本文是作者在ACMUG 2016 MySQL年会上的演讲内容，版权归作者所有。

中国MySQL用户组（China MySQL User Group）简称ACMUG。
ACMUG是覆盖中国MySQL技术爱好者的一个技术社区，是Oracle User Group Community和MairaDB Foundation共同认可的MySQL技术社区。

我们关注MySQL，MariaDB，以及其他一切周边的开源数据库和开源工具，我们交流使用经验，推广开源技术，为开源贡献力量。

我们是开放社区，欢迎任何关注MySQL及其相关技术的人加入，我愿意跟其他任何技术组织和团体保持沟通和展开合作。

我们期望在我们的活动中大家都能以开心的、轻松的姿态交流技术，分享技术，形成一个良性循环，从而每个人都可以有一份收获。

ACMUG的口号：开源，开放，开心

关注ACMUG公众号，参与社区活动，交流开源技术，分享学习心得，一起共同进步。

腾讯云

云数据库服务CDB技术演进和实践

CDB for MySQL：腾讯规模最大的关系型数据库服务

Tencent's Largest Relational Database Service

程彬 2016.12.10@ACMUG年会V0.9

大纲

1.云数据库概览

2.CDB之存储实践

3.CDB之复制实践

4.CDB之引擎实践

个人介绍

程 彬

bencheng

- 腾讯-技术工程事业群-基础架构部
- 08年加入腾讯，一直从事数据存储相关工作
- 11年开始从事MySQL Cloud相关研发工作
- 2016年4月16日，邂逅了ACMUG

