讨论课 03

1352839
饶伊文
项目 git 链接：https://github.com/BrunoQin/Open-Reuse

一、讨论课内容要求：

侧重于容器技术，讨论各种方案以及挑战

- 容器技术的理解
  - 关键技术
    - 隔离
    - 安全
    - …
  - 挑战
    - 管理(Orchestrating at Scale)
    - …
- 如何复用
  - 我们需要和希望解决的问题
  - 选取什么样的容器技术方案

二、内容分析与业界参考方案：

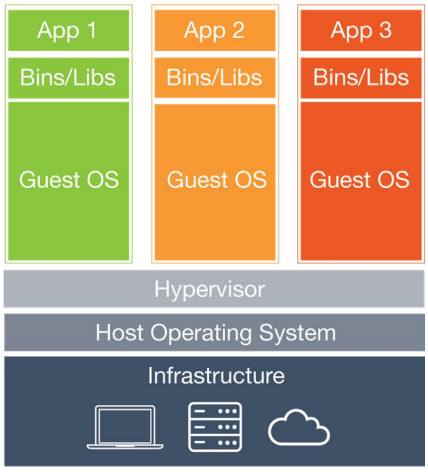
了解容器服务之前先让我们普及一下 Docker：

Docker 是一个开源的应用容器引擎，让开发者可以打包他们的应用以及依赖包到一个可移植的容器中，然后发布到任何流行的 Linux 机器上，也可以实现虚拟化。容器是完全使用沙箱机制，相互之间不会有任何接口（类似 iPhone 的 app）。几乎没有性能开销，可以很容易地在机器和数据中心中运行。最重要的是，他们不依赖于任何语言、框架包括系统。开发者在笔记本上编译测试通过的容器可以批量地在生产环境中部署，包括 VMs(虚拟机)、bare metal、OpenStack 集群和其他的基础应用平台。

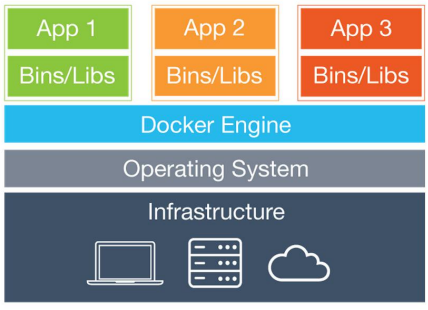
Docker 的目标是实现轻量级的操作系统虚拟化解决方案。Docker 的基础是 Linux 容器（LXC）等技术。

在 LXC 的基础上 Docker 进行了进一步的封装，让用户不需要去关心容器的管理，使得操作更为简便。用户操作 Docker 的容器就像操作一个快速轻量级的虚拟机一样简单。

下面的图片比较了 Docker 和传统虚拟化方式的不同之处，可见容器是在操作系统层面上实现虚拟化，直接复用本地主机的操作系统，而传统方式则是在硬件层面实现。



Virtual Machines



Containers

(图片来自 Docker 官方网站)【1】

可以说，目前市面上面看到的云储存计算平台大部分的引擎内核都是 Docker: 如 WPS 的云文档平台、灵雀云、时速云，放眼国外世界眼光 Docker 的影响力已经得到整个行业中许多大企业的支持，这其中包括了亚马逊、谷歌、IBM、微软、红帽和 VMware 等。【2】

隔离

容器化应用是基石，一切都封装在镜像里

Docker 提供了一种在安全隔离的容器中运行几乎所有应用的方式，这种隔离性和安全性允许你在同一主机上同时运行多个容器，而容器的这种轻量级特性，意味着你可以节省更多的系统资源，因为你不必消耗运行 hypervisor 所需要的额外负载。对于容器云而言，所有的应用都需要容器化以后才能发布，即将应用程序打包进 Docker 容器，以镜像的方式运行。容器化应用未来将会成为云端应用交付的标准。【6】

