INTERREG-V OCEAN INDIEN 2014-2020

Projet de Recherche

RENOVRISK-CYCLONES ET CHANGEMENT CLIMATIQUE

Axe-1 OT-1 OS-01a - Action I-3 TF

Livrable 17

Tutoriel pour la réalisation de simulations couplées de cyclones (installation et utilisation du modèle intégré) sur le bassin Indien sud-ouest

J. Pianezze(1), C. Barthe(1), S. Bielli(1), E. Cordier(2), D. Heron(1), S. Malardel(1), P. Tulet(1), H. Veremes(1)

(1) LACy)

(2) Observatoire des Sciences de l'Univers de la Réunion (OSU-R)

Octobre 2019

Ce livrable est associé à la sous-action 2.3 intitulée :

Simulations numériques à haute-résolution.

Il consiste en un tutoriel en langue anglaise permettant de lancer des simulations numériques couplées océan - vagues - atmosphère à partir du modèle intégré développé dans le cadre du projet

Contents

Chapter 1

Introduction

This manual describes how to set up and run a coupled simulation between Meso-NH/SurfEx, WW3 and CROCO models with the OASIS coupler. It does not provide detailed explanations on the use of the different models, it is therefore advisable to have read the documentation of each models before reading this report.

The developments were carried out as part of an Ocean-Wave-Atmosphere project group initiated by the CNRM around the SurfEx model (Voldoire et al., 2017) and were extended to the use of CROCO model (Pianezze et al., 2018). The coupling algorithm implemented allows Ocean-Wave-Atmosphere (OWA) coupled simulations. The possible exchanges between the different OWA component are shown in the Figure 1.1.

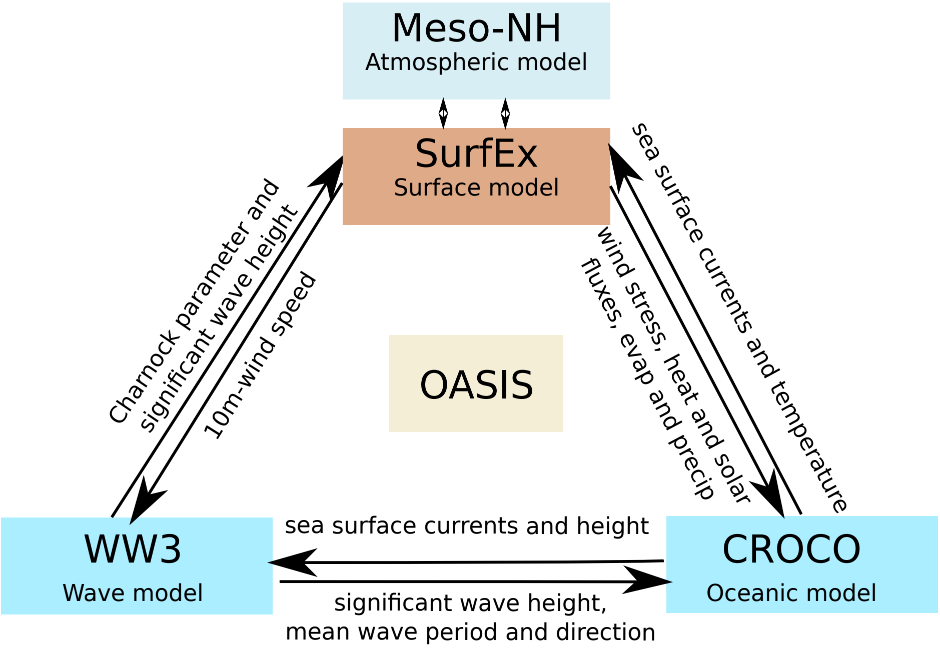
Meso-NH
Atmospheric model
SurfEx
Surface model
OASIS
WW3
Wave model
CROCO
Oceanic model
Charnock parameter and significant wave height
10m-wind speed
sea surface currents and temperature
wind stress, heat and solar fluxes, evap and precip
sea surface currents and height
significant wave height, mean wave period and direction

Figure 1.1: Diagram illustrating the different exchanges between Meso-NH/SurfEx, WW3 and CROCO models

The Figure 1.2 shows an example of a coupled simulation with OASIS between an atmospheric and an oceanic models to introduce the LAG concept explained in details in the OASIS documentation. The aim of this scheme is to explain the role of the rstrt.nc files you will have to create in the configuration set-up chapter (Chapter 3).

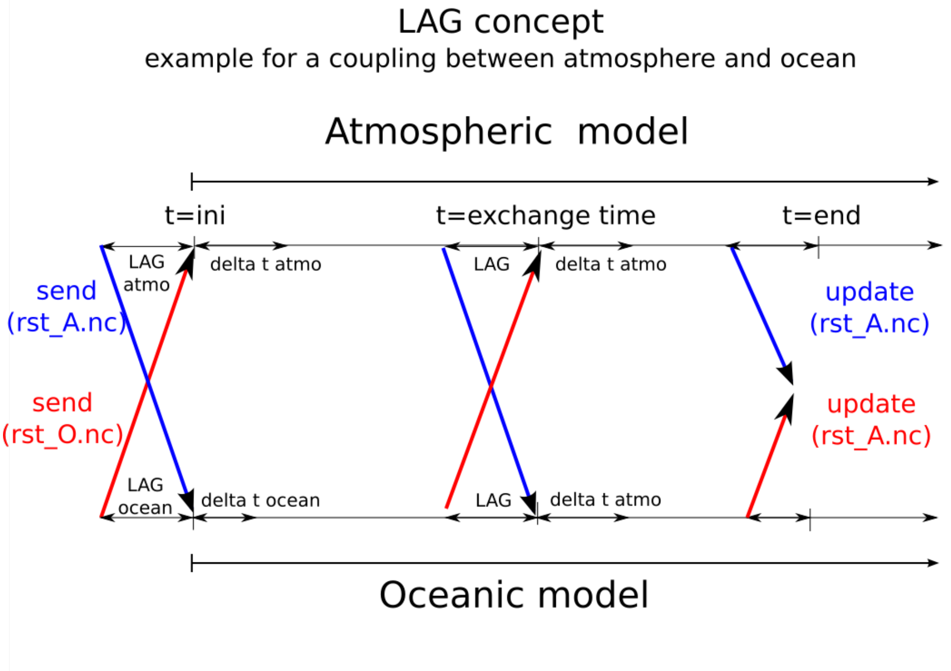

Figure 1.2: Diagram illustrating the different exchanges between Meso-NH/SurfEx and CROCO models

At the initial time (at t=ini), the atmospheric and oceanic models receive the coupling fields located in the rtsrt.nc files. As soon as the time steps of the models correspond to the exchange time (at t=exchange time) present in the namcouple, EXSEG1.nam (for Meso-NH), ww3_shel.inp (for WW3) and namcouple (for CROCO), the models send fields to the receiver models. At the end of the simulation (at t=end) the exchange fields are stored in the rstrt.nc file used at the beginning of the next simulation. This allows you to start a new simulation after this one.

