
mongoDB

MongoDB Introducing

1. What is NoSQL?
2. What is MongoDB?
3. Installing
4. Mongo Shell
5. Mongo Drivers and Client Libraries
6. REST and MongoDB
7. Sharding & Replication

What is NoSQL?

NoSQL (Not Only SQL)

> *NoSQL DEFINITION: Next Generation Databases mostly addressing some of the points: being non-relational, distributed, open-source and horizontally scalable. ... Often more characteristics apply such as: schema-free, easy replication support, simple API, eventually consistent / BASE (not ACID), a huge amount of data and more.*
>
> ***— nosql-database.org***

What is NoSQL?

- Нереляционная модель данных
- Открытый исходный код
- Хорошая горизонтальная масштабируемость
- Распределенность

Also

- Отсутствие схемы
- Простая поддержка репликации
- Простое API
- Согласованность в конечном счете (Eventually Consistent)
- Big Data

What is MongoDB?

What is MongoDB?

- Документо-ориентированная база данных
  - JSON/BSON
- Отсутствие схемы
- Производительность
  - Написана на C++
  - Поддержка индексов
  - Быстрое чтение и особенно запись

What is MongoDB?

- Масштабируемость
  - Репликация
  - Шардинг
- Коммерческая поддержка(10gen)
- Документация

Ещё

Map Reduce, GridFS, Geospatial Indexing.

Кто использует?


craigslist


Forbes


foursquare


MTV
MUSIC TELEVISION


CISCO


Disney


intuit.

