GT 5.0.1 GRAM5 : System Administrator's Guide
GT 5.0.1 GRAM5 : System Administrator's Guide

Introduction

This guide contains advanced configuration information for system administrators working with GRAM5. It provides references to information on procedures typically performed by system administrators, including installation, configuring, deploying, and testing the installation. It also describes additional prerequisites and host settings necessary for GRAM5 operation. Readers should be familiar with the Key Concepts and Implementation Approach for GRAM5 to understand the motivation for and interaction between the various deployed components.

⚠️ Important

The information in this GRAM5 Admin Guide is in addition to the basic Globus Toolkit prerequisite, overview, installation, security configuration instructions in the Installing GT 5.0.1. Read through this guide before continuing!
# Table of Contents

1. Building and Installing ................................................................. 1
   1. Local Resource Manager ......................................................... 1
   2. LRM Adapters ................................................................. 1
2. Configuring ................................................................. 2
   1. Typical Configuration ......................................................... 2
   2. Non-default Configuration .................................................. 3
   3. Configuration Details ......................................................... 3
3. Deploying ................................................................. 6
   1. Deploying GRAM5 via inetd or xinetd .................................. 6
   2. Deploying GRAM5 as a daemon ......................................... 7
4. Running the SEG ................................................................. 8
   1. Starting the SEG ............................................................. 8
   2. Stopping the SEG .......................................................... 8
5. Scalability and Performance Recommendations ................................. 9
   1. Server-side Recommendations ........................................... 9
6. Audit Logging ................................................................. 10
   1. Overview ................................................................. 10
   2. Audit and Accounting Records .......................................... 10
   3. For More Information ....................................................... 11
   4. Configuration ............................................................. 11
   5. Audit Database Interface .................................................. 11
7. Testing ................................................................. 13
   1. GRAM protocol tests ....................................................... 13
   2. GRAM client tests .......................................................... 13
   3. GRAM Job Manager Tests ............................................... 14
8. Security Considerations ................................................................ 16
9. Admin Debugging ....................................................................... 17
10. GRAM5 Admin Programs ............................................................ 18
    globus-gram-audit ................................................................. 19
    globus-job-manager-event-generator ...................................... 20
    globus-job-manager ............................................................ 21
11. Troubleshooting ............................................................... 26
    1. Troubleshooting tips ........................................................ 26
    2. Errors ....................................................................... 27
12. Usage statistics collection by the Globus Alliance .............................. 37
    1. GRAM5-specific usage statistics ....................................... 37
Glossary ......................................................................................... 40
Index .......................................................................................... 41
List of Tables

1.1. Supported LRM Adapters ................................................................................................................ 1
2.1. LRM Adapter Make targets .............................................................................................................. 2
9.1. GRAM5 Log Levels ..................................................................................................................... 17
11.1. GRAM5 Errors .......................................................................................................................... 28
List of Examples

2.1. Installing PBS LRM Adapter ............................................................................................................ 3
2.2. Configuring GRAM5 to use PBS by default .................................................................................. 5
2.3. Disabling the PBS LRM Adapter .................................................................................................. 5
4.1. Starting and Stopping the SEG ..................................................................................................... 8
7.1. Running the GRAM Protocol Test Suite ...................................................................................... 13
7.2. Running the GRAM Client Test Suite ........................................................................................... 14
7.3. Running the GRAM Job Manager Test Suite .................................................................................. 15
Chapter 1. Building and Installing

GRAM5 is built and installed as part of a default GT 5.0.1 installation. For basic installation instructions, see Installing GT 5.0.1.

1. Local Resource Manager

GRAM5 depends on a local mechanism for starting and controlling jobs. GRAM5 includes a fork local resource manager which requires no special software to execute jobs on the local host. GRAM5 can also be configured to use additional batch facilities and schedulers such as condor, torque, or LSF. Install and configure any local resource managers you intend to use prior to configuring GRAM5.

2. LRM Adapters

GRAM5 depends on LRM adapters to execute jobs described by GRAM5 RSL documents on a resource managed by a LRM.

LRM adapters included in the GT 5.0.1 release are:

Table 1.1. Supported LRM Adapters

<table>
<thead>
<tr>
<th>LRM Adapter Name</th>
<th>LRM Supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>fork</td>
<td>Unscheduled local execution</td>
</tr>
<tr>
<td>pbs</td>
<td>Torque(^1)</td>
</tr>
<tr>
<td>condor</td>
<td>Condor(^2)</td>
</tr>
<tr>
<td>lsf</td>
<td>LSF(^3)</td>
</tr>
<tr>
<td>SGE</td>
<td>Grid Engine(^4)</td>
</tr>
</tbody>
</table>

\(^{1}\) http://www.clusterresources.com/products/torque-resource-manager.php
\(^{2}\) http://www.cs.wisc.edu/condor/
\(^{3}\) http://www.platform.com/Products/Platform.LSF.Family/Platform.LSF/
\(^{4}\) http://gridengine.sunsource.net/

Note

The pbs LRM adapter may work with other systems that implement POSIX batch environment services.

For configuration details, see "Configuring LRM adapters" in the Configuring section.
Chapter 2. Configuring

1. Typical Configuration

1.1. Configuring LRM Adapters

Three GRAM5 LRM adapters included in the GT 5.0.1 installer besides the default fork LRM adapter. These are installed by using the following make rules from the installer directory:

Table 2.1. LRM Adapter Make targets

<table>
<thead>
<tr>
<th>LRM</th>
<th>Installer make target</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fork</td>
<td>gram5-fork</td>
</tr>
<tr>
<td>Condor</td>
<td>gram5-condor</td>
</tr>
<tr>
<td>LSF</td>
<td>gram5-lsf</td>
</tr>
<tr>
<td>PBS</td>
<td>gram5-pbs</td>
</tr>
<tr>
<td>SGE</td>
<td>gram5-sge</td>
</tr>
</tbody>
</table>
Example 2.1. Installing PBS LRM Adapter

% make gram5-pbs
  cd gpt && OBJECT_MODE=32 ./build_gpt
build_gpt ===> installing GPT into /opt/globus
build_gpt ===> building /usr/src/globus/gram5-beta1/gpt/support/Compress-Zlib-1.21
build_gpt ===> building /usr/src/globus/gram5-beta1/gpt/support/IO-Zlib-1.01
build_gpt ===> building /usr/src/globus/gram5-beta1/gpt/support/makepatch-2.00a
build_gpt ===> building /usr/src/globus/gram5-beta1/gpt/support/Archive-Tar-0.22
build_gpt ===> building /usr/src/globus/gram5-beta1/gpt/support/PodParser-1.18
build_gpt ===> building /usr/src/globus/gram5-beta1/gpt/support/Digest-MD5-2.20
build_gpt ===> building /usr/src/globus/gram5-beta1/gpt/packaging_tools
  /opt/globus/sbin/gpt-build   -srcdir=source-trees/core/source gcc64dbg
  gpt-build ===> Changing to /usr/src/globus/gram5-beta1/source-trees/core/source
build_gpt ===> BUILDING FLAVOR gcc64dbg

  additional lines omitted

% make install
  if [ ! -L /opt/globus/etc/globus_packages ]; then \n    cd /opt/globus/etc/; \n    ln -sf gpt/packages globus_packages; \n    fi; \n  /opt/globus/sbin/gpt-postinstall
  running /opt/globus/setup/globus/setup-globus-common..[ Changing to /opt/globus/setup/glob
  creating globus-sh-tools-vars.sh
  creating globus-script-initializer
  creating Globus::Core::Paths
  checking globus-hostname
  Done

  additional lines omitted

The LRM make targets will check all dependencies they need and try to build them if they are not yet installed.

2. Non-default Configuration

2.1. Authorization
  TODO: Discussion of Job Manager Callout

3. Configuration Details

3.1. LRM-Specific Scheduler Event Generator configuration files

In addition to the service configuration described above, there are LRM-specific configuration files for the Scheduler Event Generator modules. These files consist of name=value pairs separated by newlines. These files are:
3.1.1. $GLOBUS_LOCATION/etc/globus-fork.conf

Configuration for the Fork SEG module implementation. The attributes names for this file are:

log_path Path to the SEG Fork log (used by the globus-fork-starter and the SEG). The value of this should be the path to a world-writable file. The default value for this created by the Fork setup package is $GLOBUS_LOCATION/var/globus-fork.log. This file must be readable by the account that the SEG is running as.

3.1.2. $GLOBUS_LOCATION/etc/globus-condor.conf

Configuration for the Condor SEG module implementation. The attributes names for this file are:

log_path Path to the SEG Condor log (used by the Globus::GRAM::JobManager::condor perl module and Condor SEG module. The value of this should be the path to a world-readable and world-writable file. The default value for this created by the Fork setup package is $GLOBUS_LOCATION/var/globus-condor.log

3.1.3. $GLOBUS_LOCATION/etc/globus-pbs.conf

Configuration for the PBS SEG module implementation. The attributes names for this file are:

log_path Path to the SEG PBS logs (used by the Globus::GRAM::JobManager::pbs perl module and PBS SEG module. The value of this should be the path to the directory containing the server logs generated by PBS. For the SEG to operate, these files must have file permissions such that the files may be read by the user the SEG is running as.

3.1.4. $GLOBUS_LOCATION/etc/globus-lsf.conf

Configuration for the LSF SEG module implementation. The attributes names for this file are:

log_path Path to the SEG LSF log directory. This is used by the LSF SEG module. The value of this should be the path to the directory containing the server logs generated by LSF. For the SEG to operate, these files must have file permissions such that the files may be read by the user the SEG is running as.

3.1.5. $GLOBUS_LOCATION/etc/globus-sge.conf

Configuration for the SGE SEG module implementation. The attributes names for this file are:

log_path Path to the SGE reporting file. used by the SGE SEG module. The value of this should be the path to the cell's accounting file generated by SGE. For the SEG to operate, this file must have file permissions such that the files may be read by the user the SEG is running as and the ARCO database uploader must not be running.

3.2. Enabling an LRM-Specific SEG module to use with GRAM5

In most situations, the SEG interface provides a more efficient way to monitor jobs than the poll method. This is configured by adding the `--seg-module LRM` command-line option to the job manager configuration in $GLOBUS_LOCATION/etc/grid-services/jobmanager-LRM.
3.3. Defining a default local resource manager

A client can submit a job without specifying the local resource manager (LRM) that should execute the job.

