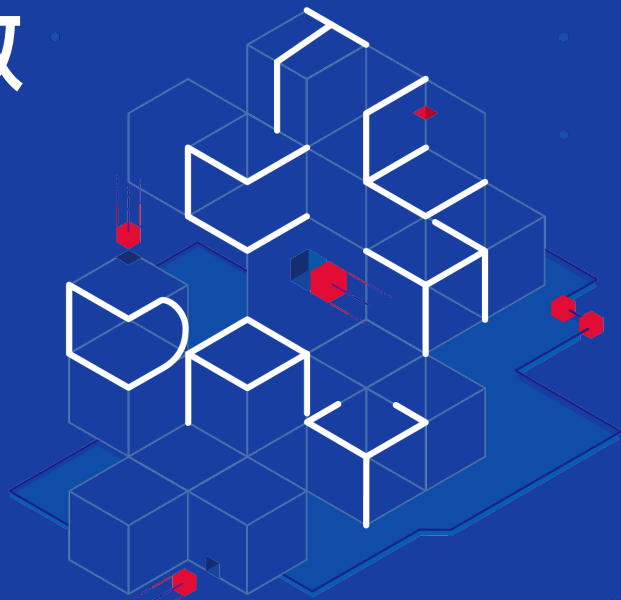
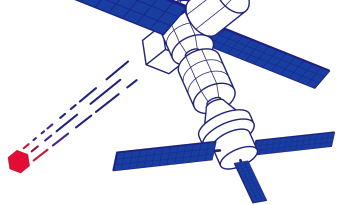


TiDB 在知乎万亿量级业务数据下的实践和挑战

孙晓光@知乎



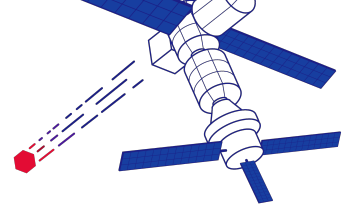


自我介绍

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

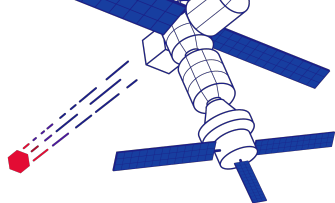
TiKV 项目 Committer



目录

- 业务场景
- 架构设计
- 关键组件
- All about TiDB

知乎：可信赖的问答社区



2.2 亿

用户数

38 万

话题

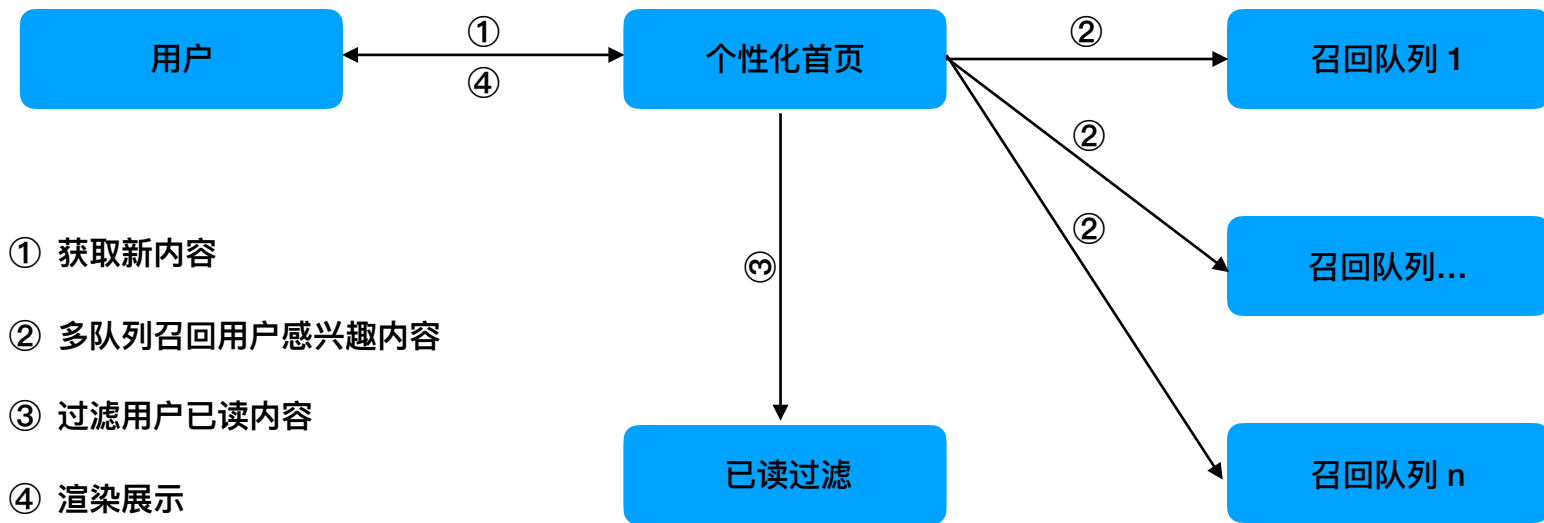
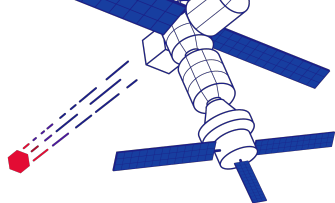
3000 万

