GT 5.0.1 GRAM5: Quality Profile

Table of Contents
1. Test coverage reports ........................................................................................................................ 1
2. Code analysis reports ....................................................................................................................... 1
3. Outstanding bugs ............................................................................................................................. 1
4. Bug Fixes ....................................................................................................................................... 3
5. GRAM5 Throughput Tests ................................................................................................................. 5
6. GRAM5 Condor-G Tests ................................................................................................................... 7

<titleabbrev>Quality Profile</titleabbrev>

1. Test coverage reports

   • GT 5.0.1 Test Coverage Status Report

2. Code analysis reports

   • No code analysis reports have been generated at this time.

3. Outstanding bugs

   • GRAM-105: Held Condor jobs should be reported as SUSPENDED
   • GRAM-136: Error message not precise when disk quota is exceeded
   • GRAM-138: GRAM5 job manager uses a lot of memory when SEG is pointed to incorrect log path
   • GRAM-139: SEG may deadlock with threads
   • GRAM-146: GRAM5 Usage stats ignores GLOBUS_USAGE_TARGETS environment
   • Bug 5621: gram2 credential refresh problems in 4.0.5
   • Bug 1934: Gatekeeper's syslog output cannot be controlled
   • Bug 2739: Gatekeeper AuthZ/Gridmap Callout result logging
   • Bug 2741: catching SIGSEGV if dynamic loading of authorization modules fails

---

1 http://jira.globus.org/browse/GRAM-105
2 http://jira.globus.org/browse/GRAM-136
3 http://jira.globus.org/browse/GRAM-138
4 http://jira.globus.org/browse/GRAM-139
5 http://jira.globus.org/browse/GRAM-146
6 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5621
7 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=1934
8 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=2739
9 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=2741
• **Bug 4199:** Patch pre-WS GRAM to use individual condor logs for jobs

• **Bug 3795:** jobmanager perl modules issues

• **Bug 4235:** globus-job-manager doesn’t exit if the job fails.

• **Bug 4730:** MPI Jobs using Globus LSF in HP XC Cluster....

• **Bug 4747:** Need evaluation of patch to JobManager.pm

• **Bug 4779:** gram GT2 log files: timestamps are not ISO 8601 compatible

• **Bug 5143:** DONE state never reported for Condor jobs when using Condor-G grid monitor

• **Bug 5429:** stdin is lost when jobtype=multiple with jobmanager-lsf

• **Bug 5554:** GRAM2 4.0.5 setup-globus-job-manager-fork.pl silent failure

• **Bug 5556:** Audit directory setup instructions are insecure

• **Bug 5775:** gram status of old jobs incorrect on some lsf systems

• **Bug 6184:** pbs.pm jobmanager fails jobs on qstat failure

• **Bug 6337:** Cannot configure globus to use different certificate path than default

• **Bug 6703:** PBS scheduler adapter assumes that Globus is installed in the same location on the headnode of a cluster and on the work nodes.

• **Bug 6768:** Held Condor jobs should be reported as SUSPENDED by GRAM

• **Bug 6815:** Support standard install locations for globus-gram-protocol

• **Bug 6819:** Missing metadata in globus-scheduler-event-generator

• **Bug 6820:** Support standard install locations for globus-gatekeeper

• **Bug 6821:** Support standard install locations for globus-gatekeeper-setup

• **Bug 6822:** Support standard install locations for globus-gram-job-manager-scripts

11 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=4199
12 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=3795
13 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=4235
14 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=4730
15 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=4747
16 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=4779
17 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5143
18 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5429
19 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5554
20 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5556
21 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5775
22 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6184
23 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6337
24 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6703
25 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6768
26 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6815
27 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6819
28 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6820
29 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6821
30 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6822
• **Bug 6823:** Support standard install locations for globus-gram-job-manager

• **Bug 6824:** Support standard install locations for globus-gram-job-manager-setup

• **Bug 6825:** Remove hardcoded paths in globus-gram-job-manager-setup-fork

• **Bug 6826:** Remove hardcoded paths in globus-gram-job-manager-setup-condor

• **Bug 6840:** The PBS job manager doesn't handle large environments well

• **Bug 6855:** Undefined variable in Makefiles

• **Bug 6862:** PBS job manager fails if job history is enabled

• **Bug 6927:** A Loadleveler LRM for GRAM5 should be very welcome

• **Bug 720:** allow gram client to detect the version of a gram server

• **Bug 851:** Add "cleanup" RSL attribute for cleaning up a job submission

• **Bug 5536:** Missing dependency in package globus_gram_job_manager_auditing

• **Bug 5537:** Missing dependency in package globus_gram_job_manager_auditing

• **Bug 3373:** globus removes the temporary job directory before pbs writes back into it

