GRAM5 validation test

Authors:
Jeffrey Dost*
Igor Sfiligoi*
*UCSD

Date:
August 12th, 2009

Abstract

The purpose of this test was to gain experience with GRAM5. We run the tests against a small scale test CE and watch the load there.

Test Description

Setup

One test CE running the Condor job manager.
GRAM alpha2 and OSG 1.2 were used.

Procedure

Submit 100 sleep jobs at one time, and observe the CE load for the duration of 1 hour. Each job was set to sleep for 300 seconds (5 min). This test was performed with GRAM5, and also with GRAM2 to compare. We observed top to collect the following information every 5 minutes:

- Average load, for the 1, 5, and 15 minute intervals
- CPU Percentages: us (user), sy (system), id (idle), and wa (I/O wait)
- Number of Processes running

When running GRAM5, we had to set GRIDMANAGER_MAX_JOBMANAGERS_PER_RESOURCE = 100 in the condor_config file on the submit machine. Otherwise, only 10 jobs showed up in the condor queue at any given time. This is currently a known limitation with Condor-G when running GRAM5 that the condor team is aware of. Also, when running GRAM5, as instructed, we set ENABLE_GRID_MONITOR = FALSE in the config file as well.

Steps

Trial 1: GRAM2 default

This was run with the default GRAM2 settings.
ENABLE_GRID_MONITOR = TRUE
GRIDMANAGER_MAX_JOBMANAGERS_PER_RESOURCE = 10

Trial 2: GRAM5
Next, we ran the test with the GRAM5 installed.
ENABLE_GRID_MONITOR = FALSE
GRIDMANAGER_MAX_JOBMANAGERS_PER_RESOURCE = 100

Trial 3: GRAM5 with monitor
Third, we tried running GRAM5 with the job monitor enabled.
ENABLE_GRID_MONITOR = TRUE
GRIDMANAGER_MAX_JOBMANAGERS_PER_RESOURCE = 100

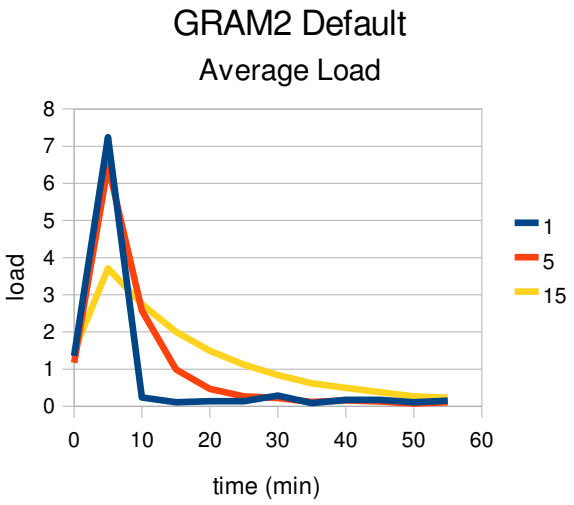
Trial 4: GRAM2 no monitor
Next, we tried GRAM2 with the grid monitor disabled.
ENABLE_GRID_MONITOR = FALSE
GRIDMANAGER_MAX_JOBMANAGERS_PER_RESOURCE = 100

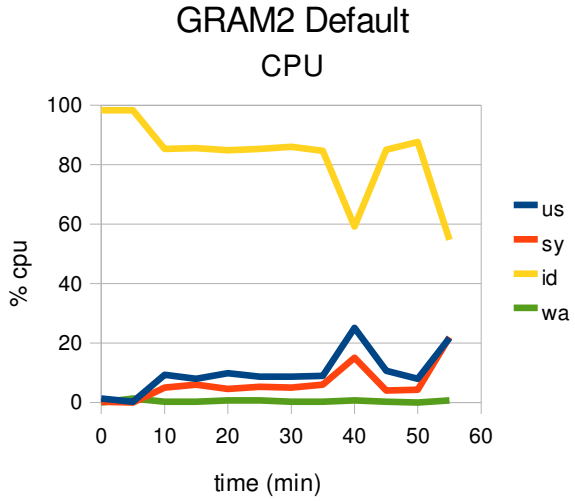
Trial5: GRAM2 max = 100
Finally, we ran the jobs with the grid monitor back on, but also kept the maximum job manager value to 100.
ENABLE_GRID_MONITOR = TRUE
GRIDMANAGER_MAX_JOBMANAGERS_PER_RESOURCE = 100

