Wikiprint Book

Title: 1. Prepare ensembles with ins_job -e

Subject: lgcmg_doc - Doc/Setup/Ensemble

Version: 37

Date: 06/29/24 09:28:41

Table of Content

Ensemble setup

This chapter describes how to setup your simulation once you have compiled your configuration at a choosen resolution.

In this chapter we describe how to generate/use **ensemble** simulation.

1. Prepare ensembles with ins_job -e

To create an ensemble configuration you need to create an ensemble.card file.

NOTE: a template of ensemble.card is given in IPSLCM6/EXPERIMENTS/IPSLCM/dcppAhindcast_CMIP6 for IPSLCM6 model.

When IPSLCM6.1.9-LR is downloaded with ./model IPSLCM6.1.9-LR it will offer the possibility to launch experiments of the decadal type.
To prepare an ensemble of simulations copy the config.card and ensemble.card files from the directory:

```
modipsl/config/IPSLCM6/EXPERIMENTS/IPSLCM/dcppAhindcast_CMIP6
```

into the directory:

```
modipsl/config/IPSLCM6/
```

Two type of esemble are allowed with this template:

1. Ens_DATE : allows to configure simulations starting from different restart dates, NON-Periodic.
1. Ens_PERTURB : allows to configure simulations of differents members from an initial condition which is perturbed, PERIODIC, (it is possible to create different initial state perturbing Sea Surface Temperature with a with noise)

All parameters for ensemble description are in ensemble.card and in config.card.

1.1. Usage

Check that COMP, POST, PARAM and DRIVER directories are present in the experiment folder.
Once ensemble.card and config.card are correctly filled, to create an ensemble simply type:

    ../../libIGCM/ins_job -e

This will create :

- all the directories of the ensemble
- Qsub.*xxx*.sh : shell script to submit all jobs (PeriodNb=5 for all simulations)
- Qclean.PeriodLengt.*xxx*.sh : shell script to clean (if necessary) all simulations after an error, and to re-launch them
- Qclean.latestPackperiod.*xxx*.sh : shell script to clean (if necessary) last packed period after an error, and to re-launch them

Note: *xxx* it will be the *JobName* configured in config.card

NOTE: If a directory exists, ins_job won't modify it, the creation of ensemble it will stopped.

1.2. Config.card

The file *config.card* is filled as a regular *config.card* (**ins_job** without the -e option). It will be used as a template for all simulations that will be created.

The important lines for the ensemble set up are in the [UserChoices] section. Make sure that JobName and ExperimentName are filled with proper values.
The variables DateBegin and DateEnd will be overidden by variables present in ensemble.card.

```
#
# This is config.card file for IPSLCM6 configuration
#
```

```
#========================================================================
#D-- Compatibility -
[Compatibility]
libIGCM=1.0
#D-- UserChoices -
[UserChoices]
#==========================
JobName=CM619-LR-dcppA-hindcast
#----- Short Name of Experiment
ExperimentName=dcppA-hindcast
#----- DEVT TEST PROD
SpaceName=DEVT
LongName="IPSLCM6.1.9-LR"
TagName=IPSLCM6
ModelName=IPSL-CM6A-LR
Member=r1i1p1f1
#D- Choice of experiment in EXPERIMENTS directory
ExpType=IPSLCM/dcppAhindcast_CMIP6
#==========================
#-- leap, noleap, 360d
CalendarType=leap
#-- Experiment dates : Beginning and ending
#-- "YYYY-MM-DD"
DateBegin=1961-01-01
DateEnd=1980-12-31
#==========================
ORCA_version=eORCA1.2
#==========================
#-- 1Y, 1M, 5D, 1D Period Length of one trunk of simulation
PeriodLength=1Y
```

A section `[Ensemble]` should also be present. It contains the information that we want to prepare an ensemble simulation with variable EnsembleRun set to y and three unset fields to be filled in the *config.card* of each member after **'ins_job -e** has run.

```
#==========================
[Ensemble]
#D- Ensemble run ? 'y' or 'n'
#D- If 'y', fill in ensemble.card !!
EnsembleRun=y
EnsembleName=
EnsembleDate=
EnsembleType=
#
```

