

Managing Apps on Kubernetes with Istio

INTRODUCING ISTIO



Kien Bui

DevOps & Platform Engineer

What is a service mesh?

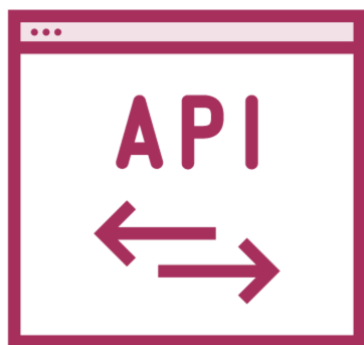
It's the communication
between software components,
made into its own thing.



web



database



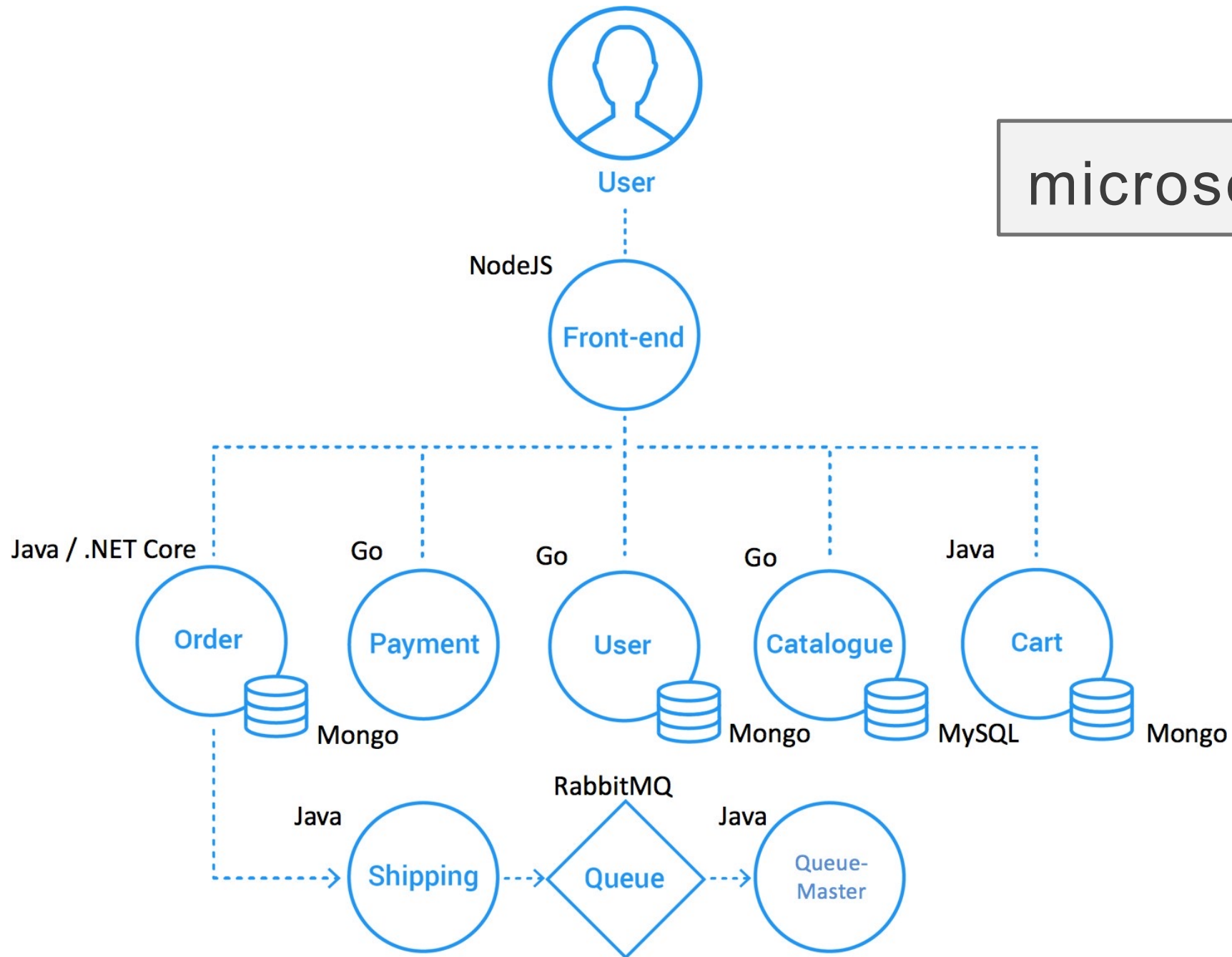
api

- Address
- Timeout
- Retry
- Encryption

- Address
- Timeout
- Retry
- Encryption



microservices-demo



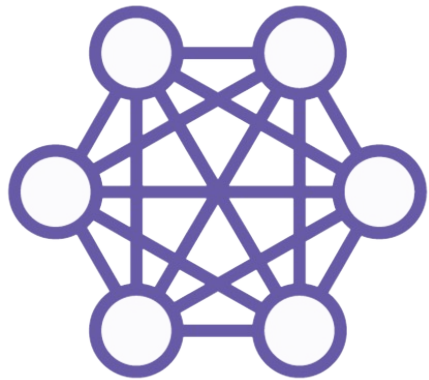
- Address
- Timeout
- Retry
- Encryption



