

# >Welcome!

## Hands-On with the EDITO Data API

Learn to explore, search, and use marine data from the EDITO Data Lake



Presented by Samuel Fooks (VLIZ)

For all the PDFs and code, check out the workshop [GitHub repository](#)





# What is EDITO?

**EDITO** stands for the **European Digital Twin of the Ocean**.

🧭 It is a European infrastructure to:

- Integrate marine data, models, and services
- Support marine policy (e.g. the Green Deal)
- Help connect EU/national initiatives and citizen science

🌐 Offers:

- Open API access to curated datasets
- Analysis-ready formats (Zarr, Parquet, COG)
- Tools to publish, process, and visualize ocean data



# Data in EDITO

The data available in the EU DTO consists of a **STAC (SpatioTemporal Asset Catalog)** as well Data storage on S3 buckets



三

# EDITO Data Storage

EDITO Data Lake uses modern cloud storage solutions to host public datasets.  
These datasets are stored in:

- S3-compatible object storage
  - Access via URL, anonymous or secure
  - High performance, cloud-native data formats

## EDITO STAC

EDITO offers a standardized **STAC (SpatioTemporal Asset Catalog)** built on **CMEMS** and **EMODnet** data, designed to integrate diverse marine and environmental datasets.

-  Based on **OGC STAC API** for easy discovery and access
-  Integrates data from multiple domains (ocean, climate, biodiversity)
-  Search by time, space, type — with direct links to S3-hosted assets
-  Supports both human users and automated workflows

A gateway to an **interoperable ocean of FAIR data**

# What is STAC?

**STAC** = SpatioTemporal Asset Catalog

A community standard for:

- Describing Earth-observation data
- Providing metadata for geospatial assets

Used across satellites, models, and in-situ data.

 Learn more: [stacspec.org](https://stacspec.org)

# STAC Structure

- ◆ **Catalogs** – High-level groupings (e.g., "All CMEMS data")
- ◆ **Collections** – Thematic datasets (e.g., temperature, sea level)
- ◆ **Items** – Individual assets with time+space (e.g., file for 2024-01-01)
- ◆ **Assets** – Actual data files: GeoTIFF, Zarr, Parquet...

Each has consistent metadata (bbox, datetime, etc.)

# 🔍 Use the EDITO STAC Viewer

[viewer.dive.edito.eu](https://viewer.dive.edito.eu)

We can follow the STAC structure to the EUROBIS database exported in parquet

Catalog -> Catalog -> Collection -> Item

EMODnet -> Biodiversity -> Occurrence data -> Occurrence data eurobis database observations

## **DEMO Using STAC Viewer**

---



Can also view in your browser [radianterarth.github.io/stac-browser](https://radianterarth.github.io/stac-browser)

# Search EDITO STAC via the API

Base URL for STAC:

`https://api.dive.edito.eu/data/`

 Docs: [Interact with Data API](#)



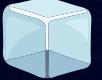
# What is ARCO Data?

**ARCO** = Analysis Ready Cloud Optimized

EDITO adopts modern cloud-friendly formats:

- High performance
- Scalable access
- Efficient for machine learning, large analytics

Let's explore each format!



# Zarr Format

Zarr is used for chunked N-dimensional arrays (like NetCDF but cloud-native)

- ✓ Ideal for model outputs, time series, climate reanalyses
- ✓ Works well with `xarray`, `kerchunk`, `zarr-python`

 [zarr.readthedocs.io](https://zarr.readthedocs.io)

```
import zarr
import xarray as xr

xr.open_zarr("https://s3...zarr/", consolidated=True)
```



# Parquet and GeoParquet

Parquet = columnar tabular format, very efficient

GeoParquet = Parquet + geospatial metadata

- ✓ Good for point observations, events, tracks, etc.
- ✓ Efficient for large queries and spatial joins

🔗 [parquet.apache.org](http://parquet.apache.org)  
🔗 [geoparquet.org](http://geoparquet.org)

# Access Parquet/GeoParquet via Arrow (Python)

```
import pyarrow.dataset as ds
import s3fs

fs = s3fs.S3FileSystem(anon=True)
dataset = ds.dataset("s3://...your-parquet-folder...",  
                     filesystem=fs, format="parquet")

df = dataset.to_table().to_pandas()
print(df.head())
```

Lets Explore the EDITO STAC, find an ARCO dataset from Biodiversity

[viewer.dive.edito.eu](https://viewer.dive.edito.eu)

# Reading parquet

Lets go read that parquet

[https://s3.waw3-  
1.cloudferro.com/emodnet/biology/eurobis\\_occurrence\\_data/eurobis\\_occurrences  
\\_geoparquet\\_2024-10-01.parquet](https://s3.waw3-1.cloudferro.com/emodnet/biology/eurobis_occurrence_data/eurobis_occurrences_geoparquet_2024-10-01.parquet)

Using a pre configured service on EDITO [explore\\_data/view\\_parquet](#)



# 🔍 Exploring STAC via the API (Python)

```
import pystac_client

url = "https://api.dive.edito.eu/data/collections"
editocollections = pystac_client.Client.open(url)
collections = list(editocollections.get_collections())

print("Found collections:", len(collections))
for col in collections[:5]:
    print(col.id, ":", col.title)
    items = col.get_items()
    itemlist = list(items)
    for item in itemlist:
        print(item.properties['title'])
        print(item.assets)
```



# Exploring STAC via the API (R)

```
library(rstac)

stac_endpoint <- "https://api.dive.edito.eu/data/"
collections <- stac(stac_endpoint) %>%
  rstac::collections() %>%
  get_request()

length(collections$collections) # how many
```

👉 R packages like `arrow`, `sf`, `terra` also help with asset processing.



# 📌 Recap: What You Can Now Do

- ✓ Understand the EDITO API and data stack
- ✓ Find and filter collections/items
- ✓ Read Parquet or Zarr data with Python or R

🧭 Go explore: [my-ocean.dive.edito.eu](https://my-ocean.dive.edito.eu)

[viewer.dive.edito.eu](https://viewer.dive.edito.eu)

💬 Questions?

✉️ Reach us at: [edito-infra-dev@mercator-ocean.eu](mailto:edito-infra-dev@mercator-ocean.eu)

🔗 Docs: [Interact with EDITO Data](#)

⌚ Happy exploring!