问题

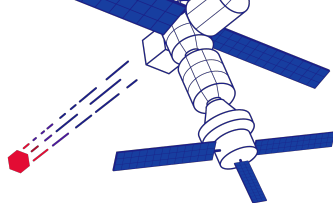
1.3 亿

回答

业务场景

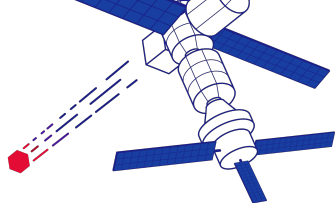


已读过滤示意图



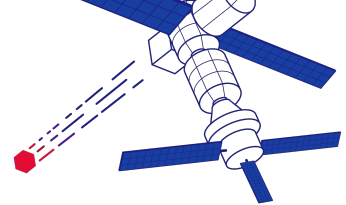
业务特点

- 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
 - 阅读历史保存三年
 - 近一万三千亿条历史记录



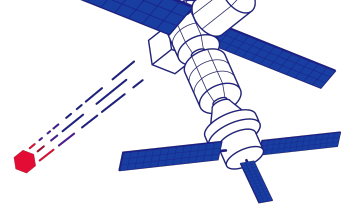
业务特点

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



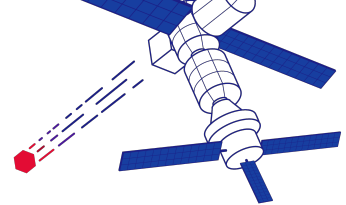
