

# Managing Apps on Kubernetes with Istio

Introducing Istio



**Elton Stoneman**

Freelance Consultant and Trainer

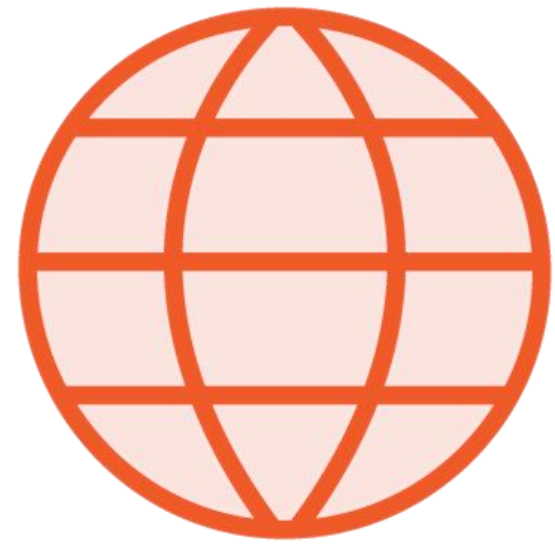
@EltonStoneman | [blog.sixeyed.com](http://blog.sixeyed.com)



# What is a service mesh?

It's the communication layer  
between software components,  
made into its own thing.

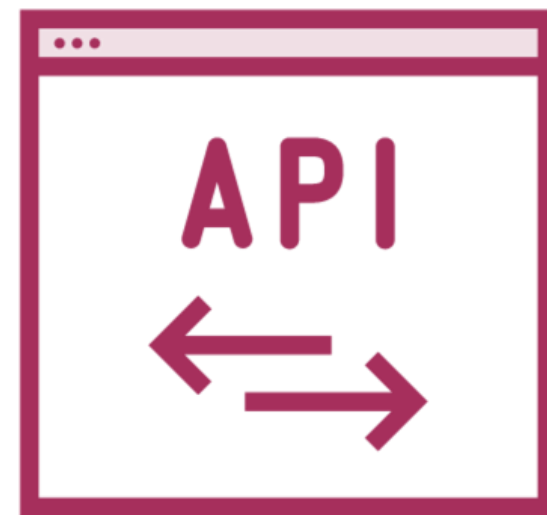




web



database

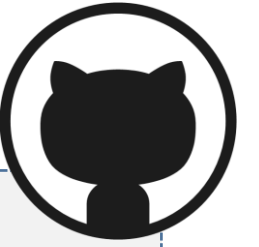


api

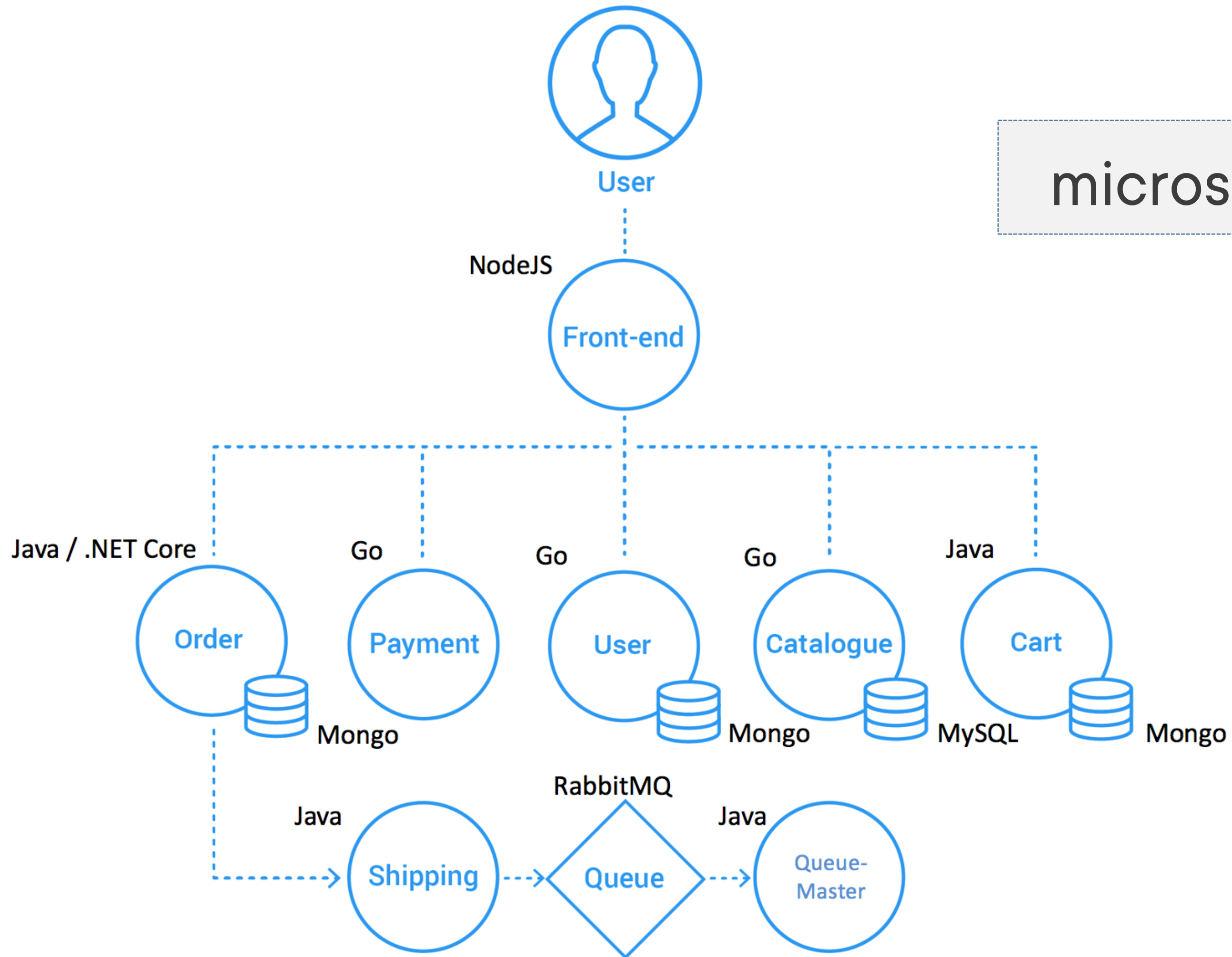
- **Address**
- **Timeout**
- **Retry**
- **Encryption**

- **Address**
- **Timeout**
- **Retry**
- **Encryption**

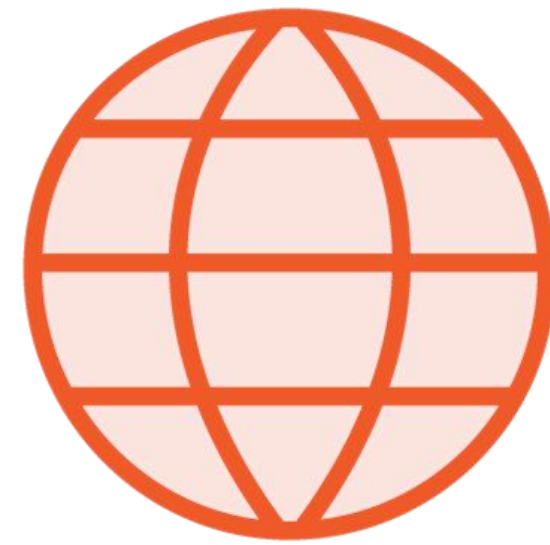




# microservices-demo



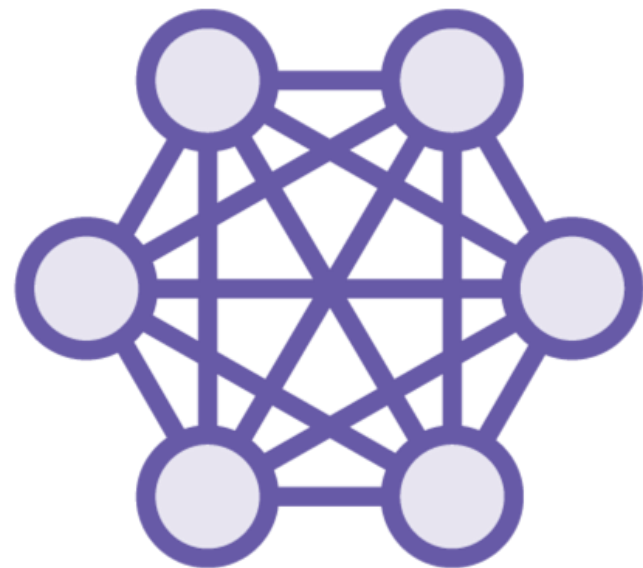
- Address
- Timeout
- Retry
- Encryption



