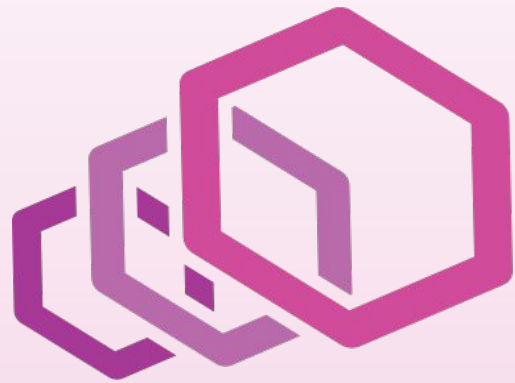




envoycon
EUROPE

CLOUD NATIVE THEATER

The Future of AI Traffic: What's New in Envoy AI Gateway 2026

Xiaolin Lin
Bloomberg



The Problems We're Solving

- Traditional API Gateways Fall Short for LLM Traffic
- Multi-Provider Integration Complexity
- Operational & Governance Gaps
- Lack of Open, Collaborative Solutions

What's Envoy AI Gateway?



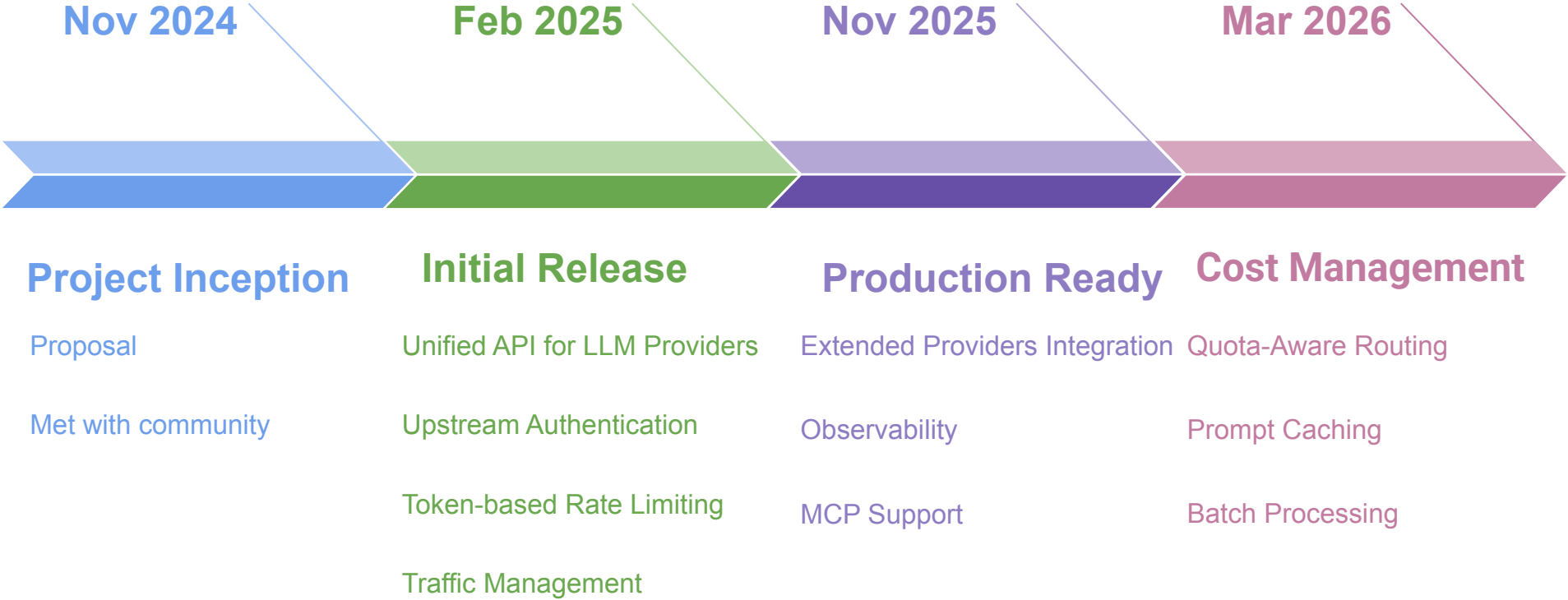
Envoy AI Gateway is an open source project for using Envoy Gateway to handle request traffic from application clients to GenAI services.