1.3. Ensemble.card

It is possile to choice between 2 type of esnembles in *ensemble.card*: Date restart ensemble which allows to configure simulations starting from different restart dates: need to fill section `[Ens_DATE]` Perturb ensemble which allows to generate members from an initial condition which is perturbed by different means: need to fill section`[Ens_PERTURB]`

The choice between 2 type of ensembles set the variable `active` to *y* of section tat you want activate: To activate Date Restart non-Periodic Ensembles:

```
[Ens_DATE]
active=y
```

To activate Perturbed Restart Periodic Ensembles: **NOTE :** that Period can be also just 1Y (to run non-periodic perturbed ensembles)

```
[Ens_PERTURB]
active=y
```

1.4. Configure a Date Restart ensemble

The « Date Restart ensemble » is implemented to configure a set of simulations using several restart dates, generally chosen for a particular point (ex : randomly, particular climate oscillation phases, volcanic activity…).

In ensemble.card all configuration items of this ensemble are in `[Ens_DATE]` section.
The configuration to define restarts dates is non-periodic one, it needs to define a list of desired restarts.
Fill the following options : `active`, `NAME`, `LENGTH`, `INITFROM` and `INITPATH`.

```
[Ens_DATE]
# for using date ensemble, 'n' else.
active=y

# name of the ensemble (used to create root directory)
NAME= ENSTAMBORA

# default length of the simulation for non periodic and duration for all periodic (in Year or Month)
LENGTH=10Y

# Experiment name to find all restart files (and default one for non-periodic)
INITFROM=v3.historical6

# Restart root directory
INITPATH=/ccc/store/cont003/dsm/p86denv/dmf_import/IGCM_OUT/IPSLCM5A/PROD/historical
```

Periodic start dates

In ensemble.card, it is possible to specify a periodic list of start dates. Restart files will be generated for each member at each date starting from `BEGIN_INIT` to `END_INIT` with a periodicity of `PERIODICITY`, using `BEGIN_RESTART` as first restart. Leave all *NON_PERIODIC* options empty (`NONPERIODIC`, `RESTART_NONPERIODIC`, `INITFROM_NONPERIODIC`, `LENGTH_NONPERIODIC`).

The following part of ensemble.card sets 10 years simulations from 1990-01-01 to 2000-01-01 every 2 years each with a restarts starting from 1814-12-31 every 2 years:

```
# start date of the first periodic simulation
BEGIN_INIT=19900101

# start date of the last periodic simulation
END_INIT=20000101

# duration between the start of 2 periodic simulations
PERIODICITY=2Y

# date for the first restart (next = first+periodicity). CAUTION of the calendar (use config.card one)!
BEGIN_RESTART=18141231
```

This will produce simulations starting at the dates : 1990-01-01, 1992-01-01, 1994-01-01, 1996-01-01, 1998-01-01, 2000-01-01. (`PERIODICITY` can be given in months for shorter periods).

The restart files are taken from `BEGIN_RESTART` every `PERIODICITY` step : 1814-12-31, 1816-12-31, 1818-12-31, etc...

The directory in which the start date is retrieved is given by `INITPATH` and `INITFROM`.

To restart from experiment *v3.historical6* in directory */ccc/store/cont003/dsm/p86denv/dmf_import/IGCM_OUT/IPSLCM5A/PROD/historical* fill:

```
# Restart name
INITFROM= v3.historical6

# Restart directory
/ccc/store/cont003/dsm/p86denv/dmf_import/IGCM_OUT/IPSLCM5A/PROD/historical
```

CAUTION: The variable CalendarType from *config.card* will be used to determine the next restart date. It should be consistent with the simulations from which you are initialising.

Non-Periodic start dates

In ensemble.card, it is also possible to specify manually all simulations running and restart dates, length, experiment names and directories to get restart files.

First, you need to left empty the periodic attributes `BEGIN_INIT`, `END_INIT`, `PERIODICITY` and `BEGIN_RESTART` in ensemble.card. Then you can list the start date of all simulations with `NONPERIODIC` variable, all restart dates with `RESTART_NONPERIODIC` one, all experiments to get restart files in `INITFROM_NONPERIODIC`, all simulations restart path using `INITPATH_NONPERIODIC` and give the length of each simulation (`LENGTH_NONPERIODIC`).

