

ArgoCon

NORTH AMERICA



ArgoCon
NORTH AMERICA

Ship AI Like You Mean It: Progressive Delivery for AI Workloads Using Agentgateway

Lasse Vierow & Kostis Kapelonis

ArgoCon
NORTH AMERICA



Lasse Vierow
Cloud Engineer
Independent



Kapelonis Kostis
Developer Advocate
Octopus Deploy
Argo Team

Agenda



1. Intro to Argo Rollouts
2. Intro to Gateway API
3. AI traffic vs traditional apps
4. Intro to agentgateway
5. Demo
6. Looking to the future
7. Wrap up



ArgoCon
NORTH AMERICA

Argo Rollouts

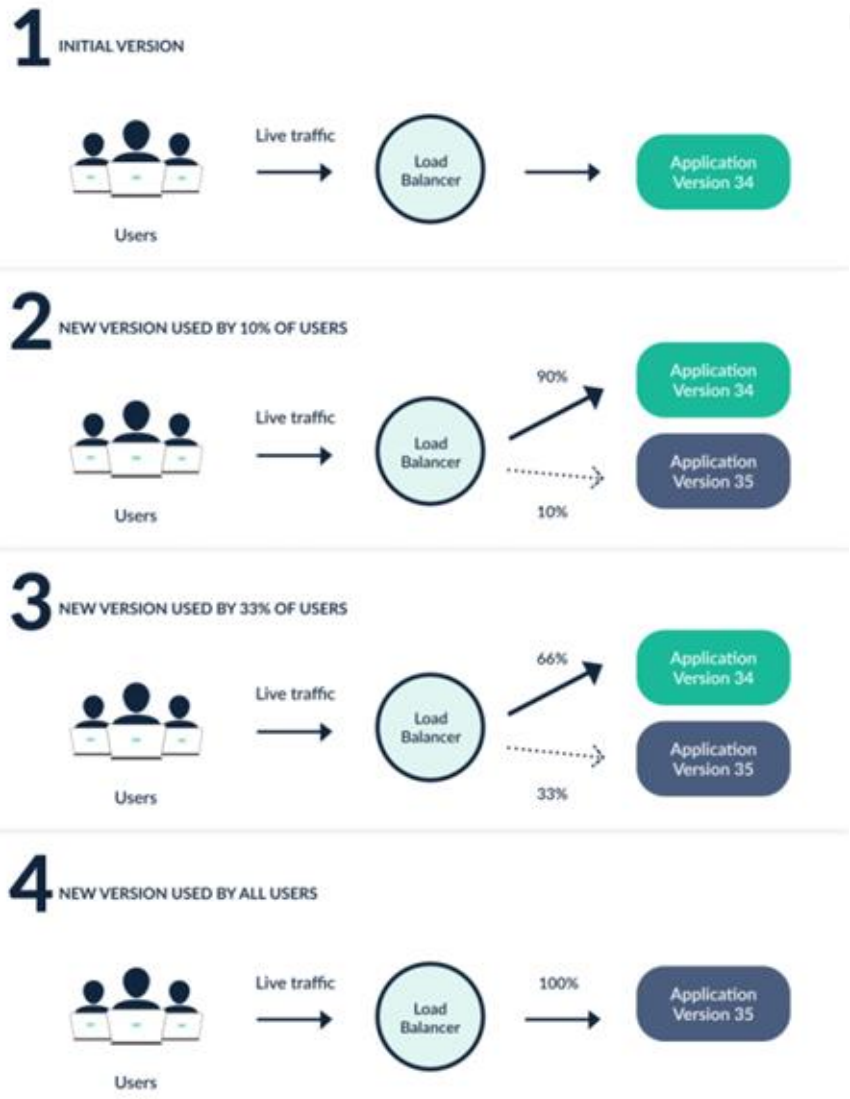
Progressive Delivery strategies

ArgoCon NORTH AMERICA

Blue/Green Deployment



Canary Release



Argo Rollouts

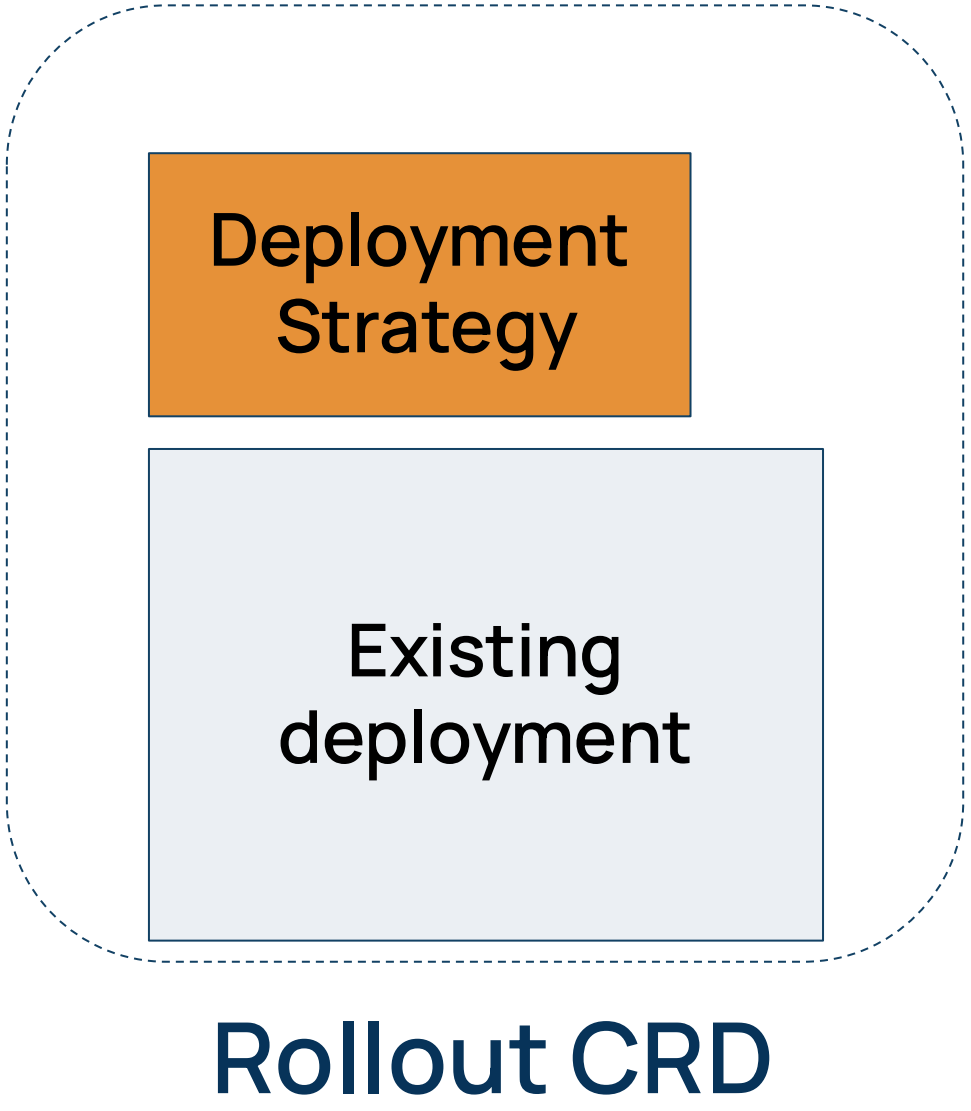
ArgoCon NORTH AMERICA

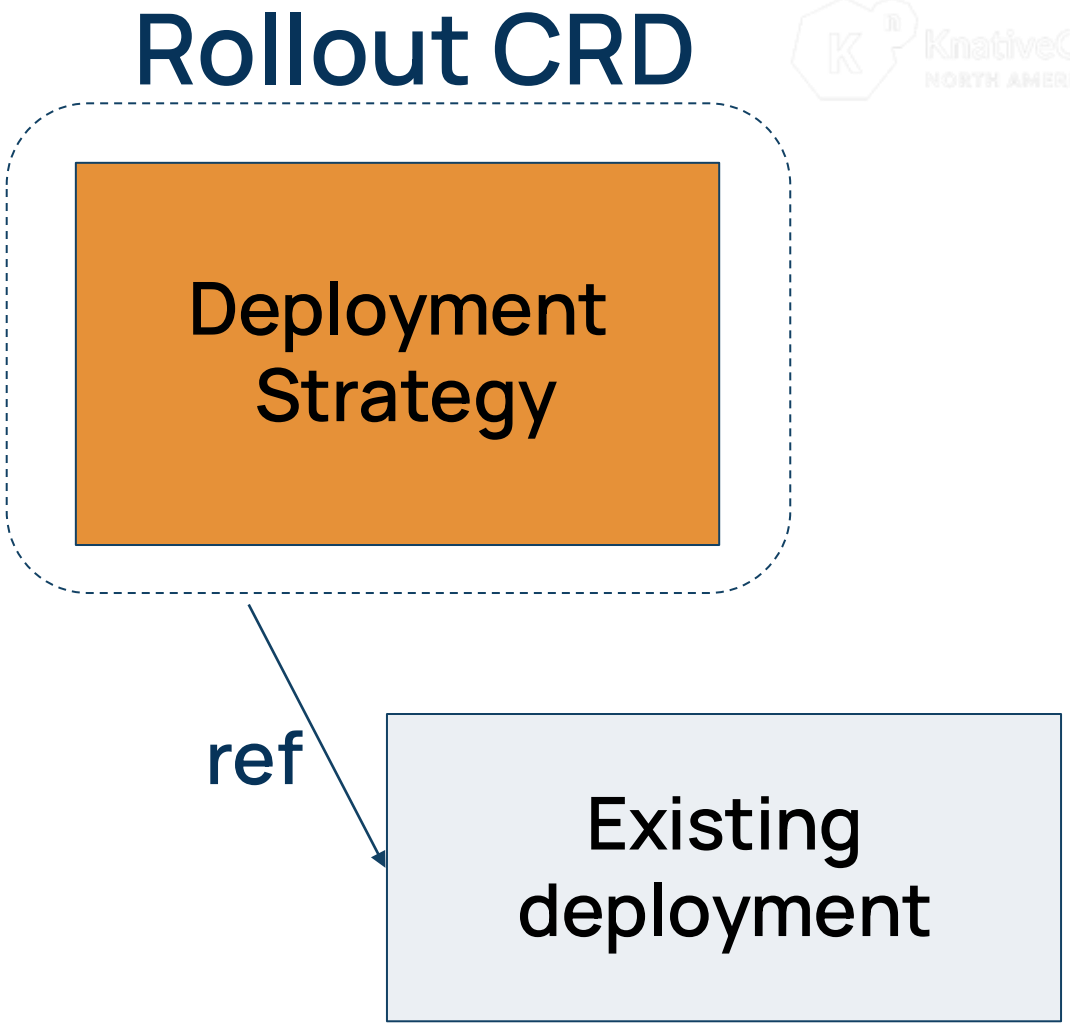
KnativeCon NORTH AMERICA

- Kubernetes native
- Standalone project
- Does **NOT** depend on Argo CD
- Blue/Green support
- Canary support
- A/B testing and other Experiments
- Zero downtime releases
- Automatic rollbacks based on metrics
- Installed on each deployment cluster



How Rollouts work



