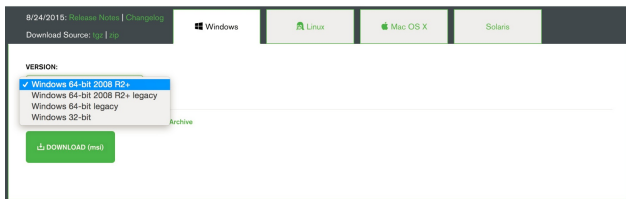


2、mongoDb环境搭建1（单节点）

1、mongo下载

官网：<https://www.mongodb.com/download-center#community>

ps：建议使用64位版本，32位MongoDB数据最大不能超过2GB。



2、mongo安装

直接打开下载的msi文件，按操作提示安装即可。

注意点：

- a) 有些系统当前用户权限不够，可能导致安装失败，请把msi文件移动到c盘，用管理员权限运行安装。
- b) mongo安装目录默认位置：C:\Program Files\MongoDB\Server
- c) 安装后的环境：mongo的服务器程序，自带shell客户端程序，备份程序，恢复程序等。

3、mongo服务器启动

a) cmd进入dos窗口，切换到mongo程序目录

例如：cd C:\Program Files\MongoDB\Server\3.1\bin

b) 运行服务器程序

mongod

注意点：默认情况mongod为启动服务器程序，监听【27017端口】进行服务，并且会把数据库数据存储在目录【C:\data\db】，如果本机没有该目录，会启动失败。所以需要手动建立该目录。

c) 服务器程序启动相关参数：（mongod -h 可查看）

--dbpath 指定数据库存储数据的路径

--port 指定服务端口号，默认端口27017

--replSet（集群：副本集才需要指定）指定副本集（集群）的名字

d) 建立自己服务器批处理文件启动服务器（防止每次输入启动参数-太过麻烦）

1) 新建文件XXX.bat

2) 把服务启动命令放入文件并保存

例如：mongod --dbpath c://ttd --port 27019

3) 打开XXX.bat文件启动服务器

```
2017-09-01T16:06:17.763+0800 I CONTROL [initandlisten] modules: none
2017-09-01T16:06:17.764+0800 I CONTROL [initandlisten] build environment:
2017-09-01T16:06:17.765+0800 I CONTROL [initandlisten] distmod: 2008plus-ssl
2017-09-01T16:06:17.766+0800 I CONTROL [initandlisten] distarch: x86_64
2017-09-01T16:06:17.767+0800 I CONTROL [initandlisten] target_arch: x86_64
2017-09-01T16:06:17.768+0800 I CONTROL [initandlisten] options: { net: { port: 27019
}, storage: { dbPath: "c://ttd" } }
2017-09-01T16:06:17.771+0800 I FTDC [initandlisten] Starting full-time diagnostic
data capture with directory 'c://ttd/diagnostic.data'
2017-09-01T16:06:17.775+0800 I NETWORK [initandlisten] waiting for connections on po
rt 27019
```

4、mongo自带的shell客户端连接服务器

a) cmd进入dos窗口，切换到mongo程序目录

例如：cd C:\Program Files\MongoDB\Server\3.1\bin

b) 运行客户端程序

例如：

mongo 默认连接【本机ip】【27017端口】的服务器，默认操作【test】数据库

mongo -host 指定连接服务器 -port 指定连接服务器的端口

```
C:\Program Files\MongoDB\Server\3.1\bin>mongo -port 27019
MongoDB shell version: 3.1.9
connecting to: 127.0.0.1:27019/test
Server has startup warnings:
2017-09-01T16:06:17.753+0800 I CONTROL [initandlisten]
2017-09-01T16:06:17.754+0800 I CONTROL [initandlisten] ** NOTE: This is a development version (3.1.9) of MongoDB.
2017-09-01T16:06:17.755+0800 I CONTROL [initandlisten] ** Not recommended for production.
2017-09-01T16:06:17.756+0800 I CONTROL [initandlisten]
```

c) 通过shell客户端（加载了js的环境-可以直接写js函数，变量，循环等）访问数据库服务器

常用操作：

show dbs; 显示所有的数据库实例

use dbname; 选择某一个数据库实例，没有也可以选择，当往数据库实例加入数据则创建

db.createCollection(name, options); 在当前的数据库实例里创建集合

db.数据库集合名.insert({x:10}); 在集合里插入文档数据（如果没有_id键，自动生成）

db.数据库集合名.find(); 查询某个集合数据

cls : 清理shell控制台

db.help(); 帮助，显示数据库操作命令

5、mongo和sql数据库概念对比映射

SQL概念	MongoDB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,MongoDB3.2后支持比较弱!
primary key	primary key	主键,MongoDB自动将_id字段设置为主键

6、mongo常见数据类型

Object ID : 文档ID, new ObjectId()

String : 字符串, 最常用, 必须是有效的UTF-8, 单引号或双引号

Boolean : 存储一个布尔值, true或false

Integer : 整数可以是32位或64位, 默认32, new NumberLong()设为64

Double : 存储双精度浮点值, 写小数

Date : 存储当前日期或时间的UNIX时间格式, new Date()

Arrays : 数组或列表, 多个值存储到一个键, []

Object : 用于嵌入式的文档, 即一个值为一个文档, {}

Null : 存储Null值, null

二进制数据 : 此数据类型用于存储二进制数据, shell不支持。

。 。 。

--- 【例子】-----

```
{ "_id": 2, "val": 9.99, "description": "Double" }
```

```
{ "_id": 3, "val": 5, "description": "int" }
```

```
{ "_id": 4, "val": NumberLong(10), "description": "Long" }
```

```
{ "_id": 5, "val": new Date(), "description": "Date" }
```

```
{ "_id": 6, "val": {name:'zxx'}, "description": "Object" }
```

ps :

a) 对应java驱动的所有的类型为org.bson.BsonValue的子类型

b) ObjectId 是具有以下结构的【12字节】BSON类型, 是mongo服务器自动生成的用于作为整个集群唯一标识, 类似于uuid, 理论上是不会重复的。

结构: 【时间: 4字节】 【机器标识: 3字节】 【进程号: 2字节】 【自增记数: 3字节】

用于

x) 插入集合文档时, 如果没有设置_id属性, 那么默认生成_id, 其类型就是ObjectId。

x) ObjectId转换为String: objId.str

c) 检查类型:

值 instanceof 类型

typeof 值