The default job manager for a particular host is determined by the contents of the file $GLOBUS_LOCATION/etc/grid-services/jobmanager. By default, this file is a symbolic link to the service definition for the first LRM adapter that is installed. To change the default LRM, change the link to point to a different LRM service definition.

Example 2.2. Configuring GRAM5 to use PBS by default

```
% cd $GLOBUS_LOCATION/etc/grid-services
% rm jobmanager
% ln -s jobmanager-pbs jobmanager
```

3.4. Disabling an already installed local resource manager adapter

When the GRAM5 gatekeeper accepts a new connection, it checks the contents of the service directory to determine what services are configured. To disable a service, remove the service entry from the service directory. The service entry can be recreated by running the `gpt-postinstall -force` command.

Example 2.3. Disabling the PBS LRM Adapter

```
% cd $GLOBUS_LOCATION/etc/grid-services
% rm jobmanager-pbs
```

If the PBS LRM was the default job manager, then the symlink $GLOBUS_LOCATION/etc/grid-services/jobmanager will no longer point to a valid file, and there will be no default job manager configured. Clients must explicitly choose some other LRM which is configured.
Chapter 3. Deploying

GRAM5 is installed as part of a standard toolkit installation. By default the fork LRM interface is installed and configured to use the poll interface.

In order to run the service, the `globus-gatekeeper` program must be started on a service node. This may be done in two supported ways, either via a super-server such as inetd or xinetd, or as a stand-alone daemon process.

1. Deploying GRAM5 via inetd or xinetd

To deploy GRAM5 to be started by inetd, define a service entry for the gsigatekeeper service in `/etc/services` file if it is not already present. The ICANN-assigned port number for the service is 2119, so add an entry like:

```
gsigatekeeper   2119/tcp
```

1.1. Deploying GRAM5 via inetd

To deploy GRAM5 to be started via inetd, modify `/etc/inetd.conf` (or your operating-system specific path to this file) by adding the following entry (on one line):

```
gsigatekeeper stream tcp nowait root /usr/bin/env env LD_LIBRARY_PATH=GLOBUS_LOCATION/lib GLOBUS_LOCATION/sbin/globus-gatekeeper -conf GLOBUS_LOCATION/etc/globus-gatekeeper.conf
```

After doing so, run the operating-system specific command to have inetd reload its configuration.

![Important](image.png)

This example assumes that the `LD_LIBRARY_PATH` environment variable is needed to be able to run the GRAM5 programs and scripts. This is operating-system specific and may need to be changed. Consult the documentation for the operating system's runtime linker in the case of runtime problems.

The `GLOBUS_LOCATION` string in the above example must be replaced with the actual path where GT has been installed.

![Note](image.png)

When deploying GRAM5 via xinetd, be sure to include the command-line option `-inetd` in the gatekeeper configuration file; otherwise, the gatekeeper will log a warning each time it is started.

1.2. Deploying GRAM5 via xinetd

To deploy GRAM5 to be started via xinetd, create a file in the `/etc/xinetd.d/` directory (or the operating-system specific path to xinetd configuration files) called `gsigatekeeper` with the following contents:

```
service gsigatekeeper
{
    socket_type = stream
    protocol = tcp
    wait = no
    user = root
    env = LD_LIBRARY_PATH=GLOBUS_LOCATION/lib
    server = GLOBUS_LOCATION/sbin/globus-gatekeeper
}
```
server_args = -conf $GLOBUS_LOCATION/etc/globus-gatekeeper.conf
disable = no
}

After doing so, run the operating-system specific command to have xinetd reload its configuration.

⚠️ Important

This example assumes that the \texttt{LD\_LIBRARY\_PATH} environment variable is needed to be able to run the GRAM5 programs and scripts. This is operating-system specific and may need to be changed. Consult the documentation for the operating system's runtime linker in the case of runtime problems.

The \texttt{GLOBUS\_LOCATION} string in the above example must be replaced with the actual path where GT has been installed.

🔍 Note

When deploying GRAM5 via xinetd, be sure to include the command-line option \texttt{-inetd} in the gatekeeper configuration file; otherwise, the gatekeeper will log a warning each time it is started.

2. Deploying GRAM5 as a daemon

To deploy GRAM5 to be started as a daemon, run the \texttt{globus-gatekeeper} at boot time. This may be done in an init script, via cron, or other system-specific methods. The typical command-line to run is

\texttt{globus-gatekeeper -conf $GLOBUS\_LOCATION/etc/globus-gatekeeper.conf}

🔍 Note

When deploying GRAM5 as a daemon, be sure to include the command-line option \texttt{-f} in the gatekeeper configuration file; otherwise, the gatekeeper will log a warning each time it is started.
Chapter 4. Running the SEG

1. Starting the SEG

GRAM5 can be configured to use the `globus-job-manager-event-generator` program to monitor job state changes. This is often more efficient than using a LRM adapter's `poll` method. This program is configured when the LRM-specific bundle is installed. However, by default, the job manager does not use the SEG. It must be explicitly enabled by adding the `-seg-module LRM` option to the job manager configuration.

To start the `globus-job-manager-event-generator` program, add the following command to your system init scripts or crontab to be run at boot time:

```
$GLOBUS_LOCATION/sbin/globus-job-manager-event-generator -scheduler LRM -background -pidfile PIDFILE
```

This will start the `globus-job-manager-event-generator` to monitor jobs for the resource named by `LRM` in the background. The process id of the command will be written to the file named by `PIDFILE`, so that processes can check if it is running and kill it if necessary.

⚠️ Important

If the job manager is configured to use a SEG module but the `globus-job-manager-event-generator` is not running for that LRM, jobs will appear to hang. It is important that the program be running whenever GRAM5 jobs might be run.

2. Stopping the SEG

To stop the SEG, kill the `globus-job-manager-event-generator` process. The `-pidfile` option makes it easy to know which process to kill. When the SEG terminates, it will remove that file.

Example 4.1. Starting and Stopping the SEG

This example shows how to start and stop the SEG using the command-line options described above.

```
% globus-job-manager-event-generator -scheduler pbs -background -pidfile $GLOBUS_LOCATION/var/globus-job-manager-seg-pbs.pid
Running in background (78258)
% kill `cat $GLOBUS_LOCATION/var/globus-job-manager-seg-pbs.pid`
```
Chapter 5. Scalability and Performance Recommendations

This document includes recommendations for increasing the scalability and performance of GRAM5 in a Grid.

1. Server-side Recommendations

1.1. Filesystem related settings

1. The GRAM5 service stores job state for crash recovery on disk. By default, the $GLOBUS_LOCATION/tmp/gram_job_state directory is used for these files. If this path is located on a distributed file system mount, locking and updating the job state files can negatively affect performance.

To configure GRAM5 to use a local disk for job state files, modify $GLOBUS_LOCATION/etc/globus-job-manager.conf or $GLOBUS_LOCATION/etc/grid-services/jobmanager-LRM so that the argument to the -state-file-dir is a local directory path. That directory must be world writable with the "sticky bit" set (mode 1777).

1.2. LRM-related settings

1. The GRAM5 service can use two different interfaces to obtain job state changes, polling and using the SEG. The default method is to poll the LRM via the GRAM5 script interface. This polling method is often less efficient than the SEG method and results in a higher load on the GRAM5 service node, even when all managed jobs are waiting in the queue for execution.

The other method, uses a program implementing the SEG interface to generate LRM events which can be stored in a log file for job managers run by many different users to process. When this is used, the multiple job managers may detect job state changes without having to directly interact with the LRM.

To enable the SEG to monitor jobs for a particular LRM, install and configure the LRM-specific gram5 bundle, run the globus-job-manager-event-generator program on a node which can access the LRM interfaces needed by the LRM-specific SEG module, and configure the LRM-specific service instance to use the SEG to monitor jobs for state changes.
Chapter 6. Audit Logging

1. Overview

GRAM5 includes mechanisms to provide access to audit and accounting information associated with jobs that GRAM5 submits to a local resource manager (LRM) such as PBS, LSF, or Condor.

Note

Remember, GRAM is not a local resource manager but rather a protocol engine for communicating with a range of different local resource managers using a standard message format.

In some scenarios, it is desirable to get general information about the usage of the underlying LRM, such as:

• What kinds of jobs were submitted via GRAM?
• How long did the processing of a job take?
• How many jobs were submitted by user X?

The following three use cases give a better overview of the meaning and purpose of auditing and accounting:

1. Group Access. A grid resource provider allows a remote service (e.g., a gateway or portal) to submit jobs on behalf of multiple users. The grid resource provider only obtains information about the identity of the remote submitting service and thus does not know the identity of the users for which the grid jobs are submitted. This group access is allowed under the condition that the remote service stores audit information so that, if and when needed, the grid resource provider can request and obtain information to track a specific job back to an individual user.

2. Query Job Accounting. A client that submits a job needs to be able to obtain, after the job has completed, information about the resources consumed by that job. In portal and gateway environments where many users submit many jobs against a single allocation, this per-job accounting information is needed soon after the job completes so that client-side accounting can be updated. Accounting information is sensitive and thus should only be released to authorized parties.

3. Auditing. In a distributed multi-site environment, it can be necessary to investigate various forms of suspected intrusion and abuse. In such cases, we may need to access an audit trail of the actions performed by a service. When accessing this audit trail, it will frequently be important to be able to relate specific actions to the user.

Audit logging in GRAM5 is done when a job completes.

2. Audit and Accounting Records

While audit and accounting records may be generated and stored by different entities in different contexts, we make the following assumptions in this chapter:

<table>
<thead>
<tr>
<th></th>
<th>Audit Records</th>
<th>Accounting Records</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generated by:</td>
<td>GRAM service</td>
<td>LRM to which the GRAM service submits jobs</td>
</tr>
<tr>
<td>Stored in:</td>
<td>Database, indexed by GJID</td>
<td>LRM, indexed by JID</td>
</tr>
<tr>
<td>Data that is stored:</td>
<td>See list below.</td>
<td>May include all information about the duration and resource-usage of a job</td>
</tr>
</tbody>
</table>
The audit record of each job contains the following data:

- **job_grid_id**: String representation of the resource EPR
- **local_job_id**: Job/process id generated by the scheduler
- **subject_name**: Distinguished name (DN) of the user
- **username**: Local username
- **idempotence_id**: Job id generated on the client-side
- **creation_time**: Date when the job resource is created
- **queued_time**: Date when the job is submitted to the scheduler
- **stage_in_grid_id**: String representation of the stageIn-EPR (RFT)
- **stage_out_grid_id**: String representation of the stageOut-EPR (RFT)
- **clean_up_grid_id**: String representation of the cleanUp-EPR (RFT)
- **globus_toolkit_version**: Version of the server-side GT
- **resource_manager_type**: Type of the resource manager (Fork, Condor, ...)
- **job_description**: Complete job description document
- **success_flag**: Flag that shows whether the job failed or finished successfully
- **finished_flag**: Flag that shows whether the job is already fully processed or still in progress
- **gateway_user**: Teragrid identity of the user which submitted the job.

