KCD HANGZHOU

CHINA OpenInfra Days

从 Pods 到 Prompts：

Kubernetes 网络演进与 AI 新时代

张晋涛

Kong Inc.

2025/11/15

张晋涛

Kong Inc.

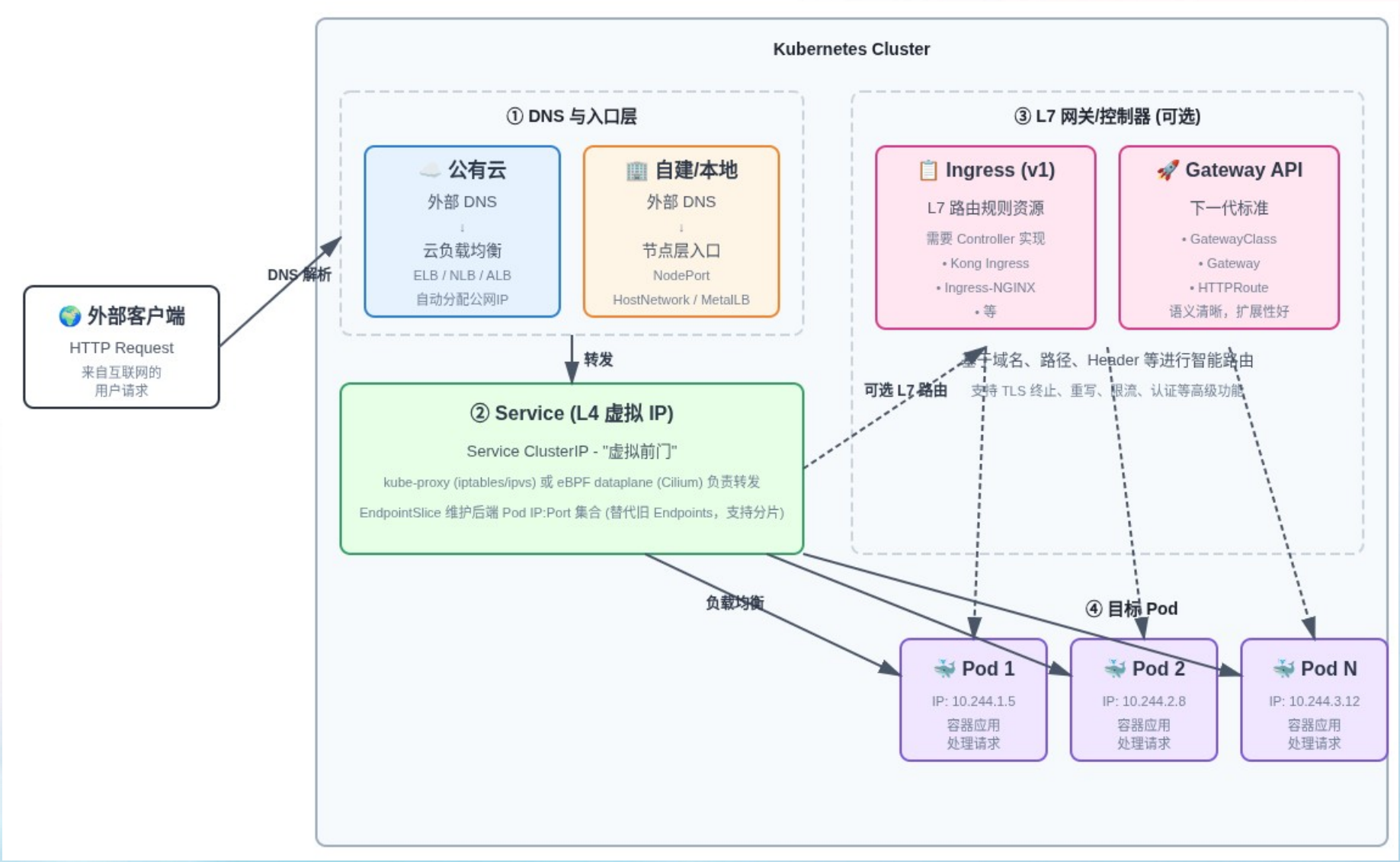
- Kubernetes Ingress-NGINX maintainer
- Microsoft MVP
- CNCF Ambassador
- LFAPAC Open Source Evangelist
- 公众号： MoeLove

