Technical Reports

CENTRO EURO-MEDITERRANEO
PER I CAMBIAMENTI CLIMATICI

ANS – Numerical Applications and Scenarios Division

CMCC-SXF025: A High-Resolution Coupled Atmosphere Ocean General Circulation Climate Model

Enrico Scoccimarro
Istituto Nazionale di Geofisica e Vulcanologia ((INGV) and Numerical Applications and Scenarios Division, CMCC

Silvio Gualdi
Istituto Nazionale di Geofisica e Vulcanologia ((INGV) and Numerical Applications and Scenarios Division, CMCC

Alessio Bellucci
Istituto Nazionale di Geofisica e Vulcanologia ((INGV) and Numerical Applications and Scenarios Division, CMCC

Andrea Carril
Istituto Nazionale di Geofisica e Vulcanologia ((INGV) and Numerical Applications and Scenarios Division, CMCC

Pier Giuseppe Fogli
Istituto Nazionale di Geofisica e Vulcanologia ((INGV) and Numerical Applications and Scenarios Division, CMCC

Antonio Navarra
Istituto Nazionale di Geofisica e Vulcanologia ((INGV) and Numerical Applications and Scenarios Division, CMCC

Centro Euro-Mediterraneo
per i Cambiamenti Climatici
www.cmcc.it

April 2007 ■ TR3

CMCC-SXF025:

A High-Resolution Coupled Atmosphere Ocean General Circulation Climate Model

Summary

This technical report summarizes the SINTEX-F025 (CMCC-SXF025) model technical structure. CMCC-SXF025 is an Atmosphere Ocean General Circulation Model (AOGCM) developed at INGV-CMCC to perform global seasonal forecasting.

This model is used to run special focus experiments in the framework of MERSEA (Marine EnviRonment and Security for the European Area EU-Project).

CMCC-SXF025 is an evolution of SINTEX and SINTEX-F and this report indicates the improvement with respect to these previous INGV AOCM models. The new model includes an ocean model with higher resolution: 0.25 degrees as horizontal resolution and 46 vertical levels. The description of model components, coupling methods, compiling and running environments is a guideline for model users.

Keywords: General Circulation Models, Coupling, Message Passing Interface.

We *wish to thank all the other INGV-CMCC people who helped to assemble this model: Andrea Alessandri, Annalisa Cherchi, Simona Masina, Elisa Manzini and Marcello Vichi.*

***Address for correspondence*:**

Enrico Scoccimarro, Istituto Nazionale di Geofisica e Vulcanologia

Via D. Creti, 12 - 40128 Bologna, Italy – E-mail: scoccimarro@bo.ingv.it

Index

Introduction 5

1 CMCC-SXF025: Technical description 5

1.1 Model components 5

1.1.1 Atmosphere 5

1.1.2 Ocean 7

1.1.3 Coupler 8

1.2 Domain decomposition 8

1.3 Coupling method 9

1.4 Code structure 11

1.4.1 Compiling environment 11

1.4.1.1 Coupler component source 12

1.4.1.2 Atmospheric component source 13

1.4.1.3 Compiling scripts 13

1.4.1.4 Ocean component source 13

1.4.1.5 Namelists repository 14

1.4.1.6 Binaries repository 14

1.4.1.7 Experiments environment 14

1.4.2 Running environment 14

2 Inputs 17

2.1 Restarts and auxiliary files 17

2.1.1 Atmospheric component input files 17

2.1.2 Ocean component input files 17

2.1.3 Coupler component input files 18

3 Outputs 20

3.1 Atmospheric model output 20

3.2 Ocean model output 20

3.3 Coupler and standard output 21

Bibliography 23

Figures 26

Appendix A 29

Appendix B 36

Introduction

This report summarizes the technical structure and the characteristics of the new global coupled climate model developed at the Centro Euro-Mediterraneo per i Cambiamenti Climatici (INGV-CMCC), with the aim to perform special focus experiments in the framework of MERSEA (Marine EnviRonment and Security for the European Area EU-Project, Proposal/Contract no.: SIP3-CT-2003-502885) – workpackage 11 for global seasonal forecasting.

The model, named SINTEX-F025 (CMCC-SXF025) (figure 1), is an evolution of the SINTEX and SINTEX-F models [Gualdi et al., 2003a, 2003b; Guilyardi et al., 2003, Luo et al. 2003].

The ocean component in SINTEX-F025 is the same used in SINTEX-F (OPA8.2), but with enhanced horizontal and vertical resolution (figure 1). Due to the partial SPMD (Single Program Multiple Data) parallelization of the OPA8.2 model, it has been necessary to modify the code in order to perform fields gathering and scattering during the coupling phase with the atmospheric model.

1. CMCC-SXF025 technical description

1.1 Model Components

The model is composed of three parts: Atmosphere, Ocean and Coupler.
All the components are described in the following sections.

1.1.1 Atmosphere

The atmospheric component is ECHAM4 [Roeckner, 1996]. The release 4.6 used, is the MPI (Message Passing Interface) parallelized version.

- Resolution:

  Horizontal res: Gaussian grid at triangular truncation T106 (about 1.125 ^{o}x1.125 ^{o}).

  Vertical res: 19 hybrid sigma-pressure levels; top level at 10 hPa; 7 layers above 200 hPa, 5 layers below 850 hPa.

- Numerical scheme/grid:

  semi-implicit leap-frog time stepping.

- List of prognostic variables: vorticity, divergence, temperature, log surface pressure, water vapour, mixing ratio of total cloud water.

- Major parameterizations:

  a. clouds: Sundquist [Sundquist, 1978] type prognostic scheme for stratiform fractional clouds; optical cloud properties and cloud water determined by Mie theory [Rockel et al., 1991; Roeckner, 1995];

  b. tracer advection: transport of water vapour, cloud water, and (optionally) tracers by a semi-lagrangian scheme [Williamson and Rasch, 1994];.

  c. convection: shallow, mid-level, and deep cumulus convection with Tiedke [Tiedke, 1989] mass flux scheme and adjustment closure for deep convection as described by Nordeng [Nordeng, 1996];

  d. boundary layer and vertical diffusion: surface fluxes of momentum, heat, water vapour, and cloud water calculated with Monin-Obukhov theory [Luis, 1979], with eddy diffusivity coefficients depending on roughness length and Richardson No.; above the surface layer, the coefficients depend on wind shear, thermal stability, and mixing length;

  e. SW radiation: Fouquart and Bonnel[Fouquart and Bonnel, 1980], LW radiation: Morcrette et al. [Morcrette et al., 1986] including methane, nitrous oxide, and 16 CFC species, ozone (14.6 μm), and various types of aerosols (optional) effects; revised water vapor continuum [Giorgetta and Wild, 1995];

f. Orographic Gravity Wave Drag.

1.1.2 Ocean

The ocean component is OPA 8.2 [Madec et al., 1999] in ORCA025 configuration: 0.25° longitude x 0.05° - 0.25° latitude.

- Resolution:

Horizontal res: quasi-isotrope tri-polar grid (2 poles in the northern hemisphere, one over Canada and the other over Siberia. 0.25° resolution Mercator grid with enhanced meridional resolution in the proximity of the equator and in Med and Red seas (0.05°);

Vertical res: 31 vertical levels with 14 levels lying in the top 150 meters.

- Numerical scheme/grid:

advection scheme : 2nd order Arakawa

time-stepping: leap-frog except for lateral diffusion (forward) and vertical diffusion (backward)

vertical coordinate : z-coordinate; free surface.

- List of prognostic variables and tracers: horizontal velocity components, temperature, salinity and kinetic energy.

- Major parameterizations:

a. eddy parameterization;

Isopycnal mixing on tracers (no horizontal background) with a constant coefficient of 500 m^2/s. Eddy induced velocity with a coefficient varying in function of the growth rate of baroclinic instability (ranges 15 m^2/s to 500 m^2/s). Note that the coefficient is set to 0 in the vicinity of the equator;

b. bottom boundary layer treatment and/or sill overflow treatment diffusive bottom boundary layer [Beckmann and Dorscher 1997];

c. mixed-layer treatment;

TKE scheme [Blanke and Delecluse JPO, 1993 + modification in Madec et al. 1999];

d. sunlight penetration with 2 master lengths [Blanke and Delecluse 1993];

e. no tidal mixing;

f. no river mouth mixing;

g. mixing isolated seas with the ocean: no mixing (Red and Med seas are explicitly connected to the remaining ocean) . For closed "seas" (Black Sea, Great lakes, Caspian Seas) the mean sea level remain constant, excess (deficit) of water been either redistributed over the world ocean (Caspian Sea) or in St Laurent river mouth (Great lakes) or Dardanel strait area (Black Sea);

h. treatment of North Pole "singularity": semi analytical tri-polar grid, no singular point in the ocean domain [Madec and Imbar,1996; Murray, 1996].

All the description of the model physics can be found in the OPA reference manual [Madec at al 1999] available on the web (http://www.lodyc.jussieu.fr/opa/).

1.1.3 Coupler

The software used to couple ocean and atmospheric components is OASIS 2.4 [Valke, 2000]. OASIS is a complete, self-consistent and portable set of Fortran 77, Fortran 90 and C routines divided into a main library, interpolation libraries and communication libraries. Its main tasks are the synchronization of the models being coupled, and the treatment and interpolation of the fields exchanged between the models. All the coupling options of the simulation are defined using an input file *namecouple* (see appendix A) that uses free formatting.

1.2 Domain decomposition

The number of processes (in UNIX sense) involved in an CMCC-SXF025 simulation is M=S+1+N: S process for the ocean model, one process for the coupler and N processes for the atmospheric model. S is equivalent to the number of boxes in which we decide to

decompose the 2D spatial domain in the ocean model; N is equivalent to the number of boxes in which we decide to decompose the domain in the atmospheric model. The parallelization of both OPA8.2 and ECHAM4.6 code is based on a domain decomposition approach. Every process only handles a limited domain of the full globe and only keeps the respective part of the data (Single Program Multiple Data paradigm).

Further explanations about the echam 4.6 domain decomposition can be found in Scoccimarro et al., 2007.

In order to avoid gathering/scattering at each model time step, the OPA 8.2 code is designed to run only with ghost cells sending/receiving, starting from one restart file and saving one model output file for each process (total number of processes = S) involved in the ocean simulation. This is performant in terms of CPU time due to the reduction of the MPI send/receive operation number. Inserting this ocean model in a coupled system it has been necessary to implement the OPA8.2 code part regarding the coupling phase: in order to send/receive the global domain to/from the atmosphere, at each coupling time step (the same described for the atmosphere in Scoccimarro et al, 2007), the fields must be gathered/scattered from/to all the ocean processes.

The code parts involved in this operation can be highlight looking for

“!E+ 06/07 coupling mod ” comment lines.

1.3 Coupling method

Only one process for each model component (i.e. 3 processes) is involved in the coupling: the master process of the ocean model gathers (scatters) all the domain subsets from (to) the other processes before (after) communicate with the coupler process. The same for the atmospheric model.

The communication between models can be driven in different ways (in OASIS sense). In CMCC-SXF025 we use the CLIM technique, a MPI2 (Message Passing Interface 2) based synchronization and exchange method.

The models are integrated in parallel. The communication between models occurs, through the coupler, at each coupling time step (120 minutes) which correspond to 10 time step for the ocean model and 10 time steps for the atmospheric model. (figure 2).

The fields exchanged between ocean and atmosphere have to be insert in the namcouple coupler namelist, coherently with the model codes. The routines modified into each model, involved in the coupling phase are the following:

ocean model:	stpcmo.F
	flx.coupled.h
	tau.coupled.h
atmospheric model:	fromcpl.F
	intocpl.F

During the coupling phase a large number of operations can be made by the coupler on each field, as specified in the namecouple namelist (see appendix A), in order to optimize the interpolation at each coupling time step (figure 3) between the different grids of the two models. As mentioned before, the two grids are different both in type and resolution.

In order to peform the coupling operations, the following auxiliaries files have been built:

- grids: grid coordinates (latitude and longitude) of all the models.
- masks: land-sea masks of all the models.
- areas: surface area of the grid cells of all the models.
- mozaic_weights: weights needed by the mozaic interpolation technique (the one used to interpolate the fields sent from the ocean to the atmosphere).

For the interpolation from the ocean grid to the atmospheric grid, we use an area-averaging method (activated by the keyword 'MOZAIC' in the OASIS control file namecouple). The only field sent from the ocean model to the atmospheric model is the Sea Surface Temperature (SST).

The fields sent from the atmospheric model to the ocean model are the following:

- Solar Heat flux
- Non Solar Heat Flux
- Water Flux
- Zonal Wind Stress
- Meridional Wind Stress

Figure 4 shows the operations performed by the coupler on each field (see OASIS 2.4 documentation, [Valke, 2000]) in the CMCC-SXF025 model: preprocessing and interpolation flags. The coupling scheme do not use any flux adjustment.

