



南京大学  
NANJING UNIVERSITY



## MongoDB简介

南京大学软件学院

任桐炜 李传艺

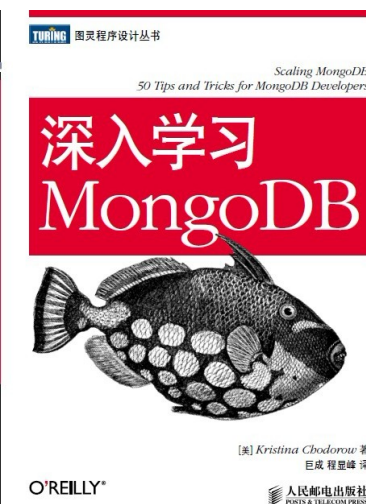
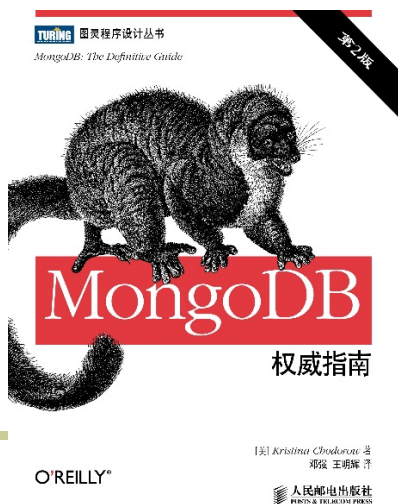
{rentw, lcy}@nju.edu.cn



## 推荐参考资料



- 《MongoDB权威指南》第二版，Kristina Chodorow著，邓强 王明辉译：<http://www.linuxidc.com/Linux/2016-12/138253.htm>
- 《深入学习MongoDB》(Scaling MongoDB: 50 Tips and Tricks for MongoDB Developers)，Kristina Chodorow著，巨成 程显峰译：<http://www.linuxidc.com/Linux/2016-04/129810.htm>
- MongoDB官方用户手册v3.4：  
<https://docs.mongodb.com/manual/>





## 目录



- 简介
- MongoDB基础知识
- 数据库基本操作
- 索引
- 分片的概念
- 建立和启动MongoDB集群
- 选择片键



# MongoDB简介 (1)



- MongoDB is an open-source, document database designed for ease of development and scaling.

- 易使用

- 面向文档的数据库，没有行的概念，支持嵌入文档和数组，没有预定义的表结构

```
{  
  name: "sue",  
  age: 26,  
  status: "A",  
  groups: [ "news", "sports" ]  
}
```

← field: value  
← field: value  
← field: value  
← field: value

- 易扩展

- 纵向扩展：增强服务器能力
- 横向扩展：增加服务器数量；数据很容易在多台服务器间分割；自动处理跨集群的数据和负载；准确路由用户请求等

- 分片Sharding



## MongoDB简介（2）



### ■ 多种开发语言的驱动

#### Getting Started

MongoDB provides a [Getting Started Guide](#) in the following editions.

[mongo Shell Edition](#)

[Python Edition](#)

[Java Edition](#)

[Ruby Edition](#)

[Node.JS Edition](#)

[C++ Edition](#)

[C# Edition](#)

