



*Research Lifecycle Management technologies for  
Earth Science Communities and Copernicus users in EOSC*

## Deliverable D4.3

### Data Cube API – Release II

Grant agreement number	101017501
Start date of the project	Reliance
Duration of the project	24 months
Type of Action	Research and Innovation Action
Coordinator	PSNC

Due date of delivery	30/06/2022
Actual date of delivery	10/07/2022
Work package	WP4
Type of deliverable	Report
Dissemination level	Public
Responsible	MEE0
Reviewer	PSNC
Version	1.0



This project has received funding from the European research infrastructures (including e-Infrastructures) under the European Union's Horizon 2020 research and innovation programme under grant agreement No 101017501

## List of authors, contributors and reviewers

Name	Role	Organization
Simone Mantovani	Author	MEE0
Sergio Ferraresi	Contributor	MEE0
Fabio Govoni	Contributor	MEE0
Mario Cavicchi	Contributor	MEE0
Raul Palma	Reviewer	PSNC

## History of changes

Version	Date	Change	Authors	Organization
0.1	08/07/2021	Final version (Release I)	Simone Mantovani	MEE0
0.2	06/07/2022	Version ready for revision	Simone Mantovani	MEE0
0.3	07/07/2022	Formatted and reviewed version	Raul Palma	PSNC
1.0	10/07/2022	Updated links and final version	Raul Palma	PSNC

---

## Glossary

Acronym	Explanation
AAI	Authentication and Authorization Infrastructure
ADAM	Advanced geospatial Data Management
API	Application Program Interface
CAMS	Copernicus Atmosphere Monitoring Service
DAS	Data Access System
EO	Earth Observation
EOSC	European Open Science Cloud
FAIR	Findable, Accessible, Interoperable, Reproducible
MEEO	Meteorological and Environmental Earth Observation
OGC	Open Geospatial Consortium
OIDC	OpenID Connect
RELIANCE	Research Lifecycle Management technologies for Earth Science Communities and Copernicus users in EOSC
RO	Research Object
SSO	Single Sign-On
WCS	Web Coverage Service

## Table of Contents

List of authors, contributors and reviewers.....	2
History of changes.....	3
Glossary.....	4
Table of Contents .....	5
<b>1 Executive Summary .....</b>	<b>7</b>
<b>2 Introduction .....</b>	<b>7</b>
2.1 Data Cubes.....	7
2.2 ADAM platform.....	8
2.2.1 Principles and architecture .....	8
1.1.1 Authentication and Users Management.....	9
1.1.2 ADAM interfaces .....	10
<b>3 Explorer .....</b>	<b>10</b>
3.1 Geometry Tab .....	11
3.1.1 Geometry Selection .....	12
3.1.2 Places .....	13
3.2 Datasets Tab: the data discovery .....	14
3.2.1 Data Discovery .....	14
3.2.2 Data analysis: subset extraction .....	15
3.2.3 Data analysis: pixel-based time series and charts .....	17
3.3 Globe Toolbox.....	18
3.4 User Menu.....	18
<b>4 adamapi.....</b>	<b>20</b>
4.1 Auth.....	20
4.2 Datasets.....	20
4.3 Search .....	20
4.4 GetData.....	20
<b>5 Open Geospatial Consortium (OGC) standard services.....</b>	<b>21</b>
5.1 Data discovery service - OpenSearch.....	21
5.1.1 Discovery Level 1: datasets discovery.....	21
5.1.2 Discovery Level 2: discovery of products.....	23
5.2 Data access service - WCS.....	25
5.2.1 GetCapabilities .....	25
5.2.2 DescribeCoverage .....	26
5.2.3 GetCoverage .....	27

## List of Figures

Figure 2-1 ADAM Data Access System (DAS) module .....	8
Figure 2-2 ADAM high-level architecture .....	9
Figure 2-3 ADAM Sign-in interface .....	10
Figure 3-1. Explorer: interactive areas.....	11
Figure 3-2. Explorer: Geometry Tab Menu. ....	11
Figure 3-3. Explorer: polygon selection. ....	12
Figure 3-4. Explorer: polygons management.....	13
Figure 3-5. Explorer: point locations management. ....	13
Figure 3-6. Explorer: the Datasets Tab. ....	14
Figure 3-7. Explorer: adding new datasets. ....	15
Figure 3-8. Explorer: visualizing data on the globe.....	15
Figure 3-9. Explorer: Subset extraction. ....	16
Figure 3-10. Explorer: extracting a Subset Time Series. ....	17
Figure 3-11. Explore: pixel-based time series.....	17
Figure 3-12. Explorer: Time series Settings. ....	18
Figure 3-13. Explorer: Globe Toolbox. ....	18
Figure 3-14. Explorer: User Menu.....	19
Figure 3-15. Explorer: Help Section. ....	19
Figure 3-16. Explorer: User Profile Section.....	20
Figure 5-1 Result example of OpenSearch request to list all available datasets in ADAM .....	22
Figure 5-2 Result example of OpenSearch request to retrieve details of a datasets in ADAM .....	23
Figure 5-3 Result example of OpenSearch request to retrieve the list of all products in ADAM .....	24
Figure 5-4 Result example of OpenSearch request to retrieve all products of a datasets in ADAM .....	25
Figure 5-5 Result example of WCS request to retrieve list of datasets (collections) in ADAM .....	26
Figure 5-6 Result example of WCS request to retrieve details of a dataset (collection) in ADAM .....	27
Figure 5-7 Result example of WCS request to retrieve a product from a dataset in ADAM .....	28
Figure 5-8 Result example of WCS request to retrieve a product from a dataset in ADAM with RAINBOW colortable .....	29

## 1 Executive Summary

RELIANCE, short for Research Lifecycle Management for Earth Science Communities and Copernicus users in EOSC, aims to realize the vision of FAIR research in EOSC by adopting a holistic research management approach based on three key and complementary technologies: i) Research objects (RO) as the overarching mechanism to manage scientific research activities; ii) data cubes as the mechanisms enabling an efficient and scalable Earth Observation data discovery and access; iii) text mining and semantic enrichment services allowing to extract machine-readable metadata from RO resources, enabling researchers to discover scientific information at scale and to structure their own research.

This document is the second release of the Data Cube API which describes how to exploit the FAIR Data Cubes made available within the RELIANCE project. The Data Cube Metadata Model adopted for describing FAIR Data Cubes and the list of Data Cube resources are detailed in the D4.1 and D4.3, respectively.

## 2 Introduction

### 2.1 Data Cubes

The concept of Data Cube applied to geospatial data has progressively gathered more and more consensus as an effective technology to support the uptake of satellite EO applications. In the international scenario, Data Cubes are being progressively adopted at national level to provide a common reference and data access infrastructure for geospatial data, mostly related to the access and exploitation of large collections of multi-temporal, multi-resolution satellite imagery. Data Cube technology is rapidly evolving thanks to the parallel evolution of data storage systems (e.g., the progressive affirmation of object storage in cloud environment), databases and elastic and scalable computing capacities.

Although the definition of Data Cube is still missing, and both users' communities and standardization bodies (e.g., Open Geospatial Consortium) are working on it, Data Cubes are widely considered as a technological enabler providing efficient and cost-effective interfaces to access large volumes (and large variety) of geospatial data (EO and non-EO).

A Data Cube generally is represented as a multi-dimensional data concept (which can be 1-dimensional, 2-dimensional, 3-dimensional, or higher-dimensional), but more in general a Data Cube consists of a comprehensive set of functions:

- to describe both the data structure (e.g., its spatial resolution or temporal dimensions and granularity), and more precisely
  - Dataset: a collection of products e.g., grouped by phase mission.
  - Product the single file which is made available through the ADAM Space service

- to describe all the available services to exploit discovery, access, view and analytics services

In the framework of RELIANCE project, the ADAM platform is used as reference implementation of Data Cube: in order to integrate in the European Open Science Cloud infrastructure, the FAIR Data Cube services and guarantee a high level of interoperability and reusability of Copernicus and non-Copernicus data, the harmonization and alignment with ongoing OGC Testbed17 activities<sup>1</sup> is also pursued.

## 2.2 ADAM platform

The Advanced geospatial Data Management platform (ADAM) is a tool to access a large variety and volume of global environmental data. ADAM allows you extracting global as well as local data, from the past, current time, as well as short term forecast and long-term projections. Most of the data are updated daily to allow users having always the most recent data to play with.

In the framework of RELIANCE project activities, an ad-hoc instance of ADAM is deployed and configured to access Copernicus and non-Copernicus satellite- and model-based data.

### 2.2.1 Principles and architecture

The core of ADAM is a Data Access System (DAS), a software module that manages a large variety of geospatial information that feature different data format, geographic / geometric and time resolution. It allows accessing, visualizing, sub-setting, combining, processing, downloading all data sources simultaneously. The DAS exposes OGC Open Search and Web Coverage Service (WCS 2.x) interfaces that allow discovering available datasets and subset them in any dimension with a single query (see figure below).