### 3. For More Information

The rest of this chapter focuses on how to configure GRAM5 to enable Audit-Logging. A case study for TeraGrid can be read [here](http://www.teragridforum.org/mediawiki/index.php?title=GRAM5_Audit), which also includes more information about how to use this data to get accounting information of a job, query the audit database for information via a Web Services interface, etc.

### 4. Configuration

Audit logging is turned off by default. To enable GRAM5 audit logging, in the job manager, add the command-line option `-audit-directory` to the job manager configuration in either `$GLOBUS_LOCATION/etc/globus-job-manager.conf` to enable it for all job manager services, or in `$GLOBUS_LOCATION/etc/grid-services/LRM_SERVICE_NAME` to enable it for a particular job manager service for a particular LRM.

### 5. Audit Database Interface

The `globus-gram-audit` program reads GRAM5 audit records and loads those records into an SQL database. This program is available as part of the `globus_gram_job_manager_auditing` package. It must be configured by installing and running the `globus_gram_job_manager_auditing_setup_scripts` setup package via `gpt-postinstall`. This setup script

---

creates the $GLOBUS_LOCATION/etc/globus-job-manager-audit.conf configuration file described
below and creates database tables needed by the audit system.

The globus-gram-audit program support three database systems: MySQL, PostgreSQL, and SQLite.

5.1. Audit Configuration File

The auditing configuration file consists of a series of line-oriented records that define various configuration attributes
used by the globus-gram-audit program. This file can be edited by hand or using the $GLOBUS_LOCATION/setup/glo-
bus/setup-globus-gram-auditing program. That program is run automatically with the default values described below
when gpt-postinstall is run after installing the audit setup package.

The values which may be defined in the configuration file are:

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Values</th>
<th>Default when not specified on the setup command-line</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRIVER</td>
<td>The name of the Perl 5 DBI driver for the database to be used. The supported drivers for this program are SQLite, Pg (for PostgreSQL), and mysql.</td>
<td>SQLite</td>
</tr>
<tr>
<td>DATABASE</td>
<td>The DBI data source specification to contact the audit database.</td>
<td>dbname=$GLOBUS_LOCATION/var/gram_audit_database/gram_audit.db</td>
</tr>
<tr>
<td>USERNAME</td>
<td>Username to authenticate as to the database</td>
<td></td>
</tr>
<tr>
<td>PASSWORD</td>
<td>Password to use to authenticate with the database</td>
<td></td>
</tr>
<tr>
<td>AUDITVERSION</td>
<td>Version of the audit database table schemas to use. May be 1 or 1TG for this version of the software.</td>
<td>1</td>
</tr>
</tbody>
</table>

Thus, the default configuration file when GLOBUS_LOCATION is /opt/globus is

DRIVER:SQLite
DATABASE:dbname=/opt/globus/var/gram_audit_database/gram_audit.db
USERNAME:
PASSWORD:
AUDITVERSION:1
Chapter 7. Testing

There are three test suites available to verify that the GRAM5 client and service are installed correctly.

1. GRAM protocol tests

The GRAM protocol test suite tests the implementation of the GRAM protocol library which is used by the job manager and GRAM clients to process messages. The following examples shows how to run the test suite.

**Example 7.1. Running the GRAM Protocol Test Suite**

```
% cd $GLOBUS_LOCATION/test/globus_gram_protocol_test
% grid-proxy-init
Your identity: /DC=org/O=example/OU=grid/CN=Joe User
Enter GRID pass phrase for this identity:
Creating proxy ................................... Done
Your proxy is valid until: Thu Nov 12 23:28:05 2009
% .TESTS.pl
globus-gram-protocol-allow-attach-test...........ok
globus-gram-protocol-error-test..................ok
globus-gram-protocol-io-test....................ok
globus-gram-protocol-pack-test...................ok
pack-with-extensions-test.........................ok
create-extensions-test............................ok
unpack-message-test..............................ok
unpack-with-extensions-test......................ok
unpack-job-request-reply-with-extensions-test....ok
unpack-status-reply-with-extensions-test.........ok
All tests successful.
Files=10, Tests=42,  1 wallclock secs ( 0.37 cusr +  0.24 csys =  0.61 CPU)
```

2. GRAM client tests

The GRAM client test suite tests the interactions between the GRAM client API implementation and the job manager service. These tests include authentication, callback, signal, job, and cancellation tests.
Example 7.2. Running the GRAM Client Test Suite

% cd $GLOBUS_LOCATION/test/globus_gram_client_test
% grid-proxy-init
Your identity: /DC=org/O=example/OU=grid/CN=Joe User
Enter GRID pass phrase for this identity:
Creating proxy ................................... Done
Your proxy is valid until: Thu Nov 12 23:28:05 2009
% ./TESTS.pl
globus-gram-client-activate-test..................ok
globus-gram-client-callback-contact-test.........ok
globus-gram-client-cancel-test...................ok
globus-gram-client-nonblocking-register-test....ok 1/4 Failed submitting job request because:
globus_xio_gsi: gss_init_sec_context failed.
GSS Major Status: Unexpected Gatekeeper or Service Name
globus_gsi_gssapi: Authorization denied: The name of the remote entity (/DC=org/O=example/)

, globus-gram-client-nonblocking-register-test....ok
globus-gram-client-refresh-credentials-test.....ok
globus-gram-client-register-test................ok
globus-gram-client-register-callback-test.......ok 1/4 Failed submitting job request because:
globus_xio_gsi: gss_init_sec_context failed.
GSS Major Status: Unexpected Gatekeeper or Service Name
globus_gsi_gssapi: Authorization denied: The name of the remote entity (/DC=org/O=example/)

, globus-gram-client-register-callback-test.......ok
globus-gram-client-register-cancel-test.........ok
globus-gram-client-ping-test....................ok
globus-gram-client-status-test...................Made 3961 calls to status in 60.416390 seconds
globus-gram-client-status-test...................ok
globus-gram-client-two-phase-commit-test........ok
globus-gram-client-stdio-size-test...............job manager returned 0 (Success) when I expected it to still be streaming output
globus-gram-client-stdio-size-test...............ok
version-test....................................ok
All tests successful.
Files=14, Tests=33, 791 wallclock secs (32.10 cusr + 5.34 csys = 37.44 CPU)

Note

Some of the test cases display messages that look like errors when running. This is to be expected. The only concern should be the final lines indicating if the tests are successful or not.

Note

By default, this suite uses tests against a personal gatekeeper running the fork LRM. To run against another service, set the environment variable CONTACT_STRING to the service contact string prior to starting the tests.

3. GRAM Job Manager Tests

The GRAM job manager test suite tests the features provided by the LRM scripts, including detecting failures, staging files, and submitting different types of jobs. The following example shows how to run the job manager tests.
Example 7.3. Running the GRAM Job Manager Test Suite

% cd $GLOBUS_LOCATION/test/globus_gram_job_manager_test
% grid-proxy-init
Your identity: /DC=org/O=example/OU=grid/CN=Joe User
Enter GRID pass phrase for this identity:
Creating proxy ................................... Done
Your proxy is valid until: Thu Nov 12 23:28:05 2009
% ./TESTS.pl
job-manager-script-test..................ok
globus-gram-job-manager-stdio-test.......ok
globus-gram-job-manager-submit-test......ok
globus-gram-job-manager-failure-test.....ok
globus-gram-job-manager-rsl-size-test....ok
globus-gram-job-manager-user-test........ok
All tests successful.
Files=6, Tests=137, 200 wallclock secs (32.10 cusr +  5.34 csys = 37.44 CPU)

Note

This test requires a GridFTP server to be running on the host running the test suite. If one is not running, then
the following test cases will fail: globus-gram-job-manager-stdio-test subtests 2, 6, 9, 12, 13, 16, 17, 18, 19.

Note

By default, this suite uses tests against a personal gatekeeper running the fork LRM. To run against another
service, set the environment variable CONTACT_STRING to the service contact string prior to starting the
tests.
Chapter 8. Security Considerations

No special security considerations exist at this time.
Chapter 9. Admin Debugging

By default, GRAM5 logs errors to $HOME/gram_search/YY/MM/DD.log where YY/MM/DD is the time of the log event in GMT. The log file format conforms to CEDPS Logging Best Practices\(^1\). GRAM5 log files are governed by the log levels defined in the job manager configuration file. The log levels available are defined below:

Table 9.1. GRAM5 Log Levels

<table>
<thead>
<tr>
<th>Level</th>
<th>Meaning</th>
<th>Default Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>FATAL</td>
<td>Problems which cause the job manager to terminate prematurely</td>
<td>Enabled</td>
</tr>
<tr>
<td>ERROR</td>
<td>Problems which cause a job or operation to fail</td>
<td>Enabled</td>
</tr>
<tr>
<td>WARN</td>
<td>Problems which cause minor problems with job execution or monitoring</td>
<td>Disabled</td>
</tr>
<tr>
<td>INFO</td>
<td>Major events in the lifetime of the job manager and its jobs</td>
<td>Disabled</td>
</tr>
<tr>
<td>DEBUG</td>
<td>Minor events in the lifetime of jobs</td>
<td>Disabled</td>
</tr>
<tr>
<td>TRACE</td>
<td>Job processing details</td>
<td>Disabled</td>
</tr>
</tbody>
</table>

To enable logging for GRAM5, modify $GLOBUS_LOCATION/etc/globus-job-manager.conf so that it has either -stdio-log PATH to log to a file or -enable-syslog to log using the syslog service. To select log levels, add -log-levels "LEVELS" to the configuration file. The LEVELS string can contain any of the log levels mentioned aboved joined by the vertical bar character `|`.

\(^1\) [http://www.cedps.net/index.php/LoggingBestPractices](http://www.cedps.net/index.php/LoggingBestPractices)
Chapter 10. GRAM5 Admin Programs
Name

globus-gram-audit -- Load GRAM4 and GRAM5 audit records into a database

globus-gram-audit [--conf CONFIG_FILE] [--check] [--delete] [--audit-directory AUDITDIR]

Description

The globus-gram-audit program loads audit records to an SQL-based database. It reads $GLOBUS_LOCATION/etc/globus-job-manager.conf by default to determine the audit directory and then uploads all files in that directory that contain valid audit records to the database configured by the globus_gram_job_manager_auditing_setup_scripts package. If the upload completes successfully, the audit files will be removed.

