# Building an agile platform in a highly regulated industry

If Insurance – The Nordic insurance company

Fredrkl

Fredrik Klingenberg

fredrkl.com

DEV**REAL**

Microsoft® Most Valuable Professional

træfik Ambassador

# Jonas Samuelsson

## If Insurance

https://medium.com/if-tech/cloudix-building-a-multi-tenant-platform-service-in-the-highly-regulated-insurance-industry-d3c42c110459

# Agenda

- If-Insurance introduction
- Constraints in the insurance industry
- Starting point
- Principles
- Building the platform
- Onboarding of teams

# This is If

## If in brief

SAMPO GROUP

Owned by Sampo plc, listed on NASDAQ Helsinki

**~7 300**
Employees (FTE)

Head office: Stockholm, Sweden
CEO: Morten Thorsrud

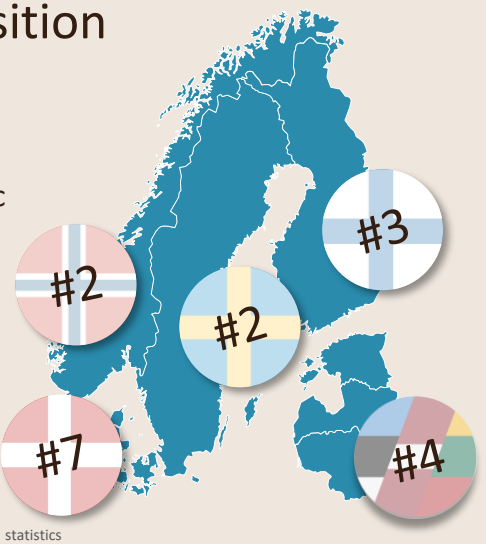
### Business areas

Private, Commercial, Industrial and Baltic

**3,9 million**
Customers

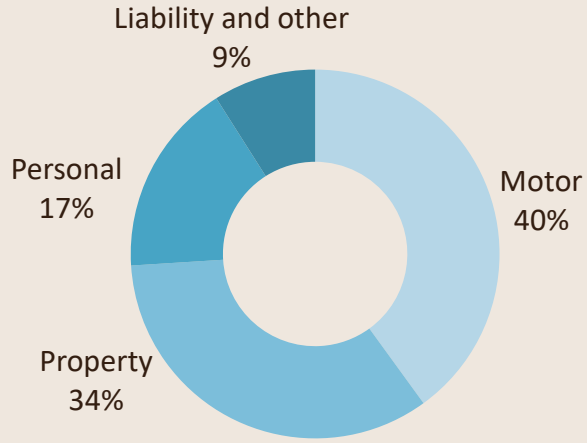
## Market position

**#1**
Nordic




Source: National insurance association statistics

## Premium by product line




## Key financials

**SEK 52 bn**
Gross written premium 2021

**81,3%**
Combined ratio 2021

## Solvency and rating

**196%**
Solvency ratio 2021

**A+**
Rating by Standard & Poor's

**A1**
Rating by Moody's

# Many different forces are affecting the insurance market today

## Technology

Smart homes

Robotics

Catbots

AI / Deep learning

CUSTOMER EXPECTATIONS

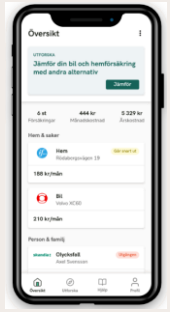
## Competitors

ICA Försäkring

Länsförsäkringar

TRYGG HANSA

## Fintech



insurely.

Get an overview of your insurances

Compare insurances

Switch insurance company

Lemonade

4.9 out of 5 25.2K Ratings

hedvig

**Root** Insurance Co

Save up to **52%** on your car insurance, using Root's mobile app-based "try before you buy" PHYD

## Partners

Connected Cars

Self driving cars

Mobility as a service

Open banking

if...

# Driving digitalization through enabling technologies and innovation…..and platforms

## If tech lab

- Building prototypes
- Supporting teams in scoping and implementing MVPs
- Exploring and evaluating new technologies
- Facilitating cross-organizational innovation

## Cloud journey

- Established Cloud and DevOps CoE
- Utilizing Microsoft Azure Cloud
- Core platform for If's digitalization journey

