

# ORCHIDEE Training course

Code management, installation, simulation,  
documentation

25-26 January 2022, Jussieu

Josefine Ghattas, IPSL



ORCHIDEE  
LAND SURFACE MODEL

**ORCHIDEE trunk and branches**

**Related configurations**

**Coding Guidelines**

**Install**

**Different platforms**

**Compilation**

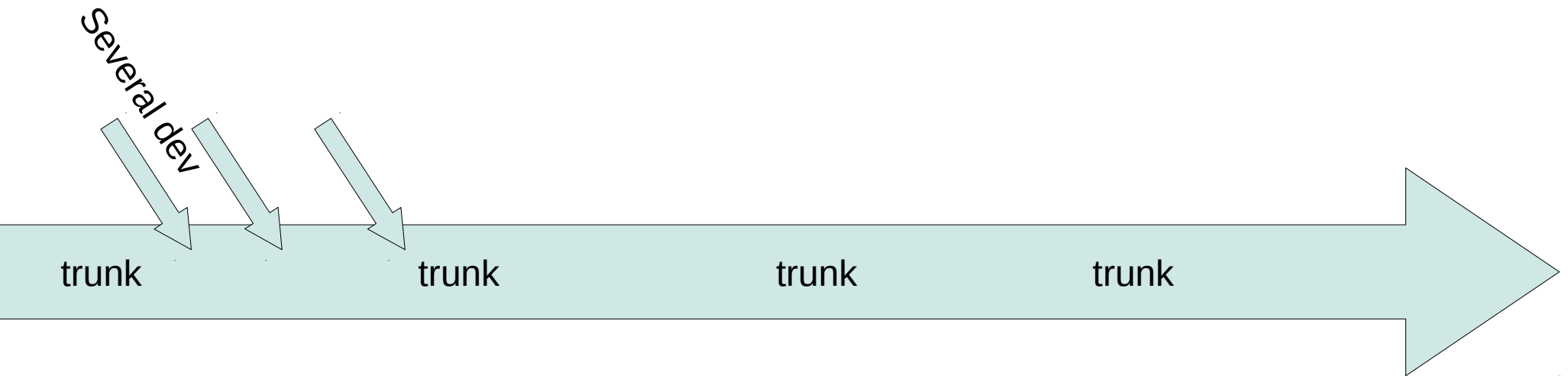
**Configure input and output files**

**Experiences with libIGCM**

**Finding information**

# ORCHIDEE trunk

= “main ORCHIDEE version evolving over time”

