

货拉拉应用架构演进，堪称单体落地微服务避坑指南

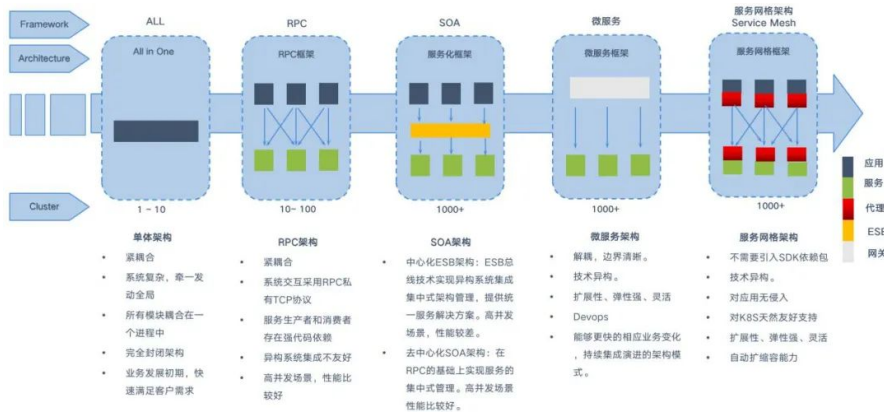
作者：徐少敏 2022-04-28 11:04:27

开发 # 架构 # 新闻

本次分享主要围绕应用架构演进以及货拉拉微服务治理的技术选型等进行思考。

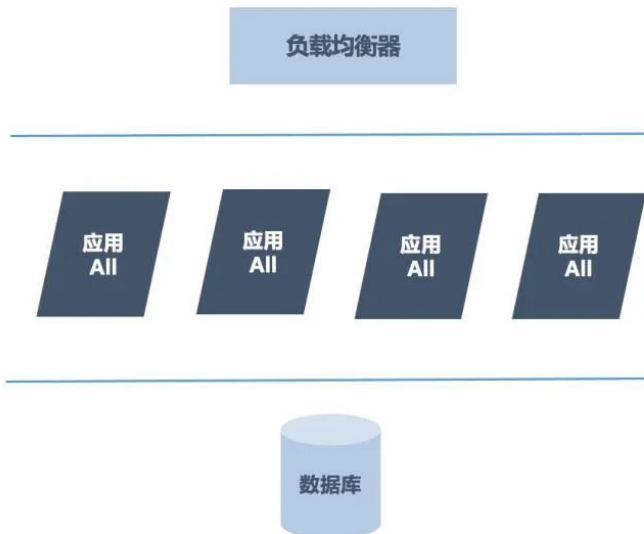
从单体到SOA架构，再从微服务架构到服务网格(Service Mesh)架构，企业应用架构领域每一次技术架构的演进都会给企业带来更多的价值：职责解耦、能力复用、关注点分离、沟通效率提升、快速演进、快速交付和快速反馈。本次分享主要围绕应用架构演进以及货拉拉微服务治理的技术选型等进行思考。

