

ORCHIDEE Training course

Code management, installation, simulation,
documentation

16-17 January 2020, IDRIS

Josefine Ghattas IPSL



ORCHIDEE
LAND SURFACE MODEL

SVN

Versions and related configurations

Coding Guidelines

Install

Different platforms

Compilation

Experiences with libIGCM

Configure input and output files

Finding information

SVN

See presentation...

Versions and related configurations with ORCHIDEE

Lastest version of the trunk ORCHIDEE

Use with configuration **ORCHIDEE_trunk** and **LMDZOR_v6.3_work**
ORCHIDEE_trunk offline configuration contains the latest version of the trunk.
For new developments this is often the version to use. Note: The trunk changes often : if you extract today and again next week there might be differences in the source code due to evolution of the trunk

Tag ORCHIDEE_2_0

Use with configuration **ORCHIDEE_2_0**, **LMDZOR_v6.1.x** and **IPSLCM6.1.x-LR**

Used for reference simulations for CMIP6.

Tag ORCHIDEE_2_2

Use with configuration **ORCHIDEE_2_2** or **LMDZOR_v6.2_work**

Newest tag of ORCHIDEE, corresponds to the current version of the trunk.

Close to ORCHIDEE_2_0 with some corrections and enhancements, includes possibility to be used with DYNAMICO.

Versions and related configurations with ORCHIDEE

Branches and other versions:

ORCHIDEE-MICT

ORCHIDEE-CN-CAN

ORCHIDEE-CNP

...

Contact the developers of each branch to know about corresponding configurations.

Coding Guidelines

All new developments in the ORCHIDEE trunk must follow the Coding Guidelines.

- **Comments in english**
- **Indentation**
- **Key words in capital letters**
- **Contain a description part in each module and subroutine**
- ...

<http://forge.ipsl.jussieu.fr/orchidee/wiki/Documentation/UserGuide/CodingGuidelines>

Use module `diffuco.f90` as example

How to install using modipsl

See hands on exercises

- modipsl is a tool used to install and compile predefined configurations, for example ORCHIDEE offline or ORCHIDEE coupled to LMDZ
 - modipsl contains scripts for extraction of predefined configurations, creation of makefiles, creation of job and some more. modipsl is also a empty file tree that will receive the models and tools.
 - use ***./model config*** to download a specific configuration
- ```
> svn co http://forge.ipsl.jussieu.fr/igcmg/svn/modipsl/trunk modipsl
> cd modipsl/util
> ./model -h # list predefined configurations
> ./model config # extract a predefined configuration
```

# Install a branch or personal version

You can modify the version of the ORCHIDEE source before extraction of a configuration. In [modipsl/util/mod.def](#), modify line:

```
#-C- ORCHIDEE_trunk trunk/ORCHIDEE HEAD 14 ORCHIDEE modeles
```

into

```
#-C- ORCHIDEE_trunk trunk/ORCHIDEE 2724 14 ORCHIDEE modeles
```

or

```
#-C- ORCHIDEE_trunk branches/xxx/yyy HEAD 14 ORCHIDEE modeles
```

For exemple:

```
#-C- ORCHIDEE_trunk branches/ORCHIDEE-MICT/ORCHIDEE HEAD 14 ORCHIDEE modeles
```



# Different platforms

---

- Compiling options of ORCHIDEE is preconfigured at following platforms :
  - obelix** at LSCE
  - irene** at TGCC
  - jeanzay** at IDRIS
  - ciclad and climserv** at IPSL
- Compiling at other machines need more time for installing pre-request (compiler, netcdf,..)

# TGCC and IDRIS

---

To work on the TGCC or IDRIS computing centres you need:

- **a login connected to an existing group/project**, discuss with your supervisor/colleagues
- **an allocation of computing resources** in this group, yearly demand, discuss with your supervisor/colleagues
- **knowledge about the environment** at these centres : different machines, file systems, etc..

# Obelix / LSCE

ORCHIDEE offline and LMDZOR\_v6 configurations can be used at obelix.

- The Job PBS directive for the core distribution needs to be changed if using more than 8 MPI with libIGCM. For example if using 32MPI, the headers will be  
**#PBS -l nodes=1:ppn=32**  
but needs to be changed to  
**#PBS -l nodes=4:ppn=8**
- Using libIGCM, archive is done on scratch filesystem :  
/home/scratch01/login/IGCM\_OUT . **Change by adding**  
**ARCHIVE=/disk/where/you/have/space in config.card**
- When running LMDZ at obelix:
  - change in run.def to: **use\_filtre\_fft=n**
  - Hybrid MPI-OpenMP mode has not been tested.