[Get Started](#)

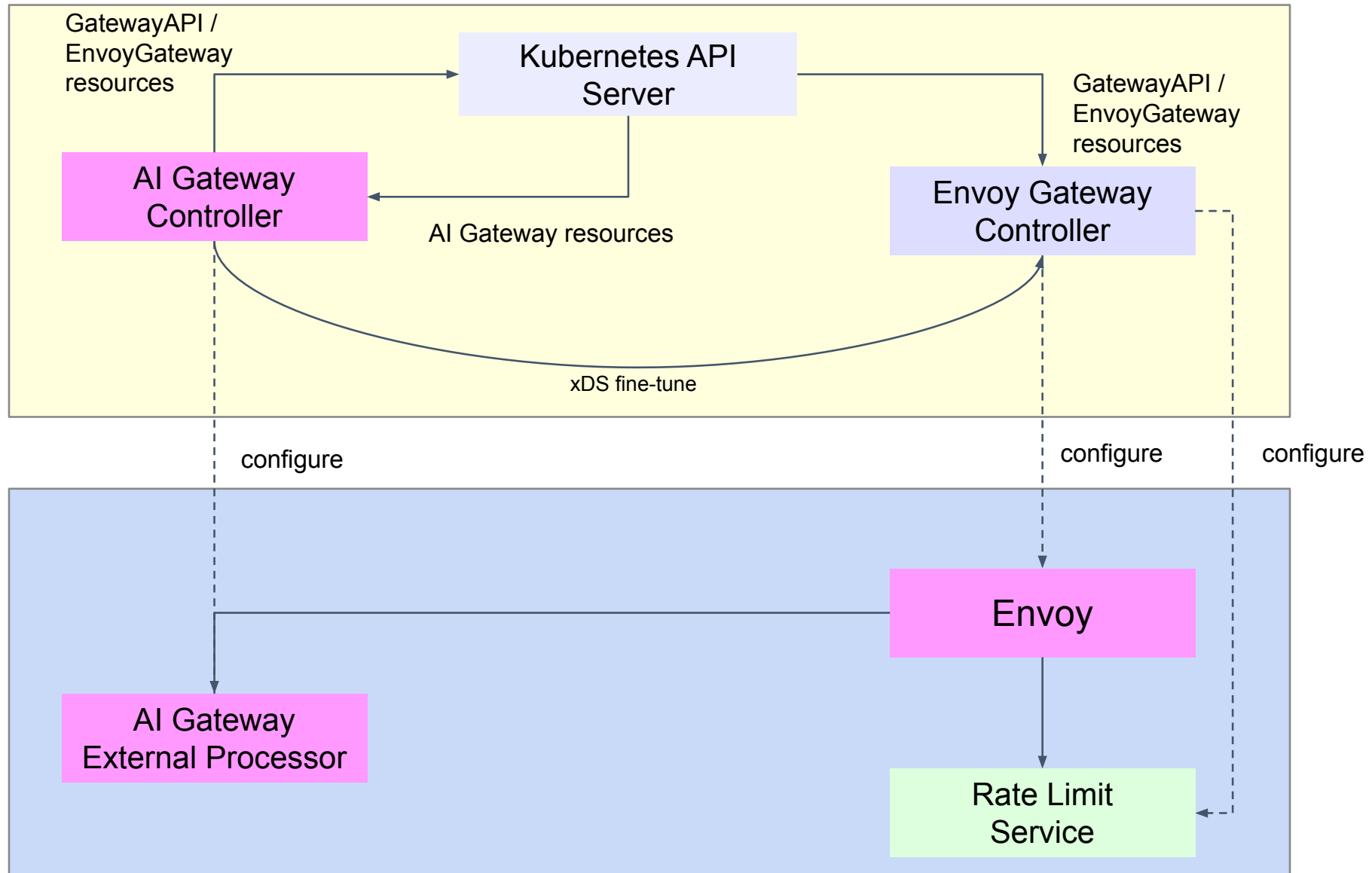
[View on GitHub](#)

