

Wikiprint Book

Title: 1. Compiling environment

Subject: Igcmg_doc - Doc/Compile

Version: 44

Date: 06/29/24 09:22:21

Table of Content

Compile	3
1. Compiling environment	3
1.1. FCM	3
2. Compilation for configurations _v6 and _v6.1.*	3
2.1. Creating the main makefile	3
2.2. The main makefile	4
2.3. How to compile?	5
3. How to change compile options for debugging or optimization of the model?	6
4. How to compile for hybrid MPI-OpenMP run mode?	6

Compile

The compiling steps are described in this chapter.

1. Compiling environment

First of all, make sure that your working environment contains all the necessary compiling tools. To this end, see the chapter [Computing environments](#)

1.1. FCM

Most of IPSL models and components use the **fcm** tool for compiling. Recent versions of the models have fcm in their source directory. Therefore no specific installation is needed anymore.

If no fcm is available in the source directory, fcm must then be installed and added to your environment variable PATH. At following machines, fcm is already installed

```
# At TGCC:
PATH=`ccc_home -u igcmg`/Tools/fcm/bin:$PATH

# At IDRIS:
PATH=

# At LSCE(obelix):
PATH=/home/orchideeshare/igcmg/fcm/FCM_V1.2/bin:$PATH
```

For other machines, extract fcm from svn and add the path to your environment variable PATH. Extract as follow:

```
svn co http://forge.ipsl.jussieu.fr/fcm/svn/PATCHED/FCM_V1.2 fcm
```

For each machine supported by the IPSL tools, 3 kinds of files ***.fcm**, ***.path**, and **.env** exist. They provide all the information needed by fcm to create a makefile. These files are stored in the directory arch/ (LMDZ/arch/, INCA/arch/).

- *.fcm : contains the compiler information
 - compiler name
 - compiling options
 - options and settings
- *.path : contains the library path names.
- *.env : contains module to be load to install properly the environment for the compilation of this model.

These files have the name of the machine as a prefix : arch-X64_IRENE.fcm, arch-X64_IRENE.path

Note : for configurations IPSLCM, arch files for OASIS and NEMO are taken in modipsl/config/IPSLCM*/SOURCE/OASIS and modipsl/config/IPSLCM6/SOURCE/NEMO

2. Compilation for configurations _v6 and _v6.1.*

2.1. Creating the main makefile

Note : The compilation of NEMO forced configuration requires a few special steps. Read more for [NEMO](#).

Each model configuration XXX has a main makefile which provides information about each model component's makefiles. The main Makefile is stored in the modipsl/config/XXXX directory. The main Makefile is created when a model configuration is downloaded and installed.

Example :

```
cd modipsl/util
./model LMDZOR_v6.1.10
```

```
(...)
Makefiles setup, scripts and data for irene

Installation in ../config/LMDZOR_v6
Path from Makefile to modipsl/util : ../../util
```

You can recreate those Makefiles by running the `ins_make` script. You have to do this if, for example, you changed the path of modipsl.

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

The main Makefile is created by concatenating the following 3 items: `AA_make.ideal` and `AA_make` located in the directory where the main Makefile will be created, and the lines corresponding explicitly to the target machine of the `util/AA_make.gdef` file. For each predefined machine the `util/AA_make.gdef` file contains the compiler settings. If your machine is not included, you can choose a target among the predefined machines in `AA_make.gdef` or you can add a new one.

```
vi AA_make.gdef

(...)
#-Q- jeanzay
(...)
#-Q- irene
(...)
#-Q- aix6
(...)
#-Q- gfortran
(...)
```

