

The top corners of the slide feature decorative geometric shapes, including a network of nodes and lines in the top-left and top-right, and a large white triangle pointing upwards in the center.

**Gdevops**

# 全球敏捷运维峰会

The bottom corners of the slide feature decorative geometric shapes, including a network of nodes and lines in the bottom-left and bottom-right, and a large white triangle pointing upwards in the center.

**腾讯云CDB(for MySQL)  
在高可用的实践**

演讲人：李志阳

# 背景

## ❑ 交易DB的特点：

- ✓ 订单领域关系比较复杂，且需要一定事务操作，因此存储选型为MySQL
- ✓ 订单数据一致性要求高、需要具备跨园区容灾能力

## ❑ 痛点：

- ✓ 采取异步复制，即使在对账的情况下，仍然会有数据不一致性的风险
- ✓ 实例部署在同个园区，园区故障后不可用，影响较大

# 解决方案

## ❑ eCDB ( 增强型CDB )

✓ 打造针对交易型业务的高可用DB平台

## ❑ 目标

✓ 1、强同步 : 减少性能损耗, 支持多MySQL分支

✓ 2、跨园区HA: 园区故障自动切换



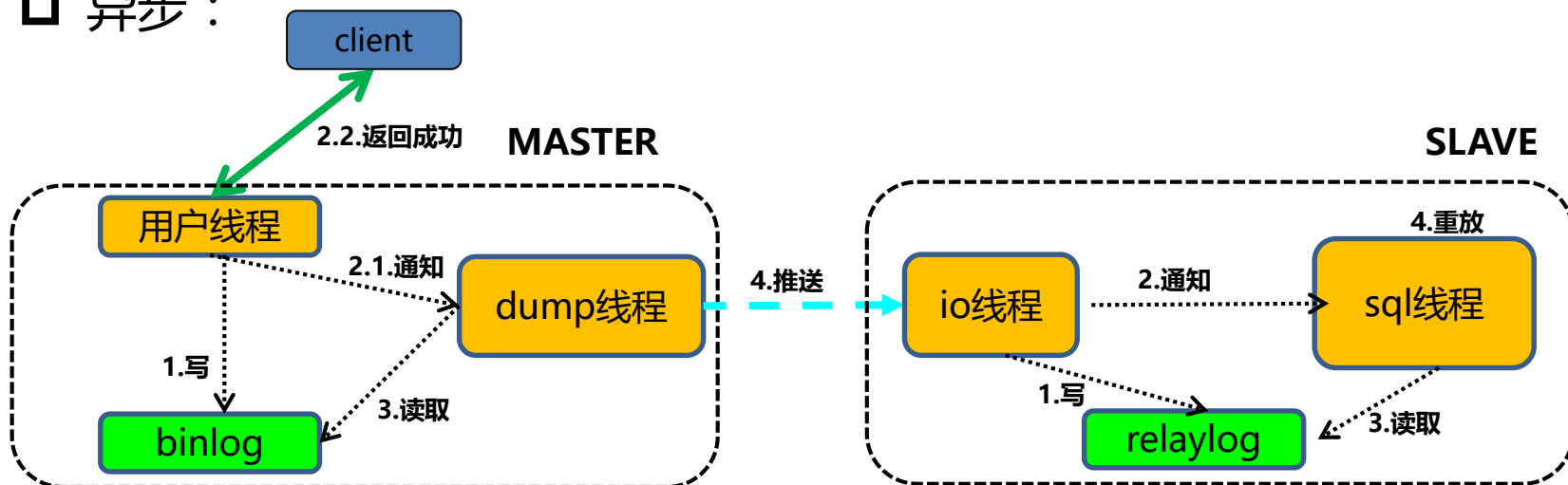
# eCDB高可用特性(1)

