

TUTORIAL 01:

CREATE THE WORKING ENVIRONEMENT

In this tutorial, we will review some basic instructions to **set up the working environment**. We will work on the **CHPC cluster** called LENGAU. In this tutorial, we will detail the steps to log onto the super-computer. Then, we will copy some CROCO directories that are already stored on the super-computer (instead of downloading some archives from the website <https://www.croco-ocean.org>). At last, we will test our working environment.

STEP 1: Logging onto the Lengau HPC cluster

→ To connect to the CHPC you must write, from a terminal/konsole, the following instruction:

```
ssh -X login@lengau.chpc.ac.za
```



Replace **login** with your corresponding account number.

→ When entering the LENGAU super-computer, you will see something similar to this in your terminal:

```
Last login: Fri Sep 27 10:27:25 SAST 2024 from XX.XXX.XX.XXX
Welcome to LENGAU
#####
#
# In order to receive notifications via email from the CHPC all users should
# be subscribed to the CHPC user distribution list. If you are not part of the
# distribution list you can subscribe at the following link:
# https://lists.chpc.ac.za/mailman/listinfo/chpc-users
#
#####
[login@login2 ~]$
```

→ This will take you to your home directory on Lengau: **/home/login**



You should **NOT** use your home directory to save large datasets or run CROCO!
You should go to your directory on **lustre** to do all the processing and saving of data.
There is a link to it in your **/home/login** directory.

→ To list the files and directories inside your home directory, just type the Linux command **ls**:

```
[login@login2 ~]$ ls
lustre
[login@login2 ~]$
```



→ If the **lustre** symbolic link does not exist in your home directory, you have to create it with the Linux command **ln**:

```
[login@login2 ~]$ ln -sf /mnt/lustre/users/login lustre
[login@login2 ~]$ ls -l
lustre -> /mnt/lustre/users/login
[login@login2 ~]$
```



→ To know where your home directory is, just type the Linux command **pwd**:

```
[login@login2 ~]$ pwd
/home/login
[login@login2 ~]$
```



→ Go to your **lustre** directory by executing the Linux change directory command **cd**:

```
[login@login2 ~]$ cd lustre
[login@login2 lustre]$ pwd
/home/login/lustre
[login@login2 lustre]$
```





→ Create a **CROCO** directory (inside **lustre** directory) by executing the Linux **mkdir** command:

```
[login@login2 lustre]$ mkdir CROCO
[login@login2 lustre]$ cd CROCO
[login@login2 CROCO]$
```



STEP 2: Requesting some interactive nodes to manipulate large files

 On the Lengau cluster, you can **NOT** manipulate large files or perform calculations with MATLAB/python/nco softwares directly on the login node.

→ You need to **request an interactive node** on the compute server that will be reserved for your personal use. You need to execute your processes on these allocated node. If you execute a process on the frontal node **it will be killed by the system**. 

→ You need to open an interactive **pbs** session to request a node on the compute server. This can be done by typing the following in the command line:

```
qsub -X -I -l select=1:ncpus=1:mpiprocs=1 -q R6060705 -l
walltime=4:00:00 -P WCHPC
```



→ In the command above, we are requesting **1** processors on one node for **4** hours. We also have to specify the project we belong to, i.e. **WCHPC**. And we run on the CROCO Summer School queue: **R6060705**.

To make things easier, I created an alias: **qsubi1** to reserve 1 node for 10 hours. This alias is defined in my **.bashrc** script. The latter is a hidden file in your **home** directory (that you can see by adding the option **-a** to the **ls** command).

→ Go back to your home directory (~) and replace your **.bashrc** file with mine.

```
[login@login2 CROCO]$ cd
[login@login2 ~]$ cp /home/apps/chpc/earth/CROCO_Workshop/
CROCO_TRAINING_Basic/3_Some_files/.bashrc ~
[login@login2 ~]$ source ~/.bashrc
[login@login2 ~]$ cd lustre
```



 **ADVICE** Do not hesitate to use **tab autocomplete**: Start typing the file/command name in the terminal and then press the “Tab” key . If there is only one file/command that matches the typed characters, the system will automatically complete the name.