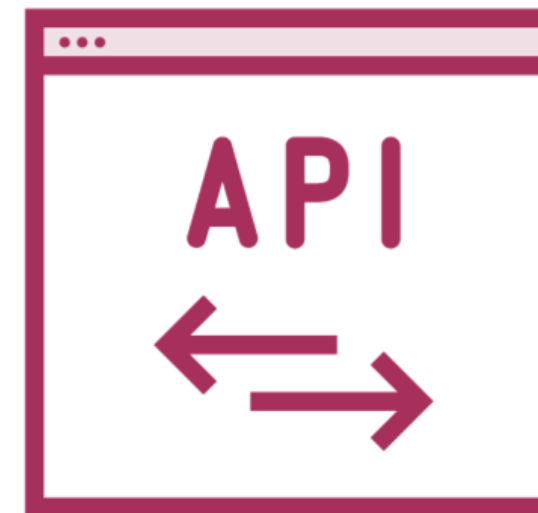
web



database

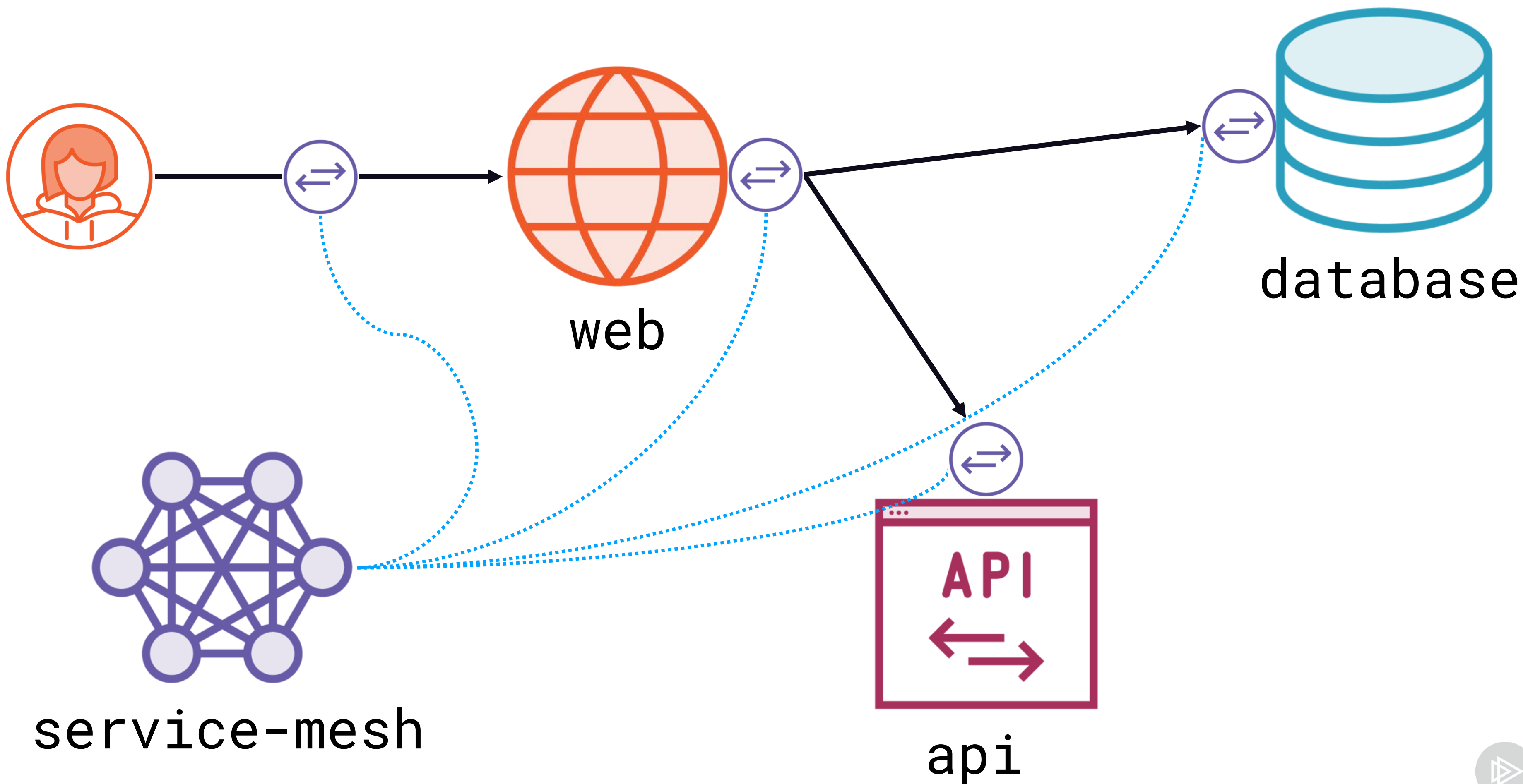


service-mesh



api





# Service Mesh Features



## Traffic Management

Discovery and routing  
Load balancing  
Failure handling



## Observability

Visualization  
Tracing  
Monitoring

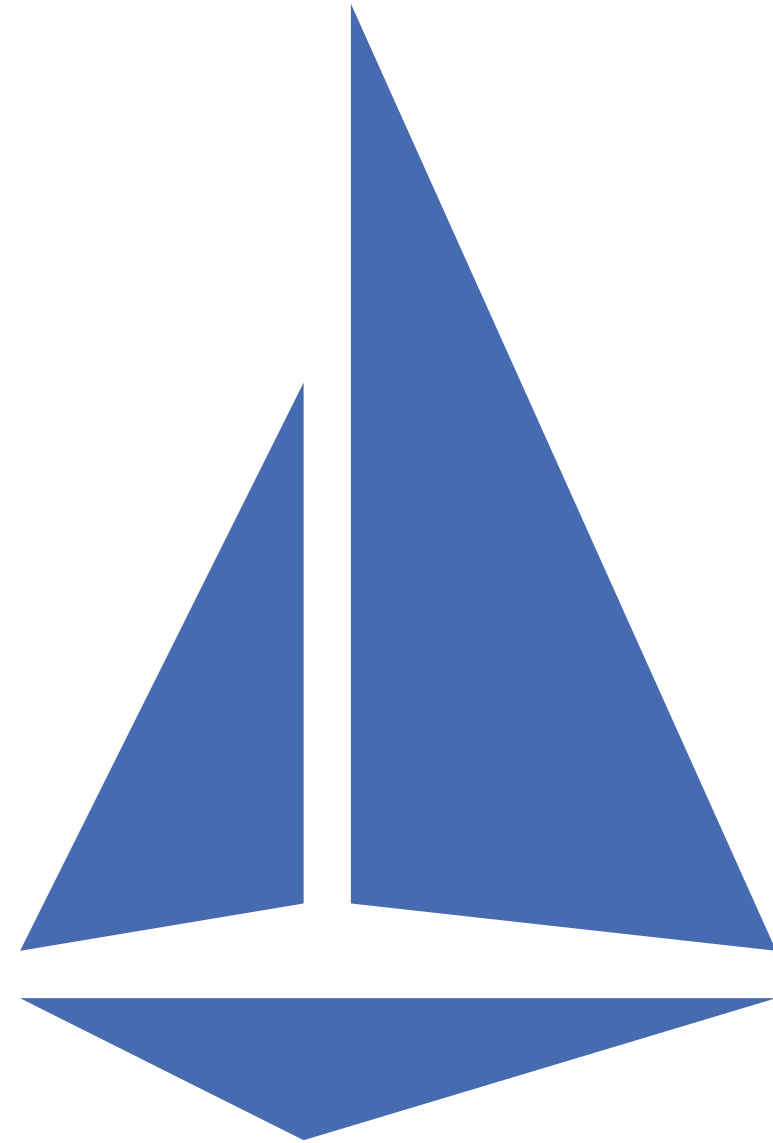


## Security

Encryption  
Authentication  
Authorization







## Introducing Istio

- Open source from Google, IBM, and Lyft
- 0.1 in 2017, 1.0 in 2018, currently 1.19

## Kubernetes add-on

- Extends core features
- Integrates VMs to the cluster

## One of many service meshes

- Linkerd, Consul, Cilium, OSM
- SMI (Service Mesh Interface)





