

韦宇

性别：男

电话：18933549212

邮箱：3658043236@qq.com

CSDN: <https://blog.csdn.net/wy990880?type=blog>

年龄：21

学历：本科

求职意向：全栈开发实习生



教育经历

广东药科大学 计算机科学与技术 本科 2022-09 ~ 2026-06

专业技能

- **Java基础**：熟悉Java编程语言，有两年使用经验，掌握**集合框架**、异常、多线程、反射等核心机制
- **前端基础**：熟悉 Vue2、Vue3，掌握 Vue Router、Pinia/Vuex、Compostion API，了解 Vue 的响应式原理
- **框架**：熟悉Spring、SpringBoot、SpringCloud微服务组件，熟悉**Spring AOP**、**IOC**等原理
- **MySQL**：熟悉MySQL基础原理、**存储引擎**、**索引原理**、MVCC、事务等机制、具备一定的SQL性能调优能力
- **Redis**：熟悉Redis底层数据结构、分布式锁、线程模型、**内存淘汰策略**等机制，熟悉**缓存击穿**、**穿透**、**雪崩**概念
- **计算机网络**：熟悉TCP、UDP、HTTP、HTTPS等网络协议，掌握**TCP三次握手**、**四次挥手**、流量控制等机制
- **操作系统**：熟悉**进程**、线程、虚拟内存、**I/O多路复用**等，掌握进程间通信和多线程同步技术
- **AI**：了解**AI Agent**、**RAG**、**FunctionCall**、**LLM**（如阿里百炼、DeepSeek）**Promot**管理和编排的基本概念及原理
- **AI Condng工具**：熟练运用如**Cursor**、**通义灵码**、**ChatGPT**、**Claude**等AI开发大模型工具

实习经历

美的集团 Java开发实习生 2024-11 ~ 2025-04

实习描述：参与开发数据供应链部门**APS系统（高级计划与排程系统）**，与美的集团**iPass（集成化生产管理平台）**、以及**MES（制造执行系统）**深度集成，实现**供应链计划→生产排程→车间执行**全链路数据协同

工作职责：

- 主要参与项目核心链路开发、负责如工单下达、工艺路线管理、车间排产全链路模块开发、完成核心代码编写工作

工作成果：

- **性能优化与缓存策略**：针对数据接口高频查询的性能瓶颈，采用**Caffeine**实现本地缓存，避免重复查询数据库，显著降低接口响应时间，将平均响应时间**从800ms减少到180ms**
- **多线程异步任务开发**：基于 **CompletableFuture**结合分页机制（**PageSize=1000**）实现数据拉取的并发处理，实现了**单批次1200条数据平均处理时间从7s优化至1.8秒**
- **优化慢SQL**：通过**慢查询日志**和 **EXPLAIN** 分析，针对生产计划的执行依赖于任务的优先级排序，建立时间和任务优先级的**联合索引**，消除了file sort的影响，解决了需要频繁进行SQL查询的性能问题，**查询时间从秒级优化到毫秒级**

开源经历

AIResume-A 智能简历制作平台（GitHub 120+星星） **贡献者**
项目介绍：AIResume 是一个基于 Vue 3+Vite+TypeScript+ Ant Design Vue 的智能简历制作平台，支持用户进行简历编辑、模板切换、A润色与深度优化、AI 简历拷打等功能，并提供 PDF 导出与多模板共创。

个人贡献：

- 参与开发基础功能的实现、例如下拉菜单、日期选择等表单组件
- 引入AI润色简历功能，平台通过 **Cloudfiare Worker** 实现对 **OpenAI-API** 的代理，用户可自由切换不同大模型接口

在线体验地址:<https://resume.404.pub/>

项目开源地址:<https://github.com/weidong-repo/AIResume>

项目经验

AsyncScheduler（异步调度框架）

项目背景：学校实验室的医学AI训练场景需要对收集到的图像进行数据采集、清洗、特征提取、分布式存储等多个步骤。为了提高开发效率，我抽象为多个异步任务并开发了一个基于Java的多阶段异步任务框架。

个人职责：

- **负责架构设计**：采用**生产者-消费者模式**。整体框架分为**Flow Server（服务层）**和**Worker（执行层）**。Flow Server向客户端（执行主要业务）和Worker层提供HTTP服务。主要接口有创建任务、拉取任务、轮询任务状态等。Worker负责消费任务
- **数据库表设计**：设计主要的三张数据库表：任务信息表、配置表、位置表。方便任务快速注册和进行任务管理，**实现低耦合**
- **任务调度设计**：支持**按相对优先级来调度任务**。综合创建时间、更新时间、重试间隔（**采用渐进式间隔重试策略**）进行相对优先级排序。
- **性能调优设计**：对框架进行压测、**发现并解决性能瓶颈**，性能从一开始的**100QPS提升到500QPS**
- **架构优化设计**：多机竞争由**Mysql行级锁优化为Redis分布式锁**，下阶段考虑引入MQ，将任务拉取和执行解耦交给MQ

技术难点：

- **性能调优涉及面广**：对接口进行压测只有100QPS，通过**Mysql连接池**、**Redis缓存**减少竞争等手段优化
- **多机竞争方案及优化问题**：多个Worker去拉取任务容易拉到同一批任务。一开始这里在Worker侧引入Redis分布式锁来解决，任务冲突率解决90%，但Worker拉取和执行任务偶尔可能会CPU飙高至80%左右的问题。考虑**引入MQ**进行水平扩展效果