



redhat.



Microsoft Azure

MONOLITHS TO MICROSERVICES: APP TRANSFORMATION

Hands-on Technical Workshop

PART 5: RESILIENT DISTRIBUTED APPS

DISTRIBUTED SERVICES ARCHITECTURES

Benefits (when implemented correctly):

- Performance
- Reliability
- Resiliency
- Extensibility
- Availability
- Robustness

DISTRIBUTED SERVICES ARCHITECTURES

Fallacies of Distributed Computing

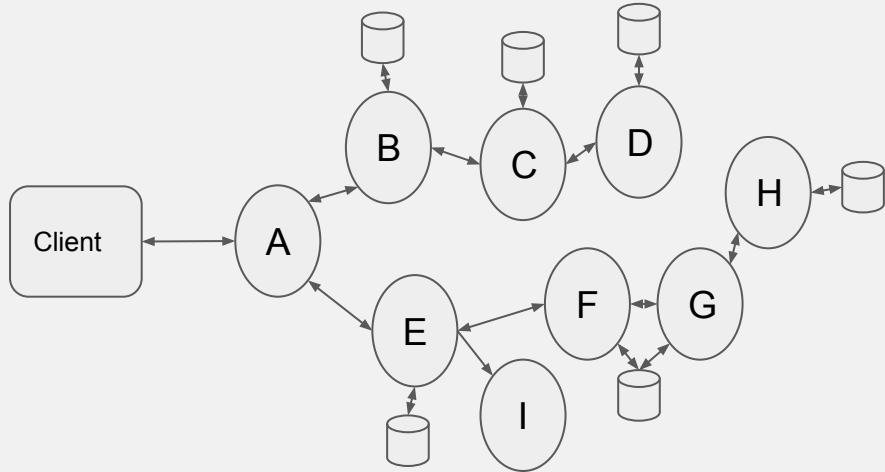
- The network is reliable.
- Latency is zero.
- Bandwidth is infinite.
- The network is secure.
- Topology doesn't change.
- There is one administrator.
- Transport cost is zero.
- The network is homogeneous.

[wikipedia.org/wiki/Fallacies_of_distributed_computing](https://en.wikipedia.org/wiki/Fallacies_of_distributed_computing)

DISTRIBUTED SERVICES ARCHITECTURES

Applications must deal with

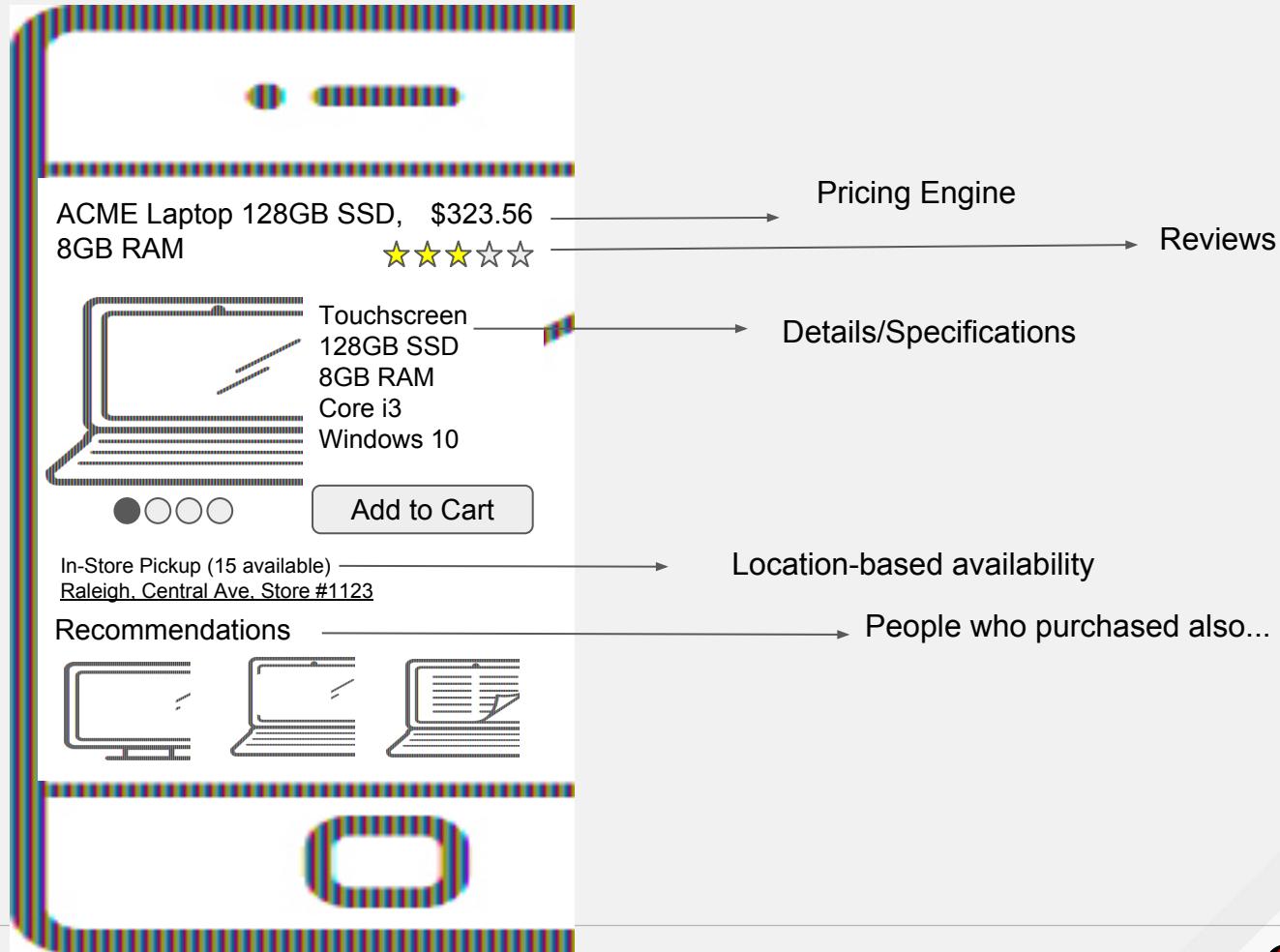
- Unpredictable failure modes
- End-to-end application correctness
- System degradation
- Topology changes
- Elastic/ephemeral/transient resources



