

Ophidia: a big data analytics framework for eScience

Dr. Sandro Fiore

Advanced Scientific Computing Division @CMCC

2014 ESGF & UV-CDAT F2F Meeting

Livermore, December 11, 2014

On behalf of the Ophidia Team

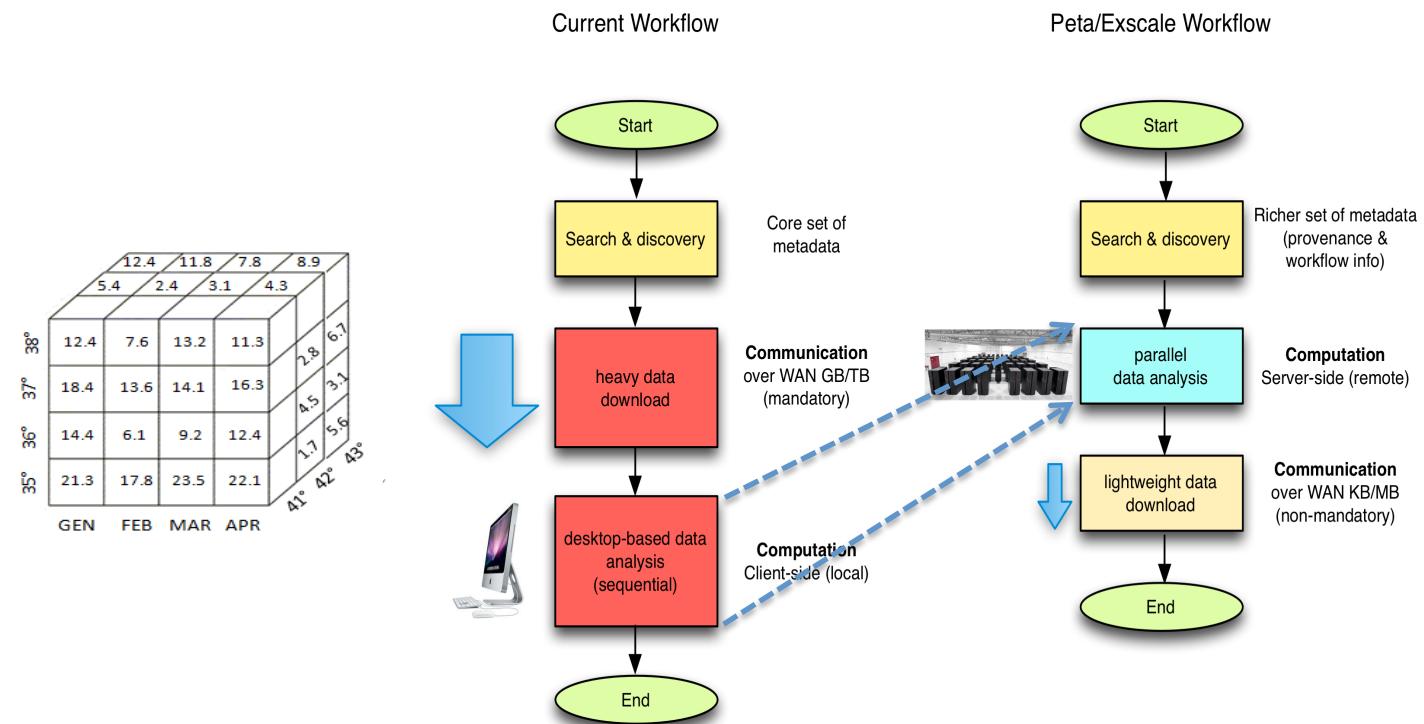
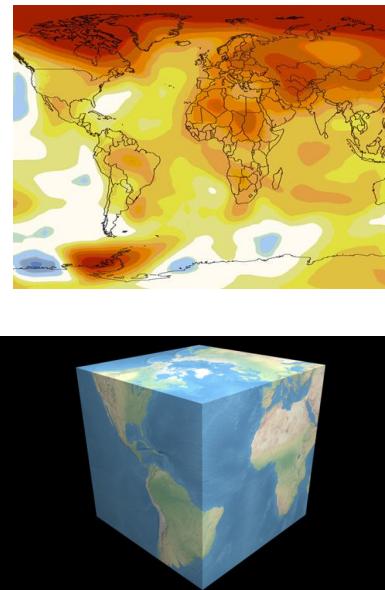
S. Fiore, D'Anca, D. Elia, A. Mariello, C. Palazzo,
P. Nassisi, G. Aloisio



cmcc
Centro Euro-Mediterraneo
sui Cambiamenti Climatici

Big data analytics for climate change

Ophidia is a research effort carried out at the **Euro Mediterranean Centre on Climate Change (CMCC)** to address *big data* challenges, issues and requirements for climate change data analytics