- Provide a unified layer for routing and managing LLM/AI traffic
- Support automatic failover mechanisms to ensure reliability
- Ensure end-to-end security, including upstream authorization for LLM/AI traffic
- Implement a policy framework to support usage limiting use cases
- Foster an open-source community to address GenAI-specific routing and quality of service needs

Envoy AI Gateway Project Status

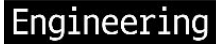


What We Left Off in 2025



Current Adopters

alan 

Bloomberg


LY Corporation



NUTANIX



Tencent Cloud



tetrate

- Unified APIs
 - OpenAI format for non-OpenAI providers
 - OpenResponses Compatible
 - Request/Response transformation
- Simplified Authentication
 - API Keys / OpenID Connect / Microsoft Entra ID / GCP Workload Federation
- Token-based Rate Limiting
- Cross-Provider/Region Fallbacks
 - OpenAI -> Azure -> Self-Hosted
 - GCP Vertex AI -> AWS Bedrock
- LLM Observability

Envoy AI Gateway CRDs are v1beta1 stable 🎉🎉🎉!!!

Approaching 2026

Cost Optimization

Source: Unsplash – Photo by Jakub Žerdzicki / Unsplash

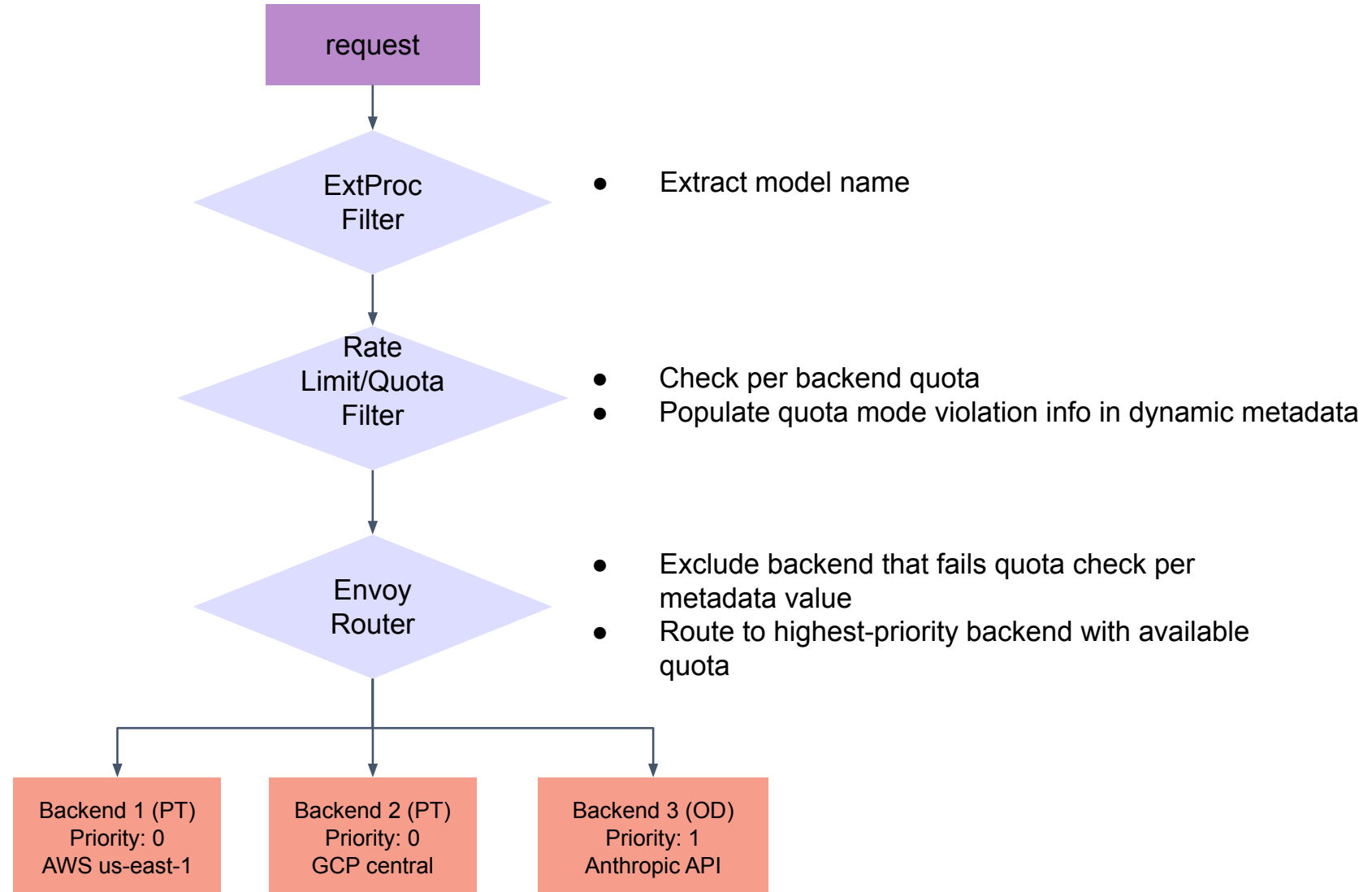
- **Quota-Aware Routing**
 - Provisioned Throughput (PT)
 - On-Demand (OD)
- **Quota Fallback**
 - Route to highest-priority back-end with available quota
 - If all PT backends exceeded, route to OD backends
- **Quota Policy**
 - Track backend quota capacity type and size

