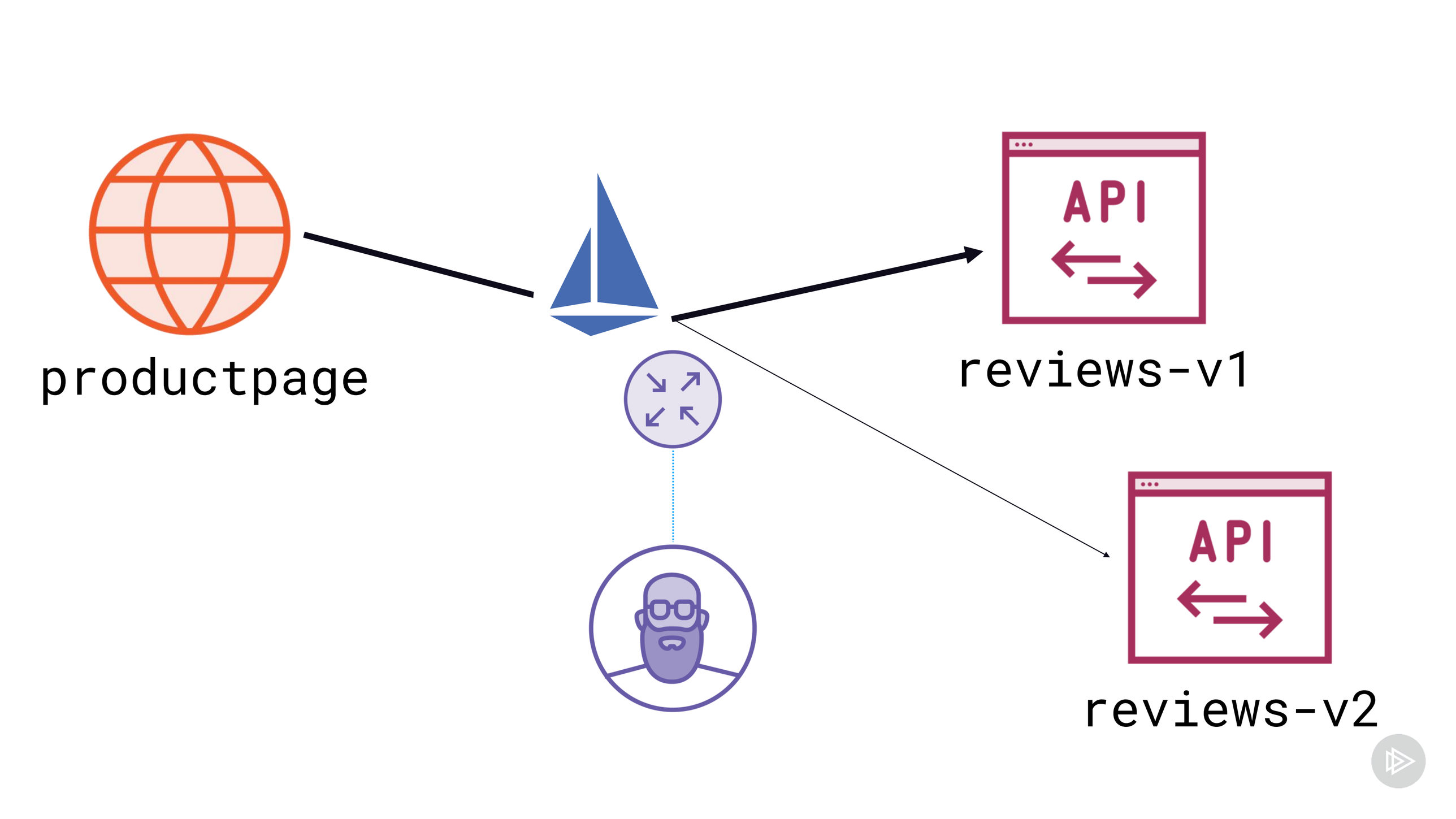
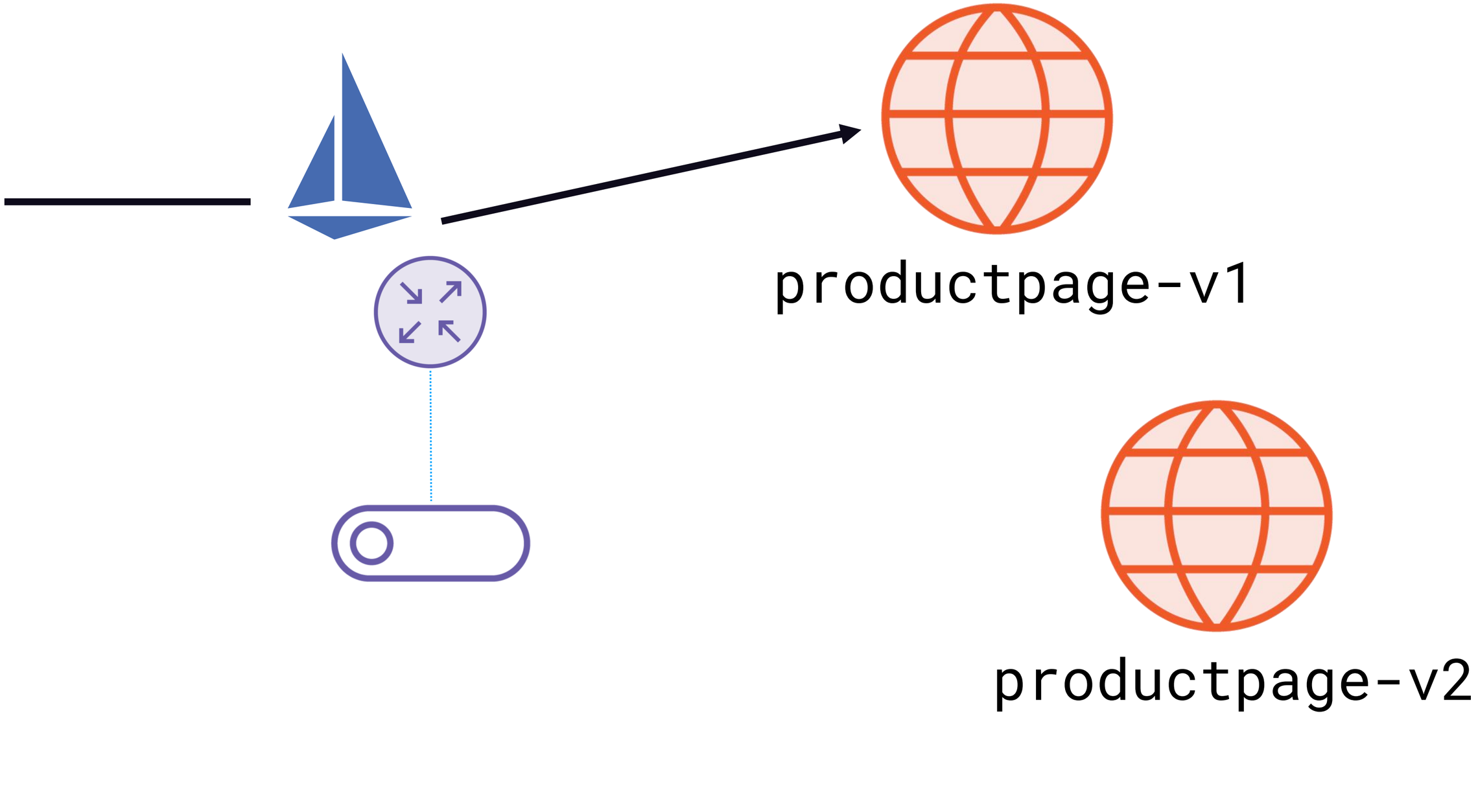
Managing Service Traffic

