知乎首页已读数据万亿规模下高吞吐低时延查询系统架构设计

孙晓光

知乎

极客时间 | 企业服务

这是一封正经“福利”贴

想做团队的领跑者 需要迈过这些“槛”

成长型企业，易忽视人才体系化培养 企业转型加快，团队能力又跟不上	VS	从基础到进阶，超100+一线实战 技术专家带你系统化学习成长
团队成员技能水平不一， 难以一“敌”百人需求	VS	解决从小白到资深技术人所遇到 80%的问题
寻求外部培训，奈何价更高且 集中式学习	VS	多样、灵活的学习方式，包括 音频、图文 和视频
学习效果难以统计，产生不良循环	VS	获取员工学习报告，查看学习 进度，形成闭环

课程顾问「橘子」

回复「QCon」
免费获取
学习解决方案

 # 极客时间企业账号 # 解决技术人成长路上的学习问题

自我介绍

孙晓光，知乎搜索后端技术负责人

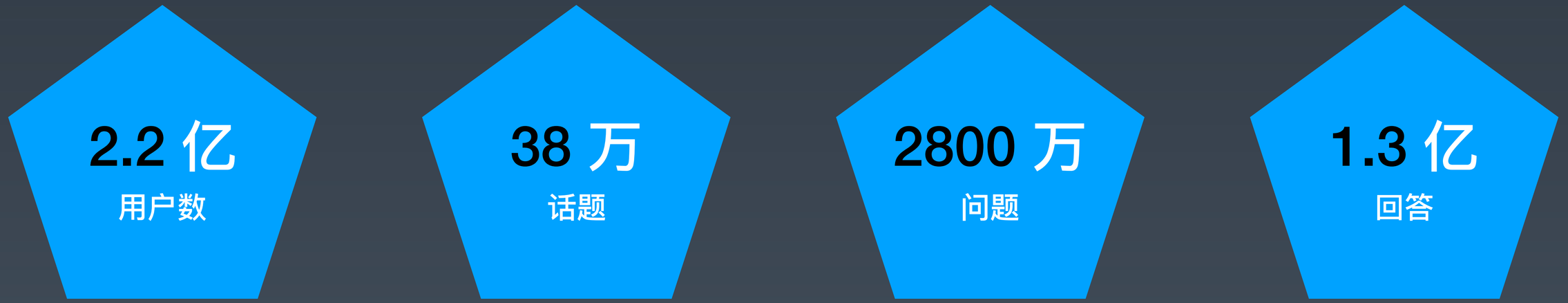
曾从事私有云相关研发，关注云原生技术

TiKV 项目 Committer

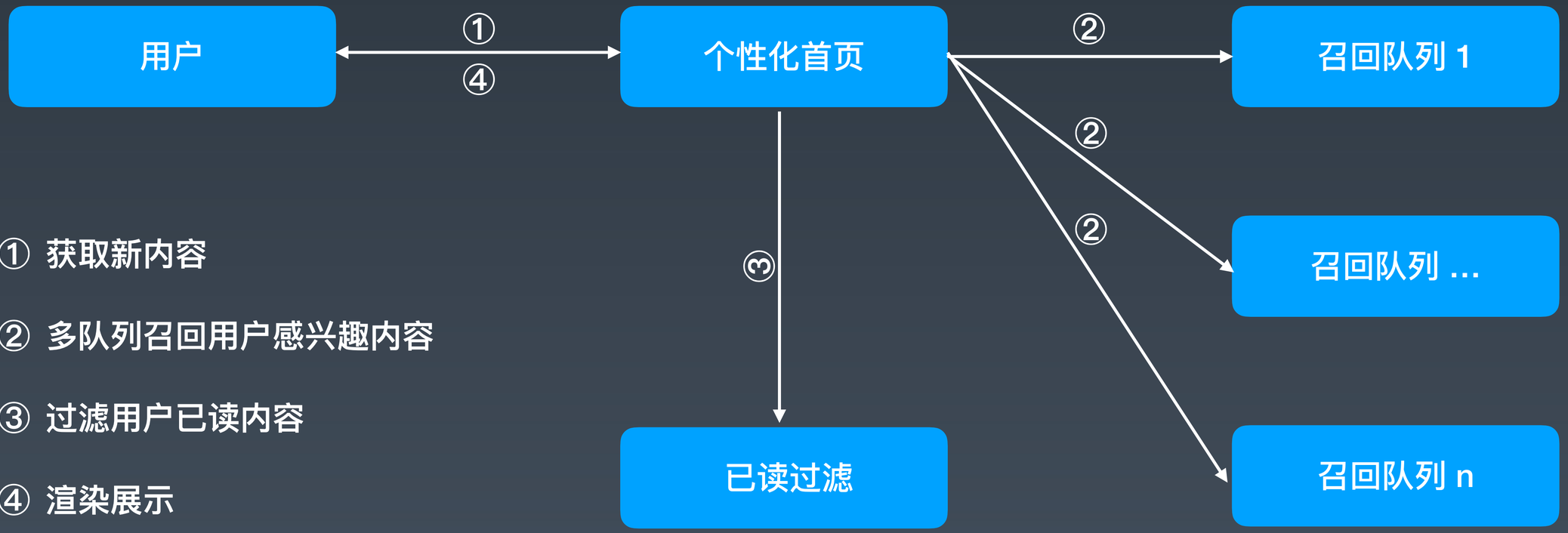
目录

- **业务场景，技术挑战**
- **早期架构，设计目标**
- **关键设计，核心组件**
- **全面云化，面向未来**

知乎：可信赖的问答社区



业务场景



① 获取新内容

② 多队列召回用户感兴趣内容

③ 过滤用户已读内容

④ 渲染展示

已读过滤示意图

业务特点

- 可用性要求 “高”
  - 个性化首页和个性化推送，最重要的流量分发渠道
- 写入量 “大”
  - 峰值每秒写入 40K+ 行记录，日新增记录近 30 亿条
- 历史数据 “长期” 保存
  - 一万二千亿条历史数据

业务特点

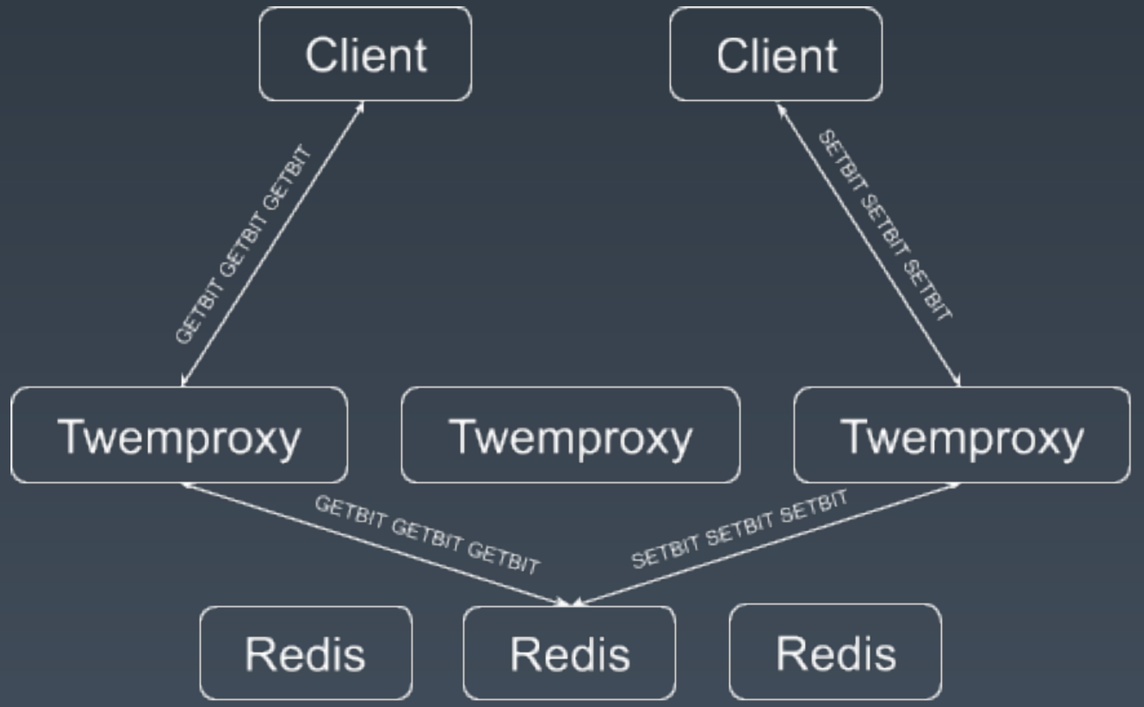
- 查询吞吐 “高”
  - 峰值 30K QPS / 12M+ 条已读检查
- 响应时间 “敏感”
  - 90ms 超时
- 可以容忍 “false positive”

