
GT 5.0.2 GRAM5: Quality Profile

Table of Contents

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

<titleabbrev>Quality Profile</titleabbrev>

1. Test coverage reports

- [GT 5.0.2 Test Coverage Status Report](#)¹

2. Code analysis reports

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

3. Outstanding bugs

- [GRAM-2](#):² Investigate how to setup GRAM5 services in a HA setup
- [GRAM-4](#):³ Add support for a "managed fork" service
- [GRAM-5](#):⁴ Add gram-level prologue and epilogue script execution for mpi jobs
- [GRAM-12](#):⁵ Gatekeeper's syslog output cannot be controlled
- [GRAM-15](#):⁶ transition from httpg to https
- [GRAM-22](#):⁷ client connections can't be timed out
- [GRAM-23](#):⁸ Improved error codes and error reporting for users
- [GRAM-24](#):⁹ Debug/verbose flags for globusrun, globus-job-run
- [GRAM-51](#):¹⁰ configurable control of number of perl scripts that can run simultaneously

¹ ../reports/coverage/index.html

² <http://jira.globus.org/browse/GRAM-2>

³ <http://jira.globus.org/browse/GRAM-4>

⁴ <http://jira.globus.org/browse/GRAM-5>

⁵ <http://jira.globus.org/browse/GRAM-12>

⁶ <http://jira.globus.org/browse/GRAM-15>

⁷ <http://jira.globus.org/browse/GRAM-22>

⁸ <http://jira.globus.org/browse/GRAM-23>

⁹ <http://jira.globus.org/browse/GRAM-24>

¹⁰ <http://jira.globus.org/browse/GRAM-51>

- [GRAM-53](#):¹¹ Generalize log path configuration
- [GRAM-79](#):¹² Add support for OSG's "NFS Lite" concept
- [GRAM-99](#):¹³ Add a high-level diagram for the approach doc
- [GRAM-104](#):¹⁴ globus-job-manager-event-generator loads all historical events the first time run
- [GRAM-105](#):¹⁵ Held Condor jobs should be reported as SUSPENDED
- [GRAM-110](#):¹⁶ softenv extensions for GRAM5
- [GRAM-119](#):¹⁷ improve the GRAM LRM adapter doc
- [GRAM-122](#):¹⁸ tracking gram client software
- [GRAM-135](#):¹⁹ Improve developer doc for a reliable client
- [GRAM-138](#):²⁰ GRAM5 job manager uses a lot of memory when SEG is pointed to incorrect log path
- [GRAM-139](#):²¹ SEG may deadlock with threads
- [GRAM-149](#):²² GRAM5 Unix domain socket misbehaves on Snow Leopard
- [GRAM-154](#):²³ GASS Cache doesn't check for updates
- [GRAM-159](#):²⁴ GRAM5 Migration guide is outdated
- [GRAM-163](#):²⁵ improve error output for globusrun
- [Bug 5621](#):²⁶ gram2 credential refresh problems in 4.0.5
- [Bug 1934](#):²⁷ Gatekeeper's syslog output cannot be controlled
- [Bug 2739](#):²⁸ Gatekeeper AuthZ/Gridmap Callout result logging
- [Bug 2741](#):²⁹ catching SIGSEGV if dynamic loading of authorization modules fails
- [Bug 4199](#):³⁰ Patch pre-WS GRAM to use individual condor logs for jobs

