BEACON

A high performance data lake and processing platform designed for access and sub-setting large quantities of climate data

Peter Thijsse

Robin Kooyman

Introducing BEACON

Current state of SeaDataNet CDI service:

- Just like in many other disciplines **marine observation data is often organised in files**
- Pan-European infrastructure for marine data management with > 100 data centres and > 2.6 million data sets for physics, chemistry, biology, geology, and bathymetry
- A functional user interface allows users to query the metadata, select files, order and download datasets.
- **But how to serve users and machines subsets?**

Challenge:

- Optimizing the CDI system for **Machine2Machine access to subsets**, enabling easy access to Jupyter Notebooks and other applications.
- **How to go from files to serving applications as an actual "Data lake"?**

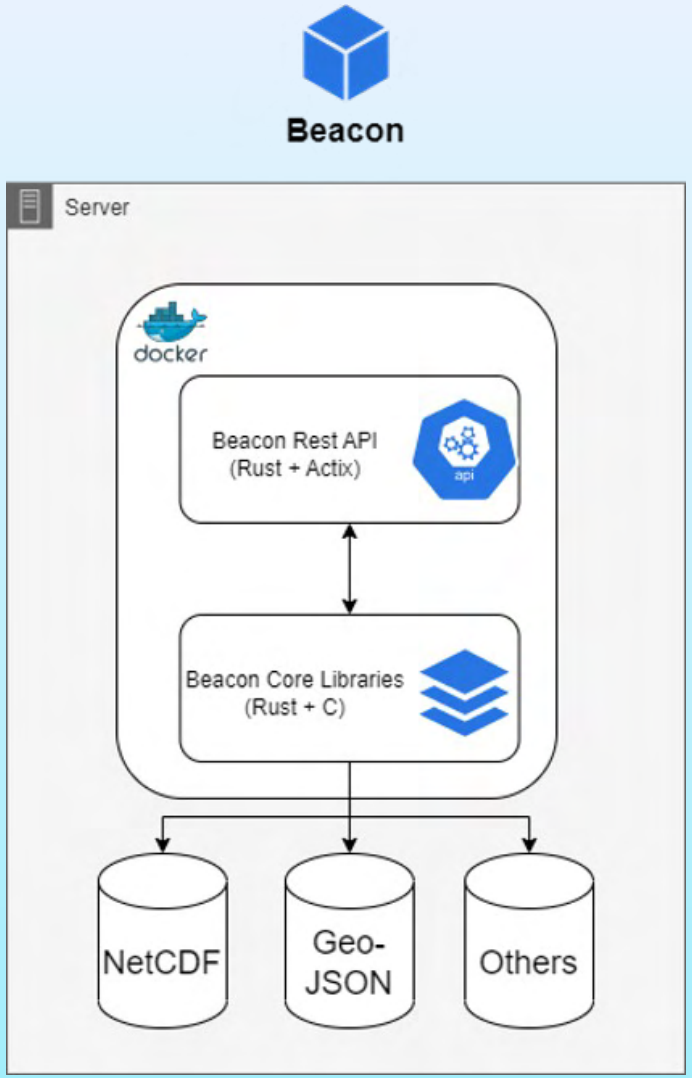
Example:

- Request: Give me all the temperature data in the North Sea, from 2010-2020, in degrees Celsius, at a depth of 0-50 m.
- Response: One NetCDF file containing exactly this data, which is then on the fly, directly usable in a Jupyter notebook and for HPC.

The goal of **BEACON** is to provide an easy-to-use, fast, reliable, and scalable solution for storing, processing, and retrieving large amounts of climate data.

How does Beacon work?

- Written in Rust + C
- High Performance Data Lake
- Runs on:
  - Linux
  - Windows
  - Docker Containers
- Consists of:
  - Rest API
  - Core Libraries
- Real-Time sub-setting
- Data harmonization
- Produces different output formats:
  - NetCDF
  - CSV
  - JSON
  - IPC (Apache Arrow)
  - GeoJSON
  - Parquet
  - BBF (Beacon Binary Format)



Features

Ultra fast performance

- The data lake is built using Rust, which provides fast performance and efficient resource usage. Together with interfacing directly with the Operating System (OS) kernel, means that data can be stored and retrieved quickly (with a peak I/O performance of 11 GB/s), even for large datasets.

Easy to use

- The data lake provides a simple and intuitive API, making it easy to store, process, and retrieve your data. Whether you're a seasoned data scientist or just getting started, you'll find the data lake easy to work with.

