BUOYANT

weaveworks

Flagger, Linkerd, Gateway API: Oh My!

Jason Morgan, Buoyant
Sanskar Jaiswal, Weaveworks

@BuoyantIO buoyant.io @weaveworks weave.works

Jason Morgan

Technical Evangelist @ Buoyant

@RJasonMorgan

JasonMorgan

@Jason Morgan (Linkerd Slack)

Sanskar Jaiswal

Software Engineer @Weaveworks

@aryan9600_

aryan9600

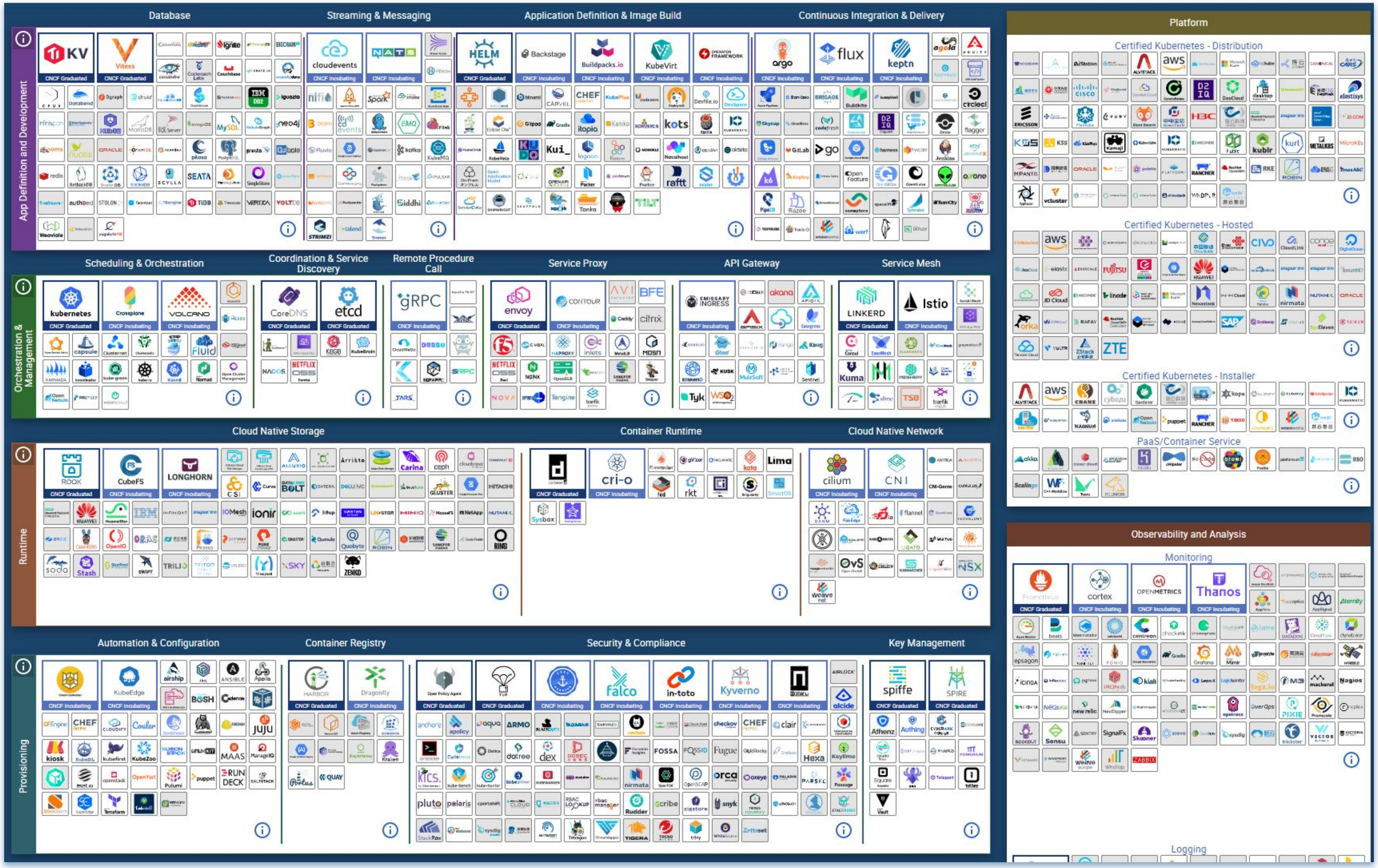
@aryan9600 (CNCF Slack)

Gateway API: Introduction

The next generation of Kubernetes networking and load balancing.

