微信PaxosStore简介

主讲人：曾楚伟
eddyzeng@tencent.com

目录

摘要

- PaxosStore是什么？

  一个在跨园区数据中心间同步复制，提供灵活的数据模式和访问接口并支持单表亿行，具备快速伸缩能力，低延迟低成本，强一致和高可用的分布式存储系统。

- PaxosStore特点：
  - 无租约Paxos工程化，多主多写，高可用
  - 针对业务特性优化，合并整体优化成本15+%
  - 同一容灾、迁移框架下，支持多种插件化存储引擎、亿行大表
  - 快速伸缩能力，基于反馈的自适应迁移系统

摘要

- PaxosStore广泛支持微信在线应用

摘要

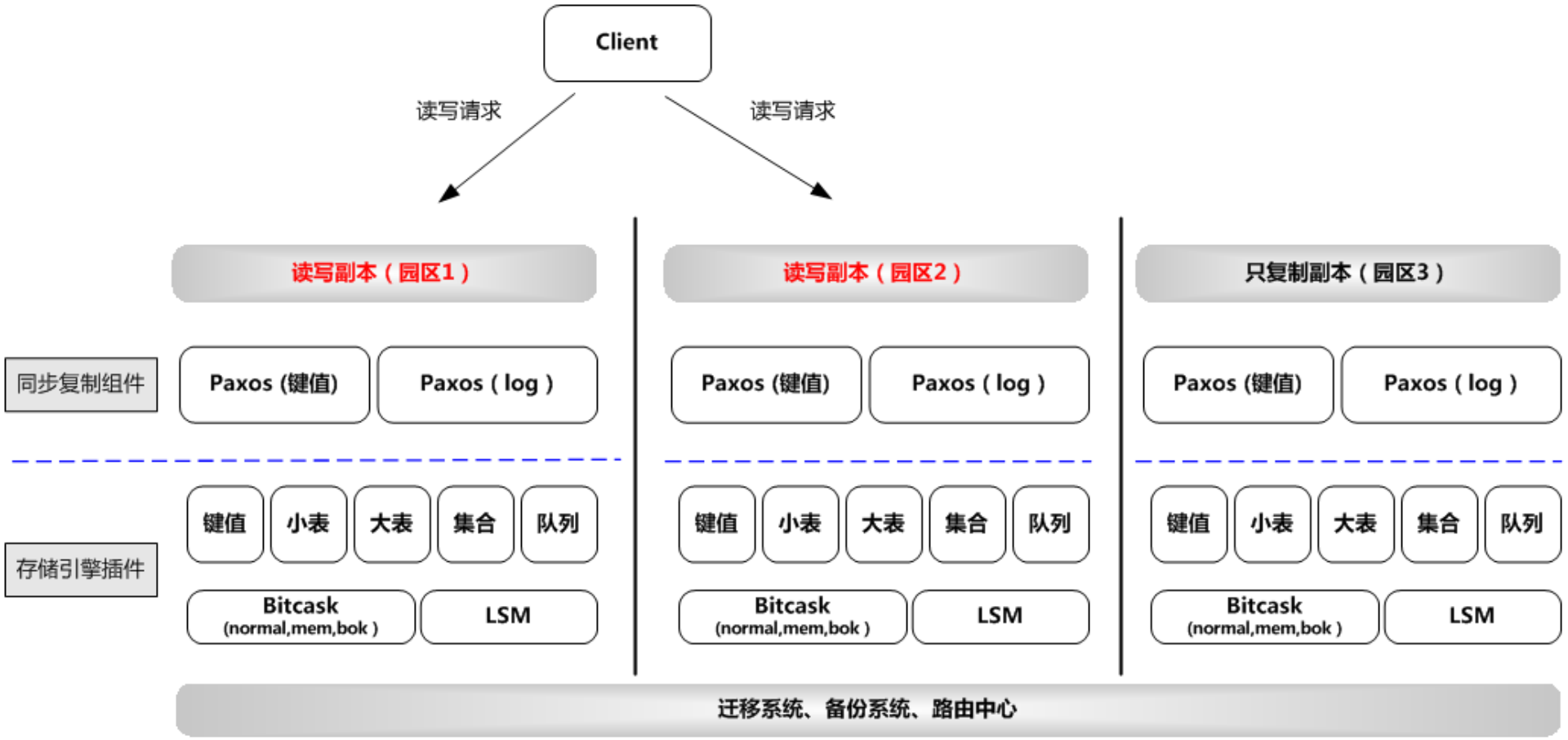
- PaxosStore部署情况：
  - 微信内部广泛部署、数千台机器
  - 数万亿/天的读写量、峰值1亿+/秒
  - PB级的结构化数据、全球多个数据中心
  - 世界上基于Paxos最大的线上存储系统？

