Service Mesh Specifications

Why They Matter in Your Deployment

Lee Calcote
Founder, Layer5

Kush Trivedi
Maintainer, Layer5

KubeCon CloudNativeCon North America 2020 Virtual

Lee Calcote

cloud native and its management

@lcalcote

layer5.io

github.com/leecalcote

calcotestudios.com/talks

linkedin.com/in/leecalcote

Kush Trivedi

@kush_1814

Layer5, Maintainer

LAYER5

THE SERVICE MESH COMMUNITY

Join the community

 # slack.layer5.io

Third step in Cloud Native journey

Announced

7 years ago
(Mar 2013)

Container

v1.0

5.5 years
(Jun 2014)

Third step in Cloud Native journey

Announced	7 years ago (Mar 2013)	5.5 years ago (Jun 2014)
	 **Container**	 **Orchestrator**
v1.0	5.5 years (Jun 2014)	4.5 years (Jul 2015)

Third step in Cloud Native journey

KubeCon | CloudNativeCon North America 2020 Virtual

Announced	7 years ago (Mar 2013)	5.5 years ago (Jun 2014)	4 years ago (Feb 2016)
	 **Container**	 **Orchestrator**	 **Mesh**
v1.0	5.5 years (Jun 2014)	4.5 years (Jul 2015)	3 years (Apr 2017)

LAYER5

layer5.io/landscape

It's meshy out there.

Strengths of Service Mesh Implementations

KubeCon CloudNativeCon North America 2020 Virtual

Linkerd	Istio	Consul	NGINX Service Mesh	Network Service Mesh
Time to Value, Performance	Powerful Feature Set, Extensibility	Support for Non-Kubernetes Workloads	Interoperability with Existing Ingresses	Layer 2 and Layer 3 Functions

a sample

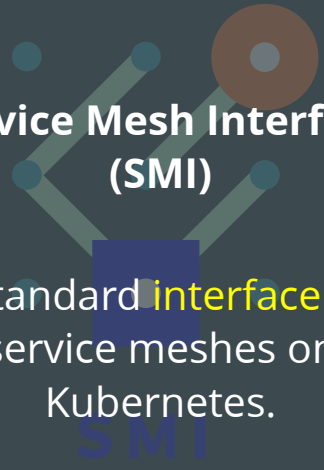
Different tools for different use cases

LAYER5

Service mesh abstractions

to the rescue

KubeCon CloudNativeCon North America 2020 Virtual



Service Mesh Interface (SMI)

A standard interface for service meshes on Kubernetes.

Multi-Vendor Service Mesh Interoperation (Hamlet)

A set of API standards for enabling service mesh federation.

Meshery is interoperable with these abstractions.

LAYER5

Service mesh abstractions

to the rescue

KubeCon CloudNativeCon North America 2020 Virtual

Service Mesh Interface (SMI)

A standard interface for service meshes on Kubernetes.

Service Mesh Performance Specification (SMPS)

A format for describing and capturing service mesh performance.

Multi-Vendor Service Mesh Interoperation (Hamlet)

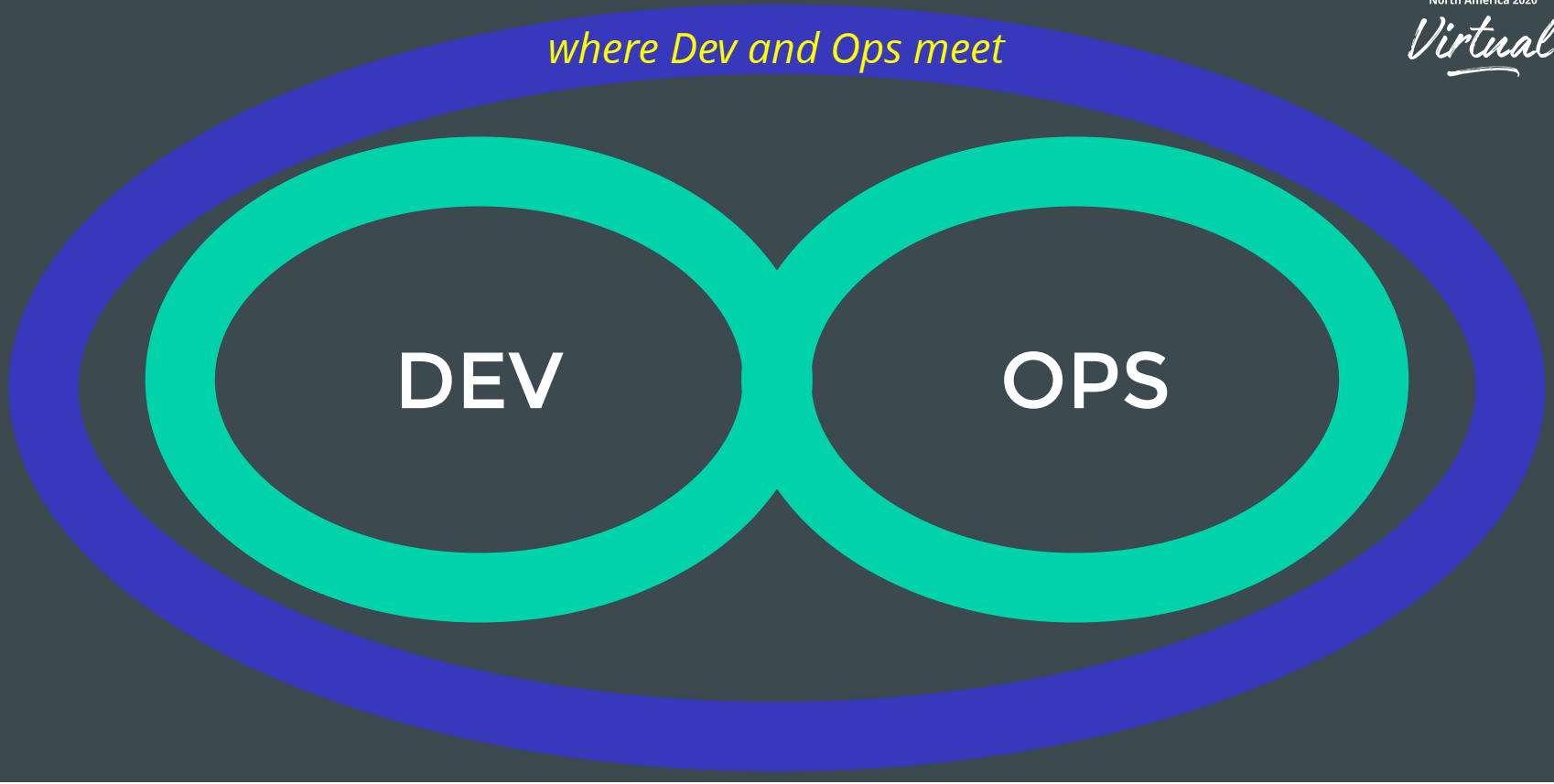
A set of API standards for enabling service mesh federation.

Meshery is interoperable with these abstractions.