Note that the fields in the initial rstrt.nc file will be overwritten at the end of the simulation.

Chapter 2

Installation and compilation

In this chapter, the installation and compilation of the different models are presented. It is possible to automatically compile OASIS during the compilation of Meso-NH or WW3, but I recommend you to compile this library by yourself to verify the different link and libraries used in the compilation.

2.1 Libraries : netcdf, zlib, hdf, mpi,, ...

It is important to compile the coupler and all models with the same netcdf, mpi, ... libraries and compilation's options. If you decide to change one of these libraries or compilation's options, you will have to recompile everything with it. Note that on beaufix, it is not necessary to compile these libraries because they are already available via the "module" command.

After the installation of these libraries (if it was necessary), you must add an environment variable into your .bashrc (for bash environment):

```
export NETCDF_CONFIG=’path/to/nc-config’
```

This environment variable will be used when creating links to the netcdf library during WW3, Meso-NH and CROCO compilations.

On beaufix, you have to put these lines into your .bashrc file:

to put in your .bashrc

```
module rm intel    ; module load intel/16.1.150 ;
module rm intelmpi ; module load intelmpi/5.1.2.150 ;
module rm netcdf   ; module load netcdf/4.4.0 ;

export NETCDF_CONFIG=/opt/softs/libraries/ICC16.1.150/netcdf-4.4.0/bin/nc-config
export WWATCH3_NETCDF=NC4
```

It is important to verify the libraries (mpi) used to compile netcdf module. It has to be the same as the one used for the compilation of the models. The following command give you these information :

```
nc-config --all
```

2.2 Download the models

For the purpose of this tutorial you can download the version of the models via the following git repository:

```
git clone https://github.com/JorisPianezze/tuto_cpl_models.git
```

In this git repository, you will find these models:

```
croco-v1.0
MNH-V5-4-1
oasis3-mct_v4
README.md
ww3_v5-16
```

2.3 Compilation of OASIS

The information presented in this section is described in detail in the OASIS user guide[1]. The OASIS source code is either available in the Meso-NH version in the src/LIB/oasis3-mct.tar (if using a Meso-NH version greater than or equal to 5.2.0) or by downloading it from the OASIS website. I recommend that you use the version available in the git repository (Section 2.2) containing the configuration for the compilation on beaufix. The version of OASIS available in the Meso-NH suite is OASIS3-MCT version 3.0 but the version 4.0 is used in this tutorial.

In the folder containing the OASIS code, there are

```
doc/ : Documentation of OASIS
lib/ : Librairies mct, psmile and scrip
examples/ : Directory with examples
util/lucia/ : Tools to analyse the exchange
util/make_dir/ : Directory to compile OASIS : Makefile, ...
```

To compile OASIS manually, it is necessary to :

1. go in the util/make_dir directory ;
2. and to modify the file make.inc and make.beaufix according to your path.

Before the compilation, you have to verify if the $NETCDF_CONFIG variable is defined. Then you can run the following command on the util/make_dir directory:

to compile OASIS

```
make realclean -f TopMakefileOasis3
make -f TopMakefileOasis3
```

At the end of the compilation, you have to look at the COMP.err and COMP.log files : if no error is inside, it means that compilation of OASIS is successfully done.

Once the compilation is done, you must define the OASISDIR environment variable into your .bashrc:

[1] https://verc.enes.org/oasis/

to put in your .bashrc

```
export OASISDIR='path/to/directory/BEAUFIX_oa3-mct'
```

This environment variable will be used to link to the OASIS library during the compilation of the WW3, Meso-NH and CROCO models.

To verify the compilation of OASIS, test cases are available under oasis_dir/examples/.

2.4 Compilation of Meso-NH/SurfEx

To benefit from developments related to OWA coupling, it's necessary to use Meso-NH version higher than 5.2.0. You can download mesonh code on mesonh website[2] but I recommend that you use the version available in the git repository (Section 2.2) containing the configuration for the compilation on beaufix.

To compile Meso-NH you have to go to the folder src/ and run the following commands (on beaufix):

```
export VER_CDF=CDFBASHRC
export VER_OASIS=OASISBASHRC
./configure
```

At the end of the ./configure command a profile_mesonh* file is created in the conf directory. You have to find these lines in the file profile_mesonh* :

```
export VER_CDF=CDFBASHRC
export VER_OASIS=OASISBASHRC
```

The configure command allows you to define the environment variables useful for the compilation and execution of Meso-NH and unzip the GRIBAPI, NETCDF, ... libraries if necessary.

If you are using beaufix, you need to modify the script job_make_mesonh_BullX and put the path of the profile_mesonh you just created, then you can run the compilation with:

```
sbatch job_make_mesonh_BullX
```

At the end of the compilations, you have to see on the joboutput, 'up to date' or 'est à jour'. If it is not the case, you need to run the compilation again. On other system (local by example), you have to run

```
export VER_CDF=CDFBASHRC
export VER_OASIS=OASISBASHRC
./configure
. ../conf/profile_mesonh
make
make installmaster
```

If you need to compile a VER_USER (source modification), you have to run the following commands:

[2] http://mesonh.aero.obs-mip.fr/mesonh54

```
export VER_CDF=CDFBASHRC
export VER_OASIS=OASISBASHRC
export VER_USER=NAMEOFYOURVERUSER
./configure
```

On beaufix, you need to modify the file job_make_mesonh_user_BullX, put the path to the new profile_mesonh and run it with:

to compile Meso-NH

```
sbatch job_make_mesonh_user BullX
```

On other system, you have to run :

```
export VER_CDF=CDFBASHRC
export VER_OASIS=OASISBASHRC
export VER_USER=NAMEOFYOURVERUSER
./configure
. ../conf/profile_mesonh
make user
make installuser
```

Notes: The make command calls several files (all located in the src folder):

- Makefile: compilation of the different libraries, if necessary.
  - Rules.$ARCH$(F).mk: definition of compilation options, etc...
    - Makefile.MESONH.mk: definition of links to libraries.

A series of program have been compiled, for our studies we will use the following ones:

Name exe.	**Function**	**Namelist**
PREP_PGD	prepare the hor. grid	PRE_PGD1.nam
PREP_REAL_CASE	prepare the ver. grid and IC and LBC	PRE_REAL1.nam
MESONH	launch the simulation	EXSEG1.nam
DIAG	diag after simulation	DIAG1.nam

Table 2.1: Executable of Meso-NH/SurfEx used in the tutorial. IC is for Initial Conditions and LBC for Lateral Boundary Conditions.

2.5 Compilation of WW3

To benefit from developments related to OWA coupling, it is necessary to use a version of WW3 higher than 5.12. It is possible to download this version directly from the NCEP or IFREMER servers, these versions differ from their environment. This tutorial is made with version of WW3 coming from IFREMER servers, it is therefore recommended to use this kind of version. I recommend that you use the version available in the git repository (Section 2.2) containing the configuration for the compilation on beaufix. The following steps are described in the WW3 manual (/ww3_dir/manual/*.pdf) but are recalled here for clarity.