摘要

- 微信为什么需要PaxosStore？
  - 更高可用性要求：
    频繁单机故障：近200台/月，机器数1万+
    偶发网络分区：跨园区部署
    海量用户
  - 业务快速增长，成本优化需求迫切
  - 大表的需求：千万行、丰富API、扩展性、一致性视图

目录

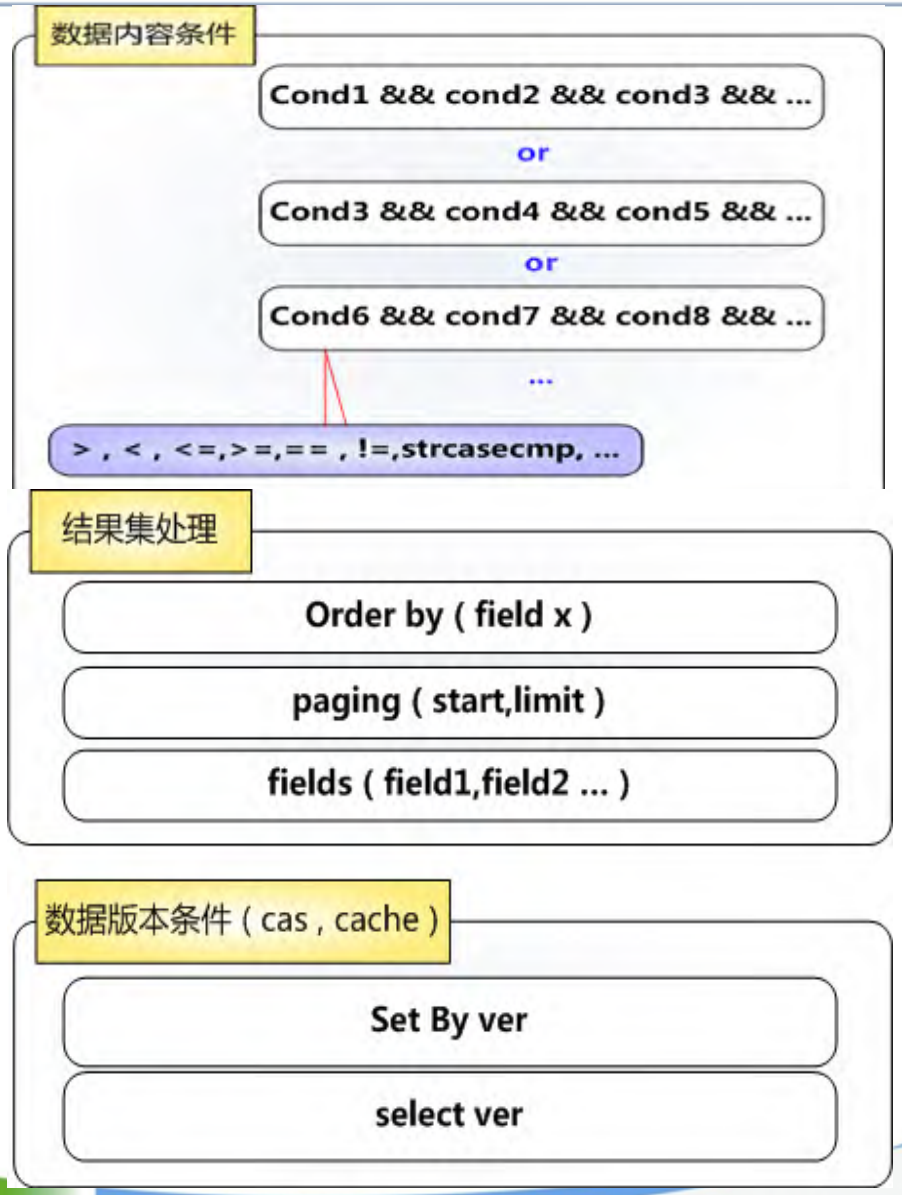
功能介绍：总体架构


Client
读写请求
读写请求
读写副本（园区1）
读写副本（园区2）
只复制副本（园区3）
同步复制组件
Paxos (键值)
Paxos (log)
Paxos (键值)
Paxos (log)
Paxos (键值)
Paxos (log)
存储引擎插件
键值
小表
大表
集合
队列
键值
小表
大表
集合
队列
键值
小表
大表
集合
队列
Bitcask
(normal,mem,bok)