LAYER5

Decoupling at Layer 5

where Dev and Ops meet

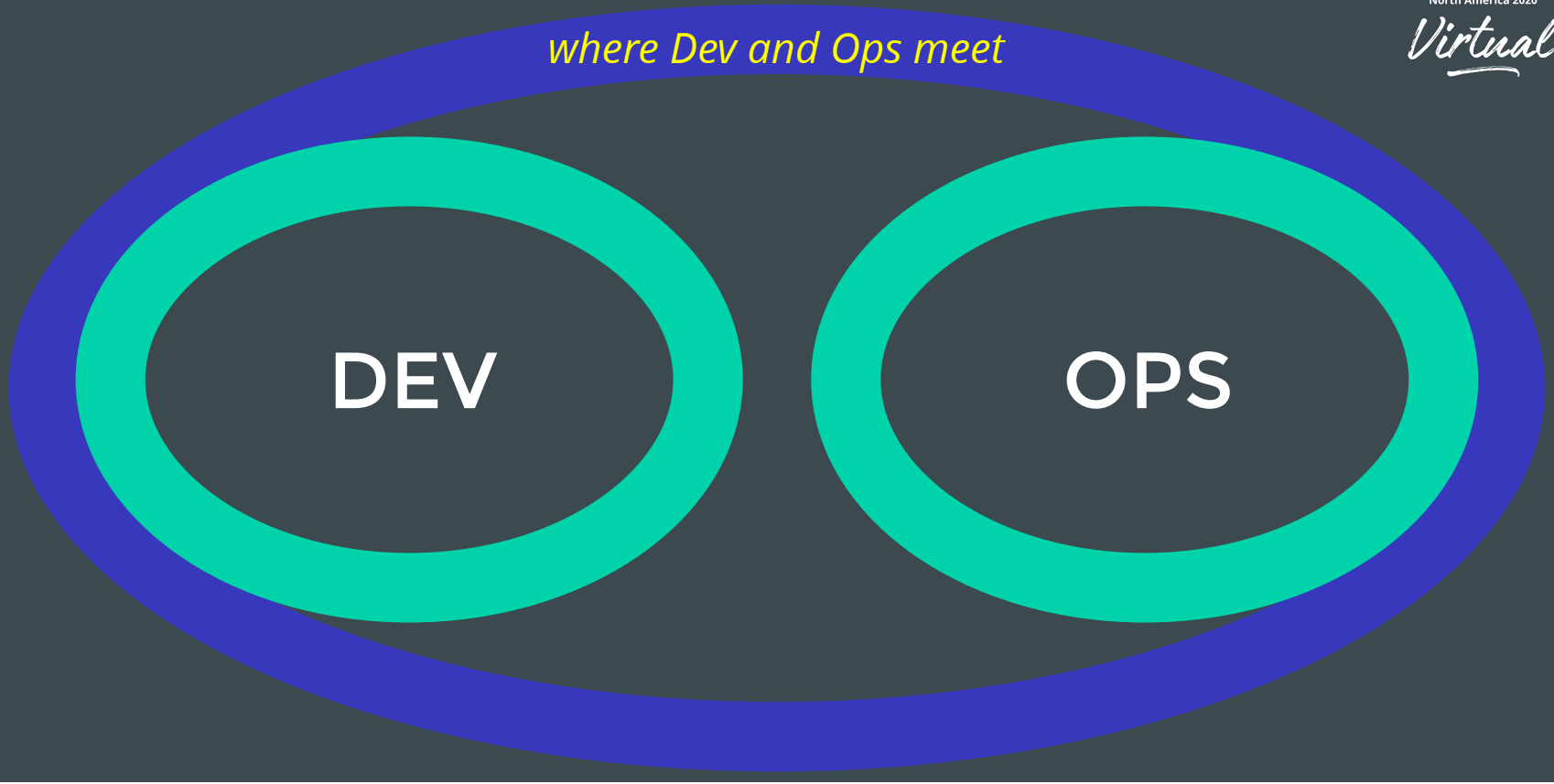


Empowered and independent teams can iterate faster

LAYER5

Decoupling at Layer 5

KubeCon | CloudNativeCon
North America 2020
Virtual

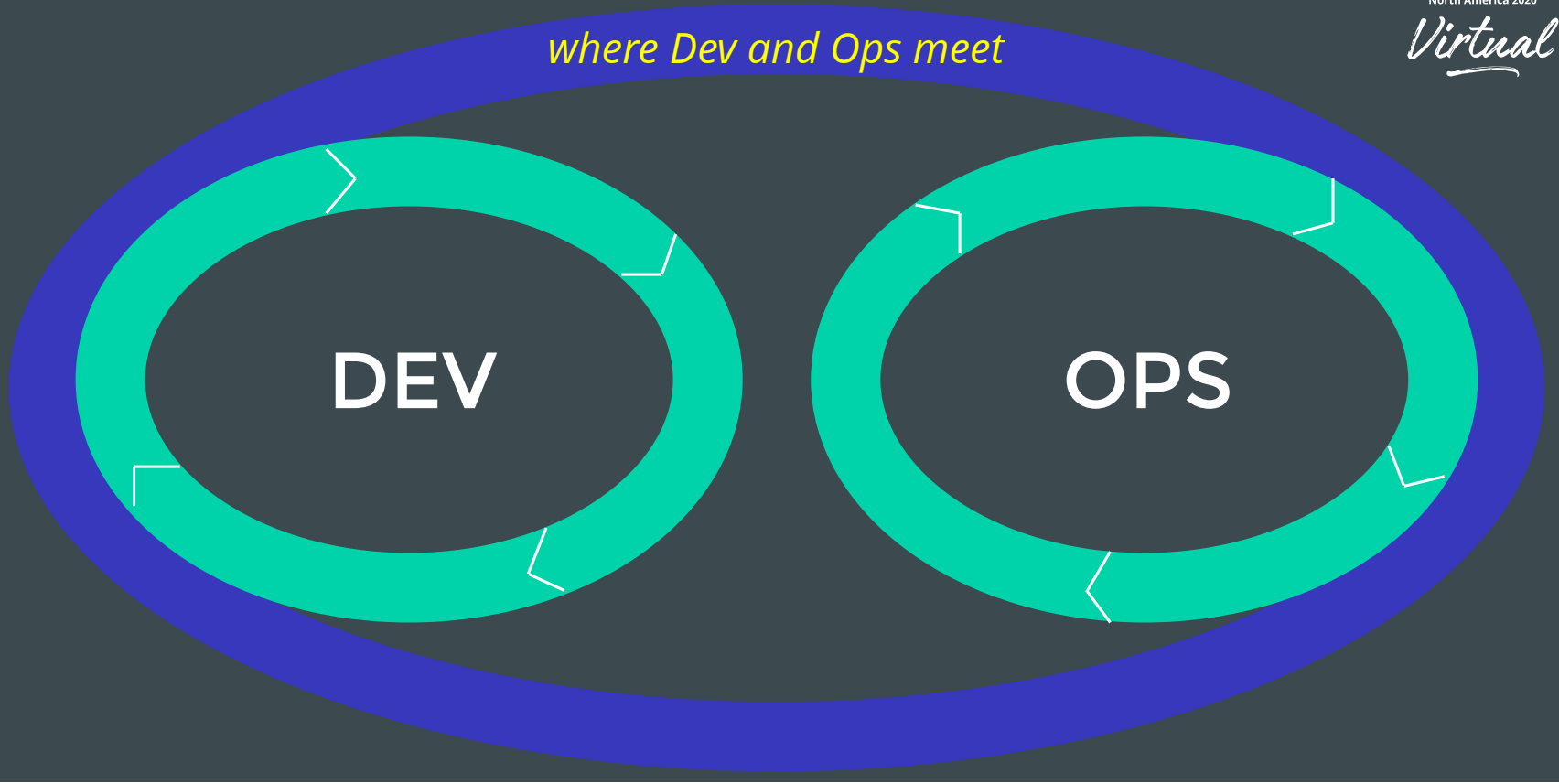


Empowered and independent teams can iterate faster

LAYER5

Decoupling at Layer 5

where Dev and Ops meet



Empowered and independent teams can iterate faster

MESHERY

THE MULTI-MESH MANAGER

MESHERY

THE MULTI-MESH MANAGER

Service Mesh Interface
(SMI)

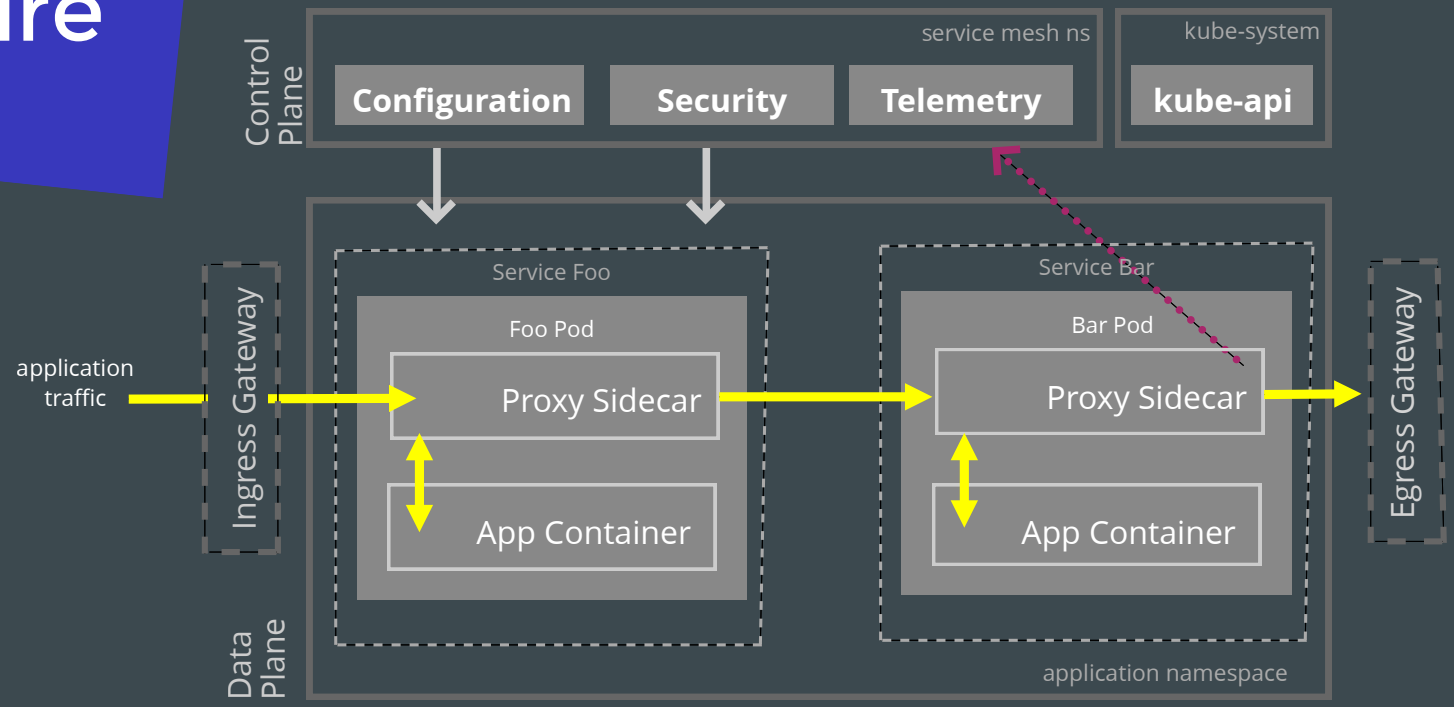
SMP
Service Mesh Performance
(SMP)

