

Package ‘rbison’

July 2, 2015

Title Interface to the 'USGS' 'BISON' 'API'

Description Interface to the 'USGS' 'BISON' 'API', a 'database' for species occurrence data on the web. Data comes from species in the United States from participating data providers. You can get data via 'taxonomic' and location based queries. A simple function is provided to help visualize data.

Version 0.4.8

License MIT + file LICENSE

URL <https://github.com/ropensci/rbison>

BugReports <https://github.com/ropensci/rbison/issues>

LazyData true

VignetteBuilder knitr

Imports methods, stats, plyr, httr, ggplot2, mapproj, grid, sp, dplyr, jsonlite

Suggests knitr, testthat, taxize

NeedsCompilation no

Author Scott Chamberlain [aut, cre]

Maintainer Scott Chamberlain <myrmecocystus@gmail.com>

Repository CRAN

Date/Publication 2015-07-02 01:34:06

R topics documented:

rbison-package	2
all_states	2
bison	3
bisonmap	5
bison_datause	6
bison_providers	7
bison_solr	8

bison_stats	10
bison_tax	11
fips	12
is.bison	12
is.bison_solr	13

Index	14
--------------	-----------

rbison-package	<i>rbison is an interface to the USGS Bison API.</i>
----------------	------------------------------------------------------

Description

To get started, see the vignette in `vignette(package="rbison")`

Details

See <http://bison.usgs.ornl.gov/doc/api.jsp> for API docs for the BISON API.

To cite rbison, do `citation(package='rbison')`

Use the following format to cite data retrieved from BISON:

Biodiversity occurrence data published by: (Accessed through Biodiversity Information Serving our Nation (BISON), bison.usgs.ornl.gov, YYYY-MM-DD).

For example:

Biodiversity occurrence data published by: Field Museum of Natural History, Museum of Vertebrate Zoology, University of Washington Burke Museum, and University of Turku (Accessed through Biodiversity Information Serving our Nation (BISON), bison.usgs.ornl.gov, 2013-04-22).

Author(s)

Scott Chamberlain <myrmecocystus@gmail.com>

all_states	<i>Data for a states map</i>
------------	------------------------------

Description

Data for a states map

bison

*Search for and collect data from the USGS Bison API.***Description**

Search for and collect data from the USGS Bison API.

Usage

```
bison(species = NULL, type = "scientific_name", tsn = NULL,
      start = NULL, count = 10, countyFips = NULL, county = NULL,
      state = NULL, aoi = NULL, aoibbox = NULL, params = NULL,
      what = "all", ...)
```

Arguments

species	(character) A species name.
type	(character) Type, one of scientific_name or common_name.
tsn	(numeric) Specifies the TSN to query by. If you supply a tsn it doesn't make sense to supply a species name as well. Example:162003.
start	(numeric) Record to start at.
count	(numeric) Number of records to return.
countyFips	(character) Specifies the county fips code to geographically constrain the search to one county. Character must be supplied as a number starting with zero may lose the zero. Eg: "49015".
county	(character) County name. As codes are a pain in the ass, you can put in the county name here instead of specifying a countyFips entry, and bison will attempt to look up the countyFips code. (character)
state	(character) Specifies the state name to geographically constrain the search. Example: Tennessee.
aoi	Specifies a WKT (Well-Known Text) polygon to geographically constrain the search. Eg.: c(-111.06 38.84, -110.80 39.377, -110.20 39.17, -110.20 38.90, -110.63 38.67, -111.06 38.84), which calls up the occurrences within the specified area. Check out the Wikipedia page here http://en.wikipedia.org/wiki/Well-known_text for an in depth look at the options, terminology, etc. (character)
aoibbox	Specifies a four-sided bounding box to geographically constrain the search (using format: minx,miny,maxx,maxy). The coordinates are Spherical Mercator with a datum of WGS84. Example: -111.31,38.81,-110.57,39.21 (character)
params	(character) String of parameters, one of providerID, resourceID, basisOfRecord, catalogNumber, year, computedStateFips, hierarchy_homonym_string, TSNs, recordedBy, occurrenceID, collectorNumber, provider, ownerInstitutionCollectionCode, eventDate, providedScientificName, scientificName, ITISscientificName, providedCommonName, ITIScommonName, kingdom, ITIStsn, centroid,

higherGeographyID, computedCountyFips, providedCounty, calculatedCounty, stateProvince, calculatedState, countryCode. See examples.

what What to return? One of 'all', 'summary', 'points', 'counties', 'states', 'raw', or 'list'. All data is returned from the BISON API, but this parameter lets you select just the parts you want, and the rest is discarded before returning the result to you.