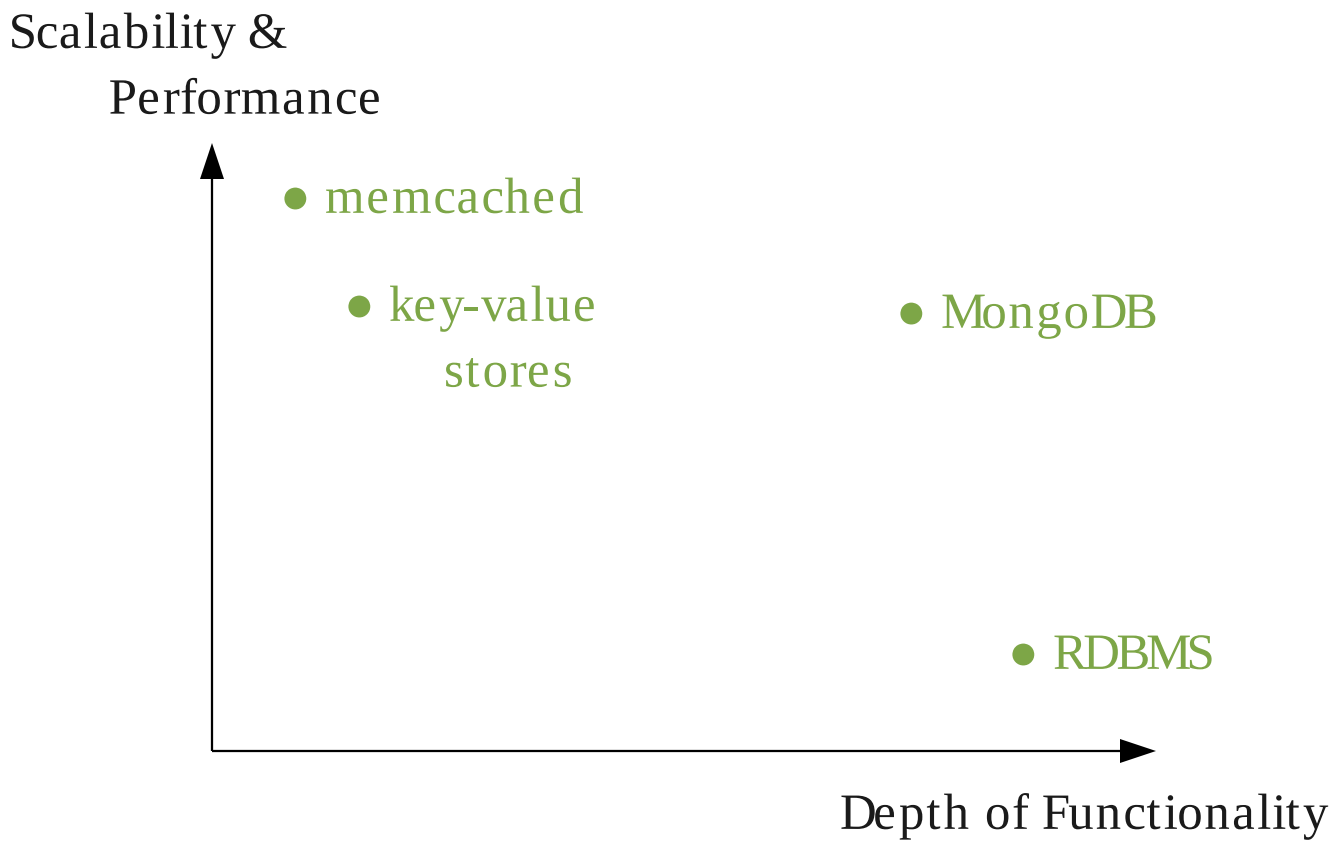

salesforce


GILT


McAfee
An Intel Company


The National Archives


shutterfly


Scalability &
Performance
memcached
key-value
stores
MongoDB
RDBMS
Depth of Functionality

Основные понятия

Collection

- Эквивалент таблицы базы данных
- Логически группирует документы
- Индексы распространяются на коллекцию

Document

- Аналог строки в RDBMS
- Представляет собой JSON(BSON)
- Иерархический
- Может ссылаться на другие документы

_id

- Специальный ключ
- Присутствует во всех документах
- Должен быть уникален в коллекции
- Любой тип

CRUD

CRUD

Create, read, update and delete

CRUD		MongoDB		SQL
Create	↔	Insert	↔	Insert
Read	↔	Find	↔	Select
Update	↔	Update	↔	Update
Delete	↔	Remove	↔	Delete

Create

```
db.collection.insert( {name : "apple",
                       color : "red", shape : "round"} );
```

Read

```
01. // Находит одну запись для { "username" : "dwight" },
                    выведет только поле "email".
02. db.collection.findOne( { "username" : "dwight" } ,
                    { "_id" : false , "email" : true } );
```

Read

```
01. // Найти документы где score=50, а type=essay,
                    отобразить только поле student
02. db.collection.find( {score:50, type:"essay"},
                    {student:true, _id:false})
03. db.collection.find().pretty(); // More readable
```

Read

```
01. // Найти все документы поле score которого
                        больше 50 и меньше 60.
02. db.collection.find( { score : { $gt : 50 , $lt : 60 } } );
03. db.collection.find( { score : { $gte : 50 ,
                    $lte : 60 } } ); // включительно
```

Операторы условий

$gt, $lt, $gte, $lte,

$ne, $in, $nin, $mod,

$all, $size, $exists,

$type, $not, $where

$regex

Фичи

```
01. db.collection.find( { 'dot.natation' : { $gte : 5 } } );
02. db.collection.find().sort( { score : -1 } )
                        .skip(50).limit(20);
03. // Сортировка в обратном порядке, пропустить 50 результатов,
                    отобразить 20
04. db.collection.count( { type : "essay", "score" : 90 } )
05. // Подсчет результатов
```

Update

```
01. db.collection.update( { field : "Value" },
                          { field1 : "InsertingNewValue" } );
02. // Находит документ с field=Value,
                          удаляет все поля кроме _id и вставляет fiel
```

Update

```
01. db.collection.update( { field : 'Value' },
                          { $set : { 'country' : 'RU' } } );
02. // Обновить поле country или добавить его.
```

Операторы модификации

$set, $unset, $inc,

$push, $pushAll, $addToSet,

$pop, $pull, $pullAll

Фичи

```
01. db.collection.update( { field : 'Value' },
                          { $set : { 'country' : 'RU' } },
                           { upsert : true } );
02. // Если документа с field=Value нет,
                          то будет создан такой элемент
```

Фичи

```
01. db.collection.update( {},
                          { $set : { 'country' : 'RU' } },
                           { multi : true } );
02. // Находим все документы, и устанавливаем страну - RU
                          для всех документов
                          (без multi - только для одного)
```