Cost Optimization - Quota-Aware Routing

```
apiVersion: ai-gateway.envoyproxy.io/v1alpha1
kind: AIServiceBackend
metadata:
  name: aws-claude-pt-us-east-1
  namespace: ai-gateway
spec:
  backendRef:
    name: bedrock-us-east-1
    port: 443
```

```
apiVersion: ai-gateway.envoyproxy.io/v1alpha1
kind: QuotaPolicy
metadata:
  name: aws-bedrock-model-quota
  namespace: ai-gateway
spec:
  perModelQuota:
    - modelName: claude-sonnet-4
      rules:
        - clientSelectors:
            - headers:
                - name: service_tier
                  value: reserved
          quotaValue:
            limit: 1M
            duration: 30s
        - clientSelectors:
            - headers:
                - name: service_tier
                  value: default
          quotaValue:
            limit: 2M
            duration: 60s
      targetRef:
        name: bedrock-us-east-1
```

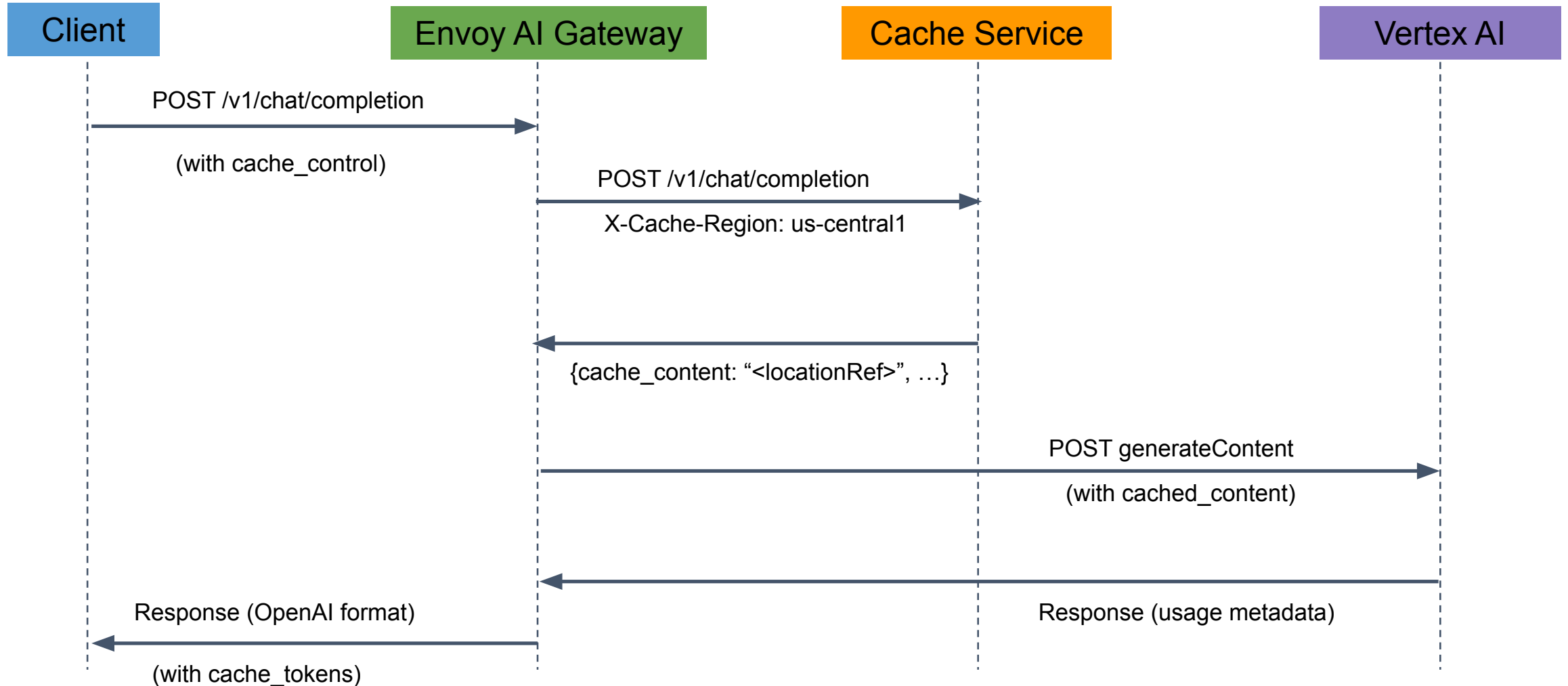
Cost Optimization - Quota-Aware Routing



Cost Optimization - Context Caching

- Implicit Context Caching
- Explicit Context Caching
 - e.g., Gemini Context Caching

Cost Optimization - Context Caching



- Batch Processing
 - Large-scale workloads
 - Higher throughput limit
 - Time-insensitive workload
 - Cost discount
 - APIs
 - /v1/files
 - /v1/batches

