

徐俊飞

微服务架构师 / 技术负责人

masonsxu@foxmail.com

GitHub: masonsxu

厦门

期望: 面议

核心竞争力

4年后端研发经验，精通Go微服务架构设计与Python全栈开发。作为技术负责人，主导公司从Python单体架构向Go微服务架构转型，独立设计并实现完整的医疗放射治疗数据管理平台（10个微服务、441个Go文件、77,600行核心代码）。擅长DDD分层架构、数据湖架构、DICOM医疗影像、Casbin权限控制、容器化部署，带领8人团队完成多个生产级系统落地，具备52家医院的系统部署经验。

专业技能

Go 微服务技术栈 (核心优势)

- 编程语言：Go 1.24+，精通并发编程 (goroutine、channel)、性能优化、内存管理
- 微服务框架：CloudWeGo (Kitex RPC、Hertz HTTP)、gRPC、etcd服务注册发现
- 架构设计：DDD分层架构、IDL-First开发模式 (48个Thrift IDL)、Google Wire依赖注入
- 数据库：PostgreSQL、MySQL、MongoDB，事务处理、索引优化、连接池配置
- 数据湖：Trino查询引擎、多数据源管理、元数据管理
- 权限管理：Casbin (RBAC/ABAC)、菜单权限控制、角色定义与分配

基础设施与工程化

- 对象存储：RustFS (S3兼容)、AWS SDK Go v2、预签名URL
- 缓存/消息：Redis (缓存设计、分布式锁)、etcd配置中心、消息队列
- 容器化：Docker、Docker Compose、Portainer，熟悉K8s
- 可观测性：链路追踪 (request_id/trace_id)、日志聚合、统一错误码体系

Python 技术栈 (补充)

- 编程语言：Python，精通asyncio、Flask、 SQLAlchemy、Pandas
- 医疗领域：医疗影像处理 (SimpleITK、Pydicom、DICOM协议)

工作经历

福建自贸试验区厦门片区 Manteia 数据科技有限公司

技术负责人 / 后端架构师

2024.07 - 至今

- 技术转型领导：主导公司从Python单体架构向Go微服务架构转型，独立设计并开发Radius医疗放射治疗数据管理平台（10个微服务、441个Go文件、77,600行核心代码）
- 团队管理：带领8人团队（3后端+2前端+2测试+1产品），推动技术架构升级和团队能力建设，建立代码评审、技术分享等机制
- 技术成果：实现完整的DDD四层架构、数据湖集成、Casbin权限管理、DICOM医疗影像处理等微服务最佳实践，成为公司标准模板
- 岗位价值：在公司核心技术转型期承担架构设计、系统落地与技术决策职责，成为核心技术负责人

Python 开发工程师

2021.06 - 2024.06

- 系统架构与研发：主导Mria（信息管理系统）、Radius（数据管理系统）、WebDicomView（放疗医疗影像系统）的重构及技术攻关
- 项目部署：完成52家医院系统上线，打通第三方系统数据接口，数据传输准确率达99.9%
- 容器化部署：推动Shell脚本部署转型为Docker容器化架构，部署时间从4小时→30分钟，故障率降低45%

项目经历

Radius 医疗放射治疗数据管理平台 - 微服务架构

技术负责人 / 架构师 | 2024.07 - 至今

项目背景：为医疗放疗领域设计完整的数据管理解决方案，涵盖身份认证、患者管理、医疗影像、病历管理、数据湖等核心业务。独立设计并实现整个平台的微服务架构，从0到1构建生产级医疗数据管理系统。

技术架构：Go 1.24、CloudWeGo (Kitex RPC、Hertz HTTP)、PostgreSQL、etcd、Trino、Casbin、RustFS、Docker、Google Wire、Thrift | 代码规模：441个Go文件，77,600行核心代码，48个Thrift IDL | 微服务：10个服务 (identity、datalake、patient、cancer、dicom、medrec、ocr、athoslide、permission + API网关)

核心设计与实现：

- 微服务分层架构：设计DDD四层架构 (Handler→Logic→DAL→Converter)，实现10个独立微服务，采用仓储模式 (20+实体)
- 数据湖集成：设计datalake_srv服务，集成Trino查询引擎，支持跨数据源联合查询，实现多数据源管理和元数据管理
- 权限管理系统：基于Casbin实现RBAC权限模型，设计多级组织权限体系（组织→部门→用户），实现菜单权限动态解析
- 医疗影像存储：设计dicom_srv服务处理DICOM医疗影像数据，集成RustFS S3兼容存储，实现预签名URL机制
- 可观测性：设计RPC追踪中间件 (request_id/trace_id)，建立统一错误码体系（6位数字），实现分层错误转换
- 工程化：基于48个Thrift IDL定义接口，使用Google Wire依赖注入，完整Docker编排方案，环境驱动配置管理

项目成果: ✓ 构建完整的医疗放射治疗数据管理平台，覆盖9大业务领域 ✓ 代码规模: 441个Go文件、77,600行核心代码、48个IDL接口定义 ✓ 成为公司微服务架构标准模板 ✓ 文档完善（超过2000行技术文档）

Mria 信息管理系统 - Python 单体应用

核心开发 | 2021.06 - 至今

技术栈: Python、Flask、Pandas、MySQL、MongoDB、WebSocket、Docker、Portainer

核心贡献:

- 核心模块开发:** 独立完成权限控制、表单设计、流程设计等核心模块，表单查询效率提升50%，叫号系统用户满意度提高40%
- 容器化部署:** 主导Shell脚本→Docker容器化转型，设计多环境Docker镜像构建策略，部署时间从4小时→30分钟，故障率降低45%
- 业务规模:** 完成52家医院的系统部署和技术对接，数据传输准确率达99.9%

Radius 数据管理系统 - Python 数据处理

后端开发 | 2021.06 - 2022.05

技术栈: Python、SimpleITK、Pydicom、SQLAlchemy、APScheduler、Docker

核心贡献:

- 数据迁移优化:** 基于APScheduler实现灵活数据迁移配置，自动化减少人工成本60%
- 医疗影像处理:** 运用SimpleITK和Pydicom设计Dicom Server节点服务，完成医疗影像数据迁移、解析与存储
- 容器化部署:** 设计基于Docker的微服务架构，部署效率提升80%，资源利用率提高35%

教育背景

河南城建学院

信息管理与信息系统（大数据方向） | 2017.07 - 2021.06

荣誉: 2021年河南省优秀学位论文、国家级单项奖学金、省级单项奖学金

自我评价

4年后端研发经验，精通Go微服务架构设计与Python全栈开发。作为**技术负责人**，主导公司技术转型，独立设计并实现完整的医疗数据管理平台（**10个微服务、441个Go文件、77,600行核心代码**）。

技术能力: 深度掌握CloudWeGo技术栈，擅长DDD分层架构、IDL-First开发模式、数据湖架构（Trino）、权限管理（Casbin）、医疗影像处理（DICOM）、容器化部署。具备扎实的架构设计能力，能够从0到1设计和实现完整的微服务平台。

管理经验: 带领8人团队推动技术架构升级，成功完成52家医院的系统部署和技术对接。4年医疗信息化领域经验，深刻理解医疗业务流程和数据安全合规要求。

最后更新: 2025年12月 | GitHub: [github.com/masons xu](#) | 核心项目: Radius 医疗数据管理平台