If it is not already done, you have to add in the .bashrc the version of netcdf used (4 in our case), if using netcdf3, refer to the WW3 doc :

```
export WWATCH3_NETCDF=NC4
```

Then it is necessary to prepare the compilation by defining the environment variables for WW3, go to the ww3_v5-16/model/bin directory and verify the path in the wwatch3.env file. Then you have to load the WWATCH3_ENV variable with:

```
export WWATCH3_ENV=$PWD/wwatch3.env
```

To verify the links, you can clean the WW3 directory:

```
./w3_clean -c
```

Then you need to prepare the files for compilation. For beaufix and for a OWA coupled simulation, you can use:

```
./w3_setup .. -c beaufix -s OASACM_OASOCM -q
```

At this step, the files link.beaufix and comp.beaufix are copied in comp and link files: if you are using beaufix, you don't have to change these files. The swicth_OASACM_OASOCM is copyed in the switch file. Note that it is in the switch file you have to defined the physical parameterizations you want to use. OASACM and OASOCM are to activate the coupling with an atmospheric model and an oceanic model, respectively (cf. manual of WW3).

Then, you can choose between compiling all the WW3 programs:

to compile WW3

```
./w3_make
```

or compile the programs one by one:

```
./w3_make ww3_nameprog
```

To recompile WW3 you have to run:

```
./w3_clean
./w3_new
./w3_make
```

To clean everything you have to run:

```
./w3_clean -c
```

To verify the compilation, many test cases are available under ww3_dir/regtests/.

A series of program have been compiled, for our studies we will use the following ones:

Name exe.	Function	Namelist
ww3_grid	prepare the grid files	ww3_grid.inp
ww3_bound	prepare the LBC	ww3_bound.inp
ww3_prnc	prepare the forcing fields (spinup)	ww3_prnc.inp
ww3_shel	launch the simulation	ww3_shel.inp
ww3_ounf	create netcdf file	ww3_ounf.inp

Table 2.2: Executable of WW3 used in the tutorial. LBC is for Lateral Boundary Conditions.

2.6 Compilation of CROCO

The CROCO model is available here: http://www.croco-ocean.org/. In this tutorial we use the version 1.0 and I recommend that you use the version available in the git repository (Section 2.2) containing the configuration for the compilation on beaufix.

First, you have to add in the .bashrc file the following variable:

to put in your .bashrc

```
export CROCODIR=/path/to/croco-v1.0/
```

Then, you have to copy the croco-v1.0/config_FANTALA_HR_10coeurs_CPLOA_CPLOW or rename it. In this directory, you have:

```
clean.sh : script used to clean the config directory
jobcomp : script to launch the compilation
Modif_Joris : directory containing the configuration files
```

In Modif_Joris directory you have to modify cppdefs.h and param.h files, before launching the compilation. The CPP keys used and the domain location must be entered in the files cppdefs.h and param.h.

In cppdefs.h file, you can choose to do coupling or not with following cpp keys :

```
# define OA_COUPLING
# define OW_COUPLING
```

The cpp keys OA_COUPLING and OW_COUPLING are to do coupling with an atmospheric model and an oceanic wave model, respectively. Then you have to give a name to your config, this name will be used in the file param.h:

```
/* Configuration Name */
# define FANTALA_HR
```

In param.h, you have to put the number of points for the CROCO domain you have (to know these points, you have to create your grid before and do a ncdump -h on th rstrt_SAVE.nc file, cf. Section 3.6).

```
#  elif defined  FANTALA_HR
parameter (LLm0=1205,   MMm0=724,   N=32)
```

The number of cores used (NNODES) during the simulation must be entered in the param.h file and the code must be recompiled each time the number of cores used is changed:

```
#ifdef MPI
integer NP_XI, NP_ETA, NNODES
parameter (NP_XI=5, NP_ETA=2,  NNODES=NP_XI*NP_ETA)
parameter (NPP=1)
parameter (NSUB_X=1, NSUB_E=1)
```

NP_XI and NP_ETA correspond to the number of cores in longitude and latitude, respectively.

Unlike WW3 and Meso-NH, there are no initial conditions or lateral boundary conditions preparation programs: these steps are performed using the matlab toolbox (cf Section 3.6).

To compile CROCO, you have to launch the following command:

to compile CROCO

```
./jobcomp
```

One program has been compiled and is use in this tutorial:

Name exe.	Function	Namelist
croco	launch the simulation	croco.in

Table 2.3: Executable of CROCO used in the tutorial

Chapter 3

Configuration set-up

3.1 Download the scripts used in this section

All the files present in this section are downloadable via a git repository. To download the files on your computer or on beaufix you can launch the following command:

```
git clone https://github.com/JorisPianezze/scripts_tuto_cpl_owa.git
```

3.2 Tree of the runs

A lot of files are used during an OWA coupled simulations, it is therefore recommended to have a readable tree. For this reason I propose to use this kind of tree, present in the git directory :

tree of the simulations

```
0_input_oasis          ! corresponding to namelist of OASIS
1_input_mnh            ! corresponding to initial conditions for Meso-NH
2_input_ww3            ! corresponding to initial conditions for WW3
3_input_croco          ! corresponding to initial conditions for CROCO
A_run_ww3_spinup       ! corresponding to the spinup WW3 run directory
B_run_mnh_croco_rmp    ! corresponding to the Meso-NH/CROCO couped run directory
C_run_mnh_ww3_croco_rmp ! corresponding to the Meso-NH/WW3/CROCO coupled run directory
```

3.3 Preparation of OASIS

3.3.1 Prepare the namelist for the coupled simulation : namcouple

In this Section, the namelist namcouple for OASIS coupler located in the **0_input_oasis** is presented. The namcouple file is the configuration file for OASIS coupler containing all information related to exchanges between models. It responds to a precise syntax that is clearly explained in the OASIS3-MCT user manual. He doesn't admit white lines! There is no syntax check of this file : the slightest error implies a crash with no indication.

First part of the file: general settings

First of all, it is necessary to identify the general parameters of the problem:

- the number of coupling fields

- the number of models
- the names of the models : mesonh (for Meso-NH), wwatch (for WW3) and crocox (for CROCO)
- the duration of the simulation in seconds
- the debug degree: amount of information that can be used for debugging

All these parameters allow the first part of the namcouple file to be written:

```
# This is a typical input file for OASIS3-MCT.
# Keywords used in previous versions of OASIS3
# but now obsolete are marked "Not used"
# Don't hesitate to ask precisions or make suggestions (oasishelp@cerfacs.fr).
#
# Any line beginning with # is ignored. Blank lines are not allowed.
#
#########################################################################
$NFIELDS
# The number of fields described in the second part of the namcouple.
#
2
$END
#########################################################################
$NBMODEL
# The number of models in this experiment + their names (6 characters)
#
2  mesonh   wwatch
$END
###########################################################################
$RUNTIME
# The total simulated time for this run in seconds
#
70
$END
###########################################################################
$NLOGPRT
# Amount of information written to OASIS3-MCT log files (see User Guide)
# Premier nombre: quantite d'information pour debugger
# Second nombre: statistics des processeurs
30 3
$END
```

Second part of the file: exchanges

The rest of the namcouple file concerns the exchanges between models. It is necessary to have a precise idea of the coupling algorithm that we want to set up. First of all, it is necessary to decide on a coupling period for the different fields.

