

# Package ‘CDSE’

December 5, 2023

**Type** Package

**Title** 'Copernicus Data Space Ecosystem' API Wrapper

**Version** 0.1.0

**Description** Provides interface to the 'Copernicus Data Space Ecosystem' API

<<https://dataspace.copernicus.eu/analyse/apis>>

, mainly for searching the catalog of available data from Copernicus Sentinel missions and obtaining the images for just the area of interest based on selected spectral bands. The package uses the 'Sentinel Hub' REST API interface

<<https://dataspace.copernicus.eu/analyse/apis/sentinel-hub>>

that provides access to various satellite imagery archives. It allows you to access raw satellite data, rendered images, statistical analysis, and other features.

**Depends** R (>= 3.6.0)

**Imports** geojsonsf, httr2, jsonlite, lubridate, lutz, sf, stats, terra, utils

**License** AGPL-3

**Encoding** UTF-8

**Config/build/clean-inst-doc** FALSE

**RoxygenNote** 7.2.2

**Suggests** maps, tibble

**NeedsCompilation** no

**Author** Zivan Karaman [aut, cre, cph] (<<https://orcid.org/0000-0002-8933-4589>>)

**Maintainer** Zivan Karaman <zivan.karaman@gmail.com>

**Repository** CRAN

**Date/Publication** 2023-12-05 18:00:02 UTC

## R topics documented:

GetArchiveImage . . . . .	2
GetCollections . . . . .	4
GetOAuthClient . . . . .	5

GetOAuthToken . . . . .	6
SearchCatalog . . . . .	7
UniqueCatalog . . . . .	9

<b>Index</b>	<b>10</b>
--------------	-----------

---

GetArchiveImage	<i>Get image from the archive</i>
-----------------	-----------------------------------

---

## Description

Retrieves the image for the area of interest using the parameters provided.

## Usage

```
GetArchiveImage(
  aoi,
  bbox,
  time_range,
  collection,
  script,
  file = NULL,
  format = c("image/tiff", "image/png", "image/jpeg"),
  mosaicking_order = c("mostRecent", "leastRecent", "leastCC"),
  pixels,
  resolution,
  buffer = 0,
  mask = FALSE,
  client,
  token,
  url = getOption("CDSE.process_url")
)
```

## Arguments

aoi	sf or sfc object, typically a (multi)polygon, describing the Area of Interest.
bbox	numeric vector of four elements describing the bounding box of interest. Specify with a coordinate pair on two (opposite) vertices of the bounding box rectangle. Coordinates need to be in longitude, latitude. Only one of either aoi or bbox may be specified.
time_range	scalar or vector (Date or character that can be converted to date) defining the time interval.
collection	character indicating which collection to search. Must be one of the collections returned by GetCollections.
script	a length one character string containing the evaluation script or the name of the file containing the script.

file	name of the file to save the image. If NULL, a SpatRaster object is returned. Default: NULL
format	character indicating the output file format. Must be one of "image/tiff", "image/png", or "image/jpeg". Default: "image/tiff"
mosaicking_order	character indicating the order in which tiles are overlapped from which the output result is mosaicked. Must be one of "mostRecent", "leastRecent", or "leastCC". Default: "mostRecent"
pixels	integer scalar or length-two vector indicating the request image width and height. Values must be integers between 1 and 2500.
resolution	numeric scalar or length-two vector indicating the spatial resolution of the request image in horizontal and vertical direction (in meters). Only one of the arguments "pixels" or "resolution" must be set at the same time. If the argument "pixels" or "resolution" is scalar, the same value is used for horizontal and vertical direction (width and height).
buffer	numeric, width of the buffer to retrieve the image of enlarged area. Default: 0
mask	logical indicating if the image should contain only pixels within Area of Interest. Default: FALSE
client	OAuth client object to use for authentication.
token	OAuth token character string to use for authentication. Exactly one of either client or token must be specified. It is recommended to use client.
url	character indicating the process endpoint. Default: Copernicus Data Space Ecosystem process endpoint

### Details

If aoi argument is provided, the result is returned in the same coordinate reference system.

### Value

SpatRaster object (from the package terra) of the requested image (if file is NULL), or the (invisible) name of the file created.

### Source

<https://documentation.dataspace.copernicus.eu/APIs/SentinelHub/Process.html>

### See Also

[GetCollections](#), [SearchCatalog](#)

## Examples

```
## Not run:
#EXAMPLE1
dsn <- system.file("extdata", "centralpark.geojson", package = "CDSE")
aoi <- sf::read_sf(dsn, as_tibble = FALSE)
script_file <- system.file("scripts", "NDVI_uint8.js", package = "CDSE")
day <- "2023-07-11"
ras <- GetArchiveImage(aoi = aoi, time_range = day, script = script_file,
  collection = "sentinel-2-l2a", format = "image/tiff",
  mosaicking_order = "leastCC", resolution = 10, client = OAuthClient)

## End(Not run)
```

---

GetCollections

*List available collections*

---

## Description

Retrieves the list of available imagery collections.

## Usage

```
GetCollections(as_data_frame = TRUE, url = getOption("CDSE.catalog_url"))
```

## Arguments

`as_data_frame` logical indicating if the result should be returned as data frame. Default: TRUE

`url` character indicating the STAC catalog search endpoint. Default: Copernicus Data Space Ecosystem STAC endpoint

## Details

This function doesn't require authentication.

## Value

A list or a data.frame of all available imagery collections and their attributes.