LSM
Bitcask
(normal,mem,bok)
LSM
Bitcask
(normal,mem,bok)
LSM
迁移系统、备份系统、路由中心

功能介绍：数据模型

- **丰富数据类型，为业务快速开发提供保障**
  - 键值
  - 队列
  - 集合
  - 二维小表
  - 大表

功能介绍：数据模型



功能介绍：数据备份

- 冷备：定期完整备份
- 热备：事务日志的增量备份

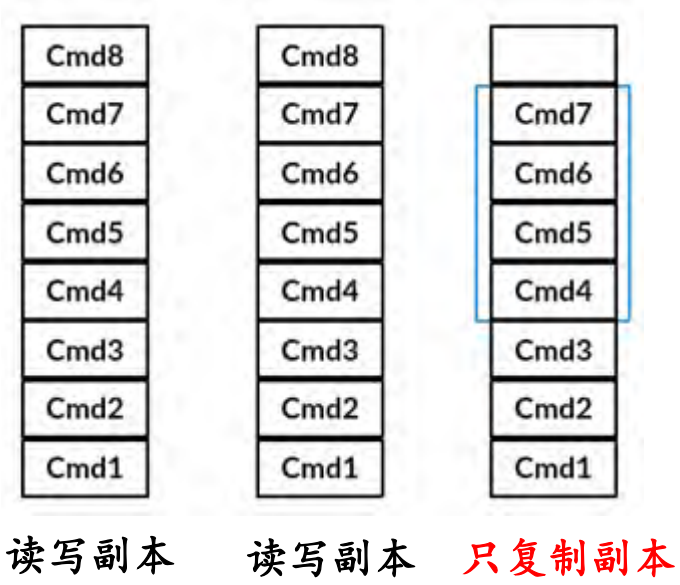
目录

可用性：复制策略

- *复制策略*
  - 异步主/备复制
  - *同步主/备复制*
  - 乐观复制
- Paxos：容错、无主的一致性算法
  - 同步复制操作日志
  - 多主多写
  - 一致性视图、ACID语义
  - 多个复制日志提升吞吐量和可用性

