

# 多点生活在 Service Mesh 上的实践

## Istio + MOSN 在 Dubbo 场景下的探索之路

陈鹏

多点生活 平台架构组研发工程师



- 陈鹏、多点生活平台架构组研发工程师
- 开源项目与云原生爱好者
- 多年网上商城、支付系统相关开发经验
- 2019 年至今从事云原生和 Service Mesh 相关开发工作



## /01 为什么需要 Service Mesh 改造

- 对比传统微服务架构
- 和 Service Mesh 化之后有哪些优缺点

## /02 探索 Istio 技术点

- MCP
- Pilot
- xDS
- MOSN

## /03 Dubbo 场景下的改造

结合 Istio 的技术点，介绍多点生活目前的探索以及服务发现 Demo 的演示



# /01

# 为什么需要 Service Mesh 改造

对比传统微服务架构和 Service Mesh 化之后有哪些优缺点



## 模块

- 安全
- 配置中心
- 调用链监控
- 网关
- 监控告警
- 注册和发现
- 容错和限流

## 特点

- 独立部署
- 强化模块边界
- 技术多样性



# Service Mesh

## 优点

- 统一的服务治理
- 服务治理和业务逻辑解藕

## 缺点

- 增加运维复杂度
- 延时
- 需要更多技术栈



/02

# 探索 Istio 技术点

了解 Istio 技术点



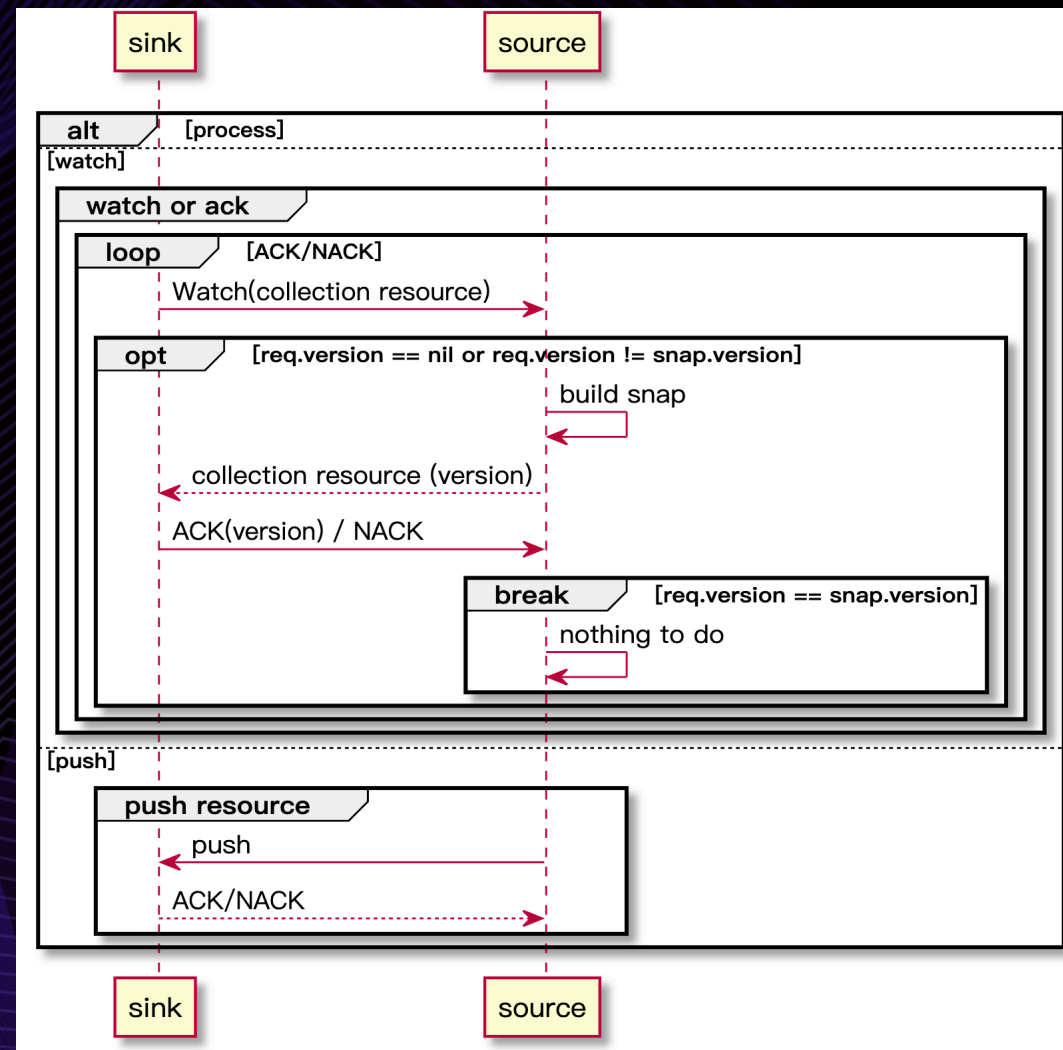
# MCP

MCP ( Mesh Configuration Protocol ) 提供了一套用于订阅、推送的 API。  
分为 Source 和 Sink :

