

Dubbo Ecosystem

Dubbo-go-pixiu 的过去、现在和未来

铁城（鐵手）

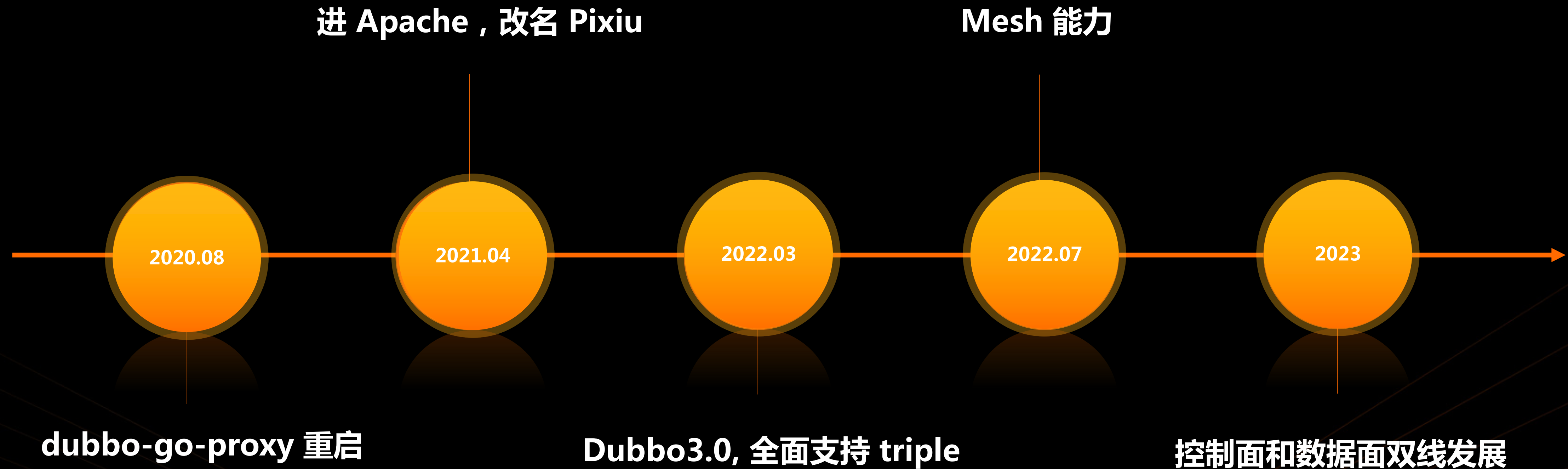
Apache Dubbogo Committer

- 1 Pixiu 的诞生和发展
- 2 Pixiu 的基础能力
- 3 云原生下的演进
- 4 Pixiu 的未来发展

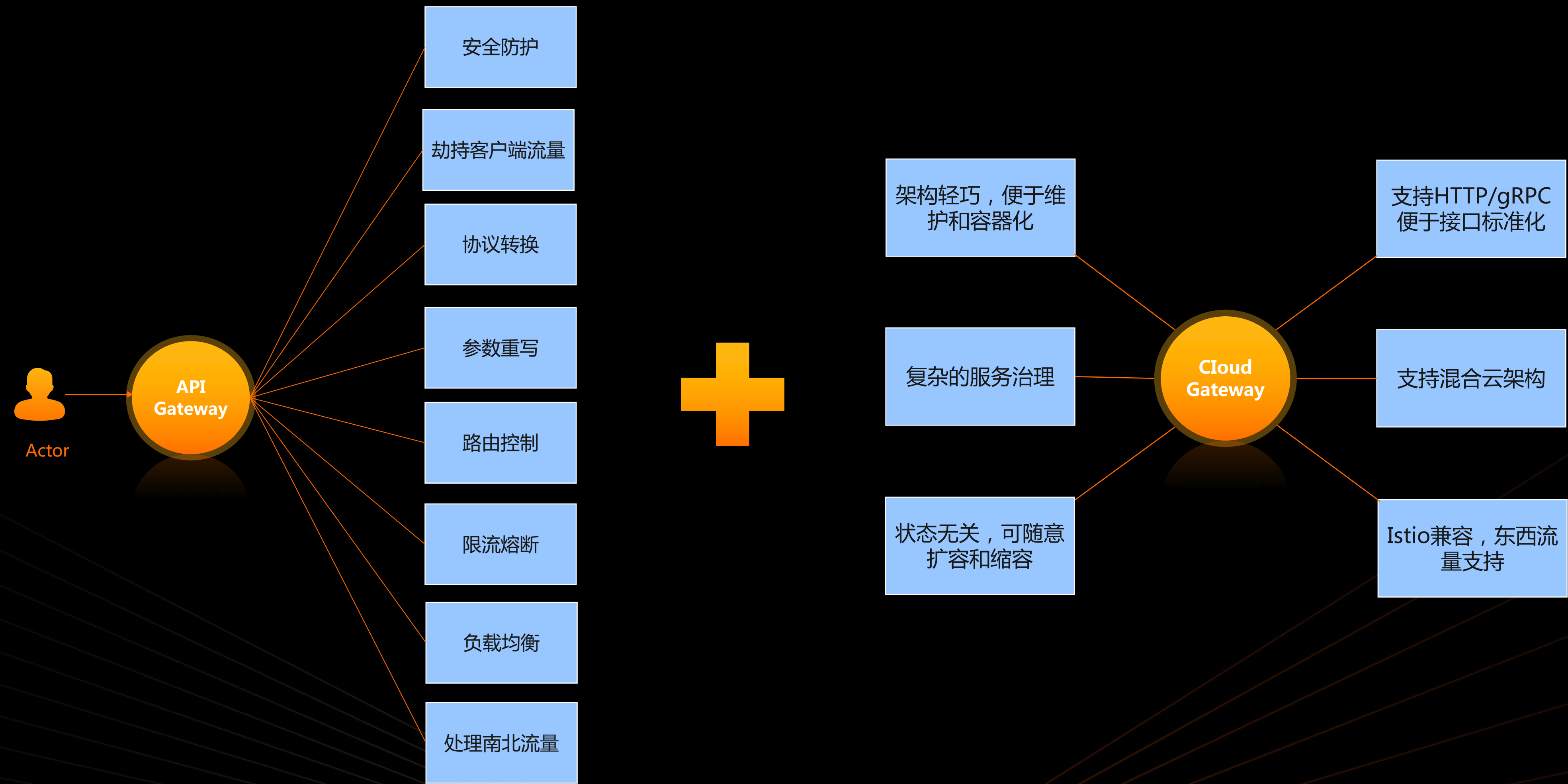
一、Pixiu 的诞生和发展

Pixiu 的历程

中间件开发者沙龙