一、应用架构的演进



应用服务架构一直处于不断演进的过程中，上图通过对比5种比较主流的架构模式，展示了应用架构的演进历程和变化。

1、单体架构



同话题下的热门内容

该不该将单体架构迁移到微服务架构

Linux 5.19 正式发布！这次用的相似话题

如何前端一个无代码的自助客

15386内容 JMS VS Kafka：苹果与橘子之

后端 「芯片法案」下周签署！台积电978内容 SK海力士谨慎观望

51CTO技术栈公众号

332内容 Java 服务 Docker 容器化优秀

项目管理 八个流行的 Python 可视化工具1582内容

全部话题 >>

编辑推荐

2017年9月编程语言排行榜：J头还能统治排行榜多久？

2017年最受欢迎的5个前端框架

2017年11月编程语言排行榜：JavaScript

2017年3月编程语言排行榜：S

最近租房有点烦！技术人如何找到如意的“小窝”？

相关专题



直播：昇腾AI开发者创享E

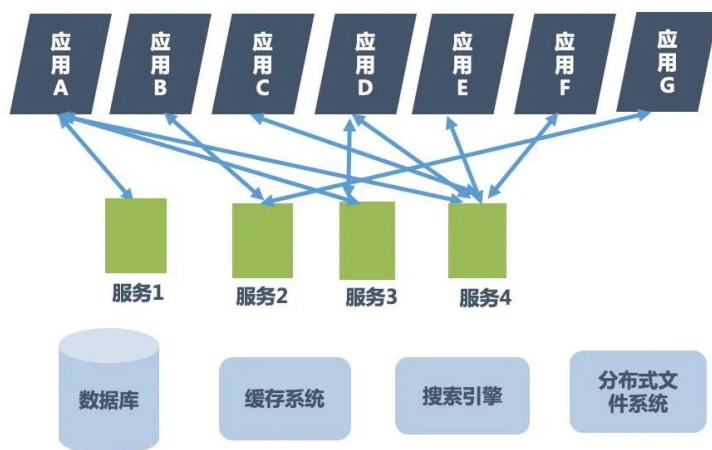


我收藏的内

理能力。

单体架构是围绕web容器打包及部署的架构模式，随着业务的快速发展，要求实现服务的快速迭代和快速交付，应用架构也演进为以服务为中心的架构模式。

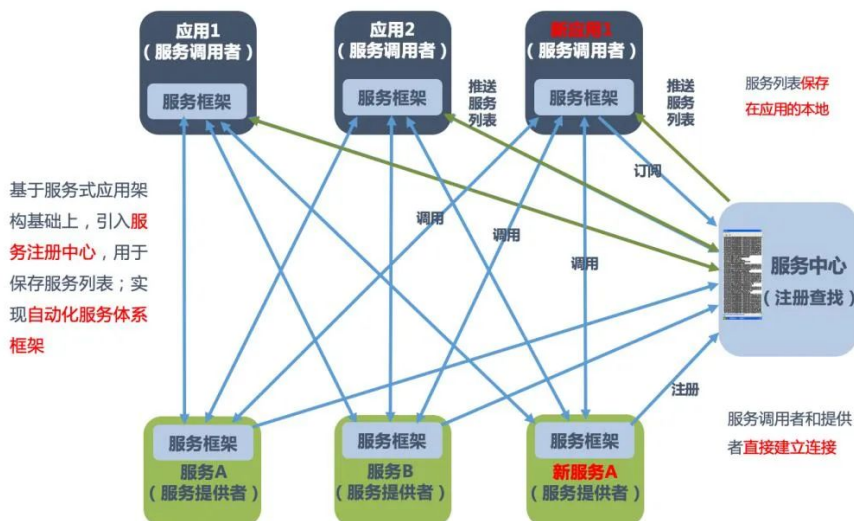
2、RPC架构



RPC架构在现在应用系统的早期还是比较常见的架构模式，就是增加服务层，把冗余的代码和可以复用的业务应用进行拆分提取，封装成服务。系统架构更加清晰，代码质量提高，利于升级和维护，稳定性高。应用层可以更专注于与前端用户交互。业务处理放在服务层来进行，服务和应用的管理不是自动化，服务层能够实现HA的功能，适用中小型WEB系统的场景和高并发场景，性能比较好。RPC就是一个典型的RPC架构。

RPC架构也存在一些问题，通过共享分布式对象实现远程方法调用，如果在其中一个对象中添加一个属性，就会对共享对象的生产者与消费者产生影响，所以RPC架构也是紧耦合的模式。系统交互采用RPC私有TCP协议，服务生产者和消费者存在强代码依赖，对异构系统集成不友好。

3、SOA架构



个人认为SOA架构经历了两个阶段，一是以ESB中心化的架构，二是以注册服务为中心的服务注册发现架构（上图）。

1) ESB中心化架构

同话题下的热门内容

该不该将单体架构迁移到微服务架构

Linux 5.19 正式发布！这次用的 MacBook.....

如何创建一个无代码的自助客户门户

JMS VS Kafka：苹果与橘子之

「芯片法案」下周签署！台积电、三星、SK海力士谨慎观望

51CTO技术栈公众号

Java 服务 Docker 容器化优秀案例

八个流行的 Python 可视化工具

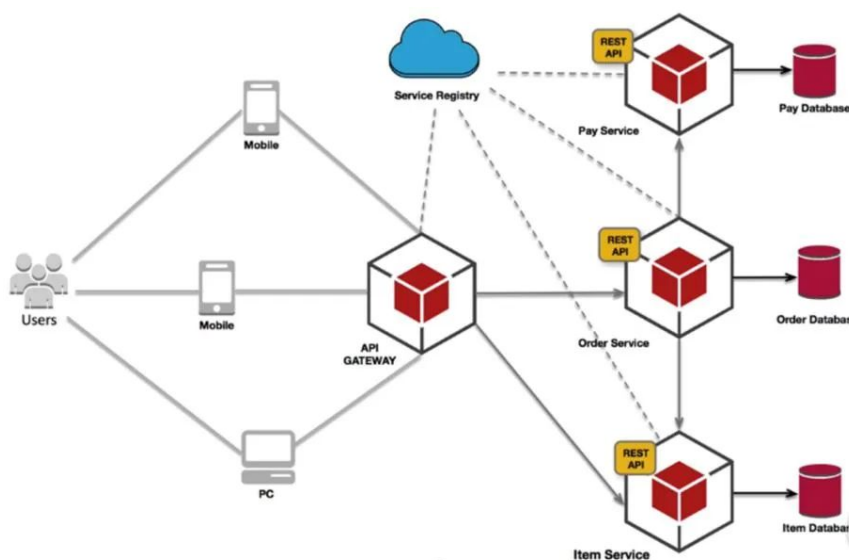
提供统一的解决方案和技术实现方式，快速集成异构系统对外服务。

2) 以注册服务为中心的服务注册发现架构

- 注册中心负责服务地址的注册与查找，相当于目录服务；
- 服务提供者和消费者只在启动和订阅后发生变化时与注册中心交互，注册中心不转发请求，压力较小；
- 应用层和服务层可以根据需求进行动态水平扩展，应用与服务实现负载均衡；
- 通过随机、轮询、权重等策略与开放式、标准化的框架，满足接口调用的服务都可以接入服务框架（RPC）监控服务调用情况，可进一步对服务层再分层，根据业务需求和服务运行情况对服务进行编排、梳理以及服务治理。

以注册服务为中心的服务注册发现架构适用大型及超大型网站应用架构。所以ESB中心化架构的问题也比较明显：中心化架构难以满足灵活性的服务迭代和需求交付。

4、微服务架构



微服务架构实现了系统解耦和持续集成，有清晰的服务边界，相对ESB架构和传统SOA架构来说粒度更小，使用轻量级的通讯机制（HTTP+REST）交互，具备更强的扩展性和弹性，能够更灵活、更快响应业务变化。

5、服务网格架构

同话题下的热门内容

该不该将单体架构迁移到微服务？

Linux 5.19 正式发布！这次用的 MacBook.....

