Parquet Cube to store and process gridded data

EGU, May 5th 2020

CLS Group :

- Elisabeth Lambert
- Jean-Michel Zigna
- Thomas Zilio

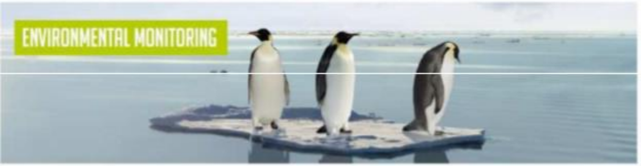
CNES :

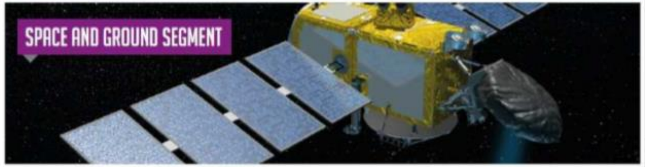
- Flavien Gouillon

Summary

- CLS needs and use case
- CLS historical spatial data infrastructure
- Challenges for gridded data in big data
- CLS Parquet implementation
- benchmarks
- Conclusion

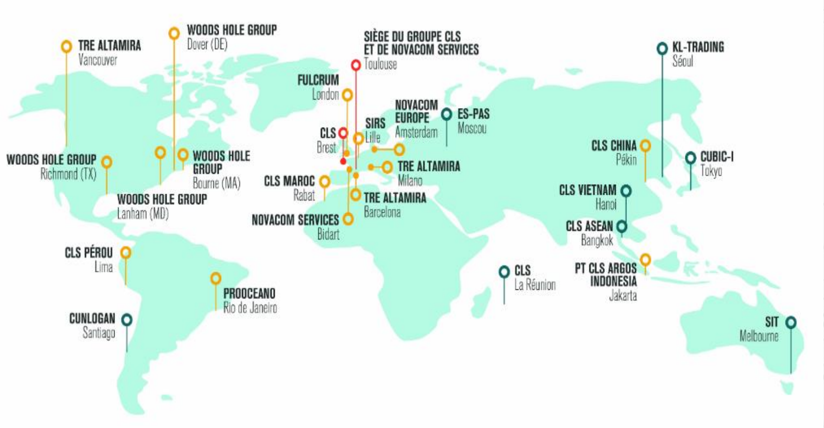
CLS needs





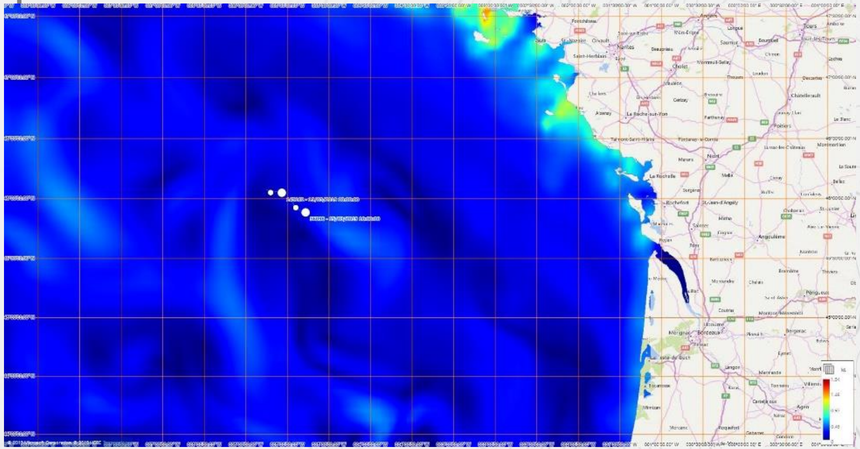
Business Services:

- Historical and NRT data
- Processing and dissemination



CLS use cases

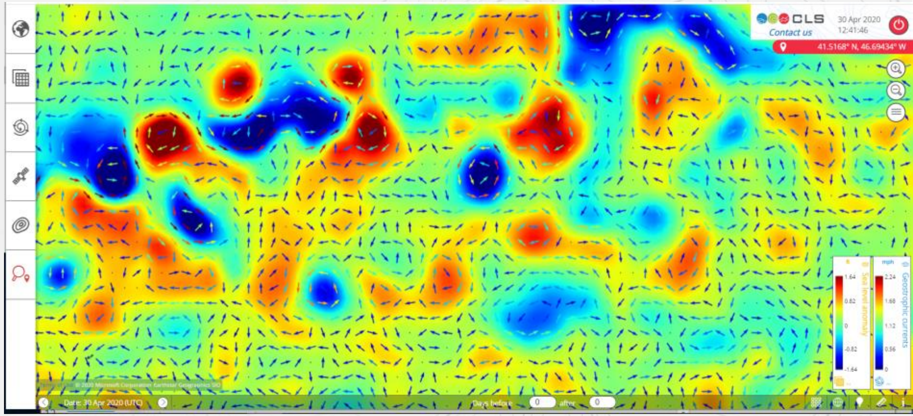
- Processing:
    - altimetry products (AVISO), C3S & CLMS products, Sargassum monitoring products, maritime currents
    - In situ data : bouys (Argo), MELOA (drifters), animal tracking, logbooks
    - Satellite images : optical and radar products
    - Vessel positions (AIS, VMS, LRIT)
- Discovery and Visualization
- Dissemination



