



Sentinel 3 Marine Copernicus Data Access User Manual

https://coda.eumetsat.int

Doc.No.:EUM/OPS-SEN3/MAN/16/880763Issue:v2B DraftDate:18 October 2018WBS/DBS:

EUMETSAT Eumetsat-Allee 1, D-64295 Darmstadt, Germany Tel: +49 6151 807-7 Fax: +49 6151 807 555 http://www.eumetsat.int

© EUMETSAT The copyright of this document is the property of EUMETSAT.



Table of Contents

I

1	INTRODUCTION	4
	1.1 CODArep Online Rolling Archive	4
	1.2 File naming convention	5
	1.3 Acronyms scheme	9
	1.4 OLCI and SLSTR bands description	11
		11
	1.4.2 SLSTR	12
2	CODA GRAPHICAL USER INTERFACE	13
	2.1 Map viewer tool	13
	2.2 Search Options	13
	221 Full Text Search	14
	2211 Wildcards and Operators	14
	2212 Search Keywords	15
	222 Advanced Search	21
	2221 Available search parameters for the advanced search	22
	2.3 Search results list	23
	2.3.1 Product Details	24
	2.4 User Profile Panel	25
	2.5 Batch download from CODA and CODAren GUI	26
	2.5.1 Download the cart using a Download Manager (aria2)	26
		20
3	APIS AND BATCH SCRIPTING	28
	3.1 OPEN DATA PROTOCOL (ODATA)	28
	3.1.1 URI Components.	28
	3.1.2 Service Metadata Document	29
	3.1.3 Querving Products in the Data Hub archive	30
	3.2 OData System Query Options	32
	3.3 Download	39
	3.4 Discover Product Nodes	40
	3.5 Download manifest file knowing the product UUID	41
	3.6 Download of quick-look file knowing the product UUID	41
	3.7 Verifying Download integrity using the MD5 checksum	41
4	OPEN SEARCH	43
	4.1 URI Components	43
	4.2 Using Open Search to discover products	43
	4.2.1 Discover the list of the products stored in the archive	43
	4.2.2 Paging results	43
	4.2.3 Sorting results	44
	4.3 Discover the products over a predefined Area Of Interest (AOI): Geographical Search	44
	4.3.1 Polygon	44
	4.3.2 Point	45
	4.4 Open Search queries examples (combined with full text search)	45
-		4-
5	BATCH SCRIPTING	47
	5.1 Query via Curi	41
	5.2 Query via Wget	41
	5.3 Download via Wget	47
	5.4 Scripts Examples	48
	5.4.1 anusget script	48
	5.4.2 Udata-demo.sh script	51
6		53
0		55



7	USEFUL LINKS	.55
8	KNOWN ISSUES AND USEFUL TIPS FOR CODA WEB SERVICE	. 56

Page 3 of 57



1 INTRODUCTION

The Sentinel-3 Marine CODA (Copernicus Online Data Access) Web Service provides complete, free and open access to Sentinel-3 user products provided by EUMETSAT on behalf of the European Commission.

In order to access CODA GUI and CODA API functionalities, it is necessary to have an Earth Observation Portal (EO Portal) user account.

The Web Service consists of a rolling archive with http access up to 1 year of data online¹.

It provides Sentinel-3 Level-1 and Level-2 (only Marine) products for the following instruments:

- OLCI (Ocean and Land Colour Instrument),
- SLSTR (Sea and Land Surface Temperature Radiometer),
- SRAL (SAR Altimeter).

Sentinel-3 Data are provided in different latency modes, as shown in the following table:

Latency modes	Latency Description modes	
Near Real- Time (NRT)	Products available to users within three hours after sensing.	1 month
Non time critical (NTC)	Products available to the users within one month after sensing. This additional delay allows consolidation of some auxiliary or ancillary data (e.g. precise orbit data)	1 year
Short time critical (STC)	Products available to the users within 48 hours after sensing, due mainly to the consolidation of some auxiliary or ancillary data (e.g. preliminary restituted orbit data) (Only for SRAL products)	1 month

Table 1: Latency mode of CODA data

The selection of a product type in terms of delivery time (NRT, STC and NTC) is a trade-off between real-time needs and the final accuracy needed.

1.1 CODArep Online Rolling Archive

Sentinel-3 data on CODA are always processed using the last available algorithms, but algorithms are periodically updated. So, in order to allow users to be able to access consistent datasets, "older" OLCI and SRAL NTC data have been reprocessed according to the latest standards. These reprocessed datasets are available via the online archive at this link: https://codarep.eumetsat.int.

¹ In case of need for data older than one year, please go to EUMETSAT Data Centre https://www.eumetsat.int/website/home/Data/DataDelivery/EUMETSATDataCentre/index.html



Like for CODA online archive, users can access CODArep archive using their Earth Observation Portal (EO Portal) credentials or setting up an EO Portal account.

In the table below, some information about the reprocessed datasets.

	SRAL reprocessed data	OLCI reprocessed data
Timeliness	Non Time Critical data (NTC)	Non Time Critical data (NTC)
Processing Baseline	2.15	2.23
Dates	From 15 June 2016 to 15 April 2017 (reprocessed data with the same baseline after 15 April 2017 onwards can be found on CODA)	From 26 April 2016 to 29 November 2017 (reprocessed data with the same baseline from 30 November 2017 onwards can be found on CODA)
Product Types and Levels	L1B (SR_1_SRA), L2 (SR_2_WAT) L1A and L1B-S to come.	L1 (OL_1_EFR and OL_1_ERR) L2 (OL_2_WFR and OL_2_WRR)

Table 2: Data on CODArep

1.2 File naming convention

The file naming convention for CODA products is identified by the sequence of fields described here:

SA3_SS_L_TTTTTT_yyyymmddThhmmss_YYYYMMDDTHHMMSS_YYYYMMDDTHHMMSS__G GG_.SEN3

EUMETSAT

EUM/OPS-SEN3/MAN/16/880763 v2B Draft, 18 October 2018

CODA user manual

Field	Size in Char	Element	Description
MMM	3	Mission ID	S3A = Sentinel-3A
SS	2	Data source of the instrument	OL = OLCI SL = SLSTR SR = SRAL
L	1	Processing level	1 = Level-1 2 = Level-2
	6 Dat	Data Type ID	Level 1 OLCI data "EFR" = TOA radiances at full resolution "ERR" = TOA radiances at reduced resolution
TTTTTT			Level 2 OLCI data "WFR" = Full Resolution Ocean Colour, Water & atmosphere parameters "WRR" = Reduced Resolution Ocean Colour, Water & atmosphere parameters
			Level 1 SLSTR data "RBT" = TOA Radiances and Brightness Temperature
			Level 2 SLSTR data "WST" = L2P Sea Surface Temperature
			Level 1 SRAL data

Page 6 of 57



EUM/OPS-SEN3/MAN/16/880763 v2B Draft, 18 October 2018

CODA user manual

			"SRA" = echo LRM, SAR Ku, SAR C "SRA_A_" = echo PLRM and SAR mode (resolution 80Hz) "SRA_BS_" = echo LRM, PLRM and SAR mode (resolution 20Hz), completed with SAR expert information
			Level 2 SRAL data "WAT"= Water products
yyyymmddThhmmss	15	Data Start time (Sensing time for the Instrument data products)	year, month, day, T, hour, minutes, seconds
YYYYMMDDTHHMMS S	15	Data Stop time (Sensing time for the Instrument data products)	year, month, day, T, hour, minutes, seconds
YYYYMMDDTHHMMS S	15	Creation Date	year, month, day, T, hour, minutes, seconds
<instance_id></instance_id>	17	 Instance ID for the instrument data products, disseminated in: Stripes: Duration,"_", cycle number, "_", relative orbit number,"_", 4 underscores "_" DDDD_CCC_LLL Frames: Duration, "_", cycle number, "_", relative orbit number, "_", frame along track coordinate DDDD_CCC_LLL_FFFF 	Duration "DDDD"= 4 digits; orbit duration Sensing data time interval in seconds. Cycle "CCC"= 3 digits; cycle number at the start sensing time of the product. Relative orbit "LLL"= 3 digits; relative orbit number within the cycle at the start sensing time of the product.

Page 7 of 57



EUM/OPS-SEN3/MAN/16/880763 v2B Draft, 18 October 2018

CODA user manual

			Frame along track coordinate "FFFF"= four digits; elapsed time in seconds from the ascending node indicating the frame start time.
GGG	3	Product Generating Centre	MAR = Marine Processing and Archiving Centre MRn = Marine Re-processing Centre number n
<class_id></class_id>	8	Identifies the class ID for instrument data products with conventional sequence "P_XX_NNN"	 P = platform (O for operational, F: for reference, D for development, R for reprocessing or 1 underscore "_" if not relevant) XX = timeliness of the processing workflow (NR for NRT, ST for STC, NT for NTC) NNN = baseline collection or data usage. 3 underscores "_" if not relevant.
<extension></extension>	4	Filename extension	The adopted extension is: .SEN3

Table 3: Sentinel-3 naming convention

Page 8 of 57



1.3 Acronyms scheme

Sentinel-3 products are provided according to the following acronyms scheme:

Product Type	Level Description		Link To EUMETSAT Product Navigator
	OLO	CI (Ocean and Land Colour Inst	trument)
OL_1_EFR1 Full resolution		Full resolution top of atmosphere radiance.	OLCI Level 1B Full Resolution in NTC OLCI Level 1b Full Resolution in NRT
OL_1_ERR 1 Reduced resolution top atmosphere radiance		Reduced resolution top of atmosphere radiance.	OLCI Level 1B Reduced Resolution in NRT OLCI Level 1B Reduced Resolution in NTC
OL_2_WFR 2 Full resolut atmosphere prod		Full resolution water & atmosphere geophysical products.	OLCI Ocean Colour Full Resolution in NRT OLCI Ocean Colour Full Resolution in NTC
OL_2_WRR	2	Reduced resolution water & atmosphere geophysical products.	OLCI Ocean Colour Reduced Resolution in NRT OLCI Ocean Colour Reduced Resolution in NTC

