Building your first app: an introduction to MongoDB

Dr Annalina Caputo

What is MongoDB?

MongoDB is a _____ database

- Document
- Open source
- High performance
- Horizontally scalable
- Full featured

Document Database

- Not for .PDF & .DOC files
- A document is essentially an associative array
- Document = JSON object
- Document = PHP Array
- Document = Python Dict
- Document = Ruby Hash
- etc

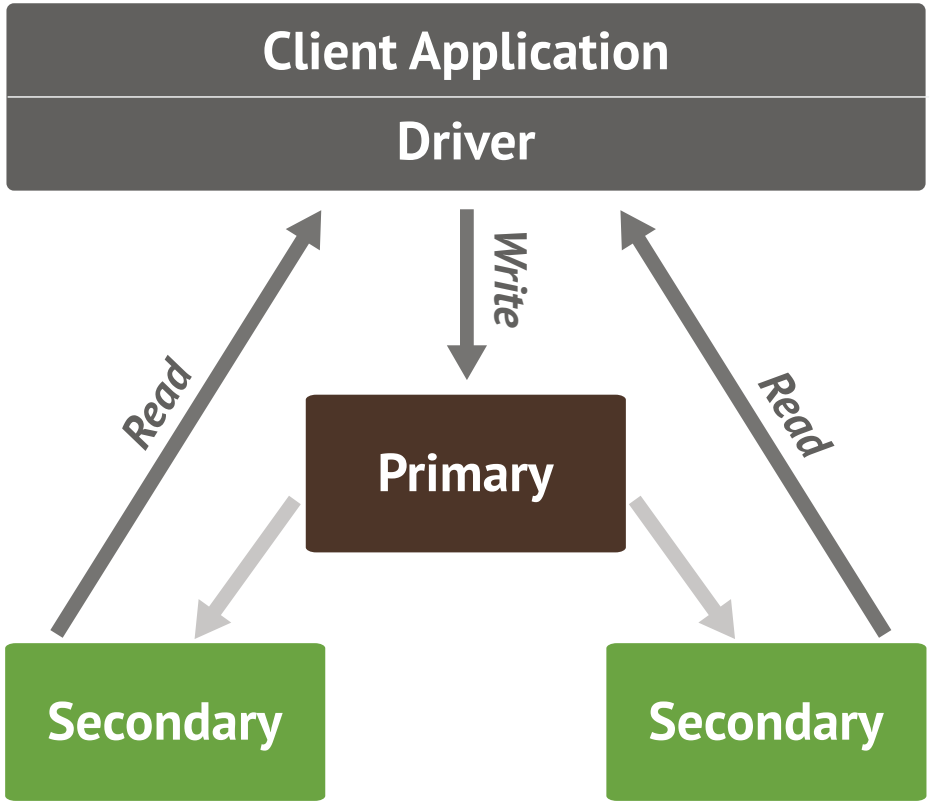
Open Source

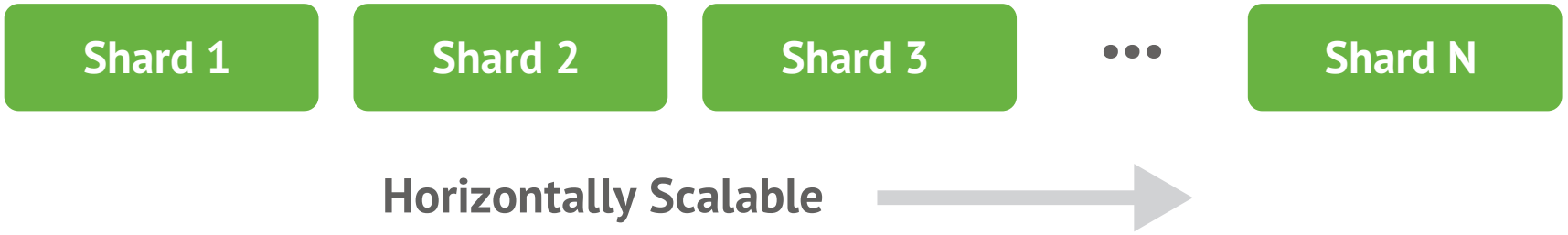
- MongoDB is an open source project
- On GitHub
- Licensed under the AGPL
- Started & sponsored by MongoDB Inc (formerly 10gen)
- Commercial licenses available
- Contributions welcome