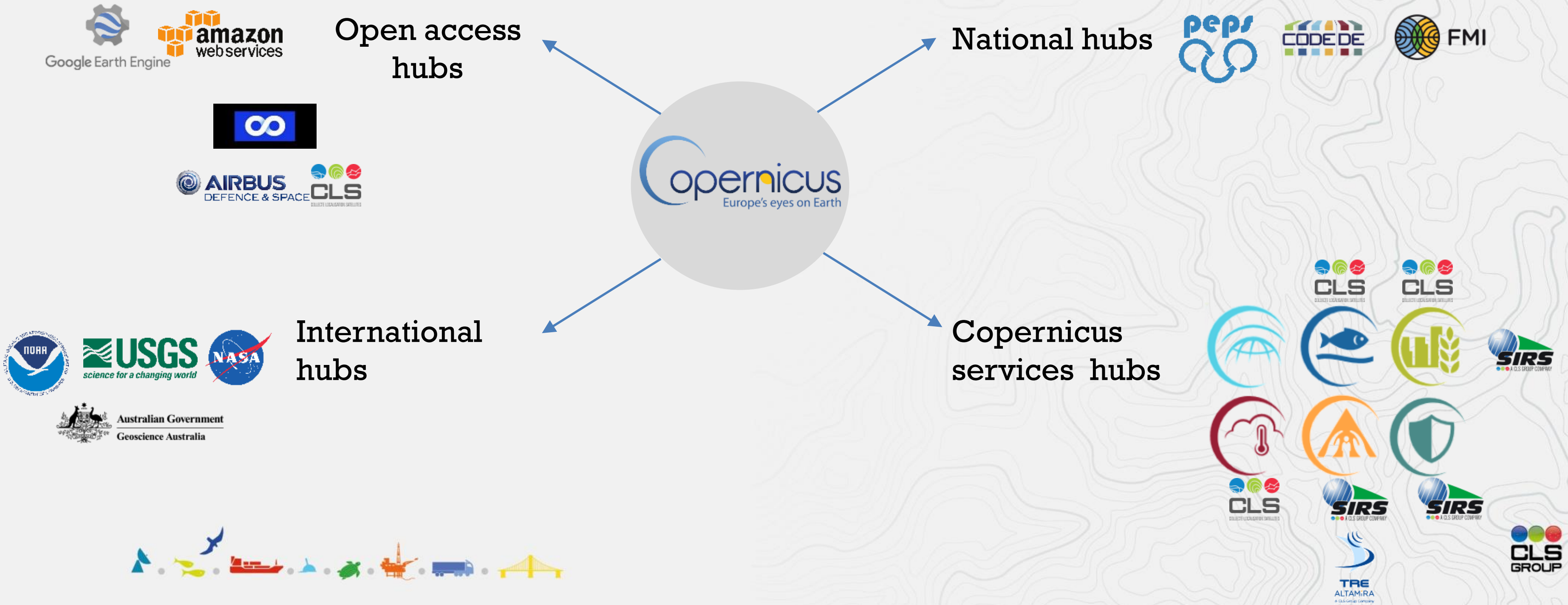
Analyze data in a geographical box or along tracks

Typical Scientific Needs

- Long term studies
- Artificial Intelligence
  - – Environmental studies
  - – Habitat modeling
  - – Fish stock modelization