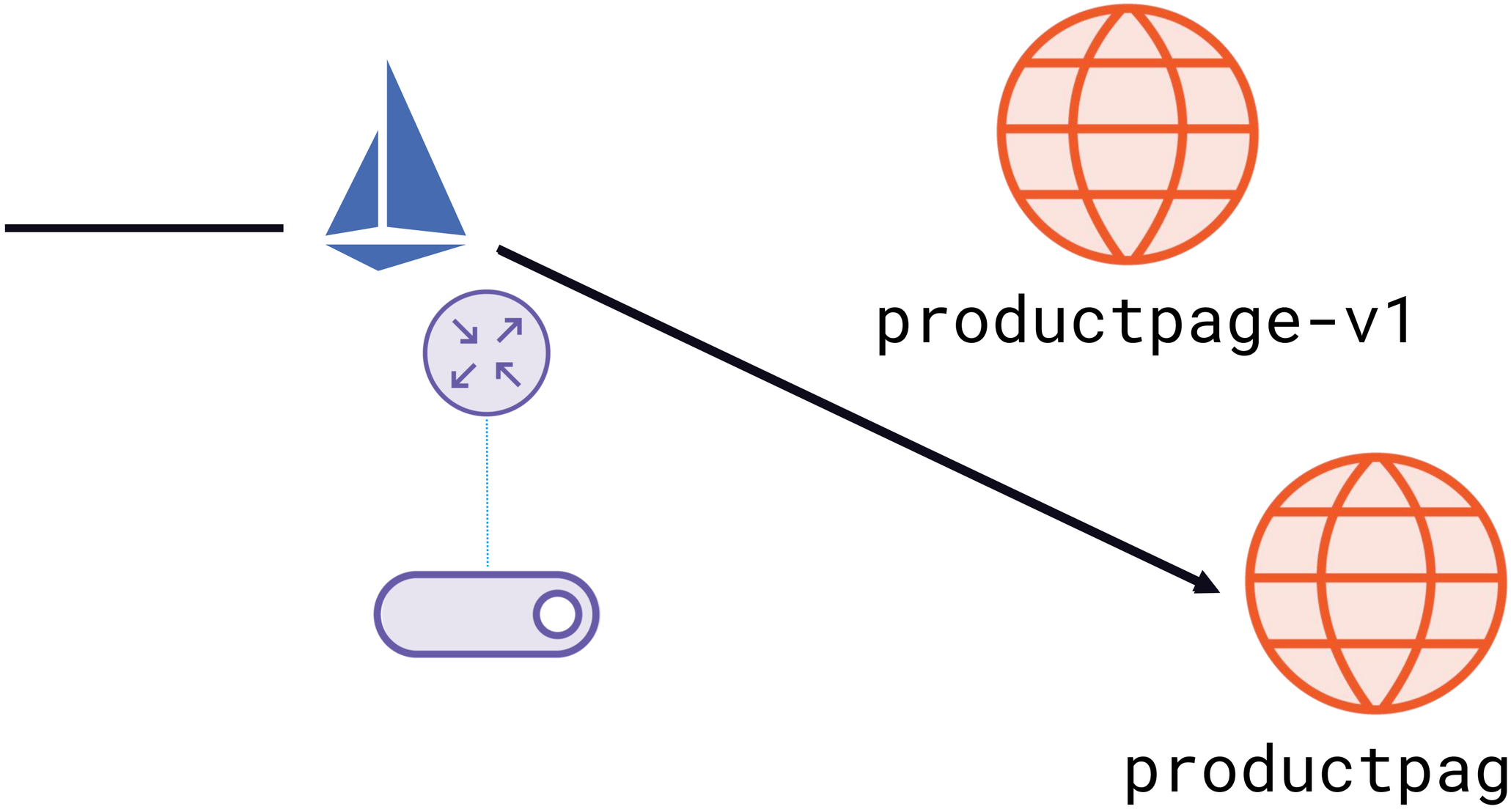
Elton Stoneman

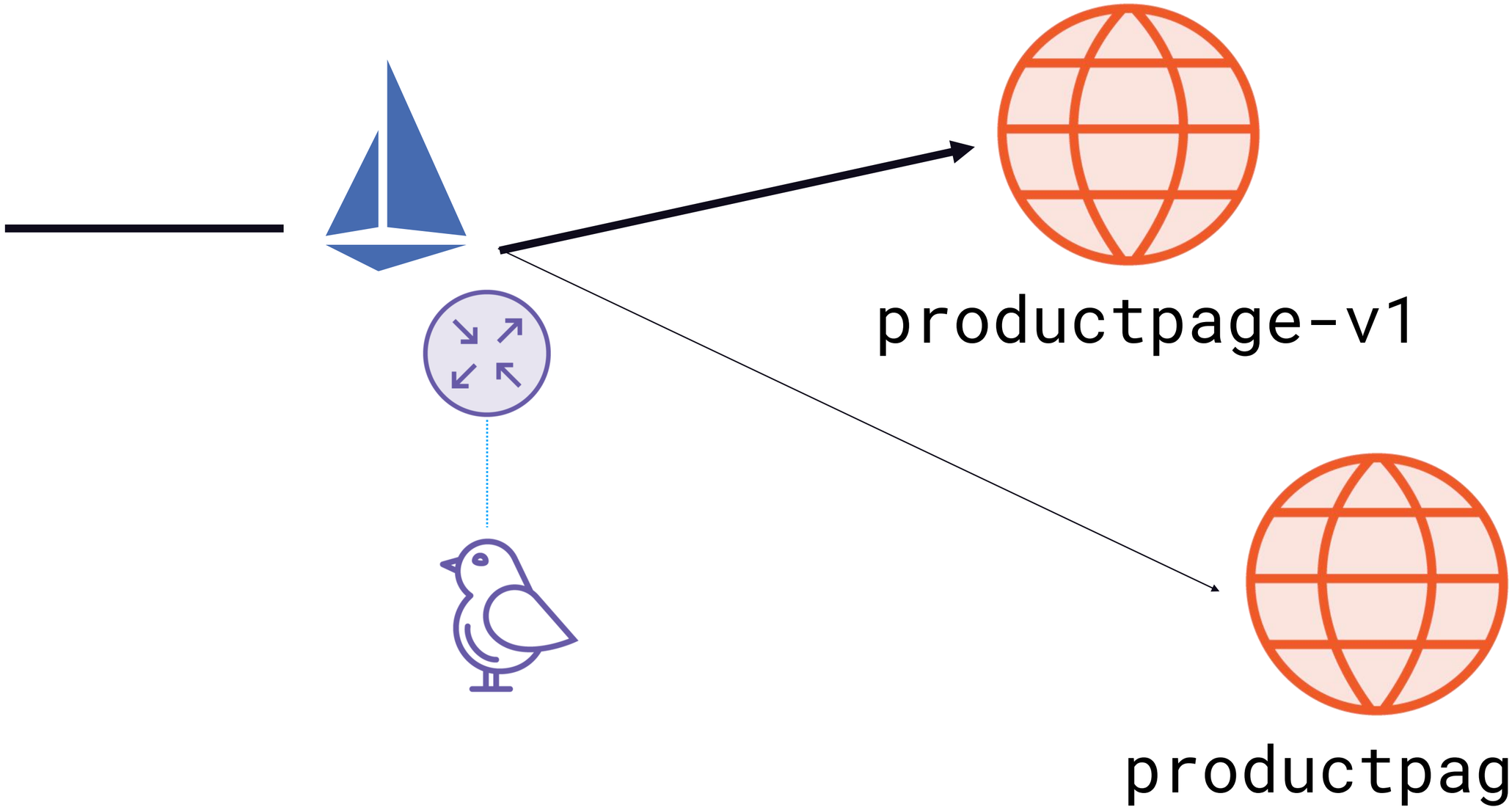
Freelance Consultant and Trainer

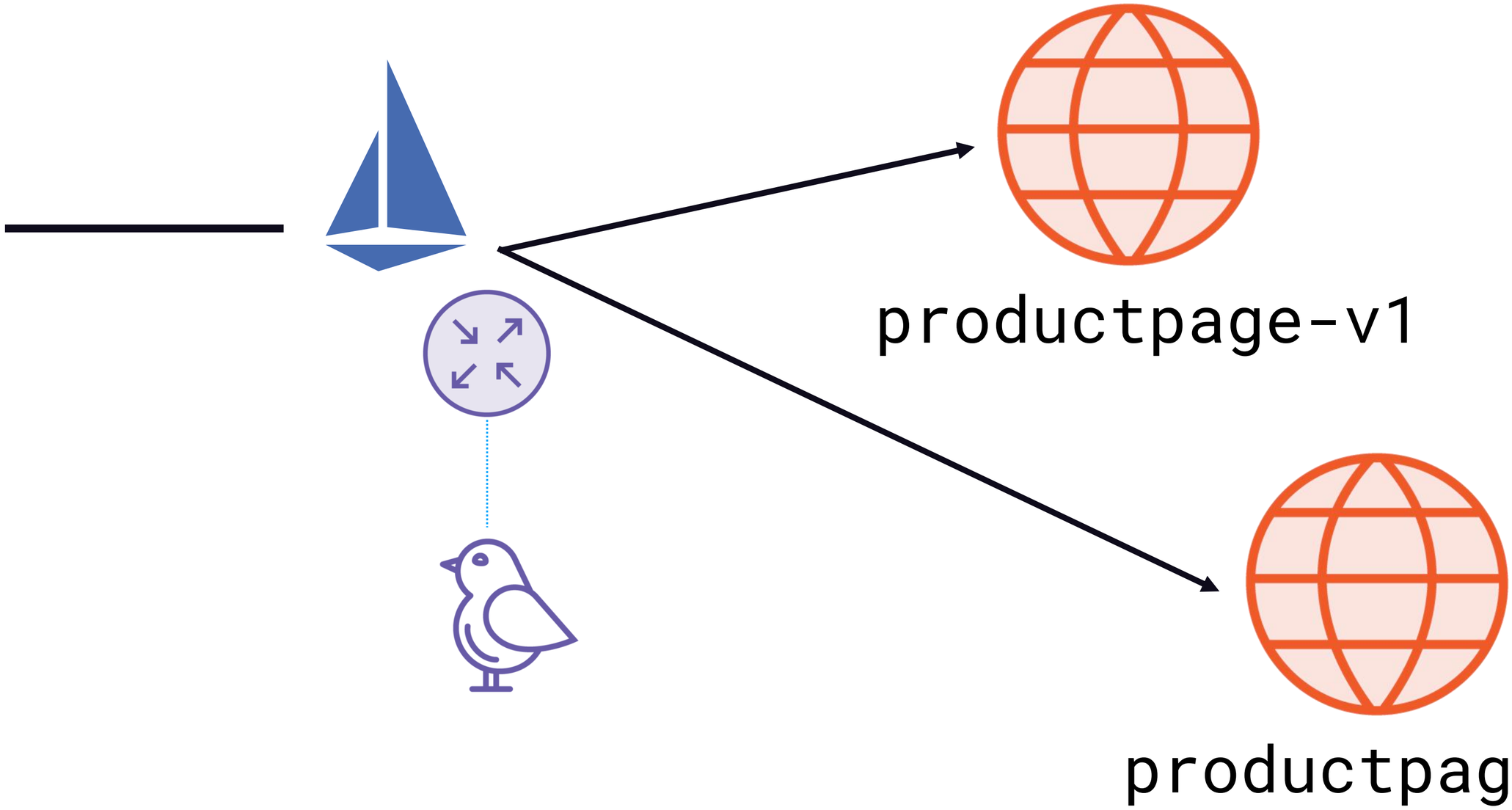
@EltonStoneman | blog.sixeyed.com

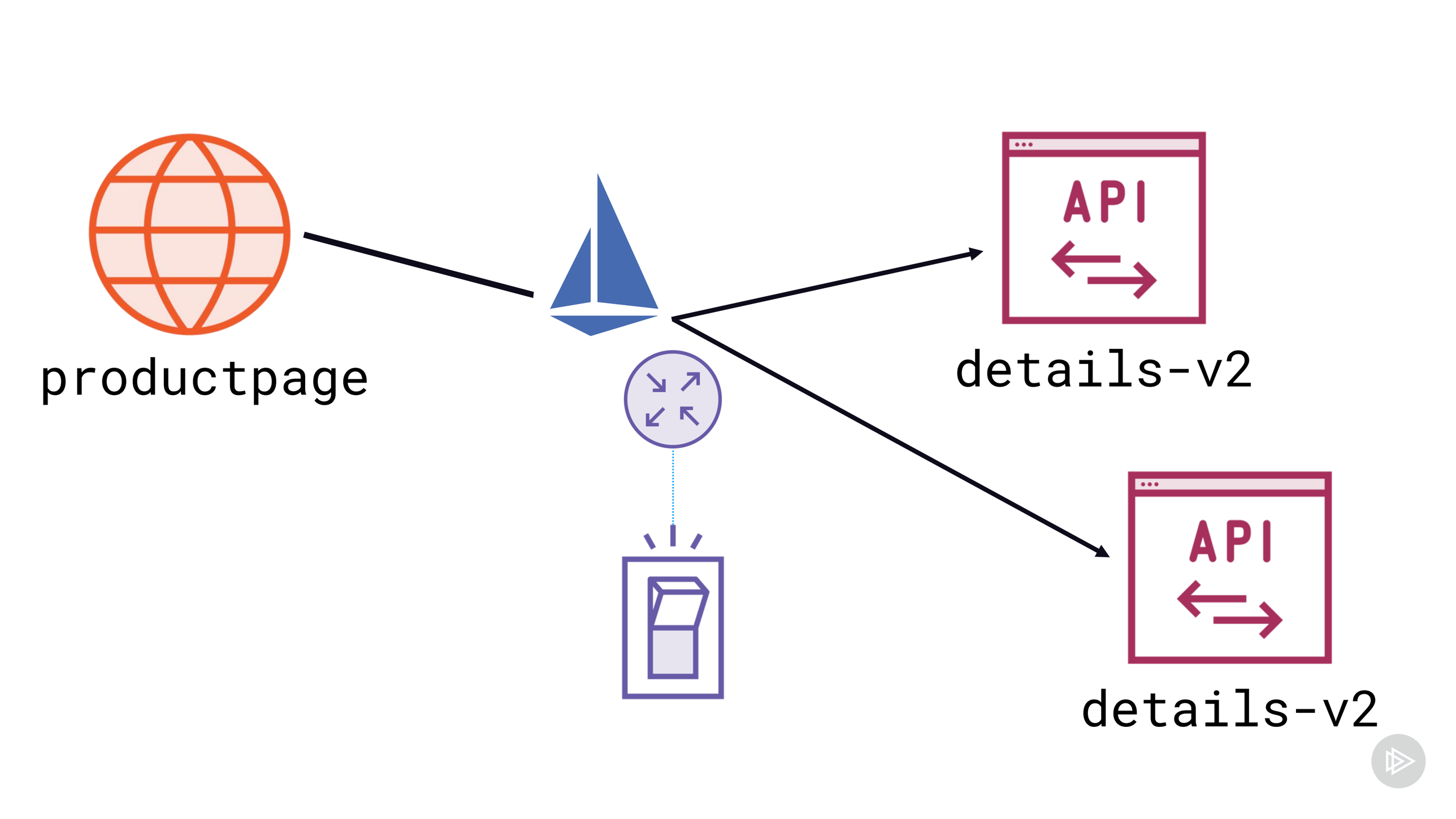

productpage
API
reviews-v1
API
reviews-v2

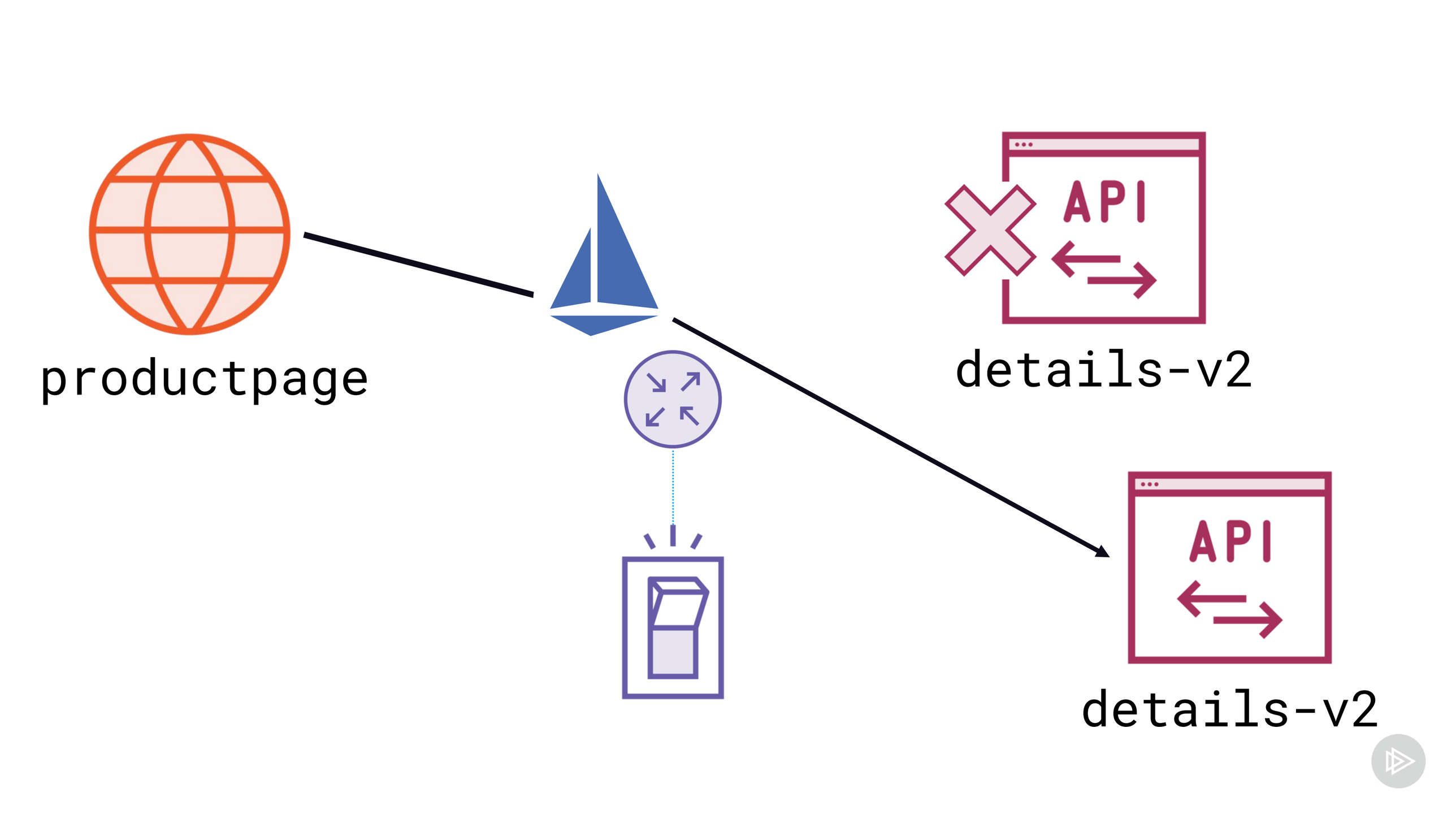

productpage-v1
productpage-v2

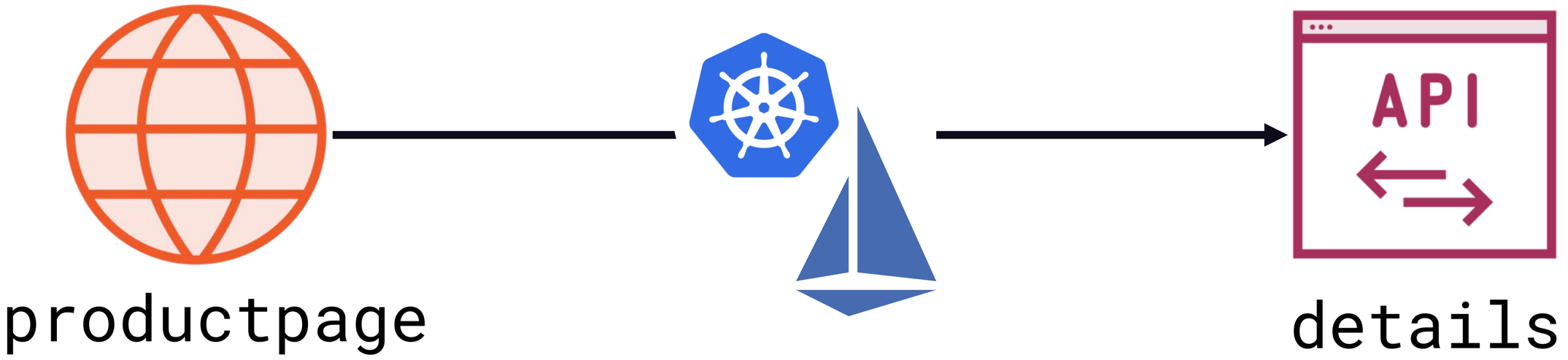

productpage-v1
productpage-v2

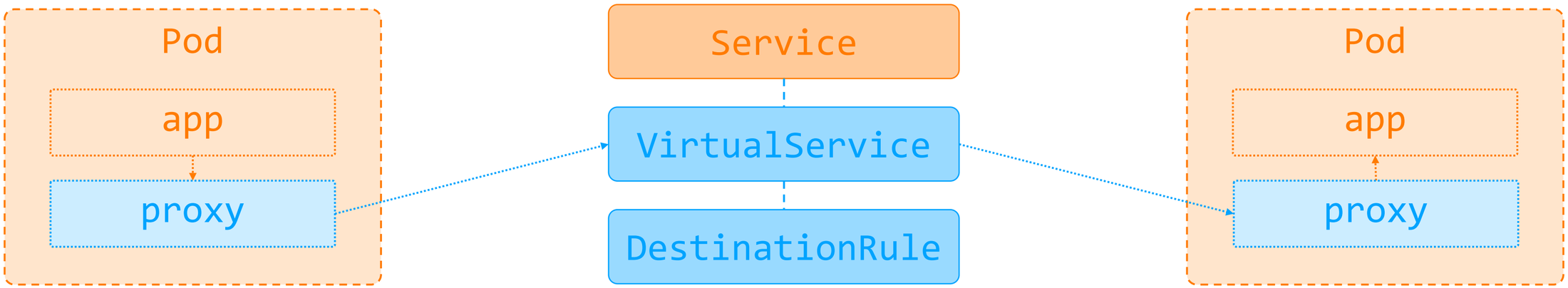

productpage-v1
productpage-v2


productpage-v1
productpage-v2


productpage
API
details-v2
API
details-v2


productpage
API
details-v2
API
details-v2


API
productpage
details


Pod
app
proxy
Service
VirtualService
DestinationRule
Pod
app
proxy

```
apiVersion: networking.istio.io/v1beta1
kind: VirtualService
spec:
  hosts:
  - reviews
  http:
  - route:
    - destination:
        host: reviews
        subset: v2
```

- The host requested by the client
- Routing for HTTP traffic
- The destination host routed by Istio
- The subset defined in the `DestinationRule`

```
apiVersion: networking.istio.io/v1beta1
kind: DestinationRule
spec:
  host: reviews
  subsets:
  - name: v1
    labels:
      version: v1
  - name: v2
    labels:
      version: v2
```

- The host requested by the client
- Available subsets for routing
- Subsets identify targets by labels