安全

Docker 是否安全

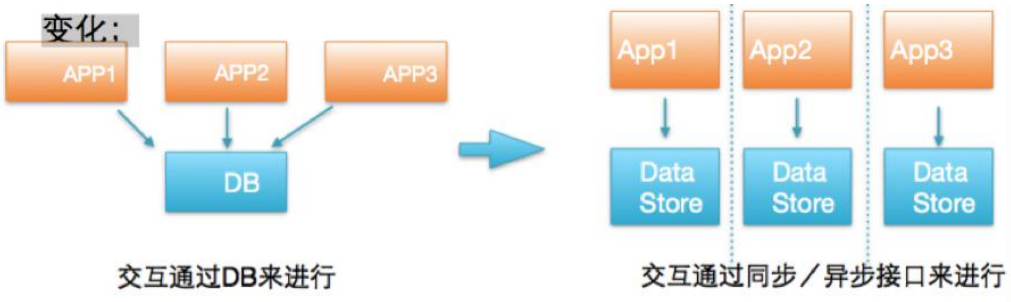
Docker 本身是共享操作系统的进程，不存在不安全一说，如果 Docker 不安全，那么所有的 linux 程序都是不安全的，而目前全球基本上 90%以上的网站都是运行在 linux 上的。Docker 本身是容器技术的一种，所谓容器就像一个盒子，我们只对暴露需要暴露的端口，比如一个网站就只暴露 80 端口。而传统的服务器和云主机，基本是开发了所有的端口，或者是大多数端口，这种暴露其实是很危险的，因此 Docker 反而会让系统更安全可靠。【7】

京东对于微服务底层的技术支持实践

京东资深架构师李鑫主要负责京东的服务框架， 他所分享的内容是对微服务底层的技术框架支持。

为何要微服务化？

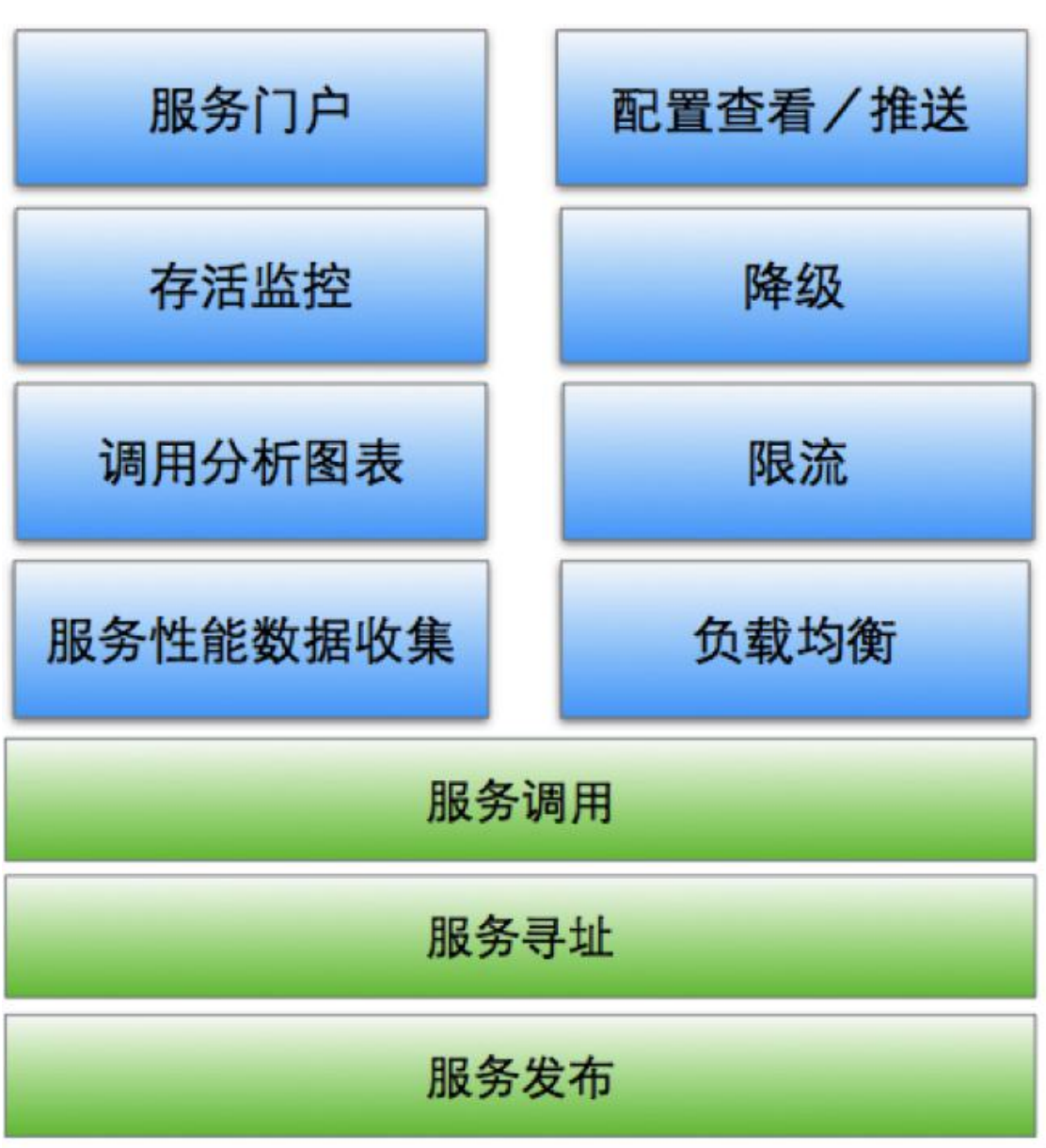
原因有以下几点。
系统规模随着业务的发展而增长，原有系统架构模式中逻辑过于耦合，不再适应；
拆分后的子系统逻辑内聚，易于局部扩展；
子系统之间通过接口来进行交互，接口契约不变的情况下可独立变化。