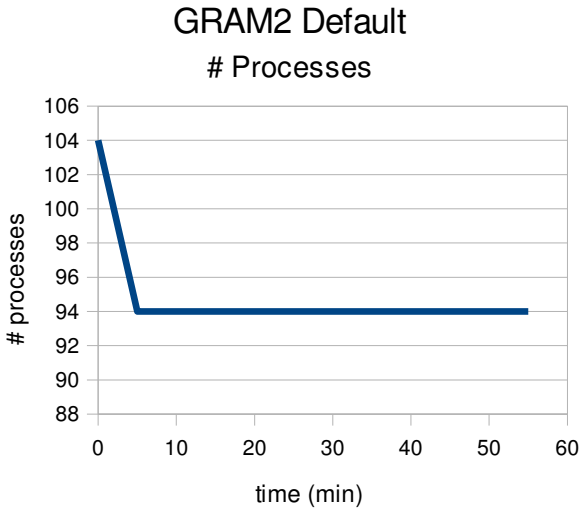
Results

Trial 1: GRAM2 default

ENABLE_GRID_MONITOR = TRUE
GRIDMANAGER_MAX_JOBMANAGERS_PER_RESOURCE = 10

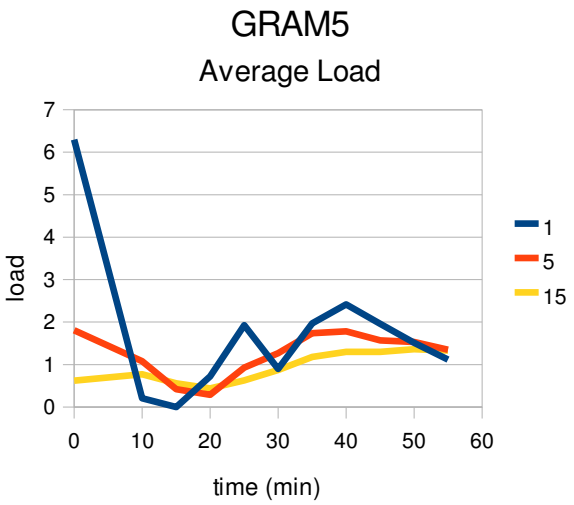


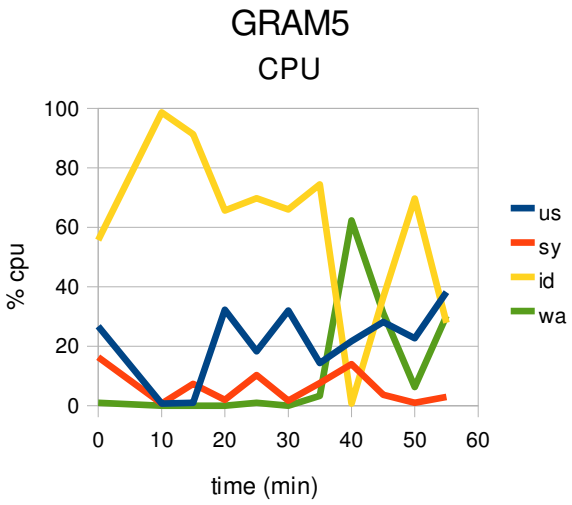


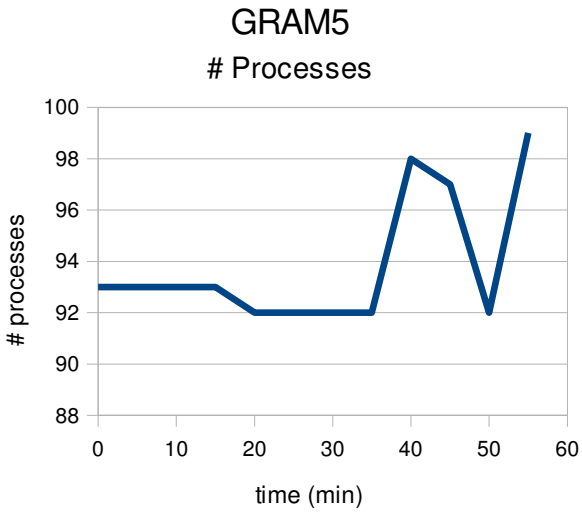


Trial 2: GRAM5

ENABLE_GRID_MONITOR = FALSE
GRIDMANAGER_MAX_JOBMANAGERS_PER_RESOURCE = 100







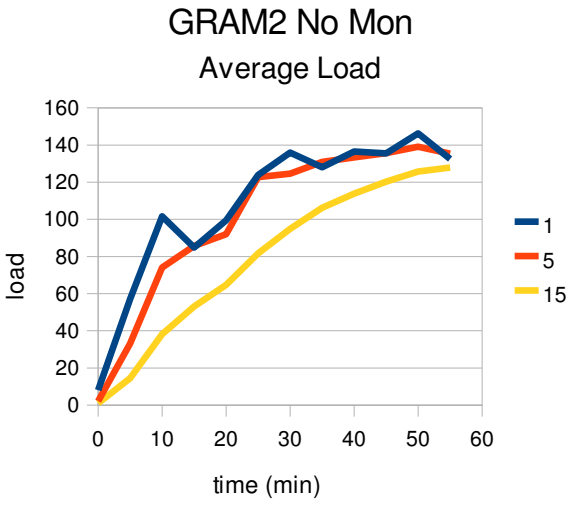
Trial 3: GRAM5 with monitor

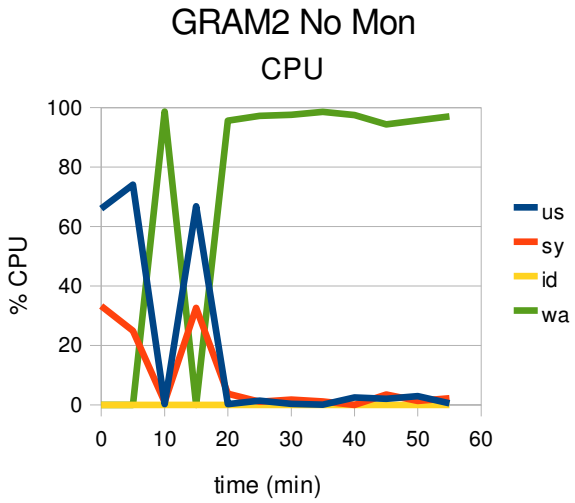
