

基于SpringCloud的航运EDI平台 IT生态环境设计

文/张树新 吴海斌 蒙辉 汤麦伦

摘要：云是当前IT行业最活跃的生态圈，当IT人员从IaaS、Docker以及微服务架构获得便利、敏捷的同时，云技术生态链的连贯性、可靠性成为平台最大的痛点。本文以航运EDI平台为背景，通过分析航运EDI平台单节点系统存在的痛点，并对采用新技术后开发测试、实施运维、集成扩展过程存在的问题进行分析，提出云计算时代IT生态环境的规划愿景，然后基于SpringCloud分布式架构构建完整的云生态环境，最终实现降低专业人员技术门槛、提供灵活高效的运维环境，达成平台具备持续集成和交付能力的目标。

关键词：SpringCloud^[2]；微服务架构；分布式；航运；EDI；IT生态环境



DOI:10.16301/j.cnki.cn12-1204/f.2018.02.040

引言

EDI电子数据交换技术在航运中拥有无可取代的地位。航运EDI主要服务于贸易商、海关、商检、港口、航运企业、货代、船代等企业部门，然而，由于企业部门使用的EDI标准不一致，航运EDI交换自动化程度较低，国内目前仅上海亿通国际建有较为成熟的航运EDI交换中心，主要通过协议适配、管道、字典等技术手段实现报文语义识别及格式转换^[1]。

作为最新一代云计算技术框架，SpringCloud 为开发者提供了在分布式系统（配置管理、服务发现、熔断、路由、微代理、控制总线、一次性token、全居琐、leader选举、分布式session、集群状态）中快速构建的工具，使用Spring Cloud的开

发者可以快速地启动服务或构建应用，同时能够快速和云平台资源进行对接^[2]。SpringCloud作为新生的微服务框架，相比阿里巴巴Dubbo等RPC框架，Spring Cloud提供的全套分布式系统解决方案更利于构建完整的IT生态环境。

本文通过分析航运EDI平台单节点系统存在的痛点，并对采用新技术后开发测试、实施运维、集成扩展过程存在的问题进行分析，提出云计算时代IT生态环境的规划愿景，然后基于SpringCloud分布式架构构建完整的云生态环境，最终实现降低专业人员技术门槛、提供灵活高效的运维环境，达成平台具备持续集成和交付能力的目标。

一、EDI平台现状及问题

目前国内航运EDI平台大多仍处于单节点系统或小工具层面，制约其发展的原因不外乎EDI标准不统一及现有平台技术发展缓慢。

技术发展缓慢并不是因为EDI核心技术复杂或是共性技术无法应用，相反EDI核心技术非常简单，也正是因此，EDI系统、小工具在小范围内应用良好。

小系统、小工具适应面有限，航运对通用性强的平台级EDI有着迫切的需求，如何适应繁杂不一的标准，不断接纳新的标准，打造开放统一的一站式交换环境，一直是平台级EDI努力的方向。

EDI平台的开放性是个复杂、综合、长期的命题，也为了有效降低成本，在新技术应用上，EDI平台从未放慢追随的脚步，而新技术的应用带来了学习成本增加、专业技术人员难觅、运维复杂度增加、集成扩展不易等问题，应用新技术的副作用将开发运维成本逐渐推高，甚至超过全新开发的成本，而违背了使用新技术的初衷。正是这样的客观情况，航运EDI平台技术的发展相对缓慢。

当然，部分平台也顺应发展，逐步应用SOA架构、阿里巴巴的Duboo微服务架构、IaaS基础云、Docker容器等新技术，很大程度上，新技术地使用降低了平台开发技术门槛，提升了平台的扩展性及可维护性，但当IT人员从这些新技术上获得便利、敏捷的同时，新技术的生态链连贯性不足、可靠性未经历考验，这都是制约平台技术更新换代的最大痛点。