It is not possible to give a coupling period less than the largest time step of the models. In addition, this coupling period must be proportional to the time steps of the models and must be positive, real ones are not allowed. OASIS only sends data when a sending date corresponds to a receipt date, it is therefore necessary to ensure that a coupling period is provided that allows the two models to "meet".

It is also necessary to choose the type of interpolation for the transition from one grid to another.

Notes: It is possible to choose different exchange periods and interpolations for each exchanged field.

Here we show an example of the namcouple parameters for the exchange of the SST in the context of the coupling between Meso-NH and CROCO. This case is generic enough to be adapted to other exchanges.

```
1  #~~~~~~~~~~~
2  # Field 1 : SEA SURFACE TEMPERATURE
3  #~~~~~~~~~~~
4  SRMSSTV0 MNH__SST 1 600 1 rst_0.nc EXPOUT
5  321 194 100 87 ocnt mnht LAG=+100
6  R 0 R 0
7  LOCTRANS BLASOLD SCRIPR
8  AVERAGE
9  1.0 1
10 CONSTANT 273.15
11 BILINEAR LR SCALAR LATLON 1
```

- (line 4) : we have here the field identifier with first the one on the source side (SRMSSTV0) and then the one on the target side (MNH__SST). The following parameter is not used but must be present to ensure compatibility with the old version of OASIS. Then we have the exchange time (600 seconds), the number of operations that will be applied to this field, the name of the start/restart file and the status of the field.
- (line 5) : we find the dimensions of the source (ocnt) and target (mnht) grids, their name and the value of the LAG. The grids here are the ones in the grids.nc, masks.nc and areas.nc files. In that case the LAG is equal to the time step of the oceanic model.
- (line 6) : source and target grid characteristics ("R": regional and "0": no overlap of grid points).
- (lines 7 to 11) : list of operations to be performed on this field followed by the parameters of each of these operations:
  - LOCTRANS : temporal transformation of the AVERAGE type (line 8)
  - BLASOLD : multiplication of the field by "1" (line 9 to which is added the value 273.15 (line 10) : conversion from Kelvin to Celsius
  - SCRIPR : interpolation whose parameters are provided (line 11) : here bilinear interpolation

You have to repeat these few lines for all exchanges fields. The identifier names can be find in the model documentations. The LAG must be the time step of the source side (Chapter 1).

3.3.2 List of files in OASIS input directory

files required for OASIS

```
0_input_oasis
| -- namcouple       ! namelist for the coupled simulation
```

3.4 Preparation of Meso-NH/SurfEx

In this section, we will use the scripts located in the directory **1_input_mnh** from the git repository (Section 3.1).

The initial and boundary conditions for Meso-NH/SurfEx can be done exclusively on beaufix. To prepare the initial and boundary conditions for Meso-NH/SurfEx it is necessary to used (at minimum), the following executables (Tab. 2.1):

- PREP_PGD : to prepare the physiographic data file : definition of the horizontal grid (number of points and grid resolution), type of cover, etc ...
- PREP_REAL_CASE : to prepare the initial and boundary conditions for the atmospheric fields (wind speed, potential temperature, mixing ratio, ...)

The Ocean-Wave-Atmosphere coupling only impacts the simulation (MESONH program). The initial and coupling files for Meso-NH are therefore obtained in the same way as for a forced simulation (PREP_PGD and PREP_REAL_CASE or PREP_IDEAL_CASE).

3.4.1 Prepare the physiographic data : PREP_PGD

First, you have to put in your .bashrc the files used to prepare the grid :

to put in your .bashrc

```
export MESONH=/home/cnrm_other/ge/mrmh/payart/mesonh
```

In this directory, there are the files ECOCLIMAP, SAND_HWSD_MOY, CLAY_HWSD_MOY and gtopo30 used to defined the type of cover and the orography. If you have no access to beaufix, these files are downloadable from the mesonh website[1].

The number of points and grid size is defined with NIMAX, NJMAX, XDX and XDY variables in the namelist PRE_PGD1.nam.

To launch the PREP_PGD executable you have to modify the different path according to your configuration files and launch the script run_prep_pgd.sh:

```
sbatch run_prep_pgd.sh
```

You have to read at the end of the OUTPUT_LISTING0 file : "PREP_PGD ends correctly". At the end of this step you will find two files in your directory : a netcdf file with fields corresponding to your domain and a des file containing the namelist used (in some program this file is empty). These files have to be used in the next section.

[1] http://mesonh.aero.obs-mip.fr/mesonh54/Download

3.4.2 Prepare the initial and boundary conditions : PREP_REAL_CASE

To prepare the initial and boundary conditions, you need to download the forcing large scale fields you will use : AROME, ECMWF or NCEP. For that purpose, you can contact Soline Bielli[2]. If you want to do a coupling with an oceanic wave model (WW3, ...), you have to ask to Soline Bielli to extract the wave energy spectra and 10m wind speed, in addition to the other fields.

Once the forcing fields are extracted, you need to modify the namelist PRE_REAL1.nam according to the vertical grid you want. Then you need to adapt the path in the script run_prep_real_case.sh depending on your environment and launch this script:

```
sbatch run_prep_real_case.sh
```

You have to read at the end of the OUTPUT_LISTING0 file : "PREP_REAL_CASE ends correctly". At the end of this step you will find two files by forcing time in your directory : a netcdf file with fields corresponding to the atmospheric fields interpolated on your grid and a des file containing the namelist used (in some program this file is empty). These files have to be used in the next section and will serve as initial and boundary conditions.

For the coupling purposes, it is necessary to create a rstrt_SAVE.nc file that will allow the model starting and restarting. On beaufix, you need to load the python module before launching this script:

for beaufix users

```
module load python
```

Then you have to modify the name of the netcdf file from the PREP_REAL_CASE corresponding to the initial time in the script create_restart_file_from_PRC_MNH_file.py. You can create the rstrt_SAVE.nc file using :

```
python create_restart_file_from_PRC_MNH_file.py
```

3.4.3 Prepare the namelist for the coupled run : EXSEG1.nam

The last step for preparing the configuration for Meso-NH is to create the EXSEG1.nam namelist corresponding to the simulated namelist. Here, only the lines corresponding to the coupling stuff are presented. The other namelists can be found in the Meso-NH documentation.

