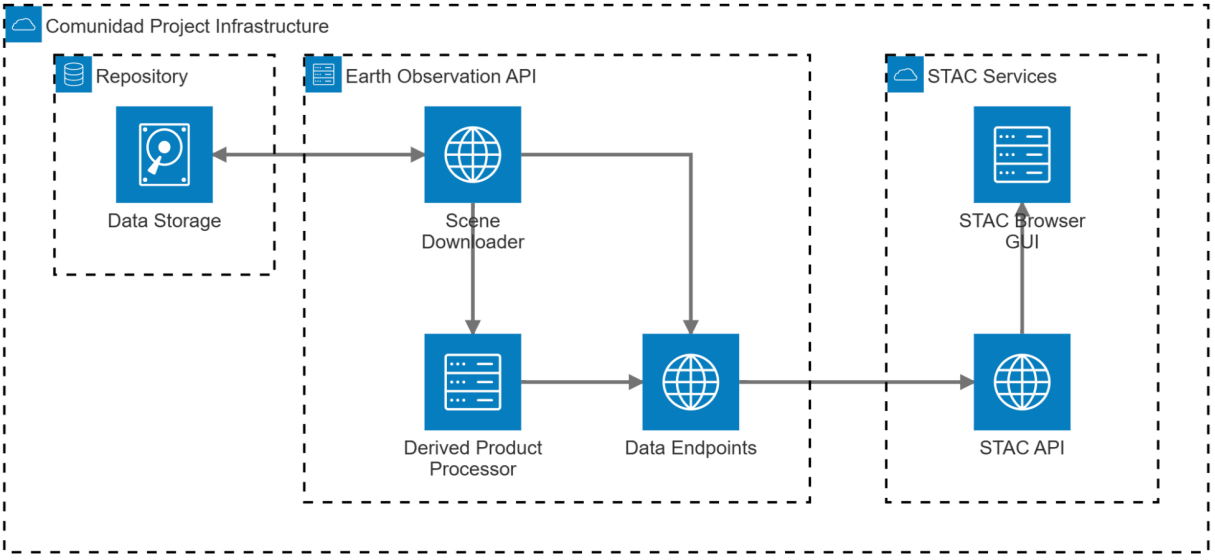
Tutorial II

How to get and work with satellite data

This tutorial describes the Comunidad project infrastructure tools for managing satellite data and its derived products. We'll walk through the whole data pipeline — from downloading raw Sentinel scenes, through generating derived products, to publishing everything via an OGC STAC API and browsing it using a graphical STAC Browser.

Architecture Overview

The Comunidad system is built around three core components:



1. **A repository of Sentinel scenes**

At the foundation is the repository of Sentinel scenes.
This repository stores raw satellite data, such as Sentinel-1 radar data or Sentinel-2 multispectral imagery.
The repository uses a file-system structure by location, year, and sensor type to ensure fast retrieval and clean organisation.

2. **An Earth Observation API** for managing downloads and further processing the data. This service has two key roles:

- Downloading and managing Sentinel scenes
- Calculating derived raster products

When you request the API to download any data or calculate, for example, an enhanced vegetation index, the raster data will be stored in the data repository, where it can be accessed via a set of API data endpoints.

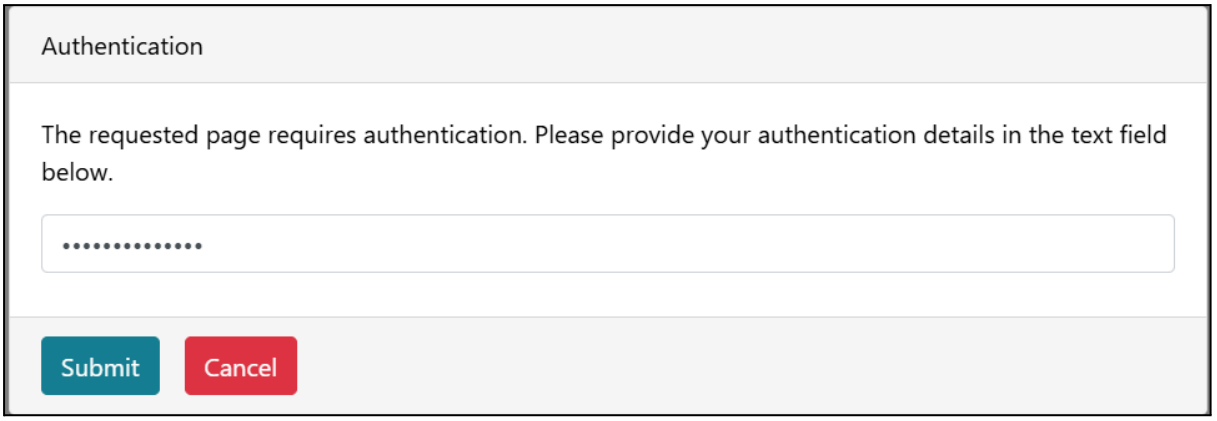
3. STAC API - the SpatioTemporal Asset Catalog

