

MongoDB 复制（副本集）

MongoDB复制是将数据同步在多个服务器的过程。

复制提供了数据的冗余备份，并在多个服务器上存储数据副本，提高了数据的可用性， 并可以保证数据的安全性。

复制还允许您从硬件故障和服务中断中恢复数据。

什么是复制？

- 保障数据的安全性
- 数据高可用性 (24*7)
- 灾难恢复
- 无需停机维护（如备份，重建索引，压缩）
- 分布式读取数据

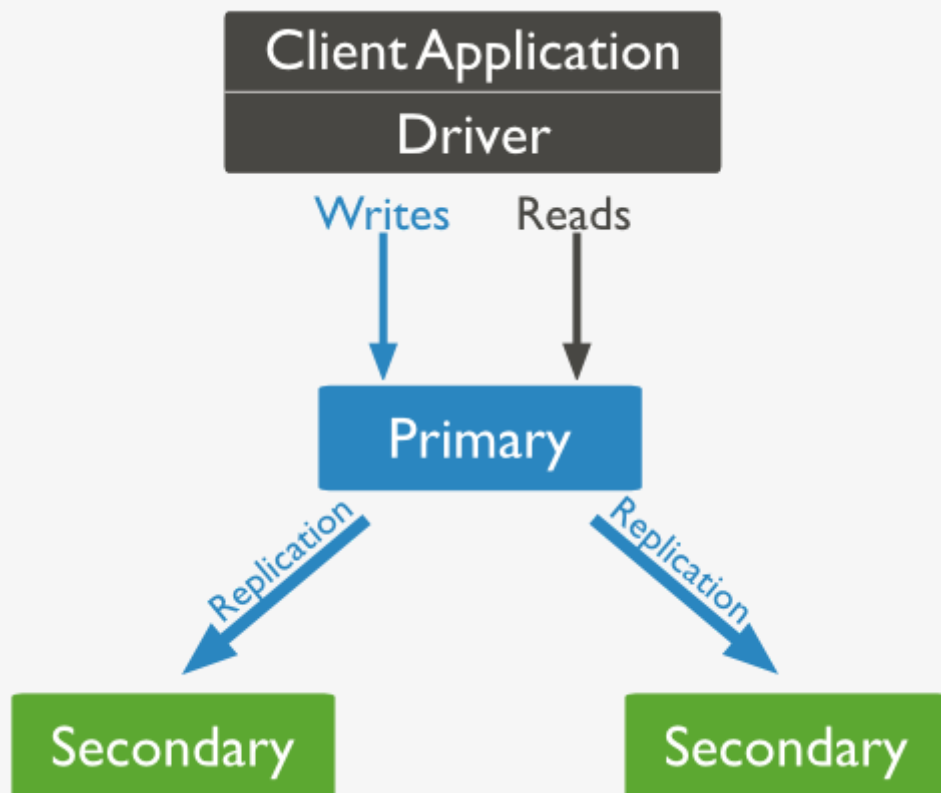
MongoDB复制原理

mongodb的复制至少需要两个节点。其中一个为主节点，负责处理客户端请求，其余的都是从节点，负责复制主节点上的数据。

mongodb各个节点常见的搭配方式为：一主一从、一主多从。

主节点记录在其上的所有操作Oplog，从节点定期轮询主节点获取这些操作，然后对自己的数据副本执行这些操作，从而保证从节点的数据与主节点一致。

MongoDB复制结构图如下所示：



以上结构图中，客户端从主节点读取数据，在客户端写入数据到主节点时，主节点与从节点进行数据交互保障数据的一致性。

副本集特征：

- N 个节点的集群
- 任何节点可作为主节点
- 所有写入操作都在主节点上
- 自动故障转移
- 自动恢复

MongoDB副本集设置

在本教程中我们使用同一个MongoDB来做MongoDB主从的实验，操作步骤如下：

1、关闭正在运行的MongoDB服务器。

现在我们通过指定 `--replSet` 选项来启动mongodb。`--replSet` 基本语法格式如下：

```
mongod --port "PORT" --dbpath "YOUR_DB_DATA_PATH" --replSet "REPLICA_SET_INSTANCE_NAME"
```

实例

```
mongod --port 27017 --dbpath "D:\set up\mongodb\data" --replSet rs0
```

以上实例会启动一个名为rs0的MongoDB实例，其端口号为27017。

启动后打开命令提示框并连接上mongoDB服务。

在Mongo客户端使用命令rs.initiate()来启动一个新的副本集。

我们可以使用rs.conf()来查看副本集的配置

查看副本集状态使用 rs.status() 命令

副本集添加成员

添加副本集的成员，我们需要使用多台服务器来启动mongo服务。进入Mongo客户端，并使用rs.add()方法来添加副本集的成员。

语法

rs.add() 命令基本语法格式如下：

```
>rs.add(HOST_NAME:PORT)
```

实例

假设你已经启动了一个名为mongod1.net，端口号为27017的Mongo服务。 在客户端命令窗口使用rs.add() 命令将其添加到副本集中，命令如下所示：

```
>rs.add("mongod1.net:27017")  
>
```

MongoDB中你只能通过主节点将Mongo服务添加到副本集中，判断当前运行的Mongo服务是否为主节点可以使用命令db.isMaster()。

MongoDB的副本集与我们常见的主从有所不同，主从在主机宕机后所有服务将停止，而副本集在主机宕机后，副本会接管主节点成为主节点，不会出现宕机的情况。

← MongoDB 聚合

MongoDB 分片 →

 点我分享笔记