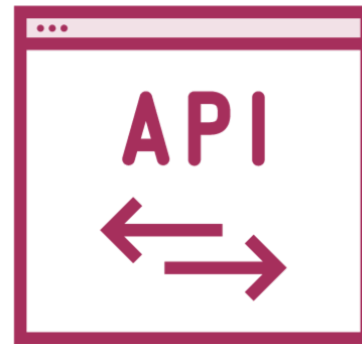
web



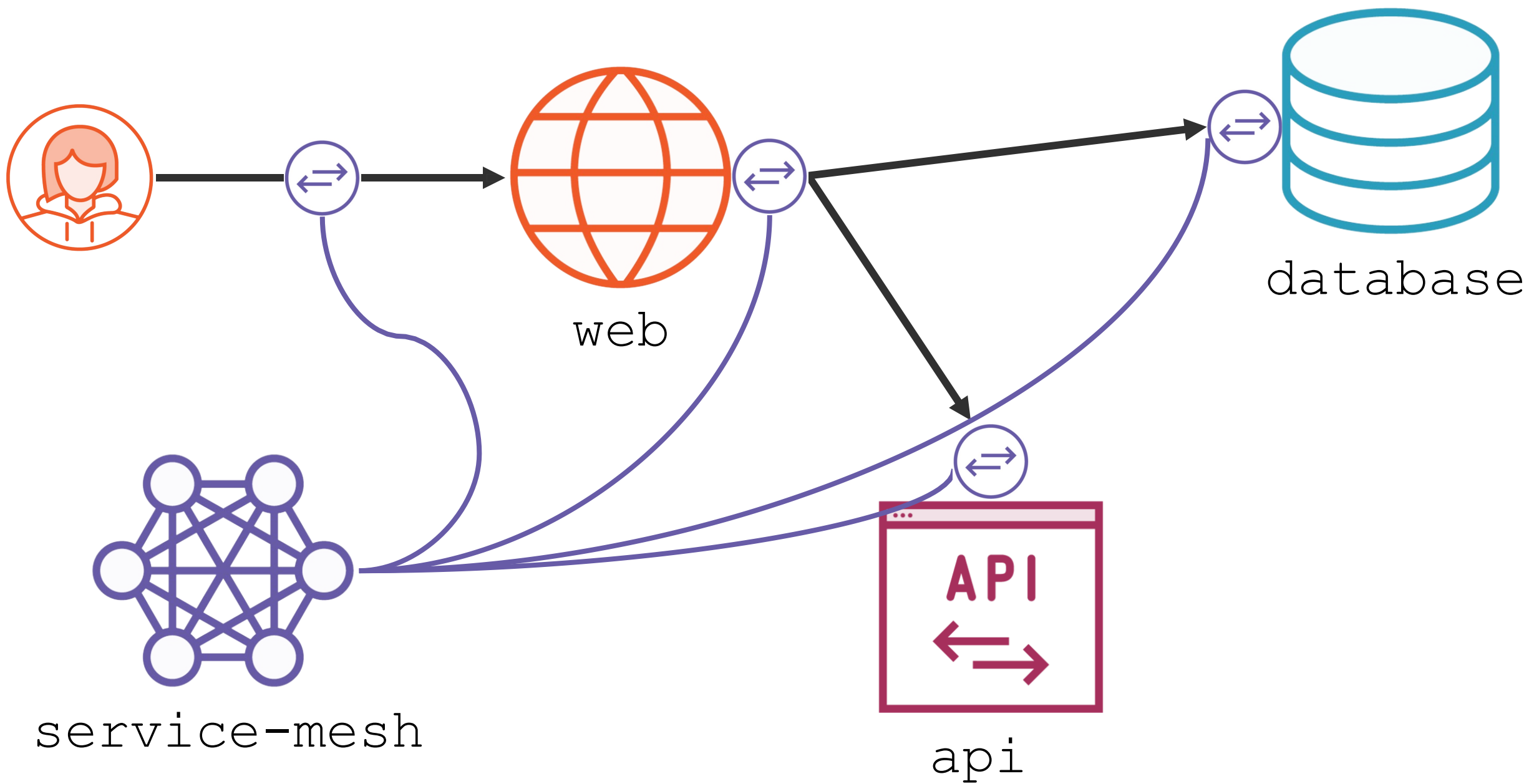
database



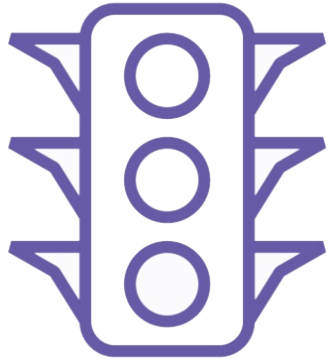
service-mesh



api



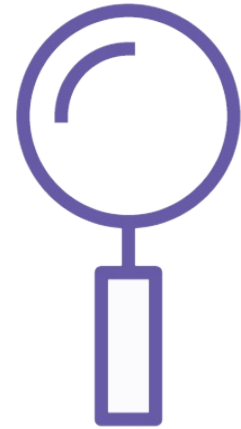
Service Mesh Features



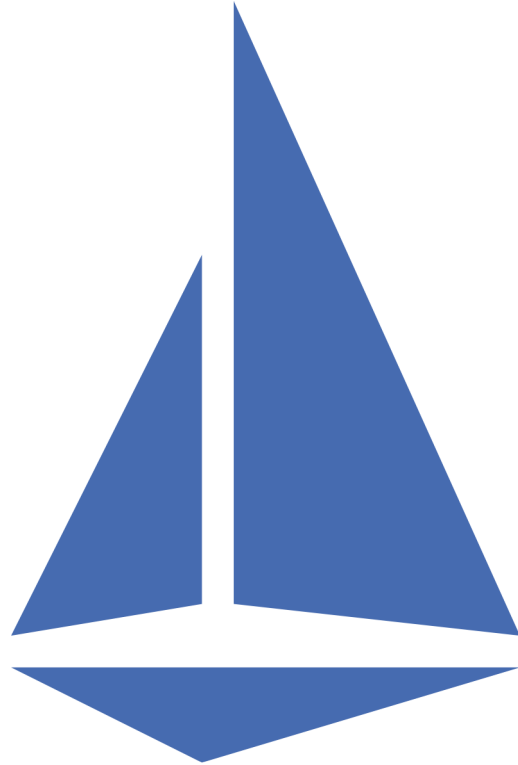
Traffic Management
Discovery & routing
Load-balancing
Failure handling



Security
Encryption
Authentication
Authorization



Observability
Visualization
Tracing
Monitoring



Introducing Istio

- Open source from Google, IBM & Lyft
