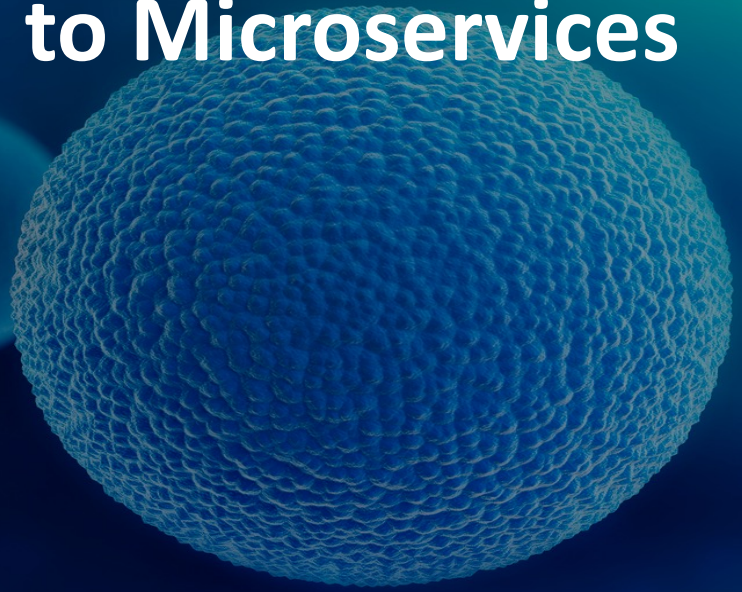
Mastering Chaos
A Netflix Guide to Microservices



Josh Evans – Engineering Leader

November 8, 2016

QCon SAN FRANCISCO

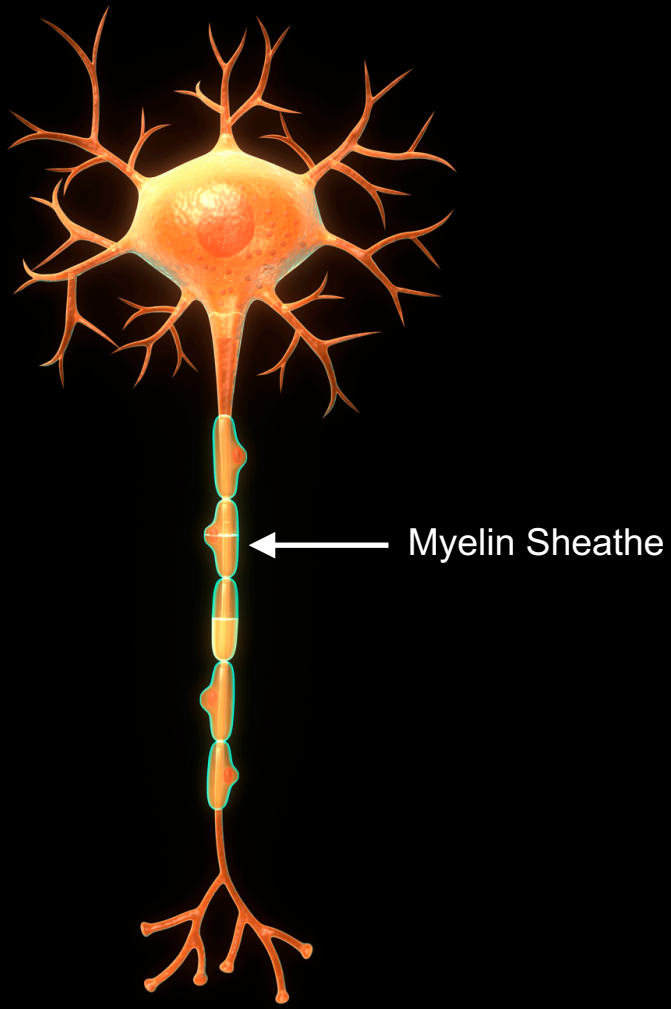
Illness in the Family

Guillain-Barré Syndrome

Autoimmune disorder

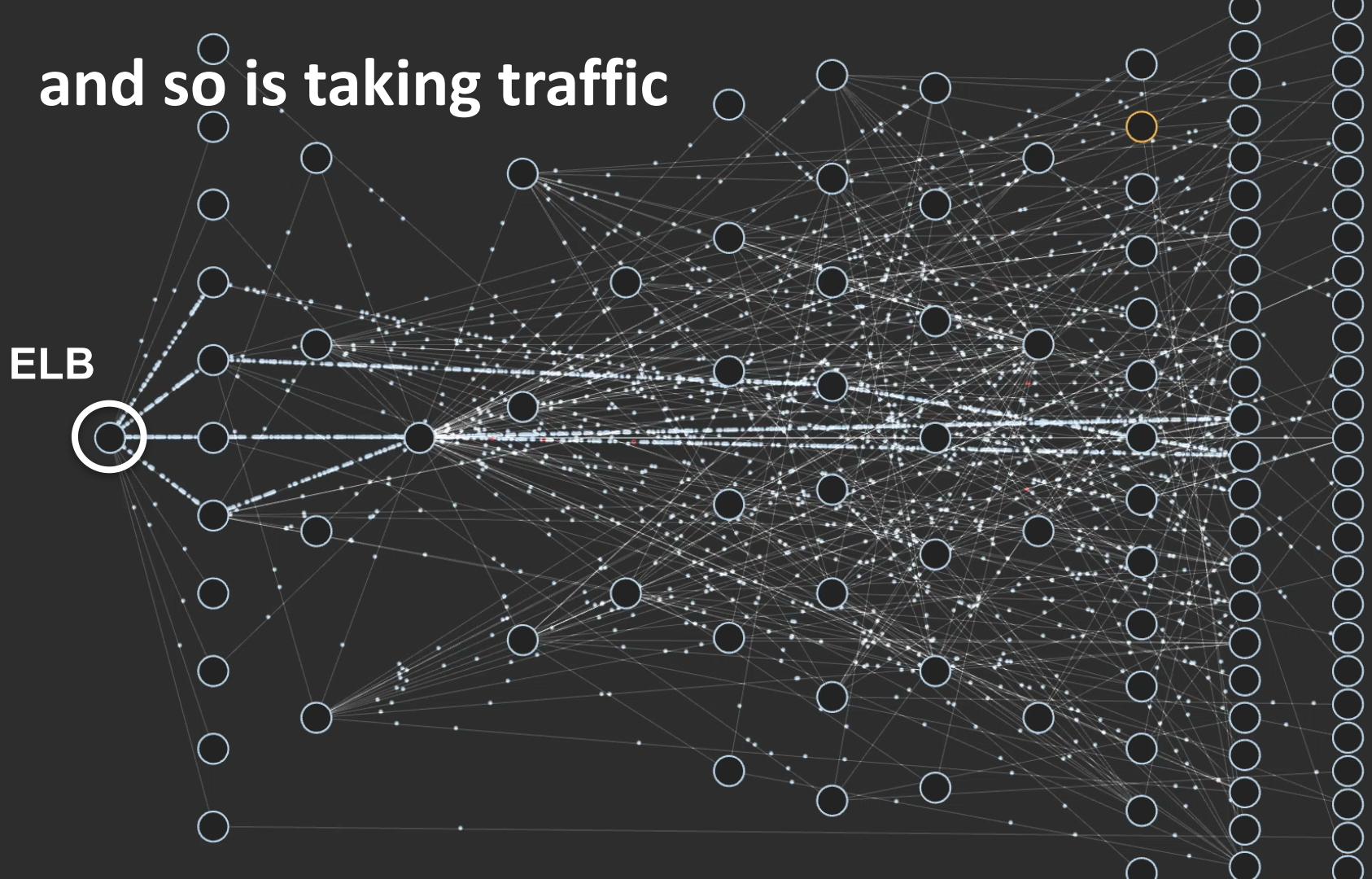
Externally trigger

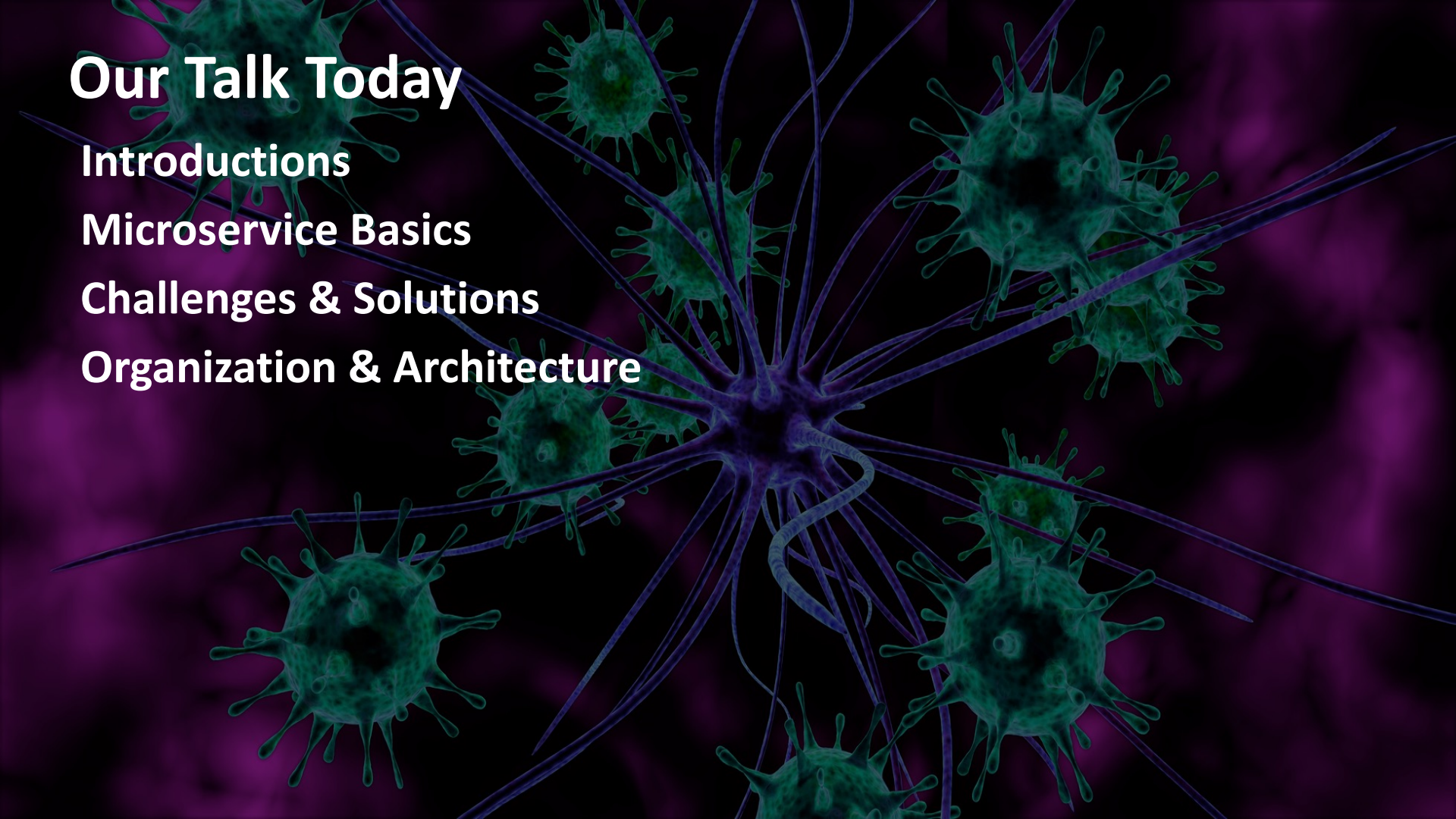
Treatable



Breathing is a miraculous act of bravery

and so is taking traffic





Our Talk Today

- Introductions
- Microservice Basics
- Challenges & Solutions
- Organization & Architecture

Our Talk Today

- Introductions
- Microservice Basics
- Challenges & Solutions
- Organization & Architecture

Josh Evans

1999 – 2009

- Engineer & Engineering Manager
- Ecommerce (DVD → Streaming)

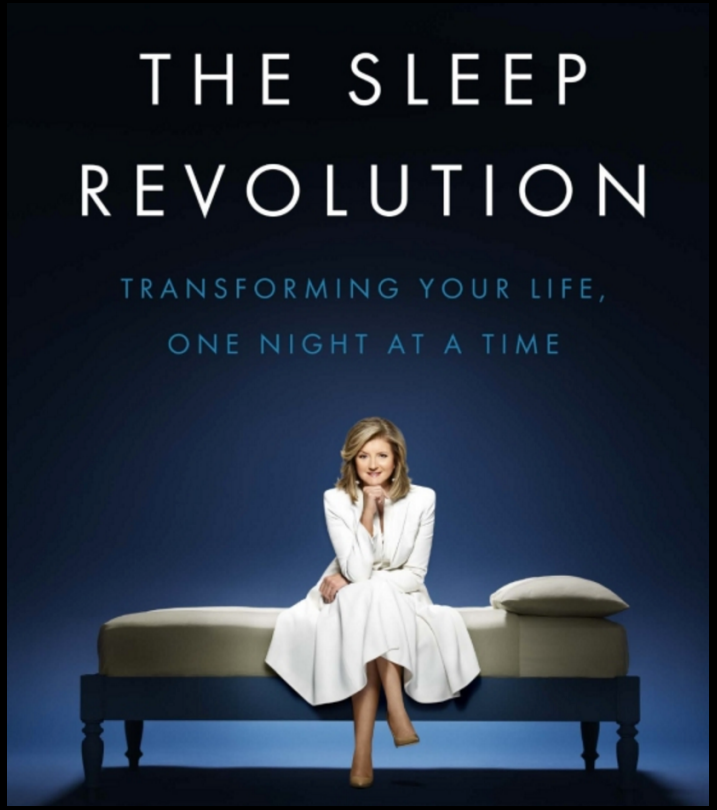
2009 – 2013

- Director of Engineering - Playback Services

2013 – 2016

- Director of Operations Engineering

Today



Taking time off

Spending time with family

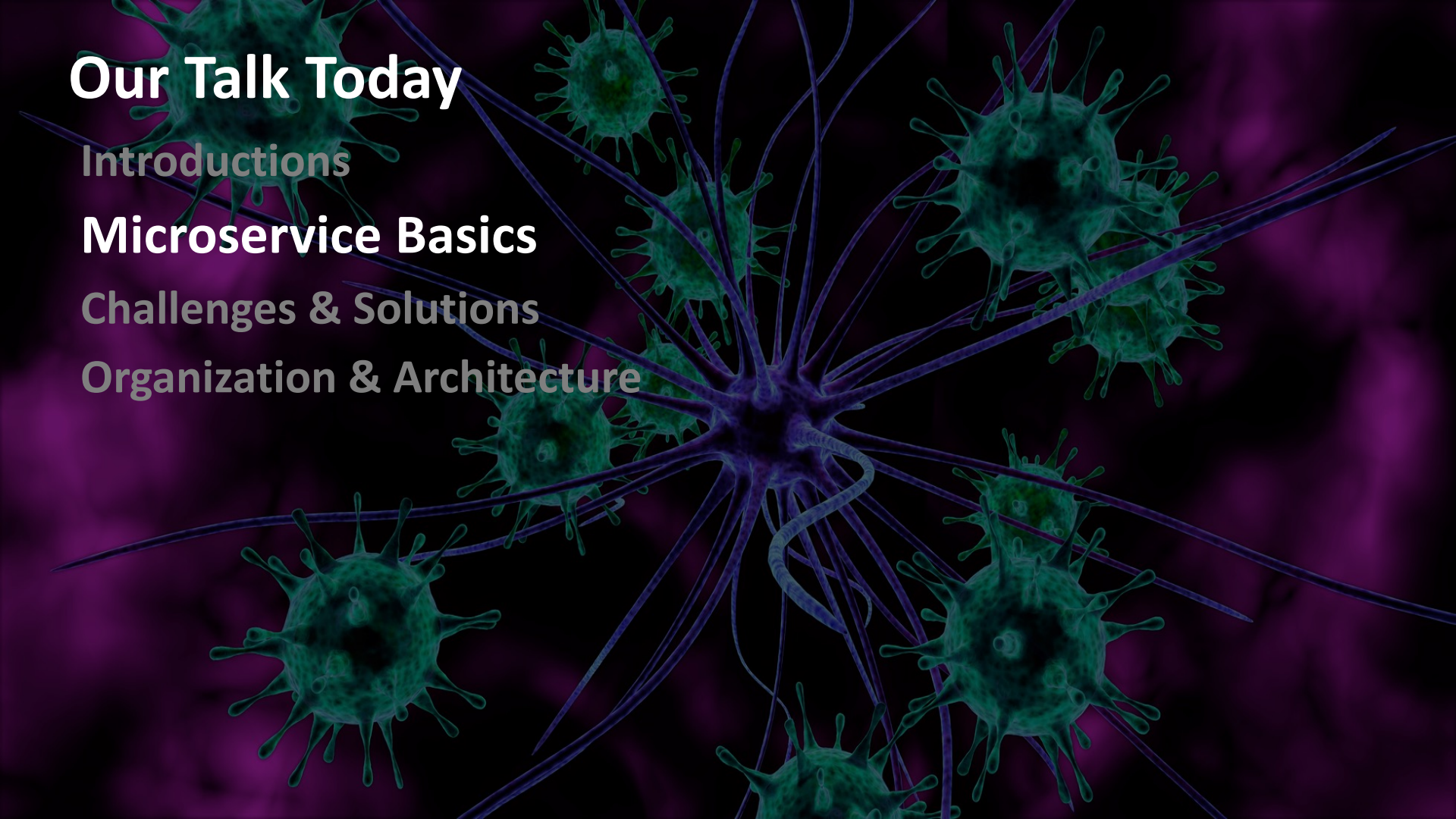
Thinking about what’s next

NETFLIX

Leader in subscription internet tv service
Hollywood, indy, local
Growing slate of original content

86 million members
~190 countries, 10s of languages
1000s of device types

Microservices on AWS



Our Talk Today

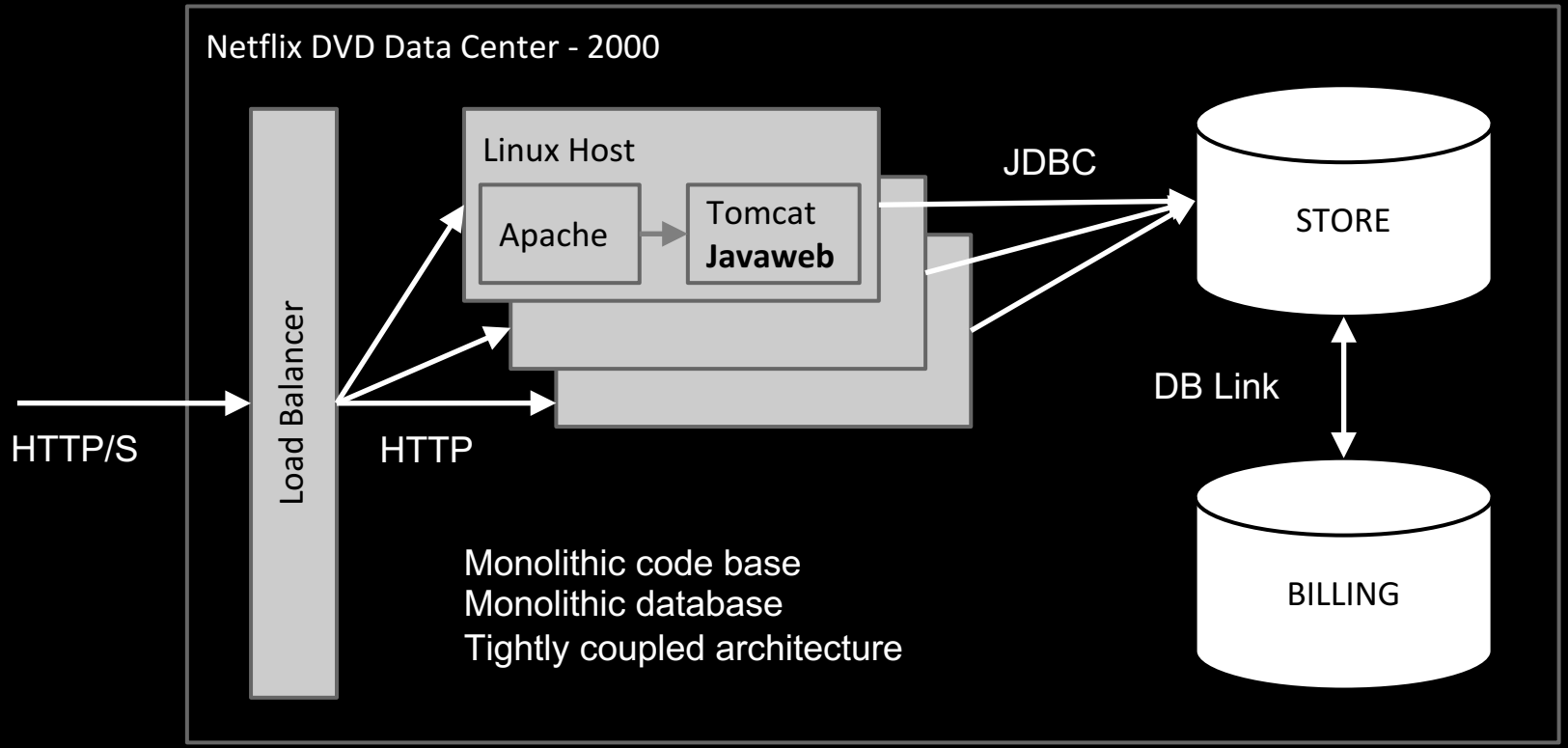
Introductions

Microservice Basics

Challenges & Solutions

Organization & Architecture

What microservices are not



What is a microservice?

…the microservice architectural style is an approach to developing a single application as a suite of small services, each running in its own process and communicating with lightweight mechanisms, often an HTTP resource API.

