Package ‘couchDB’

August 29, 2016

Type Package
Title Connect to and Work with CouchDB Databases
Version 1.4.1
Date 2016-07-25
Depends R (>= 3.0.0), RCurl, bitops, httr, rjson
SystemRequirements couchDB instance to connect to and work with.
License AGPL-3
Suggests testthat
RoxygenNote 5.0.1
NeedsCompilation no
Author Aleksander Dietrichson [aut, cre], Frans van Dunn<c3><a9> [ctb]
Maintainer Aleksander Dietrichson <sasha@xrrch.com>
Repository CRAN
Date/Publication 2016-06-26 09:21:58

R topics documented:
couchDB .............................. 2
couch_attach ........................ 3
couch_create_database .............. 3
couch_delete ........................ 4
couch_fetch ........................ 4
couch_fetch_attachment .......... 5
couch_fetch_attachment_url ..... 5
couch_fetch_default .............. 6
couch_http_connection .......... 6
couch_list_databases .......... 7
couch_list_databases_url .... 7
couch_new_object ............. 7
This package contains functions to interact with couchDB - a document database. For more information about the couchDB database see: http://couchdb.apache.org. To interact with a couchDB instance you will need to create a connection object. Communication with couchDB happens over a http protocol. A minimal connection (running on the default port and no password protection) would be:

```r
myconn <- couch_http_connection("localhost")
```

The variable "myConn" can now be used as parameters to other functions.

For convenience a default connection can also be created with `couch_set_default_connection` using the same parameters.

Once a connection object exists, you may want to make sure it is connecting correctly with the `couch_ping` function. If you are properly connected the response should be like:

```r
text/plain
size: 151 B
{"couchdb": "Welcome","uuid": "c1a367c91517195b57ddafe788a72b75","version": "1.4.0","vendor": "...
```

## Databases

On a couchDB install there can be several databases (i.e. namespaces). The function `couch_list_databases` will provide a list of available databases on the connection provided.

The function `couch_create_database` will, similarly, allow you to create a new database or namespace on the couchDB instance. To fetch, store and delete documents: Once you have a connection and a database to work with you can fetch, store and delete documents by using the corresponding functions:

- `couch_fetch` and `couch_fetch_default`
- `couch_store` and `couch_store_default`
- `couch_delete`
couch_attach

**Author(s)**
Aleksander Dietrichson

---

### couch_attach

**Add attachment to document**

**Description**
Send attachment to an existing url

**Usage**
couch_attach(location, revtag, attachment, content_type = "image/png")

**Arguments**
- `location` : url of the
- `revtag` : revision (version)
- `attachment` : file to attach
- `content_type` : Content type of the attachment (for example: "image/png")

---

### couch_create_database

**Create database**

**Description**
Creates a new database based on the dbname.
Creates a new couchDB database on the connection provided.

**Usage**
couch_create_database(conn, dbname)

**Arguments**
- `conn` : a couchDB connection object
- `dbname` : the name of the database

**Examples**

```r
## Not run:
#Note: this example assumes that there is a couchDB instance available on localhost
myConn <- couch_http_connection("localhost")
couch_create_database(myConn,"myDatabase")

## End(Not run)
```
### couch_delete

*Delete a record.*

**Description**

Delete a record on the connection provided

**Usage**

```couch_delete(conn, database, key, myOpts = NULL)```

**Arguments**

- `conn`: A couchDB connection object
- `database`: Name of database to operate on
- `key`: key (record) to delete
- `myOpts`: Additional options (not implemented in this version)

### couch_fetch

*Fetch a document/record.*

**Description**

Fetches a couch object based on the key

**Usage**

```couch_fetch(conn, database, key, myOpts = NULL)```

**Arguments**

- `conn`: A couchDB connection object
- `database`: The database to connect to.
- `key`: Key of document to fetch
- `myOpts`: Additional options (not implemented in this version)

**Value**

A list object with the values from the record.
**couch_fetch_attachment**

*Fetch attachment*

**Description**

Gets a named attachment from a resource

**Usage**

```c
couch_fetch_attachment(conn, database, key, attachment)
```

**Arguments**

- **conn**: A couchDB connection object
- **database**: The database to connect to
- **key**: Key of document
- **attachment**: Name of attachment

**couch_fetch_attachment_url**

*Get attachment url*

**Description**

Get the url for a specific attachment. This is sometimes useful for direct reads to functions, in lieu of storing tempfiles.

