

# Package ‘rglobi’

October 14, 2022

**Encoding** UTF-8

**Language** en-US

**Type** Package

**Title** R Interface to Global Biotic Interactions

**Description** A programmatic interface to the web service methods provided by Global Biotic Interactions (GloBI) (<https://www.globalbioticinteractions.org/>). GloBI provides access to spatial-temporal species interaction records from sources all over the world. rglobi provides methods to search species interactions by location, interaction type, and taxonomic name. In addition, it supports Cypher, a graph query language, to allow for executing custom queries on the GloBI aggregate species interaction data set.

**Version** 0.2.28

**Date** 2022-08-24

**URL** <https://docs.ropensci.org/rglobi/>,  
<https://github.com/ropensci/rglobi>

**BugReports** <https://github.com/ropensci/rglobi/issues>

**VignetteBuilder** knitr

**Depends** R (>= 3.0.1)

**License** MIT + file LICENSE

**Imports** rjson (>= 0.2.13), readr (>= 1.3.1), RCurl (>= 0.3.4), curl (>= 0.3.3)

**Suggests** testthat (>= 0.7), openssl, httr, markdown, knitr

**RoxygenNote** 6.1.1

**NeedsCompilation** no

**Author** Jorrit Poelen [aut, cre],  
Stephen Gosnell [aut],  
Sergey Slyusarev [aut],  
Helen Waters [aut]

**Maintainer** Jorrit Poelen <jhpoe1en@xs4a11.nl>

**Repository** CRAN

**Date/Publication** 2022-08-25 13:50:02 UTC

## R topics documented:

get_child_taxa . . . . .	2
get_data_fields . . . . .	3
get_interactions . . . . .	4
get_interactions_by_taxa . . . . .	4
get_interactions_by_type . . . . .	6
get_interactions_in_area . . . . .	7
get_interaction_areas . . . . .	7
get_interaction_matrix . . . . .	8
get_interaction_table . . . . .	9
get_interaction_types . . . . .	10
get_predators_of . . . . .	11
get_pre_y_of . . . . .	11
query . . . . .	12

<b>Index</b>	<b>13</b>
--------------	-----------

---

get_child_taxa	<i>Returns all known child taxa with known interaction of specified taxa and rank.</i>
----------------	--

---

### Description

Returns all known child taxa with known interaction of specified taxa and rank.

### Usage

```
get_child_taxa(taxon.names, rank = "Species", skip = 0, limit = 25,
  opts = list())
```

### Arguments

taxon.names	list of taxa of which child taxa should be included.
rank	selected taxonomic rank of child taxa
skip	number of child taxon names to skip before returning result. May be used for pagination.
limit	maximum number of child taxon names returned
opts	list of options including web service configuration like "port" and "host"

### Value

list of child taxon names

**See Also**

Other interactions: [get\\_interaction\\_matrix](#), [get\\_interaction\\_table](#), [get\\_interaction\\_types](#), [get\\_interactions\\_by\\_taxa](#), [get\\_interactions\\_by\\_type](#), [get\\_interactions](#), [get\\_predators\\_of](#), [get\\_preys\\_of](#)

**Examples**

```
## Not run:  
get_child_taxa(list("Aves"))  
  
## End(Not run)
```

---

get_data_fields	<i>List data fields identified in GloBI database</i>
-----------------	--

---

**Description**

Returns data frame with supported data fields

**Usage**

```
get_data_fields(opts = list(), read_csv = read_csv_online)
```

**Arguments**

opts	list of named options to configure GloBI API
read_csv	function used to find csv associated to query url, defaulting to online query method

**Value**

Returns data frame of supported data fields

**Examples**

```
## Not run:  
get_data_fields()  
  
## End(Not run)
```

---

get\_interactions      *Get Species Interaction from GloBI*

---

### Description

Get Species Interaction from GloBI

### Usage

```
get_interactions(taxon = "Homo sapiens", interaction.type = "preysOn",
  ...)
```

### Arguments

taxon	canonical scientific name of source taxon (e.g. Homo sapiens)
interaction.type	the preferred interaction type (e.g. preysOn)
...	list of options to configure GloBI API