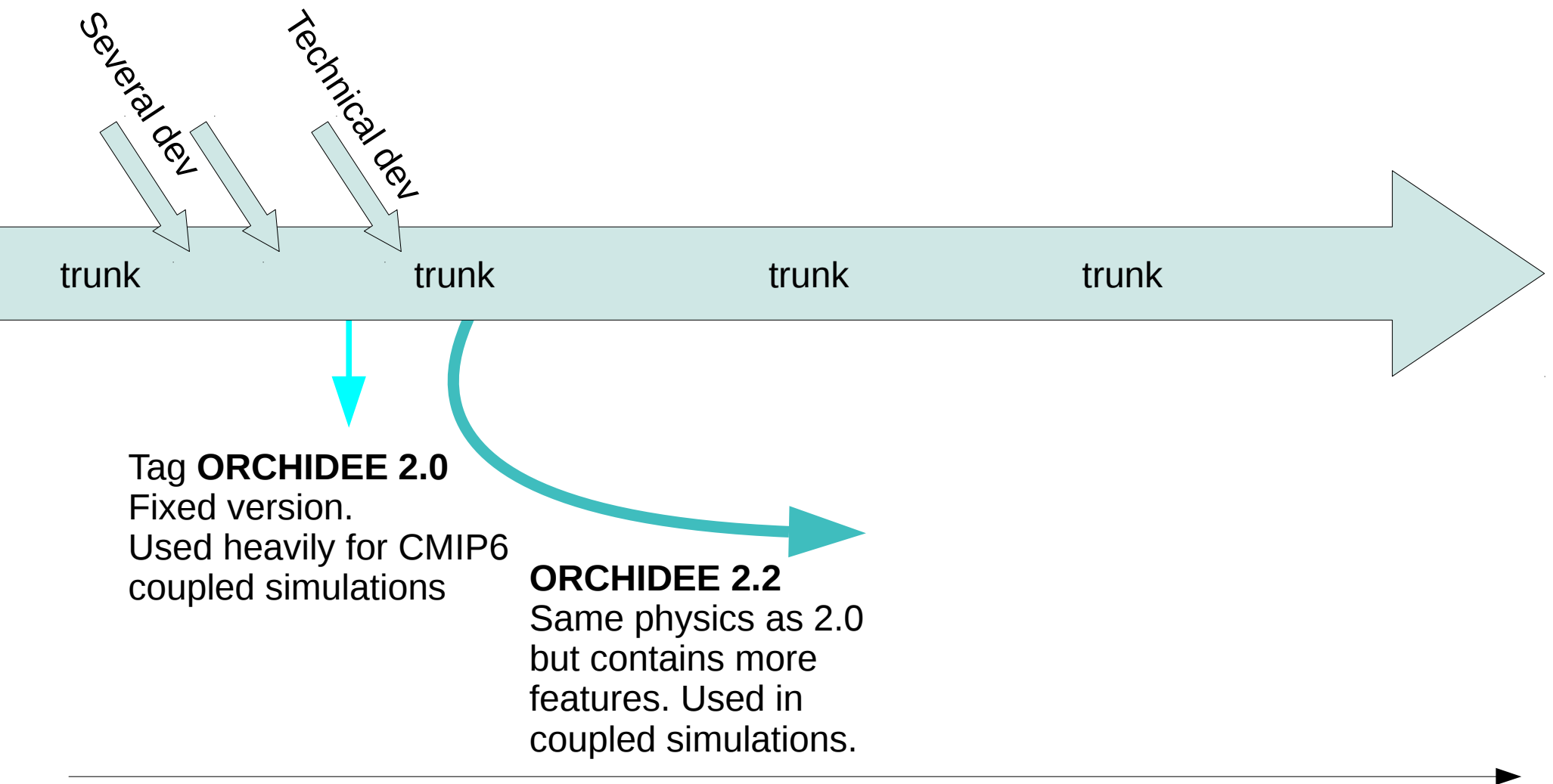


- The trunk is changing over time
- The trunk today and last week might not be the same
- You need to specify