Packaged as a new set of APIs, it addresses the shortcomings of Ingress and evolves over it through expressive, extensible, and role-oriented interfaces.

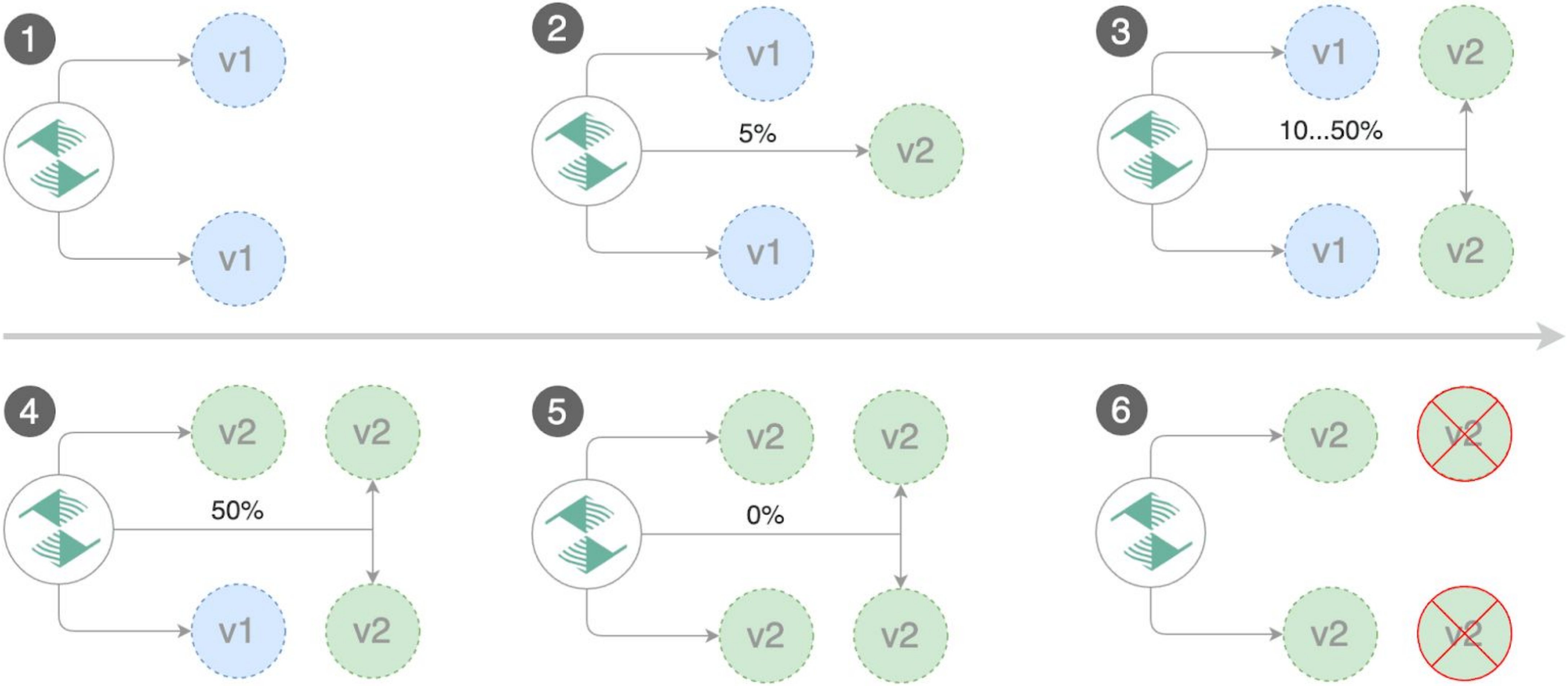
Gateway API: Why?


