# AWS App Mesh

**Shubha Rao**
Principal Product Manager
Amazon Web Services

AWS re:Invent

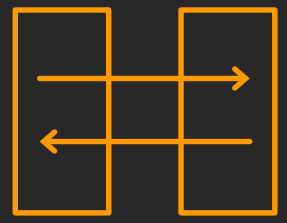


aws

Customers have lots of pieces to operate

What do developers really need to build their applications?

# App Mesh Features



Consistency across teams

Failure visibility and isolation

Fine-grained deployment controls

Authentication and Authorization controls

# Why did we built AWS App Mesh?

**Security, reliability, availability, and scale**

Managed by AWS


**Application focus**

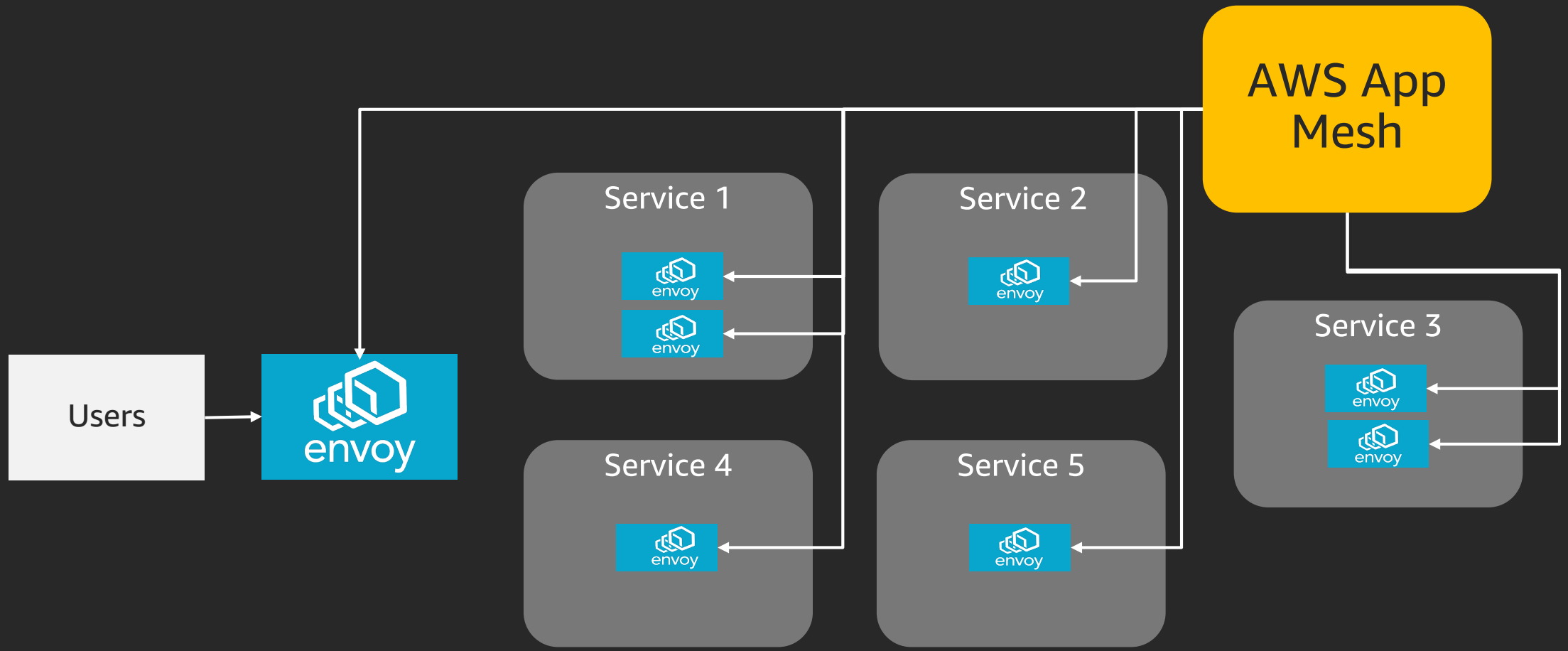
Declarative model for application communication

