



CouchDB

Content

- Introduction
- System/Operations
- Features
- Applications

Problem Statement

The image is a screenshot of a Facebook post from the University of Florida. The post is dated September 3 at 9:29pm. The text of the post reads: "Since 1970 we've lost half of our wildlife. We're working to protect what remains. For the #GatorGood." Below the text is a video thumbnail showing a close-up of a person's hand holding a camera lens, with the text "because the earth is our only home." overlaid. The post has 29K views, 746 likes, and 150 shares. The right sidebar shows a list of notes, including "Four things you didn't know about elephants" and "In defense of 'the world's ugliest color'", and a list of pages liked by this page, including Florida Gators, University of Florida Alumni Association, and University of Florida Alumni.

UF UNIVERSITY of FLORIDA

University of Florida ✓
@uforida

Home
Photos
Posts
Videos
Events
Likes
About
Notes

Create a Page

UF University of Florida
September 3 at 9:29pm · 🌐

Since 1970 we've lost half of our wildlife.
We're working to protect what remains.
For the #GatorGood.

because the earth is our only home.

29K Views

Like Comment Share

746 Top Comments

150 shares

Write a comment..

UF University of Florida Learn more:
<http://gatorgood.ufl.edu/faq/> biodiversity and the earth!

NOTES

- Four things you didn't know about elephants
August 12
- In defense of 'the world's ugliest color'
June 10
- University of Florida to launch its first Innovation ...
March 21

LIKED BY THIS PAGE

- Florida Gators ✓ Like
- University of Florida Alumni Association ✓ Like
- University of Florida Alumni Association ✓ Like

English (US) · Español · Português (Brasil) · Français (France) · Deutsch

Unstructured data example

The image shows a screenshot of a Twitter profile for Florida (@UF). The profile header includes the name "FLORIDA" with a verified badge, the handle "@UF", and statistics: 33.2K tweets, 371 following, 129K followers, 23.5K likes, and 1 list. The profile was joined in June 2009 and has 2,180 photos and videos. A tweet from "Gators Football" (@GatorsFB) is highlighted, retweeted by Florida. The tweet text is "Coming to #UKvsUF? This should help with planning your day. #WearOrange". The main content of the tweet is a promotional graphic for a football game on Saturday, September 10, 2016, titled "FLORIDA VS. KENTUCKY". The graphic features a schedule of events for the day, including tailgating, fan festivals, and game-related activities. At the bottom of the graphic, it says "THE SWAMP IS BACK" with the hashtag #SWAMP16. The right sidebar shows a list of related accounts and trending topics.

Home Moments Search Twitter Have an account? Log in

FLORIDA @UF
TWEETS 33.2K FOLLOWING 371 FOLLOWERS 129K LIKES 23.5K LISTS 1
Joined June 2009
2,180 Photos and videos

FLORIDA Retweeted
Gators Football @GatorsFB · 58m
Coming to #UKvsUF? This should help with planning your day.
#WearOrange

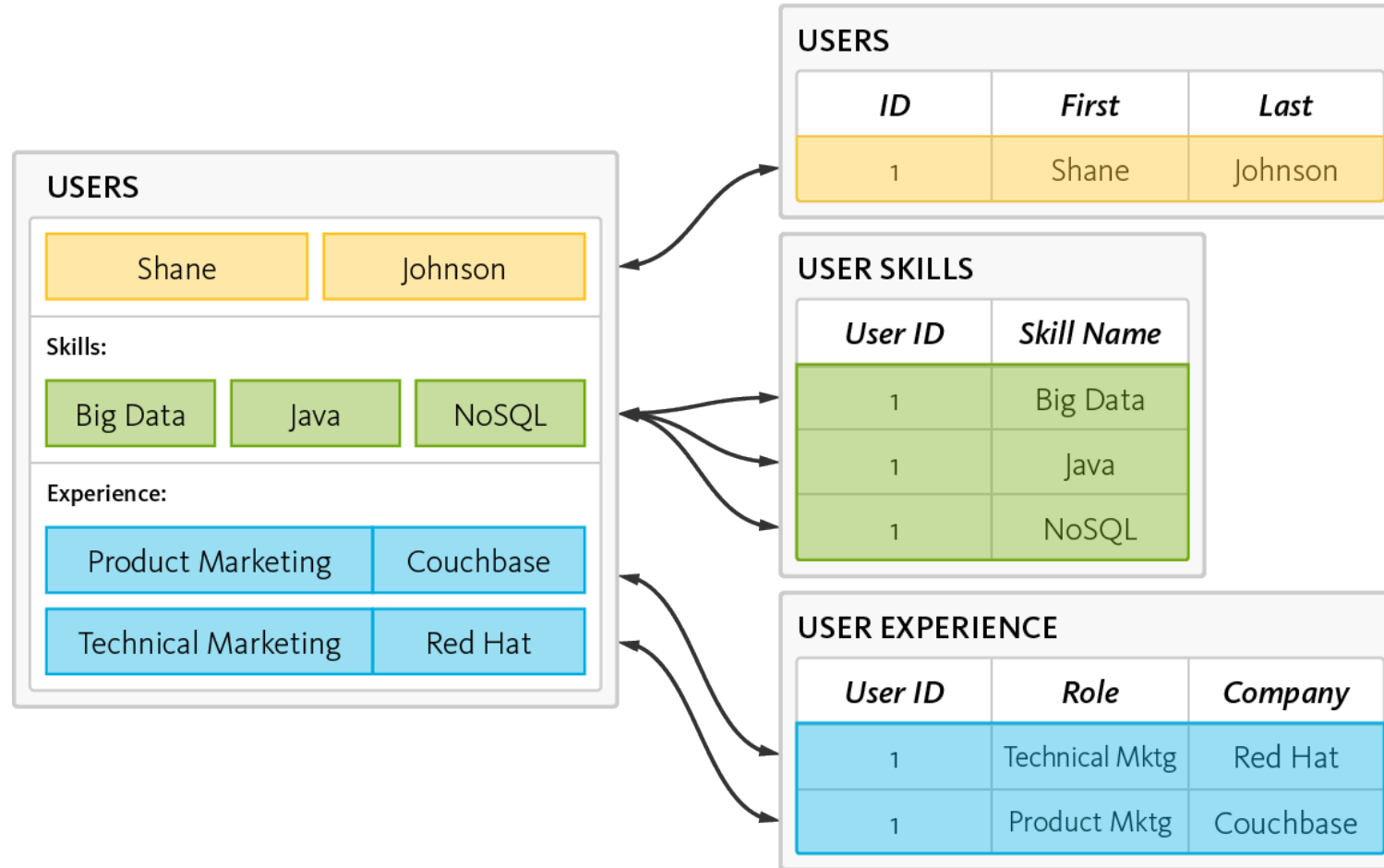
FLORIDA VS. KENTUCKY
SATURDAY, SEPTEMBER 10, 2016
WEAR ORANGE

ALL DAY	TAILGATE WITH GATOR NATION	
12:30 PM	CHECK OUT FAN FEST	O'CONNELL CENTER GATE Z
1:05 PM	HIGH FIVE THE GATORS AT #GATORWALK	THE SWAMP NORTH LAWN
2:00 PM	GATES OPEN - TELL THE WORLD! #UKvsUF	
3:30 PM	MR. TWO BITS WITH THE '06-'07 BASKETBALL TEAM!	
3:35 PM	"ONLY GATORS GET OUT ALIVE!"	
3:39 PM	KICKOFF FOR SEC OPENER	
3rd Q	WE ARE THE BOYS	
4th Q	IT'S GREAT! TO BE! A FLORIDA GATOR!	