## Robotics and AI

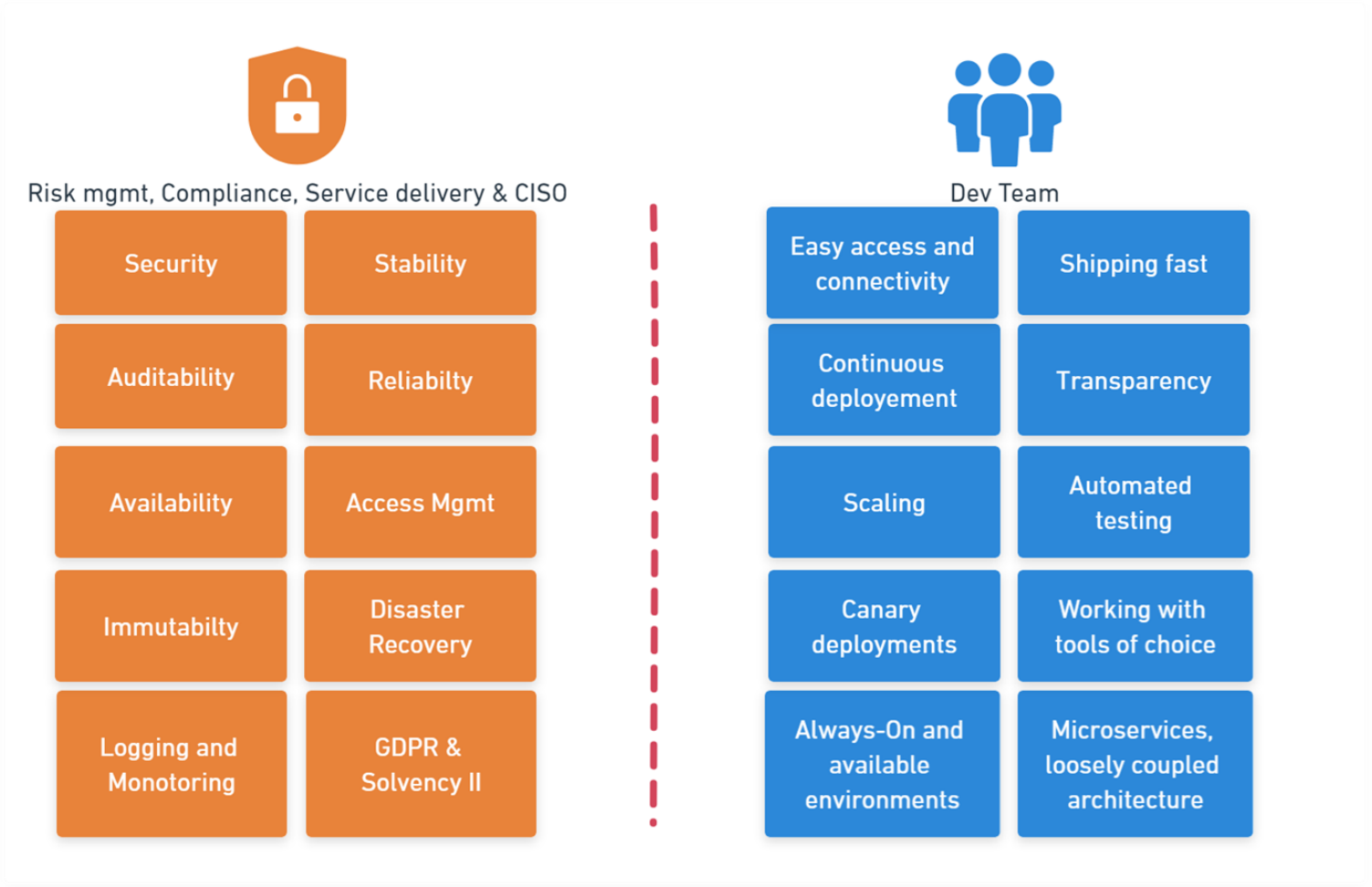
- 22 concurrent processes
- RPA can use 17 different business applications in IF
- 100+ Business Analysts trained across Nordics
- Image recognition, speech to text analysis, chatbots etc.

## Decommissioning legacy

- Waypoint
- Transition to cloud
- Decoupled architecture

# Constraints

# Balancing act



# Constraints

- Constraints in If Insurance
  - Concerns about networking and connectivity - solved by implementation together with network team using the hub/spoke model
  - Firewall – can we route all egress and ingress traffic via a central FW
  - Secure exposure (API & WAF capabilities) - connect cluster to central services being built by network team and digital solutions team
  - DMZ approach - Can it be done in some other way?
  - Https demands - Expose only https endpoints
  - Not expose cluster API to Internet - Private cluster
  - Container security – Scanning and policy implementations
  - Access to clusters – implement a division of concerns and integrate with identity provider
  - Shift left – can we introduce new ways and checks in build process for dev teams

- Starting point
  - Was to question many of the truths and anchor and find new solutions to old problems.