Typical Scientific data sources



Open access hubs

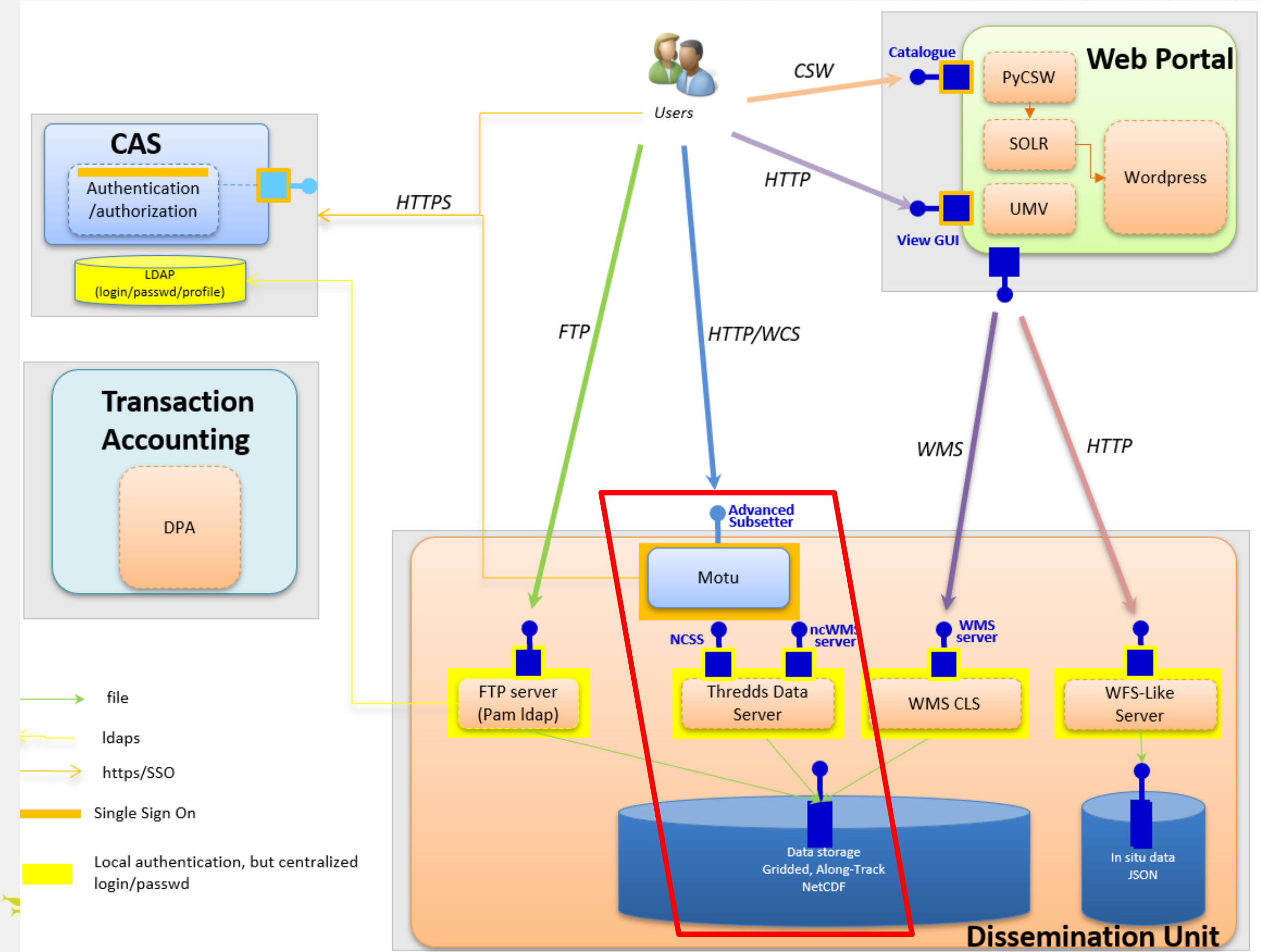
National hubs

copernicus
Europe's eyes on Earth

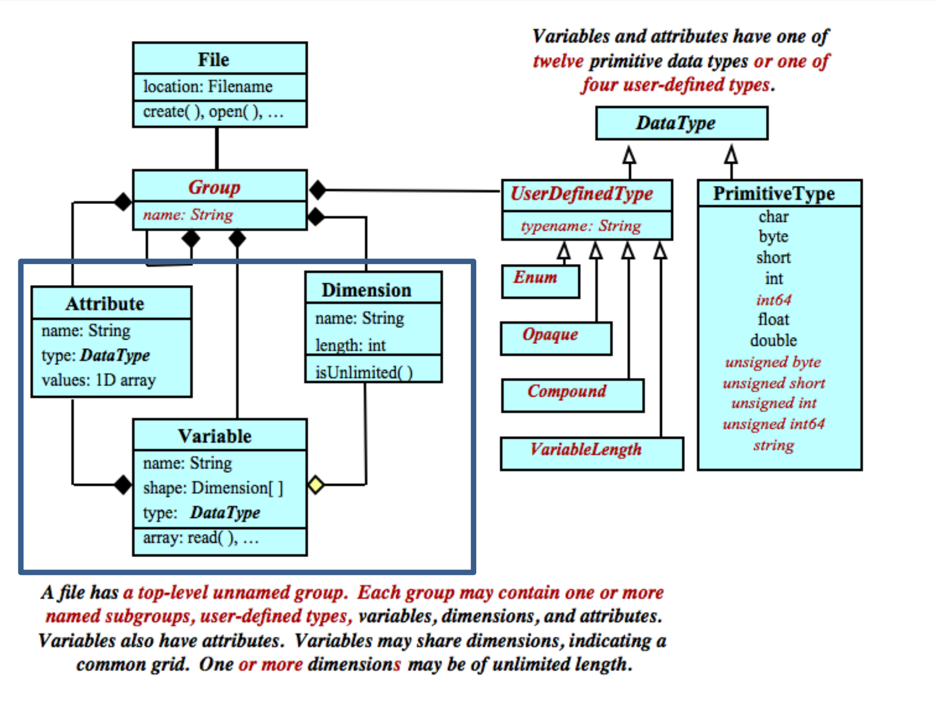
International hubs

Copernicus services hubs

Spatial Data Infrastructure: CLS Datastore

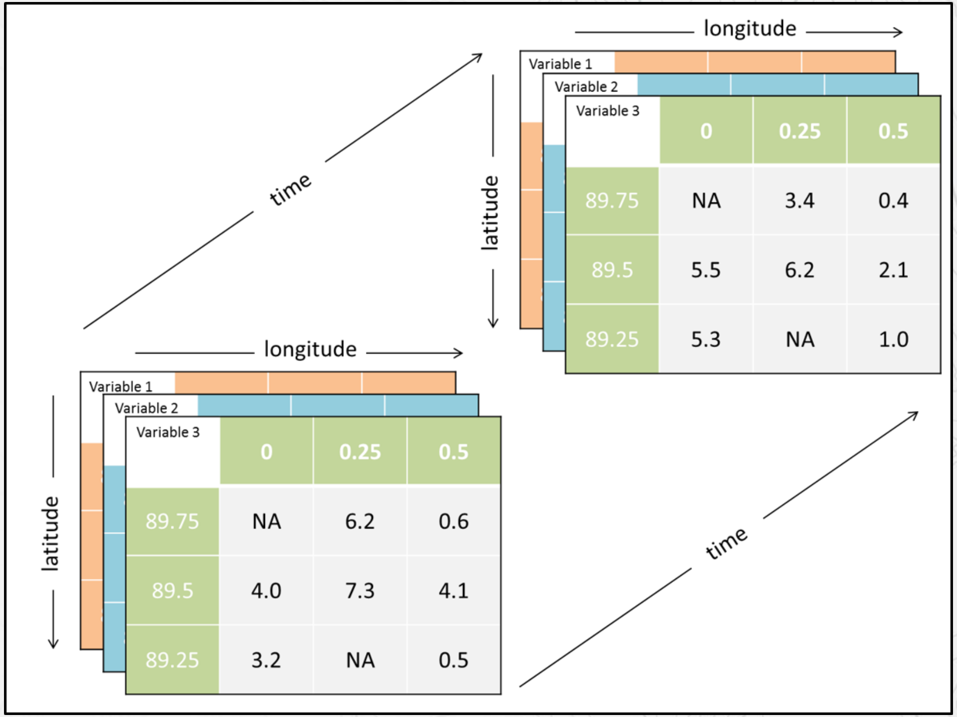


NetCDF data model



Variables and attributes have one of twelve primitive data types or one of four user-defined types.

A file has a top-level unnamed group. Each group may contain one or more named subgroups, user-defined types, variables, dimensions, and attributes. Variables also have attributes. Variables may share dimensions, indicating a common grid. One or more dimensions may be of unlimited length.



