

Package: robis (via r-universe)

September 23, 2024

Title Ocean Biodiversity Information System (OBIS) Client
Description Client for the Ocean Biodiversity Information System
(<https://obis.org>).
Version 2.11.3
Date 2022-09-24
URL <https://github.com/iobis/robis>
BugReports <https://github.com/iobis/robis/issues>
Depends R (>= 3.1.3)
Imports httr, dplyr, jsonlite, leaflet, ggplot2, tidyr, httpcache,
tibble, mapedit, sf, rlang, purrr, stringr, curl, data.table,
tidyselect, xml2, glue, memoise, stats, methods
Collate robis.R util.R area.R checklist.R dataset.R dna.R map.R mof.R
node.R occurrence.R taxon.R
License MIT + file LICENSE
Suggests testthat, knitr, rmarkdown, spelling
RoxygenNote 7.2.1
VignetteBuilder knitr
Language en-US
Repository <https://pieterprovoost.r-universe.dev>
RemoteUrl <https://github.com/iobis/robis>
RemoteRef HEAD
RemoteSha 20167fcee7b7c5788cb978e234dc076077e60021

Contents

area	2
checklist	3
dataset	4
dna_records	5

generate_citation	5
get_geometry	6
map_ggplot	6
map_leaflet	7
measurements	7
node	8
occurrence	8
robis	10
taxon	10
unnest_extension	11
Index	12

area	<i>Fetch a list of areas</i>
------	------------------------------

Description

Fetch a list of areas

Usage

```
area(verbose = FALSE)
```

Arguments

verbose logical. Optional parameter to enable verbose logging (default = FALSE).

Value

The areas.

Examples

```
areas <- area(verbose = FALSE)
```

checklist	<i>Create a checklist.</i>
-----------	----------------------------

Description

Create a checklist.

Usage

```
checklist(scientificname = NULL, taxonid = NULL, datasetid = NULL,
  nodeid = NULL, instituteid = NULL, areaid = NULL, startdate = NULL, enddate = NULL,
  startdepth = NULL, enddepth = NULL, geometry = NULL, redlist = NULL,
  hab = NULL, wrims = NULL, dropped = NULL, flags = NULL, exclude = NULL,
  verbose = FALSE)
```

Arguments

scientificname	the scientific name.
taxonid	the taxon identifier (WoRMS AphiaID).
datasetid	the dataset identifier.
nodeid	the OBIS node identifier.
instituteid	the OBIS institute identifier.
areaid	the OBIS area identifier.
startdate	the earliest date on which occurrence took place.
enddate	the latest date on which the occurrence took place.
startdepth	the minimum depth below the sea surface.
enddepth	the maximum depth below the sea surface.
geometry	a WKT geometry string.
redlist	include only IUCN Red List species.
hab	include only IOC-UNESCO HAB species.
wrims	include only WRiMS species.
dropped	only include dropped records (TRUE), exclude dropped records (NULL) or include dropped records (include).
flags	quality flags which need to be set.
exclude	quality flags to be excluded from the results.
verbose	logical. Optional parameter to enable verbose logging (default = FALSE).

Value

The checklist.

Examples

```

taxa <- checklist(scientificname = "Tellinidae")
taxa <- checklist(geometry = "POLYGON ((2.3 51.8, 2.3 51.6, 2.6 51.6, 2.6 51.8, 2.3 51.8))")
taxa <- checklist(areaid = 10181)

```

dataset

Create a list of datasets.

Description

Create a list of datasets.