• **Bug 5200:** GRAM (pre-webservices) from OSG 0.6.0 (VDT 1.6.1) has bad syslog format

• **Bug 5207:** GRAM SoftEnv extension bug

• **Bug 5250:** Does not support mpi jobtype of RSL script

• **Bug 5272:** Invalid parsing of RSL file

4. **Bug Fixes**

• **GRAM-93:** SGE LRM script doesn't handle environment vars with whitespace

• **GRAM-94:** SGE LRM doesn't check for executable existence

31 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6823
32 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6824
33 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6825
34 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6826
35 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6840
36 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6855
37 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6862
38 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6927
39 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=720
40 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=851
41 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5536
42 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5537
43 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=3373
44 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5200
45 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5207
46 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5250
47 http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5272
48 http://jira.globus.org/browse/GRAM-93
49 http://jira.globus.org/browse/GRAM-94
• **GRAM-95**: SGE LRM doesn't check for executable permissions

• **GRAM-96**: SGE LRM mishandles invalid environment definition

• **GRAM-103**: Ease two phase end commit timeout

• **GRAM-106**: Fix test failures with SGE LRM adapter

• **GRAM-108**: Fork perl zombies

• **GRAM-112**: Test case "globus-gram-client-stdio-size-test" fails but reports success

• **GRAM-113**: GRAM5 job manager sends extra state callbacks on restart

• **GRAM-114**: globus-gram-protocol-io-test crash with bad creds

• **GRAM-115**: globus_gram_protocol_unpack_status_reply crashes if given a null parameter to parse

• **GRAM-116**: job_info struct leaked in gram client

• **GRAM-117**: Add non blocking form of globus_gram_client_get_job_manager_version()

• **GRAM-120**: RSL too large log message has bad format

• **GRAM-128**: When restarting a crashed job manager, memory use can be high

• **GRAM-134**: globus-gatekeeper does not honor GLOBUS_HOSTNAME

• **GRAM-137**: Bad name for syslog in gatekeeper

• **GRAM-141**: Jobmanager version command does not delegate

• **GRAM-142**: Handling of extensions hashtable for jobmanager version command inconsistent

• **GRAM-143**: GRAM5 doesn't send STAGE_OUT callback

• **GRAM-144**: GRAM5 logging doesn't escape \r

• **GRAM-145**: GRAM5 Job Manager fails to save SEG timestamps in job state files

---

50 http://jira.globus.org/browse/GRAM-95
51 http://jira.globus.org/browse/GRAM-96
52 http://jira.globus.org/browse/GRAM-103
53 http://jira.globus.org/browse/GRAM-106
54 http://jira.globus.org/browse/GRAM-108
55 http://jira.globus.org/browse/GRAM-112
56 http://jira.globus.org/browse/GRAM-113
57 http://jira.globus.org/browse/GRAM-114
58 http://jira.globus.org/browse/GRAM-115
59 http://jira.globus.org/browse/GRAM-116
60 http://jira.globus.org/browse/GRAM-117
61 http://jira.globus.org/browse/GRAM-120
62 http://jira.globus.org/browse/GRAM-128
63 http://jira.globus.org/browse/GRAM-134
64 http://jira.globus.org/browse/GRAM-137
65 http://jira.globus.org/browse/GRAM-141
66 http://jira.globus.org/browse/GRAM-142
67 http://jira.globus.org/browse/GRAM-143
68 http://jira.globus.org/browse/GRAM-144
69 http://jira.globus.org/browse/GRAM-145
5. GRAM5 Throughput Tests

5.1. Experiment Hardware and Software Configuration

The following experiments were run on the nomer.mcs.anl.gov virtual cluster. The cluster consists of 6 partitions, each having a single Intel(R) Xeon(R) CPU E5430 @ 2.66GHz core, and 2GB RAM. The virtual machines in the cluster each had a single virtual network interface. The cluster was configured as follows:

• 1 node: Master node
• 5 nodes: Test/execution nodes

All nodes ran an apache2 http server, gmond (ganglia monitoring), and pbs_mom (torque LRM).

The master node also ran a globus-gatekeeper, globus-gridftp-server, globus-job-manager-event-generator, gmetad (Ganglia Meta Daemon), pbs_sched (Torque LRM scheduler), pbs_server (Torque LRM server), and nfsd linux kernel NFSv4 server for the execution nodes.

The test nodes (1 for single-client tests, 5 for multiple client tests) ran the gram-throughput-tester program. All 5 test/execution nodes executed the test job executables as scheduled by the LRM.

These tests were done with GT 5.0.1 beta1. The throughput tester was compiled from CVS trunk October 30, 2009.

5.2. Experiment Scenarios

The first set of tests submitted jobs to the GRAM5 service for one hour. After one hour elapsed, the client would terminate any jobs which were being managed by the GRAM5 service. The test client recorded the time of the experiment start, the time of the termination start, and the time after which all jobs had reached a DONE or FAILED state.