究其根源，不是新技术不适用，而是每项新技术仅能解决软件工程的某一方面的问题，但平台不单只是软件工程，而是集开发、测试、部署、运维、集成等一系列的工作的IT生态圈，航运EDI平台缺少的是一个连贯性好、技术门槛低、开放性佳、可方便扩展、能在未来5到10年内适用技术发展的IT生态环境。

二、EDI平台云生态愿景规划

良好的IT生态应该是简单地开发、完善地测试、快捷地实施、连贯高效地运维、持续地集成、持续地交付，这些应该都是云化的，各个阶段都有完善的解决方案和工具支持，这就是EDI平台云生态的愿景。

开放方面，应使用市场成熟、生命力旺盛、学习成本低、具备良好延续性的技术。开发语言、框架应是当下流行的技术，智力资源易于从高校或社会中获取，拥有一定的社区群体，遇到问

题能有多渠道途径解决，且开发人员应不需要很高深的专业开发能力，经过简单培训即可参与并完成功能服务开发，平台应能方便地拆分模块外包开发，并顺利组装对接。

测试方面，应具备与单元测试、自动化测试、回归测试工具良好结合的能力。

运维方面，新技术不应导致运维成本增加，而应尽可能融合云计算、容器、自动化技术提高系统可用性、鲁棒性，降低运维难度及人力需求。各微服务模块能通过Docker实现云环境自动化部署，各微服务具备路由及负载均衡能力，整体平台不存在单点故障，支持不停机平滑升级、快速高效地运维配置热更新、版本控制。

集成扩展方面，平台应可以从最小化系统功能进行搭建，逐渐集成扩展堆积新功能模块，而不影响平台代码可维护性及运行效率，并能通过简单硬件扩展提升平台服务能力。

三、基于SpringCloud的IT生态环境设计

航运EDI平台从单节点系统向SOA架构变革的过程，随着服务的增多，逐渐暴露出管理难的问题，在考察对比了阿里巴巴的Duboo服务治理架构及SpringCloud分布式架构后，最终选定了SpringCloud架构作为构建平台云生态环境的基础。

这里简单介绍基于SpringCloud的总体架构设计，重点介绍平台开发及运维生态环境的规划设计。

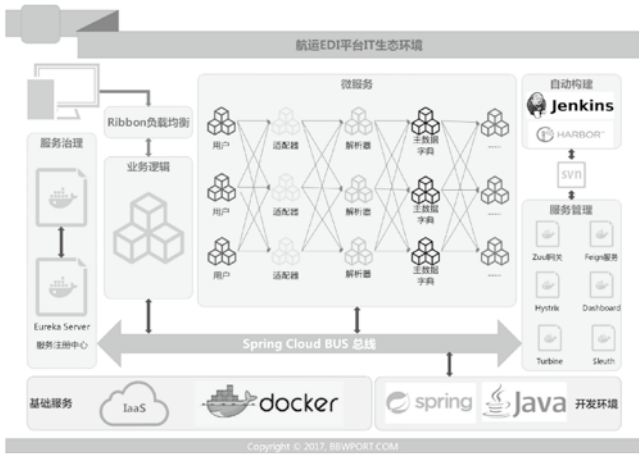


图1

3.1平台整体架构设计

图1为平台整体架构^{[2][3]}：

考虑到需构建结构完善、智力密集、人工成本合理、能应对

人才流动性冲击、可持续发展的开发团队，因此，选择生命力顽强、延续性好的通用性技术以降低学习门槛、应对冲击是至关重要的，这里选择了Java开发语言、Spring体系作为基础搭建开发生态环境，Java及Spring已经成就的大生态环境，能很好地解决组建、运营开发团队遇到的问题。

同样是在底层的平台运行生态环境，这里选择IaaS基础云、Docker容器技术进行搭建，经过近10年的发展，IaaS和PaaS云技术已经落地成熟，IaaS与Docker完美结合，SpringCloud通过SpringBoot与Docker无缝对接，在这些技术的基础上，可极大地降低平台部署运维难度，终结了起早贪黑、提醒吊胆、全天候值守的运维时代，是一个较为完美的平台运行生态环境^{[2][4][7]}。

