

# An Introduction to Service Mesh and Istio

(a platform for microservices)

**Jason Dudash**

Specialist Solutions Architect  
Emerging Technology

[github.com/dudash](https://github.com/dudash) 

**Chris Kang**

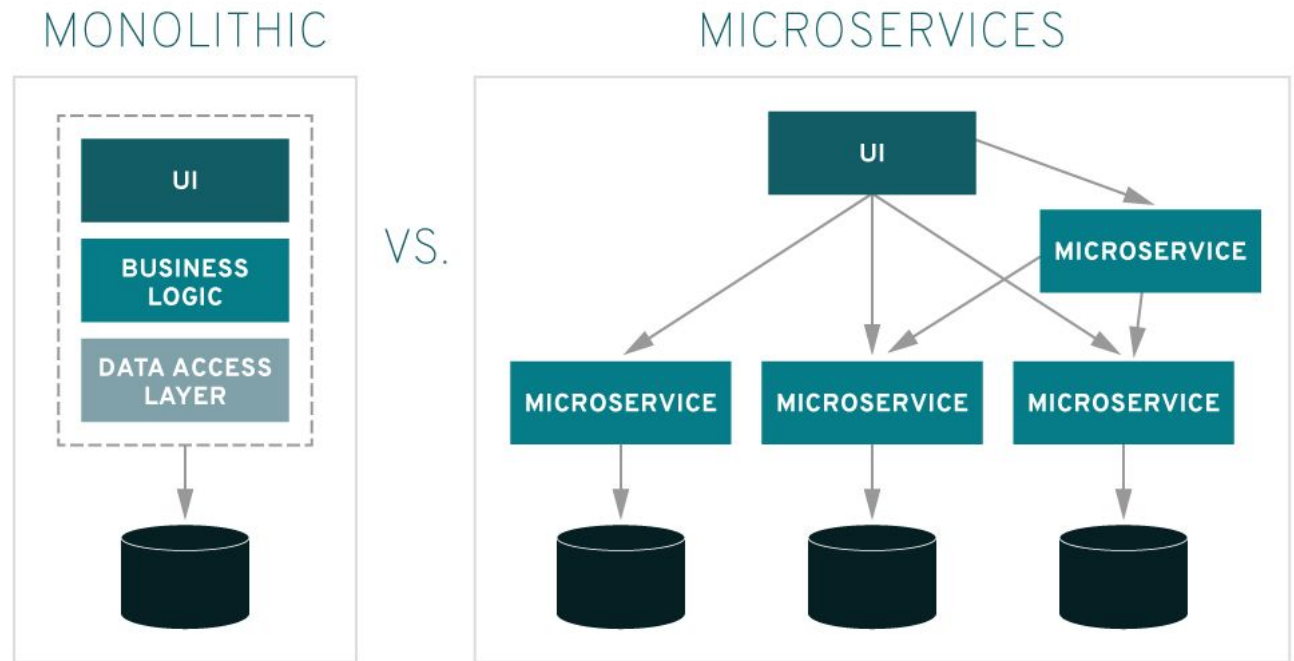
Specialist Solutions Architect  
Emerging Technology

[github.com/theckang](https://github.com/theckang) 

# What are Microservices?

an architectural style that structures an application as a collection of services

- ▶ Single purpose
- ▶ Independently deployable
- ▶ Have their context bound to a biz domain
- ▶ Owned by a small team
- ▶ Often stateless



# Benefits of Microservices



## **Agility**

Deliver updates faster and react faster to new business demands

## **Highly scalable**

Scale independently to meet temporary traffic increases, complete batch processing, or other business needs

## **Can be purpose-built**

Use the languages and frameworks best suited for the service's domain

## **Resilience**

Improved fault isolation restricts service issues, such as memory leaks or open database connections, to only affect that specific service

Many orgs have had success with Microservices - Netflix, Amazon, eBay, The Guardian

# There is inherent complexity in adopting microservices

Some common areas where organizations stumble when adopting microservices

## **Tolerance to Faults**

Cascading failure, partial outages, traffic spikes

## **DevOps and Deployments**

More failure surface, version incompatibility, untracked svcs

## **Services Communication Needs**

Latency, concurrence, distributed transactions

## **Inability to Monitor & Understand Performance**

More to monitor & different types of monitoring required

## **Securing Services**

Malicious requests, DoS, id & access control

## **Highly Distributed Logs**

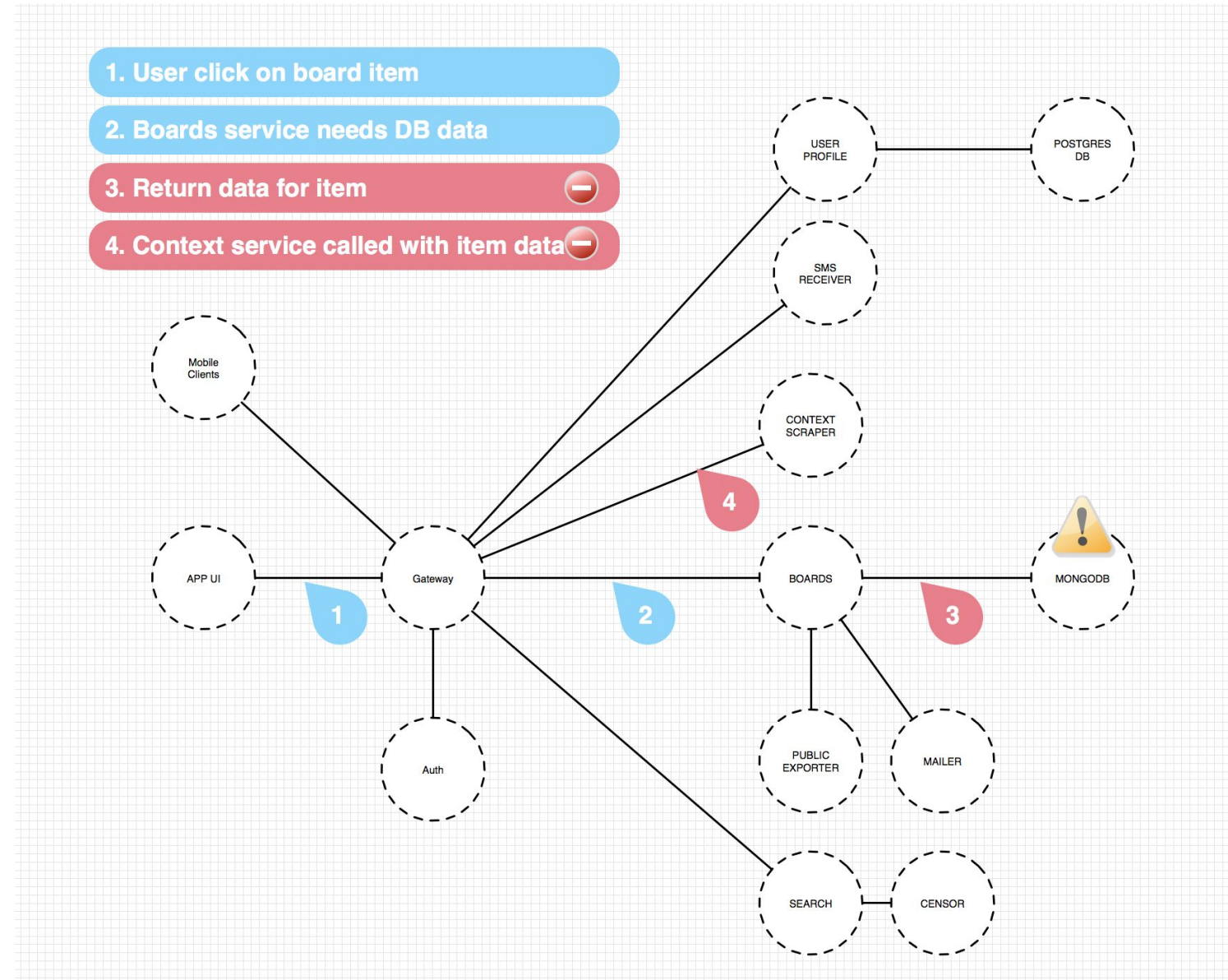
Scattered logs, lots more logs to manage, access control

