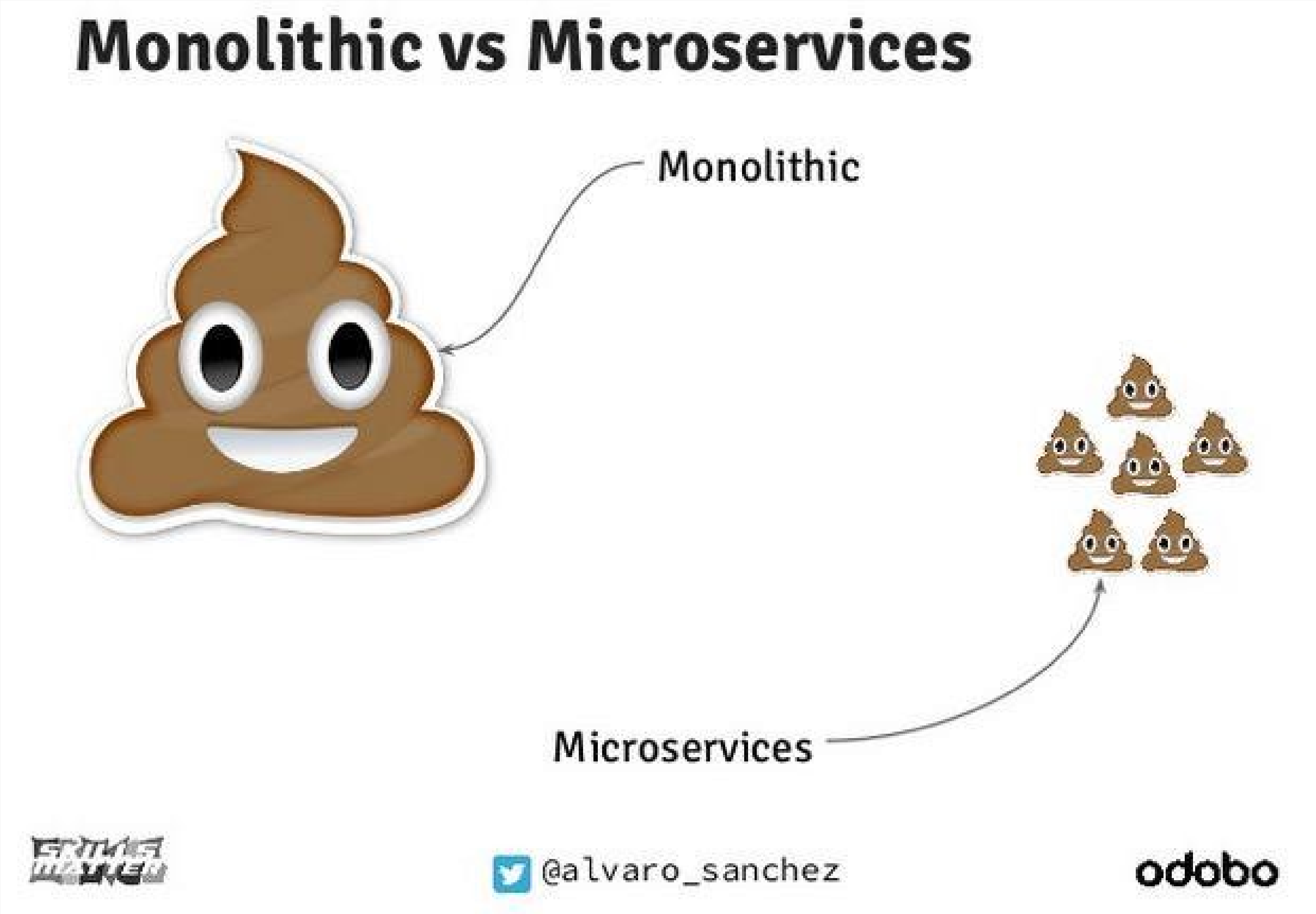
Introduction to Istio on Kubernetes

Jonh Wendell
Software Engineer, Red Hat
DevConf.US, Aug 2018

Microservices?



Monolithic vs Microservices

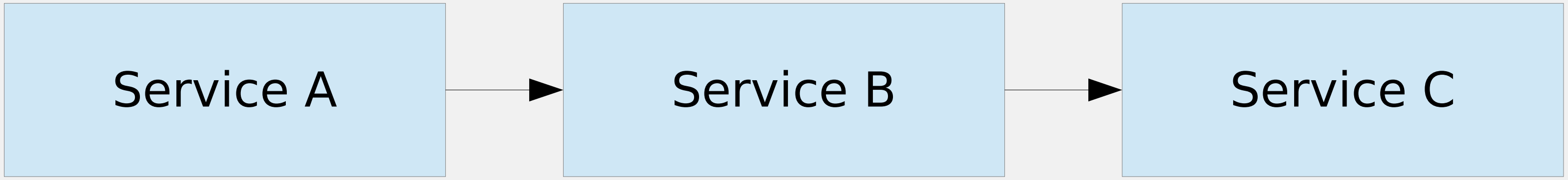
Monolithic

Microservices

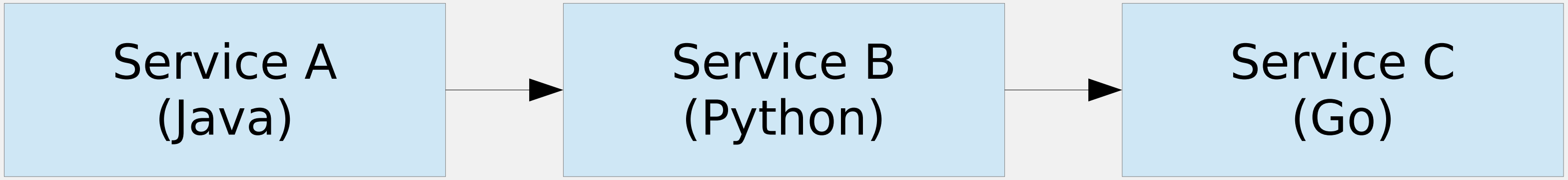
@alvaro_sanchez

odobo

Microservices



Microservices



Service A
(Java)

Service B
(Python)

Service C
(Go)

Distributed Systems

- Deployment
- Resiliency
- Networking
- Security

kubernetes

To the rescue

kubernetes

- Deployment platform
- DNS based service discovery
- Simple L3/4 load balancing

- Not so adequate for service-to-service communication

Services have to deal with

- Client Side Load Balancing
- Fault tolerance – Timeouts and Retries
- Observability
- Monitoring & Tracing
- Circuit Breaking