High Performance

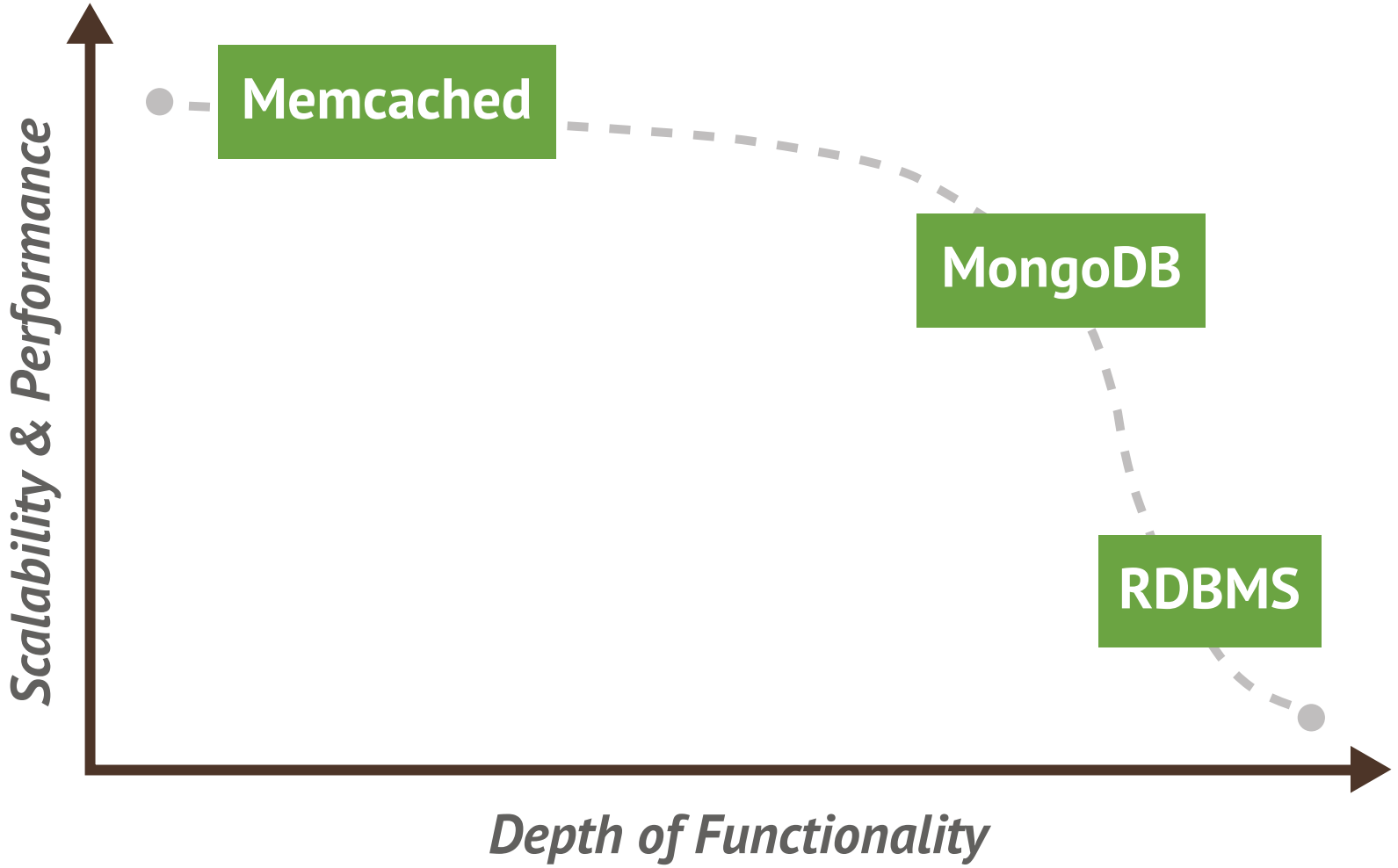
- Written in C++
- Extensive use of memory-mapped files
  i.e. read-through write-through memory caching
- Runs nearly everywhere
- Data serialized as BSON (fast parsing)
- Full support for primary & secondary indexes
- Document model = less work

