

# Access to Sentinel-3 Marine Centre data

Bruno Lucas<sup>1</sup>, Remko Scharroo<sup>2</sup>, Carolina Nogueira Loddó<sup>2</sup>, Cristina Martin-Puig<sup>2</sup>, Salvatore Dinardo<sup>1</sup>  
Vincenzo Santacesaria<sup>2</sup>, Melad Nassar<sup>3</sup>, Ilaria Parodi<sup>4</sup>

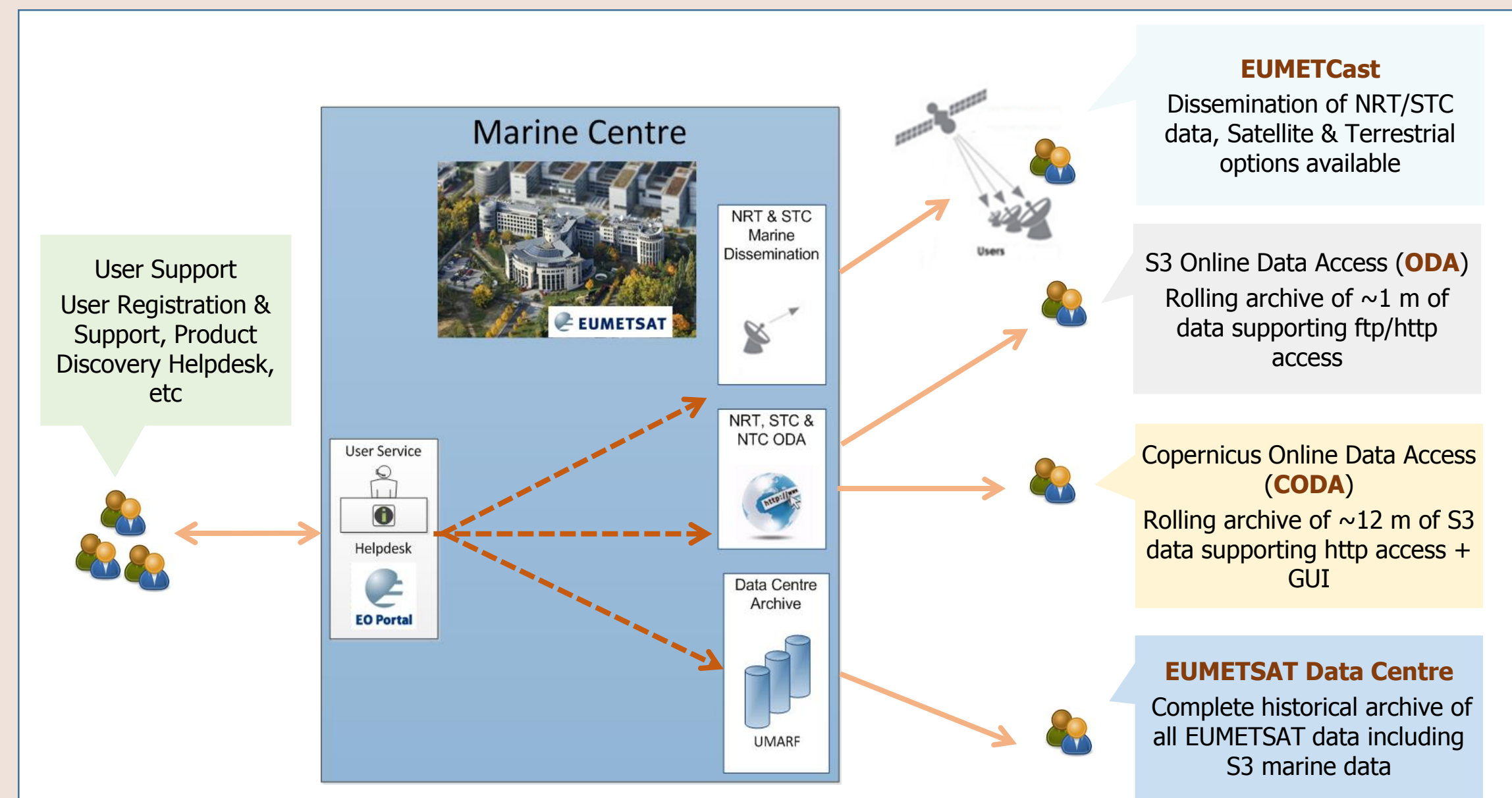
authors' contact: s3.stm@eumetsat.int

<sup>1</sup> HE Space, Darmstadt, Germany <sup>2</sup> EUMETSAT, Darmstadt, Germany <sup>3</sup> CS, Darmstadt, Germany <sup>4</sup> SCISYS, Darmstadt, Germany



## Introduction

The Copernicus Sentinel-3 (S3) Payload Data Ground Segment (PDGS) is distributed between EUMETSAT (European Organisation for the Exploitation of Meteorological Satellites) and ESA (European Space Agency). The Sentinel-3 Marine Centre is located at EUMETSAT. The Marine Centre is where NRT and offline marine products are systematically generated and the marine calibration, validation, quality control and mission performance activities are carried out. Marine Products/Data are archived in a long-term archive, made available in an online rolling archive and via EUMETCast.



- ❖ **EUMETCast** is EUMETSAT's integrated dissemination system for the delivery of near real-time environmental data including S3, Jason-2/3, Meteosat, Metop and later on S4 and S5. Service utilises low cost, user-friendly satellite receiving equipment similar to satellite TV;
- ❖ **ODA** is a rolling archive with reserved access to S3VT and Copernicus institutional Users.