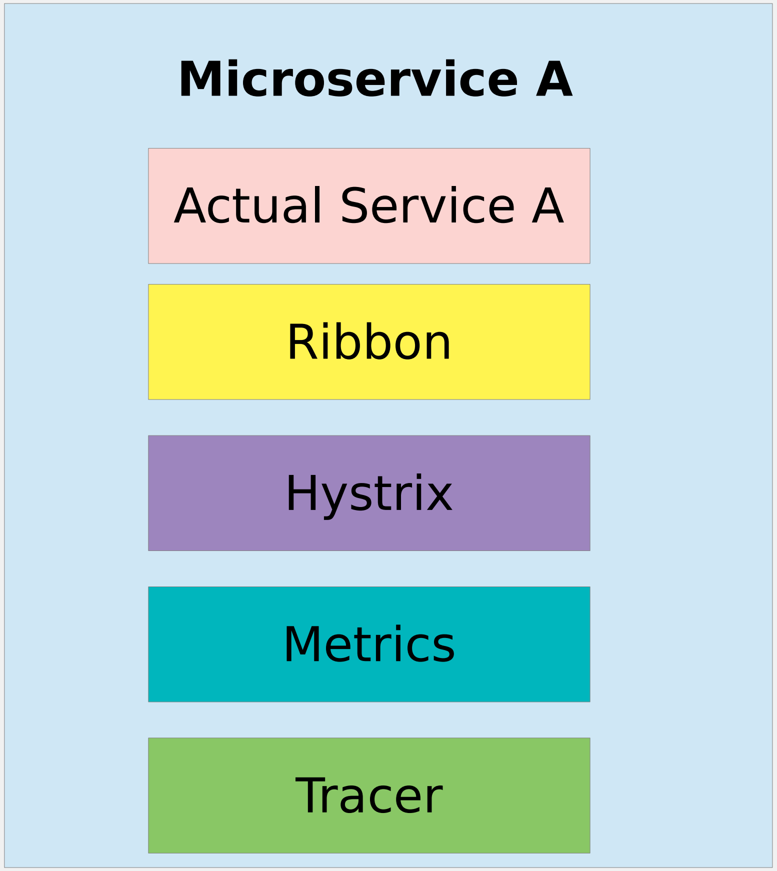
Popular tools

- Eureka – Service Registry
- Ribbon – Client side load balancer
- Hystrix – Circuit Breaker
- Zipkin – Distributed Tracing

finagle

NETFLIX OSS

“Micro” Services?



Microservice A

Actual Service A

Ribbon

Hystrix

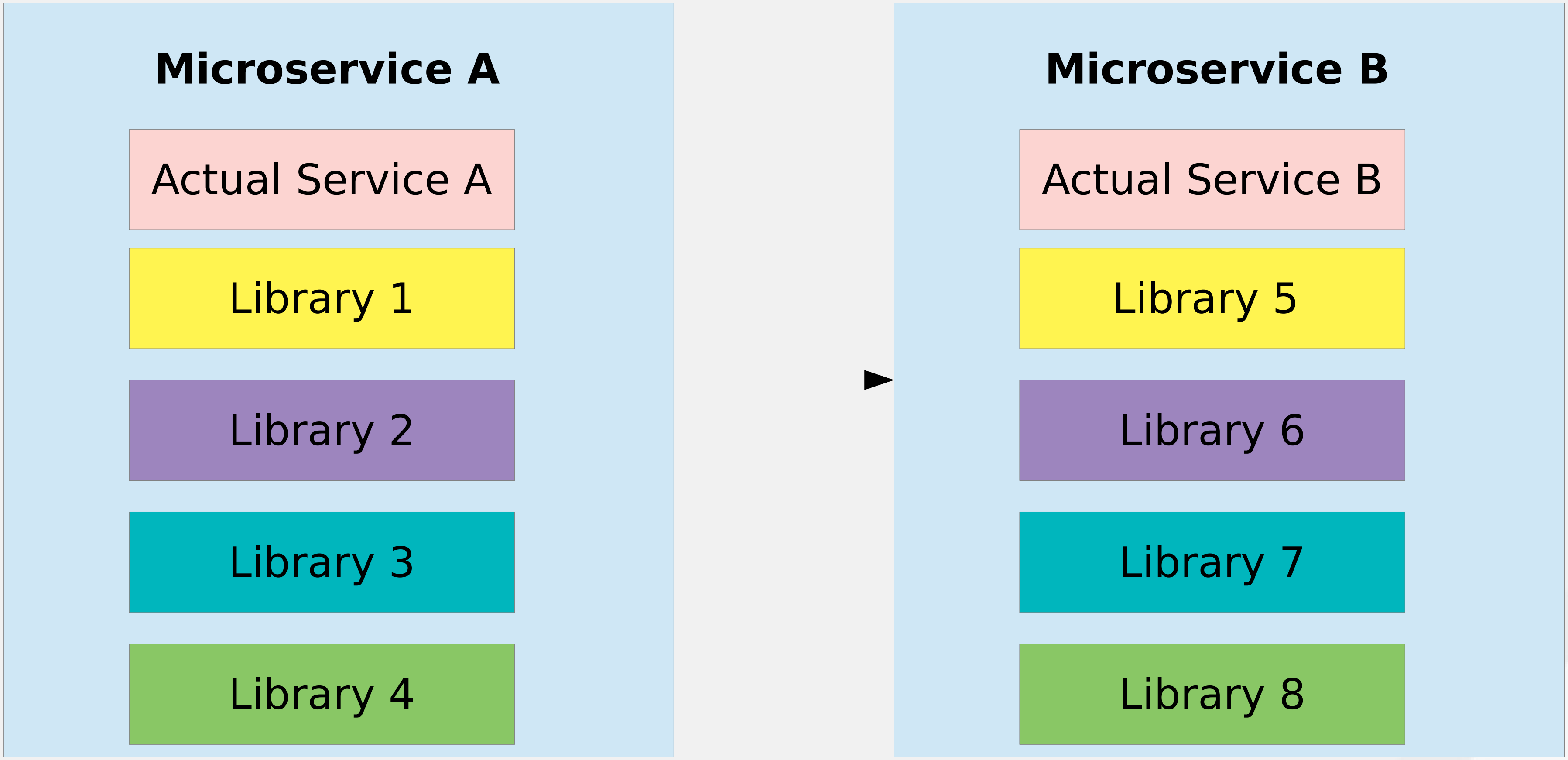
Metrics

Tracer

Library approach? What if...

- You use Spring?
- Vert.x?
- Go?
- Python?
- Ruby?
- Perl?
- I’m kidding, don’t use Perl :)

“Micro” Services?