云数据库概览 – 什么是云数据库

- 产品定义：满足云计算特征的数据库服务
- 技术定义：数据库内核+云化功能

云数据库概览 – 鹅厂的云数据库系列

p SQL

CDB for MySQL

p NoSQL

CDB For MySQL – 概览

PB+

1W+

1W+

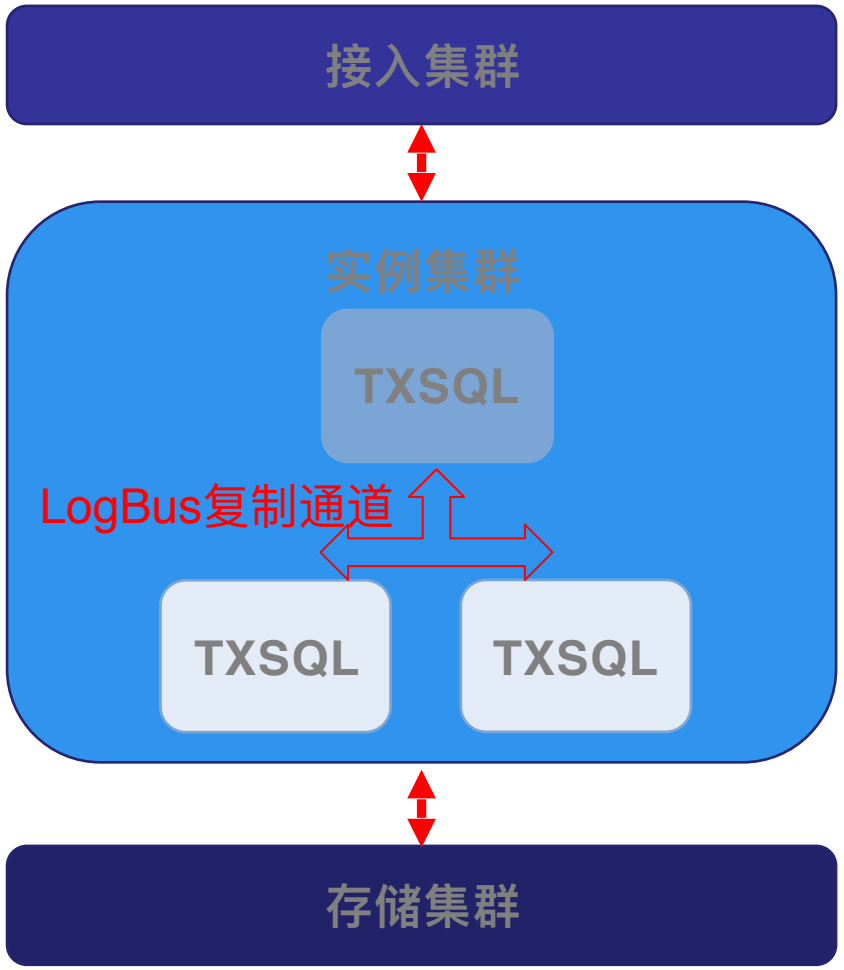
90%+

- ✓ 腾讯基于MySQL的DBaaS(DataBase as a Service)，腾讯最大规模的关系型数据库服务
- ✓ PB级别存储规模，万级别的机器数，万级别的开发商数
- ✓ 占腾讯云所有关系型数据库类产品总收入的90%+
- ✓ 工信部认证：99.95%可用性，99.9996%数据可靠性
- ✓ 覆盖行业：游戏、金融、移动、视频等