... Further args passed on to httr::GET. See examples.

See Also

[bison_solr](#) [bison_tax](#)

Examples

```
## Not run:
bison(species="Bison bison", count=50, what='summary')
bison(species="Bison bison", count=50, what='points')
bison(species="Bison bison", count=50, what='counties')
bison(species="Bison bison", count=50, what='states')
bison(species="Bison bison", count=50, what='raw')
bison(species="Bison bison", count=50, what='list')

out <- bison(species="Bison bison", count=50) # by default gets 10 results
out$summary # see summary
out$counties # see county data
out$states # see state data
bisonmap(out, tomap = "points")
bisonmap(out, tomap = "county")
bisonmap(out, tomap = "state")

# Search for a common name
bison(species="Tufted Titmouse", type="common_name", what='summary')

# Constrain search to a certain county, 49015 is Emery County in Utah
bison(species="Helianthus annuus", countyFips = "49015")

# Constrain search to a certain county, specifying county name instead of code
bison(species="Helianthus annuus", county = "Los Angeles")
bison(species="Helianthus annuus", county = "Los")

# Constrain search to a certain aoi, which turns out to be Emery County, Utah as well
bison(species="Helianthus annuus",
  aoi = "POLYGON((-111.06360117772908 38.84001566645886,
    -110.80542246679359 39.37707771107983,
    -110.20117441992392 39.17722368276862,
    -110.20666758398464 38.90844075244811,
    -110.63513438085685 38.67724220095734,
    -111.06360117772908 38.84001566645886))")

# Constrain search to a certain aoibbox, which, you guessed it, is also Emery Co., Utah
bison(species="Helianthus annuus", aoibbox = '-111.31,38.81,-110.57,39.21')
```

```

# Taxonomic serial number
bison(tsn=162003)
## If you don't have tsn's, search for a taxonomic serial number
library('taxize')
poa_tsn <- get_tsn('Poa annua')
bison(tsn=poa_tsn)

# Curl debugging and other httr options, some of these examples aren't that useful, but
# are given for demonstration purposes
library("httr")
## get curl verbose output to see what's going on with your request
bison(tsn=162003, count=1, what="points", config=verbose())
## set a timeout so that the call stops after time x, compare 1st to 2nd call
# bison(tsn=162003, count=1, what="points", config=timeout(seconds=1))
# bison(tsn=162003, count=1, what="points", config=timeout(seconds=0.1))
## set cookies
bison(tsn=162003, count=1, what="points", config=set_cookies(a = 1, b = 2))
## set cookies
bison(tsn=162003, count=1, what="points", config=user_agent("rbison"))

# Params - the params function accepts a number of search terms
## Find the provider with ID 318.
bison(params='providerID:(318)')
## Find all resources with id of '318,1902' OR '318,9151', with values separated by spaces.
bison(params='resourceID:(318,1902 "318,9151")')
## Criterion may be combined using the semicolon ';' character, which translates to a logical
## AND operator. Note that field names and values are case sensitive.
bison(params='providerID:(408 "432");resourceID:(14027)')
## Search by basisOfRecord, for specimen types in this case
bison(params='basisOfRecord:(specimen)')
## Search by computedStateFips, 01 for Alabama
bison(params='computedStateFips:01')
## Search by ITIS tsn
bison(params='ITIS tsn:162003')
## Search by countryCode
bison(params='countryCode:US')
## Search by ITIS common name
bison(params='ITIS commonName:"Canada goose"')

## End(Not run)

```

bisonmap

Make map to visualize BISON data.

Description

Make map to visualize BISON data.

Usage

```

bisonmap(input = NULL, tomap = "points", geom = geom_point,
  jitter = NULL, customize = NULL)

## S3 method for class 'bison'
bisonmap(input = NULL, tomap = "points",
  geom = geom_point, jitter = NULL, customize = NULL)

## S3 method for class 'bison_solr'
bisonmap(input = NULL, tomap = "points",
  geom = geom_point, jitter = NULL, customize = NULL)

```

Arguments

input	Input bison object.
tomap	One of points (occurrences), county (counts by county), or state (counts by state).
geom	geom_point or geom_jitter, not quoted.
jitter	jitter position, see ggplot2 help.
customize	Pass in more to the plot.

Value

Map (using ggplot2 package) of points on a map.

