

sofa - easy interface to CouchDB from R

sofa - easy interface to CouchDB from R

What is sofa?

XXXXXXX

It is getting easier to get data directly into R from the web. Often R packages that retrieve data from the web return useful R data structures to users like a `data.frame`. This is a good thing of course to make things user friendly.

However, what if you want to drill down into the data that's returned from a query to a database in R? What if you want to get that nice `data.frame` in R, but you think you may want to look at the raw data later? The raw data from web queries are often JSON or XML data. This type of data, especially JSON, can be easily stored in schemaless so-called NoSQL databases, and queried later.

There are plenty of databases you can interact with from R, so why CouchDB? For one, it makes a lot of sense to write to a NoSQL database when you have JSON, XML, HTML, etc., which aren't a good fit for databases like MySQL, SQLite, PostgreSQL, etc. ([though postgres allows you to write JSON][postgres]). It didn't have to be CouchDB, but at least to me it seems relatively easy to install, you can interact with it via an HTTP API (if you're into that, which I am), and it has a nice web interface (navigate to http://localhost:5984/_utils/ after starting `couchdb`).

A brief aside: JSON and XML

What are JSON and XML? This is what JSON looks like (ps if you ever wonder if your JSON is correct, go [here](#)):

```
{
  "name": "joe",
  "hobby": "codemonkey",
  "lives": [
    {
      "city": "San Jose",
      "state": "CA"
    }
  ]
}
```

This is what XML looks like:

```
<?xml version="1.0" encoding="UTF-8" ?>
  <name>joe</name>
  <hobby>codemonkey</hobby>
```

```
<lives>
  <city>San Jose</city>
  <state>CA</state>
</lives>
```

CouchDB help

- [Apache CouchDB](#).
- [CouchDB guide](#) - online book and hard copy.
- [Couchapp](#) - sort of like rails, for templating apps based on CouchDB.

Quick start

Start CouchDB in your terminal You can do this from anywhere in your directory. See [here](#) for instructions on how to install CouchDB. You can't use **sofa** functions without having couchdb running, either locally or on a remote server. If using a remote sever of course you don't need to have CouchDB running locally.

```
couchdb
```

Install sofa

```
# install.packages('devtools'); library(devtools);
# install_github('rbison', 'ropensci')
library(sofa)
```

Start CouchDB on your command line by typing 'couchdb'

Then start Elasticsearch if using by opening a new terminal tab/window, navigating to where it was installed and starting

On my Mac this is: `cd /usr/local/elasticsearch` then `bin/elasticsearch -f`

New to sofa? Tutorial at <https://github.com/schamberlain/sofa>.

Use `suppressPackageStartupMessages()` to suppress these startup messages in the future

Ping your local server just to check your head

```
sofa_ping()
```

```
$couchdb
[1] "Welcome"
```

```
$uuid
[1] "6524cce1bf020f7b8fc74616b57f09ff"
```

```
$version
[1] "1.3.0"
```

```
$vendor
$vendor$version
[1] "1.3.0-1"
```

```
$vendor$name
[1] "Homebrew"
```

Create a new database

```
sofa_createdb("helloworld")
```

```
$ok
[1] TRUE
```

List databases

```
sofa_listdbs()

[1] "_replicator"          "_users"
[3] "alm_couchdb"          "alm_db"
[5] "cheese"               "deldbtesting"
[7] "dudedb"               "example"
[9] "foobar"               "foodb"
[11] "getattachtesting"     "hello_earth"
[13] "hello_world"          "helloworld"
[15] "rplos_db"             "shit"
[17] "shitty"               "shitty2"
[19] "sofadb"               "test_suite_db"
[21] "test_suite_db/with_slashes" "test_suite_reports"
[23] "testr2couch"          "twitter_db"
```

More in depth stuff

Working with documents

Working with a remote database

Attachements

The changes feed

Full text search

Start elasticsearch in your terminal See [here](#) for instructions on how to install Elasticsearch and the River CouchDB plugin.

```
cd /usr/local/elasticsearch
bin/elasticsearch -f
```