# Demo



## Installing Istio

- Kubernetes setup
- Installing with Helm
- Exploring proxy containers





# Istio's Architecture and Costs



```
helm install istio-base istio/base
```

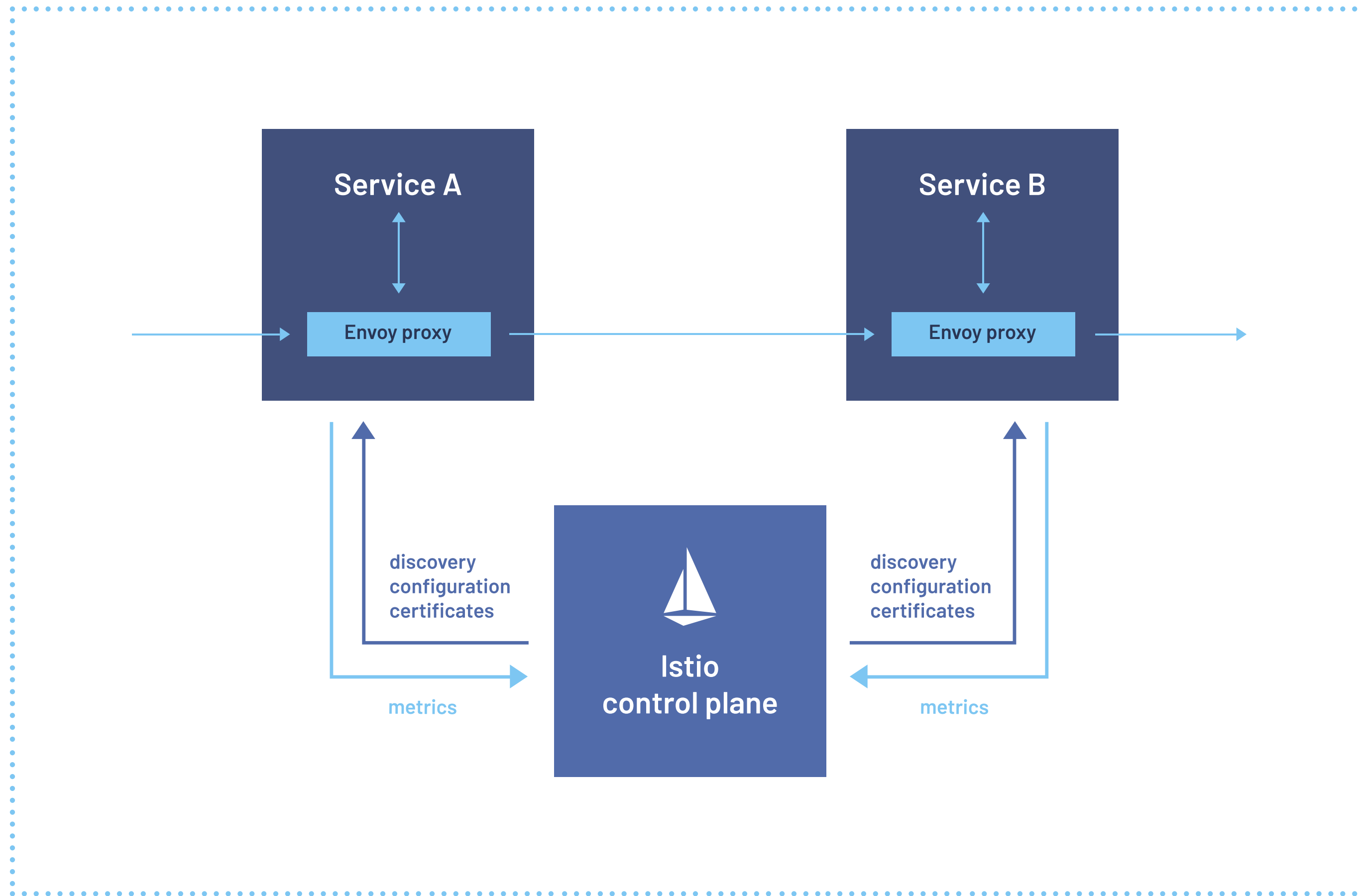
```
helm install istiod istio/istiod
```

```
helm install istio-ingress istio/gateway
```

## Deploying Istio

Standard Helm deployments: CRDs, control plane, ingress gateway



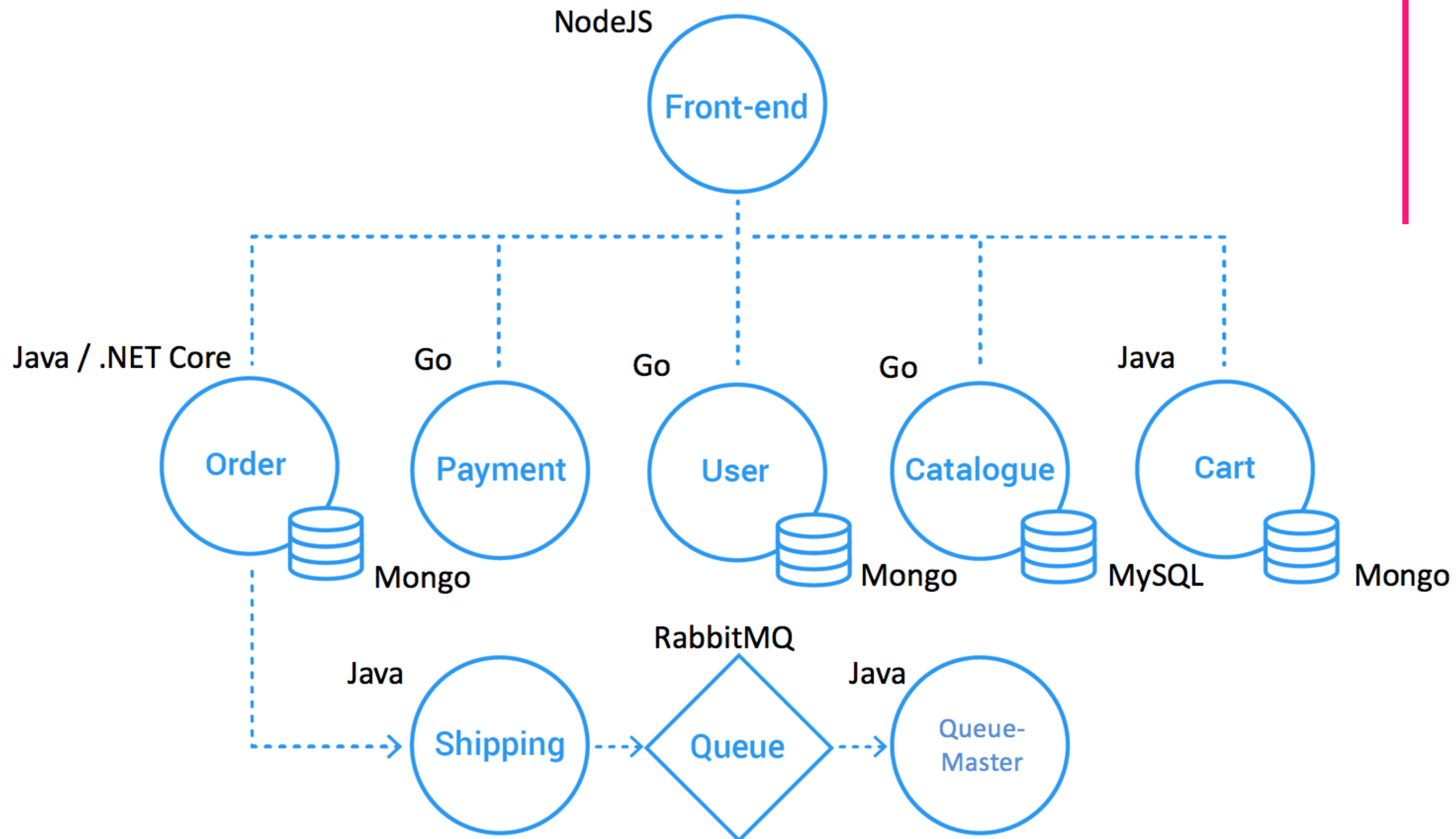


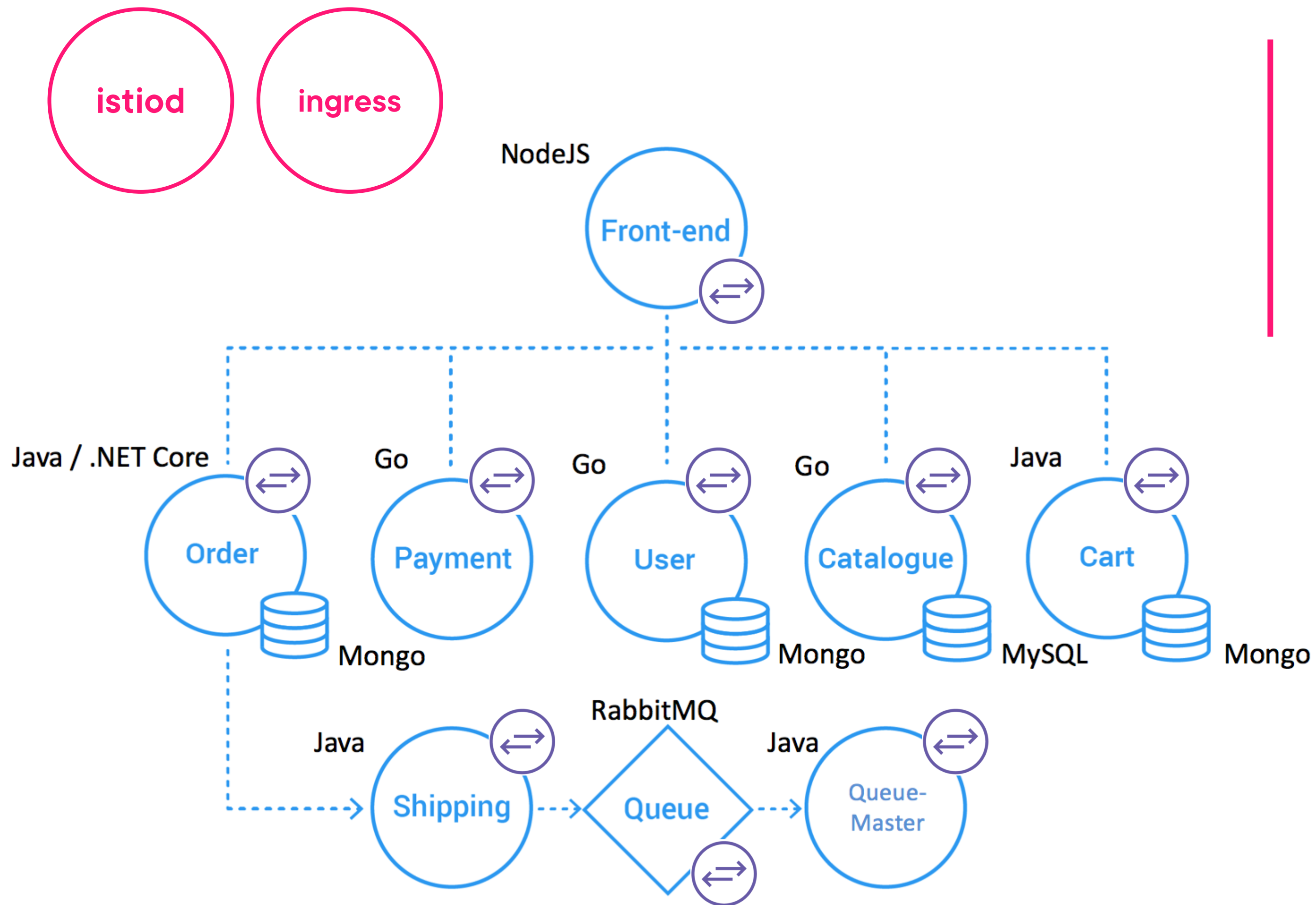
Michael Kipper <https://is.gd/ZqwLLb>

**“Here at Shopify, we’re working on deploying Istio as our service mesh. We’re doing quite well, but are hitting a wall: Cost.”**