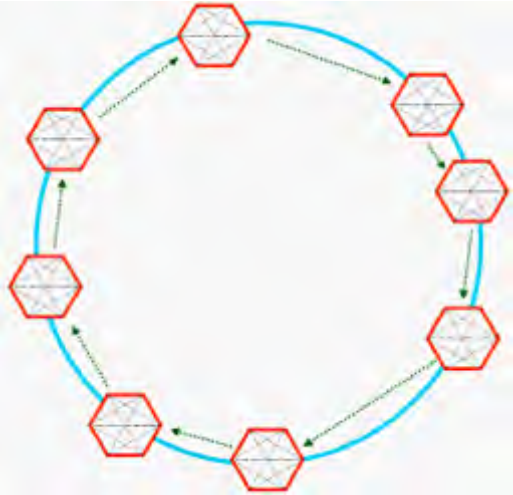
可用性：性能优化

- *Paxos协议优化：预授权*
- 网络IO按IP聚合投递
- 针对只复制副本，延迟批量Commit（削峰填谷）



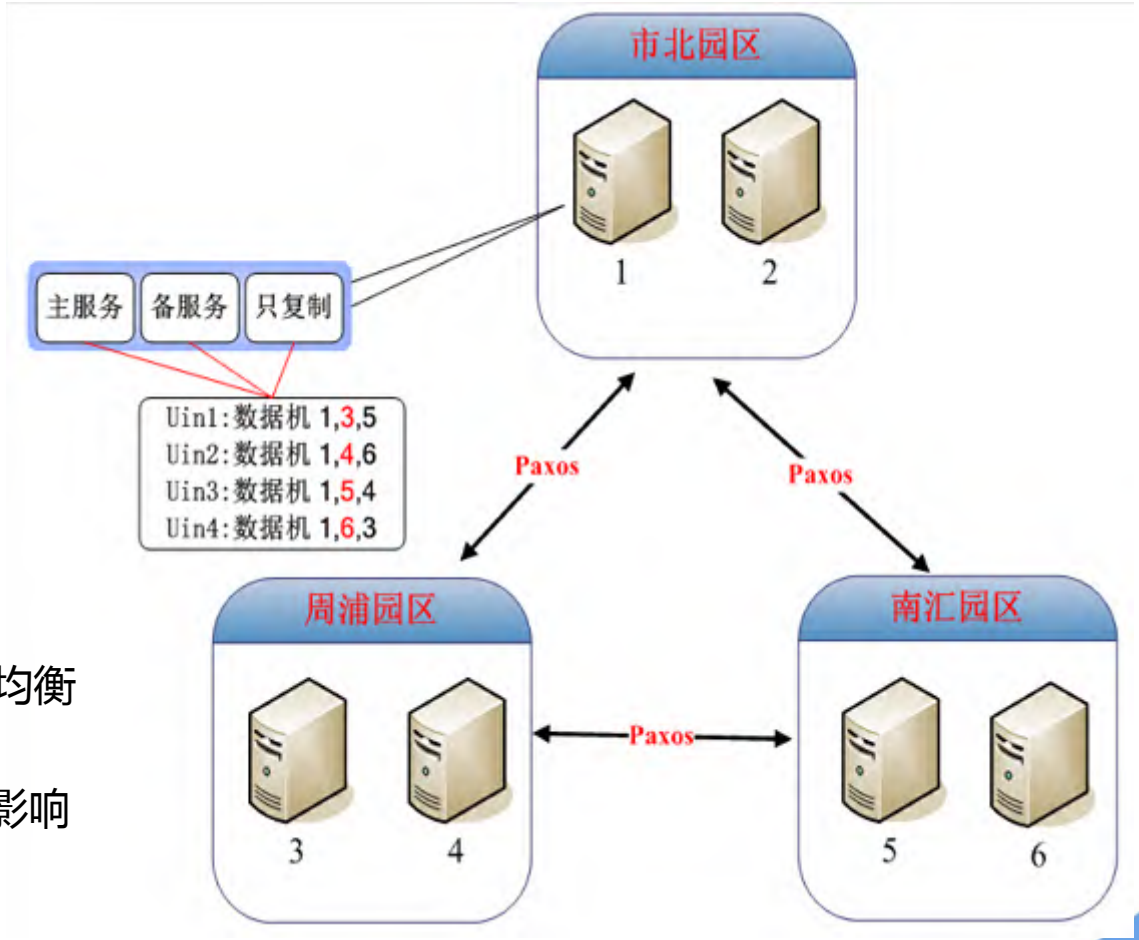