Here is an example of a configuration :

```
# list of start dates for all simulations
NONPERIODIC=(18150101 19910101 19990101)

# list of corresponding restart dates
RESTART_NONPERIODIC=(18141230 19901230 19981231)

# simulation name to restart for each simulation. IF empty all simulations will use INITFROM one.
INITFROM_NONPERIODIC=( v3.historical6 v3.historical6 v5.historical1)

# directory of the restart for each simulation. IF empty all simulations will use INITPATH one.
INITPATH_NONPERIODIC= ( path/to/1st path/to/2nd path/to/3rd )

# length of each simulation. If empty all simulations duration will be the default LENGTH option.
LENGTH_NONPERIODIC=(10Y 10Y 50Y)
```

WARNING: For list variables, use blank between values (no coma).

This will produce 3 simulations which starting at the dates : 1815-01-01, 1990-01-01 and 1999-01-01using respectively restarts from 1814-12-30, 1990-12-30, 1998-12-31 (note that the calendar should be different from the config.card one) taking into *v3.historical6* experiment for the 2 firsts and from *v5.historical1* for the last one (`INITFROM` is ignored when `INITFROM_NONPERIODIC` is filled). Restarts will be taken respectively in the 3 directories specified with `INITPATH_NONPERIODIC` (`INITPATH` is ignored when `INITPATH_NONPERIODIC` is filled). Simulations length will be 10 years for the 2 firsts and 50 years for the last one. All restart experiments should be in the directory */ccc/store/cont003/dsm/p86denv/dmf_import/IGCM_OUT/IPSLCM5A/PROD/historical.*

Notice that `INITFROM_NONPERIODIC`, `LENGTH_NONPERIODIC` and `INITPATH_NONPERIODIC` are not mandatory for non-periodic configuration. If you don't fill one of them or all the `INITFROM` value and/or `LENGTH` value and/or `INITPATH` will be used for all simulations :

```
# default length of the simulation for non periodic and duration for all periodic (in Year or Month)
LENGTH=10Y
[…]

# list of start dates for all simulations
NONPERIODIC=(18150101 19910101 19990101)

# list of corresponding restart dates
RESTART_NONPERIODIC=(18141230 19901230 19901231)

# simulation name to restart for each simulation. IF empty all simulations will use INITFROM one.
INITFROM_NONPERIODIC=

# length of each simulation. IF left empty all simulations durations will be the default LENGTH option.
LENGTH_NONPERIODIC=

# directory of the restart for each simulation. IF empty all simulations will use INITPATH one.
INITPATH_NONPERIODIC=

# Restart name
```

```
INITFROM= v3.historical6

# Restart directory
/ccc/store/cont003/dsm/p86denv/dmf_import/IGCM_OUT/IPSLCM5A/PROD/historical
```

This will produce 3 simulations starting at the dates : 1815-01-01, 1990-01-01 and 1999-01-01 using respectively restarts from 1814-12-30, 1990-12-30, 1990-12-31.

All of them use *v3.historical6* experiment in `/ccc/store/cont003/dsm/p86denv/dmf_import/IGCM_OUT/IPSLCM5A/PROD/historical`
directory to get restart files and their duration is 10 years.

1.5. Configure a Perturbed ensemble

We cover here the section which allows to generate members from an initial condition which is perturbed by different means.

There are two ways to perturb the initial condition:

- apply some random white noise of defined amplitude to the temperature field of the coupler component (CPL) restart file
- apply some previously generated 3D temperature perturbation map to the temperature field of the ocean component (OCE) restart file

Each method applies only to the relevant type of ensemble generation available inside [Ens_PERTURB] as will be explained later.

Before detailing the different functionalities available in [Ens_PERTURB] let us discuss the NAME variable.
This variable will be both the global name of the ensemble (ie directory name) and the prefix for each member:

```
# ensemble name
NAME=v3h4testB
```

JobName variable in `config.card` will be the name of the root directory that would be created containing all config and script files and the ensemble.