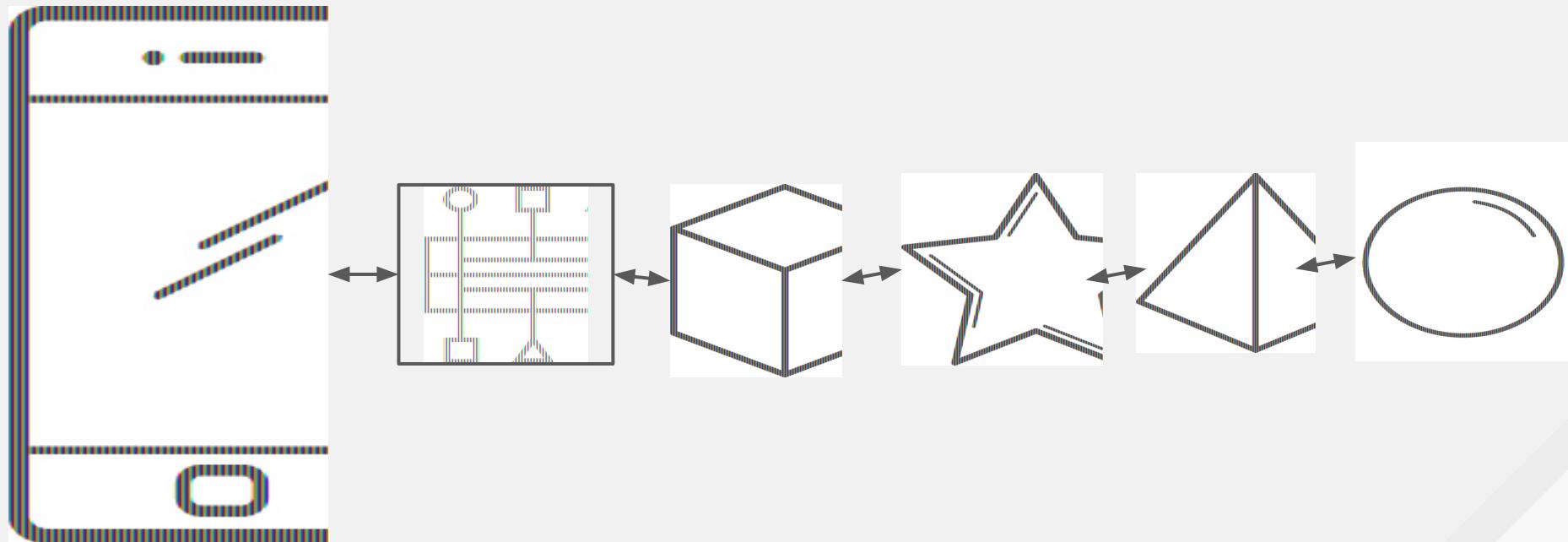


MICROSERVICES == DISTRIBUTED COMPUTING

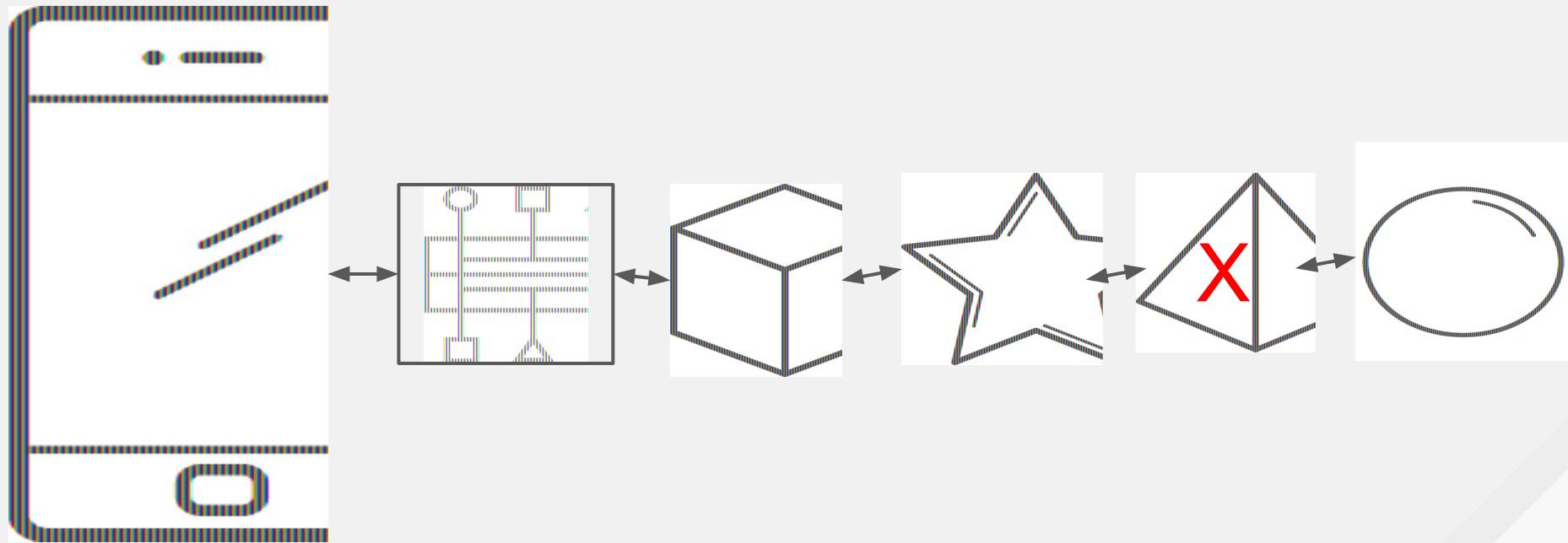
