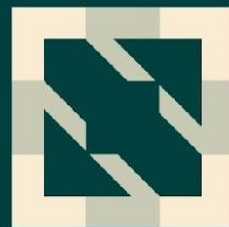




**KubeCon**



**CloudNativeCon**



**OPEN SOURCE SUMMIT**

**China 2023**



KubeCon



CloudNativeCon



OPEN SOURCE SUMMIT

China 2023

# Struggling to Manage Non-HTTP Traffic with Istio? Aeraki Mesh to Rescue!

*Huabing Zhao Tetrade.io*  
*Shilin Qin BOSS Zhipin*

- 2020 年 11 月：开源 Aeraki（控制面）
  - 支持 Envoy native Dubbo, Thrift Filter
- 2021 年 08 月：开源 MetaProtocolProxy（数据面）
  - 基于 MetaProtocolProxy 实现 Dubbo 和 Thrift
  - 基于 MetaProtocolProxy 支持私有协议
  - 实现七层路由，本地/全局限流，流量镜像等流量管理能力
- 2022 年 06 月：加入 CNCF Sandbox
- 2023 年 02 月：发布 1.2.2 版本
  - 实现服务级别的 Tracing, Logging, Metrics 可观测性能力
- 2023 年 06 月：发布 1.3.0 版本
  - Ingress Gateway 支持 MetaProtocol
  - 支持 Istio 1.16.x 系列版本
- 2023 年 08 月：发布 1.4.0 版本
  - 支持 Istio 1.18.x 系列版本





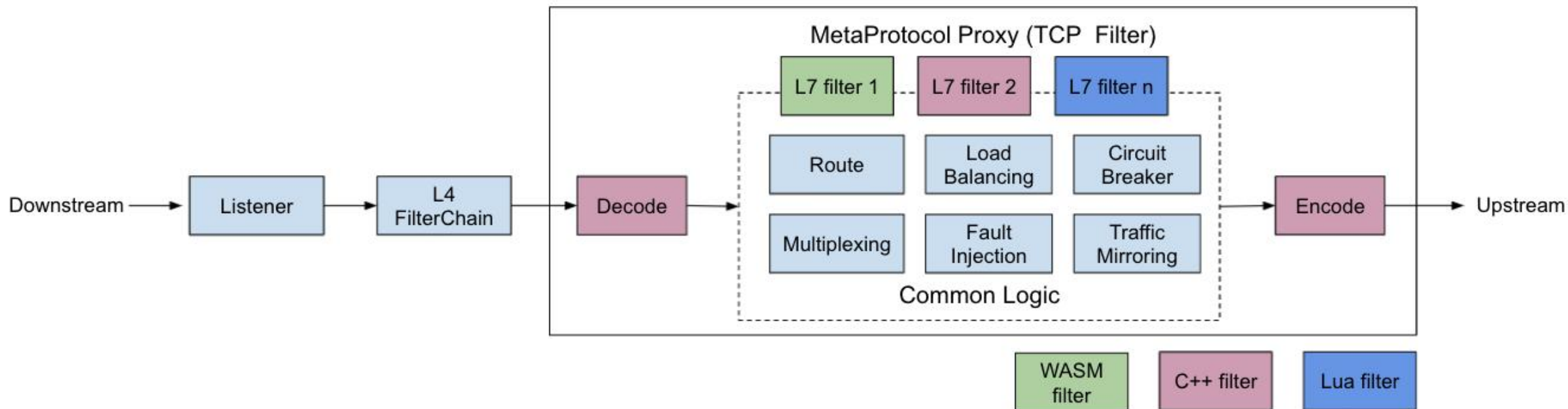
# 数据面 - MetaProtocol Proxy (Envoy)

大部分七层协议的路由、熔断、负载均衡等能力的实现逻辑是类似的，没有必要每个协议都全部从头实现，重复造轮子。

Protocol	Destination service	Parameters could be used for routing
HTTP 1.1	host	host, path, method headers
HTTP 2	pseudo header: authority	pseudo header: authority, path, method, headers
gRPC	HTTP 2 path	Request-Headers(Delivered as HTTP2 headers)
TARS	ServantName	ServantName, FuncName, Context
Dubbo	service name	service name, service version, service method
Any RPC Protocol	service name in message header	some key:value pairs in message header

# 数据面 - MetaProtocol Proxy (Envoy)

- MetaProtocol Proxy 中实现七层协议的通用逻辑：路由、Header Mutation、负载均衡、断路器、多路复用、流量镜像、Tracing、Logging、Metrics 等。
- 基于 MetaProtocol 实现一个自定义协议时，只需要实现 Decode 和 Encode 扩展点的少量代码 (C++)。
- 提供 L7 filter，用户可以实现一些灵活的自定义协议处理逻辑，例如认证授权等。



# 数据面 - MetaProtocol Proxy (Envoy)

两个重要的数据结构:

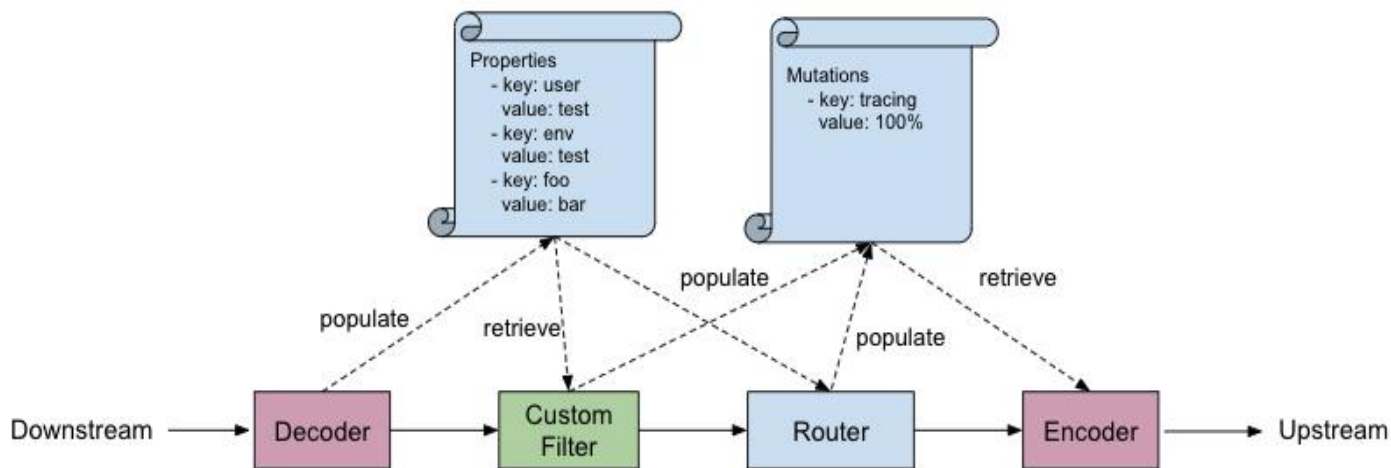
## Metadata

- decoder 从请求中提取出的元数据
- 可以是任意 key-value 键值对
- 可以被 L7 filter 使用: routing match, rate limiting match