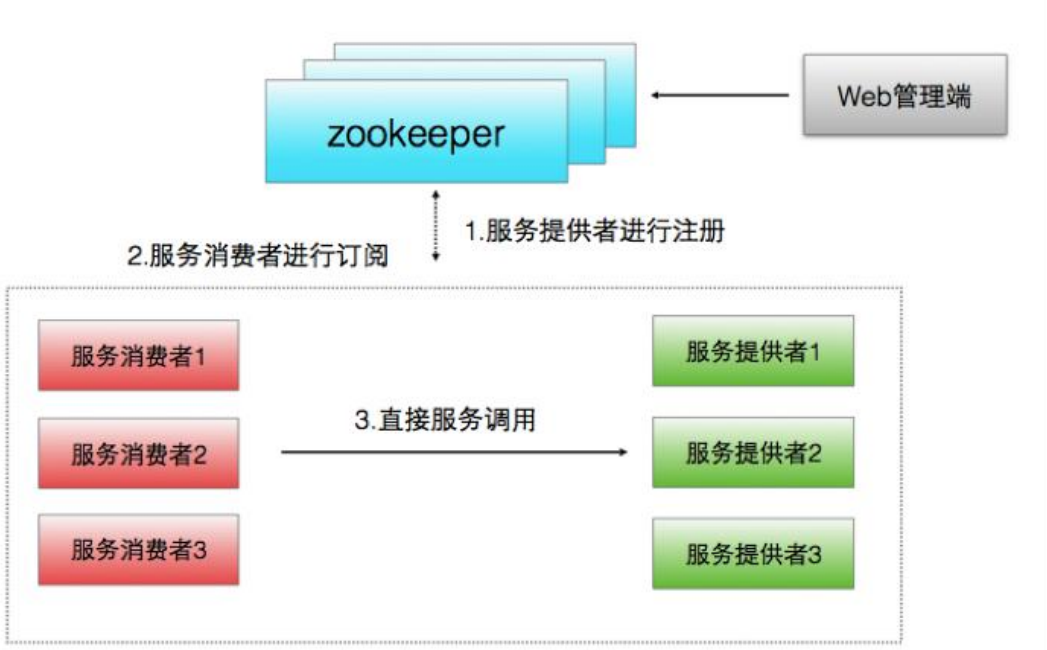
上图中，左边是几年前京东的架构，很多服务都是访问同样一个 DB。这种架构的问题在于：流量来了以后全部压力都在 DB 上。而且在之前京东的架构里比较强调快速开发，很多逻辑比如说仓储配送服务都不存在，全都依靠 BD 来进行。这样可扩展性相当差，性能也不太可控。后来，我们根据业务模块和特性进行水平拆分，应用和应用之间都要通过接口进行交互，有同步和异步接口。

