# Contents

1 GridWay DRMAA Library Module Index .................................................. 1
  1.1 GridWay DRMAA Library Modules ................................................... 1

2 GridWay DRMAA Library File Index ..................................................... 3
  2.1 GridWay DRMAA Library File List .................................................. 3

3 GridWay DRMAA Library Module Documentation ................................... 5
  3.1 SECTION 1.2 Preprocessor Directives for Handling String Output Arguments .......... 5
  3.2 SECTION 1.3 Preprocessor Directives for Control Operations ......................... 7
  3.3 SECTION 1.4 Preprocessor Directives for Job Template Compilation ..................... 10
  3.4 SECTION 1.5 Preprocessor Directives for DRMAA Error Codes ............................ 14
  3.5 SECTION 1.6 Gridway Specific Preprocessor Directives for Job Template Compilation .... 17
  3.6 SECTION 2 String List Helper Functions ........................................... 20
  3.7 SECTION 3 Session Management Functions .......................................... 24
  3.8 SECTION 4 Job Template Functions .................................................. 25
  3.9 SECTION 5 Job Submission Functions ............................................... 30
  3.10 SECTION 6 Job Status and Control Functions ...................................... 32
  3.11 SECTION 7 Job Synchronize and Wait Functions .................................... 35
  3.12 SECTION 8 Auxiliary Functions for Interpreting Wait Status Code ..................... 37
  3.13 SECTION 9 Auxiliary Functions ..................................................... 40

4 GridWay DRMAA Library File Documentation ......................................... 43
  4.1 drmaa.h File Reference .............................................................. 43
Chapter 1

GridWay DRMAA Library Module

Index

1.1 GridWay DRMAA Library Modules

Here is a list of all modules:

SECTION 1.2 Preprocessor Directives for Handling String Output Arguments . . . . . . . . . . . . . . 5
SECTION 1.3 Preprocessor Directives for Control Operations . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7
SECTION 1.4 Preprocessor Directives for Job Template Compilation . . . . . . . . . . . . . . . . . . . . . . . 10
SECTION 1.5 Preprocessor Directives for DRMAA Error Codes . . . . . . . . . . . . . . . . . . . . . . . . . . . 14
SECTION 1.6 Gridway Specific Preprocessor Directives for Job Template Compilation . . . . . . . . . . . 17
SECTION 2 String List Helper Functions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20
SECTION 3 Session Management Functions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 24
SECTION 4 Job Template Functions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 25
SECTION 5 Job Submission Functions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 30
SECTION 6 Job Status and Control Functions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 32
SECTION 7 Job Synchronize and Wait Functions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 35
SECTION 8 Auxiliary Functions for Interpreting Wait Status Code . . . . . . . . . . . . . . . . . . . . . . . . . . . . 37
SECTION 9 Auxiliary Functions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 40
Chapter 2

GridWay DRMAA Library File Index

2.1 GridWay DRMAA Library File List

Here is a list of all documented files with brief descriptions:

- drmaa.h

43
Chapter 3

GridWay DRMAA Library Module Documentation

3.1 SECTION 1.2 Preprocessor Directives for Handling String Output Arguments

Defines

• #define DRMAA_ATTR_BUFFER 1024
• #define DRMAA_CONTACT_BUFFER 1024
• #define DRMAA_DRM_SYSTEM_BUFFER 1024
• #define DRMAA_DRMAA_IMPL_BUFFER 1024
• #define DRMAA_ERROR_STRING_BUFFER 1024
• #define DRMAA_JOBNAME_BUFFER 1024
• #define DRMAA_SIGNAL_BUFFER 32

3.1.1 Define Documentation

3.1.1.1 #define DRMAA_ATTR_BUFFER 1024

Pre-defined buffer size for attribute variables which may be used in DRMAA programs in the creation of char* variables. The DRMAA_ATTR_BUFFER directive is the default length for attribute values (drmaa_attr_values_t) allocated by DRMAA functions.

3.1.1.2 #define DRMAA_CONTACT_BUFFER 1024

Pre-defined buffer size for contact string.

3.1.1.3 #define DRMAA_DRM_SYSTEM_BUFFER 1024

Pre-defined buffer size for Distributed Resource Management System (DRMS) string.
3.1.1.4  #define DRMAA_DRMAA_IMPL_BUFFER 1024
Pre-defined buffer size for drmaa_impl string.

3.1.1.5  #define DRMAA_ERROR_STRING_BUFFER 1024
Pre-defined buffer size for error string variables which may be used in DRMAA programs in the creation
of char* error variables.

3.1.1.6  #define DRMAA_JOBNAME_BUFFER 1024
Pre-defined buffer size for the job identification string variables. Job identification should not be smaller
than DRMAA_JOBNAME_BUFFER. If the size of the string passed is smaller the resultant job id string
will be truncated. Note that for GridWay, JOBNAME strings length will not be greater than 5.

3.1.1.7  #define DRMAA_SIGNAL_BUFFER 32
Pre-defined buffer size for the signal name returned by drmaa_wtermsig
3.2 SECTION 1.3 Preprocessor Directives for Control Operations

Defines

- \#define DRMAA_TIMEOUT_WAIT_FOREVER -1
- \#define DRMAA_TIMEOUT_NO_WAIT 0
- \#define DRMAA_PS_UNDETERMINED 0x00
- \#define DRMAA_PS_QUEUED_ACTIVE 0x10
- \#define DRMAA_PS_SYSTEM_ON_HOLD 0x11
- \#define DRMAA_PS_USER_ON_HOLD 0x12
- \#define DRMAA_PS_USER_SYSTEM_ON_HOLD 0x13
- \#define DRMAA_PS_RUNNING 0x20
- \#define DRMAA_PS_SYSTEM_SUSPENDED 0x21
- \#define DRMAA_PS_USER_SUSPENDED 0x22
- \#define DRMAA_PS_USER_SYSTEM_SUSPENDED 0x23
- \#define DRMAA_PS_DONE 0x30
- \#define DRMAA_PS_FAILED 0x40
- \#define DRMAA_CONTROL_SUSPEND 0
- \#define DRMAA_CONTROL_RESUME 1
- \#define DRMAA_CONTROL_HOLD 2
- \#define DRMAA_CONTROL_RELEASE 3
- \#define DRMAA_CONTROL_TERMINATE 4
- \#define DRMAA_JOB_IDS_SESSION_ANY "DRMAA_JOB_IDS_SESSION_ANY"
- \#define DRMAA_JOB_IDS_SESSION_ALL "DRMAA_JOB_IDS_SESSION_ALL"

3.2.1 Define Documentation

3.2.1.1 \#define DRMAA_CONTROL_HOLD 2

HOLD signal. A job can be held if it is in the QUEUED_ACTIVE state, and on SUCCESS will enter the USER_ON_HOLD state.

3.2.1.2 \#define DRMAA_CONTROL_RELEASE 3

RELEASE signal. Release a previously held job, only jobs in the USER_ON_HOLD state can be released. On SUCCESS the job will enter the QUEUED_ACTIVE state.

3.2.1.3 \#define DRMAA_CONTROL_RESUME 1

RESUME signal. A previously stopped job will be resumed. If re-start files are provided the job will use them to re-start execution from the last checkpointing context.

3.2.1.4 \#define DRMAA_CONTROL_SUSPEND 0

SUSPEND signal. A job will be stopped, and restart files transferred back to the client. These files if provided by the running job will be used on RESUME to re-start execution.
3.2.1.5  #define DRMAA_CONTROL_TERMINATE 4

TERMINATE signal. The job will be killed, it execution can be synchronized through the drmaa_wait() and drmaa_synchronize() function calls. However, job rusage information will not be available and these functions will return DRMAA_ERRNO_NO_RUSAGE.

3.2.1.6  #define DRMAA_JOB_IDS_SESSION_ALL "DRMAA_JOB_IDS_SESSION_ALL"

Pre-defined string used to refer to ALL the jobs submitted during a DRMAA session. Please note that "disposed" jobs will be removed from the job-list associated to the DRMAA session.

3.2.1.7  #define DRMAA_JOB_IDS_SESSION_ANY "DRMAA_JOB_IDS_SESSION_ANY"

Pre-defined string used to refer to ANY job submitted during a DRMAA session. Please note that "disposed" jobs will be removed from the job-list associated to the DRMAA session.

3.2.1.8  #define DRMAA_PS_DONE 0x30

DONE Job state. Job has been completely executed and output files are available at the client. This state corresponds to the ZOMBIE state in the GridWay system. drmaa_wait() and drmaa_synchronize() calls on the job will return immediately. Also rusage information is available.

3.2.1.9  #define DRMAA_PS_FAILED 0x40

FAILED Job state. Job execution has failed, and the "on_failure" policy is to hold it on FAILED state. This state corresponds to the FAILED state in the GridWay system.

3.2.1.10 #define DRMAA_PS_QUEUED_ACTIVE 0x10

QUEUED-ACTIVE Job state. The job has been successfully submitted and it is pending to be scheduled. This state corresponds to the PENDING state in the GridWay system.

3.2.1.11 #define DRMAA_PS_RUNNING 0x20

RUNNING Job state. The job has been successfully scheduled and dispatched to a remote host. Please note that once submitted, the job can be in any of the execution states, namely: PROLOG (file stage-in), WRAPPER (execution), EPILOG (file stage-out) or MIGRATING (to another host).

3.2.1.12 #define DRMAA_PS_SYSTEM_ON_HOLD 0x11

SYSTEM-ON-HOLD Job state. The GridWay system does NOT DEFINE a SYSTEM-ON-HOLD state (currently), and so it will not be never returned by a drmaa_job_ps() call.

3.2.1.13 #define DRMAA_PS_SYSTEM_SUSPENDED 0x21

SYSTEM-SUSPENDED Job state. The GridWay system does NOT DEFINE a SYSTEM-SUSPENDED state, and so it will not be never returned by a drmaa_job_ps() call.
3.2 SECTION 1.3 Preprocessor Directives for Control Operations

3.2.1.14 #define DRMAA_PS_UNDETERMINED 0x00

UNDETERMINED Job state. An UNDETERMINED state can either be obtained due to a communication error with the GridWay daemon, or because the job has not been initialized yet.

3.2.1.15 #define DRMAA_PS_USER_ON_HOLD 0x12

USER-ON-HOLD Job state. The job has been held by the user. This state corresponds to the HOLD state in the GridWay system.

3.2.1.16 #define DRMAA_PS_USER_SUSPENDED 0x22

USER-SUSPENDED Job state. The job has been successfully stopped. This state corresponds to the STOPPED state in the GridWay system. Once stopped, restart files (if provided by the job) have been transferred to the client.

3.2.1.17 #define DRMAA_PS_USER_SYSTEM_ON_HOLD 0x13

USER-SYSTEM-ON-HOLD Job state. The GridWay system does NOT DEFINE a USER-SYSTEM-ON-HOLD state, and so it will not be never returned by a drmaa_job_ps() call.

3.2.1.18 #define DRMAA_PS_USER_SYSTEM_SUSPENDED 0x23

USER-SYSTEM-SUSPENDED Job state. The GridWay system does NOT DEFINE a USER-SYSTEM-SUSPENDED state, and so it will not be never returned by a drmaa_job_ps() call.

3.2.1.19 #define DRMAA_TIMEOUT_NO_WAIT 0

Pre-defined timeout to be used with drmaa_wait() and drmaa_synchronize() function calls. DRMAA_-TIMEOUT_NO_WAIT, can be used to specify no timeout at all.

3.2.1.20 #define DRMAA_TIMEOUT_WAIT_FOREVER -1

Pre-defined timeout to be used with drmaa_wait() and drmaa_synchronize() function calls. DRMAA_-TIMEOUT_WAIT_FOREVER, can be used to specify an undetermined amount of time.
3.3 SECTION 1.4 Preprocessor Directives for Job Template Compilation

Defines

- #define DRMAA_REMOTE_COMMAND "drmaa_remote_command"
- #define DRMAA_V_ARGV "drmaa_v_argv"
- #define DRMAA_V_ENV "drmaa_v_env"
- #define DRMAA_INPUT_PATH "drmaa_input_path"
- #define DRMAA_OUTPUT_PATH "drmaa_output_path"
- #define DRMAA_ERROR_PATH "drmaa_error_path"
- #define DRMAA_WD "drmaa_wd"
- #define DRMAA_JOB_NAME "drmaa_job_name"
- #define DRMAA_JS_STATE "drmaa_js_state"
- #define DRMAA_SUBMISSION_STATE_ACTIVE "drmaa_active"
- #define DRMAA_SUBMISSION_STATE_HOLD "drmaa_hold"
- #define DRMAA_PLACEHOLDER_HD "$drmaa_hd_ph$"
- #define DRMAA_PLACEHOLDER_INCR "$drmaa_incr_ph$"
- #define DRMAA_PLACEHOLDER_WD "$drmaa_wd_ph$"
- #define DRMAA_DEADLINE_TIME "drmaa_deadline_time"
- #define DRMAA_DEADLINE_TIME "drmaa_deadline_time"
- #define DRMAA_BLOCK_EMAIL "drmaa_block_email"
- #define DRMAA_DURATION_HLIMIT "drmaa_duration_hlimit"
- #define DRMAA_DURATION_SLIMIT "drmaa_duration_slimit"
- #define DRMAA_JOB_CATEGORY "drmaa_job_category"
- #define DRMAA_JOIN_FILES "drmaa_join_files"
- #define DRMAA_NATIVE_SPECIFICATION "drmaa_native_specification"
- #define DRMAA_START_TIME "drmaa_start_time"
- #define DRMAATRANSFER_FILES "drmaa_transfer_files"
- #define DRMAA_V_EMAIL "drmaa_v_email"
- #define DRMAA_WCT_HLIMIT "drmaa_wct_hlimit"
- #define DRMAA_WCT_SLIMIT "drmaa_wct_slimit"

3.3.1 Define Documentation

3.3.1.1 #define DRMAA_BLOCK_EMAIL "drmaa_block_email"

Not relevant for the current GridWay implementation, will be ignored

3.3.1.2 #define DRMAA_DEADLINE_TIME "drmaa_deadline_time"

Pre-defined string to represent a deadline for job execution. GridWay WILL NOT terminate a job after the deadline, neither guarantees that the job is executed before the deadline. A deadline is specified relative to job submission time, in the form: [DD:]HH:MM where, DD is the number of days HH is the number of hours MM is the number of minutes

Example: 01:22 The job should finished one hour and 22 minutes after submission.