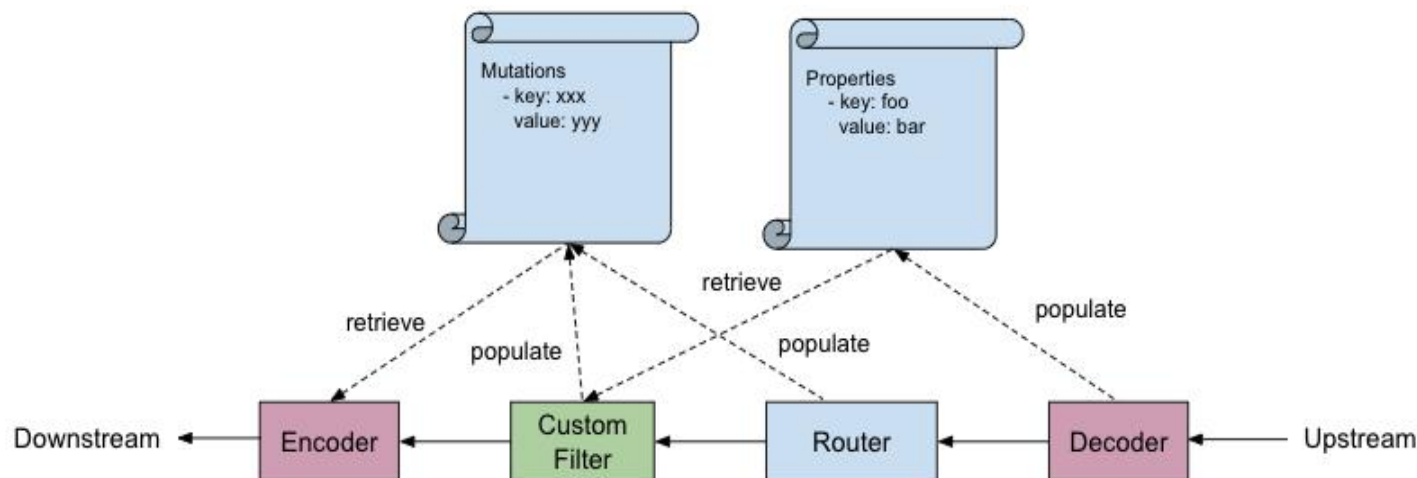
## Mutation

- L7 filter 填充 Mutation
- 可以是任意 key-value 键值对
- 被 encoder 用于修改请求后向 Upstream 发送

## Request Path



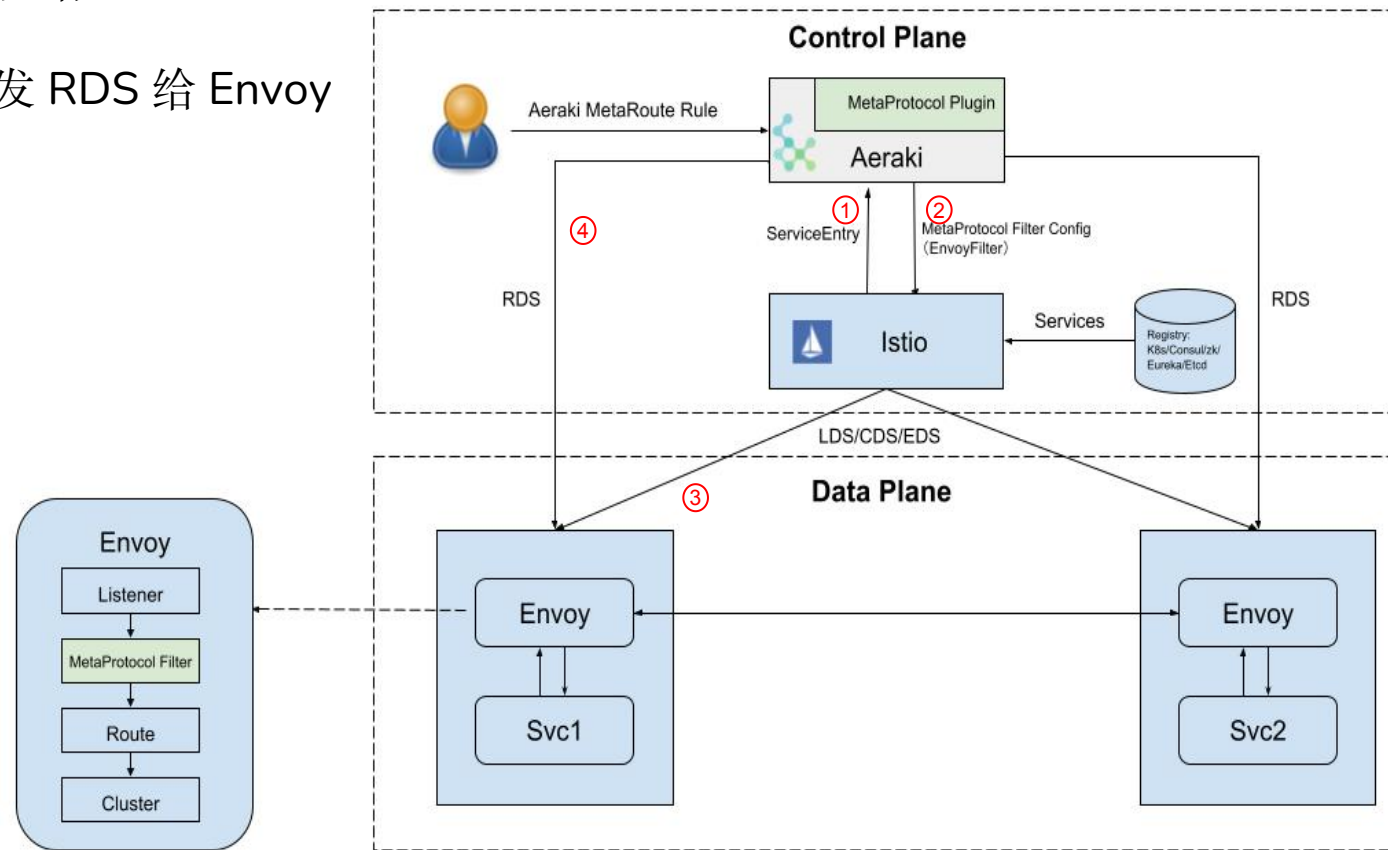
## Response Path



# 控制面 - Aeraki

通过 Aeraki 实现控制面的流量管理规则下发:

1. Aeraki 从 Istio 中获取 ServiceEntry, 通过端口命名判断协议类型 (如 tcp-metaprotocol-thrift)
2. 为 MetaProtocol 服务生成数据面所需的 Filter 配置, Filter 配置中将 RDS 指向 Aeraki, 然后通过 EnvoyFilter 下发给 Istio
3. Istio 下发 LDS(Patch)/CDS/EDS 给 Envoy
4. Aeraki 根据缺省路由或者用户设置的路由规则下发 RDS 给 Envoy



社区缺省支持 Istio 双数版本

Aeraki	MetaProtocol Proxy	Istio
1.4.x	1.4.x	1.18.x
1.3.x	1.3.x	1.16.x
1.2.x	1.2.x	1.14.x
1.1.x	1.1.x	1.12.x
1.0.x	1.0.x	1.10.x



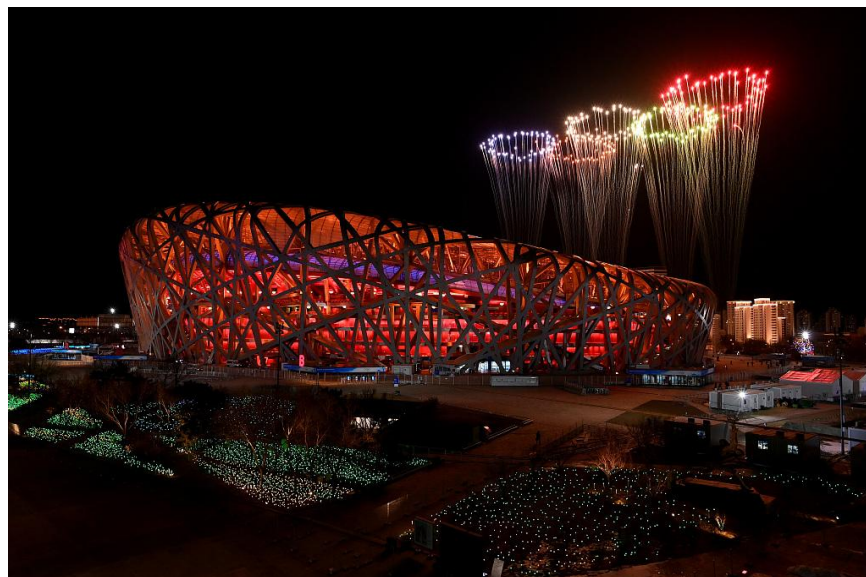
# 项目应用

## 协议支持：数十种开源及私有协议

- Dubbo
- Thrift
- bRPC (百度)
- tRPC (腾讯)
- 华为内部私有协议
- 腾讯音乐私有协议
- 腾讯融媒体私有协议
- 腾讯光子工作室私有协议
- 腾讯欢乐工作室私有协议
- 国泰君安私有协议
- .....

## 项目案例：

- 2022 年冬奥会直播软件 - 私有协议
- 腾讯音乐 - 私有协议
- Boss 直聘 - Dubbo, Thrift
- 灵雀云 - Dubbo
- 更多案例参见：<https://github.com/aeraki-mesh/aeraki/issues/105>