## 强同步

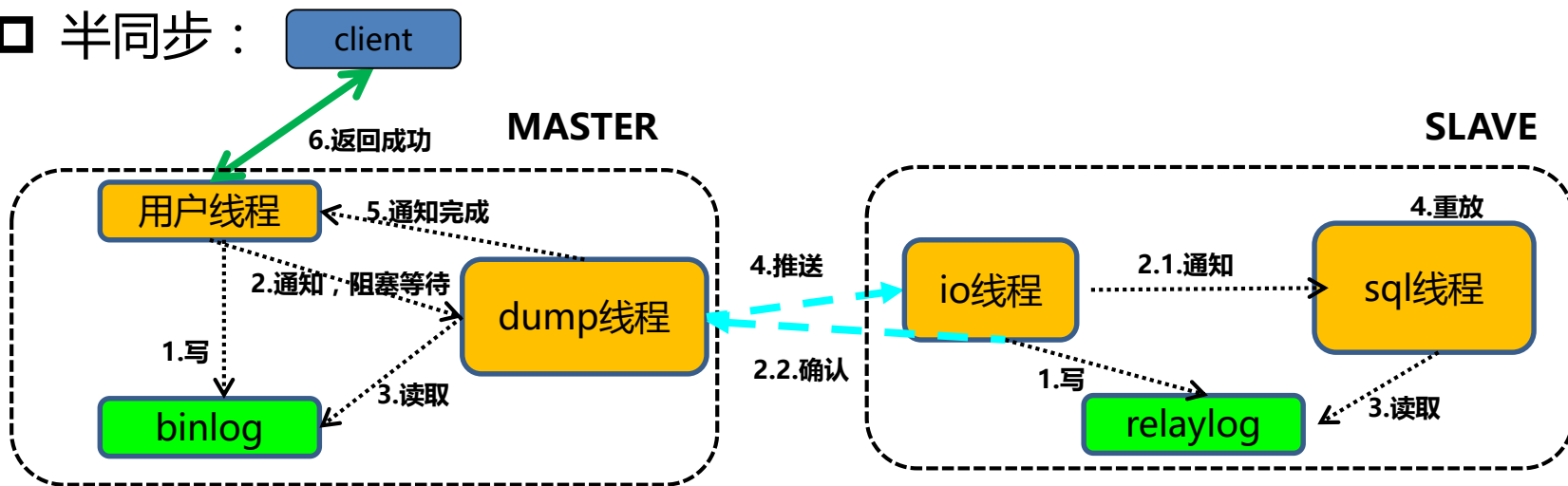


# MySQL主从复制现状

## □ 异步：



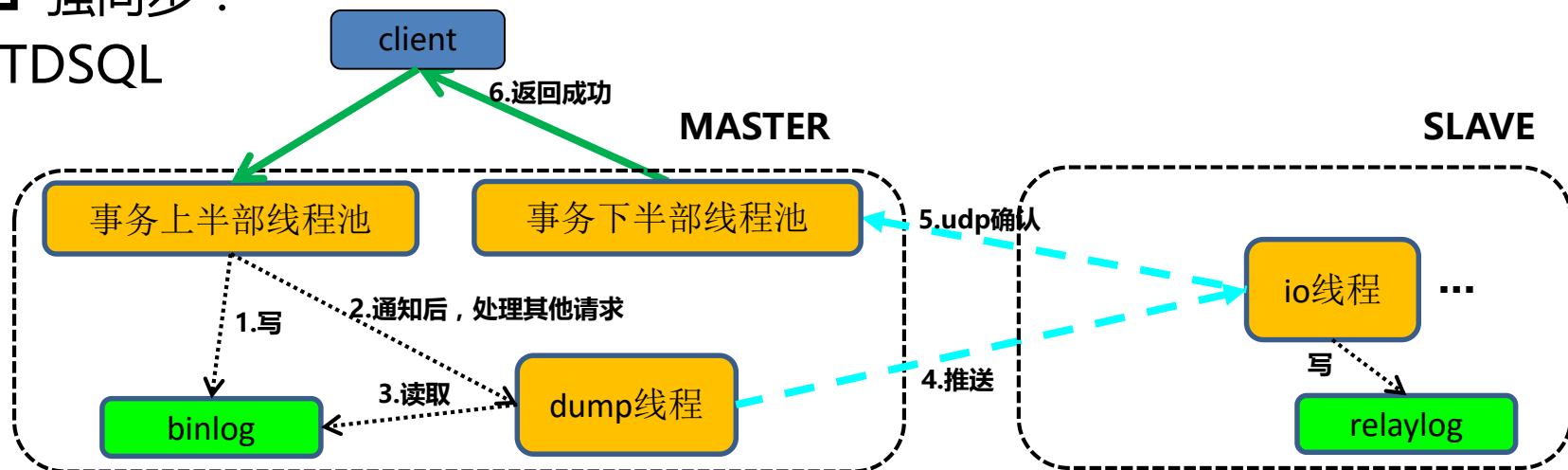
## □ 半同步：



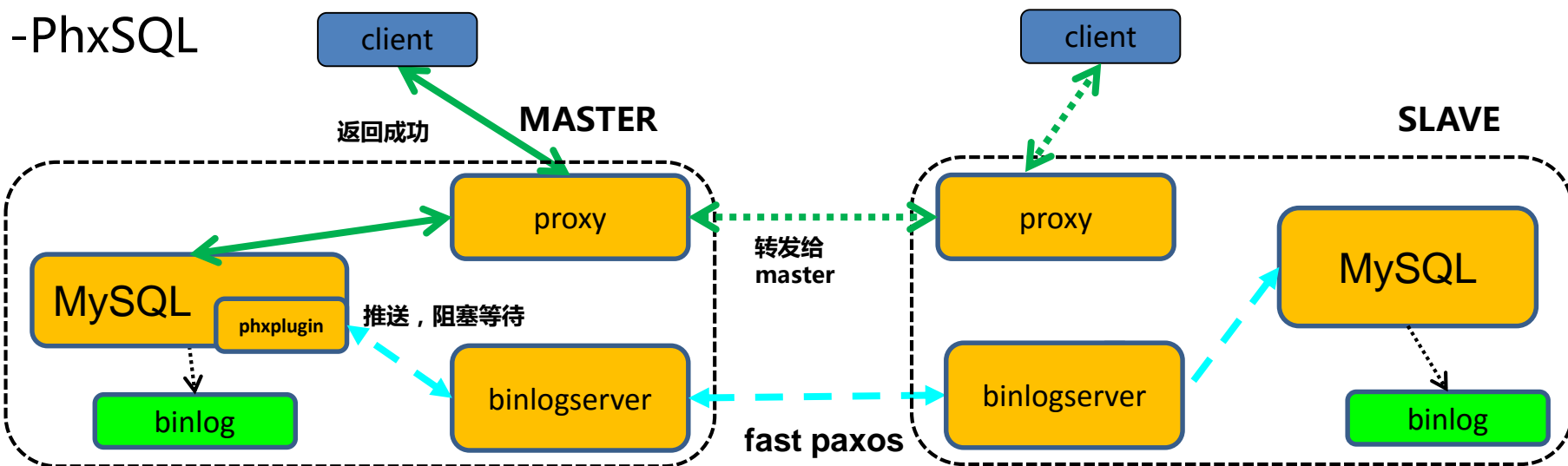
# 业界解决方案

## 强同步：

### -TDSQL



### -PhxSQL



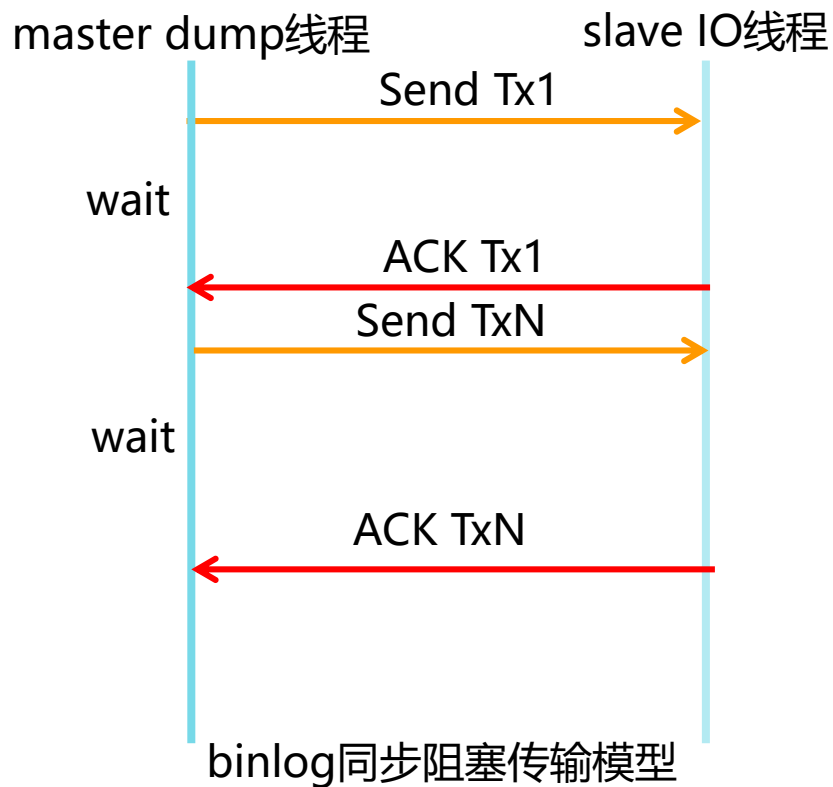


# 解决思路

- 优化master收发
- 优化slave收发

# 优化master收发(1)

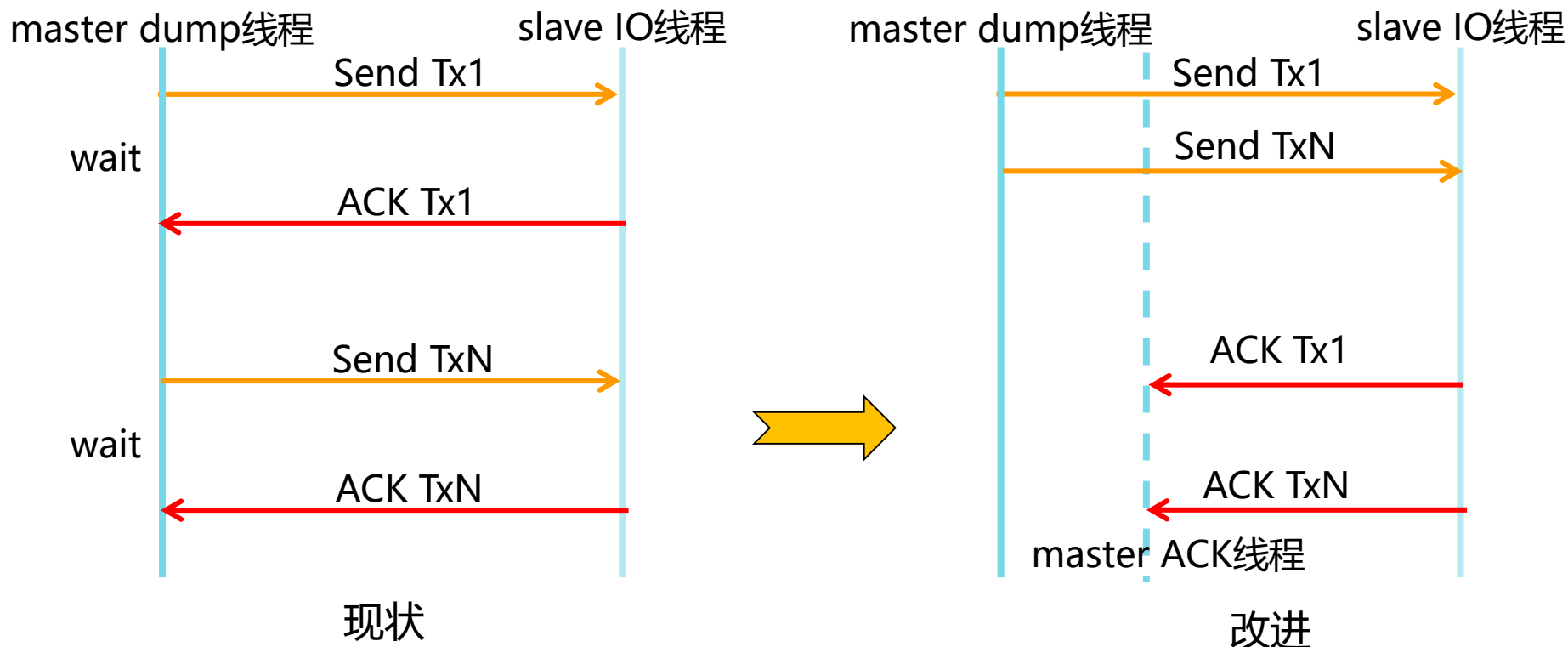