### Value

species interactions between source and target taxa

### See Also

Other interactions: [get\\_child\\_taxa](#), [get\\_interaction\\_matrix](#), [get\\_interaction\\_table](#), [get\\_interaction\\_types](#), [get\\_interactions\\_by\\_taxa](#), [get\\_interactions\\_by\\_type](#), [get\\_predators\\_of](#), [get\\_prey\\_of](#)

### Examples

```
## Not run:
get_interactions("Homo sapiens", "preysOn")
get_interactions("Insecta", "parasiteOf")

## End(Not run)
```

---

get\_interactions\_by\_taxa  
*Return interactions involving specific taxa*

---

### Description

Returns interactions involving specific taxa. Secondary (target) taxa and spatial boundaries may also be set

**Usage**

```
get_interactions_by_taxa(sourcetaxon, targettaxon = NULL,
  interactiontype = NULL, accordingto = NULL,
  showfield = c("source_taxon_external_id", "source_taxon_name",
    "source_taxon_path", "source_specimen_life_stage", "interaction_type",
    "target_taxon_external_id", "target_taxon_name", "target_taxon_path",
    "target_specimen_life_stage", "latitude", "longitude", "study_citation",
    "study_external_id", "study_source_citation"), otherkeys = NULL,
  bbox = NULL, returnobservations = F, opts = list(),
  read_csv = read_csv_online)
```

**Arguments**

sourcetaxon	Taxa of interest (consumer, predator, parasite); may be specified as "Genus species" or higher level (e.g., Genus, Family, Class).
targettaxon	Taxa of interest (prey, host); may be specified as "Genus species" or higher level (e.g., Genus, Family, Class)
interactiontype	Interaction types of interest (prey, host); may be specified as listed by <code>get_interaction_types()</code>
accordingto	Data source of interest
showfield	Data fields of interest (e. g. <code>source_taxon_external_id</code> , <code>source_taxon_name</code> ); may be specified as listed by <code>get_data_fields()</code>
otherkeys	list of key-value pairs to query any field not covered by other parameters; keys may be specified as listed by <code>get_data_fields()</code>
bbox	Coordinates in EPSG:4326 decimal degrees defining "left, bottom, right, top" of bounding box
returnobservations	if true, all individual observations are returned, else only distinct relationships
opts	list of named options to configure GloBI API
read_csv	function used to find csv associated to query url, defaulting to online query method

**Value**

Returns data frame of interactions

**Note**

For data sources in which type of interactions were not specified, the interaction is labeled "interacts\_with"

**See Also**

Other interactions: [get\\_child\\_taxa](#), [get\\_interaction\\_matrix](#), [get\\_interaction\\_table](#), [get\\_interaction\\_types](#), [get\\_interactions\\_by\\_type](#), [get\\_interactions](#), [get\\_predators\\_of](#), [get\\_prey\\_of](#)

## Examples

```
## Not run:
get_interactions_by_taxa(sourcetaxon = "Rattus")
get_interactions_by_taxa(sourcetaxon = "Aves", targettaxon = "Rattus")
get_interactions_by_taxa(sourcetaxon = "Rattus rattus",
bbox = c(-67.87, 12.79, -57.08, 23.32))

## End(Not run)
```

---

get\_interactions\_by\_type

*Get Species Interactions by Interaction Type from GloBI*

---

## Description

Get Species Interactions by Interaction Type from GloBI

## Usage

```
get_interactions_by_type(interactiontype = c("interactsWith"), ...)
```

## Arguments

interactiontype	the requested interaction type (e.g. preysOn)
...	list of options to configure GloBI API

## Value

species interactions given provided interaction type(s)

## See Also

Other interactions: [get\\_child\\_taxa](#), [get\\_interaction\\_matrix](#), [get\\_interaction\\_table](#), [get\\_interaction\\_types](#), [get\\_interactions\\_by\\_taxa](#), [get\\_interactions](#), [get\\_predators\\_of](#), [get\\_prey\\_of](#)

## Examples

```
## Not run:
get_interactions_by_type(interactiontype = c("eats", "eatenBy"))
get_interactions_by_type(interactiontype = "parasiteOf")

## End(Not run)
```

---

`get_interactions_in_area`*Return all interactions in specified area*

---

**Description**

Returns all interactions in data base in area specified in arguments

**Usage**

```
get_interactions_in_area(bbox, ...)
```

**Arguments**

<code>bbox</code>	Coordinates in EPSG:4326 decimal degrees defining "left, bottom, right, top" of bounding box
<code>...</code>	list of named options to configure GloBI API