- 0.1 in 2017, 1.0 in 2018, currently 1.4

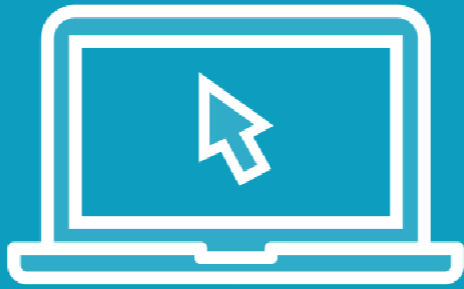
Platform-agnostic

- Optimal with Kubernetes
- Integrates with Consul or VMs

One of many servicemeshes

- Linkerd, Consul Connect, Maesh
- SMI (Service Mesh Interface)

Demo



Installing Istio

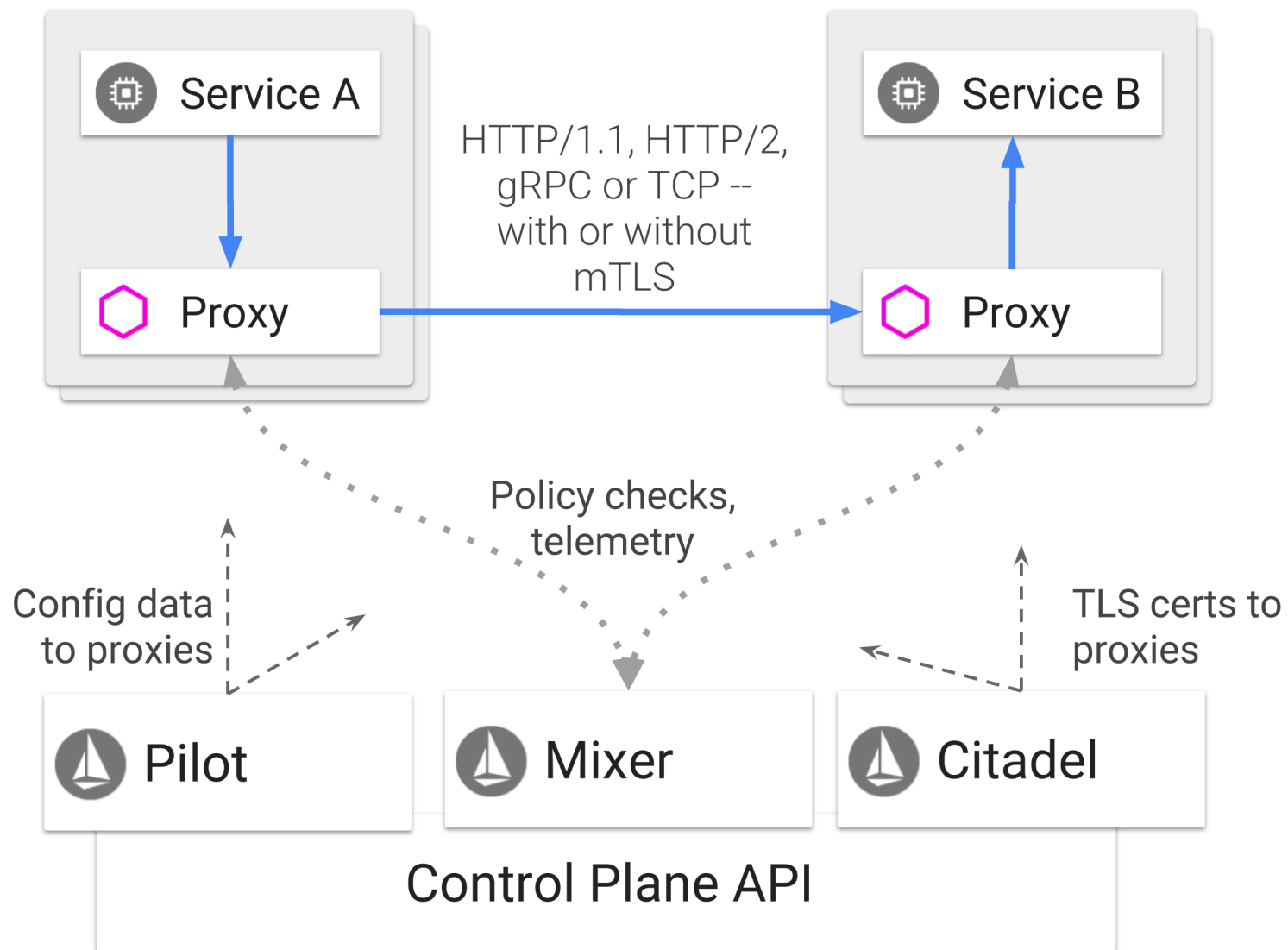
- Pre-requisites
- "Demo" configuration
- Installing with Docker Desktop

```
kubectl apply -f istio-crds.yaml
```

```
kubectl apply -f istio-demo.yaml
```

