

## **Wikiprint Book**

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

**Subject: Igcmg\_doc - Doc/FAQ**

**Version: 100**

**Date: 06/26/24 19:31:19**

## Table of Content

|  |           |
|--|-----------|
| <b>Frequently Asked Questions</b>  | <b>3</b>  |
| <b>1. FAQ : Setting up and performing a simulation</b>   | <b>3</b>  |
| 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         |
| <b>2. FAQ : Running the model</b>  | <b>3</b>  |
| 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         |
| 2.8. Restarting the model after a crash  | 7         |
| 2.8.1. What can I do ?   | 7         |
| 2.8.2. Procedure   | 7         |
| 2.9. How can I extend a 'MOSAIC' coupled simulation but use MOSAIX weights instead ?                               | 7         |
| 2.9.1. Why would you want to do that ?   | 7         |
| 2.9.2. Can you use your MOSAIC simulation restarts for running your MOSAIX simulation ?                            | 7         |
| 2.9.3. The trick to get around this  | 8         |
| 2.9.4. The procedure   | 8         |
| <b>3. FAQ : Special configurations</b>   | <b>10</b> |
| 3.1. How do I create the initial conditions for LMDZOR?  | 10        |
| 3.2. How do I deactivate STOMATE in IPSLCM5 or in LMDZOR?  | 10        |
| 3.3. How do I perform a nudged run?  | 10        |
| 3.4. How do I run simulations with specific versions of compiler and/or libraries on Irene at the TGCC ? (modules) | 11        |
| 3.5. How to have min and max value exchanged through OASIS?  | 12        |
| 3.6. How to output exchanged fields by OASIS?  | 13        |
| 3.7. How I create a 1pctCO2 experiment   | 14        |
| 3.8. How I create an abrupt-4xCO2 experiment   | 14        |
| <b>4. FAQ : Post processing</b>  | <b>14</b> |
| 4.1. Where are post processing jobs run?   | 14        |
| 4.2. How do I check that the post processing jobs were successful?   | 14        |
| 4.3. How do I read/retrieve/use files on thredds?  | 14        |
| 4.4. How do I add a variable to the Time Series?   | 15        |
| 4.5. How do I superimpose monitoring plots (intermonitoring)?  | 15        |
| 4.6. What is the Monitoring?   | 15        |
| 4.7. How do I add a plot to the monitoring?  | 15        |
| 4.8. How do I calculate seasonal means over 100 years?   | 15        |
| 4.9. There is over quota on thredds (TGCC), what can I do ?  | 17        |
| <b>5. FAQ : Unix tricks</b>  | <b>17</b> |
| 5.1. How to delete a group of files using the find command?  | 17        |
| 5.2. Allowing read-access to everybody   | 17        |
| <b>6. FAQ : Miscellaneous</b>  | <b>17</b> |
| 6.1. How do I copy a model installation directory instead of downloading from the forge (or move a directory)?     | 17        |
| 6.2. I need to compile IPSL model in debug mode. How to do that?   | 18        |
| 6.3. I receive an email alert from TGCC about my quota : what can I do ?   | 18        |
| 6.3.1. reminder of the rules on the use of the store   | 18        |
| 6.3.2. ccc_quota   | 18        |
| 6.3.3. ccc_tree  | 18        |
| 6.3.4. commands to analyze inodes by directories   | 19        |
| 6.3.5. email send to platform_users list on 2022/10/24   | 19        |
| 6.3.6. libIGCM surpack   | 19        |
| 6.3.7. ccc_pack  | 19        |

## 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. They are 2 ways to do it, one using the `purge` tool from libIGCM otherwise to delete everything manually.

##### 1. Use libIGCM purge

To purge your simulation (ie delete all outputs) just run:

```
path/to/libIGCM/purge_simulation.job
```

##### 2. Manual purge

To remove all outputs created by simulation, do the following:

1. Delete the `run.card` file in your experiment directory.
2. Delete all output directories:
  - `STORE/IGCM_OUT/TagName/(...)/JobName`
  - `WORK/IGCM_OUT/TagName/(...)/JobName`
  - `SCRATCH/IGCM_OUT/TagName/(...)/JobName`

| Space   | TGCC            | IDRIS     |
|---------|-----------------|-----------|
| WORK    | \$CCCWORKDIR    | \$WORK    |
| SCRATCH | \$CCCSCRATCHDIR | \$SCRATCH |
| STORE   | \$CCCSTOREDIR   | \$STORE   |

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](#).

