

DevFest

Microservices Made Easier with ServiceMesh



Kusuma Seshavarapu, JDA Software
@skusumakumari





We are Living in a Digital Age



Changing Customer Demands

Always on,
Available from
Anywhere,
Anytime

Reduced TCO
Self-Service

Responsive
Personalized
Experience

Lots of New
Features all the
time



SOFTWARE IS
eating
**THE
WORLD**

*Marc
Andreessen*



NETFLIX



To Survive & Thrive Businesses need:



SPEED



AGILITY

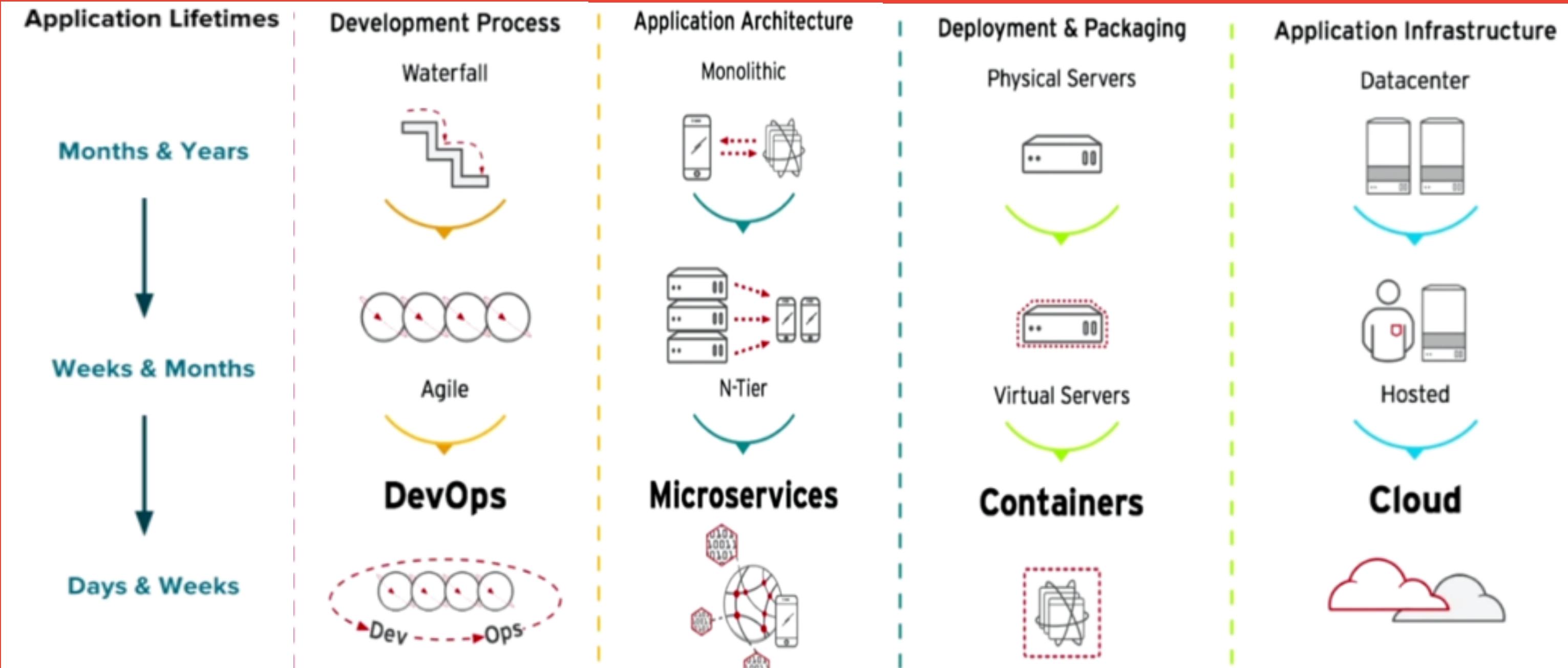


