LA-UR-12-23479
Approved for public release; distribution is unlimited.

Title:	Moab Job Priority
Author(s):	Cunningham, Robert T
Intended for:	Web

Moab Job Priority

(aka. Why is my Job not Running?)

Robert Cunningham

rtc@lanl.gov

HPC Systems Group (HPC-3)

July 19, 2012

Topics

- Introduction
- LANL Moab Configuration
- Job Scheduling Criteria
- Policies
- Prioritiy Calculation
- What to do

Why is a Job *Not* Running?

- The most frequent Moab question we receive
- 93%* of the time, the answer is, “because the cluster is busy” and the job has to wait its turn (*not really, but seems like it)
- Most users want to know why other jobs launched in front of their own
- A subset of those want to know what they can do about it
- Here is an attempt to explain it all, but you can always go to the source: http://www.adaptivecomputing.com/resources/docs/mwm/6-0/5.1.2priorityfactors.php

Our LANL Moab Configuration

- Adaptive Computing (Moab) tells us we have “Moab on Steroids”
  - Poll of other sites show less complex configurations: Ofc. of Science, DoE+DoD Labs, NASA, NCAR, NERSC, etc.
- Gradually exploit many features tailoring job scheduling to suit your needs – “complexity creep”
- Simplify for users when we can
  - project, Account, QoS, class/queue have same name
  - msub -l procs=… (vs. nodes=X:ppn=Y)
  - reduce priority calculation factors

LANL Moab Job Scheduling Criteria

- Policy
  - Queue limits
  - Account limits
  - Reservations
  - Other limits, eg. MAXIJOB, Qtime cap
  - User limits not in use today
  - http://hpc.lanl.gov/policy_lookup
- Priority
  - Simple arithmetic calculation, frequently dominated by Fairshare

Note: Some commands shown in this presentation are off limits – call ICN Consulting to obtain information

Queue-based Limits

- Use Moab's `mdiag` command to obtain information on user, class/queue, Account, QoS
- This example shows several queue-based limits

```
ct-login1> mdiag -c standard
Class/Queue Status

ClassID        Priority Flags        QDef              QOSList* PartitionList        Target Limits

standard              5 ---          ---                  ---  ---                   0.00     ---
  CAPACITY=1090 DEFAULT.FEATURES=[nonviz]  EXCL.FEATURES=vizmem  STATE=active  DEFAULT.WCLIMIT=1:00:00
MAX.WCLIMIT=8:00:00  MAX.PROC=768

ct-login1>
```

Queue-based Limits

- This example shows
  - Max procs for this queue, all users: `MAXPROC`
  - Max procs per user in this queue: `MAX.PROC`

```
tu-fe1> mdiag -c xd
Class/Queue Status

ClassID        Priority Flags        QDef             QOSList* PartitionList        Target Limits

xd                    0 ---          xd                    xd  ---                    0.00  MAXPROC=0:272
STATE=active  ADEF=xd  MAX.WCLIMIT=10:00:00  MAX.PROC=128

tu-fe1>
```

Queue-based Limits

- The Wild West free-for-all
  - Mustang: any user can grab all 38,400 cores for 16 hours
  - Pinto: all 2,848 cores for 16 hours

```
[rtc@mu-fe1 ~]$ mdiag -c
Class/Queue Status

ClassID        Priority Flags        QDef              QOSList* PartitionList        Target Limits

DEFAULT               0 ---          ---                  ---  ---                    0.00     ---
  STATE=active
down                  0 ---          ---                  ---  ---                    0.00     ---
  STATE=active
idle                  0 ---          ---                  ---  ---                    0.00     ---
  STATE=active
jreserved             0 ---          ---                  ---  ---                    0.00     ---
  STATE=active
standard              0 ---          ---                  ---  ---                    0.00     ---
  CAPACITY=38400   STATE=active   DEFAULT.WCLIMIT=1:00:00   MAX.WCLIMIT=16:00:00
```

Account-based Limits

- Another available knob
- This example shows
  - Max procs for this Account, all users: `MAXPROC`

```
tu-fe1> mdiag -a xd
evaluating acct information
Name          Priority        Flags          QDef      QOSList*        PartitionList Target  Limits

xd                   0            -            xd            xd                    -   0.00  MAXPROC=416

tu-fe1>
```

