

# MongoDB 简介

李焕贞

河北师范大学软件学院

## 本章大纲

- MongoDB数据库介绍
- 安装MongoDB数据库
- 使用 MongoDB shell访问MongoDB
- 使用Java程序访问 MongoDB

# MongoDB数据库介绍

- 名称源自“humongous”（意为“巨大无比”）
- MongoDB 是由C++语言编写的，是一个基于分布式文件存储的开源数据库系统。
- 在高负载的情况下，添加更多的节点，可以保证服务器性能。
- MongoDB 旨在为WEB应用提供可扩展的高性能数据存储解决方案。

官方网站: <https://www.mongodb.com/>

开源项目: <https://github.com/mongodb/>

下载地址: [http://dl.mongodb.org/dl/win32/x86\\_64](http://dl.mongodb.org/dl/win32/x86_64)



**mongodb**

<http://www.mongodb.com/>

# MongoDB数据库介绍

MongoDB 将数据存储为一个文档，数据结构由键值(key=>value)对组成。

MongoDB 文档类似于 JSON 对象。字段值可以包含其他文档，数组及文档数组。

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



Diagram illustrating the structure of a MongoDB document (JSON object) with field-value pairs:

- name: "sue", ← field: value
- age: 26, ← field: value
- status: "A", ← field: value
- groups: [ "news", "sports" ] ← field: value

# MongoDB数据库介绍

## 主要特点

- 提供了一个面向文档存储，操作起来比较简单和容易
- 可以设置任何属性的索引来实现更快的排序
- 具有较好的水平可扩展性
- 支持丰富的查询表达式，可轻易查询文档中内嵌的对象及数组
- 可以实现替换完成的文档（数据）或者一些指定的数据字段
- MongoDB中的Map/Reduce主要是用来对数据进行批量处理和聚合操作
- 支持各种编程语言: RUBY, PYTHON, JAVA, C++, PHP, C#等语言
- MongoDB安装简单

# MongoDB数据库介绍

## MongoDB概念解析

SQL术语/概念	MongoDB术语/概念	解释/说明
database	database	数据库
table	collection	数据库表/集合
row	document	数据记录行/文档
column	field	数据字段/域
index	index	索引
table joins		表连接, MongoDB不支持
primary key	primary key	主键, MongoDB自动将_id字段设置为主键

# MongoDB数据库介绍

id	user_name	email	age	city
1	Mark Hanks	mark@abc.com	25	Los Angeles
2	Richard Peter	richard@abc.com	31	Dallas

```
{
  "_id": ObjectId("5146bb52d8524270060001f3"),
  "age": 25,
  "city": "Los Angeles",
  "email": "mark@abc.com",
  "user_name": "Mark Hanks "
}
{
  "_id": ObjectId("5146bb52d8524270060001f2"),
  "age": 31,
  "city": "Dallas",
  "email": "richard@abc.com",
  "user_name": "Richard Peter "
}
```

# MongoDB数据库

举例：在一个关系型数据库中，一篇博客（包含文章内容、评论、评论的投票）会被打散在多张数据表中。在文档数据库MongoDB中，能用一个文档来表示一篇博客， 评论与投票作为文档数组，放在正文主文档中。这样数据更易于管理，消除了传统关系型数据库中影响性能和水平扩展性的“JOIN”操作。

author:

pid	tid	name
1	1	Jane

blogposts:

tid	cid	title
1	1	"MyFirstPost"
1	2	"MyFirstPost"

comments:

cid	by	text
1	"Abe"	"First"
2	"Ada"	"Good post"



## MongoDB数据库介绍

关系数据库中的其中一条记录，在文档数据库MongoDB中的存储方式类似如下：

```
{
  "id" : 1,
  "author" : "Jane",
  "blogposts" : {
    "tile" : "MyFirstPost",
    "comment" : {
      "by" : "Ada",
      "text" : "Good post"
    }
  }
}
```

# MongoDB数据库介绍

## 数据库

- 一个Mongodb中可以建立多个数据库
- MongoDB的默认数据库为“db”，该数据库存储在data目录中
- MongoDB的单个实例可以容纳多个独立的数据库，每一个都有自己的集合和权限，不同的数据库也放置在不同的文件中
- 多个文档组成集合，而多个集合组成了数据库
- `show dbs` 命令可以显示所有数据的列表

# MongoDB数据库介绍

关于数据库的命名需要注意以下几点：

- 不能是空字符串（""）
- 不能以\$开头
- 不能含有. 和空字符串。
- 数据库名字区分大小写（**建议数据库名全部使用小写**）
- 数据库名字长度最多64字节。
- 不要与系统保留的数据库名字相同，这些数据库包括：admin、local、 config等

## 集合

- 集合就是 MongoDB文档组，类似于RDBMS 中的表格。
- 集合存在于数据库中，没有固定的结构，这意味着可对集合可以插入不同格式和类型的数据
- MongoDB 数据类型较多，有字符型、整型、布尔型、双精度等

合法的集合名：

- 集合名不能是空字符串""
- 集合名不能含有\0字符（空字符），（表示集合名的结尾）
- 集合名不能以system. 开头，此前缀是系统本身保留的
- 集合名中不能包含\$字符（注：可包含.）

