一、遇到的坑

一个月前，我们在测试环境部署了一套 MySQL 高可用架构，也就是 MySQL 双主 + Keepalived 的模式。

在这一个月遇到了很多坑：

- 因为两个 MySQL 节点都可以写入，极其容易造成主键重复，进而导致主从同步失败。
- 同步失败后，Slave_SQL_Thread 线程就停了，除非解决了同步的错误，才能继续进行同步。
- 同步失败的错误，不会只有一条记录有问题，往往是一大片的同步问题。
- 两个节点互相缺少对方的数据。
- 主从的同步延迟，切换到新主库后，数据不是最新。
- 当出现不一致时，无法确定以哪个库为准。

造成上面问题的主要原因就是因为两个节点都支持写入 + 双主可以随时切换。

解决这种问题的方案有 改进自增主键的步长（影响未评估），使用 GTID 方案（未验证）。即使这样，双主同步的风险还是有，而且不同步后，如何处理是个大难题。

那么回到我们最初的想法：为什么会选择双主?

最开始的目的就是为了高可用。双主就是说有一台 MySQL 节点挂了，另外一台能够顶上，对于用户来说是无感的，给运维人员一定的缓冲时间来排查 MySQL 故障。另外老的主节点恢复后，不用改配置就能立即成为从节点。

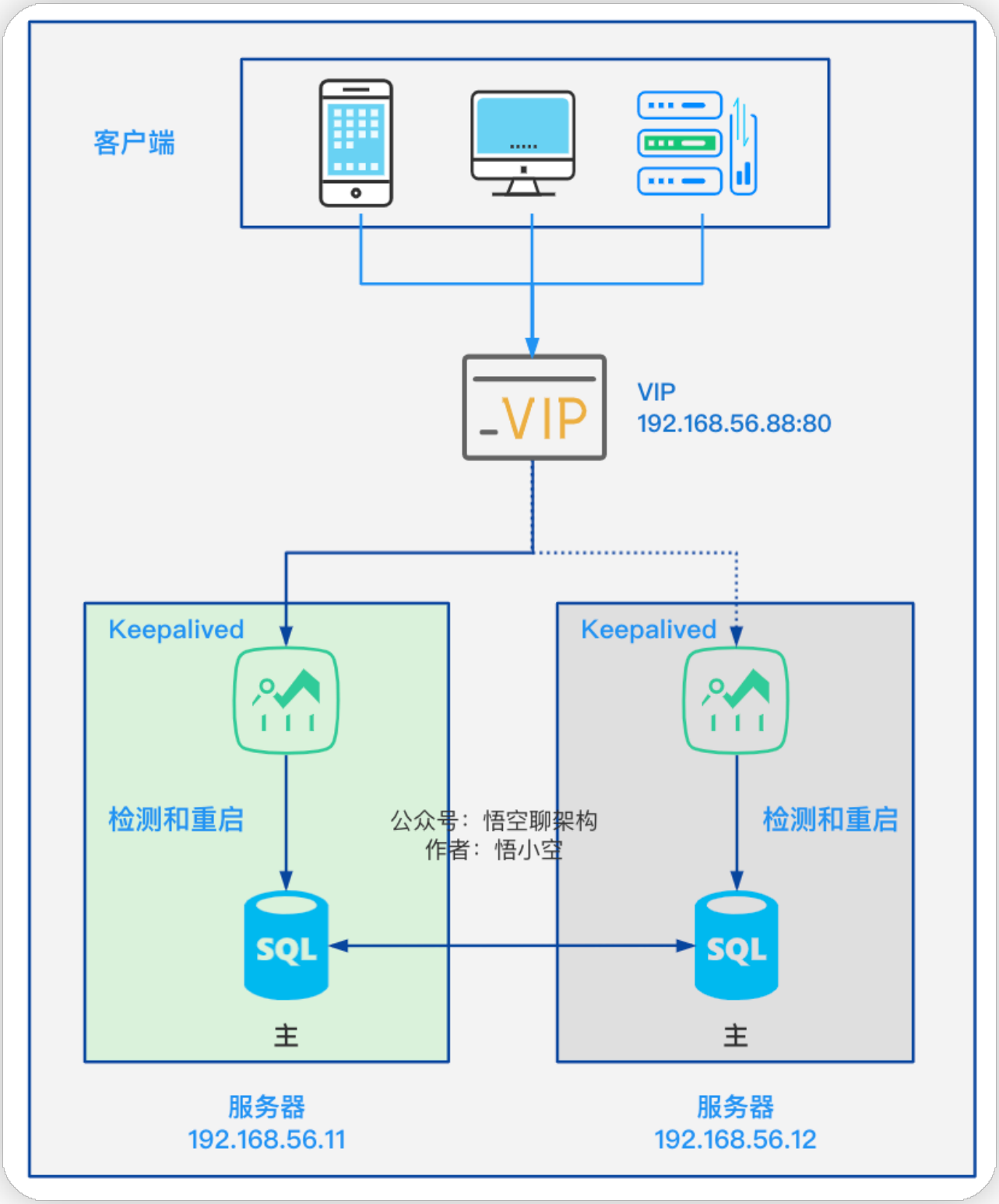
经过这一个月的 MySQL 双主模式的试运行，最后我们还是决定切换到 MySQL 主 - 从模式。