The full set of command-line options to globus-gram-audit consist of:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--conf CONFIG_FILE</td>
<td>Use CONFIG_FILE instead of the default from the configuration file for audit database configuration.</td>
</tr>
<tr>
<td>--check</td>
<td>Check whether the insertion of a record was successful by querying the database after inserting the records. This is used in tests.</td>
</tr>
<tr>
<td>--delete</td>
<td>Delete audit records from the database right after inserting them. This is used in tests to avoid filling the database with test records.</td>
</tr>
<tr>
<td>--audit-directory DIR</td>
<td>Look for audit records in DIR, instead of looking in the directory specified in the job manager configuration. This is used in tests to control which records are loaded to the database and then deleted.</td>
</tr>
<tr>
<td>--query SQL</td>
<td>Perform the given SQL query on the audit database. This uses the database information from the configuration file to determine how to contact the database.</td>
</tr>
</tbody>
</table>

FILES

The globus-gram-audit uses the following files (paths relative to $GLOBUS_LOCATION).

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>etc/globus-gram-job-manager.conf</td>
<td>GRAM5 job manager configuration. It includes the default path to the audit directory</td>
</tr>
<tr>
<td>etc/globus-gram-audit.conf</td>
<td>Audit configuration. It includes the information needed to contact the audit database.</td>
</tr>
</tbody>
</table>
Name
globus-job-manager-event-generator -- Create LRM-independent SEG files for the job manager to use

Description
The globus-job-manager-event-generator program is a utility which uses LRM-specific SEG parsers to generate a LRM-independent log file that a job manager instance can use to process job status change events. This program runs independently of all globus-job-manager instances so that only one process needs to deal with the LRM interface. The globus-job-manager-event-generator program can be run as a privileged user if required to interface with the LRM.

In order for globus-job-manager-event-generator to handle events for a particular LRM, the globus_scheduler_event_generator_job_manager_setup setup package must be configured after the LRM-specific setup package has been run. This can be forced by gpt-postinstall-force or running the command cd $GLOBUS_LOCATION/setup/globus; ./setup-seg-job-manager.pl.

The full set of command-line options to globus-job-manager-event-generator consists of:
- help                  Print command-line option summary and exit.
- scheduler LRM         Process events for the local resource manager named by LRM.
- background            Run globus-job-manager-event-generator as a background process. It will fork a new process, print out its process ID and then the original process will terminate.
- pidfile PIDPATH        Write the process ID of an instance of globus-job-manager-event-generator to the file named by PIDPATH. This file can be used to kill or monitor the globus-job-manager-event-generator process.

Files
globus-job-manager-seg.conf  Configuration file for globus-job-manager-event-generator. Each line consists of a string of the form LRM_log_path=PATH, which indicates the directory containing LRM-independent format SEG log files for the LRM. This file is created by the running the globus_scheduler_event_generator_job_manager_setup setup package.

See Also
globus-scheduler-event-generator(8), globus-job-manager(8)
Name

globus-job-manager -- Execute and monitor jobs


Description

The globus-job-manager program is a service which starts and controls GRAM jobs which are executed by a local resource management system, such as LSF or Condor. The globus-job-manager program is typically started by the globus-gatekeeper program and not directly by a user. It runs until all jobs it is managing have terminated or its delegated credentials have expired.

Typically, users interact with the globus-job-manager program via client applications such as globusrun, globus-job-submit, or tools such as CoG jglobus or Condor-G.

The full set of command-line options to globus-job-manager consists of:

- **-help** Display a help message to standard error and exit
- **-type LRM** Execute jobs using the local resource manager named LRM.
- **-conf CONFIG_PATH** Read additional command-line arguments from the file CONFIG_PATH. If present, this must be the first command-line argument to the globus-job-manager program.
- **-globus-host-manufacturer MANUFACTURER** Indicate the manufacturer of the system which the jobs will execute on. This parameter sets the value of the $ (GLOBUS_HOST_MANUFACTURER) RSL substitution to MANUFACTURER
- **-globus-host-cputype CPU-TYPE** Indicate the CPU type of the system which the jobs will execute on. This parameter sets the value of the $ (GLOBUS_HOST_CPUTYPE) RSL substitution to CPU-TYPE
- **-globus-host-osname OSNAME** Indicate the operating system type of the system which the jobs will execute on. This parameter sets the value of the $ (GLOBUS_HOST_OSNAME) RSL substitution to OSNAME
- **-globus-host-osversion OSVERSION** Indicate the operating system version of the system which the jobs will execute on. This parameter sets the value of the $ (GLOBUS_HOST_OSVERSION) RSL substitution to OSVERSION
- **-globus-gatekeeper-host HOST** Indicate the host name of the machine which the job was submitted to. This parameter sets the value of the $ (GLOBUS_GATEKEEPER_HOST) RSL substitution to HOST
-globus-gatekeeper-port PORT
Indicate the TCP port number of gatekeeper to which jobs are submitted to. This parameter sets the value of the $(GLOBUS_GATEKEEPER_PORT) RSL substitution to PORT.

-globus-gatekeeper-subject SUBJECT
Indicate the X.509 identity of the gatekeeper to which jobs are submitted to. This parameter sets the value of the $(GLOBUS_GATEKEEPER_SUBJECT) RSL substitution to SUBJECT.

-home GLOBUS_LOCATION
Indicate the path where the Globus Toolkit(r) is installed on the service node. This is used by the job manager to locate its support and configuration files.

-target-globus-location TARGET_GLOBUS_LOCATION
Indicate the path where the Globus Toolkit(r) is installed on the execution host. If this is omitted, the value specified as a parameter to -home is used. This parameter sets the value of the $(GLOBUS_LOCATION) RSL substitution to TARGET_GLOBUS_LOCATION.

-history HISTORY_DIRECTORY
Configure the job manager to write job history files to HISTORY_DIRECTORY. These files are described in the FILES section below.

-scratch-dir-base SCRATCH_DIRECTORY
Configure the job manager to use SCRATCH_DIRECTORY as the default scratch directory root if a relative path is specified in the job RSL's scratch_dir attribute.

-enable-syslog
Configure the job manager to write log messages via syslog. Logging is further controlled by the argument to the -log.levels parameter described below.

-stdio-log LOG_DIRECTORY
Configure the job manager to write log messages to files in the LOG_DIRECTORY directory. Files will be named LOG_DIRECTORY/gram_YYYYMM-DD.log. Logging is further controlled by the argument to the -log.levels parameter described below. The LOG_DIRECTORY value can include variables derived from the job manager environment using the same syntax as RSL substitutions. For example, -stdio-log $(HOME) would cause each user's logs to be stored in their individual home directories.

-log.levels LEVELS
Configure the job manager to write log messages of certain levels to syslog and/or log files. The available log levels are FATAL, ERROR, WARN, INFO, DEBUG, and TRACE. Multiple values can be combined with the | character. The default value of logging when enabled is FATAL|ERROR.

-state-file-dir STATE_DIRECTORY
Configure the job manager to write state files to STATE_DIRECTORY. If not specified, the job manager uses the default of $GLOBUS_LOCATION/tmp/gram_job_state/. This directory must be writable by all users and be on a file system which supports POSIX advisory file locks.

-globus-tcp-port-range PORT_RANGE
Configure the job manager to restrict its TCP/IP communication to use ports in the range described by PORT_RANGE. This value is also made available in the job environment via the GLOBUS_TCP_PORT_RANGE environment variable.

-x509-cert-dir TRUSTED_CERTIFICATE_DIRECTORY
Configure the job manager to search TRUSTED_CERTIFICATE_DIRECTORY for its list of trusted CA certificates and their signing policies. This value is also made available in the job environment via the X509_CERT_DIR environment variable.
Configure the job manager to use the path `GASS_CACHE_DIRECTORY` for its temporary GASS-cache files. This value is also made available in the job environment via the `GLOBUS_GASS_CACHE_DEFAULT` environment variable.

Configure the job manager to assume it is using Kerberos for authentication instead of X.509 certificates. This disables some certificate-specific processing in the job manager.

Configure the job manager to define a set of environment variables in the job environment beyond those defined in the base job environment. The format of the parameter to this argument is a comma-separated sequence of `VAR=VAL` pairs, where `VAR` is the variable name and `VAL` is the variables value.

Configure the job manager to use the schedule event generator module named by `SEG_MODULE` to detect job state changes events from the local resource manager, in place of the less efficient polling operations used in GT2. To use this, one instance of the `globus-job-manager-event-generator` must be running to process events for the LRM into a generic format that the job manager can parse.

Configure the job manager to write audit records to the directory named by `AUDIT_DIRECTORY`. This records can be loaded into a database using the `globus-gram-audit` program.

Configure the job manager to use `TOOLKIT_VERSION` as the version for audit and usage stats records.

Configure the job manager to use `SERVICE_TAG` as a unique identifier to allow multiple GRAM instances to use the same job state directories without interfering with each other's jobs. If not set, the value `untagged` will be used.

Configure the job manager to disable file streaming. This is propagated to the LRM script interface but has no effect in GRAM5.

Disable sending of any usage stats data, even if `-usagestats-target` is present in the configuration.

Send usage packets to a data collection service for analysis. The `TARGET` string consists of a comma-separated list of `HOST:PORT` combinations, each containing an optional list of data to send. See Usage Stats Packets\(^1\) for more information about the tags. Special tag strings of `all` (which enables all tags) and `default` may be used, or a sequence of characters for the various tags.

Set the architecture specification for condor jobs to be `ARCH` in job classified ads generated by the GRAM5 condor LRM script. This is required for the condor LRM but ignored for all others.

Set the operating system specification for condor jobs to be `OS` in job classified ads generated by the GRAM5 condor LRM script. This is required for the condor LRM but ignored for all others.