Service

name: reviews
selector:
app=reviews

Pod

labels: app=reviews
version=v1

Pod

labels: app=reviews
version=v1

Pod

NEW

labels: app=reviews
version=v2

Pod

NEW

labels: app=reviews
version=v2

VirtualService

host: reviews
destination: reviews

Service

name: reviews
selector:
app=reviews

Pod

labels: app=reviews
version=v1

Pod

labels: app=reviews
version=v1

Pod

NEW

labels: app=reviews
version=v2

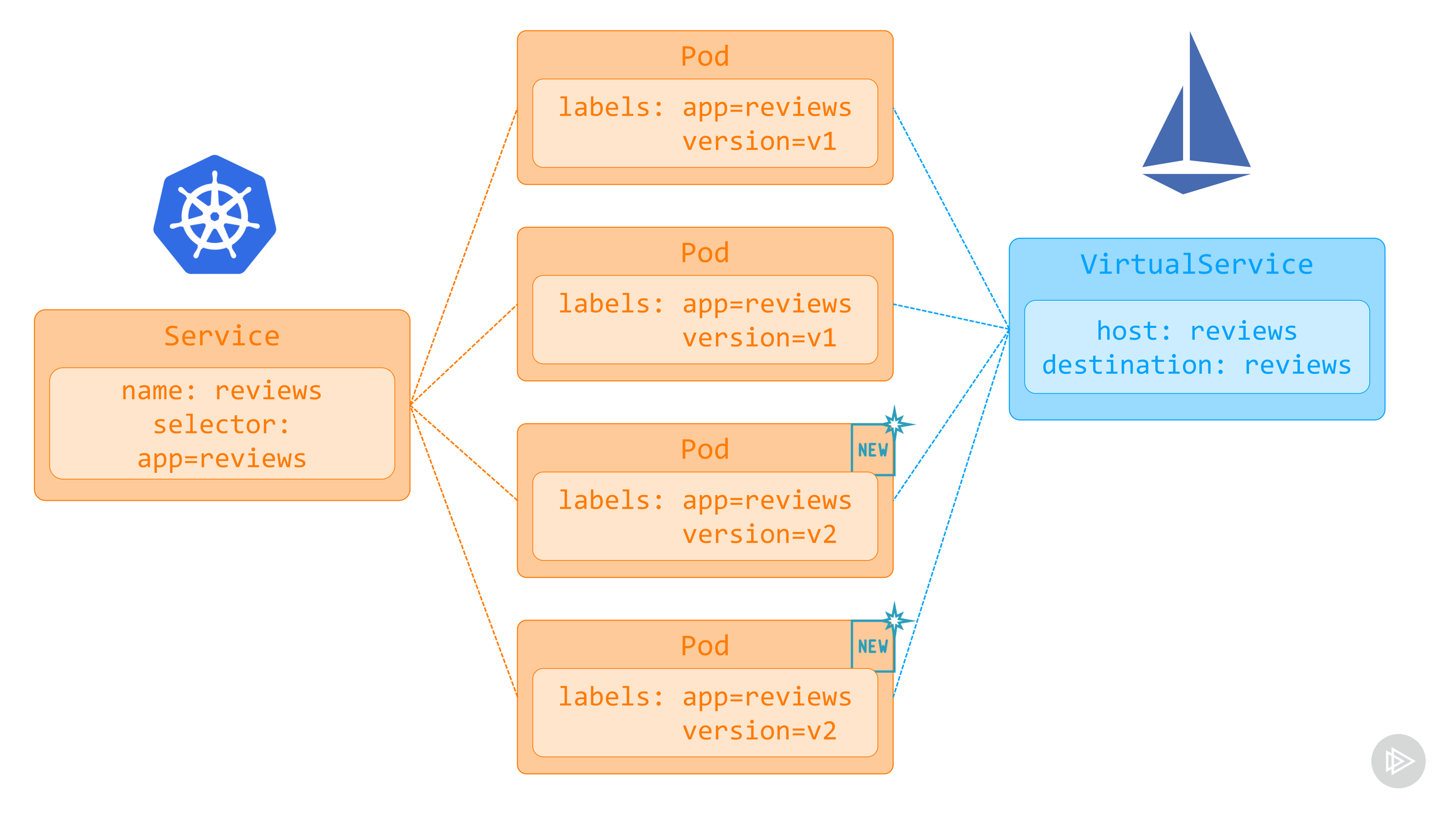
Pod

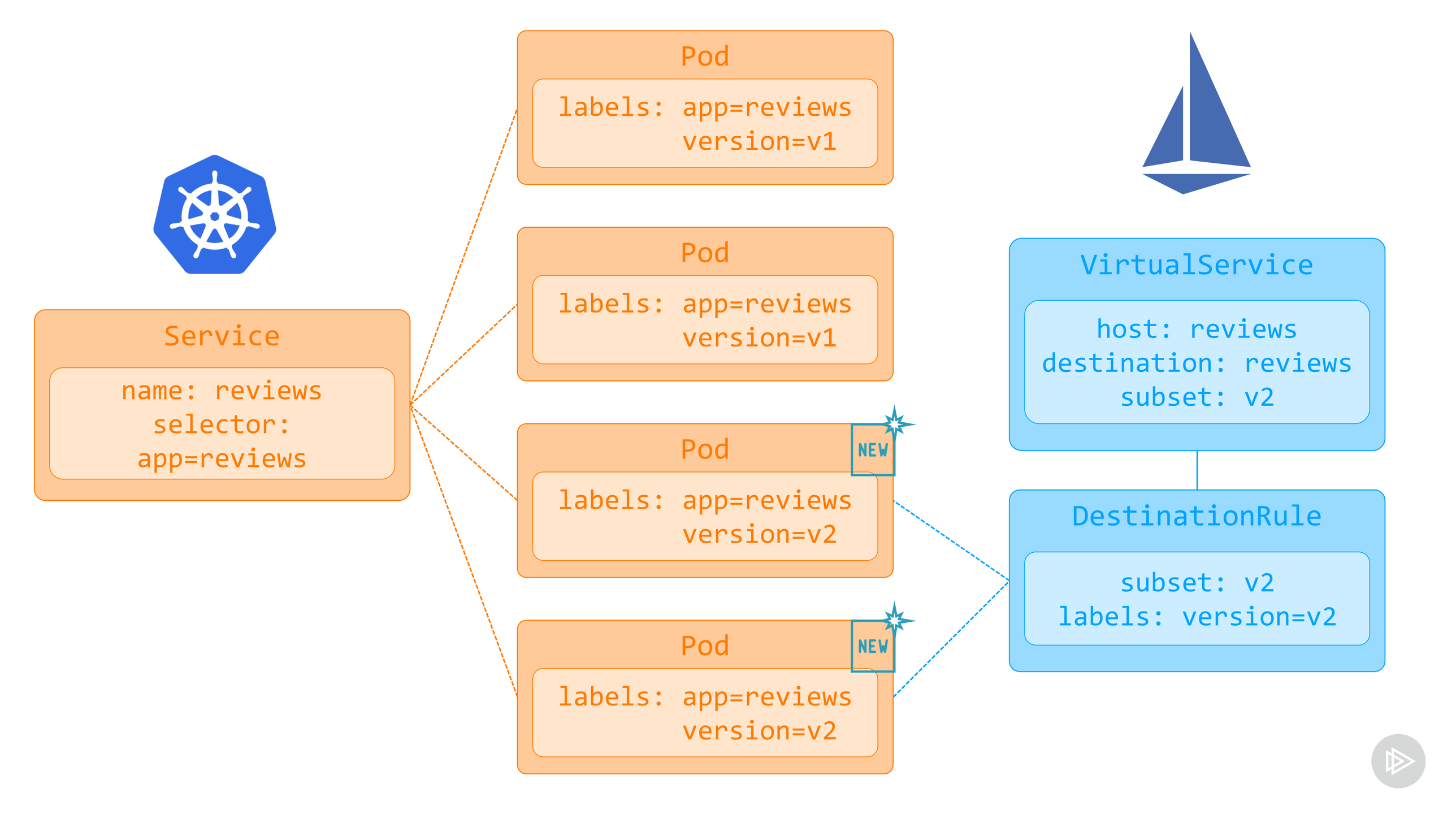
NEW

labels: app=reviews
version=v2

VirtualService

host: reviews
destination: reviews

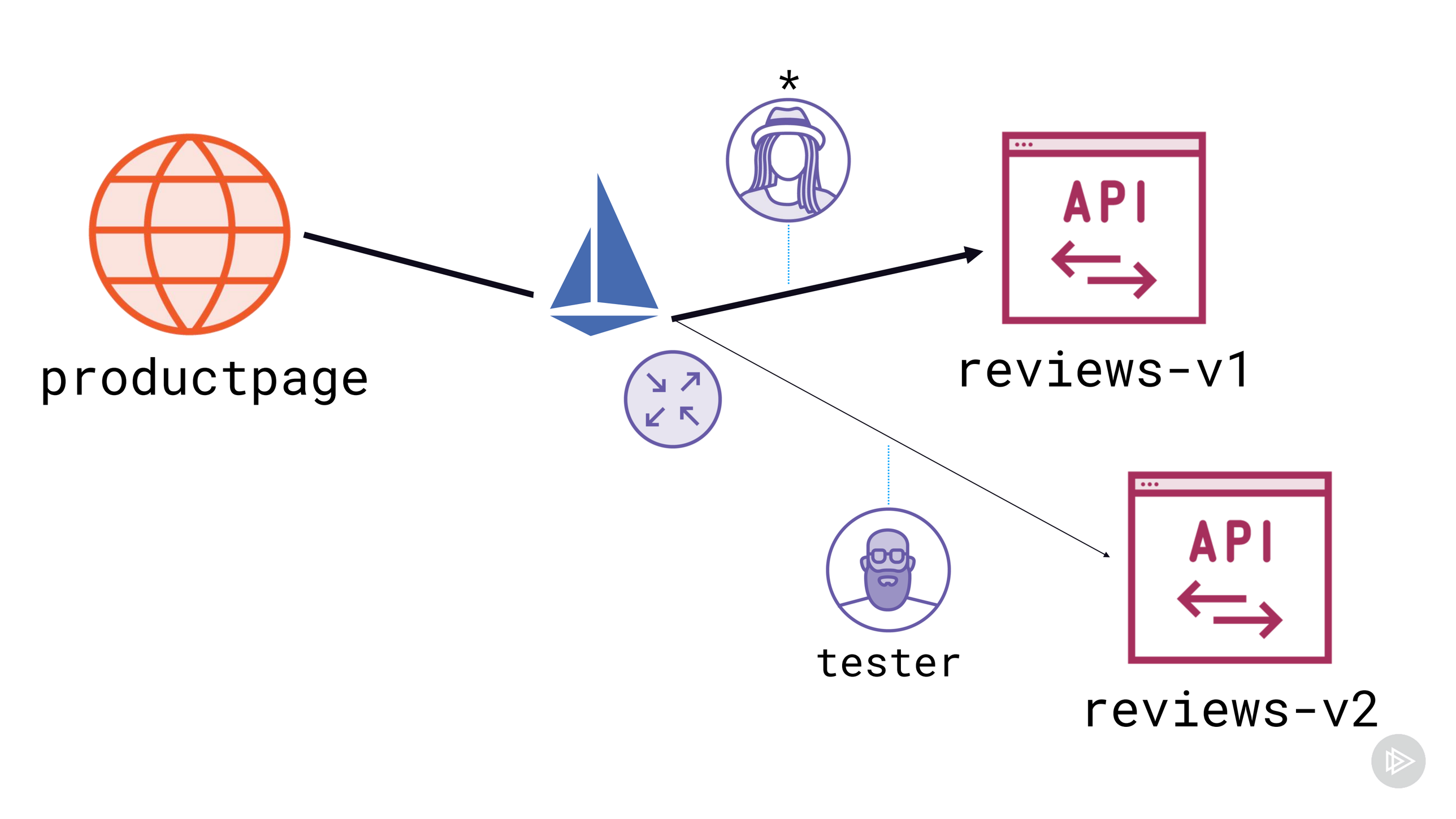

Service
name: reviews
selector:
app=reviews
Pod
labels: app=reviews
version=v1
Pod
labels: app=reviews
version=v1
Pod
NEW
labels: app=reviews
version=v2
Pod
NEW
labels: app=reviews
version=v2
VirtualService
host: reviews
destination: reviews


Service
name: reviews
selector:
app=reviews
Pod
labels: app=reviews
version=v1
Pod
labels: app=reviews
version=v1
Pod
NEW
labels: app=reviews
version=v2
Pod
NEW
labels: app=reviews
version=v2
VirtualService
host: reviews
destination: reviews
subset: v2
DestinationRule
subset: v2
labels: version=v2

Demo

Dark launch

- End-user routing
- Enable for test user
- Validate fault scenarios


*
API
productpage
reviews-v1
tester
API
reviews-v2

Custom Routing by User

DestinationRule.yaml

```yaml
host: reviews
subsets:
- name: v1
  labels:
    version: v1
- name: v2
  labels:
    version: v2
```

VirtualService.yaml

```yaml
http:
- match:
  - headers:
      end-user:
        exact: tester
  route:
  - destination:
      host: reviews
      subset: v2
```

Custom Routing by User

DestinationRule.yaml

```yaml
host: reviews
subsets:
- name: v1
  labels:
    version: v1
- name: v2
  labels:
    version: v2
```

VirtualService.yaml

```yaml
http:
- match:
  - headers:
      end-user:
        exact: tester
  route:
  - destination:
      host: reviews
      subset: v2
```

Custom Routing by User

DestinationRule.yaml

```
host: reviews
subsets:
- name: v1
  labels:
    version: v1
- name: v2
  labels:
    version: v2
```

VirtualService.yaml

```
http:
- match:
  - headers:
      end-user:
        exact: tester
  route:
  - destination:
      host: reviews
      subset: v2
```

Delay Injection

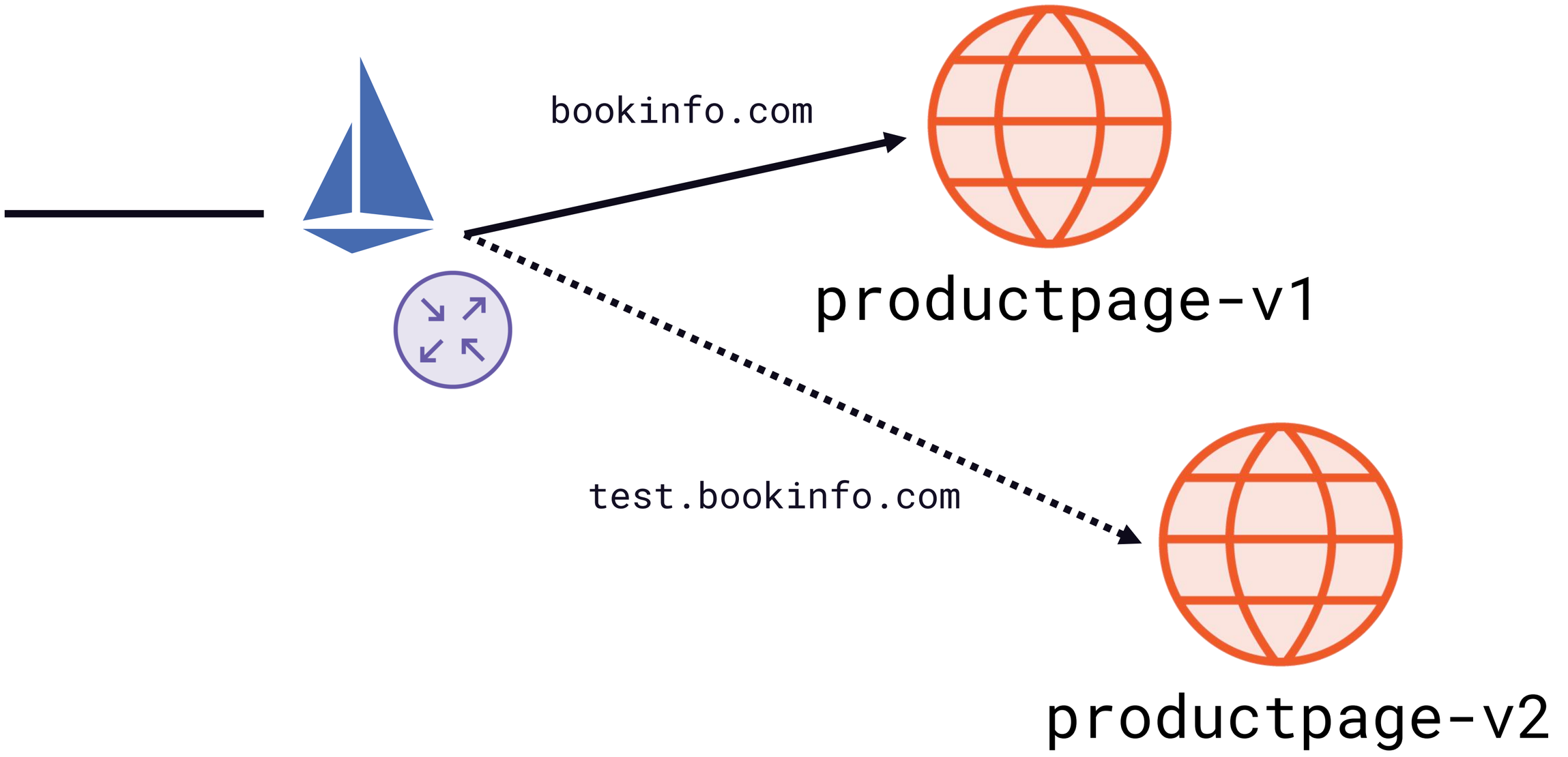
VirtualService.yaml

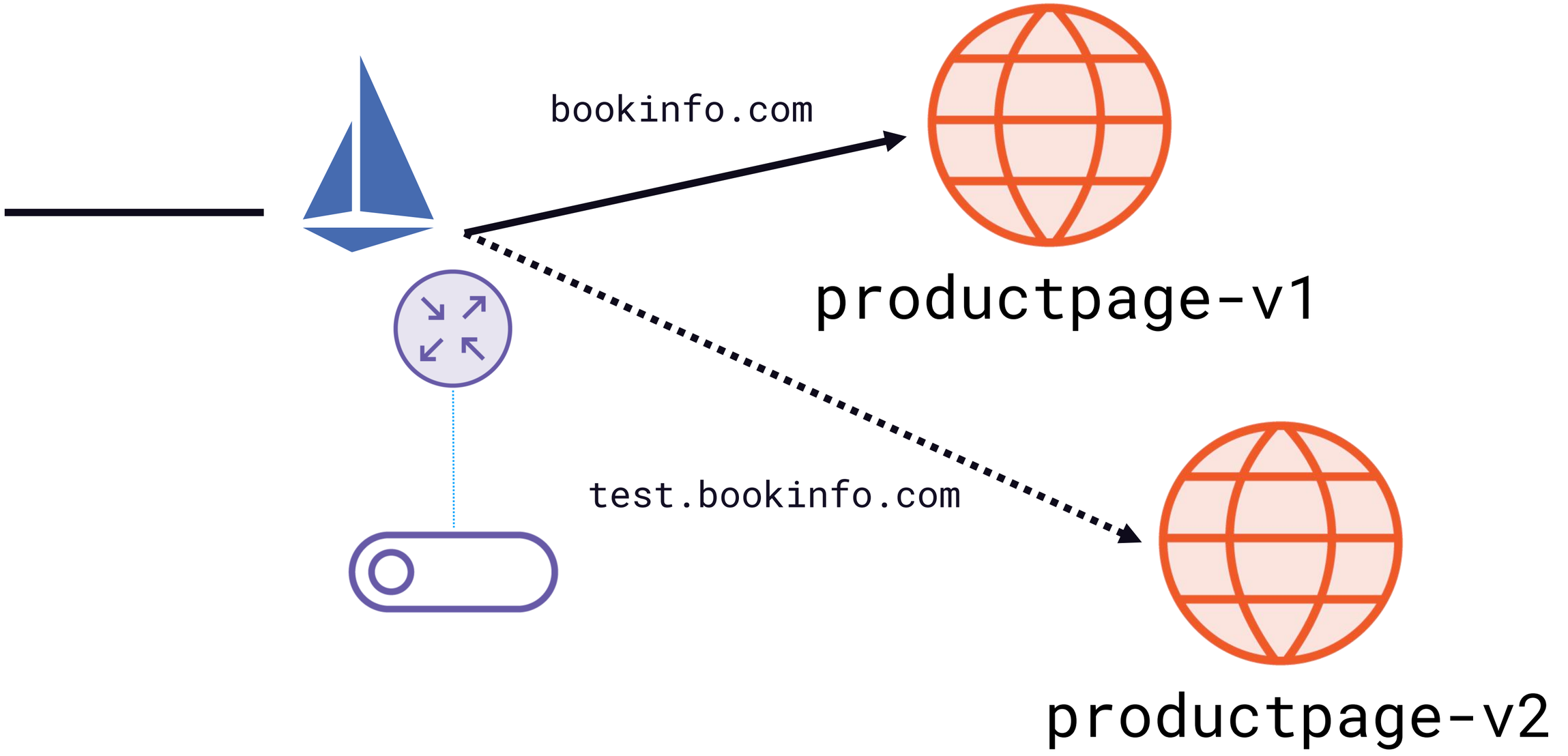
```yaml
http:
- match:
  - headers:
      end-user:
        exact: tester
  fault:
    delay:
      percentage:
        value: 100
      fixedDelay: 2.5s
  route:
  - destination:
      host: reviews
      subset: v2
- route:
  - destination:
      host: reviews
      subset: v1
```