目录

- 业务场景
- 架构设计
- 关键组件
- All about TiDB



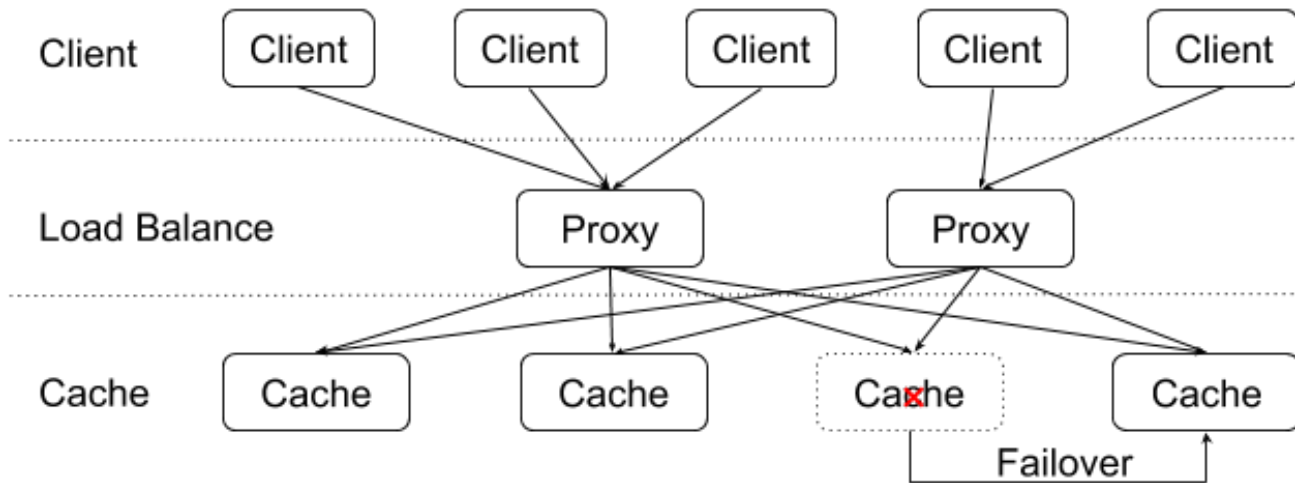
设计目标

- 高可用
- 高性能
- 易扩展



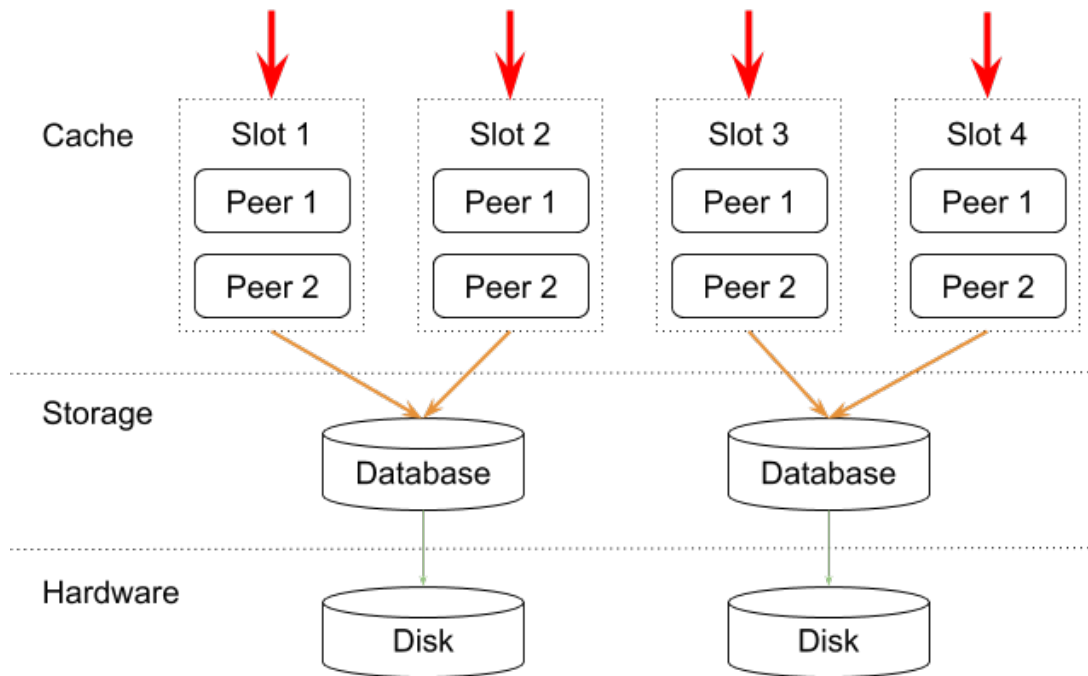
高可用

- 故障感知
- 自愈机制
- 隔离变化



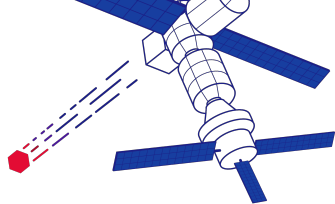
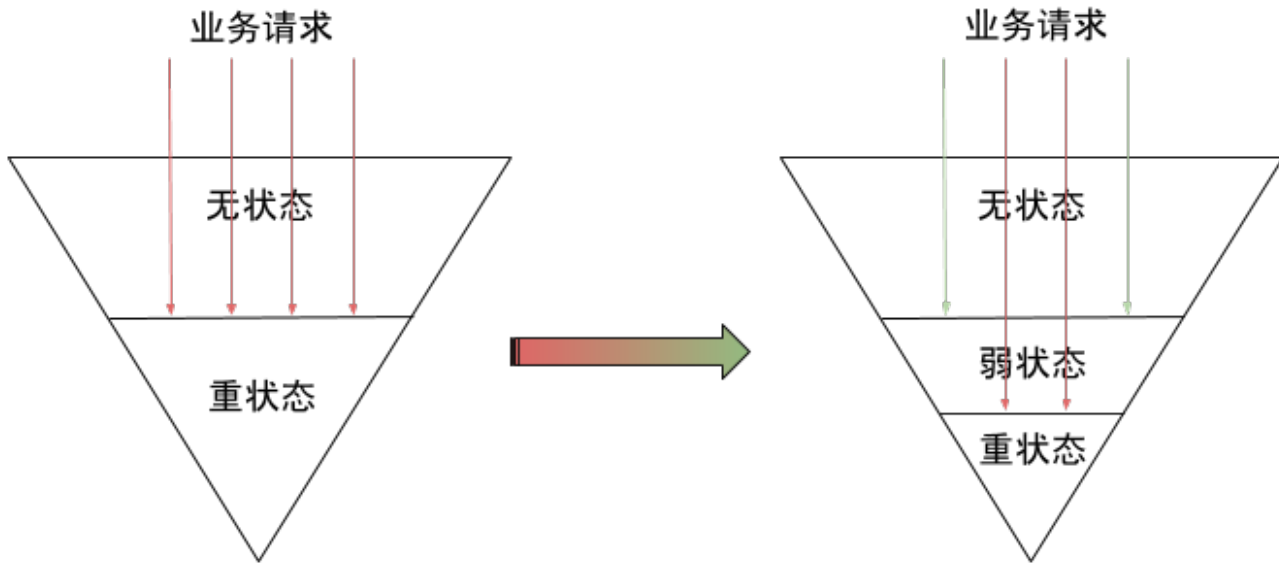
高性能

- 缓存拦截
- 副本扩展
- 压缩降压

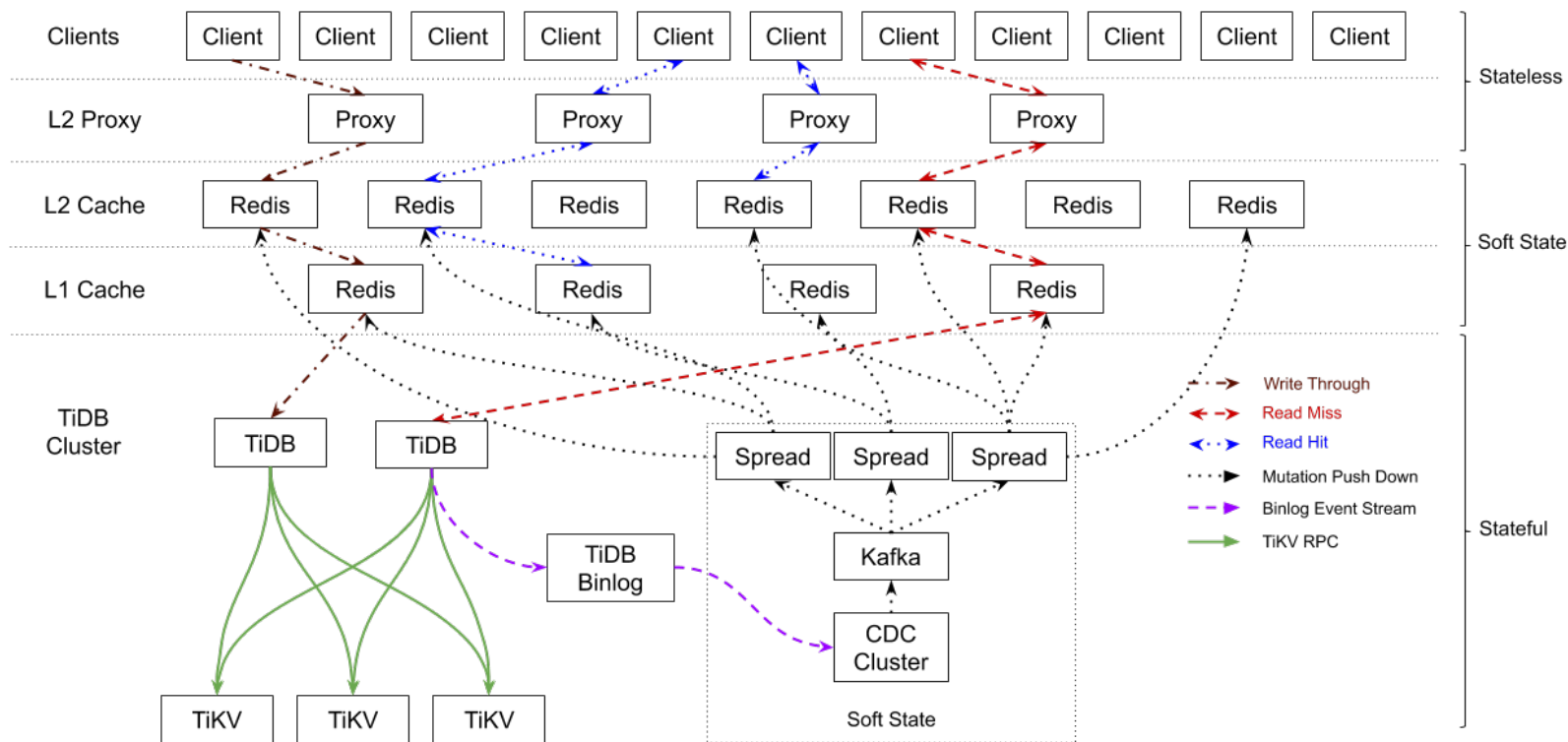
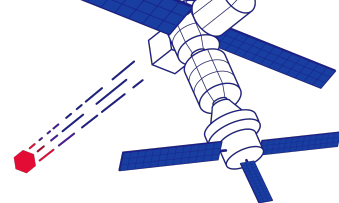


