



Journées Calcul et Données
9-11 Oct. 2019 | Toulouse



Analyse de simulations numériques de l'océan en préparation aux missions satellite : cas d'utilisation des outils PANGEO

A. Albert, F. Briol, L. Brodeau, G. Dibarboure, G. Eynard-Bontemps,
J. Le Sommer, A. Ponte



Plan de la présentation

1. Analyse de simulations océaniques
2. Les outils et la communauté PANGEO
3. Exemples d'applications et performances
4. Démonstration live
5. Conclusions et perspectives



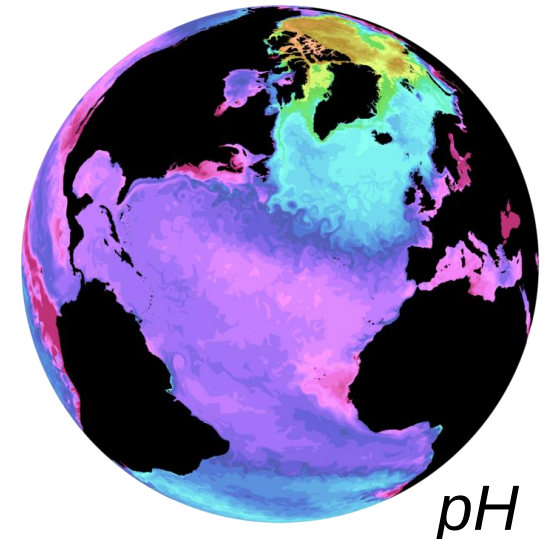
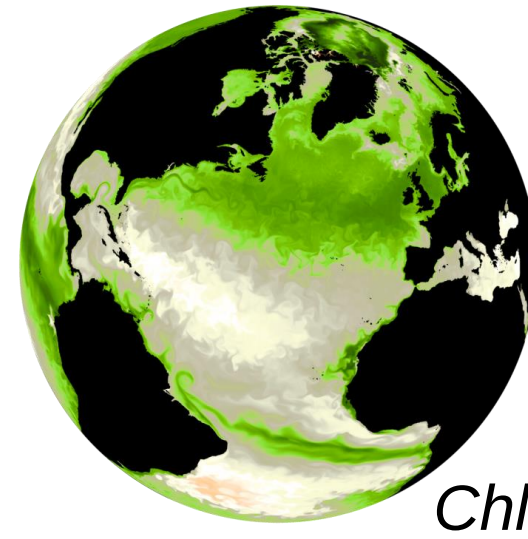
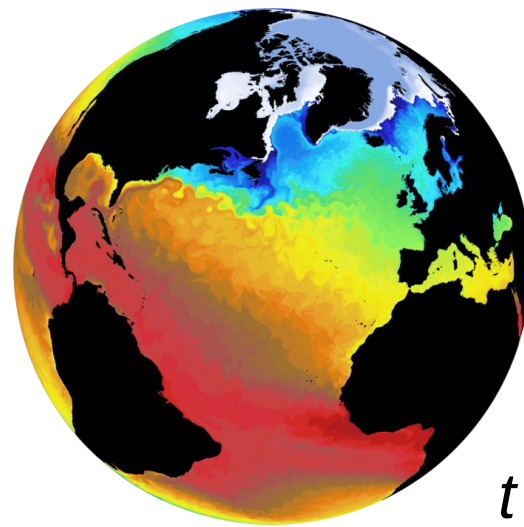
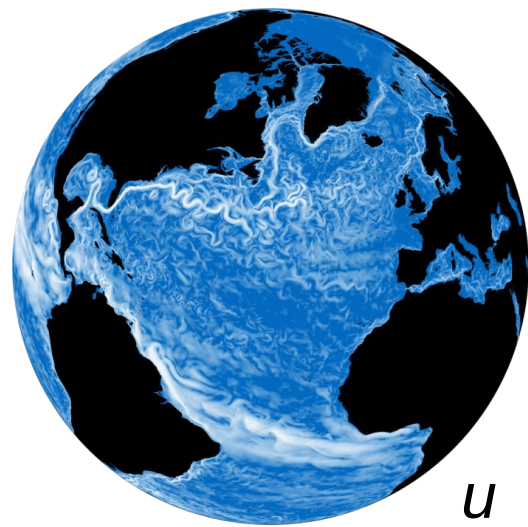
Plan de la présentation

1. Analyse de simulations océaniques
2. Les outils et la communauté PANGEO
3. Exemples d'applications et performances
4. Démonstration live
5. Conclusions et perspectives



Analyser des simulations océaniques ?