THE SWAMP IS BACK
#SWAMP16

Trends
#boycottNFL 45.5K Tweets
#WING 55 128K Tweets
#FridayFeeling 59.7K Tweets
#FlashbackFriday 11.3K Tweets
#SNLaSong Trending for 2 hours now
Russian TV 21K Tweets
Nas Album Done Trending for 2 hours now
Free Fire Trending for 2 hours now
NOCTURNAL ANIMALS Just started trending
Louie Gohmert Just started trending

Complex relationships example

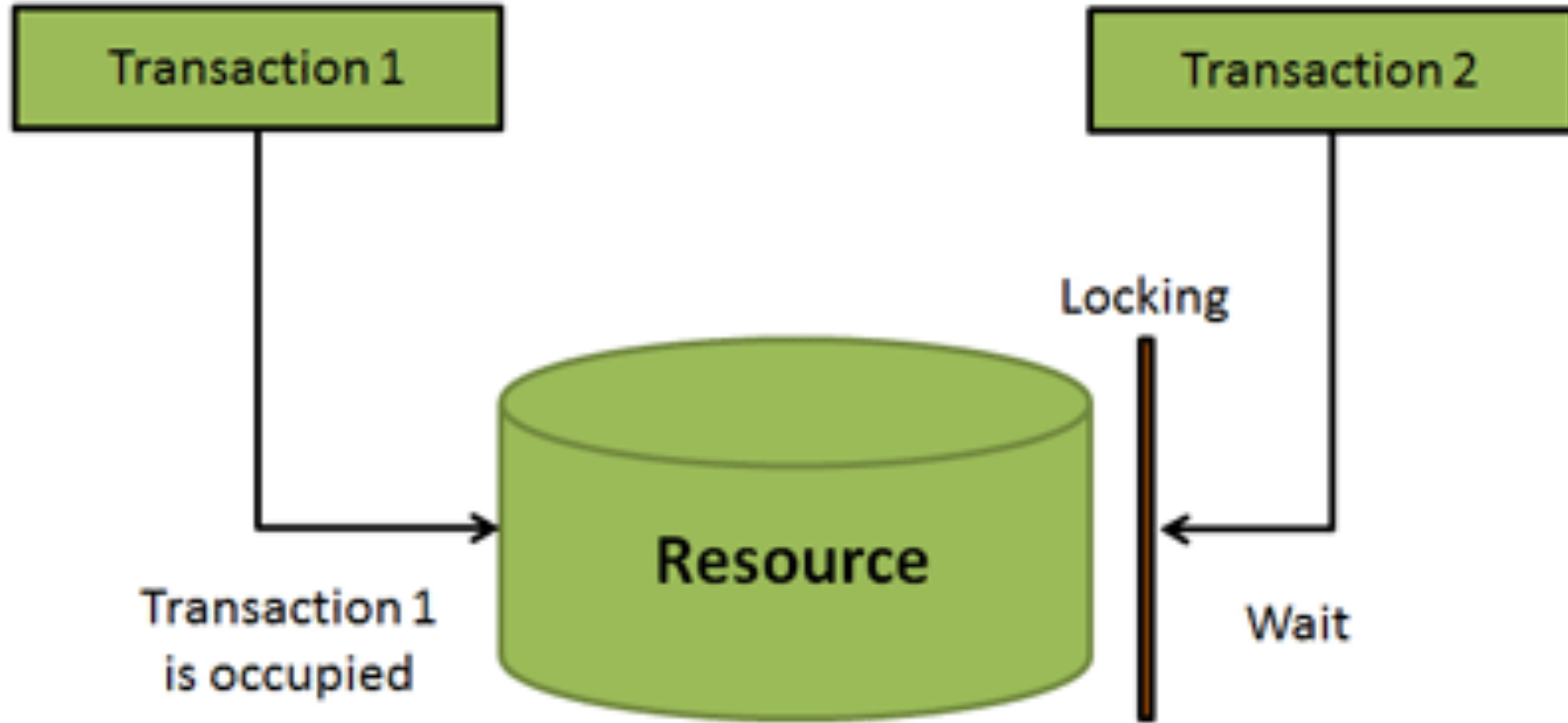


Object to relational in RDBMS

Shane	Johnson	Big Data	Product Marketing	Couchbase
Shane	Johnson	Big Data	Technical Marketing	Red Hat
Shane	Johnson	Java	Product Marketing	Couchbase
Shane	Johnson	Java	Technical Marketing	Red Hat
Shane	Johnson	NoSQL	Product Marketing	Couchbase
Shane	Johnson	NoSQL	Technical Marketing	Red Hat

Object to relational in CouchDB

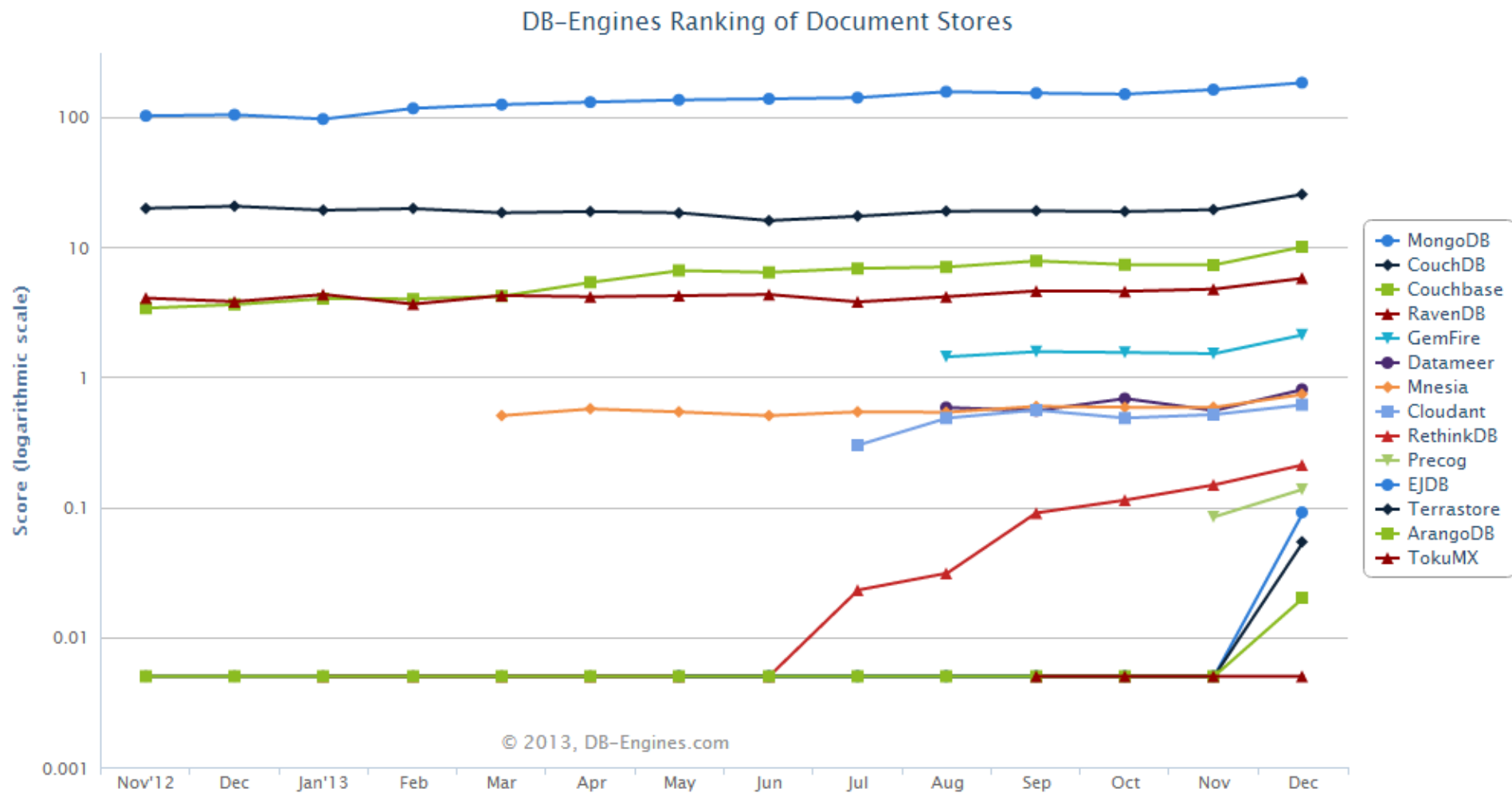
Multiversion Concurrency Control



Introduction to CouchDB

- Opensource DB by Apache
- NoSQL DB
- Document store
- Focus : Ease of use, embracing the web
- Written in Erlang programming lang
- JSON : Store data,
JavaScript : Query Lang,
HTTP : Access docs, query indices with web browser

