



Copernicus

# Standardised and Scalable EO Workflows using openEO offered by Copernicus Data Space Ecosystem

Hans Vanrompay

[dataspace.copernicus.eu](https://dataspace.copernicus.eu)



PROGRAMME OF  
THE EUROPEAN UNION





Copernicus

# Introduction

[dataspace.copernicus.eu](https://dataspace.copernicus.eu)



PROGRAMME OF THE EUROPEAN UNION



# Continental projects powered by openEO

WorldCereal



Land Cover Forest Monitoring (LCFM)



Grasslandwatch

WS: Grasslands and Savannah diversity



Time:  
Tuesday, 11 Feb 2025  
3:00pm - 4:30pm

World Ecosystem Extent Dynamics (WEED)

WS: EBVs for the GBF





Copernicus

# WorldCereal

[dataspace.copernicus.eu](https://dataspace.copernicus.eu)



PROGRAMME OF  
THE EUROPEAN UNION

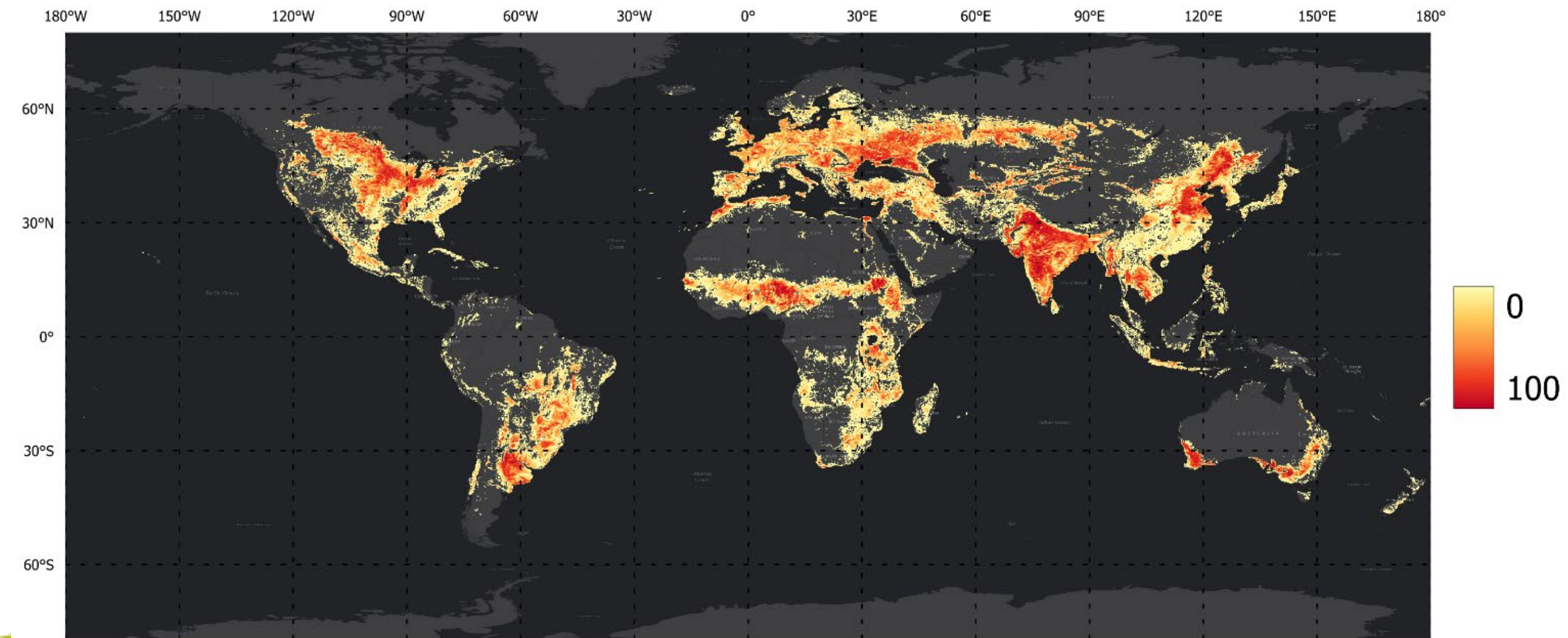


# Project objectives

## Inputs for global monitoring of food and water security

1. Where is food crop production taking place?
2. Which crop is grown when and where? (focus on maize and cereals)
3. Which fields are being irrigated and when?

WorldCereal global temporary crop extent product 2021



Sources: Esri, HERE, Garmin, FAO, NOAA, USGS, ©OpenStreetMap contributors, and the GIS User Community



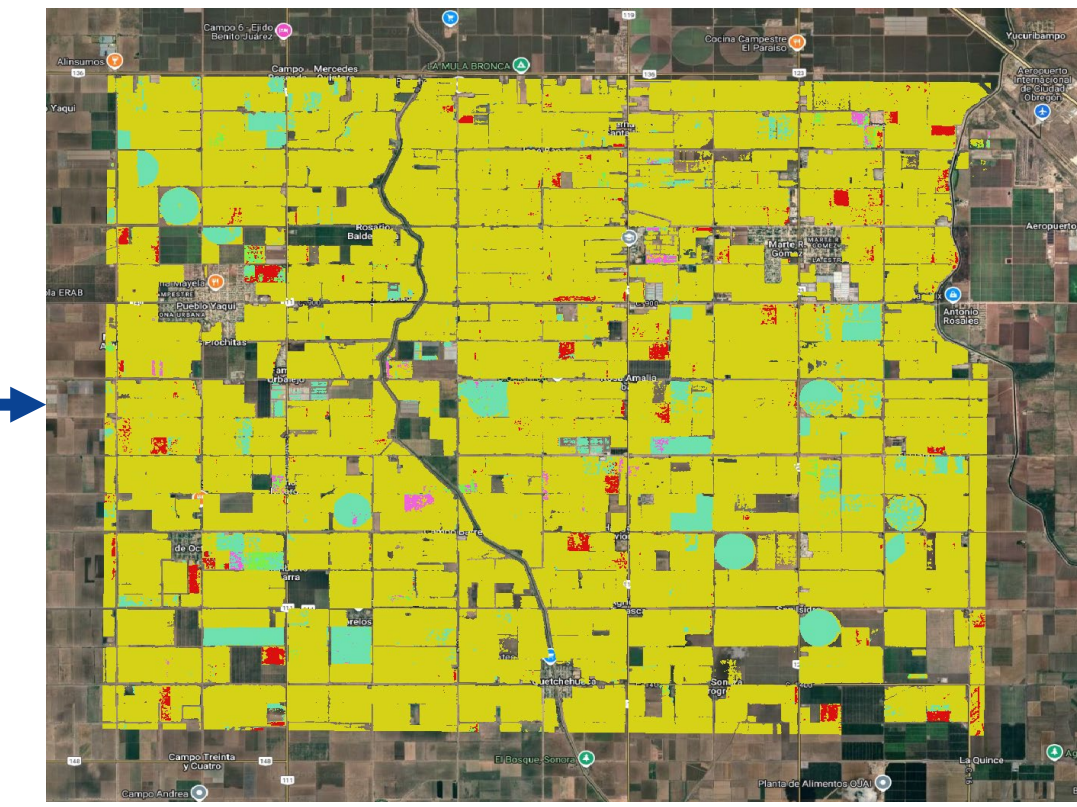
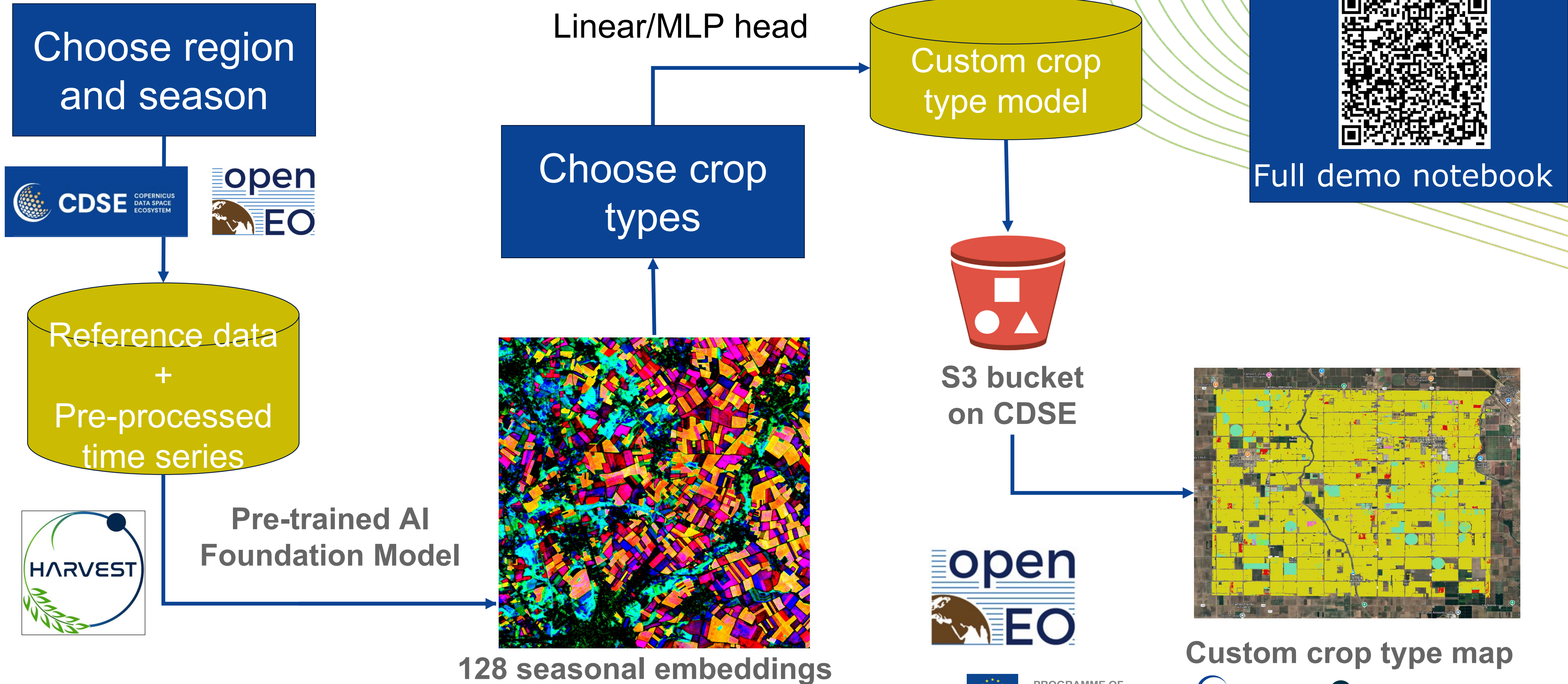
**Demonstrate scientific & technical feasibility of a global and season-specific crop monitoring system at 10m resolution**



