Tutorial 01 CROCO:
Creating your working environment

1. Purpose

In this tutorial we will review the basic instructions to use CROCO in the National Laboratory for High Performance Computing (NLHPC) located in Santiago, Chile. You will learn to configure and submit a simple simulation of the Benguela domain, and you will also learn some basic instructions for the use of the NLHPC cluster.

2. Login in to the NLPHC

2.1. From a terminal environment

To connecto to the NLHPC you have to write, from a terminal, the following command

```
ssh -XC student11@leftraru.nlhpc.cl
```

Replace *student11* with your own student account number. The password will be sent by Zoom.

Upon entry you will see in your terminal something like this

```
Laboratorio Nacional de Computacion de Alto Rendimiento (NLHPC)
Centro de Modelamiento Matematico (CMM)
Universidad de Chile

IMPORTANTE: NO EJECUTAR PROCESOS EN ESTE NODO POR T > 30 min
PARA ESO DEBEN DE USARSE LAS COLAS DE EJECUCION
.................................................................

********************************************************************************
EN CASO DE TENER DUDAS CON SU SCRIPT, LO INVITAMOS A USAR NUESTRO GENERADOR
AUTOMÁTICO EN EL SIGUIENTE LINK: https://wiki.nlhpc.cl/Generador_Scripts
********************************************************************************

********************************************************************************
ESTIMADO USUARIO, A CONTINUACIÓN SE LISTAN LOS NODOS LIBRES PARA SER UTILIZADOS:
********************************************************************************

PARTICION  NODO   ESTADO
slims*     41     idle
debug      4      idle
general    10     idle
largemem   1      idle
student11@leftraru3:~$
```

3. Obtaining the CROCO code

First we have to obtain a copy of the CROCO code in our directory. In this case, to copy the code that will be used in days 01 and 02 of the CROCO advanced course write

```
mkdir andres
cp -r /home/lmod/software/MPI/intel/2018.5.274-GCC-8.2.0-2.31.1/impi/2018.4.274/croco/1.2beta/croco ./andres
```

Don't forget to replace **andres** with your own name in this and all the following instructions.

4. Obtaining the CROCO_TOOLS code

To copy the code of CROCO_TOOLS write

```
cp -r /home/dbs/croco/croco_tools-master/* ./andres/croco/croco_tools
```

5. Creating the BENGUELA_LR working directory

A simple example of the use of CROCO is the so called BENGUELA_LR configuration, which corresponds to a domain of the Benguela upwelling region, in low resolution (hence the LR : *Low Resolution*). This configuration is the default configuration in CROCO and what we will do is similar to what is described in Penven et al. (2001).

The first step is to edit the file **create_run.bash** with the following instructions

```
cd andres/croco
nano create_run.bash
```

Now you have to modify this section to add the correct path

```
#==========================================================================================
# BEGIN USER MODIFICATIONS
#
# Get CROCO directory
CROCO_DIR="/home/courses/student11/andres/croco"
#
SOURCES_DIR="/home/courses/student11/andres/croco"
#
TOOLS_DIR="/home/courses/student11/andres/croco/croco_tools"
#
MY_CONFIG_PATH=${SOURCES_DIR}
#
# Name of the configuration directory defined by the user
#
MY_CONFIG_NAME='BENGUELA_LR'
#
#
# END USER MODIFICATIONS
#==========================================================================================
```

now type

```
./create_run.bash
```

which will give you

```
student11@leftraru3:~/andres/croco$ ./create_run.bash

Your choices :
 - SOURCES_DIR  : /home/courses/student11/andres/croco
 - TOOLS_DIR    : /home/courses/student11/andres/croco/croco_tools
 - CONFIG_DIR   : /home/courses/student11/andres/croco
 - CONFIG_NAME  : BENGUELA_LR
 Do you want to proceed ? [Y/n]
```

press **Y** and you will see

```
 Creating configuration ...

=> Copy the source files from /home/courses/student11/andres/croco
   needed to setup your own simulations

=> Copy from /home/courses/student11/andres/croco  done

=> Copy the tools from and  /home/courses/student11/andres/croco/croco_tools
   needed to setup your own simulations

=> Copy from /home/courses/student11/andres/croco/croco_tools  done
/home/courses/student11/andres/croco
student11@leftraru3:~/andres/croco$
```