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

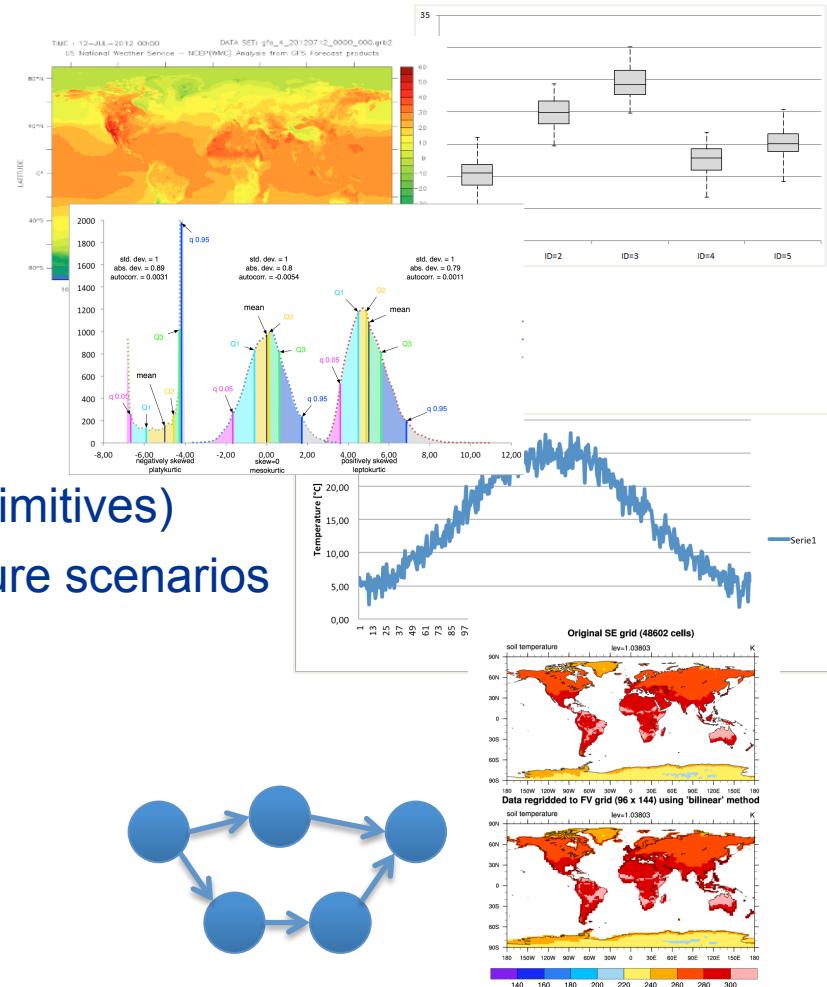
A set of requirements have been discussed with the CMCC Divisions to identify the key data analytics needs

Preliminary requirements and needs focus on:

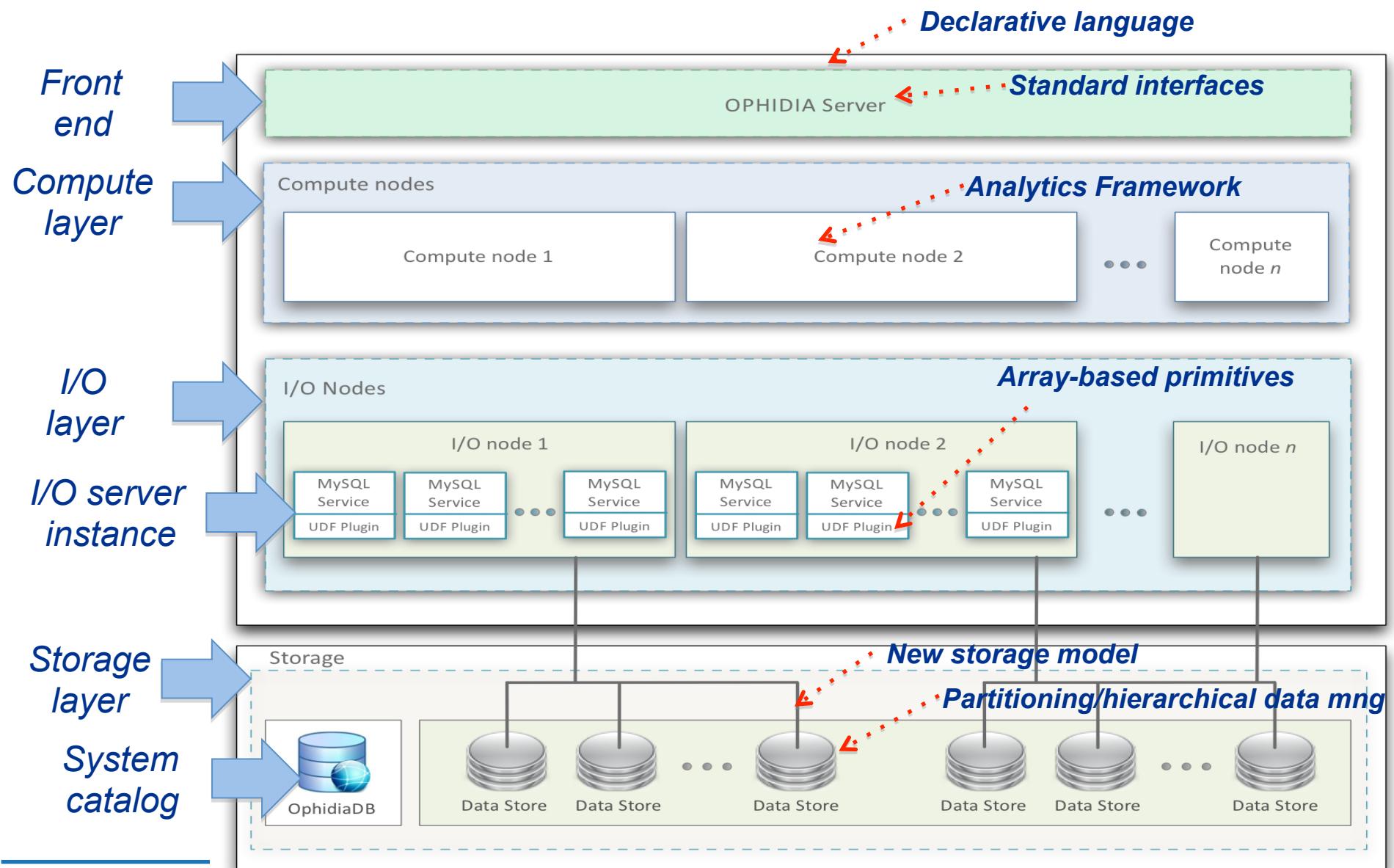
- ❖ Time series analysis
- ❖ Data reduction (e.g. by aggregation)
- ❖ Data subsetting
- ❖ Model intercomparison
- ❖ Multimodel means
- ❖ Data transformation (through array-based primitives)
- ❖ Comparison between historical data and future scenarios
- ❖ Maps generation
- ❖ Ensemble analysis
- ❖ Data analytics workflow support

But also...

