Cross-Cluster Calls Made Easy

with Istio 1.1

Matt Turner

@mt165
mt165.co.uk

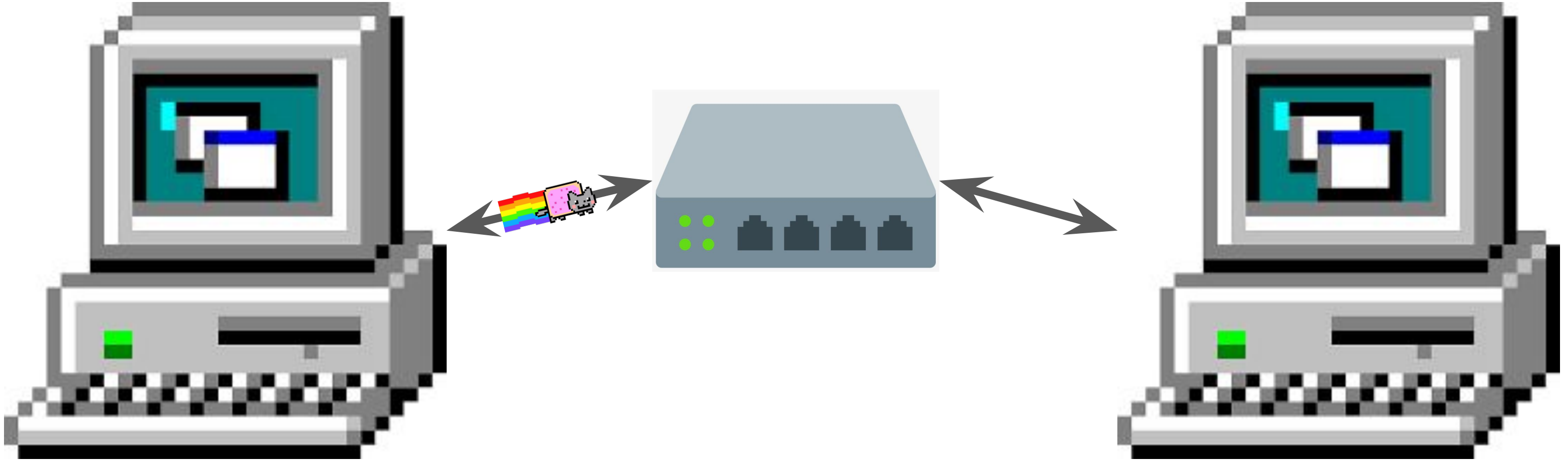
native wave

Outline

- Intra-cluster recap
- Inter-cluster calls
- Cross-cluster with Istio
- Istio 1.1
- Demo

