High-Perfomance Data Analytics in eScience

D. Elia[1,2], **S. Fiore**[1]

[1] Fondazione Centro Euro-Mediterraneo sui Cambiamenti Climatici (CMCC), Lecce, Italy
[2] University of Salento, Lecce, Italy

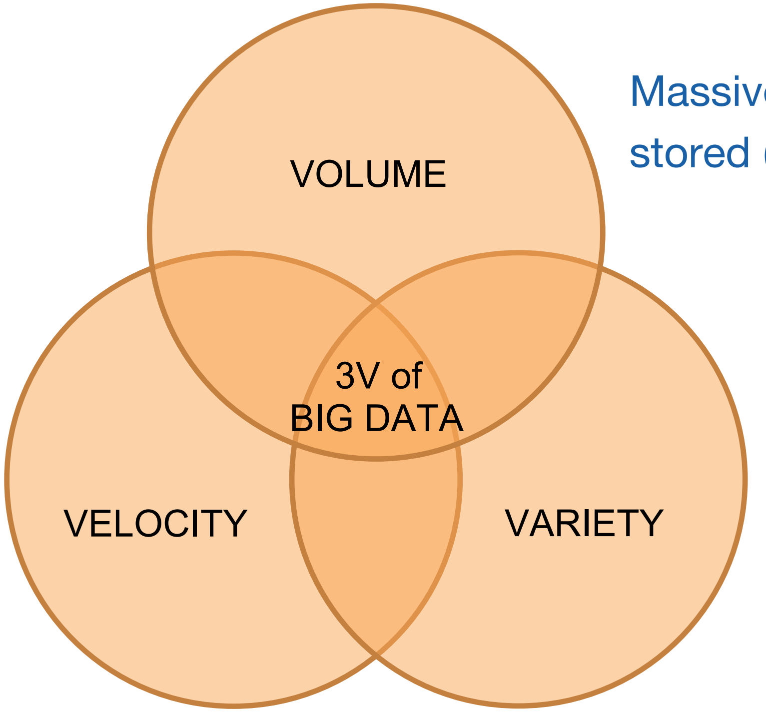
ESiWACE2 Summer School on Effective HPC for Climate and Weather
26 August 2020

Session outline

- ✔ *Introduction to HPDA and data challenges in eScience*
- ✔ *ECAS and EOSC*
- ✔ *Introduction to the Ophidia HPDA Framework*
- ✔ *Ophidia core concepts: architecture, data model, operators and primitives*
- ✔ *Analytics workflows with Ophidia*
  - ✔ *Workflow execution demo*
- ✔ *Ophidia Python bindings: PyOphidia*

3V's of Big Data

Back in 2001 D. Laney[1] described data management challenges according to 3 dimensions: the well-known 3V's model which has been then used to characterize big data.



Massive data size produced and stored (TB, PB, …)

Speed at which data are generated and need to be acquired and analyzed (e.g., real-time, near real-time)

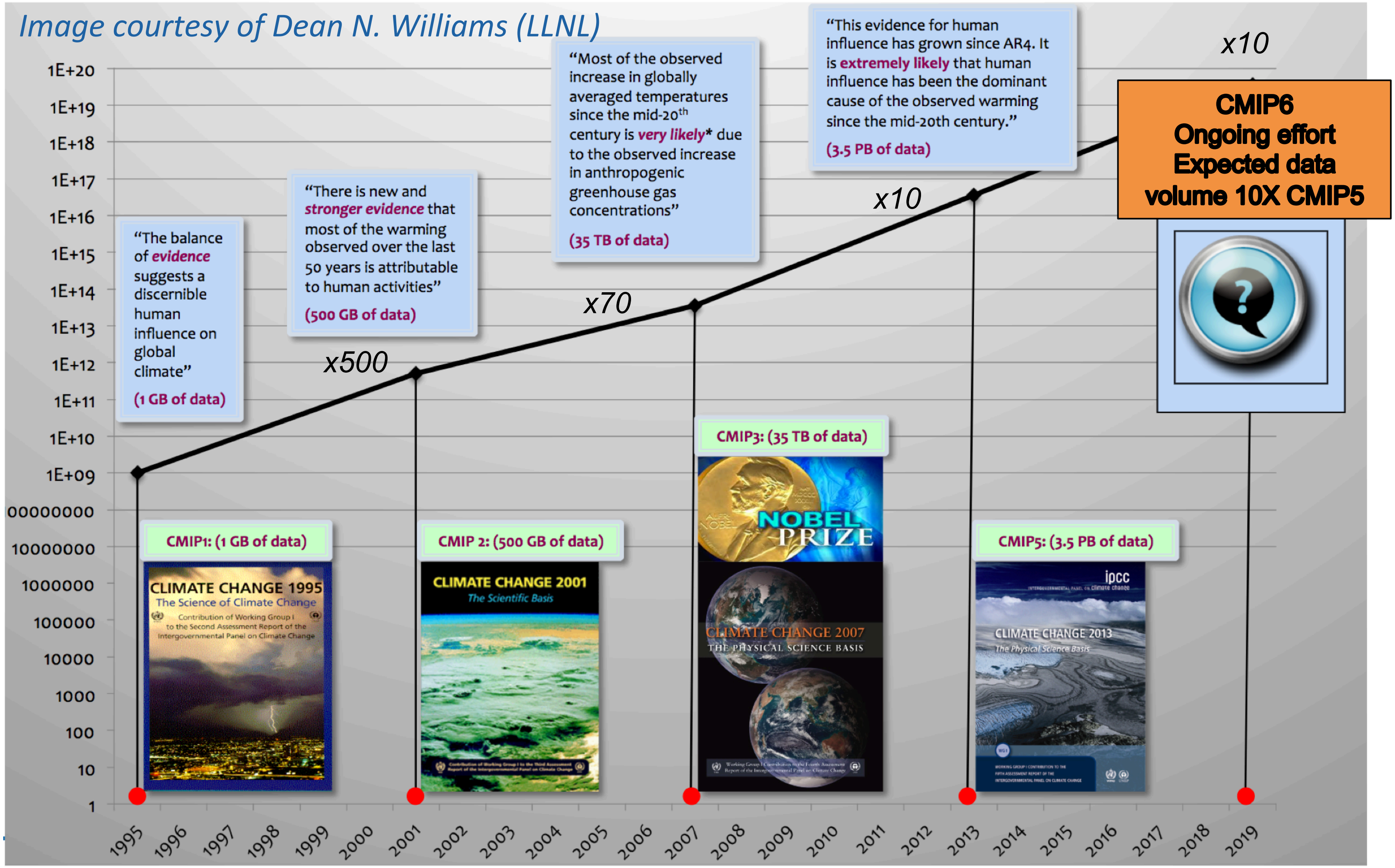
Multiple heterogeneous formats from diverse sources:

- Structured
- Semi-structured
- Unstructured

Other V's have been later identified: e.g., value, veracity, …

[1]*Laney D (2001) 3-d data management: controlling data volume, velocity and variety. META Group Research Note, 6 February*

CMIP data evolution



Data Analytics and HPC ecosystem

(Big) Data analytics ecosystem

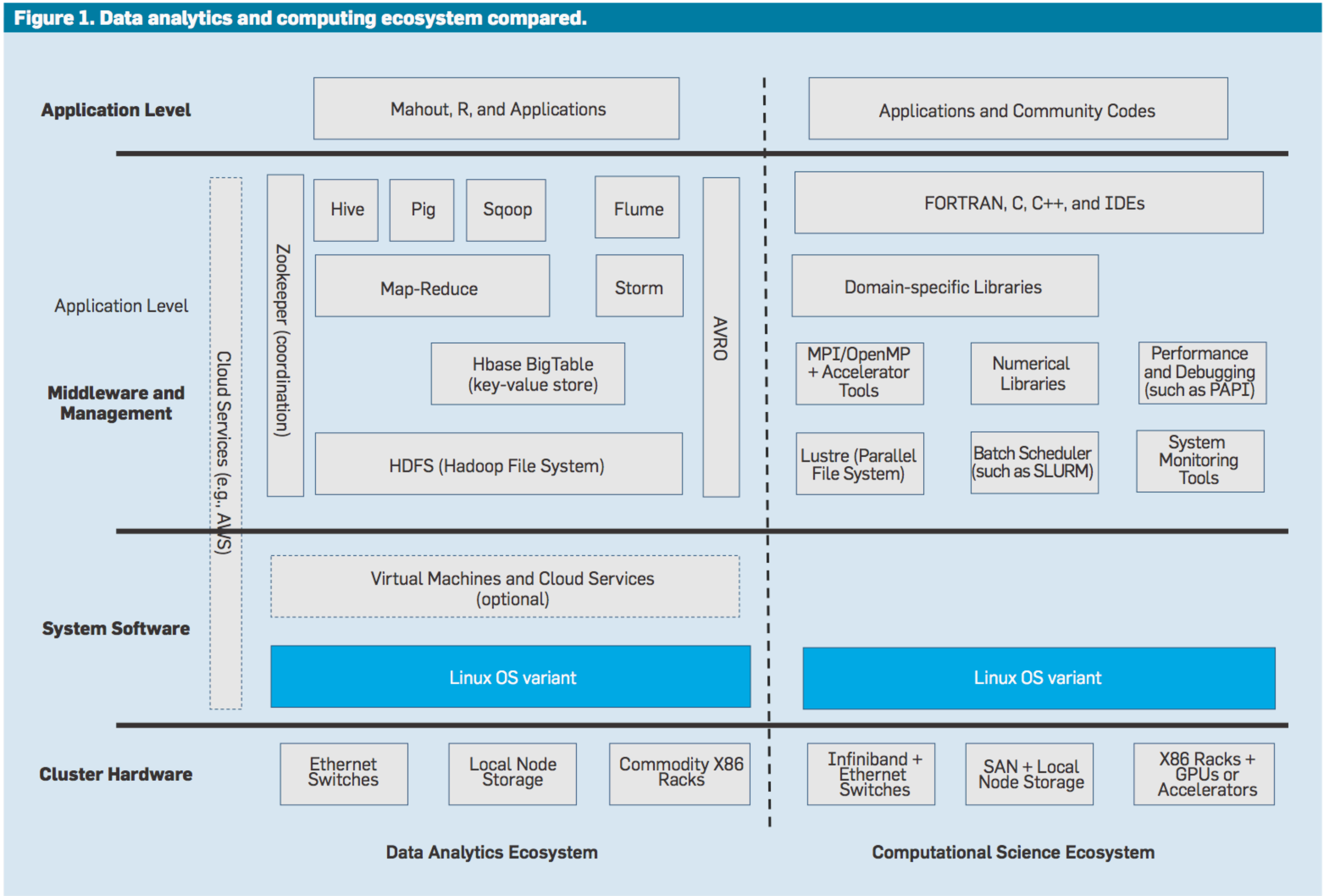
- Commodity hardware
- Shared-nothing architecture
- Dynamic resource allocation
- Heterogeneous workloads
- MapReduce computing paradigm
- High-level programming abstractions

HPC (Scientific computing) ecosystem

- High-end hardware
- Shared-disk architecture
- Fixed resource allocation
- Large batch workloads
- MPI+X -based computing paradigm
- C/Fortran code

References:
-D. A. Reed and J. Dongarra. 2015. Exascale computing and big data. Commun. ACM 58, 7 (July 2015), 56–68.
-Jha, S., Qiu, J., Luckow, A., Mantha, P., & Fox, G. C. (2014). A tale of two data-intensive paradigms: Applications, abstractions, and architectures. In 2014 IEEE Int. Congress on Big Data (pp. 645-652).
-Asch, M., et al. (2018). Big data and extreme-scale computing: Pathways to convergence-toward a shaping strategy for a future software and data ecosystem for scientific inquiry. Int. J. High Perform. Comput. Appl., 32(4), 435-479.

Data Analytics and computing ecosystem



Source: Daniel A. Reed and Jack Dongarra. 2015. Exascale computing and big data. Commun. ACM 58, 7 (July 2015), 56–68.

Convergenge of data analytics and HPC in eScience

Convergence of data-intensive analytics and HPC:

- *(Big) Data analytics ecosystem has rapidly expanded in the last 15 years, leading to a wide spectrum of new solutions, mainly outside the scientific and engineering community*

- *HPC solutions have been used for several years in different scientific fields for scientific computing (simulations and modeling)*

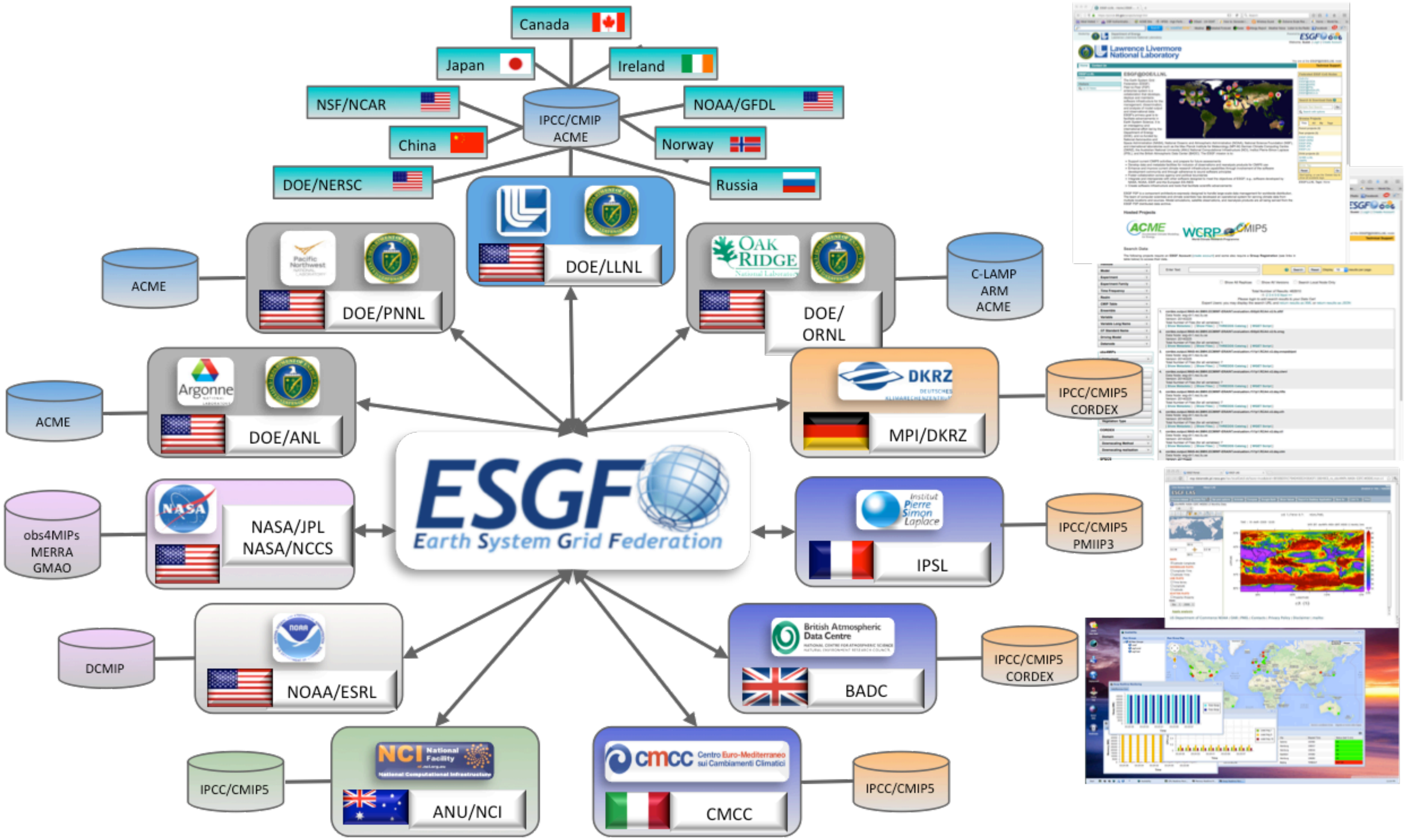
- *Computational science modeling and data analytics are both crucial in scientific research*

- *The convergence of the solutions and technology of the two ecosystems is a key factor for accelerating scientific discovery*

High-Performance Data Analytics (HPDA)

ESGF and the CMIP data archive

ESGF[1] is a coordinated multiagency, international collaboration of institutions that continually develop, deploy, and maintain software needed to facilitate and empower the study of climate.



[1]L. Cinquini, et al. (2014). The Earth System Grid Federation: An open infrastructure for access to distributed geospatial data. Future Gener. Comput. Syst. 36: 400-417.

Climate analysis challenges & issues

Several key challenges and practical issues related to large-scale climate analysis

- Setup of a data analysis experiment requires the *download of input data* (also from multiple models)
  - *Data download* is a *big barrier* for climate scientists
- The complexity of the data analysis process leads to the need for *end-to-end workflow support* solutions
  - *Data analysis* mainly performed using *client-side* approaches
  - Analysing large datasets involves *running tens/hundreds of analytics operators*
  - *Installation* and update of data analysis *tools and libraries* needed
- Large data volumes pose strong *requirements* in terms of *computational and storage* resources

New approaches for climate analysis at scale

Dedicated data intensive facilities close to the different storage hierarchies will be needed to address high-performance scientific data management & analytics

Server-side approaches will intrinsically and drastically *reduce data movement*

- download will only relate to the final results of an analysis
- they will foster re-use as well as collaborative experiments
- need for efforts toward highly interoperable tools/envs for data analysis

Workflow supports the definition and management of *complex data-intensive analytics* applications

Higher-level programming approaches for data analytics are required to effectively exploit the resources and improve productivity

- Cultural change must be faced to encourage the adoption of novel solutions and tools

Session outline

- ✔ ***Introduction to HPDA and data challenges in eScience***
- ✔ ***ECAS and EOSC***
- ✔ ***Introduction to the Ophidia HPDA Framework***
- ✔ ***Ophidia core concepts: architecture, data model, operators and primitives***
- ✔ ***Analytics workflows with Ophidia***
    - ✔ ***Workflow execution demo***
- ✔ ***Ophidia Python bindings: PyOphidia***

The European Open Science Cloud (EOSC)

The **European Open Science Cloud (EOSC)** is an ambitious program that will offer a **virtual environment** with **open** and **seamless services** for storage, management, **analysis** and **re-use of research data**, **across borders** and **scientifc disciplines** by federating existing scientifc data infrastructures, currently dispersed across disciplines and Member States.

This programme will deliver an **Open Data Science Environment** that **federates existing scientific data infrastructures** to offer European science and technology researchers and practitioners seamless access to services for storage, management, analysis and re-use of research data presently restricted by geographic borders and scientific disciplines.

EOSC-hub is a key infrastructural project in the **EOSC** landscape

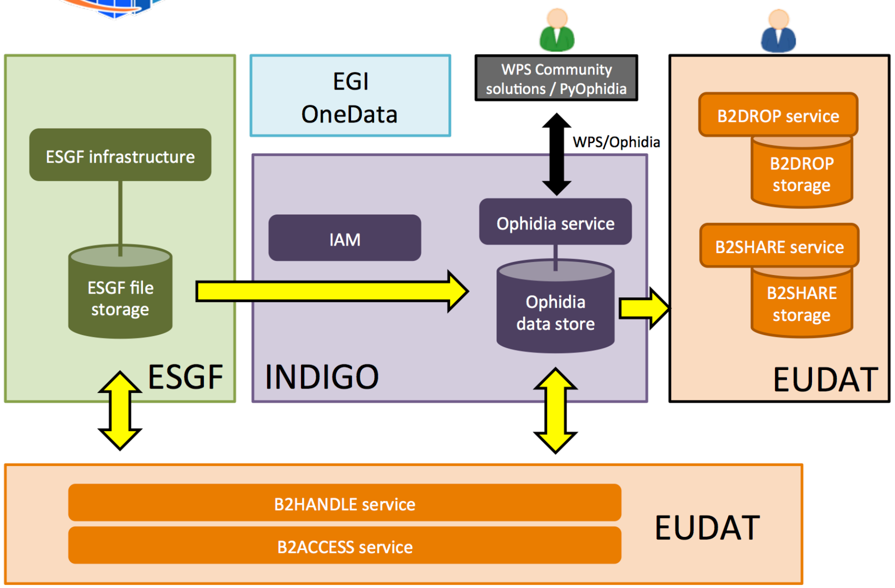
About EOSC: https://www.eosc-portal.eu/about/eosc

The ENES Climate Analytics Service (ECAS)

The **ENES Climate Analytics Service (ECAS)**, proposed by CMCC & DKRZ in EOSC-hub supports climate data analysis

It is one of the *EOSC-Hub Thematic Services*: https://www.eosc-hub.eu/services/ENES%20Climate%20Analytics%20Service

ECAS builds on top of the *Ophidia big data analytics framework* with components from INDIGO-DataCloud, EUDAT and EGI



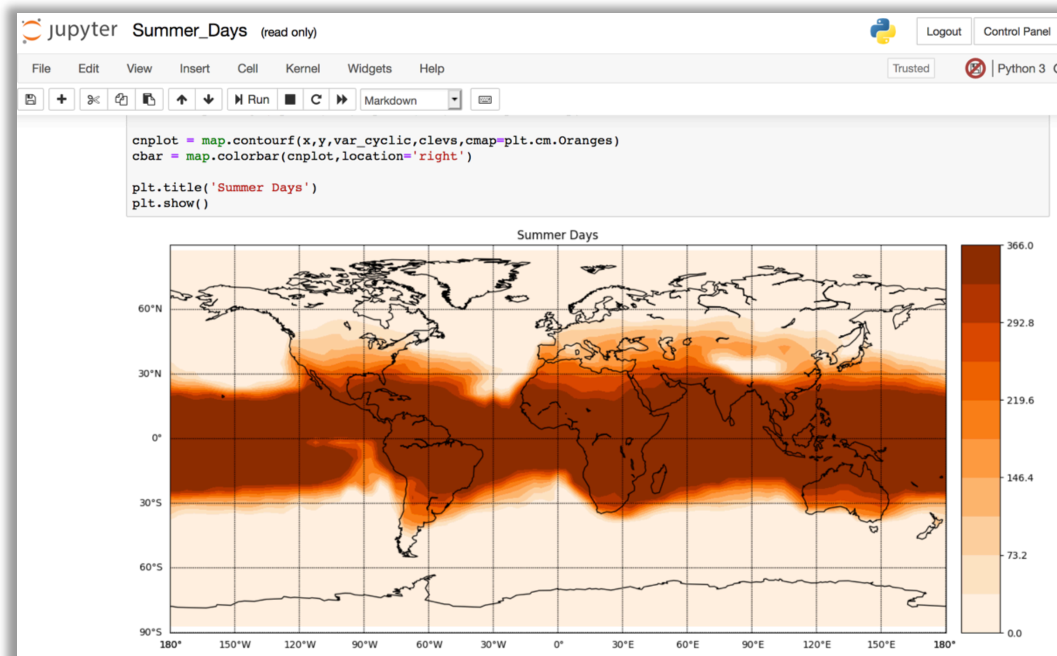
The European Commission launched the European Open ScienceCloud Initiative to capitalise on the data revolution. EOSC will provide European science, industry and public authorities with world-class digital infrastructure that bring state of the art computing and data storage capacity to the fingertips of any scientists and engineer in the EU.