**Usage**

```c
couch_fetch_attachment_url(conn, database, key = NULL, attachment = NULL)
```

**Arguments**

- **conn**: A connection object
- **database**: The database name
- **key**: Document key
- **attachment**: Name of the attachment
couch_fetch_default  Fetch document/record from default store.

Description
Fetches a document specified by Key from the default database on the default connection

Usage
couch_fetch_default(key, myOpts = NULL)

Arguments
key The key of the document to fetch
myOpts Additional options (not implemented in this version)

Value
A list object with the values from the record.

couch_http_connection  Connection to couchDB

Description
Creates a connection object on the host and ports provided

Usage
couch_http_connection(host, port = 5984, https = FALSE, service = "couchdb", user = NULL, password = NULL)

Arguments
host The IP address of the couchDB instance
port The port to connect to
https Should a ssl protocol be used
service the service used accepts either "couchdb" or "cloudant"
user Username on the database server
password Password for the database server

Examples
## Not run:
myConn <- couch_http_connection(host = "localhost")

## End(Not run)
**couch_list_databases**

List available databases.

**Description**

Lists the available databases on the connection provided.

**Usage**

`couch_list_databases(conn)`

**Arguments**

- **conn**
  - A couchDB connection object

**couch_list_databases_url**

`couch_list_databases_url`

**Description**

Format the url for fetching database-list for internal use

**Usage**

`couch_list_databases_url(conn)`

**Arguments**

- **conn**
  - A couchDB connection object.

**couch_new_object**

New couchDB document

**Description**

Creates a new object to insert to the couchDB. Takes either a list or a formatted json object as value. Any attachment to the record needs to be base64-encoded added to the list as "_attachments". If key is provided this is used, null sends a key-less record to couch and the key will have to be retrieved from the response object.

**Usage**

`couch_new_object(value, database = NULL, key = NULL)`
couch_set_default_connection

Arguments

- value: List to be converted to json for transmission or pre-formatted JSON string
- database: The database to use
- key: The key (recordname) to use for the object.

Examples

## Not run:

# This code creates a document containing a small list for storage in
# the "localhost" database with the key "testDoc".
myDoc <- couch_new_object(list(a = 1, b = 2), "localhost", "testDoc")

# Same as above but with json entered directly (not recommended).
myDoc <- couch_new_object('{"a":1,"b":2}', "localhost", "testDoc")

## End(Not run)

couch_ping

**Ping connection**

Description

Check connection to the database.

Usage

couch_ping(conn)

Arguments

- conn: A couchDB connection object.

couch_set_default_connection

**Set default connection**

Description

Sets up a couchDB connection to use as default

Usage

couch_set_default_connection(host, port = 5984, https = FALSE)
couch_set_default_database

Arguments

- **host**: The IP address of the couchDB instance
- **port**: The port to connect to
- **https**: Should a ssl protocol be used?

---

couch_set_default_database

*Set a database as default document store*

**Description**

Specifies a database to write to on a couch connection by default.

**Usage**

couch_set_default_database(database)

**Arguments**

- **database**: the database to use as default (String)

---

couch_store

*Store a record*

**Description**

Stores a record to the connection provided (database spec is in object)

**Usage**

couch_store(conn, obj, myOpts = NULL)

**Arguments**

- **conn**: A couchDB connection object
- **obj**: A list formatted by calling couch_new_object
- **myOpts**: Additional options (not implemented in this version)
**Description**

Stores a record on the default connection.

**Usage**

```couch_store_default(obj, myOpts = NULL)```

**Arguments**

- `obj`: A list formatted by calling `couch_new_object`.
- `myOpts`: Additional options (not implemented in this version)

---

**print.couch_connection**

*Print method for couchDB connection object*

**Description**

Prints the couchDB connection object.

**Usage**

```#
S3 method for class 'couch_connection'
print(conn)
```

**Arguments**

- `conn`: a couchDB connection object
Index

couch_attach, 3

couch_create_database, 2, 3

couch_delete, 2, 4

couch_fetch, 2, 4

couch_fetch_attachment, 5

couch_fetch_attachment_url, 5

couch_fetch_default, 2, 6

couch_http_connection, 6

couch_list_databases, 2, 7

couch_list_databases_url, 7

couch_new_object, 7

couch_ping, 2, 8

couch_set_default_connection, 2, 8

couch_set_default_database, 9

couch_store, 2, 9

couch_store_default, 2, 10

couchDB, 2

couchDB-package (couchDB), 2

print.couch_connection, 10