- ❖ **EUMETSAT Data Centre** provides access to the long-term archive of S3 and EUMETSAT mission data. Order options allow you to select your chosen format, spatial and spectral sub-setting and apply data compression; Resulting data can be retrieved from an online server (http-download) or via offline media, depending on the volume of data requested; The Data Centre catalogue spans the whole satellite mission life. It guarantees the long-term preservation of these data which are critical for the generation of climate data records.
- ❖ **CODA** is an online rolling archive with http access. It has a web interface and allows for automated download. It is discussed in detail in this poster.

## PDGS Jargon

- Processing Baseline** – Set of Processors (IPFs) and Static Auxiliary Data (ADFs) that generate data with a given baseline
- NRT** – Near Real Time - Data generated and disseminated within 3 hours of sensing (equivalent to OGDR in Jason nomenclature)
- STC** – Short Time Critical - Data generated and disseminated within 48 hours of sensing (equivalent to IGDR in Jason nomenclature)
- NTC** – Non Time Critical - Data generated and disseminated within 30 days of sensing (equivalent to GDR in Jason nomenclature)
- SR\_2\_WAT** – S3 Level 2 data (produced in NRT, STC, NTC)
- SR\_1\_SRA** – S3 Level 1B data (produced in NRT, STC, NTC)
- SR\_1\_SRA\_A** – S3 Level 1A data (produced in STC and NTC)
- SR\_1\_SRA\_BS** – S3 Level 1B-S data (produced in STC and NTC)

## Account Setup

- ❖ In order to use CODA or the Data Centre you need to create a EUM EOPortal account
  - ❖ Go to URL: <https://eoportal.eumetsat.int> and select 'New User – Create New Account'
  - ❖ You'll receive an email with a confirmation code (click on it to confirm your account – as per email instructions).
- ❖ **Notice:** Prior to the first use of the script, wget/curl you need to login to the specific CODA server via web browser.