Usage

```

dataset(scientificname = NULL, taxonid = NULL, datasetid = NULL,
        nodeid = NULL, instituteid = NULL, areaid = NULL, startdate = NULL,
        enddate = NULL, startdepth = NULL, enddepth = NULL, geometry = NULL,
        redlist = NULL, hab = NULL, wrims = NULL, hasextensions = NULL,
        exclude = NULL, verbose = FALSE)

```

Arguments

scientificname	the scientific name.
taxonid	the taxon identifier (WoRMS AphiaID).
datasetid	the dataset identifier.
nodeid	the OBIS node identifier.
instituteid	the OBIS institute identifier.
areaid	the OBIS area identifier.
startdate	the earliest date on which occurrence took place.
enddate	the latest date on which the occurrence took place.
startdepth	the minimum depth below the sea surface.
enddepth	the maximum depth below the sea surface.
geometry	a WKT geometry string.
redlist	include only IUCN Red List species.
hab	include only IOC-UNESCO HAB species.
wrims	include only WRiMS species.
hasextensions	which extensions need to be present (e.g. MeasurementOrFact, DNADerived-Data, default = NULL).
exclude	quality flags to be excluded from the results.
verbose	logical. Optional parameter to enable verbose logging (default = FALSE).

Value

The datasets.

Examples

```
datasets <- dataset(scientificname = "Tellinidae")
datasets <- dataset(geometry = "POLYGON ((2.3 51.8, 2.3 51.6, 2.6 51.6, 2.6 51.8, 2.3 51.8))")
datasets <- dataset(areaid = 10181)
```

dna_records	<i>Extract DNA records from occurrence data with a dna column.</i>
-------------	--

Description

Extract DNA records from occurrence data with a dna column.

Usage

```
dna_records(df, fields = "id")
```

Arguments

df	the occurrence dataframe.
fields	columns from the occurrence dataframe to include.

Value

The DNA records.

generate_citation	<i>Generate a citation from metadata elements.</i>
-------------------	--

Description

Generate a citation from metadata elements.

Usage

```
generate_citation(title, published, url, contacts)
```

Arguments

title	the dataset title.
published	the dataset published date.
url	the dataset url.
contacts	the dataset contacts as a dataframe.

Value

A citation string.

get_geometry	<i>Get a WKT geometry by drawing on a map.</i>
--------------	--

Description

Get a WKT geometry by drawing on a map.

Usage

```
get_geometry(provider_tiles = "Esri.WorldGrayCanvas")
```

Arguments

provider_tiles the base map provider.

Value

A WKT representation of a geometry.

map_ggplot	<i>Create a ggplot2 map.</i>
------------	------------------------------

Description

Create a ggplot2 map.

Usage

```
map_ggplot(data, color = "#ff3399")
```

Arguments

data the occurrences from occurrence().
color color to be used for the dots.

Value

A ggplot object.

map_leaflet	<i>Create a leaflet map.</i>
-------------	------------------------------

Description

Create a leaflet map.

Usage

```
map_leaflet(data, color = "#ff3399",  
  provider_tiles = "Esri.WorldGrayCanvas", popup = function(x) { x["id"] },  
  antarctic = FALSE)
```

Arguments

data	the occurrences from occurrence().
color	color to be used for the dots.
provider_tiles	the base map provider.
popup	function generating the popup content.
antarctic	use antarctic polar stereographic projection.

Value

A leaflet object.

measurements	<i>Extract measurements or facts from occurrence data with a mof column.</i>
--------------	--

Description

Extract measurements or facts from occurrence data with a mof column.

Usage

```
measurements(df, fields = "id")
```

Arguments

df	the occurrence dataframe.
fields	columns from the occurrence dataframe to include.

Value

The measurements.

node	<i>Fetch a list of nodes</i>
------	------------------------------

Description

Fetch a list of nodes

Usage

```
node(verbose = FALSE)
```

Arguments

verbose logical. Optional parameter to enable verbose logging (default = FALSE).

Value

The nodes

Examples

```
nodes <- node()
```

occurrence	<i>Find occurrences.</i>
------------	--------------------------

Description

Find occurrences.