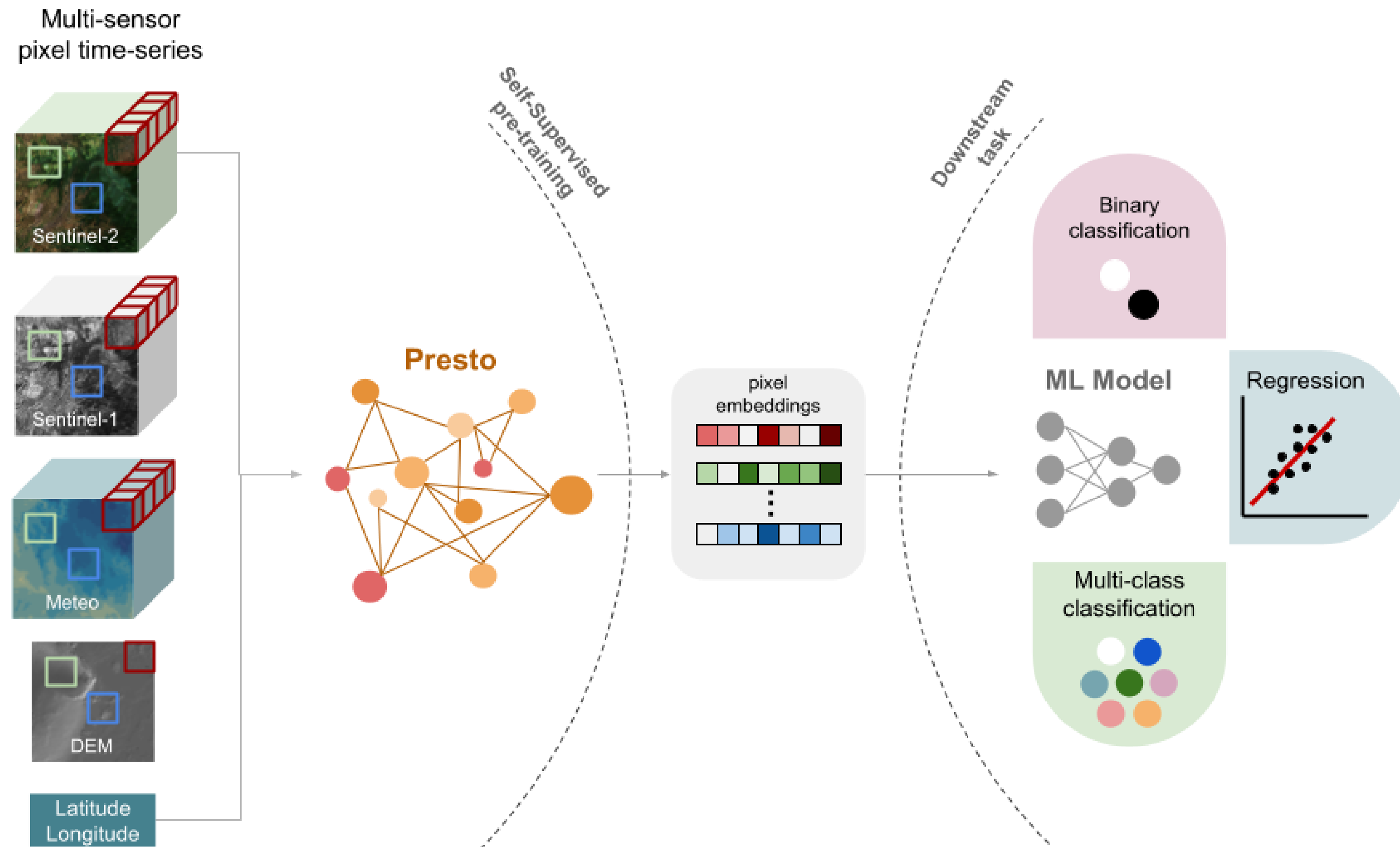
PROGRAMME OF  
THE EUROPEAN UNION




# World Cereal



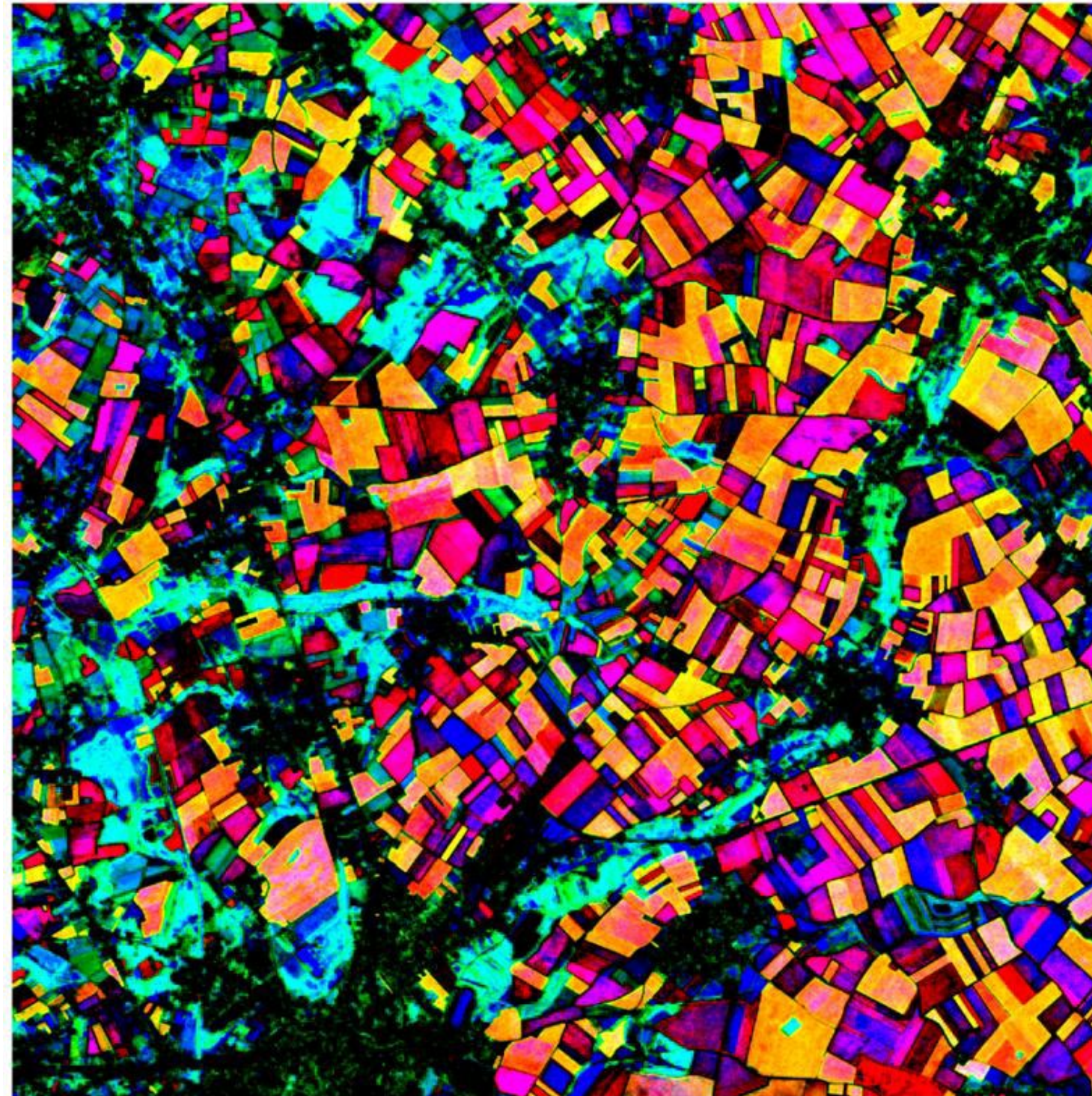
# World Cereal



- Developed by NASA Harvest 
- Learning from **millions of unlabeled data points**
- Representing multi-modal input time series as **one embeddings vector** with size **128**
- Downstream crop mapping training with **high generalizability**
- Updated for WorldCereal
- **Less labels needed**

Tseng, G., Cartuyvels, R., Zvonkov, I., Purohit, M., Rolnick, D., & Kerner, H. (2023). Lightweight, pre-trained transformers for remote sensing timeseries. arXiv preprint arXiv:2304.14065.