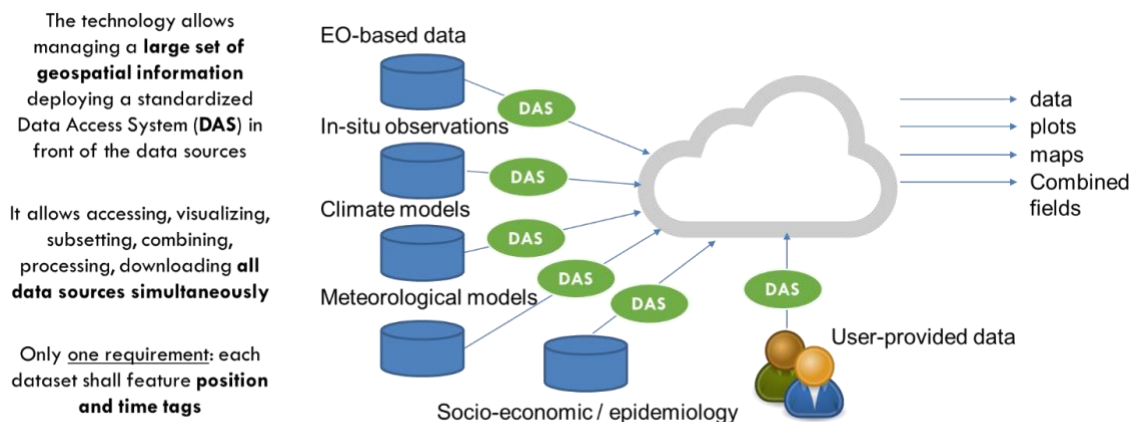


Figure 2-1 ADAM Data Access System (DAS) module

1

<https://www.ogc.org/projects/initiatives/t17>



ADAM is a very modular platform: various DAS are deployed on different data sources (DIAS Mundi, DIAS creodias, Amazon Web Services - AWS, MEEO Data Facility, SISTEMA Data facility), allowing accessing and sub-setting the available datasets without downloading / duplicating the data. Distributed data sources are made accessible through the Data Cube layer, that exposes OGC-standardised interfaces. On top of the Data Cube layer, platform-based interfaces (web application, mobile application, Jupyter Notebook and APIs) as well as third party user interfaces can be deployed.

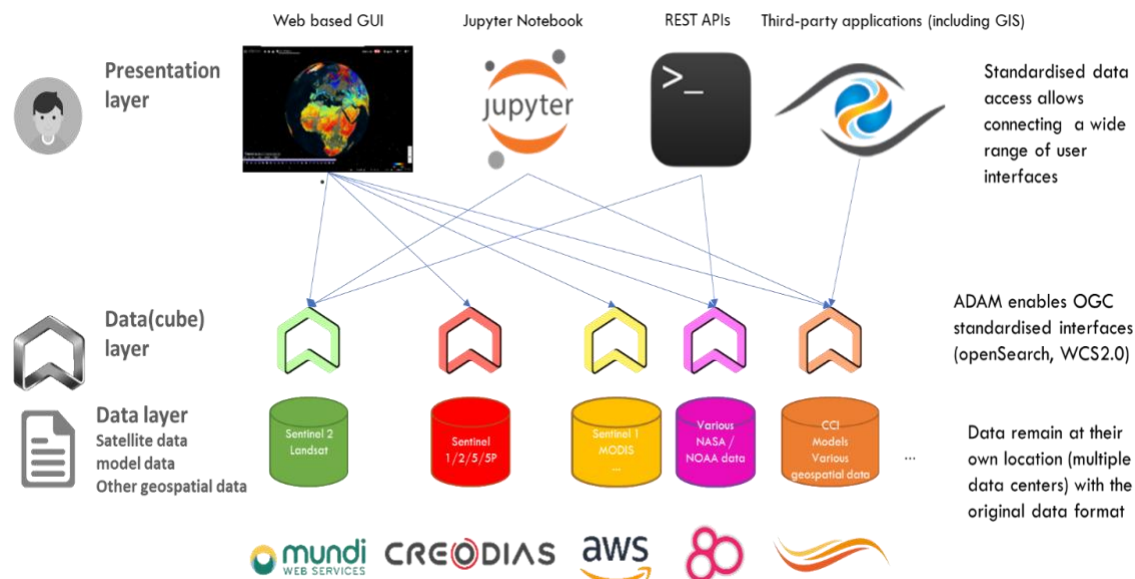


Figure 2-2 ADAM high-level architecture

ADAM, referred also as a virtual Data Cube technology, enables seamless access to all available datasets through a single interface to allow developing user-centred solutions on top of it; cutting the access barriers to a large variety of data (no need to know the exact format of any of the available datasets) it saves data preparation time and facilitates the simultaneous use of data coming from different sources.

#### 1.1.1.1 Authentication and Users Management

The Authentication, Authorization and Accounting (AAA) service is a generic module based on OAuth and OpenID Connect (OIDC) technologies, that allows the integration of customised as well as third-party authentication systems, such as NextGEOS SSO, Google, LinkedIn, and any other OAuth/OIDC provider. It can also be adapted to manage Microsoft Active Directory login. The figure below gives examples of internal (MEEO) and external OIDC providers currently managed by ADAM.

In the framework of RELIANCE project activities, the ROHub OpenID Connect endpoint has been integrated and configured to provide the RELIANCE users a unified Single Sign-On (SSO) entry point to all service components. Furthermore, since ROHub IdAM system

is connected to EOSC AAI via the EGI check-in service, it provides ADAM users SSO across all the EOSC services.

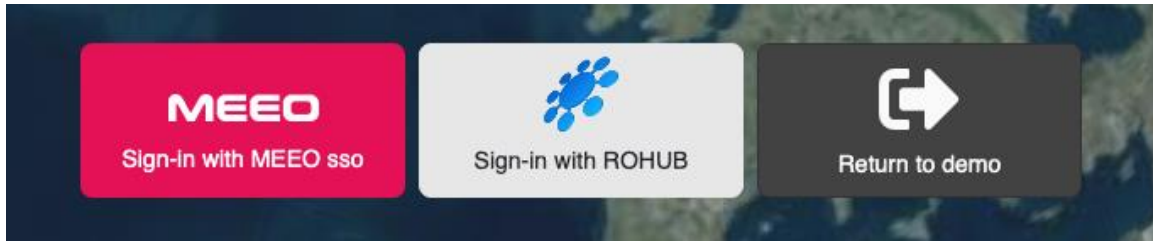


Figure 2-3 ADAM Sign-in interface

### 1.1.2 ADAM interfaces

ADAM offers the following interfaces for data access

- the **Explorer**, a web-based graphic user interface to allow users to explore, access, process and download data. Explorer includes also and Operator Interface, data processing interface and a mobile application (ADAM Mobile)
- the **Jupyter Notebook**, a web-based processing environment to allow users to import, write and execute code that runs close to the data, exploiting the power and the APIs on a remote computation environment (no user resources are used)
- The **adamapi** python-library to connect with the **Application Programming Interface (API)** ADAM data access and processing capabilities directly integrated in the user's code and applications.
- The **Open Geospatial Consortium (OGC)** standard services, that provide machine-to-machine interfaces for data discovery and data access services.

This release of the document focuses on the Explorer, Jupyter Notebook, OGC standard services and adamapi.

## 3 Explorer

The Explorer has four main interactive areas: Geometry Tab **(1)**, Datasets Tab **(2)** Globe Toolbox **(3)**, User Menu **(4)**. Each of them has specific functionalities that are described in the following sections (see Figure 3-1).

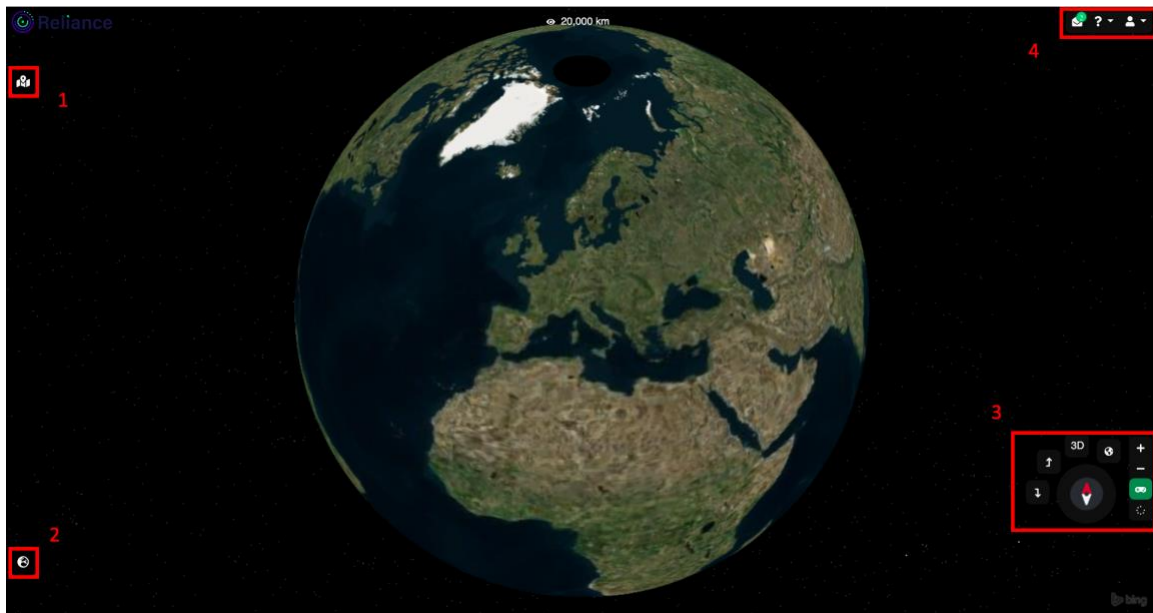


Figure 3-1. Explorer: interactive areas.