Database
Streaming & Messaging
Application Definition & Image Build
Continuous Integration & Delivery
App Definition and Development
Scheduling & Orchestration
Coordination & Service Discovery
Remote Procedure Call
Service Proxy
API Gateway
Service Mesh
Orchestration & Management
Cloud Native Storage
Container Runtime
Cloud Native Network
Runtime
Automation & Configuration
Container Registry
Security & Compliance
Key Management
Provisioning
Platform
Certified Kubernetes - Distribution
Certified Kubernetes - Hosted
Certified Kubernetes - Installer
PaaS/Container Service
Observability and Analysis
Monitoring
Logging

Gateway API: How does it help?

- More interoperability
- Fewer CRDs

Progressive Delivery



Flagger Overview

Flagger is a progressive delivery tool that automates the release process for applications running on Kubernetes.

It enables:

- ➔ **Safer Releases**: Reduce the risk of introducing a new software version in prod by gradually shifting traffic to it and rollback safely if needed.
- ➔ **Extensive Validation**: Validate SLOs/KPIs by running load tests, acceptance tests and analysing metrics collected via Prometheus, DataDog, StackDriver, etc
- ➔ **Flexible Traffic Routing**: Support for Linkerd, Istio, NGINX, Contour, Skipper, etc.

LINKERD

Ultralight, ultrafast, security-first **service mesh** for Kubernetes.

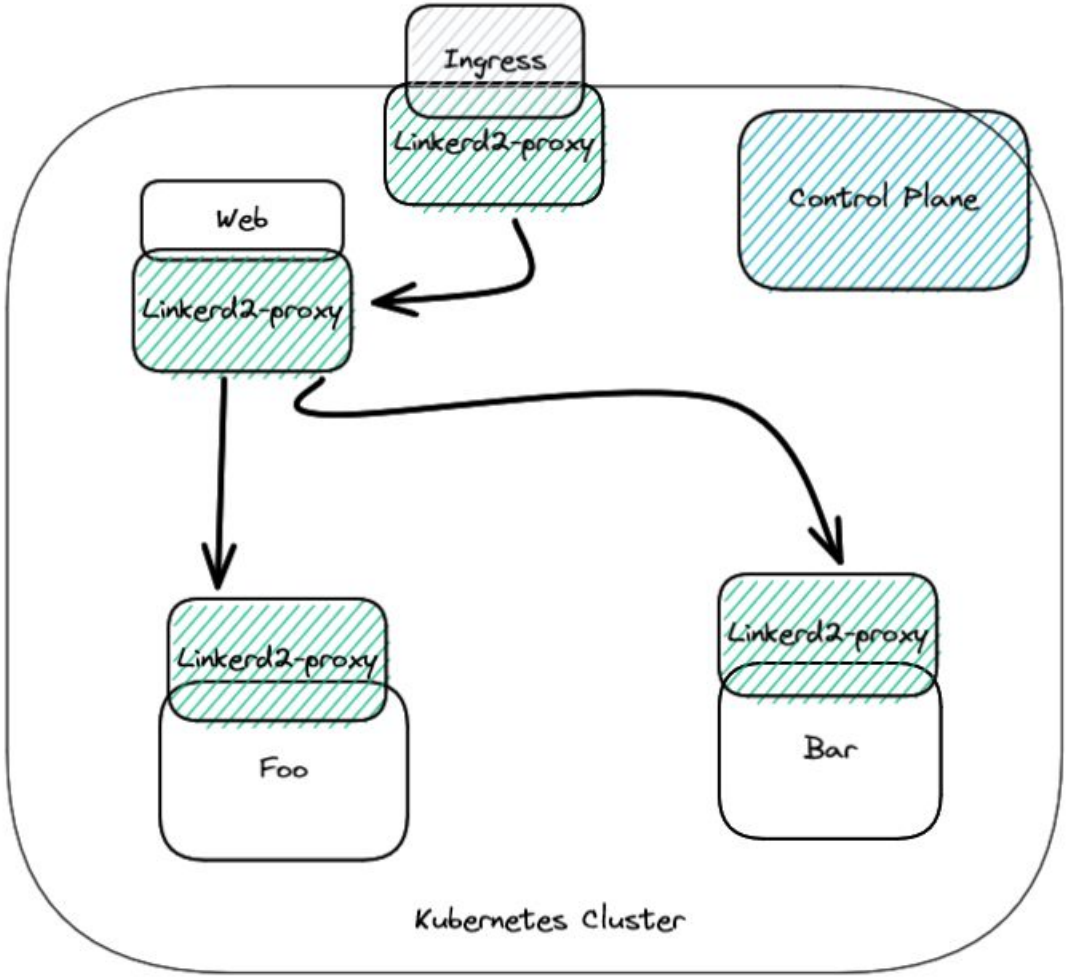
- → Created by **Buoyant**
- → **5+** years in production
- → **7,000+** Slack channel members
- → **10,000+** GitHub stars
- → **200+** contributors
- → **Weekly** edge releases
- → **Open governance**, neutral home
- → The only **CNCF graduated** mesh