1.4 Code structure

1.4.1 Compiling environment

The CMCC-SXF025 code is divided in 7 main sections:

1. Coupler component source (.../SINTEXF025/modipsl/modeles/CPL)
2. Atmospheric component source (.../SINTEXF025/modipsl/modeles/ECHAM4.6)
3. Compiling scripts (.../SINTEXF025/modipsl/modeles/SINTEXF025)
4. Ocean component source (.../SINTEXF025/modipsl/modeles/OPA/POG05B)
5. Namelists repository (.../SINTEXF025/modipsl/modeles/SINTEXF025/scripts)
6. Binaries repository (.../SINTEXF025/modipsl/bin)
7. Experiment environment (.../SINTEXF025/F01_025)

The next paragraphs show the tree of the main directories in each section and the files modified in the CMCC-SXF025 model with respect to the SINTEXF version of the

components code (files are in italic); only for the Ocean component source the comparison is not referred to the SINTEX-F model, but to the OPA8.2 in standalone ORCA025 configuration. The model is designed to run on NEC-SX6 vector super computer and the compilation can be done natively on the NEC-SX6 or using NEC cross-compiler on machines mounting the NEC-SX6 filesystem.

To compile the code, login .../SINTEXF025/modipsl/modeles/SINTEXF025 directory and tape gmake (or the cross-compiler command; $SXGMAKE is the environment variable to use on refosco SUN machine located in Bologna INGV CED). The Makefile will call sequentially the CPL, ECHAM4.6 and OPA Makefiles to obtain 3 executables (atmosx opa025 and oasis, collected in the .../SINTEXF025/modipsl/bin directory. Each Makefile will compile the relative component code.

The common routines coming from the coupler (.../SINTEXF025/modipsl/modeles /CPL/lib/clim/src_MPI2/*) needed by the other components, are included as libraries in case of OPA, and recompiled from local identical routines (.../ECHAM4.6/libclim/*) in case of ECHAM4.6.

1.4.1.1 Coupler component source

.../SINTEXF025/modipsl/modeles/CPL

|-- include

parameter.h

|-- lib

| |-- clim

clim.h

| | |-- src_MPI2

| -- src

Makefile

1.4.1.2 Atmospheric component source

.../SINTEXF025/modipsl/modeles/ECHAM4.6

|-- libclim

Makefile

inicma.F

|-- modules

|-- src

Makefile

1.4.1.3 Compiling scripts

.../SINTEXF025/modipsl/modeles/SINTEXF025

Makefile

1.4.1.4 Ocean component source

../SINTEXF025/modipsl/modeles/OPA/POG05B

common.h

dtasst.forced.h

dtrlec.F

flx.coupled.h

flxrnf.h

inicmo.F

parameter.h

stpcmo.F

tau.coupled.h

Makefile_ea_couple

|-- IOIPSL

1.4.1.5 Namelists repository

...SINTEXF025/modipsl/modeles/SINTEXF025/scripts

namcouple.base_orca025_coup

namatmos.base

namelist_orca025_coup

1.4.1.6 Binaries repository

.../SINTEXF025/modipsl/bin

atmosx

opa025

oasis

1.4.1.7 Experiments environment

.../SINTEXF025/F01_025

run.couple

F01_025.trace

1.4.2 Running environment

The experiment directory (section 1.4.1.7) contains the scripts to run a CMCC-SXF025 simulation. A restart procedure is applied to avoid too long jobs on the NEC queue. The model is designed to run 1 month of simulation for each subjob submission: a subjob is intended as a job subsample and a job as an entire simulation to perform.

The information regarding the period to integrate is stored in the text file named F01_025.trace file which contains: year, month and day of the beginning of the subjob and

number of months that have already been done in the job. The *run.couple* script, submitted to the NEC-SX6 nqs-queue system through the qsub command, executes the following steps:

- Set the environment variables related to:
  - input and output files path for all the components
  - MPI model name
  - experiment name
  - number of processes to use in OPA 8.2 and ECHAM4.6 code
  - experiment date and duration (using as input the *F01_025.trace* file) of subjob (here 1 month) and job (the entire simulation) for the entire job and for the single subjob
- Modify the namelists according to the job to run:
  - *namelist* for the ocean component
  - *namatmos* for the atmospheric component
  - *namcouple* for the coupler
- Get the executables and input files in the scratch directory where the subjob will run, according to the information stored in the *F01_025.trace* file
- Launch the model
- Save the output and restart files in the storage directories
- Update the *F01_025.trace* file for the next subjob
- Resubmit the *run.couple* script to perform the next subjob (month) (until the end of the job).

The executable launched by the run.couple at launching time is oasis, the coupler executable. oasis will start the other 2 executables, involved in the coupling: opa025 and atmosx. In case of parallel use of the OPA8.2 and ECHAM4.6 code, opa025 and atmosx will start other (S-1)+(N -1) executables (see 1.2 paragraph).

To run the ECHAM4.6 code with a different number of processes, inicmo.F and inicma.F routines must be changed: mpi_totproc(2) must be equal to the new atmosx processes

number; also the run.couple run script has to be modified indicating the new NPROCB number of processes.

To run the OPA8.2 code with a different number of processes, inicmo.F and inicma.F routines must be changed: mpi_totproc(1) must be equal to the new opa025 processes number. In the parameter.h routine, the jpnj and jpnij parameters must be equal to the processes number. Also the run.couple run script has to be modified indicating the new NCPUSORCA number of processes.

Be sure to set the right CPUs number (#@$-c)in the queue requirement setting in the run.couple script.

2. Inputs

2.1 Restarts and auxiliary files

The input files needed to perform the subjob relatives to April 1992 are described in the next sections; the input files relative to other periods are similar but with other time slices and names. In order to reduce the files number to list, the " ? " symbol in a file name indicates that S (number of ocean processes) files are considered, one for each process.

2.1.1 Atmospheric component input files

- *namatmos.base*: atmospheric model namelist
- *year106m.nc*: climate sea surface temperature field
- *F01_025_00920430_unit.31*: restart unit.31 (ECHAM4.6 type) history file
- *F01_025_00920430_unit.32*: restart unit.32 (ECHAM4.6 type) history file
- *F01_025_00920430_unit.35*: restart unit.35 (ECHAM4.6 type) history file
- *F01_025_00920430_unit.36*: restart unit.36 (ECHAM4.6 type) history file
- *F01_025_00920430_unit.37*: restart unit.37 (ECHAM4.6 type) history file

2.1.2 Ocean component input files

- *namelist_orca025_coup:* *ocean model namelist*
- *bathymetry_ORCA_R025.ascii:* *ocean bathymetry file*
- *coordinates_00?.nc:* *Grid coordinates files*
- *maskglo_float.nc:* *global land sea mask*
- *Levitus98_PHC21_01_Sal_ORCA_R025.nc:* *climatologic 3d monthly salinity file. [Levitus, 1982]*
- *Levitus98_PHC21_01_Tem_ORCA_R025.nc:* *climatologic 3d monthly temperature file[Levitus, 1982].*

- *runoff_1m_ORCA_R025.nc :* *Climatologic river run-off file.*
- *F01_025_00920430_restart_00?.nc* *. Restart history files.*
- *ECMWF_SST_POG_15459_15465.nc*
- *ECMWF_SST_POG_15466_15472.nc*
- *ECMWF_SST_POG_15473_15479.nc*
- *ECMWF_SST_POG_15480_15486.nc*
- *ECMWF_SST_POG_15487_15493.nc*
- *ECMWF_SST_POG_15494_15500.nc* *ECMWF SST files: 1 week. Daily files. 15459 indicate the starting day and 15465 indicate the last day of the week.*

To run 1 month of simulation (1 subjob) six ECMWF weekly files are needed in order to cover the entire period 15465:15493. The two weeks before and after the period are in addition in order to provide the SST fields for the interpolation in time needed for the first and last time steps of the subjob.

2.1.3 Coupler component input files

- *namcouple.base_orcao25_coup:* *coupler namelist*
- *areas_025_T106:* *areas file (see 1.3 paragraph)*
- *grids_025_T106:* *grids file (see 1.3 paragraph)*
- *masks_025_T106_mod:* *masks file (see 1.3 paragraph)*
- *mozaic_orca025_T106_mod:* *mozaic ocean to atmosphere file*
- *F01_025_00920430_flxatmos:* *atmospheric fluxes restart history file*
- *F01_025_00920430_sstocean* *sea surface temperature fluxes restart history file*
- *julday* *scripts to compute the date, called by*
- *caldat* *the script run.couple*

- *F01_025.trace* *subjob date beginning file (see 1.4.2 paragraph)*

3. Outputs

In order to reduce the files number to list, the " ? " symbol in a file name indicates that S (number of ocean processes) files are considered, the same method used for the input file description.

3.1 Atmospheric model output

- *atm_F01_025_00920501_00920530.grib:* 12h grib variables output file
- *F01_025_00920530_unit.31*: restart unit.31 (ECHAM4.6 type) history file
- *F01_025_00920530_unit.32*: restart unit.32 (ECHAM4.6 type) history file
- *F01_025_00920530_unit.35*: restart unit.35 (ECHAM4.6 type) history file
- *F01_025_00920530_unit.36*: restart unit.36 (ECHAM4.6 type) history file
- *F01_025_00920530_unit.37*: restart unit.37 (ECHAM4.6 type) history file
- *F01_025_00920501_00920530_atm.prt*: coupling information log file.

3.2 Ocean model output

- *F01_025_1mAV_9205_9206_2D_T_00?.nc:* Monthly NetCDF grid T variables output files
- *F01_025_1mAV_9205_9206_grid_U_00?.nc:* Monthly NetCDF grid U variables output files
- *F01_025_1mAV_9205_9206_grid_V_00?.nc:* Monthly NetCDF grid V variables

output files

- *F01_025_1mAV_9205_9206_grid_W_00?.nc:* Monthly NetCDF grid W variables output files
- *F01_025_1mAV_9205_9206_S3D_T_00?.nc:* Monthly NetCDF grid T 3D Salinity variables output files
- *F01_025_1mAV_9205_9206_T3D_T_00?.nc:* Monthly NetCDF grid T 3D Temperature variables output files
- *F01_025_00920530_restart_00?.nc:* NetCDF restarts files
- *F01_025_00920501_00920530_oce.output_0?:* ocean model log files
- *F01_025_00920501_00920530_oce.prt* coupling information log file.

3.3 Coupler and standard output

- *F01_025_00920501_00920530_cplout*: coupler log file
- *F01_025_00920501_00920530_Oasis.prt* coupling information log file
- *F01_025*.oXXXX running standard output (include run.couple stdout and atmospheric stdout)

All these files are stored coherently with the output path setting defined in run.couple script (see 1.4.2 paragraph). The standard output F01_025.oXXXX (XXXX is a progressive queue job number) is stored in the experiment (i.e. .../SINTEXF025/F01_025) directory, together with the updated F01_025.trace file (see 1.4.2 paragraph).

The atmospheric output file is in GRIB (GRIdded Binary) format [WMO, 1994]. In order to obtain selected variables in NetCDF format [Rew et al. 2006], it is necessary to postprocess these .grib files with the AFTERBURNER (http://www.mpimet.mpg.de/fileadmin/software/afterburner/) software. The total amount of data produced for each subjob is about 2GB, dependent on the number of variable saved.

To create the output network Common Data Form Climate and Forecast Metadata Conventions (NetCDF-CF) see http://www.cgd.ucar.edu/cms/eaton/cf-metadata).

Bibliography

Beckmann, A. and Dorscher, (1997): A method for improved representation of dense water spreading over topography in geopotential-coordinate models. J. Phys. Oceanogr., 27,581-591

Blanke, B. and Delecluse, P., (1993): Variability of the tropical Atlantic Ocean simulated by a general circulation model with two different mixed-layer physics. J. Phys. Oceanogr., 23, 1363–1388.

Fouquart, Y. and B. Bonnell, (1980): Computations of solar heating of the Earth's atmosphere. Beitr. Phys. Atmos., 53, 35-62.

Giorgetta, M. and M. Wild, (1995): The water vapor continuum and its representation in ECHAM4. MPI for Meterolo., Report No. 162, Hamburg, 38 pp.

Gualdi, S., Guilyardi, E., Navarra, A., Masina, S. and Delecluse, P., (2003a): The interannual variability in the tropical Indian Ocean as simulated by a CGCM, Clim. Dyn., 20, 567–582.

Gualdi, S., Navarra, A., Guilyardi, E., and Delecluse, P., (2003b): Assessment of the tropical Indo-Pacific climate in the SINTEX CGCM, Ann. Geophys., 46, 1 – 26.