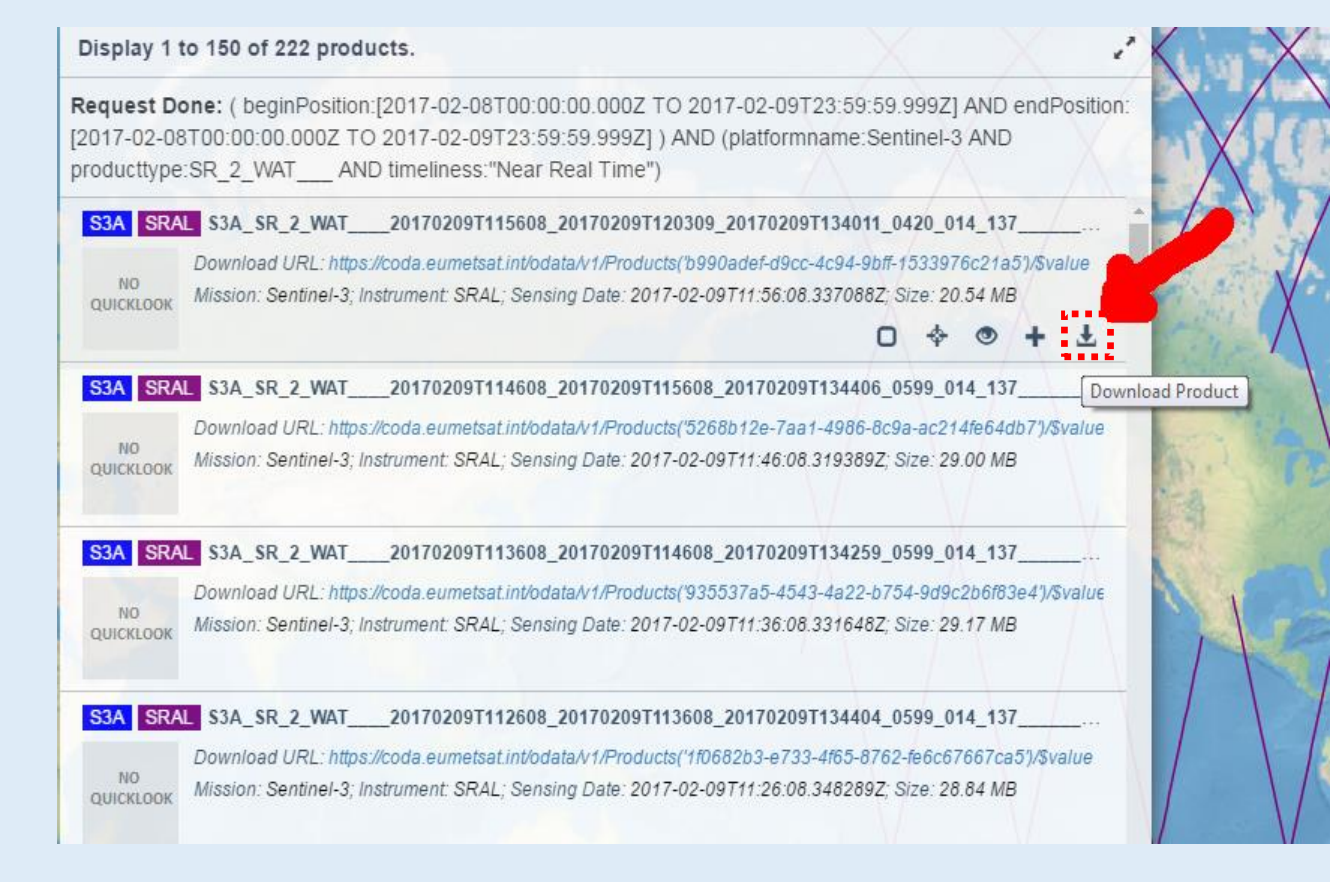