Popularity Index



Flexible schema example

Iteration 1 — First, Last

Schema Utilized

USERS		
ID	First	Last

Brendan	Bond
---------	------



USERS		
0	Brendan	Bond

Iteration 2 — First, Last, *Twitter*

Schema Utilized

USERS		
ID	First	Last

Shane	Johnson	@shane_dev
-------	---------	------------



USERS		
0	Brendan	Bond
1	Shane	Johnson



Explicit schema in RDBMS

Iteration 1 — First, Last

Brendan

Bond

```
{  
  "firstName": "Brendan",  
  "lastName": "Bond"  
}
```

Iteration 2 — First, Last, *Twitter*

Shane

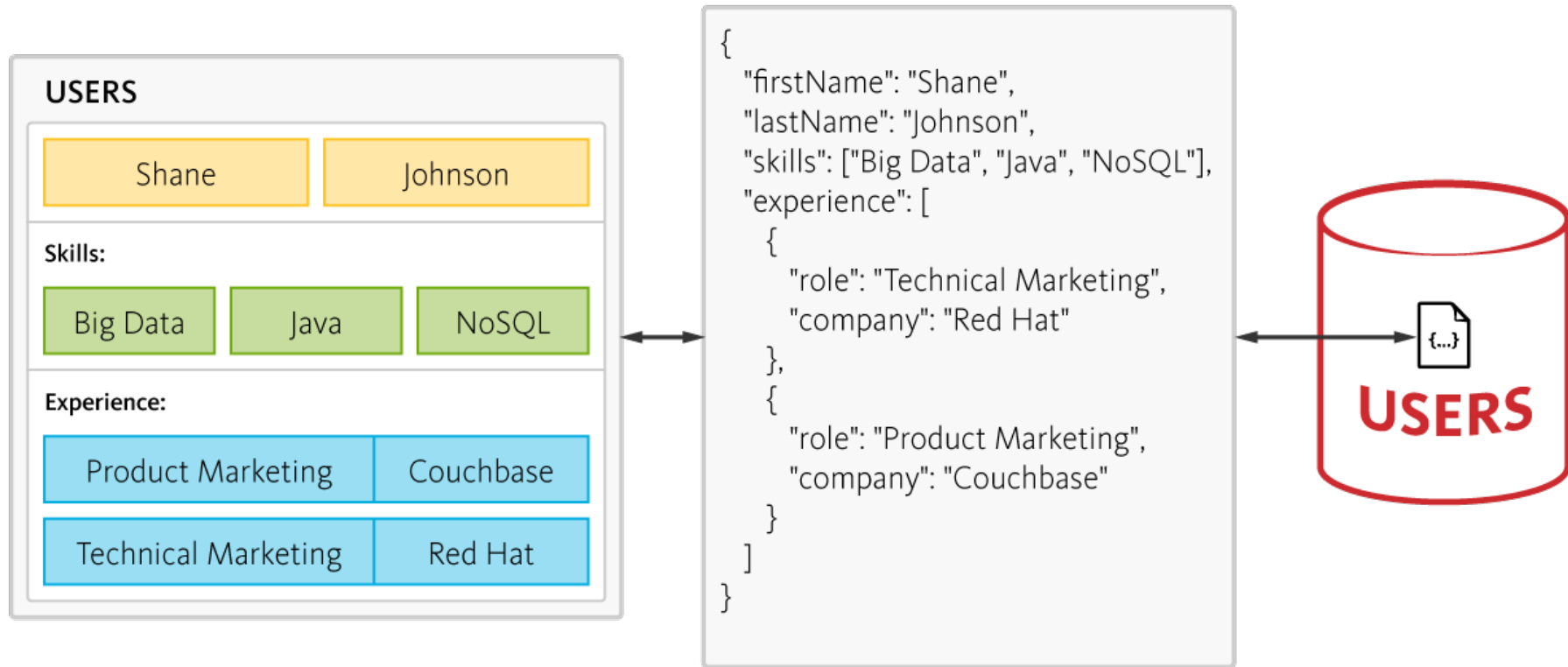
Johnson

@shane_dev

```
{  
  "firstName": "Shane",  
  "lastName": "Johnson",  
  "twitter": "@shane_dev"  
}
```



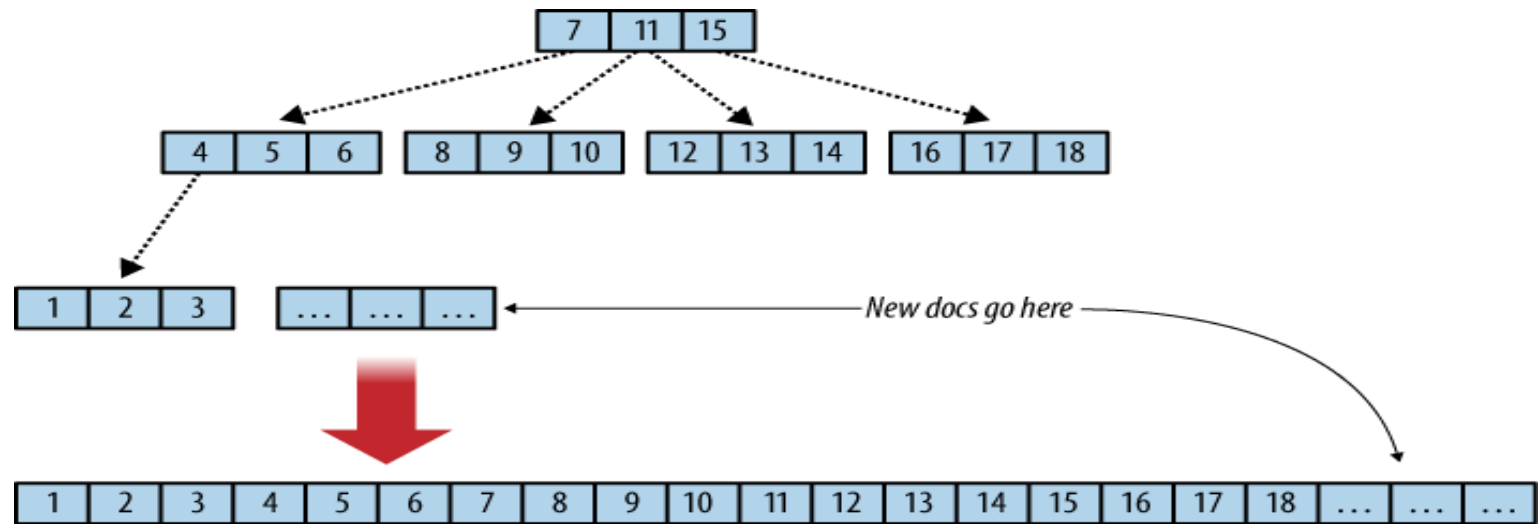
Flexible Schema in CouchDB



Object to document in CouchDB

Behind the scenes

- Append-Only B-tree Storage Engine(Robust!)
- Operations in logarithmic time
- Built-in HTTP server