- **9x Components**
- **30x Pods**
- **30x Containers**
- **5x Nodes**





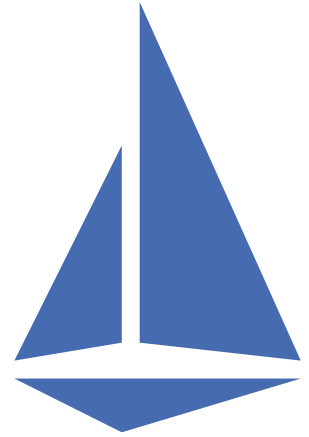
- **11x Components**
- **34x Pods**
- **64x Containers**
- **8x Nodes**







## *Lines of Code*



istio/istio

730K

envoyproxy/envoy

1.6M

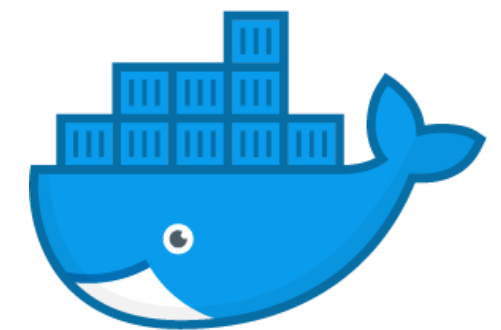


kubernetes/kubernetes

7.1M

containerd/containerd

1.4M



# Istio Bookinfo Demo App

BookInfo Sample

Sign in

## The Comedy of Errors

Summary: [Wikipedia Summary](#): The Comedy of Errors is one of **William Shakespeare's** early plays. It is his shortest and one of his most farcical comedies, with a major part of the humour coming from slapstick and mistaken identity, in addition to puns and word play.

### Book Details

Type:

paperback

Pages:

200

Publisher:

PublisherA

Language:

English

ISBN-10:

1234567890

ISBN-13:

123-1234567890

### Book Reviews

An extremely entertaining play by Shakespeare. The slapstick humour is refreshing!

— Reviewer1

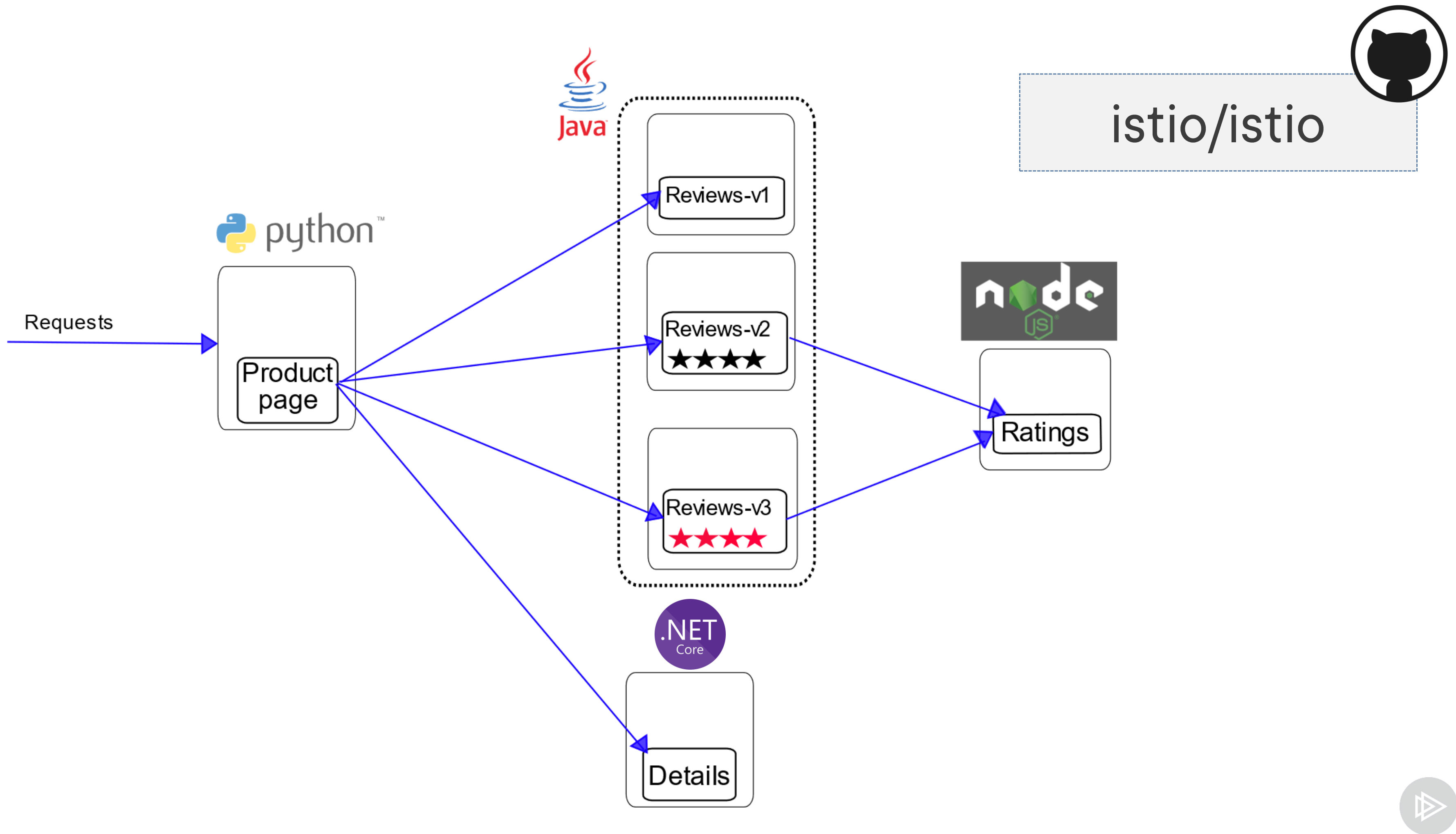
Absolutely fun and entertaining. The play lacks thematic depth when compared to other plays by Shakespeare.

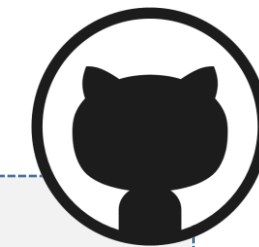
— Reviewer2

Reviews served by:

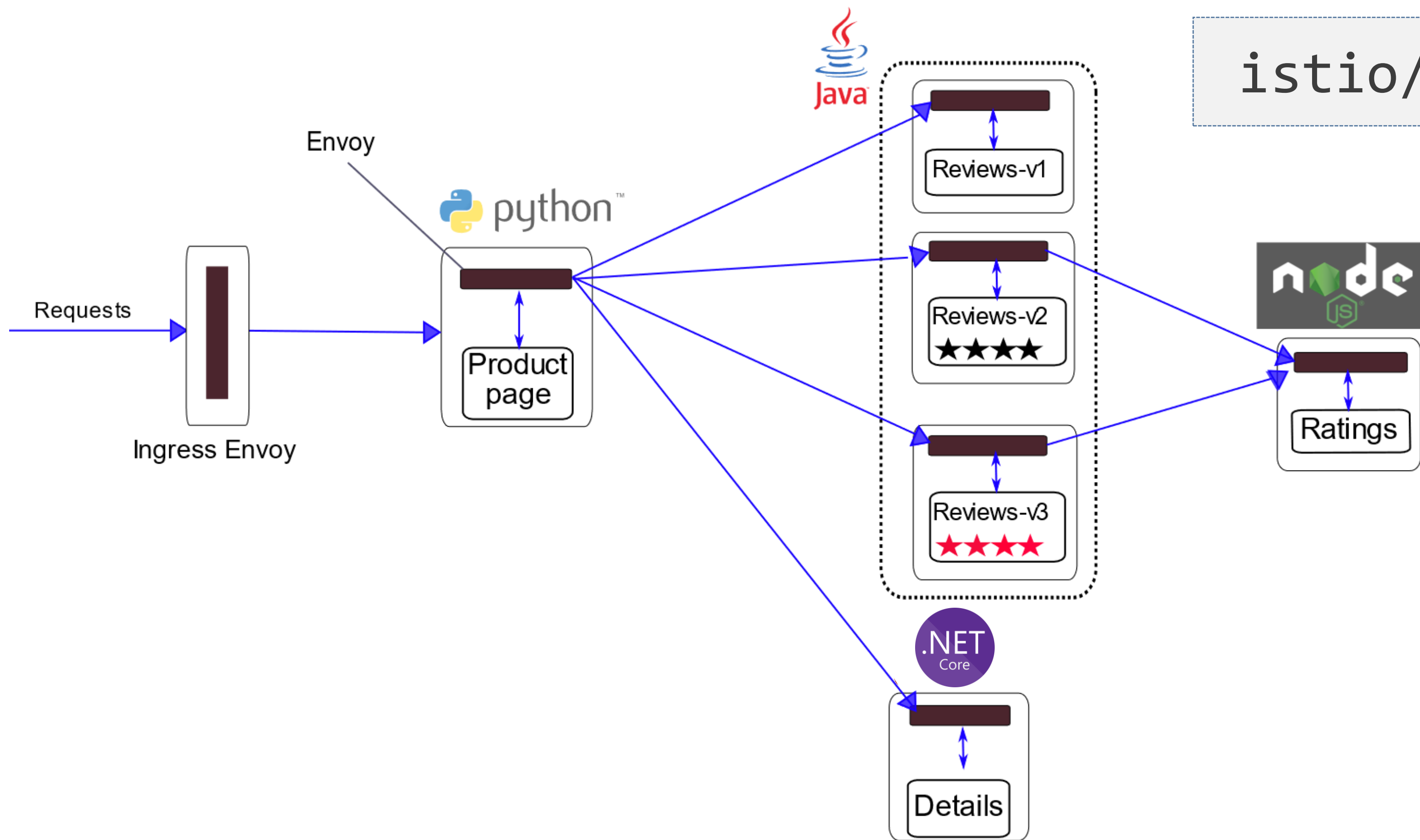
[reviews-v1-86896b7648-cttq8](#)