**Choice: picking the best tool for the job**

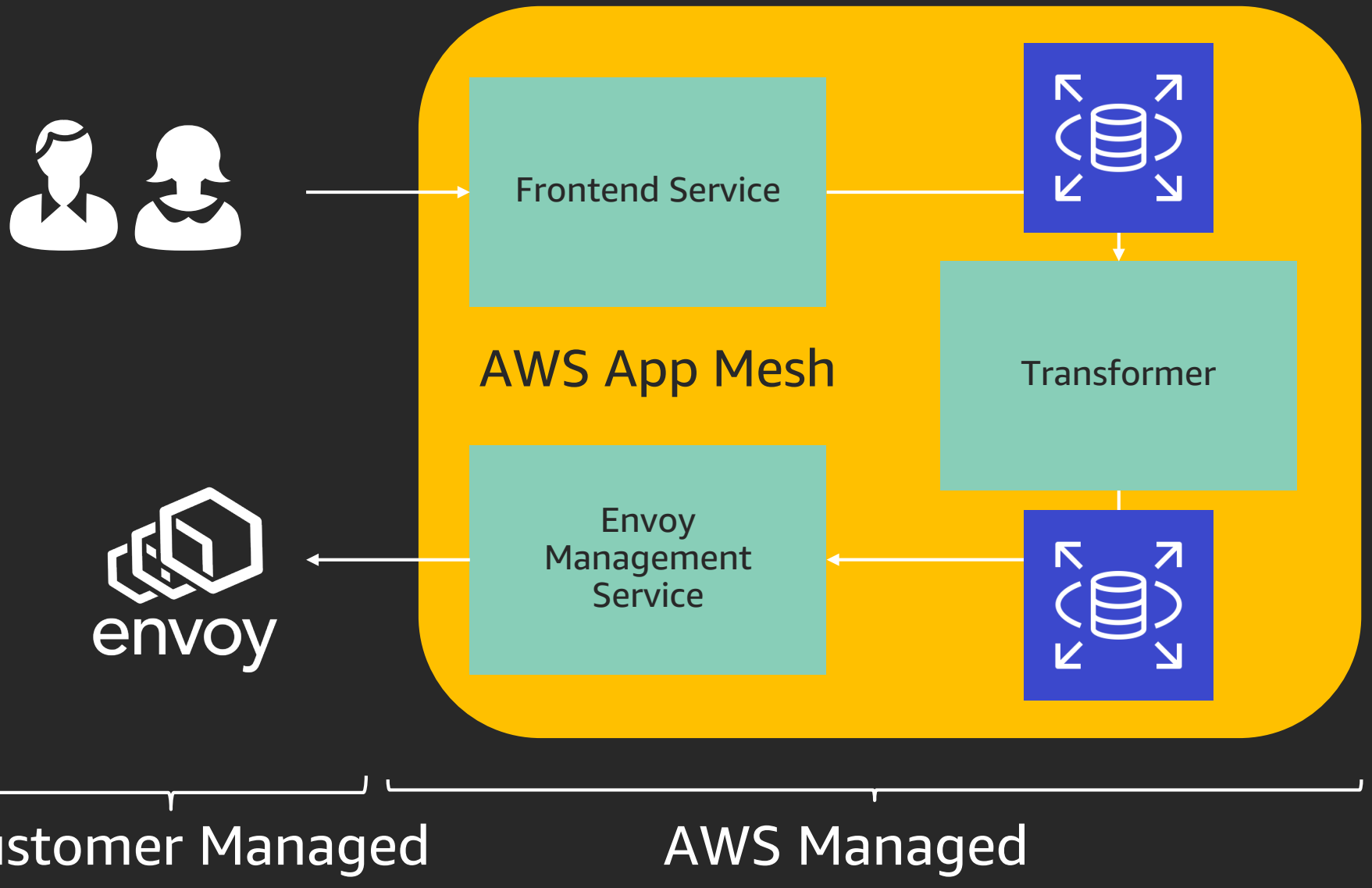
Works with multiple compute options

AWS App Mesh is our first step in building an application-aware network

# AWS App Mesh manages your proxy configuration



# High-Level App Mesh Control Plane Architecture



# App Mesh Configuration

**Mesh**

Application

**Virtual Gateway**

Ingress rules