Guilyardi, E., and Madec, G, (1997): Performance of the OPA/ARPEGE-T21 global ocean-atmosphere coupled model. Clim. Dyn., 13, 149-165.

Guilyardi, E., Delecluse, P., Gualdi, S. and Navarra, A., (2003): Mechanisms for ENSO phase change in a coupled GCM, J. Clim., 16, 1141–1158.

Levitus, S., (1982): Climatological atlas of the World Ocean, NOAA Prof. Paper 13, US Government Printing Office, Washington DC, USA.

Luis, J. F., (1979): A parametric model of vertical eddy fluxes in the atmosphere. Boundary Layer Meteorology, 17, 187-202.

Luo, J.-J., Masson, S., Becera, S., Delecluse, P., Gualdi, S., Navarra, A., and Yamagata, T., (2003): South Pacific origin of the decadal ENSO-like variation as simulated by a coupled GCM. Geophys. Res. Lett., 30(24), 2250.

Madec, G., and Imbard, I., (1996): A global ocean mesh to overcome the North Pole singularity. Climate Dyn, 12, 381–388.

Madec, G., Delecluse, P., Imbard, I. and Levy, C., (1999), OPA 8.1 Ocean General Circulation Model reference manual, Note du Pôle de modélisation,, Inst. Pierre-Simon Laplace (IPSL), France, No. 11, 91 pp.

Morcrette, J.-J., Smith, L. and Fouquart, Y., (1986): Pressure and temperature dependence of the absorption in longwave radiation parameterizations. Beitr. Physic Atmos., 59, 691-708.

Murray, R. J., (1996): Explicit generation of orthogonal grids for ocean models. J. Comput. Phys, 126, 251–273.

Nordeng, T.E., (1996): Extended versions of the convective parameterization scheme a ECMWF and their impact on the mean and transient activity of the model in the tropics. Technical Memorandum 206, ECMWF, Reading, UK.

Rew, R., Davis, G., Emmerson, S., Davies, H. and Hartnett, E., (2006): The NetCDF User's Guide, Version 3.6.1, Unidata Program Center, 62.p.

Rockel, B., Raschke, E. and Weyres, B., (1991): A parameterization of broad-band radiative transfer propertiesnof water, ice and mixed clouds. Beitr. Physik Atmos., 64, 1-12.

Roeckner, E., (1995): Parameterization of cloud radiative properties in the ECHAM4 model. In: Proc. of the WCRP Workshop on "Cloud Mikrophysics Parameterizations in Global Atmospheric Circulation Models", May 23-25, Ananaskis, Alberta, Canada, WCRP-Report No. 90, 105-116, WMO/TD-No. 713.

Roeckner, E., Arpe, K., Bengtsson, L., Christoph, M., Claussen, M., Dümenil, L., Esch, E., Giorgetta, M., Schlese, U. and Schulzweida, U., (1996): The atmospheric general circulation model ECHAM-4: Model description and simulation of present-day climate. Reports 218, Max-Planck-Institute for Meteorology, Hamburg, 90 pp.

Scoccimarro, E., Gualdi, A., Fogli, B., Manzini, E., Grezio, A., Navarra, A., (2007): INGV-SXG: A coupled atmosphere ocean sea-ice general circulation climate model. Technical Report 1, CMCC, Lecce.

Sundqvist, H. (1978): A parameterization scheme for non-convective condensation including prediction of cloud water content, Quart. J. Roy. Meteor. Soc., 104, 677–690.

Tiedtke, M., (1989): A comprehensive mass flux scheme for cumulus parameterization in large-scale models. Mon. Wea. Rev., 117, 1779-1800.

Valke, S, Terray, L, Piacentini, A (2000): The OASIS coupled user guide version 2.4, Technical Report TR/ CMGC/00-10, CERFACS

Williamsen, D. L. and Rasch, P.R., (1994): Water vapor transport in the NCAR CCM2. Tellus 46A, 34-51.

WMO, (1994): Guide to WMO binary code forms. World Meteorological Organization, Geneva, 117p.

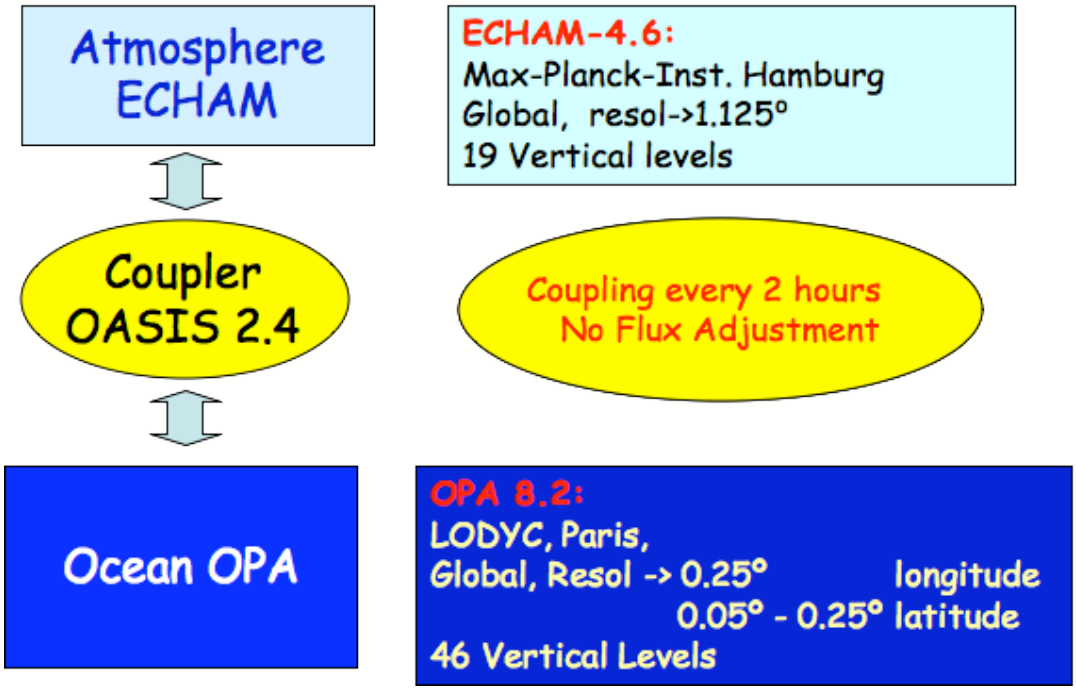

Figure 1: CMCC-SXF025 components.

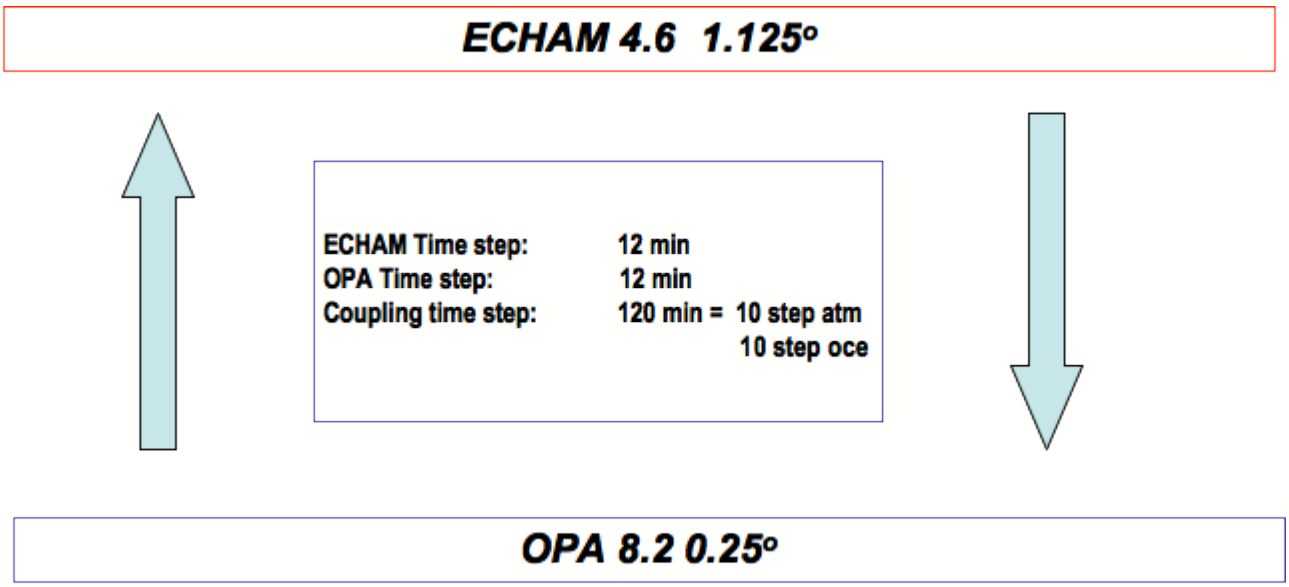

Figure 2: Time stepping.

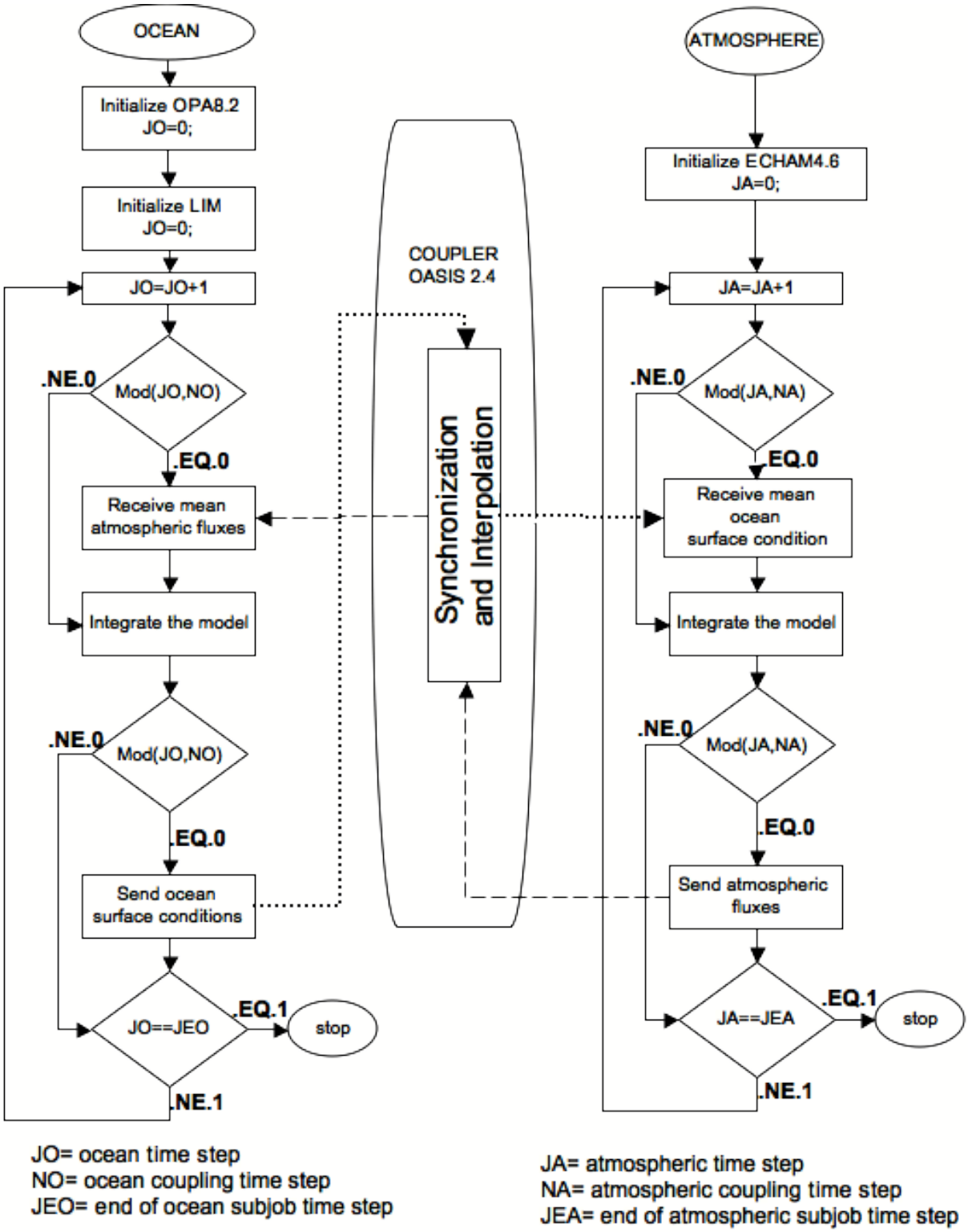

Figure 3: Scheme of time integration.