### 3.1 Geometry Tab

The Geometry Tab provides all the options that are available to select a polygon (Area of Interest) or a single location point on the globe. The user at her/his first access will also find the empty list of the personal geometries (see Figure 3-2).



Figure 3-2. Explorer: Geometry Tab Menu.

The list can be populated with the polygons selected and saved by the user, so that they will be available again for further activities.

### 3.1.1 Geometry Selection

In the Geometry Tab it is possible to select an Area of Interest with three different options: by drawing a bounding box(1), an arbitrary polygon(2), or uploading a polygon in KML or JSON format(3). Polygons defined by administrative borders can be directly obtained by using the **Search Location** feature (4) to add specific locations (see Figure 3-3).



Figure 3-3. Explorer: polygon selection.

The selected polygons are then saved in a permanent list, as long as the user decides to remove them. For each element of the lists, a series of options is available and displayed in this order: **Zoom to, Remove, Export, and Options.**(1)

Each polygon in the list can be activated and it will then appear on the globe by clicking on the related button (2). The same button can also deactivate the polygon and remove it from the globe (see Figure 3-4).



Figure 3-4. Explorer: polygons management.

If the user needs to select a single point location to analyse a specific pixel, the operations are strongly similar to the ones just explained for the polygons; the user can select manually a point location on the globe or find and add a specific location from the Geometry Tab. (1) From this step onward, the point location can be handled in the same way previously described for the polygons (2) (see Figure 3-5).

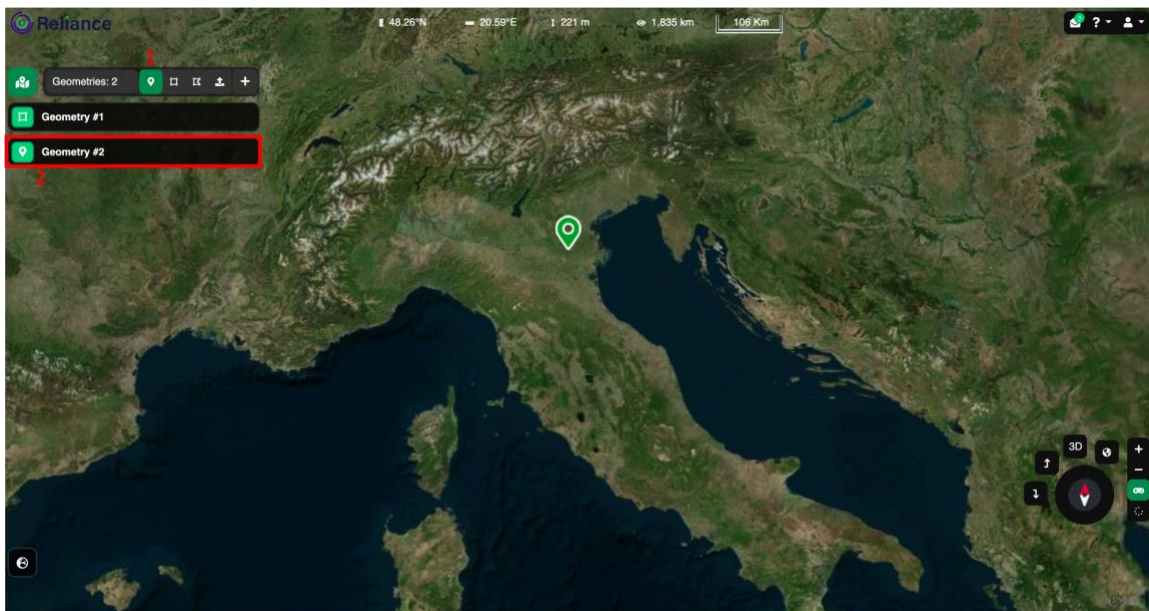


Figure 3-5. Explorer: point locations management.

### 3.1.2 Places

The **Places** tab allows to manage location-related data, which are points and polygons that are separately listed in the tab. The polygons, used to visualize or extract the data,



can be added both manually and by uploading a KML or GeoJson file. Polygons defined by administrative borders can be directly obtained by using the **Search Location** feature and then added as new polygons. (1).

The same works for point locations that can be added by clicking on the globe or searching for them, as explained about the polygons. Polygons and points are then saved in permanent lists, as long as the user will decide to remove them. For each element of the lists, a series of options is available and displayed in this order (2) (3): **Remove, Activate. Show/hide** on globe, **Zoom to, Export, and Info**.

When an element (point or polygon) is displayed on the globe, it can appear with two different colors: in red if the element is disabled, in green if it is enabled and can be used for data visualization or processing. The operation of adding a polygon or a point-location always opens a **geometry settings** pop-up as follows.

### 3.2 Datasets Tab: the data discovery

#### 3.2.1 Data Discovery

The Datasets Tab allows to handle the list of datasets that are available for each single user. At the first access, the user will find an empty list of datasets in the personal “basket”, so it will be necessary to open the “All Datasets” window to select which products to add in the basket (1) (see Figure 3-6).



Figure 3-6. Explorer: the Datasets Tab.

In the “**All Datasets**” section the user can select the needed datasets and, then it will automatically appear in the list of the Datasets Tab, ready to be used (see Figure 3-7).

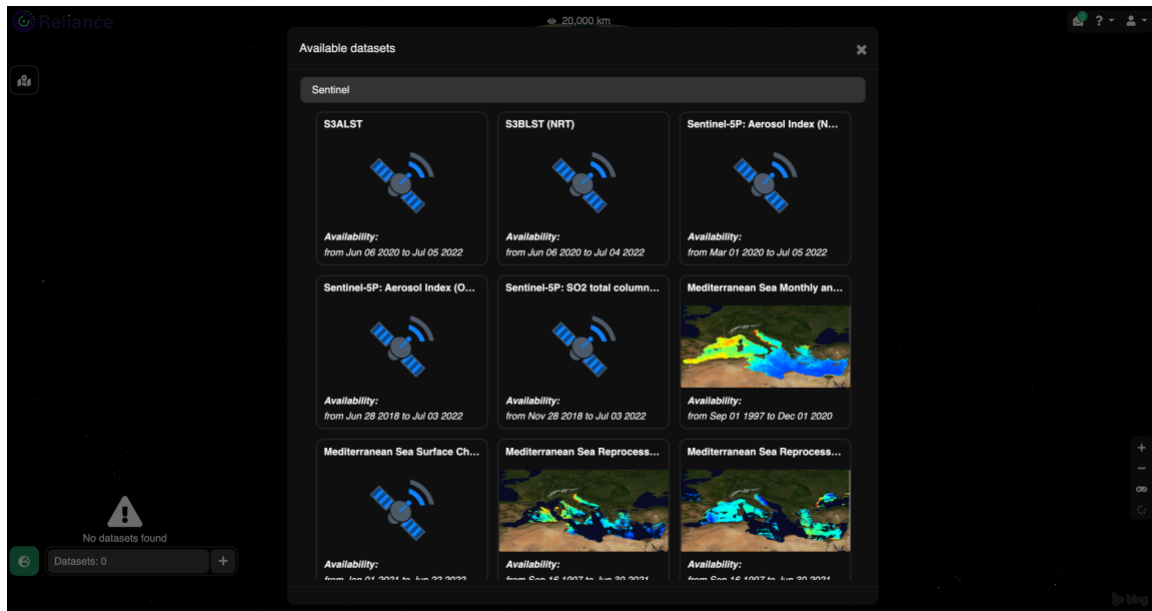


Figure 3-7. Explorer: adding new datasets.

Once the dataset is in the list, it can be activated: the product will appear on the globe, together with a timebar that allows to browse the data from one date to another, in the temporal coverage that is available for each specific dataset (1).

When enabled, each dataset has default colortable, opacity and range of values (2): these three elements can be then set by the user according to her/his needs (see Figure 3-8).

