

A. Barth, C. Troupin, S. Watelet &  
J.-M. Beckers

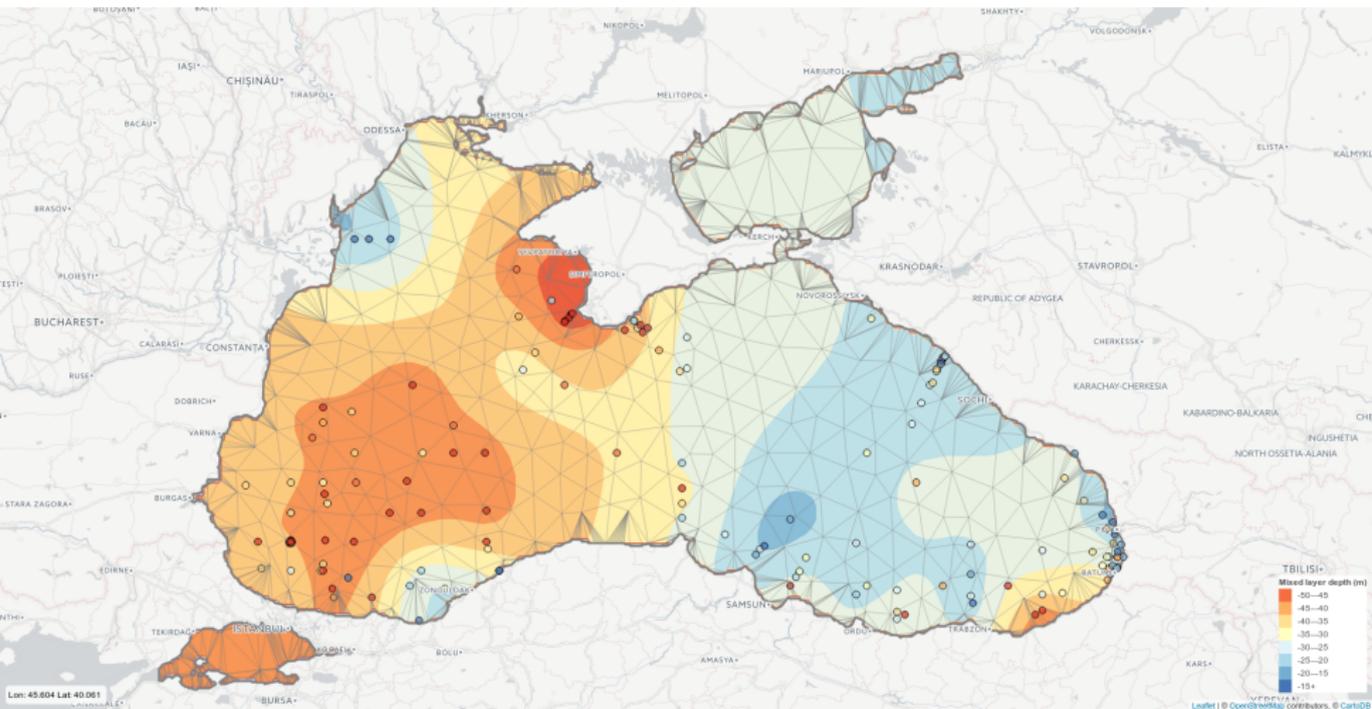
GHER-University of Liège (ULiège)



# *Progress with analysis of DIVA as cloud service*

- 1 divand is one of the tools to generate data products
- 2 divand interpolation to be run  
in notebooks  
managed via jupyterhub
- 3 Web Processing Service interface  
under development

# Diva: from in situ data to gridded fields



 <https://github.com/gher-ulg/DIVA>

 <https://www.geosci-model-dev.net/7/225/2014/gmd-7-225-2014.pdf>

## **divand-1.0: $n$ -dimensional variational data analysis for ocean observations**

A. Barth<sup>1,\*</sup>, J.-M. Beckers<sup>1</sup>, C. Troupin<sup>2</sup>, A. Alvera-Azcárate<sup>1</sup>, and L. Vandenbulcke<sup>3,4</sup>

<sup>1</sup>GHER, University of Liège, Liège, Belgium

<sup>2</sup>IMEDEA, Esporles, Illes Balears, Spain

<sup>3</sup>seamod.ro/Jailoo srl, Sat Valeni, Com. Salatrucu, Jud. Arges, Romania

<sup>4</sup>CIIMAR, University of Porto, Porto, Portugal

\* *Invited contribution by A. Barth, recipient of the EGU Arne Richter Award for Outstanding Young Scientists 2010.*

