Relevant information for submitting a GEANT & MOFIA jobs

2003 December - George Kwei
2004 November – Daniel Graves

Contents:

1.0 Overview

GEANT and/or MOFIA Submission script on Westgrid is located at:
/global/home/e614/bin/tbsub.pl

tbsub.pl submits jobs by passing environment variables as parameters. It loads the default environment variable parameters from the template file located in / global/home/e614/rundb-dev/exe/e614_EnvTmpFile The environment variable parameters used can be changed by passing arguments directly to tbsub.pl

Email notifications are implemented into tbsub.pl It is possible in certain circumstances for the script running on westgrid to send a notification email. The email notification system used sendmail on nunatak2.westgrid.ca thus if the network connection between the local node and nunatak2.westgrid.ca is down, email notifications are impossible. Email is currently only used when PBS aborts a job.

The following job directory structure is needed to setup jobs for submission:

/global/home/e614/tbsroot/<jobname>/common/

- <jobname> an arbitrary name for the job
- within /common, you need the following files:
  - **e614.com** --command file
  - **gb.sh** --sets up environment variables and calls 'e614.com'

/global/home/e614/tbsroot/<jobname>/queued/

- this is where all the ffcard files must go
- to create ffcards, use:
  /global/home/e614/e614soft/triumf/geant/devel/run/mkffcards.pl

How to use 'mkffcards.pl'
From 'mkffcards.pl --help'

Usage: mkffcards.pl ffcard-file firstrun nruns
Or: mkffcards.pl ffcard-file firstrun nruns CARD[INDEX] PARMIN PARMAX ...

Produces files mcrun<N>.ffcards from the given template with appropriate run numbers and random seeds.

If CARD arguments are given, also varies linearly the given parameters between their limits from the first to the last run.

EXAMPLES:

To prepare ffcards for 100 "identical" GEANT runs 6000 to 6099:

mkffcards.pl e614.ffcards 6000 100

To prepare ffcars files for 11 runs 6100 to 6110 changing gas degrader setting in 10% steps from 0 to 100%:

mkffcards.pl e614.ffcards 6100 11 GABS[2] 0. 1.

IMPORTANT: the script uses the same format for a variable parameter as the original ffcard file has, including the number of digits after the decimal point. For the example above to work correctly the original e614.ffcards should give GABS[2] with at least one digit after the point (e.g. 0.0, not just 0.). The number from the original file is discarded, but the format is copied.

Westgrid Example:

If the ffcard file (e614.ffcards) is in /global/home/e614/tbsroot/<jobname>, and . is /global/home/e614/tbsroot/<jobname>/queued then to create 1000 ffcard files starting with run number 8000, execute:

'~/e614soft/triumf/geant/devel/run/mkffcards.pl ../e614.ffcards 8000 1000'

2.0 Directory Structure

$HOME/tbsroot/genXXX/
 |-> common
 |-> queued
 |-> submitted
 |-> errorlogs
 |-> scripts
 |-> running
 |-> finished
 |-> to_be_deleted_in_future
 \-> unreachable

The common directory contains files common to gen or set. This includes environment variable output files which contains the environment variables used by the script and is generated at submission time. One environment variable file for GEANT submission unless the data is a set and one environment variable file for each MOFIA analysis. These files are important for script resubmission because the environment variables in these files will be loaded at resubmission time.

The queued directory contains all the ffcards to be submitted if a GEANT job is selected to be run. All of the ffcards are moved from queued to submitted once the GEANT jobs enter the PBS queue. The submitted directory then contains all the ffcards for GEANT jobs that have been submitted. When the script does a self check of the MC and if the check returns TRUE (meaning the MC is good), then the ffcard for that run in the submitted directory is removed.

The errorlogs directory contains all log files for jobs that failed (log file names are in the format <WG_JobNumber>.log) It also contains log files for jobs where the MC check returned false (log file names are in the format <JobName>.log).

The scripts directory contains all the scripts used to run jobs on Westgrid. These scripts are necessary for resubmission of both GEANT and MOFIA jobs.

The running directory contains a links to the local nodes where a GEANT job is running. If a link exists in this directory and no job is running on PBS (checked by using the PBS qstat -u e614) then tbsub_checkMC_and_resubmit.pl checks the job on the local node. If the MC check returns TRUE (MC is good) then the link is moved to the finished directory otherwise the job is resubmitted.