Table 4: OLCI acronym scheme

Product Type	Level	Description	Link To Eumetsat Product Navigator		
SLSTR (Sea and Land Surface Temperature Radiometer)					
SL_1_RBT	1	Brightness temperature and radiances.	SLSTR Level 1B Radiances and Brightness Temperatures in NRT SLSTR Level 1B Radiances and Brightness Temperatures in NTC		
SL_2_WST	L_2_WST 2 Level 2P Sea Surface Temperature (GHRSST like)		SLSTR Sea Surface Temperatures (SST) in NRT SLSTR Sea Surface Temperatures (SST) in NTC		

Table 5: SLSTR acronym scheme

Page 9 of 57



Product Type	Level	Description	Link To Eumetsat Product Navigator
SRAL (SAR Rada	ar Altimeter	;)	
SR_1_SRA	1B	Echos parameters for LRM, PLRM and SAR mode (resolution 20Hz).	SRAL Level 1B in NRT SRAL Level 1B in NTC SRAL Level 1B in STC
SR_1_SRA_A_	1A	Echos parameters for PLRM and SAR mode (resolution 80Hz)	SRAL Level 1A in NTC SRAL Level 1A in STC
SR_1_SRA_BS 1A		Echos parameters for LRM, PLRM and SAR mode (resolution 20Hz), completed with SAR expert information	SRAL Level 1B in NTC SRAL Level 1B in STC
SR_2_WAT	2	1-Hz and 20-Hz Ku and C bands parameters (LRM/SAR/PLRM), waveforms. Over Water.	SRAL Altimetry Global in NRT SRAL Altimetry Global in NTC

Table 6: SRAL acronym scheme

Page 10 of 57



1.4 OLCI and SLSTR bands description

In the following paragraphs a description of the information about the different bands for every $\underline{OLCI \text{ and } SLSTR}$ instruments is provided.

1.4.1 OLCI

OLCI observation is performed simultaneously in 21 spectral bands, described in the following table.

Band	λ centre (nm)	Width (nm)	Description
Oa1	400	15	Aerosol correction, improved water constituent retrieval
Oa2	412.5	10	Yellow substance and detrital pigments (turbidity)
Oa3	442.5	10	Chl absorption max., biogeochemistry, vegetation
Oa4	490	10	High Chl, other pigments
Oa5	510	10	Chl, sediment, turbidity, red tide
Oa6	560	10	Chlorophyll reference (Chl minimum)
Oa7	620	10	Sediment loading
Oa8	665	10	Chl (2nd Chl abs. max.), sediment, yellow substance and vegetation
Oa9	673.75	7.5	For improved fluorescence retrieval and to better account for smile together with the bands 665 and 680 nm
Oa10	681.25	7.5	Chl fluorescence peak, red edge
Oa11	708.75	10	Chl fluorescence baseline, red edge transition
Oa12	753.75	7.5	O2 absorption/clouds, vegetation
Oa13	761.25	2.5	O2 absorption band/aerosol corr.
Oa14	764.375	3.75	Atmospheric correction
Oa15	767.5	2.5	O2A used for cloud top pressure, fluorescence over land
Oa16	778.75	15	Atmos. corr./aerosol corr.
Oa17	865	20	Atmos. corr./aerosol corr., clouds, pixel co-registration
Oa18	885	10	Water vapour absorption reference band. Common reference band with SLSTR instrument. Vegetation monitoring
Oa19	900	10	Water vapour absorption/vegetation monitoring (max. reflectance)

Page 11 of 57



Oa20	940	20	Water vapour absorption, atmos./aerosol corr.
Oa21	1 020	40	Atmos./aerosol corr.

Table 7: OLCI bands description

1.4.2 SLSTR

SLSTR observation is performed simultaneously in 11 spectral bands, described in the following table (please, note that F1 and F2 fire bands are based on the same detectors as S7 and S8 but with an increased dynamic range to prevent saturation over fires).

Band	λ centre (nm)	Width (nm)	Description	Comments		Resolution (m)
S 1	0.555	0.02	Cloud screening, vegetation monitoring, aerosol	Visible		
S2	0.659	0.02	NDVI, vegetation monitoring, aerosol	Visible Near IR		
S 3	0.865	0.02	NDVI, cloud flagging, Pixel co-registration		Solar reflectan	500
S4	1.375	0.015	Cirrus detection over land		ce bands	500
85	1.61	0.06	Cloud clearing, ice, snow, vegetation monitoring	Short- Wave IR		
S 6	2.25	0.05	Vegetation state and cloud clearing			
S 7	3.74	0.38	SST, LST, Active fire	Thermal infra-red		
S8	10.85	0.9	SST, LST, Active fire	Ambient bands		
S9	12	1	SST, LST	(200 K – 320 K)		1000
F1	3.74	0.38	Active fire	Thermal infra-red		
F2	10.85	0.9	Active fire	fire emission bands		

Table 8: SLSTR bands description



2 CODA GRAPHICAL USER INTERFACE

To access EUMETSAT CODA GUI: <u>https://coda.eumetsat.int</u>. To access EUMETSAT CODArep GUI: <u>https://codarep.eumetsat.int</u> Both the map viewers have the same appearance.

<complex-block>

Figure 1: CODA Graphical User Interface

2.1 Map viewer tool

The map tool provides a geographical search feature <u>for defining an Area of Interest (AoI) either in</u> <u>the shape of a box or of a polygon.</u> (Fig. 1).

Page 13 of 57



To define a region of interest, click on the <u>button "Box" or "Polygon" to define a rectangular or a</u> polygonal AOI icon to set "Draw region of Interest" modality and then click and drag draw on the map for drawing a selection boxthe selected area. By clicking on the "Search" button the application will search for any product whose footprint intersects or is included in the selected region.

Thanks to this geographic filter, the map tool can be combined with the full text search and the advanced search to further restrict the results (see next paragraphs).

2.2 Search Options

User can discovery the data ingested on CODA and CODArep systems using the Search Panel (see Figure 2 below).



Figure 2: CODA Search Panel

The Search Panel provide the user with two different search functionalities:

- Full text search: the user has just to write the keywords in the field.
- Advanced Search: the user can apply different filter parameters to the search and order the results.

2.2.1 Full Text Search

Search queries in CODA and CODArep archives can be performed by entering a text query in the full-text search bar. This query can be entered using also wildcards and operators.

2.2.1.1 Wildcards and Operators

Wildcards and Operators are used to restrict search queries. Wildcards, in particular, are useful when performing a query on the product filename. Operators are useful when combining different search criteria. Admitted wildcards and operators are shown in the tables below:



Wildcard	Description			
*	any sequence of zero or more characters			
?	any one character			
	T			

Operator	Description
AND	Narrow search and retrieve records containing all of the words it separates.
OR	Broaden search and retrieve records containing any of the words it separates.
NOT	Narrow search and retrieve records that do not contain the term following it.
()	Group words or phrases when combining Boolean phrases and to show the order in which relationships should be considered.

Table 9: Wildcards and Operators

Examples:

- "OLCI AND L2" returns all OLCI data in L2 Product Level.
- "OL_1_EFR____OR_OL_2_WRR___" returns all OLCI L1 Full Resolution products and all OLCI L2 Reduced Resolution products.
- "SLSTR NOT NRT" returns all the SLSTR products not in Near Real Time mode.
- "(SLSTR AND L2) NOT NRT" return all SLSTR products that are also in L2 Product Level and are not in Near Real Time latency mode.

2.2.1.2 Search Keywords

It is also possible to perform full-text queries using search keywords. The syntax format to enter in the Full text search bar is the following:

<keyword>:<values>

Depending on the keyword, value(s) can be specified as a single value or range of values. Search keywords can be combined with each other using Operators, for example: <keywordl>:<values> AND <keyword2>:<values>.

The following table contains the list of the keywords and their corresponding usage:



EUM/OPS-SEN3/MAN/16/880763 v2B Draft, 18 October 2018

CODA user manual

Search Keywords	Syntax and Examples	
<i>platformname:</i> Search based on the Satellite Platform name regardless of the serial identifier (e.g. A, B, C)	<pre>platformname:<platform name=""> e platformname:Sentinel-3</platform></pre>	
 beginposition: A time interval search based on the Sensing Start Time of the products. endposition: A time interval search based on the Sensing Stop Time of the products. ingestiondate: A time interval search based on the time of publication of the product on the Data Hub. 	<pre>These three keywords are used with a time range expressed with the following syntax: <pre><keyword>:[<timestamp> TO <timestamp>] The <timestamp> value can be expressed in one of the following formats: (ISO8601) yyyy-MM-ddThh:mm:ss.SSSZ NOW NOW-<n>MINUTE(S) NOW-<n>HOUR(S) NOW-<n>DAY(S) NOW-<n>MONTH(S) Being n = 1,2,,10,,100, Examples of search keywords with timestamps: beginposition:[2017-01-01T00:00:00.000Z TO NOW] beginposition:[2017-01-01T00:00:00.000Z TO NOW] beginposition:[2017-01-01T00:00:00.000Z TO 2017-02- 01T00:00:00.000Z] beginposition:[2017-01-01T00:00:00.000Z TO NOW] beginposition:[2017-01-01T00:00:00.000Z TO NOW] </n></n></n></n></timestamp></timestamp></timestamp></keyword></pre></pre>	