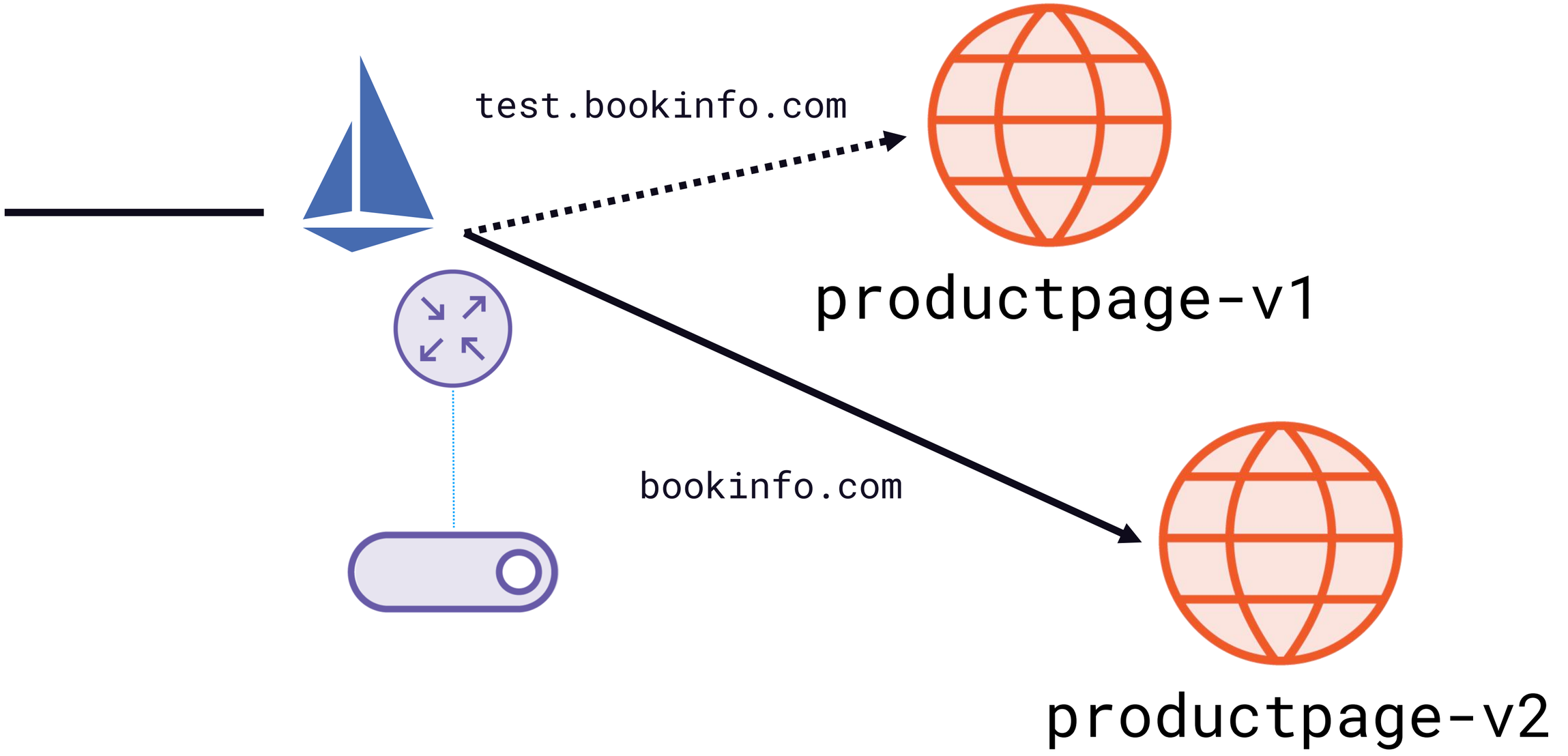
Fault Injection

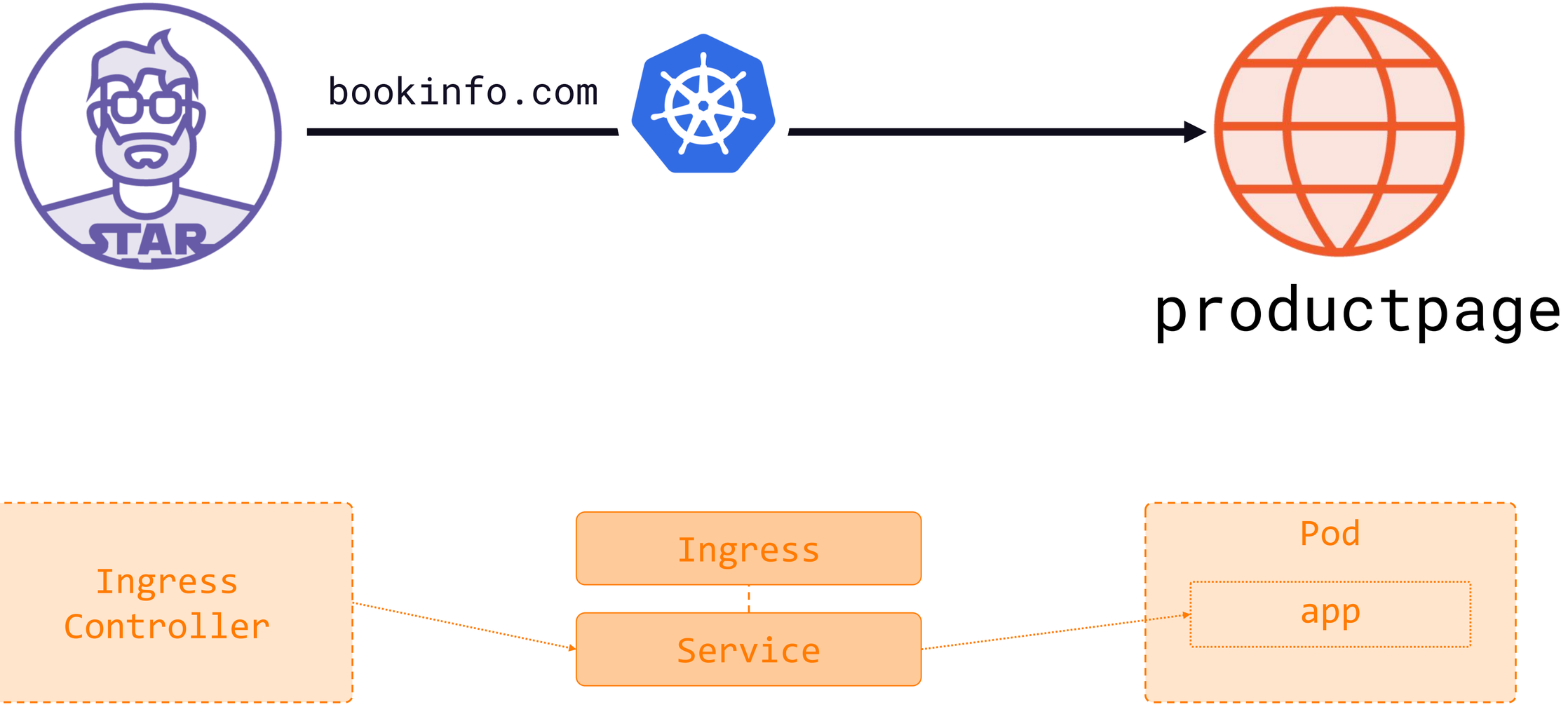
VirtualService.yaml

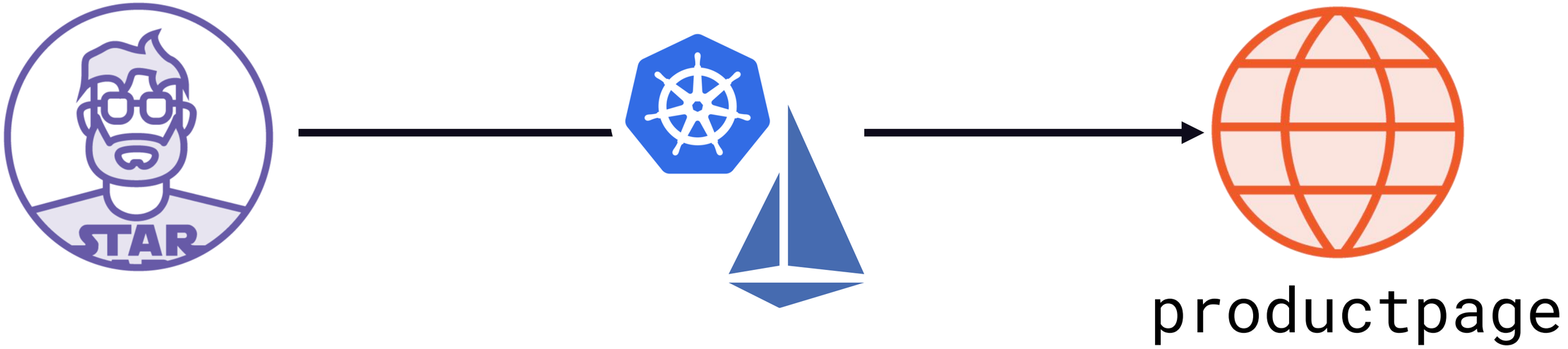
```yaml
http:
- match:
  - headers:
      end-user:
        exact: tester
  fault:
    abort:
      percentage:
        value: 50
      httpStatus: 503
  route:
  - destination:
      host: reviews
      subset: v2
- route:
  - destination:
      host: reviews
      subset: v1
```

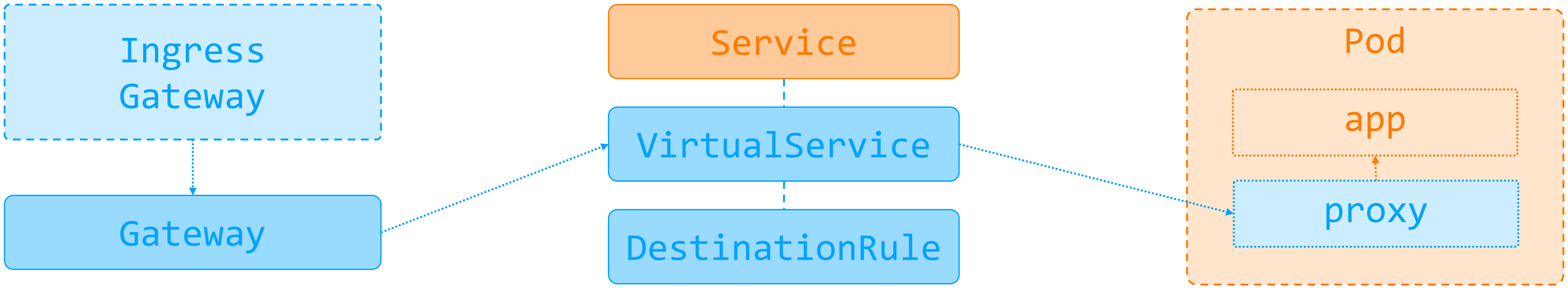

bookinfo.com
productpage-v1
test.bookinfo.com
productpage-v2

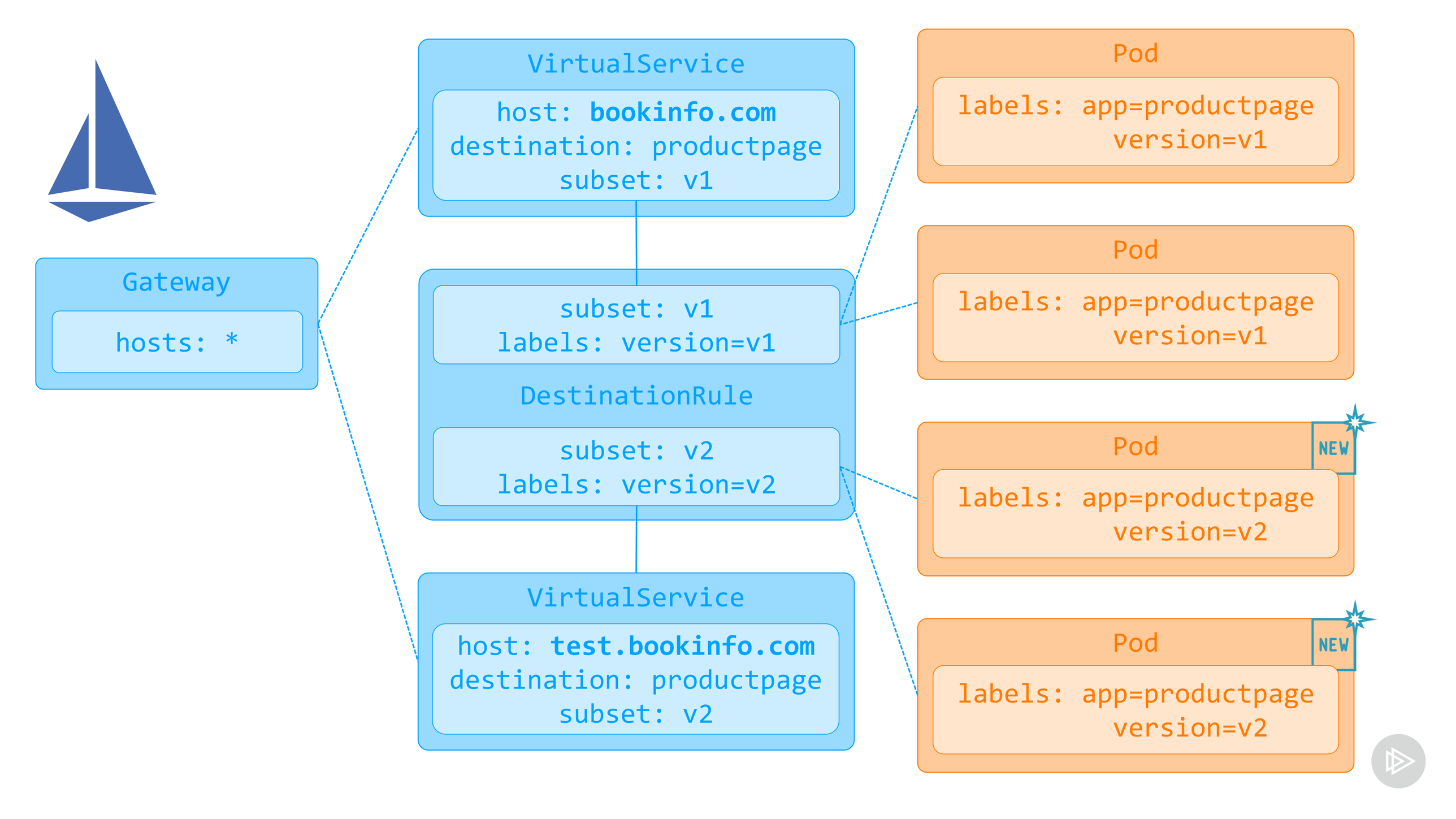

bookinfo.com
productpage-v1
test.bookinfo.com
productpage-v2


test.bookinfo.com
productpage-v1
bookinfo.com
productpage-v2


bookinfo.com
productpage
Ingress
Controller
Ingress
Service
Pod
app


productpage


Ingress
Gateway
Gateway
Service
VirtualService
DestinationRule
Pod
app
proxy


Gateway
hosts: *
VirtualService
host: bookinfo.com
destination: productpage
subset: v1
subset: v1
labels: version=v1
DestinationRule
subset: v2
labels: version=v2
VirtualService
host: test.bookinfo.com
destination: productpage
subset: v2
Pod
labels: app=productpage
version=v1
Pod
labels: app=productpage
version=v1
Pod
NEW
labels: app=productpage
version=v2
Pod
NEW
labels: app=productpage
version=v2

Gateway

hosts: *

http://bookinfo.com

VirtualService

host: **bookinfo.com**
destination: productpage
subset: v1

subset: v1
labels: version=v1

DestinationRule

subset: v2
labels: version=v2

VirtualService

host: **test.bookinfo.com**
destination: productpage
subset: v2

Pod

labels: app=productpage
version=v1

Pod

labels: app=productpage
version=v1

Pod

NEW

labels: app=productpage
version=v2

Pod

NEW

labels: app=productpage
version=v2

Gateway

hosts: *

http://test.bookinfo.com

VirtualService

host: **bookinfo.com**
destination: productpage
subset: v1

subset: v1
labels: version=v1

DestinationRule

subset: v2
labels: version=v2

VirtualService

host: **test.bookinfo.com**
destination: productpage
subset: v2

Pod

labels: app=productpage
version=v1

Pod

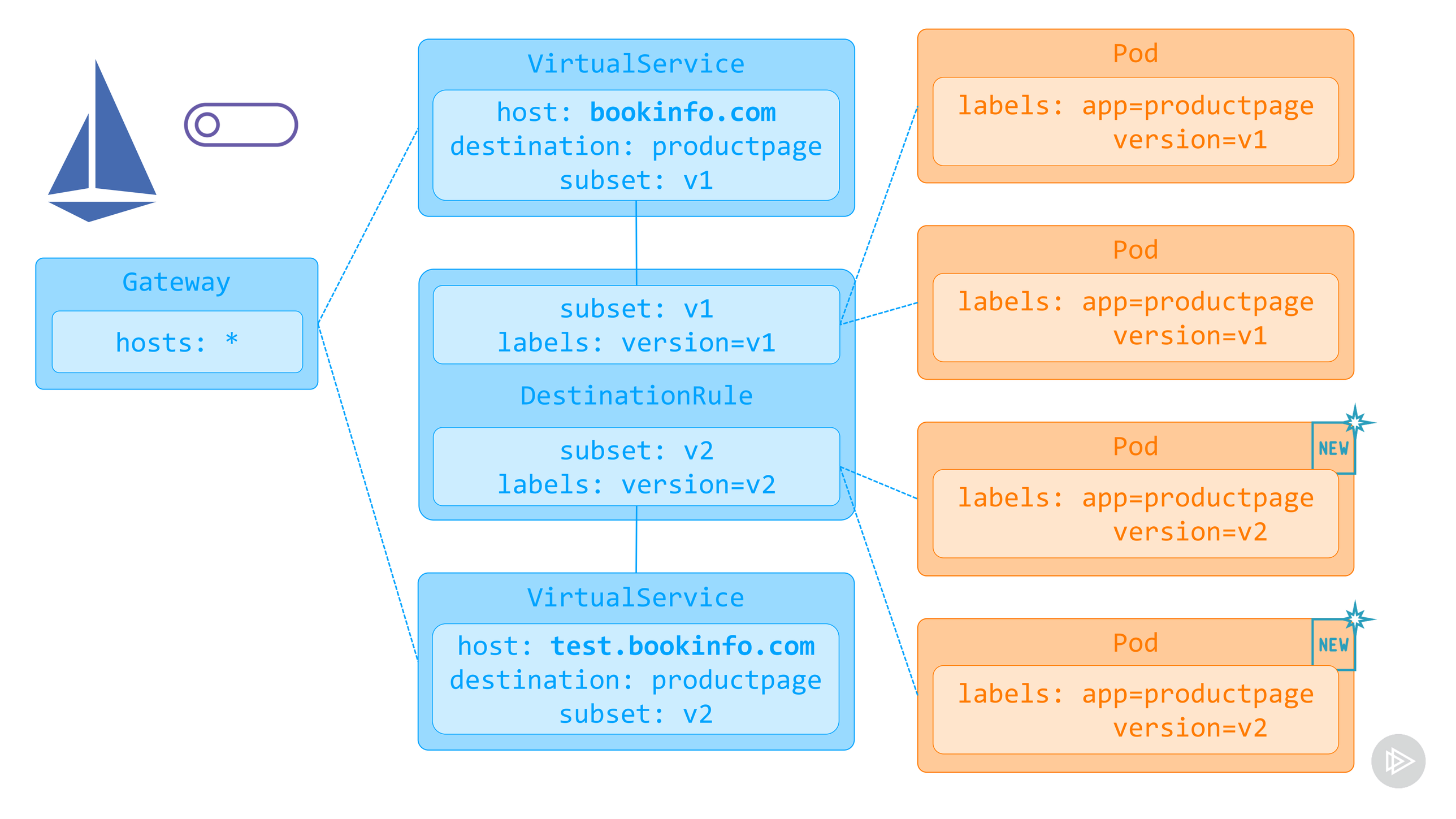
labels: app=productpage
version=v1

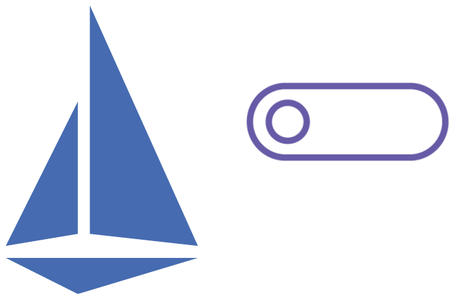
Pod NEW

labels: app=productpage
version=v2

Pod NEW

labels: app=productpage
version=v2


Gateway
hosts: *
VirtualService
host: bookinfo.com
destination: productpage
subset: v1
subset: v1
labels: version=v1
DestinationRule
subset: v2
labels: version=v2
VirtualService
host: test.bookinfo.com
destination: productpage
subset: v2
Pod
labels: app=productpage
version=v1
Pod
labels: app=productpage
version=v1
Pod
NEW
labels: app=productpage
version=v2
Pod
NEW
labels: app=productpage
version=v2