---

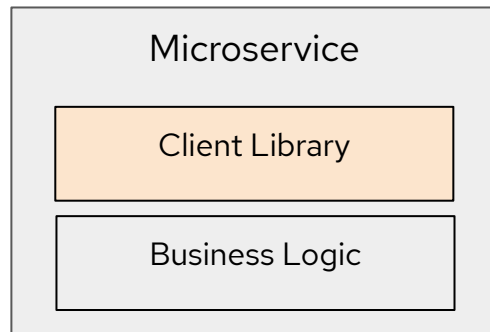
Biggest challenge:  
What used to be internal now  
needs to go across a network

# Partial Failure → Cascading Failures

- ▶ Route service's database goes offline
- ▶ Now no item data can be returned
- ▶ Default timeout for developer's HTTP framework is 2 minutes
  - This wait is happening repeatedly
- ▶ Item data is needed to pass to context scraper service - so it fails too
- ▶ User experience is poor
- ▶ Leads to unexpected case of users repeatedly mashing the refresh button
- ▶ Now the boards service begins to get more requests than it can handle



# Language Specific Libraries and Tools



## Narrow Scope

Built to address a specific problem such as fault tolerance

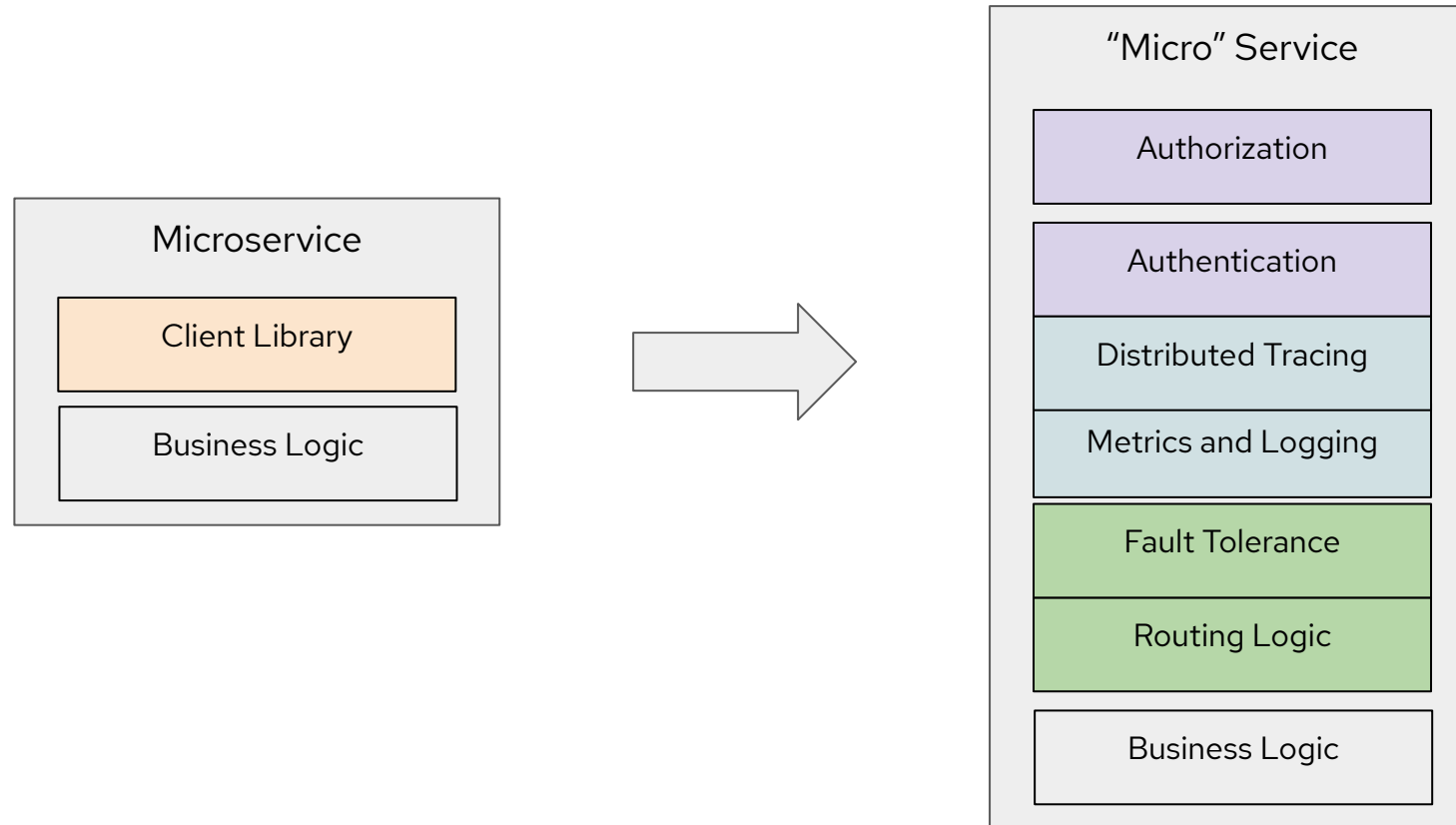
## Language Specific

Client libraries are tied to programming language

## Thick Clients

Solution results in clients with bulk of capabilities

## Is This a “Micro” Service?



Does it *really* make sense to push operational challenges to developers to deal with?



