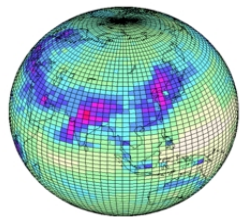
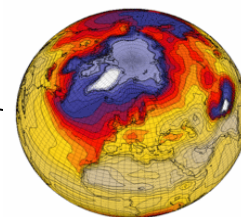
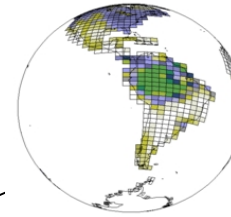
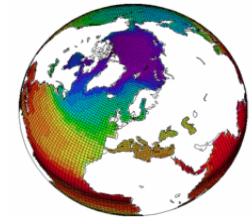
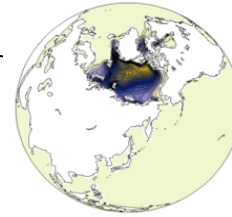




**Institut  
Pierre  
Simon  
Laplace**

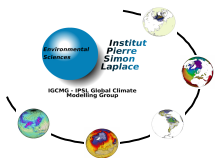
**IGCMG - IPSL Global Climate  
Modelling Group**



Décembre 2010  
Groupe de travail Plate-forme

# Le modèle couplé de l'IPSL

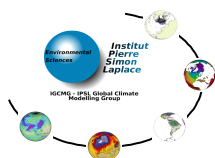
- Contexte – Présentation
  - IPSL
  - Pôle de modélisation du climat et CMIP5
  - Historique du modèle système Terre IPSL
  - Le modèle IPSLCM5A
  - Les machines utilisées
  - Les autres configurations
  - Les outils
  - La documentation
  - A venir...
- Mode d'emploi
  - Les différentes étapes
  - Les autres outils utiles : monitoring, atlas, nco, cdo,...



# Les laboratoires et les tutelles

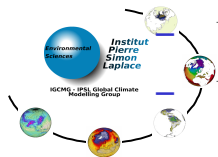


UNIVERSITE DE VERSAILLES  
 SAINT-QUENTIN-EN-YVELINES



# IPSL : Institut PS Laplace

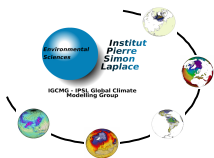
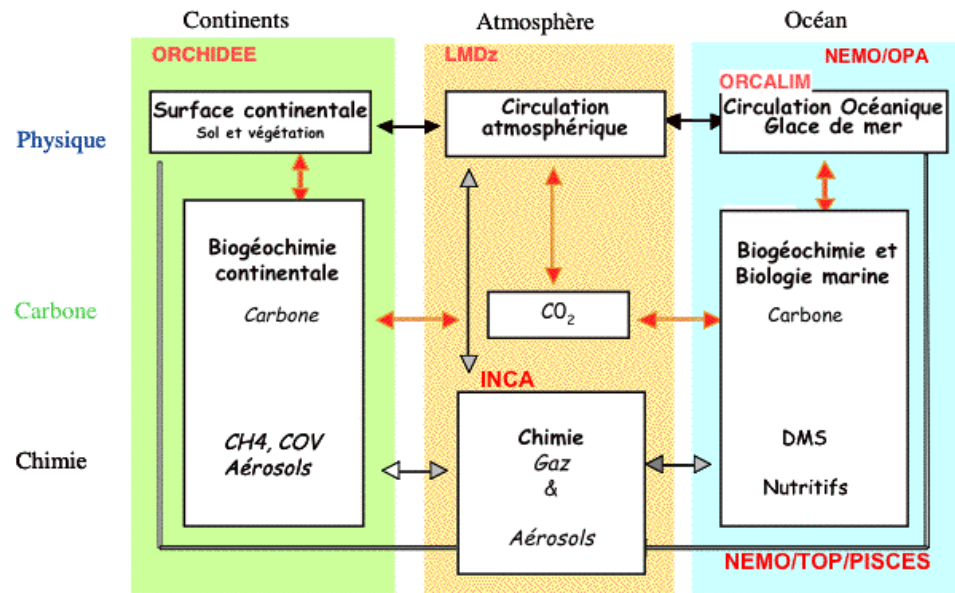
- **Fédération de 6 laboratoires, liée à 3 Observatoires des sciences de l'Univers :**
  - **Laboratoire Atmosphères, Milieux, Observations Spatiales (LATMOS) ,**
  - **Laboratoire Interuniversitaire des Systèmes Atmosphériques (LISA) ,**
  - **Laboratoire de Météorologie Dynamique (LMD) ,**
  - **Laboratoire d'Océanographie et du Climat : Expérimentation et Approches Numériques (LOCEAN),**
  - **Laboratoire de Physique Moléculaire pour l'Atmosphère et l'Astrophysique (LPMAA),**
  - **Laboratoire des Sciences du Climat et de l'Environnement (LSCE)**
- **10 tutelles**
  - **Centre National de la Recherche Scientifique (CNRS : INSU et INEE),**
  - **Université Pierre et Marie Curie - Paris 6**
  - **Université Versailles Saint-Quentin en Yvelines**
  - **Commissariat à l'Energie Atomique (CEA)**
  - **Institut de Recherche et Développement (IRD)**
  - **Ecole Normale Supérieure**
  - **Ecole Polytechnique**
  - **Centre National d'Études Spatiales (CNES).**
  - **Université Denis Diderot – Paris 7**
  - **Université Paris 12 Val de Marne**
- **Directeur : Hervé Le Treut**
- **1000 personnes**
- **Structures fédératives/Pôles scientifiques**
- **Pôle de modélisation du climat - 80 personnes – Jean-Louis Dufresne**



# Le pôle de modélisation

- Missions :
  - Fédérer les études multidisciplinaires (scientifiques ou techniques) faisant intervenir les composantes du modèle de l'IPSL
  - Identifier et coordonner les simulations de référence
  - Fédérer et rationaliser les moyens, les développements techniques
  - Animation scientifique
- Modèle climat :
  - Atmosphère
  - Océan et glace de mer
  - Surfaces continentales
  - Cycle du carbone
  - Chimie
- IPSLCM5A
- Earth System Model

Le modèle climat IPSL



# Pôle de modélisation du climat de l'IPSL : Groupes de travail

Plate-forme de modélisation  
(IPSL-ESM)

Physique et dynamique de l'atmosphère et  
de la surface  
(LMDZ, ORCHIDEE\_hydro)

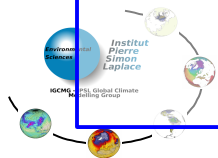
Physique et dynamique de l'océan et de la  
glace de mer  
(NEMO, LIM)

Interactions atmosphère-océan  
(IPSL-CM, différentes résolutions)

Chimie atmosphérique et aérosols  
(INCA, INCA\_aer, Reprobis)

Cycles biogéochimiques  
(PISCES, ORCHIDEE\_veget)

Distribution des données



Simulations centennales  
(20-21<sup>e</sup> siècle)

Simulations paléo, dernier millénaire

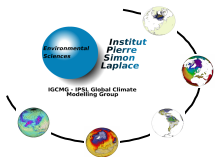
Simulations saisonnières à décennales

Simulations régionales

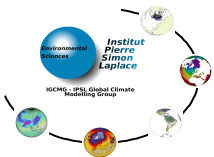
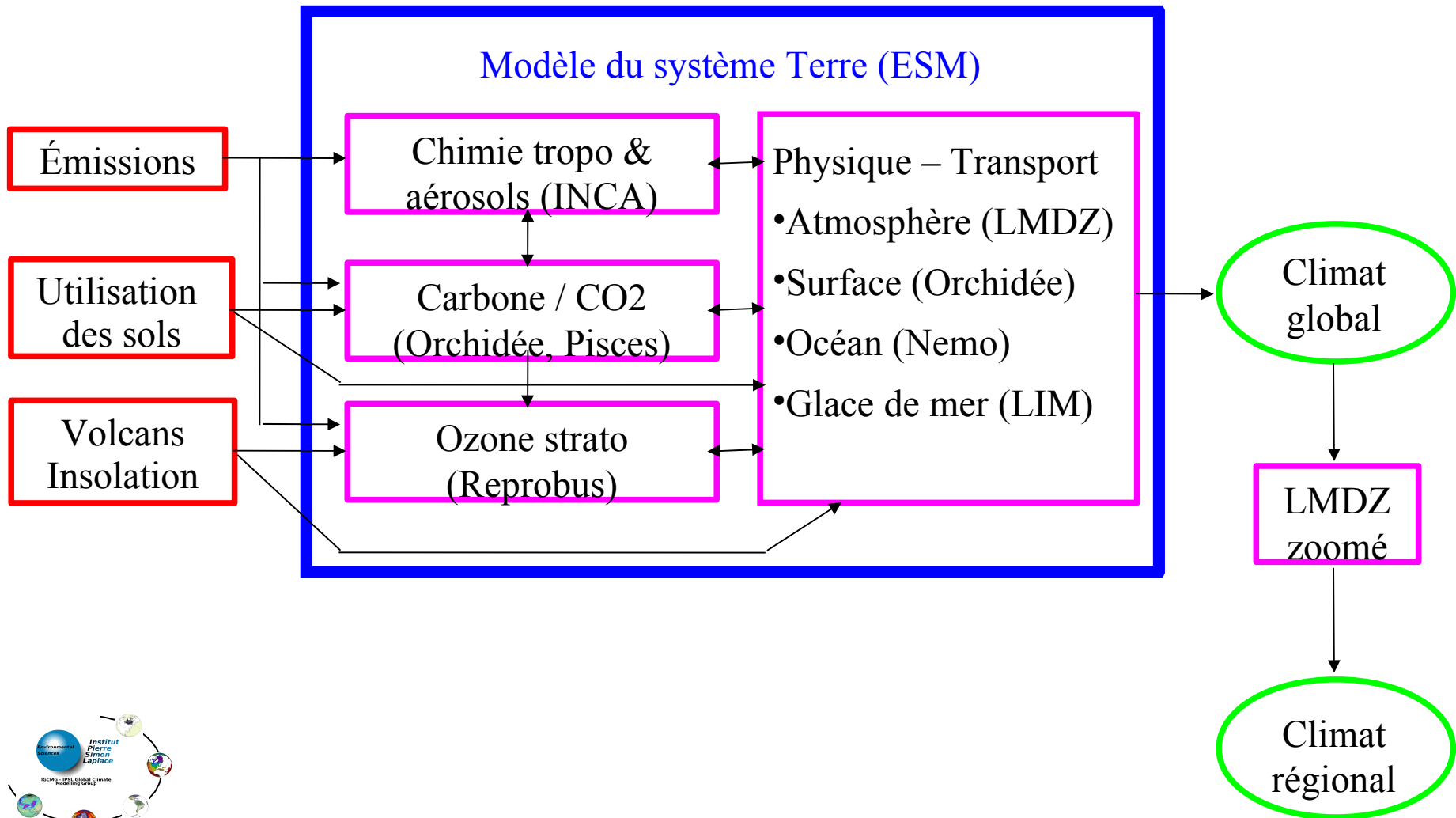
**Evaluation** des modèles  
**Analyse** du climat présent et des  
changements climatiques

# Groupe de travail Plate-forme

- Ex ESCI : Equipe Système Climat IPSL – 20 personnes
- Missions :
  - Organiser les développements techniques en accord avec les activités scientifiques du pôle
  - Assurer le lien et la cohérence des développements entre les différentes composantes et le modèle couplé
  - Support aux utilisateurs des modèles
  - Documentation
  - Animation technique, formation
  - Veille technologique

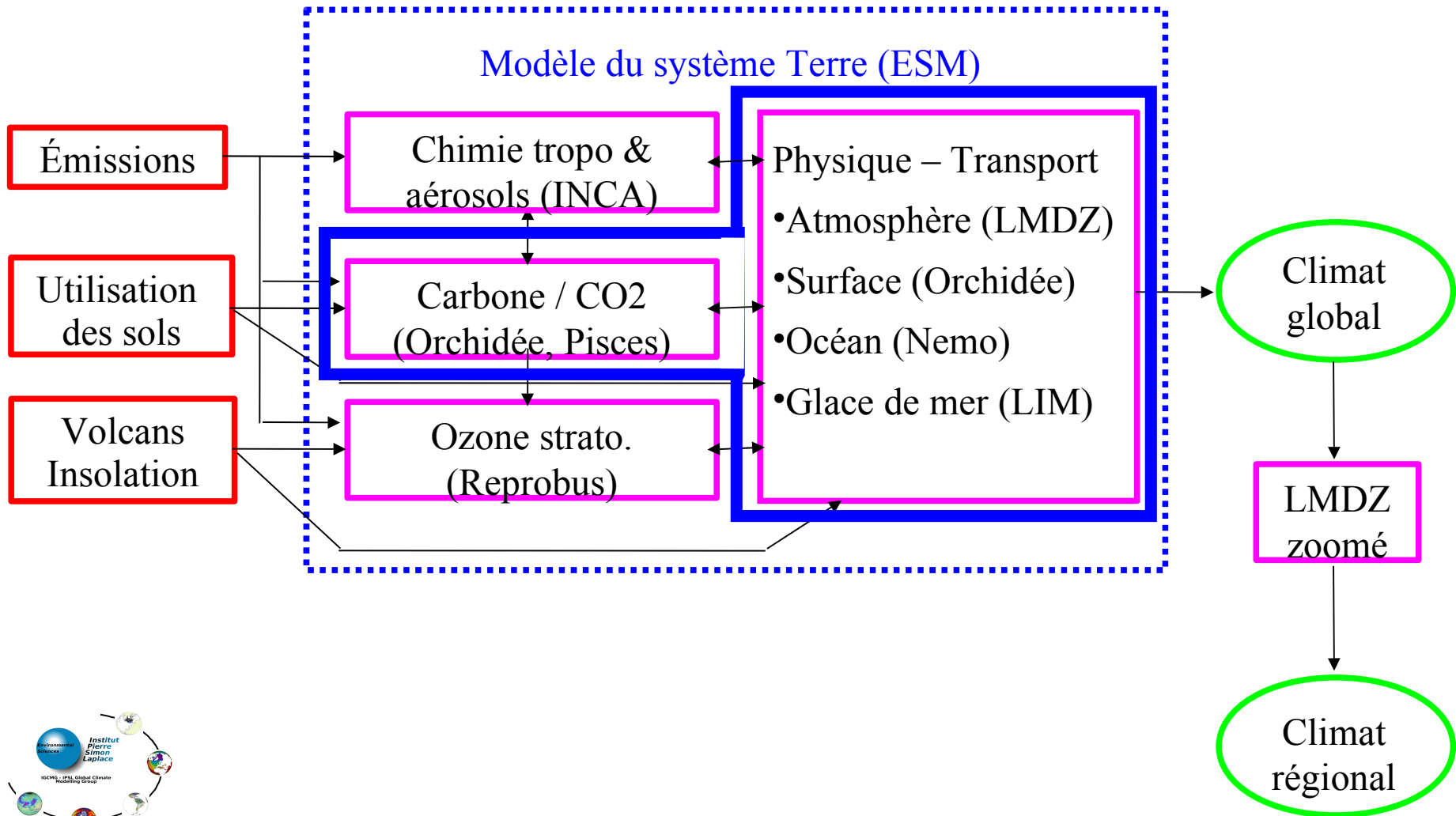


# Modèles de l'IPSL pour CMIP5

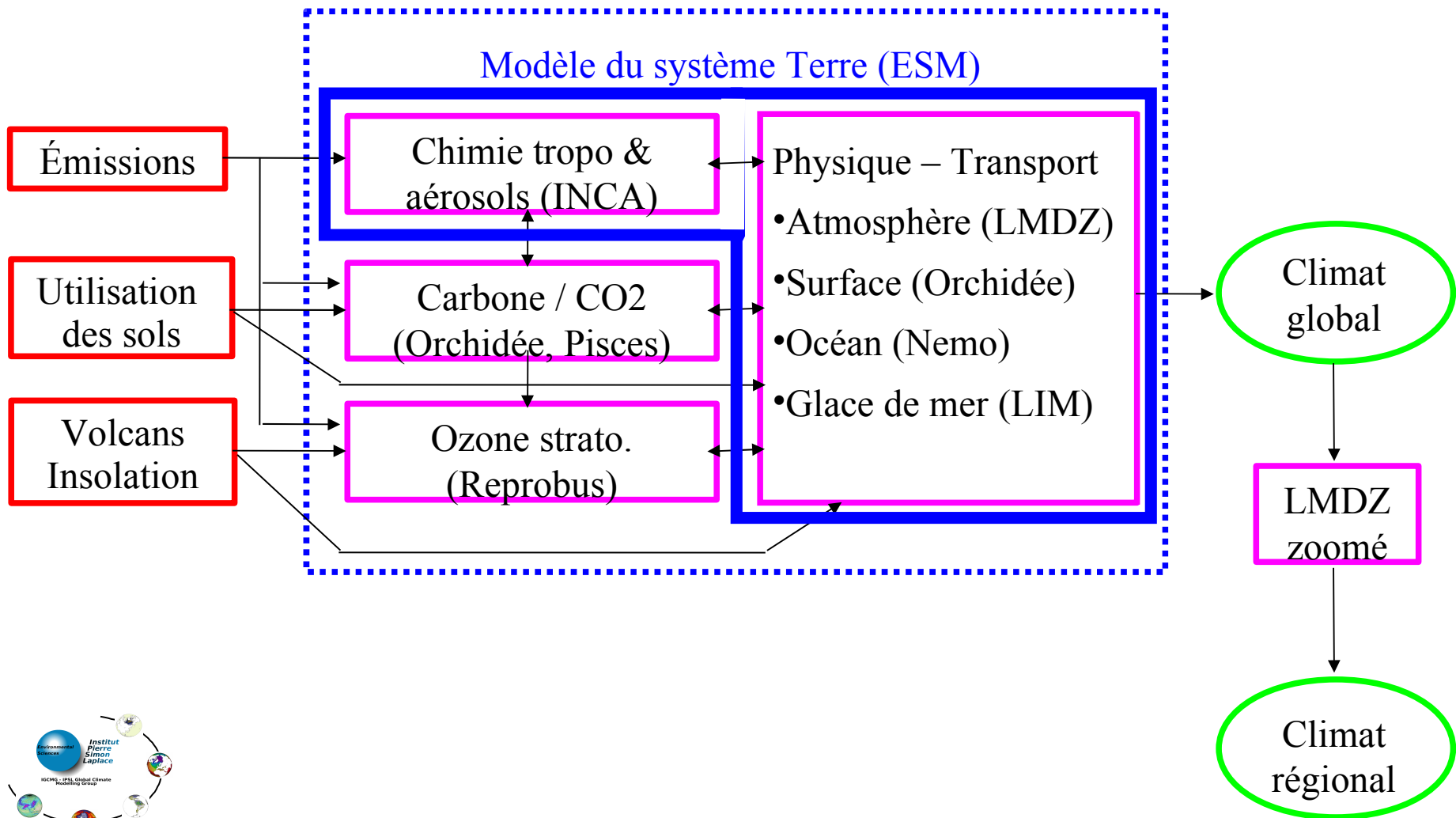




# Modèles de l'IPSL pour CMIP5



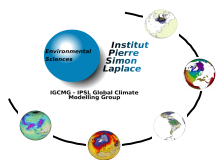
# Modèles de l'IPSL pour CMIP5



# Le modèle climat de l'IPSL

Définition : plateforme qui permet, sur les *centres de calcul usuels* :

- de *recupérer* des configurations de référence
- de *compiler* :
  - les sources des différentes composantes
  - les interfaces de couplage (océan-atmosphère) et le coupleur
- de *réaliser* une expérience type fournie (y compris fichiers entrée),
- de *suivre* son exécution,
- de *produire* et *stocker* des résultats bruts,
- de *produire, stocker* et *rendre accessible* des ATLAS et analyses systématiques



# Deux centres de calcul privilégiés

	IDRIS/CNRS	CCRT/CEA
<b>Calculateurs</b>	<ul style="list-style-type: none"> <li>• <a href="http://brodie.idris.fr">brodie.idris.fr</a> NEC SX-8, été 2006 10 noeuds 80 processeurs</li> <li>• <a href="http://vargas.idris.fr">vargas.idris.fr</a> IBM Power 6, 3584 cœurs</li> </ul>	<ul style="list-style-type: none"> <li>• <a href="http://mercure.ccc.cea.fr">mercure.ccc.cea.fr</a> NEC SX-8R, nov 2006 8 noeuds, 64 procs NEC SX-9, avril 2009, 3 noeuds, 48 procs, dédié GIEC 64 processeurs</li> <li>• <a href="http://titane.ccc.cea.fr">titane.ccc.cea.fr</a> Bull, 8544 coeurs</li> </ul>
<b>Connexion</b>	<a href="http://brodie/vargas/ulam.idris.fr">brodie/vargas/ulam.idris.fr</a> (filtrage par adresse)	<a href="http://mercure/titane/cesium.ccc.cea.fr">mercure/titane/cesium.ccc.cea.fr</a> (filtrage par adresse + port 22)
<b>Sources (conseil)</b>	\$WORKDIR	\$WORKDIR/\$SCRATCHDIR
<b>Fichiers</b>	<i>gaya</i>	\$DMFDIR
<b>Post-traitement</b>	ulam	mercure (tx7), titane, cesium
<b>Serveur DODS</b>	<a href="http://dods.idris.fr">dods.idris.fr</a>	<a href="http://dods.extra.cea.fr/data/">dods.extra.cea.fr/data/</a>
<b>Assistance-Support</b>	<a href="mailto:assist@idris.fr">assist@idris.fr</a>	<a href="mailto:hotline.ccrt@cea.fr">hotline.ccrt@cea.fr</a>
	01 69 35 85 55	01 69 26 66 66
	<a href="http://www.idris.fr">www.idris.fr</a>	<a href="http://www-ccrt.cea.fr">www-ccrt.cea.fr</a> + Intranet (firefox sur titane)



## A savoir sur brodie et IDRIS :

<http://forge.ipsl.jussieu.fr/igcmg/wiki/ModipslBeginner>

**PATH=/Txlocal/pub/svn/svn-1.3.1/bin:/home/rech/psl/rpsl035/fcm/bin:\$PATH**

pour accéder à svn et à fcm. Vérifier par `which svn` et `which fcm`

**\$WORKDIR** sur *brodie* peut être **étendu largement** (50 Go pour le groupe par exemple). Le demander à l'assistance. Pour vérifier l'occupation et la taille : `quota_u -w`

sur *brodie*, remplir le fichier *.rhosts* avec `ulam`

Faire marcher les **transferts brodie** --> **gaya** par `mfget/mfput`. *Ftuas* sur *ulam* pour faire connaître le mot de passe gaya à brodie et à toutes les machines.

