Proactive on FWGrid (Getting Started)

Follow the steps given below and check if everything works well. If it doesn't work the way it should let me know at (snandy@cs.ucsd.edu).

NEW: How to launch the same job simultaneously on multiple nodes. (May 12, 2005)

1. Create a directory called pro2 in your $HOME

2. Transfer all the files from */home/cse160/hw3/parallel/* to this directory (pro2).

3. *cd* to pro2 and give executable permission to all .sh files (*chmod a+x *.sh*).

4. Launch "*qsub launch-all.sh*"

5. Check using *qstat* if you job is over and also check your directory for the output file which should have two outputs since the script runs it on 2 nodes (there will be 2 output files created *.o*jobid* and *.p*jobid* – check the first one). This step might take a while. To ensure that you are seeing the complete output *qstat* and ensure that your job is no longer in the queue.

Once you have succeeded with this try to understand the script to see how you can use it for your purpose. The following things will come useful:

1. Line 5: -p2 mpi 2 (the two denotes the number of servers launched)

2. Lines 48, 59 - the working directory is defined as $HOME/pro2/ - change it for your need.

3. Line 62: the time the code sleeps before launching the client. You might have to change it based on your needs.

4. Lines 68-70 - this part accesses each node's URL where a server is running and launches a client for that. So we are actually launching 2 Clients in this example. Ideally you should have one client and pass on all server URLs to it. So you have to modify this part according to your program needs too.

5. **It is very important that you make sure that both your server and client quits on job completion. The present version of Hello and HelloClient has code which shows you how you can achieve this.**

~*~

1. For this homework onwards you have to use the node fwgrid-compute-server-0.ucsd.edu. **You are not allowed to use the previous node that you used for the earlier homeworks.**

2. Create a directory in your workspace (lets call it *mydir*) and cd to it.
3. Copy all the files provided in /home/cse160/hw3/ to your directory *mydir*. (Yu should have six files in the directory at the end of this step).
4. Compile the two java files Hello.java and HelloClient.java.
5. Submit the Hello progam as a job to FWGrid. To do this issue the command

qsub hw3-example-server.sh

qsub is the command you will use to submit jobs in FWGrid. Once you do this you should see a line like:

your job 2851 ("hw3-example-server.sh") has been submitted where 2851 denotes you *job_id*

6. List the files in *mydir* and you should see a file called hw3-example-server.sh.o*job_id i.e. hw3-example-server.sh.o2851* in our example.
7. Now go through the file and you should see it end with a line that looks something like

Success at binding url **//fwgrid-compute-6-14.local/**Hello

The highlighted portion shown above is the actual name of the node where your job got run. You can list all the nodes and the jobs running in the by running *qsub –f*. If you do this then you should be able to see your corresponding job running in the given machine.

8. Now launch the client by executing the script *hw3-example-client.sh* with the node name as the command line argument i.e. for our example, after changing the permissions of the script to executable run the command

./hw3-example-client.sh fwgrid-compute-6-14.local

9. Your Client should run properly and give an output that looks something like

The message is: Hello world at 05/05/2005 19:17:43 from node : //fwgrid-compute-6-14.local/Node1500939074

10. Kill the client by typing Ctrl+C
11. Kill the server by typing the command *qdel job_id* i.e. in our example we would type *qdel 2851*
12. Now change the client and server programs to pass on an array to the server's sayHello method. The server should then add the values of all the elements of the array and send it back as part of the return message. This will give you a taste of how to send data to and fro between two nodes.