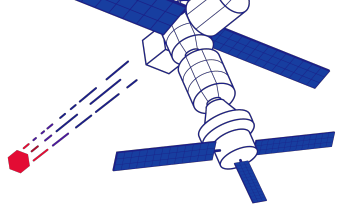
易扩展

- 无状态
- 弱状态
- 重状态



已读服务

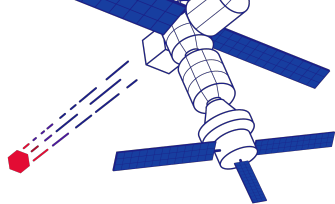




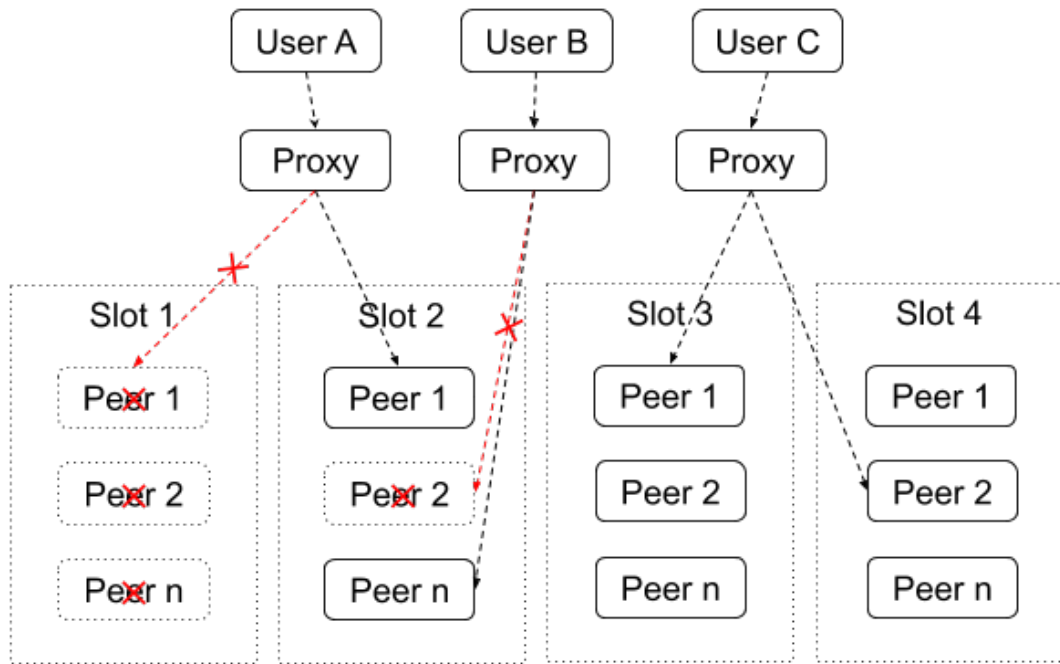
目录

- 业务场景
- 架构设计
- 关键组件
- All about TiDB

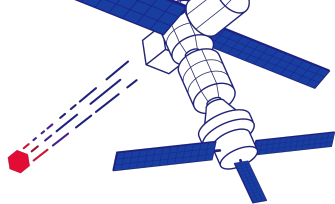
Proxy



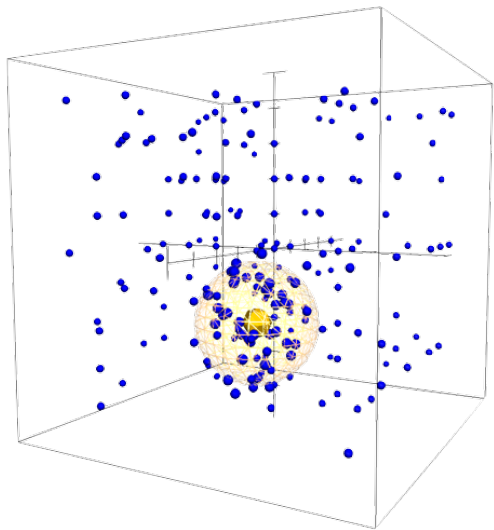
- Slot 内多副本高可用
- Slot 内会话粘滞
- Slot 间故障降级



Cache



- BloomFilter 增加缓冲密度



User

已读内容

文章 a

问题 b

回答 c

回答 d

Live e

.....

