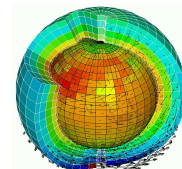


## MC-Toolkit meetings



Mickaël Lalande

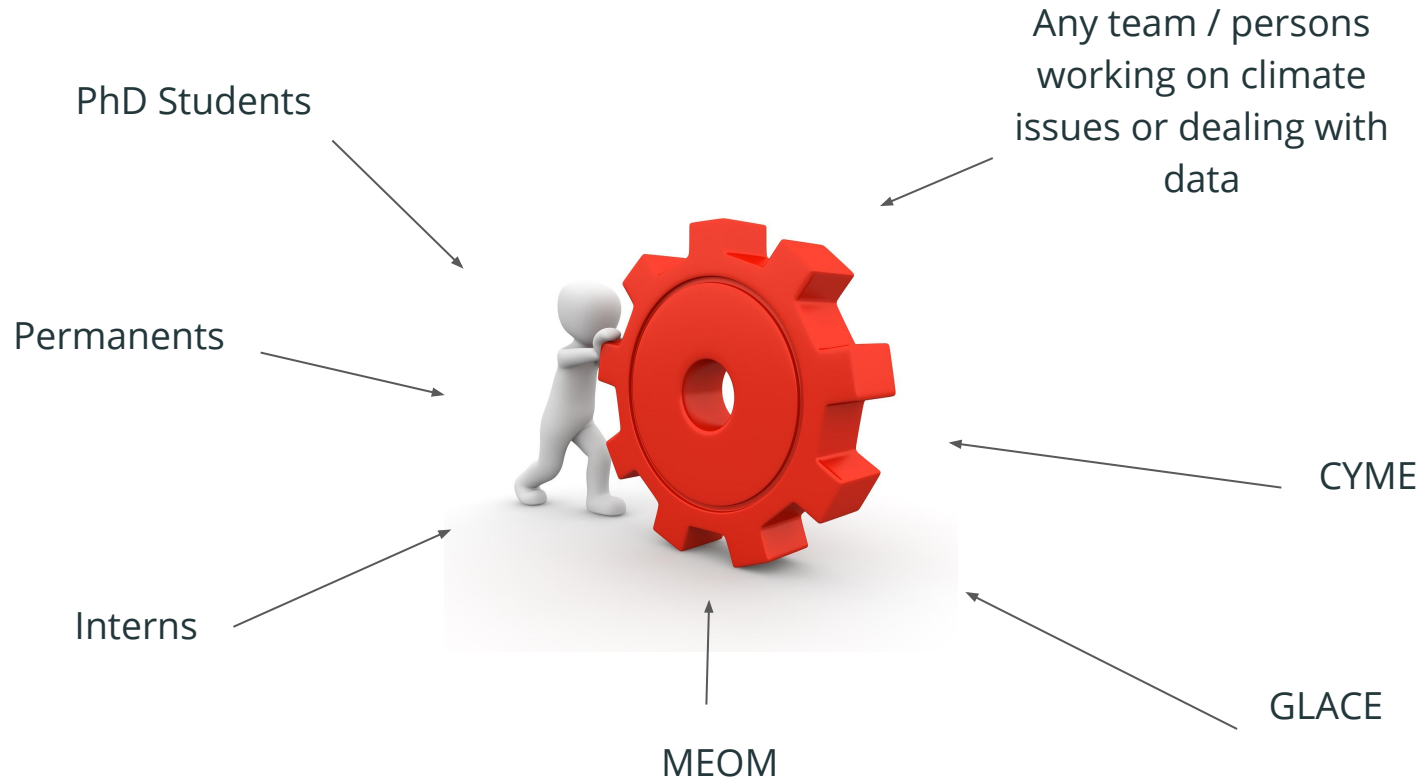


Thèse 2019-2022

Directeurs : Gerhard Krinner et Martin Ménégoz  
Institut des Géosciences de l'Environnement (IGE)



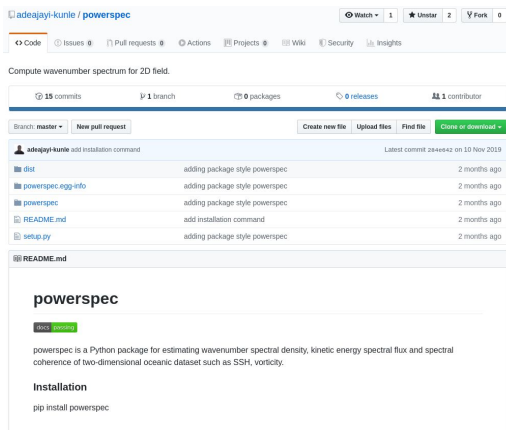
# Why this meeting?