Correspondence to: A. Barth (a.barth@ulg.ac.be)

Received: 7 June 2013 – Published in Geosci. Model Dev. Discuss.: 23 July 2013

Revised: 18 October 2013 – Accepted: 12 December 2013 – Published: 29 January 2014

2013: Octave/MATLAB

2016: Julia

faster, better, stronger

 <https://github.com/gher-ulg/divand.jl>

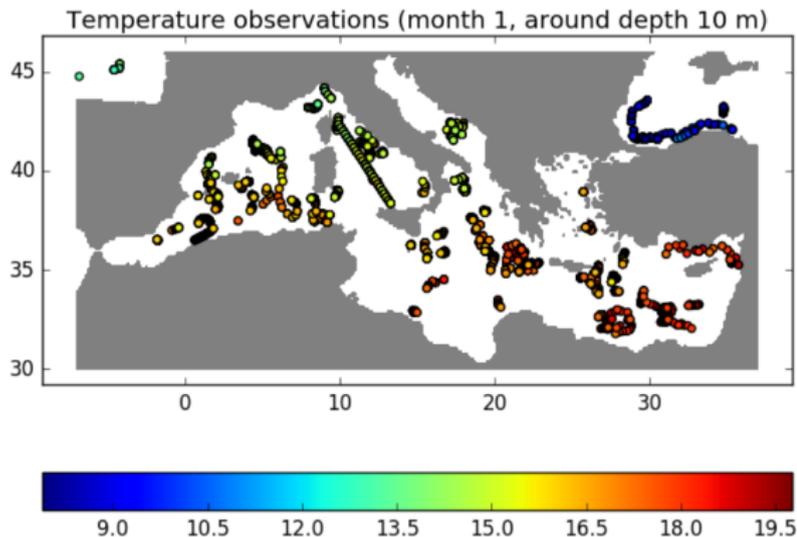
- ▶ Variational inverse method
- ▶ Smoothness and other constraints

## Differences with Diva (2D)

- ▶  $n$ -dimensional,  $n \geq 2$
- ▶ Different kernels
- ▶ Different solvers

## Jupyter-notebooks: a cool way to explain and run tools

```
In [36]: # sets the correct aspect ratio  
gca().set_aspect(1/cos(mean(latr)) * pi/180)
```



Setup the domain using the bathymetry from the file bathname.

[Example online](#)

# *Specifications for the VRE*



## Management of multiple instances of the single-user Jupyter notebook server



The screenshot shows the JupyterHub interface. At the top left is the Jupyter logo. On the right, there are buttons for "Control Panel" and "Logout". Below the logo, there are tabs for "Files", "Running", and "Clusters". A message says "Select items to perform actions on them." To the right of this message are buttons for "Upload", "New", and a refresh icon. The main area is a file browser showing the path: "Projects / SeaDataCloud / Julia". The browser contains a table of files and folders:

	Name	Last Modified
<input type="checkbox"/>	..	seconds ago
<input type="checkbox"/>	data	a month ago
<input type="checkbox"/>	test	4 months ago
<input type="checkbox"/>	DIVAnd+in+Jupyter+Notebook.ipynb	5 months ago
<input type="checkbox"/>	DIVAnd_EUDAT_example_pub.ipynb	5 months ago

 <https://github.com/jupyterhub/jupyterhub>

Demo: <https://hub-test.oceanbrowser.net/> (deployed at CINECA)

Kubernetes (<https://github.com/jupyterhub/kubespawner>)

Kubernetes = open-source system for automating deployment, scaling, and management of containerized applications

+ developed by jupyterhub

Docker swarm (<https://docs.docker.com/engine/swarm/>)

Swarm = cluster of Docker Engines

Current versions of Docker include swarm mode for natively managing a swarm

+ integrated with Docker Engine

# Specifications for the VRE



**Tool**

**jupyterhub + divand**

**OceanBrowser**

---

**Comments:**

# Specifications for the VRE



<b>Tool</b>	<b>jupyterhub + divand</b>	<b>OceanBrowser</b>
No. users	20	100

**Comments:** Based on a typical user workshop

# Specifications for the VRE



<b>Tool</b>	<b>jupyterhub + divand</b>	<b>OceanBrowser</b>
No. users	20	100
RAM	16 GB per user	16 GB (total)
CPU	4 per user	8 (total)
System disk	500 GB per user	about 2 TB in total