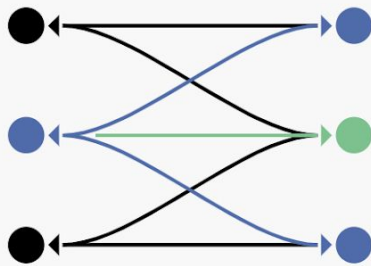
---

Don't force extra work  
on developers

There is a better way

# Istio Service Mesh

A modern way to manage the complexity of microservice applications



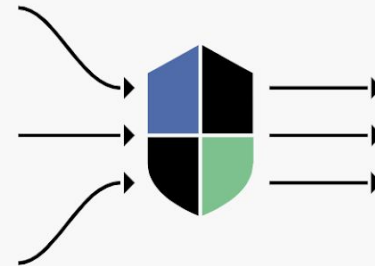
## Connect

Intelligently control the flow of traffic and API calls between services, conduct a range of tests, and upgrade gradually with red/black deployments.



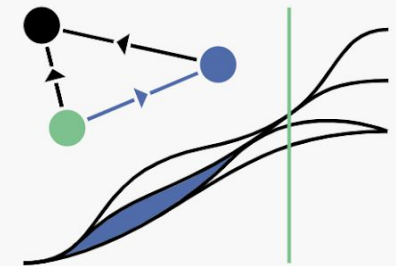
## Secure

Automatically secure your services through managed authentication, authorization, and encryption of communication between services.



## Control

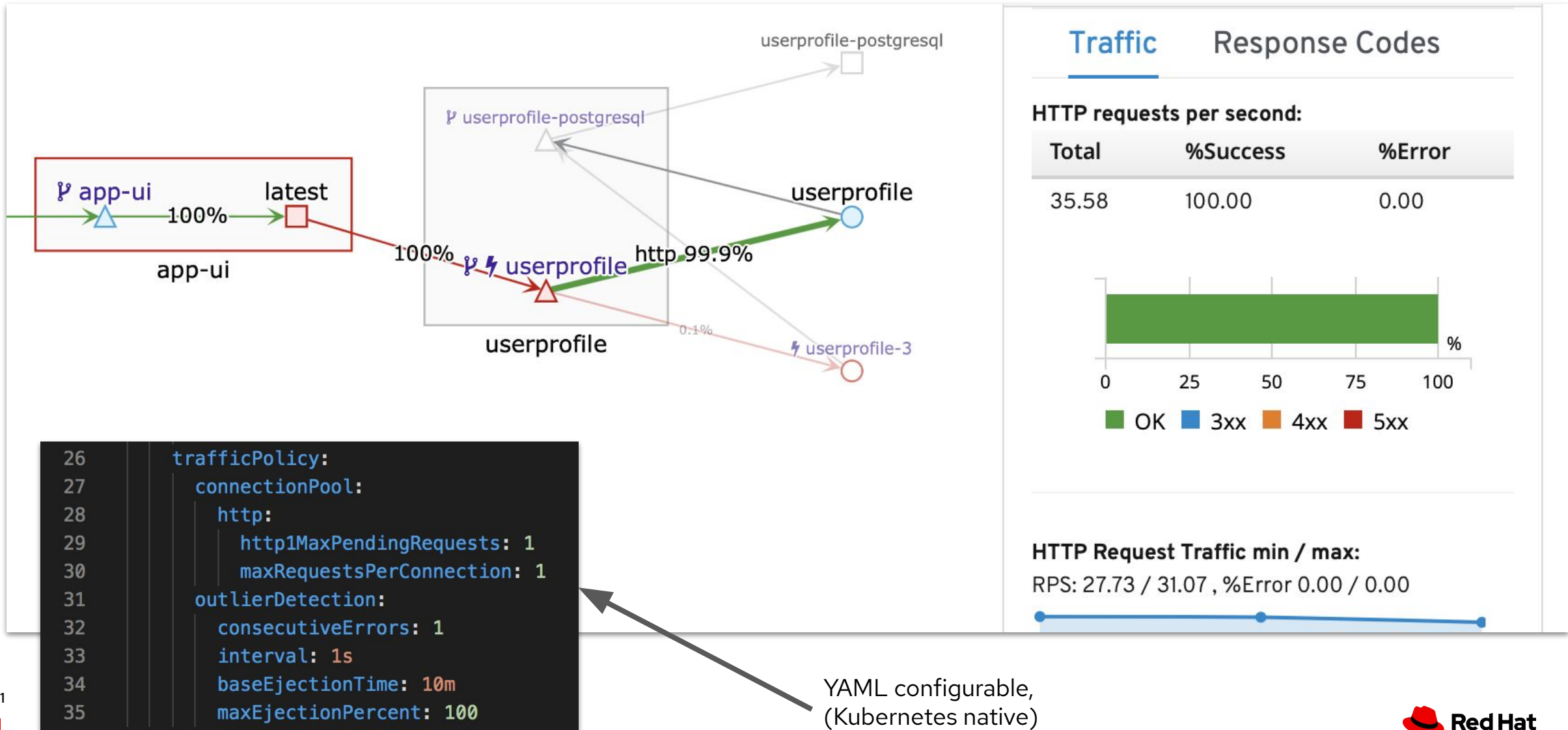
Apply policies and ensure that they're enforced, and that resources are fairly distributed among consumers.



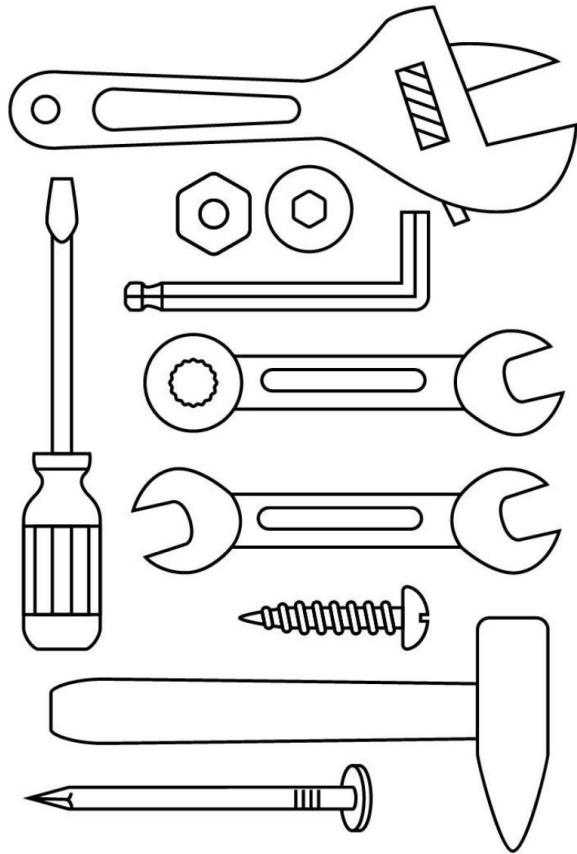
## Observe

See what's happening with rich automatic tracing, monitoring, and logging of all your services.