目录

可扩展性：数据分布



- *按Set划分，Set间一致性Hash均匀分布*
- *上下层间调用按Set对齐，隔离单Set故障和提升批量RPC效果*

可扩展性：数据分布

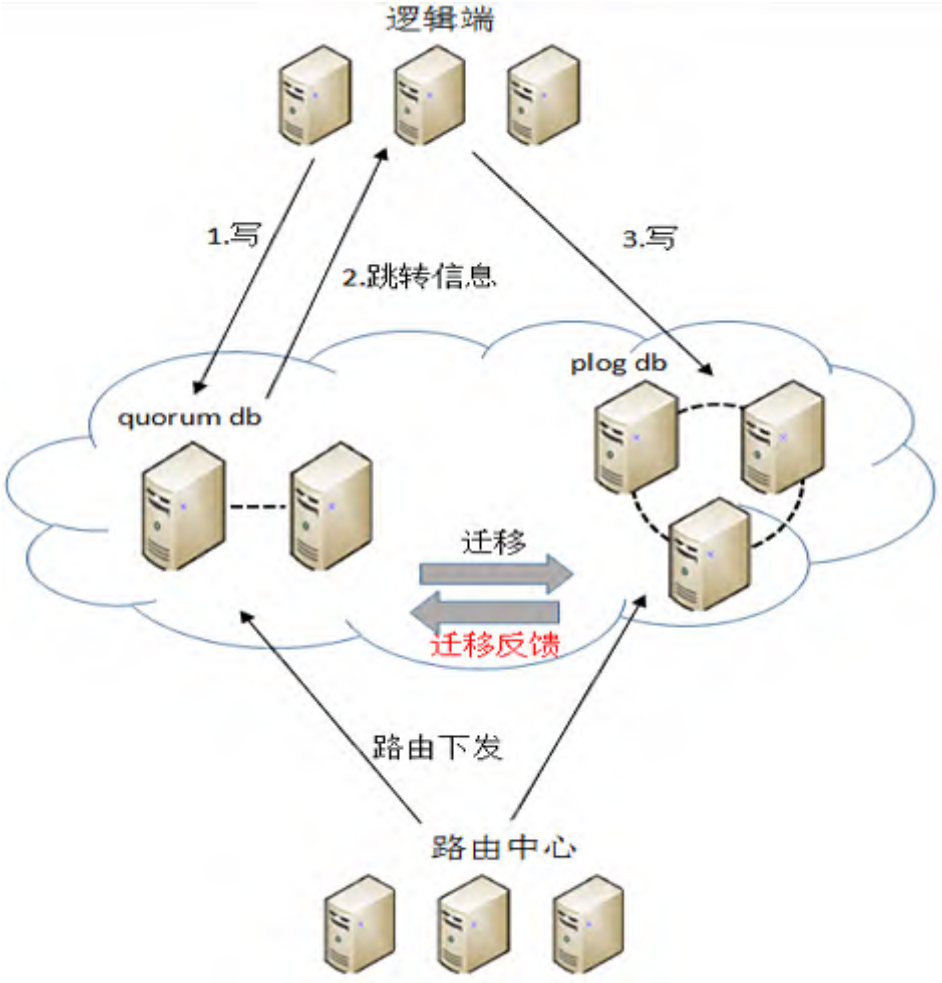


- Set内三园区互为主备， 负载均衡