**Virtual Service**

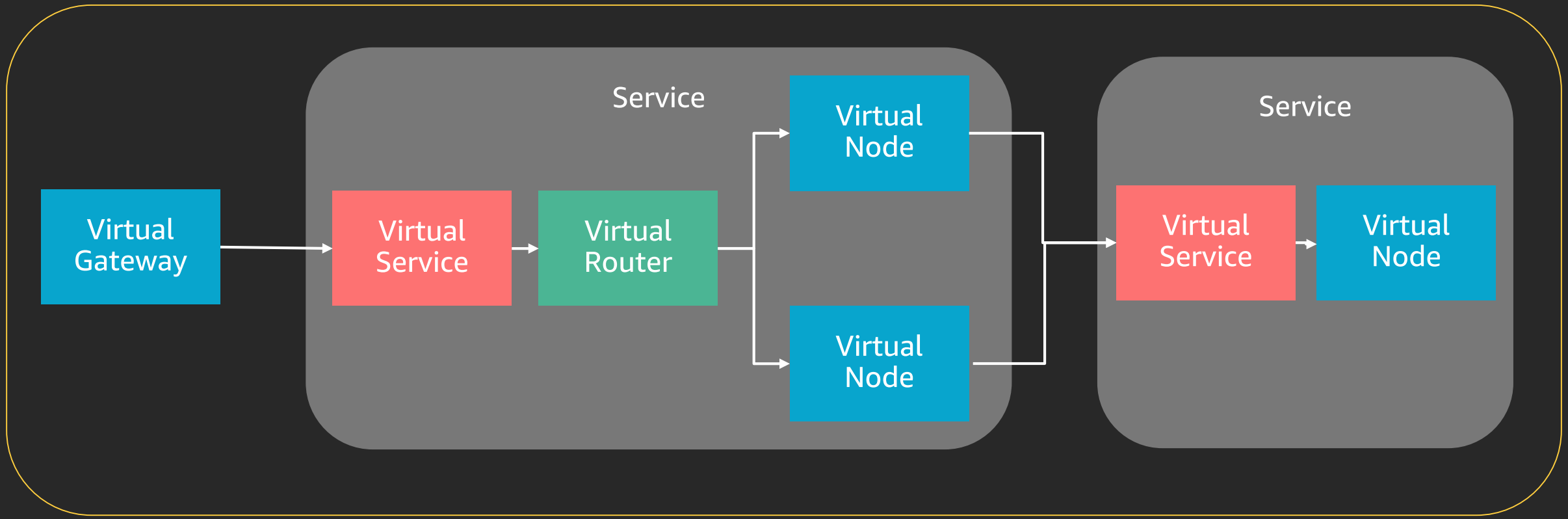
Logical name/
Service discovery

**Virtual Router**

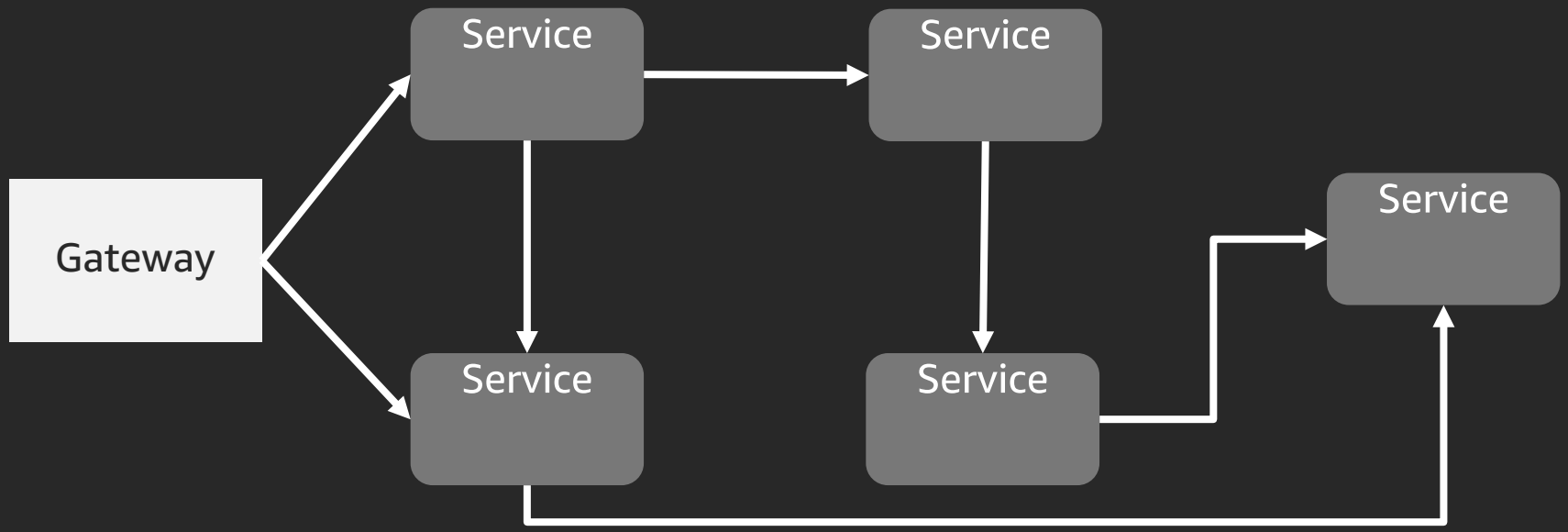
Routing match
conditions

**Virtual Node**

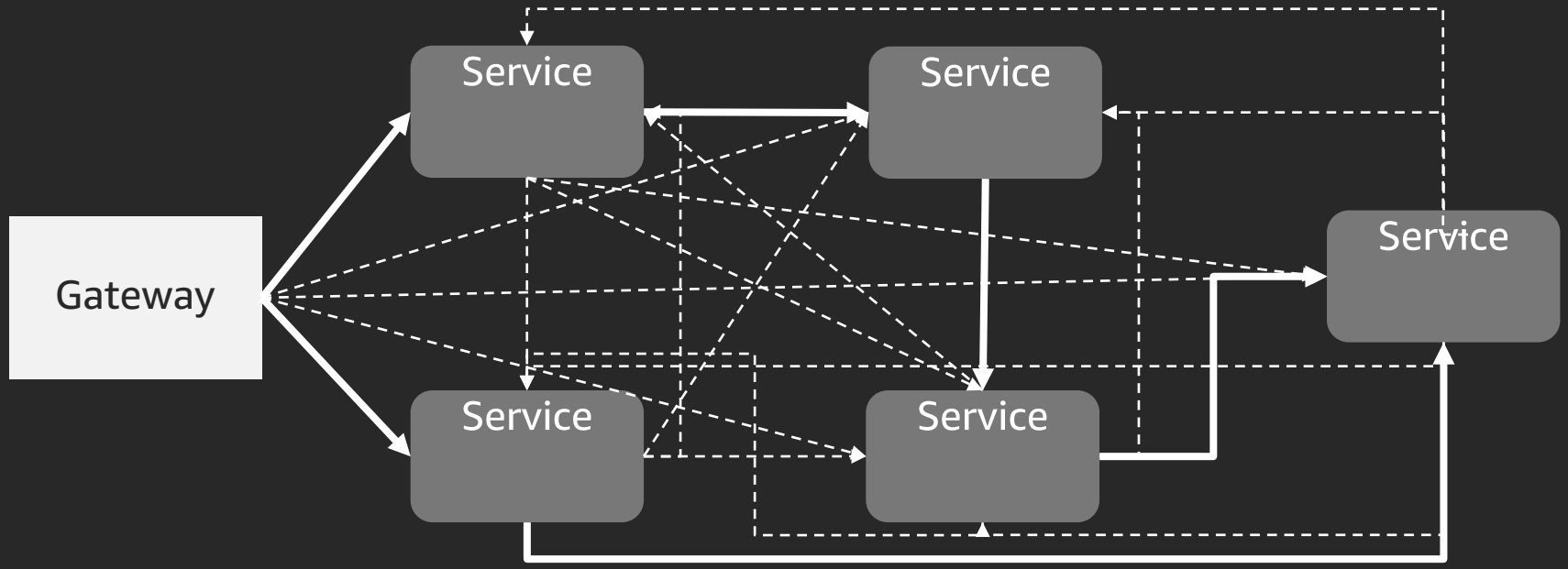
Group of service
endpoints



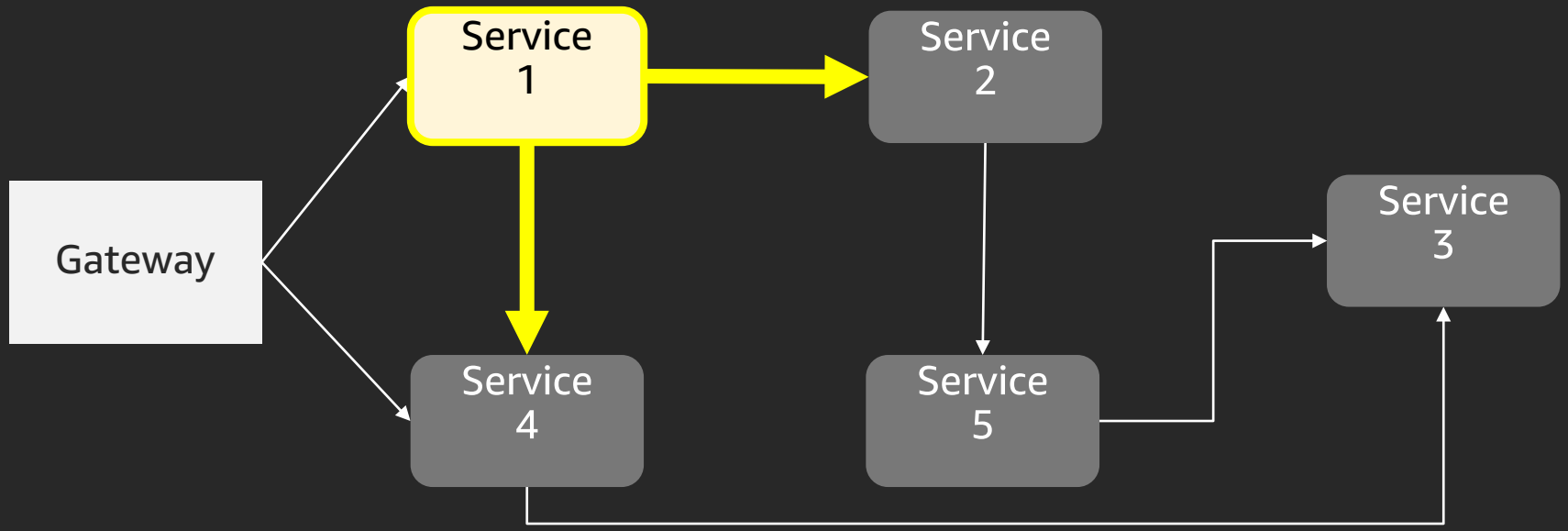
# App Mesh requires explicitly modeled dependencies



