Wikiprint Book

Title: 1. FAQ : Setting up and performing a simulation

Subject: lgcmg_doc - Doc/FAQ

Version: 99

Date: 05/21/24 00:40:19

Table of Content

Frequently Asked Questions **3**
1. FAQ : Setting up and performing a simulation **3**
1.1. How do I overwrite an existing simulation? 3
1.2. How do I continue or restart a simulation? 3
1.3. How do I setup a new experiment? 3
1.4. How can I start from another simulation? 3
1.5. How do I create the LMDZ histins.nc file? 3
2. FAQ : Running the model **3**
2.1. How do I read the Script_Output file? 3
2.2. The LMDZ parallelism and the Bands files 4
2.3. How do I define the number of MPI jobs and the number of OpenMP threads? 4
2.4. Why does the run.card file contain the keyword Fatal? 4
2.5. How do I use a different version of libIGCM? 4
2.6. How do I restart a simulation to recover missing output files? 5
2.7. How can I change the atmosphere horizontal resolutions using the same LMDZOR libIGCM configuration ? 6
3. FAQ : Special configurations **7**
3.1. How do I create the initial conditions for LMDZOR? 7
3.2. How do I deactivate STOMATE in IPSLCM5 or in LMDZOR? 7
3.3. How do I perform a nudged run? 7
3.4. How do I run simulations with specific versions of compiler and/or libraries on CURIE at the TGCC ? (modules) 7
3.5. How to have min and max value exchanged through OASIS? 9
4. FAQ : Post processing **10**
4.1. Where are post processing jobs run? 10
4.2. How do I check that the post processing jobs were successful? 10
4.3. How do I read/retrieve/use files on esgf/thredds? 10
4.4. How do I add a variable to the Time Series? 11
4.5. How do I superimpose monitoring plots (intermonitoring)? 11
4.6. What is the Monitoring? 11
4.7. How do I add a plot to the monitoring? 11
4.8. How do I calculate seasonal means over 100 years? 11
5. FAQ : Unix tricks **13**
5.1. How to delete a group of files using the find command? 13
5.2. Allowing read-access to everybody 13
6. FAQ : Miscellaneous **13**
6.1. How do I use TimeSeries_Checker.job to create files on $STORE when the output files are on DMNFS? 13
6.1.1. Example for the rcp45 simulation 13
6.2. How do I restart one simulation month which ran on $DMFDIR when the outputs are stored on $STORE? 14
6.3. How do I copy a model installation directory instead of downloading from the forge (or move a directory)? 15

Frequently Asked Questions

Frequently (and not so frequently) Asked Questions

1. FAQ : Setting up and performing a simulation

1.1. How do I overwrite an existing simulation?

If you want to relaunch a simulation from the beginning you need to delete everything created previously. All the output files must be deleted because they cannot be overwritten. Do the following:

1. Delete the `run.card` file in your experiment directory.
2. Delete all output directories:

- at TGCC
  - `$CCCSTOREDIR/IGCM_OUT/TagName/(...)/JobName`
  - `$CCCWORKDIR/IGCM_OUT/TagName/(...)/JobName`
  - `$SCRATCHDIR/IGCM_OUT/TagName/(...)/JobName`
- at IDRIS
  - ergon:`$HOME/IGCM_OUT/TagName/(...)/JobName`
  - ada : `$WORKDIR/IGCM_OUT/TagName/(...)/JobName`

1. Launch the job.

TIP: If you have already done a simulation before you could find all output paths in the Script_output* file. Delete it before starting a new simulation.

1.2. How do I continue or restart a simulation?

See here.

1.3. How do I setup a new experiment?

See here.

1.4. How can I start from another simulation?

See here.

1.5. How do I create the LMDZ histins.nc file?

You have several options. The easiest one is to change the output frequency of one of the existing files. For instance, you can change the output frequency of the histhf.nc file to instantaneous without changing the file name. To do so, keep HF in WriteFrequency in config.card:

```
[ATM]
#
WriteFrequency="1M 1D HF"
```

Change the 3rd column of the phys_out_filetimesteps parameter in PARAM/output.def_OutLevel. OutLevel is chosen in lmdz.card and by default OutLevel=low. Specify 1800.s in PARAM/output.def_low if you want the output to be saved every 30 minutes:

```
phys_out_filetimesteps = 1.mth,  1.day,  1800.s,  0.125day, 0.125day, 1800.s
```

You can also change phys_out_filelevels in the 3rd column.

