



**Data Analysis made easy with the ENES Climate Analytics Service (ECAS)**

**Sofiane Bendoukha, Donatello Elia, Fabrizio Antonio, Sandro Fiore, Tobias Weigel, Alessandro D'Anca**

**April 10, 2019 @ EGU 19  
SC1.22**

---

 [eosc-hub.eu](https://eosc-hub.eu)

 [@EOSC\\_eu](https://twitter.com/EOSC_eu)

**Dissemination level: Public**



### ● Training materials

- <https://github.com/ECAS-Lab/ecas-training>

### ● ECASLab / JupyterHub

- **ECASLab @ DKRZ** <https://ecaslaboratory.dkrz.de>
- **ECASLab @ CMCC** <https://ecaslaboratory.cmcc.it>

### ● Ophidia framework documentation

- <http://ophidia.cmcc.it/documentation/users/index.html>

## ● ECAS is part of the **EOSC-HUB** service catalogue

- ECAS enables scientific end-users to perform data analysis experiments

## ● Server-based

- Computation @ **CMCC** or **DKRZ**
- Avoid data transfer (download)
- Improved reusability of data and workflows (FAIR approach)

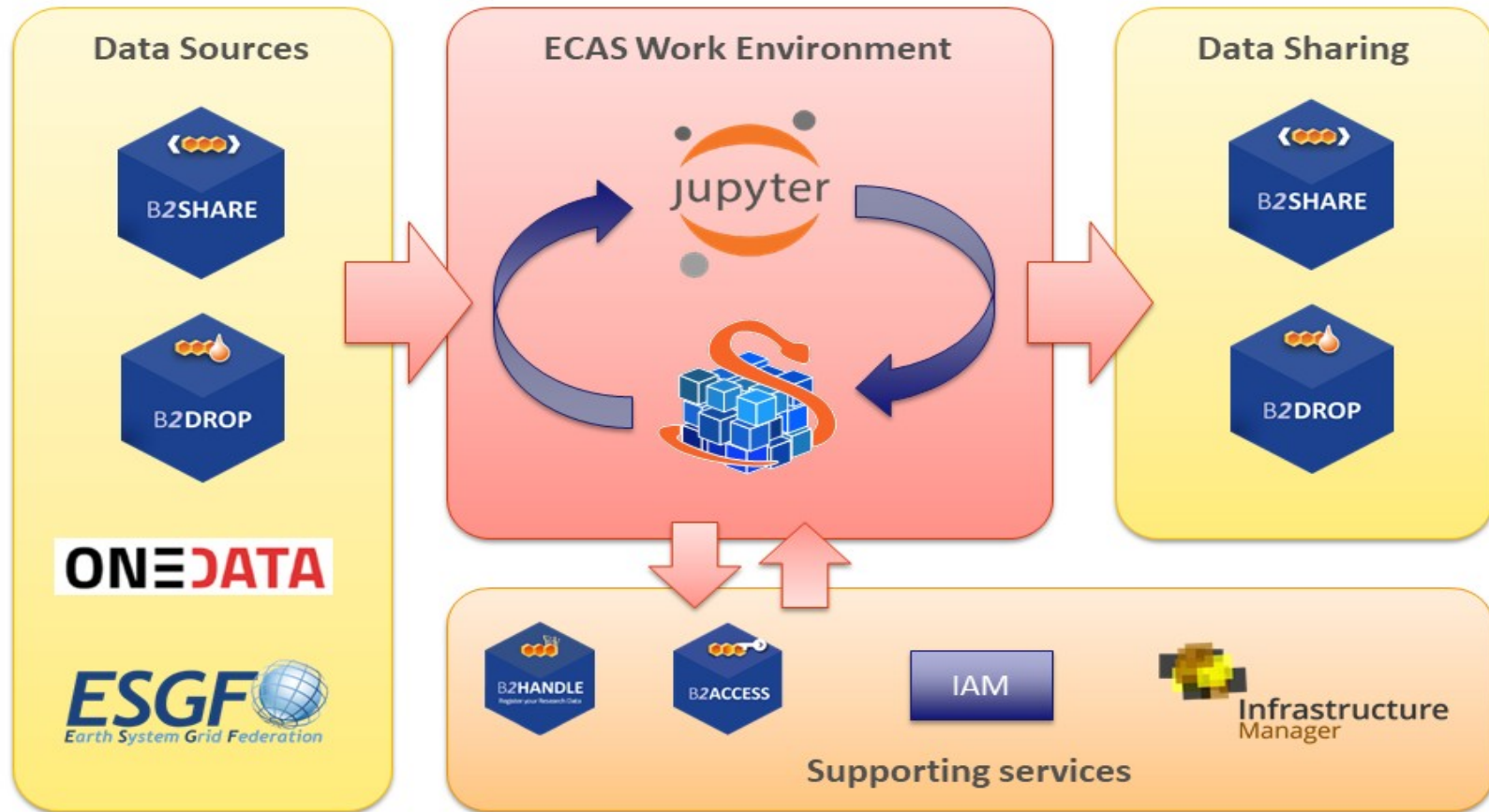


## ● ECAS supports different Auth\* providers

- Local and external AAI providers supported (LDAP, B2ACCESS, EGI Check In)