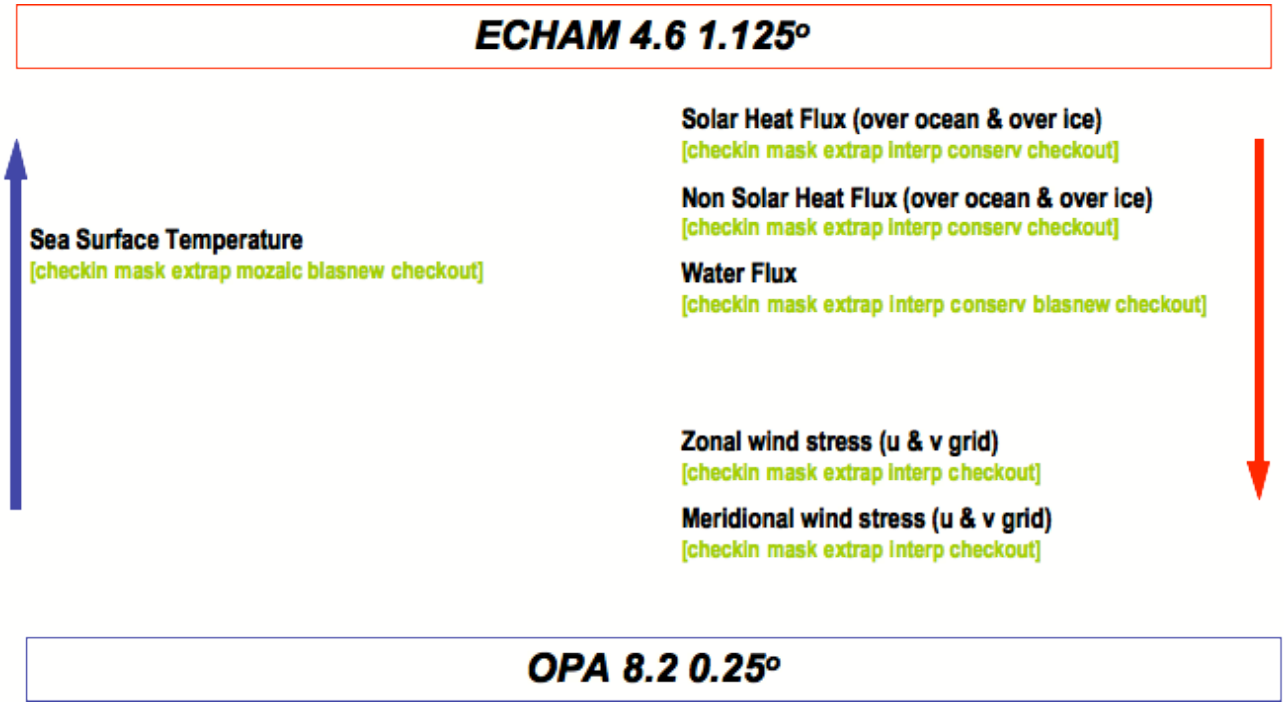

Figure 4: Coupling involved fields.

APPENDIX A

namcouple – namelist containing CMCC-SXF025 coupling settings.

```
#
 $SEQMODE
# This has to do with the time strategy. If you have all the models
# running simultaneously, you must put 1.
# Otherwise, if you have n models running sequentially, you put n
#
           1
 $END
#############################################################################
######
 $MACHINE
# This describes the type of machine you run OASIS on.
# - if it is a cray, put CRAY; otherwise put IEEE
#
             IEEE
 $END
#############################################################################
######
 $CHANNEL
# This describes the kind of message passing you want to use.
# - if you use named pipes + binary files (for synchro and data respectively)
#   you must write PIPE
# - if you use sockets for both synchro. and data (use of the Cerfacs library
#   CLIM based on PVM3.3), you must write CLIM
# - If you use system V shared memory segments and semaphores (for data and
#   synchro respectively), you must write SIPC
# - If you use OASIS as just an interpolator (i.e no models), you
#   must write NONE (furthermore you need to set NBMODEL to 0)
             CLIM
 $END
#############################################################################
######
 $CHATYPE
# This describes the type of message passing you want to use.
# - if you use PVM 3.3 version library :
#   you must write PVM3
# - if you use MPI-2 library of message passing,
#   you must write MPI2, the number of procs on which you want
#   to distribute each model, and the number of procs involved in
#   the coupling for each model.
# - Only these two types of libraries are allowed
MPI2 4 1 1 1
#############################################################################
######
 $NFIELDS
# This is the total number of fields being exchanged.
# 2 fields  Ocean -> Atm + 8 fields Atm. -> Ocean
```

```
#
# WARNING: RUNOFF not exchanged
# For the definition of the fields, see under $STRINGS keyword
#
          9
 $END
#########################################################################
######
 $JOBNAME
# This is an acronym for this given simulation
# (3 characters)
   CLI
 $END
#########################################################################
#######
 $NBMODEL
# This gives you the number of models running in this experiment +
# their names (6 characters).
#
  2 oceanx atmosx
 $END
#########################################################################
########
 $RUNTIME
# This gives you the total simulated time for this run in seconds (here 1 month)
#
2592000
 $END
#########################################################################
########
 $INIDATE
# This is the initial date of the run. It is regularly updated by the
# program. This is important if, for example, the SST field coming from
# a Pacific OGCM needs to be completed with climatological data
# of the right date.
#
00920701
 $END
#########################################################################
########
 $MODINFO
# Indicates if a header must be encapsulated within the field brick
# (YES or NOT)
  NOT
 $END
#########################################################################
########
 $NLOGPRT
# Index of printing level in output file cplout: 0 = no printing
#  1 = main routines and field names when treated, 2 = complete output
2
 $END
```

```
#####################################################################
########
 $STRINGS
#
# The above variables are the general parameters for the experiment.
# Everything below has to do with the fields being exchanged
# For each field, the first 2 lines are descriptors of the field, the
# related grid, the related model and data files.
# The third line gives the list of analysis to be performed  and the
# following lines give specific parameters  for each analysis.
# See the documentation for the analyses available in Oasis and for the
# relevant lines.
#
#####################################################################
#########
#                 OCEAN  --->>>  ATMOS
#                 --------------------
# Field 1 : sea surface temperature
#
#   First line:
# 1) and 2) Symbolic names for the field before and after interpolation
#          (8 characters maximum)
# 3) Label number for internal oasis output (cf blkdata.f)
# 4) Exchange frequency for the field in seconds (here 1 day)
# 5) Number of analysis to be performed
# 6) 7) 8) and 9) restart input binary file names + related unit numbers
# 10) Field status (EXPORTED or AUXILARY)
#
SOSSTSST SISUTESU 1 7200  6  sstocean   sstatmos  85   96  EXPORTED
#
#   Second line:
# 1) 2) 3) and 4) Number of long. and lat. on initial and final grids
# 5) and 6) locator prefix (4 characters) used to read the parameters
#          of the source and target grid
# 7) Index of the sequential position of the model generating the field
#    Meaningfull only if the SEQMODE input is > 1.
# 8) Flag used to delay the exchange of the given field in the case of models
#    running simultaneously (n = number of coupling timestep delay).
# 9) Flag to compute an extra timestep at the end (1 yes, 0 no)
# 10) Flag to compute the field integral in analyses CHECKIN and CHECKOUT
#     (1 yes, 0 no)
#
#182  149    320  160    or1t  a106     1     0      0      1
1442  1021    320  160    or1t  a106     1     0      0      1
P  2  P  0
SERIAL
#
# List of analyses
#
CHECKIN  MASK EXTRAP MOZAIC BLASNEW CHECKOUT
9.e9
NINENN  1 1 1
```

```
orcat106      91      1     117
#
# Blasnew: go from Celsius to Kelvin
# 1) mult. coeff for initial field 2) nb of additional fields
# 3) names of additional field, 4) value of multiplicative coefficient
#
  1.   1
  CONSTANT     273.15
#####################################################################
#########
#
#####################################################################
#########
# Field 2 : Sea ice extent
#
SOICECOV SIICECOV 2 7200  5  sstocean   sstatmos  85   96  EXPORTED
1442  1021    320  160    or1t  a106     1    0     0    1
P  2  P  0
SERIAL
#
CHECKIN MASK EXTRAP MOZAIC CHECKOUT
9.e9
NINENN  1 0 1
orcat106   91    1   117
#
#
#####################################################################
#########
#                 ATMOSPHERE  --->>>  OCEAN
#                 ------------------------
#####################################################################
#########
#
#####################################################################
#########
#
# Field 3 : Non solar heat flux
#
CONSFTOT SONSFLDO 6 7200  6  flxatmos  flxocean  87   89   EXPORTED
320   160  1442   1021  a106   or1t    1     0     0     1
P  0  P  2
SERIAL
#
CHECKIN  MASK EXTRAP   INTERP CONSERV  CHECKOUT
999999.9
NINENN  2 1 2
BICUBIC  G  SCALAR
GLOBAL
#
#####################################################################
#########
#
```

```
#####################################################################
#########
#
# Field 4 : Solar heat flux
#
COSHFTOT  SOSHFLDO   7   7200  6   flxatmos  flxocean  87   89  EXPORTED
320   160  1442   1021  a106   or1t  1  0   0   1
P  0  P  2
SERIAL
#
CHECKIN  MASK EXTRAP   INTERP CONSERV  CHECKOUT
999999.9
NINENN  2 1 2
BICUBIC  G  SCALAR
GLOBAL
#
#####################################################################
#########
#
#####################################################################
#########
#
# Field 5 : water flux
#
COWATFLU SOWAFLDO 25 7200  7   flxatmos  flxocean  87   89  EXPORTED
320   160   1442   1021    a106    or1t    1    0     0     1
P  0  P  2
SERIAL
#
CHECKIN  MASK EXTRAP   INTERP CONSERV  BLASNEW CHECKOUT
999999.9
NINENN  2 1 2
BICUBIC  G  SCALAR
GLOBAL
1000.  0
#
#####################################################################
#########
#
#####################################################################
#########
#####################################################################
######
#
# Field 6 : zonal wind stress -> ugrid
#
COZOTAUX  SOZOTAUX   23  7200  5   flxatmos  flxocean   87  89  EXPORTED
320  160   1442 1021  a106     or1u  1  0    0    1
P  0  P  2
SERIAL
#
 CHECKIN  MASK EXTRAP INTERP CHECKOUT
```

```
#
# NO24     SEALAND
# NORSUD    WSTEST
#
999.999
NINENN  2 1 2
BICUBIC  G   VECTOR
#
#####################################################################
######
#
# Field 7 : meridional wind stress -> vgrid
#
COMETAUY  SOMETAUY   24  7200  5   flxatmos  flxocean 87   89  EXPORTED
320   160  1442 1021  a106     or1v    1    0     0     1
P  0  P  2
SERIAL
#
 CHECKIN  MASK EXTRAP   INTERP  CHECKOUT
#
# NO24     SEALAND
#
# NORSUD    WSTEST
#
999.999
NINENN  2 0 2
BICUBIC  G   VECTOR
#
#####################################################################
######
#
# Field 8 : zonal wind stress -> vgrid
#
COZOTAUV  SOZOTAUV   23  7200  5   flxatmos  flxocean   87  89  EXPORTED
320  160   1442 1021  a106     or1v  1  0    0    1
P  0  P  2
SERIAL
#
 CHECKIN  MASK EXTRAP INTERP CHECKOUT
#
# NO24     SEALAND
#
# NORSUD    WSTEST
#
999.999
NINENN  2 0 2
BICUBIC  G    VECTOR
#
#####################################################################
######
#
# Field 9 : meridional wind stress -> ugrid
```

```
#
COMETAUU  SOMETAUU   24  7200  5   flxatmos  flxocean 87   89  EXPORTED
320   160  1442 1021  a106     or1u    1    0     0     1
P  0  P  2
SERIAL
#
 CHECKIN  MASK EXTRAP   INTERP  CHECKOUT
#
# NO24     SEALAND
#
# NORSUD    WSTEST
#
999.999
NINENN  2 0 2
BICUBIC  G  VECTOR
#
#########################################################################
#######
 $END
```