- Martin Fowler

An Evolutionary Response

Separation of concerns

- Modularity, encapsulation

Scalability

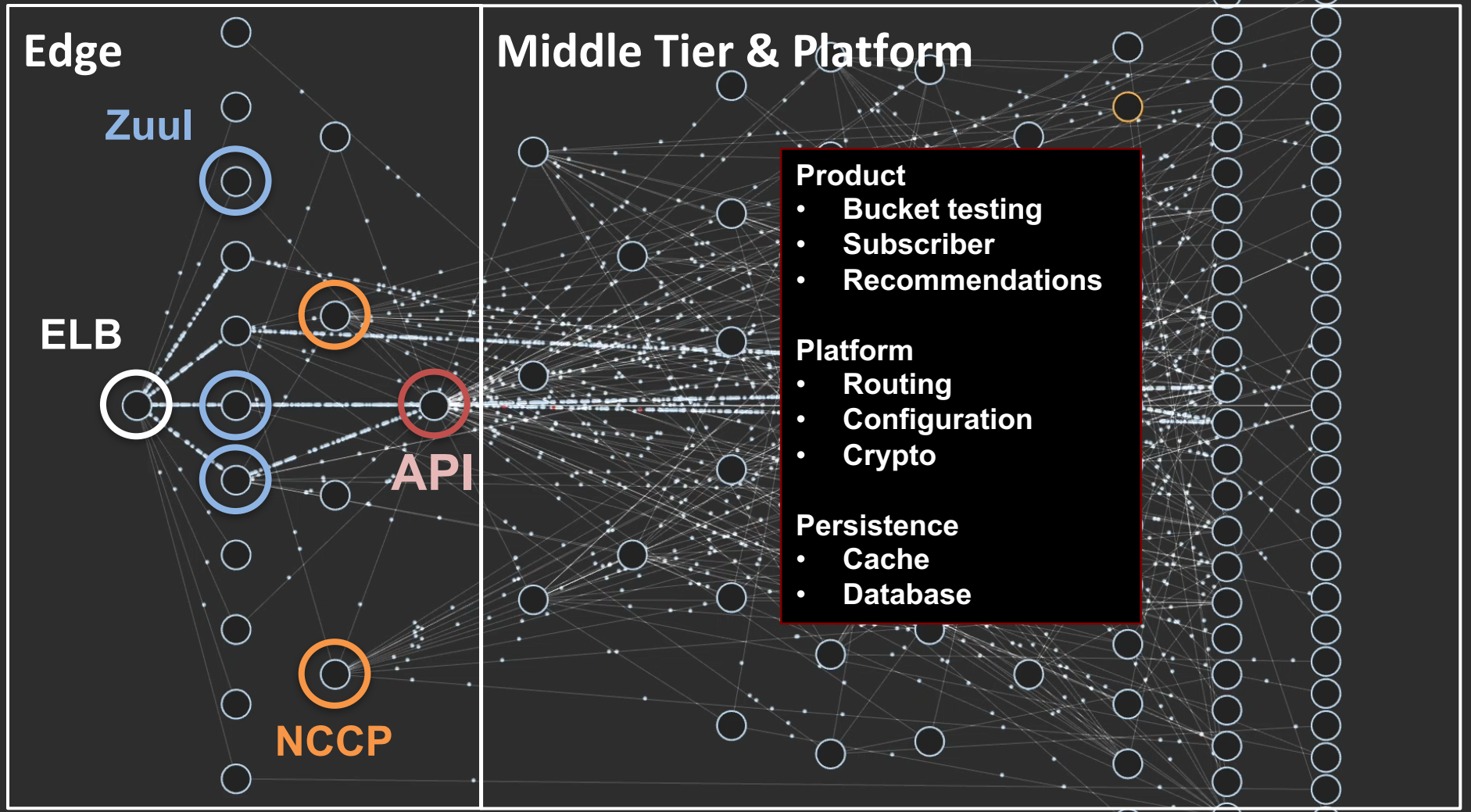
- Horizontally scaling
- Workload partitioning

Virtualization & elasticity

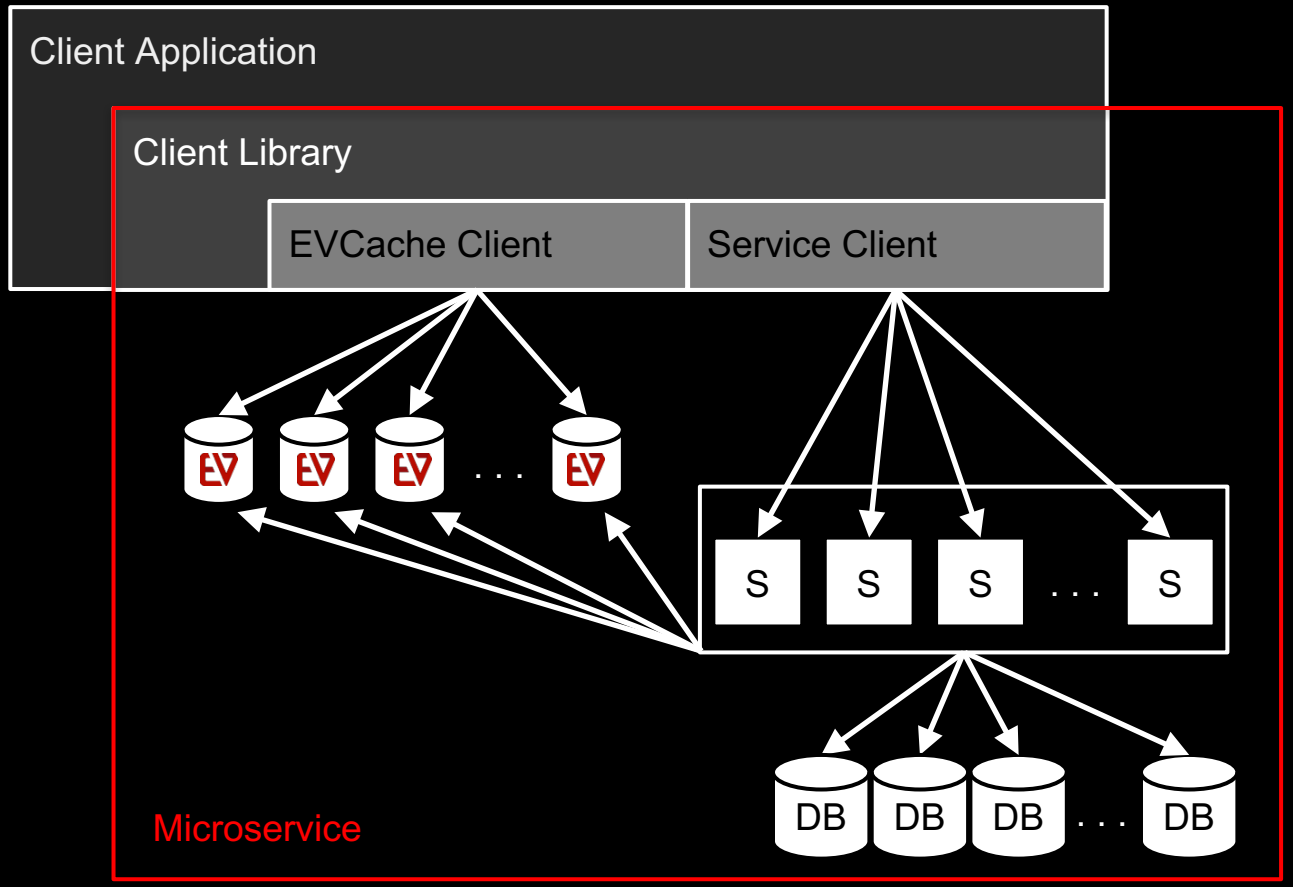
- Automated operations
- On demand provisioning

Organ Systems

- Each organ has a purpose
- Organs form systems
- Systems form an organism


Edge
Zuul
ELB
API
NCCP
Middle Tier & Platform
Product
• Bucket testing
• Subscriber
• Recommendations
Platform
• Routing
• Configuration
• Crypto
Persistence
• Cache
• Database

Microservices are an abstraction



Our Talk Today

Introductions

Microservice Basics

Challenges & Solutions

Organization & Architecture

Challenges & Solutions

Dependency

Scale

Variance

Change

Challenges & Solutions

Dependency

Scale

Variance

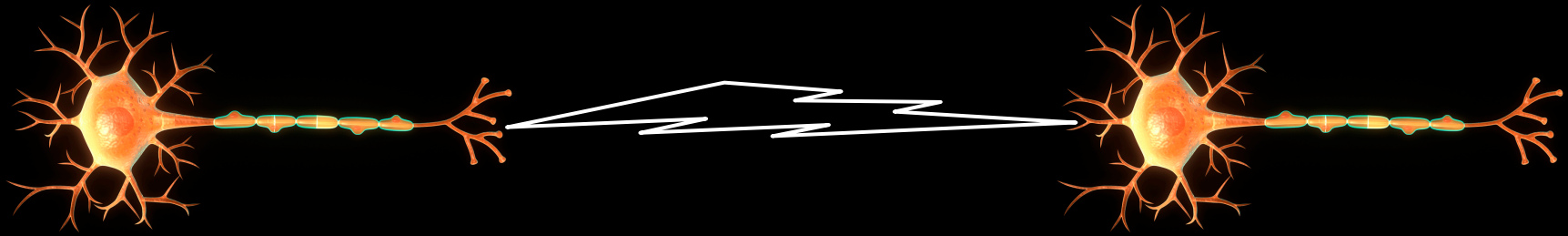
Change

Use Cases

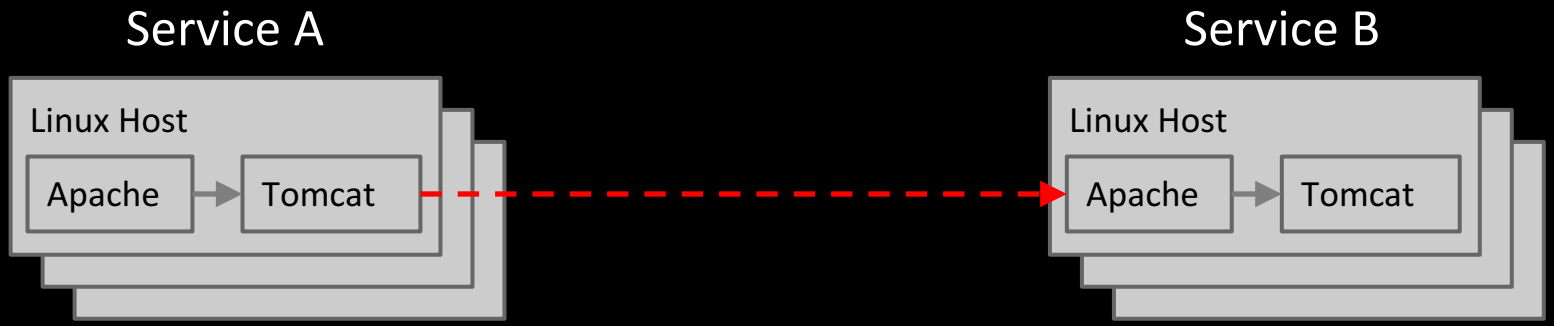
- Intra-service requests
- Client libraries
- Data Persistence
- Infrastructure

Intra-service Requests

Crossing the Chasm



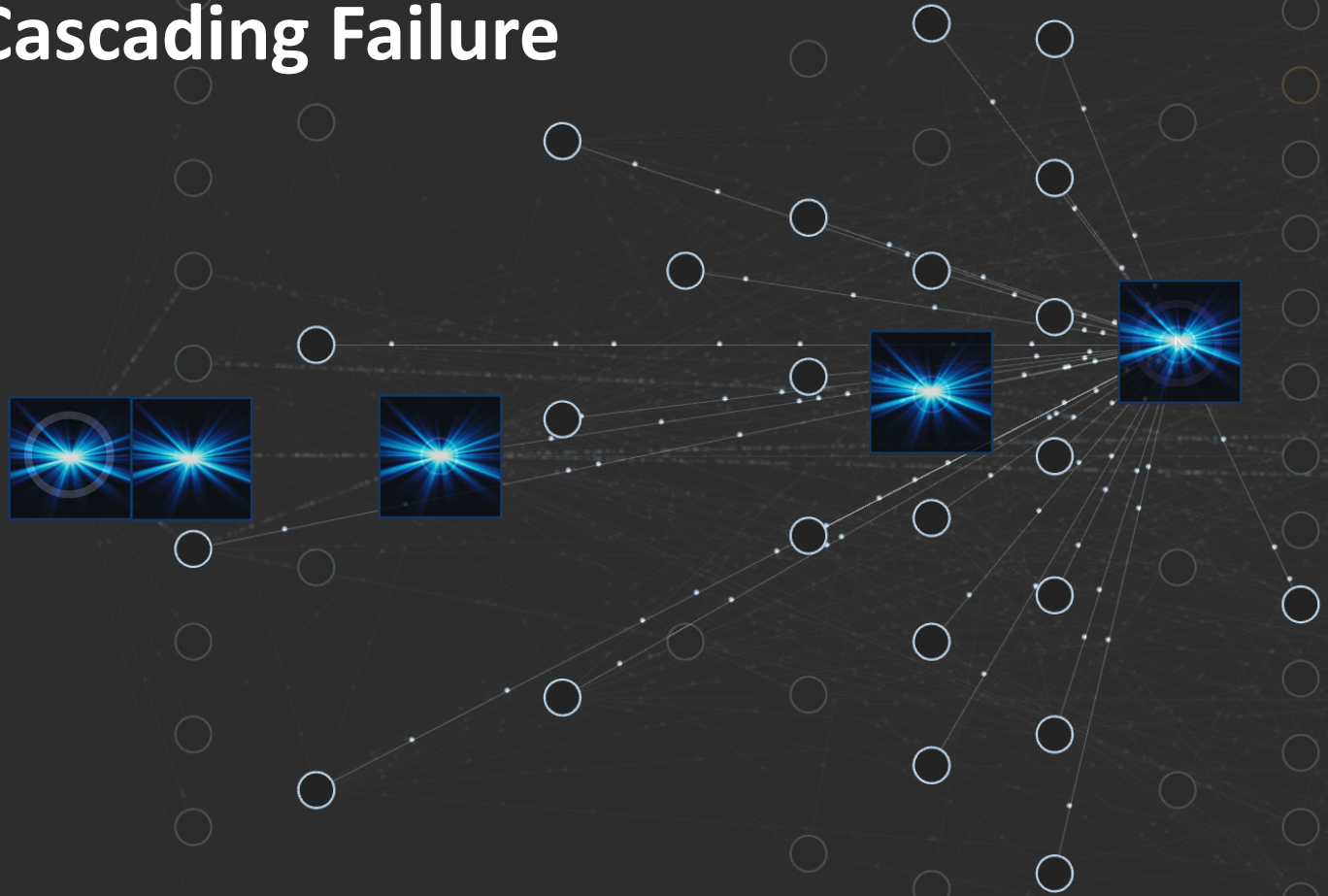
Crossing the Chasm



Network latency, congestion, failure

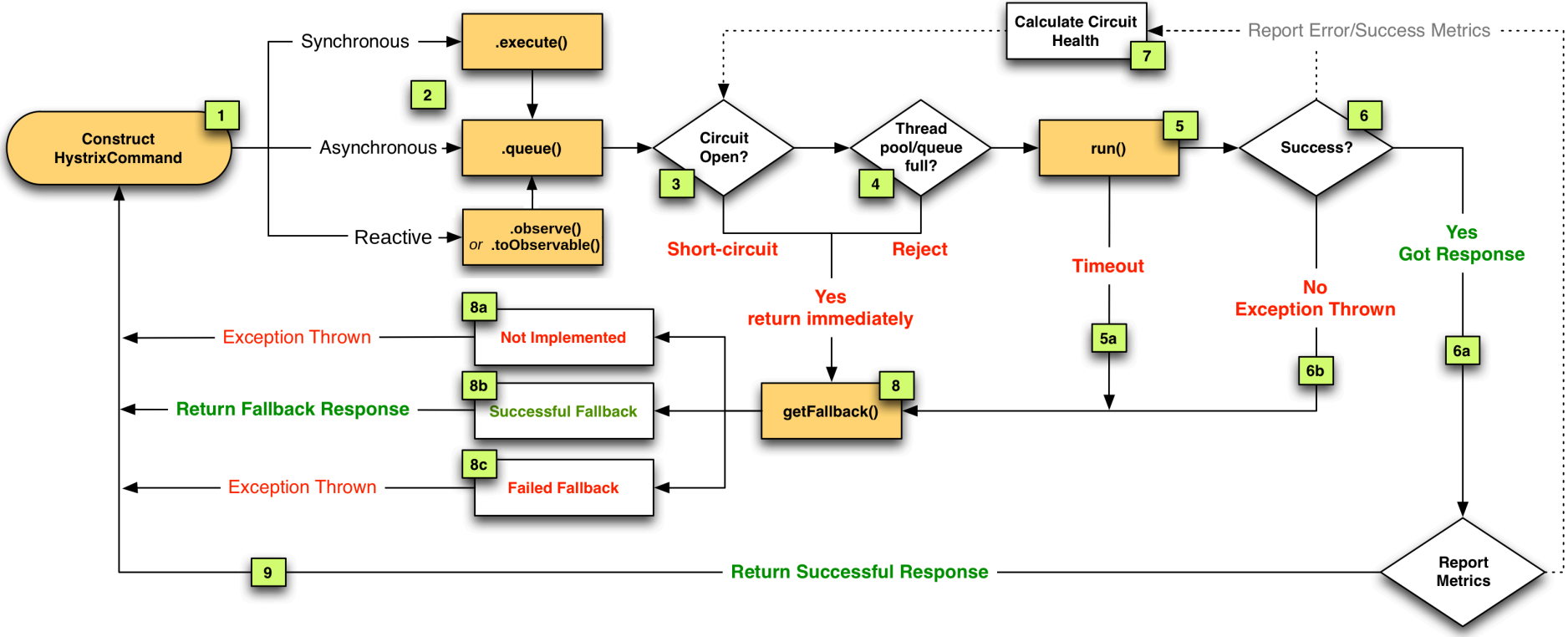
Logical or scaling failure

Cascading Failure



HYSTRIX

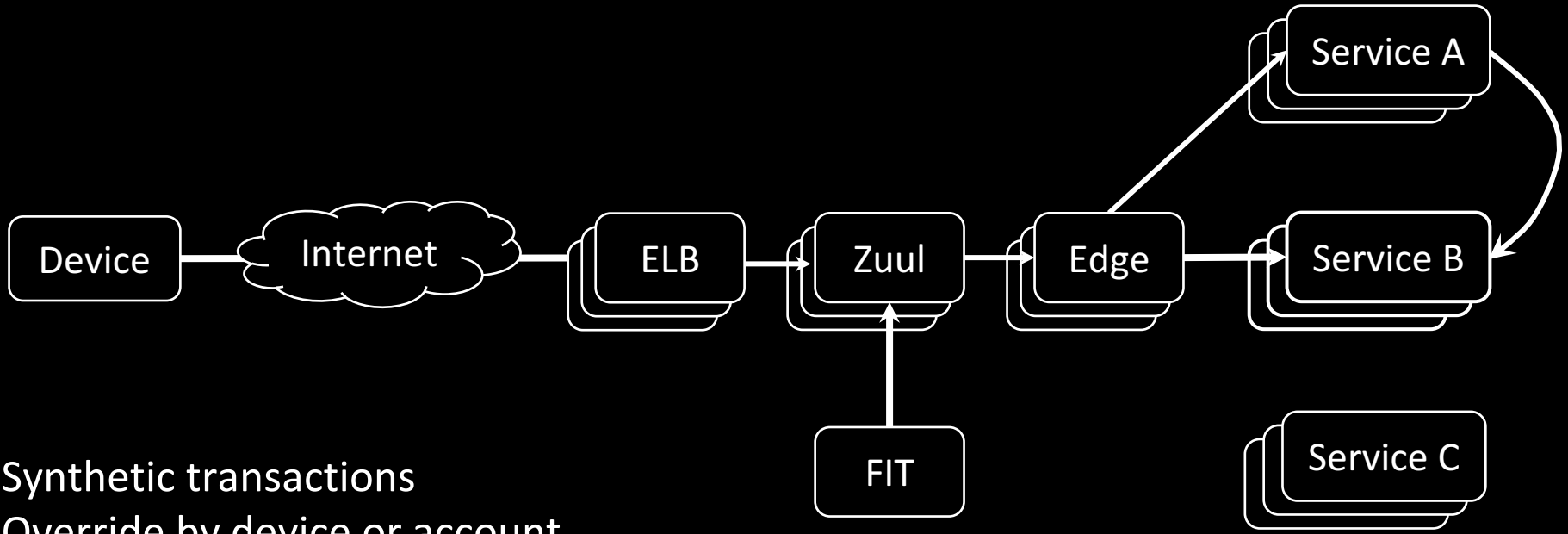
DEFEND YOUR APP



How do you know if it works?

