

一种基于微服务的飞机加油调度系统的架构研究

闵晓霜,董岩,刘红,董敏

(中国电子信息产业集团有限公司第六研究所,北京 100083)

摘要:随着民航业的高速发展,为其提供保障服务的飞机加油业务对电子化、信息化、智能化发展有着迫切的需求,目前需在各干线机场建立飞机加油调度系统,实现加油任务调度、加油作业实施等业务的电子化、自动化。针对系统业务以及对高可用、可扩展、快速部署等方面的需求,引入微服务架构,对微服务及相关技术进行研究,设计基于微服务的飞机加油调度系统架构,选用 Spring Cloud 微服务技术框架进行实现,对应用进行容器化部署。该架构在中航油智慧加油系统中应用,满足了系统对软件高速迭代、机场系统快速接入与定制化的需求。

关键词:微服务;飞机加油调度;Spring Cloud;容器

中图分类号:TP393

文献标识码:A

DOI:10.16157/j.issn.0258-7998.212166

中文引用格式:闵晓霜,董岩,刘红,等.一种基于微服务的飞机加油调度系统的架构研究[J].电子技术应用,2022,48(10):87-91,96.

英文引用格式:Min Xiaoshuang,Dong Yan,Liu Hong,et al. Research on the architecture of aircraft refueling scheduling system based on micro service[J]. Application of Electronic Technique,2022,48(10):87-91,96.

Research on the architecture of aircraft refueling scheduling system based on micro service

Min Xiaoshuang,Dong Yan,Liu Hong,Dong Min

(The Sixth Research Institute of China Electronics Corporation,Beijing 100083,China)

Abstract: With the rapid development of the civil aviation industry, the aircraft refueling business that provides support services has an urgent demand for the development of electronization, informatization and intelligence. At present, it is necessary to establish an aircraft refueling scheduling system at each trunk airport to realize the electronization and automation of refueling task scheduling, refueling operation implementation and other business. According to the system business and the requirements for high availability, scalability and rapid deployment, the micro service architecture is introduced, the micro service and related technologies are studied, the aircraft refueling scheduling system architecture based on micro service is designed, the Spring Cloud micro service technology framework is selected for implementation, and the application is container deployed. The architecture is applied in the intelligent refueling system of CNAF. It meets the requirements of the system for high-speed software iteration, rapid access and customization of airport system.

Key words: micro service; aircraft refueling dispatch; Spring Cloud; container

0 引言

民航业的飞速发展对为飞机提供油料保障的飞机加油业务也提出更高的要求。各地区机场都建设有航空加油站,提供飞机加油保障服务,但目前该业务整体信息化水平较低,大多数机场主要依靠对讲和手写油单的方式进行传统加油作业,除手工录入油单数据到航油ERP系统之外,与其他系统间缺少数据共享与协作,航油总部、所属机场等相关方面都无法实时掌握加油保障的情况。为此,需建立一套飞机加油调度系统,依靠先进的信息技术手段,不仅要优化飞机加油的作业模式,而且要打通机场与航油总部的信息共享通道,实现对加油保障运行的全局监督监控以及销售、统计等业务,同

时与航空公司进行数据、业务的对接。将在各机场建立分布式业务集群,将系统部署在云平台之上。

加油调度业务存在以下特点:(1)涉及航显信息显示、基础信息维护、加油任务调度、加油任务执行、油单上传结算等多个环节;(2)对接航油总部、航班信息提供商、机场系统、航空公司等多个系统;(3)存在调度人员Web端、加油员移动端等多端用户访问需求;(4)需要部署到多个机场。系统需要能够适应软件的高速迭代以及各机场的快速部署,并能够稳定运行为机场提供保障服务。目前流行的微服务架构能够将系统功能分散到独立的微服务中,每个微服务进行独立部署^[1],彼此之间采用轻量级通信,能够灵活地开发、部署以及扩展升级,适

应系统需求,是非常理想的解决方案。

微服务是随着信息技术迅速发展不断演进而产生的一种设计思想^[2],本文详细介绍了从单体架构、SOA到微服务架构的演进,对技术框架、容器、服务间通信等相关技术进行分析,设计了一种基于微服务的飞机加油调度系统架构,并在中航油智慧加油系统中得到应用。