云时代的网关





协议代理

- dubbo、grpc和http

协议互转

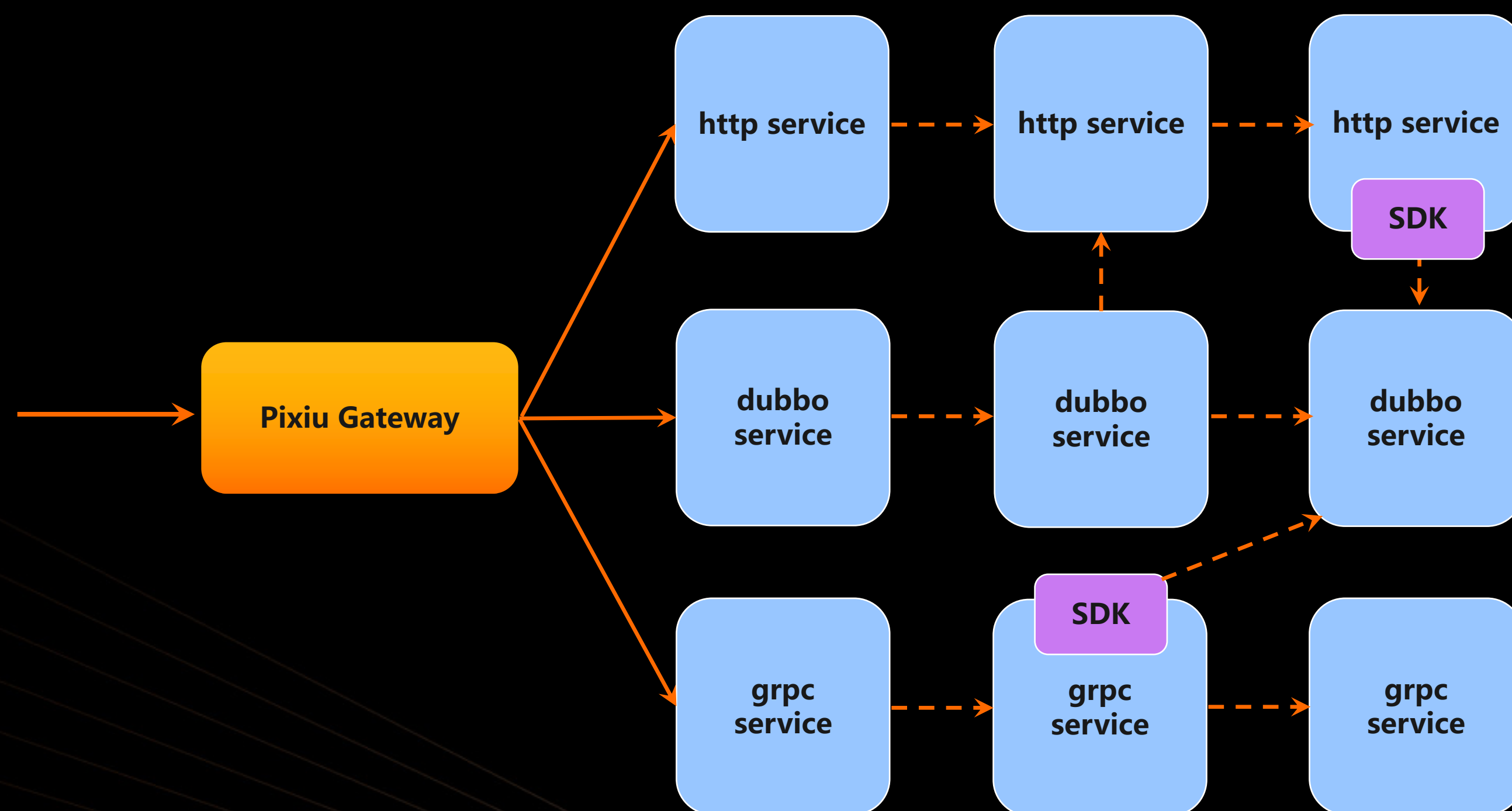
- dubbo、triple和grpc

支持 dubbo 和Spring Cloud 集群自
动路由发现

Dubbo 的多语言解决方案

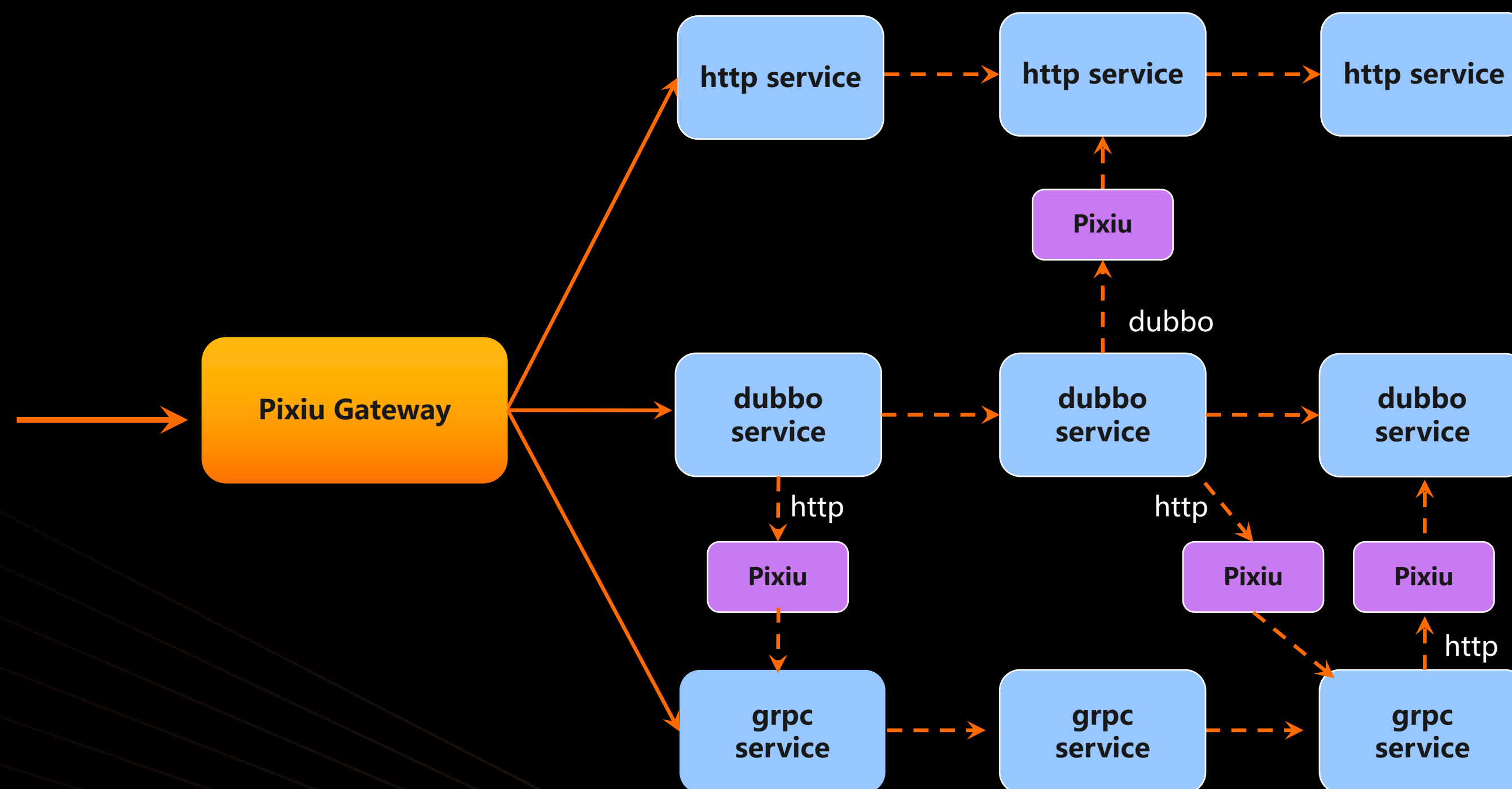
- 生态现状：主要是 Java 和 Golang的发展
- 通过 HTTP/gRPC 标准化接口，代理其他语言接入到 dubbo 集群