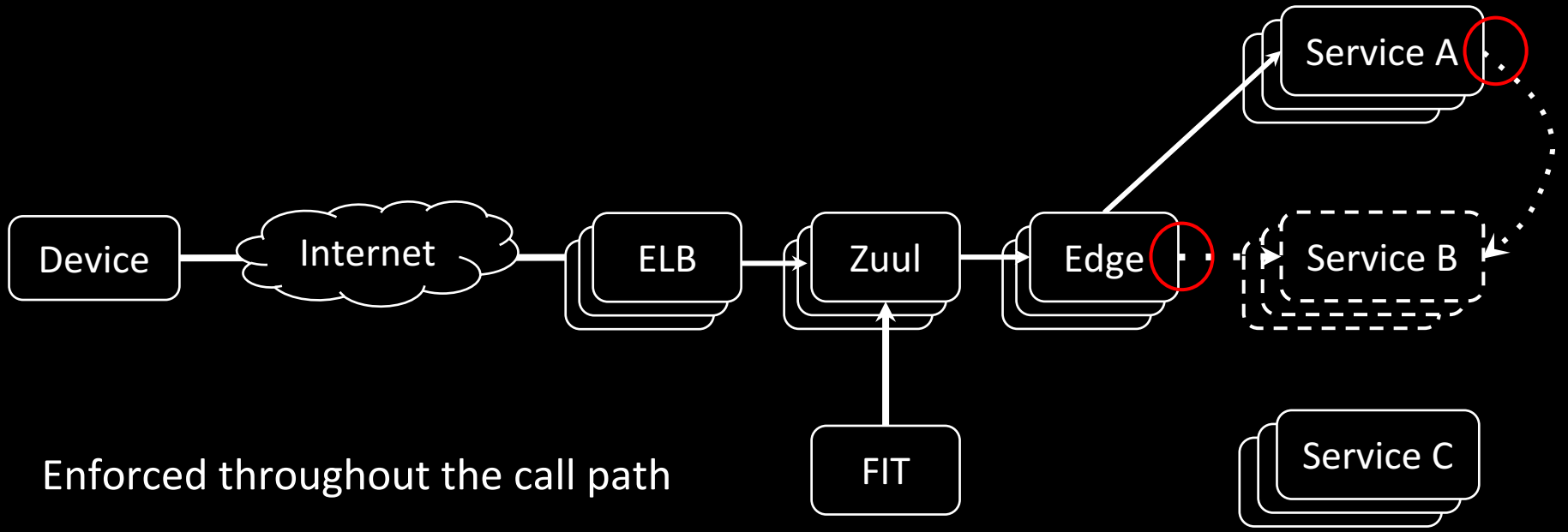
Inoculation

Fault Injection Testing (FIT)

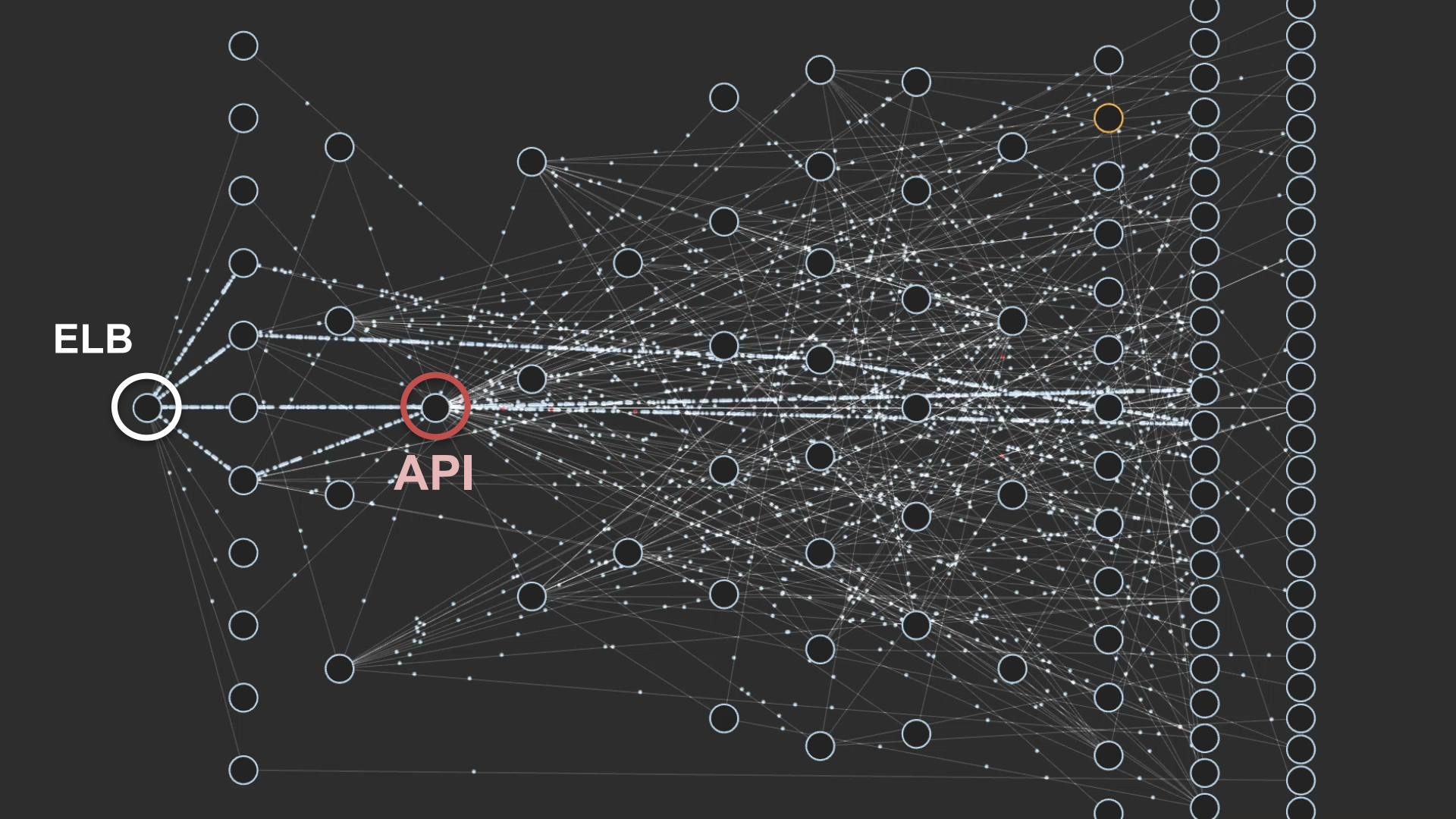


Synthetic transactions
Override by device or account
% of live traffic up to 100%

Fault Injection Testing (FIT)

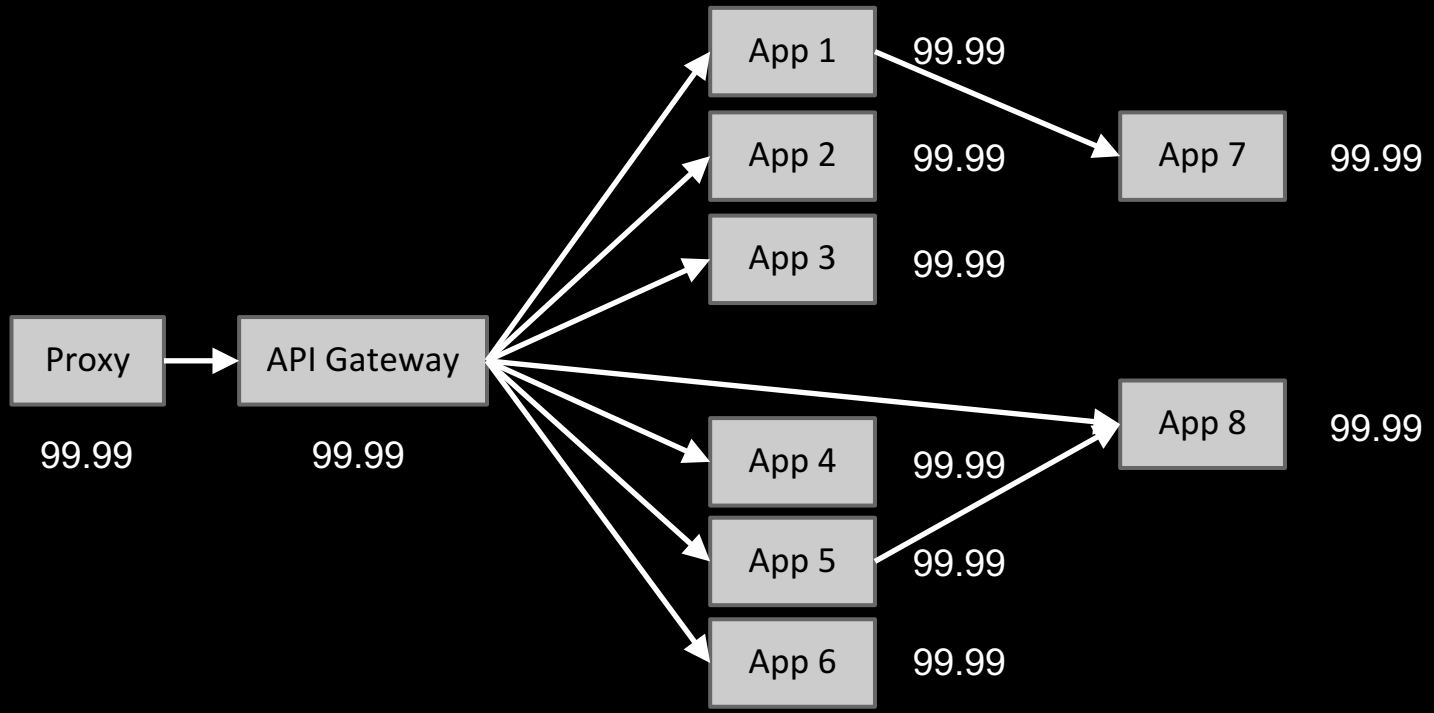


Enforced throughout the call path


ELB
API

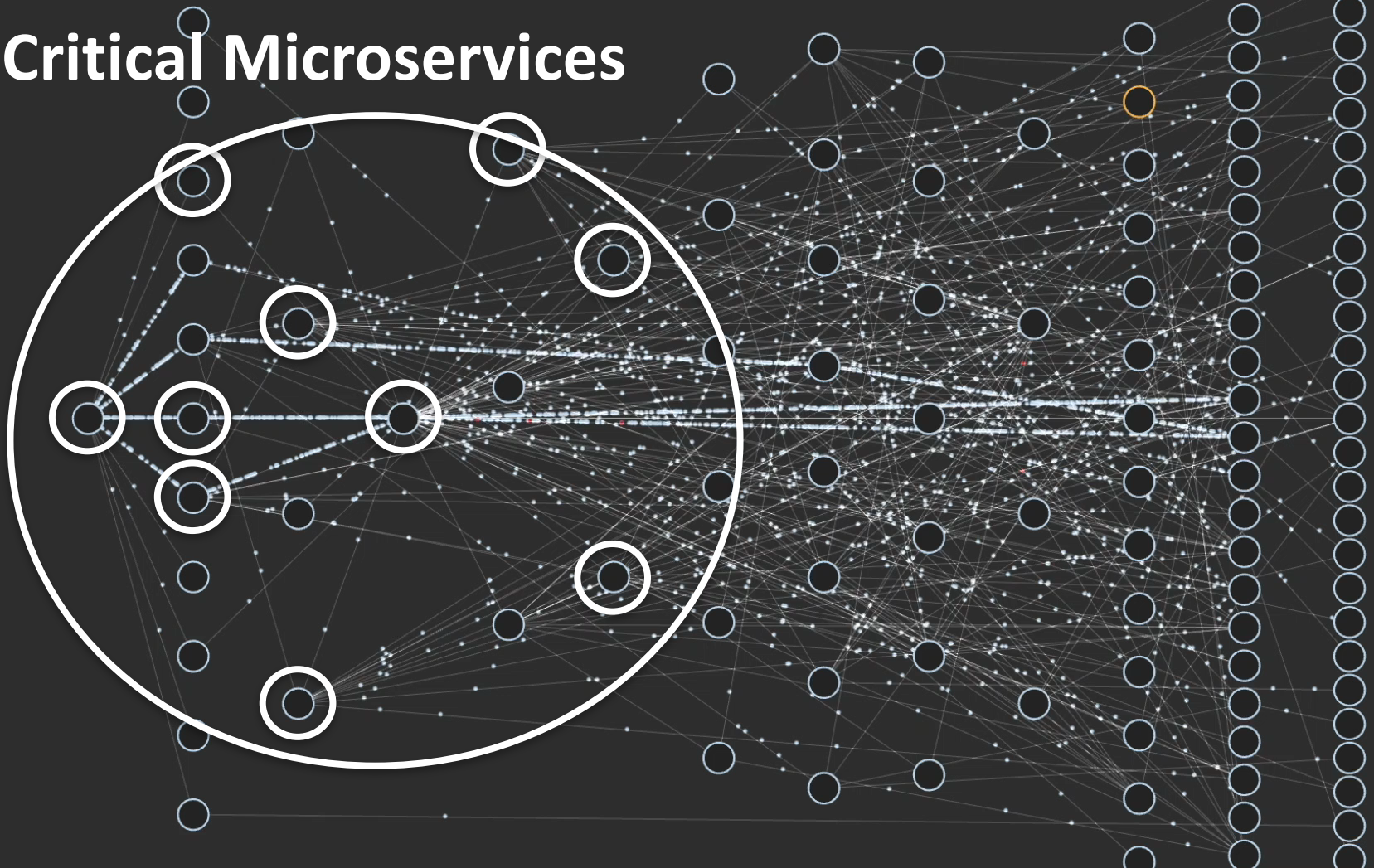
How do we constrain testing scope?