京东服务平台演变历程

下图是京东服务平台的基本功能构成。



在 2012 年初，京东开始做第一代服务框架，用的是 zookeeper 集群作为注册中心，baseon 开源的服务体系。如下图所示。



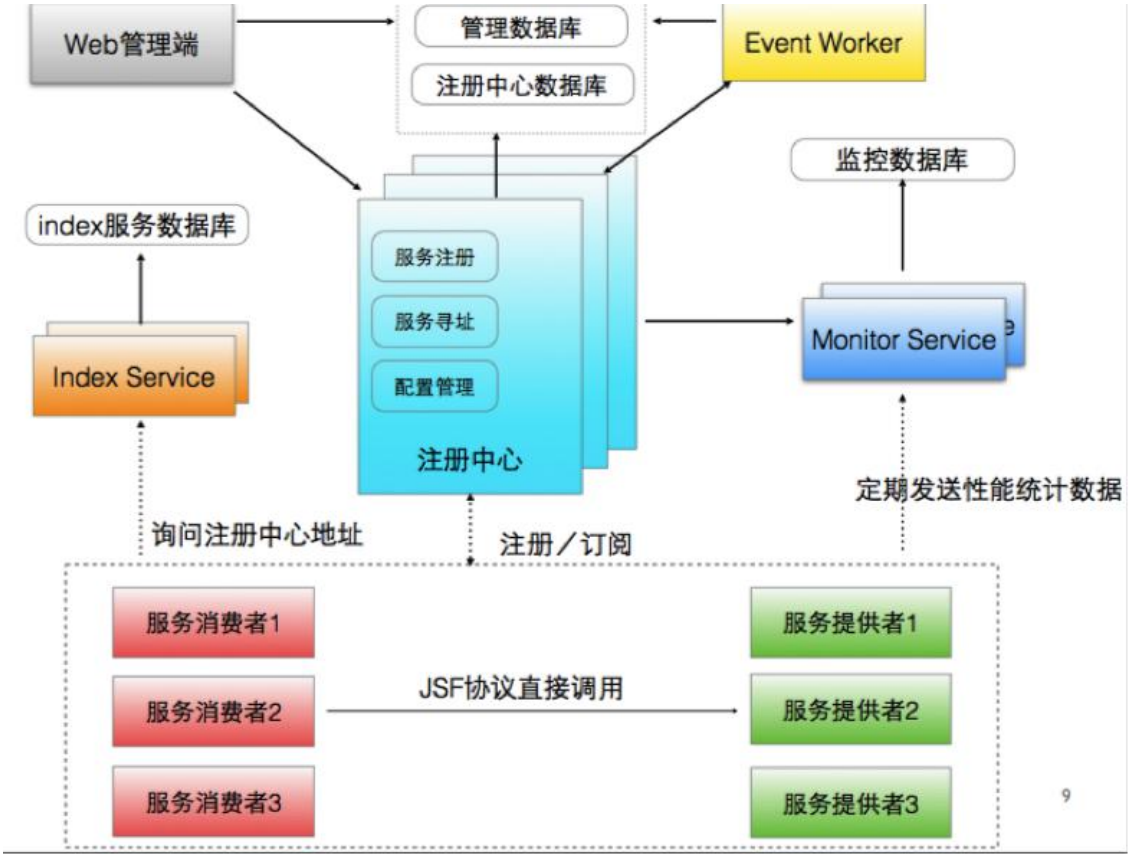
第一代架构在运营过程中，暴露出了很多问题。

1.客户端
许多逻辑放到客户端，客户端逻辑太多，一旦需要修改，就面临升级问题。

2.注册中心
将 ZooKeeper 作为注册中心，功能定制扩展受限。

3.服务治理
缺乏流控手段，大流量打爆线程池；
更改配置需要重启，对于运营不够友好；
缺乏调用监控，没有调用分析图表。

为了解决以上问题，团队在 2014 年推出了新服务平台 JSF，其框架示意图如下。



可以看出，服务注册和寻址没有太大变化，主要变化在于注册中心。采用团队自己写的服务，并提供了 index 服务数据库。相对来讲，询问注册中心地址，会进行一个服务的调用。因为这一块有自己内部的逻辑，没有办法实现，所以定期会发送性能统计数据。把 RPC 转化，生成负载均衡管理重试策略，然后经过序列化以后发送到 server 并解码，再进行实际的业务调用，然后进行一些过滤的逻辑。