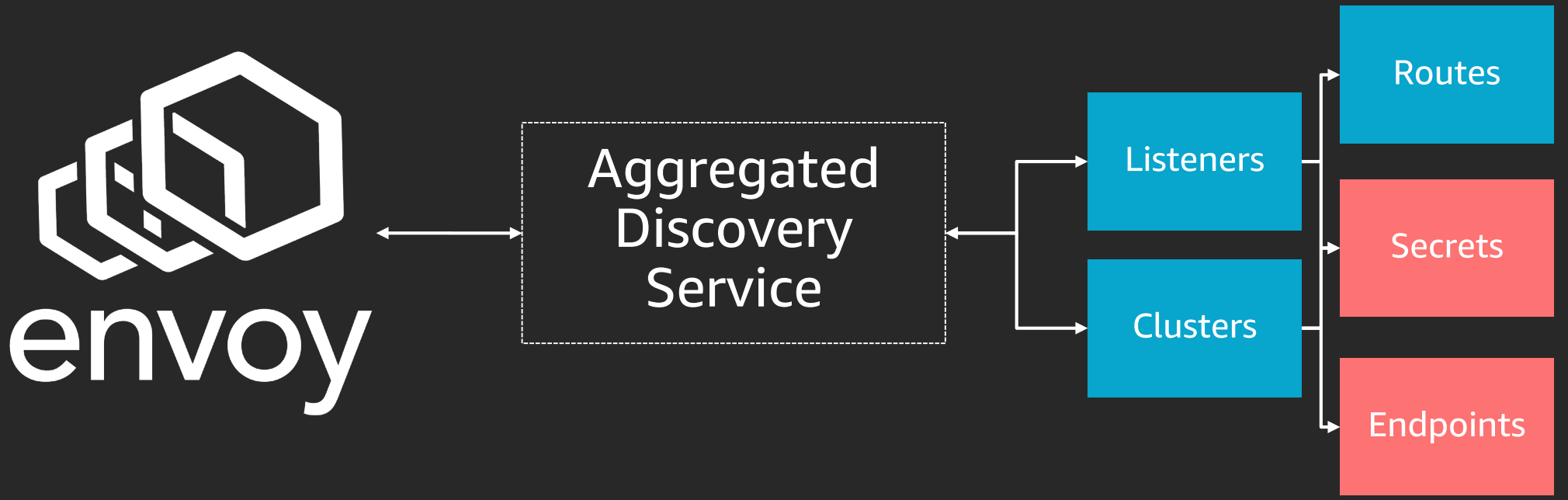
# Minimizes config size, mutation and blast radius



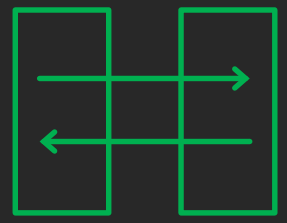
# Performs fragment parsing to customize config



# App Mesh implements Envoy xDS

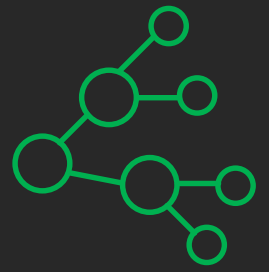


# App Mesh Features and Roadmap



Consistency across teams

Failure visibility and isolation



Fine-grained deployment controls

Authentication and Authorization controls

# Where we are and near term roadmap

**Observability**

Container Insights Mesh Dashboard

--

Access logs
Metrics

Tracing

AWS solutions and Partner integrations

**Traffic Management**

HTTP/2, gRPC

Ingress Gateway

--

Routing based on explicit dependency mapping

TCP, HTTP1.1

Path, Header based routing

Weighted Routing

Route Priorities

Retries

Health Check & Load Balancing at Envoy

**Security**

Encryption – ACM

Encryption – Bring your cert

mTLS  - Bring your cert

mTLS - ACM

Bring your AuthZ

# Where we are and near term roadmap

**Must-Haves**

Regions
CloudFormation
Scaling
PrivateLink
Backwards compatible API
Public Previews
New Envoy versions support

**Usability**

ECS Console
K8S Controllers
Helm Charts for K8S controllers
Flagger Integration
HIPAA Compliance

**Integrations**

Cross Account Support using AWS RAM
On-Prem / Hybrid with AWS Outposts
Metadata based routing with AWS Cloud Map
Cross Cluster Support using AWS Cloud Map

# Community Participation

Upstream Envoy support – SigV4 AuthN for Envoys

Envoy configuration sequencing changes

ACM integration with SPIFFE/SPIRE

X-Ray integration

Open Souring App Mesh control plane – Envoy Management Service

# Edmunds: Improving modern software development and microservices with App Mesh


Edmunds is a leading car information and shopping platform, helping millions of visitors each month find their perfect car.

