

Package: nasadata (via r-universe)

September 6, 2024

Type Package

Title Interface to Various NASA API's

Version 0.10.0

Author Eduardo Flores, Viliam Simko

Maintainer Eduardo Flores <eduardo@enelmargen.org>

Description Provides functions to access NASA's Earth Imagery and Assets API and the Earth Observatory Natural Event Tracker (EONET) webservice.

License CC0

Imports plyr, dplyr, png, jsonlite, RCurl

Suggests testthat, curl

LazyData TRUE

RoxygenNote 5.0.1

Repository <https://eflores89.r-universe.dev>

RemoteUrl <https://github.com/eflores89/nasadata>

RemoteRef HEAD

RemoteSha ac2c09c7d5cca6221098d788d37db1a593bf1ad3

Contents

earth_asset	2
earth_event	3
earth_image	4
eonet_categories	5
eonet_sources	5
plot_earth_image	6

Index	7
--------------	----------

`earth_asset`*Call Asset API*

Description

Calls NASA's Earth Imagery Assets API and returns `data.frame` with information on time and location of images between two dates.

Usage

```
earth_asset(key, lon, lat, start_date, end_date = Sys.Date())
```

Arguments

<code>key</code>	Key for API authentication.
<code>lon</code>	Longitud of coordinate position.
<code>lat</code>	Latitud of coordinate position.
<code>start_date</code>	Start date to search for image. In YYYY-MM-DD format.
<code>end_date</code>	End date to search for image. In YYYY-MM-DD format. Defaults to current system date.

Value

Returns a `data.frame` containing the following columns:

<code>date</code>	date of the sample
<code>id</code>	identifier of the sample or "NO RESULTS"
<code>type</code>	type of the sample, currently always "Point"
<code>coordinates</code>	latitude and longitude as a string delimited by a space

Examples

```
## Not run:  
key <- "123key"  
img <- earth_asset(key, -100.31008, 25.66779, "2016-01-01")  
  
## End(Not run)
```

earth_event	<i>Calls EONET webservice</i>
-------------	-------------------------------

Description

Calls NASA's Earth Observatory Natural Event Tracker (EONET) webservice and returns a list containing individual events as `data.frame`.

Usage

```
earth_event(status = c("all", "open", "closed"), sources = "all",
            category_id = "all", limit = 10, days = 20, LimitType = c("limit",
            "days", "all"), TrySimplify = TRUE)
```

Arguments

status	Accepts "open" or "closed". Defaults to "all", which includes both.
sources	Accepts character id strings from EONET sources (see <code>eonet_sources</code>)
category_id	Accepts number id strings from EONET category tree (see <code>eonet_categories</code>)
limit	Limit of events to download. If <code>LimitType = "days"</code> this is not considered. Defaults to 10.
days	Limit of days (less than today) to download events from. If <code>LimitType = "limit"</code> this is not considered. Defaults to 20.
LimitType	Type of limit to consider: "limit" (count of events), "days" (days less than today) or "all" (both limits).
TrySimplify	If TRUE tries to coerce category and event <code>data.frames</code> into one (successful if there is one category per event).

Value

Returns a list with individual events:

Events	<code>data.frame</code> - TODO description
Sources	<code>data.frame</code> - TODO description
Categories	<code>data.frame</code> - TODO description
Geography	list of <code>data.frame</code> - TODO description
Meta	<code>data.frame</code> - TODO description

Examples

```
## Not run:
event <- earth_event(limit = 1)

## End(Not run)
```

`earth_image`*Fetches image from Earth Imagery API*

Description

Calls NASA's Earth Imagery API and returns list with identification information and image.

Usage

```
earth_image(key, lon, lat, date, cloud_score = TRUE, plot = FALSE,  
            meta_only = FALSE)
```

Arguments

<code>key</code>	API Key for authentication.
<code>lon</code>	Longitude of coordinate position.
<code>lat</code>	Latitude of coordinate position.
<code>date</code>	In YYYY-MM-DD format. The API will return the image that is closest to this date.
<code>cloud_score</code>	Gives a score of percentage of cloud cover, via algorithm (see official documentation). Defaults to TRUE.
<code>plot</code>	If TRUE will plot the image via generic plot function.
<code>meta_only</code>	If TRUE will only download the meta data for the image.

Value

Returns a list of two elements:

```
image_metadata This contains a data.frame  
image_raster_data  
                This contains an array representing a raster
```

Examples

```
## Not run:  
key <- "123key"  
img <- earth_image(key, -100.31008, 25.66779, "2016-01-01")  
  
## End(Not run)
```

eonet_categories *Calls EONET category webservice.*

Description

Calls NASA's EONET Webservice and returns all categories available.

Usage

```
eonet_categories()
```

Value

Returns data.frame with 5 columns:

id	Unique id (can be used to filter earth_event)
title	Title of category
link	Direct json link (the result is equal to filtering all earth_event with category)
description	Description of category
layers	Layers of category (see oficial documentation)

Examples

```
## Not run:  
categories <- eonet_categories()  
  
## End(Not run)
```

eonet_sources *Calls EONET sources webservice*

Description

Calls NASA's EONET Webservice and returns all sources available.

Usage

```
eonet_sources()
```

Value

Returns data.frame with 4 columns:

id	Unique id (can be used to filter earth_event)
title	Title of source
source	Official source URL
link	Direct json link (the result is equal to filtering all earth_event with source)

Examples

```
## Not run:  
sources <- eonet_sources()  
  
## End(Not run)
```

plot_earth_image *Plots the image to device*

Description

To avoid S4 Classes and methods, this small wrapper simply plots an image from NASA. If the purpose is to this interactively on one image, set the parameter `plot = TRUE` in `earth_image`.

Usage

```
plot_earth_image(image_raster_data)
```

Arguments

```
image_raster_data  
                  image downloaded using earth_image.
```

Value

```
nothing
```

See Also

```
earth_image
```

Examples

```
## Not run:  
key <- "123key"  
img <- earth_image(key, -100.31008, 25.66779, "2016-01-01")  
plot_earth_image(img$image_png)  
  
## End(Not run)
```

Index

[earth_asset](#), 2
[earth_event](#), 3
[earth_image](#), 4
[eonet_categories](#), 5
[eonet_sources](#), 5

[plot_earth_image](#), 6