\(^1\) [http://confluence.globus.org/display/~bester/GRAM5+Usage+Stats+Packets](http://confluence.globus.org/display/~bester/GRAM5+Usage+Stats+Packets)
Environment

If the following variables affect the execution of `globus-job-manager`

- **HOME**: User's home directory.
- **LOGNAME**: User's name.
- **JOBMANAGER_SYSLOG_ID**: String to prepend to syslog audit messages.
- **JOBMANAGER_SYSLOG_FAC**: Facility to log syslog audit messages as.
- **JOBMANAGER_SYSLOG_LVL**: Priority level to use for syslog audit messages.
- **GATEKEEPER_JM_ID**: Job manager ID to be used in syslog audit records.
- **GATEKEEPER_PEER**: Peer information to be used in syslog audit records.
- **GLOBUS_ID**: Credential information to be used in syslog audit records.
- **GLOBUS_JOB_MANAGER_SLEEP**: Time (in seconds) to sleep when the job manager is started. [For debugging purposes only]
- **GRID_SECURITY_HT-TP_BODY_FD**: File descriptor of an open file which contains the initial job request and to which the initial job reply should be sent. This file descriptor is inherited from the `globus-gatekeeper`.
- **X509_USER_PROXY**: Path to the X.509 user proxy which was delegated by the client to the `globus-gatekeeper` program to be used by the job manager.
- **GRID_SECURITY_CONTEXT_FD**: File descriptor containing an exported security context that the job manager should use to reply to the client which submitted the job.

Files

- **$HOME/.globus/job/HOST-NAME/LRM.TAG.red**: Job manager delegated user credential.
- **$HOME/.globus/job/HOST-NAME/LRM.TAG.lock**: Job manager state lock file.
- **$HOME/.globus/job/HOST-NAME/LRM.TAG.pid**: Job manager pid file.
- **$HOME/.globus/job/HOST-NAME/LRM.TAG.sock**: Job manager socket for inter-job manager communications.
- **$HOME/.globus/job/HOST-NAME/JOB_ID/**: Job-specific state directory.
- **$HOME/.globus/job/HOST-NAME/JOB_ID/stdin**: Standard input which has been staged from a remote URL.
- **$HOME/.globus/job/HOST-NAME/JOB_ID/stdout**: Standard output which will be staged from a remote URL.
$HOME/.globus/job/HOST-NAME/JOB_ID/stderr  
Standard error which will be staged from a remote URL.

$HOME/.globus/job/HOST-NAME/JOB_ID/x509_user_proxy  
Job-specific delegated credential.

$GLOBUS_LOCATION/tmp/gram_job_state/job.HOST-NAME.JOB_ID  
Job state file.

$GLOBUS_LOCATION/tmp/gram_job_state/job.HOST-NAME.JOB_ID.lock  
Job state lock file. In most cases this will be a symlink to the job manager lock file.

$GLOBUS_LOCATION/etc/globus-job-manager.conf  
Default location of the global job manager configuration file.

$GLOBUS_LOCATION/etc/grid-services/jobmanager-LRM  
Default location of the LRM-specific gatekeeper configuration file.

See Also

globusrun(1), globus-gatekeeper(8), globus-personal-gatekeeper(1), globus-gram-audit(8)
Chapter 11. Troubleshooting

For a list of error codes generated by GRAM5, see Section 2, “Errors”.

For information about sys admin logging, see Chapter 9, Admin Debugging in the GRAM5 Admin Guide.

1. Troubleshooting tips

In case you run into problems you can do the following

• Check the GRAM5 documentation. Maybe you'll find hints here to solve your problem.

• Check the GRAM5 log for errors.

  In case you don't find anything suspicious you can increase the log-level of GRAM5 or other relevant components. Maybe the additional logging-information will tell you what's going wrong.

• Send e-mails to <gram-user@globus.org>. You'll have to subscribe to a list before you can send an e-mail to it. See here\(^1\) for general e-mail lists and information on how to subscribe to a list and here\(^2\) for GRAM specific lists.

---

\(^1\) [http://dev.globus.org/wiki/Mailing_Lists](http://dev.globus.org/wiki/Mailing_Lists)

\(^2\) [http://dev.globus.org/wiki/GRAM#Mailing_Lists](http://dev.globus.org/wiki/GRAM#Mailing_Lists)
2. Errors
### Table 11.1. GRAM5 Errors

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Reason</th>
<th>Possible Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>one of the RSL parameters is not supported</td>
<td>Check RSL documentation</td>
</tr>
<tr>
<td>2</td>
<td>the RSL length is greater than the maximum allowed</td>
<td>Use RSL substitutions to reduce length of RSL strings</td>
</tr>
<tr>
<td>3</td>
<td>an I/O operation failed</td>
<td>Enable trace logging and report to <a href="mailto:gram-dev@globus.org">gram-dev@globus.org</a></td>
</tr>
<tr>
<td>4</td>
<td>jobmanager unable to set default to the directory requested</td>
<td>Check that RSL directory attribute refers to a directory that exists on the target system.</td>
</tr>
<tr>
<td>5</td>
<td>the executable does not exist</td>
<td>Check that the RSL executable attribute refers to an executable that exists on the target system.</td>
</tr>
<tr>
<td>6</td>
<td>of an unused INSUFFICIENT_FUNDS</td>
<td>Unimplemented feature</td>
</tr>
<tr>
<td>7</td>
<td>authentication with the remote server failed</td>
<td>Check that the contact string contains the proper X.509 DN.</td>
</tr>
<tr>
<td>8</td>
<td>the user cancelled the job</td>
<td>Don’t cancel jobs you want to complete.</td>
</tr>
<tr>
<td>9</td>
<td>the system cancelled the job</td>
<td>Check RSL requirements such as maximum time and memory are valid for the job.</td>
</tr>
<tr>
<td>10</td>
<td>data transfer to the server failed</td>
<td>Check gatekeeper and/or job manager logs to see why the process failed.</td>
</tr>
<tr>
<td>11</td>
<td>the stdin file does not exist</td>
<td>Check that the RSL stdin attribute refers to a file that exists on the target system or has a valid ftp, gsiftp, http, or https URL.</td>
</tr>
<tr>
<td>12</td>
<td>the connection to the server failed (check host and port)</td>
<td>Check that the service is running on the expected TCP/IP port. Check that no firewall prevents contacting that TCP/IP port. Check $GLOBUS_LOCATION/var/globus-gatekeeper.log for runtime configuration errors.</td>
</tr>
<tr>
<td>13</td>
<td>the provided RSL 'maxtime' value is not an integer</td>
<td>Check that the RSL maxtime value evaluates to an integer.</td>
</tr>
<tr>
<td>14</td>
<td>the provided RSL 'count' value is not an integer</td>
<td>Check that the RSL count value evaluates to an integer.</td>
</tr>
<tr>
<td>15</td>
<td>the job manager received an invalid RSL</td>
<td>Check that the RSL string can be parsed by using globusrun -p RSL.</td>
</tr>
<tr>
<td>16</td>
<td>the job manager failed in allowing others to make contact</td>
<td>Check job manager log.</td>
</tr>
<tr>
<td>17</td>
<td>the job failed when the job manager attempted to run it</td>
<td>Verify that the LRM is configured properly.</td>
</tr>
<tr>
<td>18</td>
<td>an invalid paradyn was specified</td>
<td>OBSOLETE IN GRAM2</td>
</tr>
<tr>
<td>19</td>
<td>the provided RSL 'jobtype' value is invalid</td>
<td>The RSL jobtype attribute is not indicated as supported by the LRM. Valid jobtype values are single, multiple, mpi, and condor.</td>
</tr>
<tr>
<td>20</td>
<td>the provided RSL 'myjob' value is invalid</td>
<td>OBSOLETE IN GRAM5</td>
</tr>
<tr>
<td>Error Code</td>
<td>Reason</td>
<td>Possible Solutions</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>-------------------</td>
</tr>
<tr>
<td>21</td>
<td>the job manager failed to locate an internal script argument file</td>
<td>Check that $GLOBUS_LOCATION/libexec/globus-job-manager-script.pl exists and is executable. Check that the LRM-specific perl module is located in $GLOBUS_LOCATION/lib/perl/globus/GRAM/JobManager/ directory and is valid. The command <code>perl -I$GLOBUS_LOCATION/lib/perl $GLOBUS_LOCATION/lib/perl/globus/GRAM/JobManager/LRM.pm</code> can be used to check if there are any syntax errors in the script.</td>
</tr>
<tr>
<td>22</td>
<td>the job manager failed to create an internal script argument file</td>
<td>Check that your home directory is writable and not full.</td>
</tr>
<tr>
<td>23</td>
<td>the job manager detected an invalid job state</td>
<td>Check job manager logs.</td>
</tr>
<tr>
<td>24</td>
<td>the job manager detected an invalid script response</td>
<td>Check job manager logs. This is likely a bug in the LRM script.</td>
</tr>
<tr>
<td>25</td>
<td>the job manager detected an invalid script status</td>
<td>Check job manager logs. This is likely a bug in the LRM script.</td>
</tr>
<tr>
<td>26</td>
<td>the provided RSL 'jobtype' value is not supported by this job manager</td>
<td>Check that the RSL jobtype attribute is implemented by the LRM script. Note that some job types require configuration.</td>
</tr>
<tr>
<td>27</td>
<td>unused ERROR_UNIMPLEMENTED</td>
<td>LRM does not support some feature included in the job request.</td>
</tr>
<tr>
<td>28</td>
<td>the job manager failed to create an internal script submission file</td>
<td>Check that the user's home file system is not full. Check job manager log.</td>
</tr>
<tr>
<td>29</td>
<td>the job manager cannot find the user proxy</td>
<td>Check that client is delegating a proxy when authenticating with the gatekeeper. Check that the user's home filesystem and the /tmp file system are not full.</td>
</tr>
<tr>
<td>30</td>
<td>the job manager failed to open the user proxy</td>
<td>Check that the user's home filesystem and the /tmp file system are not full.</td>
</tr>
<tr>
<td>31</td>
<td>the job manager failed to cancel the job as requested</td>
<td>Check that the user's home filesystem and the /tmp file system are not full.</td>
</tr>
<tr>
<td>32</td>
<td>system memory allocation failed</td>
<td>Check job manager log for details.</td>
</tr>
<tr>
<td>33</td>
<td>the interprocess job communication initialization failed</td>
<td>OBSOLETE IN GRAM5</td>
</tr>
<tr>
<td>34</td>
<td>the interprocess job communication setup failed</td>
<td>OBSOLETE IN GRAM5</td>
</tr>
<tr>
<td>35</td>
<td>the provided RSL 'host_count' value is invalid</td>
<td>Check that the RSL host_count attribute evaluates to an integer.</td>
</tr>
<tr>
<td>36</td>
<td>one of the provided RSL parameters is unsupported</td>
<td>Check job manager log for details about invalid parameter.</td>
</tr>
<tr>
<td>37</td>
<td>the provided RSL 'queue' parameter is invalid</td>
<td>Check that the RSL queue attribute evaluates to a string that corresponds to an LRM-specific queue name.</td>
</tr>
<tr>
<td>38</td>
<td>the provided RSL 'project' parameter is invalid</td>
<td>Check that the RSL project attribute evaluates to a string that corresponds to an LRM-specific project name.</td>
</tr>
</tbody>
</table>
### Troubleshooting