The throughput test client would generate a maximum of 50 simultaneous job requests. For all but the uncapped test below, a maximum of 2000 jobs were managed by the job manager at any given time (pending or active). Once the client had submitted 2000 jobs, it would stop submissions until a job completed. A total of 10 execution slots were available for the LRM to schedule jobs into, so many were in the GRAM5 PENDING state during the duration of the test.

The different gram-throughput-tester experiments consisted of:

[70] http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6686
Table 1. GRAM5 Throughput Tester Experiments

<table>
<thead>
<tr>
<th>Experiment Name</th>
<th>LRM monitoring method</th>
<th>Number of clients</th>
<th>Number of users</th>
<th>Maximum number of simultaneous jobs / client</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-client-poll</td>
<td>POLL</td>
<td>1</td>
<td>1</td>
<td>2000</td>
</tr>
<tr>
<td>1-client-seg</td>
<td>SEG</td>
<td>1</td>
<td>1</td>
<td>2000</td>
</tr>
<tr>
<td>1-client-seg-uncapped</td>
<td>SEG</td>
<td>1</td>
<td>1</td>
<td>unlimited</td>
</tr>
<tr>
<td>5-client-seg</td>
<td>SEG</td>
<td>5</td>
<td>1</td>
<td>2000</td>
</tr>
<tr>
<td>5-client-seg-diffusers</td>
<td>SEG</td>
<td>5</td>
<td>5</td>
<td>2000</td>
</tr>
</tbody>
</table>

5.3. Throughput Test Results

The following table contains a summary of the results of these experiments. The columns contain the following information:

<table>
<thead>
<tr>
<th>Experiment</th>
<th>Experiment name, the same as in the previous section</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Jobs</td>
<td>The total number of GRAM jobs that were submitted to the GRAM5 service by the throughput tester in one hour.</td>
</tr>
<tr>
<td>Termination Tasks After 1 Hour</td>
<td>The total number of jobs that were being managed by the GRAM5 service when the one hour test period elapsed. These jobs were terminated using the GRAM5 cancel protocol message by the throughput tester.</td>
</tr>
<tr>
<td>Termination Duration (hh:mm:ss)</td>
<td>The amount of time it took for the termination tasks to complete and all jobs to reach the DONE or FAILED state.</td>
</tr>
<tr>
<td>Master Node Max 1 min. Load Average</td>
<td>The maximum value of the one-minute load average on the master node, that is, the node running the GRAM5 and Torque service.</td>
</tr>
<tr>
<td>Master Node Average 1 min. Load Average</td>
<td>The average value of the one-minute load average on the master node over the duration of the test.</td>
</tr>
<tr>
<td>Errors</td>
<td>Description of any errors which occurred on the client side that prevented operations from completing as expected.</td>
</tr>
</tbody>
</table>
Table 2. GRAM5 Throughput Tester Results Summary

<table>
<thead>
<tr>
<th>Experiment</th>
<th>Total Jobs</th>
<th>Termination Tasks After 1 Hour</th>
<th>Termination Duration (hh:mm:ss)</th>
<th>Master Node Max 1 min. Load Average</th>
<th>Master Node Average 1 min. Load Average</th>
<th>Errors</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-client-poll</td>
<td>2110</td>
<td>2000</td>
<td>00:14:18</td>
<td>11.46</td>
<td>7.96</td>
<td>None</td>
</tr>
<tr>
<td>1-client-seg</td>
<td>2110</td>
<td>2000</td>
<td>00:10:36</td>
<td>2.82</td>
<td>0.93</td>
<td>None</td>
</tr>
<tr>
<td>1-client-seg-uncapped</td>
<td>6664</td>
<td>6584</td>
<td>00:42:46</td>
<td>3.26</td>
<td>2.75</td>
<td>None</td>
</tr>
<tr>
<td>5-client-seg</td>
<td>6800</td>
<td>6434</td>
<td>00:57:20</td>
<td>3.19</td>
<td>2.49</td>
<td>Connection refused during termination</td>
</tr>
<tr>
<td>5-client-seg-diffusers</td>
<td>7226</td>
<td>6720</td>
<td>00:45:41</td>
<td>3.79</td>
<td>3.13</td>
<td>None</td>
</tr>
</tbody>
</table>

6. GRAM5 Condor-G Tests

6.1. Experiment Hardware and Software Configuration

The following experiments were run on the nomer.mcs.anl.gov virtual cluster. The cluster consists of 6 partitions, each having a single Intel(R) Xeon(R) CPU E5430 @ 2.66GHz core, and 2GB RAM. The virtual machines in the cluster each had a single virtual network interface. The cluster was configured as follows:

- 1 node: Master node
- 1 nodes: Test/execution nodes
- 4 nodes: execution nodes

All nodes ran an apache2 http server, gmond (ganglia monitoring), and pbs_mom (torque LRM).