ROBUSTNESS



and DO IT ALL at
SCALE

Changing IT trends in the industry



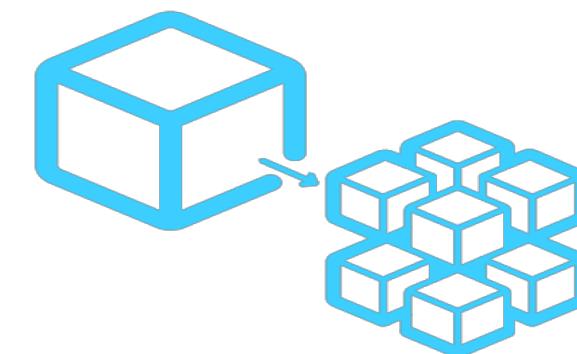
Roadmap to Digital Transformation



Remove
Hardware
Dependencies



Adopt
Containers &
CI/CD



Move to
Microservices



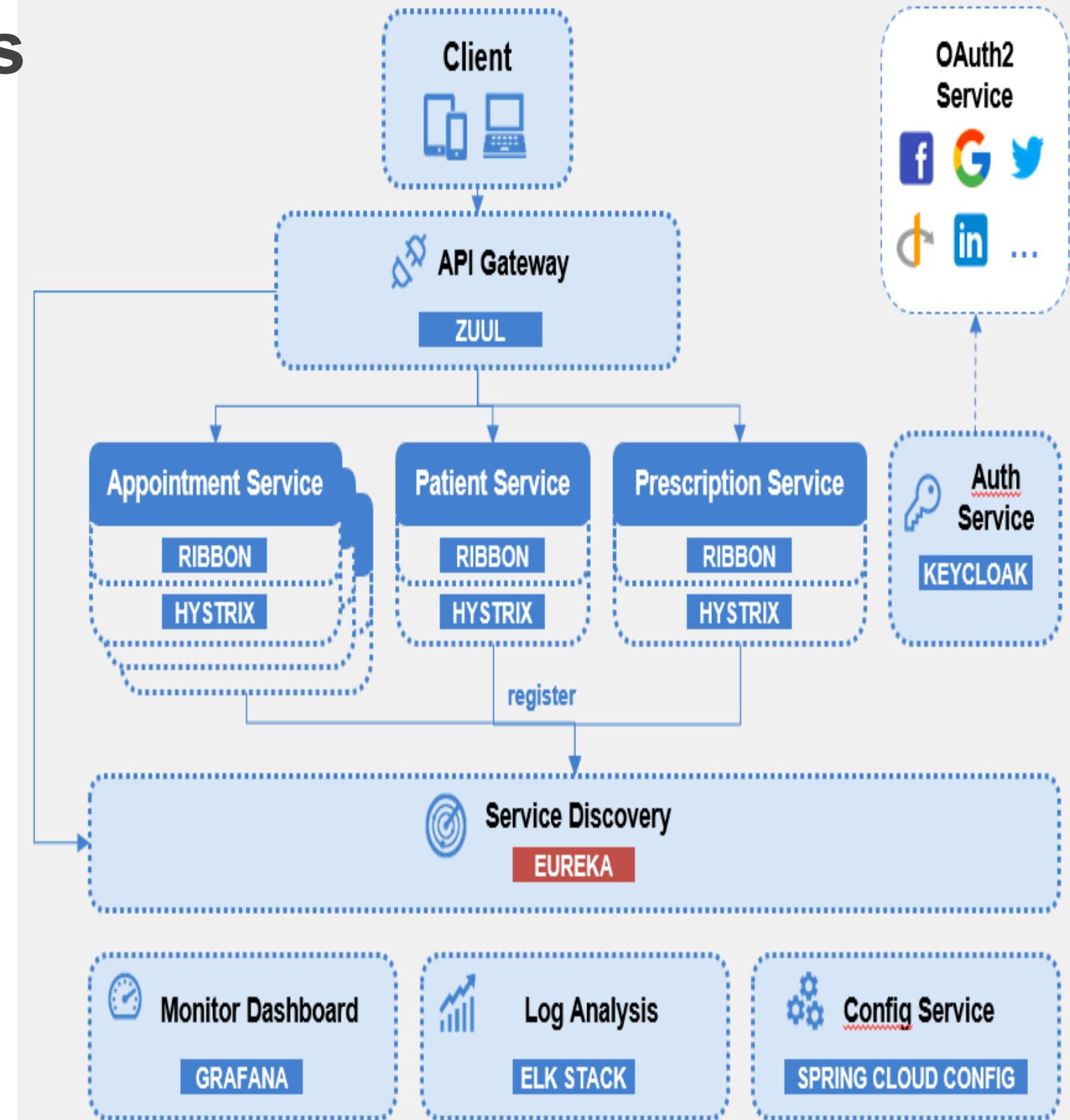


Microservices Architecture

“service-oriented architecture
composed of
loosely coupled elements
that have
bounded contexts”

Adrian Cockcroft
(former Cloud Architect at Netflix)

