Istio, Linkerd und Co. im Vergleich

Welches Service Mesh passt zu mir?

INNOQ

Jörg Müller
@joergm

Hanna Prinz
@HannaPrinz

- Software Development
- DevOps, Kubernetes, Service Mesh

Hanna Prinz
Consultant
innoQ Deutschland GmbH

hanna.prinz@innoq.com
@HannaPrinz

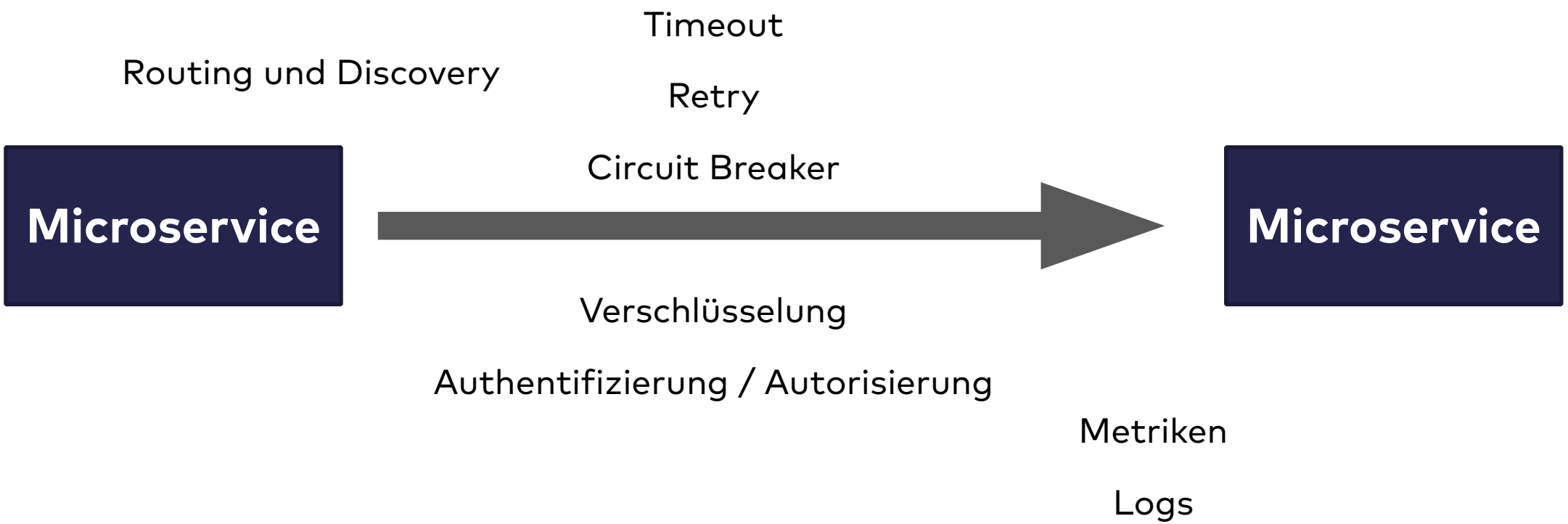
- architecture, development, DevOps
- focus on platform & infrastructure

Jörg Müller
Principal Consultant
innoQ Deutschland GmbH

joerg.mueller@innoq.com
@joergm

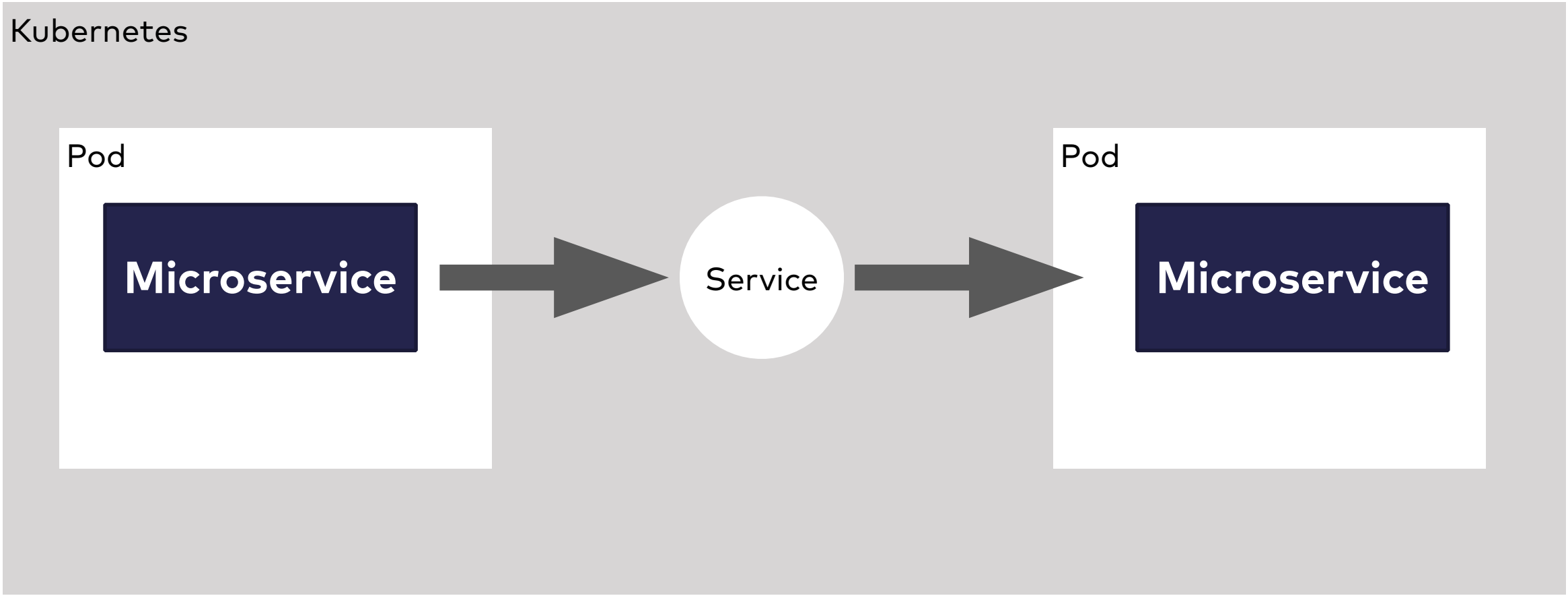
Was ist ein Service Mesh?