EOSC-hub receives funding from the EU's Horizon 2020 research and innovation programme under grant agreement No. 777536.

ECASLab: a Python environment for data analysis

ECASLab provides a user-friendly environment for scientific analysis based on:

- The *ECAS* integrated service
- A *JupyterHub* instance providing a graphical environment for user's experiments
- Bundled with a wide set of *Python scientific modules* for data manipulation, analysis and visualization, such as PyOphidia, NumPy, Pandas, Dask, Matplotlib, basemap, Cartopy
- A set of ECAS usage example notebooks (https://github.com/ECAS-Lab/ecas-notebooks)



Two major instances are hosted by:

- **CMCC** https://ecaslab.cmcc.it
- **DKRZ** https://ecaslab.dkrz.de

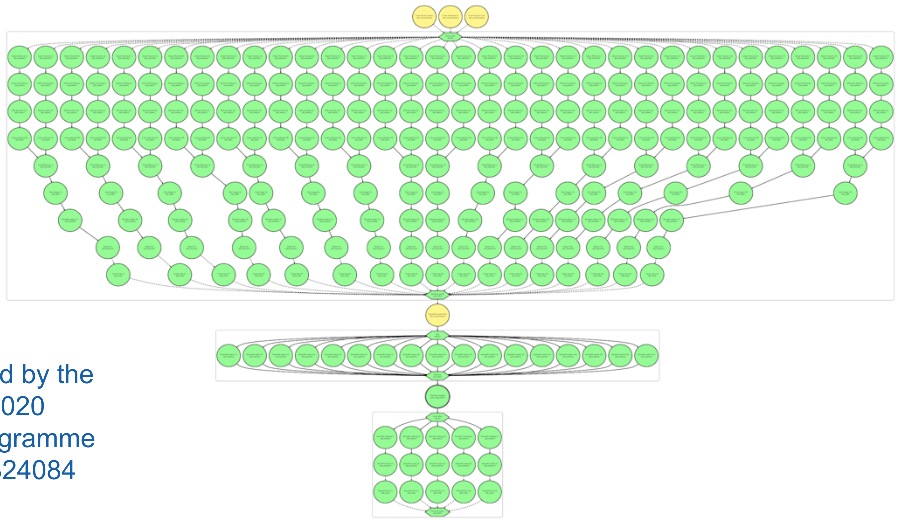
A complete environment for climate experiments

ECASLab is a complete environment for supporting scientist in their daily research activities with a focus on those from the climate change domain

- It represents a single entrypoint to *analysis tools, scientific datasets* (e.g., from ESGF data archive) and *computing resources*

- It provides the capabilities for the implementation and execution of both interactive and complex experiments (workflows), such as *multi-model CMIP-based data analysis*[1]

ECAS is also one of the *compute services* made available to climate scientists by the *EU H2020 IS-ENES3 project*



is-enes
INFRASTRUCTURE FOR THE EUROPEAN NETWORK FOR EARTH SYSTEM MODELLING

IS-ENES3 is a project funded by the European Union's Horizon 2020 research and innovation programme under grant agreement No 824084

[1]S. Fiore, D. Elia, C. Palazzo, A. D'Anca, F. Antonio, D. N. Williams, I. Foster, G. Aloisio, "Towards an Open (Data) Science Analytics-Hub for Reproducible multi-model Climate Analysis at Scale", *2018 IEEE Int. Conference on Big Data*

Session outline

- ✔ *Introduction to HPDA and data challenges in eScience*
- ✔ *ECAS and EOSC*
- ✔ *Introduction to the Ophidia HPDA Framework*
- ✔ *Ophidia core concepts: architecture, data model, operators and primitives*
- ✔ *Analytics workflows with Ophidia*
  - ✔ *Workflow execution demo*
- ✔ *Ophidia Python bindings: PyOphidia*

Ophidia High-Performance Data Analytics Framework

Ophidia (http://ophidia.cmcc.it) is a CMCC Foundation research project addressing data challenges for eScience

It provides:

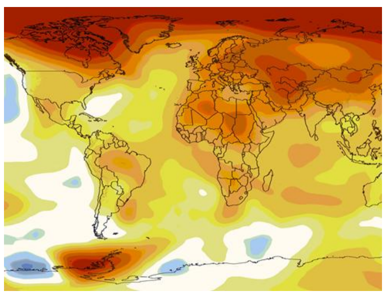
- a *High-Performance Data Analytics* (HPDA) framework for multi-dimensional scientific data joining HPC paradigms with scientific data analytics approaches
- in-memory and server-side data analysis exploiting parallel computing techniques and database approaches
- a multi-dimensional, array-based, storage model and partitioning schema for scientific data leveraging the datacube abstraction
- end-to-end mechanisms to support complex experiments and large workflows on scientific datacubes, primarily in climate domain

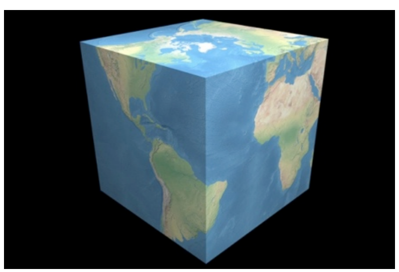
Ophidia

cmcc
Centro Euro-Mediterraneo sui Cambiamenti Climatici

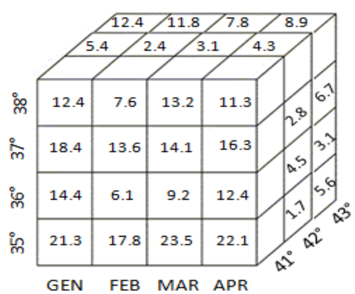
A paradigm shift

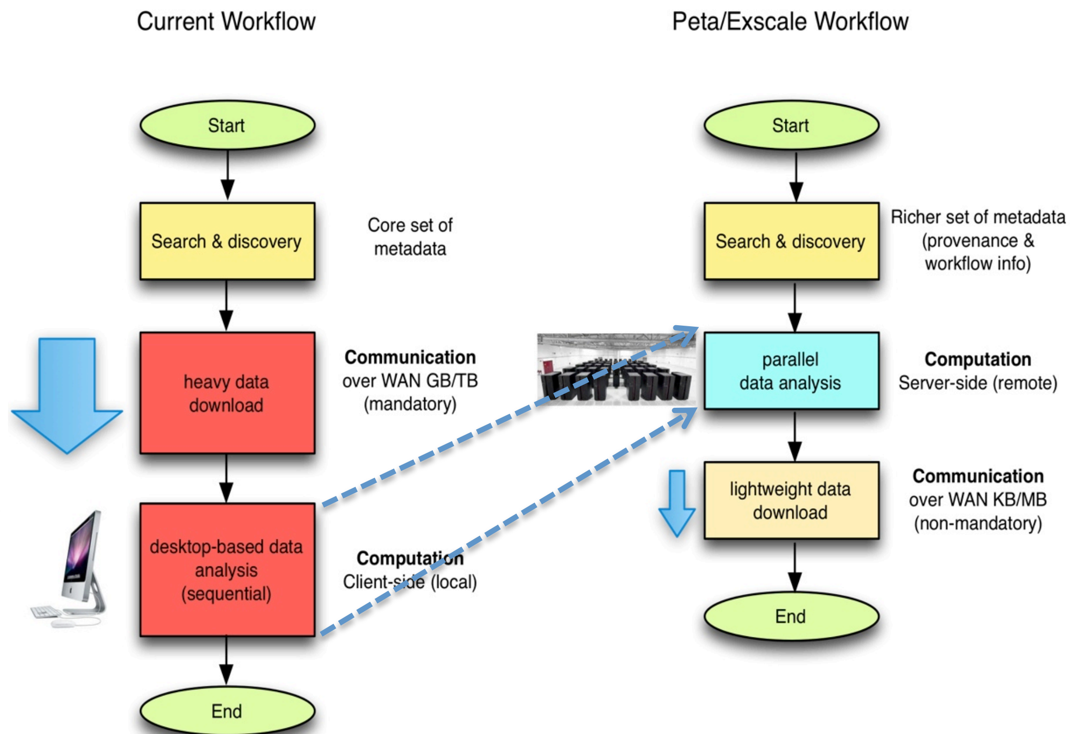
Volume, variety, velocity are key challenges for big data in general and for climate change science in particular. Client-side, sequential and disk-based workflows are three limiting factors for the current scientific data analysis tools.





Earth System Grid Federation
ESGF





S. Fiore, A. D'Anca, C. Palazzo, I. Foster, D. N. Williams, G. Aloisio, "Ophidia: toward bigdata analytics for eScience", ICCS2013 Conference, Procedia Elsevier, Barcelona, June 5-7, 2013

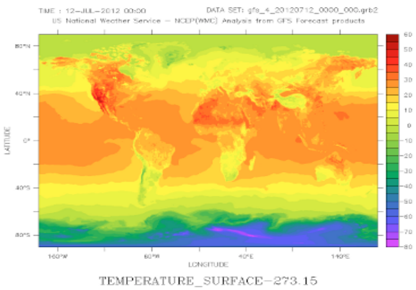
Data analytics requirements and use cases

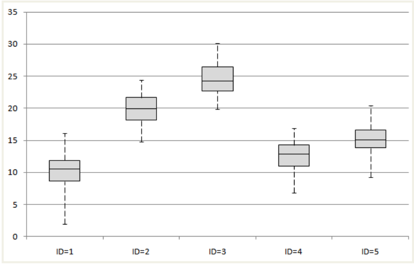
Requirements and needs focus on:

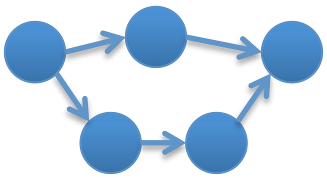
- *Time series analysis*
- *Data subsetting*
- *Model intercomparison*
- *Multimodel means*
- *Massive data reduction*
- *Data transformation (through array-based primitives)*
- *Param. Sweep experiments (same task applied on a set of data)*
- *Maps generation*
- *Ensemble analysis*
- *Data analytics workflow support*

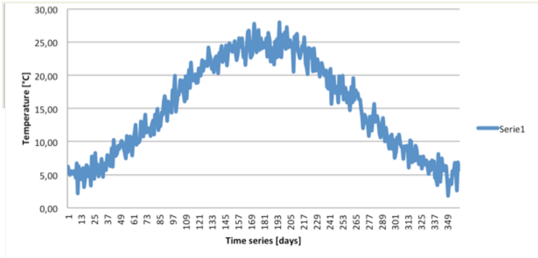
But also…

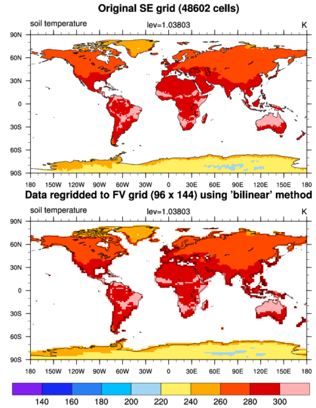
- *Performance*
- *re-usability*
- *extensibility*

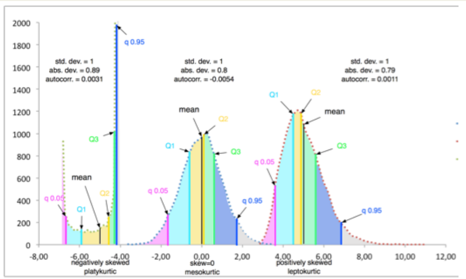




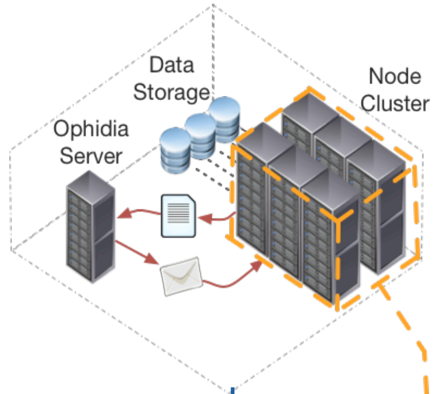


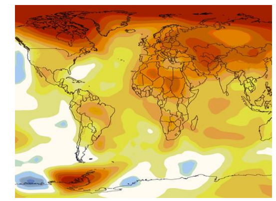


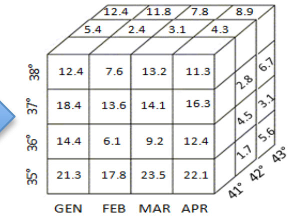




Server-side paradigm and the datacube abstraction







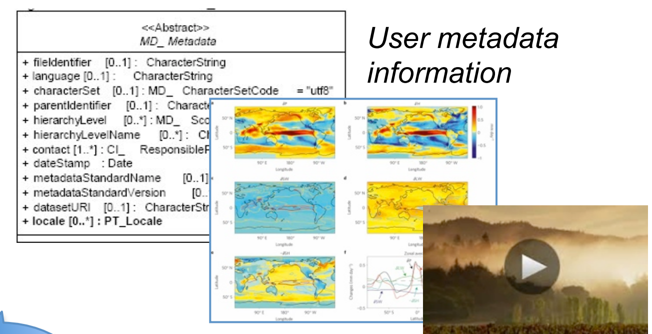
Oph_Term: a terminal-like commands interpreter serving as a client for the Ophidia framework

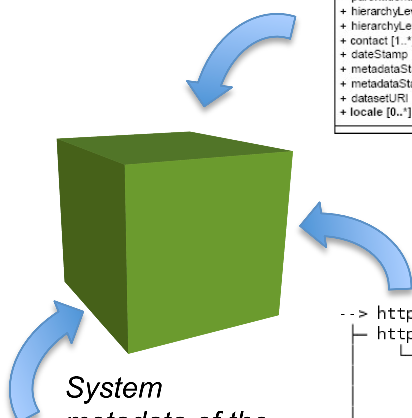
PyOphidia: a Python interface for datacube management & analytics with Ophidia

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")

User metadata information





System metadata of the datacube (size, distribution, etc.)

Metadata provenance

```
--> https://ophidia.cmcc.it:8443/162/169 (ROOT)
 ├ https://ophidia.cmcc.it:8443/162/170 (oph_reduce)
 │  └ https://ophidia.cmcc.it:8443/162/171 (oph_merge)
 │       ├ https://ophidia.cmcc.it:8443/162/172 (oph_aggregate2)
 │       └ https://ophidia.cmcc.it:8443/162/173 (oph_rollup)
 │            ├ https://ophidia.cmcc.it:8443/162/174 (oph_reduce)
 │            └ https://ophidia.cmcc.it:8443/162/175 (oph_reduce)
 ├ https://ophidia.cmcc.it:8443/162/176 (oph_aggregate)
 └ https://ophidia.cmcc.it:8443/162/177 (oph_aggregate)
```

Some international projects exploiting Ophidia

esiwace
CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER AND CLIMATE IN EUROPE

is-enes
INFRASTRUCTURE FOR THE EUROPEAN NETWORK FOR EARTH SYSTEM MODELLING

EOSC-hub

EUBRA BIGSEA
EUROPE - BRAZIL COLLABORATION OF BIG DATA SCIENTIFIC RESEARCH THROUGH CLOUD-CENTRIC APPLICATIONS

ofidia

clipc
Climate Information Portal

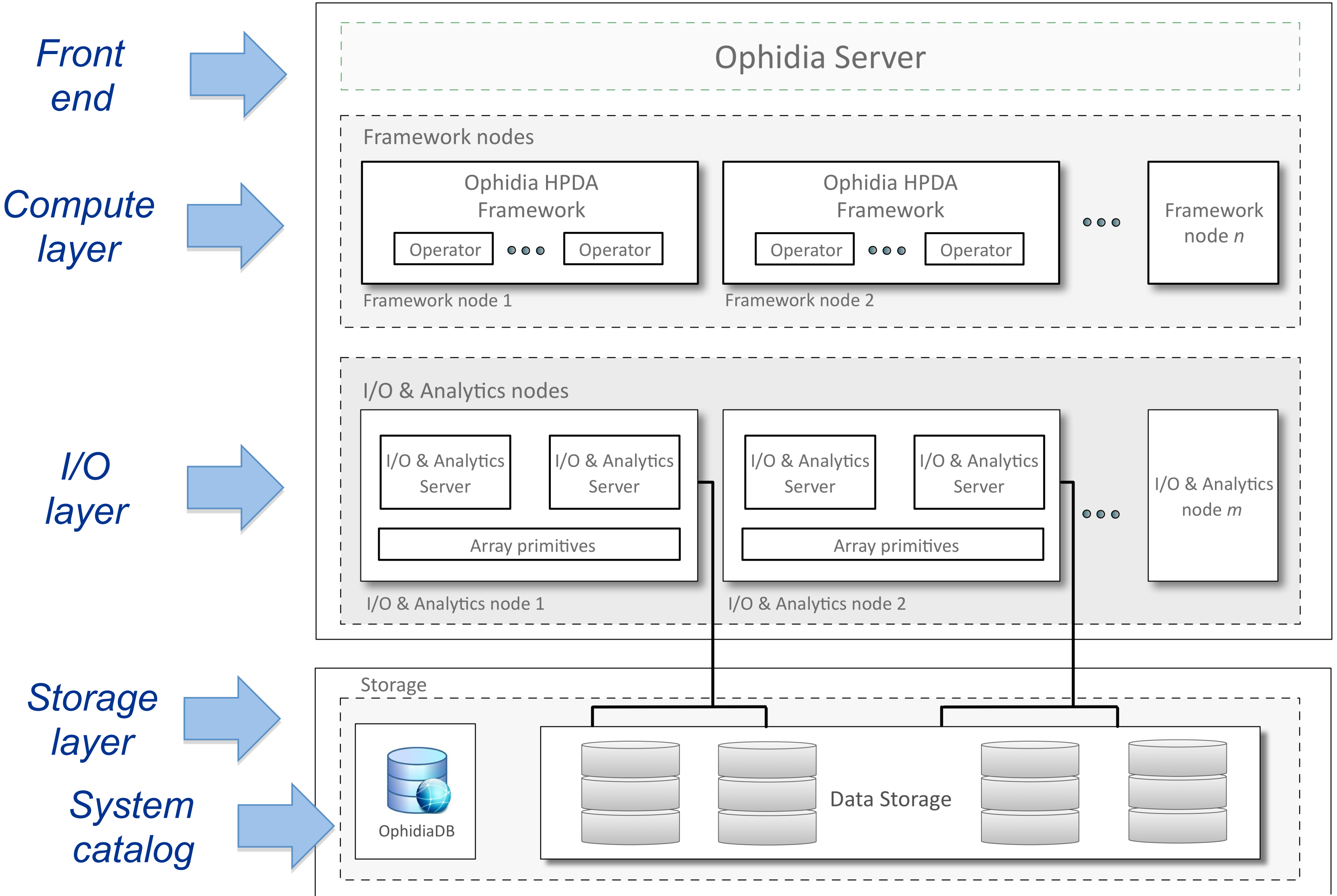
EU Brazil Cloud Connect
EU Brazil Cloud Computing for Science

INDIGO - DataCloud

Session outline

- ✔ ***Introduction to HPDA and data challenges in eScience***
- ✔ ***ECAS and EOSC***
- ✔ ***Introduction to the Ophidia HPDA Framework***
- ✔ ***Ophidia core concepts: architecture, data model, operators and primitives***
- ✔ ***Analytics workflows with Ophidia***
  - ✔ ***Workflow execution demo***
- ✔ ***Ophidia Python bindings: PyOphidia***