CDB For MySQL– 腾讯内部业务代表

除了传统的社交、游戏之外，微信红包、微信支付、QQ彩票等金融交易类业务也是重要用户

CDB For MySQL– 内核技术栈

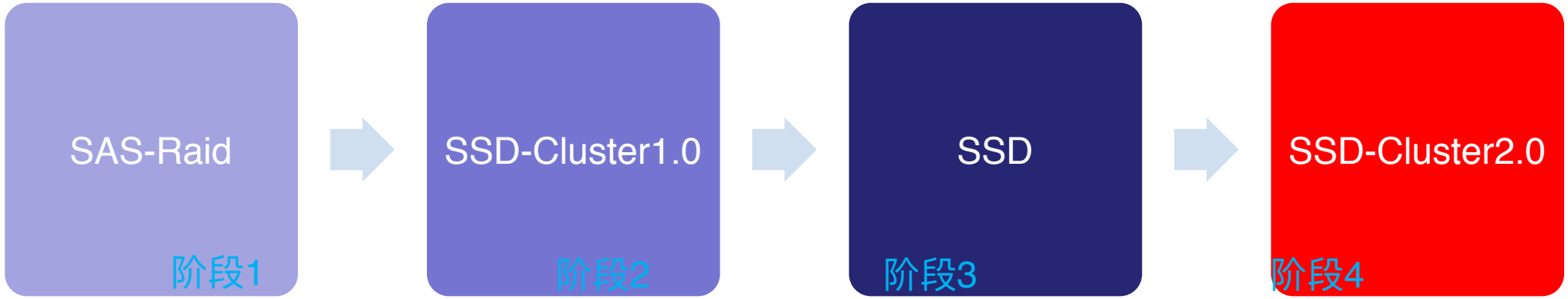


- ✓ 路由(分片)
- ✓ 安全
- ✓ HA

- ✓ 数据库引擎
- ✓ 多副本复制

- ✓ 数据存储

存储篇 – 三次存储革命

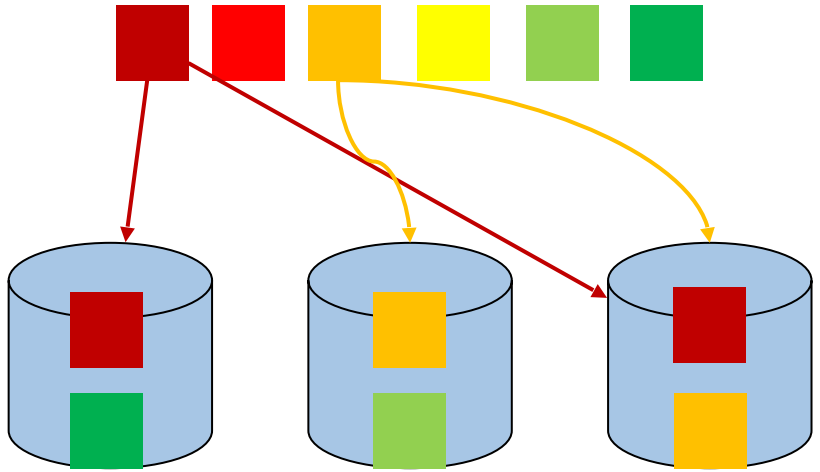


- 数据库存储的本质：面向块的存储
- 根据存储介质特性，进行数据库存储技术设计

存储篇 – 从本地SAS盘走向SSD集群1.0

- ✓ 本地SAS盘存储的主要矛盾
  - ✓ 随机IO性能低
  - ✓ 单机最大存储容量受限
  - ✓ 贝佐斯定律玩不转-给用户降价慢

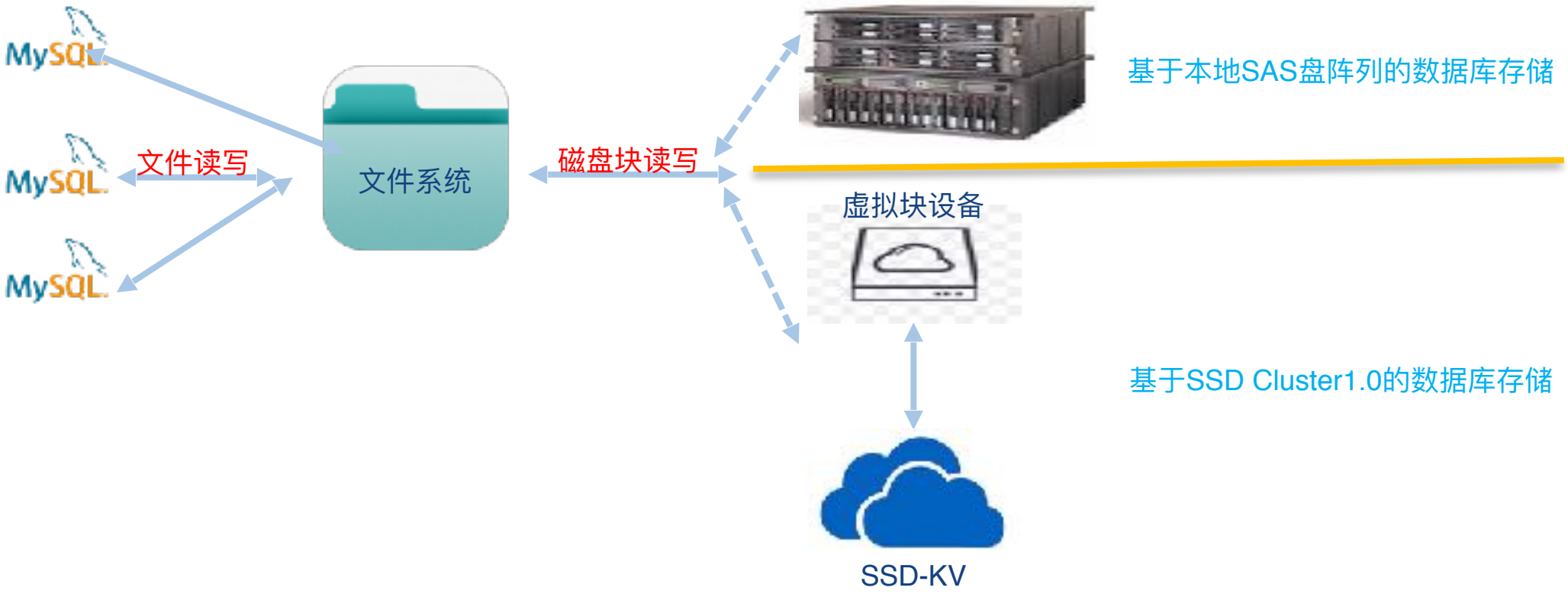
- ✓ 当时可能的解决方案
  - ✓ 性能：优化存储引擎、采用SSD
  - ✓ 容量：加更多的磁盘、换更大的盘
  - ✓ 成本：存储压缩、高风险存储超卖



