

The IPSLCM6.2 configuration is a part of the IPSLCM6 family model describes here

https://forge.ipsl.jussieu.fr/igcmg_doc/wiki/Doc/Config/IPSLCM6#IPSLCM6model

1. Resolutions and configurations

IPSLCM6.2 model will be available at different resolutions/configurations :

- **VLR** : LMDZ 96x95x39-ORCHIDEE - NEMO-LIM2-PISCES ORCA2xL31. **under development**
- **LR** : LMDZ 144x144x79-ORCHIDEE - NEMO-LIM3-PISCES eORCA1xL75
- **MR1** : LMDZ 256x256x79-ORCHIDEE - NEMO-LIM3-PISCES eORCA1xL75 **under development**
- **MR025** : LMDZ 256x256x79-ORCHIDEE - NEMO-LIM3 eORCA025xL75
- **ESMCO2** : LMDZ 144x144x79-ORCHIDEE - NEMO-LIM3-PISCES eORCA1xL75 with interactive carbon cycle ocean/atmosphere.
- **ESMAER** : LMDZ 144x144x79-ORCHIDEE-INCA - NEMO-LIM3-PISCES eORCA1xL75 with interactive aerosols on atmosphere.

The default configuration LR is exactly the same describes here https://forge.ipsl.jussieu.fr/igcmg_doc/wiki/Doc/Config/IPSLCM6_rc1#IPSLCM6-LR

IPSLCM6.2 is composed of following components and tools (Oct. 2019) :

```
#-H- IPSLCM6.2_work IPSLCM6.2_work coupled configuration
#-H- IPSLCM6.2_work This configuration is under construction
#-H- IPSLCM6.2_work NEMOGCM branch nemo_v3_6_STABLE revision 9455
#-H- IPSLCM6.2_work SHACONEMO revision 279
#-H- IPSLCM6.2_work XIOS trunk revision 1659
#-H- IPSLCM6.2_work IOIPSL src trunk 4432
#-H- IPSLCM6.2_work LMDZ6 trunk rev 3489
#-H- IPSLCM6.2_work ORCHIDEE version branches/ORCHIDEE_2_2/ORCHIDEE revision 6189
#-H- IPSLCM6.2_work OASIS3-MCT 2.0_branch rev 1818
#-H- IPSLCM6.2_work IPSLCM6.2 latest revision
#-H- IPSLCM6.2_work libIGCM trunk rev 1482
#-M- IPSLCM6.2_work arnaud.caubel@lsce.ipsl.fr
#-C- IPSLCM6.2_work IOIPSL/trunk 4432 8 IOIPSL modeles
#-C- IPSLCM6.2_work branches/ORCHIDEE_2_2/ORCHIDEE 6189 14 ORCHIDEE modeles
#-C- IPSLCM6.2_work branches/OASIS3-MCT_2.0_branch/oasis3-mct 1818 15 oasis3-mct .
#-C- IPSLCM6.2_work LMDZ6/trunk 3559 11 LMDZ modeles
#-C- IPSLCM6.2_work CONFIG/UNIFORM/v6/IPSLCM6.2 HEAD 8 IPSLCM6 config
#-C- IPSLCM6.2_work trunk/libIGCM 1488 10 libIGCM .
#-C- IPSLCM6.2_work branches/2015/nemo_v3_6_STABLE/NEMOGCM 9455 7 . modeles
#-C- IPSLCM6.2_work trunk/ORCA1_LIM3_PISCES 287 17 . modeles/NEMOGCM/CONF
#-C- IPSLCM6.2_work trunk/ORCA025_LIM3_PISCES 287 17 . modeles/NEMOGCM/CONF
#-C- IPSLCM6.2_work trunk/INCA6 873 9 INCA mode
#-C- IPSLCM6.2_work XIOS/trunk 1659 12 XIOS modeles
```

The default configuration LR is exactly the same describes here https://forge.ipsl.jussieu.fr/igcmg_doc/wiki/Doc/Config/IPSLCM6_rc1#IPSLCM6-LR.