Gateway

hosts: *

VirtualService

host: **bookinfo.com**
destination: productpage
subset: **v1**

subset: v1
labels: version=v1

DestinationRule

subset: v2
labels: version=v2

VirtualService

host: **test.bookinfo.com**
destination: productpage
subset: **v2**

Pod

labels: app=productpage
version=v1

Pod

labels: app=productpage
version=v1

Pod

NEW

labels: app=productpage
version=v2

Pod

NEW

labels: app=productpage
version=v2

Gateway

hosts: *

VirtualService

host: **bookinfo.com**
destination: productpage
subset: **v2**

subset: v1
labels: version=v1

DestinationRule

subset: v2
labels: version=v2

VirtualService

host: **test.bookinfo.com**
destination: productpage
subset: **v1**

Pod

labels: app=productpage
version=v1

Pod

labels: app=productpage
version=v1

Pod

NEW

labels: app=productpage
version=v2

Pod

NEW

labels: app=productpage
version=v2

Gateway

hosts: *

http://bookinfo.com

VirtualService

host: **bookinfo.com**
destination: productpage
subset: **v2**

subset: v1
labels: version=v1

DestinationRule

subset: v2
labels: version=v2

VirtualService

host: **test.bookinfo.com**
destination: productpage
subset: **v1**

Pod

labels: app=productpage
version=v1

Pod

labels: app=productpage
version=v1

Pod

NEW

labels: app=productpage
version=v2

Pod

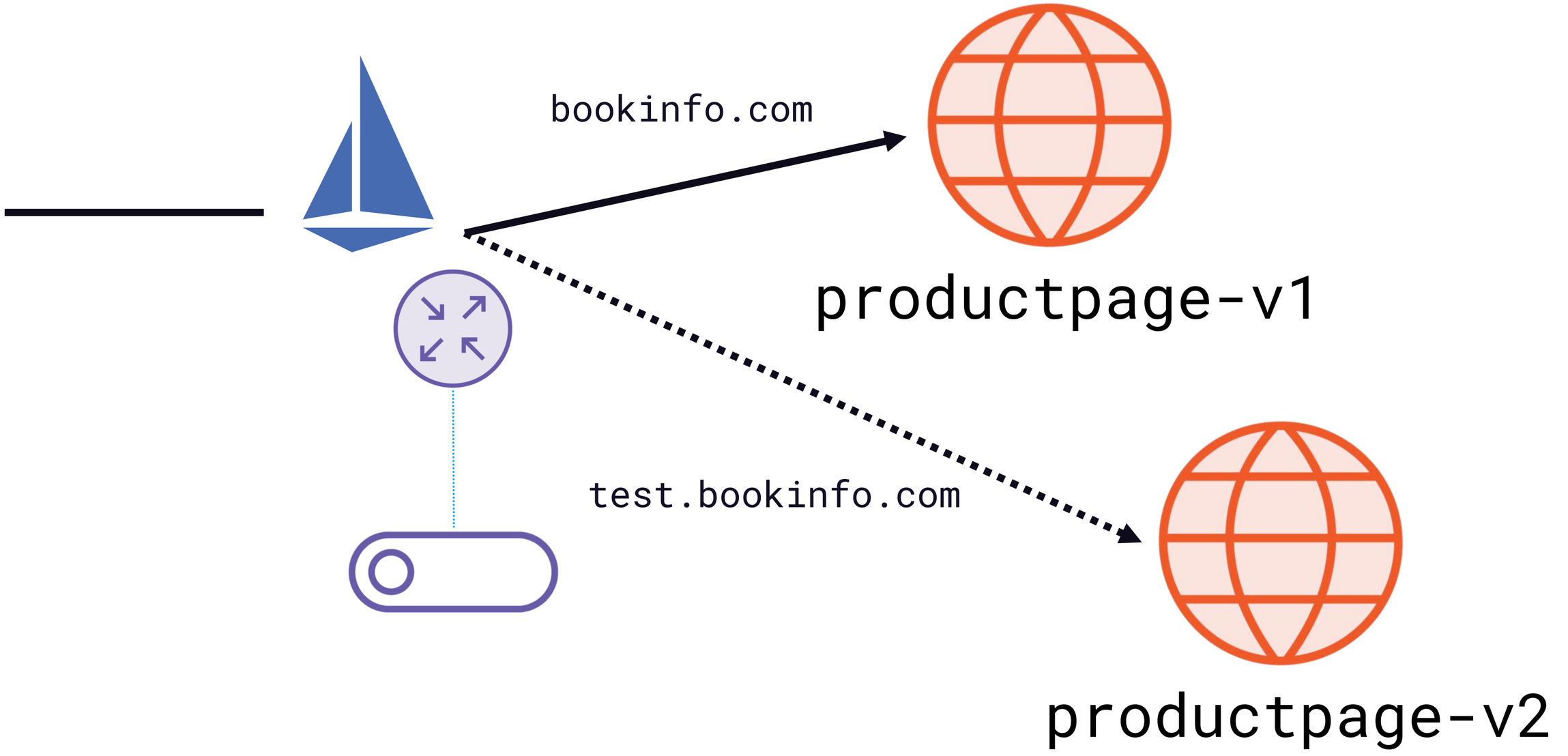
NEW

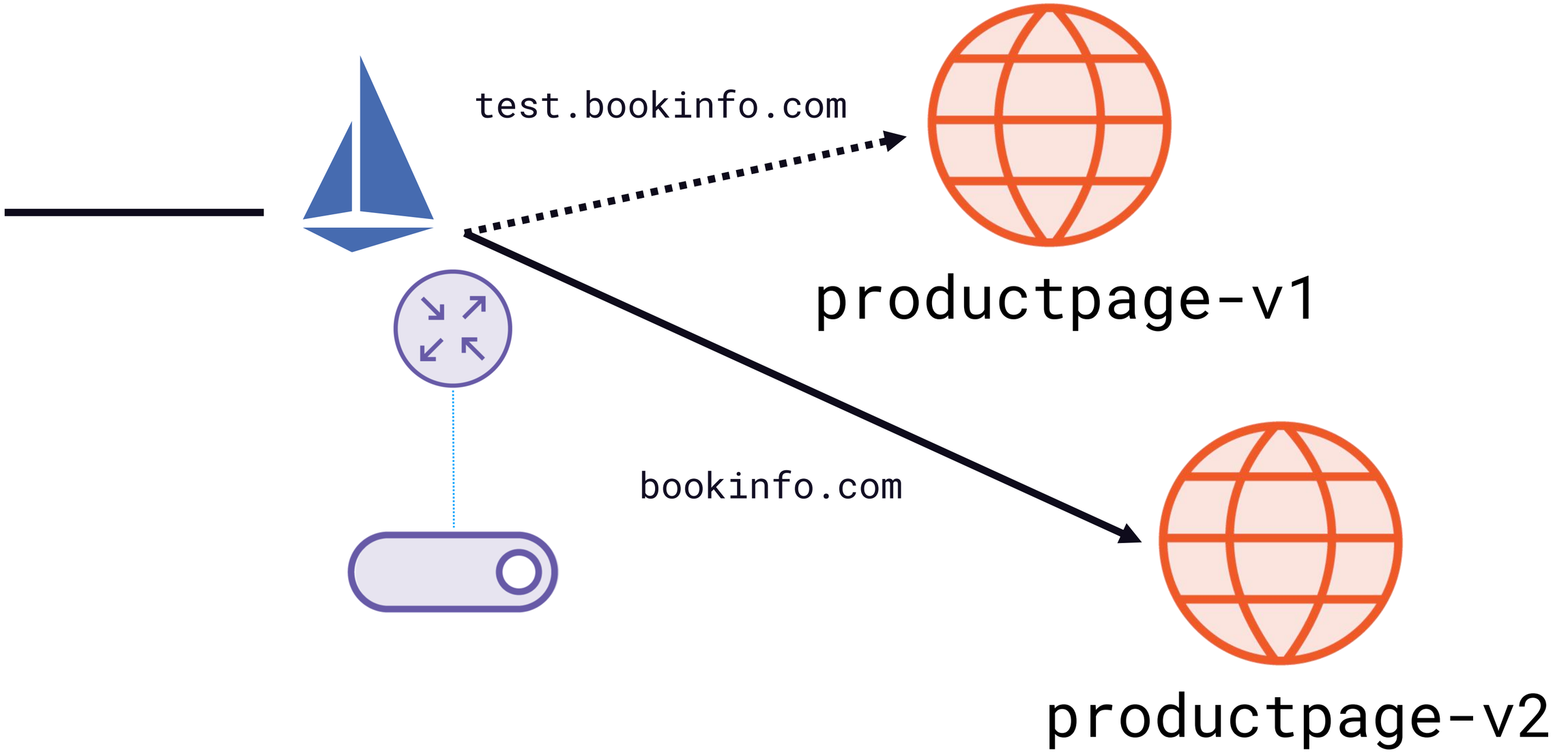
labels: app=productpage
version=v2

Demo

Blue/green deployment

- Live and test domains
- Switch from test to live
- And back again


bookinfo.com
productpage-v1
test.bookinfo.com
productpage-v2


test.bookinfo.com
productpage-v1
bookinfo.com
productpage-v2

Blue/Green Deployments

ProductPage-Deployment.yaml

```yaml
template:
  metadata:
    labels:
      app: productpage
      version: v2
  spec:
    # ...
```

DestinationRule.yaml

```yaml
spec:
  host: productpage
  subsets:
  - name: v1
    labels:
      version: v1
  - name: v2
    labels:
      version: v2
```

Blue/Green Deployments

VirtualService-Live.yaml

```yaml
hosts:
  - bookinfo.local
gateways:
  - bookinfo-gateway
http:
  - route:
      - destination:
          host: productpage
          subset: v1
```

VirtualService-Test.yaml

```yaml
hosts:
  - test.bookinfo.local
gateways:
  - bookinfo-gateway
http:
  - route:
      - destination:
          host: productpage
          subset: v2
```

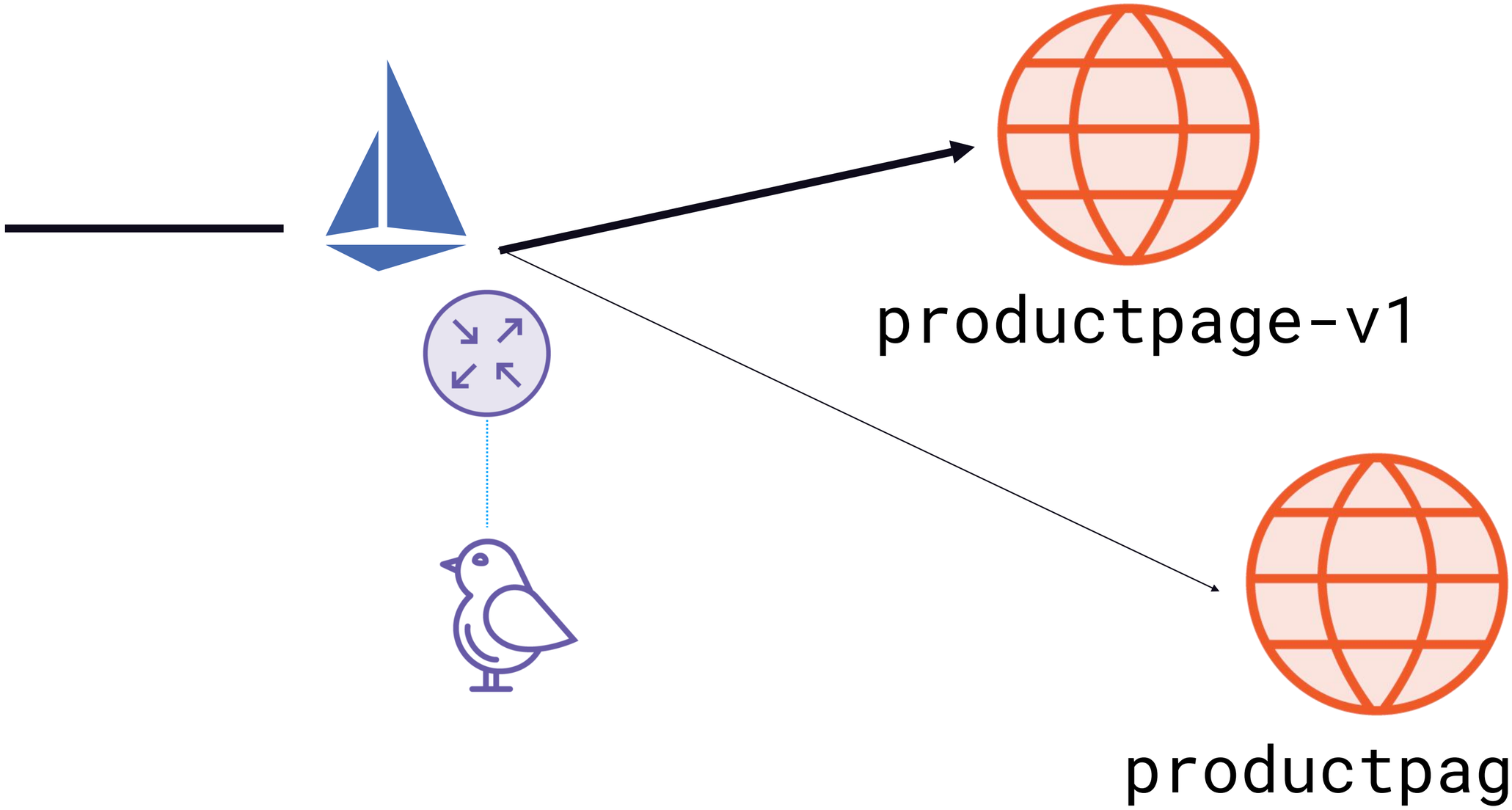
Blue/Green Deployments

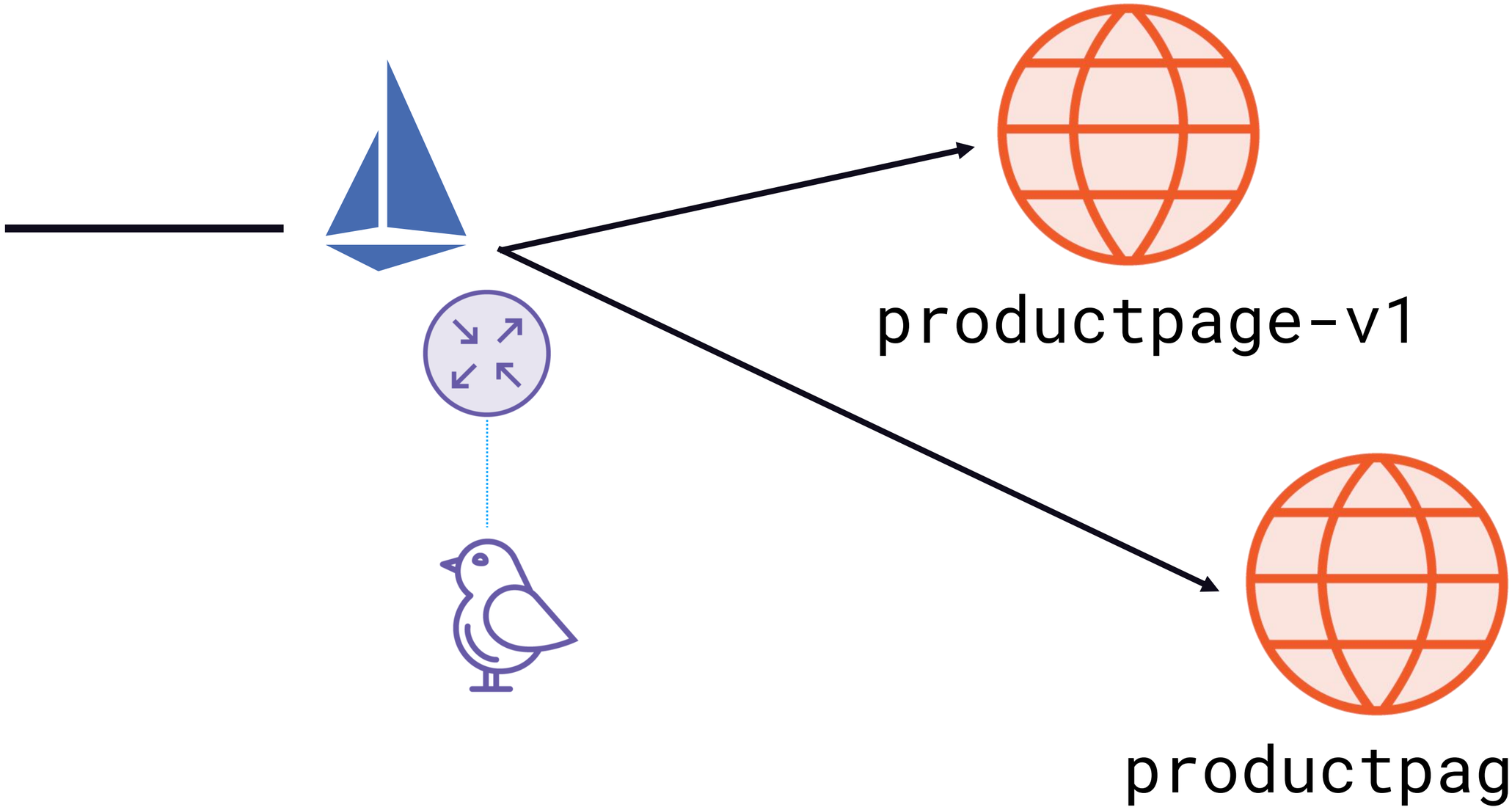
VirtualService-Live.yaml

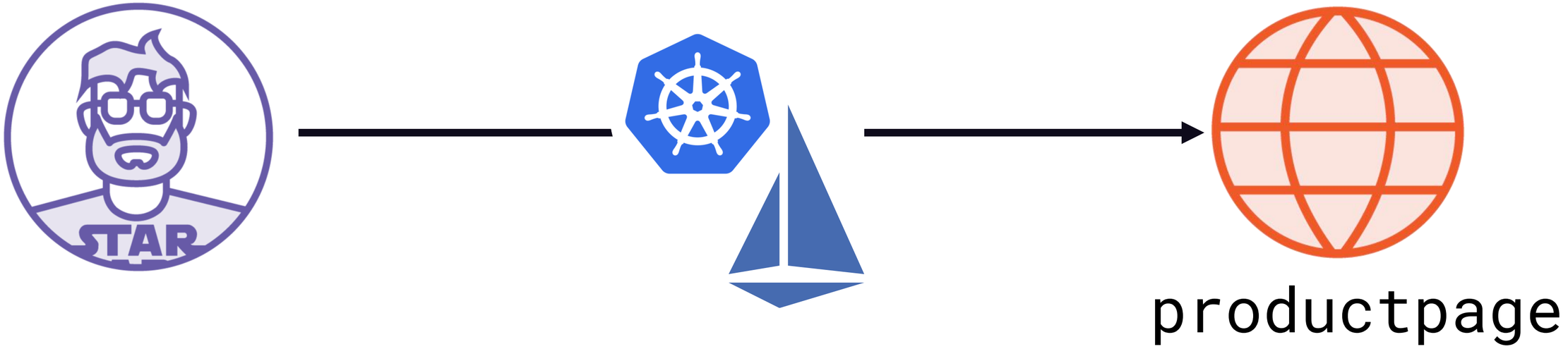
```yaml
hosts:
  - bookinfo.local
gateways:
  - bookinfo-gateway
http:
  - route:
      - destination:
          host: productpage
          subset: v2
```

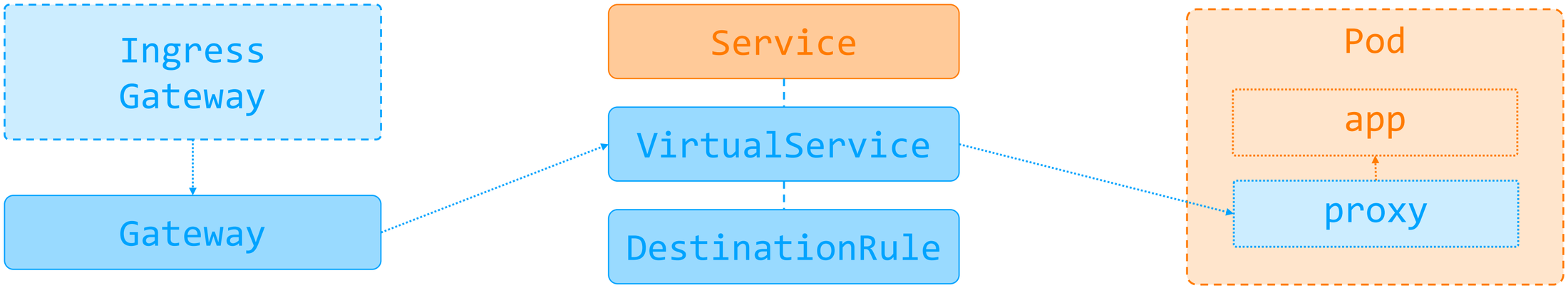
VirtualService-Test.yaml

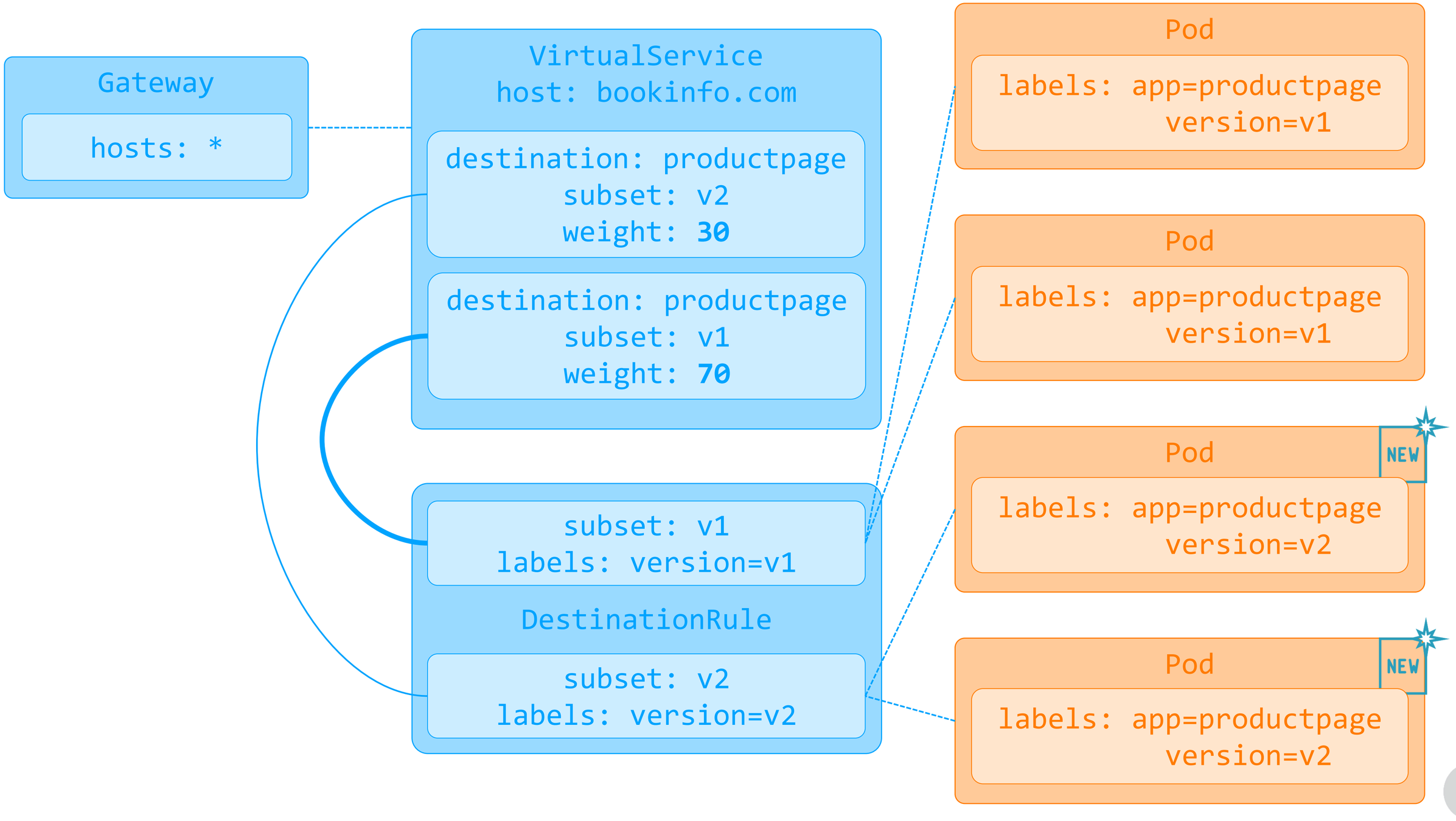
```yaml
hosts:
  - test.bookinfo.local
gateways:
  - bookinfo-gateway
http:
  - route:
      - destination:
          host: productpage
          subset: v1
```