Variable 3	0	0.25	0.5
89.75	NA	3.4	0.4
89.5	5.5	6.2	2.1
89.25	5.3	NA	1.0

Variable 3	0	0.25	0.5
89.75	NA	6.2	0.6
89.5	4.0	7.3	4.1
89.25	3.2	NA	0.5

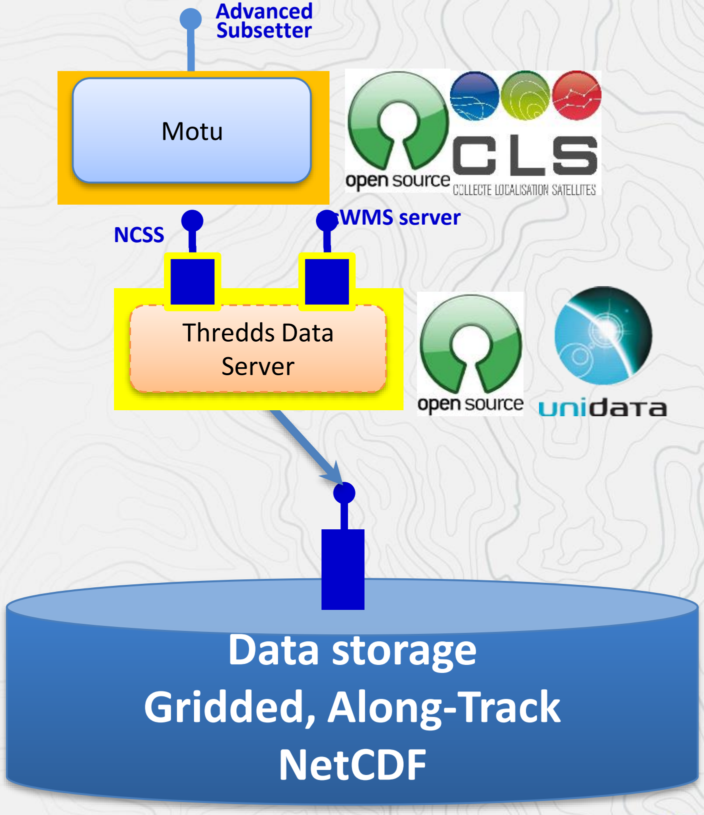
A Traditional Access to Data in Cube

- MOTU : advanced subsetter

  Area, Time Range, Variables

- THREDDS :
  - Aggregation of datasets

- Gridded NetCDF data files
  - Tree
  - nomenclature

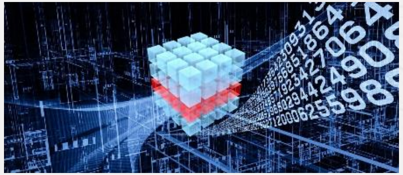


CLS Subsetting Status

- It works for reasonable volumes (Typically a few Gb/query)
- Users can download exactly what they need

- Only reduces data duplication issue
- Introduces additional layer between the data and the user
- Is not an optimal fit for large time series extraction
- High resolutions products are already reaching the limits

Challenges for gridded data in big data



Increasing the volume to store/archive

Increasing the potential of process

Various usage types

What it does imply

Distributing the storage

Distributing/Parallelising the processing

Moving the algorithm to a shared data

Why a Parquet Cube ?

Distributed storage

- HDFS
- S3

Distributed processing

- Spark
- Dask

Interactive notebooks

- Jupyter
- Zeppelin

Since…

Netcdf/HDF format is not cloud/bigdata friendly yet!

A new format

Parquet

- Popular columnar tabular format in bigdata and cloud stacks
- Key strength: Store nested data in truly columnar format using definition and repetition levels
  - Columnar storage format
  - metadata at the end of files
  - Supports most of Hadoop processing frameworks
  - Optimized for high compression and high scan efficiency

