

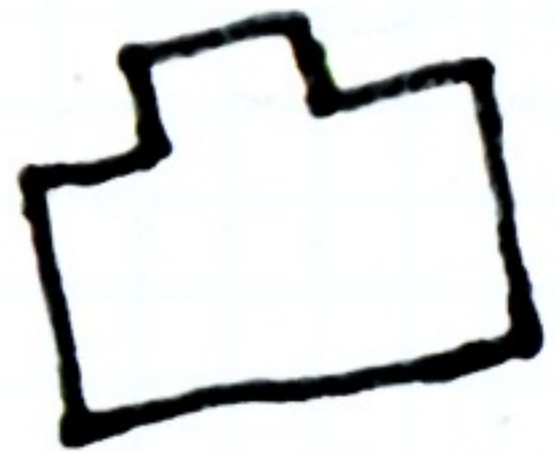
Apache  
**CouchDB**

Relax

the p2p



web.



# Hello

Chris Anderson

jchris@apache.org / @jchris

Apache CouchDB Committer

REST and JavaScript enthusiast

Director, couch.io



# About You?

Easier way to make web applications

Scalable key/value store

Peer-based Replication

Append-only IO pattern

# Scaling Up



- Hot. Literally.
- You don't have to think.

# Scaling Up



Hot. Little. Easy.

- You don't have to think.

# Scale Enforces Constraints

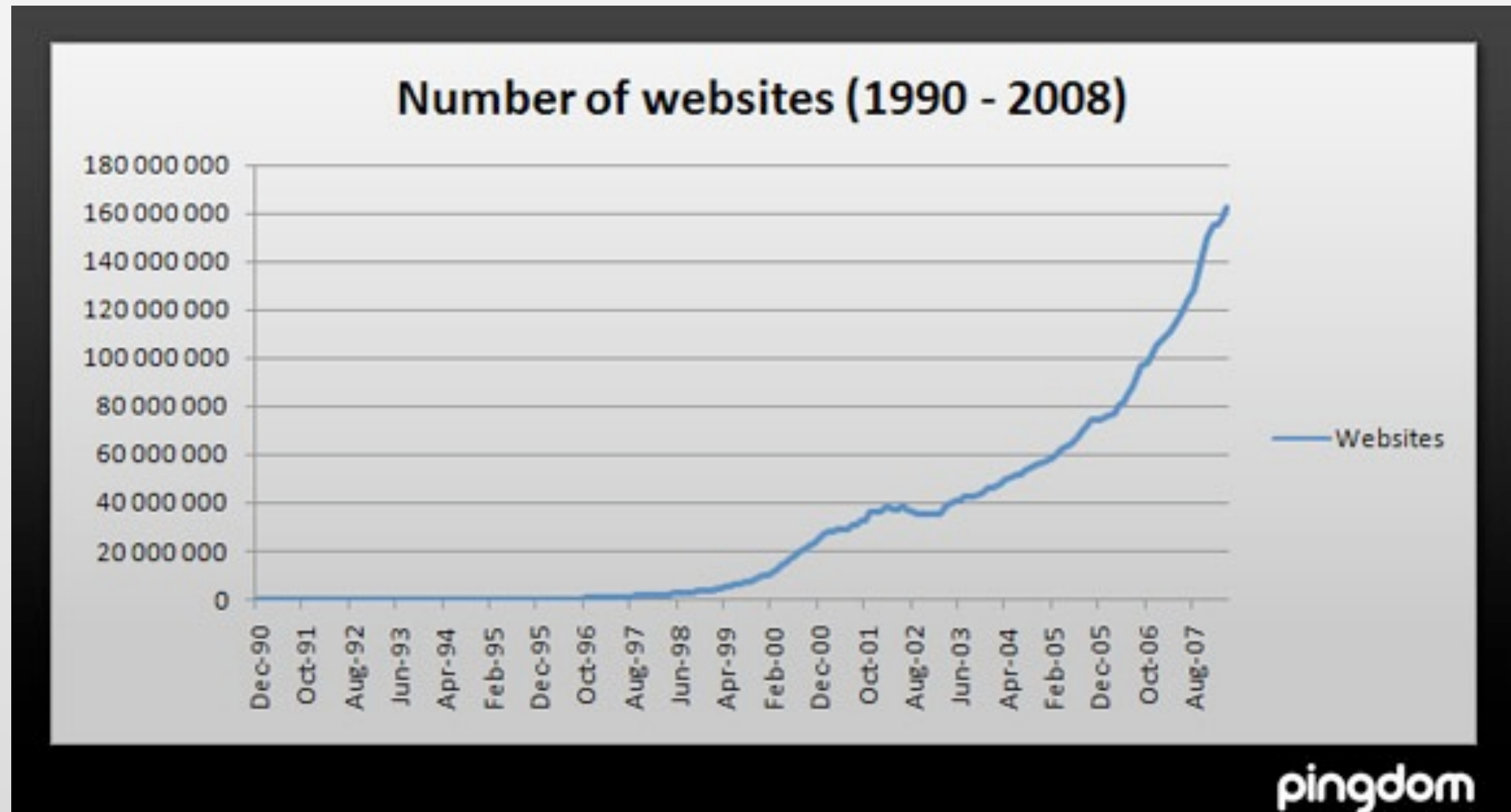
Distributed key / value stores

Context is expensive

No joins or autoincrement

Eventual consistency

# The Web Scales



# Why CouchDB?

Native to the Web

Scale from a cluster to a smartphone

Local data == low latency

Replicated applications - the p2p web

**Makes Google old-school**



# JavaScript Tests

The screenshot shows the Apache CouchDB Futon Utility Client interface. The browser address bar displays `http://127.0.0.1:5984/_utils/`. The page has two tabs: "Overview" and "Test Suite", with "Test Suite" being the active tab. Below the tabs, there are buttons for "Run All" and "Reload". A note states: "Note: Each of the tests will block the browser. If the connection to your CouchDB server is slow, running the tests will take some time, and you'll not be able to do much with your browser while a test is being executed."

The main content area displays a table of test results:

Name	Status	Elapsed Time	Details
basics	✓ success	1230ms	
conflicts	✓ success	162ms	
recreate_doc	✓ success	803ms	
copy_move_doc	✓ success	1574ms	
uuids	✓ success	114ms	
bulk_docs	✓ success	185ms	
lots_of_docs	✓ success	1728ms	
reduce	✓ success	6422ms	
reduce_false	✓ success	114ms	
multiple_rows	✓ success	170ms	
large_docs	✓ success	282ms	
utf8	✓ success	245ms	
attachments	✓ success	415ms	
content_negotiation	✓ success	71ms	
design_docs	✓ success	3008ms	
view_collation	⊙ running...		
view_conflicts	not run		
view_errors	not run		
view_include_docs	not run		

On the right side of the interface, there is a sidebar with the CouchDB logo and the text "relax". Below the logo, there is a "Tools" section with links to "Overview", "Replicator", "Configuration", and "Test Suite". The "Test Suite" link is highlighted. Below the "Tools" section, there is a "Recent Databases" section which is currently empty. At the bottom of the sidebar, there is a status bar that reads "Futon on Apache CouchDB 0.9.0 - incubating".

# Of the Web

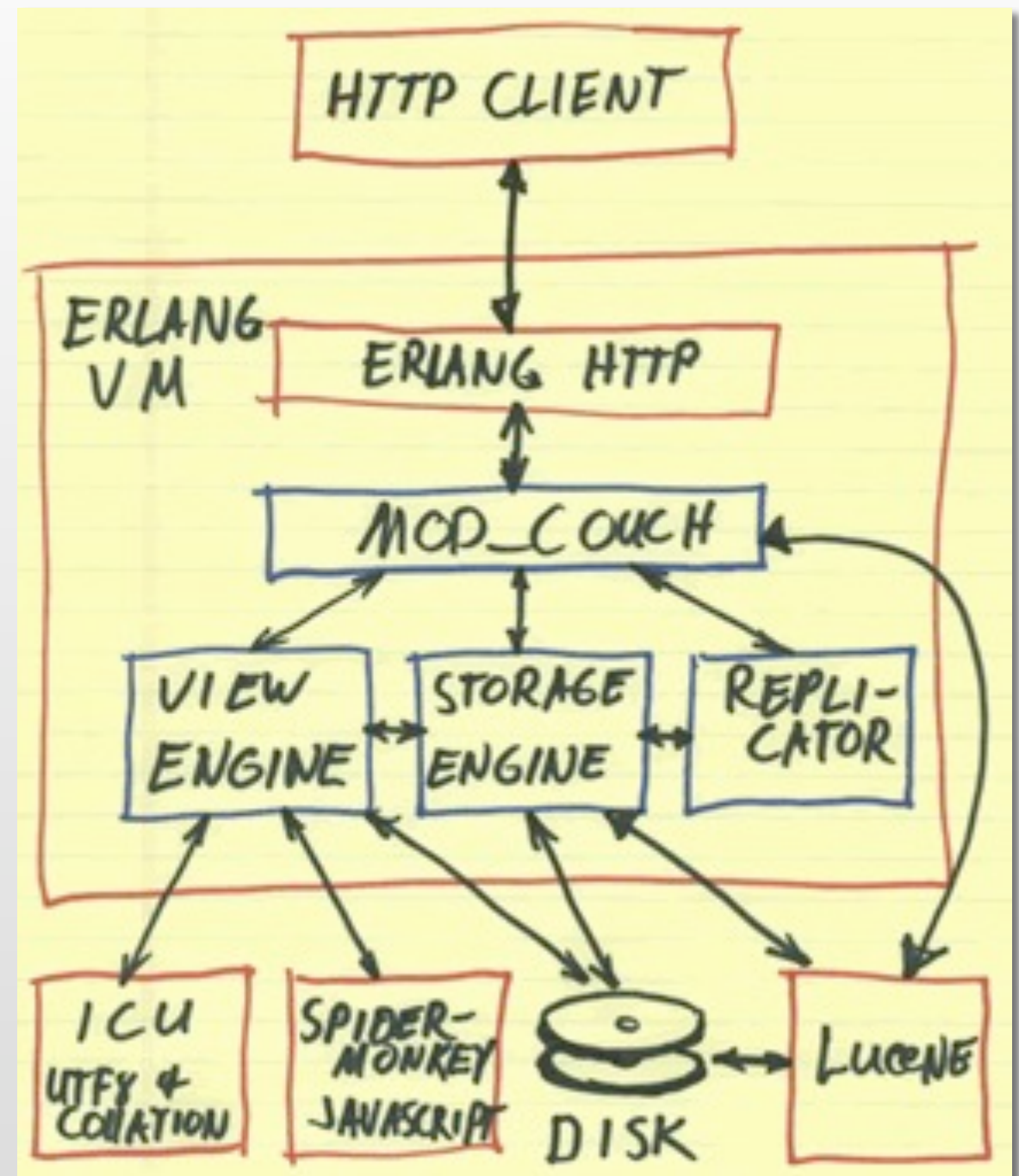
Let me tell you something: Django may be built *for* the Web, but CouchDB is built *of* the Web. I've never seen software that so completely embraces the philosophies behind HTTP. ... this is what the software of the future looks like.

Jacob Kaplan-Moss  
October 19, 2007

<http://jacobian.org/writing/of-the-web/>



Robust  
JSON  
HTTP  
Local



# Robust Storage

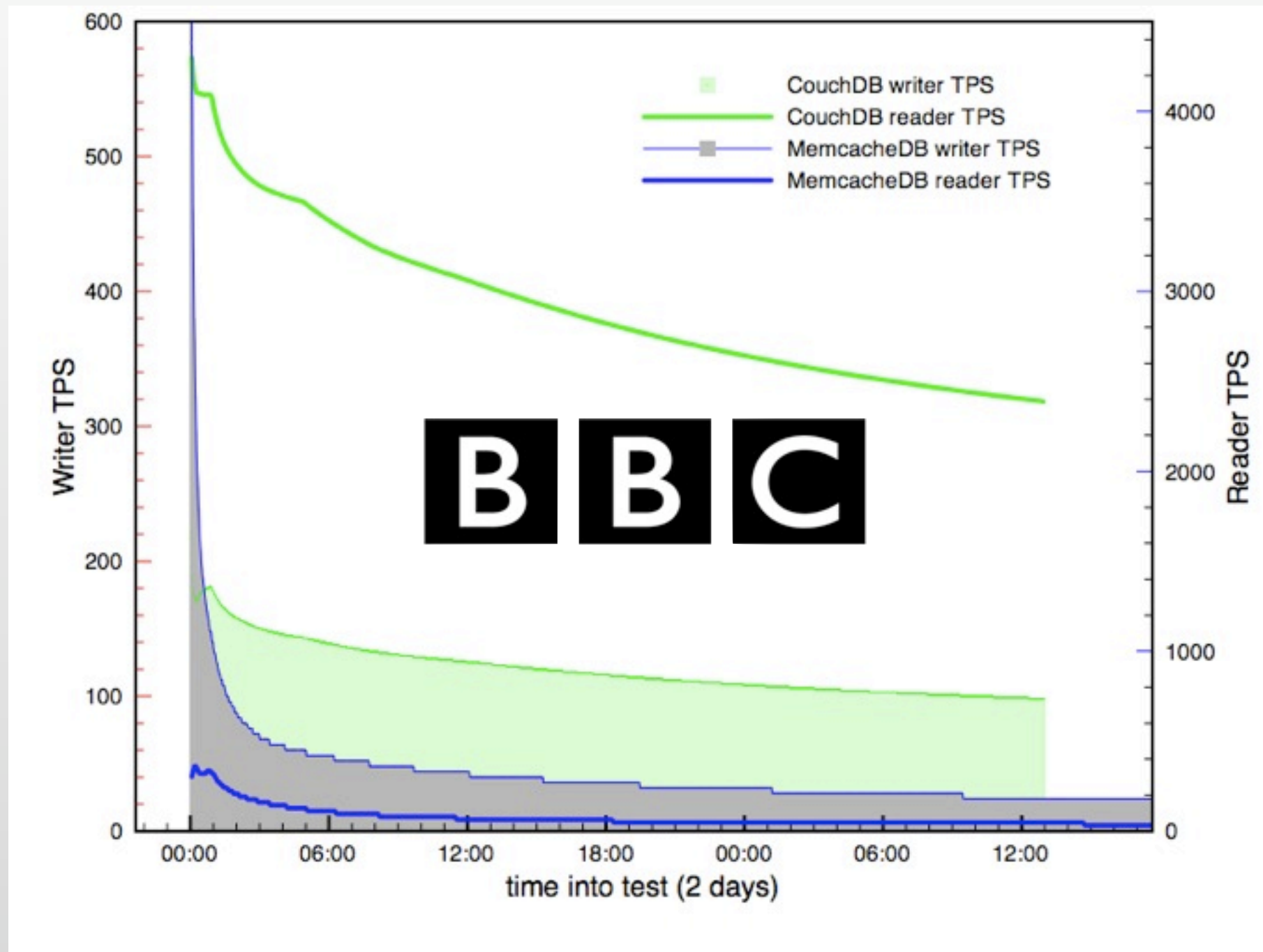
Append Only  
File Structure

Designed to Crash

Instant-On  
(no fixup phase)



# Happy IO Patterns



# JSON Documents

```
{  
  "_id": "BCCD12CBB",  
  "_rev": "1-AB764C",  
  
  "type": "person",  
  "name": "Darth Vader",  
  "age": 63,  
  "headware":  
    ["Helmet", "Sombrero"],  
  "dark_side": true  
}
```


Schema Free


Self-Contained


JavaScript Map  
Reduce Views





# JavaScript Map Reduce

Overview > **dri** 

 Create Document ...

 Compact Database

 Delete Database

Select view: **tags** 

▼ View Code \_design/sofa


Map Function:

```
function(doc) {  
  if(doc.tags && doc.tags.length) {  
    for(var idx in doc.tags) {  
      emit(doc.tags[idx], 1);  
    }  
  }  
}
```

Reduce Function (optional):

```
function(key, values) {  
  return sum(values);  
}
```

Run

Language: **javascript** 


=

Revert


Save As...

Save

Key	Value
"#leanintoit"	1
"admin"	1
"Apache"	1
"apachecon"	1
"bookmarklet"	1
"chat"	3
"code"	85
"comet"	2
"Couch"	1
"couchapp"	1

Showing 1-10 of unknown rows ← Previous Page | Rows per page: **10**  | Next Page →

**Note:** Views with a reduce function will return a single row of reduced values. If you are looking for the same results that you can see here in Futon, add a ?group=true parameter to your view query. Futon does this automatically.

