

mongoDB

@jon_fuller

- practicing apprentice
- not MongoDB expert (sorry)
- not SQL/RDBMS expert (not sorry)
- president <http://indyalt.net>

Huh?

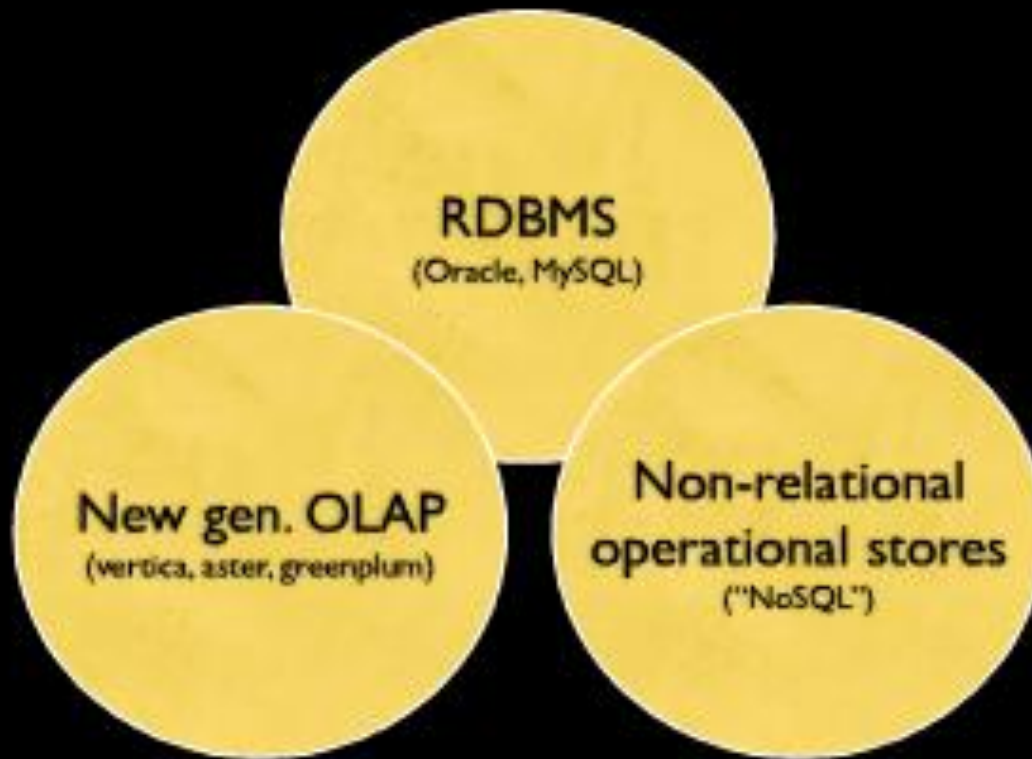
Let's Be Clear



NEO VS AGENT SMITH
THE MATRIX: REVOLUTIONS
FOX STUDIOS AUSTRALIA, DECEMBER 2003

WWW.THEMATRIX.COM

One-Size-Fits-All No Longer



Visual Guide to NoSQL Systems

Availability:
Each client can
always read
and write.

A

Data Models

Relational (comparison)
Key-Value
Column-Oriented/Tabular
Document-Oriented

CA

RDBMSs
(MySQL,
Postgres,
etc)

Aster Data
Greenplum
Vertica

AP

Dynamo
Voldemort
Tokyo Cabinet
KAI

Cassandra
SimpleDB
CouchDB
Riak

Pick Two

C

Consistency:
All clients always
have the same view
of the data.

CP

BigTable
Hypertable
Hbase

MongoDB
Terrastore
Scalaris

Berkeley DB
MemcacheDB
Redis

P

Partition Tolerance:
The system works
well despite physical
network partitions.