- 单机/园区分级容灾
- 单机 /单园区故障，服务不受影响
- 读就近访问

可扩展性：通用方案

基于反馈的自适应迁移/扩容系统

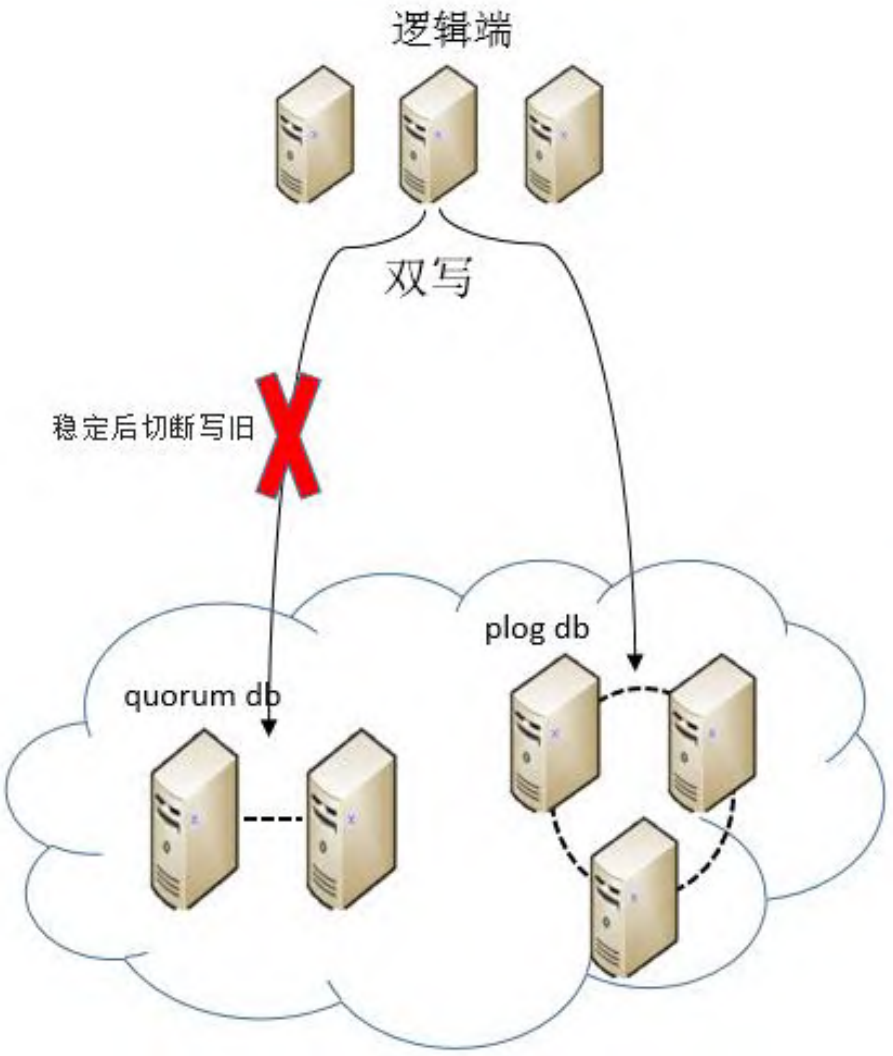
- 根据迁入机器、迁出机器资源使用情况，自动控制迁移速度
- 数据双写，支持快速回退