Scalable

- The data lake is designed to be highly scalable, allowing you to store and process large amounts of data without sacrificing performance. Even when you're dealing with terabytes of data and millions of datasets, this data lake can handle it.

Reliable

- The data lake uses a combination of modern software engineering practices together with the robustness of Rust to provide a highly reliable solution. This means that you can count on your data being safe and accessible.

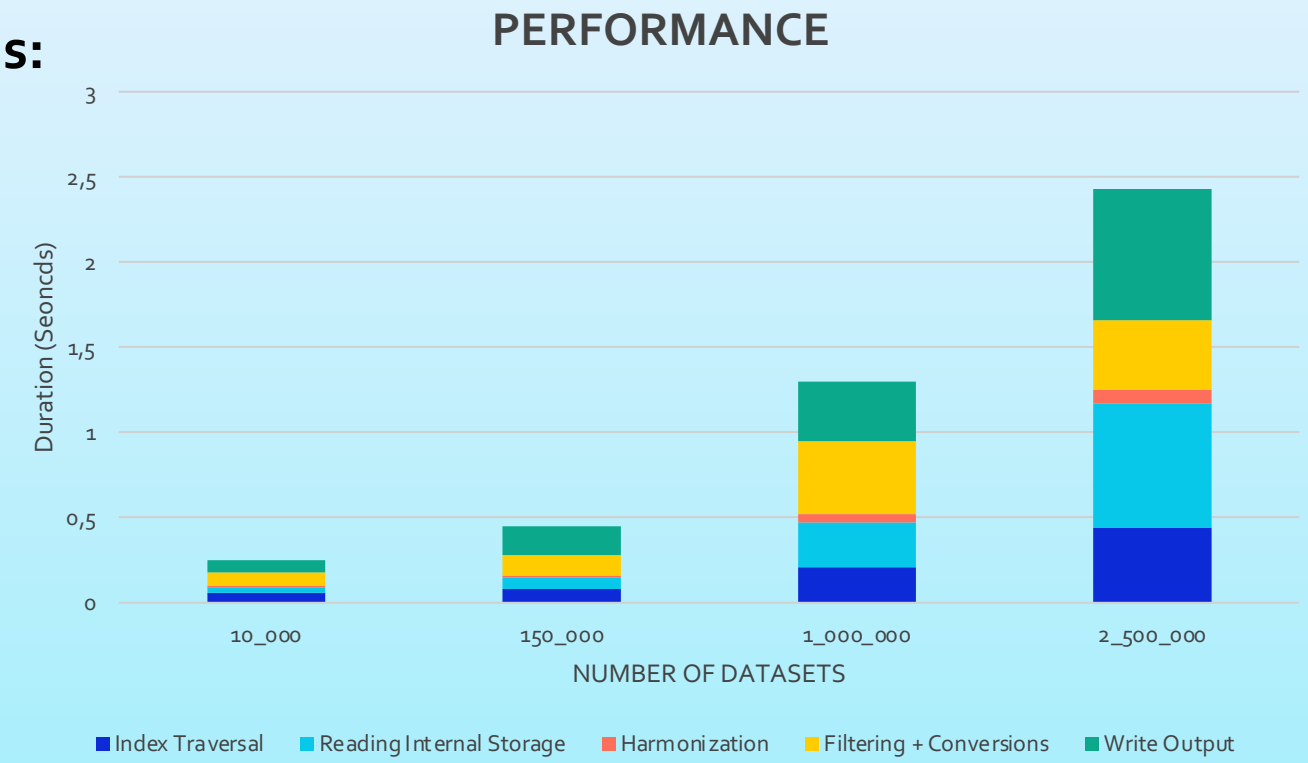
Beacon Performance

Loaded into beacon all SDN CDI records:

- 2.5 millions datasets
- > 4 billion data points
- 200GB of NetCDF Data

Query:

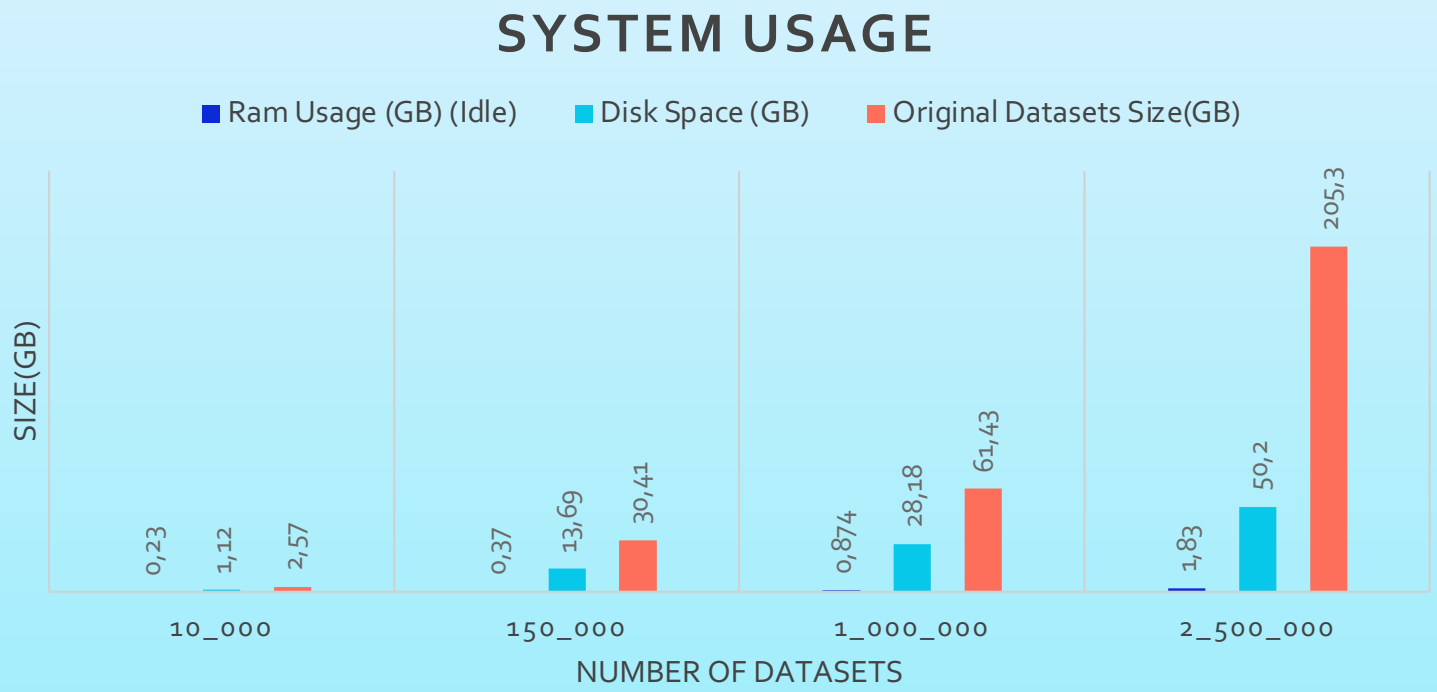
- Longitude from -8 to 12
- Latitude from 50 to 61
- Depth from 0 to 50
- Time from 2010 to 2012
- All the temperature parameters aggregated and harmonized in degrees Celsius
- Result: 12M points!



Beacon System Usage

Can run on Laptops, Home PC's and Servers

Only uses what's necessary to process the query



Importing Datasets

- Define import driver
  - Requires only 3 lines of JSON
  - Unit Attribute
  - Parameter Attribute
  - Driver name

```
{
    "parameter_attribute_name": "sdn_parameter_urn",
    "unit_attribute_name": "sdn_uom_urn",
    "compression": "lz4",
    "driver_name": "seadatanet-driver"
}
```

```
double LONGITUDE(INSTANCE) ;
        LONGITUDE:long_name = "Longitude" ;
        LONGITUDE:sdn_parameter_urn = "SDN:P01::ALONZZ01" ;
        LONGITUDE:sdn_parameter_name = "Longitude east" ;
        LONGITUDE:sdn_uom_urn = "SDN:P06::DEGE" ;
        LONGITUDE:sdn_uom_name = "Degrees east" ;
        LONGITUDE:units = "degrees_east" ;
        LONGITUDE:standard_name = "longitude" ;
        LONGITUDE:axis = "X" ;
        LONGITUDE:ancillary_variables = "POSITION_SEADATANET_QC" ;
        LONGITUDE:grid_mapping = "crs" ;
        LONGITUDE:_FillValue = -99999. ;
```