COMMUNITYBRIDGE

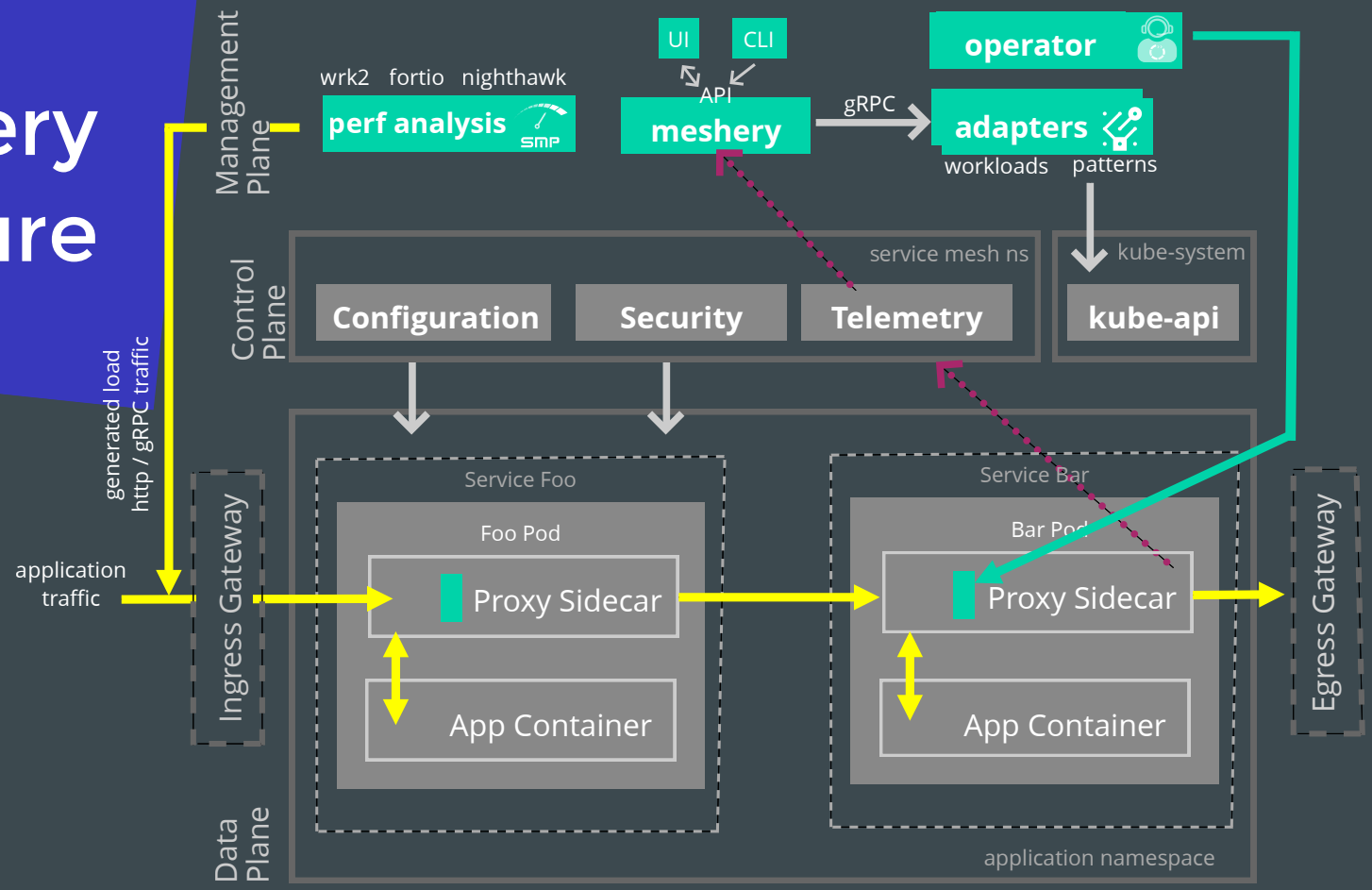
CLOUD NATIVE
Landscape

Google
Summer of Code

Meshery Architecture



Meshery Architecture



Out-of-band telemetry propagation

Control flow

Application traffic

Meshery WASM Filter

@mesheryio

Meshery Architecture

Management Plane

wrk2 fortio nighthawk

perf analysis

SMP

UI CLI

API

meshery

gRPC

operator

adapters

workloads patterns

Control Plane

service mesh ns

Configuration Security Telemetry

kube-system

kube-api

generated load

http / gRPC traffic

application traffic

Ingress Gateway

Service Foo

Foo Pod

Proxy Sidecar

App Container

Service Bar

Bar Pod

Proxy Sidecar

App Container

Egress Gateway

Data Plane

application namespace

Out-of-band telemetry propagation

Control flow

Application traffic

Meshery WASM Filter

@mesheryio

Service Mesh Interface

Service Mesh Interface aims to provide:

- Standard interface for service mesh on Kubernetes
- Basic feature set for most common mesh use cases
- Extensible to support new features
- Space for the ecosystem to innovate

APIs:

Traffic Split
Traffic Specs
Traffic Metrics
Traffic Access Control

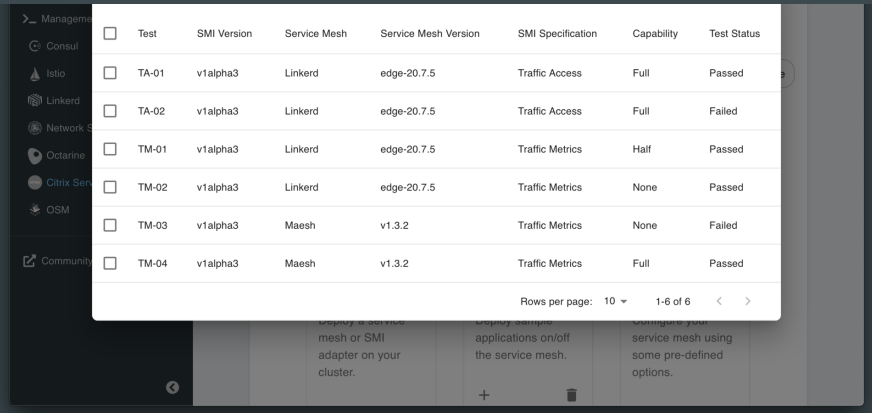
Four API specifications

LAYER5

Service Mesh Ir

✔ Learn Layer5 sample ap
used for validating test as

✔ Defines compliant behavior.
✔ Produces compatibility matrix.
✔ Ensures provenance of results.
✔ Runs a set of conformance tests.
✔ Built into participating service mesh's release pipeline.