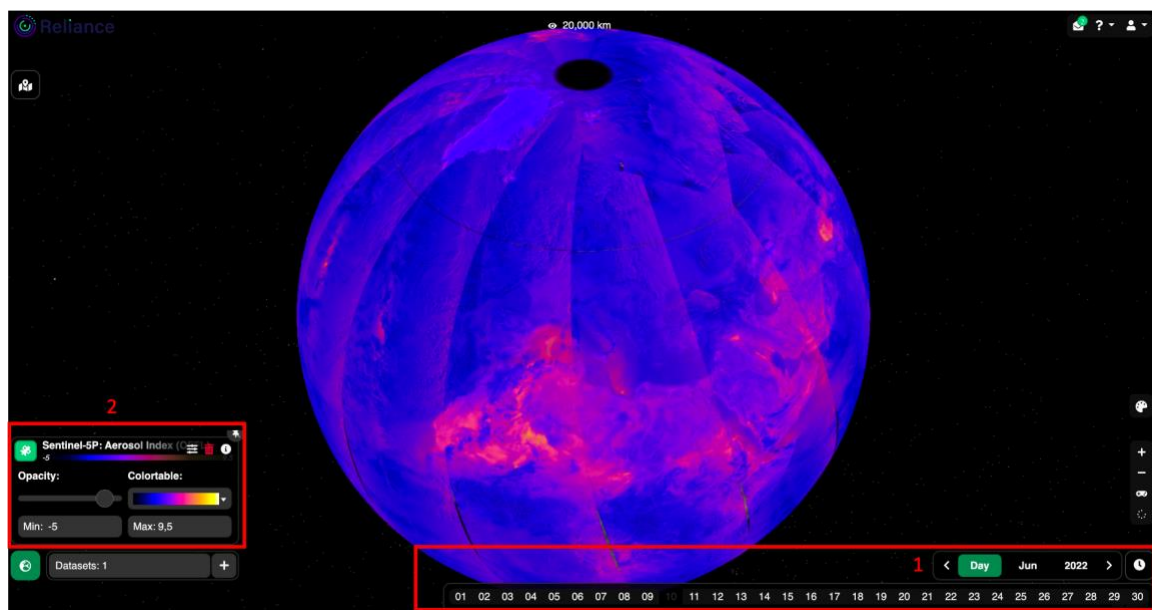


Figure 3-8. Explorer: visualizing data on the globe.

### 3.2.2 Data analysis: subset extraction

Once the dataset is enabled, besides visualizing it on a specific region of the globe, the platform mainly allows the user to extract data on a specific subset for a selected period

of time, generating a time series over an Area of Interest that can be further downloaded and processed locally. The user should first select a polygon in the personal list of geometries, or define a new geometry (1) and then enable the dataset of interest (2). A corresponding new window will appear on the right, showing a smaller version of the subset (3) (see Figure 3-9).

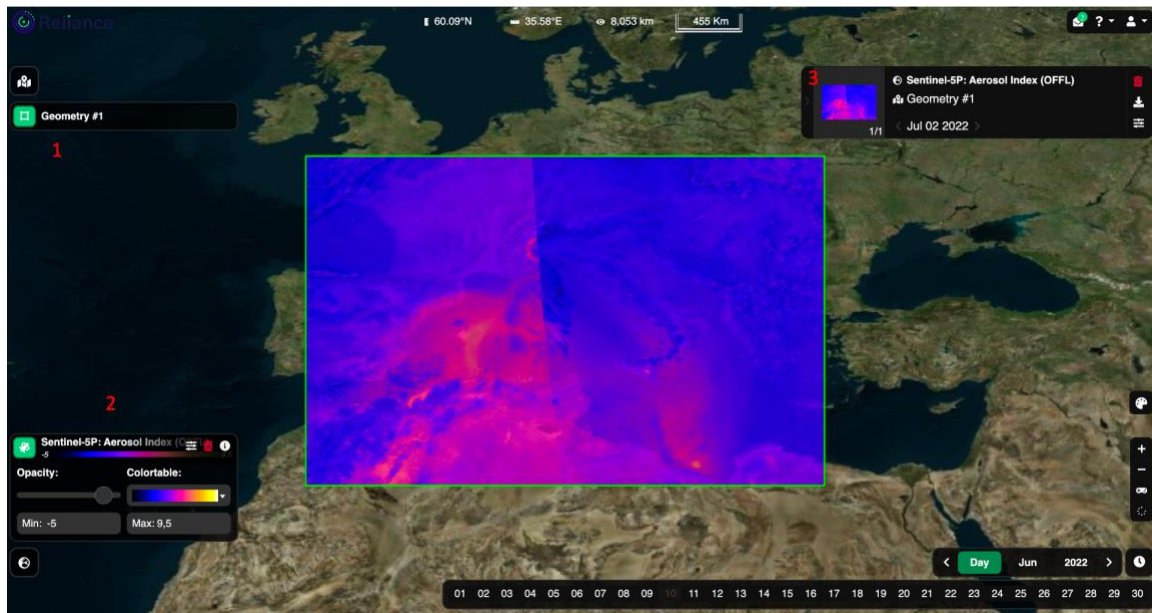


Figure 3-9. Explorer: Subset extraction.

The subset window also provides its own menu, allowing the user to delete the subset (1), download a time series of the subset (2) (with a default sequence including the latest 6 elements before the selected day) and a menu of options that depends on the dataset (3). The download action will generate a ZIP file containing all the elements of the time series and this operation will be notified in the personal message box of the user (4) (see Figure 3-10).



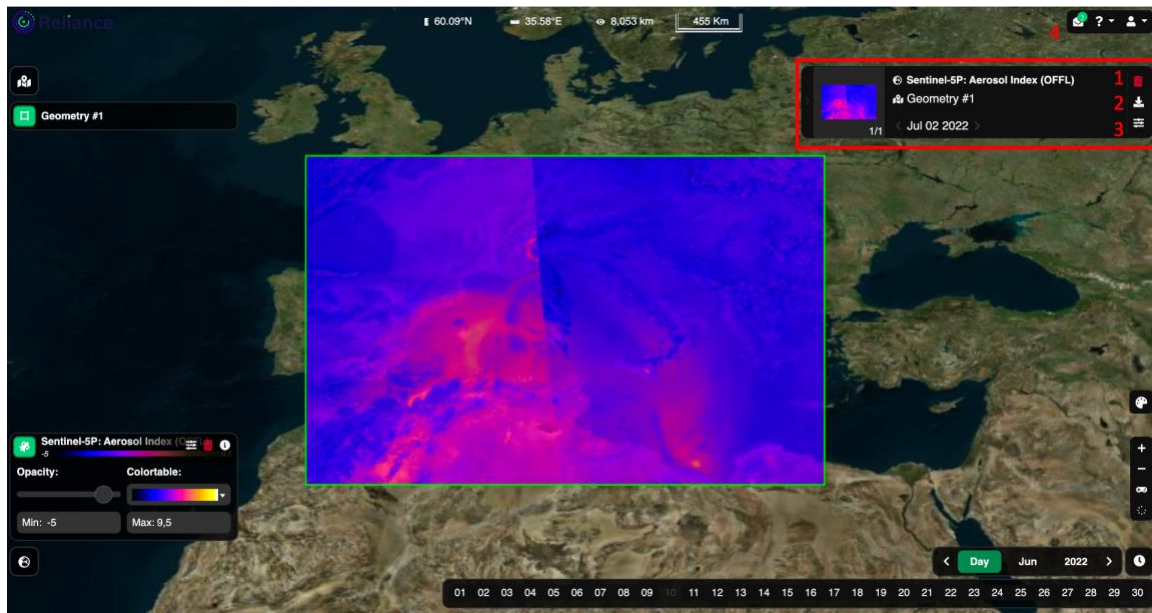


Figure 3-10. Explorer: extracting a Subset Time Series.

### 3.2.3 Data analysis: pixel-based time series and charts

If the user activates a single point location instead of a polygon, the ESA PDGS Data Cube service will generate a pixel-based time series of 10 days (if available) and a new window will appear on the right side of the interface showing a reduced view of the temporal plot. (see Figure 3-11).

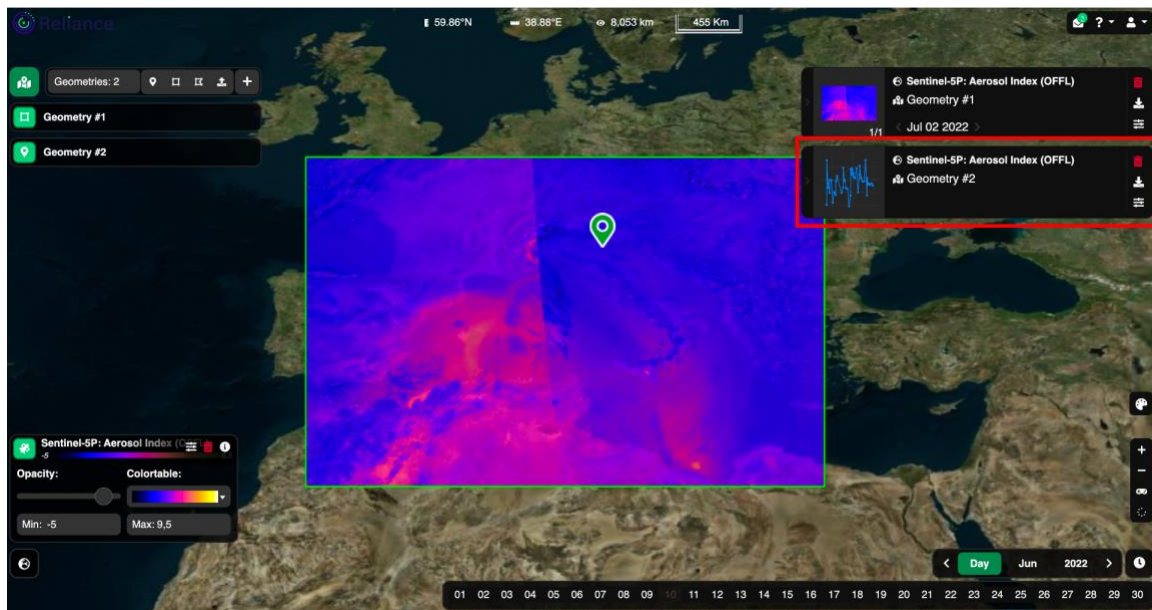


Figure 3-11. Explore: pixel-based time series.