### ■ 索引Indexing

- 通用二级索引、唯一索引、复合索引、地理空间索引、全文索引等

### ■ 聚合Aggregation

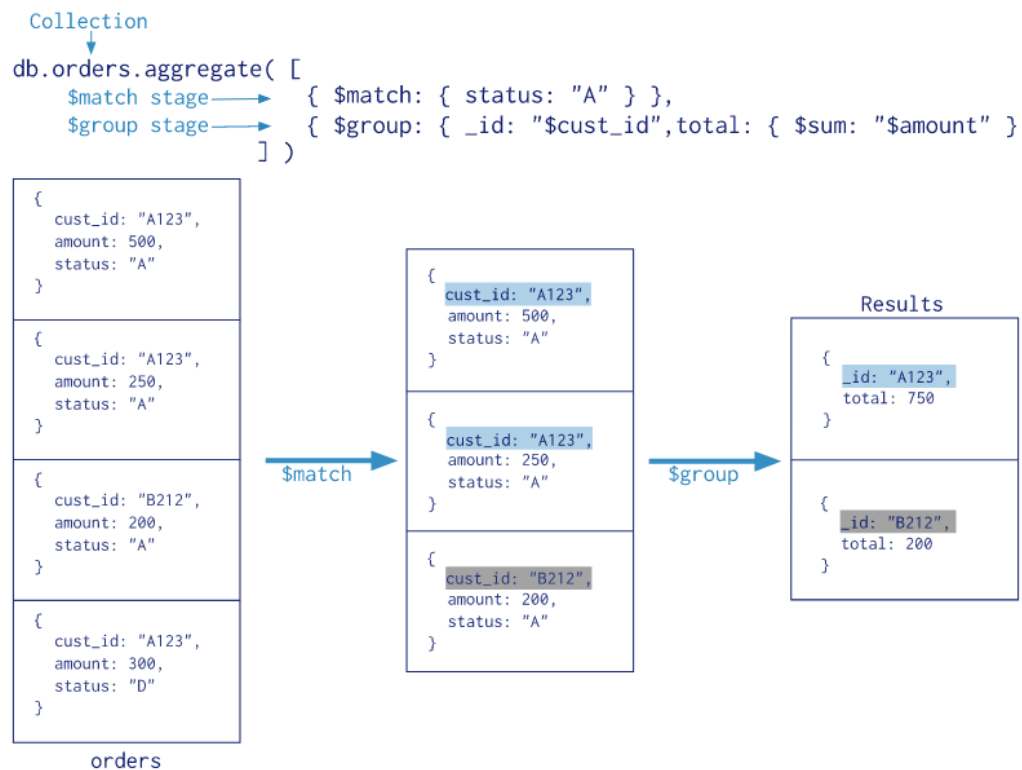
- 聚合管道、Map-Reduce

### ■ 特殊的集合类型

- 例如存在时间有限的集合

### ■ 文件存储

- 支持存储大文件和文件元数据





# MongoDB基础知识（1）



## ■ 文档Document

- 键值对的有序集
- $\{\text{key}_1:\text{value}_1, \text{key}_2:\text{value}_2, \text{key}_3:\text{value}_3, \dots, \text{key}_n:\text{value}_n\}$
- **value**可以是任意数据类型，包括数组和文档
- 同一个文档中**key**不能重复
- 其它细节要求

## ■ 集合Collection

- 一组文档就是一个集合；类似关系型数据库中的“表”
- 同一个集合中的文档不一定是相同的格式；使用的时候最好是相同的

## ■ 数据库Database

- 多个集合构成了数据库

## ■ 文档的键\_id

- 对应的键值在同一个集合中是唯一的
- 可以是系统自动生成的**ObjectId**，也可以是用户指定的



## MongoDB基础知识（2）



- MongoDB以BSON documents格式存储数据
  - BSON是JSON的二进制表示，但是比JSON支持更多的数据类型
- 数据类型包括
  - null
  - boolean
  - 数值：浮点数或者整型
  - 字符串
  - 日期
  - 正则表达式
  - 数组
  - 内嵌文档
  - 对象id
  - 二进制数据
  - 代码



## MongoDB基础知识（3）



- 安装
  - 社区版
  - 企业版
  - 不同操作系统（Linux, Mac, Windows）
  - 如何在Red Hat上安装社区版
    - <https://docs.mongodb.com/manual/tutorial/install-mongodb-on-red-hat/>
- mongo Shell
  - 连接MongoDB的交互式JavaScript Shell接口
- 可以运行JS脚本
- 是一个独立的MongoDB客户端
  - 除了JS功能，包括一些数据库操作的命令





# 数据库基本操作（1）



## ■ 创建数据库、创建集合

- 如果数据库、集合不存在，只有第一次写入数据的时候才创建

- use myNewDB
- db.myNewCollection1.insertOne( { x: 1 } )