  
**CouchDB**  
relax

Tools

Overview

Configuration

Replicator

Status

Test Suite

Recent Databases

**dri**

twitter-client

┌

Duck

└

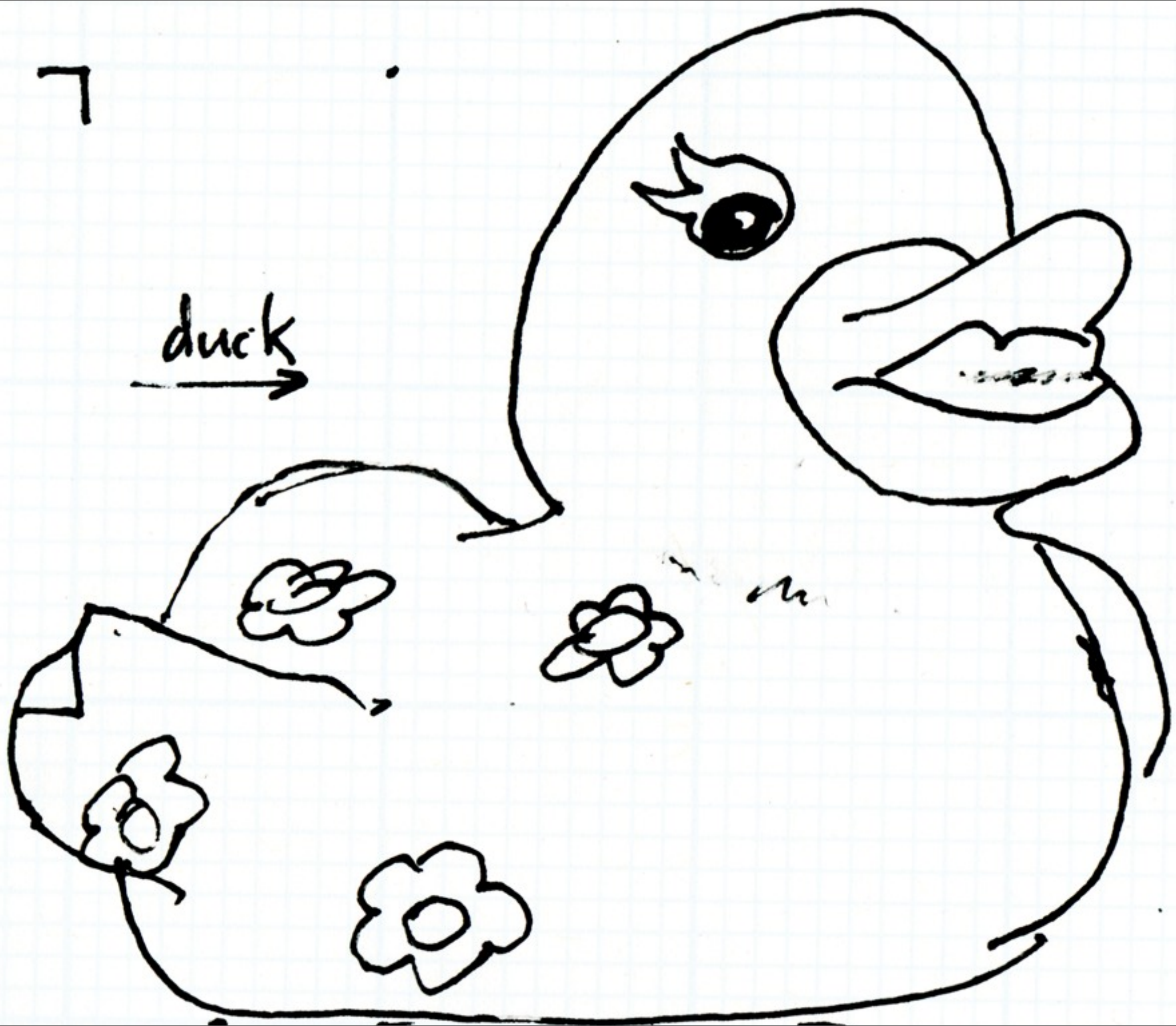
Typing

└

△

7

duck →





# REST*ful* HTTP

You already know the API

Use existing HTTP tools

Talk directly to the browser



# Partitioned Cluster w/ CouchDB-Lounge



Consistent Hashing

HTTP proxy

Nginx / Twisted Python

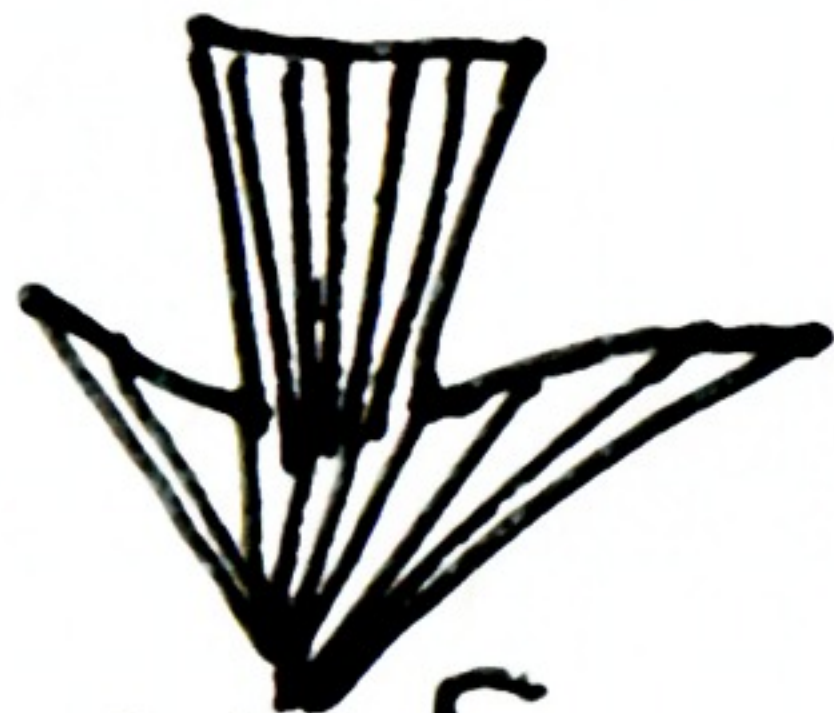
**Many machines == 1 Couch**

# Local Data

Low latency (fast)

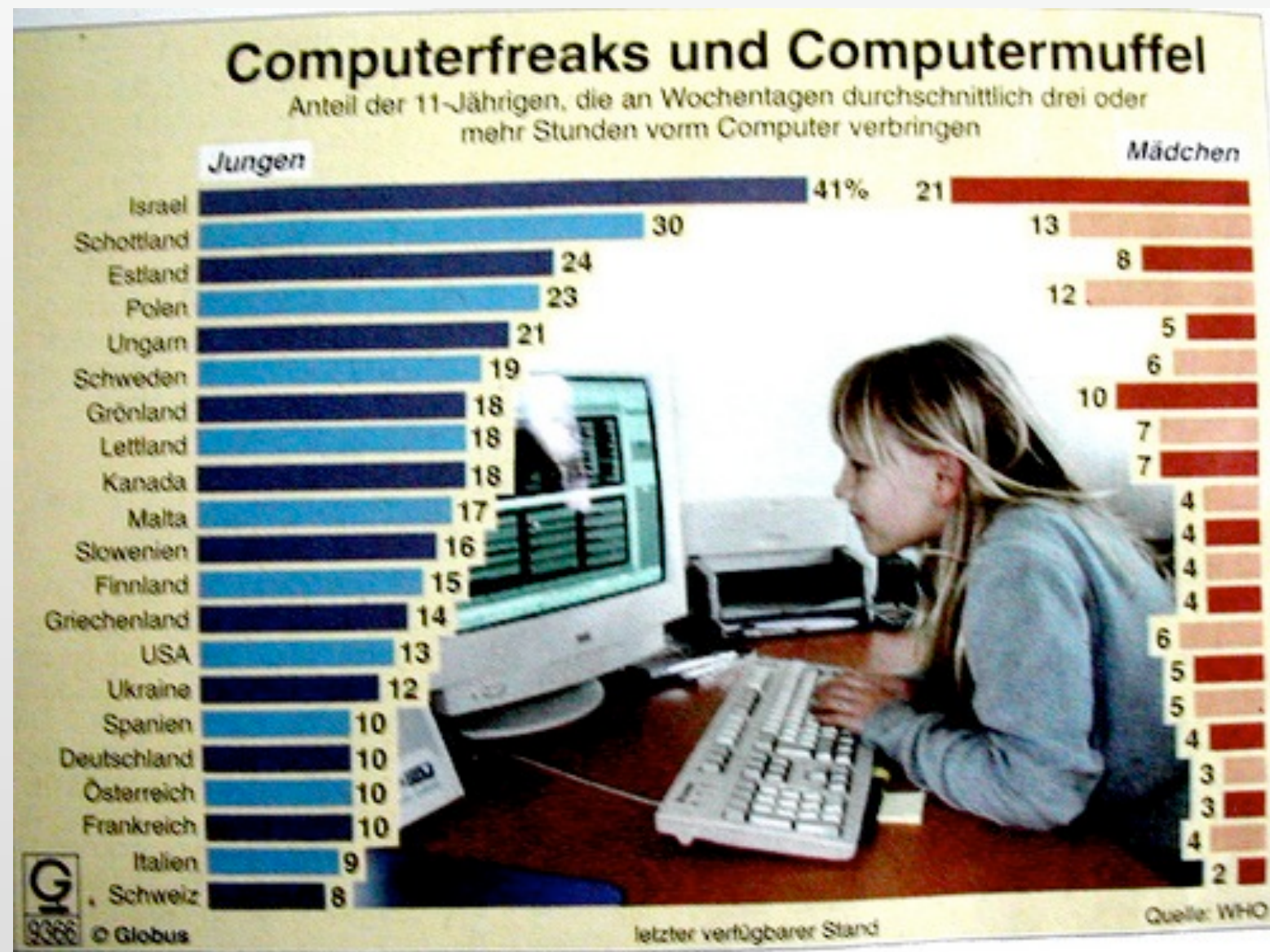
Peer-based incremental replication

View Source



Open Source

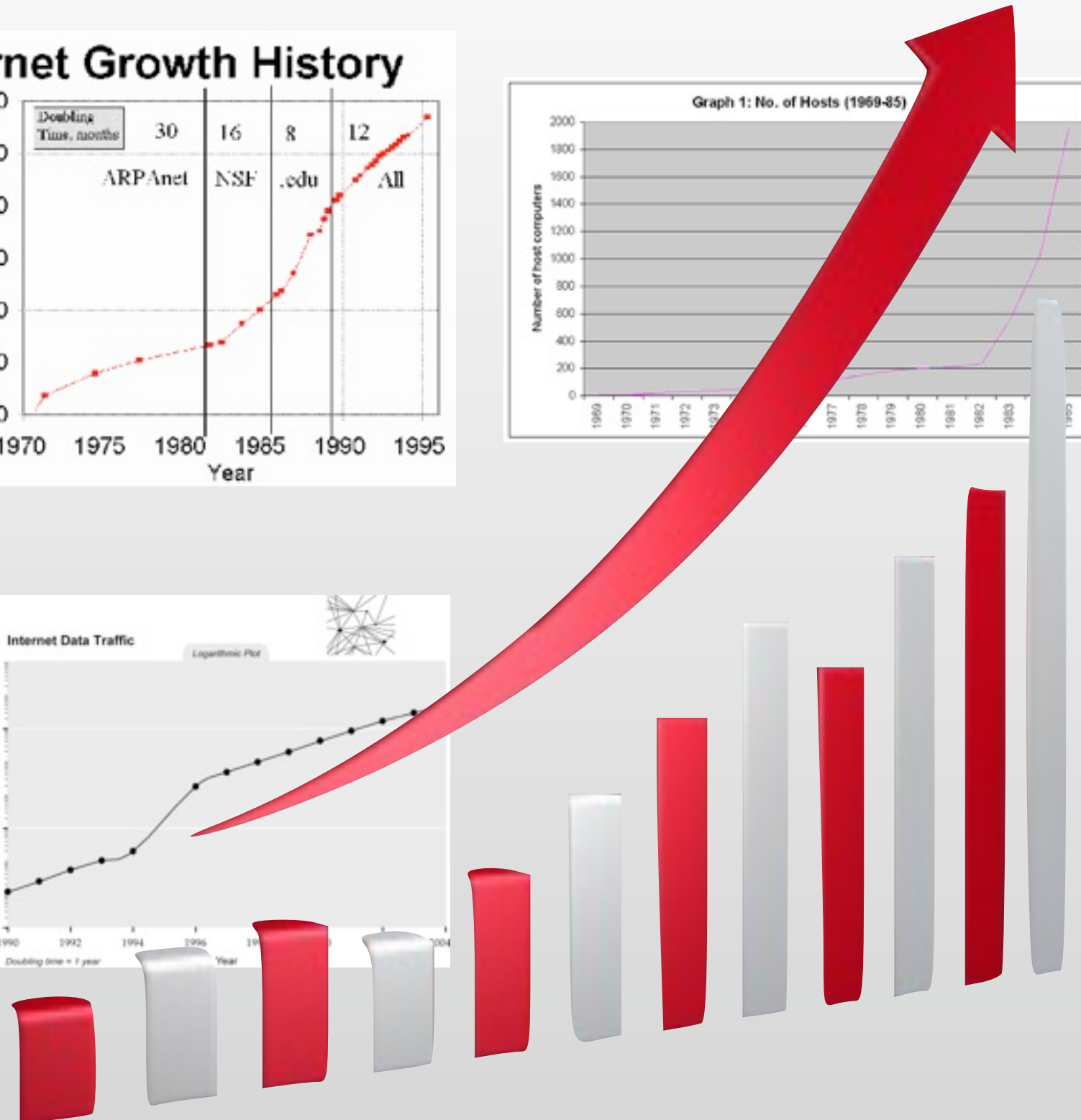
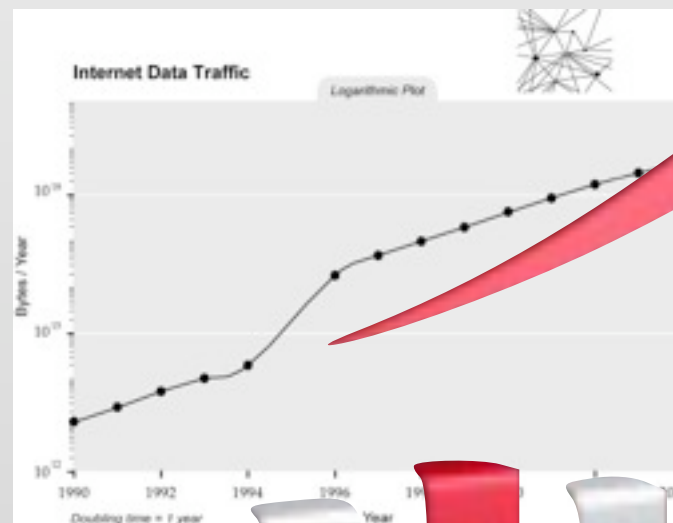
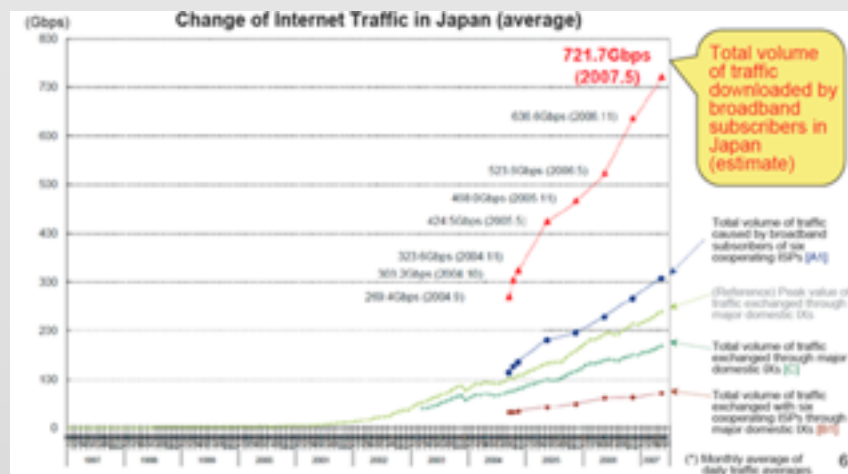
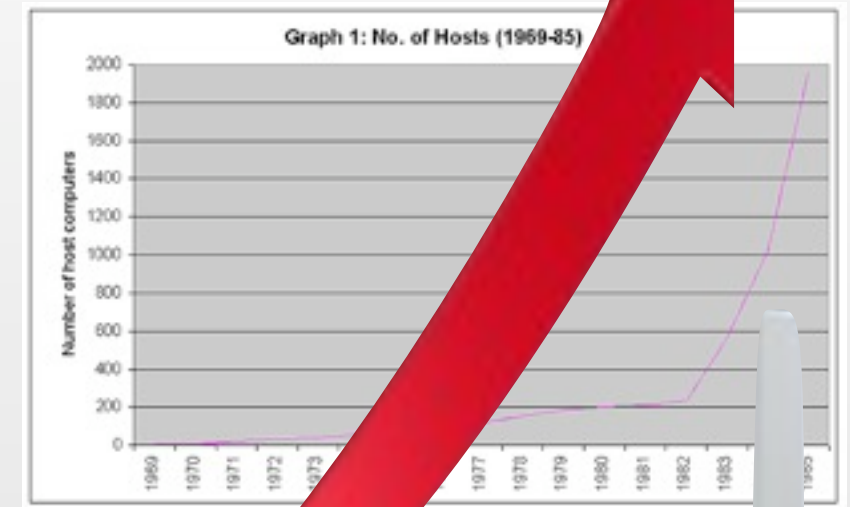
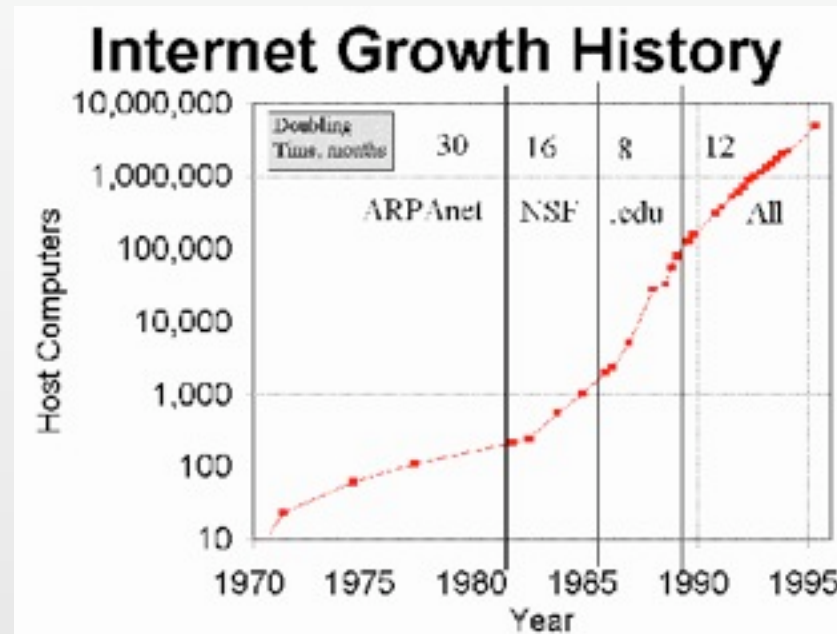
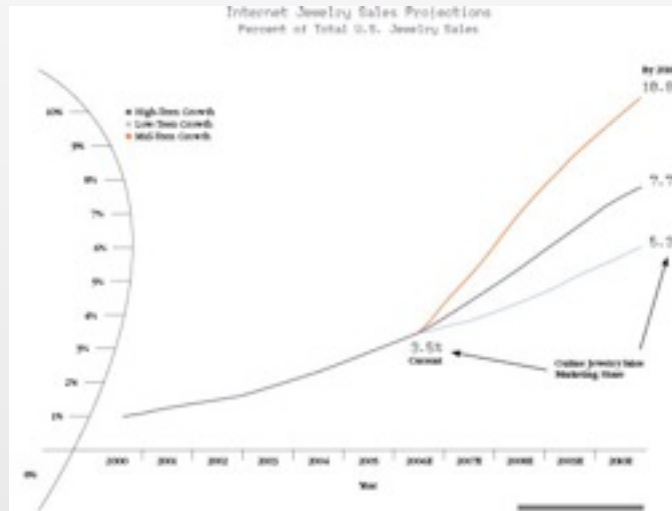
# Gives Control to Users