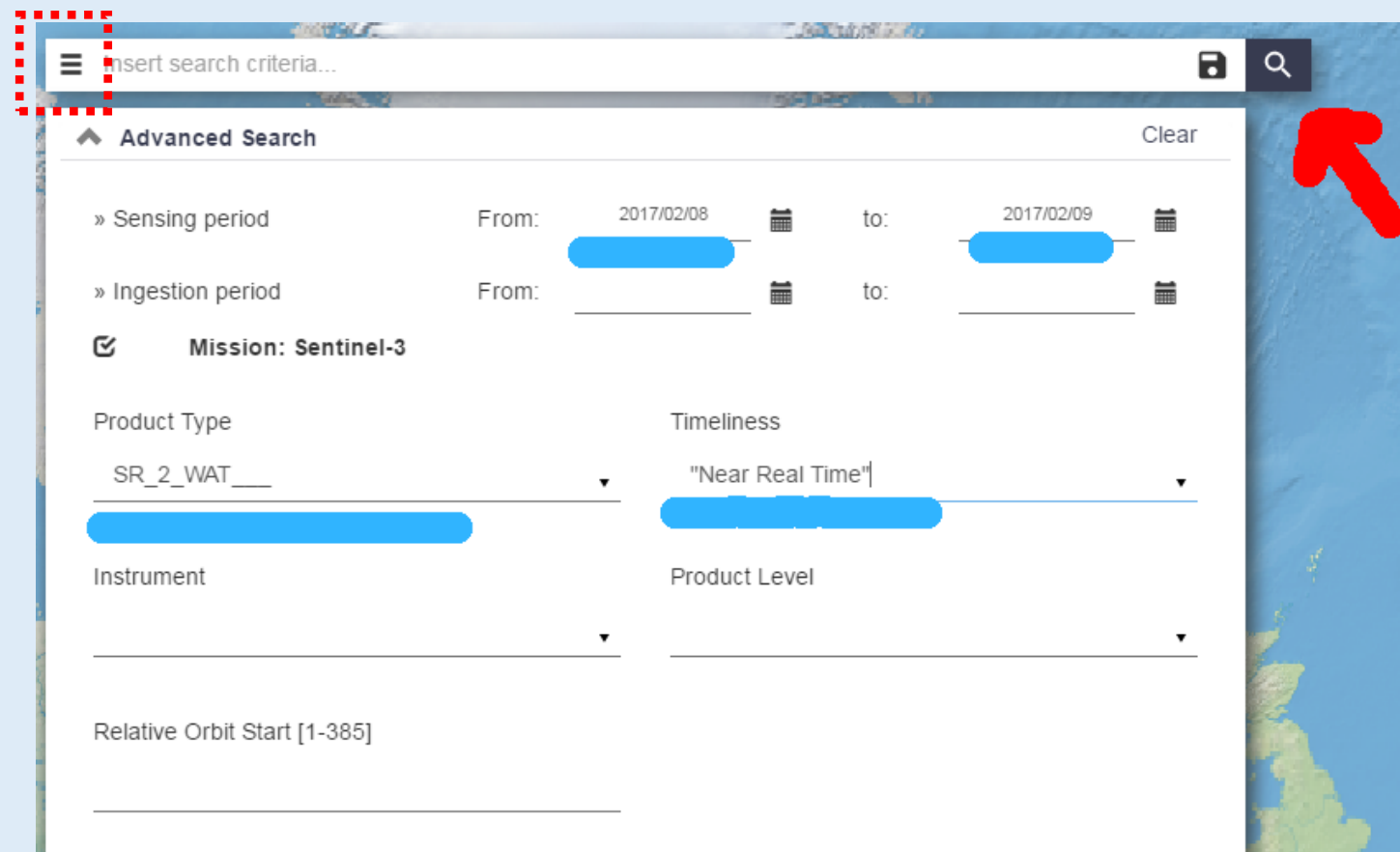
## Reprocessing "Spring 2018" Dataset