To make all stored data easily accessible, Comunidad implements a **STAC API,** which is an OGC standard for providing satellite datasets. This API describes every scene and every derived product as an **STAC Item**, grouped inside **Collections**.
It ensures interoperability with GIS tools, cloud platforms, and other EO dashboards.

The STAC API exposes:

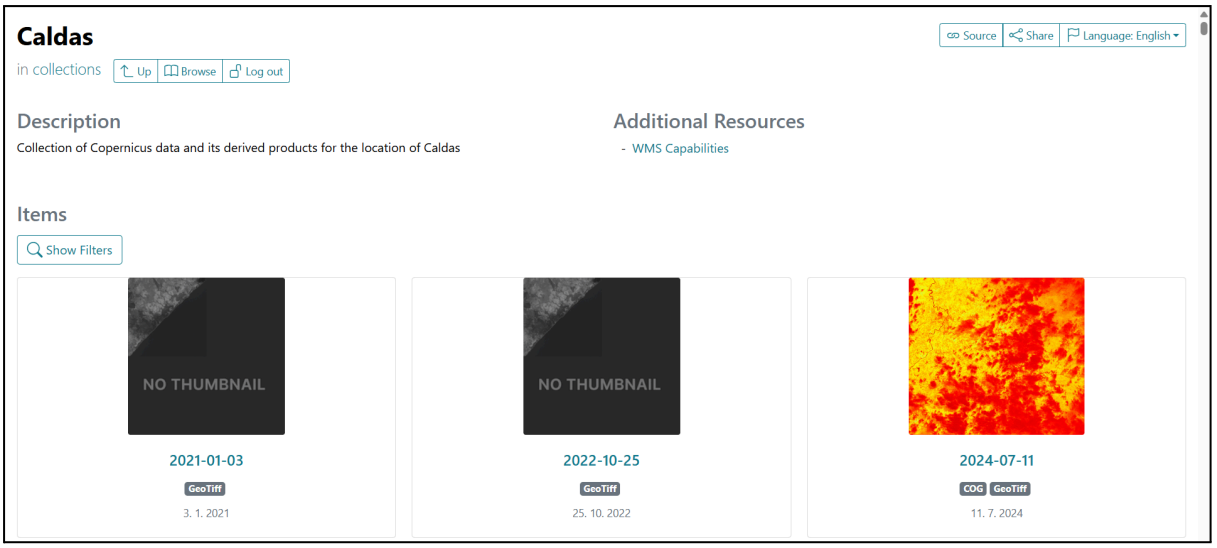
- Raw Sentinel scenes
- Derived index rasters

Then there is a STAC Browser for easy visual exploration of all the data stored in the repository. This data viewer, designed for interactive viewing and searching, can be found on the Comunidad web portal at https://comunidad-project.eu/eo/stac/. Alternatively, it can be accessed from the Comunidad website via the Platform menu item and selecting Satellite data.