- ❑ 问题：dump线程串行发送binlog，即发送某个事务binlog后，需要等待slave回包，再发送下一个事务binlog。
- ❑ 理论上，纯写 $TPS_{max} = 1000 / RTT(ms)$ 。一般情况同城跨园区之间RTT为2~3ms左右， $TPS_{max}$ 约330~500。





# 优化master收发(2)

- ❑ 方案：master binlog发送和接收异步化
- ✓ dump线程负责发送
- ✓ **ack线程负责处理回包**，通知事务线程继续提交



- 1.理论上吞吐量可做到网卡上限
- 2.基于replication插件实现，移植方便

# 优化master收发(3)

□sysbench全cache纯update压测（128线程）：

数据库版本	同步类型	TPS	单事务耗时 (ms)	同步RTT	性能基准对比
mariadb10.1	异步	56596	2.25	2.60	100.00%
mariadb10.1	半同步	374	342.26	2.60	0.66%
ecdbplugin	半同步	21547	5.94	2.60	38.10%

# 优化slave收发(1)

❑ 问题：slave异常无法接收binlog，影响master可用性

✓ 数据页损坏

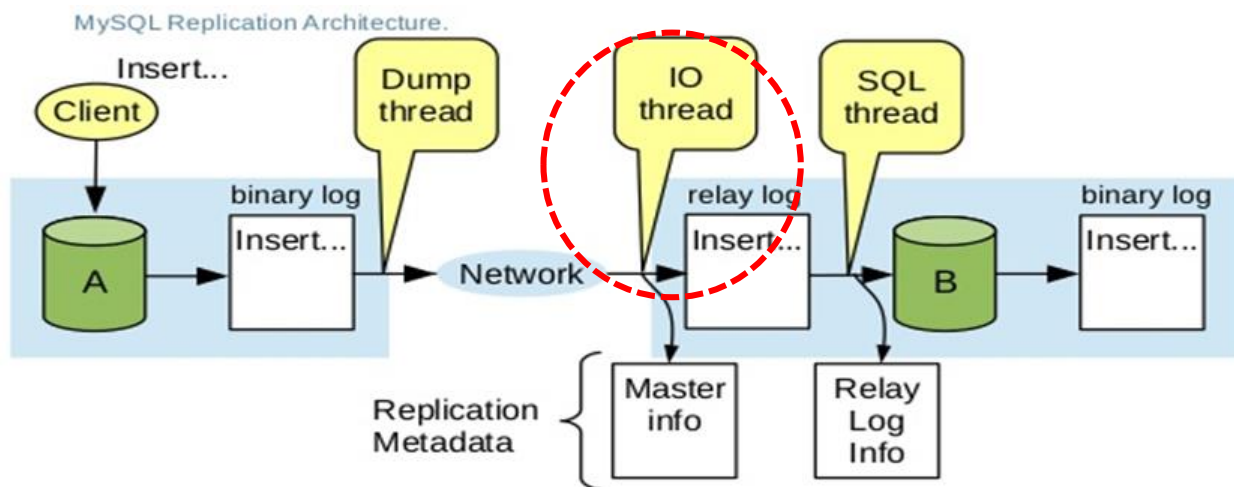
✓ OOM

❑ 问题：MySQL slave的IO线程接收效率有待优化

✓ 锁冲突：IO/SQL线程间的锁冲突(开启并行复制后，更严重)