- Additional AAI providers can be integrated (e.g. INDIGO IAM)

- ECAS provides data access via ESGF
- Coordinated Regional Climate Downscaling Experiment
  - ~ 100 Tbyte Cordex
- Coupled Model Intercomparison Project 5
  - ~ 1.2 Pbyte CMIP5 Data
- Coupled Model Intercomparison Project 6
  - ~ 250 Tbyte CMIP6 Data from the 1PByte published
- Other Data pools can be mounted on demand
  - MPI Grand ensemble (**MPI - GE**)
  - data collection exposed in the Federated Data Archive (e.g. through **OneData**)

## Service architecture and interfaces

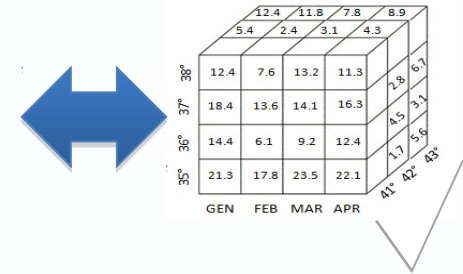
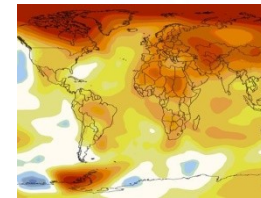
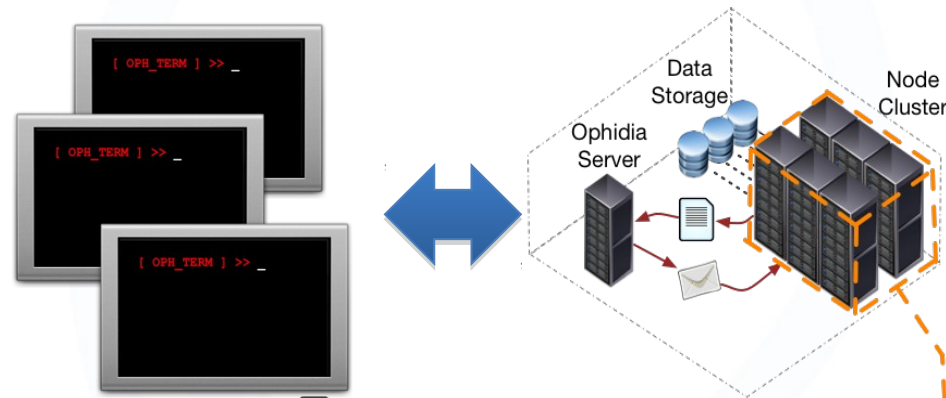


## ● The Ophidia framework addresses big data challenges for eScience

- support for declarative, parallel, server-side data analysis exploiting parallel computing techniques
- end-to-end mechanisms to support complex experiments and large processing workflows on scientific multi-dimensional datacubes