Pour les **accès dods**, il faut lancer une commande `mfodds` sur *gaya*. Cela crée le répertoire, 24h après au plus tard. Accès web : <http://dods.idris.fr/login>

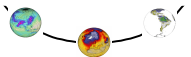
Pour les **accès dods**, sur *ulam* la commande `rsh gaya pwd` doit marcher

Pour donner les accès à tous (755 ou `drwxr-xr-x`) au **WORKDIR** de brodie, il faut **demander à l'assistance** IDRIS pour le niveau `/u/rech/grp`. Idem pour `/home_b/rech/grp` sur rhodes.

Sur *ulam* il faut explicitement charger les outils graphiques :

```
module load netcdf
```

```
module load cdo
```



# A savoir sur mercure et CCRT :

<http://forge.ipsl.jussieu.fr/igcmg/wiki/ModipslBeginner>

**PATH=\$PATH:/home/cont003/p86ipsl/fcm/bin**

pour accéder à fcm. Vérifier par `which svn` et `which fcm`

Remplir le fichier `~/forward` avec son adresse mail pour recevoir les messages de fin de simulation.

SX8 et SX9 sont 2 machines différentes :

SX9 : `module load SX9 ; module load netcdf_sx9 ;`

SX8 : `module load SX8 ; module load netcdf_sx8 ;`

(`module unload ...`)

Pour avoir accès aux différents outils, passer par les modules. Exemple:

`module load netcdf`

`module load ferret`

`module load cdo`

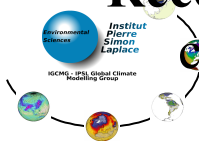
On ne travaille pas sur le `$DMFDIR` --> on transfère les fichiers sur le `$SCRATCHDIR` ou `$CSCRATCHDIR` avant de les analyser



# Les configurations cohérentes avec IPSLCM5A

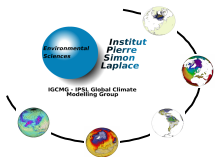
- Les forcés :
  - LMDZ4OR\_v3 : L Fairhead
  - ORCHIDEE\_OL : M Mancip
  - LMDZORINCA : A Cozic
  - ORCA2\_LIM\_PISCES : C Ethé
  - LMDZ\_REPROBUS : D Cugnet
- Les déclinaisons du Earth System model (plan 2010)
  - IPSL\_ESM\_INCA : A Cozic
  - IPSL\_ESM\_LOOP
  - IPSL\_ESM\_REPROBUS

**Recommandation** : prévenir lors de nouvelles études basées sur une de ces configurations



# A venir...

- Futur ultra proche, déjà utilisé :
  - parallélisation mixte MPI OpenMP LMDZ-ORCHIDEE => machines cibles scalaires SMP type titane ou vargas avec plus d'efficacité
  - Grand challenge au CINES, SGI, 3000 procs
    - Couplé LMDZ 1/3°- OASIS -NEMO 1/4°
  - Nouvelle physique LMDZ : IPSLCM6
- Futur moins proche
  - Serveur IO dans les modèles IPSL
  - Coupleur OASIS4 : plus de parallélisme
  - Utilisation machines scalaires MPP ~1000 procs





# Environnement

Récupération de la configuration

Serveurs CVS/SVN

Visualisation/comparaison des résultats

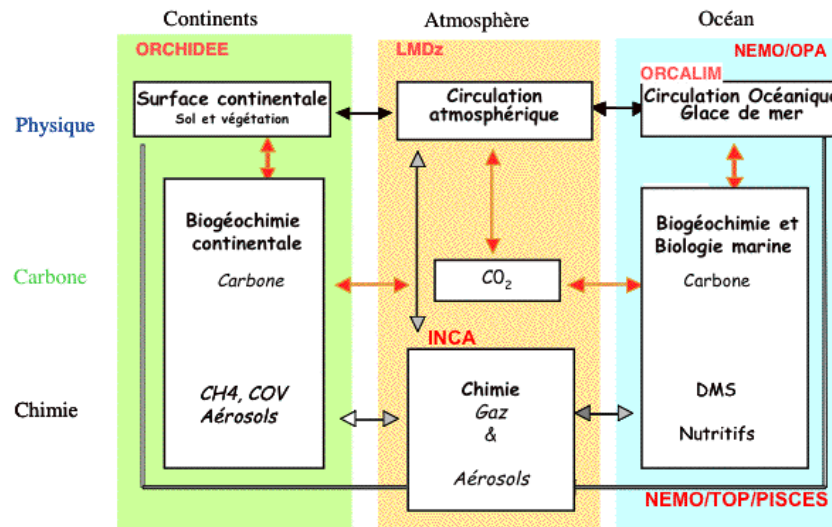
Assemblage du modèle

IOserver

Accès aux résultats

LibIGCM

Formation



Modipsl

Compilation

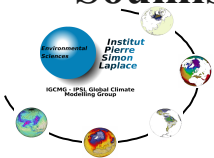
Support

Machines

Soumission/Exécution

Description d'une expérience

Documentation



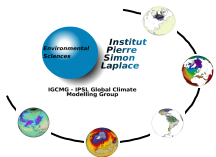
# Les outils : modipsl, ioipsl, rebuild et ioserver

- **Modipsl** : outil d'extraction, de préparation en fonction de la machine, de compilation des modèles pour créer les exécutable.

Introduction – FAQ :

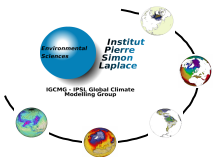
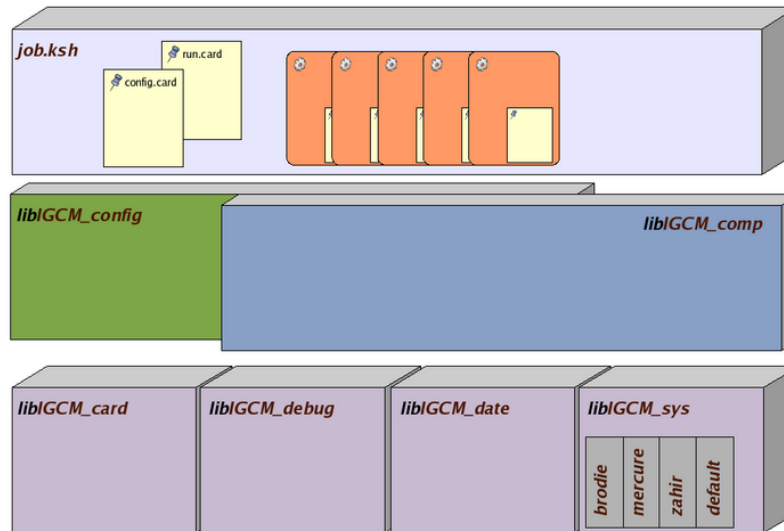
<http://forge.ipsl.jussieu.fr/igcmg/wiki/ModipslBeginner>

- **IOIPSL** : librairie commune qui gère les Entrées-Sorties (fichiers de sorties, restarts) au format NetCDF dans les modèles de l'IPSL.
- **Rebuild** : outil pour recombinaison des fichiers créés par sous-domaine, tourne maintenant sur les frontales, en asynchrone
- **IOserver** : outil gérant les Entrées/Sorties, piloté par fichier xml, capable de tourner en mode asynchrone sur 1 ou plusieurs procs.



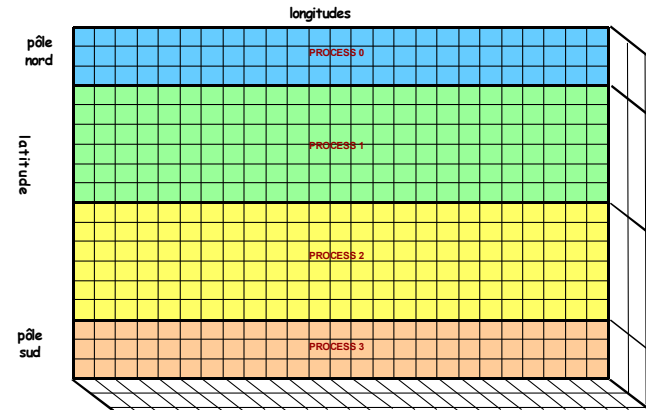
# Les outils : les scripts libIGCM

- libIGCM : ensemble de scripts de lancement de simulation et de post-traitement **modulaires** et **portables**
- Inclut le job maitre et de nombreux jobs auxiliaires de post-traitements
- Documentation : <http://forge.ipsl.jussieu.fr/libigcm>

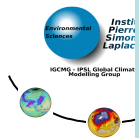


# Parallélisme

- Utilisation de plusieurs processeurs pour réduire le temps d'exécution
- Parallélisation MPI pour LMDZ/Orchidee/INCA
- Parallélisation MPI possible pour NEMO
- Machines cibles :
  - Vectoriel et parallélisme modéré  $\alpha(10)$ , MPI : Mercure (NEC SX-8R et SX-9), Brodie (NEC SX-8)
  - Scalaire SMP parallélisme massif  $\alpha(100)$  : Platine/Titane (BULL), vargas (IBM), performances améliorées mixte MPI/OpenMP
- Performances : 1 an couplé en ORCA2xLMD96x95x39

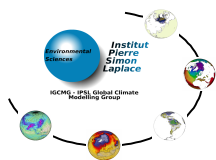


Machine	Brodie	Vargas	SX9	titane
nb procs	4 (3+1+1)	32 (26+5+1)	4 (3+1+1)	32 (26+5+1)
<b>Temps réel</b>	<b>4h</b>	<b>4h</b>	<b>3h</b>	<b>4h</b>
Temps CPU	17h	130h	11h	128h



# Documentation générale

- Wiki Pôle : <http://forge.ipsl.jussieu.fr/igcmg>
  - Introduction générale IGCMG, accessible à tous
  - Accès outils multiples
    - Accès sources SVN
    - Accès aux tickets d'incident
    - Accès aux pages wiki
  - Contrainte : être inscrit dans le projet (demande aux administrateurs) pour pouvoir **modifier** wiki, tickets et sources.
  - Machine commune de gestion des projets - Olivier Thauvin (LATMOS)



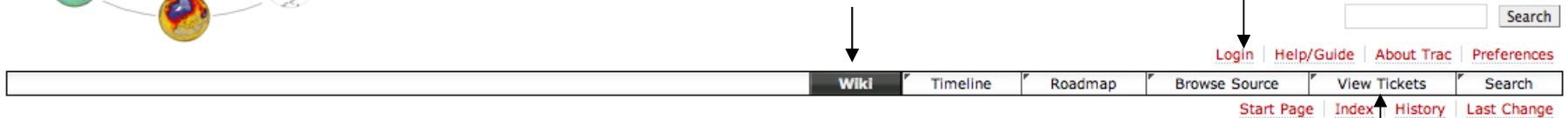
# Documentation

*forge.ipsl.jussieu.fr/igcmg*



WIKI

Sources SVN



## Welcome to IGCMG Wiki page

- [Official site for IGCMG](#) on [IPSL site](#)
- Official site for IGCMG with AR4 model documentation : <http://igcmg.ipsl.jussieu.fr/>
- Intranet site (private access) for IGCMG : [http://wiki.ipsl.jussieu.fr/wiki\\_ipsl/PoleModelisationClimat](http://wiki.ipsl.jussieu.fr/wiki_ipsl/PoleModelisationClimat)

## News

- [News](#) : all news of IGCMG group

## CMIP5 at IPSL

- [CMIP5 at IPSL](#)

## Documentation

- [Documentation](#) : all documentation about IPSL Earth System model

## The CPLIPSL group

- Direct access to [CplIpsl](#) page to have more information about the status of the IPSL global climate model.

## Projects



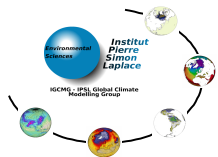
- [Start Page](#)
- [Index](#)
- [History](#)
- [Last Change](#)

Tickets

- Welcome to IGCMG Wiki page
- [News](#)
- [CMIP5 at IPSL](#)
- [Documentation](#)
- [The CPLIPSL group](#)
- [Projects](#)
- [IGCMG projects](#)
- [French research agency \(ANR\) projects](#)
- [Europeans projects](#)
- [Animations](#)

# 2ème partie

## Utilisation et démonstration



Connexion



IPSL

Gestion des sources des composantes

Serveur cvs/svn

Frontale

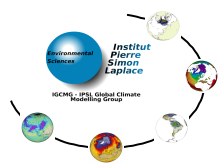
Connexion

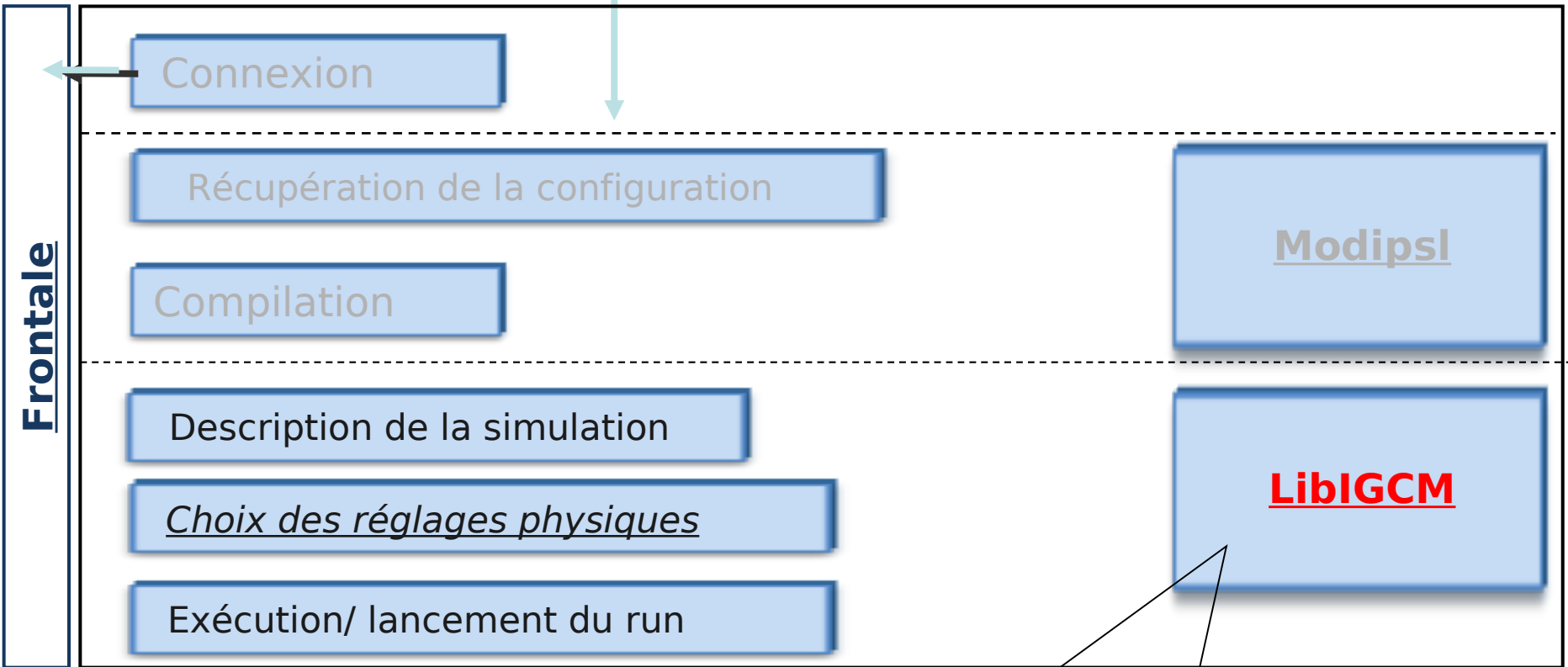
Récupération de la configuration

Compilation

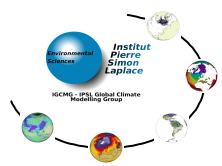
**Modipsl**

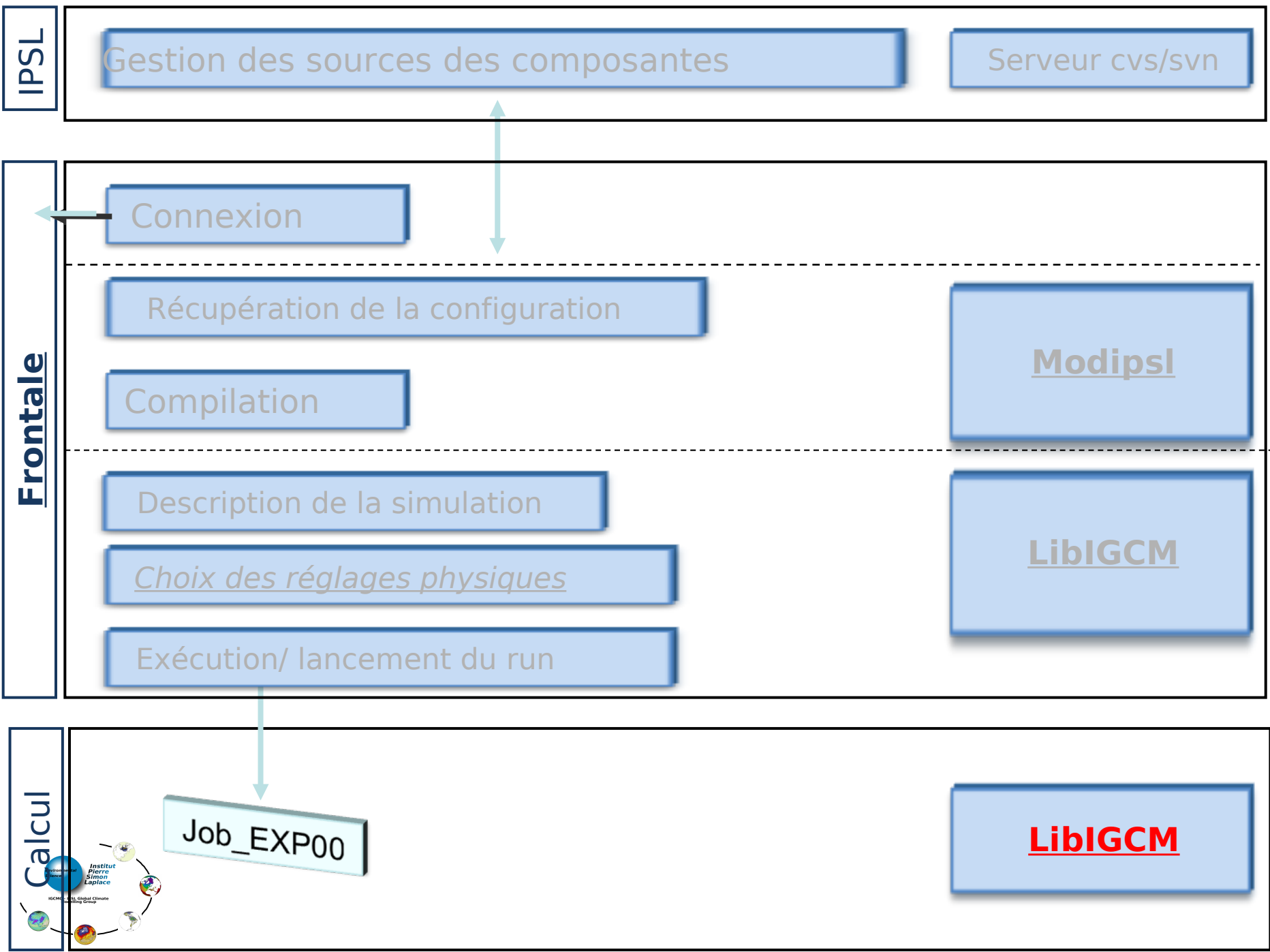
Ensemble de scripts qui va permettre de faire ces étapes sur un ensemble de machines.





ensemble de scripts de lancement de simulation et de post-traitement **modulaires et portables**