- API call with dataset path and import driver name

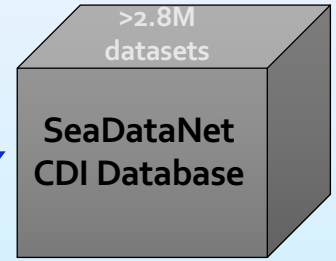
```
{
    "parameter_attribute_name": "standard_name",
    "unit_attribute_name": "units",
    "compression": "lz4",
    "driver_name": "sea-surface-reanalysis-driver"
}
```

Beacon input

- Parameter names
- Time period
- Depth range
- Bounding box

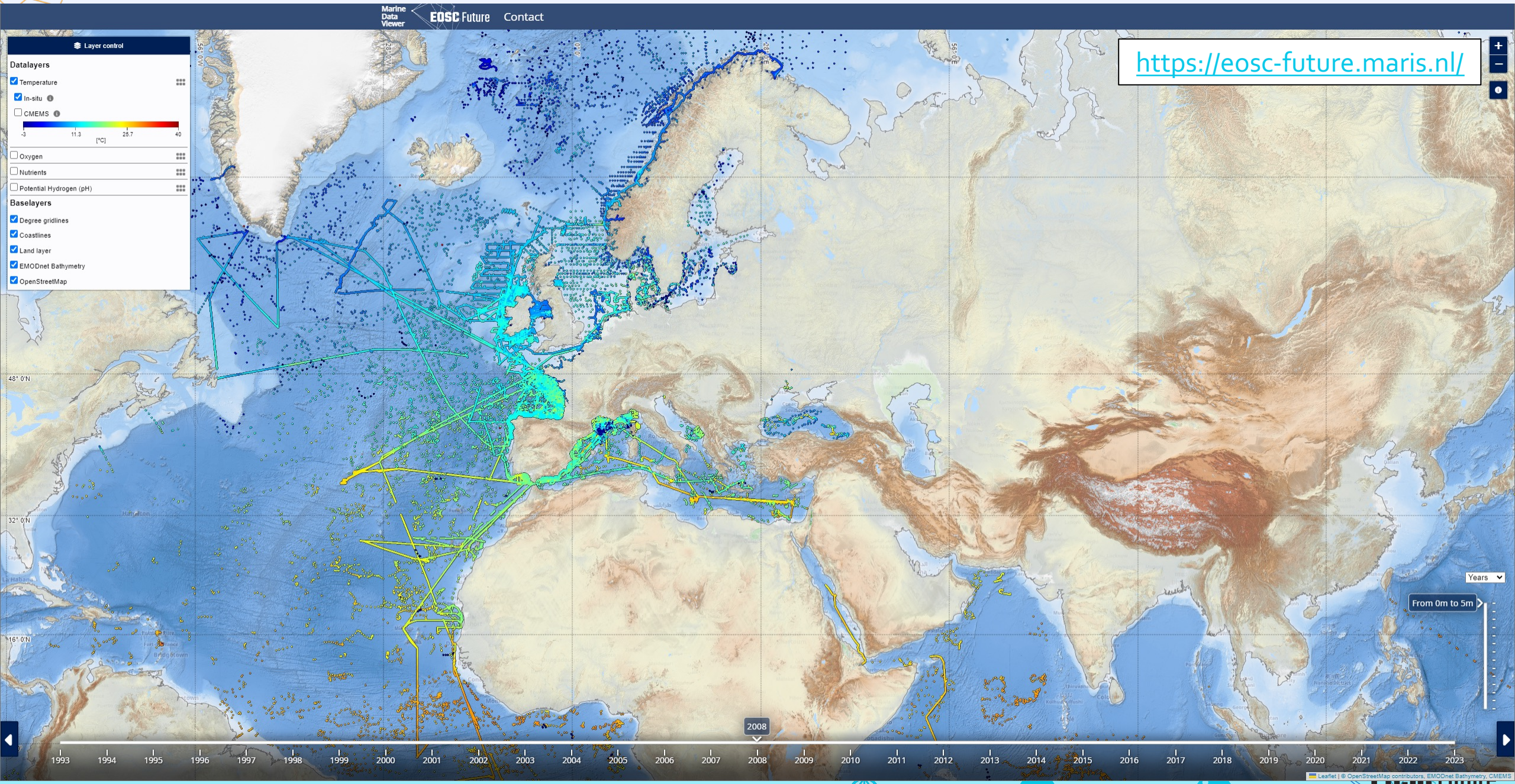
Example request

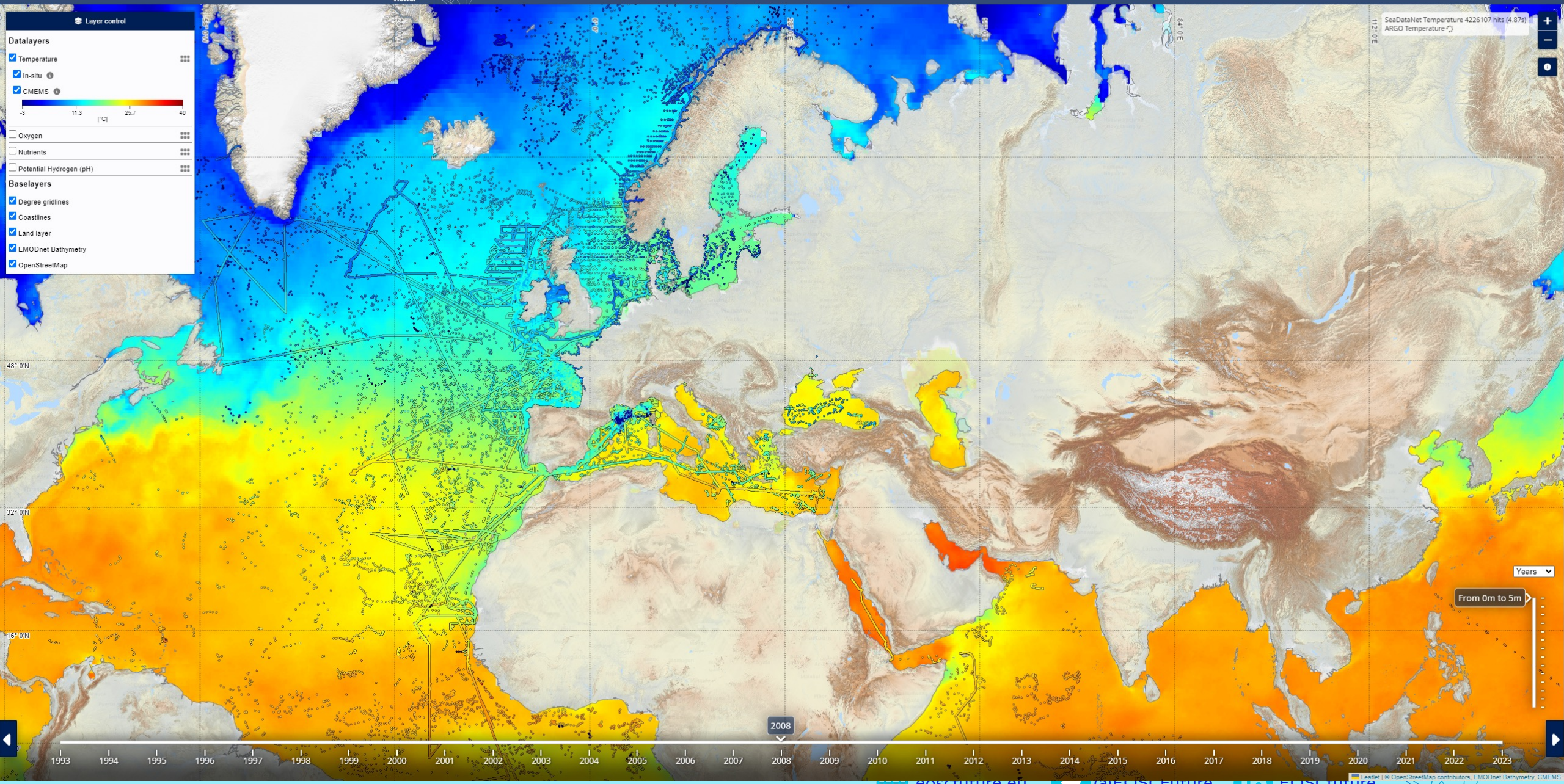
```
body = {
    "parameters": [
        {
            "parameter": {
                "name": f"EOSC_FUTURE::{parameter_name}"
            },
            "unit": {
                "name": f"SDN:P06::{unitp}"
            },
            "conversions": True,
            "include_original_data": False,
            "skip_missing_conversions": True,
            "skip_fill_values": False,
            "alias": parameter_name,
            "filter":{
                "min": parameter_min,
                "max": parameter_max
            }
        },
        {
            "parameter": {
                "name": "SDN:P01::CJDY1101"
            },
            "unit": {
                "name": "SDN:P06::UTAA"
            },
            "conversions": False,
            "include_original_data": False,
            "skip_missing_conversions": False,
            "skip_fill_values": False,
            "alias": "TIME",
            "filter":{
                "min": startjdt,
                "max": endjdt
            }
        },
        {
            "parameter": {
                "name": "EOSC_FUTURE::depth"
            },
            "unit": {
                "name": "SDN:P06::ULAA"
            },
            "conversions": True,
            "include_original_data": False,
            "skip_missing_conversions": True,
            "skip_fill_values": False,
            "alias": "DEPTH",
            "filter":{
                "min": upbound,
                "max": botbound
            }
        },
        {
            "parameter": {
                "name": "SDN:P01::ALONZZ01"
            },
            "unit": {
                "name": "SDN:P06::DEGE"
            },
            "conversions": False,
            "include_original_data": False,
            "skip_missing_conversions": False,
            "skip_fill_values": False,
            "alias": "LONGITUDE",
            "filter":{
                "min": lon_min,
                "max": lon_max
            }
        },
        {
            "parameter": {
                "name": "SDN:P01::ALATZZ01"
            },
            "unit": {
                "name": "SDN:P06::DEGN"
            },
            "conversions": False,
            "include_original_data": False,
            "skip_missing_conversions": False,
            "skip_fill_values": False,
            "alias": "LATITUDE",
            "filter":{
                "min": lat_min,
                "max": lat_max
            }
        }
    ],
    "output": "netcdf"
}
```