KCD HANGZHOU

CHINA OpenInfra Days

CONTENT
目录

Kubernetes 的网络模型


Kubernetes Cluster
① DNS 与入口层
☁️ 公有云
外部 DNS
云负载均衡
ELB / NLB / ALB
自动分配公网IP
🏢 自建/本地
外部 DNS
节点层入口
NodePort
HostNetwork / MetalLB
③ L7 网关/控制器 (可选)
📋 Ingress (v1)
L7 路由规则资源
需要 Controller 实现
• Kong Ingress
• Ingress-NGINX
• 等
🚀 Gateway API
下一代标准
• GatewayClass
• Gateway
• HTTPRoute
语义清晰，扩展性好
基于域名、路径、Header 等进行智能路由
支持 TLS 终止、重写、限流、认证等高级功能
可选 L7 路由
DNS 解析
🌍 外部客户端
HTTP Request
来自互联网的
用户请求
转发
② Service (L4 虚拟 IP)
Service ClusterIP - "虚拟前门"
kube-proxy (iptables/ipvs) 或 eBPF dataplane (Cilium) 负责转发
EndpointSlice 维护后端 Pod IP:Port 集合 (替代旧 Endpoints，支持分片)
负载均衡
④ 目标 Pod
🐳 Pod 1
IP: 10.244.1.5
容器应用
处理请求
🐳 Pod 2
IP: 10.244.2.8
容器应用
处理请求
🐳 Pod N
IP: 10.244.3.12
容器应用
处理请求