Reservations Can Get in the Way

- Example of a Standing Resevation
  - Weekdays, 7am for 12 hours, 48 nodes
  - Daytime interactive use on a development cluster, batch starts @ 7pm

```
ct-fe1> showres

ReservationID       Type S       Start         End     Duration    N/P    StartTime

283048               Job I     8:55:13    16:55:13     8:00:00   38/608  Fri May 18 19:00:00
interactive.191     User -   5:20:55:13  6:08:55:13    12:00:00   48/768  Thu May 24 07:00:00
interactive.189     User -   3:20:55:13  4:08:55:13    12:00:00   48/768  Tue May 22 07:00:00
interactive.187     User -    -3:04:47     8:55:13    12:00:00   48/768  Fri May 18 07:00:00
interactive.188     User -   2:20:55:13  3:08:55:13    12:00:00   48/768  Mon May 21 07:00:00
interactive.190     User -   4:20:55:13  5:08:55:13    12:00:00   48/768  Wed May 23 07:00:00

6 reservations located

ct-fe1>
```

Other Limits - MAXIJOB

- Maximum number of jobs per user in an Idle state that are *eligible* to launch, accumulating Qtime
- Your hundreds of jobs are still in the queue, but only a few accumulate Qtime priority
- This file is a copy and not always visible – call ICN Consulting if you need info:

```
cj-fe1.lanl.gov> grep MAXIJOB /opt/MOAB/moab.cfg
USERCFG[DEFAULT]          MAXIJOB=10
cj-fe1.lanl.gov>
```

The Crux: Job Priority Calculation

```
Priority 169972 = 100 * (0.0*10) +
100 * (1.0*1 + 0.0*10 + 5.0*100) +
 10 * (1000.*1 + 1099.*10 + 0.0*10)
```

Credentials - 0%
Fairshare - 29.4%
Service - 70.6%

Note the roundoff under the hood (ie. 169,972 vs. 170k).

```
mp-fe1.lanl.gov> mdiag -p
diagnosing job priority information (partition: ALL)

Job                    PRIORITY*    Cred( User)     FS( User:Accnt:  QOS)   Serv(QTime:XFctr:UPrio)
            Weights    --------      100(   10)    100(    1:   10:  100)     10(    1:   10:   10)

7405                     169972      0.0(  0.0)   29.4(  1.0:  0.0:  5.0)   70.6(1000.:1099.:  0.0)
10195                      8879      0.0(  0.0)   12.6(-16.1:-10.2:  1.0)   87.4(1000.:  3.7:  0.0)
6617                       8796      0.0(  0.0)   12.7(-16.1:-10.2:  1.0)   87.3(1000.:  2.9:  0.0)
10313                      7850      0.0(  0.0)   11.0(-15.7: -9.8:  1.0)   89.0(877.0:  1.9:  0.0)
10317                      4151      0.0(  0.0)   17.4(-15.7: -9.8:  1.0)   82.6(511.0:  1.5:  0.0)
10135                     -7425      0.0(  0.0)   63.2( -0.7:  0.0: -1.8)   36.8(1000.:  3.8:  0.0)
10265                     -7545      0.0(  0.0)   63.4( -0.7:  0.0: -1.8)   36.6(1000.:  2.6:  0.0)
10017                     -7582      0.0(  0.0)   63.2( -4.1:  0.0: -1.8)   36.8(1000.:  5.7:  0.0)
10019                     -7582      0.0(  0.0)   63.2( -4.1:  0.0: -1.8)   36.8(1000.:  5.7:  0.0)
```

percent contribution to total priority
100 = 0.0 + 29.4 + 70.6

Priority Factor Terminology

- Job Credentials – preassigned allocation by category (user, group, QoS, Account, queue)
- Fairshare – usage history, incl. walltime accuracy if shown, negative values indicate *overserviced* user/Account/QoS
- Service Level (requested) – includes Qtime and Xfactor
- Fairshare Unit of Measure – shares. Not time.
- Qtime – time spent Idle *and* eligible, capped at 1000 minutes
- Xfactor – eXpansion factor, `(1 + Qtime)/walltime`, favoring short duration jobs

Job Credentials

- Your job has defaults
- Change them with: `msub -A <acct> -q <queue> -l qos=`
  or else use `mjobctl` if the job is already submitted

```
cj-fe1.lanl.gov> checkjob 21320 | head
job 21320

AName: test
State: Running
Creds:  user:samrat  group:samrat  account:s11_casl  class:comp  qos:s11_casl
WallTime:   00:51:00 of 16:00:00
SubmitTime: Mon Jul 16 14:14:15
  (Time Queued  Total: 1:05:09  Eligible: 00:22:28)

StartTime: Mon Jul 16 15:19:24
cj-fe1.lanl.gov>
```

Fairshare

- Tracks usage history against targets per user, group, Account, class/queue, and QoS. See: http://hpc.lanl.gov/moab_fairshare and http://www.adaptivecomputing.com/resources/docs/mwm/6-0/6.3fairshare.php
- We can set interval, depth, decay, and policy
- We use QoS targets (in shares) to deliver allocations
- Incorporates recent history but will not stop job launch from empty queue (ie. negative priority)
- Irrigation water: cannot use large allocation at deadline

