Package ‘AntWeb’
August 29, 2016

**Type** Package  
**Date** 2014-08-13  
**Title** programmatic interface to the AntWeb  
**Description** A complete programmatic interface to the AntWeb database from the California Academy of Sciences.  
**Version** 0.7  
**Depends** R (>= 3.0.1)  
**License** CC0  
**URL** [https://github.com/ropensci/AntWeb](https://github.com/ropensci/AntWeb)  
**BugReports** [https://github.com/ropensci/AntWeb/issues](https://github.com/ropensci/AntWeb/issues)  
**LazyData** true  
**Imports** rjson, plyr, assertthat, httr, leafletR (>= 0.1-1)  
**Author** 'Karthik Ram' [aut, cre]  
**Maintainer** 'Karthik Ram' <karthik.ram@gmail.com>  
**NeedsCompilation** no  
**Repository** CRAN  
**Date/Publication** 2014-08-14 00:40:13

### R topics documented:

<table>
<thead>
<tr>
<th>Name</th>
<th>Number of lines</th>
</tr>
</thead>
<tbody>
<tr>
<td>AntWeb</td>
<td>2</td>
</tr>
<tr>
<td>aw_cbind</td>
<td>2</td>
</tr>
<tr>
<td>aw_code</td>
<td>3</td>
</tr>
<tr>
<td>aw_coords</td>
<td>3</td>
</tr>
<tr>
<td>aw_data</td>
<td>4</td>
</tr>
<tr>
<td>aw_data_all</td>
<td>5</td>
</tr>
<tr>
<td>aw_distinct</td>
<td>6</td>
</tr>
<tr>
<td>aw_images</td>
<td>6</td>
</tr>
<tr>
<td>aw_map</td>
<td>7</td>
</tr>
<tr>
<td>aw_unique</td>
<td>8</td>
</tr>
<tr>
<td>print.antweb</td>
<td>8</td>
</tr>
</tbody>
</table>
AntWeb

**Description**

AntWeb

**Details**

The AntWeb world's largest online database of images, specimen records, and natural history information on ants. The database is maintained and hosted by the California Academy of Sciences.

**aw_cbind**

**Description**

Allows for combining split AntWeb calls (e.g. paginated calls) back into one single result object

**Usage**

`aw_cbind(results)`

**Arguments**

- `results` A list of objects of class `antweb`

**Examples**

```r
## Not run:
x1 <- aw_data(genus = "crematogaster", georeferenced = TRUE)
x2 <- aw_data(genus = "crematogaster", georeferenced = TRUE, offset = 1000)
x12 <- aw_cbind(list(x1, x2))

## End(Not run)
```
**aw_code**

---

**Description**
Retrieve data by specimen id

**Usage**

\[
\text{aw_code(occurrenceid = NULL, catalogNumber = NULL)}
\]

**Arguments**

- **occurrenceid**: A unique id in the AntWeb database identifying a particular specimen
- **catalogNumber**: Specimen catalogue number

**Value**
list

**See Also**

- **aw_data**

**Examples**

```r
# data_by_code <- aw_code(occurrenceid = "CAS:ANTWEB:alas188691")
# data_by_code <- aw_code(catalognumber="inb0003695883")
```

---

**aw_coords**

---

**Description**
Retrieve AntWeb data by location. A radius argument can be supplied as a search radius around a point on the map.

**Usage**

\[
\text{aw_coords(coord = NULL, r = NULL)}
\]

**Arguments**

- **coord**: Latitude and Longitude. Should be supplied as lat,long. Example: 37.76,-122.45
- **r**: A radius in kilometers. For 2 km add r = 2
**Value**

Values

**Examples**

```r
# data_by_loc <- aw_coords(coord = "37.76,-122.45", r = 2)
```

---

**Description**

This function allows a user to query the AntWeb database by any taxonomic rank or full species name.

**Usage**

```r
aw_data(genus = NULL, species = NULL, scientific_name = NULL, georeferenced = FALSE, min_elevation = NULL, max_elevation = NULL, type = NULL, habitat = NULL, country = NULL, min_date = NULL, max_date = NULL, bbox = NULL, limit = NULL, offset = NULL, quiet = FALSE)
```

