Using EDITO Datalab

15-Minute Tutorial for Marine Researchers

From finding services to running analysis - everything you need to know!

Presented by Samuel Fooks
Flanders Marine Institute (VLIZ)

For all the code and examples, check out the workshop GitHub repository

🎯 What We'll Cover (15 minutes!)

- ✅ **Find Services** - Navigate to datalab.dive.edito.eu
- ✅ **Configure & Launch** - Choose RStudio, Jupyter, or VSCode
- ✅ **Run Analysis** - STAC search, Parquet reading, Zarr data
- ✅ **Personal Storage** - Connect, upload, and manage your data
- ✅ **Live Demos** - See it all in action!

Perfect for researchers who want to get started quickly! 🚀

🌍 Whats in the EDITO Datalab?

EDITO = European Digital Twin of the Ocean

🧭 **A European infrastructure that provides:**

- Cloud computing services for marine research
- Access to curated marine datasets
- Analysis-ready data formats (Zarr, Parquet, COG)
- Personal storage for your data

🌐 **We'll look at 3 kinds of services:**

- **RStudio** - Statistical analysis and visualization
- **Jupyter** - Machine learning and data exploration
- **VSCode** - Multi-language development

🚀 Find Services

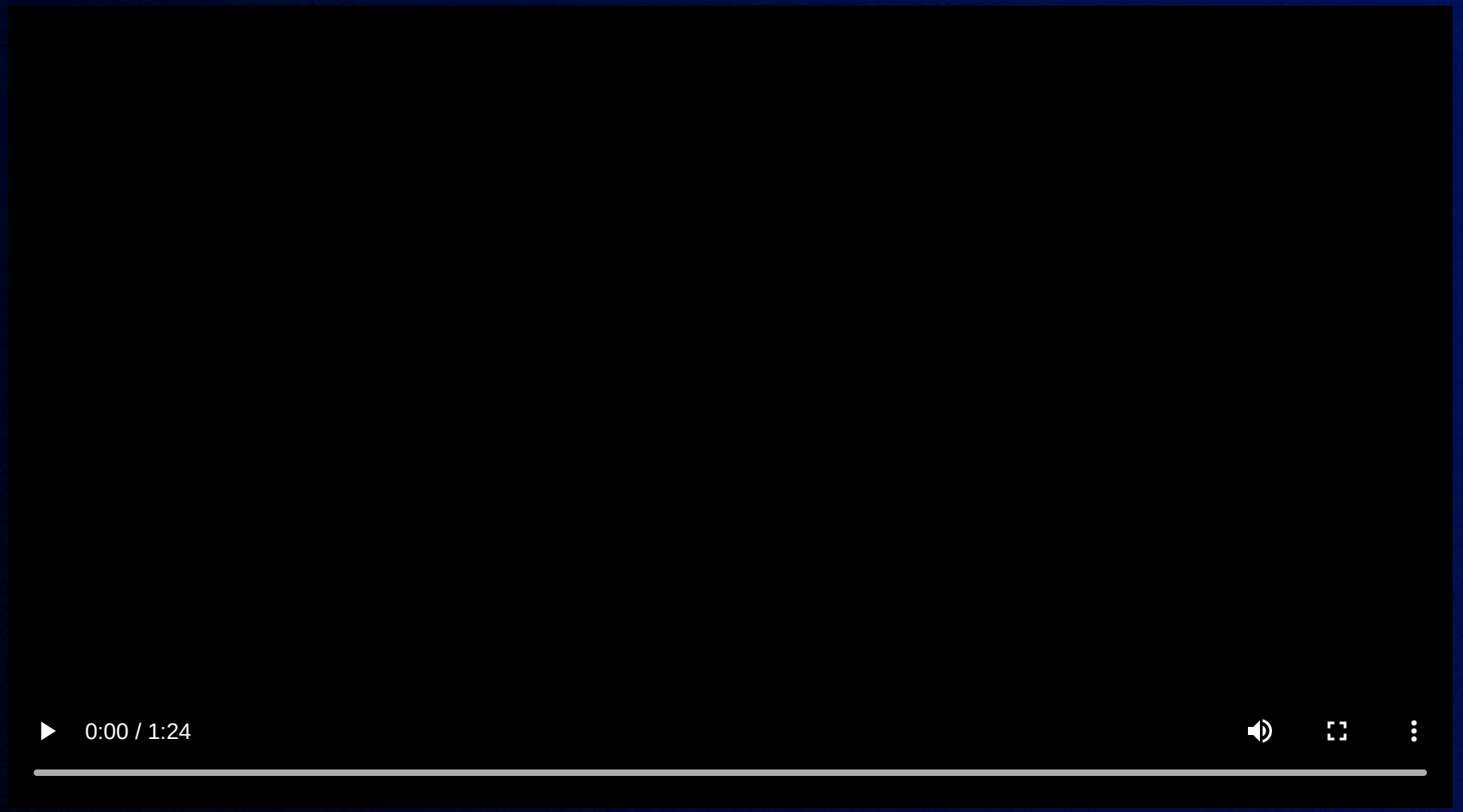
Go to EDITO Datalab

Website: datalab.dive.edito.eu

What You'll See:

- Service catalog with available options
- Resource configuration options
- Launch buttons for each service
- Creating an autolaunch link (you can use this when you create tutorials)

🎥 Navigating to datalab.dive.edito.eu and browsing services



⚙ Configure & Launch

Choose Your Service

RStudio Service

- **Perfect for**: Statistical analysis, spatial data, R users
- **Resources**: 2-8 CPU cores, 4-16GB RAM
- **Pre-installed**: R packages for marine research

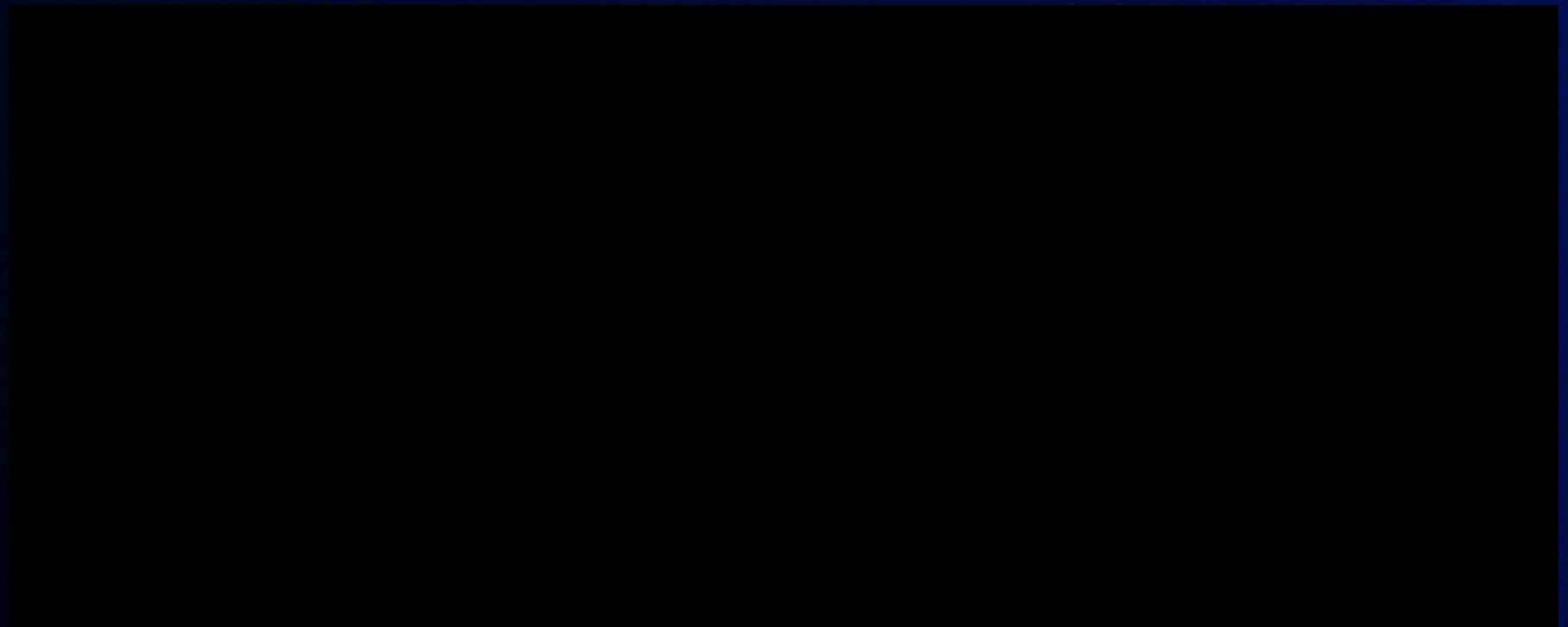
Jupyter Service

- **Perfect for**: Machine learning, data exploration, Python users
- **Resources**: 2-8 CPU cores, 4-16GB RAM
- **Pre-installed**: Python packages (pandas, xarray, etc.)