→ Request one node using the alias command **qsubi1** (it can take a few seconds/minutes):

```
[login@login2 lustre]$ qsubi1
qsub: waiting for job 4416950.sched01 to start
qsub: job 4416950.sched01 ready
[login@cnode0220 ~]$ pwd
/home/login
[login@cnode0220 ~]$
```

NODES

→ Now, the prompt is green. You can copy large files or do heavy computing .

 When you execute **qsubi1**, it sends you **back to your home directory**. You need to navigate back to the directory where you want to be in.



→ Go back into your **lustre/CROCO** directory:

```
[login@cnode0220 ~]$ cd lustre/CROCO
[login@cnode0220 CROCO]$
```

NODES

STEP 3: Copying the source code of the CROCO model (croco_v2.0.1)

We are going to copy of the CROCO code stable release version v2.0.1 (released in October 2024) in our **lustre/CROCO** directory.

 CROCO code source can be freely downloaded from the  website <https://www.croco-ocean.org/download> (<https://gitlab.inria.fr/croco-ocean/croco/-/archive/v2.0.1/croco-v2.0.1.tar.gz>)

➔ This file has **already been downloaded** and uncompressed on the Lengau cluster.

→ To copy the code in your **lustre/CROCO** directory, execute the following command:

```
cp -r /home/apps/chpc/earth/CROCO_Workshop/CROCO_TRAINING_Basic/1_CROCO_code/croco-v2.0.1 .
```

ADVICE

Do not hesitate to use **tab autocomplete**



<https://www.croco-ocean.org/download2>

CROCO includes the source code written in Fortran, and the pre- and post-processing tools, necessary to build and analyze realistic configurations, written in Matlab and Python. The code and tools are open source. The parts of the code inherited from the ROMS model are made available under the conditions of the MIT/X license, those inherited from the AGRIF library under the terms of the CeCILL-C license. The Matlab tools are made available under the terms of the GNU license, the Python tools under the terms of the GPL3 license.

Download stable release

Recommended download for non-developers : **last stable release**

- [Last CROCO stable release \(v2.0.1\)](#) : 07 October 2024, Bug-fix release.
- [CROCO_PYTOOLS stable release \(v1.0.2\)](#) : 04 October 2024, Bug-fix release.
- [Last CROCO_TOOLS stable release \(v2.0.0\)](#) : 22 April 2024, Major release.

→ Using the command **ls**, you can now see that the directory **croco-v2.0.1** has been copied in your **lustre/CROCO** directory:

```
[login@cnode0220 CROCO]$ ls
croco-v2.0.1
[login@cnode0220 CROCO]$
```

NODES


→ You can go into the **croco-v2.0.1** directory and list its content, using the following Linux commands: **cd** and **ls**:

```
[login@cnode0220 CROCO]$ cd croco-v2.0.1
[login@cnode0220 croco-v2.0.1]$ ls
AGRIF          MPI_NOLAND    OCEAN         SCRIPTS        create_config.bash
CHANGELOG.md   MUSTANG       PISCES        TEST_CASES
CVTK           OBSTRUCTION   README.md     XIOS
[login@cnode0220 croco-v2.0.1]$ cd ..
[login@cnode0220 CROCO]$
```

NODES

STEP 4: Copying the source code of the CROCO tools (crocotools_v2.0.0)

Then, we are going to get a copy of the CROCO_TOOLS (release version v2.0.0, from April 2024) in our **lustre/CROCO** directory.

 CROCO_TOOLS, along with some useful data sets, can be freely downloaded from the **CROCO** website <https://www.croco-ocean.org/download> (at https://gitlab.inria.fr/croco-ocean/croco_tools/-/archive/v2.0.0/croco_tools-v2.0.0.tar.gz; and https://data-croco.ifremer.fr/DATASETS/DATASETS_CROCOTOOLS.tar.gz).