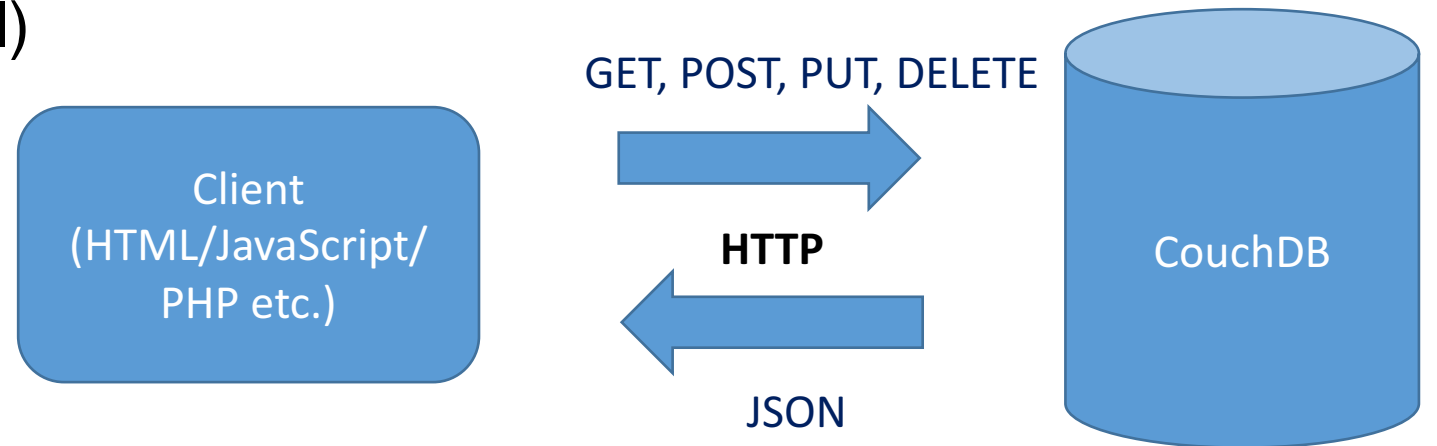
JSON

- JavaScript Object Notation
- Lightweight data-interchange format
- Language independent but uses conventions of the C-family of languages
- Supports data types such as : Number, String, Boolean, Array, Value, Object and Whitespace.

```
{
  "Time"      : "2014-02-12 14:20:05",
  "Latitude"  : 37.33233141,
  "Longitude" : -122.0312186,
  "Count"     : 101,
  "Comments"  : "Bad Data. SNOW DAY!!",
  "Luma"      : 0,
  "Habitat"   : "Back yard, grass",
  "Types"     : [
    "5",
    "6"
  ],
  "Address"   : {
    "Street"   : "2522 West Georgia",
    "City"     : "Piedmont",
    "State"    : "South Carolina",
    "Country"  : "United States",
  }
}
```

Core API

- RESTful API - All you need is HTTP!
- Subdivided into:
 - Server
 - Database
 - Documents
 - Replication(non-RestFul)



Core API – Examples

1. `curl http://127.0.0.1:5984/`

```
{"couchdb": "Welcome", "version": "0.10.1"}
```

2. `curl -X PUT http://127.0.0.1:5984/albums`

```
{"ok": true}
```

```
{"error": "file_exists", "reason": "The database could not be created, the file  
already exists."}
```


Core API – Examples

3.

```
curl -X PUT http://127.0.0.1:5984/albums/6e1295ed6c29495e54cc05947f18c8af -d '{"title":"There is Nothing Left to Lose","artist":"Foo Fighters"}'
```

```
{"ok":true,"id":"6e1295ed6c29495e54cc05947f18c8af","rev":"1-2902191555"}
```

4.

```
curl -X GET http://127.0.0.1:5984/albums/6e1295ed6c29495e54cc05947f18c8af
```

```
{"_id":"6e1295ed6c29495e54cc05947f18c8af","_rev":"1-2902191555","title":"There is Nothing Left to Lose","artist":"Foo Fighters"}
```

5.

```
> curl -vX PUT http://127.0.0.1:5984/albums/6e1295ed6c29495e54cc05947f18c8af/artwork.jpg?rev=2-2739352689 --data-binary @artwork.jpg -H "Content-Type: image/jpg"
```

Querying data with Views

- To aggregate, join, and report on documents
- Queried and indexed using **MapReduce**
- Built using JavaScript
- You don't run it yourself!
- Results sorted by key
- Types:
 - Temporary – for development, very slow
 - Permanent – for production

Views - Map

```
{
  "_id": "978-0-596-15589-6",
  "title": "CouchDB: The Definitive Guide",
  "subtitle": "Time to Relax",
  "authors": [
    "J. Chris Anderson",
    "Jan Lehnardt",
    "Noah Slater"
  ],
  "publisher": "O'Reilly Media",
  "released": "2010-01-19",
  "pages": 272
}
```

JSON Document

```
function(doc) { // JSON object representing a doc to be mapped
  if (doc.title) { // make sure this doc has a title
    emit(doc.title); // emit the doc's title as the key
  }
}
```

Map Function

key	id	value
"CouchDB: The Definitive Guide"	"978-0-596-15589-6"	null

Result

Views – Reduce

Function	Output
<code>_count</code>	Returns the number of mapped values in the set
<code>_sum</code>	Returns the sum of the set of mapped values
<code>_stats</code>	Returns numerical statistics of the mapped values in the set including the sum, count, min, and max

Built-in Reduce functions

```
function(keys, values, rereduce) {  
  if (rereduce) {  
    return sum(values);  
  } else {  
    return values.length;  
  }  
}
```

Custom Reduce function

key	value
"Ebook"	2
"Print"	3
"Safari Books Online"	2

Result

Views – The Challenges

- Map/Reduce
- JavaScript
- Multiple views, one design document
- Building/Indexing views
- View sizes on disk

Design Documents

- Contains application code
- CouchDB looks for views and other application functions here
- Static HTML pages of our application are served as attachments

```
1 {
2   "_id": "_design/sofa", ← Determines the app URL
3   "_rev": "3157636749",
4
5   "language": "javascript", (for the web)
6
7   "validate_doc_update": "function (newDoc, oldDoc, userCtx) { ... }",
8                               Application is stored as JSON data
9
10  "views": { ← Views field stores incremental
11             comments: { map reduce functions
12                       "map": "function(doc) { ... };",
13                       "reduce": "function(keys, values, rereduce) { ... };",
14                       }
15          },
16
17  },
18
19  "shows": { ← Shows functions transform
20             post: "function(doc, req) { ... }"
21          },
22
23  "attachments": { ← Attachments show
24                   up as stubs
25                   "jquery.couchapp.js": {
26                     "stub": true,
27                     "content_type": "text/javascript",
28                     "length": 7539
29                   }
30          },
31
32  },
33  "signatures": { ← CouchApp traces attachments here
34                  "jquery.couchapp.js": "80078849ad6ca281f6993bd012c708f5",
35          },
36
37  "lib": { ← CouchApp can include
38           templates: {
39             "post": "<!DOCTYPE html> ... </html>"
40           }
41         }
42     }
43 }
```

Core Features of CouchDB

- Consistency
- MVCC
- Scaling
- Replication
- Offline

Eventual Consistency

- Object Level Atomic Updates
- Durable Writes
- Eventual consistency is a consistency model that guarantees that, if no new updates are made to a given object, eventually all accesses to that object will return the last updated value

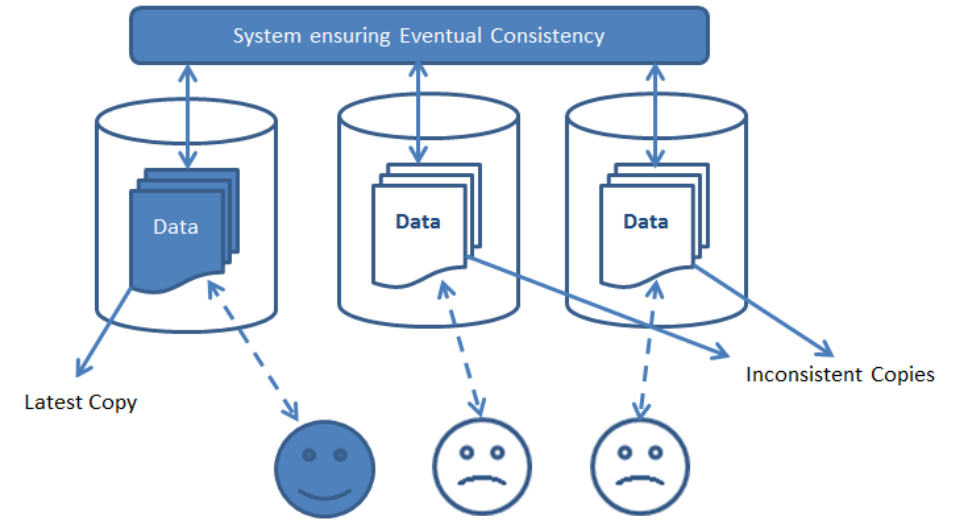
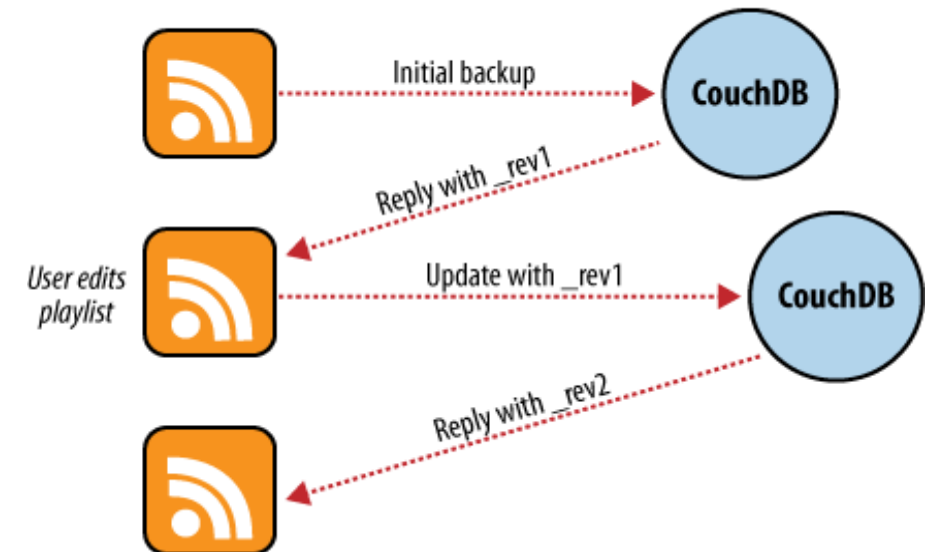


