Working on the curie machine

Table of Content

1. On-line users manual

- The command `curie.info` returns all useful information on the curie machine. Keep it in mind and use it often.
- The TGCC's storage spaces are visible from the curie machine: `$HOME` , `$CCCWORKDIR` and `$CCCSTOREDIR`
- The `$SCRATCHDIR` space only exists for the curie machine. Be careful, this space is often cleaned and only files that are less than 40 days are stored.
- You will find the users manual provided by TGCC ■here : provide your TGCC/CCRT login and password in the tab for TGCC.

2. Job manager commands

- `ccc_msub job` -> submit a job
- `ccc_mdel ID` -> kill the job with the specified ID number
- `ccc_mstat -u login` -> display all jobs submitted by login
- `ccc_mpp` -> display all jobs submitted on the machine. `ccc_mpp -n` to avoid colors.
- `ccc_mpp -u $(whoami)` ->display your jobs.

3. Before starting a job

3.1. Specify the project name

Since January 2013, you must specify in the header from which project you will use computing time:

```
#MSUB -A genxxx
```

3.2. QoS test

QoS (Quality of Service) is a test queue. You can have a maximum of 2 jobs in test queue, each of them is limited to 30min and 35 nodes (= 560 tasks). In the job header you must add:

```
#MSUB -Q test
```

and change the CPU time limit

```
#MSUB -T 1800
```

To check QoS parameters, use :

```
 ccc_mqinfo
Name     Partition  Priority  MaxCPUs  SumCPUs  MaxNodes  MaxRun  MaxSub     MaxTime
-------  ---------  --------  -------  -------  --------  ------  ------  ----------
long             *        18     2048     4096                       32  3-00:00:00
normal           *        20                                        300  1-00:00:00
test      standard        40      560      560        35               2    00:30:00
```

4. Other job manager commands

- `ccc_mpeek ID` -> display the output listing of a job. Note that the job outputs are visible while the job is running.
  `ccc_mpinfo` to find out about the classes status and about the computing requirements of the associated processors. For example (11/26/2012) :

```
/usr/bin/ccc_mpinfo
                    --------------CPUS------------  -------------NODES------------
PARTITION    STATUS   TOTAL   DOWN    USED    FREE    TOTAL   DOWN    USED    FREE     MpC  CpN SpN CpS TpC
---------    ------   ------  ------  ------  ------  ------  ------  ------  ------   ----- --- --- --- ---
standard     up        80352       0   76161    4191    5022       0    4724     298   4000  16   2   8   1
hybrid       up          264       0       0     264      33       0       0      33   2900   8   2   4   1
```

- detail of a running job. One command per line ccc_mprun :

```
ccc_mstat -H 375309
 JobID    JobName Partitio ReqCPU            Account               Start  Timelimit    Elapsed      State ExitCode
------- ---------- -------- ------ ----------------- ------------------- ---------- ---------- ---------- --------
375309 v3.histor+ standard      0   gen0826@standard 2012-05-11T16:27:53 1-00:00:00   01:49:03    RUNNING      0:0
375309+ p86maf_ru+              32   gen0826@standard 2012-05-11T16:28:16              00:14:19  COMPLETED      0:0
375309+ p86maf_ru+              32   gen0826@standard 2012-05-11T16:42:47              00:12:54  COMPLETED      0:0
375309+ p86maf_ru+              32   gen0826@standard 2012-05-11T16:55:59              00:13:30  COMPLETED      0:0
375309+ p86maf_ru+              32   gen0826@standard 2012-05-11T17:09:31              00:13:22  COMPLETED      0:0
375309+ p86maf_ru+              32   gen0826@standard 2012-05-11T17:24:06              00:13:36  COMPLETED      0:0
375309+ p86maf_ru+              32   gen0826@standard 2012-05-11T17:37:54              00:13:31  COMPLETED      0:0
375309+ p86maf_ru+              32   gen0826@standard 2012-05-11T17:51:28              00:14:19  COMPLETED      0:0
375309+ p86maf_ru+              32   gen0826@standard 2012-05-11T18:05:57              00:10:59    RUNNING      0:0
```