KubeCon



CloudNativeCon



OPEN SOURCE SUMMIT

China 2023

# Aeraki Mesh管理Dubbo协议在Boss直聘的落地实践

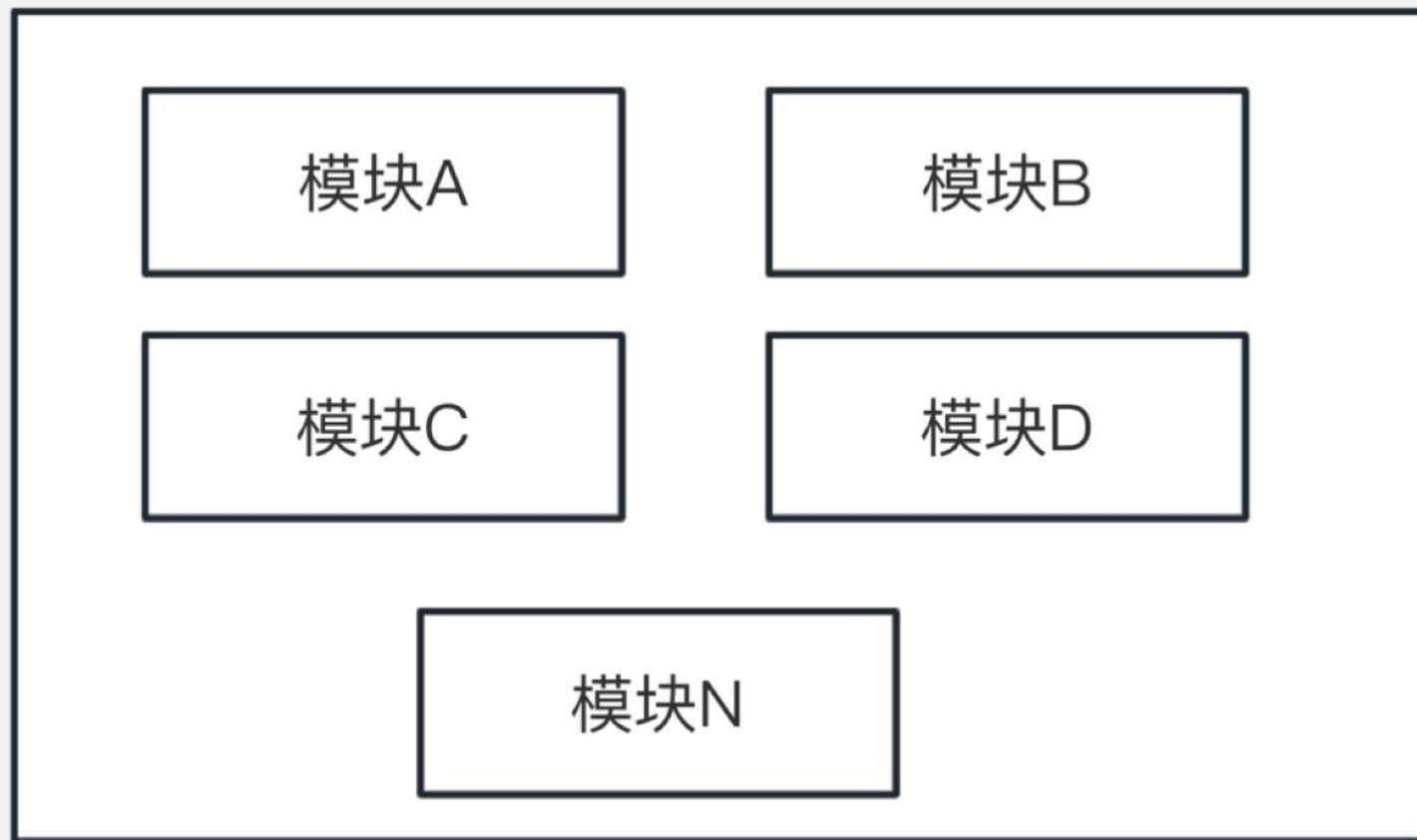
*Shilin Qin BOSS Zhipin*

*Email: qinshilin110@gmail.com*

# 业务架构的演进

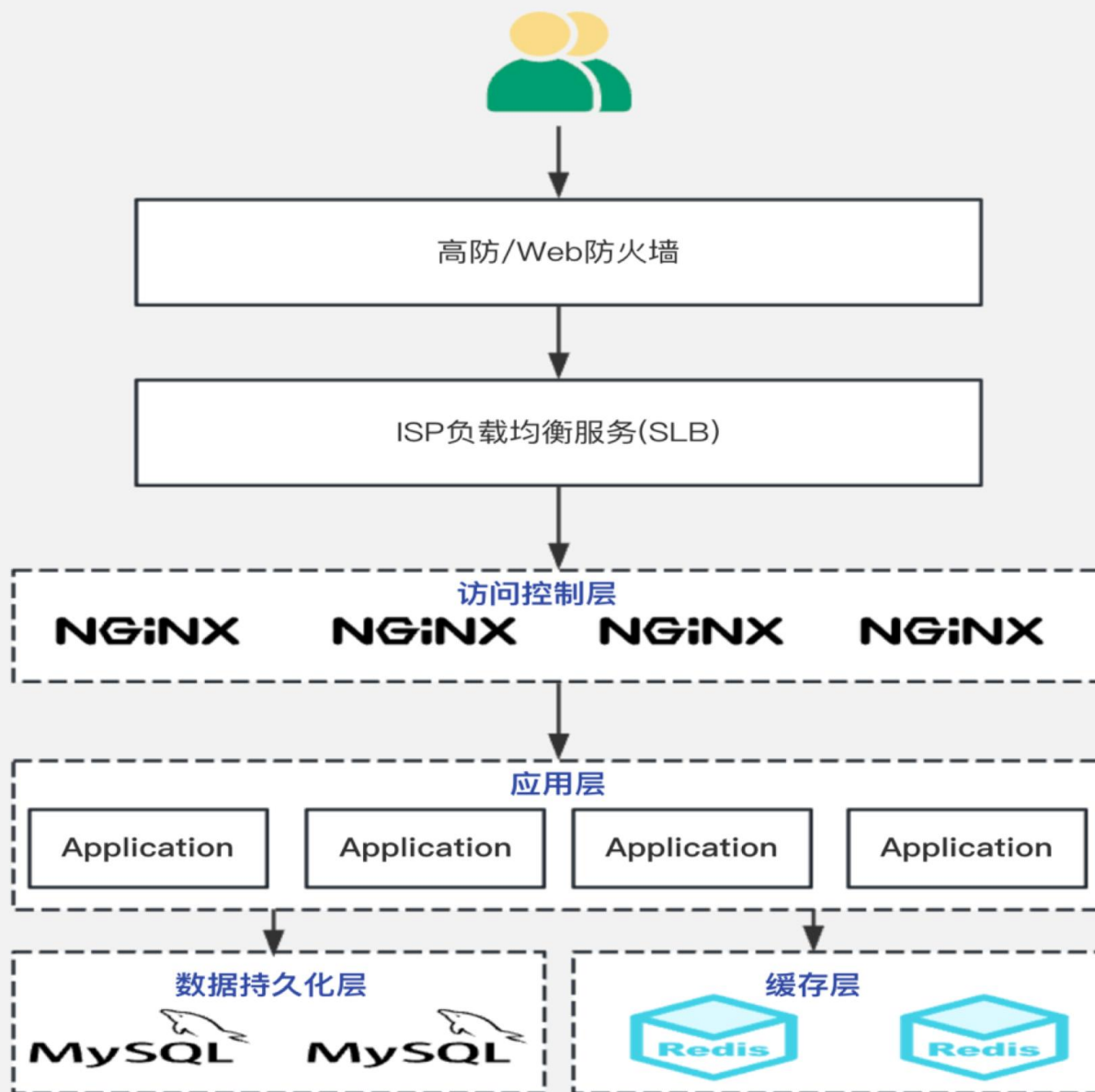
1. 单体架构
2. 集群架构
3. 面向服务架构
4. 微服务架构

# 单体架构

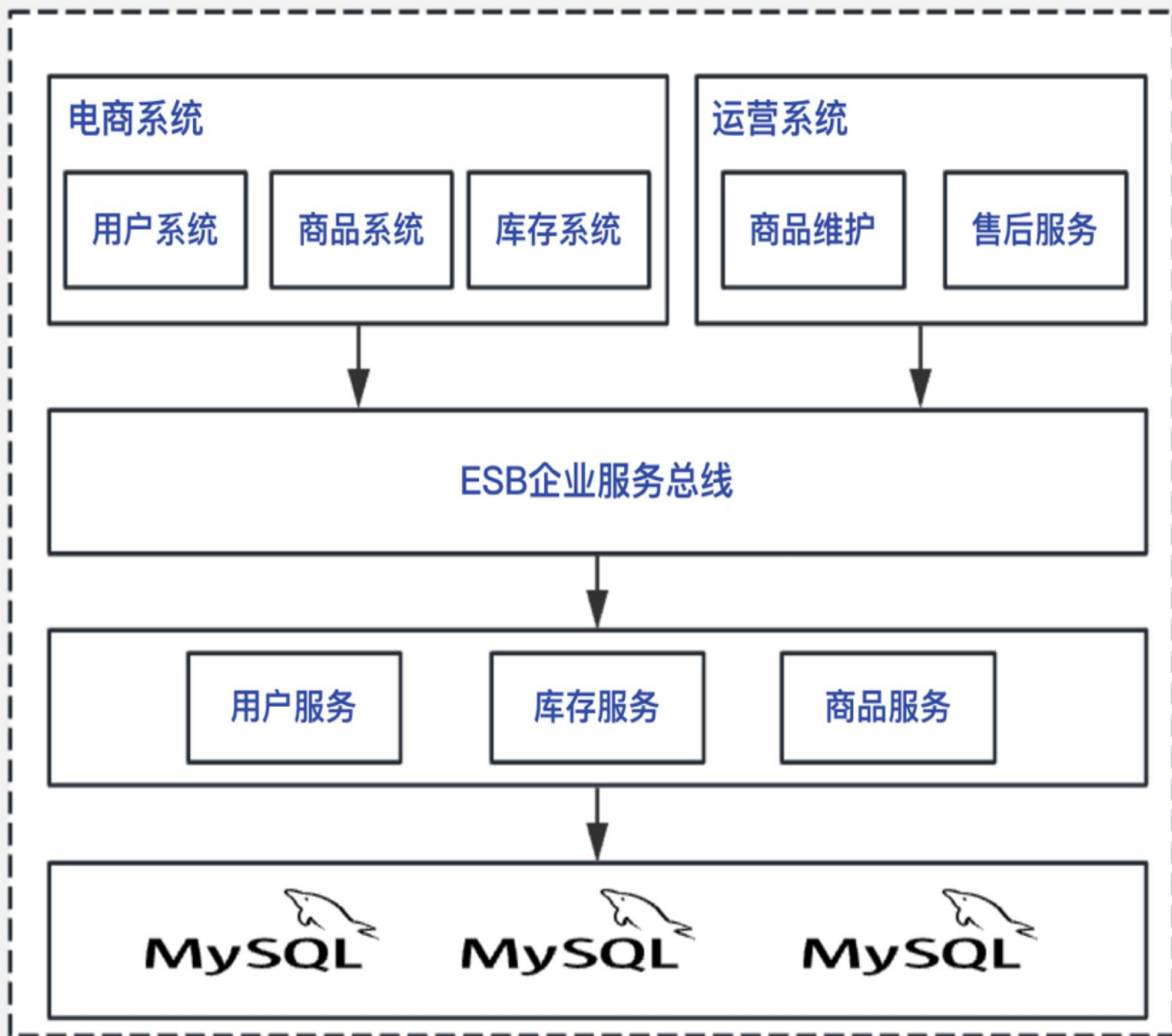




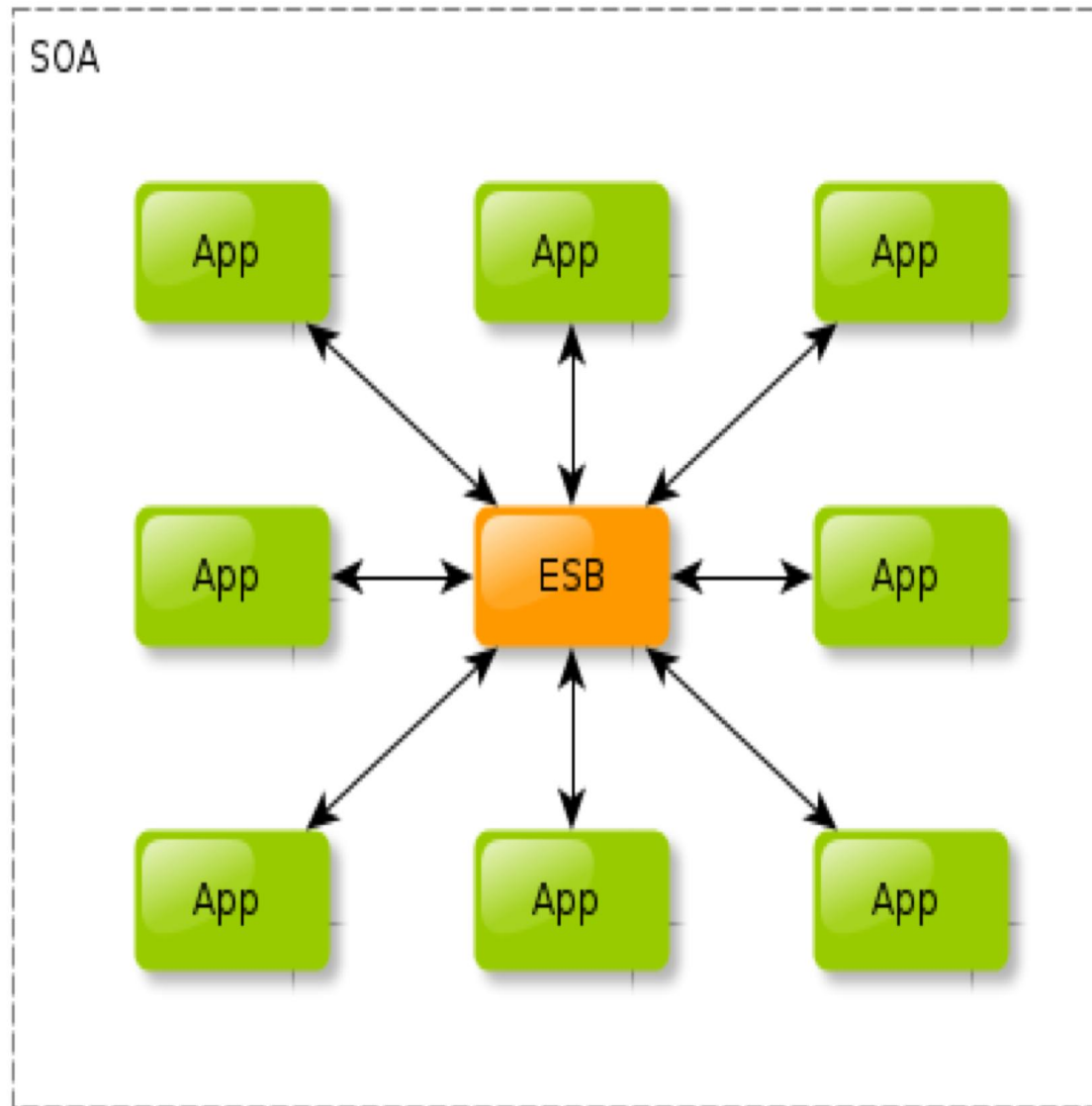
# 集群架构



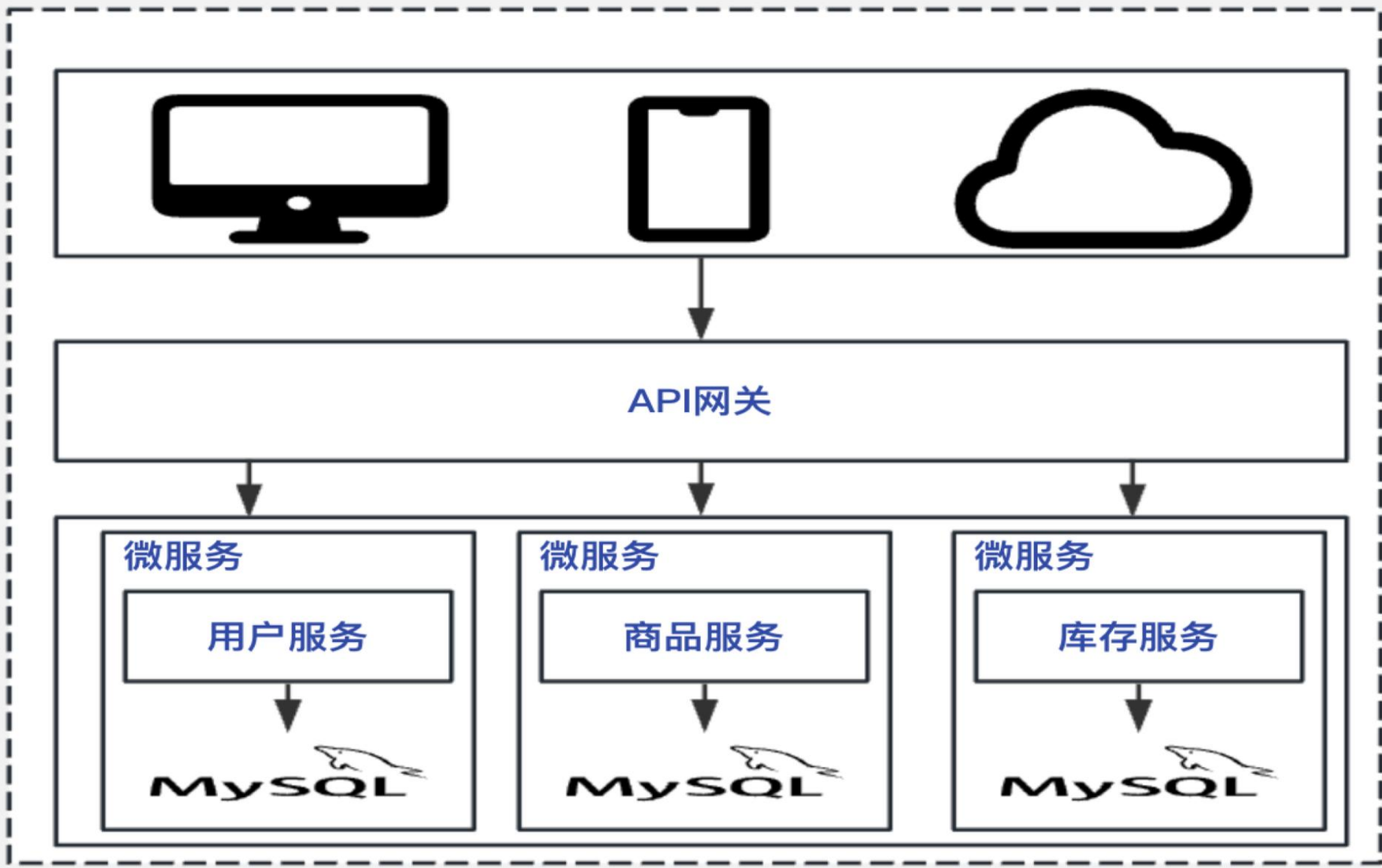
# 面向服务架构



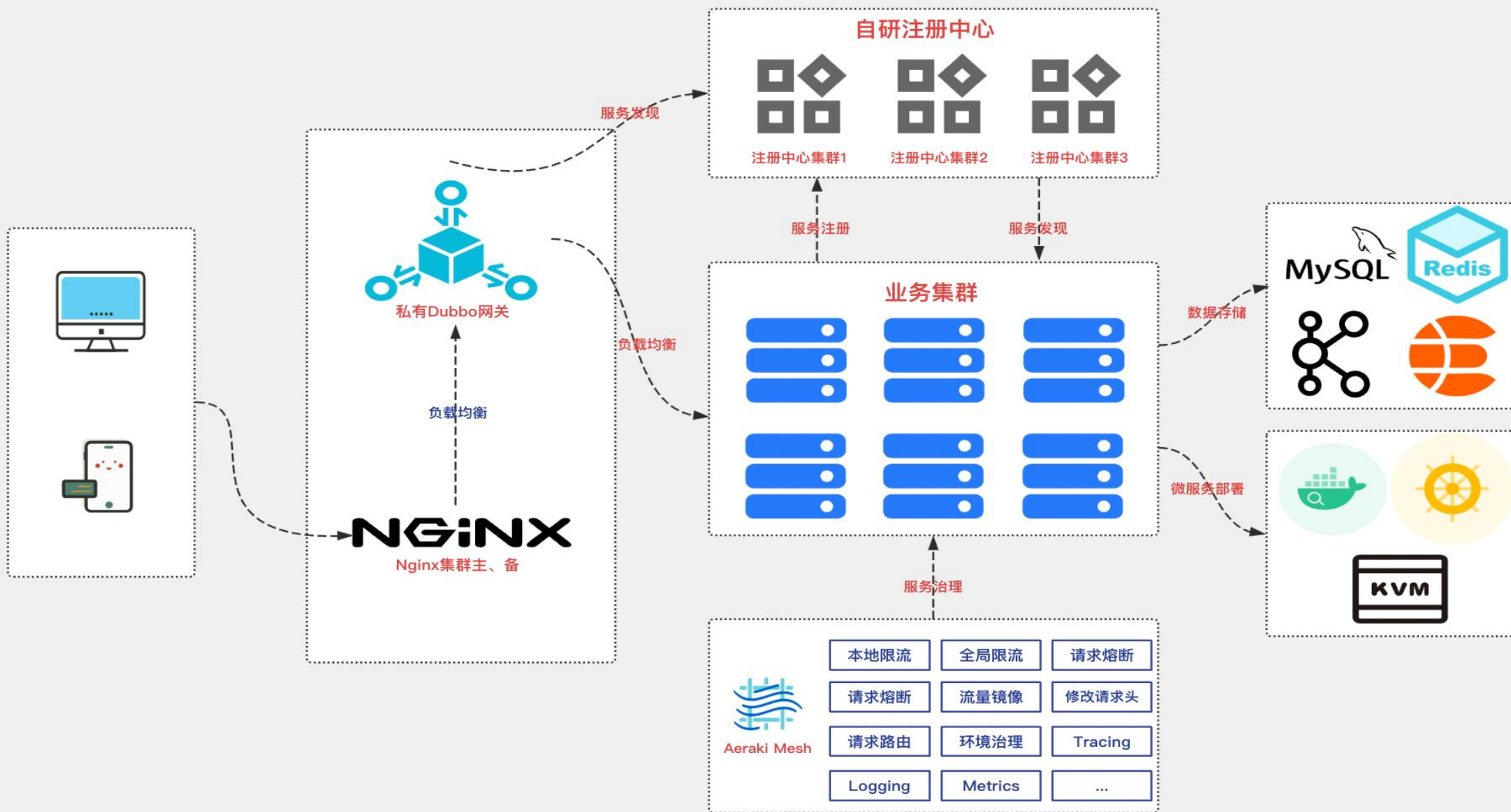
SOA



# 微服务架构

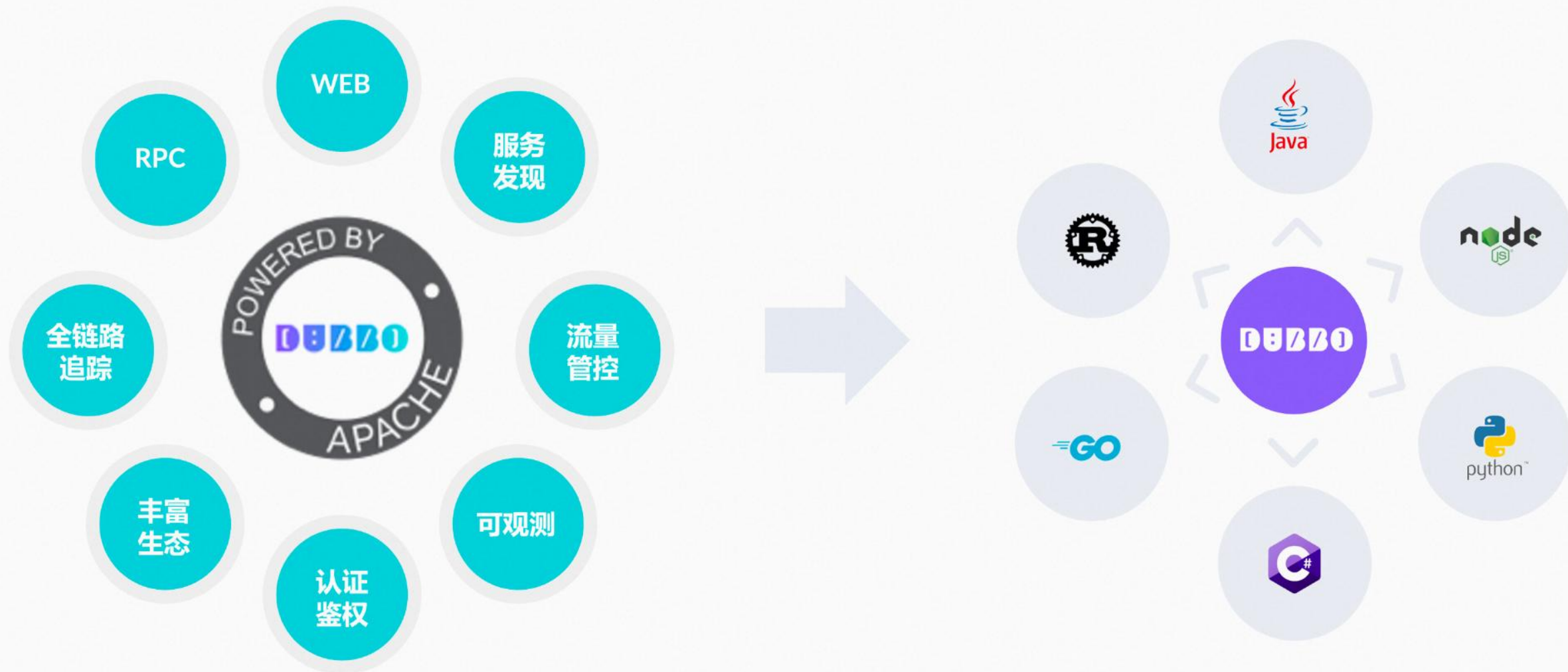


# Boss直聘微服务架构示例





# Dubbo框架



# Dubbo协议

字节

偏移字节		0								1								2								3										
偏移比特位		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31			
0	0	魔法数高位								魔法数低位								请求 响应	需要 往返	事件	序列化ID								状态							
4	32	RPC 请求ID																																		
8	64																																			
12	96	消息体数据长度																																		
16	128	dubbo version , service name , service version , method name , parameter types , arguments , attachments																																		
...	...																																			

# Dubbo服务接入注册中心

## 服务列表

新建

删除

命名空间: java-t 请选择条件进行过滤

<input type="checkbox"/> 服务名	命名空间 ▾	部门	业务	健康实例/总实例数	创建时间	修改时间	操作