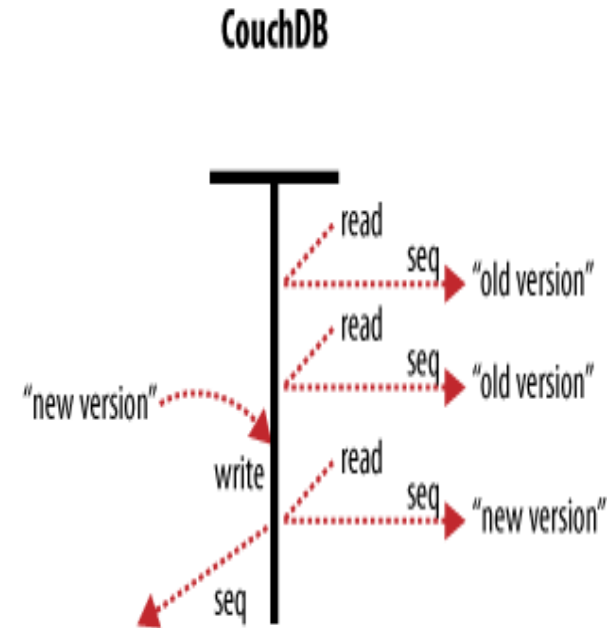
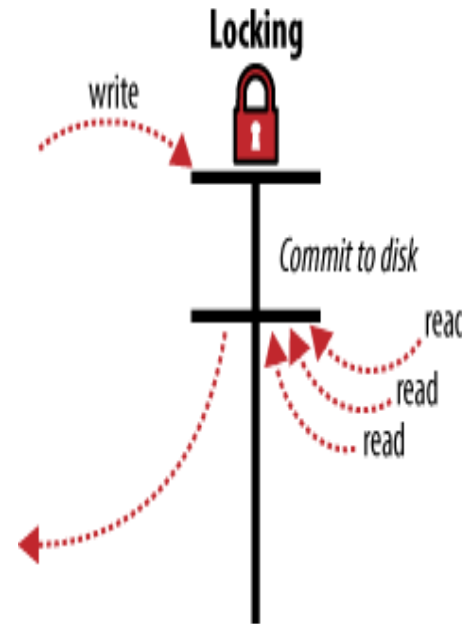
Figure showing a stale state where 2 copies of data are inconsistent with the latest one.

Usage of `_rev` keeping a single Songbird synced with CouchDB



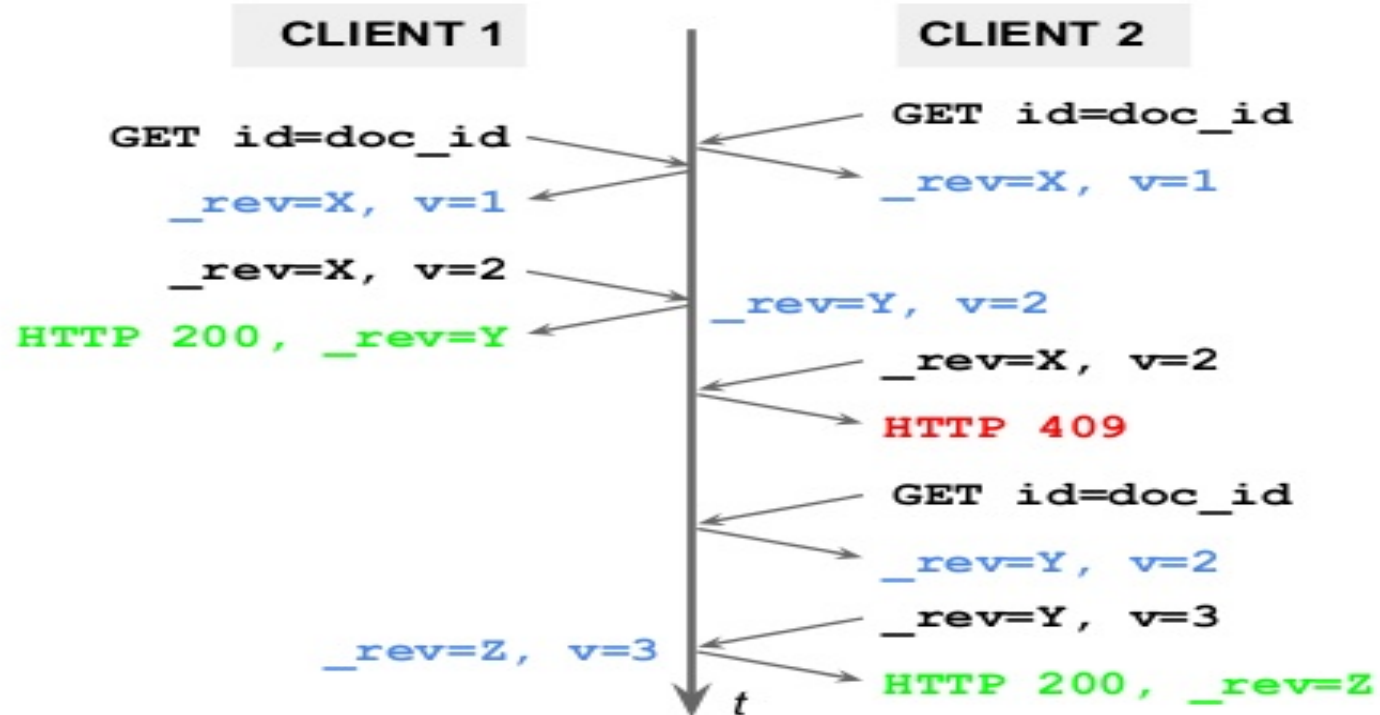
Multiversion Concurrency Control(MVCC)

- Locks in RDBMS
- Multiple revisions in CouchDB
- Each client sees a snapshot of the database at a particular instant in time.