- 可以显式地创建集合

- db.createCollection()

## ■ 文档的CRUD操作

- Create:

- db.collection.insertOne()
- db.collection.insertMany()
- db.collection.insert()

- 当更新或者查询并修改等操作发现文档不存在时，会插入新文档

- Read: db.collection.find()等

- Update: db.collection.updateOne(); updateMany(); replaceOne()

- Delete: db.collection.deleteOne(); deleteMany()

- <https://docs.mongodb.com/manual/tutorial/insert-documents/>

MongoDB CRUD Operations > Insert Documents

### Insert Documents

Mongo Shell

Python

Java (Sync)

Node.js

PHP

Other ▾

This page provides examples of insert operations in MongoDB.

#### CREATING A COLLECTION:

If the collection does not currently exist, insert operations will create the collection.

#### Insert a Single Document

New in version 3.2.



## 数据库基本操作（2）



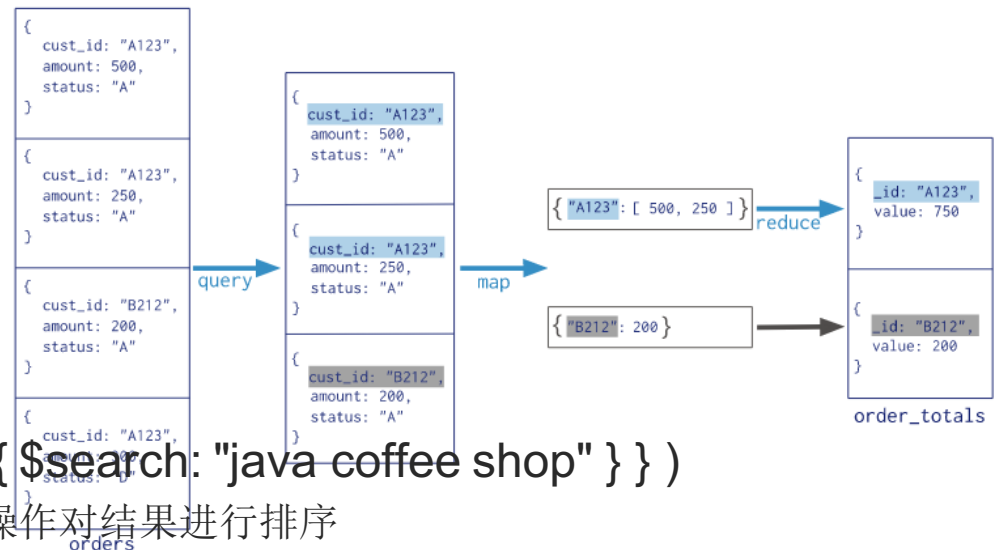
### ■ 聚合

- 聚合管道（各种操作）
- MapReduce:
  - 在满足query条件的输入上Map

```
Collection
↓
db.orders.mapReduce(
  map   → function() { emit( this.cust_id, this.amount ); },
  reduce → function(key, values) { return Array.sum( values ); },
  query → { query: { status: "A" },
  output → out: "order_totals"
}
```

### ■ 文本搜索

- 对字符串值进行索引
- 在字符串的索引上搜索
- 步骤:
  - 建立索引createIndex({k:v})
  - 使用\$text和\$search操作查询
    - `db.stores.find( { $text: { $search: "java coffee shop" } } )`
  - 还可以使用sort()方法和\$meta操作对结果进行排序
    - `db.stores.find( { $text: { $search: "java coffee shop" } },  
{ score: { $meta: "textScore" } }  
).sort( { score: { $meta: "textScore" } } )`