¹¹ <http://jira.globus.org/browse/GRAM-53>

¹² <http://jira.globus.org/browse/GRAM-79>

¹³ <http://jira.globus.org/browse/GRAM-99>

¹⁴ <http://jira.globus.org/browse/GRAM-104>

¹⁵ <http://jira.globus.org/browse/GRAM-105>

¹⁶ <http://jira.globus.org/browse/GRAM-110>

¹⁷ <http://jira.globus.org/browse/GRAM-119>

¹⁸ <http://jira.globus.org/browse/GRAM-122>

¹⁹ <http://jira.globus.org/browse/GRAM-135>

²⁰ <http://jira.globus.org/browse/GRAM-138>

²¹ <http://jira.globus.org/browse/GRAM-139>

²² <http://jira.globus.org/browse/GRAM-149>

²³ <http://jira.globus.org/browse/GRAM-154>

²⁴ <http://jira.globus.org/browse/GRAM-159>

²⁵ <http://jira.globus.org/browse/GRAM-163>

²⁶ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5621

²⁷ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=1934

²⁸ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=2739

²⁹ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=2741

³⁰ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=4199

- [Bug 3795](#).³¹ jobmanager perl modules issues
- [Bug 4235](#).³² globus-job-manager doesn't exit if the job fails.
- [Bug 4730](#).³³ MPI Jobs using Globus LSF in HP XC Cluster....
- [Bug 4747](#).³⁴ Need evaluation of patch to JobManager.pm
- [Bug 4779](#).³⁵ gram GT2 log files: timestamps are not ISO 8601 compatible
- [Bug 5143](#).³⁶ DONE state never reported for Condor jobs when using Condor-G grid monitor
- [Bug 5429](#).³⁷ stdin is lost when jobtype=multiple with jobmanager-lsf
- [Bug 5554](#).³⁸ GRAM2 4.0.5 setup-globus-job-manager-fork.pl silent failure
- [Bug 5556](#).³⁹ Audit directory setup instructions are insecure
- [Bug 5775](#).⁴⁰ gram status of old jobs incorrect on some lsf systems
- [Bug 6184](#).⁴¹ pbs.pm jobmanager fails jobs on qstat failure
- [Bug 6337](#).⁴² Cannot configure globus to use different certificate path than default
- [Bug 6703](#).⁴³ 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](#).⁴⁴ Held Condor jobs should be reported as SUSPENDED by GRAM
- [Bug 6815](#).⁴⁵ Support standard install locations for globus-gram-protocol
- [Bug 6819](#).⁴⁶ Missing metadata in globus-scheduler-event-generator
- [Bug 6820](#).⁴⁷ Support standard install locations for globus-gatekeeper
- [Bug 6821](#).⁴⁸ Support standard install locations for globus-gatekeeper-setup
- [Bug 6822](#).⁴⁹ Support standard install locations for globus-gram-job-manager-scripts
- [Bug 6823](#).⁵⁰ Support standard install locations for globus-gram-job-manager

³¹ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=3795

³² http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=4235

³³ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=4730

³⁴ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=4747

³⁵ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=4779

³⁶ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5143

³⁷ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5429

³⁸ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5554

³⁹ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5556

⁴⁰ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5775

⁴¹ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6184

⁴² http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6337

⁴³ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6703

⁴⁴ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6768

⁴⁵ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6815

⁴⁶ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6819

⁴⁷ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6820

⁴⁸ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6821

⁴⁹ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6822

⁵⁰ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6823

- [Bug 6824](#).⁵¹ Support standard install locations for globus-gram-job-manager-setup
- [Bug 6825](#).⁵² Remove hardcoded paths in globus-gram-job-manager-setup-fork
- [Bug 6826](#).⁵³ Remove hardcoded paths in globus-gram-job-manager-setup-condor
- [Bug 6840](#).⁵⁴ The PBS job manager doesn't handle large environments well
- [Bug 6855](#).⁵⁵ Undefined variable in Makefiles
- [Bug 6862](#).⁵⁶ PBS job manager fails if job history is enabled
- [Bug 6927](#).⁵⁷ A Loadleveler LRM for GRAM5 should be very welcome
- [Bug 720](#).⁵⁸ allow gram client to detect the version of a gram server
- [Bug 851](#).⁵⁹ Add "cleanup" RSL attribute for cleaning up a job submission
- [Bug 5536](#).⁶⁰ Missing dependency in package globus_gram_job_manager_auditing
- [Bug 5537](#).⁶¹ Missing dependency in package globus_gram_job_manager_auditing
- [Bug 3373](#).⁶² globus removes the temporary job directory before pbs writes back into it
- [Bug 5200](#).⁶³ GRAM (pre-webservices) from OSG 0.6.0 (VDT 1.6.1) has bad syslog format
- [Bug 5207](#).⁶⁴ GRAM SoftEnv extension bug
- [Bug 5250](#).⁶⁵ Does not support mpi jobtype of RSL script
- [Bug 5272](#).⁶⁶ Invalid parsing of RSL file

4. Bug Fixes

- [GRAM-130](#).⁶⁷ Individual Condor Logs per Job
- [GRAM-136](#).⁶⁸ Error message not precise when disk quota is exceeded
- [GRAM-146](#).⁶⁹ GRAM5 Usage stats ignores GLOBUS_USAGE_TARGETS environment