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Reason</th>
<th>Possible Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>39</td>
<td>the provided RSL string includes variables that could not be identified</td>
<td>Check that all RSL substitutions are defined before being used in the job description.</td>
</tr>
<tr>
<td>40</td>
<td>the provided RSL 'environment' parameter is invalid</td>
<td>Check that the RSL environment attribute contains a sequence of VARIABLE, VALUE pairs.</td>
</tr>
<tr>
<td>41</td>
<td>the provided RSL 'dryrun' parameter is invalid</td>
<td>Remove the RSL dryrun attribute from the job description.</td>
</tr>
<tr>
<td>42</td>
<td>the provided RSL is invalid (an empty string)</td>
<td>Include a non-empty RSL string in your job submission request.</td>
</tr>
<tr>
<td>43</td>
<td>the job manager failed to stage the executable</td>
<td>Check that the file service hosting the executable is reachable from the GRAM5 service node. Check that the executable exists on the file service node. Check that there is sufficient disk space in the user's home directory on the service node to store the executable.</td>
</tr>
<tr>
<td>44</td>
<td>the job manager failed to stage the stdin file</td>
<td>Check that the file service hosting the standard input file is reachable from the GRAM5 service node. Check that the standard input file exists on the file service node. Check that there is sufficient disk space in the user's home directory on the service node to store the standard input file.</td>
</tr>
<tr>
<td>45</td>
<td>the requested job manager type is invalid</td>
<td>OBSOLETE IN GRAM5</td>
</tr>
<tr>
<td>46</td>
<td>the provided RSL 'arguments' parameter is invalid</td>
<td>OBSOLETE IN GRAM2</td>
</tr>
<tr>
<td>47</td>
<td>the gatekeeper failed to run the job manager</td>
<td>Check the gatekeeper or job manager logs for more information.</td>
</tr>
<tr>
<td>48</td>
<td>the provided RSL could not be properly parsed</td>
<td>Check that the RSL string can be parsed by using <code>globusrun -p RSL</code>.</td>
</tr>
<tr>
<td>49</td>
<td>there is a version mismatch between GRAM components</td>
<td>Ask system administrator to upgrade GRAM service to GRAM2 or GRAM5</td>
</tr>
<tr>
<td>50</td>
<td>the provided RSL 'arguments' parameter is invalid</td>
<td>Check that the RSL arguments attribute evaluates to a sequence of strings.</td>
</tr>
<tr>
<td>51</td>
<td>the provided RSL 'count' parameter is invalid</td>
<td>Check that the RSL count attribute evaluates to a positive integer value.</td>
</tr>
<tr>
<td>52</td>
<td>the provided RSL 'directory' parameter is invalid</td>
<td>Check that the RSL directory attribute evaluates to a string.</td>
</tr>
<tr>
<td>53</td>
<td>the provided RSL 'dryrun' parameter is invalid</td>
<td>Check that the RSL dryrun attribute evaluates to either yes or no.</td>
</tr>
<tr>
<td>54</td>
<td>the provided RSL 'environment' parameter is invalid</td>
<td>Check that the RSL environment attribute evaluates to a sequence of VARIABLE, VALUE pairs.</td>
</tr>
<tr>
<td>55</td>
<td>the provided RSL 'executable' parameter is invalid</td>
<td>Check that the RSL executable attribute evaluates to a string value.</td>
</tr>
<tr>
<td>56</td>
<td>the provided RSL 'host_count' parameter is invalid</td>
<td>Check that the RSL host_count attribute evaluates to a positive integer value.</td>
</tr>
<tr>
<td>57</td>
<td>the provided RSL 'jobtype' parameter is invalid</td>
<td>Check that the RSL jobtype attribute evaluates to one of single, multiple, mpi, or condor</td>
</tr>
<tr>
<td>Error Code</td>
<td>Reason</td>
<td>Possible Solutions</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>--------------------</td>
</tr>
<tr>
<td>58</td>
<td>the provided RSL 'maxtime' parameter is invalid</td>
<td>Check that the RSL maxtime attribute evaluates to a positive integer value.</td>
</tr>
<tr>
<td>59</td>
<td>the provided RSL 'myjob' parameter is invalid</td>
<td>OBSOLETE IN GRAM5.</td>
</tr>
<tr>
<td>60</td>
<td>the provided RSL 'paradyn' parameter is invalid</td>
<td>OBSOLETE IN GRAM2.</td>
</tr>
<tr>
<td>61</td>
<td>the provided RSL 'project' parameter is invalid</td>
<td>Check that the RSL project attribute evaluates to a string value.</td>
</tr>
<tr>
<td>62</td>
<td>the provided RSL 'queue' parameter is invalid</td>
<td>Check that the RSL queue attribute evaluates to a string value.</td>
</tr>
<tr>
<td>63</td>
<td>the provided RSL 'stderr' parameter is invalid</td>
<td>Check that the RSL stderr attribute evaluates to a string value or a sequence of DESTINATION URLs with optional CACHE_TAG string parameters.</td>
</tr>
<tr>
<td>64</td>
<td>the provided RSL 'stdin' parameter is invalid</td>
<td>Check that the RSL stdin attribute evaluates to a string value.</td>
</tr>
<tr>
<td>65</td>
<td>the provided RSL 'stdout' parameter is invalid</td>
<td>Check that the RSL stdout attribute evaluates to a string value or a sequence of DESTINATION URLs with optional CACHE_TAG string parameters.</td>
</tr>
<tr>
<td>66</td>
<td>the job manager failed to locate an internal script</td>
<td>Check job manager log for more details.</td>
</tr>
<tr>
<td>67</td>
<td>the job manager failed on the system call pipe()</td>
<td>OBSOLETE IN GRAM5</td>
</tr>
<tr>
<td>68</td>
<td>the job manager failed on the system call fcntl()</td>
<td>OBSOLETE IN GRAM2</td>
</tr>
<tr>
<td>69</td>
<td>the job manager failed to create the temporary stdout filename</td>
<td>OBSOLETE IN GRAM5</td>
</tr>
<tr>
<td>70</td>
<td>the job manager failed to create the temporary stderr filename</td>
<td>OBSOLETE IN GRAM5</td>
</tr>
<tr>
<td>71</td>
<td>the job manager failed on the system call fork()</td>
<td>OBSOLETE IN GRAM2</td>
</tr>
<tr>
<td>72</td>
<td>the executable file permissions do not allow execution</td>
<td>Check that the RSL executable attribute refers to an executable program or script.</td>
</tr>
<tr>
<td>73</td>
<td>the job manager failed to open stdout</td>
<td>Check that the RSL stdout attribute refers to one or more valid destination files or URLs.</td>
</tr>
<tr>
<td>74</td>
<td>the job manager failed to open stderr</td>
<td>Check that the RSL stderr attribute refers to one or more valid destination files or URLs.</td>
</tr>
<tr>
<td>75</td>
<td>the cache file could not be opened in order to relocate the user proxy</td>
<td>Check that the user's home directory is writable and not full on the GRAM5 service node.</td>
</tr>
<tr>
<td>76</td>
<td>cannot access cache files in ~/.globus/.gass_cache, check permissions, quota, and disk space</td>
<td>Check that the user's home directory is writable and not full on the GRAM5 service node.</td>
</tr>
<tr>
<td>77</td>
<td>the job manager failed to insert the contact in the client contact list</td>
<td>Check job manager log</td>
</tr>
<tr>
<td>Error Code</td>
<td>Reason</td>
<td>Possible Solutions</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>--------------------</td>
</tr>
<tr>
<td>78</td>
<td>the contact was not found in the job-manager's client contact list</td>
<td>Don’t attempt to unregister callback contacts that are not registered</td>
</tr>
<tr>
<td>79</td>
<td>connecting to the job manager failed. Possible reasons: job terminated, invalid job contact, network problems, ...</td>
<td>Check that the job manager process is running. Check that the job manager credential has not expired. Check that the job manager contact refers to the correct TCP/IP host and port. Check that the job manager contact is not blocked by a firewall.</td>
</tr>
<tr>
<td>80</td>
<td>the syntax of the job contact is invalid</td>
<td>Check the syntax of job contact string.</td>
</tr>
<tr>
<td>81</td>
<td>the executable parameter in the RSL is undefined</td>
<td>Include the RSL executable in all job requests.</td>
</tr>
<tr>
<td>82</td>
<td>the job manager service is misconfigured. condor arch undefined</td>
<td>Add the <code>-condor-arch</code> to the command-line or configuration file for a job manager configured to use the condor LRM.</td>
</tr>
<tr>
<td>83</td>
<td>the job manager service is misconfigured. condor os undefined</td>
<td>Add the <code>-condor-os</code> to the command-line or configuration file for a job manager configured to use the condor LRM.</td>
</tr>
<tr>
<td>84</td>
<td>the provided RSL 'min_memory' parameter is invalid</td>
<td>Check that the RSL <code>min_memory</code> attribute evaluates to a positive integer value.</td>
</tr>
<tr>
<td>85</td>
<td>the provided RSL 'max_memory' parameter is invalid</td>
<td>Check that the RSL <code>max_memory</code> attribute evaluates to a positive integer value.</td>
</tr>
<tr>
<td>86</td>
<td>the RSL 'min_memory' value is not zero or greater</td>
<td>Check that the RSL <code>min_memory</code> attribute evaluates to a positive integer value.</td>
</tr>
<tr>
<td>87</td>
<td>the RSL 'max_memory' value is not zero or greater</td>
<td>Check that the RSL <code>max_memory</code> attribute evaluates to a positive integer value.</td>
</tr>
<tr>
<td>88</td>
<td>the creation of a HTTP message failed</td>
<td>Check job manager log.</td>
</tr>
<tr>
<td>89</td>
<td>parsing incoming HTTP message failed</td>
<td>Check job manager log.</td>
</tr>
<tr>
<td>90</td>
<td>the packing of information into a HTTP message failed</td>
<td>Check job manager log.</td>
</tr>
<tr>
<td>91</td>
<td>an incoming HTTP message did not contain the expected information</td>
<td>Check job manager log.</td>
</tr>
<tr>
<td>92</td>
<td>the job manager does not support the service that the client requested</td>
<td>Check that the client is talking to the correct service</td>
</tr>
<tr>
<td>93</td>
<td>the gatekeeper failed to find the requested service</td>
<td>OBSOLETE IN GRAM2</td>
</tr>
<tr>
<td>94</td>
<td>the job manager does not accept any new requests (shutting down)</td>
<td>Execute queries before the job has been cleaned up.</td>
</tr>
<tr>
<td>95</td>
<td>the client failed to close the listener associated with the callback URL</td>
<td>Call <code>globusgram_client_callback_disallow()</code> with a valid the callback contact.</td>
</tr>
<tr>
<td>96</td>
<td>the gatekeeper contact cannot be parsed</td>
<td>Check the syntax of the gatekeeper contact string you are attempting to contact.</td>
</tr>
<tr>
<td>97</td>
<td>the job manager could not find the 'poe' command</td>
<td>OBSOLETE IN GRAM2</td>
</tr>
<tr>
<td>98</td>
<td>the job manager could not find the 'mpirun' command</td>
<td>Configure the LRM script with <code>mpirun</code> in your path.</td>
</tr>
<tr>
<td>Error Code</td>
<td>Reason</td>
<td>Possible Solutions</td>
</tr>
<tr>
<td>------------</td>
<td>--------</td>
<td>--------------------</td>
</tr>
<tr>
<td>99</td>
<td>the provided RSL 'start_time' parameter is invalid</td>
<td>OBSOLETE IN GRAM2</td>
</tr>
<tr>
<td>100</td>
<td>the provided RSL 'reservation_handle' parameter is invalid</td>
<td>OBSOLETE IN GRAM2</td>
</tr>
<tr>
<td>101</td>
<td>the provided RSL 'max_wall_time' parameter is invalid</td>
<td>Check that the RSL max_wall_time attribute evaluates to a positive integer.</td>
</tr>
<tr>
<td>102</td>
<td>the RSL 'max_wall_time' value is not zero or greater</td>
<td>Check that the RSL max_wall_time attribute evaluates to a positive integer.</td>
</tr>
<tr>
<td>103</td>
<td>the provided RSL 'max_cpu_time' parameter is invalid</td>
<td>Check that the RSL max_cpu_time attribute evaluates to a positive integer.</td>
</tr>
<tr>
<td>104</td>
<td>the RSL 'max_cpu_time' value is not zero or greater</td>
<td>Check that the RSL max_cpu_time attribute evaluates to a positive integer.</td>
</tr>
<tr>
<td>105</td>
<td>the job manager is misconfigured, a scheduler script is missing</td>
<td>Check that the administrator has configured the LRM by running its setup script.</td>
</tr>
<tr>
<td>106</td>
<td>the job manager is misconfigured, a scheduler script has invalid permissions</td>
<td>Check that the administrator has installed the GLLOBUS_LOCATION/libexec/globus-job-manager-script.pl script. Check that the file system containing that script allows file execution.</td>
</tr>
<tr>
<td>107</td>
<td>the job manager failed to signal the job</td>
<td>OBSOLETE IN GRAM2</td>
</tr>
<tr>
<td>108</td>