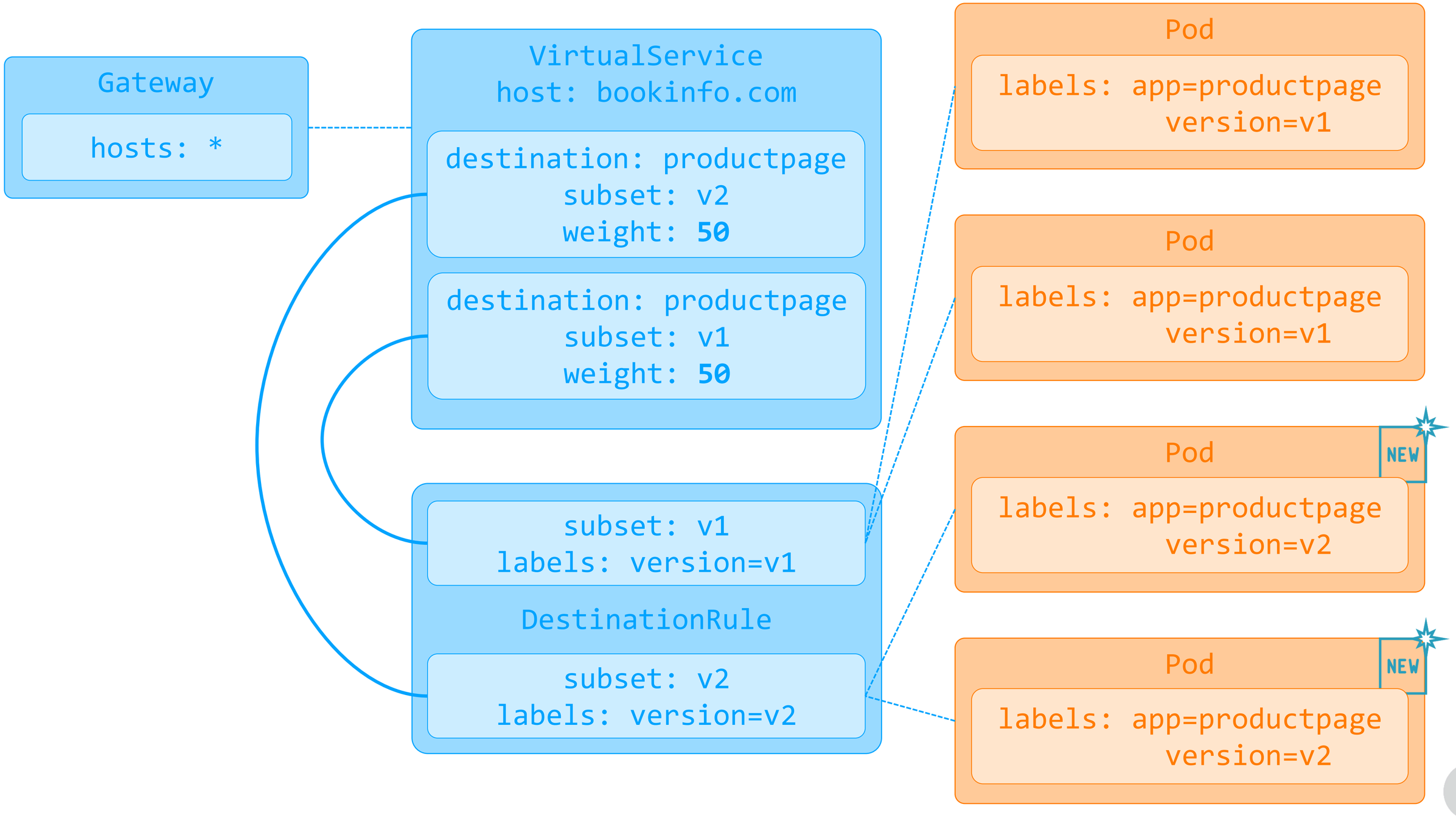

productpage-v1
productpage-v2

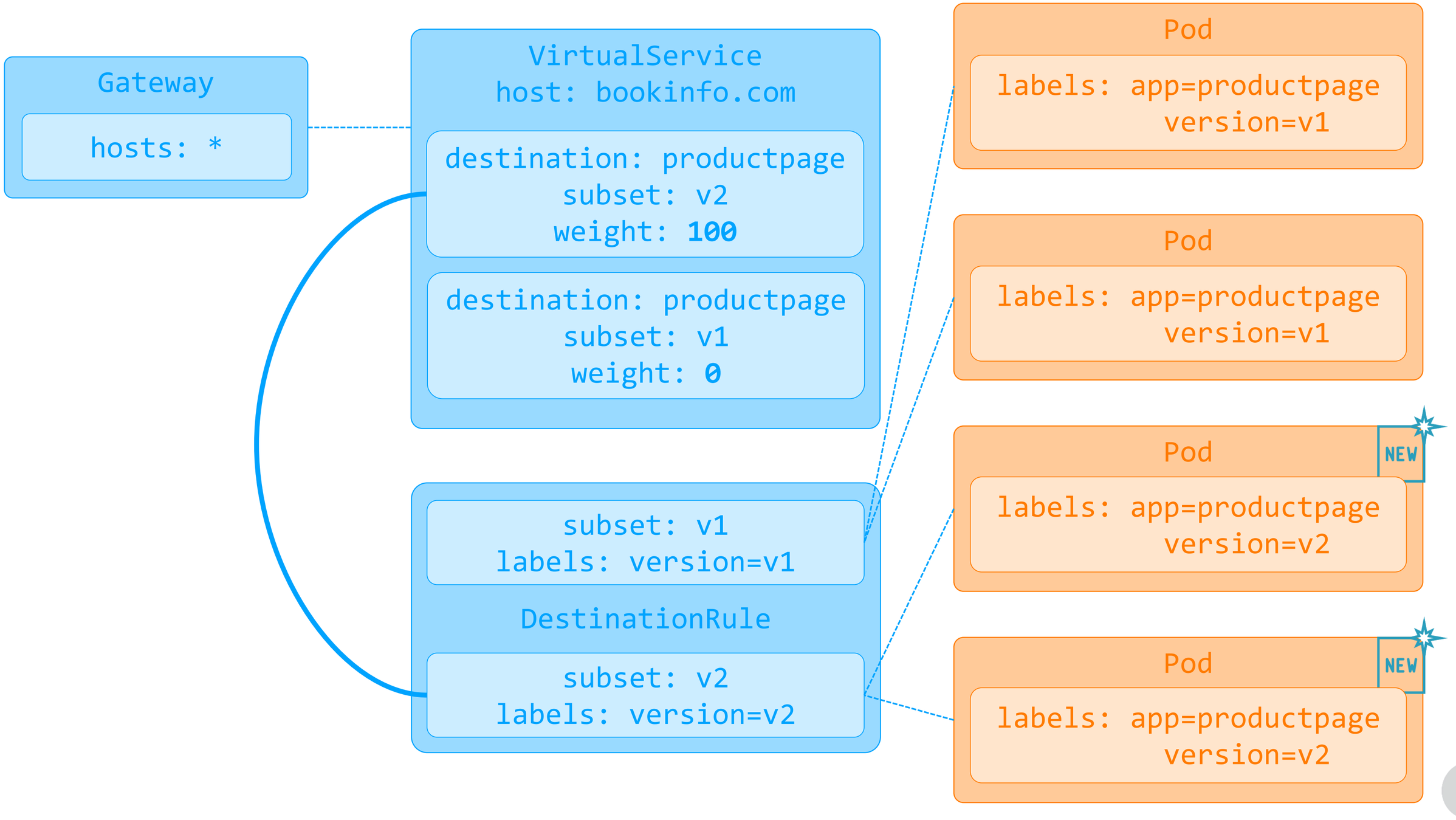

productpage-v1
productpage-v2


productpage


Ingress
Gateway
Gateway
Service
VirtualService
DestinationRule
Pod
app
proxy


Gateway
hosts: *
VirtualService
host: bookinfo.com
destination: productpage
subset: v2
weight: 30
destination: productpage
subset: v1
weight: 70
subset: v1
labels: version=v1
DestinationRule
subset: v2
labels: version=v2
Pod
labels: app=productpage
version=v1
Pod
labels: app=productpage
version=v1
Pod
NEW
labels: app=productpage
version=v2
Pod
NEW
labels: app=productpage
version=v2

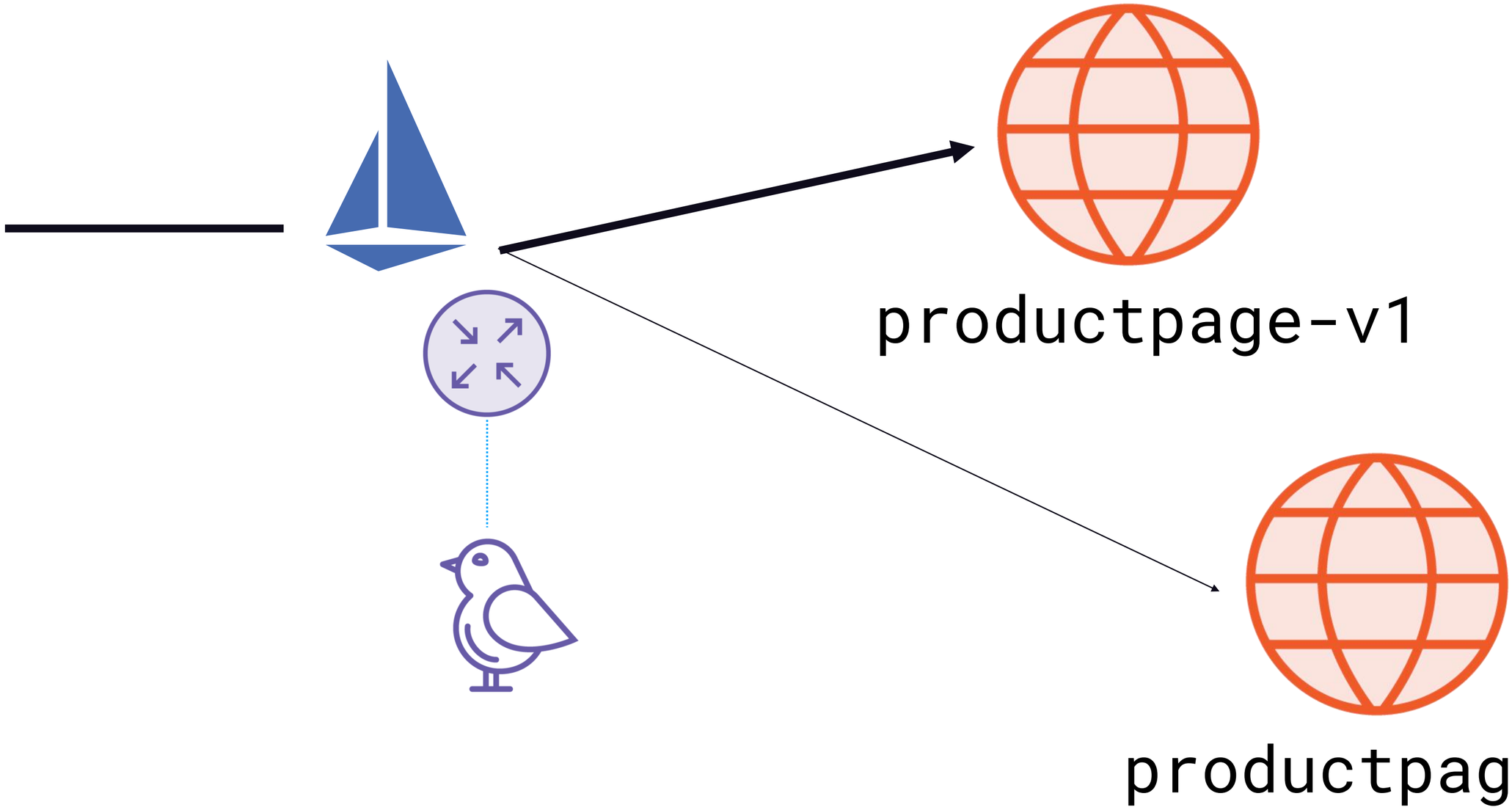

Gateway
hosts: *
VirtualService
host: bookinfo.com
destination: productpage
subset: v2
weight: 50
destination: productpage
subset: v1
weight: 50
subset: v1
labels: version=v1
DestinationRule
subset: v2
labels: version=v2
Pod
labels: app=productpage
version=v1
Pod
labels: app=productpage
version=v1
Pod
NEW
labels: app=productpage
version=v2
Pod
NEW
labels: app=productpage
version=v2


Gateway
hosts: *
VirtualService
host: bookinfo.com
destination: productpage
subset: v2
weight: 100
destination: productpage
subset: v1
weight: 0
subset: v1
labels: version=v1
DestinationRule
subset: v2
labels: version=v2
Pod
labels: app=productpage
version=v1
Pod
labels: app=productpage
version=v1
Pod
NEW
labels: app=productpage
version=v2
Pod
NEW
labels: app=productpage
version=v2

Demo

Canary deployment

- Traffic weighted to v1
- Shift traffic to v2
- Canary with sticky sessions


productpage-v1
productpage-v2

Canary Deployment

VirtualService-70-30.yaml

```yaml
http:
  - route:
      - destination:
          host: productpage
          subset: v1
        weight: 70
      - destination:
          host: productpage
          subset: v2
        weight: 30
```

VirtualService-5-95.yaml

```yaml
http:
  - route:
      - destination:
          host: productpage
          subset: v1
        weight: 5
      - destination:
          host: productpage
          subset: v2
        weight: 95
```

Canary Deployment

VirtualService-70-30.yaml

```yaml
http:
  - route:
      - destination:
          host: productpage
          subset: v1
        weight: 70
      - destination:
          host: productpage
          subset: v2
        weight: 30
```

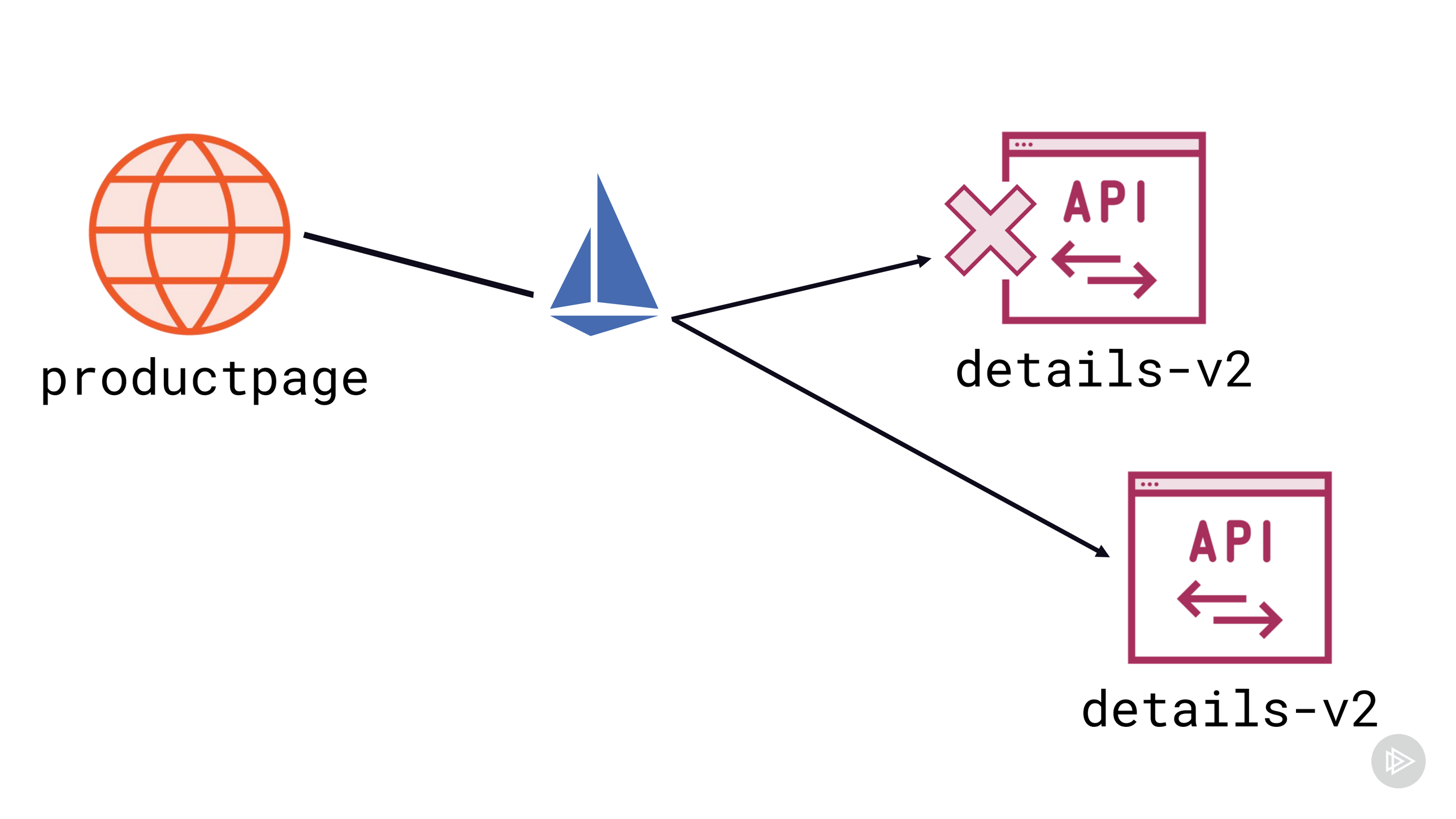
VirtualService-5-95.yaml

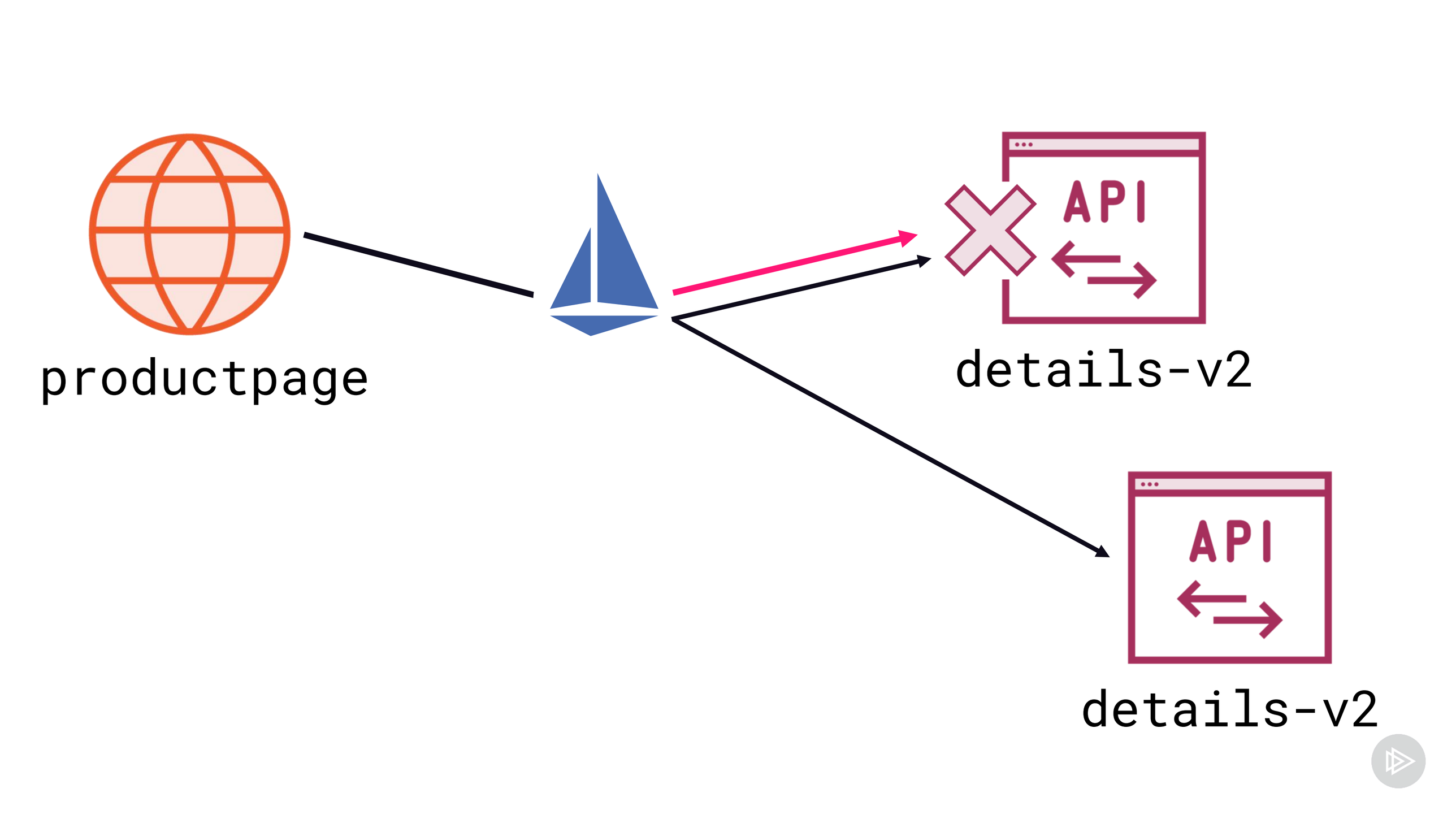
```yaml
http:
  - route:
      - destination:
          host: productpage
          subset: v1
        weight: 5
      - destination:
          host: productpage
          subset: v2
        weight: 95
```

