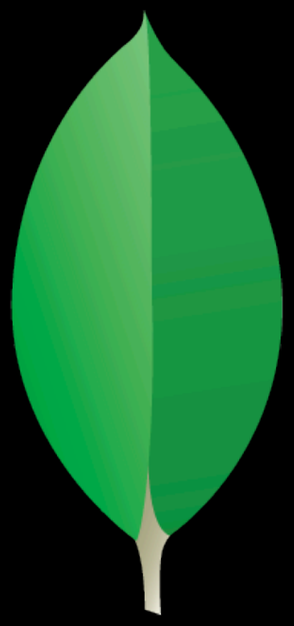
Inside MongoDB: the Internals of an Open-Source Database



mongoDB

Mike Dirolf - @mdirolf - 10gen, Inc.

http://www.mongodb.org

Inside mongoDB

http://www.flickr.com/photos/tmh9/677919415/

a word of warning

this talk might be a bit “hard”,
but MongoDB is really easy:
http://try.mongodb.org

```
db.test.insert({hello: “world”})
```

_id

if not specified drivers will add default:

```
ObjectId("4bface1a2231316e04f3c434")
```

timestamp

machine id

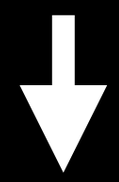
process id

counter

http://www.mongodb.org/display/DOCS/Object+IDs

BSON Encoding

```
{_id: ObjectId(XXXXXXXXXXXX),
 hello: “world”}
```



```
\x27\x00\x00\x00\x07    _    i    d\x00    X
    X    X    X    X    X    X    X    X    X
    X\x02    h    e    l    l    o\x00\x06\x00
\x00\x00    w    o    r    l    d\x00\x00
```

http://bsonspec.org

Insert Message (TCP/IP)

message length	request id	response id	op code (insert)
\x68\x00\x00\x00	\xXX\xXX\xXX\xXX	\x00\x00\x00\x00	\xd2\x07\x00\x00

reserved	collection name	document(s)
\x00\x00\x00\x00	f o o . t e s t \x00	BSON Data

http://www.mongodb.org/display/DOCS/Mongo+Wire+Protocol

Data File Allocation

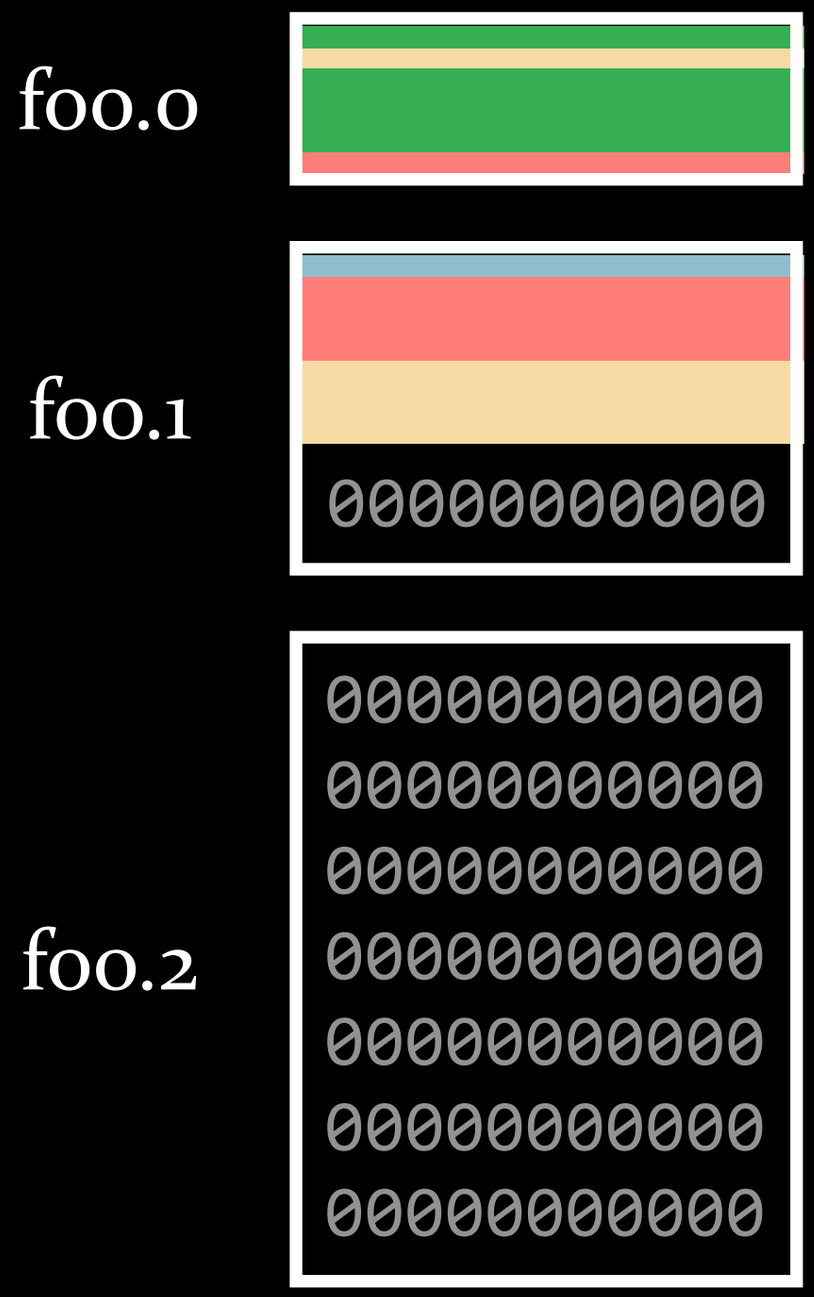
```
$ ls -sk /data/db/
   16384 foo.ns
   65536 foo.0
  131072 foo.1
   16384 bar.ns
     ...
```