KCD HANGZHOU
GROWING CLOUD NATIVE TOGETHER

CHINA OpenInfra Days

从 Ingress 到 Gateway API

Ingress NGINX Retirement: What You Need to Know

By **Tabitha Sable (Kubernetes SRC)** | Tuesday, November 11, 2025

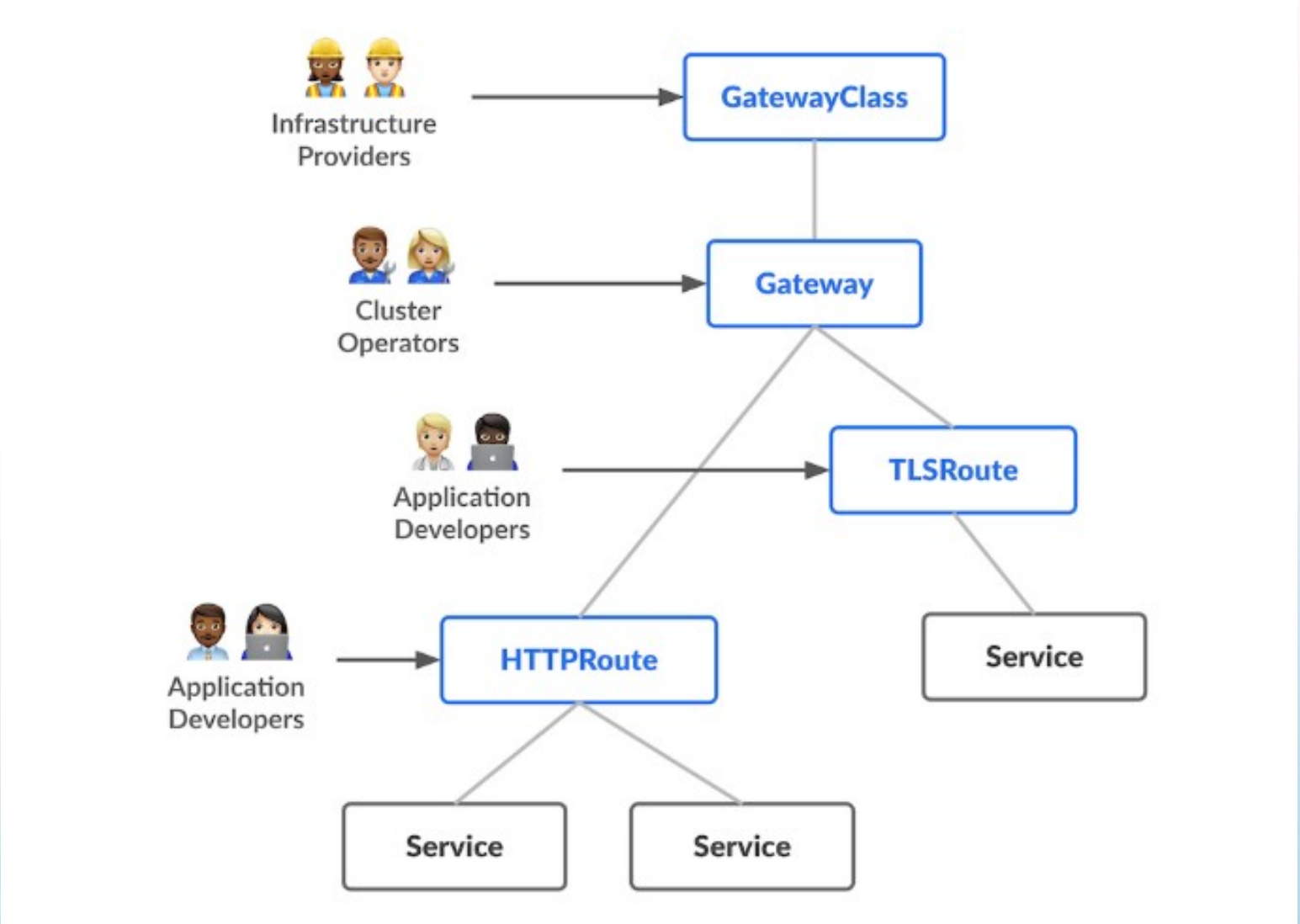
To prioritize the safety and security of the ecosystem, Kubernetes SIG Network and the Security Response Committee are announcing the upcoming retirement of Ingress NGINX. Best-effort maintenance will continue until March 2026. Afterward, there will be no further releases, no bugfixes, and no updates to resolve any security vulnerabilities that may be discovered. **Existing deployments of Ingress NGINX will continue to function and installation artifacts will remain available.**

We recommend migrating to one of the many alternatives. Consider migrating to Gateway API, the modern replacement for Ingress. If you must continue using Ingress, many alternative Ingress controllers are listed in the Kubernetes documentation. Continue reading for further information about the history and current state of Ingress NGINX, as well as next steps.