Page 16 of 57



EUM/OPS-SEN3/MAN/16/880763 v2B Draft, 18 October 2018

CODA user manual

	 endposition:[2017-01-01T00:00:00.000Z TO 2017-02- 01T00:00:00.000Z] endposition:[NOW-1HOUR TO NOW] ingestiondate:[2017-01-01T00:00:00.000Z TO NOW] ingestiondate:[2017-01-01T00:00:00.000Z TO 2017-02- 01T00:00:00.000Z] ingestiondate:[NOW-30MINUTES TO NOW]
	filename: <filename></filename>
filename:	Possible <filename> values are:</filename>
Search based on the product filename	 The full product file name expressed according to the product naming convention: MMM_BB_TTR_LFPP_YYYMMDDTHHMMSS_YYYYMMDDTHHM MSS_OOOOO_DDDDDD_CCCC Part of the filename using wildcards. Examples: filename: S3A_OL* filename:S3A_OL_1_EFR20170317T080500_20170317T0 80717_20170317T104619_0137_015_263_4140_MAR_0_NR_00 2 filename:*3A_OL_1_?20170317T080500*
	Syntax is the following:
footnrint	<pre>footprint:"intersects(<geographic type="">)"</geographic></pre>
Geographical search of the products whose footprint intersects or is included in a specific geographic type.	The <i><geographic< i=""> <i>type></i> value can be expressed as a polygon or as a point according to the syntax described below.</geographic<></i>

Page 17 of 57

EUMETSAT

EUM/OPS-SEN3/MAN/16/880763 v2B Draft, 18 October 2018

CODA user manual

 POLYGON <geographic type="">=POLYGON((PlLon PlLat, P2Lon P2Lat, , PnLon PnLat, PlLon PlLat)) where PlLon and PlLat are the Longitude and Latitude coordinates of the first point of the polygon in decimal degrees (DDD) format (e.g. 2.17403, 41.40338) and so on. The coordinates of the last point of the polygon must coincide with the coordinates of the first point of the polygon. The polygon describing the geographical area can have a maximum of 200 points that must be within an area described by 10 degrees of latitude and 10 degrees of longitude. Example: The polygon of the example is a bounding box around the Mediterranean Sea: footprint: "Intersects(POLYGON((-4.53 29.85, 26.75 29.85, 26.75 46.80, -4.53 46.80, -4.53 29.85)))"</geographic>
• POINT: <geographic type="">= Lat, Lon where the Latitude (Lat) and Longitude (Lon) values are expressed in decimal degrees (DDD) format (e.g. 41.40338, 2.17403).</geographic>
<pre>Examples: footprint:"intersects(Lat, Lon)" - Rome city centre: footprint:"intersects(41.9000, 12.5000)" - Etna Volcano: footprint:"intersects(37.7550, 14.9950)" - Bárðarbunga Volcano: footprint:"intersects(64.6300, -17.5300)"</pre>

Page 18 of 57



EUM/OPS-SEN3/MAN/16/880763 v2B Draft, 18 October 2018

CODA user manual

	 Istanbul city centre: footprint:"intersects(41.0136, 28.9550)" Paris city centre: footprint:"intersects(48.8567, 2.3508)" Mexico City city centre: footprint:"intersects(19.4333, -99.1333)" London city centre: footprint:"intersects(51.5072, 0.1275)"
<i>orbitnumber:</i> Absolute orbit number of the oldest line within the image data (the start of the product). <i>lastorbitnumber:</i>	<pre>These keywords can be used with a single value or with a range of values. The syntax is the following: orbitnumber:<orbitnumber> lastorbitnumber:<lastorbitnumber> orbitnumber:[<orbitnumber> TO <orbitnumber>] lastorbitnumber:[<lastorbitnumber> TO <lastorbitnumber>] Possible values for <orbit number=""> and <lastorbit number=""> so from</lastorbit></orbit></lastorbitnumber></lastorbitnumber></orbitnumber></orbitnumber></lastorbitnumber></orbitnumber></pre>
Absolute orbit number of the most recent line within the image data (the end of the product)	000001 to 999999. Examples: - orbitnumber:000020 (or orbitnumber:20) - lastorbitnumber:000510 (or lastorbitnumber:510) - orbitnumber:[001020 TO 001021] (or orbitnumber:[1020 TO 1021])
<i>orbitdirection:</i> Direction of the orbit (ascending, descending) for the oldest image data in the product (the start of the product).	Possible values are: Ascending, Descending Example: orbitdirection:Ascending

Page 19 of 57



EUM/OPS-SEN3/MAN/16/880763 v2B Draft, 18 October 2018

CODA user manual

<i>relativeorbitnumber:</i> Relative orbit number of the oldest line within the image data (the start of the product). <i>lastrelativeorbitnumber:</i> Relative orbit number of the most recent line within the image data (the end of the product)	<pre>These keywords can be used with a single value or with a range of values. The syntax is the following: relativeorbitnumber:<relativeorbitnumber> lastrelativeorbitnumber:<lastrelativeorbitnumber> relativeorbitnumber:[<relativeorbitnumber> TO <relativeorbitnumber>] lastrelativeorbitnumber:[<lastrelativeorbitnumber> T</lastrelativeorbitnumber></relativeorbitnumber></relativeorbitnumber></lastrelativeorbitnumber></relativeorbitnumber></pre>
<i>producttype:</i> Type of the product	Syntax is: producttype: <producttype> Possible values for for <producttype> are the following: OL_1_EFR, OL_1_ERR, OL_2_WFR, OL_2_WRR, SL_1_RBT, SL_2_WST, SR_1_SRA, SR_1_SRA_A, SR_1_SRA_BS, SR_2_WAT</producttype></producttype>

Table 10: Search Keywords in CODA and CODArep

Page 20 of 57



TIP for the search: when performing a query using the Advanced Search filters, the text of the query is displayed above the list of results (see Fig. 3).

It is possible to copy this text (copy the entire text included the brackets, excluding "Request Done:"), customize it adding new filters using Keywords and Operators as described in the tables above, and then paste the query in the text search bar and perform a new enhanced search to limit the number of results.

A Advanced Search		Display 1 to 25 of 966 products. Order By: Sensing Date 4	Select All
» Sort By:	Sensing Date	Request Done: (beginPosition [2017-03-10700 00 00 0000 2017-03-10700 00 00 00002 TO 2017-03-11723 59 59 9992 instrumentshortmame OL CI)	t TO 2017-03-11T23 59 59 9992] AND endPositi []) AND (platformname Sentinel-3 AND
 Order By: Sensing period 	Descending From: 201762413	Constant URL State (CC) Stat	235759_201703127015448_0179_015_187_2239_M succive66070ca.attl-456e.aacc-5ac85451433c5554aka 917-03-11722.54.59.2224552_0zae.640.58 MB
 Ingestion period Mission: Sentinel- Product Type 	From: II to	Contract UR: http://www.contracture.com/ Sac.OL_1_EFRJortPathT25459_201703111 Download UR: http://book.www.sac.ol/books/UR:http://books.www.sac.ol/books/UR:http://books.www.sac.ol/books/UR:http://books/UR:h	235756_201703137052044_0175_015_187_3228_M Such(R461506-442-445-9982-48-569819475sele 117-42-11723-54-59-2324582, Size 639-44 MB
instrument DLCI	Pioduct Level	COL BLA_CU_T_LFR0011011720559_20170011 Common URL https://commonwice.international.co	235459_201703131052915_0179_015_187_3059_M Andry 11174031-1037-456+8565-5014556+016-156 117-03-11722-51-58-2324582, Store 651-96 MB
Relative Orbit Start [1-385]		CON SAC.01_EFR30173311723519_20170311	235459_201703127014516_0179_015_187_3059_M kxxxx(x7342x4f-022-4633-6340-87465928204)\$kxxx (x7542-11722.5f.99.2224592,3cx; 652.92.MB

Figure 3: Filter of results using Keywords and Operators

2.2.2 Advanced Search

The advanced search panel is shown by clicking on the "Advanced Search" icon.

It consists of different search fields. Each search field allows to perform specific queries on the associated parameter. Each search field can be used singularly or in combination with: the others search fields, the full-text search bar and the geographic map tool.

It is also possible to Sort the elements of the result list using different parameters (Ingestion Date, Sensing Date) and then order the sorted list by ascending or descending order (see Fig. 4).



				Clear
Ingestion	Date			
Descendi	9			
From			to	
From			to:	
	Timel	iness		
	Produ	ict Level		
	tidiO	Number		
	Ingestion f	Ingestion Date Descending From: From: Timel Produ Produ Orbit	Ingestion Date Descending From: From: Timeliness Timeliness Oribit Number	Ingestion Date Descending From: B to: From: B to: Timeliness Product Level Orbit Number

Figure 4: CODA and CODArep Advanced Search Panel options

2.2.2.1 Available search parameters for the advanced search

Sensing Period: This search field is composed of two date entries. "Pick up date" calendars allow date selection. The query returns all the products whose sensing dates and times are included in the defined period. In particular it returns all the products that respond to both of the following criteria:

- Sensing start time equal or greater then 00:00:00 (hh:mm:ss) of the first selected date.
- Sensing stop time equal or less then 23:59:59 (hh:mm:ss) of the second selected date.

The Sensing Time corresponds to the time of the satellite on-board acquisition and it is stamped for each line of the acquired image scene. The sensing start and stop times of a product correspond to the time of the satellite on-board acquisition of respectively the first and last line of the image in the product. Click on "Clear date" for removing the search entry. Click on "Today" for selecting current date.

Ingestion period: This search field is composed of two date entries as for "Sensing Period". The query returns all the products whose publication dates and times on the Data Hub are included in the defined period. In particular it returns all the products that respond to both of the following criteria:

- Publication time equal or greater then 00:00:00 (hh:mm:ss) of the first selected date.
- Publication time equal or less then 23:59:59 (hh:mm:ss) of the second selected date.

Please note that the ingestion date does not correspond to the generation date of the product processed at the ground segment. The ingestion date is the date of publication of the product on the CODA rolling archive.