NOTE: The use of the DEADLINE_TIME as described here differs from the one specified in the standard (v1.0).
3.3.1.3  #define DRMAA_DEADLINE_TIME "drmaa_deadline_time"

Pre-defined string to represent a deadline for job execution. GridWay WILL NOT terminate a job after the
deadline, neither guarantees that the job is executed before the deadline. A deadline is specified relative to
job submission time, in the form: [[DD:]][HH:]MM where, DD is the number of days HH is the number
of hours MM is the number of minutes
Example: 01:22 The job should finished one hour and 22 minutes after submission.

NOTE: The use of the DEADLINE_TIME as described here differs from the one specified in the standard
(v1.0).

3.3.1.4  #define DRMAA_DURATION_HLIMIT "drmaa_duration_hlimit"

Not relevant for the current GridWay implementation, will be ignored

3.3.1.5  #define DRMAA_DURATION_SLIMIT "drmaa_duration_slimit"

Not relevant for the current GridWay implementation, will be ignored

3.3.1.6  #define DRMAA_ERROR_PATH "drmaa_error_path"

Pre-defined string to refer to standard error file for the DRMAA_REMOTE_COMMAND. The standard
input file IS RELATIVE TO THE WORKING DIRECTORY.

3.3.1.7  #define DRMAA_INPUT_PATH "drmaa_input_path"

Pre-defined string to refer to standard input file for the DRMAA_REMOTE_COMMAND. The standard
input file IS RELATIVE TO THE WORKING DIRECTORY.

3.3.1.8  #define DRMAA_JOB_CATEGORY "drmaa_job_category"

Not relevant for the current GridWay implementation, will be ignored

3.3.1.9  #define DRMAA_JOB_NAME "drmaa_job_name"

Pre-defined string to refer to the DRMAA job-name. The current GridWay DRMAA implementation will
generate a job template file with name DRMAA_JOB_NAME in the job working directory (DRMAA-_WD). DRMAA_JOB_NAME is a MANDATORY attribute value and MUST BE DEFINED. The default
value is "job_template".

3.3.1.10 #define DRMAA_JOIN_FILES "drmaa_join_files"

Not relevant for the current GridWay implementation, will be ignored

3.3.1.11 #define DRMAA_JS_STATE "drmaa_js_state"

Pre-defined string to refer to the job state at submission, the job will enter either the QUEUED_ACTIVE
state or HOLD state when submitted. The preprocessor directives DRMAA_SUBMISSION_STATE_
ACTIVE and DRMAA_SUBMISSION_STATE_HOLD SHOULD be used to assign the value of this attribute. The default value for DRMAA_JS_STATE is ACTIVE.

3.3.1.12 #define DRMAA_NATIVE_SPECIFICATION "drmaa_native_specification"

Not relevant for the current GridWay implementation, will be ignored

3.3.1.13 #define DRMAA_OUTPUT_PATH "drmaa_output_path"

Pre-defined string to refer to standard output file for the DRMAA_REMOTE_COMMAND. The standard input file IS RELATIVE TO THE WORKING DIRECTORY.

3.3.1.14 #define DRMAA_PLACEHOLDER_HD "$drmaa_hd_ph$"

Pre-defined string to refer the user's home directory.

3.3.1.15 #define DRMAA_PLACEHOLDER_INCR "$drmaa_incr_ph$"

Pre-defined string to be used in parametric jobs (bulk jobs). DRMAA_PLACEHOLDER_INCR will be available during job execution and can be used as an ARGUMENT for the REMOTE COMMAND, or to generate output filenames. Please note that this attribute name should be used ONLY in conjunction with a drmaa_run_bulk_jobs function call. Use DRMAA_GW_JOB_ID for "stand-alone" jobs.

3.3.1.16 #define DRMAA_PLACEHOLDER_WD "$drmaa_wd_ph$"

Pre-defined string constant to represent the current working directory when building paths for the input, output, and error path attribute values. Please note that ALL FILES MUST BE NAMED RELATIVE TO THE WORKING DIRECTORY.

3.3.1.17 #define DRMAA_REMOTE_COMMAND "drmaa_remote_command"

Pre-defined string to refer to the command to be executed on the remote host. DRMAA_REMOTE_COMMAND can be relative to the working directory (DRMAA_WD) or an absolute filename (will not be transferred). Architecture-dependent DRMAA_REMOTE_COMMAND can be generated with DRMAA_GW_ARCH.

3.3.1.18 #define DRMAA_START_TIME "drmaa_start_time"

Not relevant for the current GridWay implementation, will be ignored

3.3.1.19 #define DRMAA_SUBMISSION_STATE_ACTIVE "drmaa_active"

Pre-defined string to refer to the ACTIVE state on submission. Use this preprocessor directive to assign the value of the DRMAA_JS_STATE attribute through the drmaa_set_attribute() function call.
3.3.1.20 \#define DRMAA_SUBMISSION_STATE_HOLD "drmaa_hold"

Pre-defined string to refer to the HOLD state on submission. Use this preprocessor directive to assign the value of the DRMAA_JS_STATE attribute through the drmaa_set_attribute() function call.

3.3.1.21 \#define DRMAA_TRANSFER_FILES "drmaa_transfer_files"

Not relevant for the current GridWay implementation, will be ignored

3.3.1.22 \#define DRMAA_V_ARGV "drmaa_v_argv"

Pre-defined string to refer to the DRMAA_REMOTE_COMMAND arguments. DRMAA_V_ARGV corresponds to a NULL terminated vector attribute value.

3.3.1.23 \#define DRMAA_V_EMAIL "drmaa_v_email"

Not relevant for the current GridWay implementation, will be ignored

3.3.1.24 \#define DRMAA_V_ENV "drmaa_v_env"

Pre-defined string to refer to the DRMAA_REMOTE_COMMAND environment variables. DRMAA_V_ENV corresponds to a NULL terminated vector attribute value.

3.3.1.25 \#define DRMAA_WCT_HLIMIT "drmaa_wct_hlimit"

Not relevant for the current GridWay implementation, will be ignored

3.3.1.26 \#define DRMAA_WCT_SLIMIT "drmaa_wct_slimit"

Not relevant for the current GridWay implementation, will be ignored

3.3.1.27 \#define DRMAA_WD "drmaa_wd"

Pre-defined string to refer to the job working directory. The GridWay DRMAA implementation will generate a job template file with name DRMAA_JOB_NAME in the job working directory (DRMAA_WD). It is a MANDATORY attribute value and MUST BE DEFINED. Please note that ALL FILES ARE NAMED RELATIVE TO THE WORKING DIRECTORY. Also this is a LOCAL PATH NAME, this directory will be "recreated" in the remote host, and it will be the working directory of the job on the execution host. The default value is DRMAA_PLACEHOLDER_WD.
3.4 SECTION 1.5 Preprocessor Directives for DRMAA Error Codes

Defines

- #define DRMAA_ERRNO_SUCCESS 0
- #define DRMAA_ERRNO_INTERNAL_ERROR 1
- #define DRMAA_ERRNO_DRM_COMMUNICATION_FAILURE 2
- #define DRMAA_ERRNO_AUTH_FAILURE 3
- #define DRMAA_ERRNO_INVALID_ARGUMENT 4
- #define DRMAA_ERRNO_NO_ACTIVE_SESSION 5
- #define DRMAA_ERRNO_NO_MEMORY 6
- #define DRMAA_ERRNO_INVALID_CONTACT_STRING 7
- #define DRMAA_ERRNO_DEFAULT_CONTACT_STRING_ERROR 8
- #define DRMAA_ERRNO_DRMS_INIT_FAILED 9
- #define DRMAA_ERRNO_ALREADY_ACTIVE_SESSION 10
- #define DRMAA_ERRNO_DRMS_EXIT_ERROR 11
- #define DRMAA_ERRNO_INVALID_ATTRIBUTE_FORMAT 12
- #define DRMAA_ERRNO_INVALID_ATTRIBUTE_VALUE 13
- #define DRMAA_ERRNO_CONFLICTING_ATTRIBUTE_VALUES 14
- #define DRMAA_ERRNO_TRY_LATER 15
- #define DRMAA_ERRNO_DENIED_BY_DRM 16
- #define DRMAA_ERRNO_INV_ID_ATTRIBUTE_FORMAT 17
- #define DRMAA_ERRNO_RESUME_INCONSISTENT_STATE 18
- #define DRMAA_ERRNO_SUSPEND_INCONSISTENT_STATE 19
- #define DRMAA_ERRNO_HOLD_INCONSISTENT_STATE 20
- #define DRMAA_ERRNO_RELEASE_INCONSISTENT_STATE 21
- #define DRMAA_ERRNO_EXIT_TIMEOUT 22
- #define DRMAA_ERRNO_NO_RUSAGE 23
- #define DRMAA_ERRNO_NO_MORE_ELEMENTS 24

3.4.1 Define Documentation

3.4.1.1 #define DRMAA_ERRNO_ALREADY_ACTIVE_SESSION 10

A DRMAA session was already initialized

3.4.1.2 #define DRMAA_ERRNO_AUTH_FAILURE 3

Permission denied

3.4.1.3 #define DRMAA_ERRNO_CONFLICTING_ATTRIBUTE_VALUES 14

Conflicting attribute values

3.4.1.4 #define DRMAA_ERRNO_DEFAULT_CONTACT_STRING_ERROR 8

Default contact string error
3.4 SECTION 1.5 Preprocessor Directives for DRMAA Error Codes

3.4.1.5  #define DRMAA_ERRNO_DENIED_BY_DRM 16
Permission denied

3.4.1.6  #define DRMAA_ERRNO_DRM_COMMUNICATION_FAILURE 2
Could not contact with GWD

3.4.1.7  #define DRMAA_ERRNO_DRMS_EXIT_ERROR 11
Could not close connection with GWD

3.4.1.8  #define DRMAA_ERRNO_DRMS_INIT_FAILED 9
Unable to initialize GWD

3.4.1.9  #define DRMAA_ERRNO_EXIT_TIMEOUT 22
Time out exceeded

3.4.1.10 #define DRMAA_ERRNO_HOLD_INCONSISTENT_STATE 20
Could not hold job: wrong job state

3.4.1.11 #define DRMAA_ERRNO_INTERNAL_ERROR 1
Unexpected Error

3.4.1.12 #define DRMAA_ERRNO_INVALID_ARGUMENT 4
Invalid Argument

3.4.1.13 #define DRMAA_ERRNO_INVALID_ATTRIBUTE_FORMAT 12
Invalid attribute format

3.4.1.14 #define DRMAA_ERRNO_INVALID_ATTRIBUTE_VALUE 13
Invalid attribute value

3.4.1.15 #define DRMAA_ERRNO_INVALID_CONTACT_STRING 7
Invalid contact string
3.4.1.16  #define DRMAA_ERRNO_INVALID_JOB 17
Invalid Job ID/ it does not exist

3.4.1.17  #define DRMAA_ERRNO_NO_ACTIVE_SESSION 5
No active session

3.4.1.18  #define DRMAA_ERRNO_NO_MEMORY 6
Not enough memory

3.4.1.19  #define DRMAA_ERRNO_NO_MORE_ELEMENTS 24
No more elements (NOT IN 1.0 STANDARD!!)

3.4.1.20  #define DRMAA_ERRNO_NO_RUSAGE 23
RUSAGE not available

3.4.1.21  #define DRMAA_ERRNO_RELEASE_INCONSISTENT_STATE 21
Could not release job: wrong job state

3.4.1.22  #define DRMAA_ERRNO_RESUME_INCONSISTENT_STATE 18
Could not resume job: wrong job state

3.4.1.23  #define DRMAA_ERRNO_SUCCESS 0
Success

3.4.1.24  #define DRMAA_ERRNO_SUSPEND_INCONSISTENT_STATE 19
Could not suspend job: wrong job state

3.4.1.25  #define DRMAA_ERRNO_TRY_LATER 15
Try later (max. number of jobs reached)...