Microservice A

Actual Service A

Library 1

Library 2

Library 3

Library 4

Microservice B

Actual Service B

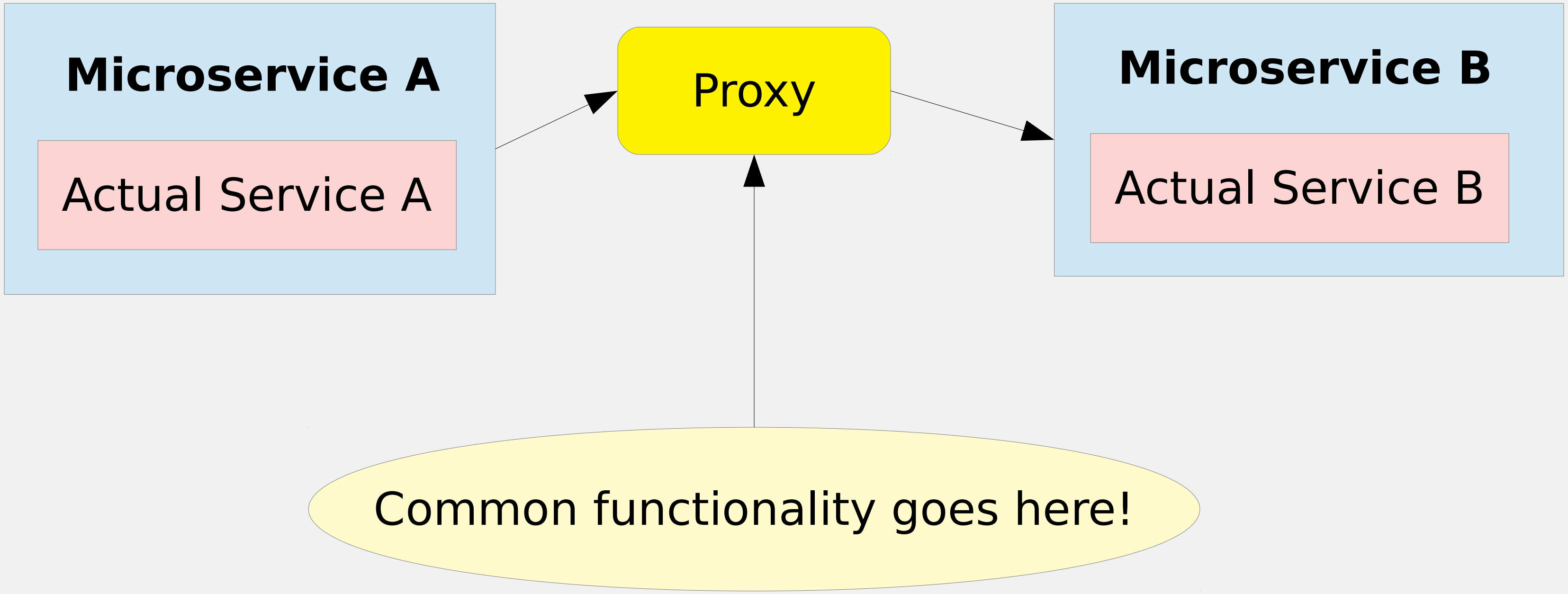
Library 5

Library 6

Library 7

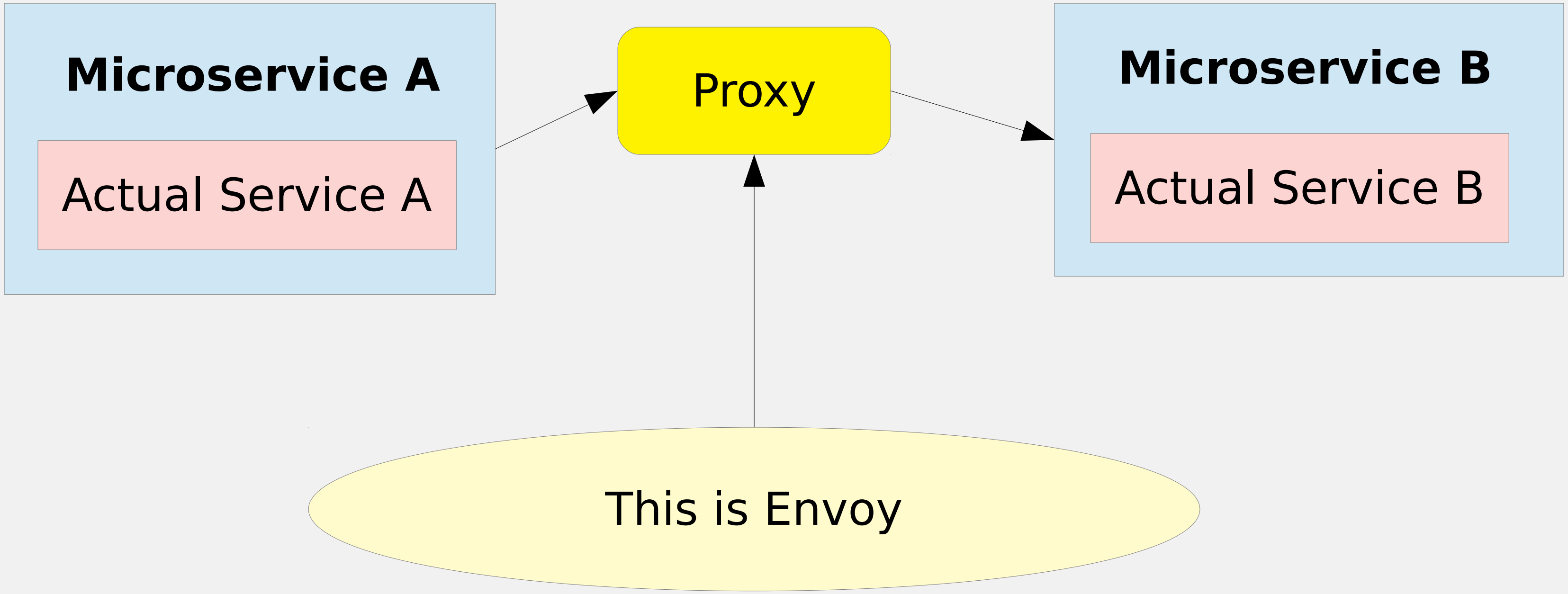
Library 8

Micro Services!