information about the error code of jobs: `ccc_macct nqsid`

- this job ran successfully:

```
> ccc_macct 698214
Jobid     : 698214
Jobname   : v5.historicalCMR4.452
User      : p86maf
Account   : gen2211@s+
Limits    : time = 1-00:00:00 , memory/task = Unknown
Date      : submit=06/09/2012 17:51:56, start=06/09/2012 17:51:57 , end= 07/09/2012 02:20:28
Execution : partition = standard , QoS = normal
Resources : ncpus = 53 , nnodes = 4
  Nodes=curie[2166,5964,6002,6176]

Memory /step
------
                      Resident (Mo)                          Virtual (Go)
JobID            Max (Node:Task)         AveTask       Max (Node:Task)          AveTask
-----------    ----------------------  -------    -------------------------  -------
698214             0(            :   0)       0    0.00(            :   0)    0.00
698214.batch      25(curie2166   :   0)       0    0.00(curie2166   :   0)    0.00
698214.0         952(curie2166   :   0)       0    3.00(curie2166   :   1)    0.00
...
698214.23        952(curie2166   :   0)       0    3.00(curie2166   :   2)    0.00

Accounting / step
------------------

      JobID      JobName   Ncpus Nnodes  Ntasks      Elapsed        State ExitCode
------------ ------------  ------ ------ ------- ------------   ---------- -------
698214       v5.historic+      53      4             08:28:31    COMPLETED    0:0
698214.batch        batch       1      1       1     08:28:31    COMPLETED
698214.0     p86maf_run_+      53      4      53     00:20:53    COMPLETED
698214.1     p86maf_run_+      53      4      53     00:20:20    COMPLETED
...
698214.23    p86maf_run_+      53      4      53     00:21:06    COMPLETED
```

- this job failed with an error code:

```
 > ccc_macct 680580
Jobid     : 680580
Jobname   : v5.historicalCMR4
User      : p86maf
Account   : gen2211@s+
Limits    : time = 1-00:00:00 , memory/task = Unknown
```

```
Date      : submit=30/08/2012 17:10:06, start=01/09/2012 04:11:30 , end= 01/09/2012 04:42:48
Execution : partition = standard , QoS = normal
Resources : ncpus = 53 , nnodes = 5
  Nodes=curie[2097,2107,4970,5413,5855]

Memory /step
------
                     Resident (Mo)                         Virtual (Go)
JobID            Max (Node:Task)         AveTask       Max (Node:Task)          AveTask
-----------    ----------------------  -------    -------------------------  -------
680580             0(            :   0)       0    0.00(            :   0)    0.00
680580.batch      28(curie2097   :   0)       0    0.00(curie2097   :   0)    0.00
680580.0         952(curie2097   :   0)       0    3.00(curie2097   :   1)    0.00
680580.1         316(curie2097   :   8)       0    2.00(curie2097   :   8)    0.00

Accounting / step
-----------------

      JobID      JobName   Ncpus Nnodes  Ntasks      Elapsed        State ExitCode
------------ ------------  ------ ------ ------- ------------   ---------- -------
680580       v5.historic+      53      5             00:31:18    COMPLETED    0:9
680580.batch        batch       1      1       1     00:31:18    COMPLETED
680580.0     p86maf_run_+      53      5      53     00:19:48    COMPLETED
680580.1     p86maf_run_+      53      5      53     00:10:06 CANCELLED b+
```

5. Thin nodes

Since April 2016, only thin nodes are available at TGCCC. The job header must include #MSUB -q standard to use thin nodes.