Cycle number: the cycle number is the number of times the satellite passed over the same geographical point on the ground. In the SENTINEL-3 operational phase, the orbit cycle is 27 days.

Page 22 of 57



Orbit Number: the absolute orbit number considers the orbits elapsed since the first ascending node crossing after launch. The relative orbit number is a count of orbits from 1 to the number of orbits contained in a repeat cycle.

Relative Orbit Start: the orbit number within a cycle. Every time a cycle starts, the relative orbit number is reset to zero. Accepted entries are: from 1 to 385

2.3 Search results list

The Search results list provides all the products matching the submitted search query (Fig. 5). Each result consists of:

- 64x64 thumbnail (when available),
- product name,
- product URI,
- Mission name, instrument name, sensing date, file size.

For every product in the results list, the following options are displayed (\bigcirc \Leftrightarrow \bigcirc \bigcirc \bigcirc \bigstar \bigstar):

- Select product,
- Zoom to product,
- View product details,
- Add product to Cart,
- Download product.



Figure 5: Product Options from search results and footprints



The footprints of the products matching the search query are displayed on the map. Each footprint gets highlighted when hovering on the corresponding product in the search results list. Red footprints show Sentinel-3 SLSTR products, blue footprints show Sentinel-3 OLCI products and purple footprints show Sentinel-3 SRAL products.

NOTE: the map tool shows only the footprints of the products in the current search list page. To widen results you can change the page size (25, 50, 75, 100, 125, and 150).

In case of searches that are recurrently performed, it is possible to save search's parameters clicking on the \mathbf{B} icon in order to save the search in the user profile and use them again the next times.

2.3.1 Product Details

When clicking on the "View Product Details" icon in one of the results of the list, a window is open showing the product preview with the following elements (Fig. 6):

- Product name,
- Product URL,
- Footprint,
- Quicklooks,
- Attributes(Summary, Product, Platform, Instrument),
- Inspection(Annotation, Measurement, Preview, Auxiliary files, manifest.xml file).

From this window, it is possible to download the entire zipped file but also the single elements of the zipped file, from the Inspector section (please, consider that when downloading a subset of the entire bulk of bands, it is always necessary to download, together with the .nc files, the xfdumanifest xml file and the auxiliary files).



Page 24 of 57



2.4 User Profile Panel

It is possible to access the User Profile Panel clicking on the 'User Profile' icon (2) on the top right of the web page (Fig. 7).



Figure 7: User Profile settings

From the User Profile Panel it is possible to check user's allowed functions (in the case of Figure 7: Search function and Download function), the Download Cart and the Saved Searches.

This is how Download Cart appears (Fig. 8):



From the Download Cart it is possible to display the entire list of products that the user added to the cart to be downloaded. For every product, the following options are available \circ - \pm :

Page 25 of 57



- Check product details,
- Remove it from the list,
- Download the file.

It is also possible to bulk download all of the products with the "Download Cart" button. In this case, a .meta4 file is saved on user's machine. Such file contains all the metalinks of the products in the cart. A download manager is required to open such type of file (a list can be found <u>here</u>), see example in the Paragraph "Batch products download from CODA and CODArep GUI".

To return to the map viewer, just click on "Map Icon" at the upper right corner. It is possible to delete the products from the cart as well, by clicking "Clear Cart".

<u>Please note</u>: due to an anomaly in the generation of the .meta4 file, no more than 100 products can be inserted in the file, even if the cart contains more.

2.5 Batch download from CODA and CODArep GUI

Once the user has selected the products to download and added them to his profile cart, it is possible to perform a batch download of all the products at once.

Download the CODA cart using the "Download Cart" button, a file with .meta4 extension shall be saved, containing the metalinks to all the products saved in the cart.

In order to download the whole list of products in batch the user can use a download manager, for example aria2.

2.5.1 Download the cart using a Download Manager (aria2)

aria2 is a is a download manager, that can be used to download the Sentinel-3 files in the list exported from CODA or CODArep (Fig 9).

Download and install aria2 from https://aria2.github.io/

To run the software, from cmd type the following command line:

```
aria2c --http-user='<username>' --http-passwd='<password>' --
check-certificate=false --metalink-file='<your path to
products.meta4>' --out=<path to directory where to store
downloaded files>
```



C:\Users\d_aria2chttp-user="downloads" a2\products.meta4"dir="C:\aria2\aria_downloads"
02/05 15:44:11 [NOTICE] Downloading 2 item(s)
02/05 15:44:11 [NOTICE] CUID#7 - Redirecting to https://coda.eumetsat.int/odata/v1/Products('9b37592f-2dcf-42ce- 4a7ee6335')/\$value
02/05 15:44:11 [NOTICE] CUID#8 - Redirecting to https://coda.eumetsat.int/odata/v1/Products('c31dbbdc-f&e9-493b- 48e8732d4')/\$value *** Download Progress Summary as of Mon Feb 05 15:45:11 2018 ***
[#f421d0 352MiB/650MiB(54%) CN:1 DL:5.4MiB ETA:555] FILE: C:/aria2/aria_downloads/S3A_OL_1_EFR20180131T073847_20180131T074147_20180131T094144_0179_027_206_2340_ _002.zip
[#3daaa6 319MiB/611MiB(52%) CN:1 DL:5.0MiB ETA:57s] FILE: C:/aria2/aria_downloads/S3A_OL_1_EFR20180131T074147_20180131T074447_20180131T094200_0179_027_206_2520_ _002.zip
[DL:10MiB][#f421d0 434MiB/650MiB(66%)][#3daaa6 399MiB/611MiB(65%)]

Figure 9: download CODA and CODArep carts with aria2 utility



3 APIS AND BATCH SCRIPTING

CODA and CODArep Web Services <u>allowsexpose</u> two dedicated Application Program Interfaces (API) for browsing and accessing the EO data stored in the rolling archive. The APIs are:

• **Open Data Protocol (OData)**: a data access protocol built on core protocols like HTTP and commonly accepted methodologies like REST that can be handled by a large set of client tools as simple as common web browsers, download-managers or computer programs such as cURL or Wget. The OData protocol can be used to build URI that the user can give in input to scripts in order to perform batch data download.

• **Open Search (Solr)**: a set of technologies that allow publishing of search results in a standard and accessible format. OpenSearch is RESTful technology and complementary to the OData. In fact, OpenSearch can be used to complementary serve as the query aspect of OData, which provides a way to access identified or located results and download them using OData.

3.1 OPEN DATA PROTOCOL (ODATA)

The Open Data Protocol (OData) enables the creation of REST-based data services, which allow resources, identified using Uniform Resource Identifiers (URIs) and defined in a data model, to be published and consumed by Web clients using simple HTTP messages.

The OData protocol provides easy access to the CODA and CODArep Web Service and can be used for building URI for performing search queries and product downloads offering to the users the capability to remotely run scripts in batch mode.

3.1.1 URI Components

A URI used by an OData service has up to three significant parts: the Service Root URI, the Resource Path and the Query Options.

- > the Service Root URI identifies the root of the OData service.
- the Resource Path identifies the resource to be interacted with. The resource path enables any aspect of the data model (CODA Web Service Products, CODA Web Service Collections, etc.) exposed by the OData service.
- > the system Query Options part refines the results.

Example of and OData URI exposed by the CODA and CODArep Web Services broken down into its component parts:

https:/	/coda.eumetsat.int/ odat	a/v1/Products?\$ski	p= 10& \$top= 50& \$for	mat=xml
	 ODATA service root URI	 resource path	 query options	
https:/	/codarep.eumetsat.int/o	data/v1/Products?\$	skip= 10& \$top= 50& \$	format=xml
	ODATA service root URI	resource path	query options	

Page 28 of 57



OData Service Root URI for CODA and CODArep Web Services: <u>https://coda.eumetsat.int/odata/v1</u> <u>https://codarep.eumetsat.int/odata/v1</u>

A generic entity type for CODA and CODArep Web Service **Resource Paths is** /Products, the related individual entity is /Products(`Id').

3.1.2 Service Metadata Document

In order to help clients discover the OData services, the CODA and CODArep Web Services OData Service Metadata Document exposes the Entity Data Model of the service including among others, the Entities and Properties that can be queried. This document can be queried as with the following URL:

https://coda.eumetsat.int/odata/v1/\$metadata https://codarep.eumetsat.int/odata/v1/\$metadata

The following example is extracted from the complete Service Metadata Document exposed by CODA Web Service OData service. Some sections or XML Namespace declarations have been removed for brevity:

```
<edmx:Edmx Version="1.0">
<edmx:DataServices>
<Namespace="CODA">
<EntityType Name="Node" m:HasStream="true">
<Key>...</Key>
<Property Name="Id" Type="Edm.String" Nullable="false"/>
                                                                     Type="Edm.String'
<Property
                                 Name="Name"
m:FC_TargetPath="SyndicationTitle"/>
<Property Name="ContentType" Type="Edm.String"/>
<property Name="ContentLength" Type="Edm.Int64"/>
<Property Name="ChildrenNumber" Type="Edm.Int64"/></property Name="ChildrenNumber" Type="Edm.Int64"/>
<Property Name="Value" Type="Edm.String"/>
</EntityType>
<EntityType Name="Product" m:HasStream="true">
<Kev>
<PropertyRef Name="Id"/>
</Key>
<Property Name="Id" Type="Edm.String" Nullable="false"/>
<Property
                                                       Name="Name" Type="Edm.String"
m:FC TargetPath="SyndicationTitle"/>
<property Name="ContentType" Type="Edm.String"/><Property Name="ContentLength" Type="Edm.Int64"/>
<Property Name="ChildrenNumber" Type="Edm.Int64"/>
<Property Name="Value" Type="Edm.String"/>
</EntityType>
<EntityType Name="Collection">
<Key>
<PropertyRef Name="Name"/>
</Key>
```

Page 29 of 57



3.1.3 Querying Products in the Data Hub archive

/Products entity

The OData URI addressing the resource /Products provides the list of entries of the individual entity/Products('Id') corresponding to the data files stored in the CODA Web Service archive.