二、Pixiu 当前的能力



协议代理

- HTTP --> DUBBO
- HTTP --> GRPC
- HTTP --> HTTP



当前支持的协议

- http to dubbo2
- http to triple
- http to grpc
- dubbo2 to triple
- dubbo2 to http
- triple to dubbo2
- triple to http

Http 自动转 dubbo

中间件开发者沙龙

```
static_resources:
  listeners:
    - name: "http-listener"
      protocol_type: "HTTP"
      address:
        socket_address:
          address: "0.0.0.0"
          port: 8888
      filter_chains:
        filters:
          - name: dgp.filter.httpconnectionmanager
            config:
              route_config:
                routes:
                  - match:
                      prefix: "*"
              http_filters:
                - name: dgp.filter.http.dubboproxy
                  config:
                    dubboProxyConfig:
                      auto_resolve: true
                      registries:
                        "zookeeper":
                          protocol: "zookeeper"
                          timeout: "3s"
                          address: "127.0.0.1:2181"
                          username: ""
                          password: ""
                      timeout_config:
```

根据 http 请求的 URL 和 Head 信息获取到泛化调用元数据

POST \${host}:\${port}/#{interfaceKey}/#{method}

"x-dubbo-http1.1-dubbo-version": 1.0.0

"x-dubbo-service-protocol": #{protocol}

"x-dubbo-service-version": #{version}

"x-dubbo-service-group": #{group}

#{body}

Dubbo 现状：

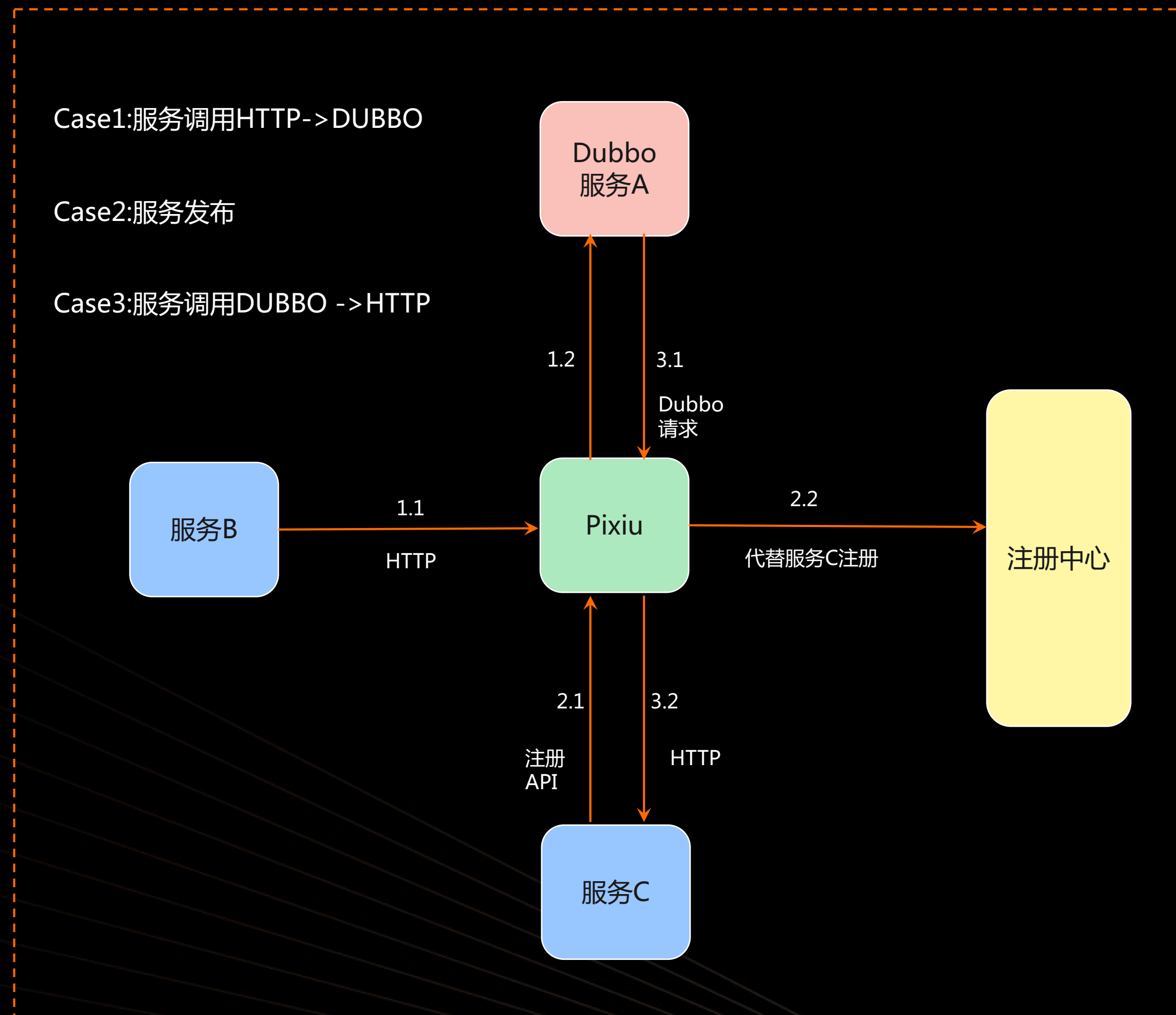
- 多语言支持能力不足，较为成熟的只有 Java 和 Go
- 框架强关联，代码耦合程度高