Les modèles de circulation océaniques



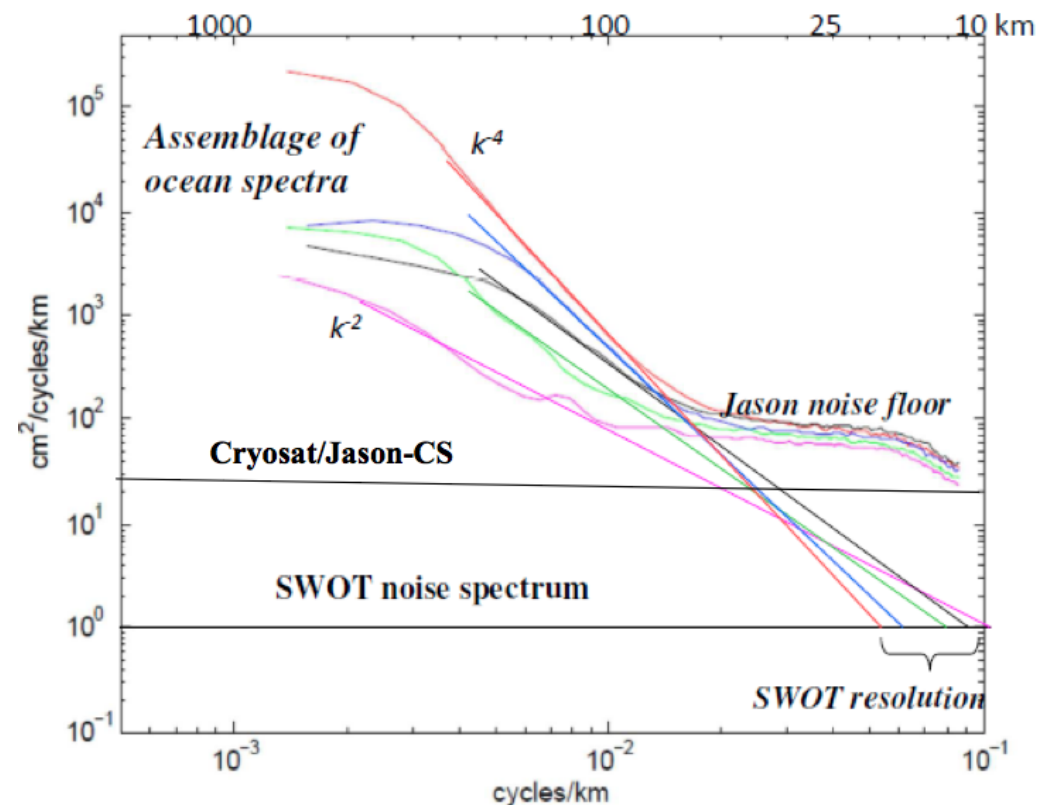
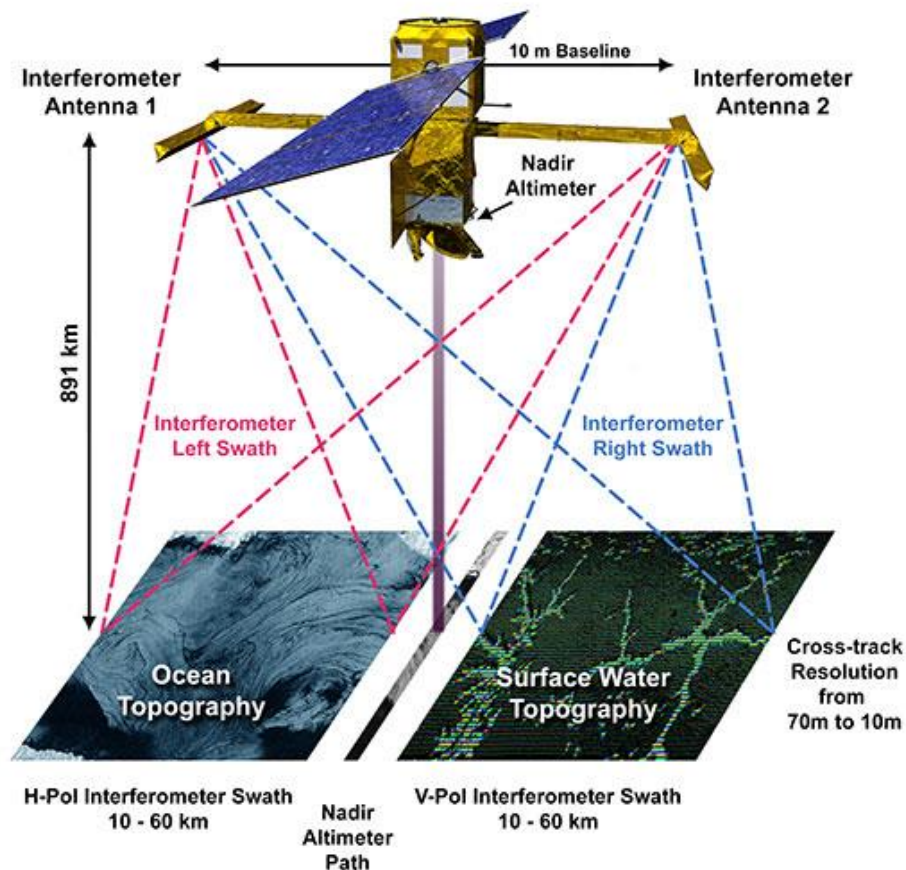
sont des outils numériques utilisés pour :

- comprendre les processus qui pilotent la variabilité océanique;
- des projections climatiques (composantes de modèles du système terre);
- estimer l'état 4D de l'océan à partir d'observation (réanalyses);
- prévoir l'évolution à court terme de l'océan (prévision opérationnelle)
- pour préparer les futurs systèmes d'observations (in situ/satellite)



Analyser des simulations océaniques ?

Préparer la future mission altimétrique SWOT



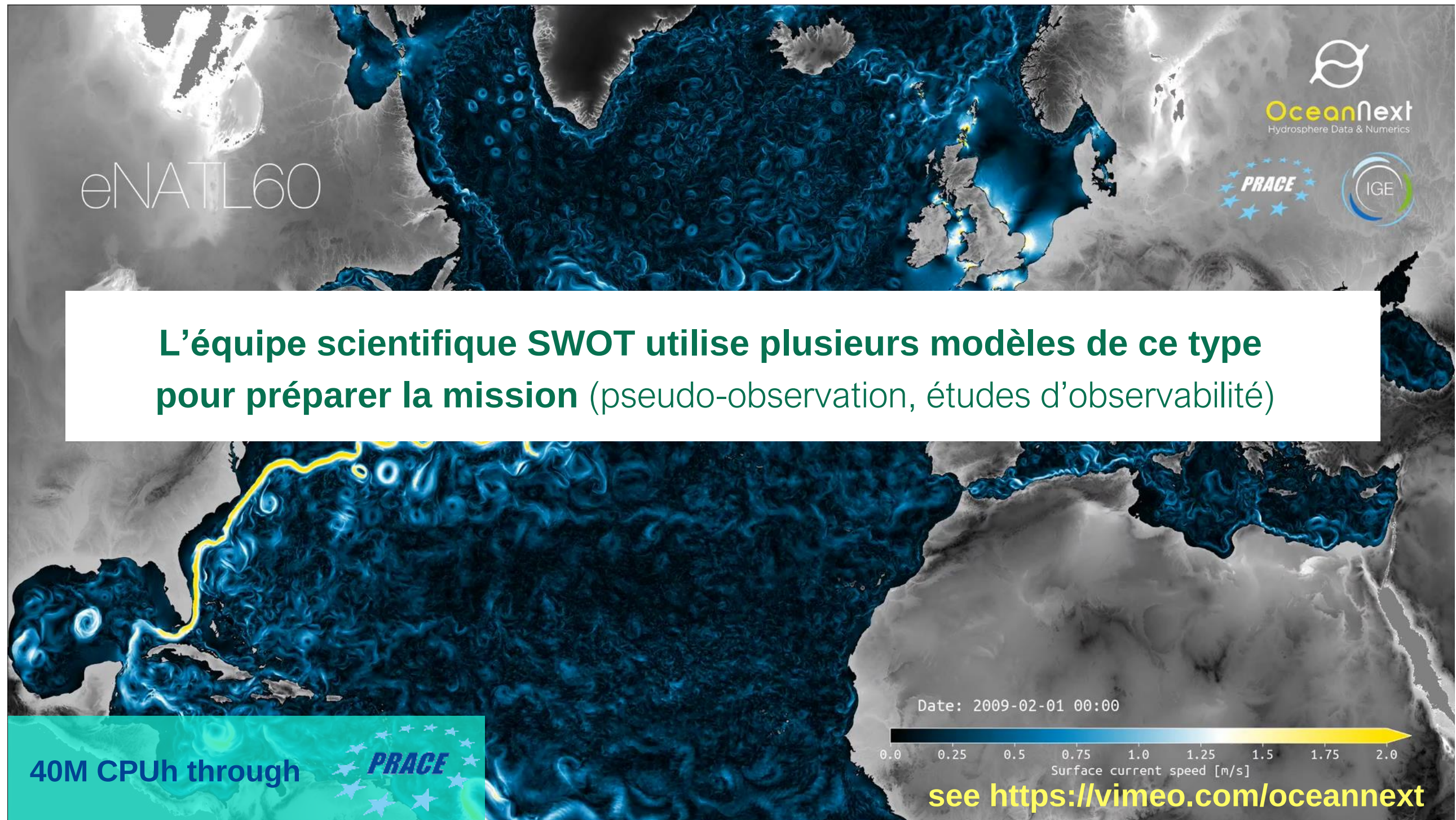
La mission SWOT mesurera la dénivellation de surface de l'océan (SSH) (lancement prévu fin 2021).
Le concept de mesure innovant de SWOT permettra d'observer la SSH jusqu'à ~15km.
Une source d'information essentielle pour les systèmes de prévision océanique.