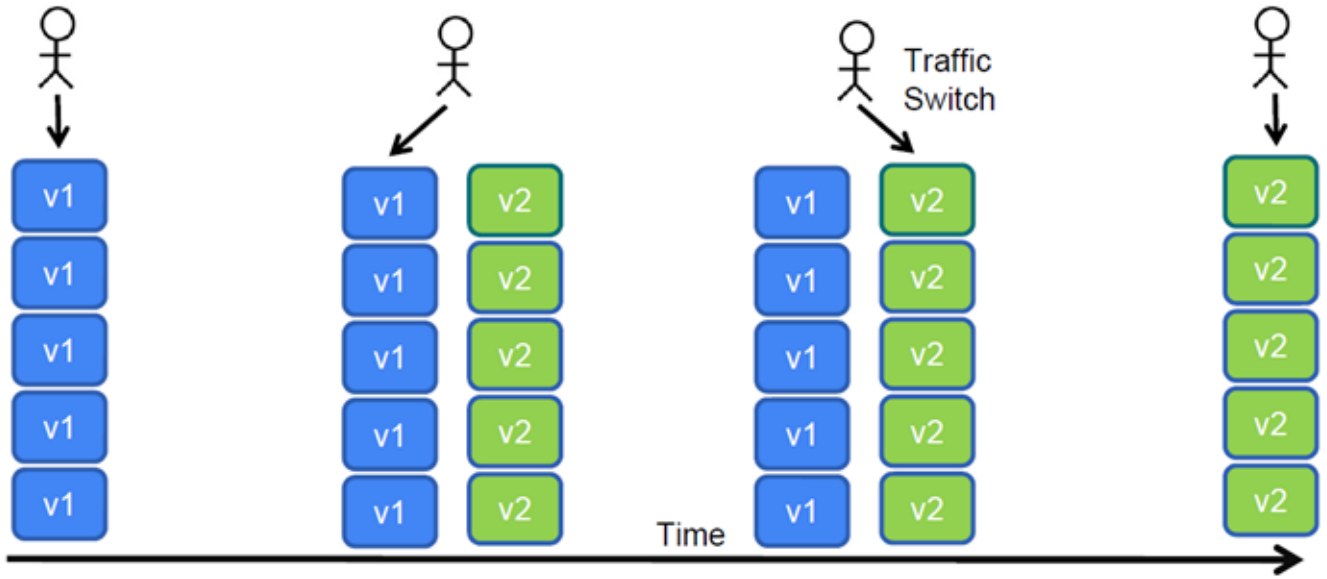


Rollout manifest


```yaml
apiVersion: argoproj.io/v1alpha1
kind: Rollout
metadata:
  name: canary-example
spec:
  replicas: 10
  revisionHistoryLimit: 2
  selector:
    matchLabels:
      app: cost-demo
  template:
    metadata:
      labels:
        app: cost-demo
    spec:
      containers:
        - name: cost-demo
          image: ghcr.io/kostis-codefresh/rollouts-autoscaling-example:v1
          ports:
            - name: http
              containerPort: 8080
              protocol: TCP
  strategy:
    canary:
      canaryService: argo-rollouts-canary-service
      stableService: argo-rollouts-stable-service
      steps:
        - setWeight: 10
        - pause:
            duration: 1h
        - setWeight: 20
        - pause: {}
```

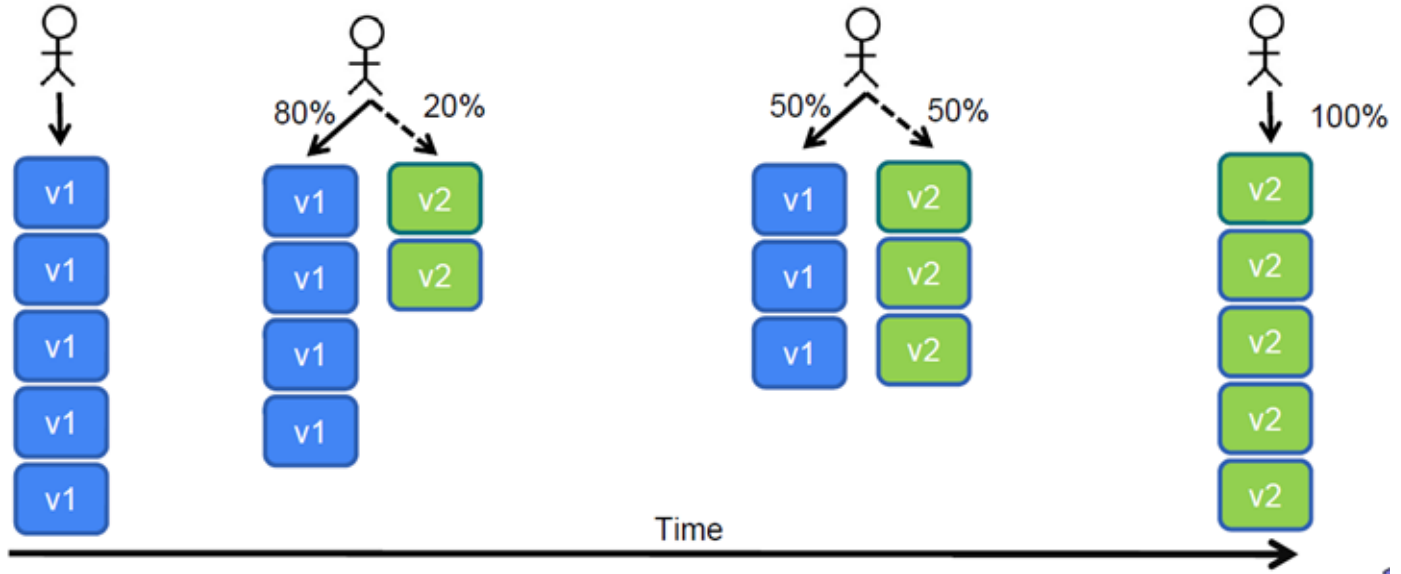
```yaml
strategy:
  canary:
    canaryService: argo-rollouts-canary-service
    stableService: argo-rollouts-stable-service
    steps:
      - setWeight: 10
      - pause:
          duration: 1h
      - setWeight: 20
      - pause: {}
```