# My personal experience in MEOM's team -> SATIM meetings



[brodeau/sosie: SOSIE is Only a Surface Interpolation Environment](#)



[adeajayi-kunle/powerspec: Compute wavenumber spectrum for 2D field.](#)



# Ideas of topics

## In the context of Mardi Café:

- xarray / cdo / climaf (netcdf, climatologies)
- cartopy / ferret / ncl / basemap / proplot (plot)
- cdo, basemap, scipy/stats, ESMx, climaf, sosie (regrid)
- Jupiter Notebook / Jupyter lab (pont ssh, Jupiter lab, plots interactifs)
- Github / Gitlab / svn / bitbucket
- dask (parallelization or MemoryError issues)
- Machine Learning ? (gpu, slack, gricad) -> ml-at-ige slack
- régression linéaire / tendances / analyses statistiques
- EOF
- analyses spectrales
- cmip6 data / reanalyses / obs
- MAR / LMDZ (how to install and launch a simulation?)

## More general:

- How to write a paper (latex, overleaf, texmaker, etc.)
- How to deal with the bibliography (mendeley, zotero, etc.)
- How to publish an article (what journals, etc.)

# Example of CLIMAF

[https://github.com/mickaellalande/MC-Toolkit/blob/master/Presentation/CLIMAF\\_example.ipynb](https://github.com/mickaellalande/MC-Toolkit/blob/master/Presentation/CLIMAF_example.ipynb)

jupyter CLIMAF\_example (auto-sauvegardé)



Se déconnecter

Fichier Édition Affichage Insérer Cellule Noyau Widgets Aide

Non fiable

Python 3

Exécuter

Markdown

## CLIMAF example (works only on CICLAD)

More informations about CLIMAF: <https://climaf.readthedocs.io/en/master/>

### Import main modules

```
Entrée [1]: from climaf.api import *

('LC debug :', False)
CLIMAF install => /ciclad-home/jservon/Evaluation/CLIMAF/climaf_installs/climaf_1.2.12
python => /prodigfs/ipslfs/dods/jservon/miniconda/envs/analyse_env_2.7/bin/python
...
Required softwares to run CLIMAF => you are using the following versions/installations:

CLIMAF version = 1.2.12

ncl 6.6.2 => /prodigfs/ipslfs/dods/jservon/miniconda/envs/analyse_env_2.7/bin/ncl
cdo 1.9.6 => /opt/nco/1.9/bin/cdo
nco (ncks) 4.5.2 => /opt/nco-4.5.2/bin/ncks
ncdump fichier => /prodigfs/ipslfs/dods/jservon/miniconda/envs/analyse_env_2.7/bin/ncdump
...

Cache directory set to : /data/mlalande/climafcache (use $CLIMAF.CACHE if set)
Cache directory for remote data set to : /data/mlalande/climafcache/remote_data (use $CLIMAF_REMOTE_CACHE if set)
warning : Binary cdfutils not found. Some operators won't work
Available macros read from ~/.climaf.macros are : []
```


```
Entrée [2]: # The years considered to plot the biases between IPSL and observational references
first_year=1984
last_year=2014
```

### "Load" model data


```
Entrée [3]: req_snow = ds(
    project='CMIP6',
    model='IPSL-CM6A-LR',
    variable='snc',
    tables='LImon',
    frequency='monthly',
    realization='r1i1p1f1',
    period=str(first_year)+'-'+str(last_year),
    ...)
```

# Example for xarray

[https://github.com/mickaellalande/MC-Toolkit/blob/master/Presentation/xarray\\_example.ipynb](https://github.com/mickaellalande/MC-Toolkit/blob/master/Presentation/xarray_example.ipynb)












 jupyter

xarray\_example (auto-sauvegardé)

 Se déconnecter

FichierÉditionAffichageInsérerCelluleNoyauWidgetsAide

FiabilitéPython 3



Exécuter

Code

## Xarray example

Load a snow data set from a model (IPSL-CM6A-LR) and compare it to a reanalysis (NOAA) by seasons during the period 1984 to 2014.