分布式存储

- ✓ 分布式存储能解决的问题
  - ✓ 性能扩展
  - ✓ 容量扩展
  - ✓ 安全的超卖：按需分配、动态伸缩

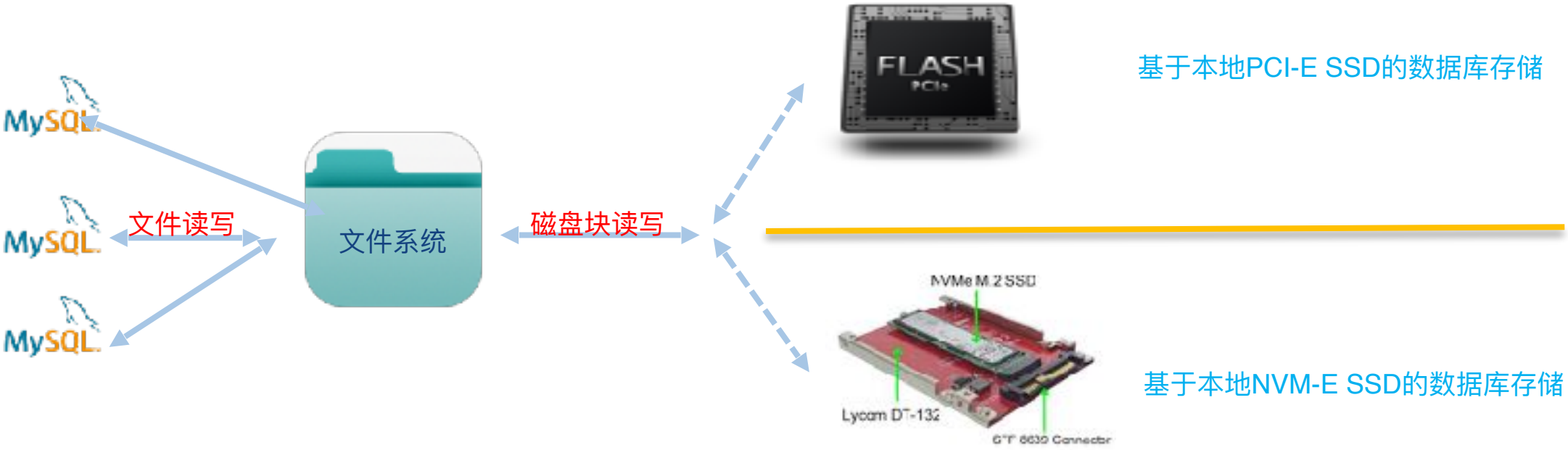
存储篇 – 从本地SAS盘走向SSD集群1.0(Cont.1)



- ✓ 分布式存储遇到的问题
  - ✓ 网络IO延时、带宽
  - ✓ 存储集群的蝴蝶效应

存储篇 – 从SSD集群1.0走回本地SSD

- ✓ SSD集群1.0的主要矛盾
  - ✓ IO密集型核心应用上云，更高性能要求
  - ✓ SSD存储介质的发展：量大价低，单机12T



- ✓ 本地SSD存储下，数据库如何建立亮点
  - ✓ 用好SSD：基础块设备调优、SPDK
  - ✓ 选择新的技术制高点：数据库引擎本身的性能和稳定性

存储篇 – 从本地SSD走向SSD集群2.0

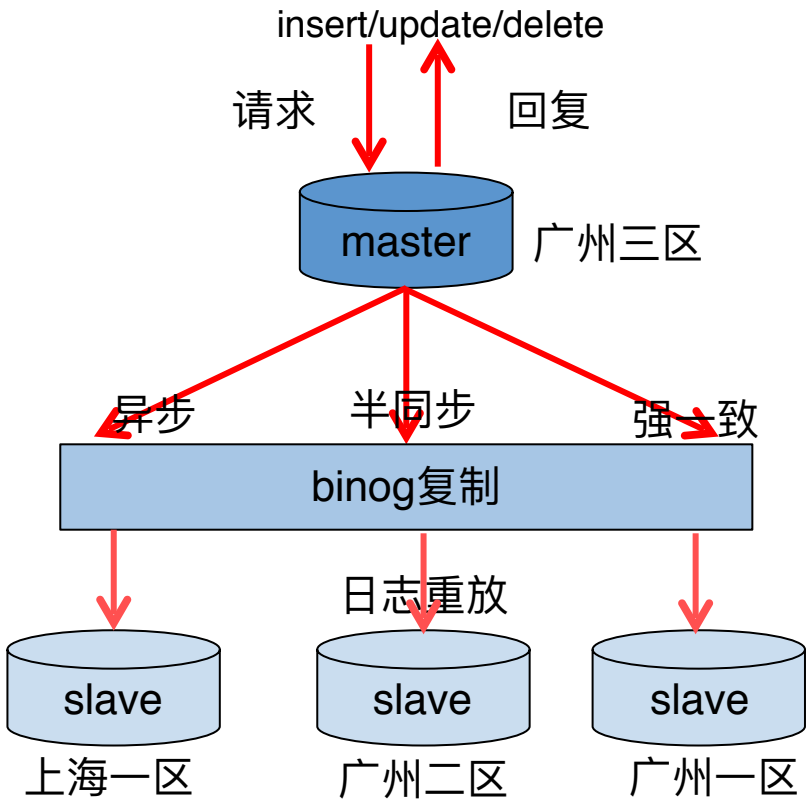
- ✓ 本地SSD的主要矛盾
  - ✓ 金融、政企等数据库上云
  - ✓ 容量无法与SAN\NAS\专有存储相比
  - ✓ 中间件sharding解决了扩展性，但兼容性有问题

