**Overview**

Sentinel-2 SAFE archives are officially distributed through the ESA [Copernicus Open Access Hub](https://scihub.copernicus.eu/). The most recent products can be directly downloaded using two API protocols: [DHUS](https://scihub.copernicus.eu/dhus/) or [API Hub](https://apihub.copernicus.eu/apihub/). ordered products are made available online after more or less one day, then ZIP archives can be directly downloaded. Each user have a quota of orders which can submit, that is different among the two infrastructures (1 product / half hour using DHUS, 10 products / 12 hours using API Hub).

*The retention period of online products was recently reduced from 12 or 18 months to 30 days*, in order to increase the capabilities of the LTA ordering infrastructures (before this change, often requests could not have been submitted because of system saturation). This change, although not affecting the possibility to directly retrieve most recent images, made particularly difficult the analysis of Sentinel-2 time series (for which sen**2r** was designed): for example, to retrieve 3-years of data (2017-2020) on the sample Area Of Interest used for sen**2r** examples (which overlaps one Sentinel-2 tile and two orbits, and which can be obtained with the command system.file("extdata/vector/barbellino.geojson", package = "sen2r")) it is necessary to retrieve 435 archives, of which 433 are offline at the time this post was written; ordering them on DHUS using an automatised script (so to have the possibility to order exactly two every hour) would require 10 days (plus 3-4 additional days to have all of them made online).

In addition, there is the concrete chance that a substantial part of them would not be usable, due to a recently encountered problem (retrieval of corrupted SAFE archives from LTA)

This background made urgent the needing to retrieve Sentinel-2 archives from an alternative data source. Among the available ones, [Google Cloud Sentinel-2 bucket](https://cloud.google.com/storage/docs/public-datasets/sentinel-2) was chosen because it offers the possibility to download data for free and without any limitations. What is needed is:

1. a Google account (no paid Google Cloud plans are required);
2. Google Cloud SDK to be installed and configured.

**Installation and configuration**

The steps required to be able to automatically search and download Sentinel-2 data from Google Cloud through sen**2r**.

1. Install (or update) the R package {sen2r} version >= 1.5.0:

install.packages("sen2r")

(note for Windows users: at the time this post was written, version 1.5.0 had just been released, so the source version of the package must be explicitly installed until the binary version 1.5.0 will be available).

1. Install Google Cloud SDK following the [official instructions](https://cloud.google.com/sdk/docs/install).
2. Configure sen**2r** to use Google Cloud SDK: this can be done running the following function:

check\_gcloud()

which automatically retrieves the path of the binary gsutil (if the automatic retrieval would fail, i.e., because Google Cloud SDK was installed in a non-standard directory, arguments gsutil\_dir and full\_scan can be used – refer to the [function documentation](https://sen2r.ranghetti.info/reference/check_gcloud.html)).

Alternatively, check\_sen2r\_deps() can be used to launch a GUI which allows graphically configuring external dependencies (including Google Cloud SDK).

**Usage**

sen**2r** can be used as usual, remembering to set Google Cloud SDK as input source (Copernicus Hub remains the default choice).

**Using sen2r from the GUI**

The function sen2r() opens the sen2r GUI. In the first sheet, the section “SAFE options” was modified to include the selector “Input servers”, which allows keeping the default “ESA Hub”, replacing it with “Google Cloud” or leaving both (in which case products are retrieved from Google Cloud if available, or from Copernicus Hub otherwise). The selector is deactivated if Google Cloud SDK was not configured (or hidden if the offline mode was selected). If only Google Cloud was selected, it is not yet necessary to set SciHub credentials.

**Using function sen2r() non-interactively**

Users can also launch their processing chain in non-interactive mode as described in the [vignette](https://sen2r.ranghetti.info/articles/sen2r_cmd.html#specify-all-processing-parameters-in-the-call-to-sen2r); in this case, they can set the argument server in one of these ways:

* server = "gcloud" (search and download on Google Cloud exclusively);
* server = c("gcloud", "scihub") (search and download on Google Cloud first, and on Copernicus Hub in case products were not found on Google Cloud);
* server = c("scihub", "gcloud") (the same, but searching on SciHub first).

