

# 微服务架构和容器技术应用分析

李红健

(湖南科技学院, 湖南 永州 425199)

**摘要:** 文章分析了微服务架构和容器技术的应用。微服务架构是一种架构概念, 通过将功能分解到各个离散的服务中以实现对解决方案的解耦, 在降低系统耦合的同时, 还提供了十分灵活的服务支持, 这种架构对应用的功能进行了充分的分离, 使开发和部署非常的便捷, 结合容器技术, 最大化地解决微服务架构中所面临的负载, 对微服务进行更有效的支撑。微服务架构中对容器技术的应用主要看中容器的对解决复杂环境和用户资源隔离问题的能力, 这种新的开发方式为开发者提供了一种新的思路。

**关键词:** 微服务; 容器; 共享; 解耦

微服务架构的产生是软件架构不断演进的结果。Web开发经历了单体结构, 集群结构, 分布式系统, 最终演化到微服务架构。微服务架构按照业务划分模块, 实现一个个高度解耦的系统架构, 其中的分布式、快速演化、自动化运维和高容错性这些优点, 受到了大批互联网公司的青睐。在保障软件架构的灵活伸缩, 系统架构分布合理的同时, Docker容器的加入, 进一步降低了微服务的成本, 反过来也使得微服务和容器结合得越来越紧密。特别是云端平台的兴起, 加速了这种趋势<sup>[1]</sup>。

## 1 设计架构的演变

### 1.1 单体结构

此时公司业务量比较小, 系统构建并不复杂, 所有的代码, 数据库, 文件都部署在一台机器上, 对系统服务进行常规的应用服务和数据服务分离, 增加缓存可以满足此时的访问量。

### 1.2 集群结构

公司业务逐渐增加, 用户量增大, 服务器压力也随之增大, 高峰期已经不能满足用户的操作, 于是增加服务器分担原有服务器的访问和压力, 同时有了负载均衡, 反向代理, 数据库读写分离的应对方案。

### 1.3 分布式系统

公司业务继续发展, 用户规模和业务复杂已经到了一个量级, 于是开始把一个系统拆成许多不同的应用, 每个应用进行独立的开发、测试、运维, 应用之间通过消息队列来进行数据分发, 也可以访问同一个数据存储来构成一个关联完整的系统。

### 1.4 微服务架构

微服务架构以去中心化为特点。大规模用户的使用需求, 对分布式系统的要求很高, 并且业务快速发展, 迭代周期很短, 子系统也不需要如原来企业计算分布式那样采用集中式存储, 通常采取前后端分离的方式, 使单个业务系统组件化, 不同的服务之间采用轻量级的交互机制进行交互, 使各个子系统做到有效分割, 结合Docker容器, 使得微服务能进行实际的应用。

## 2 微服务的特性

### 2.1 微服务架构的优势

#### 2.1.1 降低复杂性

微服务架构通过分解单体式应用为多个服务方法, 降低

了系统的复杂性, 多个子系统的分离实现了组件化, 一个个组件成为可管理的分支或服务, 使其通过模块化的方式呈现出来。通过微服务这种架构模式, 让单个服务更容易开发和维护。

#### 2.1.2 部署的独立性

每个微服务都具备相对独立的运行进程和业务处理的能力, 所以每个微服务的安装和部署都可以独立进行。在传统的架构中, 如果要对某一程序内的某一功能区进行变更, 就需要对整体架构进行统一的重新建设, 并进行重新部署。而微服务架构在变更时, 由于其单个微服务的独立性, 不需要对整个应用进行编译和部署, 这大大提高了效率, 也降低了对系统环境所造成的风险, 缩短了原有的时间操作周期。

#### 2.1.3 容错性高

在传统单一的构架中, 如果某一组的应用功能发生故障, 那么系统整体的稳定性就会受到影响, 子系统出现故障, 影响会在其他的子系统中蔓延, 轻的后果是会导致局部的系统受到影响, 部分业务无法得到处理, 严重的后果是导致整个应用系统的崩溃。在微服务构架中, 由于单个服务的独立性, 所以故障的影响可以控制在单个的应用中, 并不会对其他的服务造成影响, 而且微服务中有提前审计的功能, 有多种机制可以保证应用运行的稳定<sup>[2]</sup>。

#### 2.1.4 技术灵活

微服务构架有多种技术的选择, 常见的Java, NodeJs, Python, React Native都可以实现具体的微服务逻辑, 也可以混合使用。通过对技术的合理选择, 节省成本的同时, 做到专业分工。在使用不同的技术对微服务架构进行整合和部署的过程中, 由于微服务的相对简单, 在升级的时候面临更小的破坏风险, 微服务的技术重构也更具有可行性<sup>[3]</sup>。

#### 2.1.5 易扩展

微服务的架构中, 系统的独立性比较强, 保证了系统的扩展性也比较强, 其扩展的方向也相对较多, 在横向的扩展中可以使相同维度的业务实现无缝衔接, 当不同的组件间接口出现差异的时候, 其独立性也可以大大降低衔接的复杂性。在纵向的方向上, 组件化也使得多个维度的业务能有序地进行数据交互, 微服务架构的特点大大降低了扩展风险。

### 2.2 微服务架构面临的问题

微服务的复杂性主要体现于分布式这种架构方式上。

由于应用的是分布式系统,给开发时带来了天然的系统复杂性。开发者需要在RPC或消息传递之间选择进程间通信机制,更甚于开发者必须平衡消息传递过快或者过慢的问题,显然这加大了服务的复杂性<sup>[4]</sup>。