Deploying Istio

Standard Kubernetes deployment



* <https://istio.io/docs/concepts/what-is-istio/>

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

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



Lines of Code

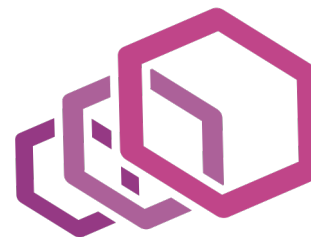


istio/istio

410K

envoyproxy/envoy

340K

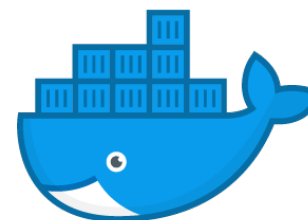


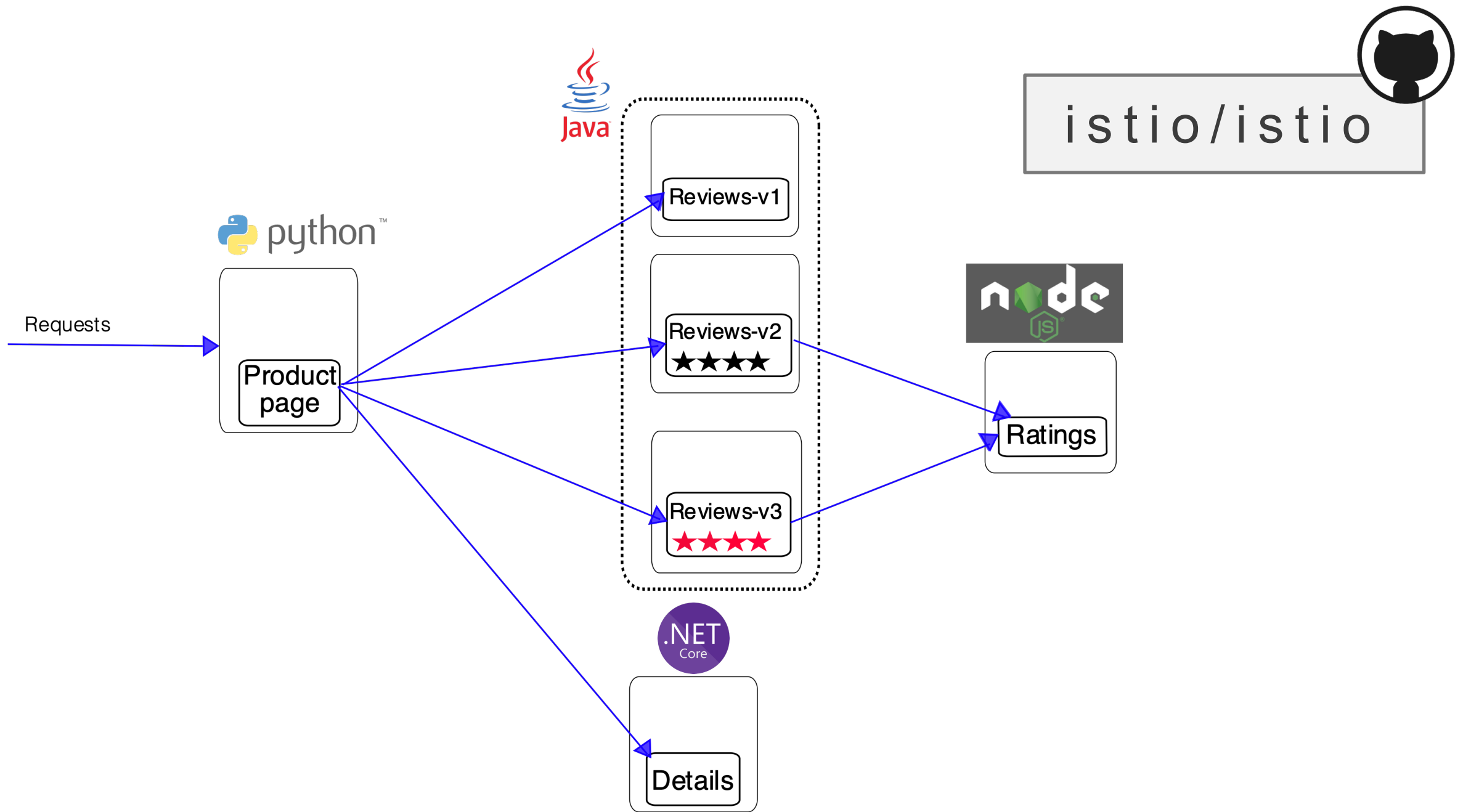
kubernetes/kubernetes

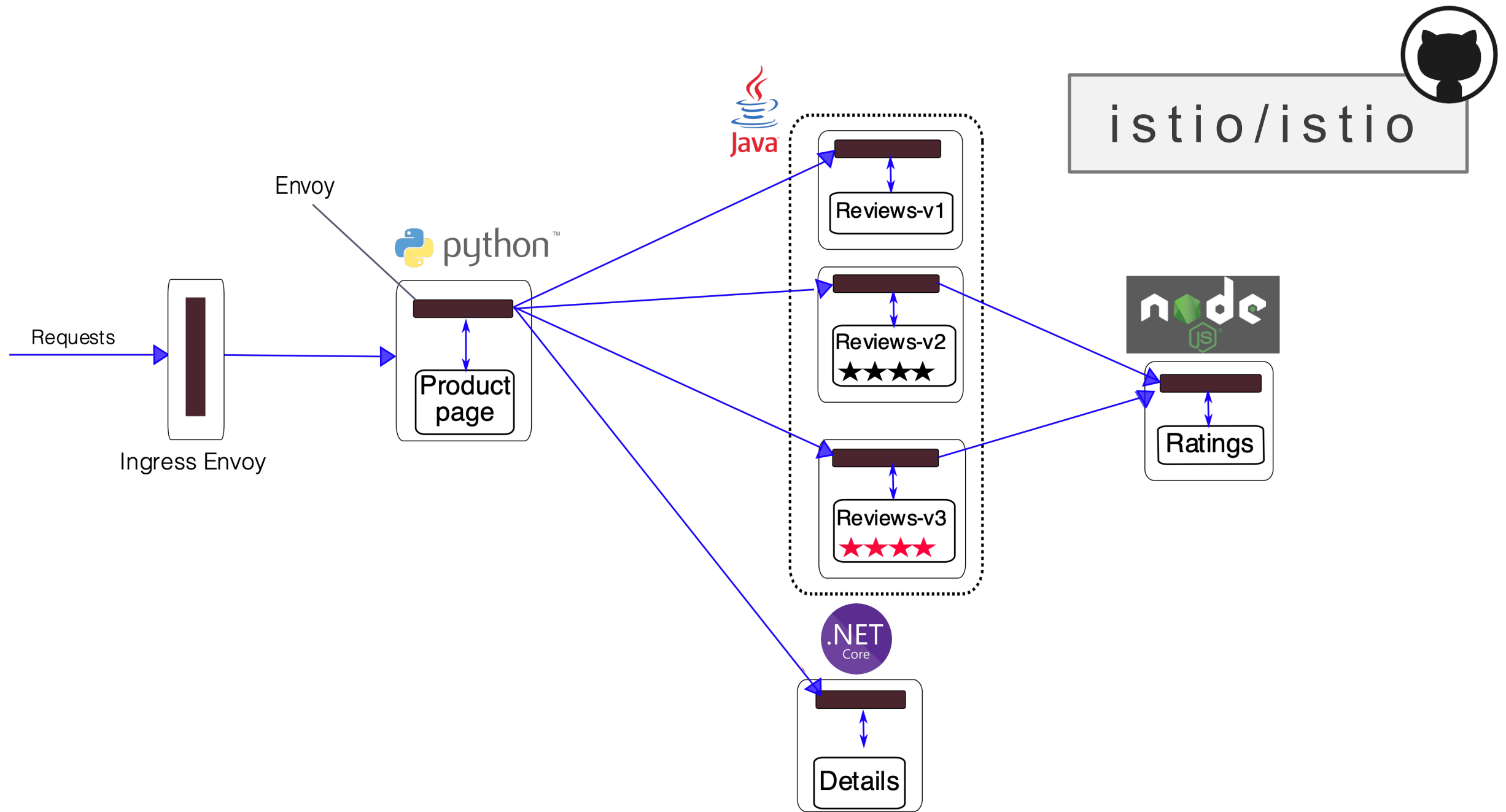
3.3M

moby/moby

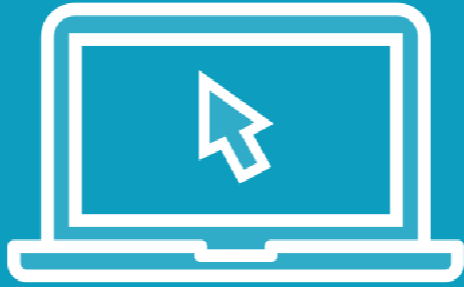