The master node also ran a globus-gatekeeper, globus-gridftp-server, globus-job-manager-event-generator, gmetad (Ganglia Meta Daemon), pbs_sched (Torque LRM scheduler), pbs_server (Torque LRM server), and nfsd linux kernel NFSv4 server for the execution nodes.

The test/execution node ran the condor-g daemons. The condor job classified ad included attributes to submit the jobs to the GRAM5 service running on the service node. The tests were done with Condor version 7.4.1. The Condor-G configuration parameters in the condor_config file were as follows:

Figure 1. Condor-G Experiment Configuration

GRIDMANAGER_MAX_PENDING_SUBMITS_PER_RESOURCE=50
GRIDMANAGER_MAX_SUBMITTED_JOBS_PER_RESOURCE=2000
GRIDMANAGER_MAX_PENDING_REQUESTS=50
GRIDMANAGER_JOB_PROBE_INTERVAL=300
GRIDMANAGER_MAX_JOBMANAGERS_PER_RESOURCE=0
ENABLE_GRID_MONITOR=FALSE
GRIDMANAGER_DEBUG= D_FULLDEBUG
GRIDMANAGER_GLOBUS_COMMIT_TIMEOUT=12000
The execution nodes executed the test job executables as scheduled by the LRM.

For this test, the test/execution node and the execution nodes where configured to run up 20 job processes each simultaneously.

### 6.2. Experiment Scenario

This test submitted a 2000 job condor job cluster, using the following classified ad:

**Figure 2. Condor-G Classified Ad**

```plaintext
Universe=grid
grid_resource = gt5 nomer1.mcs.anl.gov:2119/jobmanager-pbs
transfer_executable=False
executable=/bin/sleep
arguments=300
stream_output = False
stream_error = False
output = test.out.$(Process)
error = test.err.$(Process)
log = test.log
notification=Never
queue 2000
```

The configuration parameters are similar to the GRAM5 tests described in GRAM5 Throughput Tests section. The key difference being that this test runs until all 2000 jobs have completed and does not submit any jobs after the maximum of 2000 has been reached.

To provide a point of comparison, another test using similar parameters was run using the `gram-throughput-tester` program in place of Condor-G. Note that the Condor-G service provides file staging and a scratch directory beyond what the throughput tester job did.

The two experiments consist of:

<table>
<thead>
<tr>
<th>Table 3. GRAM5 Condor-G Experiments</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Experiment Name</strong></td>
</tr>
<tr>
<td>---------------------</td>
</tr>
<tr>
<td>Condor-G</td>
</tr>
<tr>
<td>Throughput Tester</td>
</tr>
</tbody>
</table>

### 6.3. Condor-G Test Results

The following table contains a summary of the results of these experiments. The columns contain the following information:

<table>
<thead>
<tr>
<th>Experiment</th>
<th>Experiment name, the same as in the previous section</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time to Submit 2000 Jobs</td>
<td>The total number of GRAM jobs that were submitted to the GRAM5 service by the throughput tester in one hour.</td>
</tr>
</tbody>
</table>
Time For All Jobs To Complete

The amount of time it took for all 2000 jobs to complete. The theoretical minimum value for this is 50 minutes if all 2000 jobs were submitted instantaneously and there was no overhead for them to be deployed to the 200 execution nodes.

LRM Submit Rate (Jobs/Minute)

The total number of jobs that were being managed by the GRAM5 service when the one hour test period elapsed. These jobs were terminated using the GRAM5 cancel protocol message by the throughput tester.

Master Node Max 1 min. Load Average

The maximum value of the one-minute load average on the master node, that is, the node running the GRAM5 and Torque service.

Master Node Average 1 min. Load Average

The average value of the one-minute load average on the master node over the duration of the test.

Table 4. GRAM5 Throughput Tester Results Summary

<table>
<thead>
<tr>
<th>Experiment</th>
<th>Time to Submit 2000 Jobs (hh:mm:ss)</th>
<th>Time For All Jobs To Complete (hh:mm:ss)</th>
<th>LRM Submit Rate (Jobs/Minute)</th>
<th>Master Node Max 1 min. Load Average</th>
<th>Master Node Average 1 min. Load Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Condor-G</td>
<td>00:32:34</td>
<td>02:00:56</td>
<td>94</td>
<td>6.56</td>
<td>1.64</td>
</tr>
<tr>
<td>Throughput Tester</td>
<td>00:17:58</td>
<td>01:54:49</td>
<td>111</td>
<td>2.43</td>
<td>0.53</td>
</tr>
</tbody>
</table>