EUMETSAT Data Access Client (EUMDAC) guide

V3.2, 18 Mar 2024

This user guide provides basic information for users to help with installing, configuring and using EUMETSATs Data Access Client (EUMDAC) command-line interface. It includes a number of examples of code, and details on how to customise products from the Data Tailor. If you want to explore more about EUMDAC visit our EUMDAC guide.

What is EUMDAC?

EUMDAC is the **EUM**ETSAT **D**ata **A**ccess **C**lient. It provides access to EUMETSAT data from a variety of satellite missions. As a Python library, it comes with many methods and helpers to use EUMETSATs APIs and services, like Data Store and Data Tailor. It also provides a variety of useful command-line utilities for data search, download and processing.

Note: *EUMDAC is constantly being developed further, but there are currently functional limitations in its usage. Please familiarise yourself with the current limitations in our* ***FAQ guide*** *before you start using it.*

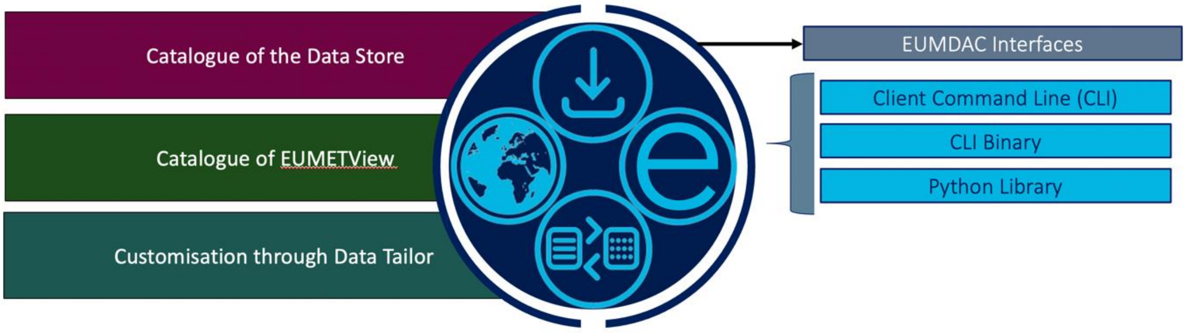

Catalogue of the Data Store
Catalogue of EUMETView
Customisation through Data Tailor
EUMDAC Interfaces
Client Command Line (CLI)
CLI Binary
Python Library

Figure 1: Schematic overview of the EUMETSAT Data Access Client

Installing EUMDAC

The installation of EUMDAC offers the possibility to use both the **command line interface** and the **Python library**. It benefits users who want to access the EUMETSAT data access services programmatically.

Dependencies

EUMDAC **requires Python 3.7** or higher.

The `requests` library is required for the use of EUMDAC. EUMDAC will install it automatically during setup if it is not already installed.

Installing from source

To install EUMDAC from the development source, clone the repository and install it locally.

```
git clone https://gitlab.eumetsat.int/eumetlab/data-services/eumdac.git
cd eumdac
pip install.
```

Installing with PIP

The EUMDAC Python package is available through **PyPI**.

```
pip install eumdac
```

Installing with Anaconda

The EUMDAC Python package is available through **Conda**.

```
conda install -c eumetsat eumdac
```

Command Line

The CLI allows you to build workflows using shell commands, either interactively within the command prompt or with a script. Many common cases are covered by CLI sub-commands. It is often more convenient to use a ready-made command, as opposed to implementing similar functionality as a Python script.

