数据库高可用架构了解一下

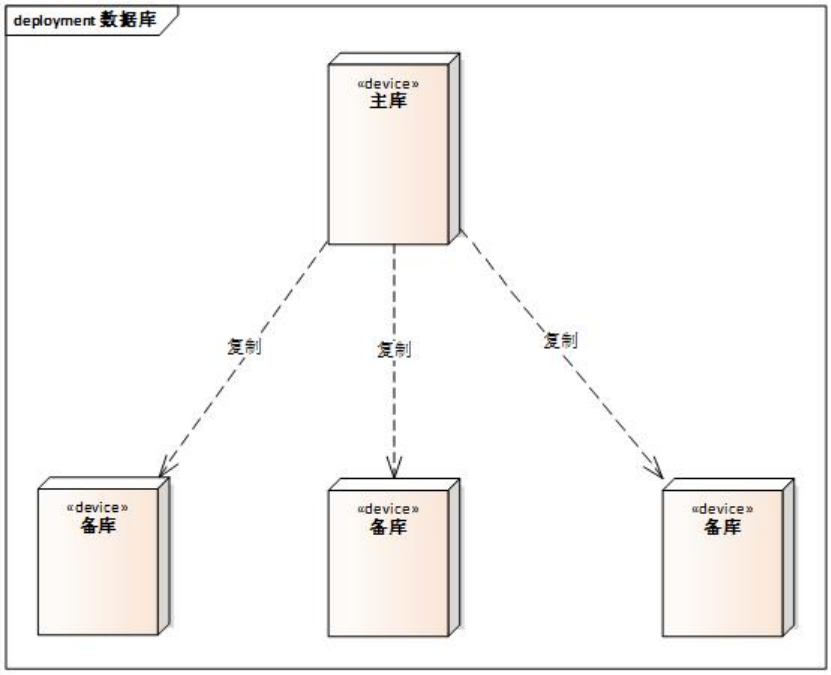
作者：jianzh5
原文链接：https://ld246.com/article/1574131661536
来源网站：链滴
许可协议：署名-相同方式共享 4.0 国际 (CC BY-SA 4.0)

看多了应用服务的高可用架构，我们来看看数据库的高可用吧。

数据存储高可用的方案本质都是通过将数据复制到多个存储设备，通过数据冗余的方式来实现高可用常见的高可用架构有主备、主从、主主、集群、分区等，接下来我们聊聊每种架构的优缺点。

主备架构

1、基本架构拓扑图如下



整体架构简单，几乎所有的数据库都提供了主备复制的功能，例如Mysql、Oracle、MongoDB等。这种架构中备库主要承担数据备份的作用，不参与实际业务读写操作，如果把备机改成主机需要人工作。

原文链接：数据库高可用架构了解一下

2、优缺点分析

主备架构的优点就是简单，具体表现有：

- 对于客户端来说，不需要感知备机的存在，即使灾难恢复后，原来的备机被人工干预修改为主机，户端只需要简单修改连接地址即可，应用架构不需要做任何改动；
- 主机和备机只需要进行数据复制，不需要进行状态判断和主备切换这类复杂操作。

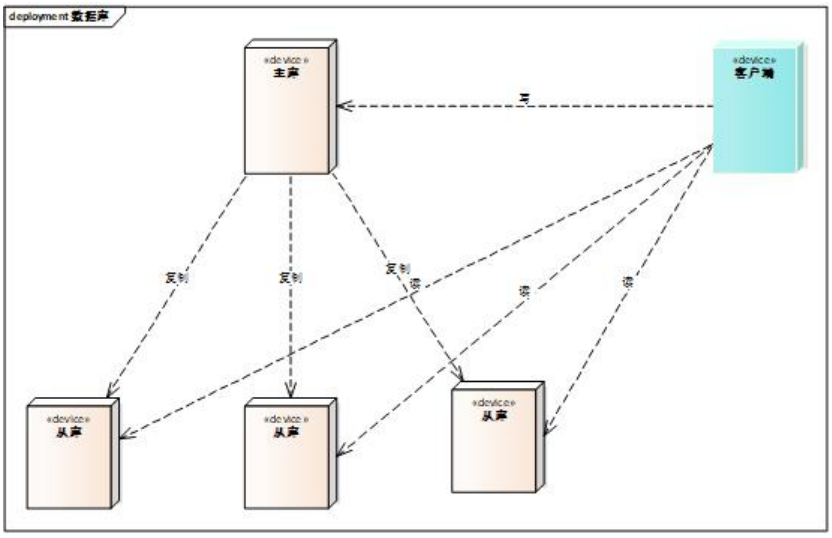
这种架构的缺点也比较明显：

- 备机主要是用于数据备份，如果应用架构没有读写分离设计时会造成成本浪费
- 故障后需要人工干预，无法自动恢复，而人工处理效率又比较低，恢复过程也容易出错。

主从架构

主从架构与主备架构只有一字之差，但是对于实际应用架构差距却很大。在主备架构中备库不参与业操作，而在主从架构中从库是需要参与业务操作的，应用架构需要做读写分离，将写操作写入主库，读操作从从库读。

1、主从基本架构拓扑图如下



2、优缺点分析

这种架构在少量写和大量读时非常有用。可以把读分摊到多个备库上，减少主库的压力，直到从库给库造成了太大的负担，或者主从之间的带宽成为瓶颈为止。

相比于主备架构，它有如下优点：

- 在主库故障时，读操作相关业务可以继续运行
- 从库对外提供读能力，发挥了硬件的性能
- 可以为不同的角色提供不同的从库

缺点：

- 主从架构中从库需要提供读业务，如果主从复制延迟大，数据会出现不一致情况；
- 应用架构需要做修改，一般会加入读写分离，复杂度比主备高；
- 故障后需要人工干预，无法自动恢复，而人工处理效率又比较低，恢复过程也容易出错。