Example Fairshare Configuration

- 32-days of history decayed by 5% per day

```
lo1-fe.lanl.gov> mdiag -f -o qos | cut -c 1-94
FairShare Information

Depth: 32 intervals   Interval Length: 1:00:00:00   Decay Rate: 0.95

FS Policy: DEDICATEDPS
System FS Settings:  Target Usage: 0.00    Flags: 0

FSInterval        %     Target       0       1       2       3       4       5       6       7
FSWeight       ------- -------  1.0000  0.9500  0.9025  0.8574  0.8145  0.7738  0.7351  0.6983
TotalUsage      100.00 -------  3380.5 56540.2 47513.9 56665.0 51379.9 60808.9 68446.1 79650.9

QOS
-------------
support           0.00    5.00  ------- ------- ------- ------- ------- ------- ------- -------
lowprio           0.33    5.00  ------- ------- ------- ------- ------- -------    0.38    0.48
standby           0.00    5.00  ------- ------- ------- ------- ------- ------- ------- -------
ASC*             45.30   58.00    47.03   37.38   33.57   34.67   52.06   53.95   50.23   50.59
longq*            0.00    8.00  ------- ------- ------- ------- ------- ------- ------- -------
s11_climate*      0.33    5.77  -------    0.44    0.31    1.51    0.36    0.63    0.26    0.65
s11_ichelp        0.00    3.64  ------- ------- ------- ------- ------- ------- ------- -------
s11_icmr-lobo     0.00    0.13  ------- ------- ------- ------- ------- ------- ------- -------
s11_casl          0.00    3.22  ------- ------- ------- ------- ------- ------- ------- -------
s12_coldatoms     0.00    1.24  ------- ------- ------- ------- ------- ------- ------- -------
w11_a-synuclein*   38.72    0.12    52.83   62.18   65.73   60.46   42.19   43.20   41.24   41.
w11_cellulase     1.67    1.96  ------- ------- ------- ------- ------- ------- ------- -------
```

Additioinal Job Priority Factors

- Res – Requested Job Resources
  - Negative Node count favors narrow jobs
  - Negative Wtime favors short duration jobs

```
lo1-fe.lanl.gov> mdiag -p | head
diagnosing job priority information (partition: ALL)

Job                  PRIORITY*   Cred( User:  QOS)    FS( User:Accnt:  QOS)  Serv(QTime:XFctr:UPrio)   Res( Node:WTime)
            Weights  --------      10(   10:    1)   100(    1:   10:  100)    10(    1:   10:   10)    1( -813:   -1)

292834                 76941     0.0(  0.0:  0.0)  52.0(  1.0:  0.0:  5.4)  34.7(1000.:264.0:  0.0)  13.4(  4.0:10800)

Percent Contribution   --------  0.0(  0.0:  0.0)  52.0(  0.1:  0.0: 51.9)  34.7(  9.5: 25.1:  0.0)  13.4(  3.1: 10.3)

* indicates absolute/relative system prio set on job
lo1-fe.lanl.gov>
```

Strictly QoS Based Priority

- Predetermined Allocation for Priority Groups
  - All but ignore Fairshare and Usage History
  - First come, first served within the QoS (Priority Group)

```
tu-fe1> mdiag -p
diagnosing job priority information (partition: ALL)

Job                  PRIORITY*   Cred( User:  QOS)   FS( User:Accnt:  QOS:  WCA)  Serv(QTime:XFctr:UPrio)   Res( Node:WTime)
             Weights  --------    100(   10:    1)  100(    1:   10:  100: 1000)   100(    1:   10:   10)     1( -813:   -1)

1710248               1016682    92.2(  0.0:10000)  4.6(  1.0:  0.0:  4.4:  0.1)   0.1(  0.0:  1.0:  0.0)   3.1( 40.0:1200.)
1710247                486084    96.8(  0.0:5000.)  0.0(  1.0:  0.0:  0.0:  0.0)   0.2(  1.0:  1.0:  0.0)   2.9(  1.0:14400)
1710249                480387    95.7(  0.0:5000.)  0.0(  1.0:  0.0:  0.0:  0.0)   0.2(  1.0:  1.0:  0.0)   4.0(  8.0:14400)

Percent Contribution  --------    94.2(  0.0: 94.2)  2.3(  0.0:  0.0:  2.1:  0.3)   0.2(  0.0:  0.1:  0.0)   3.3(  1.9:  1.4)

* indicates absolute/relative system prio set on job

tu-fe1>
```

What is a User to Do?

- Backfill: request the smallest walltime, narrowest width that your application can use
  - Your job might squeeze-in between reservations
  - Improves overall scheduler effectiveness
  - (except some platforms favor wide jobs; look before you leap)
- Job Resilience – grab one or two extra nodes to ride through failures within same allocation. Example: http://hpc.lanl.gov/resilient_script
- By all means, be a conscientious citizen

What is a User to Do?

