# Limiting User Greed: Resource Quotas

Integrated over time, fair-share scheduling should ensure that each user gets their appropriate CPU usage (provided they submit sufficient jobs). Over and above this, we want to prevent any one user dominating any host-group at any given time.

[root@bgi projects]# qconf –mrqs

## 1. Old Set

**Prevent any one user dominating the serial queue:**

```
{
  name        C6100-STD-serial.q.rqs
  description  NONE
  enabled     TRUE
  limit       users {*} queues C6100-STD-serial.q to slots=48
      #
      # ..."users {*}" means "each and every user" while "users *"
would
      #   mean "all users together"...
      #
}
```

**Limit total slot-count for each user on the main queues:**

```
{
  name        CSF.q.rqs
  description  NONE
  enabled     TRUE
  limit       users {*} queues R815.q,C6100-STD.q,C6100-STD-ib.q, \
  C6100-FAT.q,C6100-VFAT.q,R410-twoday.q to slots=256
}
```

**Discourage interactive work:**

```
{
  name        C6100-STD-interactive.q.rqs
  description  NONE
  enabled     TRUE
  limit       users {*} queues C6100-STD-interactive.q to slots=4
}
```

**Prevent any one user grabbing more than half of this one:**

```
{
  name        R815.q.rqs
  description  NONE
  enabled     TRUE
  limit       users {*} queues R815.q to slots=256
```

```
}
```

**Since we have so few M610x-hosted GPGPUs, limit to one per user:**

```
{
 name        M610x.rqs
 description  NONE
 enabled     TRUE
 limit       users {*} hosts @M610x-GPU to slots=1
}
```

# 2. New Set

**Limit total usage (sum of all users) on some queues:**

```
{
   name        CSF-Queues-total-users.rqs
   description  NONE
   enabled     TRUE
   limit       users * queues C6100-STD-serial.q to slots=144
   limit       users * queues R410-twoday-interactive.q to slots=12
   limit       users * queues R410-short-interactive.q to slots=12
}
```

**Multiple queues on some hosts, but don't want to overload them:**

```
{
   name        CSF-Hosts-slots.rqs
   description  NONE
   enabled     TRUE
   limit       hosts {@C6100-STD} to slots=12
   limit       hosts {@C6100-FAT} to slots=12
   limit       hosts {@C6100-STD-ib} to slots=12
   limit       hosts {@C6100-STD-test} to slots=12
   limit       hosts {@R815} to slots=32
   limit       hosts {@R410-twoday} to slots=12
   limit       hosts {@R410-short} to slots=12
}
```

**Don't want any individual to hog the precious IB-connected Intel nodes:**

```
{
   name        CSF-PEs-each-user.rqs
   description  NONE
   enabled     TRUE
   limit       users {*} pes orte-12-ib.pe to slots=96
}
```

**Limit MACE use of the non-IB Intel nodes as they contributed only AMD:**

```
{
   name        CSF-Usersets.rqs
   description  NONE
```

```
   enabled     TRUE
   limit       users @mace01.userset queues C6100-STD.q to slots=36
```

**Limit each user's greed on each (well, most) queues:**

```
{
   name        CSF-Queues-each-user.rqs
   description  NONE
   enabled     TRUE
   limit       users {*} queues C6100-FAT.q to slots=36
   limit       users {*} queues C6100-STD-serial.q to slots=36
   limit       users {*} queues C6100-STD-interactive.q to slots=4
   limit       users {*} queues R815.q to slots=256
   limit       users {*} queues R815.q,C6100-STD.q,C6100-STD-ib.q, \
   C6100-FAT.q,C6100-VFAT.q,R410-twoday.q to slots=256
   limit       users {*} queues M610x-GPU.q,M610x-GPU-interactive.q to
slots=3
}
```

**Limit total usage (sum of users) on some PE/Queue combos:**

```
{
   name        CSF-PEs-total-users.rqs
   description  NONE
   enabled     TRUE
## limit       users * pes orte.pe,orte-12.pe to slots=550
   limit       users * pes orte.pe,orte-12.pe queues C6100-STD.q to
slots=96
   #
   # ...above, changed one t'other...
   #
   limit       users * pes smp.pe queues C6100-STD.q to slots=440
## limit       users * pes fluent-smp.pe queues C6100-STD.q to
slots=48
    #
    # ...above, replaced by mace.userset quota...
}
```