# World Cereal



- All satellite and other inputs are **summarized** into a 128 highly informative “embeddings”
- These are the basis for classifying crops in your region
- WorldCereal brings the power of Presto to the user
- You benefit from what has been learned in the past and **train a lightweight model for your needs based on your data**



PROGRAMME OF  
THE EUROPEAN UNION



# World Cereal



[Geospatial Explorer](#)

[worldcereal-classification/notebooks/worldcereal\\_demo\\_embeddings.ipynb at main · WorldCereal/worldcereal-classification](#)

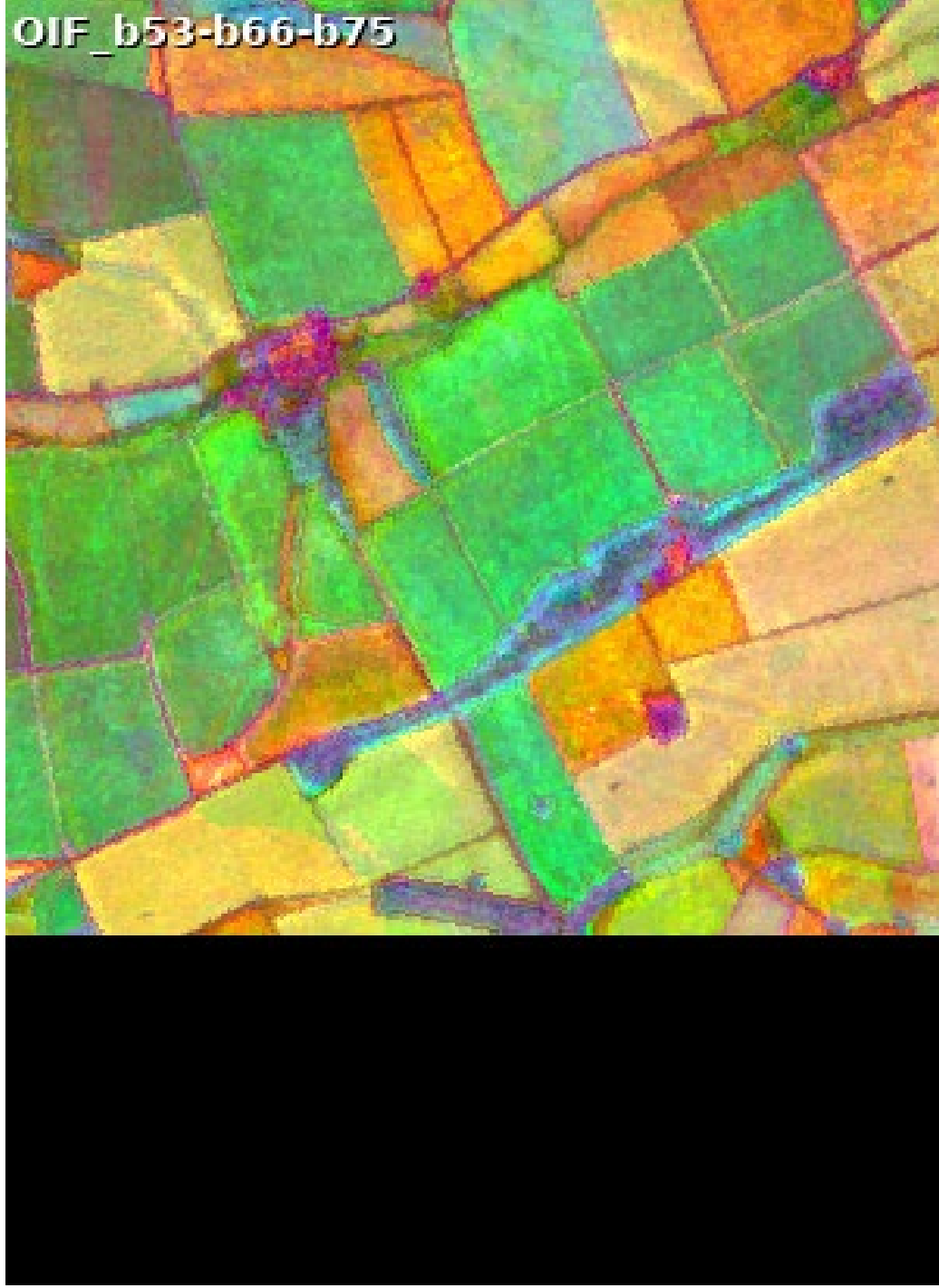
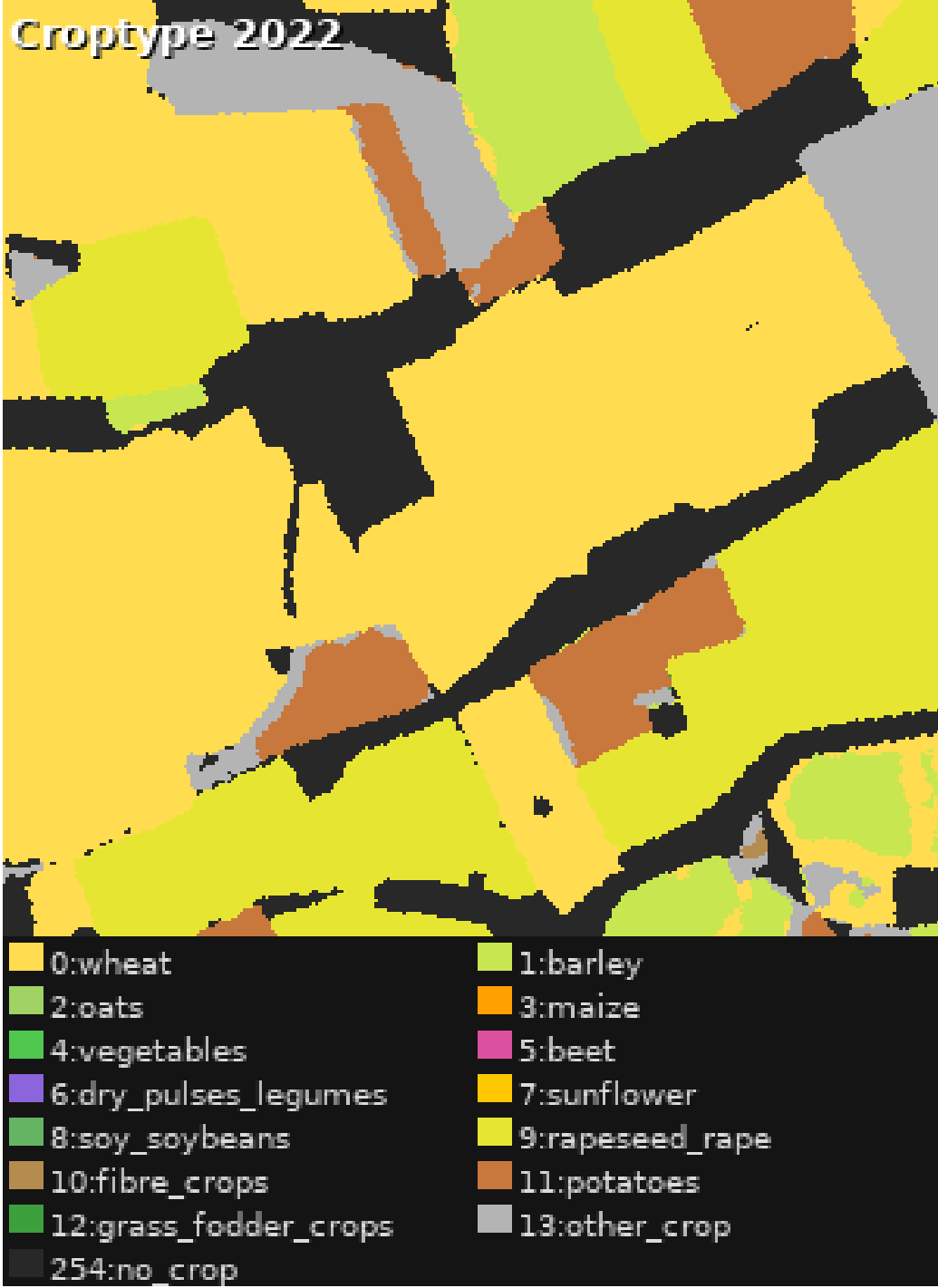


PROGRAMME OF  
THE EUROPEAN UNION



esa

# World Cereal



[worldcereal-classification/notebooks/worldcereal\\_demo\\_embeddings.ipynb](https://worldcereal-classification/notebooks/worldcereal_demo_embeddings.ipynb) at  
[main · WorldCereal/worldcereal-classification](https://main·WorldCereal/worldcereal-classification)

# From project to product with UDPs in APEX

The image displays three algorithm cards from the APEX catalogue, each featuring a satellite image with processing results. The first card, 'ESA worldcereal global crop extent detector', shows a red and black map with a 'Unstable' status. The second, 'ESA worldcereal global crop type detector', shows a multi-colored map with 'Unstable' status. The third, 'Forest Fire Mapping using Random Forest based on Sentinel-2 and Sentinel-1 data', shows a red and black map with 'No Benchmark' status. Each card includes the openEO logo, a description, and logos for CDSE and VITO. The first two cards are powered by CDSE and provided by VITO, while the third is also powered by CDSE and provided by VITO. The first two cards have tags for Agriculture, Sentinel-2, and Sentinel-1, while the third has tags for Natural Hazards, Wildfires, and Sentinel-1.

Algorithm Name	Status	Powered by	Provided by	Tags
ESA worldcereal global crop extent detector	Unstable	CDSE	VITO	Agriculture, Sentinel-2, Sentinel-1, +3
ESA worldcereal global crop type detector	Unstable	CDSE	VITO	Agriculture, Sentinel-2, Sentinel-1, +1
Forest Fire Mapping using Random Forest based on Sentinel-2 and Sentinel-1 data	No Benchmark	CDSE	VITO	Natural Hazards, Wildfires, Sentinel-1, +2