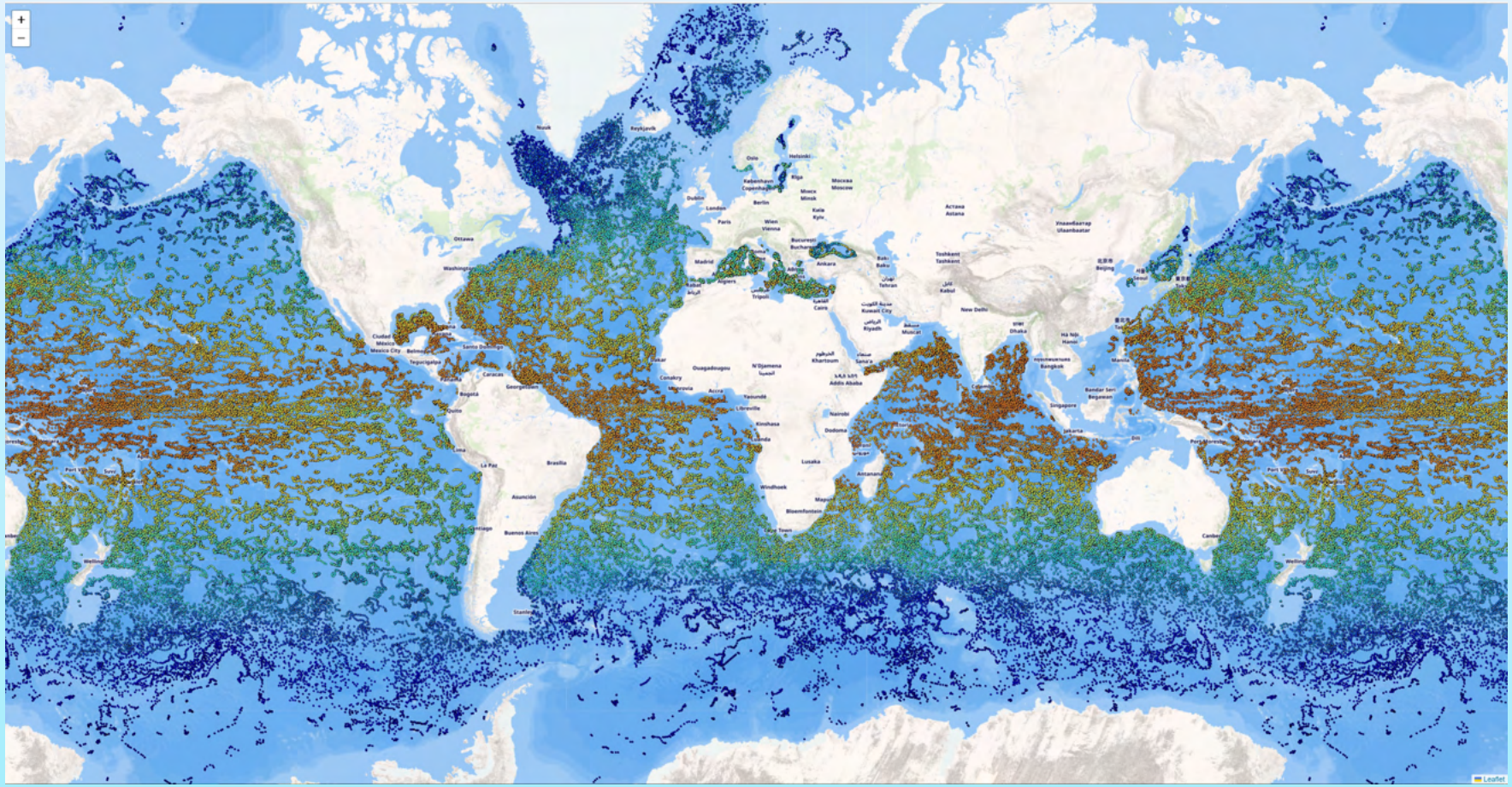
Datetime	TIME	LATITUDE	LONGITUDE	PRES	TEMPPR01	Season
1921-06-15 12:00:00.000000	2.422856e+06	48.50000	-5.20000	100.0	12.300	Spring
1921-06-15 12:00:00.000000	2.422856e+06	48.50000	-5.20000	125.0	12.300	Spring
1921-06-15 12:00:00.000000	2.422856e+06	48.50000	-5.20000	125.0	12.300	Spring
1921-06-15 12:00:00.000000	2.422856e+06	48.50000	-5.20000	50.0	12.300	Spring
1921-06-15 12:00:00.000000	2.422856e+06	48.50000	-5.20000	100.0	12.300	Spring
...	...	...	...	...	...	...
2021-10-29 18:17:30.000017	2.459517e+06	42.90297	15.14501	0.2	17.756	Autumn
2021-10-29 18:17:30.000017	2.459517e+06	42.90297	15.14501	191.5	13.002	Autumn
2021-10-29 18:17:30.000017	2.459517e+06	42.90297	15.14501	189.4	13.019	Autumn
2021-10-29 18:17:30.000017	2.459517e+06	42.90297	15.14501	6.1	17.763	Autumn
2021-10-29 18:17:30.000017	2.459517e+06	42.90297	15.14501	40.8	15.645	Autumn