IPSL

Gestion des sources des composantes

Serveur cvs/svn

Frontale

Connexion

Récupération de la configuration

Compilation

Modipsl

Description de la simulation

*Choix des réglages physiques*

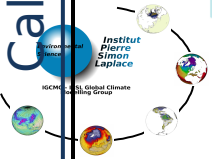
Exécution/ lancement du run

LibIGCM

Calcul

Job\_EXP00

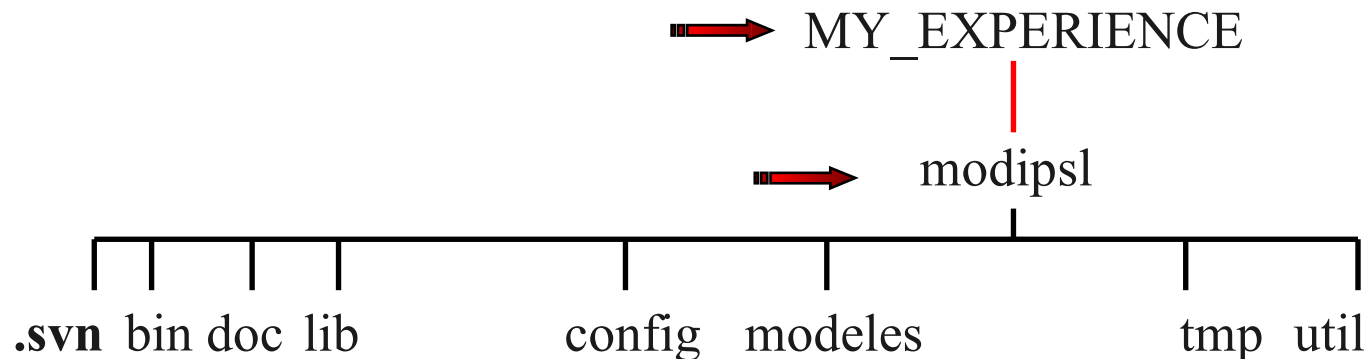
**LibIGCM**



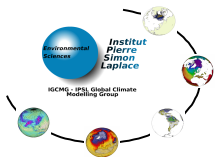
# Récupérer une configuration de modèles

Accès à *modipsl* depuis le cert ou l'idris (SVN)

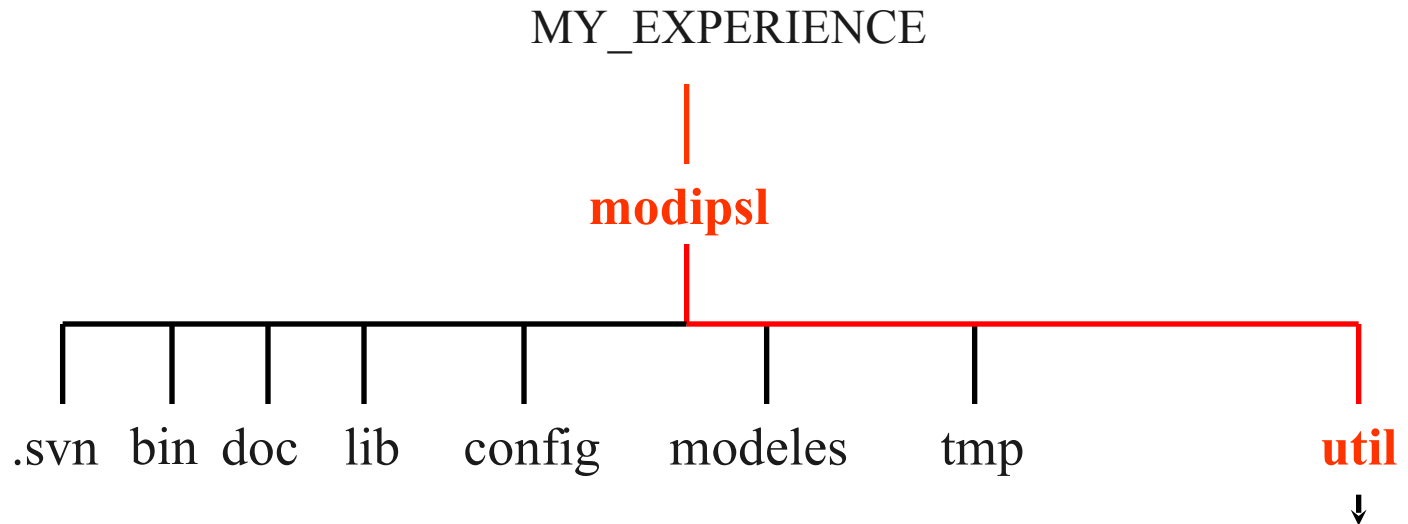
```
mkdir MY_EXPERIENCE
cd MY_EXPERIENCE
svn co http://forge.ipsl.jussieu.fr/igcmg/svn/modipsl/trunk
modipsl
```



```
cd modipsl/util/
```



# Récupérer une configuration de modèles



Définition pour chaque configuration de leurs  
composantes et de leurs tags associés



• *mod.def*

Extraction des modèles validés disponibles



• **model**

Installation et configuration des Makefiles

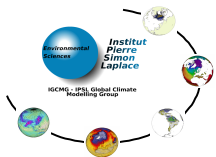


• **ins\_make**

Installation et configuration des scripts de lancement



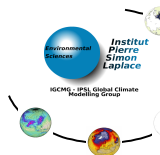
• **ins\_job**



# Récupérer une configuration de modèles

## - fichier mod.def -

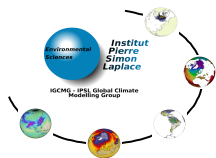
```
##- $Id$
##-----
##-This file is the definition file of the script "model".
##-----
##- Each model is defined by
##- (prefix #-H-) model informations,
##- (prefix #-M-) the email address of the model manager,
##- (prefix #-C-) elements to extract for the model, in the order :
##-   name of the component in the repository
##-   tag/revision of the component
##-   index of the repository in the server table
##-   installation path in the local working directory
##-   local working directory in modipsl
##- (prefix #-S-) containing the control system and server address.
##-
##- The tag "?" correspond to the default model version.
##- Invoking "model" with -H overrides any tag with "?".
##-----
##- Repository informations
##-
##-S- 1 cvs anonymous@cvs.ipsl.jussieu.fr:/home/ioipsl/CVSROOT
##-S- 2 cvs sechiba@cvs.ipsl.jussieu.fr:/home/ssipsl/CVSREP
##-S- 3 cvs lmdzbrowse@cvs.lmd.jussieu.fr:/home/cvsroot
##-S- 4 cvs opa@cvs.ipsl.jussieu.fr:/home/opalod/CVSROOT
##-S- 5 cvs nemo@cvs.ipsl.jussieu.fr:/home/opalod/NEMOCVSROOT
##-S- 6 cvs inca@cvs.ipsl.jussieu.fr:/home/incaipsl/CVSROOT
##-S- 7 svn http://forge.ipsl.jussieu.fr/nemo/svn
##-S- 8 svn http://forge.ipsl.jussieu.fr/igcmg/svn
##-S- 9 svn --username inca http://forge.ipsl.jussieu.fr/inca/svn
##-S- 10 svn http://forge.ipsl.jussieu.fr/libigcm/svn
##-S- 11 svn http://svn.lmd.jussieu.fr/LMDZ
##-S- 12 svn http://forge.ipsl.jussieu.fr/ioserver/svn
##-S- 13 svn http://forge.ipsl.jussieu.fr/fcm/svn
```



# Récupérer une configuration de modèles

- fichier mod.def -

```
#-H- IPSLCM5A IPSLCM5A coupled configuration
#-H- IPSLCM5A CMIP5 version 30/04/2010
#-H- IPSLCM5A NEMO svn branches/CMIP5_IPSL 1854
#-H- IPSLCM5A XMLF90 svn trunk revision 54
#-H- IPSLCM5A XMLIO_SERVER svn trunk revision 54
#-H- IPSLCM5A IOIPSL/src svn tags/v2_2_0
#-H- IPSLCM5A LMDZ4 trunk revision 1374
#-H- IPSLCM5A ORCHIDEE tag orchidee_1_9_4_2
#-H- IPSLCM5A OASIS3 tag ipslcm5a
#-H- IPSLCM5A IPSLCM5A svn
#-H- IPSLCM5A libIGCM trunk revision 265
#-M- IPSLCM5A arnaud.caubel@lsce.ipsl.fr
#-C- IPSLCM5A IOIPSL/tags/v2_2_0/src HEAD 8 IOIPSL/src modeles
#-C- IPSLCM5A ORCHIDEE orchidee_1_9_4_2 2 . modeles
#-C- IPSLCM5A OASIS3 ipslcm5a 1 prism .
#-C- IPSLCM5A LMDZ4/trunk 1374 11 LMDZ4 modeles
#-C- IPSLCM5A CONFIG/IPSLCM/IPSLCM5A HEAD 8 IPSLCM5A config
#-C- IPSLCM5A trunk/libIGCM 265 10 libIGCM .
#-C- IPSLCM5A branches/CMIP5_IPSL/NEMO 1854 7 . modeles
#-C- IPSLCM5A branches/CMIP5_IPSL/UTIL 1854 7 . modeles
#-C- IPSLCM5A XMLF90 54 12 . modeles
#-C- IPSLCM5A XMLIO_SERVER/trunk 54 12 XMLIO_SERVER modeles
```



# Récupérer une configuration de modèles

## commande model

- 1) `./model -h` --> liste des configurations de modèles disponibles
- 2) `./model -h config` --> pour une configuration donnée donne la liste des modèles la composant
- 3) `./model config` --> extraire une configuration donnée

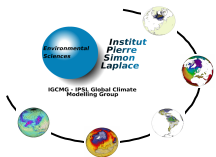
### Extraire IPSLCM5A :

```
cd modipsl/util  
./model IPSLCM5A
```

Attention:

2 mots de passe --> celui d'Orchidee, anonymous puis celui de NEMO.

Pour NEMO s'enregistrer <http://www.nemo-ocean.eu/user/register>

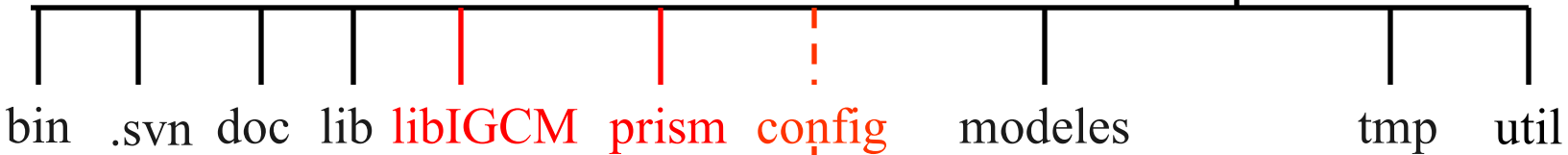




# Récupérer une configuration de modèles

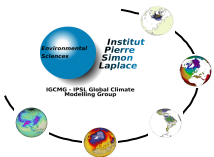
MY\_EXPERIENCE

**modipsl**



IPSLCM5A

IOIPSL UTIL ORCHIDEE NEMO LMDZ4



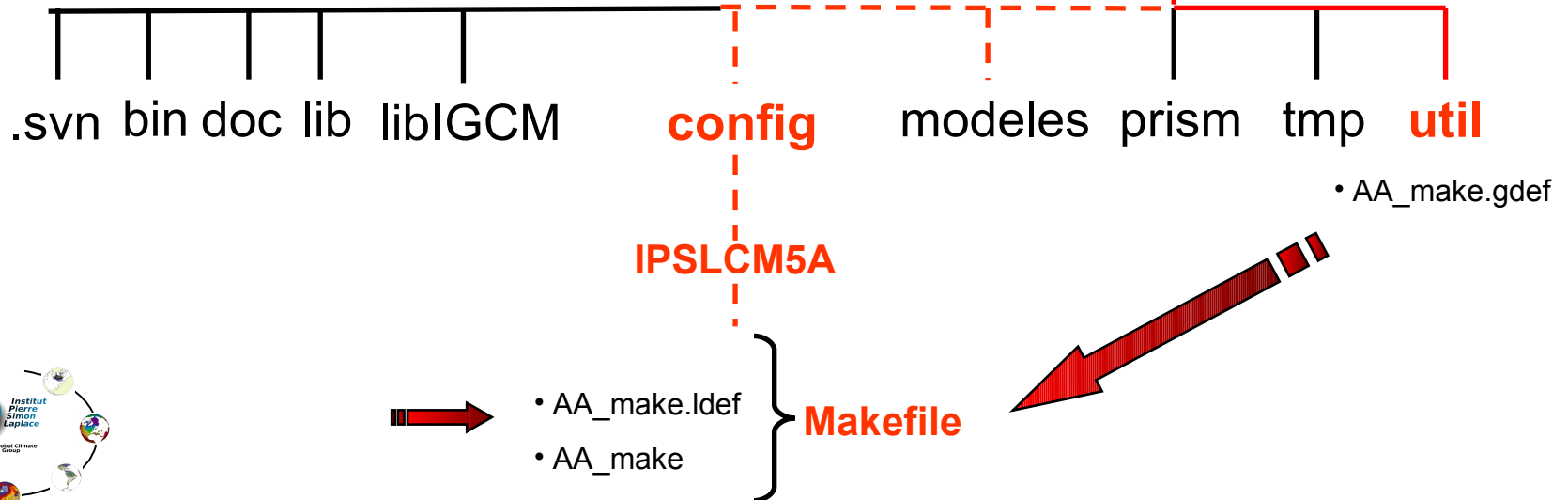
# Récupérer une configuration de modèles

## Installation des makefiles

```
cd modips1/util  
./ins_make
```

MY\_EXPERIENCE

**modips1**



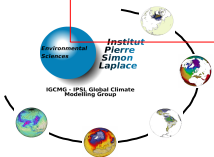
# Récupérer une configuration de modèles compilation

Lors de la compilation il faut choisir la résolution à laquelle le modèle tournera

Pour connaître les différents choix il faut ouvrir le Makefile

```
ORCA2xLMD9695 : libioipsl oasis3 liborchidee orca2 lmdz96x95x19 verif  
echo "ORCA2xLMD9695" >.resol  
echo "RESOL_ATM_3D=96x95x19" >>.resol  
echo "$(LIB_MPI)" >.libmpi
```

```
ORCA2xLMD9695-L39 : libioipsl oasis3 liborchidee orca2 lmdz96x95x39 verif  
echo "ORCA2xLMD9695-L39" >.resol  
echo "RESOL_ATM_3D=96x95x39" >>.resol  
echo "$(LIB_MPI)" >.libmpi
```



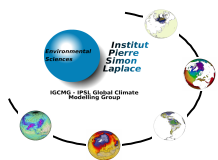
# Récupérer une configuration de modèles compilation

```
cd modips1/config/IPSLCM5A  
gmake ORCA2xLMD9695-L39
```

Création d'un fichier *.resol* à la fin de la compilation

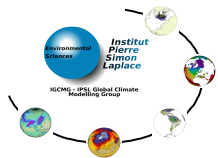
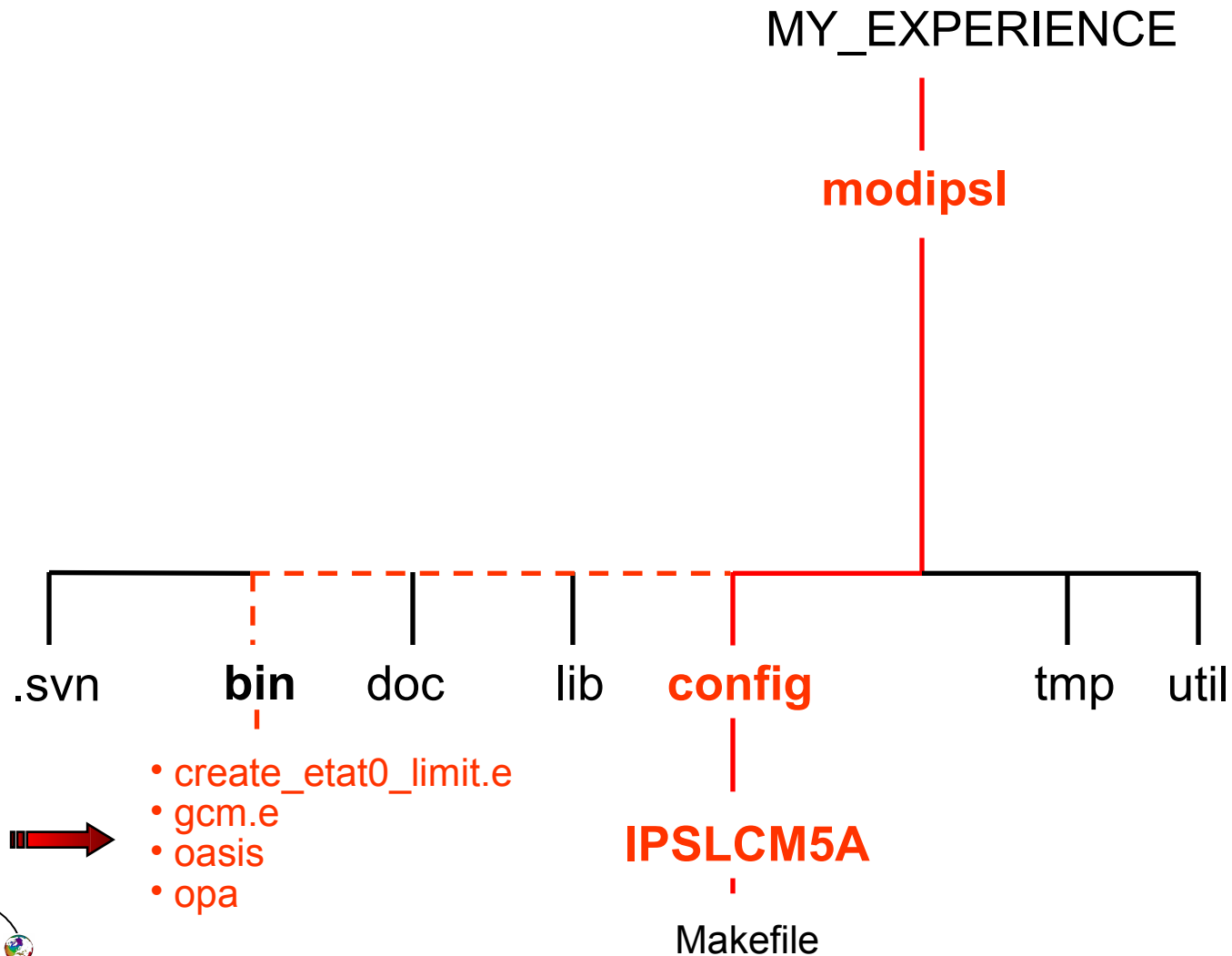
```
cd modips1/config/IPSLCM5A  
vi .resol  
  >> ORCA2xLMD9695-L39  
  >> RESOL_ATM_3D=96x95x39
```

Si la résolution indiquée dans le fichier *.resol* vous convient vous pouvez recompiler avec uniquement la commande « `gmake` »



Attention si vous recompilez alors qu'une simulation est en cours :  
lors du prochain chaînage la simulation utilisera le nouvel exécutable

# Récupérer une configuration de modèles



# Récupérer une configuration de modèles

Installation de l'expérience type

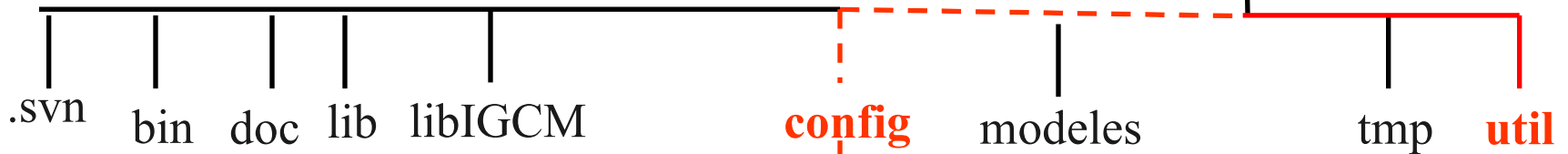
Préparation du config.card et des fichiers COMP/\*.card