2. FAQ : Running the model

2.1. How do I read the Script_Output file?

During each job execution, a corresponding `Script_Output` file is created.
Important : If your simulation stops you can look for the keyword "IGCM_debug_CallStack" in this file. This word will be preceded by a line giving more details on the problem that occurred.

See can be found here for more details.

2.2. The LMDZ parallelism and the Bands files

See here.

2.3. How do I define the number of MPI jobs and the number of OpenMP threads?

If you run your model in MPI mode only (without OpenMP) the number of MPI processes is defined in config.card by the JobNumProcTot parameter:

```
#-- Total Number of Processors
JobNumProcTot=32
```

If you run your model in hybrid mode (MPI-OpenMP), the number of MPI processes and the number of OpenMP threads are set in config.card in the section "Executable". You still need to precise the CPU total number in `JobNumProcTot` parameter by summing all components processes.

For instance, for LMDZ : 16 MPI processes and 2 OpenMP threads (ATM compenent needs `2OMP x 16MPI = 32CPU`).

```
ATM= (gcm.e, lmdz.x, 16MPI, 2OMP)
```

Notice that the job header differs from the one for openMP.

2.4. Why does the `run.card` file contain the keyword `Fatal`?

The keyword `Fatal` indicates that something went wrong in your simulation. Below is a list of the most common reasons:

- a problem was encountered while copying the input files
- the frequency settings in config.card are erroneous
- run.card has not been deleted before resubmitting a simulation, or "OnQueue" has not been specified in run.card when continuing a simulation
- a problem was encountered during the run
- the disk quotas have been reached
- a problem was encountered while copying the output files
- a post processing job encountered a problem

  `pack_xxx` has failed and caused the simulation to abort. In this case, you must find `STOP HERE INCLUDING THE COMPUTING JOB` located in the appropriate output pack file.
  - `rebuild` was not completed successfully

See the corresponding chapter about monitoring and debug for further information.

2.5. How do I use a different version of libIGCM?

libIGCM is constantly being updated. We recommend to choose the latest tag of libIGCM. Here is what to do:

- save the old libIGCM version (just in case)
- get libIGCM
- reinstall the post processing jobs
- make sure that there has been no major change in AA_job, otherwise reinstall the main job

```
cd modipsl
mv libIGCM libIGCM_old
svn checkout http://forge.ipsl.jussieu.fr/libigcm/svn/tags/libIGCM_v2.8.1 libIGCM
diff libIGCM/AA_job libIGCM_old/AA_job
libIGCM/ins_job #
```

In case you need version XXX of the trunk of libIGCM, change the "svn checkout" line into:

```
svn checkout -r XXX http://forge.ipsl.jussieu.fr/libigcm/svn/trunk/libIGCM libIGCM
```

If AA_job has been modified, you must :

- move to the experiment directory,
- delete or move old jobs

  rerun the new jobs using `ins_job`. MYCONFIG could be IPSLCM5_v5 or ORCHIDEE_OL, for example:

```
cd ...../config/MYCONFIG/MYEXP
mv Job_MYEXP OLDJOB                                          # save the old job
../../../libIGCM/ins_job
# modifier Job_MYEXP : NbPeriod, memory,... as it was done in OLDJOB
```

2.6. How do I restart a simulation to recover missing output files?

UPDATE - ON CURIE (4/18/2016 ACo) / TO BE DONE ON ADA (idea : change $STOREDIR by ergon or ada workdir)

This method shows how to rerun a complete simulation period in a different directory (REDO instead of DEVT/PROD).

Example : To rerun v3.historicalAnt1 to recompute a whole year (e.g. 1964) you must :

- On the file server (CCCSTOREDIR), create the necessary RESTART file.

```
## Directory
mkdir $CCCSTOREDIR/....IGCM_OUT/IPSLCM5A/REDO/historicalAnt/v3.historicalAnt1
cd $CCCSTOREDIR/....IGCM_OUT/IPSLCM5A/REDO/historicalAnt/v3.historicalAnt1
# RESTART
mkdir -p RESTART ; cd RESTART
ln -s ../../../../PROD/historicalAnt/v3.historicalAnt1/RESTART/v3.historicalAnt1_19640831_restart.nc v3.historicalAnt1RE
```