5.1. SSD on standard node : how to use it for rebuild job

SSD usage could accelerate rebuild job. It's very useful for medium and high resolution configuration like IPSLCM5A-MR. You have only to change header and RUN_DIR_PATH in rebuild.job. Take care you will run faster but cost will be multiplied by a factor of 16 because standard node ie 16 cpus are dedicated. Beware of the size of the /tmp (64GB/node) : if you have configuration with very high resolution and very high output frequency, the /tmp of standard node could be too small; in this case see below.

```
#MSUB -q standard # thin nodes
#MSUB -x  # exclusive node
RUN_DIR_PATH=/tmp/REBUILD_DIR_MR_$$
```

6. Job Header for MPI - MPI/OMP with libIGCM

Since october 2015 and libIGCM_v2.7, ins_job (libIGCM/ins_job) successfully completes job's header. Nevertheless you can check with job's header examples provided here.

6.1. Forced model

6.1.1. MPI

To launch a job on XXX MPI tasks

```
#MSUB -r MyJob
#MSUB -o Script_Output_MyJob.000001    # standard output
#MSUB -e Script_Output_MyJob.000001    # error output
#MSUB -eo
#MSUB -n XXX                           # number of MPI task
#MSUB -T 86400                         # Wall clock limit (seconds)
#MSUB -q standard                      # thin nodes
```

```
#MSUB -A gen****
BATCH_NUM_PROC_TOT=$BRIDGE_MSUB_NPROC
```

6.1.2. hybrid MPI-OMP

Hybrid version are only available with _v6 configurations

To launch a job on XXX MPI tasks and YYY threads OMP on each task

- first you need to modify your config.card

```
ATM= (gcm.e, lmdz.x, XXXMPI, YYYOMP)
```

- second you need to modify your job header

```
#MSUB -r MyJob
#MSUB -o Script_Output_MyJob.000001    # standard output
#MSUB -e Script_Output_MyJob.000001    # error output
#MSUB -eo
#MSUB -n XXX                           # number of MPI task
#MSUB -c YYY                           # number of threads OMP by task
#MSUB -T 86400                         # Wall clock limit (seconds)
#MSUB -q standard                      # thin nodes
#MSUB -A gen****
BATCH_NUM_PROC_TOT=XXX * YYY           # number of MPI task * OMP threads
```

6.2. Coupled model

6.2.1. MPI

To launch a job on XXX MPI tasks

```
#MSUB -r MyCoupledJob
#MSUB -o Script_Output_MyCoupledJob.000001    # standard output
#MSUB -e Script_Output_MyCoupledJob.000001    # error output
#MSUB -eo
#MSUB -n XXX                                  # number of MPI task
#MSUB -T 86400                                # Wall clock limit (seconds)
#MSUB -q standard                             # thin nodes
#MSUB -A gen****
BATCH_NUM_PROC_TOT=$BRIDGE_MSUB_NPROC
```

6.2.2. hybrid MPI-OMP

Hybrid version are only available with _v6 configurations

To launch a job on XXX (27) MPI tasks and YYY (4) threads OMP for LMDZ, ZZZ (19) MPI tasks for NEMO and SSS (1) XIOS servers :

- first you need to modify your config.card if required. By default, on curie, this is working for IPSLCM6_rc0 (IPSLCM6A_VLR) :

```
ATM= (gcm.e, lmdz.x, 27MPI, 4OMP)
SRF= ("" ,"" )
SBG= ("" ,"" )
OCE= (opa, opa.xx  , 19MPI)
ICE= ("" ,"" )
MBG= ("" ,"" )
CPL= ("", "" )
IOS= (xios_server.exe, xios.x, 1MPI)
```

- second you need to modify your job header

```
#MSUB -r MyCoupledJob
#MSUB -o Script_Output_MyCoupledJob.000001    # standard output
#MSUB -e Script_Output_MyCoupledJob.000001    # error output
#MSUB -eo
#MSUB -n 128                                  # Number of cores (XXX * YYY + ZZZ + SSS)
#MSUB -x                                      # exclusive node
#MSUB -E '--cpu_bind=none'
#MSUB -T 86400                                # Wall clock limit (seconds)
#MSUB -q standard                             # thin nodes
#MSUB -A gen***
```

7. Tricks

- export LANG=C to correctly display curie.info (by default for new logins)
- use [SHIFT] [CTL] C to copy part of a text displayed by curie.info
- ccc_quota gives usage of HOME, WORK, STORE and SCRATCHDIR. ccc_quota gives also detailed informations about CCCSTOREDIR usage. These informations are updated once a day at 1pm.
- use `module list`, `module unload`, `module load` to see info and use specific versions of compilers/libraries/tools. See ■FAQ for more help.