which specific revision of the trunk you use

time

# ORCHIDEE trunk

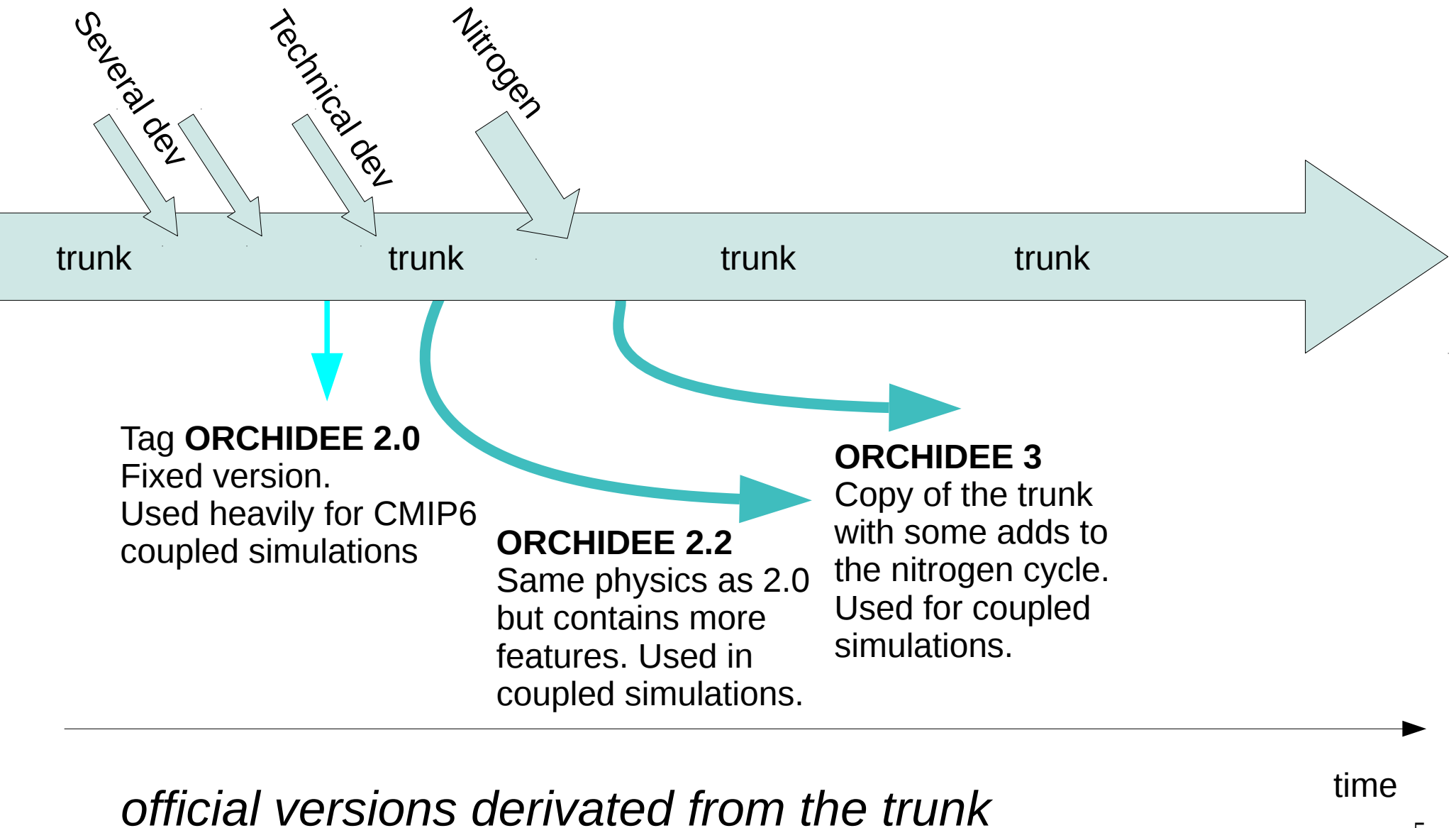
= “main ORCHIDEE version evolving over time”



