



ConchDB

Who's talking?

- ✦ Jan Lehnardt
- ✦ CouchDB Developer
- ✦ jan@apache.org

And you?

Who hasn't seen previous talk? Joe's Talk? Tim's?

Number Bragging

- Silly read-only benchmark with memory saturation
- 2,500 req/s sustained on a 2Ghz dual core Athlon

Number Bragging

- Silly read-only benchmark with memory saturation
- 2,500 req/s sustained on a 2Ghz dual core Athlon
- **Using 9.8 MB RAM**



Views

of Keys and Values

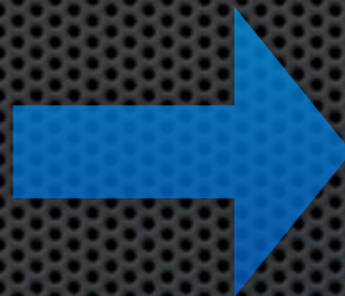
Views — Map Tags



Keys	Values
family	1
friends	1
friends	1
work	1
work	1
youtube	1
...	...

Views — Reduce Tag Count

Keys	Values
family	1
friends	1
friends	1
work	1
work	1
youtube	1
...	...



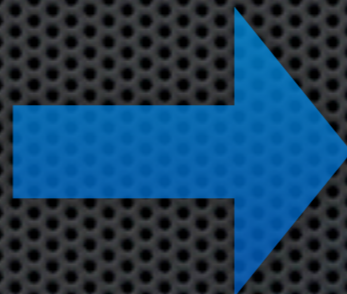
Keys	Values
family	1
friends	2
work	2
youtube	1
...	...

Views — Map Tags

```
function (doc) {  
    for(var i in doc.tags)  
        emit(doc.tags[i], 1);  
}
```


Views — Reduce Tag Count

Keys	Values
family	1
friends	1
friends	1
work	1
work	1
youtube	1
...	...



Keys	Values
family	1
friends	2
work	2
youtube	1
...	...

Views — Reduce Tag Count

```
function (key, values)
{
    var sum = 0;
    for(var i in values)
        sum += values[i];
    return sum;
}
```

View Examples – Docs by Date

```
function(doc) {  
  emit(doc.date, doc.amount);  
}
```



[2007, 10, 12, 20, 13, 12]	3000
----------------------------	------

View Examples – Docs by Date

Map

Key	Value
[2007, 10, 12, 20, 13, 12]	3000
[2007, 10, 26, 08, 37, 55]	4000
[2008, 02, 03, 10, 22, 34]	2000
[2008, 05, 01, 14, 16, 11]	6000

View Examples – Docs by Date

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

View Examples – Docs by Date

Reduce

Key	Value
null	15000

View Examples – Docs by Date

Reduce with group_level = 1

Key	Value
[2007]	7000
[2008]	8000

View Examples – Docs by Date

Reduce with group_level = 2

Key	Value
[2007, 10]	7000
[2008, 02]	2000
[2008, 05]	6000

Just the beginning, knock yourself out, averages, standard deviation, jchris twitter tag cloud ranking.



Relation(ship)s

<http://flickr.com/photos/fazen>

Relation(ship)s

- ✦ JOINS please!
- ✦ What for?
- ✦ Get data that “belongs together”

Relation(ship)s — One Big Doc

```
{  
  "type": "person",  
  "name": "Darth Vader",  
  "children": [  
    {"name": "Luke" ...},  
    {"name": "Leia" ...}  
  ],  
  "dark_side": true  
}
```

Relation(ship)s — One Big Doc

- ✦ Pros: Easy – Cons: Bad with concurrent updates
- ✦ Use for: Low volume updates

Relation(ship)s — Master-Slave Docs

- ✦ Cons: A little more complex –
Pros: Fast; good with concurrent updates; gives you tree operations
- ✦ Use for: Everything else

Relation(ship)s

```
function(doc) {  
  if(doc.is_master) {  
    emit([doc._id, doc.date], doc);  
  } else {  
    emit([doc.master_id, doc.date], doc);  
  }  
}
```

Relation(ship)s

...	...
["BAAC67", "2008-09-21"]	{"is_parent", true}
["BAAC67", "2008-09-22"]	{"...", "..."}
["BAAC67", "2008-09-23"]	{"...", "..."}
["BAAC67", "2008-09-24"]	{"...", "..."}
["DBCA82", "2008-09-21"]	{"is_parent", true}
...	...

arbitrary trees, deeper nesting

Where is my auto_increment

- ✦ What is auto_increment?
- ✦ Unique identifier
- ✦ Sequence denominator

Where is my auto_increment?

what's it used for? identification, sequences / sorting
location prefixes, datacenters, digg, facebook

Where is my auto_increment?

