

Package ‘R4CouchDB’

March 14, 2010

Type Package

Title Collection of R functions for CouchDB access

Version 0.04

Date 2010/03/14

Author wactbprot

Maintainer <thsteinbock@web.de>

Description The R4CouchDB package provides a collection of functions for basic database and document management operations such as add and delete. Every cdbFunction() gets and emits a list containing the connection set up. A unit test structure using the RUnit package is preliminary prepared. With the cdbIni() function the cdb\$serverName is set to “localhost” as a default. This means that localhost should provide a CouchDB. All (former) example sections in the man pages are removed; the unit test structure is intended to cover the example task with more and easier control.

License BSD

LazyLoad yes

Depends R (>= 2.7.0), RCurl, bitops, RJSONIO

R topics documented:

R4CouchDB-package	2
cdbAddAttachment	2
cdbAddDoc	3
cdbDeleteDoc	4
cdbGetDoc	4
cdbGetUuids	5
cdbGetView	6
cdbIni	7
cdbListDB	7
cdbMakeDB	8
cdbRemoveDB	9
cdbUpdateDoc	9

R4CouchDB-package *Collection of R functions for CouchDB access*

Description

The R4CouchDB package provides a collection of functions for basic database and document management operations such as add and delete.

Details

Package:	R4CouchDB
Type:	Package
Version:	0.1
Date:	2009-08-29
License:	BSD
LazyLoad:	yes

Every `cdbFunction()` gets and emits a list containing the connection set up. Internally this `list()` is called `cdb`. The result of requests, the CouchDB answers are stored in the `cdb$res`.

The function `cdbIni()` initially provides `cdb`. However `a <- list(serverName="localhost", port="5984")` is sufficient for calling `cdbGetUids(a)`.

Since it is not possible to provide a meaningful default `cdb$serverName` all example sections are removed.

The http connection and the json transformations are done with Duncan Temple Langs RCurl and RJSON.

Author(s)

wact.b.prot

Maintainer: thsteinbock@web.de

References

<http://www.omegahat.org/RCurl/> <http://www.omegahat.org/RJSONIO/> <http://couchdb.apache.org/>

cdbAddAttachment *Add Attachments*

Description

This function adds attachments to a doc

Usage

```
cdbAddAttachment(cdb)
```

Arguments

`cdb` The list `cdb` has to contain `cdb$fileName`, `cdb$serverName`, `cdb$DBName` and a `cdb$id`.

Details

The function uses a simple call to `strsplit(fileName, ".")` and takes the last entry of the resulting vector as the `Content-Type` in the `httpheader`

Value

`cdb` The result is stored in `cdb$res`

Author(s)

wact.b.prot thsteinbock@web.de

<code>cdbAddDoc</code>	<i>Generates a new document</i>
------------------------	---------------------------------

Description

This function adds a new document to an already existing database

Usage

```
cdbAddDoc(cdb)
```

Arguments

`cdb` The list `cdb` only has to contain a `cdb$dataList` which is not an empty `list()`.

Details

This function is called `addDoc` (which means add a new document). Therefore the `cdb$id` is requested using `cdbGetUuid()` for every document to add if no `cdb$id` is provided. If a `cdb$id` is provided the function fails when a document with the given id already exists. In this case one should use `cdbUpdateDoc()`.

Value

`cdb` The result is stored in `cdb$res`

Author(s)

wact.b.prot thsteinbock@web.de

See Also

`cdbGetDoc()`

<code>cdbDeleteDoc</code>	<i>Deletes a document from a database</i>
---------------------------	---

Description

With a given `cdb$id` this function sends a http "DELETE" request to the url `.../cdb$id?rev=cdb$rev`.

Usage

```
cdbDeleteDoc(cdb)
```

Arguments

<code>cdb</code>	Beside <code>cdb\$serverName</code> , <code>cdb\$port</code> and <code>cdb\$DBName</code> the <code>cdb\$id</code> must be given. R errors are reported in <code>cdb\$errors</code>
------------------	---

Value

<code>cdb</code>	The result of the delete request is stored in <code>cdb\$res</code> (whatever this means).
------------------	--

Author(s)

wact.b.prot thsteinbock@web.de

References

<http://couchdb.apache.org/> <http://www.omegahat.org/>

See Also

`cdbAddDoc()`

<code>cdbGetDoc</code>	<i>Get a doc from CouchDB</i>
------------------------	-------------------------------

Description

With a given `cdb$id` this function requests the document.

Usage

```
cdbGetDoc(cdb)
```

Arguments

<code>cdb</code>	Beside <code>cdb\$serverName</code> , <code>cdb\$port</code> and <code>cdb\$DBName</code> the <code>cdb\$id</code> must be given. R errors are reported in <code>cdb\$errors</code>
------------------	---

Value

`cdb` The result of the request is stored in `cdb$res` after converting the answer into a list using `fromJSON()`. If a list entry needed in `cdb` is not provided `cdb$error` gives some information.

Author(s)

wact.b.prot thsteinbock@web.de

References

<http://couchdb.apache.org/> <http://www.omegahat.org/>

See Also

`cdbAddDoc()`

<code>cdbGetUuids</code>	<i>Function for id generation</i>
--------------------------	-----------------------------------

Description

Function returns a 128bit uuid requested from CouchDB

Arguments

`cdb` Only the connection settings `cdb$port` and `cdb$serverName` is needed.

Details

Simple CouchDB API end point to `http://serverName:port/_uuids`. The number `N` of ids received from a CouchDB can be increased by `cdb$queryParam <- paste("count=", N, sep="")`. For the moment the functions using ids can handle only `N=1`!

Value

`cdb` The result of the request is stored in `cdb$id` after converting the answer into a list using `fromJSON()`.