Each entry includes: the data file name, the Id corresponding to the Universally Unique Identifier (UUID), the download URI "Products('Id')/\$value", the URIs of the navigation properties "/Nodes" and "/Attributes" and the properties of the individual entity.

The response is returned as Atom feed element.

Syntax is: <ServiceRootUri>/Products

Examples: https://coda.eumetsat.int/odata/v1/Products https://codarep.eumetsat.int/odata/v1/Products

The /Products OData URI provides, by default, a list of 50 entries sorted according to the property IngestionDate in descending order (i.e. from the latest to the earliest).

The properties available for each individual entity of / Products are listed in the table below:



EUM/OPS-SEN3/MAN/16/880763 v2B Draft, 18 October 2018

CODA user manual

/Productsproperties	Description	
Id	UUID standard unique identifier used in software construction to uniquely identify information without significant central coordination. A UUID is a 16-octet (128-bit) number. In its canonical form, a UUID is represented by 32 lowercase hexadecimal digits, displayed in five groups separated by hyphens, in the form 8-4-4- 12 for a total of 36 characters (32 alphanumeric characters and four hyphens.	
Name	Data file name.	
Contentype	Mime type.	
ContentLenght	Actual size in bytes (B) of the downloadable file (e.g. of the Zip file)	
ChildrenNumber	Number of children nodes.	
Value	Value of the Node (if present).	
CreationDate	Publication date and time of the data file (time at which the file becomes visible to the user). Time is in UTC.	
IngestionDate	Ingestion date and time of the data file in the Data Hub system (internal cataloguing time). Time is in UTC.	
EvictionDate	Date when the data file will be removed from the catalogue (it is null if the eviction policy is not applied). Time is in UTC.	
ContentDate (Start and End)	In case of EO satellite data, It corresponds to the sensing date in UTC of the observation performed by the satellite payload, also known as on-board acquisition time	
Checksum (Algorithm and Value)	Unique value for supporting download integrity check	
ContentGeometry	Footprint polygon coordinates in GML format	

Table 11: Products properties

The response of a single entity is returned as Atom entry element.

Syntax <ServiceRootUri>/Products('Id')

Example on CODA and CODArep:

- https://coda.eumetsat.int/odata/v1/Products('e9b1494f-a2d1-496f-8f79-53478bd07ec3')
- https://codarep.eumetsat.int/odata/v1/Products('a26d77e4-481e-4f04-8270-e9e454acbf94')

Page 31 of 57



The requests return an individual entity (for every request) of type /Product by the given UUID 'e9b1494f-a2d1-496f-8f79-53478bd07ec3' and the UUID 'a26d77e4-481e-4f04-8270-e9e454acbf94' (Fig. 10, the resulting xml file for the first request).



Figure 10: Product description

3.2 OData System Query Options

OData supports various kinds of query options for querying data. System query options are query string parameters that control the amount and order of the data returned for the resource identified by the URL.

The names of all system query options are prefixed with a dollar character, (\$).

A query string starts with a question mark (?), and the query options are separated by an ampersand (&). The asterisk (*) is used to specify all values. Each query option can be set on a particular value with (=).

In the table below are listed all the query option supported by the current O-Data version:



Option	Description	
\$format	Specifies the HTTP response format of the record e.g. XML or JSON	
\$filter	Specifies an expression or function that must evaluate to true for a record to be returned in the collection	
\$orderby	Determines what values are used to order a collection of records	
\$select	Specifies a subset of properties to return	
\$skip	Sets the number of records to skip before it retrieves records in a collection	
\$top	Determines the maximum number of records to return	
\$count	Allows clients to request a count of the matching resources identified by the Resource Path section of the URI	
\$inlinecount	Specifies that the response to the request includes a count of the number of the matching resources	
\$expand	Specifies the related resources to be included in line with the retrieved resources	

Table 12: OData query options

The query options follow these rules:

- Resource paths identifying generic entities allow \$filter, \$count, \$inlinecount, \$orderby, \$skip, and \$top₃
- Resource paths identifying a single entity or generic entities, allow \$expand and \$select_{\tau}
- Resource paths ending in /\$count allow \$filter;
- Resource paths not ending in /\$count allow \$format.___

\$format

When users query CODA or CODArep via the URI, the request is sent to the CODA or CODArep OData service. This service provides the responses. The default response format is Atom [RFC 4287], an XML-based document format that describes Collections of related information known as "feeds".

The responses containing a single Product entity differ from those containing a collection of Product entities. Generally speaking, the single Product entity is returned as a bare Atom entry element, while for a collection the same Atom entries denoting the Product entities are wrapped into an Atom feed element.

The response format can however be controlled from the requests through the *\$format* query option introduced above. An OData URI with a *\$format* query option specifies that a response to the request must use the media type specified by the query option. Valid values for the *\$format* query string option are listed in the following table.



\$format value	Response Media Type
atom	application/atom+xml
xml	application/xml
json	application/json
application/metalink4%2Bxml	application/metalink4+xml
text/csv	text/csv NON SUPPORTED

Table 13: OData format values

\$filter

A URI with a *filter* system query option identifies a subset of the Entities identified by the Resource Path section of the URI.

The subset is determined by selecting only the Entities that satisfy the predicate expression specified by the query option. OData supports a set of basic predicates and built-in functions for *\$filter*, including operators. They are described in the following Paragraphs.

Comparison operators

The OData API accepts the following comparison operators:

Symbol	Comparison Operator	Meaning
<	lt	Lower than
≤	le	Lower or equal than
>	gt	Greater than
≥	ge	Greater or equal than
=	eq	Equal
¥	ne	Not Equal

Table 14: OData comparison operators

Date Built-in functions

A set of functions is defined to use the \$filter system query option with temporal criteria. In particular, for the following metadata that are expressed in the UTC format:

- IngestionDate
- CreationDate
- ContentDate/Start and ContentDate/End

Page 34 of 57



The filter can be performed using the following "Date Functions" expressing temporal criteria:

Function	Description
year	The year function returns the year component of a date parameter value, evaluated in the time zone of the parameter value
month	The month function returns the month component of a date parameter value, evaluated in the time zone of the parameter value
day	The day function returns the day component of a date parameter value, evaluated in the time zone of the parameter value
hour	The hour function returns the hour component of a date parameter value, evaluated in the time zone of the parameter value
minute	The minute function returns the minute component of a date parameter value, evaluated in the time zone of the parameter value
second	The second function returns the month component of a date parameter value, evaluated in the time zone of the parameter value
datetime	The datetime function returns the date used as reference date to filter products

Table 15: OData date built-in functions

Examples for CODA and CODArep

These URIs select products published in CODA (or CODArep) during the hours from 00:00 to 12:00 UTC:

- https://coda.eumetsat.int/odata/v1/Products?\$filter=hour(Ing Field Code Changed estionDate) lt 12 Field Code Changed
 - https://codarep.eumetsat.int/odata/v1/Products?\$filter=hour(IngestionDate) lt 12

These URIs select products published in CODA (or CODArep) on December 2017:

- https://coda.eumetsat.int/odata/v1/Products?\$filter=year(Ing estionDate) eq 2017 and month(IngestionDate) eq 12
- https://codarep.eumetsat.int/odata/v1/Products?\$filter=year(IngestionDate) eq 2017 and month(IngestionDate) eq 12

These URIs select products with sensing date before 2017:

- https://coda.eumetsat.int/odata/v1/Products?\$filter=year(Con tentDate/End) le 2017
- https://codarep.eumetsat.int/odata/v1/Products?\$filter=year(ContentDate/End) le 2017

All the time-based criteria can be used with the DateTime filter which is recommended in terms of response time performances.

For example, these URIs select products that have been published in CODA or CODArep the 15 November 2017

Page 35 of 57

Field Code Changed

Field Code Changed



- https://coda.eumetsat.int/odata/v1/Products?\$filter=Ingestio
 nDate eq datetime`2017-11-15T00:00:00.000`
- https://codarep.eumetsat.int/odata/v1/Products?\$filter=Inges tionDate eq datetime'2017-11-15T00:00:00.000'

This URI selects the products that have been published in the CODA Web Service after 22^{nd} Feb 2017:

- https://coda.eumetsat.int/odata/v1/Products?\$filter=Ingestio nDate gt datetime'2017-02-22T00:00:00'

String Built-in functions

The products file name can be used for filtering the products. It shall be noticed that this query criteria is not based on the Medatata indexed from the products content but the search criteria is based on the match of predefined strings in the file name.

The filter can be performed using the following "String Functions":

Function	Description
substringof	The substring of function returns records with names containing a particular string at any position
endswith	The endswith function returns true if the first parameter string value ends with the second parameter string value, otherwise it returns false
startwith	The starts with function returns true if the first parameter string value starts with the second parameter string value, otherwise it returns false

Table 16: OData string built-in function

The syntax is: <ServiceRootUri> /Products?\$select=Id&filter=substringof('<string>', Name)

Examples:

This URI retrieves the product

"S3A_OL_1_ERR____20180206T095756_20180206T104142_20180207T143300_2626_027_29 3_____MAR_O_NT_002" from CODA archive

- https://coda.eumetsat.int/odata/vl/Products?\$filter=Name eq 'S3A_OL_1_ERR____20180206T095756_20180206T104142_20180207T14 3300_2626_027_293_____MAR_O_NT_002'

This URI retrieves products having 'OL_1_EFR' in the name (i.e. all the OLCI L1 Full Resolution Products) from CODArep:

- https://codarep.eumetsat.int/odata/v1/Products?\$filter=subst ringof('OL_1_EFR',Name)

Page 36 of 57



This URI selects the products in CODA with a filename that ends with '002': https://coda.eumetsat.int/odata/v1/Products?\$filter=endswith(Name

Please note: the "String Functions" are often used to filter products by the Sensing Time (start and stop).