**Comments:** For divand: first-guess, conservative estimate

<b>Tool</b>	<b>jupyterhub + divand</b>	<b>OceanBrowser</b>
No. users	20	100
RAM	16 GB per user	16 GB (total)
CPU	4 per user	8 (total)
System disk	500 GB per user	about 2 TB in total
Scalability	20 users simultaneously (1 - 2 weeks)	100 users simultaneously during peaks
Targeted contained	docker container per user	1 docker container

**Comments:** Docker is an elegant solution for deployments

# Specifications for the VRE



Tool	jupyterhub + divand	OceanBrowser
No. users	20	100
RAM	16 GB per user	16 GB (total)
CPU	4 per user	8 (total)
System disk	500 GB per user	about 2 TB in total
Scalability	20 users simultaneously (1 - 2 weeks)	100 users simultaneously during peaks
Targeted contained	docker container per user	1 docker container
Deployment status	demo: <a href="https://hub-test.oceanbrowser.net/">https://hub-test.oceanbrowser.net/</a> (deployed at CINECA)	deployed at CINECA
Reference data services as input	CDI data and user data from B2DROP; output B2DROP and B2SAFE	access to the data products (B2SAFE)

**Comments:** MarineID used for login

Progresses with

*Web Processing  
Service*

# Goal: call divand.jl using WPS



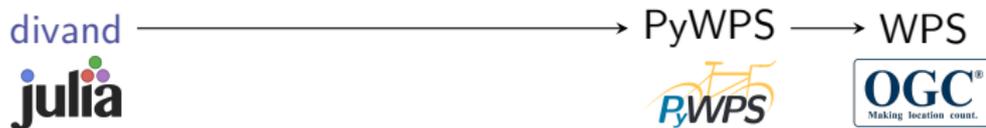
divand  
julia



WPS



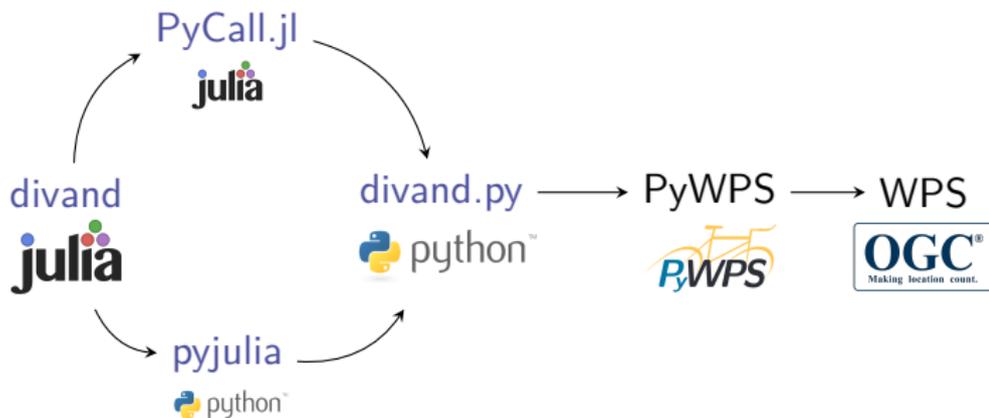
# Goal: call divand.jl using WPS



# Goal: call divand.jl using WPS



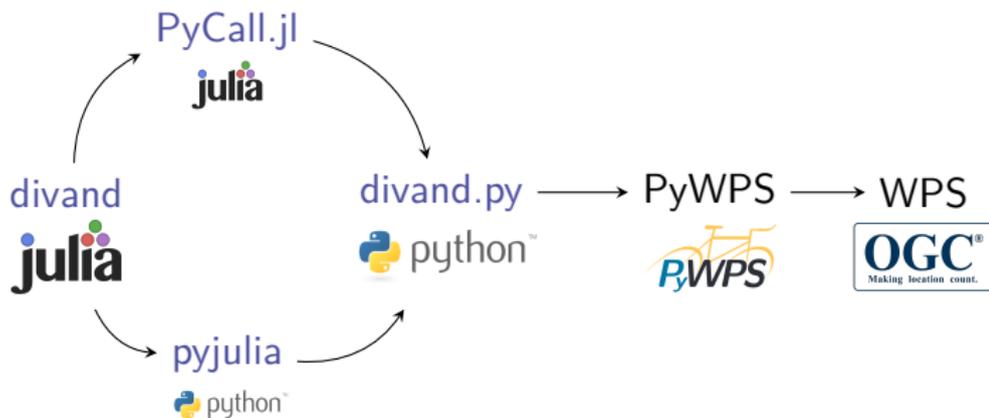
# Goal: call divand.jl using WPS



PyCall: call Python functions from Julia

pyjulia: call Julia in Python

# Goal: call divand.jl using WPS



`PyCall`: call Python functions from Julia

`pyjulia`: call Julia in Python

More tools, more problems...