Combinatorial Math



$99.99^{10} = 99.9$

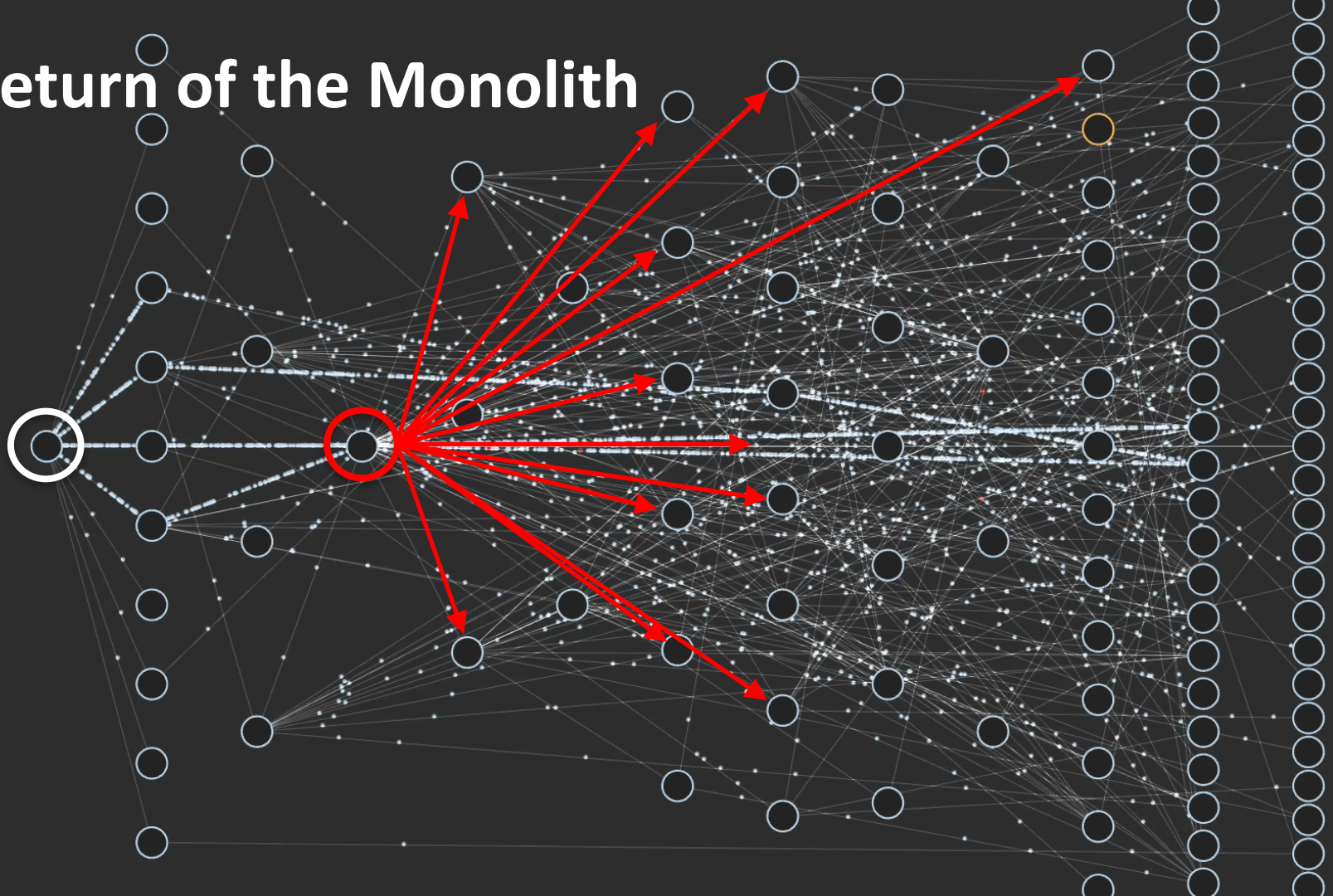
Critical Microservices



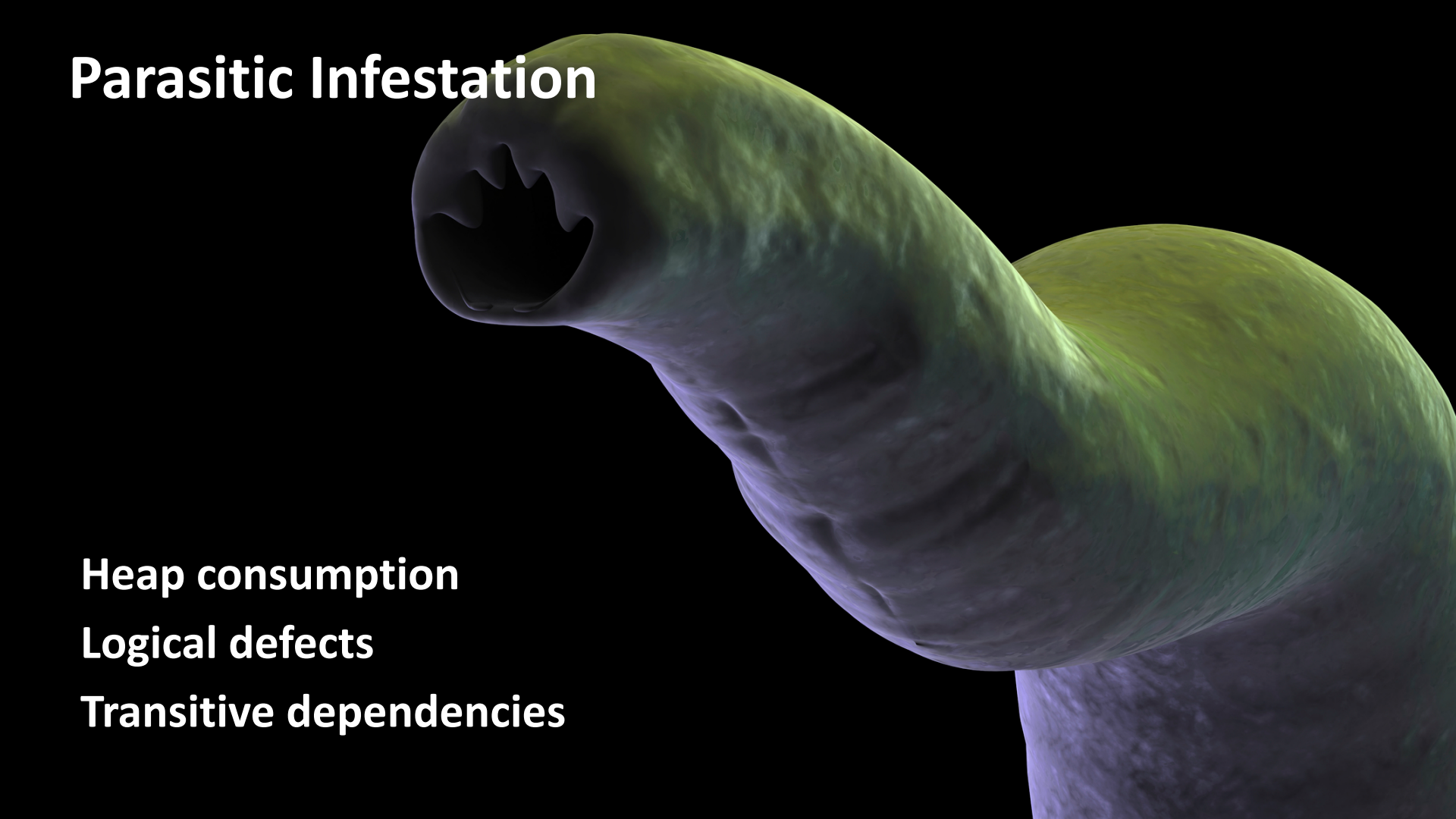
Client Libraries

- Many clients
- Common business logic
- Common access patterns

Return of the Monolith



Parasitic Infestation

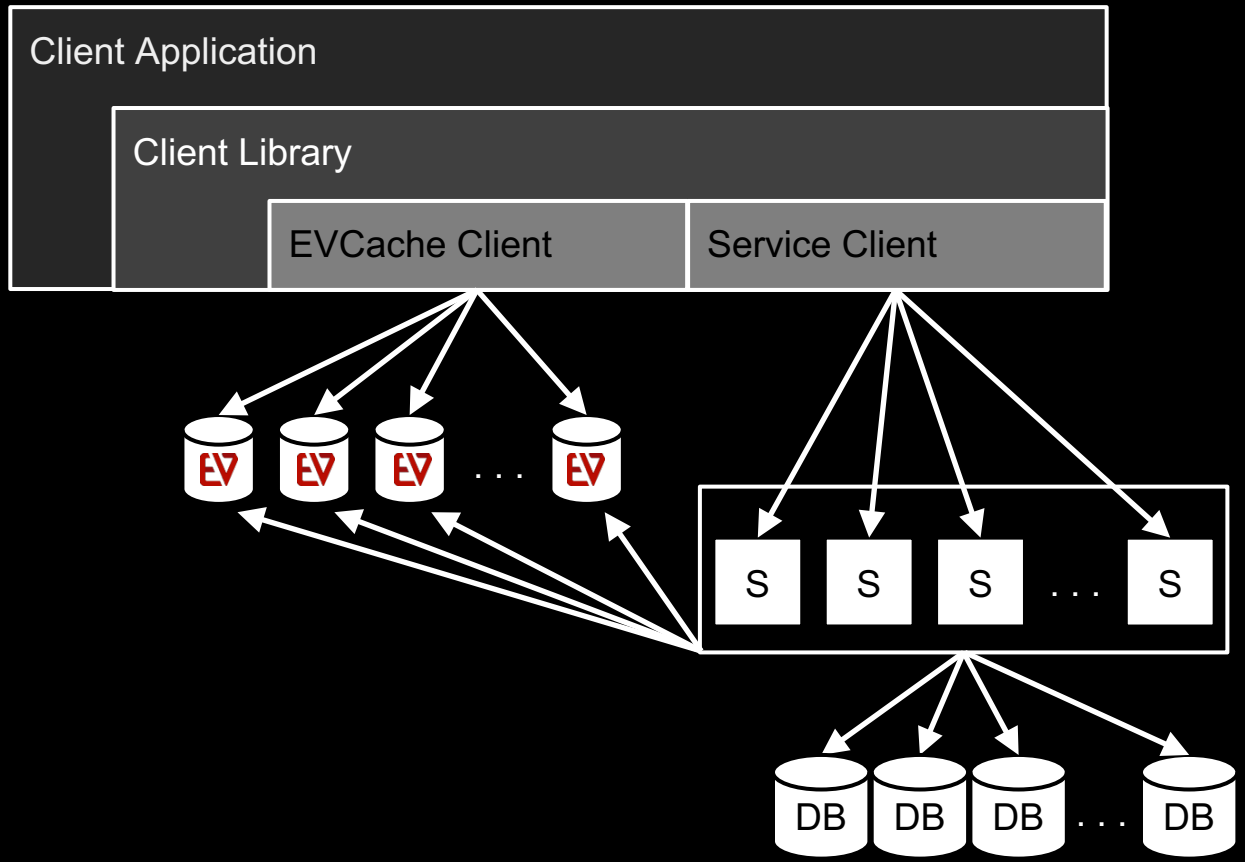


Heap consumption

Logical defects

Transitive dependencies

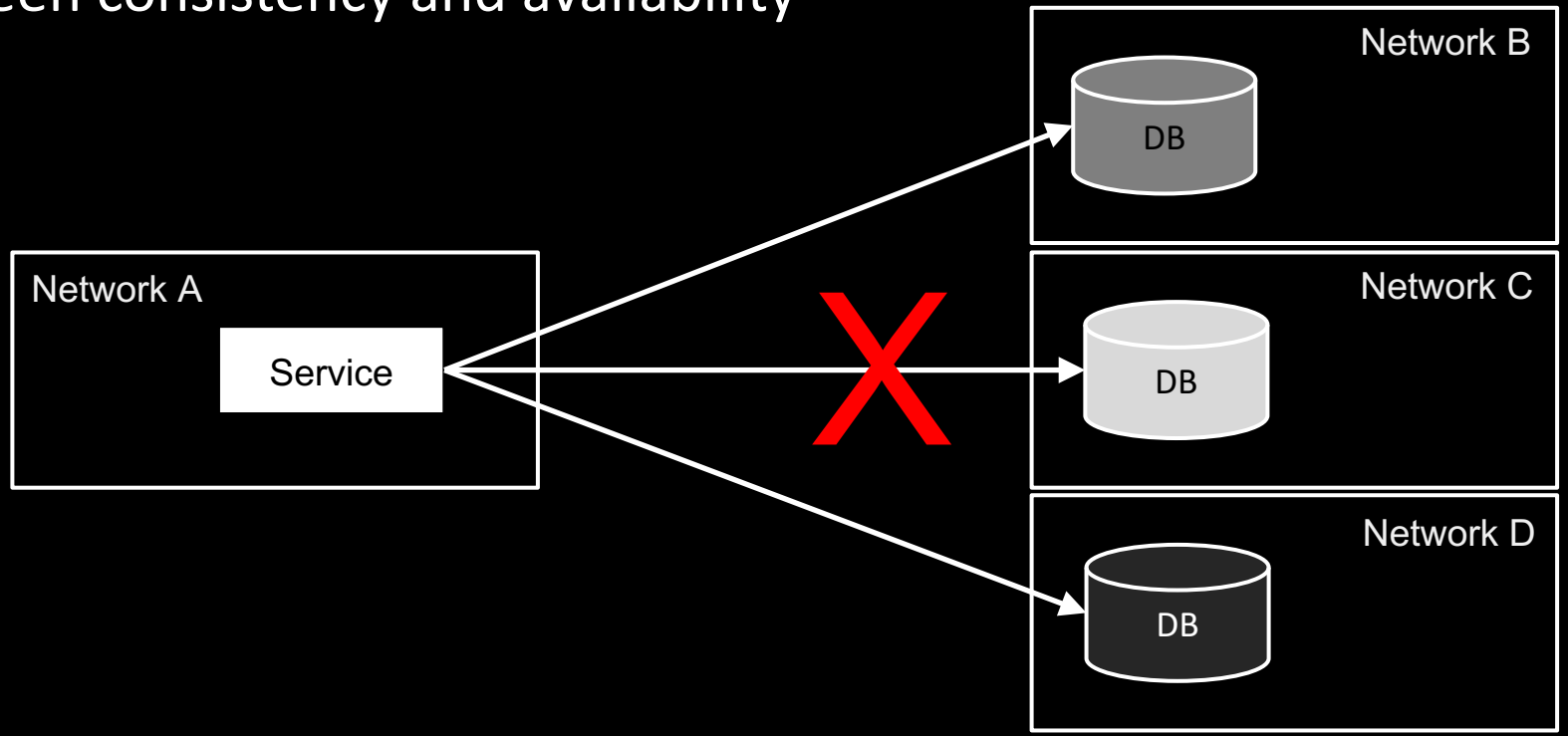
Simple Logic, Common Patterns



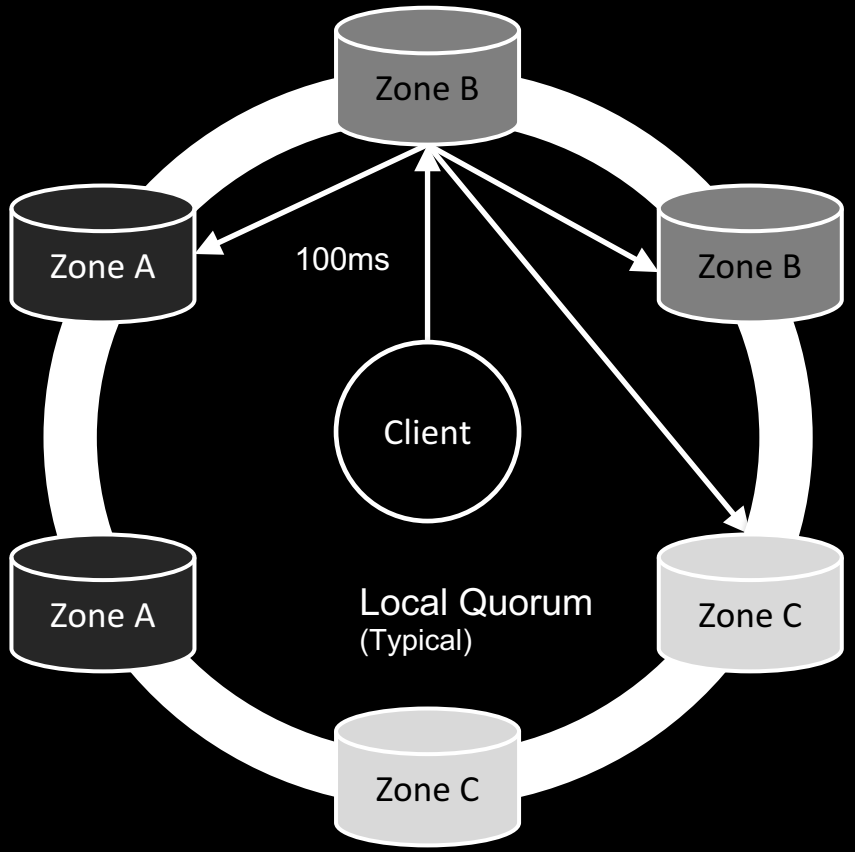
Persistence

CAP Theorem

In the presence of a network partition, you must choose between consistency and availability



Eventual Consistency



Infrastructure

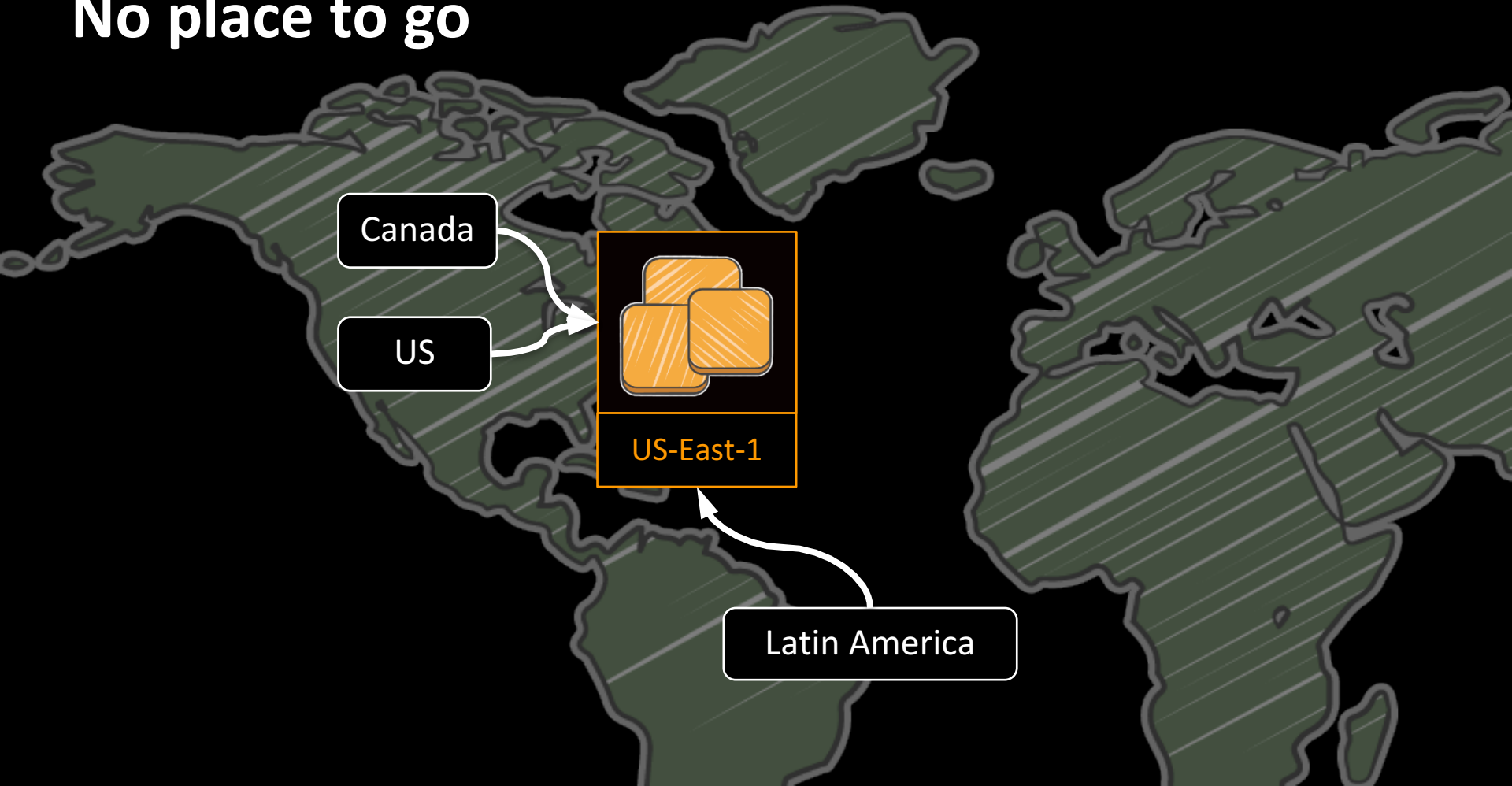
December 24th, 2012

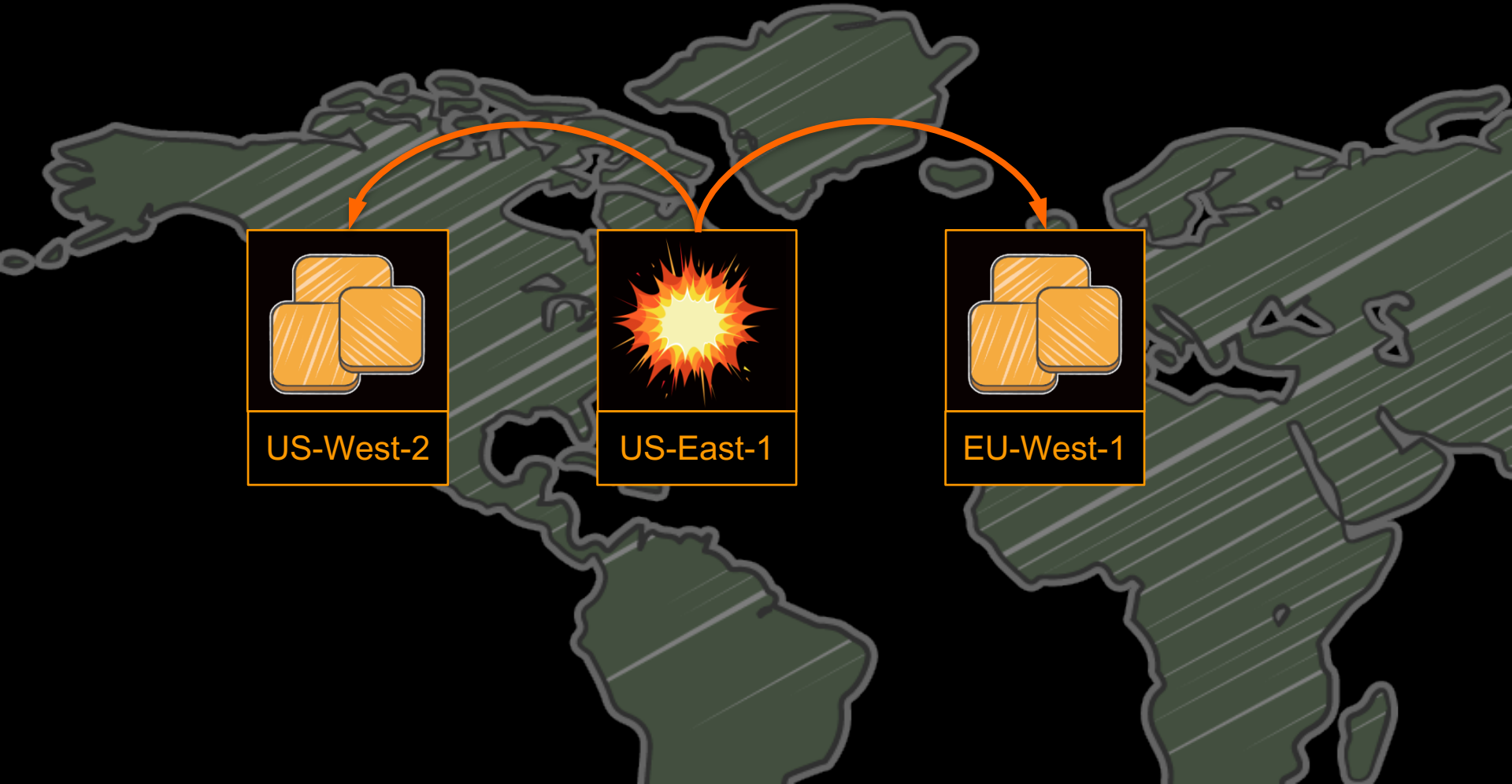
Forbes / Tech

DEC 24, 2012 @ 09:46 PM 105,201 VIEWS

Amazon AWS Takes Down Netflix On Christmas Eve

No place to go




US-West-2
US-East-1
EU-West-1

#NetflixEverywhere Global Architecture

QCon London, 2016

https://www.infoq.com/presentations/netflix-failure-multiple-regions

Challenges & Solutions

Dependency

Scale

Variance

Change

Use Cases

- Stateless services
- Stateful services
- Hybrid services

Stateless Services

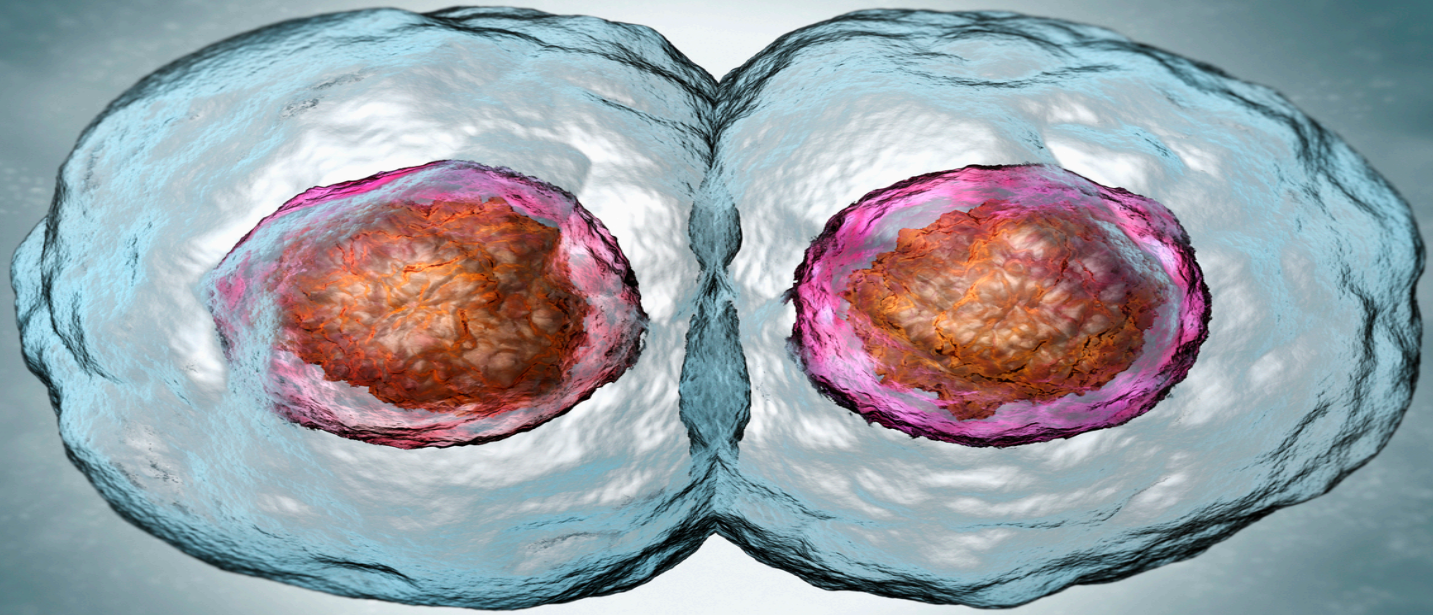
What is a stateless service?

