



# 2024 AI+研发数字峰会

AI+ Development Digital summit

AI驱动研发变革 促进企业降本增效

北京站 08/16-17

## 大语言模型服务管理的实践分享

王夕宁/马元元 阿里云



## 王夕宁

阿里云容器服务技术研发负责人

---

阿里云容器服务Kubernetes及服务 Mesh技术研发负责人,拥有100多项相关领域的国际技术专利,专注于Kubernetes/云原生/服务网格等领域。曾在IBM研发中心工作,担任资深架构师和技术专家,主导和参与了一系列 SOA 中间件和云计算领域的产品研发,并曾担任中国研发中心专利技术评审委员会主席。出席过行业内多个技术大会,包括 Kubecon、InfoQ、ArchSummit、IstioCon 和云栖大会等。同时,著有畅销书《Istio 服务网格解析与实战》。



# 目录

## CONTENTS

1. LLM服务管理的特征与挑战
2. 应对思路与方案
3. 现有的技术基础之上扩展支持
4. MSM: 用于管理 GenAI/LLM 工作负载的统一方式

# PART 01

## LLM服务管理的特征与挑战

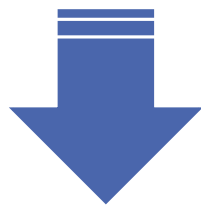


# ▶ GenAI/LLM服务管理面临独特的挑战

传统网络流量管理	GenAI/LLM流量管理
•请求/响应大小较小	•由于多模态流量，请求/响应大小较大
•许多查询可以并行处理	•单个大语言模型查询经常占用100%的TPU/GPU计算时间
•请求一到达就进行处理	•请求等待可用的计算资源
•处理时间以毫秒计算	•处理时间从几秒到几分钟不等
•相似请求可以从缓存中得到处理	•每次请求通常生成唯一内容
•请求成本由后端管理	•根据请求将流量路由到更便宜或更昂贵的模型
•传统的轮询或基于利用率的流量管理	•具备AI感知的负载均衡能力

# ▶ 流量请求调度 Traffic Request Scheduling

- 由于GenAI/LLM模型的自回归特性，LLM推理请求的有效服务面临不可预测的执行时间的挑战。
- LLM服务系统大多采用先进先出（FCFS）调度，遭受行首阻塞（head-of-line）问题。



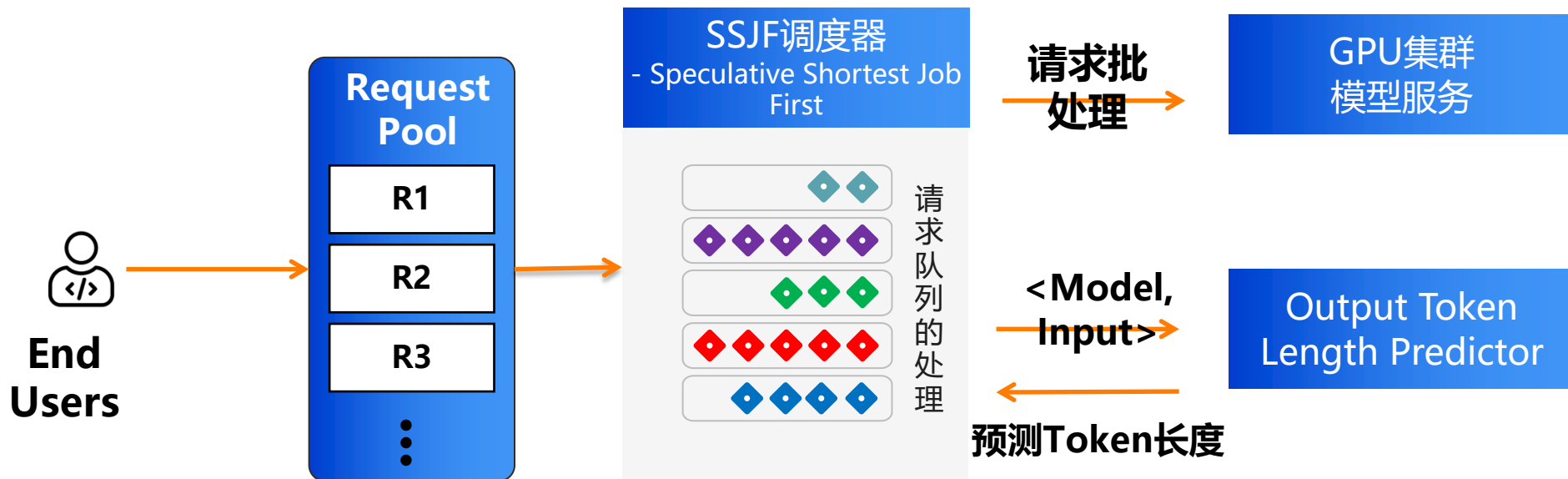
- ❖ 基于历史数据和模型特性，训练出一个代理模型，用于预测每个推理请求的序列长度。
- ❖ 利用代理模型的序列长度预测的推测最短作业优先（SSJF）调度器。

# **PART 02**

## **应对思路与方案**



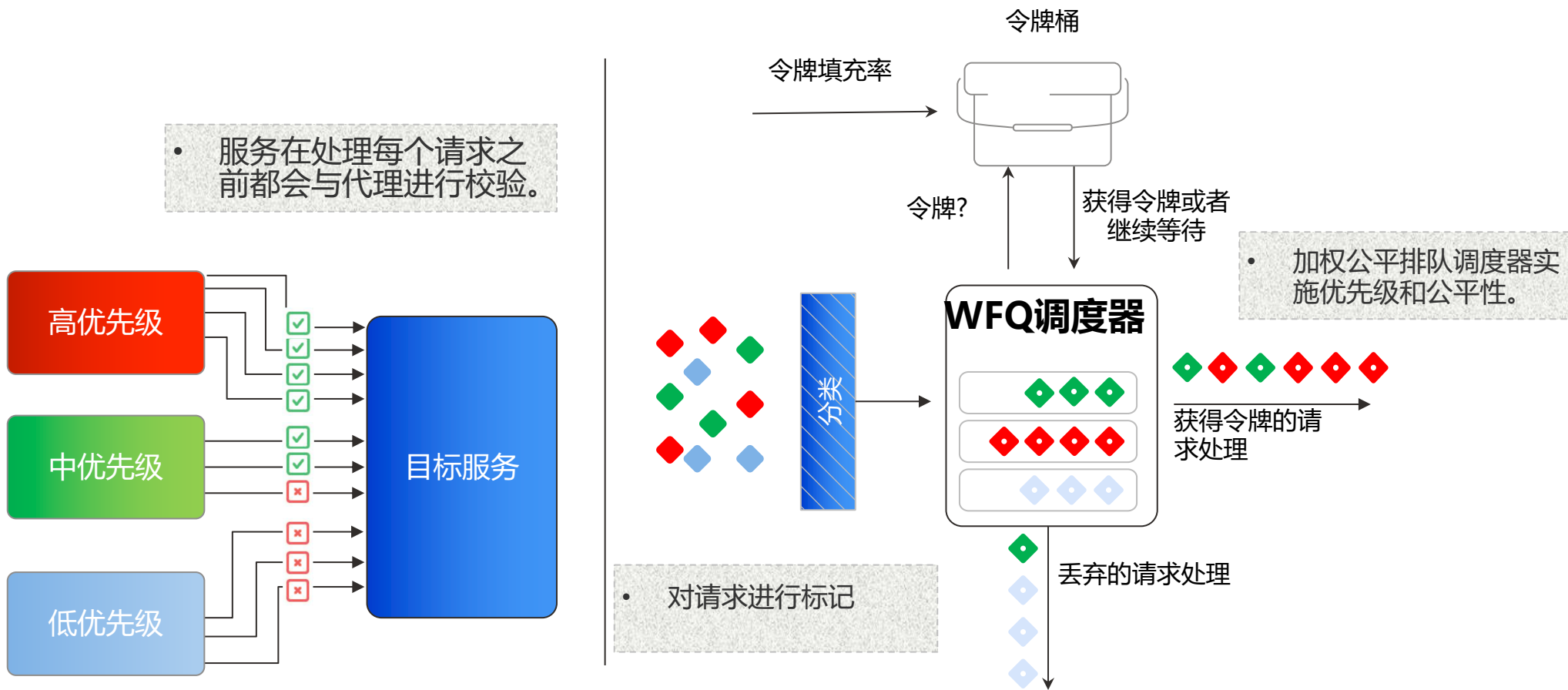
# ▶ SSJF调度器 - 引入Token长度预测器



- 输出Token长度 (N) 决定了请求的执行时间 (T) , 因为  $T = C + K \times N$  ,
  - K是生成一个标记的延迟,
  - C是模型服务系统的开销, 包括DNS查找、代理、排队和输入标记化。
  - K取决于模型优化技术 (例如, 量化) 和执行环境 (例如, 硬件) , 对于所有输入都是相同的。
- 输出Token长度决定执行时间 (线性关系)

