**MongoDB的Replica Sets+Sharding架构**

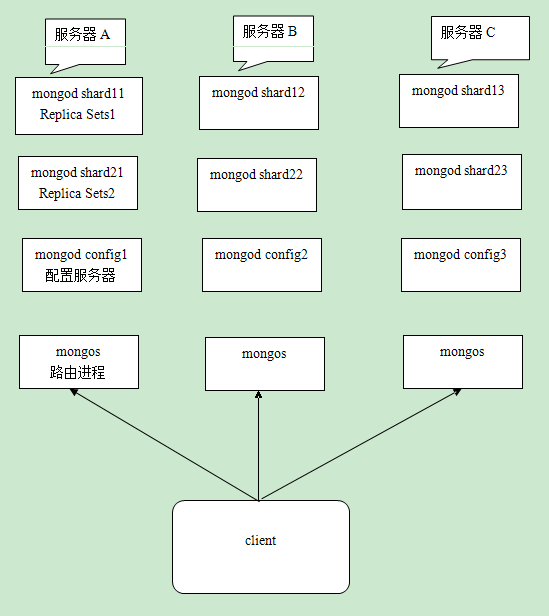
MongoDB的Sharding机制解决了海量存储和动态扩容的问题，但离生产环境的高可靠，高可用还有距 离，Sharding在单点出现故障时就无能为力了。但是MongoDB的副本集却可以很轻松的处理单点故障，所以就有了Replica Sets + Sharding的高可用，高安全的架构。

**Replica Sets+Sharding架构如下：**

1，shard服务器：使用Replica Sets确保每个数据节点都具有备份、自动容错转移、自动恢复的能力。

2，配置服务器：使用使用3个配置服务器确保元数据完整性

3，路由进程：使用3个路由进程实现平衡，提高客户端接入性能，架构如下



3个分片进程：shard11，shard12，shard13组成一个副本集，提供Sharding中shard1的功能。

3个分片进程：shard21，shard22，shard23组成一个副本集，提供Sharding中shard2的功能。

3个配置服务器进程和3个路由器进程

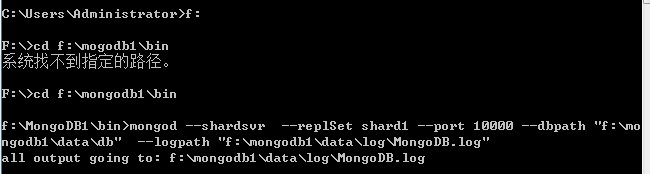
---------------------------------------------------------------------

现在我们开始搭建整个架构(因为没那么多机器，我还是用本地的目录来模拟机器)

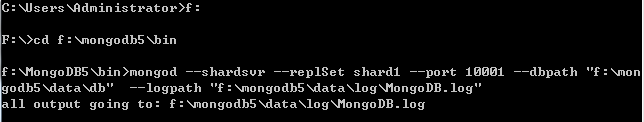
|  |  |  |
| --- | --- | --- |
| 主机 | ip | 服务及端口 |
| ServerA  mongodb1  mongodb2  mongodb3  mongodb4 | 127.0.0.1 | mongod shard11：10000  mongod shard21：20000  mongod config 1：30000  mongos ：40000 |
| ServerB  mongodb5  mongodb6  mongodb7  mongodb4 | 127.0.0.1 | mongod shard12：10001  mongod shard22：20001  mongod config 2：30001  mongos ：40000 |
| ServerC  mongodb9  mongodb8  mongodb11  mongodb4 | 127.0.0.1 | mongod shard13：10002  mongod shard23：20002  mongod config 3：30002  mongos ：40000 |