Replication




Shard 1
Shard 2
Shard 3
Shard N
Horizontally Scalable

Database Landscape



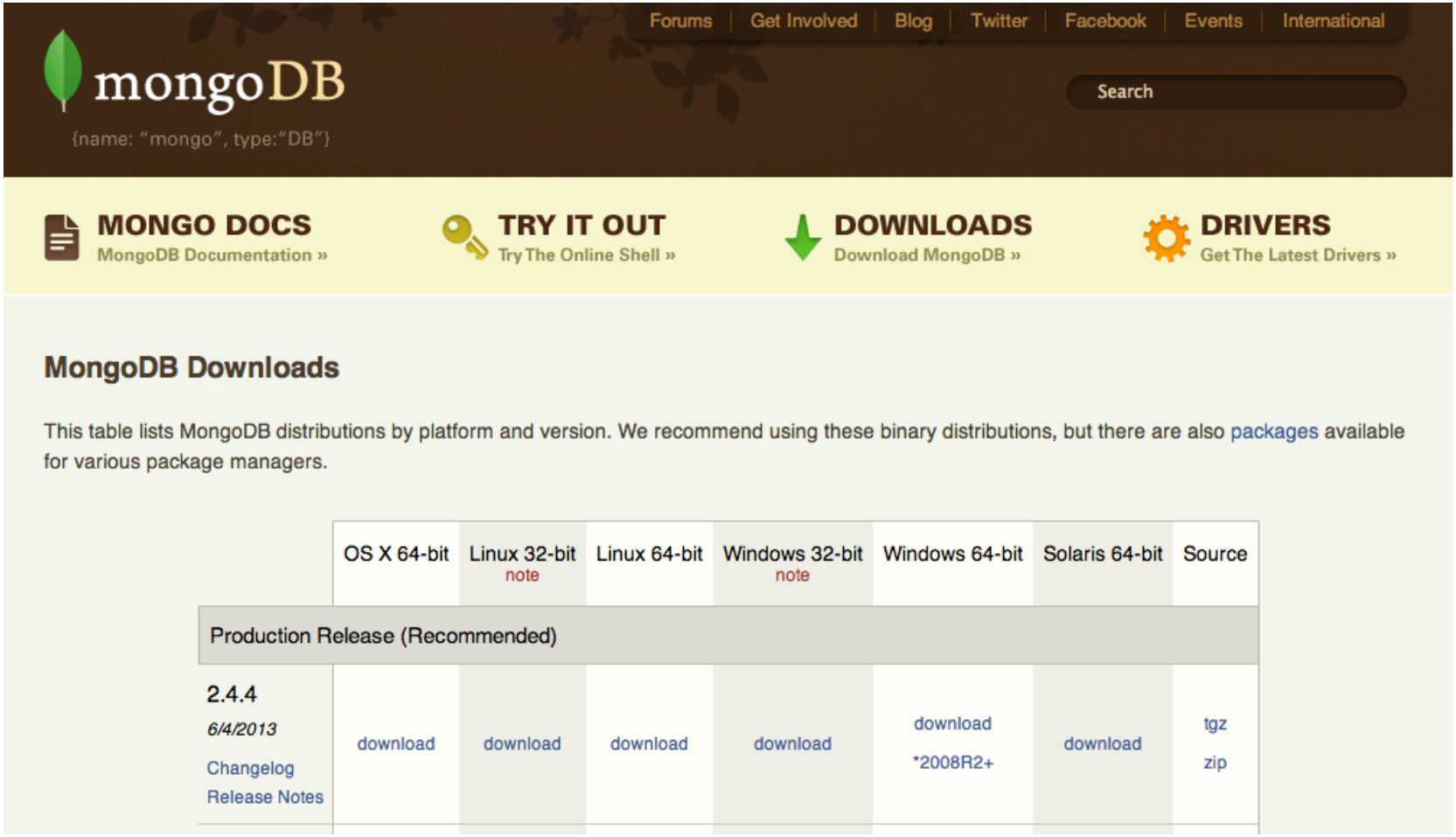
Full Featured

- Ad Hoc queries
- Real time aggregation
- Rich query capabilities
- Strongly consistent
- Geospatial features
- Support for most programming languages
- Flexible schema

Terminology

RDBMS		MongoDB
Table, View	➜	Collection
Row	➜	Document
Index	➜	Index
Join	➜	Embedded Document
Foreign Key	➜	Reference
Partition	➜	Shard