- ✓ SSD集群2.0思路
  - ✓ 目标：100TB、SQL完全兼容的传统行业DB服务
  - ✓ 思想：基于SDP-Shared Disk Parallel
  - ✓ 数据库节点和存储分离，数据库节点有主从之分
  - ✓ 尽量减少IO次数：主数据库节点才能写存储集群，从节点不会写
  - ✓ 主从之间原生binlog复制被基于block的物理复制替代

复制篇 – 三种复制模式

- 复制结合机房部署，灵活选择，达到最有效的容灾效果

复制篇 – 数据同步的要求



多zone部署下的复杂数据同步

数据库版本	同步类型	TPS	单事务耗时	同步RTT	性能基准对比
MySQL5.7	异步	33193	3.82ms	2.60ms	100%
MySQL5.7	半同步	15395	8.33ms	2.60ms	46.30%

强一致下的性能损耗

- ✓ 复杂业务环境下的数据同步要求
  - ✓ 灵活的多zone部署下的一致性要求
  - ✓ 强一致下的性能要求

复制篇 – 同步性能优化分析

✓ 问题：跨园区下，同步性能损耗大

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%
MySQL5.7	半同步	15395	8.33	2.60	46.30%

✓ 分析：同步复制下，单个事务耗时

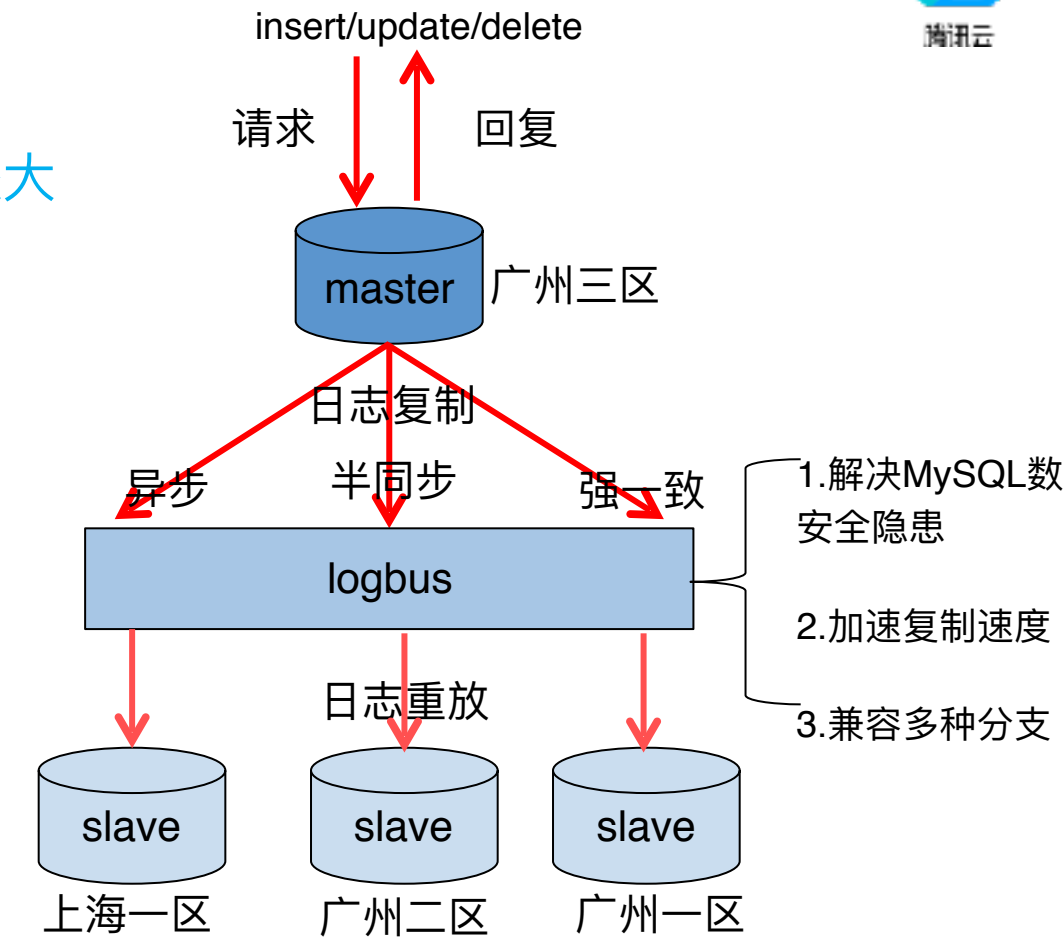
- ✓ $T_{total}=T_{sql}+T_{engine}+T_{replicate}$
- ✓ T_{sql}和T_{engine}和复制无关，受限单机处理性能
- ✓ $T_{replicate}=T_{binlog网络传输}+T_{slave落地binlog}$，$T_{binlog网络传输}$取决于RTT值
- ✓ 测试数据验证：