Création du job

```
cd modipsl/util/  
./ins_job
```

MY\_EXPERIENCE

**modipsl**



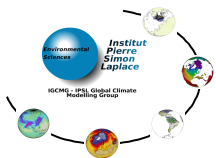
**AA\_job**

**IPSLCM5A**

**EXP00**

**config.card**

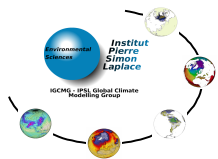
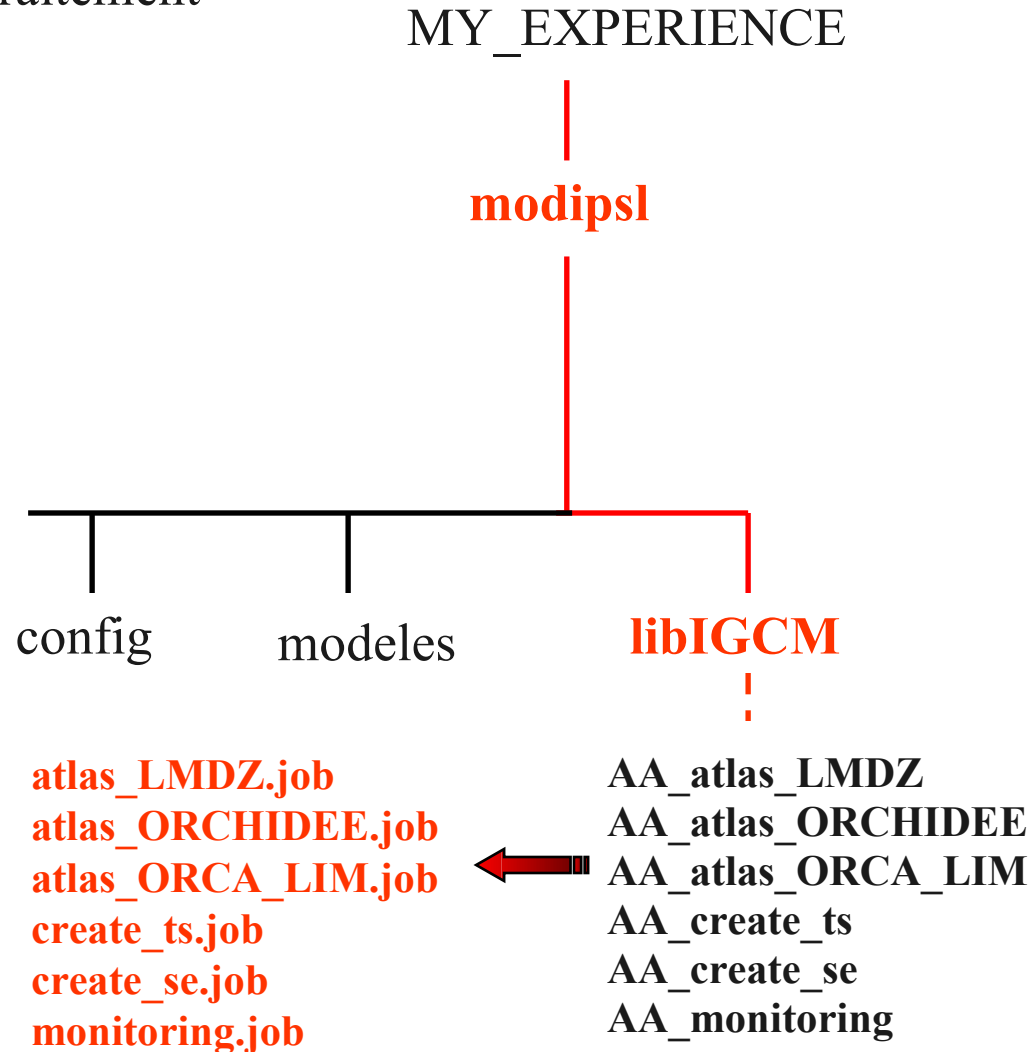
**Job\_JobName**



# Récupérer une configuration de modèles

Installation de l'expérience type  
Création des jobs de post-traitement

```
cd modipsl/util/  
./ins_job
```



# Récupérer une configuration de modèles

MY\_EXPERIENCE

Soumission du Job de lancement

```
cd modips1/config/IPSLCM5A/EXP00
qsub Job_jobname
```

modips1

**config**

**IPSLCM5A**

**EXP00**

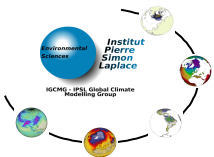
**Job\_JobName**

**COMP**

**PARAM**

lim.card  
lim.driver  
lmdz.card  
lmdz.driver  
oasis.card  
oasis.driver  
opa.card  
opa.driver  
orchidee.card  
orchidee.driver

dynami.param  
gcm.def  
geogram.param  
inice.param  
namcouple  
namelist  
offline.def  
orchidee.def  
output.param  
physiq.def  
run.def  
run.param.li  
thermo.param





# Récupérer, compiler et lancer une configuration de modèles

## 1. Accès à MODIPSL

```
svn co http://forge.ipsl.jussieu.fr/igcmg/svn/modipsl/trunk modipsl
```

## 2. Accès à IPSLCM5A

```
cd modipsl/util ; ./model IPSLCM5A
```

## 3. Installation des Makefiles

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

## 4. Compilation

```
cd modipsl/config/IPSLCM5A ; gmake + resolution choisie
```

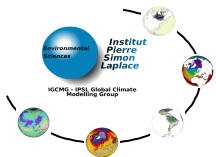
## 5. Installation de l'expérience type (et post-traitements)

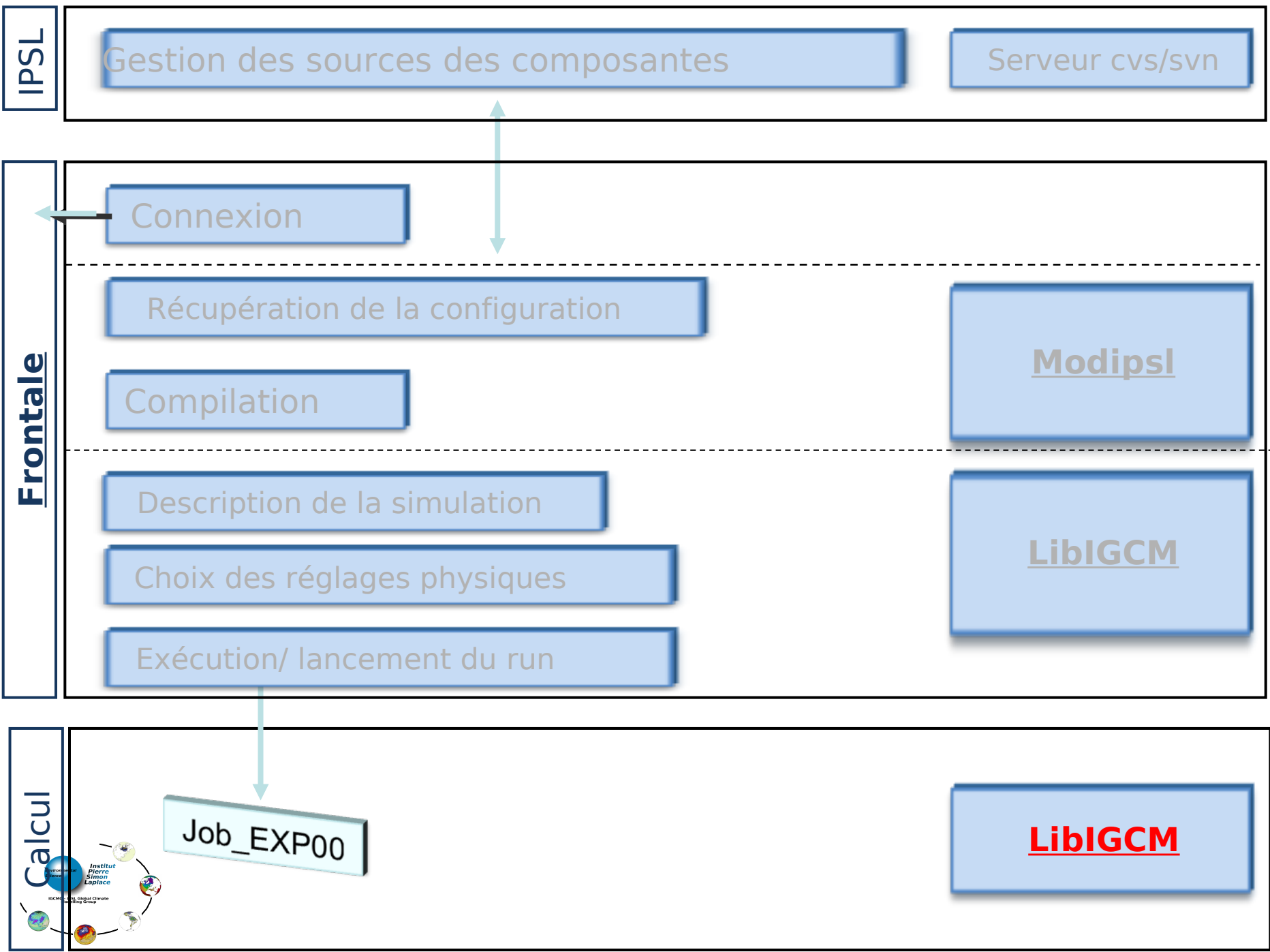
Modifier «*JobName*» dans `./config/IPSLCM5A/EXP00/config.card`

```
cd modipsl/util ; ./ins_job
```

## 6. Soumission du Job de lancement

```
cd modipsl/config/IPSLCM5A/EXP00; qsub Job_JobName
```





IPSL

Gestion des sources des composantes

Serveur cvs/svn

Frontale

Connexion

Récupération de la configuration

Compilation

Description de la simulation

Choix des réglages physiques

Exécution/ lancement du run

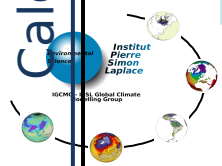
Modipsl

LibIGCM

Calcul

Job\_EXP00

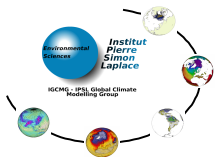
**LibIGCM**



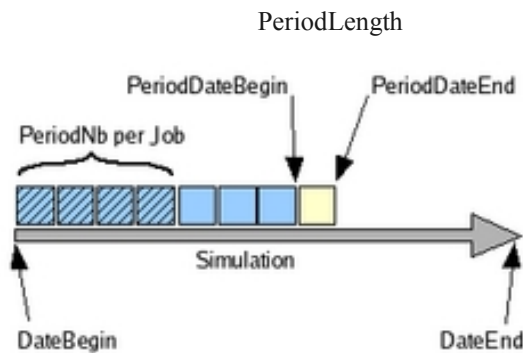
# scripts : libIGCM

Infrastructure commune cohérente de script :

- *Job\_Jobname*
  - Un job de soumission de la simulation
- config.card :
  - une fiche descriptive d'une configuration pour une simulation donnée
- COMP :
  - Des couples de fichiers *card* et *driver* **décrivent les fichiers et programment le fonctionnement** de chaque composante d'une configuration
- PARAM :
  - Des fichiers de paramètres **des différentes composantes**
- run.card (run.card.init):
  - Une fiche d'information sur la simulation en cours d'exécution
- libIGCM :
  - libIGCM/libIGCM\_card, libIGCM\_comp, libIGCM\_config, libIGCM\_date, libIGCM\_debug, libIGCM\_post, libIGCM\_sys.ksh : des bibliothèques de **fonctions** en ksh utilisées par les jobs
  - libIGCM/libIGCM\_sys/libIGCM\_sys\_brodie.ksh, libIGCM\_sys\_mercure.ksh, ... : des **fonctions** système spécifique à chaque machine

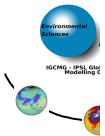


# Script de référence : AA\_Job

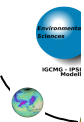
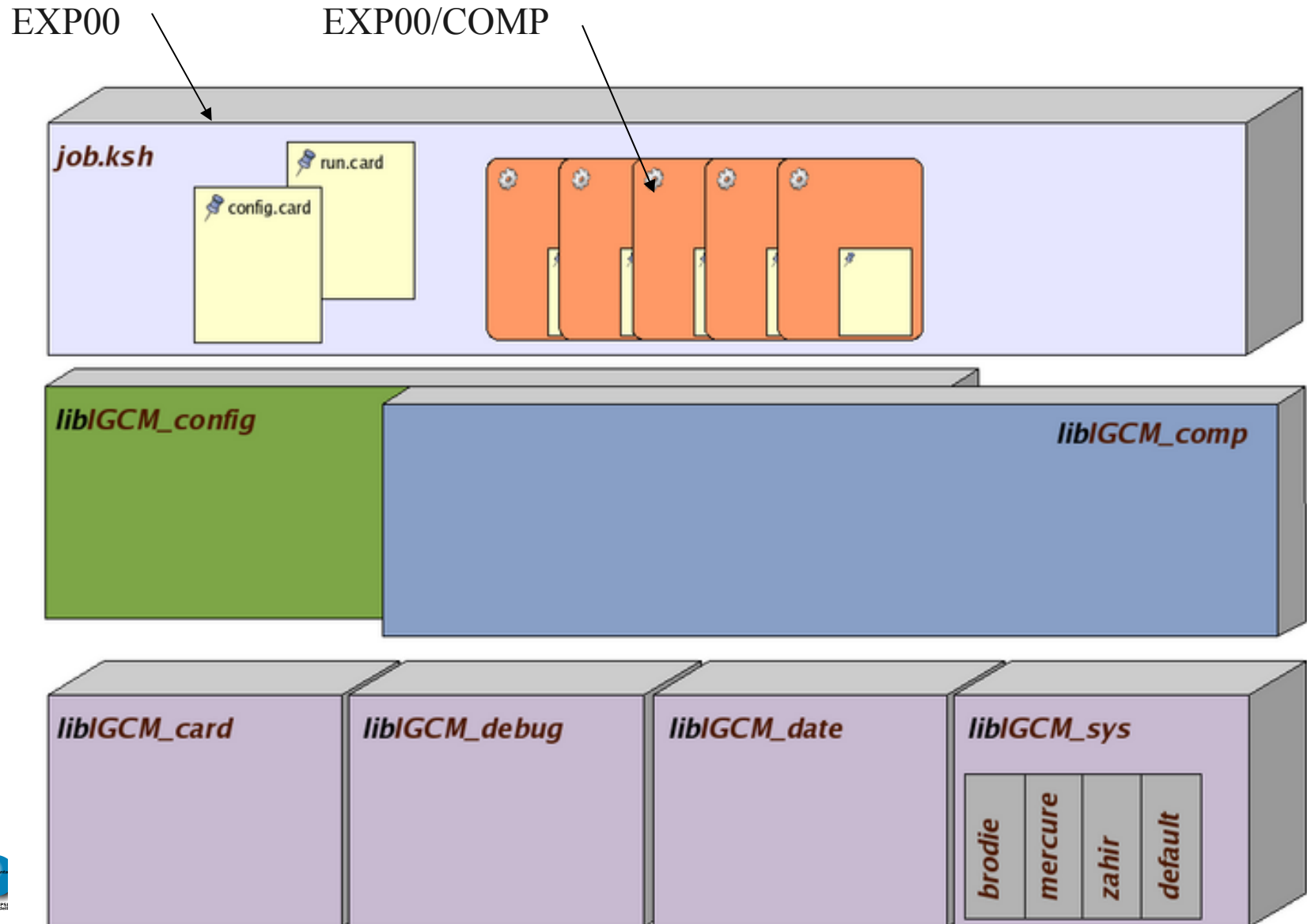


## *job.ksh*

```
IGCM_config_initialize
IGCM_comp_initialize
IGCM_config_Check
cd ${RUN_DIR}
Period=1
while [ ${Period} -le ${PeriodNb} ]; do
    echo "Starting iteration ${Period}"
    IGCM_config_PeriodStart
    IGCM_comp_GetInitialStateFiles
    IGCM_comp_GetBoundaryFiles
    IGCM_comp_GetParametersFiles
    IGCM_comp_GetRestartFiles
    IGCM_comp_Update
    ${MPIRUN_COMMAND} ${MPIRUN_OPTIONS} ./${Config_Executable_Name}
    IGCM_comp_PutRestartFiles
    IGCM_comp_PutOutputFiles
    IGCM_comp_Finalize
    IGCM_config_PeriodEnd
    echo "Ending iteration ${Period}"
    (( Period = Period + 1 ))
    (( CumulPeriod = CumulPeriod + 1 ))
done
IGCM_config_Finalize
```



# Schéma de la librairie de scripts libIGCM



# Un peu plus en détail...

MY\_EXPERIENCE

modipsl

config

IPSLCM5A

**EXP00**

**run.card.init**

**config.card**

**Job\_EXP00**

**COMP**

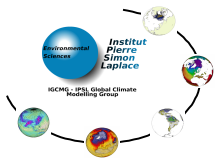
**PARAM**

lim.card  
lim.driver  
lmdz.card  
lmdz.driver  
oasis.card  
oasis.driver  
opa.card  
opa.driver  
orchidee.card  
orchidee.driver

dynami.param  
gcm.def  
geogram.param  
inice.param  
namcouple  
namelist  
offline.def  
orchidee.def  
output.param  
physiq.def  
run.def  
run.param.li  
thermo.param

Répertoire EXP00 prêt :

- COMP/\* : information sur les composantes
- config.card : fichier de configuration de la simulation
- Job\_EXP00 : Job à soumettre
- PARAM/\* : fichiers de configuration des modèles
- run.card.init : fichier de suivi original



# config.card : UserChoices

# This is config.card file for IPSLCM5A configuration

##=====

#D-- Compatibility -

[Compatibility]

libIGCM=1.0

#D-- UserChoices -

[UserChoices]

#=====

JobName=**EXP00**

#----- Short Name of Experiment

ExperimentName=**pdControl**

#----- DEVT TEST PROD

SpaceName=**DEVT**

LongName="IPSLCM5A CMIP5 DEVT phase pdControl example with limited outputs"

TagName=**IPSLCM5A**

#=====

#-- leap, noleap, 360d

CalendarType=noleap

#-- Experiment dates : Beginning and ending

#-- "YYYY-MM-DD"

DateBegin=**2000-01-01**

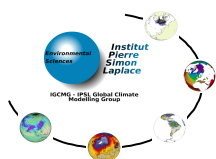
DateEnd=**2000-12-31**

#=====

#-- 1Y, 1M, 5D, 1D Period Length of one trunk of simulation

PeriodLength=**1M**

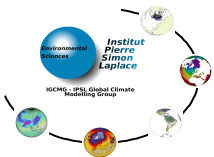
← Informations sur la simulation



# config.card : Composantes

```
#=====
#D-- ListOfComponents -
[ListOfComponents]
#D- For each component, Name of component, Tag of component
ATM= (lmdz, LMDZ4-dev)
SRF= (orchidee, ORCHIDEE_1_9_4)
OCE= (opa9, NEMO_v3_1 + revision 1340)
ICE= (lim2, LIM_2)
CPL= (oasis, OASIS3)

#=====
#D-- Executable -
[Executable]
Name=run_file
#D- For each component, Real name of executable, Name of executable for oasis
ATM= (gcm.e, lmdz.x)
SRF= ("", "")
OCE= (opa, opa.xx)
ICE= ("", "")
CPL= (oasis, oasis)
```





# config.card : Restarts

```
#=====
#D-- Restarts -
[Restarts]
#D- If you want a GENERAL RULE FOR ALL COMPONENTS RESTARTS, put this flag to 'y'
OverRule=y
#D- Last day of the experience used as restart
RestartDate=1999-12-31
#D- Define restart simulation name
RestartJobName=EXP00
#D- Path Server Group Login
RestartPath=${ARCHIVE}/IGCM_OUT/IPSLCM5A/DEVT/pdControl
```

**n** pour un démarrage à zéro

**Donner la date, le nom de la simulation  
et le chemin d'accès aux fichiers**

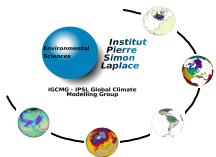
Dans cet exemple la simulation prendra comme fichiers de restart

Au ccrt :

```
/dmnfs/cont003/login/IGCM_OUT/IPSLCM5A/DEVT/pdControl/EXP00/...
/Restart/EXP00_...._19991231.nc
```

A l'Idris :

```
/u/rech/grp/login/IGCM_OUT/IPSLCM5A/DEVT/pdControl/EXP00/...
/Restart/EXP00_...._19991231.nc
```



# config.card : Post

#-----

#D-- Post -

[Post]

#D- Do we rebuild parallel output, this flag determines

#D- frequency of rebuild submission (use NONE for DRYRUN=3)

RebuildFrequency=**5Y**

#D- Do we rebuild parallel output from archive

RebuildFromArchive=**none**

**(true ) Attention à l'espace disponible pour le stockage**

#D- If you want to produce time series, this flag determines

#D- frequency of post-processing submission

TimeSeriesFrequency=**10Y**

#D- If you want to produce seasonal average, this flag determines

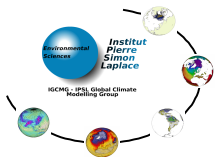
#D- the period of this average

SeasonalFrequency=**10Y**

#D- Offset for seasonal average first start dates ; same unit as SeasonalFrequency

#D- Usefull if you do not want to consider the first X simulation's years

SeasonalFrequencyOffset=**0**



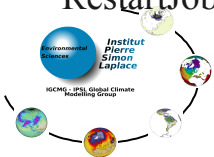
# config.card : une composante type ATM

```
#=====
#D-- ATM -
[ATM]
#
WriteFrequency="1M 1D HF"
# If config_Restarts_OverRule == 'n' all params are read
Restart= n
# Last day of the experience used as restart
RestartDate=1999-12-31
# Define restart simulation name
RestartJobName=EXP00
RestartPath=${ARCHIVE}/IGCM_OUT/IPSLCM5A/DEVT/pdControl
# Old component name for restart (if empty, use new name)
OldName=
```

y pour un redémarrage ATM depuis une autre simulation

Donner la date, le nom de la simulation et le chemin d'accès aux fichiers

```
#=====
#D-- OCE -
[OCE]
WriteFrequency="1M 1D"
Restart= n
##-- Last day of the experience used as restart
RestartDate=1999-12-31
# Define restart simulation name
RestartJobName=EXP00
RestartPath=${ARCHIVE}/IGCM_OUT/IPSLCM5A/DEVT/pdControl
# Old component name for restart (if empty, use new name)
OldName=
```