- There is nothing to do for the Bands file, it's save at previous period in PARAM directory, and the simulation know where to find it.
- If you are running a coupled model : On the scratch disk ($SCRATCHDIR/IGCM_OUT), create the mesh_mask file (TO BE CONFIRMED)

```
mkdir $SCRATCHDIR/....IGCM_OUT/IPSLCM5A/REDO/historicalAnt/v3.historicalAnt1REDO
cd $SCRATCHDIR/....IGCM_OUT/IPSLCM5A/REDO/historicalAnt/v3.historicalAnt1REDO
# mesh_mask
mkdir -p OCE/Output
cd OCE/Output
ln -s ../../../../../PROD/historicalAnt/v3.historicalAnt1/OCE/Output/v3.historicalAnt1_mesh_mask.nc v3.historicalAnt1RED
cd ../..
```

- On the computing machine:
  - create a new directory

```
cp -pr v3.historicalAnt1 v3.historicalAnt1REDO
```

  - in this new directory, change the run.card file and set the following parameters to:

```
OldPrefix= v3.historicalAnt1_19631231
PeriodDateBegin= 1964-01-01
PeriodDateEnd= 1964-01-31
CumulPeriod= xxx # Specify the proper "cad" value, i.e. the same month in the run.card cookie (ARGENT)
PeriodState= OnQueue
```

  - change the config.card file to one pack period (1 year), do not do any post processing, start rebuild month by month and specify PackFrequency.

```
JobName=v3.historicalAnt1
...
SpaceName=REDO
...
DateEnd= 1964-12-31
...
RebuildFrequency=1M
PackFrequency=1Y
...
TimeSeriesFrequency=NONE
...
SeasonalFrequency=NONE
```

- you don't need to change the name of the simulation
- restart the simulation :

```
vi run.card # check one more time
vi Job_v3.historicalAnt1 # check the time parameters and names of the output scripts
ccc_msub  Job_v3.historicalAnt1
```

- once the job is finished, if you are running a coupled mode : check that the solver.stat files are identical. The solver.stat files are stored in DEBUG :

```
sdiff OCE/Debug/v3.historicalAnt1REDO_19640901_19640930_solver.stat /dmnfs11/cont003/p86maf/IGCM_OUT/IPSLCM5A/PROD/histo
```

2.7. How can I change the atmosphere horizontal resolutions using the same LMDZOR libIGCM configuration ?

To do this you have to make some changes in your files.

- in the modipsl/config/LMDZOR directory, modify your Makefile to add the resolutions you need. Here is an example for 48x48x79 resolution:

```
LMD4848-L79 : libioipsl liborchidee lmdz48x48x79 verif
       echo "noORCAxLMD4848" >.resol_48x48x79
       echo "RESOL_ATM_3D=48x48x79" >>.resol_48x48x79

lmdz48x48x79:
       $(M_K) lmdz RESOL_LMDZ=48x48x79
```

- Also add the resolution "$(RESOL_LMDZ)" in the name of executables :

```
(cd ../../modeles/LMDZ; ./makelmdz_fcm -cpp ORCHIDEE_NOOPENMP -d $(RESOL_LMDZ) -cosp true -v true -parallel mpi -arch $(FC
(cd ../../modeles/LMDZ; ./makelmdz_fcm -cpp ORCHIDEE_NOOPENMP -d $(RESOL_LMDZ) -cosp true -v true -mem -parallel mpi -arch
```

- in modipsl/libIGCM/AA_job replace .resol by .resol_myresolution like this :

```
[ -f ${SUBMIT_DIR}/../.resol ] && RESOL=$(head -1 ${SUBMIT_DIR}/../.resol)

become

[ -f ${SUBMIT_DIR}/../.resol_myresolution ] && RESOL=$(head -1 ${SUBMIT_DIR}/../.resol_myresolution)
```

- modify the modipsl/config/LMDZOR/GENERAL/DRIVER/lmdz.driver by replacing

```
[ -f ${SUBMIT_DIR}/../.resol ] && eval $(grep RESOL_ATM_3D ${SUBMIT_DIR}/../.resol) || RESOL_ATM_3D=96x95x19

by

[ -f ${SUBMIT_DIR}/../.resol_myresolution ] && eval $(grep RESOL_ATM_3D ${SUBMIT_DIR}/../.resol_myresolution) || RESOL_ATM
```

Now you can create as many experiment as you have compiled your model.

```
cd modipsl/config/LMDZOR/
cp EXPERIMENT/LMDZOR/clim/config.card .
etc...
```

3. FAQ : Special configurations

3.1. How do I create the initial conditions for LMDZOR?

For a few configurations such as LMDZOR and LMDZREPR, you must create initial and boundary conditions in advance. This is not necessary for coupled configurations such as IPSLCM5_v5.

