Package ‘R4CouchDB’

November 8, 2009

Type Package

Title Collection of R functions for CouchDB access

Version 0.03

Date 2009/11/08

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 no

Depends R (>= 2.7.0), RCurl, RJSONIO

R topics documented:

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 `cdbGetUuids(a)`.

The Rd-examples in this package are a kind of fake. Every function tests the needed list entries before request execution. Since it is not possible to provide a default `cdb$serverName` all function calls in the example- sections set a `cdb$error <- "no cdb$serverName"`, no http- request is executed.

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/

Examples

```
## get the list structure
cdb <- cdbIni()
## this structure includes:
cdb
## the command
cdb <- cdbListDB(cdb)
## yields
cdb$error
```

`cdbAddDoc` *Generates a new document*

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 fresh for every document to add.

Value

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

Author(s)

wact.b.prot thsteinbock@web.de

See Also

`cdbGetDoc()`

`cdbDeleteDoc` *Deletes a document from a database*

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

`cdb`	Beside `cdb$serverName`, `cdb$port` and `cdb$DBName` the `cdb$id` must be given. R errors are reported in `cdb$errors`

Value

`cdb`	The result of the delete request is stored in `cdb$res`(whatever this means).

Author(s)

wact.b.prot thsteinbock@web.de

References

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

See Also

`cdbAddDoc()`

`cdbGetDoc` *Get a doc from CouchDB*

Description

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

Usage

```
cdbGetDoc(cdb)
```

Arguments

`cdb`	Beside `cdb$serverName`, `cdb$port` and `cdb$DBName` the `cdb$id` must be given. R errors are reported in cdb$errors

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()`

cdbGetUuids *Function for id generation*

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`

Examples

```
cdb <- cdbIni()
cdbGetUuids(cdb)$id
```

cdbGetView *Receive view results from CouchDB*

Description

The function provides accesses to CouchDB views.

Usage

```
cdbGetView(cdb)
```

Arguments

`cdb`	Beside the connection details (`cdb$port`,`cdb$DAName` ...) the `cdb$design` and `cdb$view` is needed.

Details

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

Value

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

Examples

```
## Not run:
## If:
## function(doc) {
##  emit(null, doc._id);
## }
## is stored under design: test, view: id
## in the database db5r

cdb <- cdbIni()
cdb$DBName <- "db5r"
cdb$design <- "test"
```

```
cdb$view <- "id"

## cdbGetView(cdb)$res
```

cdbIni *Ini function*

Description

Function returns a list with some default settings

Usage

```
cdbIni()
```

Details

The list: `cdb <- list( serverName = "", port = "5984", 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

`cdb`	see details for the default settings of this function

Author(s)

wact.b.prot thsteinbock@web.de

Examples

```
d <- cdbIni()
cdbListDB(d)$res
```

cdbListDB *Returns all databases*

Description

Gives a list of all databases

Usage

```
cdbListDB(cdb)
```

Arguments

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

Details

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

Value

cdb	The result of the request is stored in cdb$res after converting the json 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

Examples

```
cdb <- cdbIni()
cdbListDB(cdb)$res
```

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` *Function to remove a database.*

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 — *This function updates an existing doc. This essentially means that a revision, corresponding to the '_id' has to be provided (in what way ever).*

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

`cdb`	the cdb connection configuration list must contain the `cdb$serverName`, `cdb$port`, `cdb$DBName` and `cdb$id`. The data which updates the data stored in the doc is provided in `cdb$dataList`

Details

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

Value

`cdb`	The result of the request is stored in `cdb$res` after converting the answer by means of `fromJSON()`. If a needed `cdb$` list entry is not provided `cdb$error` 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()`

Index