

# 潘江明

+(86) 15988183373 ◇ panjm2001@126.com  
浙江 · 杭州市留和路 318 号 ◇ 邮编: 310023

## 教育

浙江科技大学

2020.9 - 2024.6

本科生, 计算机科学与技术 (国家级一流专业)

- 核心课程: 计算机网络、操作系统、计算机组成、数据结构、数据库等
- 评奖评优: 浙江省政府奖学金、优秀学生二等奖学金、三好学生、创新创业三等奖学金等

## 竞赛奖项

ICPC - 浙江省大学生程序设计竞赛 - 银奖

2022.4

CCCC - 中国高校计算机大赛 GPLT 全国总决赛 - 金奖

2023.4

蓝桥杯软件和信息技术专业人才大赛 C/C++ 组 (国赛) - 银奖

2023.6

RoboCom 机器人开发者大赛 CAIP 编程设计 (国赛) - 铜奖

2022.8

## 工作经历

七牛云

2023.5 - 至今

研发实习生, 基础架构部

主要负责七牛云企业级 *DevOps* 平台的多个组件设计和开发, 编程语言主要是 *Golang*。

- 负责七牛云早期企业级 CMDB 的开发, 设计相关系统, 开发后端 API 接口、插件、中间件以及日常维护。
- 尝试通过使用协程和分页的策略对海量数据迁移工具进行优化, 在保证并发安全的情况下, 将性能提升了 100 多倍, 原本的工具在 200 秒内可迁移 3,200 条数据, 优化完之后在 300 秒的时间内, 可以迁移 450,000 条数据。

## 项目 / 研究经历

RLock | 基于 Golang 开发的 Redis 分布式锁

2023.7 - 2023.8

实现了分布式系统下, 通过使用锁来对资源进行保护, 并且由于 *Redis* 数据库的过期机制, 很大程度上的避免了发生死锁的可能性。同时该锁实现了可重入、阻塞申请、异步续租等功能。在多个 *Redis* 节点的情况下, 实现了 *RedLock* 算法, 在高可用性和数据一致性之间做了一个平衡。

- 在加锁、解锁和续租操作时, 通过执行 Lua 脚本, 来满足操作的原子性。
- 使用 Hash 数据结构来实现对锁的可重入性的支持, 获得锁的节点资源若再次申请该锁时, 不会被阻塞。
- 以阻塞的方式请求锁的时候, 若 tryLock 失败, 则以自旋的方式不断的申请锁, 直到成功申请到该锁或阻塞超时。
- 使用看门狗的方式实现对 Redis 锁的异步续租, 并且通过看门狗的运行标记保证重入锁时, 看门狗不会重复启动。通过上下文传递结束命令, 实现对看门狗的终止续租操作。
- 实现 RedLock 算法, 满足在多 Redis 节点的情况下, 缓和高可用与数据一致性之间的矛盾。
- 项目地址: <https://github.com/pjimming/rlock>

CMDB | 基于 Golang 开发的基础数据管理数据库

2023.5 - 至今

实现了七牛云新一代企业级 CMDB 的后端系统的设计、研发, 并通过接入 *CI/CD* 流程进行版本发布。

- 独立设计分布式机制, 通过重构之前旧 CMDB 系统的 service 层, 保证两个系统间数据同步。并且集成操作失败告警机制, 防止数据同步时发生异常而不能感知。
- 开发 ApiLog 中间件, 通过在 route 层对 http 请求进行采集, 记录请求接口、操作人、操作耗时、IP 地址、请求是否成功等相关数据。
- 通过并发请求实现全局搜索 API, 保证并发安全, 全库搜索耗时平均 2s 左右。尝试过前缀字典树和 AC 自动机算法, 效果不理想。
- 尝试使用记忆化搜索的算法对域名解析记录进行处理, 在毫秒级的时间里处理 100,000 条域名解析记录, 可以得到 9,800,000 条【初始域名 -> IP】的记录。
- 独立设计数据权限系统, 通过 RBAC 系统来实现控制不同用户的多种角色的数据访问权限。

## 技能

---

- 掌握 Golang 基本语法，熟悉 GMP 模型，了解 Golang 底层对于并发的实现。熟练使用 C/C++、Python、Shell。
- 熟悉计算机网络相关知识，了解 UDP、TCP、HTTP、HTTPS 等常用网络协议。
- 熟悉 Redis 数据类型使用场景和内部实现，了解其持久化和过期淘汰策略，了解缓存高并发场景，如缓存雪崩、缓存穿透等。
- 熟悉 MongoDB 的增删改查操作，了解其复制集、分片等原理。
- 熟练 Linux 环境下编程，了解部分 Linux 基础命令。熟悉 Docker 容器的使用，掌握 Docker 的基本命令。
- 熟练使用 Git 工具，了解 Makefile 的使用。
- 了解分布式微服务框架，有 gRPC 的使用经验，了解基础理论如 MapReduce、GFS、Raft 等。
- 多年 ACM 算法竞赛经历，有良好的编码风格和命名规范。
- 有良好的分析问题原因，迅速定位问题和快速解决 bug 的能力。

## 其他

---

- 数学水平：微积分、线性代数以及概率论与数理统计均 A 或 A+。
- 个人博客：<https://wiki.pjmcode.top>
- 个人简历：<https://cv.pjmcode.top>
- GitHub：<https://github.com/pjimmming>