1 微服务及相关技术

1.1 微服务背景

云计算、物联网等技术的发展,以及用户对软件系统的多样化需求,促进了设计架构的一次次改变,经历了从单体架构到面向服务的 SOA 架构,再到如今灵活的微服务架构成为主流,软件系统架构的演进如图 1 所示。

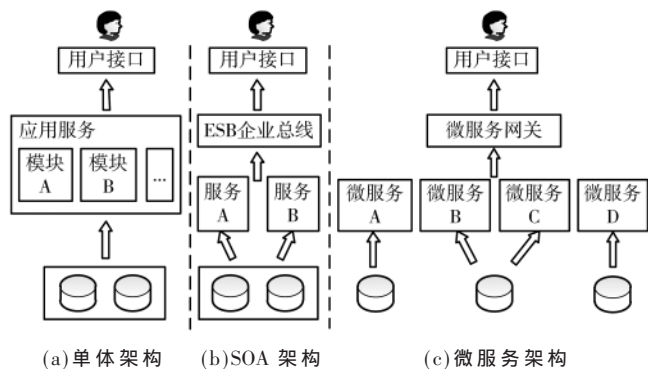


图 1 软件系统架构演进

单体架构是比较传统的架构风格,所有功能模块都被打包成一个程序包,连同数据资源部署在一台服务器上。在系统规模小、业务简单时,采用这种方式简单实用,便于开发、测试、部署以及维护;但是在用户量大、功能复杂且软件需求不断迭加时,系统开发升级会非常困难,且一个故障可能导致整个系统崩溃。

为解决上述问题,出现了面向服务的架构(Service-Oriented Architecture, SOA)。SOA 根据业务将系统进行粗粒度的拆分,形成可独立部署的服务,服务间通过 Web Service 等进行网络调用,营造了一种松耦合的环境。企业服务总线(Enterprise Service Bus, ESB)技术能够有效实现不同服务间的通信与整合,它的引入进一步促进了 SOA 的发展^[3]。虽能够解决大部分问题,但 SOA 强调自上而下的整体架构,服务拆分粒度大。

微服务架构在 SOA 之上进一步发展,融合了 SOA 服务架构、组件化以及领域建模的思想,提倡“业务彻底组件化和服务化”^[4],对服务进行更细粒度的划分,每个服务不仅能够独立部署运行,还可以采用异构的开发技术、存储技术等,服务间使用轻量的通信机制进行数据交互,如 Restful API,引入微服务网关对外部请求进行路由转发和过滤,使得系统在开发、部署以及迭代方面都更为灵活高效,系统整体响应速度也更快。

其中,微服务也被称作是细粒度的 SOA,但它们之

间的差异不只在服务粒度,还体现在以下几个方面:

(1)服务间通信机制不同,这也是二者最大的区别,微服务通信采用了轻量级设计,一般基于 REST 协议实现,而 SOA 采用重量级的网络通信协议,如 Web Service;(2)服务管理方式不同,微服务强调服务的自治性,采用去中心化管理,每个服务是独立的应用,独立管理自己的数据库,独立部署和运行,SOA 则采用集中式的管理,使用统一的数据中心;(3)适用系统规模不同,微服务更适合互联网环境,与高流量、高并发、高可用等需求密不可分,适用于软件高速迭代的系统,SOA 则主要面向企业计算,系统灵活性相对较低,服务更像是子系统,适合对大型企业应用中异构子系统的集成。

通过上述分析,采用微服务架构设计,针对飞机加油业务进行微服务的划分,能够提高系统的开发效率、方便系统扩展,可以很好地满足软件功能多样、高速迭代以及灵活部署等需求。

1.2 相关技术

1.2.1 Spring Cloud 微服务框架

目前,在微服务开发框架中 Spring Cloud 最为流行^[5]。Spring Cloud 是一套基于 Spring Boot 的微服务工具包,为用户提供了一系列的核心组件,包括:Eureka 服务发现组件、Register 服务注册组件、Zuul 服务网关组件以及 Config 分布式配置组件等,这些组件开箱即用,能够实现以 Java 作为底层编程语言的微服务架构的核心功能^[4]。