VSCode Service

- **Perfect for**: Multi-language projects, large codebases
- **Resources**: 2-8 CPU cores, 4-16GB RAM
- **Features**: Git integration, extensions, terminal

🎥 Launching VSCode Service in EDITO Datalab



🐟 Run Analysis

R Example - STAC Search & Parquet Reading

```r
# Connect to EDITO STAC API
library(rstac)
library(arrow)
library(dplyr)

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

# Read biodiversity data
parquet_url <- "https://s3.waw3-1.cloudferro.com/emodnet/biology/eurobis_occurrence_data/eurobis_occurrences_geoparquet_2024-10-01.parquet"
biodiversity_data <- arrow::read_parquet(parquet_url) %>% head(1000)

# Filter for marine species
marine_data <- biodiversity_data %>%
  filter(grepl("fish|mollusk|algae", scientificName, ignore.case = TRUE))
```

🎥 Querying STAC using R in VSCode

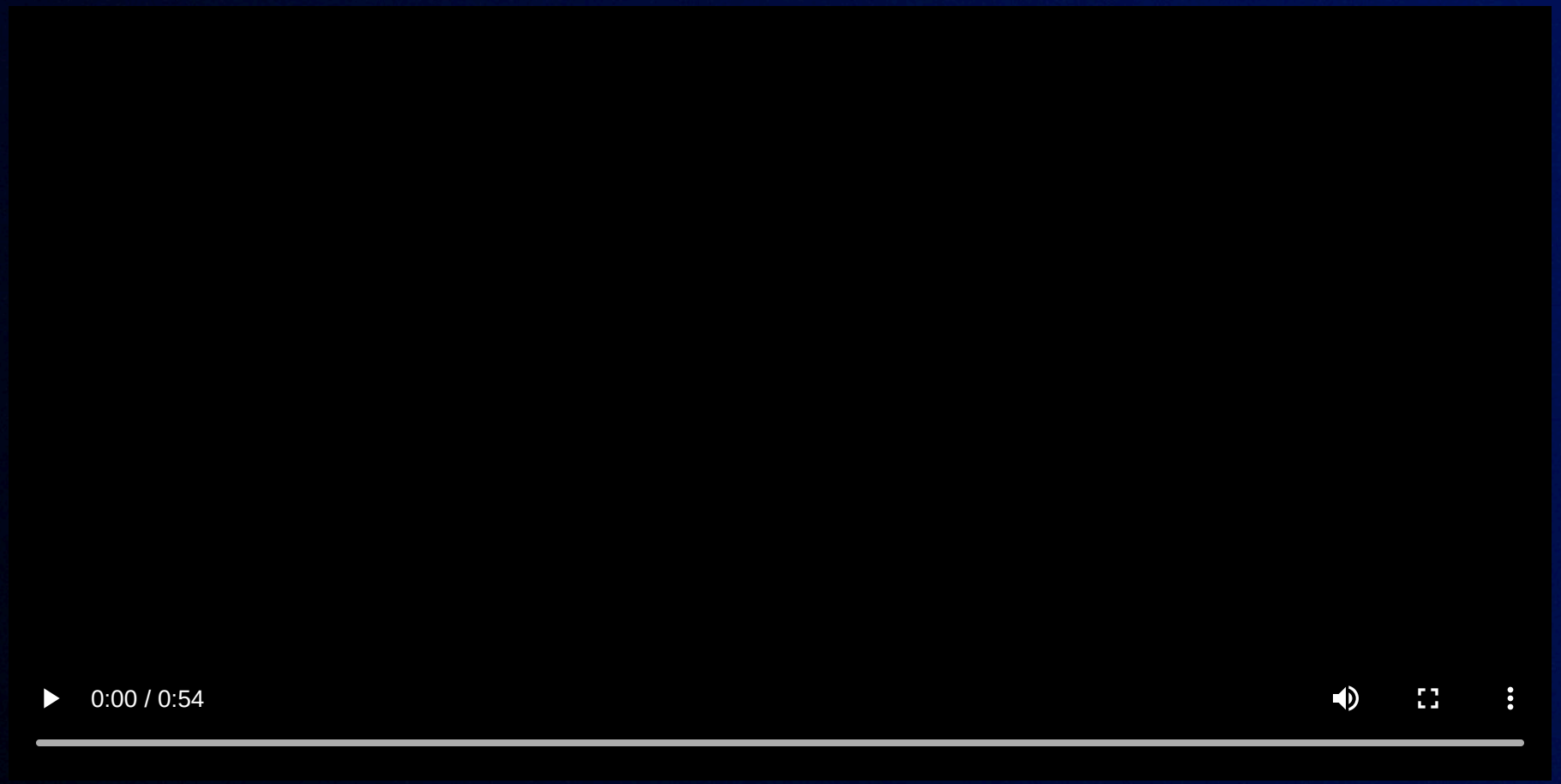
Python Example - Data Processing

```python
import pyarrow.parquet as pq
import s3fs
import pandas as pd

# Read parquet data
parquet_url = "https://s3.waw3-1.cloudferro.com/emodnet/biology/eurobis_occurrence_data/eurobis_occurrences_geoparquet_2024-10-01.parquet"
s3_path = parquet_url.split('s3.waw3-1.cloudferro.com/')[-1]
fs = s3fs.S3FileSystem(endpoint_url="https://s3.waw3-1.cloudferro.com", anon=True)

parquet_file = pq.ParquetFile(s3_path, filesystem=fs)
biodiversity_data = parquet_file.read_row_groups([0]).to_pandas().head(1000)

# Filter and process
marine_data = biodiversity_data[biodiversity_data['scientificName'].str.contains('fish|mollusk|algae', case=False)]
processed_data = marine_data.groupby('scientificName').agg({'decimalLatitude': 'mean', 'decimalLongitude': 'mean'})
```