Pour préparer les chaînes de traitement et l'exploitation scientifique de la mission, l'équipe scientifique SWOT utilise une nouvelle génération de modèles



Analyser des simulations océaniques ?

Modèles océaniques “submesoscale permitting” à large emprise



Simulations NEMO-eNATL60 de l'IGE/Ocean Next : $dx \sim 1\text{km}$, 300 levels + marée



Analyser des simulations océaniques ?

Les approches d'analyse de simulations océaniques remises en question

Pratiques classiques:

- Output : collection de fichiers format NetCDF (total: ~100 Gb to ~1 Pb)
- Les post-traitements intensifs tournent sur les machines HPC
- Codes maison spécialisés : tâches spécifiques / formats spécifiques
- Workflow à deux niveaux : post processing HPC / visualisation locale

Limites de ces pratiques

- *Complexité* : accélérer le temps du prototypage au déploiement à l'échelle
- *Reproductibilité* : pouvoir reproduire les analyses sur différents modèles
- *"big data"* : difficile de partager les données (2015:100Tb, 2019: 2Pb!)
- *Innovation* : les analyses mettent en oeuvre de l'apprentissage automatique

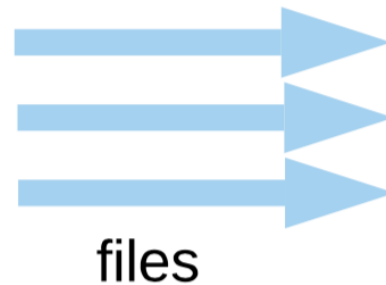
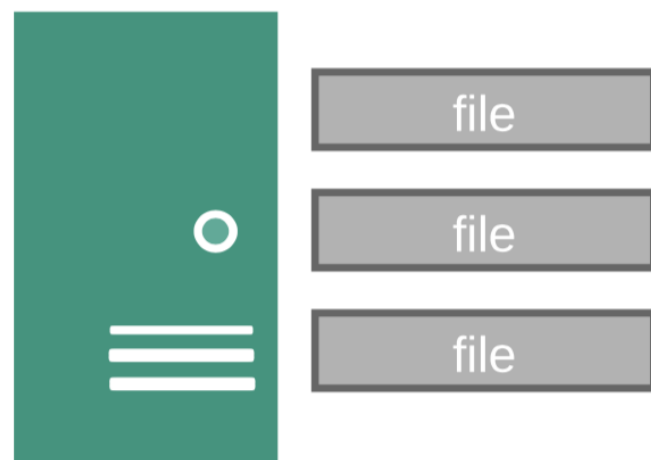


Analyser des simulations océaniques ?

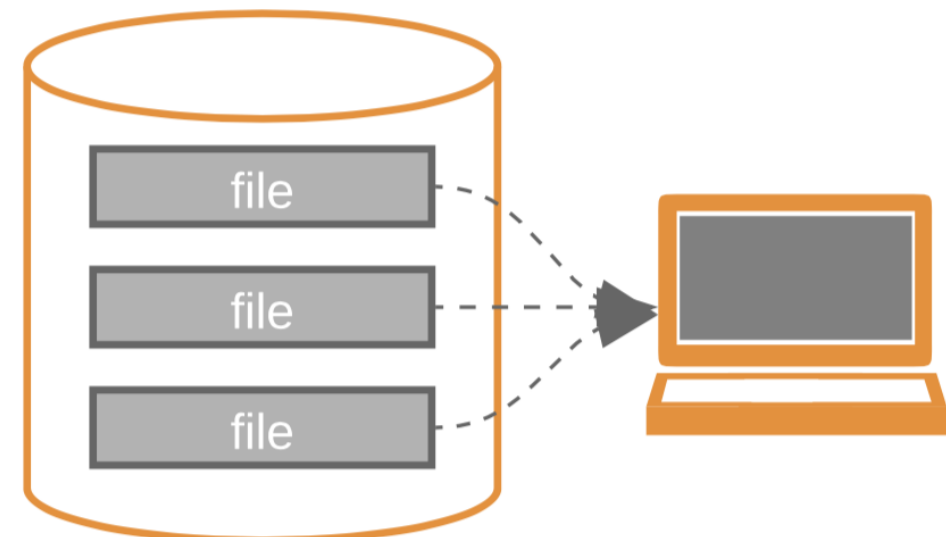
Cette présentation expose une transition en court dans nos communautés