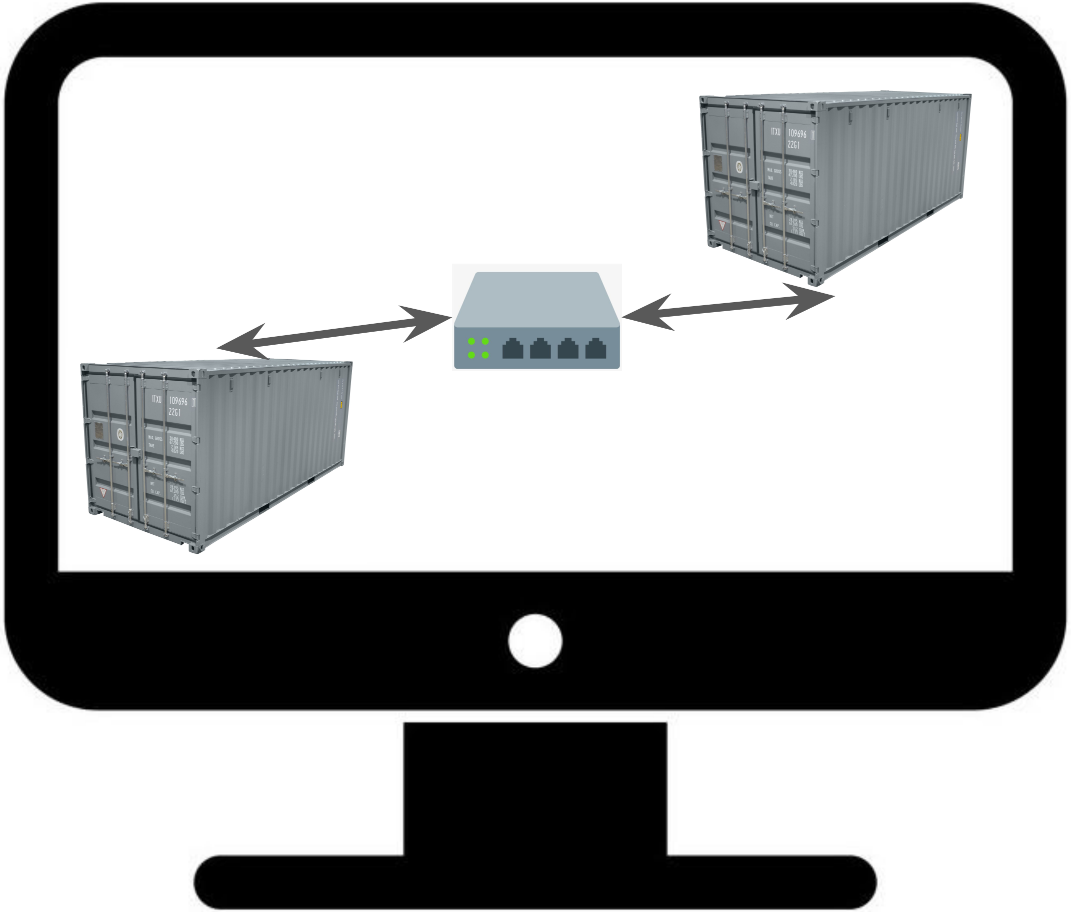
Intra-Cluster Recap

Intra-Cluster Recap

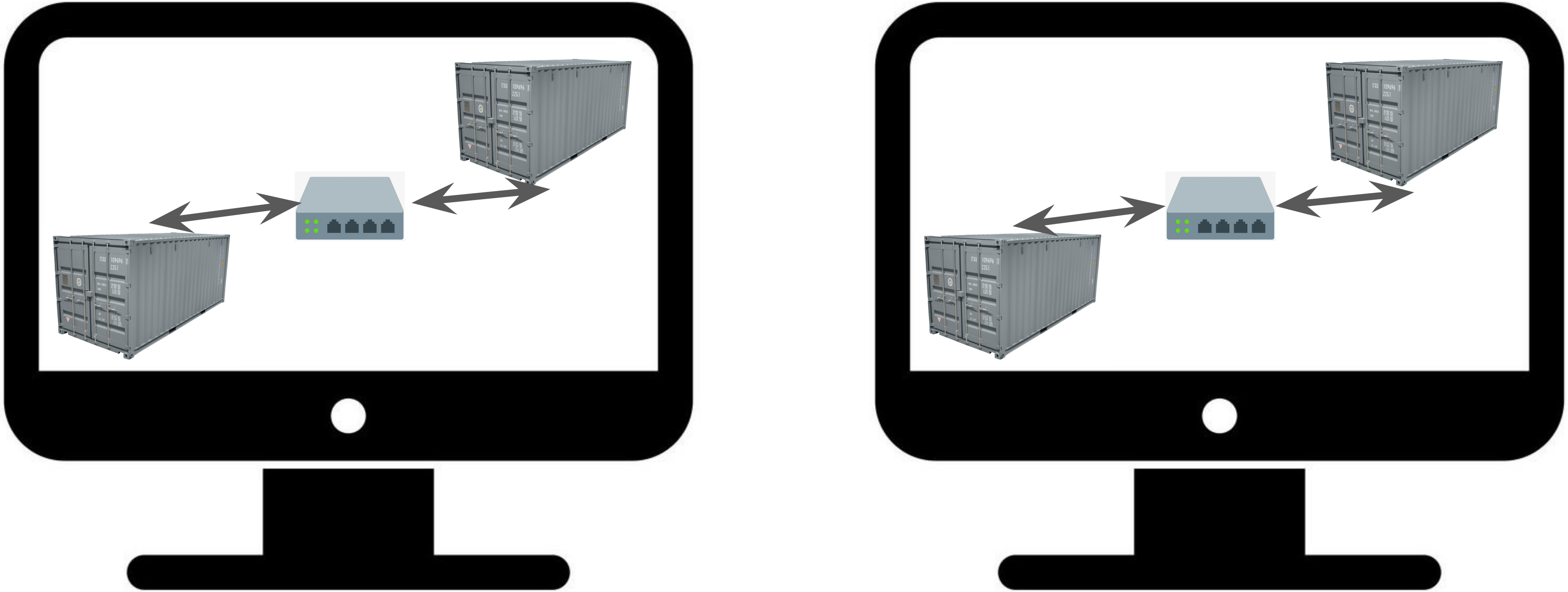


Intra-Cluster Recap

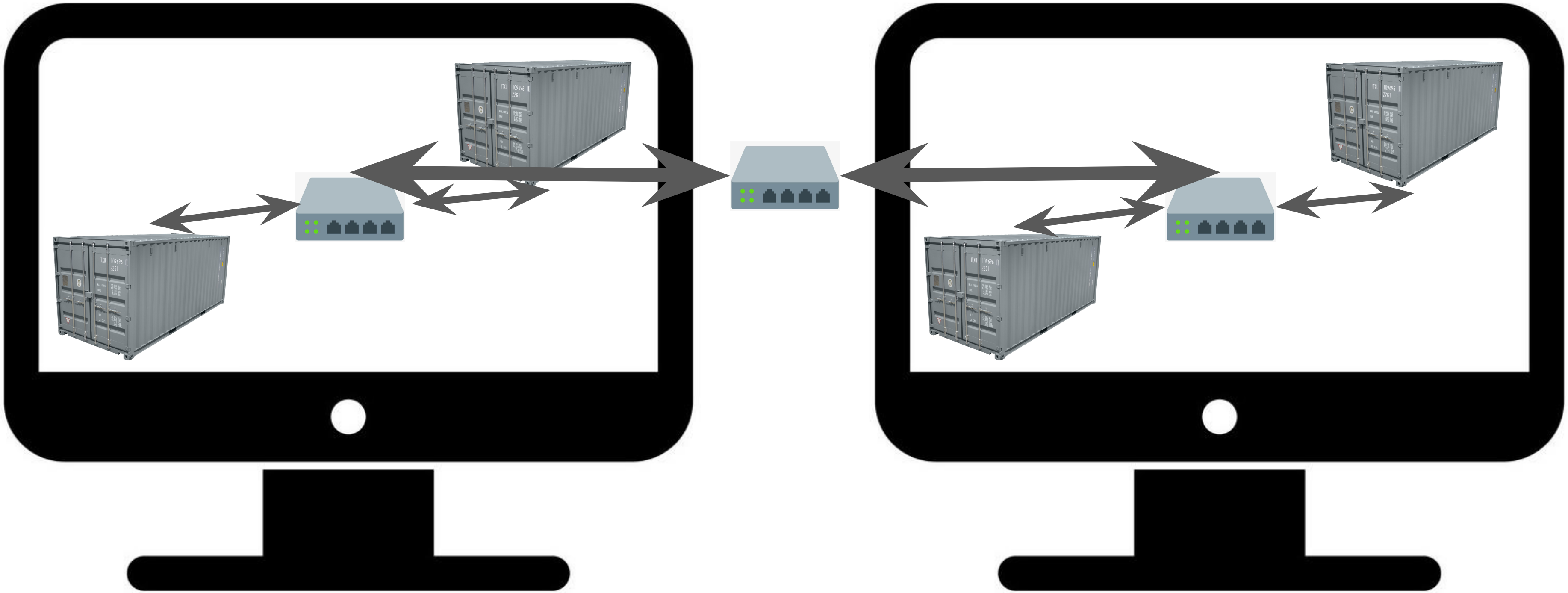
Intra-Cluster Recap



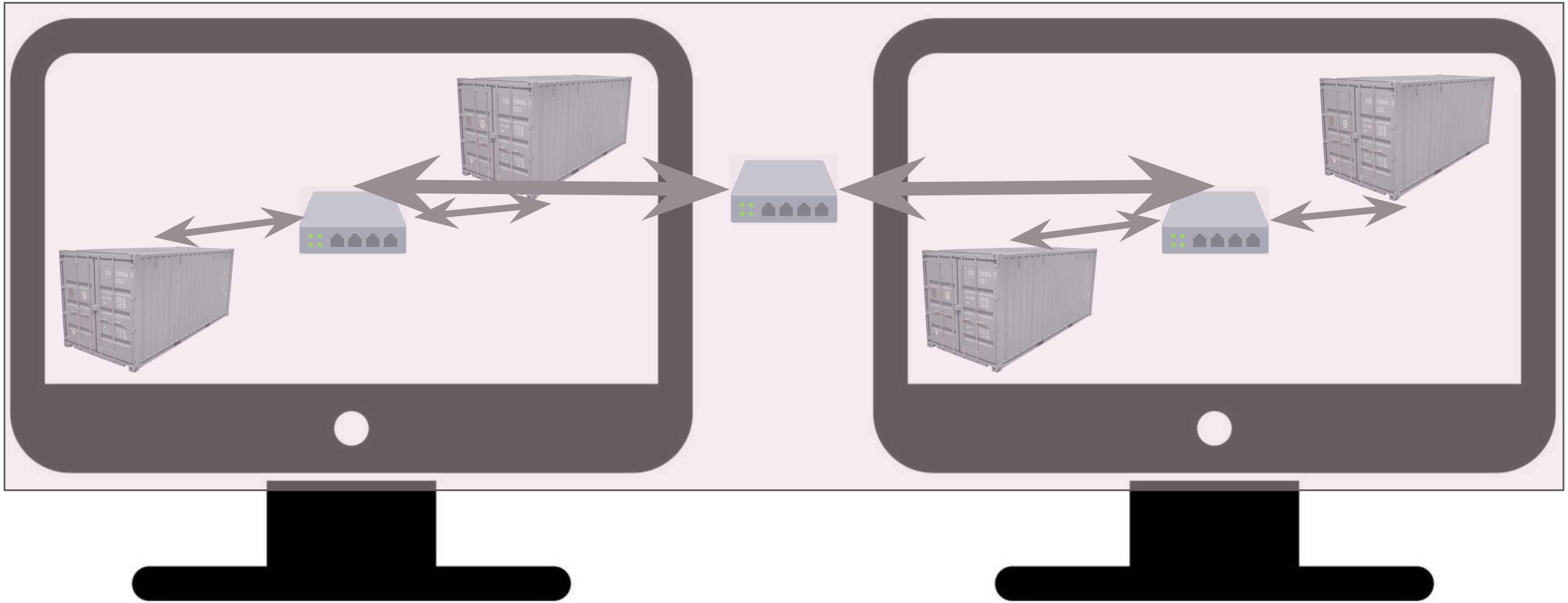
Intra-Cluster Recap



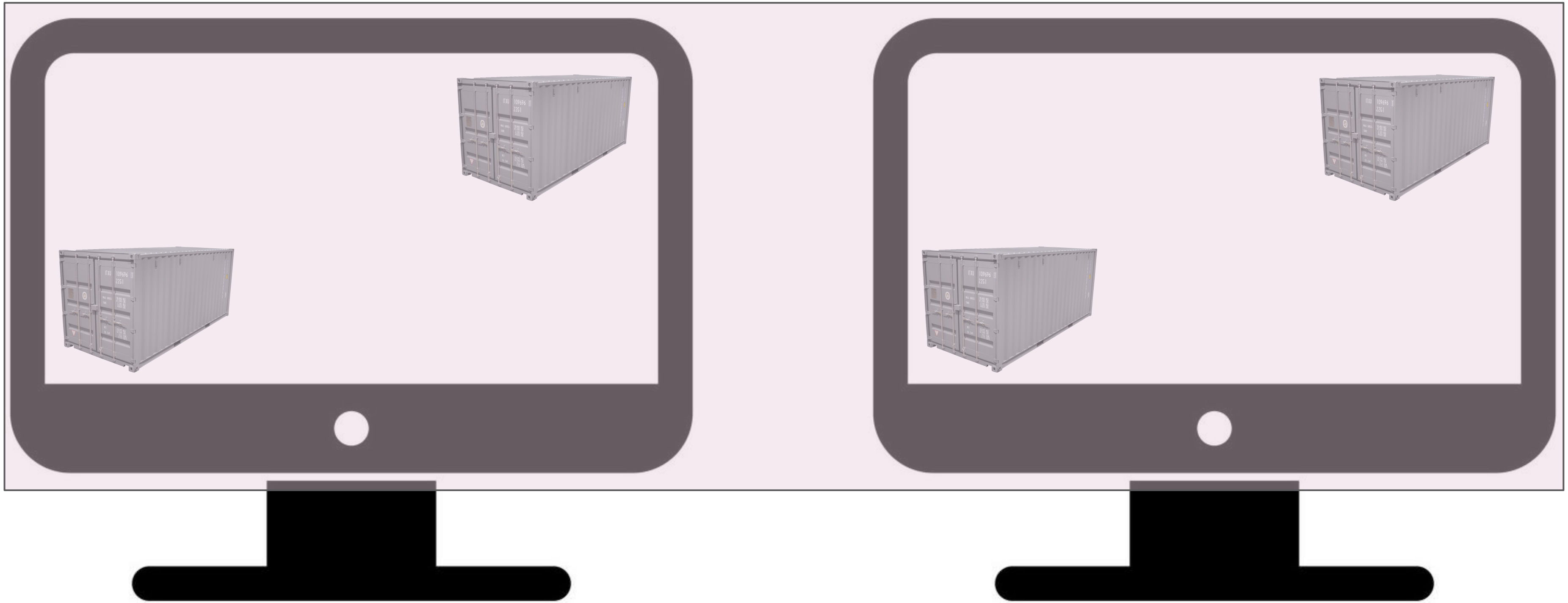
Intra-Cluster Recap



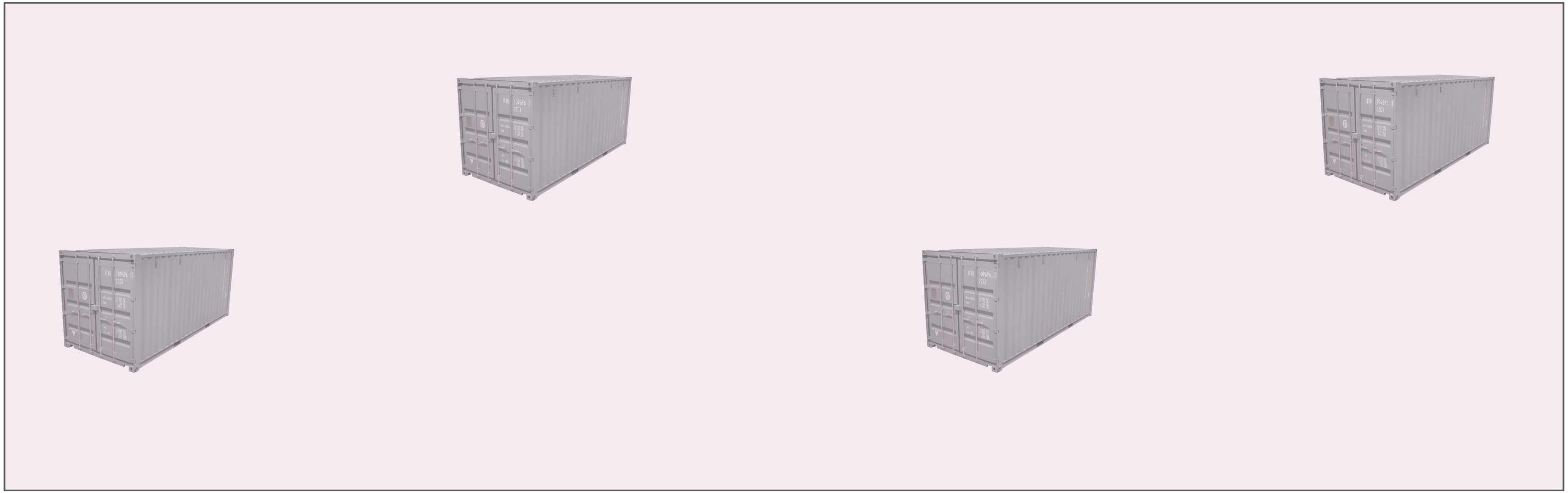
Intra-Cluster Recap



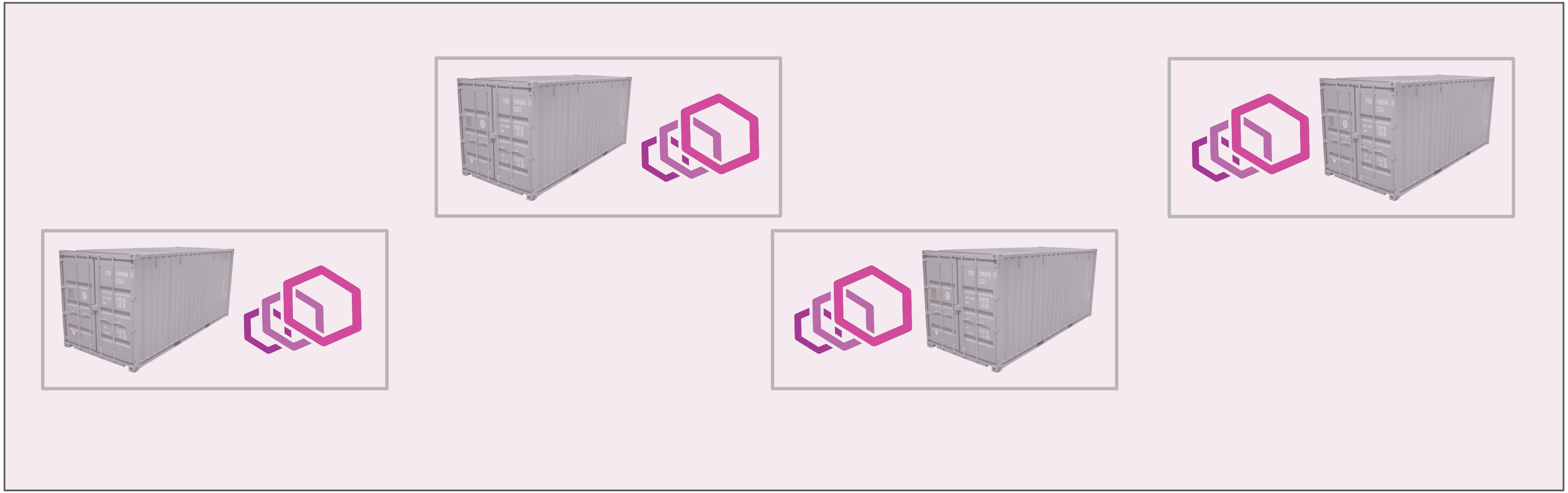