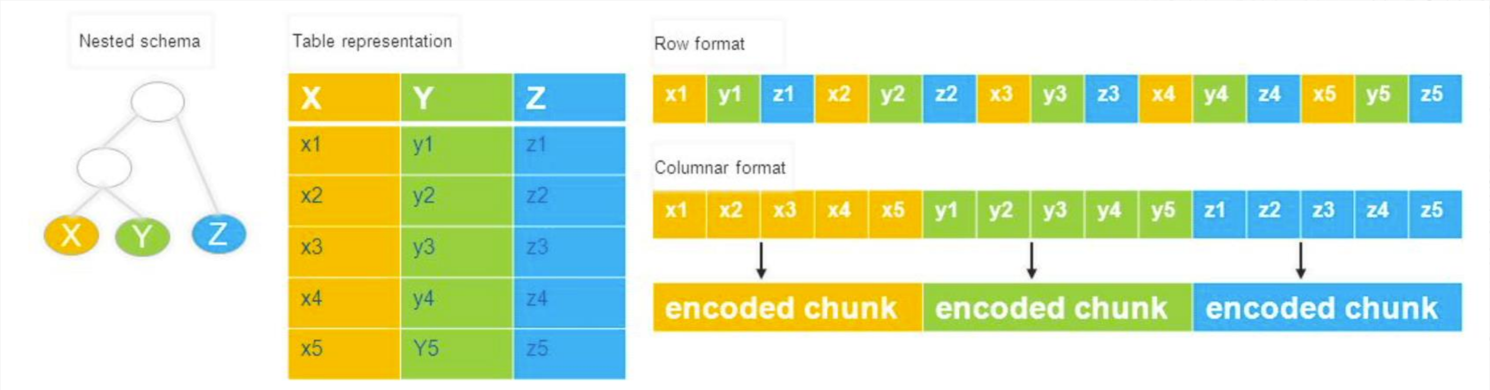


Table representation

X	Y	Z
x1	y1	z1
x2	y2	z2
x3	y3	z3
x4	y4	z4
x5	Y5	z5

Row format

x1 y1 z1 x2 y2 z2 x3 y3 z3 x4 y4 z4 x5 y5 z5

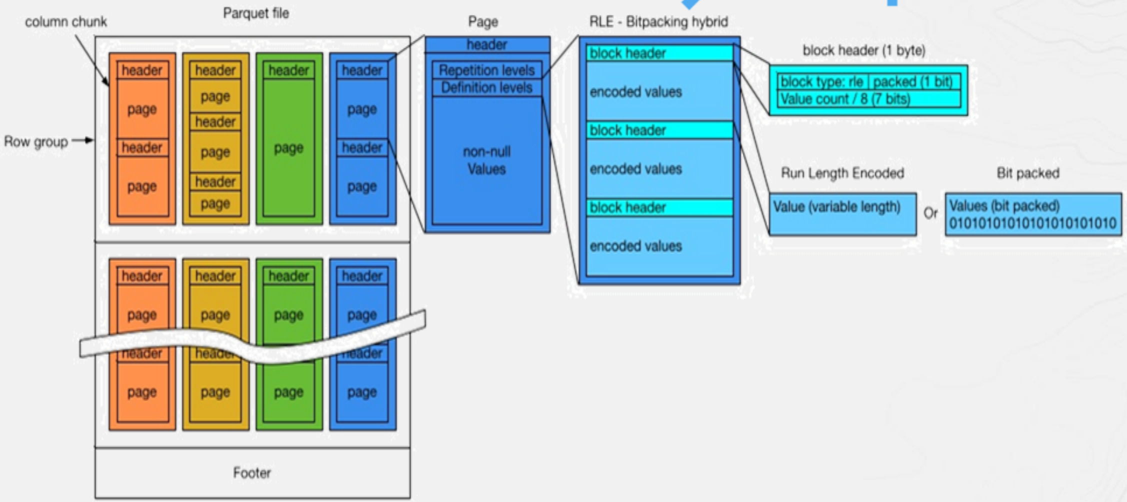
Columnar format

x1 x2 x3 x4 x5 y1 y2 y3 y4 y5 z1 z2 z3 z4 z5

encoded chunk encoded chunk encoded chunk

A new format

Parquet



```
data
 |_ index
 |   |_ alti8-h
 |   |   |_ latitude
 |   |   |_ longitude
 |   |   |_ time
 |   |   |_ depth
 |   |_ alti8-uv
 |       |_ latitude
 |       |_ longitude
 |       |_ time
 |       |_ depth
 |_ data
     |_ alti8-h
     |   |_ tspartday=2017032
     |   |_ tspartday=2017033
     |   |_ tspartday=2017034
     |   |_ tspartday=2017035
     |_ alti8-uv
         |_ tspartday=2017032
         |_ tspartday=2017033
         |_ tspartday=2017034
         |_ tspartday=2017035
```

- Stable, performant and widely used in the industry
- Numerous language bindings
- But… a tabular format

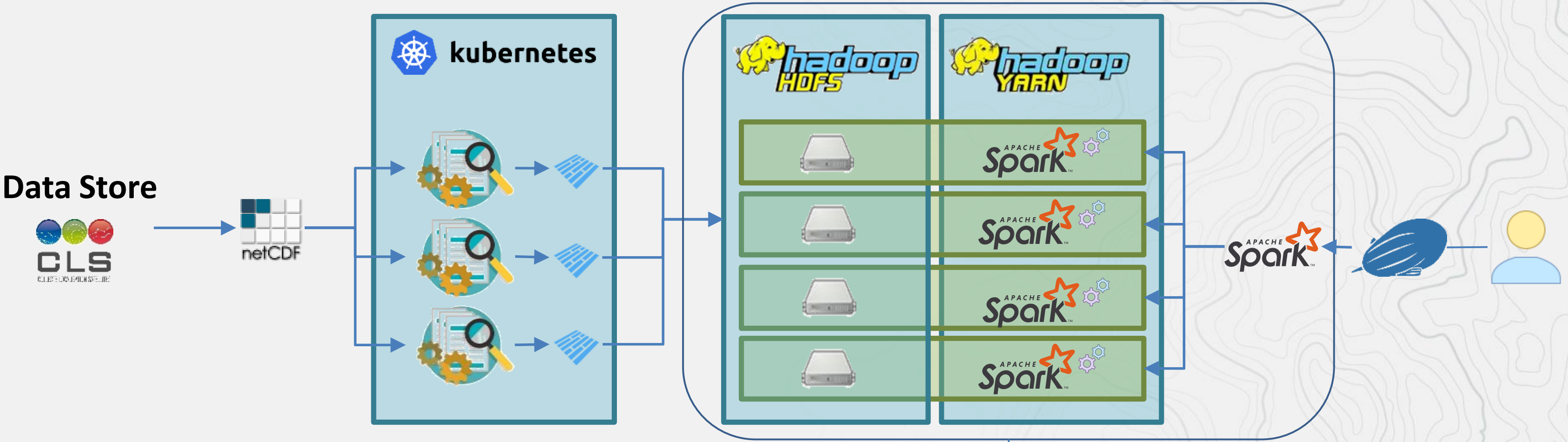
Usage of Parquet for gridded data

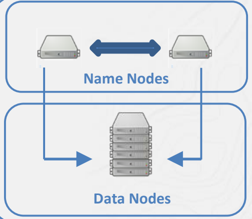
- A NetCDF file is a Grid with:
  - several dimensions:
    - Latitude
    - Longitude
    - Time
    - Depth (if 4D)
  - several variables

- The NetCDF is exploded in order to have a parquet file which each dimensions and variables are a column