anciennes pratiques

step 1: download



step 2: analyze



local disk

Data provider's responsibilities

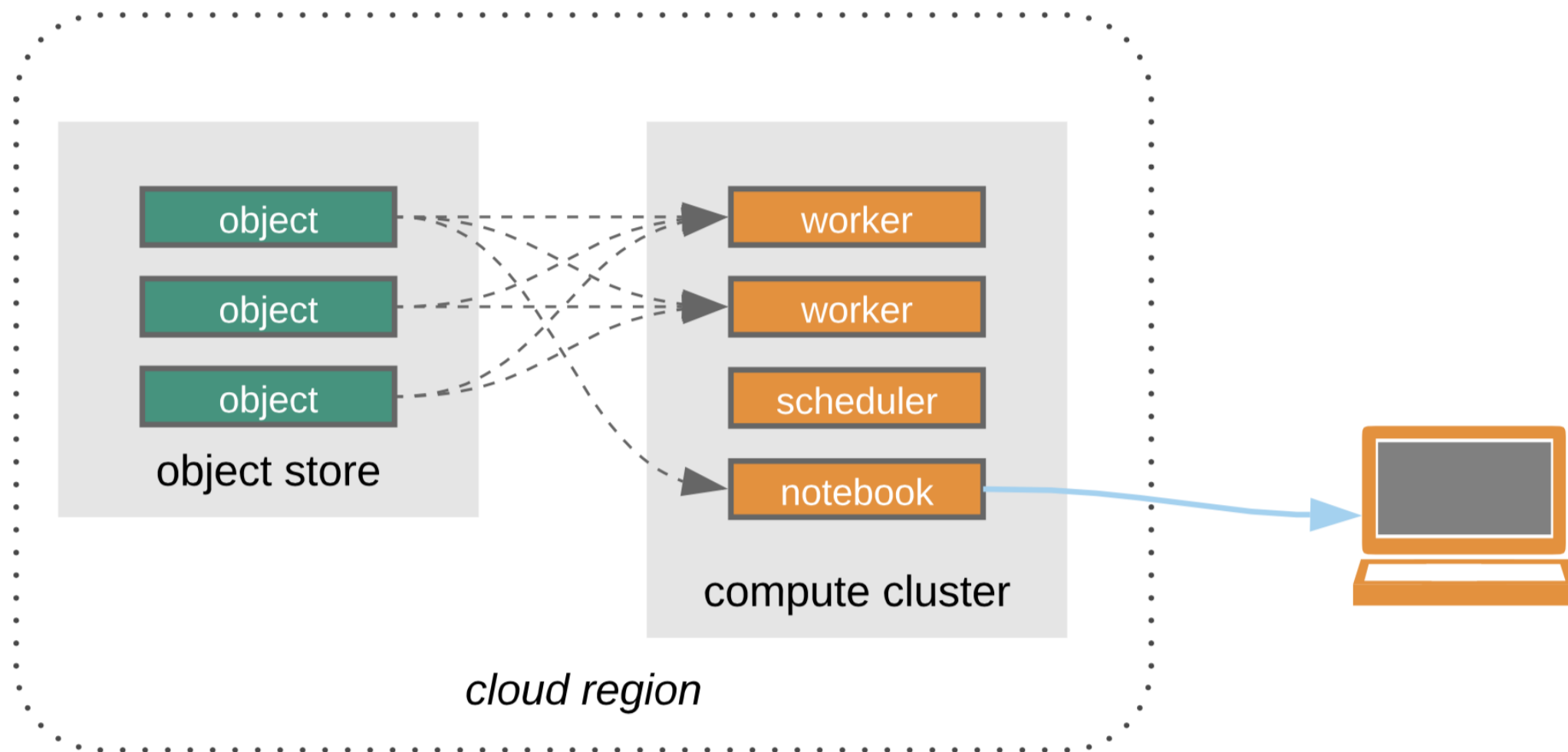
End user's responsibilities



Analyser des simulations océaniques ?

Nos approches d'analyse de simulations océaniques remises en question

pratiques émergentes



Data provider's responsibilities

End user's responsibilities



Plan de la présentation

1. Analyse de simulations océaniques
2. Les outils et la communauté PANGEO
3. Exemples d'applications et performances
4. Démonstration live
5. Conclusions et perspectives



PANGEO : qu'est-ce que c'est ?



A community platform
for Big Data geoscience

- ✦ Un écosystème logiciels “big data geosciences”
- ✦ Une communauté internationale de développeurs
- ✦ Une infrastructure partagée sur le cloud



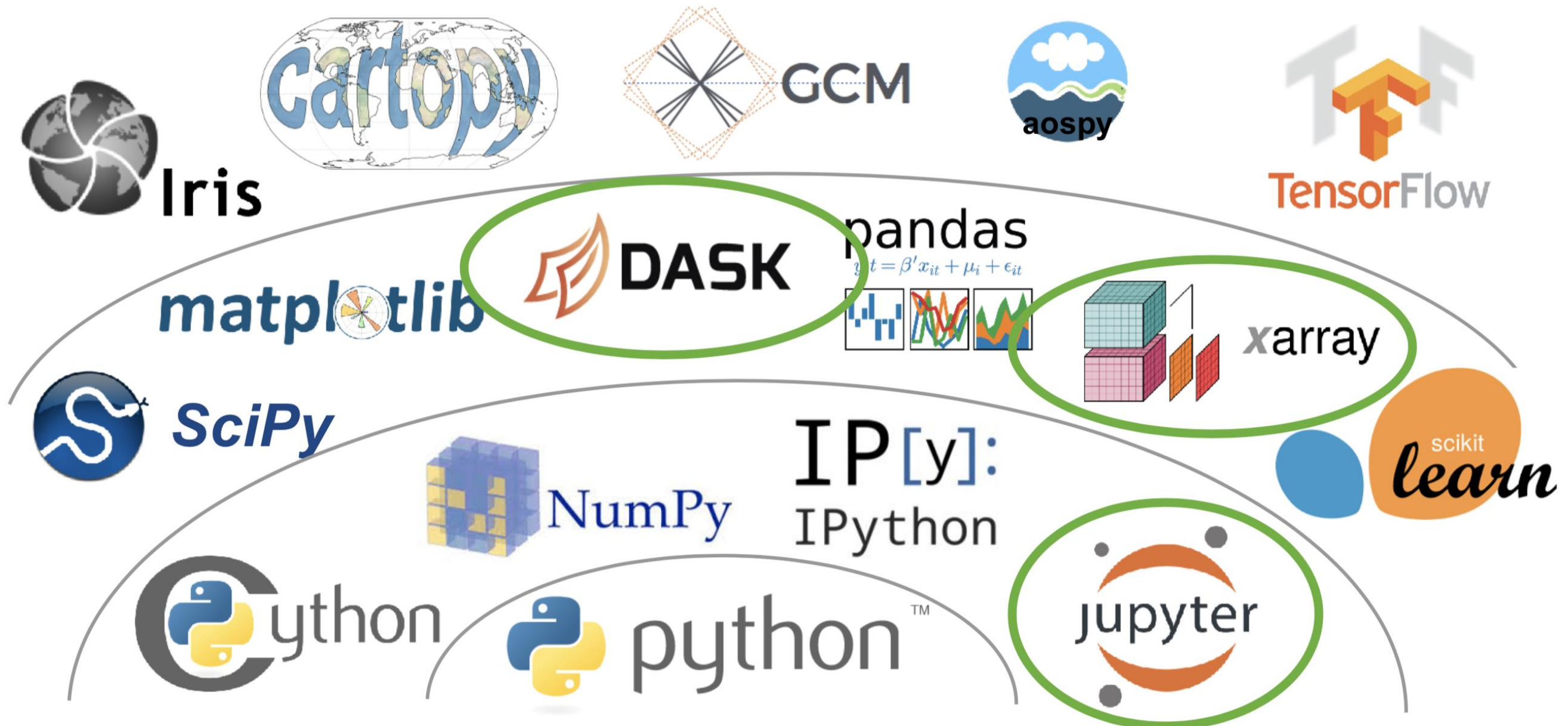
PANGEO : la communauté et les soutiens



- Une communauté rassemblant des chercheurs/développeurs :
 - à laquelle contribuent de grandes institutions (ECMWF, UKMO, CNES, ...)
 - une approche de travail ouverte et inclusive (github, blog, tutorial, etc.)
 - des workshops annuels aux US et en Europe (~30 p. à Paris en mai 2019)



