Progress report

M2 Kakeru Ishii

November 6, 2019

1 Rewrote all code depending JOGL to depend LWJGL

Recently I had worked to make my renderer working in headless environment. To make it works, my renderer can't create OpenGL context in straight forward way. Basic linux OpenGL context is typically strongly bound to X window system. It may make it harder for my renderer to work in server environment like TSUBAME.

My renderer should use EGL to create headless OpenGL context for this usage. I have used JOGL(OpenGL java wrapper) in my renderer code, but it was very hard to use EGL over this library. I switched base library JOGL to LWJGL.

2 Multi process experiment in TSUBAME

Most of multi process experiments in TSUBAME might use MPI or the other common parallel computing framework. My renderer need to be managed manually because I don't use these such kind of frameworks.

As an introduction for TSUBAME job, I need to write batch script(Figure 1) and execution code(Figure 2). The generated file located in environment variable $PE_HOSTFILE includes hostnames of slave nodes. Then qrsh command perform a sort of ssh operation for slave nodes.

3 Sync multiple server processes

I need to run 2 type of nodes on TSUBAME cluster. Then I need to sync server processes because merge node should run after running content node. And each node should know all of host names they need to connect.

I will make use of shared mounted file system for this. Each node will write their hostname to specific file to notify their hostname to the other nodes.

1. Create a file and write own hostname to shared directory
2. Wait all nodes to complete 1st step
3. Execute all of content nodes with arguments specified by node index
4. Wait all content node to be started
5. Execute all merge nodes

References

```
#!/bin/sh
#$ -cwd
# Node type and count for this job
#$ -l s_gpu=2
# Max time for this job
#$ -l h_rt=00:30:00
#$ -V

for n in $(awk '{print $1}' $PE_HOSTFILE)
do
    qrsh -inherit $n /path/to/program &
done
wait
```

Figure 1: Batch job code for TSUBAME to invoke multi process program manually

```
qsub -g <Group name in tsubame> task.sh
```

Figure 2: Shell code to put batch job in TSUBAME