Components of Microservices Architecture

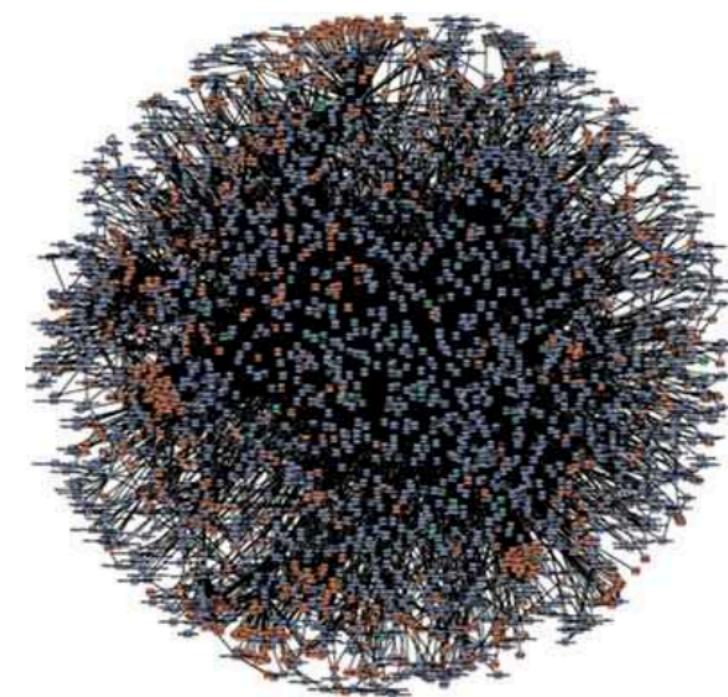


Advantages of Microservices

- **Continuous Delivery** - Faster and simpler deployments and rollbacks
- **Technology Diversity** - Right tool for the right problem
- **Loose Coupling** - can be versioned independently
- **Improved Fault Isolation** – Easier to fix
- **Independently Scalable services** – Scale based on real-world bottlenecks
- **Full Stack Autonomous Teams** - Speed & Agility



“Most complex challenge in realizing microservices is not building the services themselves, but how to establish reliable and observable communication between services”