目录

- 业务场景，技术挑战
- 早期方案，架构演进
- 核心组件，关键设计
- 全面云化，面向未来

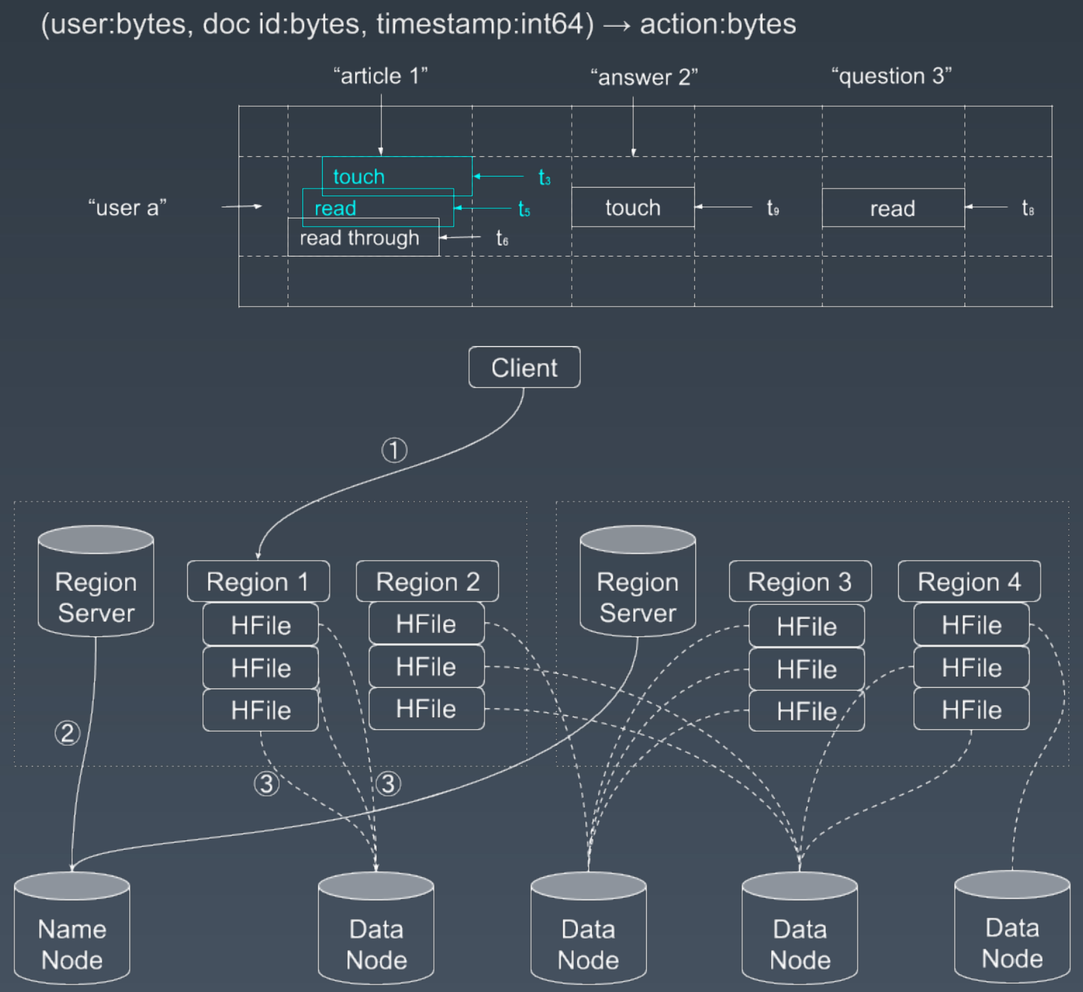
早期方案一

- BloomFilter on Redis Cluster
  - BITSET 存储
  - 操作放大严重
  - 基于内存成本高
  - 无法精确控制 False Positive Rate



早期方案二

- HBase
  - row key 存储用户 id
  - qualifier 存储文档 id
  - 访问稀疏 Cache 命中率低
  - Latency 不稳定

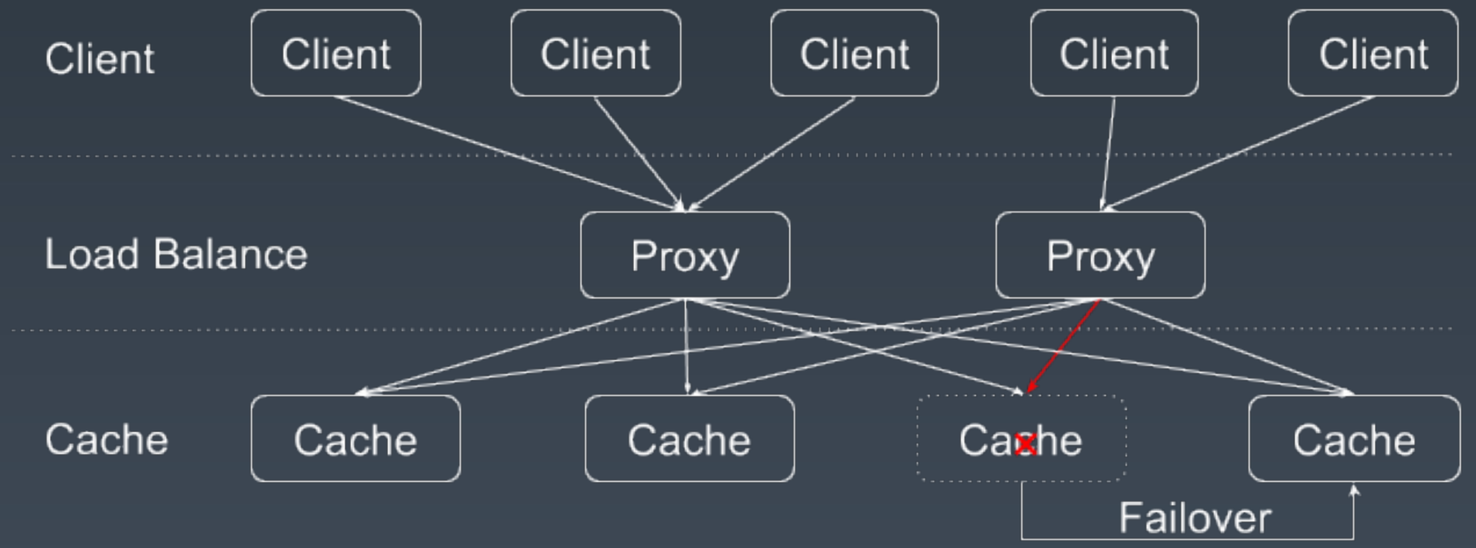


设计目标

- 高可用
  - ✓HBase
  - ✓BloomFilter on Redis Cluster
- 高性能
  - ✘HBase
  - ✓BloomFilter on Redis Cluster
- 易扩展
  - ✓HBase
  - ✘BloomFilter on Redis Cluster