## ● Ophidia supports both **batch** & **interactive** data analytics

- More than 50 datacube-oriented **operators** are available, including: data reduction and subsetting, data intercomparison, metadata and provenance management, time series analysis with array-based primitives
- A wide set of (low-level) array-based primitives (over 100) to perform, e.g. data summarization, algebraic expressions, predicates evaluation, statistical analysis
- Support for complex workflows and Python applications execution

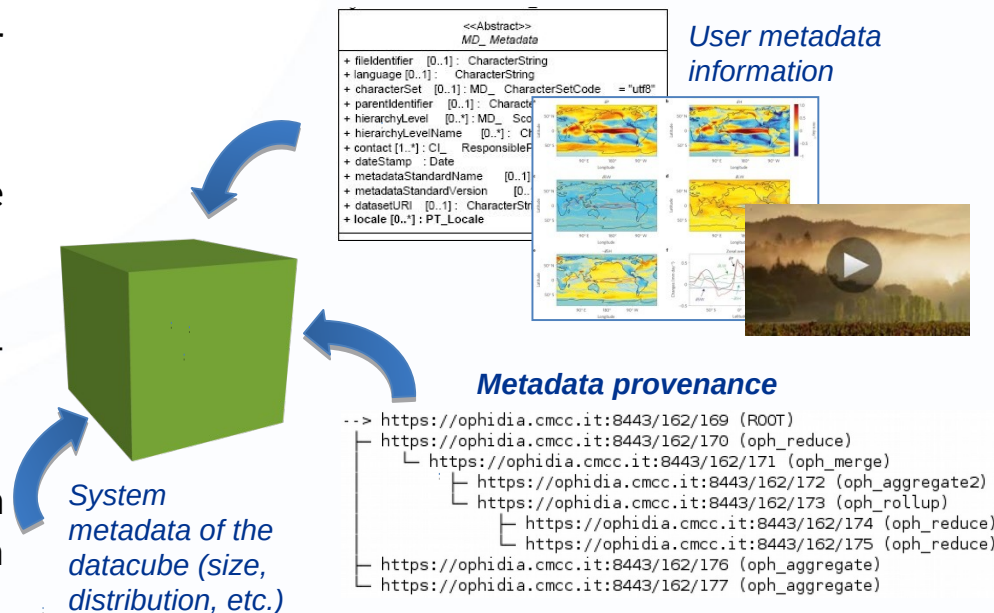


**Oph\_Term:** a terminal-like commands interpreter serving as a client for the Ophidia framework

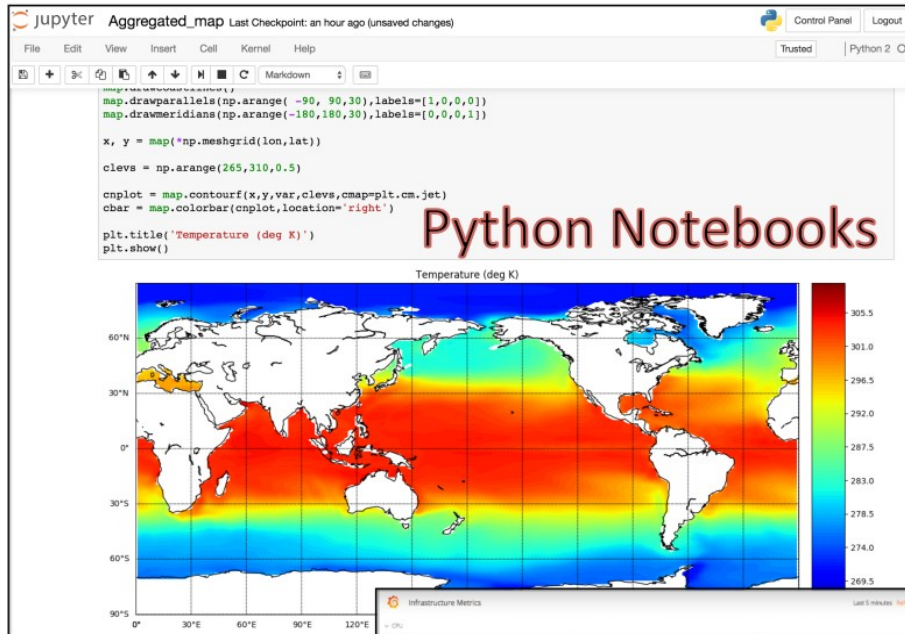
**PyOphidia:** a Python interface for datacube management & analytics with Ophidia