<div>▶  <a href="#">com.example.bpmtest.service.DemoServiceCon</a></div>	java-t	-	-		2023-07-10 14:47:06	2023-07-10 14:47:06	  
<div>▶  <a href="#">com.example.bpmtest.service.DemoService</a></div>	java-t	-	-		2023-05-25 21:02:32	2023-05-25 21:02:32	  
<div>▶  <a href="#">org.example.DemoServiceExe</a></div>	java-t	-	-		2023-07-10 17:23:37	2023-07-10 17:23:37	  
<div>▶  <a href="#">org.example.DemoService</a></div>	java-t	-	-		2023-07-14 14:37:43	2023-07-14 14:37:43	  
<div>▶  <a href="#">com.example.bpmtest.service.DemoServiceExe</a></div>	java-t	-	-		2023-05-25 21:02:34	2023-05-25 21:02:34	  

# Dubbo服务注册-同步Istio

```
apiVersion: networking.istio.io/v1beta1
kind: ServiceEntry
metadata:
  annotations:
    interface: org.apache.dubbo.samples.basic.api.DemoService
  labels:
    manager: aeraki
    registry: dubbo-zookeeper
    name: aeraki-org-apache-dubbo-samples-basic-api-demoservice
    namespace: dubbo
spec:
  hosts:
    - org.apache.dubbo.samples.basic.api.demoservice
  location: MESH_INTERNAL
  ports:
    - name: tcp-metaprotocol-dubbo
      number: 20880
      protocol: tcp
      targetPort: 20880
  resolution: STATIC
  addresses:
    - 240.240.0.18
  endpoints:
    - address: 172.17.0.97
      labels:
        anyhost: "true"