- ❖ Performance,
- ❖ re-usability and extensibility

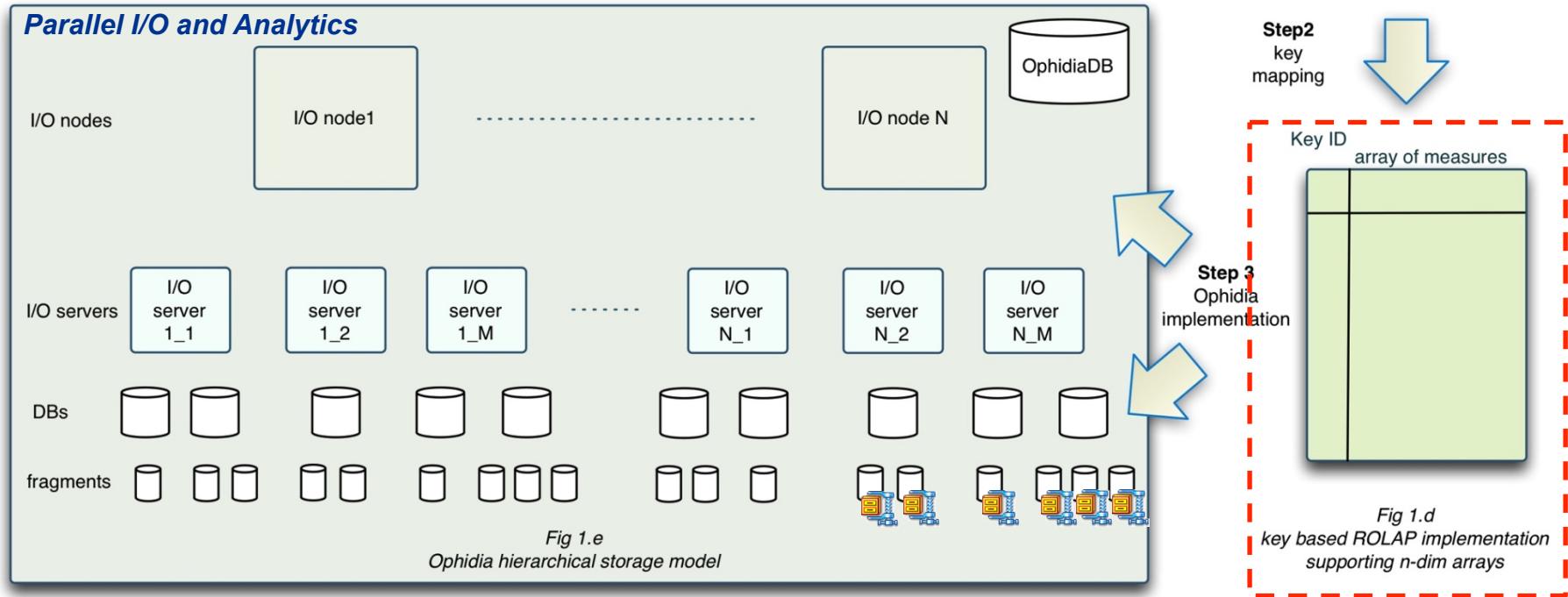
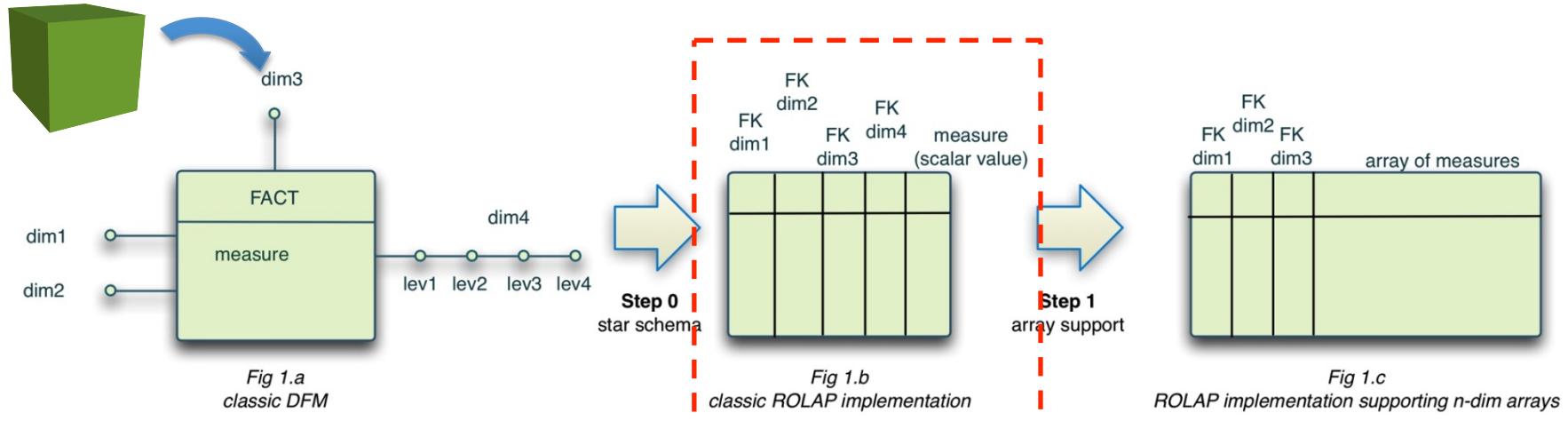