CLOUD NATIVE
COMPUTING FOUNDATION
GRADUATED PROJECT

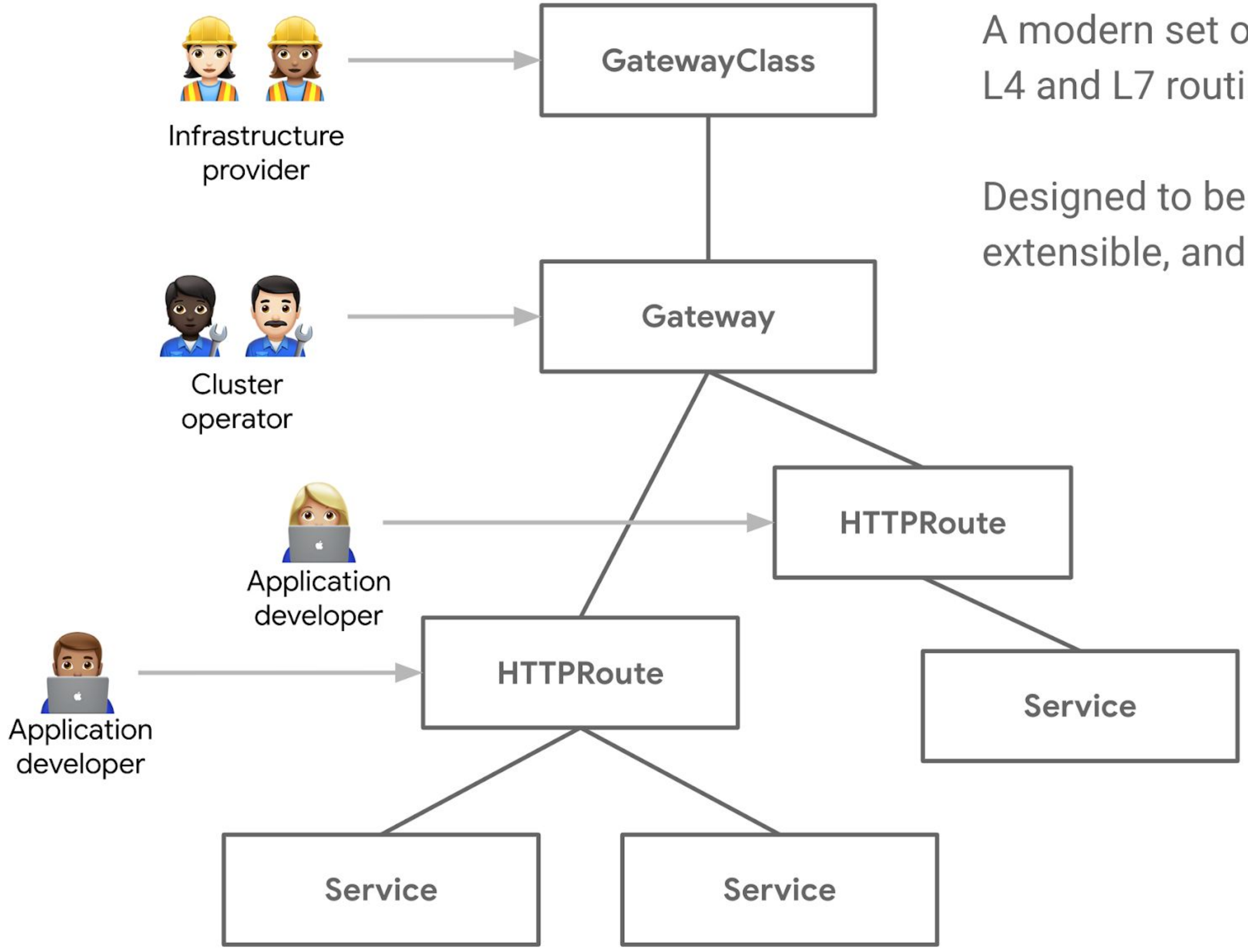
Ask
ATTEST
BIGCOMMERCE
CHASE
cisco Webex
COMCAST
Clover
ebay
EVERQUOTE
Expedia
finleap connect
GoDaddy
GoSpotCheck
H-E-B
Ladbrokes
Mercedes-Benz
Microsoft
NORDSTROM
OfferUp
Ottawa
planet.
STRAVA
SUBSPACE
transit
Walmart
XBOX