Periodic start dates

For this type of perturbed ensembles the following variables are left empty:

```
# member list (apply list of pattern to initial state)
PERTU_MAP_LIST=()

# member list of names corresponding to each member
MEMBER_NAMESLIST=()

# member pattern global name
MEMBER_INITFROM=

# member pattern global directory for name
MEMBER_INITPATH=
...
# start dates list
NONPERIODIC=()
# length list for non periodic simulation (NOTE: use length above if not fill)
LENGTH_NONPERIODIC=()
...
# Path of Mask file
MASKPATH=
```

In ensemble.card, it is possible to specify a periodic list of start dates.
Restart files will be generated for each member at each date starting from BEGIN_INIT to END_INIT with a periodicity of PERIODICITY.
The variable MEMBER sets the number of members for each start date.

The following part of ensemble.card sets 10 members from 19900101 to 20000101 every 2 years each lasting 10 years:

```
# member nb (i.e nb of perturb initial restart for each date)
MEMBER=10
...
# periodic and member list simulations length
LENGTH=10Y
# start date of the first ensemble
BEGIN_INIT=19900101
# start date of the last ensemble
END_INIT=20000101
# timestep between each periodic simulation
PERIODICITY=2Y
```

This will produce 10 members starting at the dates : 19900101, 19920101, 19940101, 19960101, 19980101, 20000101. (PERIODICITY can be given in months for shorter periods)

Each time the restart file to be perturbed in order to produce each member is taken from the previous day of the start date : 19893112, 19913112, etc...

The directory in which the start date is retrieved is given by INITPATH and INITFROM.
To restart from experiment v3h4BTxx in directory /ccc/store/cont003/gen2211/nguyens/IGCM_OUT/IPSLCM5A/PROD/historical fill:

```
# Restart name
INITFROM=v3h4BTxx
# Restart directory
INITPATH=/ccc/store/cont003/gen2211/nguyens/IGCM_OUT/IPSLCM5A/PROD/historical
```

The way the perturbed member is generated depends on PERTURB_BIN array. The first two elements are the most important. The first one is the executable to be used to produce the members, the second one is the component from which the restart is perturbed.

In the Periodic Case it is only possible to build the members by applying a randomly generated temperature pattern on the restart file of the coupler. PERTURB_BIN should look like this:

```
PERTURB_BIN=(AddNoise, CPL, sstoc, O_SSTSST, 0.1)
```

The list is interpreted as follows:

- the used executable is *AddNoise*,
- the component is the coupler (*CPL*),
- the restart file to perturb contains *sstoc* in its name,
- the variable to perturb in the restart file is *O_SSTSST*,
- the randomly generated perturbation is in [-.05;+0.05] degrees

!!NOTA!! The perturbation is not applied to grid points located under the sea ice. This condition is "hard-written" in the AddNoise code. Because of a change of the name of the sea ice cover variable from IPSL-CM5A (OIceFrac) and IPSL-CM6 (OIceFrc), a modification of the code has been made by Olivier Marti in June 2016 to allow the code to search for both names

For each member (in our example we have ten) a new restart file for the coupler will be generated using the executable *addnoise* to add some randomly generated temperature perturbation.
For the year 1990, the corresponding restart file of member 1 will be stored in

```
$WORKDIR/IGCM_IN/v3h4testB190/v3h4testB190A/CPL/Restart/
```

Non-Periodic start dates

For this type of perturbed ensembles the following variables are left empty:

```
# member list (apply list of pattern to initial state)
PERTU_MAP_LIST=()

# member list of names corresponding to each member
MEMBER_NAMESLIST=()
```

```
# member pattern global name
MEMBER_INITFROM=

# member pattern global directory for name
MEMBER_INITPATH=
...
# start dates list
NONPERIODIC=()
# length list for non periodic simulation (NOTE: use length above if not fill)
LENGTH_NONPERIODIC=()
...
# start date of the first ensemble
BEGIN_INIT=
# start date of the last ensemble
END_INIT=
...
# Path of Mask file
MASKPATH=
```

The variable LENGTH must be set to something but is not used, PERIODICITY must be set to NONE:

```
# periodic and member list simulations length
LENGTH=10Y
...
# timestep between each periodic simulation (NONE for nonperiodic)
PERIODICITY=NONE
```

To set 10 members for the starting dates 1990 and 1992 for a duration of 10 years, set MEMBER, NONPERIODIC and LENGTH_NONPERIODIC as follows:

```
# member nb (i.e nb of perturb initial restart for each date)
MEMBER=10
...
# start dates list
NONPERIODIC=(19900101 19920101)
# length list for non periodic simulation (NOTE: use length above if not fill)
LENGTH_NONPERIODIC=(10Y 10Y)
```

This results in 20 simulations in total.