3.5 SECTION 1.6 Gridway Specific Preprocessor Directives for Job Template Compilation

Defines

- `#define DRMAA_GW_TOTAL_TASKS "${TOTAL_TASKS}"`
- `#define DRMAA_GW_JOB_ID "${JOB_ID}"`
- `#define DRMAA_GW_TASK_ID "${TASK_ID}"`
- `#define DRMAA_GW_PARAM "${PARAM}"`
- `#define DRMAA_GW_MAX_PARAM "${MAX_PARAM}"`
- `#define DRMAA_GW_ARCH "${ARCH}"`
- `#define DRMAA_V_GW_INPUT_FILES "INPUT_FILES"`
- `#define DRMAA_V_GW_OUTPUT_FILES "OUTPUT_FILES"`
- `#define DRMAA_V_GW_RESTART_FILES "RESTART_FILES"`
- `#define DRMAA_GW_RESCHEDULE_ON_FAILURE "RESCHEDULE_ON_FAILURE"`
- `#define DRMAA_GW_NUMBER_OF_RETRIES "NUMBER_OF_RETRIES"`
- `#define DRMAA_GW_RANK "RANK"`
- `#define DRMAA_GW_REQUIREMENTS "REQUIREMENTS"`
- `#define DRMAA_GW_TYPE "TYPE"`
- `#define DRMAA_GW_TYPE_SINGLE "single"`
- `#define DRMAA_GW_TYPE_MPI "mpi"`
- `#define DRMAA_GW_NP "NP"`

3.5.1 Define Documentation

3.5.1.1 `#define DRMAA_GW_ARCH "${ARCH}"`

Pre-defined string to refer to the remote host architecture as returned by the resource selector module. DRMAA_GW_ARCH will be available during job execution and can be used to generate architecture-dependent REMOTE COMMAND executables.

3.5.1.2 `#define DRMAA_GW_JOB_ID "${JOB_ID}"`

Pre-defined string to refer to the job unique identification as provided by the GridWay system. DRMAA_GW_JOB_ID will be available during job execution and can be used as an ARGUMENT for the REMOTE COMMAND. It is also useful to generate output filenames, since it is available in the main DRMAA program as returned by `drmaa_run_bulk_jobs()` and `drmaa_run_job()` function calls.

3.5.1.3 `#define DRMAA_GW_MAX_PARAM "${MAX_PARAM}"`

Pre-defined string to refer to the max value of the custom parameter in bulk jobs. This value is equal to `<start> + <total_tasks> * <increment>`.

3.5.1.4 `#define DRMAA_GW_NP "NP"`

Pre-defined string to refer to the number of process requested by a MPI job
3.5.1.5  #define DRMAA_GW_NUMBER_OF_RETRIES "NUMBER_OF_RETRIES"
Pre-defined string to refer to the NUMBER_OF_RETRIES GridWay scheduler parameter, the number of times to retry the execution on a given host. Default value is 3.

3.5.1.6  #define DRMAA_GW_PARAM "${PARAM}"
Pre-defined string to refer to a custom parameter in bulk jobs. This value is equal to <start> + <task_id> * <increment>, where <start> and <increment> are drmaa_run_bulk_job() arguments. DRMAA_PLACEHOLDER_INCR should be used for portability reasons.

3.5.1.7  #define DRMAA_GW_RANK "RANK"
Pre-defined string to refer to the RANK job template parameter. The RANK is a mathematical expression evaluated for each candidate host (those for which the REQUIREMENTS expression is true). Those candidates with higher ranks are used first to execute your jobs. Example: "(CPU_MHZ * 2) + FREE_MEM_MB;" (NOTE: Must end with ";")

3.5.1.8  #define DRMAA_GW_REQUIREMENTS "REQUIREMENTS"
Pre-defined string to refer to the REQUIREMENTS job template parameter. The REQUIREMENTS is a boolean expression evaluated for each host in the Grid, if it is true the host will be considered to submit the job. Example: "ARCH = "i686" & CPU_MHZ > 1000;" (NOTE: Must end with ";")

3.5.1.9  #define DRMAA_GW_RESCHEDULE_ON_FAILURE "RESCHEDULE_ON_FAILURE"
Pre-defined string to refer to the ON_FAILURE GridWay scheduler parameter. If set to "yes" GridWay will reschedule the job after retrying execution on a given host DRMAA_GW_NUMBER_OF_RETRIES times. Values are "yes" or "no". Default value for this attribute is "no".

3.5.1.10 #define DRMAA_GW_TASK_ID "${TASK_ID}"
Pre-defined string to refer to the task unique identification as provided by the GridWay system. DRMAA_GW_TASK_ID will be available during job execution and can be used as an ARGUMENT for the REMOTE COMMAND. It is also usefull to generate output filenames of bulk jobs. DRMAA_GW_TASK_ID ALWAYS ranges from 0 to DRMAA_GW_TOTAL_TASKS -1. Please note that this attribute name should be used ONLY in conjuntion with a drmaa_run_bulk_jobs() function call.

3.5.1.11 #define DRMAA_GW_TOTAL_TASKS "${TOTAL_TASKS}"
Pre-defined string to refer to the number of total tasks in a bulk job. DRMAA_GW_TOTAL_TASKS will be available during job execution and can be used as an ARGUMENT for the REMOTE COMMAND. This attribute name should be used ONLY in conjuntion with a drmaa_run_bulk_jobs() function call.

3.5.1.12 #define DRMAA_GW_TYPE "TYPE"
Pre-defined string to refer to the job type: "single" or "mpi". Jobs of both types can be further combined in array or workflow structures. MPI jobs spawn within a single resource and NOT across multiple resources.
3.5.1.13  
#define DRMAA_GW_TYPE_MPI "mpi"
Pre-defined string to define MPI (Message Passing Interface) jobs.

3.5.1.14  
#define DRMAA_GW_TYPE_SINGLE "single"
Pre-defined string to define single (one process) jobs.

3.5.1.15  
#define DRMAA_V_GW_INPUT_FILES "INPUT_FILES"
Pre-defined string to refer to the input files of DRMAA_REMOTE_COMMAND. DRMAA_V_GW_INPUT_FILES corresponds to a NULL terminated vector attribute value. Each vector entry is a pair of the form "source destination" filenames. If the destination filename is missing, the source filename will be preserved in the execution host. Input files (sources) ARE RELATIVE TO THE WORKING DIRECTORY or can be a GSIFTP URL. Example: input_file[0]="param."DRMAA_GW_TASK_ID" param" will copy the local file param.2 (for task 2) as param in the remote working directory.

3.5.1.16  
#define DRMAA_V_GW_OUTPUT_FILES "OUTPUT_FILES"
Pre-defined string to refer to the output files of DRMAA_REMOTE_COMMAND. DRMAA_V_GW_OUTPUT_FILES corresponds to a NULL terminated vector attribute value. Each vector entry is a "source destination" filenames pair. If the destination filename is missing, the source filename will be preserved in the client host. Output files can be a GSIFTP URL. Example: output_file[0]="binary binary."DRMAA_GW_ARCH will copy the output file "binary" to the client host with name binary.i686 (architecture of remote host is i686)

3.5.1.17  
#define DRMAA_V_GW_RESTART_FILES "RESTART_FILES"
Pre-defined string to refer to the re-start files generated by DRMAA_REMOTE_COMMAND. DRMAA_V_GW_RESTART_FILES corresponds to a NULL terminated vector attribute value. Each vector entry is the name of a checkpointing file. Re-start files can be used to preserve the execution context (at the application level) of the DRMAA_REMOTE_COMMAND on job migration or stop/resume actions.
3.6 SECTION 2 String List Helper Functions

Functions

- `int drmaa_get_next_attr_name (drmaa_attr_names_t *values, char *value, size_t value_len)`
- `int drmaa_get_next_attr_value (drmaa_attr_values_t *values, char *value, size_t value_len)`
- `int drmaa_get_next_job_id (drmaa_job_ids_t *values, char *value, size_t value_len)`
- `int drmaa_get_num_attr_names (drmaa_attr_names_t *values, size_t *size)`
- `int drmaa_get_num_attr_values (drmaa_attr_values_t *values, size_t *size)`
- `int drmaa_get_num_job_ids (drmaa_job_ids_t *values, size_t *size)`
- `void drmaa_release_attr_names (drmaa_attr_names_t *values)`
- `void drmaa_release_attr_values (drmaa_attr_values_t *values)`
- `void drmaa_release_job_ids (drmaa_job_ids_t *values)`

3.6.1 Function Documentation

3.6.1.1 `int drmaa_get_next_attr_name (drmaa_attr_names_t *values, char *value, size_t value_len)`

This function gets the next name of a `drmaa_attr_names_t` list. Each call to this function returns a different value of the list. `DRMAA_ATTR_BUFFER` can be used to instantiate `value_len`, and to define the value buffer in the form `char value[DRMAA_ATTR_BUFFER]`.

Parameters:

- `values` The names list.
- `value` The value buffer.
- `value_len` The length of value buffer, if the value is greater than `value_len`, the value string will be truncated.

Returns:

- `DRMAA_ERRNO_SUCCESS` on success
- `DRMAA_ERRNO_INVALID_ARGUMENT` if `values` is NULL
- `DRMAA_ERRNO_NO_ACTIVE_SESSION` no active session
- `DRMAA_ERRNO_NO_MORE_ELEMENTS` no more attribute names are available A Subsequent call to `drmaa_get_next_attr_value()` will return the first name again.

3.6.1.2 `int drmaa_get_next_attr_value (drmaa_attr_values_t *values, char *value, size_t value_len)`

This function gets the next value of a `drmaa_attr_values_t` list. Each call to this function returns a different value of the list. `DRMAA_ATTR_BUFFER` can be used to instantiate `value_len`, and to define the value buffer in the form `char value[DRMAA_ATTR_BUFFER]`.

Parameters:

- `values` The value list.
- `value` The value buffer.
3.6 SECTION 2 String List Helper Functions

**value_len** The length of value buffer, if the value is greater than value_len, the value string will be truncated.

Returns:

- DRMAA_ERRNO_SUCCESS on success
- DRMAA_ERRNO_INVALID_ARGUMENT if values is NULL
- DRMAA_ERRNO_NO_ACTIVE_SESSION no active session
- DRMAA_ERRNO_NO_MORE_ELEMENTS no more attribute names are available A Subsequent call to drmaa_get_next_attr_value() will return the first value again.

3.6.1.3 int drmaa_get_next_job_id (drmaa_job_ids_t *values, char *value, size_t value_len)

This function gets the next jobid of a drmaa_job_ids_t list. Each call to this function returns a different jobid of the list. DRMAA_GW_JOBID_BUFFER can be used to instantiate value_len, and to define the value buffer in the form char value[DRMAA_GW_JOBID_BUFFER].

Parameters:

- **values** The jobid list.
- **value** The value buffer, it should be of length.
- **value_len** The length of value buffer, if the value is greater than value_len, the value string will be truncated.

Returns:

- DRMAA_ERRNO_SUCCESS on success
- DRMAA_ERRNO_INVALID_ARGUMENT if values is NULL
- DRMAA_ERRNO_NO_ACTIVE_SESSION no active session
- DRMAA_ERRNO_NO_MORE_ELEMENTS no more attribute names are available A Subsequent call to drmaa_get_next_job_id() will return the first jobid again.

3.6.1.4 int drmaa_get_num_attr_names (drmaa_attr_names_t *values, size_t *size)

This function stores the number of names in drmaa_attr_names_t list in size.

Parameters:

- **values** The names list.
- **size** The number of elements in the attribute list

Returns:

- DRMAA_ERRNO_SUCCESS on success
- DRMAA_ERRNO_INVALID_ARGUMENT if values is NULL
- DRMAA_ERRNO_NO_ACTIVE_SESSION no active session
3.6.1.5 int drmaa_get_num_attr_values (drmaa_attr_values_t * values, size_t * size)

This function stores the number of values in drmaa_attr_values_t list in size.

Parameters:

- values The attributes list.
- size The number of elements in the values list

Returns:

- DRMAA_ERRNO_SUCCESS on success
- DRMAA_ERRNO_INVALID_ARGUMENT if values is NULL
- DRMAA_ERRNO_NO_ACTIVE_SESSION no active session

3.6.1.6 int drmaa_get_num_job_ids (drmaa_job_ids_t * values, size_t * size)

This function stores the number of ids in a drmaa_job_ids_t list in size.

Parameters:

- values The job ids list.
- size The number of job ids in the list

Returns:

- DRMAA_ERRNO_SUCCESS on success
- DRMAA_ERRNO_INVALID_ARGUMENT if values is NULL
- DRMAA_ERRNO_NO_ACTIVE_SESSION no active session

3.6.1.7 void drmaa_release_attr_names (drmaa_attr_names_t * values)

This function de-allocates memory of drmaa_attr_names_t list. The drmaa_attr_names_t list MUST be previously allocated by a drmaa_wait() function call.

Parameters:

- values The drmaa_attr_names_t list.

3.6.1.8 void drmaa_release_attr_values (drmaa_attr_values_t * values)

This function de-allocates memory of drmaa_attr_values_t list. The drmaa_attr_values_t list MUST be previously allocated by a drmaa_get_attribute_names() or drmaa_get_vector_attribute_names()

Parameters:

- values The drmaa_attr_values_t list.
3.6.1.9  void drmaa_release_job_ids (drmaa_job_ids_t * values)

This function de-allocates memory of drmaa_job_ids_t list. The drmaa_job_ids_t list MUST be previously allocated by a drmaa_run_bulk_jobs() function call.

Parameters:

values  The drmaa_job_ids_t list.
3.7 SECTION 3 Session Management Functions

Functions

- int drmaa_init (const char *contact, char *error_diagnosis, size_t error_diag_len)
- int drmaa_exit (char *error_diagnosis, size_t error_diag_len)

3.7.1 Function Documentation

3.7.1.1 int drmaa_exit (char * error_diagnosis, size_t error_diag_len)

Disengage from DRMAA library. This routine ends this DRMAA Session, but does not effect any jobs (i.e. queued and running jobs remain queued and running).