Example



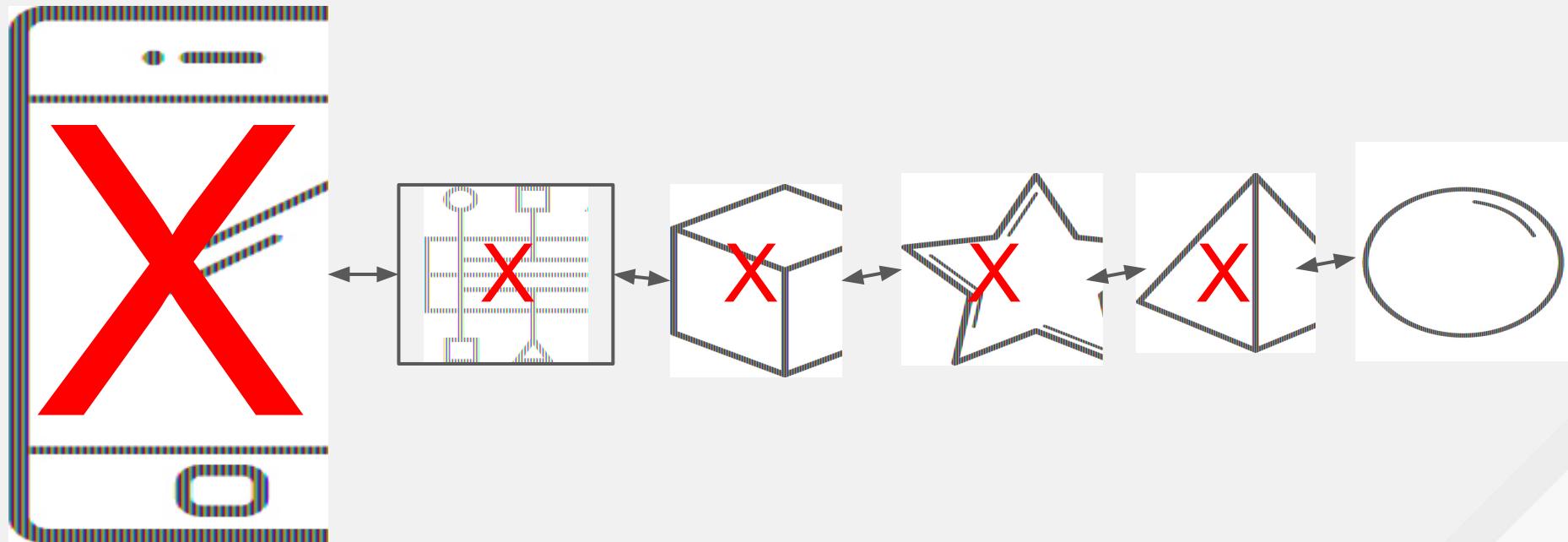
Chaining



Chaining (Fail)



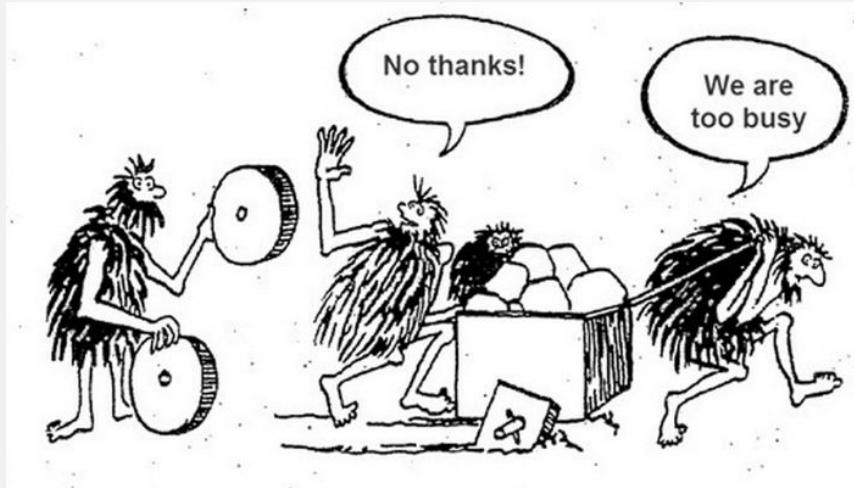