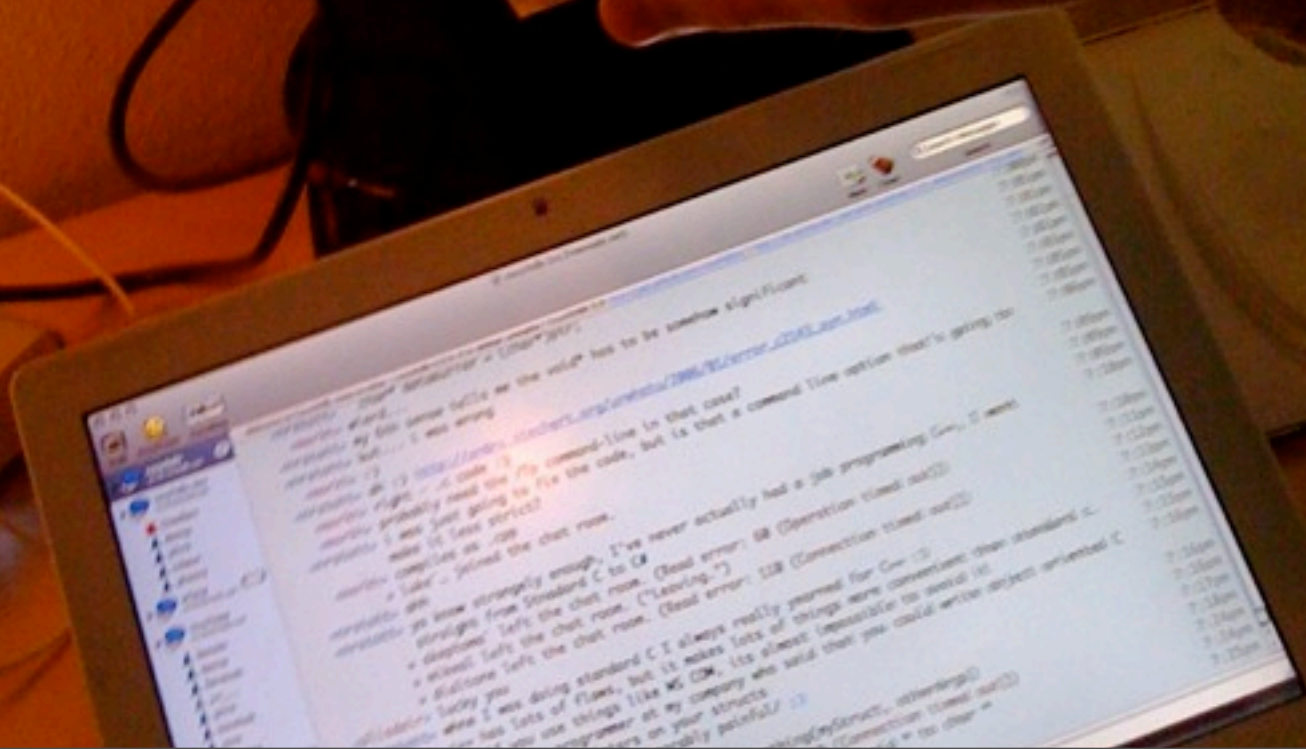
CC-BY-SA <http://www.flickr.com/photos/kelleys/492253912/>



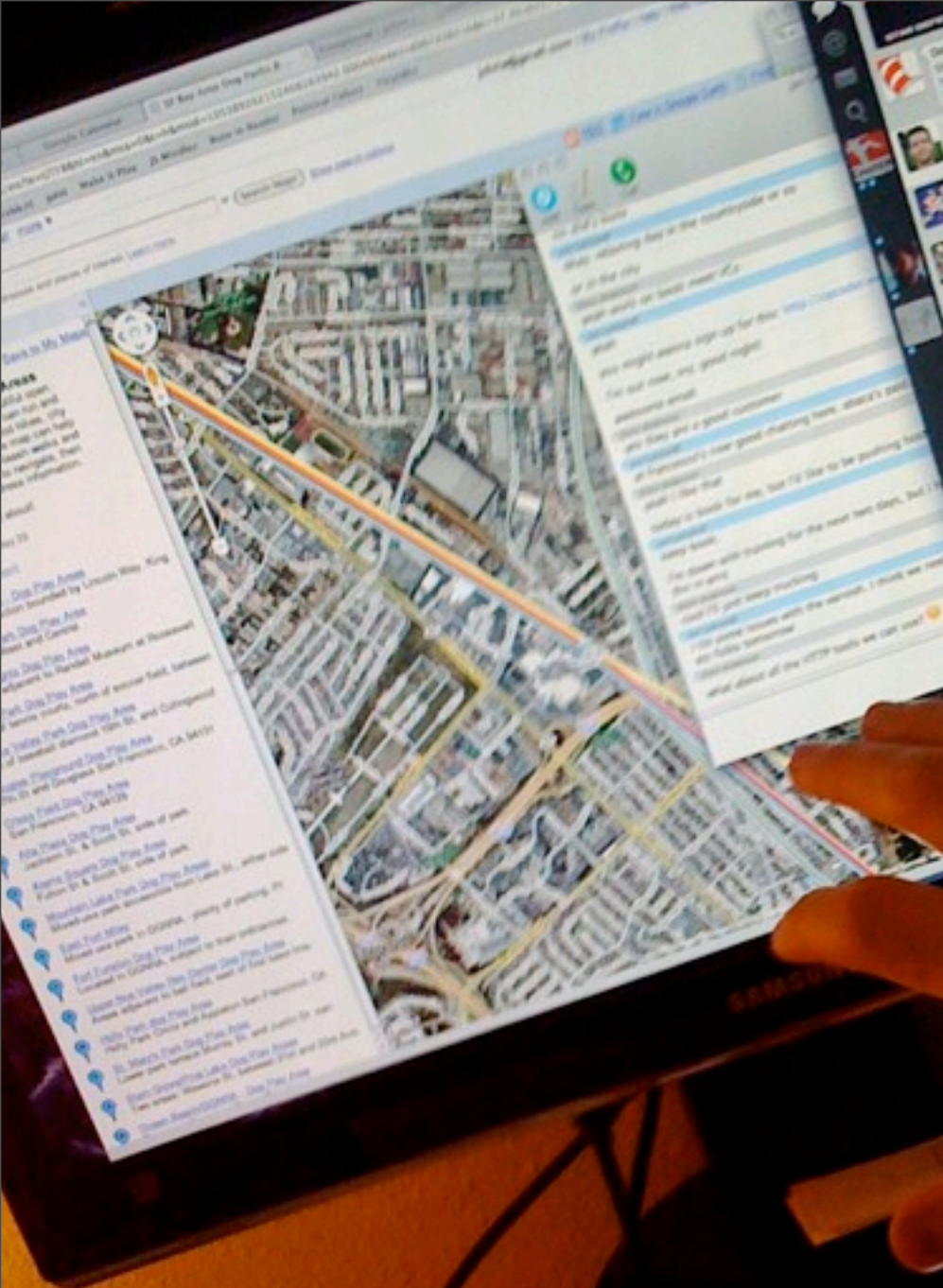
# Bandwidth Explosion











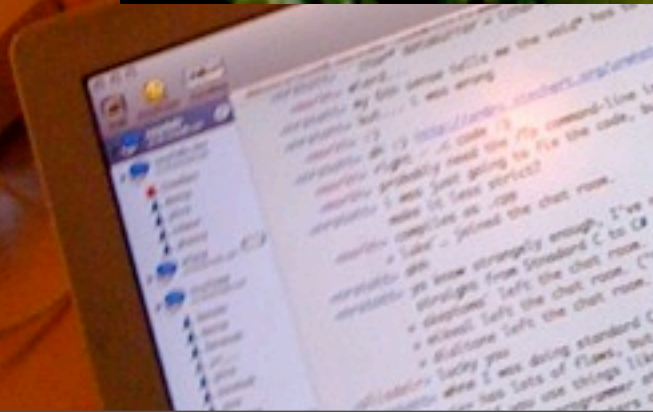
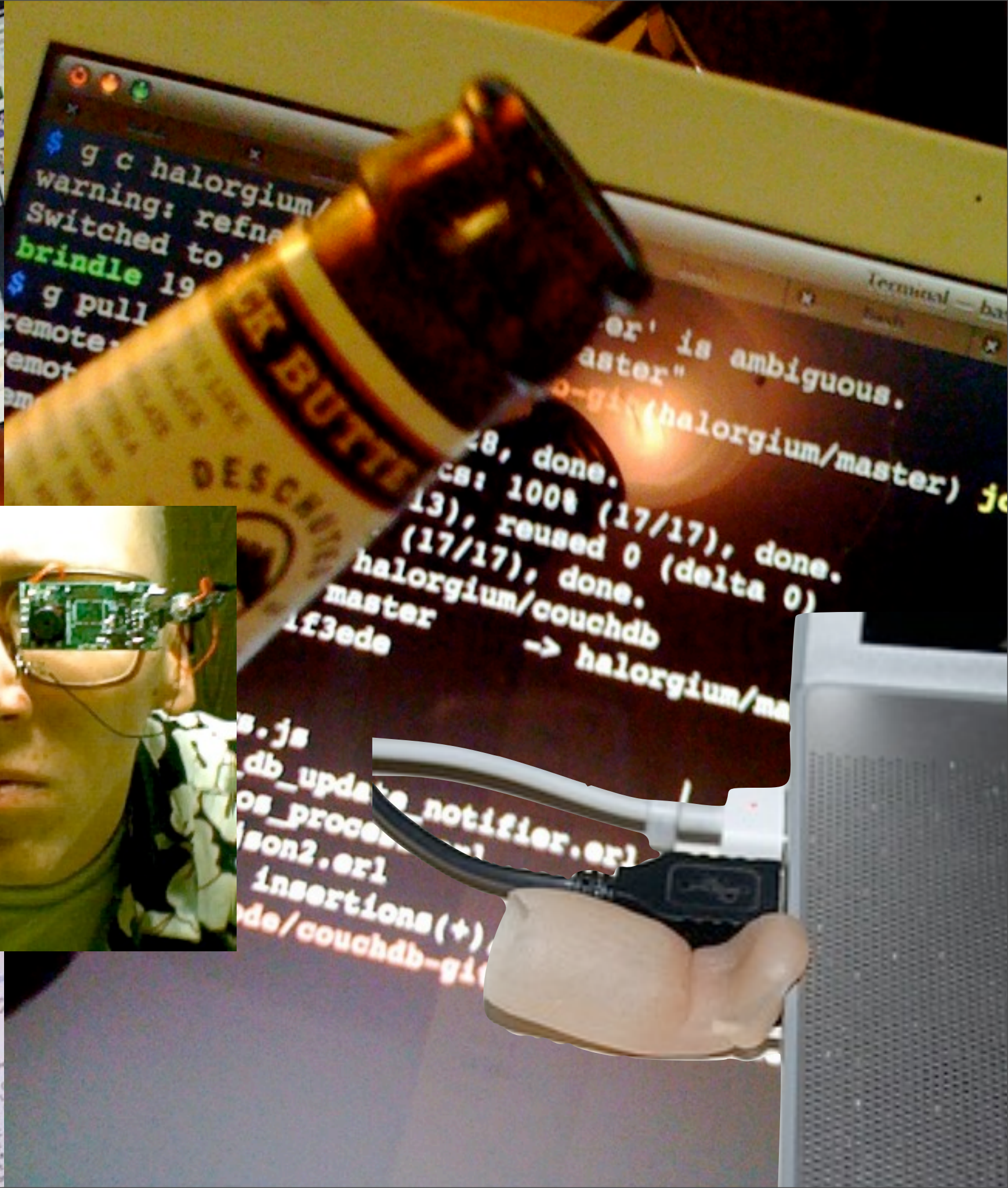
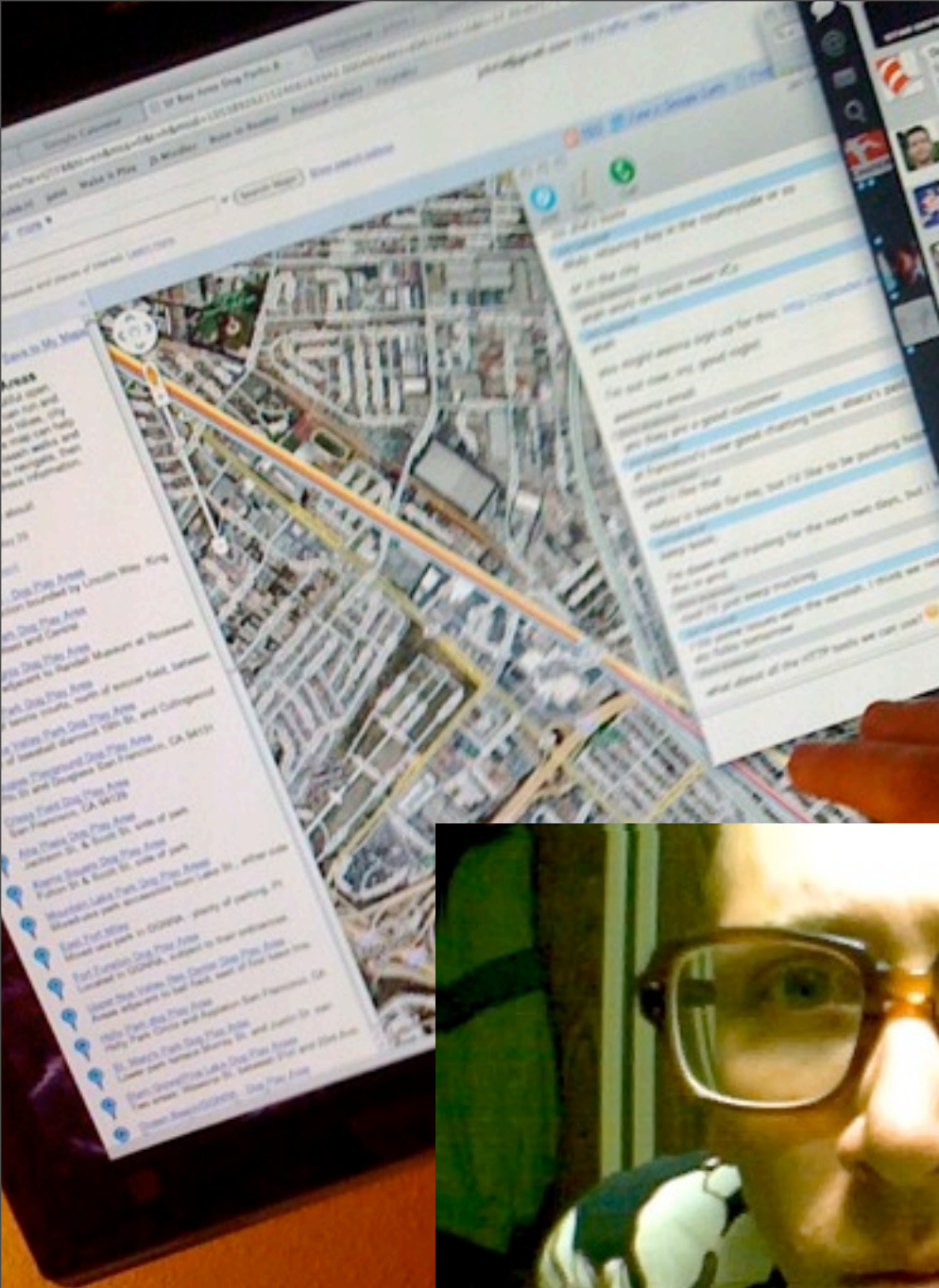
```
$ g c halorgium/
warning: refne
Switched to
brindle 19
$ g pull
remote:
remot

er' is ambiguous.
aster"
o-git(halorgium/master) jo

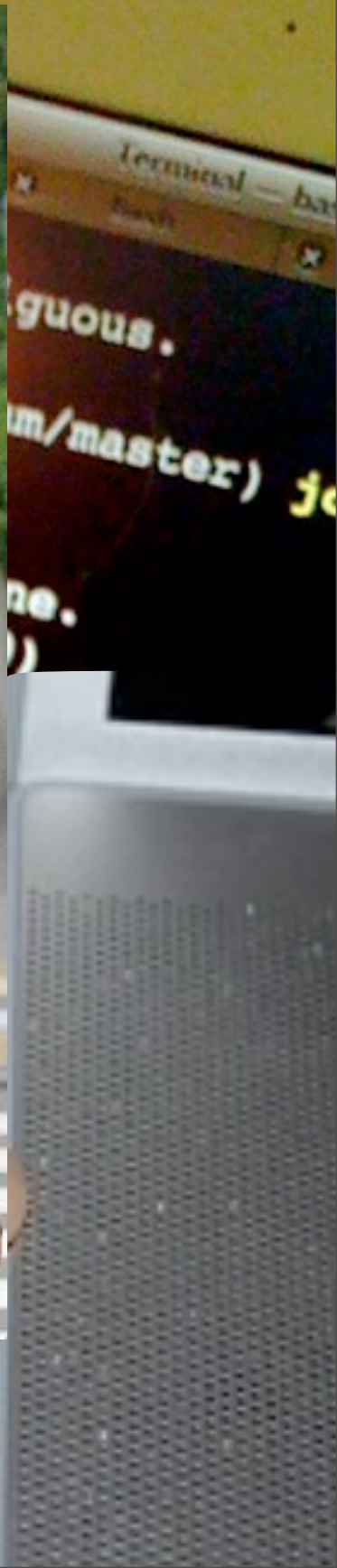
48, done.
cs: 100% (17/17), done.
13), reused 0 (delta 0)
(17/17), done.
halorgium/couchdb
master
f3ede
-> halorgium/master

ews.js
ouch_db_update_notifier.erl
ouch_os_process.erl
mochijson2.erl
ed, 34 insertions(+), 0 deletions(-)
09 -/code/couchdb-git(halorgium/master) johris
```