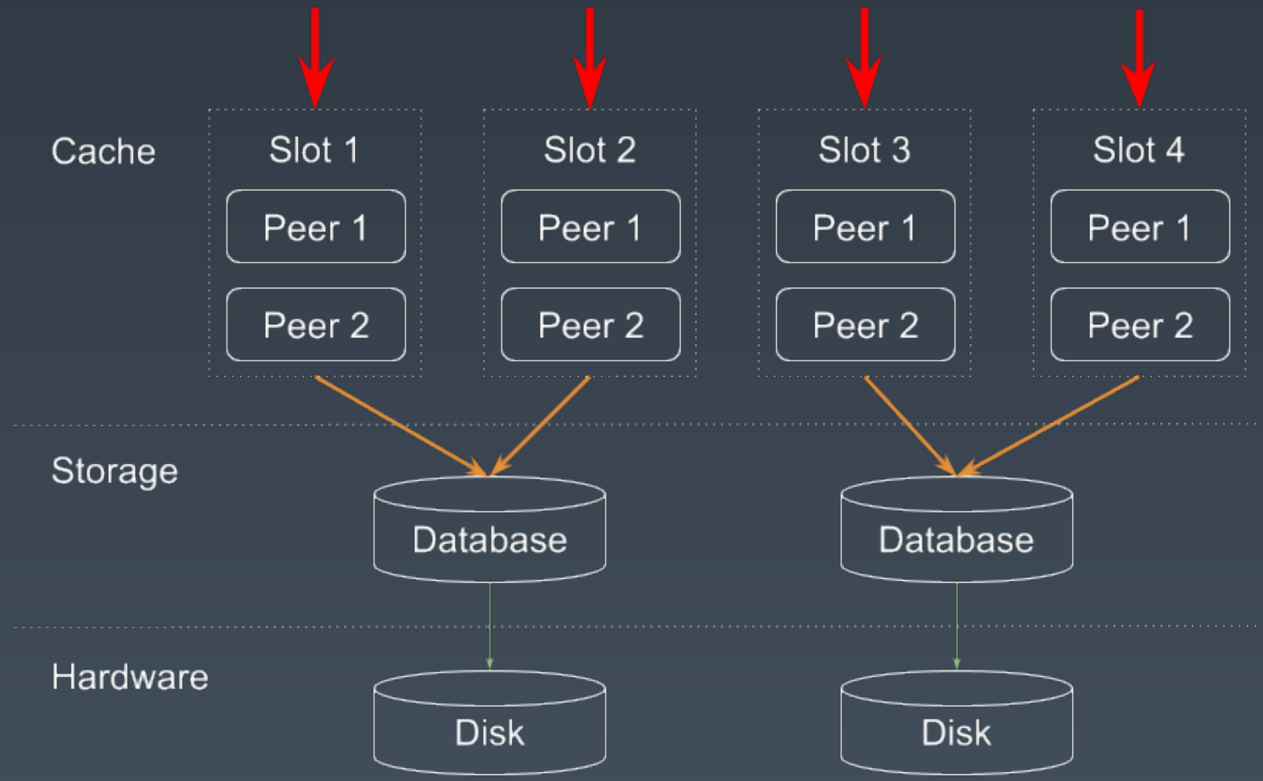
设计思路

- 高可用
  - 故障感知
  - 自愈机制
  - 隔离变化



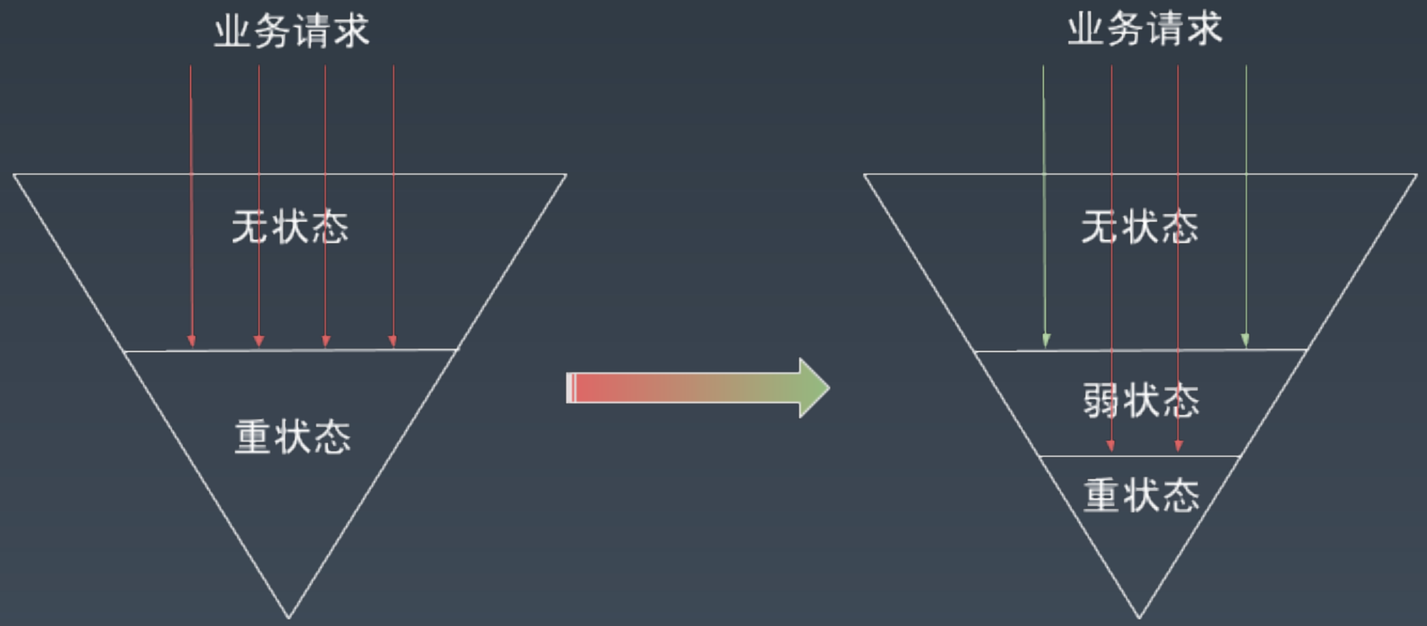
设计思路

- 高性能
  - 缓存拦截
  - 副本扩展
  - 压缩降压

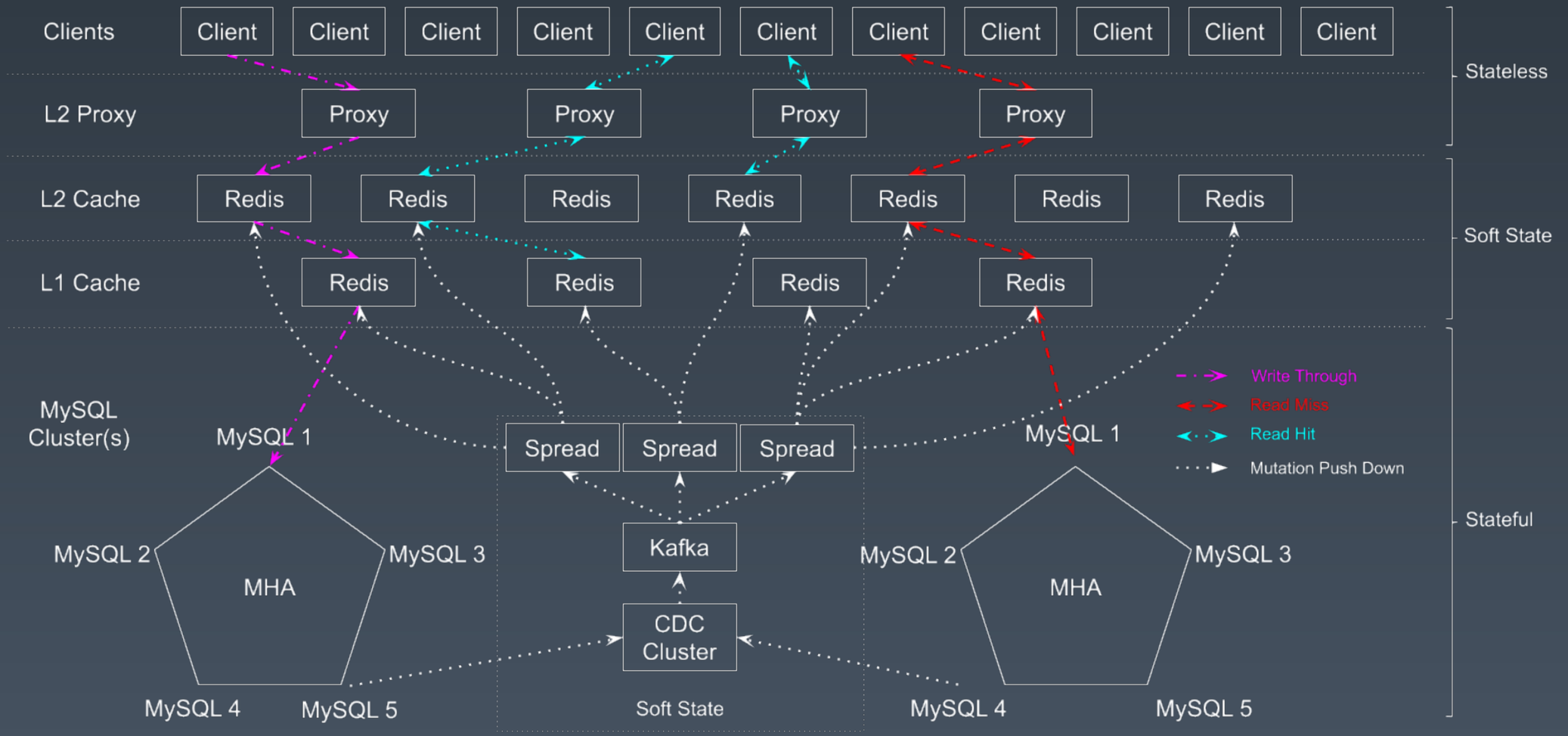


设计思路

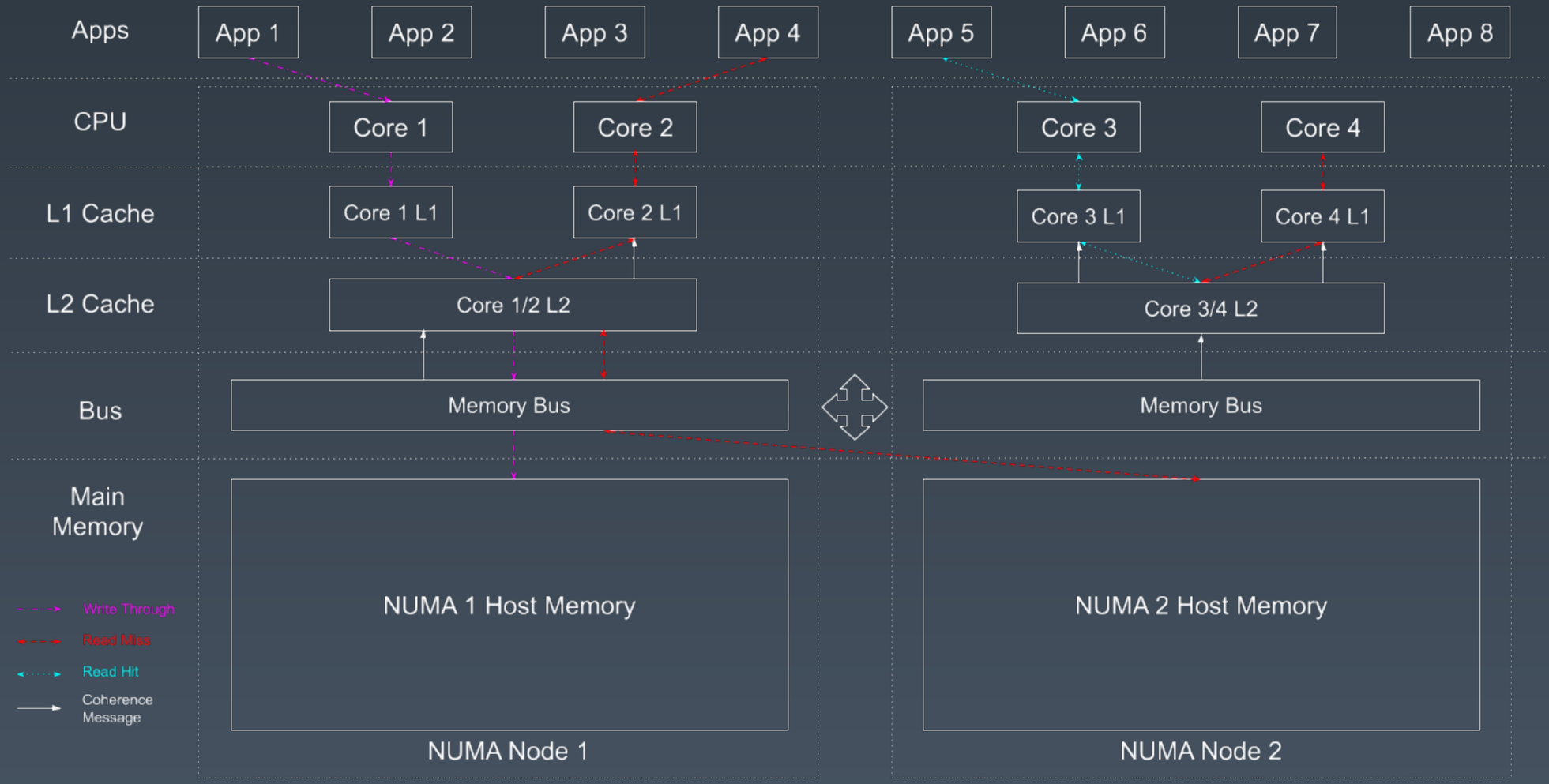
- 易扩展
  - 无状态
  - 弱状态
  - 重状态



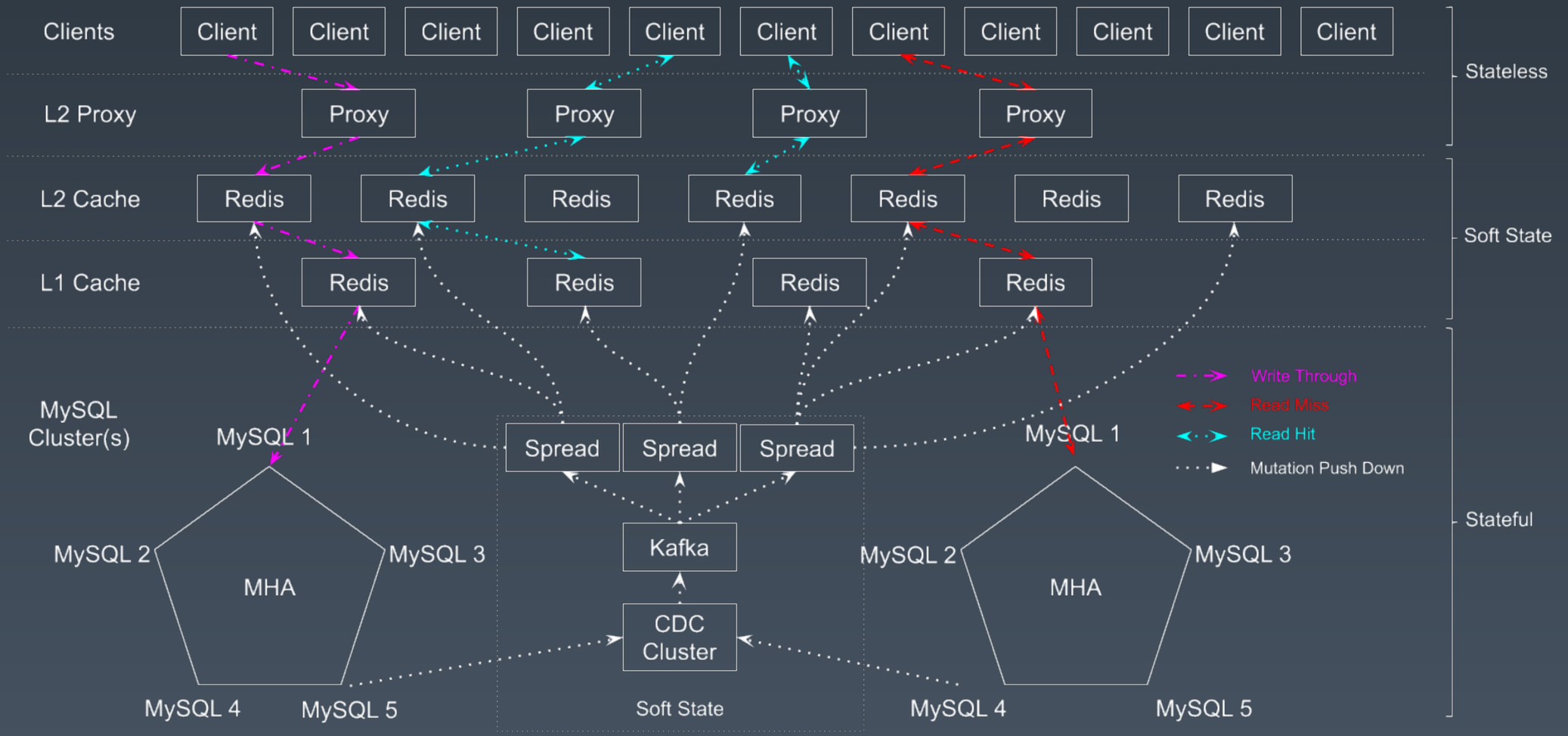
已读服务


Clients
Client
Client
Client
Client
Client
Client
Client
Client
Client
Client
Client
Stateless
L2 Proxy
Proxy
Proxy
Proxy
Proxy
L2 Cache
Redis
Redis
Redis
Redis
Redis
Redis
Redis
Soft State
L1 Cache