- ❖ Generated with PB 2.27, spans from 2016-03-01 to 2018-01-19
- ❖ Since 2018-04-04 the data is created with PB 2.33, a minor evolution to PB 2.27.
- ❖ The STM Reprocessing dataset is available in a special CODA server: <https://codarep.eumetsat.int>
- ❖ For **Download via Web browser** just go to the above site and proceed as in the normal CODA server. Notice that at the time of writing only SRAL L2, L1B are available (in NTC timeliness).
- ❖ For **Download with script** add the following parameter to your *dhusget.sh* script invocation: `-d codarep.eumtsat.int`
- ❖ More details about the S3 Processing Baselines can be found here: <https://www.eumetsat.int/website/home/Data/Sentinel3Services/Altimetry/index.html>
- ❖ A video on how to download and visualize Altimetry data from CODA is available: here: <https://youtu.be/vucr65nUf>

## CODA (Copernicus Online Data Access)

- ❖ **Download via web browser**
  - ❖ After the register you can now download data from CODA, either via the icon on your EO Portal account or by going directly to the URL: <https://codarep.eumetsat.int>
  - ❖ You may be requested to login with your EOPortal account, please do so.

Example – Download SR\_2\_WAT NRT from 2017-02-08 (sensing) – if the product is no longer available – due to the rolling archive nature of CODA – please try with more recent dates.



Click the highlighted icon to download the product (it will be zipped)

Other options are:  
Select/unselect Product  
Zoom to product  
View Product Details  
Add to Cart

Click on the icon of the left search field to open the 'Advance Search' and after the magnifying glass icon to do a search

## Download via Script

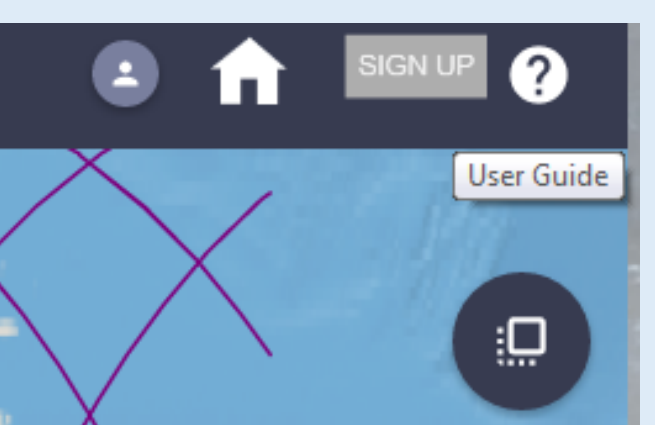
- ❖ Using **dhusget** script (<https://codarep.eumetsat.int/manual/dhusget.sh>)
  - ❖ Uses *wget* and *bash* and runs on Linux/Unix
- ❖ `bash dhusget.sh -u $USER -p $PASS -T SR_2_WAT -S 2017-02-08T00:00:00.000Z -E 2017-02-08T23:59:59.999Z -F 'Timeliness: "Near Real Time"' -o product`

```
bash dhusget.sh --help
```

There are a lot of other options to the script, below are some of the more relevant

- t <time in hours> : search for products ingested in the last <time in hours> (integer)
- s <ingestion\_date\_FROM> : Search for products ingested after the date
- e <ingestion\_date\_TO> : Search for products ingested before the date
- S <sensing\_date\_FROM> : Search for products with sensing date greater than the date
- E <sensing\_date\_TO> : Search for products with sensing date less than the date
- c <lon1,lat1:lon2,lat2> : Search for products intersecting a rectangular Area of Interest
- T <product type> : Search products according to the specified product type  
SR\_2\_WAT; SR\_1\_SRA; SR\_1\_SRA\_A; SR\_1\_SRA\_BS
- F <free OpenSearch query> : free text OpenSearch query. The query must be written enclosed by single apexes '<query>'.  
(e.g. -F 'platformname:Sentinel-1 AND producttype:SLC')
- o <download> : THIS OPTION IS MANDATORY FOR DOWNLOADING.  
Accepted values for <download> are:
  - product : download the Product ZIP files (manifest file included)
  - manifest : download only the manifest files
  - all : download both the Product ZIP files and the manifest files, and provide them in separate folders.
- q <XMLfile> : write the OpenSearch query results in a specified XML file.
- C <CSVfile> : write the list of product results in a specified CSV file.
- r <file> : download the products listed in an input <file>
- d <CODA server> : CODA server to be used (on the case of reprocessing use codarep.eumtsat.int)

User Guide (PDF) Support for Web interface, script, APIs...



Examples on usage with OPEN DATA PROTOCOL (ODATA), curl and wget are shown.