# Handling Partial Failures with the Service Mesh



# Features needed to meet the challenges of microservices



CIRCUIT BREAKING AND BULKHEADS

RATE LIMITING

MIRRORING / TRAFFIC SHIFTING

VERSION BASED ROUTING

AUTOSCALING

STAGED ROLLOUTS

CANARY DEPLOYMENTS

BLUE/GREEN DEPLOYMENTS

DISTRIBUTED TRACING

VISUAL SERVICE HEALTH

DISTRIBUTED LOGGING

COLLECTING AND VISUALIZING METRICS

CHAOS ENGINEERING

SERVICE SECURITY

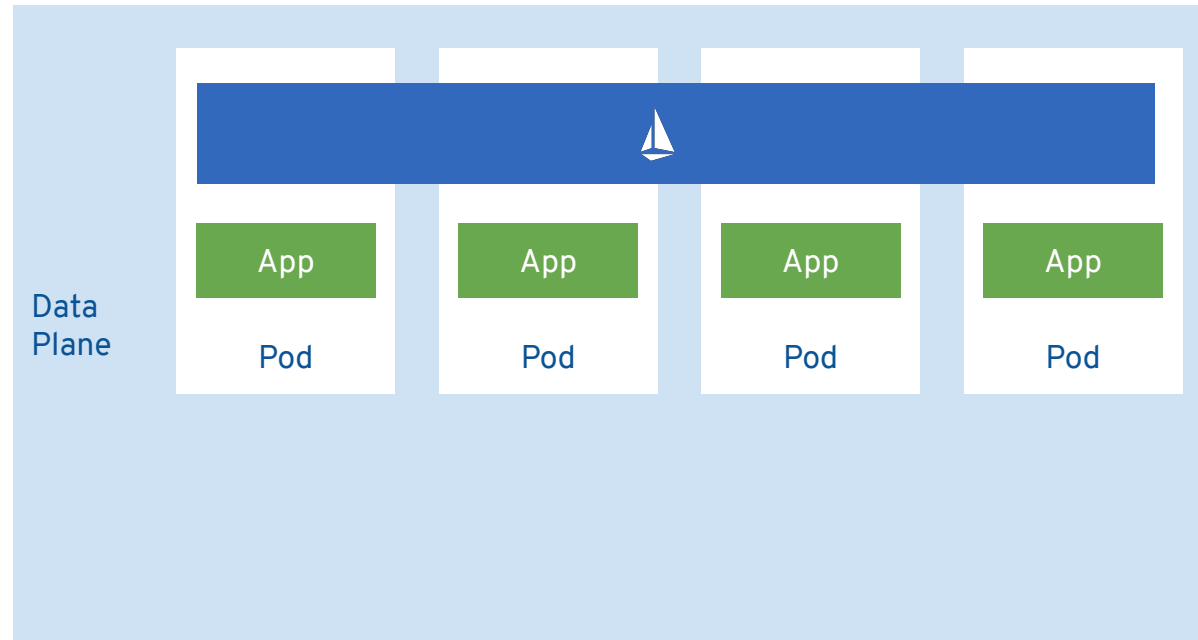
CONTAINER BUILD AUTOMATION

**Table Stakes** for  
doing microservices

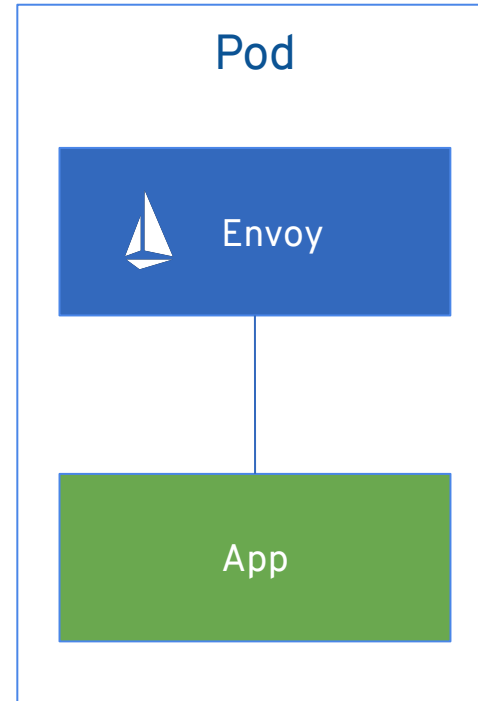
---

The service mesh is critical in addressing the inherent complexity of microservices

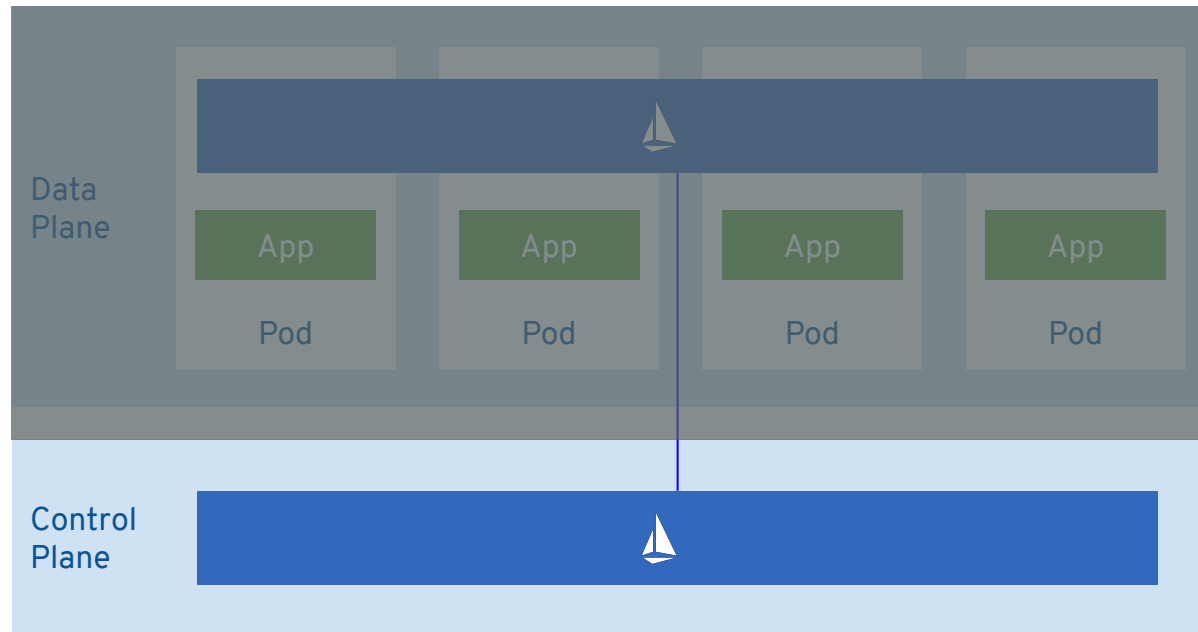
## Your Services are in a “Data Plane”



