Yeti Operating & Executive Committees Meeting
Wednesday, April 15, 2015

Attendees: Greg Bryan (Chair), Rob Lane (Presenter), Kathryn Johnston (Chair of SRCPAC), Tian Zheng (Statistics), Marc Spiegelman (APAM), Brent Stockwell (Chemistry), Chris Marianetti (APAM), Harmen Bussemaker (Biological Sciences), Ryan Abernathey (LDEO), Derrick Lim (CCLS), Evan Riehl (Economics, for Bentley MacLeod), Patrick de Perio (XENON/Astrophysics), Roslyn Hui (ZMBBI)

Staff: Alan Crosswell, Halayn Hescock, George Garrett, Marley Bauce, Dali Plavsic, Alex Bergier

*~~Per Ryan Abernathey's request, Rob Lane will send PowerPoint to members afterwards.~~ Marley sent PowerPoint to group already.

Usage Report

In March, we did 60,000+ core days, with a theoretical maximum of 80,000. Finally starting to hit capacity from time-to-time (though not at all times), which results in many triggered mechanisms meant to control how system behaves. We must thus think about how we need for the system to behave when at maximum capacity.

Rob expects the "Other Users" group to expand significantly over the next few months.

In 2015, over 164 different users ran at least one job on the system. 24 different groups.

1.2 Million jobs since January 2015, averaging 1.5 cores per job, with an average will time of 0.1 days and average job size 1.2 nodes, average wait time 2.2 hours, and average execution time 4.5 hours.

Tian Zheng questions why wall and execution times are not similar; Rob replies that it must account for queue wait time.

7 queues employed in 2015, with a newfound "*batch2a*" to deal with multi-node jobs. As you progress through queues/batches, the jobs get larger and longer. Ryan asks for clarification on how jobs are measured; Rob replies size is measured by memory. Each queue has different average cores, wait and wall time (see PPT slide for specific statistics).

Interlong is a special queue set-up on request, which will be returned to at a later point.

Rob then reviews queue statistics for March 18 – March 31. Since batch2a was very new, this more recent time period gives a better representation of the typical distribution of jobs to the various queues.

Brent notes that his lab has had difficulty getting interactive jobs longer than four hours; that the request would have to wait indefinitely. Rob replies that the maximum wall time for interactive jobs is precisely four hours. Average wait time for *interlong* queue is 1.3 hours. Rob will investigate this issue, and notes that the cluster is now maxing-out for the first time ever.

Harmen adds that his lab needs interactive time all day. Ryan asks whether Harmen's researchers fully utilize every single moment of time; Harmen agrees, and Brent agrees as well. Harmen wants an increase in the wall time of interactive jobs.

Chris asks Harmen if he actually requires Infiniband time, and his type of parallelization. Harmen replies he does primarily individual jobs, but with some multi-threading via Java.

George asks Brent whether he needs interactive time wherein they must watch all progress, or whether they can submit a batch job to Yeti, not watch it, then never need parallel processing. George and Brent will discuss needs after the meeting.

Harmen requests that the wall time is increased to at least eight hours, perhaps even twelve hours. Ryan is concerned whether the use of the system is used or whether user is simply "parked" there to take-up space, and whether other users are thus disadvantaged. Ryan reiterates that people can better utilize batch jobs so as to make the best use of a shared system.

Alan notes that the best way to utilize the system is batch time versus think time. Kathryn comments that she has not seen long interactive queues in other systems; Greg replies that typically these interactive queues are very short.

Greg believes there is a benefit of an interactive mode, but we must strike a balance that does not allow people to take an unfair portion of queueing time.

Free Tier, Education Tier, Publications

Free Tier includes 6 users. Any CU affiliate can use the free tier if they receive a researcher's endorsement.

There is currently an 8 core hard limit for free users; if they have a current job taking up 8 cores, they cannot use any more cores with another job. Marc clarifies that this limit is per-person, not per-group. 16:32 soft:hard limit was approved at previous Operating Committee meeting.