回答 n

Hash Functions

BloomFilter

0

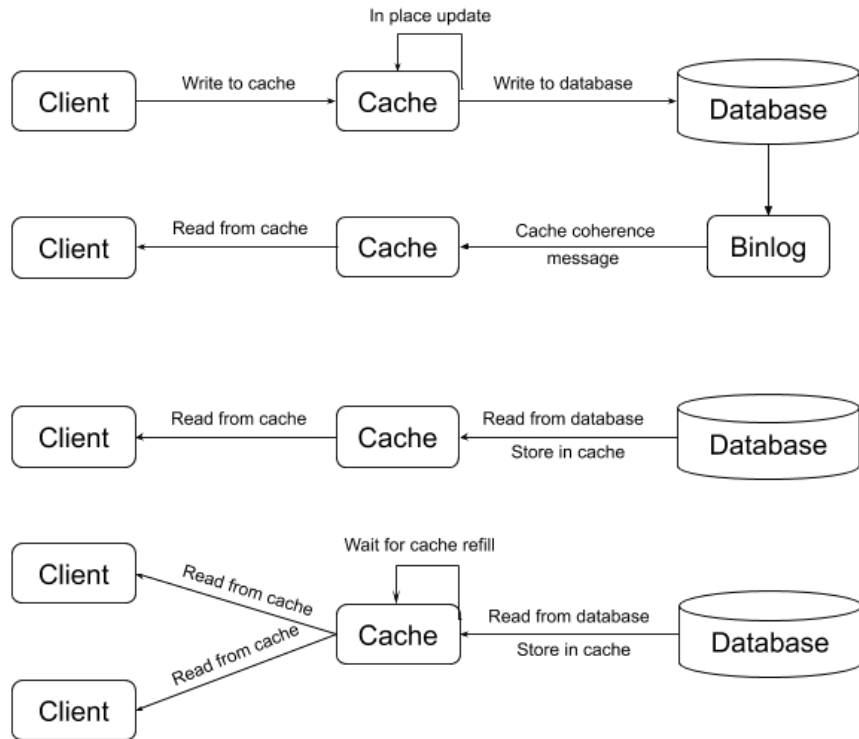
1

1

0

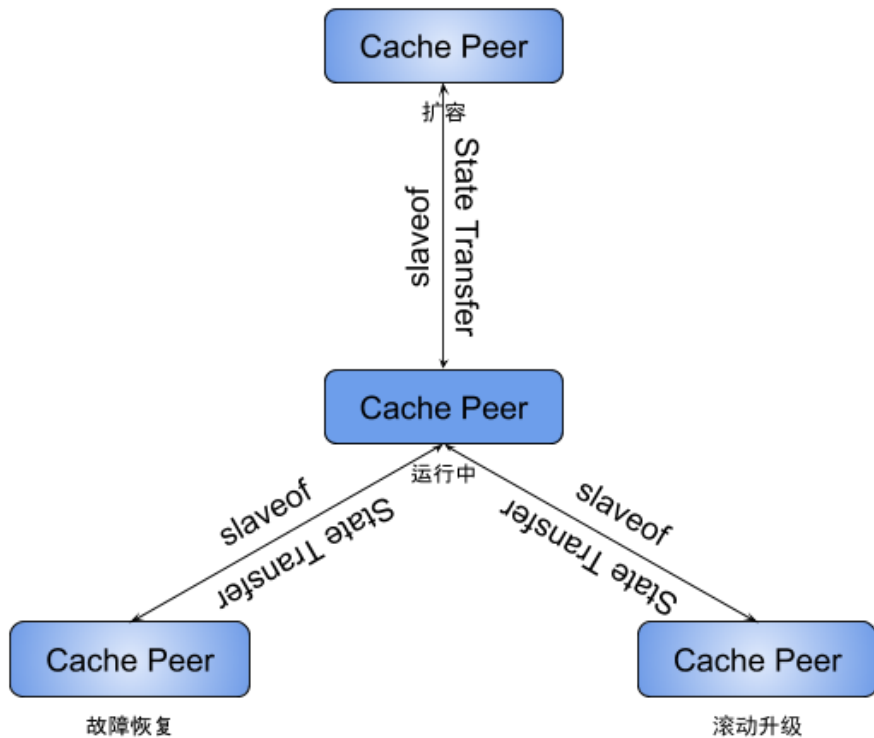
Cache

- 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

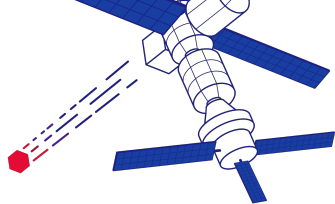


Cache

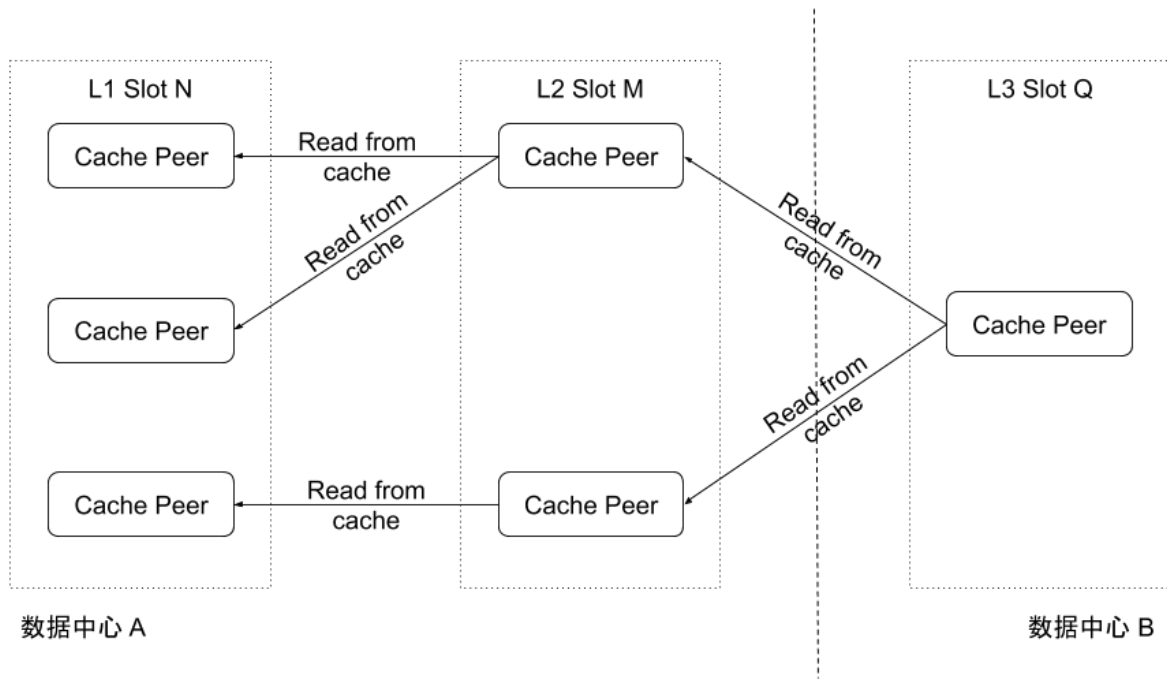
- 缓冲状态迁移
 - 平滑扩容
 - 平滑滚动升级
 - 故障快速恢复



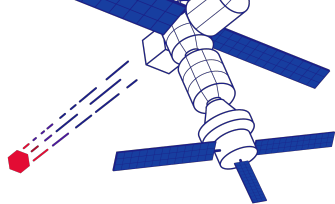