## [APEX Algorithm Catalogue](#)



PROGRAMME OF THE EUROPEAN UNION





Copernicus

openEO - CDSE

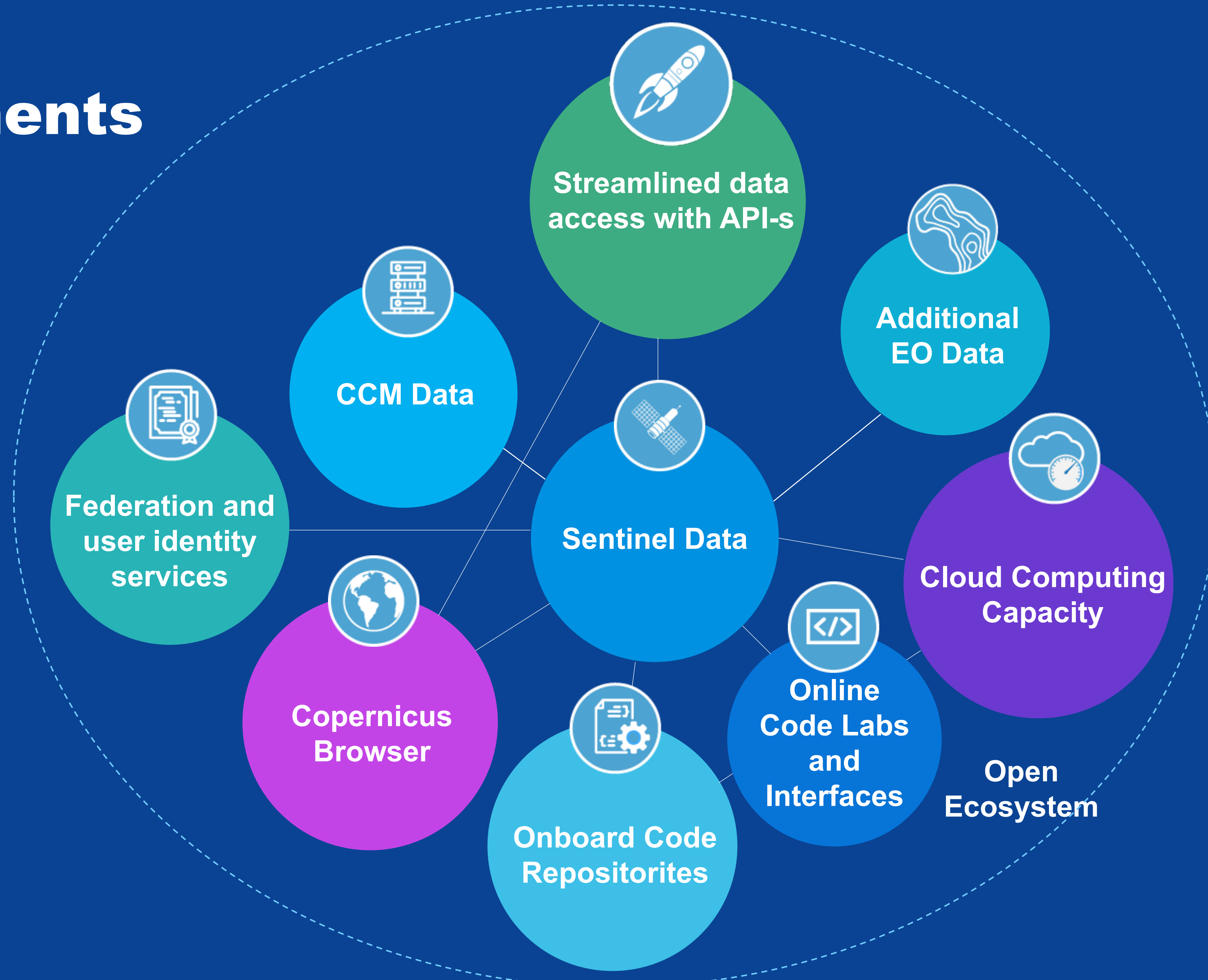
[dataspace.copernicus.eu](https://dataspace.copernicus.eu)



PROGRAMME OF  
THE EUROPEAN UNION



# CDSE components



# Data Access and Processing API: openEO

### Data

This section provides an overview of the EO data available from Copernicus Data Space Ecosystem.

The data offer will gradually extend starting from January 2023.

For the latest information about available satellite data, users and stakeholders can follow them in [Copernicus Sentinel Operations Dashboard](#).

- Sentinel-5P**: The Copernicus Sentinel-5 Precursor mission is the first Copernicus mission dedicated to monitoring our atmosphere.
- Sentinel-6**: Copernicus Sentinel-6 Michael Freilich includes two satellites that will fly sequentially, launched in 2020 and 2025, carrying a state-of-the-art optimized payload.
- Sentinel-2**: The Copernicus Sentinel-2 mission comprises a land monitoring constellation of two polar-orbiting satellites placed in the same sun-synchronous orbit, phased at 180° to each other. It aims at monitoring...
- Sentinel-1**: The Sentinel-1 radar imaging mission is composed of a constellation of two polar-orbiting satellites providing continuous all-weather, day and night imagery for Land and Marine Monitoring, C-band synthetic...
- Sentinel-3**: The main objective of the Copernicus Sentinel-3 mission is to measure ocean and land surface colour, sea and land surface temperature, and sea surface topography with high accuracy and reliability to support ocean forecasting...
- Landsat-5**: The Landsat programme is a joint USGS and NASA-led enterprise...

### APIs

This section gives an overview on the APIs provided by Copernicus Data Space Ecosystem.

#### Catalog APIs

- OData**: OData is an SOI/EC approved, OASIS standard, which is based on https://odata.org/
- OpenSearch Catalog web service**: The OpenSearch catalogue allows
- STAC product catalogue**: STAC (SpatioTemporal Asset Catalog) is a relatively new web

#### Additionally

- Access to EO data via S3**: S3 API is one of the main access methods for EO data. It is suitable for Third Party applications that require high-performance parallel...
- On-Demand Production API**: On-demand processing capability for CARD-BS and CARD-COH6/12 is available in the Copernicus Data Space Ecosystem. This service is offered via a limited pool of resources...
- Traceability Service**: Traceability Service allows the user to track a data product's lifecycle. It acts as a historian of the product's lifecycle, collecting the traces of all related events...

### Applications

This section provides an overview of the EO Applications available from Copernicus Data Space Ecosystem.

- openEO Algorithm Plaza**: The openEO Algorithm Plaza is a marketplace within Copernicus Data Space Ecosystem that allows user to discover and share different Earth...
- About the Browser**: The Copernicus Data Space Ecosystem Browser serves as a central hub for accessing, exploring and utilizing the wealth of Earth observation and...
- Copernicus Data Space Ecosystem Dashboard**: The Copernicus Data Space Ecosystem Dashboard (hereinafter the Dashboard) is...
- About Data Workspace**: The Data Workspace is a valuable tool for managing and...
- JupyterLab**: JupyterLab is an advanced interactive development environment (IDE) that offers a flexible and feature-rich interface for working with notebooks, code, and data. It allows...
- Sentinel Hub QGIS Plugin**: The Sentinel Hub QGIS Plugin allows you to view satellite image data from the Copernicus Data Space Ecosystem or from Sentinel Hub directly within a QGIS workspace. All...
- openEO Web Editor**: The openEO Web Editor is a web-based graphical user interface (GUI) that allows users to interact with the openEO API and perform various tasks related to Earth observation...

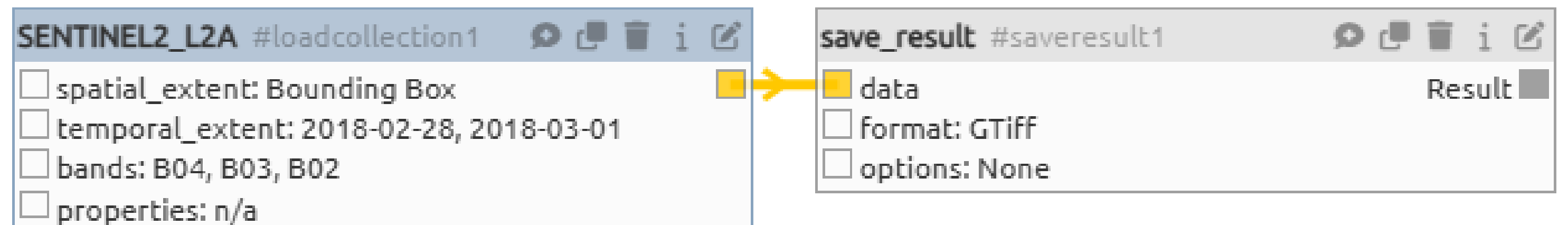
# What is openEO?

*It is an **open-source API** that provides standardised access to EO data and simplifies the deployment of processing workflows as **scalable, cost-efficient services**.*



# What is openEO?

```
{
  "process_graph": {
    "loadcollection1": {
      "arguments": {
        "bands": ["B04", "B03", "B02"],
        "id": "SENTINEL2_L2A",
        "spatial_extent": {"east": -76.19, "north": 3, "south": 2.99, "west": -76.2},
        "temporal_extent": ["2018-02-28", "2018-03-01"]
      },
      "process_id": "load_collection"
    },
    "saveresult1": {
      "arguments": {
        "data": {
          "from_node": "loadcollection1"
        },
        "format": "GTiff",
        "options": {}
      },
      "process_id": "save_result",
      "result": true
    }
  }
}
```