如何创建一个无代码的自助客户门户？

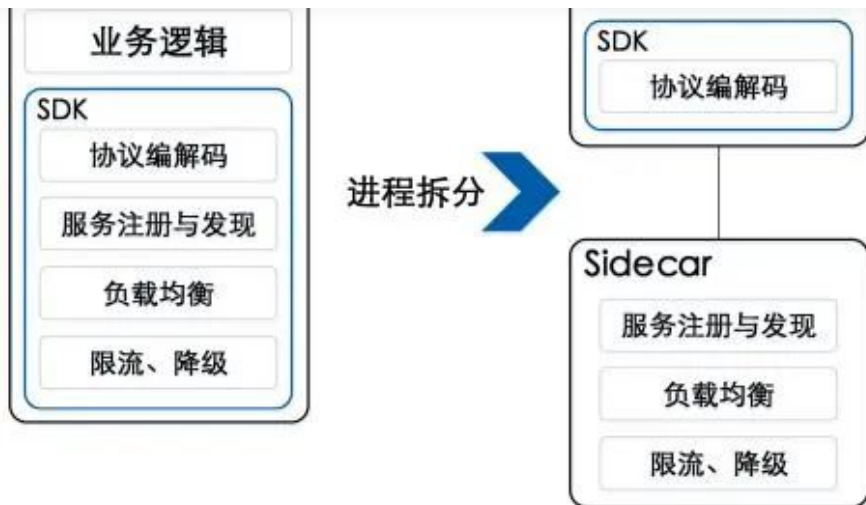
JMS VS Kafka：苹果与橘子之

「芯片法案」下周签署！台积电、三星、SK海力士谨慎观望

51CTO技术栈公众号
51CTO技术栈公众号

Java 服务 Docker 容器化优秀案例

八个流行的 Python 可视化工具



同话题下的热门内容

该不该将单体架构迁移到微服务架构

Linux 5.19 正式发布！这次用的 MacBook.....

如何创建一个无代码的自助客户门户

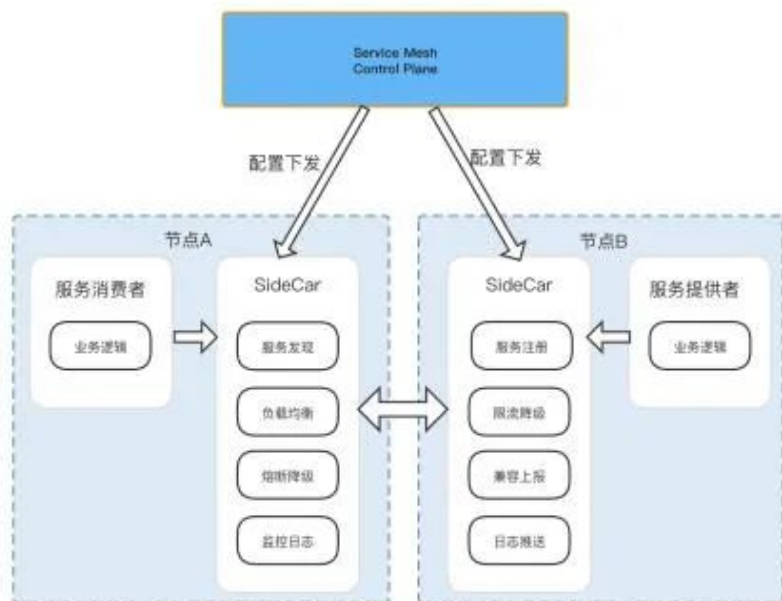
JMS VS Kafka：苹果与橘子之

「芯片法案」下周签署！台积电、三星、SK海力士谨慎观望

51CTO技术栈公众号

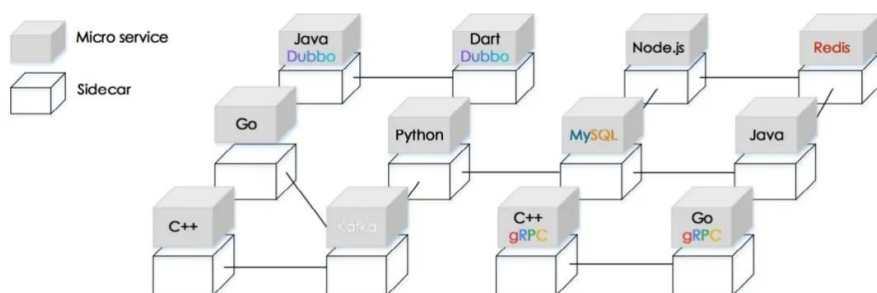
Java 服务 Docker 容器化优秀案例

八个流行的 Python 可视化工具



服务网格架构是容器化的产物，引入了类似代理的Sidecar，在微服务SDK里面保留协议编解码能力，把服务注册与发现、负载均衡、熔断、限流、降级等服务治理能力下沉到Sidecar。当该 sidecar 在微服务中大量部署时，这些 sidecar 节点自然就形成了一个网格。