Meet Envoy

envoy

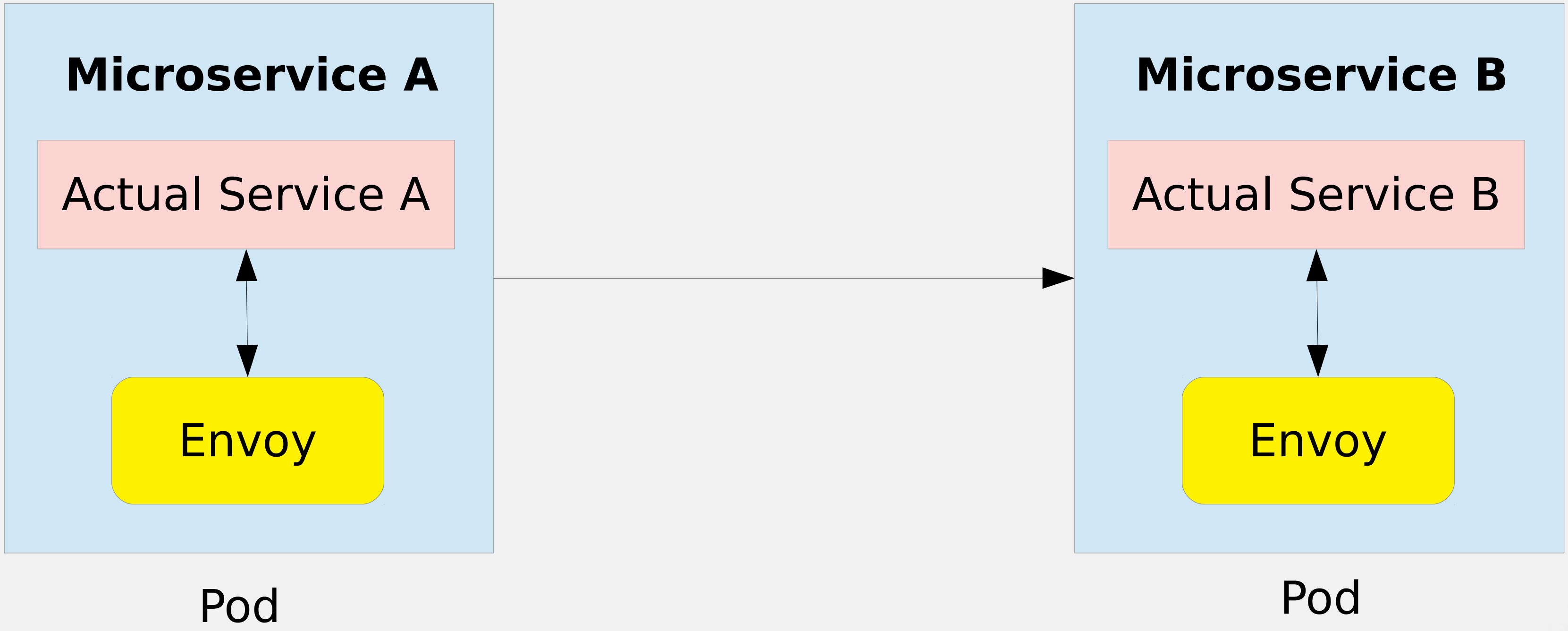


What is Envoy?

- service proxy, developed by Lyft
- written in C++, highly parallel, non-blocking
- L3/4 network filter
- out of the box L7 filters
- HTTP 2, including gRPC
- baked in service discovery/health checking
- advanced load balancing
- stats, metrics, tracing