double in size (up to 2 gigs)

allocated per database

Memory Management

Extent Allocation



allocated per *namespace*:

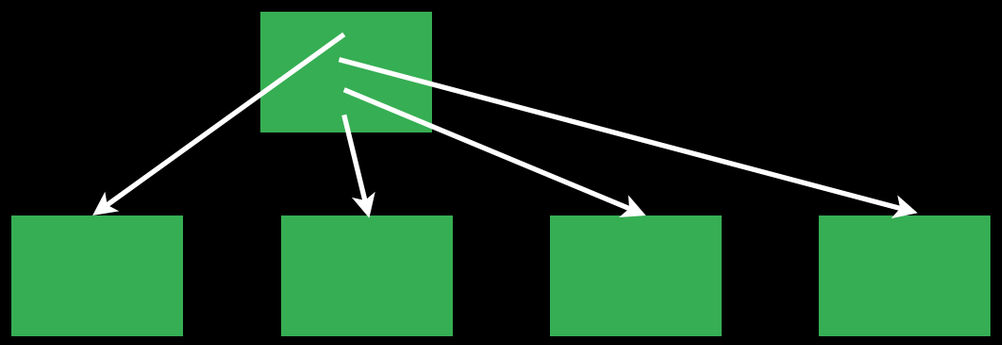
- foo.test
- foo.bar
- foo.baz
- foo.$freelist
- 0000 preallocated space

ns details stored in foo.ns

Record Allocation

Indexing

B-Tree indexes, stored in own namespaces



```
> db.system.namespaces.find()
{ "name" : "foo.system.indexes" }
{ "name" : "foo.test" }
{ "name" : "foo.test.$_id_" }
```

http://www.mongodb.org/display/DOCS/Indexes

```
db.test.find({hello: “world”})
```

Query Language

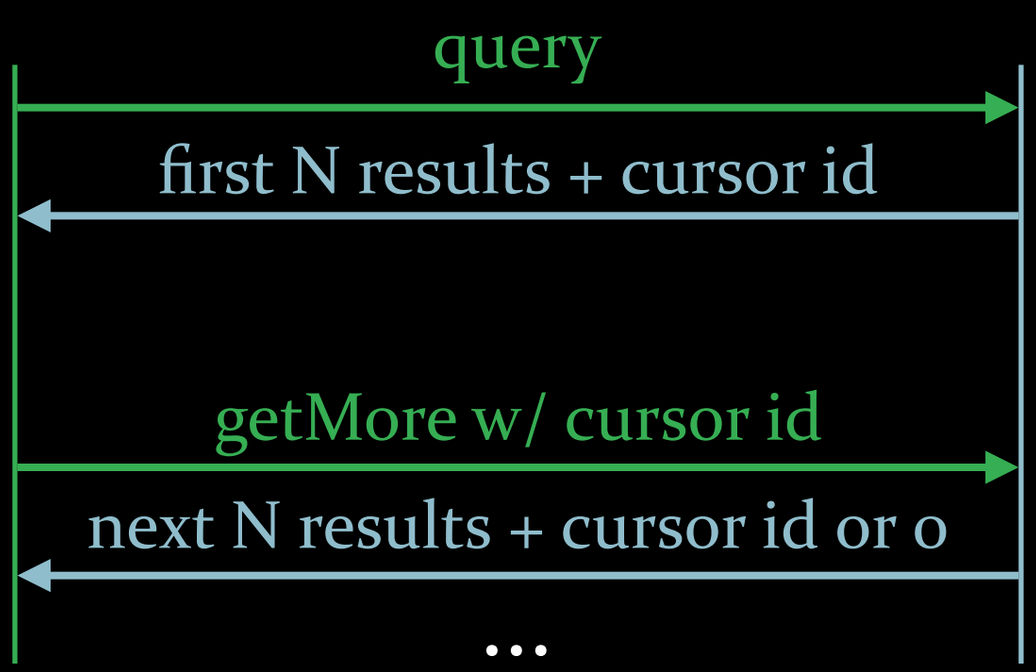
"query by example" plus $ modifiers:

```
{first_name: "Mike",
 age: {$gte: 20, $lt: 40}}
```