# Audiability and tracability

- Build it into the developer flow
- Ease the auditing process
- Quickly be able to rollback

# The starting point

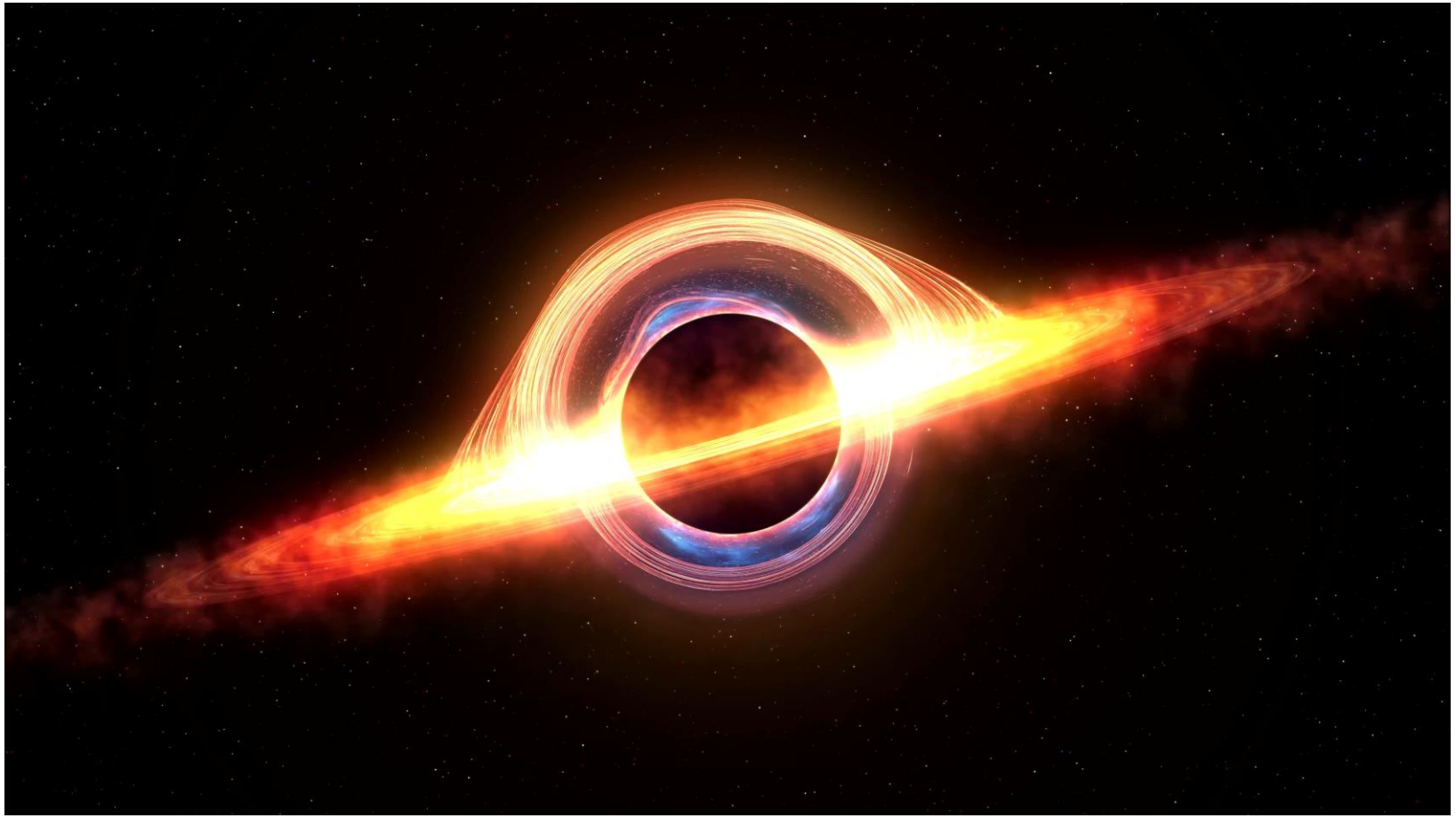
# The starting point

- Interviews
- Managing K8s takes a lot of work
- File->New->K8s does not cut it

# The starting point

- On Prem connectivity?
- How many environments?
  - Why that many?
- AuthN/AuthZ
- Monitoring
- Event, Alarm and Incident handling
- Scaling
- Security
- …