<td>the job manager did not recognize/support the signal type</td>
<td>Check that your signal operation is using the correct signal constant.</td>
</tr>
<tr>
<td>109</td>
<td>the job manager failed to get the job id from the local scheduler</td>
<td>OBSOLETE IN GRAM2</td>
</tr>
<tr>
<td>110</td>
<td>the job manager is waiting for a commit signal</td>
<td>Send a two-phase commit signal to the job manager to acknowledge receiving the job contact from the job manager.</td>
</tr>
<tr>
<td>111</td>
<td>the job manager timed out while waiting for a commit signal</td>
<td>Send a two-phase commit signal to the job manager to acknowledge receiving the job contact from the job manager. Increase the two-phase commit time out for your job. Check that the job manager contact TCP/IP port is reachable from your client.</td>
</tr>
<tr>
<td>112</td>
<td>the provided RSL 'save_state' parameter is invalid</td>
<td>Check that the RSL save_state attribute is set to yes or no.</td>
</tr>
<tr>
<td>113</td>
<td>the provided RSL 'restart' parameter is invalid</td>
<td>Check that the RSL restart attribute evaluates to a string containing a job contact string.</td>
</tr>
<tr>
<td>114</td>
<td>the provided RSL 'two_phase' parameter is invalid</td>
<td>Check that the RSL two_phase attribute evaluates to a positive integer.</td>
</tr>
<tr>
<td>115</td>
<td>the RSL 'two_phase' value is not zero or greater</td>
<td>Check that the RSL two_phase attribute evaluates to a positive integer.</td>
</tr>
<tr>
<td>116</td>
<td>the provided RSL 'stdout_position' parameter is invalid</td>
<td>OBSOLETE IN GRAM5</td>
</tr>
<tr>
<td>117</td>
<td>the RSL 'stdout_position' value is not zero or greater</td>
<td>OBSOLETE IN GRAM5</td>
</tr>
<tr>
<td><strong>Error Code</strong></td>
<td><strong>Reason</strong></td>
<td><strong>Possible Solutions</strong></td>
</tr>
<tr>
<td>----------------</td>
<td>---------------------------------------------------------------------------</td>
<td>---------------------------------------------</td>
</tr>
<tr>
<td>118</td>
<td>the provided RSL 'stderr_position' parameter is invalid</td>
<td>OBSOLETE IN GRAM5</td>
</tr>
<tr>
<td>119</td>
<td>the RSL 'stderr_position' value is not zero or greater</td>
<td>OBSOLETE IN GRAM5</td>
</tr>
<tr>
<td>120</td>
<td>the job manager restart attempt failed</td>
<td>OBSOLETE IN GRAM2</td>
</tr>
<tr>
<td>121</td>
<td>the job state file doesn't exist</td>
<td>Check that the job contact you are trying to restart matches one that the job manager returned to you.</td>
</tr>
<tr>
<td>122</td>
<td>could not read the job state file</td>
<td>Check that the state file directory is not full.</td>
</tr>
<tr>
<td>123</td>
<td>could not write the job state file</td>
<td>Check that the state file directory is not full.</td>
</tr>
<tr>
<td>124</td>
<td>old job manager is still alive</td>
<td>Contact the returned job manager contact to manage the job you are trying to restart.</td>
</tr>
<tr>
<td>125</td>
<td>job manager state file TTL expired</td>
<td>OBSOLETE in GRAM2</td>
</tr>
<tr>
<td>126</td>
<td>it is unknown if the job was submitted</td>
<td>Check job manager log.</td>
</tr>
<tr>
<td>127</td>
<td>the provided RSL 'remote_io_url' parameter is invalid</td>
<td>Check that the RSL <code>remote_io_url</code> attribute evaluates to a string value.</td>
</tr>
<tr>
<td>128</td>
<td>could not write the remote io url file</td>
<td>Check that the user's home file system on the job manager service node is writable and not full.</td>
</tr>
<tr>
<td>129</td>
<td>the standard output/error size is different</td>
<td>Send a stdio update signal to redirect the job manager output to a new URL.</td>
</tr>
<tr>
<td>130</td>
<td>the job manager was sent a stop signal (job is still running)</td>
<td>Submit a restart request to monitor the job.</td>
</tr>
<tr>
<td>131</td>
<td>the user proxy expired (job is still running)</td>
<td>Generate a new proxy and then submit a restart request to monitor the job.</td>
</tr>
<tr>
<td>132</td>
<td>the job was not submitted by original job-manager</td>
<td>OBSOLETE IN GRAM2</td>
</tr>
<tr>
<td>133</td>
<td>the job manager is not waiting for that commit signal</td>
<td>Do not send a commit signal to a job that is not waiting for a commit signal.</td>
</tr>
<tr>
<td>134</td>
<td>the provided RSL scheduler specific parameter is invalid</td>
<td>Check the LRM-specific documentation to determine what values are legal for the RSL extensions implemented by the LRM.</td>
</tr>
<tr>
<td>135</td>
<td>the job manager could not stage in a file</td>
<td>Check that the file service hosting the file to stage is reachable from the GRAM5 service node. Check that the file to stage exists on the file service node. Check that there is sufficient disk space in the user's home directory on the service node to store the file to stage.</td>
</tr>
<tr>
<td>136</td>
<td>the scratch directory could not be created</td>
<td>Check that the directory named by the RSL <code>scratch_dir</code> attribute exists and is writable. Check that the directory named by the RSL <code>scratch_dir</code> attribute is not full.</td>
</tr>
<tr>
<td>137</td>
<td>the provided 'gass_cache' parameter is invalid</td>
<td>Check that the RSL <code>gass_cache</code> attribute evaluates to a string.</td>
</tr>
<tr>
<td>138</td>
<td>the RSL contains attributes which are not valid for job submission</td>
<td>Do not use restart- or signal-only RSL attributes when submitting a job.</td>
</tr>
<tr>
<td>Error Code</td>
<td>Reason</td>
<td>Possible Solutions</td>
</tr>
<tr>
<td>------------</td>
<td>------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>139</td>
<td>the RSL contains attributes which are not valid for stdio update</td>
<td>Do not use submit- or restart-only RSL attributes when sending a stdio update signal to a job.</td>
</tr>
<tr>
<td>140</td>
<td>the RSL contains attributes which are not valid for job restart</td>
<td>Do not use submit- or signal-only RSL attributes when restarting a job.</td>
</tr>
<tr>
<td>141</td>
<td>the provided RSL 'file_stage_in' parameter is invalid</td>
<td>Check that the RSL file_stage_in attribute evaluates to a sequence of SOURCEDESTINATION pairs.</td>
</tr>
<tr>
<td>142</td>
<td>the provided RSL 'file_stage_in_shared' parameter is invalid</td>
<td>Check that the RSL file_stage_in_shared attribute evaluates to a sequence of SOURCEDESTINATION pairs.</td>
</tr>
<tr>
<td>143</td>
<td>the provided RSL 'file_stage_out' parameter is invalid</td>
<td>Check that the RSL file_stage_out attribute evaluates to a sequence of SOURCEDESTINATION pairs.</td>
</tr>
<tr>
<td>144</td>
<td>the provided RSL 'gass_cache' parameter is invalid</td>
<td>Check that the RSL gass_cache attribute evaluates to a string.</td>
</tr>
<tr>
<td>145</td>
<td>the provided RSL 'file_cleanup' parameter is invalid</td>
<td>Check that the RSL file_clean_up attribute evaluates to a sequence of strings.</td>
</tr>
<tr>
<td>146</td>
<td>the provided RSL 'scratch_dir' parameter is invalid</td>
<td>Check that the RSL scratch_dir attribute evaluates to a string.</td>
</tr>
<tr>
<td>147</td>
<td>the provided scheduler-specific RSL parameter is invalid</td>
<td>Check the LRM-specific documentation to determine what values are legal for the RSL extensions implemented by the LRM.</td>
</tr>
<tr>
<td>148</td>
<td>a required RSL attribute was not defined in the RSL spec</td>
<td>Check that the RSL executable attribute is present in your job request RSL. Check that the RSL restart attributes is present in your restart RSL.</td>
</tr>
<tr>
<td>149</td>
<td>the gass_cache attribute points to an invalid cache directory</td>
<td>Check that the RSL gass_cache attribute evaluates to a directory that exists or can be created. Check that the user's home file system is writable and not full.</td>
</tr>
<tr>
<td>150</td>
<td>the provided RSL 'save_state' parameter has an invalid value</td>
<td>Check that the RSL save_state attribute has a value of yes or no.</td>
</tr>
<tr>
<td>151</td>
<td>the job manager could not open the RSL attribute validation file</td>
<td>Check that $GLOBUS_LOCATION/share/globus_gram_job_manager/globus-gram-job-manager.rvf is present and readable on the job manager service node. Check that $GLOBUS_LOCATION/share/globus_gram_job_manager/LRM.rvf is readable on the job manager service node if present.</td>
</tr>
<tr>
<td>152</td>
<td>the job manager could not read the RSL attribute validation file</td>
<td>Check that $GLOBUS_LOCATION/share/globus_gram_job_manager/globus-gram-job-manager.rvf is valid. Check that $GLOBUS_LOCATION/share/globus_gram_job_manager/LRM.rvf is valid if present.</td>
</tr>
<tr>
<td>153</td>
<td>the provided RSL 'proxy_timeout' is invalid</td>
<td>Check that RSL proxy_timeout attribute evaluates to a positive integer.</td>
</tr>
<tr>
<td>154</td>
<td>the RSL 'proxy_timeout' value is not greater than zero</td>
<td>Check that RSL proxy_timeout attribute evaluates to a positive integer.</td>
</tr>
<tr>
<td>Error Code</td>
<td>Reason</td>
<td>Possible Solutions</td>
</tr>
<tr>
<td>------------</td>
<td>------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>155</td>
<td>the job manager could not stage out a file</td>
<td>Check that the source file being staged exists on the job manager service node. Check that the directory of the destination file being staged exists on the file service node. Check that the directory of the destination file being staged is writable by the user. Check that the destination file service is reachable by the job manager service node.</td>
</tr>
<tr>
<td>156</td>
<td>the job contact string does not match any which the job manager is handling</td>
<td>Check that the job contact string matches one returned from a job request.</td>
</tr>
<tr>
<td>157</td>
<td>proxy delegation failed</td>
<td>Check that the job manager service node trusts the signer of your credential. Check that you trust the signer of the job manager service node's credential.</td>
</tr>
<tr>
<td>158</td>
<td>the job manager could not lock the state lock file</td>
<td>Check that the file system holding the job state directory supports POSIX advisory locking. Check that the job state directory is writable by the user on the service node. Check that the job state directory is not full.</td>
</tr>
<tr>
<td>159</td>
<td>an invalid globus_io_clientattr_t was used.</td>
<td>Check that you have initialized the globus_io_clientattr_t attribute prior to using it with the GRAM client API.</td>
</tr>
<tr>
<td>160</td>
<td>an null parameter was passed to the gram library</td>
<td>Check that you are passing legal values to all GRAM API calls.</td>
</tr>
<tr>
<td>161</td>
<td>the job manager is still streaming output</td>
<td>OBSOLETE IN GRAM5</td>
</tr>
<tr>
<td>162</td>
<td>the authorization system denied the request</td>
<td>Check with your GRAM system administrator to allow a particular certificate to be authorized.</td>
</tr>
<tr>
<td>163</td>
<td>the authorization system reported a failure</td>
<td>Check with your system administrator to verify that the authorization system is configured properly.</td>
</tr>
<tr>
<td>164</td>
<td>the authorization system denied the request - invalid job id</td>
<td>Check with your system administrator to verify that the authorization system is configured properly. Use a credential which is authorized to interact with a particular GRAM job.</td>
</tr>
<tr>
<td>165</td>
<td>the authorization system denied the request - not authorized to run the specified executable</td>
<td>Check with your system administrator to verify that the authorization system is configured properly. Use a credential which is authorized to interact with a particular GRAM job.</td>
</tr>
<tr>
<td>166</td>
<td>the provided RSL 'user_name' parameter is invalid.</td>
<td>Check that the RSL user_name attribute evaluates to a string.</td>
</tr>
<tr>
<td>167</td>
<td>the job is not running in the account named by the 'user_name' parameter.</td>
<td>Ask with the GRAM system administrator to add an authorization entry to allow your credential to run jobs as the specified user account.</td>
</tr>
</tbody>
</table>
Chapter 12. Usage statistics collection by the Globus Alliance