Ref: <https://github.com/James-QiuHaoran/LLM-serving-with-proxy-models>

# 智能工作负载优先级调度



# 智能工作负载管理 – 流量调度管理套件

## 工作负载优先级调度策略

- 优先处理工作负载，保障关键用户体验路径
- 使用权重公平排队，根据业务价值和请求紧急程度调整资源配置，来实现应用程序的优雅降级

## 并发速率限制策略

- 自适应调整请求速率限制，保护服务不受过载和级联故障的影响
- 通过细粒度标签识别单个用户，根据业务特定标签控制爆发能力和填充速率；
- 限制每个用户或全局并发中请求的并发量；

## 配额调度策略

- 使用全局令牌桶和智能请求排队，根据重要性安排请求
- 和限流不同，若请求速率超过限制，此时请求不会被直接拒绝，而是进入一个优先级队列，在保证请求速率始终在限制内的同时对请求进行优先级调度。

## 并发调度策略

- 通过限制并发中请求的数量，防范服务突然过载。
- 超出此限制的任何请求将进入队列，并根据它们的优先级在有能力提供服务时予以处理
- 用于根据重要性调度请求，同时确保应用遵守并发限制。

## 负载坡道策略

- 基于闭环反馈来逐步增加系统的工作负荷或请求量，而不是瞬间施加大的负载。
- 能够帮助系统逐步适应增加的负荷，从而确保系统在负载增加过程中仍然稳定运行，并最大限度地减少对系统的冲击。

## 流量缓存策略

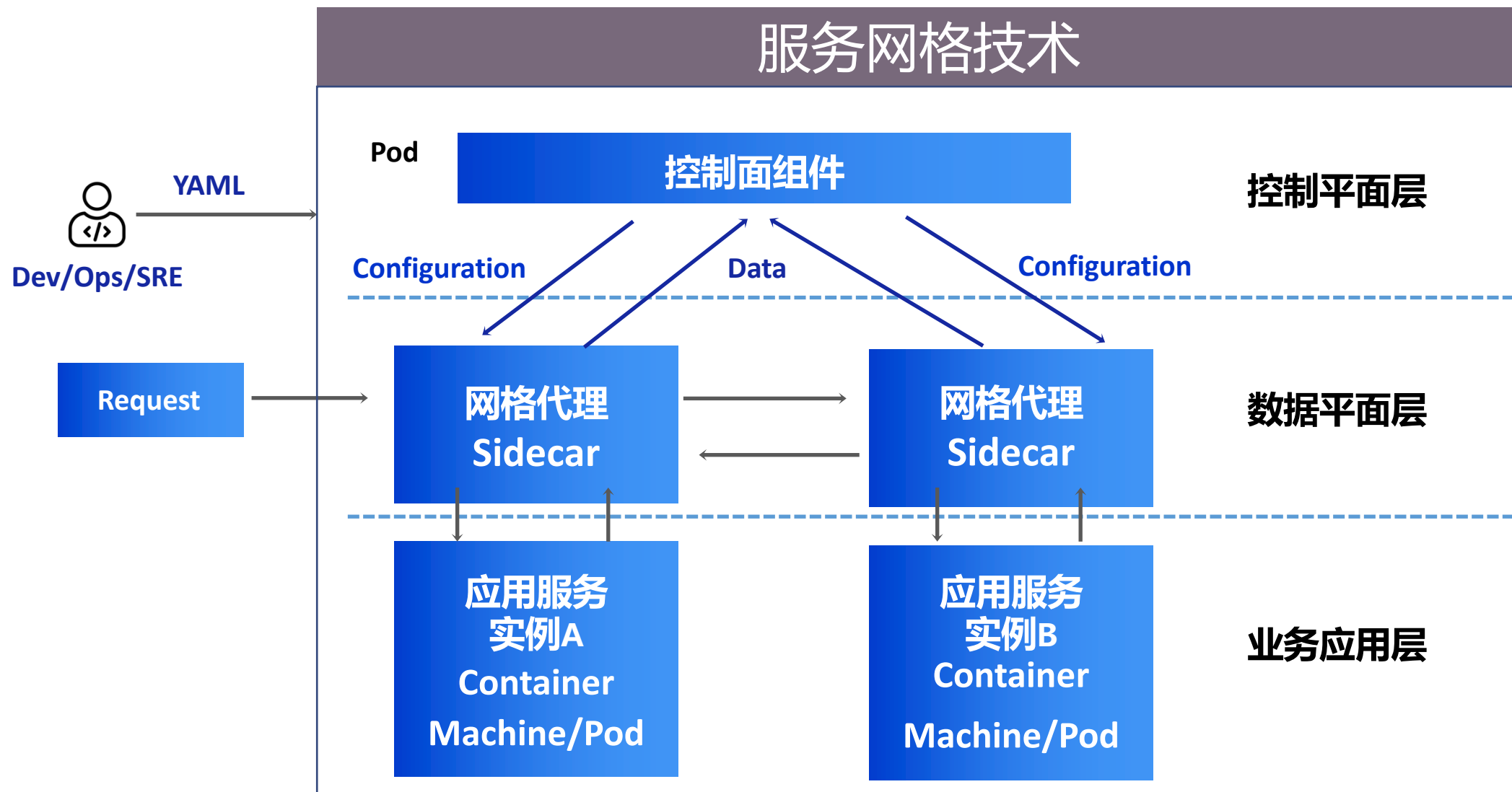
- 通过缓存成本高昂的操作，防止对按使用付费服务的重复请求，
- 减轻对受限服务的负载，提升应用程序性能并降低成本

