

课程内容

- 主从架构详解
- 高可用架构分析
- 分库分表场景分析
- 表拆分方式

主从架构

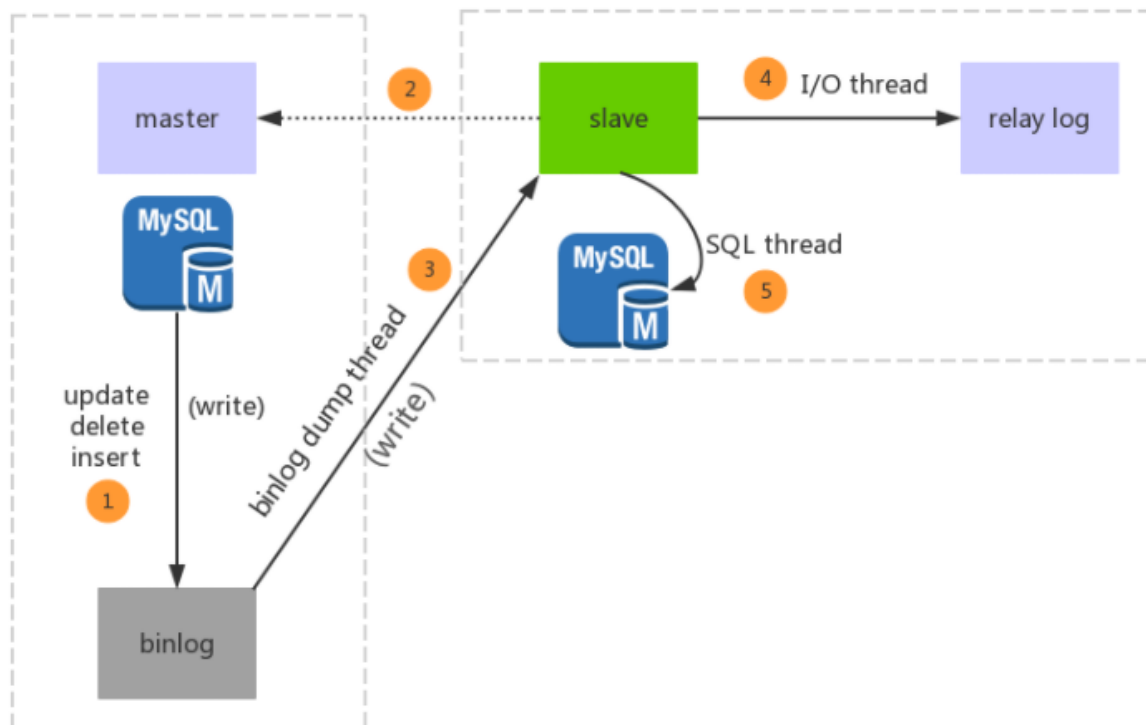
为什么要主从架构？

- 1.如果主服务器出现问题，可以快速切换到从服务器提供的服务
- 2.可以在从服务器上执行查询操作，降低主服务器的访问压力
- 3.可以在从服务器上执行备份，以避免备份期间影响主服务器的服务

主从方案

(课前思考)