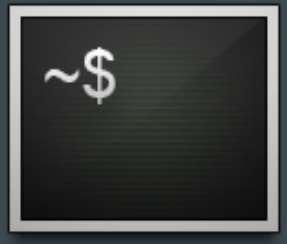


Operate and upgrade with confirmation of SMI compatibility

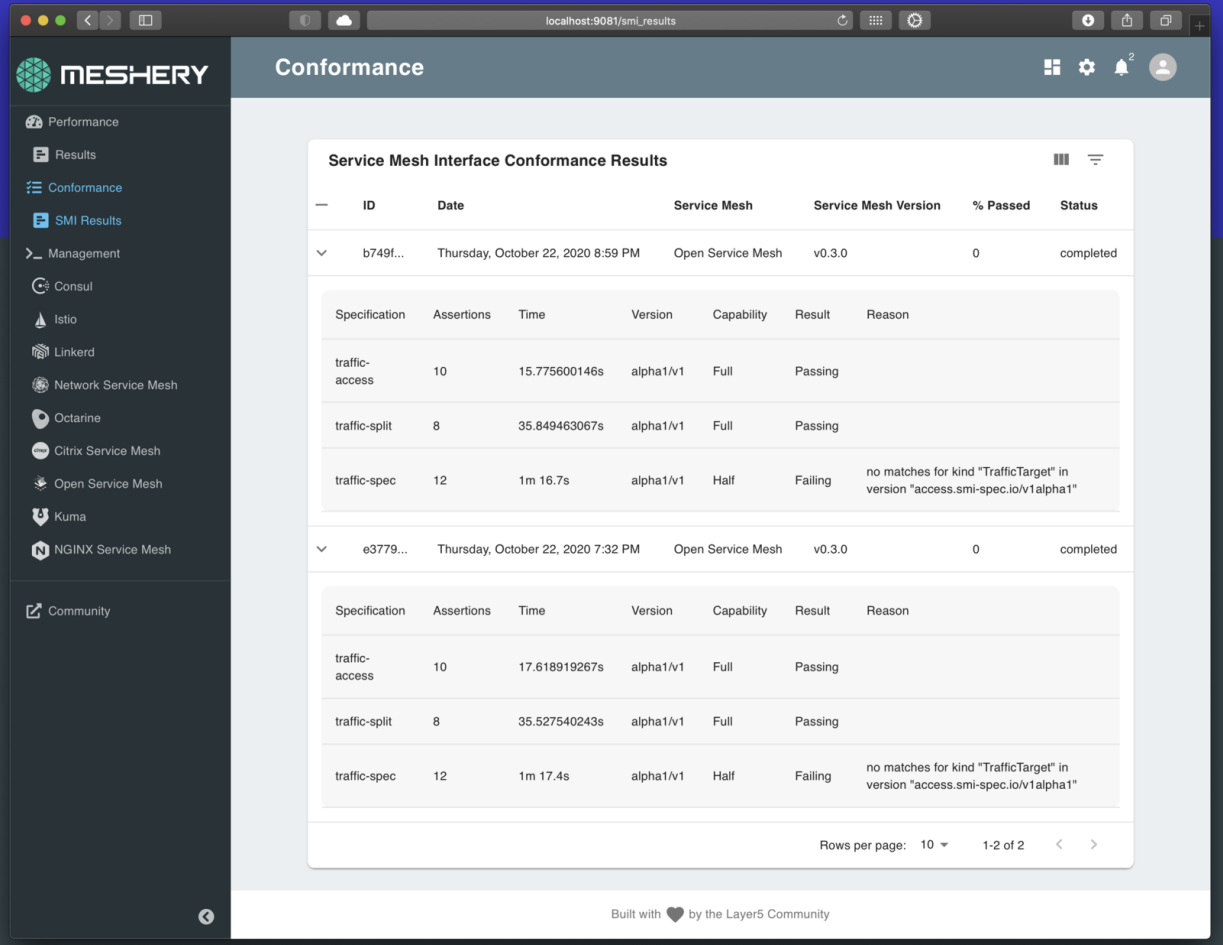
SMI

LAYER5

SMI Conformance



Demo



SMI Conformance application - github.com/layer5io/learn-layer5

Service Mesh Performance

Directly provides:

- a vendor neutral specification for capturing details of infrastructure capacity, service mesh configuration, and workload metadata.

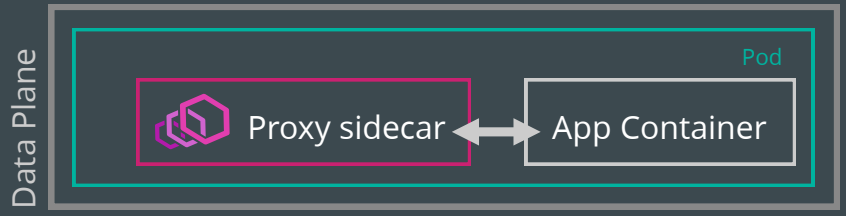
Indirectly facilitates:

- benchmarking of service mesh performance
- exchange of performance information from system-to-system / mesh-to-mesh
- apples-to-apples performance comparisons of service mesh deployments.
- MeshMark - a universal performance index to gauge a service mesh's efficiency against deployments in other organizations' environments

An optimization game

with many variables

KubeCon CloudNativeCon North America 2020 Virtual



Data plane performance depends on many factors, for example:

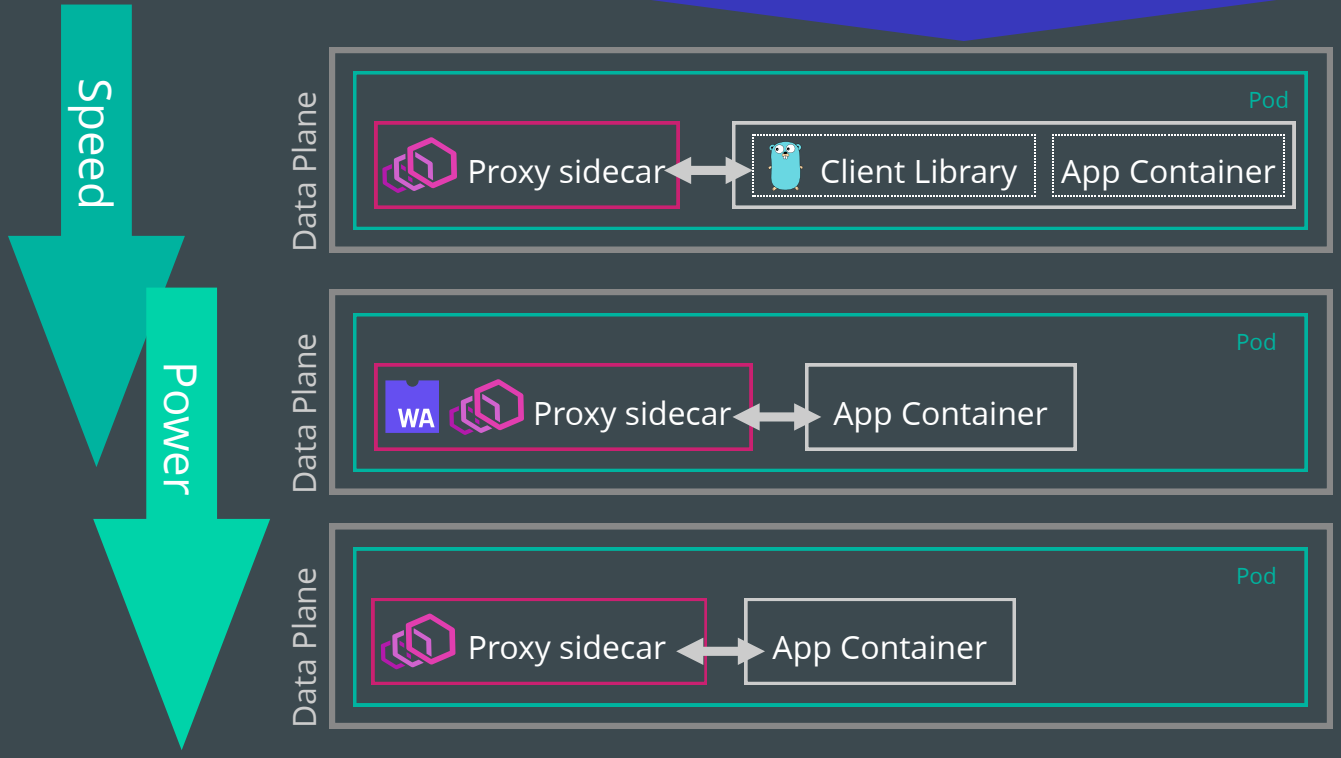
- Number of client connections
- Target request rate
- Request size and Response size
- Number of proxy worker threads
- Protocol
- CPU cores
- Number and types of proxy filters

Latency, throughput, and the proxies' CPU and memory consumption affected by these factors

LAYER5

Comparing types of Data Plane filtering

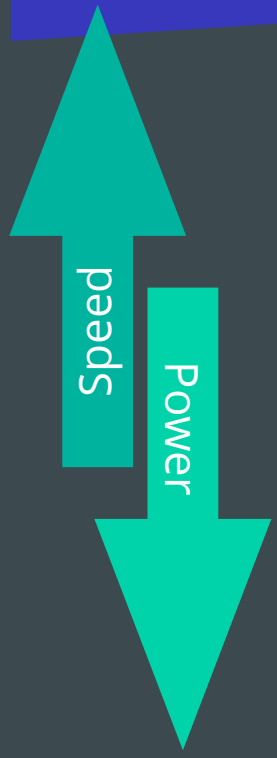
KubeCon CloudNativeCon North America 2020 Virtual