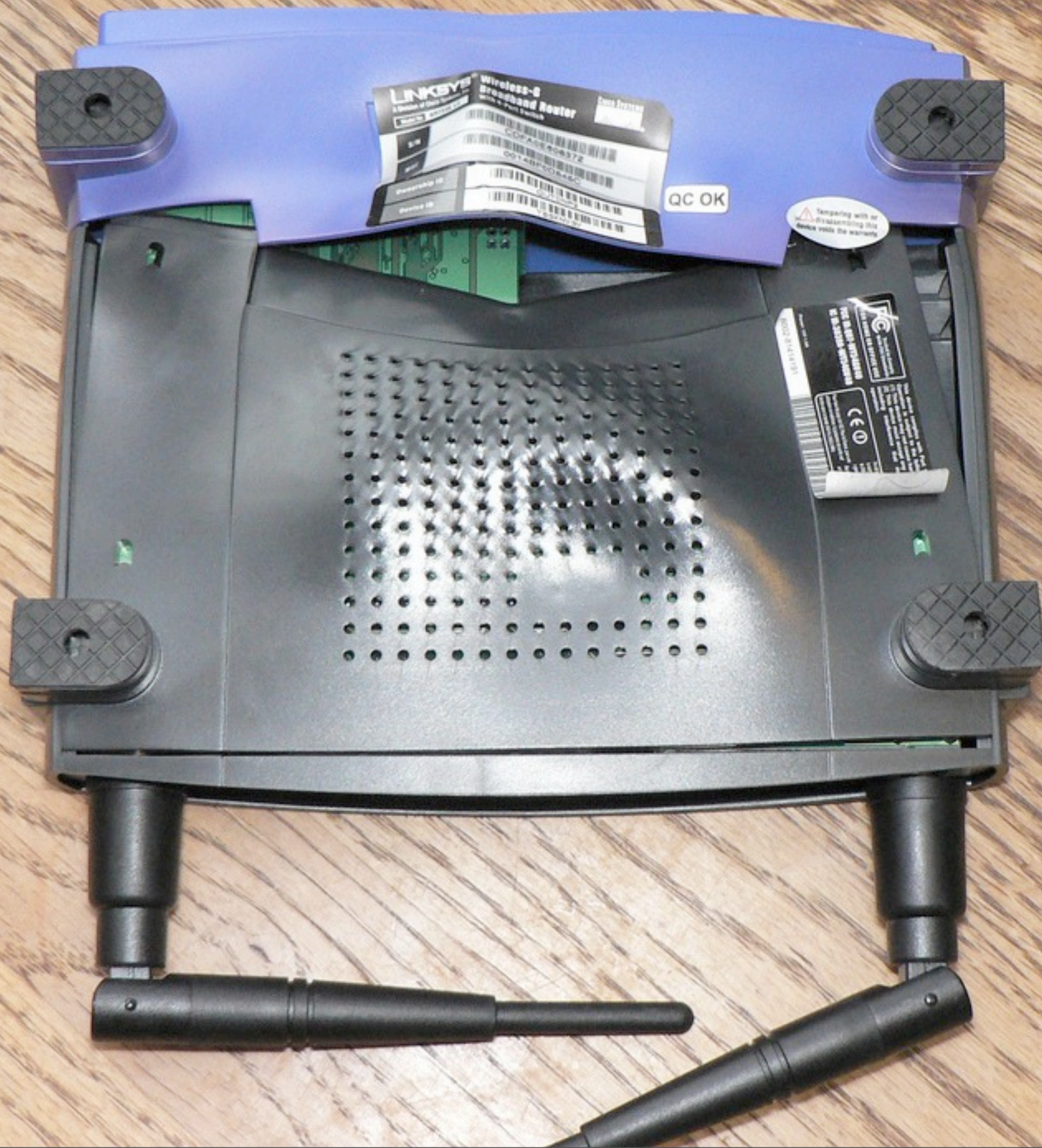














“no bars”



FREMONT  
5 CARS - BOARD CENTER  
PLATFORM 2

1808

1859



# Latency Sucks

$$rt^b = rt^a + \sum_{i \in \text{Links}} (l_i^b - l_i^a) \cdot \vec{\nabla} \bar{rt}(l_i)$$

# “Ground Computing”

@jhuggins



# Scaling Down

4 MB RAM





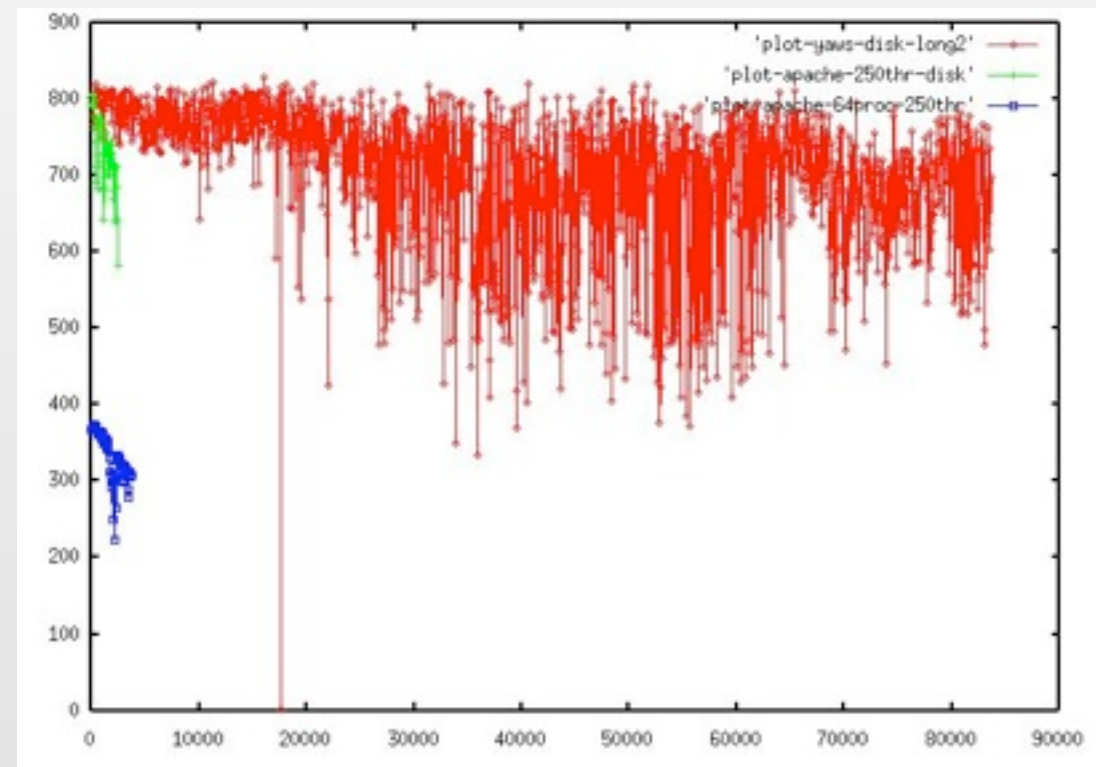
# Erlang

Parallel

Fault tolerant

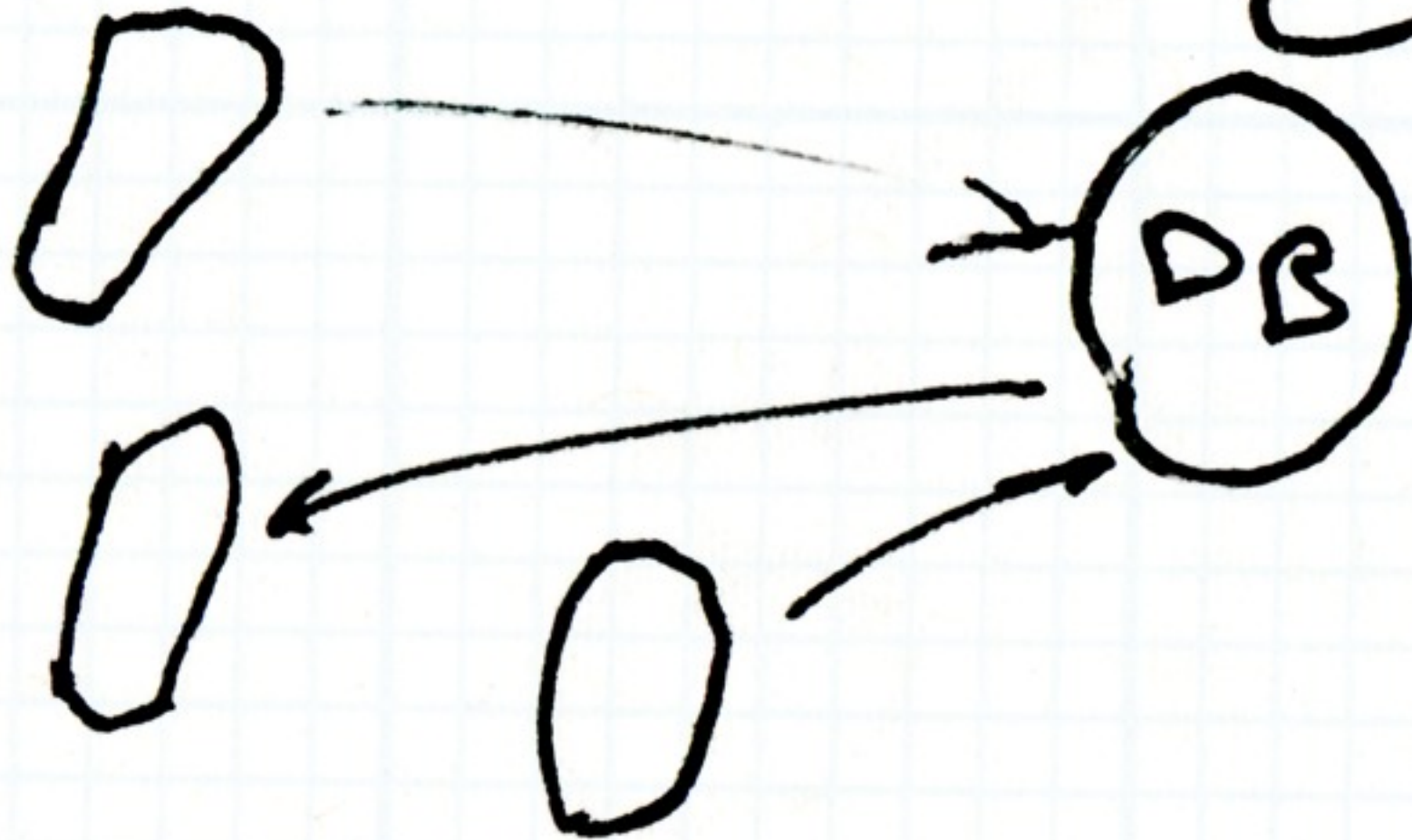
Addictive

Ninja Syntax



<http://www.sics.se/~joe/apachevsyaws.html>

# Concurrency



"smart phones"

## BrowserCouch Test Suite

This file contains a simple test runner and suite for BrowserCouch.

The suite can be run [here](#).

### The Runner

The `Testing` namespace is a simple test framework that supports the skipping of tests (for when the host system doesn't support required functionality for a test's execution), as well as asynchronous tests.

It doesn't currently have the ability to detect when an asynchronous test has failed, however.

### Browser Couch Tests

You can read the source code documentation for these tests [here](#).

```
testDictionary OK
testViewMap_async OK
testViewMapFindRow_async OK
testViewProgress_async OK
testViewMapReduceFindRow_async OK
testViewMapReduceWebWorker_async running
testViewMapReduce_async pending
testLocalStorage_async pending
```

```
testDictionary OK
testViewMap_async OK
testViewMapFindRow_async OK
testViewProgress_async OK
testViewMapReduceFindRow_async OK
testViewMapReduceWebWorker_async running
testViewMapReduce_async pending
testLocalStorage_async pending
```

### The Suite

The `Tests` namespace contains the actual testing suite for BrowserCouch.