Scalability & Performance

• memcached

• key/value
stores

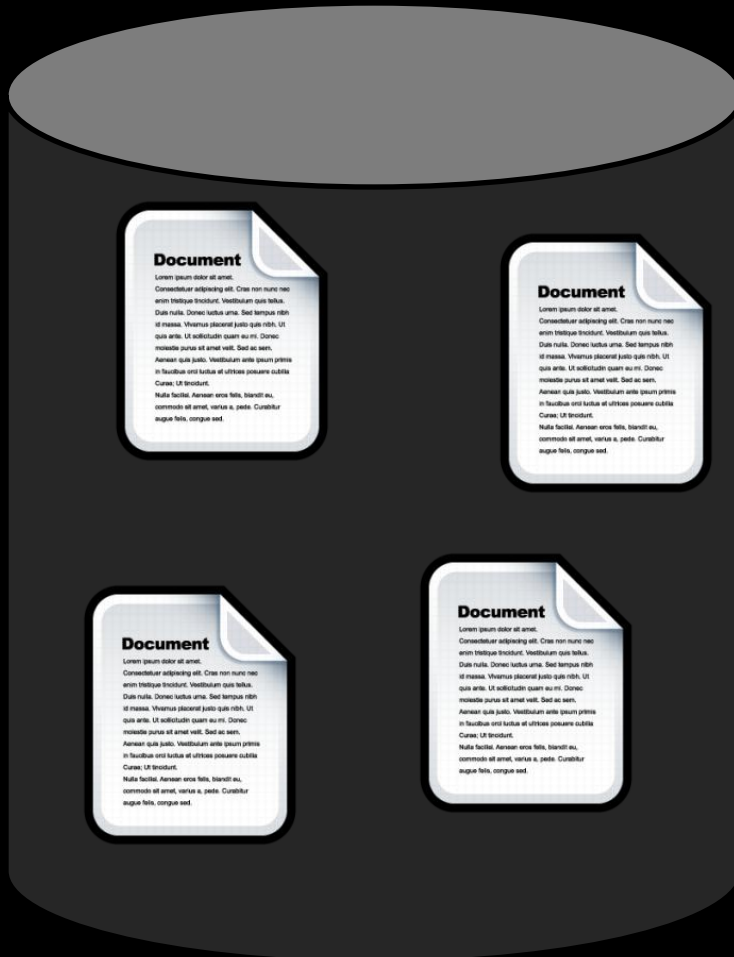
• MongoDB



• RDBMS

Depth of Functionality

Document DB



```
first_name: 'Jon'
```

```
last_name: 'Fuller'
```

address:

```
line_1: '659 Danforth'
```

```
line_2: 'Apt 234'
```

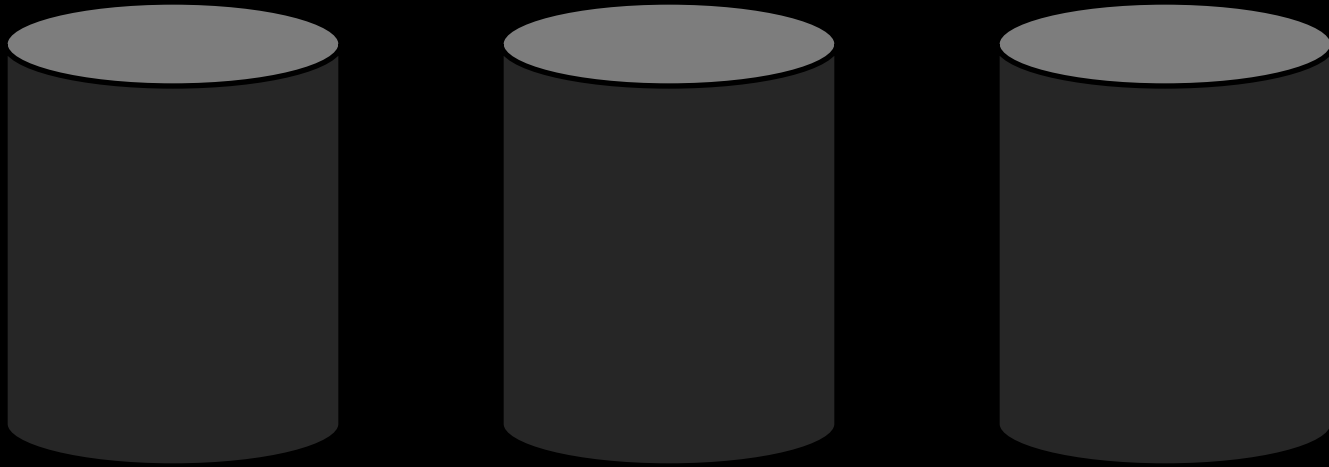
zip: 46032

NoSQL

NoSchema

No Joins!