32446479 rows × 6 columns


Marine Data Viewer
EOSC Future
Contact
https://eosc-future.maris.nl/
Layer control
Datalayers
Temperature
In-situ
CMEMS
-3
11.3
25.7
40
[°C]
Oxygen
Nutrients
Potential Hydrogen (pH)
Baselayers
Degree gridlines
Coastlines
Land layer
EMODnet Bathymetry
OpenStreetMap
48° 0'N
32° 0'N
16° 0'N
Years
From 0m to 5m
2008
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
Leaflet | © OpenStreetMap contributors, EMODnet Bathymetry, CMEMS


Layer control
Datalayers
Temperature
In-situ
CMEMS
-3
11.3
25.7
40
[°C]
Oxygen
Nutrients
Potential Hydrogen (pH)
Baselayers
Degree gridlines
Coastlines
Land layer
EMODnet Bathymetry
OpenStreetMap
SeaDataNet Temperature 4226107 hits (4.87s)
ARGO Temperature
48° 0'N
32° 0'N
16° 0'N
Years
From 0m to 5m
2008
1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020 2021 2022 2023
Leaflet | © OpenStreetMap contributors, EMODnet Bathymetry, CMEMS

Server side drawing instead of client side



Argo measurements at surface layer for 2020

Action takes place on the web server instead of the client's computer.

- Using Beacon Binary Format to cache API results
- Tiling WMS on top of beacon API instead of geoJSON
- Better browser performance (no memory issues, responsive UI)

Availability Beacon

- Tested at MARIS and EMODnet Physics
- Stable first version now available: Non-commercial use, slow roll-out based on request
- Aim to build a community (for API and file drivers)
  - See: https://beacon.maris.nl
- Working towards higher TRL in FAIR-EASE and Blue-Cloud 2026
  - E.g. including Cloud AWS support
  - Looking for further application cases

Get in touch

Peter Thijsse (peter@maris.nl)
Robin Kooyman (robin@maris.nl)

https://beacon.maris.nl