# Browser Couch

JavaScript port

Uses HTML5 storage

Replicates with CouchDB

<http://hg.toolness.com/browser-couch/>

```
var Testing = {
  run: function(listener, container, setTimeout) {
    if (!setTimeout)
      setTimeout = window.setTimeout;

    var tests = [];

    for (name in container)
      if (name.indexOf("test") == "0") {
        var test = {
          name: name,
          func: container[name],
          isAsync: name.indexOf("_async") != -1,
          id: tests.length,
          assertEquals: function assertEquals(a, b) {
            if (a != b)
              throw new Error(a + " != " + b);
          }
        };
        tests.push(test);
      }

    listener.onReady(tests);
    var nextTest = 0;

    function runNextTest() {
      if (nextTest < tests.length) {
        var test = tests[nextTest];
        listener.onRun(test);
        test.skip = function() {
          listener.onSkip(this);
          setTimeout(runNextTest, 0);
        };
        test.done = function() {
          listener.onFinish(this);
          setTimeout(runNextTest, 0);
        };
        test.func.call(container, test);
        if (!test.isAsync)
          test.done();
        nextTest++;
      }
    }

    runNextTest();
  }
};

var Tests = {
  testDictionary: function(self) {
    var dict = new BrowserCouch._Dictionary();
    dict.set('foo', {a: 'hello'});
    dict.set('bar', {b: 'goodbye'});
    self.assertEqual(dict.get('foo').a, 'hello');
    self.assertEqual(dict.get('bar').b, 'goodbye');
    self.assertEqual(dict.getKeys().length, 2);
    self.assertEqual(dict.has('foo'), true);
    self.assertEqual(dict.has('bar'), true);
    self.assertEqual(dict.has('spatula'), false);
    dict.remove('bar');
    self.assertEqual(dict.getKeys().length, 1);
    self.assertEqual(dict.has('foo'), true);
  },
  _setupTestDb: function(cb) {
    var documents = this._testDbContents;
    BrowserCouch.get(
      "blarg",
      function(db) {
        db.wipe(
          function() {
            db.put(
              documents,
              function() {
                ModuleLoader.require(
                  "JSON",
                  function() { cb(db); }
                );
              }
            );
          }
        );
      }
    );
    new FakeStorage()
  },
  _testDbContents: [{id: "monkey",
    content: "hello there dude"},
    {id: "shovel",
    content: "hello there dude"}]
```

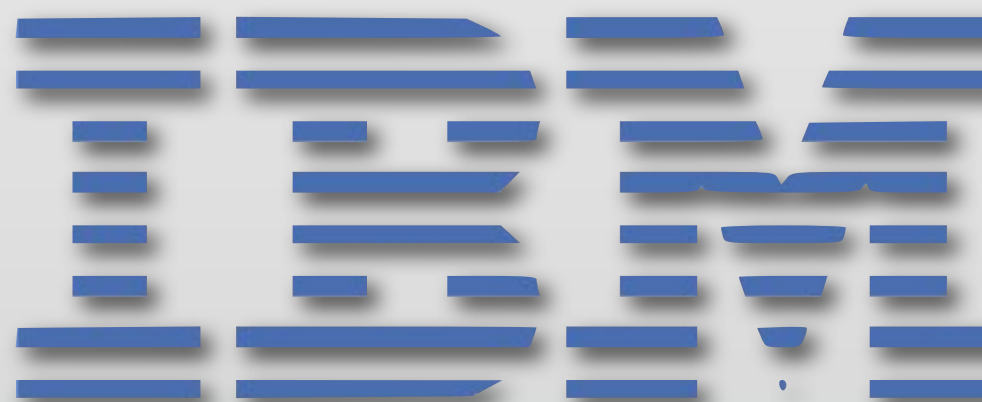


A dramatic photograph of a storm at night or dusk. The sky is dark and filled with heavy, dark clouds. A massive, bright white and yellow lightning bolt strikes down from a large, glowing cloud mass in the upper center. Several other smaller lightning bolts are visible, branching out from the main cloud. The foreground is dark, showing the silhouettes of trees and hills. The overall mood is intense and powerful.

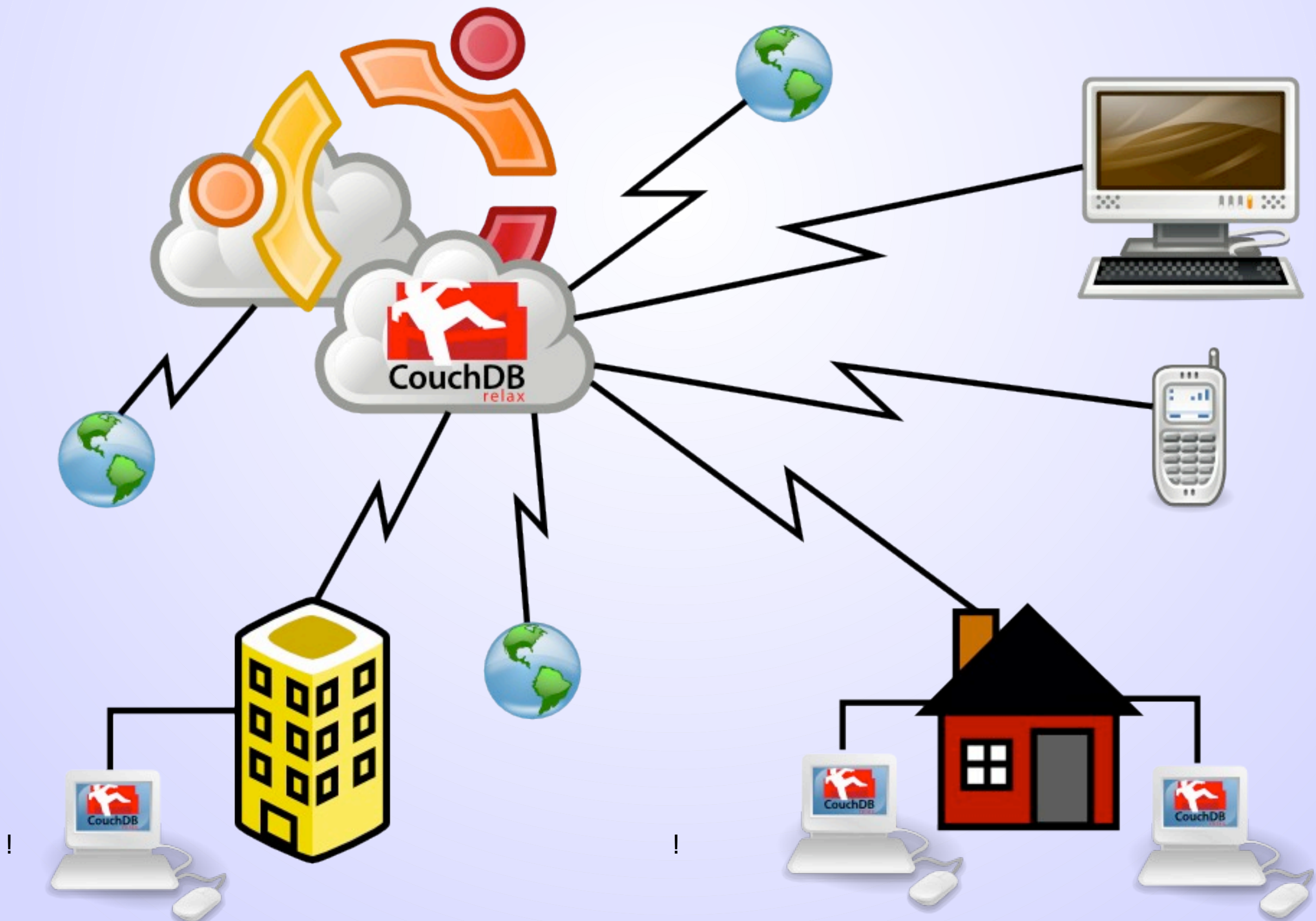
**“Obvious.”**

BBC

meebo












COUCH DB  
to  
the  
EDGE



A hand-drawn diagram on a grid background. It features a large, irregular polygon on the left side, composed of several connected line segments. To the right of this polygon, the text 'COUCH DB' is written in a large, bold, sans-serif font. Below 'COUCH DB', the words 'to' and 'the' are written in a smaller, cursive script, followed by the word 'EDGE' in a large, bold, sans-serif font. The entire diagram is enclosed within a simple rectangular frame.

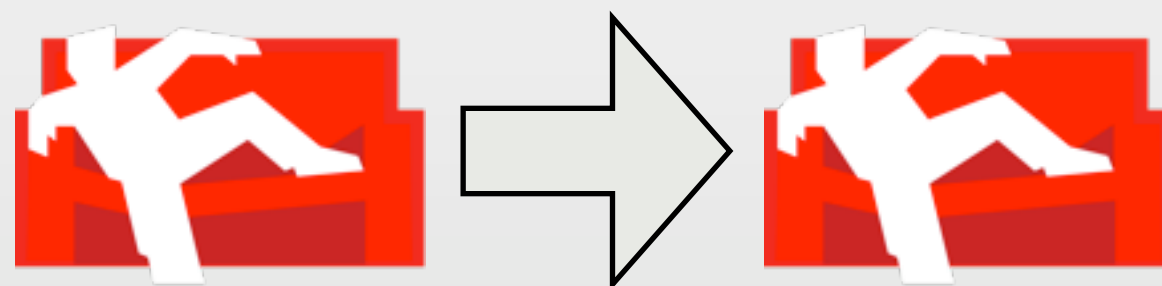
# Incremental Peer-Based Replication



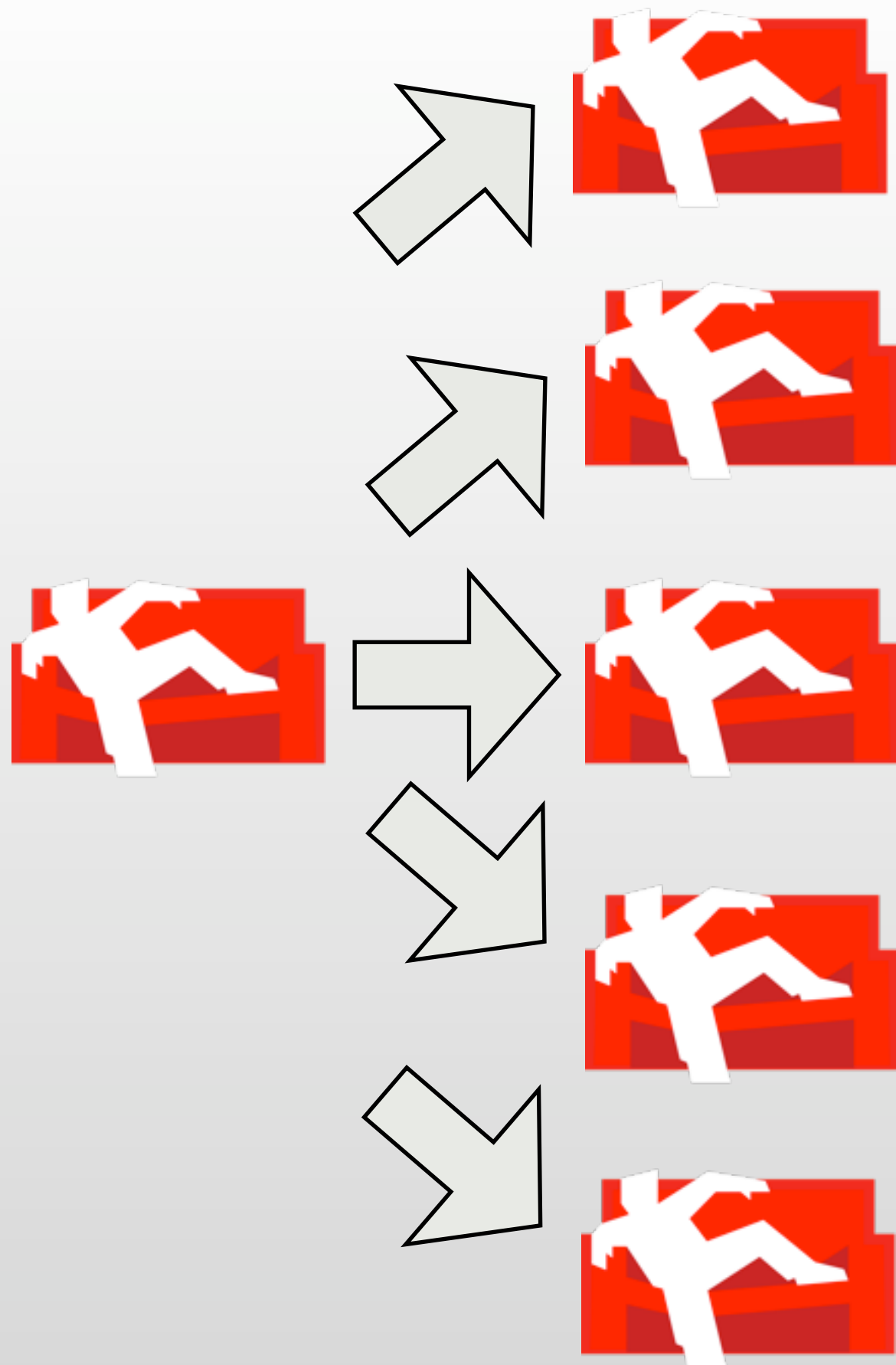
```
curl -X POST http://127.0.0.1:5984/_replicate -d  
'{"source":"http://couch.example.com:5984/remote-db",  
"target":"local-db"}'
```