The finished directory contains a links to the local nodes where a GEANT job finished running and the MC check return TRUE. All links in the finished directory always point to a local node with MC that was checked and the result returned TRUE.

The to_be_delete_in_future directory contains links to the local nodes where a GEANT job finished running but the MC check returned FALSE. There are 2 different links that can appear here. The first link is created when a job is submitted to PBS (before it starts running – link name ends with 'queued' and this points to the script in the scripts directory). This link allows jobs that happen to vanish from PBS with being executed to be resubmitted. This link gets deleted when the job starts running. The second type of links in the to_be_deleted_in_future directory point to nodes with incomplete MC data for whatever reason. tbsub_checkMC_and_resubmit.pl will resubmit all jobs with a link in the to_be_deleted_in_future directory that are not in the PBS queue or running on a local node.

The unreachable directory only exists if any of the Westgrid nodes could not be reached (ssh time-out) by tbsub_checkMC_and_resubmit.pl when it checks MC. It contains links to the local nodes that are unreachable. When tbsub_checkMC_and_resubmit.pl is executed again, it checks to see if the unreachable directory contains links. If it does it adds those nodes to the list of nodes to be checked so that the nodes are checked again.

3.0 Environment Variable File

The template file is located $HOME/rundb-dev/exe/e614_EnvTmpFile
It contains all the default values for the environment variables that get passed to the scripts. This is a template file because genXXX and analYYY are variables that get replaced at submission time with the appropriate values.

An output environment variable file is produced and placed in $HOME/tbsroot/genXXX/common when jobs are submitted. This is for record-keeping and it is also used for ensuring a consistent state between the first submission and subsequent resubmissions. Thus is it important that these files don't overwritten or modified once they are produced. tbsub.pl checks to make sure it doesn't overwrite any of these environment variable files.

When the GEANT jobs for genXXX are submitted, one output file is produced in the common directory called genXXX-EnvFile . If MOFIA jobs are chained to the GEANT jobs or if MOFIA jobs are run on their own with the --mofia_only option, one output file per MOFIA analysis is produced in the common directory called genXXXanalYYY-EnvFile

During resubmission of GEANT jobs which is only done using tbsub_checkMC_and_resubmit.pl, the genXXX-EnvFile is read to load all the environment variables used for the first GEANT submission with tbsub.pl

During resubmission of MOFIA jobs which is only done using tbsub.pl with the --mofia_only option, the genXXXanalYYY-EnvFile is read to load all the environment variables used for the first MOFIA submission with tbsub.pl

Only a few environment variables can be overwritten by passing parameters to tbsub.pl during MOFIA resubmission: CHAIN_NUM, NO_EMAIL, and PBSPARAM. All other tbsub.pl parameters are ignored to ensure a consistent state during resubmission. When resubmitting either GEANT or MOFIA it is possible but not recommended to change the corresponding environment variable file in the common directory. Changing these files would result in an inconsistency between how GEANT or MOFIA are resubmitted and how GEANT or MOFIA were originally submitted.

4.0 Submitting a Job

Once the job directory is setup, call **/global/home/e614/bin/tbsub.pl** to submit a GEANT and/or MOFIA job. See *westgrid_scripts.sxw* for usage of tbsub.pl

- Once 'tbsub.pl' is executed, it will submit jobs to qsub, creating directories running, finished, submitted, scripts, to_be_deleted_in_future and errorlogs

- A given ffcard for a run will be moved from /queued to /submitted and copied to to the output destination on the local node upon submission of the GEANT job

- A link to the script will be created at submission time in the to_be_deleted_in_future directory with the name run<run_number>-queued This link is used by tbsub_checkMC_and_resubmit.pl to resubmit jobs that vanish from the PBS queue without a trace. The link is removed once the job starts running.

- Once the job in the queue starts running, a symbolic link of the output destination will be created in /global/home/e614/tbsroot/<jobname>/running

- Once the GEANT job is finished running, the symbolic link from /running will be moved. The location it is moved depends on the result from check MC. If check MC returns the MC is good then the link is moved to /global/home/e614/tbsroot/<jobname>/finished If check MC returns the MC is bad then then link is moved to /global/home/e614/tbsroot/<jobname>/to_be_deleted_in_future and waits to resubmitted by tbsub_checkMC_and_resumbit.pl

- If a script failed during execution or if the check MC returned that the MC is bad, the script's log file will be written to /global/home/e614/tbsroot/<jobname>/errorlogs

- tbsub.pl submits MOFIA jobs using either the links located in /global/home/e614/tbsroot/<jobname>/finished if MOFIA runs using MC or /global/scratch/twist/rawdata_symlinks/<jobname> if MOFIA runs using data

- When a MOFIA job is complete checktrees is automatically executed and either a goodlink or a badlink is created for that run.