*official versions derivated from the trunk*

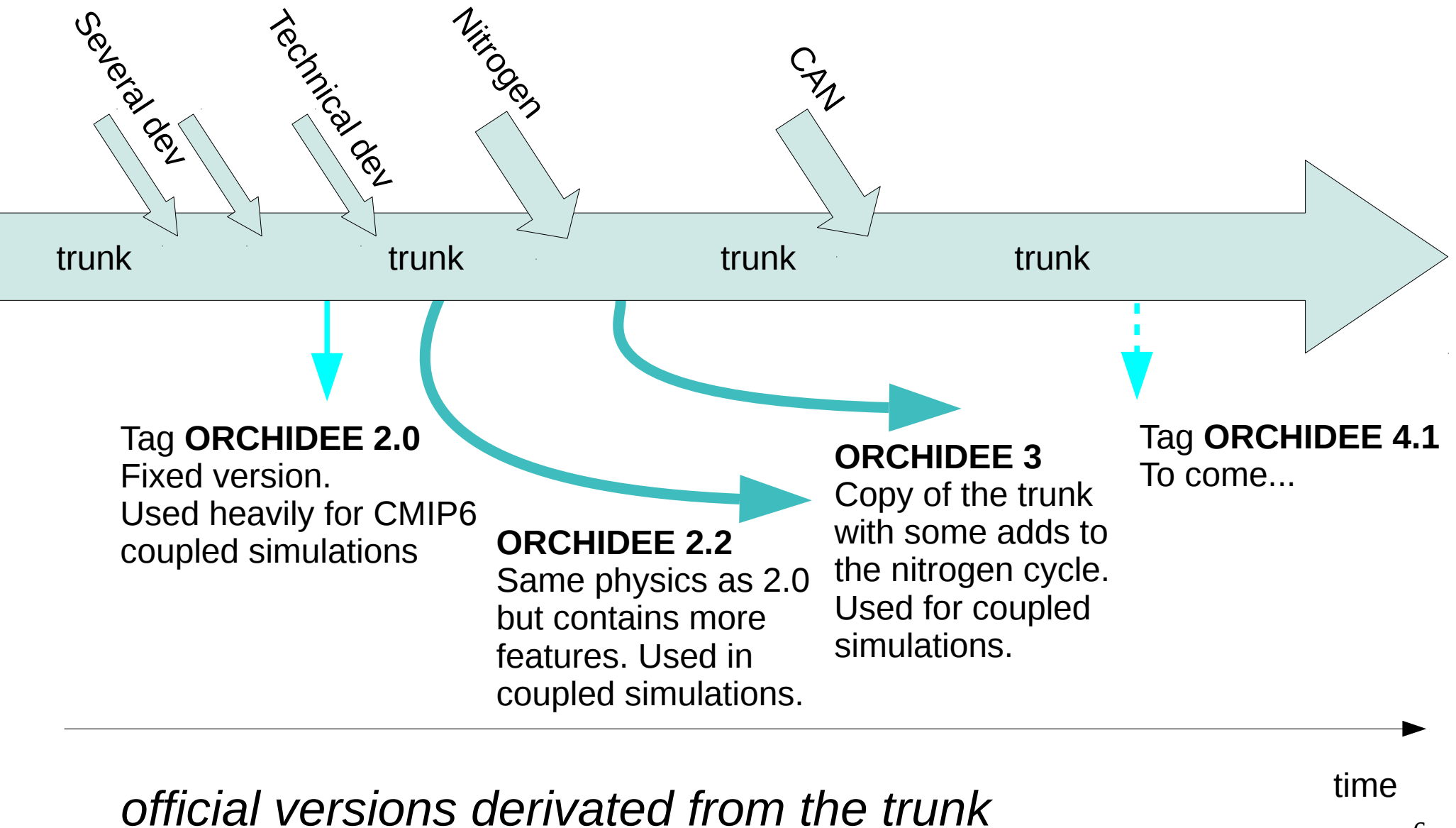
# ORCHIDEE trunk

= “main ORCHIDEE version evolving over time”



# ORCHIDEE trunk

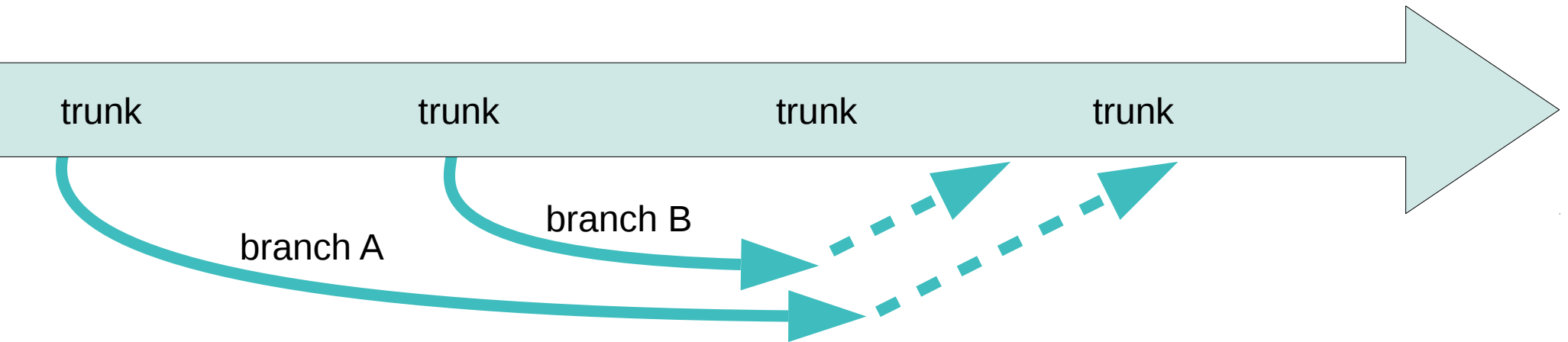
= “main ORCHIDEE version evolving over time”



# ORCHIDEE branches

= “other versions of ORCHIDEE not yet in the trunk”

- A branch can be seen as a “temporary” version of the model used during the development and validation phase.
- A branch starts as a copy of the trunk and then the new developments are added. When the developments are finished and validated, the branch should be integrated in the trunk.



# ORCHIDEE branches

= “other versions of ORCHIDEE not yet in the trunk”

---

- A branch can be seen as a “temporary” version of the model used during the development and validation phase.
- A branch starts as a copy of the trunk and then the new developments are added. When the developments are finished and validated, the branch should be integrated in the trunk.
- A branch is created when several people work together on a new development.
- When only one person work on a development, a personal version can be created. A personal version is technically the same as a branch.
- Each developer of ORCHIDEE can have a space on the SVN server to store one or several personal versions.
- Integration in the trunk of finalized developments must be planed in time with the ORCHIDEE project group.



