Juggling Chainsaws

David Golden
Senior Engineer, MongoDB

YAPC::NA 2015

I lead the MongoDB Perl driver project

Do I need to explain what MongoDB is?

NoSQL

Not NoSQL

Not NoSQL

Documents

Auto-sharding

Native replication

Pluggable storage engines

Aggregation framework queries

Comprehensive secondary indexes

Multiple geospatial indexes and text search

ACID guarantees available at the document level

Fault tolerance and disaster recovery through automated failover

Enterprise-version has extensive capabilities for authentication, authorization, auditing and encryption

Ops Manager and MMS provide continuous incremental backup, point-in-time recovery of replica sets and consistent snapshots of sharded clusters.

Docu

A
C
Mu
ACI
Fault t
Enterprise
Ops Manager an
ed failover
ization, auditing and encryption
overy of replica sets and consistent snapshots of sharded clusters.

"London Juggler" © Leon Benjamin https://flic.kr/p/cmTzL (CC BY)

"London Juggler" © Leon Benjamin https://flic.kr/p/cmTzL (CC BY)

Perl and MongoDB are surprisingly similar

ster · e · o · type

noun

a **widely held** but **fixed** and **oversimplified** image or idea of a particular person or thing

Trolls love stereotypes

Trolls

Perl and MongoDB

Trolls

Perl and MongoDB

Let's play

Guess the meme!

"____ is dead"

"Dead camel" © Nick Brooks https://flic.kr/p/ivZdP (CC BY-NC-SA)

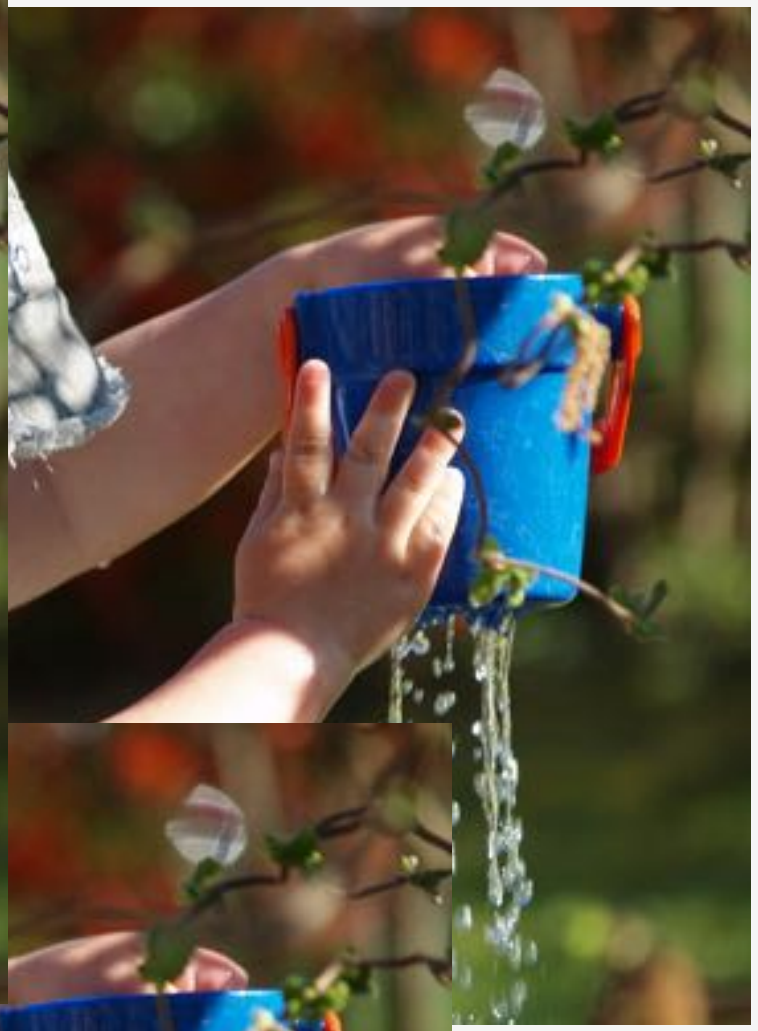
"___ is web scale"

"MongoDB is web Scale" © gar1t http://www.mongodb-is-web-scale.com/

"___ is line noise"