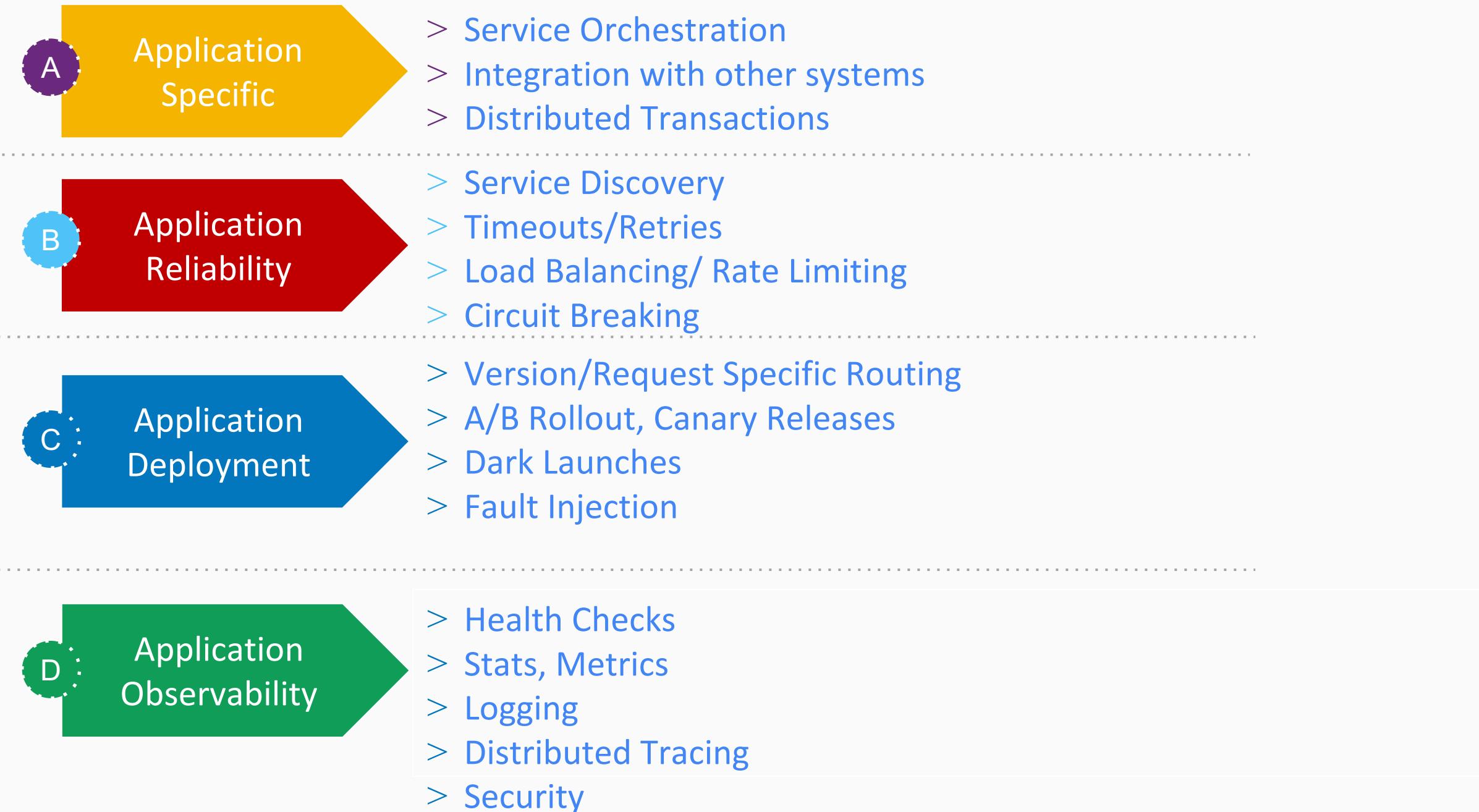


amazon.com



The Netflix logo is located in the bottom right corner of the screen. It consists of the word "NETFLIX" in a white, sans-serif font, with each letter having a thin black outline. The background behind the text is a solid red color.

Complexities of Microservices



Service Mesh

“A service mesh is a dedicated infrastructure layer for handling service-to-service communication in a transparent and language agnostic way“

It decouples development and operations for services



Service Mesh Implementations



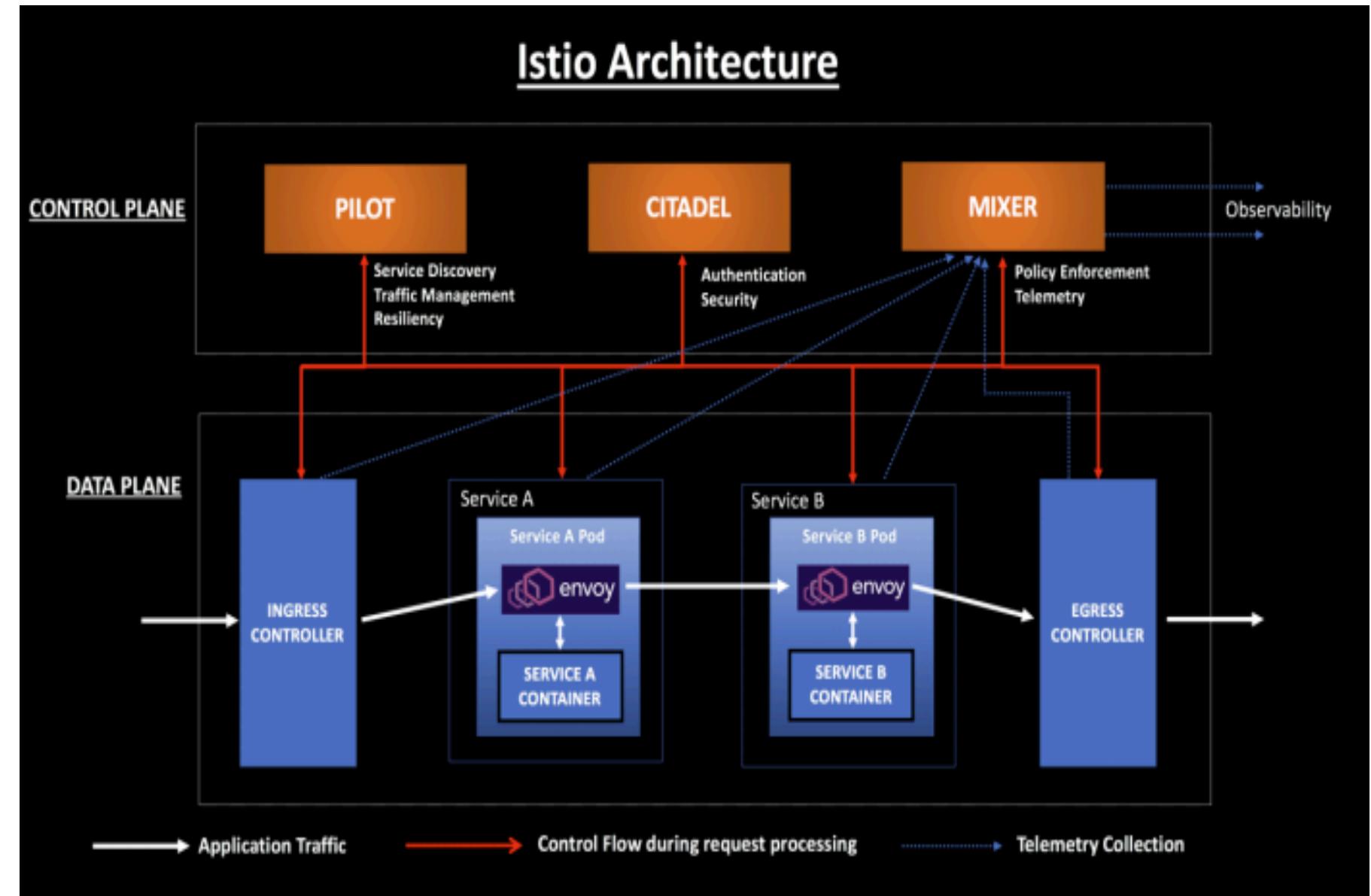
Components of Service Mesh(Istio Example)