代理/Sidecar模式：

- 无语言耦合
- 轻量级
- 流量管控
- 性能损耗

Pixiu 多语言解决方案

中间件开发者沙龙



核心功能：

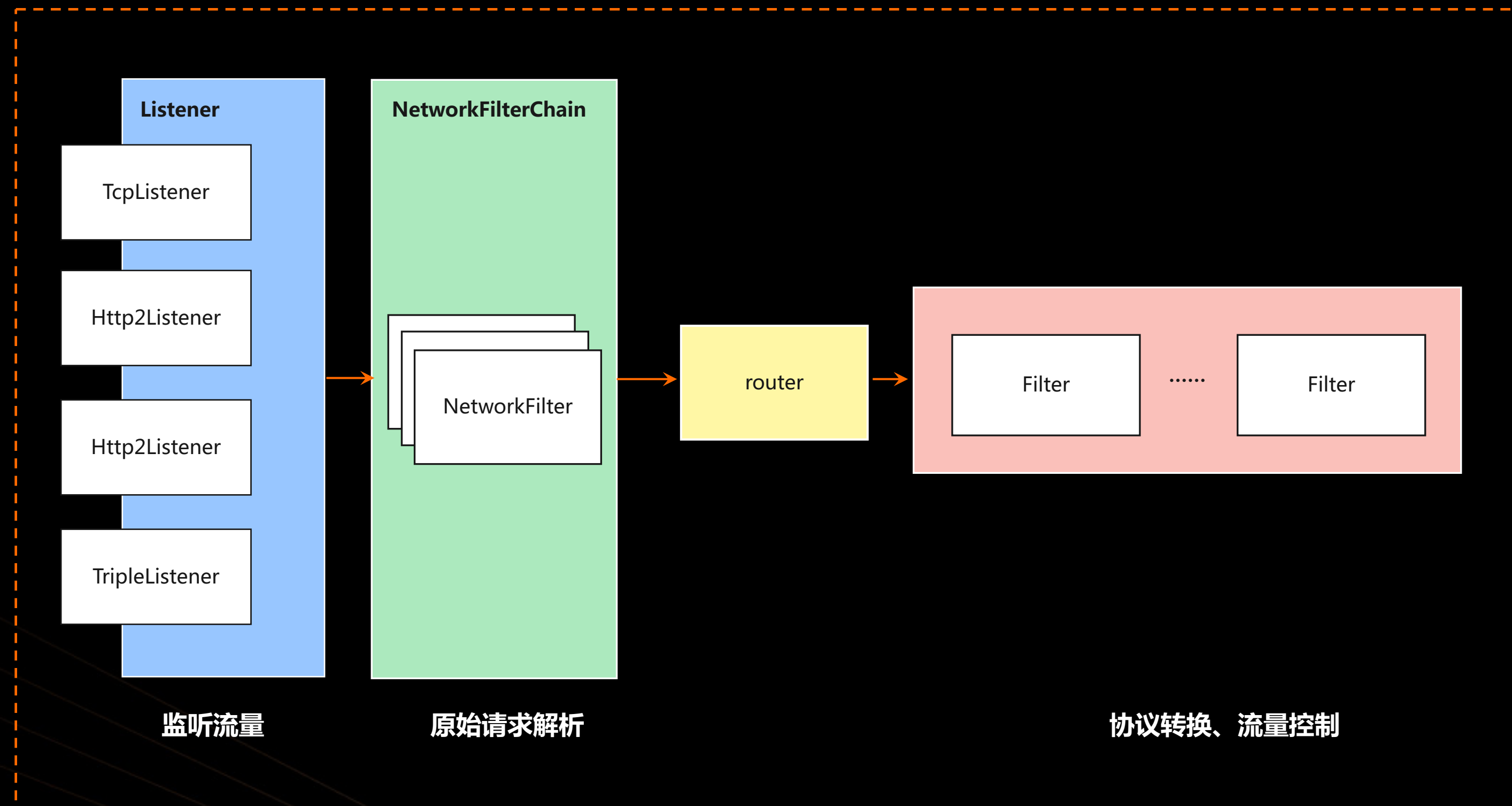
- 请求代理
- 代理注册
- 流量治理

使用场景：

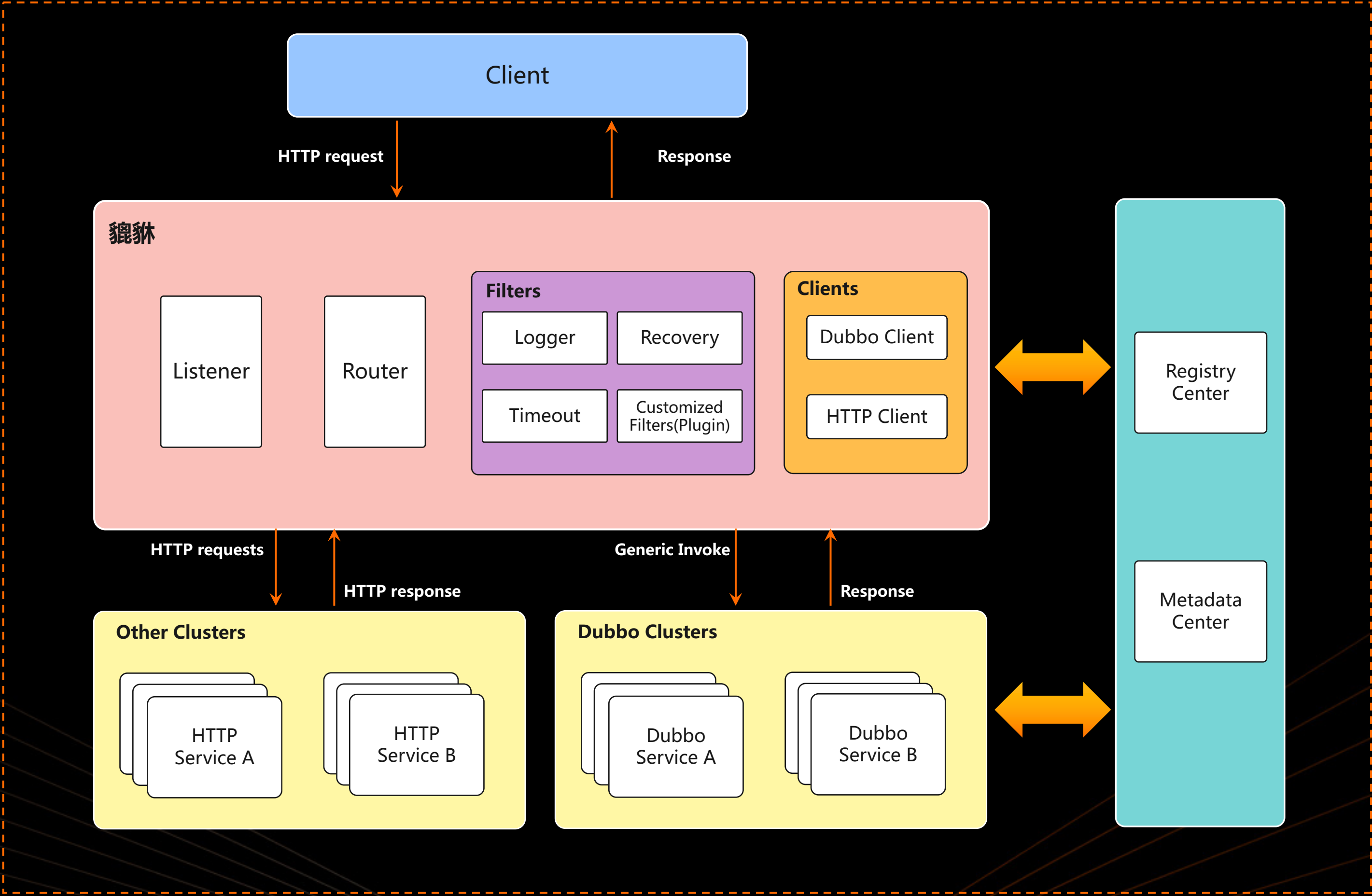
- python , js 服务接入 dubbo 集群
- 不同协议的 dubbo 集群互通