**Value**

Returns data frame of interactions

**See Also**

Other areas: [get\\_interaction\\_areas](#)

**Examples**

```
## Not run:  
get_interactions_in_area(bbox = c(-67.87, 12.79, -57.08, 23.32))  
  
## End(Not run)
```

---

`get_interaction_areas` *Find locations at which interactions were observed*

---

**Description**

Returns all locations (latitude,longitude) of interactions in data base or area specified in arguments

**Usage**

```
get_interaction_areas(bbox = NULL, read_csv = read_csv_online, ...)
```

**Arguments**

bbox	Coordinates in EPSG:4326 decimal degrees defining "left, bottom, right, top" of bounding box
read_csv	function used to find csv associated to query url, defaulting to online query method
...	list of named options to configure GloBI API

**Value**

Returns data frame of coordinates

**See Also**

Other areas: [get\\_interactions\\_in\\_area](#)

**Examples**

```
## Not run:
get_interaction_areas ()
get_interaction_areas (bbox=c(-67.87,12.79,-57.08,23.32))

## End(Not run)
```

---

get\_interaction\_matrix

*Get Interaction Matrix. Constructs an interaction matrix indicating whether source taxa (rows) or target taxa (columns) are known to interact with given type.*

---

**Description**

Get Interaction Matrix. Constructs an interaction matrix indicating whether source taxa (rows) or target taxa (columns) are known to interact with given type.

**Usage**

```
get_interaction_matrix(source.taxon.names = list("Homo sapiens"),
  target.taxon.names = list("Mammalia"), interaction.type = "eats",
  opts = list(), read_csv = read_csv_online)
```

**Arguments**

source.taxon.names	list of source taxon names (e.g. list('Mammalia', 'Aves', 'Ariopsis felis'))
target.taxon.names	list of target taxon names



interaction.type	the preferred interaction type (e.g. preysOn)
opts	list of options to configure GloBI API
read_csv	function used to find csv associated to query url, defaulting to online query method

**Value**

matrix representing species interactions between source and target taxa

**See Also**

Other interactions: [get\\_child\\_taxa](#), [get\\_interaction\\_table](#), [get\\_interaction\\_types](#), [get\\_interactions\\_by\\_taxa](#), [get\\_interactions\\_by\\_type](#), [get\\_interactions](#), [get\\_predators\\_of](#), [get\\_prey\\_of](#)

**Examples**

```
## Not run:
get_interaction_matrix("Homo sapiens", "Mammalia", "interactsWith")

## End(Not run)
```

---

get\_interaction\_table *Returns all known child taxa with known interaction of specified source and target taxa on any rank.*

---

**Description**

Returns all known child taxa with known interaction of specified source and target taxa on any rank.

**Usage**

```
get_interaction_table(source.taxon.names = list(),
  target.taxon.names = list(), interaction.type = "preysOn",
  skip = 0, limit = 100, opts = list())
```

**Arguments**

source.taxon.names	list of taxon names for source
target.taxon.names	list of taxon names for target
interaction.type	kind of interaction
skip	number of records skipped before including record in result table, used in pagination
limit	maximum number of interaction to include
opts	connection parameters and other options

**Value**

table of matching source, target and interaction types

**See Also**

Other interactions: [get\\_child\\_taxa](#), [get\\_interaction\\_matrix](#), [get\\_interaction\\_types](#), [get\\_interactions\\_by\\_taxa](#), [get\\_interactions\\_by\\_type](#), [get\\_interactions](#), [get\\_predators\\_of](#), [get\\_preym\\_of](#)

**Examples**

```
## Not run:  
get_interaction_table(source.taxon.names = list("Aves"), target.taxon.names = list('Insecta'))  
  
## End(Not run)
```

---

get\_interaction\_types *List interactions identified in GloBI database*

---

**Description**

Returns data frame with supported interaction types

**Usage**

```
get_interaction_types(opts = list(), read_csv = read_csv_online)
```

**Arguments**

opts	list of named options to configure GloBI API
read_csv	function used to find csv associated to query url, defaulting to online query method