### 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.

Click [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?

It's define in config.card file, and the script ins\_job will define what it's need in the job header.

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". For example, for LMDZOR : we choose to run with 71 MPI processes and 8 OpenMP threads for LMDZ, and 1 MPI for XIOS

```
ATM= (gcm.e, lmdz.x, 71MPI, 8OMP)
SRF= ("", "")
SBG= ("", "")
IOS= (xios_server.exe, xios.x, 1MPI)
```

In this case the job will ask for  $71 \times 8 + 1 = 569$  CPU

If we don't use OpenMP parallelization

```
ATM= (gcm.e, lmdz.x, 71MPI, 1OMP)
SRF= ("", "")
SBG= ("", "")
IOS= (xios_server.exe, xios.x, 1MPI)
```

In this case the job will ask for  $71 + 1 = 72$  CPU

## 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 (for ORCHIDEE\_OL)

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 new 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 -r `number_revision` http://forge.ipsl.fr/libigcm/svn/trunk/libIGCM libIGCM
```

where `number_revision` is specified by someone from [PlatForm?](#) group.

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 IPSLCM6 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?

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

For reminder

| Space   | TGCC            | IDRIS     |
|---------|-----------------|-----------|
| WORK    | \$CCCWORKDIR    | \$WORK    |
| SCRATCH | \$CCCSCRATCHDIR | \$SCRATCH |
| STORE   | \$CCCSTOREDIR   | \$STORE   |

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

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

```
## Directory REDO
mkdir STORE/...IGCM_OUT/IPSLCM5A/REDO/historical/v3.hist
cd STORE/...IGCM_OUT/IPSLCM5A/REDO/historical/v3.hist

# RESTART
mkdir -p RESTART ; cd RESTART
ln -s ../../../../PROD/historical/v3.hist/RESTART/v3.hist_19631231_restart.nc v3.hist_19631231_restart.nc
```

- If you are running a coupled model : On the scratch disk (SCRATCH/IGCM\_OUT), create the `mesh_mask` file

```
mkdir SCRATCH/...IGCM_OUT/IPSLCM5A/REDO/historical/v3.hist
cd SCRATCH/...IGCM_OUT/IPSLCM5A/REDO/historical/v3.hist

# mesh_mask
mkdir -p OCE/Output
cd OCE/Output
ln -s ../../../../PROD/historical/v3.hist/OCE/Output/v3.hist_mesh_mask.nc v3.hist_mesh_mask.nc
```

- On the computing machine:
  - create a new directory

```
cp -pr v3.hist v3.histREDO
```

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

```
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
PeriodState= OnQueue
SubmitPath=...../v3.histREDO
```

- change the config.card file to one pack period (1 year), do not do any post processing, start rebuild month by month (only for ORCHIDEE\_OL) and specify PackFrequency. *JobName and DateBegin will not be changed.*

```
JobName=v3.hist
...
SpaceName=REDO
...
DateEnd= 1964-12-31
...
RebuildFrequency=1M # only for ORCHIDEE_OL
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.hist # check the time parameters and names of the output scripts
ccc_msub Job_v3.hist
```

- 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 STORE/.../REDO/.../OCE/Debug/v3.hist_19640901_19640930_solver.stat STORE/IGCM_OUT/IPSLCM5A/PROD/historical/v3.hist
```

## 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 verific
    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; ./make_lmdz_fcm -cpp ORCHIDEE_NOOPENMP -d $(RESOL_LMDZ) -cosp true -v true -parallel mpi -arch $(FC)
(cd ../../modeles/LMDZ; ./make_lmdz_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...
```

**Warning: you'll need to get parameter files and maybe some forcing ones corresponding to the resolution.**

## 2.8. Restarting the model after a crash

Sometimes the coupled model crashes. Most common crashes occurs either in LMDZ or NEMO.

LMDZ triggers an emergency stop when the temperature reaches values outside an acceptable range. It is mainly due to instabilities in the convection scheme. High resolutions are more prone to crash.

NEMO triggers an emergency stop when the salinity becomes negative, or when the current is larger than 10 m/s. It happens mostly in ice covered regions. This is mainly due to and instable vertical advection scheme when a high vertical resolution is used (dz=1 m).

### 2.8.1. What can I do ?

Often, it is possible to restart the model. The procedure consists of adding a small perturbation to the coupler restarts. You can use a procedure that will add a random perturbation to a selected field. Generally, with perturb the sea surface temperature. It will change the trajectoire of the model and, hopefully, avoid the crash. If the models crashes often, you better discuss with some model specialist(s)

### 2.8.2. Procedure

- In the model launching directory, run `../../../../libIGCM/clean_PeriodLength.job` to cleanup the last period.  
In the file `COMP/oasis.driver`, add in the `[UserChoices]` section the lines :