"Camel JAPH" © Erudil http://www.cpan.org/misc/japh

"___ will lose your data"

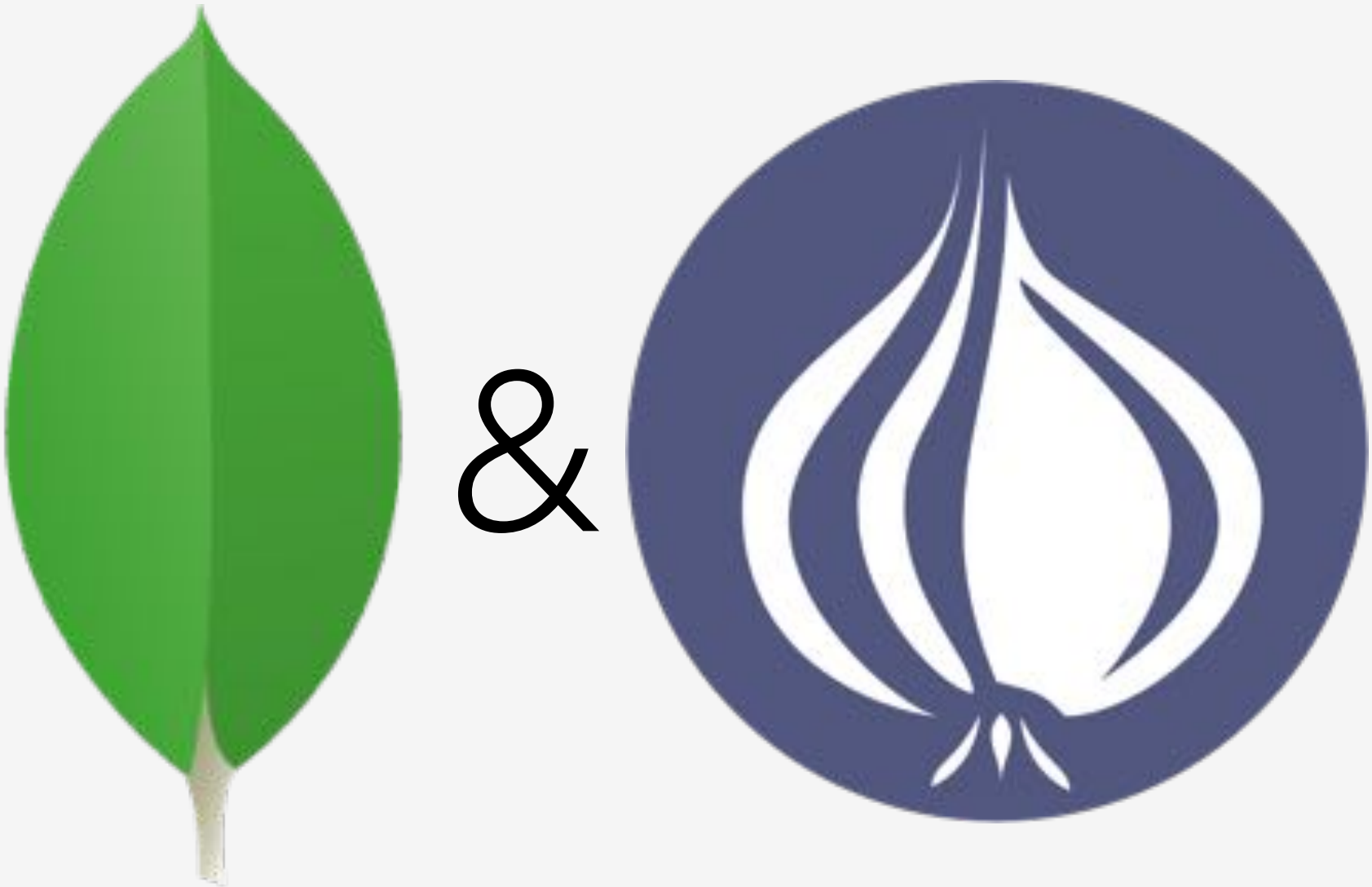


"___ is read only"

"fresh ramen noodle" © Kropsoq http://goo.gl/mu2fIQ (CC BY-SA)

"__ is unsafe by default"

TRICK QUESTION!