# Why openEO?

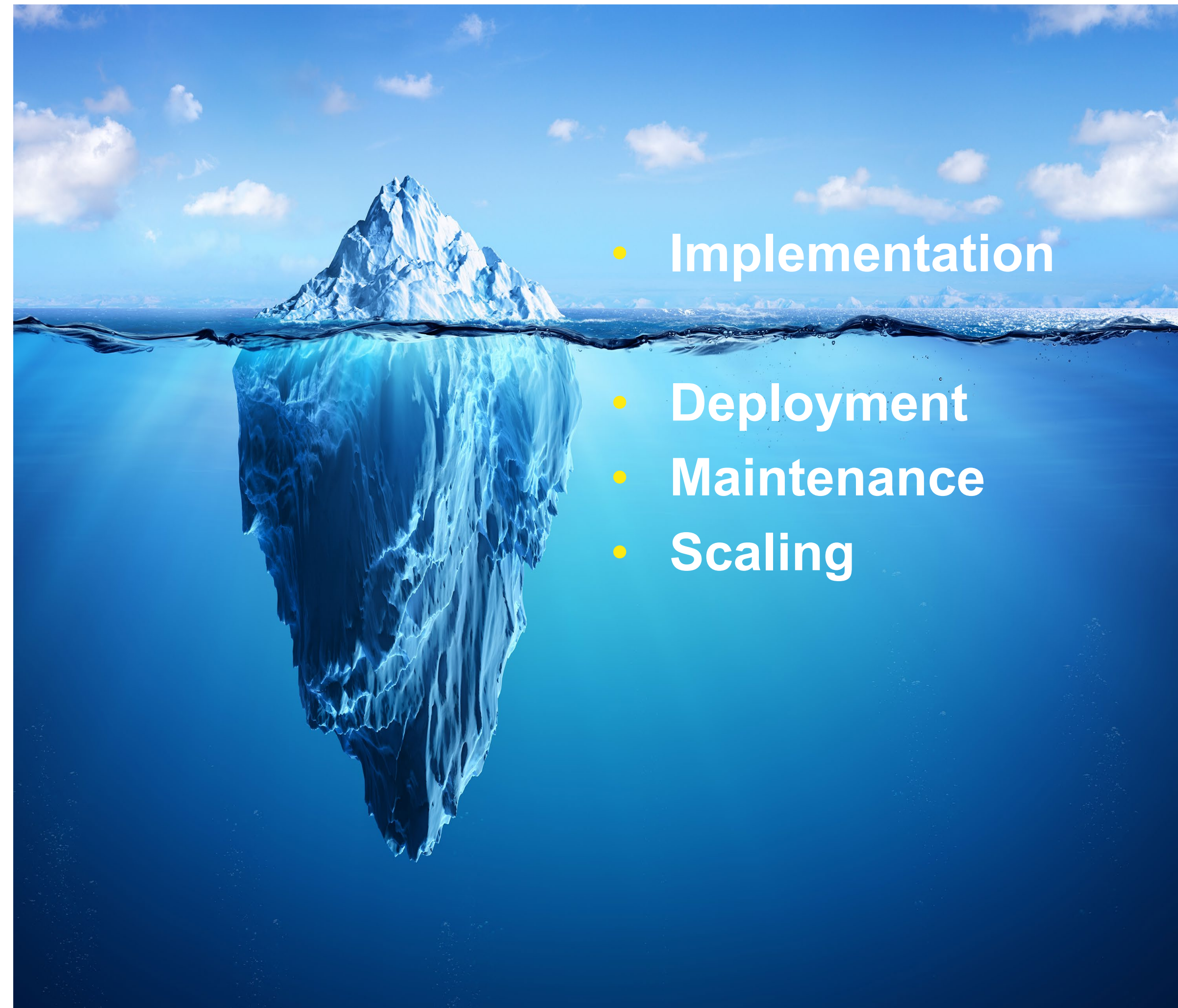
- **Simple data access & processing** for multiple Earth observation datasets
- Scalable and efficient **cloud computation**
- A **standardized system** that works across different platforms
- **Focus on the task at hand**, not the data engineering
- Supporting principles of **FAIR** (Findable, Accessible, Interoperable, and Reusable) data and **Open Science**

openEO instance in Copernicus Data Space Ecosystem

# openEO



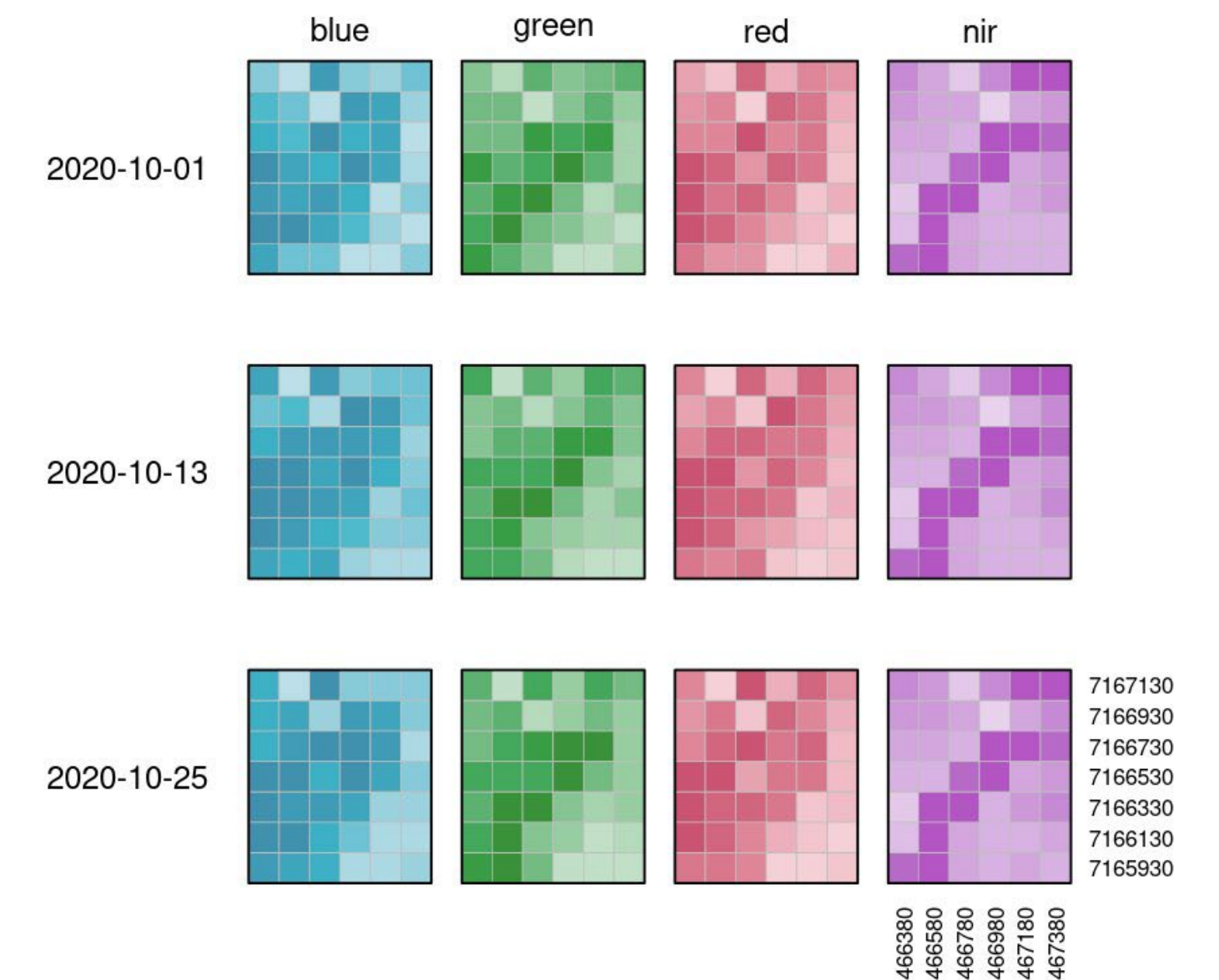
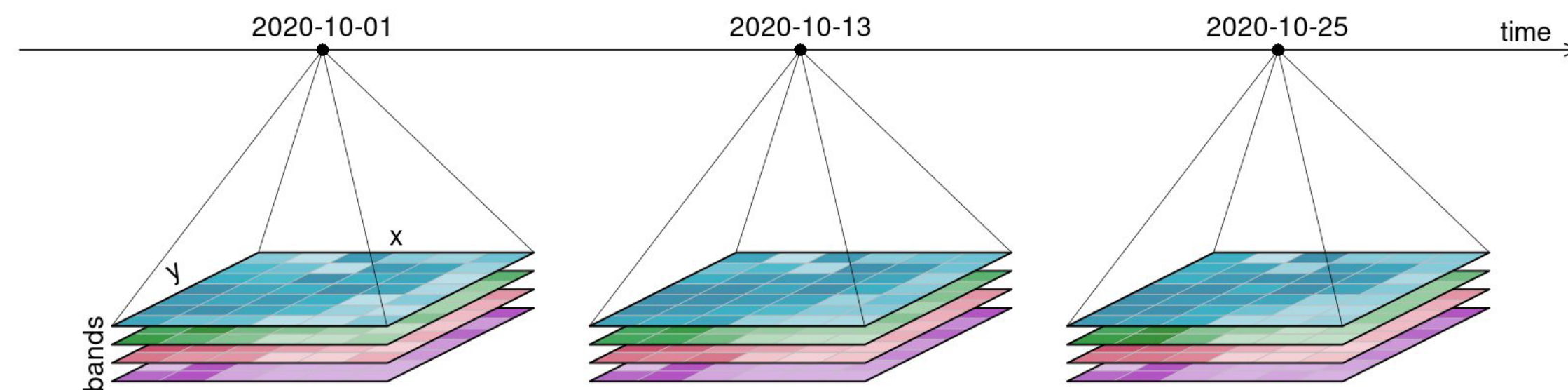
# openEO



- Implementation
- Deployment
- Maintenance
- Scaling

# openEO concept: Datacube

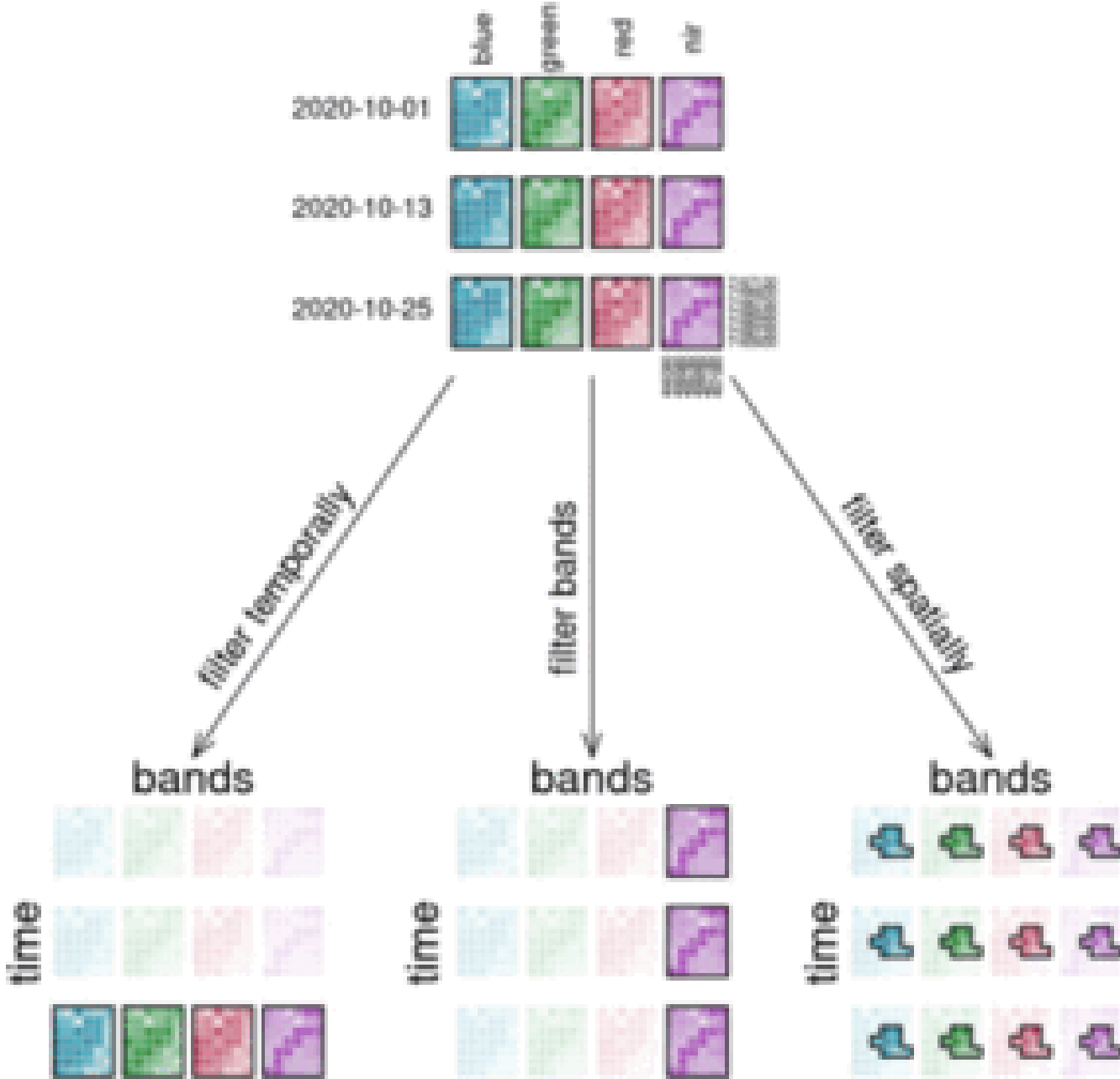
- multidimensional arrays with one or more spatial or temporal dimensions
- Data in OpenEO is represented in this way
- Any representation of the data cube is fine (meaning – dimensions can be switched in display)