This script creates a directory with the name that you defined in **CONFIG_NAME**. This directory contains all the necessary codes to run your simulation, this will be your working directory. The content of this folder should be similar to this:

```
student11@leftraru3:~/andres/croco$ ls BENGUELA_LR/
cppdefs.h            domain_def.xml         namelist_pisces_ref      SCRATCH
create_run.bash.BCK  field_def.xml_full     namelist_pisces_ref.1    sediment.in
CROCO_FILES          iodef.xml              oct_start.m              start.m
croco.in             jobcomp                param.h                  TEST_CASES
croco.in.1           Misc                   README_XIOS              xios_launch.file
croco_inter.in       NAMELIST_OANALYSIS     run_croco.bash
crocotools_param.m   namelist_pisces_cfg    run_croco_forecast.bash
DATA                 namelist_pisces_cfg.1  run_croco_inter.bash
```

6. Compiling CROCO

First compile the CROCO executable file with the following instructions.
Modify **cppdefs.h** to activate the **MPI** paralelization flag

```
cd BENGUELA_LR
nano cppdefs.h
```

were we change

```
# undef MPI
```

to

```
# define MPI
```

Now we compile the code

```
ml purge
ml intel/2019b
ml netCDF-Fortran/4.4.4
./jobcomp
```

and the compilation starts with the following information in the screen

```
student11@leftraru3:~/andres/croco/BENGUELA_LR$ ./jobcomp
OPERATING SYSTEM IS: Linux
cp: omitting directory ‘/home/courses/student11/andres/croco/OCEAN/../PISCES/SED’
  file namelist_pisces exists in Run directory
Checking COMPILEAGRIF...
Checking COMPILEMPI...
Checking COMPILEXIOS...
Checking COMPILEOASIS...
Checking COMPILEOMP...
cpp -traditional -DLinux -DIfort -P -I/home/lmod/software/MPI/intel/
2018.5.274-GCC-8.2.0-2.31.1/impi/2018.4.274/netCDF-Fortran/4.4.4/
include -ICROCOFILES/AGRIF_INC mpc.F > mpc_.f
mpiifort -O3 -72 -fno-alias -i4 -r8 -fp-model precise
-mcmodel=medium -axCORE-AVX512,AVX,SSE4.2  -o mpc mpc_.f
```

Now we have to wait a couple of minutes while the **mpiifort** compiler generates the executable file. If everything works fine you will see the following lines in your screen

```
re/MPI/intel/2018.5.274-GCC-8.2.0-2.31.1/impi/2018.4.274/imkl/2018.4
.274/lib -L/home/lmod/software/MPI/intel/2018.5.274-GCC-8.2.0-2.31.1
/impi/2018.4.274/imkl/2018.4.274/mkl/lib/intel64
-L/home/lmod/software/MPI/intel/2018.5.274-GCC-8.2.0-2.31.1/impi/
2018.4.274/imkl/2018.4.274/lib
-L/home/lmod/software/MPI/intel/2018.5.274-GCC-8.2.0-2.31.1/impi/
2018.4.274/netCDF/4.6.2/lib64 -lnetcdf -lnetcdf -liomp5 -lpthread
mv a.out croco
student11@leftraru3:~/andres/croco/BENGUELA_LR$
```

In the last line you can see how the compiled file, whose name by default is **a.out** is renamed as **croco**. This is the executable file we will use.

7. Creating input files

To create the input files that will be read by **croco** we will use a series of functions that are named **CROCO_TOOLS**. This functions are written for Matlab (Penven et al., 2008) and many of them have been adapted to work with Octave and the OCTCDF package.

7.1. The CROCO_TOOLS code

To be able to use the CROCO_TOOLS functions, edit the file **start.m**

```
nano start.m
```

and modify the following line

```
tools_path='/home/courses/student11/andres/croco/croco_tools/';
```

Copy this file to your local directory

```
cp start.m /home/courses/student11/andres/croco/croco_tools/
```

so you don’t need to repeat this step every time you use *create_run.bash* to create a new domain.