核心设计

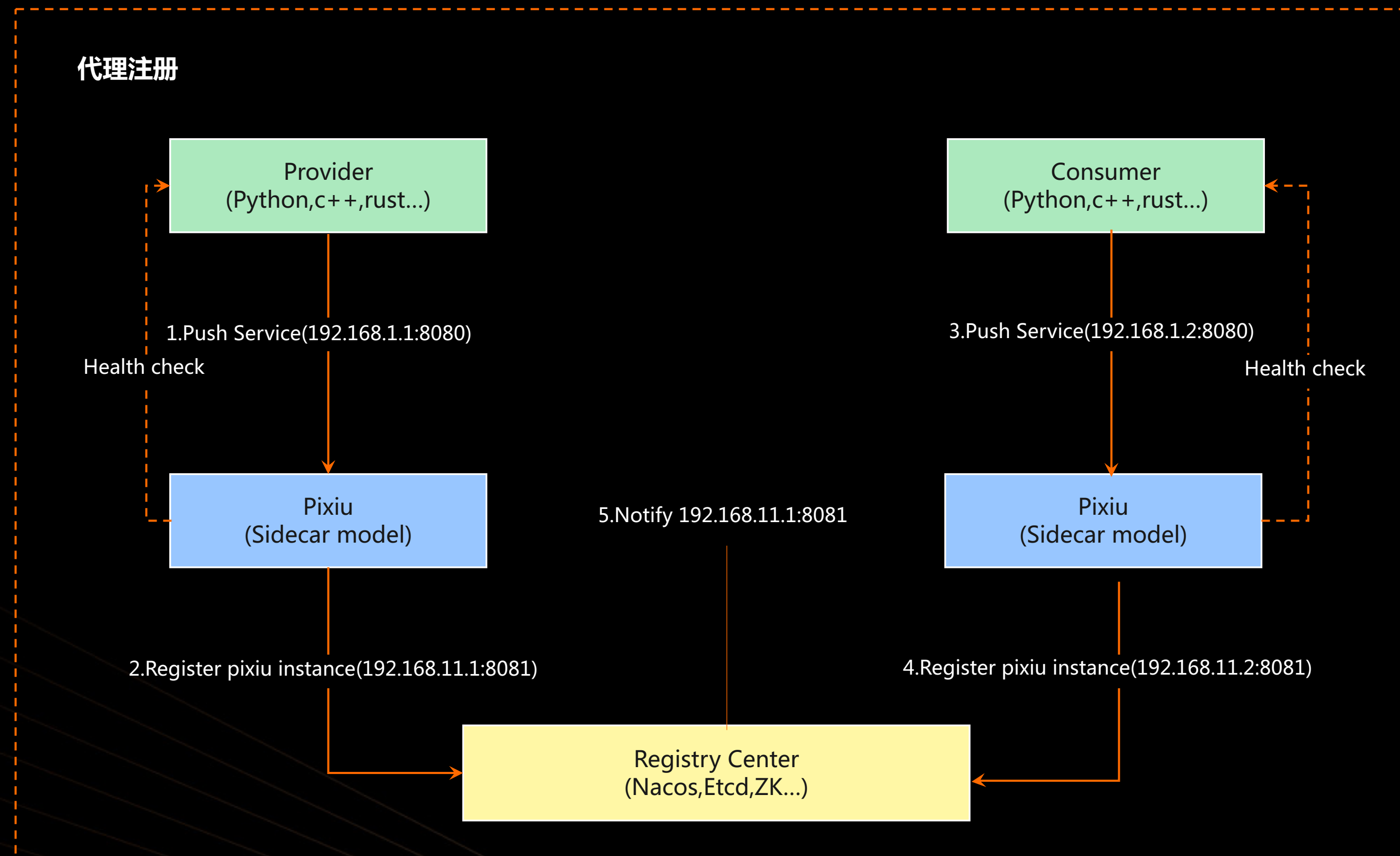
中间件开发者沙龙



- Listener
- NetworkFilter
- HttpFilter&DubboFilter