Sidecar model

A kind of deployment in Kubernetes



Microservice A

Actual Service A

Envoy

Pod

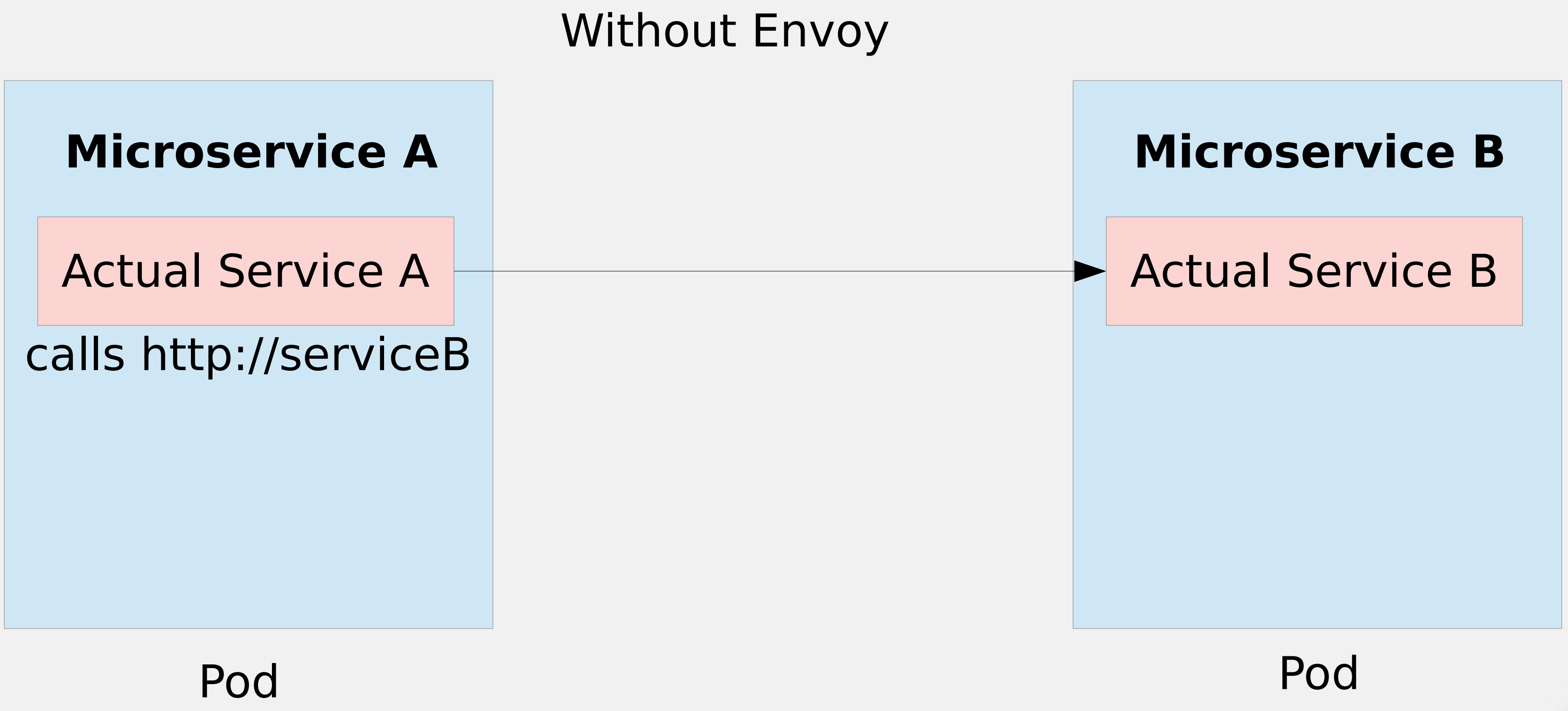
Microservice B

Actual Service B

Envoy

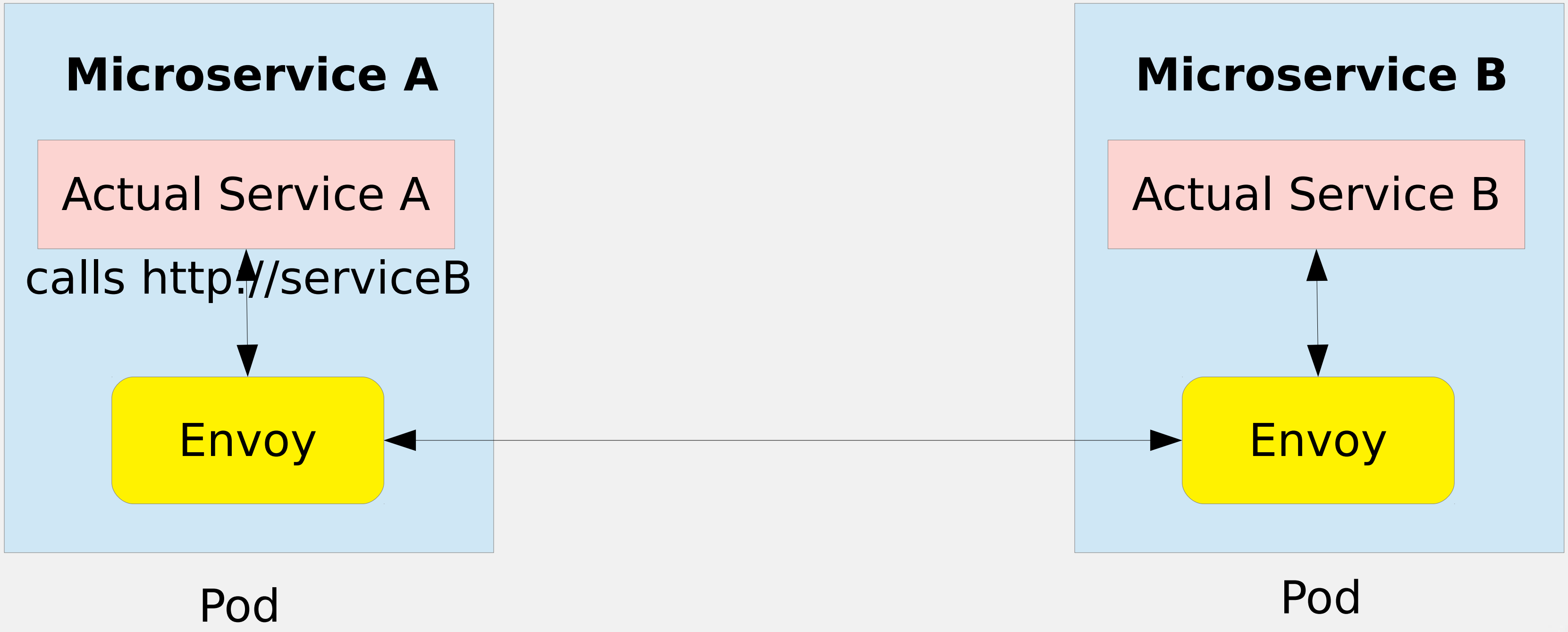
Pod

Service communication



Service communication

With Envoy



Microservice A

Actual Service A

calls http://serviceB

Envoy

Pod

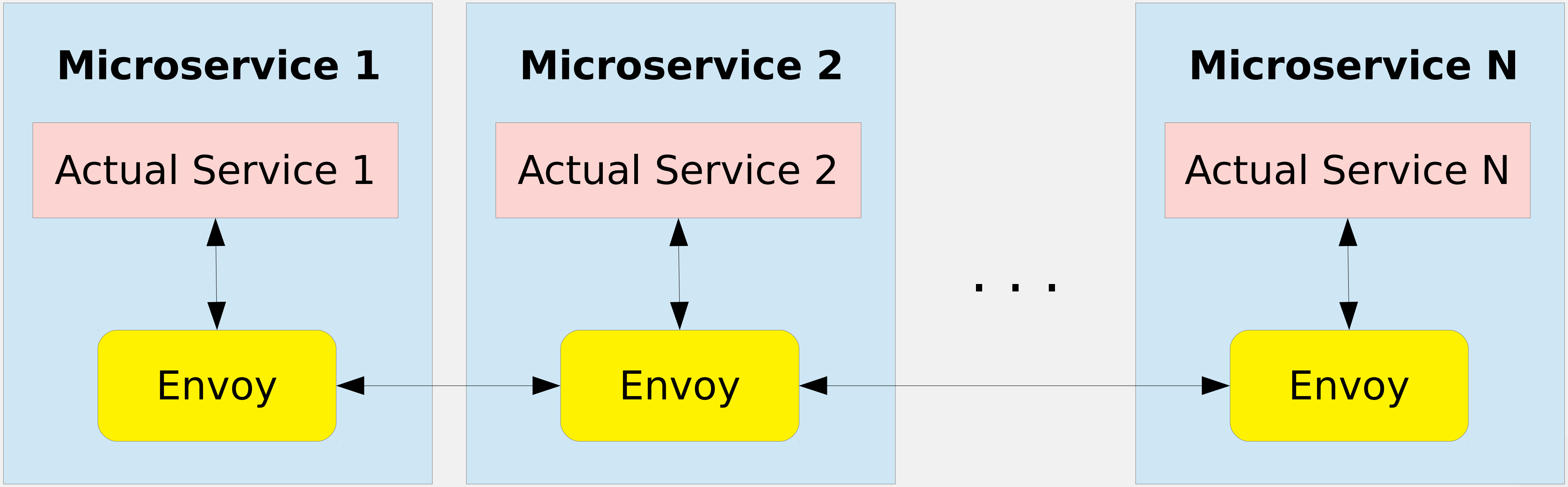
Microservice B

Actual Service B

Envoy

Pod

Service Mesh



Configure a fleet of Envoys can be verbose and error prone without automation.