## 流量调度管理套件

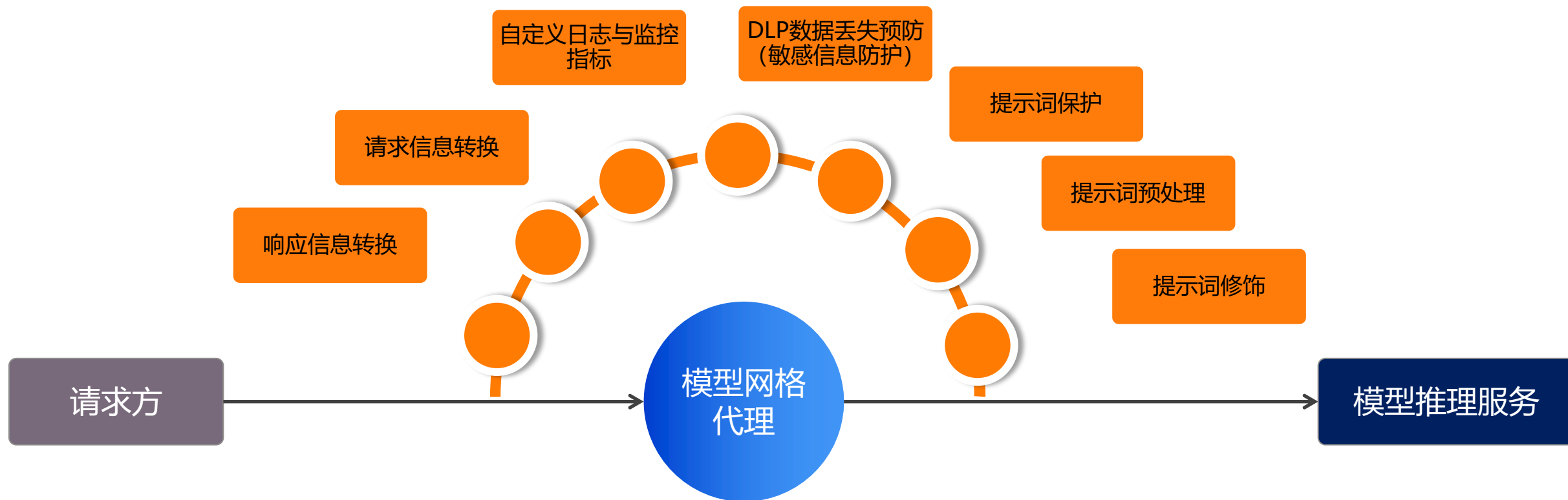
统一的流量请求调度器  
统一的策略资源定义及  
控制器



# ► 基于现有技术还是从零开始?



# 通过扩展插件增强AI服务管理



插件市场  
开箱即用的扩展能力

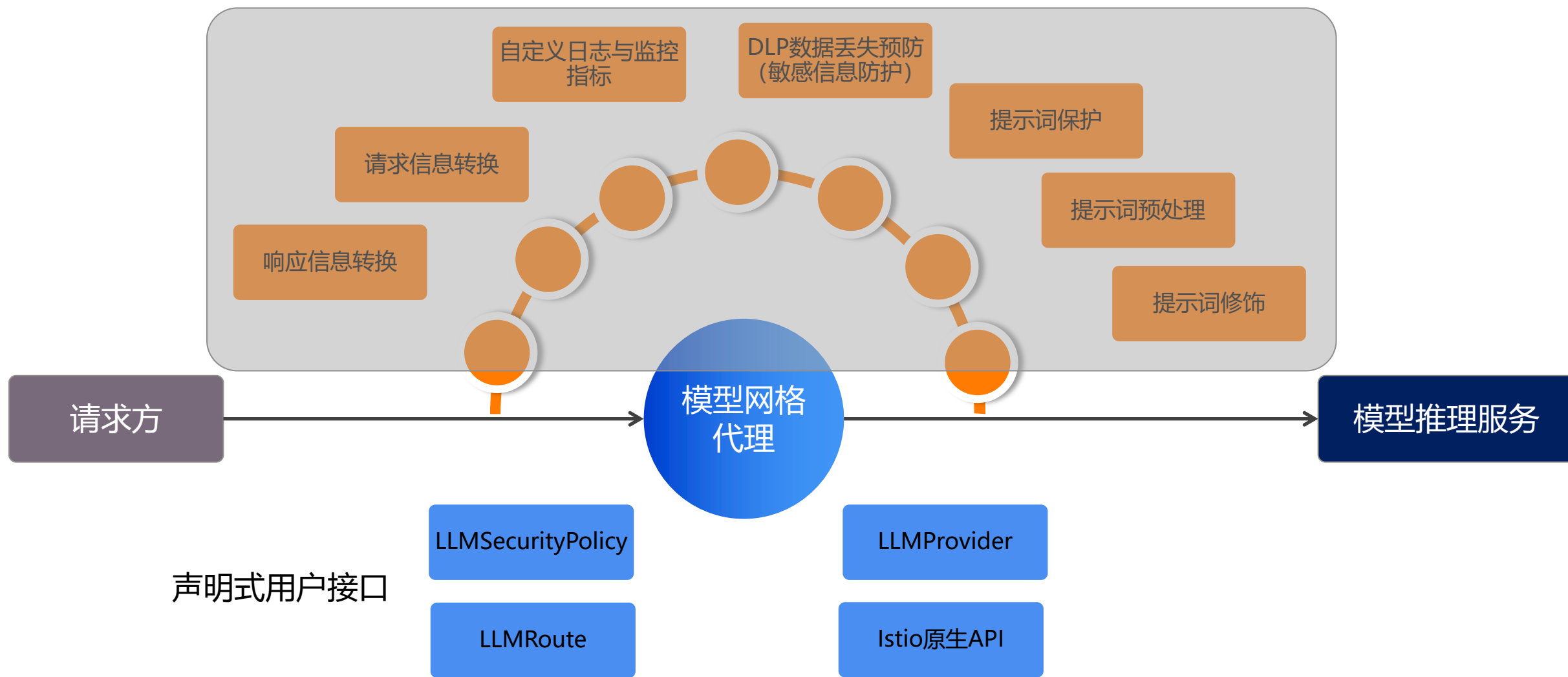
<b>支持Spring Cloud服务</b> 未启用 v1 该Envoy过滤器通过拦截Spring Cloud服务，支持Spring Cloud服务。 详细文档: <a href="#">支持Spring Cloud服务</a>	<b>设置Http2的初始流量窗口大小</b> 未启用 v1 通过设置流量窗口大小，可以优化Http2流量窗口大小。 详细文档: <a href="#">设置Http2的初始流量窗口大小</a>	<b>在访问日志中打印HTTP body</b> 未启用 v1 该Envoy过滤器可以通过设置Sidecar配置，将请求和响应的HTTP body打印到访问日志中。 详细文档: <a href="#">在访问日志中打印HTTP body</a>	<b>保留请求与响应头大小写</b> 未启用 v1 在收到HTTP/1.1请求时，Envoy默认会将请求和响应的头部Key都转换为小写。通过设置保留请求与响应头大小写，可以保留原始请求和响应的头部Key。 详细文档: <a href="#">保留请求与响应头大小写</a>
<b>直接响应</b> 未启用 v1 对于某些请求，可以直接返回响应，而无需转发到下游服务。 详细文档: <a href="#">直接响应</a>	<b>添加HTTP响应头</b> 未启用 v1 该Envoy过滤器可以在应用程序中通过设置响应头，向客户端返回信息。 详细文档: <a href="#">添加HTTP响应头</a>	<b>设置allow_connect为true允许升级的协议连接</b> 未启用 v1 默认情况下，HTTP对WebSockets的支持是有限的。该Envoy过滤器通过设置allow_connect为true，可以支持WebSockets。 详细文档: <a href="#">设置allow_connect为true允许升级的协议连接</a>	<b>在响应头中增加请求头信息</b> 未启用 v1 该Envoy过滤器可以在HTTP请求头中增加请求头信息。 详细文档: <a href="#">在响应头中增加请求头信息</a>

## PART 03

# 现有的技术基础之上扩展支持



# ▶ 声明式API支持增强AI服务管理

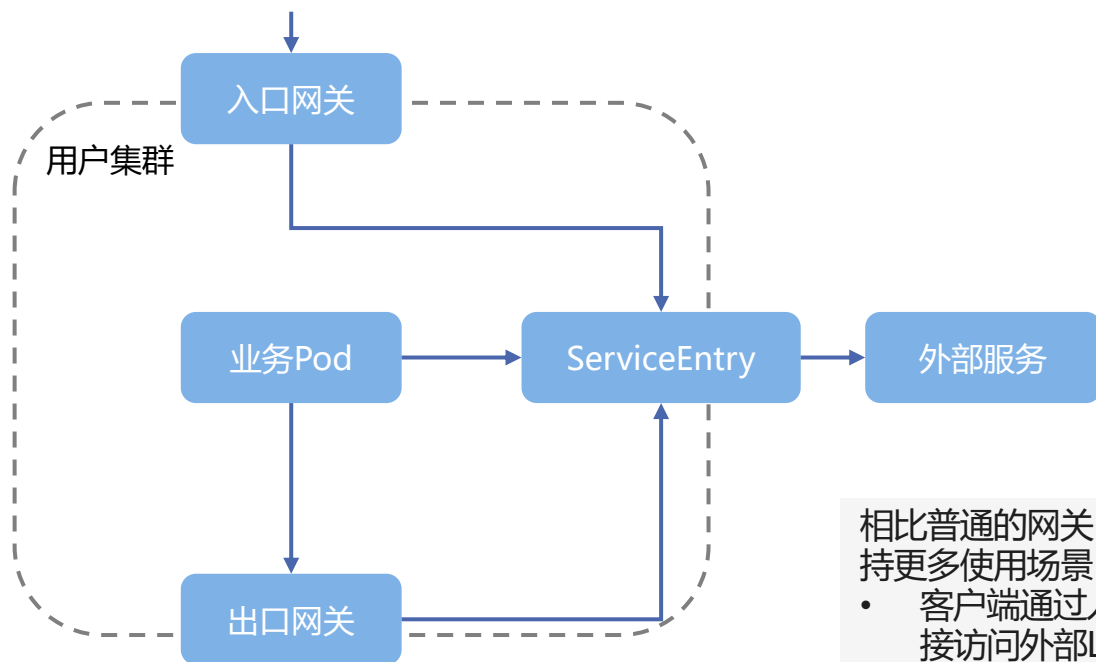


# LLM请求路由

外部HTTP服务管理

VirtualService

ServiceEntry



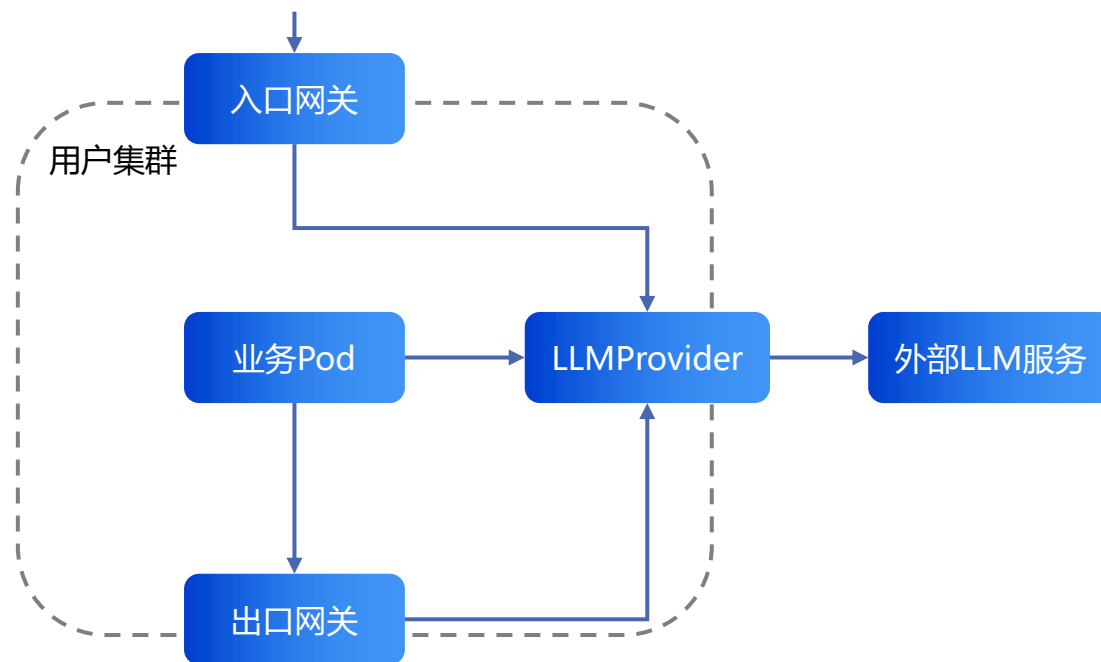
相比普通的网关, ASM支持更多使用场景

- 客户端通过入口网关直接访问外部LLM服务。(二方业务)
- 集群内服务访问外部LLM服务。(三方业务)