1.1M







Demo



Deploying an Istio app

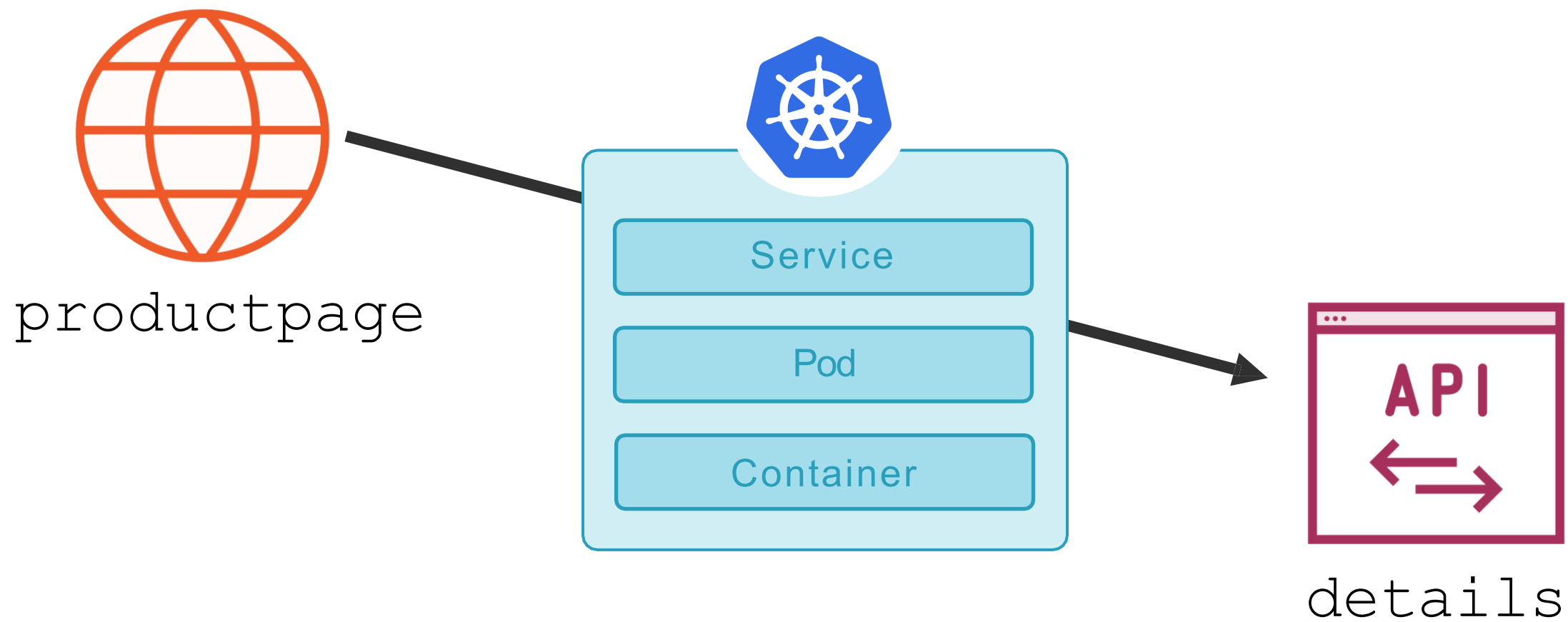
- Kubernetes services & pods
- Istio ingress gateway
- Testing the app

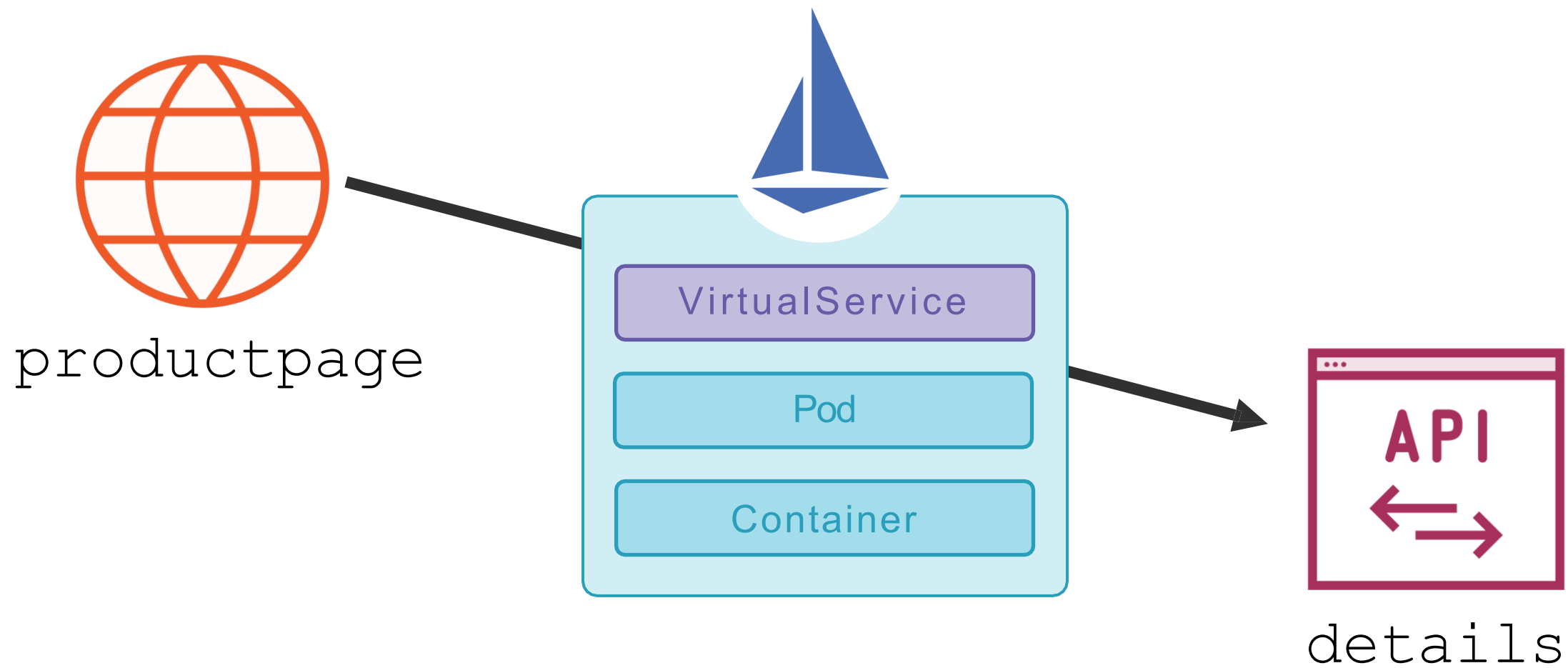

```
kubectl apply -f bookinfo-v1.yaml
```