A circular button with a right-pointing arrow and a square icon inside, typically used for navigating to the next slide or page.





istio/istio



# Demo



## Deploying an Istio app

- Services and deployments
- Bookinfo without Istio
- Bookinfo with Istio and Gateway



# Using VirtualServices to Manage Traffic



```
kubectl label namespace bookinfo istio-injection=enabled
```

```
kubectl apply -f bookinfo.yaml
```

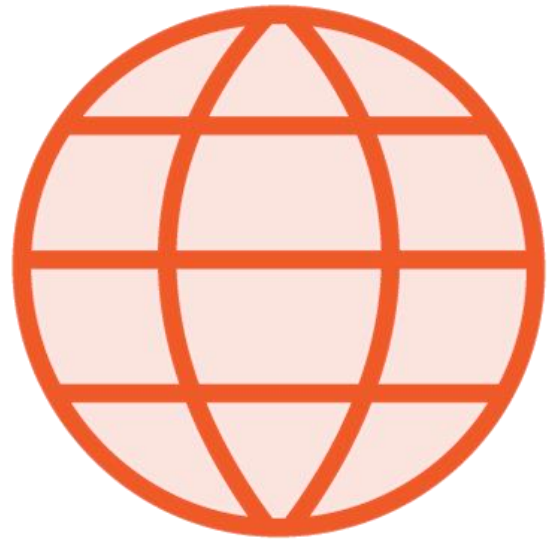
```
kubectl apply -f bookinfo-gateway.yaml
```

## Deploying BookInfo

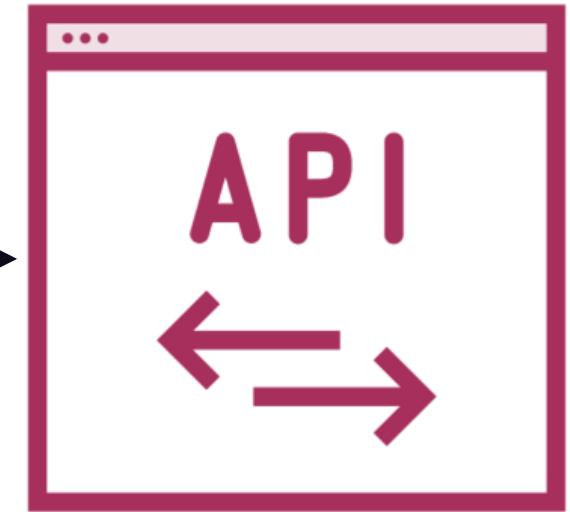
Standard Kubernetes deployment – with Istio injection and gateway



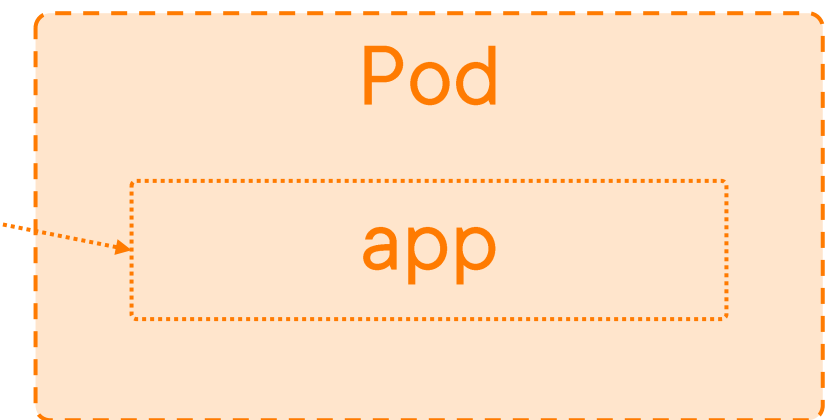
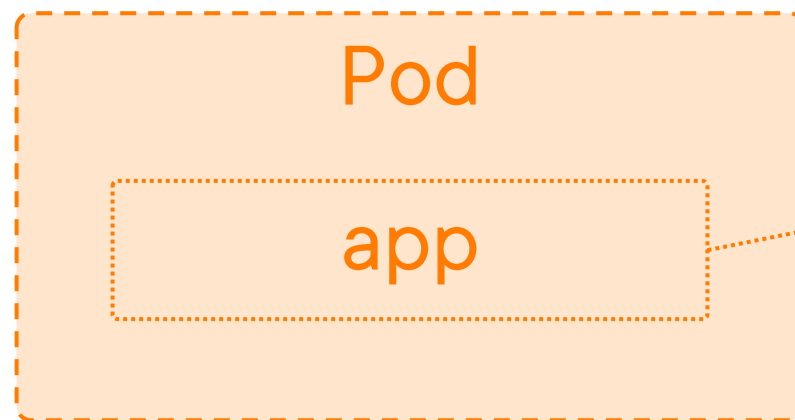


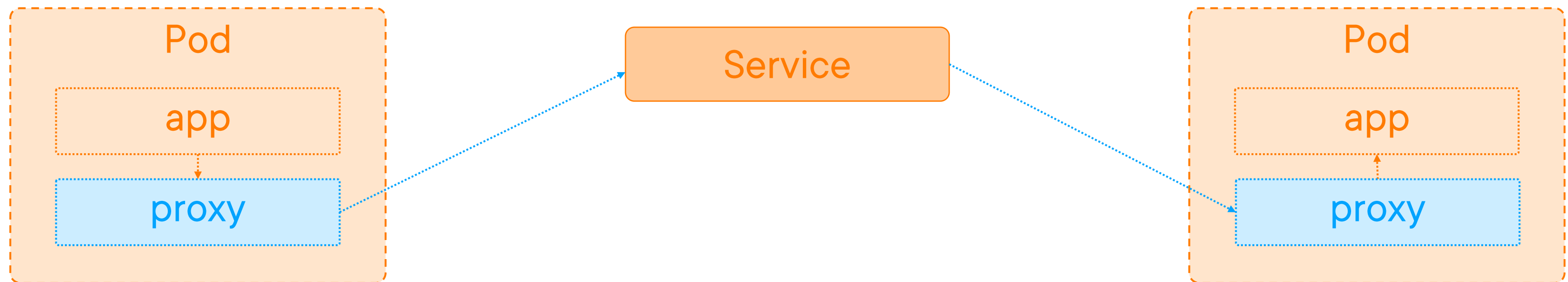
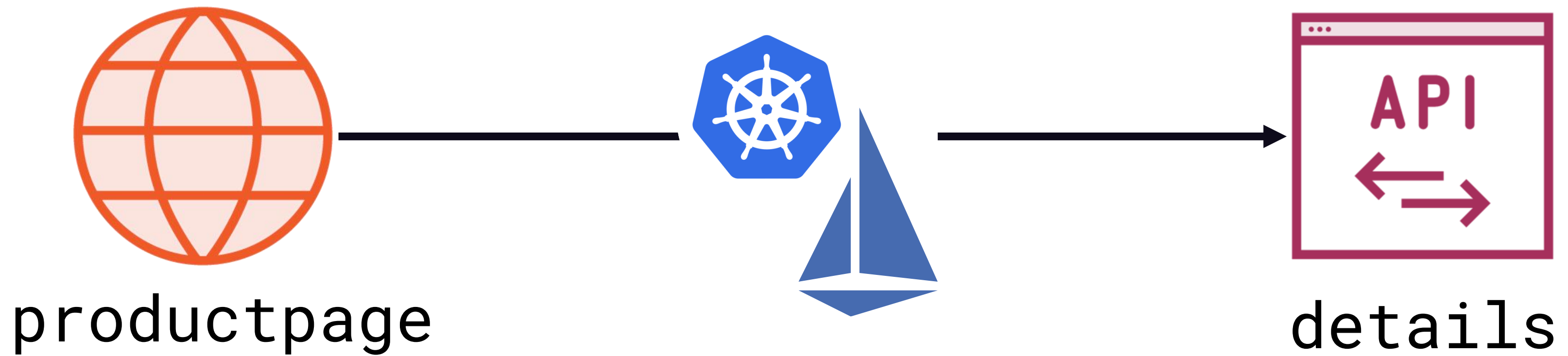