Intra-Cluster Recap



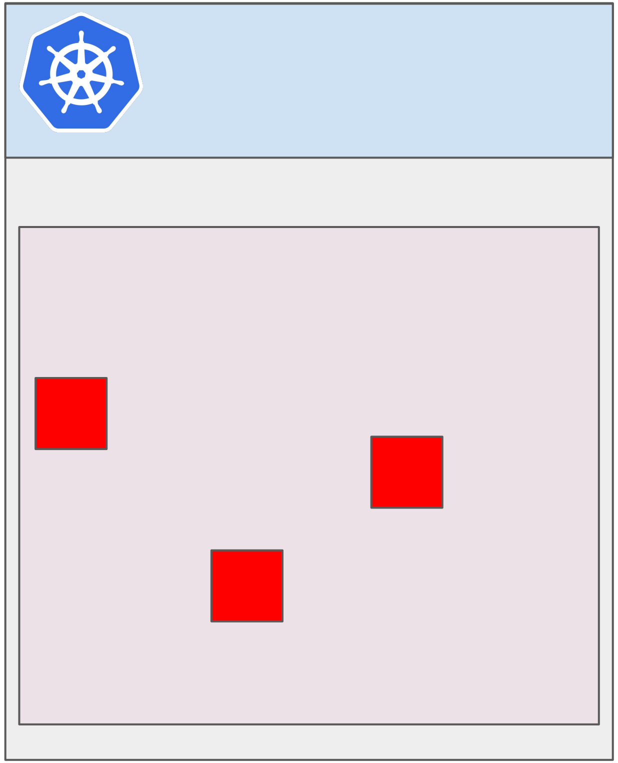
Intra-Cluster Recap



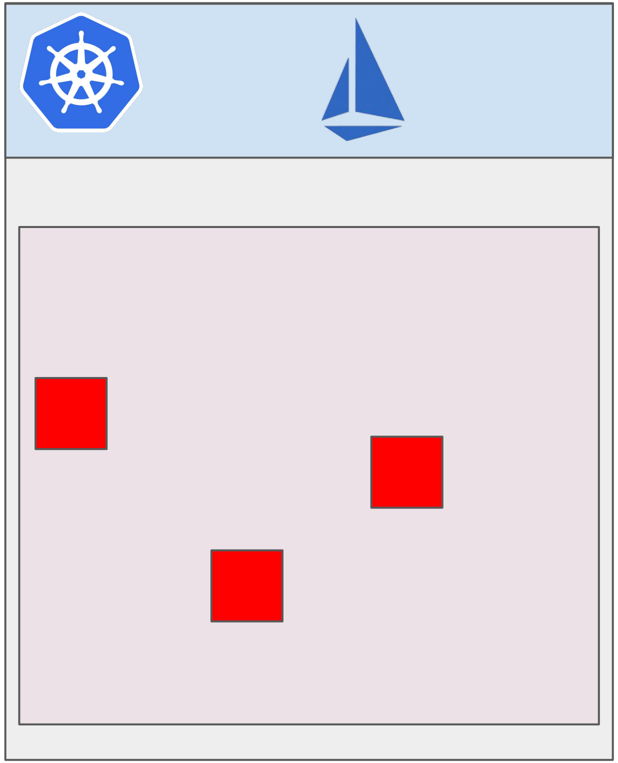
Service Meshes



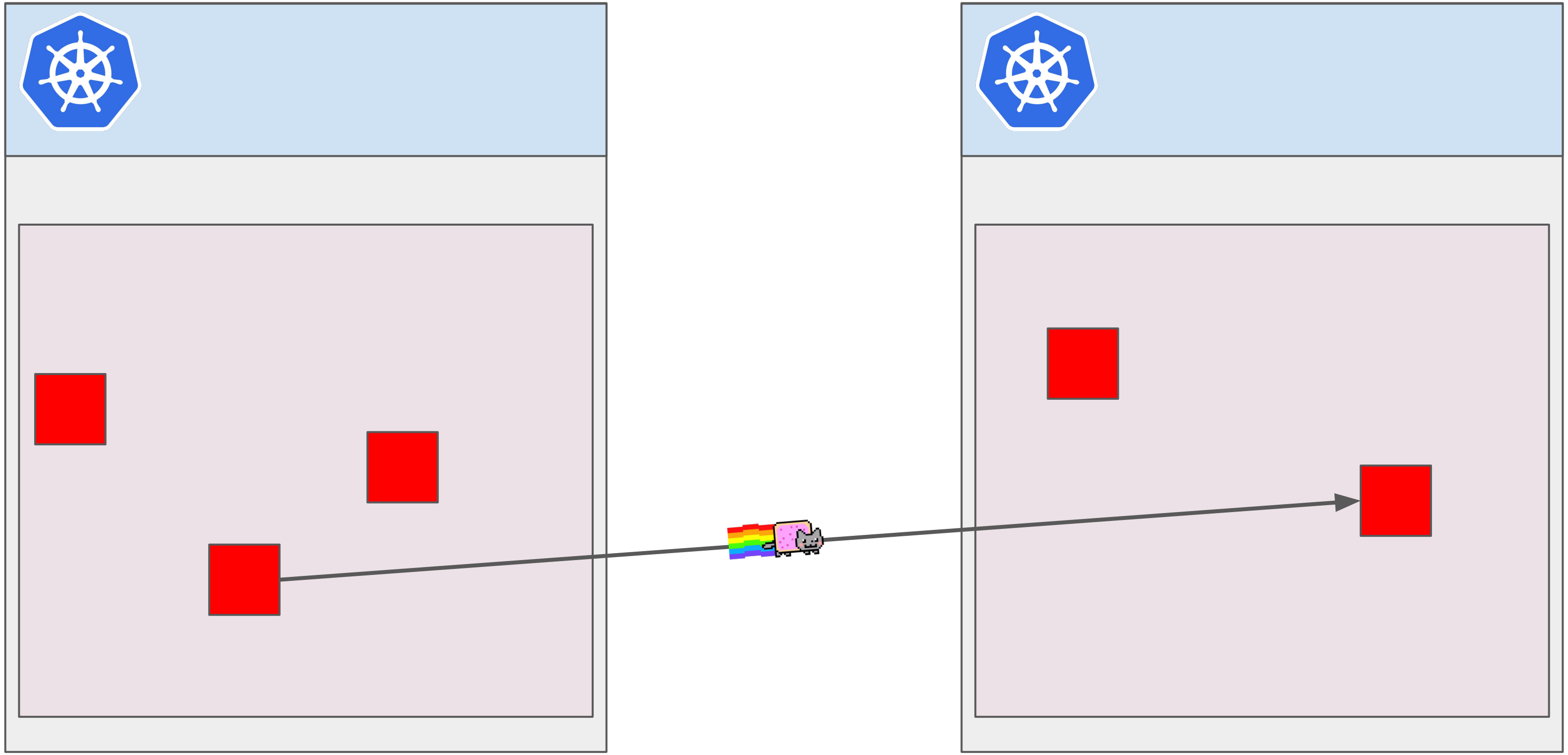
Service Meshes



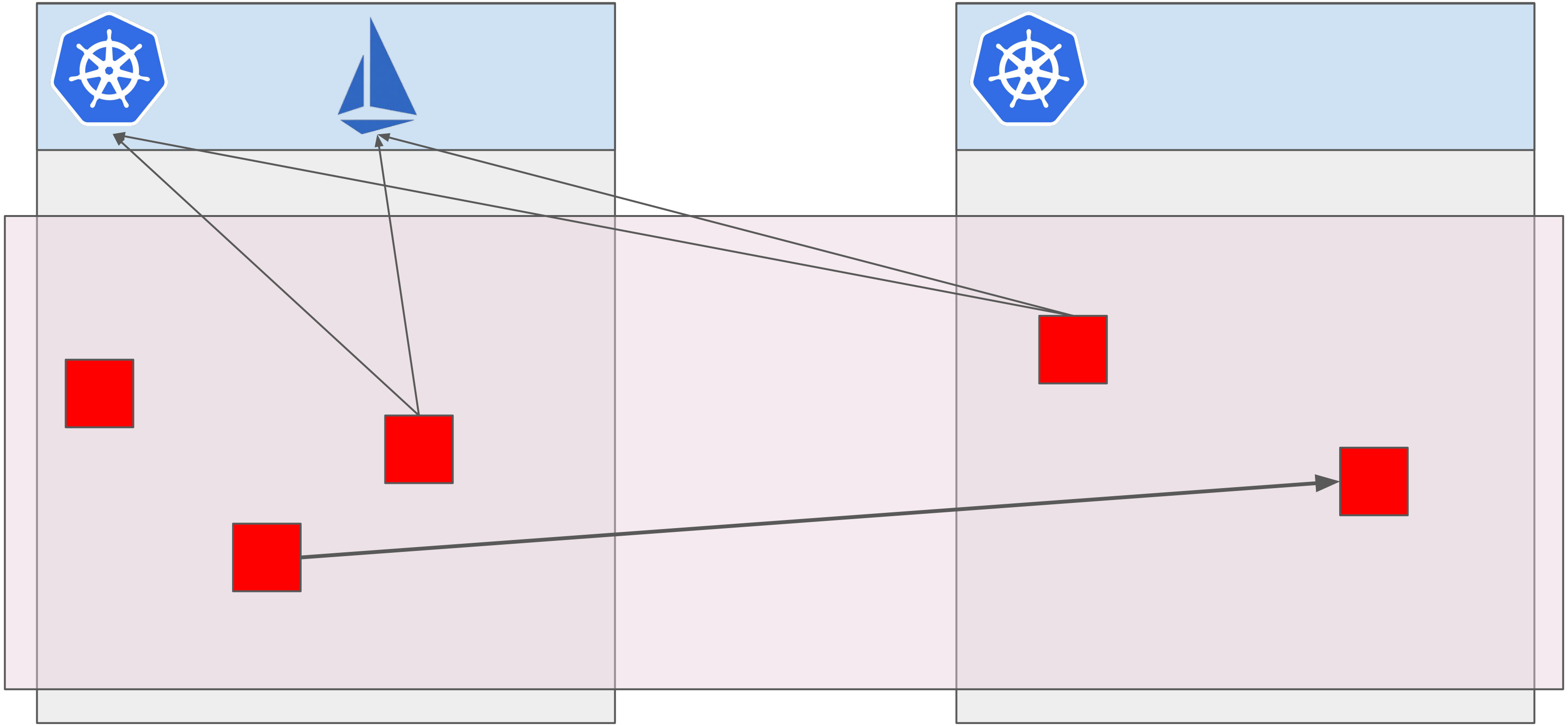
Service Meshes



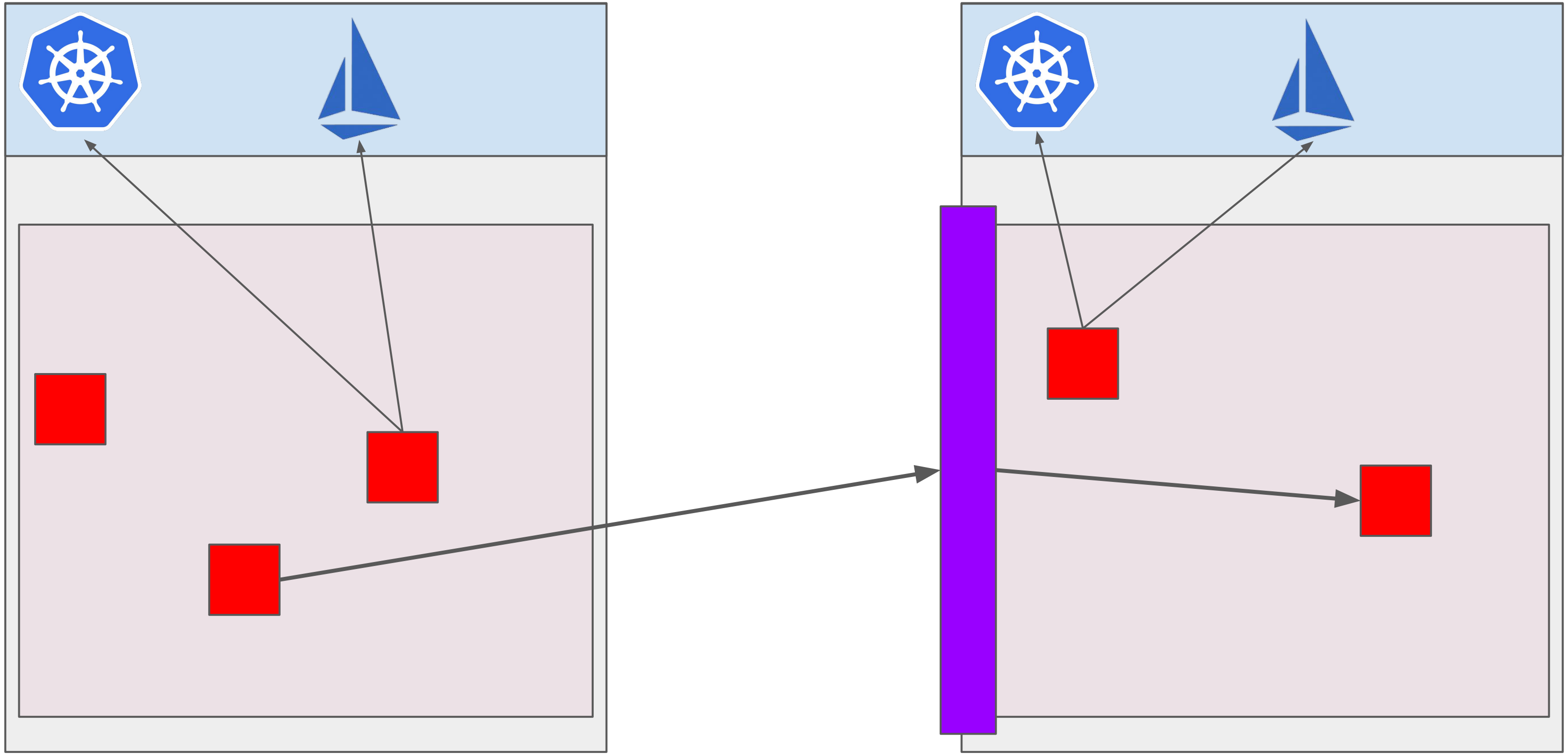
Inter-Cluster Calls



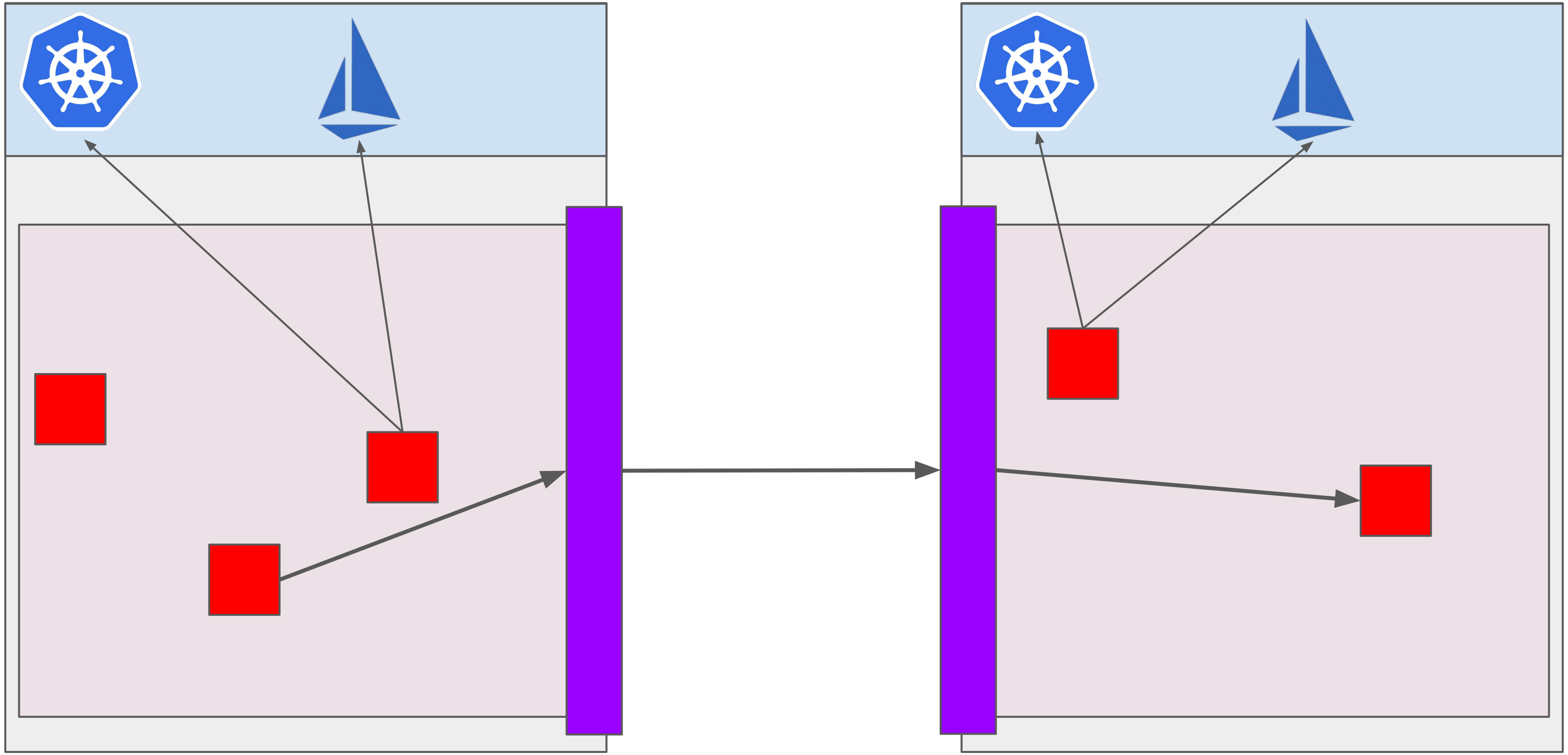
One Istio



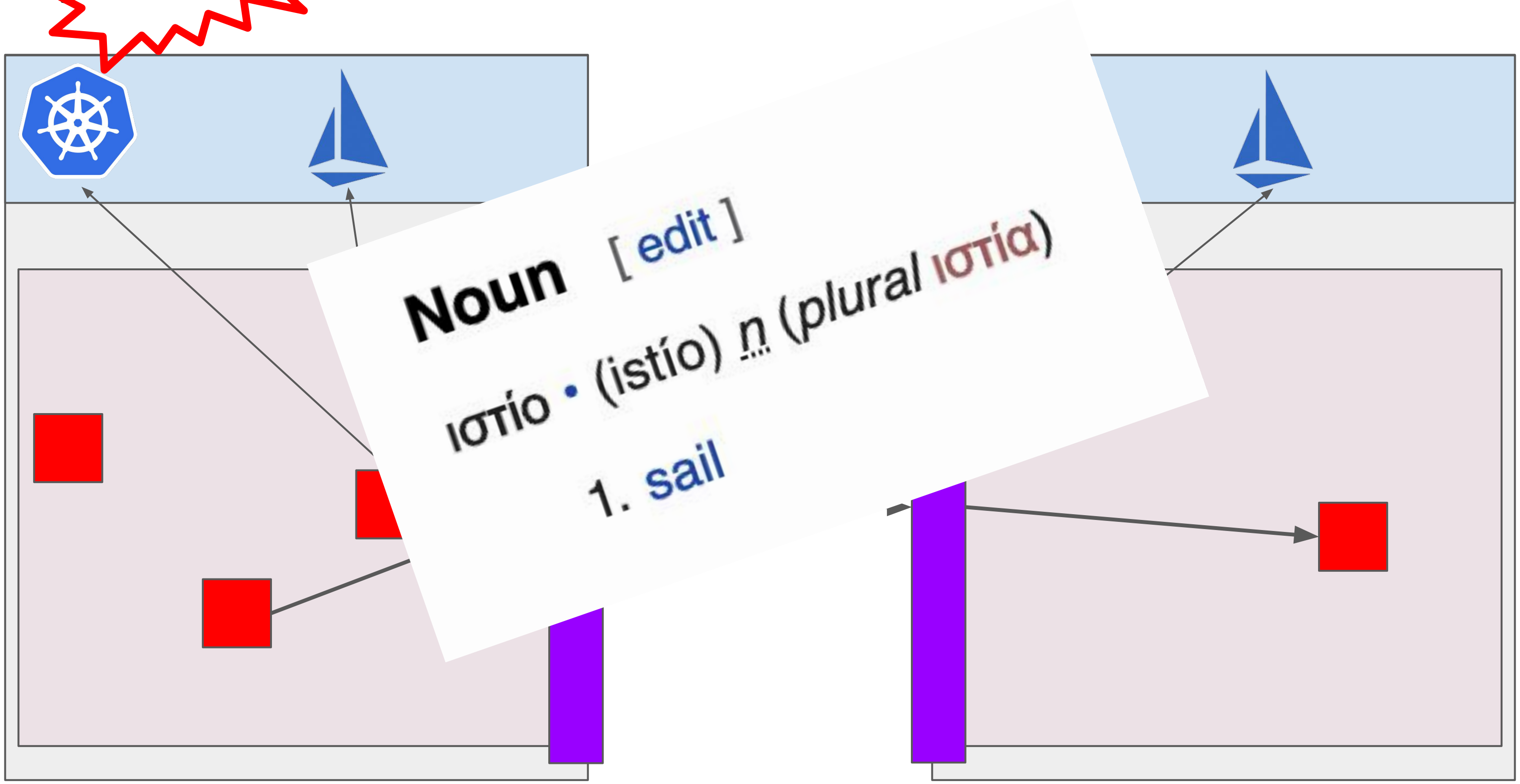