APPENDIX B

run.couple – Script to submit a CMCC-SXF025 simulation on NEC-SX6 vector machine.

```
#@$-q large
#@$-s /usr/bin/ksh -x
#@$-c 6
#@$-ac acca
#@$-eo
#@$-r F01_025
#
#####################################################################
##########
##
##      Based on
##      Script for coupled model LMDZ 72x45 / ORCA 4 degres
##      $Header: /home/ioipsl/CVSROOT/IPSLCM4/scripts/AA_job,v 1.11
##2002/05/10 14:19:19 adm Exp $
#####
##      Modified for NEC SX-6 cabernet.bo.ingv.it by Annalisa Cherchi Ottobre
##2003
##      Modified for coupling with orca025 by Enrico Scoccimarro Ottobre 2006
##
##      Eric MAISONNAVE / Marie-Alice FOUJOLS
##      Sources d'origine:
##                      Lancement Coupleur : Oasis team
##                      Lancement ORCA : Olivier Marti
##                      Lancement LMDZ : Laurent Fairhead
##                      Systeme de relances :  Olivier Marti
##
##
#####################################################################
##########
################## ENVIRONMENT VARIABLES TO CHANGE
###############
F_PROGINF=detail
export F_PROGINF
######################### MPI ENVIRONMENT VARIABLES
MPISUSPEND=on
export MPISUSPEND
MPIPROGINF=all_detail
export MPIPROGINF
##
#####################################################################
###########
##-- print the date
date
```

```
##-- print the "echo" of the commands before and after execution
set -vx
##
##
#######################################################################
############
#######################################################################
############
##
##    I. Definitions
##
#######################################################################
############
#######################################################################
############
##
##
##=====================================================
=============
##-----------------------------------------------------------------
##    I.1 General definitions
##-----------------------------------------------------------------
##=====================================================
=============
#
##-- Model Name
MODNAME=SINTEXF025
##-- Experience Name: to change
CEXPER=F01_025
##-- Directory where modipsl is located
MODIPSL=${HOMEC}/${MODNAME}/modipsl
## Number of cpu used in ECHAM 4.6: to change (according to -c and
NCPUS)
##NPROCA=4
NPROCA=1
NPROCB=1
NCPUSORCA=4
ATMPOSTPROC=/cabernet/${HOMEC}/${MODNAME}/${CEXPER}/runpost-
proc
#
##=====================================================
=============
##-----------------------------------------------------------------
##    I.2 definitions of experiment and restart lengths
##-----------------------------------------------------------------
##=====================================================
=============
##
```

```
##-- The experimement is divided into jobs (called with "qsub" command)
##    that are themselves divided into subjob (directly called by ./Job.
##    as a shell script without qsub )
##
##-- Eg: an 11 years experiment contains A) 1 year of echam spin up (not
##       managed by this script) and then B) 10 years of coupled
##       experiment that is divided into
##--        20 jobs of 6 months duration that are themselves divided into
##--        6 subjobs of 1 month duration.
##
##-- Calendar type, 2 cases are possible: 30d (for 30 days/month calendar)
##   or greg (for using the real gregorian calendar)
#
CALTYPE=30d
#
##-- First day of the experiment (including the spin-up of ECHAM)
##   As the calendar is included in ECHAM restarts files, if we want
##   that ORCA and ECHAM use the same calendar (necessary in the case
##   of the true calendar), we must consider the number of years of
##   ECHAM spin-up to define the calendar origin of the experiment.
#
DAY_BEGIN_EXP=1  ; MONTH_BEGIN_EXP=1  ; YEAR_BEGIN_EXP=1
#
##  Number of years already done in the echam spin-up:
#
NYSPIN=1
##--
##-- duration of the experiment (in months), only the coupled part,
##   without ECHAM spin-up
# to Change
CPEXP_DUR=1119
##--
##-- duration of one job (in months)
JOB_DUR=1119
##--
##-- maximum duration of one subjob (in months)
MAXSUBJOB_DUR=1
##--
##
##============================================================
============
##------------------------------------------------------------------
##    I.3 Definition of the IO directories
##------------------------------------------------------------------
##============================================================
============
##
## Only $STOREIN directory must exist before running the experiment.
```

```
## All the other directories will be automatically created if they
## are not existing.
##
## You have to define only: $STOREIN, $LOCAL and $STOREOUT
directories.
## The name of the other directories will be automatically defined.
## WARNING: Check the definition of $TOWORK. If the directory where
## the model is running is a temporary directory automatically defined
## by a enironment variable, use it to define TOWORK.
##   e.g.: TOWORK=$TMPDIR
##
# General purpose directories
#-------
HOMEDIR=${HOMEC}/${MODNAME}/${CEXPER}
cd $HOMEDIR
#
LOCAL=${SCRATCHC}/${MODNAME}/${CEXPER}
if [ ! -f ${SCRATCHC}/${MODNAME}/${CEXPER} ]; then
   mkdir -p ${SCRATCHC}/${MODNAME}/${CEXPER}
fi
# Remote Storage of Input data
#-------
STOREIN=${SCRATCHC}/${MODNAME}
#-------
# Remote Storage of model outputs
#-------
STOREOUT=${SCRATCHC}/${MODNAME}/${CEXPER}
#
# More specific directories...
#
# ----------------- ECHAM4 ---------------------
#
# local input from the atmosphere
LOCAL_INATM=${STOREIN}/In/Atm
# remote storage of the atmosphere results
STOUT_AFILES=${STOREOUT}/Atm/Results
#STOUT_AFILES=${SCRATCHC}/${MODNAME}/${CEXPER}/Atm/Results
if [ ! -f $STOUT_AFILES ]; then
  mkdir -p $STOUT_AFILES
fi
STOREOUT_ARES=${STOREOUT}/Atm/Restarts
if [ ! -f $STOREOUT_ARES ]; then
  mkdir -p $STOREOUT_ARES
fi
#
# ----------------- ORCA ----------------------
#
LOCAL_INOCE=${STOREIN}/In/Oce
```

```
STOUT_OFILES=${STOREOUT}/Oce/Results
#STOUT_OFILES=${SCRATCHC}/${MODNAME}/${CEXPER}/Oce/Results
if [ ! -f $STOUT_OFILES ]; then
  mkdir -p $STOUT_OFILES
fi
STOREOUT_ORES=${STOREOUT}/Oce/Restarts
if [ ! -f $STOREOUT_ORES ]; then
  mkdir -p  $STOREOUT_ORES
fi
#
# ----------------- OASIS ---------------------
#
LOCAL_INCPL=${STOREIN}/In/Cpl
STOREOUT_CRES=${STOREOUT}/Cpl/Restarts
if [ ! -f $STOREOUT_CRES ]; then
  mkdir -p $STOREOUT_CRES
fi
#
# ----------------- OTHERS ---------------------
#
# remote storage of Namelists of each models
NAMDIR=${HOMEDIR}/Namelists
if [ ! -f $NAMDIR ]; then
  mkdir -p $NAMDIR
fi
# remote storage of the main outputs files of each models
STOUT_OUT=${HOMEDIR}/Out
if [ ! -f $STOUT_OUT ]; then
  mkdir -p $STOUT_OUT
fi
# remote storage of other outputs files!
STOUT_MORE=${HOMEDIR}/More
if [ ! -f $STOUT_MORE ]; then
  mkdir -p $STOUT_MORE
fi
#
# ----------- directory where the model is running -----------
#
TOWORK=${LOCAL}/Towork
if [ ! -f $TOWORK ]; then
  mkdir -p $TOWORK
fi
#!E echo ls -l of $LOCAL
#!E ls -l *
cd ${TOWORK} || exit
#- cleanning
\rm -f *
##
```

```
##
if [ ! -f ${LOCAL}/Res/Cpl ]; then
  mkdir -p ${LOCAL}/Res/Cpl
fi
#
if [ ! -f ${LOCAL}/Res/Atm ]; then
  mkdir -p ${LOCAL}/Res/Atm
fi
#
if [ ! -f ${LOCAL}/Res/Oce ]; then
  mkdir -p ${LOCAL}/Res/Oce
fi

##############################################################################
############
##############################################################################
############
##
##    II. General check/definition of Calendar and dates
##
##############################################################################
############
##############################################################################
############
##
##
#
SCRIPTDIR=${HOMEC}/$MODNAME/modipsl/modeles/${MODNAME}/scripts
#
# Get calendar computation tools
#
cp ${SCRIPTDIR}/julday .
cp ${SCRIPTDIR}/caldat .
#
# somme additional calendar computation tools...
#
valid_date () { caldat `julday $1 $2 $3 $CALTYPE` $CALTYPE ; }
yyyymmdd_fmt () { valid_date $1 $2 $3 | read m d y ; echo $( printf "%04s\n"
$y)$( printf "%02s\n" $m)$( printf "%02s\n" $d) ; }
#
##============================================================
============
##------------------------------------------------------------
##    II.1 Calendar computation linked to the experiment
##------------------------------------------------------------
##============================================================
============
#
```

```
# first date of the experiment in format yyyymmdd
DATE_BEGIN_EXP=`yyyymmdd_fmt $MONTH_BEGIN_EXP
$DAY_BEGIN_EXP $YEAR_BEGIN_EXP`
#
# first date of the coupled experiment
valid_date $(( $MONTH_BEGIN_EXP + $NYSPIN*12 )) $DAY_BEGIN_EXP
$YEAR_BEGIN_EXP | read MONTH_BEGIN_CPEXP DAY_BEGIN_CPEXP
YEAR_BEGIN_CPEXP
# first date of the coupled experiment in format yyyymmdd
DATE_BEGIN_CPEXP=`yyyymmdd_fmt $MONTH_BEGIN_CPEXP
$DAY_BEGIN_CPEXP $YEAR_BEGIN_CPEXP`
#
# last date of the experiment
valid_date $(( $MONTH_BEGIN_EXP + $CPEXP_DUR + $NYSPIN*12 )) $((
$DAY_BEGIN_EXP-1 )) $YEAR_BEGIN_EXP | read MONTH_END_EXP
DAY_END_EXP YEAR_END_EXP
# last date of the experiment in format yyyymmdd
DATE_END_EXP=`yyyymmdd_fmt $MONTH_END_EXP $DAY_END_EXP
$YEAR_END_EXP`
##
##============================================================
============
##------------------------------------------------------------
##    II.2 definition of the auxiliary file used in the restart procedure
##------------------------------------------------------------
##============================================================
============
##
##  Creation of ${CEXPER}.trace if it doesn't exist.
##  This file is used for the restart proceddure.
##  It contains:
##    1) Year Month Day of the beginning of the subjob.
##    2) The number of months that have already been done in the job
##
## WARNING WARNING WARNING WARNING WARNING WARNING
## WARNING WARNING WARNING WARNING WARNING WARNING
##
##    Delete the file ${CEXPER}.trace if you want
##    to start a new experiment from the beginning.
##
## WARNING WARNING WARNING WARNING WARNING WARNING
## WARNING WARNING WARNING WARNING WARNING WARNING
##
#
if [ ! -f ${HOMEDIR}/${CEXPER}.trace ]
then