Ophidia architecture: overview

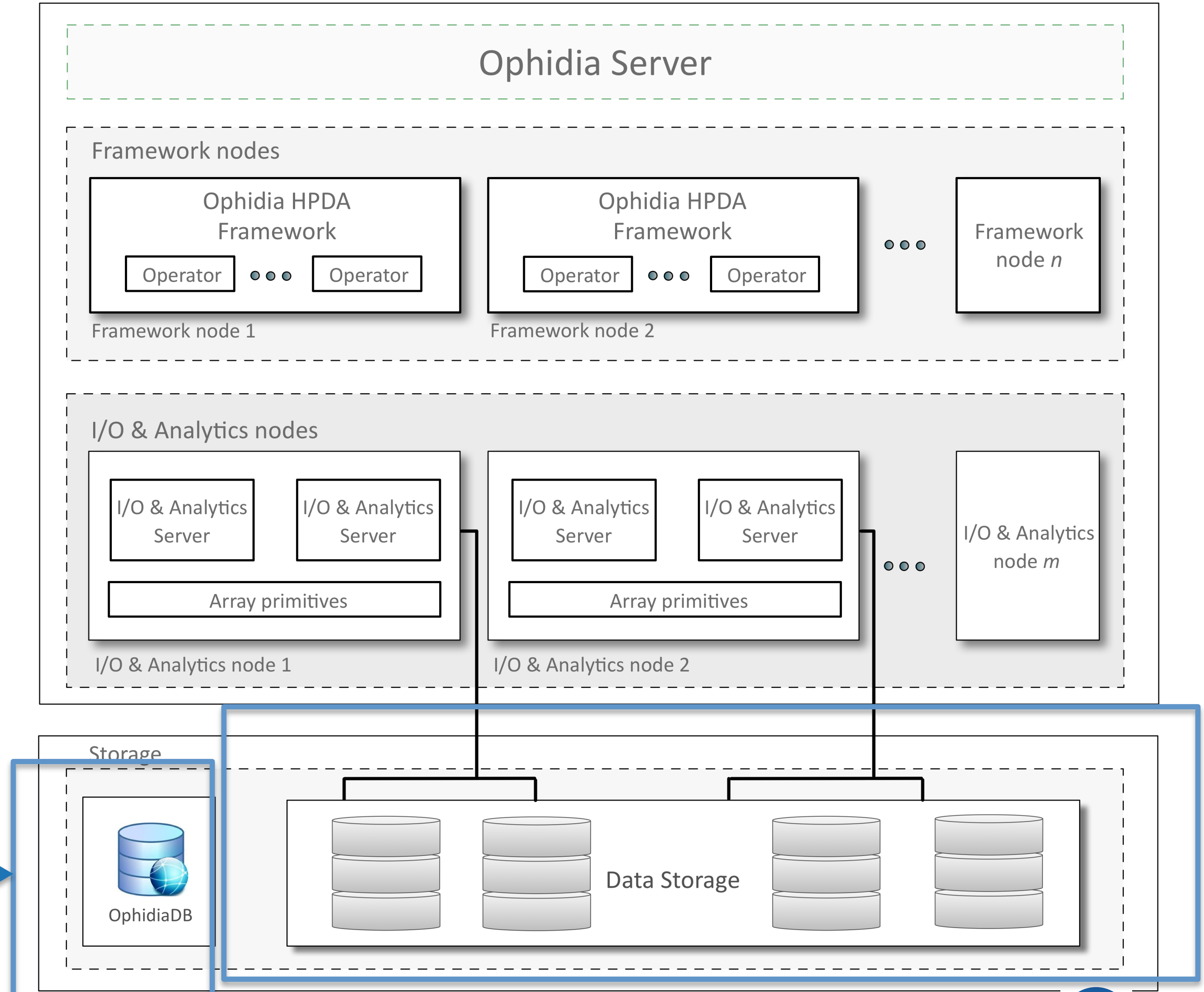


Ophidia architecture: storage layer

Distributed *hardware resources to manage storage*

Data partitioned *in a hierarchical fashion over the storage according to the storage model & partitioning schema*

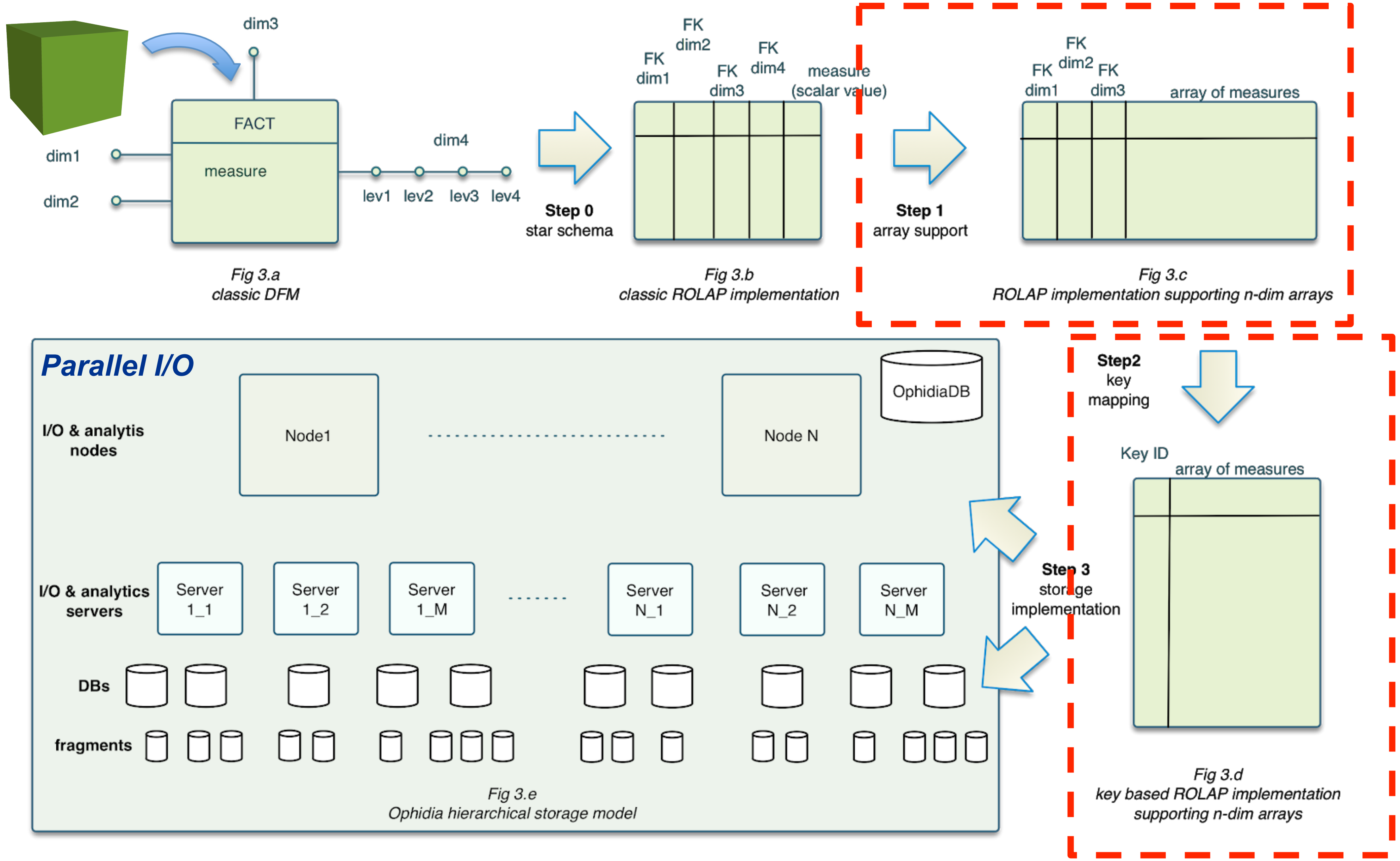
OphidiaDB is the system catalog: maps data fragmentation and tracks metadata



Ophidia storage model

- Ophidia implements the **datacube** abstraction from OLAP
- The Ophidia storage model is a **two-step based evolution** of the **star schema** to support **scientific data management**
- It relies on **implicit** (array-based) and **explicit** (tuple-based) **dimensions** for specific representations of data
- The first step includes the **support for array**-based data
- The second step includes a **key mapping** related to a set of foreign keys
- This second step makes the Ophidia storage model and implementation **independent of the number of dimensions**!

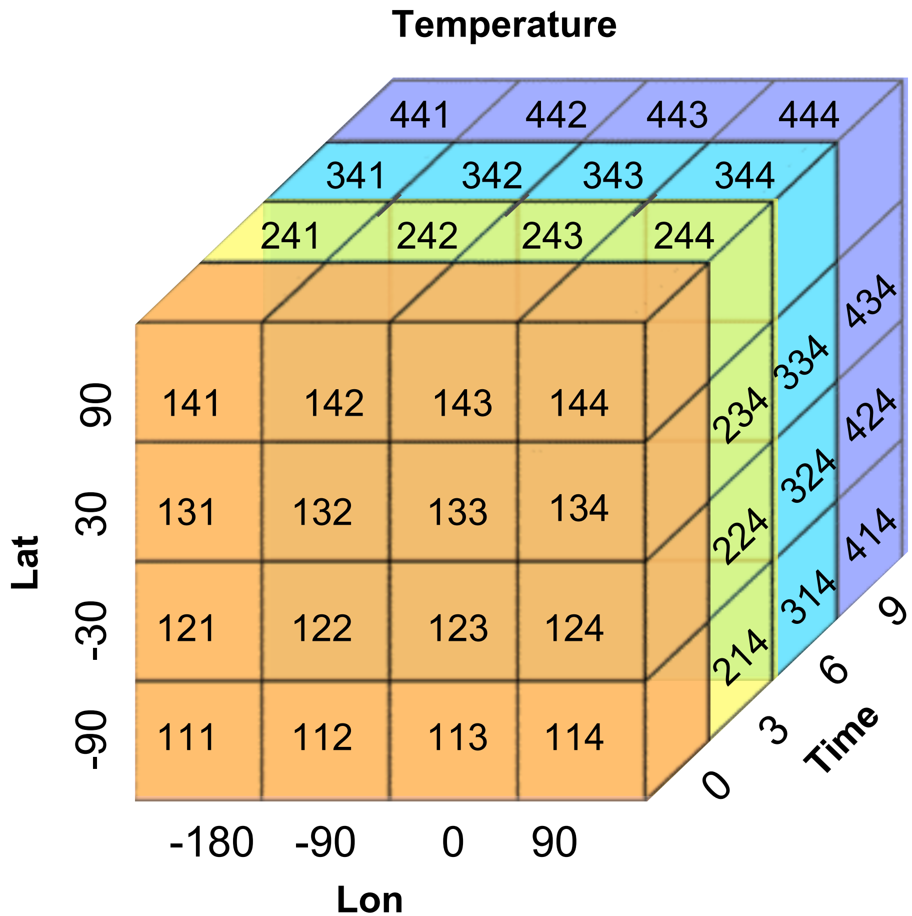
Multi-dimensional storage model implementation



S. Fiore, D. Elia, C. Palazzo, F. Antonio, A. D'Anca, I. Foster and G. Aloisio, "Towards High Performance Data Analytics for Climate Change", ISC High Performance 2019. Lecture Notes in Computer Science, vol. 11887, pp. 240-257, 2019.

From NetCDF to datacube

```
netcdf test {
dimensions:
    lat = 4 ;
    lon = 4 ;
    time = UNLIMITED // (4 currently) ;
variables:
    double lon(lon) ;
    double lat(lat) ;
    double time(time) ;
    float Temperature(time, lat, lon) ;
data:
    lon = -180, -90, 0, 90 ;
    lat = -90, -30, 30, 90 ;
    time = 0, 3, 6, 9 ;
    temperature =
        111, 112, 113, 114,
          121, 122, 123, 124,
          131, 132, 133, 134,
          141, 142, 143, 144,
        211, 212, 213, 214,
          221, 222, 223, 224,
          231, 232, “33, 234,
          241, 242, 243, 244,
        ...
```



The datacube abstraction naturally fits for scientific multi-dimensional data, like climate data

From NetCDF to Ophidia

```
netcdf test {
dimensions:
    lat = 4 ;
    lon = 4 ;
    time = UNLIMITED // (4 currently) ;
variables:
    double lon(lon) ;
    double lat(lat) ;
    double time(time) ;
    float Temperature(time, lat, lon) ;
data:
    lon = -180, -90, 0, 90 ;
    lat = -90, -30, 30, 90 ;
    time = 0, 3, 6, 9 ;
    temperature =
        111, 112, 113, 114,
          121, 122, 123, 124,
          131, 132, 133, 134,
          141, 142, 143, 144,
        211, 212, 213, 214,
          221, 222, 223, 224,
          231, 232, 233, 234,
          241, 242, 243, 244,
        311, 312, 313, 314,
          321, 322, 323, 324,
          331, 332, 333, 334,
          341, 342, 343, 344,
        ...
```

NetCDF

		Temperature			
lat	lon	time[0]	time[1]	time[2]	time[3]
-90	-180	111	211	311	411
-90	-90	112	212	312	412
-90	0	113	213	313	413
-90	90	114	214	314	414
-30	-180	121	221	321	421
-30	-90	122	222	322	422
-30	0	123	223	323	423
-30	90	124	224	324	424
30	-180	131	231	331	431
30	-90	132	232	332	432
30	0	133	233	333	433
30	90	134	234	334	434
90	-180	141	241	341	441
90	-90	142	242	342	442
90	0	143	243	343	443
90	90	144	244	344	444

Ophidia

From NetCDF to Ophidia

```
netcdf test {
dimensions:
    lat = 4 ;
    lon = 4 ;
    time = UNLIMITED // (4 currently) ;
variables:
    double lon(lon) ;
    double lat(lat) ;
    double time(time) ;
    float Temperature(time, lat, lon) ;
data:
    lon = -180, -90, 0, 90 ;
    lat = -90, -30, 30, 90 ;
    time = 0, 3, 6, 9 ;
    temperature =
        111, 112, 113, 114,
          121, 122, 123, 124,
          131, 132, 133, 134,
          141, 142, 143, 144,
        211, 212, 213, 214,
          221, 222, 223, 224,
          231, 232, 233, 234,
          241, 242, 243, 244,
        311, 312, 313, 314,
          321, 322, 323, 324,
          331, 332, 333, 334,
          341, 342, 343, 344,
        ...
```

NetCDF

		Temperature			
lat	lon	time[0]	time[1]	time[2]	time[3]
-90	-180	111	211	311	411
-90	-90	112	212	312	412
-90	0	113	213	313	413
-90	90	114	214	314	414
-30	-180	121	221	321	421
-30	-90	122	222	322	422
-30	0	123	223	323	423
-30	90	124	224	324	424
30	-180	131	231	331	431
30	-90	132	232	332	432
30	0	133	233	333	433
30	90	134	234	334	434
90	-180	141	241	341	441
90	-90	142	242	342	442
90	0	143	243	343	443
90	90	144	244	344	444

Ophidia

From NetCDF to Ophidia

```
netcdf test {
dimensions:
    lat = 4 ;
    lon = 4 ;
    time = UNLIMITED // (4 currently) ;
variables:
    double lon(lon) ;
    double lat(lat) ;
    double time(time) ;
    float Temperature(time, lat, lon) ;
data:
    lon = -180, -90, 0, 90 ;
    lat = -90, -30, 30, 90 ;
    time = 0, 3, 6, 9 ;
    temperature =
        111, 112, 113, 114,
          121, 122, 123, 124,
          131, 132, 133, 134,
          141, 142, 143, 144,
        211, 212, 213, 214,
          221, 222, 223, 224,
          231, 232, 233, 234,
          241, 242, 243, 244,
        311, 312, 313, 314,
          321, 322, 323, 324,
          331, 332, 333, 334,
          341, 342, 343, 344,
        ...
```

NetCDF

ID	Array			
1	111	211	311	411
2	112	212	312	412
3	113	213	313	413
4	114	214	314	414
5	121	221	321	421
6	122	222	322	422
7	123	223	323	423
8	124	224	324	424
9	131	231	331	431
10	132	232	332	432
11	133	233	333	433
12	134	234	334	434
13	141	241	341	441
14	142	242	342	442
15	143	243	343	443
16	144	244	344	444

Ophidia

From NetCDF to Ophidia

```
netcdf test {
dimensions:
    lat = 4 ;
    lon = 4 ;
    time = UNLIMITED // (4 currently) ;
variables:
    double lon(lon) ;
    double lat(lat) ;
    double time(time) ;
    float Temperature(time, lat, lon) ;
data:
    lon = -180, -90, 0, 90 ;
    lat = -90, -30, 30, 90 ;
    time = 0, 3, 6, 9 ;
    temperature =
        111, 112, 113, 114,
          121, 122, 123, 124,
          131, 132, 133, 134,
          141, 142, 143, 144,
        211, 212, 213, 214,
          221, 222, 223, 224,
          231, 232, 233, 234,
          241, 242, 243, 244,
        311, 312, 313, 314,
          321, 322, 323, 324,
          331, 332, 333, 334,
          341, 342, 343, 344,
        ...
```

NetCDF

		Temperature			
lat	lon	time[0]	time[1]	time[2]	time[3]
-90	-180	111	211	311	411
-90	-90	112	212	312	412
-90	0	113	213	313	413
-90	90	114	214	314	414
-30	-180	121	221	321	421
-30	-90	122	222	322	422
-30	0	123	223	323	423
-30	90	124	224	324	424
30	-180	131	231	331	431
30	-90	132	232	332	432
30	0	133	233	333	433
30	90	134	234	334	434
90	-180	141	241	341	441
90	-90	142	242	342	442
90	0	143	243	343	443
90	90	144	244	344	444

Ophidia

From NetCDF to Ophidia

```
netcdf test {
dimensions:
    lat = 4 ;
    lon = 4 ;
    time = UNLIMITED // (4 currently) ;
variables:
    double lon(lon) ;
    double lat(lat) ;
    double time(time) ;
    float Temperature(time, lat, lon) ;
data:
    lon = -180, -90, 0, 90 ;
    lat = -90, -30, 30, 90 ;
    time = 0, 3, 6, 9 ;
    temperature =
        111, 112, 113, 114,
          121, 122, 123, 124,
          131, 132, 133, 134,
          141, 142, 143, 144,
        211, 212, 213, 214,
          221, 222, 223, 224,
          231, 232, 233, 234,
          241, 242, 243, 244,
        311, 312, 313, 314,
          321, 322, 323, 324,
          331, 332, 333, 334,
          341, 342, 343, 344,
        ...
```

NetCDF

FRAG1

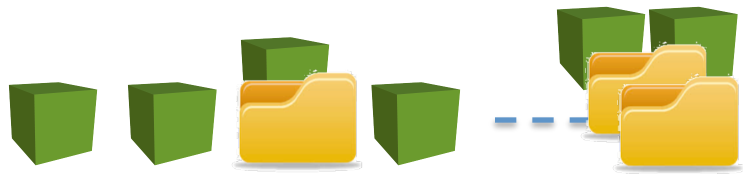
lat	lon	Temperature			
		time[0]	time[1]	time[2]	time[3]
-90	-180	111	211	311	411
-90	-90	112	212	312	412
-90	0	113	213	313	413
-90	90	114	214	314	414
-30	-180	121	221	321	421
-30	-90	122	222	322	422
-30	0	123	223	323	423
-30	90	124	224	324	424

FRAG2

lat	lon	Temperature			
		time[0]	time[1]	time[2]	time[3]
30	-180	131	231	331	431
30	-90	132	232	332	432
30	0	133	233	333	433
30	90	134	234	334	434
90	-180	141	241	341	441
90	-90	142	242	342	442
90	0	143	243	343	443
90	90	144	244	344	444

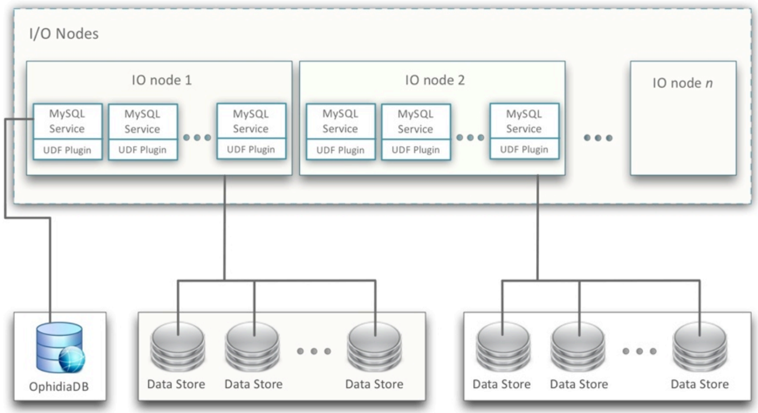
Ophidia

Data abstraction: cube space perspective

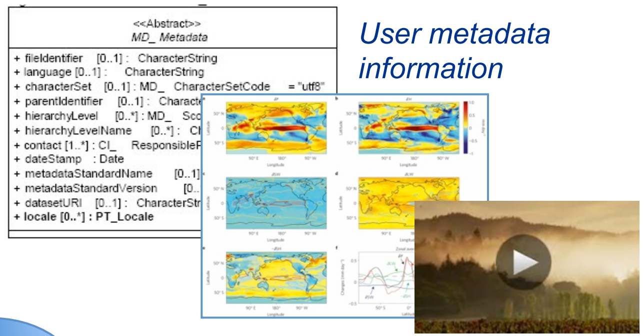


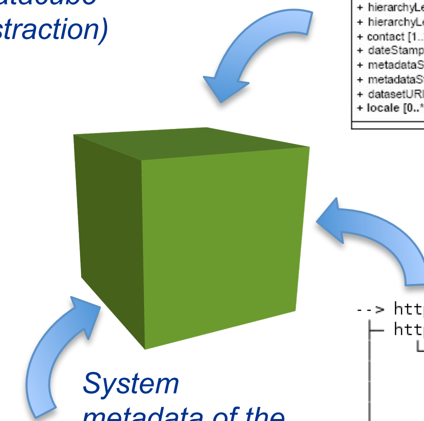
User perspective (datacube abstraction)

System perspective (internal storage representation)



User metadata information





System metadata of the datacube (size, distribution, etc.)

