



周航航

📞 18822157735 📧 2603167476@qq.com 🌐 <https://github.com/LEFTX1>

💎 在读 🎓 意向职位：服务端实习生 📍 北京-天津-上海-深圳

教育经历

天津工业大学 (双一流) 计算机科学与技术 本科

2022-09 ~ 至今

专业技能

- 后端技术: 熟练掌握 Golang, 深入理解其并发模型 (Goroutine/Channel/Context), GMP 调度, GC 机制; 熟悉计算机网络 (HTTP/TCP) 与 常见数据结构; 了解 Java 基础及 springboot 框架。
- 数据存储与缓存: 熟悉 MySQL, 掌握 InnoDB 核心原理 (B+Tree, ACID, MVCC), 具备分库分表设计经验; 熟悉 Redis, 深入理解其核心数据结构、持久化 (RDB/AOF), 理解高可用方案 (哨兵/Cluster) 实现原理, 掌握缓存设计模式 (防穿透/雪崩/击穿) 与分布式锁 (RedLock) 原理。

项目经历

Go-Link

2024-11 ~ 2025-03

go-zero+mysql+redis+etcd+rabbitmq

<https://www.laoyujianli.com/>

项目介绍：为提升内部营销（短信 & App Push）活动效果与追踪能力，构建了一套短链接服务平台，提供链接创建、管理、高性能跳转服务，并生成详细的点击统计报表以供分析。

- 高性能短链接生成：采用雪花算法，并通过本地缓存记录 LastTimeStamp，利用序列号缓解时钟回拨id重复问题。与8位随机字符串/MurmurHash结合布隆过滤器去重的方案相比，将API平均延迟降低了约45%（从11ms降至6ms），P99延迟降低了约37%（从16ms降至10ms），显著提升了在短链接创建数量日渐增长的业务背景下的响应能力和可靠性。
- 解耦统计与短链接跳转：将短链接跳转时的统计数据操作异步化，通过 Goroutine 将统计任务推送到 RabbitMQ 消息队列，由后台多个 Goroutine 消费者异步、批量地将数据写入数据库，确保短链接跳转接口性能仅依赖 Redis 的1.5ms响应。
- 提升读路径效率与抗压能力：针对短链接跳转场景，在入口处引入布隆过滤器，高效拦截大量无效的请求，减少后端压力。结合空值缓存策略，有效防御请求不存在链接的缓存穿透攻击。针对热点短链接可能引发的缓存击穿问题，引入 Singleflight 机制，合并对同一资源的并发数据库请求，显著降低DB负载，确保高并发下的服务稳定与低延迟。
- 采用MySQL数据库表水平分片策略：将核心业务表（链接表、跳转表、用户表、分组表）按CRC32 哈希算法划分为16个物理分片，实现了千万级数据的高效存储与查询。