可扩展性：特殊方案

消息存储迁移/扩容方案

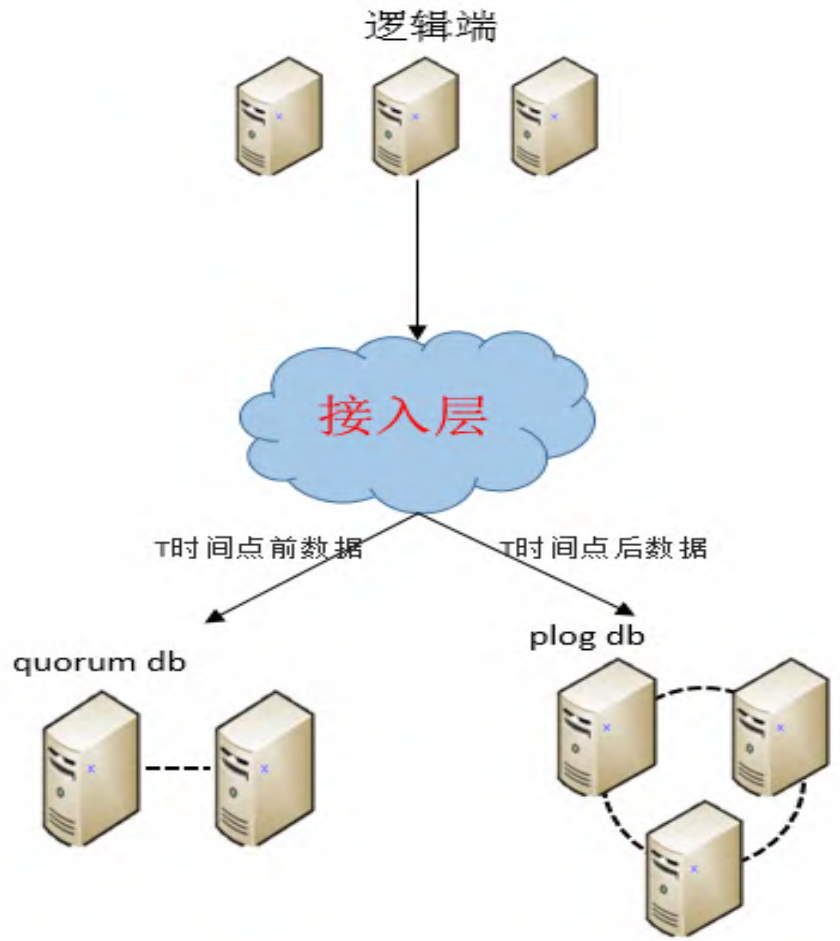
- 逻辑端双写
- 数据失效后，切新集群读写
- 双写期间支持回退



可扩展性：特殊方案

朋友圈存储迁移/扩容方案

- 服务器双写
- 不涉及旧数据的迁移
- 根据数据的时间分发key



目录

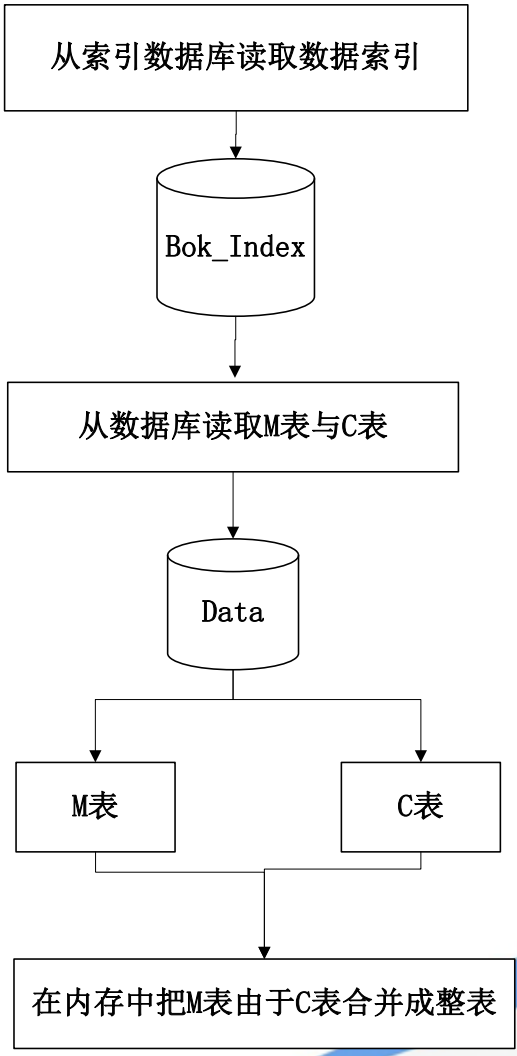
存储引擎

同一容灾/迁移架构下的多种存储引擎，应对不同业务模型

- **多种存储引擎**
  - 原始bitcask、
  - 内存bitcask（全内存方案）
  - 双层bitcask（突破单机key量限制）
  - 基于LSM的存储引擎：消息存储系统、通用大表系统