Microservices sind verteilte Systeme

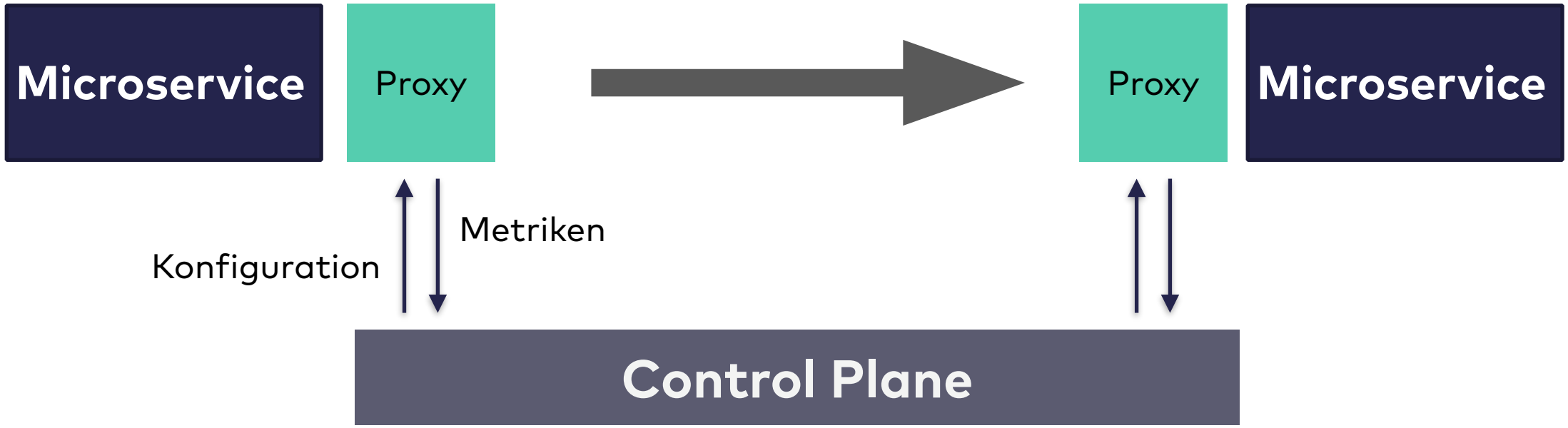


Libraries können helfen

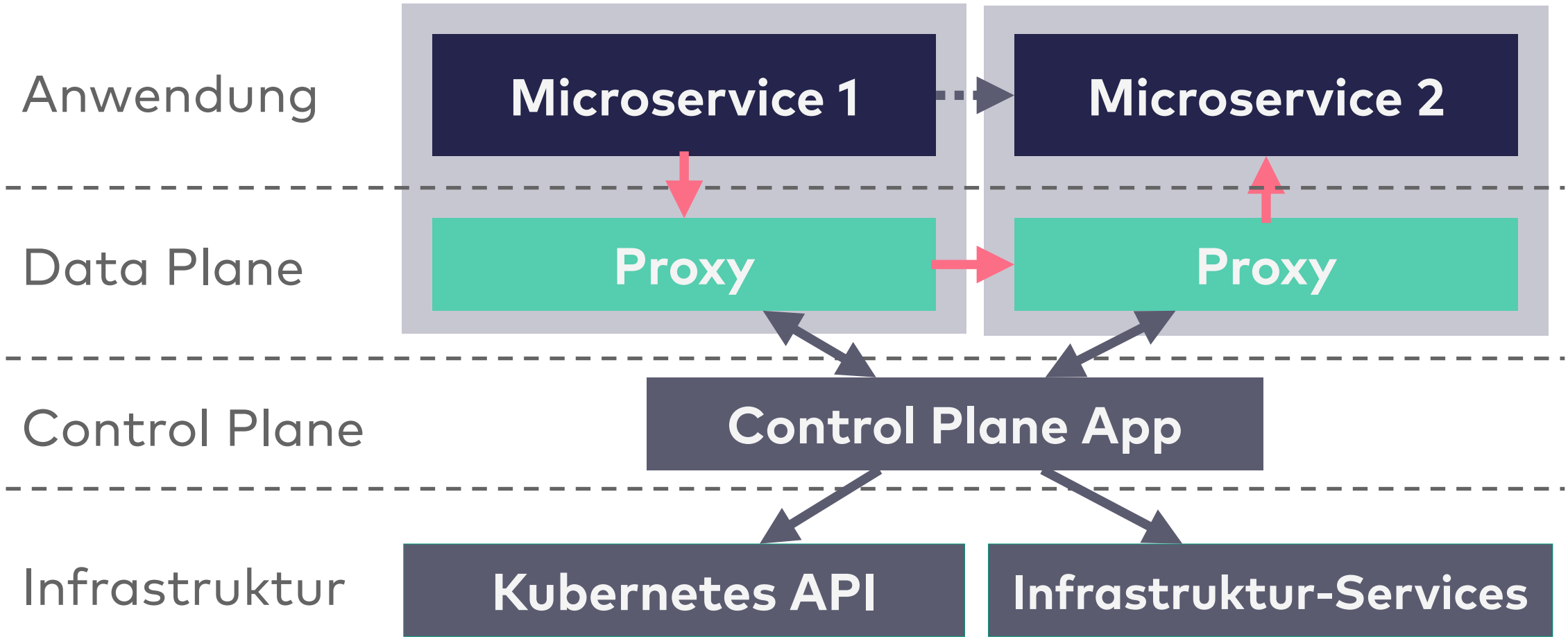
Kubernetes hilft auch



Service Mesh Ansatz



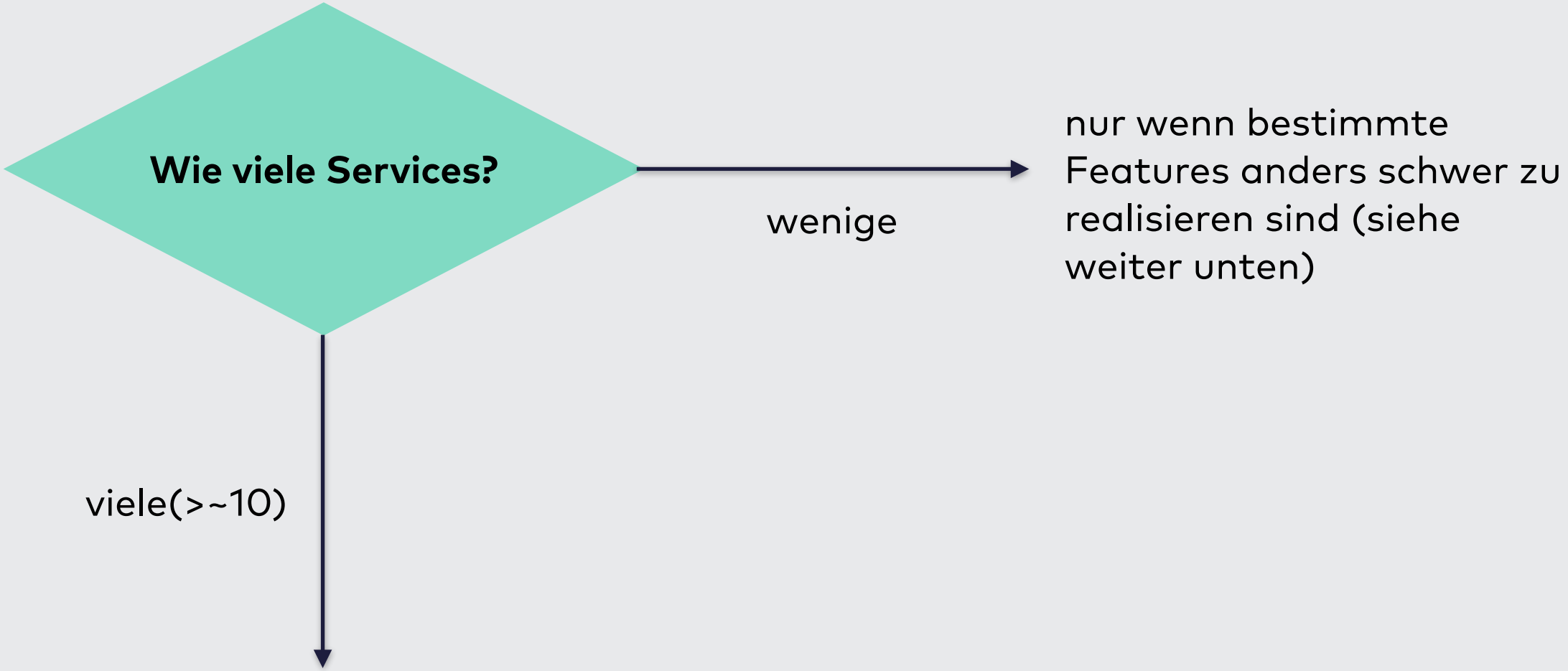
Service Mesh Architektur

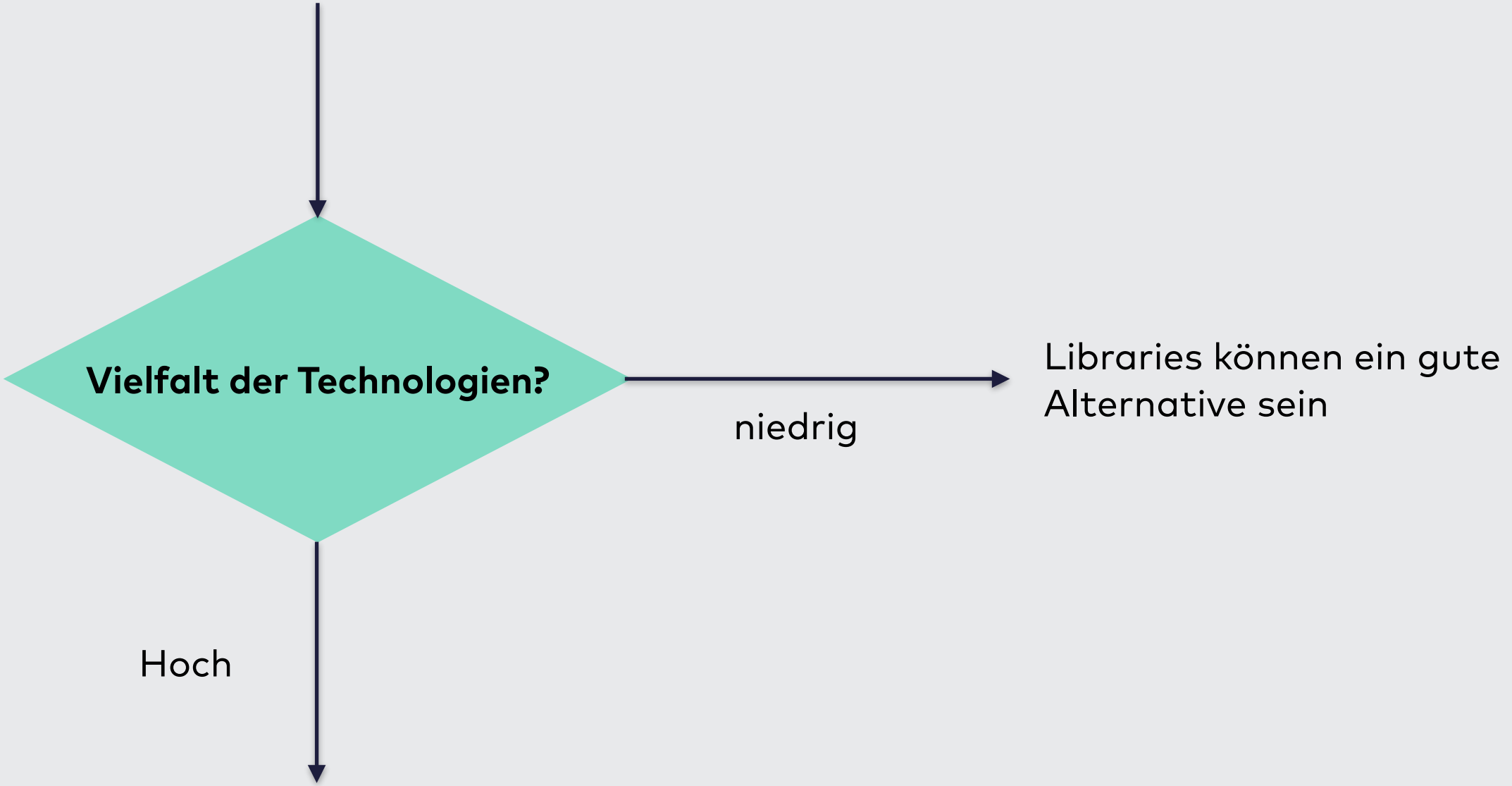


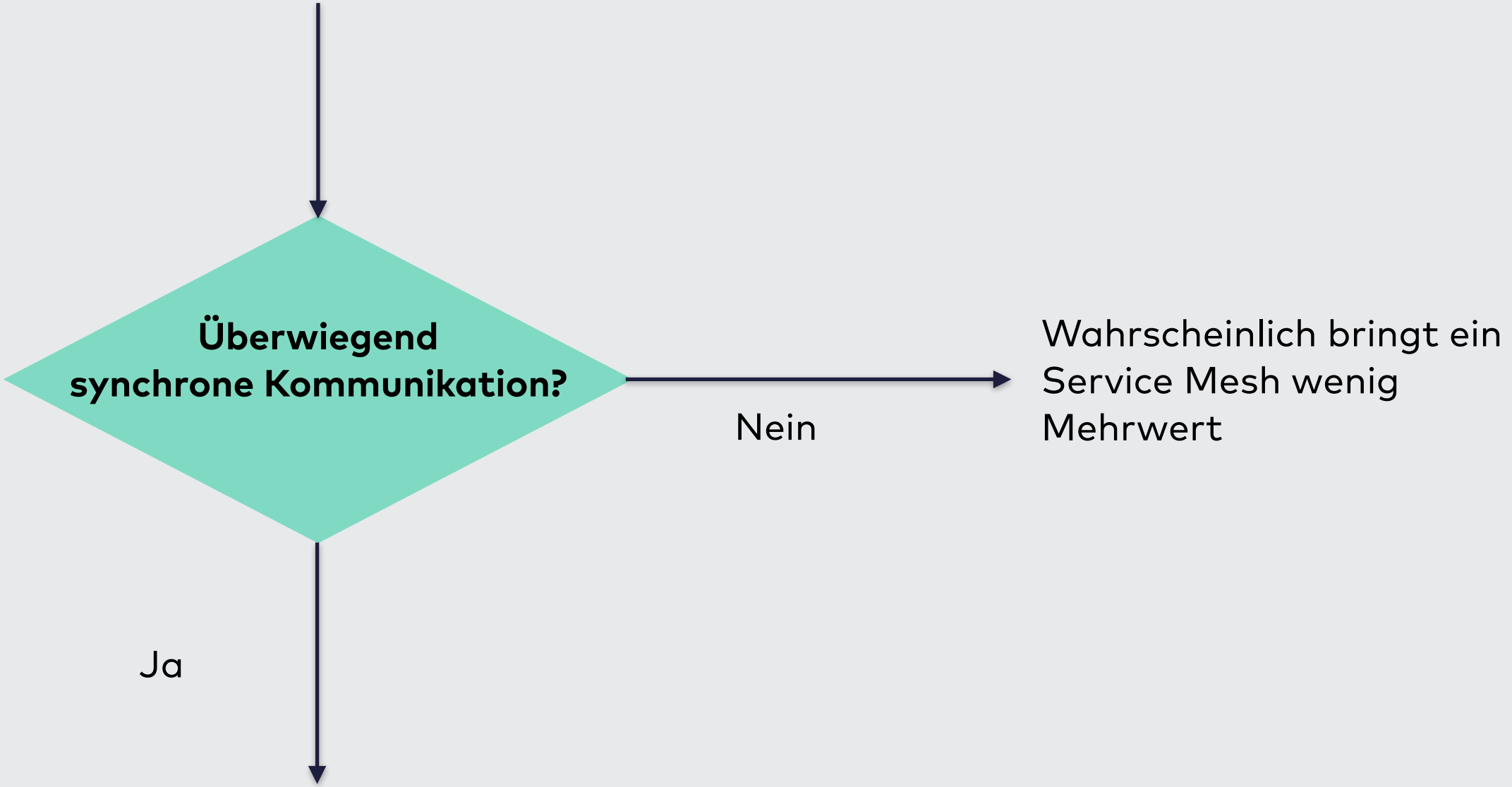
Ist ein Service Mesh überhaupt sinnvoll?