```
ByPass_addnoise_sst=y
ByPass_PerturbExe=AddNoise
ByPass_FileOut=sstoc
ByPass_PerturbVar=O_SSTSST
ByPass_PerturbAmp=0.1
```

Note that the model will run `AddNoise` only once, and switch automatically to `ByPass_addnoise_sst=n`

## 2.9. How can I extend a 'MOSAIC' coupled simulation but use MOSAIX weights instead ?

**Important** To apply this procedure only the interpolation weights used for OASIS should differ. All the other boundary conditions and forcings should be the same : bathymetry, icesheet, aerosols, solar forcings, etc...

### 2.9.1. Why would you want to do that ?

Extending a simulation using MOSAIC weights and switching to MOSAIX weights is mostly relevant for long simulations at equilibrium (for instance paleoclimates deeptime simulations). The improved interpolation schemes used in MOSAIX result in a better conservation of some quantities. One of the desirable effects it can have is a reduction of sea surface height drift which can become non negligible with simulations over several millenia.

### 2.9.2. Can you use your MOSAIC simulation restarts for running your MOSAIX simulation ?

The short answer is : no, you can't do it.

The long answer is : no, not as an actual restart for all the components, but you can still use your MOSAIC simulation to initialise a MOSAIX run with the same ocean state.

## Why is it so ?

The problem originates from the combination of two facts.

### Firstly

Because the o2a.nc files produced by MOSAIC and MOSAIX differ (they have different coastlines), when interpolating from the ocean grid to the atmosphere grid points very close to the coast, land points can become sea points and vice versa if you switch from the MOSAIC to the MOSAIX weights.

### Secondly

The o2a.nc file is read only once by LMDZ : when producing restart files and boundary conditions for a forced simulation with create\_etat0\_limit.e and is embedded in the restart files. It is then transmitted through the restarts to all subsequent simulations.

**The actual problem** When you try to use MOSAIX interpolation weights and restart LMDZ from a simulation with an o2a.nc file produced by MOSAIC embedded in the LMDZ restarts, you will get wrong values for the interpolation of quantities located at the points where the coastlines differ. This can result in numerical instability and the code blowing or, much worse, wrong fluxes at the faulty points and false budgets but the code running fine !

### 2.9.3. The trick to get around this

The problem being with the atmosphere, you can still use your NEMO restart files as they are to restart your MOSAIX coupled simulation. (i.e. keep the dynamical state of the ocean)

For LMDZ you will have to restart the atmosphere from an initial state created with create\_etat0\_limit.e to make sure you do use the MOSAIX o2a.nc file.

Still, it is possible to improve the initial state of the atmosphere by running a forced simulation with the ocean surface quantities from the MOSAIC simulation.

You can use the last year of the MOSAIC run or a climatology of the last years to equilibrate the atmosphere and then use this as an initial state for your MOSAIX run.

### 2.9.4. The procedure

- take the last year of sst and sea ice coverage (sic) from LMDZ in your MOSAIC coupled simulation.

For a simulation named CM5A2WPIREF we can extract the variables `tsol_oce` and `pourc_sic` from the LMDZ `histmth.nc` file `CM5A2WPIREF_20400101_20491231_1M_histmth.nc` and extract the last year with `nco` and `cdo`:

```
ncks -v tsol_oce,pourc_sic CM5A2WPIREF_20400101_20491231_1M_histmth.nc CM5A2WPIREF_20400101_20491231_1M_tsol_oce_pourc_sic.nc
cdo seldate,2049-01-16,2049-12-16 CM5A2WPIREF_20400101_20491231_1M_tsol_oce_pourc_sic.nc CM5A2WPIREF_20490101_20491231_1M_
```

- create restart files and `limit.nc` for forcing the atmosphere with `create_etat0_limit.nc`.

You will need to edit `COMP/lmdz.card`:

```
[BoundaryFiles]
List= ()

ListNonDel= (${R_IN}/ATM/Albedo.nc, .), \
  (${R_IN}/ATM/ECDYN.nc.20020101, ECDYN.nc), \
  (${R_IN}/ATM/ECDYN.nc.20020101, ECPHY.nc), \
  (${R_IN}/ATM/INPUT_CE0L/Relief_orig.nc, Relief.nc), \
  (${R_IN}/ATM/Rugos.nc, .), \
  (${R_IN}/ATM/INPUT_CE0L/landiceref_orig.nc, landiceref.nc), \
  (${R_IN}/ATM/Ozone/HYBRIDE/v2.clim/tro3_1995.new.nc, climoz.nc), \
  (/ccc/store/cont003/gen2212/nguyens/IGCM_OUT/IPSLCM5A2/PROD/piControl/CM5A2WPIREF/ATM/Output/MO/CM5A2WPIREF_20490101_20491231_1M_histmth.nc, histmth.nc), \
  (/ccc/store/cont003/gen2212/nguyens/IGCM_OUT/IPSLCM5A2/PROD/piControl/CM5A2WPIREF/ATM/Output/MO/CM5A2WPIREF_20490101_20491231_1M_tsol_oce_pourc_sic.nc, tsol_oce_pourc_sic.nc), \
  (/ccc/work/cont003/gen2212/nguyens/MOSAIXLGM/MOSAIXPI/LMD9695_grid_maskFrom_ORCA2.3_MX_v2.nc, o2a.nc)
```

In particular, the last three lines point toward the surface forcings coming from a `histmth.nc` file and the `o2a.nc` file generated by MOSAIX.

The names `histmth_sst.nc` and `histmth_sic.nc` are imposed by the type of file given to LMDZ, i.e. an `histmth.nc` file. For using PCMDI `amip` files you would give LMDZ different files named `amipbc_sic_1x1.nc` and `amipbc_sst_1x1.nc` containing the data with different variable names.

**Now you can run create\_etat0\_limit.e**

- run LMDZOR to prepare an atmosphere at equilibrium with the boundary conditions

For this prepare a LMDZOR/clin\_pdControl with external forcings adjusted to your desired period (pi, pd, paleo, annual...). Modify COMP/lmdz.card to impose these conditions and use the restart and limit files:

```
[UserChoices]

...

LMDZ_Physics=AP      # for IPSLCM5A2 physics parameterizations

# Set the variable CREATE needed further below to find the initial and boundary condition files.
CREATE=YOUR_CREATE_ETAT0_SIMULATION

# Set ConfType to choose parameters for aerosols, solar and green house gazes.
# The parameter file PARAM/config.def_$ConfType will be used.
# ConfType=preind/actuel/annuel
ConfType=preind

...

[InitialStateFiles]
List=  (${ARCHIVE}/IGCM_OUT/LMDZ/${CREATE}/ATM/Output/Restart/${CREATE}_clim_start.nc, start.nc),\ # should point t
      (${ARCHIVE}/IGCM_OUT/LMDZ/${CREATE}/ATM/Output/Restart/${CREATE}_clim_startphy.nc, startphy.nc)

[BoundaryFiles]
List=()
ListNonDel= (${ARCHIVE}/IGCM_OUT/LMDZ/${CREATE}/ATM/Output/Boundary/${CREATE}_clim_limit.nc, limit.nc),\ # should point t
            (${R_IN}/ATM/${RESOL_ATM}/AR5/HISTORIQUE/aerosols_11YearsClim_1855_v5.nc, aerosols.nat.nc), \ # the following l
            (${R_IN}/ATM/${RESOL_ATM}/AR5/HISTORIQUE/aerosols_11YearsClim_1855_v5.nc, aerosols1980.nc), \
            (${R_IN}/ATM/${RESOL_ATM}/AR5/HISTORIQUE/climoz_LMDZ_1855_v2.nc, climoz_LMDZ.nc)
```

Edit OMP/orchidee.card to use the appropriate initial files:

```
[InitialStateFiles]
List=  (${R_IN}/SRF/soils_param.nc, . ), \
      (${R_IN}/SRF/routing.nc, . ), \
      (${R_IN}/SRF/PFTmap_1850to2005_AR5_LUHa.rc2/PFTmap_IPCC_1860.nc, PFTmap.nc), \
      (${R_IN}/SRF/cartepente2d_15min.nc, .)
```

**Now you can run 2 to 10 years of LMDZOR** to equilibrate the atmosphere with the oceanic forcings.

- Create a coupler restart file flxat.nc using the end of the previous LMDZOR run

You will need to download and run the [CPLRESTART](#) tools.

In some working directory download the CPLRESTART tools from the "IPSL forge" svn repository with:

```
svn co http://forge.ipsl.fr/igcmg/svn/TOOLS/CPLRESTART .
```

This will create a CPLRESTART directory. Use CreateRestartAtm4Oasis.bash shell script to create the flxat.nc file from the histmth.nc file for the last year or your previous LMDZOR run.

```
CreateRestartAtm4Oasis.bash --oce ORCA2.3 $PATHTOFILE/LMDZOR_18690101_18691231_1M_histmth.nc
```

The --oce ORCA2.3 will just write some information in the flxat file and name it accordingly to help you remember the origin of this file. You may need to adjust the modules loaded to make the shell script and its associated python script create\_flxat.py work on your machine:

```

module purge          # works on irene skl with redhat 7
module load hdf5
module load netcdf-c
module load nco/4.9.1
module load cdo/1.9.5
module load python3/3.7.2
module load datadir/igcmg
module list

```

- You can now run your MOSAIX simulation

Fill your config.card in order to:

**restart all the oceanic components from the MOSAIX coupled simulation** restart the atmosphere and vegetation components from the LMDZOR forced simulation **restart the coupler from initial state files:**

```

#=====
#D-- Restarts -
[Restarts]
...

#=====
#D-- CPL -
[CPL]
WriteFrequency="1M"
# If config_Restarts_OverRule == 'n' next 4 params are read
Restart= n
...

```

Lastly, modify the COMP/oasis.card file to get the initial files for the coupler:

```

[InitialStateFiles]
List= (/ccc/work/cont003/gen2212/nguyens/PALEO/CPLRESTART/PIMX/flxat_LMD9695_maskFrom_ORCA2.3.nc, flxat.nc), \
      (${R_IN}/RESTART/IPSLCM5A2/PROD/piControl/CM5A2.1.pi.debug/CPL/Restart/CM5A2.1.pi.debug_50091231_sstoc.nc, sstoc.nc)

```

**You can now run your MOSAIX coupled simulation extending the previous MOSAIX one**

### 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 IPSLCM6.

For more information, see [this chapter](#).

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

[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 Irene)

For example:

```
[BoundaryFiles]
List= ....\
      (work_subipsl/subipsl/ECMWF{your_resolution}/AN${year}/u_ecmwf_${year}${month}.nc, u.nc)\
      (work_subipsl/subipsl/ECMWF{your_resolution}/AN${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 Irene 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_irene.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
. $CCCHOME/../../dsm/p86ips1/.atlas_env_netcdf4_irene_ksh > /dev/null 2>&1
# to run with netcdf 3.6.3 ie compilation done before 17/2/2014
# uncomment 2 lines :
# module unload netcdf
# module load netcdf/3.6.3
# set the proper modules
module unload nco
module unload netcdf
module unload hdf5
module unload intel
module load intel/12.1.9.293
module load netcdf/3.6.3_p1
module load hdf5/1.8.8
module load nco/4.1.0
#set the proper modules end
export PATH=${PATH}:$CCCHOME/../../dsm/p86ips1/AddNoise/src_X64_IRENE/bin
export PATH=${PATH}:$CCCHOME/../../dsm/p86ips1/AddPerturbation/src_X64_IRENE/bin
else
. $CCCHOME/../../dsm/p86ips1/.atlas_env_netcdf4_irene_ksh > /dev/null 2>&1
PCMDI_MP=$CCCHOME/../../dsm/p86ips1/PCMDI-MP
fi

```

This way you can launch experiments on IRENE 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 libGCM 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 (LOGPRT 1) , to add 2 operations (4 instead of 2 operations, CHECKIN CHECKOUT) and to describe them (INT=1 added for CHECKIN and for CHECKOUT). Then you will find information in output text file.

Example :

- Modification in `namcouple` :
  - Before :