## What's a Sidecar have to do with containers?

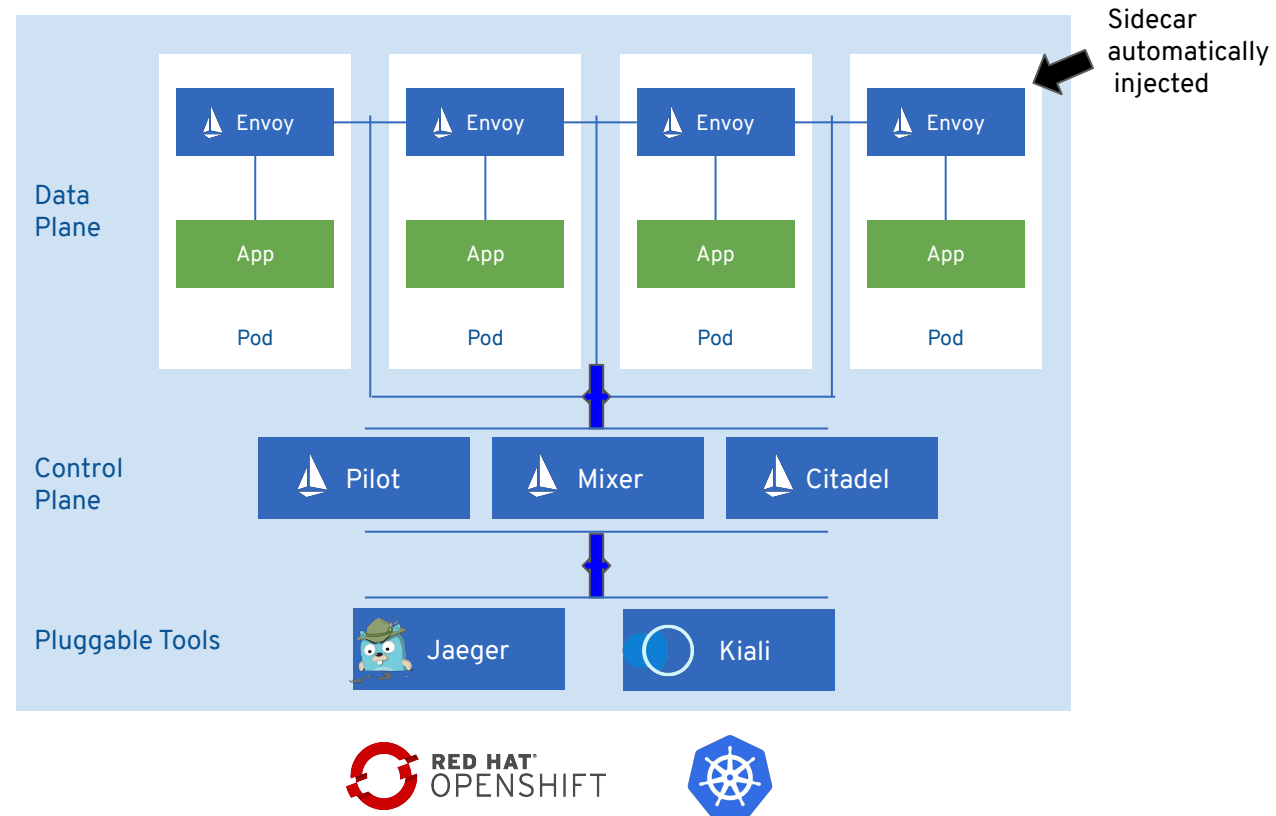


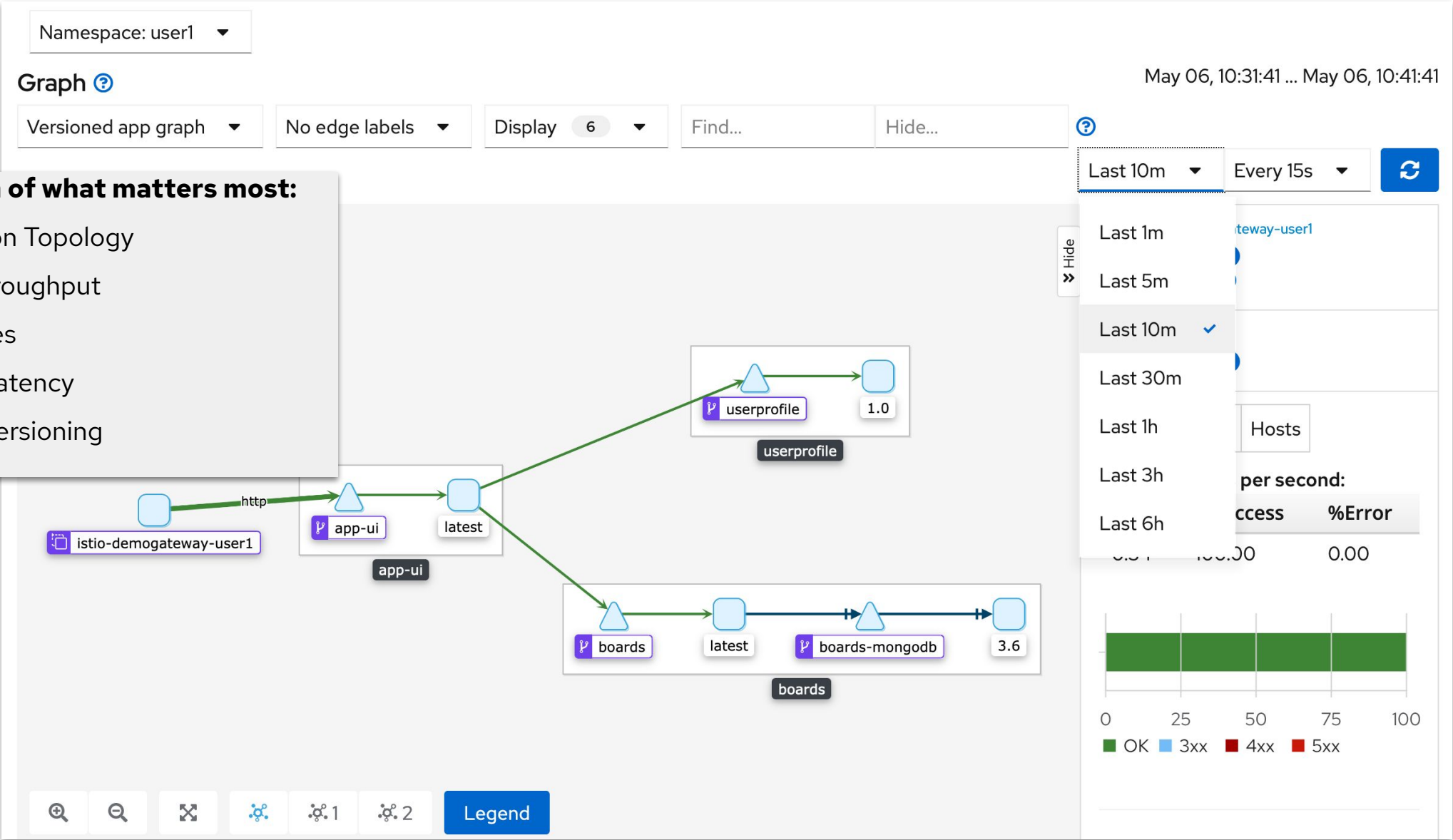
## Your Policy Makes Up a “Control Plane”