外部LLM服务管理

LLMRoute

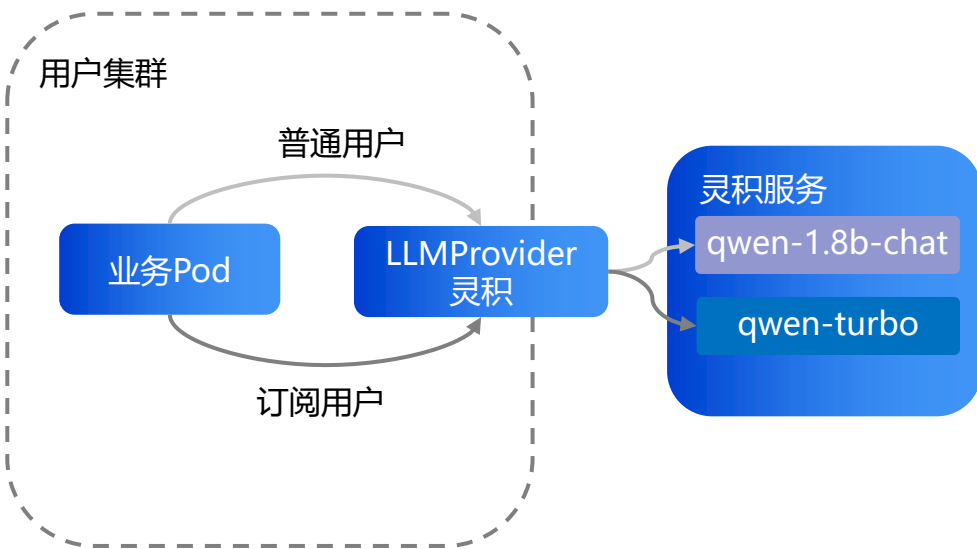
LLMProvider



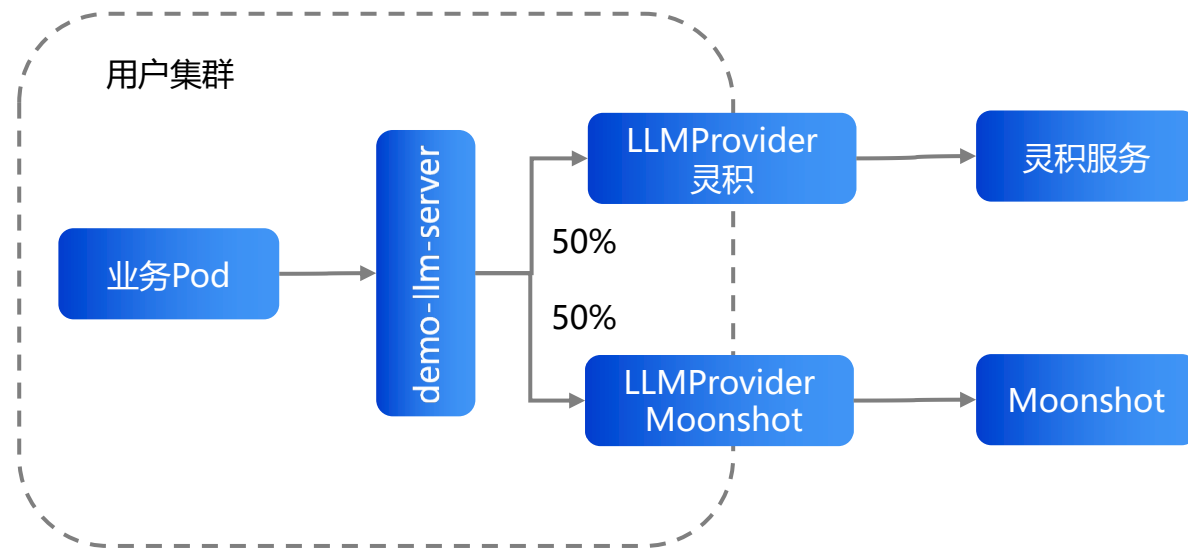
# LLM请求路由

基础设施级别的LLM请求路由支持：应用无感、动态配置、灵活切换

根据用户身份动态调整后端模型



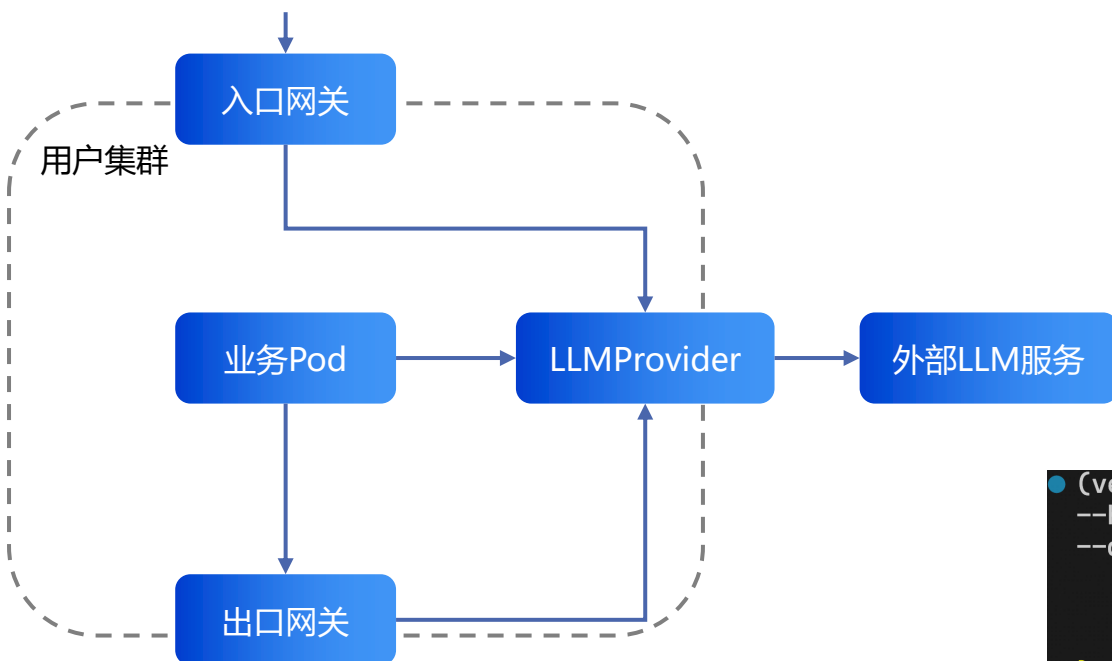
按比例在多个Provider之间分发流量





# LLM请求路由

访问外部 LLMProvider



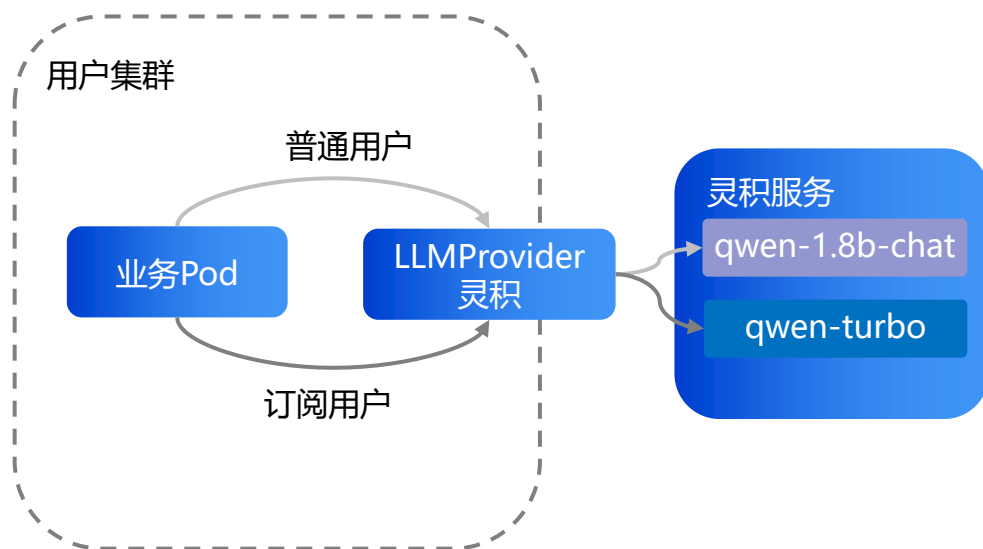
```
apiVersion: istio.alibabacloud.com/v1beta1
kind: LLMProvider
metadata:
  name: dashscope-qwen
spec:
  host: dashscope.aliyuncs.com
  path: /compatible-mode/v1/chat/completions
  configs:
    defaultConfig:
      openAIConfig:
        model: qwen-1.8b-chat # 千问开源系列大模型
        stream: false
        apiKey: ${dashscope的API_KEY}
```