# ciclad and climserv / IPSL

=> Contact orchidee-help if you want to run at ciclad/climserv

ORCHIDEE\_trunk and LMDZOR\_v6 can be used in standard configurations with modipsl and libIGCM on ciclad and climserv. See specific information here :

[http://forge.ipsl.jussieu.fr/igcmg\\_doc/wiki/Doc/ComputingCenters/ESPRImesocenter](http://forge.ipsl.jussieu.fr/igcmg_doc/wiki/Doc/ComputingCenters/ESPRImesocenter)

- Sequential netcdf library is used as default. This is done using argument **--netcdf\_lib netcdf4\_seq** for the compilation of XIOS. Using sequential netcdf makes it impossible to use more than one server XIOS with one\_file mode.
- Change PBS directive recommended as for obelix, for 32 MPI set **#PBS -l nodes=4:ppn=8**

# Steps to follow for compilation at a new platform

---

- 1. Install modipsl and the configuration ORCHIDEE\_trunk
- 2. Modify compile options in following files:
  - modipsl/util/**AA\_make.gdef** (used for compilation of IOIPSL)
  - modipsl/modeles/**XIOS/arch/arch/arch-yourtarget.[fcm/path/env]**
  - modipsl/modeles/**ORCHIDEE/arch/arch/arch-yourtarget.[fcm/path]**

*Note: the variable FCM\_ARCH in AA\_make.gdef is the name of the arch files in ORCHIDEE/arch and XIOS/arch.*

- 3. Recreate makefiles with target chosen above and compile as usual  
cd modipsl/util; ./ins\_make -t yourtarget

=> Requirements are MPI and netCDF4 library.

Additional requirements: parallel library NetCDF4/HDF5

=> It is possible to compile and use without XIOS and without MPI.

# Main makefile for ORCHIDEE offline

## config/ORCHIDEE\_OL/Makefile

```
Main targets
#####
Default method : Compiling ORCHIDEE library and offline executables with XIOS
all : with_xios

without_xios : Compiling ORCHIDEE and IOIPSL. Do not compile or link to XIOS.
without_xios : ioipsl driver verif

with_xios : Compiling ORCHIDEE, XIOS and IOIPSL.
with_xios : xios ioipsl driver_xios verif

clean : Remove everything created during compilation including the executables
clean :
 (cd ../../modeles/IOIPSL/src ; ${M_K} clean ;)
 (cd ../../modeles/ORCHIDEE ; ./makeorchidee_fcm -clean ;)
 (rm -rf ../../modeles/XIOS/bin/ ../../modeles/XIOS/inc ../../modeles/XIOS/obj ../../modeles/XIOS/lib ;)
 (rm -f ../../bin/orchidee_ol ; rm -f ../../bin/teststomate ; rm -f ../../bin/forcesoil ; rm -f ../../bin/xios_server.exe)

Internal targets
#####
xios : Only compiling XIOS (using fcm)
xios :
 (cd ../../modeles/XIOS ; ./make_xios \
 --prod --arch ${FCM_ARCH} --job 8 ; cp bin/xios_server.exe ../../bin/. ;)

ioipsl : Only compiling IOIPSL (standard Makefile)
ioipsl :
 (cd ../../modeles/IOIPSL/src ; ${M_K} ;)

driver : Only compiling ORCHIDEE without linking to XIOS (with fcm method)
driver :
 (cd ../../modeles/ORCHIDEE ; ./makeorchidee_fcm -j 8 -parallel mpi -prod -arch ${FCM_ARCH} -driver ;)

driver_xios : Only compiling ORCHIDEE and linking to XIOS 1.0 (with fcm method)
Note: For linking with XIOS 2.0 change -xios into -xios2
driver_xios :
 (cd ../../modeles/ORCHIDEE ; ./makeorchidee_fcm -xios2 -j 8 -parallel mpi -prod -arch ${FCM_ARCH} -driver ;)

verif : List contents in executable directory
verif: ../../bin
 ls -lrt ../../bin
```

# Main makefile for ORCHIDEE offline

## config/ORCHIDEE\_OL/Makefile

```
Main targets
#####
Default method : Compiling ORCHIDEE library and offline executables with XIOS
all : with_xios

without_xios : Compiling ORCHIDEE and IOIPSL. Do not compile or link to XIOS.
without_xios : ioipsl driver verif

with_xios : Compiling ORCHIDEE, XIOS and IOIPSL.
with_xios : xios ioipsl driver_xios verif

clean : Remove everything created during compilation including the executables
clean :
 (cd ../../modeles/IOIPSL/src ; ${M_K} clean ;)
 (cd ../../modeles/ORCHIDEE ; ./makeorchidee_fcm -clean ;)
 (rm -rf ../../modeles/XIOS/bin/ ../../modeles/XIOS/inc ../../modeles/XIOS/obj ../../modeles/XIOS/lib ;)
 (rm -f ../../bin/orchidee_ol ; rm -f ../../bin/teststomate ; rm -f ../../bin/forcesoil ; rm -f ../../bin/xios_server.exe)

Internal targets
#####
xios : Only compiling XIOS (using fcm)
xios :
 (cd ../../modeles/XIOS ; ./make_xios \
 --prod --arch ${FCM_ARCH} --job 8 ; cp bin/xios_server.exe ../../bin/. ;)

ioipsl : Only compiling IOIPSL (standard Makefile)
ioipsl :
 (cd ../../modeles/IOIPSL/src ; ${M_K} ;)

driver : Only compiling ORCHIDEE without linking to XIOS (with fcm method)
driver :
 (cd ../../modeles/ORCHIDEE ; ./makeorchidee_fcm -j 8 -parallel mpi -prod -arch ${FCM_ARCH} -driver ;)

driver_xios : Only compiling ORCHIDEE with XIOS (with fcm method)
driver_xios :
 (cd ../../modeles/ORCHIDEE ; ./makeorchidee_fcm -j 8 -parallel mpi -prod -arch ${FCM_ARCH} -driver_xios ;)

verif : List contents in executable directory
verif: ../../bin
 ls -lrt ../../bin
```