# MongoDB数据库介绍

**文档**是MongoDB最核心的概念，本质是一种类JSON的**BSON**格式的数据。

BSON是一种类JSON的二进制格式数据，它可以理解为JSON基础上添加了一些新的数据类型，包括日期、int32、int64等。

BSON是由一组组键值对组成，它具有轻量性、可遍历性和高效性三个特征。**可遍历性是MongoDB将BSON作为数据存储的主要原因。**

BSON官网地址：<http://bsonspec.org/>

```
{  
  field1: value1,  
  field2: value2,  
  field3: value3,  
  ...  
  fieldN: valuen  
}
```

# MongoDB数据库介绍

使用MongoDB文档时需要注意以下问题：

- MongoDB中写操作的原子性限制在文档级别，对文档的保存、修改、删除等都是原子操作
- 单个文档占用的存储空间不能超过16MB
- MongoDB会尽量保持文档被插入时键值对的顺序

关于文档键的命名需要注意以下几点：

- `_id`是系统保留的关键字，它是默认的主键，该值在集合中必须唯一，且不可修改
- 键不能包含`\0`或空字符。（这个字符用于表示键的结尾）
- 不能以`$`开头
- 不能包含`.`（点号）
- 键是区分大小写的且不能重复 例如：`{foo:1, Foo:1}`



## MongoDB文档与集合-命名空间

把数据库名添加到集合名字前面，中间用点号连接，得到集合的完全限定名，就是命名空间，例如：命名空间parent.sub。

需要说明的是，点号还可以出现在集合名字中，例如：  
parent.subling0, parent.subling2可以将subling0和subling2集合看作是parent集合的子集合。

使用子集合可以使我们更好的组织数据，使数据的结构更加清晰明了。

## 适用场景

- 1、数据缓存
- 2、JSON格式的数据
- 3、高伸缩性场景
- 4、弱事务型业务

MongoDB更多适合于**大数据量、高并发、弱事务**的互联网应用，其内置的水平扩展机制提供了从几百万到十亿级别的数据处理能力，可以很好的满足Web2.0和移动互联网应用的数据存储要求。

## 本章大纲

- MongoDB数据库介绍
- 安装MongoDB数据库
- 使用 MongoDB shell访问MongoDB
- 使用Java程序访问 MongoDB

# 安装MongoDB

## Window平台安装 MongoDB

MongoDB提供了可用于32位和64位系统的预编译二进制包，你可以从MongoDB官网下载安装，MongoDB预编译二进制包下载地址：<http://www.mongodb.org/downloads>

## Linux平台安装MongoDB

MongoDB提供了linux平台上32位和64位的安装包，你可以在官网下载安装包。

下载地址：<http://www.mongodb.org/downloads>

## 启动 MongoDB服务

只需要在MongoDB安装目录的bin目录下执行' mongod' 即可

## 本章大纲

- MongoDB数据库介绍
- 安装MongoDB数据库
- 使用 MongoDB shell访问MongoDB
- 使用Java程序访问 MongoDB

# 使用MongoDB shell访问MongoDB

## 1、连接服务器

使用 MongoDB shell 来连接 MongoDB 服务器

```
mongodb://localhost
```

## 2、登陆数据库

使用用户名和密码连接登陆到指定数据库：

```
mongodb://admin:123456@localhost/test
```

## 3、创建数据库

语法格式： `use DATABASE_NAME`

## 4、创建集合

在插入数据的时候，MongoDB会自动创建对应的集合。

## 5、插入文档

```
db.COLLECTION_NAME.insert(document)
```

## 使用MongoDB shell访问MongoDB

admin数据库： 权限数据库，如果创建新用户，将该用户添加到admin数据库中，

那么该用户就自动继承了所有数据库的权限

local数据库： 用来存储本地服务器的任意集合

config数据库： 使用分片模式时，config用于保存分片信息

# 使用Java访问MongoDB

## 1、环境配置

- 在Java程序中如果要使用MongoDB，需要确保已经安装了Java环境及MongoDB JDBC 驱动。
- 下载mongo jar包，下载地址：  
<https://repo1.maven.org/maven2/org/mongodb/>，请确保下载最新版本。
- 需要将mongodb-driver.jar, bson.jar, mongodb-driver-core.jar包含在你的 classpath 中



# 使用Java访问MongoDB

## 2、连接数据库

```
MongoClient mongoClient = new MongoClient("localhost",27017);
MongoDatabase db = mongoClient.getDatabase("student");
MongoCollection col=db.getCollection("class1");
FindIterable<Document> findIterable = col.find();
MongoCursor<Document> mongoCursor = findIterable.iterator();
while(mongoCursor.hasNext()) {
    System.out.println(mongoCursor.next());
}

System.out.println("连接成功");
```

## 使用Java访问MongoDB

3、创建集合 可以使用com.mongodb.DB类中的createCollection()来创建集合

```
MongoClient mongoClient = new MongoClient("localhost",27017);  
MongoDatabase db = mongoClient.getDatabase("student");  
MongoCollection col=db.getCollection("class1");  
db.createCollection("test01");  
System.out.println("连接成功");
```

## 使用Java访问MongoDB

- 4、插入文档 使用com.mongodb.client.MongoCollection类的 insertOne() 方法来插入一个文档

```
MongoClient mongoClient = new MongoClient("localhost", 27017);
MongoDatabase db = mongoClient.getDatabase("student");

MongoCollection col = db.getCollection("test011");
Document doc = new Document("title", "MongoDB").append("descri:
    .append("url", "http://www.w3cschool.cc/mongodb/").app

col.insertOne(doc);
System.out.println("连接成功");
```