其中,微服务网关的 Zuul 组件承担着身份认证、动态路由等重要功能,客户端请求服务调用会通过 Zuul 转发到后端微服务的 Restful API,从而可以调用不同的微服务。每个微服务都有自己专属的数据库供其存储查询,以此来满足用户的各种功能需求。

1.2.2 虚拟化与容器技术

虚拟机是传统的虚拟技术,它模拟了完整的系统软硬件,相当于一个完整的计算机系统。可以将应用部署到虚拟机上,但是由于虚拟机携带了操作系统,每次启动、运行的负担较重。

Docker 是一种轻量化的虚拟技术,不同于虚拟机,它可以将应用及其依赖包打包到一个容器中^[6],然后发布到 Linux 主机上,容器是运行在操作系统之上的一个进程,采用沙箱机制,相互之间隔离。Docker 与虚拟机的对比如表 1 所示。

Docker 的轻量级特性使其非常适合微服务的部署

表 1 Docker 与虚拟机的对比

项目	Docker	虚拟机
启动时间	秒级	分钟级
内存访问效率	1 次映射(虚拟内存-物理内存)	2 次映射(虚拟内存-虚拟物理内存-物理内存)
镜像大小	MB 级	GB 级
可启动数量	上万个	最多几十个

与维护。

Docker 有三大核心概念——仓库、镜像和容器,几乎所有操作都是围绕它们进行的。Docker 镜像相当于一个只读模板,将应用程序及依赖包打包到镜像文件,文件中还包含着创建 Docker 容器的说明;容器是通过镜像创建的可运行实例,可通过 Docker API 来启动、停止、移动以及删除容器^[7];仓库是用来保存镜像的地方,镜像通过 push 命令上传到仓库,使用时用 pull 命令下载即可。其中,镜像是容器运行的前提,镜像之于容器,就像面向对象思想中的类和对象。

容器运行所需资源少,启动快速、运行轻量、方便移植、管理简洁,便于定位与修复故障^[8]。因此,采用 Docker 容器技术来实现微服务部署优势显著。

1.2.3 服务间通信

微服务都各自运行于独立的进程中,服务间采用轻量级通信机制进行信息交互与消息传递。在微服务框架下,服务间的通信模式可分为两种:同步和异步。

(1)同步模式。即请求/响应模式,当客户端发出请求时,服务器能够即时进行响应。该模式要求响应是快速即时的,没有代理的中间件,实现简单,但通信过程中可能会引起进程的阻塞。可以采用 REST 协议和 Thrift 协议,对于 Spring Boot 集成的微服务架构,也可以选用 RPC,其跨语言性、社区支持性较好。

(2)异步模式。异步通信时,对于客户端的请求,服务器不必即时响应,也不会阻塞进程。这种模式通过中间件代理^[9],在客户端与服务器之间进行消息缓冲,实现了微服务间的解耦,但是异步通信的复杂性大大增强,不适合处理太丰富的数据主题。消息队列是一种主要的

异步通信机制,常见的有 Kafka、RabbitMQ 等,消息生产者将消息发送到消息队列,然后由消息的消费者从消息队列中获取,而后进行业务处理^[10]。对于不频繁的异步调用,也可以采用 Restful 风格的 HTTP 协议。

两种模式各有优缺点与适用的情况,在微服务框架中,可以采用两种模式混合的通信方式,根据实际情况选用不同的通信机制,提高系统的整体性能。

2 基于微服务的飞机加油调度系统架构

2.1 系统总体架构

飞机加油调度系统基于微服务架构构建,整体采用前后端分离的设计,从架构设计层面分为前端应用层、BFF 机场渠道服务层和后台无状态微服务层三部分,前端应用层包含浏览器端 Web 应用程序与移动端 App。通过 BFF 机场渠道服务将业务请求转化为无状态请求,通过内部接口(RPC、HTTP、Kafka)访问后台无状态服务,对业务进行解耦,使应用程序插件化、组件化。系统的微服务架构如图 2 所示。