### Import main modules

For installation see the Readme at the root of this repository

```
Entrée [1]: # This first line allows to have interactive plots inside the notebook
%matplotlib notebook

import xarray as xr # xarray is for dealing with NetCDF files
import matplotlib.pyplot as plt # for plot
import numpy as np # for maths
import xesmf as xe # for regridding
```

### Load model data

Note: xarray doesn't directly load the data. It uses "lazy load", and only load data when a operation is occurring. There are many possibilities to take advantage of this + using Dask if memory issues. See xarray\_dask\_xesmf\_proplot folder (when it will be available).

```
Entrée [2]: model_dataset = xr.open_dataset(
    "data/snc_LImon_IPSL-CM6A-LR_historical_rlilplf1_gr_185001-201412.nc"
)
model_dataset
```

```
Out[2]: <xarray.Dataset>
Dimensions:      (axis_nbounds: 2, lat: 143, lon: 144, time: 1980)
Coordinates:
  * lat          (lat) float32 -90.0 -88.73239 -87.46479 ... 88.73239 90.0
  * lon          (lon) float32 0.0 2.5 5.0 7.5 10.0 ... 350.0 352.5 355.0 357.5
  * time         (time) datetime64[ns] 1850-01-16T12:00:00 ... 2014-12-16T12:00:00
Dimensions without coordinates: axis_nbounds
Data variables:
  time_bounds    (time, axis_nbounds) datetime64[ns] ...
  snc            (time, lat, lon) float32 ...
Attributes:
  Conventions:    CF-1.7 CMIP-6.2
  creation date:  2018-07-11T07:36:35Z
```

# Example “ma cuisine”

[https://github.com/mickaellalande/MC-Toolkit/blob/master/Presentation/ma\\_cuisine.py](https://github.com/mickaellalande/MC-Toolkit/blob/master/Presentation/ma_cuisine.py)

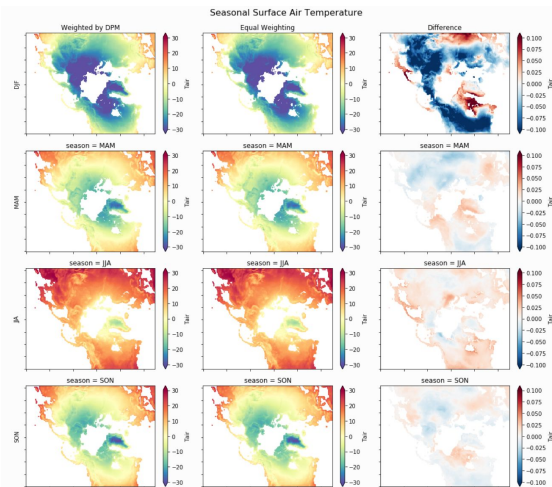
```
jupyter ma_cuisine.py il y a quelques secondes Se déconnecter
Fichier Édition Affichage Langage Python

49 # =====
50 # Compute monthly weighted data
51 # =====
52 # http://xarray.pydata.org/en/stable/examples/monthly-means.html
53 dpm = {'noleap': [0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31],
54        '365 day': [0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31],
55        'standard': [0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31],
56        'gregorian': [0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31],
57        'proleptic_gregorian': [0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31],
58        'all leap': [0, 31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31],
59        '366 day': [0, 31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31],
60        '360 day': [0, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30]}
61
62 def leap_year(year, calendar='standard'):
63     """Determine if year is a leap year"""
64     leap = False
65     if ((calendar in ['standard', 'gregorian',
66                      'proleptic_gregorian', 'julian']) and
67         (year % 4 == 0)):
68         leap = True
69     if ((calendar == 'proleptic_gregorian') and
70         (year % 100 == 0) and
71         (year % 400 != 0)):
72         leap = False
73     elif ((calendar in ['standard', 'gregorian']) and
74          (year % 100 == 0) and (year % 400 != 0) and
75          (year < 1583)):
76         leap = False
77     return leap
78
79 def get_dpm(time, calendar='standard'):
80     """
81     return a array of days per month corresponding to the months provided in 'months'
82     """
83     month_length = np.zeros(len(time), dtype=np.int)
84     cal_days = dpm[calendar]
85     for i, (month, year) in enumerate(zip(time.month, time.year)):
86         month_length[i] = cal_days[month]
87         if leap_year(year, calendar=calendar) and month == 2:
88             month_length[i] += 1
89     return month_length
90
91 # Seasonal climatology (on monthly data set)
92 def season_clim(ds, calendar='standard'):
93     """
94     # Make a DataArray with the number of days in each month, size = len(time)
95     month_length = xr.DataArray(get_dpm(ds.time.to_index(), calendar=calendar),
96
```

How to take into account the number of days in a month for monthly data analyse?

