Explaining a Job's Priority

Your job's priority determines when your job will run, relative to all of the other waiting jobs on a cluster. For most clusters and most users, a job's priority value is primarily based upon two factors:

- Fair-Share component: computed according to a fair-share, half-life of decay algorithm. In a nutshell, if you use more cycles than your allocation, you get a negative number; if you use less than you are allocated, you get a positive number. Note that allocations are hierarchical and shared by everyone in your bank. Usage values decay with time - currently a two week period.
- Queue Time: how long your job has been sitting in the idle queue to run. The longer you wait, the more this increases your priority.

There are two other factors that occasionally come into play, QOS (quality of service) and job size, but these will be ignored for now because they don't factor into the usual case.

Note that in the **`|grep`** example commands shown below, the column headers have been shown for clarity. In reality, they will not be seen because the grep command will strip them out.

First, let's assume that your userid is "user4", and you submit a job on a cluster called "hera". The Moab **`showq -i`** command can be used to list information for all waiting jobs, sorted in decreasing system priority. Since that list can be very long, and also include other clusters in the Moab grid, the quickest way for you to see only your jobs on the hera cluster would be like this:

```
% showq -i -p hera | grep user4

eligible jobs----------------------
JOBID                 PRIORITY  XFACTOR  Q  USERNAME    ACCNT  NODES     WCLIMIT     CLASS      SYSTEMQUEUETIME
404067*                 -244137      2.3 no     user4    bdivp     24  1:00:00:00    pbatch   Wed Jan 18 14:29:22
```

In this example, you have only one job #404067 with a priority of -244137. One of the first questions you may ask is how does this compare to all of the other waiting jobs? Use the **`mdiag -p`** command to see the priority sorted list of waiting jobs and how the Fair-Share and Queue Time components are factored in. For example, to see all of the idle jobs with priorities on hera:

```
% mdiag -p -t hera
diagnosing job priority information (partition: hera)

Job                    PRIORITY*   Cred(  QOS)    FS( User)  Serv(QTime:UPrio)
             Weights   --------       1(    1)    85(    1)     1(   15:    1)

404050                  -228291     0.0(  0.0)  90.0(-3022)  10.0(1910.:  0.0)
404979                  -237681     0.0(  0.0)  98.2(-2849)   1.8(299.0:  0.0)
405108                  -240846     0.0(  0.0)  99.5(-2849)   0.5( 88.0:  0.0)
405109                  -240891     0.0(  0.0)  99.5(-2849)   0.5( 85.0:  0.0)
404067                  -244137     0.0(  0.0)  90.9(-3190)   9.1(1802.:  0.0)
404851                  -253727     0.0(  0.0)  94.3(-3177)   5.7(1088.:  0.0)
405229                  -254273     0.0(  0.0)  97.9(-3057)   2.1(372.0:  0.0)
405231                  -254273     0.0(  0.0)  97.9(-3057)   2.1(372.0:  0.0)
405238                  -254423     0.0(  0.0)  98.0(-3057)   2.0(362.0:  0.0)
[ remainder of output truncated ]
```

EXPLANATION:

In this example, your job 404067 has the priority -244137 which is the sum of the Fair-Share (FS) component and Queue Time (QTime) component:

```
      (FS Factor 85 * FS user value -3190) + (QTime factor 15 * QTime value 1802)
```

Note that the fair-share user value is truncated in the above report - to see the actual value, use a command similar to the one shown below:

```
% mdiag -f -v | grep user4
user4             1.96 -------    6.64    0.85 ------- ------- ------- -------   14.01    6.23    2.25 ------- ------- -------
------- -------
        - user4                  14.86      1.00  of   168.00 (user: 20062513.80) (-3190.21)


The user4 fair-share value is negative, which reflects actual usage exceeding allocation.  For example:

% mshare -u user4
Partition rzzeus
USERNAME       ACCOUNT             --------SHARES--------  ---USAGE---
                                   ALLOCATED   NORMALIZED   NORMALIZED
user4          bdivp                     1.0     0.02008%     2.13977%
```

For additional information, please see the "Understanding Job Priority Calculation on Livermore Computing's Moab-Scheduled Machines" located at https://computing.llnl.gov/jobs/crm/moab_jobs.pdf