为了应对软件的高速迭代与机场系统的快速接入与定制化的需求,系统采用容器技术结合 DevOps 容器云平台构建,每个微服务均独立部署在 Docker 容器中。通过创建面向机场的系统镜像,当需要接入新机场的时候通过配置文件与机场镜像快速部署新的机场实例。

(1)前端应用层:前端分为航班调度、加油员手持调度和基础信息管理 3 个系统,其中航班调度和基础信息的管理在 Web 端,加油员手持调度运行在移动端。

(2)BFF 机场渠道服务层:用于前端的后端(Backend For Frontend, BFF)是一种为解决微服务模式前后端交互问题而产生的模式,通过增加中间层隔离前端对后端

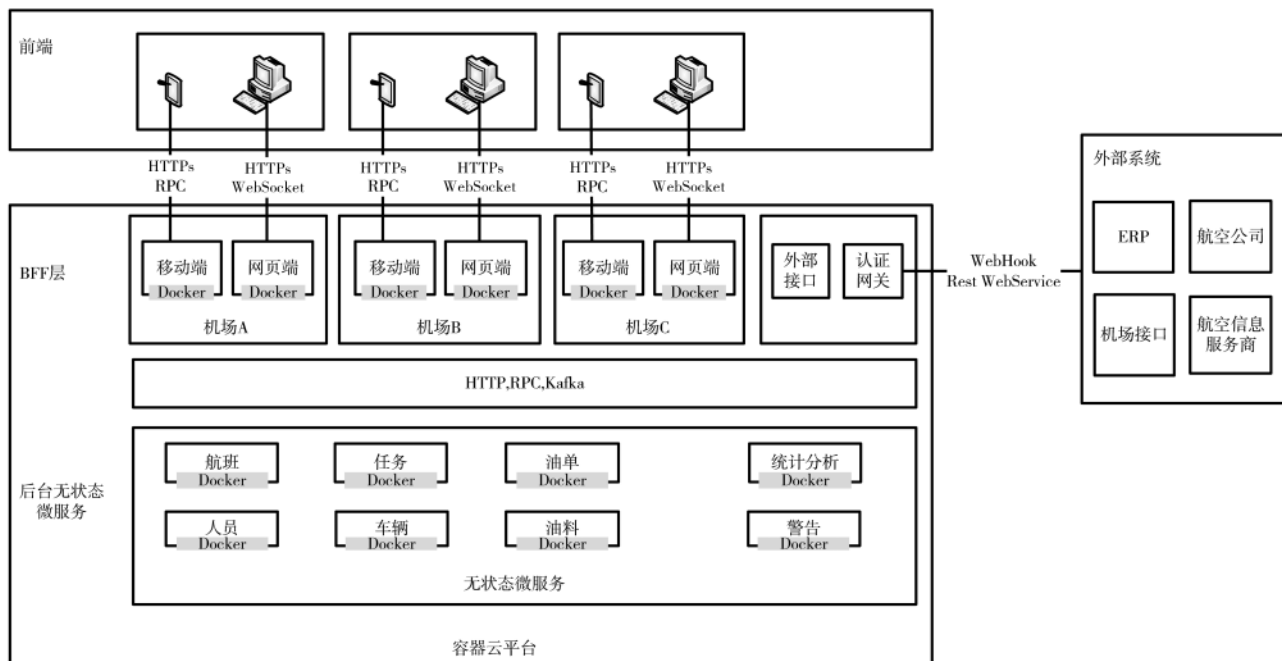


图 2 系统微服务架构

服务的直接访问,可以过滤、转发、组合前端请求,是服务于前端的后端服务,相当于 API 网关。BFF 机场渠道服务层采用 BFF 模式并结合微服务中服务网关的思想,构建对应于特定机场特定前端的机场渠道服务(在干线机场要部署本机场系统及其所管理的所有支线机场系统),以及过滤各种外部系统访问的认证网关服务,建立了系统内外访问接口与后端服务间的屏障。

(3)后台无状态微服务层:后台微服务不对外暴露,是无状态的,访问时要进行 token 认证。根据系统业务对微服务进行拆分,主要分为业务服务、基础服务和其他服务。业务服务包括航班服务、任务服务、油单服务等;基础服务包括人员服务、车辆服务、油料服务等;其他服务包括统计分析服务、警告服务等。