Many Istia



Many Istia



Many Istia



Noun [edit]

ιστίο • (istío) n (plural ιστία)

1. sail

Inter-**Mesh** Calls

We need:

- Service Discovery
- To actually move the traffic

Istio Resources Recap

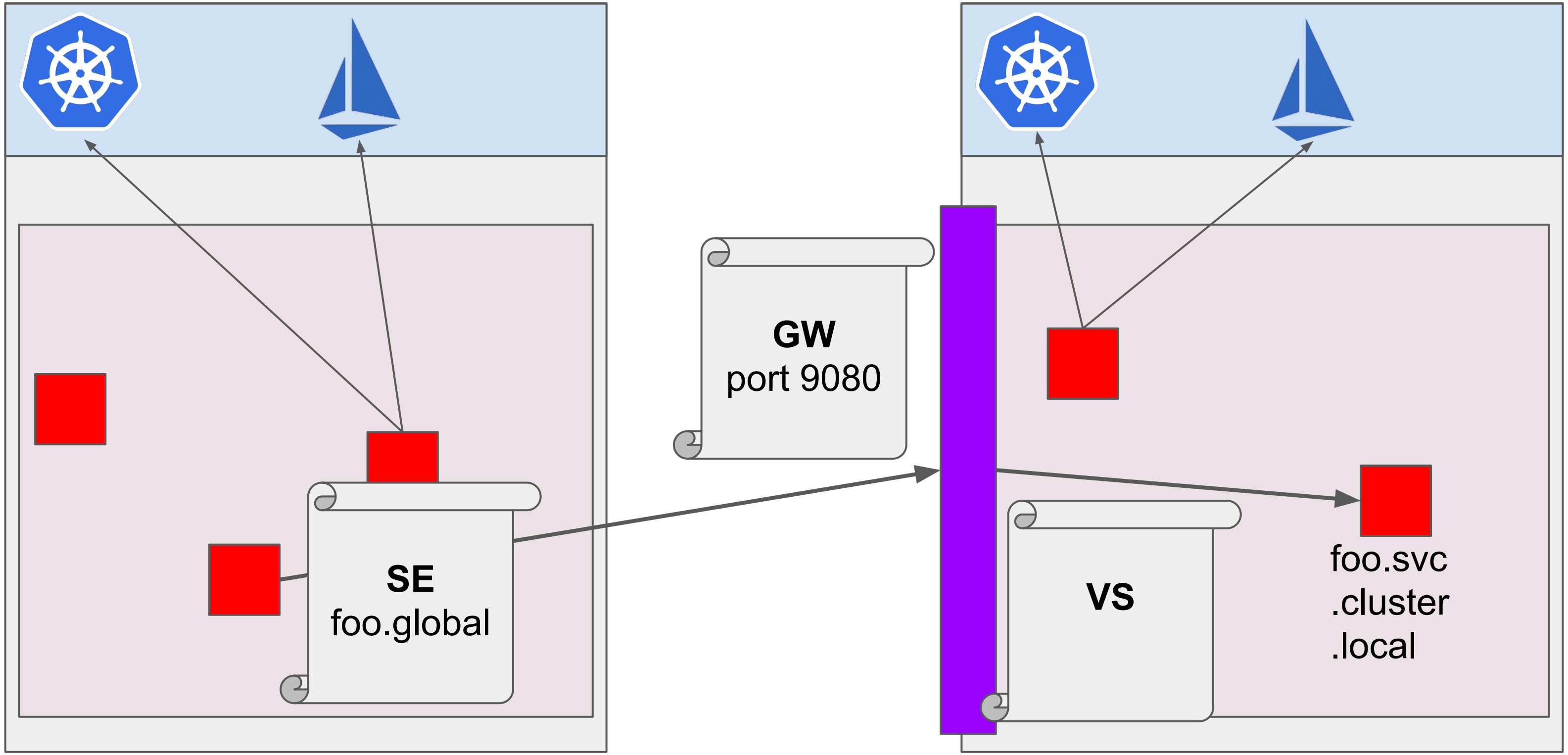
Gateway: “listen on this port.”

ServiceEntry: “what names exist?”

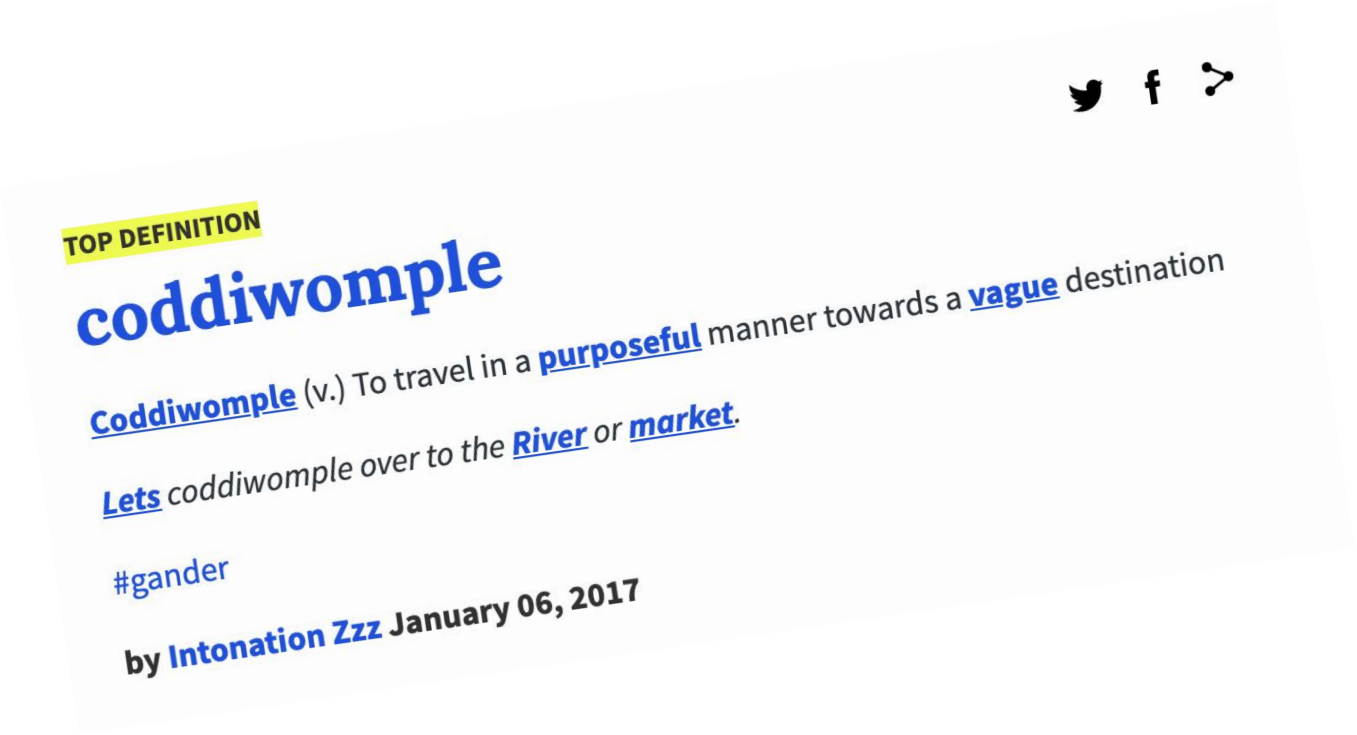
VirtualService: “for a name, how do I route to backends?”