5.0 Output

The destination of the output is stored locally on the 'ICE' machines for GEANT. To determine which machine the job ran on see the links in /global/home/e614/tbsroot/<jobname>/finished or /global/home/e614/tbsroot/<jobname>/to_be_deleted_in_future, the database (see next section), or see qsub while the job is running.

The structure on the ICE machines is: /data/twist/e614/tbsdata/<jobname>/run<run_number>/

- within /run<run_number>, you will find files similar to:
  - bat614 --> /global/home/e614/tbsroot/genXXX/common/bat614
  - e614.com -->/global/home/e614/tbsroot/genXXX/common/e614.com
  - gb.sh --> /global/home/e614/tbsroot/genXXX/common/gb.sh
  - mcrun<run_number>.ffcards
  - run<run_number>.dat
  - run<run_number>.hbook
  - tbslog.txt

For MOFIA execution the output is stored on /global/scratch/twist/systematics/mc/genXXX/analYYY/root/run<run_number> if the data is from GEANT. If the data is from a data set then the output is stored on /global/scratch/twist/systemeatics/data/genXXX/analYYY/root/run<run_number> The directory format is the same regardless of where GEANT data or a set is used.

- within /run<run_number>, you will find files similar to:
  - tree<run_number>.root
  - mofialog.dat
  - <run_number>.root
  - run<run_number>log.txt

6.0 Resubmission – What if a job dies?

If a GEANT job simply vanishes from PBS before it starts running, it is resubmitted using tbsub_checkMC_and_resubmit.pl If a MOFIA job simply vanishes from PBS before it starts running, it is resubmitted using tbsub.pl

If a job is killed while running, its symlink will remain in ~/tbsroot/<jobname>/running. Its original ffcard will have been moved to /submitted and copied to a compute node. Resubmission is done using tbsub_checkMC_and_resubmit.pl

To resubmit a killed GEANT job, execute tbsub_checkMC_and_resubmit.pl It will resubmit all crashed jobs (for whatever reason), all jobs that just vanished and all jobs where the check MC returned that the MC was bad.xit

To resubmit a job where MOFIA crashed, simply use tbsub.pl which automatically checks for jobs already running or queued, and for jobs already finished. It will resubmit all MOFIA jobs that have a badlink or have crashed.

7.0 Database

The database contains the following "new" tables:

```
Table "userjobs"
 Column |  Type   |                  Modifiers
---------+---------+----------------------------------------------
 jobid   | integer | not null default nextval('userjob_seq'::text)
 uid     | integer | not null
 jobname | text    | not null
```

- each job has a unique 'jobid'
- each job is associated with a user, 'uid'
- the job has a 'jobname'

```
Table "userruns"
  Column   |  Type   |                  Modifiers
-----------+---------+----------------------------------------------
 runid     | integer | not null default nextval('userrun_seq'::text)
 jobid     | integer | not null
 runnum    | integer | not null
 fsid      | integer | not null
 diskspace | integer | not null
 stats     | text    |
```

- each run has unique 'runid'
- each run is associated with a specific job, 'jobid'
- each run has a 'runnum'
- each run is outputted to a specific location, determined by 'fsid'
- 'diskspace', how much space is used
- 'stats', performance characteristics

```
Table "filesystems"
   Column    |  Type   |        Modifiers
-------------+---------+--------------------------
 fsid        | integer | not null
 fspath      | text    | not null
 maxuse      | bigint  |
 minfree     | bigint  | not null default 2100000
 free_target | bigint  | not null default 2700000
 can_write   | boolean | not null default 't'::bool
 can_expire  | boolean | not null default 'f'::bool
 diskgroups  | text    |
 acl         | text    |
 checked_by  | text    |
```

fshost | text |

- each filesystem has a unique file system id,'fsid'
- the filesystem has a specific path, 'fspath'
- etc...