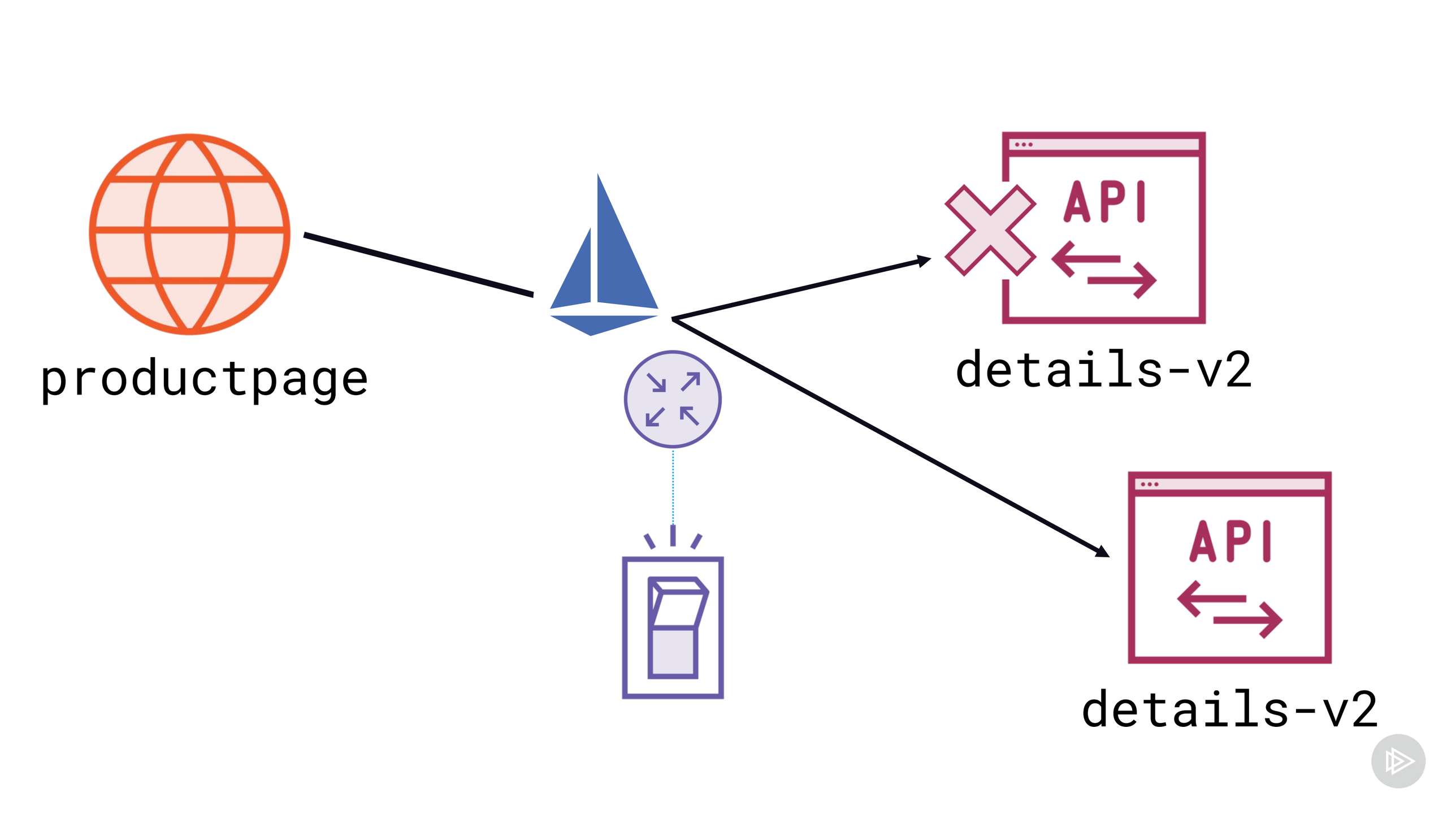
Canary Deployment with Sticky Sessions

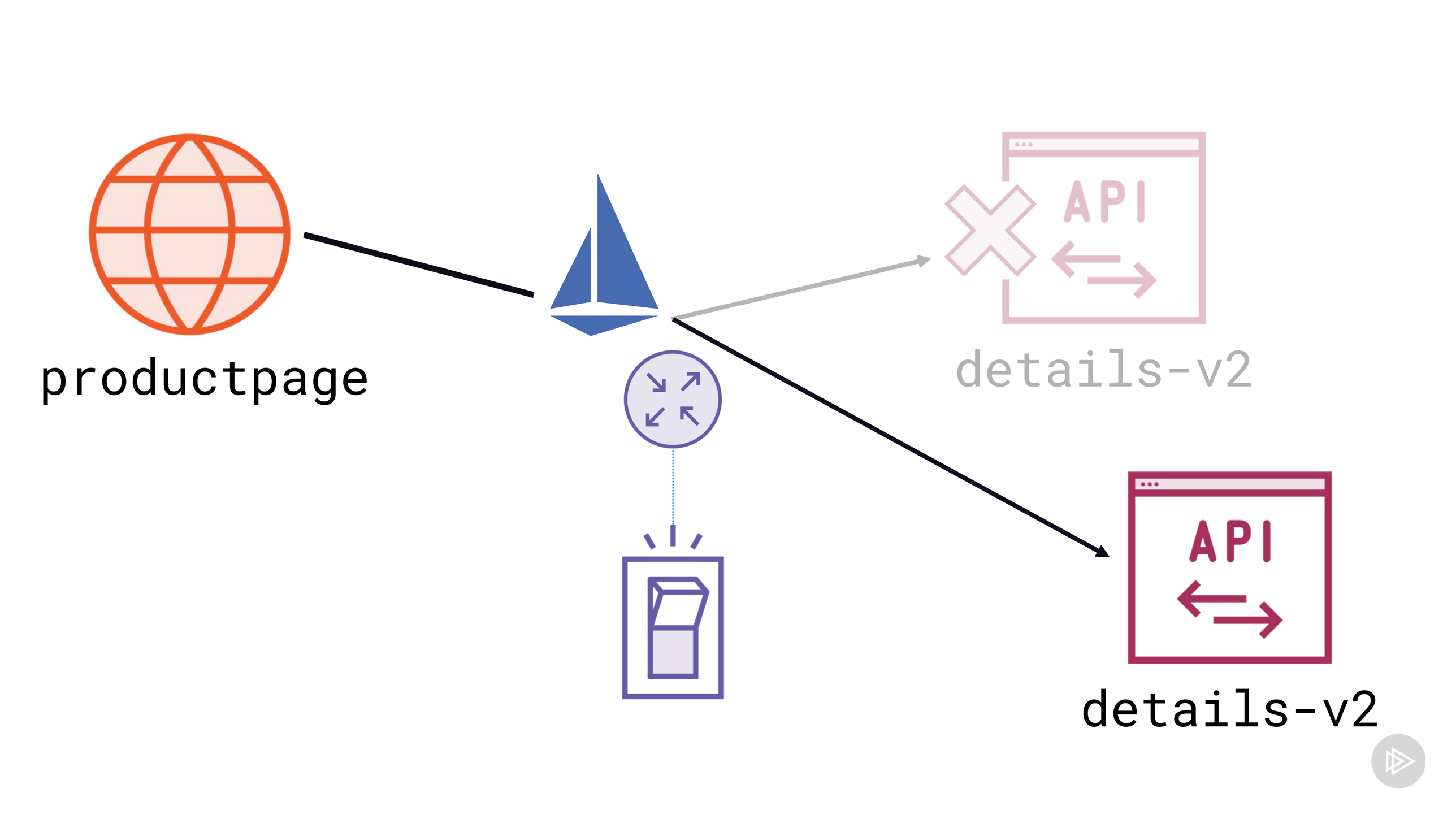
VirtualService.yaml

```yaml
http:
  - match:
      - headers:
          cookie:
            regex: "^(.*?;)?(product-page=v2)(;.*)?$"
    route:
      - destination:
          host: productpage
          subset: v2
  - route:
      - destination:
          host: productpage
          subset: v1
        weight: 70
      - destination:
          host: productpage
          subset: v2
        weight: 30
```