DestinationRule: “how do I talk to those backends?”

Inter-Mesh with Istio - the hard way



A quick coddiwomple to dictionary corner



TOP DEFINITION

coddiwomple

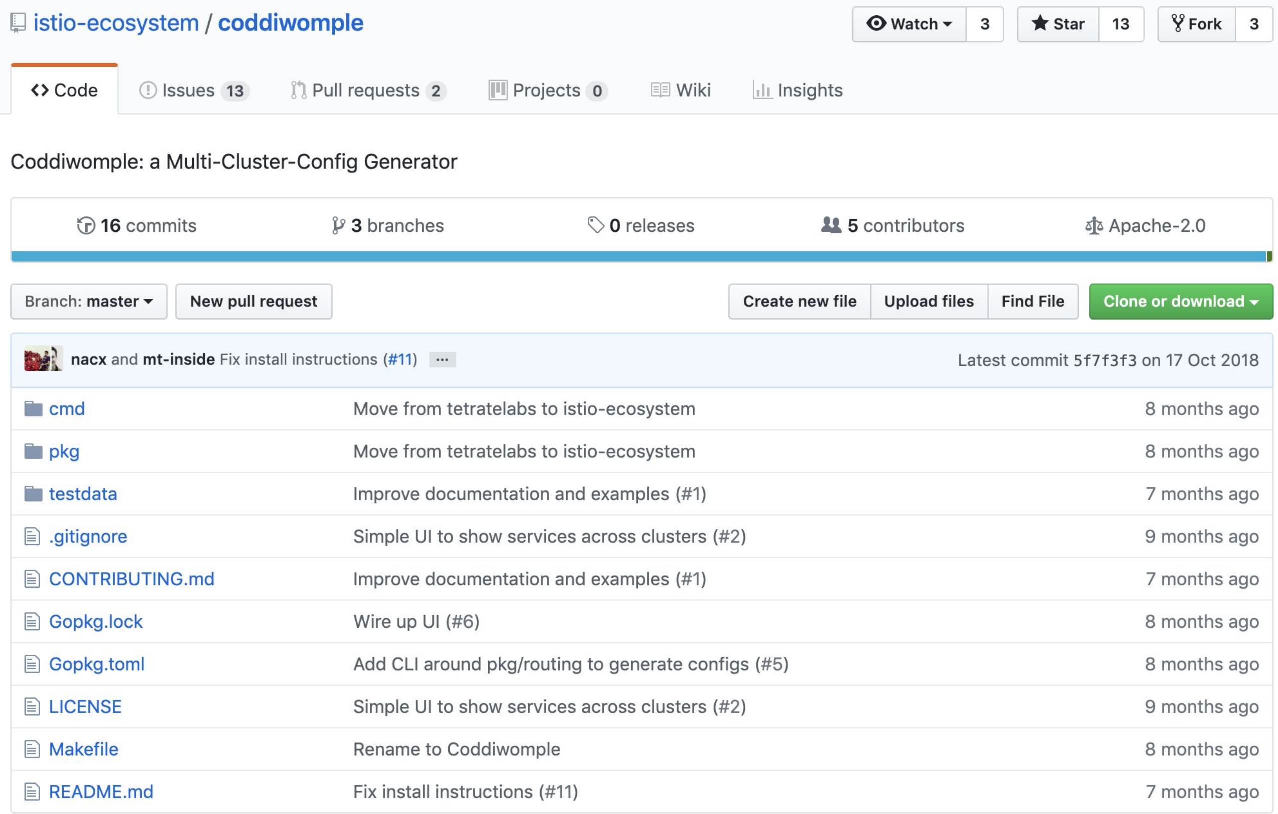
Coddiwomple (v.) To travel in a purposeful manner towards a vague destination

Lets coddiwomple over to the River or market.

#gander

by Intonation Zzz January 06, 2017

Inter-Mesh with Istio - the hard way



Coddiwomple: a Multi-Cluster-Config Generator

16 commits · 3 branches · 0 releases · 5 contributors · Apache-2.0

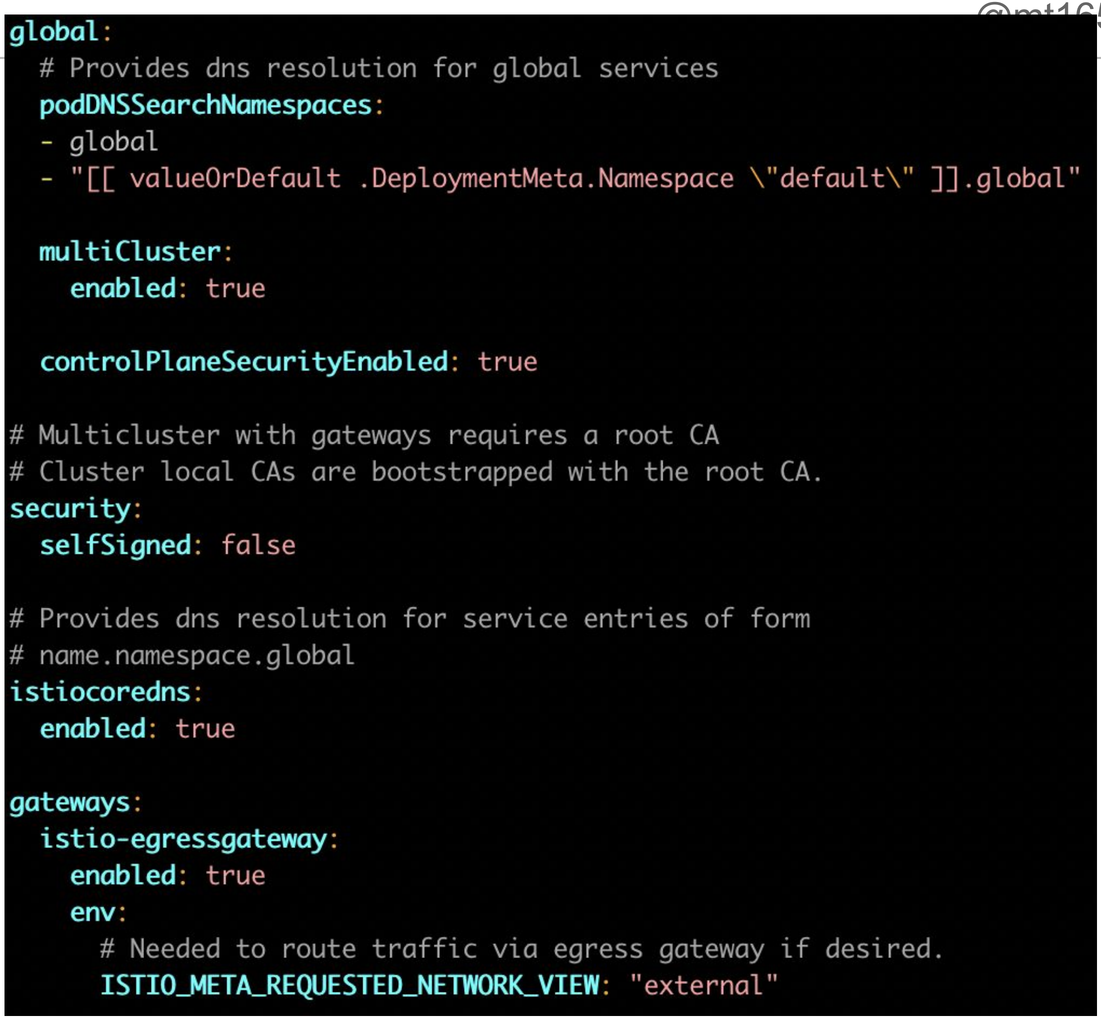
nacx and mt-inside Fix install instructions (#11) ··· Latest commit 5f7f3f3 on 17 Oct 2018

cmd	Move from tetratelabs to istio-ecosystem	8 months ago
pkg	Move from tetratelabs to istio-ecosystem	8 months ago
testdata	Improve documentation and examples (#1)	7 months ago
.gitignore	Simple UI to show services across clusters (#2)	9 months ago
CONTRIBUTING.md	Improve documentation and examples (#1)	7 months ago
Gopkg.lock	Wire up UI (#6)	8 months ago
Gopkg.toml	Add CLI around pkg/routing to generate configs (#5)	8 months ago
LICENSE	Simple UI to show services across clusters (#2)	9 months ago
Makefile	Rename to Coddiwomple	8 months ago
README.md	Fix install instructions (#11)	7 months ago

Enter Istio 1.1

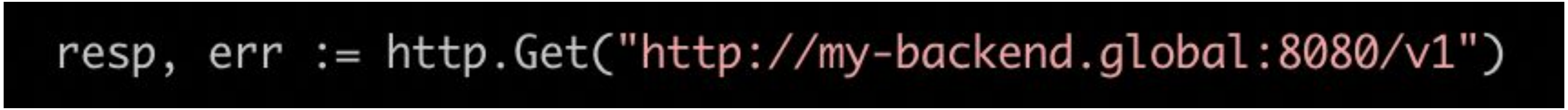
- Reconfigure the mesh at install time
- Apply (fewer) YAMLs to make it do inter-mesh routing
- Change your services slightly

Install-Time Config

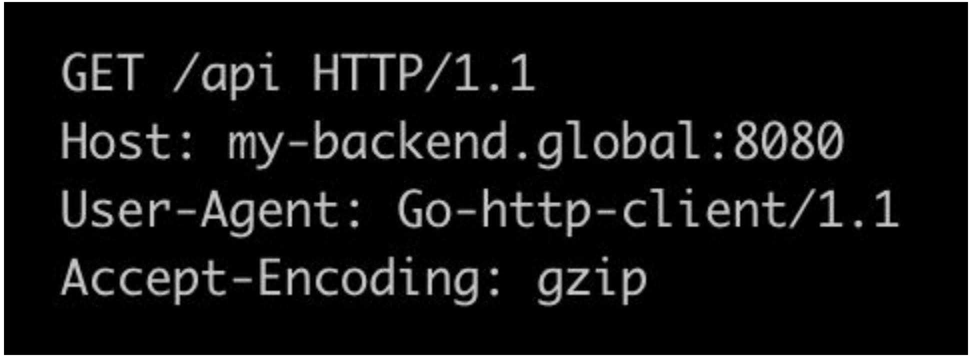


```yaml
global:
  # Provides dns resolution for global services
  podDNSSearchNamespaces:
  - global
  - "[[ valueOrDefault .DeploymentMeta.Namespace \"default\" ]].global"

  multiCluster:
    enabled: true

  controlPlaneSecurityEnabled: true

# Multicluster with gateways requires a root CA
# Cluster local CAs are bootstrapped with the root CA.
security:
  selfSigned: false

# Provides dns resolution for service entries of form
# name.namespace.global
istiocoredns:
  enabled: true

gateways:
  istio-egressgateway:
    enabled: true
    env:
      # Needed to route traffic via egress gateway if desired.
      ISTIO_META_REQUESTED_NETWORK_VIEW: "external"
```