For example, this URI lists the CODA products having in the file name the string 20180210T081356, i.e. products with a predefined sensing time start (in this case 2018-02-10, T081356)

https://coda.eumetsat.int/odata/v1/Products?\$filter=substringof('
20180210T081356',Name)

This URI lists the CODArep products having in the file name the string 20170909, i.e. products with a predefined sensing time start day (in this case 2017-09-09) https://codarep.eumetsat.int/odata/v1/Products?\$filter=substringo

f('_20170909_', Name)

\$orderby

The *sorderby* system query option allows clients to request resources in either ascending order using asc or descending order using desc. If asc or desc not specified, then the resources will be ordered in ascending order.

Examples:

This OData URI lists the set of Products in CODArep entity ordered by IngestionDate arranged in descending order.

https://codarep.eumetsat.int/odata/v1/Products?\$orderby=Ingestion
Date desc

This search skips the first 100 records of the products published on the CODA Web Service and then returns the next 15:

https://coda.eumetsat.int/odata/v1/Products?\$orderby=IngestionDat
e desc&\$top=15&\$skip=100

\$select

The *sselect* system query option allows the clients to request a limited set of properties for each entity. The value of a *sselect* is a comma-separated list of selection clauses.

Each selection clause may be a Property name, Navigation Property name, or the "*" character. If the *\$select* query option had listed a Property that identified a Complex Type, then all Properties defined on the Complex Type must be returned.

Examples:

Page 37 of 57



This URI queries the ID property of all the Products in CODA ordered by IngestionDate arranged in descending order.

https://coda.eumetsat.int/odata/v1/Products?\$select=Id

This URI queries the Name and the Creation date properties of all the Products in CODArep. https://codarep.eumetsat.int/odata/v1/Products?\$select=Name,Creat ionDate

\$skip and \$top

A data service URI with a \$top system query option identifies a subset of the Entities in an EntitySet identified by the Resource Path section of the URI. This subset is formed by selecting only the first M items of the set.

Syntax is:

<ServiceRootUri> /Products?\$skip='N'&\$top='M'

Where:

- **?\$skip='N'** is the number of records to skip before it retrieves records in a collection,
- **\$top='M'** is the maximum number of records to return.

NOTE: \$top accepts as maximum value 100. For higher values it will result in the following error: "*Product page size exceeds the authorized size (100)*".

A data service URI with a *\$skip* system query option identifies a subset of the Entities in an EntitySet identified by the Resource Path section of the URI.

That subset is defined by seeking N Entities into the EntitySet and selecting only the remaining Entities (starting with Entity N+1). N is an integer greater than or equal to zero specified by this query option.

Examples:

This OData URI allows to list 50 products on CODA archive skipping the first 10. https://coda.eumetsat.int/odata/v1/Products?\$skip=10&\$top=50

This OData URI allows to list 30 products on CODArep archive skipping the first 10. https://codarep.eumetsat.int/odata/v1/Products?\$skip=10&\$top=30

\$count and \$inlinecount

The \$count system query option allows clients to request a count of the matching resources included with the resources in the response: the number of the matching resources is returned as result.

Page 38 of 57



The *\$inlinecount* system query option specifies that the count of the number of entities matching the queries has to be included in the response. The set of valid values for the *\$inlinecount* query option are:

- allpages
- none

This last values is equivalent to an URI that does not include \$inlinecount query string parameter.

\$count and \$inlinecount query options are useful to know the number of entities which are identified by the resource path section of the URI after having applied some filters.

Examples:

This OData URI returns the number of OLCI Level 1 products available on CODArep: https://codarep.eumetsat.int/odata/v1/Products/\$count?\$filter=sta rtswith(Name,'S3A_OL_1')

This OData URI identifies the first 5 Product entities of Sentinel 3 and includes a count of the total number of SLSTR Level 1 products on CODA:

https://coda.eumetsat.int/odata/v1/Products?\$inlinecount=allpages
&\$top=5&\$filter=startswith(Name,'S3A_SL_1')

\$expand

The \$expand system query option specifies the related resources to be included in line with retrieved resources.

The syntax of \$expand is a comma-separated list of Navigation Properties. Additionally each Navigation Property can be followed by a forward slash and another Navigation Property to enable identifying a multi-level relationship.

Examples:

This OData URI shows the product properties including the nested ones, in this case the first product node.

https://coda.eumetsat.int/odata/v1/Products?('e9b1494f-a2d1-496f-8f79-53478bd07ec3')?\$expand=Nodes

3.3 Download

OData URL can be used for download full products as well as partial products by using the combination of the Nodes and Attributes indexed for the products and adding \$value at the end of the URI.

Page 39 of 57



Download full product from its UUID To download a full products the syntax is: <ServiceRootUri>/Products('Id')/\$value

Example for CODA: https://coda.eumetsat.int/odata/v1/Products('e9b1494f-a2d1-496f-8f79-53478bd07ec3')/\$value

Example for CODArep: https://codarep.eumetsat.int/odata/v1/Products('2146b12c-fd9b-4386-b271-b401ed1294df')/\$value

3.4 Discover Product Nodes

Products from different processing levels (Level-1 and Level-2) are disseminated in SENTINEL-SAFE format.

The data delivered is packaged as a file structure containing a manifest file in XML format (xfdumanifest.xml) listing general product metadata and subfolders for measurement data, annotations, previews and support files.

The SENTINEL-SAFE format wraps a folder containing image data in a binary data format and product metadata in XML. This flexibility allows the exploration of parts of the products without download the complete product.

In this section we will provide examples to how explore products nodes.

To return the XML file including the list of the first level nodes, the syntax is: <ServiceRootUri>/Products('<Id>')/Nodes

Generally the first node is the one whose name is [Product_Name.SEN3] and the URI to visualize the XML schema of this node is the following:

<ServiceRootUri>/Products('<Id>')/Nodes('[PRODUCT_NAME.SEN3]')/No
des

Example: https://coda.eumetsat.int/odata/v1/Products('e9b1494f-a2d1-496f-<u>8f79-</u> 53478bd07ec3')/Nodes('S3A_OL_2_WRR___20180206T095756_20180206T10 4142_20180207T143828_2626_027_293___MAR_0_NT_002.SEN3')/Nodes

Example of the nodes of a level-1 Sentinel Product are:

- Manifest.safe
- Annotation
- Measurement
- Preview
- Support



3.5 Download manifest file knowing the product UUID

The syntax for downloading a manifest file knowing the UUID (Id) is: <ServiceRootUri>/Products('Id')/Nodes('Filename')/Nodes('manifest .safe')/\$value

Example:

https://coda.eumetsat.int/odata/v1/Products('e9b1494f-a2d1-496f-8f79-53478bd07ec3')/Nodes('S3A_OL_2_WRR___20180206T095756_20180206T10 4142_20180207T143828_2626_027_293___MAR_O_NT_002.SEN3')/Nodes('xfdumanifest.xml')/\$value

3.6 Download of quick-look file knowing the product UUID

Quicklooks can be extracted and downloaded knowing the 'Id' of the product. The syntax is: <ServiceRootUri>/Products('Id')/Products('Quicklook')/\$value

Example:

Quicklook from a product in CODA

https://coda.eumetsat.int/odata/v1/Products('e9b1494f-a2d1-496f-8f79-53478bd07ec3')/Products('Quicklook')/\$value

Quicklook from a product in CODArep

https://codarep.eumetsat.int/odata/v1/Products('d7626de7-5f35-42c6-a243-0e6e762d3e7b')/Products('Quicklook')/\$value

3.7 Verifying Download integrity using the MD5 checksum

This can help to reveal if the download was incomplete. Each product published on CODA and CODArep provides an MD5 checksum of the downloadable ZIP file.

The Message Digit of the file can be discovered using the following OData query: <ServiceRootUri>/Products('Id')/Checksum/Value/\$value

Examples:

This URI queries CODA archive to get the MD5 checksum for the product: https://coda.eumetsat.int/odata/v1/Products('82963bc6-8dce-455ba8e0-65f421835a13')/Checksum/Value/\$value The result is the following MD5: BF1CAF087FE6B14A42BDCB9C34AF559D

Same to get MD5 checksum for files in CODArep:

Page 41 of 57



https://codarep.eumetsat.int/odata/v1/Products('4d4052dc-8219-4adb-a595-a79c76a3b47c')/Checksum/Value/\$value

The integrity of the downloaded product can be checked by comparing the MD5 value of the product at the source with the MD5 of the product located on the users' system. If they are identical the download has been successful.

Different Operating Systems require different methods for verifying the checksum of the downloaded products. Find below a list of the most common ones:

Linux	Open a terminal and type: md5sum path/filename	
OS X	Open a terminal and type: md5 path/filename	
Windows	The File Checksum Integrity Verifier (FCIV) utility does not come pre- installed on windows and needs to be downloaded. Instructions can be found <u>here</u> . After having downloaded the FCIV, open a command prompt and type: FCIV -md5 -sha1 path\filename	

OpenSSL

This method can be done on any system with OpenSSL installed (i.e. Windows, Linux and MAC). Open a command prompt or terminal and type:

openssl md5 path/filename



4 OPEN SEARCH

OpenSearch (Solr) is a set of technologies that allow publishing of search results in a standard and accessible format. OpenSearch is RESTful technology and complementary to the OData. In fact, OpenSearch can be used to complementary serve as the query aspect of OData, which provides a way to access identified or located results and download them. The CODA Web Service implementation uses the Apache Solr search engine.

4.1 URI Components

The following Open Search URI addresses the resource $/ {\tt Products}$ exposed by the Open Search Service.