mongodb.org/downloads



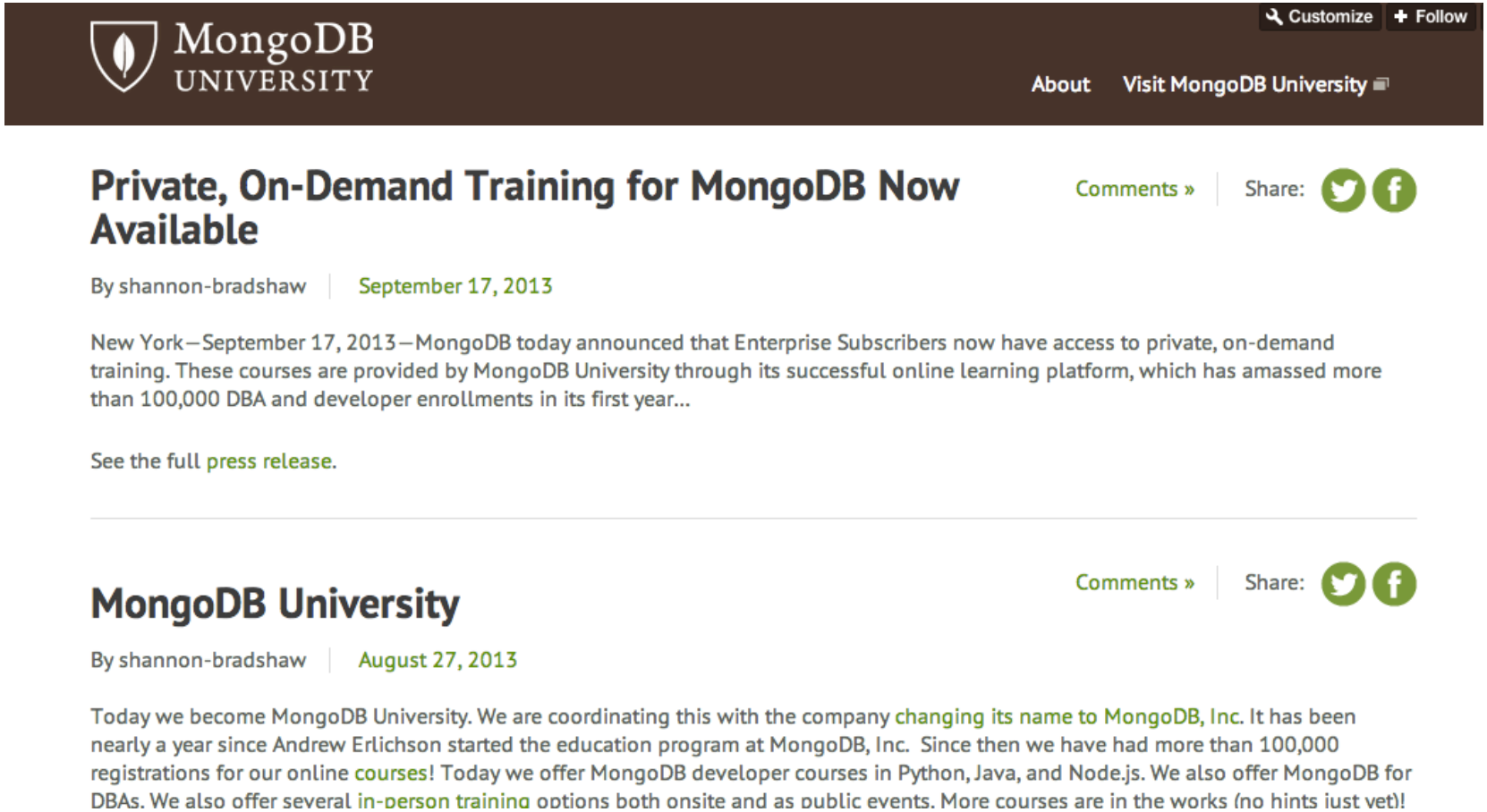
Running MongoDB

```
$ tar –z xvf mongodb-osx-x86_64-2.4.x.tgz
$ cd mongodb-osx-i386-2.4.4/bin
$ mkdir –p /data/db
$ ./mongod
```

Mongo Shell

```
MacBook-Air-:~ $ mongo
MongoDB shell version: 2.4.4
connecting to: test
> db.test.insert({text: 'Welcome to MongoDB'})
> db.test.find().pretty()
{
        "_id" : ObjectId("51c34130fbd5d7261b4cdb55"),
        "text" : "Welcome to MongoDB"
}
```