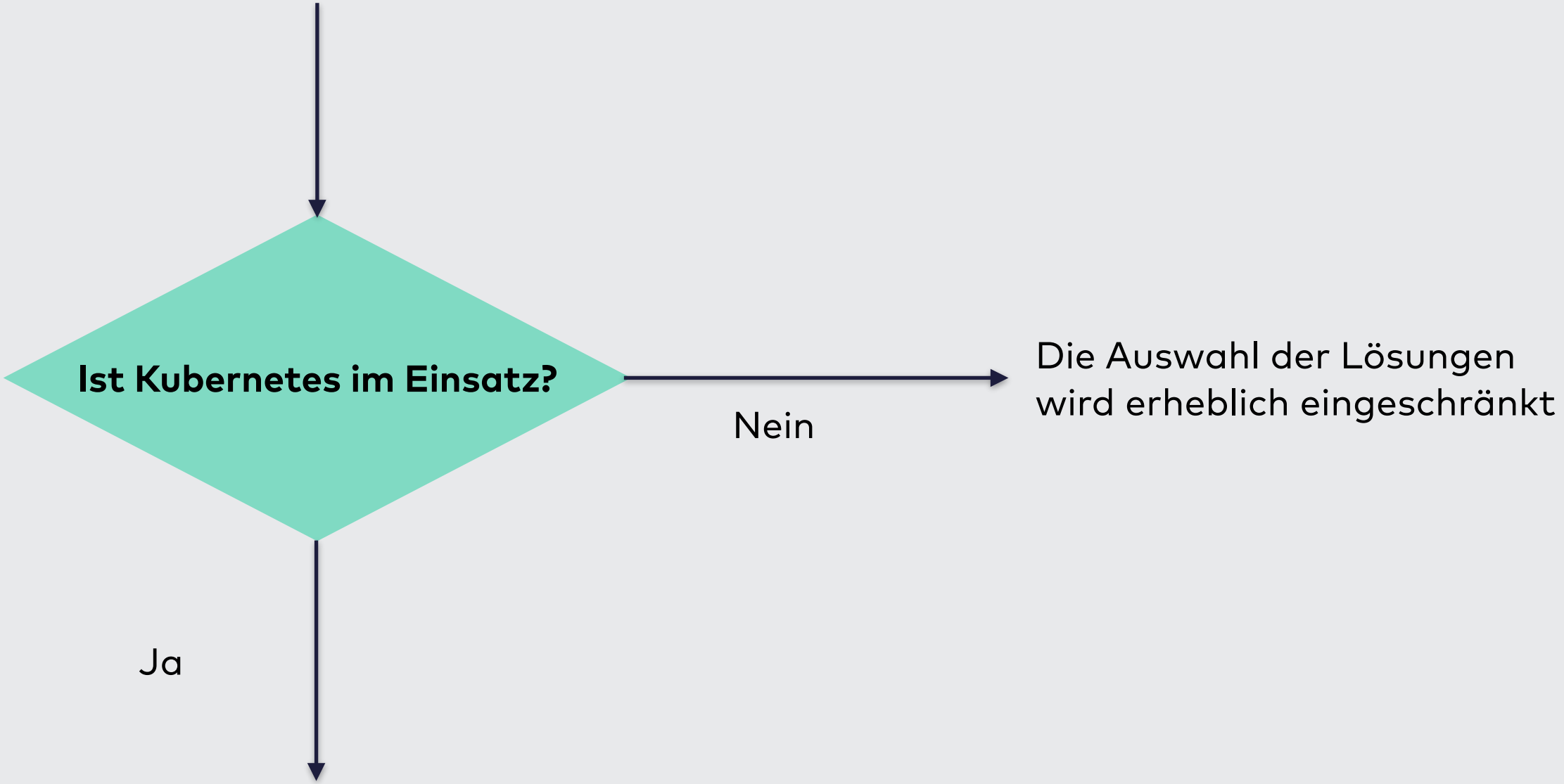
Oft lautet die Antwort:

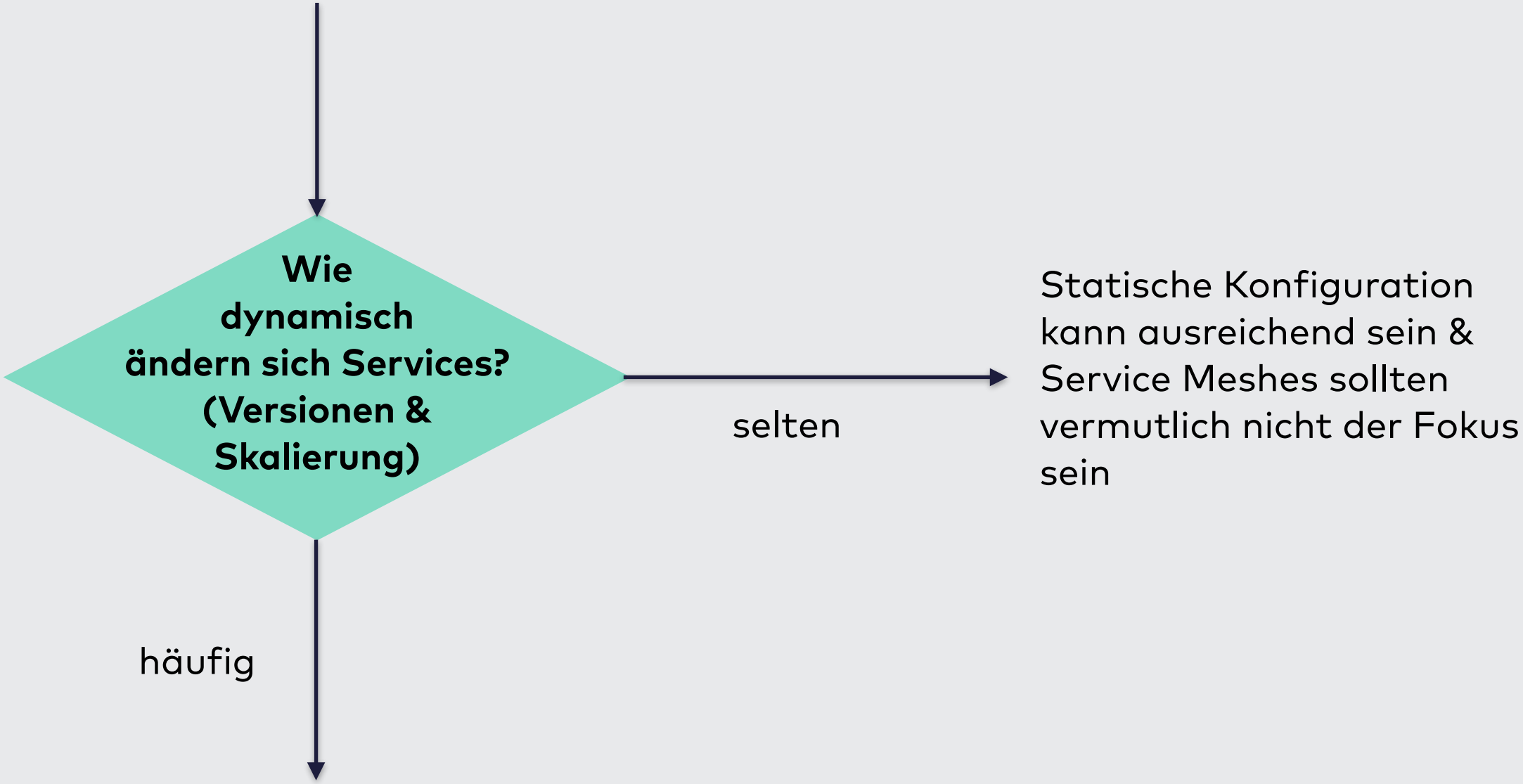
Nein

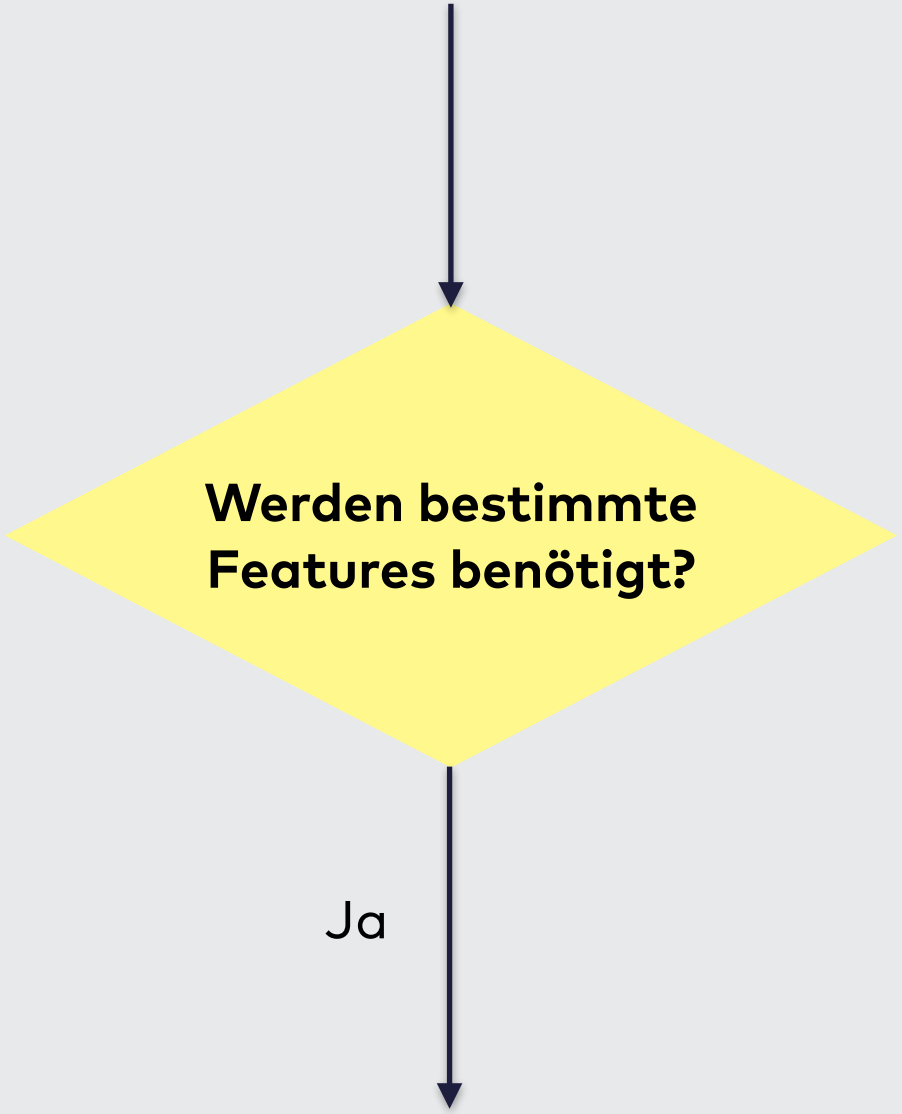

Wie viele Services?
wenige
nur wenn bestimmte Features anders schwer zu realisieren sind (siehe weiter unten)
viele(>~10)


Vielfalt der Technologien?
niedrig
Libraries können ein gute Alternative sein
Hoch


Überwiegend synchrone Kommunikation?
Nein
Wahrscheinlich bringt ein Service Mesh wenig Mehrwert
Ja


Ist Kubernetes im Einsatz?
Nein
Die Auswahl der Lösungen wird erheblich eingeschränkt
Ja


Wie dynamisch ändern sich Services? (Versionen & Skalierung)
selten
Statische Konfiguration kann ausreichend sein & Service Meshes sollten vermutlich nicht der Fokus sein
häufig

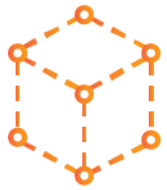


- mTLS
- Tracing
- Routing
- spezielle Rollouts

Aktuelle Implementierungen

Service Mesh Implementierungen

LINKERD



AWS App Mesh

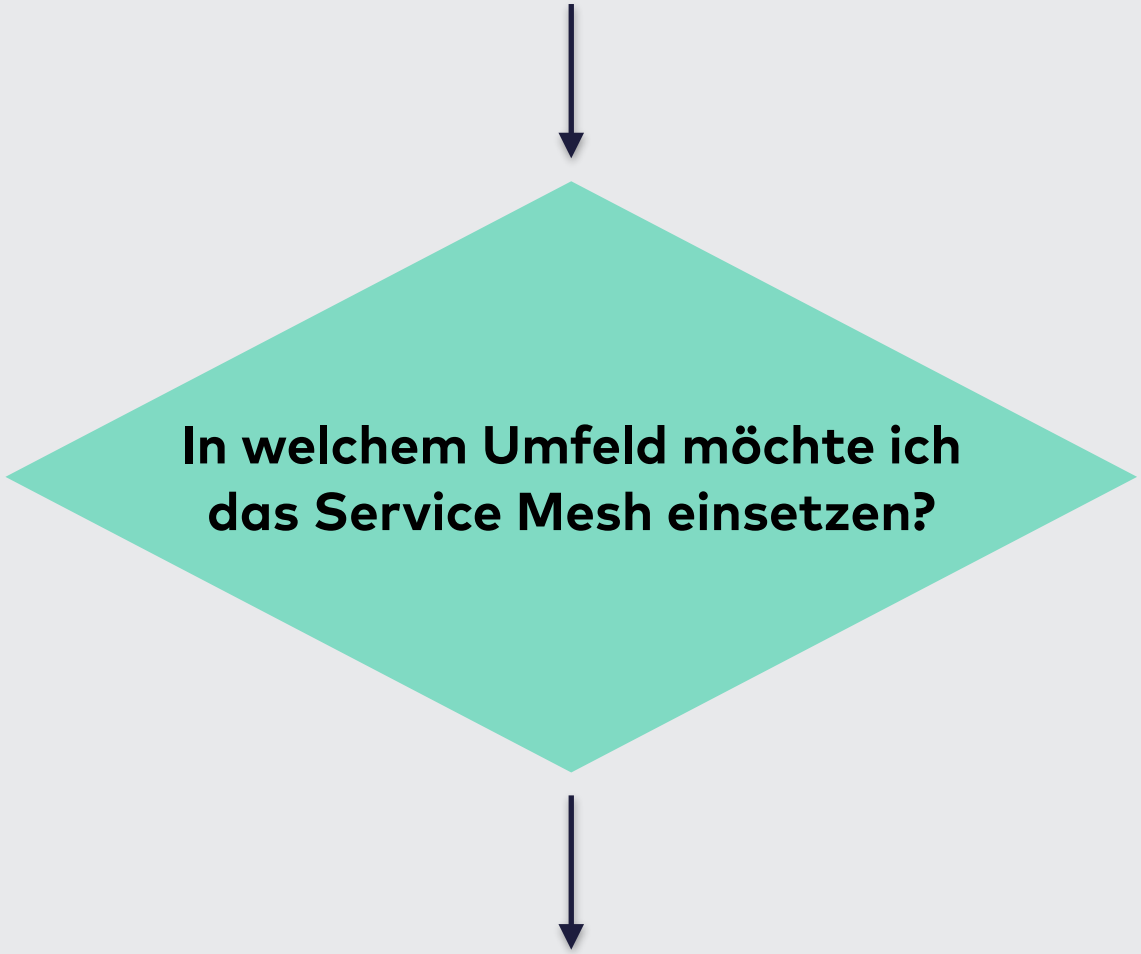
Istio

Open Service Mesh

servicemesh.es

	Istio	Linkerd 2	AWS App Mesh	Consul Connect	Maesh	Kuma	Open Service Mesh (OSM)
Current version	1.7	2.8		1.8	1.3	0.7	0.3
License	Apache License 2.0	Apache License 2.0	Closed Source	Mozilla License	Apache License 2.0	Apache License 2.0	MIT License
Developed by	Google, IBM, Lyft	Buoyant	AWS	HashiCorp	Containous	Kong	Microsoft
Service Proxy	Envoy	linkerd-proxy	Envoy	defaults to Envoy, exchangeable	Traefik	Envoy	Envoy
Ingress Controller	Envoy / Own Concept	any		Envoy and Ambassador in Kubernetes	any	any	Nginx, Azure Application Gateway Ingress Controller
Governance	see Istio Community and Open Usage Commons	see Linkerd Governance and CNCF Charter	AWS	see Contributing to Consul	see Contributing notice	see Contributing notice, Governance, and CNCF Charter	see Microsoft OpenSource
	Istio Tasks	Linkerd Tasks	AWS App	HashiCorp Learn	Maesh	Install Kuma on	Install OSM

Eine Implementierung wählen


In welchem Umfeld möchte ich
das Service Mesh einsetzen?

Fragen zum Umfeld

- Welche Infrastruktur habe ich im Einsatz?
- Gibt es präferierte Cloudanbieter?
- Welches Wissen ist vorhanden?
- Wie flexibel wollen wir sein?

Kubernetes

nur auf Kubernetes

Istio

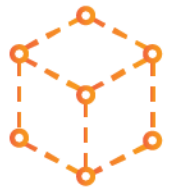
LINKERD

træfik mesh

Open Service Mesh

auch ohne Kubernetes

HashiCorp Consul



AWS App Mesh

Kuma

Cloud Anbieter

- Der Einfluss ist gering, so lange Kubernetes zum Einsatz kommt
- Viele AWS Services (insbesondere Fargate/ECS) können ein Indikator für AWS App Mesh sein
- Google Cloud hat sehr guten Istio Support
- Microsoft Azure wird sich vermutlich in Richtung OSM bewegen

Unabhängigkeit durch SMI?

- "A standard interface for service meshes on Kubernetes"
- Features:
  - Traffic Access Control
  - Traffic Metrics
  - Traffic Specs
  - Traffic Split
- Definiert CRDs in Kubernetes, auf die die Implementierungen zugreifen

