

Package ‘rdatacite’

August 29, 2016

Type Package

Title 'DataCite' Client for 'OAI-PMH' Methods and their Search 'API'

Description Client for the web service methods provided by 'DataCite' (<<https://www.datacite.org/>>), including functions to interface with their 'OAI-PMH' 'metadata' service, and a 'RESTful' search 'API'. The 'API' is backed by 'SOLR', allowing expressive queries, including faceting, statistics on variables, and 'more-like-this' queries.

Version 0.1.0

License MIT + file LICENSE

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

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

Imports oai (>= 0.2.0), solrium (>= 0.3.0)

Suggests knitr, testthat

RoxygenNote 5.0.1

NeedsCompilation no

Author Scott Chamberlain [aut, cre]

Maintainer Scott Chamberlain <myrmecocystus@gmail.com>

Repository CRAN

Date/Publication 2016-02-12 00:49:48

R topics documented:

rdatacite-package	2
dc_oai_getrecord	3
dc_oai_identify	3
dc_oai_listidentifiers	4
dc_oai_listmetadataformats	5
dc_oai_listrecords	5
dc_oai_listsets	6
dc_search	7
Prefixes	9

rdatacite-package *DataCite R client.*

Description

DataCite R client.

OAI-PMH functions

- [dc_oai_getrecord](#) - Get records
- [dc_oai_identify](#) - identify the OAI-PMH service
- [dc_oai_listidentifiers](#) - List identifiers
- [dc_oai_listmetadataformats](#) - List metadata formats
- [dc_oai_listrecords](#) - List records
- [dc_oai_listsets](#) - List sets

Search functions

- [dc_search](#) - General search, only returns documents
- [dc_facet](#) - Faceting only (w/o general search)
- [dc_mlt](#) - More like this (w/o general search)
- [dc_stats](#) - Stats search (w/o general search)

Vignettes

Coming soon...

Author(s)

Scott Chamberlain <myrmecocystus@gmail.com>

dc_oai_getrecord *Retrieve an individual record from the DataCite repository*

Description

Retrieve an individual record from the DataCite repository

Usage

```
dc_oai_getrecord(id)
```

Arguments

id DataCite identifier, e.g., "oai:oai.datacite.org:6718729". If left blank, get all metadataformats

Examples

```
## Not run:
# you can pass in just the numeric part of the ID
dc_oai_getrecord(56225)
dc_oai_getrecord(c(56225, 6667400))

# Or, the entire thing
dc_oai_getrecord("oai:oai.datacite.org:56225")

# Or, mixed
dc_oai_getrecord(c("56225", "oai:oai.datacite.org:6667400"))

today <- format(Sys.Date(), "%Y-%m-%d")
temp <- dc_oai_listidentifiers(from = today)
dc_oai_getrecord(temp$identifier[1:2])

## End(Not run)
```

dc_oai_identify *Learn about the Dryad OAI-PMH service*

Description

Learn about the Dryad OAI-PMH service

Usage

```
dc_oai_identify(...)
```

Arguments

... Curl options passed on to [GET](#)

Examples

```
## Not run:
dc_oai_identify()

## End(Not run)
```

dc_oai_listidentifiers

Retrieve record headers

Description

Retrieve record headers

Usage

```
dc_oai_listidentifiers(from = NULL, until = NULL, set = NULL,
  prefix = "oai_dc", token = NULL, ...)
```

Arguments

from	specifies that records returned must have been created/update/deleted on or after this date.
until	specifies that records returned must have been created/update/deleted on or before this date.
set	specifies the set that returned records must belong to.
prefix	specifies the metadata format that the records will be returned in. One of: oai_dc (default), oai_datacite, or datacite. See Prefixes for more info.
token	a token previously provided by the server to resume a request where it last left off.
...	Curl options passed on to GET