服务网格架构的优势：支持用多语言开发业务、省去或轻量化SDK，为异构服务框架/平台创造了融合和发展的机会，让服务框架/平台的演进更自主、更敏捷，让业务开发聚焦业务本身，无需关心安全、灰度、熔断、限流、降级等普遍服务，治理能力更敏捷、更好管控，加速业务探索。



Service Mesh的终局：Mesh所有协议或框架。目前货拉拉已经实现了Redis Mesh。

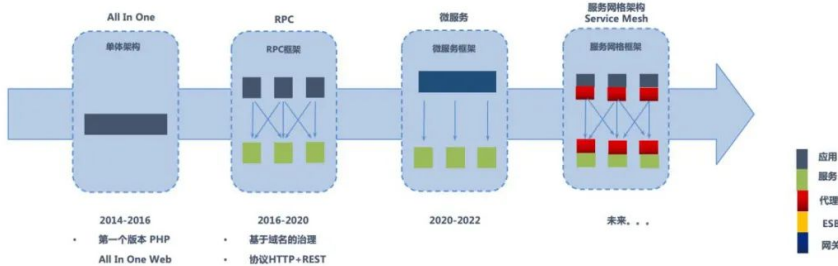
时，需要考虑这个架构是否能满足当前业务的需求，业务需求能否随着架构的演进实现增量式的迭代。

- 技术维度，要满足非功能需求，使业务快速跟上技术生态的发展。

综上所述，应用架构演进的底层逻辑就是：一切为了敏捷（低投入，快速满足业务需求）。

二、货拉拉的All In One Web到微服务治理

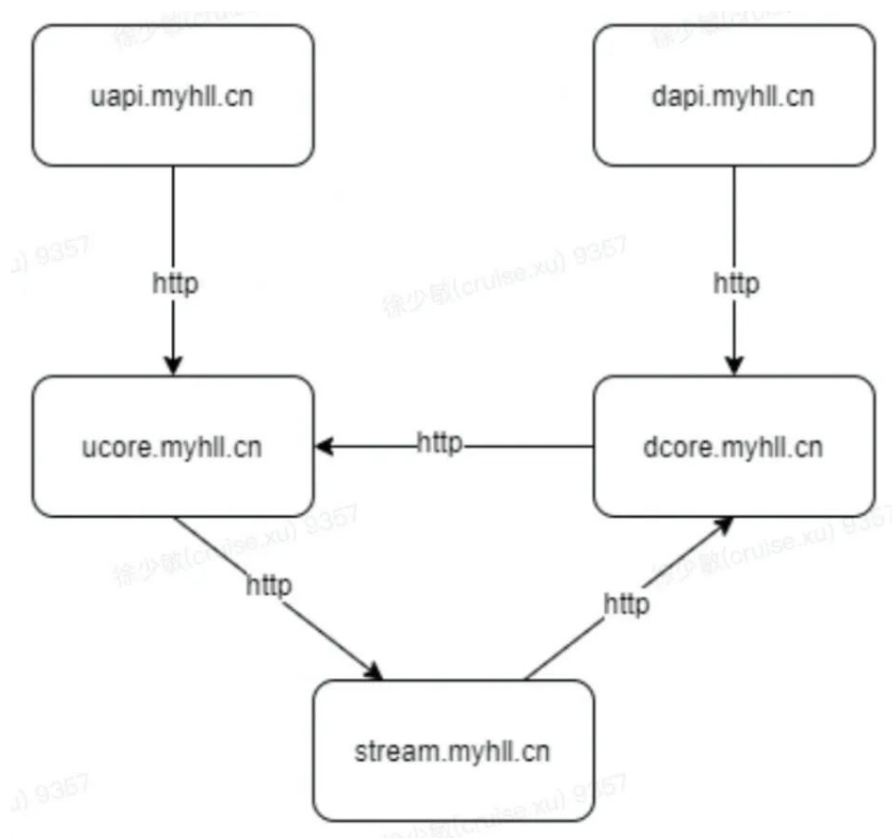
货拉拉应用架构到现在为止经历了All In One Web的单体架构，RPC架构，与现在的微服务架构。



- 单体架构阶段：2014年发布的第一个版本就是PHP的All In One Web，一直延续到2016年。
- RPC架构阶段：从2016年开始，业务量不断上升，单体架构难以满足业务需求，从单体应用里面拆出几十个dcore，ucore等核心服务。虽然服务之间采用HTTP+REST的调用方式，不是RPC的调用方式，但这阶段和RPC架构一样，都不具备服务治理能力。
- 微服务阶段：从2020年开始微服务化改造，一直到目前都还处于微服务阶段。

1、货拉拉微服务治理的背景

1) 微服务化碰到的问题



从2020年开始微服务化改造，当时阶段属于类似的RPC架构，是基于域名+HTTP的服务交互方式（图上）。痛点在于：

同话题下的热门内容

该不该将单体架构迁移到微服务

Linux 5.19 正式发布！这次用的 MacBook.....