ENABLE_GRID_MONITOR = TRUE
GRIDMANAGER_MAX_JOBMANAGERS_PER_RESOURCE = 100

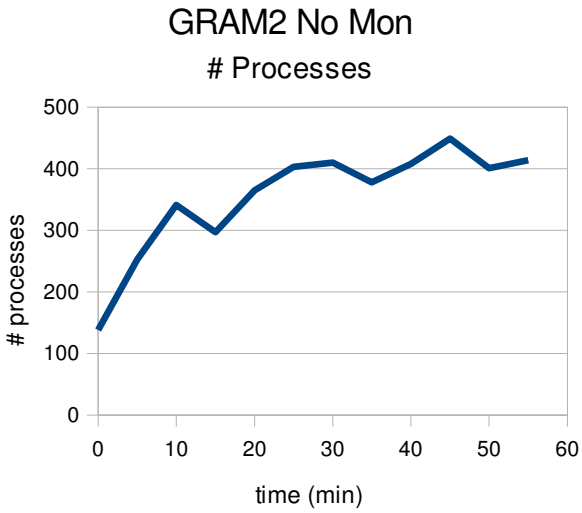
We tried running one job using GRAM5 with the grid monitor enabled and the job successfully ran on the CE but once it finished, it never reported back to the submit machine. The status of the job on the submit machine remained "running" indefinitely in the condor queue and had to be removed manually. We assume this is related to the currently known Condor-G problem.