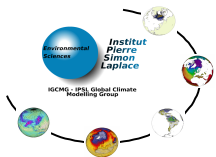


# RebuildFromArchive

- A l'Idris nous conseillons le RebuildFromArchive
- Au CCRT nous conseillons le RebuildFromWorkdir --> dans ce cas là il faut faire très attention aux quotas du scratchdir :

Pour 10 ans de simulation

- historical de IPSLCM5A ~ 400 Go
  - PiControl de IPSLCM5A ~ 210 Go
- > donc toujours bien vérifier l'espace disque disponible  
--> réduire le RebuildFrequency en conséquence



# Un peu plus en détail...

MY\_EXPERIENCE

modipsl

config

IPSLCM5A

**EXP00**

**run.card.init**

**config.card**

**Job\_EXP00**

**COMP**

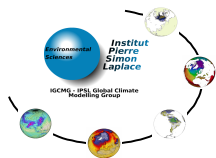
**PARAM**

lim.card  
lim.driver  
lmdz.card  
lmdz.driver  
oasis.card  
oasis.driver  
opa.card  
opa.driver  
orchidee.card  
orchidee.driver

dynami.param  
gcm.def  
geogram.param  
inice.param  
namcouple  
namelist  
offline.def  
orchidee.def  
output.param  
physiq.def  
run.def  
run.param.li  
thermo.param

Répertoire EXP00 prêt :

- COMP/\* : information sur les composantes
- config.card : fichier de configuration de la simulation
- Job\_EXP00 : Job à soumettre
- PARAM/\* : fichiers de configuration des modèles
- run.card.init : fichier de suivi original



# Un peu plus en détail...

MY\_EXPERIENCE

|  
modipsl

|  
config

|  
IPSLCM5A

|  
**EXP00**

|  
**run.card.init**

|  
**config.card**

|  
**Job\_CM5**

|  
**COMP**

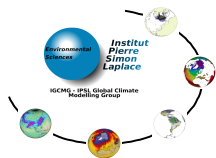
|  
**PARAM**

lim2.card  
lim2.driver  
lmdz.card  
lmdz.driver  
oasis.card  
oasis.driver  
opa9.card  
opa9.driver  
orchidee.card  
orchidee.driver

dynami.param  
gcm.def  
geogram.param  
inice.param  
namcouple  
namelist  
offline.def  
orchidee.def  
output.param  
physiq.def  
run.def  
run.param.li  
thermo.param

Répertoire EXP00 prêt :

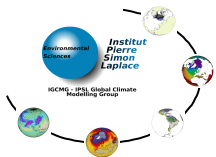
- COMP/\* : information sur les composantes
- config.card : fichier de configuration de la simulation
- Job\_EXP00 : Job à soumettre
- PARAM/\* : fichiers de configuration des modèles
- run.card.init : fichier de suivi original



# Flux des données

Un fichier descriptif par composante (par ex opa9.card)

- Fichiers d'entrée texte (namelist)
- Fichiers d'entrée binaires :
  - conditions initiales
  - conditions limites
- Fichiers de sorties binaires (netCDF)
- Fichiers de sorties texte
- Fichiers de redémarrage



# Flux des données

## Les fichiers COMP/\* .card

Les différentes rubriques :

```
[Compatibility]
```

```
LibIGCM=1.0
```

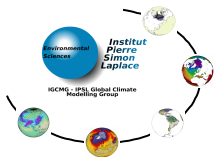
--> compatibilité avec la librairie libIGCM

```
[UserChoices]
```

```
[InitialStateFiles]
```

```
List= ($ {R_INIT}/SRF/${config_UserChoices_TagName}/soils_param.nc,      . ), \
      ($ {R_INIT}/SRF/${config_UserChoices_TagName}/routing.nc,          . ), \
      ($ {R_INIT}/SRF/${config_UserChoices_TagName}/PFTmap_IPCC_1850.nc,
PFTmap.nc)
```

--> fichiers d'états initiaux





# Flux de données

## les fichiers COMP/\*.card

```
[BoundaryFiles]
```

```
List=(/dmnfs/cont003/p24data/ECMWF320x160/AN${year}/165_${year}${month}.nc,  
u10mec.nc)
```

```
ListNonDel=(${R_INIT}/CHM/LMDZORINCA/CH4/INCA${RESOL_CHM}/o3clim.nc,o3clim.nc) \  
            (${R_INIT}/CHM/LMDZORINCA/CH4/INCA${RESOL_CHM}/so4.nc ,so4.nc )
```

--> Liste des fichiers contenant les conditions aux limites. Deux catégories : List (fichiers rechargés à chaque période de simulation) et ListNonDel (chargés une seule fois)

```
[SmoothFiles]
```

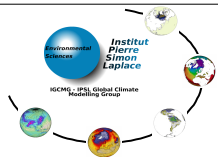
```
List=(${R_BC}/SRF/${  
{config_UserChoices_TagName}/PFTmap_1850to2005_AR5_LUHa.rc2/PFTmap_IPCC_${  
{year}.nc, PFTmap.nc, 12:12:)
```

--> Liste de fichiers rechargés toutes les n-périodes de simulations

```
[ParametersFiles]
```

```
List= (${SUBMIT_DIR}/PARAM/orchidee.def, .)
```

--> Liste des fichiers de paramètres (normalement stockés dans PARAM/)



# Flux de données

## Les fichiers COMP/\*.card

```
[RestartFiles]
```

```
List=(${config_UserChoices_JobName}_${NEMO_END}_restart.nc, restart.nc,  
restartopa.nc)
```

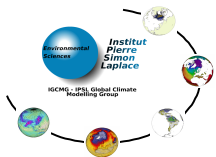
```
[RestartFiles]
```

```
List=(restart.nc, restart.nc, start.nc), \  
      (restartphy.nc, restartphy.nc, startphy.nc)
```

--> Liste des fichiers de restarts pour le chainage des simulations

Syntaxe :

```
List= (restart en sortie du modèle, restart stocké, restart en entrée du  
modèle pour le chainage)
```



# Flux de données

## Les fichiers COMP/\* .card

```
[OutputText]
```

```
List= (physiq.def, run.def)
```

--> Liste des fichiers textes que l'on veut conserver en fin de simulation

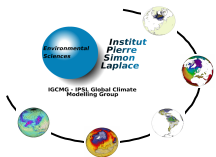
```
[OutputFiles]
```

```
List= (sechiba_history.nc, ${R_OUT_SRF_O_M}/$  
{PREFIX}_1M_sechiba_history.nc, Post_1M_sechiba_history), \  
(orchidee_watchout.nc, ${R_OUT_SRF_O_M}/$  
{PREFIX}_1M_watchout.nc, NONE)
```

--> Liste des fichiers output

Syntaxe :

```
List= (output file, stockage, post-traitement appliqué)
```



# Flux de données

## Fichiers COMP/\*.card

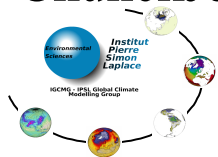
```
[Post_1M_sechiba_history]
Patches= ()
GatherWithInternal= (lon, lat, veget, time_counter, time_counter_bnds, Areas,
Contfrac)
TimeSeriesVars2D= (nobiofrac, alb_nir, alb_vis, bqsb, evap, fluxlat, fluxsens,
gqsb, netrad, qair, rain, runoff, snow, snownobio, snowf, subli, tair, temp_sol,
tsol_max, tsol_min, drainage,mrsos, mrso, mrros, mrro, prveg, evspsblveg,
evspsblsoi, tran, treeFrac, grassFrac, cropFrac, baresoilFrac, residualFrac)
ChunckJob2D= NONE
TimeSeriesVars3D= (lai, maxvegetfrac, vegetfrac, CO2FLUX, ptn, nee)
ChunckJob3D= NONE
Seasonal= ON
```

--> Liste des post-traitements à appliquer aux fichiers d'Output

GatherWithInternal = liste des variables à ajouter aux times series

ChunckJob2D/3D = NONE si time series sur toute la simul,

= 50Y si time series sur une partie de la simulation



# Fichiers COMP/\* .card

Variables et syntaxe des fichiers COMP/\* .card :

Au ccrt :

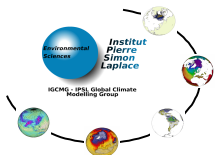
$\${R\_INIT}$  = /dmnfs/cont003/p86ips1/IGCM/INIT/

$\${R\_BC}$  = /dmnfs/cont003/p86ips1/IGCM/BC/

A l'Idris

$\${R\_INIT}$  = /u/rech/psl/rpsl035/IGCM/INIT/

$\${R\_BC}$  = /u/rech/psl/rpsl035/IGCM/BC/



# Fichiers COMP/\* .card

Sur toutes les machines :

$\{\text{SUBMIT\_DIR}\}$  = répertoire d'expérience

$\{\text{year}\}$  = année en cours

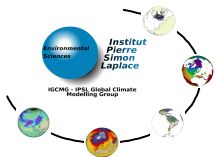
$\{\text{month}\}$  = mois en cours

$\{\text{config\_UserChoices\_JobName}\}$  = nom du Job (JobName dans config.card)

(path/filename, newfilename) = cp path/filename newfilename

( ..., ... ) , \ = la liste continue à la ligne suivante

Attention : ne pas mettre d'espace en fin de ligne



# Un peu plus en détail...

MY\_EXPERIENCE

|  
modipsl

|  
config

|  
IPSLCM5A

|  
**EXP00**

**run.card.init**

**config.card**

**Job\_EXP00**

**COMP**

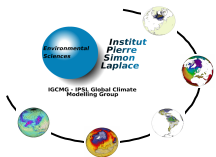
**PARAM**

lim.card  
lim.driver  
lmdz.card  
lmdz.driver  
oasis.card  
oasis.driver  
opa.card  
opa.driver  
orchidee.card  
orchidee.driver

dynami.param  
gcm.def  
geogram.param  
inice.param  
namcouple  
namelist  
offline.def  
orchidee.def  
output.param  
physiq.def  
run.def  
run.param.li  
thermo.param

Répertoire EXP00 prêt :

- COMP/\* : information sur les composantes
- config.card : fichier de configuration de la simulation
- Job\_EXP00 : Job à soumettre
- PARAM/\* : fichiers de configuration des modèles
- run.card.init : fichier de suivi original



# Job\_ *JobName*

## Initialisation des paramètres de batch (exemple PBS)

1. Définition de la mémoire limite

`#PBS -l memsz_job=15.0gb`                      limite mémoire

2. Définition du nombre de processeurs

`#PBS -v PBS_NUM_PROC_TOT=::JobNumProcTot::` (provient de config.card via `./ins_job`)

3. Définition des limites temps CPU

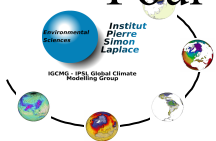
**Sur Brodie :**

`#PBS -l cputim_job=1:00:00`      limite en temps CPU pour l'ensemble du job

**Sur Mercure :**

`#PBS -l elapstim_req=1:00:00`      limite en temps réel 'elapsed' pour l'ensemble du job

Pour connaître les différentes queues disponibles sur une machine il faut utiliser la commande « `class` »





# Job\_ *JobName* : PBS

Caractéristiques pour une expérience de 1 mois  
ORCA2xLMD9695-L39

Plateforme	Temps CPU	Temps écoulé	Mémoire
Brodie : 4 procs	4000 s	1300 s	12 Gb
Mercure-SX9 : 4 procs	2400 s	800 s	12 Gb

Classe du job:

brodie

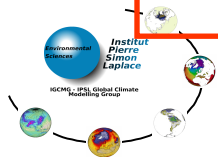


p4t2

mercure



para-sx9



# Job\_ *JobName* : PeriodNb

Lancement de plusieurs périodes par job

Pour éviter de lancer une foule de petits jobs qui reprennent la file d'attente à chaque fois, il est possible de lancer en boucle n périodes par job. Le paramètre à modifier est dans *Job\_JobName* (1 par défaut) : **PeriodNb=1**

**Attention!** Modifier le paramètre PBS du temps en conséquence.

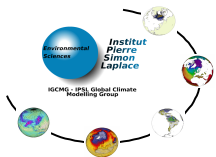
Définition des limites temps CPU

**Sur Brodie :**

**#PBS -l cputim\_job=10:00:00** temps CPU pour l'ensemble du job

**Sur Mercure :**

**#PBS -l elapstim\_req=10:00:00** temps réel 'elapsed' pour l'ensemble du job



# Soumission – contrôle

- **qsub Job\_EXP00**
- **Contrôle**
  - Mercure : qstat, mpp
  - Brodie : qstat
- Répertoire EXP00 en cours de simulation :
  - COMP/\*
  - PARAM/\*
  - run.card.init
  - **run.card**
  - config.card
  - Job\_EXP00
  - **Script\_Output\***

MY\_EXPERIENCE

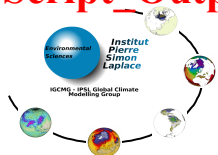
|  
modips1

|  
config

|  
IPSLCM5A

|  
**EXP00**

Script\_Output\* run.card run.card.init config.card Job\_EXP00 COMP PARAM

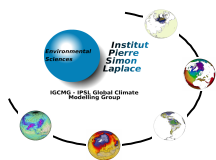


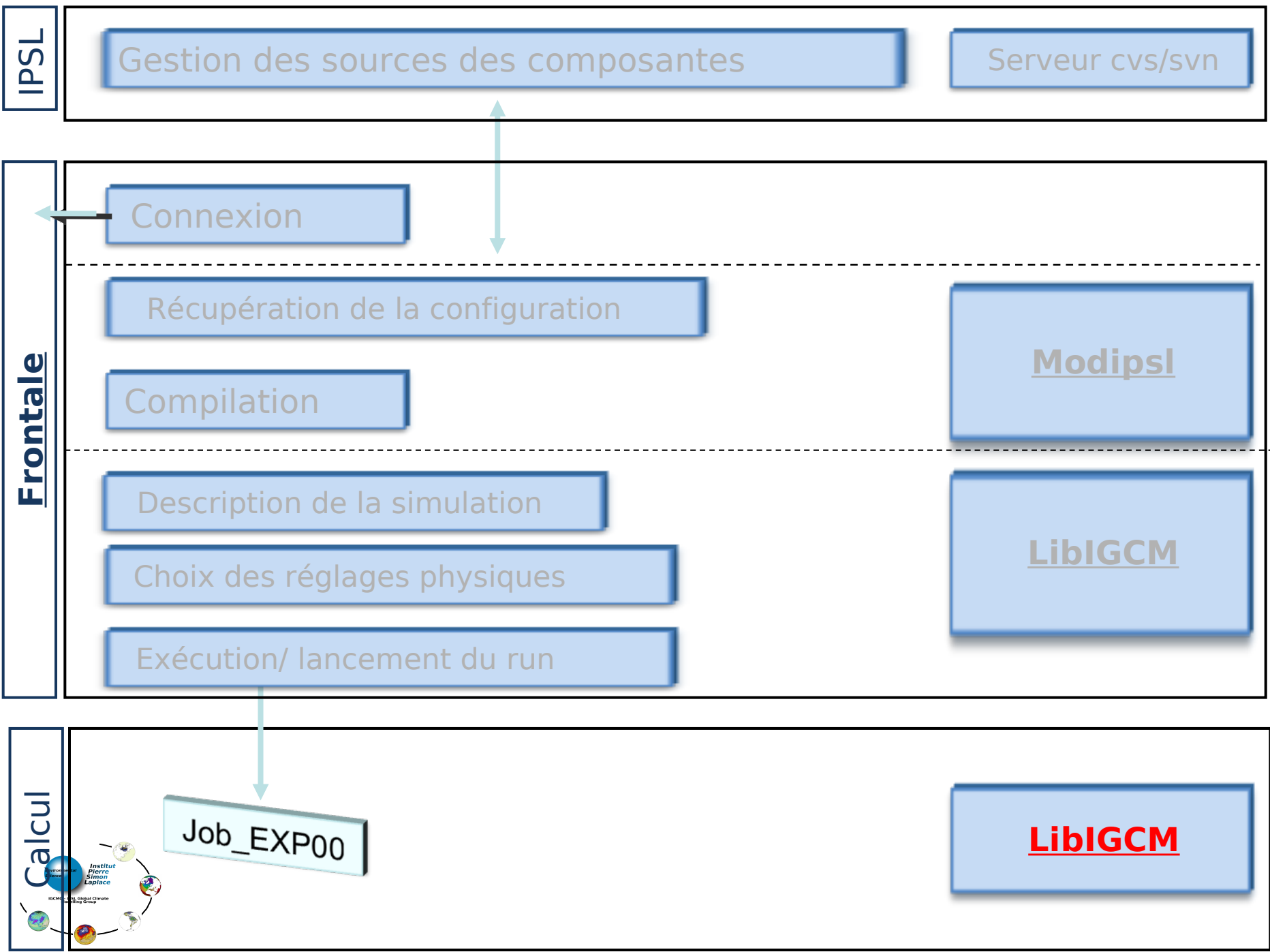
# run.card : le fichier de suivi

```
# contient la date de la periode en cours ou en attente
# last date of loop == .suivi
[Configuration]
#last PREFIX
OldPrefix= EXP00_20000131
#Compute date of loop == .suivi
PeriodDateBegin= 2000-02-01
PeriodDateEnd= 2000-02-28
CumulPeriod= 2
# State of Job "Start", "Running", "OnQueue", "Completed"
PeriodState= Running

[PostProcessing]

TimeSeriesRunning=n
TimeSeriesCompleted=
```





IPSL

Gestion des sources des composantes

Serveur cvs/svn

Frontale

Connexion

Récupération de la configuration

Compilation

Modipsl

Description de la simulation

Choix des réglages physiques

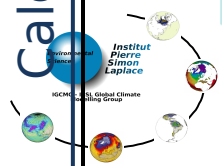
LibIGCM

Exécution/ lancement du run

Calcul

Job\_EXP00

**LibIGCM**



Calcul

Job\_EXP00



Job\_EXP00



**LibIGCM**



Calcul

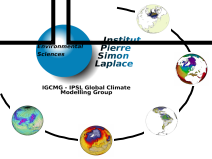


**LibIGCM**

Machine de post-traitement



**LibIGCM**



Calcul



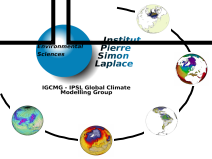
**LibIGCM**

Machine de post-traitement



Outil de recombinaison des fichiers produits par chaque processus de calcul. Etape en mode asynchrone.

**LibIGCM**



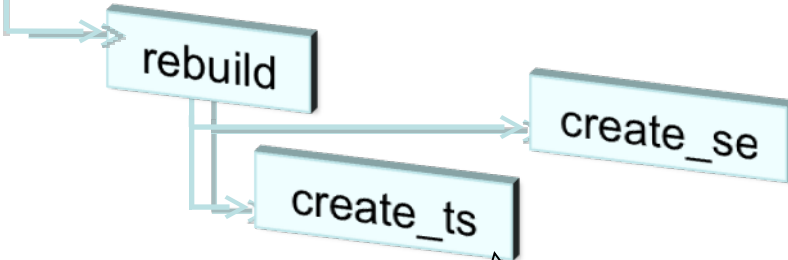


Calcul



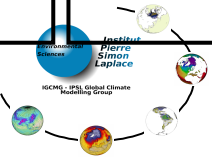
**LibIGCM**

Machine de post-traitement



Génération de séries temporelles de variables spécifiques.