🎥 Data Analysis using Python scripts

Using your EDITO S3 Storage

🎥 Using MyFiles in an EDITO Service



💾 Saving into EDITO Storage

Your Storage is Ready!

Your personal storage credentials are automatically available in EDITO services!

R Example

```r
# Check credentials and save data
if(Sys.getenv("AWS_ACCESS_KEY_ID") != "") {
  # Process and save data
  processed_data <- marine_data %>% group_by(scientificName) %>% summarise(count = n())
  write.csv(processed_data, "marine_analysis.csv", row.names = FALSE)

  # Upload to storage
  aws.s3::s3write_using(processed_data, FUN = write.csv,
                        bucket = "your-bucket", object = "marine_analysis.csv")
}
```

Python Example

```python
import boto3
import os

# Connect to storage
s3 = boto3.client("s3", endpoint_url=f"https://{os.getenv('AWS_S3_ENDPOINT')}",
                  aws_access_key_id=os.getenv('AWS_ACCESS_KEY_ID'),
                  aws_secret_access_key=os.getenv('AWS_SECRET_ACCESS_KEY'))

# Save and upload data
processed_data.to_csv('marine_analysis.csv', index=False)
s3.put_object(Bucket='your-bucket', Key='marine_analysis.csv',
              Body=processed_data.to_csv(index=False))
```

🎥 Save Data Analysis results to EDITO storage

🎯 Complete Workflow

4 Simple Steps

1. **Find Services** → Go to datalab.dive.edito.eu
2. **Launch Service** → Choose RStudio, Jupyter, or VSCode
3. **Run Analysis** → STAC search, read Parquet data, process results
4. **Save Data** → Upload to your personal storage (MyFiles)

Key Benefits

✅ **Marine Data** - Direct access to EDITO datasets
✅ **Multiple Languages** - R, Python, and more
✅ **Interactive** - Step-by-step guided workflows

🚀 Try It Now!

Get Started in 2 Minutes

1. **Go to**: datalab.dive.edito.eu
2. **Launch RStudio or Jupyter**
3. **Run the code examples** from this presentation
4. **Save your results** to personal storage (MyFiles)

SOS Questions?

Main docs and support

Email: edito-infra-dev@mercator-ocean.eu

Documentation: EDITO Tutorials

Ready to dive into marine data analysis?