## 索引 (1)



- 通用二级索引
  - 索引必须依赖一个key
  - Dense Index(指向非排序全局数据); Sparse Index(指向排序的Dense Index)
  - Cluster Index(将全局数据按照primary key排序后使用的Sparse Index); Secondary Index(指向非primary key列的Dense Index的Sparse Index)
- 复合索引
  - 在多个key上建立一个索引
- 索引对象和数组
  - 对嵌套文档的内容建立索引、对数组里每个元素建立索引
- 唯一索引
  - 文档在索引指定键的值具有唯一性
- 全文索引



## 索引 (2)



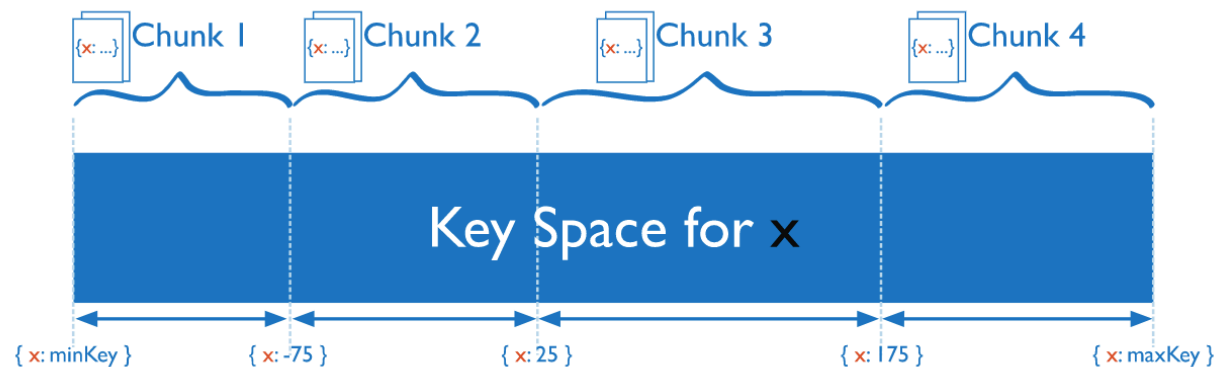
- 地理空间索引
- 地理位置信息保存为GeoJSON或者legacy coordinate pairs中
- GeoJSON Object
  - `<field>: { type: <GeoJSON type> , coordinates: <coordinates> }`
  - 2dsphere索引
  - `db.collection.createIndex( { <location field> : "2dsphere" } )`
- Legacy coordinate pairs
  - `<field>: [ <x>, <y> ]`
  - `<field>: { <field1>: <x>, <field2>: <y> }`
  - 2d索引
  - `db.collection.createIndex( { <location field> : "2d" } )`
- 一系列基于地理空间索引的查询操作
  - `$geoWithin`
  - `$near`
  - `$nearSphere`等



## 分片（1）



- 分片 **Shard** 是 MongoDB 用来将大型集合分割到不同服务器上采用的方法
- 一个集合可以被分为多个分片
- 一个分片存储在一台服务器上
  - 一个分片可以有多个副本，副本存在其他服务器上
  - 存储同一个分片的多台服务器称为副本集
- 用来对集合进行分片的键称为“片键”
- 一个分片就是集合数据中片键在
  - 某个区间上的数据子集：ranged sharding; (还有 Hashed Sharding)