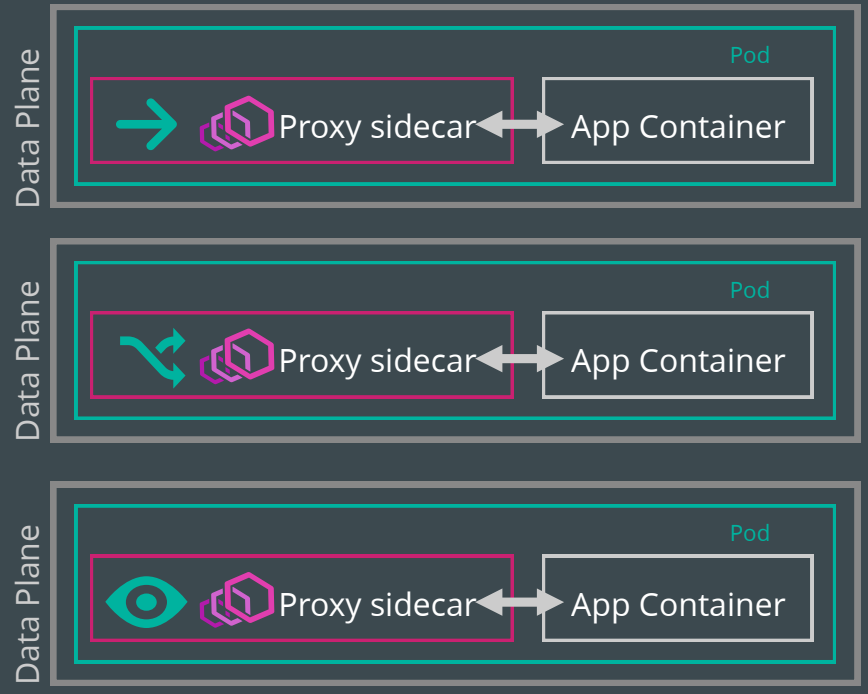


Comparing approaches to data plane filtering

LAYER5

Comparing types of functions





Path-based routing

Round robin
load balancing

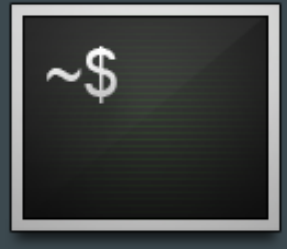
Context-based routing

Understanding the trade-off between power and speed

LAYER5

Service Mesh Performance

KubeCon CloudNativeCon North America 2020 Virtual



Demo

Performance Testing Best Practices

KubeCon | CloudNativeCon North America 2020 Virtual

Use Meshery's powerful performance management features

- easily reproduce tests
- persist test results
- use different load generators
- baseline and compare over time
- test your workloads on and off the mesh
- tweak configurations and try again
- manage 9 different service meshes and counting...

Hamlet

service catalog federation

KubeCon CloudNativeCon North America 2020 Virtual

Hamlet specifies a set of API standards for enabling service mesh federation.

Specifications:

1. **Federated Resource Discovery API**

   API to authenticate and securely distribute resources between federated service meshes.

2. **Federated Service Discovery API**

   API to discover, reach, authenticate, and securely communicate with federated services.

https://github.com/vmware/hamlet

LAYER5

Service Mesh Patterns by the Book

KubeCon | CloudNativeCon North America 2020 Virtual

List of service mesh patterns

LAYER5

THE SERVICE MESH COMMUNITY

Lee Calcote

@lcalcote

Kush Trivedi

@kush_1814

Join the community

slack.layer5.io

Service Mesh Functionality

KubeCon CloudNativeCon North America 2020 Virtual

Traffic Control

content-based traffic steering

Resiliency

control over chaos

Observability

what gets people hooked on service metrics

Security

identity and policy

Expect more from your infrastructure

LAYER5

Why use a Service Mesh?

to avoid...

- Bloated service (application) code
- Duplicating work to make services production-ready
  - Load balancing, auto scaling, rate limiting, traffic routing...
- Inconsistency across services
  - Retry, tls, failover, deadlines, cancellation, etc., for each language, framework
  - Siloed implementations lead to fragmented, non-uniform policy application and difficult debugging
- Diffusing responsibility of service management

Service Mesh Architectures

Service Mesh Architecture

Management Plane

- Provides multi-mesh federation, backend system integration, expanded policy and governance, continuous delivery integration, workflow, chaos engineering, configuration and performance management.

Control Plane

- Provides policy, configuration, and platform integration.
- Takes a set of isolated stateless sidecar proxies and turns them into a service mesh.
- Does not touch any packets/requests in the data path.

Ingress Gateway

Data Plane

- Touches every packet/request in the system.
- Responsible for service discovery, health checking, routing, load balancing, authentication, authorization, and observability.

Egress Gateway

You need a management plane.

LAYER5

Image Hub

a sample app

KubeCon | CloudNativeCon North America 2020 Virtual

Functionality	In the app	In the filter
User / Token	✓	✗
Subscription Plans	✓	✗
Plan Enforcement	✗	✓