Rollout = deployment PLUS strategy

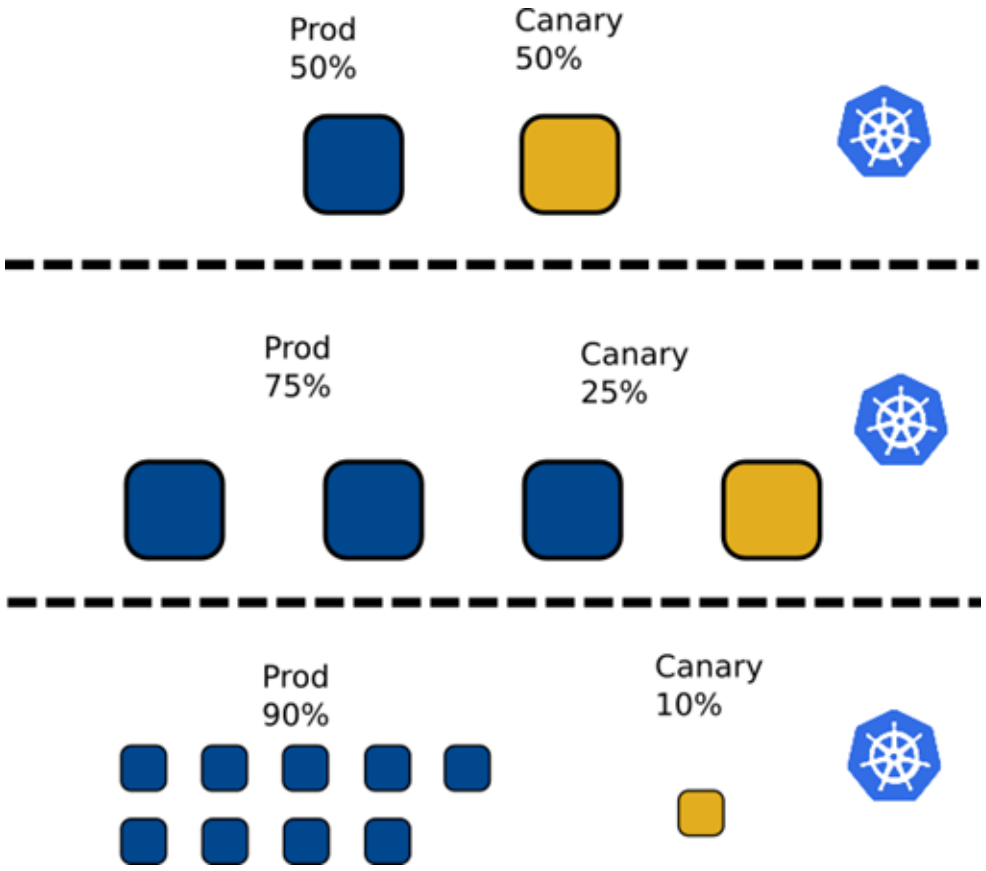


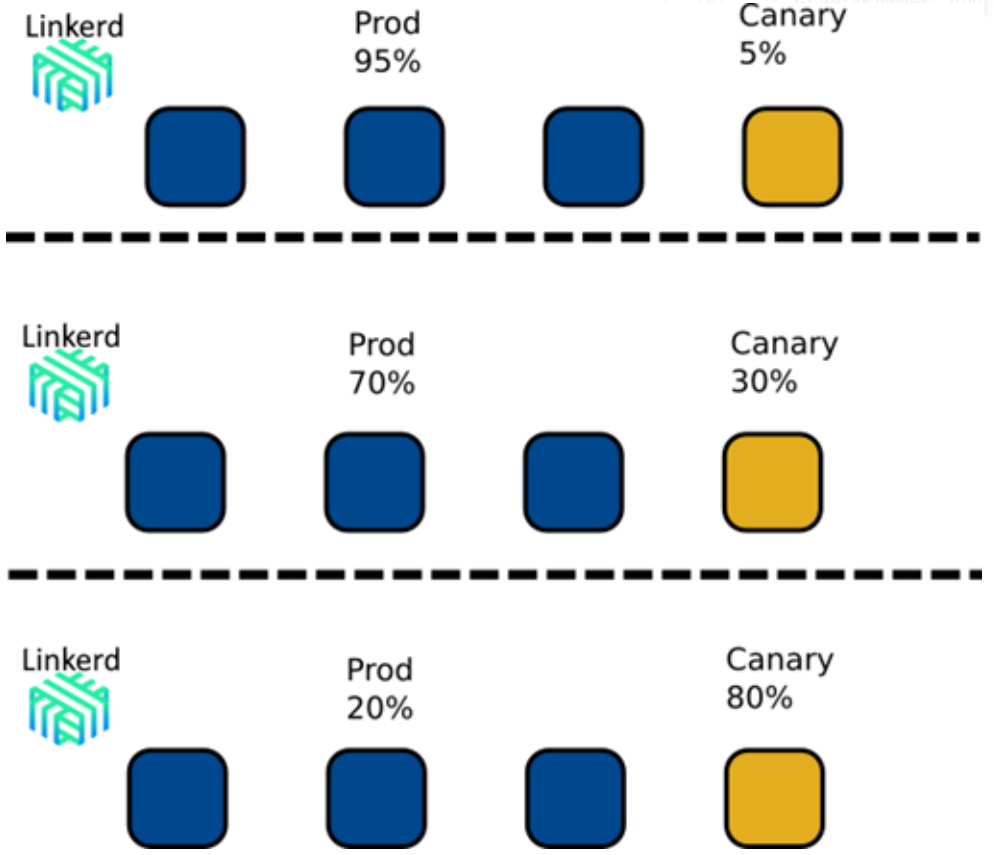
Kubernetes
Progressive
Delivery

KnativeCon NORTH AMERICA

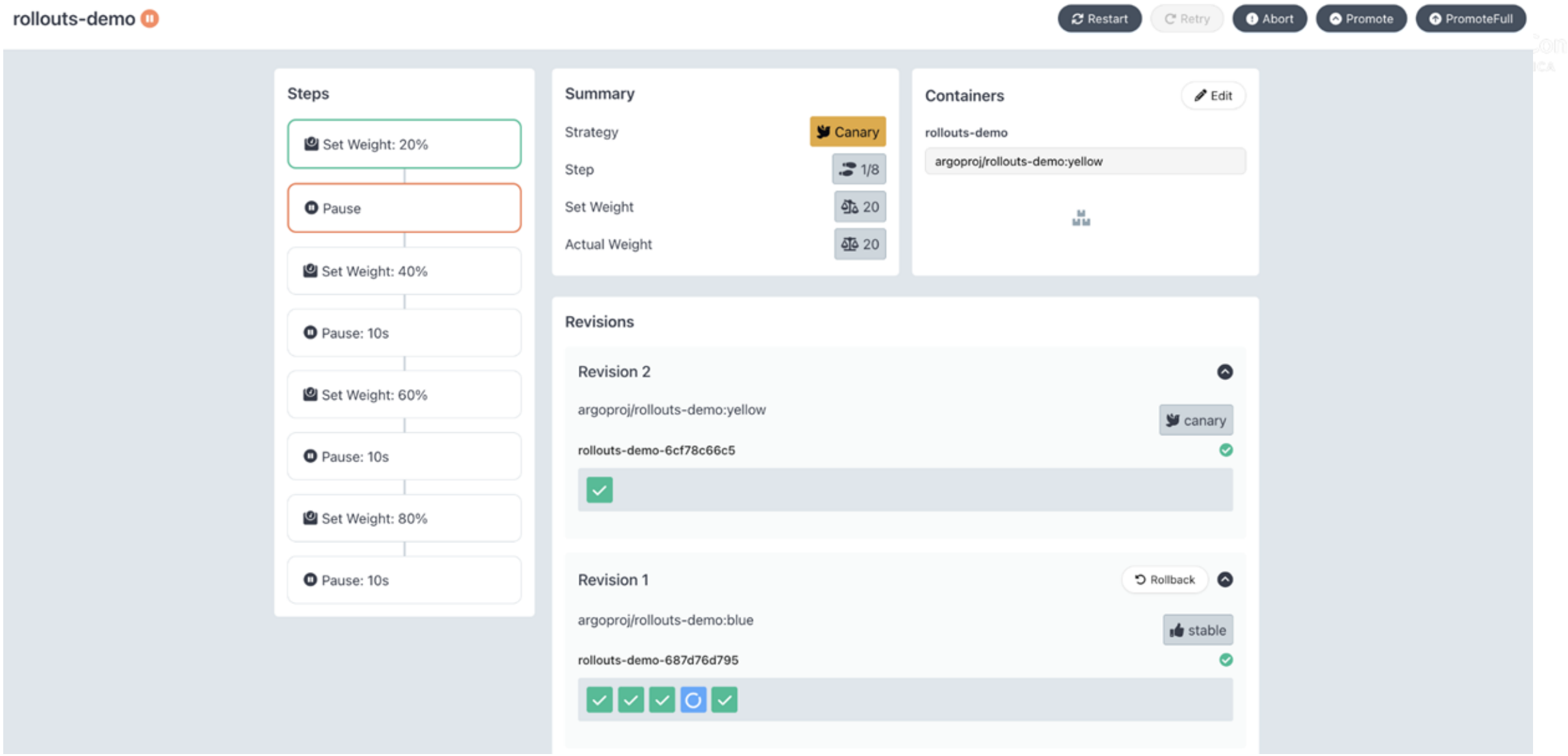


With/Without traffic managers

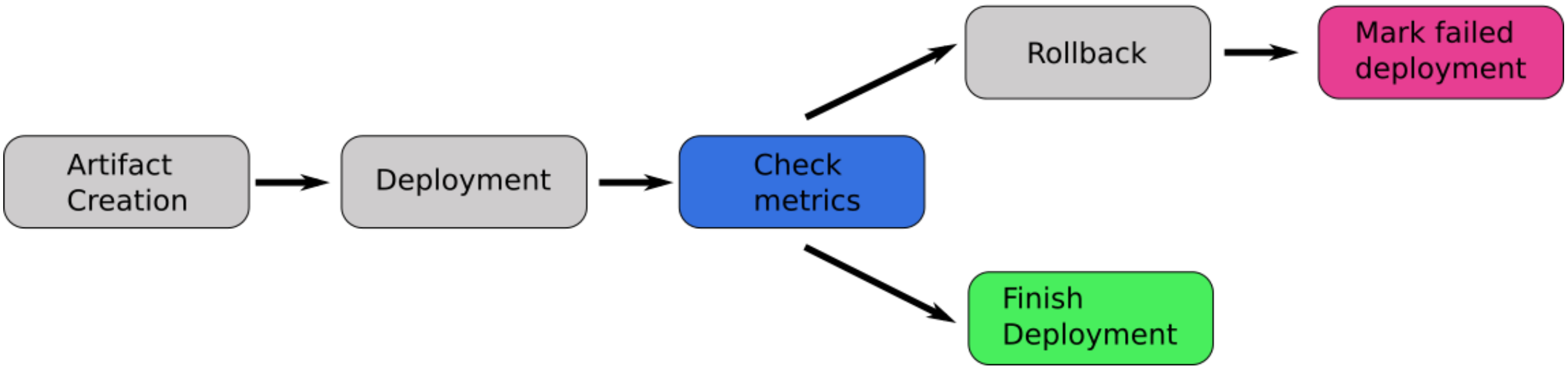




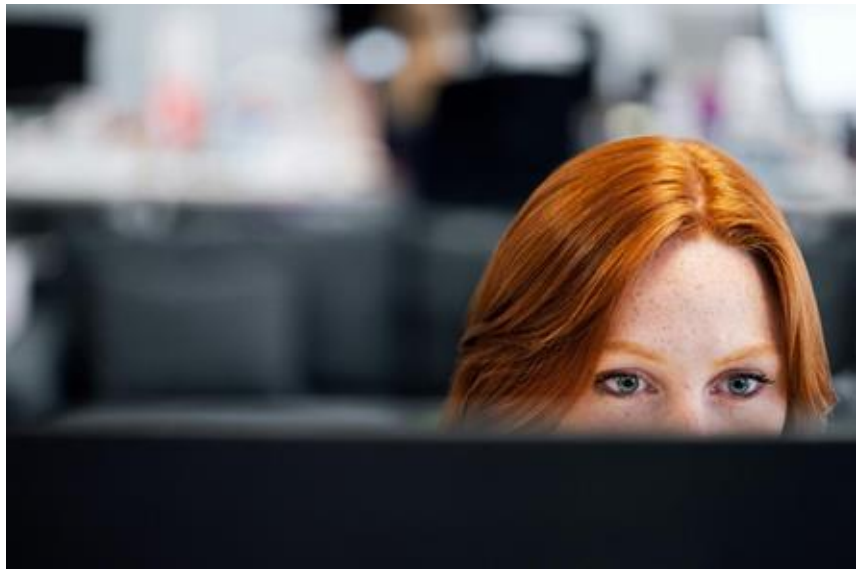
Graphical dashboard



Rollback automatically

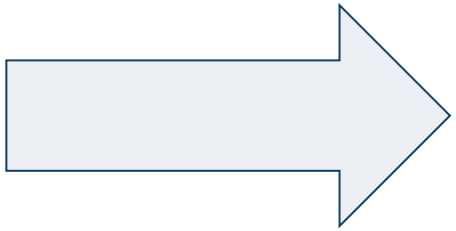


Deploy on Friday at 5:00





Deployment happens at 5.00 pm on Friday





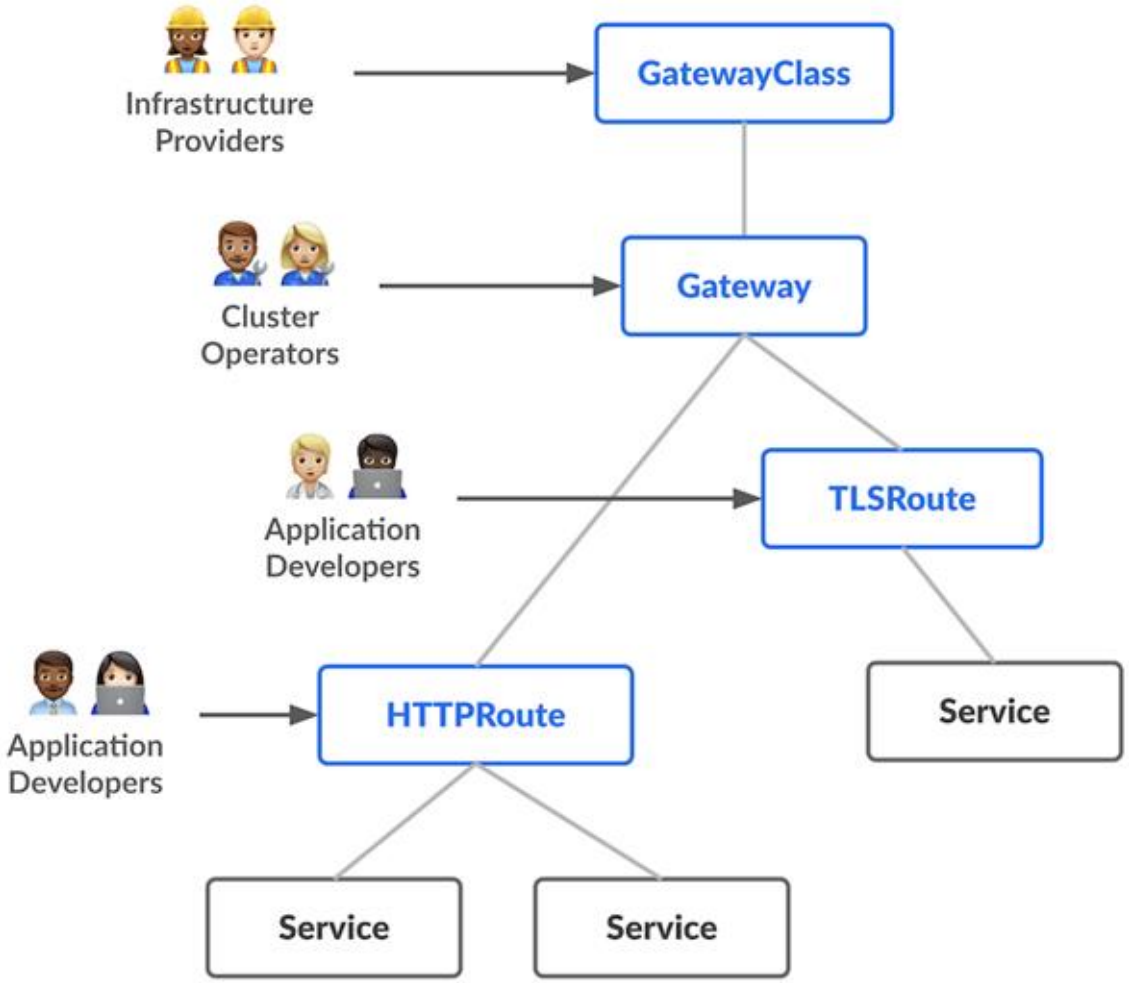
5:15 the whole team is at the pub



ArgoCon
NORTH AMERICA

Gateway API

Gateway API is next-gen ingress



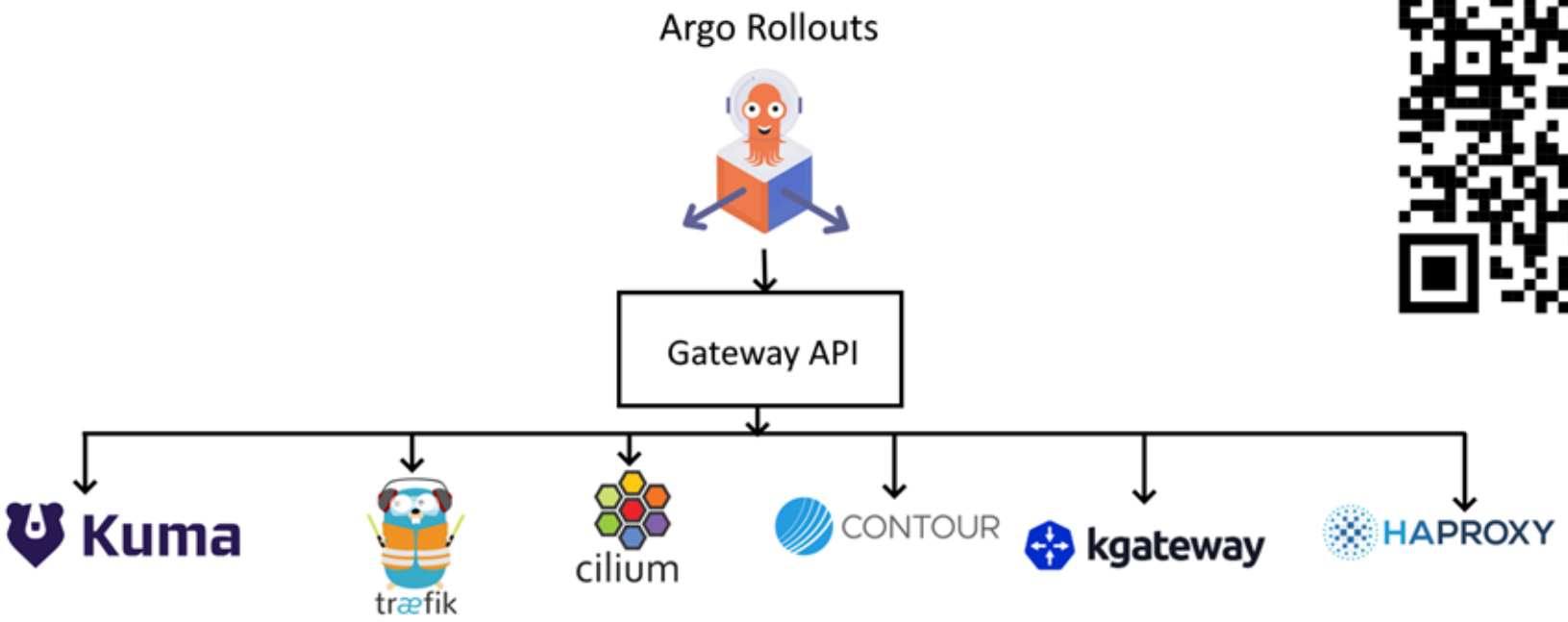
https://gateway-api.sigs.k8s.io/

Supported Traffic managers



- AWS Ingress Controller
- Ambassador Labs
- Apache APISIX
- Linkerd
- Istio
- Kong
- Nginx
- Traefik
- Openshift Routes
- kgateway
- Contour
- Cilium
- Envoy Gateway
- Gateway API

One API to rule them all



https://github.com/argoproj-labs/rollouts-plugin-trafficrouter-gatewayapi