http://www.mongodb.org/display/DOCS/Advanced+Queries

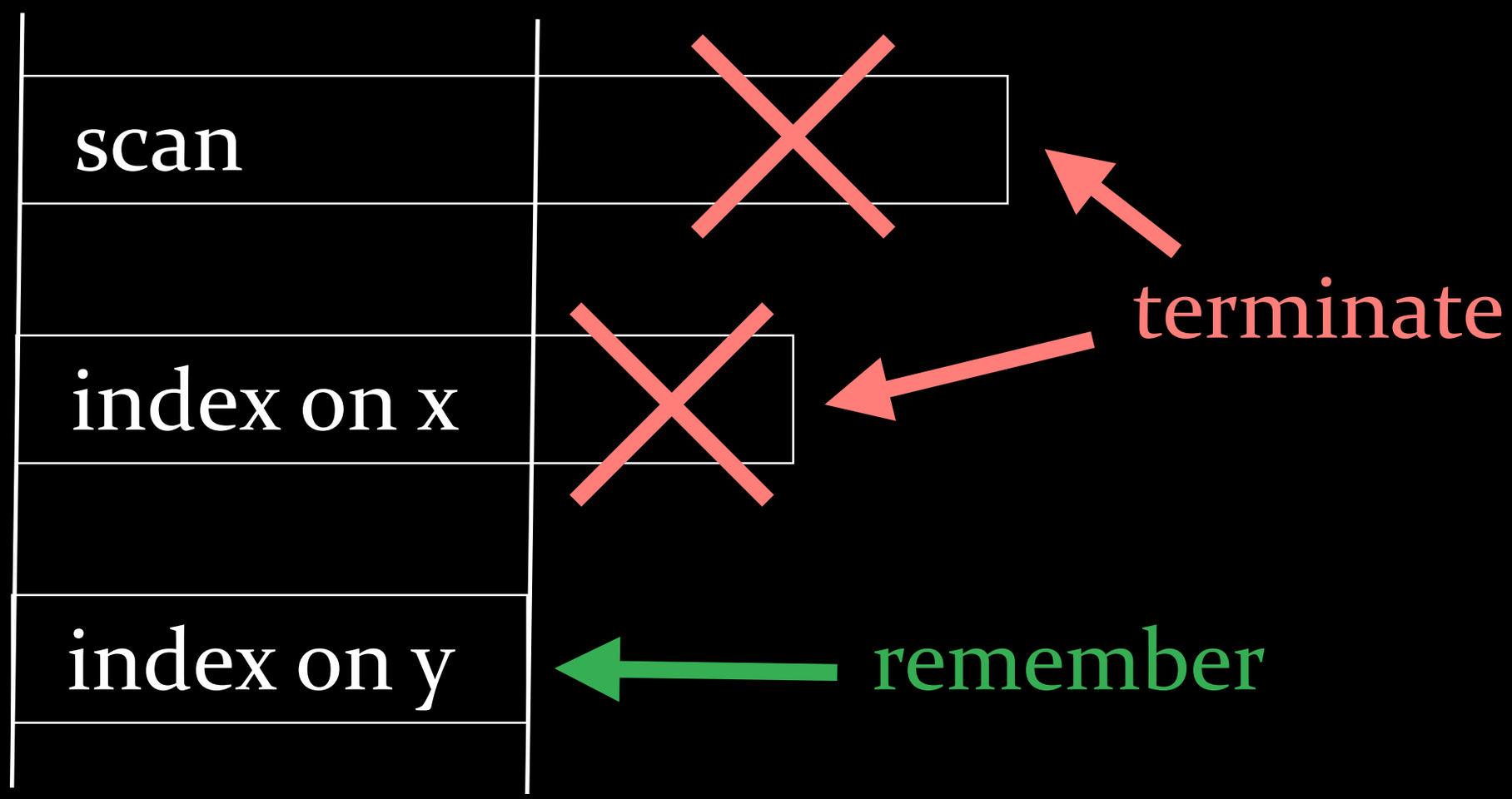
Cursors

```
> var c = db.test.find({x: 20}).skip(20).limit(10)
> c.next()
> c.next()
    ...
```