PANGEO : briques logicielles de base



Credit: Stephan Hoyer, Jake Vanderplas (SciPy 2015)



PANGEO : briques logicielles de base



- Un environnement de calcul interactif “web-based”
- Un format de document reproductible :
 - Code
 - Texte
 - Equations (LaTeX)
 - Visualisations

...back to oxygen

Set contour levels to non-uniform intervals and make the colormap centered at the hypoxic threshold using `DivergingNorm`.

```
In [7]: levels = [0, 10, 20, 30, 40, 50, 60, 80, 100, 125, 150, 175, 200, 225,
                250, 275, 300]

norm = colors.DivergingNorm(vmin=levels[0], vmax=levels[-1], vcenter=60.)
```

Add a cyclic point to accomodate the periodic domain.

```
In [8]: from cartopy.util import add_cyclic_point
field, lon = add_cyclic_point(ds.O2, coord=ds.lon)
lat = ds.lat
```

Putting it all together...

```
In [9]: fig = plt.figure(figsize=(12, 8))
ax = fig.add_subplot(1, 1, 1, projection=ccrs.Robinson(central_longitude=305.0))

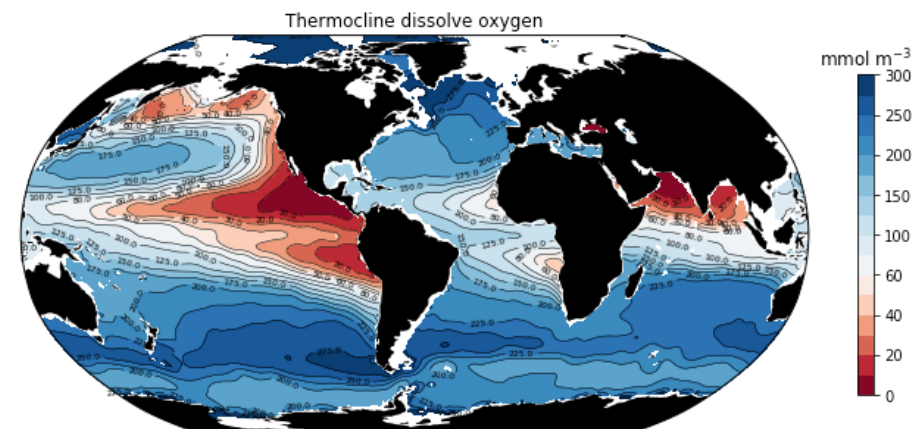
# filled contours
cf = ax.contourf(lon, lat, field, levels=levels, norm=norm, cmap='RdBu',
                transform=ccrs.PlateCarree());

# contour lines
cs = ax.contour(lon, lat, field, colors='k', levels=levels, linewidths=0.5,
                transform=ccrs.PlateCarree())

# add contour labels
lb = plt.clabel(cs, fontsize=6, inline=True, fmt='%x');

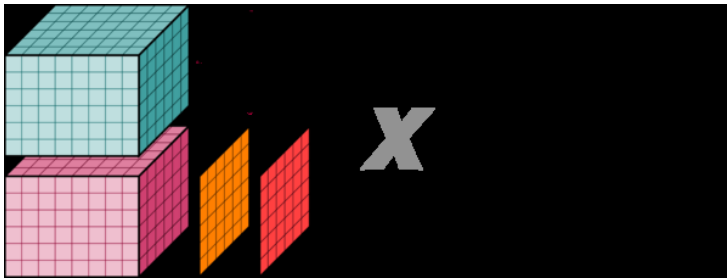
# land
land = ax.add_feature(
    cartopy.feature.NaturalEarthFeature('physical', 'land', '110m', facecolor='black'))

# colorbar and labels
cb = plt.colorbar(cf, shrink=0.5)
cb.ax.set_title('mmol m-3')
ax.set_title('Thermocline dissolve oxygen');
```

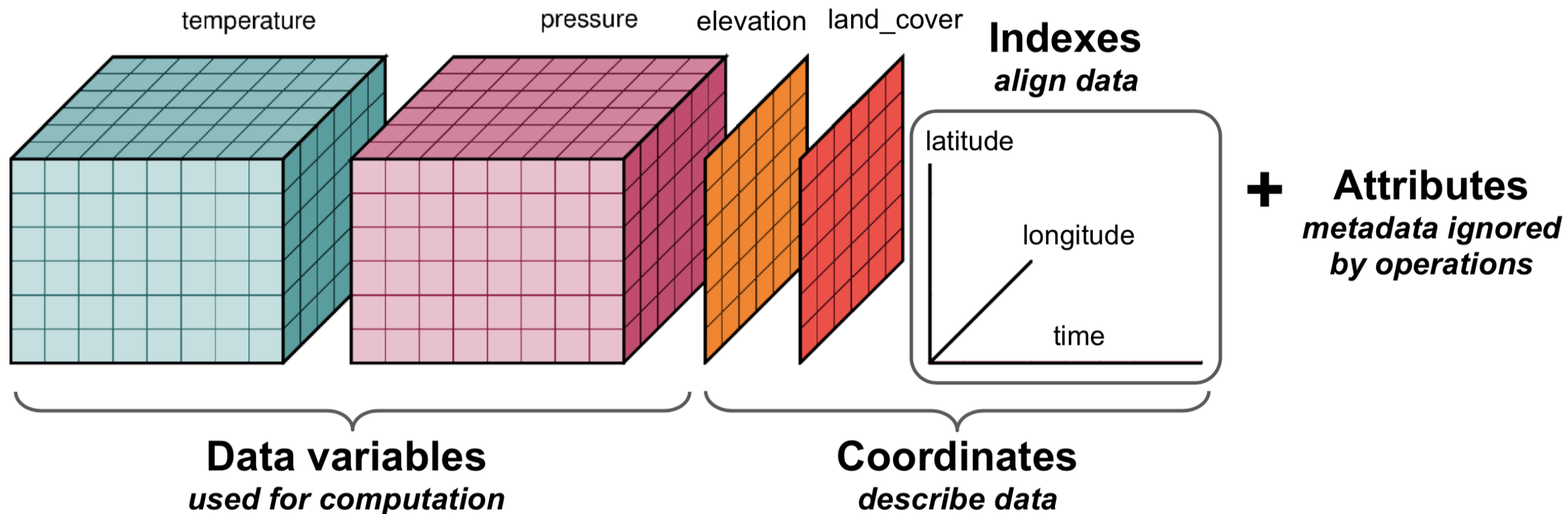




PANGEO : briques logicielles de base



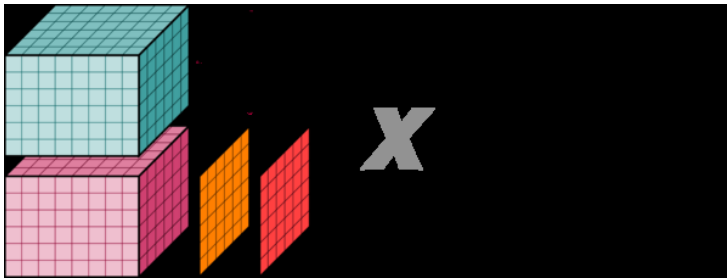
- Structures de données pour tableaux N-D
- Une API inspirée de pandas
- Interopérable avec les outils standards pydata