Dimensions				Variables		
latitude	longitude	time	depth	var1	....	varn

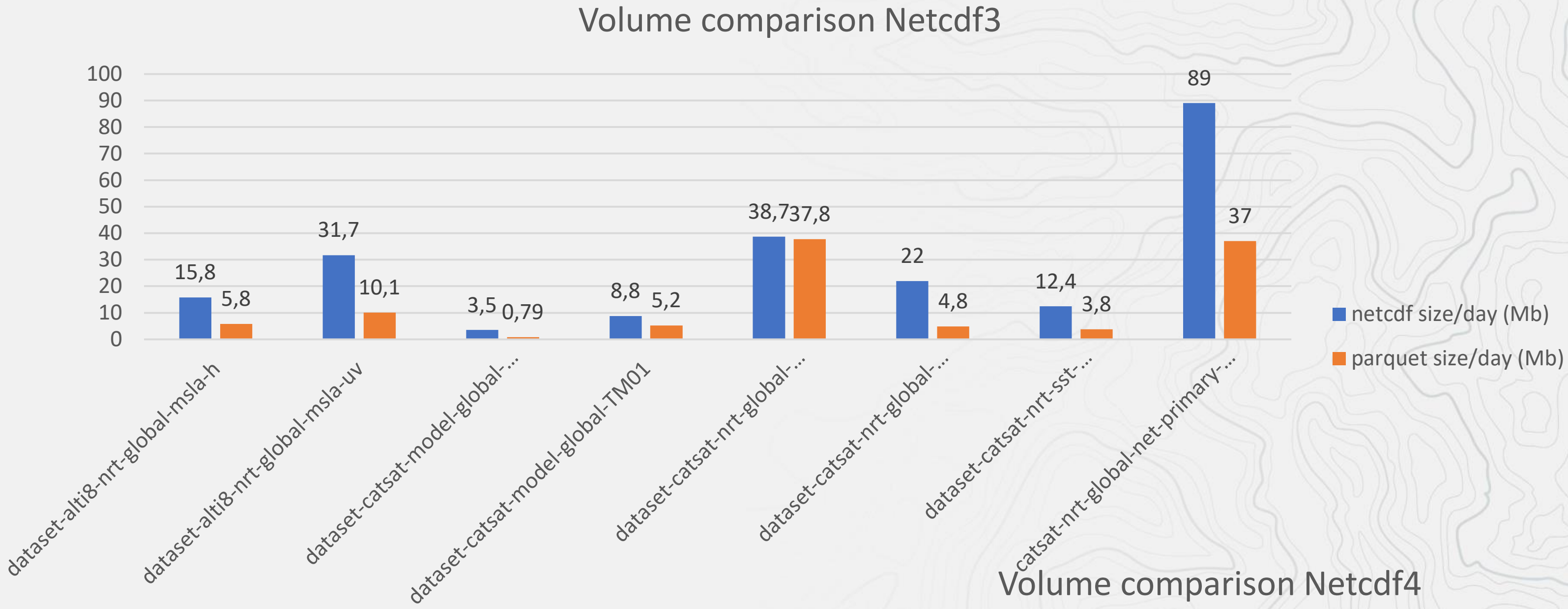
Software/Hardware architecture

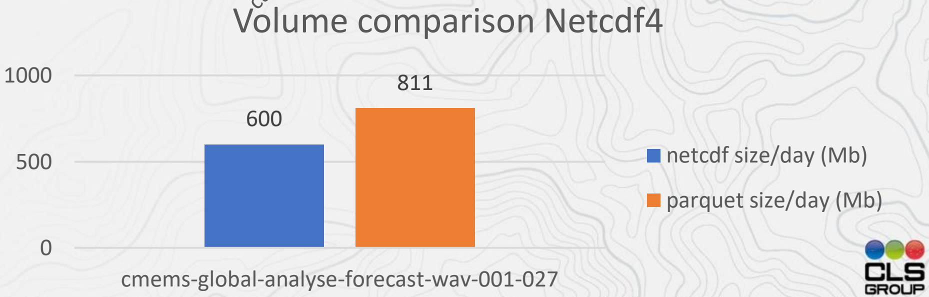




	Name Node (x2)	Data node (x6)
Processor	Xeon, 14 core	
RAM	128 Go	
Disk	2x1,2 To	8x1,2 To

Usage of Parquet for gridded data





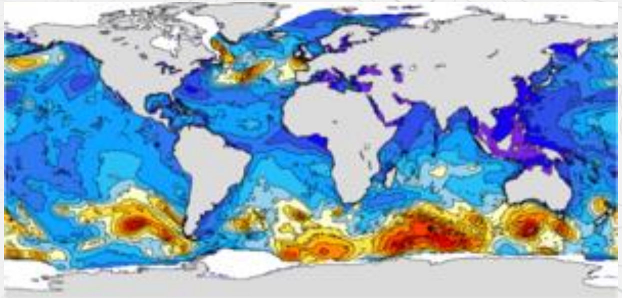
Benchmarks Along Tracks & bounding box extractions

- Dataset :
  - cmems-global-analyse-forecast-wav-001-027
  - NetCDF4: 600 Mb/day

Each day (17 variables):

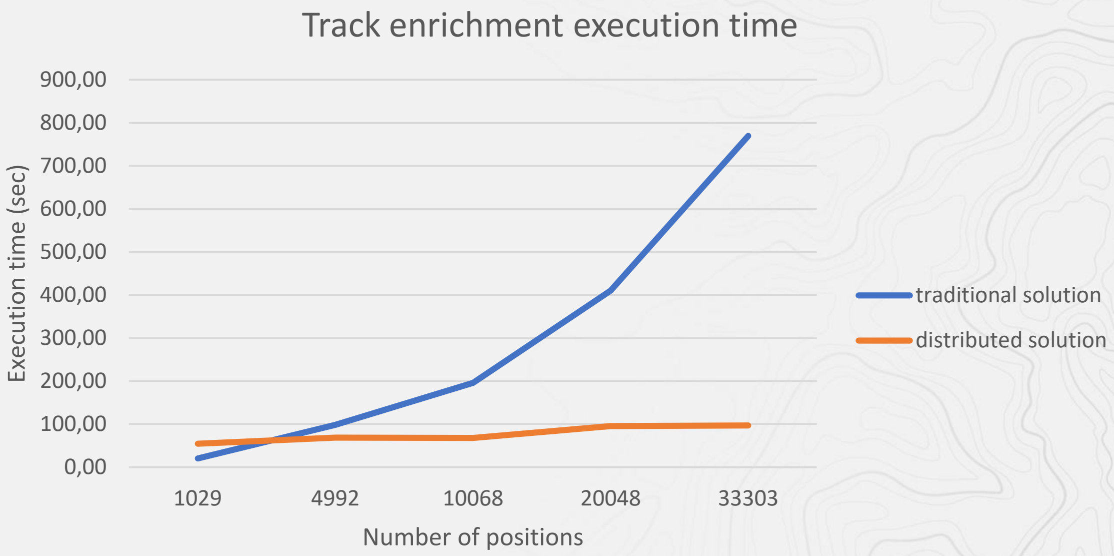
- 8 time steps
- 2041 latitude values
- 4320 longitude values