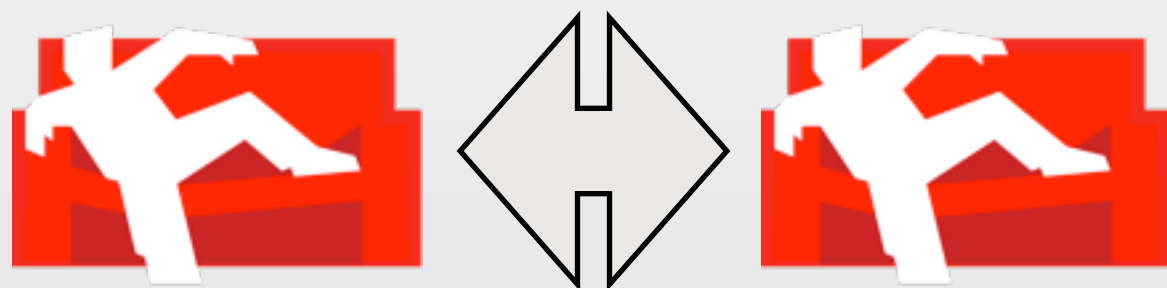




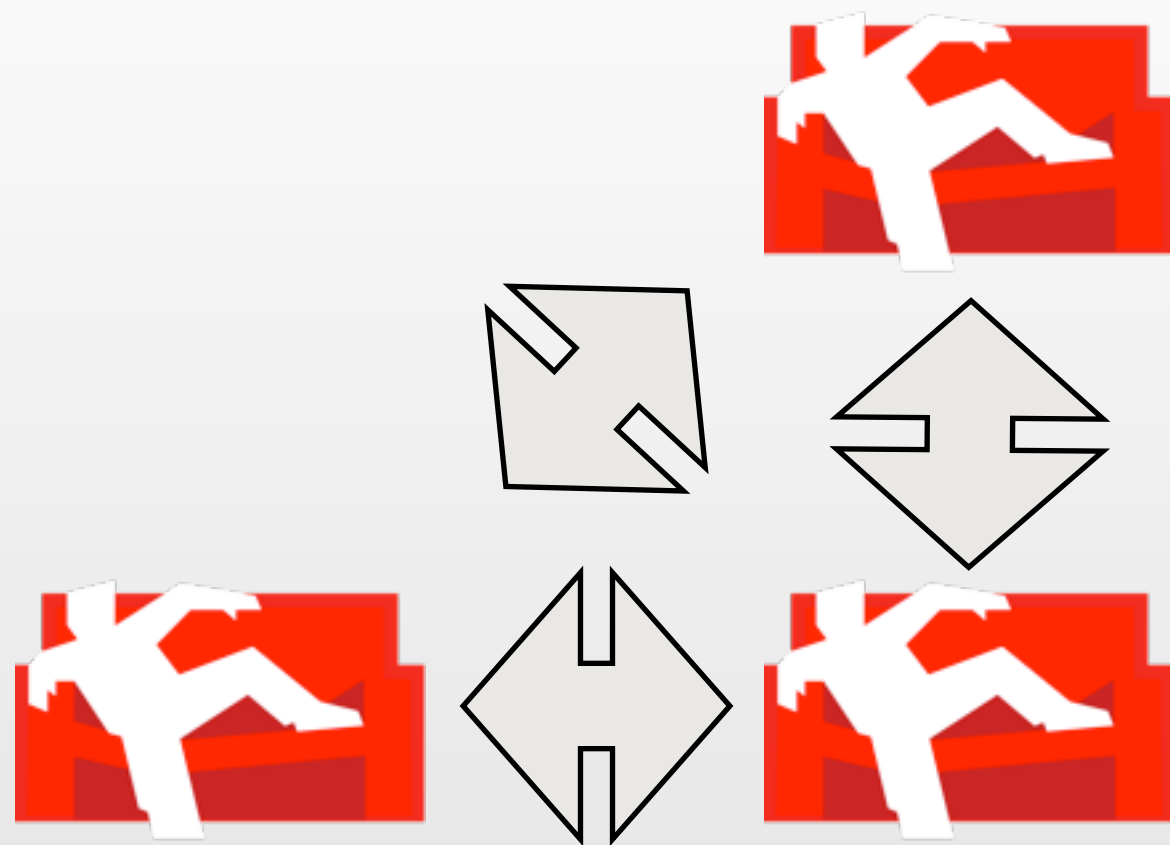


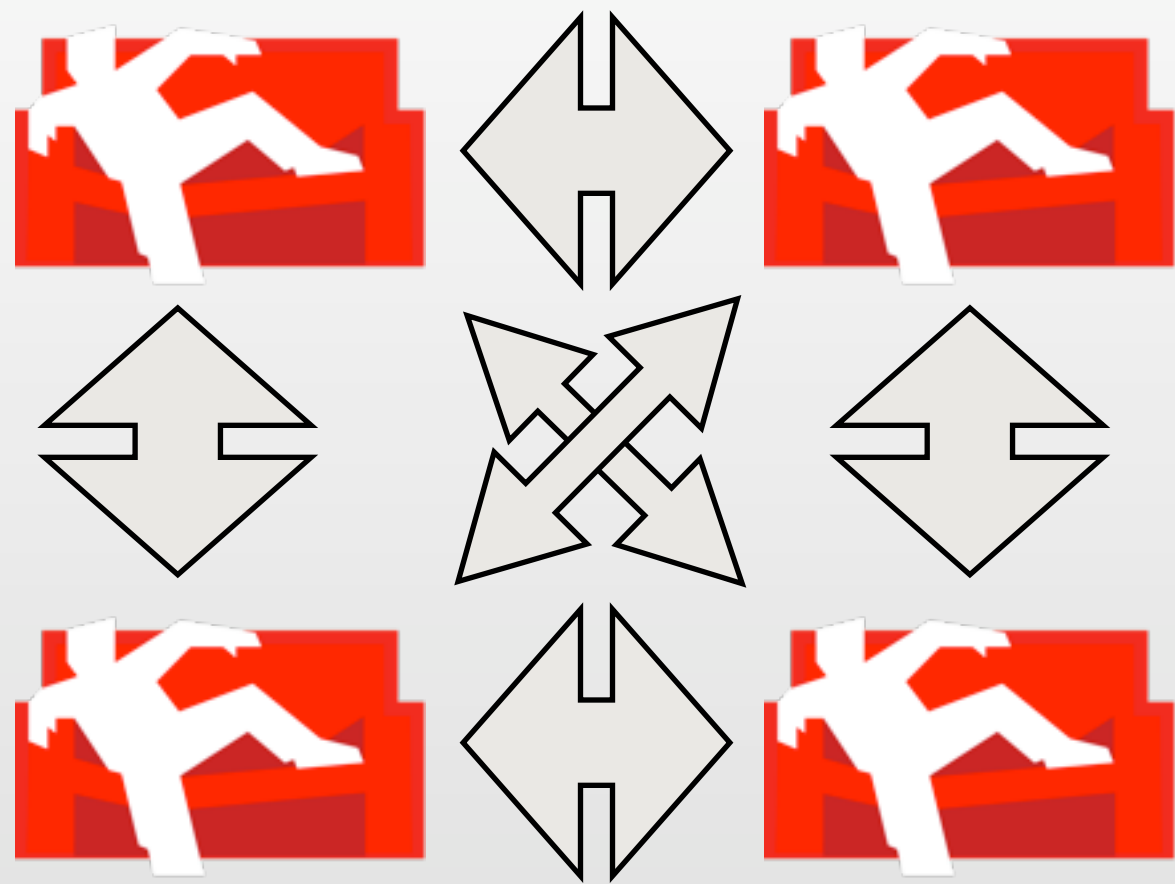


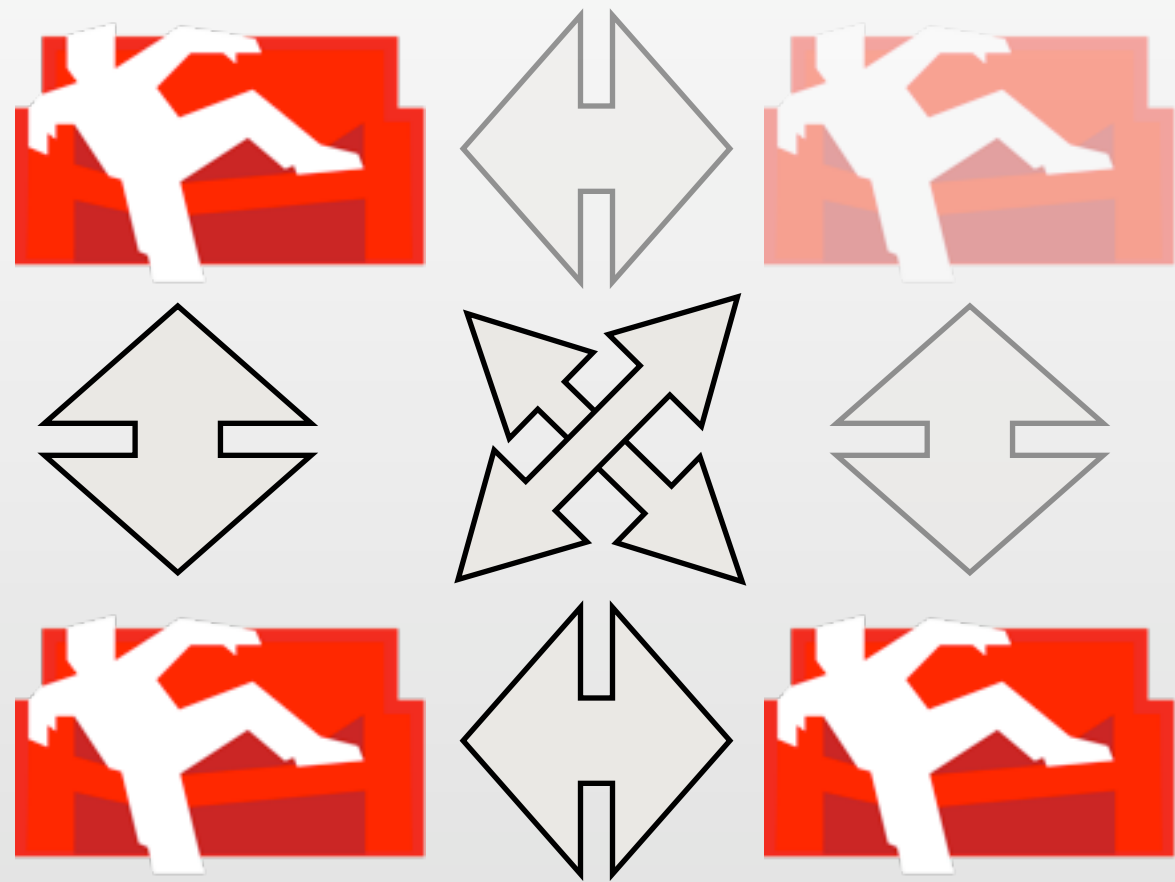




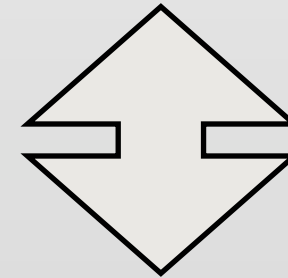
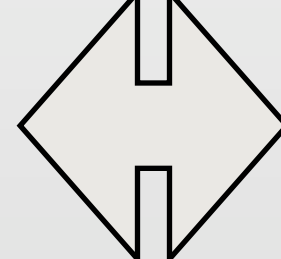
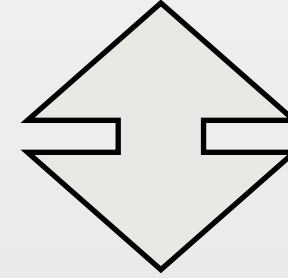
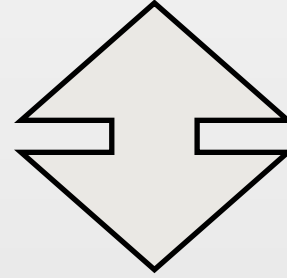
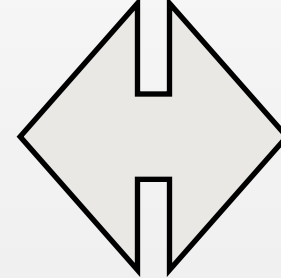
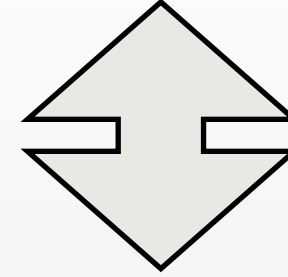


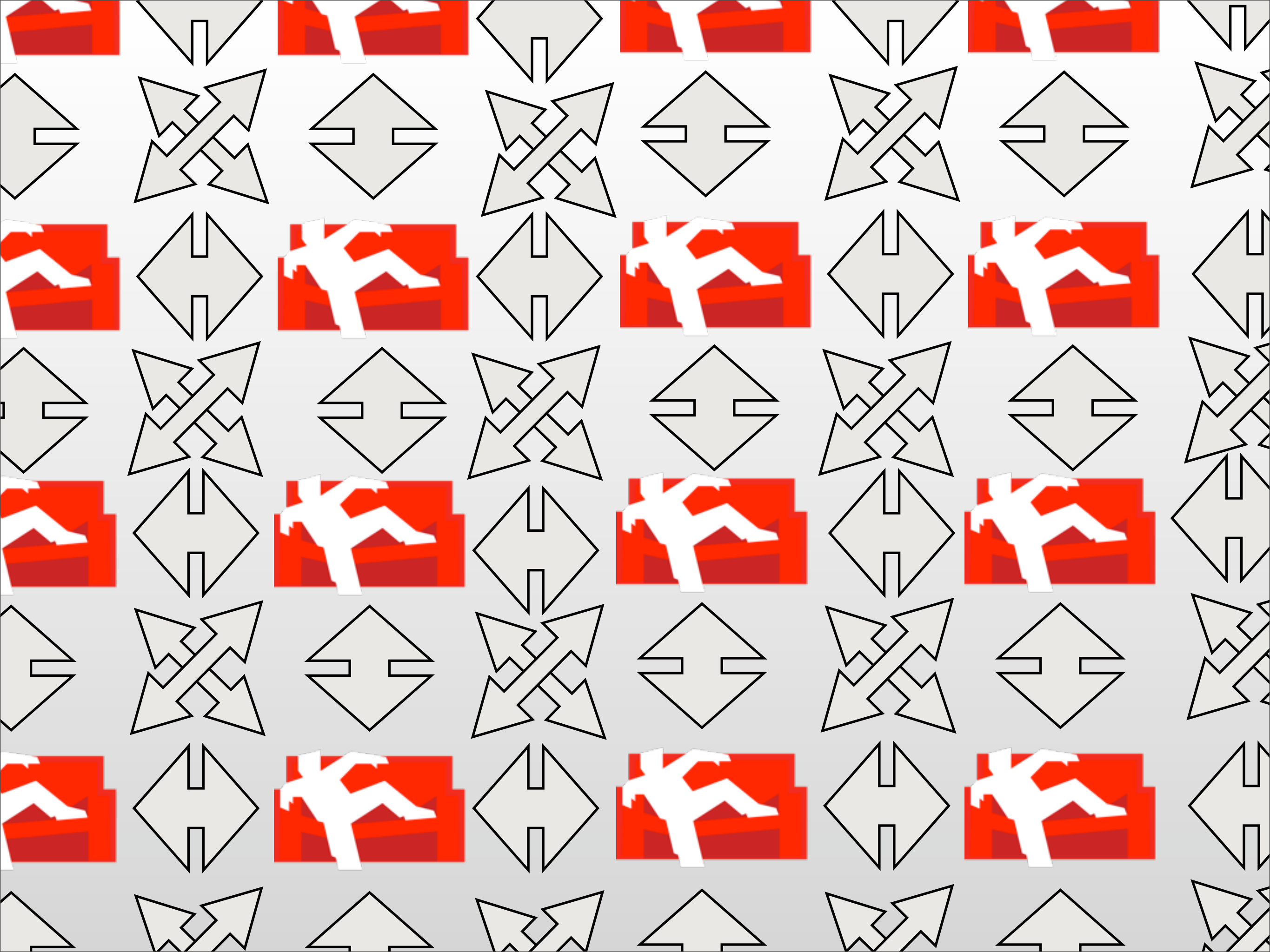












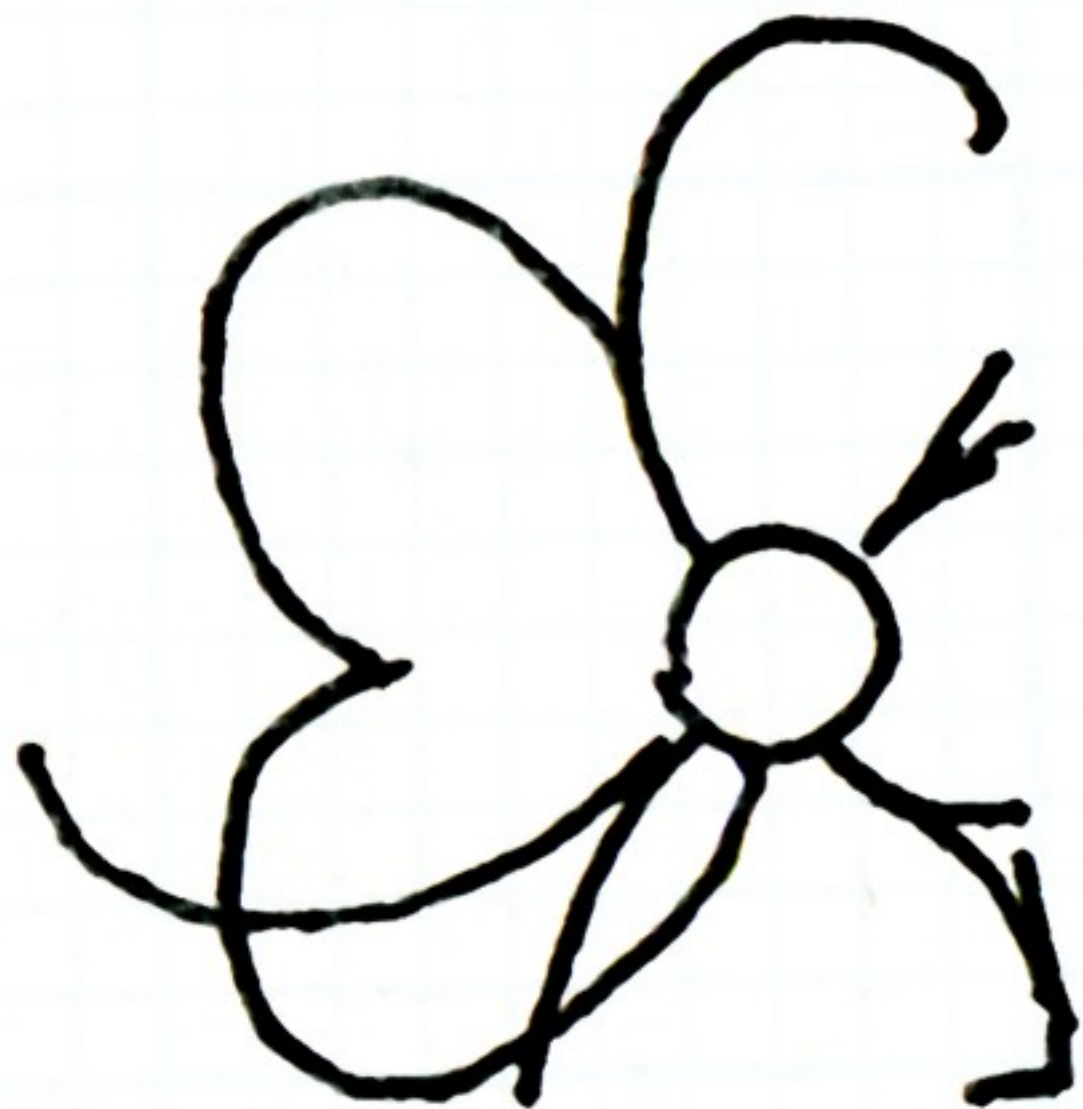
20



Programming

at the

Edge

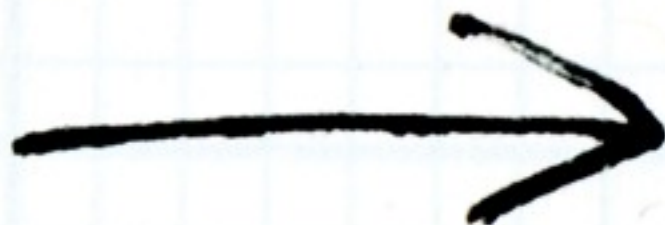


Applications

are

Documents.