Chaining (Cascading Fail)



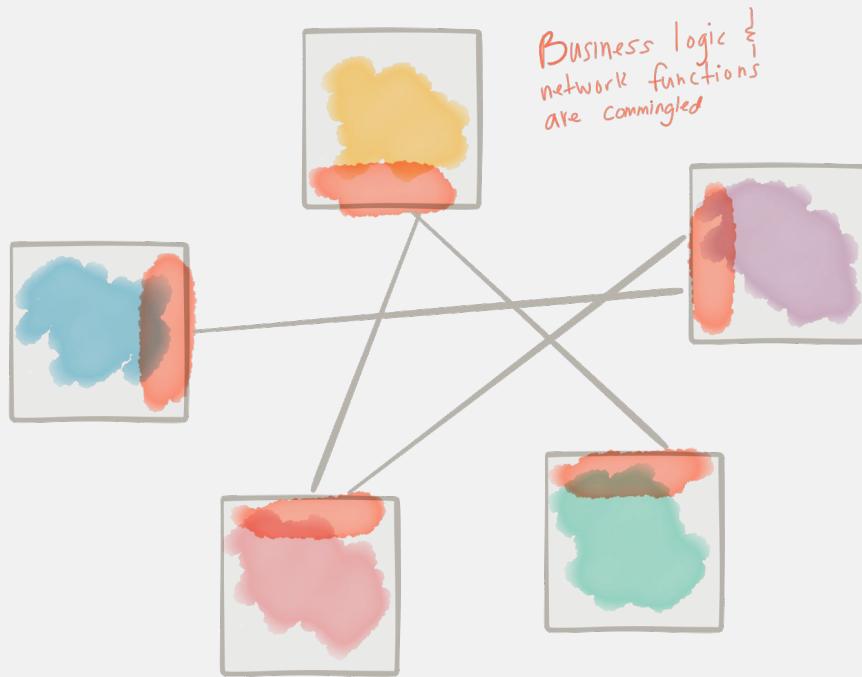
POSSIBLE SOLUTIONS

Today, Developers do this:

- Circuit Breaking
- Bulkheading
- Timeouts/Retries
- Service Discovery
- Client-side Load Balancing



TOO MUCH INFRASTRUCTURE IN BUSINESS LOGIC



BUT I'M USING...

spring

```
spring-cloud-netflix-hystrix  
spring-cloud-netflix-zuul  
spring-cloud-netflix-eureka-client  
spring-cloud-netflix-ribbon  
spring-cloud-netflix-atlas  
spring-cloud-netflix-spectator  
spring-cloud-netflix-hystrix-stream  
...  
@Enable....150MagicThings
```



```
org.wildfly.swarm.hystrix  
org.wildfly.swarm.ribbon  
org.wildfly.swarm.topology  
org.wildfly.swarm.camel-zookeeper  
org.wildfly.swarm.hystrix  
org.wildfly.swarm.hystrix  
...
```

VERT.X

```
vertx-circuit-breaker  
vertx-service-discovery  
vertx-dropwizard-metrics  
Vertx-zipkin  
...
```

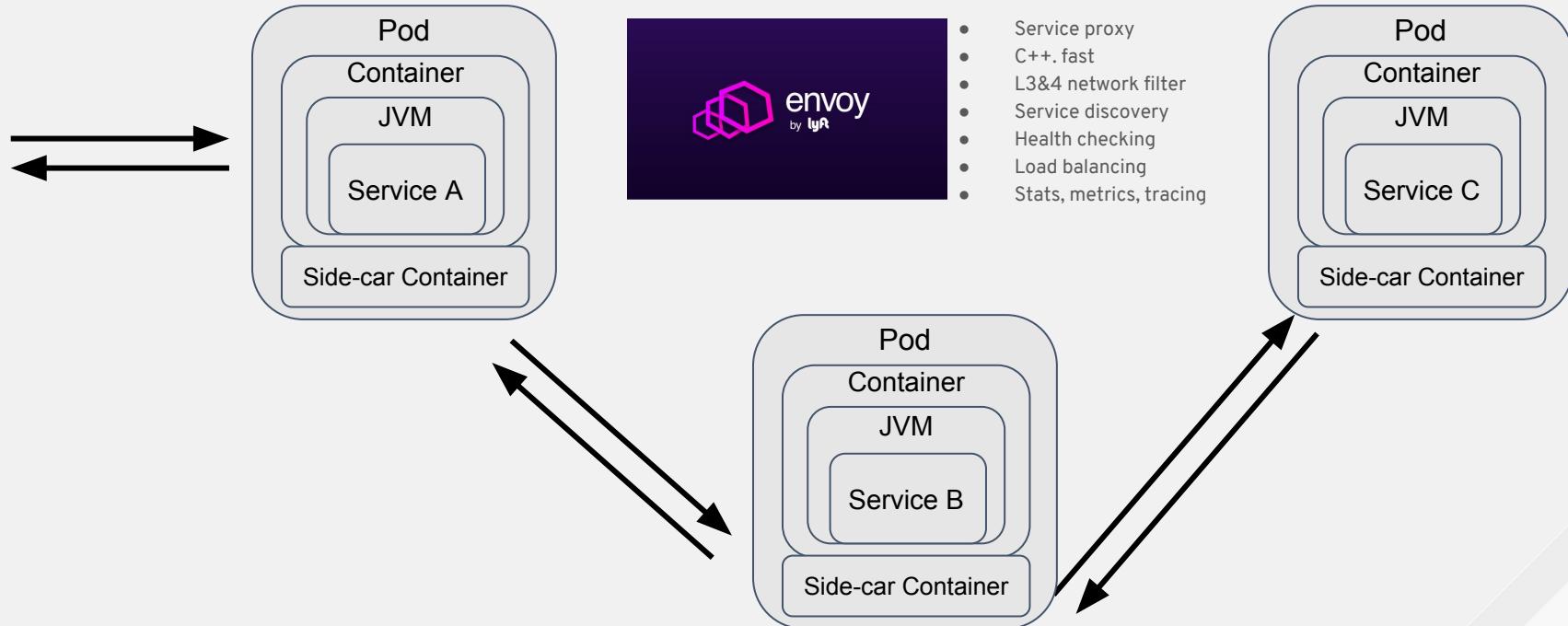
- + Node.js
- + Go
- + Python
- + Ruby
- + Perl
- +