Let's Build a Blog



First step in any application is

Determine your entities

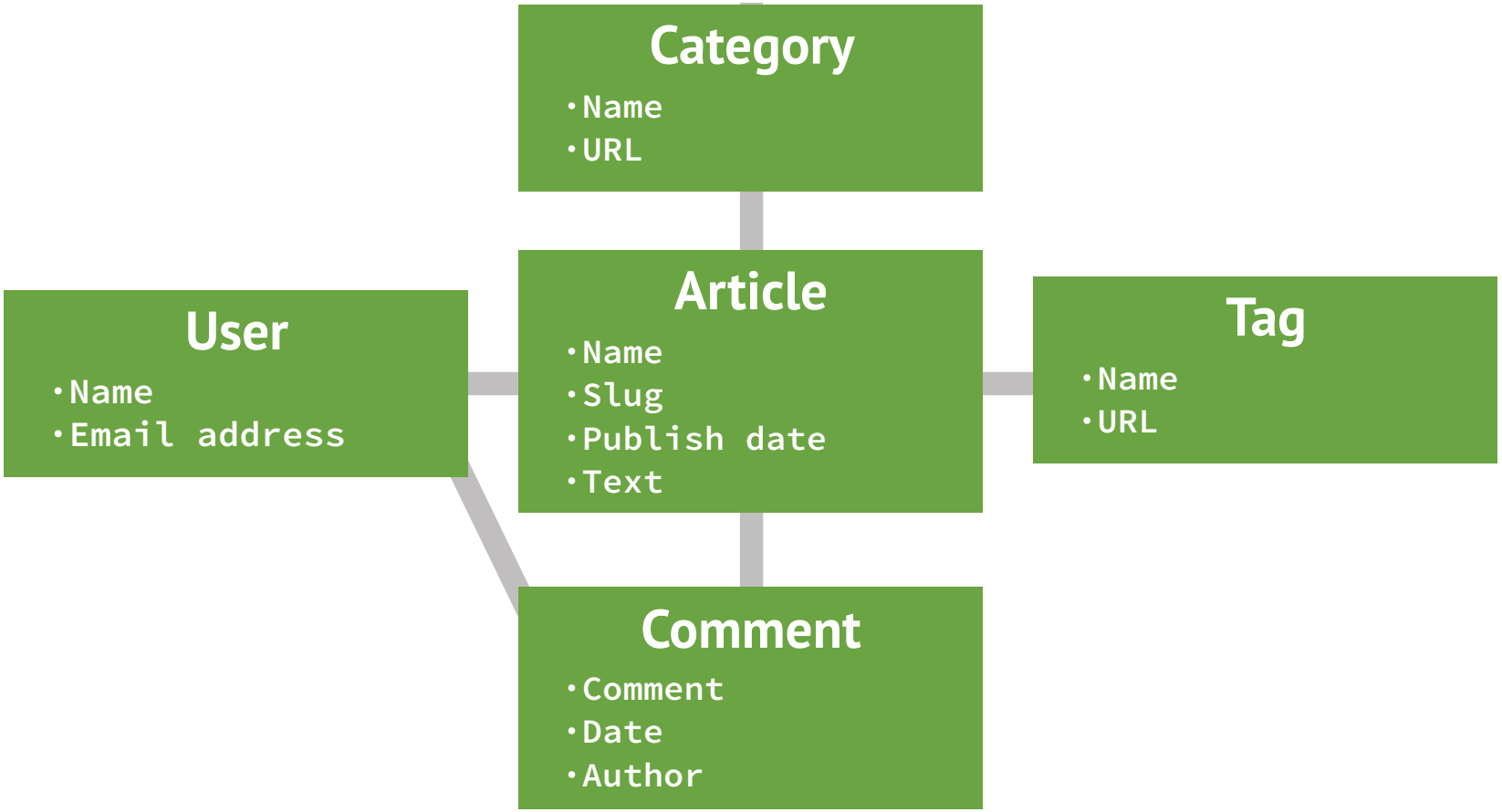
Entities in our Blogging System

- Users (post authors)
- Article
- Comments
- Tags

In a relational base app

We would start by doing schema design

Typical (relational) ERD

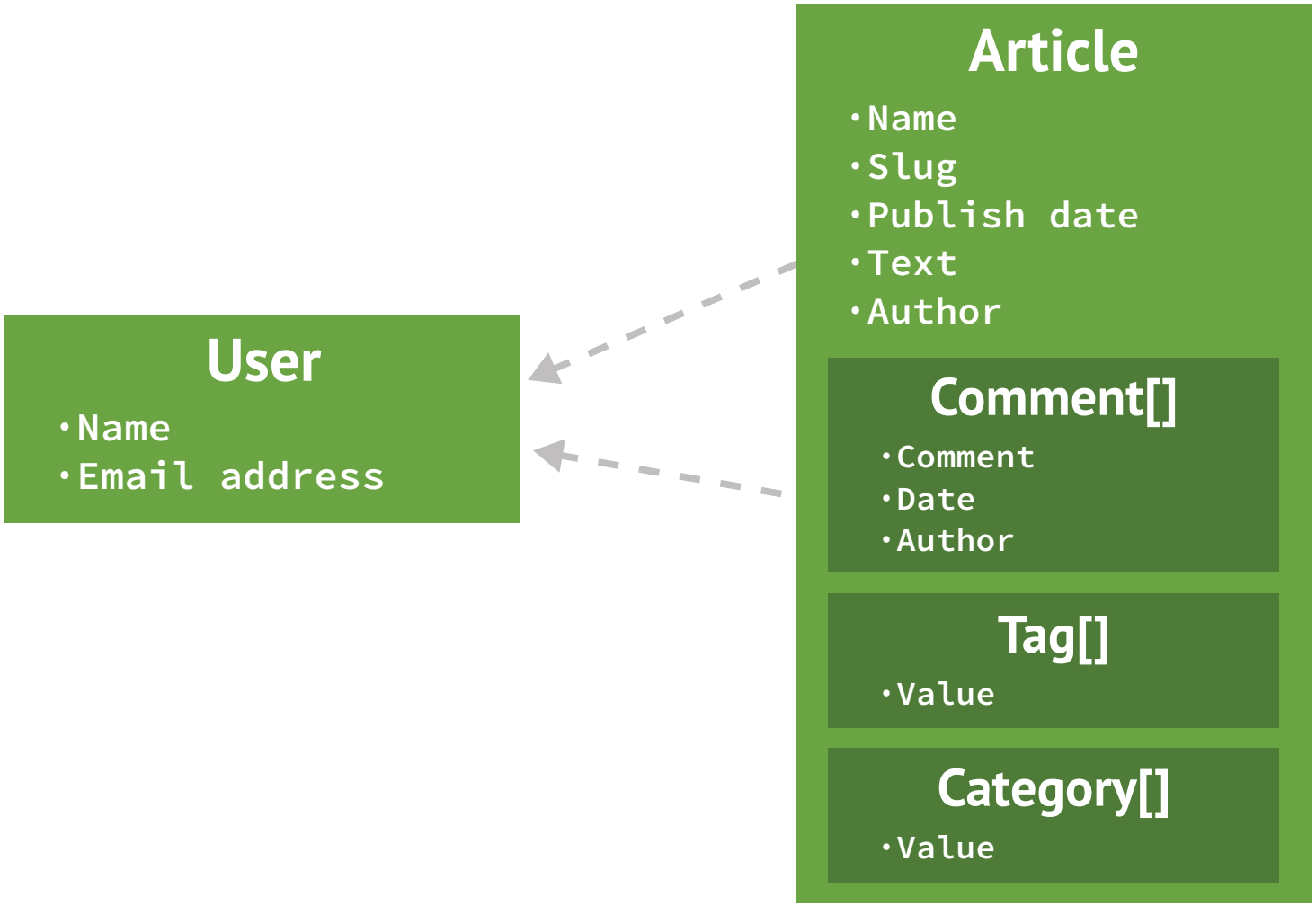


In a MongoDB based app

We start building our app

and let the schema evolve

MongoDB ERD


Article
•Name
•Slug
•Publish date
•Text
•Author
Comment[]
•Comment
•Date
•Author
Tag[]
•Value
Category[]
•Value
User
•Name
•Email address

Start with an object (or array, hash, dict, etc)

```
var user = {
                    username:   'erlichson',
                    first_name: 'Andrew',
                    last_name:  'Erlichson',
}
```

Switch to Your DB

```
>db
test
> use blog
 switching to db blog

> db.users.insert( user )
```

Insert the Record

```
> db.users.insert(user)
```

No collection creation necessary

Find One Record

```
> db.users.findOne()
{
        "_id" : ObjectId("50804d0bd94ccab2da652599"),
        "username" : ”erlichson",
        "first_name" : ”Andrew",
        "last_name" : ”Erlichson"
}
```

_id

- _id is the primary key in MongoDB
- Automatically indexed
- Automatically created as an ObjectId if not provided
- Any unique immutable value could be used

ObjectId

- ObjectId is a special 12 byte value
- Guaranteed to be unique across your cluster

```