part of EXSEG1.nam corresponding to the coupling

```
&NAM_OASIS LOASIS      = .TRUE.,
           LOASIS_GRID = .TRUE.,
           CMODEL_NAME = 'mesonh' /

&NAM_SFX_WAVE_CPL XTSTEP_CPL_WAVE = 600.0,
                  CWAVE_U10       = 'MNH__U10',
                  CWAVE_V10       = 'MNH__V10',
                  CWAVE_CHA       = 'MNH__CHA',
                  CWAVE_UCU       = 'MNH_WSSU',
                  CWAVE_VCU       = 'MNH_WSSV',
                  CWAVE_HS        = 'MNH___HS',
```

[2] soline.bielli-bousquet .at. univ-reunion.fr

```
                  CWAVE_TP        = 'MNH___TP' /

&NAM_SFX_SEA_CPL XTSTEP_CPL_SEA  =  600.0,
                 CSEA_FWSU       = 'MNH_TAUX',
                 CSEA_FWSV       = 'MNH_TAUY',
                 CSEA_HEAT       = 'MNH_HEAT',
                 CSEA_SNET       = 'MNH_SNET',
                 CSEA_WIND       = '        ',
                 CSEA_FWSM       = '        ',
                 CSEA_EVAP       = '        ',
                 CSEA_RAIN       = '        ',
                 CSEA_SNOW       = '        ',
                 CSEA_WATF       = 'MNH_EVPR',
                 CSEA_PRES       = '        ',
                 CSEA_SST        = 'MNH__SST',
                 CSEA_UCU        = '        ',
                 CSEA_VCU        = '        ' /
```

Namelists specific to coupling have been created and are presented in the following section, the other namelists concerning EXSEG1.nam are identical. The character strings as to be the same as ones defined in the namcouple file. If you don't want to use the coupling with an oceanic wave model you have to put XTSTEP_CPL_WAVE=-1.0 or remove the namelist NAM_SFX_WAVE_CPL.

Namelist NAM_OASIS

- LOASIS : Flag to activate OASIS coupling ;
- LOASIS_GRID : Flag to create rmp* files during simulation ;
- CMODEL_NAME : Name of the model, must be the same as the one in the namcouple.

Namelist NAM_SFX_SEA_CPL

- XTSTEP_CPL_SEA : Exchange time step for oceanic fields

The CSEA_* character strings correspond to the OASIS identifiers of the variables exchanged between the ocean and the atmosphere and must be identical to those in the namcouple file. This identifier must contain 8 characters long. Putting an empty space of 8 characters corresponds to the default value: no exchange of this variable. These variables correspond to the name of the OASIS identifier for:

- CSEA_FWSU : the zonal momentum flux sent from the atmosphere to the ocean ;
- CSEA_FWSV : the meridian momentum flux sent from the atmosphere to the ocean ;
- CSEA_HEAT : the non-solar heat flux sent from the atmosphere to the ocean ;
- CSEA_SNET : the net radiation flux sent from the atmosphere to the ocean ;
- CSEA_WIND : the wind speed at 10m sent from the atmosphere to the ocean ;
- CSEA_FWSM : the momentum flux sent from the atmosphere to the ocean ;

- CSEA_EVAP : the evaporation sent from the atmosphere to the ocean ;
- CSEA_RAIN : the rain sent from the atmosphere to the ocean ;
- CSEA_SNOW : the snow sent from the atmosphere to the ocean ;
- CSEA_WATF : the freshwater sent from the atmosphere to the ocean ;
- CSEA_PRES : the surface pressure sent from the atmosphere to the ocean ;
- CSEA_SST : the sea surface temperature sent from the ocean to the atmosphere ;
- CSEA_UCU : the zonal current sent from the ocean to the atmosphere ;
- CSEA_VCU : the meridian current sent from the ocean to the atmosphere.

Namelist NAM_SFX_WAVE_CPL

- XTSTEP_CPL_WAVE : exchange time step for oceanic wave fields

The CWAVE_* character strings correspond to the OASIS identifiers of the variables exchanged between waves and atmosphere and must be identical to those present in the namcouple file. This identifier must contain 8 characters long. Putting an empty space of 8 characters corresponds to the default value: no exchange of this variable. These variables correspond to the name of the OASIS identifier for:

- CWAVE_U10 : the zonal wind speed at 10m sent from the atmosphere to the waves ;
- CWAVE_V10 : the meridian wind speed at 10m sent from the atmosphere to the waves ;
- CWAVE_CHA : the Charnock coefficient sent from the waves to the atmosphere ;
- CWAVE_UCU : the zonal current sent from the wave to the atmosphere ;
- CWAVE_VCU : the meridian current sent from the wave to the atmosphere ;
- CWAVE_HS : the significant wave height sent from the waves to the atmosphere ;
- CWAVE_TP : the peak period sent from the waves to the atmosphere.

Notes: The choice of the parameterization of the turbulent fluxes with consideration of the waves is made with &NAM_SEAFLUXn NGRWAVES and is available only with CSEA_FLUX='COARE3'. Refer to the SurfEx doc for more details.

3.4.4 List of files in Meso-NH/SurfEx input directory

files required for Meso-NH/SurfEx

```
1_input_mnh
  | -- PGD_*.des, PGD_*.nc            ! files from PREP_PGD program
  | -- PRC_*.des, PRC_*.nc            ! files from PREP_REAL_CASE program
  | -- rstrt_SAVE.nc                  ! from python script
  | -- EXSEG1.nam                     ! namelist for the coupled simulation
```

3.5 Preparation of WW3

In this section, we will use the scripts located in the directory **2_input_ww3** from the git repository (Section 3.1). The grid has to be generated in local because the software used is in matlab. Once the grid is created, all the other steps can be done exclusively on beaufix.

3.5.1 Create the grid with gridgen toolbox (matlab)

The creation of the grid with gridgen has to be done in local because we have to use matlab which is not installed on beaufix.

First, you have to download the gridgen software in the account beaufix on tramontane:

```
/home/beaufix/JORIS/tuto_cpl_owa/gridgen.tar
```

To prepare the grid files for WW3 you have to go in the cas_reels directory and modify the path in the beginning of the script create_grid_tuto.m. Then you have to define the name of your config and the size of the dimension and size of the horizontal grid (dx and dy are in degree):

in create_grid_tuto.m

```
fname = 'tuto';
dx = 1.0;
dy = 1.0;
lon1d = [ 30.00:dx: 60.00];
lat1d = [-30.00:dy:  0.00];
```

By default, the bathymetry data file used by gridgen is located in reference_data directory of gridgen and is etopo1.nc. ETOPO1 is a global database, with a horizontal grid spacing of 1 arc minutes (approximately 2 km).

After the modifications of the create_grid_tuto.m script, you can launch matlab, open this script and run it. A series of figure are created at the end (if matlab is well installed) of the script and some files are created in the cas_reels/outputs directory:

- tuto.bot : file for SWAN model, not used here
- tuto.depth_ascii : file with depth at each grid point
- tuto.mask : mask of each grid point
- tuto.maskorig_ascii : mask original at each grid point
- tuto.meta : metadata for ww3_grid.inp namelist
- tuto.obstr_lev1 : file obstruction of each grid point

Only the files present in the tuto.meta file has to be used : tuto.depth_ascii, tuto.mask and tuto.obstr_lev1. Once these files are created, you have to transfer them to beaufix.