Service Mesh Interface Support

voll

Istio

Open Service Mesh

teilweise

LINKERD

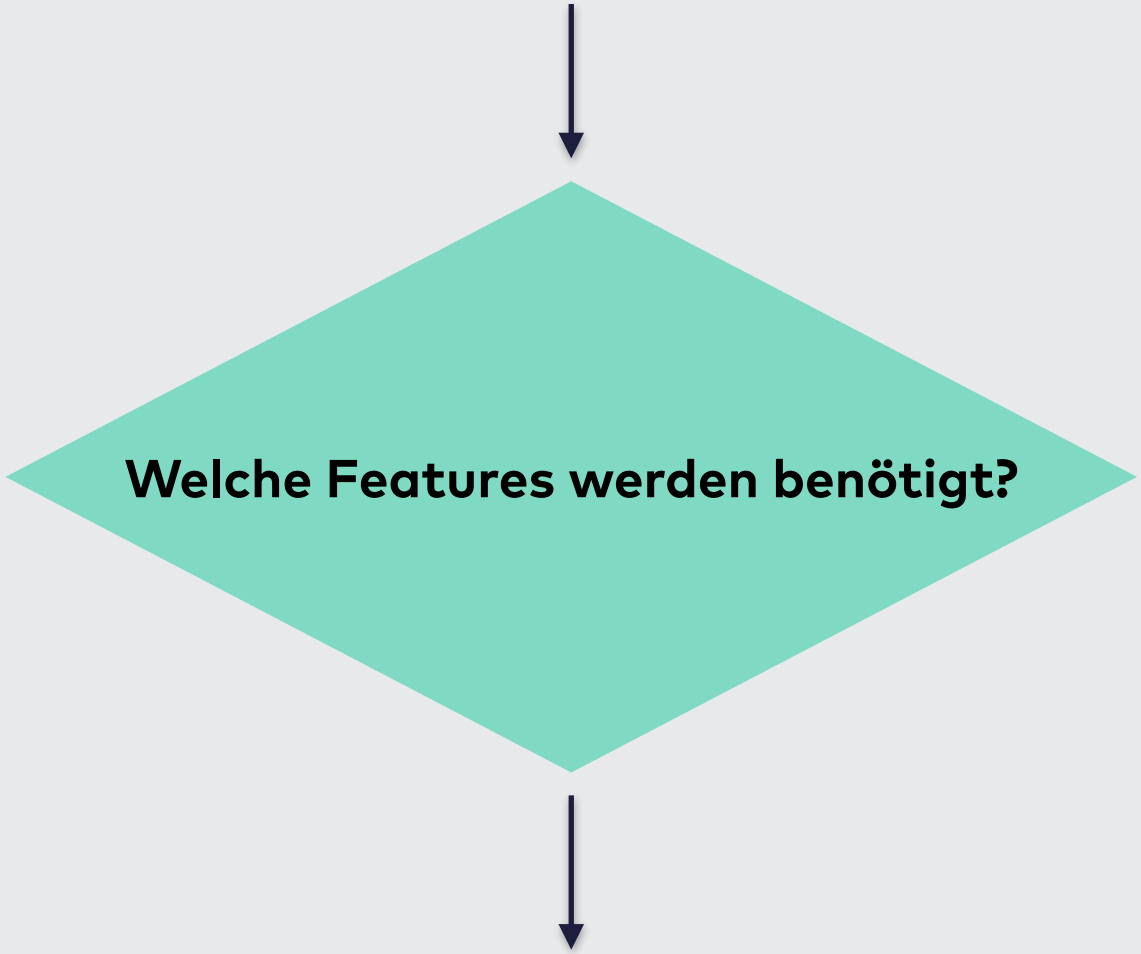
traefik mesh

HashiCorp Consul

ohne

AWS App Mesh

Kuma


Welche Features werden benötigt?

Fragen zu Features

- Welche Probleme gibt es aktuell im Projekt?
- Gibt es Must-Haves/Nice-to-Haves?
- Welches Level an Konfigurierbarkeit ist nötig?
- Welcher Aufwand soll betrieben werden?