Metadata provenance

```
--> https://ophidia.cmcc.it:8443/162/169 (ROOT)
├ https://ophidia.cmcc.it:8443/162/170 (oph_reduce)
│  └ https://ophidia.cmcc.it:8443/162/171 (oph_merge)
│     ├ https://ophidia.cmcc.it:8443/162/172 (oph_aggregate2)
│     └ https://ophidia.cmcc.it:8443/162/173 (oph_rollup)
│        ├ https://ophidia.cmcc.it:8443/162/174 (oph_reduce)
│        └ https://ophidia.cmcc.it:8443/162/175 (oph_reduce)
├ https://ophidia.cmcc.it:8443/162/176 (oph_aggregate)
└ https://ophidia.cmcc.it:8443/162/177 (oph_aggregate)
```

Manage the Ophidia file system

CMD	BEHAVIOR
cd	change directory
mkdir	create a new folder
rm	remove an empty folder or hide (logically delete) a container
ls	list subfolders and containers in a folder
mv	move/rename a folder or a container
…	…

Metadata associated to the datacubes

TYPE	CONTENT
Text	Plain text metadata
image	Binary string representation of an image
video	Binary string representation of a video
audio	Binary string representation of an audio stream
url	Text representing an URL

Search & Discovery

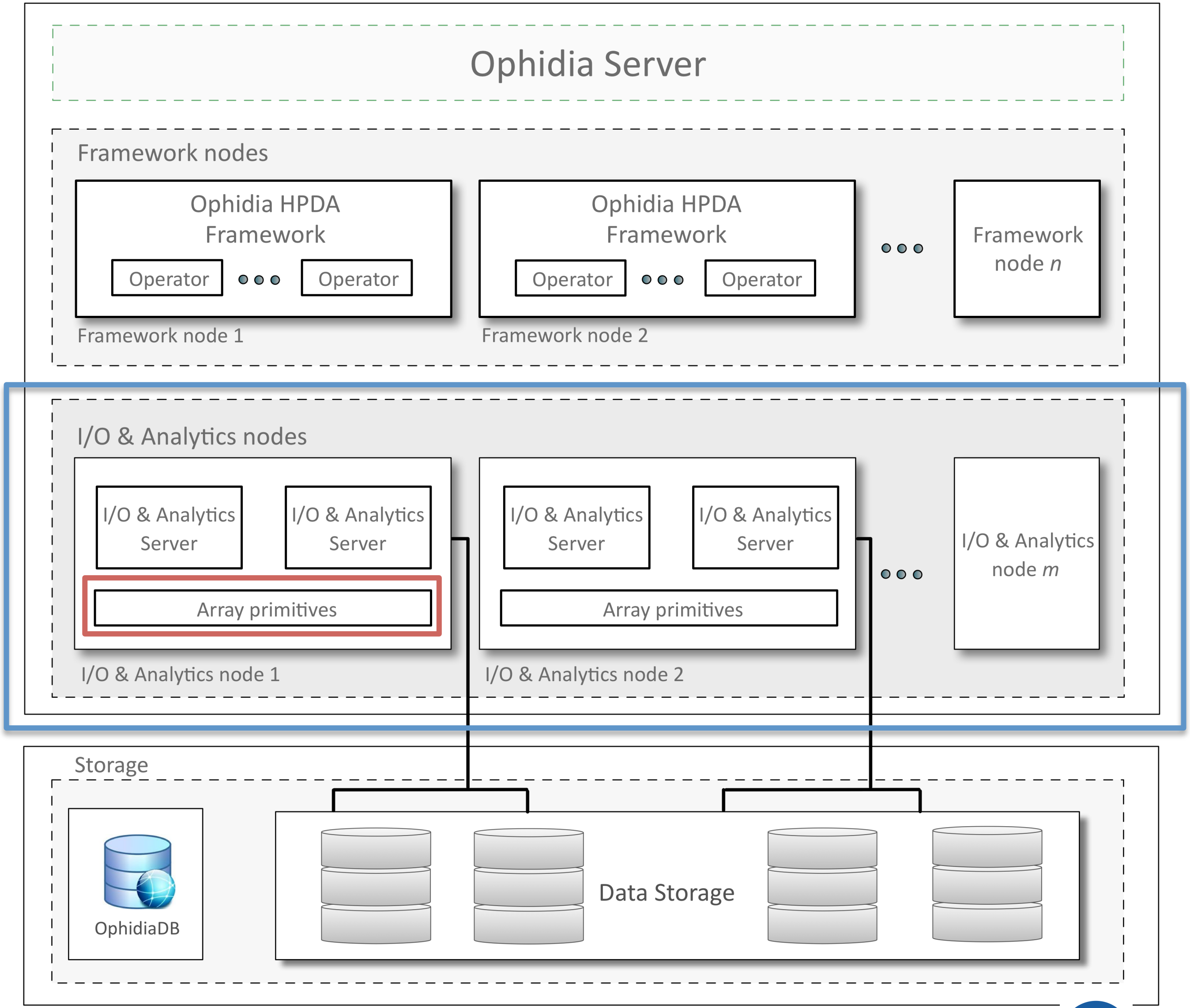
Ophidia architecture: I/O & Analytics layer

*Multiple **I/O & analytics nodes** execute one or more servers*

*Servers run the array-based **primitives** (UDF)*

Servers can transparently interface to different storage back-ends

Support for a native in-memory array-based analytics & I/O engine



Ophidia array-based primitives

Ophidia provides a **wide set of array-based primitives** (around 100) to perform:

- data summarization, sub-setting, predicates evaluation, statistical analysis, array concatenation, algebraic expression, regression, etc.

Primitives come as plugins (UDF) and are applied on a single datacube chunk (fragment)

Primitives can be nested to get more complex functionalities

New primitives can be easily integrated as additional plugins

oph_apply operator to run any primitive on a datacube

```
oph_apply(oph_predicate(measure,'x-298.15','>0','1','0'))
```

Ophidia Primitives documentation: http://ophidia.cmcc.it/documentation/users/primitives/index.html

Array-based primitives: OPH_MATH

*oph_math("OPH_DOUBLE", measure, **OPH_MATH_FUNCTION**)*

SQL query

Ophidia math plugin

Ophidia typing

OPH_MATH_FUNCTION can be one of the macros in the table below

OPH_MATH_FUNCTION MACROS

OPH_MATH_ABS	OPH_MATH_DEGREES	OPH_MATH_RAND
OPH_MATH_ACOS	OPH_MATH_EXP	OPH_MATH_ROUND
OPH_MATH_ASIN	OPH_MATH_FLOOR	OPH_MATH_SIGN
OPH_MATH_ATAN	OPH_MATH_LN	OPH_MATH_SIN
OPH_MATH_CEIL	OPH_MATH_LOG10	OPH_MATH_SQRT
OPH_MATH_COS	OPH_MATH_LOG2	OPH_MATH_TAN
OPH_MATH_COT	OPH_MATH_RADIANS	…

Array based primitives: nesting feature

oph_boxplot(oph_subarray(oph_uncompress(measure), 1,18))

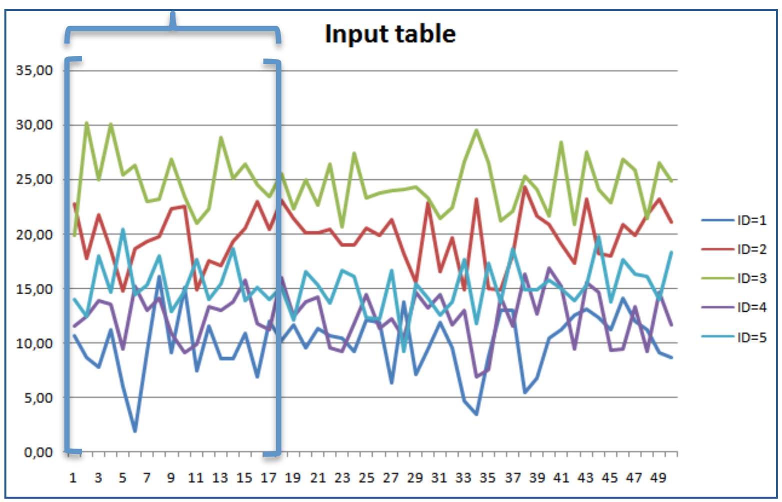
Single chunk or fragment (input)

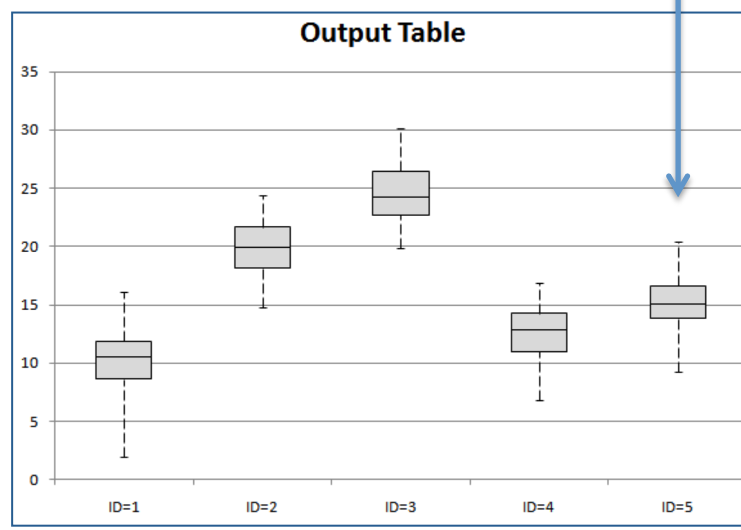
INPUT TABLE 5 tuples x 50 elements											
ID	**MEASURE**										
1	10,73	8,66	7,83	11,20	6,02	1,95	...	16,11	...	8,70
2	22,85	17,84	21,82	18,57	14,81	18,71	...	19,83	...	21,13
3	19,89	30,17	24,95	30,07	25,40	26,31	...	23,18	...	24,82
4	11,60	12,49	13,91	13,53	9,48	15,27	...	14,17	...	11,66
5	13,94	12,43	17,95	14,70	20,41	14,46	...	18,00	...	18,30

Single chunk or fragment (output)

OUTPUT TABLE 5 tuples x 5 elements (summary)					
ID	**MEASURE**				
1	1,95	8,64	10,47	11,87	16,11
2	14,81	18,14	19,93	21,66	24,35
3	19,89	22,74	24,24	26,45	30,17
4	6,87	10,99	12,85	14,28	16,93
5	9,23	13,87	15,05	16,61	20,41

subarray(measure, 1,18)





Array based primitives: oph_aggregate

***oph_aggregate**(measure,"oph_avg")*

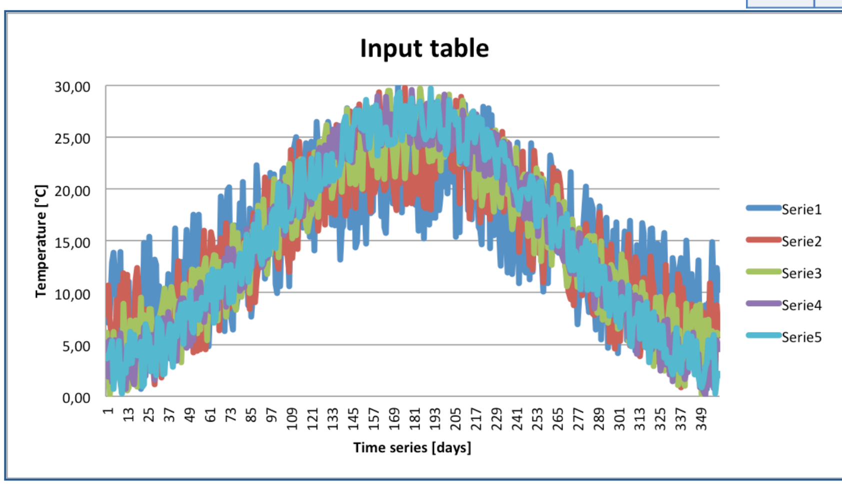
Single chunk or fragment (input)

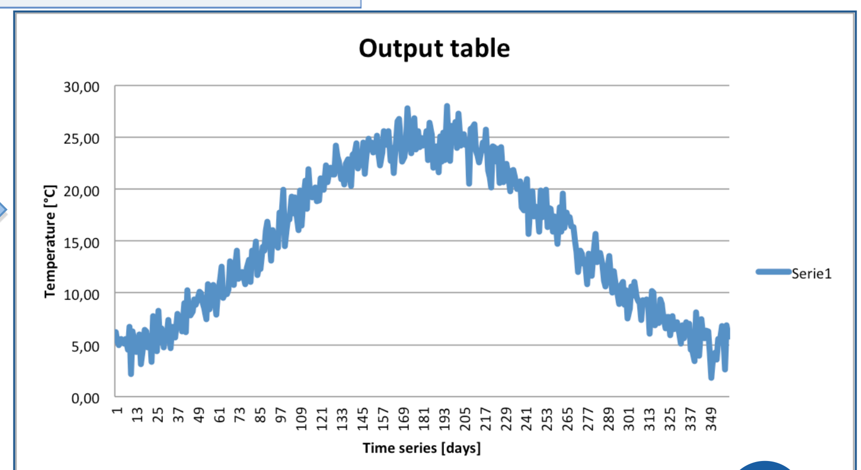
INPUT TABLE 5 tuples x 360 elements										
ID	**MEASURE**									
1	8,40	7,73	7,36	12,68	13,34	11,17	9,09	2,04	...	7,75
2	7,85	10,71	7,23	5,14	4,68	2,61	9,17	8,50	...	6,57
3	6,40	3,48	0,44	2,81	6,16	2,01	3,61	3,83	...	5,88
4	5,60	4,68	5,54	5,84	5,47	5,37	5,30	7,24	...	3,06
5	3,55	4,10	4,59	5,07	6,97	2,07	3,06	3,06	...	7,88

Vertical aggregation

OUTPUT TABLE 1 tuple x 360 elements							
ID	**MEASURE**						
1	6,25	5,35	5,00	5,57	5,41	...	5,11

Single chunk or fragment (output)



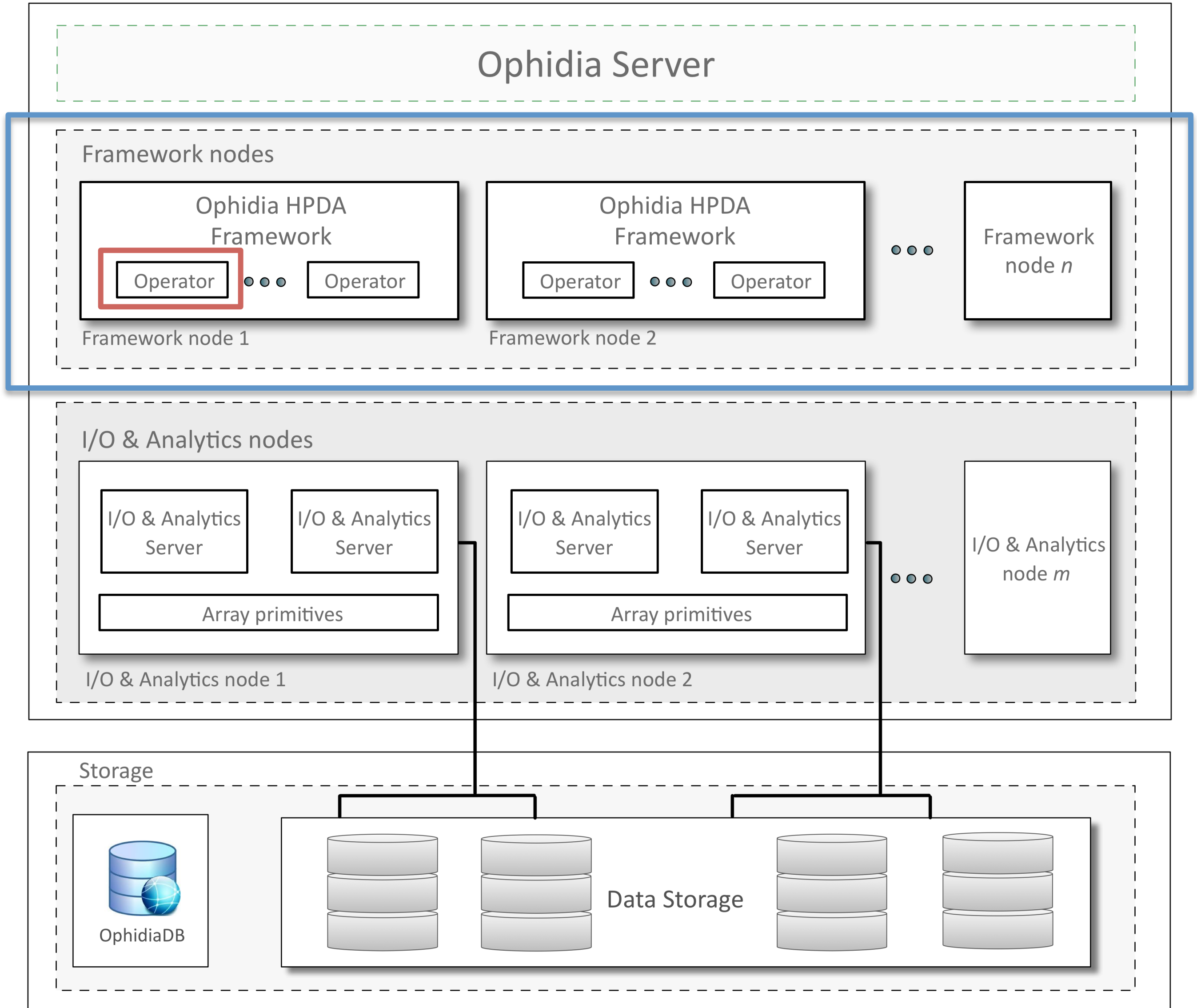


Ophidia architecture: framework layer

The Ophidia analytics framework can be executed with multiple processes/threads

*Provides the environment for the execution of parallel MPI/Pthread-based **operators***

*Operators manipulate the entire set of fragments associated to a **whole datacube***



Ophidia operators

CLASS	PROCESSING TYPE	OPERATOR(S)
I/O	Parallel	OPH_IMPORTNC, OPH_EXPORTNC, OPH_CONCATNC, OPH_RANDUCUBE
Time series processing	Parallel	OPH_APPLY
Datacube reduction	Parallel	OPH_REDUCE, OPH_REDUCE2, OPH_AGGREGATE
Datacube subsetting	Parallel	OPH_SUBSET
Datacube combination	Parallel	OPH_INTERCUBE, OPH_MERGECUBES
Datacube structure manipulation	Parallel	OPH_SPLIT, OPH_MERGE, OPH_ROLLUP, OPH_DRILLDOWN, OPH_PERMUTE
Datacube/file system management	Sequential	OPH_DELETE, OPH_FOLDER, OPH_FS
Metadata management	Sequential	OPH_METADATA, OPH_CUBEIO, OPH_CUBESCHEMA
Datacube exploration	Sequential	OPH_EXPLORECUBE, OPH_EXPLORENC

About 50 operators for data and metadata processing

Ophidia operators documentation: http://ophidia.cmcc.it/documentation/users/operators/index.html

Operators commands

```
[ OPH_TERM ] >> oph_operator param1=val1;param2=val2;...paramN=valN;
```

Ophidia Operator such as "oph_list"

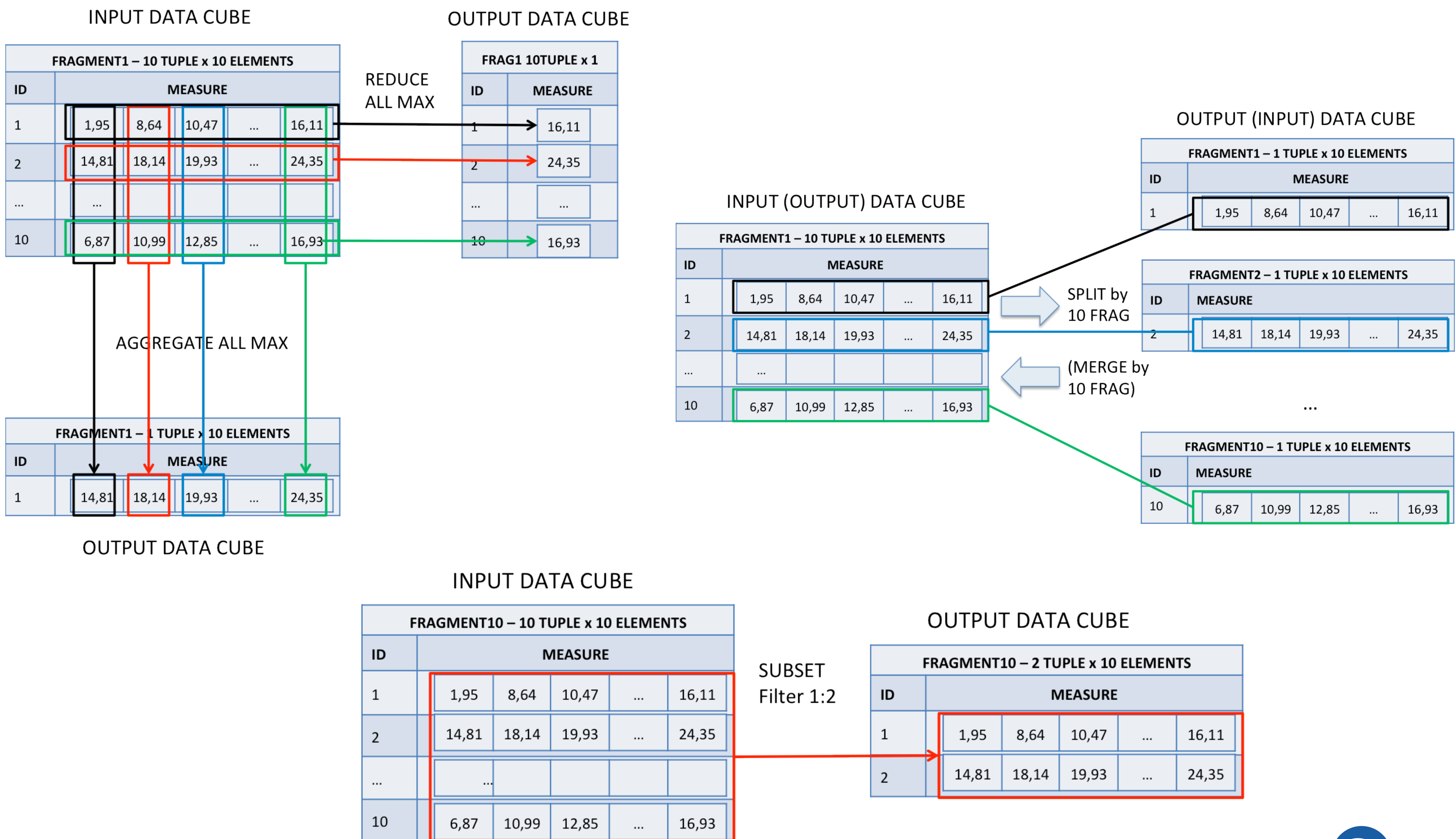
Sequence of arguments passed to the operator

The operator strings follow a declarative style, with common arguments shared among all operators and some operator-specific arguments

Special arguments:

- "exec_mode": specifies if the command is executed in synchronous (”sync") or asynchronous mode (”async") which is the default;
- "ncores": it specifies the number of parallel processes requested for the execution of the operator (default is 1);
- "nthreads": it specifies the number of parallel threads per each processes requested for the execution of the operator (default is 1);
- "cube": it specifies the input datacube. It is automatically added by the terminal exploiting the last produced cube.