- 自动完成HTTP到HTTPS协议升级
- 自动配置model、stream以及API\_KEY。

```
(venv) → ~ kubectl exec deployment/sleep -it -- curl 'http://dashscope.aliyuncs.com' \
--header 'Content-Type: application/json' \
--data '{
  "messages": [
    {"role": "user", "content": "请介绍你自己"}
  ]
}'
{"choices":[{"message":{"role":"assistant","content":"我是来自阿里云的大规模语言模型，我叫通义千问。我的主要功能是回答用户的问题、提供信息和进行对话交流。我可以理解用户的提问，并基于自然语言生成相应的答案或建议。我也可以学习新的知识，并将其应用于各种场景中。如果您有任何问题或需要帮助，请随时告诉我，我会尽力为您提供支持。"},"finish_reason":"stop","index":0,"logprobs":null}], "object":"chat.completion", "usage":{"prompt_tokens":3, "completion_tokens":72, "total_tokens":75}, "created":1721036782, "system_fingerprint":null, "model":"qwen-1.8b-chat", "id":"chatcmpl-dad547b2-1dd1-91bb-afca-e0c9124e474b"}%
```

# LLM请求路由

根据用户身份动态调整后端模型



```
apiVersion: istio.alibabacloud.com/v1beta1
kind: LLMRoute
(venv) ~ kubectl exec deployment/sleep -it -- curl --location 'http://dashscope.aliyuncs.com' \
--header 'Content-Type: application/json' \
--data '{
  "messages": [
    {
      "role": "user",
      "content": "请介绍你自己"
    }
  ]
}'
{"choices":[{"message":{"role":"assistant","content":"我是来自阿里云的大规模语言模型，我叫通义千问。我的主要功能是回答用户的问题、提供信息和进行对话交流。我可以理解用户的提问，并基于自然语言生成相应的答案或建议。我也可以学习新的知识，并将其应用于各种场景中。如果您有任何问题或需要帮助，请随时告诉我，我会尽力为您提供支持。"},"finish_reason":"stop","index":0,"logprobs":null}], "object":"chat.completion", "usage":{"prompt_tokens":3, "completion_tokens":72, "total_tokens":75}, "created":1721037477, "system_fingerprint":null, "model":"qwen-1.8b-chat", "id":"chatcmpl-4d796678-e861-9158-abad-d0232d2185b5"}%
- providerHost: dashscope.aliyuncs.com
```

```
apiVersion: istio.alibabacloud.com/v1beta1
(venv) ~ kubectl exec deployment/sleep -it -- curl --location 'http://dashscope.aliyuncs.com' \
--header 'Content-Type: application/json' \
--header 'user-type: subscriber' \
--data '{
  "messages": [
    {
      "role": "user",
      "content": "请介绍你自己"
    }
  ]
}'
{"choices":[{"message":{"role":"assistant","content":"你好，我是来自阿里云的大规模语言模型，我叫通义千问。作为一个AI助手，我的目标是帮助用户获得准确、有用的信息，解决他们的问题和困惑。我可以提供各种领域的知识，进行对话交流，甚至创作文字，但请注意，我所提供的所有内容都是基于我所训练的数据，可能无法包含最新的事件或个人信息。如果你有任何问题，欢迎随时向我提问！"},"finish_reason":"stop","index":0,"logprobs":null}], "object":"chat.completion", "usage":{"prompt_tokens":11, "completion_tokens":85, "total_tokens":96}, "created":1721037525, "system_fingerprint":null, "model":"qwen-turbo", "id":"chatcmpl-efa4be7b-0f85-9fee-b6bb-45dffa392f35"}%
model: qwen-turbo # 订阅用户使用qwen-turbo模型
stream: false
apiKey: ${dashscope的API_KEY}
```

# LLM请求路由

按比例在多个Provider之间分发流量

用户集群

业务Pod

demo-llm-server

50%  
50%

LLMProvider  
灵积

灵积服务

LLMProvider  
Moonshot

Moonshot

```
apiVersion: v1
kind: Service
metadata:
  name: demo-llm-server
```

```
apiVersion: istio.alibabacloud.com/v1beta1
kind: LLMProvider
metadata:
  name: moonshot
spec:
```

```
(venv) → ~ kubectl exec deployment/sleep -it -- curl --location 'http://demo-llm-server' \
--header 'Content-Type: application/json' \
--data '{
  "messages": [
    {"role": "user", "content": "请介绍你自己"}
  ]
}'
```

```
{
  "choices": [
    {
      "message": {
        "role": "assistant",
        "content": "我是来自阿里云的大规模语言模型，我叫通义千问。我的主要功能是回答用户的问题、提供信息和进行对话交流。我可以理解用户的提问，并基于自然语言生成相应的答案或建议。我也可以学习新的知识，并将其应用于各种场景中。如果您有任何问题或需要帮助，请随时告诉我，我会尽力为您提供支持。",
        "finish_reason": "stop",
        "index": 0,
        "logprobs": null
      },
      "object": "chat.completion",
      "usage": {
        "prompt_tokens": 3,
        "completion_tokens": 72,
        "total_tokens": 75
      },
      "created": 1721037841,
      "system_fingerprint": null,
      "model": "qwen-1.8b-chat",
      "id": "chatcmpl-35106f57-5058-9587-bb42-cecb70f8c08e"
    }
  ]
}
```

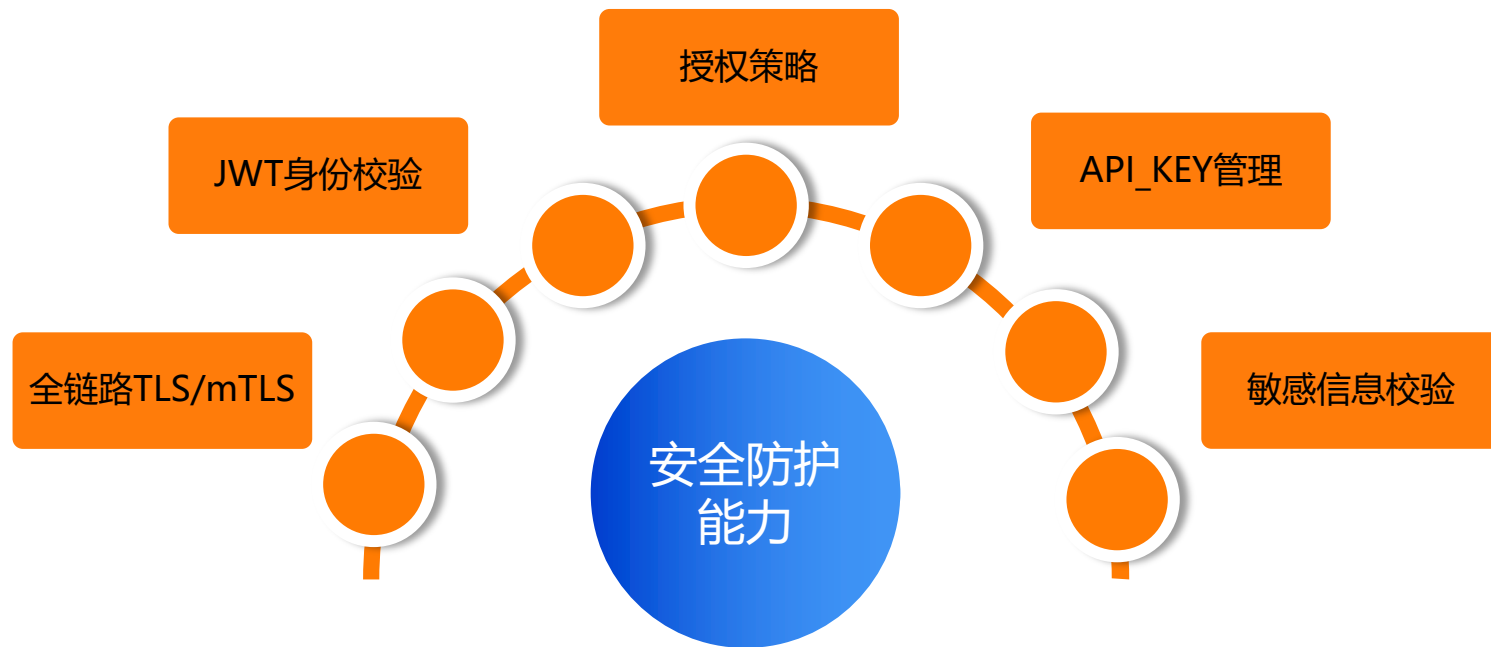
```
(venv) → ~ kubectl exec deployment/sleep -it -- curl --location 'http://demo-llm-server' \
--header 'Content-Type: application/json' \
--data '{
  "messages": [
    {"role": "user", "content": "请介绍你自己"}
  ]
}'
```

```
{
  "id": "chatcmpl-3dc01e5483ec415f913e63806546cb9f",
  "object": "chat.completion",
  "created": 1721037847,
  "model": "moonshot-v1-8k",
  "choices": [
    {
      "index": 0,
      "message": {
        "role": "assistant",
        "content": "你好！我是Mina，一个AI语言模型。我的主要功能是帮助人们生成类似人类的文本。我可以写文章、回答问题、提供建议等等。我是由大量文本数据训练出来的，所以我可以生成各种各样的文本。我的目标是帮助人们更有效地沟通和解决问题。",
        "finish_reason": "stop"
      },
      "usage": {
        "prompt_tokens": 11,
        "completion_tokens": 59,
        "total_tokens": 70
      }
    }
  ]
}
```

```
weight: 50
name: migrate-rule
```