Parameters:

- error_diagnosis string of characters with error related diagnosis information. The error diagnosis buffer will be filled in case of error. If the size of error_diagnosis buffer passed is smaller than the error message the resultant string will be truncated.
- error_diag_len size of the error_diagnosis buffer. DRMAA_ERROR_STRING_BUFFER can be used for error диаг_len when appropriate.

Returns:

- DRMAA_ERRNO_SUCCESS on success
- DRMAA_ERRNO_NO_ACTIVE_SESSION no active session

3.7.1.2 int drmaa_init (const char * contact, char * error_diagnosis, size_t error_diag_len)

Initialize DRMAA API library and create a new DRMAA Session. drmaa_init() function MUST BE called once per DRMAA program BEFORE any DRMAA related functions are used.

Parameters:

- contact is an implementation dependent string which may be used to specify which DRM system to use. The current GridWay DRMAA implementation contact MUST be NULL or "localhost".
- error_diagnosis string of characters with error related diagnosis information. The error diagnosis buffer will be filled in case of error. If the size of error_diagnosis buffer passed is smaller than the error message the resultant string will be truncated.
- error_diag_len size of the error_diagnosis buffer. DRMAA_ERROR_STRING_BUFFER can be used for error diag_len when appropriate.

Returns:

- DRMAA_ERRNO_SUCCESS on success
- DRMAA_ERRNO_DRM_COMMUNICATION_FAILURE if the DRMAA runtime library could not contact the GridWay daemon
- DRMAA_ERRNO_INVALID_CONTACT_STRING if contact is not NULL or "localhost"
- DRMAA_ERRNO_ALREADY_ACTIVE_SESSION when drmaa_init() has been called previously.
3.8 SECTION 4 Job Template Functions

Functions

- int drmaa_allocate_job_template (drmaa_job_template_t **jt, char *error_diagnosis, size_t error_diag_len)
- int drmaa_delete_job_template (drmaa_job_template_t *jt, char *error_diagnosis, size_t error_diag_len)
- int drmaa_set_attribute (drmaa_job_template_t *jt, const char *name, const char *value, char *error_diagnosis, size_t error_diag_len)
- int drmaa_get_attribute (drmaa_job_template_t *jt, const char *name, char *value, size_t value_len, char *error_diagnosis, size_t error_diag_len)
- int drmaa_set_vector_attribute (drmaa_job_template_t *jt, const char *name, const char *value[], char *error_diagnosis, size_t error_diag_len)
- int drmaa_get_vector_attribute (drmaa_job_template_t *jt, const char *name, drmaa_attr_values_t **values, char *error_diagnosis, size_t error_diag_len)
- int drmaa_get_attribute_names (drmaa_attr_names_t **values, char *error_diagnosis, size_t error_diag_len)
- int drmaa_get_vector_attribute_names (drmaa_attr_names_t **values, char *error_diagnosis, size_t error_diag_len)

3.8.1 Function Documentation

3.8.1.1 int drmaa_allocate_job_template (drmaa_job_template_t **jt, char * error_diagnosis, size_t error_diag_len)

This function allocates a new job template. This template is used to describe the job to be submitted. This is accomplished by setting the desired scalar and vector attributes to their appropriate values.

Parameters:

- **jt** Reference to a job template pointer. The DRMAA API runtime library will allocate memory for the new template. This memory MUST be freed with a subsequent call to drmaa_delete_job_template() function
- **error_diagnosis** string of characters with error related diagnosis information. The error diagnosis buffer will be filled in case of error. If the size of error_diagnosis buffer passed is smaller than the error message the resultant string will be truncated.
- **error_diag_len** size of the error_diagnosis buffer. DRMAA_ERROR_STRING_BUFFER can be used for error_diag_len when appropriate.

Returns:

- DRMAA_ERRNO_SUCCESS on success
- DRMAA_ERRNO_NO_ACTIVE_SESSION no active session
- DRMAA_ERRNO_INTERNAL_ERROR unexpected error
- DRMAA_ERRNO_NO_MEMORY if there is not enough system memory to allocate the job template.
3.8.1.2 int drmaa_delete_job_template (drmaa_job_template_t *jt, char *error_diagnosis, size_t error_diag_len)

This function deallocates a job template.

Parameters:

jt Pointer to a job_template structure. The job template *jt MUST BE previously allocated with a drmaa_allocate_job_template() function call.

error_diagnosis string of characters with error related diagnosis information. The error diagnosis buffer will be filled in case of error. If the size of error_diagnosis buffer passed is smaller than the error message the resultant string will be truncated.

error_diag_len size of the error_diagnosis buffer. DRMAA_ERROR_STRING_BUFFER can be used for error_diag_len when appropriate.

Returns:

• DRMAA_ERRNO_SUCCESS on success
• DRMAA_ERRNO_NO_ACTIVE_SESSION no active session

3.8.1.3 int drmaa_get_attribute (drmaa_job_template_t *jt, const char *name, char *value, size_t value_len, char *error_diagnosis, size_t error_diag_len)

This function returns the value of a given attribute name. If the attribute name is not defined in the GridWay system an empty string will be returned with exit code DRMAA_ERRNO_INVALID_ARGUMENT.

Parameters:

jt Pointer to a job_template structure. The job template *jt MUST BE previously allocated with a drmaa_allocate_job_template() function call.

name Name of the job template attribute to be set.

value Value associated to the given attribute name.

value_len size of the attribute value buffer. The pre-defined

error_diagnosis string of characters with error related diagnosis information. The error diagnosis buffer will be filled in case of error. If the size of error_diagnosis buffer passed is smaller than the error message the resultant string will be truncated.

error_diag_len size of the error_diagnosis buffer. DRMAA_ERROR_STRING_BUFFER can be used for error_diag_len when appropriate.

Returns:

On success value will hold the value of the specified attribute name up to value_len characters.

• DRMAA_ERRNO_SUCCESS on success
• DRMAA_ERRNO_INVALID_ARGUMENT if name, value or jt are NULL, or value_len is 0. Also if attribute name is not defined by GridWay.
• DRMAA_ERRNO_NO_ACTIVE_SESSION no active session
3.8 SECTION 4 Job Template Functions

3.8.1.4  
int drmaa_get_attribute_names (drmaa_attr_names_t **values, char *error_diagnosis, size_t error_diag_len)

This function returns the set of supported scalar attribute names by the GridWay DRMAA implementation.

Parameters:
- values  The string vector containing the attribute names
- error_diagnosis string of characters with error related diagnosis information.
- error_diag_len size of the error_diagnosis buffer.

Returns:
- DRMAA_ERRNO_SUCCESS on success.
- DRMAA_ERRNO_NO_ACTIVE_SESSION no active session

3.8.1.5  
int drmaa_get_vector_attribute (drmaa_job_template_t *jt, const char *name, drmaa_attr_values_t **values, char *error_diagnosis, size_t error_diag_len)

This function stores in values a values string vector containing the values of the vector attribute, name’s value in the given job template. If the attribute name is not defined in the GridWay system vector with an empty string will be returned i.e. {"\0","\0"}.

Parameters:
- jt  Pointer to a job_template structure. The job template *jt MUST BE previously allocated with a drmaa_allocate_job_template() function call.
- name  Name of the job attribute whose values will be retrieved.
- values A opaque string vector containing the attribute values
- error_diagnosis string of characters with error related diagnosis information. The error diagnosis buffer will be filled in case of error. If the size of error_diagnosis buffer passed is smaller than the error message the resultant string will be truncated.
- error_diag_len size of the error_diagnosis buffer. DRMAA_ERROR_STRING_BUFFER can be used for error_diag_len when appropriate.

Returns:
- DRMAA_ERRNO_SUCCESS on success.
- DRMAA_ERRNO_INVALID_ARGUMENT if jt or name are NULL, or attribute name is not defined by GridWay.
- DRMAA_ERRNO_NO_MEMORY if there is not enough system memory to allocate the values list.
- DRMAA_ERRNO_NO_ACTIVE_SESSION no active session

3.8.1.6  
int drmaa_get_vector_attribute_names (drmaa_attr_names_t **values, char *error_diagnosis, size_t error_diag_len)

This function returns the set of supported vector attribute names by the GridWay DRMAA implementation.
Parameters:

- **values** The string vector containing the vector attribute names
- **error_diagnosis** string of characters with error related diagnosis information.
- **error_diag_len** size of the error_diagnosis buffer.

Returns:

- DRMAA_ERRNO_SUCCESS on success.
- DRMAA_ERRNO_NO_ACTIVE_SESSION no active session

### 3.8.1.7 int drmaa_set_attribute (drmaa_job_template_t *jt, const char *name, const char *value, char *error_diagnosis, size_t error_diag_len)

The function `drmaa_set_attribute()` sets a scalar attribute to a given value. Attribute names should be instantiated with the pre-defined attribute names. Several calls to `drmaa_set_attribute()` with the same attribute name will override its value.

Parameters:

- **jt** Pointer to a job_template structure. The job template *jt MUST BE previously allocated with a `drmaa_allocate_job_template()` function call.
- **name** Name of the job template attribute to be set.
- **value** Value associated to the given attribute name.
- **error_diagnosis** string of characters with error related diagnosis information. The error diagnosis buffer will be filled in case of error. If the size of error_diagnosis buffer passed is smaller than the error message the resultant string will be truncated.
- **error_diag_len** size of the error_diagnosis buffer. DRMAA_ERROR_STRING_BUFFER can be used for error_diag_len when appropriate.

Returns:

- DRMAA_ERRNO_SUCCESS on success
- DRMAA_ERRNO_INVALID_ARGUMENT if jt, name or value are NULL or the attribute name is not defined by GridWay.
- DRMAA_ERRNO_NO_MEMORY if there is not enough system memory to allocate a string to store the value in the job template.
- DRMAA_ERRNO_NO_ACTIVE_SESSION no active session

### 3.8.1.8 int drmaa_set_vector_attribute (drmaa_job_template_t *jt, const char *name, const char *value[], char *error_diagnosis, size_t error_diag_len)

This function sets an attribute to a given vector value. Vector attribute names should be instantiated with the pre-defined vector attribute names. Several calls to `drmaa_set_vector_attribute()` with the same attribute name will override its value.

Parameters:

- **jt** Pointer to a job_template structure. The job template *jt MUST BE previously allocated with a `drmaa_allocate_job_template()` function call.
**name** Name of the job template attribute to be set. The name buffer should be of length DRMAA_ATTR_BUFFER.

**value[]** A NULL terminated list of values.

**error_diagnosis** string of characters with error related diagnosis information. The error diagnosis buffer will be filled in case of error. If the size of error_diagnosis buffer passed is smaller than the error message the resultant string will be truncated.

**error_diag_len** size of the error_diagnosis buffer. DRMAA_ERROR_STRING_BUFFER can be used for error_diag_len when appropriate.

**Returns:**

- DRMAA_ERRNO_SUCCESS on success.
- DRMAA_ERRNO_INVALID_ARGUMENT if name, value or jt are NULL, or attribute is a scalar or not defined by GridWay
- DRMAA_ERRNO_NO_MEMORY if there is not enough system memory to allocate the vector values.
- DRMAA_ERRNO_NO_ACTIVE_SESSION no active session
3.9 SECTION 5 Job Submission Functions

Functions

- int drmaa_run_job (char *job_id, size_t job_id_len, drmaa_job_template_t *jt, char *error_diagnosis, size_t error_diag_len)
- int drmaa_run_bulk_jobs (drmaa_job_ids_t **jobids, drmaa_job_template_t *jt, int start, int end, int incr, char *error_diagnosis, size_t error_diag_len)

3.9.1 Function Documentation

3.9.1.1 int drmaa_run_bulk_jobs (drmaa_job_ids_t **jobids, drmaa_job_template_t *jt, int start, int end, int incr, char *error_diagnosis, size_t error_diag_len)

Submits a set of parametric jobs that can be run concurrently. For each parametric job the same template is used, and so must be properly set. Each job is identical except of it’s index:

- **DRMAA_PLACEHOLDER_INCR** ranges form start to start+(incr*TOTAL_TASKS) in increments of size "incr", where TOTAL_TASKS is ((end - start)/incr)+1
- **DRMAA_GW_TASKID** ranges form 0 to TOTAL_TASKS in increments of size "1"
- **DRMAA_GW_JOBID** the job unique identifier assigned by GridWay. These values can be used as arguments for each task and to generate input/output filenames.

GridWay will internally rescale the start-end range to 0-total_tasks. The coherence of start, end and incr values are not check by drmaa_run_job(). Their coherence SHOULD be guarantee by the calling program.

Parameters:

- **jobids** Vector containing job identifiers. Its values can be access with the drmaa_get_next_job_id() function call.
- **jt** Pointer to a job_template structure. The job template *jt MUST BE previously allocated with a drmaa_allocate_job_template() function call. Job template values MUST be previously defined with drmaa_set_attribute() drmaa_set_vector_attribute() function calls.
- **start** index associated to the first job, i.e. for this job **DRMAA_PLACEHOLDER_INCR** will be start.
- **end** index associated to the last job, i.e. for this job **DRMAA_PLACEHOLDER_INCR** will be start+(incr+TOTAL_TASKS).
- **incr** increment used to obtain the total number of job. This value could be negative
- **error_diagnosis** string of characters with error related diagnosis information. The error diagnosis buffer will be filled in case of error. If the size of error_diagnosis buffer passed is smaller than the error message the resultant string will be truncated.
- **error_diag_len** size of the error_diagnosis buffer. DRMAA_ERROR_STRING_BUFFER can be used for error_diag_len when appropriate.

Returns:

On success jobids will hold the jobs unique identifications as provided by the GridWay system, up to DRMAA_GW_JOBID_BUFFER characters.