✓ Grief from haters

✓ Similar data model

"A hash brown patty" © Mike.lifeguard http://goo.gl/2we5ye (CC BY-SA)

"buy it in bottles" © Pete https://flic.kr/p/qbvaRb (CC BY)

perldsc

Data Structures Cookbok

HoH, AoH, HoA, AoA. . .

NOW I HAVE

A MACHINE GUN

NOW I HAVE
A MACHINE GUN

Ho-Ho-HoA

Perl programmers use dynamic data structures all the time!!!

MongoDB

Data Structures

```
{
  name      => 'The Tick',
  battlecry => 'Spoon!',
  birthday  => '1973-07-16',
  relations => {
    friends => [
      'Arthur',
      'Die Fledermaus',
      'American Maid',
    ],
    foes    => [
      'The Terror',
      'Thrakkorzog',
    ],
}
```

Sort heroes by
first friend's name?

```
{
  name      => 'The Tick',
  battlecry => 'Spoon!',
  birthday  => '1973-07-16',
  relations => {
    friends => [
      'Arthur',
      'Die Fledermaus',
      'American Maid',
    ],
    foes    => [
      'The Terror',
      'Thrakkorzog',
    ],
}
```

```
# Perl using Schwartzian Transform
my @sorted =
  map  { $_->[0] }
  sort { $a->[1] cmp $b->[1] }
  map  { [$_, $_->{relations}{friends}[0] };
```

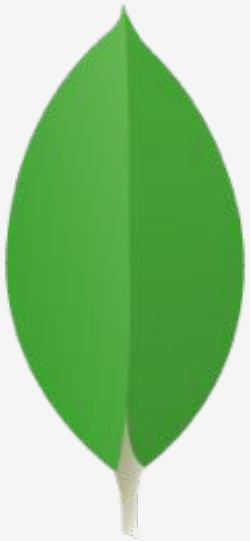
```
# MongoDB using query with sort clause
my @sorted = $heroes->find(
  {}, { sort => [ 'relations.friends.0' => 1 ] }
)->all;
```

```perl
# Perl using Schwartzian Transform
my @sorted =
  map  { $_->[0] }
  sort { $a->[1] cmp $b->[1] }
  map  { [$_, $_->{relations}{friends}[0] };
```

```perl
# MongoDB using query with sort clause
my @sorted = $heroes->find(
  {}, { sort => [ 'relations.friends.0' => 1 ] }
)->all;
```

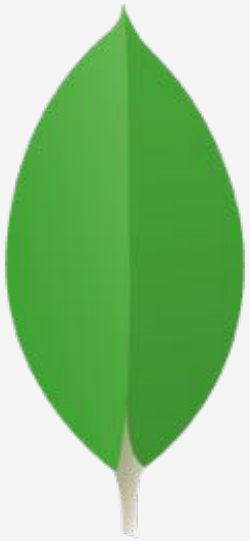
✓ Missing things others take for granted