Author(s)

wact.b.prot thsteinbock@web.de

References

<http://www.omegahat.org/Rcurl/> <http://www.omegahat.org/RJSONIO/> <http://couchdb.apache.org/>

See Also

`cdbMakeDB`

`cdbGetView`*Receive view results from CouchDB*

Description

The function provides accesses to CouchDB views.

Usage

```
cdbGetView(cdb)
```

Arguments

<code>cdb</code>	Beside the connection details (<code>cdb\$port</code> , <code>cdb\$DName</code> ...) the <code>cdb\$design</code> and <code>cdb\$view</code> is needed.
------------------	--

Details

Query params e.g. `"reduce=false"` or `"group_level=1"` can be provided in `cdb$queryParam`

Value

<code>cdb</code>	The result of the request is stored in <code>cdb\$res</code> after converting the json answer into a list using <code>fromJSON()</code> . If a needed <code>cdb</code> list entry was not provided <code>cdb\$error</code> says something about the R side
------------------	--

Note

For the moment only one `cdb$queryParam` is possible. In the future maybe Duncans `RJavaScript` package can be used to generate views without leaving R.

Author(s)

wact.b.prot thsteinbock@web.de

References

<http://www.omegahat.org/RCurl/> <http://www.omegahat.org/RJSONIO/> <http://couchdb.apache.org/> <http://www.omegahat.org/RJavaScript/>

<i>cdbIni</i>	<i>Ini function</i>
---------------	---------------------

Description

Function returns a list with some default settings

Usage

```
cdbIni()
```

Details

The list: `cdb <- list(serverName = "", port = "5984", curl=getCurlHandle(), DBName="", newDBName="", id="", rev="", dataList=list(), design="", view="", queryParam="", date=toString(Sys.Date()), error="", res="")` is returned if the packages `library(RCurl)` and `library(RJSONIO)` are successfully loaded.

Value

<code>cdb</code>	see details for the default settings of this function
------------------	---

Author(s)

wact.b.prot thsteinbock@web.de

<i>cdbListDB</i>	<i>Returns all databases</i>
------------------	------------------------------

Description

Gives a list of all databases

Usage

```
cdbListDB(cdb)
```

Arguments

<code>cdb</code>	Only the connection settings <code>cdb\$port</code> and <code>cdb\$serverName</code> is needed.
------------------	---

Details

Simple CouchDB API end point: http://localhost:5984/_all_dbs

Value

<code>cdb</code>	The result of the request is stored in <code>cdb\$res</code> after converting the json answer into a list using <code>fromJSON()</code> .
------------------	---

Author(s)

wact.b.prot thsteinbock@web.de

References

<http://www.omegahat.org/RCurl/> <http://www.omegahat.org/RJSONIO/> <http://couchdb.apache.org/>

See Also

cdbMakeDB

cdbMakeDB

Creates a new database

Description

The name of the new database is taken from `cdb$newDBName`.

Usage

```
cdbMakeDB (cdb)
```

Arguments

`cdb` The `cdb` have to provide `cdb$serverName`, `cdb$port` and `cdb$newDBName`

Details

The work is done by `getURLContent()`. After creating the new database the function makes the shortcut `cdb$DBName <- cdb$newDBName` so that further operations happen on the new created database.

Value

`cdb` The CouchDB answer is stored in `cdb$res`. Any problems on the R side are reportet in `cdb$error`

Note

The convention for database naming should be implemented.

Author(s)

wact.b.prot thsteinbock@web.de

References

<http://couchdb.apache.org/>

See Also

cdbRemoveDB

cdbRemoveDB	<i>Function to remove a database.</i>
-------------	---------------------------------------

Description

Removing a database means sending a http- "DELETE"- request to `http://cdb$serverName:cdb$port/...`

Usage

```
cdbRemoveDB (cdb)
```

Arguments

cdb	The cdb has to provide cdb\$serverName, cdb\$port and cdb\$DBName
-----	---

Details

In cdb maybe a entry cdb\$delDBName should be provided for more explicit deleting respectively more secure removing.

Value

cdb	The CouchDB answer is stored in cdb\$res. Any problems on the R side are reportet in cdb\$error
-----	---

Author(s)

wact.b.prot thsteinbock@web.de

References

<http://couchdb.apache.org/>

See Also

cdbMakeDB

cdbUpdateDoc	<i>This function updates an existing doc. This essentially means that a revision, corresponding to the '_id' has to be provided (in what way ever).</i>
--------------	---

Description

Updating a doc at couchdb means executing a http "PUT" request. The cdb list must contain the cdb\$serverName, cdb\$port, cdb\$DBName, cdb\$id. If the cdb\$rev is not given the function uses cdbGetDoc() to receive the latest revision number.

Usage

```
cdbUpdateDoc (cdb)
```

Arguments

<code>cdb</code>	the <code>cdb</code> connection configuration list must contain the <code>cdb\$serverName</code> , <code>cdb\$port</code> , <code>cdb\$DBName</code> and <code>cdb\$id</code> . The data which updates the data stored in the doc is provided in <code>cdb\$dataList</code>
------------------	---

Details

`getURLContent ()` with `customrequest = "PUT"` does the work.

Value

<code>cdb</code>	The result of the request is stored in <code>cdb\$res</code> after converting the answer by means of <code>fromJSON ()</code> . If a needed <code>cdb\$</code> list entry is not provided <code>cdb\$error</code> maybe says something about the R side.
------------------	--

Author(s)

wact.b.prot thsteinbock@web.de

References

<http://www.omegahat.org/RCurl/> <http://www.omegahat.org/RJSONIO/> <http://couchdb.apache.org/>

See Also

`cdbInit ()`