- DRMAA_ERRNO_SUCCESS on success.
- DRMAA_ERRNO_INTERNAL_ERROR if the job template file could not be generated.
3.9 SECTION 5 Job Submission Functions

- DRMAA_ERRNO_DRM_COMMUNICATION_FAILURE could not contact GridWay daemon
- DRMAA_ERRNO_TRY_LATER If the number of jobs per session limit has been reached (1000 by default)
- DRMAA_ERRNO_NO_ACTIVE_SESSION if drmaa_init() function has not been previously called.
- DRMAA_ERRNO_NO_MEMORY if there is not enough system memory to allocate the job_ids list.

3.9.1.2 int drmaa_run_job (char *job_id, size_t job_id_len, drmaa_job_template_t *jt, char *error_diagnosis, size_t error_diag_len)

This function submits a single job with the attributes defined in the job template.

Parameters:

job_id  Job unique identification as provided by the GridWay system, up to job_id_len characters. job_id string SHOULD be of size DRMAA_GW_JOBID_BUFFER

job_id_len  size of the job_id buffer. DRMAA_GW_JOBID_BUFFER SHOULD be used for job_id_len.

jt  Pointer to a job_template structure. The job template *jt MUST BE previously allocated with a drmaa_allocate_job_template() function call. Job template values MUST be previously defined with drmaa_set_attribute() drmaa_set_vector_attribute() function calls.

error_diagnosis  string of characters with error related diagnosis information. The error diagnosis buffer will be filled in case of error. If the size of error_diagnosis buffer passed is smaller than the error message the resultant string will be truncated.

error_diag_len  size of the error_diagnosis buffer. DRMAA_ERROR_STRING_BUFFER can be used for error_diag_len when appropriate.

Returns:

On success job_id will hold the job unique identification as provided by the GridWay system, up to job_id_len characters.

- DRMAA_ERRNO_SUCCESS on success.
- DRMAA_ERRNO_INTERNAL_ERROR if the job template file could not be generated.
- DRMAA_ERRNO_DRM_COMMUNICATION_FAILURE could not contact GridWay daemon
- DRMAA_ERRNO_TRY_LATER If the number of jobs per session limit has been reached (1000 by default)
- DRMAA_ERRNO_NO_ACTIVE_SESSION if drmaa_init() function has not been previously called.
3.10 SECTION 6 Job Status and Control Functions

Functions

- int drmaa_control (const char *jobid, int action, char *error_diagnosis, size_t error_diag_len)
- int drmaa_job_ps (const char *job_id, int *remote_ps, char *error_diagnosis, size_t error_diag_len)

3.10.1 Function Documentation

3.10.1.1 int drmaa_control (const char *jobid, int action, char * error_diagnosis, size_t error_diag_len)

This function allows the job specified by jobid to be controlled according to a given action. Possible action to be performed over a given job are:

- **DRMAA_CONTROL_SUSPEND** A job will be stopped, and restart files transferred back to the client. These files if provided by the running job will be used on RESUME to re-start execution.

- **DRMAA_CONTROL_RESUME** A previously stopped job will be resumed. If re-start files are provided the job will used them to re-start execution, from the last checkpointing context.

- **DRMAA_CONTROL_TERMINATE** The job will be killed, it execution can be synchronized through the drmaa_wait and drmaa_synchronize function calls. However, job usage information will not be available.

- **DRMAA_CONTROL_HOLD** The job will be held, it execution will not start until it is released. Only jobs in the QUEUED_ACTIVE state can be held.

- **DRMAA_CONTROL_RELEASE** The job will be released and scheduled, only jobs in the USER_ON_HOLD state can be released.

Parameters:

- **jobid** String with the job unique identification as provided by the GridWay system. The jobid SHOULD be obtained from a drmaa_run_job() or drmaa_run_bulk_jobs() function calls.

- **action** The action to be performed over the job whose value may be one of the following: DRMAA_CONTROL_SUSPEND, DRMAA_CONTROL_RESUME, DRMAA_CONTROL_TERMINATE, DRMAA_CONTROL_HOLD or DRMAA_CONTROL_RELEASE.

- **error_diagnosis** string of characters with error related diagnosis information. The error diagnosis buffer will be filled in case of error. If the size of error_diagnosis buffer passed is smaller than the error message the resultant string will be truncated.

- **error_diag_len** size of the error_diagnosis buffer. DRMAA_ERROR_STRING_BUFFER can be used for error_diag_len when appropriate.

Returns:

- **DRMAA_ERRNO_SUCCESS** on success.
- **DRMAA_ERRNO_INTERNAL_ERROR** if action is not defined or supported.
- **DRMAA_ERRNO_DRM_COMMUNICATION_FAILURE** could not contact GridWay
- **DRMAA_ERRNO_NO_ACTIVE_SESSION** if drmaa_init() function has not been previously called.
3.10 SECTION 6 Job Status and Control Functions

- DRMAA_ERRNO_INVALID_ARGUMENT undefined control operation
- DRMAA_ERRNO_INVALID_JOB the job does not exist or has already been reaped
- DRMAA_ERRNO_HOLD_INCONSISTENT_STATE the HOLD action could not be performed
- DRMAA_ERRNO_RELEASE_INCONSISTENT_STATE the RELEASE action could not be performed
- DRMAA_ERRNO_RESUME_INCONSISTENT_STATE the RESUME action could not be performed
- DRMAA_ERRNO_SUSPEND_INCONSISTENT_STATE the SUSPEND action could not be performed

3.10.1.2 int drmaa_job_ps (const char *job_id, int *remote_ps, char *error_diagnosis, size_t error_diag_len)

Obtains the status of a given job.

Parameters:

*job_id* String with the job unique identification as provided by the GridWay system. The jobid SHOULD be obtained from a drmaa_run_job() or drmaa_run_bulk_jobs() function calls.

*remote_ps* The actual state of the job. remote_ps can be one of the following:

  - DRMAA_PS_UNDETERMINED: An UNDETERMINED state can either obtained due to a communication error with the GridWay daemon, or because the job has not been initialized yet.

  - DRMAA_PS_QUEUED_ACTIVE The job has been successfully submitted and it is pending to be scheduled.

  - DRMAA_PS_RUNNING The job has been successfully submitted to a remote host. Please note that once submitted, the job can be in any of the execution stages, namely: prolog (file stage-in), wrapper (execution), epilog (file stage-out) or migrating (to another host).

  - DRMAA_PS_USER_ON_HOLD The job has been held by the user

  - DRMAA_PS_DONE Job has been completely executed and output files are available at the client. drmaa_wait() and drmaa_synchronize() calls on the job will return immediately. Also rusage information is available.

  - DRMAA_PS_DONE Job has been completely executed and output files are available at the client. drmaa_wait() and drmaa_synchronize() calls on the job will return immediately. Also rusage information is available.

  - DRMAA_PS_FAILED Job execution has failed, and the "on_failure" policy is to hold it on FAILED state.

The GridWay DRMAA implementation does not define the following actions: DRMAA_PS_SYSTEM_ON_HOLD, DRMAA_PS_USER_SYSTEM_ON_HOLD, DRMAA_PS_SYSTEM_SUSPENDED and DRMAA_PS_USER_SYSTEM_SUSPENDED.
Parameters:

`error_diagnosis` string of characters with error related diagnosis information. The error diagnosis buffer will be filled in case of error. If the size of `error_diagnosis` buffer passed is smaller than the error message the resultant string will be truncated.

`error_diag_len` size of the `error_diagnosis` buffer. DRMAA_ERROR_STRING_BUFFER can be used for `error_diag_len` when appropriate.

Returns:

- DRMAA_ERRNO_SUCCESS on success.
- DRMAA_ERRNO_INTERNAL_ERROR if action is not defined or supported.
- DRMAA_ERRNO_DRM_COMMUNICATION_FAILURE could not contact GridWay (remote_ps will be DRMAA_PS_UNDETERMINED)
- DRMAA_ERRNO_INVALID_JOB the job does not exist or has already been reaped
- DRMAA_ERRNO_NO_ACTIVE_SESSION if drmaa_init() function has not been previously called.
3.11 SECTION 7 Job Synchronize and Wait Functions

Functions

- int drmaa_synchronize (const char *job_ids[], signed long timeout, int dispose, char *error_diagnosis, size_t error_diag_len)
- int drmaa_wait (const char *job_id, char *job_id_out, size_t job_id_out_len, int *stat, signed long timeout, drmaa_attr_values_t **rusage, char *error_diagnosis, size_t error_diag_len)

3.11.1 Function Documentation

3.11.1.1 int drmaa_synchronize (const char * job_ids[], signed long timeout, int dispose, char * error_diagnosis, size_t error_diag_len)

This function blocks until all jobs specified by job_ids have completed execution or fail.

Parameters:

*job_ids[] A NULL terminated list of jobid strings. The jobids SHOULD be obtained from a drmaa_run_job() or drmaa_run_bulk_jobs() function calls. The pre-defined value DRMAA_JOB_IDS.getSession_ALL can be used to synchronize all jobs submitted during the DRMAA session. Please note that "disposed" jobs will be removed from the job-list associated to the DRMAA session.

timeout specifies the time elapsed before the function call returns. DRMAA_TIMEOUT_WAIT_FOREVER can be used to wait indefinitely for a result. The value DRMAA_TIMEOUT_NO_WAIT returns immediately if no result is available. Alternatively, a number of seconds can be specified.

dispose If dispose is equal to 1 the jobid will be killed, and its resources freed in the GridWay system. Therefore subsequent calls on this job will fail. However, if dispose is equal to 0 the job remains in DRMAA_PS_DONE state in the GridWay system and its rusage statistics can be obtained with drmaa_wait() function call. Also these jobid will not make subsequent calls to drmaa_synchronize() function call to fail.

error_diagnosis string of characters with error related diagnosis information. The error diagnosis buffer will be filled in case of error. If the size of error_diagnosis buffer passed is smaller than the error message the resultant string will be truncated.

error_diag_len size of the error_diagnosis buffer. DRMAA_ERROR_STRING_BUFFER can be used for error_diag_len when appropriate.

Returns:

- DRMAA_ERRNO_SUCCESS on success.
- DRMAA_ERRNO_INVALID_ARGUMENT if timeout is not DRMAA_TIMEOUT_WAIT_FOREVER
- DRMAA_ERRNO_INVALID_JOB the job does not exist or has already been reaped
- DRMAA_ERRNO_DRM_COMMUNICATION_FAILURE could not contact GridWay
- DRMAA_ERRNO_NO_ACTIVE_SESSION if drmaa_init() function has not been previously called.
3.11.1.2 int drmaa_wait (const char *job_id, char *job_id_out, size_t job_id_out_len, int *stat, 
       signed long timeout, drmaa_attr_values_t **rusage, char *error_diagnosis, size_t error_diag_len)

This function waits for a given job to either finish executing or fail. If successfully waited, the jobs rusage information has been reaped, and further calls to drmaa_wait() with this job_id will return DRMAA-_ERRNO_INVALID_JOB.

Parameters:

*job_id* String with the job unique identification as provided by the GridWay system. The job_id SHOULD be obtained from a drmaa_run_job() or drmaa_run_bulk_jobs() function calls. DRMAA_JOB_IDS_SESSION_ANY can be used to wait on any job submitted during the DRMAA session. Please note that "disposed" jobs will be removed from the job-list associated to the DRMAA session.

job_id_out String that holds the job unique identification of the job that finished its execution, up to job_id_out_len characters. job_id_out string SHOULD be of size DRMAA_GW_JOBID_BUFFER.

job_id_out_len size of the job_id_out buffer. DRMAA_GW_JOBID_BUFFER SHOULD be used for job_id_out_len.

stat The exit status of job job_id_out. It can be interpreted with the drmaa_wifexited(), drmaa_wexitstatus(), drmaa_wifsignaled() and drmaa_wtermsig() functions.

timeout specifies the time elapsed before the function call returns. DRMAA_TIMEOUT_WAIT_FOREVER can be used to wait indefinitely for a result. The value DRMAA_TIMEOUT_NO_WAIT returns immediately if no result is available. Alternatively, a number of seconds can be specified.

**rusage** Array of values with the remote resource usage made by job job_id_out. The GridWay DRMAA implementation provides the following strings:

- "start_time=HH:MM:SS" The time the job entered the GridWay system.
- "exit_time=HH:MM:SS" The time the job completed its execution, i.e. entered DRMAA_PS_DONE or DRMAA_PS_FAILED state.
- "cpu_time=HH:MM:SS" Overall execution time on remote host.
- "xfr_time=HH:MM:SS" Overall file transfer time (stge-in + stage-out) rusage values can be accessed with the drmaa_get_netxt_attr_values() function call. rusage memory MUST be de-allocated by calling drmaa_release_attr_values().

error_diagnosis string of characters with error related diagnosis information. The error diagnosis buffer will be filled in case of error. If the size of error_diagnosis buffer passed is smaller than the error message the resultant string will be truncated.

error_diag_len size of the error_diagnosis buffer. DRMAA_ERROR_STRING_BUFFER can be used for error_diag_len when appropriate.