- Documents have `_ids`

Where is my auto_increment?

- Documents have `_ids`
- Sequences in distributed databases are...

Where is my auto_increment?

- ✦ Documents have `_ids`
- ✦ Sequences in distributed databases are...
- ✦ ...not

Where is my auto_increment?

- ✦ Documents have `_ids`
- ✦ Sequences in distributed databases are...
- ✦ ...not
- ✦ Use natural keys

Transactions

- ✦ Run multiple operations at once
- ✦ They all succeed or none gets applied

Transactions

POST

```
{  
  "docs": [  
    {"_id": "0", "int": 0, "str": "0"},  
    {"_id": "1", "int": 1, "str": "1"},  
    {"_id": "2", "int": 2, "str": "2"}  
  ]  
}
```

Transactions — Watch out

- ✦ Statement transaction, not data transaction
- ✦ No roundtripping
- ✦ No multi-node transactions
(It's a good thing)

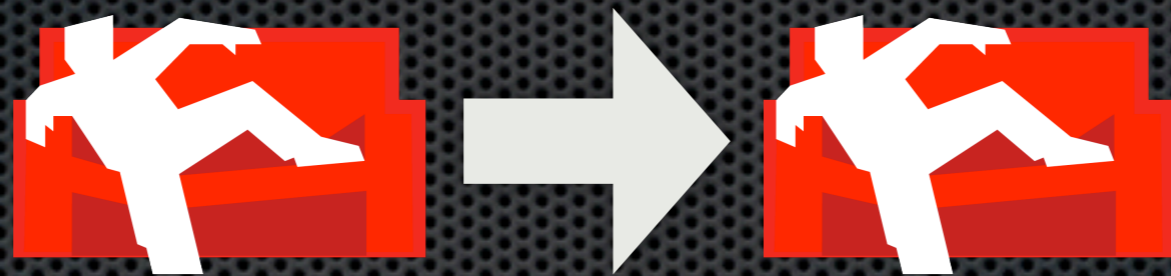
Multi-Node Transactions

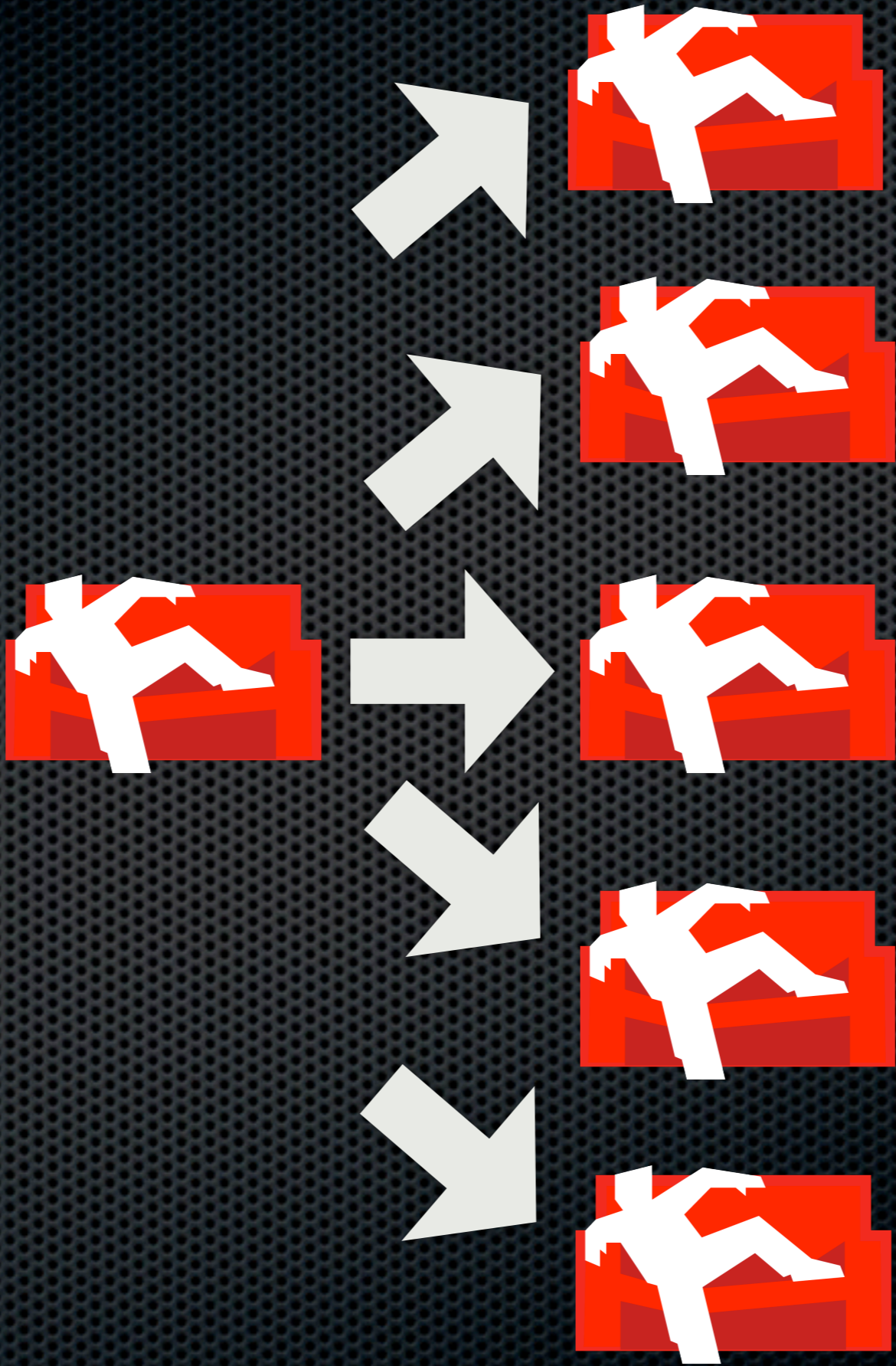
- Why? – Data redundancy
- Use an HTTP proxy
- Nice and easy to build on standard protocols
- Projects in development for consistent hashing & Paxos