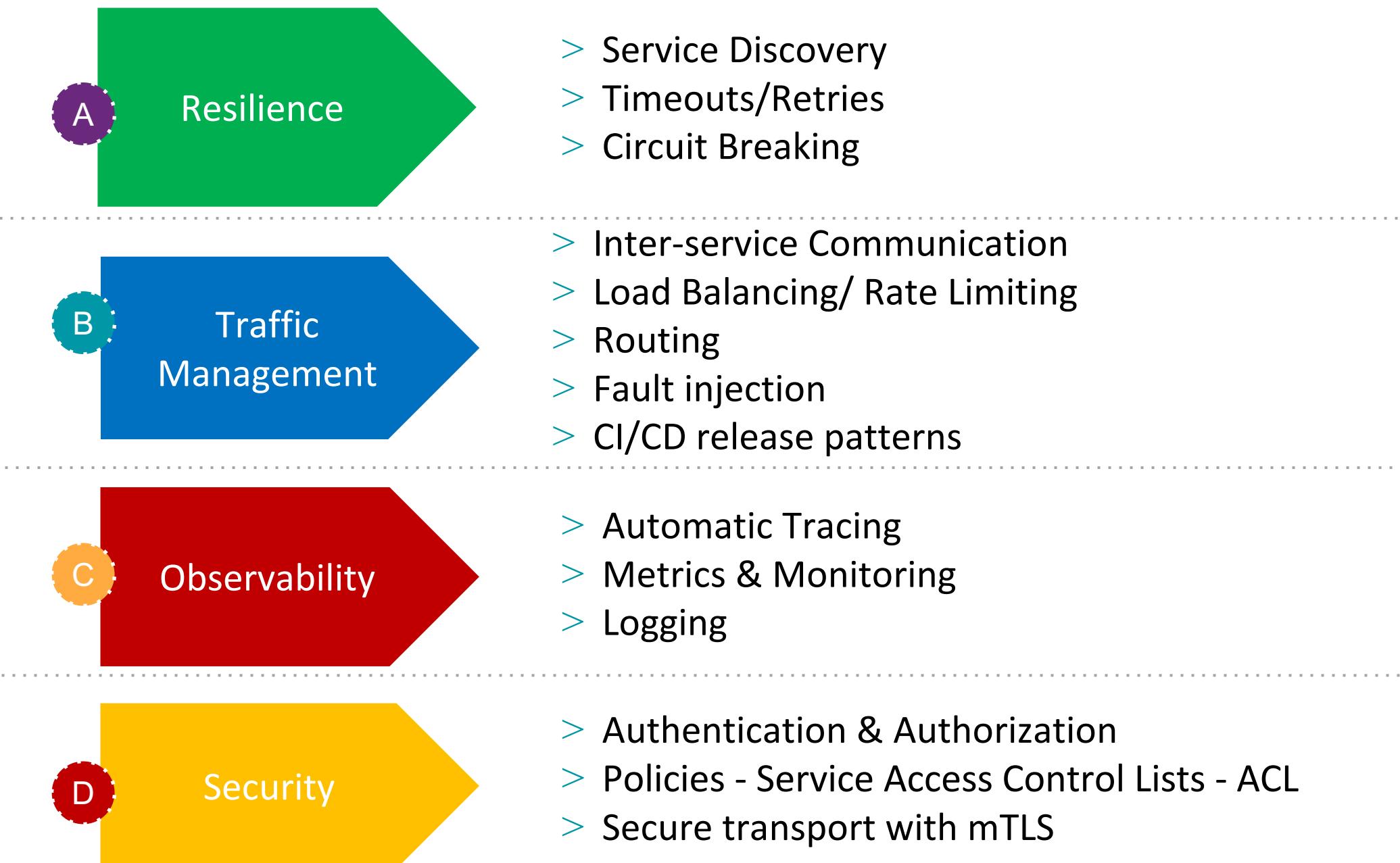
- Configure Service Proxies
- Controls Data Plane