Returns:

- DRMAA_ERRNO_SUCCESS on success.
- DRMAA_ERRNO_INVALID_ARGUMENT if timeout is not DRMAA_TIMEOUT_WAIT_FOREVER or job_id_out is NULL
- DRMAA_ERRNO_INVALID_JOB the job does not exist or has already been reaped
- DRMAA_ERRNO_DRM_COMMUNICATION_FAILURE could not contact GridWay
- DRMAA_ERRNO_NO_RUSAGE the job has been killed and no usage is available for this job.
- DRMAA_ERRNO_NO_ACTIVE_SESSION if drmaa_init() function has not been previously called.
3.12 SECTION 8 Auxiliary Functions for Interpreting Wait Status Code

Functions

- int drmaa_wifexited (int *exited, int stat, char *error_diagnosis, size_t error_diag_len)
- int drmaa_wexitstatus (int *exit_status, int stat, char *error_diagnosis, size_t error_diag_len)
- int drmaa_wifsignaled (int *signaled, int stat, char *error_diagnosis, size_t error_diag_len)
- int drmaa_wtermsig (char *signal, size_t signal_len, int stat, char *error_diagnosis, size_t error_diag_len)
- int drmaa_wcoredump (int *core_dumped, int stat, char *error_diagnosis, size_t error_diag_len)
- int drmaa_wifaborted (int *aborted, int stat, char *error_diagnosis, size_t error_diag_len)

3.12.1 Function Documentation

3.12.1.1 int drmaa_wcoredump (int * core_dumped, int stat, char * error_diagnosis, size_t error_diag_len)

This function always returns 0 in core_dumped

Parameters:

  core_dumped Always 0
  stat The status code of a finished job
  error_diagnosis string of characters with error related diagnosis information.
  error_diag_len size of the error_diagnosis buffer.

Returns:

  • DRMAA_ERRNO_SUCCESS on success
  • DRMAA_ERRNO_NO_ACTIVE_SESSION if drmaa_init() function has not been previously called.

3.12.1.2 int drmaa_wexitstatus (int * exit_status, int stat, char * error_diagnosis, size_t error_diag_len)

This function returns into exit_status the exit code extracted from stat.

Parameters:

  exit_status The job’s exit status (equals to stat)
  stat The status code of a finished job
  error_diagnosis string of characters with error related diagnosis information.
  error_diag_len size of the error_diagnosis buffer.

Returns:

  • DRMAA_ERRNO_SUCCESS on success
  • DRMAA_ERRNO_NO_ACTIVE_SESSION if drmaa_init() function has not been previously called.
3.12.1.3  int drmaa_wifaborted (int * aborted, int stat, char * error_diagnosis, size_t error_diag_len)

This function always returns 0 in aborted.

Parameters:
- **aborted** Always 0
- **stat** The status code of a finished job
- **error_diagnosis** string of characters with error related diagnosis information.
- **error_diag_len** size of the error_diagnosis buffer.

Returns:
- DRMAA_ERRNO_SUCCESS on success
- DRMAA_ERRNO_NO_ACTIVE_SESSION if drmaa_init() function has not been previously called.

3.12.1.4  int drmaa_wifexited (int * exited, int stat, char * error_diagnosis, size_t error_diag_len)

This function returns into exited a non-zero value if stat was returned for a job that terminated normally. The job exit status can be retrieved using drmaa_wexitstatus(). The exited parameter is zero if the job terminated abnormally, drmaa_wifsignaled() can be used to gather more information. NOTE: The status code is interpreted in a bash fashion

Parameters:
- **exited** non-zero if the job has an exit status available
- **stat** The status code of a finished job obtained with the drmaa_wait() function
- **error_diagnosis** string of characters with error related diagnosis information.
- **error_diag_len** size of the error_diagnosis buffer.

Returns:
- DRMAA_ERRNO_SUCCESS on success
- DRMAA_ERRNO_NO_ACTIVE_SESSION if drmaa_init() function has not been previously called.

3.12.1.5  int drmaa_wifsignaled (int * signaled, int stat, char * error_diagnosis, size_t error_diag_len)

This function evaluates into signaled a non-zero value if stat was returned for a job that terminated due to the receipt of a signal. NOTE: The status code is interpreted in a bash fashion

Parameters:
- **signaled** non-zero if the job terminated on a signal
- **stat** The status code of a finished job
- **error_diagnosis** string of characters with error related diagnosis information.
- **error_diag_len** size of the error_diagnosis buffer.
Returns:

- DRMAA_ERRNO_SUCCESS on success
- DRMAA_ERRNO_NO_ACTIVE_SESSION if drmaa_init() function has not been previously called.

3.12.1.6 int drmaa_wtermsig (char ∗ signal, size_t signal_len, int stat, char ∗ error_diagnosis, size_t error_diag_len)

This function fills signal with up to signal_len characters of the signal name that causes the termination of the job. Only signals by POSIX are returned. For non-POSIX signals, the returned name is "UNKNOWN".

Parameters:

- **signal** The signal name
- **signal_len** The size in characters of the signal buffer
- **stat** The status code of a finished job
- **error_diagnosis** string of characters with error related diagnosis information.
- **error_diag_len** size of the error_diagnosis buffer.

Returns:

- DRMAA_ERRNO_SUCCESS on success
- DRMAA_ERRNO_NO_ACTIVE_SESSION if drmaa_init() function has not been previously called.
### 3.13 SECTION 9 Auxiliary Functions

Functions

- `const char * drmaa_strerror (int drmaa_errno)`
- `int drmaa_get_contact (char *contact, size_t contact_len, char *error_diagnosis, size_t error_diag_len)`
- `int drmaa_version (unsigned int *major, unsigned int *minor, char *error_diagnosis, size_t error_diag_len)`
- `int drmaa_get_DRM_system (char *drm_system, size_t drm_system_len, char *error_diagnosis, size_t error_diag_len)`
- `int drmaa_get_DRMAA_implementation (char *drmaa_impl, size_t drmaa_impl_len, char *error_diagnosis, size_t error_diag_len)`
- `const char * drmaa_gw_strstatus (int drmaa_state)`

#### 3.13.1 Function Documentation

##### 3.13.1.1 int drmaa_get_contact (char * contact, size_t contact_len, char * error_diagnosis, size_t error_diag_len)

If called before `drmaa_init()`, this function returns a string containing a comma-delimited list of default contact hosts for the GridWay daemon. If called after `drmaa_init()`, this function returns the contact string (hostname) where GridWay is running. The client library has been initialized by contacting this host.

Parameters:

- `contact` The contact string(s)
- `contact_len` The size in characters of the contact string buffer
- `error_diagnosis` string of characters with error related diagnosis information.
- `error_diag_len` size of the error_diagnosis buffer.

Returns:

- DRMAA_ERRNO_SUCCESS on success.

##### 3.13.1.2 int drmaa_get_DRM_system (char * drm_system, size_t drm_system_len, char * error_diagnosis, size_t error_diag_len)

This function always returns "GridWay" in `drm_system`, the only DRM system supported by the GridWay DRMAA implementation.

Parameters:

- `drm_system` Always "GridWay"
- `drm_system_len` The size in characters of the DRM system identifier buffer
- `error_diagnosis` string of characters with error related diagnosis information.
- `error_diag_len` size of the error_diagnosis buffer.

Returns:

- DRMAA_ERRNO_SUCCESS on success.
3.13 SECTION 9 Auxiliary Functions

3.13.1.3 int drmaa_get_DRMAA_implementation (char *drmaa_impl, size_t drmaa_impl_len, char *error_diagnosis, size_t error_diag_len)

This function returns the DRMAA implementation. It always returns "DRMAA for GridWay M.m" where M is the GridWay major version number and m is the minor version number.

Parameters:
- `drmaa_impl` Always "DRMAA for GridWay M.m"
- `drmaa_impl_len` The size in characters of the DRMAA implementation identifier buffer
- `error_diagnosis` string of characters with error related diagnosis information.
- `error_diag_len` size of the error_diagnosis buffer.

Returns:
- DRMAA_ERRNO_SUCCESS on success.

3.13.1.4 const char * drmaa_gw_strstatus (int drmaa_state)

This function returns a state string describing the DRMAA state of a job. WARNING: THIS FUNCTION IS NOT PART OF DRMAA STANDARD DO NOT USE IT IN YOUR DRMAA CODES.

Parameters:
- `drmaa_state` The state of a job as obtained with the drmaa_ps() for which a string description is to be returned

3.13.1.5 const char * drmaa_strerror (int drmaa_errno)

This function returns the error string describing the DRMAA error number drmaa_errno

Parameters:
- `drmaa_errno` The error code for which a string description is to be returned

3.13.1.6 int drmaa_version (unsigned int *major, unsigned int *minor, char *error_diagnosis, size_t error_diag_len)

This function sets major and minor to the major and minor versions of the DRMAA C binding specification implemented by the DRMAA implementation. Current implementation is 1.0

Parameters:
- `major` Major version number
- `minor` Minor version number
- `error_diagnosis` string of characters with error related diagnosis information.
- `error_diag_len` size of the error_diagnosis buffer.

Returns:
- DRMAA_ERRNO_SUCCESS on success.
Chapter 4

GridWay DRMAA Library File Documentation

4.1 drmaa.h File Reference

Defines

- `#define DRMAA_ATTR_BUFFER 1024`
- `#define DRMAA_CONTACT_BUFFER 1024`
- `#define DRMAA_DRM_SYSTEM_BUFFER 1024`
- `#define DRMAA_DRMAA_IMPL_BUFFER 1024`
- `#define DRMAA_ERROR_STRING_BUFFER 1024`
- `#define DRMAA_JOBNAME_BUFFER 1024`
- `#define DRMAA_SIGNAL_BUFFER 32`
- `#define DRMAA_TIMEOUT_WAIT_FOREVER -1`
- `#define DRMAA_TIMEOUT_NO_WAIT 0`
- `#define DRMAA_PS_UNDETERMINED 0x00`
- `#define DRMAA_PS_QUEUED_ACTIVE 0x10`
- `#define DRMAA_PS_SYSTEM_ON_HOLD 0x11`
- `#define DRMAA_PS_USER_ON_HOLD 0x12`
- `#define DRMAA_PS_SYSTEM_SUSPENDED 0x21`
- `#define DRMAA_PS_USER_SYSTEM_ON_HOLD 0x13`
- `#define DRMAA_PS_USER_SUSPENDED 0x22`
- `#define DRMAA_PS_USER_SYSTEM_SUSPENDED 0x23`
- `#define DRMAA_PS_RUNNING 0x20`
- `#define DRMAA_PS_FAILED 0x40`
- `#define DRMAA_CONTROL_SUSPEND 0`
- `#define DRMAA_CONTROL_RESUME 1`
- `#define DRMAA_CONTROL_HOLD 2`
- `#define DRMAA_CONTROL_RELEASE 3`
- `#define DRMAA_CONTROL_TERMINATE 4`
- `#define DRMAA_JOB_IDS_SESSION_ANY "DRMAA_JOB_IDS_SESSION_ANY"
- `#define DRMAA_JOB_IDS_SESSION_ALL "DRMAA_JOB_IDS_SESSION_ALL"
- `#define DRMAA_REMOTE_COMMAND "drmaa_remote_command"`
• #define DRMAA_V_ARGV "drmaa_v_argv"
• #define DRMAA_V_ENV "drmaa_v_env"
• #define DRMAA_INPUT_PATH "drmaa_input_path"
• #define DRMAA_OUTPUT_PATH "drmaa_output_path"
• #define DRMAA_ERROR_PATH "drmaa_error_path"
• #define DRMAA_WD "drmaa_wd"
• #define DRMAA_JOB_NAME "drmaa_job_name"
• #define DRMAA_JS_STATE "drmaa_js_state"
• #define DRMAA_SUBMISSION_STATE_ACTIVE "drmaa_active"
• #define DRMAA_SUBMISSION_STATE_HOLD "drmaa_hold"
• #define DRMAA_PLACEHOLDER_HD "$drmaa_hd_ph$"
• #define DRMAA_PLACEHOLDER_INCR "$drmaa_incr_ph$"
• #define DRMAA.Placeholder.WD "$drmaa_wd_ph$"
• #define DRMAA_DEADLINE_TIME "drmaa_deadline_time"
• #define DRMAA_BLOCK_EMAIL "drmaa_block_email"
• #define DRMAA_DEADLINE_TIME "drmaa_deadline_time"
• #define DRMAA_DURATIONS_HLIMIT "drmaa_duration_hlimit"
• #define DRMAA_DURATIONS_SLIMIT "drmaa_duration_slimit"
• #define DRMAA_JOB_CATEGORY "drmaa_job_category"
• #define DRMAA_JOIN_FILES "drmaa_join_files"
• #define DRMAA_NATIVE_SPECIFICATION "drmaa_native_specification"
• #define DRMAA_START_TIME "drmaa_start_time"
• #define DRMAA_TRANSFER_FILES "drmaa_transfer_files"
• #define DRMAA_V_EMAIL "drmaa_v_email"
• #define DRMAA_WCT_HLIMIT "drmaa_wct_hlimit"
• #define DRMAA_WCT_SLIMIT "drmaa_wct_slimit"
• #define DRMAA_ERRNO_SUCCESS 0
• #define DRMAA_ERRNO_INTERNAL_ERROR 1
• #define DRMAA_ERRNO_DRM_COMMUNICATION_FAILURE 2
• #define DRMAA_ERRNO_AUTH_FAILURE 3
• #define DRMAA_ERRNO_INVALID_ARGUMENT 4
• #define DRMAA_ERRNO_NO_ACTIVE_SESSION 5
• #define DRMAA_ERRNO_NO_MEMORY 6
• #define DRMAA_ERRNO_INVALID_CONTACT_STRING 7
• #define DRMAA_ERRNO_DEFAULT_CONTACT_STRING_ERROR 8
• #define DRMAA_ERRNO_DRMS_INIT_FAILED 9
• #define DRMAA_ERRNO_ALREADY_ACTIVE_SESSION 10
• #define DRMAA_ERRNO_DRMS_EXIT_ERROR 11
• #define DRMAA_ERRNO_INVALID_ATTRIBUTE_FORMAT 12
• #define DRMAA_ERRNO_INVALID_ATTRIBUTE_VALUE 13
• #define DRMAA_ERRNO_CONFLICTING_ATTRIBUTE_VALUES 14
• #define DRMAA_ERRNO_TRY_LATER 15
• #define DRMAA_ERRNO_DENIED_BY_DRM 16
• #define DRMAA_ERRNO_INVALID_JOB 17
• #define DRMAA_ERRNO_RESUME_INCONSISTENT_STATE 18
• #define DRMAA_ERRNO_SUSPEND_INCONSISTENT_STATE 19
• #define DRMAA_ERRNO_HOLD_INCONSISTENT_STATE 20
• #define DRMAA_ERRNO_RELEASE_INCONSISTENT_STATE 21
• #define DRMAA_ERRNO_EXIT_TIMEOUT 22
• #define DRMAA_ERRNO_NO_RUSAGE 23
#define DRMAA_ERRNO_NO_MORE_ELEMENTS 24
#define DRMAA_GW_TOTAL_TASKS "$(TOTAL_TASKS)"
#define DRMAA_GW_JOB_ID "$(JOB_ID)"
#define DRMAA_GW_TASK_ID "$(TASK_ID)"
#define DRMAA_GW_PARAM "$(PARAM)"
#define DRMAA_GW_MAX_PARAM "$(MAX_PARAM)"
#define DRMAA_GW_ARCH "$(ARCH)"
#define DRMAA_V_GW_INPUT_FILES "INPUT_FILES"
#define DRMAA_V_GW_OUTPUT_FILES "OUTPUT_FILES"
#define DRMAA_V_GW_RESTART_FILES "RESTART_FILES"
#define DRMAA_GW_RESCHEDULE_ON_FAILURE "RESCHEDULE_ON_FAILURE"
#define DRMAA_GW_NUMBER_OF_RETRIES "NUMBER_OF_RETRIES"
#define DRMAA_GW_RANK "RANK"
#define DRMAA_GW_REQUIREMENTS "REQUIREMENTS"
#define DRMAA_GW_TYPE "TYPE"
#define DRMAA_GW_TYPE_SINGLE "single"
#define DRMAA_GW_TYPE_MPI "mpi"
#define DRMAA_GW_NP "NP"