SIDECARS



INSERT DESIGNATOR, IF NEEDED

PODS WITH TWO CONTAINERS





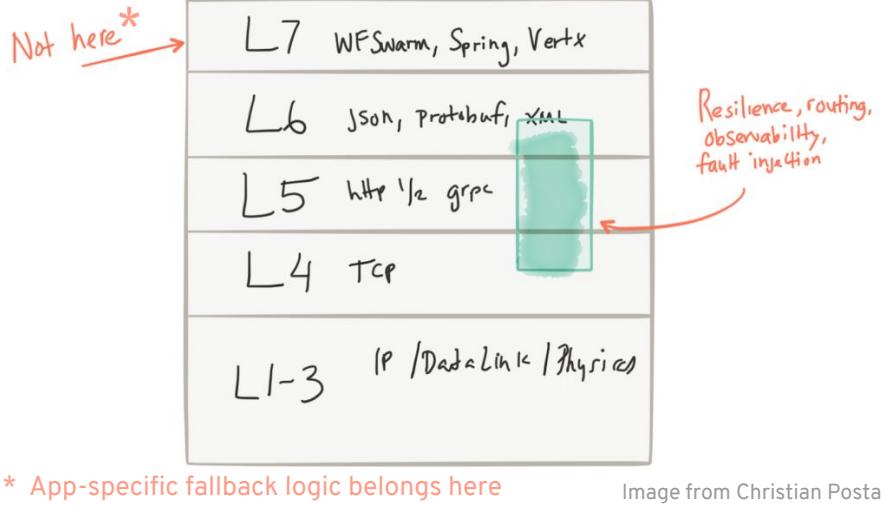
Istio - Sail

(Kubernetes - Helmsman or ship's pilot)

ISTIO - A ROBUST SERVICE MESH FOR MICROSERVICES

Key Features

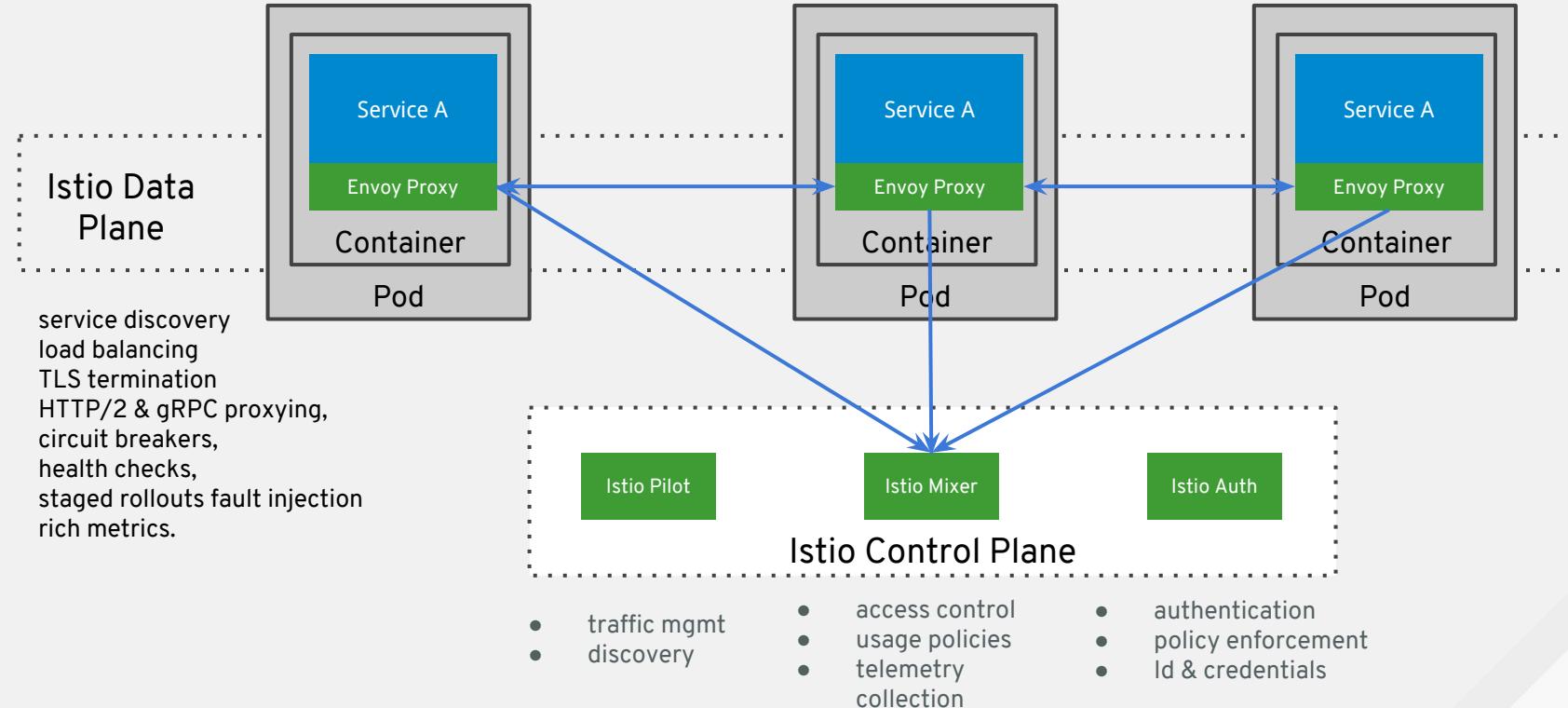
- Intelligent routing and load balancing
- Fleet-wide, in-depth observability
- Resiliency across languages and platforms
- Fault injection
- Developer productivity
- Policy driven ops
- Circuit breaking, outlier detection
- Timeouts/retries
- Rate limiting
- Secure by default
- Incremental, unobtrusive adoption