Ophidia Architecture 1.0



Storage model (dimension-independent) & implementation

Array-based support and hierarchical storage



Array based primitives

- *The array data type support is not enough to provide scientific data management capabilities... primitives are needed as well.*
- *A set of array-based primitives have been implemented*
- *A primitive is applied to a single array*
- *They come in the form of plugins (I/O server extensions)*
- *So far, Ophidia provides a wide set of plugins (about 100) to perform data reduction (by aggregation), sub-setting, predicates evaluation, statistical analysis, compression, and so forth.*
 - *Support is provided both for byte-oriented and bit-oriented arrays*
 - *Plugins can be nested to get more complex functionalities*
 - *Compression is provided as a primitive too*



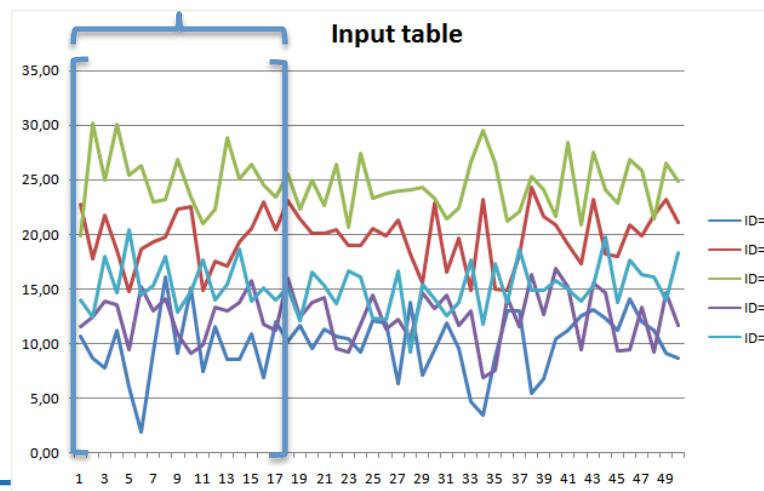
Array based primitives: nesting feature

`oph_boxplot(oph_subarray(oph_uncompress(measure), 1,18), "OPH_DOUBLE")`

Single chunk or fragment (input)

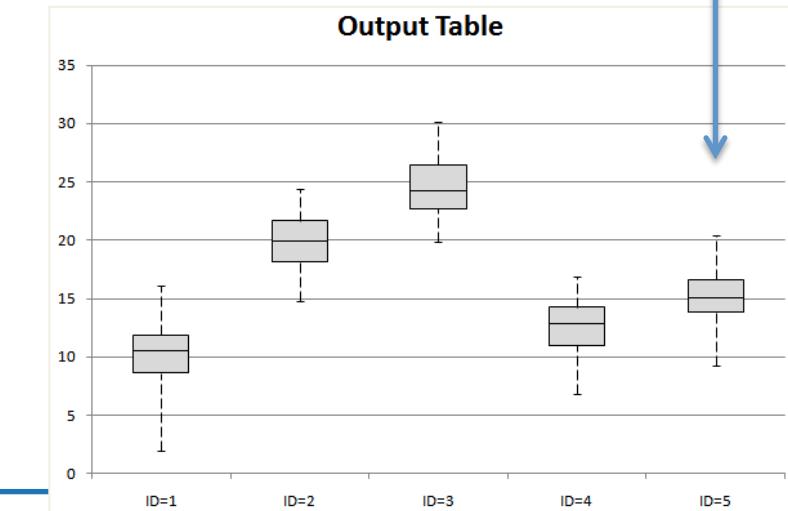
INPUTTABLE 5 tuples x 50 elements	
ID	MEASURE
1	10,73 8,66 7,83 11,20 6,02 1,95 ... 16,11 ... 8,70
2	22,85 17,84 21,82 18,57 14,81 18,71 ... 19,83 ... 21,13
3	19,89 30,17 24,95 30,07 25,40 26,31 ... 23,18 ... 24,82
4	11,60 12,49 13,91 13,53 9,48 15,27 ... 14,17 ... 11,66
5	13,94 12,43 17,95 14,70 20,41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

`subarray(measure, 1,18)`

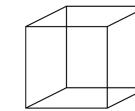
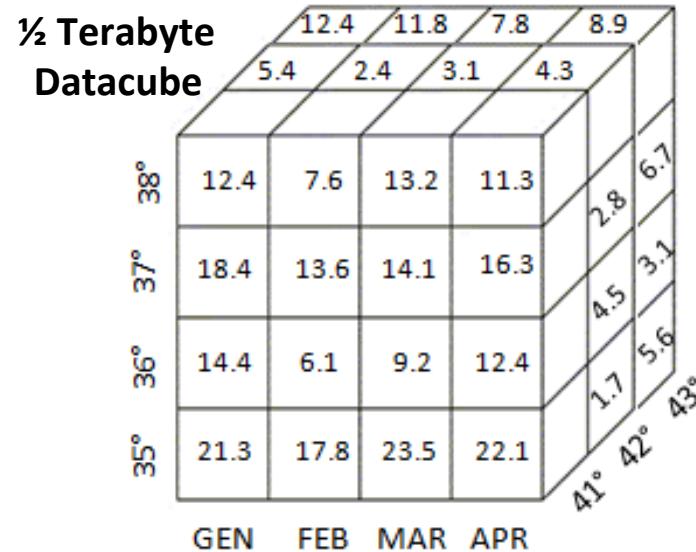


Single chunk or fragment (output)

OUTPUTTABLE 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



Mapping a dataset into the Ophidia back-end



FRAGMENTS – 10^4 TUPLE x 10^5 ELEMENTS

FRAGMENT4 – 10⁴ TUPLE x 10⁵ ELEMENTS

FRAG64 – 10⁴ TUPLE x 10⁵ ELEMENTS

ID MEASURE

ID	MEASURE
1	1,95 8,64 ... 16,11
2	14,81 18,14 ... 24,35
...	...
10	...
10 ⁴	6,87 10,99 ... 16,93

x64



FRAGMENT1 – 1 TUPLE x 1 ELEM.