Functions

- int drmaa_get_next_attr_name (drmaa_attr_names_t *values, char *value, size_t value_len)
- int drmaa_get_next_attr_value (drmaa_attr_values_t *values, char *value, size_t value_len)
- int drmaa_get_next_job_id (drmaa_job_ids_t *values, char *value, size_t value_len)
- int drmaa_get_num_attr_names (drmaa_attr_names_t *values, size_t *size)
- int drmaa_get_num_attr_values (drmaa_attr_values_t *values, size_t *size)
- int drmaa_get_num_job_ids (drmaa_job_ids_t *values, size_t *size)
- void drmaa_release_attr_names (drmaa_attr_names_t *values)
- void drmaa_release_attr_values (drmaa_attr_values_t *values)
- void drmaa_release_job_ids (drmaa_job_ids_t *values)
- int drmaa_init (const char *contact, char *error_diagnosis, size_t error_diag_len)
- int drmaa_exit (char *error_diagnosis, size_t error_diag_len)
- int drmaa_allocate_job_template (drmaa_job_template_t **jt, char *error_diagnosis, size_t error_diag_len)
- int drmaa_delete_job_template (drmaa_job_template_t *jt, char *error_diagnosis, size_t error_diag_len)
- int drmaa_set_attribute (drmaa_job_template_t *jt, const char *name, const char *value, char *error_diagnosis, size_t error_diag_len)
- int drmaa_get_attribute (drmaa_job_template_t *jt, const char *name, char *value, size_t value_len, char *error_diagnosis, size_t error_diag_len)
- int drmaa_set_vector_attribute (drmaa_job_template_t *jt, const char *name, const char *value[], char *error_diagnosis, size_t error_diag_len)
- int drmaa_get_vector_attribute (drmaa_job_template_t *jt, const char *name, char *value[], char *error_diagnosis, size_t error_diag_len)
- int drmaa_get_attribute_names (drmaa_attr_names_t **values, char *error_diagnosis, size_t error_diag_len)
- int drmaa_get_vector_attribute_names (drmaa_attr_names_t **values, char *error_diagnosis, size_t error_diag_len)
- int drmaa_run_job (char *job_id, size_t job_id_len, drmaa_job_template_t *jt, char *error_diagnosis, size_t error_diag_len)
• int `drmaa_run_bulk_jobs` (drmaa_job_ids_t **jobids, drmaa_job_template_t *jt, int start, int end, int incr, char *error_diagnosis, size_t error_diag_len)
• int `drmaa_control` (const char *jobid, int action, char *error_diagnosis, size_t error_diag_len)
• int `drmaa_job_ps` (const char *job_id, int *remote_ps, char *error_diagnosis, size_t error_diag_len)
• int `drmaa_synchronize` (const char *job_ids[], signed long timeout, int dispose, char *error_diagnosis, size_t error_diag_len)
• int `drmaa_wait` (const char *job_id, char *job_id_out, size_t job_id_out_len, int *stat, signed long timeout, drmaa_attr_values_t **rusage, char *error_diagnosis, size_t error_diag_len)
• int `drmaa_wifexited` (int *exited, int stat, char *error_diagnosis, size_t error_diag_len)
• int `drmaa_wexitstatus` (int *exit_status, int stat, char *error_diagnosis, size_t error_diag_len)
• int `drmaa_wifsignaled` (int *signaled, int stat, char *error_diagnosis, size_t error_diag_len)
• int `drmaa_wifaborted` (int *aborted, int stat, char *error_diagnosis, size_t error_diag_len)
• int `drmaa_strerror` (int drmaa_errno)
• int `drmaa_get_contact` (char *contact, size_t contact_len, char *error_diagnosis, size_t error_diag_len)
• int `drmaa_version` (unsigned int *major, unsigned int *minor, char *error_diagnosis, size_t error_diag_len)
• int `drmaa_get_DRM_system` (char *drm_system, size_t drm_system_len, char *error_diagnosis, size_t error_diag_len)
• int `drmaa_get_DRMAA_implementation` (char *drmaa_impl, size_t drmaa_impl_len, char *error_diagnosis, size_t error_diag_len)
• const char * `drmaa_gw_strstatus` (int drmaa_state)