3.5.2 Prepare the grid : ww3_grid

Once the files created by gridgen are transfer on beaufix, you need to convert it into files readable by WW3. For that purpose you need to use the ww3_grid program on beaufix. This program uses the namelist ww3_grid.inp (you can find an example in **2_input_ww3**). In this namelist you have to add the information contained in the tuto.meta at the location corresponding to the grid definition, take care about the path of your grid files.

in ww3_grid.inp

```
$
'RECT'  T  'NONE'
1275     750
1.08     1.08   60.00
44.5300         -17.6700         1.00
$
$ Bottom Bathymetry
$
-0.10   2.50  40  0.001000  1  1 '(....)'  NAME  'Fantala_2km.depth_ascii'
$
$ Sub-grid information
$
50  0.010000  1  1  '(....)'  NAME  'Fantala_2km.obstr_lev1'
$
$ Mask Information
60  1  1  '(....)'  NAME  'Fantala_2km.maskorig_ascii'
$
```

Then you can modify the path in the script prep_ww3_grid_beaufix.sh and run it via the following command:

```
sbatch prep_ww3_grid_beaufix.sh
```

At the end of this step, you have to find at the end of ww3_grid.out file "End of program" without errors. This program generated 3 files : mod_def.ww3, mask.ww3 and mapsta.ww3. These files will be used by the other programs, they correspond to the WW3 grid in binary format. Depending of the wind-wave parameterization you use, a file ST4* can also be generated at this step : this file is also necessary for the other program.

3.5.3 Prepare the forcing fields for the spinup simulation : ww3_prnc

Contrary to Meso-NH/SurfEx and CROCO, a spinup simulation is necessary to prepare the initial condition of WW3. To prepare this simulation, you need 10m wind speed fields to force the wave model. To be consistent with the atmospheric fields used to initialize and force at the lateral boundary condition Meso-NH we will use the 10m wind speed from ECMWF servers (operational analyses at 9 km resolution).

To extract the 10m wind speed from ECMWF servers, you can follow the documentation of Emmanuel Cordier (Appendix A) or ask to Soline Bielli (cf. Section 3.4.2). At the end of these steps, you will have a netcdf file containing the 10 m wind speed fields for your period of interest. I recommend to do a spinup simulation of about 7 days so you have to extract the 10 m wind speed for at minimum 7 days before the beginning of your coupled simulation.

If you are using the Emmanuel Cordier documentation, you have to convert the file created by using the bash script:

```
. convert_netcdf_from_ecget_to_ww3_prnc.sh
```

Then, you have to modify the ww3_prnc.inp namelist according to your data and launch the script prep_ww3_prnc_beaufix.sh.

```
sbatch prep_ww3_prnc_beaufix.sh
```

At the end of this step, you will have the file wind.ww3 which is created. This file has to be used in the spinup simulation.

3.5.4 Prepare the lateral boundary conditions : ww3_bound

To force the spinup and coupled simulations at the lateral boundary condition, you need wave energy spectra. For that purpose, you have to ask to Soline Bielli to download the wave energy spectra from ECMWF. Once the wave energy spectra from ECMWF are extracted, you have to convert the grib files into spec files readable by ww3 programs.

First, you have to fill the file list_points with the latitude and longitude of the points you need for your lateral boundary conditions. Depending of your domain size and location, I recommend you to use at minimum a spacing of 1 degree between the points. You don't have to have a point in double and the first number has to be 000000. The longitude and latitude you choose has to be a multiple of the ecmwf grid. No space interpolation is performed at that time and need to be developed.

Once the file list_points is created, you can compile the fortran programs on beaufix (the scripts are in **2_input_ww3** but you can move it in another directory):

```
./compile_codes.sh
```

Two fortran programs are compiled :

- the extract_spectra_for_ww3.f90 program. Thie program is an adaptation of the script given by Jean Bidlot at ECMWF. It convert the grib files containing the spectra into txt files.
- the spectra_from_mfwam_to_ww3_spectra.f program. This program reads the txt files created by the previous program and created spec files readable by ww3_bound program.

After the compilation of the fortran program, you need to create a directory called "spectre" where the files will be created. Then, you need to adapt the date according to your directories in the run_codes.sh and run it (on beaufix).

```
./run_codes.sh
```

It is also possible to use these scripts in local after the installation of ECCODES program[3]. At the end of this program, files are created in the spectre directory (ww3.*.spec) : one files for

[3] https://confluence.ecmwf.int/display/ECC

one lateral boundary condition points with all time steps required in these files.

Once the *.spec files are created you have to modify the namelist ww3_bound.inp and put the path to these files in it. Then you can run the program ww3_bound to prepare the lateral boundary conditions for WW3:

```
sbatch prep_ww3_bound_beaufix.sh
```

At the end ot this step, you have to find “Writing data for time :” at the end of the ww3_bound.out file. This program generated a file called nest.ww3 corresponding to the lateral boundary conditions for WW3. This file will be used in the spinup and coupled simulations.

3.5.5 Prepare the initial condition for the coupled simulation : spinup

To prepare the initial condition for WW3, you need to run a forced WW3 spinup simulation.

For that purpose, you can go into the directory A_ww3_spinup, modify the script run_ww3.sh and the namelist ww3_shel.inp_A_ww3_spinup according to your period of interest. This namelist is in the 1_input_ww3 directory. Simulations of 7 days is enough to prepare your initial conditions. You can run the script run_ww3.sh:

```
sbatch run_ww3.sh
```

At the end of the spinup simulation you have to find “End of program” at the end of ww3_shel.out file. In the directory of your simulation you have to find files called restart*.ww3. If you do a simulation of 7 days, you have to find 7 files. The last file correspond to the date of the beginning of your coupled simulation (number of restart files is defined in the ww3_shel.inp namelist). An other file called out_grd.ww3 is created and correspond to the grid outpur parameters.

To verify the spinup, you need to convert the ww3 output files in netcdf by using the ww3_ounf program and look at the significant wave height. You need to modify the script run_ww3_ounf.sh and ww3_ounf.inp according to your path and configuration then you can run the run_ww3_ounf.sh program:

```
sbatch run_ww3_ounf.sh
```

At the end ot this step a netcdf file is created containing all the grid parameters required. The spinup is done if the boundary conditions information is propagated in all the domain at the end of the simulation.