ObjectId("50804d0bd94ccab2da652599")
          TS------Mac---PID-Count-
```

Creating a Blog Post

```
> db.article.insert({
                        title: 'Hello World',
                        body:  'This is my first blog post',
                        date: new Date('2013-06-20'),
                        username: 'erlichson',
                        tags: ['adventure', 'mongodb'],
                        comments: [ ]
})
```

Finding the Post

```
> db.article.find().pretty()
{
        "_id" : ObjectId("51c3bafafbd5d7261b4cdb5a"),
        "title" : "Hello World",
        "body" : "This is my first blog post",
        "date" : ISODate("2013-06-20T00:00:00Z"),
        "username" : "erlichson",
        "tags" : [
                "adventure",
                "mongodb"
        ],
        "comments" : [ ]
}
```

Querying An Array

```
> db.article.find({tags:'adventure'}).pretty()
{
        "_id" : ObjectId("51c3bcddfbd5d7261b4cdb5b"),
        "title" : "Hello World",
        "body" : "This is my first blog post",
        "date" : ISODate("2013-06-20T00:00:00Z"),
        "username" : "erlichson",
        "tags" : [
                "adventure",
                "mongodb"
        ],
        "comments" : [ ]
}
```

Using Update to Add a Comment

```
> db.article.update({_id:

      new ObjectId("51c3bcddfbd5d7261b4cdb5b")},

      {$push:{comments:

      {name: 'Steve Blank', comment: 'Awesome Post'}}})
>
```