### 4.1.1 Detailed Description
Index

drmaa.h, 43  
drmaa_allocate_job_template  
S4, 25  
DRMAA_ATTRIB_BUFFER  
S12, 5  
DRMAA_BLOCK_EMAIL  
S14, 10  
DRMAA_CONTACT_BUFFER  
S12, 5  
drmaa_control  
S6, 32  
DRMAA_CONTROL_HOLD  
S13, 7  
DRMAA_CONTROL_RELEASE  
S13, 7  
DRMAA_CONTROL_RESUME  
S13, 7  
DRMAA_CONTROL_SUSPEND  
S13, 7  
DRMAA_CONTROL_TERMINATE  
S13, 7  
DRMAA_DEADLINE_TIME  
S14, 10  
drmaa_delete_job_template  
S4, 25  
DRMAA_DRM_SYSTEM_BUFFER  
S12, 5  
DRMAA_DRMAA_IMPL_BUFFER  
S12, 5  
DRMAA_DURATION_HLIMIT  
S14, 11  
DRMAA_DURATION_SLIMIT  
S14, 11  
DRMAA_ERRNO_ALREADY_ACTIVE_SESSION  
S15, 14  
DRMAA_ERRNO_AUTH_FAILURE  
S15, 14  
DRMAA_ERRNO_CONFLICTING_ATTRIBUTE_VALUES  
S15, 14  
DRMAA_ERRNO_DEFAULT_CONTACT_STRING_ERROR  
S15, 14  
DRMAA_ERRNO_DENIED_BY_DRM  
S15, 14  
DRMAA_ERRNO_DRM_COMMUNICATION_FAILURE  
S15, 15  
DRMAA_ERRNO_DRMS_EXIT_ERROR  
S15, 15  
DRMAA_ERRNO_DRMS_INIT_FAILED  
S15, 15  
DRMAA_ERRNO_EXIT_TIMEOUT  
S15, 15  
DRMAA_ERRNO_HOLD_INCONSISTENT_STATE  
S15, 15  
DRMAA_ERRNO_INTERNAL_ERROR  
S15, 15  
DRMAA_ERRNO_INVALID_ARGUMENT  
S15, 15  
DRMAA_ERRNO_INVALID_ATTRIBUTE_FORMAT  
S15, 15  
DRMAA_ERRNO_INVALID_ATTRIBUTE_VALUE  
S15, 15  
DRMAA_ERRNO_INVALID_CONTACT_STRING  
S15, 15  
DRMAA_ERRNO_INVALID_CONTACT  
S15, 15  
DRMAA_ERRNO_INVALID_JOB  
S15, 15  
DRMAA_ERRNO_NO_ACTIVE_SESSION  
S15, 16  
DRMAA_ERRNO_NO_MEMORY  
S15, 16  
DRMAA_ERRNO_NO_MORE_ELEMENTS  
S15, 16  
DRMAA_ERRNO_NO_RUSAGE  
S15, 16  
DRMAA_ERRNO_Release_INCONSISTENT_STATE  
S15, 16  
DRMAA_ERRNO_RESUME_INCONSISTENT_STATE  
S15, 16  
DRMAA_ERRNO_SUCCESS  
S15, 16
DRMAA_ERRNO_SUSPEND_-INCONSISTENT_STATE
  S15, 16
DRMAA_ERRNO_TRY_LATER
  S15, 16
DRMAA_ERROR_PATH
  S14, 11
DRMAA_ERROR_STRING_BUFFER
  S12, 6
drmaa_exit
  S3, 24
drmaa_get_attribute
  S4, 26
drmaa_get_attribute_names
  S4, 26
drmaa_get_contact
  S9, 40
drmaa_get_DRM_system
  S9, 40
drmaa_get_DRMAA_implementation
  S9, 40
drmaa_get_next_attr_name
  S2, 20
drmaa_get_next_attr_value
  S2, 20
drmaa_get_next_job_id
  S2, 21
drmaa_get_num_attr_names
  S2, 21
drmaa_get_num_attr_values
  S2, 21
drmaa_get_num_job_ids
  S2, 22
drmaa_get_vector_attribute
  S4, 27
drmaa_get_vector_attribute_names
  S4, 27
DRMAA_GW_ARCH
  S16, 17
DRMAA_GW_JOB_ID
  S16, 17
DRMAA_GW_MAX_PARAM
  S16, 17
DRMAA_GW_NP
  S16, 17
DRMAA_GW_NUMBER_OF_RETRIES
  S16, 17
DRMAA_GW_PARAM
  S16, 18
DRMAA_GW_RANK
  S16, 18
DRMAA_GW_REQUIREMENTS
  S16, 18
DRMAA_GW_RESCHEDULE_ON_FAILURE
  DRMAA_GW_TASK_ID
  S16, 18
DRMAA_GW_TASK_ID
  S16, 18
DRMAA_GW_TYPE
  S16, 18
DRMAA_GW_TYPE_MPI
  S16, 19
DRMAA_GW_TYPE_SINGLE
  S16, 19
drmaa_init
  S3, 24
DRMAA_INPUT_PATH
  S14, 11
DRMAA_JOB_CATEGORY
  S14, 11
DRMAA_JOB_IDS_SESSION_ALL
  S13, 8
DRMAA_JOB_IDS_SESSION_ANY
  S13, 8
DRMAA_JOB_NAME
  S14, 11
drmaa_job_ps
  S6, 33
DRMAA_JOBNAMES_BUFFER
  S12, 6
DRMAA_JOIN_FILES
  S14, 11
DRMAA Js_STATE
  S14, 11
DRMAA_NATIVE_SPECIFICATION
  S14, 12
DRMAA_OUTPUT_PATH
  S14, 12
DRMAA_PLACEHOLDER_HD
  S14, 12
DRMAA_PLACEHOLDER_INCR
  S14, 12
DRMAA_PLACEHOLDER_WD
  S14, 12
DRMAA_PS_DONE
  S13, 8
DRMAA_PS_FAILED
  S13, 8
DRMAA_PS_QUEUED_ACTIVE
  S13, 8
DRMAA_PS_RUNNING
  S13, 8
DRMAA_PS_SYSTEM_ON_HOLD
  S13, 8
DRMAA_PS_SYSTEM_SUSPENDED
<table>
<thead>
<tr>
<th>Page</th>
<th>Line</th>
<th>Symbol/Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>S13, 8</td>
<td>DRMAA_PS_UNDETERMINED</td>
<td>S13, 8</td>
</tr>
<tr>
<td>S13, 9</td>
<td>DRMAA_PS_USER_ON_HOLD</td>
<td>S13, 9</td>
</tr>
<tr>
<td>S13, 9</td>
<td>DRMAA_PS_USER_SUSPENDED</td>
<td>S13, 9</td>
</tr>
<tr>
<td>S13, 9</td>
<td>DRMAA_PS_USER_SYSTEM_ON_HOLD</td>
<td>S13, 9</td>
</tr>
<tr>
<td>S13, 9</td>
<td>DRMAA_PS_USER_SYSTEM_SUSPENDED</td>
<td>S13, 9</td>
</tr>
<tr>
<td>S2, 22</td>
<td>drmaa_release_attr_names</td>
<td></td>
</tr>
<tr>
<td>S2, 22</td>
<td>drmaa_release_attr_values</td>
<td></td>
</tr>
<tr>
<td>S2, 22</td>
<td>drmaa_release_job_ids</td>
<td></td>
</tr>
<tr>
<td>S14, 12</td>
<td>DRMAA_REMOTE_COMMAND</td>
<td></td>
</tr>
<tr>
<td>S5, 30</td>
<td>drmaa_run_bulk_jobs</td>
<td></td>
</tr>
<tr>
<td>S5, 31</td>
<td>drmaa_run_job</td>
<td></td>
</tr>
<tr>
<td>S4, 28</td>
<td>drmaa_set_attribute</td>
<td></td>
</tr>
<tr>
<td>S4, 28</td>
<td>drmaa_set_vector_attribute</td>
<td></td>
</tr>
<tr>
<td>S12, 6</td>
<td>DRMAA_SIGNAL_BUFFER</td>
<td></td>
</tr>
<tr>
<td>S14, 12</td>
<td>DRMAA_START_TIME</td>
<td></td>
</tr>
<tr>
<td>S9, 41</td>
<td>drmaa_strerror</td>
<td></td>
</tr>
<tr>
<td>S14, 12</td>
<td>DRMAA_SUBMISSION_STATE_ACTIVE</td>
<td></td>
</tr>
<tr>
<td>S14, 12</td>
<td>DRMAA_SUBMISSION_STATE_HOLD</td>
<td></td>
</tr>
<tr>
<td>S7, 35</td>
<td>drmaa_synchronize</td>
<td></td>
</tr>
<tr>
<td>S13, 9</td>
<td>DRMAA_TIMEOUT_NO_WAIT</td>
<td></td>
</tr>
<tr>
<td>S13, 9</td>
<td>DRMAA_TIMEOUT_WAIT_FOREVER</td>
<td></td>
</tr>
<tr>
<td>S14, 13</td>
<td>DRMAA_TRANSFER_FILES</td>
<td></td>
</tr>
<tr>
<td>S14, 13</td>
<td>DRMAA_V_ARGV</td>
<td></td>
</tr>
<tr>
<td>S14, 13</td>
<td>DRMAA_V_EMAIL</td>
<td></td>
</tr>
<tr>
<td>S14, 13</td>
<td>DRMAA_V_ENV</td>
<td></td>
</tr>
<tr>
<td>S14, 13</td>
<td>DRMAA_V_GW_INPUT_FILES</td>
<td></td>
</tr>
<tr>
<td>S16, 19</td>
<td>DRMAA_V_GW_RESTART_FILES</td>
<td></td>
</tr>
<tr>
<td>S16, 19</td>
<td>drmaa_version</td>
<td></td>
</tr>
<tr>
<td>S7, 35</td>
<td>drmaa_wait</td>
<td></td>
</tr>
<tr>
<td>S8, 37</td>
<td>drmaa_wcoredump</td>
<td></td>
</tr>
<tr>
<td>S14, 13</td>
<td>DRMAA_WCT_HLIMIT</td>
<td></td>
</tr>
<tr>
<td>S14, 13</td>
<td>DRMAA_WCT_SLIMIT</td>
<td></td>
</tr>
<tr>
<td>S14, 13</td>
<td>DRMAA_WD</td>
<td></td>
</tr>
<tr>
<td>S8, 37</td>
<td>drmaa_wexitstatus</td>
<td></td>
</tr>
<tr>
<td>S8, 37</td>
<td>drmaa_wifaborted</td>
<td></td>
</tr>
<tr>
<td>S8, 38</td>
<td>drmaa_wifexited</td>
<td></td>
</tr>
<tr>
<td>S8, 38</td>
<td>drmaa_wifsignaled</td>
<td></td>
</tr>
<tr>
<td>S8, 39</td>
<td>drmaa_wtermmsg</td>
<td></td>
</tr>
<tr>
<td>S16, 19</td>
<td>DRMAA_ATTR_BUFFER</td>
<td></td>
</tr>
<tr>
<td>S12</td>
<td>DRMAA_ATTR_BUFFER, 5</td>
<td></td>
</tr>
<tr>
<td>S12</td>
<td>DRMAA_CONTACT_BUFFER, 5</td>
<td></td>
</tr>
<tr>
<td>S14</td>
<td>DRMAA_DRM_SYSTEM_BUFFER, 5</td>
<td></td>
</tr>
<tr>
<td>S14</td>
<td>DRMAA_DRAA_IMPL_BUFFER, 5</td>
<td></td>
</tr>
<tr>
<td>S6</td>
<td>DRMAA_ERROR_STRING_BUFFER, 6</td>
<td></td>
</tr>
<tr>
<td>S6</td>
<td>DRMAA_JOBNAME_BUFFER, 6</td>
<td></td>
</tr>
<tr>
<td>S6</td>
<td>DRMAA_SIGNAL_BUFFER, 6</td>
<td></td>
</tr>
<tr>
<td>S13</td>
<td>DRMAA_CONTROL_HOLD, 7</td>
<td></td>
</tr>
<tr>
<td>S13</td>
<td>DRMAA_CONTROL_RELEASE, 7</td>
<td></td>
</tr>
<tr>
<td>S13</td>
<td>DRMAA_CONTROL_RESUME, 7</td>
<td></td>
</tr>
<tr>
<td>S13</td>
<td>DRMAA_CONTROL_SUSPEND, 7</td>
<td></td>
</tr>
<tr>
<td>S13</td>
<td>DRMAA_CONTROL_TERMINATE, 7</td>
<td></td>
</tr>
<tr>
<td>S8</td>
<td>DRMAA_JOB_IDS_SESSION_ALL, 8</td>
<td></td>
</tr>
<tr>
<td>S8</td>
<td>DRMAA_JOB_IDS_SESSION_ANY, 8</td>
<td></td>
</tr>
<tr>
<td>S8</td>
<td>DRMAA_PS_DONE, 8</td>
<td></td>
</tr>
<tr>
<td>S8</td>
<td>DRMAA_PS_FAILED, 8</td>
<td></td>
</tr>
<tr>
<td>S8</td>
<td>DRMAA_PS_QUEUE_ACTIVE, 8</td>
<td></td>
</tr>
<tr>
<td>S8</td>
<td>DRMAA_PS_RUNNING, 8</td>
<td></td>
</tr>
<tr>
<td>S8</td>
<td>DRMAA_PS_SYSTEM_ON_HOLD, 8</td>
<td></td>
</tr>
<tr>
<td>S8</td>
<td>DRMAA_PS_SYSTEM_SUSPENDED, 8</td>
<td></td>
</tr>
<tr>
<td>S8</td>
<td>DRMAA_PS_UNDETERMINED, 8</td>
<td></td>
</tr>
<tr>
<td>S9</td>
<td>DRMAA_PS_USER_ON_HOLD, 9</td>
<td></td>
</tr>
<tr>
<td>S9</td>
<td>DRMAA_PS_USER_SUSPENDED, 9</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>DRMAA_PS_USER_SYSTEM_ON_HOLD, 9</td>
<td></td>
</tr>
</tbody>
</table>

Generated on Wed Feb 28 10:06:55 2007 for GridWay DRMAA Library by Doxygen
DRMAA_PS_USER_SYSTEM_SUSPENDED, 9
DRMAA_TIMEOUT_NO_WAIT, 9
DRMAA_TIMEOUT_WAIT_FOREVER, 9

DRMAA_BLOCK_EMAIL, 10
DRMAA_DEADLINE_TIME, 10
DRMAA_DURATION_HLIMIT, 11
DRMAA_DURATION_SLIMIT, 11
DRMAA_ERROR_PATH, 11
DRMAA_INPUT_PATH, 11
DRMAA_JOB_CATEGORY, 11
DRMAA_JOB_NAME, 11
DRMAA_JOIN_FILES, 11
DRMAA_JS_STATE, 11
DRMAA_NATIVE_SPECIFICATION, 12
DRMAA_OUTPUT_PATH, 12
DRMAA_PLACEHOLDER_HD, 12
DRMAA_PLACEHOLDER_INCR, 12
DRMAA_PLACEHOLDER_WD, 12
DRMAA_REMOTE_COMMAND, 12
DRMAA_START_TIME, 12
DRMAA_SUBMISSION_STATE_ACTIVE, 12
DRMAA_SUBMISSION_STATE_HOLD, 12
DRMAA_TRANSFER_FILES, 13
DRMAA_VARGV, 13
DRMAA_VEMAIL, 13
DRMAA_VENV, 13
DRMAA_WCT_HLIMIT, 13
DRMAA_WCT_SLIMIT, 13
DRMAA_WD, 13

DRMAA_ERRNO_ALREADY_ACTIVE_SESSION, 14
DRMAA_ERRNO_AUTH_FAILURE, 14
DRMAA_ERRNO_CONFLICTING_ATTRIBUTE_VALUES, 14
DRMAA_ERRNO_DEFAULT_CONTACT_STRING_ERROR, 14
DRMAA_ERRNO_DENIED_BY_DRM, 14
DRMAA_ERRNO_DRM_COMMUNICATION_FAILURE, 15
DRMAA_ERRNO_DRMS_EXIT_ERROR, 15
DRMAA_ERRNO_DRMS_INIT_FAILED, 15
DRMAA_ERRNO_EXIT_TIMEOUT, 15
DRMAA_ERRNO_HOLD_INCONSISTENT_STATE, 15
DRMAA_ERRNO_INTERNAL_ERROR, 15
DRMAA_ERRNO_INVALID_ATTRIBUTE_FORMAT, 15
DRMAA_ERRNO_INVALID_ATTRIBUTE_VALUE, 15
DRMAA_ERRNO_INVALID_CONTACT_STRING, 15
DRMAA_ERRNO_INVALID_JOB, 15
DRMAA_ERRNO_NO_ACTIVE_SESSION, 16
DRMAA_ERRNO_NO_MEMORY, 16
DRMAA_ERRNO_NO_MORE_ELEMENTS, 16
DRMAA_ERRNO_NO_RUSAGE, 16
DRMAA_ERRNO_RELEASE_INCONSISTENT_STATE, 16
DRMAA_ERRNO_RESUME_INCONSISTENT_STATE, 16
DRMAA_ERRNO_SUCCESS, 16
DRMAA_ERRNO_SUSPEND_INCONSISTENT_STATE, 16
DRMAA_ERRNO_TRY_LATER, 16

DRMAA_GW_ARCH, 17
DRMAA_GW_JOB_ID, 17
DRMAA_GW_MAX_PARAM, 17
DRMAA_GW NP, 17
DRMAA_GW_NUMBER_OF_RETRIES, 17
DRMAA_GW_PARAM, 18
DRMAA_GW_RANK, 18
DRMAA_GW_REQUIREMENTS, 18
DRMAA_GW_RESCHEDULE_ON_FAILURE, 18
DRMAA_GW_TASK_ID, 18
DRMAA_GW_TOTAL_TASKS, 18
DRMAA_GW_TYPE, 18
DRMAA_GW_TYPE_MPI, 19
DRMAA_GW_TYPE_SINGLE, 19
DRMAA_VGW_INPUT_FILES, 19
DRMAA_VGW_OUTPUT_FILES, 19
DRMAA_VGW_RESTART_FILES, 19

S2
drmaa_get_next_attr_name, 20
drmaa_get_next_attr_value, 20
drmaa_get_next_job_id, 21
drmaa_get_num_attr_names, 21
drmaa_get_num_attr_values, 21
drmaa_get_num_job_ids, 22
drmaa_release_attr_names, 22
drmaa_release_attr_values, 22
drmaa_release_job_ids, 22

S3
drmaa_exit, 24
drmaa_init, 24
drmaa_allocate_job_template, 25
drmaa_delete_job_template, 25
drmaa_get_attribute, 26
drmaa_get_attribute_names, 26
drmaa_get_vector_attribute, 27
drmaa_get_vector_attribute_names, 27
drmaa_set_attribute, 28
drmaa_set_vector_attribute, 28

S5
drmaa_run_bulk_jobs, 30
drmaa_run_job, 31

S6
drmaa_control, 32
drmaa_job_ps, 33

S7
drmaa_synchronize, 35
drmaa_wait, 35

S8
drmaa_wcoredump, 37
drmaa_wexitstatus, 37
drmaa_wifaborted, 37
drmaa_wifexited, 38
drmaa_wifsignaled, 38
drmaa_wifsigterm, 39

S9
drmaa_get_contact, 40
drmaa_get_DRM_system, 40
drmaa_get_DRMAA_implementation, 40
drmaa_gw_strstatus, 41
drmaa_strerror, 41
drmaa_version, 41

SECTION 1.2 Preprocessor Directives for Handling String Output Arguments, 5
SECTION 1.3 Preprocessor Directives for Control Operations, 7
SECTION 1.4 Preprocessor Directives for Job Template Compilation, 10
SECTION 1.5 Preprocessor Directives for DRMAA Error Codes, 14
SECTION 1.6 Gridway Specific Preprocessor Directives for Job Template Compilation, 17
SECTION 2 String List Helper Functions, 20
SECTION 3 Session Management Functions, 24
SECTION 4 Job Template Functions, 25
SECTION 5 Job Submission Functions, 30
SECTION 6 Job Status and Control Functions, 32
SECTION 7 Job Synchronize and Wait Functions, 35
SECTION 8 Auxiliary Functions for Interpreting Wait Status Code, 37
SECTION 9 Auxiliary Functions, 40