Here is described how to use the existing **ESMCO2** configuration

1.0.1. How to use it

To retrieve the configuration :

```
mkdir YOUR_DIRECTORY ; cd YOUR_DIRECTORY
svn_anon # svn co http://forge.ipsl.jussieu.fr/igcmg/svn/modipsl/trunk modipsl
cd modipsl/util
./model IPSLCM6.2_work
cd ../config/IPSLCM6.2
```

The compilation slightly differs from what has been done so far. Now we used a script named **compile_ipslcm6.sh** with several option (resolution, level of optimisation, full or partial recompilation) to compile the model

```
#####
# Usage of the script compile_ipslcm6.sh
#
#####

./compile_ipslcm6.sh [Options]

Options: [LR / VLR / MR1 / MR025] Model resolution, choose only one. Default: LR.
        [ESMCO2] Compile IPSLCM6 for CO2 interactif ocean/atmosphere.
        [ESMAER] Compile IPSLCM6 for AER interactif on atmosphere
        [-full] Full recompilation of all components. This option can be added to all other options.
        [-cleannemo] Full recompilation of NEMO component only.
        [-debug / -dev / -prod] Level of optimization. One of these can be added to all other compile options. Default: -p

Example 1: Default compilation of IPSLCM6 for resolution LR
          (Resolution atmos: 144x142x79, ocean: ORCA1)
./compile_ipslcm6.sh

Example 2: Compilation of IPSLCM6 for resolution MR025
          (atmos: 256x256x79, ocean: ORCA025, NOPISCES)
./compile_ipslcm6.sh MR025

Example 3: Compilation of IPSLCM6 for ESM CO2
./compile_ipslcm6.sh ESMCO2 -cleannemo

Example 4: Default resoluition (LR) compiled in debug mode
./compile_ipslcm6.sh -debug

Example 5: Default compilation with full recompilation of all components. No clean is needed.
./compile_ipslcm6.sh -full

Example 6: Full recompilation of resolution MR05 in debug mode
./compile_ipslcm6.sh MR025 -debug -full
```

To compile and create the Job for **ESMCO2**

```
./compile_ipslcm6.sh ESMCO2

cp EXPERIMENTS/IPSLESM/CO2/piControl_TEST/config.card .
vi config.card # modify JobName (at least) : MYJOBNAME, restarts
../util/ins_job # Check and complete job's header

cd MYJOBNAME
vi Job_MYJOBNAME # modify PeriodNb, adjust the time, headers ...
sbatch Job_MYJOBNAME # IDRIS
ccc_msub Job_MYJOBNAME # TGCC
```

1.0.2. Restart files

Not available yet. Waiting for reference simulations.

1.0.3. Output level

By default, only **monthly outputs** and **low output levels** are activated.

1.0.4. Lengths, frequencies

1.0.4.1. Period length

Default period length is 1Y, i.e in config.card :

```
PeriodLength=1Y
```

Note that clean_PeriodLength.job will remove last period files, i.e last simulated year files.

1.0.4.2. Pack Frequency

Default pack frequency is 1Y, i.e in config.card :

```
PackFrequency=1Y
```

1.0.4.3. Rebuild frequency

Since we run with XIOS (server mode) as output library, **the rebuild step is not needed anymore.**

1.0.4.4. How to add a parameter in NEMO's namelist?

- let find the parameter in namelist_ref. For example in modeles/NEMOGCM/CONFIG/SHARED/namelist_ice_lim3_ref
- let find the namelist's name : for example &namicedyn
- let add a line with the new of the parameter in the file PARAM/namelist_lim3_ORCA1_cfg in the &namicedyn section

1.0.4.5. What is the maximum length for a simulation name?

Due to limitation in NEMO, a simulation should have **less than 39 characters.**

1.0.5. Computing centres

1.0.5.1. TGCC Bull Curie thin nodes

Default configuration on **598 cores** allows you to run **3 simulated years per day**. Because of load-balancing (difference between ocean computing time and atmosphere computing time), not all configurations (in terms of number of process/threads) are efficient. If you want to run a configuration with less cores, ask Arnaud Caubel what would be the optimum configuration.