- Stateful API
- Designed for agentic workflows
- Context management
- OpenResponses API compatible

Turn 1 — store the response

```
curl $GATEWAY_URL/v1/responses \  
-H "Authorization: Bearer $TOKEN" \  
-H "Content-Type: application/json" \  
-d '{  
  "model": "gpt-5.1",  
  "input": "My name is Alice and I live in Paris.",  
  "store": true,  
  "context_management": "auto"  
}'  
# Returns: "id": "resp_abc123"
```

Turn 2 — reference previous response

```
curl $GATEWAY_URL/v1/responses \  
-H "Authorization: Bearer $TOKEN" \  
-H "Content-Type: application/json" \  
-d '{  
  "model": "gpt-5.1",  
  "input": "Where do I live?",  
  "previous_response_id": "resp_abc123",  
  "store": true,  
  "context_management": "auto"  
}'
```

- RAG - Retrieval Augmented Generation
 - /v1/embeddings
 - /v1/rerank

- Multimodal
 - Text-to-speech
 - /v1/audio/speech
 - Image generation
 - /v1/images/generations

- Data Plane
 - Millisecond-level performance
- Control Plane
 - Scale to 2,000 AI Gateway routes
 - Zero routing failure
 - Minimal resource consumption increase
 - default gRPC message size 4MB -> 25 MB

- **Website:** <https://aigateway.envoyproxy.io/>
- **GitHub:** <https://github.com/envoyproxy/ai-gateway>
- **Envoy / CNCF Slack:** #envoy-ai-gateway