存储引擎：大表

- **背景**
  - Kvsvr-Table 小表基于bitcask/双层bitcask
  - 读写需要获取全表数据（MC表合并）
  - 表大小20M限制
- **挑战：**
  - 高性能大表存储（支持亿行）
- **方案：**
  - 基于Lsm构建通用大表系统（支持二级索引）



目录

基于PaxosStore的存储案例: 消息

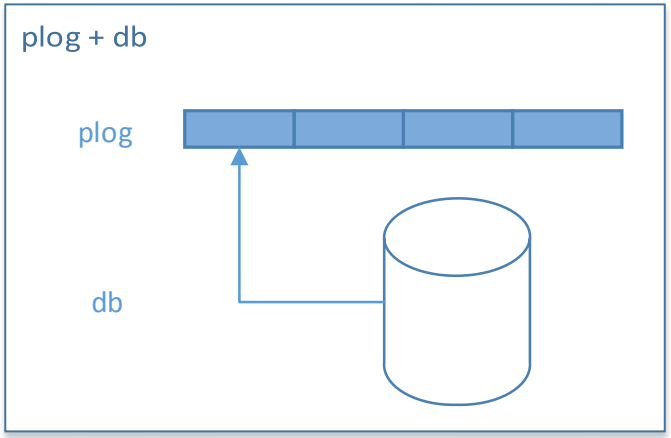
- 背景
  - 核心数据：用户消息
  - 1000+台机器，PB级数据
  - 消息写量：*亿级/天
  - 存储转发机制
- 优化
  - raid10 * 2 · no raid *3
  - 单副本改造：群消息按群聚集，群用户存索引
  - 索引强一致读写，单副本可扩展
  - 红包消息分离，利用集群闲置资源
  - 快消型日志，DB支持增量恢复

基于PaxosStore的存储案例: 朋友圈

- 背景
  - 核心数据：朋友圈正文、朋友圈评论
  - Key为64位ID，包含毫秒级时间戳
- 优化
  - 基于时间快速升级到PaxosStore
    - 1天: 70%+访问切至新存储，数据量占比0.3%
    - 1月: 90%+访问切至新存储，数据量占比6.5%
  - 冷热分离架构
    - PB级存量数据
    - 热集群+冷集群
  - 增加缓存层和汇聚层

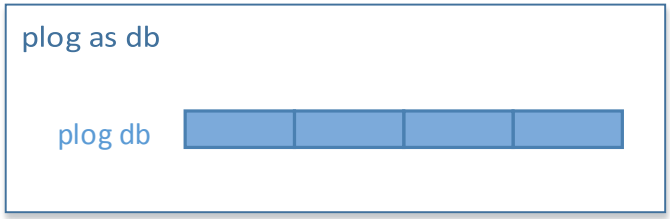
基于PaxosStore的存储案例：微信帐号

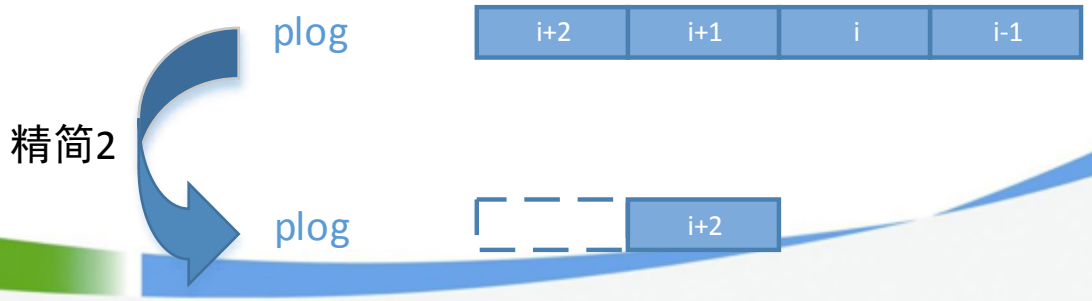
- *背景*
  - 微信帐号
  - *Key-Value场景*



- *优化*
  - 复制日志作为 DB
  - *只保留最新Log Entry*

精简1





目录

微信PaxosStore简介

Q&A