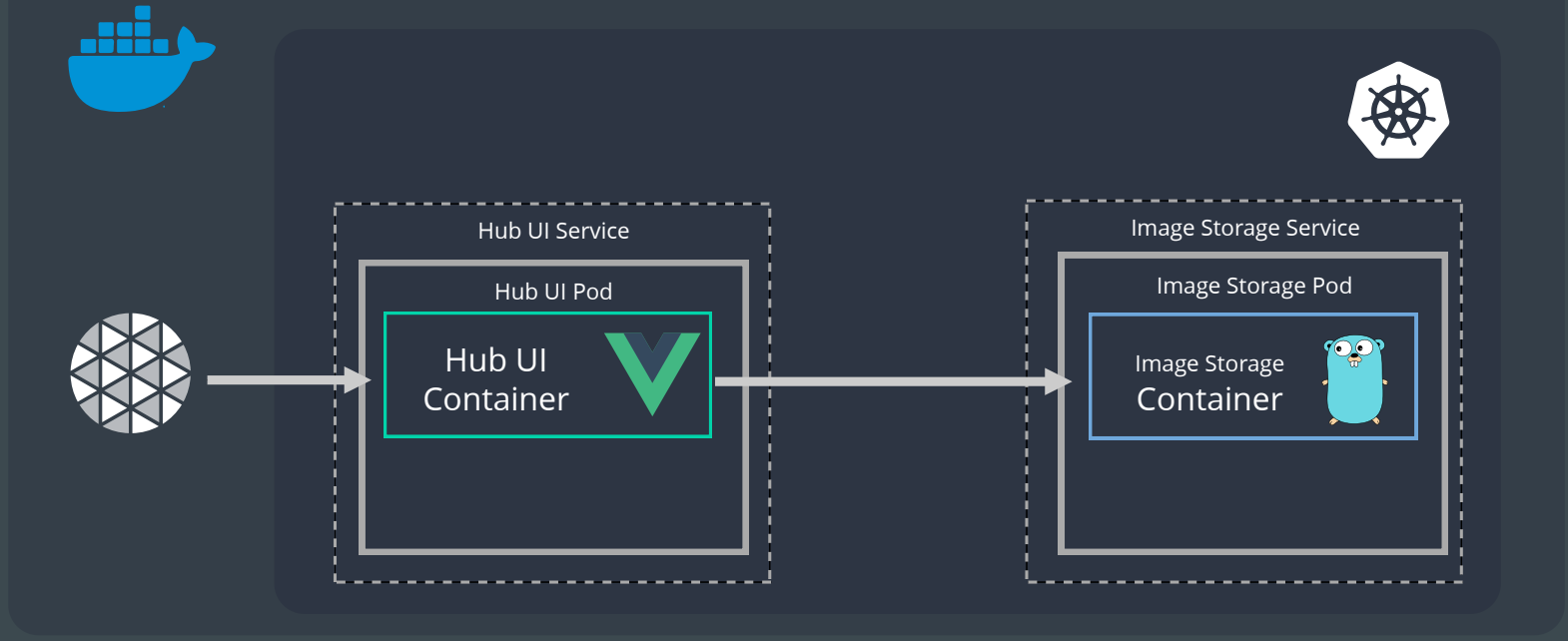
Two application containers

github.com/layer5io/image-hub

LAYER5

Image Hub on Docker Desktop

KubeCon CloudNativeCon Europe 2020 Virtual



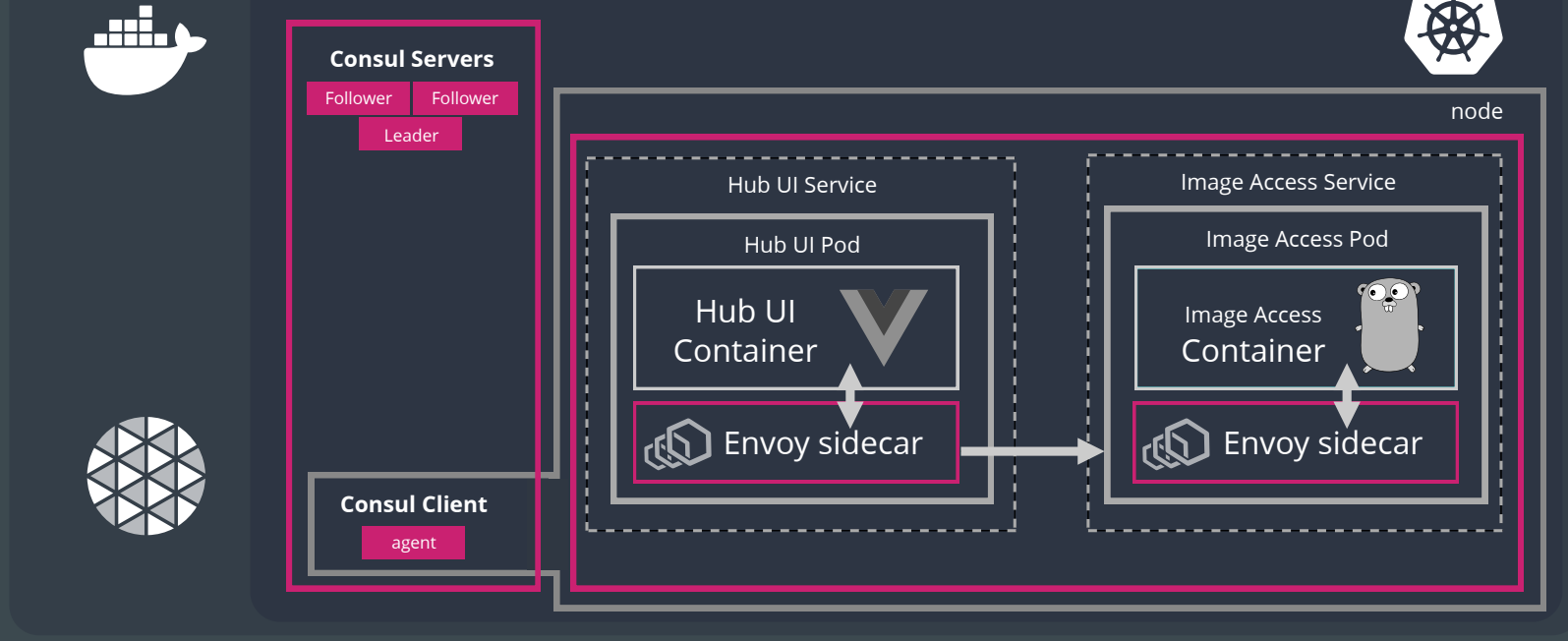
github.com/layer5io/image-hub

LAYER5

Image Hub on a Service Mesh

with Consul

KubeCon CloudNativeCon Europe 2020 Virtual

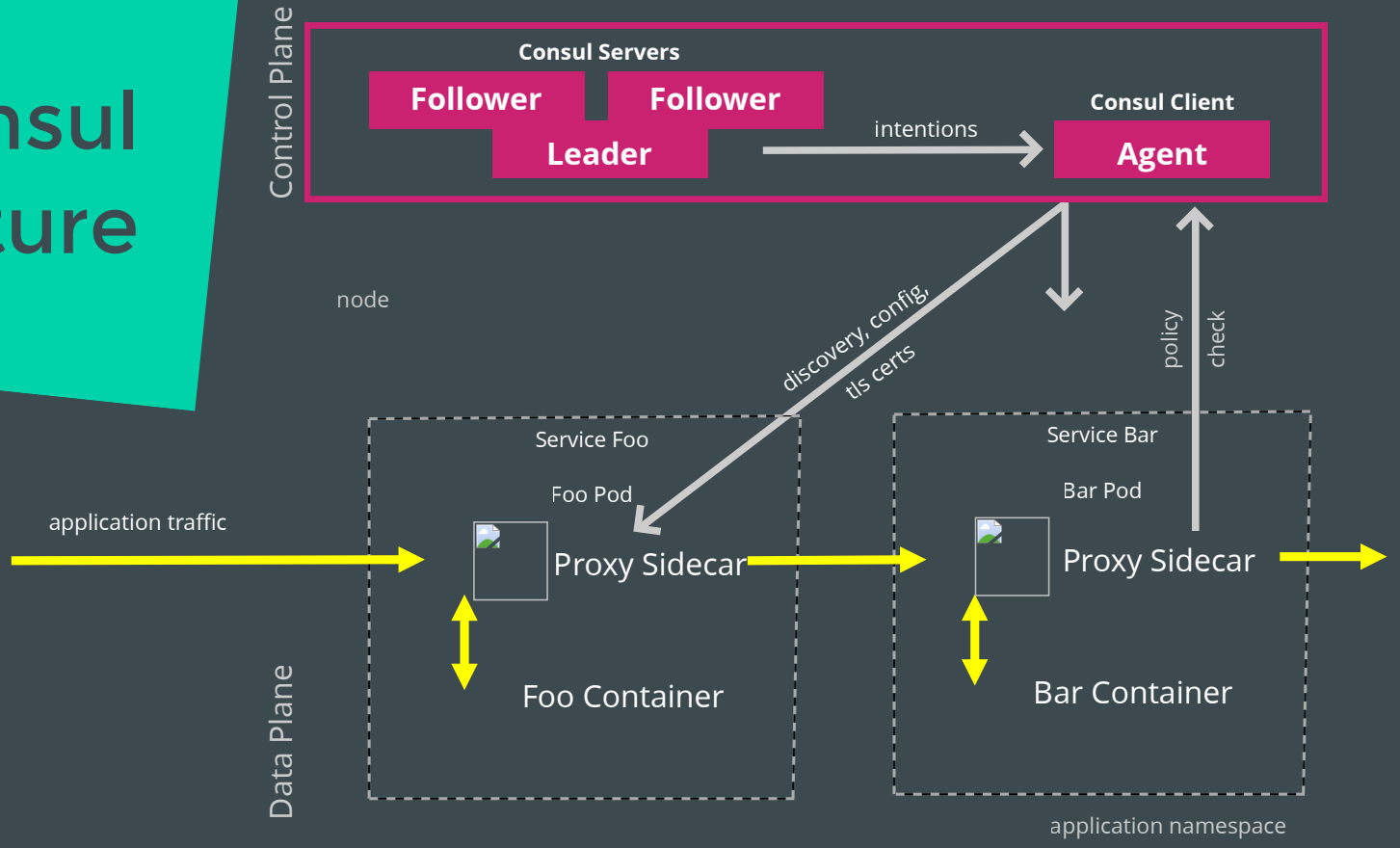


github.com/layer5io/image-hub

LAYER5

Consul Architecture

Consul Architecture



Consul Architecture

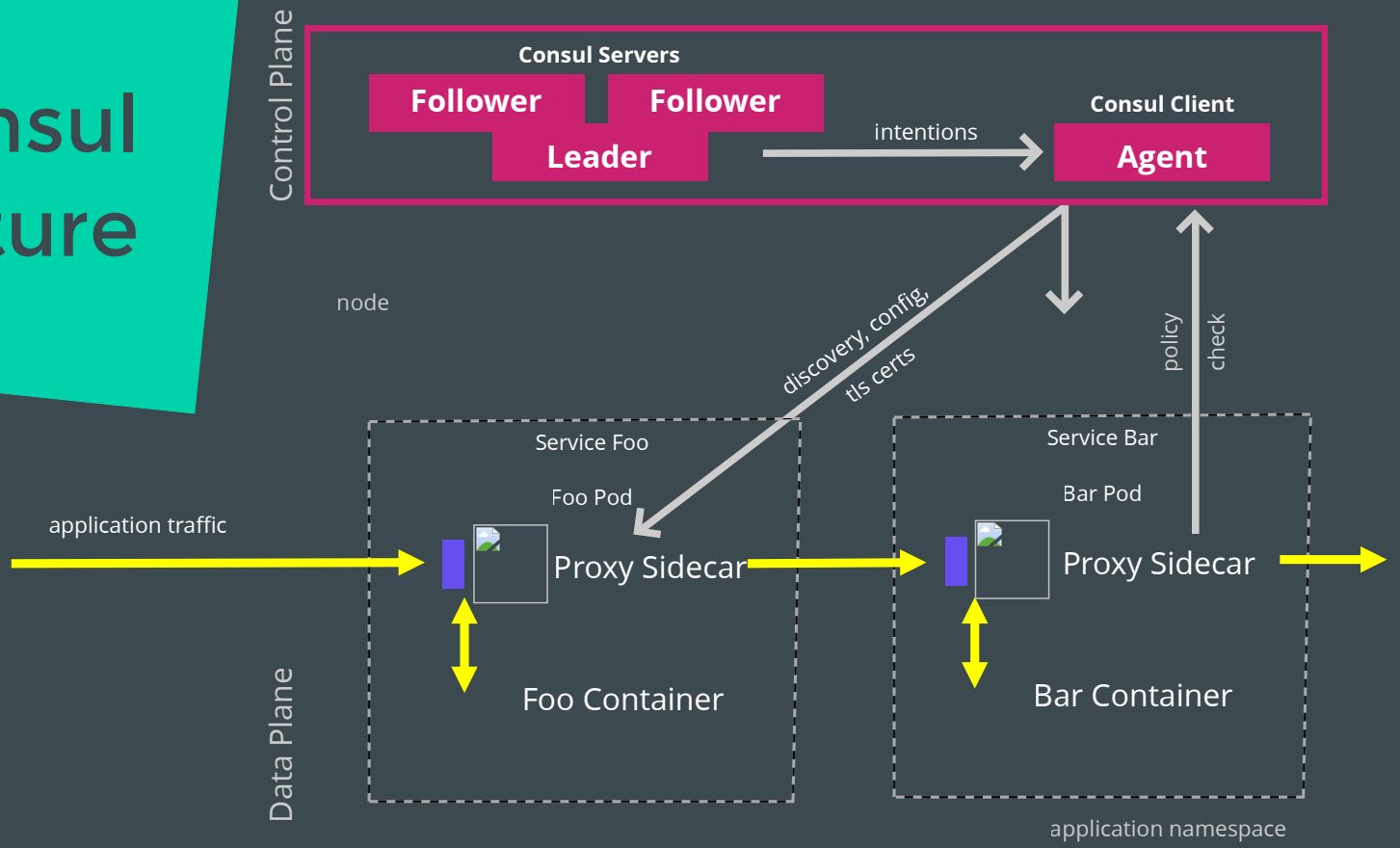
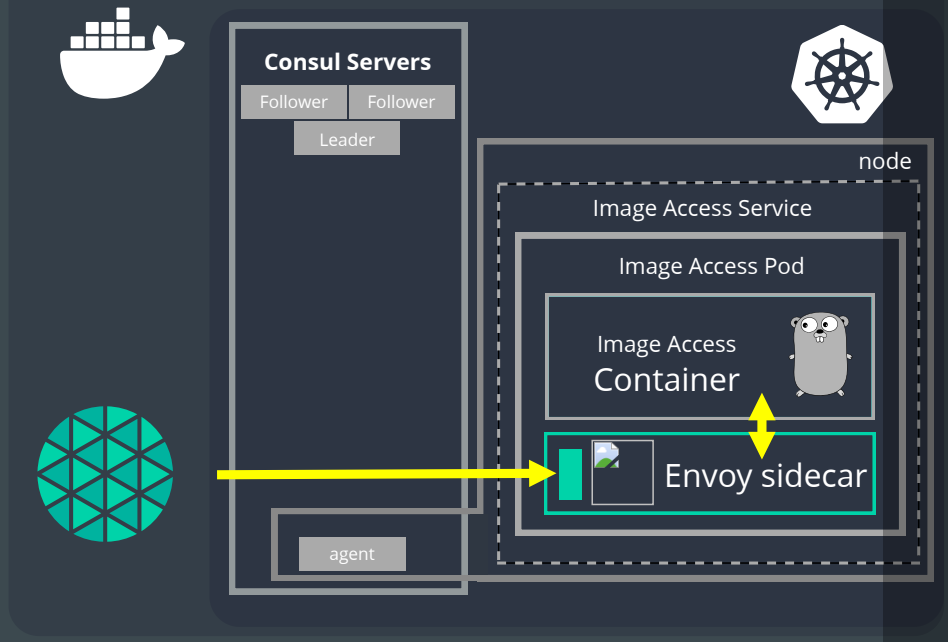