Architectures



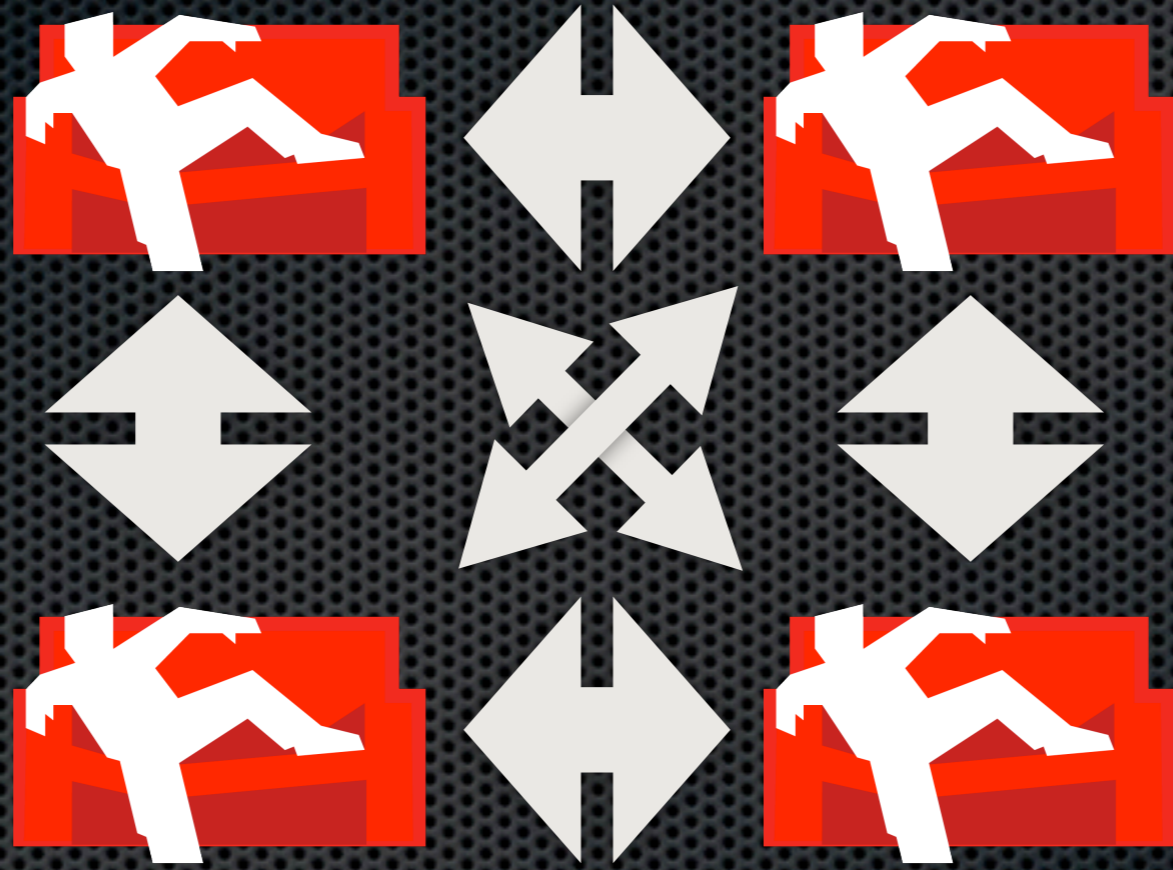






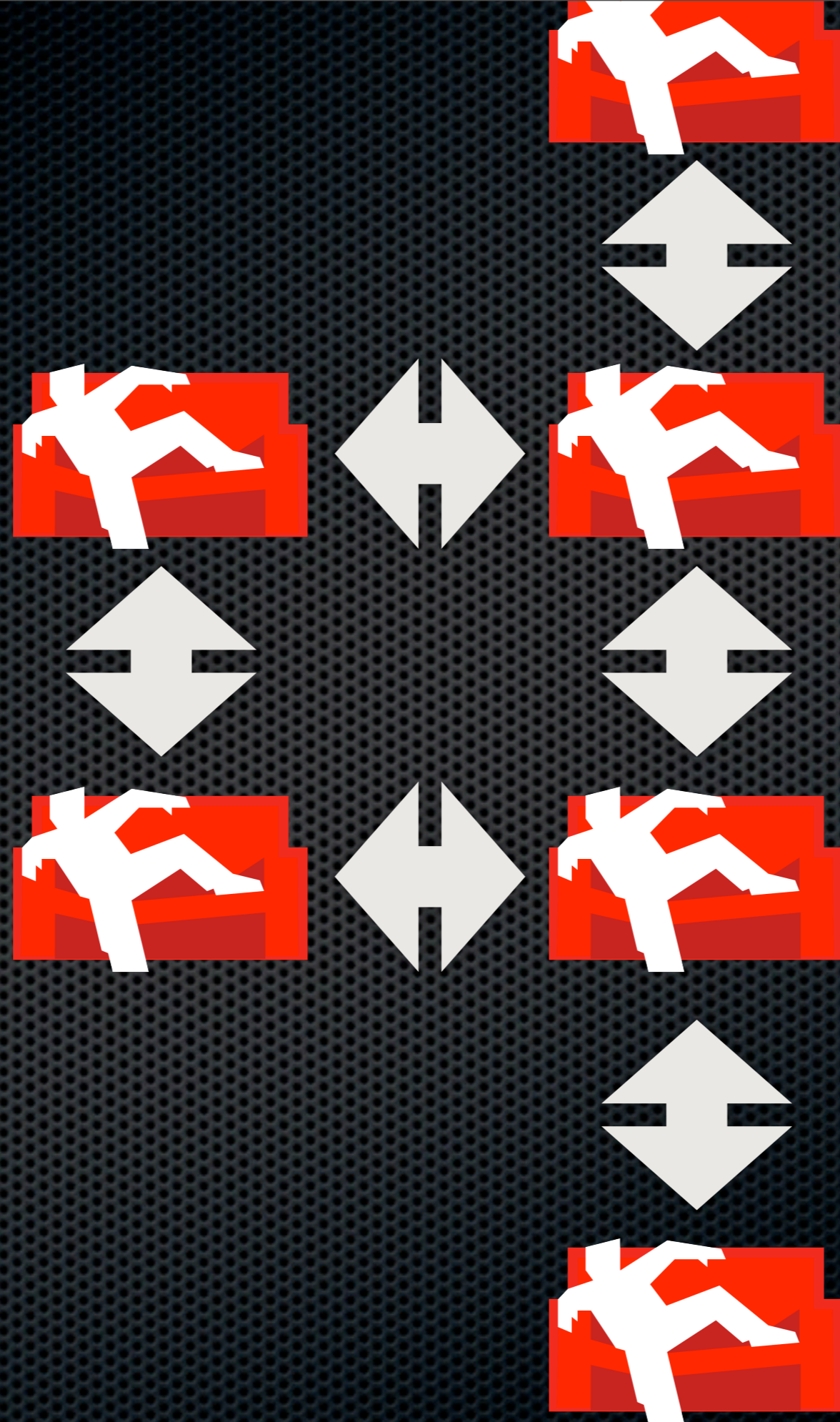


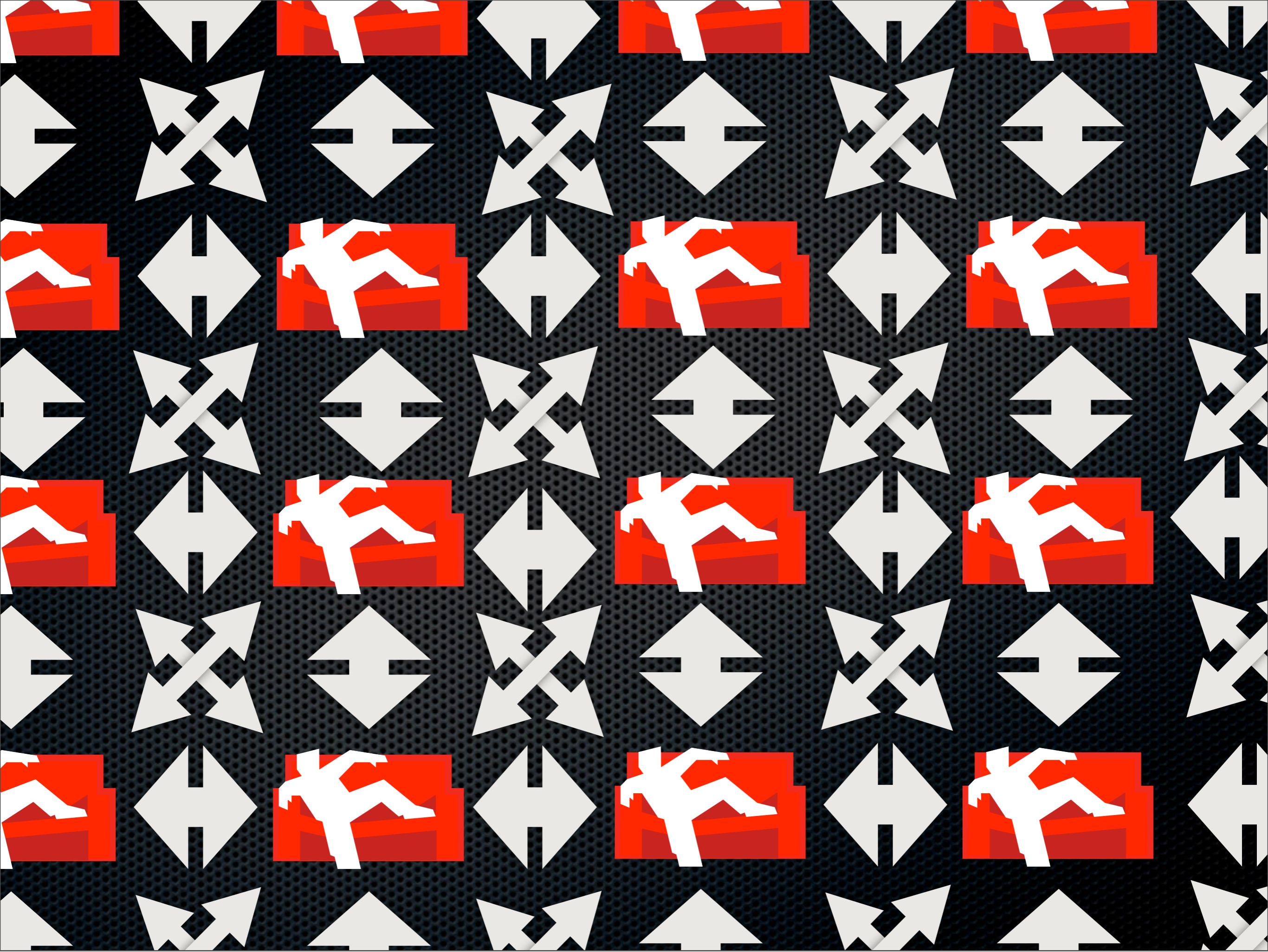






periodic, notifier system





P2P apps

Hot backup?

- POSIX compliant

Hot backup?

```
* $ cp -r /var/lib/couchdb/* \  
  /mnt/backup
```

Commercial Break

The Book

- ✦ O'Reilly
- ✦ <http://books.couchdb.org/relax>
- ✦ Apache 2.0 Licensed
- ✦ Summer 2009

The Book — Can't wait?

- ✦ Help CouchDB
- ✦ Hire me for Consulting, Training, Development
- ✦ jan@apache.org

Resources

- ✦ Twitter: @CouchDB & <http://couchdb.org/>
- ✦ Dress like a Couch:
<http://shop.couchdb.com>
- ✦ <http://damienkatz.net/> & <http://jan.prima.de/>
- ✦ <http://blog.racklabs.com/?p=74>
- ✦ <https://peepcode.com/products/couchdb-with-rails>

Thank You

Really, thanks.

Got it?

Questions