(sort of)



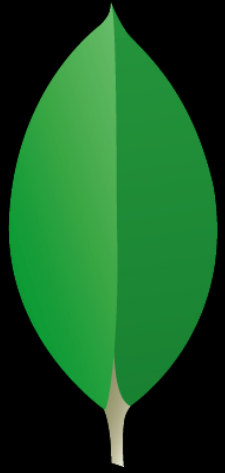
Client

New Models



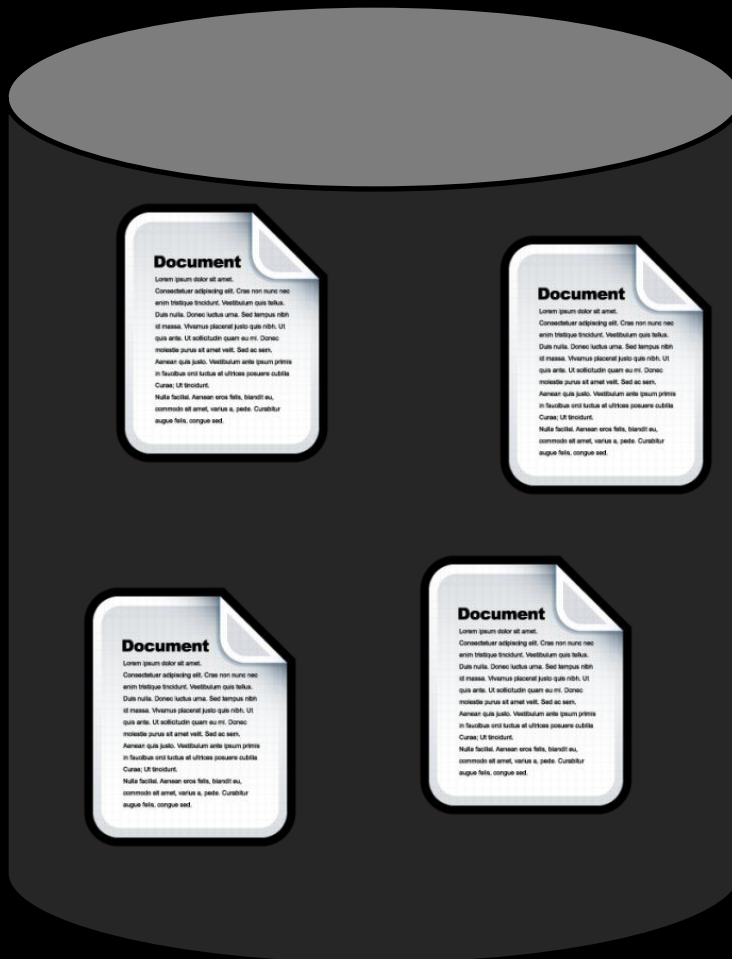


Paperless



mongoDB

JSON

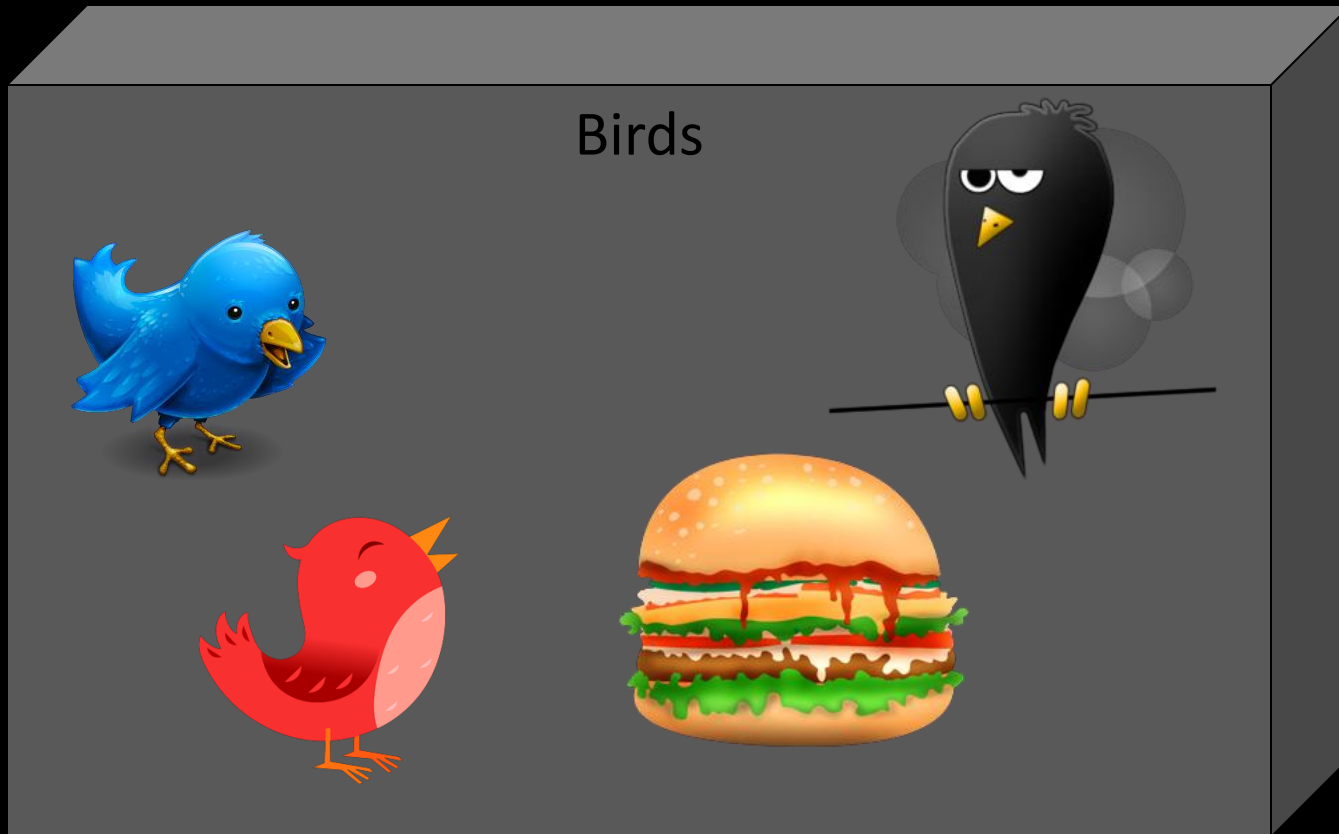


```
{  
  first_name: 'Jon',  
  
  last_name: 'Fuller',  
  
  address: {  
  
    line_1: '659 Danforth',  
  
    line_2: 'Apt 234',  
  
    zip: 46032}  
}
```

Table: RDBMS :: Collection: MongoDB

(sort of)

Collections



Static:Dynamic::RDBMS:MongoDB

(typing)



```
db.papers.save({  
    date:'5/22/2010',  
    tags:['paystub'],  
    views:0,  
    original_filename:'paystub.pdf',  
    content: <BINARY>});
```



```
db.papers.save({  
  date:'5/22/2010',  
  tags:['paystub'],  
  views:0,  
  original_filename:'paystub.pdf',  
  content: <BINARY>});
```

GridFS



```
db.papers.save({  
  date:'5/22/2010',  
  tags:[],  
  views:0,  
  original_filename:'paystub.pdf',  
  grid_fs_name: <SHA1 of file>});
```



Query

```
db.papers.find();
```

Query

```
db.papers.find();
```

```
db.papers.find({date:'5/22/2010'});
```

Query

```
db.papers.find();
```

```
db.papers.find({date:'5/22/2010'});
```

```
db.papers.find({tags:'paystub'});
```

Index

```
db.papers.ensureIndex({tags:1});
```

```
db.papers.find({tags:'paystub'});
```

Query

```
db.papers.find(  
  {tags: {$all:['paystub', 'jon']}});
```

Query

```
db.papers.find(  
  {tags: {$all:['paystub', 'jon']}});
```

```
db.papers.find(  
  {tags: {$in:['paystub', 'jon']}});
```