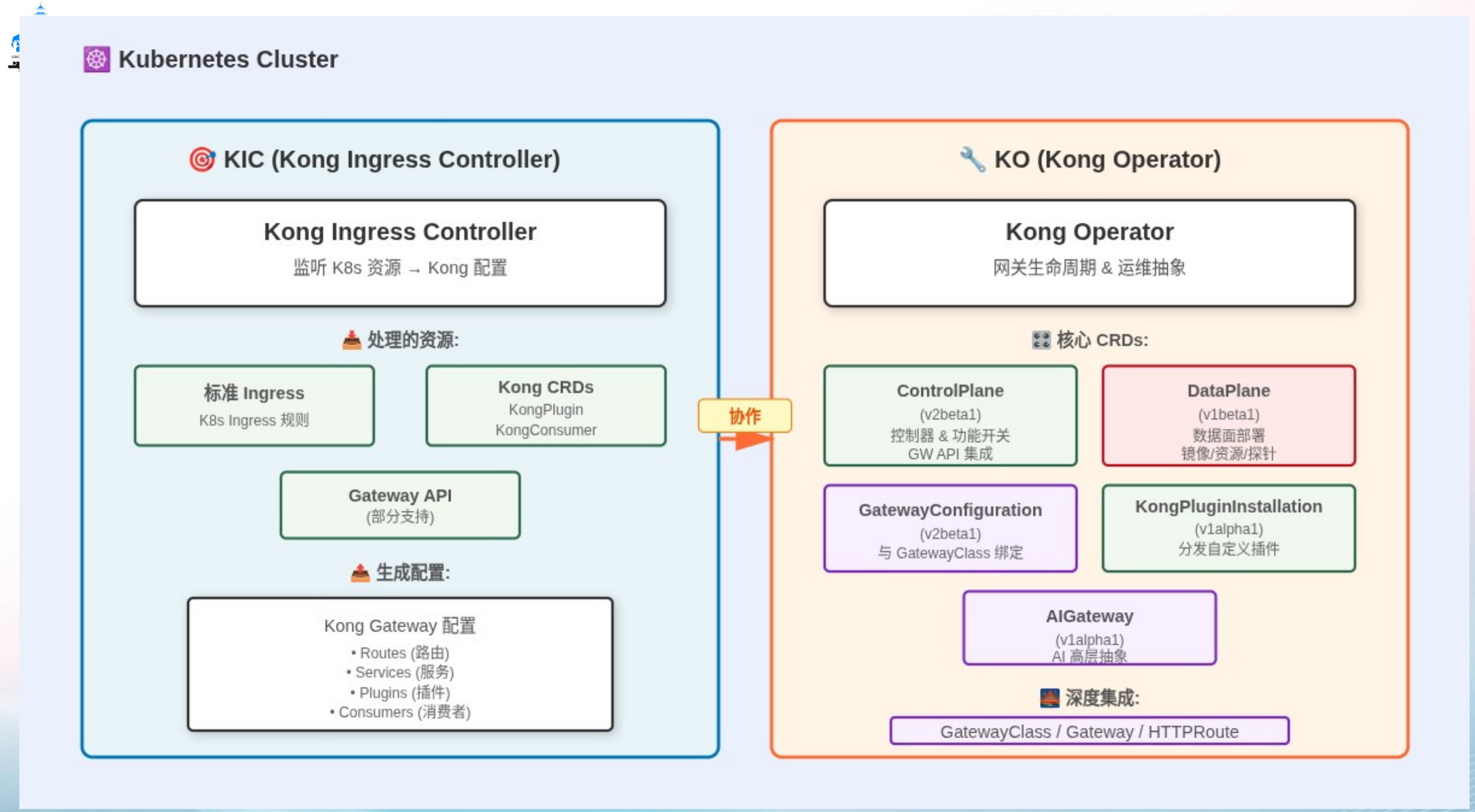
About Ingress NGINX

Ingress is the original user-friendly way to direct network traffic to workloads running on Kubernetes. (Gateway API is a newer way to achieve many of the same goals.) In order for an Ingress to work in your cluster, there must be an Ingress controller running. There are many Ingress controller choices available, which serve the needs of different users and use cases. Some are cloud-provider specific, while others have more general applicability.

Ingress NGINX was an Ingress controller, developed early in the history of the Kubernetes project as an example implementation of the API. It became very popular due to its tremendous flexibility, breadth of features, and independence from any particular cloud or infrastructure provider. Since those days, many other Ingress controllers have been created within the Kubernetes project by community groups, and by cloud native vendors. Ingress NGINX has continued to be one of the most popular, deployed as part of many hosted Kubernetes platforms and within innumerable independent users' clusters.