7.2. Using Matlab

To create the input files with Matlab, the instructions, from the **BENGUELA_LR** working directory are

```
ml purge
ml Matlab/2017

LD_PRELOAD=/home/lmod/software/Core/ifort/2019.2.187-GCC-8.2.0-2.31.1/
compilers_and_libraries_2019.6.324/linux/compiler/lib/intel64/libirc.so

matlab -nodesktop -nosplash
```

Once in Matlab define the search paths for CROCO_TOOLS

```
start
```

which gives you

```
>> start
Add the paths of the different toolboxes
Arch : x86_64 - Matlab version : 2017a
Use of mexnc and loaddap in 64 bits.
```

Now write the command that generates the model grid, which will be saved in the **croco_grd.nc** file, in the **CROCO_FILES** directory

```
>> make_grid
mkdir: cannot create directory ‘/home/courses/student11/andres/croco/BENGUELA_LR/CROCO_FILES/’:
File exists

 Making the grid: /home/courses/student11/andres/croco/BENGUELA_LR/CROCO_FILES/croco_grd.nc

 Title: Benguela Model

 Resolution: 1/3 deg

 Do you want to use interactive grid maker ?
 (e.g., for grid rotation or parameter adjustments) : y,[n]
```

press **n** and then you get

```
 Create the grid file...
 LLm = 41
 MMm = 42

 Fill the grid file...

 Compute the metrics...

 Min dx=29.1913 km - Max dx=33.3244 km
 Min dy=29.2434 km - Max dy=33.1967 km

 Fill the grid file...

 Add topography...
   CROCO resolution : 31.3 km
   Topography data resolution : 3.42 km
   Topography resolution halved 4 times
   New topography resolution : 54.6 km
Processing coastline_l.mat ...

 Do you want to use editmask ? y,[n]

```

here we will see Fig. 1, and we choose again **n**

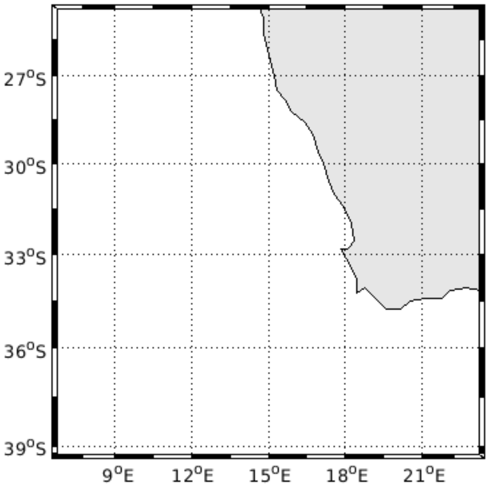


Figura 1: Approximate map for the Benguela domain

finally we will see the following messages

```
 Filter topography ...
 Apply a filter on the Deep Ocean to reduce isolated seamounts :
   4 pass of a selective filter.
 Apply a selective filter on log(h) to reduce grad(h)/h :
   20 iterations - r_max = 0.27931
   29 iterations - r_max = 0.24975
 Smooth the topography a last time to prevent 2DX noise:
   2 pass of a hanning smoother.

 Write it down...
```

The next instruction is **make_forcing** which created the file **croco_frc.nc** with all the atmospheric forcing information.

```
>> make_forcing
mkdir: cannot create directory ‘/home/courses/student11/andres/croco/BENGUELA_LR/CROCO_FILES/’:
File exists

Benguela Model

 Read in the grid...

 Create the forcing file...
Getting taux for time index 1
Getting tauy for time index 1
Getting taux for time index 2
Getting tauy for time index 2
Getting taux for time index 3
Getting tauy for time index 3
Getting taux for time index 4
....
Getting shortrad for time index 7
Getting shortrad for time index 8
Getting shortrad for time index 9
Getting shortrad for time index 10
Getting shortrad for time index 11
Getting shortrad for time index 12
>>
```

And finally type **make_clim** which creates the file **croco_clm.nc** with the oceanic boundary condition

```
make_clim
>>mkdir: cannot create directory ‘/home/courses/student11/andres/croco/BENGUELA_LR/CROCO_FILES/’:
File exists

 Making the clim: /home/courses/student11/andres/croco/BENGUELA_LR/CROCO_FILES/croco_clm.nc

 Title: Benguela Model

 Read in the grid...

 Create the climatology file...

 Creating the file : /home/courses/student11/andres/croco/BENGUELA_LR/CROCO_FILES/croco_clm.nc

 VTRANSFORM = 2
...
PSI: 72 iterations
Flux correction : 5.2674e-15
========================
Initial

 Creating the file : /home/courses/student11/andres/croco/BENGUELA_LR/CROCO_FILES/croco_ini.nc
 VTRANSFORM = 2

 Temperature...
 Time index: 1

 Salinity...
 Time index: 1

 Compute potential temperature from in-situ...
   getpot: Time index: 1 of total: 1
>>
```