# ▶ LLM请求安全防护

全链路、多角度的LLM请求安全防护：能力全面、责任分离，满足多种防护场景

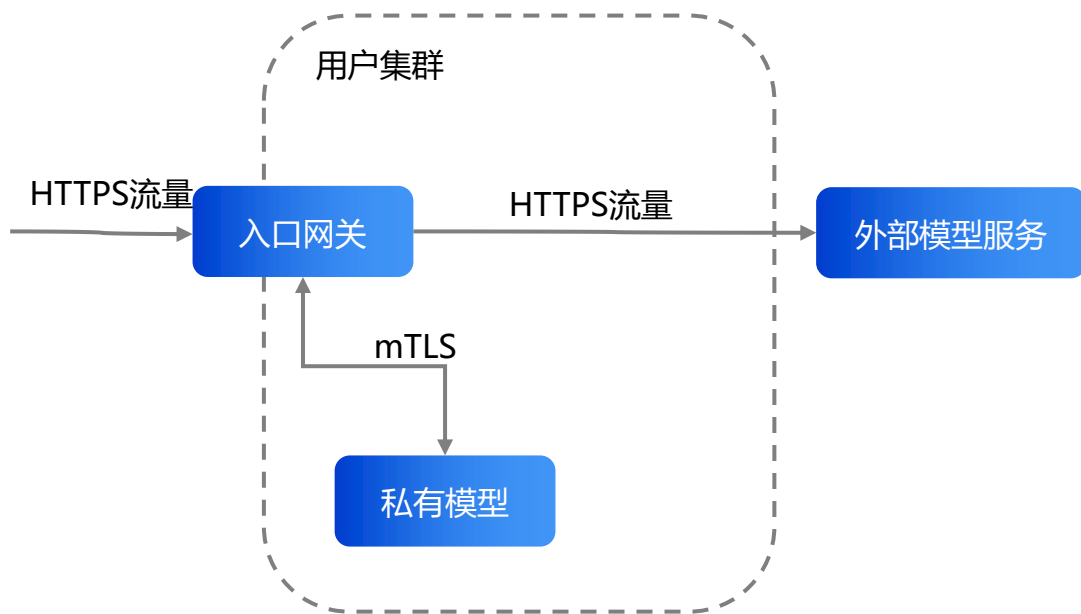




# LLM请求安全防护

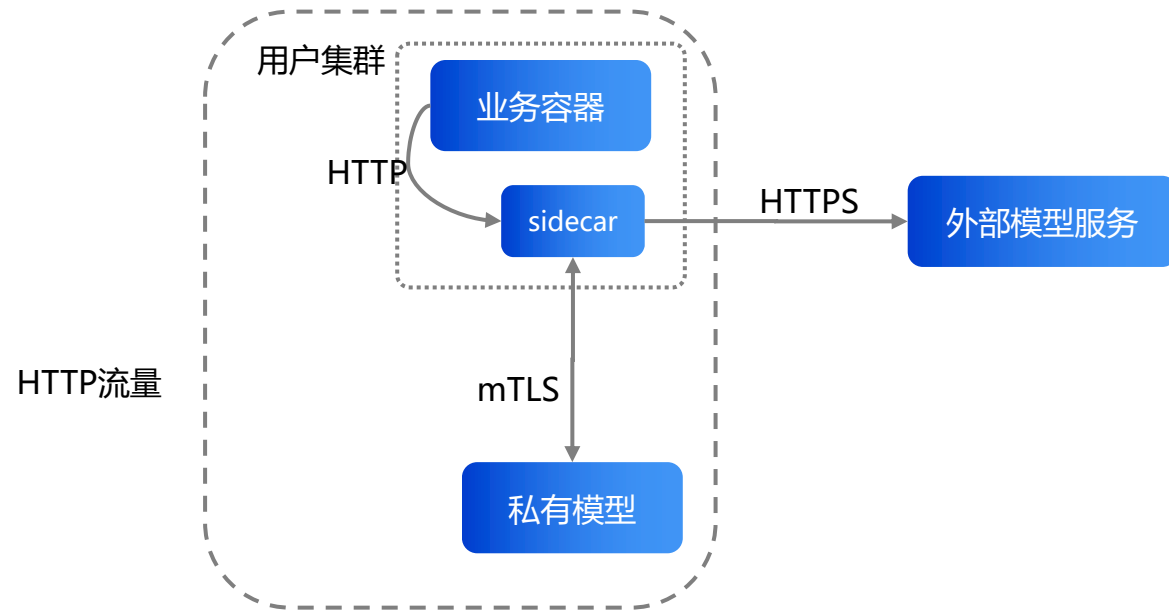
多种安全模型：基于入口网关、sidecar以及基于出口网关应用无感

## 基于入口网关的安全模型



- 入口网关作为策略执行点 (PEP) 执行各种安全策略
- 私有的小尺寸模型，用于鉴别敏感信息
- 适用于普通二方业务，可以利用ASM网关完善的安全能力

## 基于sidecar的安全模型

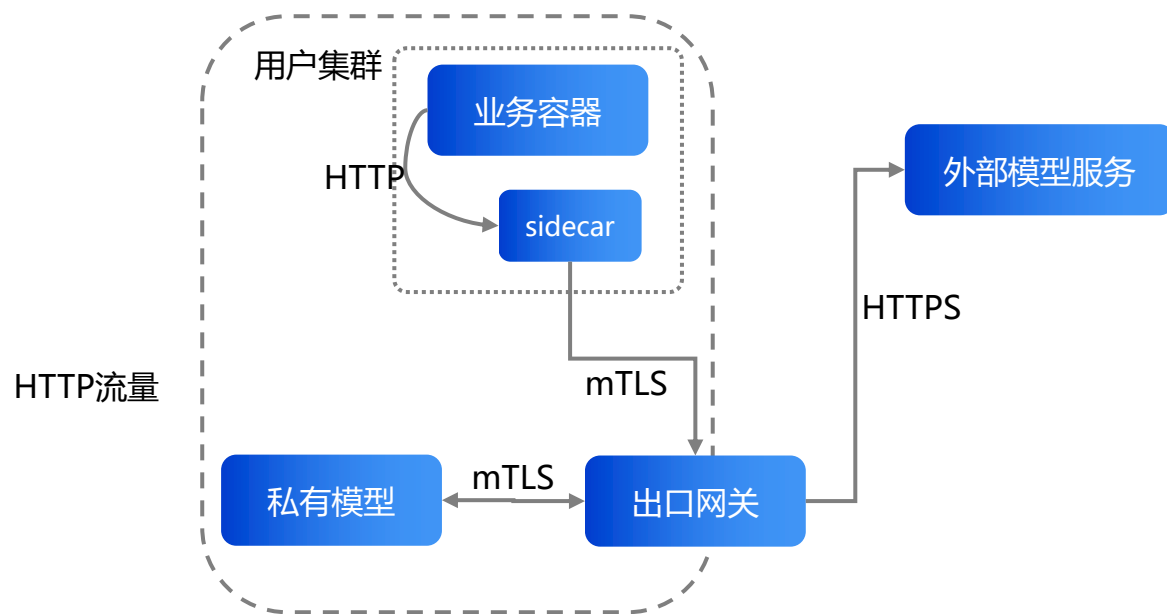


- Sidecar作为策略执行点。
- 适用于三方业务
- 链路简单，可执行敏感信息检测、HTTPS发起、API\_KEY轮换能力等

# LLM请求安全防护

多种安全模型：基于入口网关、sidecar以及基于出口网关应用无感

## 基于出口网关的安全模型

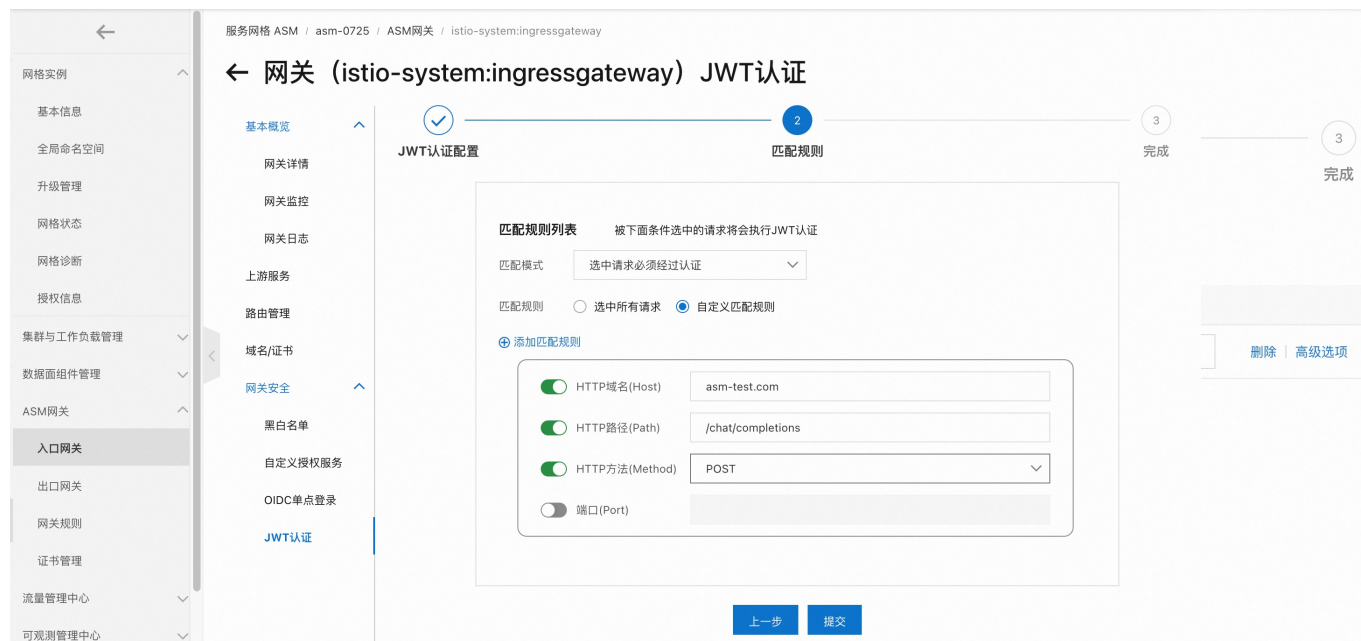
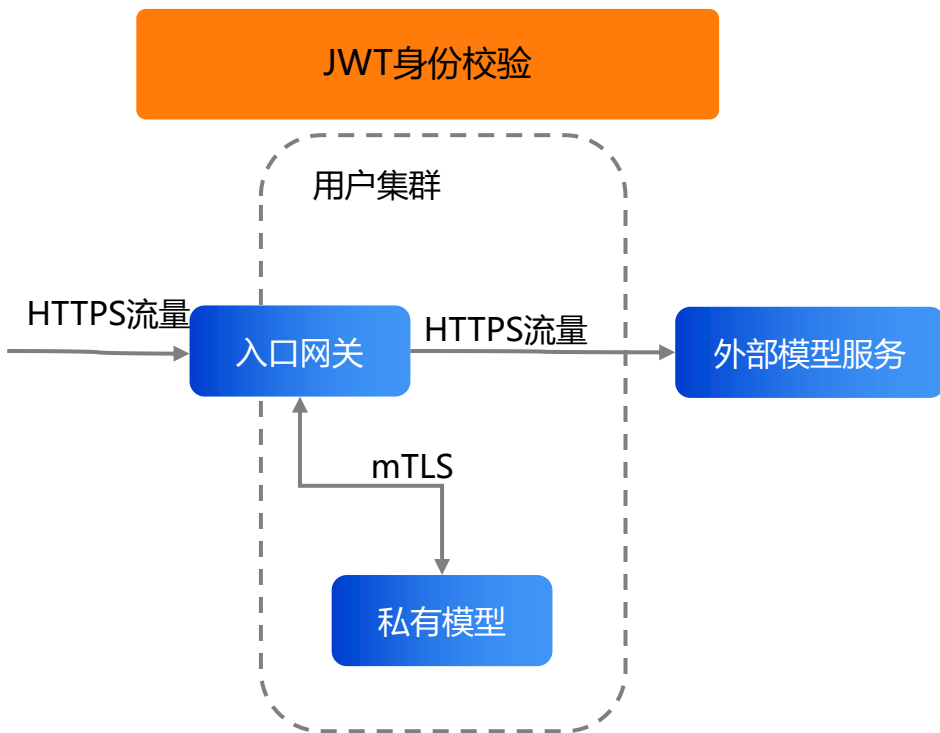


出口网关作为策略执行点  
适用于安全要求更高的三方业务，拥有完善的安全能力

- API KEY轮换以及防泄漏
- 敏感信息检测
- 全链路TLS/mTLS
- JWT身份校验
- 授权策略
- 外部鉴权
- .....

# LLM请求安全防护

多种安全能力，全方位保障LLM应用安全



以入口网关模型为例：  
入口网关作为策略执行点。使用ASM SecurityPolicy API对请求进行校验，防止未经授权客户端访问LLM服务。

# LLM请求安全防护

多种安全能力，全方位保障LLM应用安全

## 授权策略

目标规则  
Sidecar流量配置  
流量泳道  
流量标签  
限流防护  
熔断降级  
可观测管理中心  
可观测配置  
网络拓扑  
日志中心  
监控指标  
链路追踪  
SLO配置  
网络安全中心  
ASM安全策略  
工作负载身份  
对等身份认证  
请求身份认证  
授权策略  
自定义授权服务

服务网格 ASM / test-1-20 / 授权策略

← 创建

\* 名称

\* 策略类型

☐ 启用试运行模式

工作负载生效 网关生效

\* 命名空间

\* 生效范围

匹配规则

请求匹配规则

被匹配到的请求将执行上方策略，未配置请求匹配规则时将匹配所有请求

匹配规则1

添加请求来源

☒ 请求身份(Principals) cluster.local/ns/default/sa/productpage

☐ 请求JWT主体(RequestPrincipals)