Query Optimizer

```
find({x: 10, y: “foo”})
```



```
db.foo.drop()
```

Commands

drop, count, copydb,
findAndModify, serverStatus, ...

http://www.mongodb.org/display/DOCS/Commands

```
db.foo.drop();
```

=

```
db.foo.runCommand({drop: "foo"});
```

=

```
db.$cmd.findOne({drop: "foo"});
```

=

```
db.$cmd.find({drop: "foo"}).limit(-1);
```

Capped Collections

preallocated
auto LRI age-out
no default _id index
always in insertion order

http://www.mongodb.org/display/DOCS/Capped+Collections

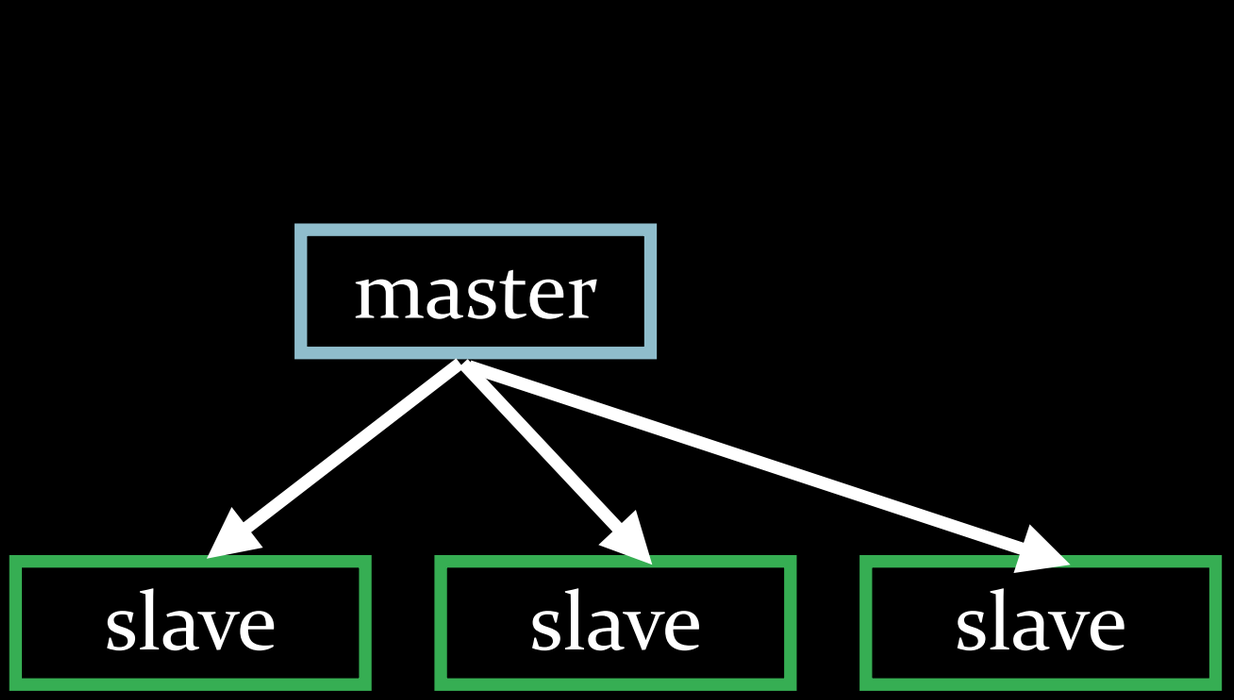
Replication Oplog