Redis
Redis
Redis
Redis
MySQL Cluster(s)
MySQL 1
Spread
Spread
Spread
MySQL 1
Write Through
Read Miss
Read Hit
Mutation Push Down
Stateful
MySQL 2
MySQL 3
MHA
Kafka
CDC Cluster
MySQL 2
MySQL 3
MHA
MySQL 4
MySQL 5
Soft State
MySQL 4
MySQL 5

灵感来源


Apps
App 1
App 2
App 3
App 4
App 5
App 6
App 7
App 8
CPU
Core 1
Core 2
Core 3
Core 4
L1 Cache
Core 1 L1
Core 2 L1
Core 3 L1
Core 4 L1
L2 Cache
Core 1/2 L2
Core 3/4 L2
Bus
Memory Bus
Memory Bus
Main Memory
NUMA 1 Host Memory
NUMA 2 Host Memory
Write Through
Read Miss
Read Hit
Coherence Message
NUMA Node 1
NUMA Node 2

已读服务

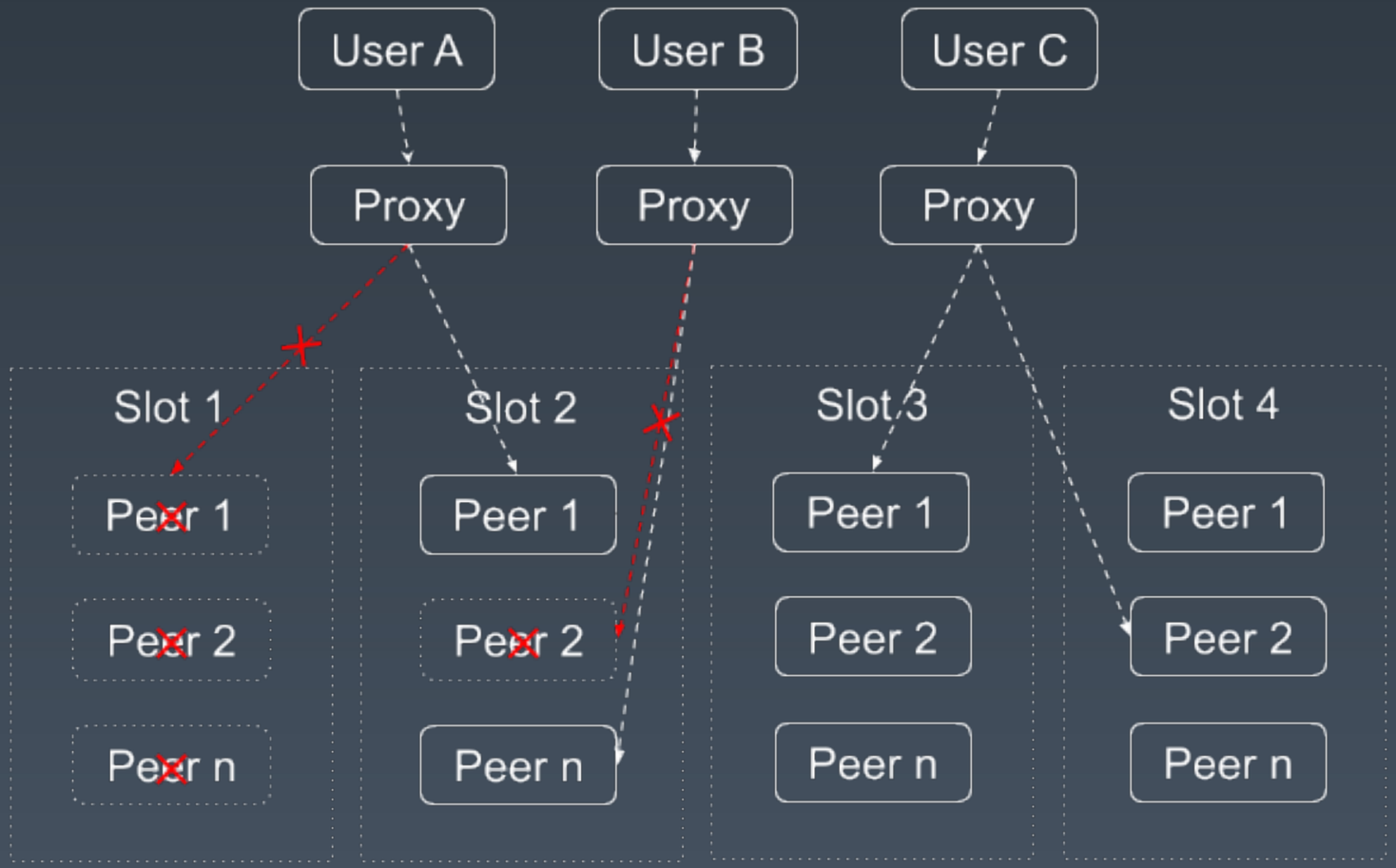

Clients
Client
Client
Client
Client
Client
Client
Client
Client
Client
Client
Client
Stateless
L2 Proxy
Proxy
Proxy
Proxy
Proxy
L2 Cache
Redis
Redis
Redis
Redis
Redis
Redis
Redis
Soft State
L1 Cache
Redis
Redis
Redis
Redis
MySQL Cluster(s)
MySQL 1
Spread
Spread
Spread
MySQL 1
Write Through
Read Miss
Read Hit
Mutation Push Down
Stateful
MySQL 2
MySQL 3
MHA
Kafka
CDC Cluster
MySQL 2
MySQL 3
MHA
MySQL 4
MySQL 5
Soft State
MySQL 4
MySQL 5