Cache



- 多层缓冲
 - 时间维度 / 空间维度
 - 跨数据中心部署

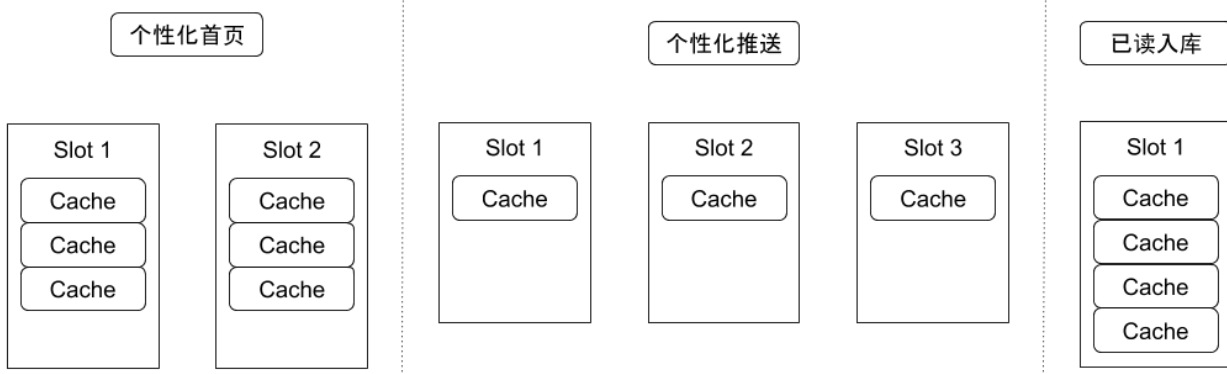


Cache



- 分组隔离

- 在线离线隔离
- 业务多租户隔离



Storage

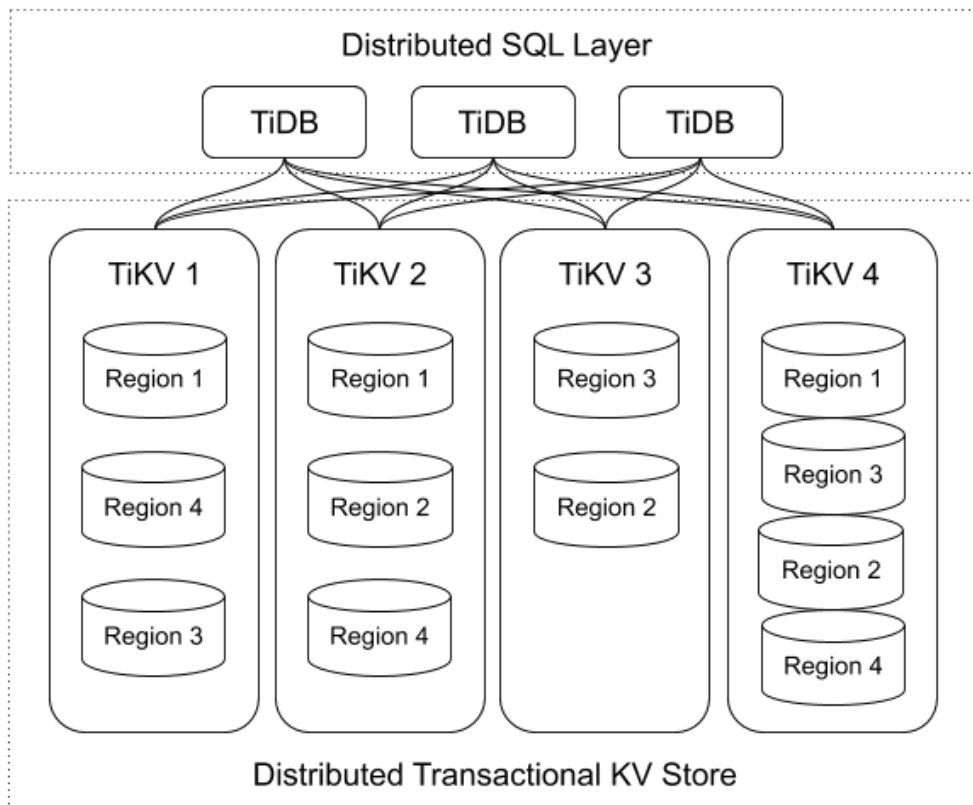
✗ MySQL 分库分表 + MHA

✗ 难以扩展

✓ TiDB

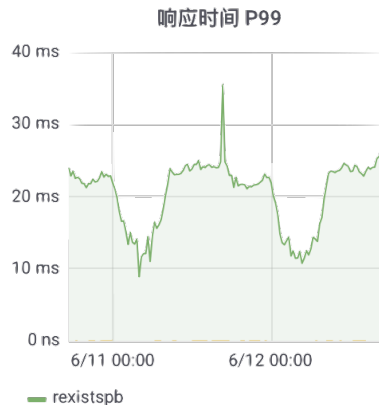
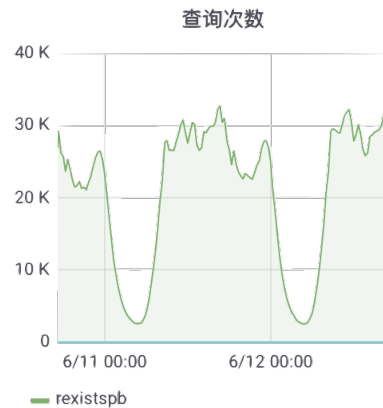
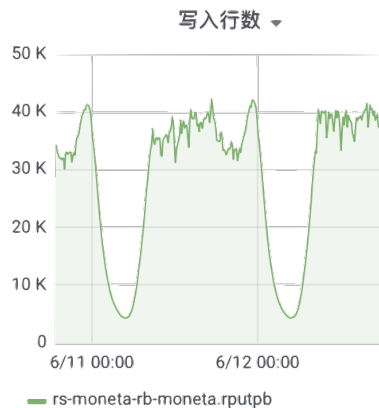
✓ 强一致高可用