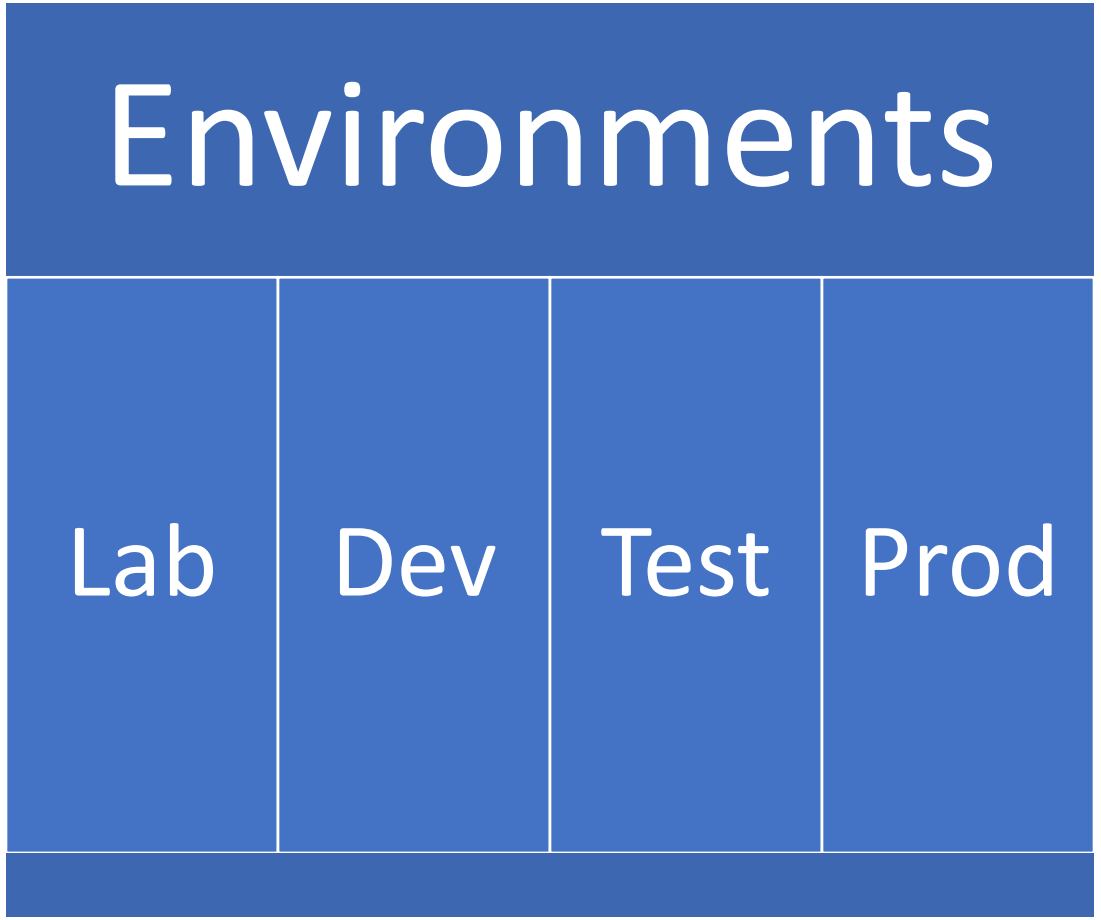
# Black Hole



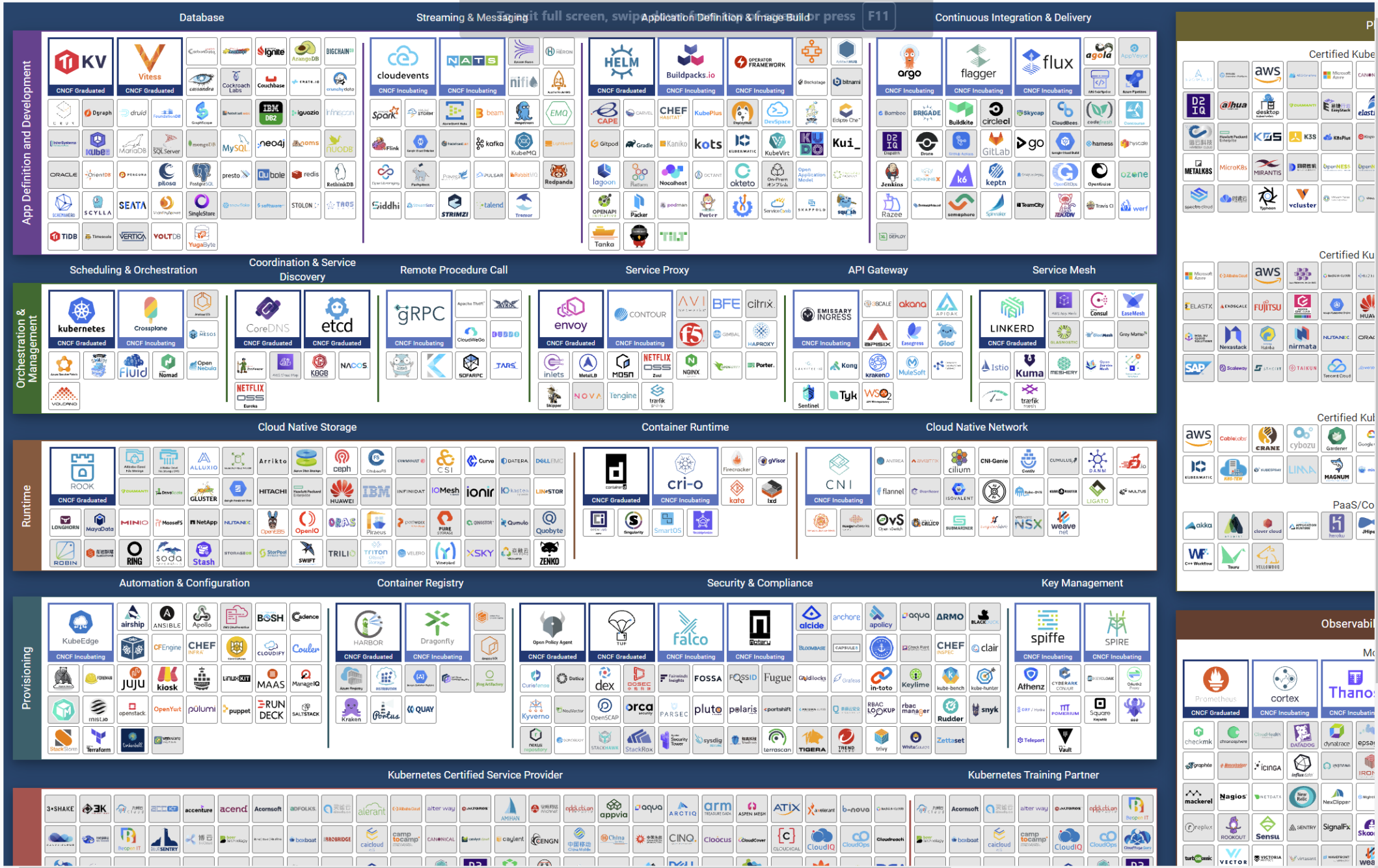
Do you want XYZ that will increase security?

# Environments

- Varied from team to team
- Common theme: many environments
- Focus on stabilization
- Friendly challenge



# Guiding Principles


Database
Streaming & Messaging
Application Definition & Image Build
Continuous Integration & Delivery
App Definition and Development
Scheduling & Orchestration
Coordination & Service Discovery
Remote Procedure Call
Service Proxy
API Gateway
Service Mesh