Service Discovery

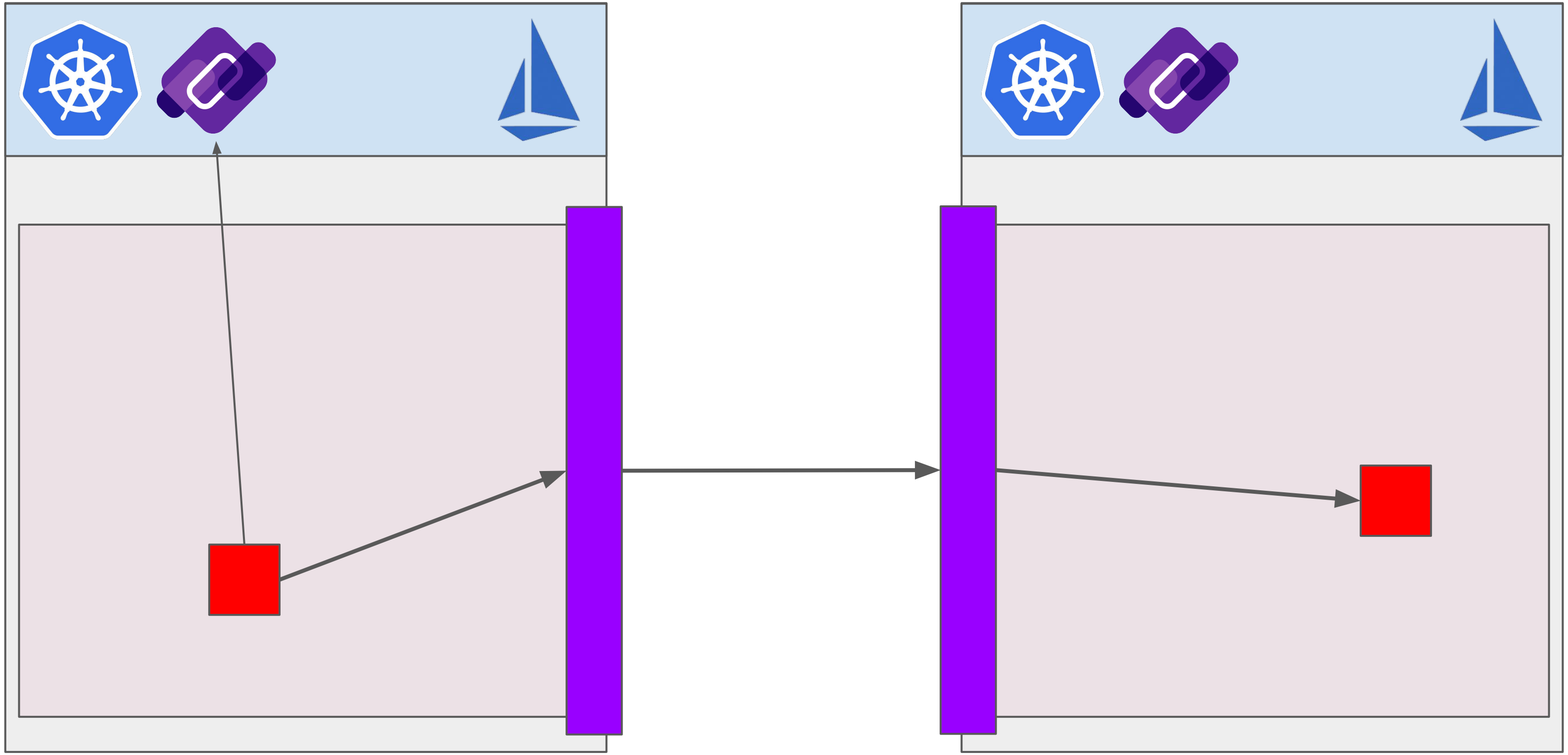


```
resp, err := http.Get("http://my-backend.global:8080/v1")
```

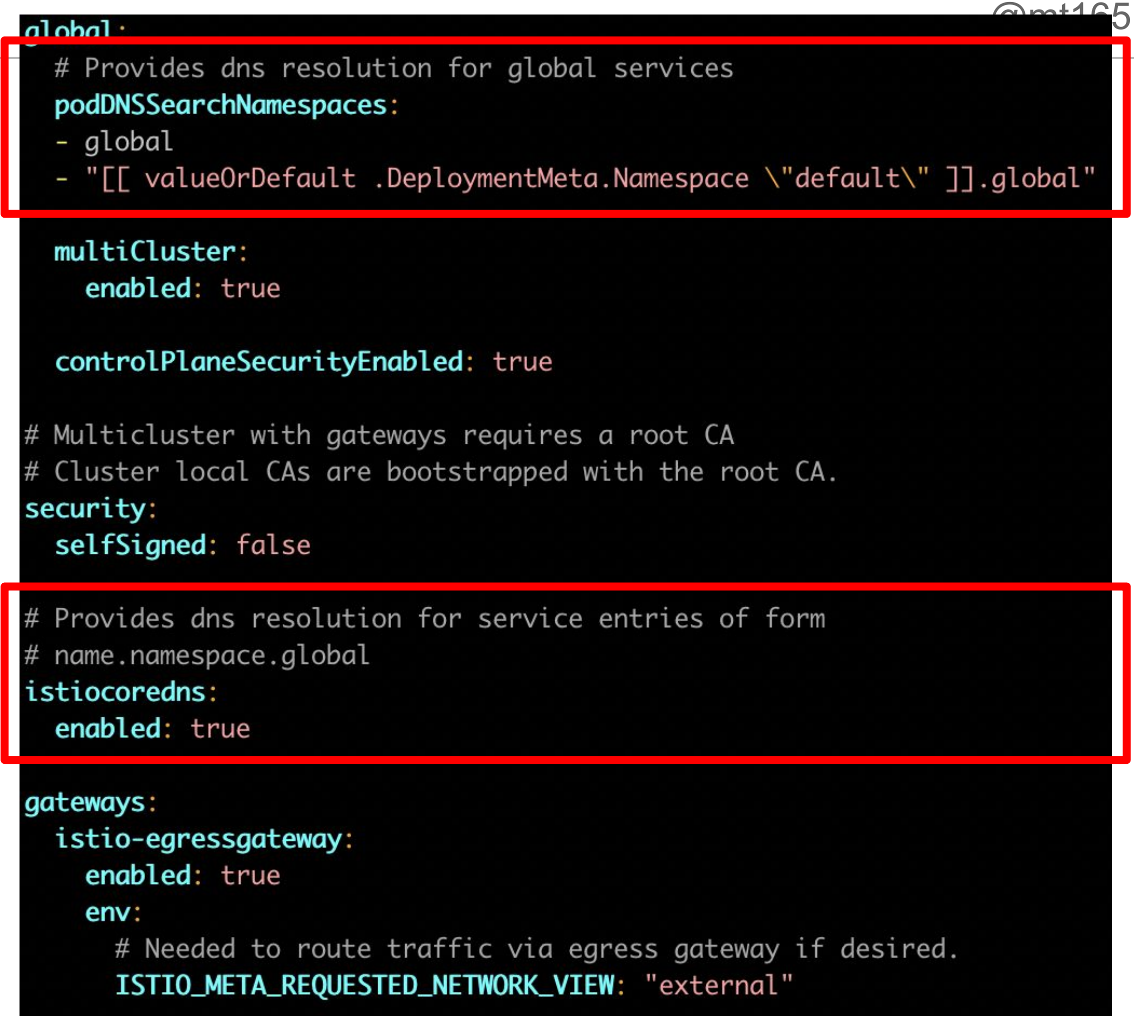


```
GET /api HTTP/1.1
Host: my-backend.global:8080
User-Agent: Go-http-client/1.1
Accept-Encoding: gzip
```

