

微服务架构及相应云平台解析

刘 为

(武汉商学院信息工程学院 湖北·武汉 430056)

摘 要 本文介绍了主要软件服务的设计架构,分析了新兴的微服务架构,并对其实现的先决条件——容器化技术进行了研究。文章还分析了两种微服务架构在云服务中的实现形式。

关键词 服务架构 微服务 容器 CaaS BaaS

中图分类号: TP309

文献标识码: A

DOI:10.16400/j.cnki.kjdx.2017.01.013

Micro Service Architecture and the Corresponding Cloud Platform Analysis

LIU Wei

(School of information engineering, Wuhan Business University, Wuhan, Hubei 430056)

Abstract This paper introduces the design framework of major software services, analyzes the emerging micro service architecture, and studies the prerequisite of implementation - container technology. This paper also analyzes the two micro service architecture in the form of cloud services.

Keywords service architecture; micro service; container; CaaS; BaaS

1 网络服务架构

现在社会的各行各业均无法离开网络服务系统,例如学校有学生管理系统、教务系统等;企业有人事系统、财务系统、客户关系管理系统等;手机上运行的各种APP以及游戏,都要与后台服务器相连接,接受后台系统的服务。

软件系统的服务主要是为软件的功能提供支持,以及为其它的服务提供接口。随着网络的普及,软件的服务与软件本身往往是分离的,服务由网络上的服务器提供(后台),而软件运行在本地或浏览器上(前端),后台服务是软件开发中最关键的部分。

1.1 单体服务架构

上世纪90年代以前的软件服务多为单体架构,所有的功能采用集成化、过程化的开发,之后被编译为一个文件并放置于容器中运行,这种架构易于开发、部署和测试,但其主要问题包括:(1)代码集成在一起,几乎无法协同开发。(2)代码功能高度耦合,后期维护、功能扩展困难。后期系统业务变更或服务整合会导致整个系统需要重构,大幅提高IT实施的成本。(3)小修改就可能导致重构整个项目,迭代时间大幅提升。(4)稳定性差,由于系统高度集中在一起,任何一个小的问题可能导致整个系统瘫痪。

1.2 SOA 架构

随着软件的模块化概念开始流行,一种被称为面向服务的架构(SOA, Service Oriented Architecture)在2000年后被软件企业广泛应用。SOA架构基于服务(即基于模块化的组件),服务提供接口,服务间可通过XML进行信息交换。SOA基本架构包含ESB(Enterprise Service Bus),Web服务、XML和SOAP。SOA的最大特征就是松耦合,结合分层次的开发理念,解决了协同开发、测试等问题,也更易于后期扩充功能。但SOA本身只是一个架构,并非实施标准,所以在生产环境中应用SOA存在一些问题:

(1)需要共享ESB,松耦合服务实际边界非常模糊,修改某一模块会导致需要修改其它模块,从而导致代码维护困难,系统迭代速度大幅降低。(2)系统后期扩充将导致系统臃肿、性能下降;某些功能扩充甚至需要部分重构ESB及其它服务。(3)系统开发长期被一种开发技术绑定,开发人员、技术、架构变更的灵活度大幅下降。

2 微服务架构

微服务架构并非一个全新概念,它更像是SOA架构的一种实现。微服务架构依然是面向服务,但是其将松耦合做到了极致。该架构是去中性化的,其中没有ESB,每一个服务可以单独开发、测试、运行和部署,甚至能够有自己的数据库。服务间的通信与开发语言无关,一般采用基于HTTPs的RESTful API(Representational State Transfer API)。微服务的出现顺应了敏捷开发的浪潮,先开发先上线,后开发再扩充,能对系统进行快速迭代。微服务架构有以下特点:

(1)服务小:每一个服务代码量少、复杂度低,仅专注某一项功能。(2)能够独立运行:每个服务可以运行在独立的进程中。(3)与语言无关的通信机制:例如XML、JSON、REST API等。(4)松耦合:开发、部署和运行均处于独立状态,几乎无外部依赖。(5)去中心化:没有ESB,可完全分布式部署。(6)数据独立:微服务可以有自己的数据库,其它服务只能通过接口获得该服务的数据。

2.1 微服务架构的优势

根据微服务以上的特性,实施过程中有如下优势:

(1)依照服务来划分开发团队:每一个服务对应一个完整团队(包含后台、前端、数据库、中间件等),从开发、测试、上线及后期维护均由该团队负责,一个跨职能的团队能够完全掌控自己的微服务。

(2)服务的异构性:能够针对不同业务选择合适的开发方案、开发语言、框架及部署环境,无须像单体或是SOA架构一



样,选择统一的技术方案。在传统架构中,初期技术方案一旦选定,很长时间内,整个系统就会在所选技术框架内进行开发,到了后期想要尝试新的技术,则有很大可能要重构系统,不但开发难度大,而且项目越大,失败风险越高。而微服务架构采用的是独立的扩展方式,不但无须重构原系统,还可以极小风险测试新的服务,一旦新服务达不到预期,则可直接终止,这也仅是停止使用一个服务而已,对整个系统影响极小。