Orchestration & Management
Cloud Native Storage
Container Runtime
Cloud Native Network
Runtime
Automation & Configuration
Container Registry
Security & Compliance
Key Management
Provisioning
Kubernetes Certified Service Provider
Kubernetes Training Partner
Certified Kube
Certified Ku
Certified Kul
PaaS/Co
Observabi
Mo

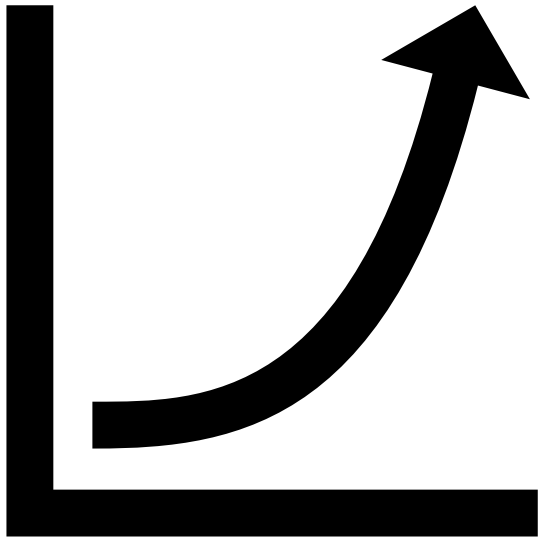
# Principles

- Aspect Oriented Programming
- Hard to make mistakes
- Maintenance Cost > Development Cost
- Reduce complexity
- Ubiquitous language
- __ as Code
- Minimize developer friction