Infrastructure Providers
GatewayClass
Cluster Operators
Gateway
Application Developers
TLSRoute
Application Developers
HTTPRoute
Service
Service
Service


kind: HTTPRoute
apiVersion:gateway.networking.k8s.io/v1beta1
metadata:
name: orders-httproute
spec:
parentRefs:
- group: gateway.networking.k8s.io
kind: Gateway
name: kong-gateway
hostnames:
- orders.example.com
rules:
- matches:
- path:
type: PathPrefix
value: /orders
backendRefs:
- name: orders
port: 8000


Kubernetes Cluster
KIC (Kong Ingress Controller)
Kong Ingress Controller
监听 K8s 资源 → Kong 配置
处理的资源:
标准 Ingress
K8s Ingress 规则
Kong CRDs
KongPlugin
KongConsumer
Gateway API
(部分支持)
生成配置:
Kong Gateway 配置
• Routes (路由)
• Services (服务)
• Plugins (插件)
• Consumers (消费者)
协作
KO (Kong Operator)
Kong Operator
网关生命周期 & 运维抽象
核心 CRDs:
ControlPlane
(v2beta1)
控制器 & 功能开关
GW API 集成
DataPlane
(v1beta1)
数据面部署
镜像/资源/探针
GatewayConfiguration
(v2beta1)
与 GatewayClass 绑定
KongPluginInstallation
(v1alpha1)
分发自定义插件
AIGateway
(v1alpha1)
AI 高层抽象
深度集成:
GatewayClass / Gateway / HTTPRoute