如何创建一个无代码的自助客

JMS VS Kafka：苹果与橘子之

「芯片法案」下周签署！台积电

西 三星 SK海力士谨慎观望

51CTO技术栈公众号

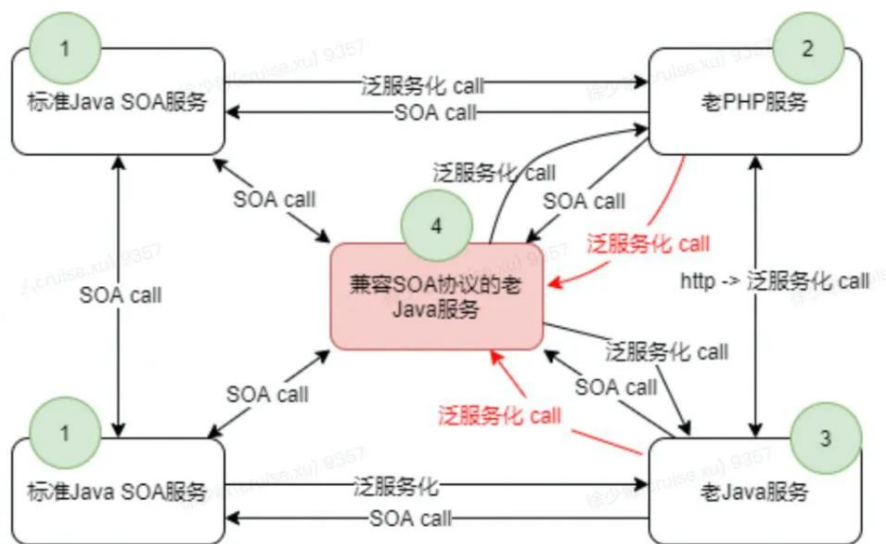
Java 服务 Docker 容器化优秀

八个流行的 Python 可视化工具

FORM表单方式。

- 没服务治理能力：没有服务注册中心，没有熔断、降级等保障能力。服务之间强关联，调用链脆弱。若某旁支服务不可用，可能导致整条关键链路不可用，稳定性无法保证。
- 技术转型：公司业务快速发展，大部分业务还是PHP开发，需要向Java转型或者使用Java来重构。内部500+的应用/服务不能同时转型或者重构，需逐步推进。

2) 微服务化需要解决的问题



- 跨语言：支持内部PHP和JAVA服务相互调用。
- 时间窗口：开发周期3个月左右。
- 服务治理能力：具备熔断、限流、降级等服务治理能力以及全链路灰度发布能力。
- 协议兼容要求：具有泛化调用能力（需要类似FeignClient的调用方式，兼容老的PHP和Java服务）。
- 协议规范要求：提供标准化RPC的能力。
- 服务注册发现：基于APPID的方式，考虑到后续容器化Service Mesh方式的服务治理要求。

2、货拉拉微服务治理的框架选型

综合上述需要解决的问题，技术选型和参考的框架：

- SOA架构：Dubbo
- 微服务架构：Spring Cloud
- 服务网格架构：Istio

公司内部基础设施还未全面容器化，所以服务网格架构方式Istio先淘汰，剩下就是Dubbo和Spring Cloud。我们选型思考的出发点：框架上的缺点或者不足点是不是我们能接受或者克服的。

1) Dubbo2.X的缺点

2020年Dubbo 3.0还未Release，所以我们研究了Dubbo 2.X。

- Dubbo 2.X协议对云原生支持不友好；
- Dubbo 2.X不支持熔断、限流、降级等服务治理能力，需要另外的Sentinel等框架支撑；
- Dubbo 3.X Release时间和版本稳定时间未知；
- 很强的RPC契约，没有泛化调用能力，达不到兼容老的Java和PHP服务要求；
- Dubbo 2.X基于接口/服务(Interface/Service)的服务注册发现方式，对未来Service Mesh方向即基于APPID的服务注册发现方式支持不友好。

同话题下的热门内容

该不该将单体架构迁移到微服务

Linux 5.19 正式发布！这次用的 MacBook.....