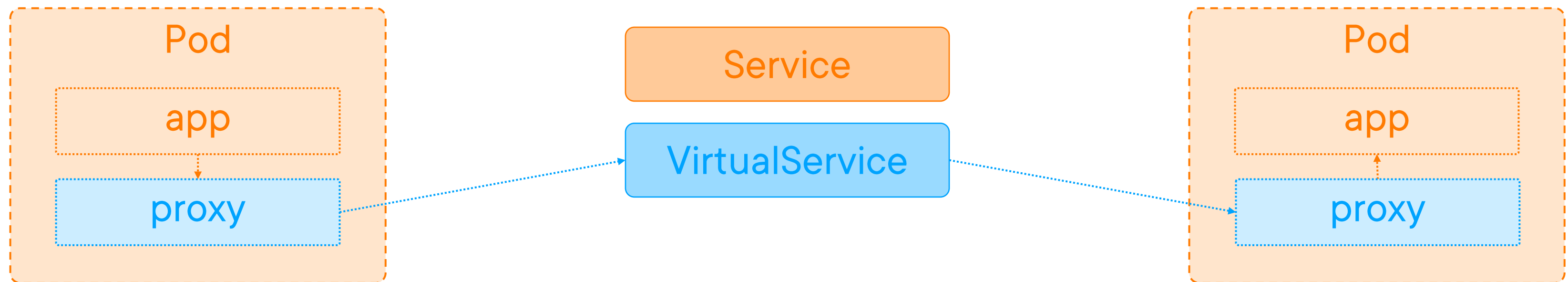
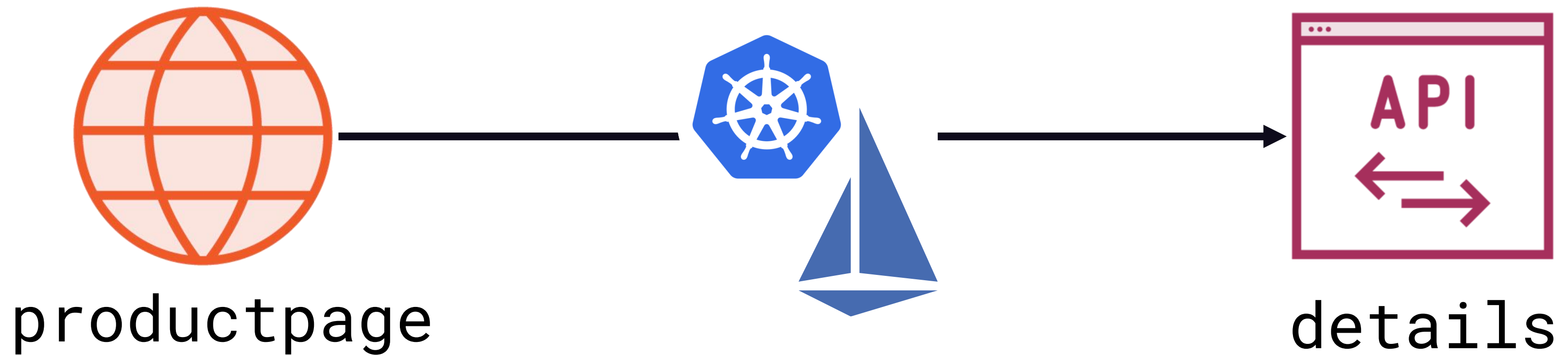
productpage



details







```
apiVersion: networking.istio.io/v1beta1
```

```
kind: VirtualService
```

```
spec:
```

```
  hosts:
```

```
    - details
```

```
  http:
```

```
    - route:
```

```
      - destination:
```

```
        host: details
```

◀ **Istio custom resource**

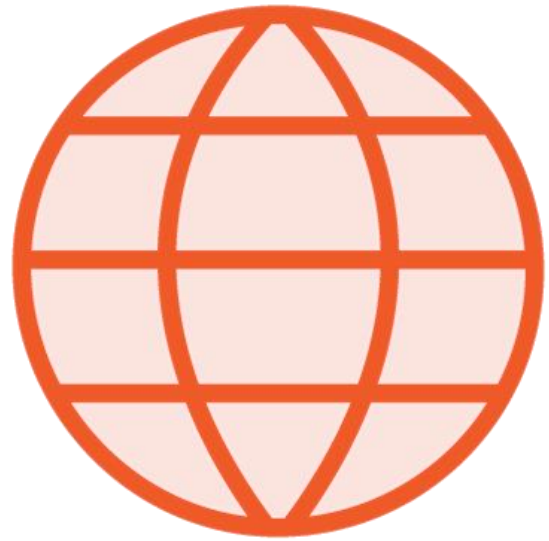
◀ **Defines routing rules**

◀ **The host DNS requested by the client**

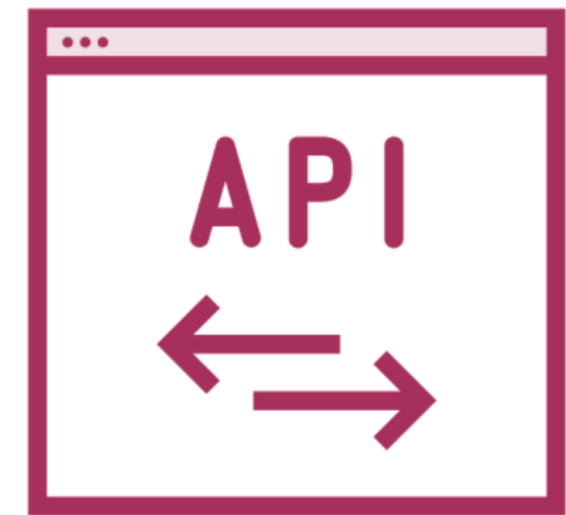
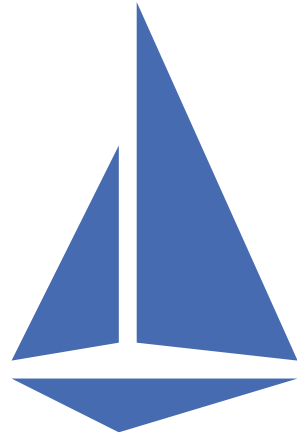
◀ **Routing for HTTP/S traffic**