<CODAhostname>:<port>/<path>/search?q=<query>[&start=<page_start_ value>][&rows=<page_size>]

where:

- <CODA_hostname>:<port>/<path> is the Service Root (in our case is https://coda.eumetsat.int/ or https://codarep.eumetsat.int/)
- /search?q=<query>[&start=<page_start_value>][&rows=<page_siz
 e>] is the Open Search query

4.2 Using Open Search to discover products

4.2.1 Discover the list of the products stored in the archive

The syntax is: <CODA_hostname>:<port>/search?q=*

Examples:

https://coda.eumetsat.int/search?q=*
https://codarep.eumetsat.int/search?q=*

The above URIs return an XML file including the list of the nodes of every products stored in the CODA (or CODArep) Web Service archives.

4.2.2 Paging results

A query can be completed with the pagination options. The available options are:

- rows=<page_size>, where page_size is the number of results listed per page
- start=<page_start_value>, where page_start_value identifies from which
 result position starts the pagination.

If nothing is specified by default page_size and page_start_value are set to rows=10 and start=0.

Page 43 of 57



So the complete search will be: <CODA_hostname>:<port>/<path>/search?q=*&rows= <N> &start= <N>

Note: The maximum number of rows to be returned in a single query is set to 100. Requests for more than the maximum supported number of rows will result in an error (http 400).

Results sets over the maximum can be obtained through paging of from different start values.

- Page 1: https://coda.eumetsat.int/search?start=0&rows=100&q=*
- Page 2: https://coda.eumetsat.int/search?start=100&rows=100&q=*
- Page 3: https://coda.eumetsat.int/search?start=200&rows=100&q=*
- Page 1 for CODArep:

https://codarep.eumetsat.int/search?start=0&rows=100&q=*

4.2.3 Sorting results

The list of results can be sorted using the \$orderby option with the following possible values:

- beginposition asc: sorts results by sensing date arranged in ascending order
- beginposition desc: sorts results by sensing date arranged in descending order
- ingestiondate asc: sorts results by ingestion date arranged in ascending order
- ingestiondate desc: sorts results by ingestion date arranged in descending order

Examples:

https://coda.eumetsat.int/search?q=*&orderby=beginpositionasc asc https://codarep.eumetsat.int/search?q=*&orderby=beginpositionasc asc

4.3 Discover the products over a predefined Area Of Interest (AOI): Geographical Search

It is possible to search products on the basis of a geographical area of interest, e.g. get the list of
products over a geographic area delimitated by the polygon having vertices:
<CODA_hostname>:<port>/<path>/search?q=footprint:"Intersects(
<geographic type>)"

The **<geographic type>** value can be expressed as a polygon or as a point according to the syntaxes described below.

4.3.1 Polygon

<geographic type> =POLYGON((PlLon PlLat, P2Lon P2Lat, ..., PnLon
PnLat, PlLon PlLat))

Page 44 of 57



where PlLon and PlLat are the Longitude and Latitude coordinates of the first point of the polygon in decimal degrees (DDD) format (e.g. 2.17403, 41.40338) and so on.

The coordinates of the last point of the polygon must coincide with the coordinates of the first point of the polygon.

The polygon describing the geographical area can have a maximum of 200 points that must be within an area described by 10 degrees of latitude and 10 degrees of longitude. Examples

The following URIs perform a search in CODA and CODArep archives for an AOI which is a bounding box around the Mediterranean Sea:

https://coda.eumetsat.int/search?q=footprint:"Intersects(POLYGON((-4.53 29.85, 26.75 29.85, 26.75 46.80,-4.53 46.80,-4.53 29.85)))" https://codarep.eumetsat.int/search?q=footprint:"Intersects(POLYG ON((-4.53 29.85, 26.75 29.85, 26.75 46.80,-4.53 46.80,-4.53 29.85)))"

4.3.2 Point

<geographic type>= Lat, Lon

where the Latitude (Lat) and Longitude (Lon) values are expressed in decimal degrees (DDD) format (e.g. 41.40338, 2.17403).

Examples: https://coda.eumetsat.int/search?q=footprint:"Intersects(41.9000, 12.5000)" https://codarep.eumetsat.int/search?q=footprint:"Intersects(44.40 00, 8.9500)"

4.4 Open Search queries examples (combined with full text search)

The query in the open search URI will follow the same syntax used in the full text search. The syntax is the following: <CODA_hostname>:<port>/<path>/search?q=

It can be combined with different query options. Here below some examples, for searches both in CODA and CODArep archives.

Page 45 of 57



Example **Open Search** https://coda.eumetsat.int/search?q=OL_ Searches every product with 1_EFR OL_1_EFR___ product type or products containing the string "OL_1_EFR____" in the metadata. https://coda.eumetsat.int/search?q=(platformname:Senti Searches every product with SLSTR instrument and SM product nel-3 AND instrumentshortname:SLSTR AND productlevel:L1) Level 1. Search every products sensedin the https://coda.eumetsat.int/search?q= beginPosition: [NOW-1DAYS TO NOW] last day Search every products ingested in https://coda.eumetsat.int/search?q=ing estionDate: [NOW-30DAYS TO NOW] the last month Search every products sensed in the https://coda.eumetsat.int/search?q=beg last 5 hours. inPosition: [NOW-5HOUR TO NOW] https://coda.eumetsat.int/search?q=beg Search every products having sensing in the last three months inPosition: [NOW-3MONTHS TO NOW] AND endPosition: [NOW-3MONTHS TO NOW] Search every products delimitated https://coda.eumetsat.int/search?q=foo tprint:"Intersects(POLYGON((-4.53)) by the polygon vertices: -4.53 29.85, 26.75 29.85, 26.75 46.80,-29.85, 26.75 29.85, 26.75 46.80, -4.53 46.80,-4.53 29.85)))" 4.53 46.80, -4.53 29.85

Table 17: Solr geographical search examples

Other details and further examples are illustrated in the Full Text Search section.



5 BATCH SCRIPTING

The above OData and OpenSearch URIs can be combined to create complex queries to be executed in non-interactive scripts using programs like <u>cURL</u> and <u>Wget</u>.

5.1 Query via Curl

Using cURL it is possible to create a script to login to the CODA or CODArep Web Services via the following command line:

curl -u {USERNAME}:{PASSWORD} <URI_QUERY>

where:

<URI_QUERY>: valid OData URI or OpenSearch URI.

-u <username>:<password>: EO Portal username and password

5.2 Query via Wget

It is possible to use the wget command to create batch scripts: wget --no-check-certificate --user={USERNAME} -password={PASSWORD} --output-document={FILE} "<URI QUERY>"

Where {USERNAME} is the valid account username, {PASSWORD} is the corresponding authentication password value and {FILE} is the name of the file where to print the output of the query. If ?-? is used as {FILE}, documents will be printed to standard output.

Please, note that the "-" symbol corresponds to the Hyphen-Minus (U+002D). Pay attention to not confuse it with other symbols or you will get an error.

The following example shows how to make an OpenSearch query using Wget. The query searches for all the products in the CODA archive. The first 25 results are printed in a file named query_results.txt:

wget --no-check-certificate --user={USERNAME}
password={PASSWORD} --output-document=query_results.txt

The following example shows how to make an OpenSearch query using Wget for searching products filtered by product type and ingestion date:

wget --no-check-certificate --user={USERNAME} -password={PASSWORD} --output-document=query_results.txt "https:// coda.eumetsat.int/search?q=ingestiondate:[NOW-1DAY TO NOW] AND producttype:SLC&rows=100&start=0&format=json"

5.3 Download via Wget

It is also possible to download the products from the Data Hub archive using Wget .

The following example shows how to download a single product, identified by its own Data Hub universally unique identifier {UUID}, using an OData URI:

Page 47 of 57



wget --no-check-certificate --user={USERNAME} -password={PASSWORD}

"https://coda.eumetsat.int/odata/v1/Products('{UUID}')/\$value"

The option --continue is very useful when downloads do not complete due to network problems. Wget will automatically try to continue the download from where it left off, and repeat this until the whole file has been retrieved.

The following example shows how to download the manifest file of a Sentinel-1 product using and Odata URI with Wget only identified by the universally unique identifier {UUID}:

```
wget --no-check-certificate --user={USERNAME} --
password={PASSWORD}
"https://coda.eumetsat.int/odata/v1/Products('{UUID}')/Nodes('{PR
ODUCT_FILENAME}')/Nodes('manifest.safe')/$value"
```

where {UUID} is the value of the universally unique identifier of the product, and {PRODUCT_FILENAME} is the filename of the product.

5.4 Scripts Examples

Please, note that, since because those scripts are provided in a shell (.sh) file, they can be used under a Unix/Linux environment.

dhusget functionality is guaranteed in Linux environment. Linux emulators can cause failures.

5.4.1 dhusget script

dhusget.sh is a simple demo script illustrating how to use OData and OpenSearch APIs to query and download the products from any CODA and CODArep Web Services. It allows:

- Search products over a pre-defined AOI
- Filter the products by ingestion time, sensing time and coordinates
- Filter the products by instrument and product type
- Save the list of results in CSV and XML files
- Download the products
- Download the manifest files only
- Perform the MD5 integrity check of the downloaded products

You can download the script from <u>here</u> It requires the installation of Wget.

- Tip 1: once the shell file is downloaded, open it with a text editor, go to the section 'Load input parameter' and change the values of USERNAME and PASSWORD with your username and password. In this way it will be not necessary to pass them as arguments when running the script.
- Tip 2: once you downloaded the script, check the permission you have on the script. To make it executable, use the following command: chmod +x dhusget.sh

Page 48 of 57



• TIP 3 (**IMPORTANT**): the dhusget.sh script, is set to poll CODA Web Service. In order to perform search and downloads of file on **CODArep** archive, add the login option '-d https://codarep.eumetsat.int/'.

-		
11	sage	٠
U	sage	٠

dhusget.sh [LOGIN OPTIONS]... [SEARCH QUERY OPTIONS]...
[SEARCH RESULT OPTIONS]...