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

Deploying BookInfo

Standard Kubernetes deployment - with Istio ingress





```
apiVersion: networking.istio.io/v1alpha3
```

```
kind: VirtualService
```

```
spec:
```

```
  hosts:
```

```
    - details
```

```
  http:
```

```
    - route:
```

```
      - destination:
```

```
        host: details
```

t Istio custom resource

t Defines routing rules

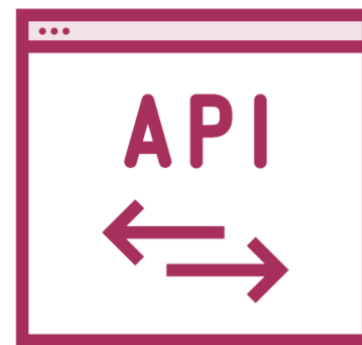
t The host requested by the client

t Routing for HTTP traffic

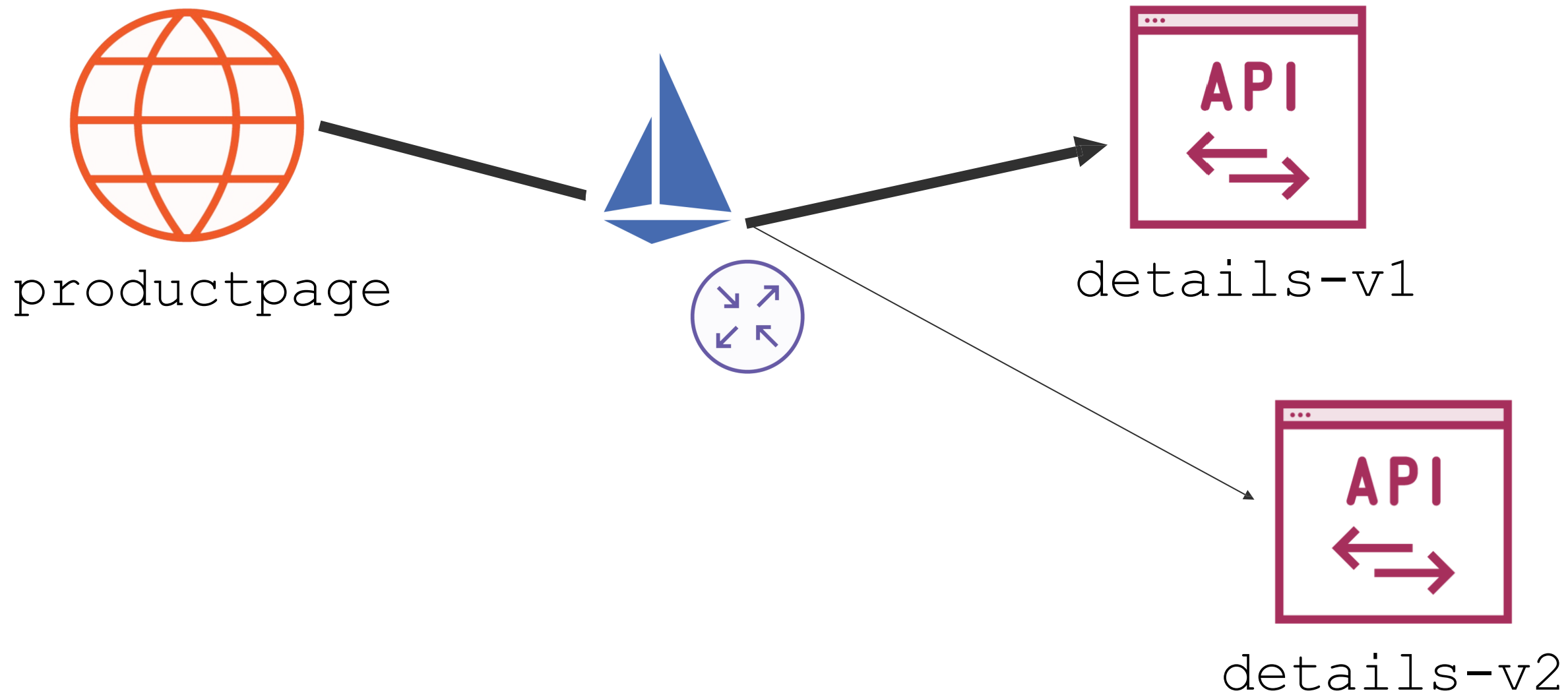
