



教育经历

浙江大学 硕士 软件工程

2021.09-2024.06

浙江工业大学 本科 信息与计算科学

2017.09-2021.07

工作经历

滴滴出行 实习生 基础平台-文件存储

2022.11-2024.05

滴滴出行 后端开发工程师 基础平台-文件存储

2024.07-2025.10

部门介绍: 部门产品为分布式文件存储服务, 为公司内部部门提供存储能力

离线 EC 迁移

2024.07-2024.10

背景: 早期数据由多副本进行冗余存储, 存在资源浪费、读取无法条带化的问题, 需要高效可靠地将数据迁移至 EC (Erasure Coding, 纠删码) 存储

工作: 负责开发离线数据迁移模块, 按分组粒度将数据从多副本迁移至 EC

挑战: 保证单分组十亿级别小文件的高效迁移, 确保不漏迁、不错迁

解决: 采用类似 Map-Reduce 的架构设计迁移模块: 多个 Master 将待迁移数据 Hash 至多个 Migrate 中间表, 多个 Worker 抢占任务执行迁移
将任务拆分为数据上传、校验、元数据更新等流水线阶段, 提高并发与吞吐率的同时保证正确性

思考: 为快速上线, 部分功能未完备, 例如缺乏前后台限流管理、动态扩缩容与幂等性校验等

BS2.0 存储引擎

2025.02-2025.09

背景: BS 是文件存储底座, 负责存储文件二进制数据。老版本基于小集群设计, 每个集群由独立节点管理数据, 运维成本高且磁盘利用率低。BS2.0 采用大集群设计, 所有数据由调度框架 BG 和注册中心 RS 统一管理, 并新增更多数据保障功能

工作: 设计并开发 BS 后台任务, 包含数据正确性任务 (数据校验、数据修复) 以及并发垃圾回收任务
为以上后台任务设计调度框架 BG, 包含任务创建、查询、异常任务重启、限流等操作

挑战: 后台任务限流, 需兼顾前台延迟与后台吞吐
设计高效的任务调度框架, 调度数十亿个数据副本的后台任务
实现高效的垃圾回收策略

解决: 后台任务限流从按磁盘类型的硬限制, 演进为独立计算前后台流量、定期扫描自动调整前后台限流比例, 同时通过预测前台延迟, 在高负载时拒绝后台请求
注册中心 RS 通过将数据副本分区的方式减少单次调度处理的数据量, 多个 BG 节点通过抢占分区分布式锁获取调度权限, 从 RS 拉取集群信息进行检查并生成调度 OP, 对应的 BG 执行调度 OP
RS 通过 BS 心跳检查副本数是否完整, 减轻数据修复过程中 BG 扫描压力
垃圾回收阶段进行两阶段回收, 尽可能减少对写入的影响

项目经历

Go实现消息队列<https://github.com/issac1998/go-queue>

2025.07-至今

项目描述: 使用 Go 实现的强一致持久化消息队列, 消息通过文件形式落盘, 同一 Topic 的不同 Partition 由 Raft Group 管理, 持久化在多台机器上
由 Controller 负责 Partition 分配与重平衡
实现 Exactly-Once、Consumer Group、Dead Letter、延时队列、顺序消息、生产者/消费者事务, 同时支持消息压缩与去重等功能

秒杀系统设计<https://github.com/issac1998/mall>

2025.09-至今

项目描述: 秒杀系统的后端设计, 涵盖用户登录、下单、支付场景, 实现秒杀库存扣减、订单创建流程
使用消息队列实现削峰, 并通过延时消息实现过期订单的检查
实现本地及基于 Redis 的全局限流, 包含用户、IP、活动、总请求量等多个维度
提供排队、抽奖、限流、报错四种降级策略, 后台定期根据活动库存及机器状态选择降级方案
提供完善的 K6 测试脚本, 在容器内对不同方案进行压测及性能分析
实现多种订单库存扣减策略: Redis 预减库存定期同步至 MySQL / Redis 预减库存通过乐观锁扣减 MySQL / 服务器本地缓存预减
秒杀商品数量较少时, 考虑将一个活动拆分成多个 Shard 以减轻Redis单实例压力