“netCDF meets pandas.DataFrame”



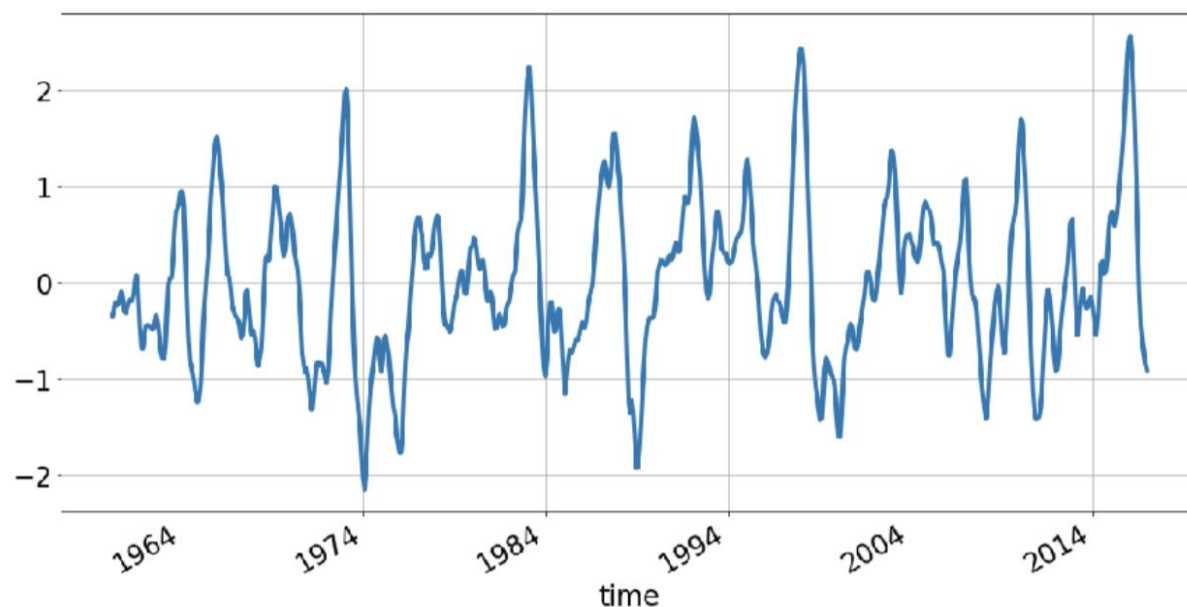
PANGEO : briques logicielles de base



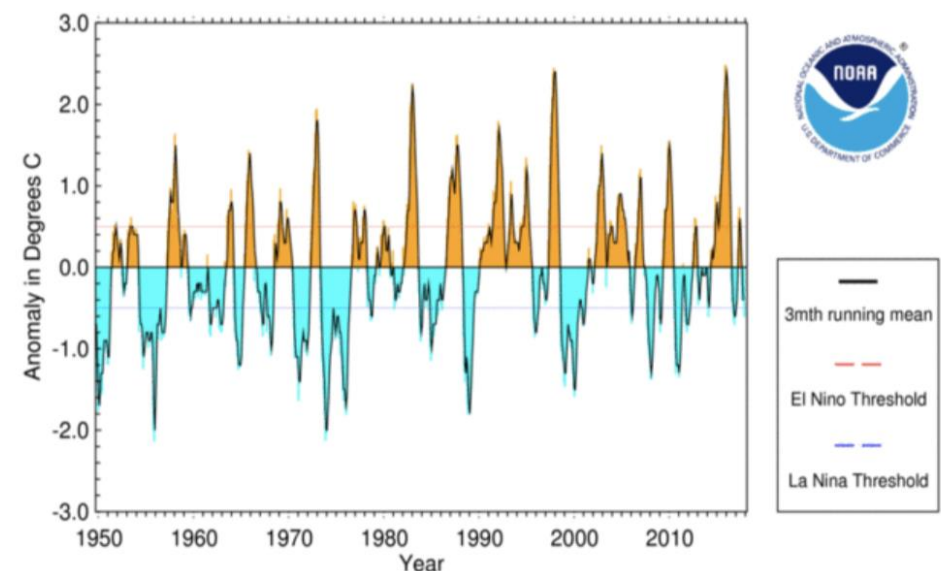
- Structures de données pour tableaux N-D
- Une API inspirée de pandas
- Interopérable avec les outils standards pydata

```
sst_clim = sst.groupby('time.month').mean(dim='time')
sst_anom = sst.groupby('time.month') - sst_clim
nino34_index = (sst_anom.sel(lat=slice(-5, 5), lon=slice(190, 240))
                .mean(dim=('lon', 'lat'))
                .rolling(time=3).mean(dim='time'))

nino34_index.plot()
```



SST Anomaly in Nino 3.4 Region (5N-5S,120-170W)