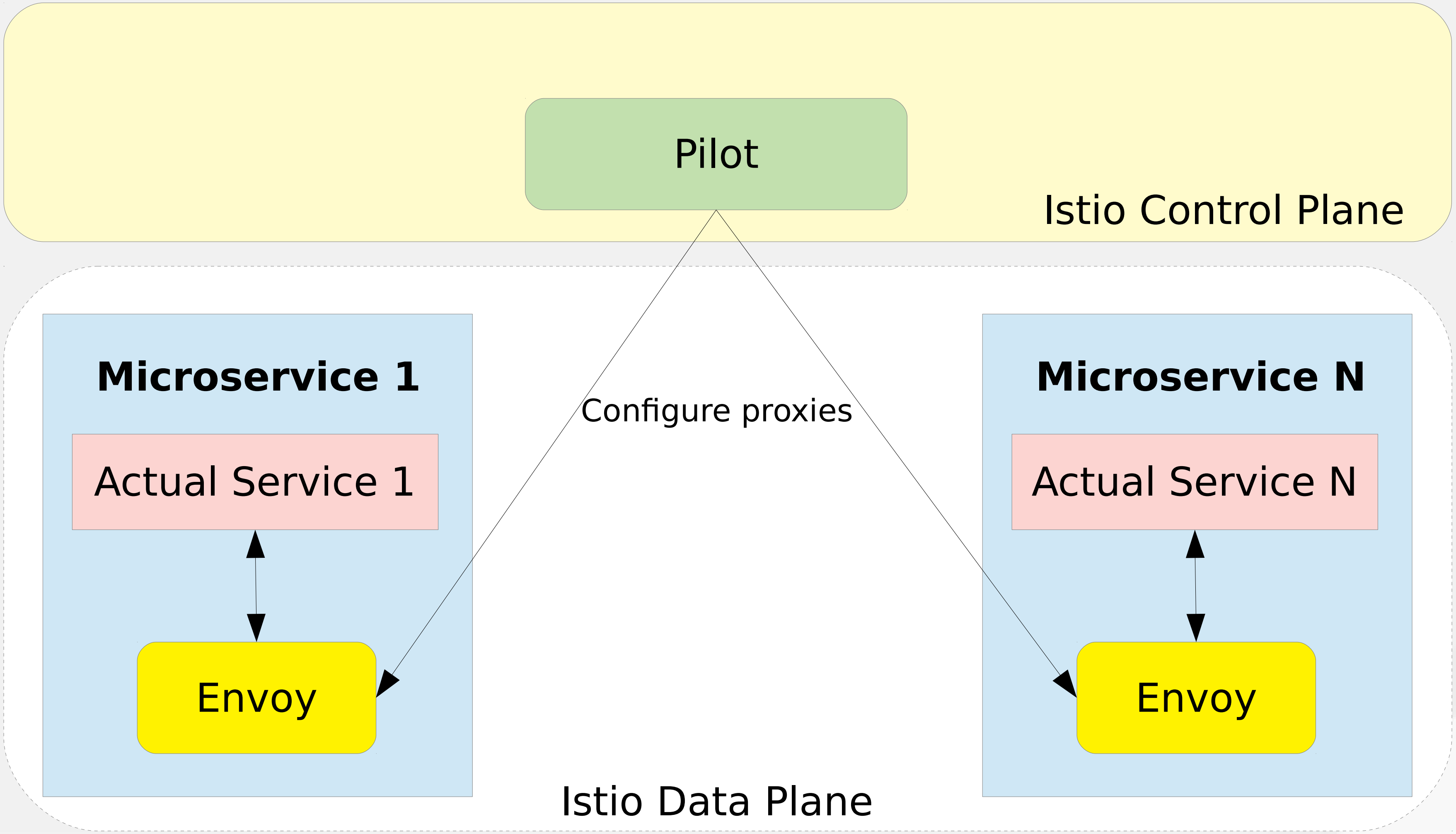
We need a Control Plane.

Meet Istio Service Mesh

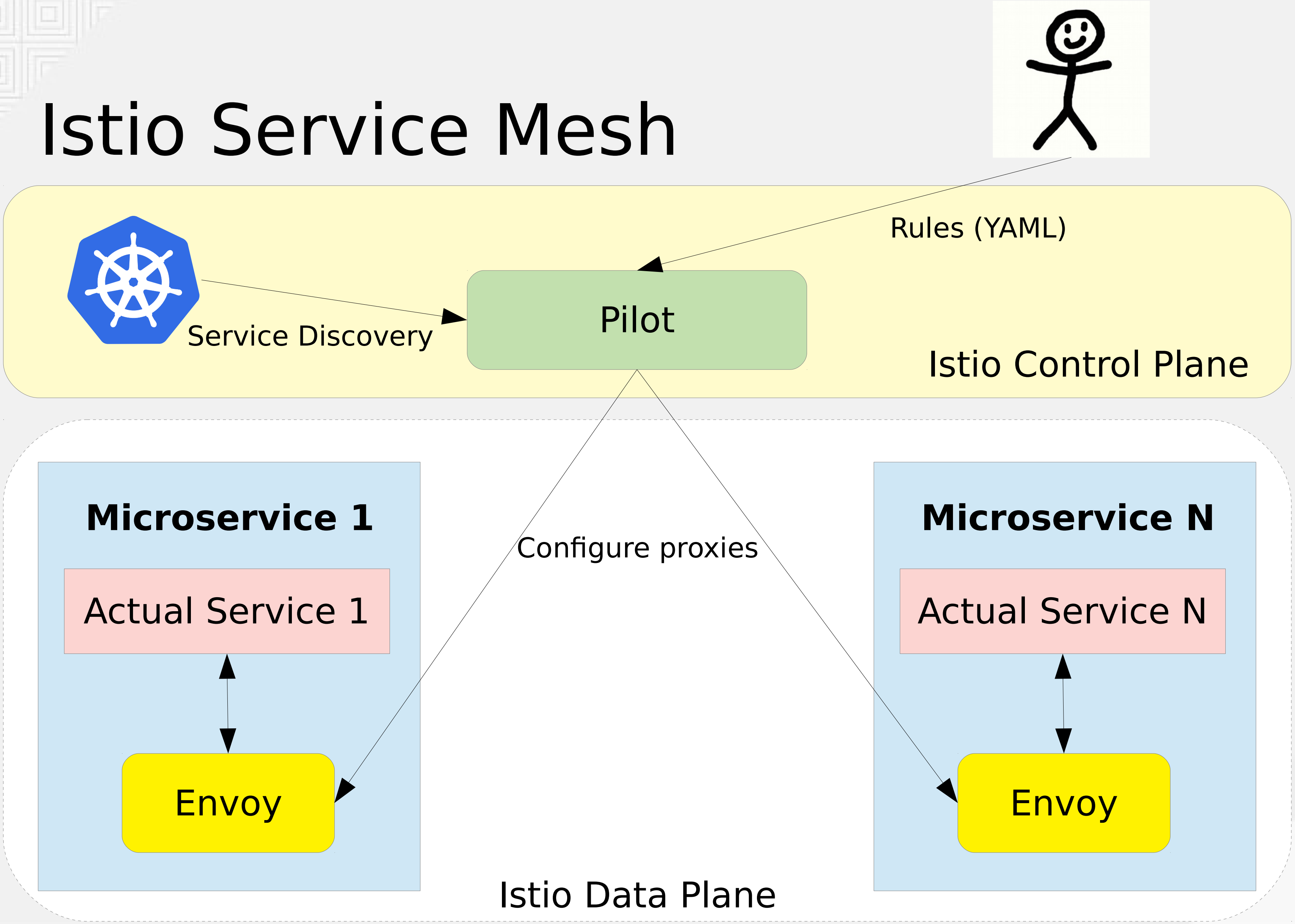
What is Istio?