(3)独立测试、部署及容错能力:可以对微服务进行单独的测试和部署,无须对整个系统进行测试、编译和从新部署,降低了系统运行风险。当一个微服务出现运行故障时,不会影响系统中其它的服务,避免了系统全面停摆。

(4)强扩展性和可控的系统复杂度:每个微服务通过定义良好的接口实现服务间边界,专注于某一种功能,从而能够根据系统需求,实现细粒度的自由扩展。同时,由于单个微服务复杂度低,即使出现多个服务堆叠的情况,也能够较容易掌控整个系统的复杂度。

2.2 微服务架构的不足

(1)部署较以往架构更加复杂:系统由众多微服务搭建,每个微服务需要单独部署,从而增加部署的复杂度,容器技术能够解决这一问题。

(2)性能问题:由于微服务注重独立性,互相通信时只能通过标准接口,可能产生延迟或调用出错。例如一个服务需要访问另一个服务的数据,只能通过服务间接口来进行数据传输,如果是频繁访问,则可能带来较大的延迟。

(3)数据一致性问题:作为分布式部署的微服务,在保持数据一致性方面需要比传统架构更加困难。

3 微服务的云平台

由于云平台的广泛使用,很多网络服务的后台会部署在云端。基于 PaaS(Platform as a Service)的云平台非常适合微服务后台的构建,容器技术则是云端微服务的基础。

3.1 容器技术

微服务系统采用类似于搭积木的构建方式,开发一个服务就可上线一个服务,这就意味着每次部署新的服务,不能影响其它已存在的系统。更重要的是,同一个系统的微服务可能采用了不同的开发技术、数据库等,与原来存在的服务所使用的技术完全不兼容,如果需要加入新的服务,还需要为其搭建不同运行环境。为了解决这些问题,容器技术成为了最好的选择。容器的广泛应用并非因为微服务架构,但其却成为了微服务架构实践的先决条件。

容器是 Linux 系统下用以实现在单一主机提供多个隔离的 Linux 系统环境的虚拟化技术。与虚拟机不同,容器无须运行虚拟操作系统,而是共享本地主机的操作系统来实现虚拟环境。容器技术最先在 2013 年由 Docker 公司应用于自家的 PaaS 云服务平台,并迅速被广大开发者所认可,越来越多的开发者开始将网络服务部署在容器中。对于容器技术而言,现在处在起步上升阶段,还面临着一些问题,例如容器系统的容错性:当宿主机出现故障,如何能让容器在其它机器上迅速运行起来;以及容器的扩容性:一个宿主机能够容纳的容器是有限,如何进行后期分布式扩容等,这一切集中在于系统需要一个好的容器编排工具管理容器。

3.2 CaaS

CaaS(Container as a Service)云服务是一种完全基于容器的 PaaS 系统。平台内有容器镜像池,包含了各种各样的镜像,镜像实际是封装好的包含各种软件所组成的运行环境。镜像可以由平台提供,也可以由个人发布共享,平台用户也可以自己创建镜像,并且不用担心平台无法运行镜像。

CaaS 云服务特别适合微服务的部署,因为每一个微服务可以单独部署在一个容器中,整个系统由容器搭建而成,包含不同技术架构的容器之间严格隔离。当开发者部署某一服务时,只需选择(或自己搭建)镜像,并将服务部署在其中,即可让其在云台上运行。

3.3 BaaS

BaaS(Backend as a Service)是一种能够直接提供微服务的云平台,其实际是 SaaS(Software as a Service)架构,一般服务对象是移动应用开发者。平台内提供了移动应用常见的后台服务,包含存储、即时通讯、数据统计和分析、消息推送、应用内搜索、用户反馈、第三方登陆、分享等移动应用中通用的功能。移动开发者在选用平台功能后,能够大幅降低开发难度和时间。BaaS 平台实际也是基于微服务,因为其功能可以在后期不断扩充。

4 结语

微服务是近 3 年出现的新技术,就现在的关注度而言,极有可能成为未来构建系统服务的主流架构;相应的,容器技术有可能取代虚拟机,成为服务器上最重要的虚拟化技术。

★基金项目:1、武汉市教育局教学研究项目《基于移动设备的游戏设计课程体系研究》(编号 2015090)

2、本研究获得武汉市属高校数字城市专业重点实训基地资助

参考文献

- [1] 邓杰文,曹彩凤.微服务若干关键问题研究[J].五邑大学学报(自然科学版),2016,30(2).
- [2] 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(11):15-20.
- [3] 鞠春利,刘印锋.基于 Docker 的私有 PaaS 系统构建[J].轻工科技,2014(10):80-80.
- [4] 刘思尧,李强,李斌.基于 Docker 技术的容器隔离性研究[J].软件,2015,36(4):110-113.
- [5] 陈春霞.基于容器的微服务架构的浅析[J].信息系统工程,2016(3):95-96.