## if ${CEXPER}.trace doesn't exist
## this is the beginning of the coupled experiment.
```

```
## start from the beginnig of the coupled experiment
   YEAR_BEGIN_SUBJOB=${YEAR_BEGIN_CPEXP}
   MONTH_BEGIN_SUBJOB=${MONTH_BEGIN_CPEXP}
   DAY_BEGIN_SUBJOB=${DAY_BEGIN_CPEXP}
## 0 days have been done in the job
   MONTHS_DONE=0
## store the informations in ${CEXPER}.trace
   cat <<EOF >${CEXPER}.trace
$YEAR_BEGIN_SUBJOB $MONTH_BEGIN_SUBJOB
$DAY_BEGIN_SUBJOB $MONTHS_DONE
EOF
## define the file ${CEXPER}.log that contains jobs history
   echo `date` " Begining of The Coupled Experience : " ${CEXPER} " in
directory :" ${R_EXPER} > ${CEXPER}.log
else
  cp ${HOMEDIR}/${CEXPER}.trace .
fi
##
##============================================================
##------------------------------------------------------------
##    II.3 Calendar computation linked to the subjob
##------------------------------------------------------------
##============================================================
##-------
## get, via ${CEXPER}.trace, year month day of subjob beginning and the
number
## of days already done in the Job
##-------
cat ${CEXPER}.trace | read YEAR_BEGIN_SUBJOB
MONTH_BEGIN_SUBJOB DAY_BEGIN_SUBJOB MONTHS_DONE
echo In ${CEXPER}.trace:
echo $YEAR_BEGIN_SUBJOB $MONTH_BEGIN_SUBJOB
$DAY_BEGIN_SUBJOB $MONTHS_DONE
##--------
## beginning date of subjob in format yyyymmdd
##--------
DATE_BEGIN_SUBJOB=`yyyymmdd_fmt $MONTH_BEGIN_SUBJOB
$DAY_BEGIN_SUBJOB $YEAR_BEGIN_SUBJOB`
## !E scoccimarro restart mod
MONTH_BEGIN_SUBJOB_l2=$(printf "%02s\n" $MONTH_BEGIN_SUBJOB)
echo $MONTH_BEGIN_SUBJOB_l2
##--------
## Duration of the subjob in months
##--------
SUBJOB_DUR=$(( $JOB_DUR - $MONTHS_DONE ))
[ ${SUBJOB_DUR} -gt ${MAXSUBJOB_DUR} ] &&
SUBJOB_DUR=${MAXSUBJOB_DUR}
##-------
```

```
## year month day of the end of subjob
##-------
valid_date $(( $MONTH_BEGIN_SUBJOB+$SUBJOB_DUR )) $((
$DAY_BEGIN_SUBJOB-1 )) $YEAR_BEGIN_SUBJOB | read
MONTH_END_SUBJOB DAY_END_SUBJOB YEAR_END_SUBJOB
##-------
## last date of subjob in format yyyymmdd
##-------
DATE_END_SUBJOB=`yyyymmdd_fmt $MONTH_END_SUBJOB
$DAY_END_SUBJOB $YEAR_END_SUBJOB`
##-------
## dates interval of the subjob
##-------
DATES=${DATE_BEGIN_SUBJOB}_${DATE_END_SUBJOB}
##-------
## beginning of the experiment in julian calendar (in days):
##-------
JDAY_FIRST_EXP=`julday ${MONTH_BEGIN_EXP} ${DAY_BEGIN_EXP}
${YEAR_BEGIN_EXP} $CALTYPE`
##-------
## beginning of the coupled experiment in julian calendar (in days):
##-------
JDAY_FIRST_CPEXP=`julday ${MONTH_BEGIN_CPEXP}
${DAY_BEGIN_CPEXP} ${YEAR_BEGIN_CPEXP} $CALTYPE`
##-------
## beginning and end of subjob in julian calendar (in days):
##-------
JDAY_FIRST_SUBJOB=`julday ${MONTH_BEGIN_SUBJOB}
${DAY_BEGIN_SUBJOB} ${YEAR_BEGIN_SUBJOB} $CALTYPE`
JDAY_LAST_SUBJOB=`julday ${MONTH_END_SUBJOB}
${DAY_END_SUBJOB} ${YEAR_END_SUBJOB} $CALTYPE`
##-------
## Subjob length (in days)
##-------
(( NB_DAY_SUBJOB = ${JDAY_LAST_SUBJOB} - ${JDAY_FIRST_SUBJOB}
+ 1 ))
##
##
##=========================================================
##---------------------------------------------------------
##    II.4 calendar computations for restart procedure
##---------------------------------------------------------
##=========================================================
##
##-------
## year month day of the restart
##-------
```

```
valid_date $MONTH_BEGIN_SUBJOB $(( $DAY_BEGIN_SUBJOB-1 ))
$YEAR_BEGIN_SUBJOB | read MONTH_RESTART DAY_RESTART
YEAR_RESTART
##-------
## restart date in format yyyymmdd
##-------
DATE_RESTART=`yyyymmdd_fmt $MONTH_RESTART $DAY_RESTART
$YEAR_RESTART`
#
##
##============================================================
##------------------------------------------------------------
##    II.5 calendar computations  for ORCA (to redefine ORCA namelist)
##------------------------------------------------------------
##============================================================
##
#-------
## Number of time step per day in ORCA
#-------
### !E scoccimarro + 06/07 coupling orca025<->echam4.6
### 24 se tstep=60min
### 72 se tstep=20min
# ORCA_NTS_DAY=24
timestep=720.
ORCA_NTS_DAY=120
### !E -
#-------
## frequency of the outputs write
#-------
### !E scoccimarro + 06/07 coupling orca025<->echam4.6
## (( ORCA_NWRITE = ORCA_NTS_DAY*5 )) # 5 days
(( ORCA_NWRITE = ORCA_NTS_DAY*30 )) # 30 days
### !E -
#-------
## computation of fist/last time step and number of time steps in subjob
#-------
(( ORCA_NIT000 = ( ${JDAY_FIRST_SUBJOB} - ${JDAY_FIRST_CPEXP} ) *
${ORCA_NTS_DAY} + 1 ))
(( ORCA_NITEND = ( ${JDAY_LAST_SUBJOB} - ${JDAY_FIRST_CPEXP} +
1 ) * ${ORCA_NTS_DAY}     ))
(( ORCA_NTS = ORCA_NITEND - ORCA_NIT000 + 1 ))
#-------
## length of subjob (in seconds)
#-------
(( L_SUBJOB_SEC = 3600 * 24 * $NB_DAY_SUBJOB ))
#
##
##============================================================
```

```
##-----------------------------------------------------------------
##    II.6 calendar computations  for ECHAM (to redefine ECHAM namelist)
##-----------------------------------------------------------------
##=================================================================
##
#-------
## Number of time step per day in ECHAM
#-------
ECHAM_NTS_DAY=120
#-------
## frequency of the outputs write (in timesteps)
#-------
#NPTIME=$(( 5 * $ECHAM_NTS_DAY )) # 5 days
NPTIME=60
# NPTIME=60 # 0.5 days
#-------
## When shall we stop echam (in timesteps) ?
#-------
#NSTOP=$(( ( $JDAY_LAST_SUBJOB - $JDAY_FIRST_EXP + 1 ) *
$ECHAM_NTS_DAY ))
#
#****************************************
#****************************************
# DA 9603 IN POI
#
#NWTIME=$(( ( $JDAY_LAST_SUBJOB - $JDAY_FIRST_EXP + 1 ) *
$ECHAM_NTS_DAY ))
#NWTIME=-30
#NWTIME=3600
NSTOP=$(( ( ( ${YEAR_BEGIN_SUBJOB} - 1 ) * 12 +
${MONTH_BEGIN_SUBJOB} ) * 3600  ))
NSTOP=$(( $NSTOP -1 ))
#NSTOP=-30
##
##
##=================================================================
##-----------------------------------------------------------------
##    II.7 General echo
##-----------------------------------------------------------------
##=================================================================
##
##-------
## store date into ${CEXPER}.log
##-------
echo "`date` $YEAR_BEGIN_SUBJOB $MONTH_BEGIN_SUBJOB
$DAY_BEGIN_SUBJOB begin" >> ${CEXPER}.log
##-------
##-------
```

```
echo $YEAR_BEGIN_EXP $MONTH_BEGIN_EXP $DAY_BEGIN_EXP
echo $YEAR_BEGIN_CPEXP $MONTH_BEGIN_CPEXP
$DAY_BEGIN_CPEXP
echo $YEAR_END_EXP $MONTH_END_EXP $DAY_END_EXP
echo $DATE_BEGIN_EXP $DATE_BEGIN_CPEXP $DATE_END_EXP
#
echo $YEAR_BEGIN_SUBJOB $MONTH_BEGIN_SUBJOB
$DAY_BEGIN_SUBJOB
echo $YEAR_END_SUBJOB $MONTH_END_SUBJOB $DAY_END_SUBJOB
echo $DATE_BEGIN_SUBJOB $DATE_END_SUBJOB
#
echo $JDAY_FIRST_EXP $JDAY_FIRST_CPEXP
echo $JDAY_FIRST_SUBJOB $JDAY_LAST_SUBJOB $NB_DAY_SUBJOB
#
echo $DATE_RESTART
#
echo $ORCA_NTS_DAY $ORCA_NWRITE
echo $ORCA_NIT000 $ORCA_NITEND $ORCA_NTS
#
echo $ECHAM_NTS_DAY $NPTIME
echo $NSTOP
##
##
#####################################################################
############
#####################################################################
############
##
##    III. executables
##
#####################################################################
############
#####################################################################
############
##
##
##=============================================================
##-------------------------------------------------------------
##    III.1 get the executables files
##-------------------------------------------------------------
##=============================================================
##
#
DIREXE=${HOMEC}/$MODNAME/modipsl/bin
# --- ECHAM ------
#cp ${DIREXE}/atmosx.6cpu.andru atmosx
cp ${DIREXE}/atmosx atmosx
# --- ORCA -------
```

```
cp ${DIREXE}/opa025 oceanx
# --- OASIS ------
#cp ${DIREXE}/oasis.6cpu.andru oasis
cp ${DIREXE}/oasis oasis
#
##
##============================================================
##------------------------------------------------------------
##    III.2 Number of Cpu used in ECHAM 4.6
##          => verification of the "oasis part" of ECHAM and ORCA...
##------------------------------------------------------------
##============================================================
============
##
#
NCPUSECHAM=$(( $NPROCA * $NPROCB ))
#INORCA=$MODIPSL/modeles/OPA/SRC_ORCA/inicmo.F
#INECHAM=$MODIPSL/modeles/ECHAM4.6/libclim/inicma.F
## atmosx has the same number in ORCA and ECHAM?
#grep cmpi_modnam.*atmosx $INORCA | cut -f2 -d"(" | cut -f1 -d")" | read
numatmosx_orca
#grep cmpi_modnam.*atmosx $INECHAM | cut -f2 -d"(" | cut -f1 -d")" | read
numatmosx_echam
#echo numatmosx_orca $numatmosx_orca numatmosx_echam
$numatmosx_echam
#[ $numatmosx_orca -ne $numatmosx_echam ] && echo "atmosx has not the
same number in ORCA and ECHAM, we stop, check the files "$INORCA and
$INECHAM && exit
##
## number of CPU defined for ECHAM in the Oasis part or ORCA
#grep mpi_totproc\($numatmosx_orca $INORCA | cut -f2 -d"=" | read
procsnum_orca
#echo procsnum_orca $procsnum_orca
#[ $procsnum_orca -ne $NCPUSECHAM ] && echo "The number of CPU
defined for ECHAM in the Oasis part or ORCA is not equal to
NPROCA*NPROCB, we stop, check the file "$INORCA  && exit
##
## number of CPU defined for ECHAM in the Oasis part or ECHAM
#grep mpi_totproc\($numatmosx_echam $INECHAM | cut -f2 -d"=" | read
procsnum_echam
#echo procsnum_echam $procsnum_echam
#[ $procsnum_echam -ne $NCPUSECHAM ] && echo "The number of CPU
defined for ECHAM in the Oasis part or ECHAM is not equal to
NPROCA*NPROCB, we stop, check the file "$$INORCA && exit
##
##
##============================================================
##------------------------------------------------------------
```

```
##    III.3 ORCA Verifications
##-------------------------------------------------------------
##=============================================================
##
#-------
# Check ORCA Resolution (must be 2)
#-------
grep key_orca
${HOMEC}/$MODNAME/modipsl/modeles/OPA/WORK/KEY_CPP | sed
s/.*key_orca_r// | sed s/-Wp.*// | read RESOL_ORCA

echo RESOL_ORCA ${RESOL_ORCA}
if [ ${RESOL_ORCA} -ne "2" ] ; then
    echo " Bad resolution of ORCA : " ${RESOL_ORCA}
    exit
fi
#
#-------
# Check real and integer size in ORCA executable
#-------