# Building & managing the Platform

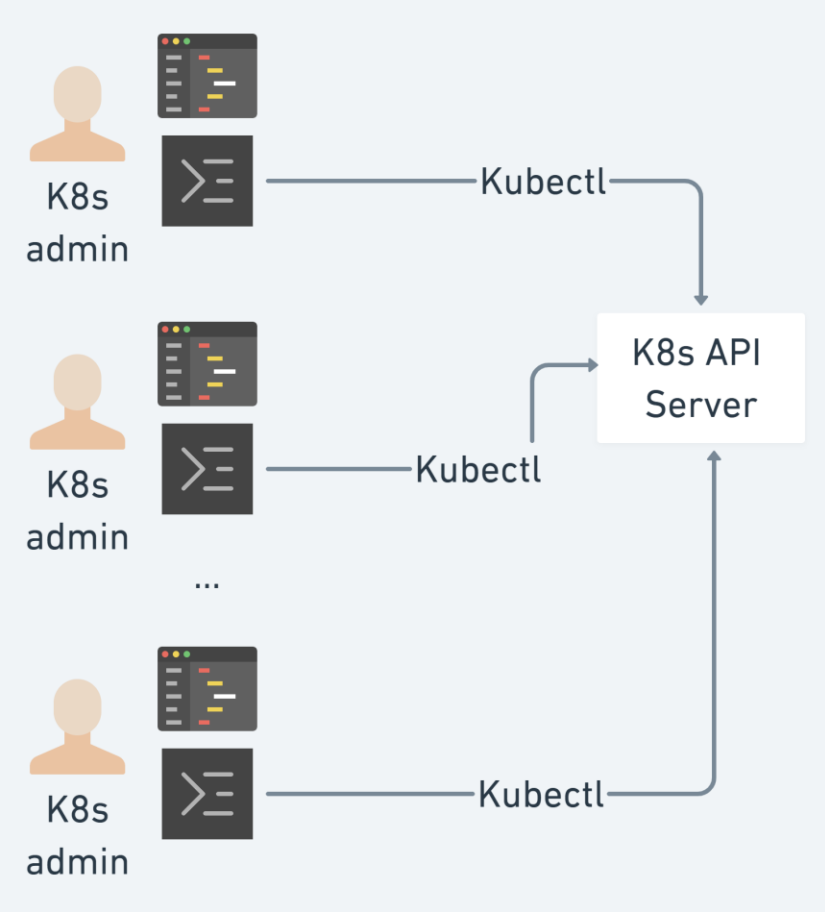
# The problem of scale

- Number of people, platform team size
- # Clusters
- # Environments



# The problem of scaling the team

- Race conditions on changes
- Disaster recovery scenario
  - New API servers in cloud environments
- No tooling to enforce QAs
- Difficult to get an overview over changes
- Missing audit trail