- Control plane for a service mesh
- Abstracts Envoy concepts and configurations
- Easy to operate: YAML files a la Kubernetes
  - Kubectl or Istioctl can be used
- Created by Google, with the help of other companies
- New project, just reached 1.0

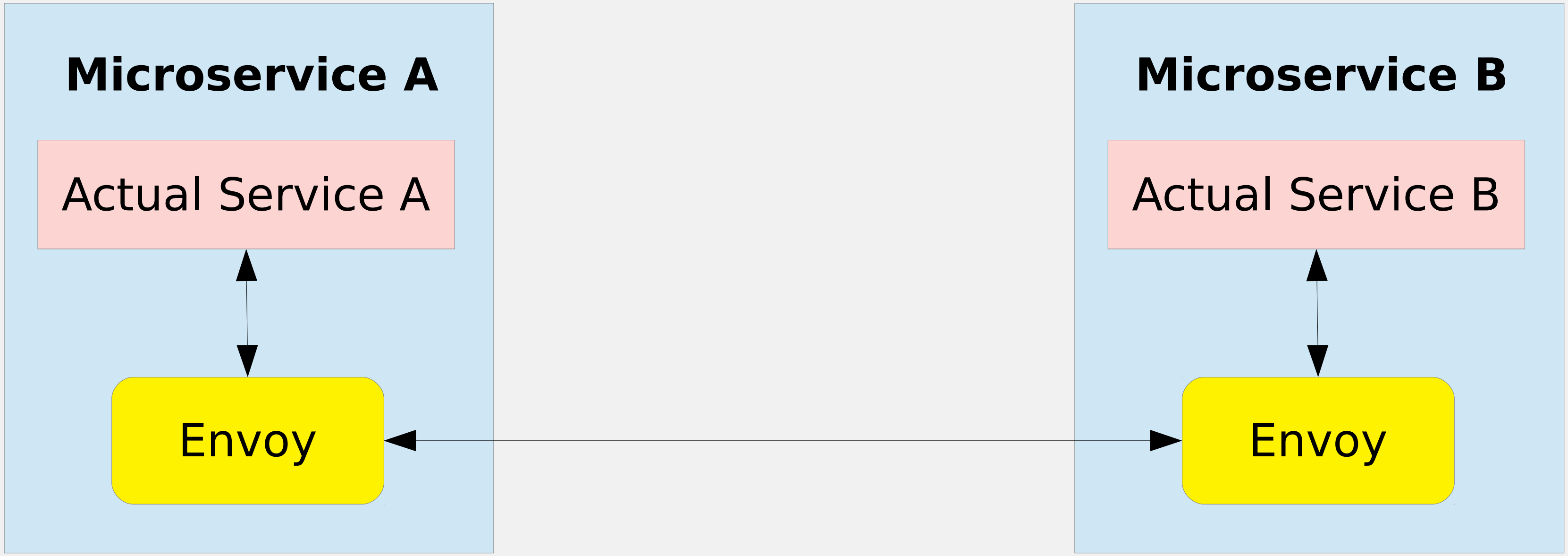
Istio Service Mesh



Istio Service Mesh

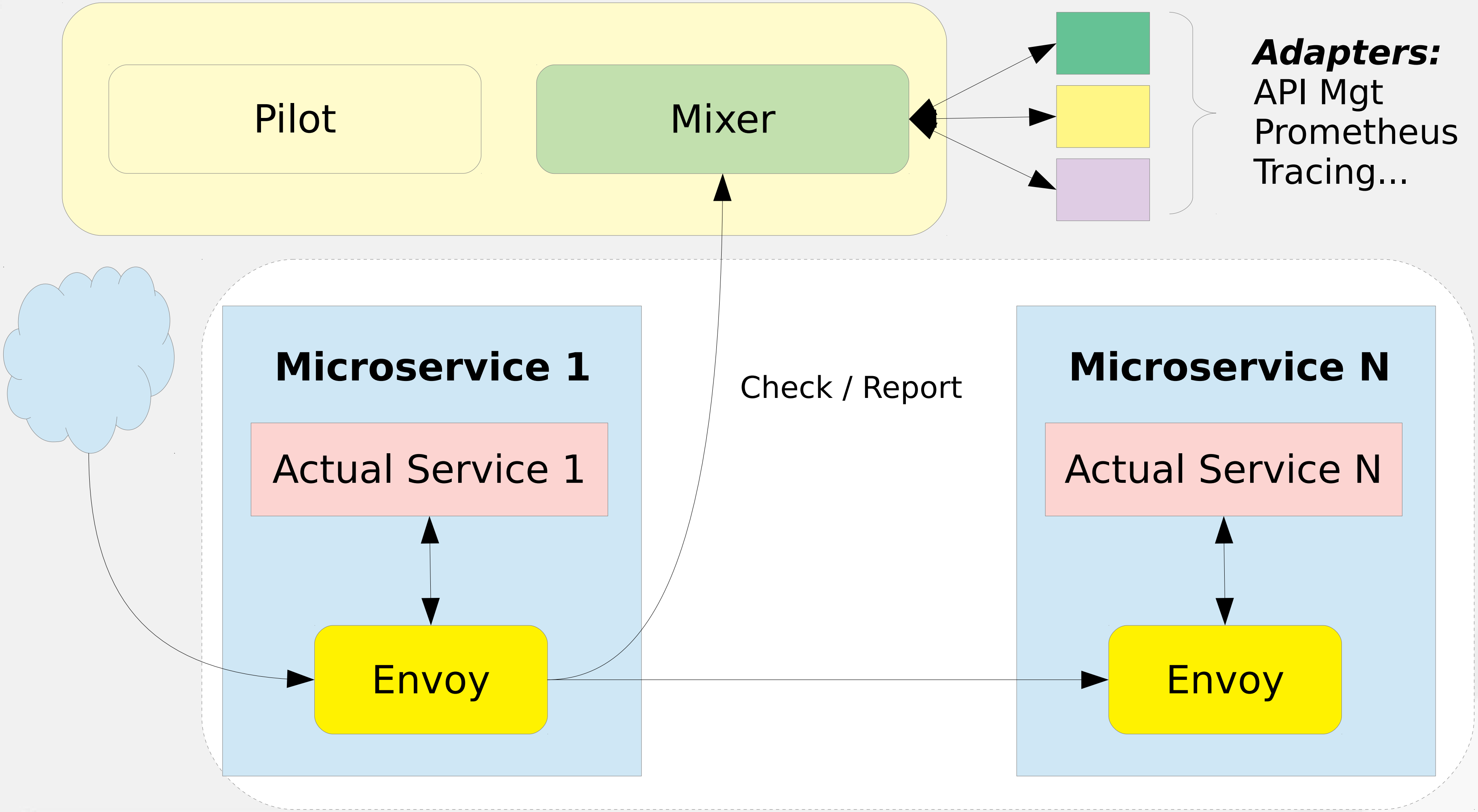


What **pilot** can do?