#### OPT_COMPIL_ORCA=` mcs -p oceanx | grep key_orca | head -1 `
IR_ORCA=` mcs -p oceanx | grep key_orca | head -1 | grep "-ew" `
if [ -n ${IR_ORCA} ]; then
   echo 'real size :   8 '
   echo 'integer size : 8 '
else
   echo ' Warning: ORCA exectable is not using I8R8 '
   exit
fi
#
##
###############################################################################
############
###############################################################################
############
##
##    IV. namatmos/namelists/namcouple/
##
###############################################################################
############
###############################################################################
############
##
##
#
#NAMBASEDIR=$MODIPSL/modeles/$MODNAME/scripts
NAMBASEDIR=${HOMEC}/$MODNAME/modipsl/modeles/$MODNAME/script
s
```

```
#
##
##============================================================
##------------------------------------------------------------
##    IV.1 ECHAM4.6: namatmos
##------------------------------------------------------------
##============================================================
##
#-------
# get
#-------
cp ${NAMBASEDIR}/namatmos.base namatmos
#-------
# extract some values from namatmos...
#-------
LY365OLD=$( grep 'LY365'  namatmos)
NDSTARTOLD=$( grep 'NDSTART'  namatmos)
NSTOPOLD=$( grep 'NSTOP'  namatmos)
NWTIMEOLD=$( grep 'NWTIME'  namatmos)
NPROCAOLD=$( grep 'NPROCA'  namatmos)
NPROCBOLD=$( grep 'NPROCB'  namatmos)
NPTIMEOLD=$( grep 'NPTIME'  namatmos)
#-------
# define the new values
#-------
[ $CALTYPE = 30d ] && LY365=.false. || LY365=.true.
#-------
# change some namelist values
#-------
#sed -e "s/$LY365OLD/  LY365=$LY365,/" \
#    -e "s/$NDSTARTOLD/  NDSTART=$DATE_BEGIN_EXP,/" \
#    -e "s/$NSTOPOLD/  NSTOP=$NSTOP,/" \
#    -e "s/$NWTIMEOLD/  NWTIME=$NWTIME,/" \
#    -e "s/$NPROCAOLD/  NPROCA=$NPROCA,/" \
#    -e "s/$NPROCBOLD/  NPROCB=$NPROCB,/" \
#    -e "s/$NPTIMEOLD/  NPTIME=$NPTIME,/" \
#    namatmos > namtmp || exit
#
#sed -e "s/$LY365OLD/  LY365=$LY365,/" \
#    -e "s/$NDSTARTOLD/  NDSTART=$DATE_BEGIN_EXP,/" \
#    -e "s/$NSTOPOLD/  NSTOP=$NSTOP,/" \
#    -e "s/$NPROCAOLD/  NPROCA=$NPROCA,/" \
#    -e "s/$NPROCBOLD/  NPROCB=$NPROCB,/" \
#    -e "s/$NWTIMEOLD/  NWTIME=$NWTIME,/" \
#    -e "/$NPTIMEOLD/d" \
#    namatmos > namtmp || exit
#
sed -e "s/$LY365OLD/  LY365=$LY365,/" \
```

```
   -e "s/$NDSTARTOLD/  NDSTART=$DATE_BEGIN_EXP,/" \
   -e "s/$NSTOPOLD/  NSTOP=$NSTOP,/" \
   -e "s/$NPROCAOLD/  NPROCA=$NPROCA,/" \
   -e "s/$NPROCBOLD/  NPROCB=$NPROCB,/" \
   -e "/$NWTIMEOLD/d" \
   -e "/$NPTIMEOLD/d" \
   namatmos > namtmp || exit
mv namtmp namatmos
#
##
##============================================================
##------------------------------------------------------------
##    IV.2 ORCA: namelist & run.param.li
##------------------------------------------------------------
##============================================================
##
# ----------------- ORCA ---------------------
#
#-------
# get
#-------
cp ${NAMBASEDIR}/namelist_orca025_coup namelist
#-------
# transformation namelist: fortran 77 --> fortran 90
#-------
sed -e /:/d -e s/" \&END"/"\/"/ -e s/" \&"/"\&"/ namelist > namtmp || exit
mv namtmp namelist
#-------
# extract some namelist values
#-------
PAT_CEXPER=$( grep 'cexper' namelist )
PAT_NIT000=$( grep 'nit000' namelist )
PAT_NITEND=$( grep 'nitend' namelist )
PAT_NWRITE=$( grep 'nwrite' namelist )
#PAT_RESTAR=$( grep 'lrstar' namelist )
PAT_RESTAR=$( grep 'lrstar' namelist|grep -v lrstarfl )
PAT_NSTOCK=$( grep 'nstock' namelist )
PAT_NRSTAR=$( grep 'nrstdt' namelist )
PAT_NMSH=$( grep 'nmsh' namelist )
PAT_NDATE0=$( grep 'ndate0' namelist )
PAT_NCLOSEA=$( grep 'nclosea' namelist )
PAT_NBISEX=$( grep 'nbisex' namelist )
PAT_NPRINT=$( grep 'nprint' namelist )
#-------
# define the new values
#-------
[ $CALTYPE = 30d ] && nbisex=30 || nbisex=1
#-- check of the number of time step per day
```

```
ORCA_NSTOCK=$(echo $PAT_NSTOCK | sed "s/[a-z,A-Z,=]//g" )
ORCA_RDT=$( egrep 'rdt *=' namelist | sed 's/ *rdt *=//' | sed 's/\. *,//' )
ORCA_NSTOCK="$ORCA_NTS"
(( NB_SEC_DAY_MODEL = ORCA_NTS_DAY * ORCA_RDT ))
(( NB_SEC_DAY = 60 * 60 * 24 ))
if [ ${NB_SEC_DAY_MODEL} -ne ${NB_SEC_DAY} ]
then
        echo " Check ORCA_NTS_DAY ... "
        exit
fi
#-------
# change some namelist values
#-------
# General changes
sed    -e "s/$PAT_CEXPER/       cexper=\"$CEXPER\",/" \
   -e "s/$PAT_NIT000/       nit000=$ORCA_NIT000,/" \
   -e "s/$PAT_NITEND/       nitend=$ORCA_NITEND,/" \
   -e "s/$PAT_NWRITE/       nwrite=$ORCA_NWRITE,/" \
   -e "s/$PAT_NSTOCK/       nstock=$ORCA_NSTOCK,/" \
   -e "s/$PAT_NDATE0/       ndate0=$DATE_BEGIN_SUBJOB,/" \
   -e "s/$PAT_NCLOSEA/      nclosea=0,/" \
   -e "s/$PAT_NBISEX/       nbisex=$nbisex,/" \
   -e "s/$PAT_NPRINT/       nprint=1,/" \
   namelist > nam.tmp
   sed "s/tstp./$timestep/g" nam.tmp > nam.tmp2
# Changes only if it is the fist subjob of the experiment
if [ ${DATE_BEGIN_SUBJOB} -eq ${DATE_BEGIN_CPEXP} ]
then
  sed    -e "s/$PAT_RESTAR/       lrstar=.FALSE.,/" \
    -e "s/$PAT_NMSH/       nmsh=1,/" \
    -e "s/$PAT_NRSTAR/       nrstdt=0,/" \
    nam.tmp2 > namelist
else
# Changes only if we use restart
  sed    -e "s/$PAT_RESTAR/       lrstar=.TRUE.,/" \
    -e "s/$PAT_NMSH/       nmsh=1,/" \
    -e "s/$PAT_NRSTAR/       nrstdt=1,/" \
    nam.tmp2 > namelist
fi
\rm -f nam.tmp
\rm -f nam.tmp2
#
# ---ICE---
#
#    Not implemented---Not implemented
#    Not implemented---Not implemented
#GETNAM run.param.li
#if [ ${DATE_BEGIN_SUBJOB} -eq ${DATE_BEGIN_CPEXP} ]
```

```
#then
#  sed -e "s/KKKKKKK/2/" run.param.li >run.param.tmp
#else
#  sed -e "s/KKKKKKK/1/" run.param.li >run.param.tmp
#fi
#- take care about ice result storage
#(( ICE_NSTOCK = ORCA_NSTOCK ))
#sed -e "s/SSSSSSS/${ICE_NSTOCK}/" run.param.tmp >run.param.li
#cat run.param.li
#    Not implemented---Not implemented
#    Not implemented---Not implemented
#
##
##============================================================
##------------------------------------------------------------
##    IV.3 OASIS: namcouple
##------------------------------------------------------------
##============================================================
##
#
#-------
# get
#-------
cp ${NAMBASEDIR}/namcouple.base_orca025_coup namcouple
#-------
# change some values
#  - Index of printing level in output file cplout
#  - Initial date of the run.
#  - Total simulated time for this run in seconds
#-------
sed -e "s/^ *2 *$/0/" \
    -e "s/^ *0*10101 *$/${DATE_BEGIN_SUBJOB}/" \
    -e "s/^ *2592000 *$/${L_SUBJOB_SEC}/" \
    -e "s/^ *MPI2  *1  *1  *1  *1/MPI2 $NCPUSORCA $NCPUSECHAM 1 1/" \
    -e "s/^0$/2/" \
    namcouple > nam.tmp
##!E    -e "s/^ *MPI2  *1  *1  *1  *1/MPI2 1 $NCPUSECHAM 1 1/" \
##!E    -e "s/^0$/1/" \
mv nam.tmp namcouple
#
##
##
#################################################################
############
#################################################################
############
##
##    V. Inputs files and restarts
```

```
##
#####################################################################
############
#####################################################################
############
##
##==========================================================
##----------------------------------------------------------
##  V.1 ECHAM4.6
##----------------------------------------------------------
##==========================================================
#
#-------
# input files
#-------
cp ${LOCAL_INATM}/year106m.nc unit.20
#-------
# restart files
#-------
if [ ${DATE_BEGIN_SUBJOB} -eq ${DATE_BEGIN_CPEXP} ]
   then
   for unit in 31 32 35 36 37 ; do
        cp ${STOREIN}/Res/Atm/atm_restart0_unit.$unit unit.$unit
   done
else
   for unit in 31 32 35 36 37 ; do
        cp ${LOCAL}/Res/Atm/${CEXPER}_${DATE_RESTART}_unit.$unit
unit.$unit
   done
fi
F_FF99=namatmos
export F_FF99
#
##==========================================================
##----------------------------------------------------------
##  V.2 ORCA & Ice
##----------------------------------------------------------
##==========================================================
#
#-------
# -  Inputs files for OPRCA
#-------
# levitus data -> feeback in closed seas and mediterranean sea.
###!E cp ${LOCAL_INOCE}/LEVITUS_1m_Salinity_Ice_nomask.nc .
###!E cp ${LOCAL_INOCE}/LEVITUS_1m_Temperature_Pot_Ice_nomask.nc
.
# for the ice IF
```

```
###!E cp ${LOCAL_INOCE}/REYNOLDS_1d_seasonnalcycle_82-89.interp.nc
sst_1d.nc
# eddy viscosity coef
###!E cp ${LOCAL_INOCE}/ahmcoef .
###!E cp ${LOCAL_INOCE}/bathymetry .
# for the atlantic overturning (key_diaznl)
###!E cp ${LOCAL_INOCE}/bathymetry.atlantic .
cp ${LOCAL_INOCE}/coordinates_ORCA_R025.nc ./coordinates.nc
  ln -s coordinates.nc coordinates_000.nc
  ln -s coordinates.nc coordinates_001.nc
  ln -s coordinates.nc coordinates_002.nc
  ln -s coordinates.nc coordinates_003.nc

  ln -s ${LOCAL_INOCE}/maskglo_float.nc maskglo.nc
  ln -s ${LOCAL_INOCE}/bathymetry_ORCA_R025.ascii bathymetry
#  ln -s ${LOCAL_INOCE}/Levitus98_PHC21_01_Sal_ORCA_R025.nc
Levitus98_PHC21_05_Sal_ORCA_R025.nc
#  ln -s ${LOCAL_INOCE}/Levitus98_PHC21_01_Tem_ORCA_R025.nc
Levitus98_PHC21_05_Tem_ORCA_R025.nc
  ln -s ${LOCAL_INOCE}/Levitus98_PHC21_01_Sal_ORCA_R025.nc
Levitus98_PHC21_${MONTH_BEGIN_SUBJOB_l2}_Sal_ORCA_R025.nc
  ln -s ${LOCAL_INOCE}/Levitus98_PHC21_01_Tem_ORCA_R025.nc
Levitus98_PHC21_${MONTH_BEGIN_SUBJOB_l2}_Tem_ORCA_R025.nc

  ln -s ${LOCAL_INOCE}/EMPave.dat.0.POG-05 EMPave_old.dat
#### !E per fare la prima settimana del maggio 1992 che poi qui e' 92 ..c'e' un
trucco in dtasst.forced.h
  ln -s
/scratch/users/enrico/data/ORCA025/ECMWF_SST_POG_15340_15346_200
40408_1315.nc ECMWF_SST_POG_15459_15465.nc
  ln -s
/scratch/users/enrico/data/ORCA025/ECMWF_SST_POG_15340_15346_200
40408_1315.nc ECMWF_SST_POG_15466_15472.nc
  ln -s
/scratch/users/enrico/data/ORCA025/ECMWF_SST_POG_15340_15346_200
40408_1315.nc ECMWF_SST_POG_15473_15479.nc
  ln -s
/scratch/users/enrico/data/ORCA025/ECMWF_SST_POG_15340_15346_200
40408_1315.nc ECMWF_SST_POG_15480_15486.nc
  ln -s
/scratch/users/enrico/data/ORCA025/ECMWF_SST_POG_15340_15346_200
40408_1315.nc ECMWF_SST_POG_15487_15493.nc
  ln -s
/scratch/users/enrico/data/ORCA025/ECMWF_SST_POG_15340_15346_200