**Value**

Returns data frame of supported interaction types

**See Also**

Other interactions: [get\\_child\\_taxa](#), [get\\_interaction\\_matrix](#), [get\\_interaction\\_table](#), [get\\_interactions\\_by\\_taxa](#), [get\\_interactions\\_by\\_type](#), [get\\_interactions](#), [get\\_predators\\_of](#), [get\\_preym\\_of](#)

**Examples**

```
## Not run:  
get_interaction_types()  
  
## End(Not run)
```

---

get\_predators\_of      *Get a List of Predators of a Given Prey Taxon*

---

**Description**

Get a List of Predators of a Given Prey Taxon

**Usage**

```
get_predators_of(taxon = "Rattus rattus", ...)
```

**Arguments**

taxon	scientific name of prey taxon. Can be any taxonomic rank (e.g. Rattus rattus, Decapoda)
...	list of named options to configure the GloBI API

**Value**

list of recorded prey-predator interactions that involve the desired prey taxon.

**See Also**

Other interactions: [get\\_child\\_taxa](#), [get\\_interaction\\_matrix](#), [get\\_interaction\\_table](#), [get\\_interaction\\_types](#), [get\\_interactions\\_by\\_taxa](#), [get\\_interactions\\_by\\_type](#), [get\\_interactions](#), [get\\_preying\\_of](#)

**Examples**

```
## Not run:  
get_predators_of("Rattus rattus")  
get_predators_of("Primates")  
  
## End(Not run)
```

---

get\_preying\_of      *Get a List of Prey for given Predator Taxon*

---

**Description**

Get a List of Prey for given Predator Taxon

**Usage**

```
get_preying_of(taxon = "Homo sapiens", ...)
```

**Arguments**

taxon	scientific name of predator taxon. Can be any taxonomic rank (e.g. Homo sapiens, Animalia)
...	list of named options to configure GloBI API

**Value**

list of recorded predator-prey interactions that involve the desired predator taxon

**See Also**

Other interactions: [get\\_child\\_taxa](#), [get\\_interaction\\_matrix](#), [get\\_interaction\\_table](#), [get\\_interaction\\_types](#), [get\\_interactions\\_by\\_taxa](#), [get\\_interactions\\_by\\_type](#), [get\\_interactions](#), [get\\_predators\\_of](#)

**Examples**

```
## Not run:
get_prey_of("Homo sapiens")
get_prey_of("Primates")

## End(Not run)
```

---

query

---

*Executes a Cypher Query Against GloBI's Neo4j Instance*


---

**Description**

Executes a Cypher Query Against GloBI's Neo4j Instance

**Usage**

```
query(cypherQuery, opts = list())
```

**Arguments**

cypherQuery	Cypher query (see <a href="http://github.com/globalbioticinteractions/globalbioticinteractions/wiki/cypher">http://github.com/globalbioticinteractions/globalbioticinteractions/wiki/cypher</a> for examples)
opts	list of named options to configure GloBI API

**Value**

result of cypher query string

# Index

## \* areas

get\_interaction\_areas, 7  
get\_interactions\_in\_area, 7

## \* database

get\_data\_fields, 3  
get\_interaction\_areas, 7  
get\_interaction\_types, 10  
get\_interactions\_by\_taxa, 4  
get\_interactions\_in\_area, 7

## \* data

get\_data\_fields, 3

## \* interactions

get\_child\_taxa, 2  
get\_interaction\_matrix, 8  
get\_interaction\_table, 9  
get\_interaction\_types, 10  
get\_interactions, 4  
get\_interactions\_by\_taxa, 4  
get\_interactions\_by\_type, 6  
get\_predators\_of, 11  
get\_prey\_of, 11

get\_child\_taxa, 2, 4–6, 9–12  
get\_data\_fields, 3  
get\_interaction\_areas, 7, 7  
get\_interaction\_matrix, 3–6, 8, 10–12  
get\_interaction\_table, 3–6, 9, 9, 10–12  
get\_interaction\_types, 3–6, 9, 10, 10, 11,  
12  
get\_interactions, 3, 4, 5, 6, 9–12  
get\_interactions\_by\_taxa, 3, 4, 4, 6, 9–12  
get\_interactions\_by\_type, 3–5, 6, 9–12  
get\_interactions\_in\_area, 7, 8  
get\_predators\_of, 3–6, 9, 10, 11, 12  
get\_prey\_of, 3–6, 9–11, 11

query, 12