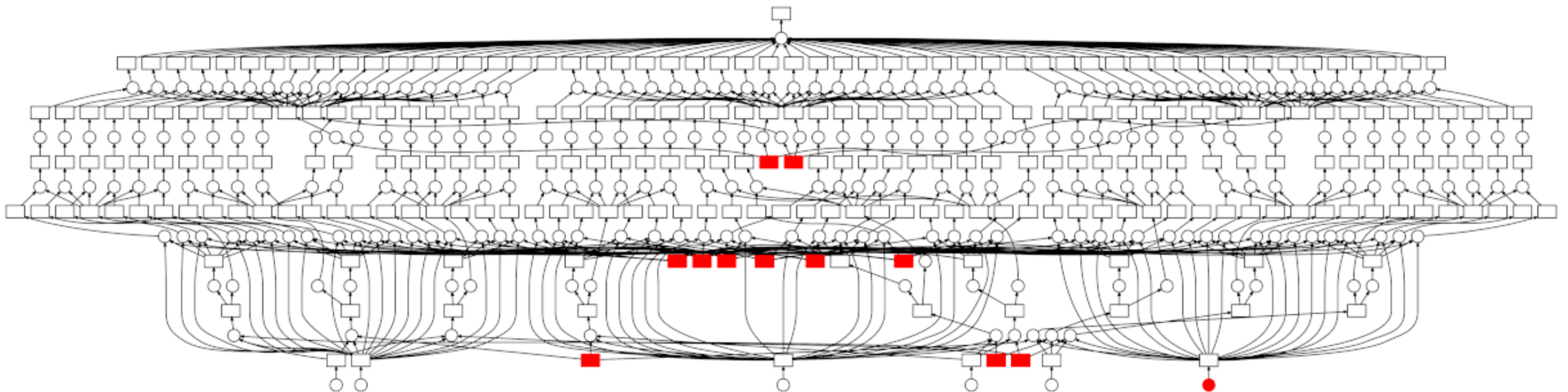




PANGEO : briques logicielles de base

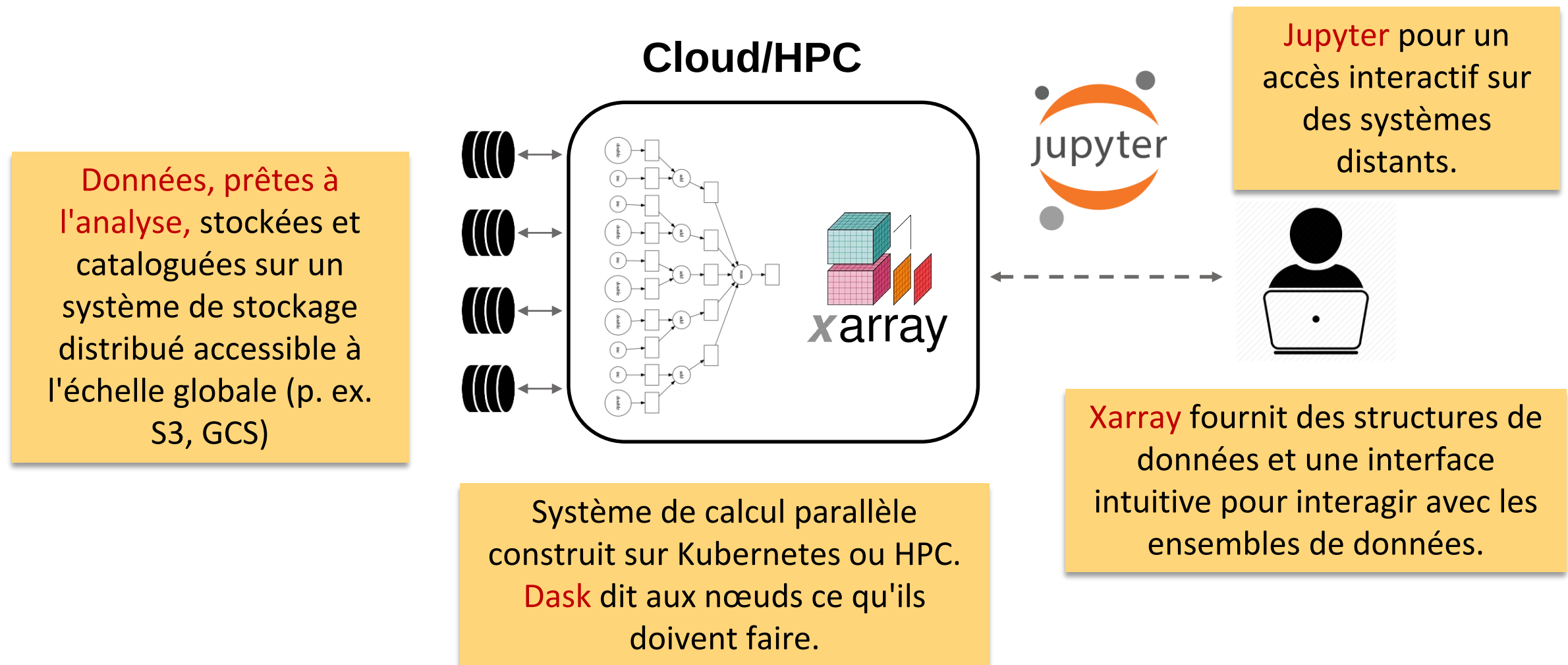


- Une librairie de calcul parallèle pour python
- Permet de passer à l'échelle Numpy, Pandas, Scikit-Learn, Xarray...
- Peut être déployé sur des systèmes HPC ou sur le cloud
- Culturellement issue du monde du calcul scientifique
- Apporte un gestionnaire d'exécution de graphes de tâches





PANGEO : infrastructure et déploiements



- Un déploiement ouvert sur le cloud (Jupyterhub) pangeo.pydata.org
- Nombreux déploiements sur cluster/HPC : <https://pangeo.io/deployments.html>
- Dont les systèmes HPC du NCAR et de la NASA, et le cluster HAL du CNES.
- Des tests en cours sur les HPC GENCI (Occigen) (sans jupyterhub)



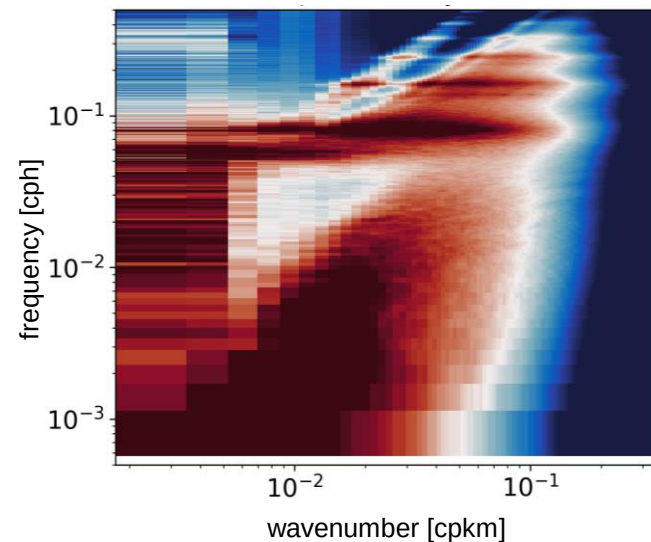
Plan de la présentation

1. Analyse de simulations océaniques
2. Les outils et la communauté PANGEO
3. Exemples d'applications et performances
4. Démonstration live
5. Conclusions et perspectives



Exemples d'analyses en préparation à SWOT

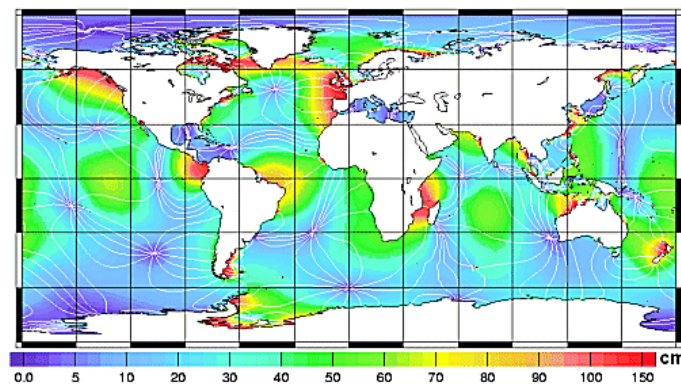
Analyse temps-espace des courants de surface



données ~1Tb

**Analyse de simulations réalisées dans
la SWOT Science Team française en
2019 avec PANGEO.**

Décomposition harmonique des hauteurs de surface



données ~5Tb

Machines :