and many more...

How does Linkerd work?



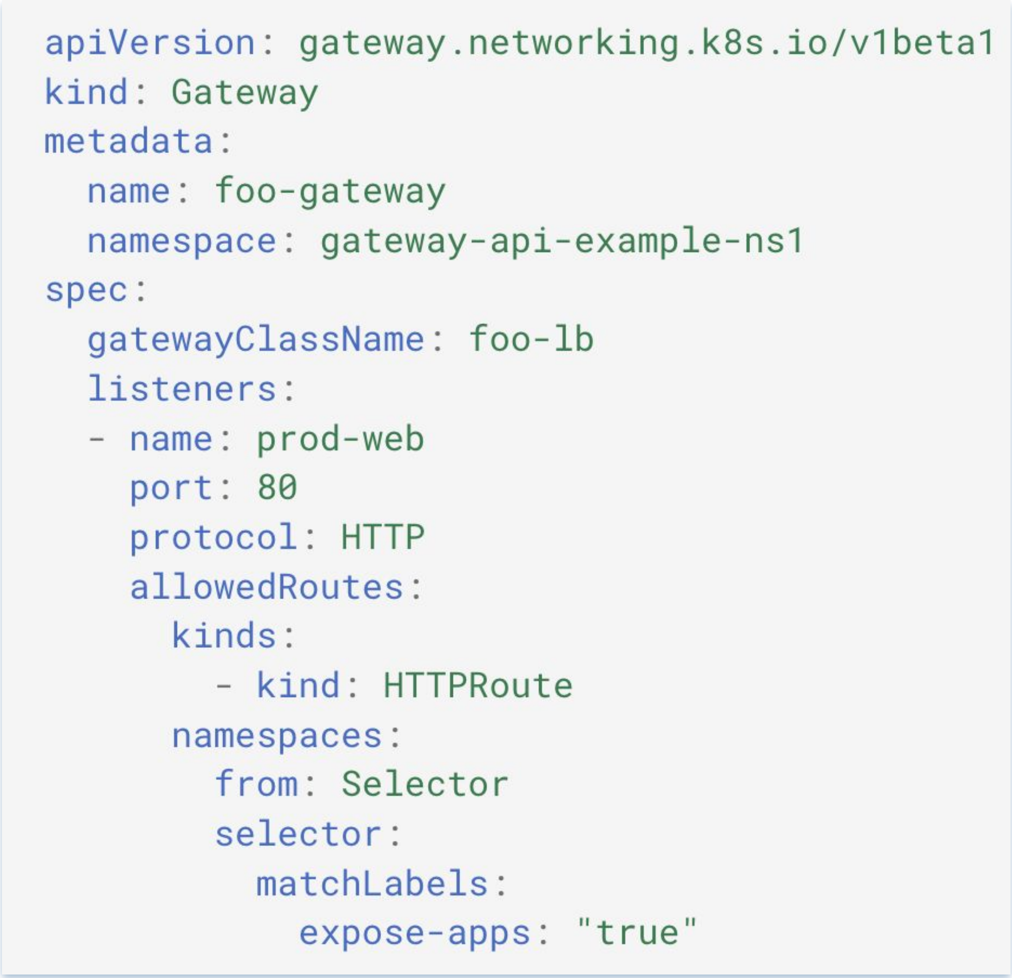
Gateway API: Networking Model