ID MEASURE

1	24,35
---	-------

x1



The analytics framework: datacube operators

OPERATOR NAME	OPERATOR DESCRIPTION
Operators “Data processing” – Domain-agnostic	
OPH_APPLY(<i>datacube_in</i> , <i>datacube_out</i> , <i>array based primitive</i>)	Creates the <i>datacube_out</i> by applying the <i>array-based primitive</i> to the <i>datacube_in</i>
OPH_DUPLICATE(<i>datacube_in</i> , <i>datacube_out</i>)	Creates a copy of the <i>datacube_in</i> in the <i>datacube_out</i>
OPH_SUBSET(<i>datacube_in</i> , <i>subset_string</i> , <i>datacube_out</i>)	Creates the <i>datacube_out</i> by doing a sub-setting of the <i>datacube_in</i> by applying the <i>subset string</i>
OPH_MERGE(<i>datacube_in</i> , <i>merge_param</i> , <i>datacube_out</i>)	Creates the <i>datacube_out</i> by merging groups of <i>merge_param</i> fragments from <i>datacube_in</i>
OPH_SPLIT(<i>datacube_in</i> , <i>split_param</i> , <i>datacube_out</i>)	Creates the <i>datacube_out</i> by splitting into groups of <i>split_param</i> fragments each fragment of the <i>datacube_in</i>
OPH_INTERCOMPARISON (<i>datacube_in1</i> , <i>datacube_in2</i> , <i>datacube_out</i>)	Creates the <i>datacube_out</i> which is the element-wise difference between <i>datacube_in1</i> and <i>datacube_in2</i>
OPH_DELETE(<i>datacube_in</i>)	Removes the <i>datacube_in</i>

*Data Access
(sequential and parallel operators)*

*Metadata management
(sequential and parallel operators)*

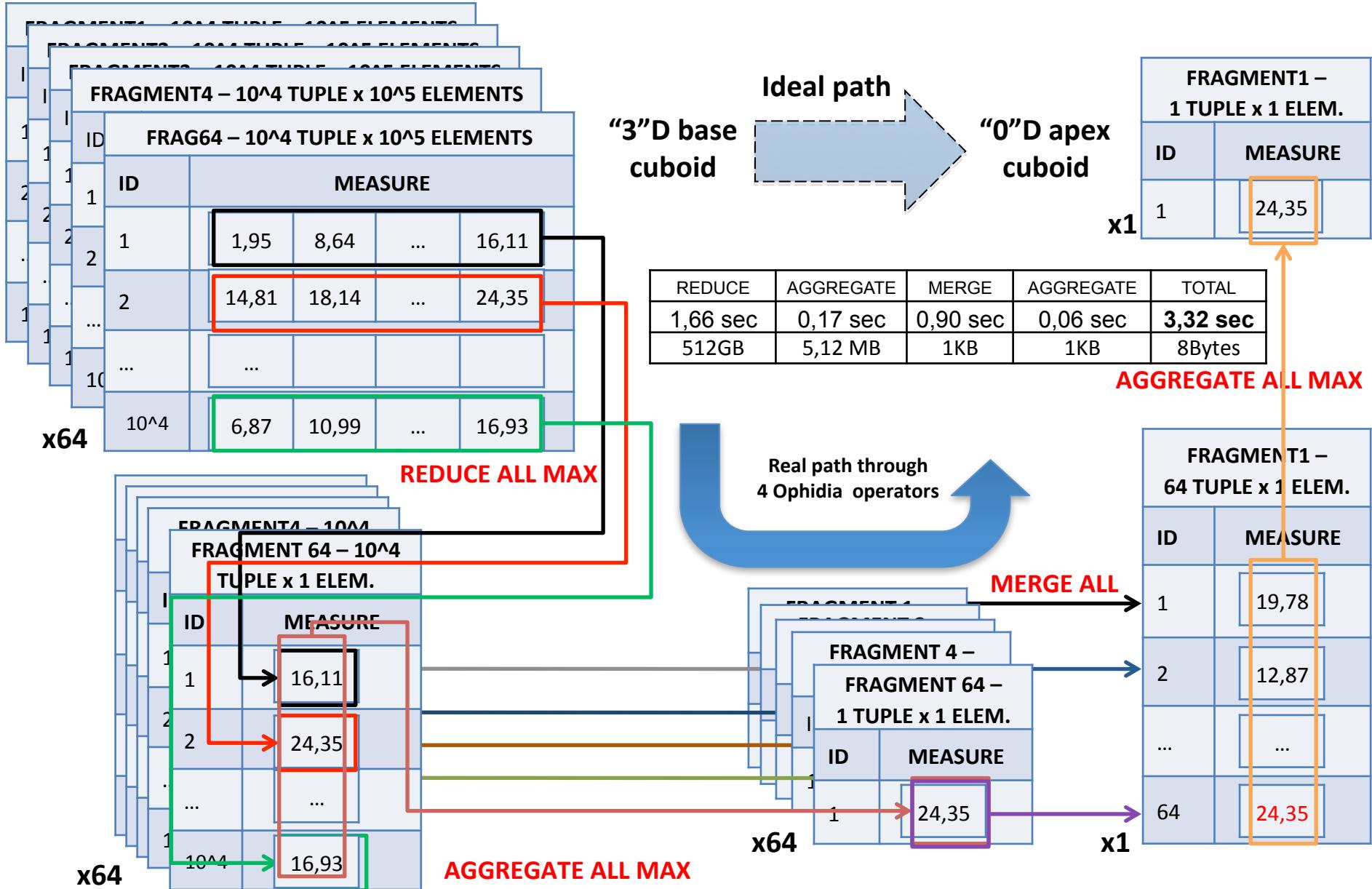
*Data processing
(parallel operators, MPI &
OpenMP based)*