→ These files have already been downloaded and uncompressed on the Lengau cluster.

<https://www.croco-ocean.org/download>

Datasets

CROCO_TOOLS uses several datasets also available for download:

All external datasets (8.9G): [DATASETS_CROCOTOOLS.tar.gz](#) [last update : 03/09/2024 ; see below]

- CARS2009 (7.7G): [CARS2009.tar.gz](#)
- COADS05 (103M): [COADS05.tar.gz](#)
- GOT99 (8.9M): [GOT99.2.tar.gz](#)
- GSHHS (256M): [gshhs.tar.gz](#)
- m_map (126M): [m_map1.4f.tar.gz](#)
- QuickSCAT (43M): [QuikSCAT_clim.tar.gz](#)
- SeaWifs: [SeaWifs.tar.gz](#)
- SST Pathfinder (14M): [SST_pathfinder.tar.gz](#)
- Topo (101M): [Topo.tar.gz](#)
- TPX06 (145M): [TPX06.tar.gz](#) [03/09/2024 updated version : add units attribute]
- TPX07 (148M): [TPX07.tar.gz](#) [03/09/2024 updated version : add units attribute]
- WOA2009 (219M): [WOA2009.tar.gz](#) [03/01/2022 updated version : bug in o2 fields units corrected]

→ To copy the CROCO_TOOLS code that we will use, execute the following command:

```
cp -r /home/apps/chpc/earth/CROCCO_Workshop/CROCO_TRAINING_Basic/2_CROCO_tools/croco_tools-v2.0.0 .
```

ADVICE

Do not hesitate to use **tab autocomplete**



→ Using the Linux command **ls**, you can now see that the directory **croco_tools-v2.0.0** has been copied in your **lustre/CROCO** directory:

```
[login@cnode0220 CROCO]$ ls
croco_v2.0.1 croco_tools-v2.0.0
[login@cnode0220 CROCO]$
```

NODES

→ You can go into the **croco_tools-v2.0.0** directory and list its content:

```
[login@cnode0220 CROCO]$ cd croco_tools-v2.0.0
[login@cnode0220 croco_tools-v2.0.0]$ ls
Aforc_CFSR          Diagnostic_tools    README.md
Aforc_ECMWF         example_job_prepro_matlab.pbs Rivers
Aforc_ERA5          Forecast_tools     RUNOFF_DAI
Aforc_NCEP          job_prepro_matlab.pbs start.m
Aforc_QuikSCAT      Nesting_tools      Tides
CHANGELOG.md        oct_start.m        Town
Coupling_tools      Oforc_OGCM         UTILITIES
croco_pyvisu        Opendap_tools      Visualization_tools
crocotools_param.m  Opendap_tools_no_loaddap
DATASETS_CROCOTOOLS Preprocessing_tools
[login@cnode0220 croco_tools-v2.0.0]$ cd ..
[login@cnode0220 CROCO]$
```

NODES

STEP 5: Testing the MATLAB Software

The CROCO_TOOLS are an ensemble of routines written to help you create input files that are needed to run your CROCO simulations (pre-processing) and make some analyses of the model outputs (post-processing). This toolbox uses the software MATLAB. Let's test if MATLAB works correctly on your account:



→ Using the Linux command **cp**, copy two test files in your **lustre/CROCO** directory:



```
cp /home/apps/chpc/earth/CROCO_Workshop/CROCO_TRAINING_Basic/3_Some_files/TP* .
```

→ List the content of your **lustre/CROCO**:

```
croco-v2.0.1 croco_tools-v2.0.0 TP0_test_file.nc TP0_test_script.m
[login@cnode0220 CROCO]$
```



→ Launch MATLAB software by typing **matlab -nodesktop** (or the alias **mat**):