```

$NLOGPRT
0
...
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
...
O_SSTSST SISUTESW 1 5400 4 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 in debug.root.01 file.

```

> 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 in debug.root.02

```

> 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_AlIce
oasis_advance_run at         -2700          0 SEND: O_OCury1
oasis_advance_run at         -2700          0 SEND: O_OCury1
oasis_advance_run at         -2700          0 SEND: O_OCury1
oasis_advance_run at          2700        5400 SEND: O_SSTSST
diags:          O_SSTSST  0.271306391122    304.852847163    31680753.5627
...

```

### 3.6. How to output exchanged fields by OASIS?

To have output of exchanged fields by OASIS, one have to set 3 parameters :

- OutputMode=y in COMP/oasis.card
- WriteFrequency="1M 1D" : Add 1D write frequency in config.card for CPL section
- RebuildFrequency=1D : Add a post rebuild step ie frequency for rebuild in config.card for Post section

Then you will obtain 2 types of files :

- DA/...\_1M\_cpl\_oce.nc and ...\_1M\_cpl\_atm.nc variables in ocean (resp. atmosphere) received or sent to the other component, for each exchange. (17, 16, 3 or 2 values per day., 0, 1, 14 or 15 extra values forced to 0)
- MO/...\_1M\_cpl\_oce.nc and ...\_1M\_cpl\_atm.nc variables in ocean (resp. atmosphere) received or sent to the other component, averaged per month. result of cdo monavg and ncatted -a axis,time,c,c,T -a long\_name,time,c,c,Time axis -a title,time,c,c,Time -a calendar,time,c,c,noleap -a units,time,c,c,seconds since ... -a time\_origin,time,c,c,...

On last improvement still to be done to have the calendar of the simulation and the right number of values.

### 3.7. How I create a 1pctCO2 experiment

Take a preindus simulation, and in lmdz.card add the CO2.txt file

```
ListNonDel= (...),\
            (${R_IN}/ATM/GHG/CMIP6/1pctCO2/CO2_CMIP6_1pctCO2_1850_2100.txt, CO2.txt)
```

in config.card change the ExperimentName

```
ExperimentName=1pctCO2
```

### 3.8. How I create an abrupt-4xCO2 experiment

Take a preindus simulation, and modify the CO2 concentration in config.def\_preind file

```
co2_ppm = 1137.28
```

in config.card modify the ExperimentName

```
ExperimentName=abrupt-4xCO2
```

You can find some information [here](#)

## 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          |
|--------|-------------------|--------------------------|
| TGCC   | Irene             | xlarge node, -q standard |
| IDRIS  | JeanZay           | --partition=prepost      |

### 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 thredds?

Visit the following website <https://thredds-su.jpsl.fr/thredds/catalog/catalog.html> and:

- at IDRIS, select idris\_thredds, work, your login, your configuration, your simulation and the component (for example ATM) then the Analyse subdirectory for TS and SE, as well as ATLAS or MONITORING;

- at TGCC, select `tgcc_thredds`, `work` or `store`, your login, your configuration, your simulation and the component (for example ATM) then the Analyse subdirectory for TS and SE, as well as ATLAS or MONITORING;  
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 replace catalog by `dodsC` in the url. For example:

```
ciclad : ferret ...
> use "https://thredds-su.ipsl.fr/thredds/dodsC/MACHINE_thredds/store_or_work/yourlogin/.../file.nc"
```

More information on Monitoring can be found here: [Doc/Running](#)

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

See this [section](#).

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

You can use the intermonitoring webservice:

■ [Audio](#)

Short link :

- for esgf type : <http://webservices2017.ipsl.fr/interMonitoring/>

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 **Run and post-proc**, section *Monitoring and Intermonitoring* [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 10-year ATM atlas for CM61-LR-pi-03 here : [■SE ATM 2000-2009](#)

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/modips1/libIGCM}
```

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

```
libIGCM=${libIGCM:=.../POST_CMIP5/libIGCM_v1_10/modips1/libIGCM}
SpaceName=${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/modips1/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`).

## 4.9. There is over quota on thredds (TGCC), what can I do ?

The thredds space is regularly over quota in number of inodes.

Reminder: Normally no file is stored only in this space: there are only hard links of files stored on the `workdir` of your projects. These hard links are not counted in the volume quota. Here is the command to locate files that follow the rule.