Not a cache or a database

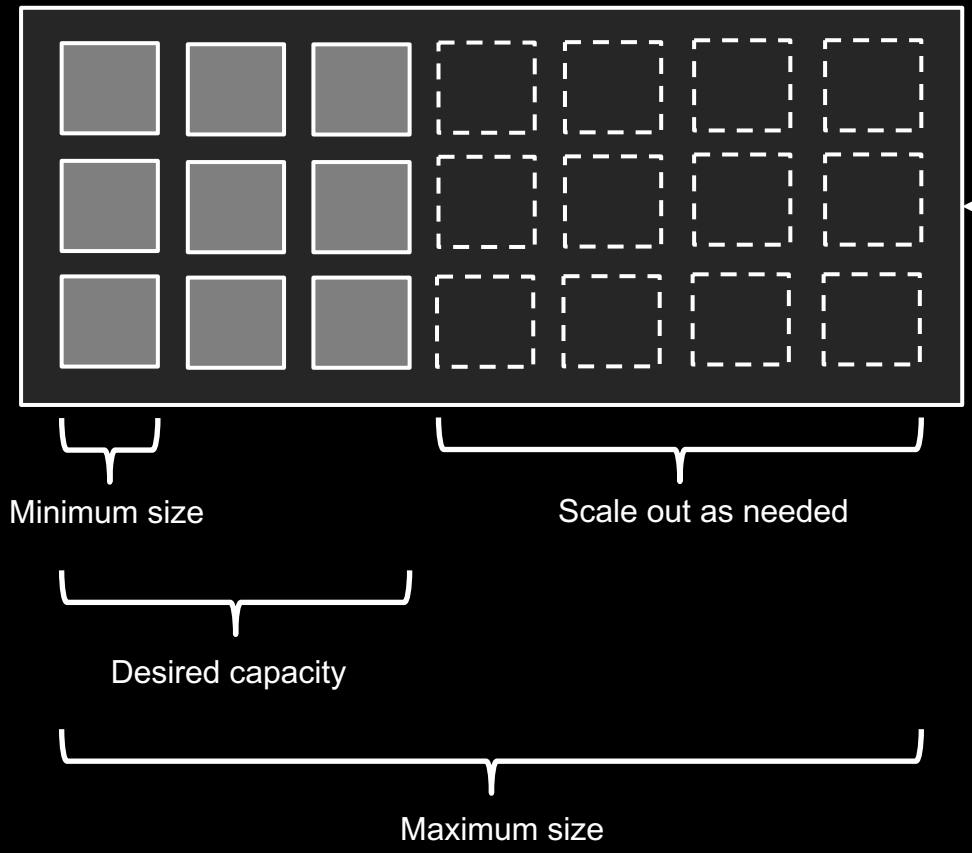
Frequently accessed metadata

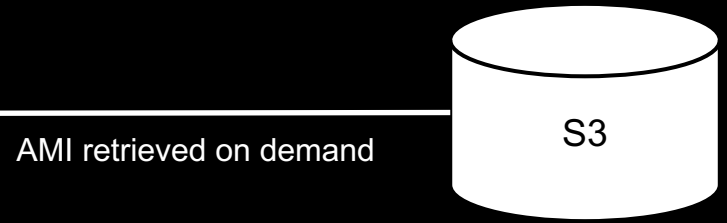
No instance affinity

Loss a node is a non-event



Auto Scaling Groups

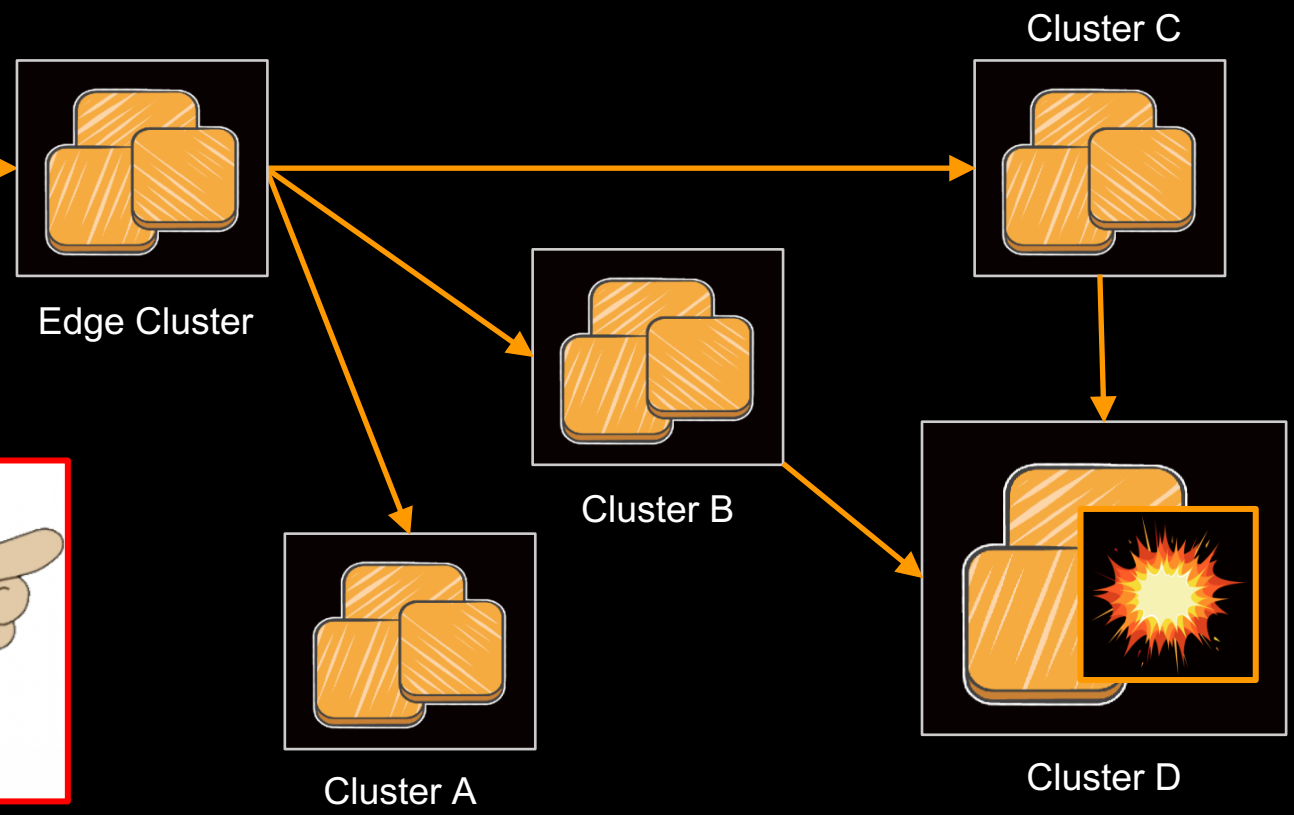




Compute efficiency
Node failure
Traffic spikes
Performance bugs

amazon web services

Surviving Instance Failure



Stateful Services

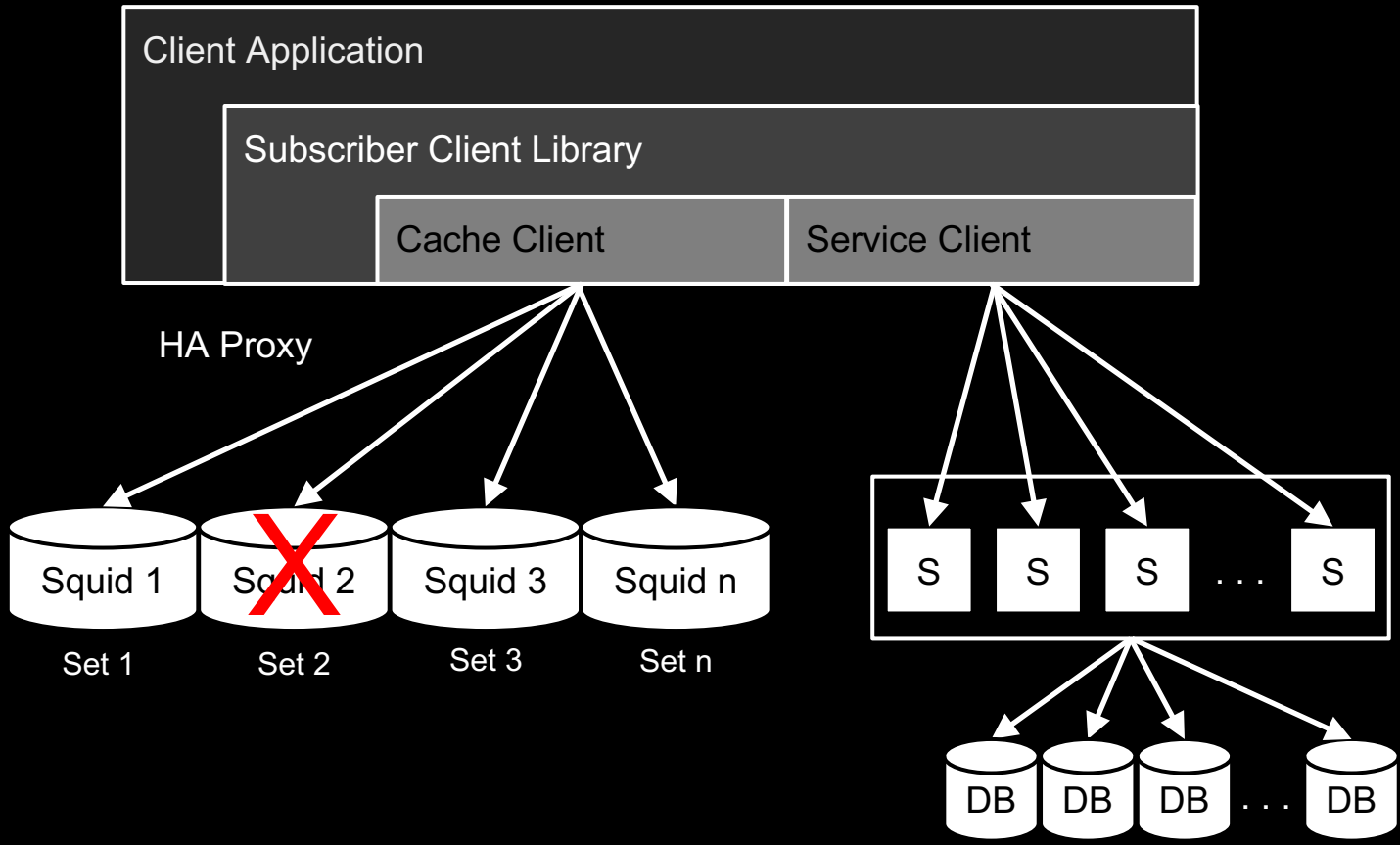
What is a stateful service?