1，启动Shard1进程并配置Replica Sets

启动mongod shard11进程，副本集名称：shard1



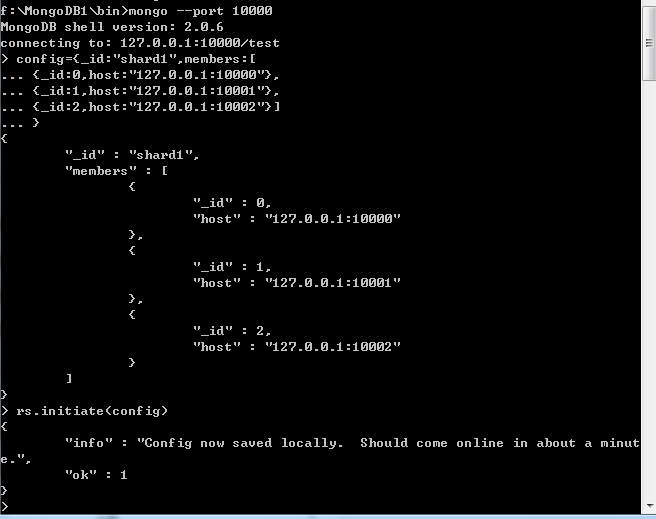
启动mongod shard12进程，并设置副本集：shard1



启动mongod shard13进程，并设置副本集：shard1



把这三个进程配置成副本集，新开一个cmd,用来执行各种非启动命令，连接到上面三个进程中的任何一个，把他们配置成副本集，操作如下

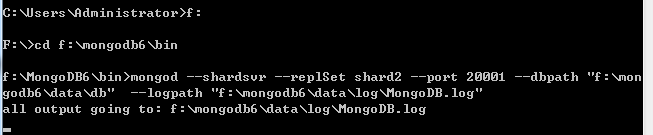


2，启动Shard2进程并配置Replica Sets

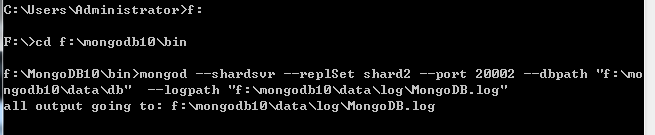
启动mongod shard21进程，副本集名称：shard2



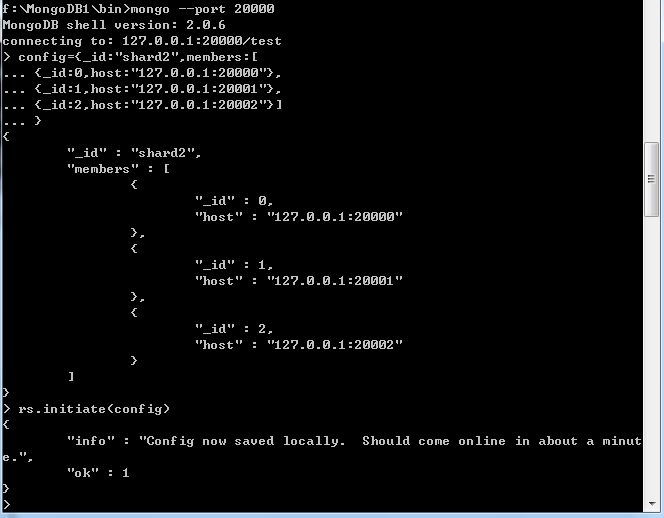
启动mongod shard22进程，并设置副本集：shard2



启动mongod shard23进程，并设置副本集：shard2



把这三个进程配置成副本集，操作如下

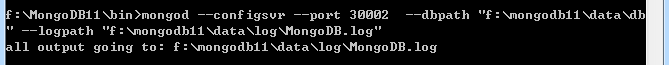


到此两个副本集够成的分片已经配置完成，下面配置Config server和Route process

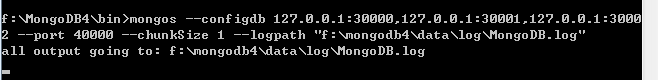
3，配置3个Config Server





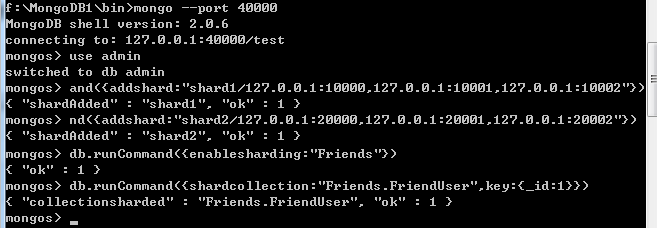


4配置Route Process



chunk大小为1M，方便我们测试效果。

5配置分片的表和片键



我用的还是Friends库中的FriendUser表来做分片，片键是\_id,因为cmd宽度太小了添加分片的命令显示不完全，我手动把他们列出来

添加分片

db.runCommand({addshard:"shard1/127.0.0.1:10000,127.0.0.1:10001,127.0.0.1:10002"})

db.runCommand({addshard:"shard2/127.0.0.1:20000,127.0.0.1:20001,127.0.0.1:20002"})

到此整个构架已经配置完成了，我们来验证下配置的情况，我通过客户端添加10000条数据到数据库中



可以看到分片已经执行。

---------------------------------------------------------------------

现在做下容灾的测试，我停掉shard1-1，看看结果会如何。

打开shard11的 cmd窗口，Ctrl+C停止进程

查看下状态



状态完好，我在插入20000条数据，看看效果



可以看到依然可以运行。

这里会出现这种情况：当有三台机器做副本集的时候，只能是一台服务器当掉，当有两台当掉的时候，第三台不能由从库变为主库。