⁵¹ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6824

⁵² http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6825

⁵³ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6826

⁵⁴ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6840

⁵⁵ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6855

⁵⁶ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6862

⁵⁷ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=6927

⁵⁸ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=720

⁵⁹ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=851

⁶⁰ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5536

⁶¹ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5537

⁶² http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=3373

⁶³ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5200

⁶⁴ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5207

⁶⁵ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5250

⁶⁶ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5272

⁶⁷ <http://jira.globus.org/browse/GRAM-130>

⁶⁸ <http://jira.globus.org/browse/GRAM-136>

⁶⁹ <http://jira.globus.org/browse/GRAM-146>

- [GRAM-155](#):⁷⁰ Leak in file_clean_up
- [GRAM-156](#):⁷¹ job-failure-code not provided by jobmanager in response to status query
- [GRAM-157](#):⁷² RSL substitutions not updated on a restart request
- [GRAM-158](#):⁷³ Remove references to setup-seg-job-manager.pl from documentation
- [GRAM-160](#):⁷⁴ double free in job manager
- [GRAM-161](#):⁷⁵ Don't count two phase commit time until state is sent
- [GRAM-164](#):⁷⁶ Invalid RSL value error doesn't indicate what values are valid
- [GRAM-167](#):⁷⁷ SGE LRM doesn't interpret path to stdout and stderr relative to directory
- [GRAM-168](#):⁷⁸ zombie job manager processes on Ranger SGE
- [GRAM-169](#):⁷⁹ Error messages in globus-job-get-output
- [GRAM-170](#):⁸⁰ globus-job-submit fails
- [GRAM-171](#):⁸¹ Simplify debugging of script errors
- [GRAM-172](#):⁸² Job Manager doesn't exit when proxy expires if jobs are present
- [GRAM-173](#):⁸³ SGE LRM doesn't set exit code for multiple jobs

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).

⁷⁰ <http://jira.globus.org/browse/GRAM-155>

⁷¹ <http://jira.globus.org/browse/GRAM-156>

⁷² <http://jira.globus.org/browse/GRAM-157>

⁷³ <http://jira.globus.org/browse/GRAM-158>

⁷⁴ <http://jira.globus.org/browse/GRAM-160>

⁷⁵ <http://jira.globus.org/browse/GRAM-161>

⁷⁶ <http://jira.globus.org/browse/GRAM-164>

⁷⁷ <http://jira.globus.org/browse/GRAM-167>

⁷⁸ <http://jira.globus.org/browse/GRAM-168>

⁷⁹ <http://jira.globus.org/browse/GRAM-169>

⁸⁰ <http://jira.globus.org/browse/GRAM-170>

⁸¹ <http://jira.globus.org/browse/GRAM-171>

⁸² <http://jira.globus.org/browse/GRAM-172>

⁸³ <http://jira.globus.org/browse/GRAM-173>

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 5.0.2 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:

Table 1. GRAM5 Throughput Tester Experiments

Experiment Name	LRM monitoring method	Number of clients	Number of users	Maximum number of simultaneous jobs / client
1-client-poll	POLL	1	1	2000
1-client-seg	SEG	1	1	2000
1-client-seg-uncapped	SEG	1	1	unlimited
5-client-seg	SEG	5	1	2000
5-client-seg-diffusers	SEG	5	5	2000

5.3. Throughput Test Results

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

Experiment	Experiment name, the same as in the previous section
Total Jobs	The total number of GRAM jobs that were <i>submitted</i> to the GRAM5 service by the throughput tester in one hour.
Termination Tasks After 1 Hour	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.
Termination Duration (hh:mm:ss)	The amount of time it took for the termination tasks to complete and all jobs to reach the DONE or FAILED state.
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.
Errors	Description of any errors which occurred on the client side that prevented operations from completing as expected.

Table 2. GRAM5 Throughput Tester Results Summary

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

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 were configured to run up to 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

```

Universe=grid
grid_resource = gt5 nomer1.mcs.anl.gov:2119/jobmanager-pbs
executable=/bin/sleep
arguments=300
transfer_executable=False
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 3. GRAM5 Condor-G Experiments

Experiment Name	LRM monitoring method	Number of clients	Number of users	Total number of jobs
Condor-G	SEG	1	1	2000
Throughput Tester	SEG	1	1	2000

6.3. Condor-G Test Results

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

Experiment	Experiment name, the same as in the previous section
Time to Submit 2000 Jobs	The total number of GRAM jobs that were <i>submitted</i> to the GRAM5 service by the throughput tester in one hour.
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

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