Clicking on the plot, the user will get an enlarged view of it, that also provides an **Edit Menu** (1) where it is possible to change the time range of the plot and to download the time series as a zipped version of the **CSV** file (2) (see Figure 3-12).

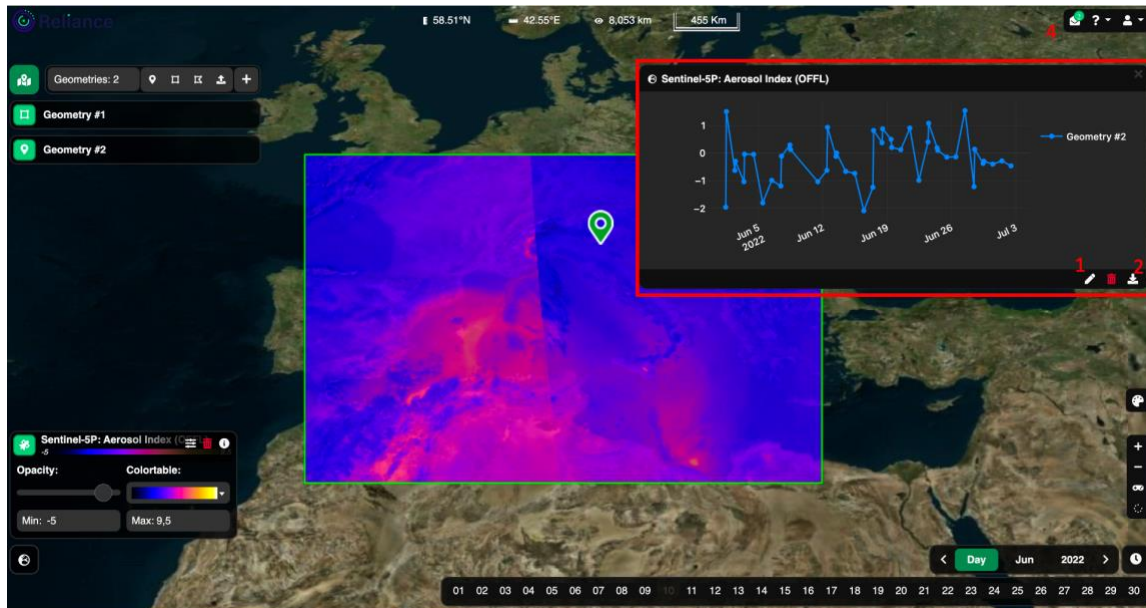


Figure 3-12. Explorer: Time series Settings.

### 3.3 Globe Toolbox

The **Globe Toolbox** provides the following features: (see Figure 3-13)

- zoom functionality (+/- buttons), and a controller sub-menu (1)
- basemap style options (2)
- 3D /2D-map projections switch (3)
- elevation exaggeration (raise/lower) (4)(5)

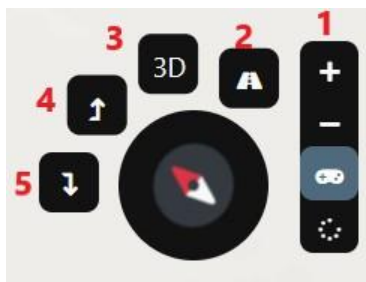


Figure 3-13. Explorer: Globe Toolbox.

The Globe Toolbox allows the user to visualize and navigate a 3D view of a subset that has been selected for a specific collection. In the following image, a Sentinel2 True Color image is displayed after moving from a 2D to a 3D view.

### 3.4 User Menu

The **User Menu** includes two three different sections (see Figure 3-14):

- Jupyter Hub Section Account Message Box (1)

- Help Section (2)
- User Profile Section (3)

The **Jupyter Hub Section Account message box** provides a Jupyter Hub interface where the user can develop, modify and test his own code using the Jupyter Notebooks with a Python programming environment. It contains all the messages about internal communications ( account verification, submitted orders, etc. ) (see Fig. 3-14).

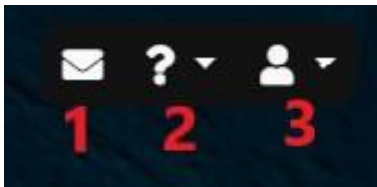


Figure 3-14. Explorer: User Menu.

The **Help Section** provides info about the platform, a link to get in contact with the platform maintainers or the reference page for documentation, tutorials and video tutorials (see Figure 3-15).

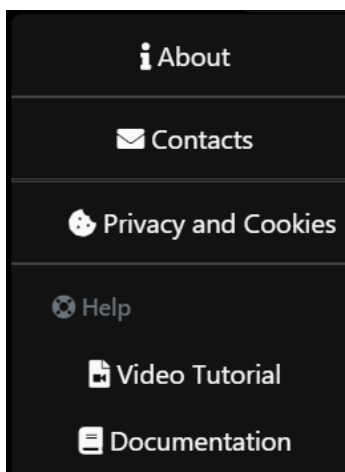


Figure 3-15. Explorer: Help Section.

The **User Profile Section** shows three options to each user: Besides making the User Profile (1), from this section is possible to enter the internal message box, where the user could receive information about his submitted product orders or official communications about the platform. Providing the personal API key that the user could need for an authentication procedure, the User Profile section also allows to set a functional parameter of the service, the **Temporal Bar Days Range**, that is the default number of elements that are extracted in a time series (see Figure 3-16).

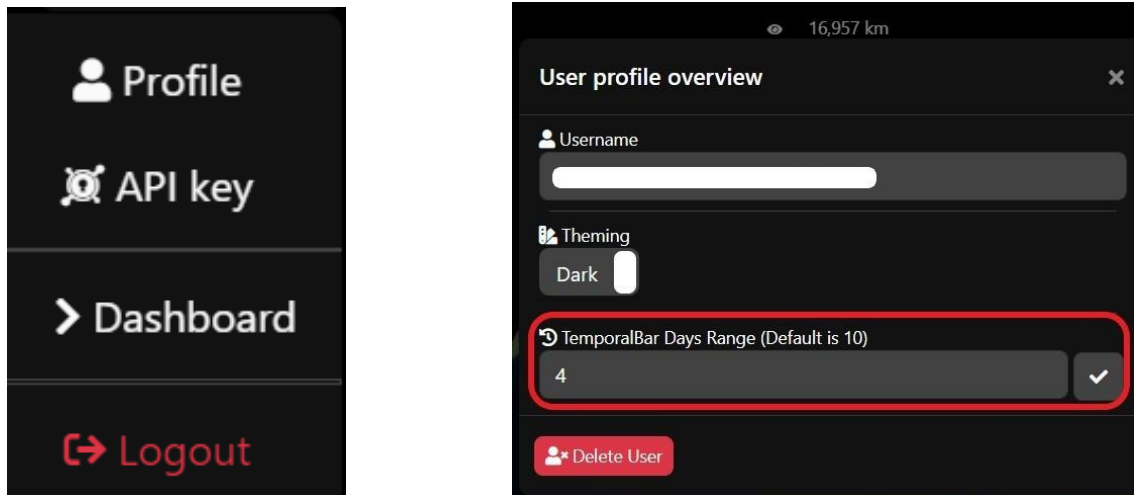


Figure 3-16. Explorer: User Profile Section.

## 4 adamapi

*adamapi* is a python package distributed through the official Python Package Index (PYPI), that allows the users to interact with all the available Data Cubes exposed in ADAM.

This package is divided in 4 modules: Auth, Datasets, Search and GetData.

All the detailed documentation for this and for the future versions will be available here: <https://pypi.org/project/adamapi/>

### 4.1 Auth

Auth module takes care of user authentication and authorization.

Without instanting an object of this module other components don't work.

Auth module is based on the ADAMAPI\_KEY, a key that uniquely identifies the user, available through ADAM portal user area.

### 4.2 Datasets

Datasets module provides the possibility to discover Data Cubes and returns a list of them or, if you request a specific Data Cube returns all the metadata associated to it.

### 4.3 Search

Search module has the same behaviour of the Datasets module but work on the available products of a Data Cube with the possibilities to execute temporal e spatial filtering.

### 4.4 GetData

GetData module is the last in the chain because its behaviour is that to download the requested products in different formats or to execute sub-setting or processing and download the results.