Further Reading :

<https://blog.openshift.com/red-hat-istio-launch/>
<https://istio.io/blog/istio-service-mesh-for-microservices.html>
<http://blog.christianposta.com/microservices/the-hardest-part-of-microservices-calling-your-services/>

ISTIO - A ROBUST SERVICE MESH FOR MICROSERVICES



MICROSERVICES 3.0 - SERVICE MESH

Code Independent:

- Intelligent Routing and Load-Balancing
 - A/B Tests
 - Canary Releases
 - Dark Launches
- Distributed Tracing
- Circuit Breakers
- Fine grained Access Control
- Telemetry, metrics and Logs
- Fleet wide policy enforcement



LAB: DETECTING AND PREVENTING ISSUES IN DISTRIBUTED APPS WITH ISTIO

GOAL FOR LAB

In this lab you will learn:

- How to **install Istio** onto OpenShift Container Platform
- How to deploy apps with **sidecar proxies**
- How to generate and visualize **deep metrics** for apps
- How to **alter routing** dynamically
- How to **inject faults** for testing
- How to do **rate limiting**
- How Istio implements **circuit breaking** and **distributed tracing**

SAMPLE APP: “BookInfo”

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "BookInfo Sample" and has the URL "192.168.99.101:31463/productpage". The page content is as follows:

The Comedy of Errors

Wikipedia Summary: The Comedy of Errors is one of **William Shakespeare's** early plays. It is his shortest and one of his most farcical comedies, with a major part of the humour coming from slapstick and mistaken identity, in addition to puns and word play.

Book Details

Paperback:
200 pages
Publisher:
PublisherA
Language:
English
ISBN-10:
1234567890
ISBN-13:
123-1234567980

A red rectangular box highlights the "Book Details" section.

An extremely entertaining play by Shakespeare. The slapstick humour is refreshing!

Reviewer1 Affiliation1

★★★★★

Absolutely fun and entertaining. The play lacks thematic depth when compared to other plays by Shakespeare.