目录

- 业务场景，技术挑战
- 早期方案，架构演进
- 核心组件，关键设计
- 全面云化，面向未来

关键组件设计

- Proxy
  - Slot 内多副本高可用
  - Slot 内会话粘滞
  - Slot 间故障降级

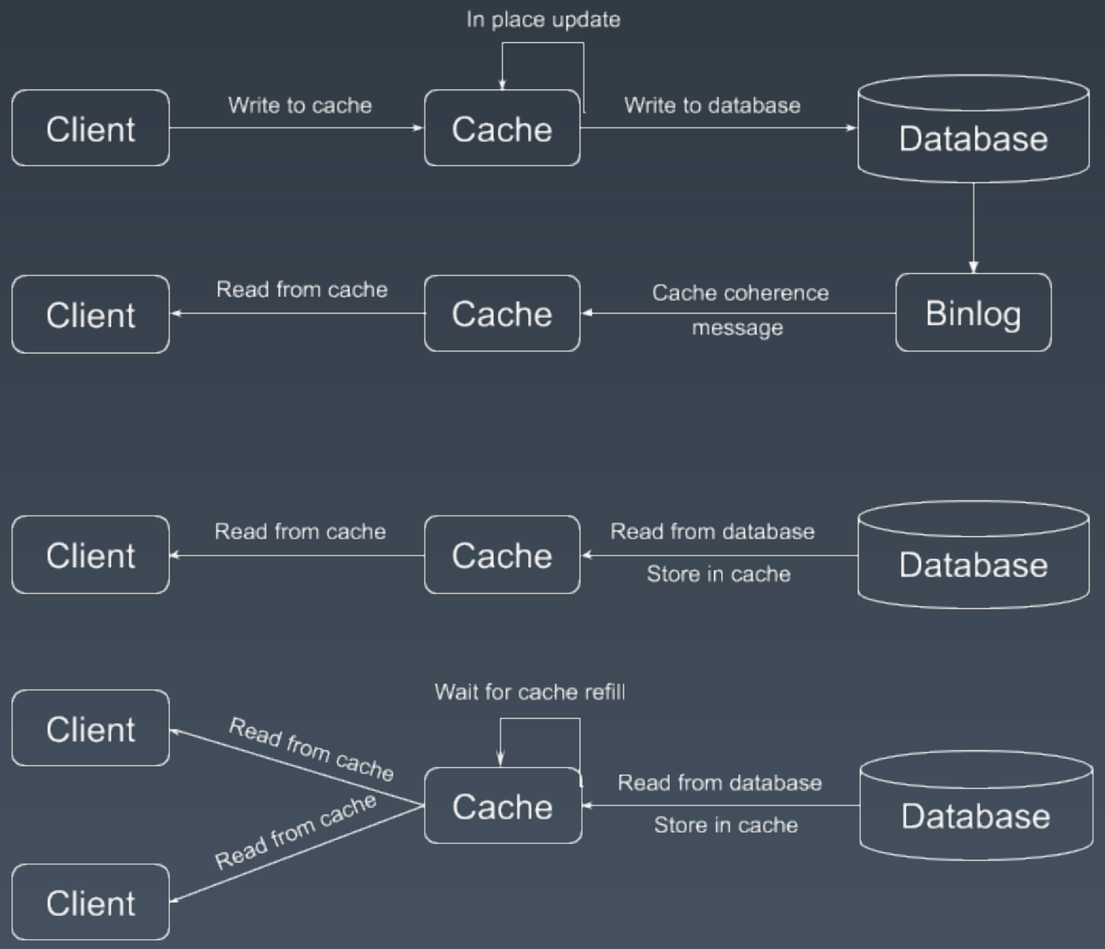


关键组件设计

- Cache
  - BloomFilter 增加缓冲密度

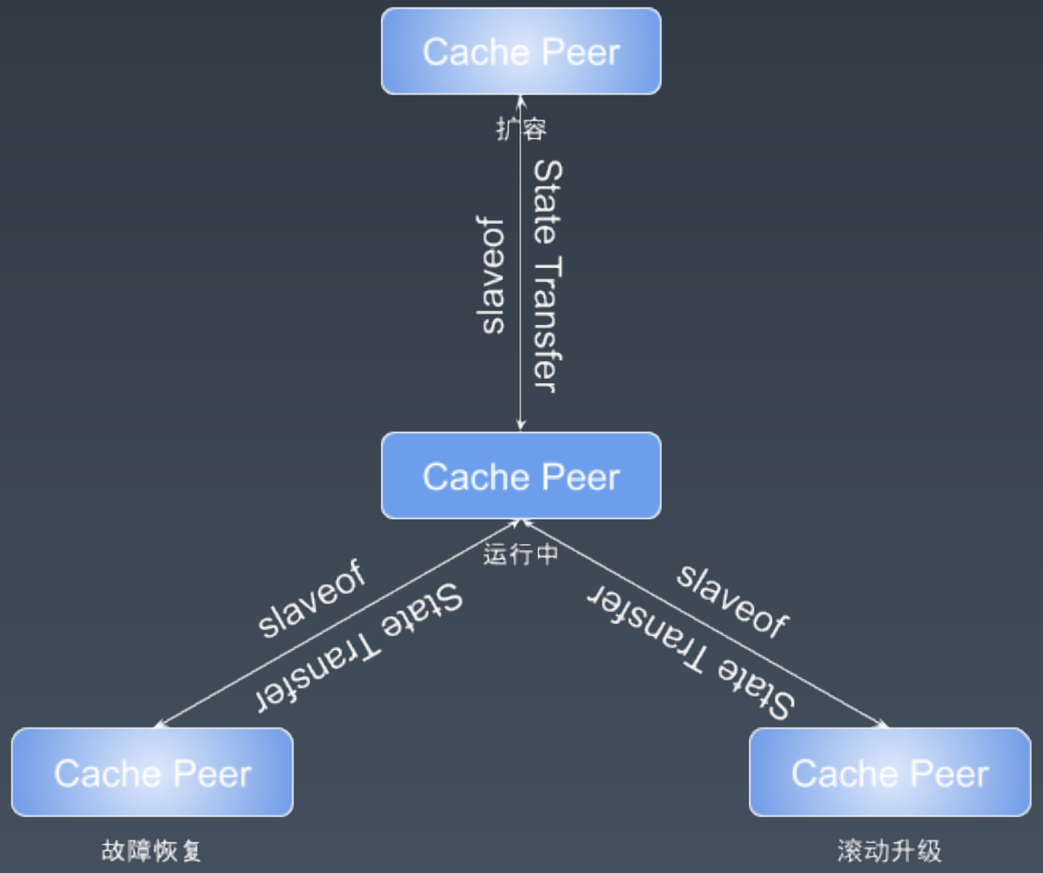
关键组件设计

- Cache
  - In Place 更新，不失效缓存
  - 数据变更订阅，副本间 Cache 状态最终一致
  - 避免惊群



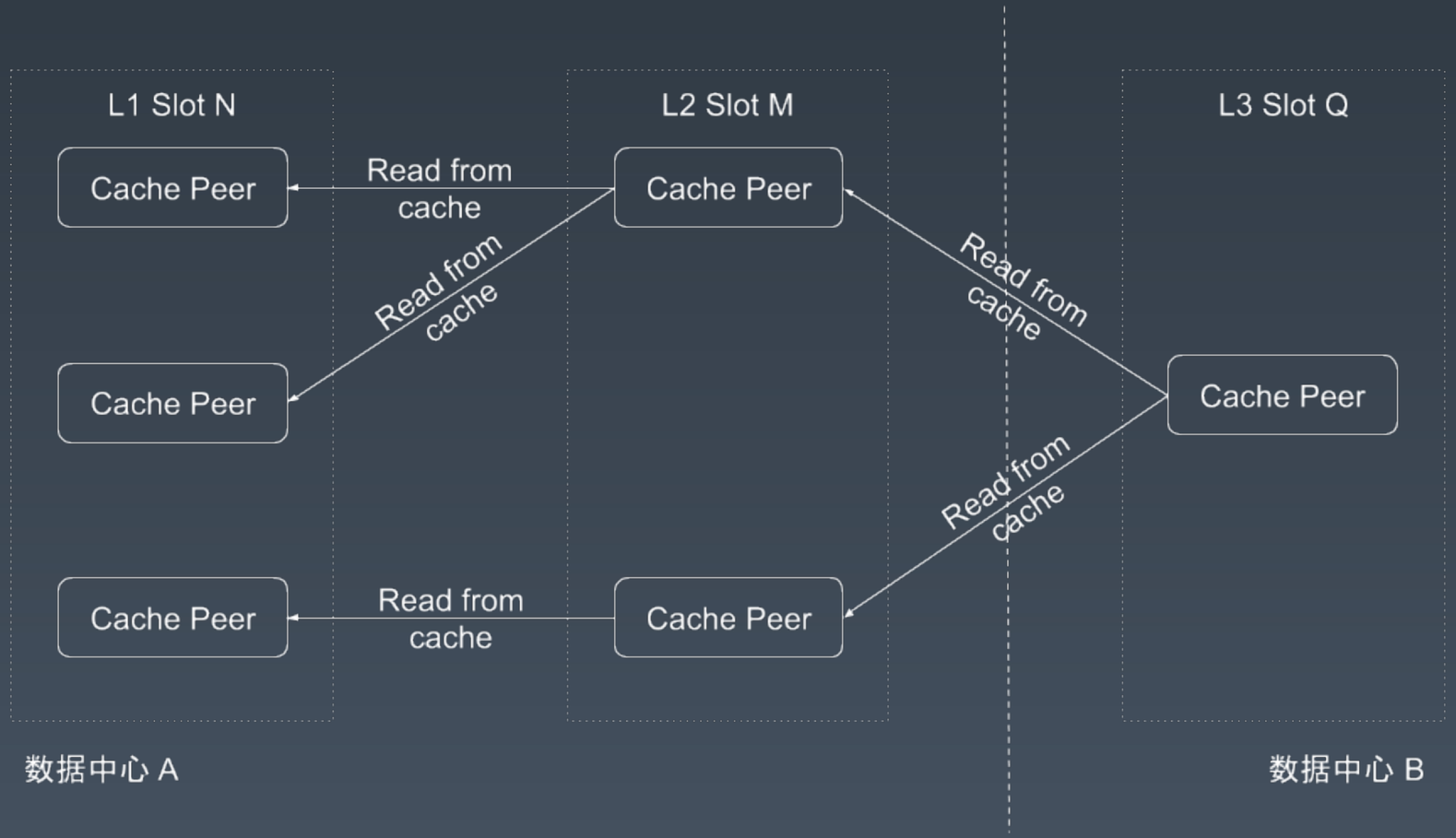
关键组件设计

- Cache
  - 缓冲状态迁移
    - 平滑扩容
    - 平滑滚动升级
    - 故障快速恢复



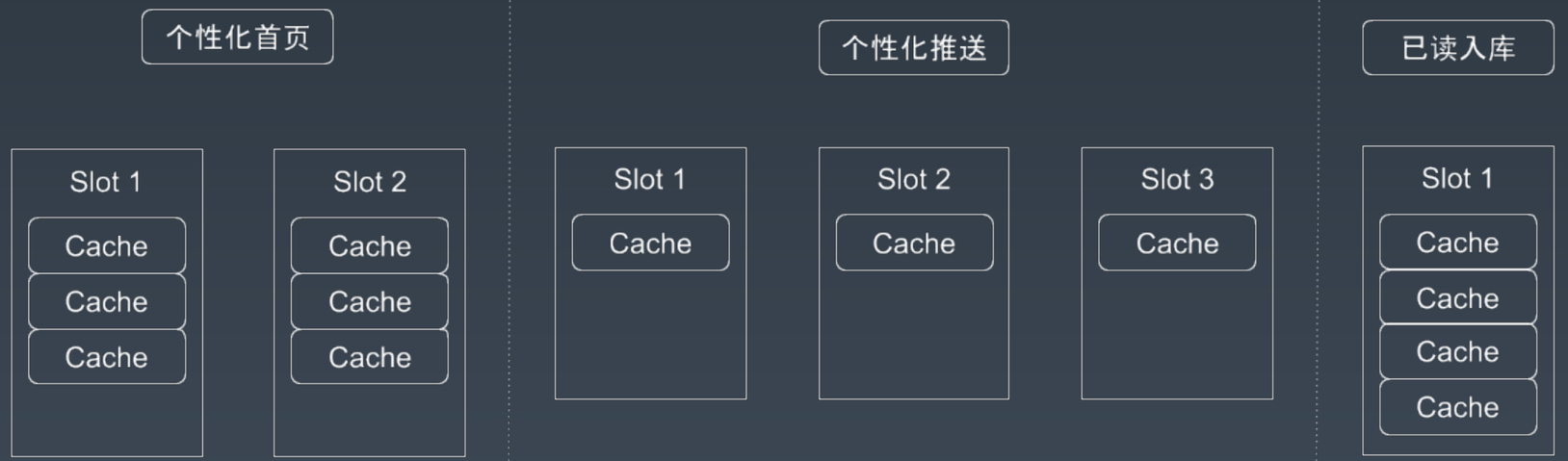
关键组件设计

- Cache
  - 多层缓冲
    - 时间维度 / 空间维度
    - 跨数据中心部署



关键组件设计

- Cache
  - 分组隔离
    - 离线在线隔离
    - 业务多租户隔离

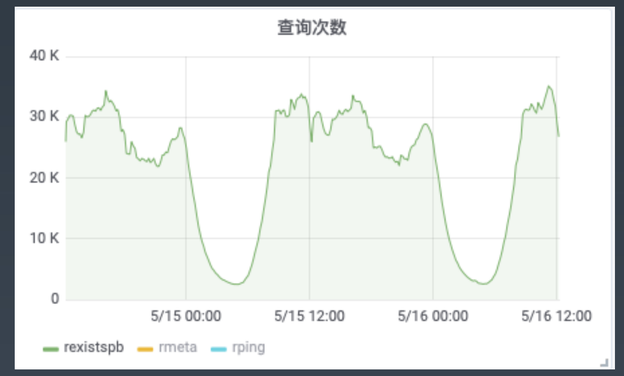


关键组件设计

- MySQL
  - TokuDB 引擎
  - 分库分表
  - MHA