## 5 Open Geospatial Consortium (OGC) standard services

ADAM exposes standardised Open Geospatial Consortium (OGC) services for data discovery, data visualization, access and data processing, to facilitate the integration and exploitation of dataset and product resources within user's stacks.

According to the metadata model [D4.1], a Data Cube provides a series of **functions** to discover, access, visualise and process the **datasets**, namely the aggregation of **products** with common properties into a multi-dimensional digital object with its spatial and temporal dimensions.

The complete list of FAIR Data Cubes can be found in D4.3: as example of the Data Cube resources requested by RELIANCE user communities, the **CAMS European air quality forecasts Data Cube** will provide access to hourly products organized in 13 datasets (i.e., different environmental variables, such as ammonia, carbon\_monoxide, ...) over Europe from July 2018 to 2021.

An OpenSearch catalogue service (**OpenSearch**) is exposed to discover all data cubes within the platform through the various data facilities. The connection to INSPIRE compliant catalogues is also supported.

The Web Coverage Service (**WCS**) is the core part of the DAS module. WCS can be queried directly via REST calls.

### 5.1 Data discovery service - OpenSearch

The discovery service exposes the **OpenSearch Geo and Time Extensions** interface to explore the Data Cube resource in each of its own dimensions (space, time, ...). The metadata model adopted by the OpenSearch service is describe in [D4.1]: the description of the service API, including examples are described in the next sections of the document. The pagination mechanism is used to handle the navigation through results.

The service description, including supported operations, can be obtained through the describe operation

- <https://reliance-das.adamplatform.eu/opensearch/describe.xml>

The OpenSearch supports a 2 steps discovery service.

#### 5.1.1 Discovery Level 1: datasets discovery

The generic syntax for the “datasets” operation is:

`http://mwcs-test.meeo.datacenter/opensearch/datasets?datasetId={datasetId}&startDate={date:`



YYYY-MM-DD | YYYY-MM-DDTHH:MM:SSZ}&endDate={date: YYYY-MM-DD | YYYY-MM-DDTHH:MM:SSZ}&startIndex={num}&maxRecords={num}

where:

- *datasetId* is the identifier of the Data Cube
- *startDate* / *endDate* refer to the time range of the products indexed in the Data Cube
- *startIndex* is the parameter used by the pagination mechanism to flow through the Data Cube resources
- *maxRecords* is the parameter to set the number of resources to be listed in a page

The result of this operation provides the list available datasets, each with the metadata description as defined in [D4.1]. Using the pagination options, it is possible to navigate through the complete list of datasets.

### Sample requests

- To get the list of all available datasets: [/datasets](#)

The operation returns the total number of datasets available (i.e., *totalResults*) as well as the paginated list of datasets (i.e., *datasetId*) with the metadata description.

JSON	Dati non elaborati	Header
Salva	Copia	Comprimi tutto
Espandi tutto (lento) <input type="text" value="Filtro JSON"/>		
type:	"FeatureCollection"	
properties:	<div> <div>id:</div> <div>"https://reliance-das.adamplatform.eu/opensearch/datasets?"</div> </div> <div> <div>query:</div> <div> <div>request:</div> <div>[...]</div> </div> <div> <div>itemsPerPage:</div> <div>"10"</div> </div> <div> <div>startIndex:</div> <div>"0"</div> </div> <div> <div>totalResults:</div> <div>56</div> </div> </div> <div> <div>links:</div> <div> <div>0:</div> <div> <div>first:</div> <div>"https://reliance-das.adamplatform.eu/opensearch/datasets?startIndex=0"</div> </div> <div>1:</div> <div>[...]</div> </div> <div>2:</div> <div>[...]</div> </div> <div> <div>3:</div> <div>description:</div> <div>"https://reliance-das.adamplatform.eu/opensearch/describe"</div> </div>	

Figure 5-1 Result example of OpenSearch request to list all available datasets in ADAM

- To get the dataset details: [/datasets?datasetId=MODh20chl\\_4km](#)

The operation returns the metadata description of the selected dataset (i.e., *MODh20chl\_4km*)

JSON	Dati non elaborati	Header
Salva	Copia	Comprimi tutto
Espandi tutto		
Filtro JSON		
type:	"FeatureCollection"	
properties:		
id:	"https://reliance-das.adamplatform.eu/opensearch/datasets?datasetId=M00h20chl_4km&startIndex=0"	
query:		
request:		
0:		
datasetId:	"M00h20chl_4km"	
itemsPerPage:	"1"	
startIndex:	"0"	
totalResults:	1	
links:		
0:		
first:	"https://reliance-das.adamplatform.eu/opensearch/datasets?datasetId=M00h20chl_4km&maxRecords=10&startIndex=0"	
1:	[...]	
2:		
description:	"https://reliance-das.adamplatform.eu/opensearch/describe"	
features:		
0:		
datasetId:	"M00h20chl_4km"	
creationDate:	"2008-01-01T00:40:01Z"	
dataType:	"Float32"	
epsg:	"4326"	
keywords:	[...]	
license:	[...]	
maxDate:	"2020-12-01T00:00:00Z"	
maxValue:		
0:	20	
minDate:	"2008-01-01T00:40:01Z"	
minValue:		
0:	0.005911331730664	

Figure 5-2 Result example of OpenSearch request to retrieve details of a datasets in ADAM

### 5.1.2 Discovery Level 2: discovery of products

The generic query syntax for the search operation is

http://<service endpoint>/opensearch/search?datasetId={datasetId}&startDate={date: YYYY-MM-DD | YYYY-MM-DDTHH:MM:SSZ}&endDate={date: YYYY-MM-DD | YYYY-MM-DDTHH:MM:SSZ}&geometry={geojson}&startIndex={num}&maxRecords={num}

where:

- *datasetId* is the identifier of the Data Cube
- *startDate* / *endDate* refer to the time range of the products indexed in the Data Cube
- *startIndex* is the parameter used by the pagination mechanism to flow through the Data Cube resources
- *maxRecords* is the parameter to set the number of resources to be listed in a page

The result of this operation provides the list available products, each with the metadata description as defined in [D4.1]. Using the pagination options, it is possible to navigate through the complete list of products.

### Sample requests

- To get the list of all products: </search>

The operation returns the total number of products available (i.e., *totalResults*) as well as the paginated list of each product (i.e., *productId*) with metadata description.

JSON	Dati non elaborati	Header
Salva	Copia	Comprimi tutto
Espandi tutto		
Filtro JSON		
<pre> type:   "FeatureCollection" properties:   id:     "https://reliance-das.adamplatform.eu/opensearch/search?"   query:     request:       0:         {}       itemsPerPage:         "10"       Index:         "0"       totalResults:         719430     links:       0:         first:           "https://reliance-das.adamplatform.eu/opensearch/search?maxRecords=10&amp;startIndex=0"         1:           {...}         2:           last:             "https://reliance-das.adamplatform.eu/opensearch/search?maxRecords=10&amp;startIndex=719429"         3:           description:             "https://reliance-das.adamplatform.eu/opensearch/describe"   features:     0:       _id:         \$id:           "60c33cd76dfbc006c332e5"         datasetId:           "SRTH_XSAR"         productDate:           "2001-01-01T00:00:00Z"         productId:           "SRTH_XSAR_20010101_E010N20_XSAR_DEM.tif"         geometry:           {...}         insertDate:           "2021-06-11T10:37:11Z"         productPath:           "/SRTH/2001/01/01/SRTH_XS_01_E010N20_XSAR_DEM.tif"         source:           "nfs://192.168.1.156:/nfs_01_vol_02/reliance"         status:           "Online"       1: </pre>		

Figure 5-3 Result example of OpenSearch request to retrieve the list of all products in ADAM

- To get the list of all available products for a specific dataset: [/search?datasetId=MODh20chl\\_4km](/search?datasetId=MODh20chl_4km)

The operation returns the total number of products available (i.e., *totalResults*) per the selected dataset (i.e., *MODh20chl\_4km*) as well as the paginated list of each product (i.e., *productId*) with metadata description.