Trial 4: GRAM2 no monitor

ENABLE_GRID_MONITOR = FALSE
GRIDMANAGER_MAX_JOBMANAGERS_PER_RESOURCE = 100

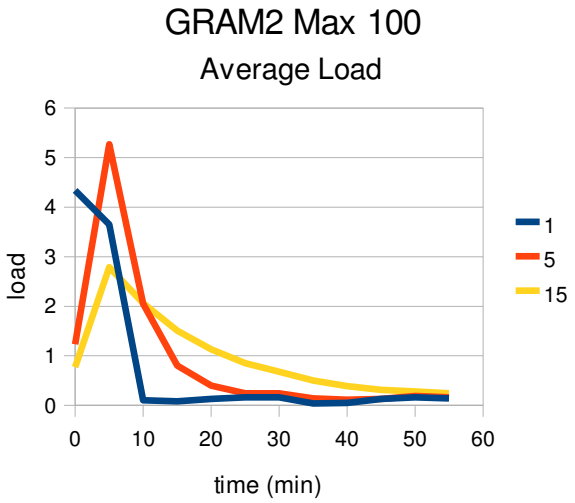


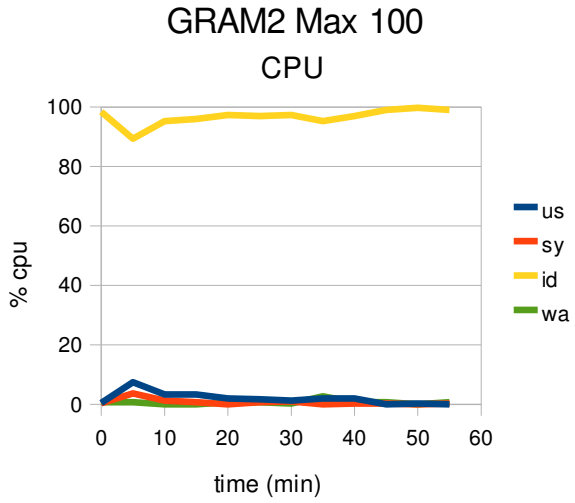


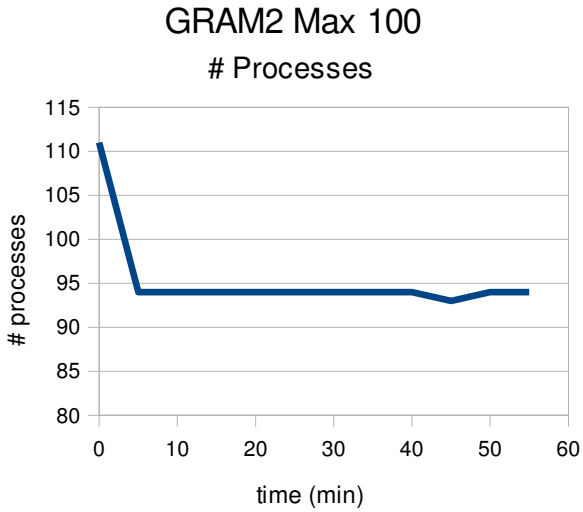


Trial5: GRAM2 max = 100

ENABLE_GRID_MONITOR = TRUE
GRIDMANAGER_MAX_JOBMANAGERS_PER_RESOURCE = 100







Here we were surprised to see that there was no appreciable difference between these results and the results of Trial 1 when the jobmanagers were set to 10.