## Let's Break it Down a Little Bit More

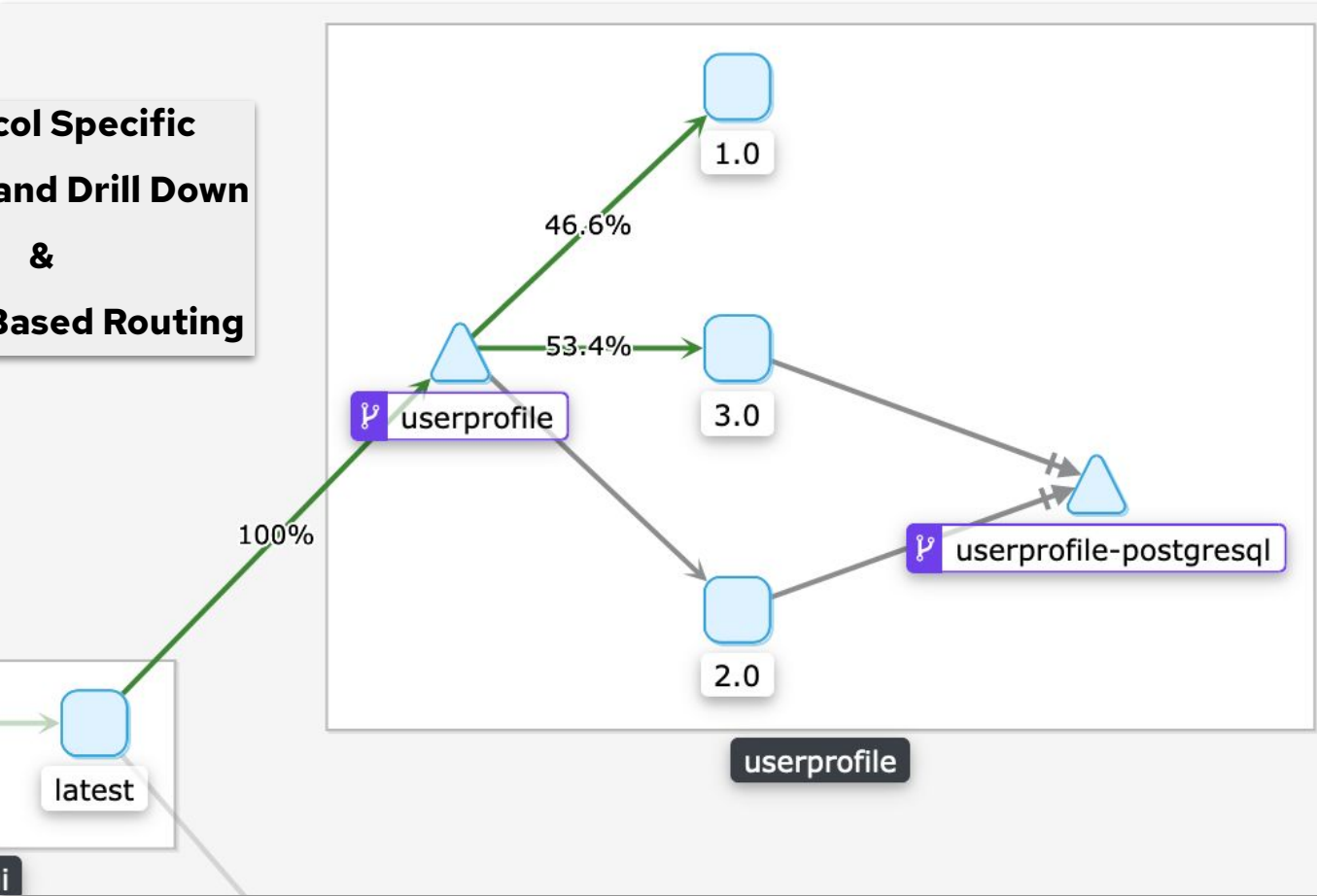




**Visualization of what matters most:**

- Application Topology
- Traffic throughput
- Error Rates
- Service Latency
- Service Versioning

Protocol Specific  
Analysis and Drill Down  
&  
Version Based Routing



Hide  
>>

NS useri

Current Graph:

- 4 apps (7 versions)
- 5 services
- 12 edges

Incoming	Outgoing	Total
----------	----------	-------

HTTP (requests per second):




Total	%Success	%Error
4.82	100.00	0.00



**Guided Config of  
Traffic Policies**

That's real-time  
version based routing!

Create Weighted Routing

WORKLOAD	TRAFFIC WEIGHT
 reviews-v1	<div><div></div><div>-5+</div><div>%</div><div></div></div>
 reviews-v2	<div><div></div><div>-80+</div><div>%</div><div></div></div>
 reviews-v3	<div><div></div><div>-15+</div><div>%</div><div></div></div>

Evenly distribute traffic

Hide Advanced Options

VirtualService Hosts

reviews

The destination hosts to which traffic is being sent. Enter one or multiple hosts separated by comma.