AI 时代与 Kong AI Gateway

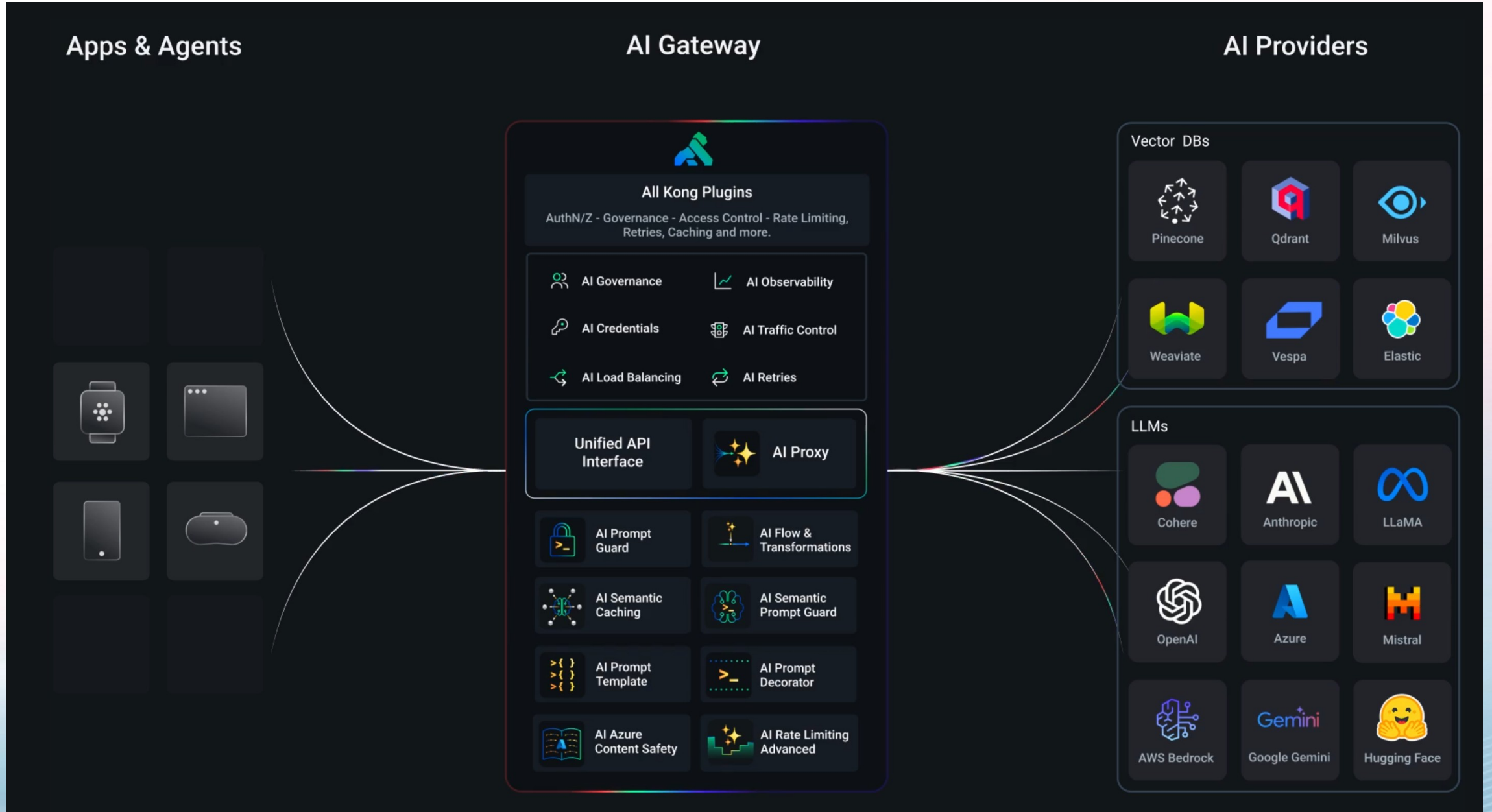
从 API 到 AI：新的挑战

传统 API 时代

- → 治理单位：API 请求
- → 计量单位：请求数量 (QPS)
- → 成本模型：按请求计费
- → 统一接口：RESTful / GraphQL
- → 安全重点：认证授权、参数校验

🆕 AI 时代

- → 治理单位：LLM 调用
- → 计量单位：Token / 上下文成本 / 模型类型
- → 成本模型：Token 计费（输入+输出）
- → 多厂商接口：各有差异需归一化
- → 安全重点：Prompt 注入、内容安全、数据泄露
- → 新需求：配额管理、成本预算、语义缓存


Apps & Agents
AI Gateway
AI Providers
All Kong Plugins
AuthN/Z - Governance - Access Control - Rate Limiting, Retries, Caching and more.
AI Governance
AI Observability
AI Credentials
AI Traffic Control
AI Load Balancing
AI Retries
Unified API Interface
AI Proxy
AI Prompt Guard
AI Flow & Transformations
AI Semantic Caching
AI Semantic Prompt Guard
AI Prompt Template
AI Prompt Decorator
AI Azure Content Safety
AI Rate Limiting Advanced
Vector DBs
Pinecone
Qdrant
Milvus
Weaviate
Vespa
Elastic
LLMs
Cohere
Anthropic
LLaMA
OpenAI
Azure
Mistral
AWS Bedrock
Google Gemini
Hugging Face