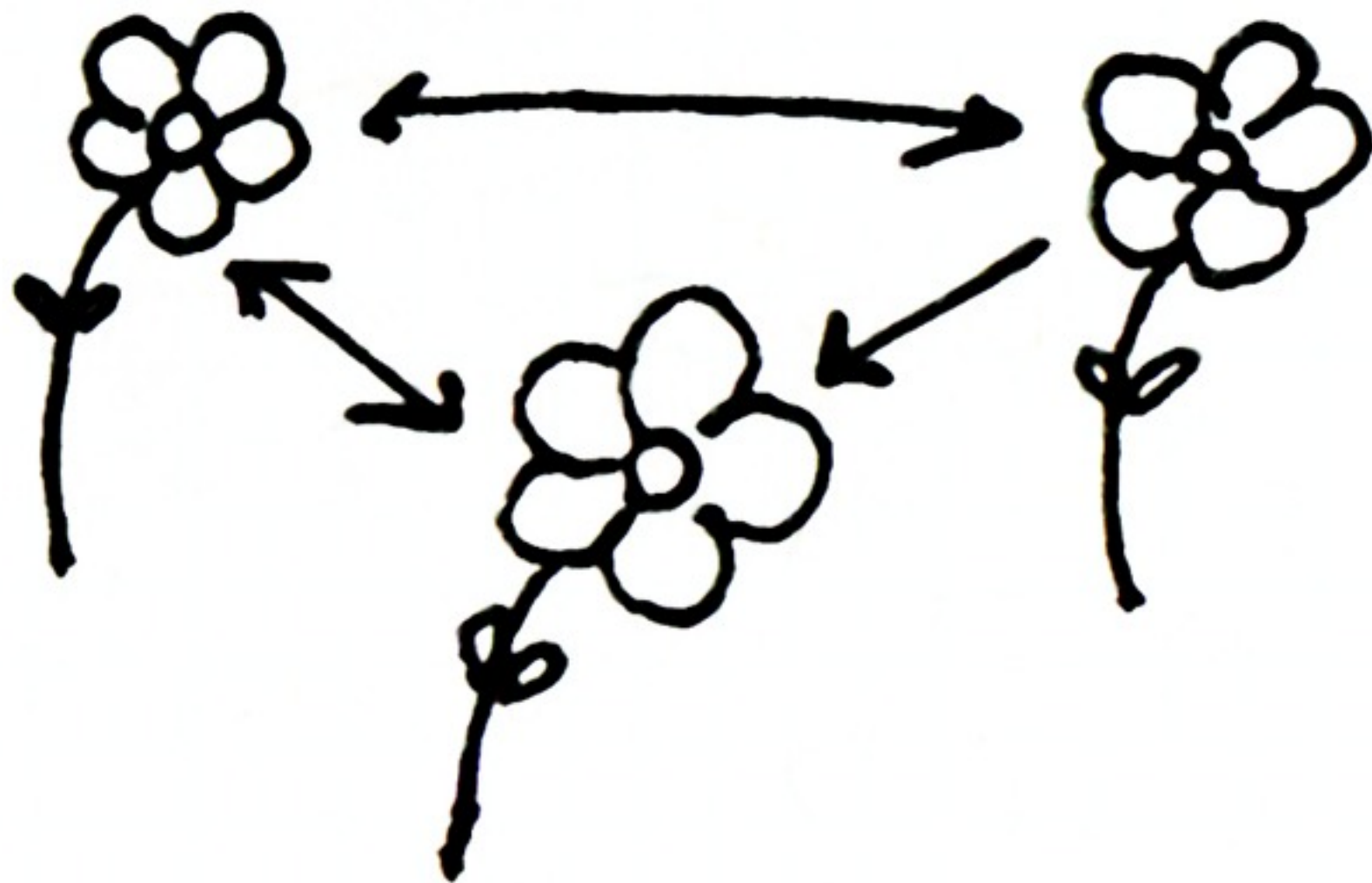






- map/reduce
- validations
- render html

# Replication



"normal"



server

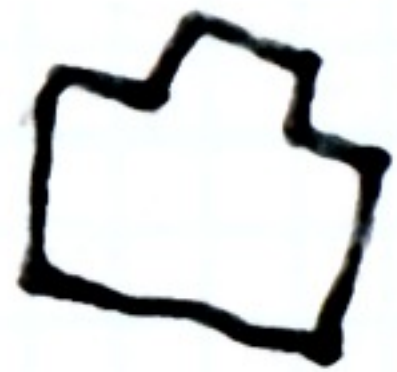


offline

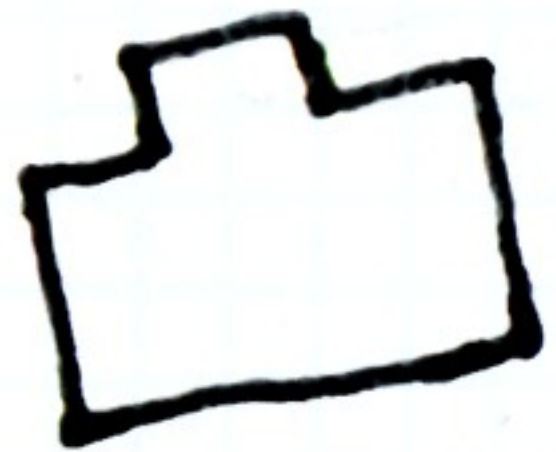
mode

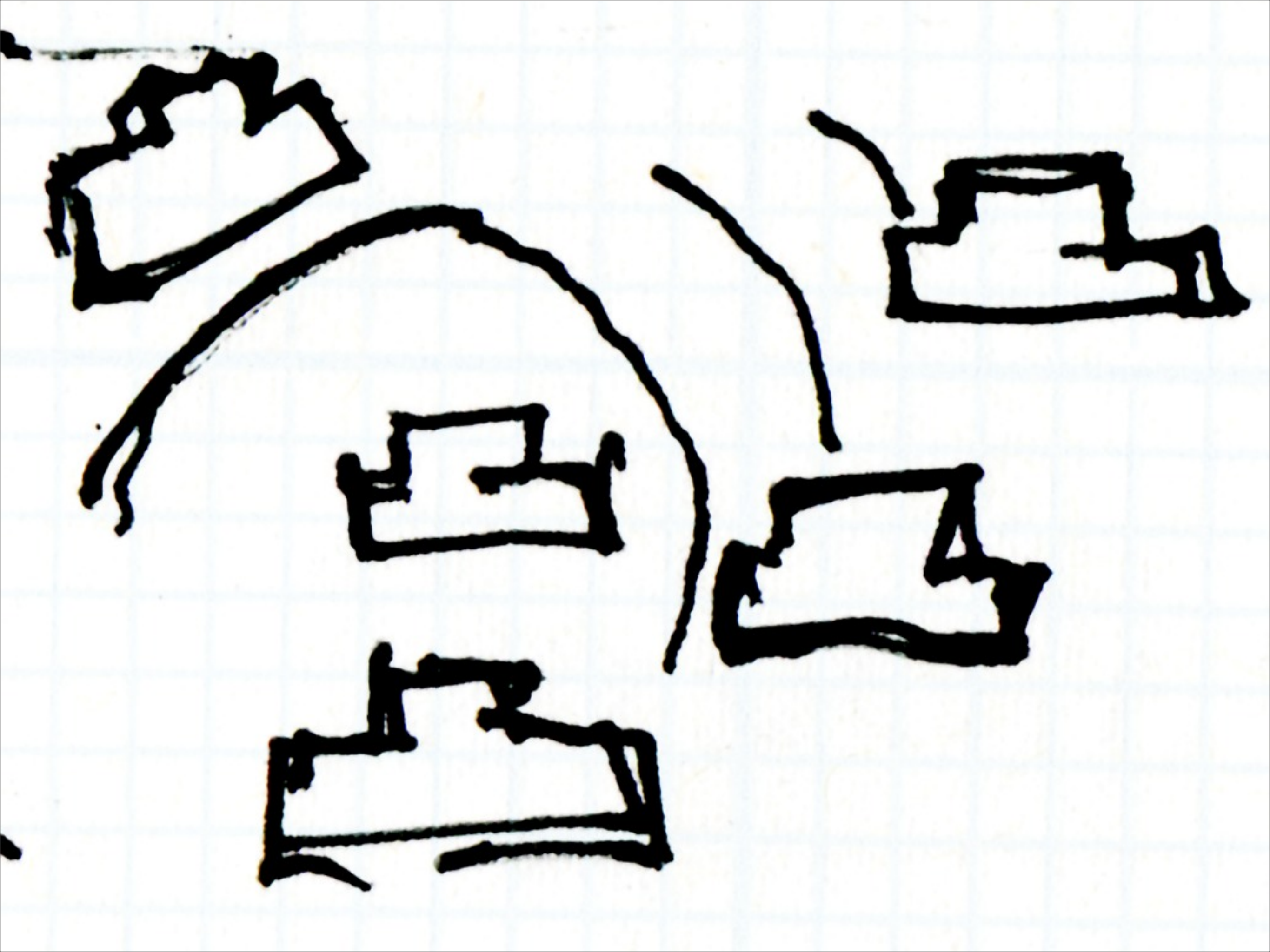


the p2p



web.







Context is  
Expensive



# Identity

Can we trust  
intermediate servers?

Key-Based Identity

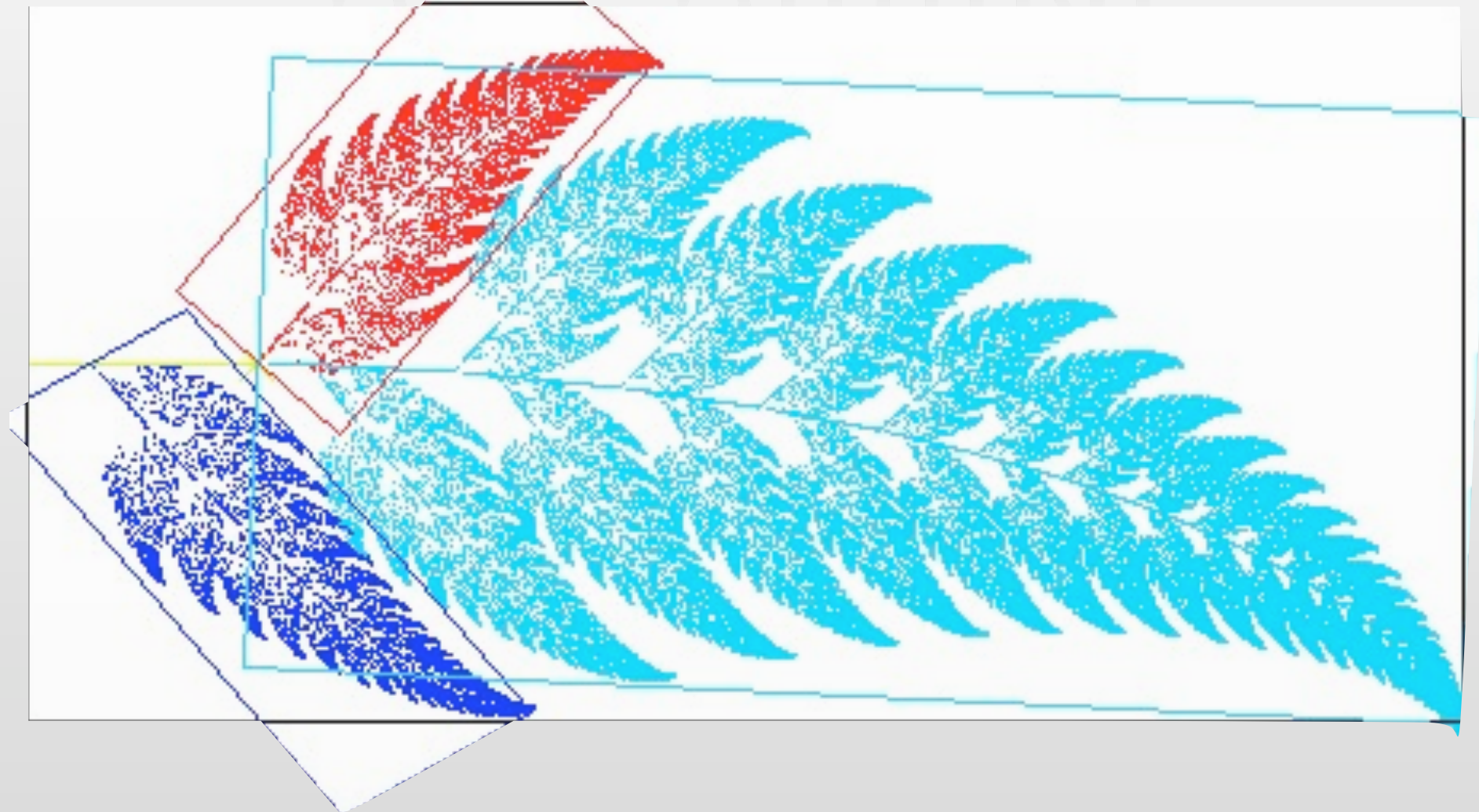
Signed Documents

Web of Trust



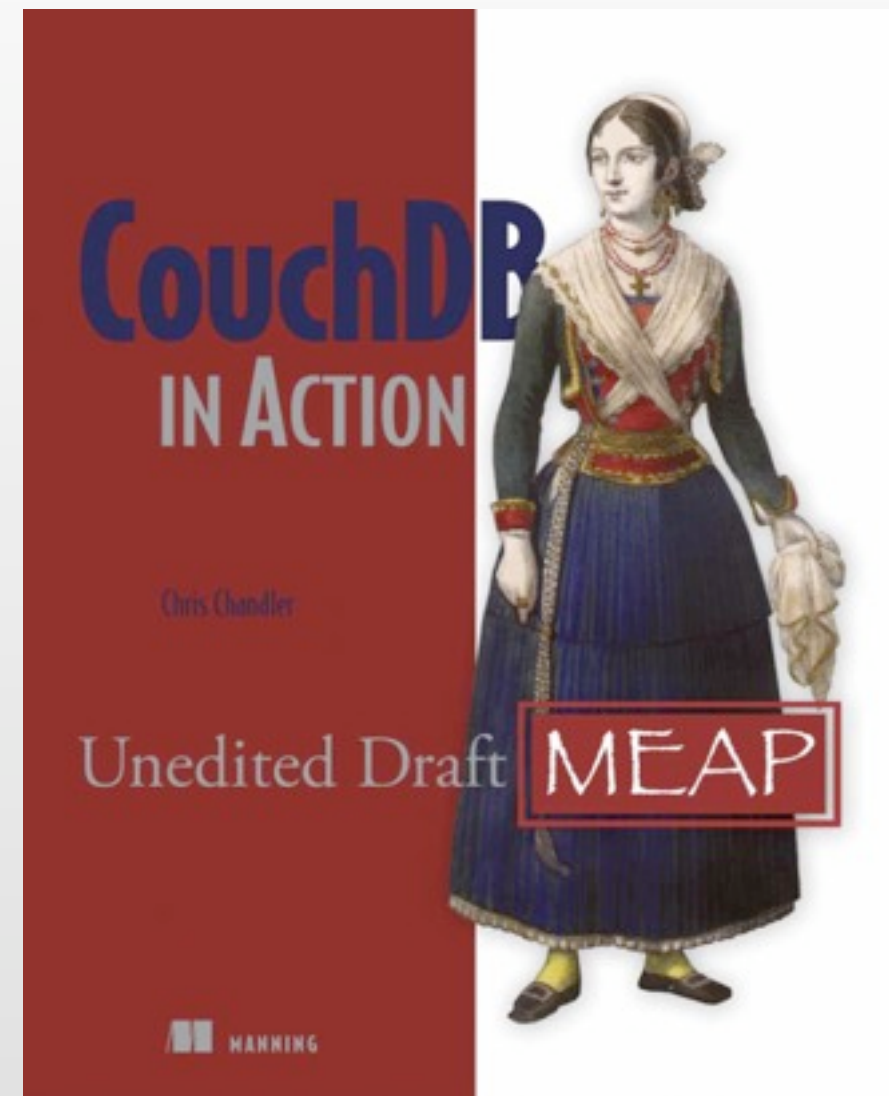
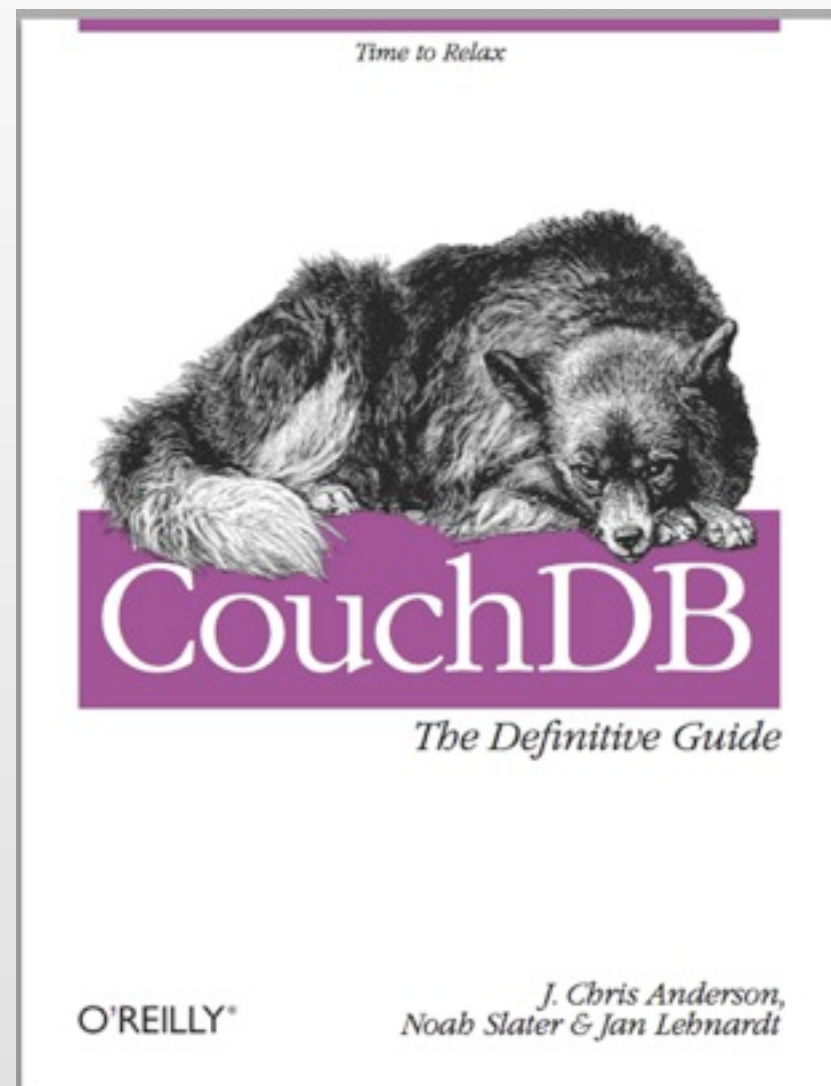
# Self-Similar