Service Discovery

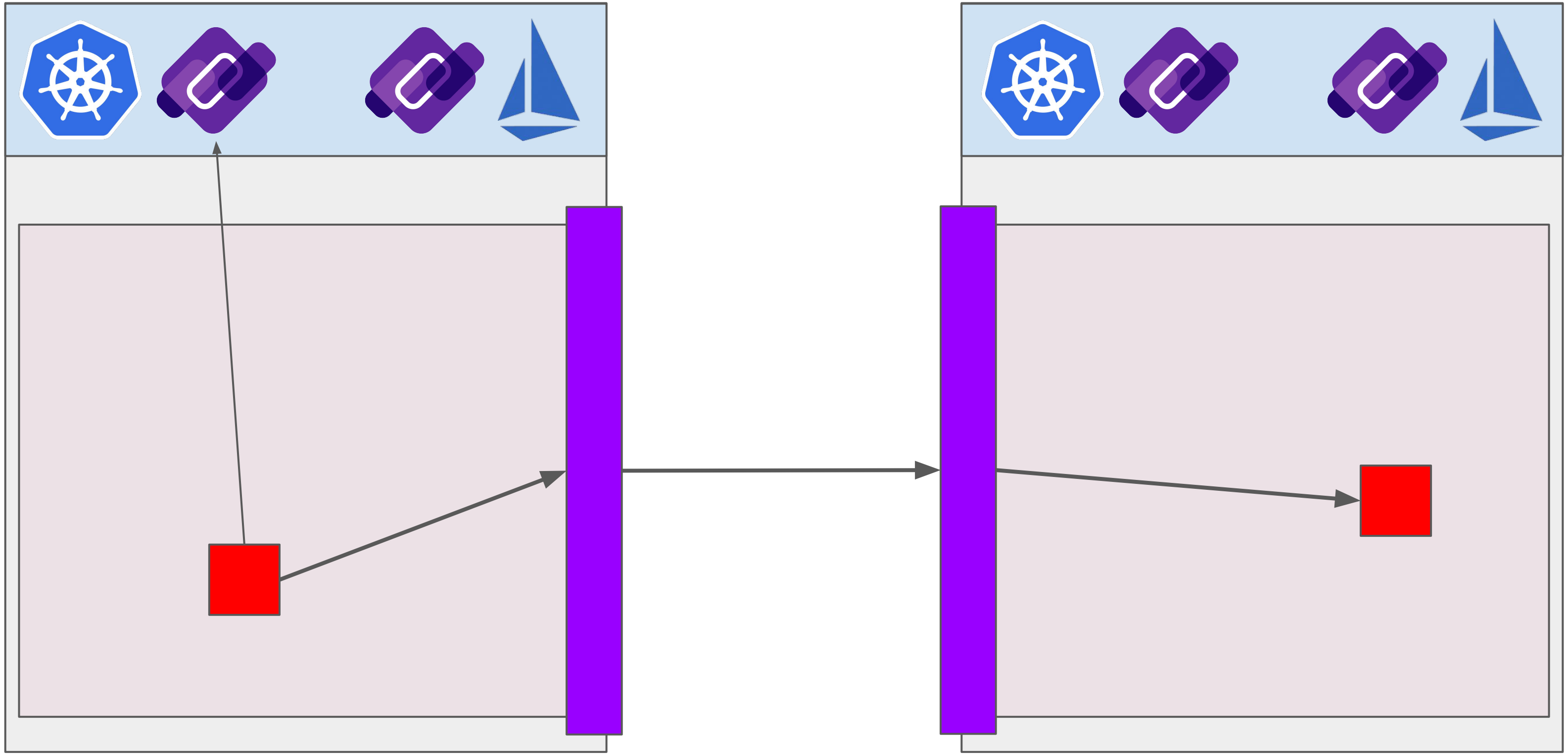


Service Discovery

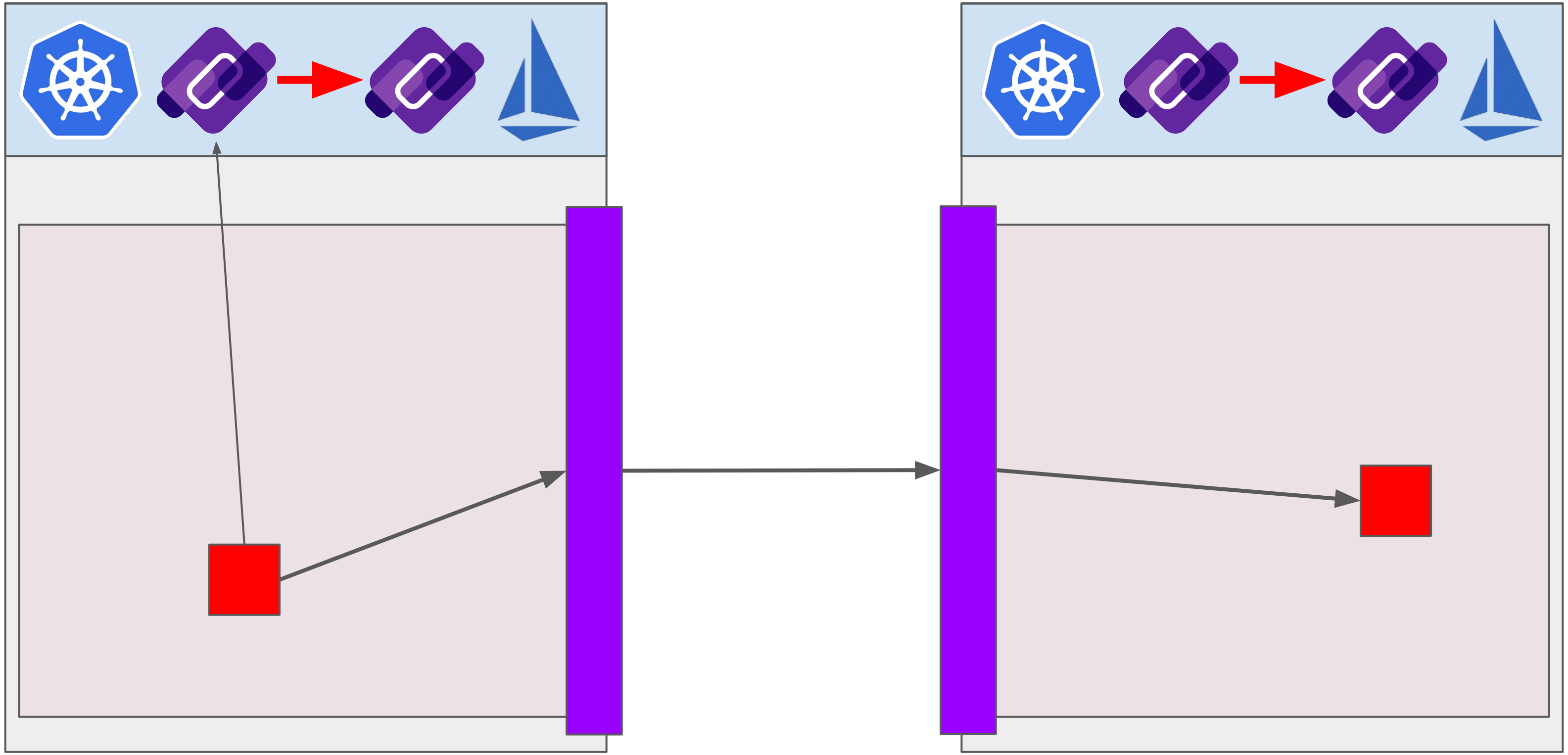


```
global:
  # Provides dns resolution for global services
  podDNSSearchNamespaces:
  - global
  - "[[ valueOrDefault .DeploymentMeta.Namespace \"default\" ]].global"

  multiCluster:
    enabled: true

  controlPlaneSecurityEnabled: true

# Multicluster with gateways requires a root CA
# Cluster local CAs are bootstrapped with the root CA.
security:
  selfSigned: false

# Provides dns resolution for service entries of form
# name.namespace.global
istiocoredns:
  enabled: true

gateways:
  istio-egressgateway:
    enabled: true
    env:
      # Needed to route traffic via egress gateway if desired.
      ISTIO_META_REQUESTED_NETWORK_VIEW: "external"
```

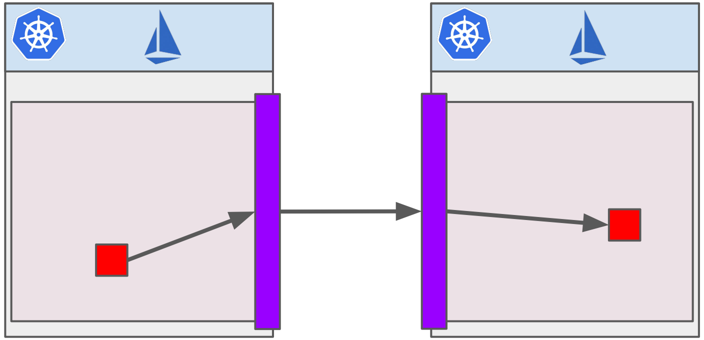
Service Discovery

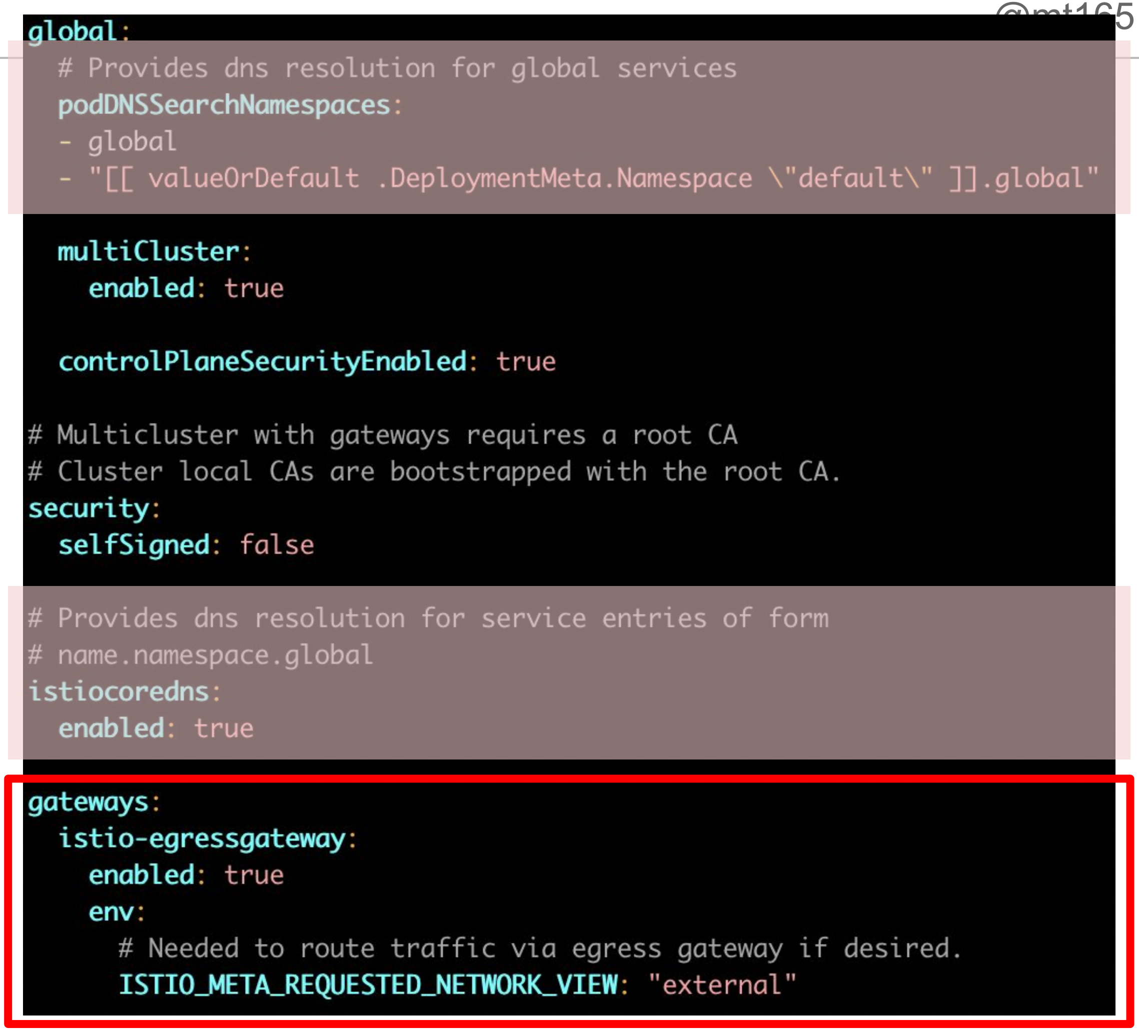


Service Discovery



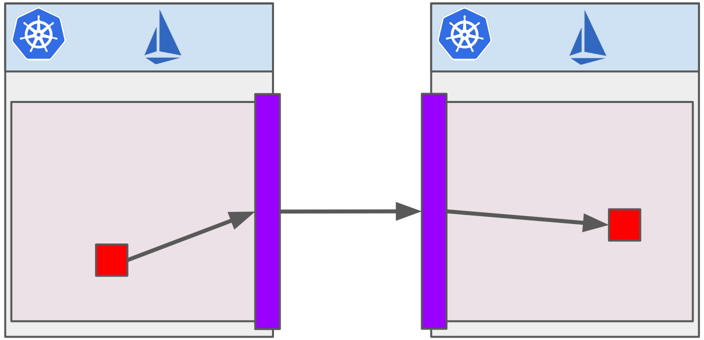
Moving the Traffic

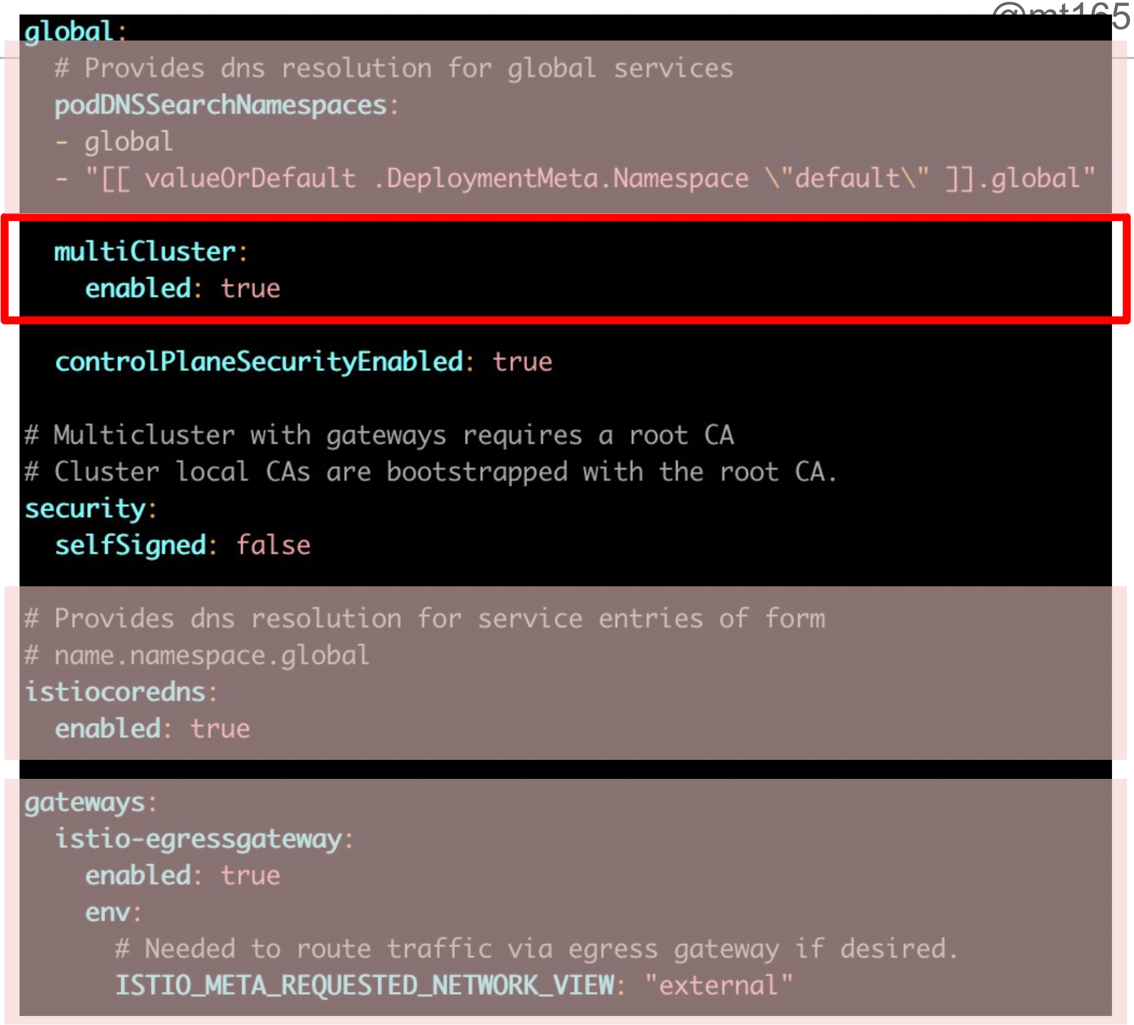




```
global:
  # Provides dns resolution for global services
  podDNSSearchNamespaces:
  - global
  - "[[ valueOrDefault .DeploymentMeta.Namespace \"default\" ]].global"

  multiCluster:
    enabled: true

  controlPlaneSecurityEnabled: true

# Multicluster with gateways requires a root CA
# Cluster local CAs are bootstrapped with the root CA.
security:
  selfSigned: false

# Provides dns resolution for service entries of form
# name.namespace.global
istiocoredns:
  enabled: true

gateways:
  istio-egressgateway:
    enabled: true
    env:
      # Needed to route traffic via egress gateway if desired.
      ISTIO_META_REQUESTED_NETWORK_VIEW: "external"
```