        application: dubbo-sample-provider
        default: "true"
        deprecated: "false"
        dubbo: 2.0.2
        dynamic: "true"
        generic: "false"
        interface: org.apache.dubbo.samples.basic.api.DemoService
        metadata-type: remote
        methods: testVoid-sayHello
        pid: "7"
        registryName: default
        release: 1.0-SNAPSHOT
        revision: 1.0-SNAPSHOT
        service_group: batchjob
        side: provider
        timestamp: "1663761491410"
        version: v2
      locality: bj/800005
  ports:
    - tcp-metaprotocol-dubbo: 20880
  serviceAccount: default
```

Dubbo interface 放到 annotation 中

Dubbo interface 作为服务 host, 提供 interface 级别的服务治理 (注意由于dns规范原因, 采用全小写)

port 名前缀 tcp-metaprotocol-dubbo



# Dubbo服务注册-同步Istio

```
apiVersion: networking.istio.io/v1alpha3
kind: ServiceEntry
metadata:
  name: dubbo-demoservice
  namespace: meta-dubbo
  annotations:
    interface: org.apache.dubbo.samples.basic.api.DemoService
spec:
  hosts:
    - org.apache.dubbo.samples.basic.api.demoservice
  ports:
    - number: 20880
      name: tcp-metaprotocol-dubbo
      protocol: TCP
  workloadSelector:
    labels:
      app: dubbo-sample-provider
  resolution: STATIC
```