The restart files to be perturbed to produce each member are sought in directory INITFROM which PATH is INITPATH.

```
# Restart name
INITFROM=v3h4BT00
# Restart directory
INITPATH=/ccc/store/cont003/gen0826/labetoul/dmf_import/IGCM_OUT/IPSLCM5A/PROD/historical
```

This will result in using restarts from experiment *v3h4BT00* located in directory
/ccc/store/cont003/gen0826/labetoul/dmf_import/IGCM_OUT/IPSLCM5A/PROD/historical.

The perturbation executable must be *AddNoise*.

```
PERTURB_BIN=(AddNoise, CPL, sstoc, O_SSTSST, 0.1)
```

List of members for a single start date

For this type of perturbed ensembles the following variables are left empty:

```
# member nb (i.e nb of perturb initial restart for each date)
MEMBER=
# timestep between each periodic simulation (NONE for nonperiodic)
PERIODICITY=NONE
# start dates list
NONPERIODIC=()
# length list for non periodic simulation (NOTE: use length above if not fill)
LENGTH_NONPERIODIC=()
```

It is important to leave PERIODICITY set to NONE and LENGTH_NONPERIODIC as an empty list: the list of member method only works for a single start date and neither with periodic start dates nor with non periodic start dates.

The variables BEGIN_INIT and END_INIT are set to the same date, only BEGIN_INIT will be used to provide the start date of the simulation for each member.

```
# start date of the first ensemble
BEGIN_INIT=20560101
# start date of the last ensemble
END_INIT=20560101
```

The variable LENGTH is the computation time which is the same for all members.

```
# periodic and member list simulations length
LENGTH=10Y
```

MEMBER_NAMESLIST is the list of names given to each member. It gives the names of the subdirectories from which the Job is submitted for each member as well as the subdirectories in which the results are stored for each member.

PERTU_MAP_LIST (previously named as *MEMBER_LIST*) is the list of perturbation maps files names prefix to apply to the restart file. It is implied that the files are named *prefix.nc*.

MEMBER_INITFROM is the directory in which the perturbations maps are stored.

MEMBER_INITPATH is the path to this directory.

```
# member list of names corresponding to each member
MEMBER_NAMESLIST=(OWN3DTA, OWN3DTB, OWN3DTC, OWN3DTD)

# member list (apply list of pattern to initial state)
PERTU_MAP_LIST=(OWN3DT_A, OWN3DT_B, OWN3DT_C, OWN3DT_D)

# member pattern global directory name
MEMBER_INITFROM=OWN3DTpf

# member pattern global directory for name
MEMBER_INITPATH=/ccc/work/cont003/gen2211/nguyens/PERTU/VECTORS
```

The variables INITFROM and INITPATH are still used to point to the directory where the restart files including the one to be perturbed are available.

```
# Restart name
INITFROM=piControl2

# Restart directory
INITPATH=/ccc/store/cont003/dsm/p86caub/dmf_import/IGCM_OUT/IPSLCM5A/PROD/piControl
```

For the member list perturbation type we use the executable AddPertu3DOCE and set PERTURB_BIN this way:

```
# perturbation type
PERTURB_BIN=(AddPertu3DOCE, OCE, restart, tn, ORCA2_mesh_mask.nc)
```

The elements of the list mean:

- the executable to be called to generate the perturbation is *AddPertu3DOCE*
- the component is the Ocean (*OCE*)
- the restart file to perturb is **restart*.nc*
- the field to perturb in the restart file is *tn*
- the meshmask file to tell if the gridcell is land or sea is *ORCA2_mesh_mask.nc*

The path to the mesh mask file is given in MASKPATH.

```
# Path of Mask file
MASKPATH=/ccc/cont003/home/gen2211/nguyens/addpertu
```

Once config.card and ensemble.card properly filled the directories containing the jobs to launch the simulations are created by issuing the command:

```
ins_job -e # Check and complet job's header
```