Post with Comment Attached

```
> db.article.findOne({_id: new ObjectId("51c3bcddfbd5d7261b4cdb5b")})
{
        "_id" : ObjectId("51c3bcddfbd5d7261b4cdb5b"),
        "body" : "This is my first blog post",
        "comments" : [
                {
                        "name" : "Steve Blank",
                        "comment" : "Awesome Post"
                }
        ],
        "date" : ISODate("2013-06-20T00:00:00Z"),
        "tags" : [
                "adventure",
                "mongodb"
        ],
        "title" : "Hello World",
        "username" : "erlichson"
}
```

MongoDB Drivers

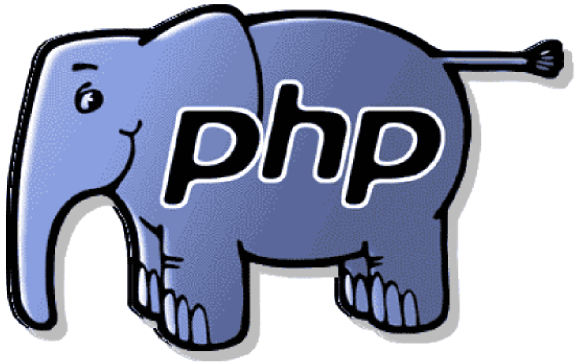
Real applications are not built in the shell