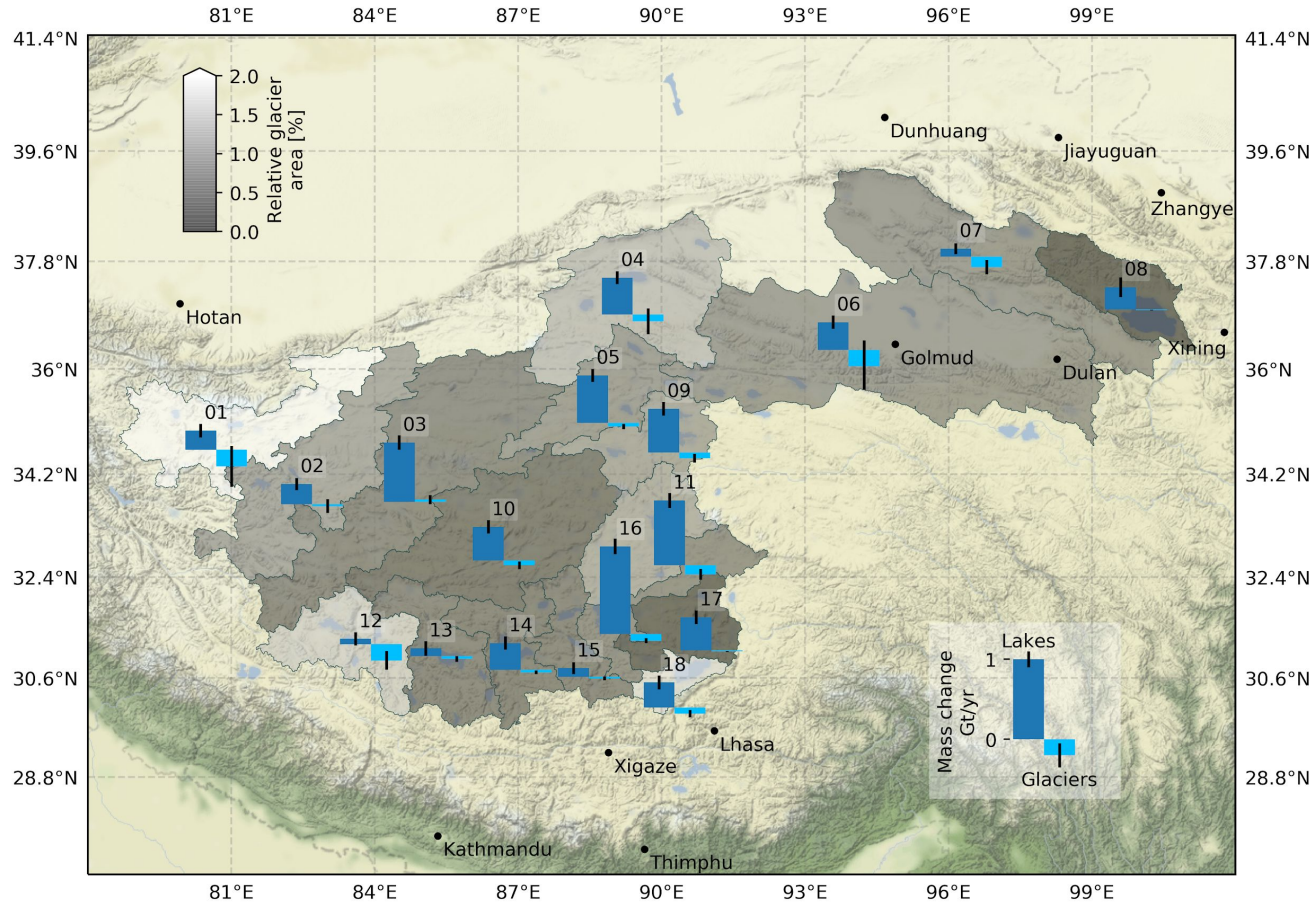
Code inspired from:

[Calculating Seasonal Averages from Timeseries of Monthly Means — xarray 0.14.1 documentation](#)





# Example plots Fanny (with Cartopy)

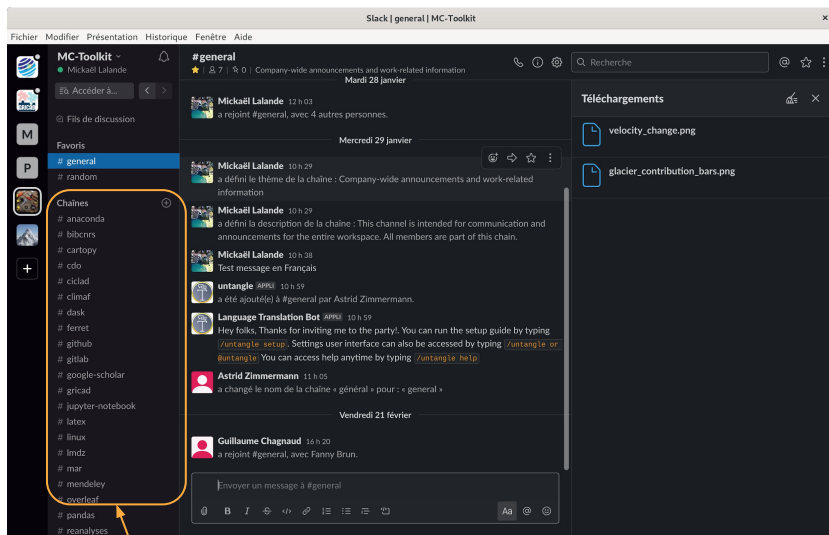




# Slack + github

Lien d'invitation **Slack**:

[https://join.slack.com/t/mc-toolkit-ige/shared\\_invite/enQtOTgxMjgzNzgyMjYzLWNlZGUzNTBIN2E0ZDIwNmI1MWRmNTVmNWE3NWUzMjMxN2Y1NGIODE2M2NlZDgxNjYxODAyMjIwODdhMTA0ODA](https://join.slack.com/t/mc-toolkit-ige/shared_invite/enQtOTgxMjgzNzgyMjYzLWNlZGUzNTBIN2E0ZDIwNmI1MWRmNTVmNWE3NWUzMjMxN2Y1NGIODE2M2NlZDgxNjYxODAyMjIwODdhMTA0ODA)

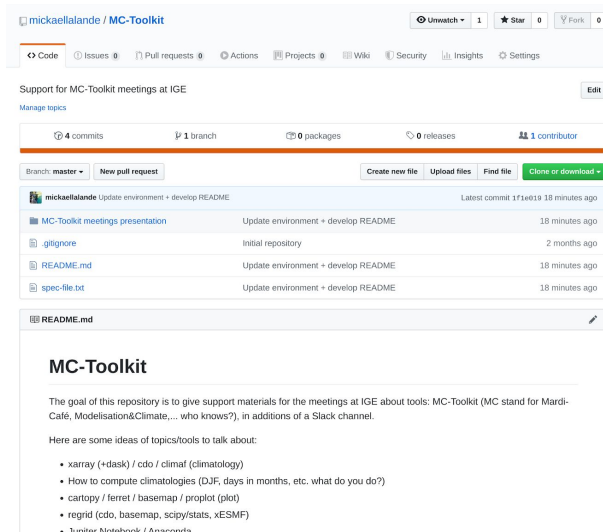


**Add the channels that you are interested**  
So that we can ask question and help each other  
with the regarding tool

## Github

(to find back the code examples shown during these meetings):

<https://github.com/mickaellalande/MC-Toolkit>



# Next meeting?

Same time?

In the context of Mardi Café or other?

MC for Mardi Café / Modélisation Climat / ??

I can start with xarray, climato (ma cuisine), proplot, xesmf  
(+ anaconda/jupyter-notebooks/github ? May be before ?)

## **Possible next meetings:**

0. Anaconda/Miniconda + Jupyter-Notebook?
  1. Martin: CLIMAF + R tool
  2. Mickaël: xarray + “ma cuisine”
  3. Mickaël / Fanny: plots (cartopy/proplot)
  4. Aurélie: Dask+xarray / Pangeo
  5. ...