If spinpup is ok, you have to use the script create_restart_file_from_WW3_file.py (located in **2_input_ww3**) to create the rstrt_SAVE.nc file. This script will read the netcdf file created previously, the fields inside the rstrt_SAVE.nc file has to be the time of the end of the spinup simulation.

```
python create_restart_file_from_WW3_file.py
```

3.5.6 Prepare the namelist for the coulped run : ww3_shel.inp

To finish, you have to prepare the namelist for the coupled run called ww3_shel.inp_C_mnh_ww3_croco_rmp.

part of ww3_shel.inp corresponding to the coupling

```
C F     Water levels
C F     Currents
C F     Winds

$
$ Type 7 : Coupling (must be fully commented if not used)
$          Diagnostic fields to exchange (same format as output fields)
$
20160414 000000   100  20180618 000000
N
$
$   - Sent fields by ww3:
$       - Ocean model : TOM1 OHS DIR BHD TWO UBR FOC TAW LM DRY
$       - Atmospheric model : CUR CHA AHS FP TP FWS
$
TOM1 OHS DIR ACHA AHS CUR FWS
$
$   - Received fields by ww3:
$       - Ocean model : SSH CUR DRY
$       - Atmospheric model : WND
$
SSH CUR WND
$
```

3.5.7 List of files in WW3 input directory

files required for WW3

```
2_input_ww3
  | -- mapsta.ww3             ! from ww3_grid program
  | -- mask.ww3               ! from ww3_grid program
  | -- mod_def.ww3            ! from ww3_grid program
  | -- ST4ABUHF2.bin          ! from ww3_grid program
  | -- nest.ww3               ! from ww3_bounc program
  | -- restart.ww3            ! from spinup simulation
  | -- rstrt_SAVE.nc          ! from python script
  | -- ww3_shel.inp           ! namelist for the coupled simulation
```

3.6 Preparation of CROCO

In this section, we will use the scripts located in the directory 3_input_croco from the git repository (Section 3.1). All the following steps has to be done in local because matlab is not installed on beaufix.

To prepare the configuration of CROCO, it is necessary to

- download MERCATOR data files over a longer period than the desired simulation, for example extract 2-3 days before and 2-3 days after the period of interest. If you decide to change the simulation period you will not have to download the MERCATOR data again.

- convert the extracted file into a file readable by the croco_tools: choose as a first step, the day before the one you will choose as the initial time (crocotoolparam.m)
- create grid, initial and condition files at the lateral boundaries for CROCO with croco_tools

3.6.1 Download MERCATOR fields : motu_client

To initialize and force CROCO at the lateral boundary conditions, we use the MERCATOR data from COPERNICUS servers. To extract MERCATOR data you need to create an account on http://marine.copernicus.eu/services-portfolio/access-to-products/. After your account successfully created you have to install motu_client.py, modify the script extract_mercator_data.sh according to your needs and launch the script:

```
. extract_mercator_data.sh
```

At the end of this extraction, you will have a netcdf file with the corresponding fields.

3.6.2 Convert MERCATOR fields for croco toolbox

To use the file extracted from MERCATOR database with croco_tools, you need to convert this file and to calculate some variables.

First, you need to compile the fortran routine called integrate.f90 with :

```
sudo f2py -c integrate.f90 -m integrate
```

Then you have to modify the path and name of the mercator data file extracted and choose the index of the time required for the initial conditions in the script convert_mercator*.py (choose the date before the date requested). After these modifications, you can launch the script python:

```
python convert_mercator_netcdf_file_to_croco_tools_netcdf_file.py
```

At the end of this step, a new netcdf file called output.nc is created, this file will be read by croco_tools.

3.6.3 Using matlab toolbox : make_grid, make_OGCM_frcst

First, you need to download the croco_tools from the croco website but I recommend that you use the version available on the account beaufix on tramontane server:

```
/home/beaufix/JORIS/tuto_cpl_owa/croco_tools.tar
```

To create your configuration you need to modify the path in the start.m file and modify some parameters according to your configuration in crocotools_param.m file located in the **2_input_croco** directory.

parameters to modify in the crocotools_param.m file

```
lonmin =  44.53 ; % Minimum longitude [degree east]
lonmax =  67.46 ; % Maximum longitude [degree east]
latmin = -17.66 ; % Minimum latitude  [degree north]
latmax = -4.18  ; % Maximum latitude  [degree north]
%
```

```
% Grid resolution [degree]
%
dl = 0.0190;
%
%  CROCOTOOLS directory
%
CROCOTOOLS_dir = '/home/jorisp/02_MODELS/croco_tools/';
%
%  Global data directory (etopo, coads, datasets download from ftp, etc..)
%
DATADIR = '/home/jorisp/03_WORK/3_DATA_CROCO/';
Ymin          = 2016;          % first forcing year
Ymax          = 2016;          % last  forcing year
Mmin          = 4;             % first forcing month
Mmax          = 4;             % last  forcing month
%
Dmin          = 14;            % Day of initialization
Hmin          = 0;             % Hour of initialization
Min_min       = 0;             % Minute of initialization
Smin          = 0;             % Second of initialization
```

You also have to modify the script croco_tools/Forecast_tools/make_OGCM_frcst.m according to the date of the netcdf file (rundate_str).

parameters to modify in the make_OGCM_frcst.m file

```
rundate_str='21-Mar-2019'
OGCM_name=[FRCST_dir,'output_20190321_20190402.nc'];
```

Then you have to launch matlab and go into the directory of the crocotools_param script. To create the grid, the initial and lateral boundary conditions you have to execute the following commands (under matlab environment):

```
start
make_grid
make_OGCM_frcst
```

At the end of this step, 3 files are created : (i) croco_grd.nc corresponding to the grid, (ii) croco_ini.nc corresponding to the initial condition and (iii) croco_bry.nc corresponding to the boundary conditions.

Note: tstart must be the same in the croco_ini.nc and croco_bry.nc files and must correspond to the day the simulation starts minus one.

Once the times are verified, you have to transfer these files on beaufix and go into the 3_input_croco to execute the create_restart_file_from_ini_CROCO_file.py script to create the rstrt_SAVE.nc file for CROCO. Modification of the path into the python script is necessary.

```
python create_restart_file_from_ini_CROCO_file.py
```

3.6.4 Prepare the namelist for the coupled simulation

No information about OASIS has to be defined in croco.in namelist (which is in the directory **3_input_croco**) : CROCO read the namcouple file to access to the corresponding information. To modify this namelist, please refer to the CROCO documentation.

3.6.5 List of files in CROCO input directory

files required for CROCO

```
3_input_croco
  | -- croco_grd.nc
  | -- croco_ini.nc
  | -- croco_bry.nc
  | -- rstrt_SAVE.nc
  | -- croco.in
```

Chapter 4

Simulation

4.1 Launch a coupled simulation to test and create the rmp files

First, you have to launch a coupled simulation to create the rmp files and to verify that the coupling is working. This simulation has to be short in time (few exchange). The creation of rmp files can take a few hours and can only be created once, the idea is to use them for the final simulations.

Depending on what you want to do: an OA coupled simulation or an OWA coupled simulation you have to go to in the B_mnh_croco_rmp directory or the C_mnh_ww3_croco_rmp, respectively. In these directory you have to change the path and launch the script:

```
sbatch script_run*.sh
```

At the end of this simulation you have to read "SUCCESSFUL RUN" at the end of the debug.root.XX files. The standard outputs for Meso-NH/SurfEx, WW3 and CROCO are the same as in forced simulations. Few files called rmp*.nc are created, these files corresponds to the interpolation weight used by OASIS to interpolate grid to grid. These files have to be used during the coupled simulation.