2.1.1 数据库分库

微服务与数据库通过回调 MyBatis 数据库中间件进行数据交互。数据库以业务作为分库依据,为与之相对的微服务提供支撑,主要采用 MySQL 做持久化存储,采用 Redis 进行数据缓存。系统的数据库架构如图 3 所示,整体划分为基础数据、业务数据以及共享数据 3 个数据区,与微服务相对应,基础数据区包含人员、车辆、机场等基础信息,这部分信息是进行业务工作的基础支撑数据;业务数据区包含航班、任务、油单等业务数据,根据业务产生;Redis 对人员登录 token、权限等高实时性的信息进行缓存。

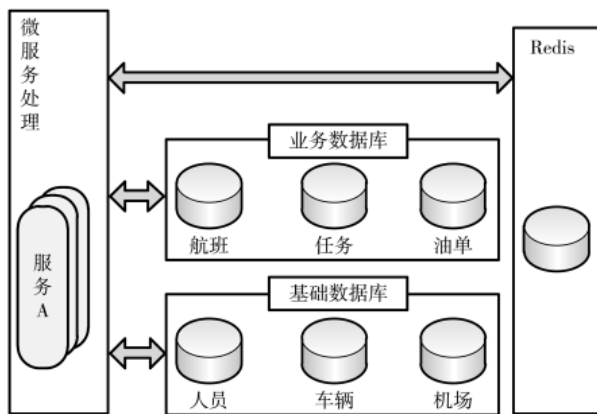


图 3 系统数据库架构

2.1.2 系统内部接口

为了保证数据在公网中的数据安全,从外网接入系统的包括网页、移动端的数据都需要进行 SSL 数据加密。

(1)用户终端到 BFF 层

移动端 App 和浏览器向 BFF 机场渠道服务发起异步请求,如登录、业务初始化等时,使用 HTTPs;要建立长连接时,移动端采用 RPC、浏览器采用 WebSocket,创建通信隧道进行数据传输。

(2)BFF 层到后台无状态服务

BFF 机场渠道服务后台无状态服务,异步请求如微

服务调用,使用 Restful 的 HTTP 协议;要建立长连接时,采用 RPC 协议创建通信隧道。

(3)Kafka 消息通信

当前后端需要不间断的发送数据并同时进行处理,如实时数据采集、判断、告警等;当消息和消费者成 1 对多关系的时候使用消息队列来构建内部接口,如数据超过规定值时告警等,需要使用 Kafka 来建立消息队列,对数据进行处理。

Kafka 消费接口:接收外部系统的航班信息。通过实时接收 Kafka 消息队列里的航班信息,解析该航班信息并进行一定的数据整理,最后将其写入到数据库中。

Kafka 生产接口:发送任务相关的一些保障数据和告警信息。在任务保障的主要时间节点,如任务下发、接收、执行以及油单生成、上传等时刻,将保障数据发送到 Kafka 的消息队列中由第三方消费该数据。对于一些告警信息(如油料参数报警、任务延迟报警、油料不足报警、航班延误等),通过消息对列将实时数据推送给无状态的告警服务,告警服务判断是否报警并将数据存入 Redis,然后通过 Redis 的发布订阅功能推送给相关工作人员。

2.1.3 系统外部接口

(1)接入信息

当需要接入外部系统的实时信息时,提供 REST 接口供其调用。例如从服务商接入的航班信息;从总部 ERP 接入人员、车辆、航空公司等基本信息。

考虑到接入的安全性,采用 JWT 认证网关进行接入验证,通过 token+IP 信息进行接口验证。

(2)推送信息

向外部系统实时推送信息的场景,采用 WebHook 接口。例如向航空公司推送实时的任务进度以及油单数据等;向总部 ERP 发送油单数据等;向机场发送加油保障进度等。

2.2 系统功能架构

飞机加油调度系统通过收集航班信息的起、落地时间,结合加油员、加油车以及停机坪位置等信息,对机场内所有航班的加油保障任务进行指挥调度,为加油员派发调度任务,加油员接受任务并进行加油作业,加油成功后生成电子油单,经签字确认后上传至云平台作为结算的依据,确保加油作业安全、高效、及时。