Self-Similar  
Self-Similar  
Self-Similar





@couchdbinaction



<http://books.couchdb.org/relax>

# Resources

@CouchDB

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

Dress like a Couch: <http://shop.couchdb.com>

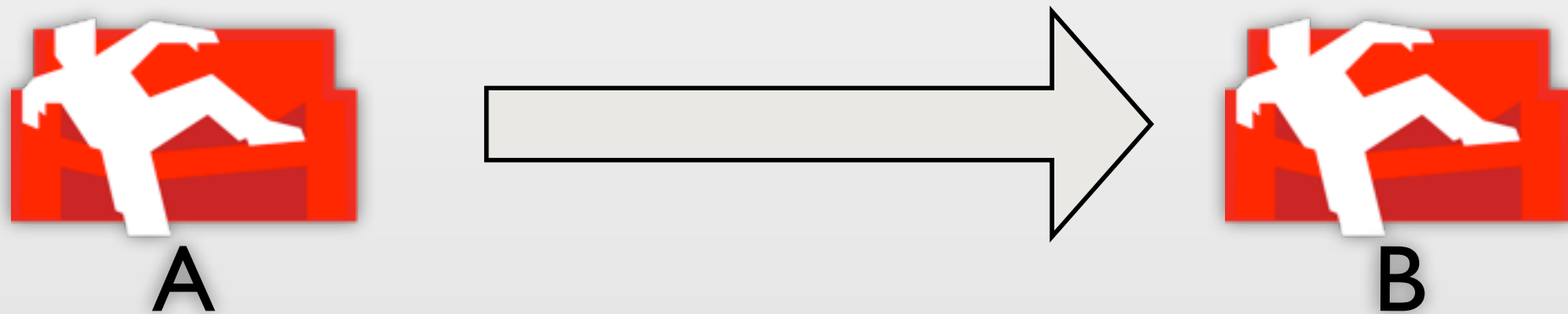
<http://planet.couchdb.org/>

<https://peepcode.com/products/couchdb-with-rails>





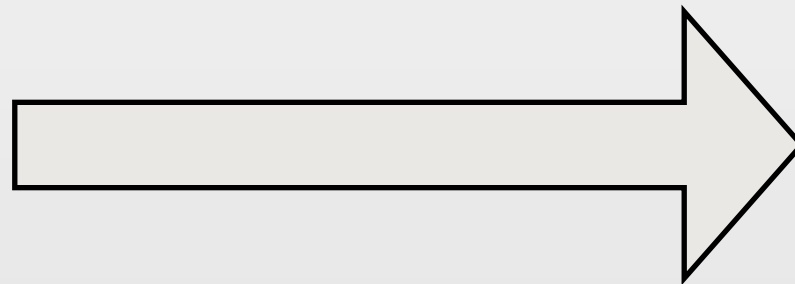
# Conflict resolution by example



# Conflict resolution by example



A



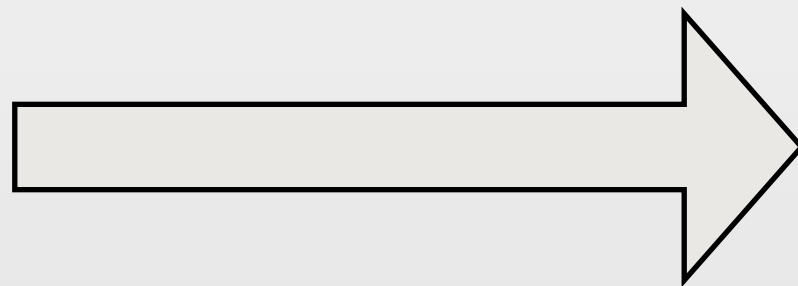
B



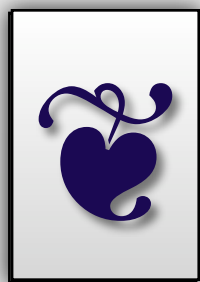
# Conflict resolution by example



A

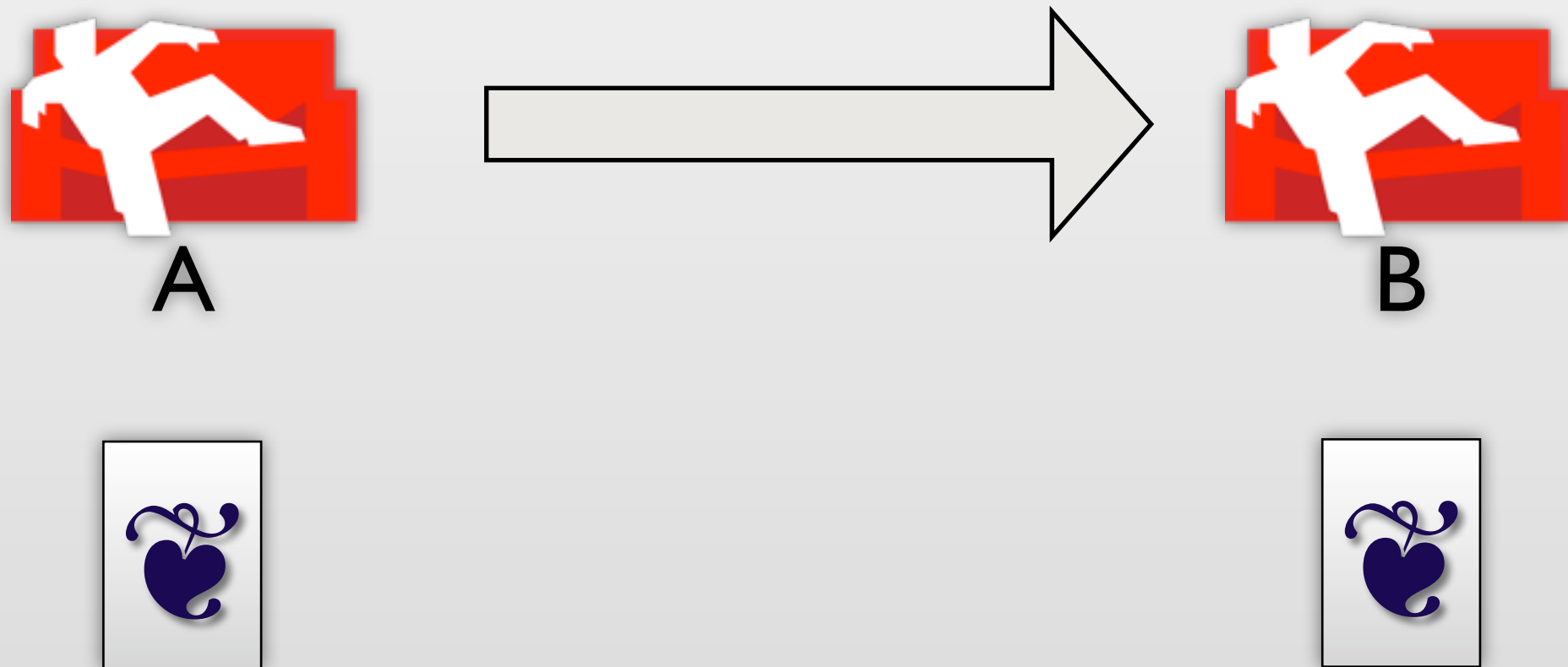


B





# Conflict resolution by example



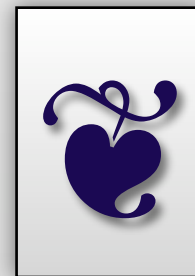
# Conflict resolution by example



A



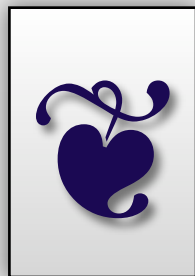
B



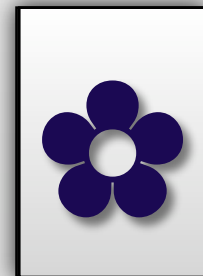
# Conflict resolution by example



A



B

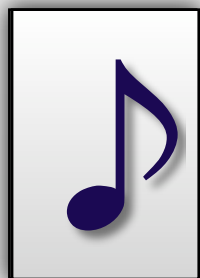




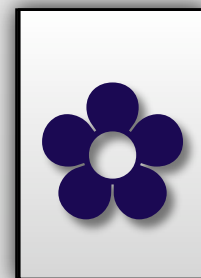
# Conflict resolution by example



A



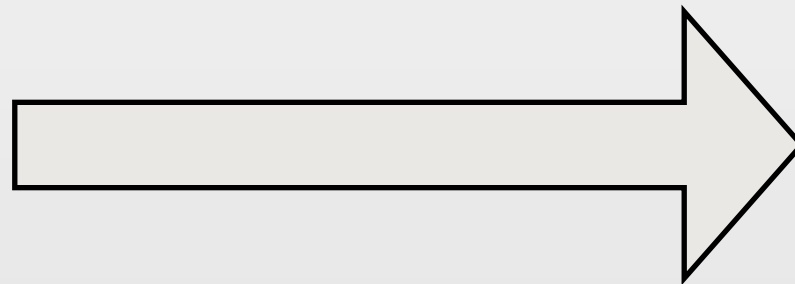
B



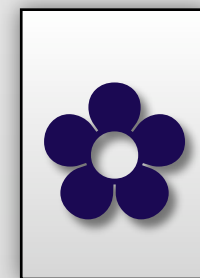
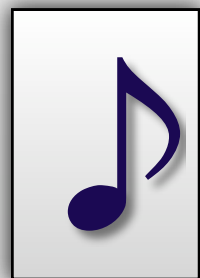
# Conflict resolution by example



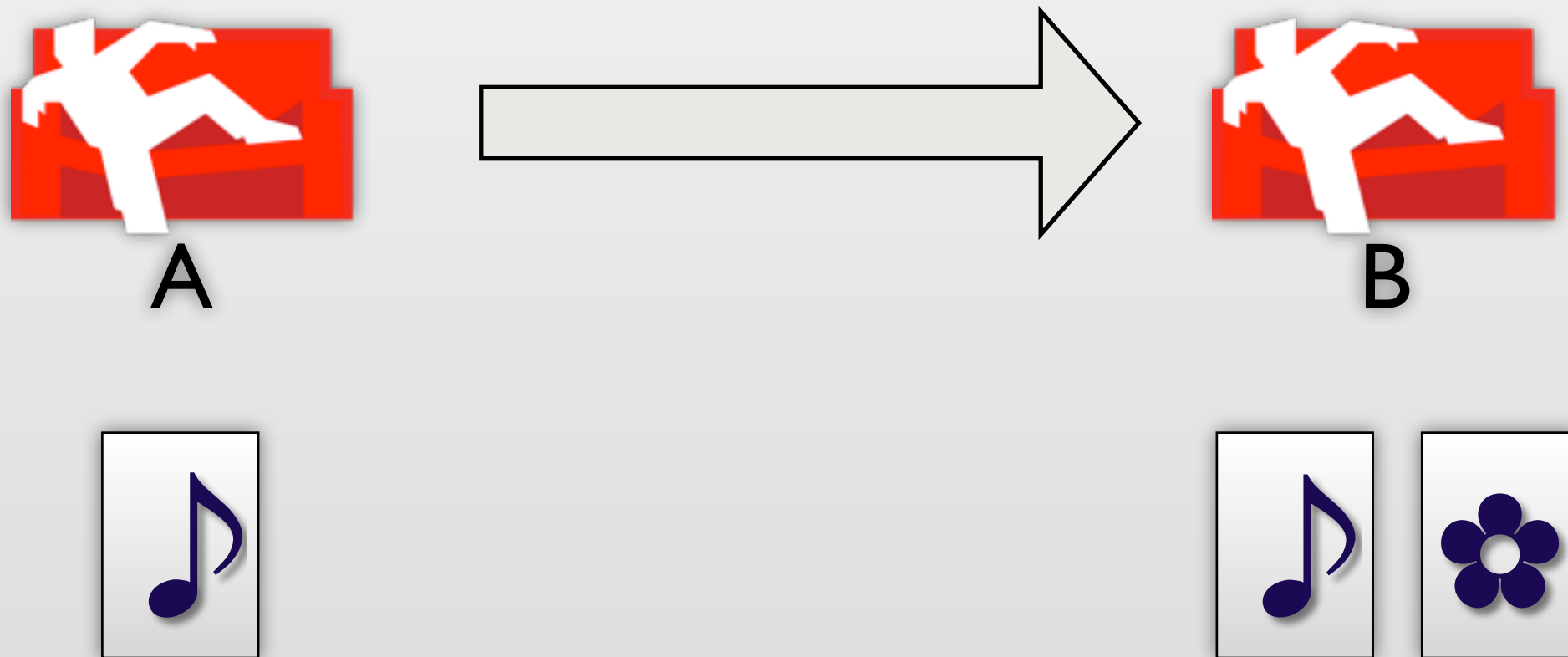
A



B

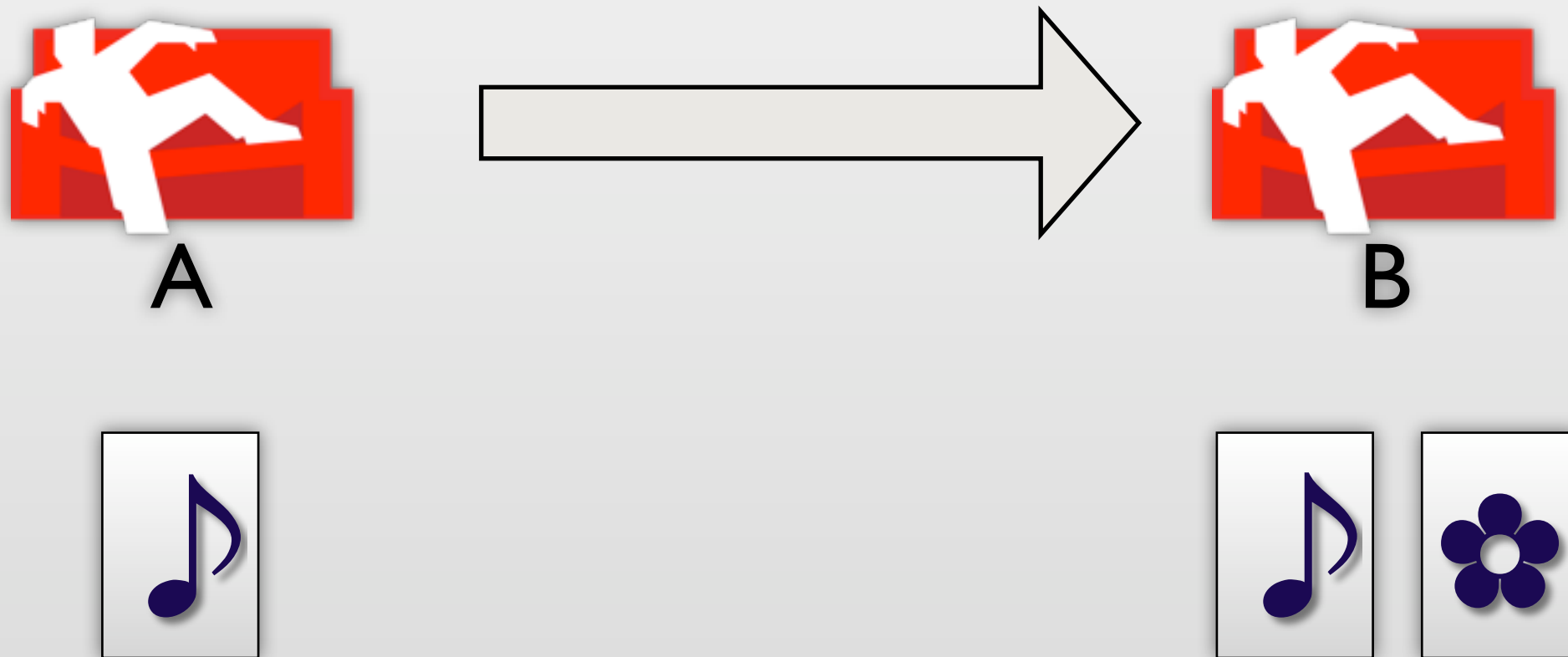


# Conflict resolution by example





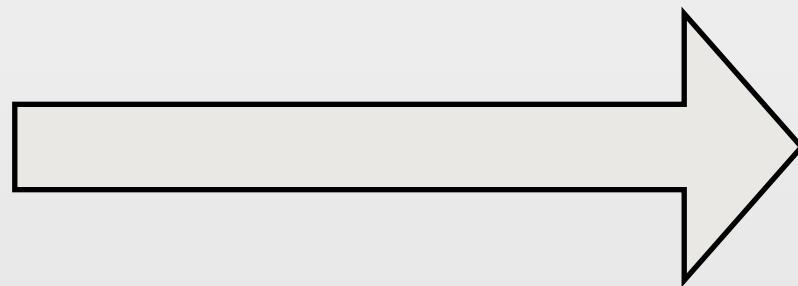
# Conflict resolution by example



# Conflict resolution by example



A



B



# couch.io

---

Berlin – London – Portland



Thanks!