# Coding Guidelines

---

All new developments to be integrated in the ORCHIDEE trunk must follow the Coding Guidelines:

- **A bench of technical tests must be ok** (restartability, debug/prod mode, running on all platforms, ...)
- **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

# Related configurations with ORCHIDEE

---

- A configuration contains the **model ORCHIDEE** and other codes needed to run the model such as **IOIPSL, XIOS and libIGCM**.
- There are **offline configurations** where only ORCHIDEE model is used and **coupled configurations** where ORCHIDEE is coupled to the atmospheric model LMDZ and sometimes other modeles as well.
- **All predefined configurations are listed in modipsl**. Modipsl is a tool developed at IPSL which we use to install the model. See hands on session.

# Related configurations with ORCHIDEE

---

## **ORCHIDEE trunk**

Use with configuration **ORCHIDEE\_trunk** and **LMDZOR\_v6.4\_work**  
ORCHIDEE\_trunk offline configuration contains the latest version of the trunk.  
For new developments this is often the version to use but to be discussed with your colleagues/supervisors depending on the project.

## **Branch ORCHIDEE\_2\_2**

Use with configuration **ORCHIDEE\_2\_2** or **LMDZOR\_v6.2\_work**  
Close to ORCHIDEE\_2\_0 with some corrections and enhancements, includes possibility to be used with DYNAMICO.

## **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.

# Different platforms

---

- Compiling options and running environment of ORCHIDEE are 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,..) and setting up compile options for ORCHIDEE.

# How to install a configuration using modipsl

---

- **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 also use modipsl to install other versions such as a branches or a personal version, or a specific revision of the trunk.

For example for offline 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 example:

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

# Compiling ORCHIDEE

---

- Default compilation in production mode, containing optimization, to be used for simulations:
  - > cd modipsl/config/ORCHIDEE\_OL
  - > **./compile\_orchidee\_ol.sh**
- With debug options, to be used to check your code and to search for errors:
  - > **./compile\_orchidee\_ol.sh -debug**
- For coupled models, a similar script is found in the corresponding config folder. For example, for LMDZOR\_v6.2 configuration:
  - > cd modipsl/config/LMDZOR\_v6
  - > **./compile\_lmdzor.sh**

# Compiling ORCHIDEE

---

- The compilation works as default at the machines that are maintained at IPSL: irene/TGCC, jeanzay/IDRIS, obelix/LSCE and ciclad/climserv/IPSL.
- The compilation script loads all modules needed for compilation and is therefore independent of personal environment on the machine. The modules are taken from the file: **modipsl/config/ORCHIDEE\_OL/ARCH/arch-X.env** where X corresponds to the current platform (X=ifort\_LSCE for obelix, X=ifort\_CICLAD for ciclad and climserv,..)
- **The same modules needs to be launched while running the model.** In the run script, or in the terminal:  
> source modipsl/config/ORCHIDEE\_OL/ARCH/arch.env



# More about compiling ORCHIDEE

---

- The compilation script will launch compilation of all components: IOIPSL, XIOS, and finally ORCHIDEE (for the offline case)
- Inside the script, another 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.

# More about compiling ORCHIDEE

---

- In older versions of the model, **a main Makefile was used instead of a compilation script**. In the same way as the compilation script, the main makefile will launch compilation of all components: IOIPSL, XIOS, and finally ORCHIDEE (for the offline case)
  - > cd modipsl/config/ORCHIDEE\_OL
  - > gmake
- Open the makefile to see which arguments it can take.
- The modules needed for compilation must be loaded in the terminal before starting the compilation with the makefile.

# Configure input parameters and variables

```
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)
.
.
.
```

Parameters and variables that needs to be set at run time, can be coded in ORCHIDEE using:

**CALL getin\_p(“VARNAME”,var)**

“VARNAME” can now be set in one of the .def files: run.def, orchidee.def or orchidee\_pft.def without recompilation of the model. Note that this function is case sensitive.

# Configure output files

---

ORCHIDEE is installed and compiled together with XIOS.

XIOS is a tool which handles reading and writing of files. It is used to produce output files containing diagnostic variables used to analyse the simulations.

Adapt the file `file_def_orchidee.xml` to set the output you want.

See presentation tomorrow.

# libIGCM: a tool for running

---

- Running the model is done using libIGCM.
- libIGCM is a script library developed at IPSL and used to run all different type of configurations, coupled as well as offline.
- Several predefined experiments are available for each configuration.
- A training course in IPSL modeling tools and environment (modipsl and libIGCM) is set up each year. This year it has been reported to April. It is highly recommended to follow this training.

# Finding information

## Wiki and web site

---

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

<http://orchidee.ipsl.fr>

**ORCHIDEE wiki** (updated 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 @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

## “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.