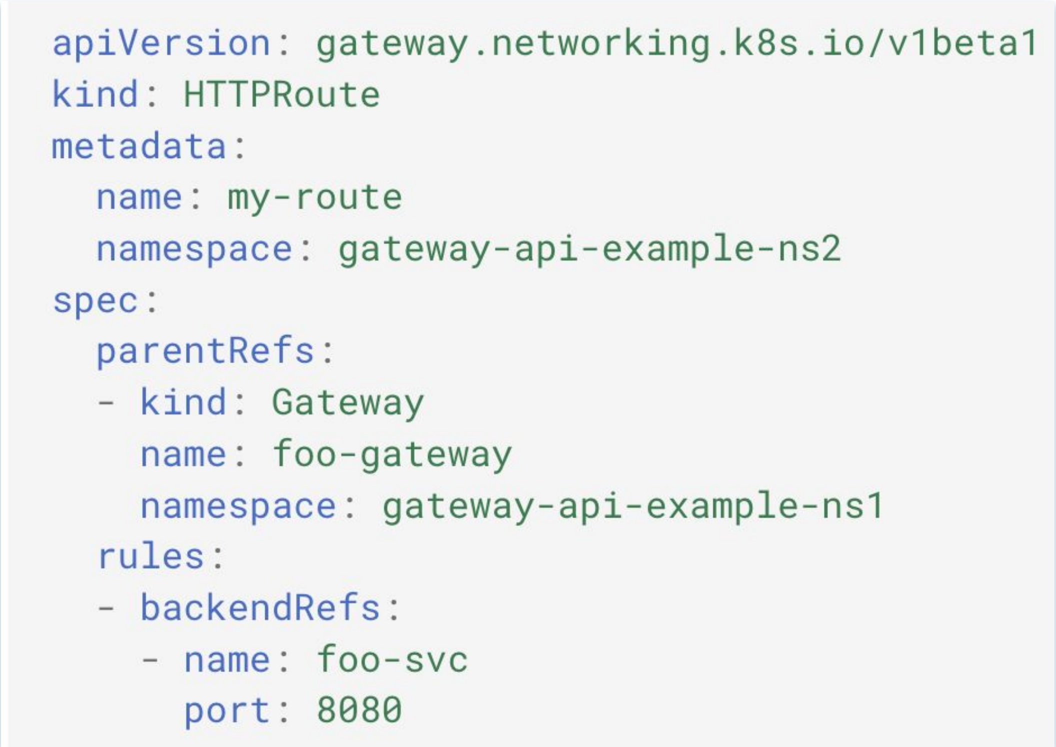
A modern set of APIs for deploying L4 and L7 routing in Kubernetes