Starting to use EUMDAC

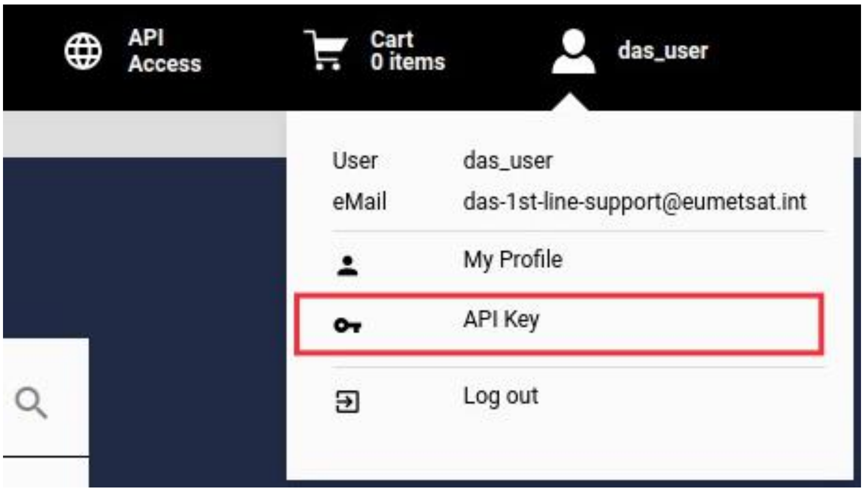
The following command in EUMDAC will authenticate the user. The placeholders <ConsumerKey> and <ConsumerSecret> need to be replaced with the user's personal credentials copied from the API Key Management page shown below.

```
eumdac --set-credentials <ConsumerKey> <ConsumerSecret>
```

The authentication with the EUMDAC Python library looks as the following:

```
import eumdac
        # Insert your personal key and secret into the single quotes
        consumer_key = 'YOUR_CONSUMER_KEY'
        consumer_secret = 'YOUR_CONSUMER_SECRET'
        credentials = (consumer_key, consumer_secret)
        token = eumdac.AccessToken(credentials)
        print(f"This token '{token}' expires {token.expiration}")
```

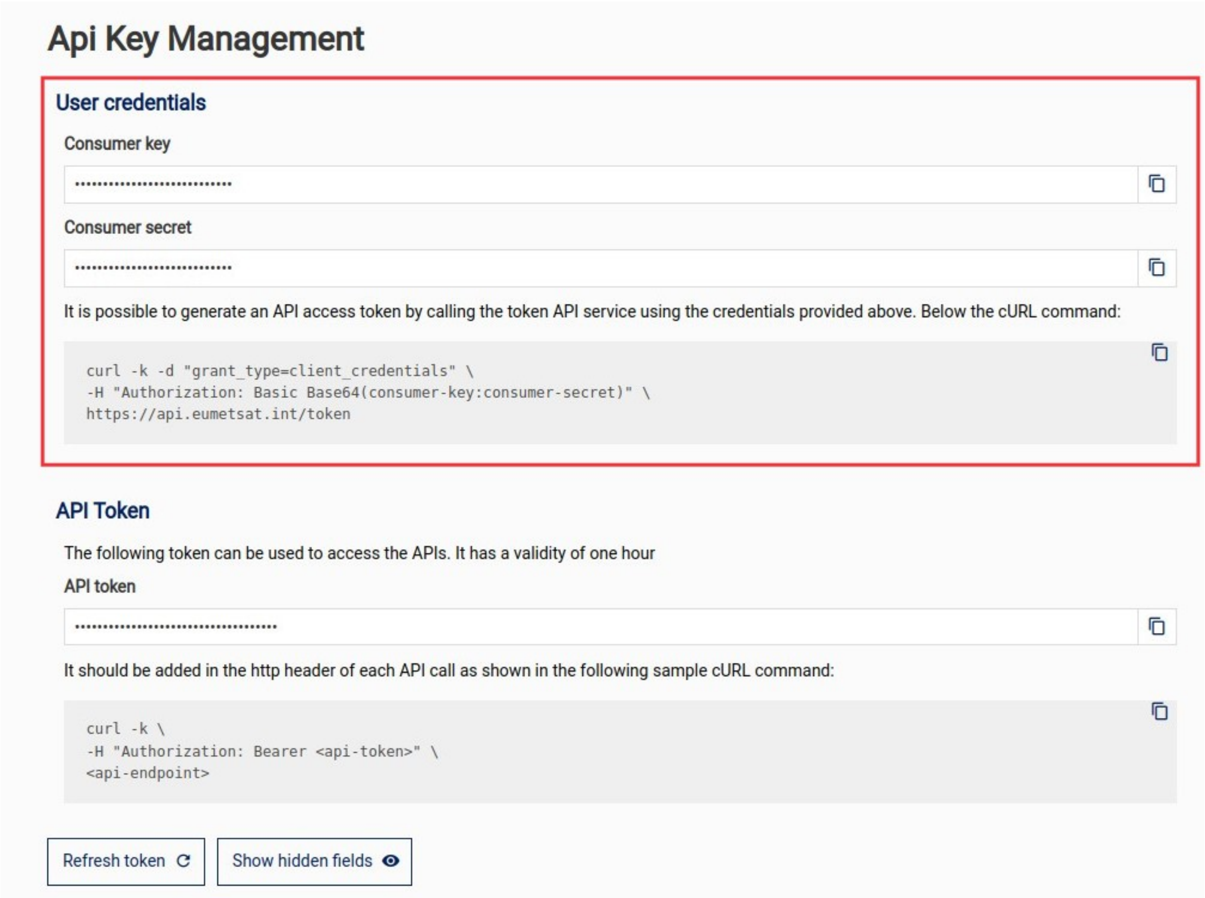
Users with EO Portal credentials can authenticate while using EUMDAC with the help of the consumer key and secret. In order to retrieve these details, you would need to click on your username on the top right hand corner and then select 'API Key'.