- Traffic control – enforce route rules & policies
- Resiliency – circuit breaker, timeouts, retries

Istio Service Mesh

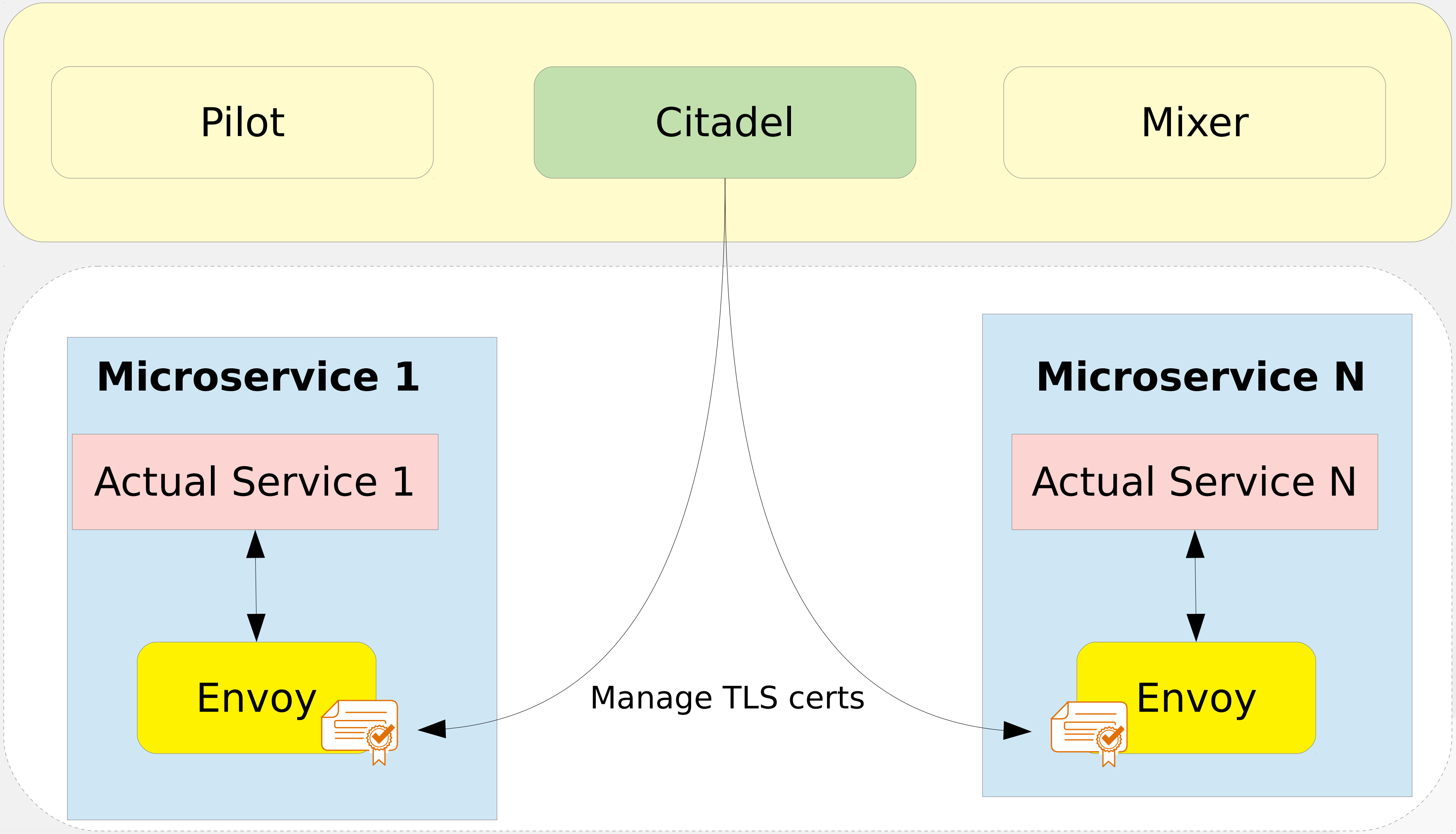


Adapters:
API Mgt
Prometheus
Tracing...

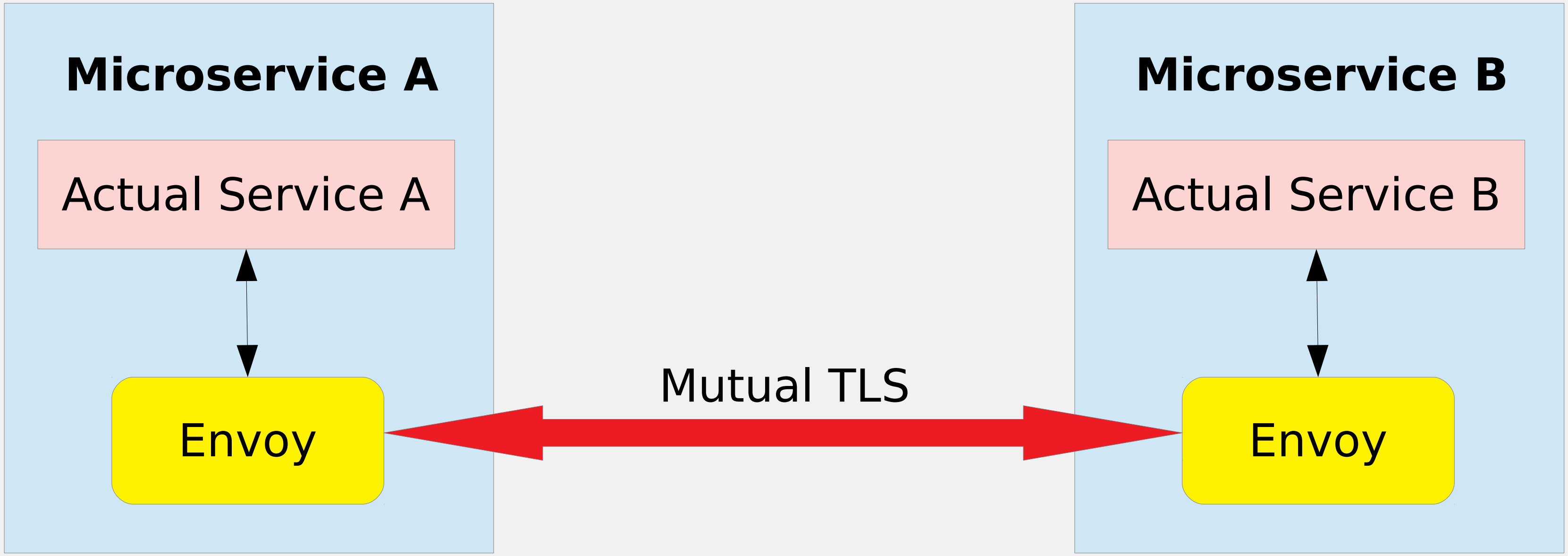
What **mixer** can do?

- Quota / API Management
- Telemetry (Prometheus, ...)
- Tracing (Jeager, ...)
- Your own integration (pluggable model)

Istio Service Mesh



What **citadel** can do?



- Enforce mTLS between services
- Along with mixer and pilot allows authorization and auditing

Demo?

Thank you

- https://istio.io
- https://learn.openshift.com

- Email: jonh.wendell@redhat.com
- Twitter: @jwendell