Examples

```
## Not run:
today <- format(Sys.Date(), "%Y-%m-%d")
dc_oai_listidentifiers(from = today)
dc_oai_listidentifiers(from = '2011-06-01T', until = '2011-07-01T')
dc_oai_listidentifiers(set = "REFQUALITY")

## End(Not run)
```

`dc_oai_listmetadataformats`

List available metadata formats that the DataCite repository can disseminate

Description

List available metadata formats that the DataCite repository can disseminate

Usage

```
dc_oai_listmetadataformats(id = NULL, ...)
```

Arguments

<code>id</code>	DataCite identifier, e.g., "oai:oai.datacite.org:6718729". If left blank, get all metadataformats
<code>...</code>	Curl options passed on to GET

Examples

```
## Not run:  
dc_oai_listmetadataformats()  
dc_oai_listmetadataformats("oai:oai.datacite.org:6718729")  
  
## End(Not run)
```

`dc_oai_listrecords` *List the records in the DataCite metadata repository.*

Description

List the records in the DataCite metadata repository.

Usage

```
dc_oai_listrecords(from = NULL, until = NULL, set = NULL,  
  prefix = "oai_dc", token = NULL, ...)
```

Arguments

from	specifies that records returned must have been created/update/deleted on or after this date.
until	specifies that records returned must have been created/update/deleted on or before this date.
set	specifies the set that returned records must belong to.
prefix	specifies the metadata format that the records will be returned in. One of: oai_dc (default), oai_datacite, or datacite. See Prefixes for more info.
token	a token previously provided by the server to resume a request where it last left off.
...	Curl options passed on to GET

Examples

```
## Not run:
dc_oai_listrecords(from = '2011-06-01T', until = '2011-07-01T')
dc_oai_listrecords(from = '2011-06-01T', until = '2011-11-01T', set = "REFQUALITY")

## End(Not run)
```

dc_oai_listsets	<i>Retrieve the set structure of DataCite</i>
-----------------	---

Description

Useful for selective harvesting

Usage

```
dc_oai_listsets(token = NULL, ...)
```

Arguments

token	a token previously provided by the server to resume a request where it last left off.
...	Curl options passed on to GET

Examples

```
## Not run:
dc_oai_listsets()

## End(Not run)
```

dc_search

Use the DataCite RESTful API for search, facet, stats, and mlt queries.

Description

Use the DataCite RESTful API for search, facet, stats, and mlt queries.

Usage

```
dc_search(..., verbose = TRUE, errors = "simple", proxy = NULL,
  callopts = list())
```

```
dc_facet(..., verbose = TRUE, errors = "simple", proxy = NULL,
  callopts = list())
```

```
dc_stats(..., verbose = TRUE, errors = "simple", proxy = NULL,
  callopts = list())
```

```
dc_mlt(..., verbose = TRUE, errors = "simple", proxy = NULL,
  callopts = list())
```

Arguments

...	Solr parameters passed on to the respective <code>solrium</code> package function.
verbose	(logical) Whether to print help messages or not. E.g., if TRUE, we print the URL on each request to a Solr server for your reference. Default: TRUE
errors	(character) One of simple or complete. Simple gives http code and error message on an error, while complete gives both http code and error message, and stack trace, if available.
proxy	List of arguments for a proxy connection, including one or more of: url, port, username, password, and auth. See use_proxy for help, which is used to construct the proxy connection.
callopts	Further args passed on to GET

Details

See the `solrium` package documentation for available parameters. For each of `dc_search`, `dc_facet`, `dc_stats`, and `dc_mlt` see the equivalently named function in `solrium`:

- `dc_search` - [solr_search](#)
- `dc_facet` - [solr_facet](#)
- `dc_stats` - [solr_stats](#)
- `dc_mlt` - [solr_mlt](#)

Fields you can search on and retrieve back are:

- doi
- creator
- publisher
- publicationYear
- title
- alternateIdentifier
- contributor
- date
- description
- format
- language
- relatedIdentifier
- resourceType
- resourceTypeGeneral
- rights
- size
- subject
- version
- allocator symbol of the DataCite member (e.g. BL = British Library)
- datacentre symbol of the datacentre, which uploaded the metadata (e.g. BL.ADS = Archeology Data Service)
- indexed datestamp of indexing
- prefix DOI prefix (e.g. 10.5284)
- refQuality reference quality flag (1 or 0)
- uploaded datestamp of metadata uploading

If you want to use `wt='csv'` you'll need `solrium` package version 0.1.8.99 or greater. Installation instructions at <https://github.com/ropensci/solrium#quick-start>.