Designed to be generic, expressive, extensible, and role-oriented

Gateway API: Resources

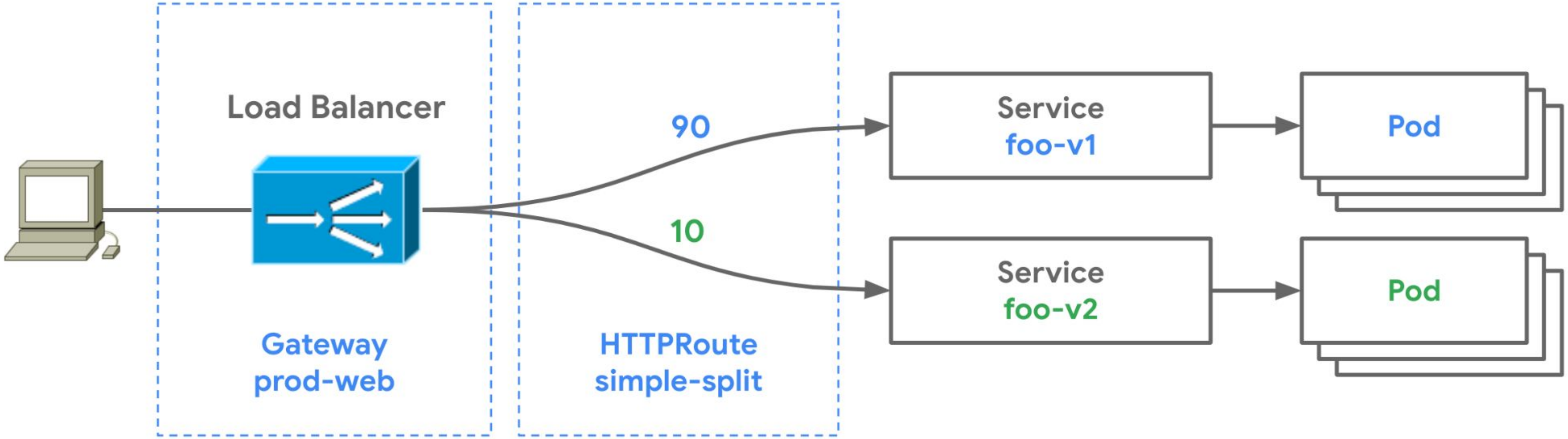


```yaml
apiVersion: gateway.networking.k8s.io/v1beta1
kind: Gateway
metadata:
  name: foo-gateway
  namespace: gateway-api-example-ns1
spec:
  gatewayClassName: foo-lb
  listeners:
  - name: prod-web
    port: 80
    protocol: HTTP
    allowedRoutes:
      kinds:
        - kind: HTTPRoute
      namespaces:
        from: Selector
        selector:
          matchLabels:
            expose-apps: "true"
```