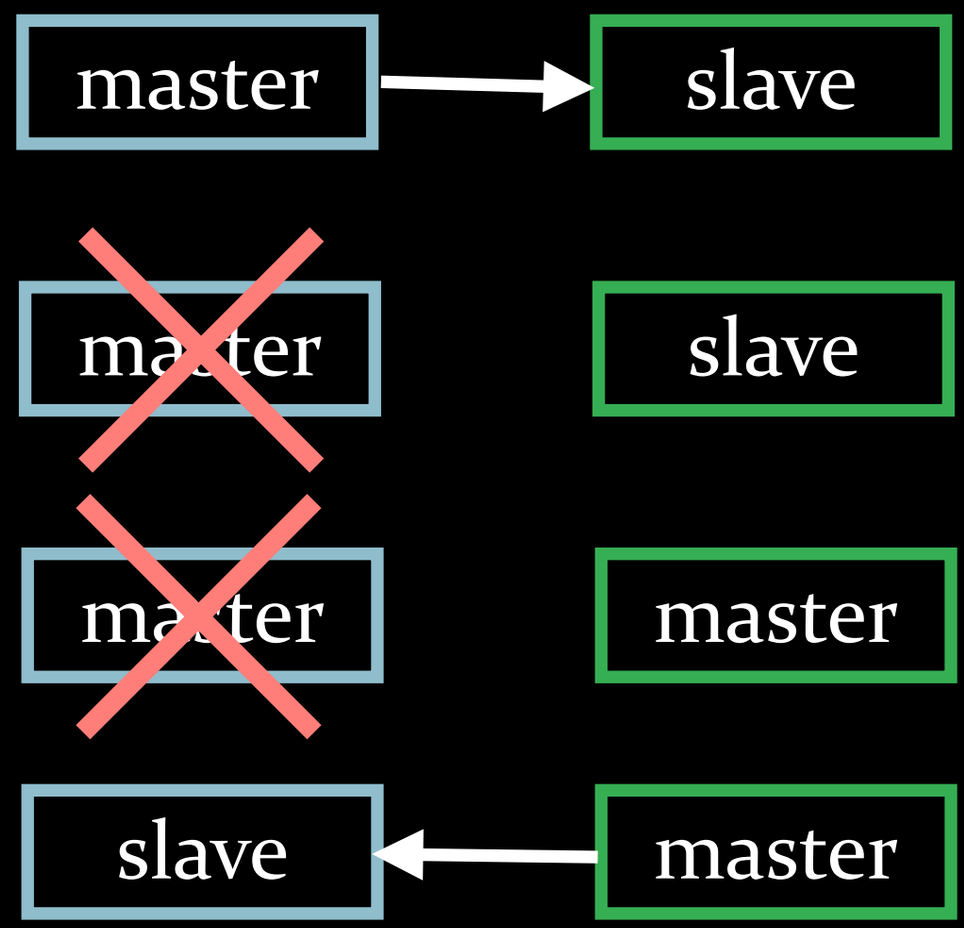
```
> use foo
switched to db foo
> db.test.insert({x: 1, url:
"http://dirolf.com"});

> db.test.update({url: "http://
dirolf.com"}, {$inc: {x: 1}});
```

```
> use local
switched to db local
> db.oplog.$main.find()
{ts: ..., op: "n", ns: "", o: {}}
{ts: ..., op: "n", ns: "", o: {}}
{ts: ..., op: "i", ns: "foo.test",
 o: {_id: ObjectId("..."),
     x: 1,
     url: "http://dirolf.com"}}
{ts: ..., op: "n", ns: "", o: {}}
{ts: ..., op: "u", ns: "foo.test",
 o2: {_id: ObjectId("...")},
 o: {$set: {x: 2}}}
```