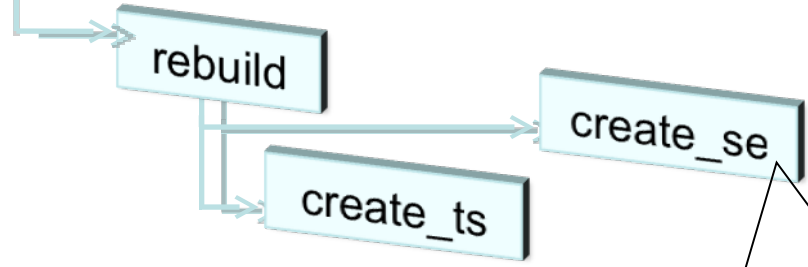
**LibIGCM**



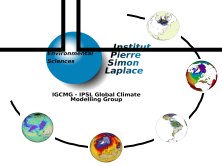
Calcul



Machine de post-traitement



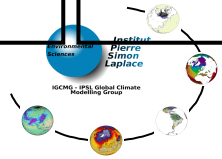
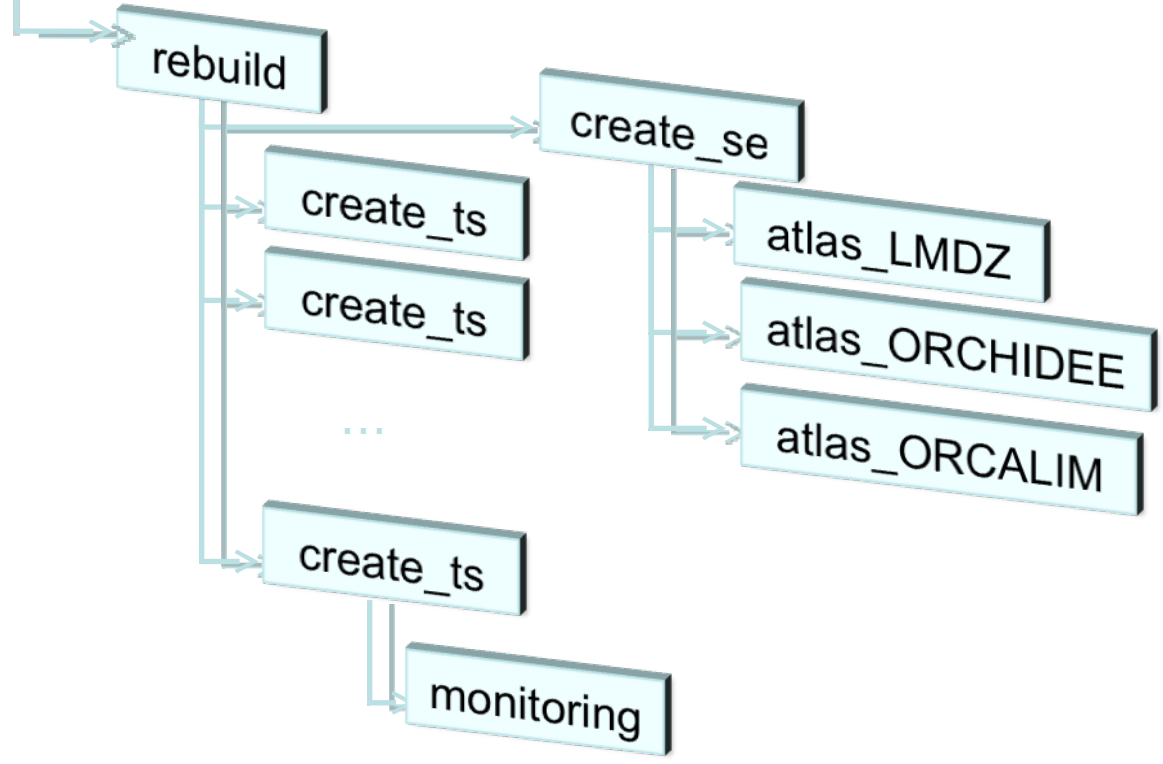
Génération de moyennes saisonnières.



Calcul



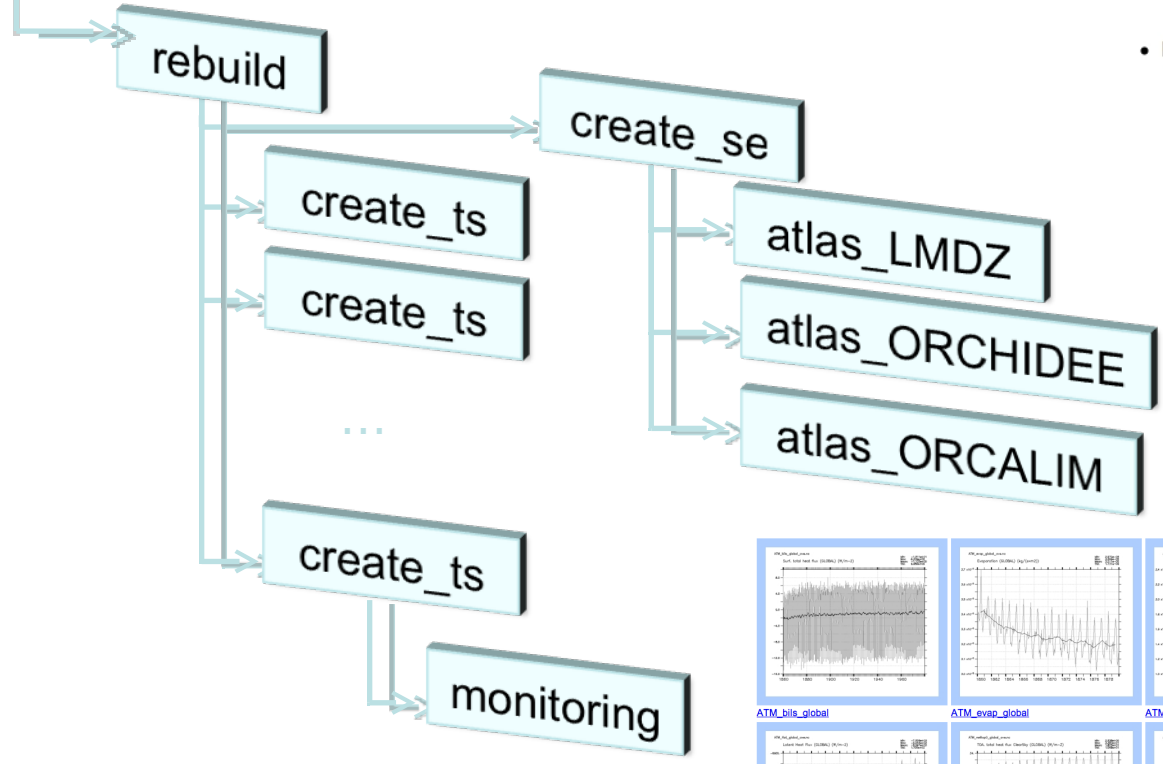
Machine de post-traitement



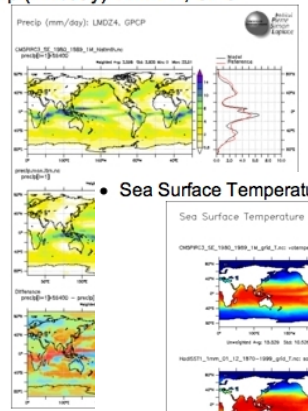
Calcul



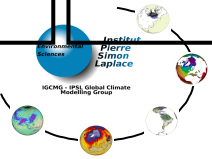
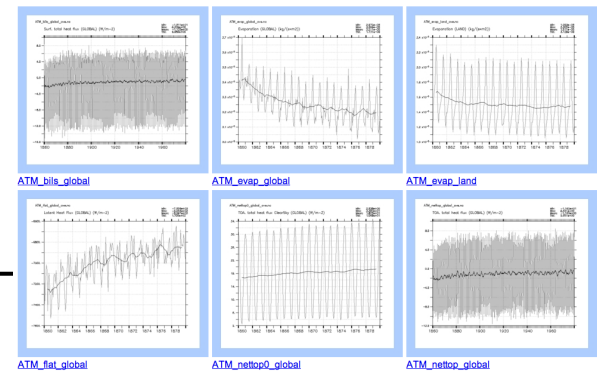
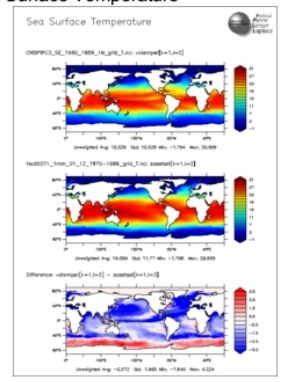
Machine de post-traitement



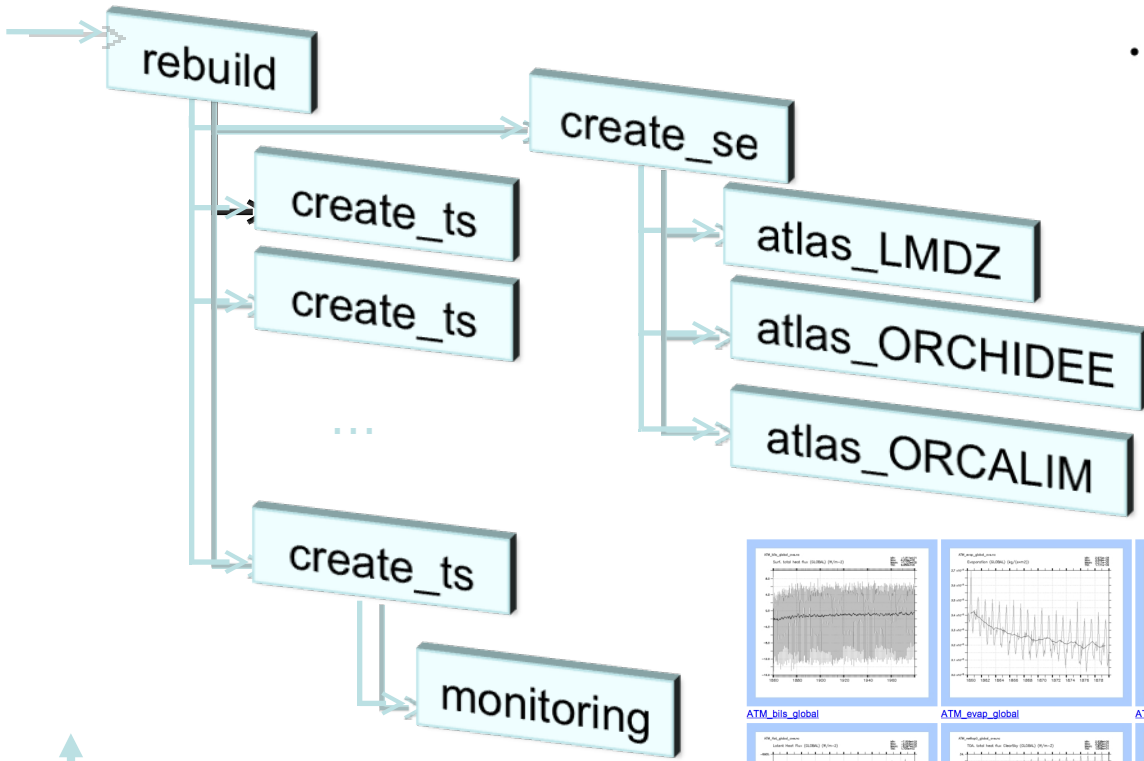
• Precip (mm/day): LMDZ4, GPCP



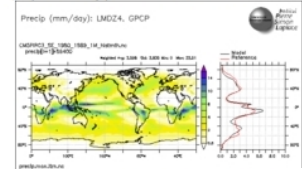
• Sea Surface Temperature



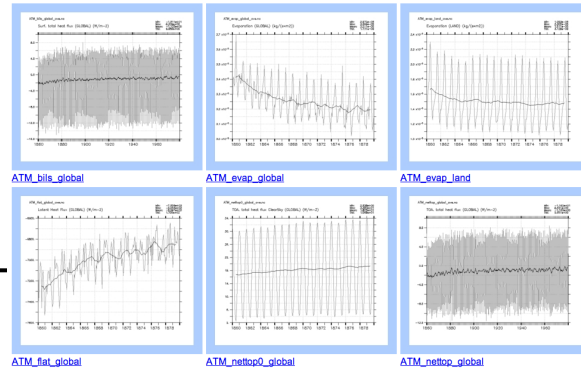
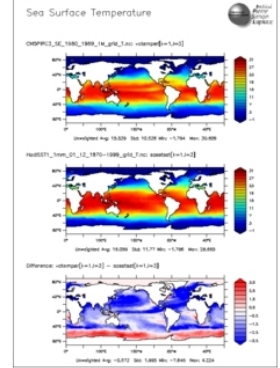
Machine de post-traitement



• Precip (mm/day): LMDZ4, GPCP



• Sea Surface Temperature

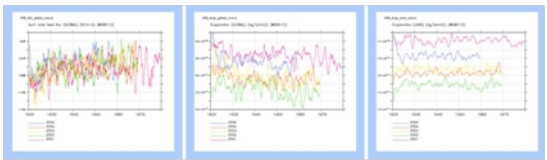
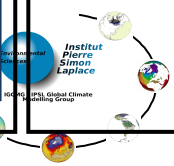


Archive

Fichiers bruts

Fichiers post-traités et analyses

Web



Fichiers post-traités et analyses

# Les utilitaires de post-traitement CCRT, IDRIS

MY\_EXPERIENCE

**create\_ts.job** : séries temporelles tous les 10 ans

**create\_se.job** : moyennes saisonnières tous les 10 ans

**Retour des jobs de post-traitement là :**

**ulam** : \$WORKDIR/IGCM\_OUT/IPSLCM5A/*JobName*

**mercure** : \$SCRATCHDIR/IGCM\_OUT/IPSLCM5A/*JobName*

**atlas\_ORCA\_LIM** : pour océan et glace de mer

**atlas\_LMDZ** : pour atmosphère

**atlas\_ORCHIDEE** : pour surfaces continentales

Les atlas sont basés sur *ferret* et sur *fast* :

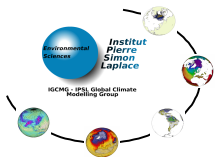
<http://dods.ipsl.jussieu.fr/fast/>

<http://wiki.ipsl.jussieu.fr/IGCMG/Outils/ferret/60MinutesAvecFerret>

modipsl

**libIGCM**

- **create\_ts**
- **create\_se**
- **atlas\_...**
- **monitoring**
- **clean\_month**



# Message en fin de simulation

A la fin d'une expérience, vous recevrez un message de ce type :

Dear *login*,

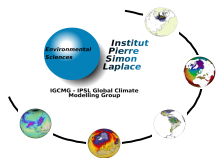
Simulation EXP00 is finished on supercomputer brodie03.

Job started : 20000101

Job ended : 20001231

Output files are available in

.../IGCM\_OUT/IPSLCM5A/DEVT/pdControl/EXP00



# Comment vérifier que cela s'est bien passé?

- **run.card** :
  - PeriodState=Completed
- Message de fin de simu reçu
- Fichiers sur le serveur de fichiers
- Post-traitements lancés puis finis
- ATLAS et monitoring sur serveur dods

MY\_EXPERIENCE

|  
modipsl

|  
config

|  
IPSLCM5A

|  
**EXP00**

Script\_Output

**run.card**

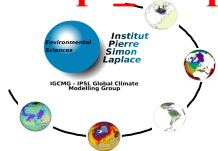
run.card.init

config.card

Job\_EXP00

COMP

PARAM



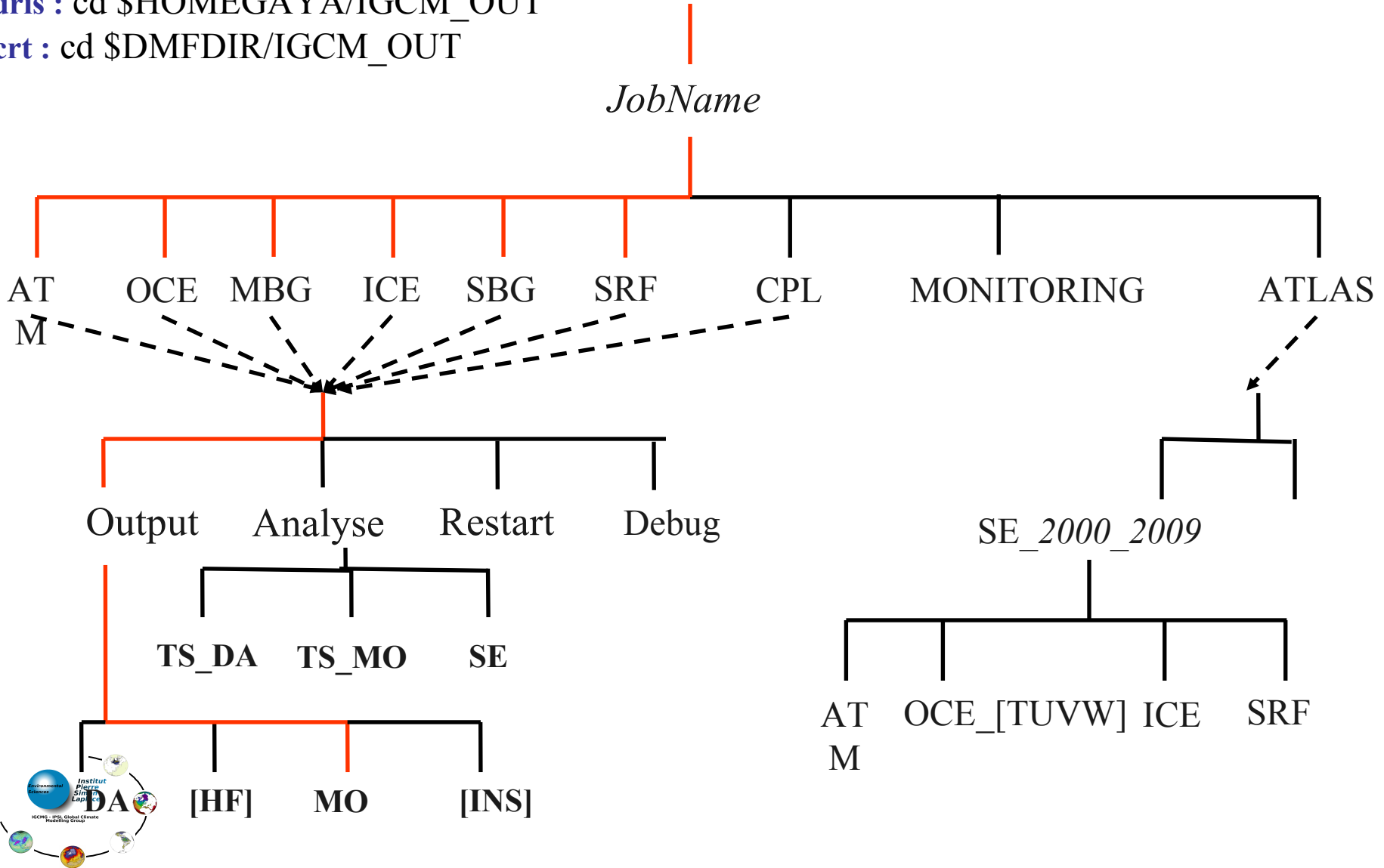


# Arborescence sur serveur fichiers

IPSLCM5A/DEVT/pdControl

Idris : cd \$HOMEGAYA/IGCM\_OUT

ccrt : cd \$DMFDIR/IGCM\_OUT



# Arborescence sur serveur fichiers

```
idris : cd $HOMEGAYA
```

```
cert : cd $DMFDIR
```

```
IGCM_OUT/
```

```
`-- IPSLCM5A/DEVT/pdControl
```

```
  |-- JobName
```

```
    |-- ATLAS
```

```
    |-- ATM
```

```
      |-- Analyse
```

```
        |-- SE
```

```
        |-- TS_DA
```

```
        |-- TS_HF
```

```
        |-- `-- TS_MO
```

```
      |-- Debug
```

```
      |-- Output
```

```
        |-- DA
```

```
        |-- `-- HF
```

```
        |-- `-- MO
```

```
      |-- `-- Restart
```

```
    |-- CPL
```

```
      |-- Analyse
```

```
        |-- `-- SE
```

```
      |-- Debug
```

```
      |-- Output
```

```
        |-- MO
```

```
        |-- `-- Restart
```

```
      |-- Exe
```

```
    |-- ICE
```

```
      |-- Analyse
```

```
        |-- SE
```

```
        |-- `-- TS_MO
```

```
      |-- Debug
```

```
      |-- Output
```

```
        |-- `-- MO
```

```
      |-- `-- Restart
```

```
  |-- MONITORING
```

```
  |-- OCE
```

```
    |-- Analyse
```

```
      |-- SE
```

```
      |-- `-- TS_MO
```

```
    |-- Debug
```

```
    |-- Output
```

```
      |-- DA
```

```
      |-- `-- MO
```

```
    |-- `-- Restart
```

```
  |-- Out
```

```
  |-- SRF
```

```
    |-- Analyse
```

```
      |-- SE
```

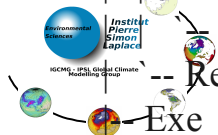
```
      |-- `-- TS_MO
```

```
    |-- Debug
```

```
    |-- Output
```

```
      |-- `-- MO
```

```
    |-- `-- Restart
```



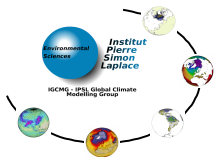
# Accès aux résultats de simulations

- Mise en ligne du monitoring et des atlas sur les serveurs dods :

dods IDRIS : <http://dods.idris.fr/monlogin>

dods CCRT : <http://dods.extra.cea.fr/data/monlogin>

- Accès aux simulations de référence
  - Ensembles : <http://mc2.ipsl.jussieu.fr/ensembles.html>
- Site de transition CMIP5 en cours de mise en route
- ESG (projet Prodiguer)



# Nomenclature des noms des fichiers de sortie

## Output, Analyse, Debug, ...

$\{\text{JobName}\}_{\_}\{\text{PeriodDateBegin}\}_{\_}\{\text{PeriodDateEnd}\}_{\_}\text{XX}_{\_}\text{NomFichier}$

Output/**DA** et Analyse/TS\_**DA**:

XX       $\longrightarrow$       **1D**

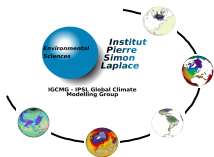
Output/**MO** et Analyse/TS\_**MO**

XX       $\longrightarrow$       **1M**

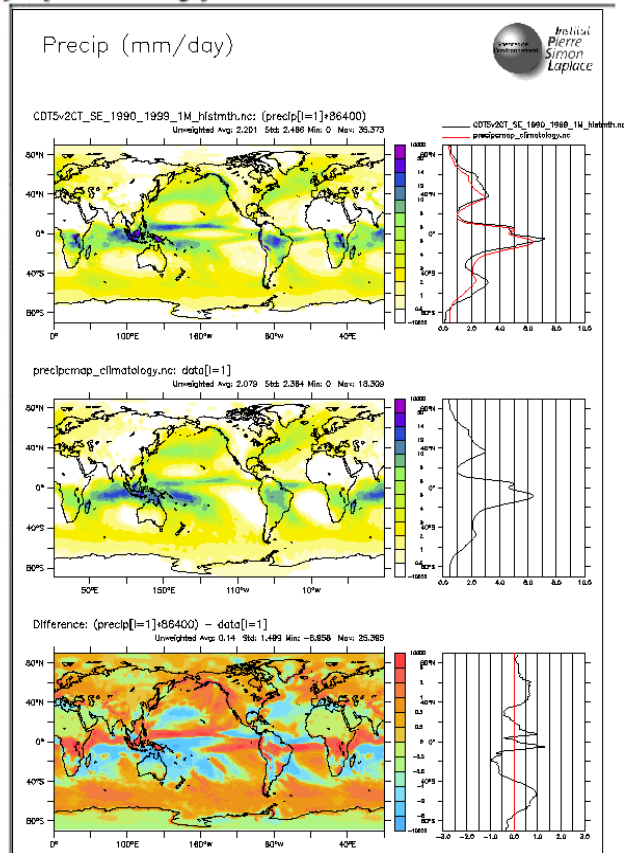
## Analyse/SE :

$\{\text{JobName}\}_{\_}\text{SE}_{\_}\{\text{PeriodDateBegin}\}_{\_}\{\text{PeriodDateEnd}\}_{\_}\text{NomFichier}$

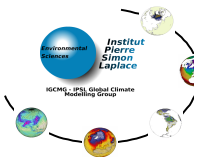
**Restart** :  $\{\text{JobName}\}_{\_}\{\text{PeriodDateEnd}\}_{\_}\text{NomFichier}$



- Atm
  - [Atm.pdf](#) (25328 Ko)
- Precip (mm/day)



- [precip\\_13185/precip.gif](#) (608 Ko)
- [precip\\_13185/precip.pdf](#) (1088 Ko)



# pdControl\_EXP00 monitoring

at 2010-04-22 17:08:37

Cards Analysis Monitoring Board About

ALL

Filter : \*

Images : 077 / 077

ATM

CHM

ICE

MBG

OCE

SBG

SRF

XOR

CLR

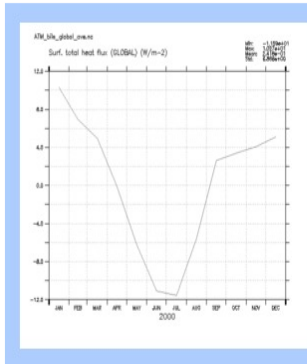
land

ocean

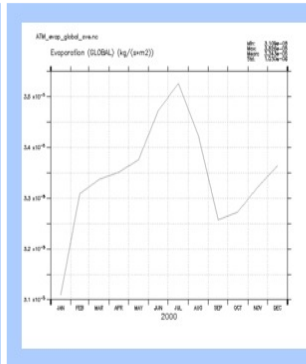
north

south

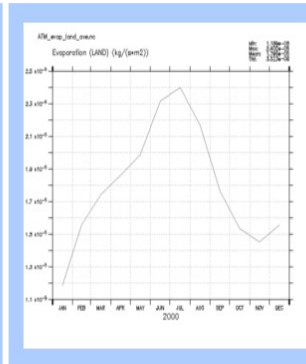
global



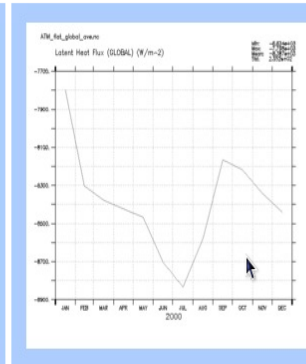
ATM\_bils\_global



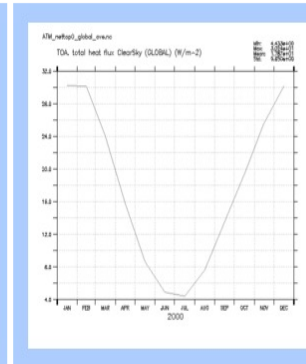
ATM\_evap\_global



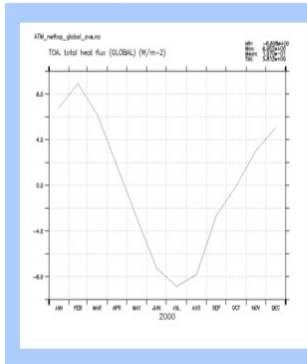
ATM\_evap\_land



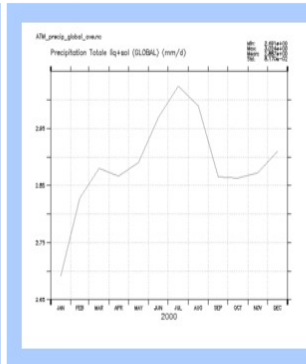
ATM\_flat\_global



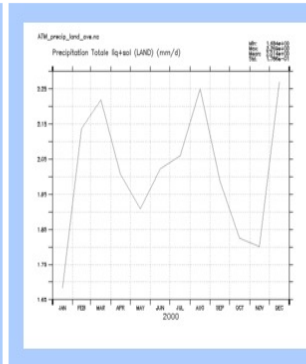
ATM\_nettop0\_global



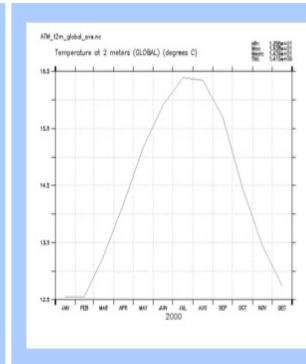
ATM\_nettop\_global



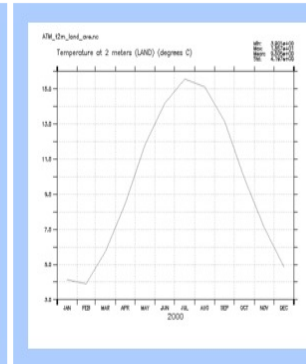
ATM\_precip\_global



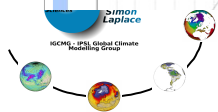
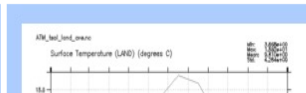
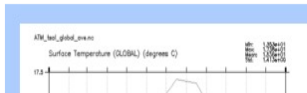
ATM\_precip\_land



ATM\_t2m\_global



ATM\_t2m\_land








# IGCMG Web services : <http://igcmg.lsce.ipsl.fr/>

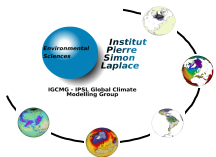
**Welcome to the IGCMG Web Services portal**



---

## Services

-  [Trusting Web Service](#)
-  [Machine Load Status](#)
-  [Inter Monitoring Web Service](#)
-  [Meta Atlas Web Service](#)
-  [Metrics Web Service](#)



# Trusting Web Service

## Trusting

Release: 0.60

Check reliability of configurations on the SX9 from the CCRT.

[Ask for support](#)  
[Discover other services](#)

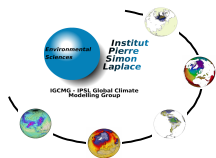
## Configuration IPSLCM5\_v2

2010-06-07T06:10

IOIPSL/src	svn checkout -r HEAD http://forge.ipsl.jussieu.fr/igcmg/svn/IOIPSL/tags/v2_1_9/src IOIPSL/src
ORCHIDEE	cvs -d :pserver:sechiba@cvs.ipsl.jussieu.fr:/home/ssipl/CVSREP checkout -r orchidee_1_9_4_2 ORCHIDEE
OASIS3	cvs -d :pserver:anonymous@cvs.ipsl.jussieu.fr:/home/ioipsl/CVSRROOT checkout -r ipslcm5a -d prism OASIS3
LMDZ4	svn checkout -r 1329 http://svn.lmd.jussieu.fr/LMDZ/LMDZ4/trunk LMDZ4
IPSLCM5	svn checkout -r HEAD http://forge.ipsl.jussieu.fr/igcmg/svn/CONFIG/IPSLCM/IPSLCM5/branches/IPSLCM5_v2 IPSLCM5
libIGCM	svn checkout -r 246 http://forge.ipsl.jussieu.fr/libigcm/svn/trunk/libIGCM libIGCM
NEMO	svn checkout -r HEAD http://forge.ipsl.jussieu.fr/nemo/svn/tags/nemo_v3_2/NEMO
UTIL	svn checkout -r HEAD http://forge.ipsl.jussieu.fr/nemo/svn/tags/nemo_v3_2/UTIL
XMLF90	svn checkout -r 54 http://forge.ipsl.jussieu.fr/ioserver/svn/XMLF90
XMLIO_SERVER	svn checkout -r 54 http://forge.ipsl.jussieu.fr/ioserver/svn/XMLIO_SERVER/trunk XMLIO_SERVER

## Trusting log

Date ▾	Status	Step	Comments	C++	F90	MPI	CROSSKIT	NETCDF	IOIPSL/src	ORCHIDEE	OASIS3	LMDZ4	IPSLCM5	libIGCM	NEMO	UTIL
2010-06-07T06:10	OK	Code is reliable	-	087	400	8.0.4	18.1/3	3.6.1	HEAD 740	orchidee_1_9_4_2 2010-05-27T10:30:35	ipslcm5a 2010-03-18T15:16:18	1329	HEAD 951	246	HEAD 1779	HEAD 1773
2010-06-06T06:10	OK	Code is reliable	-	087	400	8.0.4	18.1/3	3.6.1	HEAD 740	orchidee_1_9_4_2 2010-05-27T10:30:35	ipslcm5a 2010-03-18T15:16:18	1329	HEAD 951	246	HEAD 1779	HEAD 1773
2010-06-05T06:10	OK	Code is reliable	-	087	400	8.0.4	18.1/3	3.6.1	HEAD 740	orchidee_1_9_4_2 2010-05-27T10:30:35	ipslcm5a 2010-03-18T15:16:18	1329	HEAD 951	246	HEAD 1779	HEAD 1773
2010-06-04T06:10	OK	Code is reliable	-	087	400	8.0.4	18.1/3	3.6.1	HEAD 740	orchidee_1_9_4_2 2010-05-27T10:30:35	ipslcm5a 2010-03-18T15:16:18	1329	HEAD 951	246	HEAD 1779	HEAD 1773



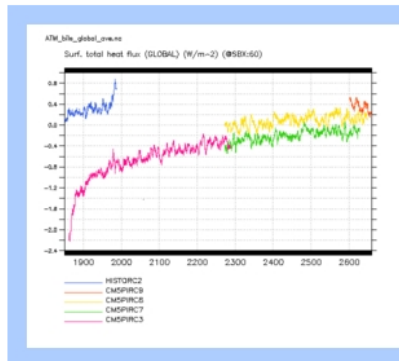


## Monitoring comparison: HISTORC2 vs CM5PIRC9 vs CM5PIRC8 vs CM5PIRC7 vs CM5PIRC3

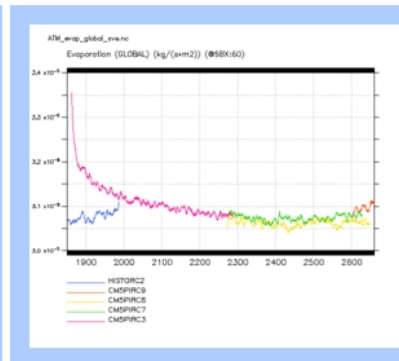
at 2010-05-03 11:33:30

ALL Filter :  Images : 070 / 070

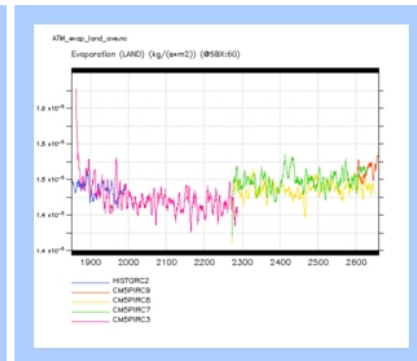
ATM	CHM	ICE	MBG	OCE	SBG	SRF	<b>XOR</b>	CLR
land	ocean	north	south	global				



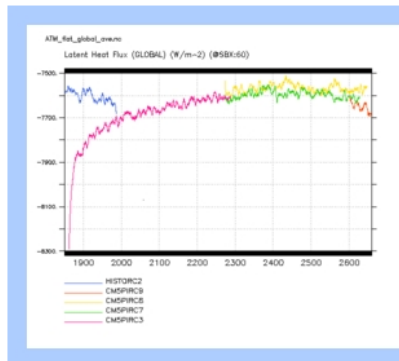
[ATM\\_hfl\\_global\\_ave](#)



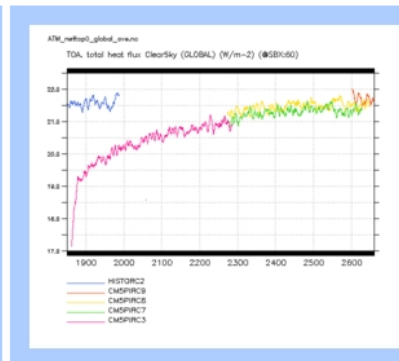
[ATM\\_evap\\_global\\_ave](#)



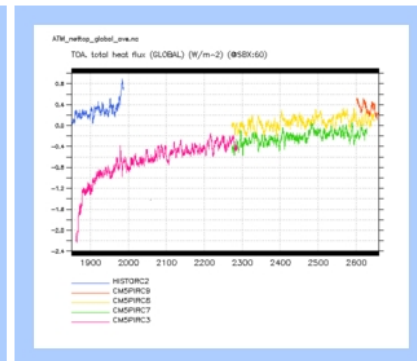
[ATM\\_evap\\_land\\_ave](#)



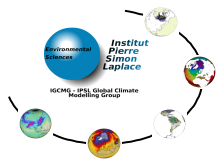
[ATM\\_flat\\_global\\_ave](#)



[ATM\\_nettop0\\_global\\_ave](#)



[ATM\\_nettop\\_global\\_ave](#)



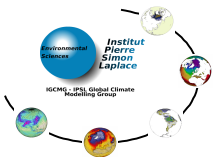
# Questions fréquentes sur les simulations

- Comment écraser une simulation existante ?
  - Effacer le fichier run.card
  - Effacer le répertoire d'output sur le disque de stockage
  - Soumettre
- Comment continuer une simulation ?
  - Vérifier que le fichier run.card est prêt pour la période suivante

Ex: on vient de finir le mois de janvier 2001 et on se prépare pour février

```
OldPrefix= EXP00_20000131  
PeriodDateBegin= 2000-02-01  
PeriodDateEnd= 2000-02-28  
PeriodState=OnQueue
```

**Attention** : si vous continuez une simulation qui a buggué, il faut vérifier qu'elle n'a pas créé de fichiers pour le mois que vous allez relancer (utilisation du script `clean_month` dans libIGCM)



# Comment savoir où est le problème ?

Si dans **run.card** : `PeriodState=Fatal`

--> Regarder le fichier **Script\_Outputxxxx**

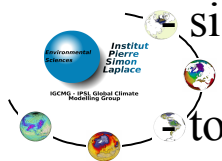
```
#####  
#          ANOTHER GREAT SIMULATION          #  
#####  
1ère partie  
#####  
#          DIR BEFORE RUN EXECUTION          #  
#####  
2ème partie  
#####  
#          DIR AFTER RUN EXECUTION          #  
#####  
3ème partie
```

## Erreurs dans la première partie - différentes pistes:

- si on commence une nouvelle simulation : est-ce qu'il y avait ou non un `run.card` ?

si on continue une simulation : a-t-on bien mis `PeriodState=OnQueue` dans le `run.card`

tous les fichiers d'input existent-ils ?



# Comment savoir où est le problème ?

## Erreurs dans la deuxième partie :

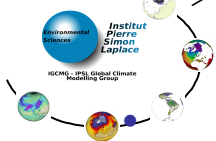
- a-t-on demandé assez de temps d'exécution ?
- a-t-on demandé assez de mémoire ?
- si on a le message suivant :

```
=====
EXECUTION of : mpirun -f ./run_file > out_run_file 2>&1
Return code of executable : 1
IGCM_debug_Exit : EXECUTABLE

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!! IGCM_debug_CallStack !!
!-----!

!-----!
IGCM_sys_Cp : out_run_file xxxxxxxxxxxx_out_run_file_error
=====
```

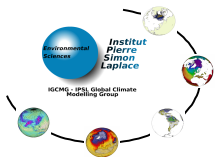
-> analyser le fichier xxxxxxxxxxxx\_out\_run\_file\_error



# Comment savoir où est le problème ?

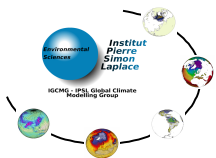
## Erreurs dans la troisième partie :

- Si on écrase une simulation a-t-on bien effacé les fichiers de sorties de la simulation d'origine ?
- Si il n'y a pas de fichier de restart, vérifier qu'il n'y ait pas une erreur dans la deuxième partie
  - Si il manque un fichier de restart et pourtant vous n'avez pas de message d'erreur dans la seconde partie vous êtes peut être tombés sur un garde-fou (dans LMDZ par ex). Il faut alors relancer la simulation en indiquant `RUN_DIR_PATH=$SCRATCHDIR` dans `Job_jobname` et ensuite analyser le fichier `out_gcm.e` dans le répertoire créé dans le `$SCRATCHDIR`.
- Tout s'est bien passé mais les post-traitements n'ont pas fonctionné --> relancer les post-traitements



# Comment savoir où est le problème ?

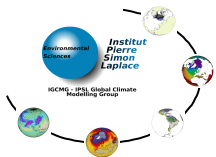
- Regarder le fichier **xxxxx\_error** dans le répertoire de soumission
  - Contient le texte de sortie de LMDZ  
LMDZ s'arrête souvent dans hgardfou  
**Stopping in hgardfou**
  - Contient les **erreurs brutales** de toutes les composantes
- Regarder le fichier texte de sortie de NEMO
  - Serveur de fichiers
  - **EXP00/OCE/Debug/EXP00\_xxxxxx\_ocean.output**
  - Les erreurs NEMO sont synthétisées à la fin.
- Regarder les fichiers texte de sortie de ORCHIDEE
  - Serveur de fichiers
  - **EXP00/SRF/Debug/EXP00\_xxxxxx\_output\_orchidee\_0000**  
, ... \_0003



# Lancer les post-traitements à postériori

Remarque : Tous les fichiers devant être reconstruits sont sauvegardés

- Si `RebuildFromArchive=.true.` : dans `IGCM_OUT/.../EXP00/TMP/`
  - Si `RebuildFromArchive=NONE` : sur le `$$SCRATCHDIR` au `ccrt` et sur le `$WORKDIR` à l'idris (attention aux quotas limités)
- 
- Si ce sont les rebuilds qui n'ont pas fonctionné vous pouvez les relancer et ils lanceront automatiquement les TS et SE en découlant
  - Si ce sont les TS ou les SE vous pouvez les relancer sans passer par les rebuilds.
  - Vous pouvez également utiliser l'outil `TimeSeries_Checker` qui vérifie les séries temporelles existantes et relance les jobs `create_TS` nécessaires pour reconstruire les TS manquantes.
- > toutes les explications sont sur la page `ModipslBeginner`

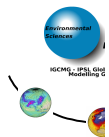


<http://forge.ipsl.jussieu.fr/igcmg/wiki/ModipslBeginner>

## MODIPSL pour les débutants

MODIPSL est l'interface d'accès aux modèles de l'IPSL. Cette page résume ce qu'un nouvel utilisateur de MODIPSL doit connaître.

Cette page veut rassembler les informations pour les nouveaux utilisateurs et pointer vers les pages plus complètes de chaque configuration ou de chaque outil quand nécessaire.