**./makeorchidee\_fcm -xios2 -j 8 -parallel mpi -prod -arch \${FCM\_ARCH} -driver**

← -prod or -debug  
← Suffix for file with compile options

# Compiling ORCHIDEE

---

- The main Makefile launch compilation of XIOS, IOIPSL and finally ORCHIDEE
- Inside the main Makefile, the script **makeorchidee\_fcm** is **launched to compile ORCHIDEE**. This compile script is based on the tool FCM.
- **Dependencies between modules are determined automatically.** No modifications are needed if you add a module in one of the existing src\_ directories.
- **Specific platform dependent compile options are set in modipsl/modeles/ORCHIDEE/arch/** directory: 2 files per platform: arch-ifort\_LSCE.fcm and arch-ifort\_LSC.path.
- **Change -prod into -debug** to use debugging options



# Offline experiences using libIGCM

---

## **OOL\_SEC\_STO\***

Experiment set up with sechiba and stomate, land use change activated

## **OOL\_SEC**

Experiment set up with sechiba only, lai file read

## **SPINUP\_ANALYTIC\_FG1**

Experiment set up with sechiba, stomate and spinup\_analytic activate. In this experiment, the forcing is set to loop over 10 years.

## **SPINUP and ENSEMBLE**

More complex experiences not taught here...

## **FORCESOIL**

Specific experiment to be used for spinup of the soil carbon pools, under development

# Modify parameters with libIGCM

In PARAM/run.def and in ORCHIDEE/src\_xml some parameters are modified by orchidee\_ol.driver, sechiba.driver and stomate.driver. These parameters are always marked:

**AUTO** : These parameters can be changed using options in *comp.card* or *config.card*. You can also change them directly in the run.def, for this case the drivers will not change them again.

**AUTOBLOCKER** : The job will stop if you modify these parameters. They are set by the *comp.driver* mainly using the information from config.card.

For example, in PARAM/run.def:

```
STOMATE_RESTART_FILEIN = _AUTOBLOCKER_
XIOS_ORCHIDEE_OK = _AUTO_
```

=> You can add or modify parameters directly in PARAM/run.def

# Configure input parameters

Input parameters are set in the file run.def.

Parameters are read in ORCHIDEE using:

**CALL getin\_p(“VARNAME”,var)**

```
SUBROUTINE slowproc_xios_initialize

 CHARACTER(LEN=255) :: filename
 LOGICAL :: lerr
 REAL(r_std) :: slope_noreinf
 LOGICAL :: get_slope
 INTEGER :: l

 IF (printlev>=3) WRITE(numout,*) 'In slowproc_xios_initialize'
 !! 1. Prepare for reading of soils_param file
 ! Get the file name from run.def file and set file attributes accordingly

 filename = 'soils_param.nc'
 CALL getin_p('SOILCLASS_FILE',filename)
 .
 .
 .
```

# Configure output files

---

**If time, presentation about XIOS and how to configure output files...**

# Finding information @listes.ipsl.fr

All ORCHIDEE user's are invited to subscribe to the email lists:  
**orchidee-dev** Discussion and information about ORCHIDEE  
**platform-users** Ask and answer questions about libIGCM  
Information about IPSL-cmc tools

2 email addresses for contact:

**orchidee-help** For technical questions

**orchidee-projet** To contact the ORCHIDEE project team

See how to subscribe :

<http://forge.ipsl.jussieu.fr/orchidee/wiki/GroupActivities/Contact>

# Finding information

## Wiki and web site

---

**ORCHIDEE official web site** (update once a year)

<http://orchidee.ipsl.fr>

**ORCHIDEE wiki** (update frequently)

On the wiki you find useful information about on-going developments and help to use the model. Technical information in Documentation/UserGuide

<http://forge.ipsl.jussieu.fr/orchidee/wiki>

You need a **“login forge” to write on the wiki**. This login is also needed to see the full content of the wiki and also to see the SVN repository on the web interface. Write to orchidee-help to get a login.

# Finding information

## “Developer's meeting”

---

**All users and developers are welcome to ORCHIDEE developer's meetings** organized several times during the year. These meetings consist in a presentation of a specific topic followed by discussions and questions. Meeting place at Jussieu/Paris or LSCE/Saclay but often a videoconference is set up.

See reports and presentations here :

<http://forge.ipsl.jussieu.fr/orchidee/wiki/GroupActivities/Meetings>

Information about these meetings are done at orchidee-dev email list.