ArgoCon
NORTH AMERICA

AI traffic vs traditional apps

Same request, two worlds

ArgoCon NORTH AMERICA

WEB APP

REQUEST

```
GET /api/products/42
Accept: application/json
```

RESPONSE

200 OK 12 ms · 0.3 KB

```
{"name": "Coffee mug", "price": 12}
```

Correct: code decides the answer

AI MODEL

REQUEST

```
POST /v1/chat/completions
"What are the continents?"
```

RESPONSE

200 OK 3 s · 90 tokens

There are five continents: Africa, America, Asia, Europe and Australia.

Wrong: there are seven continents

200 OK only means: the model answered. It does not mean: the answer is right.

Why AI Traffic Breaks the Old Playbook



	TRADITIONAL WEB WORKLOAD	AI / AGENTIC WORKLOAD
Request shape	Milliseconds, request / response	Seconds, often streamed
Cost per request	Fractions of a cent, CPU-bound	Cents to dollars, driven by tokens
Failure mode	Loud: 5xx, crashes, instant alerts	Silent: confident, wrong, still “200 OK”
Attack surface	Input validation, DB injection	Prompt injection, jailbreaks, exfiltration

A canary for AI needs new checks

ArgoCon NORTH AMERICA

CLASSIC CANARY

Error rate
5xx responses

Latency
Slow responses

Still needed for AI.
But not enough.

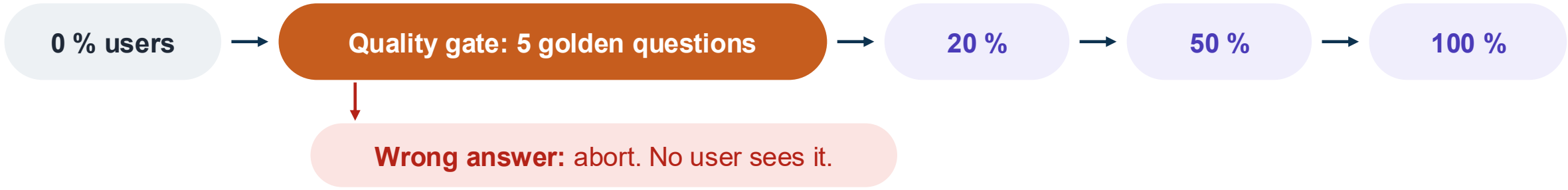
AI CANARY: ALSO CHECK

Answer quality
Ask golden questions. Are the answers right?

Tokens & cost
Does the new version use more tokens?

Safety
Does the guardrail still block bad prompts?

IN OUR DEMO



What we need

ArgoCon NORTH AMERICA
KnativeCon NORTH AMERICA

- Agent-to-agent (A2A)
- Model Context Protocol (MCP)
- REST APIs as agent-native tools
- AI routing to cloud and local LLMs
- Support for the Gateway API Inference Extension project



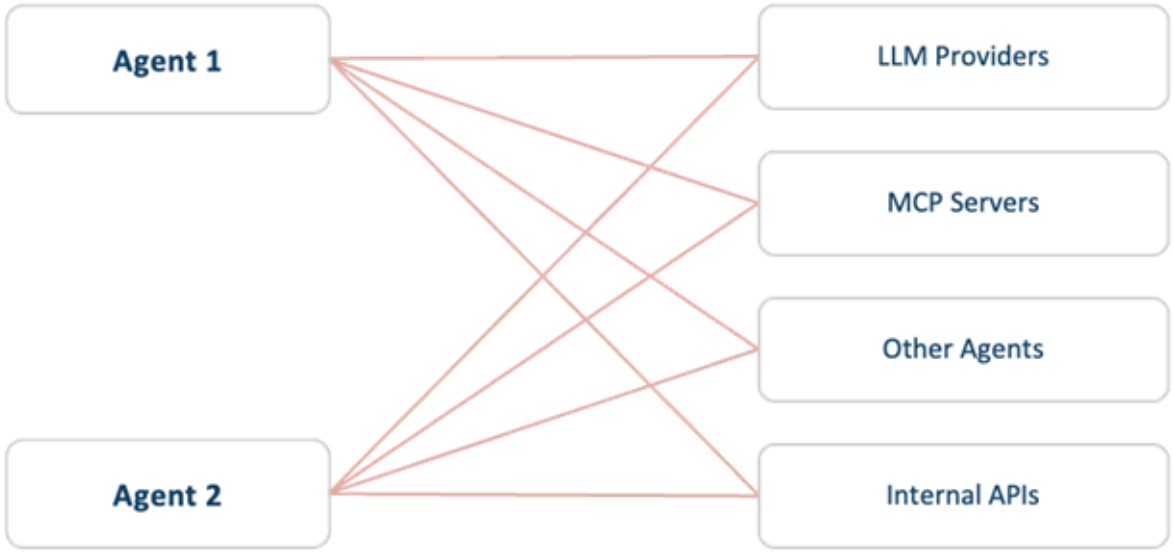
ArgoCon
NORTH AMERICA



agentgateway

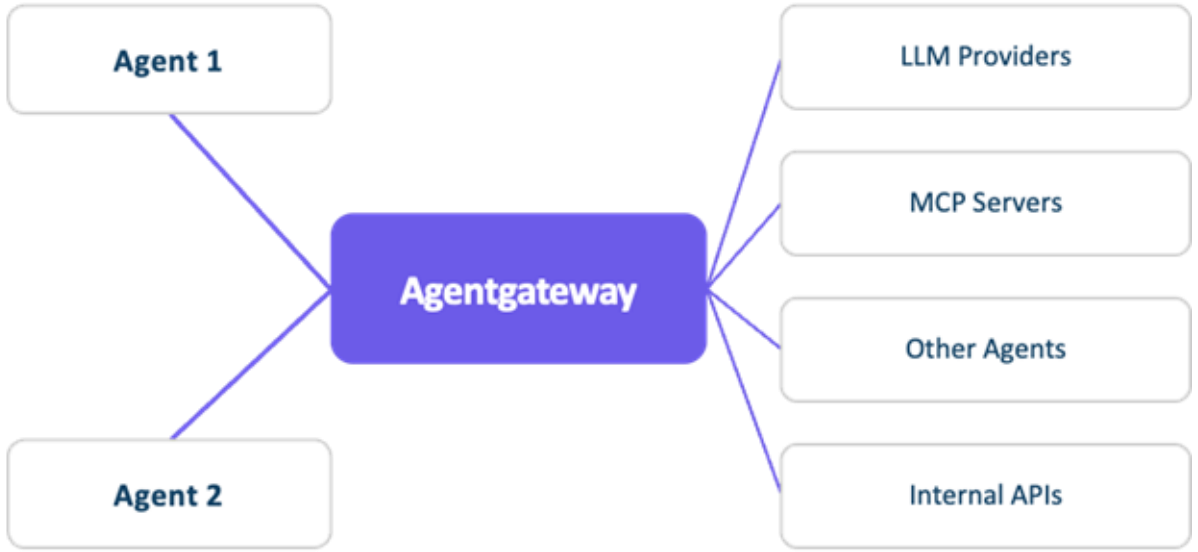
KnativeCon NORTH AMERICA

WITHOUT A GATEWAY



Every team wires its own auth, retries, rate limits, logging

WITH AGENTGATEWAY



One governed hop — auth, cost, guardrails, routing, all in one place

What is agentgateway?