8. How to use the ddt debugger for the coupled model (or any other MPMD mode)

8.1. MPI only

- compile the model you wish to debug with the -g option (necessary in order to have access to sources from the ddt interface)
- create a debug directory which includes the model executables and the input files required by the model
- create a simplified debug job which allows you to start a run in the debug directory
- add the command "module load ddt" to your job
- add the creation of configuration run_file
- add a ddt start command in your job
- delete the environment variable SLURM_SPANK_AUKS : unset SLURM_SPANK_AUKS

```
...
module load ddt
unset SLURM_SPANK_AUKS

echo "-np 1 ${DDTPATH}/bin/ddt-client ${TMPDIR_DEBUG}/oasis" > run_file
echo "-np 26 ${DDTPATH}/bin/ddt-client ${TMPDIR_DEBUG}/lmdz.x" >> run_file
echo "-np 5 ${DDTPATH}/bin/ddt-client ${TMPDIR_DEBUG}/opa.xx" >> run_file

ddt
```

- connect yourself to curie in SSH mode with graphic export (option -X) and enter your password (if you have SSH keys on the front-end machine, move the ~/.ssh/authorized_keys* files outside of the directory, disconnect and reconnect yourself)
- start the job with graphic export : ccc_msub -X Job
- when the ddt window appears:
  - click on "Run and Debug a Program"
  - in Application select one of the 3 model executables (which one does not matter)
  - in MPI Implementation choose the "OpenMPI (Compatibility)" mode
  - in mpirun arguments put "--app ${TMPDIR_DEBUG}/run_file" with TMPDIR_DEBUG = debug directory
  - click on "Run" then on the "play" key in the upper left corner

8.2. Hybrid MPI-OpenMP (use of mpirun -rankfile method)

- compile the model you wish to debug with the -g option (necessary in order to have access to sources from the ddt interface)

- create a debug directory which includes the model executables and the input files required by the model
- create a simplified debug job which allows you to start a run in the debug directory
- add the command "module load ddt" to your job
- example with 4 OpenMP threads for lmdz and 1 OpenMP thread for nemo :

```
ddt -start -n 51 -mpiargs "-rankfile rankfile.txt --tag-output \
-np 20 -x KMP_STACKSIZE=3g -x KMP_LIBRARY=turnaround -x MKL_SERIAL=YES -x OMP_NUM_THREADS=4 ./lmdz.x : \
-np 31 -x OMP_NUM_THREADS=1 ./opa.xx
"
```

- start the job with graphic export : ccc_msub -X Job
- if not default option, select "Autoselect Bullx MPI" in Options and relaunch the job.

9. Errors on curie when running simulations

9.1. Job error: KILLED ... WITH SIGNAL 15

```
slurmd[curie1006]: error: *** STEP 639264.5 KILLED AT 2012-08-01T17:00:29 WITH SIGNAL 15 ***
```

This error message means that the time limit is exceeded. To solve the problem type clean_month, increase the time limit (or decrease PeriodNb) and restart.

9.2. Isn't there restart files for LMDZ?

Problem:

- If the coupled model does not run successfully, the whole chain of commands stops because there is no restart file for LMDZ. Read carefully the out_execution file.

