

Apache CouchDB

Myles Braithwaite

me@mylesb.ca | <http://mylesb.ca> | @mylesb

What is CouchDB?

Apache CouchDB is a database
that uses **JSON** for documents,
JavaScript for **MapReduce**
indexes, and regular **HTTP** for its
API.

Django may be built *for* the Web,
but **CouchDB** is built *of* the
Web.

— Jacob Kaplan-Moss, *Django Developer*

Document Based Key/Value Store

Unlike a relational database (i.e. Postgres & MySQL), CouchDB doesn't store its data and relationships in tables. Instead, each database is a collection of independent JSON documents.

Data Types

- `"string": "abcdefghijklmnopqrstuvwxyz"`
- `"number": 123`
- `"float": 123.45`
- `"dict": {}`
- `"list": []`
- `"bool": true`

```
{
  "_id": "30b0ed91384411e4af34c42c03094720",
  "_rev": "1-3573aeb5384411e4b121c42c03094720",
  "name": {
    "given_name": "Myles",
    "family_name": "Braithwaite"
  },
  "emails": [
    {
      "type": "personal",
      "email": "me@mylesb.ca"
    },
    {
      "type": "work",
      "email": "myles@monkeyinyoursoul.com"
    }
  ]
}
```

HTTP Based API for Interacting with your Data

- **Create** = INSERT = PUT
- **Retrieve** = SELECT = GET
- **Update** = UPDATE = POST
- **Delete** = DELETE = DELETE

Examples

- Written in Python using the Kenneth Reitz's *requests* library.

```
from myles_custom_urllib_parse import urljoin
```

```
from requests import get, post, put, delete, request
```

```
COUCHDB_URL = "http://127.0.0.1:5984/"
```

```
DB_NAME = "contacts"
```

```
r = requests.get(COUCHDB_URL)

print r.json()

{
    "couchdb": "Welcome",
    "uuid": "f9d2966e384711e499cfc42c03094720",
    "vendor": {
        "name": "The Apache Software Foundation",
        "version": "1.4.0"
    },
    "version": "1.4.0"
}
```

Create a Database

```
r = put(urljoin(COUCHDB_URL, DB_NAME))

if not r.status_code == 201:
    print ERROR_RESPONSE[r.status_code]

print r.json()

{"ok": true}
```

Create

```
data = {"name": {"first_name": "Myles", "last_name": "Braithwaite"}}

r = post(urljoin(COUCHDB_URL, DB_NAME), data)

if not r.status_code == 201:
    print ERROR_RESPONSE[r.status_code]

print r.json()

{"ok": true, "id": "30b0ed91384411e4af34c42c03094720", "rev": "1-3573aeb5384411e4b121c42c03094720"}
```

Create *(with a non-automatic ID)*

```
data = {"name": {"first_name": "Myles", "last_name": "Braithwaite"}}
DOC_ID = "9999-myles-braithwaite"
r = put(urljoin(COUCHDB_URL, DB_NAME, DOC_ID), data)

if not r.status_code == 201:
    print ERROR_RESPONSE[r.status_code]

print r.json()

{"ok": true, "id": "30b0ed91384411e4af34c42c03094720", "rev": "1-3573aeb5384411e4b121c42c03094720"}
```

Retrieve

```
DOC_ID = "30b0ed91384411e4af34c42c03094720"

r = get(urljoin(COUCHDB_URL, DB_NAME, DOC_ID))

if not r.ok:
    print ERROR_RESPONSE[r.status_code]

r.json()
```

Update

```
DOC_ID = "30b0ed91384411e4af34c42c03094720"

r = get(urljoin(COUCHDB_URL, DB_NAME, DOC_ID))
data = r.json()

data['emails']: [
    {
        "type": "Personal", "email": "me@mylesb.ca",
        "type": "Work", "email": "myles@miys.net"
    }
]

r = post(urljoin(COUCHDB_URL, DB_NAME, DOC_ID), data)

if r.ok:
    print ERROR_RESPONSE[r.status_code]

r.json()
```

```
{"ok": true, "id": "30b0ed91384411e4af34c42c03094720", "rev": "2-f57cf4d1384a11e4a3edc42c03094720"}
```

Delete

```
DOC_ID = "myles-braithwaite"
```

```
r = delete(  
    urljoin(COUCHDB_URL, DB_NAME, DOC_ID) + "?rev=%s" % DOC_REV)  
)
```


Attachment

```
curl -vX \
  PUT $COUCHDB_URL/$DB_NAME/$DOC_ID/headshot.jpg?rev=$REV \
  --data-binary @avatar.jpg -H "Content-Type: image/jpeg"
```

Copy

```
r = request(  
    'COPY',  
    urljoin(COUCHDB_URL, DB_NAME, DOC_ID),  
    {'destination': 'new-document'}  
)
```

Other Features

- Replication
- Document Revisions
- Futon (*similar to PHPMyAdmin*)
- Auth (*Basic, Cookie, Database*)
- JavaScript based Map/Reduce

PouchDB

JavaScript clone of CouchDB that
can run well within a web
browser.

```
var db = new PouchDB('contacts');

db.put({
  _id: 'myles-braithwaite',
  first_name: 'Myles',
  last_name: 'Braithwaite'
})

db.replicate.to('http://127.0.0.1:5984/contacts/')
```

Questions