MongoDB has native bindings for over 12 languages


C++


Microsoft®
.NET


ERLANG


php


Scala


C

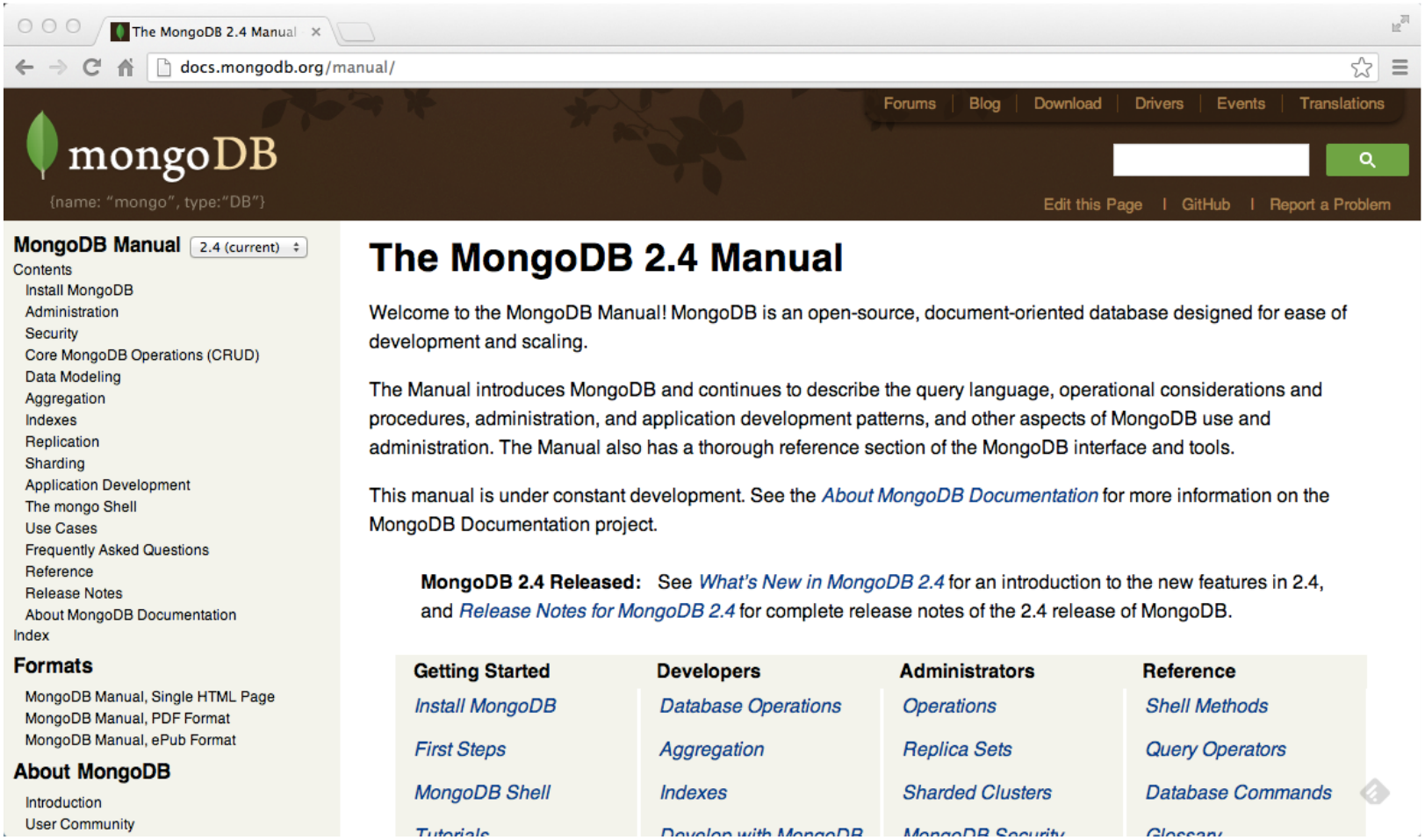

node.JS


ProLog
Lua
Microsoft®
.NET
ERLANG
small
TALK
Scala
php
nodeJS

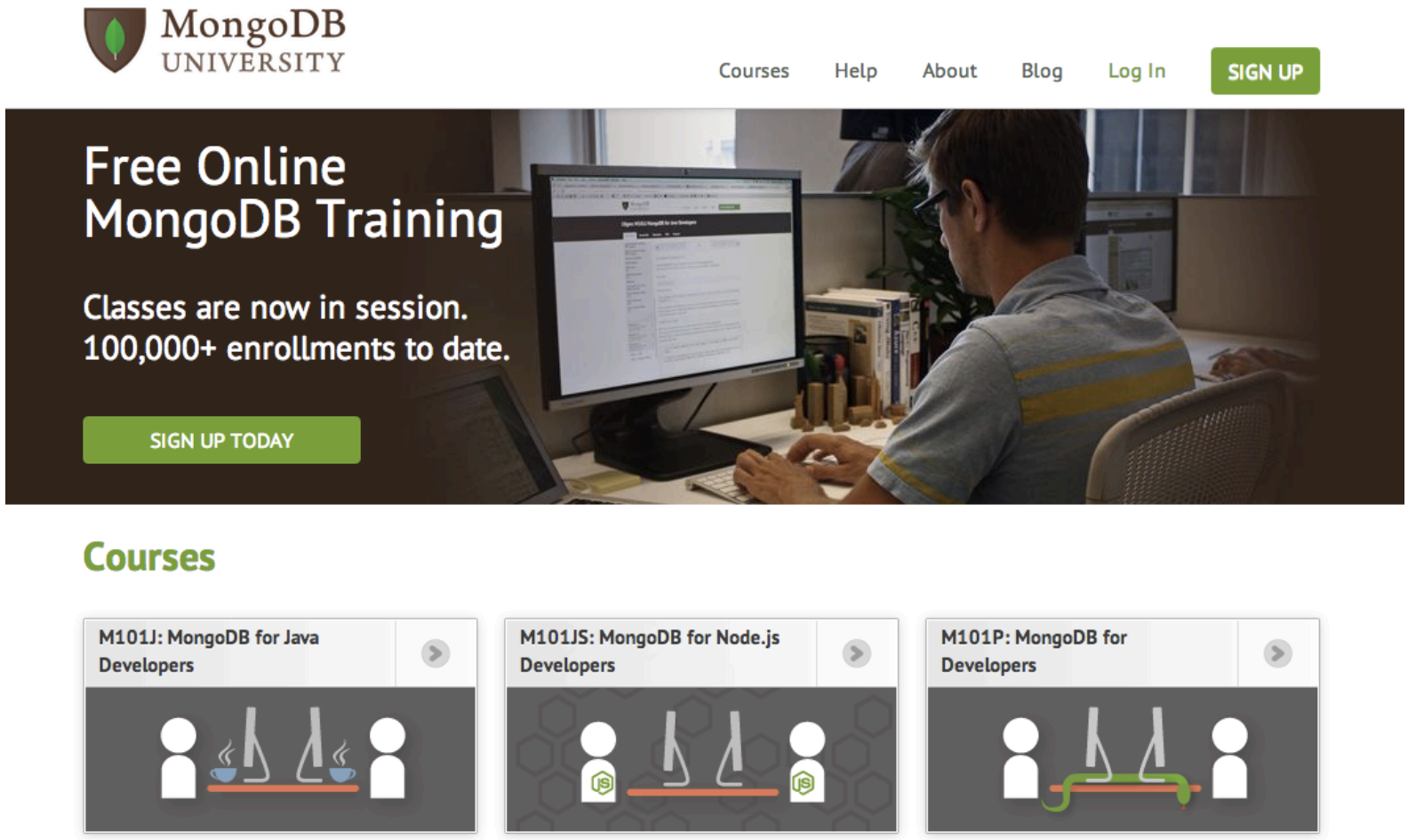
MongoDB Drivers

- Official Support for 12 languages
- Community drivers for tons more
- Drivers connect to mongo servers
- Drivers translate BSON into native types
- mongo shell is not a driver, but works like one in some ways
- Installed using typical means (npm, pecl, gem, pip)

docs.mongodb.org



Online Training at MongoDB University



Schema Design @