Notice that his command will also create the file with the initial conditions **croco_ini.nc**, which can be also created using the command **make_ini**

It is important to use **make_forcing** before **make_clim**. These commands will create the input files for CROCO, and they will be in the NetCDF format. These files will be placed inside the **CROCO_FILES** directory and they are

```
croco_clm.nc croco_frc.nc croco_grd.nc croco_ini.nc croco_oa.nc
```

The files that you will obtain should be the same as those found in

```
http://mosa.dgeo.udec.cl/CROCO2022/AdvancedCourse/Tutorial01/InitialFiles/
```

if you had problems in this setp, copy those files to the **CROCO_FILES** directory to advance to the next section using the commands

```
cd CROCO_FILES
wget http://mosa.dgeo.udec.cl/CROCO2022/AdvancedCourse/Tutorial01/InitialFiles/croco_grd.nc
wget http://mosa.dgeo.udec.cl/CROCO2022/AdvancedCourse/Tutorial01/InitialFiles/croco_frc.nc
wget http://mosa.dgeo.udec.cl/CROCO2022/AdvancedCourse/Tutorial01/InitialFiles/croco_clm.nc
wget http://mosa.dgeo.udec.cl/CROCO2022/AdvancedCourse/Tutorial01/InitialFiles/croco_ini.nc
```

8. Launching the simulation

To launch the simulation you need the file **run_nlhpc.bash** which you can get with

```
wget http://mosa.dgeo.udec.cl/CROCO2022/AdvancedCourse/Tutorial01/run_nlhpc.bash
```

Now type

```
sbatch run_nlhpc.bash
```

You will get the ID number for this run. we can check the status of the simulation with

```
squeue
```

8.1. Stopping a simulation

If you need to stop early a simulation, use the command

```
scancel  21125780
```

where you should replace **21125780** with the ID number of your process.

9. Output files

Once the simulation end successfully, you with file, in the **CROCO_FILES** directory, the following output files

```
croco_avg.nc
croco_his.nc
croco_rst.nc
```

Those files should be identical to those found in

```
http://mosa.dgeo.udec.cl/CROCO2022/AdvancedCourse/Tutorial01/Results/
```

If you had problems with the previous steps, copy those files to the **CROCO_FILES** directory to advance to the next section

```
cd CROCO_FILES
wget http://mosa.dgeo.udec.cl/CROCO2022/AdvancedCourse/Tutorial01/Results/croco_avg.nc
wget http://mosa.dgeo.udec.cl/CROCO2022/AdvancedCourse/Tutorial01/Results/croco_his.nc
wget http://mosa.dgeo.udec.cl/CROCO2022/AdvancedCourse/Tutorial01/Results/croco_rst.nc
```

10. Visualización of results

10.1. ncdump

The **ncdump** command is very useful to look at the content of a NetCDF file. Remember that both input and output files in CROCO are in NetCDF.

The commands

```
ml purge
ml netCDF-Fortran/4.4.4
ncdump -h CROCO_FILES/croco_avg.nc  | less
```

will show is the structure of the **croco_avg.nc** file

```
netcdf croco_avg {
dimensions:
        xi_rho = 43 ;
        xi_u = 42 ;
        eta_rho = 44 ;
        eta_v = 43 ;
        s_rho = 32 ;
        s_w = 33 ;
        time = UNLIMITED ; // (10 currently)
        auxil = 4 ;
variables:
        char spherical ;
                spherical:long_name = "grid type logical switch" ;
                spherical:option_T = "spherical" ;
                spherical:option_F = "cartesian" ;
        float xl ;
                xl:long_name = "domain length in the XI-direction" ;
                xl:units = "meter" ;
```