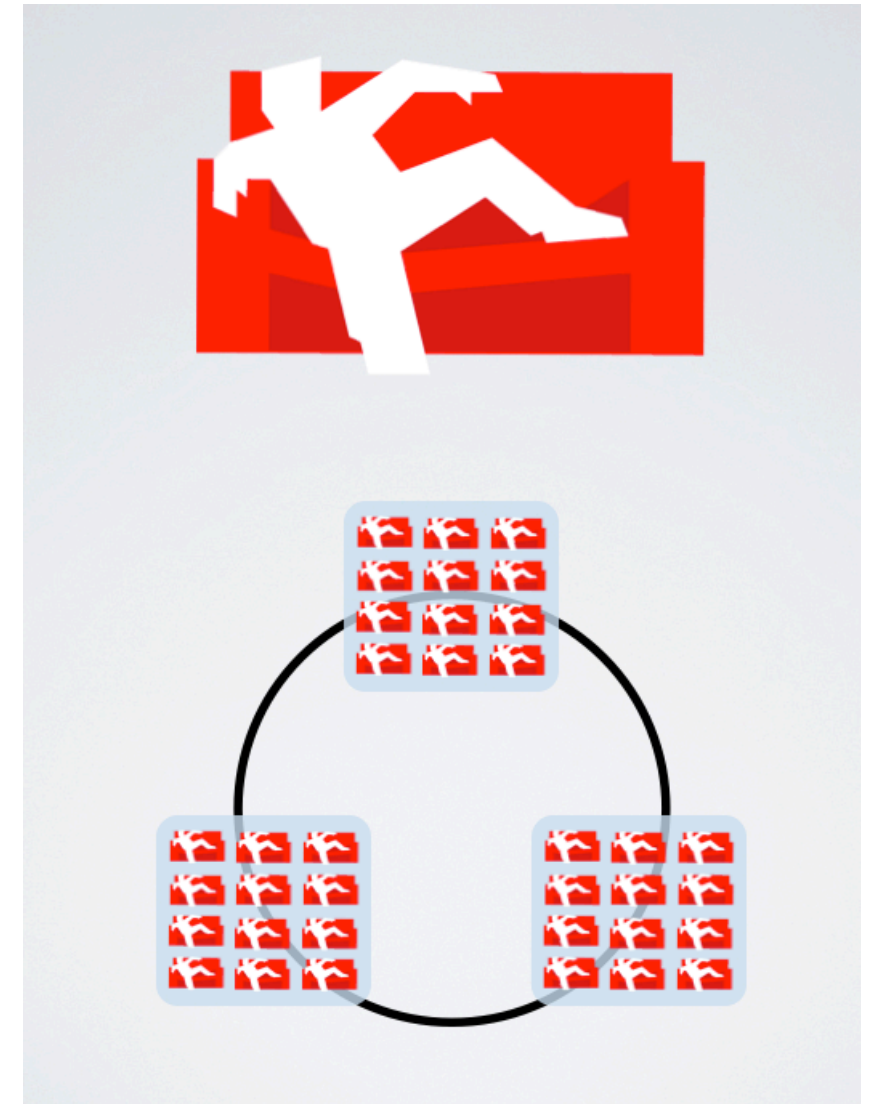
MVCC Example

Multiversion Concurrency Control (MVCC)



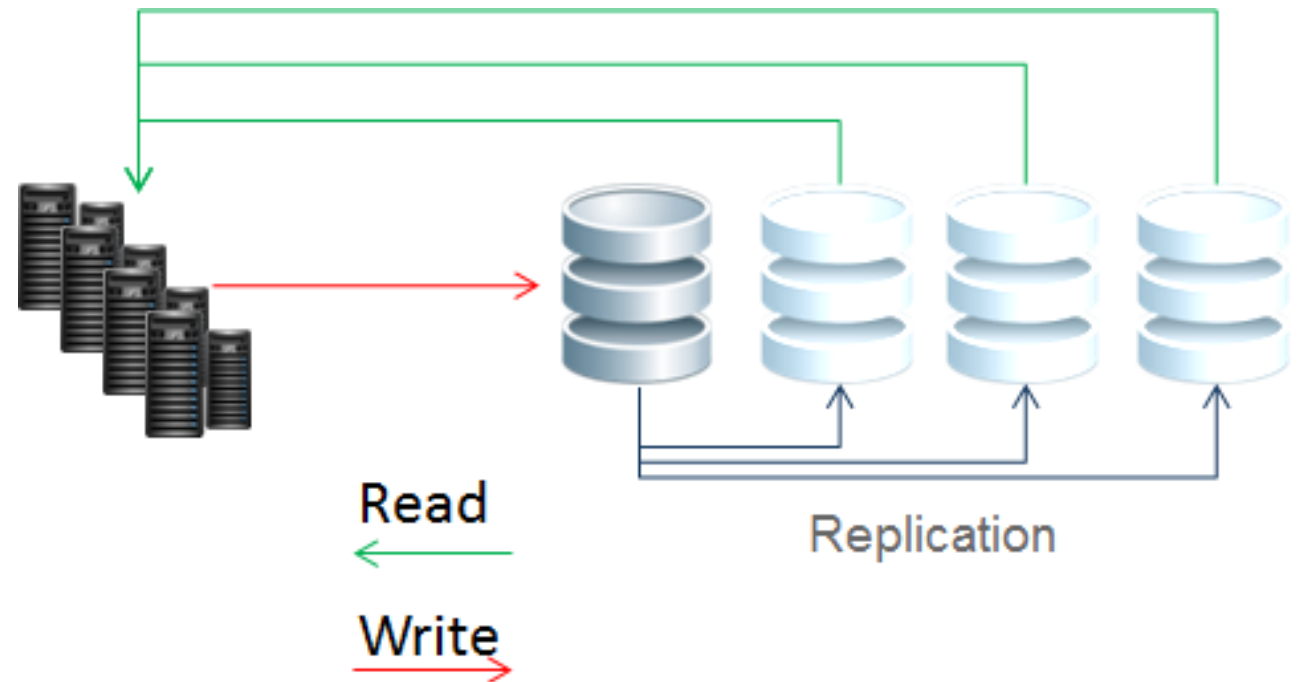
Scaling

- Need for Scaling
- Main challenges:
 - Search
 - Concurrency
 - Consistency
 - Speed
- Scaling up and Scaling out
- Clustering with Big Couch
- <http://instagram-engineering.tumblr.com/>



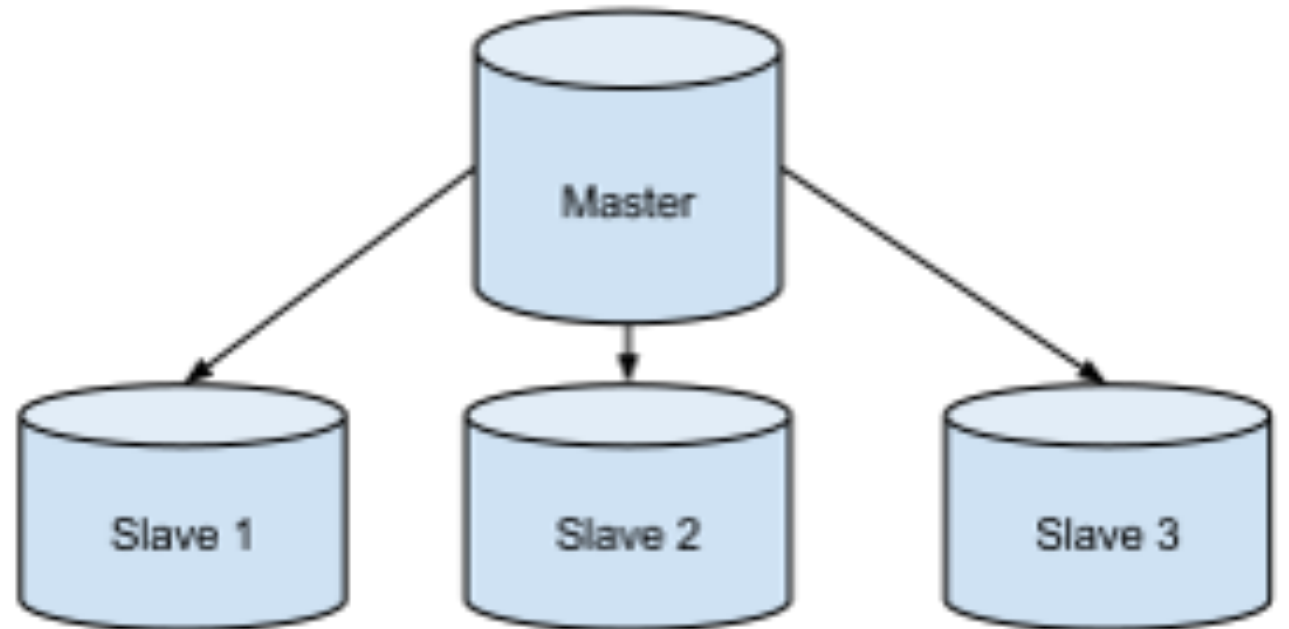
Replication

- What is Replication?
- Why do we need Replication?
 - Replication for Backup
 - Replication across Data Centers