Databases & caches

Custom apps which hold large amounts of data

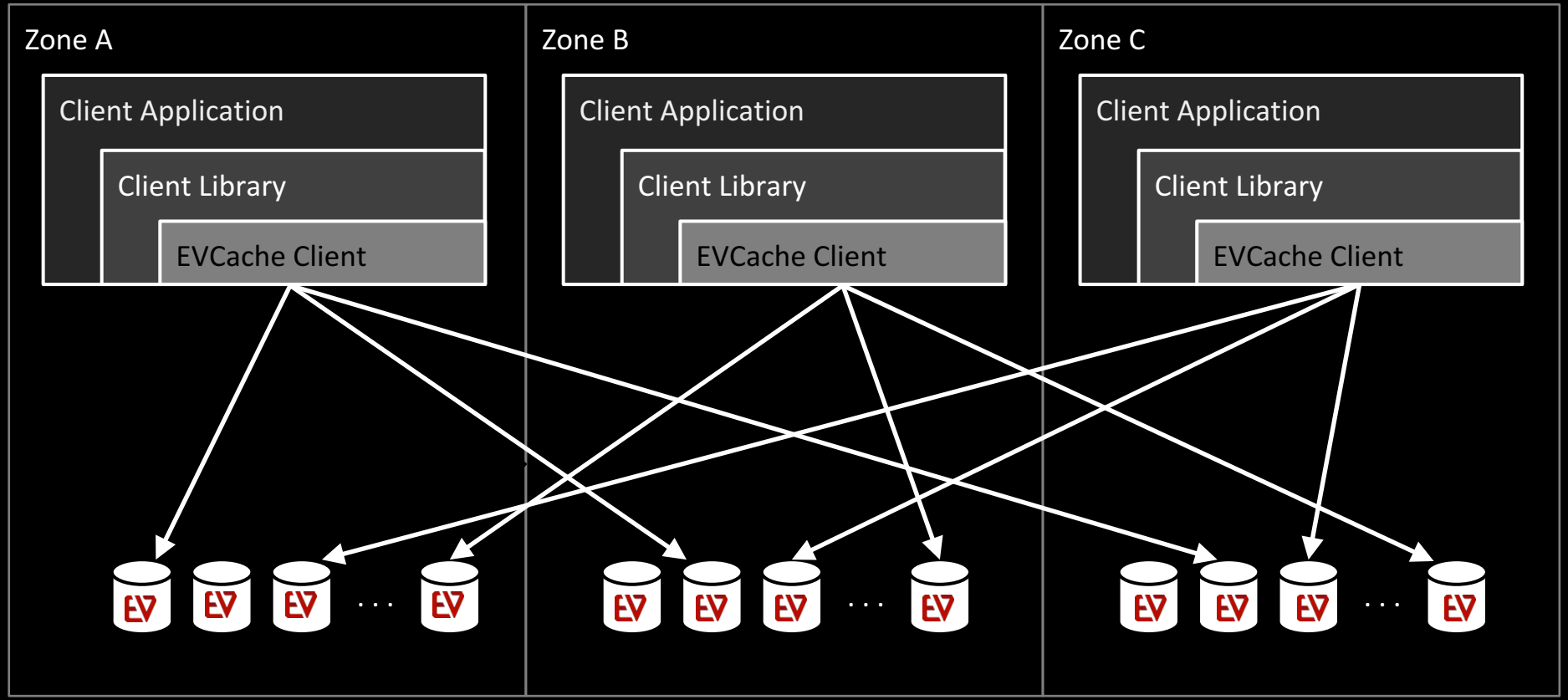
Loss of a node is a notable event

Dedicated Shards – An Antipattern

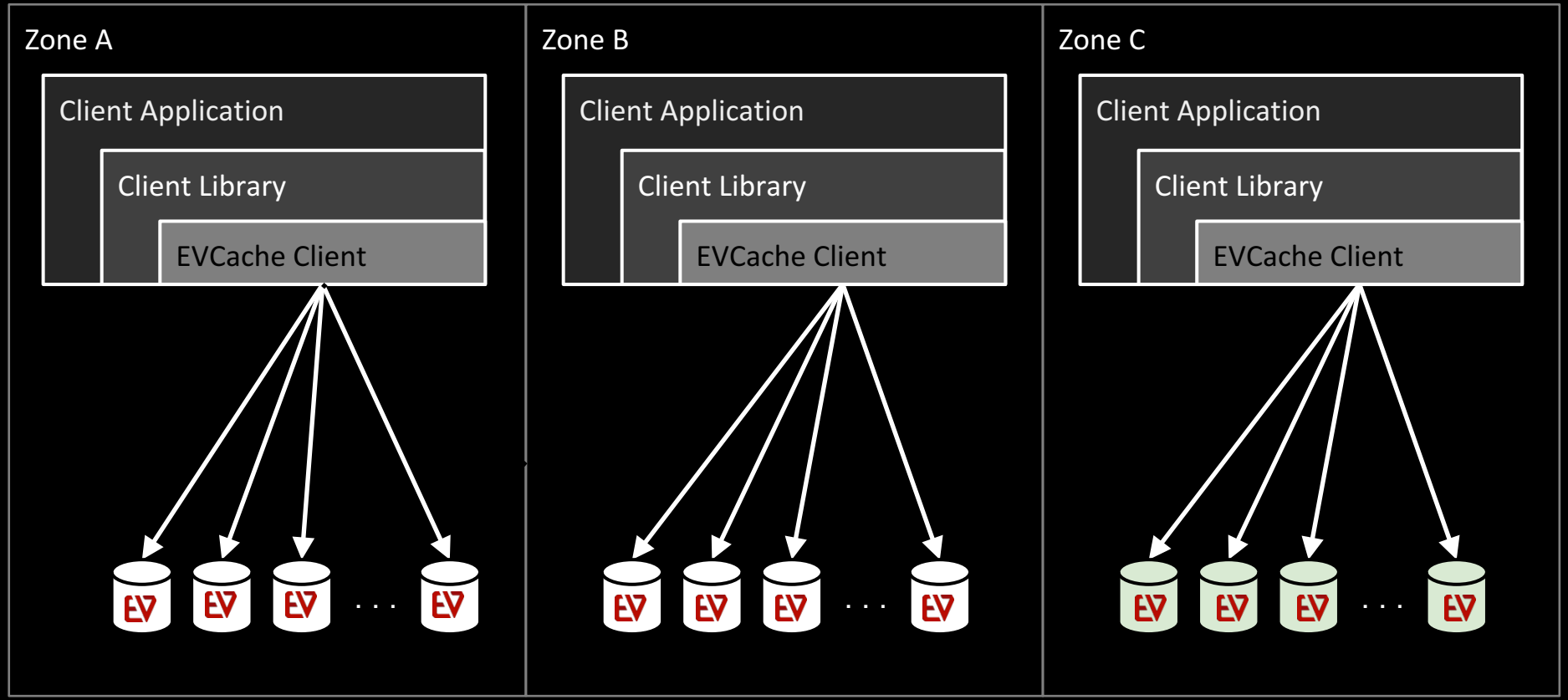


Redundancy is fundamental

EVCache Writes

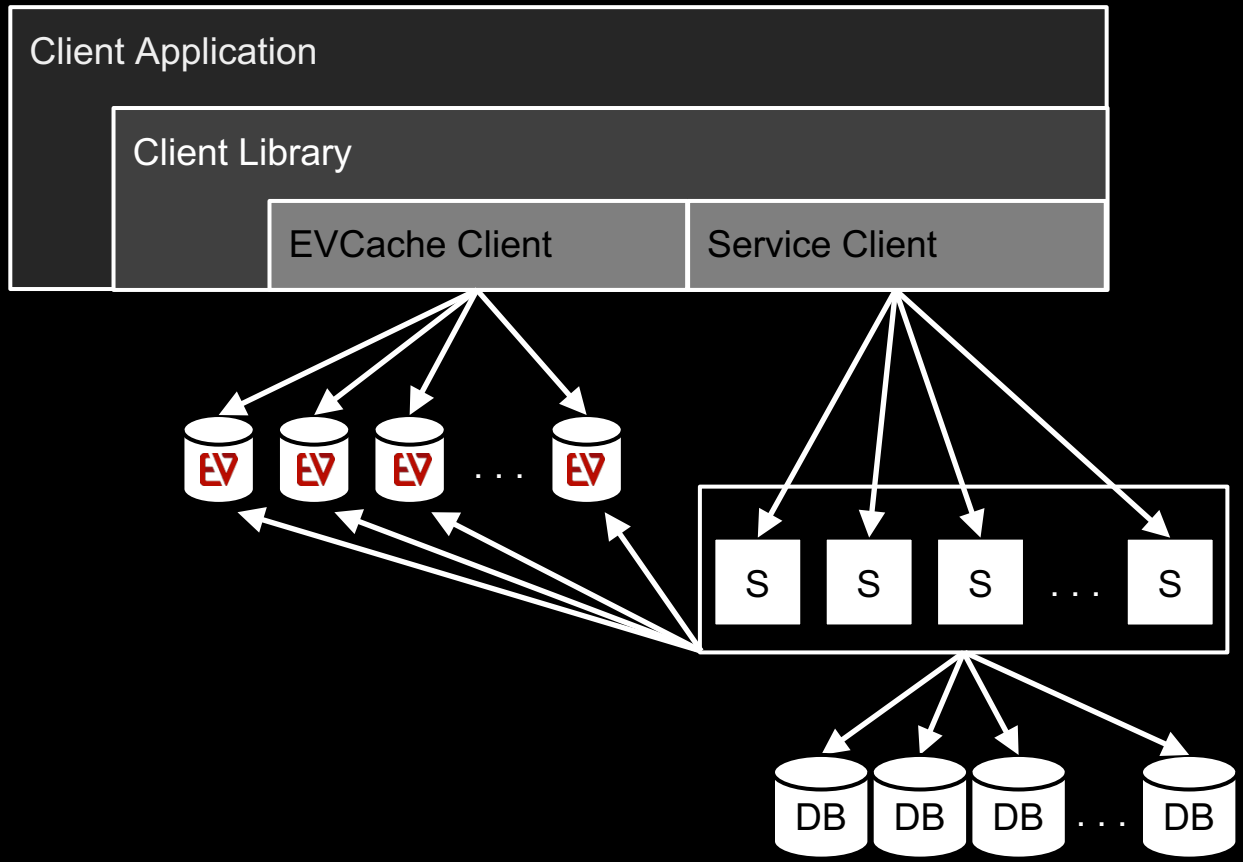


EVCache Reads



Hybrid Services

Hybrid Microservice



It’s easy to take EVCache for granted

30 million requests/sec
2 trillion requests per day globally

Hundreds of billions of objects
Tens of thousands of memcached instances

Milliseconds of latency per request

Excessive Load

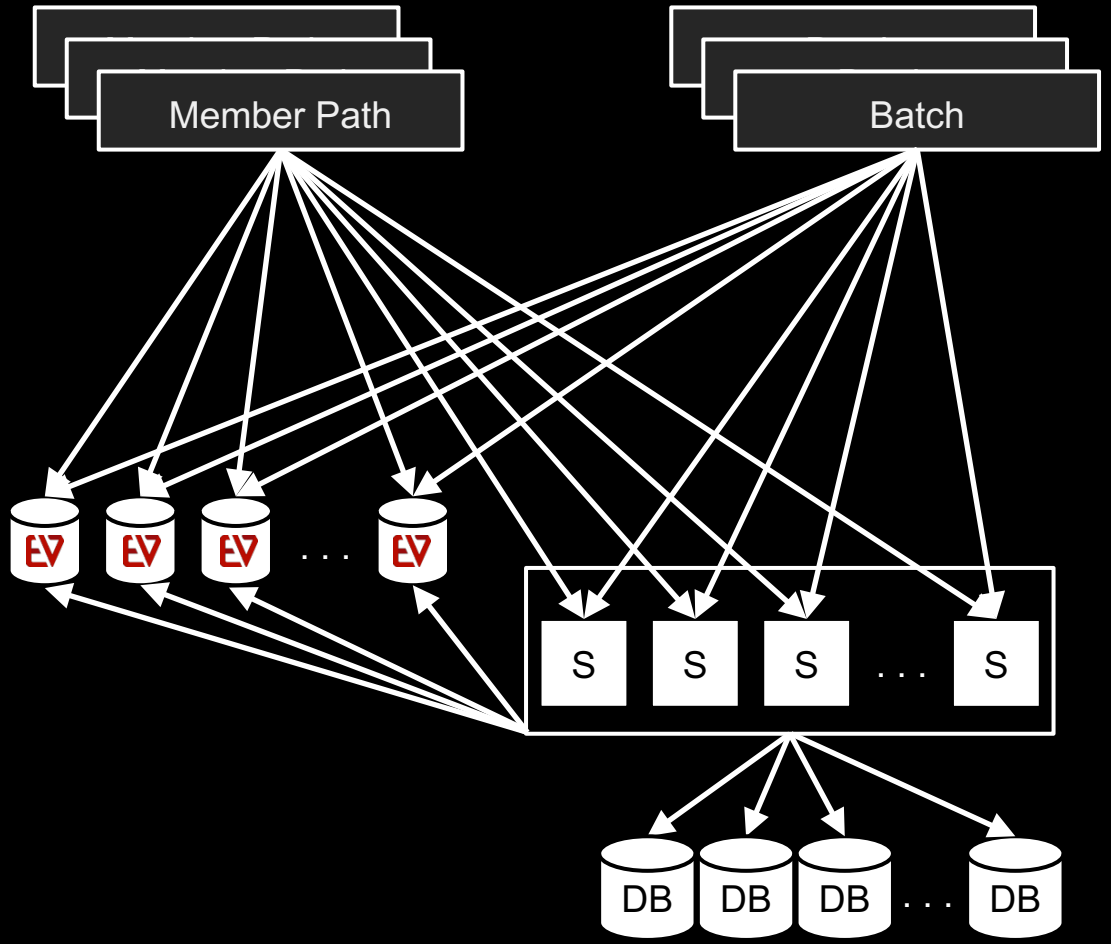
Called by many services

Online & offline clients

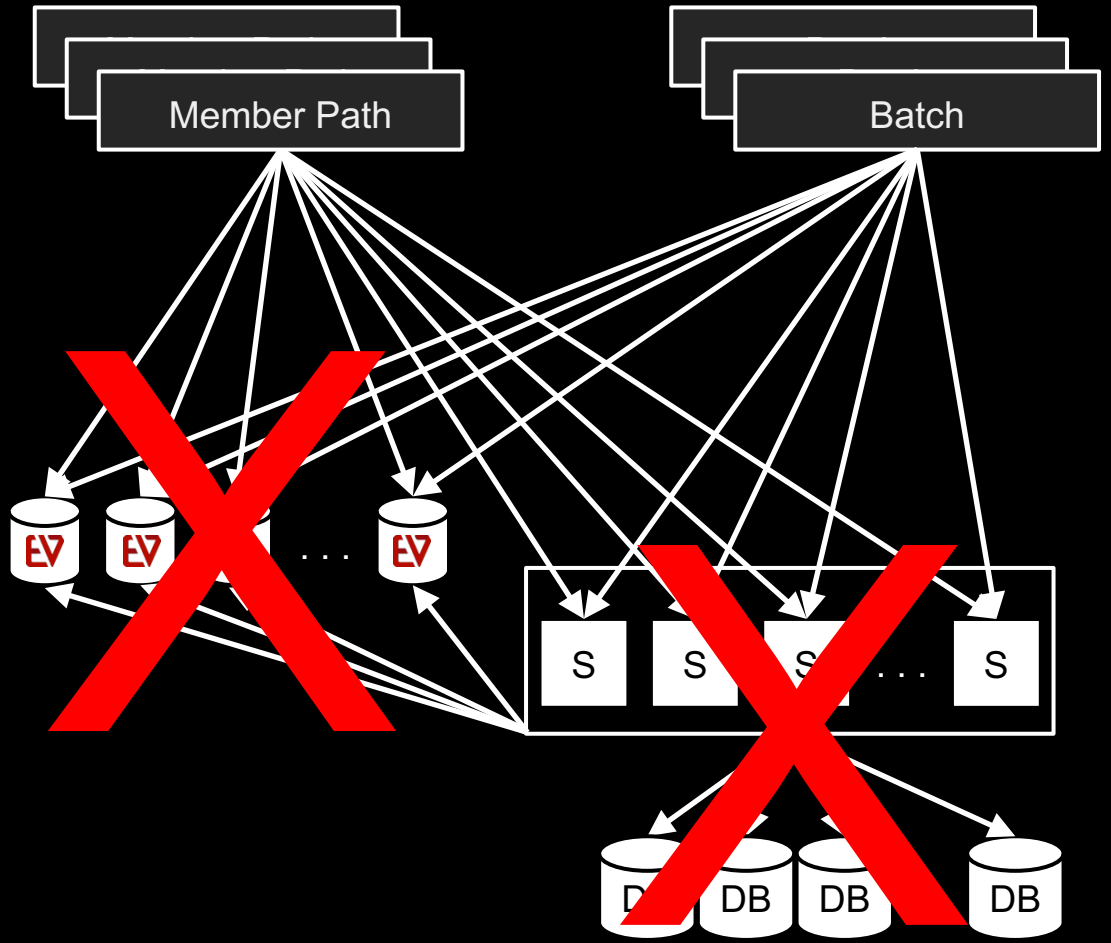
Called many times / request

800k – 1M RPS

Fallback to service/db



Excessive Load



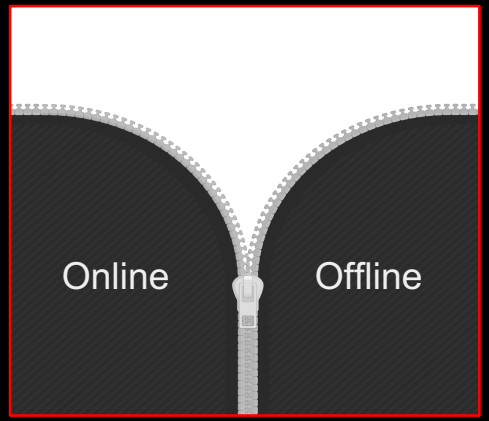
Solutions

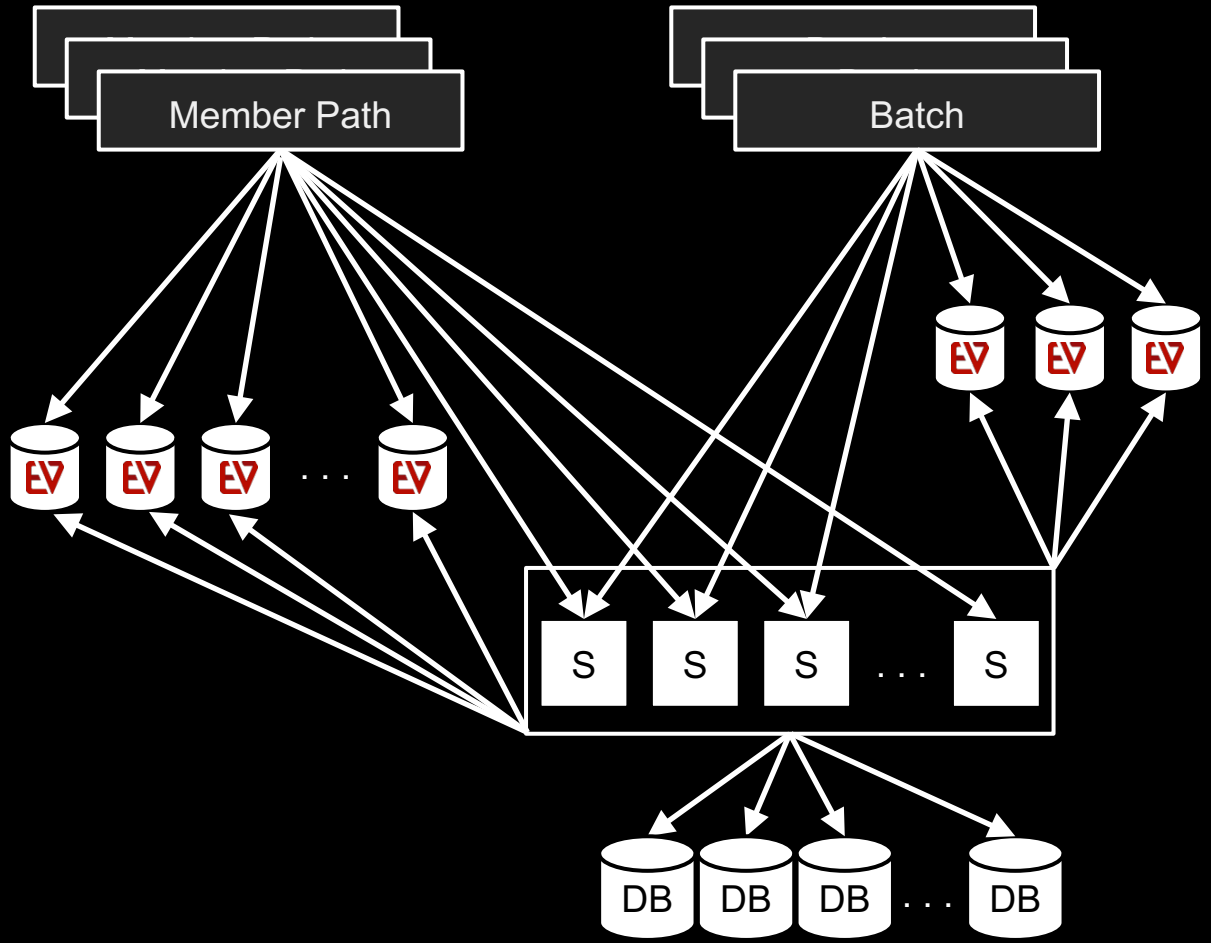
Workload partitioning

Request-level caching

Secure token fallback

Chaos under load





Challenges & Solutions

Dependency

Scale

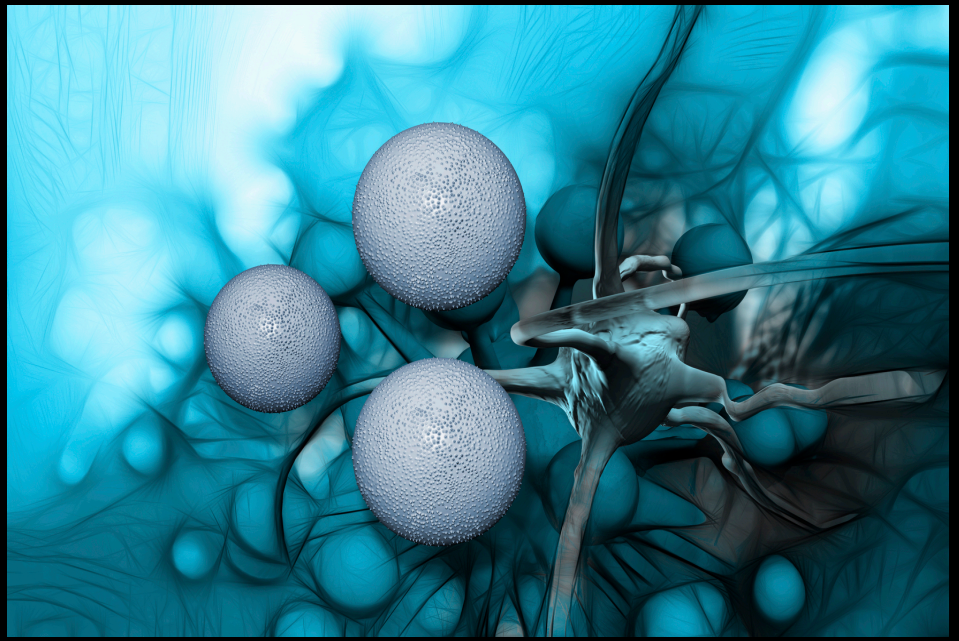
Variance

Change

Use Cases

Operational drift

Polyglot & containers



Operational Drift
(Unintentional Variance)

Operational Drift

Over time

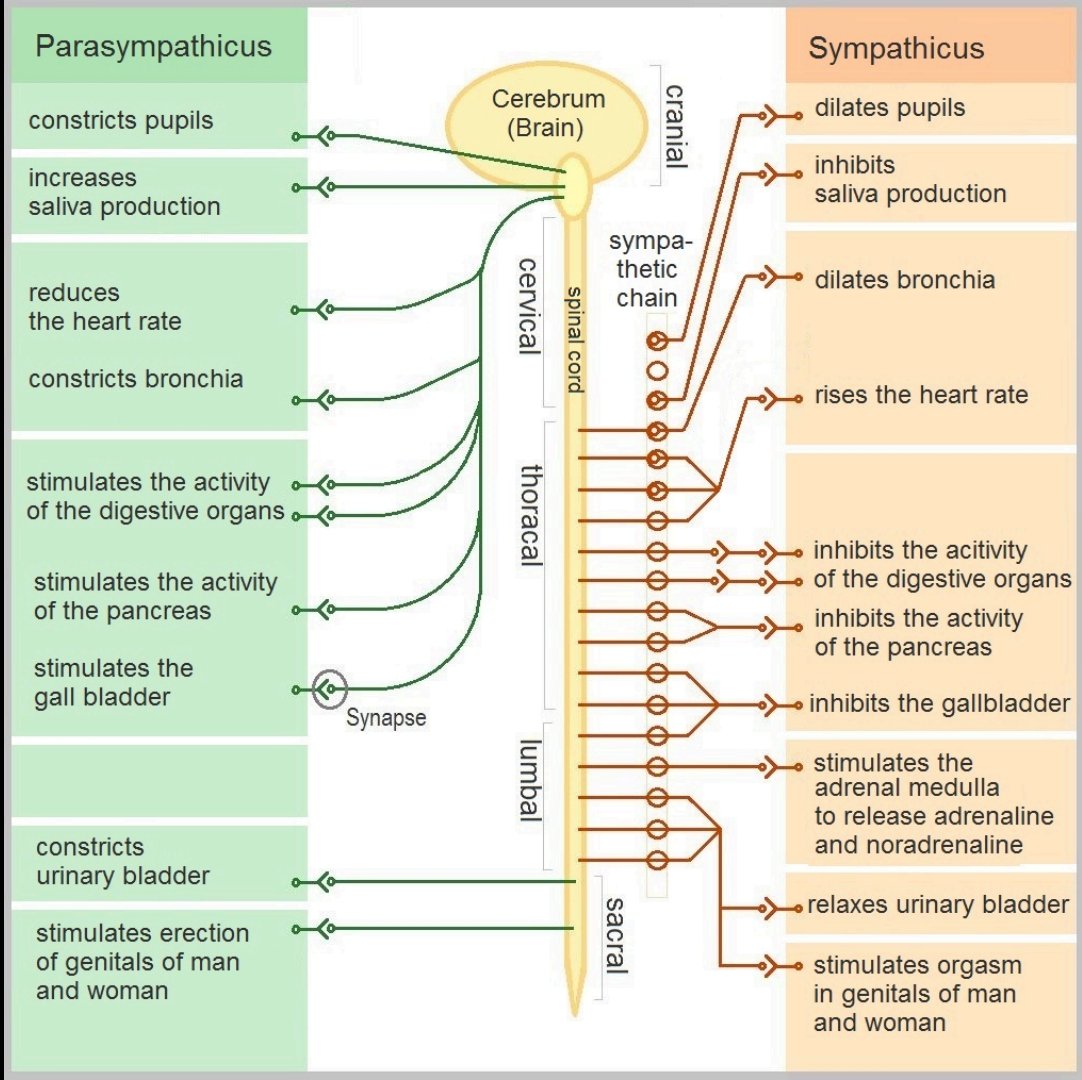
- Alert thresholds
- Timeouts, retries, fallbacks
- Throughput (RPS)

Across microservices

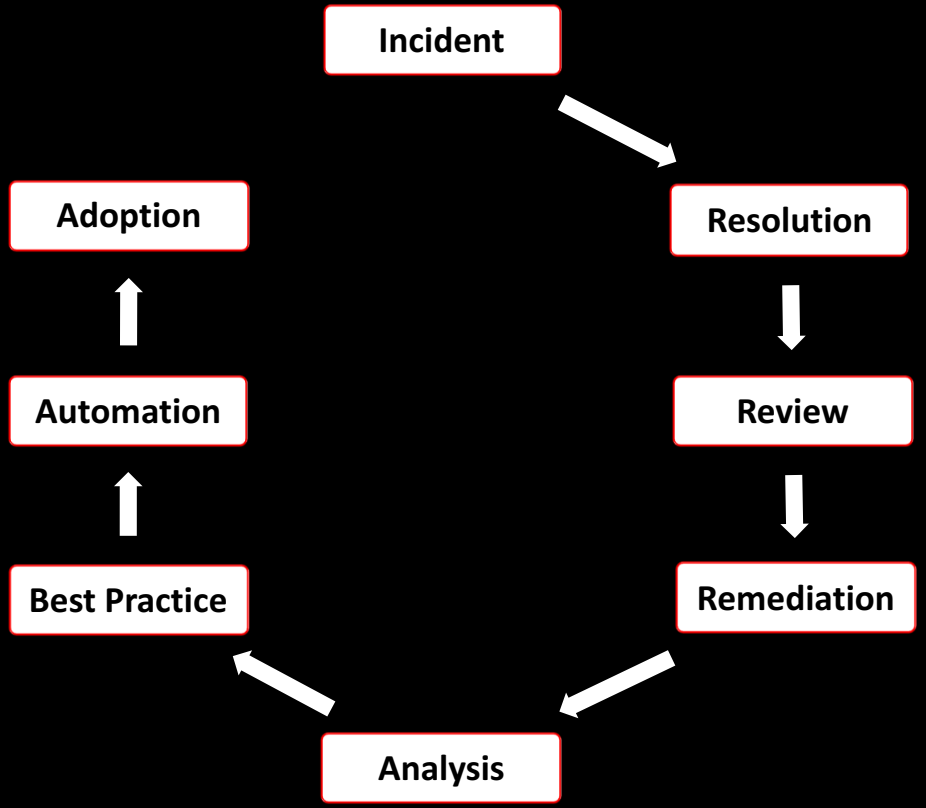
- Reliability best practices

Autonomic Nervous System

You don’t have to think about digestion or breathing

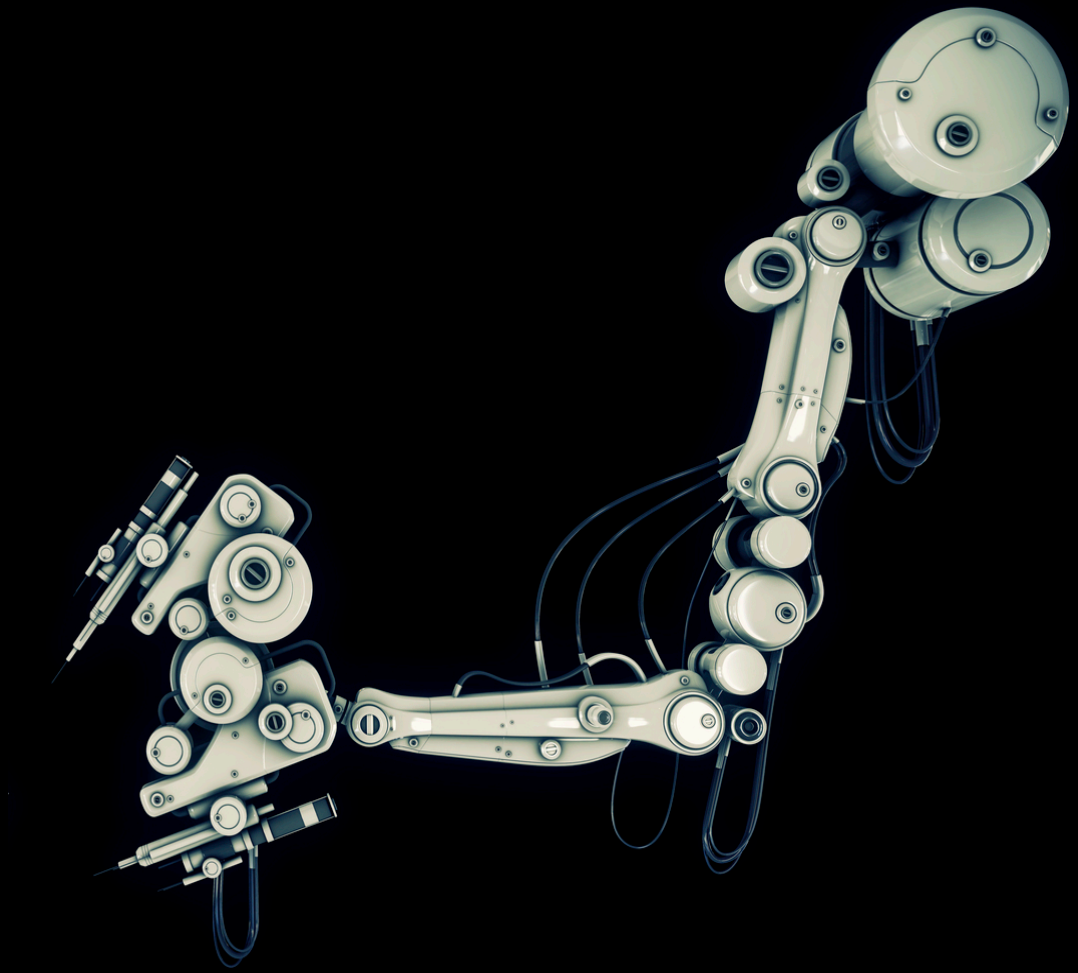


Continuous Learning & Automation



Production Ready

Alerts
Apache & Tomcat
Automated canary analysis
Autoscaling
Chaos
Consistent naming
ELB config
Healthcheck
Immutable machine images
Squeeze testing
Staged, red/black deployments
Timeouts, retries, fallbacks

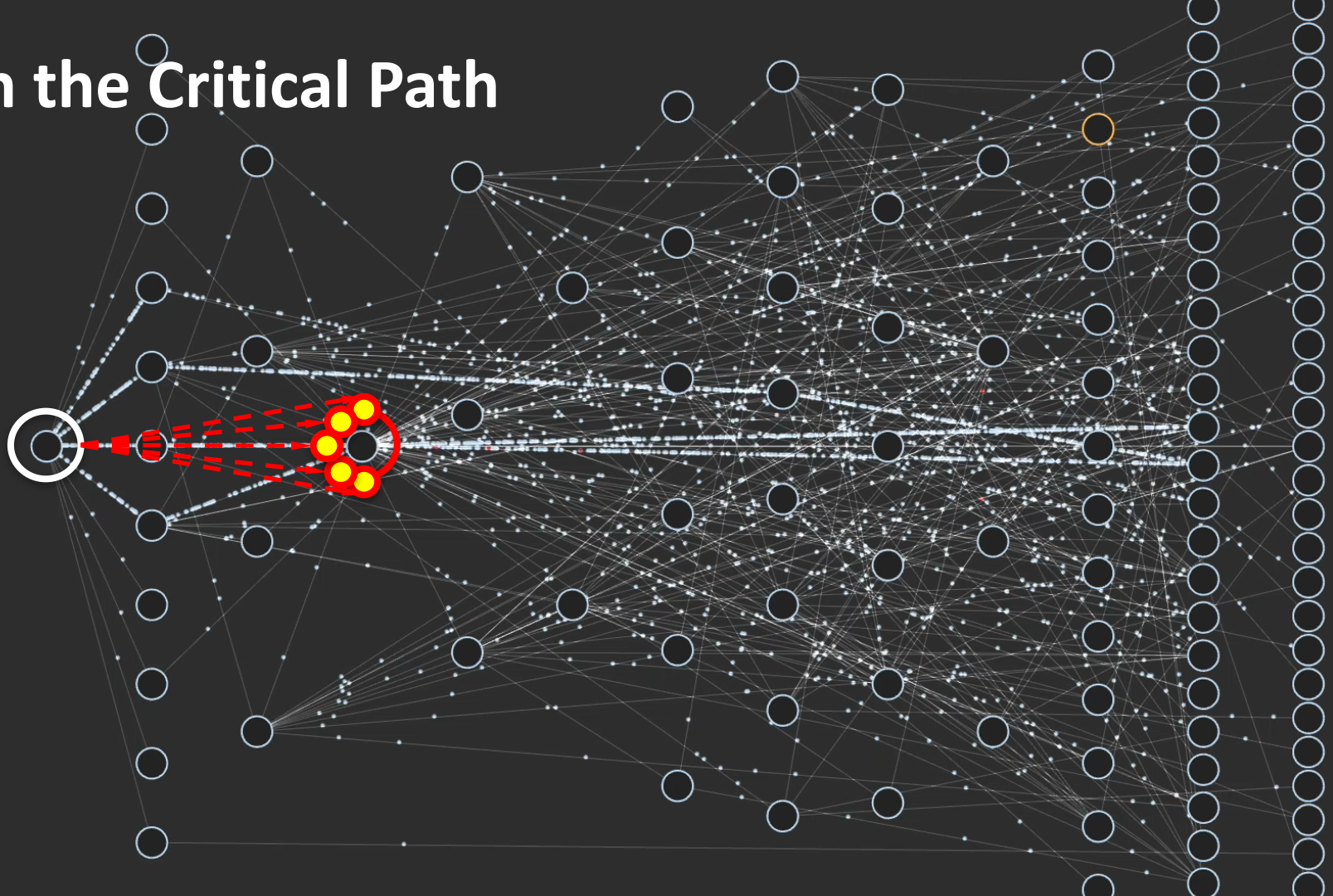


Polyglot & Containers
(Intentional Variance)