2.2.1 系统功能

系统主要分为航班任务调度、加油员加油调度、基础信息管理以及统计信息显示 4 个功能,系统功能架构如图 4 所示。

(1)航班任务调度

调度员登入航班任务调度系统,根据航班信息对机场内的加油任务进行手动或智能化的调度。功能包括:

- ①列表显示:对航班、加油员、所属机场航班、加油任务、已下发任务等信息分别进行列表显示;
- ②调度功能:

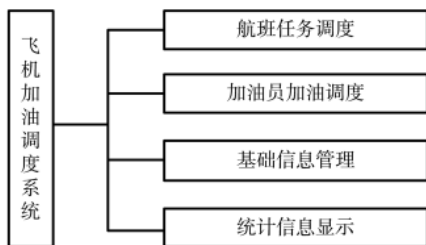


图4 系统功能架构

包括调度员手动调度和智能调度,手动调度由调度员根据各种信息进行人工判断,将任务派给加油员,智能调度提供智能任务计划并展示给调度员,经调度员确认后自动下发任务给加油员;③油单管理:对已完成任务生成的油单进行管理,可进行油单下载、更新等;④管理配置:包括人员登录登出、航显配置管理、加油员分组管理等。

(2)加油员加油调度

加油员采用手持PAD登入加油员加油调度系统,驾驶加油车进行飞机加油保障。功能包括:①加油员登录登出,人车绑定;②任务列表、任务详情、航班列表;③任务申请,加油员可主动申请未下发的航班加油任务,经调度员同意后申请成功;④电子签名、油单管理等。

(3)基础信息管理

系统管理员对人员、车辆、油料、航班、航空公司、机场、机位油井以及常用语等数据的设定管理及油单审核等业务,实现统一的基础信息管理管控。

(4)统计信息显示

对加油业务相关的一些信息进行统计,体现历史加油情况,为管理提供依据。功能包括:①航空公司统计:对航空公司的销售量、油单数量等进行统计;②加油员统计:对加油员的工作时间、加油量、加油飞机架次等进行统计;③车辆统计:对加油车的作业时间、加油量、加油飞机架次等进行统计;④对各类统计信息的搜索、导出。

2.2.2 系统业务流程

飞机加油调度业务的主要流程分为调度任务生成、调度任务执行、调度任务完成3个阶段,业务流程如图5所示。

(1)调度任务生成阶段

系统从外部系统获取航班信息,生成航班列表,再根据航班列表生成航班加油任务并呈现给调度员,调度员也可手动添加临时航班。调度员可对辖区内加油员进行任务派发,同时,加油员也可根据情况主动申请未派发的任务,最后由调度员确认是否予以通过。

(2)调度任务执行阶段

接受任务后,加油员到达指定机位并确认到位。任务所属飞机停至机位后,加油员对任务信息进行确认,然后开始实施加油作业。加油过程中,加油员需按照燃油加注操作流程规范上报作业进度,完成加注并确认数