The [function documentation](https://sen2r.ranghetti.info/reference/sen2r.html) can be accessed for additional details.

**Using functions s2\_list() and s2\_download()**

These two functions can be used to specifically search SAFE archives and download them. In this case, the argument server must be set in the function s2\_list() in the same way seen for main function sen2r() (see also the [function documentation](https://sen2r.ranghetti.info/reference/s2_list.html)). This function returns a SAFE list with the specific Google Cloud URLs.

As an example:

example\_s2\_list\_scihub <- s2\_list(

tile = "32TNS", orbit = "065",

time\_interval = c("2021-05-01", "2021-05-15")

)

example\_s2\_list\_scihub

A named vector with 3 SAFE archives.

S2B\_MSIL2A\_20210501T101559\_N0300\_R065\_T32TNS\_20210501T135123.SAFE

"<https://apihub.copernicus.eu/apihub/odata/v1/Products('e9f01beb-8978-428c-89e6-f4c71156526b')/$value>"

S2A\_MSIL2A\_20210506T102021\_N0300\_R065\_T32TNS\_20210506T132458.SAFE

"<https://apihub.copernicus.eu/apihub/odata/v1/Products('9eb7cf57-49ab-4dc5-bb6b-79882196c7d9')/$value>"

S2B\_MSIL2A\_20210511T101559\_N0300\_R065\_T32TNS\_20210511T134528.SAFE

"<https://apihub.copernicus.eu/apihub/odata/v1/Products('1b35b71c-804c-4b9f-931c-de8e291393a4')/$value>"

The following attributes are included: mission, level, id\_tile, id\_orbit, sensing\_datetime, ingestion\_datetime, clouds, footprint, uuid, online.

By default outputs are searched on Copernicus Hub (as it can be noticed by product URLs). At the time this post was written, 2 of the 3 archives were not available online:

safe\_is\_online(example\_s2\_list\_scihub)

1 out of 3 products are online.

S2B\_MSIL2A\_20210501T101559\_N0300\_R065\_T32TNS\_20210501T135123.SAFE

FALSE

S2A\_MSIL2A\_20210506T102021\_N0300\_R065\_T32TNS\_20210506T132458.SAFE

FALSE

S2B\_MSIL2A\_20210511T101559\_N0300\_R065\_T32TNS\_20210511T134528.SAFE

TRUE

To search them on Google Cloud, the argument server = "gcloud" can be set:

example\_s2\_list\_gcloud <- s2\_list(

server = "gcloud",

tile = "32TNS", orbit = "065",

time\_interval = c("2021-05-01", "2021-05-15")

)

example\_s2\_list\_gcloud

A named vector with 3 SAFE archives.

S2B\_MSIL2A\_20210501T101559\_N0300\_R065\_T32TNS\_20210501T135123.SAFE

"gs://gcp-public-data-sentinel-2/L2/tiles/32/T/NS/S2B\_MSIL2A\_20210501T101559\_N0300\_R065\_T32TNS\_20210501T135123.SAFE/"

S2A\_MSIL2A\_20210506T102021\_N0300\_R065\_T32TNS\_20210506T132458.SAFE

"gs://gcp-public-data-sentinel-2/L2/tiles/32/T/NS/S2A\_MSIL2A\_20210506T102021\_N0300\_R065\_T32TNS\_20210506T132458.SAFE/"

S2B\_MSIL2A\_20210511T101559\_N0300\_R065\_T32TNS\_20210511T134528.SAFE

"gs://gcp-public-data-sentinel-2/L2/tiles/32/T/NS/S2B\_MSIL2A\_20210511T101559\_N0300\_R065\_T32TNS\_20210511T134528.SAFE/"

The following attributes are included: mission, level, id\_tile, id\_orbit, sensing\_datetime, ingestion\_datetime, clouds, footprint, uuid, online.