如何创建一个无代码的自助客

JMS VS Kafka：苹果与橘子之

「芯片法案」下周签署！台积电

西 三星 SK海力士谨慎观望

51CTO技术栈公众号

Java 服务 Docker 容器化优秀

八个流行的 Python 可视化工具

- 服务调用协议采用HTTP+REST方式，协议管控能力较弱。

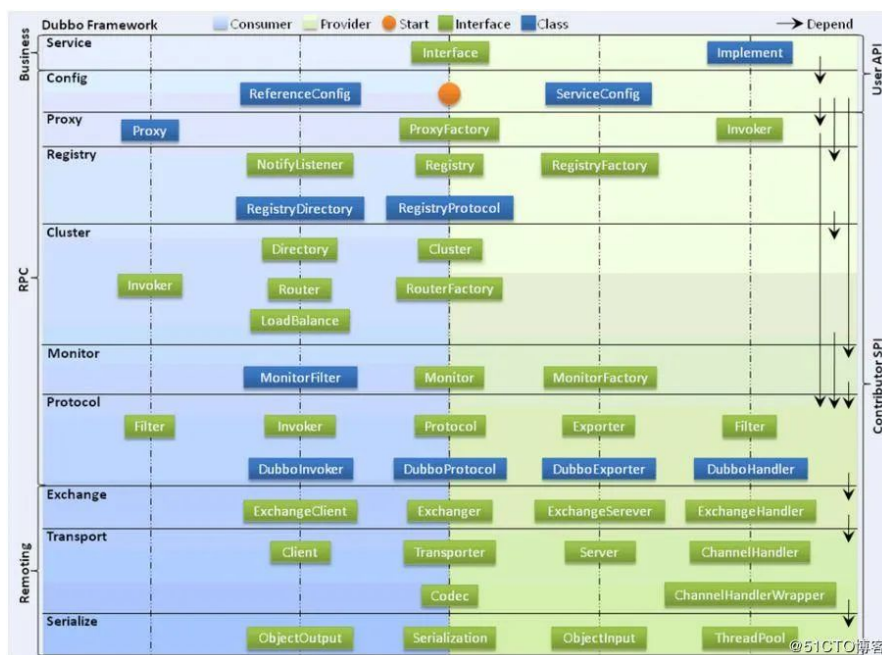
3) 自研微服务框架

我们对比后的结论是自研微服务框架，具体微服务体系中组件选型和设计思考如下：

- 标准服务调用协议：JSON-RPC（支持Java和PHP服务相互调用）；
- 注册中心：Consul；
- 服务治理：熔断Hystrix（Hystrix有PHP版本支持）；
- 泛化调用：参考FeignClient的客户端和jsonrpc4j服务定义机制；
- 框架设计：参考Dubbo的分层机制和优秀设计。

3、货拉拉微服务治理的实现思路

框架设计参考Dubbo代码的分层架构和优秀设计：



- dubbo-common
- dubbo-config
- dubbo-filter
- dubbo-metadata
- dubbo-monitor
- dubbo-registry
- dubbo-remoting
- dubbo-rpc
- dubbo-serialization
- dubbo-springboot

1) 泛化调用的实现方式

- ① 参考FeignClient定义接口方式

同话题下的热门内容

该不该将单体架构迁移到微服务

Linux 5.19 正式发布！这次用的 MacBook.....

如何创建一个无代码的自助客

JMS VS Kafka：苹果与橘子之

「芯片法案」下周签署！台积电
西 三星 SK海力士谨慎观望

51CTO技术栈公众号

Java 服务 Docker 容器化优秀

八个流行的 Python 可视化工具

```
* @author: wtg
* @createTime: 2019-01-31 10:49
*/
@FeignClient(value = "XC-SERVICE-MANAGE-CMS") //指定远程调用的服务名
public interface CmsPageClient {
    //根据页面id查询页面信息，远程调用cms请求数据
    @GetMapping("/cms/page/get/{id}") //用GetMapping标识远程调用的http的方法类型
    public CmsPage findCmsPageById(@PathVariable("id") String id);
}
```

必须指定

```
1. @FeignClient(value= "XC-SERVICE-MANAGE-CMS")
```

复制

其中XC-SERVICE-MANAGE-CMS为下游应用的APPID，FeignClient在Spring Cloud体系中以APPID的方式发现服务。

② 参考jsonrpc4j接口的定义方式

```
1. @JsonRpcService("/path/to/MyService")
2. interface MyService {
3.     ... service methods ...
4. }
```

复制

③ 泛化调用的接口定义方式

```
@SOAService(value = "/bj", appId = "1800C06-provider-svc")
public interface BodyJsonService {
    @SOAMethod(value = "queryUnPackage", paramsMode = SOAParamsMode.BODY_JSON, resultMode = SOAResultMode.OBJECT)
    BodyJsonDTO queryUnPackage(@SOAParam("id") long id, @SOAParam("age") int age); //参数转json

    @SOAMethod(value = "mapParam", paramsMode = SOAParamsMode.BODY_JSON, resultMode = SOAResultMode.OBJECT)
    BodyJsonDTO mapParam(Map map); //map转json

    @SOAMethod(value = "queryObj", paramsMode = SOAParamsMode.BODY_JSON, resultMode = SOAResultMode.OBJECT)
    BodyJsonDTO queryObj(BodyJsonDTO bodyJsonDTO); //对象转json
}
```

泛化调用方式定义

④ 标准JSONRPC接口的定义方式

```
package cn.lalaframework.lala.soa.demo.common;

import cn.lalaframework.soa.annotation.SOAService;

@SOAService(value = "/demo", appId = "jsonrpc-demo-provider")
public interface DemoService {

    String sayHello(String name);

}
```

标准RPC定义方式

2) 服务治理能力整体实现方式

- 熔断能力：集成Hystix框架。
- 治理配置：集成公司内部的配置中心。
- 监控能力：集成公司内部的监控中心。
- 治理控制台：开发治理管控平台soa-admin。

同话题下的热门内容

该不该将单体架构迁移到微服务

Linux 5.19 正式发布！这次用的 MacBook.....