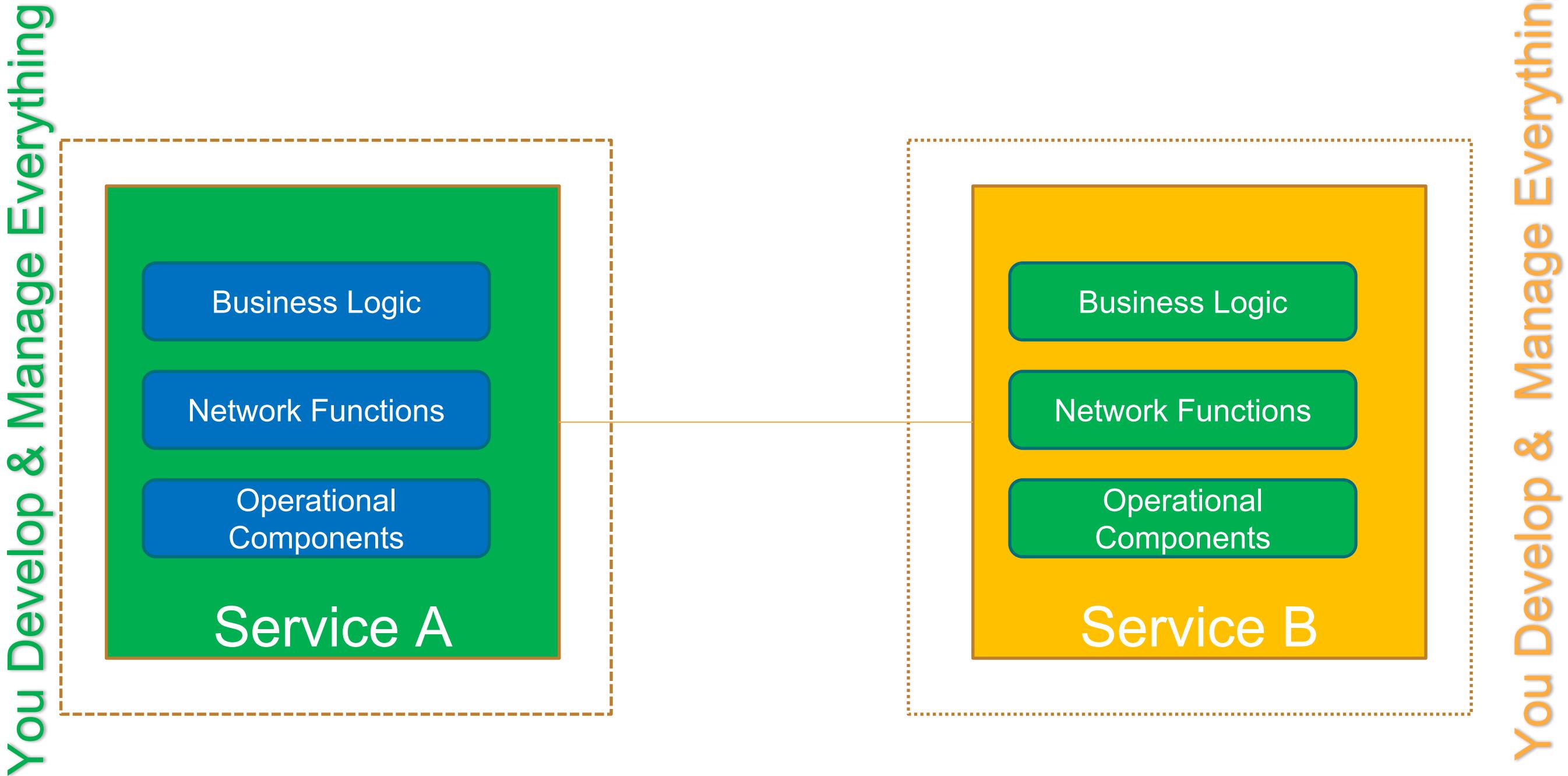
- Traffic Management
- Health checking
- Load balancing
- Circuit breaking
- Timeouts & Retries
- Security