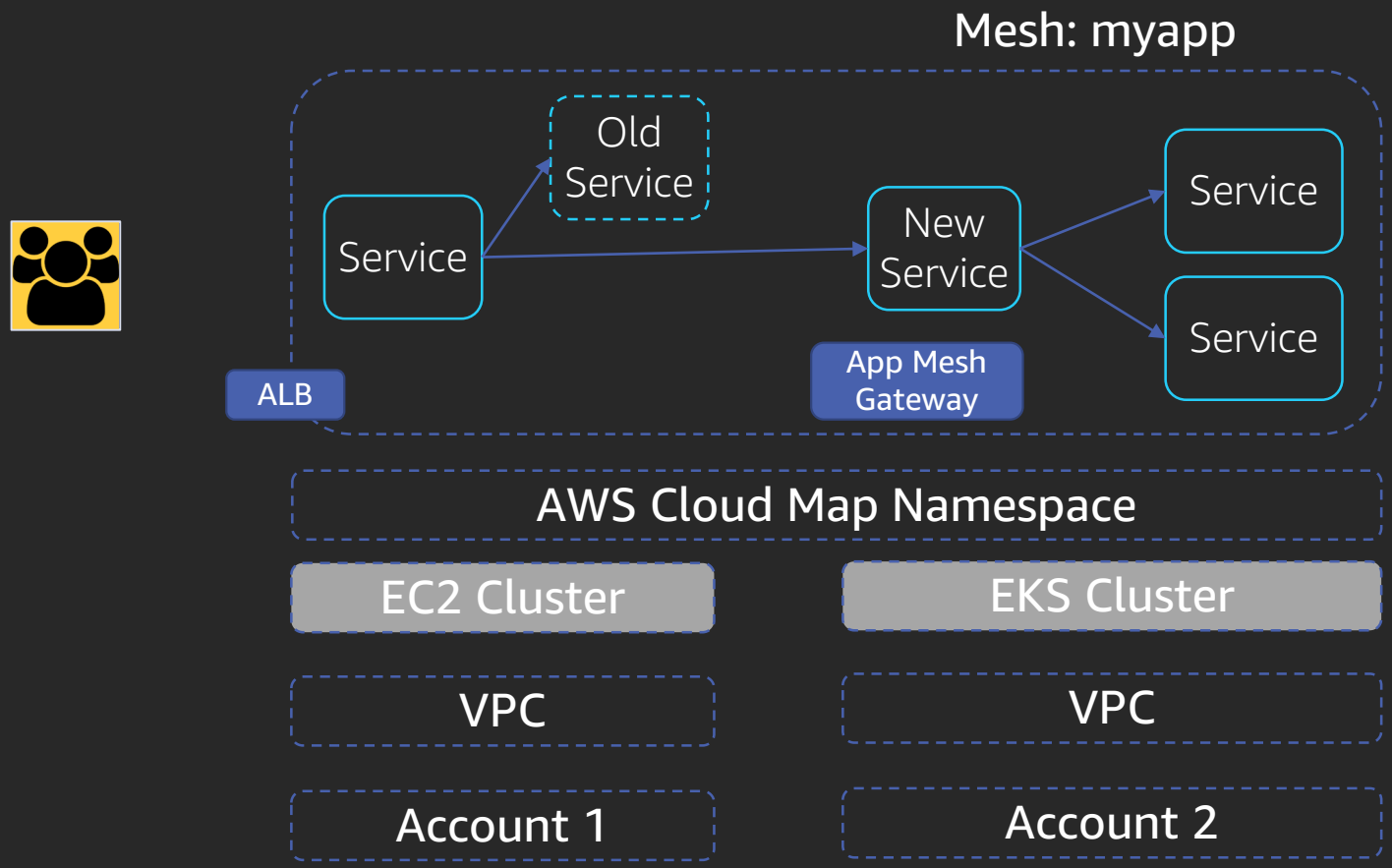
Challenge: There are a number of challenges with Microservices, including control over service-to-service communication, visibility into service-to-service communication and ensure trust by automating security and compliance on small Dev-Ops teams.

Solution: *"App Mesh provide us a consistent communications management, complete visibility, failure isolation, protection, and fine-grained deployment controls." - Nitin Mahajan, Executive Director, service engineering, Edmunds*