Rob asks for approval to revert to previous ratio of 16:32. Marc asks for a longer-term usage report to make an informed decision.

0 Education Tier users, but will support at least one class this Fall 2015 semester. Encourages members to notify professors of this opportunity.

Rob and Kathryn remind of upcoming RCEC meeting on May 15th, and the need for Yeti publications tally in order to rationalize future support. Marley will remind members to send publications.

Operations Overview

Everyone on the system is part of a group, each group with a target share based on the number of servers purchased.

"Recent" usage is tracked; at previous Operating Committee meetings, we agreed to only look back one week's time in order to calculate usage, then compares usage to the "fair share" target. Usage is measured in core hours. If actual usage exceeds fair share usage, jobs belonging to that user and group will have their priority lowered.

Brent asks why we only look back one week to measure actual usage, and requests a longer frame for measurement. Rob explains there is no reason for this timeframe, although there is a cost of collecting and retaining data. Rob notes that one month is the maximum amount of time that could be met, although the previous Operating Committee meeting agreed upon one week. Ryan notes that the current algorithm would never get him to his target because usage varies considerably based on the week. Rob notes that this would advantage Ryan because of the fluctuation of usage, and the drop in usage for some weeks.

Ryan reiterates to ask whether the weeklong algorithm is achieving the targets of balancing actual usage with fair share. Greg notes that at the Fall 2015 Operating Committee, we produce actual usage data over a longer period of time. Tian is curious about the difference in wait and wall times segmented by individual groups, which will allow her to answer to her users whether her wait time is typical or atypical, as well as so she can recommend to her department to buy more hardware.

Rob asks whether any groups think their system usage data is confidential, about whether it can be made available to the group. Nobody in the group sees a problem with this; Ryan notes that any "ambitious" user could already find out this information.

Tian asks that we not share the raw data with the entire group because of group-identifying information.

Rob identifies the original configuration for job queues, but does not have any recommendations for changes.

Batch2a is a workaround for MPI jobs.

Currently, no user can go above 512 cores running; disagreements must be raised. There is a 256 soft and 512 hard limit on individual jobs. Ryan gains clarification that this is a limit by user, not by group.

Groups have priority access to the systems that they purchased; if that system becomes available and someone in that group has a job in the queue and has priority, they get added to that system. Wall time must be below 12 hours in order to run on another group's system. Chris asks whether a user in this workflow will need their specific node, or just an available node of that same type; Rob answers that users must need *that specific node that they own*, not a node that someone else's node, for that other owner will thereby lose that guarantee.

Alan expresses concern that this arrangement requires some nodes to remain idle; Kathryn comments that this is a necessity in order to allow for sharing of a resource. Ryan says that this arrangement is an impediment to queue most efficiently.

Chris adds that the arrangement should be edited to be that all users need is a free node of the same variety as the one that their group purchased.

Greg comments that the RCS group will investigate the feasibility of instituting this model, though Rob disagrees.

Alan asks if Rob has data to examine current Guaranteed Access policy to measure efficacy.

Operations Discussion

Greg thanks everyone for feedback on improvements, which ranges from need for additional software, to clarification of policies and rules. Much feedback does not need group discussion, and so will be replied to on a personal basis by individual RCS staff.

A few bits of feedback require Operating Committee discussion. Firstly, the Job Queue for MPI jobs. Most Infiniband jobs are run on a node-level, which causes problems when single core is being used by another job. Greg suggests creating a parallel Infiniband job queue that feeds exclusively into that cluster. Rob comments that this is an inevitability to segment nodes. Other queues would thereby lose access to all Infiniband nodes, with possible exceptions (very short jobs of 1-2 hours; pre-emptable jobs). Ryan clarifies that anyone can use the new queue.

Rob proposes this policy to include a maximum wall time of 36 hours, and access to full nodes only. No memory limit (which becomes irrelevant when not sharing); a user may run no more than 8 jobs run all at once.