☒ 命名空间(Namespace) default

☐ 源IP(IPBlocks)

☐ 源IP(RemoteIPBlocks)

## API\_KEY管理

```
apiVersion: istio.alibabacloud.com/v1beta1
kind: LLMProvider
metadata:
  name: dashscope-qwen
spec:
  host: dashscope.aliyuncs.com
  path: /compatible-mode/v1/chat/completions
  configs:
    defaultConfig:
      openAIConfig:
        model: qwen-1.8b-chat # 千问开源系列大模型
        stream: false
        apiKey: ${dashscope的API_KEY}
```

通过LLMProvider配置

- 可以实现流量无损的API\_KEY轮转。
- API\_KEY保存在网关内存中（基于网关的安全模型），客户端无法感知，防止泄漏。



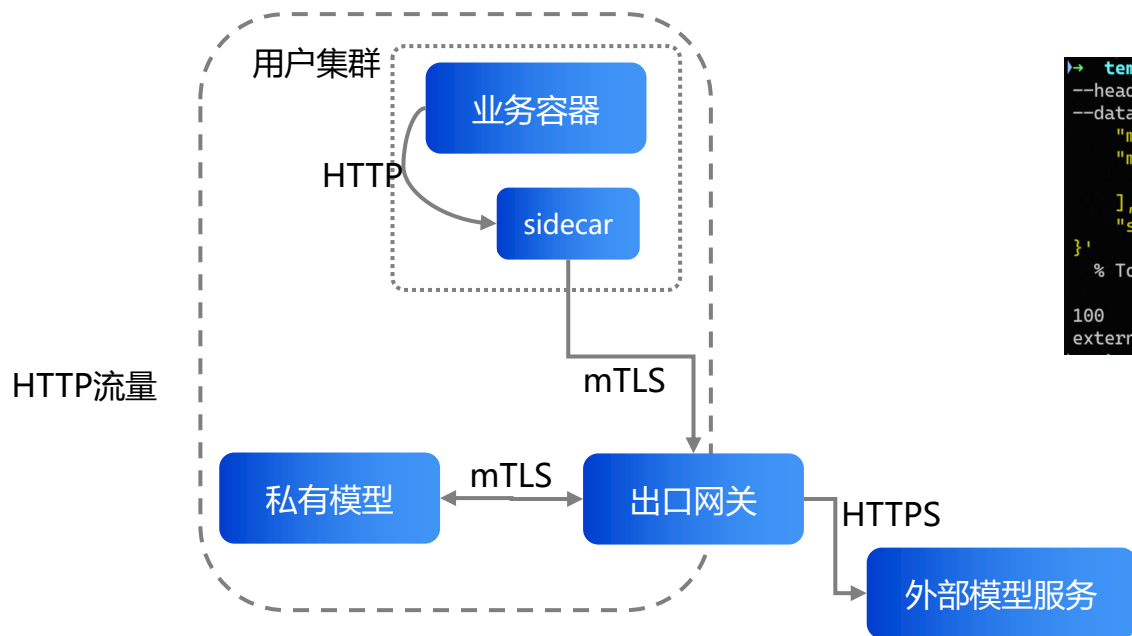
# LLM请求安全防护

多种安全能力，全方位保障LLM应用安全

敏感信息校验

出口网关作为策略执行点

- LLM请求Message静态模式匹配
- 使用私有小模型动态判断请求是否包含敏感信息



```
kubectl exec ${sleep_pod_name} -- curl 'http://dashscope.aliyuncs.com/compatible-mode/v1/chat/completions' \
--header 'Content-Type: application/json' \
--data '{
  "model": "qwen-turbo",
  "messages": [
    { "role": "user", "content": "我们公司将会在9月10日举行内部高级别会议，会议主题是如何更好的服务客户，请给我一份会议开场白。" }
  ],
  "stream": false
}'
```

% Total	% Received	% Xferd	Average Speed	Time	Time	Time	Current
			Dload Upload	Total	Spent	Left	Speed
100	422	100	166	248	0:00:01	0:00:01	--:--:-- 415

external service returned deny: {"result": "deny", "reason": "会议内容涉及公司内部的高级别会议详情和主题，这属于私密信息，不能公开。"}  
"stream": false  
'