Dubbo interface 放到 annotation 中

host 必须填写为 dubbo interface 的小写形式

selector 匹配 deployment 的 label, 和 k8s service 类似

# Dubbo服务发现

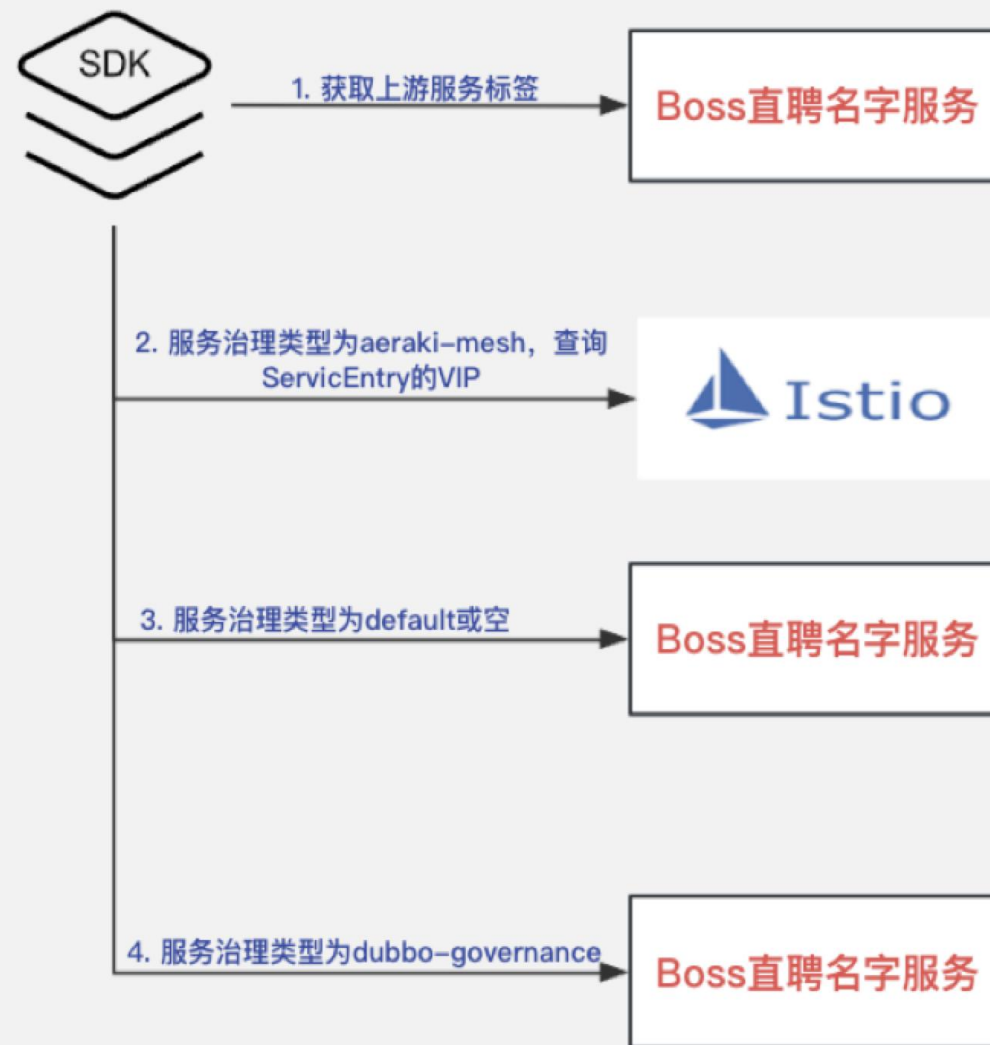
← com.tencent.polaris.dubbox.discovery.example.api.BidService(a-test)

应用实例 路由规则 熔断规则 限流规则 应用信息

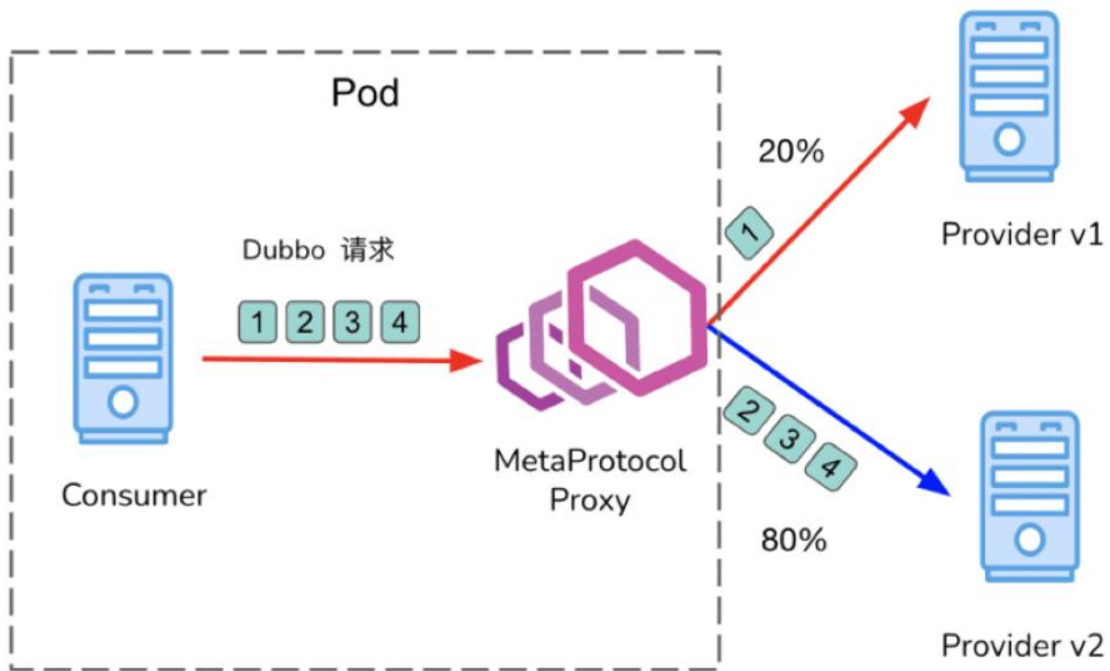
服务	a-test
服务名	com.tencent.polaris.dubbox.discovery.example.api.BidService
部门	-
业务	-
创建时间	2023-08-16 13:37:48
修改时间	2023-09-11 10:55:56
版本号	6d62022af0384157975ffa77d08c1987
应用标签(1个)	traffic-governance-type:aeraki-mesh
描述	-
就近访问	关闭

服务治理类型:

1. aeraki-mesh
2. dubbo-governance
3. defult

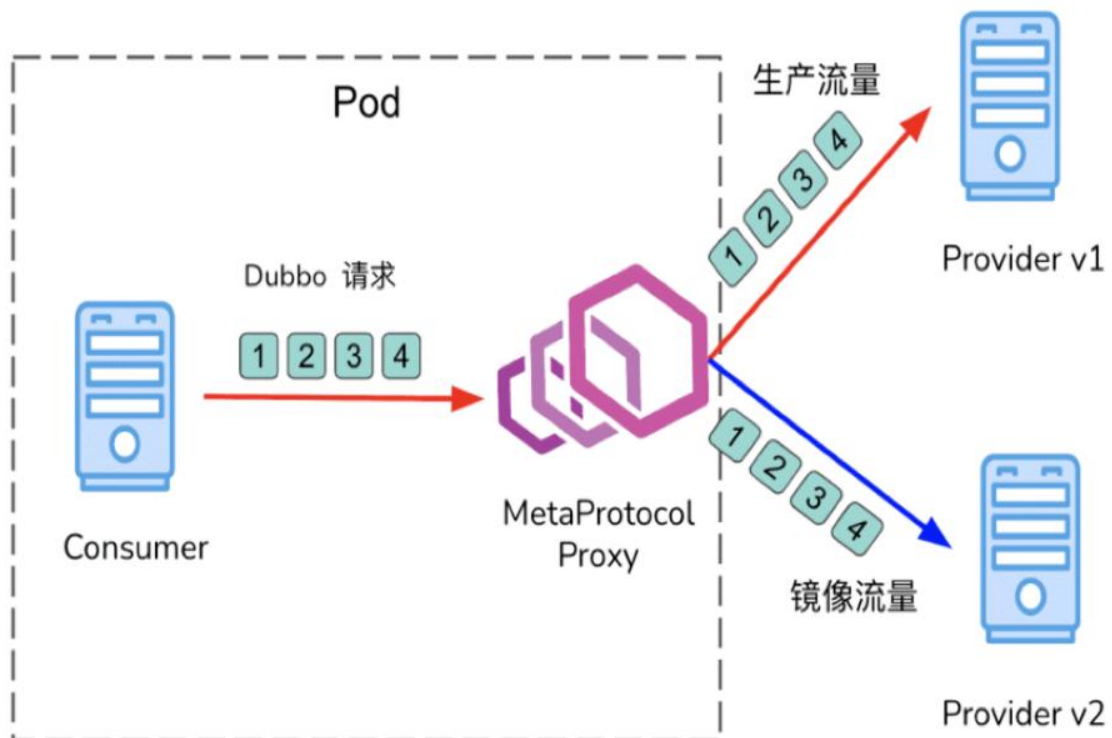


# Dubbo请求路由



```
apiVersion: metaprotocol.aeraki.io/v1alpha1
kind: MetaRouter
metadata:
  name: test-metaprotocol-dubbo-route
  namespace: meta-dubbo
spec:
  hosts:
    - org.apache.dubbo.samples.basic.api.demoservice
  routes:
    - name: traffic-split
      match:
        attributes:
          interface:
            exact: org.apache.dubbo.samples.basic.api.DemoService
          method:
            exact: sayHello
          foo:
            exact: bar
      route:
        - destination:
            host: org.apache.dubbo.samples.basic.api.demoservice
            subset: v1
            weight: 20
        - destination:
            host: org.apache.dubbo.samples.basic.api.demoservice
            subset: v2
            weight: 80
```