https://youtu.be/1UDRGlmbiZA

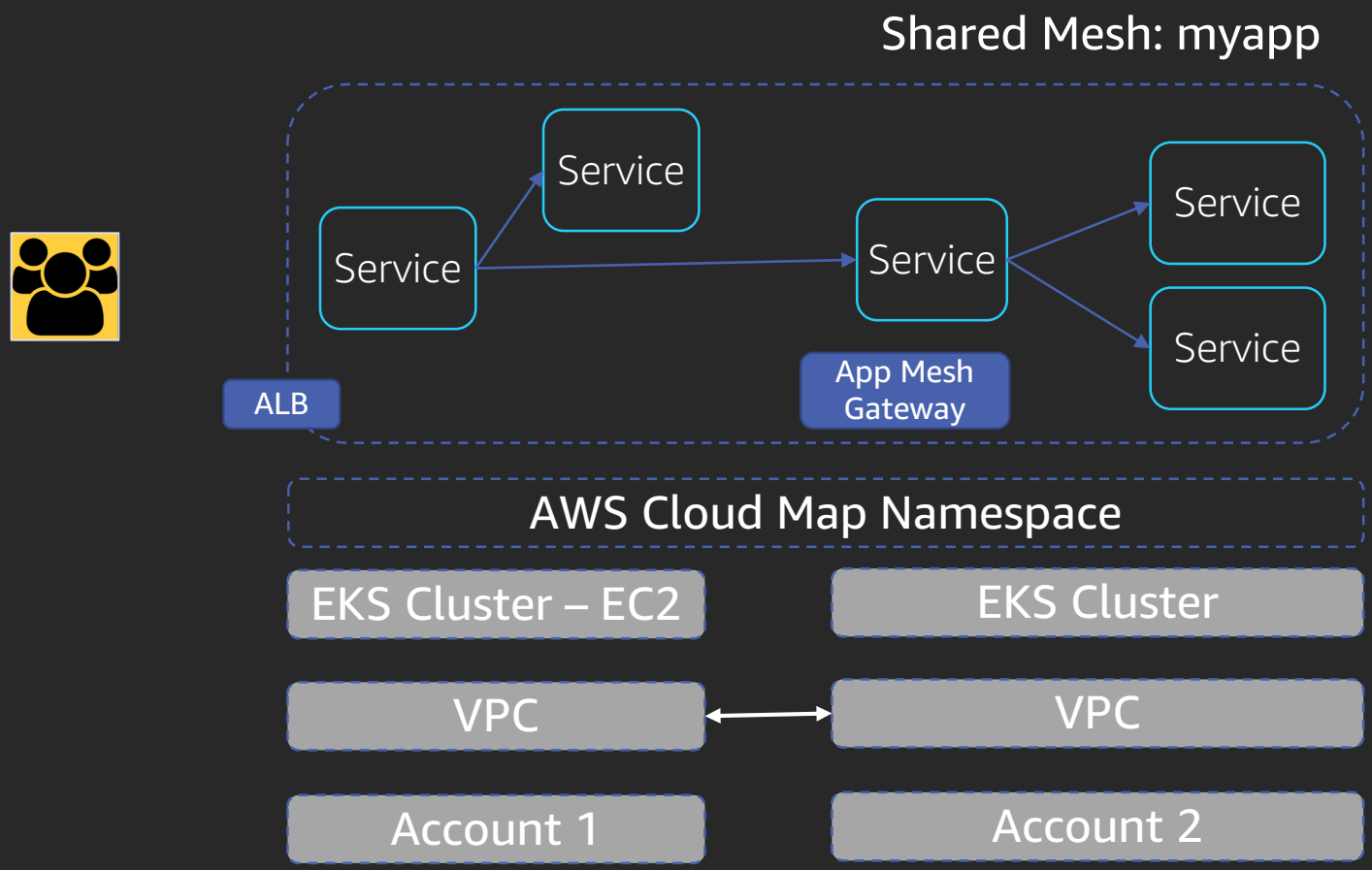
# Use Cases

## 1. Migrate from EC2 or self-managed Kubernetes to EKS



# Use Cases

## 2. Span a mesh across compute mode, accounts, clusters and VPCs

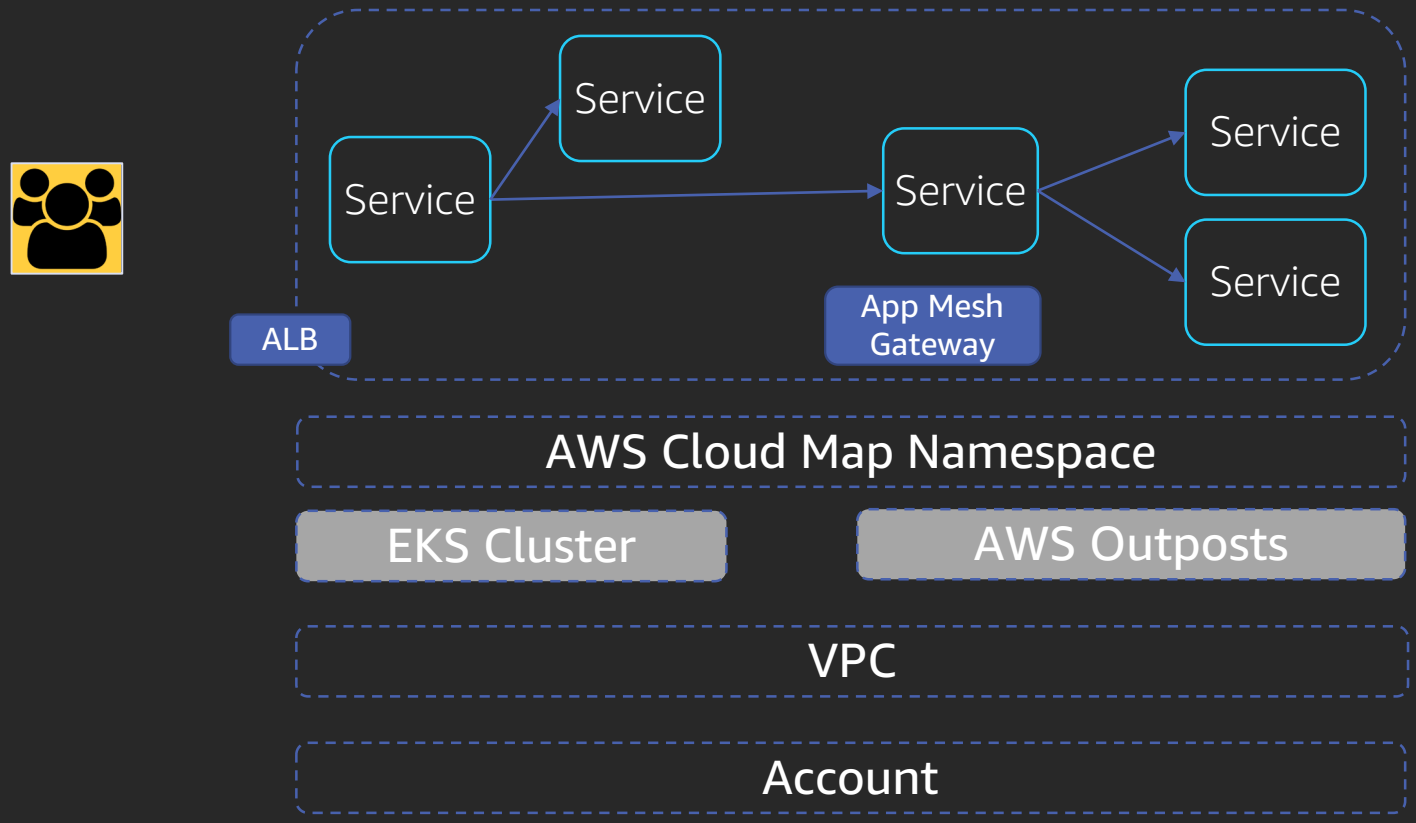



Assumes flat network – one of shared VPC via RAM, VPC Peering, PrivateLink, Transit GW