ArgoCon NORTH AMERICA

Open source
Linux Foundation project (Agentic AI Foundation)

Written in Rust
A small and very fast proxy

Speaks AI
LLM APIs, MCP and A2A, plus normal HTTP and gRPC

Kubernetes Gateway API
Gateway, HTTPRoute: like any other gateway

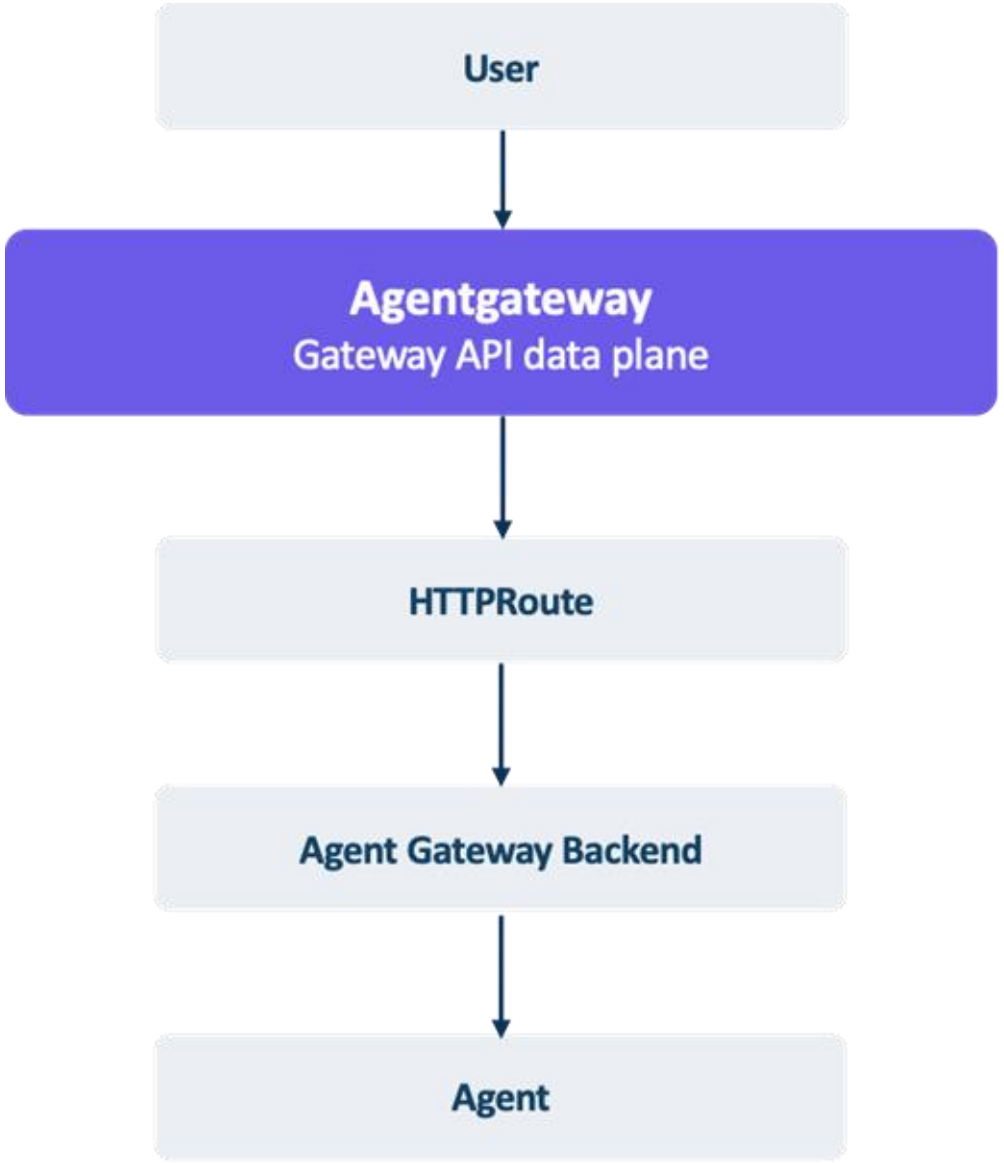
Extra CRDs for AI
AgentgatewayBackend, AgentgatewayPolicy

IN ONE SENTENCE

An API gateway that understands prompts, models and tokens.

It can read what is inside an AI request, not only the URL and the status code.

How it works


User
Agentgateway
Gateway API data plane
HTTPRoute
Agent Gateway Backend
Agent

One request through agentgateway

ArgoCon NORTH AMERICA

EVERY REQUEST

STEP 1

Check prompt

Bad prompt? Stop it before the model sees it.

→

STEP 2

Pick backend

Stable or canary: by weight or by header.

→

STEP 3

Call model

Forward the request to the model server.

→

STEP 4

Read answer

Count the tokens and add response headers.

→

STEP 5

Log it

Prompt, answer and tokens in one log line.

IN OUR DEMO

STEP 1	STEP 2	STEP 3	STEP 4	STEP 5
`prompt injection → 403 Forbidden`	`x-canary: eval → canary`	`vLLM Qwen2.5-0.5B`	`x-total-tokens x-served-by`	`access log: prompt + answer`

All of this is plain YAML: one HTTPRoute + one AgentgatewayPolicy `(4-agentgateway-policies.yaml)`

K8s service vs Agent Gateway backend

ArgoCon NORTH AMERICA

	KUBERNETES SERVICE `kind: Service`	AGENTGATEWAYBACKEND `agentgateway.dev/v1alpha1`
Points to	Pods, selected by labels	LLM provider, MCP server or A2A agent, cloud or in-cluster
Gateway sees	Generic HTTP: status codes & latency	LLM traffic: tokens, model & cost as `gen_ai.*` telemetry
Policies	Retries, timeouts, TLS, request limits	Plus provider auth, token-based limits, failover, guardrails
Argo Rollouts	Shifts stable / canary weights today	Not yet: plugin workloads would enable it

Service = where the traffic goes

AgentgatewayBackend = what the traffic is

Benefits

ArgoCon NORTH AMERICA
KnativeCon NORTH AMERICA

Cost controls & budgets

Invoice-grade attribution, cost dashboard, spend limits per model.

Layered guardrails

Moderation, AWS Bedrock Guardrails, Google Model Armor, custom webhooks.

Token-based rate limiting

Local and global limits, measured the way an LLM bill actually works.

Failover & content routing

Multiple providers, model aliasing, routing on request content.

MCP & A2A + tool RBAC

Static, dynamic and virtual MCP, with tool-level access control.

Traces with token usage

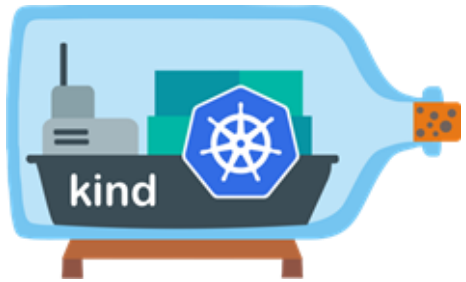
Full observability stack — tokens as a first-class span attribute.



ArgoCon
NORTH AMERICA

DEMO

Tools used in the demo



kind

Local Kubernetes cluster



Argo Rollouts

Progressive delivery

agentgateway

agentgateway

AI-aware traffic routing

vLLM

vLLM

LLM inference serving

Today: the demo uses Kubernetes Services

ArgoCon NORTH AMERICA

```yaml
# 3-httproute.yaml
kind: HTTPRoute
spec:
  parentRefs:
    - name: agentgateway-proxy
  rules:
    - backendRefs:
        - name: assistant-stable   # Service
          weight: 100
        - name: assistant-canary   # Service
          weight: 0
```

Argo Rollouts changes the weights: **0 → 20 → 50 → 100**

Why Services?



Argo Rollouts manages pods

It needs a stable and a canary Service.



The plugin shifts weights

Between these two Services in the HTTPRoute.