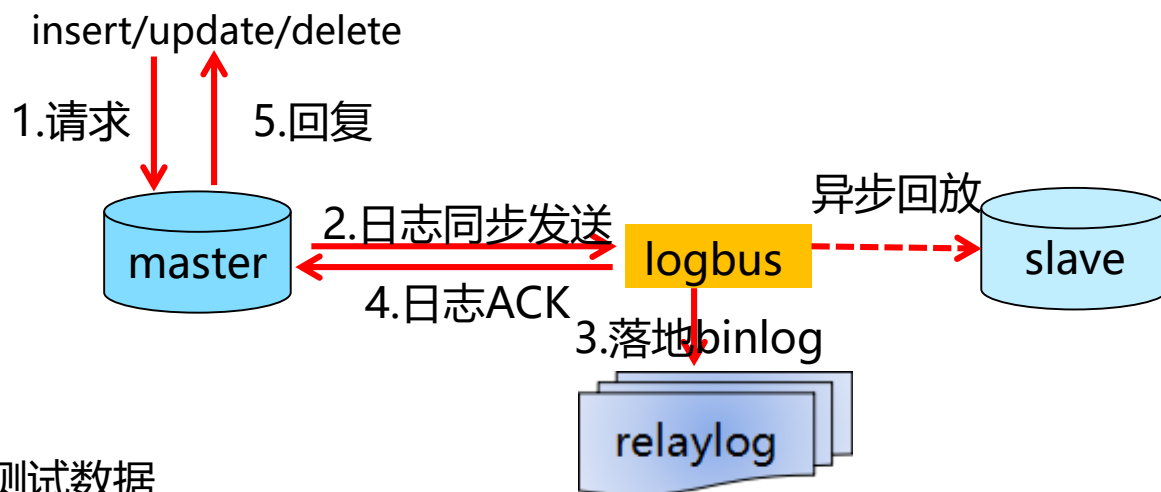
✓ 小IO消耗：IO线程离散小磁盘IO消耗过多的IOPS

✓ 串行化：IO线程接收和落盘操作串行(fsync影响性能)



# 优化slave收发(2)

- ❑ 方案：构建独立于mysqld的快速复制通道logbus，完成binlog的接收
- ✓ 模拟slave向master建立主从关系，同步binlog
- ✓ 落盘合并优化，避免极端情况的Partial Write带来的数据不一致
- ✓ 异步化Network IO线程和Disk IO线程(**group commit**)
- ✓ 支持异步回放到slave中



