

谭志勇

6年 | 男 | 28岁 | 湖南衡阳 13421843857 | newbiwtan@163.com 全日本科 | 湖南科技大学 (2013-2017) <http://github.com/tony-is-coding> | <https://tony-is-coding.github.io/>

专业技能

- **Java开发**: 深入掌握多线程、网络IO、集合框架,JVM,具有丰富生产问题分析实践经验
- **语言框架**: 熟练使用Spring、SpringBoot、MyBatis, 熟悉Netty网络编程
- **中间件**: 熟练掌握Mysql、Redis、RabbitMQ, 了解其原理及优化
- **微服务架构**: 熟悉 SpringCloudAlibaba, Dubbo使用, 熟悉微服务设计模式与实现
- **其他编程**: 熟练使用Python进行脚本编程, 熟悉Linux操作环境运维管理
- **行业知识**: 4年金融行业开发经验, 对金融业务设计有一定的认知了解

工作经历

深圳市小赢科技有限公司 **Java工程师 2022.07~2024.04**

内容: 1、优化升级公司资金放款调度平台调度架构模式, 提升系统稳定性与可检测性 2、通过策略算法实现资金匹配放款运营额度自动化生成, 减轻财务运营配置负担

深圳市点米二号科技有限公司 **Java工程师 2020.05~2022.07**

内容: 1、主导考勤核算系统设计与开发, 完成系统从Python到Java的重构升级, 实现性能提升目标 2、推进完成公司 86个Java项目的 CICD模式迁移 及 基于docker容器化部署升级

深圳市乐赢科技有限公司 **Java工程师 2018.03~2022.05**

内容: 1、独立负责公司 现金贷项目 登录注册、审核后台、逾期计算等模块功能开发 2、负责公司内部征信数据转接平台功能开发, 作为适配层转接外部征信系统到内部平台

项目经历

小赢资金放款调上度平台升级优化 功能模块负责人 **2022.10-2023.04**

原系统基于调用/回调方式实现订单在 内部额度过滤中、资金募集中、还款计划生成中、资方放款中、 订单放款成功/失败 多个状态间的扭转, 但是其中好几个阶段都依赖外部系统回调的稳定性, 导致系统整体的 可管理度以及容错都很低; 优化后采用基于 任务扫描DB驱动 + 回调 + 主动查询结果 的设计方案, 由放款调度系统自身来进行调度 与 结果把控, 实现绝大多数外部、内部异常场景可重试、无回调丢失;

个人职责:

1、负责设计系统新架构及编码工作, 保证新架构下系统运行稳定 2、引入任务重试设计、幂等性+ 数据库多版本CAS 设计方案以增强系统的容错性 3、推动外部新增处理状态查询接口, 完成系统改造

考勤核算三期 (2号人事部) 核心研发 / 项目主导人 2020.11-2021.06

考勤核算是 2号人事部 考勤业务线 的核心计算后端, 提供实时的员工打卡、企业考勤核算能力, 考勤 核算三期是在原有基础上进行从Python到Java的语言重构; 原核算二期采用分布式计算, 采用了Redis作为数据透传载体, 多任务协调以及大量数据透传导致性能 无意义损耗; 其次未规范化的业务设计致使数据IO很随意, 未合理合并IO查询与冗余数据查询导致网络浪费 此外Python语言本身天然具有GIL锁导致无法充分利用多核CPU特性; 核算三期使用Java语言对原系统进行重构, 引入多线程技术、IO计算分离设计思想, 规范数据库等耗时 IO的使用, 达成了较之前核算二期 核心应用场景下性能提升 10 倍, 并且机器资源节约75%

个人职责:

1、搭建基于SpringBoot、MySQL、Redis、RabbitMQ、Consul、Mybatis等技术的基础项目框架 2、负责主导项目重构的架构设计、功能模块划分; 主导各个阶段的方案编写与评审 3、引入多线程技术来提升 多数据库表IO、多员工数据计算等场景性能 4、设计基于「企业」维度的灰度方案来 实现 旧系统到新系统的逐步切换, 以及异常回滚能力 5、基于python开发 预发布数据验证系统, 基于两套系统数据结果分析来保证新系统的功能正确性

Neptune 开源作者 2021年 - 至今

Neptune 是一款基于Netty的RPC框架, 目标是提供10W级别的RPC访问并发, 致力于发展成一款 高性能、高可扩展的分布式RPC框架, 项目地址 <https://github.com/CloudNativeCompany/Neptune>

设计思想

- 应用层 服务注册与发现, 服务路由, 执行代理, 负载均衡, 集群策略, 序列化反序列化
- 连接层 连接分组管理, 服务隔离, 连接线程绑定, 重连管理
- 传输层 心跳机制, 断线重连, 空闲链路检测