TLS

DISABLE

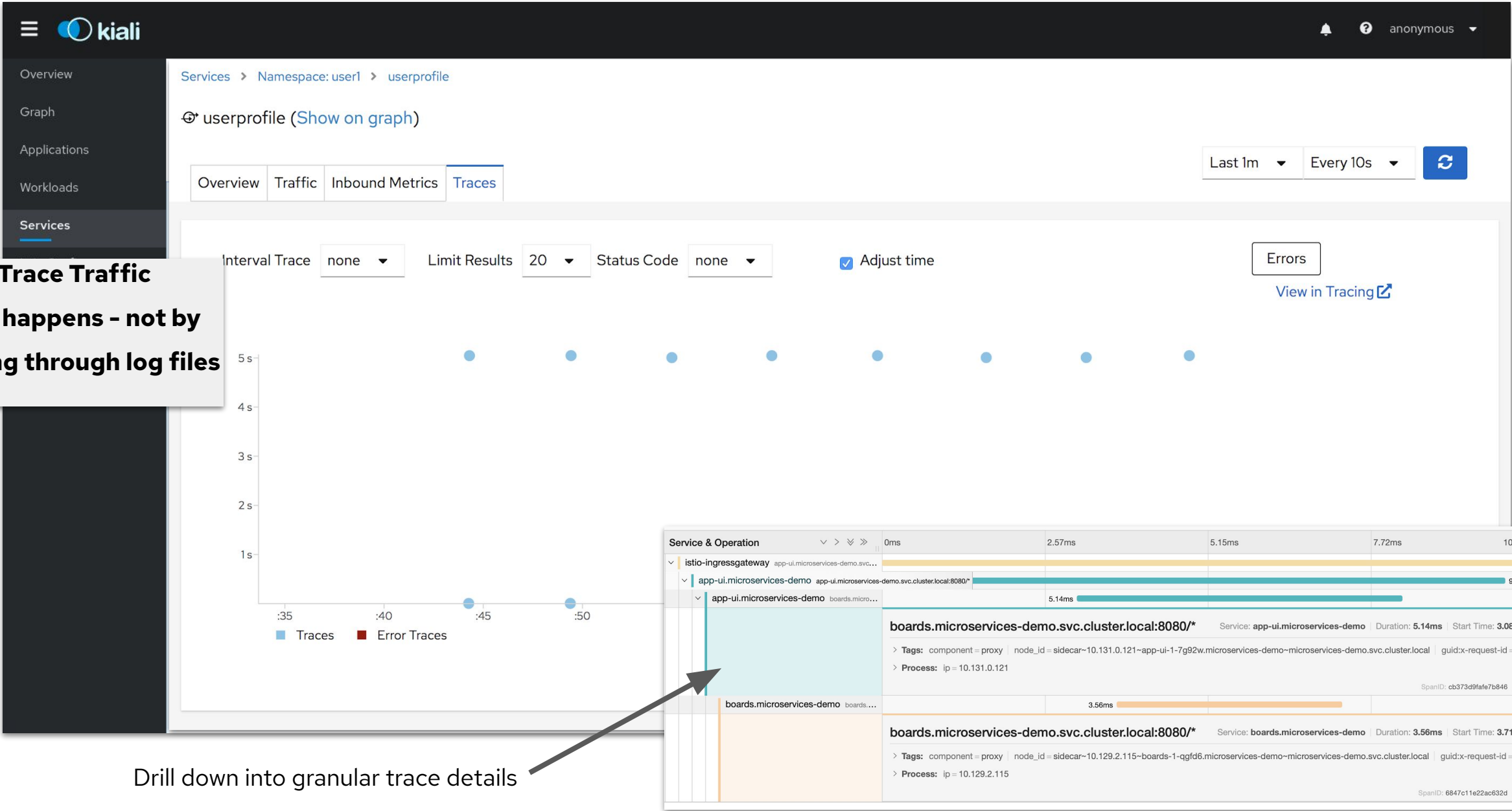
Add LoadBalancer

OFF

Add Gateway

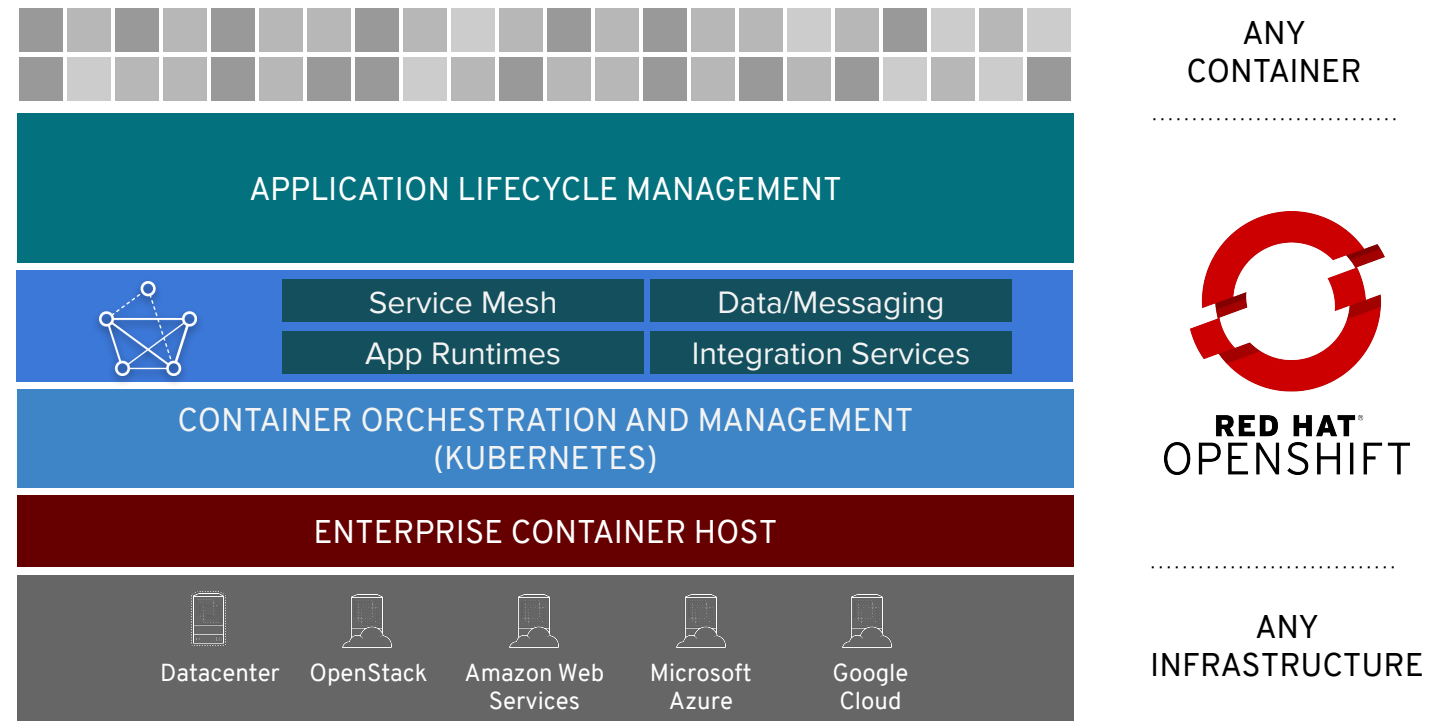
OFF

**Trace Traffic**  
as it happens - not by  
digging through log files



Drill down into granular trace details

# Bundled Platform for Microservices



---

# The future is Istio

## Summary and Resources

- ▶ Microservices are great but they come with challenges
- ▶ Expecting developers to deal with all the challenges of microservices is unrealistic
- ▶ Service Mesh can reduce the development and operational complexity of microservices based applications
- ▶ Service Mesh 1.1 out now! (included with OpenShift)
- ▶ Self paced hands-on: <https://learn.openshift.com/servicemesh/>
- ▶ Guided technical workshop: Reach out to your Red Hat account rep.