性能指标

- 峰值每秒写入 40K+ 行
- 峰值每秒 30K+ 查询，查询 12M+ 文档
- 查询 P99 ≈ 25ms, P999 ≈ 50ms



核心痛点

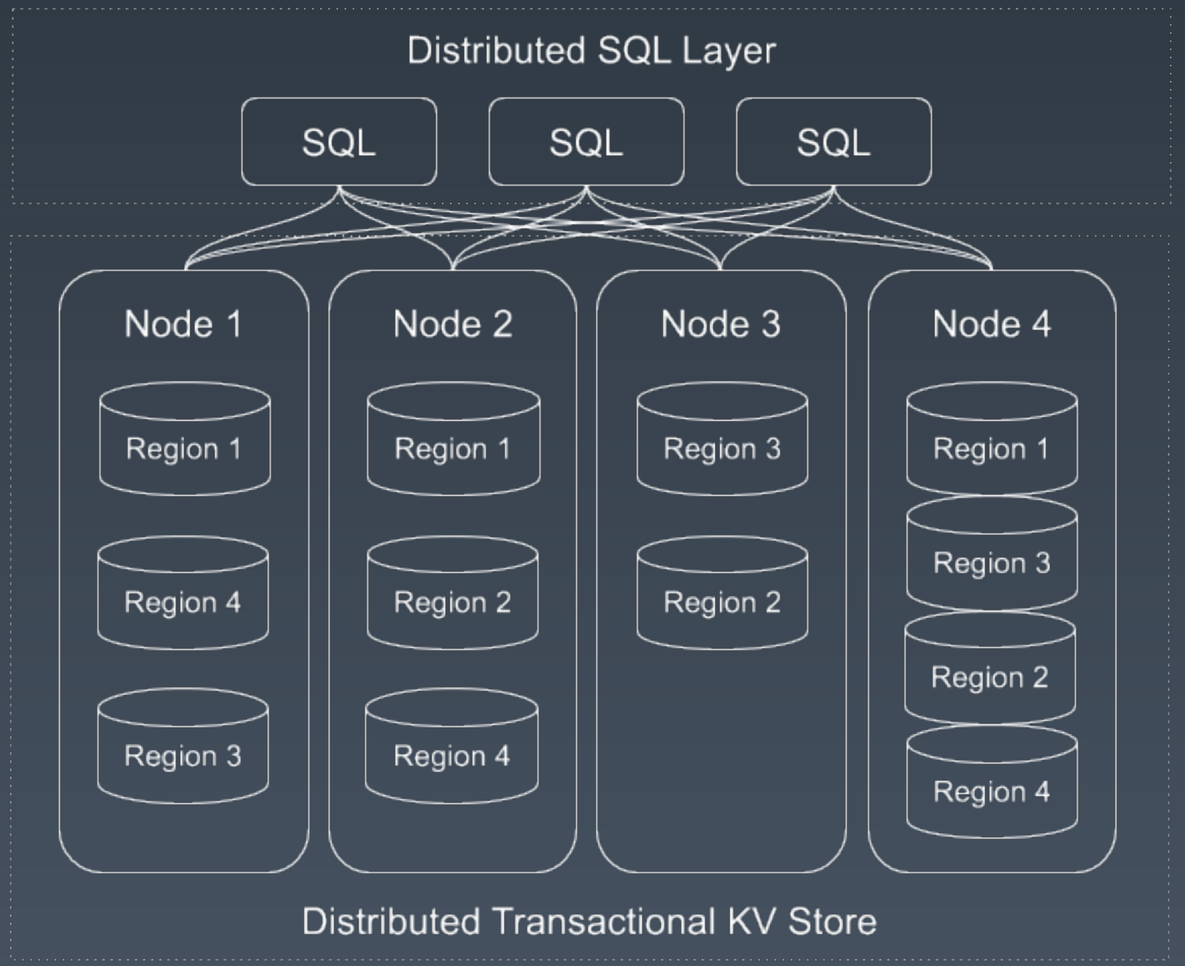
- MySQL 集群运维负担
  - 数据可靠性
  - 系统可用性
  - 系统扩展性
- 缺乏数据分析能力

目录

- 业务场景，技术挑战
- 早期方案，架构演进
- 核心组件，关键设计
- 全面云化，面向未来

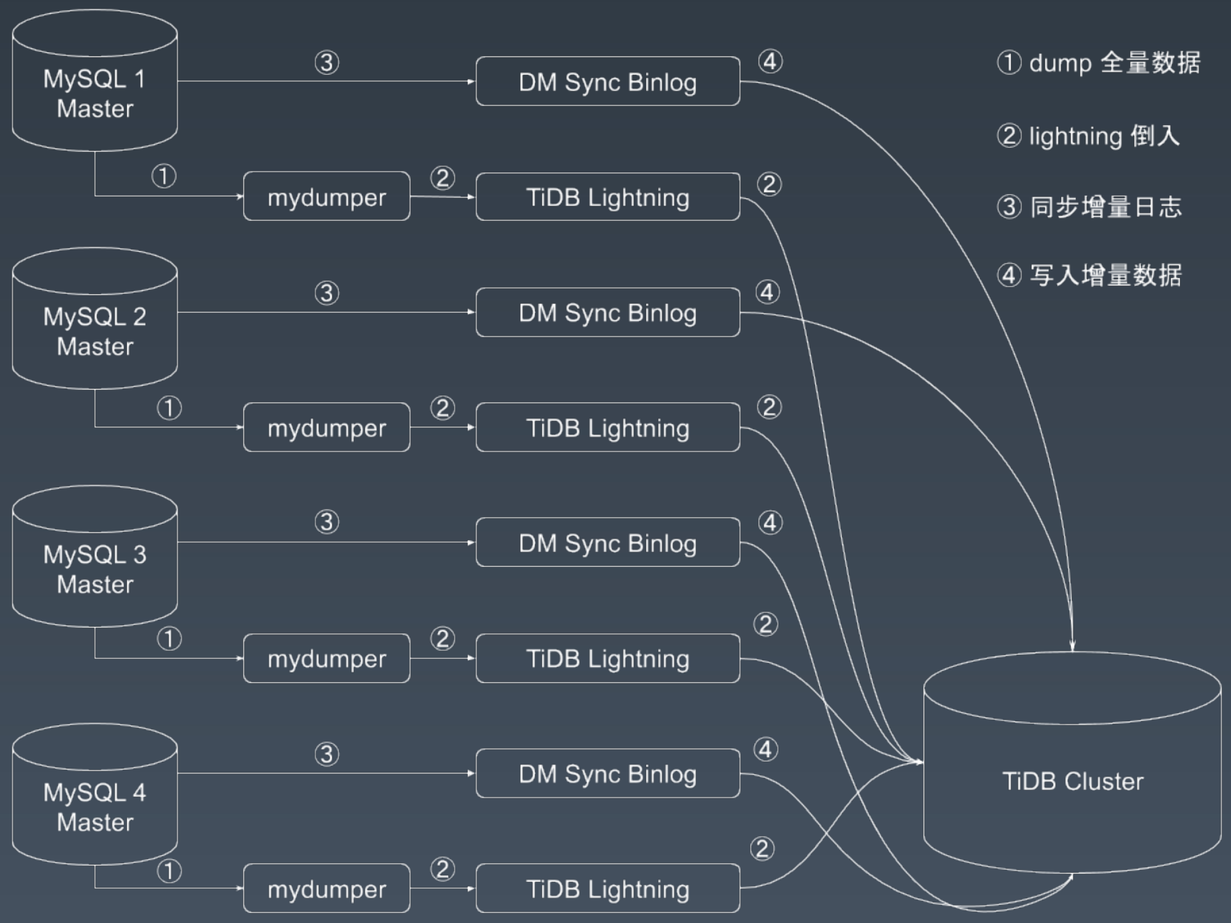
拥抱云原生

- 原生分布式数据库
  - CockroachDB
  - TiDB



拥抱云原生

- 迁移 TiDB
  - TiDB Lightning 全量数据导入
  - DM 增量数据同步



拥抱云原生

- 迁移 TiDB
  - 移植 MySQL Binlog 到 TiDB Binlog
  - 调整 TiDB 和 TiKV 满足已读的时延要求

迁移的经验教训

- 全量数据直接 TiDB 写入不可行
  - 逻辑写入预计耗时 1 个月
- TiDB Lightning 需独立部署
  - 资源消耗巨大
- TiDB Binlog 写入量过大导致 Kafka 过载
  - 调整 TiDB Binlog 按 Database 或 Table 选择分区
- 独立部署 TiDB 服务 Latency 敏感查询
  - 避免 Latency 敏感查询被其它任务抢占