**Arguments**

- **genus**: An ant genus name
- **species**: A species name
- **scientific_name**: An easier way to pass the Genus and species name together, especially when the data are derived from other packages.
- **georeferenced**: Default is `FALSE`. Set to `TRUE` to return only data with lat/long information. Note that this filtering takes place on the client-side, not server side.
- **bbox**: A lat long bounding box. Format is `lat,long,lat,long`. Use this website: http://boundingbox.klokantech.com/ to quickly grab a bbox (set format on bottom left to csv and be sure to switch the order from long, lat, long, lat to lat, long, lat, long) Just set the format on the bottom left to CSV.
- **type**: A holotype
- **habitat**: A fuzzy search by any habitat
- **country**: A country name
- **min_date**: A lower date bound in the format `yyyy-mm-dd`
- **max_date**: An upper date bound in the format `yyyy-mm-dd`
- **min_elevation**: A lower elevation bound
- **max_elevation**: An upper elevation bound
- **limit**: A numeric value to limit number of records
- **offset**: An offset best used with limit as a way to paginate records
- **quiet**: If true, any informative messages will be suppressed
**Description**

This is a thin wrapper around `aw_data`.

**Usage**

```r
aw_data_all(..., progress = "text")
```

**Arguments**

- `...` All the same arguments that get passed to `aw_data`.
- `progress` Default is on and set to `text`. Set to `none` to suppress.

**See Also**

`aw_data`.

**Examples**

```r
## Not run:
# crem <- aw_data_all(genus = "crematogaster", georeferenced = TRUE)

## End(Not run)
```
Description

Retrieves a data.frame of distinct ranks based on various restrictions

Usage

aw_distinct(rank = "genus", habitat = NULL, country = NULL, min_elevation = NULL, max_elevation = NULL, limit = 1000, offset = NULL)

Arguments

rank = "genus" Default is genus. But you can also use phylum, sub-phylum etc
habitat The habitat type
country Country name
min_elevation Min elevation recorded for specimen
max_elevation Max elevation recorded for specimen
limit = 1000 Default limit. Set higher if necessary
offset To be used in conjunction with limit

Examples

## Not run:
aw_distinct(rank = "genus", country = "Madagascar")

## End(Not run)

Description

Download ant images based on time elapsed and/or type.

Usage

aw_images(since = NULL, img_type = NULL)
aw_map

Arguments

- **since**: number of days in the past to query
- **img_type**: h for head, d for dorsal, p for profile, and l for label. If a img_type is not specified, all images are retrieved.

Value
data.frame

Examples

```r
## Not run:
z <- aw_images(since = 5)
z1 <- aw_images(since = 5, img_type = "d")

## End(Not run)
```

---

aw_map

*LeafletJS Map*

Description

Builds an interactive map of locations for any list of species

Usage

```r
aw_map(aw_obj, dest = tempdir(), title = "AntWeb species map", incl.data = TRUE)
```

Arguments

- **aw_obj**: Result from a search on AntWeb
- **dest**: Location where the html file and geojson file should be stored. Default is the temp directory
- **title**: Title of the map.
- **incl.data**: Default is TRUE. Writes geoJSON data into the html file to get around security restrictions in browsers like Google Chrome. Set to FALSE to read from a separate local geoJSON file.

Examples

```r
## Not run:
acanthognathus_df <- aw_data(genus = "acanthognathus", georeferenced = TRUE)
aw_map(acanthognathus_df)

# Or just plot data by habitat. So for e.g. using sandstone as a substrate
sandstone <- aw_data(habitat = "sandstone")
aw_map(sandstone)

## End(Not run)
```
aw_unique

Description
Get a list of unique names within any taxonomic rank

Usage
aw_unique(rank = NULL, name = NULL)

Arguments
- rank: A taxonomic rank. Allowed values are subfamily, genus or species
- name: Optional. If left blank, the query will return a list of all unique names inside the supplied rank.

Value
data.frame

See Also
aw_data

Examples
## Not run:
subfamily_list <- aw_unique(rank = "subfamily")
# genus_list <- aw_unique(rank = "genus")
# species_list <- aw_unique(rank = "species")
## End(Not run)

print.antweb

Description
Print a summary for an antweb object

Usage
## S3 method for class 'antweb'
print(x, ...)

print.antweb
Print a summary for an antweb object

Arguments

x  An object of class antweb
  . . .  additional arguments
Index

*Topic data
  aw_data, 4
  aw_data_all, 5
*Topic download
  aw_data, 4
  aw_data_all, 5
*Topic map
  aw_map, 7

AntWeb, 2
AntWeb-package (AntWeb), 2
aw_cbind, 2
aw_code, 3
aw_coords, 3
aw_data, 3, 4, 4, 8
aw_data_all, 5
aw_distinct, 6
aw_images, 6
aw_map, 7
aw_unique, 8

print.antweb, 8