Ophidia "data" operators



Ophidia “data” operators

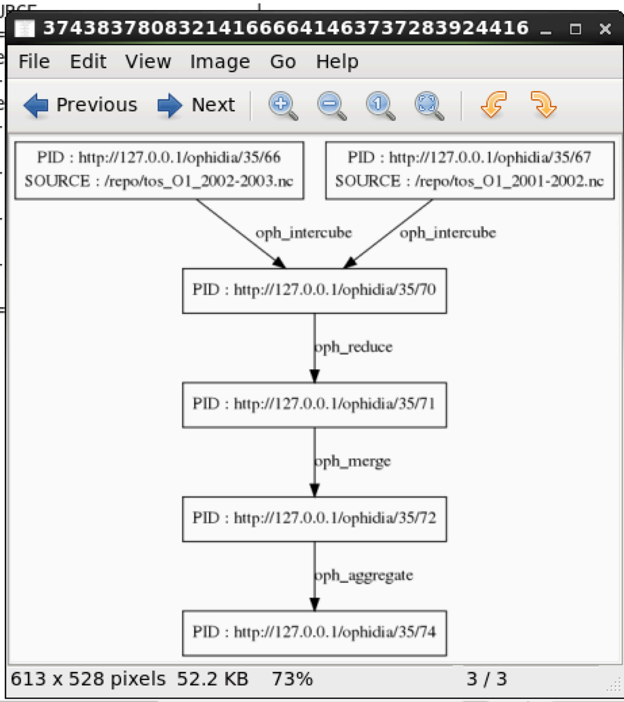
```
[37..4416] >> oph_explorecube cube=http://127.0.0.1/ophidia/35/67;subset_dims=lat|lon|time;subset_filter=39:42|15:19|1:275;show_time=yes;
[Request]:
operator=oph_explorecube;cube=http://127.0.0.1/ophidia/35/67;subset_dims=lat|lon|time;subset_filter=39:42|15:19|1:275;show_time=yes;sessionid=http://127.0.0.1/ophidia/sessions/3
74383780832141666641463737283924416/experiment;exec_mode=sync;ncores=1;cwd=/;

[JobID]:
http://127.0.0.1/ophidia/sessions/374383780832141666641463737283924416/experiment?106#224

[Response]:
tos
---
```

lat	lon	tos
39.500000	15.000000	1.00000002e+20, 1.00000002e+20, 1.00000002e+20, 1.00000002e+20, 1.00000002e+20, 1.00000002e+20, 1.00000002e+20, 1.00000002e+20, 1.00000002e+20
39.500000	17.000000	287.3930664062, 286.8287048340, 286.5860595703, 286.9228210449, 288.5254516602, 292.3968200684, 295.8656921387, 297.2062072754, 295.7126464844
39.500000	19.000000	287.6926879883, 287.0508117676, 286.7896118164, 287.0781555176, 288.6802062988, 292.6882629395, 296.4769287109, 297.6632385254, 296.3418273926
40.500000	15.000000	1.00000002e+20, 1.00000002e+20, 1.00000002e+20, 1.00000002e+20, 1.00000002e+20, 1.00000002e+20, 1.00000002e+20, 1.00000002e+20, 1.00000002e+20
40.500000	17.000000	287.1098632812, 286.5683593750, 286.2949829102, 286.5216674805, 288.0316772461, 291.7698974609, 295.4139709473, 296.8489685059, 295.4132995605
40.500000	19.000000	287.4010009766, 286.7818298340, 286.4914245605, 286.7260742188, 288.3006286621, 292.1842346191, 296.0237731934, 297.2694702148, 295.9751892090
41.500000	15.000000	1.00000002e+20, 1.00000002e+20, 1.00000002e+20, 1.00000002e+20, 1.00000002e+20, 1.00000002e+20, 1.00000002e+20, 1.00000002e+20, 1.00000002e+20
41.500000	17.000000	286.5835876465, 286.0175781250, 285.7146911621, 285.9142761230, 287.4476623535, 291.1032104492, 294.7090454102, 296.0852355957, 294.7053222656
41.500000	19.000000	286.9717712402, 286.3946838379, 286.0617675781, 286.1446228027, 287.6101989746, 291.2955017090, 295.2700195312, 296.5146179199, 295.3194274902

```
Summary
-------
Selected 9 rows out of 9
```

Ophidia “metadata” operators

```
[37..4416] >> oph_cubeio
[Request]:
operator=oph_cubeio;sessionid=http://127.0.0.1/ophidia/sessions/374383780832141666641463737283924416/experiment;exec_mode=sync;ncores=1;cube=http://127.0.0.1/ophidia/35/74;cwd=/
;

[JobID]:
http://127.0.0.1/ophidia/sessions/374383780832141666641463737283924416/experiment?82#176

[Response]:
Cube Provenance
---------------

+=================================================================+===============+================================+=========================+
| INPUT CUBE                                                      | OPERATION     | OUTPUT CUBE                    | SOURCE
+=================================================================+===============+================================+====
|                                                                 | ROOT          | http://127.0.0.1/ophidia/35/66 | /re
|-----------------------------------------------------------------|---------------|--------------------------------|----
|                                                                 | ROOT          | http://127.0.0.1/ophidia/35/67 | /re
|-----------------------------------------------------------------|---------------|--------------------------------|----
| http://127.0.0.1/ophidia/35/66 - http://127.0.0.1/ophidia/35/67 | oph_intercube | http://127.0.0.1/ophidia/35/70 |
|-----------------------------------------------------------------|---------------|--------------------------------|----
| http://127.0.0.1/ophidia/35/70                                  | oph_reduce    | http://127.0.0.1/ophidia/35/71 |
|-----------------------------------------------------------------|---------------|--------------------------------|----
| http://127.0.0.1/ophidia/35/71                                  | oph_merge     | http://127.0.0.1/ophidia/35/72 |
|-----------------------------------------------------------------|---------------|--------------------------------|----
| http://127.0.0.1/ophidia/35/72                                  | oph_aggregate | http://127.0.0.1/ophidia/35/74 |
+=================================================================+===============+================================+====

Cube Provenance Graph
---------------------

Directed Graph DOT string :
digraph DG {
        node    [shape=box]

        0       [label="PID : http://127.0.0.1/ophidia/35/74\n"]
        1       [label="PID : http://127.0.0.1/ophidia/35/72\n"]
        2       [label="PID : http://127.0.0.1/ophidia/35/71\n"]
        3       [label="PID : http://127.0.0.1/ophidia/35/70\n"]
        4       [label="PID : http://127.0.0.1/ophidia/35/66\nSOURCE : /repo/tos_01_2002-2003.nc\n"]
        5       [label="PID : http://127.0.0.1/ophidia/35/67\nSOURCE : /repo/tos_01_2001-2002.nc\n"]

        1->0    [label="oph_aggregate"]
        2->1    [label="oph_merge"]
```



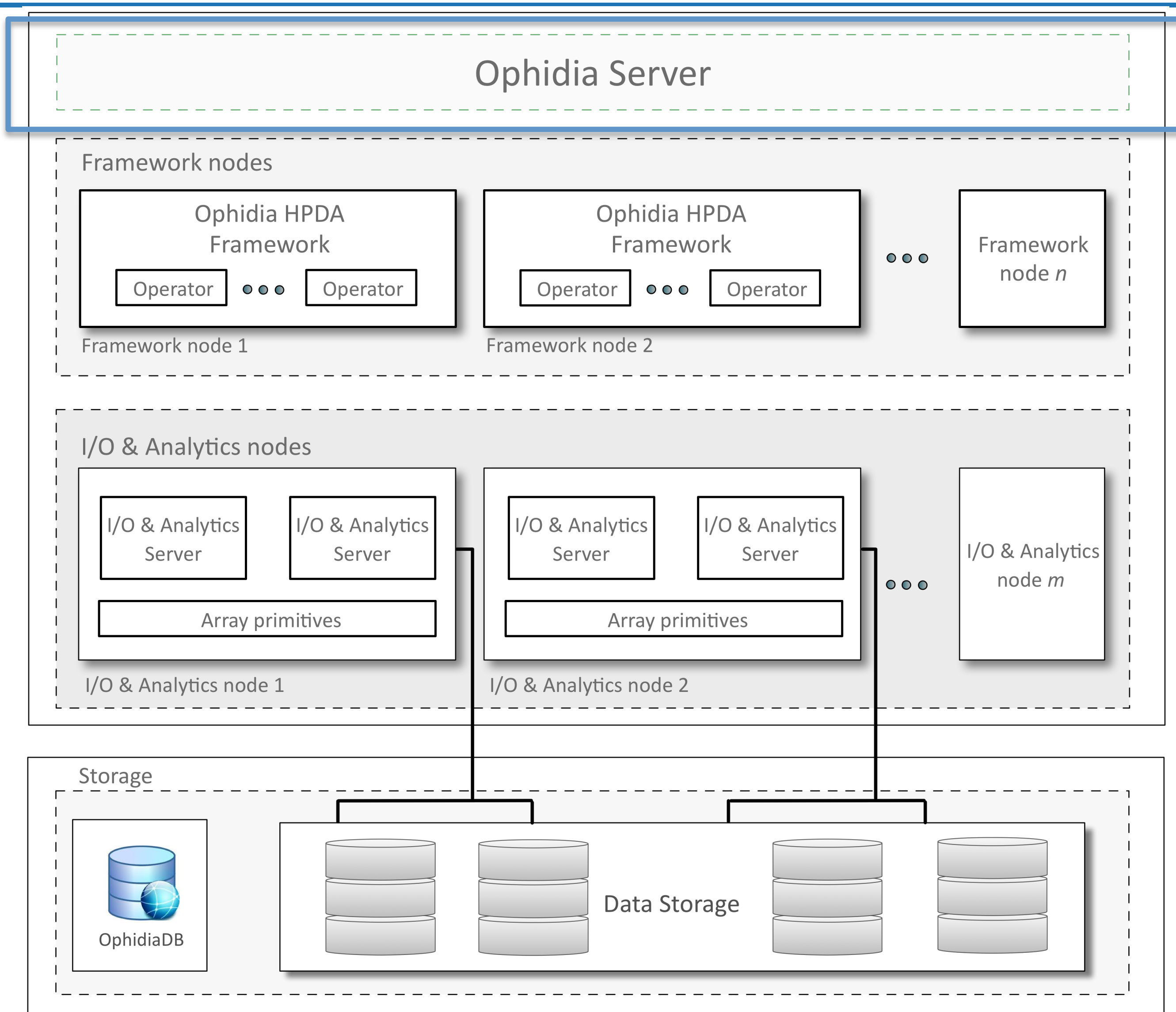
Ophidia architecture: front-end layer

Multi-interface *server front-end*

*Manages user **authN/authZ, sessions** and requests*

*Manages **task/ workflow** execution*

Remote interactions with a CLI, WPS clients and Python modules



Ophidia Terminal

The **Ophidia Terminal**, a CLI bash-like client for the Ophidia framework:

- Executing *interactive* data analytics sessions;
- Executing *batch* data analytics tasks of *workflows*;
- Experiment and operators *debugging*;
- *File system exploration* and *environment management*.

```
[11..4495] >> oph_list level=2;
[Request]:
operator=oph_list;path=;level=2;sessionid=http://127.0.0.1/ophidia/sessions/1112
38695229505952271558621818154495/experiment;exec_mode=sync;cdd=/;

[JobID]:
http://127.0.0.1/ophidia/sessions/111238695229505952271558621818154495/experiment?2#45

[Response]:
Ophidia Filesystem: /
---------------------

+===+=====================+============================================+=============+
| T | PATH                | DATACUBE PID                               | DESCRIPTION |
+===+=====================+============================================+=============+
| f | testFolder/         |                                            |             |
|---|---------------------|--------------------------------------------|-------------|
| c | test                | http://127.0.0.1/ophidia/2917/374976       |             |
+===+=====================+============================================+=============+
```

Three levels of parallelism

Datacube-level parallelism

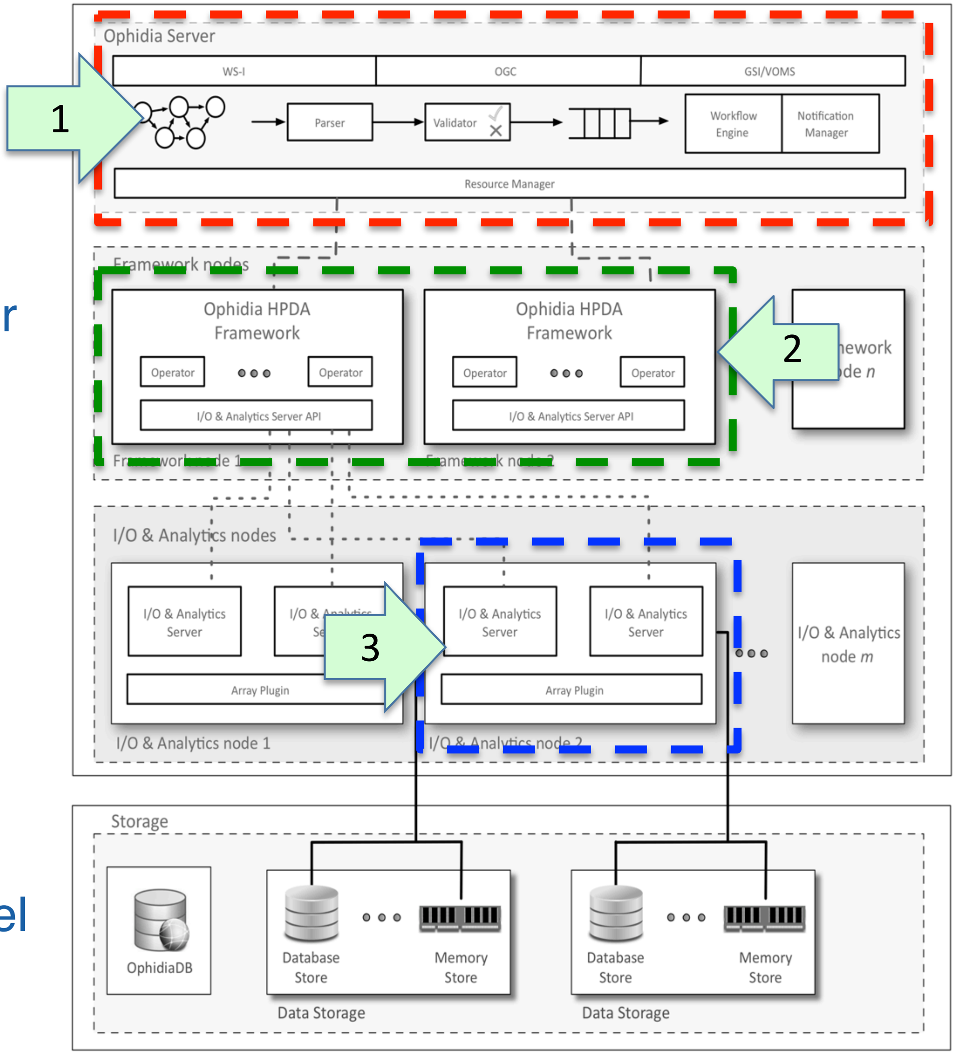
- HTC paradigm
- At the front-end level
- Based on the "massive" operator concept

Framework-level parallelism

- HPC paradigm
- MPI/Pthread
- At the HPDA framework level

Fragment-level parallelism

- OpenMP based
- At the I/O & analytics server level



On-demand instantiation of an Ophidia custer

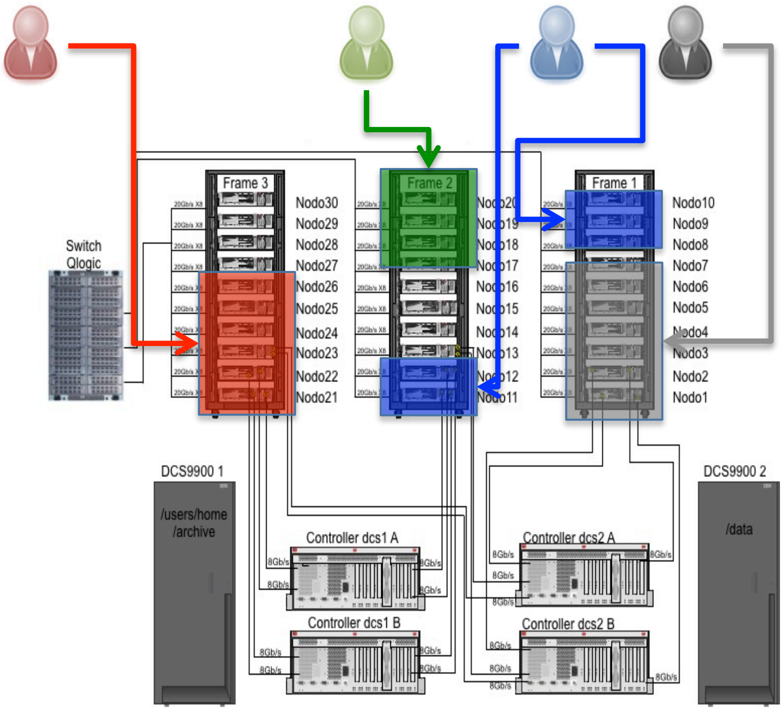
Target environment: HPC cluster

Deployment of I/O & analytics servers

- `oph_cluster action=deploy;nhost=64;cluster_name=new;`
- `oph_cluster action=undeploy;cluster_name=new;`

Multiple isolated instances can be deployed simultaneously by different teams/users

