

## Project 3: Running Jobs with a Portable Batch System (PBS)

- Please read this document carefully. It might save you a significant amount of confusion.
- **We will time your submission code for grading purposes from the 200SD machines!**
- **Introduction**
  - *What is a batch queue system?*
    - i. A batch queue system (and a scheduler) attempts to load-balance a set of machines so that the computational resources are balanced efficiently. When you submit a job to the batch queue, it is scheduled for execution on one of the cluster nodes. This scheduling process takes into account parameters such as machine load and available number of cores in making a decision.
  - *Why should I care?*
    - i. Last semester, students had problems effectively timing OpenMP code when many people were trying to run multi-threaded jobs upon the the machines in 200SD. Basically, the machines were becoming sluggish due to overloading (especially close to the submission deadline) and made it difficult to obtain good data. This semester, we've implemented a PBS server in hopes of alleviating some of these problems.
- **Usage**
  - **The batch queue has been implemented on the machines in 330 Soda (the so-called "hive" machines). These machines are numbered `hive1.cs.berkeley.edu` to `hive28.cs.berkeley.edu`.**
  - To use the batch queue, you must follow several steps:
    - i. **Log into `hive1`, `hive27`, or `hive26`.** These nodes are login nodes, and are the only ones on the network from which you can submit jobs to the PBS.
    - ii. **Type "make clean", then recompile your code using the project Makefile.** This step is critical, as the machines in 200SD are OSX and the hive machines are currently running Linux. Because of this, **code compiled on the 200SD machines will not work on hive and vice versa!!**
    - iii. **Edit your qsub script to take into account the proper number of threads and the correct name of your executable binary.** Here is an example qsub script:

```

#!/bin/bash
#PBS -N CS61C
#PBS -V
# CHANGE PPN to set number of cores to use...
#PBS -l nodes=1:ppn=8
#PBS -q batch

cd $PBS_O_WORKDIR
# workaround to fix a thread affinity problem in Linux
export GOTOBLAS_MAIN_FREE=1
# Change OMP_NUM_THREADS to set number of threads
export OMP_NUM_THREADS=8
# name of the file to execute
./bench-openmp

```

This script tells the PBS server to schedule a job that runs the program “./bench-openmp” with 8 threads. **You need to change both the PPN value and OMP\_NUM\_THREADS for the program to run with the proper number of threads!**

- iv. **Submit your job to the PBS via the qsub command.** This should look like this:

```
qsub myQsubScript.sh
```

where “myQsubScript.sh” is the name of the script file you created in the previous step. Qsub will report the job number followed by the hostname of the PBS server machine. For example, “142.hive1.cs.berkeley.edu” means that job number 142 has been submitted to the PBS server running on hive1.cs.berkeley.edu. You can check on the status of jobs in the queue by looking for your jobs in the output of the “qstat” command:

```
~/proj3/matmulProjectNew $ qstat
```

Job id	Name	User	Time Use	S	Queue
143.hive1	CS61C	cs61c-tf	00:00:00	C	batch
144.hive1	CS61C	cs61c-tf		0	R

This output indicates that the job id 143.hive1 was submitted by User “cs61c-tf” and is now complete “C”. Other useful state indicators are “Q” (queued) and “R” (running). So, job id 144.hive1 is still running.

Once your job is complete, two files will be written into the directory from which you called “qsub”. These files are: CS61C.e[num] and

CS61C.o[num], where “[num]” is the job number. These text files contain a batch program’s error and standard output, respectively. In the case of the above job, the error file will be “CS61C.e143” and the standard output file will be “CS61Co143”. You will be able to find timing data for your matrix multiplication runs in the standard output file.

- **Additional Comments**

- **Make sure to use the updated Makefile to compile code for the hive machines.** This fixes a few bugs that showed up when we moved to a Linux platform.