Solution:

- look if a file like *error exists in the Debug subdirectory. It contains clear message errors.
- in the executable directory $SCRATCHDIR/RUN_DIR/xxxx/IPSLCM5A/xxxx look for the out_execution file. If it contains:

```
srun: First task exited 600s ago
srun: tasks 0-40,42-45: running
srun: task 41: exited abnormally
srun: Terminating job step 438782.1
slurmd[curie1150]: *** STEP 438782.1 KILLED AT 2012-06-10T18:45:41 WITH SIGNAL 9 ***
slurmd[curie1151]: *** STEP 438782.1 KILLED AT 2012-06-10T18:45:41 WITH SIGNAL 9 ***
srun: Job step aborted: Waiting up to 2 seconds for job step to finish.
slurmd[curie1150]: *** STEP 438782.1 KILLED AT 2012-06-10T18:45:41 WITH SIGNAL 9 ***
slurmd[curie1151]: *** STEP 438782.1 KILLED AT 2012-06-10T18:45:41 WITH SIGNAL 9 ***
```

don't ask questions! Type clean_month and restart the simulation.

9.3. Errors when creating or transferring files

The file system $CCCWORKDIR, $CCCSTOREDIR, $SCRATCHDIR are delicate. The error messages look like:

```
 Input/output error
Cannot send after transport endpoint shutdown
```

Don't ask question and resubmit the job.

9.4. Job error: Segmentation fault

```
/var/spool/slurmd/job637061/slurm_script: line 534:   458 Segmentation fault      /bin/ksh -x ${TEMPO_SCRIPT}
```

If you have this kind of message don't ask question and resubmit the job.

9.5. Error when submitting jobs

This message:

```
error: Batch job submission failed: Job violates accounting policy (job submit limit, user's size and/or time limits)
```

means that you have submitted too many jobs (wait for the jobs to end and resubmit), that your headers are not properly written, or that you did not specify on which genci project the computing time must be deducted. The ccc_mqinfo command returns the maximum number of jobs (to this day: 300 for 24h-max jobs, 8 for 72h-max jobs and 2 for test jobs (30 min and max 8 nodes)):

```
ccc_mqinfo
Name    Priority  MaxCPUs  MaxNodes  MaxRun  MaxSub     MaxTime
------  --------  -------  --------  ------  ------  ----------
long          18     1024                 2       8  3-00:00:00
normal        20                              300  1-00:00:00
test          40                8              2    00:30:00
```

9.6. Long waiting time before a job execution

The computation of the users priority is based on 3 cumulated criteria:

- Selected QOS (test or not)
- The fair-share value of the account (computed from the project and/or partner computation share and the previous use)
- Job's age

If your job is far down the waiting list and if you are working on different projects, use the project with the least computing time used.

This computation is not satisfying because we would prefer to encourage long simulations. We are looking for real examples of abnormal waiting situations. Please take the time to give us your feedback.

9.7. Disk quota exceeded

Be careful to quotas on /scratch! Monitor them with the command ccc_quota. Destroy the temporary directories created by jobs that ended too early and that did not clear the $SCRATCHDIR/TMPDIR_IGCM and $SCRATCHDIR/RUN_DIR directories. You should have a 20 To quota on curie.

```
> ccc_quota
Disk quotas for user xxxx:

            ------------------ VOLUME --------------------  ------------------- INODE --------------------
Filesystem       usage        soft        hard       grace       files        soft        hard       grace
----------       -----        ----        ----       -----       -----        ----        ----       -----
   scratch       3.53T         20T         20T           -      42.61k          2M          2M           -
     store           -           -           -           -      93.76k        100k        101k           -
      work     232.53G          1T        1.1T           -      844.8k        1.5M        1.5M           -
```

9.8. A daemon (pid unknown) died unexpectedly with status 1 while attempting to launch so we are aborting.

This message appears when time limit is reached. Increase requested time in job's header or reduce NbPeriod in your job to reduce the number of loop's iteration.

10. REDO

Simulations with the IPSLCM5/IPSLCM6 coupled model are reproducible if you use the same Bands file for LMDZ. See trusting TGCC/curie on this web page: ■http://webservices.ipsl.jussieu.fr/trusting/