Redis主从复制

Redis主从复制的原理。

一个主服务器（Master）,一个或者多个从服务器（Slave），当Master服务器内存中的数据发生改变时，会执行写命令更新到从服务器的内存中，保证数据一致性。

作用：可以有效分担高并发环境下单机服务器的压力，只需要制定一定的策略让不同的用户去请求不同IP的服务器即可。因为Redis主从集群一般为内网环境。通过专线连接以后又可以突破带宽带来的数据同步障碍。

配置：

打开Redis安装目录，在端口配置行下面加上以下配置。

Slaveof host port

当重新启动slave服务器的时候，会自动连接到Master并且同步数据，为了数据一致性，默认Slave服务器没有write权限。

也可以在redis-cli连接上服务器以后设置slaveof host port，但是这个命令只会在内存中生效，当重启Redis以后依然加载的是Redis.conf配置文件中的设置。

**Ps: slaveof no one命令可以终止主从复制过程。**

步骤 主服务器操作 从服务器操作

1. Master等待命令进入 slave连接(或者重连接)主服务器，发送SYNC命令
2. master开始执行BGSAVE，并使用缓冲区记录BGSAVE之后执行的所有写命令 slave根据配置选项来决定时继续使用现在的数据来处理客户端命令，还是向发送请求的客户端返回错误

3 Master BGSAVE执行完毕，向从服务器发送快照文件，并在发送期间继续使用缓冲区记录被执行的写命令 Slave丢弃所有旧的数据，开始载入主服务器发来的快照文件

4 Master快照文件发送完毕，开始向从服务器发送存储在缓冲区里面的写命令 Slave完成对快照文件的解释操作，像往常一样开始接受命令请求

5 Master缓冲区存储的写命令发送完毕；从现在开始，每执行一个写命令，就像从服务器发送相同的写命令 Slave执行主服务器发来的所有存储在缓冲区里面的写命令；从现在开始，接收并执行主服务器传来的每个写命令

问题：当多个从服务器同时连接主服务器时，会出现以下问题。

1. 如果步骤3尚未执行，所有的从服务器会接收到相同的快照文件和缓冲区写命令。
2. 步骤3已经开始执行或者执行完毕，主服务器首先会与连接时间较早的从服务器执行完1~5步。然后再与新连接执行1~5步骤。

当出现多个从服务器连接时，对网络的开销非常大，很容易宕机。解决方案：可以根据业务并发量大小设计如下方案。这样，就把12台服务器的同步简化成了 1带3的架构。二层即做slave又做第三层的Master。系统压力均匀分担到了一二层四台服务器上。