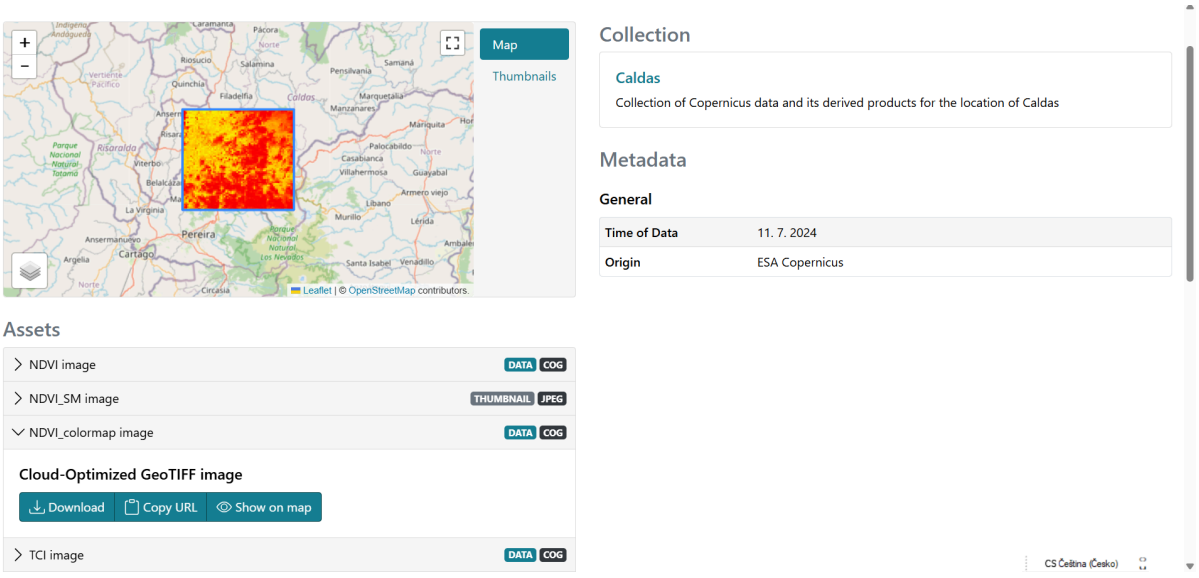


You will be prompted for the API key. Enter: ***eo-api-key-dev***

Here, data is sorted according to the STAC specification into so-called collections. Each collection corresponds to one location for which data is available in the repository. Collections contain sets of STAC Items divided according to the date of their acquisition.



Inside the STAC Item, you can find all the available raster images for the selected date. You can preview them in an integrated map or download the raw data.



How to work with the APIs

Let's now show some examples of how to work with these APIs. Documentation for the Earth Observation API can be found here:
https://comunidad-project.eu/eo-api/docs

The STAC API is completely described on the website of its authors:
https://github.com/radiantearth/stac-api-spec?tab=readme-ov-file#stac-api-foundation-specifications

Prepare the satellite data

We will start by downloading Sentinel-2 images. We will send an HTTP request to the Earth observation API, which initiates an asynchronous job that downloads the requested data to the repository and reports the job status via the /jobs endpoint.
The input is expected to be the name of the location (for classification into STAC collections) and a geometry in WKT format that specifies the boundaries of the area of interest. And finally, a time range.

```
POST https://comunidad-project.eu/eo-api/download

Content-Type: application/json

x-api-key: eo-api-key-dev

{

    "product": "Sentinel-2",

    "location_name": "Caldas",

    "start_date": "2025-11-05",

    "end_date": "2025-11-07",

    "geometry": "POLYGON((14.0 46.0, 14.5 46.0, 14.5 46.5, 14.0 46.5, 14.0 46.0))"

}
```

Generate derived products

The EO API is also responsible for generating **derived products**, such as NDVI, EVI, SAVI, or other vegetation indices. The API can also be easily extended to add any custom vegetation or environmental index. After calculation, the outputs are saved back into the repository alongside the raw data, ready for visualisation or further analysis.

The following request will start a job to calculate NDVI for a specified location and date range:

```
POST https://comunidad-project.eu/eo-api/indices/ndvi

Content-Type: application/json

x-api-key: eo-api-key-dev

{

    "location_name": "Caldas",

    "start_date": "2025-11-05",

    "end_date": "2025-11-07",

    "api_key": "eo-api-key-dev"

}
```

Access the data via STAC endpoints

First, list all the collections in the repository. Each collection corresponds to one location. Remember that *location_name* is one of the parameters when you download the satellite data.

```
GET
https://comunidad-project.eu/eo-api/stac/collections?api_key=eo-api-key-dev
```

To get the list of STAC Items contained in the location “Caldas”, call:

```
GET
https://comunidad-project.eu/eo-api/stac/collections/Caldas/items?api_key=eo-api-key-dev
```

And to access all the assets (raster images) in that STAC Item (date), send the following request:

```
GET
https://comunidad-project.eu/eo-api/stac/collections/Caldas/items/2025-11-05?api_key=eo-api-key-dev
```

You will receive all the URLs for the specific raster images, which you can then download and use locally for your own analyses.