run. deferring - 5
nodes unavailable to start reserved job after 87 seconds (job 12346 has exceeded
wallclock limit on node node01 - check job)
```

The job that has supposedly exceeded it's wallclock limit hasn't started yet.

**Actual Problem:** This can happen if there is another priority reservation on one of the nodes that starts right after the first job is scheduled to finish. If one of the reserved nodes for the first job has a stray process that is consuming most or all of the resources on that node. Even though the node should be free Moab will not start the job on that node if the utilization is high. Moab will apply the NODEBUSYSTATEDELAYTIME to push this job in the future and give the node time to have the utilization come back down. When it pushes the job reservation in the future it then overlaps with the following job's reservation. This causes Moab to report that the overlap is due to the following job exceeding it's wallclock limit, even though it hasn't started yet.

One clue that this is happening is in the logs. You should see a log entry stating that the NodeStateDelayNC is some non-zero value, like this:

```
Jul 15 18:29:09 moab01 moab[64116]: extending reservation for job 12345,
NodeStateDelayNC=5, OSDelayCount=0
```

This indicates that there is a problem with one or more nodes and the job reservation is being pushed out the amount of time specified by the NODEBUSYSTATEDELAYTIME.

Unique solution ID: #1114

Author: Ben Roberts

Last update: 2016-07-15 19:24