迁移的经验教训

- Query Latency 比 MySQL 稍高
  - P99 45ms vs 25ms
  - P999 95ms vs 50ms
- 资源消耗比 MySQL 高
  - Master/Slave 两副本 vs Raft 三副本
  - 联合主键非聚簇索引的空间消耗
  - 计算存储分离网络要求高

TiDB Cluster

MySQL Cluster

TiDB Cluster

MySQL Cluster

迁移的收益

- 全系统高可用、规模随需扩展
- 计算/存储独立扩展
  - Cache / TiDB / TiKV 层可独立扩展
- TiDB 快速迭代持续改进
  - 3.0 rc.1 在测试环境性能表现优秀
- TiFlash 提供分析能力
  - 列存副本应对分析类查询

总结

- 理解业务
  - 对症下药
  - 抽象提炼
- 高可用
  - 故障感知
  - 自动恢复
  - 状态多副本
- 高性能
  - 弹性可伸缩
  - 分层去并发
- 拥抱新技术 Cloud Native from Ground Up

GTLC
GLOBAL
TECH LEADERSHIP
CONFERENCE
全球技术领导力峰会

Geekbang. 极客邦科技 | TGO 鲲鹏會

500+高端科技领导者与你一起探讨
技术、管理与商业那些事儿

2019年6月14–15日 | 上海圣诺亚皇冠假日酒店

扫码了解更多信息

THANKS!

QCon 10th