No function signatures



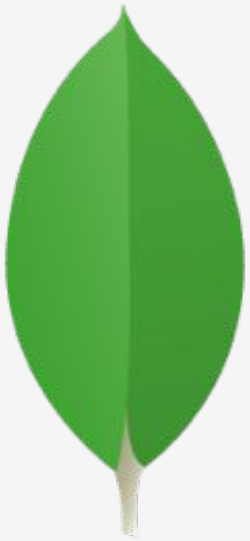
No read committed

No concurrency



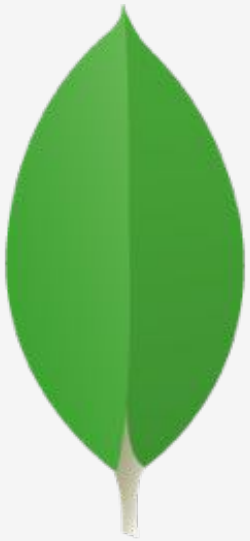
No transactions

No (real) OO



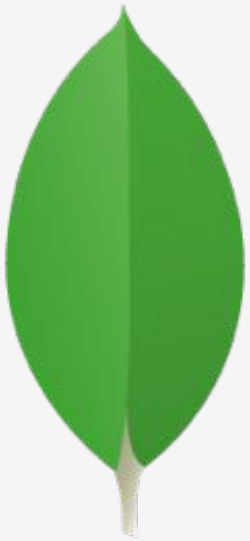
No joins

No booleans



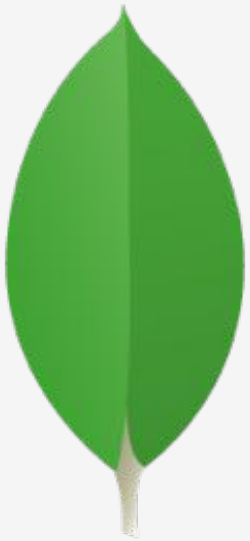
No stored procedures

✓ Experienced users work around flaws



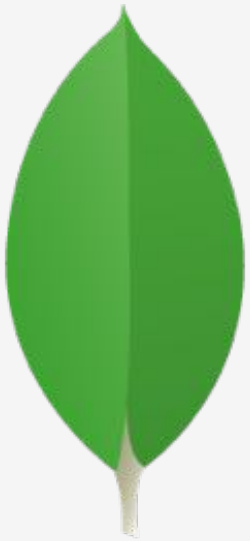
Non-idempotent writes

Strings for errors



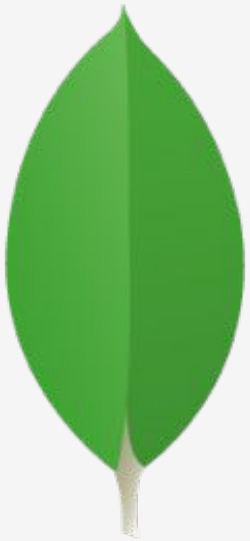
Consensus read

Version numbers



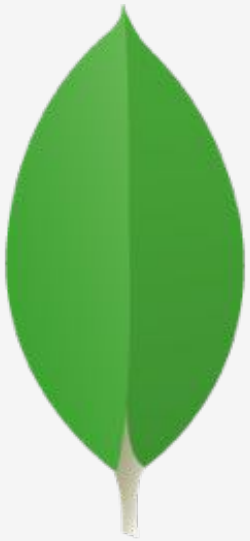
Query "language"

Ambiguous scalar type



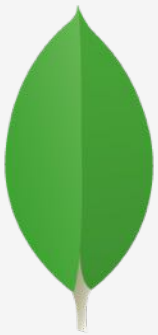
16MB doc limit

IO layers



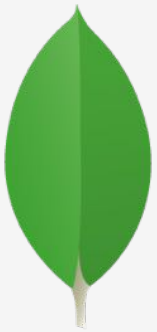
Rollback on failover

✓ Vibrant, enthusiatic communities



MUGs

Perl Mongers



Conferences
Mailing Lists
IRC

✓ Similar philosophies

Manipulexity
Whipuptitude

Manipulexity

Manipulation of complex things

— *Larry Wall*

Whipuptitude

Aptitude for whipping things up

— *Larry Wall*

Tools for getting the job done

Swiss-Army knife?

Swiss-Army chainsaw!!!

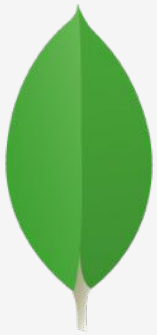
Henry Spencer described the Perl scripting language as a "**Swiss-Army chainsaw**", intending to convey his evaluation of the language as **exceedingly powerful but ugly and noisy and prone to belch noxious fumes**.

— *The Jargon File*

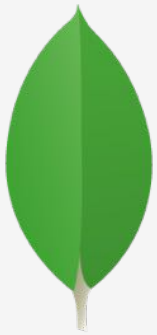
This had two results: (1) Perl fans adopted the epithet as a badge of pride, and (2) it entered more general usage to describe software that is **highly versatile but distressingly inelegant**.

— *The Jargon File*

Highly versatile but distressingly inelegant?



YES



Great at a few things

Good enough
for many things

And if one is good,
wouldn't two be better?