Update

```
db.papers.update(  
  {_id:<ID>, {$inc: {views:1}}});
```

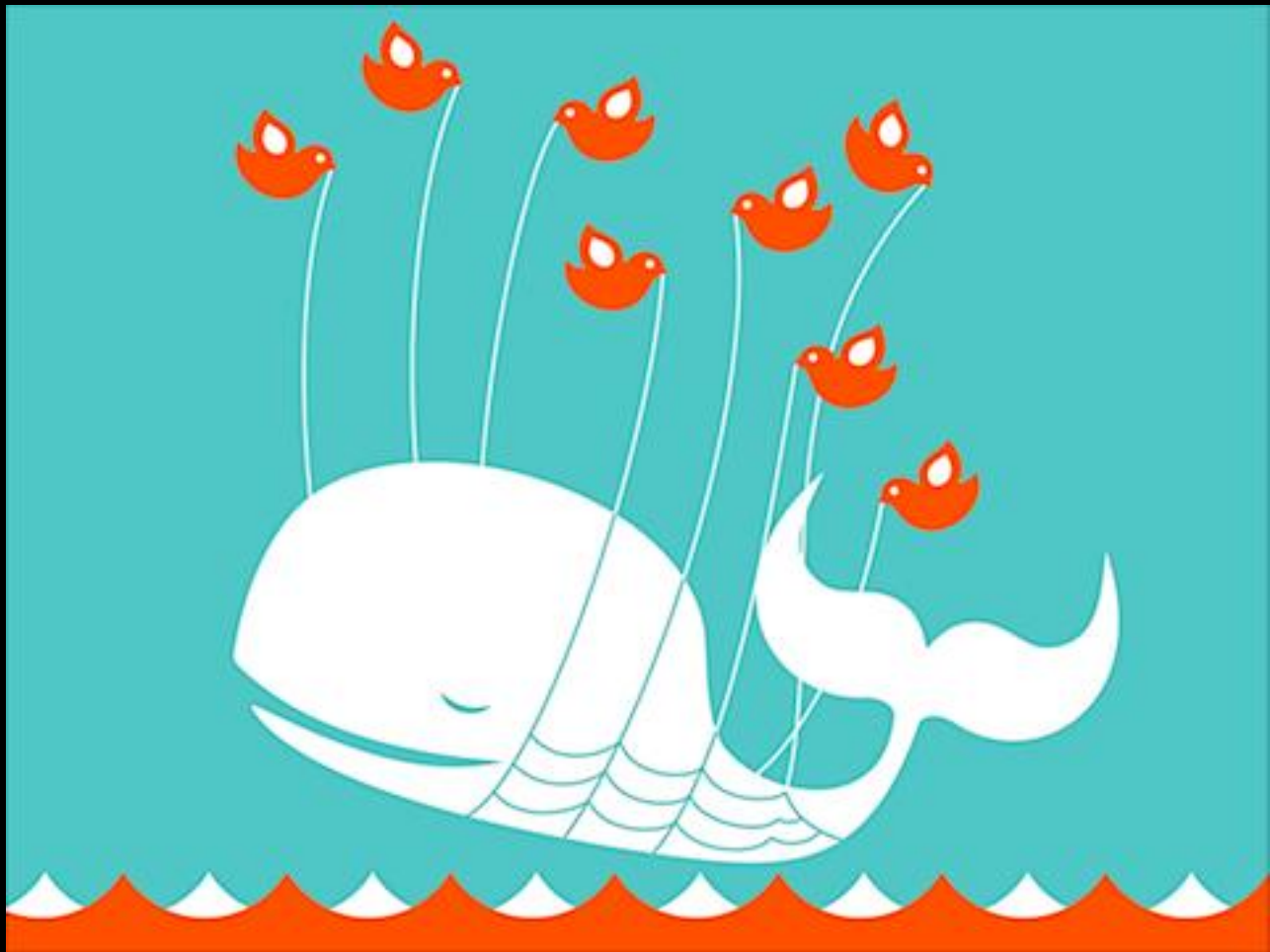
Update

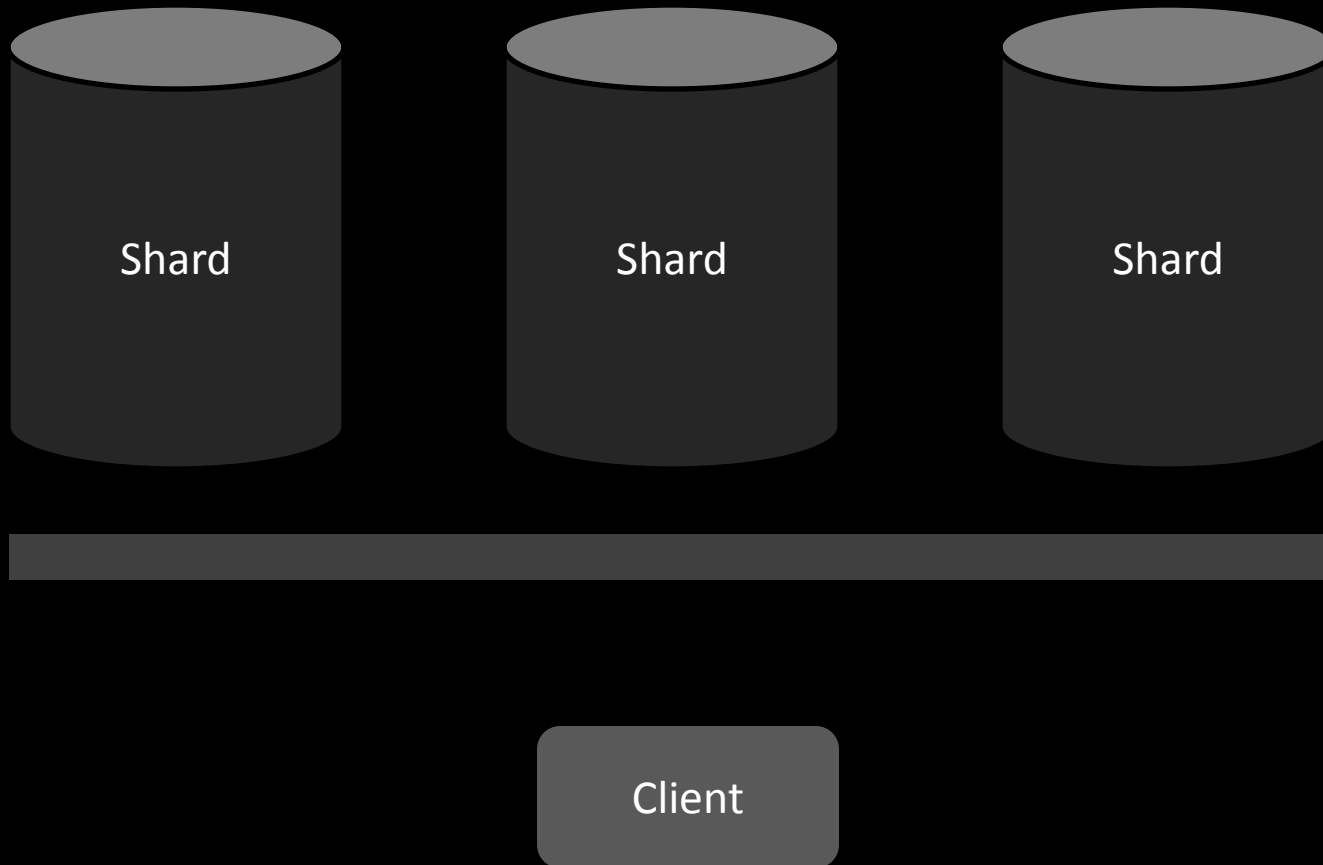
```
db.papers.update(  
  {_id:<ID>, {$push: {tags:'jon'}}});
```

Update

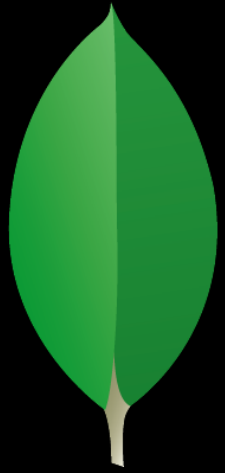
```
db.papers.update(  
  {_id:<ID>, {$pull: {tags:'paystub'}}});
```





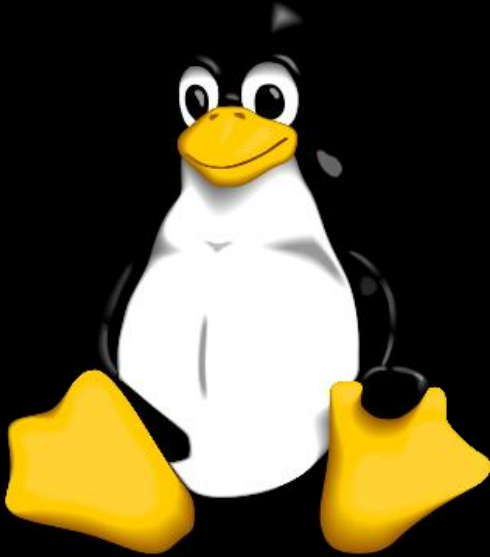






mongoDB

Available Platforms



Map/Reduce

```
function(){  
  this.tags.forEach(  
    function(tag){  
      emit(tag, {count: 1}); })  
}
```

```
function(key, values){  
  var total = 0;  
  for ( var i=0; i<values.length; i++ ) {  
    total += values[i].count;  
  }  
  return {count: total};  
}
```

C#

- NoRM
 - <http://github.com/atheken/NoRM/>
- mongodb-csharp
 - <http://github.com/samus/mongodb-csharp>

mongodb.org

try.mongodb.org

github.com/jonfuller/paperless

github.com/jonfuller/presentations

Comments/Questions/Feedback

(please!!)