In this case, the script becomes

```
./ins_make -t target
```

2.2. The main makefile

The Makefile is available for different resolutions. Example with LMDZOR_v6.1.10 :

```
LMD5655 : libioipsl xios liborchidee lmdz56x55x19 verif
        echo "noORCAxLMD5655" >.resol
        echo "RESOL_ATM_3D=56x55x19" >>.resol

LMD9671 : libioipsl xios liborchidee lmdz96x71x19 verif
        echo "noORCAxLMD9671" >.resol
        echo "RESOL_ATM_3D=96x71x19" >>.resol

LMD9695 : libioipsl xios liborchidee lmdz96x95x19 verif
        echo "noORCAxLMD9695" >.resol
        echo "RESOL_ATM_3D=96x95x19" >>.resol

LMD9695-L39 : libioipsl xios liborchidee lmdz96x95x39 verif
        echo "noORCAxLMD9695-L39" >.resol
        echo "RESOL_ATM_3D=96x95x39" >>.resol

LMD9695-L79 : libioipsl xios liborchidee lmdz96x95x79 verif
        echo "noORCAxLMD9695-L79" >.resol
        echo "RESOL_ATM_3D=96x95x79" >>.resol

LMD128118-L39 : libioipsl xios liborchidee lmdz128x118x39 verif
        echo "noORCAxLMD128118-L39" >.resol
```

```

    echo "RESOL_ATM_3D=128x118x39" >>.resol

LMD144142 : libioipsl xios liborchidee lmdz144x142x19 verif
    echo "noORCAxLMD144142" >.resol
    echo "RESOL_ATM_3D=144x142x19" >>.resol

LMD144142-L39 : libioipsl xios liborchidee lmdz144x142x39 verif
    echo "noORCAxLMD144142-L39" >.resol
    echo "RESOL_ATM_3D=144x142x39" >>.resol

LMD144142-L59 : libioipsl xios liborchidee lmdz144x142x59 verif
    echo "noORCAxLMD144142-L59" >.resol
    echo "RESOL_ATM_3D=144x142x59" >>.resol

LMD144142-L79 : libioipsl xios liborchidee lmdz144x142x79 verif
    echo "noORCAxLMD144142-L79" >.resol
    echo "RESOL_ATM_3D=144x142x79" >>.resol

LMD512360-L79 : libioipsl xios liborchidee lmdz512x360x79 verif
    echo "noORCAxLMD512360-L79" >.resol
    echo "RESOL_ATM_3D=512x360x79" >>.resol

```

Available resolutions are :

- LMD5655
- LMD9671
- LMD9695
- LMD9695-L39
- LMD9695-L79
- LMD128118-L39
- LMD144142
- LMD144142-L39
- LMD144142-L59
- LMD144142-L79
- LMD512360-L79

2.3. How to compile?

Once you chose a resolution you can start compiling:

```
gmake my_resolution
```

Example for LMDZOR_v6.1.10:

```
cd modipsl/config/LMDZOR_v6
gmake LMD144142-L79
```

An hidden file `.resol` is created when compiling was successful. This file contains information about the resolution you have just compiled. This file will be used later when setting up the simulation, in particular to locate the input files.

Going back to the previous example, the `.resol` file looks as follows :

```
noORCAxLMD144142-L79
RESOL_ATM_3D=144x142x79
```

Once the file `.resol` is created, you can recompile your configuration using the `gmake` script without specifying the resolution.

For ORCHIDEE_OL, the resolution is not specified during run time. Compiler using only **gmake**. The file `.resol` will not be created.

3. How to change compile options for debugging or optimization of the model?

- You can change your optimization options for IOIPSL models in the AA_make.gdef file. To this end you must find your target machine and change the associated lines. You will have to recreate the Makefile.
- Three levels of optimization for the INCA and LMDZ models are predefined in the arch/ files. You can select them in the modipsl/config/.../Makefile file. The three options are: **-debug**, **-dev**, **-prod**. XIOS offers the same three levels of compilations, accessible through the flags `--debug`, `--dev` and `prod`.

Example: to compile LMDZ you can add the keyword "-debug" :

```
(cd ../../modeles/LMDZ; ./make_lmdz_fcm (...) -debug (...) -arch $(FCM_ARCH) gcm ; cp bin/gcm_$(RESOL_LMDZ)_phylmd_para_orc
```

The default option is `-prod`

- For ORCHIDEE in all `_v5` configurations, the compile options are also set in AA_make.gdef file, change as for IOIPSL. For LMDZOR_v5.2 configuration which is using the trunk of ORCHIDEE, the compile options are set in models/ORCHIDEE/arch. Switch between `-prod`, `-dev` or `-debug` as for LMDZ in the main makefile.

4. How to compile for hybrid MPI-OpenMP run mode?

This is possible only for configuration LMDZOR_v5.2 at curie or ada. In that case you must change the main makefile and recompile:

```
cd modipsl/config/LMDZOR_v5.2
# Change in AA_make all 3 occurrences for "-parallel mpi" into "-parallel mpi_omp"
../../util/ins_make
gmake clean
gmake
```