外置logbus的原因：

- 1. 减少对MySQL的侵入
  - > 支持多版本
  - > 提升稳定性
- 2. HA：支持多版本

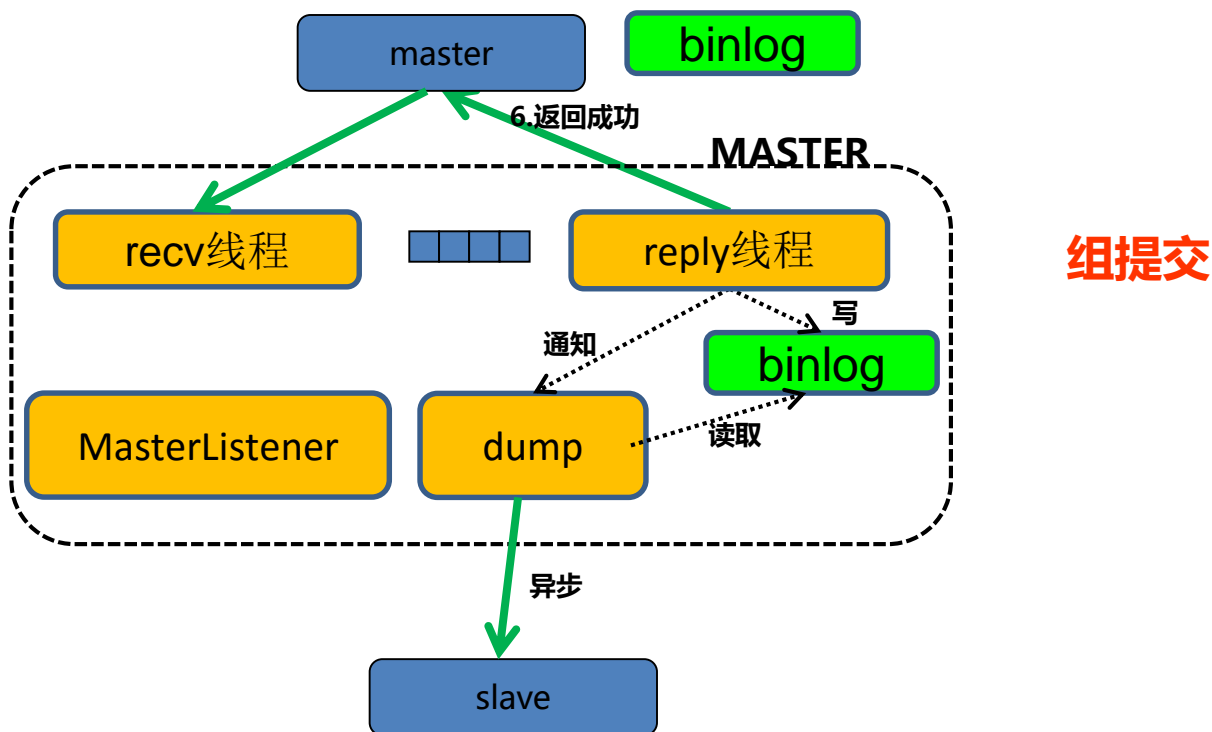
原型测试数据

数据库版本	同步类型	TPS	单事务耗时(ms)	同步RTT	性能基准对比
MySQL5.7	异步	33193	3.82	2.60	100.00%
MySQL5.7	半同步	15395	8.32	2.60	46.30%
MySQL5.7	logbus	21607	5.92	2.60	65.00%

# 优化slave收发(3)

## □ logbus功能：

- ✓ 支持异步、半同步，与master建立主从
- ✓ 支持mysql GTID、mariadb GTID以及filepos等方式定位初始binlog位置
- ✓ 保存的binlog文件名与偏移与master一样（类似mysqlbinlog）
- ✓ 异步回放到slave（支持流控）





# eCDB高可用特性(2)

## 跨园区



# 挑战

- ❑ 物理机房不支持跨园区VIP漂移
- ❑ 网络分区的情况下，如何自动切换并保证不出现双写

# 解决思路

## □ 规避双写

- ✓ **替换TGW**：使用中间件在前端收敛访问（**proxy集群**）
  - 确保该中间件访问同个后端实例，规避双写
  - 影响可用性，必须支持跨园区高可用
- ✓ **租约**：实例根据租约提供服务（**zookeeper**）
  - 在租期内可提供读写或者只读服务，不在则不提供任何服务
  - 影响可用性，必须支持跨园区高可用

## □ 使用新的机制通知业务访问ip:port发生变化

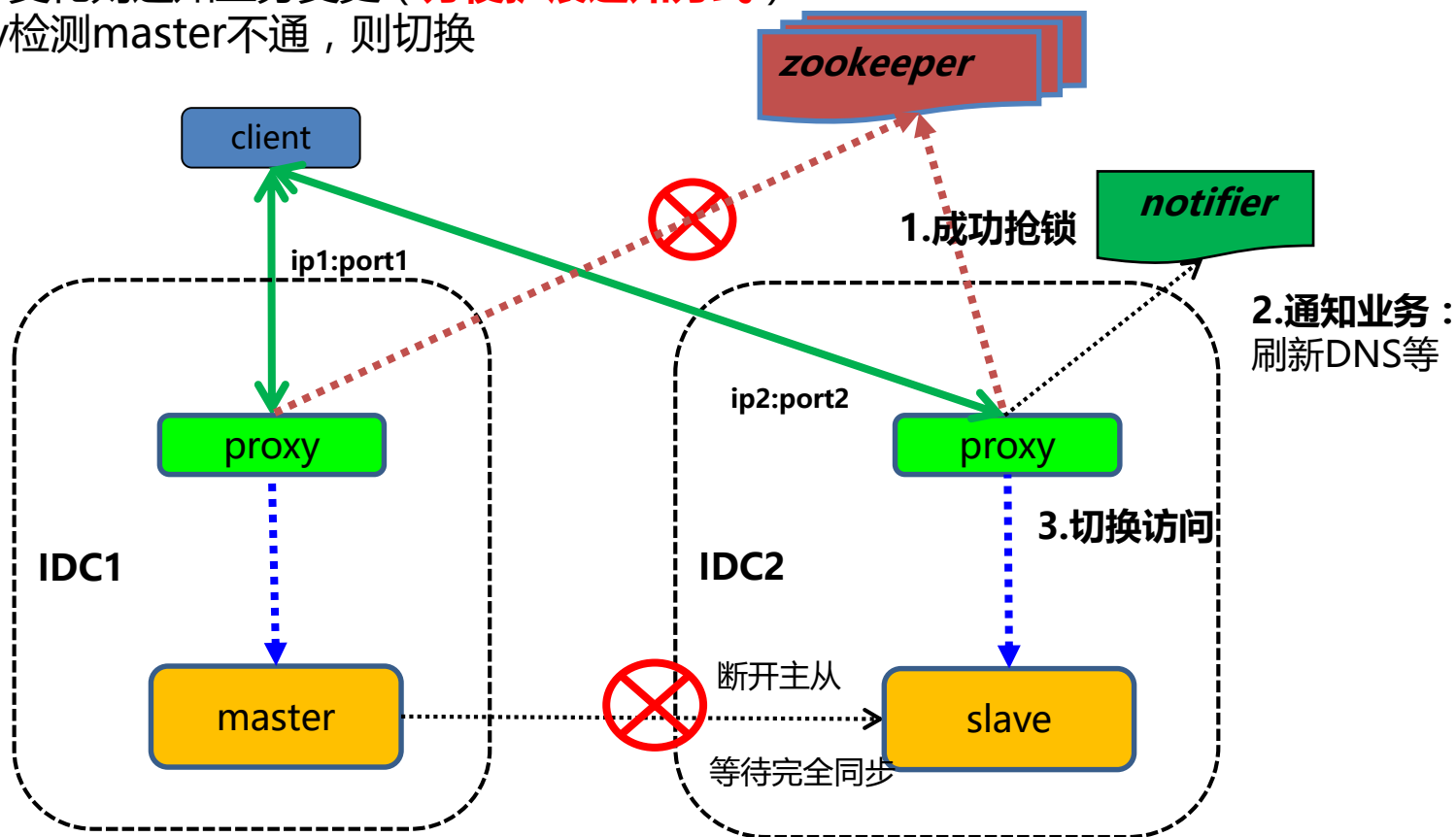
- ✓ DNS：通用，但更新时间不可控
- ✓ 回调url：业务提供url，变更时，CDB主动回调
- ✓ zookeeper：提供zk路径给业务订阅