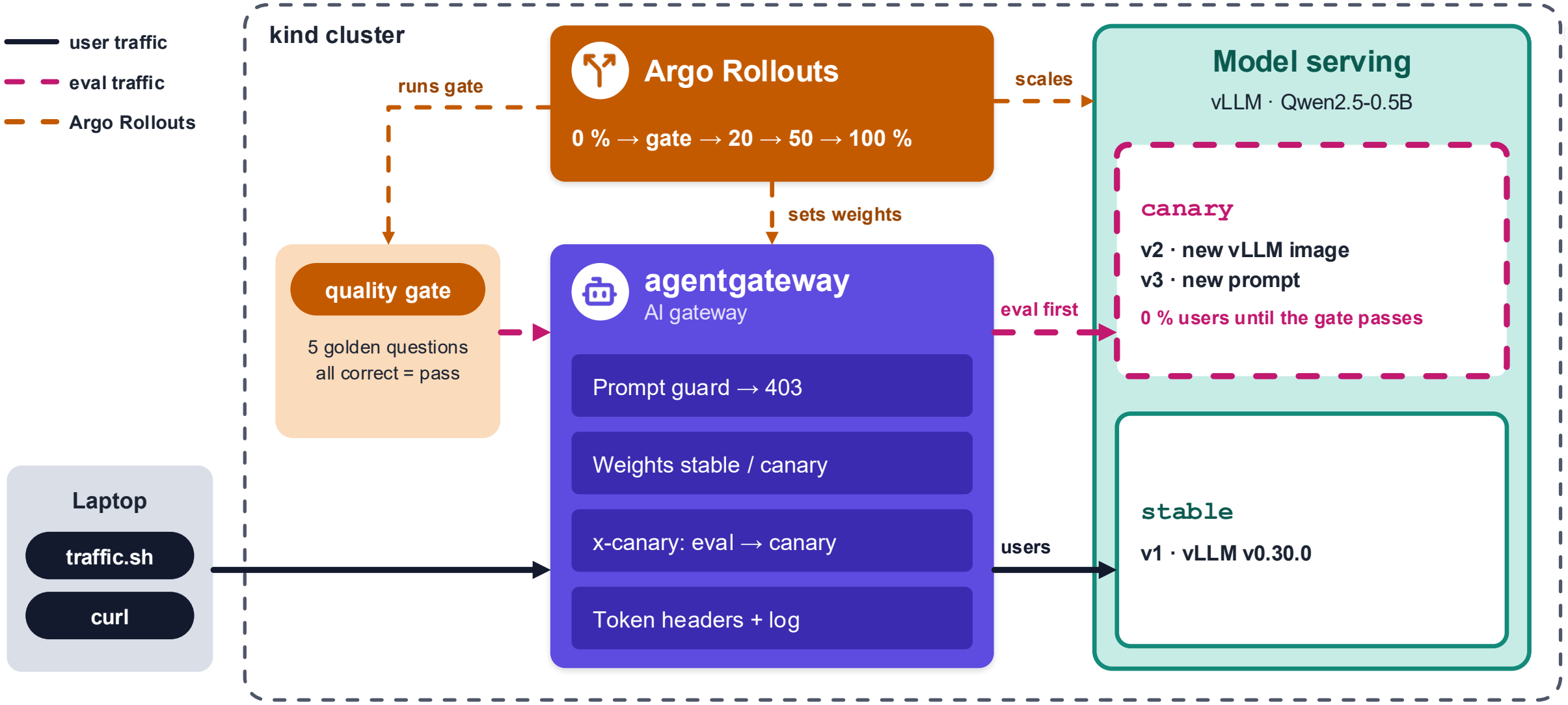


AgentgatewayBackend: not yet

More in “Looking to the future”.

Still AI-aware: the AgentgatewayPolicy on the route gives us the guardrail, token headers and logs.

Demo setup: what runs where



If you want to follow

ArgoCon NORTH AMERICA

Demo repo on GitHub

Runs on your laptop: kind

`0-platform.yaml`	Gateway + Rollouts plugin
`1-prompts.yaml`	Good and broken prompt
`2-services.yaml`	Stable + canary Service
`3-httproute.yaml`	Route via agentgateway
`4-agentgateway-policies.yaml`	Guardrail, tokens, logs
`5-quality-gate.yaml`	5 golden questions
`6-rollout-v1|v2|v3.yaml`	The three releases
`traffic.sh`	User traffic

github.com/Lasse4/ArgoConNorthAmerica2026



ArgoCon
NORTH AMERICA

DEMO



ArgoCon
NORTH AMERICA

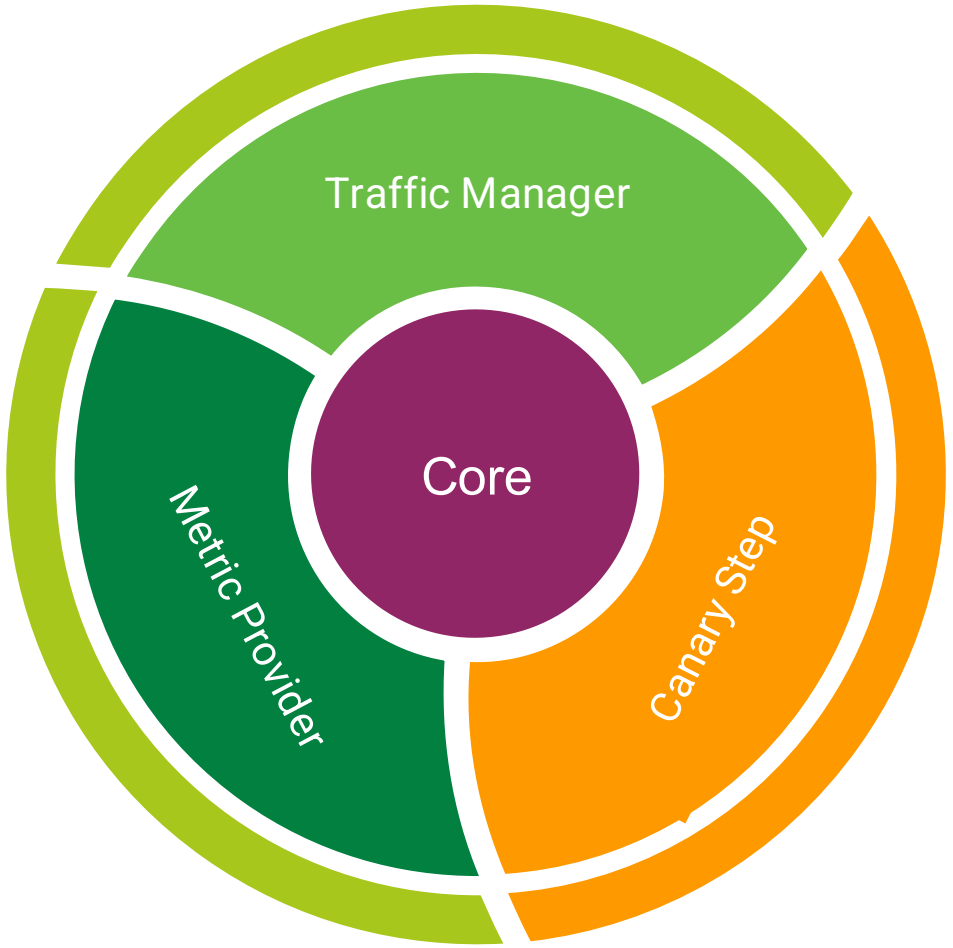
looking to the future

Argo Rollouts Plugin architecture

ArgoCon NORTH AMERICA

KnativeCon NORTH AMERICA

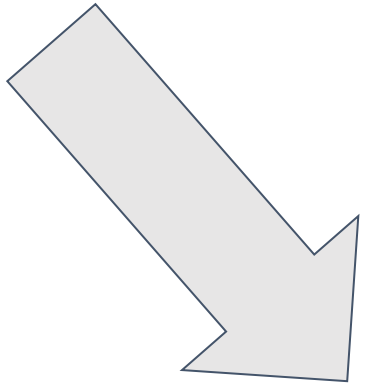
1. Traffic manager plugins
2. Metric provider plugins
3. Step plugins