+

"Camel warning sign UAE" © Katzenmeier http://commons.wikimedia.org/wiki/File:Camel_warning_sign_UAE.jpg (PD)

CPAN Meta file indexing

META.json

```
{
   "abstract" : "File path utility",
   "author" : [
      "David Golden <dagolden@cpan.org>"
   ],
  "name" : "Path-Tiny",
  "prereqs" : {
      "runtime" : {
         "requires" : {
            "Carp" : "0",
            "Cwd" : "0",
            "Digest" : "1.03",
            "Digest::SHA" : "5.45",
            ...
         },
      },
      ...
   },
   "provides" : {
      "Path::Tiny" : {
         "file" : "lib/Path/Tiny.pm",
         "version" : "0.068"
      },
      ...
   },
   "release_status" : "stable",
   ...
   "x_contributors" : [
      "Alex Efros <powerman@powerman.name>",
      ...
   ],
   ...
}
```

Thought experiment: Rebuild CPAN?

Rebuilding CPAN?

- Tarball storage
- Tarball processing
- Worker logs
- Index lookup
- Text search

Storage: GridFS

```
# store a file
open $fh, "<", $tarball_path;
$gridfs = $db->get_gridfs("tarballs");
$id     = $gridfs->put($fh, $metadata);


# get a file
$bytes = $gridfs->get($id)->slurp;
```

Processing: MongoDBx::Queue

```
# in supervisor
my $queue = MongoDBx::Queue->new(...);
$queue->add_task({ tarball_id => $id });

# in worker
while ( my $task = $queue->reserve_task )
{
    # ... process tarball ...
    $queue->remove_task( $task );
}
```

Logging: capped collection

```
$db->run_command([
    create => "pause_log",
    capped => true,
    max => 1000,
]);


$log->find(
  {}, { cursorType => 'tailable' }
);
```

Index lookup: query language

```
$packages->find(
  {
    mod => "Foo",
    ver => {
      '$gte' => $v_min,
      '$lte' => $v_max,
    },
  }
);
```

Index lookup: query

Index lookup (take 2)

```
$meta->find(
  {
    '_authorized.Foo' => true
    'release_status'  => 'stable',
  }
);

# sort through versions client side
# with version.pm
```

Text search: text indexes

```
$meta->indexes->create_one([
  abstract: "text",
  keywords: "text",
]);

$meta->find( {
  '$text' => {
    '$search' => "dist zilla plugin"
  }
});
```

Some other handy tools

- Geographic indexing and search
- TTL indexes (auto-expiration)

"The Long Road" © Corey Leopold https://flic.kr/p/6AX7aq (CC BY)

Coming soon:
MongoDB v1.0.0 Beta 1

Coming ...

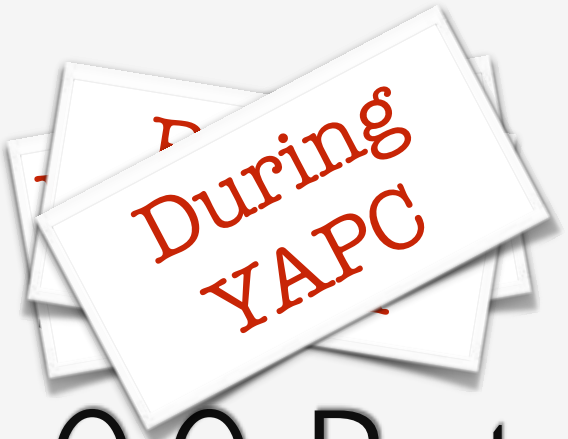
Right after this talk!

MongoDB v1.0.0 Beta 1

Coming 

MongoDB v1.0.0 Beta 1

Coming



MongoDB v1.0.0 Beta 1

Coming

MongoDB v1.0.0 Beta 1

- Better server discovery, monitoring, failover
- Immutable config (and no more globals!)
- New CRUD and Index APIs
- Consistent error handling
- Progress towards a pure-Perl option
- BSON round-trippability

"Dub Dub Dub" © Owen B https://flic.kr/p/8jGEnz (CC BY-NC)

What I ask of you...

Try it out!

Or, try it out again!

Get involved!

- Download MongoDB
- Try the beta driver
- Find a MUG near you
- Follow my blog (dagolden.com)
- Follow me on twitter (@xdg)
- ++ the driver on MetaCPAN
- Ask questions

Questions?

david@mongodb.com