Examples on filtering based on location, generation time, ingestion time are also shown

Get this poster at: [http://bit.ly/s3eum\\_coda](http://bit.ly/s3eum_coda)



DON'T PANIC!!! ☺  
Write an email to OPS@EUMETSAT.INT

## Advanced CODA Usage

### Download Using CURL or WGET directly

- ❖ In the case the script *dhusget.sh* is too restrictive or does not fit your needs you can use directly *wget* or *curl*
  - ❖ `curl -u {USERNAME}:{PASSWORD} <URI_QUERY>`
  - ❖ `wget --no-check-certificate --user={USERNAME} --password={PASSWORD} <URI_QUERY>`
  - ❖ Where <URI\_QUERY> is a valid OData URI or OpenSearch URI. Replace {USERNAME} and {PASSWORD} with your credentials.
- ❖ Example, list the L2 NRT products ingested in CODA in the last day; List a maximum of 100 products (hard limit):
  - ❖ `wget --no-check-certificate --user={USERNAME} --password={PASSWORD} --output-document=query_results.txt "https://codarep.eumetsat.int/search?q=SR_2_WAT AND ingestionDate:[NOW-1DAY TO NOW] AND timeliness: \"Near Real Time\" &rows=100"`
  - ❖ The output file will contain a list of products with all the details of the products. This file is in JSON format. To download the actual data products with *wget* it is necessary to parse this file and obtain the IDs and SEN3 filenames.
- ❖ **Download full product** (it will be downloaded compressed inside a zip):
  - ❖ `wget --no-check-certificate --user={USERNAME} --password={PASSWORD} "https://codarep.eumetsat.int/odata/v1/Products('371e49c5-300a-4593-be21-3f3f7ea43753')/\$value" -O some.SEN3.zip`
  - ❖ This zip contains a L2 product. Inside you'll find the SEN3 SAFE format (directory with manifest `xdffmanifest.xml`, `standard_measurements.nc`, `enhanced_measurements.nc` and `reduced_measurements.nc`)
- ❖ **Download only one netCDF**, for instance the `standard_measurements.nc` for one SEN3 product:
  - ❖ `wget --no-check-certificate --user={USERNAME} --password={PASSWORD} "https://codarep.eumetsat.int/odata/v1/Products('371e49c5-300a-4593-be21-3f3f7ea43753')/Nodes('S3A_SR_2_WAT_20171017T055959_20171017T07060959_20171017T072402_0599_023_233_MAR_O_NR_002.SEN3')/Nodes('standard_measurement.nc')/\$value" -O standard_measurement.nc`
  - ❖ More examples are available on the online manual.

### Systematic download and Filtering

- ❖ It is suggested to enable a *crontab* job to download systematically data. For instance, every Hour download the data ingested in the last Hour. Use the flag `-t N_Hours` in the download *dhusget.sh* script
- ❖ Filtering shall be done outside of the *dhusget.sh* script and a selected list of files can be provided to the download script:
  - ❖ Product ID and SEN3 filename can be obtained from the returned list of files search (`-c list_files_search`). This file has the following format:
    - ❖ `CODA_URL('ID')`,
    - ❖ `SEN3_PRODUCT_NAME_NO_EXTENSION`,
    - ❖ `http://codarep.eumetsat.int/odata/v1/Products('d3302dce-d671-40e5-acc3-2914a3e9c7e8')`,
    - ❖ `S3A_SR_2_WAT_20171016T065631_20171016T065820_20171016T090914_0108_023_220_MAR_O_NR_002`,
  - ❖ Filter the desired files to download and provide the list of SEN3 files with `-r input_file`. This *input\_file* needs to have the following format (without '), one per line:
    - ❖ `' x ID x SEN3_product_minus_extension'`
    - ❖ `' x 23e4563f-807e-4623-b20f-b41f7d4dea00 x`
    - ❖ `S3A_SR_2_WAT_20171016T203421_20171016T211904_20171016T140104_2683_023_199_MAR_O_ST_002'`