Use AWS Resource Access Manager to share the mesh resource

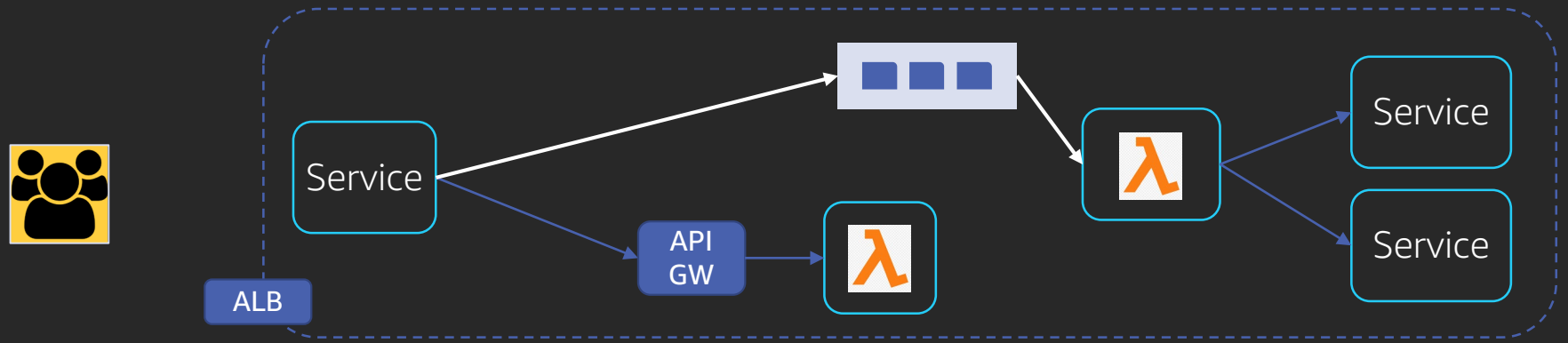
# Use Cases

## 3. Span a mesh across hybrid deployments



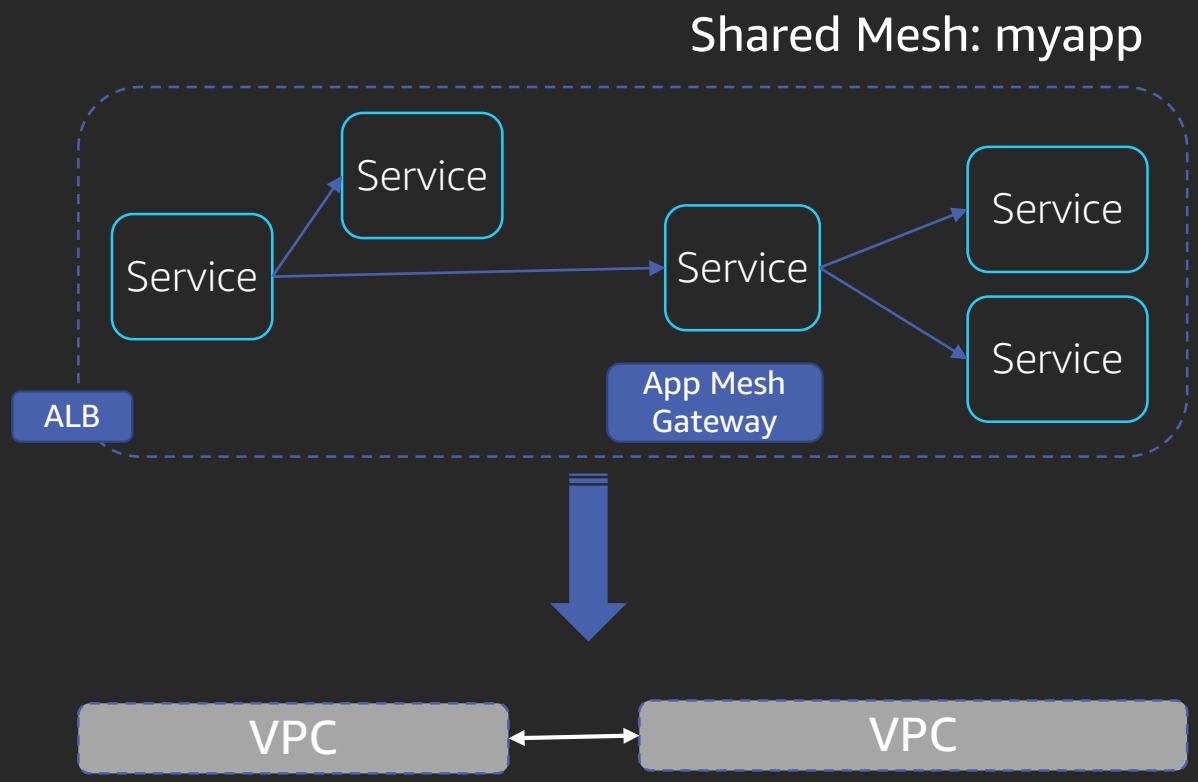
# Future Directions- Brainstorming (not on roadmap)

## 1. Lambda Support



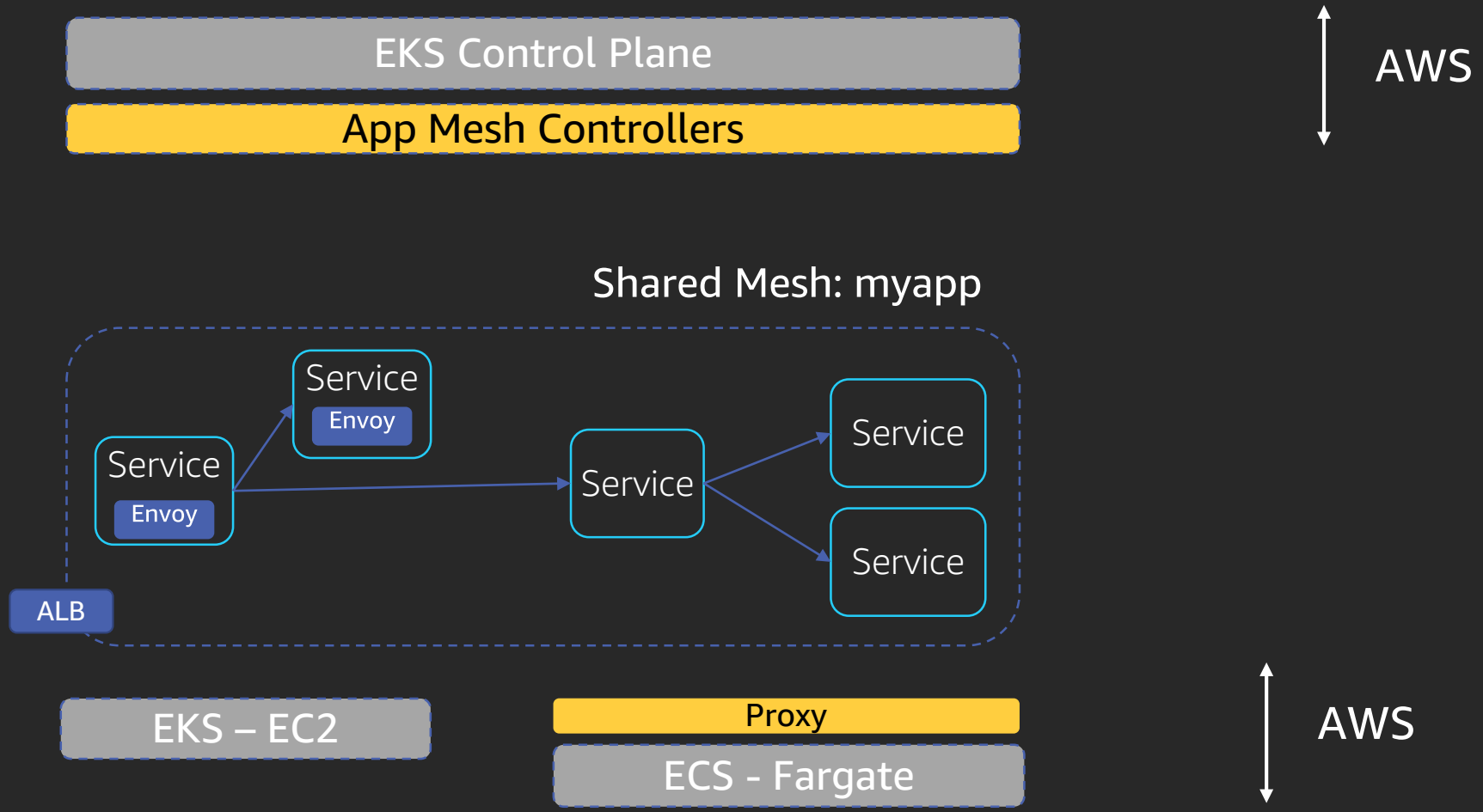
# Future Directions- Brainstorming (not on roadmap)