A Service Mesh is critical to run microservices and requires little to no changes to your code



# Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.

 [linkedin.com/company/red-hat](https://linkedin.com/company/red-hat)

 [youtube.com/user/RedHatVideos](https://youtube.com/user/RedHatVideos)

 [facebook.com/redhatinc](https://facebook.com/redhatinc)

 [twitter.com/RedHat](https://twitter.com/RedHat)

---

# Backup Slides

openshift-service-mesh/citadel-rhel8

Red Hat Service Mesh Citadel OpenShift container image

by [Red Hat, Inc.](#)
 | in Product [Red Hat OpenShift Service Mesh](#)

- Overview
- Get This Image
- Tech Details
- Support
- Tags

Show images built for:

AMD64 (9)

Tag Name	Date Pushed	Image Advisory	Health Index	Docker Image ID
<div>1.0.8-2</div> <div>1.0</div> <div>latest</div> <div>1.0.8</div>	8 days ago	<a href="#">RHEA-2020:0624</a>	<div>B</div> <div><div></div><div></div><div></div><div></div><div></div></div>	f633f954de6d
<div>1.0.7-1</div> <div>1.0.7</div>	22 days ago	<a href="#">RHBA-2020:0479</a>	<div>B</div> <div><div></div><div></div><div></div><div></div><div></div></div>	ded4935f2d0a
<div>1.0.6-1</div> <div>1.0.6</div>	a month ago	<a href="#">RHEA-2020:0320</a>	<div>C</div> <div><div></div><div></div><div></div><div></div><div></div></div>	6b5e724207d9
<div>1.0.5-1</div> <div>1.0.5</div>	a month ago	<a href="#">RHEA-2020:0277</a>	<div>C</div> <div><div></div><div></div><div></div><div></div><div></div></div>	3df685ae7447
<div>1.0.4-1</div> <div>1.0.4</div>	2 months ago	<a href="#">RHEA-2020:0104</a>	<div>C</div> <div><div></div><div></div><div></div><div></div><div></div></div>	defa731519d7
<div>1.0.3-1</div> <div>1.0.3</div>	3 months ago	<a href="#">RHBA-2019:4224</a>	<div>C</div> <div><div></div><div></div><div></div><div></div><div></div></div>	464f4f8b1d7e
<div>1.0.2-1</div> <div>1.0.2</div>	4 months ago	<a href="#">RHEA-2019:3810</a>	<div>D</div> <div><div></div><div></div><div></div><div></div><div></div></div>	02de2cf8369f
<div>1.0.1-1</div> <div>1.0.1</div>	5 months ago	<a href="#">RHBA-2019:3042</a>	<div>C</div> <div><div></div><div></div><div></div><div></div><div></div></div>	0b3eb8ab1526
<div>1.0.0-3</div> <div>1.0.0</div>	6 months ago	<a href="#">RHEA-2019:2677</a>	<div>D</div> <div><div></div><div></div><div></div><div></div><div></div></div>	385163f0f645



## PROPOSED OUTLINE

1. Microservices Defined
2. Challenges with Microservices
  - a. Building, debugging, connecting, securing
3. The biggest challenge – communications are now network calls (and FoDC)
4. Breakdown an examples of a problem
5. But the benefits are often worth the challenges (so we solve them)
6. Past way to solve challenges with microservices (it's a lot of work)
  - a. Java libs like netflix, built in lang constructs, etc.
  - b. K8s – service discovery and orchestration – but it's not enough
7. A better way – introducing service mesh
8. Details on service mesh – builds on top of K8s
9. Data plane, control plane
10. Envoy proxy and sidecar containers
11. Revisiting the challenges and how a mesh addresses them
  - a. Observability + visualization
  - b. Tracing
  - c. Traffic control
  - d. Security
12. Other things a mesh can do
13. The microservices future is Istio – a foundation for other tech – highlight project popularity, commercial use, etc.
14. How this brings value to your business / gov't agencies (?)
15. Summary
  - a. Plug the upcoming workshop / whitepaper?

*References and places to borrow content from:*

- BU INTRO TO SERVICE MESH
  - [https://docs.google.com/presentation/d/1afNJ2zJ8agr7zTyDY3ud8-ZRKUmxUR94X0TF\\_XXNT64/edit#slide=id.g38d28c45af\\_2\\_14](https://docs.google.com/presentation/d/1afNJ2zJ8agr7zTyDY3ud8-ZRKUmxUR94X0TF_XXNT64/edit#slide=id.g38d28c45af_2_14)
- BU PRODUCT VISION
  - [https://docs.google.com/presentation/d/1Nz8JVtcOLyaSBzZohRU6XHDG5BMFtpP66jY7NPLlulE/edit#slide=id.gb6f3e2d2d\\_2\\_219](https://docs.google.com/presentation/d/1Nz8JVtcOLyaSBzZohRU6XHDG5BMFtpP66jY7NPLlulE/edit#slide=id.gb6f3e2d2d_2_219)
- DUDASH SUMMIT 2019
  - [https://docs.google.com/presentation/d/1\\_CNuErKYe0sXlojhA3t7HeJXZIATN7-mxGzFNvcFv7g/edit](https://docs.google.com/presentation/d/1_CNuErKYe0sXlojhA3t7HeJXZIATN7-mxGzFNvcFv7g/edit)
- REDHAT.COM WHAT IS A SERVICE MESH
  - <https://www.redhat.com/en/topics/microservices/what-is-a-service-mesh>
- REDHAT.COM WHY CHOOSE RED HAT SERVICE MESH
  - <https://www.redhat.com/en/topics/microservices/why-choose-openshift-service-mesh>
- REDHAT.COM MICROSERVICES
  - <https://www.redhat.com/en/topics/microservices>
- JOHN AND JASON'S LAST WEBINAR
  - [https://docs.google.com/presentation/d/1j\\_6kX2V-\\_ds5BRR4BW6S1vxCSvC8T\\_b0Skr15mTr\\_14/edit#slide=id.p10](https://docs.google.com/presentation/d/1j_6kX2V-_ds5BRR4BW6S1vxCSvC8T_b0Skr15mTr_14/edit#slide=id.p10)
-