User das_user
eMail das-1st-line-support@eumetsat.int

Figure 2: Access API Key Management page from the Data Store

This will open the API Key Management page where the consumer key and consumer secret can be seen in the User Credentials section.



Api Key Management

User credentials

Consumer key

••••••••••••••••••••••••••••••

Consumer secret

••••••••••••••••••••••••••••••

It is possible to generate an API access token by calling the token API service using the credentials provided above. Below the cURL command:

```
curl -k -d "grant_type=client_credentials" \
-H "Authorization: Basic Base64(consumer-key:consumer-secret)" \
https://api.eumetsat.int/token
```

API Token

The following token can be used to access the APIs. It has a validity of one hour

API token

••••••••••••••••••••••••••••••••••••••

It should be added in the http header of each API call as shown in the following sample cURL command:

```
curl -k \
-H "Authorization: Bearer <api-token>" \
<api-endpoint>
```

Figure 3: Obtain user credentials from the API Key Management page

Get and describe available collections using EUMDAC

To explore the collections through EUMDAC Command Line Interface, you can use your terminal. This step assumes you have already installed EUMDAC on your machine or you are using the standalone binary CLI.

Below you can find the basic eumdac describe command, which list you all available collections:

```
eumdac describe
```

To search for a specific collection, eg all Sentinel-3 collections, you can use the '–filter' argument:

```
eumdac describe --filter "*Sentinel-3*"
```

To get more details about a specific collection, you can use the argument '-c' to indicate the desired collection:

```
eumdac describe -c EO:EUM:DAT:METOP:OSI-104
```

Or print collection attributes with the EUMDAC Python library:

```
datastore = eumdac.DataStore(token)
        # Select collection
        selected_collection = datastoreget_collection('
EO:EUM:DAT:METOP:OSI-104')
        # Print collection attributes
        print(selected_collection.title)
        print(selected_collection.abstract)
```

For additional arguments, refer to our EUMDAC

The following GIF shows you how your terminal should look like when using the "describe" command.

```
$
```