# openEO concept: Processes

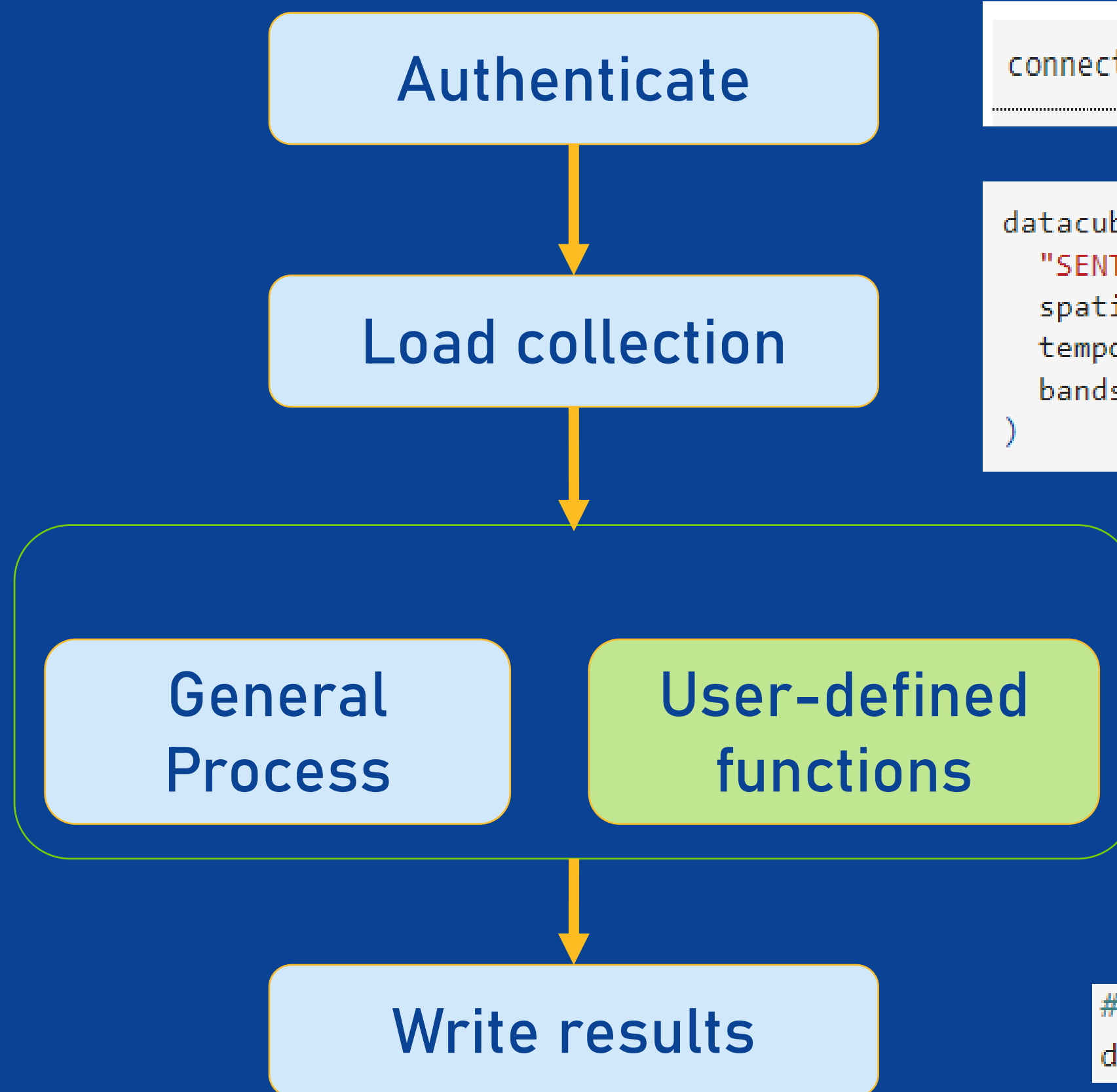
Apply processes on your datacube

- Apply (values, kernels, temporal, ...)
- Resample
- Filter
- Aggregate
- Reduce
- ...



# openEO Workflow

- Built-in OpenEO processes
- User Defined Functions (UDF)



```
connection = openeo.connect("openeo.dataspace.copernicus.eu").authenticate_oidc()
```

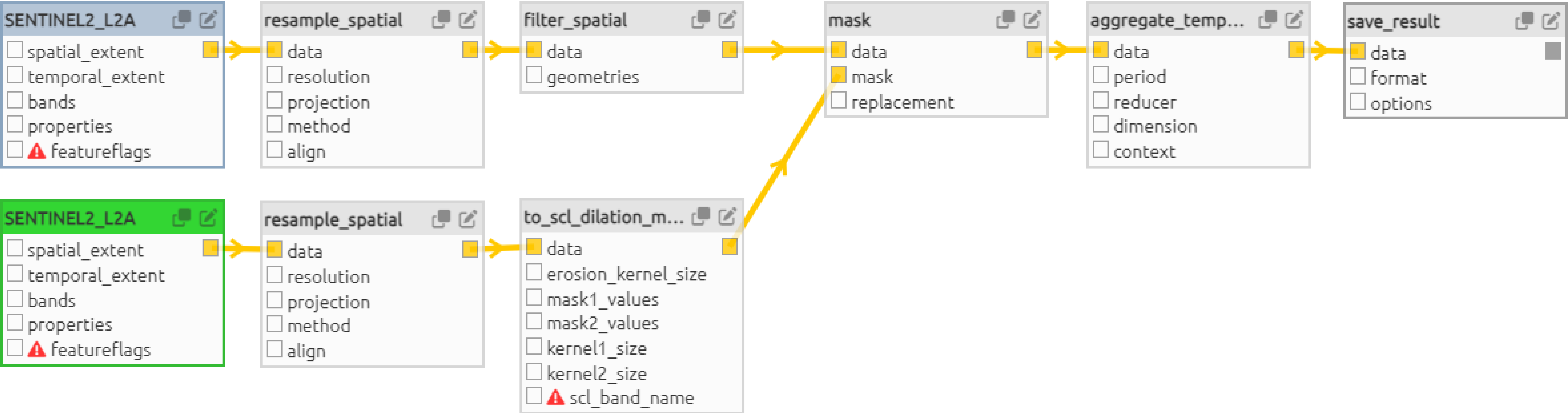
```
datacube = connection.load_collection("SENTINEL1_GRD", spatial_extent={"west": 16.06, "south": 48.06, "east": 16.65, "north": 48.35}, temporal_extent=["2017-03-01", "2017-04-01"], bands=["VV", "VH"])
```

<https://processes.openeo.org/>  
<https://open-eo.github.io/openeo-python-client/udf.html>

```
datacube = datacube.min_time()
```

```
# Let's download the data  
datacube.execute_batch(title="Simple execution", outputfile="Sentinel1.nc")
```

# Constructing a Process graph (LCFM)



# Practical Exercise

[openeo-community-examples/python/1\\_GettingStarted/GettingStarted.ipynb at main · Open-EO/openeo-community-examples](https://github.com/Open-EO/openeo-community-examples/blob/main/python/1_GettingStarted/GettingStarted.ipynb)

# With openEO

- Simple data access & processing for multiple Earth observation datasets
- Scalable and efficient processing capabilities
- A standardized system that works across different platforms
- Independence from underlying technologies and software libraries
- Supporting principles of FAIR (Findable, Accessible, Interoperable, and Reusable) data and Open Science, e.g. transparent workflows



Copernicus

# Advanced Workflows

[dataspace.copernicus.eu](https://dataspace.copernicus.eu)



PROGRAMME OF  
THE EUROPEAN UNION



# Continental projects powered by openEO

WorldCereal



Land Cover Forest Monitoring (LCFM)