Now URLs refer to Google Cloud locations.

The “mixed” research can be performed in two ways:

example\_s2\_list\_mixed1 <- s2\_list(

server = c("gcloud","scihub"), availability = "check",

tile = "32TNS", orbit = "065",

time\_interval = c("2021-05-01", "2021-05-15")

)

example\_s2\_list\_mixed1

A named vector with 3 SAFE archives.

S2B\_MSIL2A\_20210501T101559\_N0300\_R065\_T32TNS\_20210501T135123.SAFE

"gs://gcp-public-data-sentinel-2/L2/tiles/32/T/NS/S2B\_MSIL2A\_20210501T101559\_N0300\_R065\_T32TNS\_20210501T135123.SAFE/"

S2A\_MSIL2A\_20210506T102021\_N0300\_R065\_T32TNS\_20210506T132458.SAFE

"gs://gcp-public-data-sentinel-2/L2/tiles/32/T/NS/S2A\_MSIL2A\_20210506T102021\_N0300\_R065\_T32TNS\_20210506T132458.SAFE/"

S2B\_MSIL2A\_20210511T101559\_N0300\_R065\_T32TNS\_20210511T134528.SAFE

"gs://gcp-public-data-sentinel-2/L2/tiles/32/T/NS/S2B\_MSIL2A\_20210511T101559\_N0300\_R065\_T32TNS\_20210511T134528.SAFE/"

The following attributes are included: mission, level, id\_tile, id\_orbit, sensing\_datetime, ingestion\_datetime, clouds, footprint, uuid, online.

Giving priority to Google Cloud, products available both on ESA Hub and Google Cloud (all products, in the example above) are retrieved from Google Cloud. Conversely, giving priority to ESA Hub products are taken from Copernicus if they are available online, and from Google Cloud if they are on LTA:

example\_s2\_list\_mixed2 <- s2\_list(

server = c("scihub","gcloud"),

tile = "32TNS", orbit = "065",

time\_interval = c("2021-05-01", "2021-05-15")

)

example\_s2\_list\_mixed2

A named vector with 3 SAFE archives.

S2B\_MSIL2A\_20210501T101559\_N0300\_R065\_T32TNS\_20210501T135123.SAFE

"gs://gcp-public-data-sentinel-2/L2/tiles/32/T/NS/S2B\_MSIL2A\_20210501T101559\_N0300\_R065\_T32TNS\_20210501T135123.SAFE/"

S2A\_MSIL2A\_20210506T102021\_N0300\_R065\_T32TNS\_20210506T132458.SAFE

"gs://gcp-public-data-sentinel-2/L2/tiles/32/T/NS/S2A\_MSIL2A\_20210506T102021\_N0300\_R065\_T32TNS\_20210506T132458.SAFE/"

S2B\_MSIL2A\_20210511T101559\_N0300\_R065\_T32TNS\_20210511T134528.SAFE

"<https://apihub.copernicus.eu/apihub/odata/v1/Products('1b35b71c-804c-4b9f-931c-de8e291393a4')/$value>"

The following attributes are included: mission, level, id\_tile, id\_orbit, sensing\_datetime, ingestion\_datetime, clouds, footprint, uuid, online.

In the example above the last product is available for direct download from ESA Hub and so it is retrieved from this source; first two ones are instead retrieved from Google Cloud.

Notice that, in this case (and only in this case), availability of SciHub products is checked inside s2\_list().

At this point, s2\_download() can be used as usual without the needing to set any additional arguments:

s2\_download(example\_s2\_list\_mixed2, outdir = tempdir())

**Conclusions**

Starting from version 1.5.0, sen**2r** is able to download Sentinel-2 products from Google Cloud, so to provide a way to download “old” SAFE archives without the needing to order them from Long Term Archive, wait for their availability and have the change to obtain corrupted products.