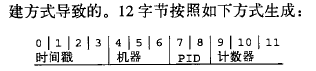
mongodb的自动生成的唯一id是根据：



因此可以通过这个id后的当时的时间戳;

>db.tt.find()

{ "\_id" : ObjectId("58037b045a808d2420e8fa53"), "a" : 1 }

>ObjectId("58037b045a808d2420e8fa53").getTimestamp() //得到建立时的时间戳

ISODate("2016-10-16T13:05:08Z")

[mongoDB排序](http://bbs.csdn.net/topics/380197294)？

Mongodb再插入时如果没有指定id值的话会自动生成，而排序默认可能按照这个排序返回结果，而此值是根据时间戳机器码进程值和随机数决定的，因此对于同一台机器而言（不同的机器插入的数据也应该一样），只是时间的不同而排序又是二进制排序（一个字节一个字节的排序），因此返回的结果还是按照插入时的顺序。

负载均衡：

通过./mongod --dbpath master --port 10000 --master启动一个mongodb实例。  
2、通过./mongod --dbpath slave --port 10001 --slave启动另一个mongodb实例，与上一个实例在同一台服务器上。  
3、在slave的shell中添加了一个source是localhost:10000。  
  
经过以上步骤，slave会自动同步master上的数据。（连接了slave后需要rs.slaveOk()就可以执行查询命令了）

Mongdo数据库快照;

1、数据库快照是数据库（称为“源数据库”）的只读静态视图。在创建时，每个数据库快照在事务上都与源数据库一致。在创建数据库快照时，源数据库通常会有打开的事务。在快照可以使用之前，打开的事务会回滚以使数据库快照在事务上取得一致。

2、内存快照，某一时刻内存数据状态

BSON只支持UTF8编码。

Mongodb查询特殊字符要转义

>db.zhengze.find({"title":/吗\(\)你/i}) //括号要转义

{ "\_id" : ObjectId("5809bee9219b0cb57c5ead2a"), "title" : "你好吗()你好吗" }

Mongodb的数据大小限制：

由于在MongoDB中，1.7版本之前，BSON对象的大小只有4MB的限制，1.7-1.8版本，大小限制是16MB

Mongodb复制一个数据库中的数据到另一个数据库：

没有用户名和密码则为空,没有todb则会新建

语法：db.copyDatabase(fromdb, todb, fromhost:port, username, password)

> db.copyDatabase("ir","ir","localhost:27020","","")

{ "ok" : 1 }

>db.copyDatabase("MoliFresh","MoliFresh","192.168.100.249","")

{ "ok" : 1 }

Mongodump命令帮助：

D:\mongo\bin>mongodump /version

mongodump version: r3.2.9

git version: 4a4e7d30773b28cf66f75e45bc289a5d3ca49ddd

Go version: go1.6.2

os: windows

arch: amd64

compiler: gc

OpenSSL version: OpenSSL 1.0.1p 9 Jul 2015

删除单个文档的一个属性：

> delete d.\_id

>d

{

"\_t" : "StarNode",

"Name" : null,

"Desc" : null,

"ImageUrl" : null,

"PageUrl" : null,

"TimeCreated" :ISODate("0001-01-01T00:00:00Z"),

"TimeModified" :ISODate("2016-10-13T09:11:56.634Z"),

"Status" : 0,

"Deleted" : 0,

"Locked" : 1,

"OriginalId" : null,

"NodeType" : 0,

"Relations" : [ ],

"Sex" : null,

"Country" : null,

"Birthday" : null,

"ForeignName" : null,

"OtherNames" : null,

"Astrology" : null,

"Occupation" : null

}

mongodb的mapreduce学习

参考：<http://www.cnblogs.com/loogn/archive/2012/02/09/2344054.html>

MapReduce应该算是MongoDB操作中比较复杂的了，自己开始理解的时候还是动了动脑子的，所以记录在此！

命令语法：[详细看](http://www.mongodb.org/display/DOCS/MapReduce)

|  |
| --- |
| db.runCommand(   { mapreduce : 字符串，集合名,     map : 函数,见下文     reduce : 函数，见下文     [, query : 文档，发往map函数前先给过渡文档]     [, sort : 文档，发往map函数前先给文档排序]     [, limit : 整数，发往map函数的文档数量上限]     [, out : 字符串，统计结果保存的集合]     [, keeptemp: 布尔值，链接关闭时临时结果集合是否保存]     [, finalize : 函数，将reduce的结果送给这个函数，做最后的处理]     [, scope : 文档,js代码中要用到的变量]     [, jsMode : 布尔值，是否减少执行过程中BSON和JS的转换，默认true] //注：false时 BSON-->JS-->map-->BSON-->JS-->reduce-->BSON,可处理非常大的mapreduce,<br>　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　//true时BSON-->js-->map-->reduce-->BSON     [, verbose : 布尔值，是否产生更加详细的服务器日志，默认true]   }  ); |