- Source 是资源提供方 ( server ) , 资源变化了推送给订阅者 ( Pilot ) , Istio 1.5 之前这个角色就是 Galley 或者自定义 MCP Server ;
- Sink 资源的订阅者 ( client ) , 在 Istio1.5 之前这个角色就是 Pilot 和 Mixer , 都是订阅 Galley 或者自定义 MCP Server 的资源 ;



# MCP





# Pilot

Pilot 负责网格中的流量管理以及控制面和数据面之间的配置下发，在 Istio1.5 之后合并了 Galley、Citadel、Sidecar-Inject 和 Pilot 成为 Istiod。

## 功能

- 根据不同的平台（Kubernetes、Console）获取一些资源（Kubernetes 中使用 Informer 机制获取 Node、Endpoint、Service、Pod 变化）
- 根据用户的配置（CR、MCP 推送，文件）触发推送流程

## 推送流程

- 记录变化的资源类型
- 根据变化的资源类型整理本地数据
- 根据变化的资源类型判断需要下发的 xDS 资源
- 构建 xDS 资源，下发到连接的 Sidecar



Sidecar 通过动态获取服务信息，对服务的发现 API 被称为 xDS。

- 协议部分（ADS、控制发送的顺序以及返回的确认数据）
- 数据部分（CDS、EDS、LDS、RDS、SDS）



# 资源变化

资源名称	CDS	EDS	LDS	RDS
Virtualservices			✓	
Gateways				
Serviceentries				
Destinationrules				
Envoyfilters				
Sidecars				
ConfigClientQuotaspecs				
ConfigClientQuotaspecbindings				
Authorizationpolicies				
Requestauthentications				
Peerauthentications				
Other				



# MOSN

MOSN 是一款使用 Go 语言开发的网络代理软件，作为云原生的网络数据平面，旨在为服务提供多协议、模块化、智能化、安全的代理能力。MOSN 是 Modular Open Smart Network 的简称。MOSN 可以与任何支持 xDS API 的 Service Mesh 集成，亦可以作为独立的四、七层负载均衡，API Gateway，云原生 Ingress 等使用。

## 配置文件

- mosn\_config : MOSN 的配置信息
- listener : LDS
- routers : RDS
- cluster : CDS 和 EDS



# MOSN-listener

```
"0.0.0.0_9080":{  
  "name":"0.0.0.0_9080",  
  "address":"0.0.0.0:9080",  
  "filter_chains":[  
    {  
      "match":"prefix_ranges:<address_prefix:\"10.13.160.94\" prefix_len:<value:32 > > ",  
      "filters":[  
        {  
          "type":"proxy",  
          "config":{  
            "downstream_protocol":"Auto",  
            "router_config_name":"9080",  
            "upstream_protocol":"Auto"  
          }  
        }  
      ]  
    }  
  ],  
  "inspector":true  
},
```



# MOSN-routers

```
"9080":{
  "router_config_name":"9080",
  "virtual_hosts":[
    @Object{...},
    @Object{...},
    @{
      "name":"ratings.default.svc.cluster.local:9080",
      "domains":[
        "ratings.default.svc.cluster.local",
        "ratings.default.svc.cluster.local:9080",
        "ratings",
        "ratings:9080",
        "ratings.default.svc.cluster",
        "ratings.default.svc.cluster:9080",
        "ratings.default.svc",
        "ratings.default.svc:9080",
        "ratings.default",
        "ratings.default:9080",
        "10.13.163.195",
        "10.13.163.195:9080"
      ],
      "routers":[
        @{
          "match":@{
            "prefix":"/"
          },
          "route":@{
            "cluster_name":"outbound|9080|ratings.default.svc.cluster.local",
            "timeout":"0s",
            "retry_policy":@{
              "retry_on":true,
              "retry_timeout":"0s",
              "num_retries":2
            }
          }
        }
      ]
    }
  ],
  @Object{...}
}
```



# MOSN-cluster

```
"outbound|9080||ratings.default.svc.cluster.local":⊖{
  "name":"outbound|9080||ratings.default.svc.cluster.local",
  "type":"EDS",
  "lb_type":"LB_ROUNDROBIN",
  "circuit_breakers":⊕Array[1],
  "health_check":⊕Object{...},
  "spec":⊕Object{...},
  "lb_subset_config":⊕Object{...},
  "original_dst_lb_config":⊕Object{...},
  "tls_context":⊕Object{...},
  "hosts":⊖[
    ⊖{
      "address":"10.13.160.85:9080",
      "metadata":⊕Object{...}
    }
  ]
},
```