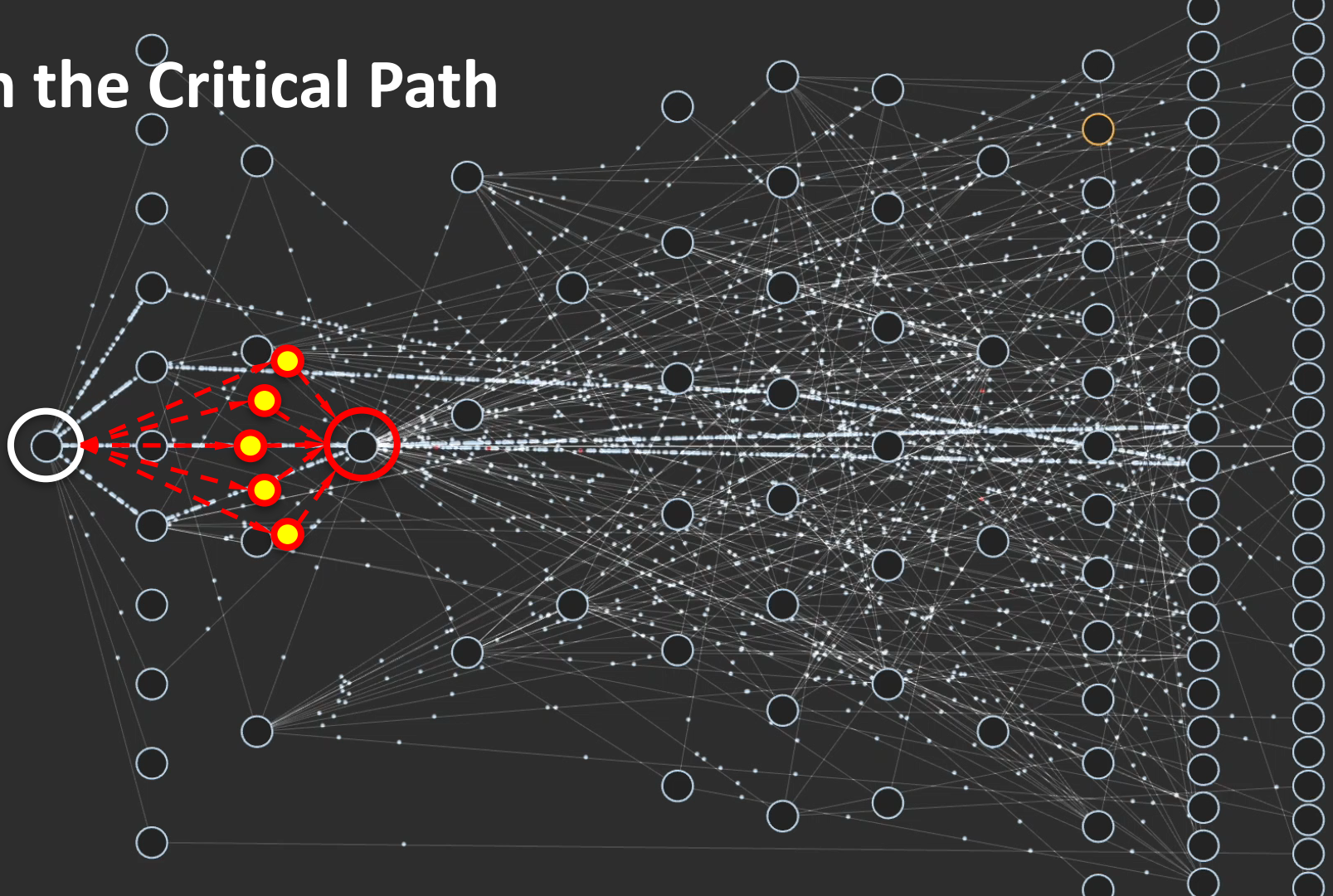

The Paved Road
Stash
Nebula/Gradle
BaseAMI/Ubuntu
Jenkins
Spinnaker
Runtime Platform


node js
python
docker
Ruby

In the Critical Path



In the Critical Path



Cost of Variance

Productivity tooling

Insight & triage capabilities

Base image fragmentation

Node management

Library/platform duplication

Learning curve - production expertise

Strategic Stance

Raise awareness of costs

Constrain centralized support

Prioritize by impact

Seek reusable solutions

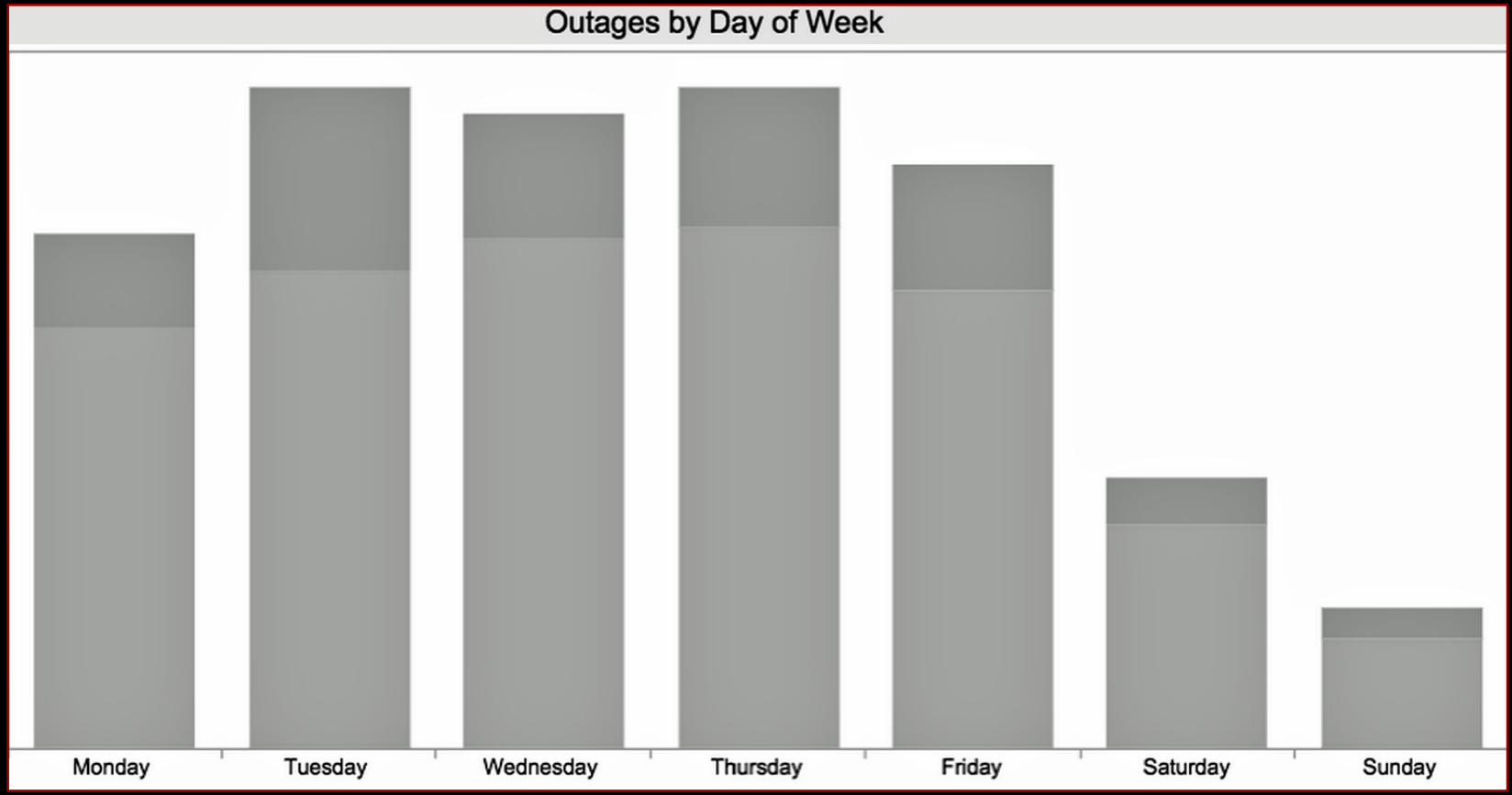
Challenges & Solutions

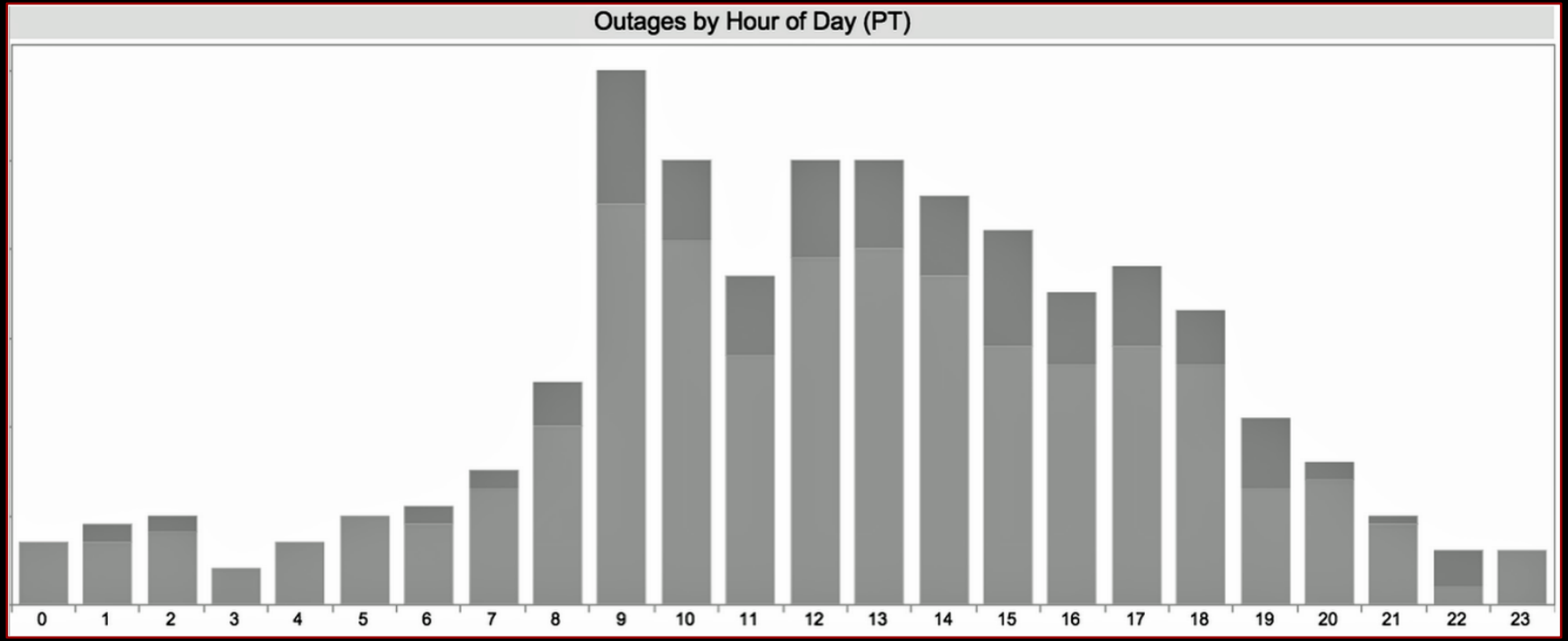
Dependency

Scale

Variance

Change

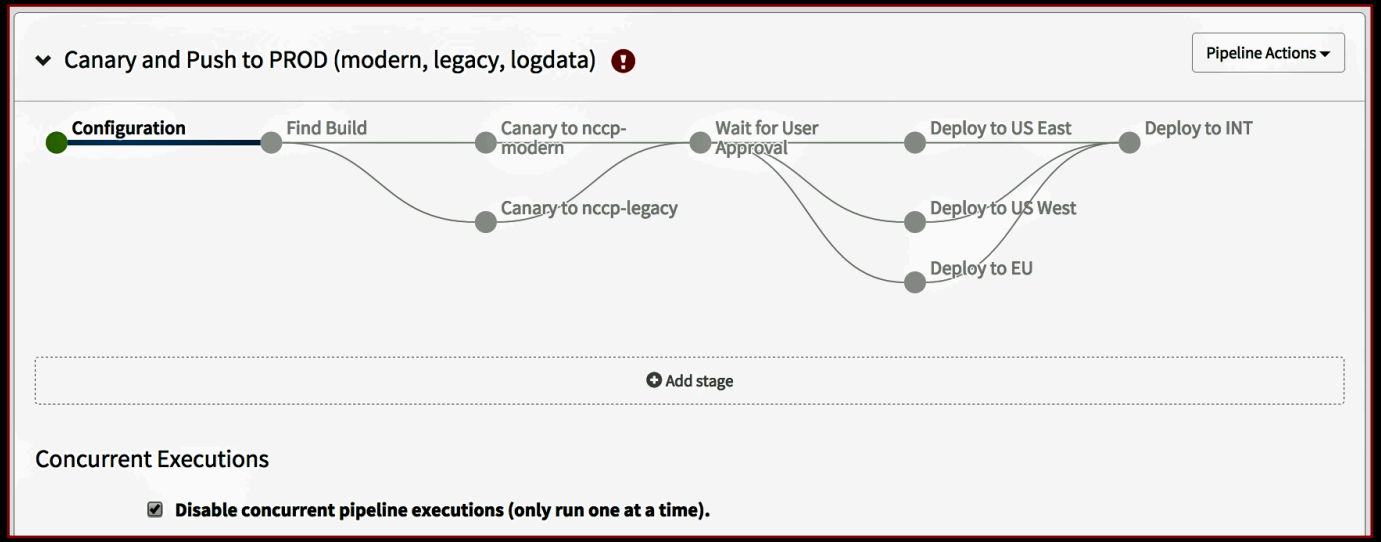

Outages by Day of Week
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Sunday


Outages by Hour of Day (PT)
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23

How do we achieve velocity with confidence?

Global Cloud Management & Delivery

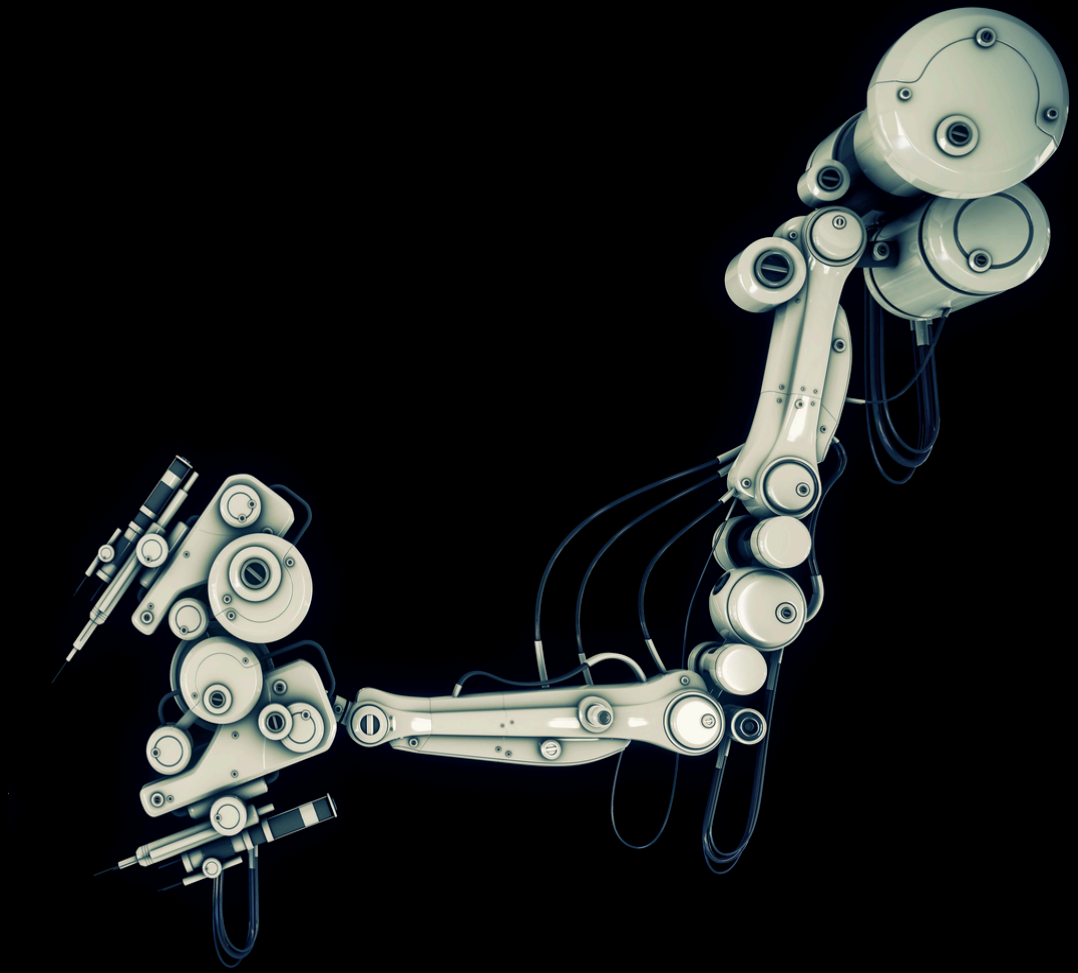
Integrated, Automated Practices

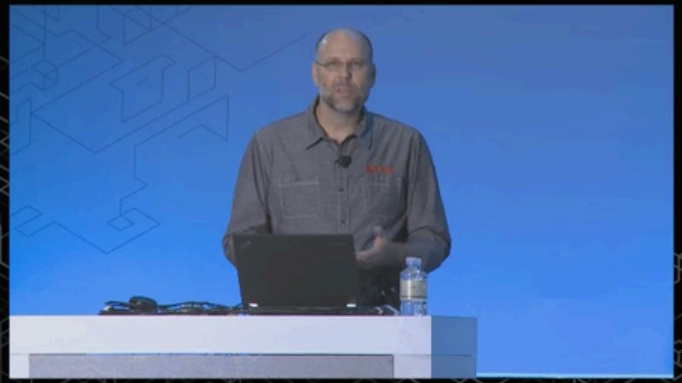


Conformity checks
Red/black pipelines
Automated canaries
Staged deployments
Squeeze tests

Production Ready

Alerts
Apache & Tomcat
Automated canary analysis
Autoscaling
Chaos
Consistent naming
ELB config
Healthcheck
Immutable machine images
Squeeze testing
Staged, red/black deployments
Timeouts, retries, fallbacks