服务层采用为微服务架构的设计思想，微服务较SOA架构更轻量级，是对SOA架构的更小划分，将SOA的思想融入到单个业务系统内部，实现真正的组件化^[5]，这里构建了用户、日志、审计、权限、文件、报文协议适配器、报文解析器、主数据字典、报文生成器等微服务，各服务独立成系统，拥有自己的数据库，业务逻辑。这样的服务生态环境设计，能很好地将平台复杂的功能分解成一个个功能独立的单表维护组件，符合软件工程总体规划分步实施的原则，让开发外包、通过简单扩充智力资源提高开发效率成为可能，且随着微服务的组件化，服务的复用度、开发难度大大降低^[3]。

随着微服务架构的应用，势必导致服务碎片化，管理难度会随着微服务的增加而呈现几何级增长，为了解决这一问题，这里构建了服务管理生态环境，包含配置管理、服务发现、熔断控制、服务路由、负载均衡，服务状态监控，服务路径跟踪、ESB总线等，将微服务的变化产生的涟漪进行有效监控，结合IaaS和PaaS的高可用特性，通过简单地重启、重新部署等方式解决服务生存周期出现的问题^[8]。

整个EDI平台生态环境还包括自动化测试构建、平台健康管理生态等等，这里就不过多介绍了。

3.2平台开发生态环境设计

为确保航运EDI平台能顺利更新换代，平台开发环境的生态体系至关重要。

1.选择Spring家族产品，可有效解决技术门槛、学习环境、资源获取、人才需求、疑难解决问题，这为开发生态环境打下了良好的基础。Spring系列架构从2004年至今，其引领的

SSH俨然已成为JAVA学习的必修课程，作为SpringCloud底层的SpringBoot更是将Spring的学习门槛降低到了极致。

2.通过Spring Cloud Netflix组件轻松实现服务治理，解决SOA架构下服务调用困难的问题。使用Eureka可完成服务的注册和发现，使用Zuul网关实现服务的路由及访问，使用Feign实现服务间的互访，使用Hystrix组件实现断路器模式，解决微服务下单点故障传播后的“雪崩”问题^[6]。

3.使用Spring Cloud Bus 将分布式的节点和轻量的消息代理连接起来，通过消息总线的模式，结合异步执行，解决服务串行调用开销过大导致的性能问题，并能在消息总线上方便地实现微服务的热插拔，在松耦合的环境下，解决平台基础扩展及动态交付问题^[6]。

4.SpringCloud通过SVN与Jenkins、harbor结合，可实现与Docker的无缝对接，实现自动化构建部署，打造高效的开发编译环境。

通过以上设计，平台将具备完善的开发环境生态链，即获得了便利、敏捷，又解决开发混乱、管理困难的问题，形成健壮的开发生态环境，而不至于因某个服务或个人的异常而影响整个平台质量。

3.3平台运维生态环境设计

运维作为整个生态环境里的重要组成部分，对航运EDI平台至关重要。系统的性能、稳定性、可靠性、不间断的服务、无缝升级，都是航运EDI平台所必须的，这些将是构建运维生态环境的重要考量因素^[9]。

1.通过Ribbon组件提供的负载均衡功能，可通过Docker在很短时间内拉起多个相同功能的服务，共同注册到Eureka上，实现服务的负载均衡，可有效解决系统性能瓶颈问题^{[7][10]}。

2.通过Ribbon负载均衡组件结合Hystrix断路器组件，可轻松实现服务的高可用，确保业务的稳定连贯，让不间断服务成为可能，在此基础上，单服务的无缝升级也能轻松实现，分布式服务优势在运维生态上得到完美体现。