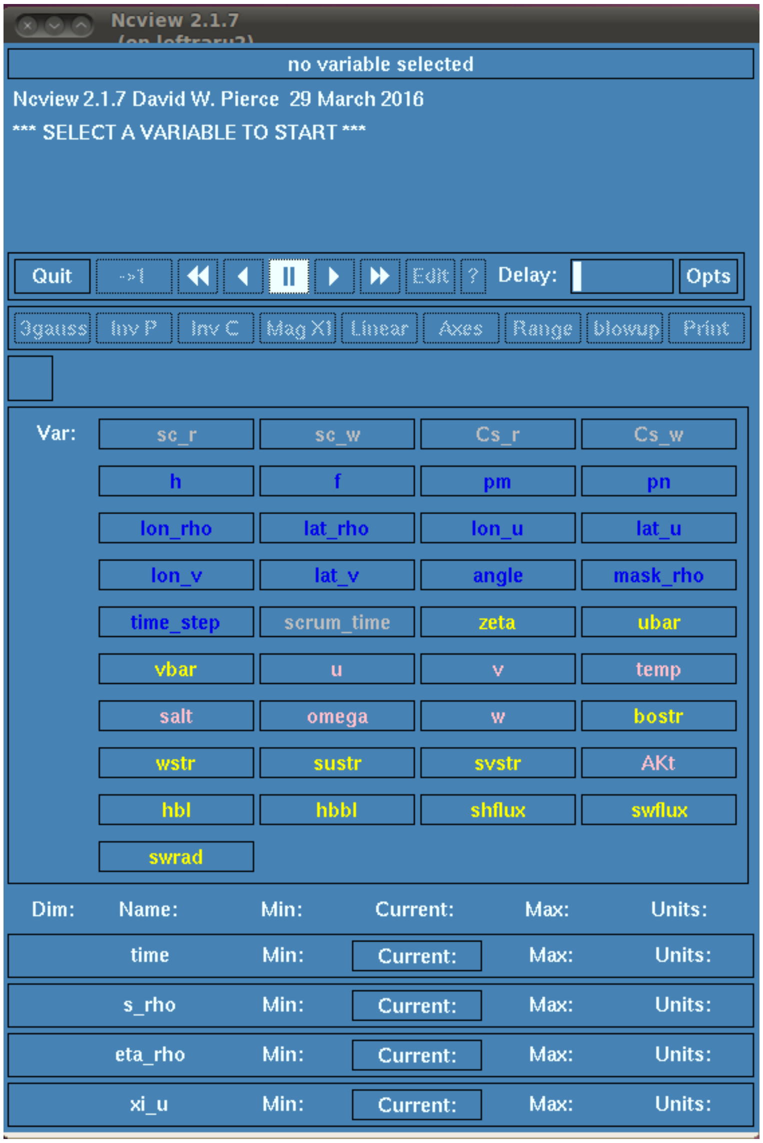
This way we can see details such as the size of our model domain and the number of steps stored. We can use this information with our estimations of the number of steps that should have been saved. To exit **ncdump** press **q**.

10.2. ncview

ncview is a very useful program to take a quick look of a NetCDF file. To use it type

```
ml purge
ml icc/2019.2.187-GCC-8.2.0-2.31.1 impi/2019.2.185 ncview/2.1.7
ncview CROCO_FILES/croco_avg.nc
```

this will show is the following graphical interface



Ncview 2.1.7

no variable selected

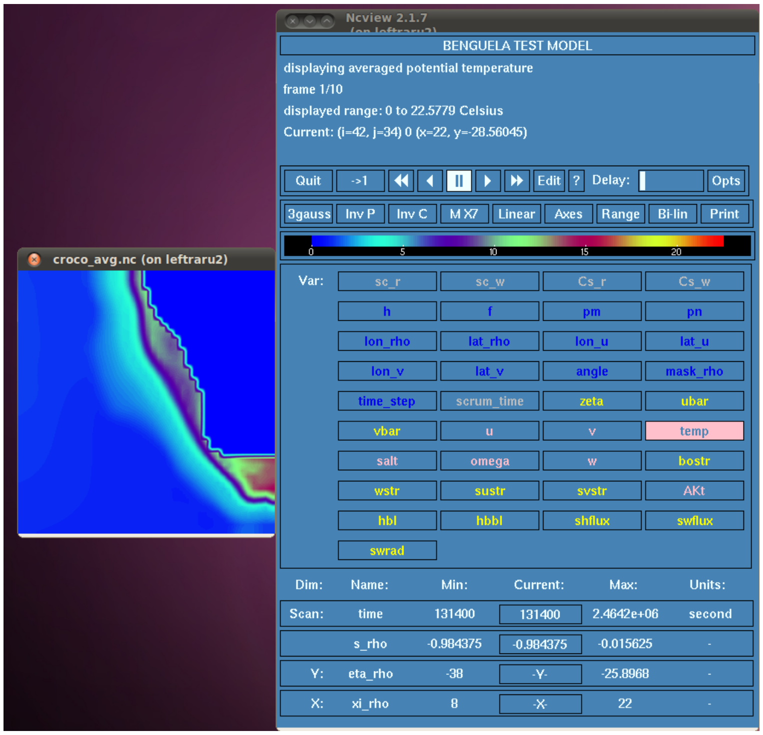
Ncview 2.1.7 David W. Pierce 29 March 2016