For more information, see this chapter.

3.2. How do I deactivate STOMATE in IPSLCM5 or in LMDZOR?

The IPSLCM5 model has not been evaluated for these cases.

Here is how to do it.

3.3. How do I perform a nudged run?

Atmospherical nudging
This paragraph describes how to perform a nudged run for configurations that include LMDZ. To do so, you have to:

- activate option `ok_guide` in the `lmdz.card` file (this option enables you to activate the corresponding flag_ in `PARAM/guide.def`)
- check that the wind fields specified are contained in `BoundaryFiles`. (Several forcing are available on curie)

For example:

```
[BoundaryFiles]
List= ....\
     (/ccc/work/cont003/subipsl/subipsl/ECMWF96x95/AN${year}/u_ecmwf_${year}${month}.nc, u.nc)\
     (/ccc/work/cont003/subipsl/subipsl/ECMWF96x95/AN${year}/v_ecmwf_${year}${month}.nc, v.nc)\
```

- choose the proper dates in `config.card` (pay attention to leap years)

Oceanic nudging
To force the oceanic model in salinity or SST you could find the procedure in ■NEMO official documentation (section 7.12.3: *Surface restoring to observed SST and/or SSS*)

Notice that NEMO uses the salinity nudging, by default, when it's used in oceanic forced configurations.

3.4. How do I run simulations with specific versions of compiler and/or libraries on CURIE at the TGCC ? (modules)

For various reasons you may want to run simulations with different versions of compiler or libraries (mainly netCDF).

The first thing is to keep a dedicated installation of modipsl for this specific setup since you will have to modify the libIGCM associated with the simulations.

Keep in mind that you need the modules of the libraries you want to use to be properly loaded at both:

- compile time
- run time

Compile time

You can create a script shell that unloads the modules of the default configuration and loads the modules you want to use. Here is an example of the file `modules.sh` to use intel/12 and netCDF3.6.3: (the order in which you unload and load the modules is important)

```
#!/bin/bash

#set -vx

# unload modules
module unload nco    #/4.1.0
module unload netcdf #/4.2_hdf5_parallel
module unload hdf5   #/1.8.9_parallel
module unload intel

# load modules
module load intel/12.1.9.293
module load netcdf/3.6.3
module load hdf5/1.8.8
module load nco/4.1.0
```

You have to make sure the modules you want to be used by your code are loaded before each compilation of your configuration. Use module list to view the currently loaded modules. If necessary source `module.sh` before compiling.

Runtime

The proper modules have to be loaded for the dynamic linking to your libraries to succeed.

You can source `modules.sh` before submitting (ccc_msub), however this is not very convenient.

A better way is to modify `libIGCM_sys_curie.ksh` in your libIGCM installation (`(...)/modipsl/libIGCM/libIGCM_sys/` directory).

Locate the part where the environment tools are set in this file and add module unload and load commands:

```
#=====================================================
# Set environment tools (ferret, nco, cdo)
#=====================================================
if [ X${TaskType} = Xcomputing ] ; then
 . /ccc/cont003/home/dsm/p86ipsl/.atlas_env_netcdf4_curie_ksh > /dev/null 2>&1
# to run with netcdf 3.6.3 ie compilation done before 17/2/2014
# uncomment 2 lines :
#  module unload netcdf
#  module load netcdf/3.6.3
 # set the proper modules
 module unload nco
 module unload netcdf
 module unload hdf5
 module unload intel
 module load intel/12.1.9.293
 module load netcdf/3.6.3_p1
 module load hdf5/1.8.8
 module load nco/4.1.0
 #set the proper modules end
 export PATH=${PATH}:/ccc/cont003/home/dsm/p86ipsl/AddNoise/src_X64_CURIE/bin
 export PATH=${PATH}:/ccc/cont003/home/dsm/p86ipsl/AddPerturbation/src_X64_CURIE/bin
else
 . /ccc/cont003/home/dsm/p86ipsl/.atlas_env_netcdf4_curie_ksh > /dev/null 2>&1
 PCMDI_MP=/ccc/work/cont003/dsm/p86ipsl/PCMDI-MP
fi
```

This way you can launch experiments on CURIE without having to source your `module.sh` file.

