CS 598: Parallel Migratable Objects

Fall 2013

Due date: September 3, 10pm

This MP is an introductory assignment to learn how to build Charm++ and run Charm++ programs. You will also learn how to use `Projections` performance analysis and visualization tool.

For learning Charm++ programming and concepts, refer to Charm++ Manual and the Charm++ Webpage.

Step 1: To run Charm++ programs, you need to have a Charm++ build installed. If you want to build your own installation, follow the build instructions here. Both on EWS and Taub Caumpus cluster, there are two installations of Charm++ that can be used instead of building Charm++ yourself. Path is the same in both machines.

- /home/acun2/charm/net-linux-x86_64
- /home/acun2/charm/net-linux-x86_64-smp

For this MP0, you only need to use EWS. Taub has a batch system to submit jobs which will be required to use later, however for this MP it is optional.
For EWS login instructions go to EWS FAQ page.
For Taub login instructions go to Taub User Guide.

Step 2: An svn repository has been created for the class here at EWS, where you can find the Charm++ code for MP0.

- Checkout the code from the repository by:
  svn co https://subversion.ews.illinois.edu/svn/fa13-cs598lvk/netID
  where netID is your actual netID
- A directory named mp0 should be created, go into that directory:
  cd netID/mp0
- Analyze the code and run the program with *make test* command. It should print "Total number of primes within the range [0 - 100000] is 9591." In the program Master chare fires k chares, i'th Worker chare fired is responsible

for computing number of primes between [i*M ..(i+1)*M]. They return the counts to the Master chare. M and k are command line arguments. You can change these numbers in the `Makefile` and experiment.

To run a program in Taub cluster you need to submit batch scripts, sample running script is added to your SVN repositories.

Step 3: After running the program, you will see `Projections` logs created in the same directory. Anaylze them using the `Projections` tool, take a snapshot of the timeline of the program and turn it in. `Projections` can be downloaded from here and the `Projections` manual can be found here.

Submission:
Submission will be done to SVN repositories.

- For each file F you create, that you want to check in, do:
  svn add F
  and frequently (after you have modified F, and have the next better version) do:
  svn ci F
- There will be a penalty for late submissions.