主从切换

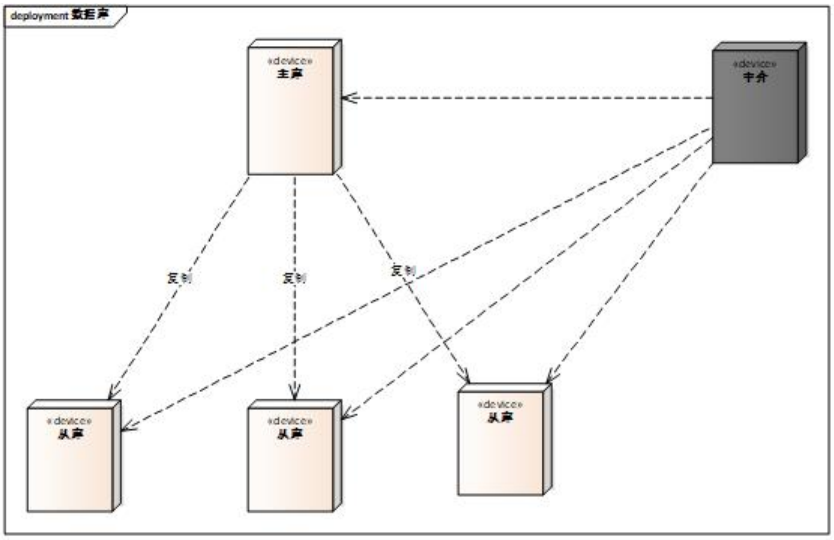
上面两种架构都存在两个共同问题：

- 主库故障后，无法进行写操作
- 主库出了问题后需要人工干预才能将从库切换到主库，而人工切换又可能出现不及时或者切换故障问题。

基于以上两个问题我们需要一个能自动切换的架构，当主库出了故障后能自动将从库切换成主库，无运维人员干预。

要实现主从切换架构必须要考虑一个关键点：必须要有一个机制能监测到数据库节点的运行状态，以来决定是否切换。

这种架构我们一般会引入一个第三方中介，数据库节点定时向第三方中介汇报自己的状态信息；或者三方中介定时去数据库节点拉取数据库状态；



优点：

- 解决了人工干预的问题，大大减少了故障时间，一定程度上保护了运维人员的人生安全

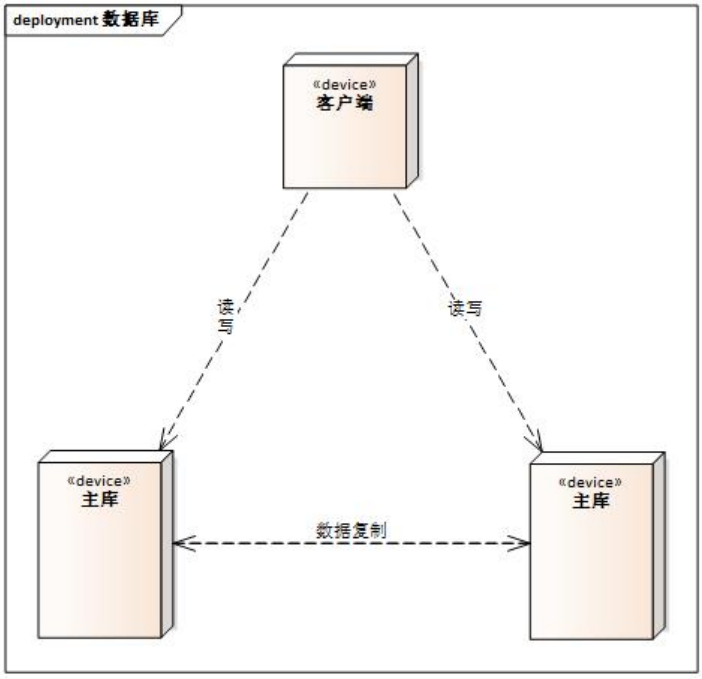
缺点：

- 架构复杂，引入了第三方中介后又需要保证第三方中介的高可用。

这里推荐大家了解一下mysql的MHA架构，或者使用ZK、Keepalived自己搭建主从切换架构。

主主架构

主主架构又叫主主复制，两台数据库都是主库，互相将数据复制给对方，客户端可以挑选任意一台数库进行读写操作。



相比于主从切换，主主架构有如下优点：

- 两台数据库都是主库，不存在切换的概念
- 客户端无需区分不同角色的主机，随便将读写操作发给哪台数据库。
- 架构简单

但是允许向两台主数据库写入是一件很危险的事：

- AB两台数据库采用自增长主键，A库插入用户后id是1，B库插入用户后id也是1，数据冲突
- 同时对数据库数据进行更新会出现大问题，加入AB库的表 tb都有1个字段col，数值为1。如A库行 update tb set col = col +1,B库执行update tb set col = col * 2,最终执行完一台数据的值变成了，另一台数据库的值变成了3，而且没有任何复制错误，一旦出了问题需要好久才能定位。

所以主主架构必须要保证数据能够双向复制，对数据的设计有严格的要求，一般适用于那些临时性，丢失、可覆盖的数据场景。

原文链接：数据库高可用架构了解一下