[MODIPSL for beginner](#)

[Environnements de calculs](#)

[IDRIS](#)

[CCRT](#)

[Extraire modipsl](#)

[Mode utilisateur](#)

[Mode administrateur](#)

[Pour en savoir plus sur SVN](#)

[Présentation des répertoires de modipsl](#)

[Liste des configurations disponibles via modipsl](#)

[Travailler avec une configuration choisie](#)

[Extraction](#)

[Mots de passe](#)

[FCM](#)

[Compilation](#)

[Spécificité SX9](#)

[Toutes machines](#)

[A qui signaler quand cela ne marche pas?](#)

[Options de compilations de LMDZ](#)

[Lancer une simulation](#)

[Cas général](#)

[Présentation du répertoire d'expérience](#)

[Etapas avant la création du job de simulation](#)

[Création du job](#)

[Temps d'exécution du couplé IPSLCM5\\_v2](#)

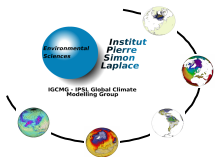
[Exécution de la simulation](#)



# Contributions

L'ensemble du groupe de travail Plate-forme :

[esci@ipsl.jussieu.fr](mailto:esci@ipsl.jussieu.fr)



# Annexe :Caractéristiques d'un fichier NetCDF

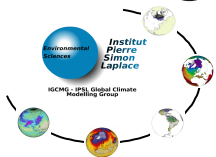
**Auto descriptif** ➡ Le fichier contient l'information sur les variables contenues

**Portable** ➡ Fichiers accessibles par des machines ayant des modes différents de stockage des entiers, des caractères et des nombres à virgules flottantes

**à Accès direct** ➡ Possibilité d'accéder à une donnée sans avoir à parcourir l'ensemble des données qui la précède

**Modifiable** ➡ Possibilité d'ajouter des données dans un fichier

**Partageable** ➡ Possibilité d'avoir simultanément un accès en écriture et plusieurs accès en lecture



# NetCDF, nco, cdo Convention CF

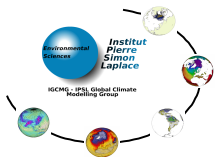
Netcdf : <http://www.unidata.ucar.edu/packages/netcdf/>

nco : <http://nco.sourceforge.net/>

cdo : <http://www.mpimet.mpg.de/fileadmin/software/cdo/>

Convention CF :

<http://www.cgd.ucar.edu/cms/eaton/cf-metadata/>



# Structure du fichier NetCDF – En-tête

- Informations sur les dimensions



```
dimensions:  
  lon = 72 ;  
  lat = 46 ;  
  presnivs = 19 ;  
  time_counter = UNLIMITED ; // (1 currently)
```

- Informations sur les attributs

(voir conventions CF)



```
// global attributes:  
  :Conventions = "GDT 1.3" ;  
  :file_name = "histmth.nc" ;  
  :production = "An IPSL model" ;  
  :TimeStamp = "2003-MAR-05 10:37:38 GMT+0100" ;  
  :associate_file = "dyn_hist_ave.nc dynzon.nc histhf.nc  
histmth.nc sechiba_out.nc cpl_atm_tauflx.nc cpl_atm_sst.nc" ;
```

**ncdump -h**  
**COURS\_1m\_19790101\_197901**  
**30\_histmth.nc**

- Informations sur les attributs des

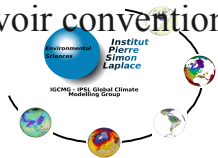
variables

( sans leurs valeurs)

(voir conventions CF)



```
variables:  
  float lon(lon) ;  
    lon:units = "degrees_east" ;  
    lon:valid_min = -180.f ;  
    lon:valid_max = 175.f ;  
    lon:long_name = "Longitude" ;  
    lon:nav_model = "Default grid" ;  
  float lat(lat) ;  
    lat:units = "degrees_north" ;  
    lat:valid_min = -90.f ;  
    lat:valid_max = 90.f ;  
    lat:long_name = "Latitude" ;  
    lat:nav_model = "Default grid" ;  
  float presnivs(presnivs) ;  
    presnivs:units = "mb" ;  
    presnivs:positive = "unknown" ;  
    presnivs:valid_min = 388.2433f ;  
    presnivs:valid_max = 100426.5f ;  
    presnivs:title = "presnivs" ;  
    presnivs:long_name = "Vertical levels" ;  
  float time_counter(time_counter) ;  
    time_counter:units = "seconds since 1979-01-01 00:00:00" ;  
    time_counter:calendar = "360d" ;  
    time_counter:title = "Time" ;  
    time_counter:long_name = "Time axis" ;  
    time_counter:time_origin = " 1979-JAN-01 00:00:00" ;  
  float tsol(time_counter, lat, lon) ;  
    tsol:units = "K" ;  
    tsol:missing_value = 1.e+20f ;  
    tsol:valid_min = 1.e+20f ;  
    tsol:valid_max = -1.e+20f ;  
    tsol:long_name = "Surface Temperature" ;  
    tsol:short_name = "tsol" ;  
    tsol:online_operation = "ave(X)" ;  
    tsol:axis = "TYX" ;  
    tsol:interval_operation = 1800.f ;  
    tsol:interval_write = 2592000.f ;  
    tsol:associate = "time_counter nav_lat nav_lon" ;
```



# Structure du fichier NetCDF - Données

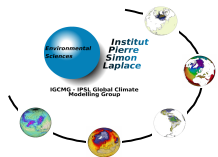
- données de taille fixe
- données de taille variable



data:

```
tsol =  
  246.818, 246.818, 246.818, 246.818, 246.818, 246.818,  
  246.818, 246.818,  
    246.818, 246.818, 246.818, 246.818, 246.818,  
  246.818, 246.818, 246.818,  
    246.818, 246.818, 246.818, 246.818, 246.818,  
  246.818, 246.818, 246.818,  
    246.818, 246.818, 246.818, 246.818, 246.818,  
  246.818, 246.818, 246.818,  
    246.818, 246.818, 246.818, 246.818, 246.818,  
  246.818, 246.818, 246.818,  
    246.818, 246.818, 246.818, 246.818, 246.818,  
  246.818, 246.818, 246.818,  
    246.818, 246.818, 246.818, 246.818, 246.818,  
  246.818, 246.818, 246.818,  
    248.3489, 248.3532, 248.3445, 248.003, 247.5628,  
  247.1862, 246.7824,
```

...



# Utilitaires nco

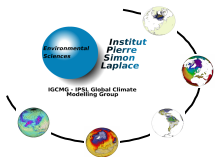
**ncdump** : génère sur la sortie standard une représentation textuelle CDL d'un ensemble de meta-données netCDF avec la possibilité d'exclure tout ou partie de données variables. La sortie de ncdump doit pouvoir servir d'entrée à ncgen.

**ncgen** : génère un fichier netCDF ou un programme C ou FORTRAN permettant de créer un fichier netCDF

ncdump et ncgen peuvent donc être utilisées comme fonctions inverses pour passer d'une représentation textuelle à une représentation binaire et inversement.

## Exemple :

```
> ncdump -p15 -b f  
    COURS_1m_19790101_19790130_histmth.nc >  
    COURS_1m_19790101_19790130.cdl  
> emacs COURS_1m_19790101_19790130.cdl &  
> ncgen -o COURS_1m_19790101_19790130.nc  
    COURS_1m_19790101_19790130.cdl
```

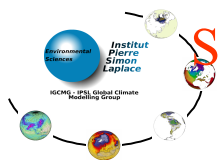


# Utilitaires nco

**ncdiff** soustrait les variables d'un fichier file\_1 à celles d'un fichier file\_2 correspondantes et stocke les résultats dans un fichier file\_3.

**ncrcat** concatène des variables enregistrées parmi un nombre arbitraire de fichiers d'entrée. La dimension du fichier netCDF de sortie est par défaut la somme des dimensions des fichiers netCDF d'entrée. Les fichiers d'entrée peuvent avoir des tailles différentes mais tous doivent avoir des dimensions spécifiées. L'enregistrement des coordonnées doit avoir la même syntaxe.

Exemple : `ncrcat -v tsol COURS_1m_19790[1-9]01_19790[1-9]30_histmth.nc COURS_1m_19791[0-2]01_19791[0-2]30_histmth.nc COURS_1m_19880[1-9]01_19880[1-9]30_histmth.nc COURS_1m_19881[0-2]01_19881[0-2]30_histmth.nc COURS_1m_19790101_19880130_TSOL.nc`



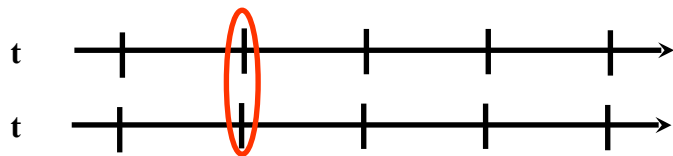
Série temporelle de la variable TSOL sur 10 ans

# Utilitaires nco

**ncra** calcule la moyenne sur un nombre variable de fichiers d'entrée. C'est une moyenne temporelle sur la grille spatiale. Ce qui donne 1 seule valeur dans les fichiers de sorties. **ncra** ne calcule pas de moyenne pondérée.

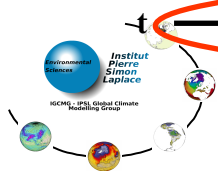
**ncea** calcule la moyenne sur un nombre variable de fichiers d'entrée. C'est une moyenne spatiale sur la grille temporelle. Ce qui donne autant de valeurs moyennes que de pas de temps.

**ncea** fait la moyenne « fichier à fichier » sur chaque point de l'axe des temps



Nombre de valeurs de moyenne égale au nombre de sorties par fichiers

**ncra** fait la moyenne « fichier à fichier » sur l'ensemble des points de l'axe des temps



1 unique valeur de moyenne



# Utilitaires nco

**ncks** permet d'extraire une série de données qu'il écrit sur la sortie standard sous forme ASCII (comme **ncdump**) et qu'il écrit également sous forme d'un fichier binaire netCDF

Exemple :

```
ncks -v sosstsst COURS_1m_19790101_19790130_grid_T.nc  
      COURS_1m_19790101_19790130_SOSSTSST.nc
```

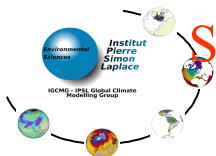
...

```
ncks -v sosstsst COURS_1m_19881201_19881230_grid_T.nc  
      COURS_1m_19881201_19881230_SOSSTSST.nc
```

```
ncrcat -v sosstsst COURS_1m_19790[1-9]01_19790[1-9]30_grid_T.nc  
      COURS_1m_19791[0-2]01_19791[0-2]30_grid_T.nc  
      COURS_1m_19880[19]01_19880[1-9]30_grid_T.nc  
      COURS_1m_19881[0-2]01_19881[02]30_grid_T.nc  
      COURS_1m_19790101_19880130_grid_T.nc
```

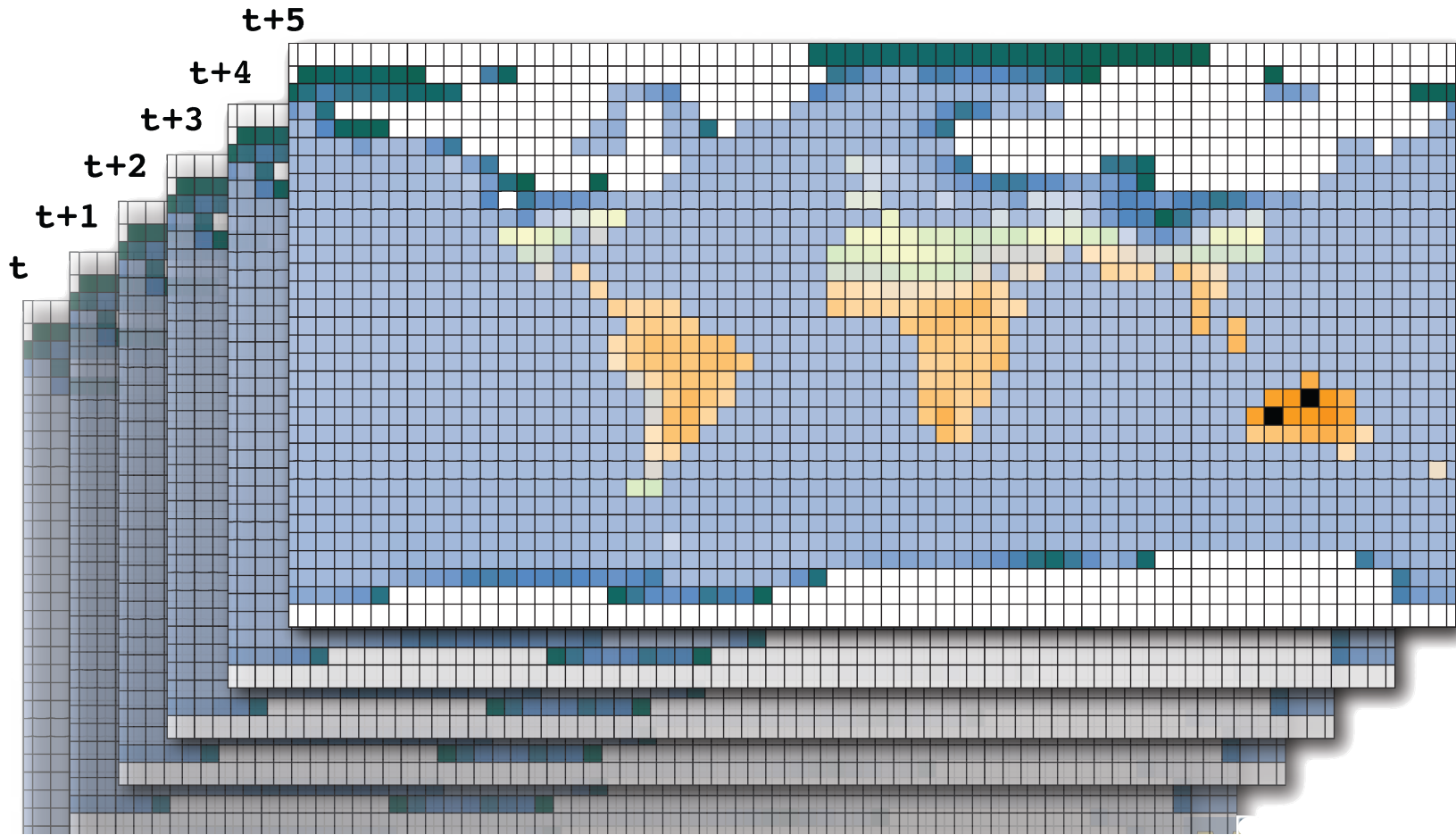


Série temporelle de la variable SOSSTSST sur 10 ans



# Utilitaires cdo (Climate Data Operator)

2d fields timeseries



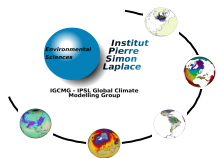
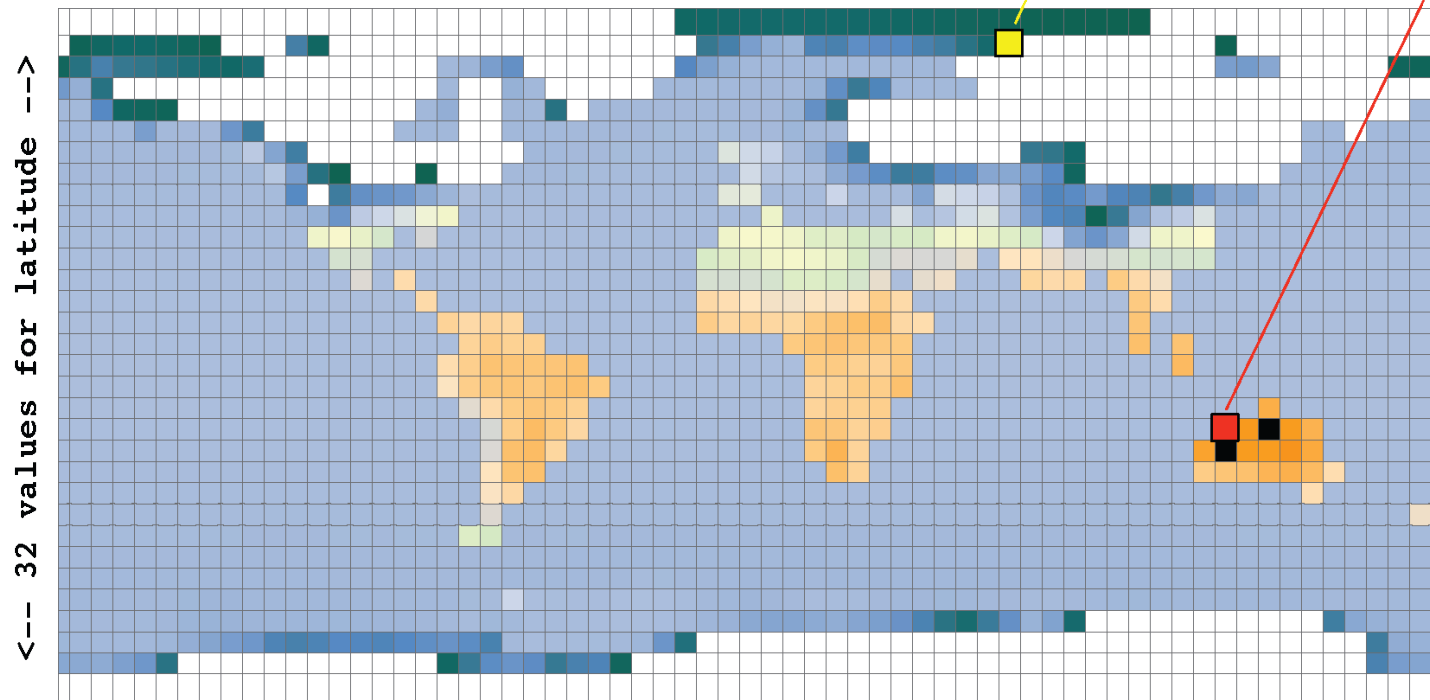
# Utilitaires cdo (Climate Data Operator)

```
tiny 23%cdo info tsurf.1.nc
```

-1 :	Date	Time	Code	Level	Size	Miss :	Minimum	Mean	Maximum
1 :	1978-01-02	00:00	169	0	2048	0 :	226.98	268.46	311.08

cdo info : Processed 1 variable 1 timestep. ( 0.00s )

<-- 64 values for longitude -->



# Utilitaires cdo (Climate Data Operator)

- File information (info, sinfo, diff, diffv, ...)
- File operations (copy, cat, merge, split, ...)
- Selection (selcode, selvar, sellevel, seltimestep, ...)
- Missing values (setctomiss, setmisstoc, setrtomiss)
- Arithmetic (add, sub, mul, div, ...)
- Mathematical functions (sqrt, exp, log, sin, cos, ...)
- Comparison (eq, ne, le, lt, ge, gt, ...)
- Conditions (ifthen, ifnotthen, ifthenc, ifnotthenc)
- Field statistic (fldsum, fldavg, fldstd, fldmin, fldmax, ...)
- Vertical statistic (vertsum, vertavg, vertstd, vertmin, ...)
- Time range statistic (timavg, yearavg, monavg, dayavg, ...)
- Ensemble statistic (enssum, ensavg, ensstd, ensmin, ...)
- Regression (detrend)
- Field interpolation (remapbil, remapcon, remapdis, ...)
- Vertical interpolation (ml2pl, ml2hl)
- Time interpolation (inttime, intyear)



# Job\_ *JobName* : PBS tableau des classes\* IDRIS

## brodie : news class

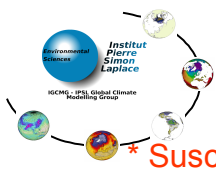
```
=====  
Classes multiprocesseurs (<=8) au sein d'un noeud (MPI ou OpenMP)  
=====
```

Parametres NQSII a specifier :

```
#PBS -q multi
```

```
#PBS -l cpunum_job=<Nproc> # Nombre de processeurs (1 <= Nproc <= 8)
```

```
      ^ -l cputim_job (limite en temps CPU par job)  
      |  
12:00:00 +-----+  
  (12H) |           |  
        |           | p2t2  
        |           | 1 <= Nproc <= 2  
        |           | TMPDIR <= 45Gb  
1:00:00 +-----+  
  (1H)  |           |  
        |           | p2t1  
        |           | 1 <= Nproc <= 2  
        |           | TMPDIR <= 45Gb  
+-----+-----+--> -l memsz_job (limite memoire par job)  
                               15Gb
```

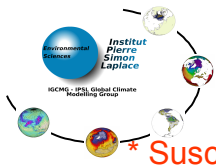


\* Susceptible de changement permanent

# Job *JobName* : PBS tableau des classes\* IDRIS

brodie : news class (suite)

```
^ -l cputim_job
|
48:00:00 +-----+
(48H)
|
|           p8t2
|
|       7 <= Nproc <= 8
|       TMPDIR <= 300Gb
|
2:00:00 +-----+
(2H)
|
|           p8t1
|
|       7 <= Nproc <= 8
|       TMPDIR <= 100Gb
|
+-----> -l memsz_job
|
|           60Gb
```



\* Susceptible de changement permanent

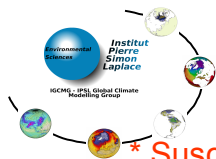
# Job\_JobName : PBS

## tableau des classes\* CCRT

### mercure : class

Classe de soumission:

Queue	Act	Ena	Nod	Cpu	Mem	Time	Type	RL	URL	UAL	Hosts
scal-new	ENA	ACT	1	1	8G	24h	-	-	-	-	mercure02,mercure03
scalaire	ENA	ACT	1	1	8G	24h	-	-	-	-	mercure
admin-sx	ENA	ACT	8	8	52G	Inf	-	-	-	-	mercure10,mercure11,... mercure17
admin-sx	ENA	ACT	3	16	962G	Inf	-	-	-	-	mercure20,mercure21,mercure22
para- <b>sx9</b>	ENA	ACT	3	16	962G	24h	Normal	-	-	-	mercure20,mercure21,mercure22
prio- <b>sx9</b>	ENA	ACT	3	16	962G	24h	Special	-	-	-	mercure20,mercure21,mercure22
test- <b>sx9</b>	ENA	ACT	3	16	962G	1h	Urgent	-	-	1	mercure20,mercure21,mercure22
bigmem	ENA	ACT	1	1	64G	24h	Normal	-	-	2	mercure10,mercure11
bigtime	ENA	ACT	1	1	32G	100h	Normal	-	-	2	mercure12,mercure13,... mercure16
parallel	ENA	ACT	4	8	40G	24h	Normal	-	-	2	mercure10,mercure11,... mercure16
prod	ENA	ACT	1	1	20G	24h	Normal	-	-	12	mercure10,mercure11,... mercure17
test	ENA	ACT	1	1	32G	1h	Urgent	8	1	-	mercure17
testpara	ENA	ACT	4	8	52G	0.5h	Urgent	1	1	-	mercure10,mercure11,... mercure16



\* Susceptible de changement permanent