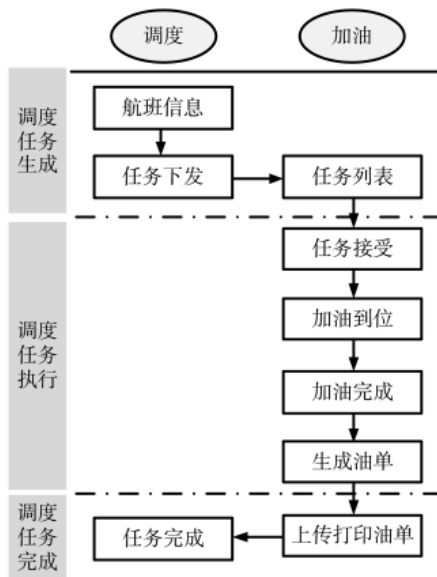


图5 业务流程

据无误后,系统将生成电子油单,由飞机机长在PAD上签字确认。

(3)调度任务完成阶段

通过车载打印机打印油单,并将油单及任务数据上传到机场云服务器,至此加油任务完成。机场云服务器将接收到的油单数据实时发送到总部进行结算。

3 应用

本系统在中航油智慧加油系统中应用,采用Spring Cloud微服务框架实现。系统前端分为航班调度(Web端)、加油员手持调度(PAD端)和基础信息管理3个系统,以react为基础,同时支持PC和PAD两个操作环境;后端采用Spring Cloud为微服务提供支持,主要分为业务操作、基础信息管理以及其他功能的微服务,微服务均独立部署在Docker容器中。利用Spring Cloud的Zuul工程作为网关来访问服务器的服务集群,其中包含了对权限的相关处理,通过过滤器filter实现权限的配置与检查,权限相关数据在Redis中缓存;同时,利用token进行分布式的人员状态管理,利用WebSocket进行前后端通信,对实时性高的数据进行交换。通过Kafka和外部接口服务来与其他系统进行数据交互。

飞机加油调度系统实现了航班加油任务的手动调度与智能调度功能,使得加油员能够通过手持PAD接收并执行任务,支持油单的管理与上传,也实现了对基础信息的集中管控以及多项信息的综合统计功能,系统部署在各地干线机场,负责本机场与所管辖支线机场的加油调度。

4 结论

在飞机加油保障信息化系统建设的背景下,本文针对飞机加油调度业务的特点和实际需求,采用微服务作

(下转第96页)

参考文献

- [1] 于亚萍, 孙立宁, 张峰峰, 等. 基于小波变换的多特征融合 sEMG 模式识别[J]. 传感技术学报, 2016, 29(4): 512-518.
- [2] BABITA, KUMARI P, NARAYAN Y, et al. Binary movement classification of sEMG signal using linear SVM and Wavelet Packet Transform[C]//IEEE International Conference on Power Electronics. IEEE, 2017.
- [3] 胡晓, 李莉, 任小梅, 等. 基于小波系数熵的表面肌电信号识别[J]. 中国医学物理学杂志, 2007, 24(3): 212-214.
- [4] 陈保家, 陈学力, 沈保明, 等. CNN-LSTM 深度神经网络在滚动轴承故障诊断中的应用[J]. 西安交通大学学报, 2021, 55(6): 28-36.
- [5] 陈红松, 陈京九. 基于 ResNet 和双向 LSTM 融合的物联网入侵检测分类模型构建与优化研究[J]. 湖南大学学报(自然科学版), 2020, 47(8): 1-8.
- [6] 郑直, 张华钦, 潘月. 基于改进鲸鱼算法优化 LSTM 的滚动轴承故障诊断[J]. 振动与冲击, 2021, 40(7): 274-280.
- [7] 李兴秀, 唐建军, 华晶. 结合 CNN 与双向 LSTM 的心律失常分类[J]. 计算机科学与探索, 2021, 15(12): 2353-2361.
- [8] 李达, 张照生, 刘鹏, 等. 基于改进长短时记忆神经网络-自适应增强算法的多天气车辆分类方法[J]. 汽车工程, 2020, 42(9): 1248-1255.
- [9] VERMA H, KUMAR S. An accurate missing data prediction method using LSTM based deep learning for health care[C]//Proceedings of the 20th International Conference on Distributed Computing and Networking, 2019: 372-376.
- [10] 陈景良, 李东新. 基于 LMS 的语音信号去噪算法[J]. 国

外电子测量技术, 2017, 36(6): 22-25, 30.

- [11] 石欣, 范智瑞, 张杰毅, 等. 基于 LMS-随机森林的肌电信号下肢动作快速分类[J]. 仪器仪表学报, 2020, 41(6): 218-224.
- [12] KAMAVUAKO E, FARINA D, JENSEN W. Use of sample entropy extracted from intramuscular EMG signals for the estimation of force[C]//Proc. of the 15th Nordic-Baltic Conference on Biomedical Engineering and Medical Physics. Berlin: Springer, 2011: 125-128.
- [13] 陈思佳, 罗志增. 基于长短时记忆和卷积神经网络的手势肌电识别研究[J]. 仪器仪表学报, 2021, 42(2): 162-170.
- [14] 王红旗, 李林伟, 毛啊敏. 基于小波包的表面肌电信号特征表示与识别[J]. 计算机工程与应用, 2015, 51(9): 217-220, 247.
- [15] 张毅, 冯晓林, 罗元. 基于改进小波包与样本熵的表面肌电信号特征提取[J]. 计算机应用研究, 2015, 32(3): 701-704.