主从复制原理

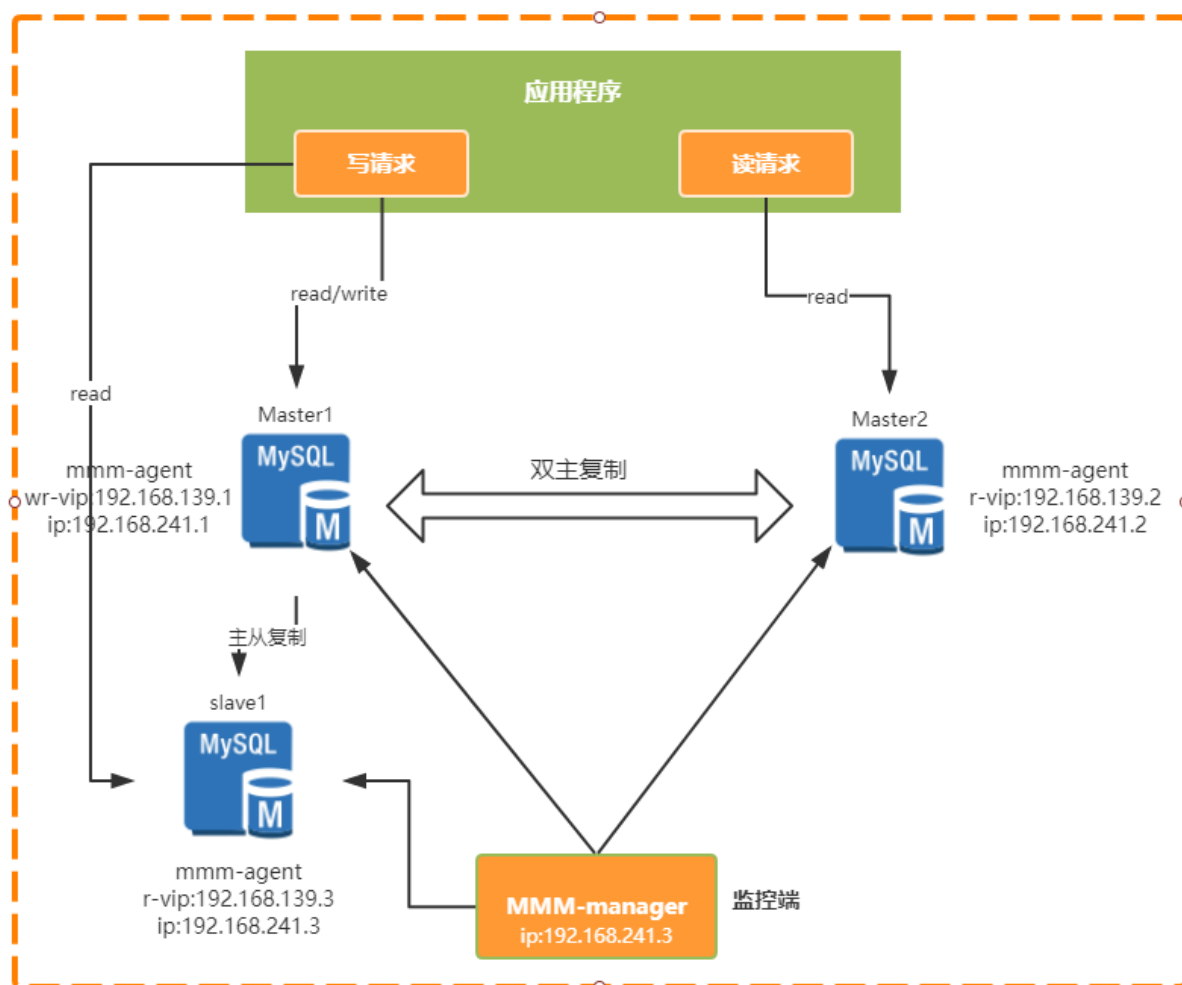


高可用架构方案总结

一、MMM高可用方案

MMM(Master-Master replication managerfor Mysql, Mysql主主复制管理器)是一套灵活脚程序，基于perl实现，用来对mysql replication进行监控和故障迁移，并能管理mysql Master-Master复制的配置(同一时间只有一个节点是可写的)

MMM高可用方案



1.1 优点

(1) 高可用性，扩展性好，出现故障自动转移，对于主主同步，在同一时间只提供一台数据库写操作，保证数据的一致性。

(2) 配置简单，容易操作。

1.2 缺点

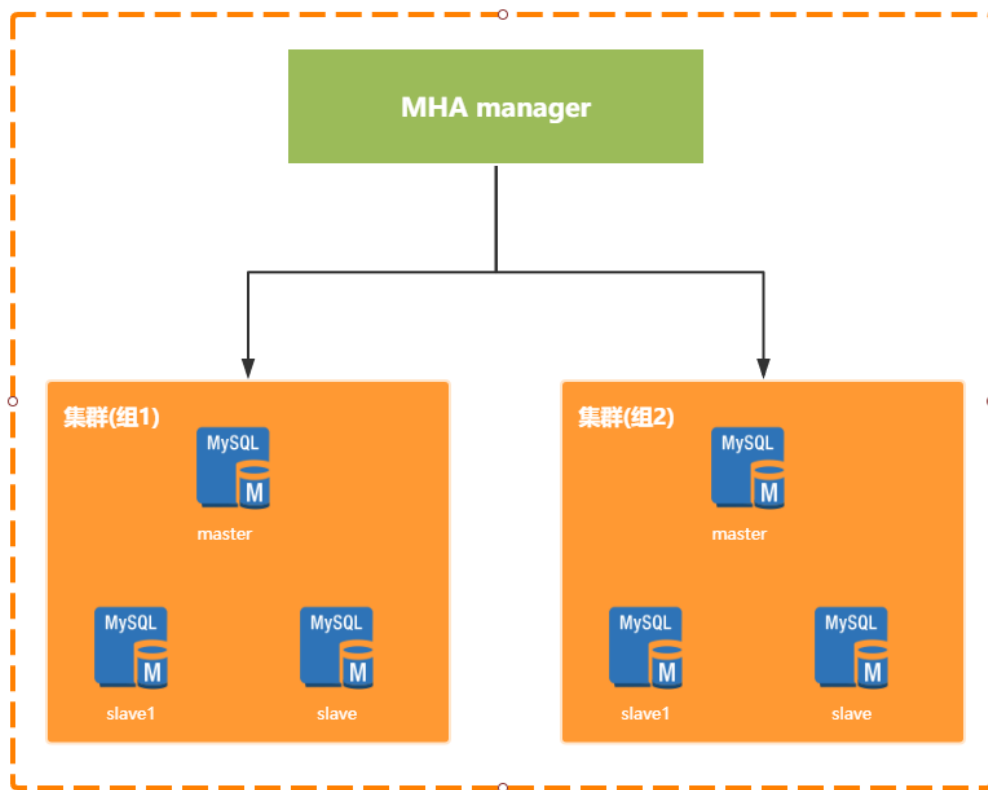
(1) 需要一台备份服务器，浪费资源

(2) 需要多个虚拟IP

(3) agent可能意外终止，引起裂脑。

二、MHA方案

MHA服务，有两种角色，MHA Manager(管理节点)和MHA Node(数据节点)。在MySQL故障切换过程中，MHA能做到在0~30秒之内自动完成数据库的故障切换操作，目前MHA主要支持一主多从的架构，要搭建MHA,要求一个复制集群中必须最少有三台数据库服务器。



2.1 优点

- (1) 不需要备份服务器
- (2) 不改变现有环境
- (3) 操作非常简单
- (4) 可以进行日志的差异修复
- (5) 可以将任意slave提升为master

2.2 缺点

- (1) 需要全部节点做ssh密钥
- (2) MHA出现故障后配置文件会被修改，如果再次故障转移需要重新修改配置文件。
- (3) 自带的脚本还需要进一步补充完善，且用perl开发，二次开发困难。

分库分表

什么是分库分表？

将一个表拆分多张表(库内分表与分库分表)

为什么需要分库分表？

(课前思考)

有哪些拆分方式？

(课前思考)

拆分后产生的问题

(课前思考)

分库分表组件

- shardingsphere (京东数科)
- Mycat (阿里巴巴)
- Tddl
- Atlas (奇虎360)
- 还有哪些你熟知的？