Grasslandwatch

WS: Grasslands and Savannah diversity



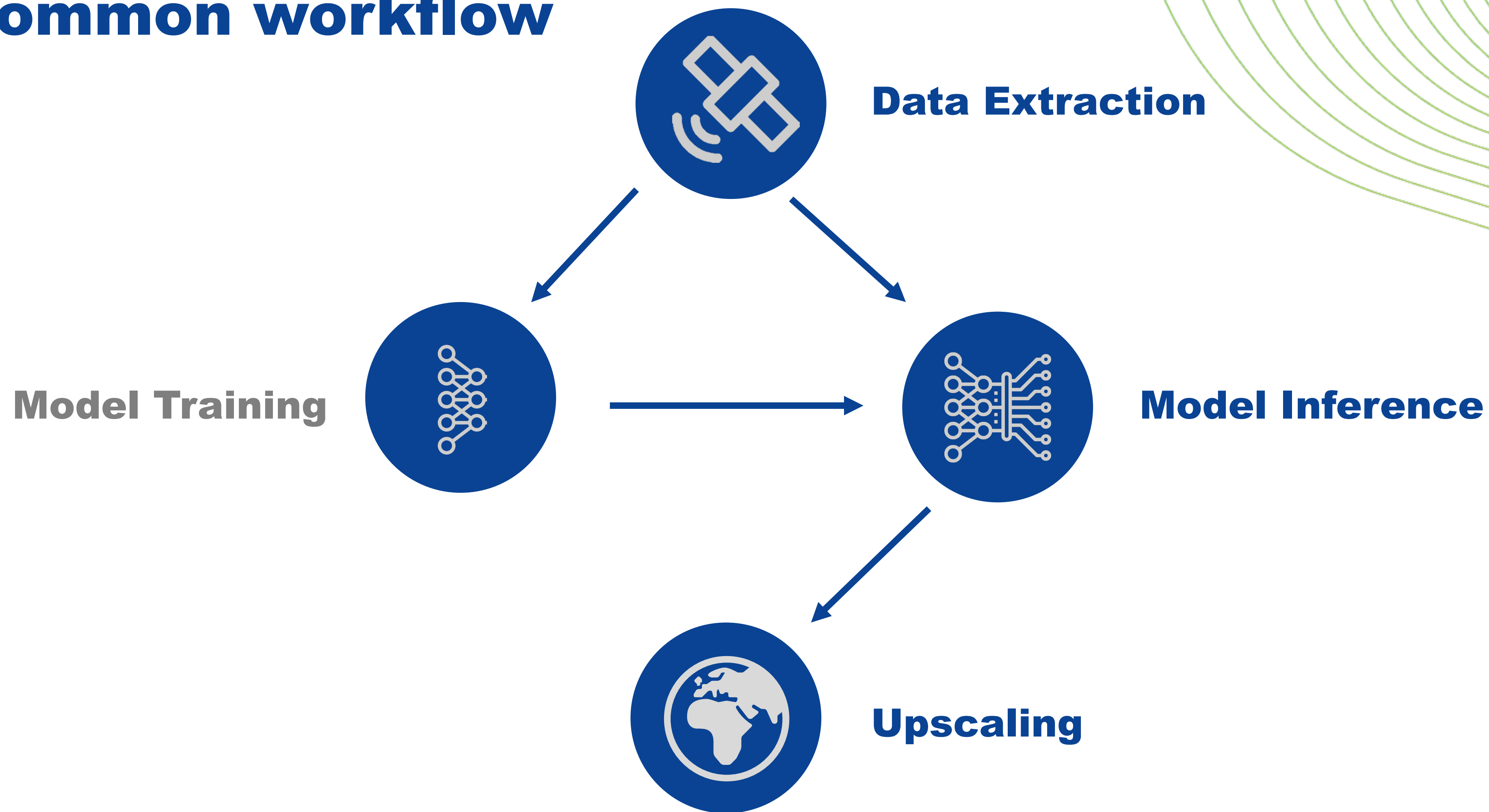
Time:  
Tuesday, 11 Feb 2025  
3:00pm - 4:30pm

World Ecosystem Extent Dynamics (WEED)

WS: EBVs for the GBF



# Common workflow





Copernicus

# Data Extraction

[dataspace.copernicus.eu](https://dataspace.copernicus.eu)



PROGRAMME OF  
THE EUROPEAN UNION





# Data extraction

1. Create an openEO processing pipeline
2. Create a dataframe containing the input parameters for the areas of interest
3. Efficiently scale up the execution with openEO's MultiBackendJobManager

[openeo-community-examples/python/ManagingMultipleLargeScaleJobs](https://openeo-community-examples/python/ManagingMultipleLargeScaleJobs) at main · Open-EO/openeo-community-examples



Copernicus

# Model Inference

[dataspace.copernicus.eu](https://dataspace.copernicus.eu)



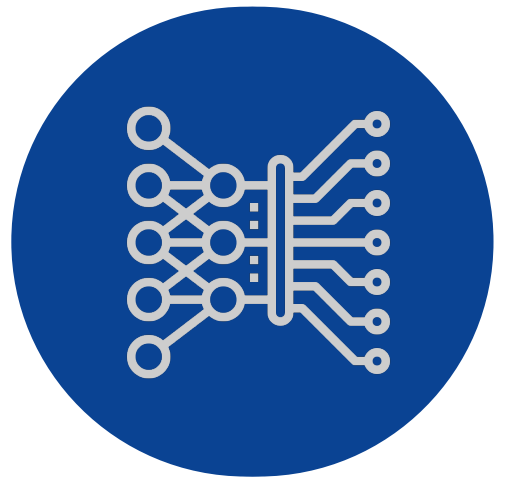
PROGRAMME OF  
THE EUROPEAN UNION



# Model Inference

1. Convert your model to ONNX
2. Create a Model Specific UDF
3. Create an openEO job which executes the UDF

[openeo-community-examples/python/OnnxMLInference/Onnx ML Inference.ipynb](https://openeo-community-examples.python-onnxmlinference.onnx_ml_inference.ipynb) at main · OpenEO/openeo-community-examples



# User-Defined Function

```
def apply_datacube(cube: xr.DataArray, context: Dict) -> xr.DataArray:
    """
    Run the inference on the given input data using the provided ONNX runtime session.
    This method is called for each timestep in the chunk received by apply_datacube.
    """
    # Load the ONNX model
    ort_session = _load_ort_session("test_model.onnx") # name of the model in the archive

    # Make sure the input dimensions are in the expected order and save the original shape
    input = cube.transpose("bands", "y", "x")

    # Get the underlying np.ndarray and reshape it to the expected shape.
    output = ort_session(input)

    # Return the output as an xarray DataArray
    return xr.DataArray(
        output.reshape(input.shape), # Reshape the output to the original shape (bands, y, x)
        dims=["bands", "y", "x"],
    )
```

# Managing Dependencies

- External Python packages may not be available in the base environment

→ Inline dependency specification (PEP 723)

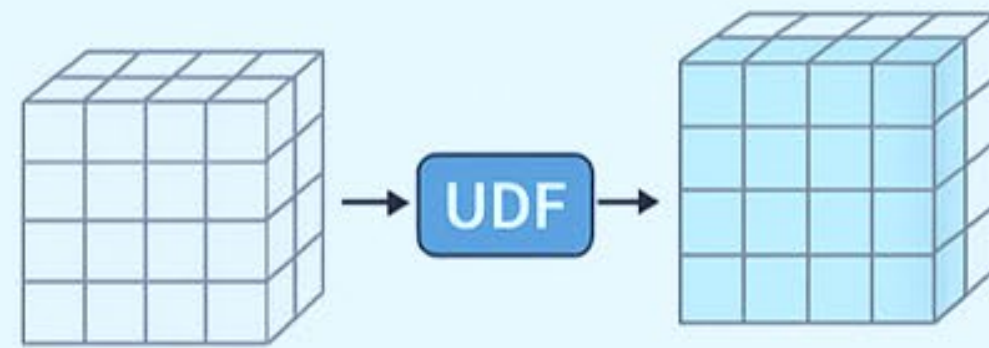
- Alternative: Create custom dependencies:
  - Create a virtual environment with required packages.
  - Upload the zip file to an accessible URL.
  - Specify the archive in the job options

```
# /// script
# dependencies = [
#     "numpy",
#     "scikit-learn",
# ]
# ///
```

# Applying the UDF

## apply

Applies a UDF to each pixel individually



```
datacube = datacube.apply(my_udf)
```

## apply\_dimension

Applies a UDF along a specified dimension



```
datacube = datacube.apply_dimension(dimension='t', process=my_udf)
```

## reduce\_dimension

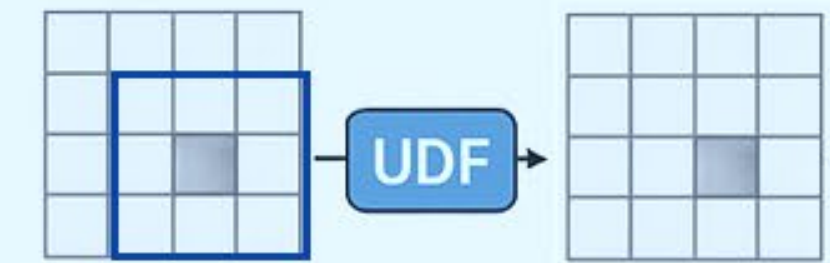
Applies a UDF along a specified dimension, after which it collapses



```
datacube = datacube.reduce_dimension(dimension='t', reducer=my_udf)
```

## apply\_neighborhood

Applies a UDF to each pixel considering its neighborhood



```
datacube = datacube.apply_neighborhood(process=my_udf, size=[...], overlap=[...])
```



Copernicus

# Standardisation

[dataspace.copernicus.eu](https://dataspace.copernicus.eu)



PROGRAMME OF  
THE EUROPEAN UNION



# From project to product with UDPs in APEX

openEO ● Unstable

### ESA worldcereal global crop extent detector

Detects crop land at 10m resolution, trained for global use. Based on Sentinel-1 and 2 data, this algorithm can be ...

Powered by Provided by

Agriculture Sentinel-2 Sentinel-1 +3

openEO ● Unstable

### ESA worldcereal global crop type detector

A crop type detection algorithm for global use on a limited number of crop types. Detects crops at 10m resolution. ...

Powered by Provided by

Agriculture Sentinel-2 Sentinel-1 +1

openEO ● No Benchmark

### Forest Fire Mapping using Random Forest based on Sentinel-2 and Sentinel-1 data

An openEO process example for mapping forest fires using Random Forest based on Sentinel-2 and Sentinel-1 data.