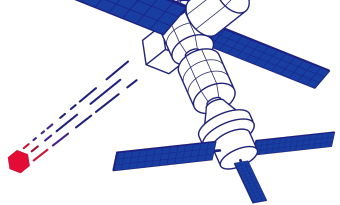
✓ 线性扩展



性能指标

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

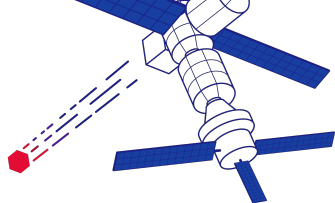




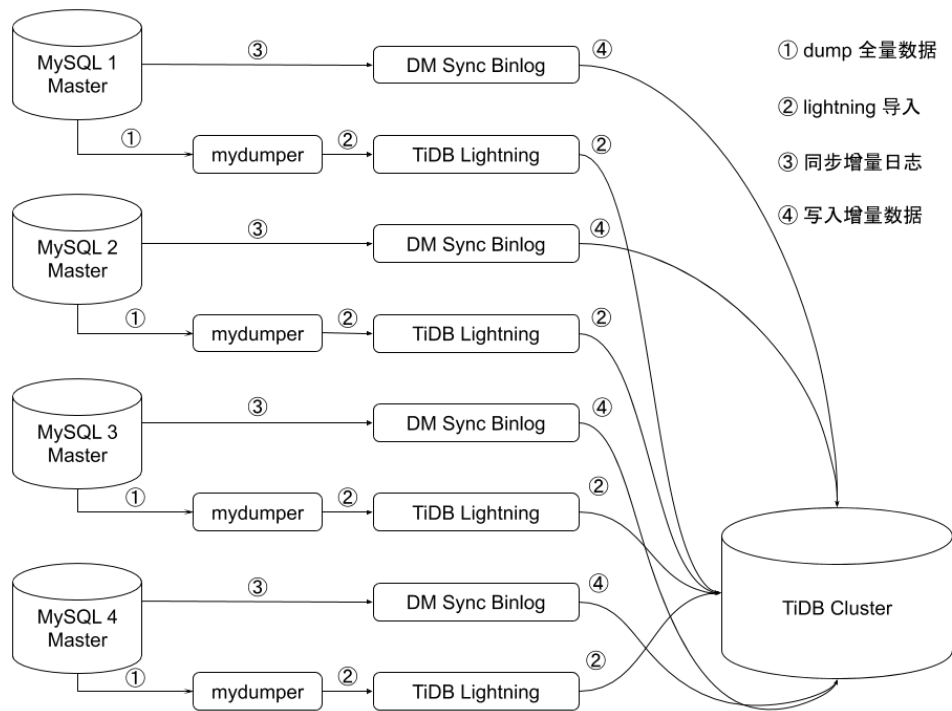
目录

- 业务场景
- 架构设计
- 关键组件
- All about TiDB

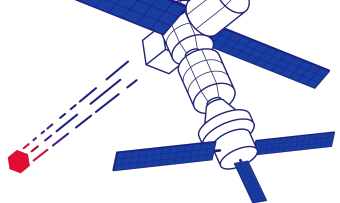
MySQL to TiDB



- TiDB Lightning 全量导入
 - 大数据量导入
 - 高投入高产出
- DM 数据增量同步

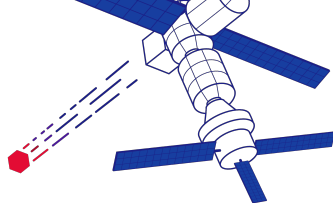


MySQL to TiDB

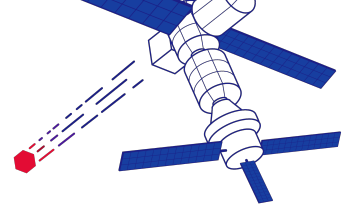


- 调优 TiDB 和 TiKV 满足苛刻时延要求
 - Latency 敏感 Query 部署独立 TiDB
 - 充分利用 SQL Hint
 - 低精度 TSO
 - 复用 Prepared Statement

MySQL to TiDB

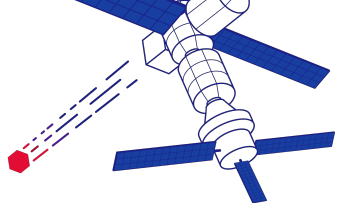


- 移植 MySQL Binlog 到 TiDB Binlog