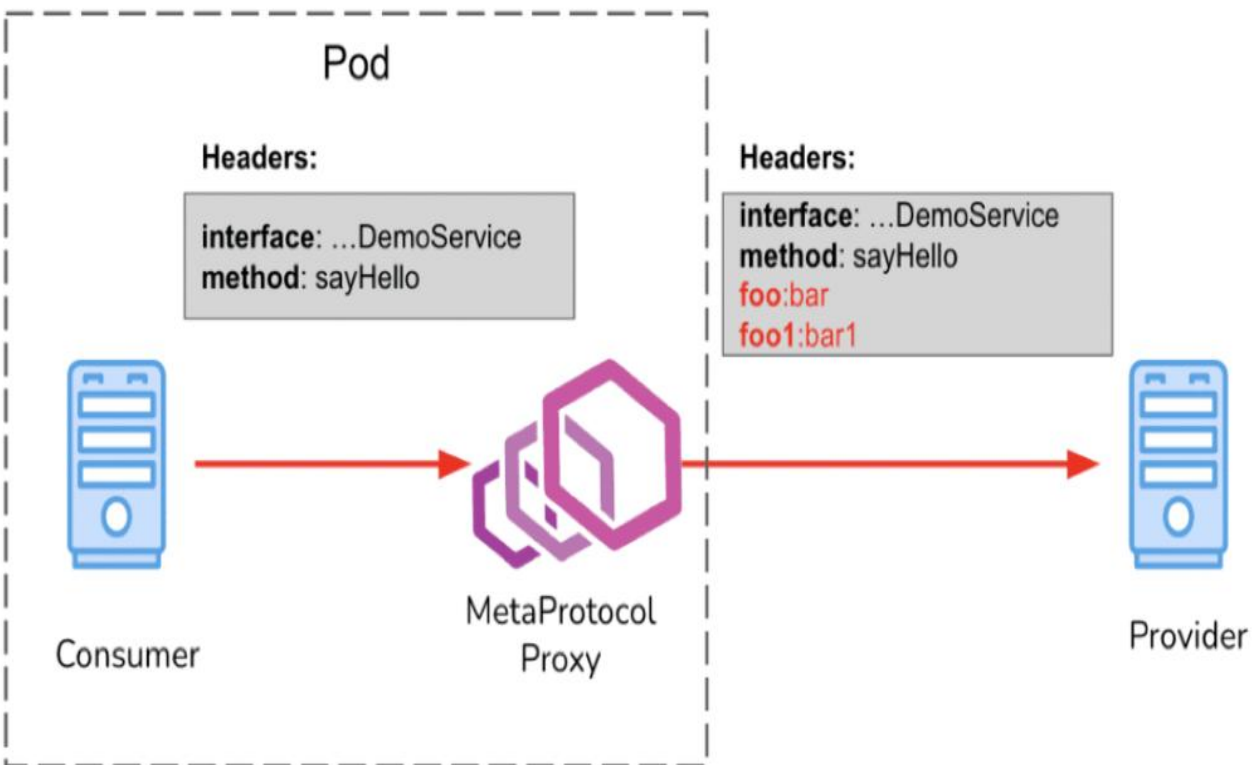
# Dubbo流量镜像



```
apiVersion: metaprotocol.aeraki.io/v1alpha1
kind: MetaRouter
metadata:
  name: test-metaprotocol-dubbo-route
  namespace: meta-dubbo
spec:
  hosts:
    - org.apache.dubbo.samples.basic.api.demoservice
  routes:
    - name: traffic-mirroring
      route:
        - destination:
            host: org.apache.dubbo.samples.basic.api.demoservice
            subset: v1
  mirror:
    host: org.apache.dubbo.samples.basic.api.demoservice
    subset: v2
  mirrorPercentage:
    value: 100.0
```

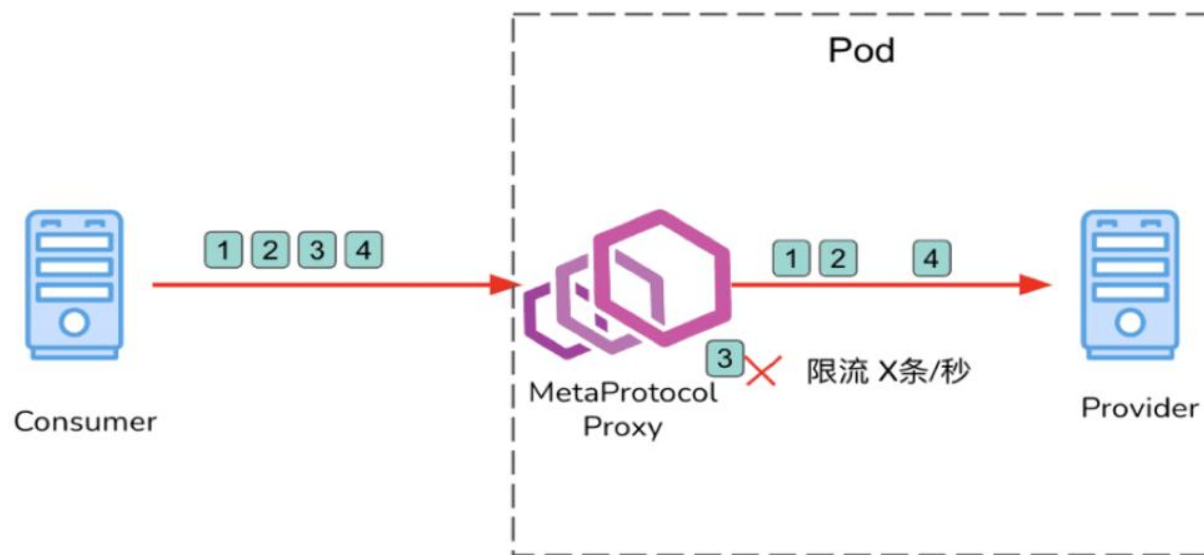


# Dubbo请求头修改



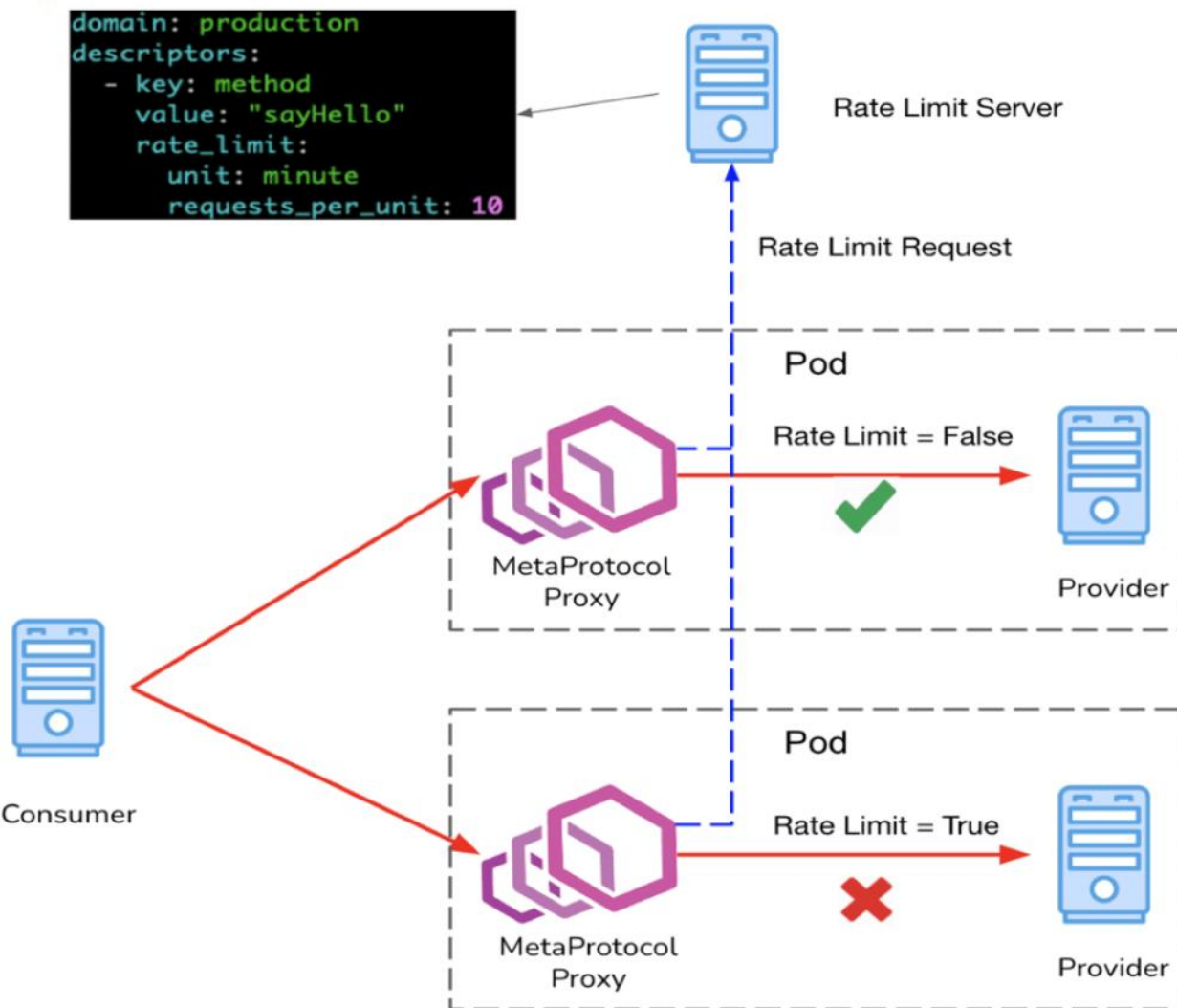
```
apiVersion: metaprotocol.aeraki.io/v1alpha1
kind: MetaRouter
metadata:
  name: test-metaprotocol-dubbo-route
  namespace: meta-dubbo
spec:
  hosts:
    - org.apache.dubbo.samples.basic.api.demoservice
  routes:
    - name: header-mutation
      route:
        - destination:
            host: org.apache.dubbo.samples.basic.api.demoservice
      requestMutation:
        - key: foo
          value: bar
        - key: foo1
          value: bar1
```

# Dubbo本地限流



```
apiVersion: metaprotocol.aeraki.io/v1alpha1
kind: MetaRouter
metadata:
  name: test-metaprotocol-dubbo-route
  namespace: meta-dubbo
spec:
  hosts:
    - org.apache.dubbo.samples.basic.api.demoservice
  localRateLimit:
    tokenBucket:
      fillInterval: 60s
      maxTokens: 10
      tokensPerFill: 10
    conditions:
      - tokenBucket:
          fillInterval: 20s
          maxTokens: 5
          tokensPerFill: 5
        match:
          attributes:
            method:
              exact: sayHello
  routes:
    - name: v1
      route:
        - destination:
            host: org.apache.dubbo.samples.basic.api.demoservice
            subset: v1
```