Examples

```

## Not run:
# Using function bison
library("ggplot2")
out <- bison(species="Accipiter", type="scientific_name", count=300)
bisonmap(input=out)
bisonmap(input=out, geom=geom_jitter, jitter=position_jitter(width = 0.3, height = 0.3))

# Using function bison_solr
out <- bison_solr(scificName='Ursus americanus', rows=200)
bisonmap(out)

## End(Not run)

```

bison_datause	<i>Get BISON data use agreement details and examples for how to cite data.</i>
---------------	--------------------------------------------------------------------------------

Description

Get BISON data use agreement details and examples for how to cite data.

Usage

```
bison_datause()
```

```
bison_citation()
```

References

<http://bison.usgs.ornl.gov/doc/api.jsp#data>

bison_providers	<i>Get information about BISON data providers.</i>
-----------------	----------------------------------------------------

Description

Get information about BISON data providers.

Usage

```
bison_providers(details = FALSE, provider_no = NULL, ...)
```

Arguments

details	(logical) If TRUE, returns a list of data.frame's for each provider, including their resource details. If FALSE (default), only coarse grained data returned.
provider_no	(numeric) Provider number. If this parameter is provided, details is forced to be FALSE.
...	Further args passed on to httr::GET. See examples in bison

Value

A data.frame or list of data.frame's

Examples

```
## Not run:  
head(bison_providers())  
head(bison_providers(provider_no=131))  
out <- bison_providers(details=TRUE)  
out$National_Herbarium_of_New_South_Wales  
  
## End(Not run)
```

bison_solr	<i>Search for and collect occurrence data from the USGS Bison API using their solr endpoint.</i>
------------	--------------------------------------------------------------------------------------------------

Description

This fxn is somewhat similar to [bison](#), but interacts with the SOLR interface <http://bison.usgs.ornl.gov/doc/api.jsp#solr> instead of the OpenSearch interface <http://bison.usgs.ornl.gov/doc/api.jsp#opensearch>, which [bison](#) uses.

Usage

```
bison_solr(decimalLatitude = NULL, decimalLongitude = NULL, year = NULL,
  providerID = NULL, resourceID = NULL, pointPath = NULL,
  basisOfRecord = NULL, eventDate = NULL, computedCountyFips = NULL,
  computedStateFips = NULL, scientificName = NULL,
  hierarchy_homonym_string = NULL, TSNS = NULL, recordedBy = NULL,
  occurrenceID = NULL, catalogNumber = NULL, ITISCommonName = NULL,
  kingdom = NULL, callopts = list(), verbose = TRUE, ...)
```

Arguments

decimalLatitude	Geographic coordinate that specifies the north south position of a location on the Earth surface.
decimalLongitude	Geographic coordinate that specifies the east-west position of a location on the Earth surface.
year	The year the collection was taken.
providerID	(character) Unique identifier assigned by GBIF.
resourceID	(character) A unique identifier that is a concatenation of the provider identifier and the resource id seperated by a comma.
pointPath	A dynamic field that contains the location in longitude and latitude followed by the basis of record and an optional Geo (Spatial) precision. Geo (Spatial) precision is an added descriptor when the record is a county centroid.
basisOfRecord	One of these enumerated values: Observation, Germplasm, Fossil, Specimen, Literature, Unknown, or Living.
eventDate	The date when the occurrence was recorded.
computedCountyFips	County FIPS code conforming to standard FIPS 6-4 but with leading zeros removed.
computedStateFips	The normalized case sensitive name. For example q=state_code:"New Mexico" will return all of the occurrences from New Mexico.
scientificName	The species scientific name that is searchable in a case insensitive way.

hierarchy_homonym_string	hierarachy of the accepted or valid species name starting at kingdom. If the name is a taxonomic homonym more than one string is provided seperated by `;`.
TSNs	Accepted or valid name is provided. If the name is a taxonmic homonym more than one TSN is provided.
recordedBy	Individual responsible for the scientific record.
occurrenceID	Non-persistent unique identifier.
catalogNumber	Unique key for every record (occurrence/row) within a dataset that is not manipulated nor changed (nor generated, if not provided) during the data ingest.
ITIScommonName	Common name(s) from ITIS, e.g. "Canada goose"
kingdom	Kingdom name, from GBIF raw occurrence or BISON provider.
callopts	Further args passed on to httr::GET for HTTP debugging/inspecting. In bison, bison_providers, and bison_stats, ... is used instead of callopts, but ... is used here to pass additional Solr params.
verbose	Print message with url (TRUE, default).
...	Additional SOLR query arguments. See details.