Usage

```
occurrence(scientificname = NULL, taxonid = NULL, datasetid = NULL,  
  nodeid = NULL, instituteid = NULL, areaid = NULL, startdate = NULL, enddate = NULL,  
  startdepth = NULL, enddepth = NULL, geometry = NULL,  
  measurementtype = NULL, measurementtypeid = NULL, measurementvalue = NULL,  
  measurementvalueid = NULL, measurementunit = NULL, measurementunitid = NULL,  
  redlist = NULL, hab = NULL, wrims = NULL, extensions = NULL, hasextensions = NULL,  
  mof = NULL, dna = NULL, absence = NULL, event = NULL, dropped = NULL,  
  flags = NULL, exclude = NULL, fields = NULL, qcfields = NULL, verbose = FALSE)
```


Arguments

scientificname	the scientific name.
taxonid	the taxon identifier (WoRMS AphiaID).
datasetid	the dataset identifier.
nodeid	the OBIS node identifier.
instituteid	the OBIS institute identifier.
areaid	the OBIS area identifier.
startdate	the earliest date on which occurrence took place.
enddate	the latest date on which the occurrence took place.
startdepth	the minimum depth below the sea surface.
enddepth	the maximum depth below the sea surface.
geometry	a WKT geometry string.
measurementtype	the measurement type to be included in the measurements data.
measurementtypeid	the measurement type ID to be included in the measurements data.
measurementvalue	the measurement value to be included in the measurements data.
measurementvalueid	the measurement value ID to be included in the measurements data.
measurementunit	the measurement unit to be included in the measurements data.
measurementunitid	the measurement unit ID to be included in the measurements data.
redlist	include only IUCN Red List species.
hab	include only IOC-UNESCO HAB species.
wrims	include only WRiMS species.
extensions	which extensions to include (e.g. MeasurementOrFact, DNADerivedData, default = NULL).
hasextensions	which extensions need to be present (e.g. MeasurementOrFact, DNADerivedData, default = NULL).
mof	include measurements data (default = NULL).
dna	include DNA data (default = NULL).
absence	only include absence records (TRUE), exclude absence records (NULL) or include absence records (include).
event	only include pure event records (TRUE), exclude pure event records (NULL) or include event records (include).
dropped	only include dropped records (TRUE), exclude dropped records (NULL) or include dropped records (include).
flags	quality flags which need to be set.
exclude	quality flags to be excluded from the results.
fields	fields to be included in the results.
qcfields	include lists of missing and invalid fields (default = NULL).
verbose	logical. Optional parameter to enable verbose logging (default = FALSE).

Value

The occurrence records.

Examples

```
records <- occurrence(scientificname = "Abra sibogai")
records <- occurrence(taxonid = 141438, startdate = as.Date("2007-10-10"))
```

robis

R client for the OBIS API

Description

This is an R client for the Ocean Biodiversity Information System (OBIS) API which is documented at <https://api.obis.org>.

Code and issues for this package are managed at <https://github.com/iobis/robis>.

Author(s)

Pieter Provoost, <p.provoost@unesco.org>

Samuel Bosch

taxon

Get taxon by taxon ID.

Description

Get taxon by taxon ID.

Usage

```
taxon(taxonid, verbose = FALSE)
```

Arguments

taxonid the taxon identifier (WoRMS AphiaID).

verbose logical. Optional parameter to enable verbose logging (default = FALSE).

Value

Tibble containing taxon records.

Examples

```
taxon(c(141433, 141434))
```

unnest_extension	<i>Extract extension records from occurrence data with nested extension column.</i>
------------------	---

Description

Extract extension records from occurrence data with nested extension column.

Usage

```
unnest_extension(df, extension, fields = "id")
```

Arguments

df	the occurrence dataframe.
extension	the extension type (e.g. 'MeasurementOrFact', 'DNADerivedData').
fields	columns from the occurrence dataframe to include.

Value

The extension records.

Index

[area](#), [2](#)

[checklist](#), [3](#)

[dataset](#), [4](#)

[dna_records](#), [5](#)

[generate_citation](#), [5](#)

[get_geometry](#), [6](#)

[map_ggplot](#), [6](#)

[map_leaflet](#), [7](#)

[measurements](#), [7](#)

[node](#), [8](#)

[occurrence](#), [8](#)

[robis](#), [10](#)

[taxon](#), [10](#)

[unnest_extension](#), [11](#)