Powered by Provided by

Natural Hazards Wildfires Sentinel-1 +2

## [APEX Algorithm Catalogue](#)



PROGRAMME OF  
THE EUROPEAN UNION



# What's Next?



PROGRAMME OF  
THE EUROPEAN UNION



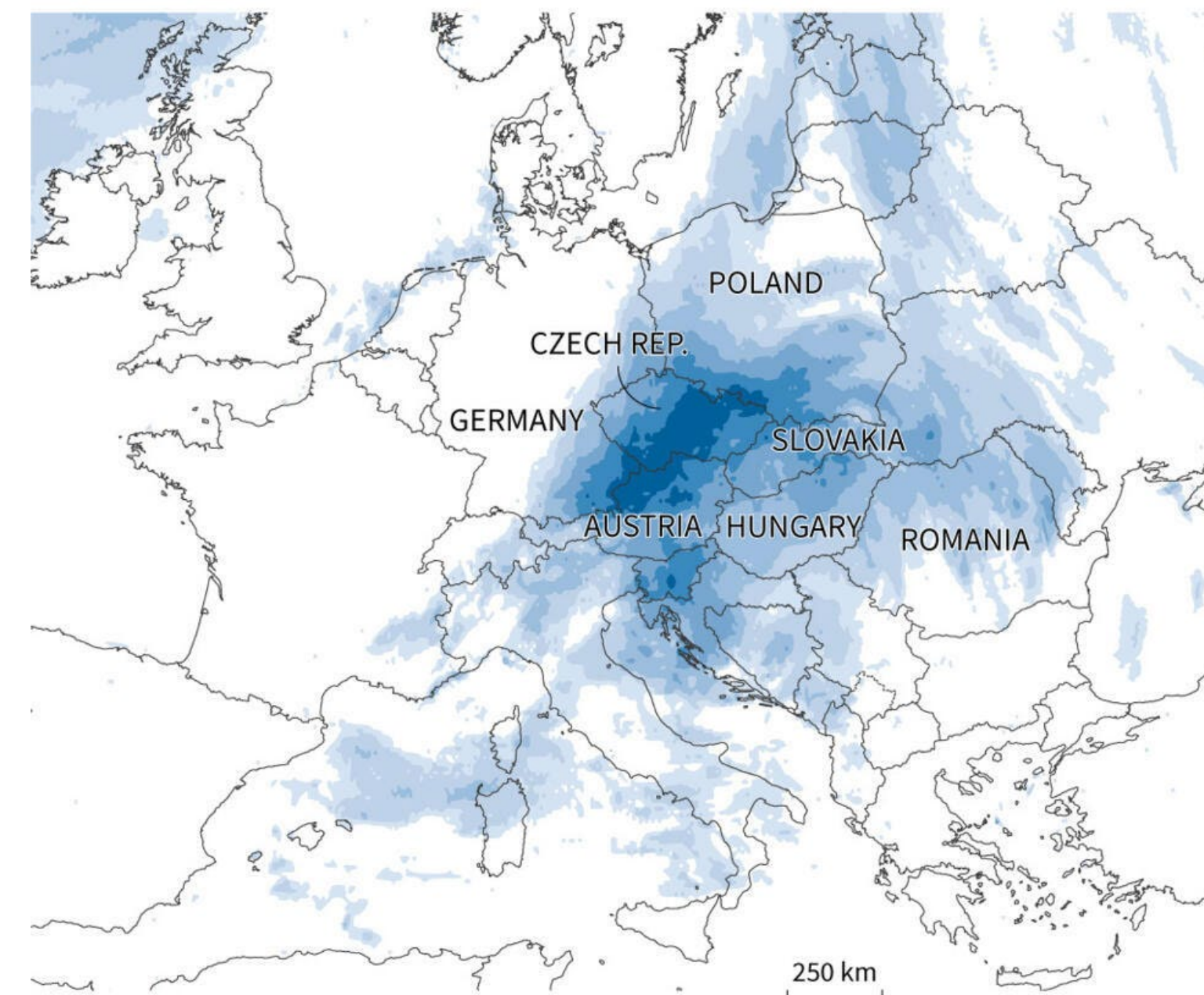
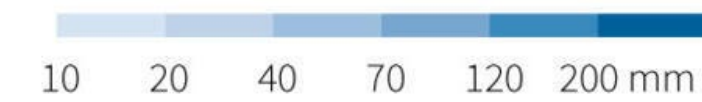
# Introduction

You are a rescue worker with less than 12 hours to decide; to evacuate or not?



## Storm Boris

3-day accumulated rain, as of September 14



Source: NASA Global Precipitation Measurement



What if we could go seamlessly from data to decision?



PROGRAMME OF  
THE EUROPEAN UNION



# Solution

- **AI powered agent that:**
  - Translates user-intent into openEO code
  - Validates and executes the code
  - Explainability of open-source code
  - Returns on demand maps

Hi, Hans Vanrompay!



Type your message...

Hide Tool Calls

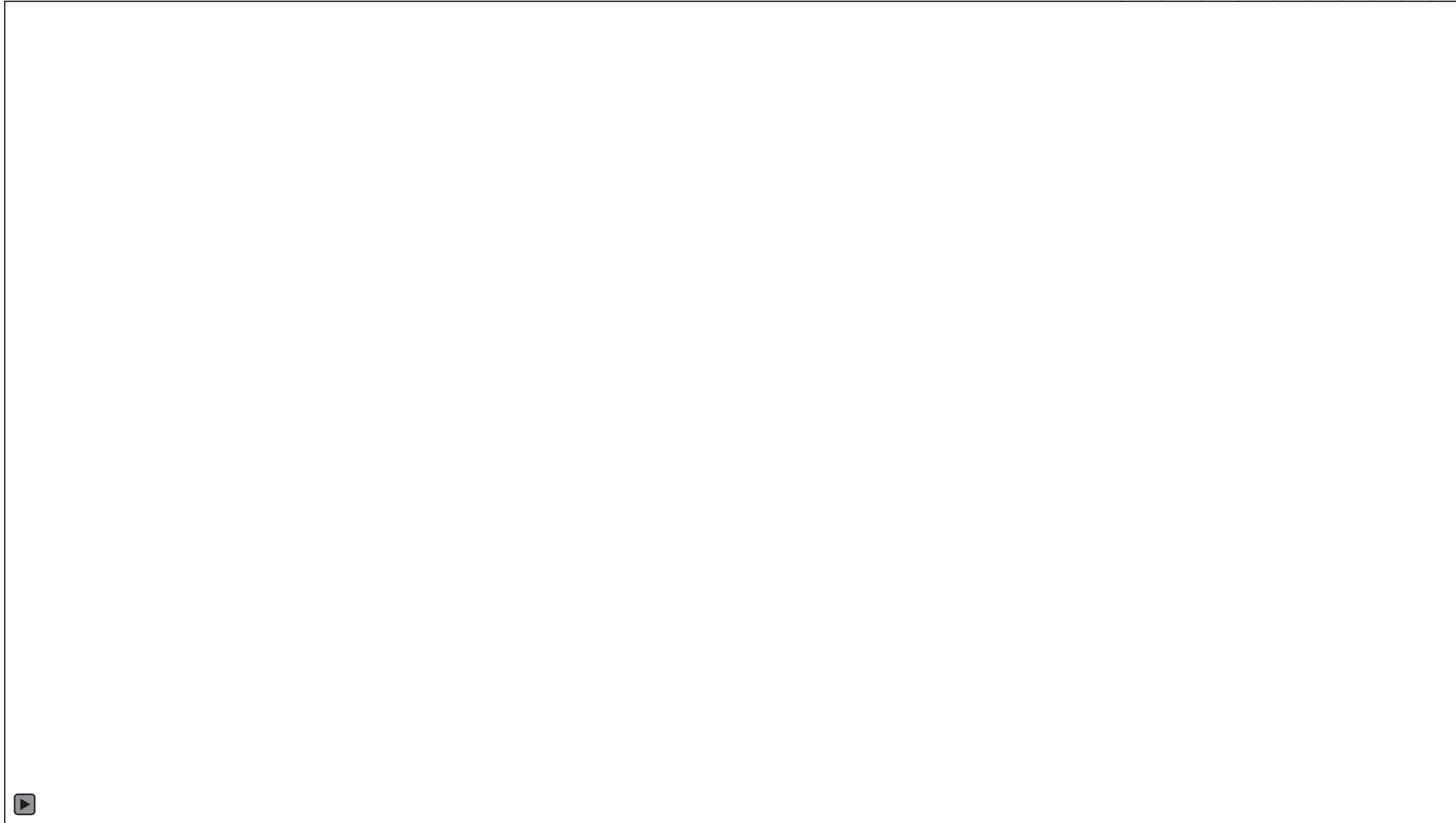
Send



PROGRAMME OF  
THE EUROPEAN UNION



# Solution



PROGRAMME OF  
THE EUROPEAN UNION



# AI meets openEO



PROGRAMME OF  
THE EUROPEAN UNION





PROGRAMME OF  
THE EUROPEAN UNION



# Thank You



PROGRAMME OF  
THE EUROPEAN UNION



## Relevant links:

- **Forum:** <https://forum.dataspace.copernicus.eu/>
- **Documentation:** <https://documentation.dataspace.copernicus.eu/Home.html>
- **Submit a ticket:** <https://helpcenter.dataspace.copernicus.eu/hc/en-gb/requests/new>
- **openEO endpoint:** “openeo.dataspace.copernicus.eu”
- **openEO web-editor:** <https://openeo.dataspace.copernicus.eu/>
- **openEO Python client documentation:** <https://open-eo.github.io/openeo-python-client/index.html>
- **openEO community examples:** <https://github.com/Open-EO/openeo-community-examples/tree/main/python>
- **openEO Algorithm Plaza:** <https://marketplace-portal.dataspace.copernicus.eu/catalogue>
- **Youtube videos:** <https://www.youtube.com/@copernicusdataspaceecosystem>
- **FAQ:** <https://documentation.dataspace.copernicus.eu/FAQ.html>