productpage
API
details-v2
API
details-v2


productpage
API
details-v2
API
details-v2

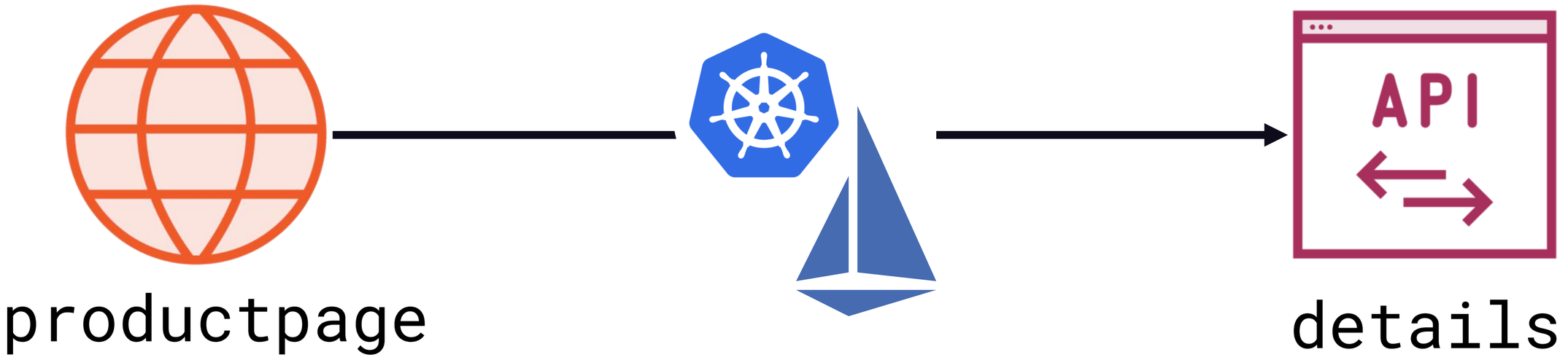

productpage
API
details-v2
API
details-v2

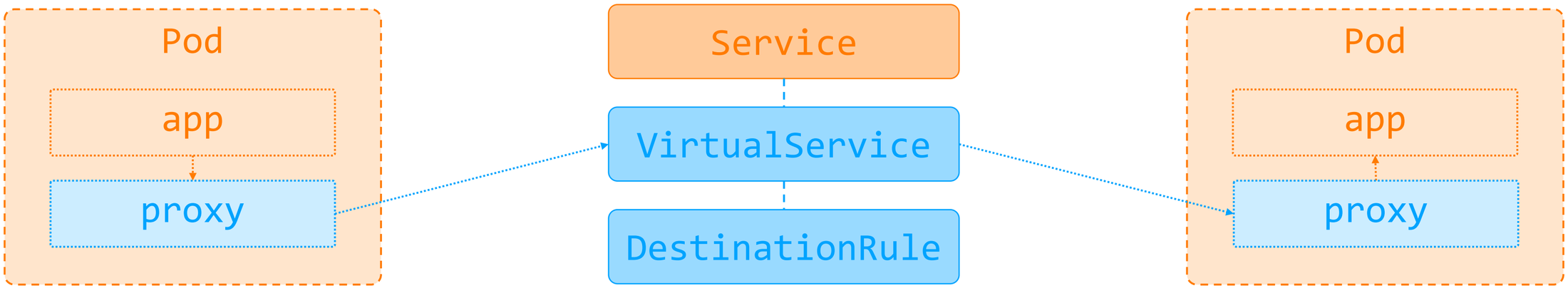

productpage
API
details-v2
API
details-v2

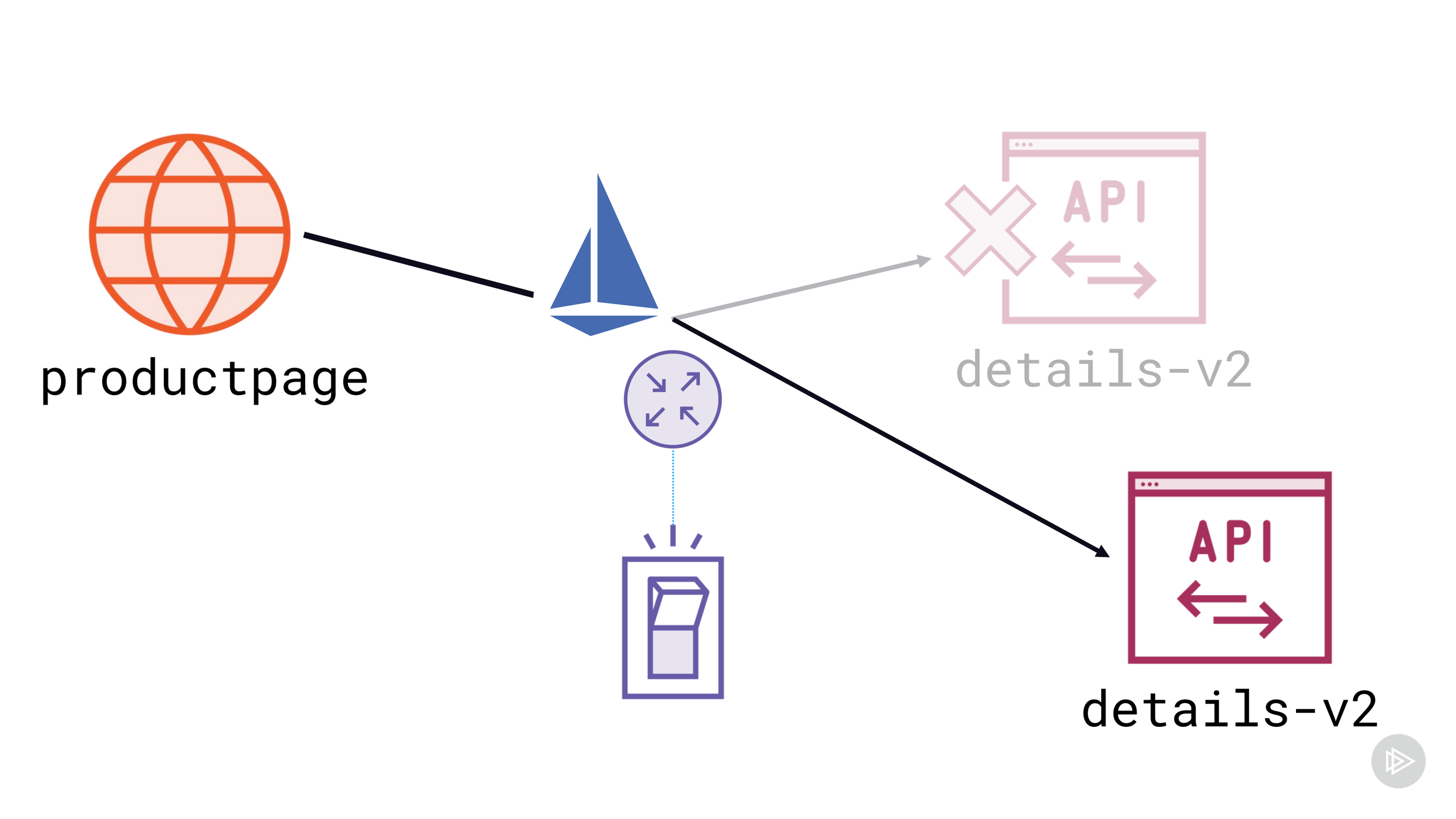
Readiness Probes in Pods

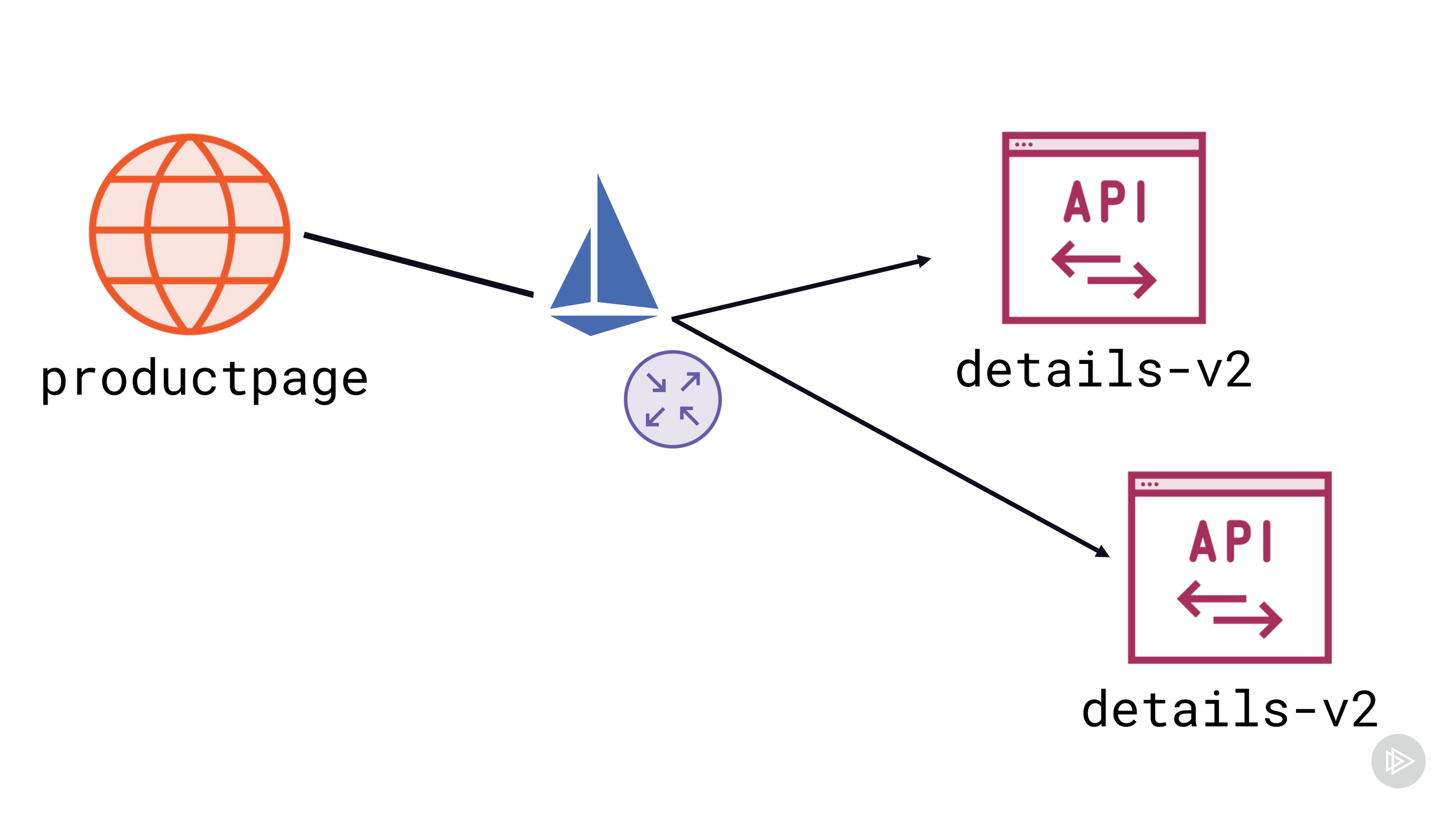
Deployment.yaml

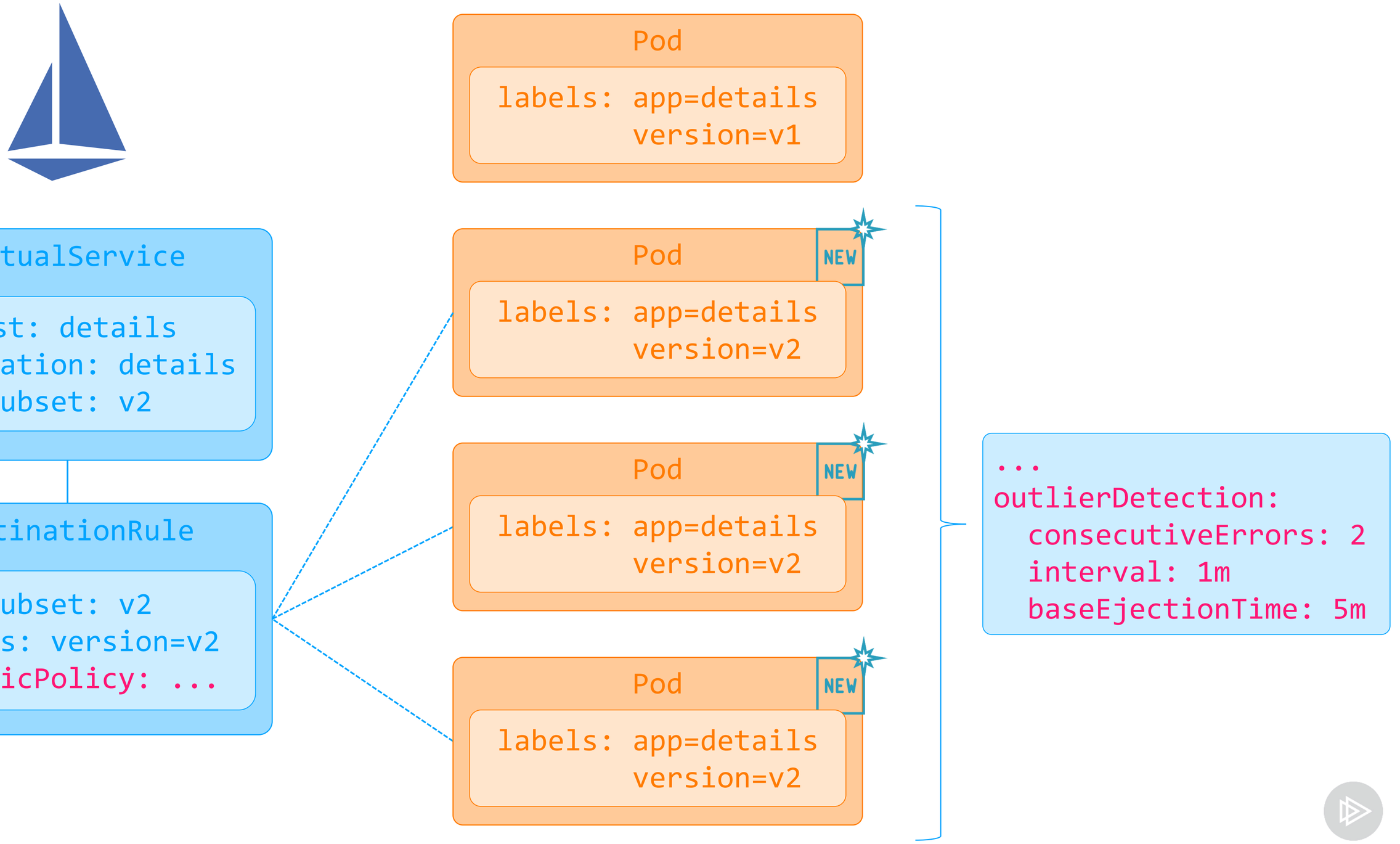
```yaml
spec:
 containers:
 - name: app
   image: web/app
   ports:
     - containerPort: 8080
       name: http
   readinessProbe:
     httpGet:
       path: /health
       port: http
     periodSeconds: 30
     failureThreshold: 1
```

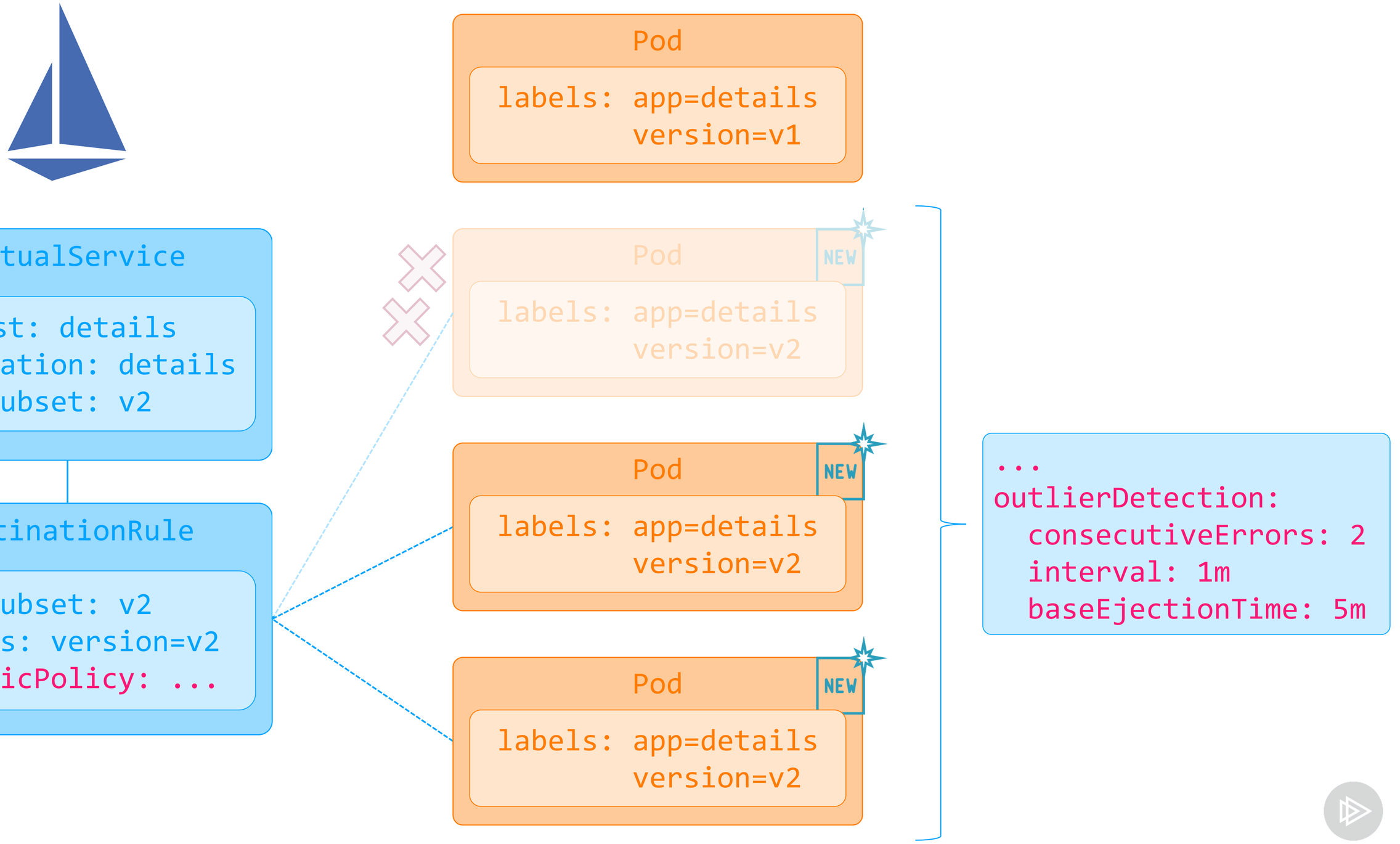

API
productpage
details


Pod
app
proxy
Service
VirtualService
DestinationRule
Pod
app
proxy


productpage
API
details-v2
API
details-v2


productpage
API
details-v2
API
details-v2

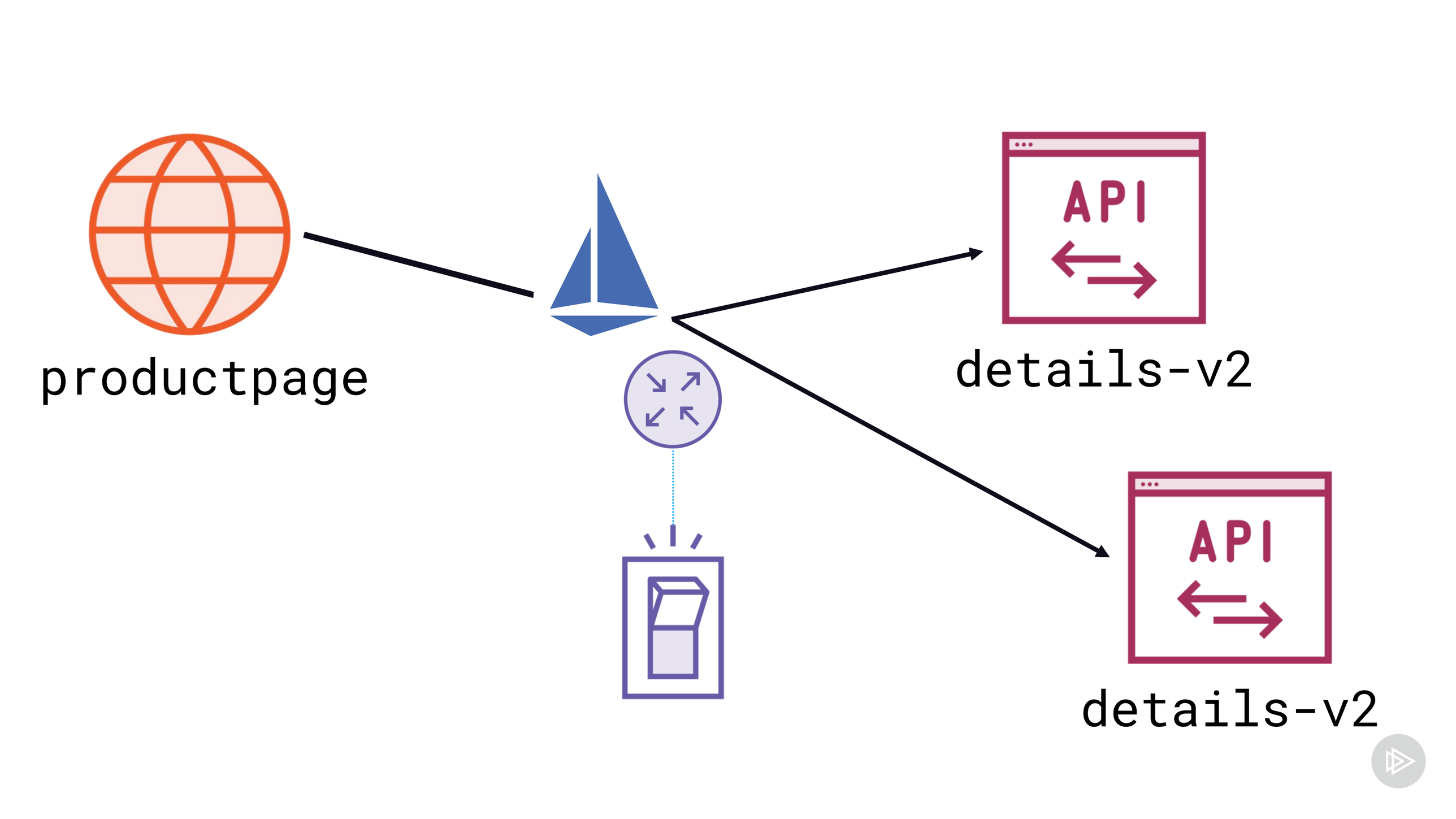

VirtualService
host: details
destination: details
subset: v2
DestinationRule
subset: v2
labels: version=v2
trafficPolicy: ...
Pod
labels: app=details
version=v1
Pod
NEW
labels: app=details
version=v2
Pod
NEW
labels: app=details
version=v2
Pod
NEW
labels: app=details
version=v2
...
outlierDetection:
consecutiveErrors: 2
interval: 1m
baseEjectionTime: 5m

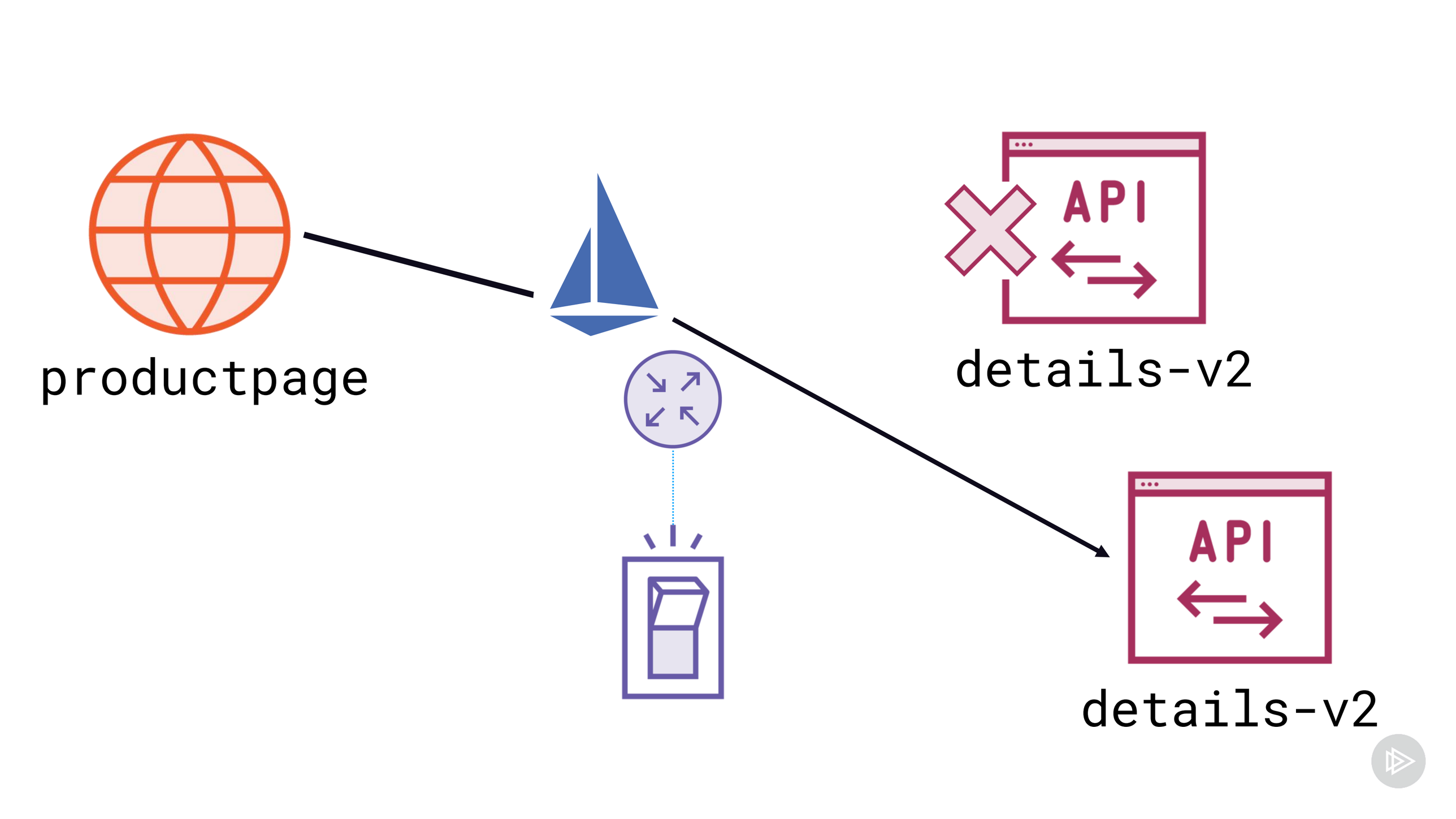

VirtualService
host: details
destination: details
subset: v2
DestinationRule
subset: v2
labels: version=v2
trafficPolicy: ...
Pod
labels: app=details
version=v1
Pod
NEW
labels: app=details
version=v2
Pod
NEW
labels: app=details
version=v2
Pod
NEW
labels: app=details
version=v2
...
outlierDetection:
consecutiveErrors: 2
interval: 1m
baseEjectionTime: 5m

Demo

Circuit breaker

- Deploy service update
- Failing instances
- Route with outlier detection


productpage
API
details-v2
API
details-v2


productpage
API
details-v2
API
details-v2

Circuit Breaker with Outlier Detection

DestinationRule.yaml

```yaml
spec:
  host: details
  subsets:
    - name: v1
      labels:
        version: v1
    - name: v2
      labels:
        version: v2
  trafficPolicy:
    outlierDetection:
      consecutiveErrors: 2
      interval: 1m
      baseEjectionTime: 5m
      maxEjectionPercent: 100
```

Summary

Managing traffic

- Gateway
- VirtualService
- DestinationRule

Deployment strategies

- Dark launch
- Blue/green
- Canary

Self-healing apps

- Circuit breaker

Up Next:

Securing Communication with Mutual TLS