-d	<coda codarepurl=""></coda>	URL of the CODA Web Service to be polled (default is <u>https://coda.eumetsat.int/</u> . To connect to CODArep the address is <u>https://codarep.eumetsat.int/</u>).
-u	<username></username>	EO Portal username.
-p	<password></password>	CODA Web Service password provided after registration.
Searc	ch Query options	
-i <	instrument name>	Instrument name.
-t	<time hours="" in=""></time>	Search for products ingested in the last <time hours="" in=""> (integer) from the time of execution of the script. (e.g. '-t 24' to search for products ingested in the last 24 Hours).</time>
-s <ing< td=""><td>gestion_date_FROM></td><td>Search for products ingested after the date and time specified by <ingestion_date_from>. The date format is ISO 8601:YYYY-MM-DDThh:mm:ss.cccZ (e.gs 2017-10-02T06:00:00.000Z).</ingestion_date_from></td></ing<>	gestion_date_FROM>	Search for products ingested after the date and time specified by <ingestion_date_from>. The date format is ISO 8601:YYYY-MM-DDThh:mm:ss.cccZ (e.gs 2017-10-02T06:00:00.000Z).</ingestion_date_from>
-e	<ingestion_date_to></ingestion_date_to>	Search for products ingested before the date specified by <ingestion_date_to>. The date format is ISO 8601: YYYY-MM-DDThh:mm:ss.cccZ (e.ge 2017-03-23T12:00:00.000Z).</ingestion_date_to>
-S	<sensing_date_from></sensing_date_from>	Search for products with sensing date greater than the date and time specified by <sensing_date_from>. The date format is ISO 8601: YYYY-MM-DDThh:mm:ss.cccZ (e.g S 2017-03-23T06:00:00.000Z).</sensing_date_from>
-E	<sensing_date_to></sensing_date_to>	Search for products with sensing date less than the date and time specified by <sensing_date_to>. The date format is ISO 8601: YYYY-MM-DDThh:mm:ss.cccZ (e.gE 2017-10-10T12:00:00.000Z).</sensing_date_to>
-f	<file></file>	Search for products ingested after the date and time provided through the input <file>. The file is updated at the end of the script execution with the ingestion date of the last successful downloaded product.</file>



<pre>-c <coordinates i.e.:<br="">lon1,lat1:lon2,lat2></coordinates></pre>	Coordinates of two opposite vertices of the rectangular area of interest.
-T <product type=""></product>	Product type of the product to search (for examples: OL_1_EFR).
-F <free opensearch<br="">query></free>	Free text OpenSearch query. The query must be written enclosed by single apexes ' <query>' (e.gF 'platformname:Sentinel-3 AND producttype:OLCI'). Note: the free text OpenSearch query can be combined with the other possible specified search options.</query>
Search Result options	
-l <results></results>	Maximum number of results per page [1,2,3,4,]; default value = 25
-P <page></page>	Page number $[1,2,3,4,]$; default value = 1
-q <xmlfile></xmlfile>	Write the OpenSearch query results in a specified XML file. Default file is './OSquery-result.xml'
-C <csvfile></csvfile>	Write the list of product results in a specified CSV file. Default file is './products-list.csv'
Download options	
-o <option></option>	 What to download; the possible options are: 'manifest' to download the manifest of all products returned from the search 'product' to download all products returned from the search 'all' to download both
-0 <folder></folder>	Save the Product ZIP files in a specified folder.
-N <file></file>	Write in the specified file the list of products that have failed the MD5 integrity check. By default the list is written in ./failed_MD5_check_list.txt. The format of the output file is compatible with option -r
-D	If specified, remove the products that have failed the MD5 integrity check from disk. By default products are not removed;
-r <file></file>	 Download the products listed in an input <file> written according to the following format:</file> One product per line. <space><one character=""><space><uuid><space><one character=""><space><filename>.</filename></space></one></space></uuid></space></one></space>
-L <lock folder=""></lock>	By default only one instance of dhusget can be executed at a time. This is ensured by the creation of a temporary lock folder /home/dhus/dhusget_tmp/lock which is removed a the end of each run.

Page 50 of 57



1 (14 1 4
his value doesn't override for the user
d

Table 18: dhusget usage options

Examples:

./dhusget.sh -1 100 -o product to download the latest 100 products

./dhusget.sh -l 100 -c '-4.530,29.850:26.750,46.800' to download the latest 100 products intersecting the specified coordinates

5.4.2 Odata-demo.sh script

The script OData-demo is a demo script performing the following selective actions:

- 1. List the collections
- 2. List <n> products from a specified collection
- 3. List first 10 products matching part of product name
- 4. List first 10 products matching a specific ingestion date
- 5. List first 10 products matching a specific acquisition date
- 6. List first 10 products since last <n> days, by product type and intersecting an AOI
- 7. Get product id from product name
- 8. Get polarisation from a product id
- 9. Get relative orbit from a product id
- 10. Download Manifest file from a product id
- 11. Download quick-look from a product id
- 12. Download full product from its id

You can download the script <u>here</u> NOTE: It requires the installation of xmlstarlet.

USAGE # odata-demo.sh [OPTIONS]

Page 51 of 57



	Parameter/value	Description
-h,	help	Displays a help message
-j,	json	Use json output format for OData (default is xml)
-s,	server=SERVER	Use SERVER as URL of CODA or CODArep Web Services
-u,	user=NAME	EO Portal username
-p,	password=PASSWORD	EO Portal password
-v,	verbose	Display curl command lines and results
-V,	version	Display the current script version

Table 19: Odata-demo usage options



6 OLCI ORBIT PREDICTION

It is possible to access OLCI orbit prediction files via a dedicate ftp website: ftp://ftp.eumetsat.int/pub/OPS/out/S-3_User_Data, so that users can plan observation campaign. The ftp contains a folder with the year, which contain 12 sub-folders, one for each month. In every sub-folder folder, there is a file for every day of the month. The file naming convention is: *S3A OS yyymmdd nnn* (with nnn= Day of Year).

Files are provided in KML format, which can be open using Google Earth. When displaying a KML single-day file in Google Earth, all the orbits swaths for that day are displayed in different colours, together with their satellite track. It is possible to plot the polygons of the reduced resolution swath and of the full resolution swath.

For every satellite track, there are different placeholders with the time of the satellite pass on that point (based on the reference orbit), see Figure 11.



Figure 11: orbit prediction satellite track with time pass



When the file is open in Google Earth, a folder in the Places sidebar is displayed. Every single-day file contains one sub-folder for every orbit in that day, with the naming convention *nnnnn*. *(mmm[pp])*, with: *nnnnn*: absolute orbit number *mmm*: relative orbit number *pp*: cycle number.

Inside the folder there are different elements: the satellite track, the polygons displaying the Reduced Resolution swath (ERR) and the Full Resolution swath (EFR) with the frame numbers, and the placeholders with the satellite pass times.

Page 54 of 57



7 **USEFUL LINKS**

This paragraph lists the URLs to Sentinel-3 technical documents that can be useful for users.

User Handbooks

Sentinel-3 OLCI: Sentinel-3 OLCI User Handbook Sentinel-3 SLSTR: Sentinel-3 SLSTR User Handbook Sentinel-3 SRAL: Sentinel-3 SRAL User Handbook

Available Product Types

Sentinel-3 OLCI: Sentinel-3 OLCI product types	 Field Code Changed
Sentinel-3 SLSTR: Sentinel-3 SLSTR product types	 Field Code Changed
Sentinel-3 SRAL: <u>Sentinel-3 SRAL product types</u>	 Field Code Changed

Product naming Convention: Sentinel-3 OLCI: <u>Sentinel-3 OLCI Naming Convention</u> Sentinel-3 SLSTR: <u>Sentinel-3 SLSTR Naming Convention</u> Field Code Changed Sentinel-3 SRAL: Sentinel-3 SRAL Naming Convention

Page 55 of 57



8 KNOWN ISSUES AND USEFUL TIPS FOR CODA WEB SERVICE

In this paragraph we are going to list the already known issues about CODA and CODArep Web Service, and some useful tips to improve the usability of the application, meanwhile we are working on the improvement of the service in order to give our users a better experience. Please, take note of these limitations, and consider that we are working to fix them in the next Services updates.

Known issues	what user can do for now
It is not possible to connect to CODA using EO Portal username and password.	Password must NOT contain any of the following special characters: ~!@#\$%^&*+=` \(){}[]:;"' <>,.?/
When a search is performed, the map is not centred on the resulting images footprint, but it remains the same.	Map does not automatically update the centre and the zoom based on search's results. If you want to centre the map on one of the resulting products, click on one of the results in the list, and then click on the icon "Zoom to product" (see image below).
The same button has two functionalities: Navigate On Map that allows the user to move on the base layer map Draw Region of Interest that allows the user to draw a rectangle on the map to filter the search results geographically. This button seems to have the opposite functions with respect the icon it is displaying.	Please, note that when the Navigate on Map icon is displayed it is possible to draw a rectangle on the map and that when Draw Region of Interest icon is displayed it is possible to move the map. The icon represents the functionality that will be activated when the button is clicked.
 There are two ways to add a product to Cart: Select the product you want to add to Cart then click on the icon 'Add Product to Cart' (see figure below) Control (see figure below)	If you want to access your cart with the list of products ready to be downloaded, click on 'User Profile' button and select 'Cart'. This will lead you to the list of products you chose.
In the current DHuS version, there is a known anomaly in the generation of the .meta4 file. Only 100 products of the cart are inserted in the file, even if the cart contains more.	Keep no more than 100 products in the Cart.

Formatted Table

Page 56 of 57



The map viewer is missing the zoom in and out buttons. Zoom in and out with the scroll wheel or use the pinch-zoom on the touchpad.

Table 20: CODA and CODArep known issues

Page 57 of 57