~ $70*10^6$ data point per variable per day

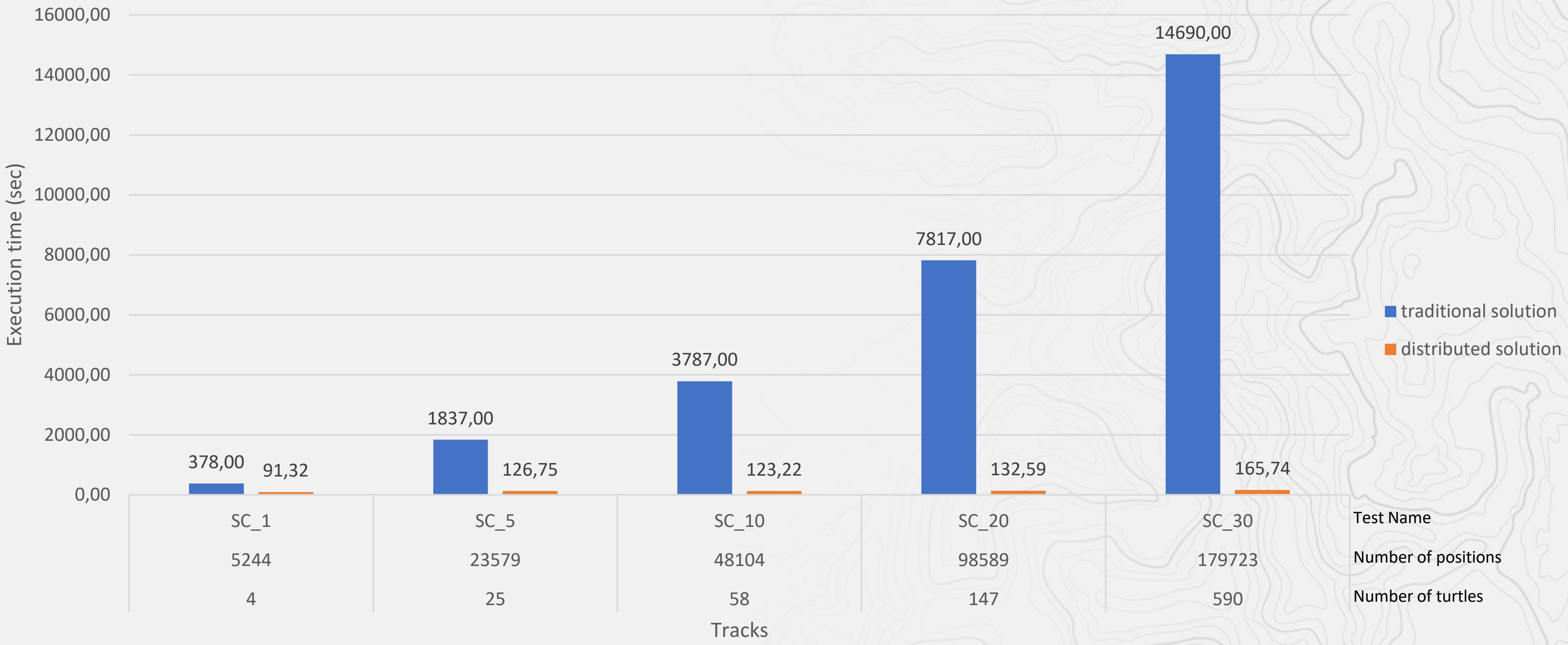


- Extracted variables:
  - sea_surface_wave_stokes_drift_x_velocity
  - sea_surface_wave_stokes_drift_y_velocity