Image Hub on Consul

with a Rust-based WASM filter

KubeCon CloudNativeCon Europe 2020 Virtual




```yaml
apiVersion: apps/v1
kind: Deployment
spec:
  template:
    metadata:
      labels:
        app: api-v1
      annotations:
        "consul.hashicorp.com/connect-inject": "true"
        "consul.hashicorp.com/service-meta-version": "1"
        "consul.hashicorp.com/service-tags": "v1"
        "consul.hashicorp.com/connect-service-protocol": "http"
        "consul.hashicorp.com/connect-wasm-filter-add_header":
"/filters/optimized.wasm"
    spec:
      containers:
      - name: api
        image: layer5/image-hub-api:latest
```

github.com/layer5io/image-hub

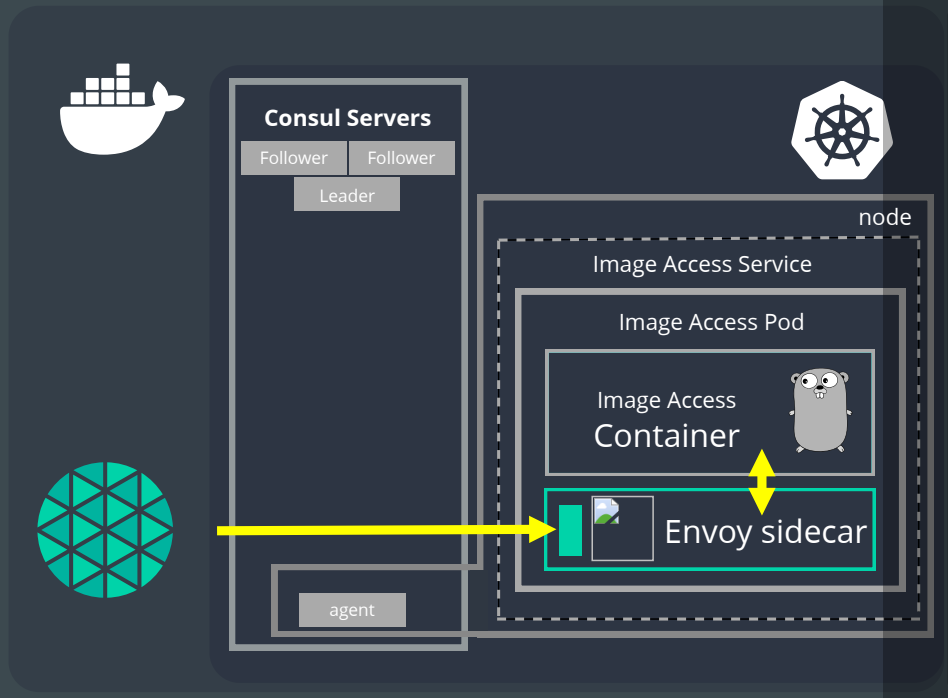
WASM Filter

LAYER5

Image Hub on Consul

with a Rust-based WASM filter

KubeCon CloudNativeCon Europe 2020 Virtual




```yaml
apiVersion: apps/v1
kind: Deployment
spec:
  template:
    metadata:
      labels:
        app: api-v1
      annotations:
        "consul.hashicorp.com/connect-inject": "true"
        "consul.hashicorp.com/service-meta-version": "1"
        "consul.hashicorp.com/service-tags": "v1"
        "consul.hashicorp.com/connect-service-protocol": "http"
        "consul.hashicorp.com/connect-wasm-filter-add_header": "/filters/optimized.wasm"
    spec:
      containers:
      - name: api
        image: layer5/image-hub-api:latest
```

github.com/layer5io/image-hub

WASM Filter

LAYER5

PERFORMANCE

What is WebAssembly?

for the web, malware and beyond

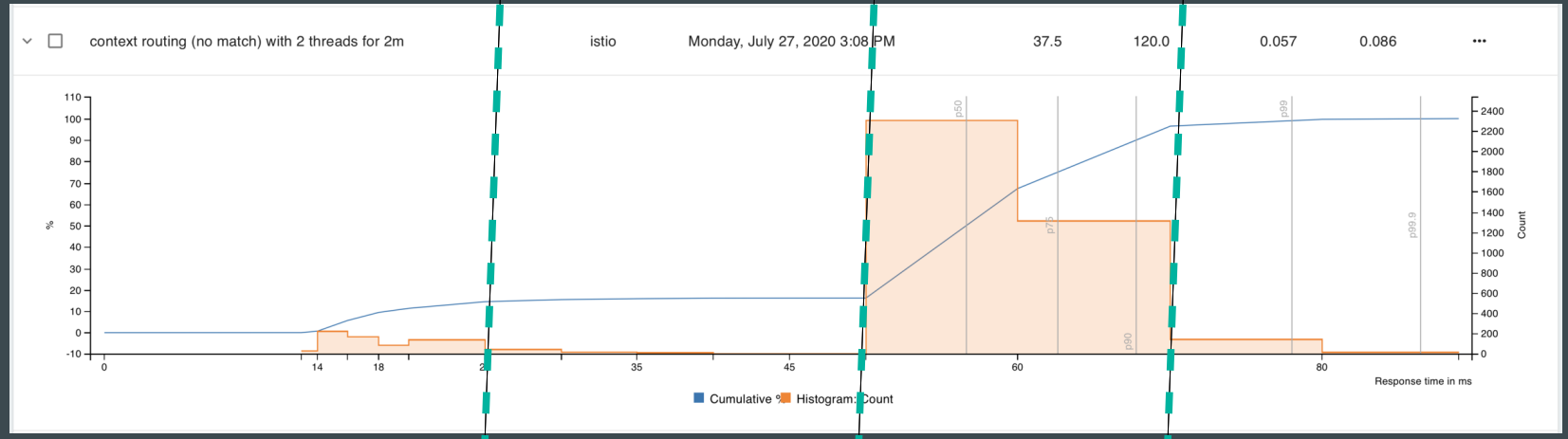
KubeCon CloudNativeCon Europe 2020 Virtual

- A small, fast binary format that promises near-native performance for web applications.
- Most modern browsers support it.
- Safe and sandboxed execution environment.
- Over 40 languages that support WASM as a compilation target.
- Originally used to speed up large web-applications.