Keep in mind that the code has to be compiled with the same modules that the ones that are loaded by libIGCM at runtime.

In case of module mismatch you will have a runtime error stating a library was not found.

3.5. How to have min and max value exchanged through OASIS?

To add min max sum values of one field exchanged through OASIS, one has to add verbose mode, to add 2 operations and to describe them. Then you will find information in output text file.

Example :

- Modification in namcouple :
  - Before :

```
 $NLOGPRT
  0
$END
O_SSTSST  SISUTESW 1 5400  2  sstoc.nc  EXPORTED
362 332 144 143 torc  tlmd  LAG=2700
P  2 P  0
LOCTRANS MAPPING
# LOCTRANS CHECKIN MAPPING CHECKOUT
# LOCTRANS: AVERAGE to average value over coupling period
AVERAGE
# CHECKIN: calculates the global minimum, the maximum and the sum of the field
# INT=1
# Mozaic: 1) mapping filename 2) connected unit 3) dataset rank 4) Maximum
#         number of overlapped neighbors
rmp_torc_to_tlmd_MOSAIC.nc src
# CHECKOUT: calculates the global minimum, the maximum and the sum of the field
# INT=1
#
```

  - After :

```
 $NLOGPRT
  1
$END
O_SSTSST  SISUTESW 1 5400  2  sstoc.nc  EXPORTED
362 332 144 143 torc  tlmd  LAG=2700
P  2 P  0
# LOCTRANS MAPPING
LOCTRANS CHECKIN MAPPING CHECKOUT
# LOCTRANS: AVERAGE to average value over coupling period
AVERAGE
# CHECKIN: calculates the global minimum, the maximum and the sum of the field
INT=1
# Mozaic: 1) mapping filename 2) connected unit 3) dataset rank 4) Maximum
#         number of overlapped neighbors
rmp_torc_to_tlmd_MOSAIC.nc src
# CHECKOUT: calculates the global minimum, the maximum and the sum of the field
INT=1
#
```

- Informations :
  - min, max and sum for received field in component 1 : atmosphere

```
> egrep 'oasis_advance_run  at .*RECV|diags:' debug.root.01|more
oasis_advance_run  at            0           0  RECV: SISUTESW
  diags:         SISUTESW    0.00000000000        304.540452041        3548934.08936
oasis_advance_run  at            0           0  RECV: SIICECOV
oasis_advance_run  at            0           0  RECV: SIICEALW
```

```
oasis_advance_run  at            0           0  RECV: SIICTEMW
oasis_advance_run  at            0           0  RECV: CURRENTX
oasis_advance_run  at            0           0  RECV: CURRENTY
oasis_advance_run  at            0           0  RECV: CURRENTZ
oasis_advance_run  at         5400        5400  RECV: SISUTESW
  diags:         SISUTESW    0.00000000000        304.569482446        3549053.65992
...
```

- min, max and sum for sent field from component 2 : ocean

```
>  egrep 'oasis_advance_run  at.*SEND|diags:' debug.root.02|more
oasis_advance_run  at        -2700           0  SEND: O_SSTSST
  diags:         O_SSTSST   0.271306415433        304.835436600        31678793.3366
oasis_advance_run  at        -2700           0  SEND: OIceFrc
oasis_advance_run  at        -2700           0  SEND: O_TepIce
oasis_advance_run  at        -2700           0  SEND: O_AlbIce
oasis_advance_run  at        -2700           0  SEND: O_OCurx1
oasis_advance_run  at        -2700           0  SEND: O_OCury1
oasis_advance_run  at        -2700           0  SEND: O_OCurz1
oasis_advance_run  at         2700        5400  SEND: O_SSTSST
  diags:         O_SSTSST   0.271306391122        304.852847163        31680753.5627
...
```

4. FAQ : Post processing

4.1. Where are post processing jobs run?

libIGCM allows you to perform post processing jobs on the same machine as the main job. You can also start post processing jobs on other machines dedicated particularly to post processing. It is not done anymore.