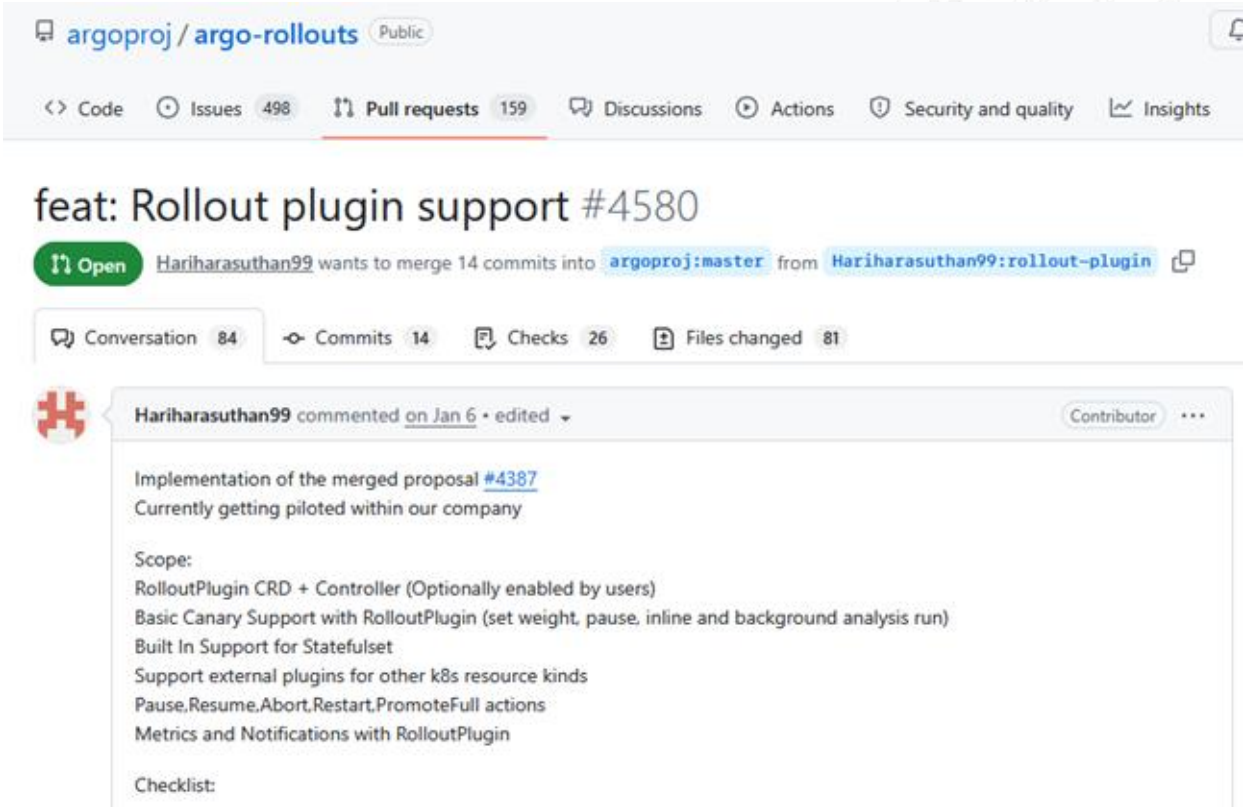


Keep Argo Rollouts core simple and minimal

Current trends for Argo Rollouts

1. Multi-cluster Rollouts
2. Stateful Rollouts
3. Plugin workloads





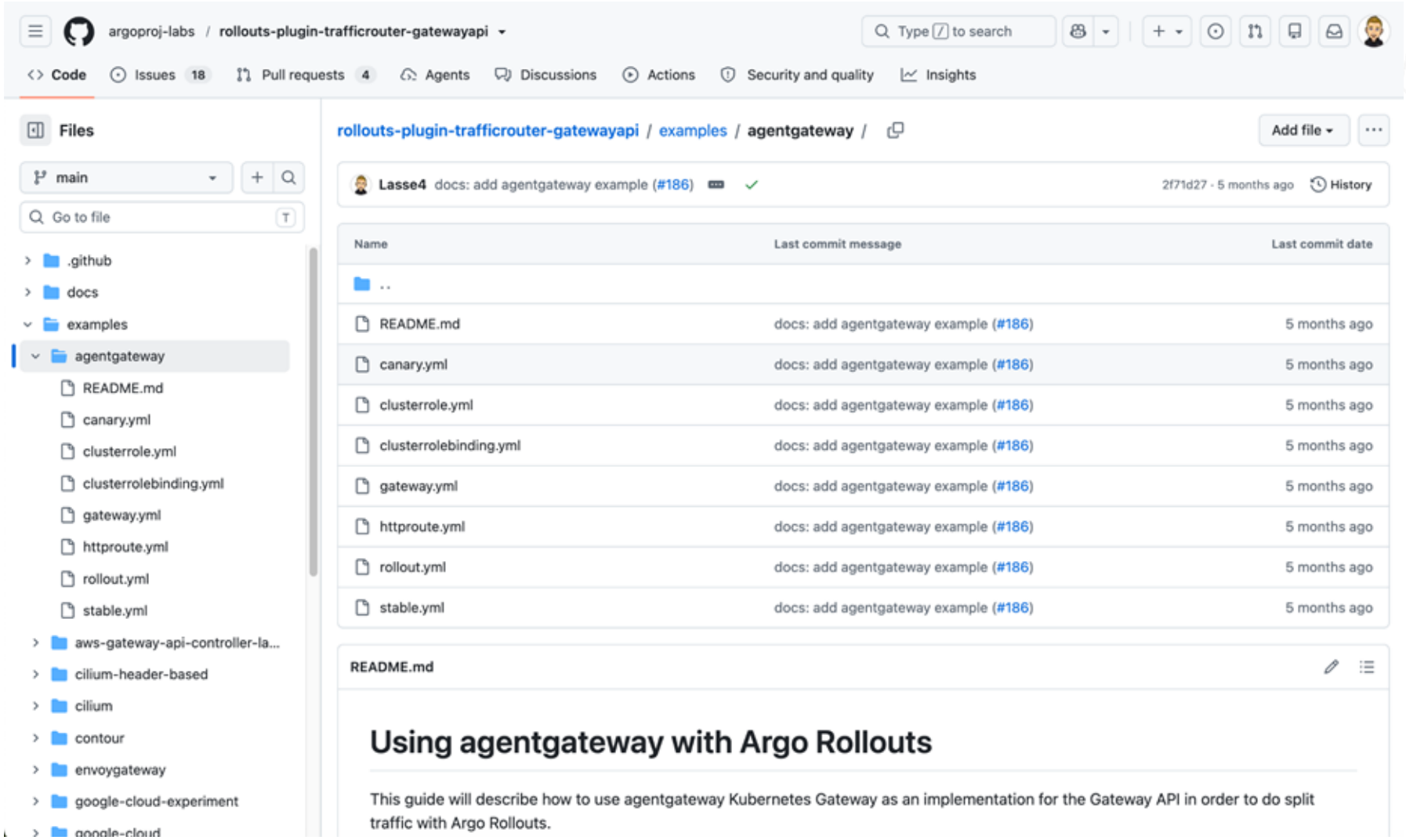
https://github.com/argoproj/argo-rollouts/pull/4580



ArgoCon
NORTH AMERICA

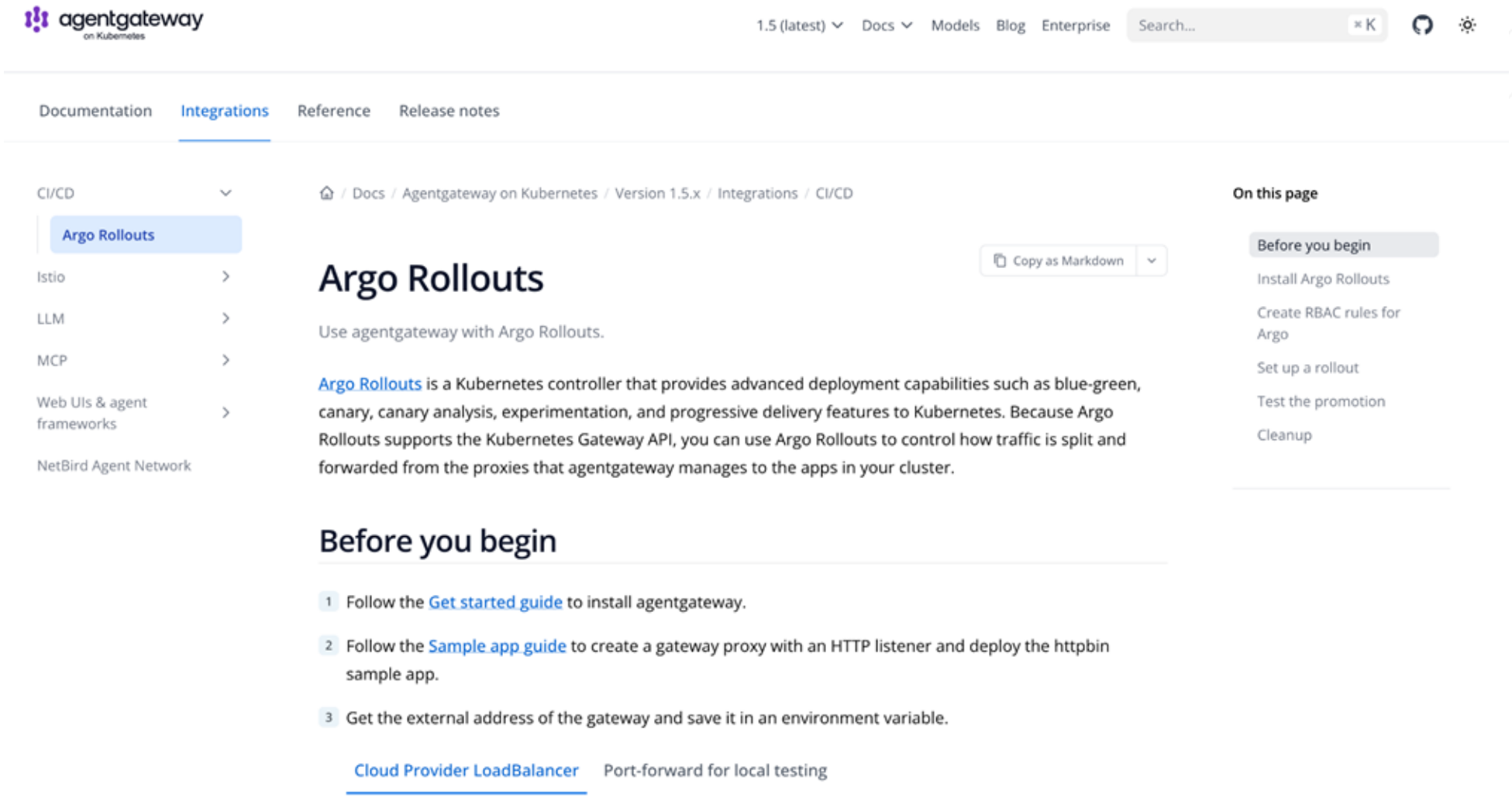
Wrap Up

How to get started



https://github.com/argoproj-labs/rollouts-plugin-trafficrouter-gatewayapi/tree/main/examples/agentgateway

How to get started



https://agentgateway.dev/docs/kubernetes/latest/integrations/ci-cd/argo/

More Questions?



agentgateway

CNCF Slack:

https://cloud-native.slack.com/archives/C01U781DW2E

Discord:

https://discord.gg/y9efgEmppm



Lasse Vierow

lasse@vierow.de

Ship AI Like You Mean It: Progressive Delivery for AI Workloads Using Agentgateway



Kapelonis Kostis

kostis.kapelonis@octopus.com