Optimizing your average response time

KubeCon CloudNativeCon Europe 2020 Virtual

Bucket 1 | Bucket 2 | Bucket 3 | Bucket 4

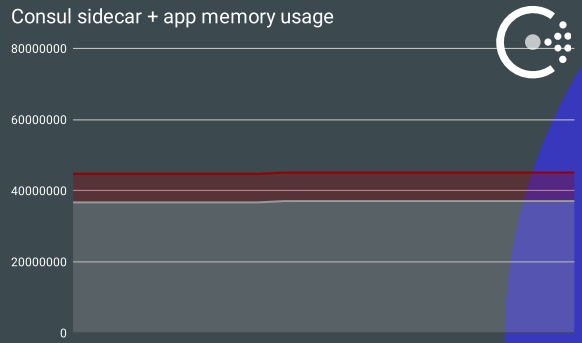


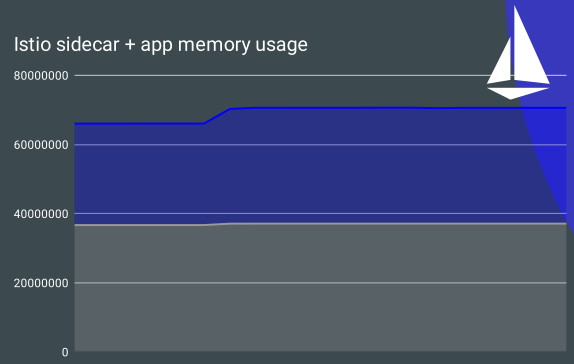
In the presence of Bucket 1...

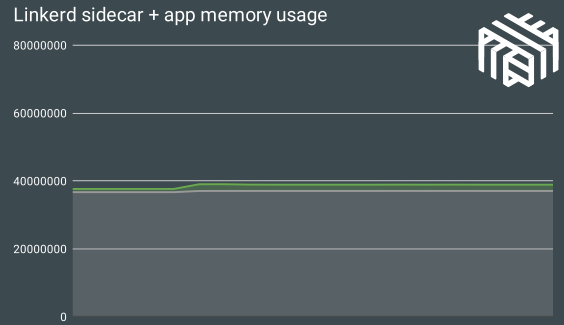
...take your largest segment by count and divide by your number of cores

identifying your optimal configuration for most requests

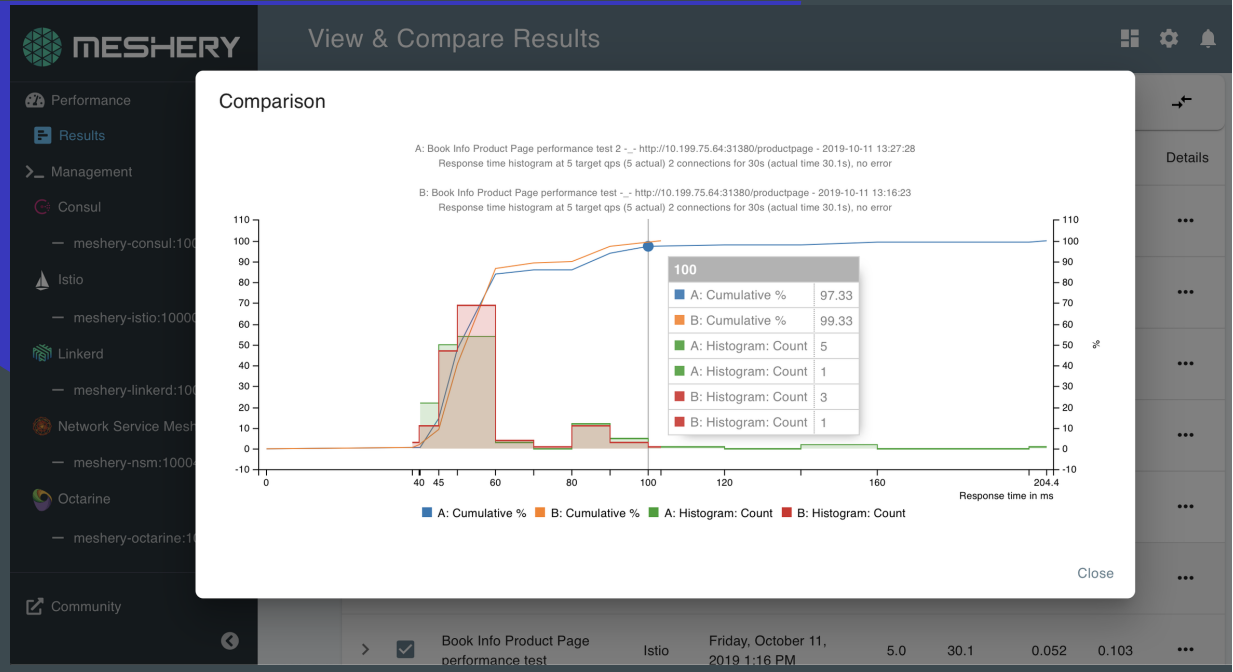
LAYER5







Understand value vs overhead



Service Mesh Architecture

KubeCon CloudNativeCon Europe 2020 Virtual

Ingress Gateway

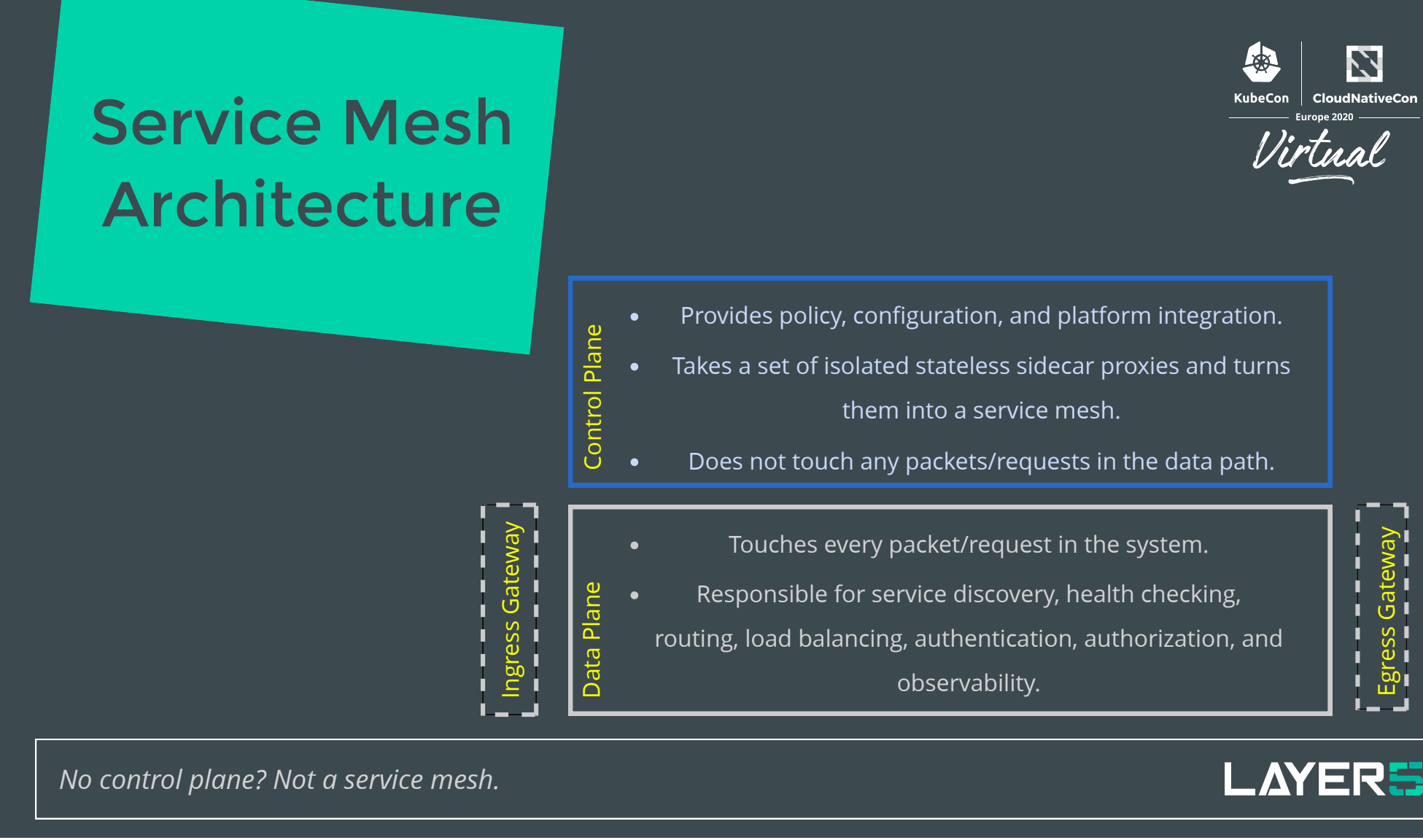
Data Plane

- Touches every packet/request in the system.
- Responsible for service discovery, health checking, routing, load balancing, authentication, authorization, and observability.

Egress Gateway

LAYER5

Service Mesh Architecture



Comparing types of Data Plane filtering

KubeCon CloudNativeCon Europe 2020 Virtual

Speed

Power

Data Plane

Pod

Proxy sidecar

Client Library

App Container

Data Plane

Pod

Proxy sidecar

App Container

envoy

Data Plane

Pod

Proxy sidecar

App Container

WA + envoy

Comparing approaches to data plane filtering

LAYER5