*Import/Export
(parallel operators)*

OPERATOR NAME	OPERATOR DESCRIPTION
Operators “Data processing” – Domain-oriented	
OPH_EXPORT_NC (<i>datacube_in</i> , <i>file_out</i>)	Exports the <i>datacube_in</i> data into the <i>file_out</i> NetCDF file.
OPH_IMPORT_NC (<i>file_in</i> , <i>datacube_out</i>)	Imports the data stored into the <i>file_in</i> NetCDF file into the new <i>datacube_out</i>
Operators “Data access”	
OPH_INSPECT_FRAG (<i>datacube_in</i> , <i>fragment_in</i>)	Inspects the data stored in the <i>fragment_in</i> from the <i>datacube_in</i>
OPH_PUBLISH(<i>datacube_in</i>)	Publishes the <i>datacube_in</i> fragments into HTML pages
Operators “Metadata”	
OPH_CUBE_ELEMENTS (<i>datacube_in</i>)	Provides the total number of the elements in the <i>datacube_in</i>
OPH_CUBE_SIZE (<i>datacube_in</i>)	Provides the disk space occupied by the <i>datacube_in</i>
OPH_LIST(void)	Provides the list of available datacubes.
OPH_CUBEIO(<i>datacube_in</i>)	Provides the provenance information related to the <i>datacube_in</i>
OPH_FIND(<i>search_param</i>)	Provides the list of datacubes matching the <i>search_param</i> criteria



Preliminary performance insights on a massive “*in-memory*” data reduction workflow



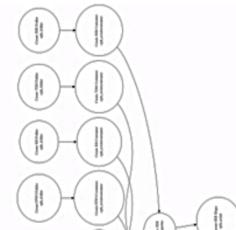
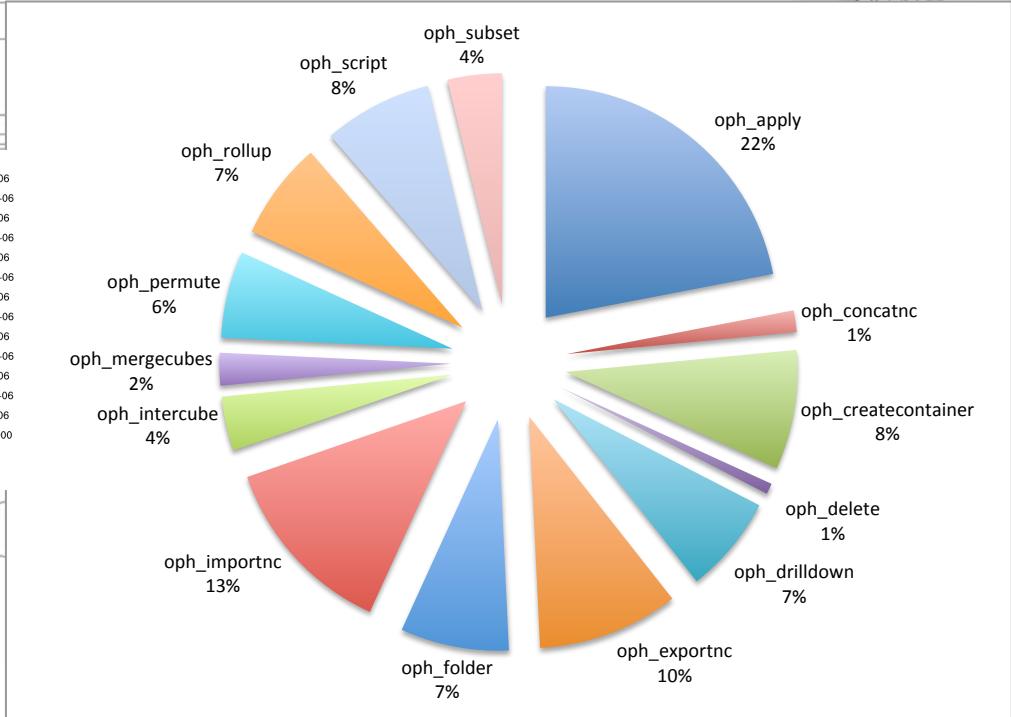
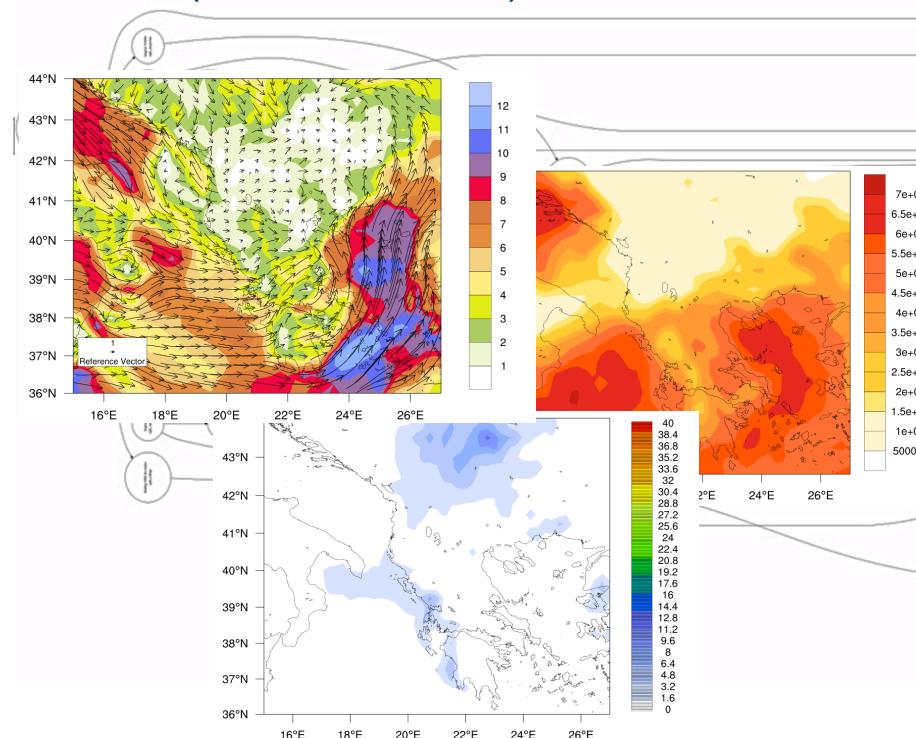
A real use case on fire danger prevention