Kathryn asks for feedback on the Short Queue prospect. Rob further clarifies that the system adds for reservations. Ryan comments that we need to envision a scenario for a queueing system that accommodates very complex jobs.

Chris notes that we paid a premium for Infiniband nodes. He is fine with the new policy provided that current jobs from Infiniband purchasers do not have their jobs interrupted.

Greg outlines other option for adding GPU, and requests to hear feedback via email after the meeting as to whether members want this.

Rob adds that we will soon add an *Admin* queue for administrator use only.

We can also add Special Request Queues on a short-term basis for researchers with special needs that fall outside of the current queue structure, especially those with longer wall times.

Greg identifies all "Requested Topics" that will be followed-up on. (See PowerPoint for specific information.)

Rob will email group contacts to set up optional meetings to "meet and greet" to explain policies, outline structures, etc. Members can email hpc-support@columbia.edu to also request this meeting. Harmen wants a formal event scheduled to give a large demo. Rob has already explored this as educational courses in conjunction with the Libraries. Greg notes that the Education Subcommittee is covering this topic.

Executive Session

Faculty Attendees: Chris Marianetti, Harmen Bussemaker, Ryan Abernathey, Tian Zheng

Greg describes format of EC to make tentative and practical decisions about queue restricting based on Operating Committee discussion, which will be tried out prior to next Yeti meeting in Fall 2015.

Primary item for discussion is to remove *batch2a* and replace with *Infiniband* queue, and potential *GPU* queue. (There has been a fair amount of GPU usage thus far.) The detriment of adding these queues is balkanizing the system. However, this system would incentivize the use of GPU.

Ryan asks "philosophical" question: What is the definition of a queue, and why do we have so many? Rob replies that this is the way we apply different rules to different kinds of jobs, especially to differentiate how we handle very large jobs.

Chris adds that serial and MPI queues don't need to be on the same system. Ryan replies that he is confused how creating a queue is different from creating a rule; Rob and Greg reply this is the same thing.

Batch 1-4 and *Interactive* are automated: the system figures out where they go. For *Infiniband* and *GPU*, the user must actively decide to utilize this queue and where they want their job to go.

Tian asks question about user training and behavior reporting: Despite all fair share rules, when new users sign-up is there basic training so they know how to responsibly use a shared resource. Should there be a requirement for training? (Equates to Human Subjects research and the university requiring basic knowledge.)

Ryan admits to having spent a couple of weeks watching queues and usage patterns, some of which he cannot understand. For example, some users will request one job on multiple nodes with very little memory. Suggests departments and groups establish Standard Operating Procedures for how to run the system fairly and efficiently; suggests that part of the problem is ignorant users who do not know they are disadvantaging educated users.

Greg replies by urging Executive Committee to have their departments create a best practices manual for the usage of the system. Halayn inquires whether we can run reports to show different departments' usage.

Rob asks what conditions must be in place to distribute job logs publicly.

Greg identifies that if people notice jobs unfairly hogging the system, to not contact the suspected users but instead the HPC staff to arbitrate fair usage.

Chris suggests a tweak to increase Max User Run on *Infiniband* to 10. Ryan wants to eliminate all unnecessary constraints.

No objections to new proposal.

Resolutions

- Create Infiniband queue with 36 hour wall time and Max User Run = 10. Allow submission of short serial jobs with wall time up to 2 hours to the Infiniband nodes.
- Create GPU queue.
- Create Admin queue.
- Interlong queue temporarily reauthorized. RCS to work with Stockwell group to determine whether it can be eliminated by use of batch processing.
- Remove stack limits.
- Enable job submission from compute nodes (requires rebuild of Torque).
- Distribution of job logs to participating groups is acceptable.

If this does not work, we will revisit and devise a new policy. RCS will schedule a cluster downtime on or around May 26 to implement these configuration changes.