Zeus SuperComputer at CMCC: 1.2 PetaFlops, 348 nodes



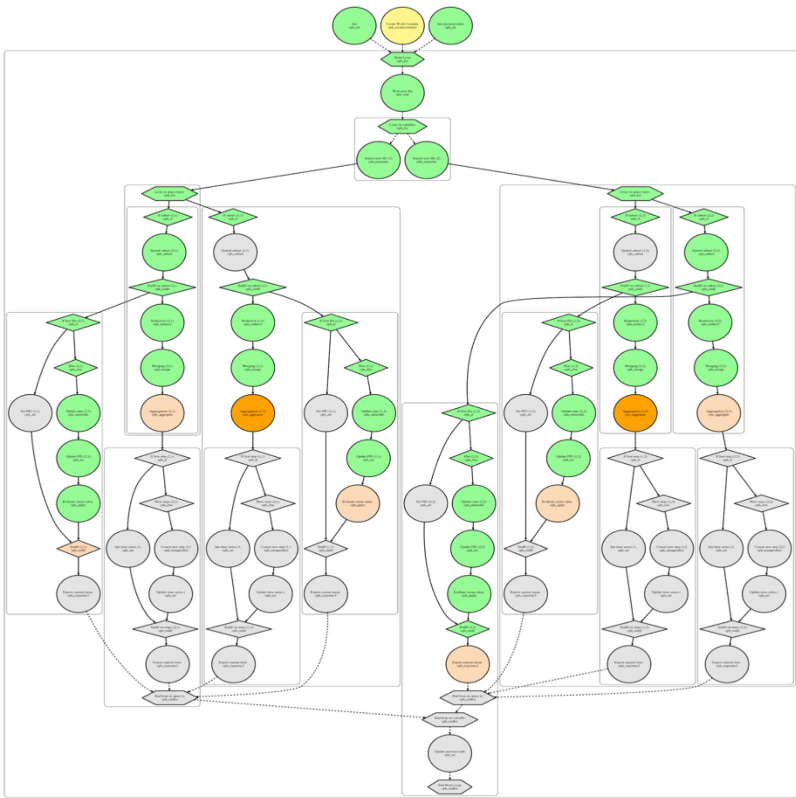
Session outline

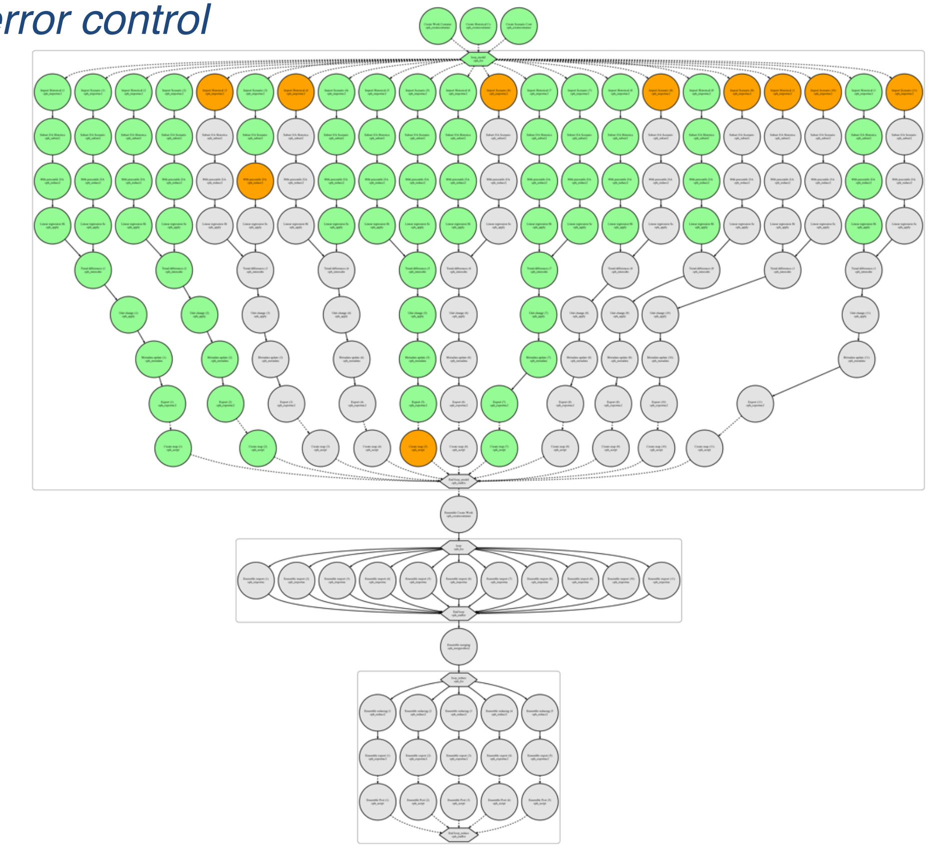
- ✔ *Introduction to HPDA and data challenges in eScience*
- ✔ *ECAS and EOSC*
- ✔ *Introduction to the Ophidia HPDA Framework*
- ✔ *Ophidia core concepts: architecture, data model, operators and primitives*
- ✔ *Analytics workflows with Ophidia*
  - ✔ *Workflow execution demo*
- ✔ *Ophidia Python bindings: PyOphidia*

Analytics workflows

Ophidia supports the execution of complex workflows of operators.

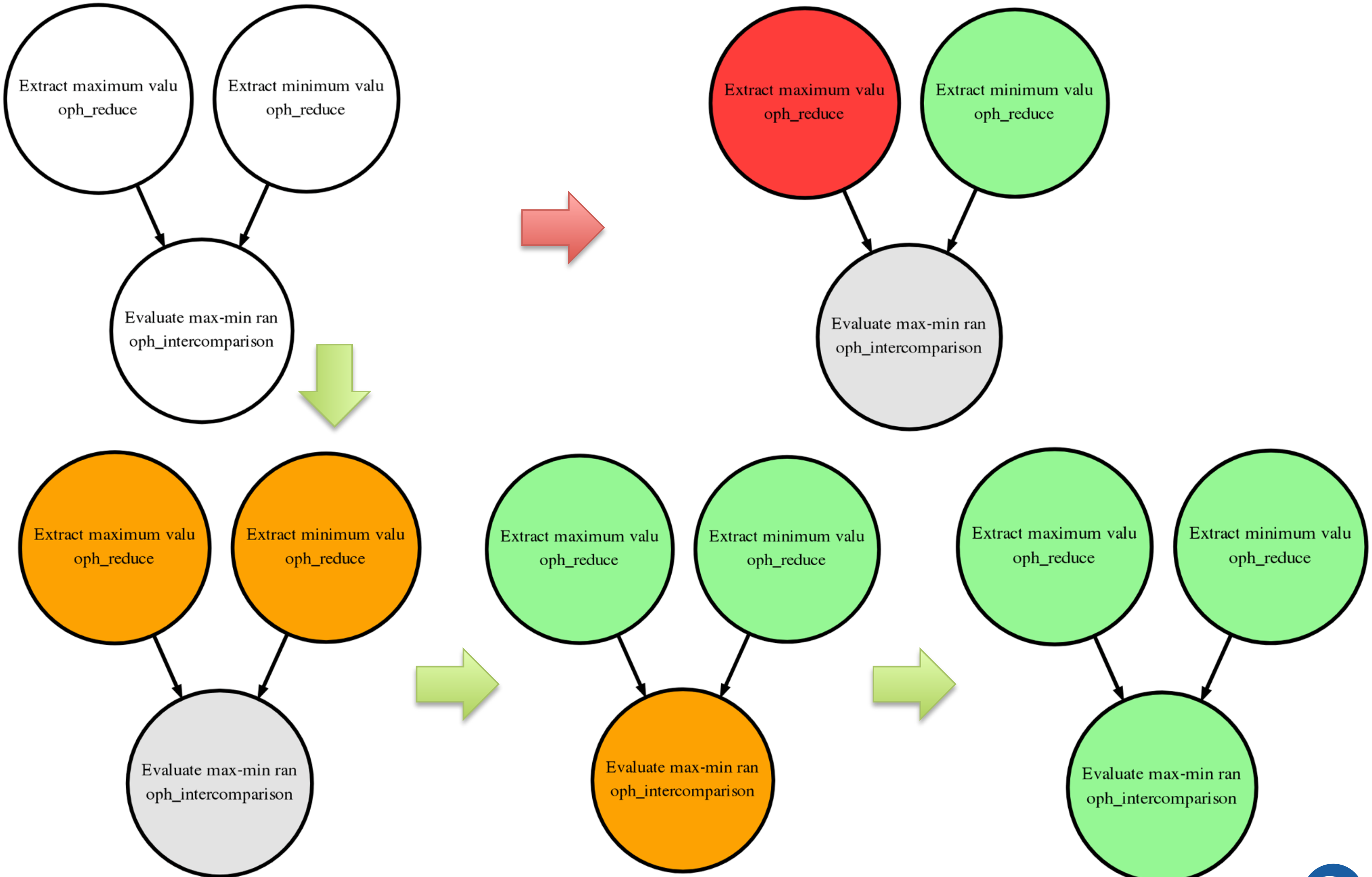
- It defines a **JSON representation** for the workflow DAG specification
- Supports different constructs: *dependencies; massive tasks; iterative (group of) tasks; parallel (group of) tasks; flow and error control*





C. Palazzo, A. Mariello, S. Fiore, A. D'Anca, D. Elia, D. N. Williams, G. Aloisio, "A Workflow-Enabled Big Data Analytics Software Stack for eScience", HPCS 2015, pp. 545-552

Workflow support



Analytics workflows constructs

Workflow Management

This group includes a number of flow control operators that could be used within an Ophidia workflow to implement complex data processing in batch mode. In particular, they implement several advanced features: setting of run-time variables, iterative and parallel interface, selection interface, interactive workflows, interleaving workflows, etc.

NAME	DESCRIPTION
OPH_ELSE	Start the last sub-block of a selection block "if".
OPH_ELSEIF	Start a new sub-block of a selection block "if".
OPH_ENDFOR	Close a loop "for".
OPH_ENDIF	Close a selection block "if".
OPH_FOR	Implement a loop "for".
OPH_IF	Open a "if" selection block.
OPH_INPUT	It sends commands or data to an interactive task.
OPH_SET	Set a parameter in the workflow environment.
OPH_WAIT	Wait until an event occurs.

Ophidia workflow documentation: http://ophidia.cmcc.it/documentation/users/workflow/index.html

Behind the scene: workflow JSON representation

ophrpm@ophidiarpm:~/workflow

ophrpm@ophidiarpm:~/workflow

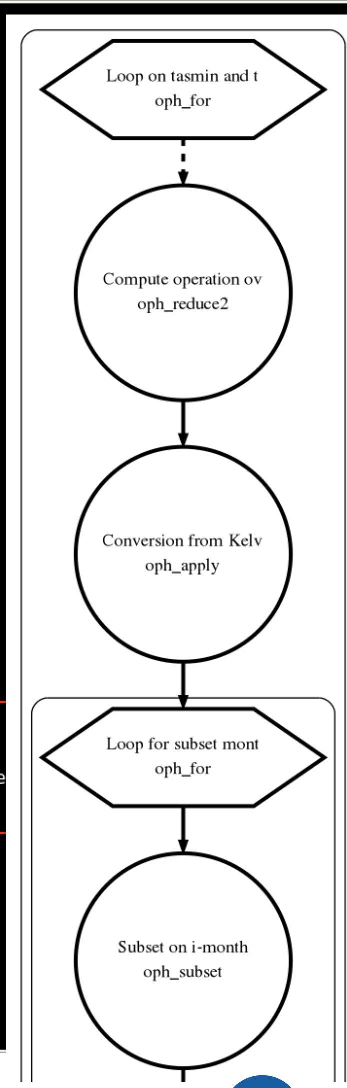
```
    "tasks": [
        {
            "name": "Loop on tasmin and tasmax cubes",
            "operator": "oph_for",
            "arguments": [ "name=cube","counter=1:2", "values=${1}|${2}", "parallel=yes" ]
        },
                {
                        "name": "Compute operation over time",
                        "operator": "oph_reduce2",
                        "arguments": [
                                "cube=@{cube}",
                                "dim=time",
                                "concept_level=M",
                                "midnight=00",
                                "operation=$3",
                                "container=tmp"
                        ],
                        "dependencies": [
                                { "task": "Loop on tasmin and tasmax cubes" }
                        ]
                },
                {
                        "name": "Conversion from Kelvin to Celsius degrees",
                        "operator": "oph_apply",
                        "arguments": [
                                "query=oph_sum_scalar('oph_float','oph_float',measure,-273.15)"
                        ],
                        "dependencies": [{
                                "task": "Compute operation over time",
                                "type": "single"
                        }]
                },
                {
                        "name": "Loop for subset months",
                        "operator": "oph_for",
                        "arguments": [ "name=index","counter=1:12", "values=Jan|Feb|Mar|Apr|May|Jun|Jul|Aug|Sep|Oct|Nov|Dec", "parallel=yes" ],
                        "dependencies": [ { "task":"Conversion from Kelvin to Celsius degrees", "type": "single" } ]
                },
                        {
                                "name": "Subset on i-month",
                                "operator": "oph_subset",
                                "arguments": [
                                        "subset_dims=time",
                                        "subset_filter=&index:12:end"
                                ],
                                "dependencies": [
--More--(65%)
```

Behind the scene: workflow JSON representation

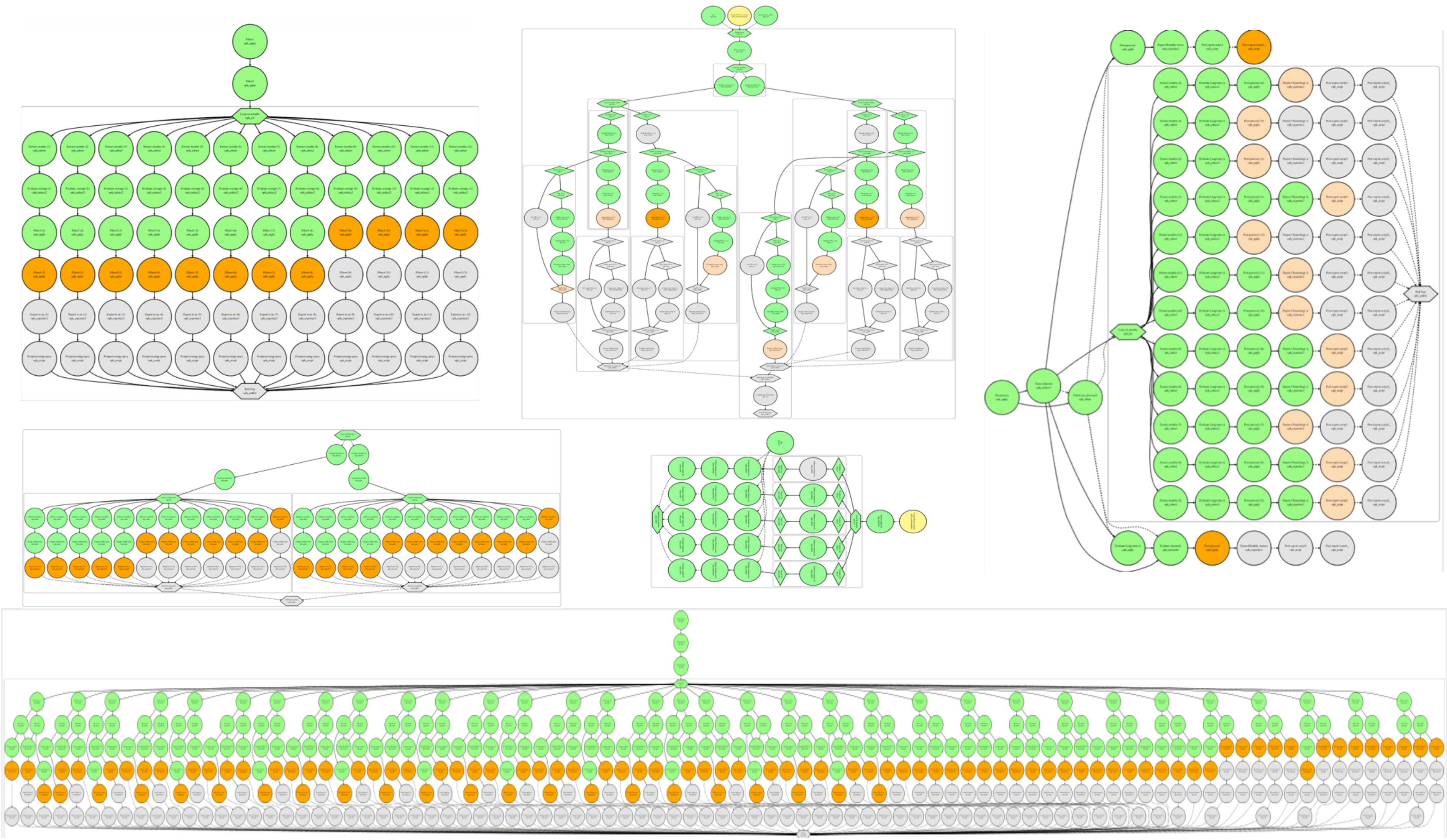
ophrpm@ophidiarpm:~/workflow


```
    "tasks": [
        {
            "name": "Loop on tasmin and tasmax cubes",
            "operator": "oph_for",
            "arguments": [ "name=cube","counter=1:2", "values=${1}|${2}", "parallel=yes" ]
        },
                {
                        "name": "Compute operation over time",
                        "operator": "oph_reduce2",
                        "arguments": [
                                "cube=@{cube}",
                                "dim=time",
                                "concept_level=M",
                                "midnight=00",
                                "operation=$3",
                                "container=tmp"
                        ],
                        "dependencies": [
                                { "task": "Loop on tasmin and tasmax cubes" }
                        ]
                },
                {
                        "name": "Conversion from Kelvin to Celsius degrees",
                        "operator": "oph_apply",
                        "arguments": [
                                "query=oph_sum_scalar('oph_float','oph_float',measure,-273.15)"
                        ],
                        "dependencies": [{
                                "task": "Compute operation over time",
                                "type": "single"
                        }]
                },
                {
                        "name": "Loop for subset months",
                        "operator": "oph_for",
                        "arguments": [ "name=index","counter=1:12", "values=Jan|Feb|Mar|Apr|May|Jun|Jul|Aug|Sep|Oct|N
                        "dependencies": [ { "task":"Conversion from Kelvin to Celsius degrees", "type": "single" } ]
                },
                        {
                                "name": "Subset on i-month",
                                "operator": "oph_subset",
                                "arguments": [
                                        "subset_dims=time",
                                        "subset_filter=&index:12:end"
                                ],
                                "dependencies": [
--More--(65%)
```


ophrpm@ophidiarpm:~/workflow



Analytics workflows support and interfaces

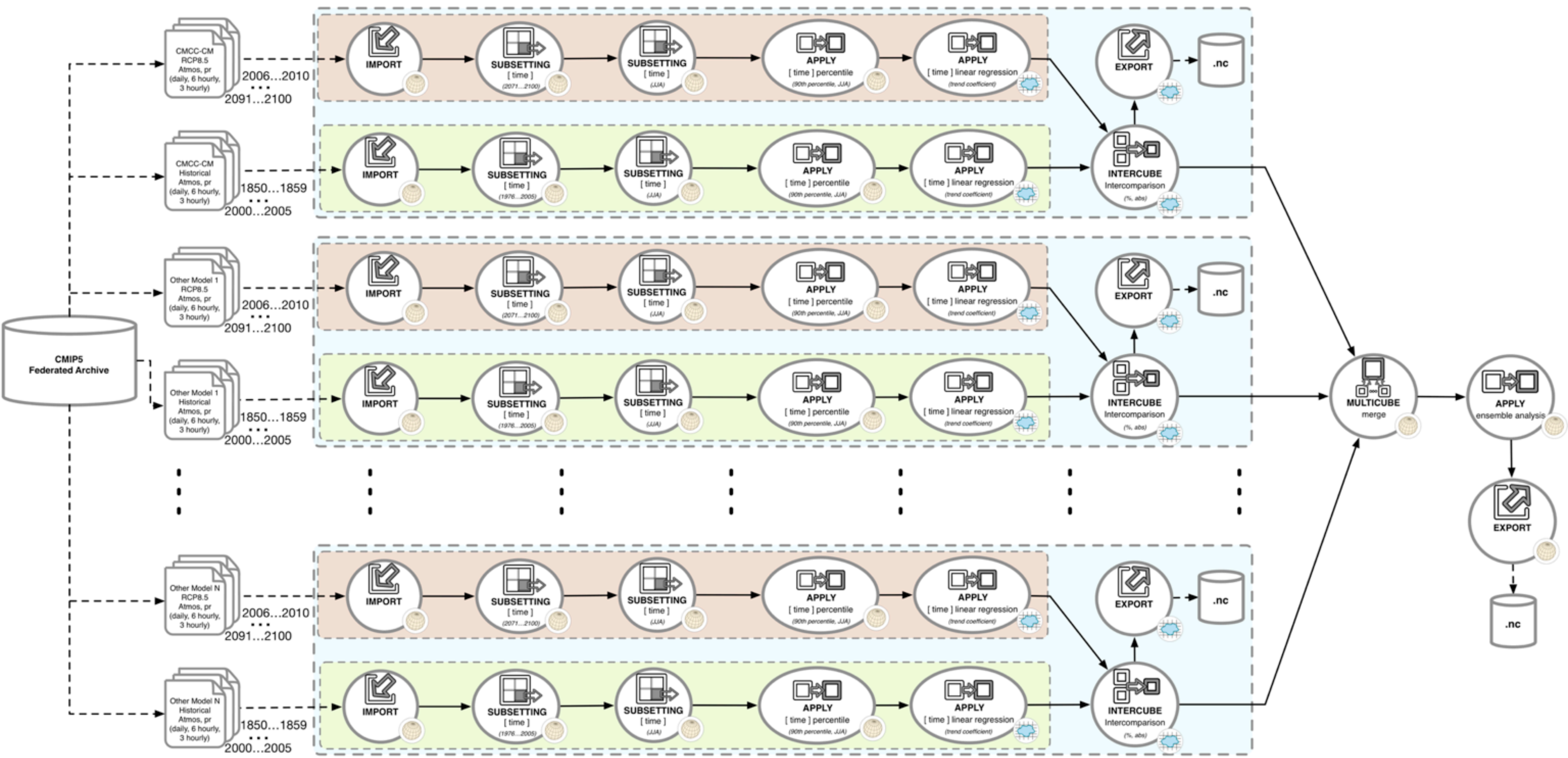


Session outline

- ✔ *Introduction to Big Data, HPDA and data challenges in eScience*
- ✔ *ECAS and EOSC*
- ✔ *Introduction to the Ophidia HPDA Framework*
- ✔ *Ophidia core concepts: architecture, data model, operators and primitives*
- ✔ *Analytics workflows with Ophidia*
  - ✔ *Workflow execution demo*
- ✔ *Ophidia Python bindings: PyOphidia*