◀ **The destination host DNS routed by Istio**



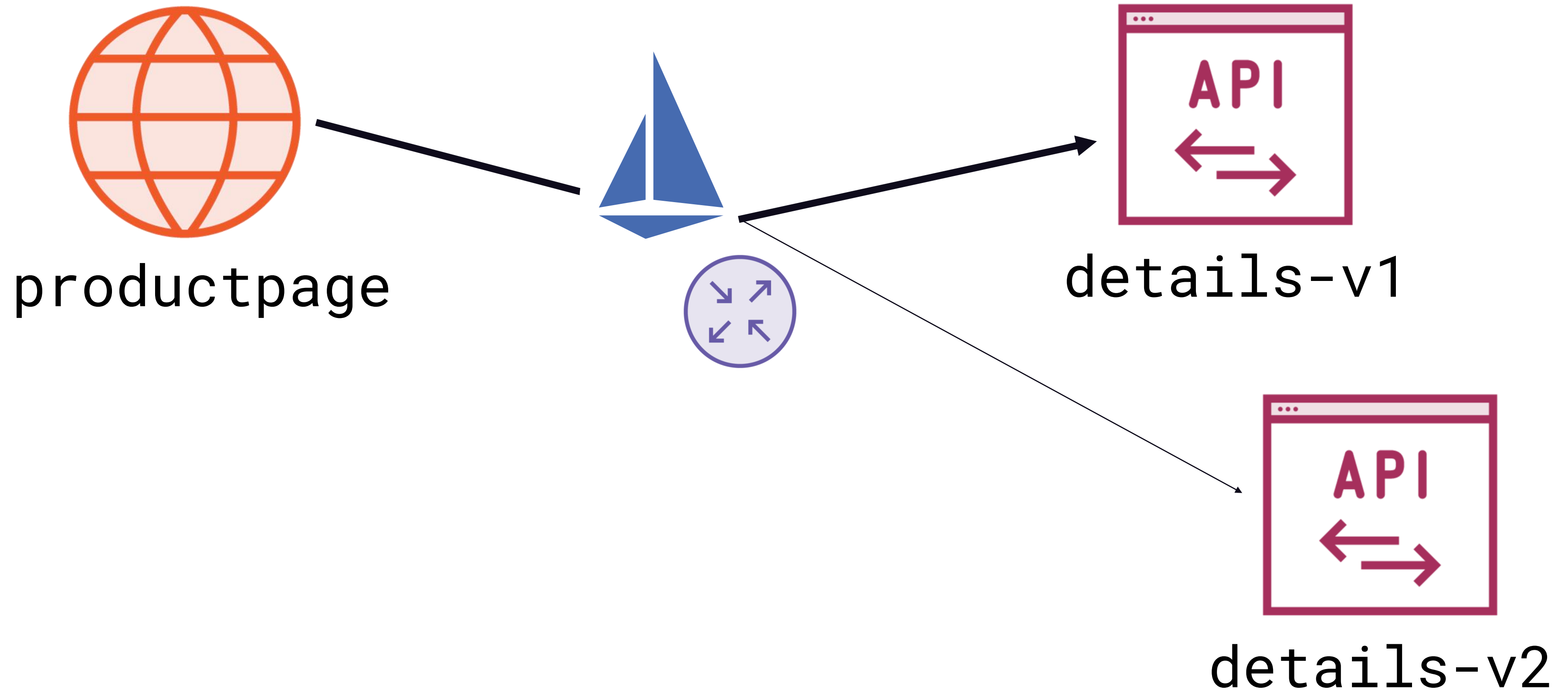


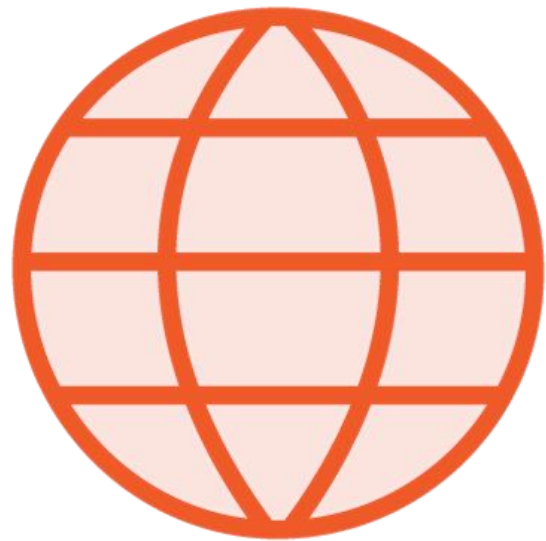
productpage



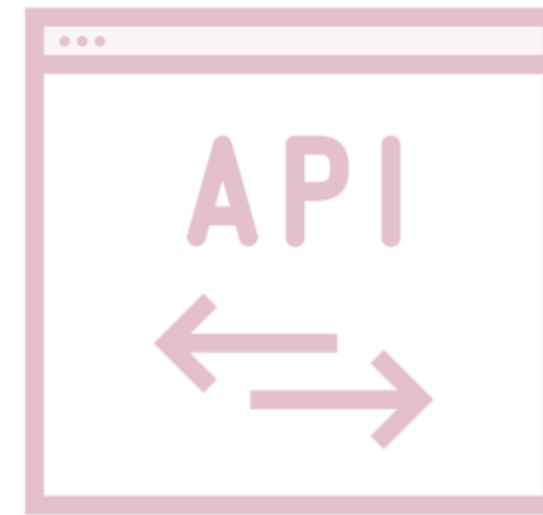
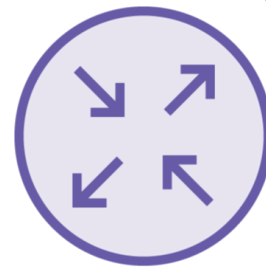
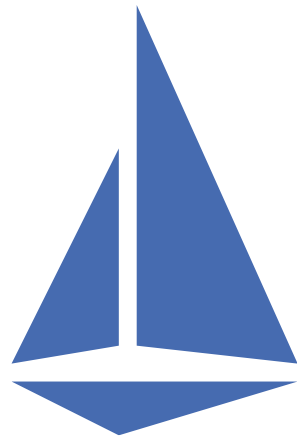
details



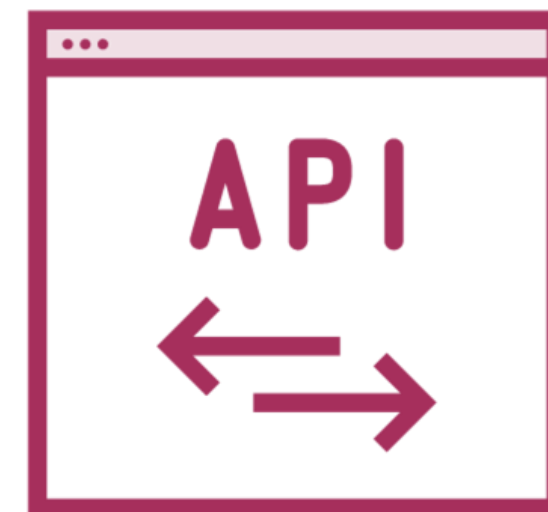




productpage



details-v1



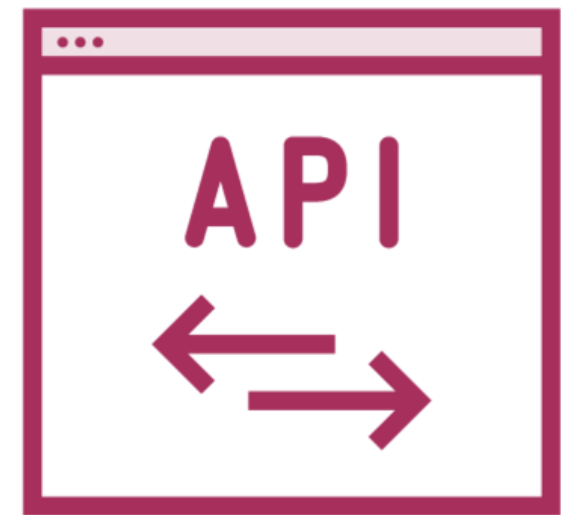
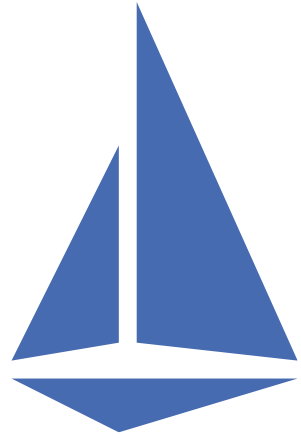
details-v2





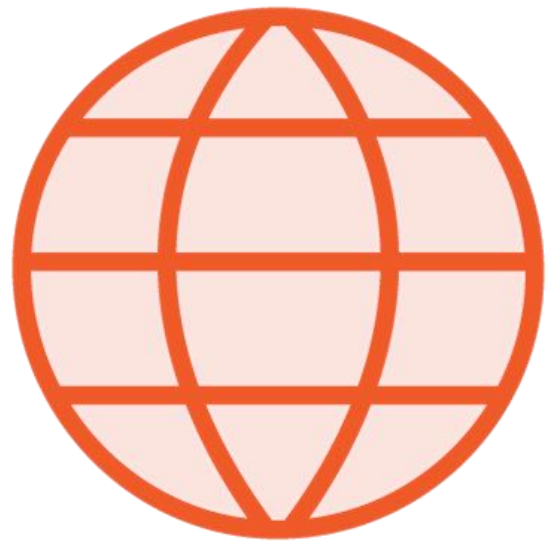


productpage

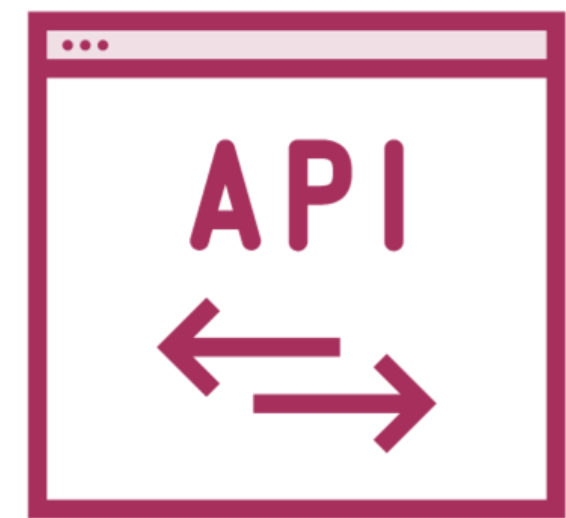
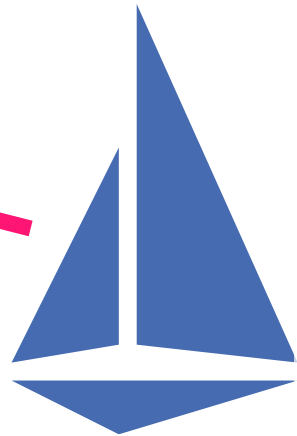


details



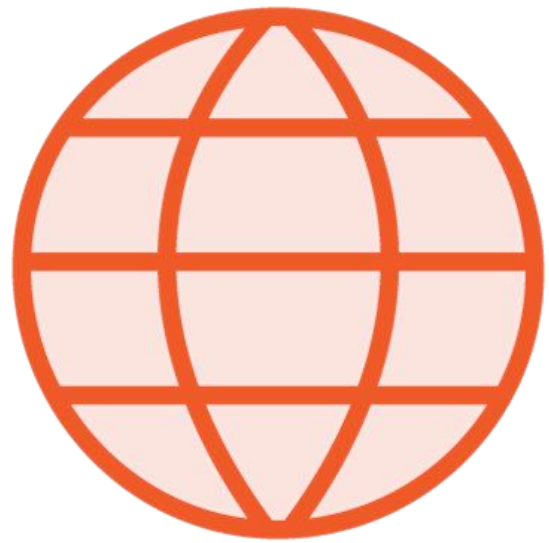


productpage

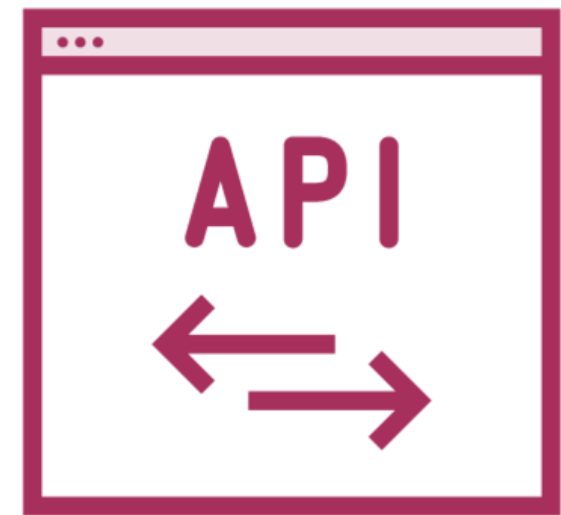
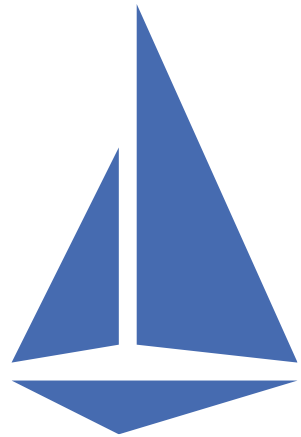