## 分片（2）



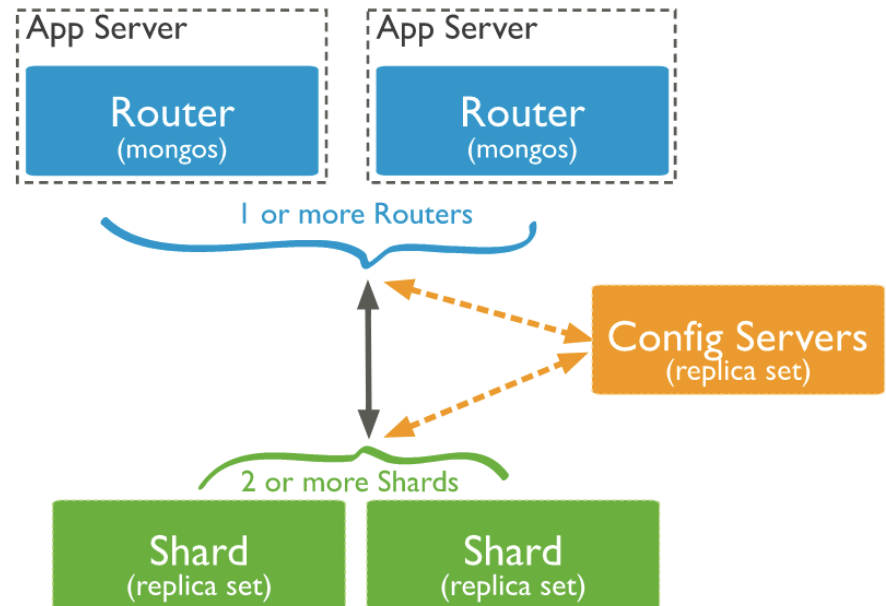
- 一个分片包含多个片键区间
  - 每个区间称为数据“块”
  - 当块变得越来越大时，会将其分割成两个较小的块
  - 设置块的大小：默认200M
- 分片平衡
  - 通过移动块使得分片的数据量得到平衡
    - 新加一个分片服务器
      - 已有分片上的数据都移动一部分到新分片上
    - 原有分片平衡
      - 例如，当一个分片比另一个分片多出9个块，触发平衡操作



## 建立和启动MongoDB集群



- 配置服务器 mongod
  - “1个或3个”
  - 测试环境使用1个就够了
  - 配置服务器之间交互比较复杂，生产环境使用3个比较适合
- mongos进程
  - 隐藏集群内分片的复杂性，向用户提供简洁的单服务器接口
- 分片
  - 选择片键
  - 数据分片
  - 新增分片
  - 移除分片





## 选择片键（1）



### ■ 好片键的反例

#### ○ 小基数片键

- 利用取值很少的字段作为片键，如性别、民族等

#### ○ 升序片键

- 例如使用类似时间戳的字段作为片键
- 优点是近段时间需要访问的数据在内存中的概率更大
- 问题是导致新插入数据总是在同一个片键上，单个分片压力过大

#### ○ 随机片键

- 选择取值随机的字段作为片键
- 开始效果不错，但是随着数据量增大，效果变差
- 考虑当需要对分片进行平衡时，会将一个分片的一些块放到另一些分片上
- 结果是：每个分片上随机分布着数据，可能需要放在内存中的常用数据也随机分布在各个分片中，使得磁盘压力变大，导致速度慢





## 选择片键（2）



### ■ 好片键

- 具备良好的数据局部性特征——高效查询
  - 相关数据会保持在同一个局部内，避免内存数据高频率切换
- 不会产生数据热点——高效集群
  - 数据的插入、修改操作均匀的分布在多个分片上，而不是集中在某一个分片

### ■ 使用组合片键

- 一个字段用于将数据分块
- 一个字段用于保持数据局部性特征



## 选择片键（3）



### ■ 选择片键应该思考的问题

- 写操作是怎样的？需要插入的文档类型和大小？
- 系统写入数据的频率如何？
- 哪些字段是随机值，哪些是增长的？
- 读操作是怎样的？经常访问的数据有哪些？
- 系统读取数据的频率如何？
- 数据有索引吗？是否需要索引？
- 数据总量有多少？



## 集群的使用和管理



- 用户访问的是mongos进程，已经封装好集群的内部的复杂过程
- 集群的管理
  - 通过mongo shell进行集群管理
  - 监控
    - 最全面的监控命令mongostat
    - 使用web管理界面
  - 可能出现的服务器错误
    - 分片停机
    - 多数分片停机
    - 配置服务器停机
    - mongos进程死掉：尝试重启崩溃的mongos进程



## MongoDB开发技巧50例



- 通过使用MongoDB的实践增强对MongoDB机制的理解，并提高使用技巧
- 经常查阅《MongoDB开发技巧50例》解决遇到的问题



谢 谢！