JSON	Dati non elaborati	Header
Salva	Copia	Comprimi tutto
Espandi tutto		
Filtro JSON		
type:	"FeatureCollection"	
properties:		
id:	"https://reliance-das.adamplatform.eu/opensearch/search?datasetId=MODh20chl_4km&startIndex=0"	
query:		
request:		
0:		
datasetId:	"MODh20chl_4km"	
itemsPerPage:	"10"	
Index:	"0"	
totalResults:	13	
links:		
0:		
first:	"https://reliance-das.adamplatform.eu/opensearch/search?datasetId=MODh20chl_4km&maxRecords=10&startIndex=0"	
1:		
next:	"https://reliance-das.adamplatform.eu/opensearch/search?datasetId=MODh20chl_4km&maxRecords=10&startIndex=10"	
2:		
last:	"https://reliance-das.adamplatform.eu/opensearch/search?datasetId=MODh20chl_4km&maxRecords=10&startIndex=12"	
3:		
description:	"https://reliance-das.adamplatform.eu/opensearch/describe"	
features:		
0:		
_id:		
id:	"68a27b85c4f40e61f16c68"	
datasetId:	"MODh20chl_4km"	
productDate:	"2008-01-01T00:40:01Z"	
productId:	"NETCDF:\mnt\NetApp\nan_hlor_a_4km.nc\":"chlor_a"	
band:	1	
bandDescription:	"mass_concentration_chlorophyll_concentration_in_sea_water"	
geometry:		
name:		

Figure 5-4 Result example of OpenSearch request to retrieve all products of a datasets in ADAM

- To get the list of available products for a specific dataset over an area of interest and time of interest: [/search?datasetId=MODh20chl\\_4km&startDate=2000-03-05&endDate=2000-03-15&geometry={\"type\":\"Polygon\",\"coordinates\":\[\[\[-60,-40\],\[60,-40\],\[60,40\],\[-60,40\],\[-60,-40\]\]\]}](https://reliance-das.adamplatform.eu/opensearch/search?datasetId=MODh20chl_4km&startDate=2000-03-05&endDate=2000-03-15&geometry={\)

The operation returns the total number of products available (i.e., *totalResults*) per the selected dataset (i.e., *MODh20chl\_4km*) over the time of interest and area of interest, as well as the paginated list of each product (i.e., *productId*) with metadata description.

## 5.2 Data access service - WCS

The data access service exposes the **Web Coverage Service (WCS)** interface to access the Data Cube resources. The description of the service API, including examples are described in the next sections of the document.

The WCS supports the following operations:

### 5.2.1 GetCapabilities

This operation allows at retrieving the list of available collections

<https://reliance-das.adamplatform.eu/wcs?service=WCS&Request=GetCapabilities>

This operation is equivalent to the */datasets* request, with a limited metadata schema with respect to OpenSearch service.

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="2.0" encoding="UTF-8"?>
<wms:Capabilities xmlns:wms="http://www.opengis.net/wms/2.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:crs="http://www.opengis.net/wms/service-extension/crs/1.0" xmlns:ows="http://www.opengis.net/ows/2.0" xmlns:gml="http://www.opengis.net/gml/3.2"
  xmlns:xlink="http://www.w3.org/1999/xlink" xsi:schemaLocation="http://www.opengis.net/wms/2.0 http://schemas.opengis.net/wms/2.0/wmsAll.xsd"
  version="2.0.1">
  <wms:ServiceIdentification xmlns="http://www.opengis.net/ows/2.0">
    <ows:Title>MEOO WCS Server - MWCS (Ver. 2.1-29)</ows:Title>
    <ows:Abstract>The WCS Server implementation to access datacubes and images on filesystem</ows:Abstract>
    <ows:ServiceType>OGC WCS</ows:ServiceType>
    <ows:ServiceTypeVersion>2.0.1</ows:ServiceTypeVersion>
    <ows:Profile>http://www.opengis.net/spec/WCS_coverage-encoding_jpeg2000/1.0/</ows:Profile>
    <ows:Profile>http://www.opengis.net/spec/WCS_protocol-binding_get-rest/1.0/conf/get-rest</ows:Profile>
    <ows:Profile>http://www.opengis.net/spec/GMLCOV/1.0/conf/gml-coverage</ows:Profile>
    <ows:Profile>http://www.opengis.net/spec/WCS_service-extension_processing/2.0/conf/processing</ows:Profile>
    <ows:Profile>http://www.opengis.net/spec/WCS_protocol-binding_post-xml/1.0/</ows:Profile>
    <ows:Profile>http://www.opengis.net/spec/WCS_protocol-binding_soap/1.0/</ows:Profile>
    <ows:Profile>http://www.opengis.net/spec/WCS_coverage-encoding_netcdf/1.0/</ows:Profile>
    <ows:Profile>http://www.opengis.net/spec/GMLJP2/2.0/</ows:Profile>
    <ows:Profile>http://www.opengis.net/spec/WCS_protocol-binding_get-kvp/1.0/conf/get-kvp</ows:Profile>
    <ows:Profile>http://www.opengis.net/spec/WCS_service-extension_range-subsetting/1.0/conf/</ows:Profile>
    <ows:Profile>http://www.opengis.net/spec/WCS_service-extension_scaling/1.0/conf/scaling</ows:Profile>
    <ows:Profile>http://www.opengis.net/spec/WCS_service-extension_interpolation/1.0/conf/interpolation</ows:Profile>
    <ows:Profile>http://www.opengis.net/spec/WCS_coverage-encoding_geotiff/1.0/</ows:Profile>
  </ows:ServiceIdentification>
  <wms:ServiceProvider xmlns="http://www.opengis.net/ows/2.0">
    <ows:ProviderName>MEOO S.r.l.</ows:ProviderName>
    <ows:ProviderSite xlink:href="http://www.meeo.it/"></ows:ProviderSite>
    <wms:ServiceContact>
      <ows:IndividualName>MEOO Help Desk</ows:IndividualName>
      <ows:ContactInfo>
        <ows:Address>
          <ows:City>Ferrara</ows:City>
          <ows:PostalCode>I-44123</ows:PostalCode>
          <ows:Country>Italy</ows:Country>
          <ows:ElectronicMailAddress>helpdesk@meeo.it</ows:ElectronicMailAddress>
        </ows:Address>
      </ows:ContactInfo>
      <ows:Role>Service Provider</ows:Role>
    </wms:ServiceContact>
  </wms:ServiceProvider>
</wms:Capabilities>
```

Figure 5-5 Result example of WCS request to retrieve list of datasets (collections) in ADAM

### 5.2.2 DescribeCoverage

This operation returns the WCS metadata schema for a specific collection

<https://test.adamplatform.eu/wcs?service=WCS&Request=DescribeCoverage&version=2.0.0&coverageID=<CoverageID>>

This operation is equivalent to the `/datasets&datasetId={}` request, with a limited metadata schema with respect to OpenSearch service.

The term CoverageId correspond to datasetId.

### Sample request

[https://reliance-das.adamplatform.eu/wcs?service=WCS&Request=DescribeCoverage&version=2.0.0&coverageID=MODh20chl\\_4km](https://reliance-das.adamplatform.eu/wcs?service=WCS&Request=DescribeCoverage&version=2.0.0&coverageID=MODh20chl_4km)

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="1.0" encoding="UTF-8"?>
<wcs:CoverageDescriptions xmlns:wcs="http://www.opengis.net/wcs/2.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:crs="http://www.opengis.net/wcs/service-extension/1.0" xmlns:ows="http://www.opengis.net/ows/2.0" xmlns:gml="http://www.opengis.net/gml/3.2"
  xmlns:xlink="http://www.w3.org/1999/xlink" xsi:schemaLocation="http://www.opengis.net/wcs/2.0 http://schemas.opengis.net/wcs/2.0/wcsAll.xsd">
  <wcs:CoverageDescription xmlns="http://www.opengis.net/gml/3.2" xmlns:gmlcov="http://www.opengis.net/gmlcov/1.0"
    xmlns:swe="http://www.opengis.net/swe/2.0" gml:id="MODh20chl_4km">
    <boundedBy>
      <Envelope srsName="http://www.opengis.net/def/crs-compound?
        l=http://www.opengis.net/def/crs/EPSSG/0/4326&2=http://www.opengis.net/def/crs/OGC/0/Temporal?epoch="1970-01-01T00:00:00"&uom="s";"
        axisLabels="Lat Long t" uomLabels="degree degree s" srsDimension="3">
          <lowerCorner>0.000000 0.000000 1199148001</lowerCorner>
          <upperCorner>0.000000 0.000000 1606780800</upperCorner>
        </Envelope>
      </boundedBy>
    <domainSet>
      <gml:grid:ReferenceableGridByVectors xmlns:gmlrgrid="http://www.opengis.net/gml/3.3/rgrid" dimension="3" gml:id="MODh20chl_4km-grid"
        xsi:schemaLocation="http://www.opengis.net/gml/3.3/rgrid http://schemas.opengis.net/gml/3.3/referenceableGrid.xsd">
        <limits>
          <GridEnvelope>
            <low>0 0 0</low>
            <high>2147483647 2147483647 14</high>
          </GridEnvelope>
        </limits>
        <axisLabels>Long Lat t</axisLabels>
      </gmlrgrid:origin>
      <Point gml:id="MODh20chl_4km-origin" srsName="http://www.opengis.net/def/crs-compound?
        l=http://www.opengis.net/def/crs/EPSSG/0/4326&2=http://www.opengis.net/def/crs/OGC/0/Temporal?epoch="1970-01-01T00:00:00"&uom="s";"
        axisLabels="Lat Long t" uomLabels="degree degree s" srsDimension="3">
          <pos>0.000000 0.000000 1199148001</pos>
        </Point>
      </gmlrgrid:origin>
      <gmlrgrid:generalGridAxis>
        <gmlrgrid:offsetVector srsName="http://www.opengis.net/def/crs-compound?
          l=http://www.opengis.net/def/crs/EPSSG/0/4326&2=http://www.opengis.net/def/crs/OGC/0/Temporal?epoch="1970-01-01T00:00:00"&uom="s";"
          axisLabels="Lat Long t" uomLabels="degree degree s" srsDimension="3">0 0.000000 0</gmlrgrid:offsetVector>
        </gmlrgrid:offsetVector>
        <gmlrgrid:gridAxesSpanned>Long</gmlrgrid:gridAxesSpanned>
        <gmlrgrid:sequenceRule axisOrder="+1">Linear</gmlrgrid:sequenceRule>
      </gmlrgrid:generalGridAxis>
    </domainSet>
  </wcs:CoverageDescription>
</wcs:CoverageDescriptions>
```