*** SELECT A VARIABLE TO START ***

Dim:	Name:	Min:	Current:	Max:	Units:
	time	Min:	Current:	Max:	Units:
	s_rho	Min:	Current:	Max:	Units:
	eta_rho	Min:	Current:	Max:	Units:
	xi_u	Min:	Current:	Max:	Units:

Figura 2: GUI for ncview

press in the **temp** variable to get



Ncview 2.1.7 (on leftraru2)

BENGUELA TEST MODEL

displaying averaged potential temperature

frame 1/10

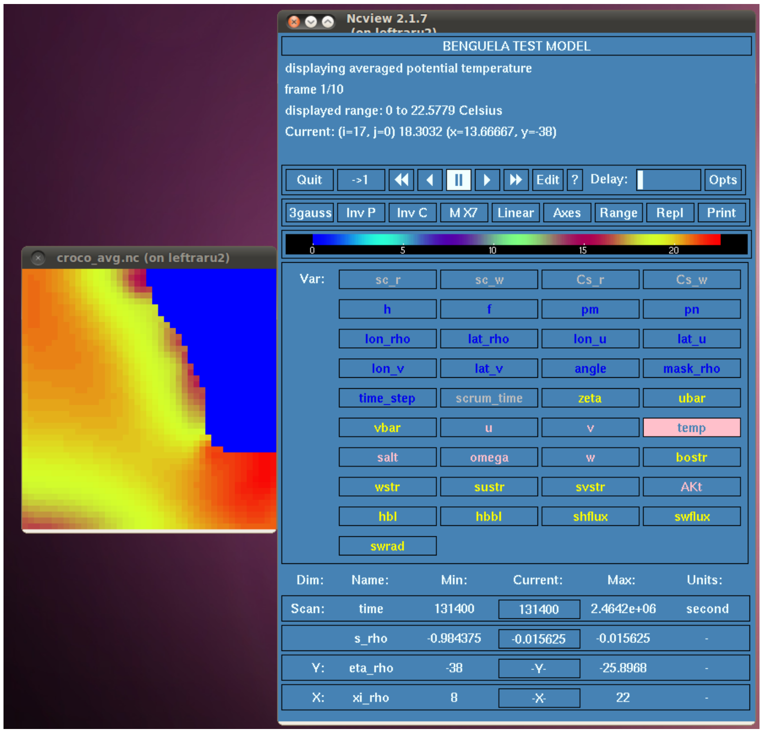
displayed range: 0 to 22.5779 Celsius

Current: (i=42, j=34) 0 (x=22, y=-28.56045)

Dim:	Name:	Min:	Current:	Max:	Units:
Scan:	time	131400	131400	2.4642e+06	second
	s_rho	-0.984375	-0.984375	-0.015625	-
Y:	eta_rho	-38	-Y-	-25.8968	-
X:	xi_rho	8	-X-	22	-

Figura 3: Sea temperature

This figure show us the sea temperature in the bottom layer of the model. If we want to look at the sea surface temperature, press the right button of our mouse over the box that ha the value **-0.984375** which is the **s_rho** vertical level that is being shown. When you do this the value of this box changes to **-0.015625**. It is also convenient to press the **Bi-lin** button, as ncview interpolated, by default the values that are being shown. Once you do this, you will get



Ncview 2.1.7 (on leftraru2)