京东服务平台核心技术

1.协议
采用异步事件通讯框架 Netty 来实现　络协议栈;
同一端口同时支持 Http、TCP 协议访问，根据数据包情况挂载不同解码器;
TCP　链接下使用用自定义二进制协议；
HTTP　关来应对跨语言言访问。

2.RPC-callback
TCP　长链接是双工工的,服务方方可以主动推送消息到调用用方方;
调用用端检测到参数列表中有 Callback 类型,登记相应的 callback 对象;服务端收到调用用时,生生成相应的反向调用用代理;
服务端持有此代理,并在需要时调用用此代理来推送消息。

3.负载均衡
一个服务至少部署两个以上实例，挂了一个以后，另外一个可以正常服务。服务消费者这一块，有一个负载均衡的算法选择服务提供者，可以设置权重。有一个可用列表，还有一个重连列，系统会定时连接。还有一个是非健康列表，虽然可以长链接建立起来，这个时候给出的反馈是不正常的，维护的时候会把它挪动可用列表里。

4.性能优化
首先有一个批量处理，请求先写入 RingBuffer 为，打成序列化与反序列化这种耗时的操作从 Netty 的 I / O 线程中挪到用户线程池中。启用压缩以应对大数据量的请求，默认 snappy 压缩算法。最后是定制 msgpack 序列化，序列化模板，同时还支持 fast json、hessian 等多种序列化协议。

5.注册中心
京东有很多机房，对于跨机房来说是耗用更多资源，也占用了专线资源。这里实现了优先访问本机房的注册中心，不可用的话才会去连起来机房。对于注册中心来说，后面会连一个数据库。但是这一块连数据库失败的话，也会有一些单点的问题。对于这种情况，实现了一个 LDS，在写数据库的同时会写到本地的存储，在后台进行数据的同步。如果说 DB 服务不可用的话，还是可以取得相应的注册。

6.配置
配置这一块是服务提供者列表维护，动态推送。可以查看当前服务生效的配置和状态，可以看出调用的平均耗时和失败的次数。对服务动态分组无须启动，而且机房也有备份，出现问题直接调用就可以了。

7.限流
每一个服务调用者都有可能成为潜在的 DDOS 攻击者，这里会给服务的所有调用者带上标示，在系统环境变量中带上 APPID。此外，开发了计数器服务，限定单位时间内最大调用次数。如果说你是一个推送服务，限定每分钟四百次，其实调用已经超限了。为了保护服务端，系统同时限定了并发数，服务端执行时检查请求的状态，如等待时间大于超时时间，直接丢弃。

8.降级
每个服务接口的每个方法都有灾备降级开关;
配置 mock 逻辑,返回的结果用 json 格式预先设好;
降级开关打开时将在 consumer 端短路 RPC 调用,直接返回 JSON 结 果。

9.弹性云部署
在京东内部，CAP 负责资源调度，JDOS 负责资源的虚拟和部署。CAP 会定时调度 JSF 监控平台的方法，如果说告诉它调用的次数已经超了，CAP 会调用 JDOS，分配具体的服务资源，把一些新服务自动部署。最终，到注册中心进行注册。

下一步研发方向

首先会做的是服务治理，根据应用 ID 的一系列管理增强。增强接口文档管理，建立接口文档中心，帮助用户使用接口。最后是增强跨语言支持，对于一些比较小规模的，直接通过网站来更新。【3】

三、参考资料：

【1】什么是容器服务（Container as a Service）
http://doc.tenxcloud.com/doc/v1/getting_started/what-is-container.html
【2】Docker 并不神秘 时速云为你讲述“CaaS 容器云”
http://www.csdn.net/article/a/2015-09-10/15828030
【3】基于容器的微服务架构剖析
http://blog.qiniu.com/archives/3900
【4】基于容器云的微服务架构实践
http://dockone.io/article/516
【5】CNUTCon 全球容器大会，灵雀云微服务架构实践引关注
http://www.jianshu.com/p/b100d0aeeba6#
【6】【深入浅出容器云】关于容器云你不得不知的十大特性
http://dockone.io/article/1082
【7】docker 是否安全
http://docs.ghostcloud.cn/faq/dockershi_fou_an_quan.html