40408_1315.nc ECMWF_SST_POG_15494_15500.nc
##
```

```
  ln -s
/scratch/users/enrico/data/ORCA025/ECMWF_SST_POG_15340_15346_200
40408_1315.nc ECMWF_SST_POG_15501_15507.nc
  ln -s
/scratch/users/enrico/data/ORCA025/ECMWF_SST_POG_15340_15346_200
40408_1315.nc ECMWF_SST_POG_15508_15514.nc
  ln -s
/scratch/users/enrico/data/ORCA025/ECMWF_SST_POG_15340_15346_200
40408_1315.nc ECMWF_SST_POG_15515_15521.nc
  ln -s
/scratch/users/enrico/data/ORCA025/ECMWF_SST_POG_15340_15346_200
40408_1315.nc ECMWF_SST_POG_15522_15528.nc
  ln -s
/scratch/users/enrico/data/ORCA025/ECMWF_SST_POG_15340_15346_200
40408_1315.nc ECMWF_SST_POG_15529_15535.nc

###  !E
# new runoff parametrisation
  ln -s ${LOCAL_INOCE}/runoff_1m_ORCA_R025.nc runoff_1m_nomask.nc
#-------
# - restarts ORCA
#-------
#
if [ ${DATE_BEGIN_SUBJOB} -eq ${DATE_BEGIN_CPEXP} ]
then
  echo "NO RESTARTS FOR ORCA and UCL"
else
  echo "RESTART ORCA and UCL"
#  cp ${LOCAL}/Res/Oce/${CEXPER}_${DATE_RESTART}_orcaini.nc
orcaini.nc
  cp ${LOCAL}/Res/Oce/${CEXPER}_${DATE_RESTART}_restart_000.nc
restart_000.nc
  cp ${LOCAL}/Res/Oce/${CEXPER}_${DATE_RESTART}_restart_001.nc
restart_001.nc
  cp ${LOCAL}/Res/Oce/${CEXPER}_${DATE_RESTART}_restart_002.nc
restart_002.nc
  cp ${LOCAL}/Res/Oce/${CEXPER}_${DATE_RESTART}_restart_003.nc
restart_003.nc
fi
#
##================================================================
============
##----------------------------------------------------------------
##  V.3 OASIS
##----------------------------------------------------------------
##================================================================
============
#
```

```
#-------
# -  Inputs files
#-------
#
cp ${LOCAL_INCPL}/areas_025_T106 areas
cp ${LOCAL_INCPL}/grids_025_T106 grids
#cp ${LOCAL_INCPL}/masks_025_T106 masks
cp ${LOCAL_INCPL}/masks_025_T106_mod masks
# for mozaic interpolation
#cp ${LOCAL_INCPL}/mozaic_orca025_T106 orcat106
cp ${LOCAL_INCPL}/mozaic_orca025_T106_mod orcat106
## !E cp ${LOCAL_INCPL}/mozaic_t106_orca2closea_70neig_r8 t106orca
#-------
# -  restart
#-------
if [ ${DATE_BEGIN_SUBJOB} -eq ${DATE_BEGIN_CPEXP} ]
then
   cp ${STOREIN}/Res/Cpl/cpl_restart0_flxatmos flxatmos
   cp ${STOREIN}/Res/Cpl/cpl_restart0_sstocean sstocean
else
   cp ${LOCAL}/Res/Cpl/${CEXPER}_${DATE_RESTART}_flxatmos flxatmos
   cp ${LOCAL}/Res/Cpl/${CEXPER}_${DATE_RESTART}_sstocean
sstocean
fi
#
##
##
###########################################################################
############
###########################################################################
############
##
##    VI. Run the model!
##
###########################################################################
############
###########################################################################
############
##
##
#
#
ls -alF
#
# NEC f90 environment variables
# Direct Acess files record length in bytes
export F_RECLUNIT=BYTE
# set line size for standard output
```

```
export F_SYSLEN=1000
# number of runtime errors before aborting
# if oasis stops
###export F_ERRCNT=999999
export F_ERRCNT=0
export F_ABORT=YES
export F_ERRHALT=YES
#
export ASSCOM="echo "
export MACH=IEEE
echo "$MACHINE OK"
#
uname -a
pwd
#
# test sizes of executables
size oceanx oasis atmosx
# Oasis and model launching
mpiexec -v -np 1 -max_np $(( 1+$NCPUSORCA+$NCPUSECHAM )) oasis
#mpirun -v -np 1 -max_np $(( 2+$NCPUSECHAM )) oasis

stat=$?
if [ $stat -ne 0 ] ; then
exit
fi

ls -alF $LOCAL/*
#
NIT_END=$( cat time.step )
echo Last time step of ORCA: $NIT_END
#
#
##
##
#####################################################################
############
#####################################################################
############
##
##    VII. post-processing
##
#####################################################################
############
#####################################################################
############
##
##
#
```

```
(( MONTHS_DONE = ${MONTHS_DONE} + ${SUBJOB_DUR} ))
[ ${MONTHS_DONE} -eq ${JOB_DUR} ] && JOB_FINISHED=1 ||
JOB_FINISHED=0
#
#
##
##
##-------------------------------------------------------------
##    VII.1  ECHAM4
##-------------------------------------------------------------
##============================================================
#
# model outputs
cp atmosx.prt ${STOUT_OUT}/${CEXPER}_${DATES}_atm.prt || exit
# namelist
####cp namatmos $NAMDIR/${CEXPER}_${DATES}_namatmos || exit
#
# Results
cp atm*__* ${STOUT_AFILES}/atm_${CEXPER}_${DATES}.grib || exit
#rsh refosco ${ATMPOSTPROC} T106
${MODNAME}/${CEXPER}/Atm/Results
${MODNAME}/${CEXPER}/Atm/Results ${YEAR_END_SUBJOB}
${MONTH_END_SUBJOB} atm_${CEXPER}_${DATES}.grib ">&"
outpostproc${YEAR_END_SUBJOB}${MONTH_END_SUBJOB}
# restarts
#
#
POSTPROC=/cabernet${HOMEC}/${MODNAME}/${CEXPER}/post/postprocn
c
#Andru: inclusion of the second argument CEXPER
###rsh refosco -n nohup ${POSTPROC} atm_${CEXPER}_${DATES}.grib
${CEXPER} "&" &
##rsh refosco ${POSTPROC} atm_${CEXPER}_${DATES}.grib ${CEXPER} <
/dev/null ">&" post.out "&"

#
#
if [ ${JOB_FINISHED} -eq 1 ]
then
   for unit in 31 32 35 36 37 ; do
        cp unit.$unit
${STOREOUT_ARES}/${CEXPER}_${DATE_END_SUBJOB}_unit.$unit
   done
else
   for unit in 31 32 35 36 37 ; do
       cp unit.$unit
${STOREOUT_ARES}/${CEXPER}_${DATE_END_SUBJOB}_unit.$unit
```

```
      mv unit.$unit
${LOCAL}/Res/Atm/${CEXPER}_${DATE_END_SUBJOB}_unit.$unit
   done
fi
#
##============================================================
##------------------------------------------------------------
##  VII.2 ORCA
##------------------------------------------------------------
##============================================================
#
#  outputs
cp oceanx.prt ${STOUT_OUT}/${CEXPER}_${DATES}_oce.prt || exit
cp ocean.output_01 ${STOUT_OUT}/${CEXPER}_${DATES}_oce.output_01 ||
exit
cp ocean.output_02 ${STOUT_OUT}/${CEXPER}_${DATES}_oce.output_02 ||
exit
cp ocean.output_03 ${STOUT_OUT}/${CEXPER}_${DATES}_oce.output_03 ||
exit
cp ocean.output_04 ${STOUT_OUT}/${CEXPER}_${DATES}_oce.output_04 ||
exit
# Results
##!E for file in *_grid_* *_diagap* *_diaznl* *icemod*
for file in *1mAV*
  do
   [ -f $file ] && cp $file ${STOUT_OFILES}/${file}
  done
[ -f output.abort.nc ] && cp output.abort.nc ${STOUT_OFILES}/output.abort.nc
# remote storage of restarts
mv ${CEXPER}_*_restart_000.nc orcaini_000.nc
mv ${CEXPER}_*_restart_001.nc orcaini_001.nc
mv ${CEXPER}_*_restart_002.nc orcaini_002.nc
mv ${CEXPER}_*_restart_003.nc orcaini_003.nc
if [ ${JOB_FINISHED} -eq 1 ]
then
   cp orcaini_000.nc
${STOREOUT_ORES}/${CEXPER}_${DATE_END_SUBJOB}_restart_000.nc
   cp orcaini_001.nc
${STOREOUT_ORES}/${CEXPER}_${DATE_END_SUBJOB}_restart_001.nc
   cp orcaini_002.nc
${STOREOUT_ORES}/${CEXPER}_${DATE_END_SUBJOB}_restart_002.nc
   cp orcaini_003.nc
${STOREOUT_ORES}/${CEXPER}_${DATE_END_SUBJOB}_restart_003.nc
else
# local storage of restarts
   cp orcaini_000.nc
${STOREOUT_ORES}/${CEXPER}_${DATE_END_SUBJOB}_restart_000.nc
```

```
   cp orcaini_001.nc
${STOREOUT_ORES}/${CEXPER}_${DATE_END_SUBJOB}_restart_001.nc
   cp orcaini_002.nc
${STOREOUT_ORES}/${CEXPER}_${DATE_END_SUBJOB}_restart_002.nc
   cp orcaini_003.nc
${STOREOUT_ORES}/${CEXPER}_${DATE_END_SUBJOB}_restart_003.nc
   mv orcaini_000.nc
${LOCAL}/Res/Oce/${CEXPER}_${DATE_END_SUBJOB}_restart_000.nc
   mv orcaini_001.nc
${LOCAL}/Res/Oce/${CEXPER}_${DATE_END_SUBJOB}_restart_001.nc
   mv orcaini_002.nc
${LOCAL}/Res/Oce/${CEXPER}_${DATE_END_SUBJOB}_restart_002.nc
   mv orcaini_003.nc
${LOCAL}/Res/Oce/${CEXPER}_${DATE_END_SUBJOB}_restart_003.nc
fi
#
##============================================================
##------------------------------------------------------------
##   VII.3  OASIS
##------------------------------------------------------------
##============================================================
#
#  outputs
cp Oasis.prt ${STOUT_OUT}/${CEXPER}_${DATES}_Oasis.prt
cp cplout ${STOUT_OUT}/${CEXPER}_${DATES}_cplout
# namelist
####cp namcouple $NAMDIR/${CEXPER}_${DATES}_namcouple
# restarts
mv ocesst sstocean
cat oceice >> sstocean
if [ ${JOB_FINISHED} -eq 1 ]
then
   cp sstocean
${STOREOUT_CRES}/${CEXPER}_${DATE_END_SUBJOB}_sstocean
   cp flxatmos
${STOREOUT_CRES}/${CEXPER}_${DATE_END_SUBJOB}_flxatmos
else
   cp sstocean
${STOREOUT_CRES}/${CEXPER}_${DATE_END_SUBJOB}_sstocean
   cp flxatmos
${STOREOUT_CRES}/${CEXPER}_${DATE_END_SUBJOB}_flxatmos
   mv sstocean
${LOCAL}/Res/Cpl/${CEXPER}_${DATE_END_SUBJOB}_sstocean
   mv flxatmos
${LOCAL}/Res/Cpl/${CEXPER}_${DATE_END_SUBJOB}_flxatmos
fi
#
##
```

```
##
#####################################################################
###########
#####################################################################
###########
##
##    VIII call of the next subjob
##
#####################################################################
###########
#####################################################################
###########
##
##
#-------
# year month day of the beginning of next subjob
#-------
valid_date $MONTH_END_SUBJOB $(( $DAY_END_SUBJOB + 1 ))
$YEAR_END_SUBJOB | read MONTH_NEXT_SUBJOB
DAY_NEXT_SUBJOB YEAR_NEXT_SUBJOB
#-------
# update the log file
#-------
echo "`date` $YEAR_END_SUBJOB $MONTH_END_SUBJOB
$DAY_END_SUBJOB done" >> ${CEXPER}.log
cp ${CEXPER}.log $SCRIPTDIR/
#-------
# save the previous output file
#-------
# CMDFRONT "mv $SCRIPTDIR/output_LO1.1
$SCRIPTDIR/${CEXPER}_${DATE_RESTART}_JobOutput"
#[ -f ${SCRIPTDIR}/output_LO1.1 ] && mv ${SCRIPTDIR}/output_LO1.1
${SCRIPTDIR}/${CEXPER}_${DATE_RESTART}_JobOutput
#[ -f ${SCRIPTDIR}/stderr_LO1.1 ] && mv ${SCRIPTDIR}/stderr_LO1.1
${SCRIPTDIR}/${CEXPER}_${DATE_RESTART}_JobStderr
#
if [ ${JOB_FINISHED} -eq 0 ] ; then
#-------
# The Job is not finished, we must call Job for to perform a new subjob
#-------
##then
# update the trace file
#-------rm -f ${CEXPER}.trace
cat <<EOF >${CEXPER}.trace
$YEAR_NEXT_SUBJOB $MONTH_NEXT_SUBJOB $DAY_NEXT_SUBJOB
$MONTHS_DONE
EOF
cp ${CEXPER}.trace $HOMEDIR/
```

```
#
   echo ""
   echo ""
   echo
"%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%"
   echo "%%%%%%%%%%%%%%   NEW SUBJOB
%%%%%%%%%%%%%%"
   echo
"%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%"
   echo ""
   echo ""
#
# call again Job to perform a subjob
cd $HOMEDIR
qsub run.couple
fi
```