Search products using EUMDAC

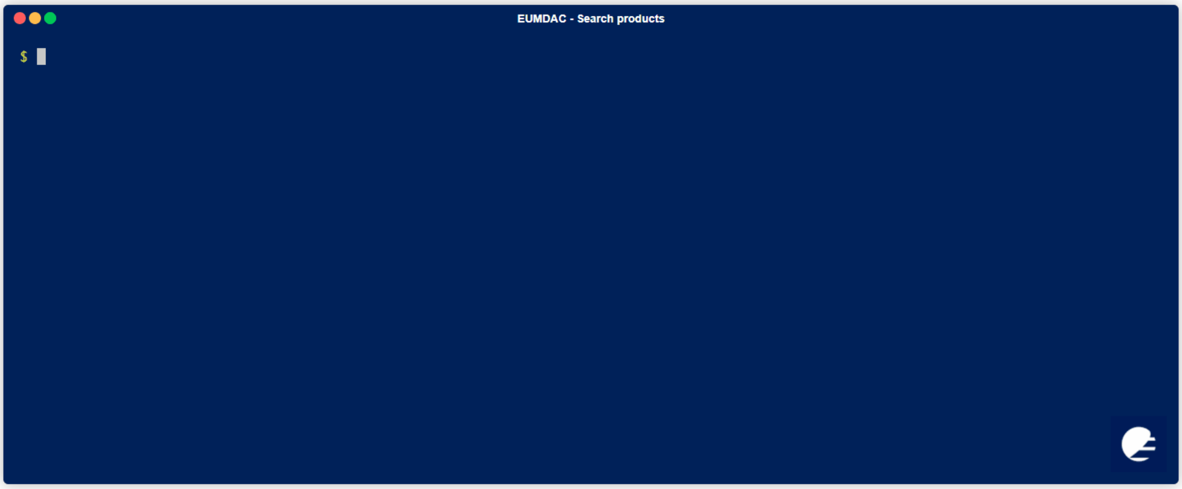
To search the collections through EUMDAC Command Line Interface, you can your terminal. Below you find an example where the argument '-c' indicates the collection; '-s' and '-e' define the starting and ending time, respectively and; '--bbox' defines the area of interest.

```
eumdac search -c EO:EUM:DAT:METEOP:OSI104 -s 2012-12-21 -e 2012-12-22
--bbox 51.28 51.69 0.51 0.33
```

Use the EUMDAC Python library to search and filter products:

```
# Set search parameters
        area = '51.28, 51.69, 0.51, 0.33'
        start = datetime.datetime(2021, 11, 10, 8, 0)
        end = datetime.datetime(2021, 11, 10, 9, 0)
        # Retrieve datasets that match our filter
        products = selected_collection.search(
         bbox=area,
         dtstart=start,
         dtend=end)
        # Print found datasets
        for product in products:
            print(product)
```

The following GIF shows you how your terminal should look like when using the "search" command.



```
$ 
```

Download uncustomised products using EUMDAC

To download your uncustomised products through EUMDAC Command Line Interface, you can use your terminal. Below you find an example of download command for a an uncustomised product.

To customise your products through EUMDAC Command Line Interface, you can use your terminal. Below you find an example of a customisation chain.

```
eumdac tailor post -c EO:EUM:DAT:MSG:HRSEVIRI -p MSG4-SEVI-MSG15-
0100-NA-20220304101243.253000000Z-NA --chain "product: HRSEVIRI,
format: geotiff"
```

For additional filtering arguments, refer to our **Data Store detailed guide** or our **Jupyter notebooks** for comprehensive Python examples. All the filters available in the GUI correspond to a specific command in EUMDAC.

Customising your products through Data Tailor

To customise your products through EUMDAC Command Line Interface, you can use your terminal. Below you find an example of a customisation chain.

```
eumdac tailor post -c EO:EUM:DAT:MSG:HRSEVIRI -p MSG4-SEVI-MSG15-0100-NA-20220304101243.253000000Z-NA --chain "product: HRSEVIRI, format: geotiff"
```

For additional filtering arguments, refer to our EUMDAC Guide. If you scroll down after the GIF listed, you will find a table of Commands. Investigate it to understand all the different arguments you can use.

Download tailored products using EUMDAC

To download your customised products through EUMDAC Command Line Interface, you can use your terminal. Below you find an example of download command.

```
eumdac tailor download [JobID] -o user/Desktop
```