如何创建一个无代码的自助客

JMS VS Kafka：苹果与橘子之

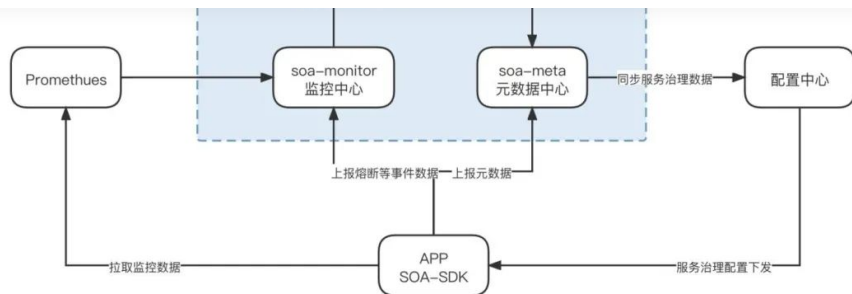
「芯片法案」下周签署！台积电

三星 SK海力士谨慎观望

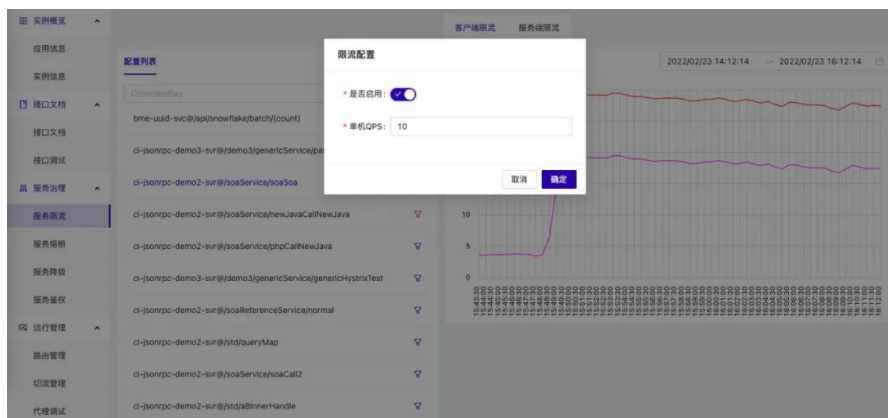
51CTO技术栈公众号

Java 服务 Docker 容器化优秀

八个流行的 Python 可视化工具



服务治理管控平台：soa-admin控制台



同话题下的热门内容

该不该将单体架构迁移到微服务

Linux 5.19 正式发布！这次用的 MacBook.....

如何创建一个无代码的自助客

JMS VS Kafka：苹果与橘子之

「芯片法案」下周签署！台积电

西 三星 SK海力士谨慎观望

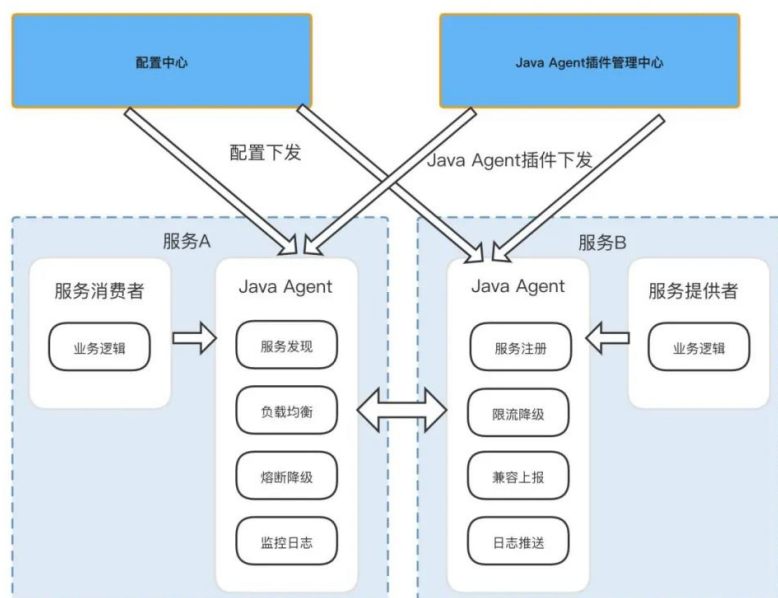
51CTO技术栈公众号

Java 服务 Docker 容器化优秀

八个流行的 Python 可视化工具

3) Java Agent版本服务治理能力实现方式

- 服务治理能力下沉到Java Agent；
- Java Agent模块化、插件化，目前插件有Metric、Trace、SOA；
- Java Agent插件支持动态升级和灰度升级。



4、货拉拉微服务治理的未来规划

未来往Service Mesh演进，但生产落地仍有挑战。

1) 增加的复杂性

在容器编排器(如Kubernetes)之上添加Istio之类的服务网格，通常需要运维人员成为这两种技术的专家。

3) 延迟

服务网格是一种入侵的、复杂的技术，它能向服务架构中添加显著的延迟。

4) 平台的依赖性

服务网格的侵入性迫使开发人员和运维人员适应一个高度自治的平台，并遵守其规则。

责任编辑：张燕妮 来源：dbaplus社群

架构 微服务 技术



分享到微信

分享到微博

相关推荐



云服务商解决方案业务演进方向

云服务商解决方案及其演进方向随着电信产业步入超宽带网络 and 全联接时代，传统的语音、消息体验和人与人的通信已经不能满足全面的个性化的用户需求。

2022-08-08 12:47:26

云服务商 解决方案 电信产业



宜信微服务架构落地及其演进|分享实录

从单体到SOA架构，从分布式服务化架构再到微服务架构，企业应用架构领域每一次技术架构的演进都会给企业带来更多的价值：职责解耦、能力复用、关注点分离、沟通效率提升、快速演进、快