http://www.mongodb.org/display/DOCS/Replication

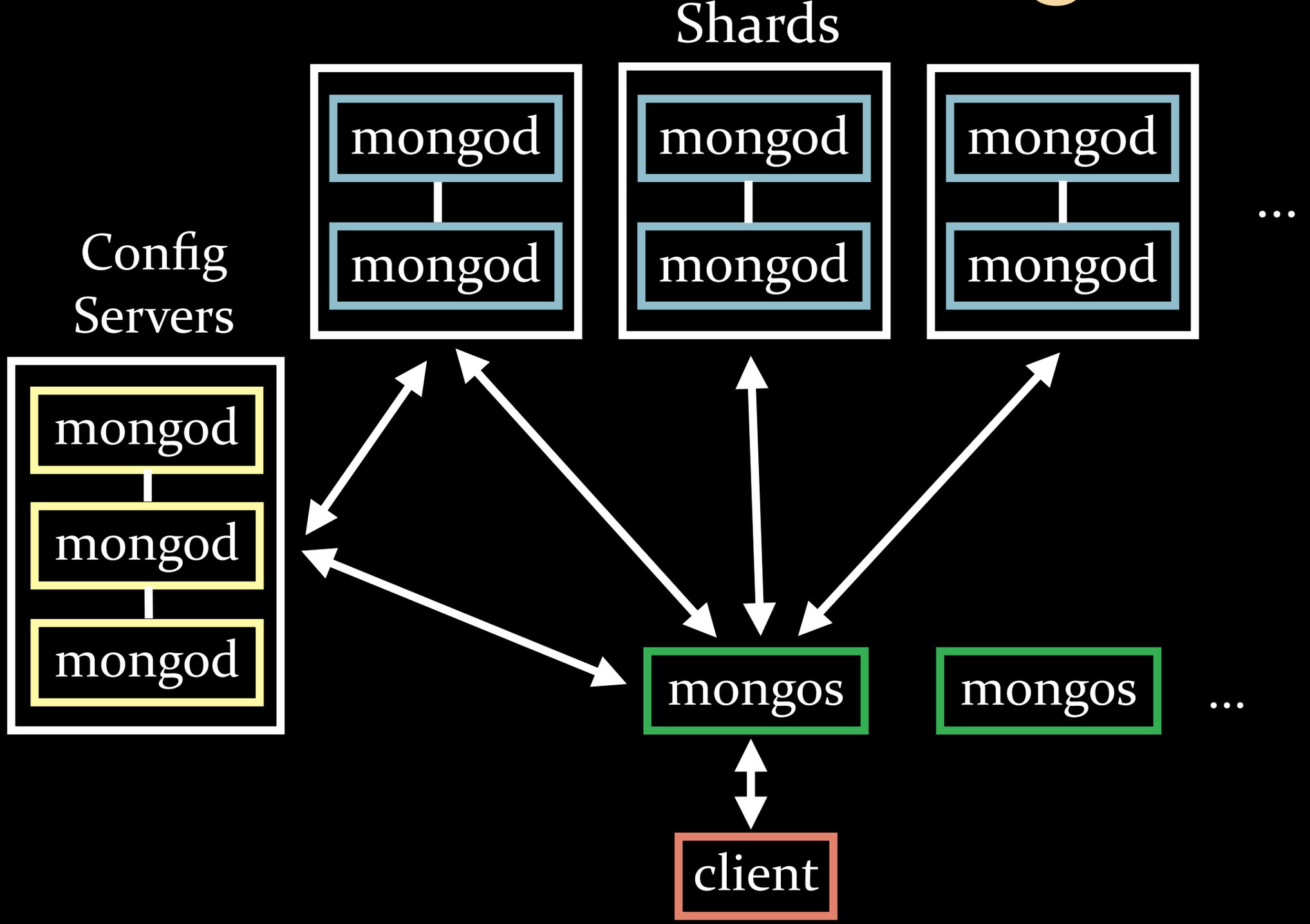
Replication Topology





http://www.mongodb.org/display/DOCS/Replication

Auto-Sharding



http://www.mongodb.org/display/DOCS/Sharding

Geohashing

(20, 10)

(0001 0100, 0000 1010)

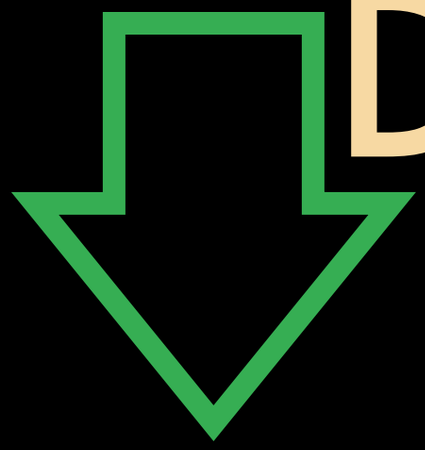
0000 0010 0110 0100

maps close coordinates (21, 9) to close hashes:

0000 0010 0110 0011

tricky part happens at bit-flips (127 vs 128)

http://www.mongodb.org/display/DOCS/Geospatial+Indexing



Download MongoDB

http://www.mongodb.org

these slides are available at http://dirolf.com