Examples

```
## Not run:
# Search
## various searches
### containing laser
dc_search(q = "laser", fl=c('doi','publicationYear'), rows=5)

### containing laser and safety
dc_search(q = "laser safety", fl=c('doi','publicationYear'), rows=5)

### containing laser and not medicine
dc_search(q = "laser -medicine", fl=c('doi','publicationYear'), rows=5)
```



```

### containing laser and red or green
dc_search(q = "laser AND (red OR green)", fl=c('doi','publicationYear'), rows=5)

### containing wind turbine as a phrase in the title
dc_search(q = 'title:"wind turbine"', fl=c('doi','title'), rows=5)

### containing geology in metadata field subject
dc_search(q = "subject:geology", fl=c('doi','subject'), rows=5)

### belonging to DataCite members BL or TIB
dc_search(q = "allocator:(BL OR TIB)", fl=c('doi','allocator'), rows=5)

### published between 2000 and 2005
dc_search(q = "publicationYear:[2000 TO 2005]", fl=c('doi','publicationYear'), rows=5)

### uploaded to DataCite in the last 5 days
dc_search(q = "uploaded:[NOW-5DAYS TO NOW]", fl=c('doi','uploaded'), rows=5)

## search with csv output
### If you want to use wt=csv, install development version of solr package, see Details
dc_search(q = 'wind', fl=c('doi','title'), wt='csv')

# Faceting
dc_facet(q = "wind", facet.field='publisher_facet')

# Stats
dc_stats(q = "ecology", stats.field='date')

# More like this (aka mlt)
dc_mlt(q = "ecology", mlt.fl='title', mlt.count=5, fl=c('doi','title'))

## End(Not run)

```

Prefixes

Metadata Prefixes

Description

These three options can be used in `dc_oai_*`(`)` functions as input to the `prefix` parameter.

Details

- `oai_dc` - OAI Dublin Core - As a minimum requirement for OAI-PMH compliance, metadata must be made available in the OAI Dublin Core format. For more information please see the OAI-PMH web site.
- `oai_datacite` - OAI DataCite - This metadata format has been specifically established for the dissemination of DataCite records using OAI-PMH. In addition to the original DataCite metadata, this format contains several other elements describing the version of the metadata, whether it is of reference quality, and the registering datacentre. For more information about this format and its schema please see the Datacite OAI schema web site.

- *datacite* - DataCite Direct - This metadata format contains only the original DataCite metadata without additions or alterations. Because there are multiple versions of DataCite metadata in the MDS, there is no one schema that they will all adhere to. Therefore the schema for this format does not exist and metadata will not validate against it. Please note that this format is not OAI-PMH version 2.0 compliant for the previously stated reasons.

Index

*Topic **package**

rdaticite-package, 2

dc_facet, 2

dc_facet (dc_search), 7

dc_mlt, 2

dc_mlt (dc_search), 7

dc_oai_getrecord, 2, 3

dc_oai_identify, 2, 3

dc_oai_listidentifiers, 2, 4

dc_oai_listmetadataformats, 2, 5

dc_oai_listrecords, 2, 5

dc_oai_listsets, 2, 6

dc_search, 2, 7

dc_stats, 2

dc_stats (dc_search), 7

GET, 4–7

Prefixes, 9

rdaticite (rdaticite-package), 2

rdaticite-package, 2

solr_facet, 7

solr_mlt, 7

solr_search, 7

solr_stats, 7

use_proxy, 7