Parallelism:

- Inter-task parallelism
- Intra-task parallelism

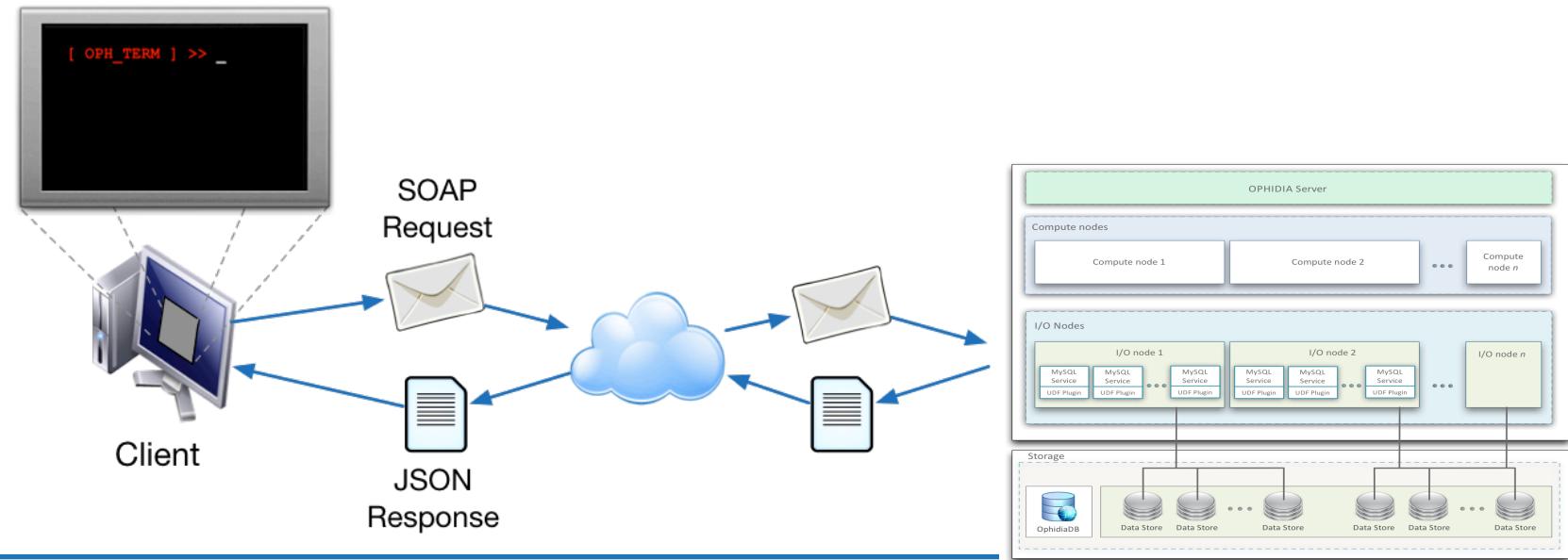
Workflow tasks:

- Data analytics, maps generation, data archiving, import/export of data (different formats)



The Ophidia Terminal: interactive environment (shell)

- The Ophidia terminal provides an effective and lightweight way to interact with the Ophidia server
- Bash-like environment (commands interpreter)
- Terminal with history management, auto-completion, specific environment variables and commands with integrated help
- Easy installation as an only one executable using a small number of well-known and open-source libraries
- More than 15 KLOC
- Simple enough for a novice and at the same time powerful enough for an expert



Terminal: (system) metadata information

```
[ OPH_TERM ] >> oph_submit "operator=cubeschema;datacube_input=container1.virtualdb1;"  
[Response]:
```

DATACUBE INFORMATION								
NAME	CREATION DATE	MEASURE	MEASURE TYPE	NUM.ELEMENTS				
virtualdb1	2013-08-09 17:21:25	Pressure	double	800				
DESCRIPTION	-							
HOSTxCUBE DBMSxHOST DBxDBMS FRAGxDB ROWxFRAG ELEMxROW COMPRESSED CUBESIZE [MB]								
1	2	2	2	10	10	yes	0.012085	

DIMENSION INFORMATION					
NAME	TYPE	SIZE	ARRAY	LEVEL	
x	double	2	no	1	
y	double	4	no	2	
z	double	10	no	3	
time	double	10	yes	1	