Delete

```
01. db.scores.remove( { score : { $lt : 60} } );
02. // Удалить все записи, у которых поле score
                        имеет значение меньше 60
```

Индексирование

- Индекс
- Multikey Indexes
- Geospatial Indexes
- TTL Indexes
- Index Explain
- Profiling
- Force using some index

Aggregation Framework

- Pipelines:
- Project - $project - select keys, reshape - 1:1
- Match - $match - filter - n:1
- Group - $group - aggregation - n:1
- Sort - $sort - 1:1
- Skip - $skip - n:1
- Limit - $limit - n:1
- Unwind - $unwind - 1:n

Aggregation Framework - Sample

```
db.collection.aggregate( [
    { $match : { state : "NY" } },
    { $group : { _id : "city", population: { $sum : "$pop" }
    zip_codes : { $addToSet : "$_id" } } },
    { $project : { _id : 0, city : "$_id",
     population : "$population", zip_codes : 1 } }
] );
```

Installing

Установка

- `$ wget http://fastdl.mongodb.org/linux/mongodb-linux-i686-2.2.3.tgz`
- `$ tar -xf mongodb-linux-i686-2.2.3.tgz`
- `$ cd mongodb-linux-i686-2.2.3/bin`
- `$ mkdir -p ./data/db`
- `$ ./mongod --dbpath ./data/db/ & # Start Server`
- `$ ./mongo # Start Shell`

Просто!

Mongo Shell

MongoDB Drivers and Client Libraries

- C
- C++
- C#
- Erlang
- Java
- JavaScript

MongoDB Drivers and Client Libraries

- Node.js
- Perl
- PHP
- Python
- Ruby
- Scala

+Community

ActionScript, Clojure, Delphi, F#, Go, Groovy, Lua, Objective C, Smalltalk, etc.

Полный список: Community Supported Drivers

REST

Accessing MongoDB via REST

- Чтобы включить базовый REST интерфейс:
  - `./mondod --rest`
- REST интерфейс использует port+1000
  - http://127.0.0.1:28017/database/collection/
  - http://127.0.0.1:28017/database/collection/?filter&Field=Value

Replication

Replication

A MongoDB replica set is a cluster of mongod instances that replicate amongst one another and ensure automated failover. Most replica sets consist of two or more mongod instances with at most one of these designated as the primary and the rest as secondary members. Clients direct all writes to the primary, while the secondary members replicate from the primary asynchronously.