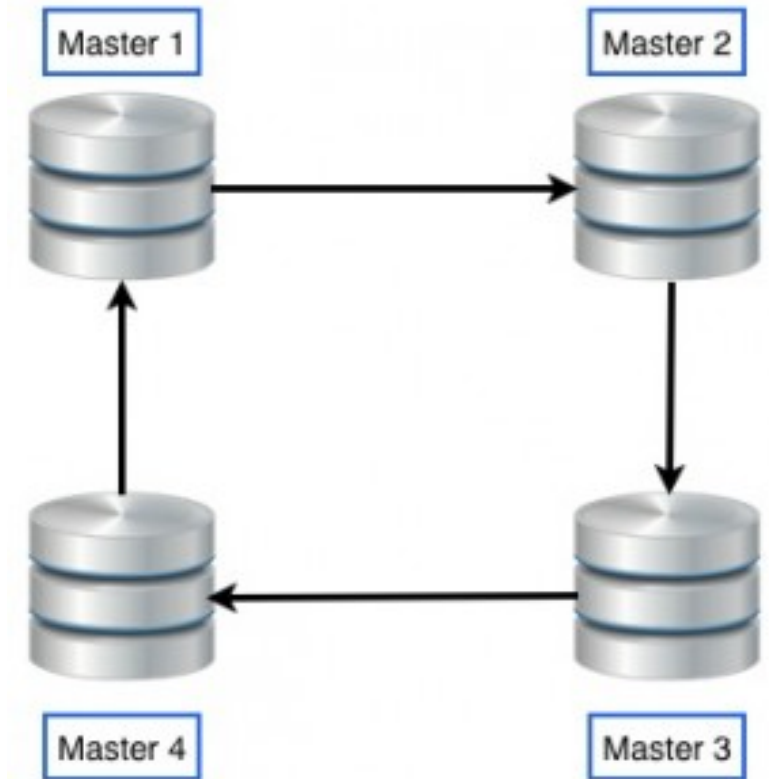
Master-Slave Replication

- Architecture used in CouchDB



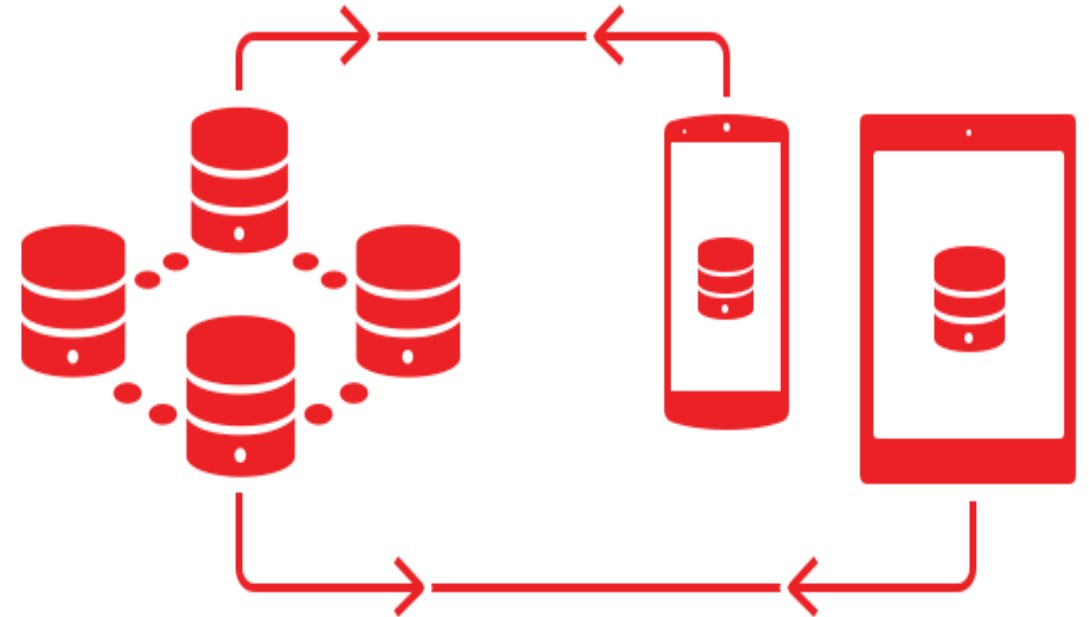
Multi-Master Replication

- Group of computers store the data instead of just one and updates can be done by any member of the group.
- Members are responsive to client data queries.
- Conflict resolution and Fault tolerance.



CouchDB-Offline

- Designed for Offline
- Scaling down rather than scaling out
- Local Data is King
- Use of Multi-Master replication

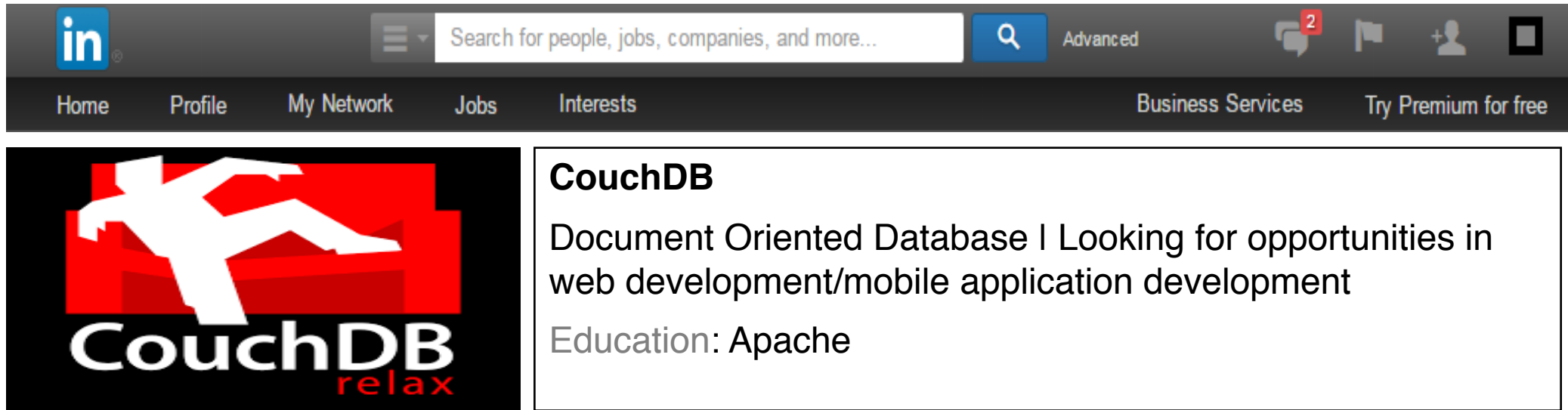


Career fair of NoSQL Databases

Architect

CouchDB





CouchDB
Document Oriented Database | Looking for opportunities in web development/mobile application development
Education: Apache

Summary:

Created by Damien Katz, a former IBM'er

Inspired from IBM's Lotus Notes database, which is also a document db.

Combination of unreliable commodity hardware

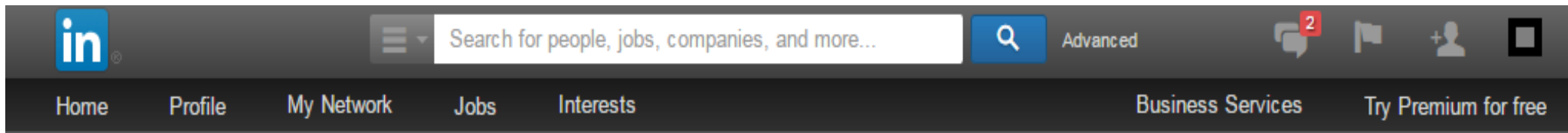
Interests:

I don't want to be your database; I want to be your website.