统一 LLM 接入

将不同厂商 API 归一化处理，支持统一路由配置、模型提供商切换

插件：ai-proxy, ai-proxy-advanced

Token 维度治理

按 Token 数量限速、响应成本控制、多时间窗口策略、Redis 分布式支持

插件：ai-rate-limiting-advanced

Prompt 安全与治理

防止 Prompt 注入、内容安全检测、模板管理、请求装饰与增强

插件：ai-prompt-guard, ai-prompt-decorator, ai-prompt-template

请求/响应增强

请求/响应转换、语义缓存、响应内容安全检测、智能降级

插件：ai-request-transformer, ai-response-transformer, ai-semantic-cache

可观测性与成本核算

LLM 调用统计、Token 使用追踪、成本核算报表、异常告警

结合网关指标、日志、Analytics 平台

认证、授权与配额

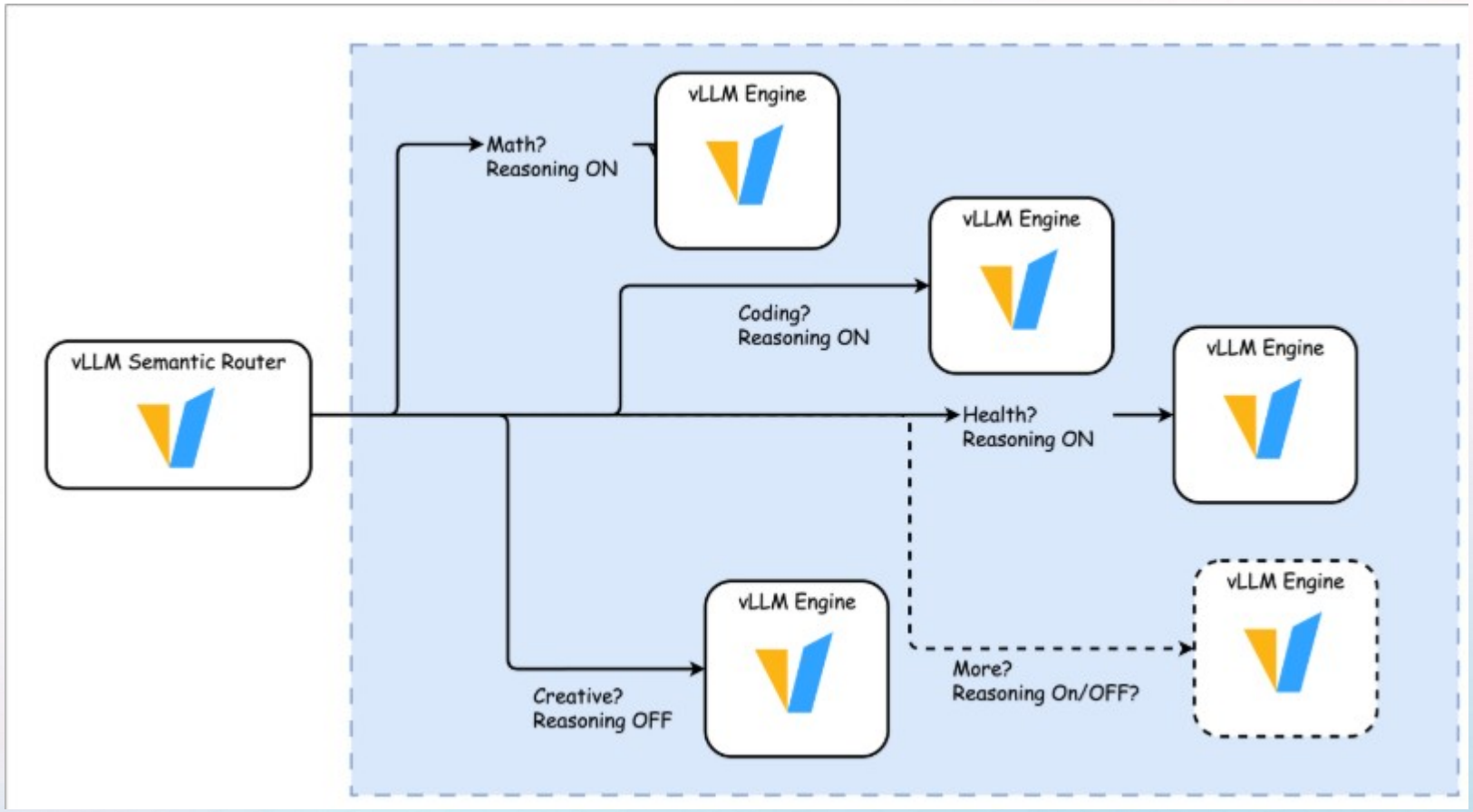
统一身份认证、细粒度授权、配额管理、多租户隔离

插件：key-auth, oauth2, acl, quota-management

KCD HANGZHOU
GROWING CLOUD NATIVE TOGETHER

CHINA OpenInfra Days

AI + Cloud Native


vLLM Semantic Router
Math?
Reasoning ON
vLLM Engine
Coding?
Reasoning ON
vLLM Engine
Health?
Reasoning ON
vLLM Engine
Creative?
Reasoning OFF
vLLM Engine
More?
Reasoning On/OFF?
vLLM Engine

Kong Konnect

KCD HANGZHOU

CHINA OpenInfra Days