- 1 Demo version using DIVA 2D basic tools
- 2 Basic implementation (local) with a test case

*New climatologies  
and products*

## Spatial resolution:

Grid resolution  $\neq$  real resolution!!

1 divand:  $1/8^\circ \times 1/8^\circ$

2 post-processing:  $1/24^\circ \times 1/24^\circ$

based on data availability

match model resolution

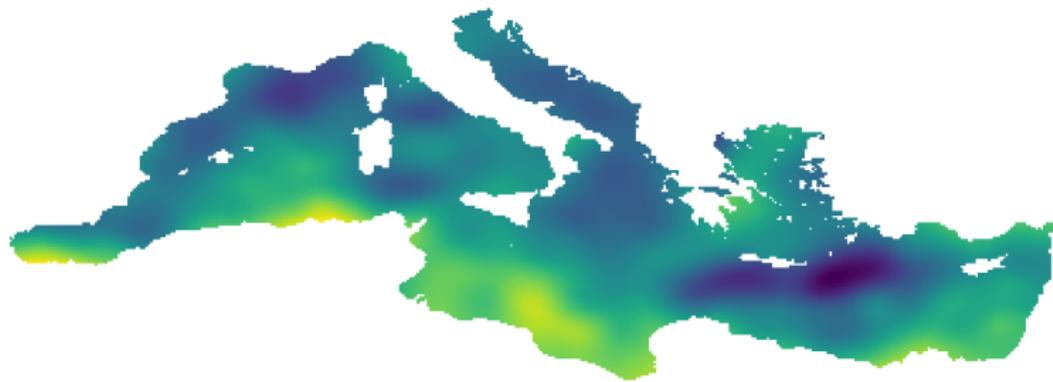
## Depth levels:

- ▶ Common to all the products
  - ▶ Follow IODE levels
  - ▶ Consider World Ocean Atlas
- allows combined product  
33 levels from 0 to 5500 m  
103 levels!

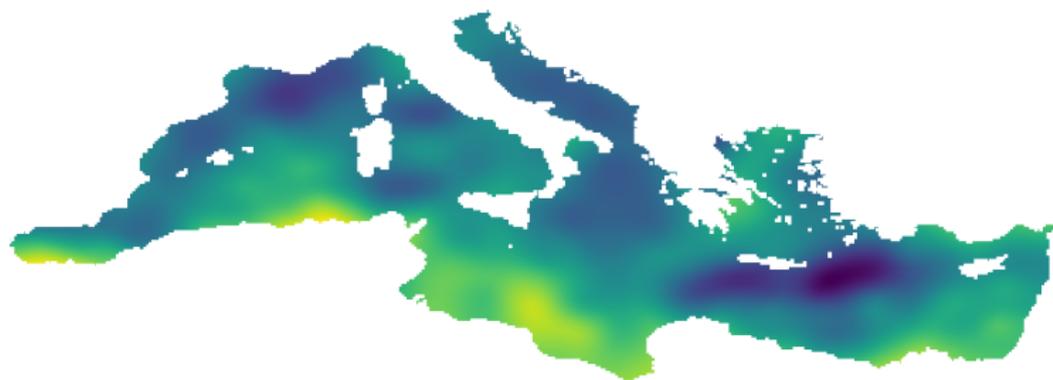
# Find the differences between Product 1 & Product 2



Product 1



Product 2



# Find the differences between Product 1 & Product 2



- 1 Field 1 is  $161 \times 426$   
Field 2 is  $641 \times 1701$

# Find the differences between Product 1 & Product 2



- 1 Field 1 is  $161 \times 426$   
Field 2 is  $641 \times 1701$
- 2 File 2 is 30 times larger





- nco = netCDF Operators
- = set of standalone programs to manipulate netCDF files
- renaming, averaging, **re-gridding**, binary operations...

<http://nco.sourceforge.net/>

Re-gridding based on Earth System Modeling Framework (ESMF, <https://www.earthsystemcog.org/projects/esmf/>)

Usage:

```
ncremap -i data.nc -d grid.nc -o output.nc
```

where:

`data.nc` = original netCDF containing field

`grid.nc` = file containing the new (finer) grid

`output.nc` = new netCDF with field interpolated  
onto the new grid