Currently used machines:

Center	Computing machine	Post processing
CCRT	Titane	Titane, queue mono
TGCC	Curie	Curie, thin node, -q standard
IDRIS	Ada	Ada (ongoing)

4.2. How do I check that the post processing jobs were successful?

See here.

4.3. How do I read/retrieve/use files on esgf/thredds?

- At IDRIS, visit the following website:
  - ■http://prodn.idris.fr/thredds and select ipsl_public, your login, your configuration, your simulation and the ATM component (then the `Analyse` subdirectory) as well as `ATLAS` or `MONITORING`.
- At TGCC, visit the following website:

  ■https://esgf.extra.cea.fr/thredds and select:
  - work, your login, your configuration, your simulation, etc. for ATLAS and MONITORING
  - store, your login, your configuration, your simulation, etc. and ATM or other component to access Analyse files (TS or SE)

  Once you found a netcdf file (suffix `.nc`), you can download it by clicking on it or you can analyze it with openDAP functions. To do so add `thredds/dodsC` to the address right after the server address. For example:

```
 ciclad : ferret ...
> use "https://esgf.extra.cea.fr/thredds/dodsC/store/yourlogin/.../file.nc"
> use "https://prodn.idris.fr/thredds/dodsC/ipsl_public/yourlogin/.../file.nc"
```

More information on Monitoring can be found here: DocFsimu

4.4. How do I add a variable to the Time Series?

See this section.

4.5. How do I superimpose monitoring plots (intermonitoring)?

■Audio
Short link :

- for esgf type : `http://webservices.ipsl.fr/monitoring`

To select simulations from two centers or for two different logins, you must go back to step 1 and click on **append directories** to add new simulations.

4.6. What is the Monitoring?

See chapter **Simulation and post-processing**, section *Postprocessing with libIGCM* here

4.7. How do I add a plot to the monitoring?

The answer to this question is here.

4.8. How do I calculate seasonal means over 100 years?

In order to compute a seasonal mean over 100 years, check that all decades are on the file server (`SE_checker`). Then run the job `create_multi_se` on the post processing machine.

Note that an atlas for these 100 years will also be created. See the example for the 100-year atlas for piControlMR1 here : ■SE 2000 2099

1. If not done yet, create a specific post processing directory. See the chapter on how to run or restart post processing jobs for details.
2. Copy `create_se.job`, `SE_checker.job` and `create_multi_se.job`
   Check/change the following variables in `create_se.job`:

```
libIGCM=${libIGCM:=.../POST_CMIP5/libIGCM_v1_10/modipsl/libIGCM}
```

4. Check that all decades exist.
5. Check/change the variables in SE_checker.job:

```
libIGCM=${libIGCM:=.../POST_CMIP5/libIGCM_v1_10/modipsl/libIGCM}
SpaceName=${SpaceName:=PROD}
ExperimentName=${ExperimentName:=piControl}
JobName=${JobName:=piControlMR1}
CARD_DIR=${CARD_DIR:=${CURRENT_DIR}}
```