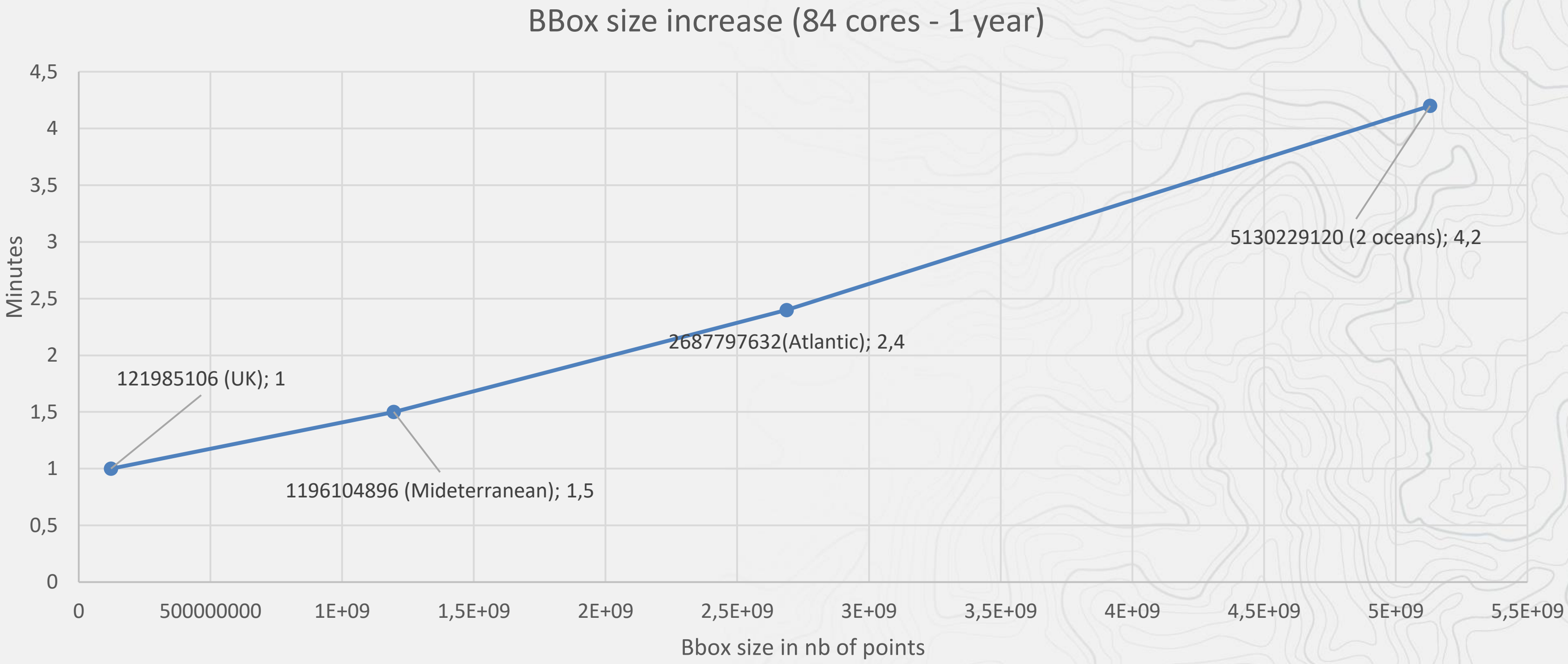
Comparison for along track enrichment over six days



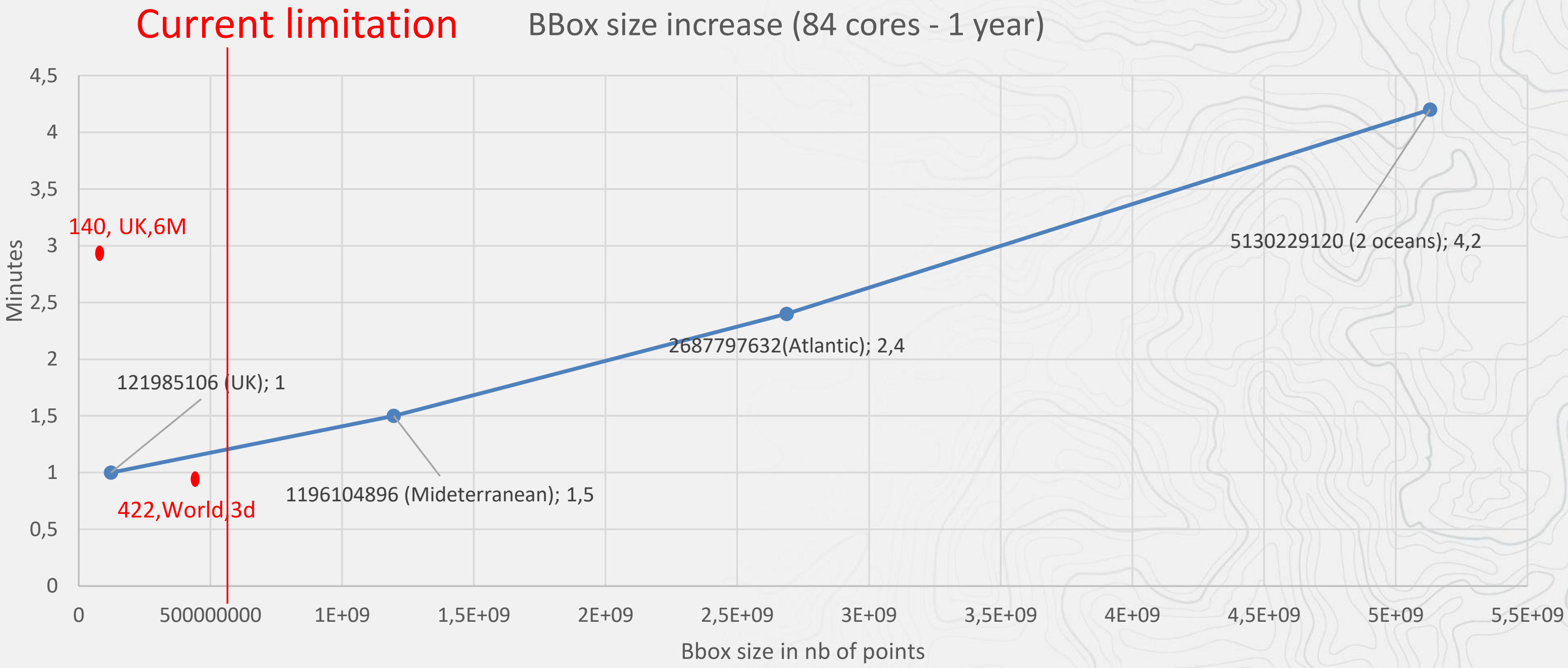
Comparison for along track enrichment over a month



Spatio temporal extractions with bigData solution



Comparison for bounding box extractions



Conclusions

- Parquet is a good fit performance-wise for gridded data
- The along track and bbox extraction services are promising
- Ingestion using Spark for inputs of gigabytes size
- Opened to both Spark and Dask frameworks

- Is a tabular format
- Metadata are maintained elsewhere in JSON

- We remain open to trending technologies
- We would like to make this Parquet Cube ingestion Open Source