**Ophidia framework:** declarative, parallel server-side processing

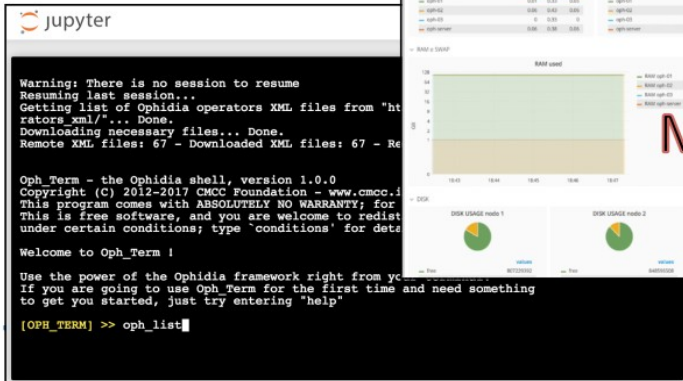
Through **oph\_term/PyOphidia** the user run ("send") commands ("operators") to the Ophidia framework to manipulate datasets ("datacubes")







## ECAS Terminal



```

Warning: There is no session to resume
Resuming last session...
Getting list of Ophidia operators XML files from "ht
rators.xml"... Done.
Downloading necessary files... Done.
Remote XML files: 67 - Downloaded XML files: 67 - Re

```

**Monitoring**



**Quick Start**

OphidiaLab provides two different ways to get access to its scientific eco-system: JupyterHub and Ophidia client.

Jupyter supports interactive data science and scientific computing. OphidiaLab includes a JupyterHub installation and, thanks to the Jupyter Notebooks, scientists can create and share documents that contain live code, equations, visualizations and explanatory text.

The JupyterHub interface is available [here](#).\*

After you login, open "Quick Start.ipynb" notebook available under the *quickstart/* folder in your home to get started with OphidiaLab environment capabilities.

\*Please note that for security reasons, the access to our JupyterHub instance is restricted to authorised users only and needs an additional step after the registration process.

**QuickStart**

The Ophidia Terminal is a robust, comprehensive, and user-friendly Ophidia client, developed with characteristics similar to the bash shell present in almost all Unix-like environments. Please have a look at the online available documentation to learn more about the basic functionalities of the [Ophidia terminal](#) as well as some [advanced features](#) useful for more skilled users.

Two short guides ([basic](#), [advanced](#)) in pdf format are also available.

Several examples of real-world usage of the terminal are also available on the Ophidia website tutorial section. The latest client RPM for CentOS is available [here](#). The related DEB package can be downloaded from [here](#).

Once installed you can simply run:

```

/usr/local/ophidia/oph-terminal/bin/oph_term -H
ophidiab.cmcc.it -u <username> -p <password> -P 11732

```



# Thank you for your attention!

---

*Questions?*

Sofiane Bendoukha [bendoukha@dkrz.de](mailto:bendoukha@dkrz.de)  
Donatello Elia [donatello.elia@cmcc.it](mailto:donatello.elia@cmcc.it)  
Fabrizio Antonio [fabrizio.antonio@cmcc.it](mailto:fabrizio.antonio@cmcc.it)



## **EOSC-hub**

 [eosc-hub.eu](http://eosc-hub.eu)  [@EOSC\\_eu](https://twitter.com/EOSC_eu)



This material by Parties of the EOSC-hub Consortium is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

### ● Training materials

- <https://github.com/ECAS-Lab/ecas-training>

### ● ECASLab / JupyterHUB

- **ECASLab @ DKRZ** <https://ecaslaboratory.dkrz.de>
- **ECASLab @ CMCC** <https://ecaslaboratory.cmcc.it>

### ● Ophidia framework documentation

- <http://ophidia.cmcc.it/documentation/users/index.html>