Multi-model experiment design

Precipitation Trend Analysis use case implemented as an Ophidia workflow

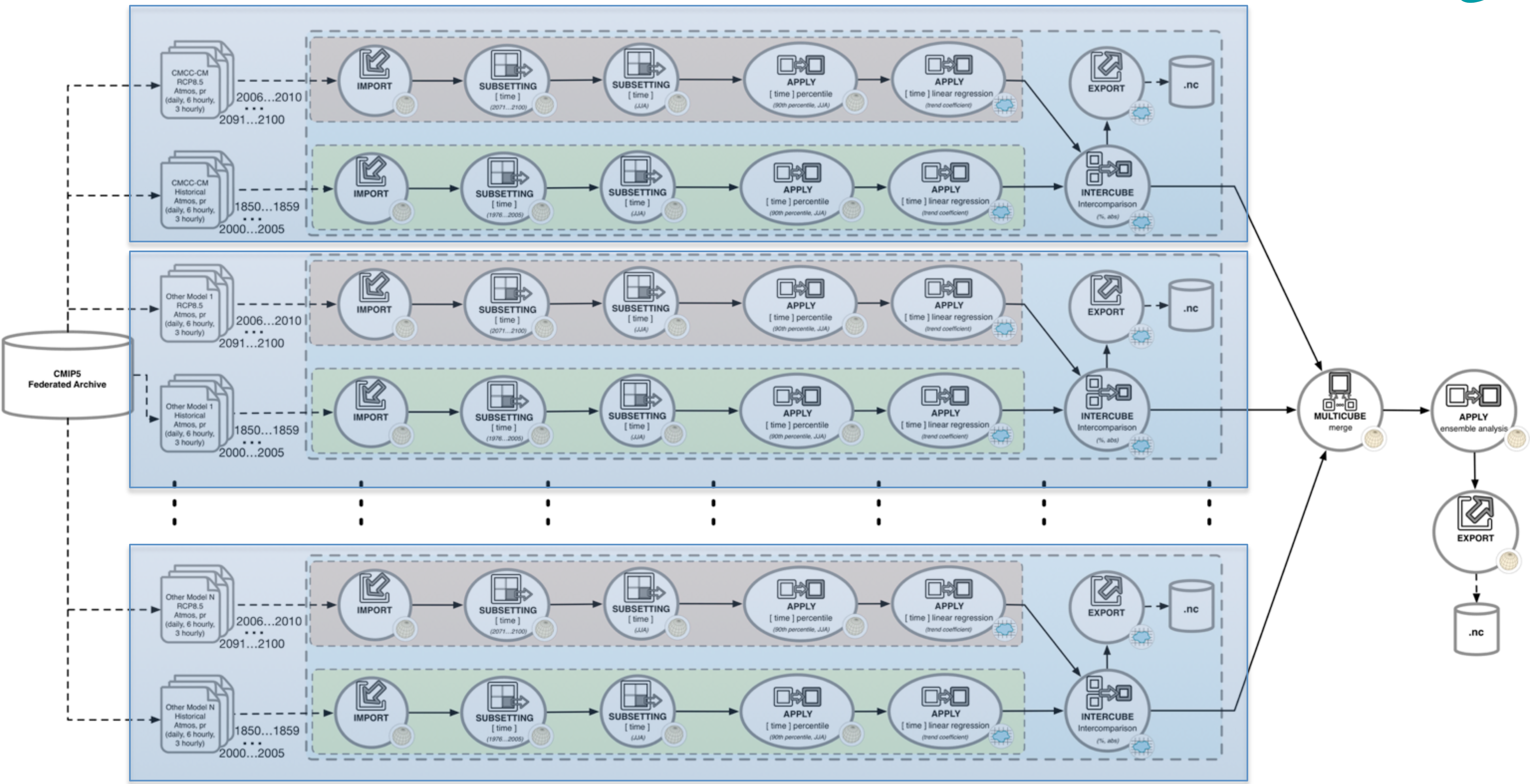


S. Fiore, et al., "Distributed and cloud-based multi-model analytics experiments on large volumes of climate change data in the earth system grid federation eco-system". In Big Data (Big Data), 2016 IEEE Int. Conference on. IEEE, 2016. pp. 2911-2918

Multi-model experiment design

Precipitation Trend Analysis use case implemented as an Ophidia workflow

Single model precipitation trend analysis



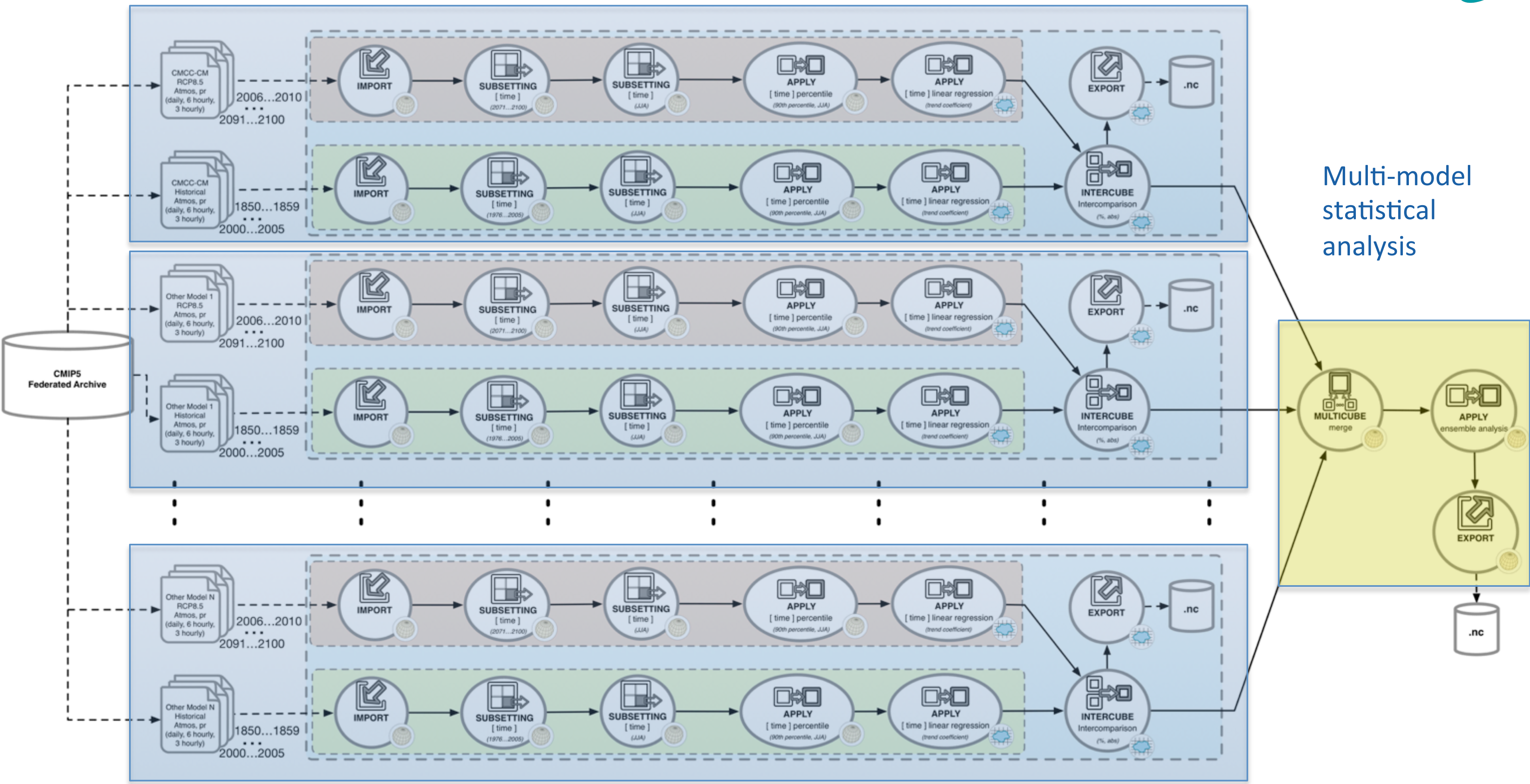
S. Fiore, et al., "Distributed and cloud-based multi-model analytics experiments on large volumes of climate change data in the earth system grid federation eco-system". In Big Data (Big Data), 2016 IEEE Int. Conference on. IEEE, 2016. pp. 2911-2918

Multi-model experiment design

Precipitation Trend Analysis use case implemented as an Ophidia workflow

Single model precipitation trend analysis

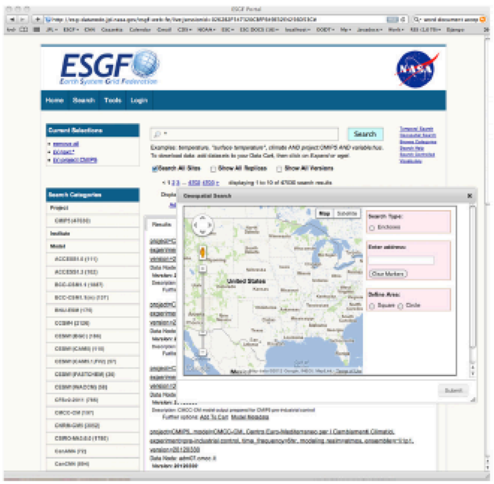
Multi-model statistical analysis

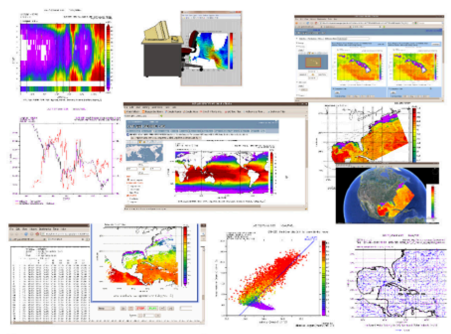


S. Fiore, et al., "Distributed and cloud-based multi-model analytics experiments on large volumes of climate change data in the earth system grid federation eco-system". In Big Data (Big Data), 2016 IEEE Int. Conference on. IEEE, 2016. pp. 2911-2918

Multi-model experiment input data

ESGF[1] is a coordinated multiagency, international collaboration of institutions that continually develop, deploy, and maintain software needed to facilitate and empower the study of climate.



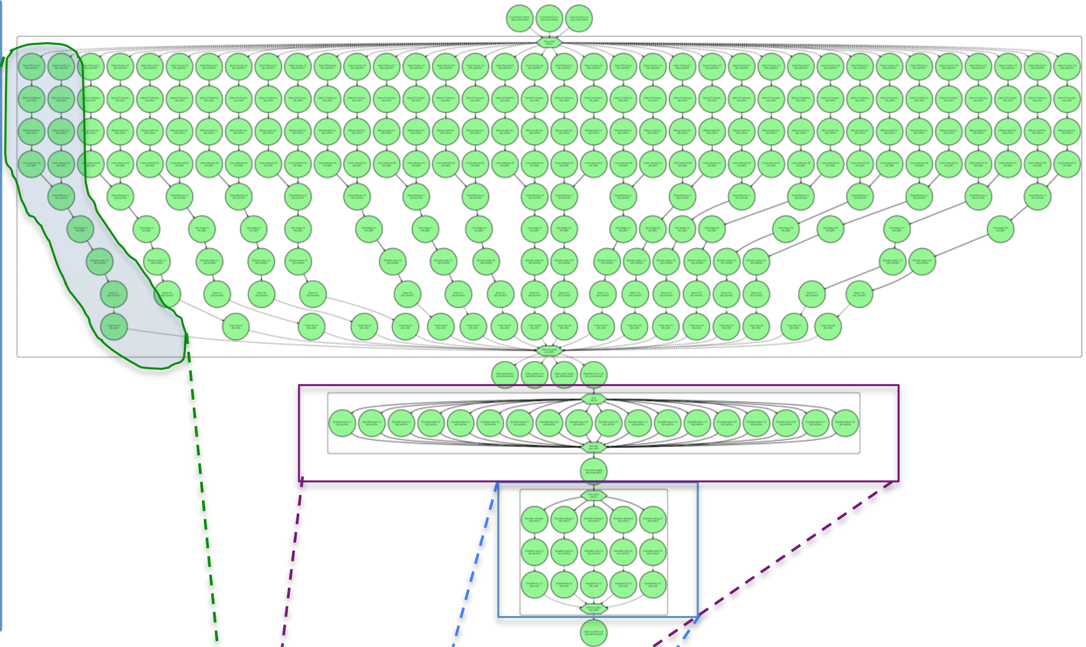


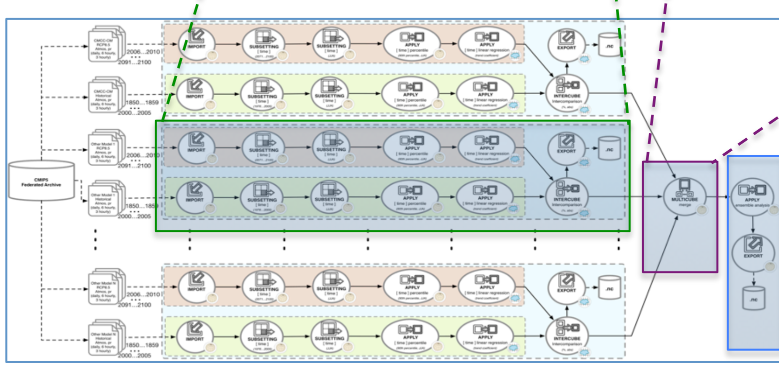
Model acronym	Model expansion	Institute
CCSM4	Community Climate System Model, v4	National Center for Atmospheric Research (NCAR)
CMCC-CESM	CMCC - Community Earth System Model	Euro-Mediterranean Center on Climate Change (CMCC)
CMCC-CMS	CMCC - Coupled Modeling System	Euro-Mediterranean Center on Climate Change (CMCC)
CMCC-CM	CMCC - Climate Model	Euro-Mediterranean Center on Climate Change (CMCC)
CNRM-CM5	CNRM - Coupled Global Climate Model, v5	Centre National de Recherches Météorologiques (CNRM)/Centre Européen de Recherche et de Formation Avancée en Calcul Scientifique (CERFACS)
CSIRO Mk3.6.0	CSIRO Mark, v3.6.0	Commonwealth Scientific and Industrial Research Organisation (CSIRO) in collaboration with Queensland Climate-Change Centre of Excellence (QCCCE)
CanESM2	Second Generation Canadian Earth System Model	Canadian Centre for Climate Modelling and Analysis (CC-Cma)
GFDL-CM3	GFDL Climate Model, v3	National Oceanic and Atmospheric Administration (NOAA)/Geophysical Fluid Dynamics Laboratory (GFDL)
GFDL-ESM2G	GFDL Earth System Model with Generalized Ocean Layer Dynamics (GOLD) component	National Oceanic and Atmospheric Administration (NOAA)/Geophysical Fluid Dynamics Laboratory (GFDL)
GFDL-ESM2M	GFDL Earth System Model with Modular Ocean Model 4 (MOM4) component	National Oceanic and Atmospheric Administration (NOAA)/Geophysical Fluid Dynamics Laboratory (GFDL)
HadGEM2-CC	Hadley Centre Global Environment Model, v2 (Carbon Cycle)	Met Office (UKMO) Hadley Centre (HC)
HadGEM2-ES	Hadley Centre Global Environment Model, v2 (Earth System)	Met Office (UKMO) Hadley Centre (HC)
INM-CM4.0	INM Coupled Model, v4.0	Institute of Numerical Mathematics (INM)
IPSL-CM5A-MR	IPSL Coupled Model, version 5, coupled with NEMO, mid resolution	L'Institut Pierre-Simon Laplace (IPSL)
MIROC5	Model for Interdisciplinary Research on Climate, v5	Atmosphere and Ocean Research Institute (The University of Tokyo), National Institute for Environmental Studies, and Japan Agency for Marine-Earth Science and Technology
MPI-ESM-MR	MPI Earth System Model, medium resolution	Max Planck Institute for Meteorology (MPI-M)
MRI-CGCM3	MRI Coupled Atmosphere - Ocean General Circulation Model, v3	Meteorological Research Institute (MRI)
NorESM1-M	Norwegian Earth System Model, v1 (intermediate resolution)	Norwegian Climate Centre (NCC)

Multi-model experiment implementation & execution

JSON implementation of the workflow

```
{
    "name": "90th percentile JJA Historical",
    "operator": "oph_reduce2",
    "arguments": [
        "operation=quantile",
        "dim=time",
        "concept_level=y",
        "order=${5}"
    ],
    "dependencies": [
        { "task": "Subset JJA Historical", "type": "single" }
    ]
},
{
    "name": "Linear regression Historical",
    "operator": "oph_apply",
    "arguments": [
        "query=oph_gsl_fit_linear_coeff(measure)",
        "measure_type=auto"
    ],
    "dependencies": [
        { "task": "90th percentile JJA Historical", "type": "single" }
    ]
},
{
    "name": "Import Type Selection Scenario",
    "operator": "oph_if",
    "arguments": [ "condition=${10}" ],
    "dependencies": [
        { "task": "loop_model" }
    ]
},
```

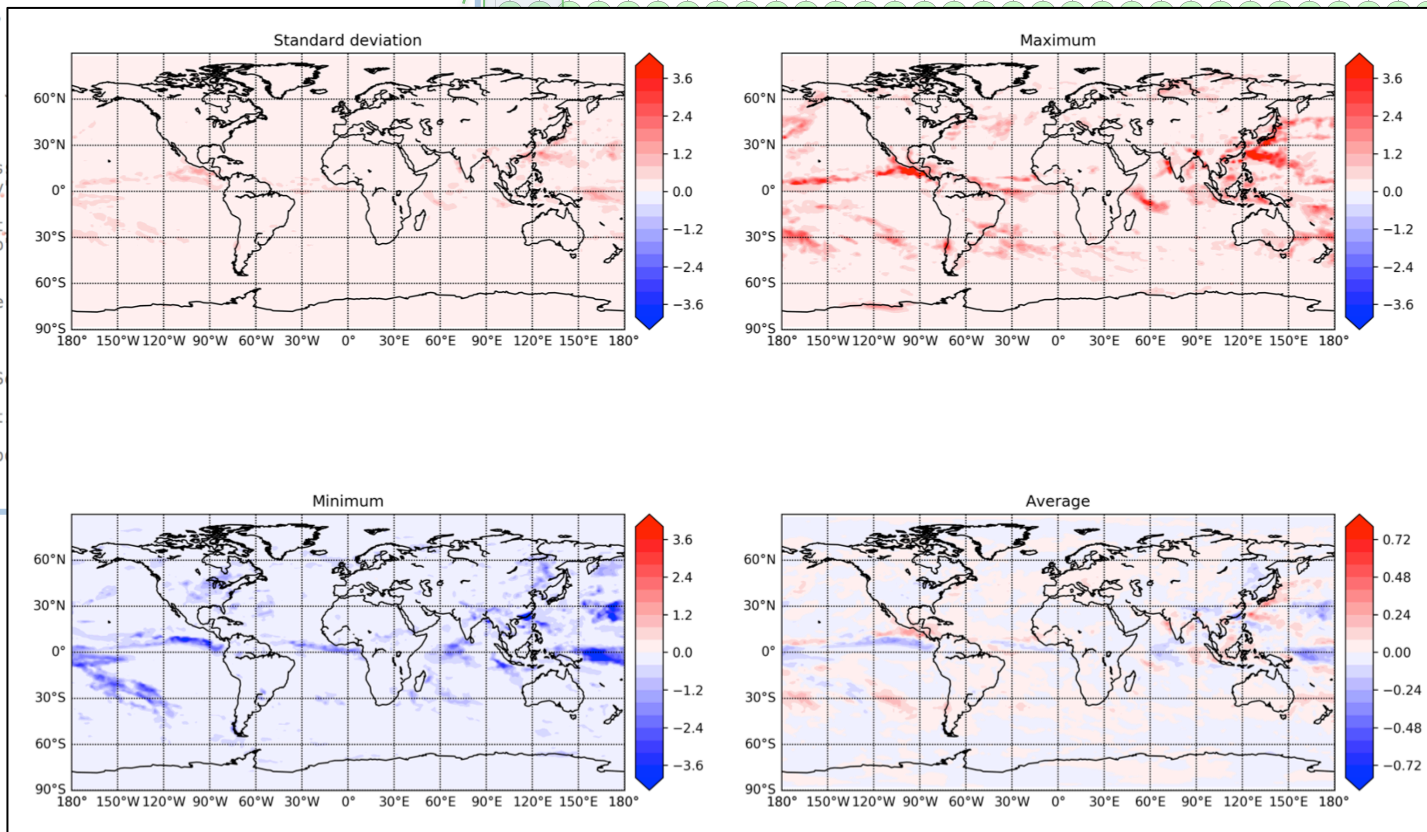


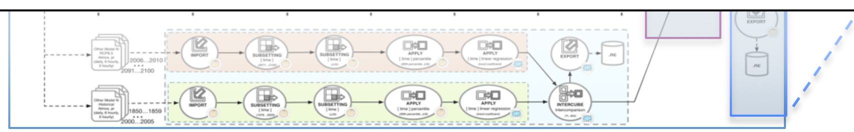


Multi-model experiment implementation & execution

JSON implementation of the workflow

```
{
    "name": "90th percentile JJA Historical",
    "operator": "oph_reduce2",
    "arguments": [
        "operation=quantile",
        "dim=time",
        "concept_level=y",
        "order=${5}"
    ],
    "dependencies": [
        { "task": "Subset
    ]
},
{
    "name": "Linear regres
    "operator": "oph_apply
    "arguments": [
        "query=oph_gsl_fit
        "measure_type=auto
    ],
    "dependencies": [
        { "task": "90th pe
    ]
},
{
    "name": "Import Type S
    "operator": "oph_if",
    "arguments": [ "condit
    "dependencies": [
        { "task": "loop_mo
    ]
},
```





Session outline

- ✔ *Introduction to HPDA and data challenges in eScience*
- ✔ *ECAS and EOSC*
- ✔ *Introduction to the Ophidia HPDA Framework*
- ✔ *Ophidia core concepts: architecture, data model, operators and primitives*
- ✔ *Analytics workflows with Ophidia*
  - ✔ *Workflow execution demo*
- ✔ *Ophidia Python bindings: PyOphidia*

Session outline

- ✔ *Introduction to HPDA and data challenges in eScience*
- ✔ *ECAS and EOSC*
- ✔ *Introduction to the Ophidia HPDA Framework*
- ✔ *Ophidia core concepts: architecture, data model, operators and primitives*
- ✔ *Analytics workflows with Ophidia*
  - ✔ *Workflow execution demo*
- ✔ *Ophidia Python bindings: PyOphidia*

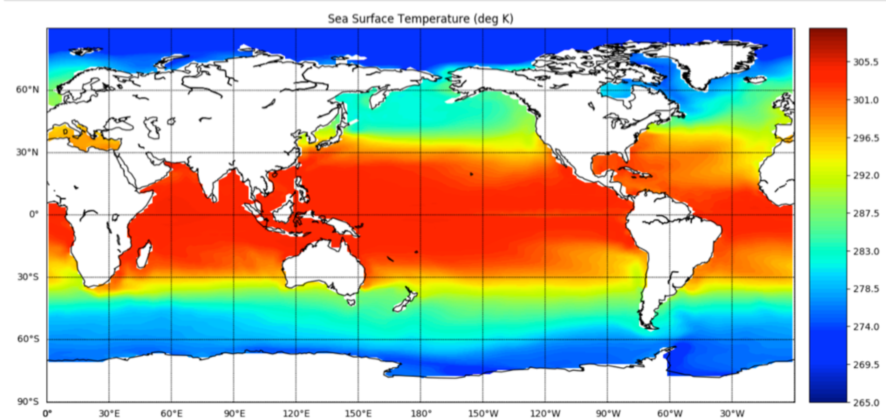
Python programmatic access to Ophidia

PyOphidia is a GPLv3-licensed Python module to interact with the Ophidia framework.