 - 按 database/table 拆分 binlog 到多分区
 - drainer 持续优化
- 资源评估
 - master/slave 两副本 vs raft 三副本
 - 联合主键非聚簇索引的空间消耗
 - 计算存储分离网络要求高



| 关于 TiDB 3.0

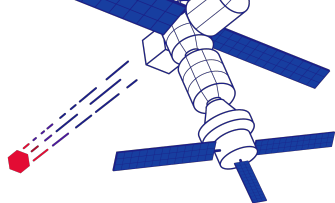
业务	日增行数	日增尺寸	查询时效	保存时效
已读	30 亿	140 GB	3 年	3 年
反作弊	80 亿	1.5 TB	2 天	6 年



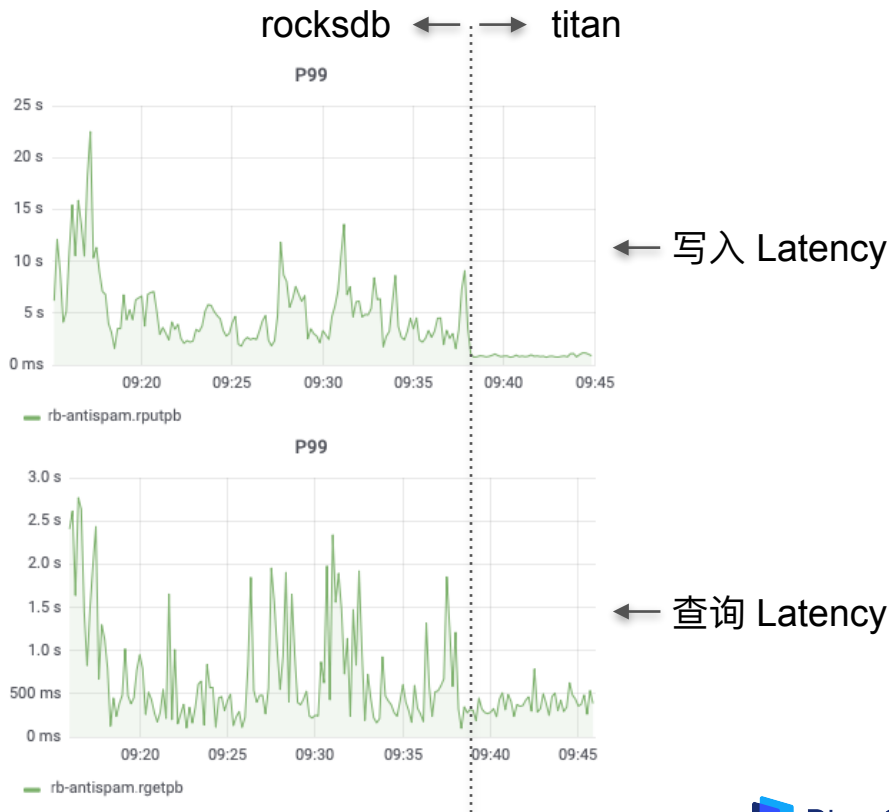
| 关于 TiDB 3.0

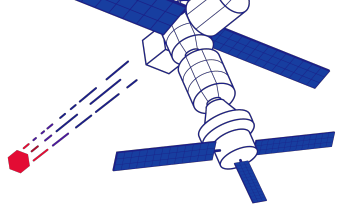
- 已读
 - 高写入
 - gRPC Batch Message
 - 多线程 Raft Store
 - Latency: Plan Management
 - 分析能力: TiFlash

关于 TiDB 3.0



- 反作弊
 - 高写入
 - gRPC Batch Message
 - 多线程 Raft Store
 - 大记录: Titan 存储引擎
 - 查询时效: Table Partition





总结

- 理解业务
 - 对症下药
 - 抽象提炼
- 参与社区，贡献社区
- 拥抱新技术 Cloud Native from Ground Up

Thank You !