另一个挑战是数据库分区架构。在微服务的应用中,应用需要同时服务多个数据库。NoSQL数据库和信息代理的机制的系统,并不支持分布式交易,这对开发者带来了很高的挑战。

### 3 容器技术

容器技术的使用,使得微服务器架构中所面临的压力得到很大程度上的缓解。容器技术的特点为微服务构架提供了落地的机会,其中的内核机制可以实现不同的容器之间的联系,容器之间资源也能实现完全的隔离,它们中有一个典型的代表——Docker。

容器技术的高速发展为计算机的云计算问题提供了解决的可能,现在多重容器技术都已经成为容器的标准规范。Linux容器具有较多的功能,也实现了十分规范的管理。微服务通过将单个应用程序分解,实现了组件化,又通过Kubemetes等技术将原有的集群统一地编织在一起,提供应用的部署、维护、扩展机制等功能,实现对不同容器的有效管理<sup>[5]</sup>。

Docker是容器技术中的代表,特点体现在具有标准的镜像结构,实现了对不同资源实行不同存储的功能,也能满足大规模的托管服务,对于有主机集群的云服务平台,通过分解应用构建、发布等方式实现对云计算技术的开发,在实现云计算平台的构建的同时,还可以进行优化和自动化维护环境,使得工作的效率能够得到有效的提升,在降低成本的同时,满足了微服务架构所需要的资源。

Docker的体系中,最关键的有两个,Docker Register和Docker Engine,前者负责构建和分发应用镜像,后者负责构建容器。这种组合方式,是云服务的软件即服务(Software-

as-a-Service, SaaS)理念,用户可以在各自的数据中心内建立私有的Docker Register,形成属于自己的私有集群,以应对大规模的应用扩展需求<sup>[6]</sup>。Docker很像一个集装箱,通过Lxc技术先进行整合镜像,再集中汇总进行分发。

普通的虚拟机与容器技术有一定的相似性,但是容器技术在很多细节和虚拟机并不相同。虚拟机建立在硬件平台上,而容器技术建立在操作系统(Operating System, OS)上,可以把容器看作是虚拟机轻量化的实现。Docker在实现应用隔离的同时,没有虚拟机必须的虚拟化管理层,对比虚拟机太长的启动时间,容器的启动与停止可以在毫秒级这个范围内启动。比较这些特性,Docker容器显然更胜虚拟机一筹。

### 4 基于容器的微服务应用

Docker的细粒度松散耦合和微服务架构相得益彰。我们可以让Docker容器装载这样一个场景功能,按照不同的角色分类,每一个容器里装一个服务和应用,一个服务器中运行多个容器,也可以将多个容器分散到多个服务器上运行。整个项目架构按照业务逻辑的规划以细粒度的方式分散到了各个Docker中,并可以根据Rest接口的方式进行整合联动<sup>[7]</sup>。一个典型的例子可以是负载均衡层、综合业务服务层、单业务服务层、存储层。这种多分层的方式,可以很好地保证容器对微服务的支持,高效地保证每一层服务的运行。当然,这种应用方式也是有些许弊端,在架构设计的前期,需要花费较多时间来进行详细的系统分析和逻辑划分。

### 5 结语

微服务架构和容器的结合在程序开发中应用已经成为一种新的开发方式,通过不同的微服务实现业务架构的粒度化,通过不同的容器承载不同的业务,为用户提供更多的开发选择。微服务构架中采用容器技术后,一方面更加微型化;另一方面容器使微服务开发更加的便捷。这种开发方式将随着时间的推移越来越流行。

### [参考文献]

- [1]杨鸣,张羿,耿贞伟.微服务架构在容器云中的应用实践[J].电脑与电信,2017(7):79-81.
- [2]张晶,黄小锋,李春阳.微服务框架的设计与实现[J].计算机系统应用,2017(6):259-262.
- [3]王纪军,张斌,顾永生,等.云环境中Web应用的微服务架构评估[J].计算机系统应用,2017(5):9-15.
- [4]刘为.微服务架构及相应云平台解析[J].科教导刊,2017(1):27-28.
- [5]佚名.容器+微服务成为驱动混合IT关键[J].邮电设计技术,2017(1):5.
- [6]黄小锋,张晶.微服务框架介绍与实现[J].电脑与信息技术,2016(6):14-16.
- [7]王健,李冬睿.从单一模式系统架构往微服务架构迁移转化技术研究[J].科教导刊,2016(9):43-44.

## Analysis on the application of microservice architecture and container technology

Li Hongjian

(Hunan University of Science and Technology, Yongzhou 425199, China)

**Abstract:** This paper analysis on the application of microservice architecture and container technology. The infrastructure of micro services is an architecture concept, to achieve decoupling of the project, the micro services decomposing the functionality into various discrete services which can reducing system coupling, it also provides very flexible service support. This infrastructure is fully separated from the functions of the application making it extremely convenient to develop and deploy. Combined with container technology, can be maximized to solve the load in the micro service architecture and support the micro service more effectively. The application of container technology in the micro service mainly focuses on the capability of the container to resolve complex circumstances and user resource isolation. This new development regime provides a new way of thinking for developers.

**Key words:** micro service; container; sharing; decoupling