BENGUELA TEST MODEL

displaying averaged potential temperature

frame 1/10

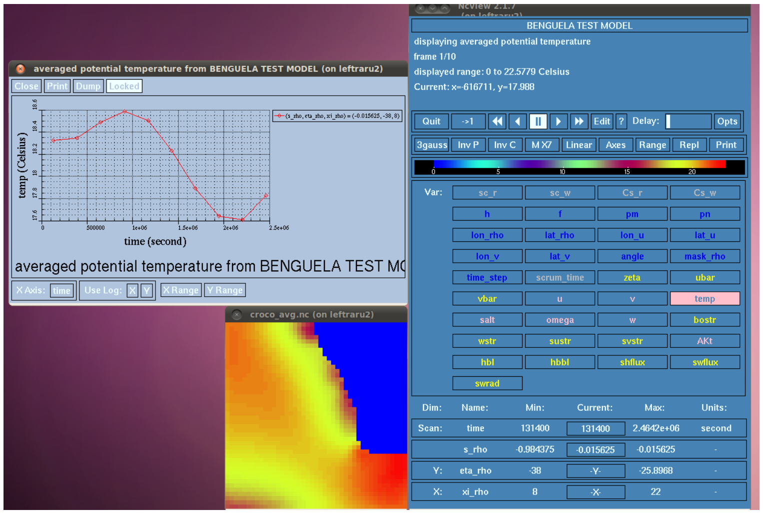
displayed range: 0 to 22.5779 Celsius

Current: (i=17, j=0) 18.3032 (x=13.66667, y=-38)

Dim:	Name:	Min:	Current:	Max:	Units:
Scan:	time	131400	131400	2.4642e+06	second
	s_rho	-0.984375	-0.015625	-0.015625	-
Y:	eta_rho	-38	-Y-	-25.8968	-
X:	xi_rho	8	-X-	22	-

Figura 4: Sea surface temperature

If we click with the mouse over any point of the ocean, ncview will show a time series, at the current sigma level, of that variable.



averaged potential temperature from BENGUELA TEST MODEL (on leftraru2)

averaged potential temperature from BENGUELA TEST MO

Ncview 2.1.7 (on leftraru2)

BENGUELA TEST MODEL

displaying averaged potential temperature

frame 1/10

displayed range: 0 to 22.5779 Celsius

Current: x=-616711, y=17.988

croco_avg.nc (on leftraru2)

Dim:	Name:	Min:	Current:	Max:	Units:
Scan:	time	131400	131400	2.4642e+06	second
	s_rho	-0.984375	-0.015625	-0.015625	-
Y:	eta_rho	-38	-Y-	-25.8968	-
X:	xi_rho	8	-X-	22	-

Figura 5: Time series

As was said before, **ncview** is a very fast and simple tool that can be used to quickly visualize the results that were obtained. It helps to check if what was obtained is reasonable.

10.3. CROCO_TOOLS

CROCO_TOOLS contains many useful functions that can be used to visualize our model results. These functions can be used to create vertical sections, surface maps, vertical profiles, time series, etc.

11. Advanced details

The key aspects of a **CROCO** simulation depend on the files

```
crocotools_param.m
cppdefs.h
param.h
croco.in
```

In this case everything should work ok as these files are configured for the default domain,**BENGUELA_LR**. In another tutorial you will find information on how to modify these files to study an area of your interest.

12. Conclusion

In this tutorial you learned how to access the NLHPC servers, to copy the files to run the **CROCO** model and to prepare the input files using the **CROCO_TOOLS** toolbox. You also launched a basic simulation for the Benguela domain using basic instructions of a cluster environment, and you also visualized the results with **ncview**.

For further information:
Andrés Sepúlveda (asepulveda@dgeo.udec.cl)
Contributions from:
Nicolás Aubone
Marcela Contreras
Gabriel Montecinos
Mauro Santiago

13. References

Penven, P., Roy, C., Brundrit, G. B., De Verdière, A. C., Fréon, P., Johnson, A. S., Lutjeharms J. R. E. & Shillington, F. A. (2001). A regional hydrodynamic model of upwelling in the Southern Benguela. South African Journal of Science, 97(11-12), 472-475.

Penven, P., Marchesiello, P., Debreu, L., & Lefèvre, J. (2008). Software tools for pre-and post-processing of oceanic regional simulations. Environmental Modelling & Software, 23(5), 660-662.

14. Useful links

14.1. The CROCO model

http://www.croco-ocean.org

14.2. The CROCO users forum

https://forum.croco-ocean.org/questions/