Wikiprint Book

Title: 1. IDRIS

Subject: Igcmg_doc - Doc/ComputingCenters

Version: 38

Date: 06/26/24 16:13:58

Table of Content

Computing centers and environments

The supported machine types (also called computing environment) that can be used with the IPSL Climate Modeling Center tools and models are described in this chapter.

1. IDRIS

IDRIS environment

2. TGCC

TGCC environment

3. CINES

CINES environment (work in progress in september 2015).

4. LSCE

LSCE computing environment

5. IPSL

IPSL clusters CICLAD and ClimServ

6. Shared files

Shared files in synchronized repository IGCM?

7. How to use the IPSL models and tools with a local PC

7.1. Compiling

It is recommanded to first make a test installation of XIOS following the guide here: ■http://forge.ipsl.jussieu.fr/ioserver/wiki/documentation Then use the same libraires and compiler versions for all components as for XIOS.

Some configurations can be installed without XIOS.

7.1.1. Choose a target in AA_make.gdef

The ins_make script from modipsl will probably not recognize your local PC to create makefiles. You must choose an existing or create a new target in the util/AA_make.gdef file. Adapt this target to your computing environment (compiler, options, path to netcdf, etc), and create a new makefile specifying this target:

```
./ins_make -t new_target
```

Please refer to the model managers of your model configuration to learn more about how to compile the IPSL models on a local PC and which compilers can be used with the specific models you need.

7.1.2. Example for installing ORCHIDEE offline

Before installing ORCHIDEE, make a test installation of XIOS, see guide here: ■http://forge.ipsl.jussieu.fr/ioserver/wiki/documentation . If this fails, it is still possible to install ORCHIDEE without XIOS, see notes below.

```
# 1. Install a new modipsl
svn co http://forge.ipsl.jussieu.fr/igcmg/svn/modipsl/trunk modipsl


# 2. Install the configuration ORCHIDEE_trunk
cd modipsl/util
./model ORCHIDEE_trunk


# 3. Adapt and add compile options to your machine

# 3a. Compile options for component IOIPSL:
# Add a section NEW_ARCH in modipsl/util/AA_make.gdef
# Do this by coping the section ifort_CICLAD for example and adapt all options according to your machine and your compiler
# Make sur to have the variable FCM_ARCH correct as follow:
#-Q- NEW_ARCH FCM_ARCH = NEW_ARCH

# 3b. Add compile options for component XIOS
# Add following files (by coping and adapting one of the existing targets closest to your machine)
modipsl/modeles/XIOS/arch/arch_NEW_ARCH.fcm
modipsl/modeles/XIOS/arch/arch_NEW_ARCH.path
modipsl/modeles/XIOS/arch/arch_NEW_ARCH.env

# 3b. Add compile options for component ORCHIDEE
# Add following files (by coping and adapting one of the existing targets closest to your machine)
modipsl/modeles/ORCHIDEE/arch/arch_NEW_ARCH.fcm
modipsl/modeles/ORCHIDEE/arch/arch_NEW_ARCH.path
modipsl/modeles/ORCHIDEE/arch/arch_NEW_ARCH.env


# 4. Create makefiles with target NEW_ARCH
cd modipsl/util
./ins_make -t NEW_ARCH


# 5. Compile
cd modipsl/config/ORCHIDEE_OL
gmake
```

Notes:

- The name NEW_ARCH can be changed to another name but needs to be same everywhere
- All arch_NEW_ARCH.* files and AA_make.gdef must be coherent for the libraries and options choosen
- It is still possible to install without XIOS. For that case, you do not need to do 3b above and the compilation is done using **gmake without_xios**

7.1.3. Example for installing LMDZOR_v5.2 sequential mode with gfortran

```
# 1. Install configuration LMDZOR_v5.2 in a new modipsl
svn co http://forge.ipsl.jussieu.fr/igcmg/svn/modipsl/trunk modipsl
cd modipsl/util; ./model LMDZOR_v5.2

# 2. Adapt the path to your netcdf which must also be compiled with gfortran, in 3 files:
modipsl/modeles/LMDZ/arch/arch-gfortran.path
modipsl/modeles/ORCHIDEE/arch/arch-gfortran.path
modipsl/util/AA_make.gdef (section gfortran)

# 3. Change default compiling to sequential run mode in main makefile.
In modipsl/config/LMDZOR_v5.2/AA_make
```

```
change "-parallel mpi" into "-parallel none" at 3 places.
The name of the executables changes also, therefore change "_phylmd_para_orch.e" into "_phylmd_seq_orch.e" at 2 places

# 4. Recreate makefiles with target gfortran
cd modipsl/util
./ins_make -t gfortran

# 5. Compile as usual
cd modipsl/config/LMDZOR_v5.2; gmake
```

Note 1: for installing in parallel mode with MPI, do as above but adapt the files arch-gfortran.path, arch-gfortran.fcm and AA_make.gef with suitable compile options. You can use target gfortran_CICLAD in AA_make.gdef as example. Do not do point 3 above.

Note 2: for older version of ORCHIDEE, such as in LMDZOR_v5, the files in modeles/ORCHIDEE/arch do not exist. Compile options in AA_make.gdef is used for ORCHIDEE in this case.

The v6 configurations cannot be compiled in sequential mode because XIOS do not compile or run without the MPI library.

7.2. Simulation with libIGCM

When using libIGCM on a local PC, the parameters of the default system described in the libIGCM_sys/libIGCM_sys_default.ksh file will be used. You may have to change this file to match your system.

A minimum subset of files located on the shared IGCM account must be downloaded and installed. This directory is called R_IN in libIGCM_sys_default.ksh and it has the default path /home/${LOGIN}/IGCM.

7.3. Installing rebuild

The output of old version of the models (ORCHIDEE, LMDZ, INCA, REPROBUS) must be recombined to the total horizontal domain when simulation is done in parallel mode.

The tool for this is called *rebuild*. The rebuild tool is called in the post-processing phase by libIGCM. rebuild is a fortran code included in IOIPSL. rebuild must be installed and compiled on the local machine. The rebuild can then be used interactively outside libIGCM or in the post-processing phase.

Installation

```
cd modipsl/util
./model IOIPSL_PLUS
# Modify AA_make.gdef for the compiling as above
./ins_make -t new_target
cd ../modeles/IOIPSL/tools
gmake
```

For use with libIGCM, add the path to your rebuild in libIGCM_sys/libIGCM_sys_default.ksh.

Use in interactive mode

rebuild can also be used interactively. For example create the global file sechiba_history.nc as following :

```
./rebuild -h
./rebuild -o sechiba_history.nc sechiba_history_00*
```