# proxy方案

## □ 设计要点

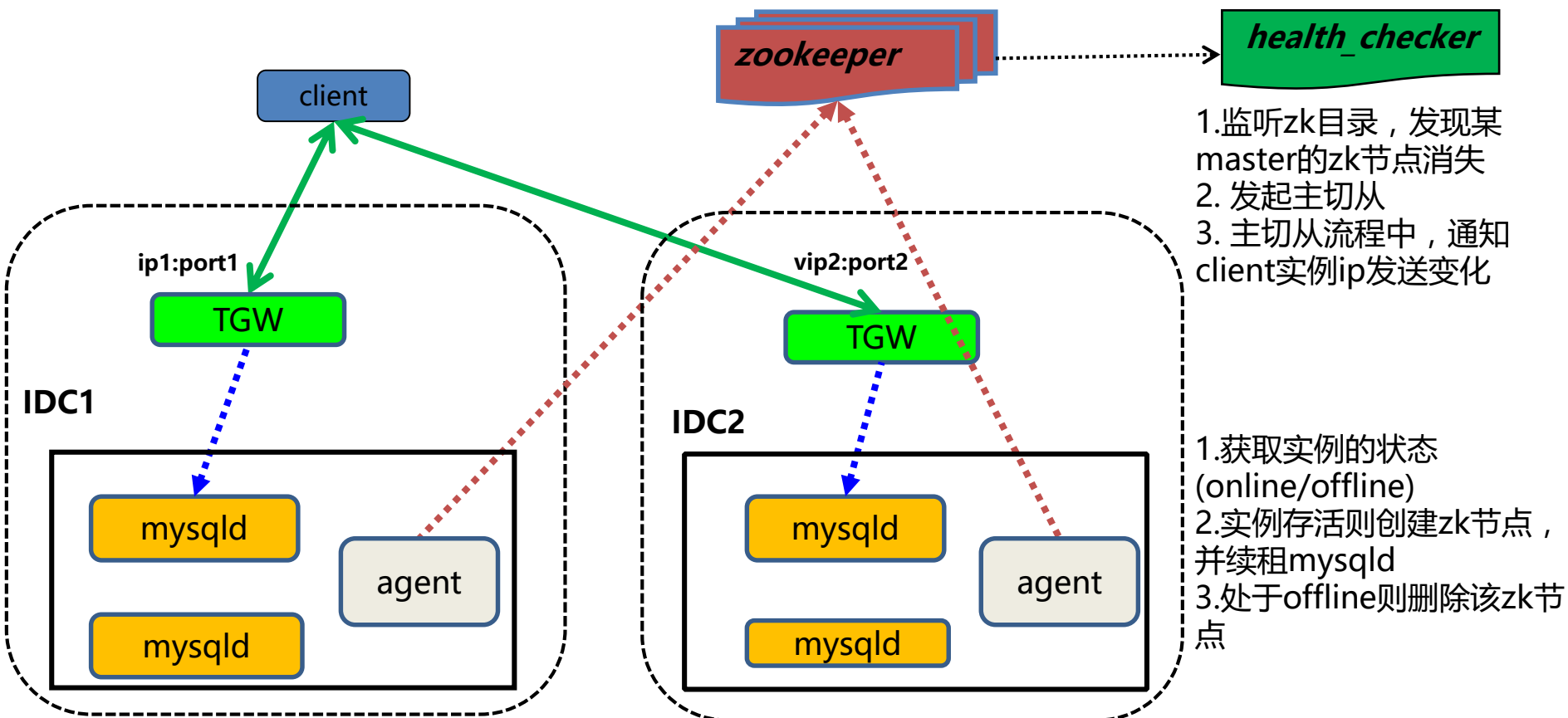
- ✓ 只有**抢到zk锁**的proxy才能提供服务（**规避双写**）
- ✓ proxy无状态（状态数据存DB），使用zk轻松实现跨园区HA
- ✓ 通知与切换分离：
  - proxy变化则通知业务变更（**方便扩展通知方式**）
  - proxy检测master不通，则切换