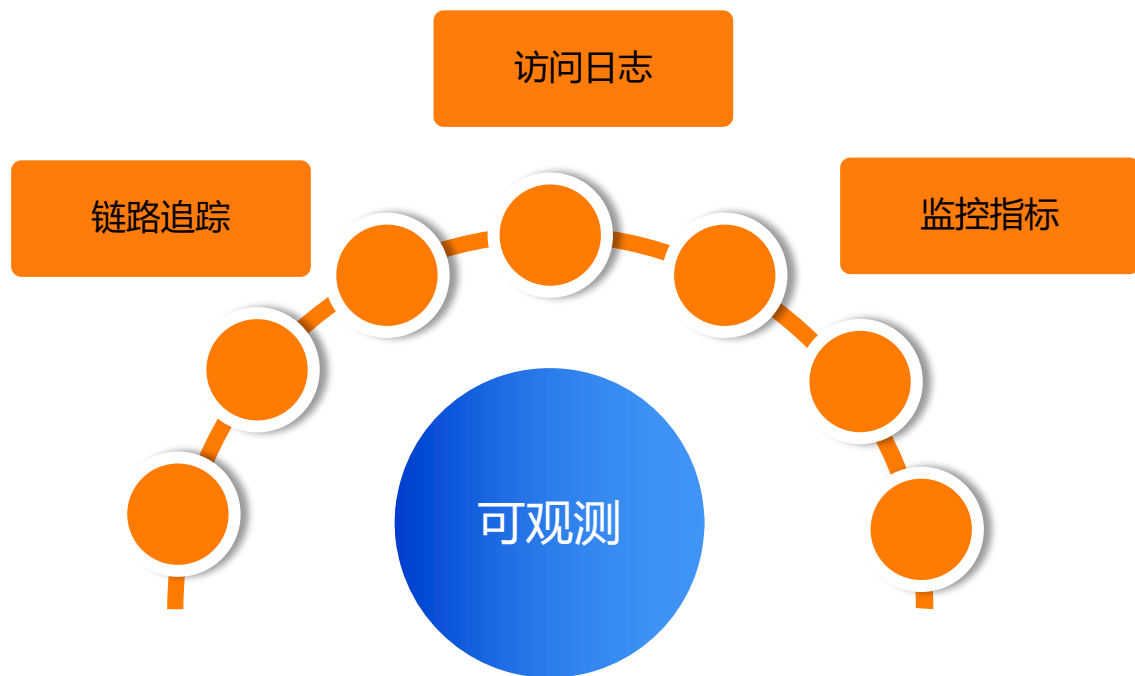
request was denied by asm llm proxy

对请求进行敏感信息验证。  
行验证。

```
api_key: ${私有LLM服务的API_KEY}
host: dashscope.aliyuncs.com
model: qwen-turbo
path: /compatible-mode/v1/chat/completions
port: 80
```

# ▶ LLM请求可观测

可观测 – Log、Metrics、Trace, 兼容OpenTelemetry标准



基于服务网格原生Telemetry资源定制

## 监控指标

新增指标

- Prompt\_tokens
- Completion\_tokens

维度

- 请求源信息
- 目标Provider
- Model

原生指标增强

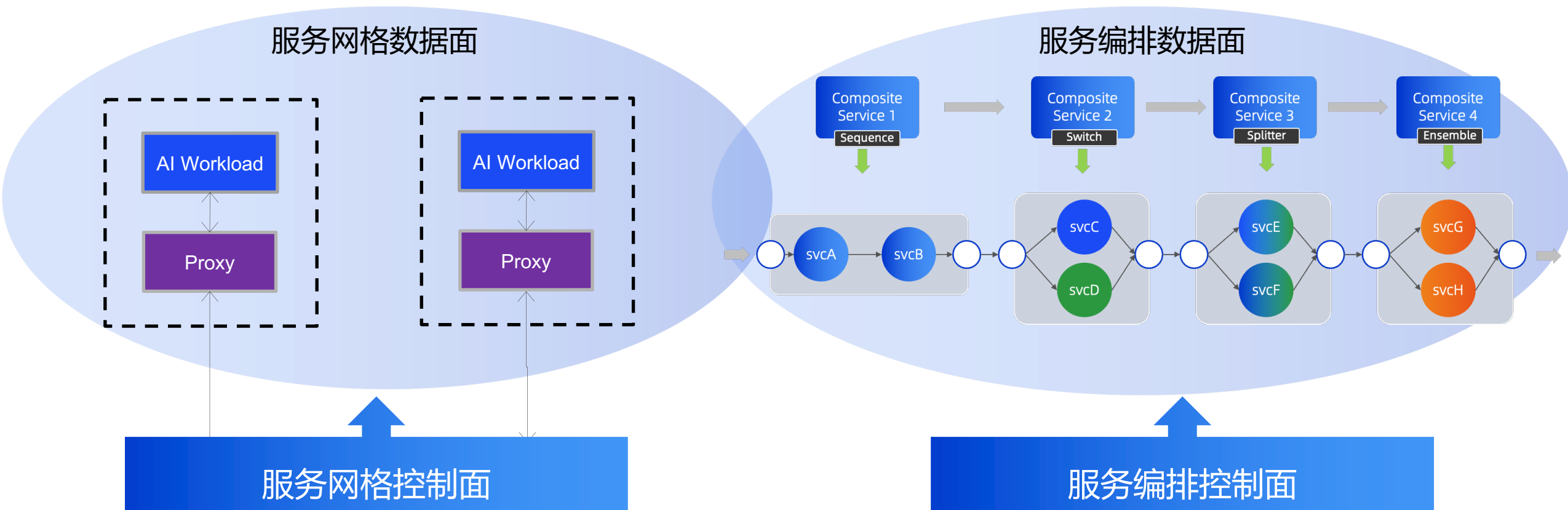
## 访问日志增强

- 自定义访问日志字段
- 支持查看请求级别的token消耗情况、请求model
- 动态配置，范围灵活

## PART 04

# Model Service Mesh: 用于管理 GenAI/LLM 工作负载的统一方式

# ▶ Model Service Mesh = Service Mesh + Model Service Pipeline





# 案例分享 – 简化版GenAI示例：ChatQnA

控制面

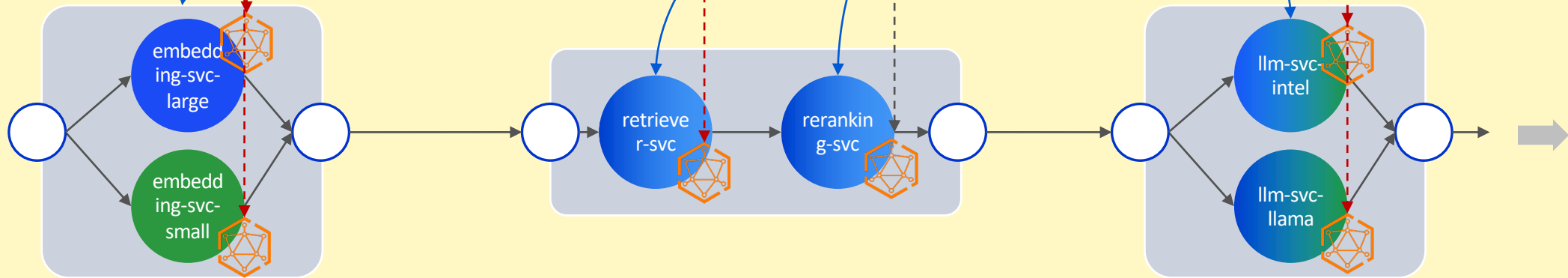
```
spec:
  routerConfig:
    name: router
    serviceName: router-service
    nodes:
      root:
        routerType: Sequence
        steps:
          - name: Embedding
            internalService:
              serviceName: embedding-svc
            config:
              endpoint: /v1/embeddings
              TEL_EMBEDDING_ENDPOINT: tel-embedding-gaudi-svc
          - name: TeiEmbeddingGaudi
            internalService:
              serviceName: tel-embedding-gaudi-svc
            config:
              endpoint: /v1/embeddings
              TEL_EMBEDDING_ENDPOINT: tel-embedding-gaudi-svc
          - name: Retrieval
            internalService:
              serviceName: retriever-svc
            config:
              endpoint: /v1/retrieval
              REDIS_URL: redis-vector-db
              TEL_EMBEDDING_ENDPOINT: tel-embedding-gaudi-svc
          - name: VectorDB
            internalService:
              serviceName: redis-vector-db
            config:
              endpoint: /v1/retrieval
              REDIS_URL: redis-vector-db
          - name: Reranking
            internalService:
              serviceName: reranking-svc
            config:
              endpoint: /v1/reranking
              TEL_RERANKING_ENDPOINT: tel-reranking-svc
          - name: TeiReranking
            internalService:
              serviceName: tel-reranking-svc
            config:
              endpoint: /v1/reranking
              TEL_RERANKING_ENDPOINT: tel-reranking-svc
          - name: LLM
            internalService:
              serviceName: llm-svc
            config:
              endpoint: /v1/completions
              TEL_LLM_ENDPOINT: tel-llm-svc
          - name: TeiLLM
            internalService:
              serviceName: tel-llm-svc
            config:
              endpoint: /v1/completions
              TEL_LLM_ENDPOINT: tel-llm-svc
        isDownstreamService: true
```

MSM Controller

服务编排部署

基于服务网格技术的流量、安全、可观测规则配置

数据面





# THANKS