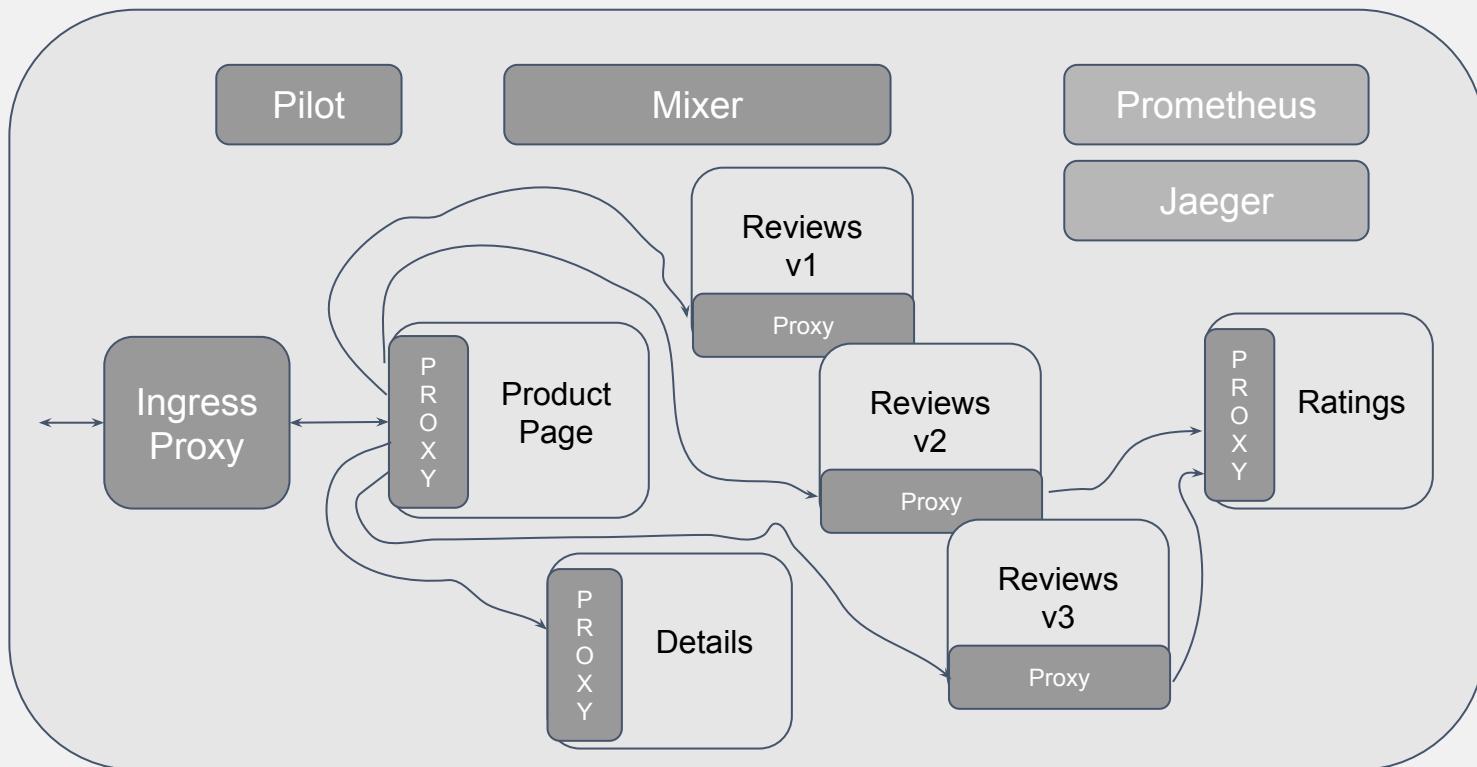
— Reviewer2 Affiliation2

★★★★★

precedence: 1
route:
- tags:
 - version: v1
 - weight: 100
Go refresh the page

A small icon of a lightbulb is located at the bottom left of the page.

SAMPLE APP: “BookInfo”



LAB: DETECTING AND PREVENTING ISSUES IN DISTRIBUTED APPS WITH ISTIO

SCENARIO 7

PREVENT AND DETECT ISSUES IN A DISTRIBUTED SYSTEM

WRAP-UP AND DISCUSSION

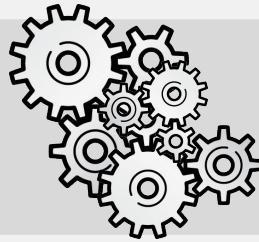
RESULT OF LAB

In this lab you learned:

- How to install Istio onto OpenShift Container Platform
- How to deploy apps with sidecar proxies
- How to generate and visualize deep metrics for apps
- How to alter routing dynamically
- How to inject faults for testing
- How to do rate limiting
- How Istio implements circuit breaking and distributed tracing
- Use cases for service mesh

MICROSERVICES 4.0?

Service



- > Autonomous
- > Loosely-coupled

Microservice



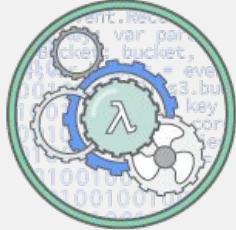
- > Single Purpose
- > Stateless
- > Independently Scalable
- > Automated

Function

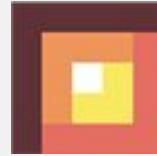


- > Single Action
- > Event-sourced
- > Ephemeral

SERVERLESS PROJECTS / SERVICES



APEX



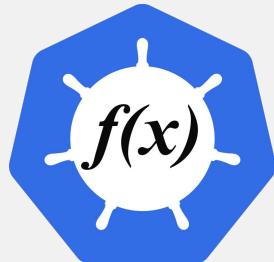
webtask

syncano



APACHE
OpenWhisk™

Iron.io



Back &
SERVERLESS

serverless-docker



<http://funcatron.org>

stdlib>

Microsoft Azure

fission

CLOUD FUNCTIONS BETA



redhat.

THANK YOU



plus.google.com/+RedHat



facebook.com/redhatinc



linkedin.com/company/red-hat



twitter.com/RedHatNews



youtube.com/user/RedHatVideos