It provides a programmatic access to Ophidia features, allowing:

- Submission of commands to the Ophidia Server and retrieval of the results
- Management of (remote) data objects in the form of datacubes
- Easy exploitation from Jupyter Notebooks and integration with other Python modules

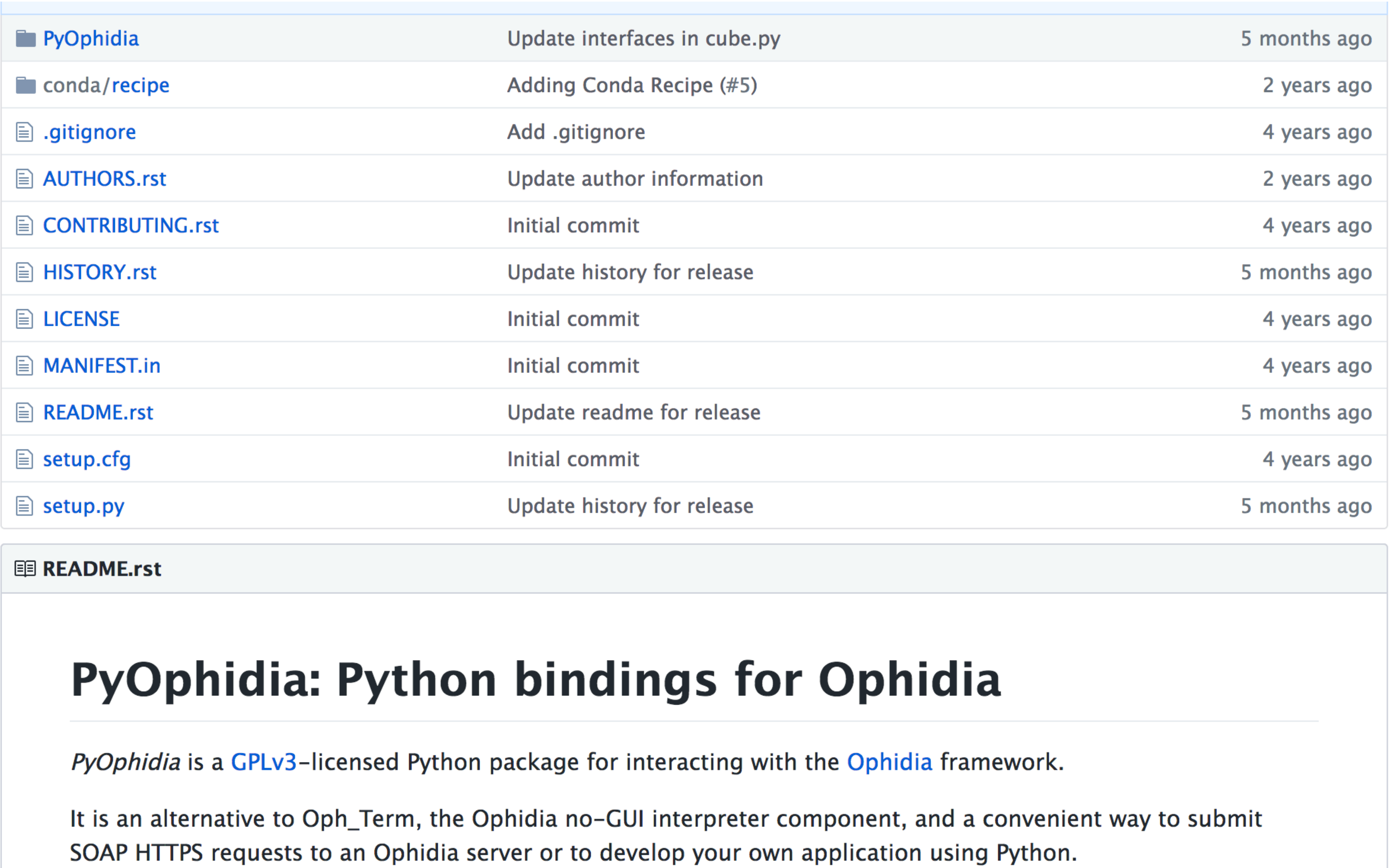
```python
from PyOphidia import cube, client
cube.Cube.setclient(read_env=True)

mycube =
cube.Cube.importnc(src_path='/public/data/ecas_training
/file.nc', measure='tos', imp_dim='time',
import_metadata='yes', ncores=5)
mycube2 = mycube.reduce(operation='max',ncores=5)
mycube3 = mycube2.rollup(ncores=5)
data = mycube3.export_array()

mycube3.exportnc2(output_path='/home/test',
export_metadata='yes')
```



Export result to NetCDF file

```python
]: mycube3.exportnc2(output_path='/home/' + cube.Cube.client.username,export_metadata='yes')
```

PyOphidia Repository



PyOphidia	Update interfaces in cube.py	5 months ago
conda/recipe	Adding Conda Recipe (#5)	2 years ago
.gitignore	Add .gitignore	4 years ago
AUTHORS.rst	Update author information	2 years ago
CONTRIBUTING.rst	Initial commit	4 years ago
HISTORY.rst	Update history for release	5 months ago
LICENSE	Initial commit	4 years ago
MANIFEST.in	Initial commit	4 years ago
README.rst	Update readme for release	5 months ago
setup.cfg	Initial commit	4 years ago
setup.py	Update history for release	5 months ago

README.rst

PyOphidia: Python bindings for Ophidia

PyOphidia is a GPLv3-licensed Python package for interacting with the Ophidia framework.

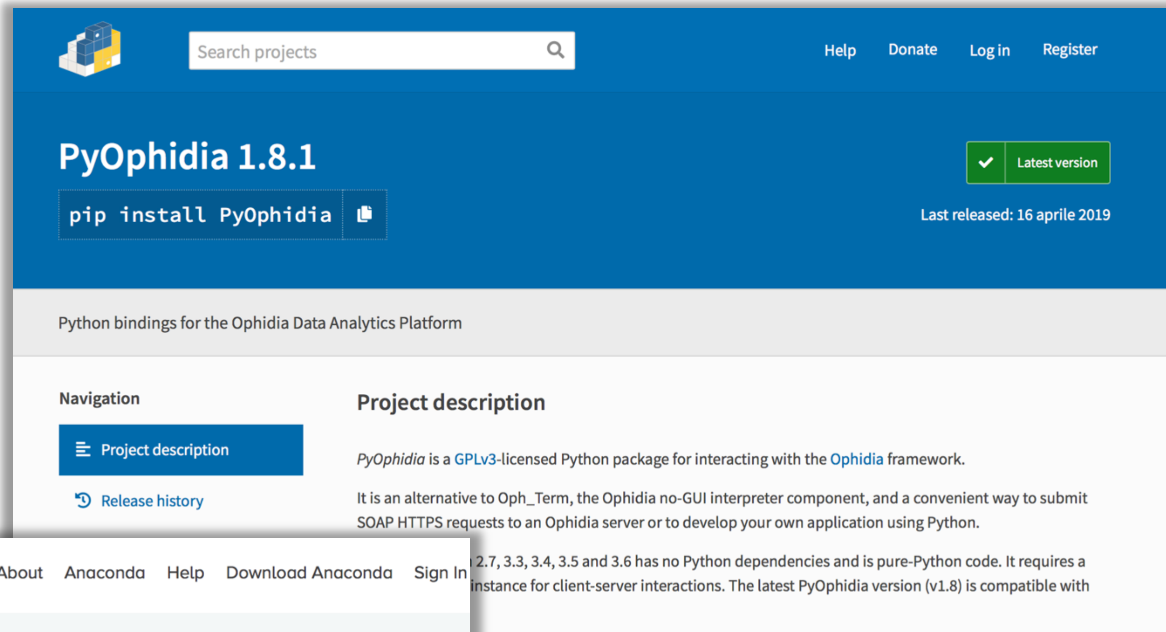
It is an alternative to Oph_Term, the Ophidia no-GUI interpreter component, and a convenient way to submit SOAP HTTPS requests to an Ophidia server or to develop your own application using Python.

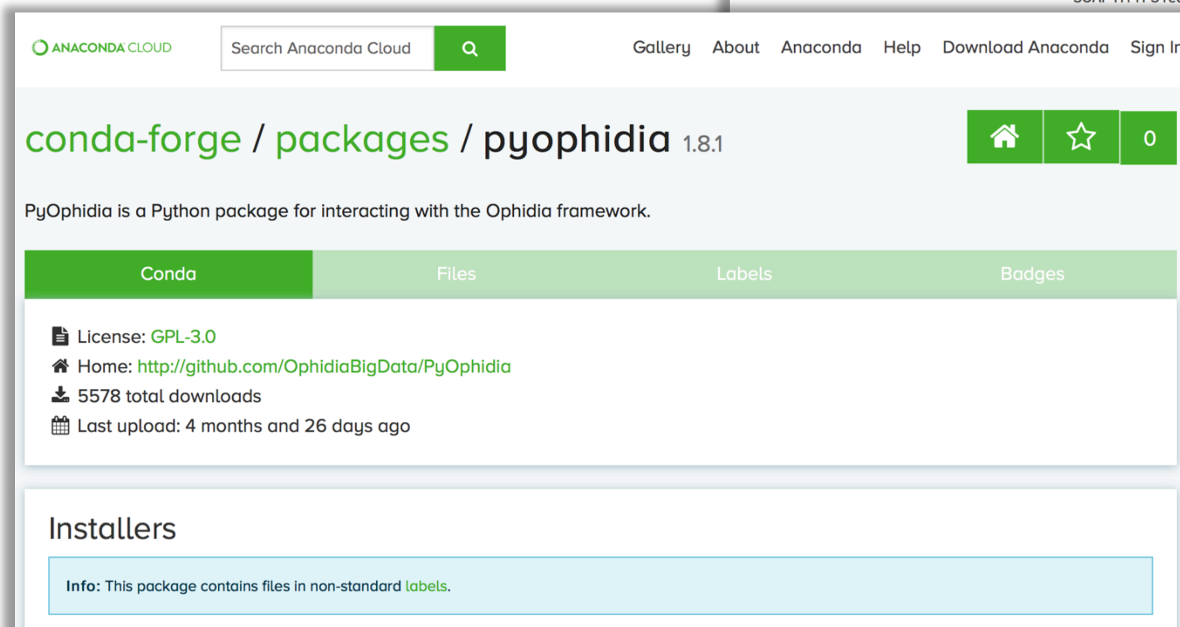
PyOphidia on Github: https://github.com/OphidiaBigData/PyOphidia

PyOphidia Repository

https://pypi.org/project/PyOphidia/

```
pip3 install pyophidia
```





https://anaconda.org/conda-forge/pyophidia

```
conda install -c conda-forge pyophidia
```

The PyOphidia library

PyOphidia implements two main classes:

- **Client class**: supports the submissions of Ophidia commands and workflows, as well as the management of session from Python code (similar to the Ophidia Terminal)
    - It allows to run all the Ophidia operators, including massive tasks and workflows
- **Cube class**: provides the datacube type abstraction and the methods to manipulate, process and get information on cubes objects and it builds on the client class
    - Defines a object-oriented approach allowing a handle the datacubes more naturally

While the cube module provides a user-friendly interface, the client module allows a finer specification of the operators.

Python and HPC infrastructure transparency

PyOphidia class hides the HPC environment complexity

```
In [ ]: from PyOphidia import cube, client
        cube.Cube.setclient(read_env=True)

In [ ]: cube.Cube.cluster(action='deploy',host_partition='test_partition',nhost=4)

In [ ]: myCube = cube.Cube(src_path='/work/ophidia/tests/tasmax_day_CMCC-CESM_rcp85.nc',
            measure='tasmax', import_metadata='yes', imp_dim='time', description='Max Temps',
            nfrag=16, nhosts=4,
            host_partition='test2',
            ncores=2, nthreads=8
            )

In [ ]: myCube2 = maxtemp.apply(
            query="oph_predicate('oph_float','oph_int',measure,'x-298.15','>0','1','0')",
            ncores=2, nthreads=8
        )

In [ ]: myCube3 = myCube2.subset(subset_filter=1, subset_dims='time')

In [ ]: pythonData = myCube3.export_array(show_time='yes')

In [ ]: print(pythonData)

In [ ]: cube.Cube.cluster(action='undeploy',host_partition='test_partition')
```

Python and HPC infrastructure transparency

PyOphidia class hides the HPC environment complexity

```
In [ ]: from PyOphidia import cube, client
        cube.Cube.setclient(read_env=True)

In [ ]: cube.Cube.cluster(action='deploy',host_partition='test_partition',nhost=4)

In [ ]: myCube = cube.Cube(src_path='/work/ophidia/tests/tasmax_day_CMCC-CESM_rcp85.nc',
            measure='tasmax', import_metadata='yes', imp_dim='time', description='Max Temps',
            nfrag=16, nhosts=4,
            host_partition='test2',
            ncores=2, nthreads=8
            )

In [ ]: myCube2 = maxtemp.apply(
            query="oph_predicate('oph_float','oph_int',measure,'x-298.15','>0','1','0')",
            ncores=2, nthreads=8
        )

In [ ]: myCube3 = myCube2.subset(subset_filter=1, subset_dims='time

In [ ]: pythonData = myCube3.export_array(show_time='yes')

In [ ]: print(pythonData)

In [ ]: cube.Cube.cluster(action='undeploy',host_partition='test_partition')
```

Dynamic I/O & Analytics nodes allocation

Data partitioning and distribution

Framework operator parallelism

Ophidia-notebook data translation and transfer

I/O & Analytics nodes undeployment

Summary

- ✔ *Joining HPC and data-intensive analytics is an enabling factor for scientific applications*
- ✔ *Scientific data management and analytics pose challenges requiring novel and efficient software solution*
- ✔ *ECAS: a solutions for server-side, parallel data analysis in the EOSC landscape*
- ✔ *In-depth overview of the Ophidia HPDA framework and how it addresses data analytics challenges for scientific analysis*
  - *Scalable architecture, data distribution, parallel operators and HPC-oriented deployment*
- ✔ *Real experiments can be modeled as (complex) workflows composed of hundreds of tasks*
  - *Multi-model climate analysis example*

References and further readings (1)

- *Laney D (2001) 3-d data management: controlling data volume, velocity and variety. META Group Research Note, 6 February.*
- *Chen, M., Mao, S., & Liu, Y. (2014). Big data: A survey. Mobile networks and applications, 19(2), 171-209.*
- *Chen, C. P., & Zhang, C. Y. (2014). Data-intensive applications, challenges, techniques and technologies: A survey on Big Data. Information sciences, 275, 314-347.*
- *D. A. Reed and J. Dongarra. (2015). Exascale computing and big data. Commun. ACM 58, 7 (July 2015), 56–68.*
- *Jha, S., Qiu, J., Luckow, A., Mantha, P., & Fox, G. C. (2014). A tale of two data-intensive paradigms: Applications, abstractions, and architectures. In 2014 IEEE Int. Congress on Big Data, 645-652.*
- *Asch, M., et al. (2018). Big data and extreme-scale computing: Pathways to convergence-toward a shaping strategy for a future software and data ecosystem for scientific inquiry. Int. J. High Perform. Comput. Appl., 32(4), 435-479.*
- *Luca Cinquini, et al. (2014). The Earth System Grid Federation: An open infrastructure for access to distributed geospatial data. Future Gener. Comput. Syst. 36: 400-417.*
- *GMD topical editors (Eds.), V. Eyring (coordinator) (2012). Coupled Model Intercomparison Project Phase 6 (CMIP6) Experimental Design and Organization [Special Issue]. Geosci. Model Dev.* https://gmd.copernicus.org/articles/special_issue590.html
- *G. Aloisio, S. Fiore, I. Foster, D. Williams (2013). Scientific big data analytics challenges at large scale. Big Data and Extreme-scale Computing (BDEC), April 30 to May 01, 2013, Charleston, South Carolina, USA (position paper).*
- *S. Fiore, D. Elia, C. Palazzo, A. D'Anca, F. Antonio, D. N. Williams, I. Foster, G. Aloisio, "Towards an Open (Data) Science Analytics-Hub for Reproducible multi-model Climate Analysis at Scale", 2018 IEEE Int. Conference on Big Data, pp. 3226-3234.*

References and further readings (2)

- *S. Fiore, A. D'Anca, C. Palazzo, I. T. Foster, D. N. Williams, G. Aloisio (2013). Ophidia: Toward Big Data Analytics for eScience. ICCS 2013, volume 18 of Procedia Computer Science, pp. 2376-2385.*
- *S. Fiore, A. D'Anca, D. Elia, C. Palazzo, I. Foster, D. Williams, G. Aloisio (2014). "Ophidia: A Full Software Stack for Scientific Data Analytics", proc. of the 2014 Int. Conference on High Performance Computing & Simulation (HPCS 2014), pp. 343-350.*
- *S. Fiore, D. Elia, C. Palazzo, F. Antonio, A. D'Anca, I. Foster and G. Aloisio (2019), "Towards High Performance Data Analytics for Climate Change", ISC High Performance 2019. Lecture Notes in Computer Science, vol. 11887, pp. 240-257.*
- *D. Elia, S. Fiore, A. D'Anca, C. Palazzo, I. Foster, D. N. Williams, G. Aloisio (2016). "An in-memory based framework for scientific data analytics". In Proc. of the ACM Int. Conference on Computing Frontiers (CF '16), pp. 424-429.*
- *C. Palazzo, A. Mariello, S. Fiore, A. D'Anca, D. Elia, D. N. Williams, G. Aloisio (2015), "A Workflow-Enabled Big Data Analytics Software Stack for eScience", HPCS 2015, pp. 545-552*
- *A. D'Anca, et al. (2017), "On the Use of In-memory Analytics Workflows to Compute eScience Indicators from Large Climate Datasets," 2017 17th IEEE/ACM Int. Symposium on Cluster, Cloud and Grid Computing (CCGRID), pp. 1035-1043.*
- *S. Fiore, et al. (2016). "Distributed and cloud-based multi-model analytics experiments on large volumes of climate change data in the earth system grid federation eco-system". In Big Data (Big Data), 2016 IEEE Int. Conference on. IEEE. pp. 2911-2918.*

Thank you!

These activities are supported in part by ESiWACE2, EOSC-Hub and IS-ENES3 projects:

esiwace
CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER AND CLIMATE IN EUROPE

ESiWACE2 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823988.

EOSC-hub

EOSC-hub receives funding from the EU's Horizon 2020 research and innovation programme under grant agreement No. 777536.

is-enes
INFRASTRUCTURE FOR THE EUROPEAN NETWORK FOR EARTH SYSTEM MODELLING

IS-ENES3 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824084

Ophidia website: http://ophidia.cmcc.it

Contact: donatello.elia AT cmcc.it

cmcc
Centro Euro-Mediterraneo sui Cambiamenti Climatici