- Route
- Cluster



三、云原生下的演进

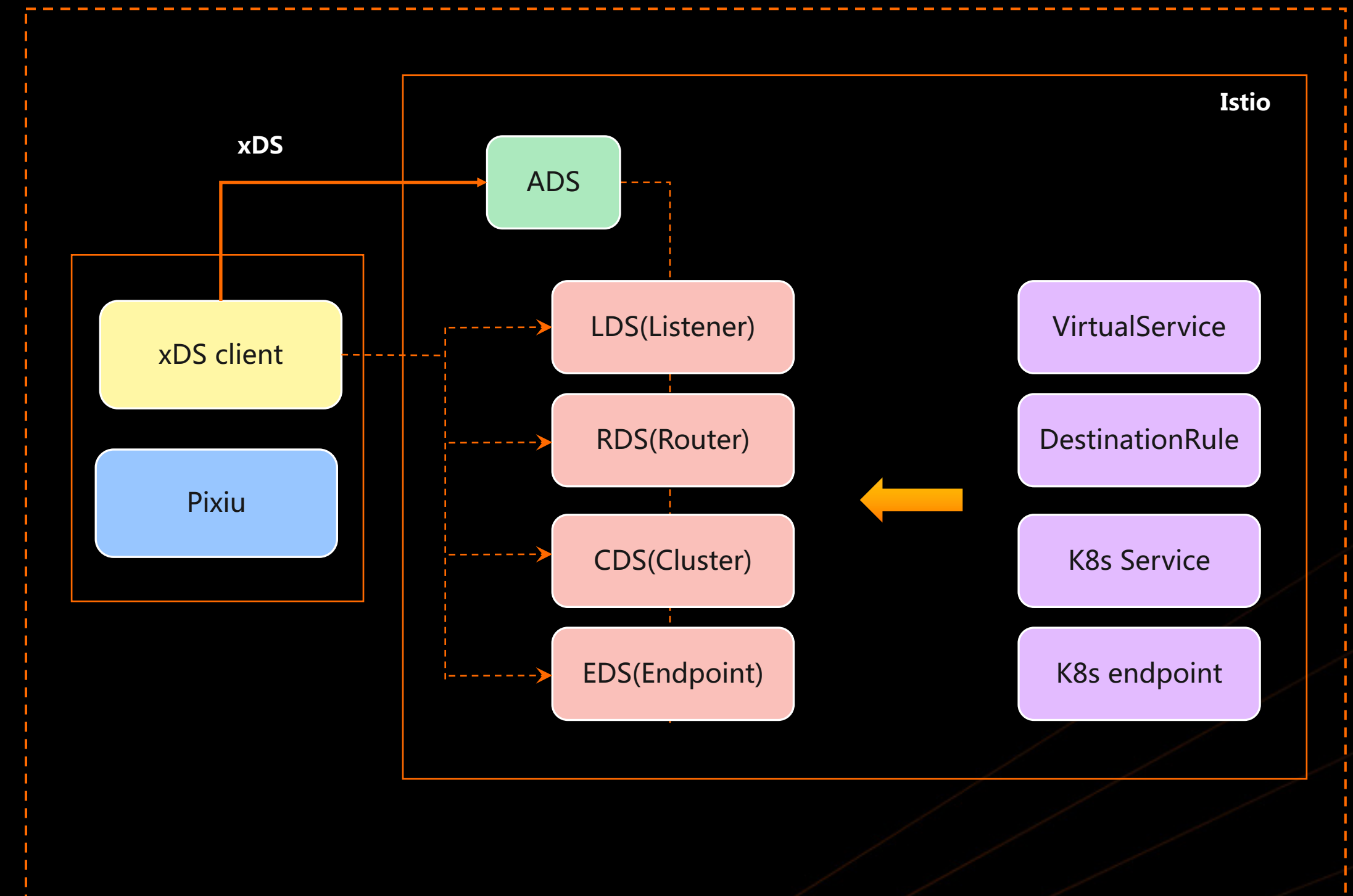
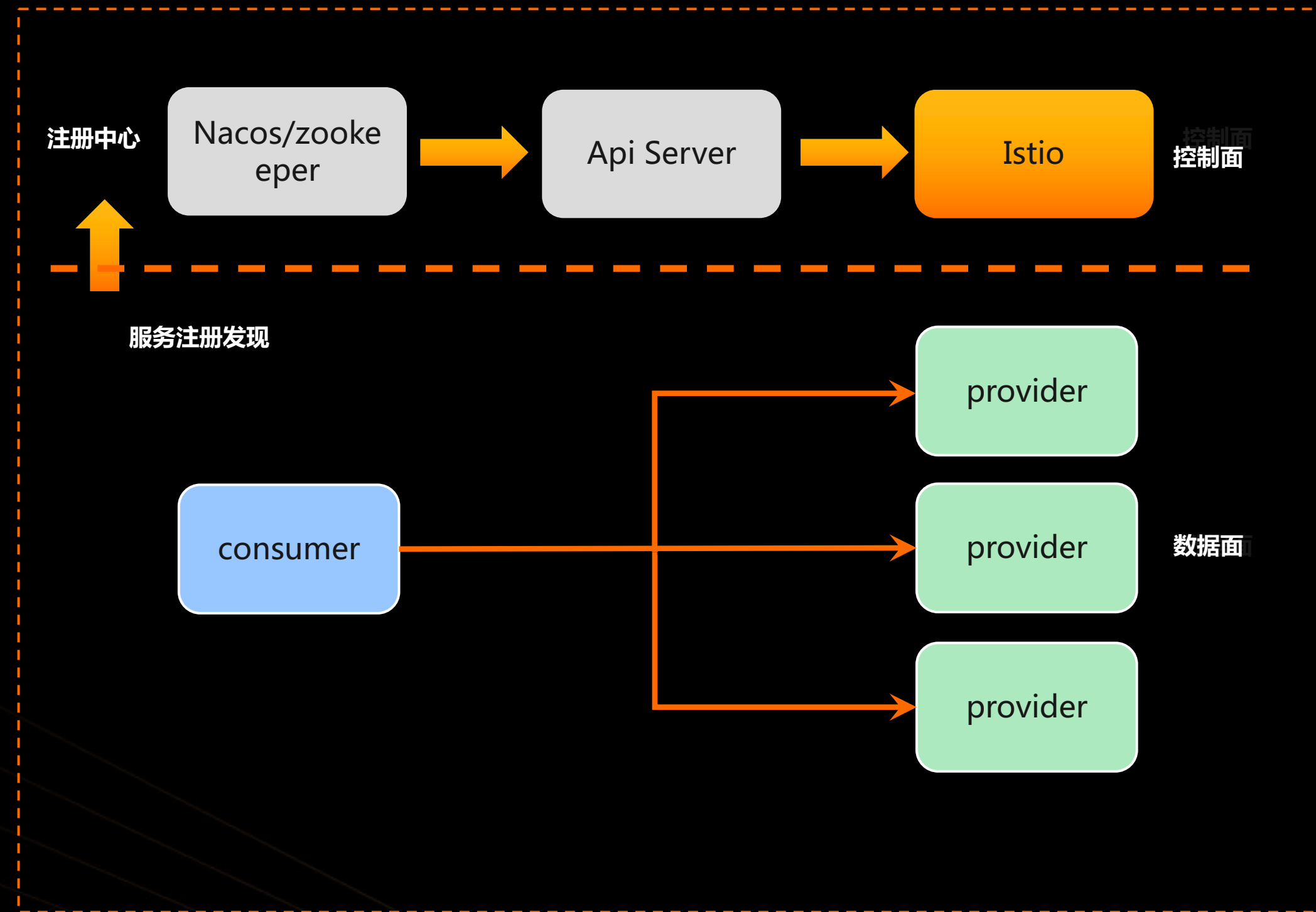