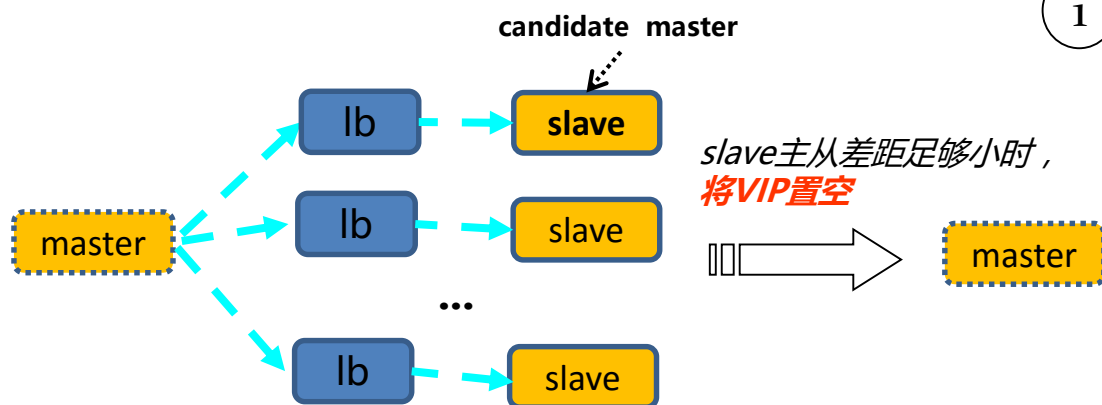
# 租约方案

## □ 设计要点

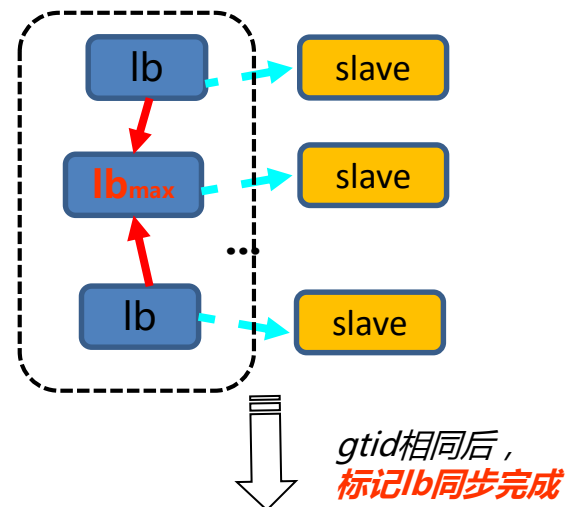
- ✓ 使用zk发放租约
- ✓ agent负责给该机器所有使用中的实例申请租约
- ✓ mysql接收agent的租约更新，过期则拒绝服务
- ✓ checker模块监听租约情况，过期则发起主切从



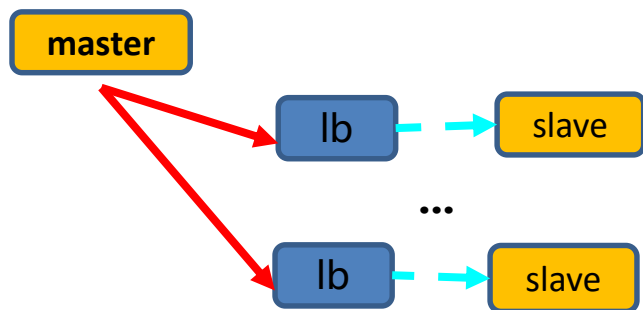
# HA流程



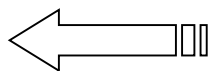
- 1 选择gtid最大的lb：**lb<sub>max</sub>**  
lb与lb<sub>max</sub>建立主从，等待同步



- 3 关闭新master的logbus  
lb与新master建立连接



candidate完成后，  
绑定vip，提升为  
master



- 2 等待slave重放完成

# 内容回顾

## □ 强同步--优化性能

- ✓ master侧：优化半同步插件，支持强同步
- ✓ slave侧：引入外部模块logbus，快速回复ack

## □ 跨园区--规避双写

- ✓ Proxy：访问都通过proxy，proxy通过zk锁确保唯一
- ✓ zookeeper：agent负责续租zk和mysqld，确保mysqld ≤ zk租约过期时间

# 结束

## □ eCDB仍在努力：

- ✓ 单机性能 ：引入业界优秀优化思想
- ✓ 可运营 ：完善MySQL的管理能力

The top corners of the slide feature decorative geometric shapes. On the left, there is a dark blue sphere with a network of white lines and dots. On the right, there is a similar structure, a dark blue sphere with a network of white lines and dots. The background is a solid blue color with white geometric lines forming a large 'V' shape in the center and several diagonal lines extending from the corners towards the center.

**Gdevops**

# 全球敏捷运维峰会

The bottom corners of the slide feature decorative geometric shapes. On the left, there is a dark blue sphere with a network of white lines and dots. On the right, there is a similar structure, a dark blue sphere with a network of white lines and dots. The background is a solid blue color with white geometric lines forming a large 'V' shape in the center and several diagonal lines extending from the corners towards the center.

THANK YOU !