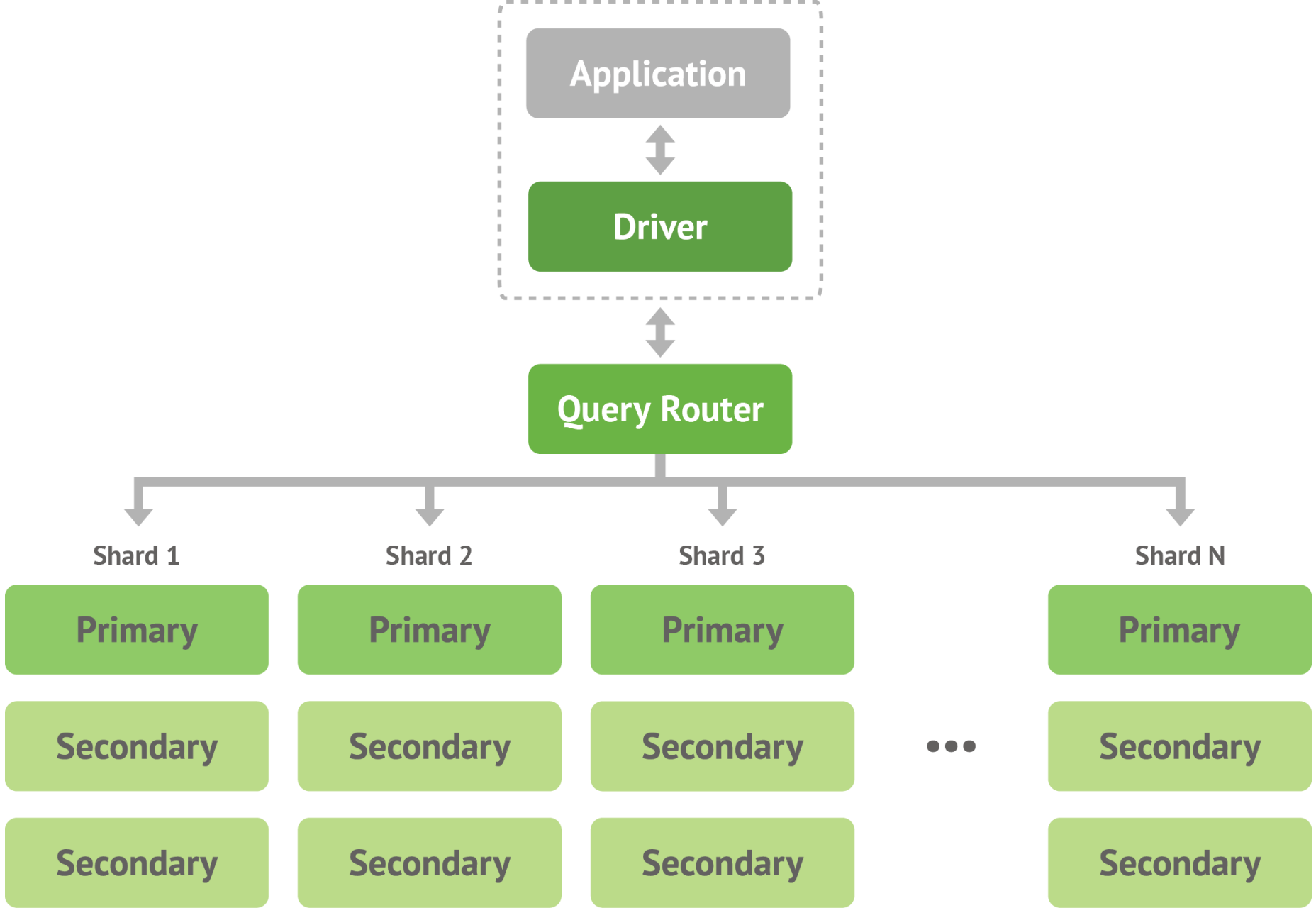
Member Configuration Properties

- Secondary-Only
- Hidden
- Delayed
- Arbiters
- Non-Voting

Sharding

Sharding

Sharding is MongoDB's approach to scaling out. Sharding partitions a collection and stores the different portions on different machines. When a database's collections become too large for existing storage, you need only add a new machine. Sharding automatically distributes collection data to the new server. Sharding automatically balances data and load across machines. Sharding provides additional write capacity by distributing the write load over a number of mongod instances. Sharding allows users to increase the potential amount of data in the working set.


Application
Driver
Query Router
Shard 1
Shard 2
Shard 3
Shard N
Primary
Primary
Primary
Primary
Secondary
Secondary
Secondary
Secondary
Secondary
Secondary
Secondary
Secondary

Обучение

Книги

https://www.10gen.com/books

http://cookbook.mongodb.org

The Little MongoDB Book [RU]

Курсы

https://education.10gen.com/

- M101J: MongoDB for Java Developers **Starts Feb 25, 2013**
- M101P: MongoDB for Developers(Python) **Starts Jan 21, 2013**
- M102: MongoDB for DBA **Starts Jan 21, 2013**

Курсы

По окончании выдается сертификат(>70%):

10gen | The MongoDB Company

CERTIFICATE
Dec. 24th, 2012

This is to certify that

Ruslan Ibragimov

successfully completed

M101: MongoDB for Developers

a course of study offered by 10gen, The MongoDB Company

Dwight Merriman
Chief Executive Officer
10gen, Inc.

Andrew Erlichson
Vice President, Education
10gen, Inc.

Authenticity of this certificate can be verified at https://education.10gen.com/downloads/certificates/0235a75717f04884949bd1516d7b2791/Certificate.pdf

Ресурсы в интернете

- https://groups.google.com/forum/?fromgroups#!forum/mongodb-user
- irc://irc.freenode.net/#mongodb
- http://mongotips.com/
- http://www.mongodb.org/

Спасибо!

?