双主模式就是两个节点即是主节点也是从节点，那我们现在切换到一主一从模式，就可以认为是降级。接下来我们聊聊双主换成主从的思路和步骤。

二、双主降为主从

双主模式

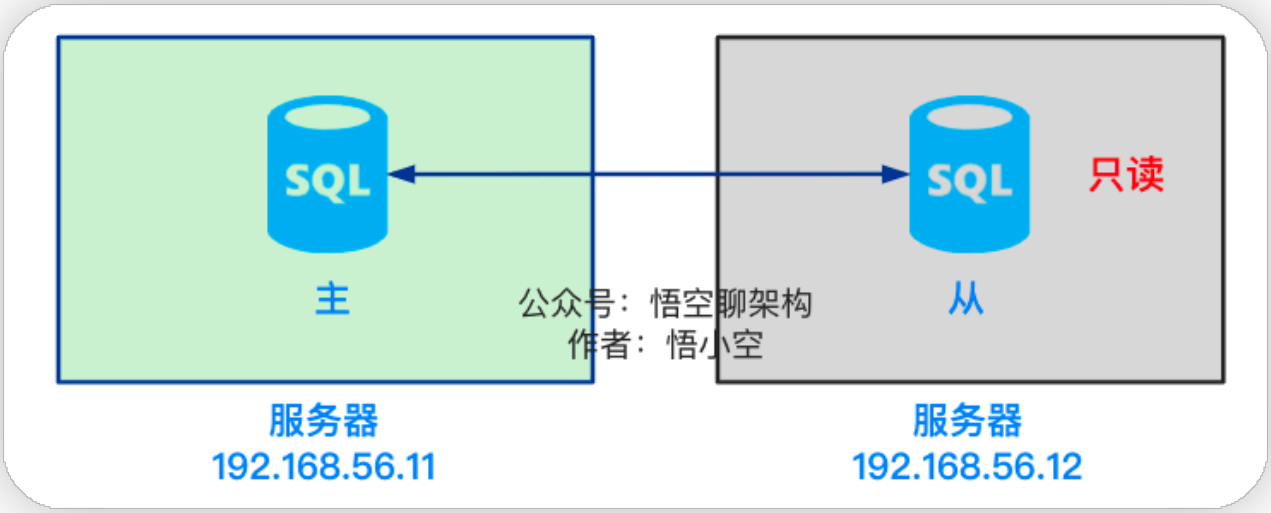
双主模式的原理图如下:



两个主节点，都安装了 KeepAlived 高可用组件，对外提供了一个 VIP，只有一个节点接管 VIP，客户端访问的请求都是到这个 VIP，另外一个节点处于待机状态。

主从模式

和双主不一样的地方如下，从节点是只读的。



一主一从是主从模式中的一种，具有以下特点：

- 一个主节点，一个从节点，主节点提供给客户端访问，从节点只通过主节点的 binlog 进行数据同步。
- 从节点是只读的。从节点可以作为只读节点提供类似报表查询等耗时读操作。
- 主节点宕机后，从节点成为主节点，也是高可用的一种方案。

相对于双主的高可用方案，不同之处如下：

- 主从切换需要用脚本将从库设置为可读可写。
- 主从切换后，需要将从库设置为不同步老主库。
- 主从切换后，老的主库恢复后，需要人工设置为只读，且开启同步新主库的功能。

这样来看，主从模式在异常情况下，多了些人工操作。

在异常情况下，主从切换一般是这样处理的：通过脚本监测主节点是否宕机，如果主库宕机了，则从库自动切换为新的主库，待老主库恢复后，就作为从库同步新主库数据，新主库上的 Keepalived 接管 VIP。