Service Mesh Features

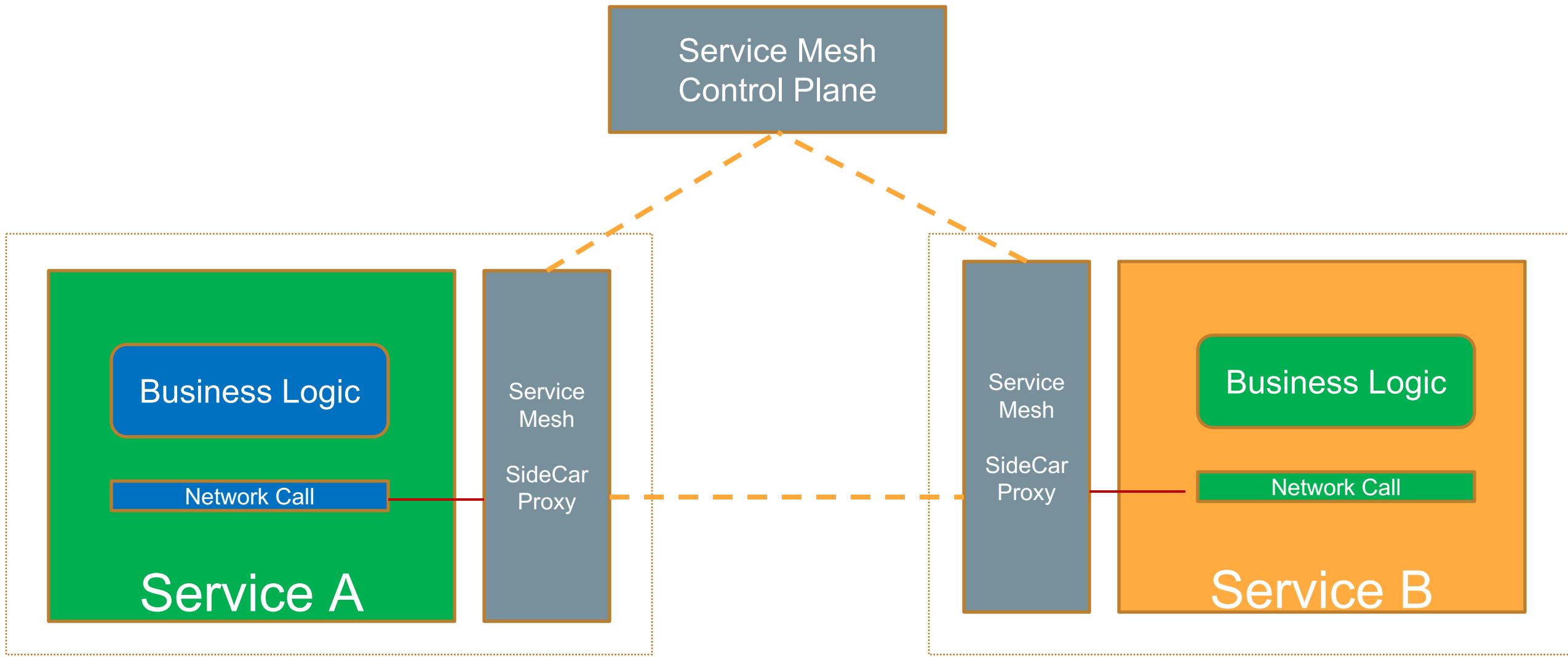


Microservices without Service Mesh



Microservices with Service Mesh

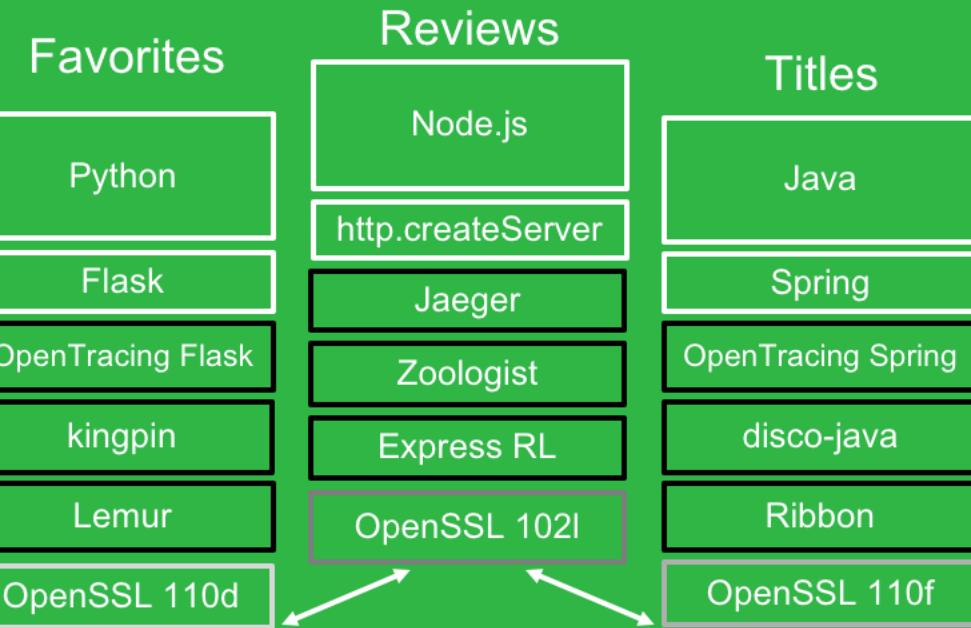
You Focus on Business Logic



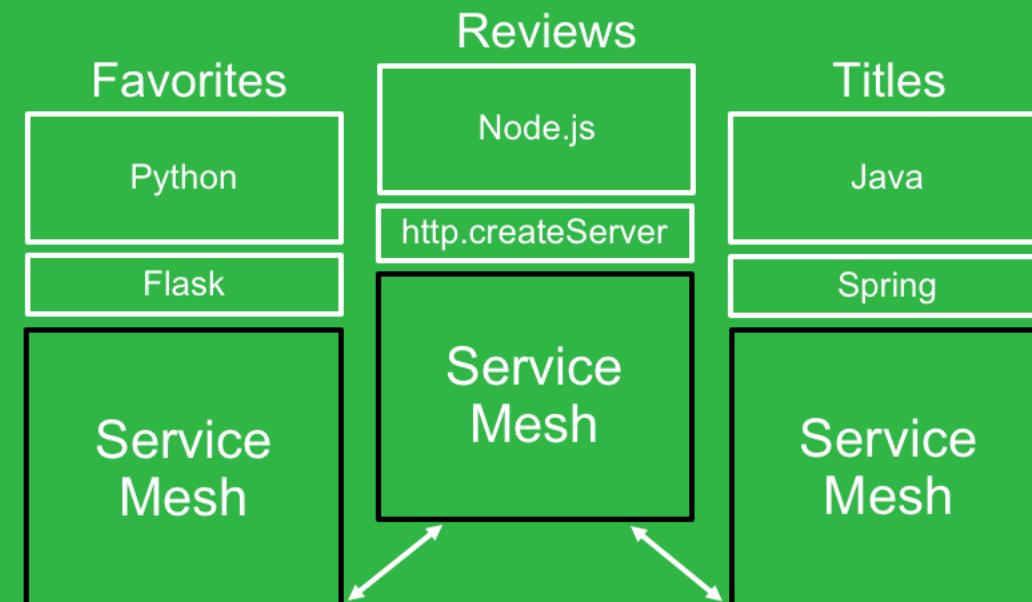
You Focus on Business Logic

Delegate Networking & Operations to Service Mesh

Before Service Mesh

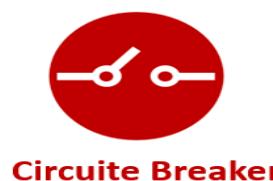


After Service Mesh



Advantages of Service Mesh

Solves most of the problems in Microservices



Circuite Breaker



Service Discovery



Timeouts and Retry



Security



Loadbalancing



Dynamic Routing



Access Control



Observability

All Without
Writing any
Code

Language Agnostic

Works well with polyglot microservices

Platform Agnostic

Works on Docker, Kubernetes, Mesos, Cloud, On-premise or any stand-alone machine

Disadvantages of Service Mesh

Adds extra hops: Each service call has to go through an extra hop(through service mesh sidecar proxy).

Does not address all the complexities of microservices: Only Internal Traffic is handled, Need to rely on API gateway for external traffic.

Resource intensive if lot of instances are used

Immature: Service mesh technologies are relatively new and evolving.

How to Inject Service Mesh Proxy?

Manually inject using the CLI commands provided by the service mesh

```
$ kubectl apply -f \<(istioctl kube-inject-f  
samples/bookinfo/platform/kube/bookinfo.yaml)}
```

Automate the injection using kubernetes initializers
`kubectl apply -f deployments/envoy-initializer.yaml`

Use Helm Chart



When to Adopt a Service Mesh?

When your application is composed of 20+ services

When you want to decouple application logic from networking logic

When there is a deep hierarchy of service calls

When you have Polyglot services

When the shared library approach does not work/scale

When you want to simplify the microservices development

DevFest

Thank you!



Kusuma Seshavarapu, JDA Software
@skusumakumari



DevFest

Q&A