Unterschiede zwischen den Meshes

Routing

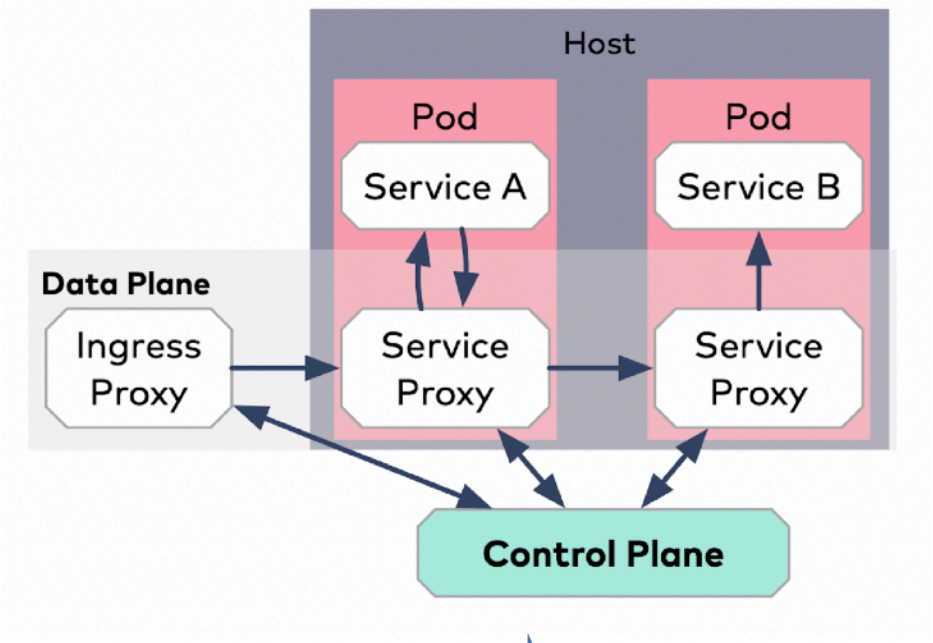
Resilience

Security

Observability

Proxy & Ingress

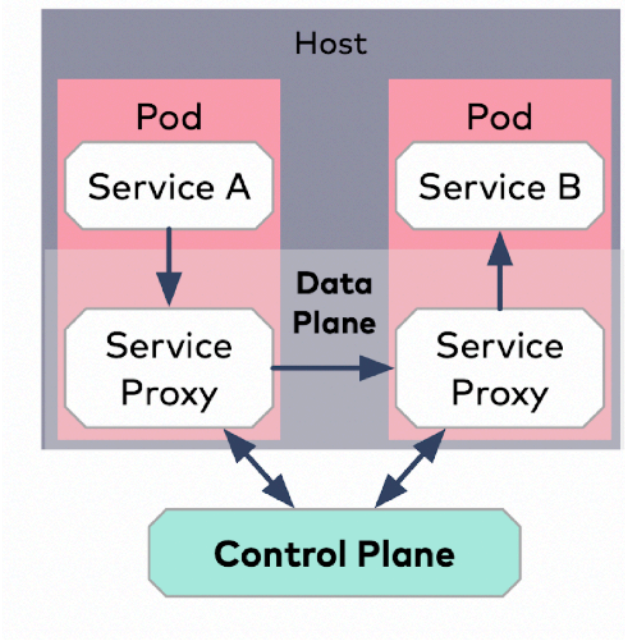
Proxy pro Service & als Ingress Gateway



Istio

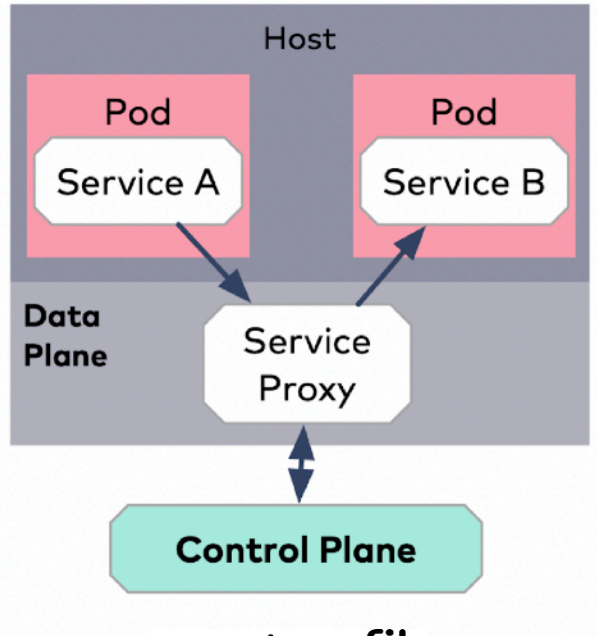
Consul

Proxy pro Service



Andere Service Meshes

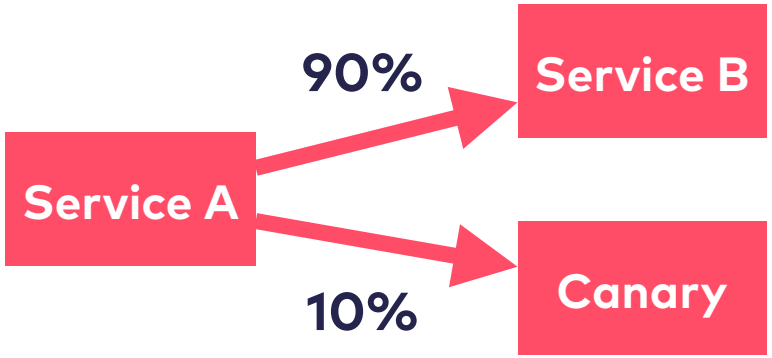
Proxy pro Node



traefik mesh

Canary Releasing & A/B Testing

Rein prozentual

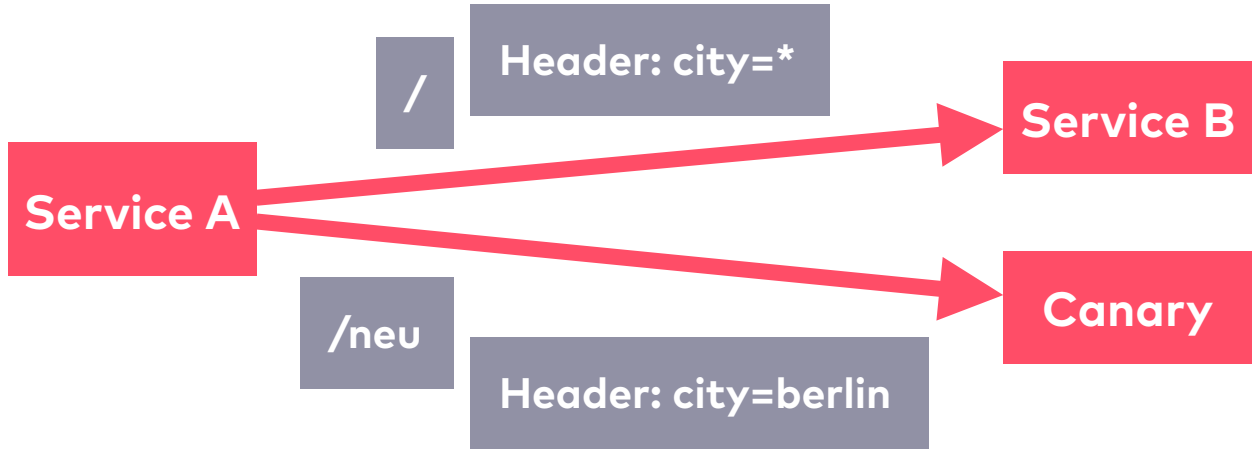


Kuma

LINKERD

træfik mesh

+ Header- oder Pfad-basiert



Open Service Mesh

Istio

AWS App Mesh

HashiCorp Consul

Unterschiede zwischen den Meshes

Routing

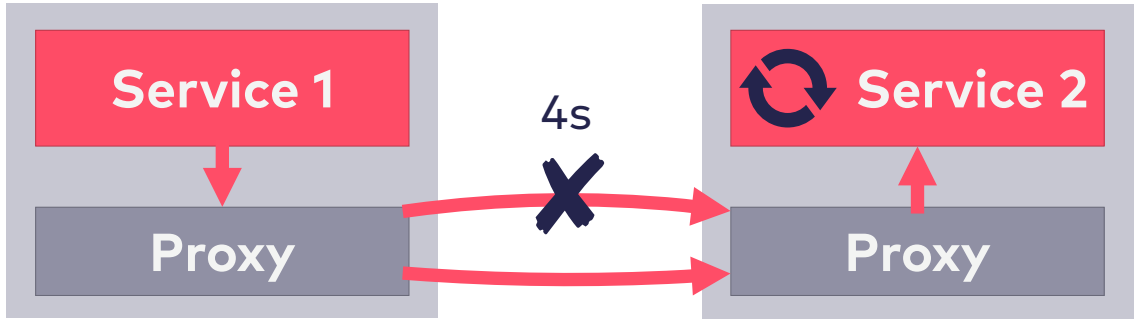
Resilience

Security

Observability

Resilience Unterschiede

Timeout
Retry
Circuit Breaking

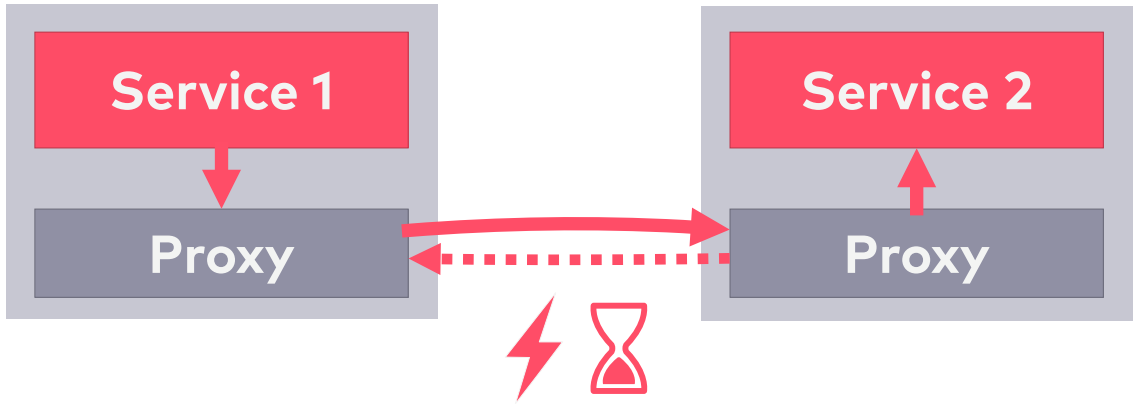


- Nicht von allen Service Meshes unterstützt
- Retry-Config gilt teilweise pro Service

  → keine Ausnahme von nicht-idempotenten Endpunkten wie HTTP POST!

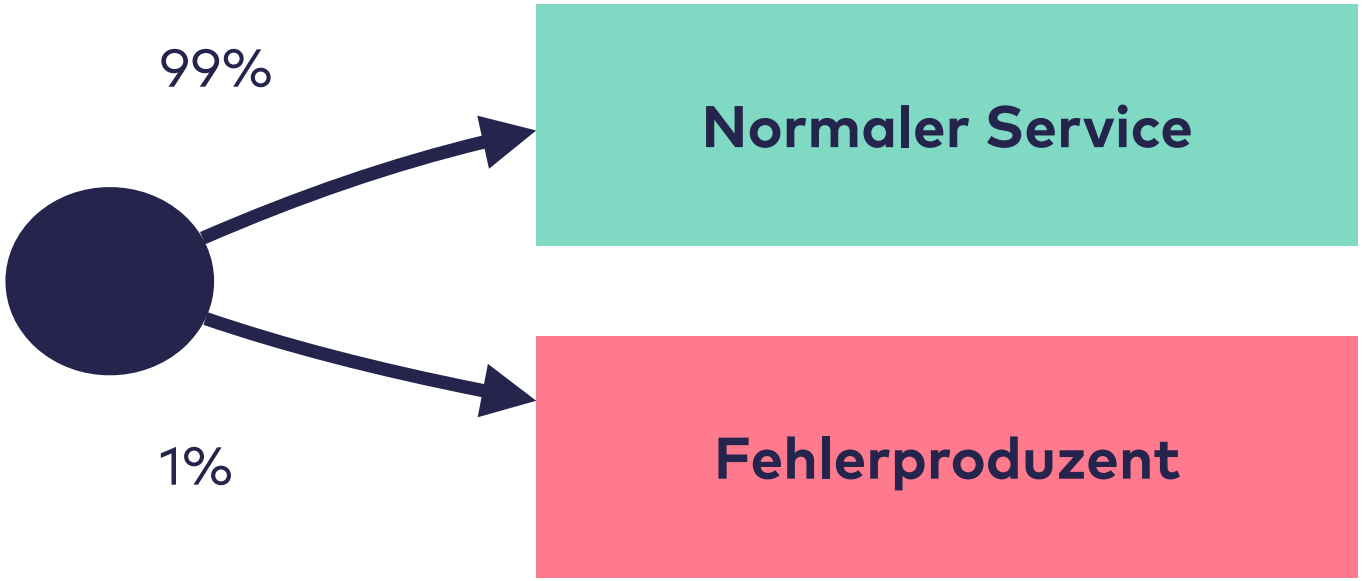
Chaos Engineering

Fault Injection
Delay Injection



- In Istio, Kuma und teilw. Linkerd unterstützt
- Umsetzung teilweise über zusätzliches Deployment + SMI Traffic Split

Fault Injection durch Traffic Split



→ Z.B. bei Linkerd 2

→ Benötigt zusätzliches Deployment, das Fehler produziert

Unterschiede zwischen den Meshes

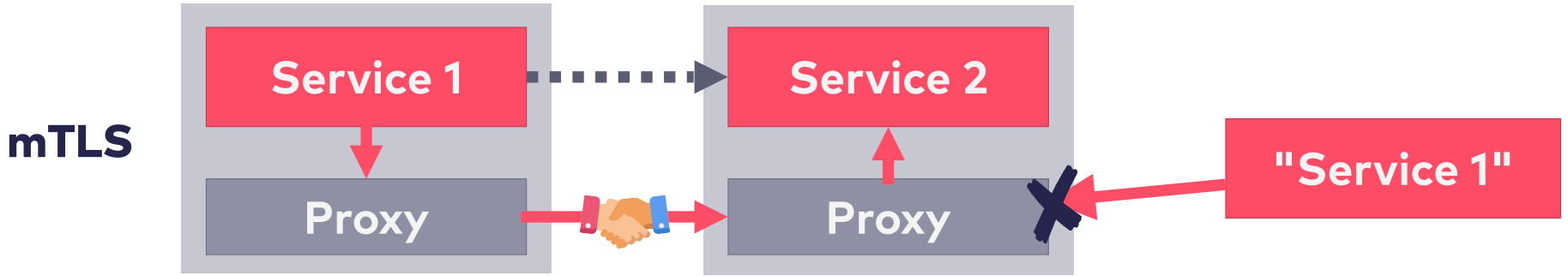
Routing

Resilience

Security

Observability

Security Unterschiede



- Alle Meshes bis auf Traefik Mesh unterstützen mTLS
- Unterschiede gibt es
  - bei mTLS für TCP-Verbindungen
  - beim Erzwingen von mTLS (TLS Enforcement)

Unterschiede zwischen den Meshes

Routing

Resilience

Security

Observability

Observability Unterschiede

- Qualität des Dashboards

LINKERD

Namespace > default

CLUSTER

- Namespaces
- Control Plane

DEFAULT

WORKLOADS

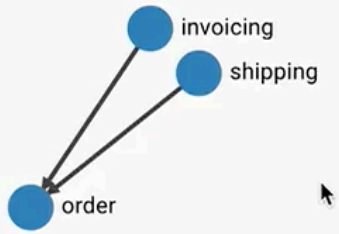
- Cron Jobs
- Daemon Sets
- Deployments
- Jobs
- Pods
- Replica Sets
- Replication Controllers
- Stateful Sets