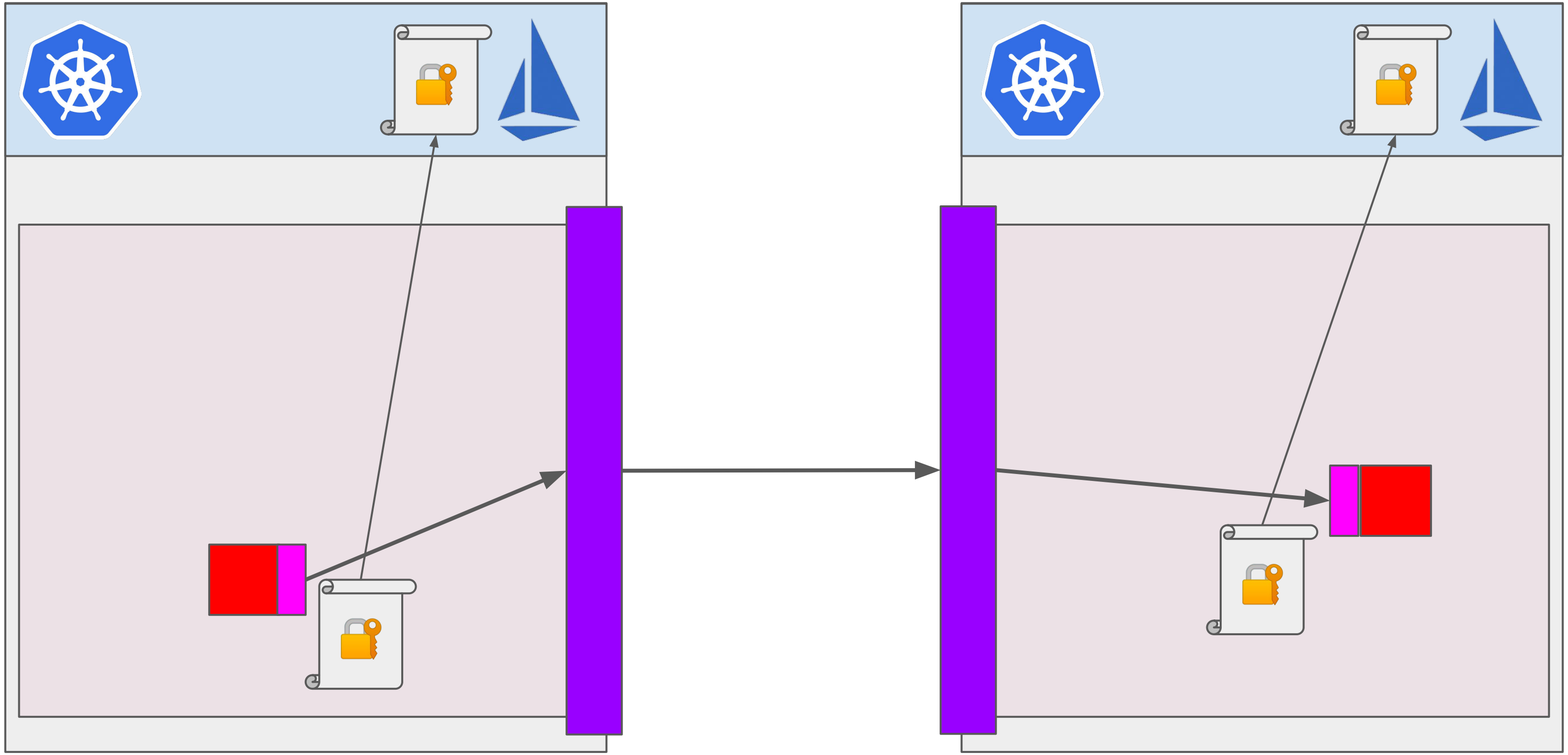
Moving the Traffic



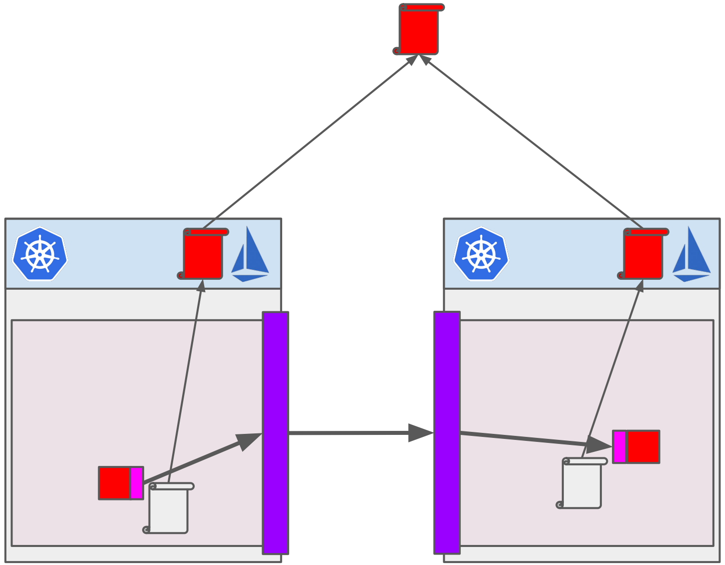


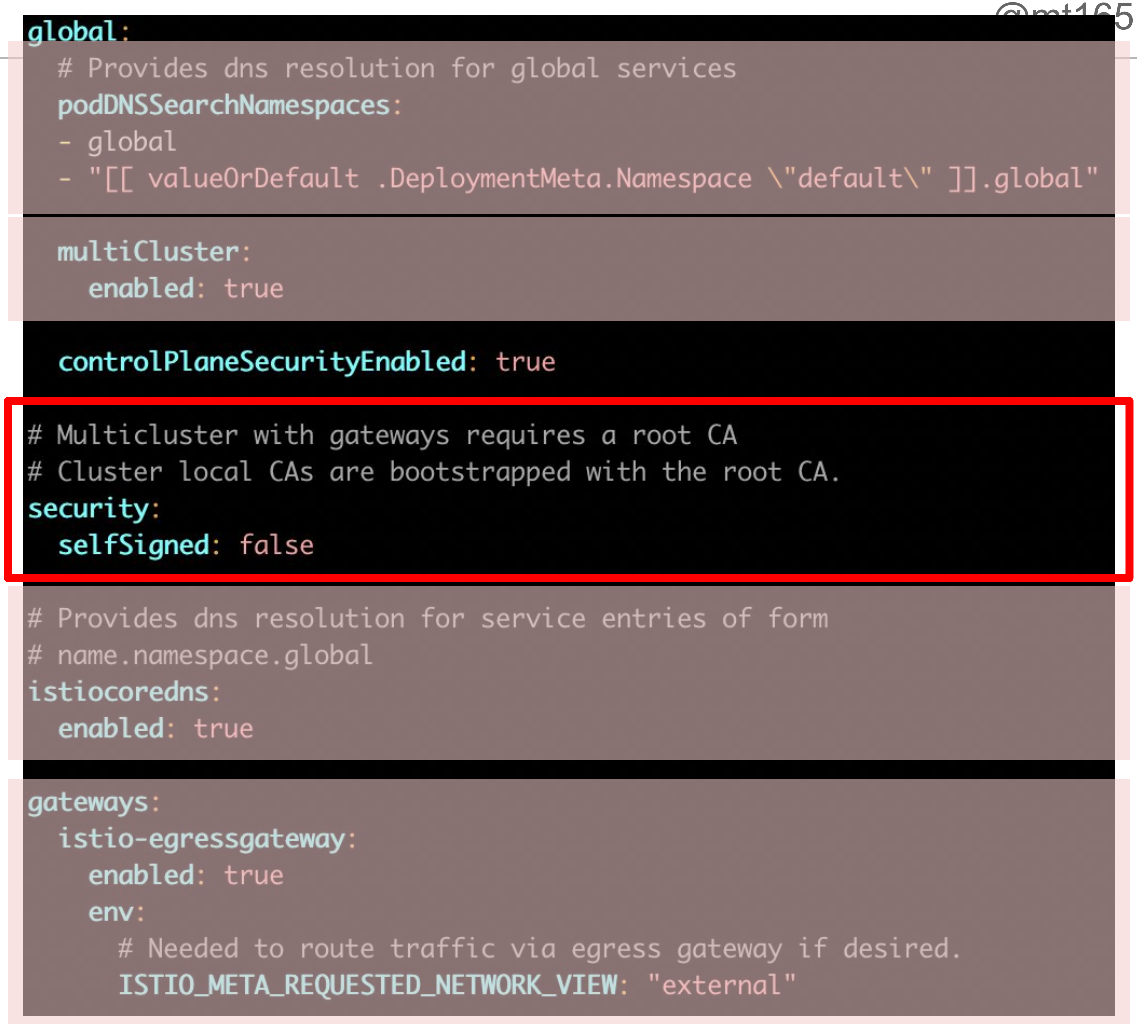
```
global:
  # Provides dns resolution for global services
  podDNSSearchNamespaces:
  - global
  - "[[ valueOrDefault .DeploymentMeta.Namespace \"default\" ]].global"

  multiCluster:
    enabled: true

  controlPlaneSecurityEnabled: true

# Multicluster with gateways requires a root CA
# Cluster local CAs are bootstrapped with the root CA.
security:
  selfSigned: false

# Provides dns resolution for service entries of form
# name.namespace.global
istiocoredns:
  enabled: true

gateways:
  istio-egressgateway:
    enabled: true
    env:
      # Needed to route traffic via egress gateway if desired.
      ISTIO_META_REQUESTED_NETWORK_VIEW: "external"
```

Moving the Traffic



Moving the Traffic





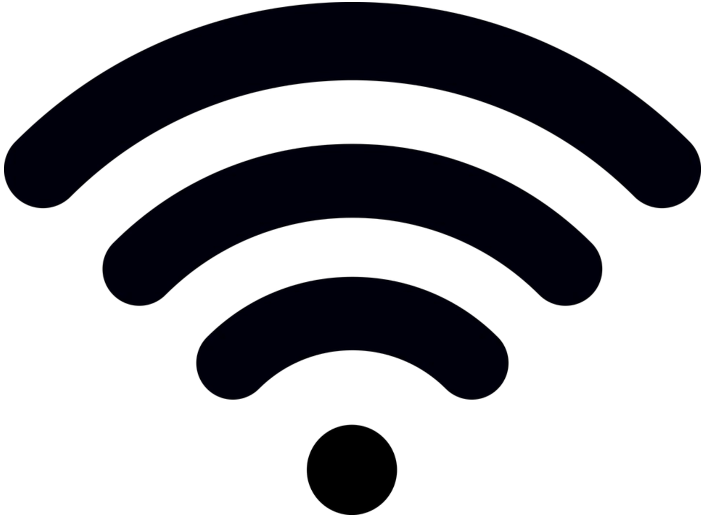
```
global:
  # Provides dns resolution for global services
  podDNSSearchNamespaces:
  - global
  - "[[ valueOrDefault .DeploymentMeta.Namespace \"default\" ]].global"

  multiCluster:
    enabled: true

  controlPlaneSecurityEnabled: true

# Multicluster with gateways requires a root CA
# Cluster local CAs are bootstrapped with the root CA.
security:
  selfSigned: false

# Provides dns resolution for service entries of form
# name.namespace.global
istiocoredns:
  enabled: true

gateways:
  istio-egressgateway:
    enabled: true
    env:
      # Needed to route traffic via egress gateway if desired.
      ISTIO_META_REQUESTED_NETWORK_VIEW: "external"
```

Demo Time!

Pray to the demo gods...

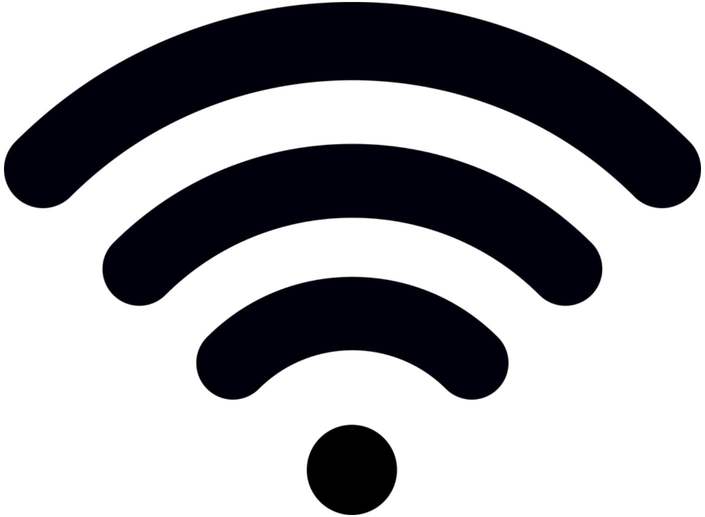
Pray to the demo gods...

The old...



Pray to the demo gods...

The old...



...and the new

Recap

- Many options for getting traffic between Clusters
- Service Mesh is the best choice
- Proper architecture is one mesh per cluster
- With Istio 1.1 it’s quite easy to make this work

Thanks!

@mt165

Slides
Videos
Demo code

mt165.co.uk