场景：

- 覆盖常规 dubbo -> http
- Dubbo 2.0 和 3.0 混部
- 两个独立集群，通过 Pixiu 交互

xDS 集成

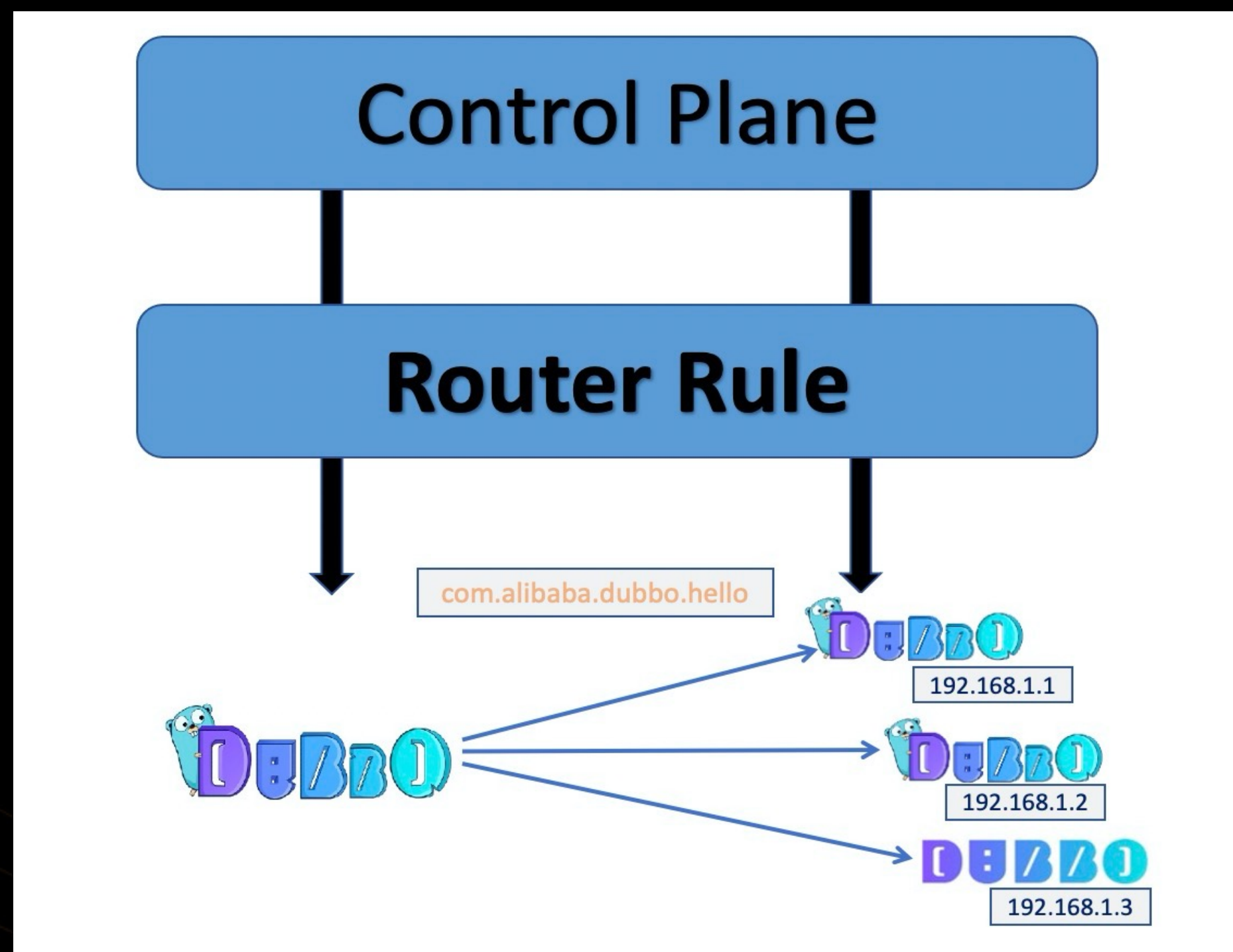
中间件开发者沙龙



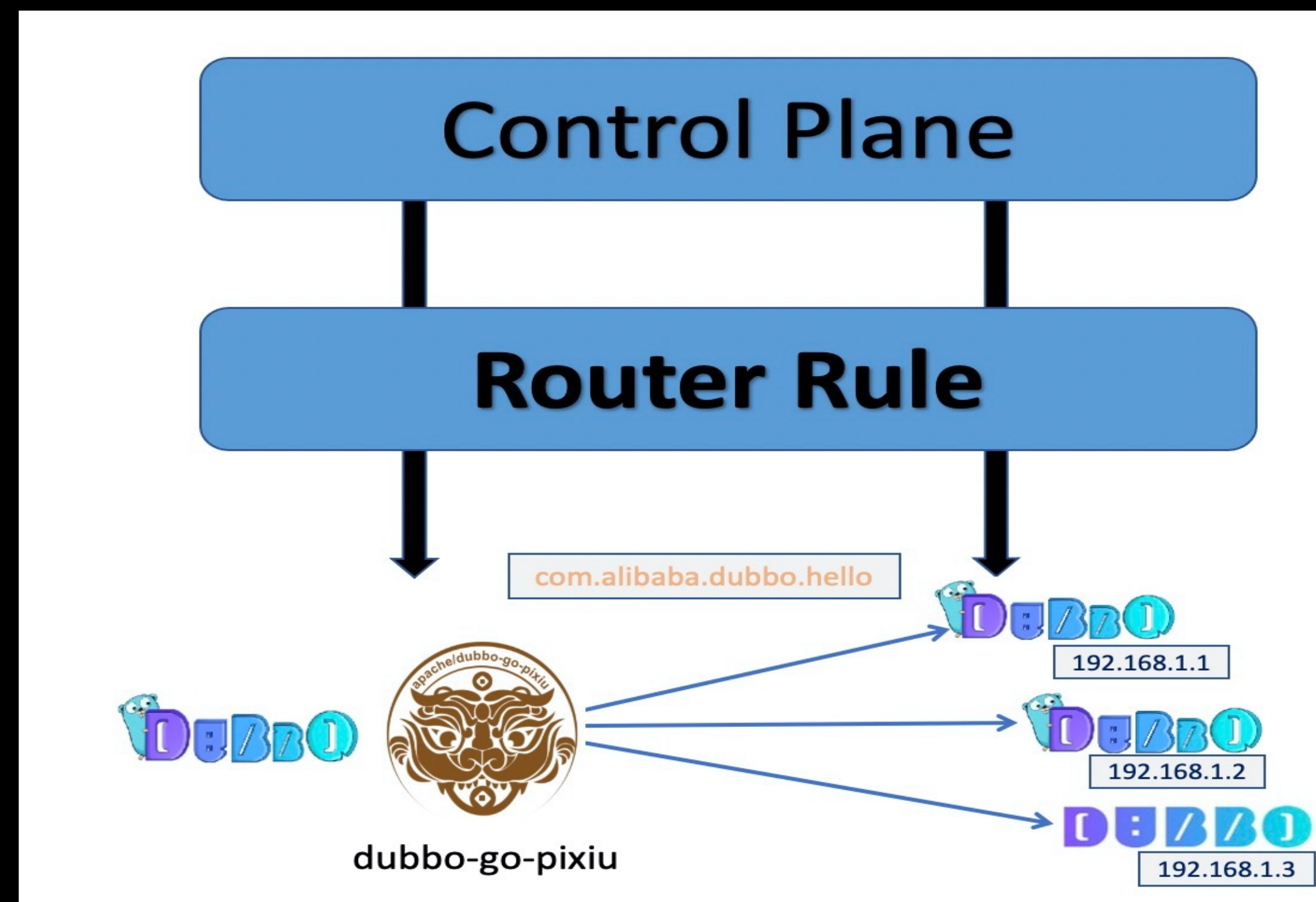
四、未来发展

Mesh 发展

中间件开发者沙龙



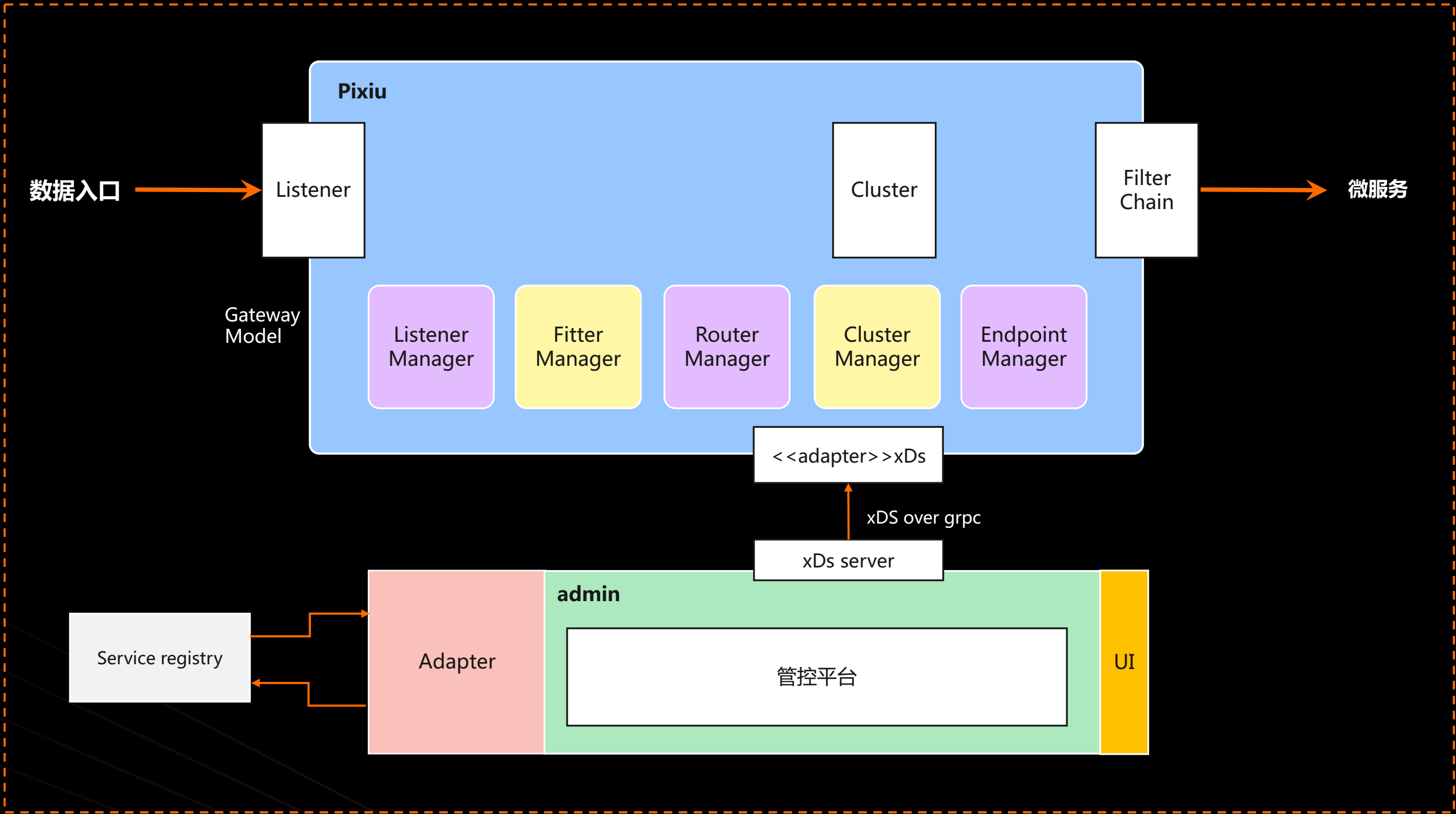
Proxyless Service Mesh



Proxy Service Mesh

未来数据面和控制板双线发展

中间件开发者沙龙

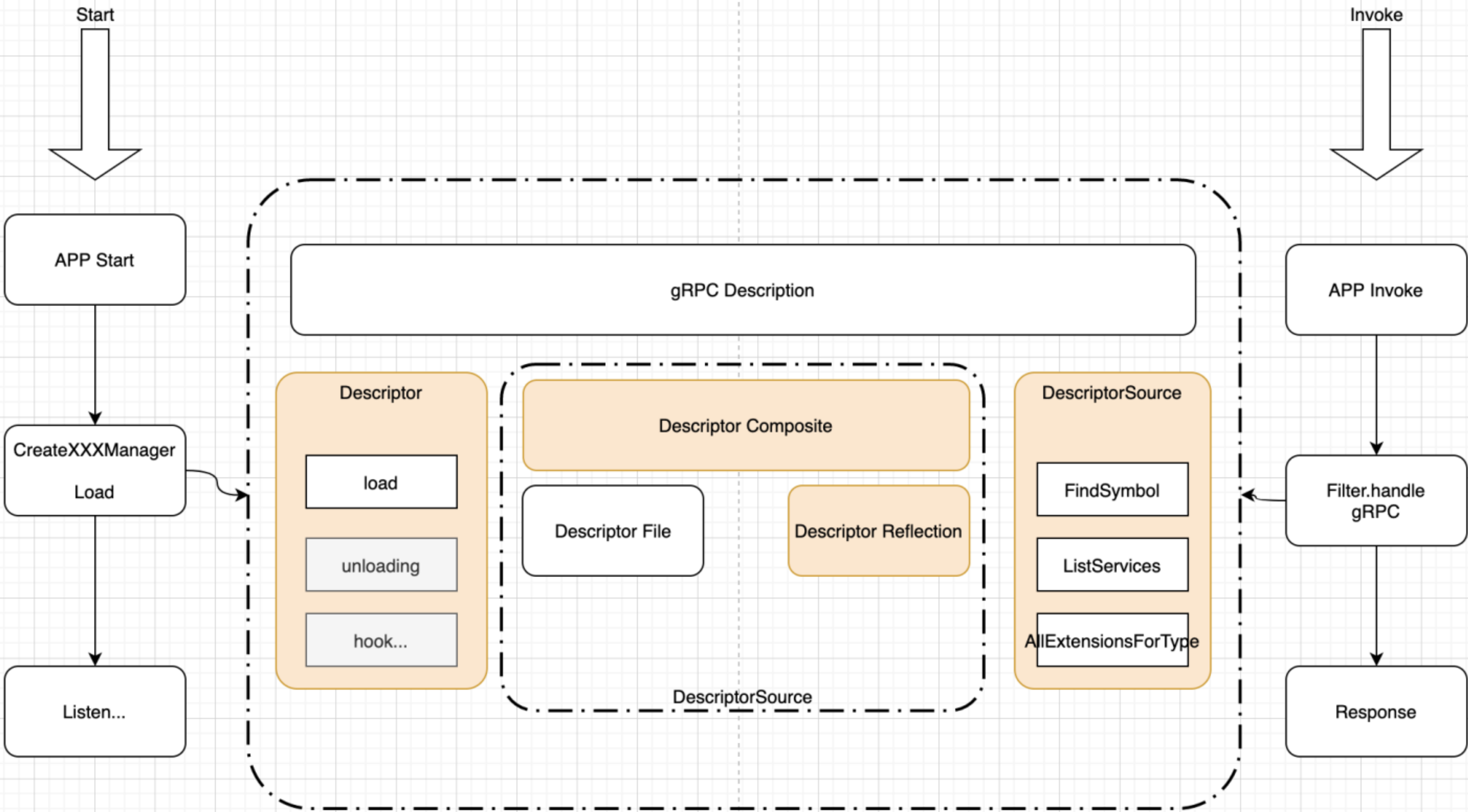


能力建设：

- 基于go-control-plane
完善、增强 xDs 能力
- 已支持 CDS 和 LDS
- 待支持 LDS 和 RDS

中间件开发者沙龙

Pixiu & gRPC V2



```
adapters:
  - id: "springcloud"
    name: "dgp.adapter.springcloud"
    config:
      freshInterval: 60s # 刷新配置时间
    registry:
      name: naocs
      protocol: nacos
      group: DEFAULT_GROUP
      timeout: 3s
      address: 127.0.0.1:8848
```

Adapter模块

Adapter 则表达 pixiu 和外界进行元数据获取的能力。能够根据服务中的服务元数据，进行路由和集群信息的动态获取。目前 pixiu 支持两种adapter，分别是 dubbo 集群和 springcloud 集群获取信息。右侧图是springcloud 集群adapter的配置信息。


```
POST /com.dubbogo.pixiu.UserService/GetUserByname
```

```
x-dubbo-http1.1-dubbo-version: 1.0.0
```

```
x-dubbo-service-protocol: dubbo
```

```
x-dubbo-service-version: 1.0.0
```

```
x-dubbo-service-group: test
```

```
.....
```

```
{"types":"string","values":"tc"}
```

```
- path: '/api/v1/test-dubbo/user'
  type: restful
  methods:
    - httpVerb: POST
      inboundRequest:
        requestType: http
      integrationRequest:
        requestType: dubbo
        mappingParams:
          - name: requestBody._all
            mapTo: 0
            mapType: "object"
        interface: "com.dubbogo.pixiu.UserService"
        method: "CreateUser"
        group: "test"
        version: 1.0.0
```