details



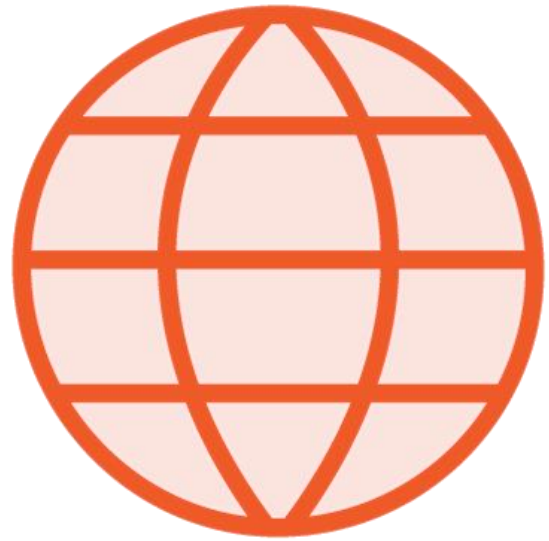


productpage

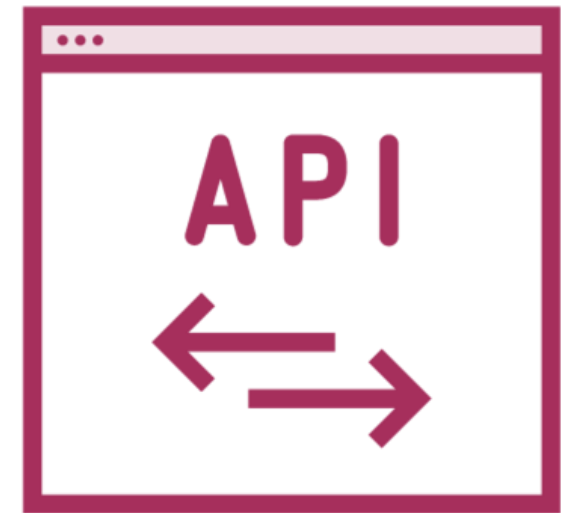
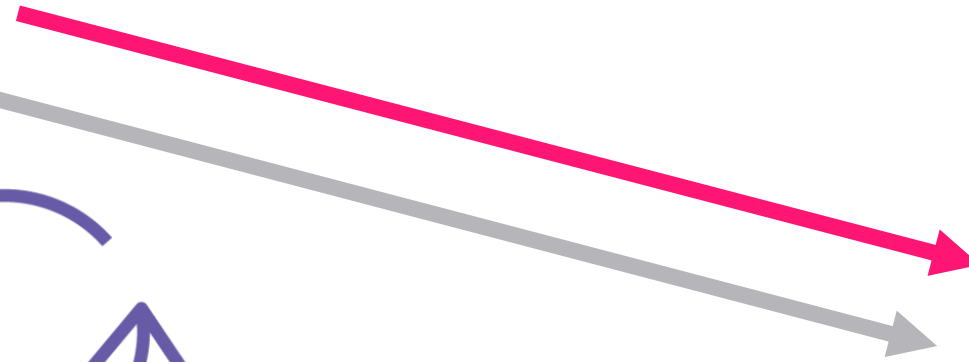
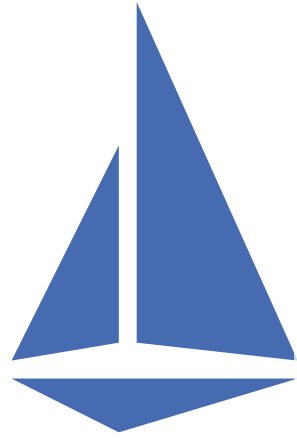


details





productpage



details



# Demo



## Managing traffic with Istio

- Fault-tolerance
- Client timeouts
- Automatic retries



# Traffic Routing

## virtual-service.yaml

```
apiVersion: networking.istio.io/v1beta1
```

```
kind: VirtualService
```

```
metadata:
```

```
  name: details
```

```
spec:
```

```
  hosts:
```

```
  - details
```

```
  http:
```

```
  - route:
```

```
    - destination:
```

```
      host: details
```



# Network Timeouts

## virtual-service.yaml

```
apiVersion: networking.istio.io/v1beta1
kind: VirtualService
metadata:
  name: details
spec:
  hosts:
  - details
  http:
  - route:
    - destination:
        host: details
      timeout: 5s
```



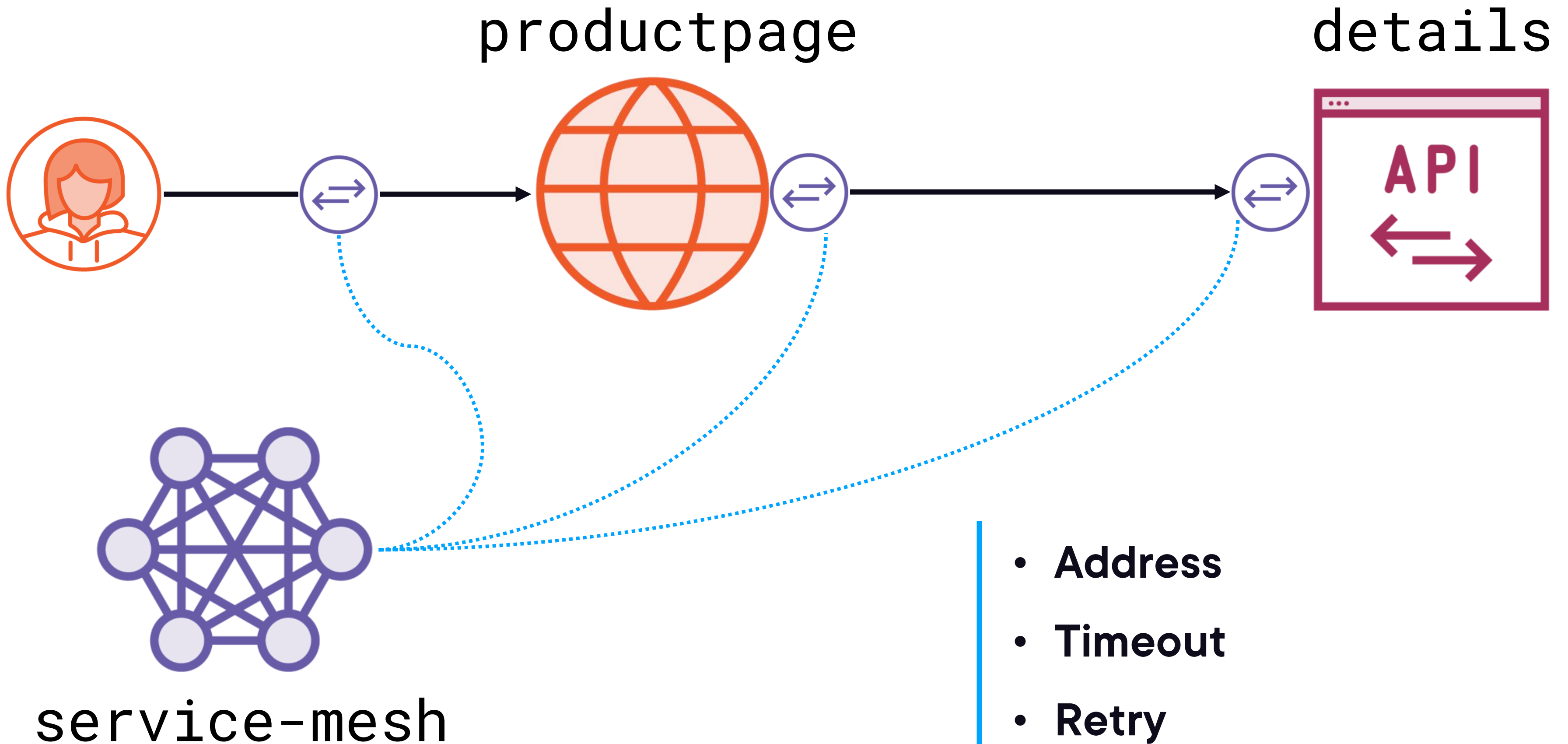


# Network Retries

## virtual-service.yaml

```
apiVersion: networking.istio.io/v1beta1
kind: VirtualService
metadata:
  name: details
spec:
  hosts:
  - details
  http:
  - route:
    - destination:
        host: details
      timeout: 5s
    retries:
      retryOn: 5xx
      perTryTimeout: 2s
      attempts: 2
```





## Summary



### **Istio is a communication control plane**

- Runs in Kubernetes (and more)
- Deployment with Helm (and more)
- Injects network proxies

### **Creating a service mesh**

- Traffic management
- Security
- Observability

**Up Next:**

# **Managing Service Traffic**

---