2019-12-31 10:33:48

架构 运维 技术



七个阶段模型，帮助微服务架构落地！

业务的飞速发展以及变化多端的动态组合一直推着以 IaaS、PaaS 和 SaaS 形式表现的云计算不断发展，随之微服务的实施方案也需要与时俱进。

2022-01-04 07:00:21

模型 微服务 架构



系统由单体架构到微服务架构到底是如何演进的？

随着互联网的发展，互联网企业的业务也在不断的飞速发展，进而导致系统的架构也在不断的发生变化。

2021-06-07 10:13:01

从单体式架构迁移到微服务架构：三个策略叙述

同话题下的热门内容

该不该将单体架构迁移到微服务架构？

Linux 5.19 正式发布！这次用的 MacBook.....

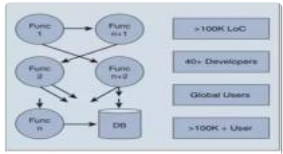
如何创建一个无代码的自助客户门户？

JMS VS Kafka：苹果与橘子之

「芯片法案」下周签署！台积电、三星、SK海力士谨慎观望
51CTO技术栈公众号

Java 服务 Docker 容器化优秀案例

八个流行的 Python 可视化工具



如何从复杂单体应用快速迁移到微服务？

想必你已知道了微服务及其工作原理，现在是时候探讨如向微服务转变这个关键话题了。

2019-01-07 08:10:54

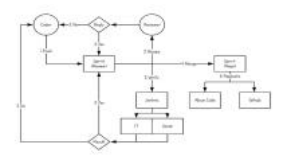
微服务 单体 Web



微服务落地，我们在考虑什么？

微服务已经成为过去几年软件架构设计的“事实标准”，大多数企业在推动内部数字化转型的过程中，服务软件系统开始由单一或者SOA服务向微服务转型。那么转型过程需要遵循哪些原则呢？本

2019-04-04 12:59:03



一个可供小团队参考的微服务落地实践

微服务是否适合小团队是个见仁见智的问题。但小团队并不代表出品的一定是小产品，当业务变得越来越复杂，如何使用微服务分而治之就成为一个不得不面对的问题。

2019-04-04 09:11:41

微服务 CDP Linkflow



大火的“微服务架构”详解与实践

架构肯定是为业务需求而生的，先来看看我们面对的业务需求及其特点。平台最主要满足两大类业务需求：面向餐饮企业在餐饮新零售下的经营和运营需求和面向产品及运营团队。

2019-12-26 15:49:14

微服务 架构 业务



技术干货分享：微服务浅谈服务治理的演变过程

本篇文章先简单介绍了互联网架构的演变，进而介绍了服务化，最后介绍了微服务及最新的服务网格(Service Mesh)。

2019-08-16 08:59:33



在阿里“救了八年火”的程序猿讲述大型项目架构演进过程

一个大型服务系统都是从小一步一步走过来的，在每个阶段，找到对应该阶段网站架构所面临的问题，然后在不断解决这些问题，在这个过程中整个架构会一直演进。

2018-01-09 22:18:18



Uber 刘彦东：当Uber开拓送餐服务后，全世界的外卖都震惊了

Uber不仅能叫车，原来还能点餐。

2017-07-26 18:49:00

京东 机器学习 人工智能



CIO半月刊第十六期|Uber 刘彦东：当Uber开拓送餐服务后，全世界的外卖都震惊了

Uber不仅能叫车，原来还能点餐。

2017-08-08 15:02:00

半月刊

同话题下的热门内容

该不该将单体架构迁移到微服务？

Linux 5.19 正式发布！这次用的 MacBook.....

如何创建一个无代码的自助客户门户

JMS VS Kafka：苹果与橘子之

「芯片法案」下周签署！台积电、三星、SK海力士谨慎观望

51CTO技术栈公众号

Java 服务 Docker 容器化优秀案例

八个流行的 Python 可视化工具

51CTO业务

- 媒体

社区

教育
- 51CTO

51CTO博客

51CTO学堂
- CIOAge

开源基础软件社区

精培
- HC3i

企业培训
- CTO训练营



关于我们 & 版权

- 关于我们

北京市海淀区中关村南1条甲1号ECO中科爱克大厦6-7层
- 站点地图

北京市公安局海淀分局备案编号：110108002980号
营业执照 京ICP备09067568号
- 网站大事

Copyright © 2005-2022 51CTO.COM 京ICP证060544 版权所有
有 未经许可 请勿转载
- 意见反馈
- English
- 用户协议
- 隐私协议

友情链接

同话题下的热门内容

新辰科技 微讯科技

该不该将单体架构迁移到微服务

Linux 5.19 正式发布，这次用的 MacBook.....

速途网 中国经济新闻网

如何创建一个无代码的自助客户

工联网 极客公园

JMS VS Kafka：苹果与橘子之

中国IDC圈 企业级BNet

「芯片法案」下周签署！台积电

西二星 SK海力士谨慎观望

51CTO技术栈公众号

51CTO技术栈公众号

Java 服务 Docker 容器化优秀

八个流行的 Python 可视化工具