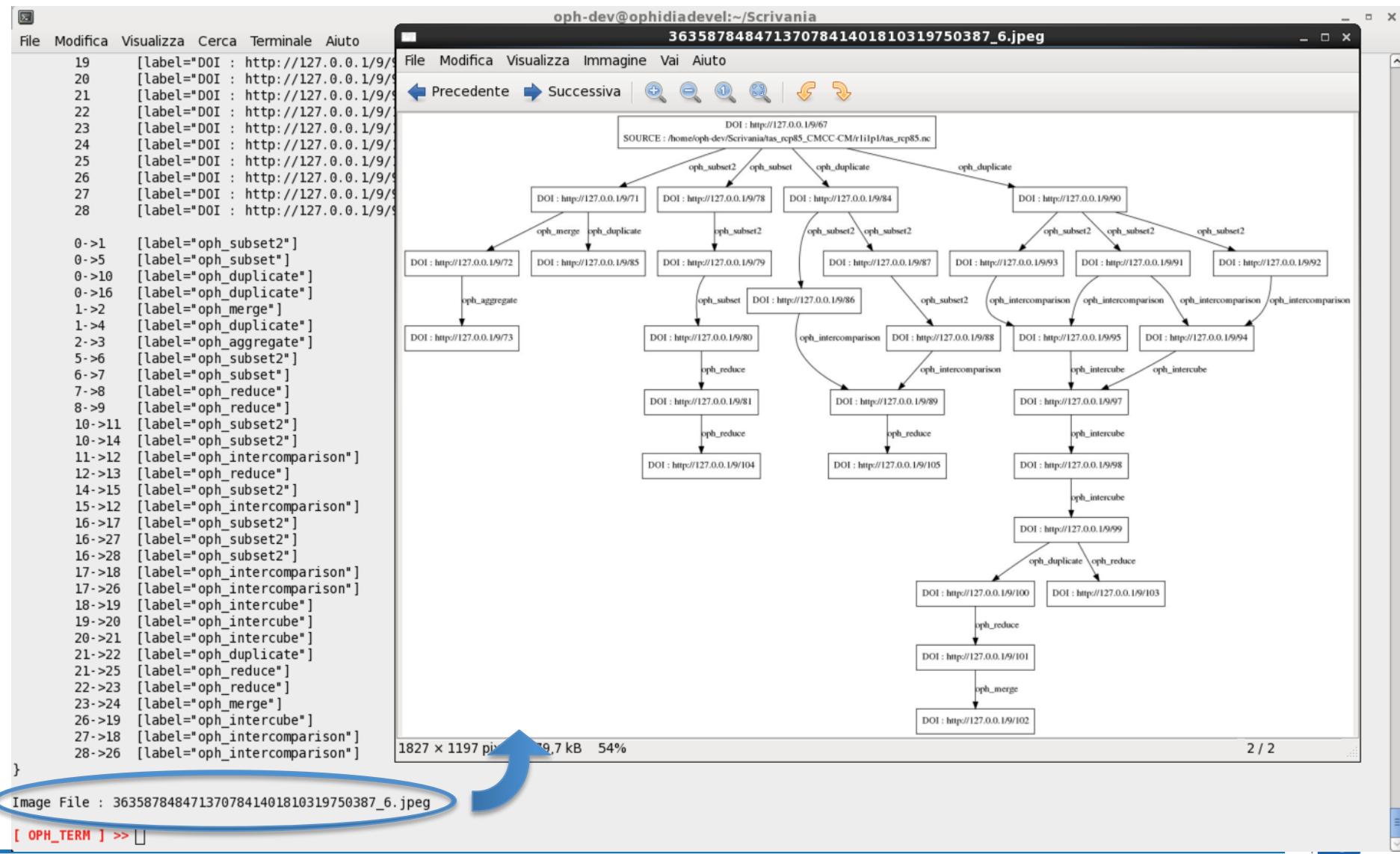
```
Proc 0: Total execution: Time 0,024378 sec
```

[JobID]:

<https://ophidia.cmcc.it:8443/sessions/417510192530258072601378724535658888/document#9>



Terminal: provenance



Summary

- *The Ophidia framework provides:*
 - *an internal storage model for multidimensional data*
 - *about 100 array-based primitives*
 - *about 50 operators for data analytics and metadata management*
 - *a Web Service interface (WS-I⁺) and a RESTful one*
 - *support for heterogeneous back-end storages*
 - *self-issued persistent identifier for the datacubes managed by the system*
 - *Workflow capabilities and provenance management*
- *APIs available at 5 different levels of the stack*
 - *server front-end, I/O server, storage device, array-based primitives, datacube operators*
- *Client application available as user interface*
 - *Complete bash-like environment, production-level, lightweight, well-documented*
- *Cloud-based deployment scenarios in the FP7 EUBrazilCC project*
- *Official release soon available on github (end of December/January)*



References

- [1] G. Aloisio, S. Fiore, I. Foster, D. N. Williams , “**Scientific big data analytics challenges at large scale**”, *Big Data and Extreme-scale Computing (BDEC)*, April 30 to May 01, 2013, Charleston, USA (position paper).
- [2] S. Fiore, G. Aloisio, I. Foster, D. N. Williams , “**A software infrastructure for big data analytics**”, *Big Data and Extreme-scale Computing (BDEC)*, February 26-28, 2014, Fukuoka, Japan (position paper).
- [3] S. Fiore, A. D'Anca, C. Palazzo, I. Foster, Dean N. Williams, Giovanni Aloisio, “**Ophidia: Toward Big Data Analytics for eScience**”, *ICCS 2013*, June 5-7, 2013 Barcelona, Spain, Procedia Computer Science, Elsevier, pp. 2376-2385.
- [4] S. Fiore, C. Palazzo, A. D'Anca, I. Foster, D. N. Williams, G. Aloisio, “**A big data analytics framework for scientific data management**”, Workshop on “*Big Data and Science: Infrastructure and Services*”, *IEEE International Conference on BigData 2013*, October 6-9, 2013, Santa Clara, USA, pp. 1-8.
- [5] S. Fiore, A. D'Anca, D. Elia, C. Palazzo, I. Foster, D. Williams, G. Aloisio, “**Ophidia: A Full Software Stack for Scientific Data Analytics**”, proc. of the *2014 International Conference on High Performance Computing & Simulation (HPCS 2014)*, July 21 – 25, 2014, Bologna, Italy, pp. 343-350, ISBN: 978-1-4799-5311-0



Thanks