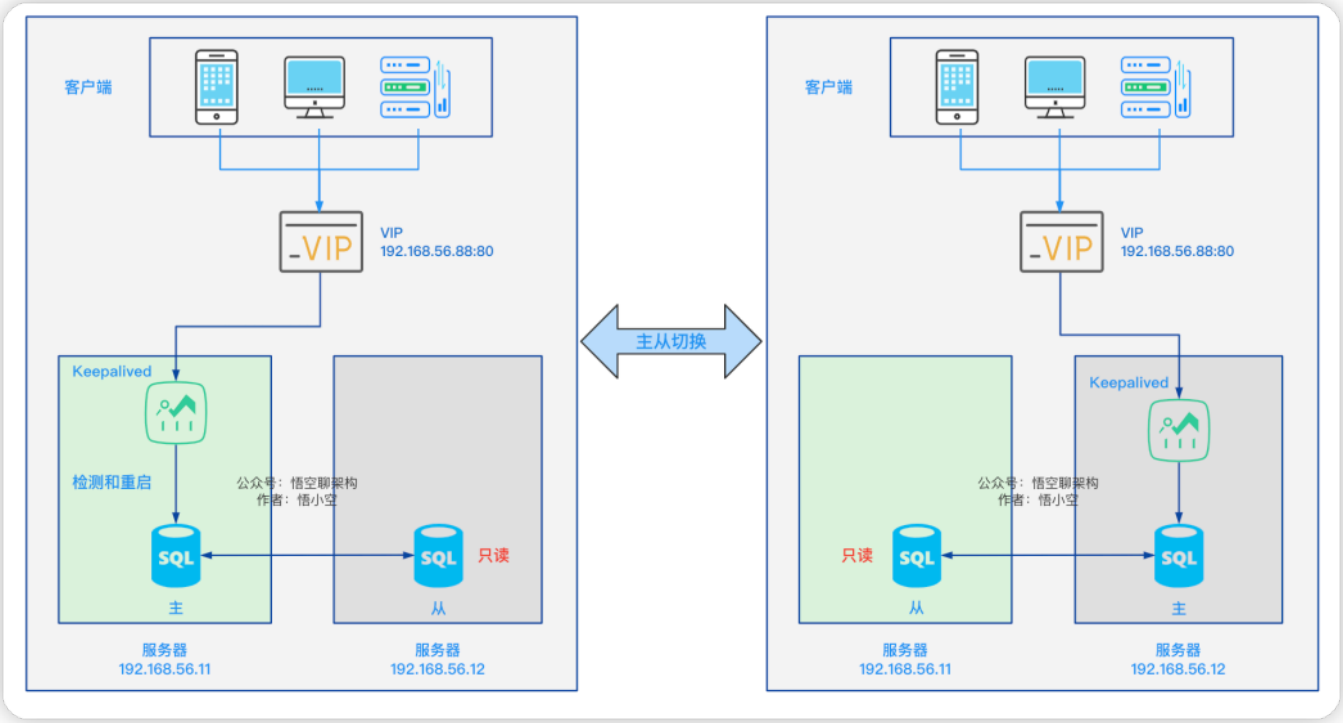
目前改为主从模式有两种方式：

- 简单方式：人工切换模式，主节点故障后需要人工切换主从。
- 复杂方式：高可用方式，主节点故障后，主从自动切换，读写分离自动切换。

本篇只涉及简单方式，复杂方式的原理和配置步骤放到下篇专门讲解。

三、改为主从的简单方式

简单方式的主从切换流程如下：



和双主模式的主从切换的区别是，从节点是只读的，Keepalived 没有启动，需要人工操作主从切换和启动 Keepalived。

修改配置的步骤如下：

① 为了避免从节点上的 Keepalived 自动接管 VIP 的情况出现，将从节点的 Keepalived 停止，如果遇到主节点故障，则需要人工干预来进行主从切换。从节点切换为主节点后，重新启动从节点 Keepalived。

```
systemctl status keepalived
```

② 保留主节点的 Keepalived，保证 MySQL 的连接信息都不需要变。

③ 主节点 node1 停用 MySQL 的同步线程。

```
STOP SLAVE
```

④ 从节点 node2 设置 MySQL 为只读模式。

```
# 修改 my.cnf 文件read_only = 1
```

⑤ 移除主节点 node1 同步 node2 MySQL 的权限。

⑥ 从节点 node1 的开机启动项中移除 keepalived 服务自启动。

```
# 修改启动项配置sudo vim /etc/rc.local# 移除以下脚本systemctl start keepalived
```

四、总结

双主高可用的坑确实比较多，没有 MySQL 的硬核知识真的很难搞定。笔者在这一个月的实践中，深刻体会到了双主同步的难点所在，最后还是选择了一主一从的模式。

另外因为最开始的配置都是双主模式下的，所以要修改一些配置，来改为主从模式。因项目时间比较紧，目前采取的是非高可用的主从模式。