4.2 Launch a coupled simulation

This step is not in the git repository and has to be adapt according to how you want to work. To launch the "real" coupled simulation, I recommend you to create a new folder called B_mnh_croco_date for example, in which you will copy the previously created rmp* to avoid having to create them at each simulation.

The different namelists and the job script must therefore be adapted according to what you want.

4.3 Outputs of a coupled simulation

At the end of your simulation, a lot of files is created:

- output from Meso-NH /SurfEx : standard output (the same as a forced simulation)
- output from WW3 : standard output (the same as a forced simulation)

- output from CROCO : standard output (the same as a forced simulation)
- output from OASIS : debug files, ...

Appendix A

Protocol to extract ECMWF 10m wind speed for WW3 spinup run

OSU-RÉUNION

Rapatriement des données ECMWF

Fiche résumée

ECMWF

Emmanuel Cordier ecordier@univ-reunion.fr

1. **Connexion machine virtuelle (VM)**

   - sous Linux ou Os X

   ```
   >> ssh -X getecmwf@get-ecmwf.univ.run
   password: getecdata2012
   ```

   - sous Windows

   ```
   utiliser le logiciel putty (https://www.putty.org/)
   ```

   - répertoire d'accueil de la VM

   ```
   get-ecmwf: ~
   getecmwf $> l
   total 68K
   drwxr-xr-x 2 getecmwf getecmwf  52K mars  8 06:53 downloadecmwf/
   drwxr-xr-x 2 getecmwf getecmwf 4,0K mars 13 11:14 privatetools/
   drwxr-xr-x 4 getecmwf getecmwf 4,0K mars  8 09:33 python_scripts_NE_PAS_MODIFIER/
   drwxr-xr-x 4 getecmwf getecmwf 4,0K mars  8 12:15 users_scripts/
   ```

2. **Création d'un dossier utilisateur** (à effectuer à la première connexion et/ou si le dossier n'existe pas)
   nom du dossier : première lettre du prénom suivie du nom de famille

   ```
   # eg. create the directory pdurand for Patrick Durand
   >> mkdir users_scripts/pdurand
   ```

3. **Choix du script de téléchargement**
   Les scripts de téléchargement sont écrits en Python.
   Ils se trouvent dans le répertoire python_scripts_NE_PAS_MODIFIER, organisé en deux sous-répertoires pour les données erainterim ou operationaldata.
   Il existe 2 cagtégories de scripts :
   ecget_cc_tt_pp.py pour télécharger des données d'une longueur maximale d'un mois.
   ecget_cc_tt_pp_multiyears.py pour télécharger des données sur plusieurs mois ou années.

   Les noms des scripts suivent une nomenclature simple ***ecget_cc_tt_pp.py*** où [a] ecget : partie fixe du fichier qui ne change pas, [b] cc : classe de donnée ei (erainterim) ou od (operationaldata), [c] tt : type de donnée an (analysis) ou fc (forecast), [d] pp : paramètre de donnée ml (model levels), pl (pressure levels) ou sfc (surface).
   Ex. :
   ecget_ei_an_pl_multiyears.py *permet de télécharger les analyses (an) erainterim (ei) en niveau de pression (pl) sur plusieurs années.*
   ecget_od_fc_sfc.py *permet de télécharger les forecasts (fc) des données opérationnelles (od) au niveau de la surface (sfc) pour de courtes périodes.*

!!!
NE PAS MODIFIER CES SCRIPTS
LES COPIER DANS SON REPERTOIRE UTILISATEUR CREE EN ETAPE 2
!!!

4. **Copie du script de téléchargement**

```
# copy script to download operational data (od) forecasts (fc) on the surface (sfc) for short periods
>> cp python_scripts_NE_PAS_MODIFIER/operationaldata/ecget_od_fc_sfc.py users_scripts/pdurand/.
```

5. **Configuration de la requête**
L'utilisateur peut éditer et modifier le scripts copié dans son répertoire personnel.

```
# edit script with vi
>> vi users_scripts/pdurand/ecget_od_fc_sfc.py
OU
# edit script with gedit (requires the -X option in ssh command; see step1)
>> gedit users_scripts/pdurand/ecget_od_fc_sfc.py
```

Les paramètres du script que l'utilisateur peut adapter à ses besoins sont commentés et expliqués. Il s'agit principalement [a] des dates, ou de l'emprise temporelle (dans le cas multiyears), [b] des heures d'analyses, [c] des heures de forecasts (dans le cas fc), [d] des niveaux de pression (dans le cas pl) ou des niveaux de modèles (dans le cas ml) [e] des paramètres, [f] de la résolution de la grille, [g] de l'emprise géographique.

6. **Exécution de la requête, téléchargement des données sur la machine virtuelle**

```
# execute python script to download data
>> cd users_scripts/pdurand
>> python ecget_od_fc_sfc.py pdurand mydatafilename
```

Cette commande permet :

[a] de créer le dossier pdurand_gribdata dans le répertoire downloadecmwf/

[b] de télécharger les données depuis le centre européen au format GRIB dans le répertoire downloadecmwf/pdurand_gribdata

[c] de nommer les fichiers de données mydatafilename.grb

La requête envoyée au centre européen est mise en file d'attente sur leurs serveurs. Selon le nombre de requêtes soumises au centre européen, le téléchargement peut ne pas démarrer dans l'immédiat. Pour contrôler l'activité ou le statut de votre requête, se rendre sur la page : `http://apps.ecmwf.int/webapi-activity/`. **La requête sera identifiée sous le nom d'utilisateur auj**.

7. **Transformation des données GRIB au format NETCDF**

```
# transformation d'un seul fichier grib en netcdf
>> cd downloadecmwf/pdurand_gribdata
>> grib2netcdf -D NC_FLOAT -o mydatafilename.nc mydatafilename.grb

# transformation d'un lot de fichiers grib en netcdf
>> cd downloadecmwf/
>> grib2netcdf_multifiles pdurand_gribdata # le dossier contenant les fichiers grib
```

8. **Récupération des données localement**
Sous Linux, OS X ou Windows, utiliser le client FTP/SSH FILEZILLA `https://filezilla-project.org/`.

Bibliography

Pianezze, J. et al. (2018): A New Coupled Ocean-Waves-Atmosphere Model Designed for Tropical Storm Studies: Example of Tropical Cyclone Bejisa (2013-2014) in the South-West Indian Ocean. Journal of Advances in Modeling Earth Systems, 10, Nr. 3, 801–825 ⟨URL: http://doi.wiley.com/10.1002/2017MS001177⟩ – visited on 2018-08-06, ISSN 19422466

Voldoire, Aurore et al. (2017): SURFEX v8.0 interface with OASIS3-MCT to couple atmosphere with hydrology, ocean, waves and sea-ice models, from coastal to global scales. Geoscientific Model Development, 10, Nr. 11, 4207–4227 ⟨URL: https://www.geosci-model-dev.net/10/4207/2017/⟩ – visited on 2018-08-06, ISSN 1991–9603