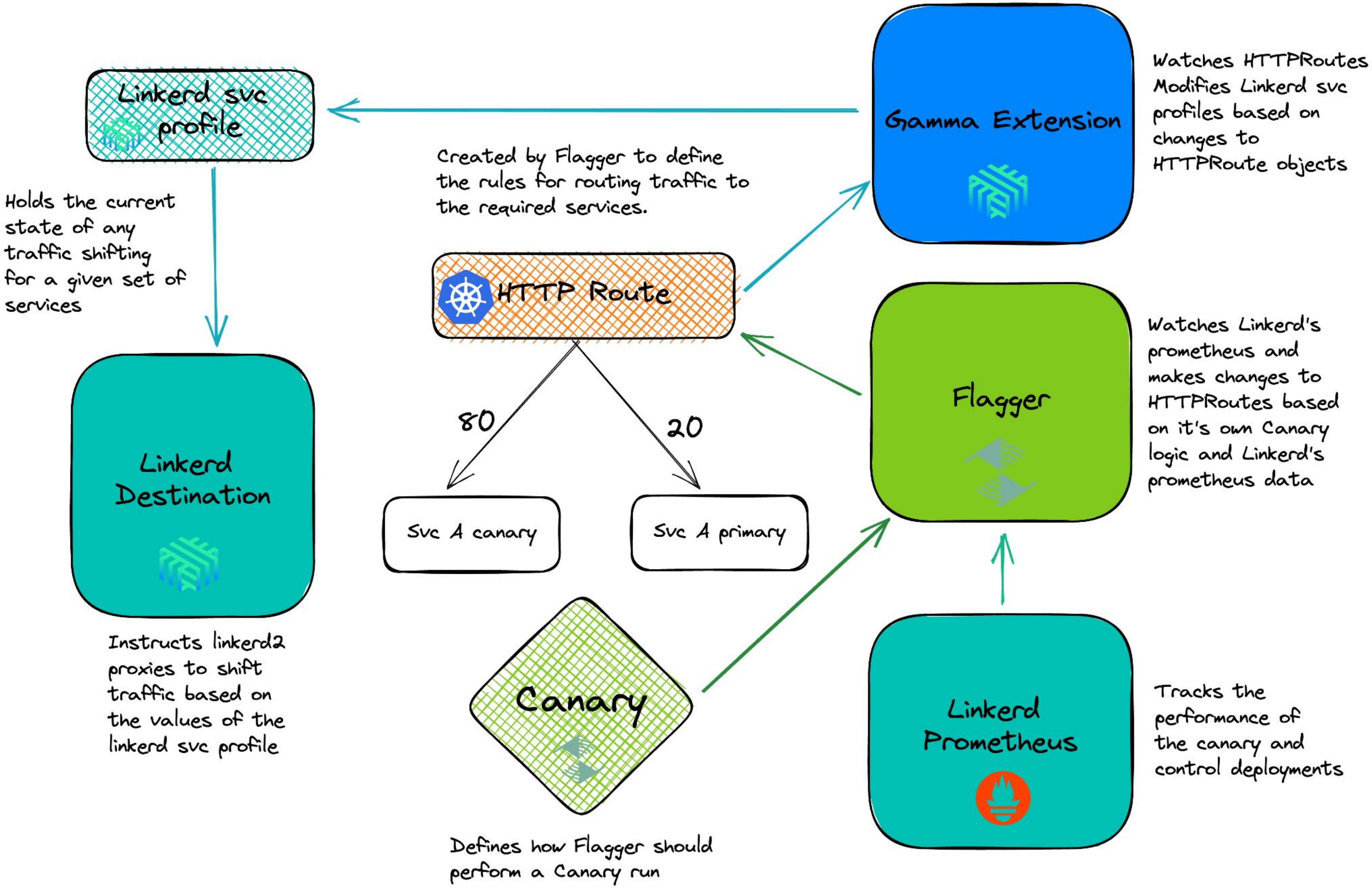


```yaml
apiVersion: gateway.networking.k8s.io/v1beta1
kind: HTTPRoute
metadata:
  name: my-route
  namespace: gateway-api-example-ns2
spec:
  parentRefs:
  - kind: Gateway
    name: foo-gateway
    namespace: gateway-api-example-ns1
  rules:
  - backendRefs:
    - name: foo-svc
      port: 8080
```

Gateway API: Traffic Splitting



How does it all fit?


Linkerd svc profile
Gamma Extension
Watches HTTPRoutes Modifies Linkerd svc profiles based on changes to HTTPRoute objects
Created by Flagger to define the rules for routing traffic to the required services.
Holds the current state of any traffic shifting for a given set of services
HTTP Route
Flagger
Watches Linkerd's prometheus and makes changes to HTTPRoutes based on it's own Canary logic and Linkerd's prometheus data
80
20
Linkerd Destination
Svc A canary
Svc A primary
Instructs linkerd2 proxies to shift traffic based on the values of the linkerd svc profile
Canary
Linkerd Prometheus
Tracks the performance of the canary and control deployments
Defines how Flagger should perform a Canary run

DEMO TIME

Please scan the QR Code above to leave feedback on this session