t The destination host routed by Istio



productpage

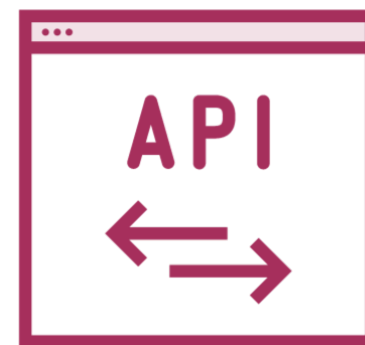


details





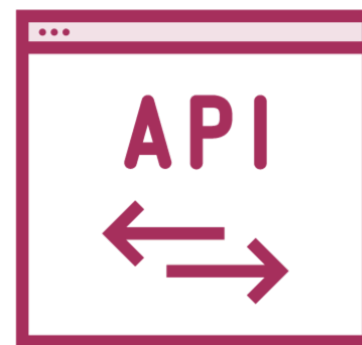
productpage



details

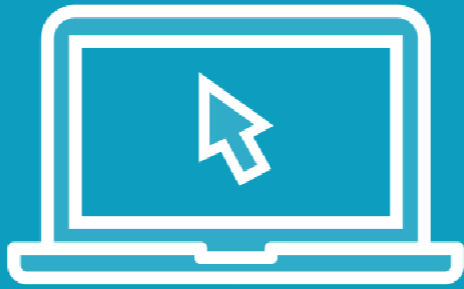


productpage



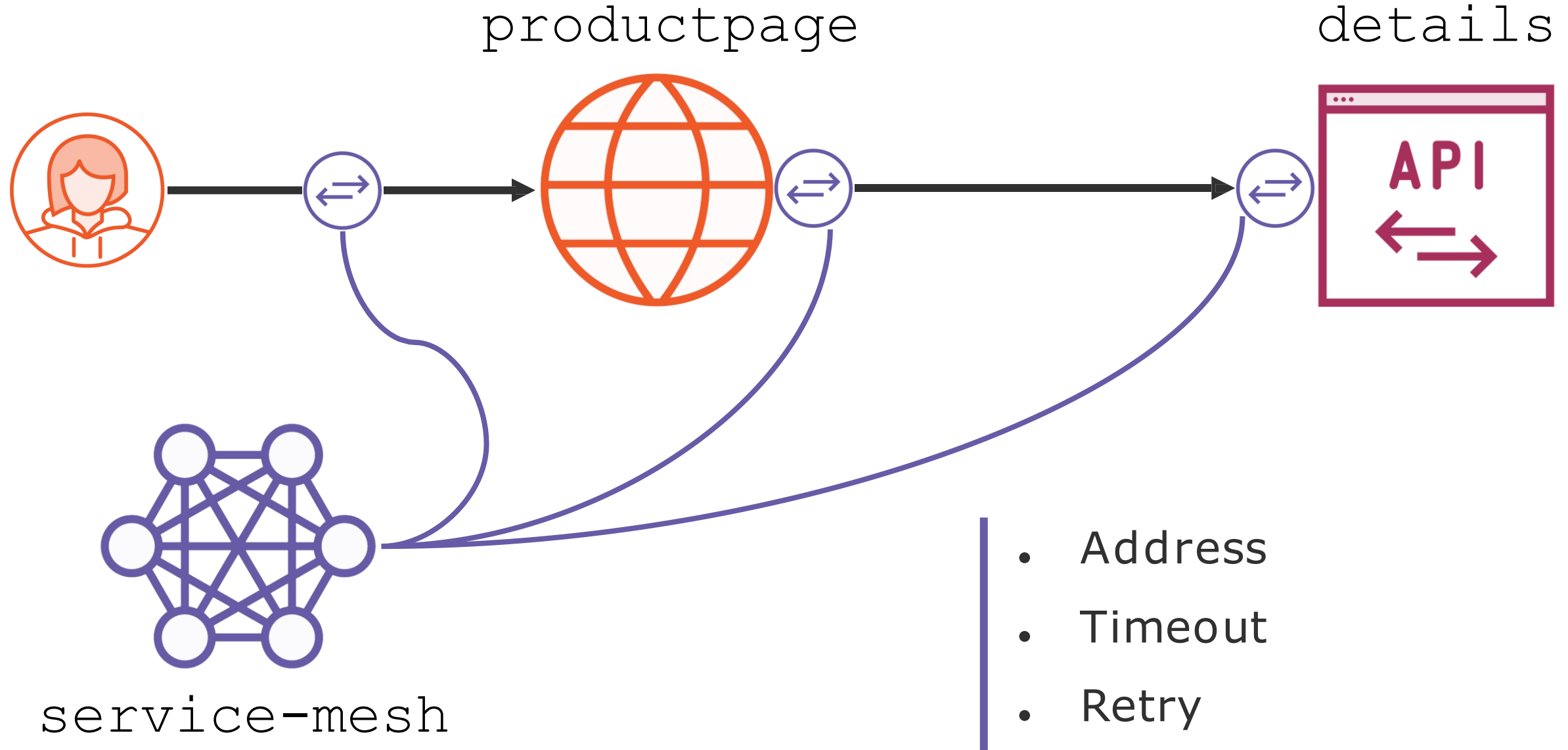
details

Demo



Managing traffic with Istio

- Fault-tolerance
- Client timeouts
- Automatic retries



Summary



Istio is a communication control plane

- Runs in Kubernetes
- Injects network proxies

Creating a service mesh

- Traffic management
- Security
- Observability

Up Next:
Managing Service Traffic