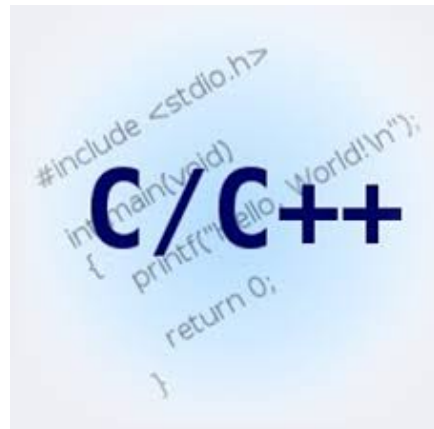
Building CouchApps

When web developers come to know they don't need separate middle tier





Languages



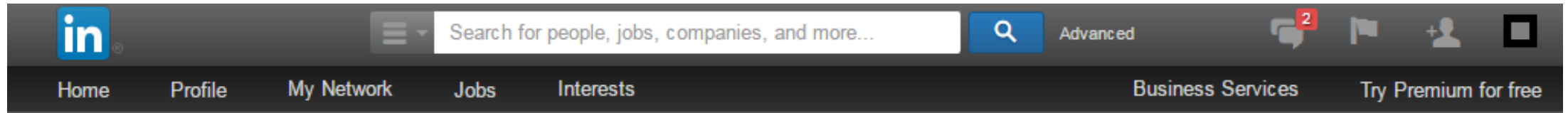
Logo courtesy - LinkedIn

Projects

1. The **Compact Muon Solenoid Experiment (CMS)** at CERN (2010):

- Capacity to handle large amount of data
- No complex replication infrastructure
- Works well with other systems

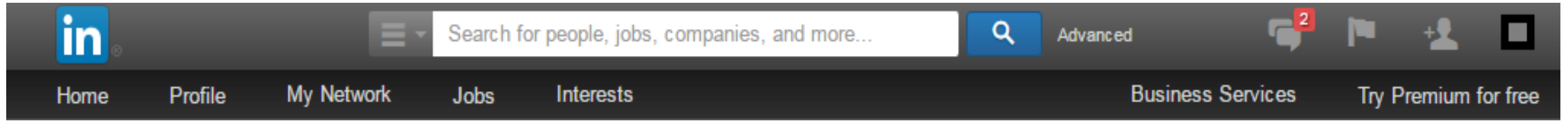




Projects

2. Improving healthcare with CouchDB (Zambia)

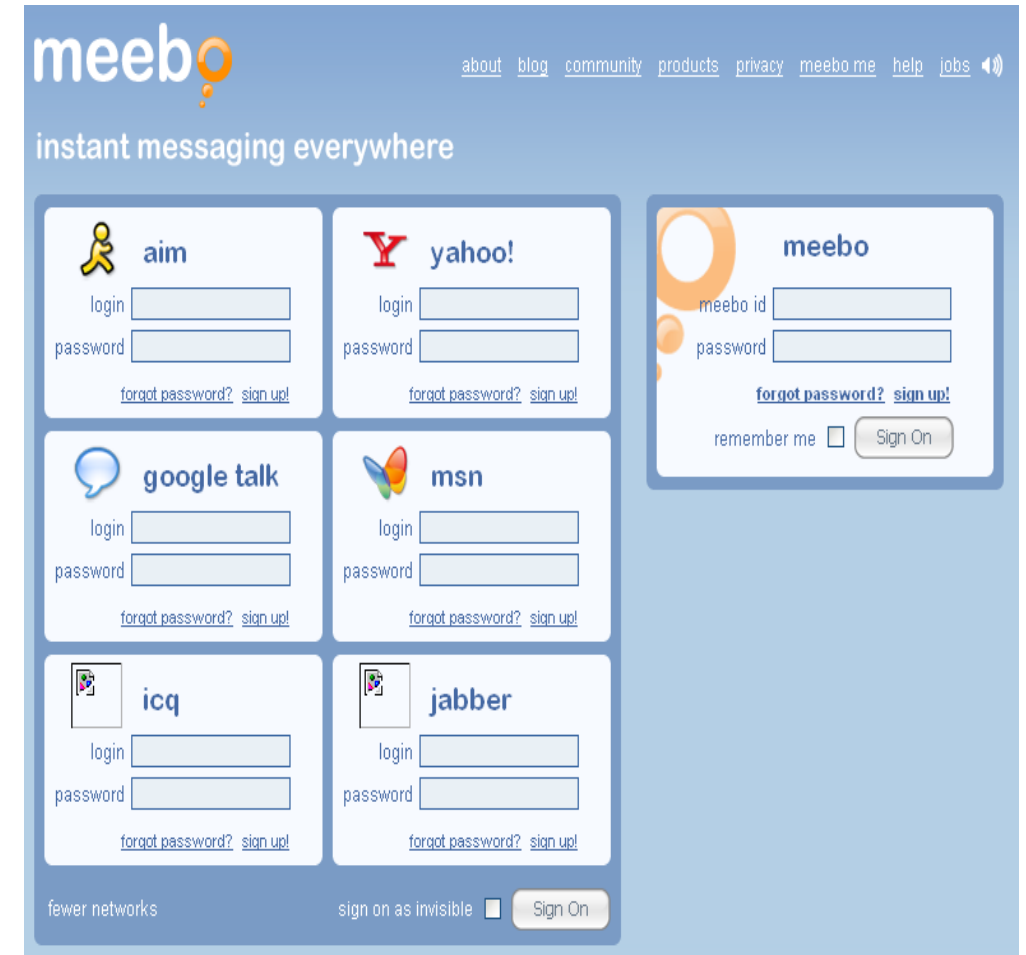
- Extremely remote, rural clinics
- Very poor and intermittent Internet connection
- Star topology of CouchDB servers

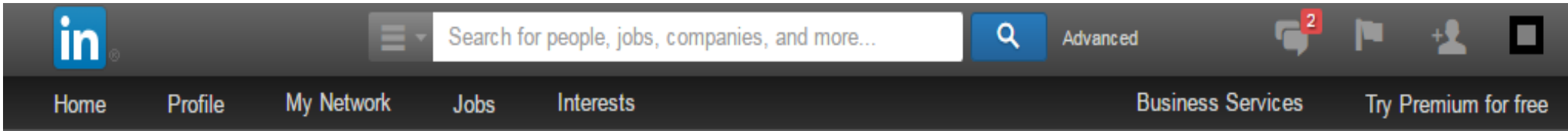


Projects

3. Meebo (2005):

- Simultaneous uniform access to multiple IM
- File transfer and videoconferencing features
- online games, multi-user chat (Meebo rooms)





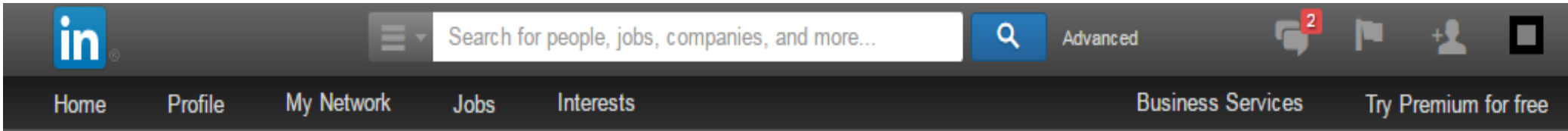
Projects (Failure)

4. SAUCELABS says Goodbye to CouchDB

- They liked many things about CouchDB, but...
- View indexes are only updated when queried
- Interesting ideas but not relevant to our needs

When not to use me?

- Application for banking, airline reservations , online shopping



Recommendations

Django may be built *for* the Web, but CouchDB is built *of* the Web. I've never seen software that so completely embraces the philosophies behind HTTP. CouchDB makes Django look old-school in the same way that Django makes ASP look outdated.

—*Jacob Kaplan-Moss, Django developer*

References

1. <http://guide.couchdb.org/>
2. <https://nolanlawson.com/2013/11/15/couchdb-doesnt-want-to-be-your-database-it-wants-to-be-your-web-site/>
3. <http://thewebhacker.com/rapid-app-prototyping-with-angularjs-and-couchdb/>
4. <https://saucelabs.com/blog/goodbye-couchdb>
5. <https://www.ateamsystems.com/tech-blog/why-i-switched-to-couchdb-for-web-applications/>
6. <http://johnpwood.net/2009/07/21/couchdb-views-the-challenges/>

Advanced Databases
CIS 6930 | Fall 2016
Dr. Markus Schneider

Group 4

- Raji Sundararajan
- Divyalakshmi Mahendran
- Ram Gandikota
- Atharva Borkar



I'LL BE THERE FOR YOU