1. GRAM5-specific usage statistics

The following usage statistics are sent by default in a UDP packet (in addition to the GRAM component code, packet version, timestamp, and source IP address) at the end of each job.

- Job Manager Session ID
- dryrun used
- RSL Host Count
- Timestamp when job hit GLOBUS_GRAM_PROTOCOL_JOB_STATE_UNSUBMITTED
- Timestamp when job hit GLOBUS_GRAM_PROTOCOL_JOB_STATE_FILE_STAGE_IN
- Timestamp when job hit GLOBUS_GRAM_PROTOCOL_JOB_STATE_PENDING
- Timestamp when job hit GLOBUS_GRAM_PROTOCOL_JOB_STATE_ACTIVE
- Timestamp when job hit GLOBUS_GRAM_PROTOCOL_JOB_STATE_FAILED
- Timestamp when job hit GLOBUS_GRAM_PROTOCOL_JOB_STATE_FILE_STAGE_OUT
- Timestamp when job hit GLOBUS_GRAM_PROTOCOL_JOB_STATE_DONE
- Job Failure Code
- Number of times status is called
- Number of times register is called
- Number of times signal is called
- Number of times refresh is called
- Number of files named in file_clean_up RSL
- Number of files being staged in (including executable, stdin) from http servers
- Number of files being staged in (including executable, stdin) from https servers
- Number of files being staged in (including executable, stdin) from ftp servers
- Number of files being staged in (including executable, stdin) from gsiftp servers
- Number of files being staged into the GASS cache from http servers
- Number of files being staged into the GASS cache from https servers
- Number of files being staged into the GASS cache from ftp servers
- Number of files being staged into the GASS cache from gsiftp servers
- Number of files being staged out (including stdout and stderr) to http servers
- Number of files being staged out (including stdout and stderr) to https servers
- Number of files being staged out (including stdout and stderr) to ftp servers
- Number of files being staged out (including stdout and stderr) to gsiftp servers
- Bitmask of used RSL attributes (values are $2^{\text{id}}$ from the gram5_rsl_attributes table)
- Number of times unregister is called
- Value of the count RSL attribute
- Comma-separated list of string names of other RSL attributes not in the set defined in globus-gram-job-manager.rvf
- Job type string
- Number of times the job was restarted
- Total number of state callbacks sent to all clients for this job

The following information can be sent as well in a job status packet but it is not sent unless explicitly enabled by the system administrator:
- Value of the executable RSL attribute
- Value of the arguments RSL attribute
- IP address and port of the client that submitted the job
- User DN of the client that submitted the job

In addition to job-related status, the job manager sends information periodically about its execution status. The following information is sent by default in a UDP packet (in addition to the GRAM component code, packet version, timestamp, and source IP address) at job manager start and every 1 hour during the job manager lifetime:
- Job Manager Start Time
- Job Manager Session ID
- Job Manager Status Time
- Job Manager Version
- LRM
- Poll used
- Audit used
- Number of restarted jobs
- Total number of jobs
Usage statistics collection by the Globus Alliance

- Total number of failed jobs
- Total number of canceled jobs
- Total number of completed jobs
- Total number of dry-run jobs
- Peak number of concurrently managed jobs
- Number of jobs currently being managed
- Number of jobs currently in the UNSUBMITTED state
- Number of jobs currently in the STAGE_IN state
- Number of jobs currently in the PENDING state
- Number of jobs currently in the ACTIVE state
- Number of jobs currently in the STAGE_OUT state
- Number of jobs currently in the FAILED state
- Number of jobs currently in the DONE state

Also, please see our policy statement\(^1\) on the collection of usage statistics.

\(^1\) ../../../Usage_Stats.html
## Glossary

### C

**Condor**


### L

**LSF**

A job scheduler mechanism supported by GRAM.

For more information, see [http://www.platform.com/Products/Platform.LSF.Family/Platform.LSF/](http://www.platform.com/Products/Platform.LSF.Family/Platform.LSF/).

### S

**Scheduler Event Generator (SEG)**

The Scheduler Event Generator (SEG) is a program which uses scheduler-specific monitoring modules to generate job state change events. Depending on scheduler-specific requirements, the SEG may need to run with privileges to enable it to obtain scheduler event notifications. As such, one SEG runs per scheduler resource. For example, on a host which provides access to both PBS and fork jobs, two SEGs, running at (potentially) different privilege levels will be running. One SEG instance exists for any particular scheduled resource instance (one for all homogeneous PBS queues, one for all fork jobs, etc). The SEG is implemented in an executable called the globus-scheduler-event-generator, located in the Globus Toolkit's libexec directory.

---

4 [http://www.platform.com/Products/Platform.LSF.Family/Platform.LSF/](http://www.platform.com/Products/Platform.LSF.Family/Platform.LSF/)
Index

A
audit logging, 10

C
configuration interface, 2
  default local resource manager, 5
  disabling local resource manager adapter, 5
  lrm-specific, 3
  non-default, 3
  authorization, 3
  typical, 2
    LRM adapters, 2
configuring, 2
  default local resource manager, 5
  disabling local resource manager adapter, 5
  lrm-specific, 3
  non-default, 3
  authorization, 3
  typical, 2
    LRM adapters, 2

D
deploying, 6

E
errors, 27

I
installing
  prerequisites
    local scheduler, 1
    lrm adapter, 1

P
performance guide, 9
  server-side, 9

S
SEG, 8

T
troubleshooting, 26
  check documentation, 26
  errors, 26
  gram log, 26
  mailing lists, 26

U
usage statistics, 37