(收稿日期: 2021-08-17)

作者简介:

孔康(1999-), 男, 本科, 主要研究方向: 机械工程。

李德盈(2000-), 男, 本科, 主要研究方向: 机械设计。

孙中圣(1978-), 通信作者, 男, 博士, 副教授, 主要研究方向: 气动控制、肌电控制, E-mail: billsun1978@163.com。



扫码下载电子文档

(上接第 91 页)

为系统的基础架构, 详细阐述了微服务架构的历史演进过程及各种架构的适用场景, 对实现微服务架构的相关技术进行分析, 提出了基于微服务的飞机加油调度系统架构, 并在中航油智慧加油系统改造项目成功应用。系统可用性、可扩展性高, 部署灵活高效。

目前, 微服务架构的应用还处于起步阶段, 随着系统后期建设与扩展升级, 对微服务的研究与应用将更加深入, 以期系统稳定运行与性能优化提供支撑。

参考文献

- [1] 付琳琳, 邹素雯. 微服务容器化部署的研究[J]. 计算技术与自动化, 2019, 38(4): 151-155.
- [2] 郑明钊, 张建强. 基于微服务的大平台系统架构演进探讨[J]. 软件, 2017, 38(12): 165-169.
- [3] 冯培培, 王辉. 基于 ESB 技术的系统集成框架的研究[J]. 通信技术, 2010, 43(1): 195-197.
- [4] 冯志勇, 徐砚伟, 薛霄, 等. 微服务技术发展的现状与展望[J]. 计算机研究与发展, 2020, 57(5): 1103-1122.
- [5] 吴化尧, 邓文俊. 面向微服务软件开发方法研究进展[J]. 计算机研究与发展, 2020, 57(3): 525-541.

- [6] 王亚玲, 李春阳, 崔蔚, 等. 基于 Docker 的 PaaS 平台建设[J]. 计算机系统应用, 2016, 25(3): 72-77.
- [7] 刘国庆. 基于 Docker 的云集成开发环境的研究与实现[D]. 成都: 电子科技大学, 2017.
- [8] 吴坤安, 黄文思, 韩泽华, 等. 基于 Docker 的数据库微服务系统设计与实现[J]. 国外电子测量技术, 2017, 36(12): 57-62.
- [9] 陆庆, 周世杰, 秦志光, 等. 消息队列中间件系统中消息队列与消息分发技术研究[J]. 计算机应用研究, 2003(8): 51-53.
- [10] 何双元. 高并发下消息队列模型的研究与应用[D]. 武汉: 武汉理工大学, 2015.

(收稿日期: 2021-09-17)

作者简介:

闵晓霜(1987-), 女, 硕士, 高级工程师, 主要研究方向: 信息系统软件、工业控制与信息安全。

董岩(1976-), 男, 本科, 工程师, 主要研究方向: 自动控制、智能信息处理及智能信息控制。

刘红(1986-), 女, 硕士, 工程师, 主要研究方向: 工控系统及安全。



扫码下载电子文档

版权声明

经作者授权，本论文版权和信息网络传播权归属于《电子技术应用》杂志，凡未经本刊书面同意任何机构、组织和个人不得擅自复印、汇编、翻译和进行信息网络传播。未经本刊书面同意，禁止一切互联网论文资源平台非法上传、收录本论文。

截至目前，本论文已经授权被中国期刊全文数据库（CNKI）、万方数据知识服务平台、中文科技期刊数据库（维普网）、DOAJ、美国《乌利希期刊指南》、JST 日本科技技术振兴机构数据库等数据库全文收录。

对于违反上述禁止行为并违法使用本论文的机构、组织和个人，本刊将采取一切必要法律行动来维护正当权益。

特此声明！

《电子技术应用》编辑部

中国电子信息产业集团有限公司第六研究所