- Something may be broken – give us a call
  - If `/tmp` is full, then `msub` cannot submit a job
  - Moab runs on a separate server in a private network and may lose contact with the cluster
  - It uses a database, collaborates with a resource manager (Torque or SLURM), and sometimes gets in a weird state
- Emergency (deadline, etc.): request DAT or reservation http://hpc.lanl.gov/DAT_process

Question? Answer: ICN Consulting Office

consult@lanl.gov 505-665-4444 Option 3

Backup Material

What Do Others Implement?

- Manual Intervention
- Combination of User FS, Qtime, Job Width
- Strictly Qtime
- Qtime-based, with a hit from fairshare history
- Fairshare based, with Account+class/queue favoring, plus Qtime growth (some day all jobs will run)

Samples of Other Sites

```
Job                    PRIORITY*   Cred( User:Accnt:  QOS:Class)    FS(Accnt:Class)  Serv(QTime)
            Weights    --------      1( 1440: 1440: 1440: 1440)     1(   30:   60)     1(    1)

gaea.2183684          1000000996*   99.6(  0.0:-365.:  0.0:  1.0)   0.1(-13.3:  0.0)   0.3(1520.)
gaea.2183690          1000000995*   99.6(  0.0:-365.:  0.0:  1.0)   0.1(-13.3:  0.0)   0.3(1518.)
gaea.2183680          1000000994*   99.6(  0.0:-365.:  0.0:  1.0)   0.1(-13.3:  0.0)   0.3(1520.)
gaea.2194532              15917     90.5(  0.0:  0.0:  0.0: 10.0)   9.3( 13.9: 17.8)   0.2( 31.0)
gaea.2193946               1977     72.8(  0.0:  0.0:  0.0:  1.0)  21.1( 13.9:  0.0)   6.1(120.0)

-------------------------------------------------------------------------------------

Job                    PRIORITY*   Cred(  QOS)    FS( User)  Serv(QTime:UPrio)   Res( Node)
             Weights   --------      1(    1)   600(    1)     1(  100:    1)    30(   10)


-------------------------------------------------------------------------------------

Job                    PRIORITY*   Serv(QTime)
            Weights    --------      1(    1)

7073987                    5910   100.0(5910.)
7074381                    5897   100.0(5897.)
7240529                    1367   100.0(1367.)
7240561                    1366   100.0(1366.)
7241188                    1349   100.0(1349.)
```

Samples of Other Sites

```
Job                    PRIORITY*    Cred(  QOS)    FS( User)  Serv(QTime:UPrio)    Res( Node)
            Weights    --------       1(    1)    60(    1)     1(    1:    1)     30(    1)

188668                   182369     0.0(  0.0)   9.9(302.1)  90.0(16421:  0.0)    0.0(  1.0)
188722                   182345     0.0(  0.0)   9.9(302.1)  90.0(16419:  0.0)    0.0(  1.0)
7473                      21542     0.0(  0.0)  45.5(-1825)  54.5(13105:  0.0)    0.0(  1.0)
182480                    21064     0.0(  0.0)   2.4(  8.5)   0.3( 65.0:  0.0)   97.3(683.0)
182495                    21049     0.0(  0.0)   2.4(  8.5)   0.2( 50.0:  0.0)   97.3(683.0)

-------------------------------------------------------------------------------------------

Job                    PRIORITY*    Cred(Accnt:Class)     FS(User:Accnt)   Serv(QTime)
            Weights    --------       5(   10:    1)     50(  100:100)      1(    3)

7075841                  35597    14.7(100.0: 50.0)   85.1(  4.9:1.1)    0.2( 19.0)
7075842                  35597    14.7(100.0: 50.0)   85.1(  4.9:1.1)    0.2( 19.0)
7075843                  35597    14.7(100.0: 50.0)   85.1(  4.9:1.1)    0.2( 19.0)
7075844                  35597    14.7(100.0: 50.0)   85.1(  4.9:1.1)    0.2( 19.0)
7075845                  35597    14.7(100.0: 50.0)   85.1(  4.9:1.1)    0.2( 19.0)
7075846                  35597    14.7(100.0: 50.0)   85.1(  4.9:1.1)    0.2( 19.0)
7075847                  35597    14.7(100.0: 50.0)   85.1(  4.9:1.1)    0.2( 19.0)

Percent Contribution   --------    14.7( 14.0:  0.7)   85.1( 69.1:16.0)    0.2(  0.2)
```

Go to the Source

- Fairshare described here: http://www.adaptivecomputing.com/resources/docs/mwm/6-0/6.3fairshare.php