Start the `./SE_checker.job` in interactive mode. All needed jobs `create_se.job` will be started. For example:

```
 ./SE_Checker.job

==================================================
Where do we run ? cesium21
Linux cesium21 2.6.18-194.11.4.el5 #1 SMP Tue Sep 21 05:04:09 EDT 2010 x86_64
==================================================

sys source cesium Intel X-64 lib.

--Debug1--> DefineVariableFromOption : config_UserChoices
--------------Debug3--> config_UserChoices_JobName=piControlMR1
--------------Debug3--> config_UserChoices_CalendarType=noleap
```

```
--------------Debug3--> config_UserChoices_DateBegin=1800-01-01
--------------Debug3--> config_UserChoices_DateEnd=2099-12-31

--Debug1--> DateBegin/End for SE : 1800_1809
--Debug1--> ATM
--Debug1--> SRF
--Debug1--> SBG
--Debug1--> OCE
--Debug1--> ICE
--Debug1--> MBG
--Debug1--> CPL
...
--Debug1--> DateBegin/End for SE : 2030_2039
--Debug1--> ATM
--Debug1--> 2 file(s) missing for ATM :
--Debug1--> piControlMR1_SE_2030_2039_1M_histmth.nc
--Debug1--> piControlMR1_SE_2030_2039_1M_histmthNMC.nc
--Debug1--> SRF
--Debug1--> 1 file(s) missing for SRF :
--Debug1--> piControlMR1_SE_2030_2039_1M_sechiba_history.nc
--Debug1--> SBG
--Debug1--> 2 file(s) missing for SBG :
--Debug1--> piControlMR1_SE_2030_2039_1M_stomate_history.nc
--Debug1--> piControlMR1_SE_2030_2039_1M_stomate_ipcc_history.nc
--Debug1--> OCE
--Debug1--> 4 file(s) missing for OCE :
--Debug1--> piControlMR1_SE_2030_2039_1M_grid_T.nc
--Debug1--> piControlMR1_SE_2030_2039_1M_grid_U.nc
--Debug1--> piControlMR1_SE_2030_2039_1M_grid_V.nc
--Debug1--> piControlMR1_SE_2030_2039_1M_grid_W.nc
--Debug1--> ICE
--Debug1--> 1 file(s) missing for ICE :
--Debug1--> piControlMR1_SE_2030_2039_1M_icemod.nc
--Debug1--> MBG
--Debug1--> 3 file(s) missing for MBG :
--Debug1--> piControlMR1_SE_2030_2039_1M_ptrc_T.nc
--Debug1--> piControlMR1_SE_2030_2039_1M_diad_T.nc
--Debug1--> piControlMR1_SE_2030_2039_1M_dbio_T.nc
--Debug1--> CPL
--Debug1--> 2 file(s) missing for CPL :
--Debug1--> piControlMR1_SE_2030_2039_1M_cpl_atm.nc
--Debug1--> piControlMR1_SE_2030_2039_1M_cpl_oce.nc
--------Debug2--> Submit create_se  for period 2030-2039
IGCM_sys_MkdirWork : .../POST_CMIP5/piControl/piControlMR1/OutScript
IGCM_sys_QsubPost : create_se
Submitted Batch Session 179472
...
```

7. Wait for the `create_se` jobs to be completed
8. Copy `create_multi_se.job`
9. Check/change the variables :

```
libIGCM=${libIGCM:=.../POST_CMIP5/libIGCM_v1_10/modipsl/libIGCM}
```

If needed, adjust the number of decades in `config.card`: default=`50Y` (i.e. 50 years). Add the following line to the `POST` section, i.e. at the end after the keyword `[POST]`

```
MultiSeasonalFrequency=100Y
```

11. Run the `create_multi_se.job` job:`ccc_msub create_multi_se.job`

12. The years used for the calculations are those between `DateEnd` (set in `config.card` in the local directory) and `DateEnd - MultiSeasonalFrequency`.

The mean values are stored in the "Analyse" directories of each model component in the subdirectory `SE_100Y` (e.g. `ATM/Analyse/SE_100Y`).

5. FAQ : Unix tricks

5.1. How to delete a group of files using the find command?

We recommend to also read the find manual.

Examples :

- command recursively deleting all files in a directory containing DEMO in their name:

```
find . -name '*DEMO*' -exec rm -f {} \;
```

- command recursively deleting all files in a directory containing DEMO, TEST or ENCORE in their name:

```
find . \( -name "*DEMO*" -o -name "*TEST*"  -o -name "*ENCORE*" \) -print -exec rm -f {} \;
```

- command recursively computing the number of files in the current directory:

```
find . -type f | wc -l
```

5.2. Allowing read-access to everybody

The `chmod -R ugo+rX *` command gives access to everybody to all files and subdirectories in the current directory.

6. FAQ : Miscellaneous

6.1. How do I use `TimeSeries_Checker.job` to create files on `$STORE` when the output files are on `DMNFS`?

- You need libIGCM v1_12 to use DMNFS as input
- Change TimeSeries_Checker.job to use STORE
- Change create_ts.job to use STORE