Details

Some SOLR search parameters:

- fl Fields to return in the query
- rows Number of records to return
- sort Field to sort by, see examples
- facet Facet or not, logical
- facet.field Fields to facet by

You can also use highlighting in solr search, but I'm not sure I see a use case for it with BISON data, though you can do it with this function.

For a tutorial see here http://lucene.apache.org/solr/3_6_2/doc-files/tutorial.html

Value

An object of class `bison_solr` - which is a list with slots for number of records found (`num_found`), records, highlight, or facets.

See Also

[bison_tax bison](#)

The USGS BISON Solr installation version as of 2014-10-14 was 4.4.

Examples

```
## Not run:
bison_solr(scientificName='Ursus americanus')

## End(Not run)
```

`bison_stats`*Get statistics about BISON downloads.*

Description

Get statistics about BISON downloads.

Usage

```
bison_stats(what = "stats", ...)
```

Arguments

`what` (character) One of stats (default), search, download, or wms. See Details.
`...` Further args passed on to `httr::GET`. See examples in `bison`

Details

For the 'what' parameter:

- `stats` - Retrieve all data provider accumulated statistics.
- `search` - Retrieve data provider statistics for BISON searches.
- `download` - Retrieve data provider statistics for data downloads from BISON.
- `wms` - Retrieve data provider statistics for BISON OGC WMS tile requests.

Value

A list of data frame's with names of the list the different data sources

Examples

```
## Not run:  
out <- bison_stats()  
out <- bison_stats(what='search')  
out <- bison_stats(what='download')  
out <- bison_stats(what='wms')  
out$Arctos  
out$Harvard_University_Herbaria  
out$ZooKeys  
  
## End(Not run)
```

bison_tax	<i>Search for and collect taxonomic name data from the USGS Bison API using solr.</i>
-----------	---------------------------------------------------------------------------------------

Description

See the SOLR documentation here <http://lucene.apache.org/solr/> for other parameters you can use.

The following two methods are possible, as far as I know you can only use one at a time:

- vernacularName The species specific common names that is searchable in a case insensitive way.
- scientificName The species scientific name that is associated with a common name that is searchable in a case insensitive way.

Usage

```
bison_tax(query, method = "vernacularName", exact = FALSE, parsed = TRUE,
          callopts = list(), ...)
```

Arguments

query	Name to search for. Required.
method	The field to query by. See description below for details.
exact	Exact matching or not. See examples. Defaults to FALSE.
parsed	If TRUE (default) creates data.frame of names data output. Otherwise, a list.
callopts	Further args passed on to htr::GET for HTTP debugging/inspecting. In bison, bison_providers, and bison_stats, ... is used instead of callopts, but ... is used here to pass additional Solr params.
...	Further solr arguments passed in to the query. See examples below.

Value

A list.

See Also

[bison_solr](#) [bison](#)

Examples

```
## Not run:
# All taxa
bison_tax("*:*)

# Some example calls
```

```

bison_tax(query="*bear")
bison_tax(query="Helianthus", method="scientificName")

# Exact argument, here nothing found with latter call as '*bear' doesn't exist,
# which makes sense
bison_tax(query="*bear", exact=FALSE)
bison_tax(query="*bear", exact=TRUE)

# Using solr arguments (not all Solr arguments work)
## Return a certain number of rows
bison_tax(query="*bear", method="vernacularName", rows=3)
## Return certain fields
bison_tax(query="*bear", method="vernacularName", fl='vernacularName')

# Curl options
library("httr")
bison_tax(query='*dolphin', callopts=verbose())

## End(Not run)

```

fips	<i>Fips codes for states and counties</i>
------	-------------------------------------------

Description

Fips codes for states and counties

is.bison	<i>Check if object is of class bison</i>
----------	------------------------------------------

Description

Check if object is of class bison

Usage

```
is.bison(x)
```

Arguments

x	input
---	-------

`is.bison_solr` *Check if object is of class `bison_solr`*

Description

Check if object is of class `bison_solr`

Usage

`is.bison_solr(x)`

Arguments

`x` input

Index

*Topic **data**

all_states, [2](#)

fips, [12](#)

*Topic **package**

rbison-package, [2](#)

all_states, [2](#)

bison, [3](#), [8](#), [9](#), [11](#)

bison_citation (bison_datause), [6](#)

bison_datause, [6](#)

bison_providers, [7](#)

bison_solr, [4](#), [8](#), [11](#)

bison_stats, [10](#)

bison_tax, [4](#), [9](#), [11](#)

bisonmap, [5](#)

fips, [12](#)

is.bison, [12](#)

is.bison_solr, [13](#)

rbison (rbison-package), [2](#)

rbison-package, [2](#)