## 2. Connection management



# Future Directions- Brainstorming (not on roadmap)

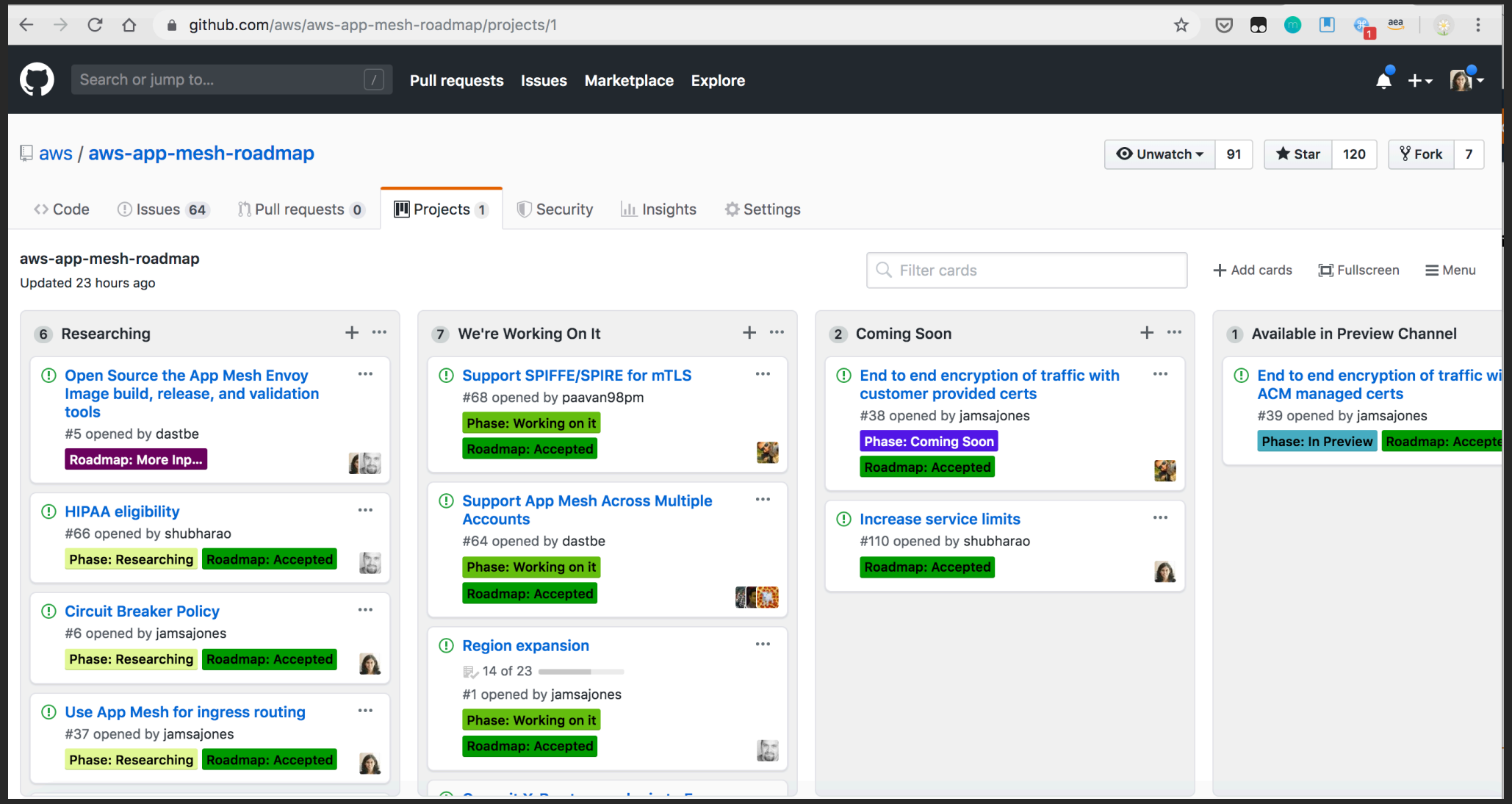
## 3. More Managed (Think Fargate)



# Give us your input

**surveymonkey.com/r/appmesh**

# Roadmap: https://github.com/aws/aws-app-mesh-roadmap



# Engage with us!

- Take the App Mesh Workshop
- Meet our team at the booth
- Check example apps and config on Github: https://github.com/aws/aws-app-mesh-examples/tree/master/walkthroughs
- Read our latest post on Containers blog: https://aws.amazon.com/blogs/containers/cross-amazon-eks-cluster-app-mesh-using-aws-cloud-map/
- Tweet with #appmesh

# Engage with us


Meet us at our booth


aws-app-mesh-roadmap

#appmesh or _shubha

# Vision

“Our goal is that if you are running applications on AWS, you should not have to worry about managing networking infrastructure. It should be handled by our application-aware network … App Mesh, as it exists today, is the first step in this journey.”

~Werner Vogels

https://www.allthingsdistributed.com/2019/03/redefining-application-communications-with-aws-app-mesh.html

# Give us your input: surveymonkey.com/r/appmesh