# Dubbo全局限流



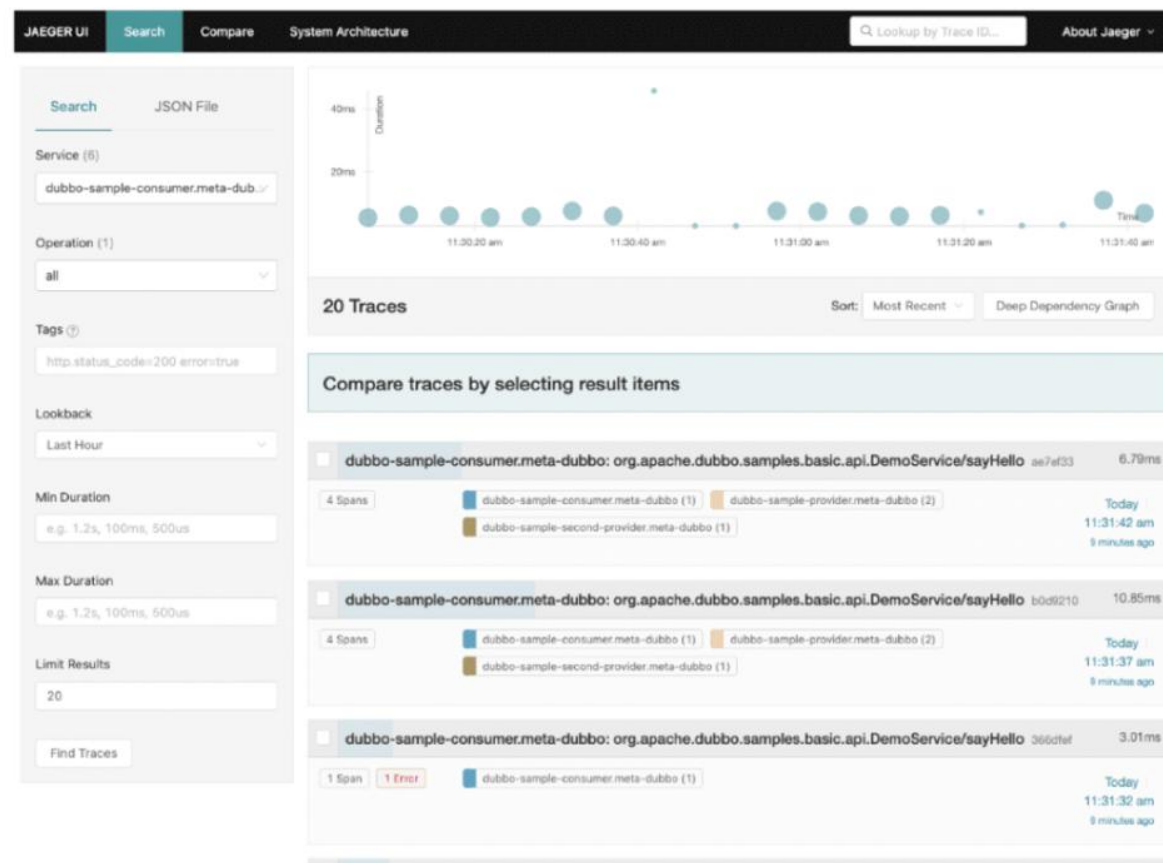
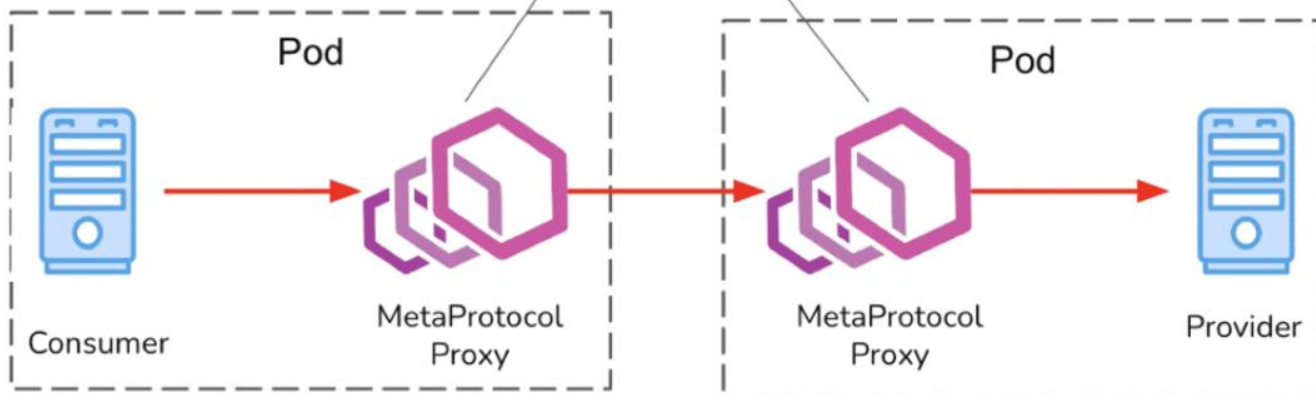
```
apiVersion: metaprotocol.aeraki.io/v1alpha1
kind: MetaRouter
metadata:
  name: test-metaprotocol-dubbo-route
  namespace: meta-dubbo
spec:
  hosts:
  - org.apache.dubbo.samples.basic.api.demoservice
  globalRateLimit:
    domain: production
    match:
      attributes:
        method:
          exact: sayHello
    rateLimitService: outbound|8081|rate-limit-server.meta-dubbo.svc.cluster.local
    requestTimeout: 100ms
    denyOnFail: true
    descriptors:
    - property: method
      descriptorKey: method
```

# Dubbo分布式追踪 (零代码修改)



client span

server span



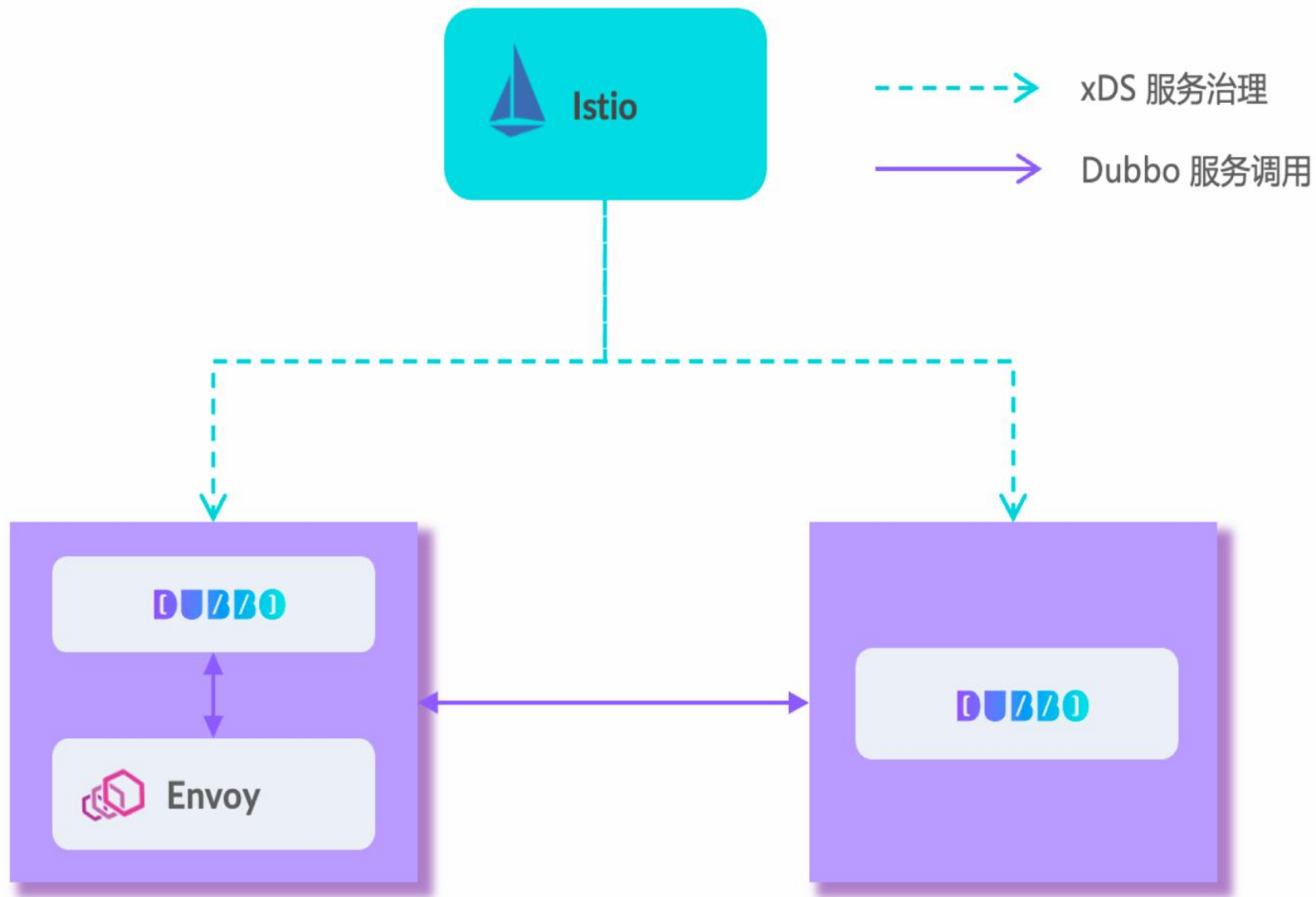
备注：因为 Dubbo 会在同一个线程的上下游请求之间传递 header，因此可以通过 Aeraki Mesh 实现完全无侵入的调用跟踪



# Dubbo Metrics



# 补充1



## 打造基于Dubbo框架以及Aeraki Mesh的统一服务治理平台