3.Hystrix Dashboard监控面板及Turbine监控聚合组件无需做过多的配置就可以实现对所有微服务实例的运行状态、性能、服务调用消耗等信息进行统一查看，让运维更精准及时。

4.微服务运维过程中，最繁琐的当属服务配置更新，当微服务数量达到一次程度时，对所有服务进行一次配置更新不但费时

费力,还容易出错,管理相当困难, Spring Cloud Config提供了服务配置中心的管理概念,可与SVN、Git进行结合,对环境配置进行版本管理,与Spring Cloud Bus总线结合后,还能实现微服务配置的自动批量更新,这是运维生态环境的有力支撑^[2]。

5.随着EDI业务的不断扩张,服务之间互相调用会越来越复杂,业务逻辑层的一个接口可能需要很多个服务协同才能完成这个接口功能,如果链路上任何一个服务出现问题或者网络超时,都会形成导致接口调用失败, Spring Cloud Sleuth提供在分布式系统中追踪服务调用的解决方案,以及可视化的呈现。

通过这些有益的设计,平台具备简单、高效的运维生态环境。

四、结束语

本文较为宽泛的对航运EDI平台提出了IT生态环境设计,而SpringCloud作为新生分布式架构套件,具有连贯性好、健壮性强、全组件化的优点,且文中设计的IT生态环境在众多行业中具有共性,有着非常强的应用价值。■

(作者单位:广西北部湾国际港务集团有限公司/广西北港电子商务有限公司/广西电视台)

参考文献

[1]倪莹莹,王晓峰,毕坤.基于模式类的航运EDI共享系统的

设计与实现[J].上海海事大学学报, 2016.

[2]Pivotal Software, Inc..Spring Cloud[J].<http://projects.spring.io/spring-cloud/>.2017.

[3]郭栋,王伟,曾国荪.一种基于微服务架构的新型云件PaaS平台[J].信息安全, 2015.

[4]shuijieshuijie.Spring Cloud和dubbo对比[J].<http://blog.csdn.net/shuijieshuijie/article/details/53133082>.2016.

[5]风的姿态.我所理解的SOA和微服务[J].<http://www.cnblogs.com/fengzheng/p/5847441.html>.2016.

[6]Long J.Building Microservices with Spring Cloud - O'Reilly Media Free, Live Events[J].2017.

[7]师德清.基于Docker的PaaS架构设计研究[J].信息与电脑(理论版), 2017.

[8]谢璐俊,杨鹤彪.基于Dubbox的分布式服务架构设计与实现[J/OL].软件导刊, 2016.

[9]蒋炜,刘远,王宁.电力企业一体化云运维管理平台研究[J].信息通信, 2015.

[10]陈世宜,叶德建.基于SOA架构的新型云平台服务管理中间件[J].微型电脑应用, 2016.

★ 简讯 ★

天津机场拟建国际航空物流中心

国家发改委、民航局日前印发的《推进京津冀民航协同发展实施意见》提出,2020年,北京新机场建成投入使用,首都机场国际旅客占比提高2至3个百分点,北京“双枢纽”机场与天津机场、石家庄机场实现与轨道交通有效衔接,初步形成统一管理、差异化发展的格局,京津冀机场群协同发展水平显著提升,整体服务水平、智能化水平、运营管理力争达到国际先进水平。《意见》明确了十项重点任务,在这十项重点任务中,包括改扩

建天津滨海机场,增强区域枢纽作用,建设我国国际航空物流中心;加快完善天津机场、石家庄机场区域枢纽航线网络布局,提升区域机场综合保障能力;以天津机场为核心,加快京津城际铁路延伸至天津机场、京滨城际铁路、天津地铁Z2线建设,有序实施沿海高速铁路天津至潍坊段、天津至承德铁路等;鼓励“三地四场”主要基地航空公司实行枢纽化运营,支持天津机场发展航空物流;实施京津冀144小时过境免签等。