AWS re:Invent

ISM301

Engineering Netflix Global Operations in the Cloud

Josh Evans - Director of Operations Engineering

amazon web services

https://www.youtube.com/watch?v=IkPb15FfuQU

Our Talk Today

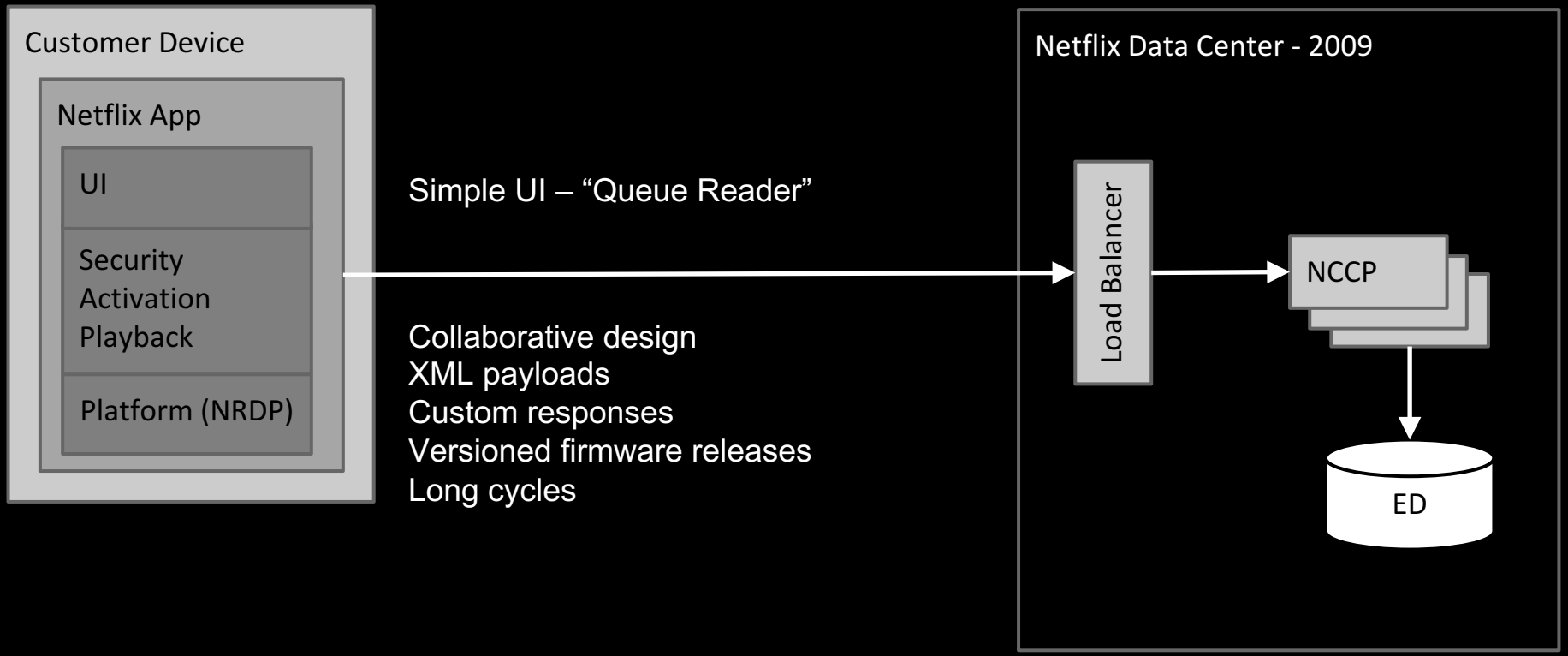
Introductions

Microservice Basics

Challenges & Solutions

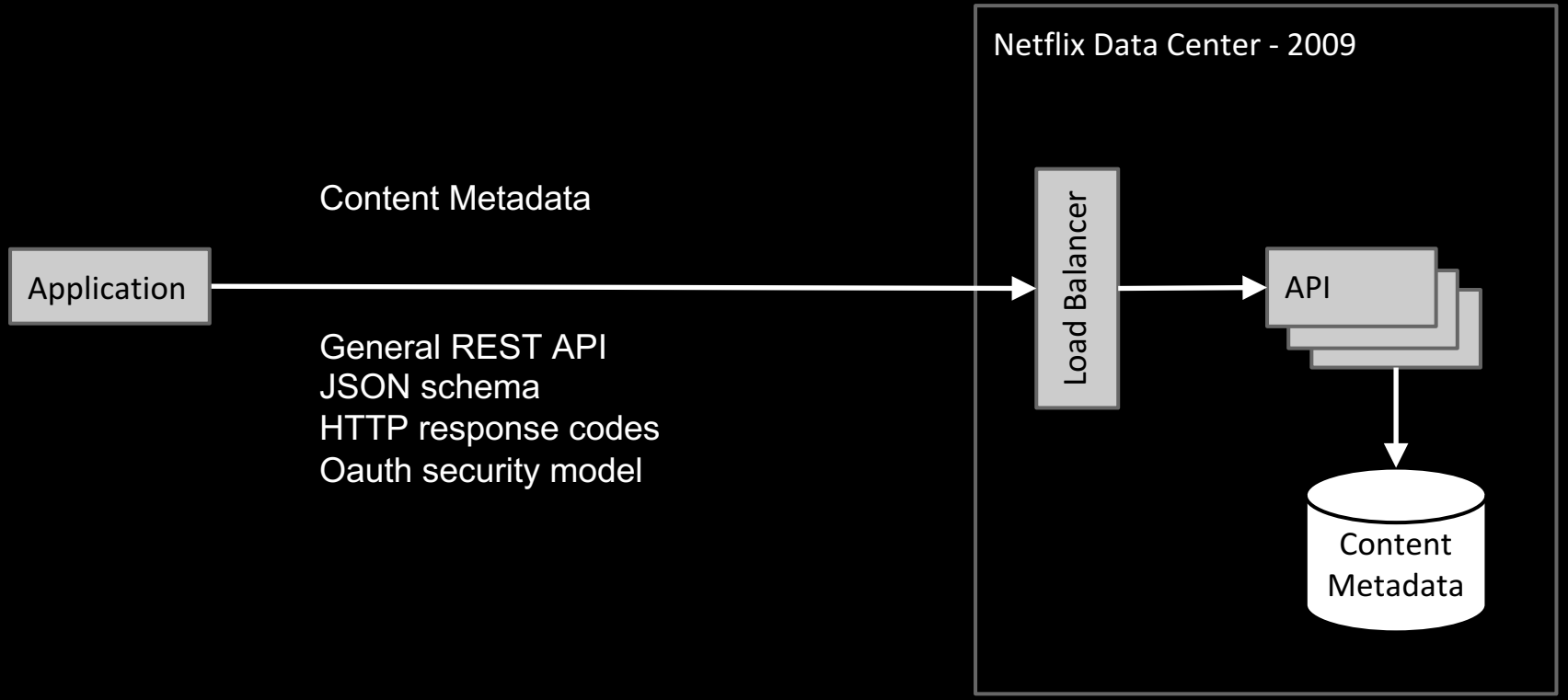
Organization & Architecture

Electronic Delivery - NRDP 1.x

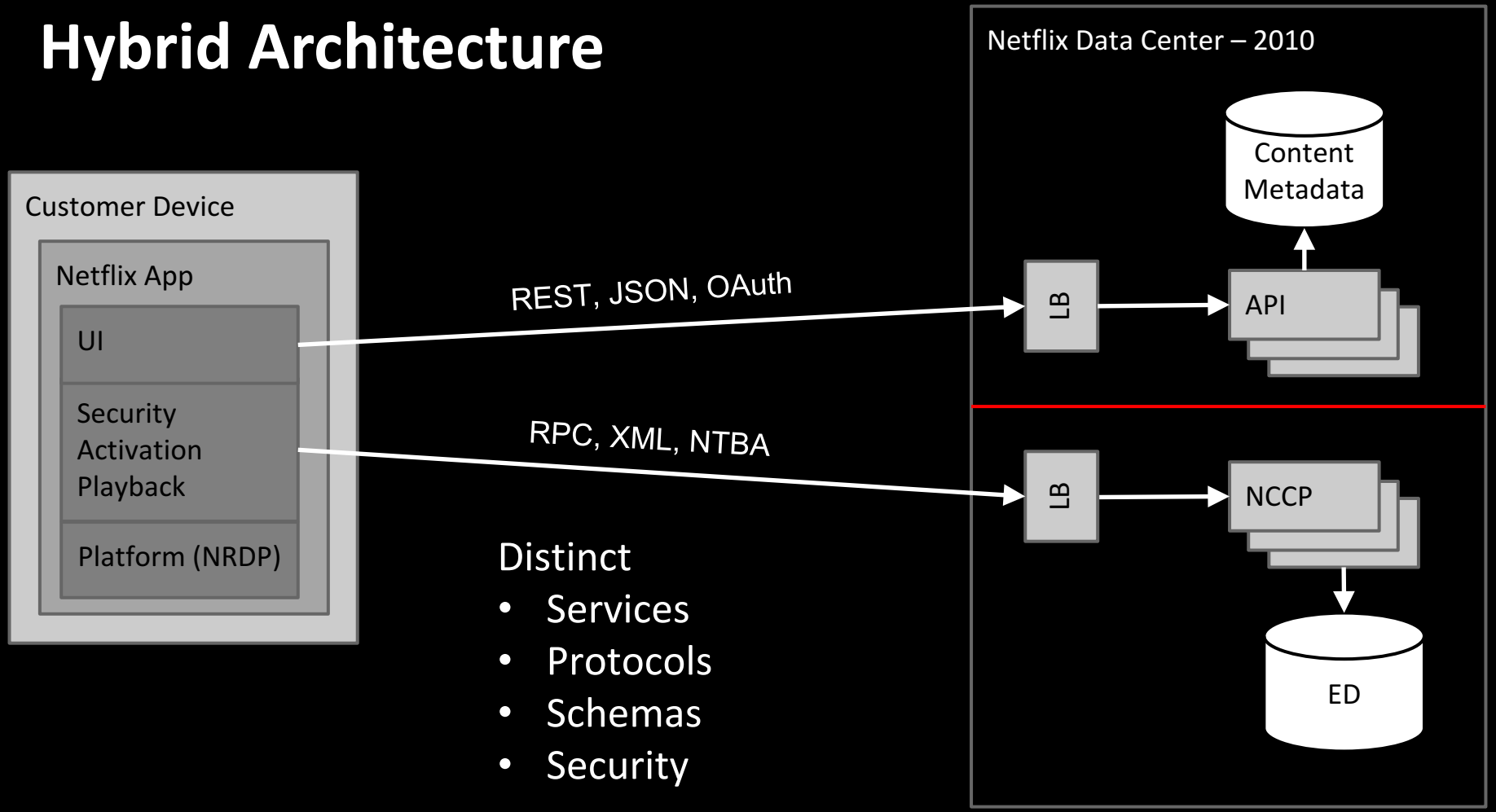


Netflix API - let a 1000 flowers bloom!

Netflix API – from public to private



Hybrid Architecture



Distinct

- Services
- Protocols
- Schemas
- Security

Josh: what is the right long term architecture?

Peter: do you care about the organizational implications?

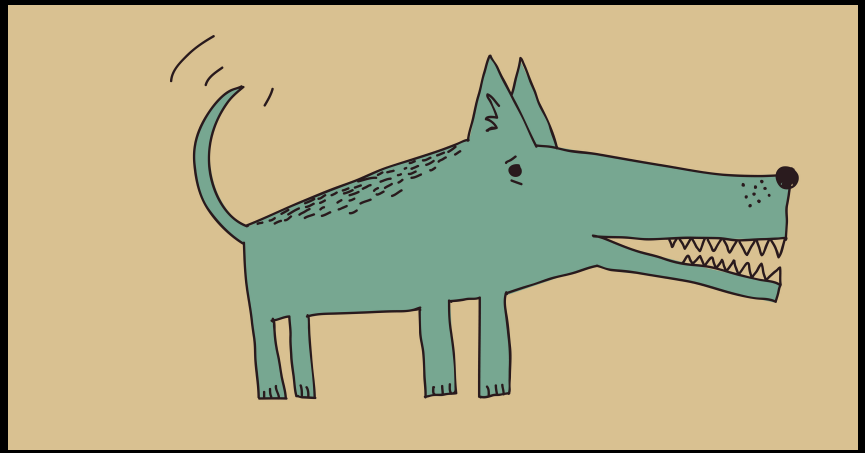
Conway's Law

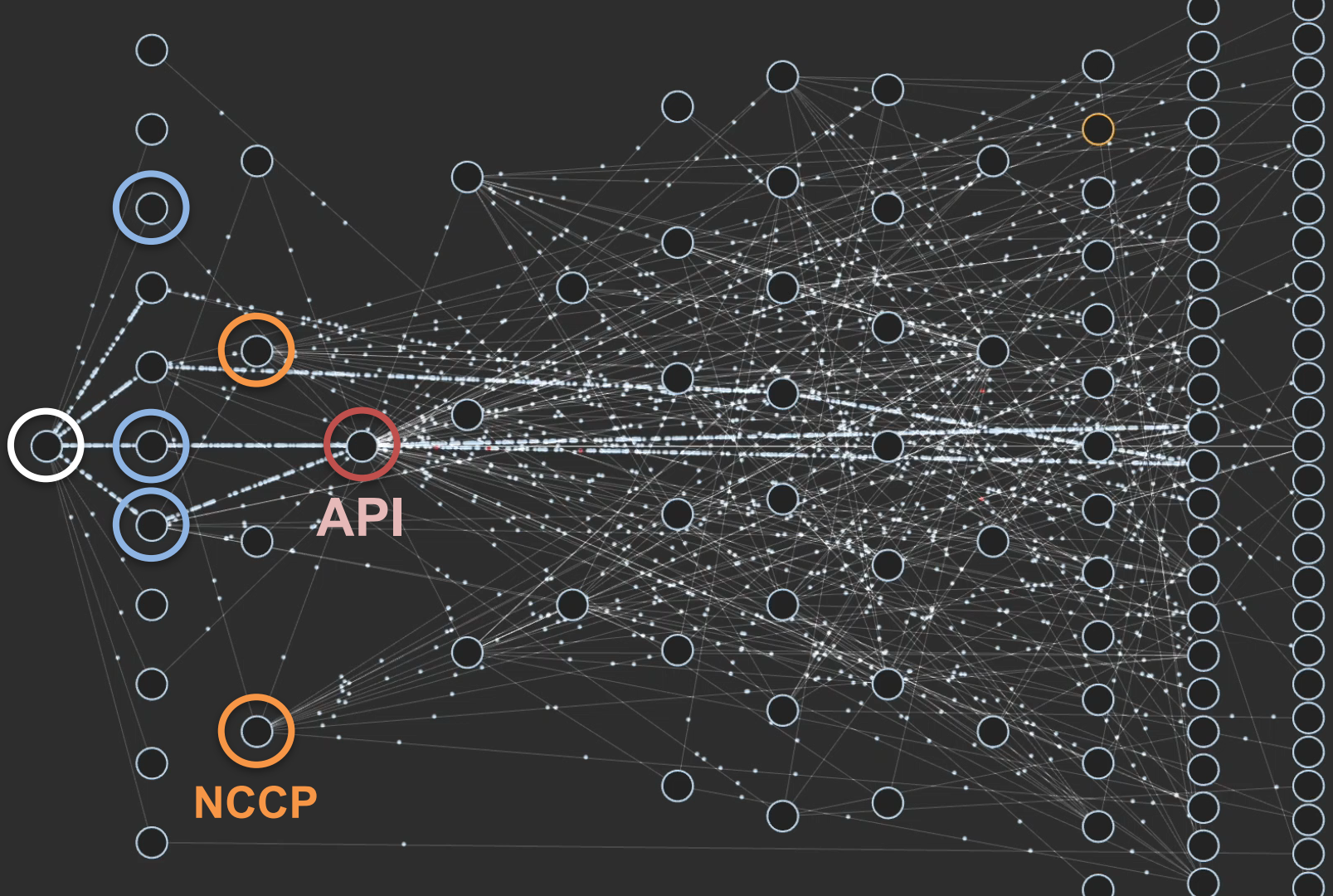
Organizations which design systems are constrained to produce designs which are copies of the communication structures of these organizations.

Any piece of software reflects the organizational structure that produced it.

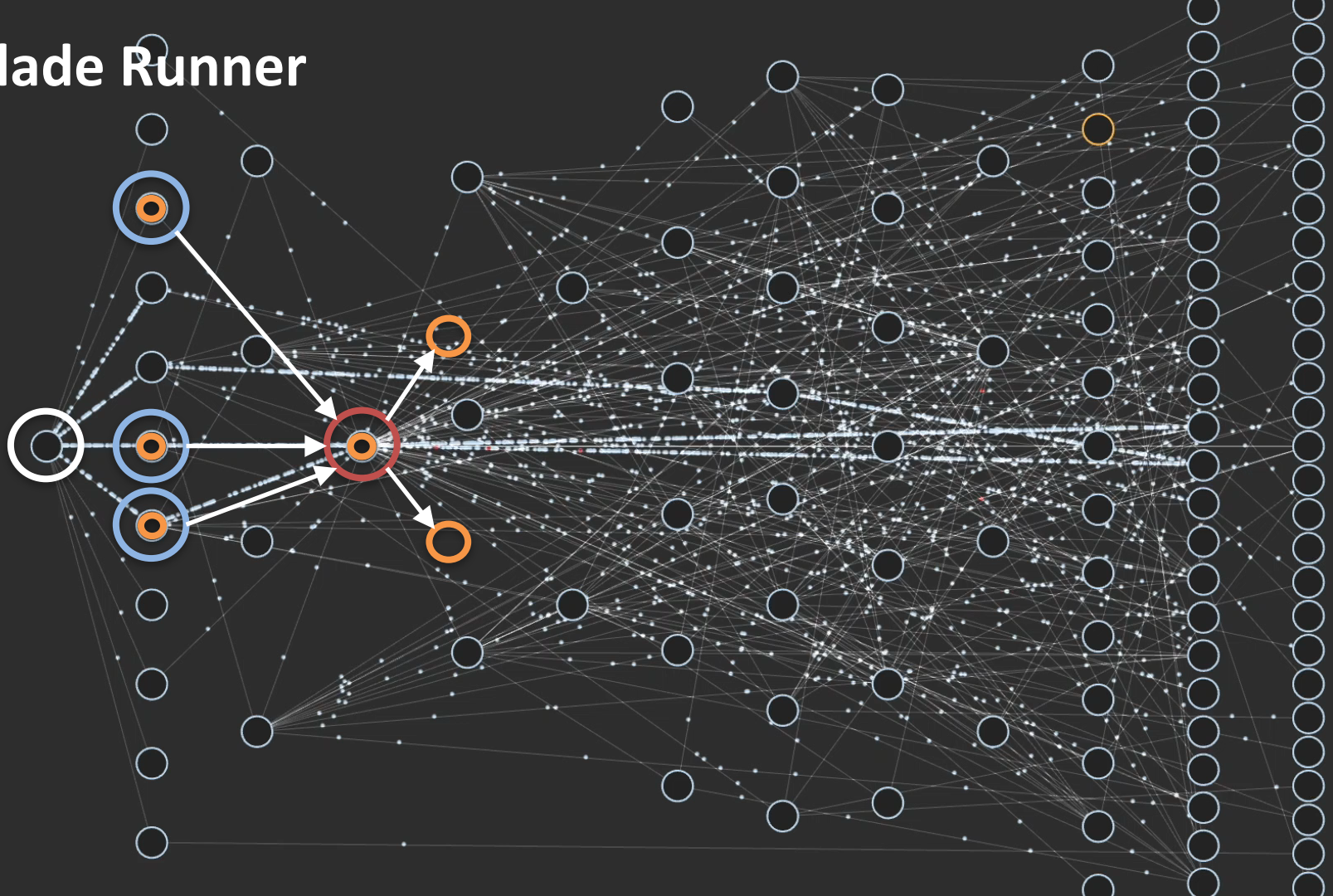
Conway’s Law

If you have four teams working on a compiler you will end up with a four pass compiler




API
NCCP

Blade Runner



Outcomes & Lessons

Outcomes

- Productivity & new capabilities
- Refactored organization

Lessons

- Solutions first, team second
- Reconfigure teams to best support your architecture

Our Talk Today

Introductions

Microservice Basics

Challenges & Solutions

Organization & Architecture

Recap

Microservice architectures are complex and organic

Health depends on discipline and chaos

Dependency

- Circuit breakers, fallbacks, chaos
- Simple clients
- Eventual consistency
- Multi-region failover

Variance

- Engineered operations
- Understood cost of variance
- Prioritized support by impact

Organization & Architecture

- Solutions first, team second

Scale

- Auto-scaling
- Redundancy – avoid SPoF
- Partitioned workloads
- Failure-driven design
- Chaos under load

Change

- Automated delivery
- Integrated practices

NETFLIX | OSS

Data Persistence

Storing and Serving data in the Cloud.

Common Runtime Services & Libraries

Runtime containers, libraries and services that power microservices

The cloud platform is the foundation and technology stack for the majority of the services within Netflix. The cloud platform consists of cloud services, application libraries and application containers. Specifically, the platform provides service discovery through Eureka, distributed configuration through Archaius, resilent and intelligent inter-process and service communication through Ribbon. To provide reliability beyond single service calls, Hystrix is provided to isolate

Build and Delivery Tools

Taking code from desktop to the cloud

Netflix has open sourced many of our Gradle plugins under the name Nebula. Nebula started off as a set of strong opinions to make Gradle simple to use for our developers. But we quickly learned that we could use the same assumptions on our open source projects and on other Gradle plugins to make them easy to build, test and deploy. By standardizing plugin development, we've lowered the barrier to generating them, allowing us to keep our build modular and composable.

We require additional tools to take these builds from the developers' desks to AWS. There are tens of thousands of instances running Netflix. Every one of these runs on top of an image created by our open source tool Aminator. Once packaged, these AMIs are deployed to AWS using our cloud deployment and management tool, Spinnaker.

netflix.github.io

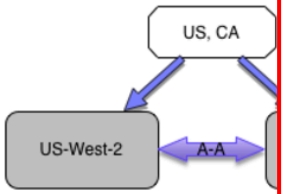
NETFLIX

The Netflix Tech Blog

Wednesday, March 30, 2016

Global Cloud - Active-Active and Beyond

This is a continuing post on the Netflix architecture for Global Availability. In the past we talked about efforts like Isthmus and Active-Activ
end of the Active-Active project in 2013. V
members in the Americas, where the vast
Our European members, however, were



Wednesday, August 3, 2016

Vizceral Open Source

V

Previously we wrote about our traffic int
share about this project. First, we have
open source!

Tuesday, March 1, 2016

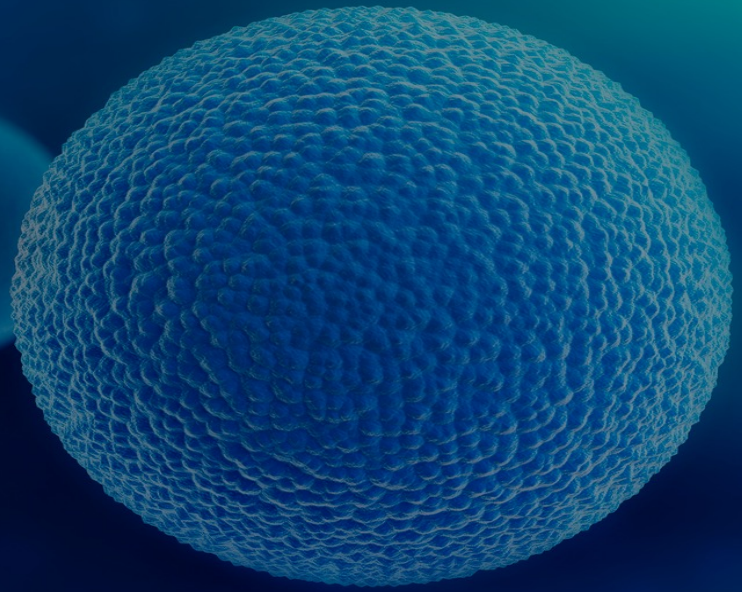
Caching for a Global Netflix

#CachesEverywhere

Netflix members have come to expect a great user experience when interacting with our service. There are many things that go into delivering a customer-focused user experience for a streaming service, including an outstanding content library, an intuitive user interface, relevant and personalized recommendations, and a fast service that quickly gets your favorite content playing at very high quality, to name a few.

techblog.netflix.com

Questions?



QCon
SAN FRANCISCO

Josh Evans
@Ops_Engineering