```
cd $CCCWORKDIR/../../thredds/VOTRELOGIN find -links 1
```

Command to remove these files after having carefully check the list.

```
cd $CCCWORKDIR/../../thredds/VOTRELOGIN find -links 1 -exec \rm {} \;
```

## 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 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/modips1/util
./ins_make
```

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

```
cd NewInstall/modips1/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/modips1/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/modips1/libIGCM/ins_job` or `NewInstall/modips1/util/ins_job` for older versions.

Check that the .job files are properly generated in `NewInstall/modips1/libIGCM/` and you are set.

## 6.2. I need to compile IPSL model in debug mode. How to do that?

You have to modify Makefile to add debug option for each component :

```
(cd ../../modeles/ORCHIDEE/ ; ./makeorchidee_fcm -debug -parallel mpi_omp -arch $(FCM_ARCH) -j 8 -xios2)
```

```
(cd ../../modeles/LMDZ ; ./makelmdz_fcm -d $(RESOL_LMDZ) -mem -debug ...
```

```
cd ../../modeles/XIOS ; ./make_xios --arch $(FCM_ARCH) --debug
```

and in SOURCES/NEMO/arch-X64\_IRENE.fcm add traceback :

```
%FCFLAGS -i4 -r8 -O3 -traceback -fp-model precise
```

```
gmake clean
gmake
```

## 6.3. I receive an email alert from TGCC about my quota : what can I do ?

### 6.3.1. reminder of the rules on the use of the store

- average files size > 1Gb
- less than 80% of files with a size < 550MB

### 6.3.2. ccc\_quota

- the `ccc_quota` command will give you informations the average files size, and the percentage of file smaller than 500MB
- use option `-h` to know how use this command

### 6.3.3. ccc\_tree

- the `ccc_tree` command will give you specific informations on each directory of your storidir. For each of them you will know the repartition between small files and big files

### 6.3.4. commands to analyze inodes by directories

sort sub-directories by numbers of inodes

```
find -maxdepth 1 -type d | while read -r dir; do printf "%s:\t" "$dir"; find "$dir" | wc -l; done | sort -n -k 2
```

sort sub-directories by number of inodes below of 32M (size can be modify)

```
find -maxdepth 1 -type d | while read -r dir; do printf "%s:\t" "$dir"; find "$dir" -type f -size -32M | wc -l; done | sort -n -k 2
```

### 6.3.5. email send to platform\_users list on 2022/10/24

As a reminder the TGCC has a monitoring system that triggers alerts when on the storidir of an account there are: more than 500 files AND { 80% of the files are smaller than 500MB OR the average file size is smaller than 1GB }

The `ccc_quota` command allows you to know, among other things, the percentage of files whose size is less than 500MB (very small files), as well as those whose size is less than 1GB (small files), and gives you the average size of your files.

The `ccc_tree` command will tell you directory by directory the proportion of very small files, small files, the average size of the files, but also the number of inodes that are not files. All this corresponds to a score out of 20. If the score is higher than 10/20 the directory appears in green, otherwise it appears in red. Ideally you should have a maximum of green directories.

As a reminder, beyond the conditions for triggering an alert, it is strongly recommended to store only large files on the storidir and to store small files on the workdir.

### 6.3.6. libGCM surpack

You can use the IPSL tools to create [surpack](#) on your simulation.

### 6.3.7. ccc\_pack

This command will pack your simulation with several tar on your storidir or on your workdir. You need to launch it from the directory where your simulation is stored. Before to do it you need to demigrate your files

```
ccc_hsm get -r your_directory
```

and ask top not remigrate them

```
ccc_hsm hint your_directory
```

Then you can use the command as describe here

```
ccc_pack -j 1 --partition=skylake --account=gen*** --src=*** --dst=*** --log-file=pack.txt --filesystem=work,store --auto-
```

where : *dst* is the complete path where we will store the archive packed (you need to create the directory if it not exist already) / *src* is the directory we want to pack / *log-file* is the name of the log file of the pack.

For example:

```
to pack simulation call "Simulation_test"

cd /path/store/Simulation_test
cd ..

ccc_hsm get -r Simulation_test
ccc_hsm hint Simulation_test

mkdir Simulation_test_pack/
```

```
ccc_pack -j 1 --partition=skylake --account=gen*** --src=Simulation_test/ --dst=Simulation_test_pack/ --log-file=Simulation
```