Figure 5-6 Result example of WCS request to retrieve details of a dataset (collection) in ADAM

### 5.2.3 GetCoverage

This operation allows at accessing the products from a dataset. The generic query syntax for the search operation is

[https://test.adamplatform.eu/wcs?service=WCS&Request=GetCoverage&version=2.0.0&coverageID=<CoverageID>&subset=\[Long\]\[N\]\(min,max\)&subset=\[Lat\]\[E\]\(min,max\)&subset=t\(minDate,maxDate\)&format=<#####>&colortable=<ID>&scale=<nn>&size=<output raster size>&token=<user\\_token>](https://test.adamplatform.eu/wcs?service=WCS&Request=GetCoverage&version=2.0.0&coverageID=<CoverageID>&subset=[Long][N](min,max)&subset=[Lat][E](min,max)&subset=t(minDate,maxDate)&format=<#####>&colortable=<ID>&scale=<nn>&size=<output raster size>&token=<user_token>)

where

- Spatial sub-settings values depend on the dataset (see DescribeCoverage output axisLabels).
  - If <axisLabels>Long Lat t</axisLabels> then latitude and longitude values in degrees must be provided
  - If <axisLabels>N E t</axisLabels> then North and East values in meters shall be provided
  - If a 3D collection is provided, subset=h() subsets the vertical dimension
- Time sub-setting, dates / times can be provided in ansi or unix time (see examples below)
- format=<##>, where ## = image/tiff, image/png, image/jp2, image/envi, image/gif, text/plain, text/csv, application/xml, application/gml+xml, application/tar, application/json

- colortable=<ID>, ID = [https://www.ncl.ucar.edu/Document/Graphics/color\\_table\\_gallery.shtml](https://www.ncl.ucar.edu/Document/Graphics/color_table_gallery.shtml)
- colorrange=<min,max>, to apply the colortable between the min/max values
- scale=<nn>, nn is resampling factor (1 native resolution, 0.1 resolution degraded to 10%)
- size=<x,y>, to define the output raster size
- token=<user\_token> to specify the user token with permissions to retrieve the data

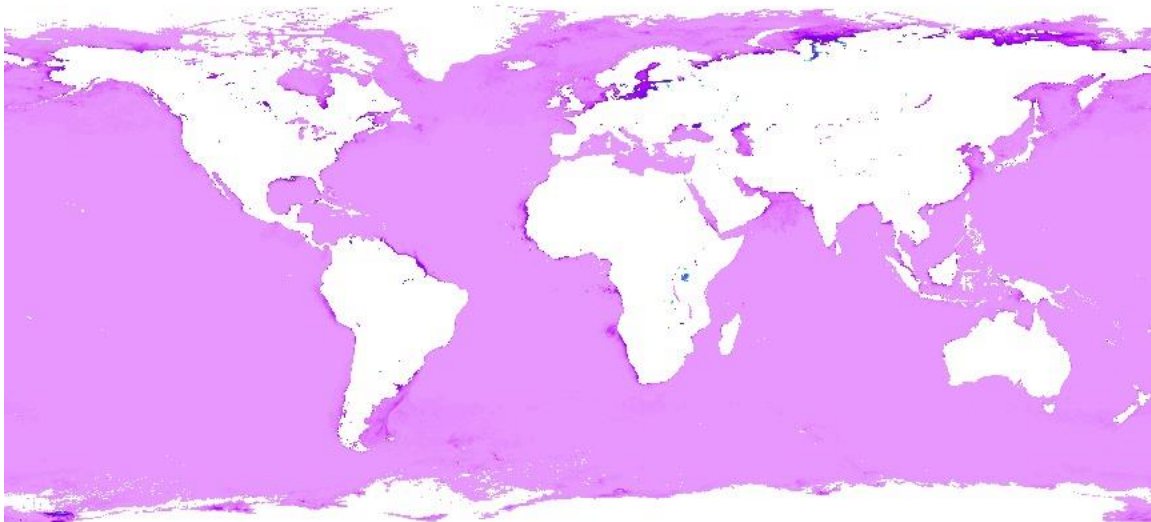
**Note:** the user token can be obtained via adamapi using the authorize methods, e.g.,

```
adam_key = open(os.path.join(os.environ['HOME'], "adam-key")).read().rstrip()
a = Auth()
a.setKey(adam_key)
a.setAdamCore('https://reliance.adamplatform.eu/')
a.authorize()
a.getAuthToken()
```

### Sample requests

MODIS Chlorophyll global on January 1<sup>st</sup> 2008 in PNG format

[https://reliance-das.adamplatform.eu/wcs?service=WCS&Request=GetCoverage&version=2.0.0&coverageID=MODh20chlYR\\_4km&subset=unix\(2008-01-01T00:40:01Z\)&format=image/png&token=<user\\_token>&scale=0.1](https://reliance-das.adamplatform.eu/wcs?service=WCS&Request=GetCoverage&version=2.0.0&coverageID=MODh20chlYR_4km&subset=unix(2008-01-01T00:40:01Z)&format=image/png&token=<user_token>&scale=0.1)



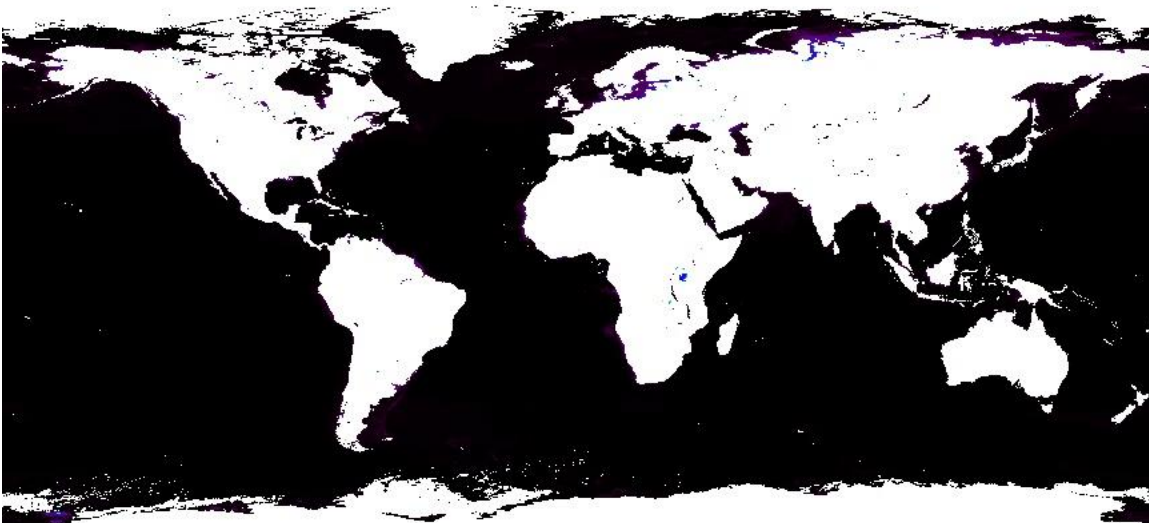
*Figure 5-7 Result example of WCS request to retrieve a product from a dataset in ADAM*

Same as above, but times provided in ANSI format

[https://reliance-das.adamplatform.eu/wcs?service=WCS&Request=GetCoverage&version=2.0.0&coverageID=MODh20chlYR\\_4km&subset=t\(1199148001\)&format=image/png&token=e62a785215f943c3ae0a2ca3cb58b4f7&scale=0.1](https://reliance-das.adamplatform.eu/wcs?service=WCS&Request=GetCoverage&version=2.0.0&coverageID=MODh20chlYR_4km&subset=t(1199148001)&format=image/png&token=e62a785215f943c3ae0a2ca3cb58b4f7&scale=0.1)

Same as above, but with the RAINBOW colortable

[https://reliance-das.adamplatform.eu/wcs?service=WCS&Request=GetCoverage&version=2.0.0&coverageID=MODh20chlYR\\_4km&subset=t\(1199148001\)&format=image/png&colortable=RAINBOW&token=4d9a7c61297342498776cf7539a02789&scale=0.1](https://reliance-das.adamplatform.eu/wcs?service=WCS&Request=GetCoverage&version=2.0.0&coverageID=MODh20chlYR_4km&subset=t(1199148001)&format=image/png&colortable=RAINBOW&token=4d9a7c61297342498776cf7539a02789&scale=0.1)



*Figure 5-8 Result example of WCS request to retrieve a product from a dataset in ADAM with RAINBOW colortable*