6.1.1. Example for the rcp45 simulation

```
cd modipsl
mv libIGCM libIGCM.old
svn checkout http://forge.ipsl.jussieu.fr/libigcm/svn/tags/libIGCM_v1_12 libIGCM
modipsl/ins_job
```

```
TimeSeries_Checker.job révision 658

line 169
#R_SAVE=${R_OUT}/${config_UserChoices_TagName}/${config_UserChoices_SpaceName}/${config_UserChoices_ExperimentName}/${conf

R_SAVE=${CCCSTOREDIR}/IGCM_OUT/${config_UserChoices_TagName}/${config_UserChoices_SpaceName}/${config_UserChoices_Experime
```

```
create_ts.job révision 316

line 300
```

```
#DIRECTORY=${R_SAVE}/${comp}/Analyse/${TS_Dir}
DIRECTORY=${CCCSTOREDIR}/IGCM_OUT/IPSLCM5A/PROD/rcp45/v3.rcp45.strat/${comp}/Analyse/${TS_Dir}

line 768
#eval IGCM_sys_Put_Out ${file_out} \${R_OUT_${comp}}/Analyse/${TS_Dir}/${file_out}
IGCM_sys_Put_Out ${file_out} ${CCCSTOREDIR}/IGCM_OUT/IPSLCM5A/PROD/rcp45/v3.rcp45.strat/${comp}/Analyse/${TS_Dir}/${file_o

line 780
#eval IGCM_sys_Put_Out ${file_out_YE} \${R_OUT_${comp}}/Analyse/TS_MO_YE/${file_out_YE}
IGCM_sys_Put_Out ${file_out_YE} ${CCCSTOREDIR}/IGCM_OUT/IPSLCM5A/PROD/rcp45/v3.rcp45.strat/${comp}/Analyse/TS_MO_YE/${file
```

6.2. How do I restart one simulation month which ran on $DMFDIR when the outputs are stored on $STORE?

Example: the past1000 simulation

```
login on titane: ssh titane.ccc.cea.fr

1455-12 to be reran on STORE, original on dmnfs
1477-12 to be reran on STORE, original on dmnfs
1517-11 to be reran on STORE, original on dmnfs

cd $DMFDIR/IGCM_OUT/IPSLCM5A/TEST/PD_TEST

dmget LMCMP5/???/Restart/*14551130* LMCMP5/???/Restart/*14771130* LMCMP5/???/Restart/*15171030* LMCMP5/OCE/Output/LMCMP5_m

tar cvf $SCRATCHDIR/IGCM_OUT/IPSLCM5A/TEST/PD_TEST/RESTART.REDO.past1000.tar LMCMP5/???/Restart/*14551130* LMCMP5/???/Rest

cd $SCRATCHDIR/IGCM_OUT/IPSLCM5A/TEST/PD_TEST

tar xvf RESTART.REDO.past1000.tar

cd /work/cont003/p25khod/IPSLCM5A/modipsl/config/IPSLCM5A/LMCMP5_newlibIGCM.REDO

==> edit run.card

==> PackFrequency=NONE in config.card

==> qsub

#============================================================================
#D-- Post -
[Post]
#D- Do we rebuild parallel output, this flag determines
#D- frequency of rebuild submission (use NONE for DRYRUN=3)
RebuildFrequency=1Y
#D- Do we rebuild parallel output from archive (use NONE to use SCRATCHDIR as buffer)
RebuildFromArchive=NONE
# Pas de PACK
PackFrequency=NONE
#D- If you want to produce time series, this flag determines
#D- frequency of post-processing submission (NONE if you don't want)
TimeSeriesFrequency=NONE
#D- If you want to produce seasonal average, this flag determines
#D- the period of this average (NONE if you don't want)
SeasonalFrequency=NONE
#D- Offset for seasonal average first start dates ; same unit as SeasonalFrequency
#D- Useful if you do not want to consider the first X simulation's years
SeasonalFrequencyOffset=0
```

6.3. How do I copy a model installation directory instead of downloading from the forge (or move a directory)?

Copy or move the target installation:

```
cp -r OldInstall NewInstall
 or
mv OldInstall NewInstall
```

Regenerate the makefiles to account for the new path:

```
cd NewInstall/modipsl/util
./ins_make
```

Recompile if you've done modifications in the source code:

```
cd NewInstall/modipsl/config/[YourConfig]
gmake clean
gmake [target]
```

Update your libIGCM installation:

- install the latest version of libIGCM by following these explanations
- or remove and regenerate the .job files in your libIGCM directory as follows:

```
rm NewInstall/modipsl/libIGCM/*.job
```

Prepare a new experiment as usual and launch `ins_job` to generate the `.job` files in your `libIGCM` directory and your experiment directory. Depending on your `libIGCM` version you will have to launch `NewInstall/modipsl/libIGCM/ins_job` or `NewInstall/modipsl/util/ins_job` for older versions.

Check that the `.job` files are properly generated in `NewInstall/modipsl/libIGCM/` and you are set.