Occigen@CINES, HAL@CNES, cloud pangeo

N. B. Analyses répétées sur plusieurs modèles.



Deux outils clés pour ces analyses

DASK-JOBQUEUE

- Pour déployer Dask sur des gestionnaires de job (PBS, Slurm, etc...)
- Un projet spin-off de Pangeo (avec de contributeurs toulousains !)
- Une interface “pythonique” pour manipuler les dask workers/clusters

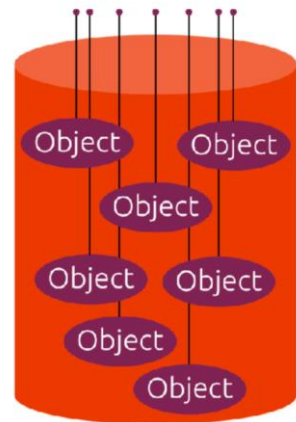
```
from dask_jobqueue import PBSCluster
from distributed import Client
cluster = PBSCluster(project=..,
    queue=.., cores=1, processes=1,
    memory="100GB", walltime=...)
# Ask for 10 nodes
cluster.scale(10)
# OR scale adaptively based on load
cluster.adapt(minimum=1, maximum=100,
    wait_count=60)
# Connect to remote workers
client = Client(cluster)
```

```
from dask_jobqueue import SLURMCluster
from distributed import Client
cluster = SLURMCluster(project=..,
    queue=.., cores=1, processes=1,
    memory="100GB", walltime=...)
# Ask for 10 nodes
cluster.scale(10)
# OR scale adaptively based on load
cluster.adapt(minimum=1, maximum=100,
    wait_count=60)
# Connect to remote workers
client = Client(cluster)
```



Deux outils clés pour ces analyses

ZARR OBJECT STORAGE



Zarr Group: *group_name*

.zgroup

.zattrs

Zarr Array: *array_name*

.zarray

.zattrs

0.0

0.1

1.0

1.1

2.0

2.1

- Bibliothèque open source pour le stockage de tableaux à N-dimensions, compressés et morcelés.
- Créer par Alistar Miles (Imperial) pour la recherche génomique (@alimanfoo); maintenant soutenu par la communauté
- Les tableaux sont divisés en morceaux définis par l'utilisateur; chaque morceau est optionnellement compressé (zlib, zstd, etc.).
- Peut stocker des tableaux en mémoire, des répertoires, des fichiers zip, ou n'importe quelle interface de mappage mutable (dictionnaire)
- Les bibliothèques externes (s3fs, gcsf) fournissent un moyen de stocker ces fichiers dans le cloud.
- Implémentation en Python (référence), C++, Java, Julia
- Futur conteneur des fichiers NetCDF

Les analyses sur le cloud requièrent de basculer vers un stockage objet (Zarr versus NetCDF)



Exemples de tests de performance

Operation	Machine	Ressources	Timing
Opening of netcdf - 1Tb	Personnal Computer	8 cores - 34Gb	30min
	Cluster cal1	20 cores - 62Gb	2min
	HPC hal CNES	48 cores - 240Gb	36mn
	HPC occigen CINES	48 cores - 135Gb	37min
	PANGEO cloud	20 cores - 115Gb	X
Opening of zarr - 1Tb	Personnal Computer	8 cores - 34Gb	8s
	Cluster cal1	20 cores - 62Gb	12s
	HPC hal CNES	48 cores - 240Gb	11s
	HPC occigen CINES	48 cores - 135Gb	10s
	PANGEO cloud	20 cores - 115Gb	8min

L'utilisation du format Zarr accélère également les calculs sur les systèmes HPC



Plan de la présentation

1. Analyse de simulations océaniques
2. Les outils et la communauté PANGEO
3. Exemples d'applications et performances
4. Démonstration live
5. Conclusions et perspectives



Démonstration Live

hub.binder.pangeo.io/user/pangeo-data-pan-torial-agu-2018-opq6wpcm/notebooks/notebooks/ocean_ssh_example.ipynb

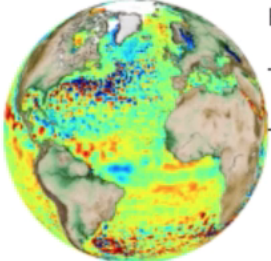
Applications SquirrelMail Getting your Mac... GitHub opendap communauté My CoRe NEMO netCDF_to_zarr.ip... OCS cal1-hub ocean pangeo » Autres favoris

jupyter ocean_ssh_example (auto-sauvegardé) Logout

File Edit View Insert Cell Kernel Widgets Help Sans Confiance Python 3

Exécuter

Sea Surface Altimetry Data Analysis



For this example we will use gridded [sea-surface altimetry data from The Copernicus Marine Environment](#):

This is a widely used dataset in physical oceanography and climate.

The dataset has already been extracted from copernicus and stored in google cloud storage in [xarray-zarr](#) format.

```
Entrée [ ]: %matplotlib inline

import numpy as np
import xarray as xr
import matplotlib.pyplot as plt
import dask.array as dsa

plt.rcParams['figure.figsize'] = (15,10)
```

Initialize Dataset

Here we load the dataset from the zarr store. Note that this very large dataset initializes nearly instantly, and we can see the full list of variables and coordinates.

```
Entrée [ ]: import intake
cat = intake.Catalog("https://raw.githubusercontent.com/pangeo-data/pangeo-datastore/master/logs/ocean.yaml")
ds = cat["sea_surface_height"]
```

QuickTime Player

<https://binder.pangeo.io/v2/gh/pangeo-data/pangeo-tutorial/master>



Plan de la présentation

1. Analyse de simulations océaniques
2. Les outils et la communauté PANGEO
3. Exemples d'applications et performances
4. Démonstration live
5. Conclusions et perspectives



Conclusions

L'analyse des simulations numériques de l'océan en préparation aux observations spatiales nécessite de **repenser nos workflows de post-traitement** (taille, reproductibilité, innovation)

PANGEO offre un **cadre adapté à l'analyse de simulations océaniques à haute résolution**

- Des outils pour manipuler des archives de fichier NetCDF **de l'ordre de 10-100Tb**
- Une utilisation similaire sur **machine locale, cloud et systèmes HPC**
- Des outils qui passent bien à **l'échelle** sur nos applications
- Un bon compromis entre **performance et flexibilité** pour nos applications
- Catalyse la création de **librairies open source** pour l'analyse de données géoscientifiques

Cette utilisation émergente de PANGEO pose également des **questions aux centres HPC**

- Faut-il faciliter l'utilisation d'outil d'**analyses interactif** sur les centres HPC (JupyterHub) ?
- Concilier l'utilisation de "**ressources élastiques**" avec optimisation de l'utilisation des machines ?
- Positionnement et articulation des centres HPC vis-à-vis des **clouds commerciaux** ?