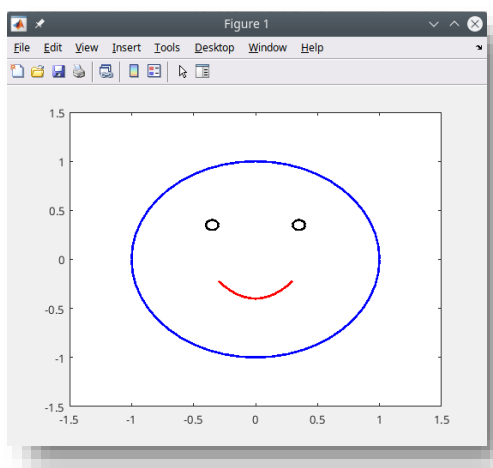
```
[login@cnode0220 CROCO]$ matlab -nodesktop
[login@cnode0220 CROCO]$
```



→ When MATLAB is launched, you will see something similar to this in your terminal:

```
< M A T L A B (R) >
Copyright 1984-2020 The MathWorks, Inc.
R2020a Update 8 (9.8.0.1873465) 64-bit (glnxa64)
February 3, 2022

To get started, type doc.
For product information, visit www.mathworks.com.
>>
```



→ Inside MATLAB, you can list the file using the MATLAB command **ls**:

```
>> ls
croco-v2.0.1 croco_tools-v2.0.0
TP0_test_file.nc TP0_test_script.m
>>
```



→ Execute the Matlab script **TP0_test_script**:

```
>> TP0_test_script
>>
```



→ The script opens a NetCDF file and plot its content. You should get a first plot revealing a smiling face.

```
>> exit
```

→ You can remove the **TP0_test_*** files:

```
[login@cnode0220 CROCO]$ rm TP0_test_*
rm: remove regular file 'TP0_test_file.nc'? y
rm: remove regular file 'TP0_test_script.m'? y
[login@cnode0220 CROCO]$ ls
croco-v2.0.1 croco_tools-v2.0.0
[login@cnode0220 CROCO]$
```



STEP 6: Exiting

→ Give back the interactive node and logout from Lengau:

```
[login@cnode0220 CROCO]$ exit
logout
qsub: job 4416950.sched01 completed
[login@login2 lustre]$ exit
```



STEP 7: Check List

→ Here is the list of the essential commands that you must execute during this hands-on session. The following table can help you confirm that you have executed all of them:

Commands



STEP 1	1	ssh -X login@lengau.chpc.ac.za	
	2	ls	
	3	ln -sf /mnt/lustre/users/login lustre	
	4	ls -l	
	5	pwd	
	6	cd lustre	
	7	pwd	
	8	mkdir CROCO	
	9	cd CROCO	
STEP 2	1	cd	
	2	cp /home/apps/chpc/earth/CROCCO_Workshop/CROCO_TRAINING_Basic/3_Some_files/.bashrc ~	
	3	source ~/.bashrc	
	4	cd lustre	
	5	qsub1	
	6	pwd	
	7	cd lustre/CROCO	
STEP 3	1	cp -r /home/apps/chpc/earth/CROCCO_Workshop/CROCO_TRAINING_Basic/1_CROCO_code/croco-v2.0.1 .	
	2	ls	
	3	cd croco-v2.0.1	
	4	ls	
	5	cd ..	
STEP 4	1	cp -r /home/apps/chpc/earth/CROCCO_Workshop/CROCO_TRAINING_Basic/2_CROCO_tools/croco_tools-v2.0.0 .	
	2	ls	
	3	cd croco_tools-v2.0.0	
	4	ls	
	5	cd ..	
STEP 5	1	cp /home/apps/chpc/earth/CROCCO_Workshop/CROCO_TRAINING_Basic/3_Some_files/TP0* .	
	2	ls	
	3	matlab -nodesktop	
	4	ls	
	5	TP0_test_script	
	6	exit	
	7	rm TP0_test*	
	8	ls	
STEP 6	1	exit	
	2	exit	