# /03

# Dubbo 场景下的改造

从数据面、控制面两个方面来介绍如何改造



# 改造方案1

## Istio+Envoy

- 通过创建 EnvoyFilter 资源来给 xDS 资源打 patch
- Envoy 解析 Dubbo 协议中的 Service 和 Method
- 根据路由策略配置把流量转发到对应的 Provider
- 通过WASM扩展

华为云 : [https://support.huaweicloud.com/bestpractice-istio/istio\\_bestpractice\\_3005.html](https://support.huaweicloud.com/bestpractice-istio/istio_bestpractice_3005.html)



# 改造方案2

## MOSN+Dubbo-go

- MOSN 提供 Subscribe、Unsubscribe、Publish、Unpublish 服务
- SDK 发送请求 MOSN 提供的服务
- MOSN 通过 Dubbo-go 直接和注册中心连接



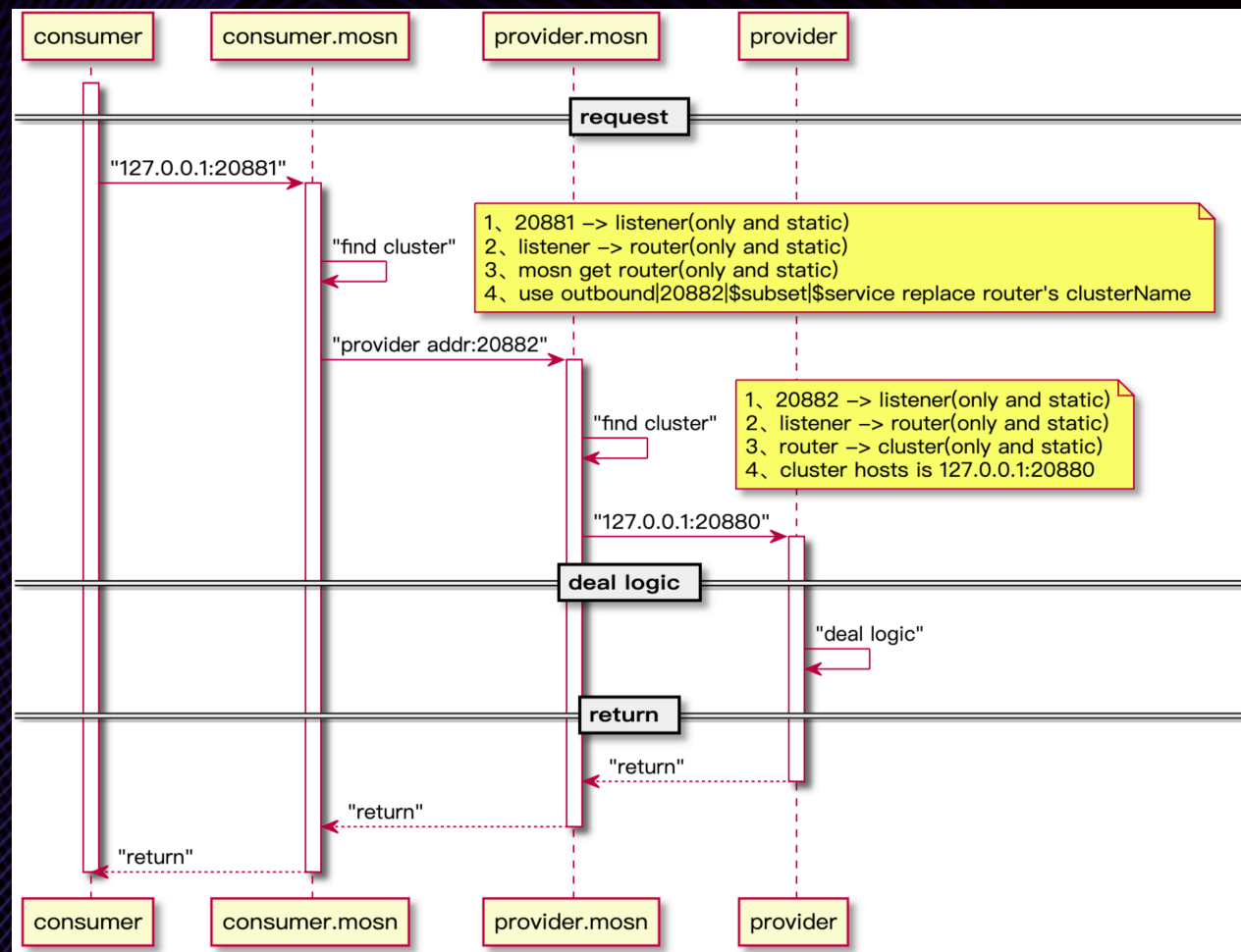
# 改造方案3

## Istio+MOSN

- 数据面改造
- 控制面适配



# 数据面改造





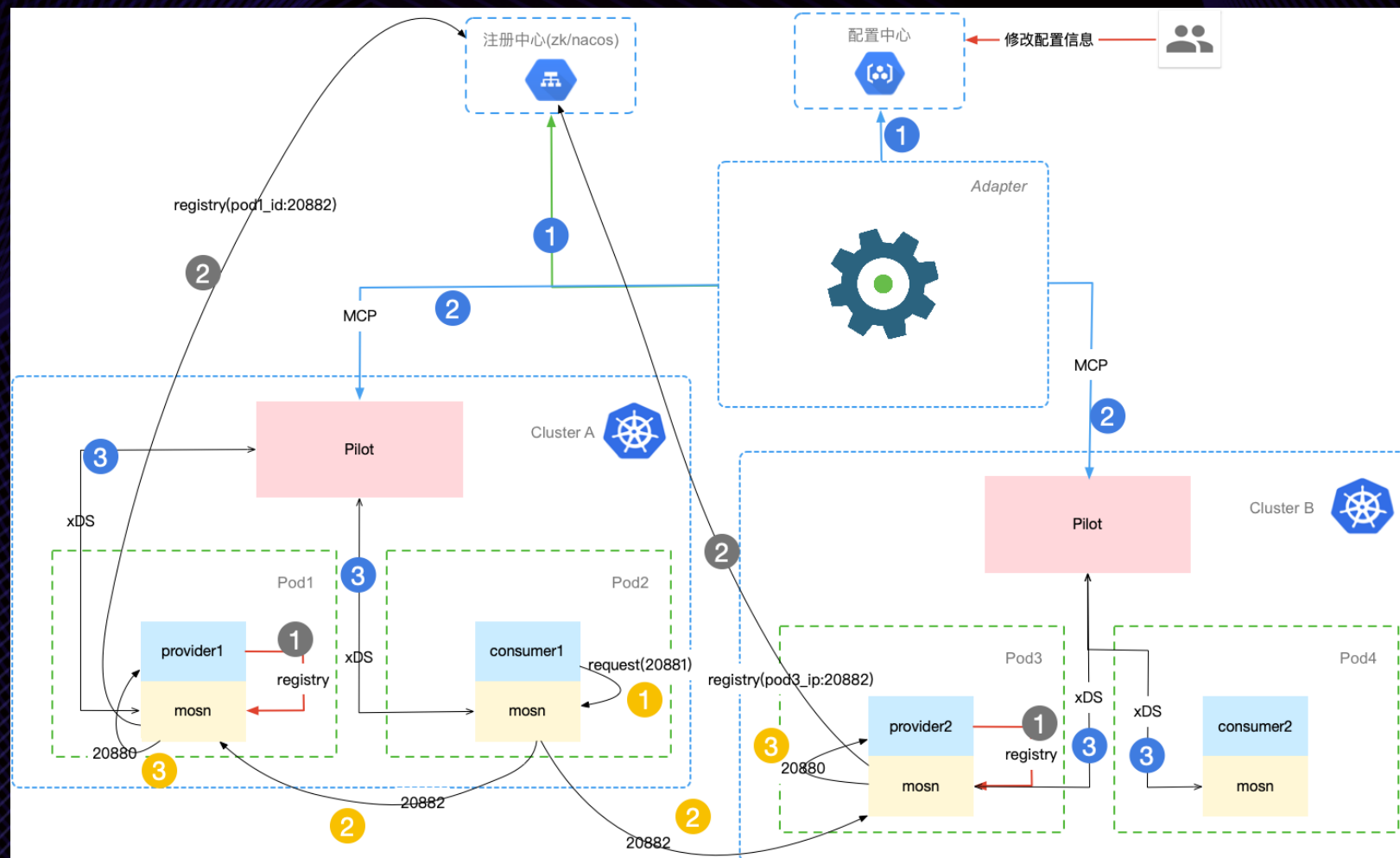
# 控制面适配

通过一个独立的进程 ( Adapter )

- 对接注册中心
- 对接配置中心
- 连接多个集群
- 根据服务的信息结合配置信息下发对应的集群的资源



# 多点Service Mesh改造架构图





# Demo 演示

## 说明

- 由于没有真正的注册，所以使用手动添加 ServiceEntry 的方式代替 Adapter 功能
- Listener 和 Routers 配置信息目前是固定的
- Provider 只注册到本地 zk
- Sidecar 注入到方式使用的是多个 Container



感谢聆听



欢迎关注，获取最新分布式架构内容



关注服务网格，关注 ServiceMesher