## Source

<https://documentation.dataspace.copernicus.eu/APIs/SentinelHub/Catalog.html>

## See Also

[GetArchiveImage](#), [SearchCatalog](#)

**Examples**

```
## Not run:  
#EXAMPLE1  
  GetCollections(as_data_frame = TRUE)  
  
## End(Not run)
```

---

GetOAuthClient	<i>Get OAuth client</i>
----------------	-------------------------

---

**Description**

Gets an OAuth authentication client (httr2 OAuth client object)

**Usage**

```
GetOAuthClient(id, secret, url = getOption("CDSE.auth_url"))
```

**Arguments**

id	character, user OAuth client id
secret	character, user OAuth client secret
url	character, endpoint for requesting tokens. Default: Copernicus Data Space Ecosystem OAuth endpoint

**Details**

The client can be used in queries requiring the authentication.

**Value**

httr2 OAuth client object

**Source**

<https://documentation.dataspace.copernicus.eu/APIs/SentinelHub/Overview/Authentication.html>

**See Also**

[GetAuthToken](#)

## Examples

```
## Not run:
#EXAMPLE1
id <- "... "
secret <- "... "
OAuthClient <- GetOAuthClient(id = id, secret = secret)

## End(Not run)
```

---

GetOAuthToken

*Get OAuth token*

---

## Description

Gets an OAuth authentication token (long character string)

## Usage

```
GetOAuthToken(id, secret, url = getOption("CDSE.auth_url"))
```

## Arguments

id	character, user OAuth client id
secret	character, user OAuth client secret
url	character, endpoint for requesting tokens. Default: Copernicus Data Space Ecosystem OAuth endpoint

## Details

The token can be used in queries requiring the authentication.

## Value

Long character string containing the authentication token.

## Source

<https://documentation.dataspace.copernicus.eu/APIs/SentinelHub/Overview/Authentication.html>

## See Also

[GetOAuthClient](#)

**Examples**

```
## Not run:
#EXAMPLE1
id <- "... "
secret <- "... "
token <- GetOAuthToken(id = id, secret = secret)

## End(Not run)
```

---

SearchCatalog

*Search collection for available images*


---

**Description**

Searches the specified collection for available images in the given time interval and intersecting with the bounding box or the area of interest.

**Usage**

```
SearchCatalog(
  aoi,
  bbox,
  from,
  to,
  collection,
  as_data_frame = TRUE,
  with_geometry = TRUE,
  client,
  token,
  url = getOption("CDSE.catalog_url")
)
```

**Arguments**

aoi	sf or sfc object, typically a (multi)polygon, describing the Area of Interest.
bbox	numeric vector of four elements describing the bounding box of interest. Specify with a coordinate pair on two (opposite) vertices of the bounding box rectangle. Coordinates need to be in longitude, latitude. Only one of either aoi or bbox may be specified.
from	start of the time interval to search.
to	end of the time interval to search. from and to can be either Date or character that can be converted to date by as.Date. Open interval (one side only) can be obtained by providing the NA or NULL value for the corresponding argument.

collection	character indicating which collection to search. Must be one of the collections returned by GetCollections.
as_data_frame	logical indicating if the result should be returned as data frame. Default: TRUE
with_geometry	logical indicating if the granule geometries should be included in the data.frame. Default: TRUE
client	OAuth client object to use for authentication.
token	OAuth token character string to use for authentication. Exactly one of either client or token must be specified. It is recommended to use client.
url	character indicating the STAC catalog search endpoint. Default: Copernicus Data Space Ecosystem STAC endpoint

### Details

If no images found, a NULL value is returned.

### Value

A list, data.frame or a sf object.

### Source

<https://documentation.dataspace.copernicus.eu/APIs/SentinelHub/Catalog.html>

### See Also

[GetCollections](#), [GetArchiveImage](#)

### Examples

```
## Not run:
#EXAMPLE1
dsn <- system.file("extdata", "luxembourg.geojson", package = "CDSE")
aoi <- sf::read_sf(dsn, as_tibble = FALSE)
images <- SearchCatalog(aoi = aoi, from = "2023-07-01", to = "2023-07-31",
  collection = "sentinel-2-l2a", with_geometry = TRUE, client = OAuthClient)

## End(Not run)
```



---

`UniqueCatalog`*Produce image catalog without multiple entries per date*

---

## Description

Sometimes several images could be available for the given day. It can be useful to have a list where for any given day there is just one row in the list. This unique row can be selected to represent either the least cloud coverage or the biggest coverage of the are of interest.

## Usage

```
UniqueCatalog(  
  imageCatalog,  
  by = c("areaCoverage", "tileCloudCover"),  
  keep = names(imageCatalog)  
)
```

## Arguments

<code>imageCatalog</code>	data.frame as returned by the SearchCatalog function.
<code>by</code>	character indicating which attribute is used to select the best image per date. Can be either "areaCoverage" or "tileCloudCover".
<code>keep</code>	list of columns to keep in output. Default: all columns in input.

## Details

By default, the returned data.frame has the same columns as the input catalog. User can specify a subset of columns to include in the output through the keep parameter.

## Value

data.frame with one row per date.

## See Also

[SearchCatalog](#)

## Examples

```
## Not run:  
#EXAMPLE1  
dsn <- system.file("extdata", "luxembourg.geojson", package = "CDSE")  
aoi <- sf::read_sf(dsn, as_tibble = FALSE)  
images <- SearchCatalog(aoi = aoi, from = "2023-07-01", to = "2023-07-31",  
  collection = "sentinel-2-l2a", with_geometry = TRUE, client = OAuthClient)  
best_daily <- UniqueCatalog(images, by = "areaCoverage")  
  
## End(Not run)
```

# Index

GetArchiveImage, [2](#), [4](#), [8](#)

GetCollections, [3](#), [4](#), [8](#)

GetOAuthClient, [5](#), [6](#)

GetOAuthToken, [5](#), [6](#)

SearchCatalog, [3](#), [4](#), [7](#), [9](#)

UniqueCatalog, [9](#)