CONFIGURATION

- Traffic Splits

TOOLS

- Tap
- Top

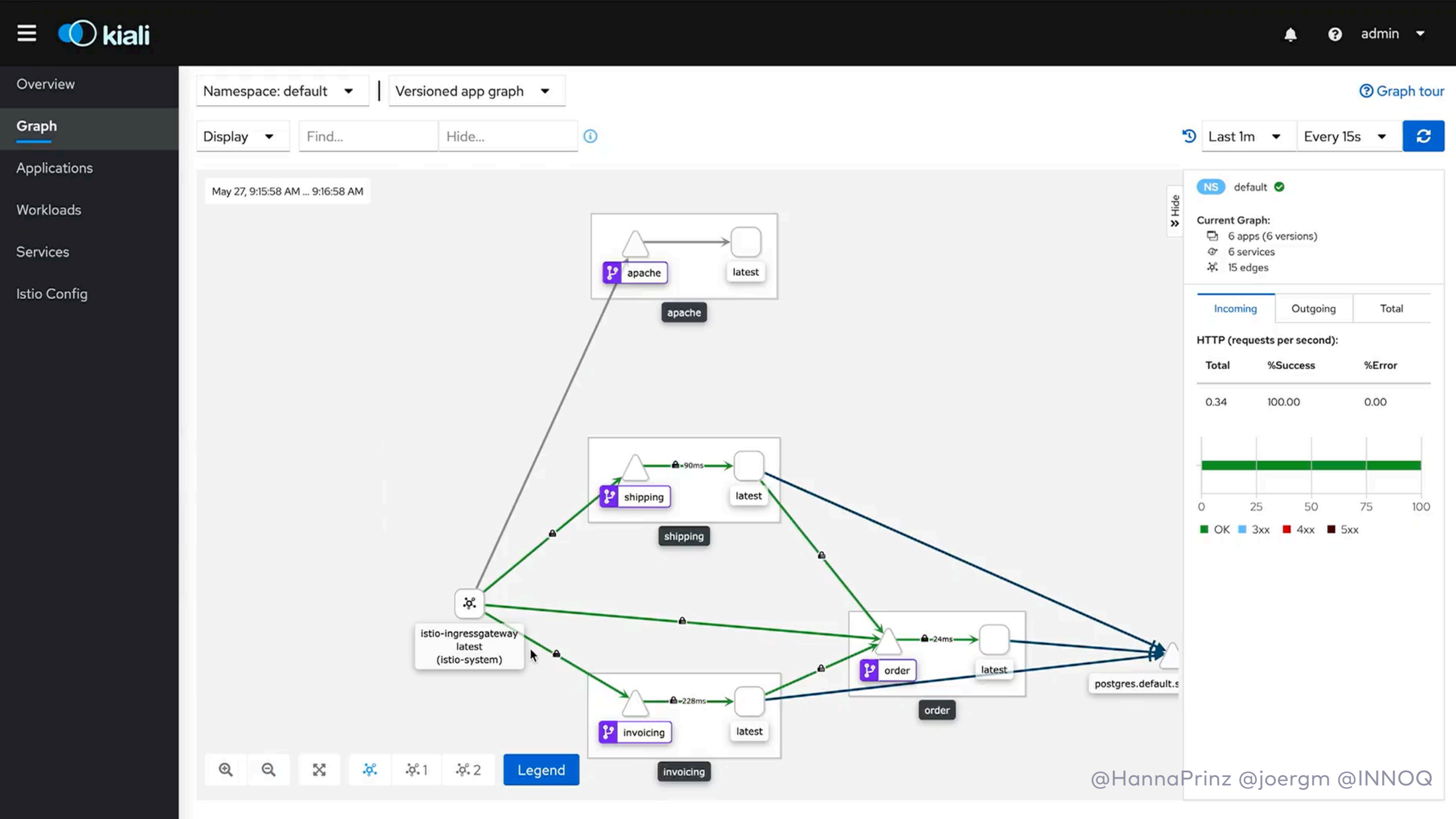


Deployments

Deployment	Meshed	Success Rate	RPS	P50 Latency	P95 Latency	P99 Latency	Grafana
apache	1/1	100.00%	0.42	1 ms	1 ms	1 ms	
invoicing	1/1	100.00%	0.83	6 ms	16 ms	19 ms	
order	1/1	100.00%	1.75	17 ms	29 ms	30 ms	
postgres	1/1	---	---	---	---	---	
shipping	1/1	100.00%	0.83	10 ms	19 ms	20 ms	

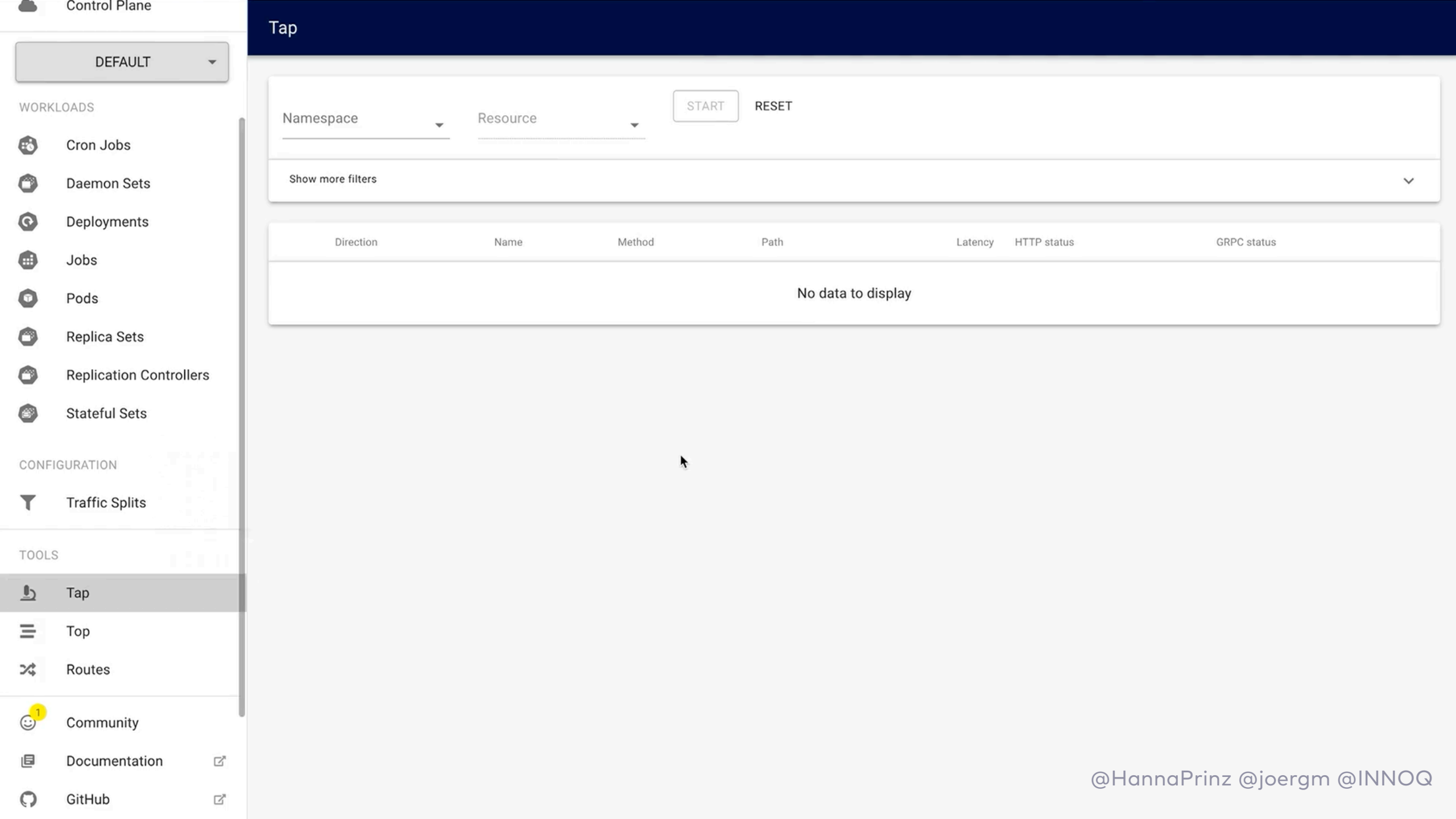
Pods

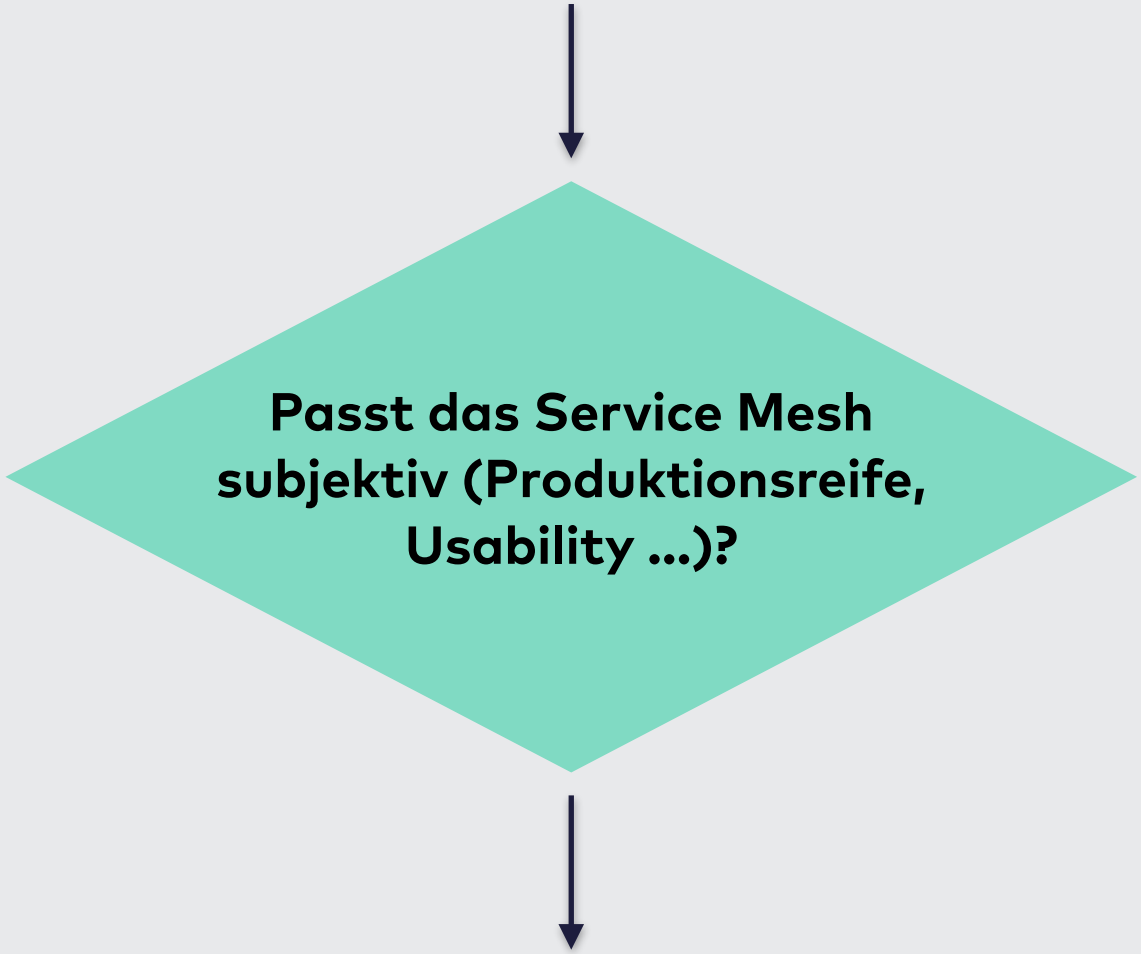
Pod	Meshed	Success Rate	RPS	P50 Latency	P95 Latency	P99 Latency	Grafana


kiali
admin
Overview
Graph
Applications
Workloads
Services
Istio Config
Namespace: default
Versioned app graph
Graph tour
Display
Find...
Hide...
Last 1m
Every 15s
May 27, 9:15:58 AM ... 9:16:58 AM
apache
latest
apache
shipping
90ms
latest
shipping
istio-ingressgateway
latest
(istio-system)
order
24ms
latest
order
invoicing
228ms
latest
invoicing
postgres.default.s
1
2
Legend
Hide
NS
default
Current Graph:
6 apps (6 versions)
6 services
15 edges
Incoming
Outgoing
Total
HTTP (requests per second):
Total
%Success
%Error
0.34
100.00
0.00
0
25
50
75
100
OK
3xx
4xx
5xx
@HannaPrinz @joergm @INNOQ

Observability Unterschiede

- Qualität des Dashboards
- Vorkonfiguriertes Prometheus, Grafana und Jaeger
- Tracing-Support
- Access Logs
  - Alternative von Linkerd: Echtzeit-Logs mit "tap"


Control Plane
DEFAULT
WORKLOADS
Cron Jobs
Daemon Sets
Deployments
Jobs
Pods
Replica Sets
Replication Controllers
Stateful Sets
CONFIGURATION
Traffic Splits
TOOLS
Tap
Top
Routes
Community
Documentation
GitHub
Tap
Namespace
Resource
START
RESET
Show more filters
Direction
Name
Method
Path
Latency
HTTP status
GRPC status
No data to display


Passt das Service Mesh subjektiv (Produktionsreife, Usability ...)?