$$T_{slave落地binlog} = 8.33ms - 3.82ms(约等于sql加engine耗时) – 2.60(RTT耗时)$$
$$= 1.91ms$$

- ✓ 模拟写binlog：SAS盘上顺序写512B的write+fsync的耗时为0.13ms
- ✓ 问题来了：1.91 – 0.13 =1.78ms去哪儿呢?

复制篇 – 同步性能优化实现

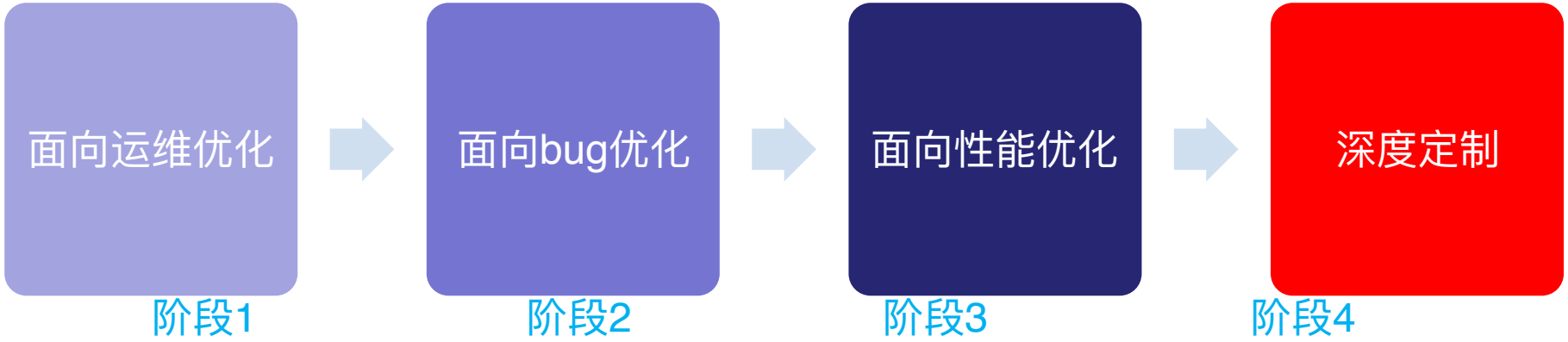
- ✓ 问题：MySQL slave的IO线程接收master的binog延时比较大
- ✓ 分析：对slave IO线程耗时进行深入分析
  - ✓ 锁冲突：IO/SQL线程间的锁冲突，如元数据文件锁
  - ✓ 小IO消耗：IO线程离散小磁盘IO消耗过多的IOPS
  - ✓ 串行化：IO线程接收和落盘操作串行
- ✓ 方案：构建独立于MySQL的快速复制通道logbus



基于logbus的复制架构

数据库版本	同步类型	TPS	单事务耗时	同步RTT	性能对比
MySQL5.7	异步	33193	3.82ms	2.60ms	100%
MySQL5.7	半同步	15395	8.32ms	2.60ms	46.30%
MySQL5.7	logbus	22169	5.92ms	2.60ms	66.79%

引擎篇 – Tencent MySQL(TXSQL)



- 自研技术10+、社区红利30+
- 积极拥抱开源社区

引擎篇 – TXSQL特性功能概览

腾讯云

CDB for MySQL：腾讯规模最大的关系型数据库服务

欢迎关注公众号：腾讯云数据库CDB

Tencent's Largest Relational Database Service