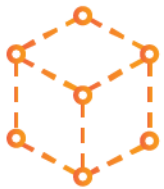
Produktionsreife

Spitzengruppe

Istio

LINKERD

Verfolgerfeld

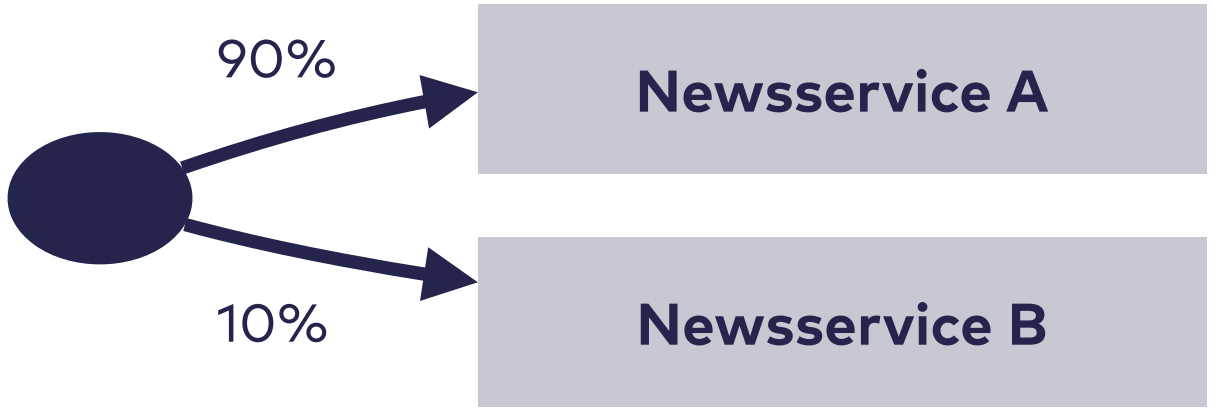


AWS App Mesh

Sehr Neu

Open Service Mesh

Komplexität der Konfiguration



Beispiel Traffic Split

Traffic Split Istio

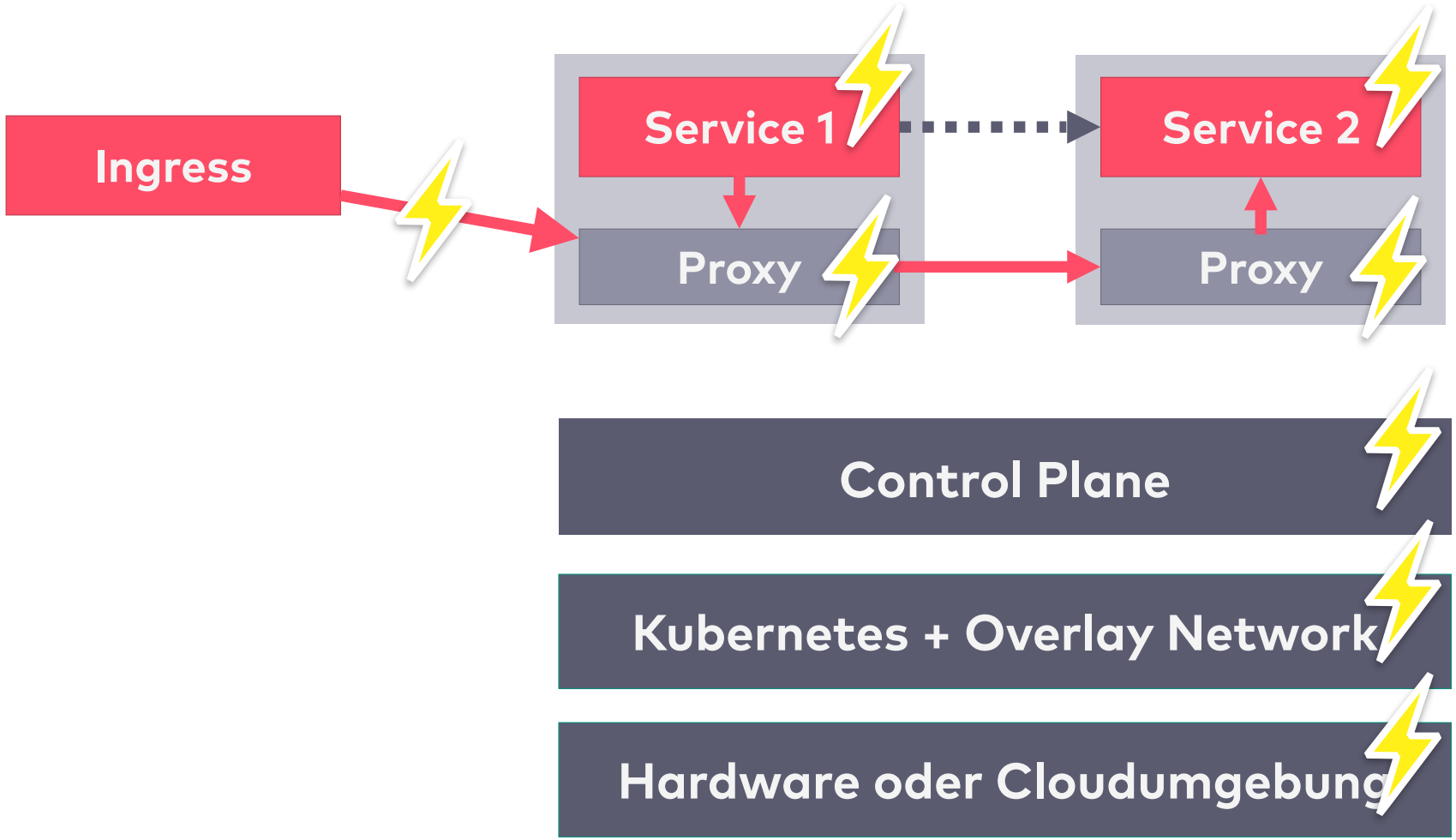
```yaml
apiVersion: networking.istio.io/v1alpha3
kind: VirtualService
metadata:
  name: newsservice-istio
spec:
  hosts:
  - newsservice
  http:
  - route:
    - destination:
        host: newsservice
        subset: A
      weight: 90
    - destination:
        host: newsservice
        subset: B
      weight: 10
```

```yaml
apiVersion: networking.istio.io/v1alpha3
kind: DestinationRule
metadata:
  name: newsservice-istio
spec:
  host: newsservice
  subsets:
  - name: A
    labels:
      version: "A"
  - name: B
    labels:
      version: "B"
```

Traffic Split SMI

```yaml
apiVersion: split.smi-spec.io/v1alpha1
kind: TrafficSplit
metadata:
  name: newsservice-smi
spec:
  service: newsservice
  backends:
  - service: newsservice-a
    weight: 90
  - service: newsservice-b
    weight: 10
```

Komplexes Debugging



Performance & Benchmarking

- Zusätzliche Latenz: ca. 3ms
- Zusätzliche CPU & Memory-Ressourcen
- Abhängig von Architektur, Traffic und Service Mesh

--> https://github.com/kinvolk/service-mesh-benchmark

→ Eigener Benchmark!

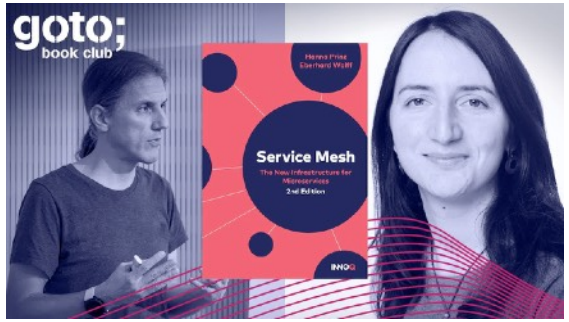
Fazit

Schritte zum Service Mesh

0. Ist ein Service Mesh der sinnvolle nächste Schritt?
   → liegt das Problem woanders?
1. Was ist die technische Umgebung?
   → Kubernetes/Cloud/Infrastruktur-Tools
2. Welche Features werden benötigt?
   → Must-Haves/Nice-to-Haves?
3. Passt das Service Mesh subjektiv?
   → Produktionsreife, Developer Experience, Config, Performance

Mehr zum Thema

- Service Mesh Vergleich auf servicemesh.es
- Gut gesiebt: Service Meshes im Vergleich
  https://www.innoq.com/de/articles/2020/10/service-meshes-im-vergleich/
- Artikel: Glücklich ohne Service Mesh
  https://www.innoq.com/de/blog/gluecklich-ohne-service-mesh/
- Podcast zu Service Meshes
  https://www.innoq.com/de/podcast/059-service-meshes-1/
- Beispiel-Anwendung auf GitHub
  https://github.com/ewolff/microservice-istio
- Tutorial von Linkerd
  https://linkerd.io/2/tasks/
- Tutorial von Istio
  https://istio.io/docs/setup/getting-started/
- Artikel: Alle 11 Minuten verliebt sich ein Microservice in Linkerd
  https://www.innoq.com/en/articles/2020/01/linkerd2/
- Beispiel-Anwendung mit Istio und Linkerd Tutorial on GitHub
  https://github.com/ewolff/microservice-istio https://github.com/ewolff/microservice-linkerd



GOTO 2020 • Getting started with Service Mesh
https://www.youtube.com/watch?v=w14ge2838Vs

Technology Lunch 2020 • Service Mesh
https://www.youtube.com/watch?v=cTfp7It1eEo

Danke! Fragen?

Hanna Prinz
hanna.prinz@innoq.com
@HannaPrinz

Jörg Müller
joerg.mueller@innoq.com
@joergm

Service Mesh Primer - 2nd Edition
Auf leanpub.com/service-mesh-primer

INNOQ
www.innoq.com

innoQ Deutschland GmbH

Krischerstr. 100
40789 Monheim am Rhein
Germany
+49 2173 3366-0

Ohlauer Str. 43
10999 Berlin
Germany
+49 2173 3366-0

Ludwigstr. 180E
63067 Offenbach
Germany
+49 2173 3366-0

Kreuzstr. 16
80331 München
Germany
+49 2173 3366-0

Hermannstrasse 13
20095 Hamburg
Germany
+49 2173 3366-0

innoQ Schweiz GmbH

Gewerbestr. 11
CH-6330 Cham
Switzerland
+41 41 743 0116