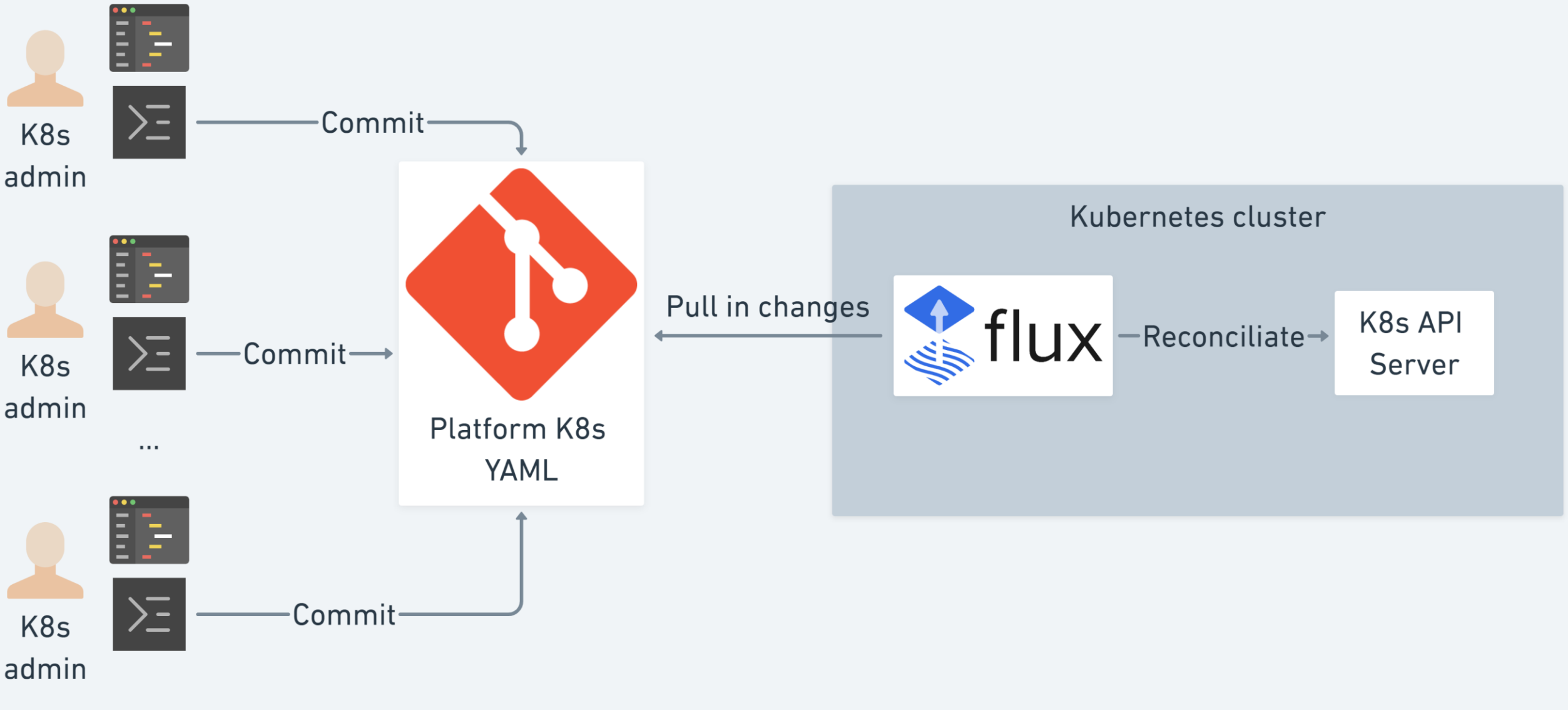


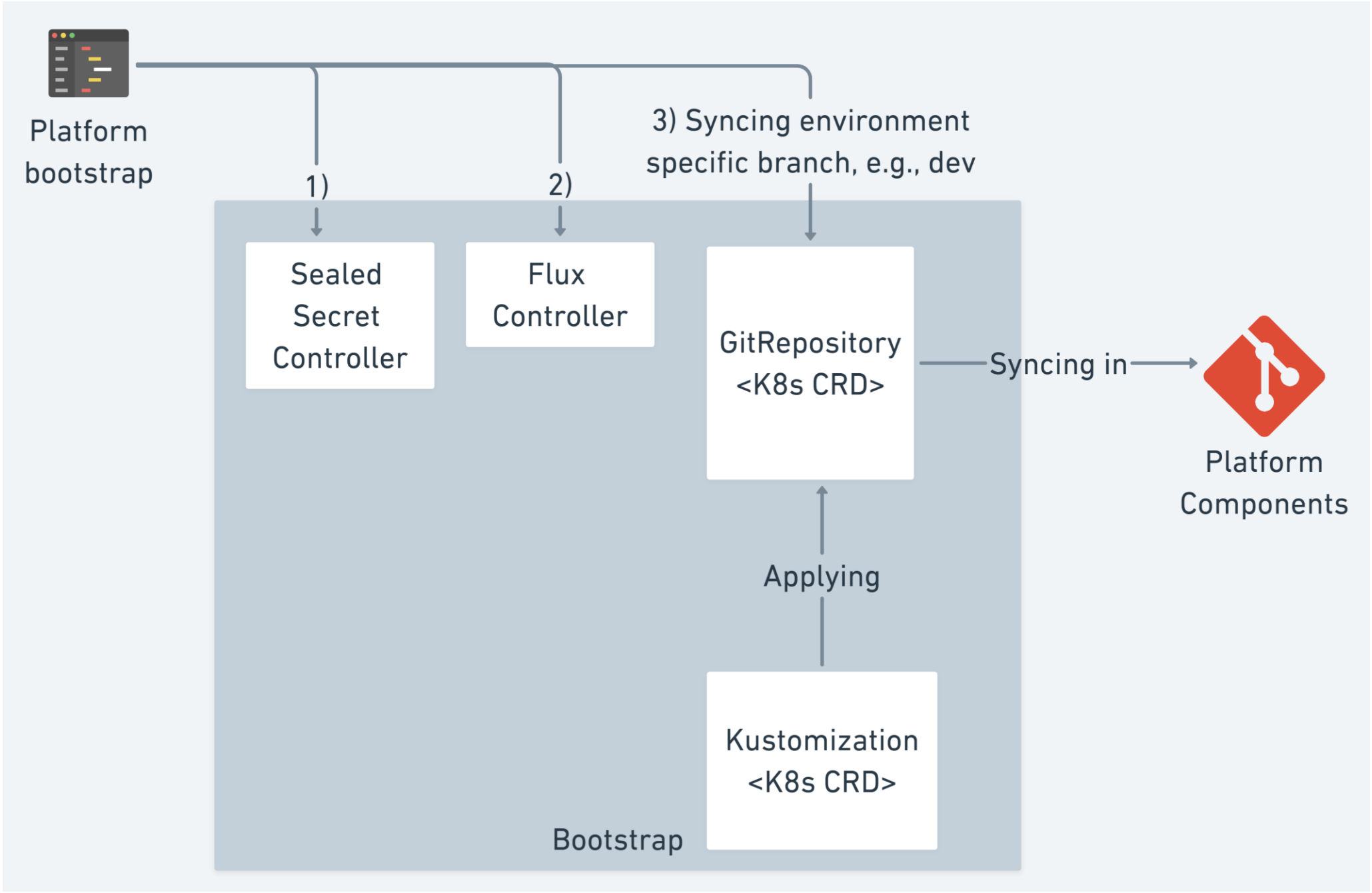
# GitOps

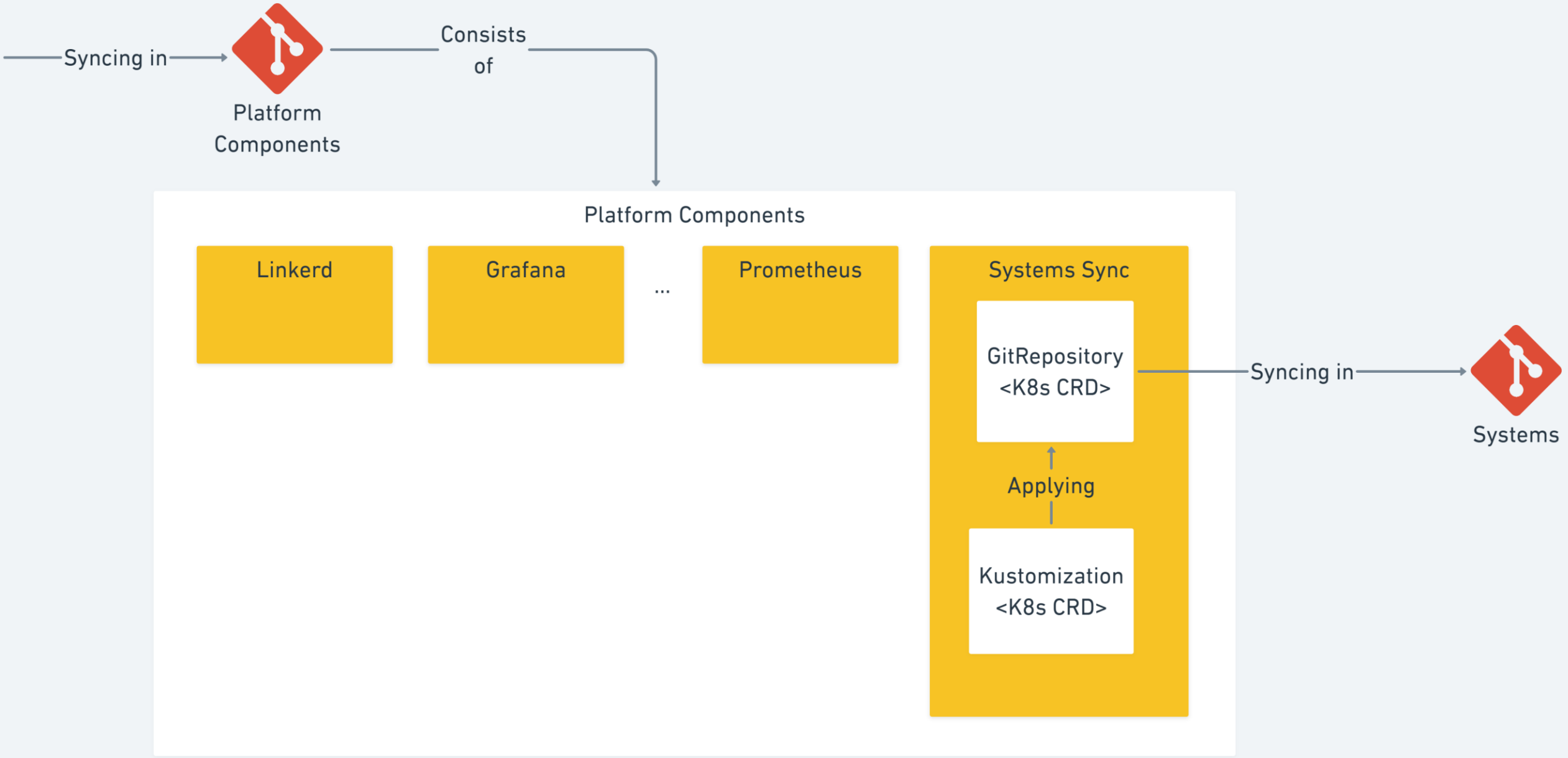
- Platform and teams
- An operating model
  - Push to Pull based model
  - Sharp separation between CI & CD
- Infrastructure vs Configuration

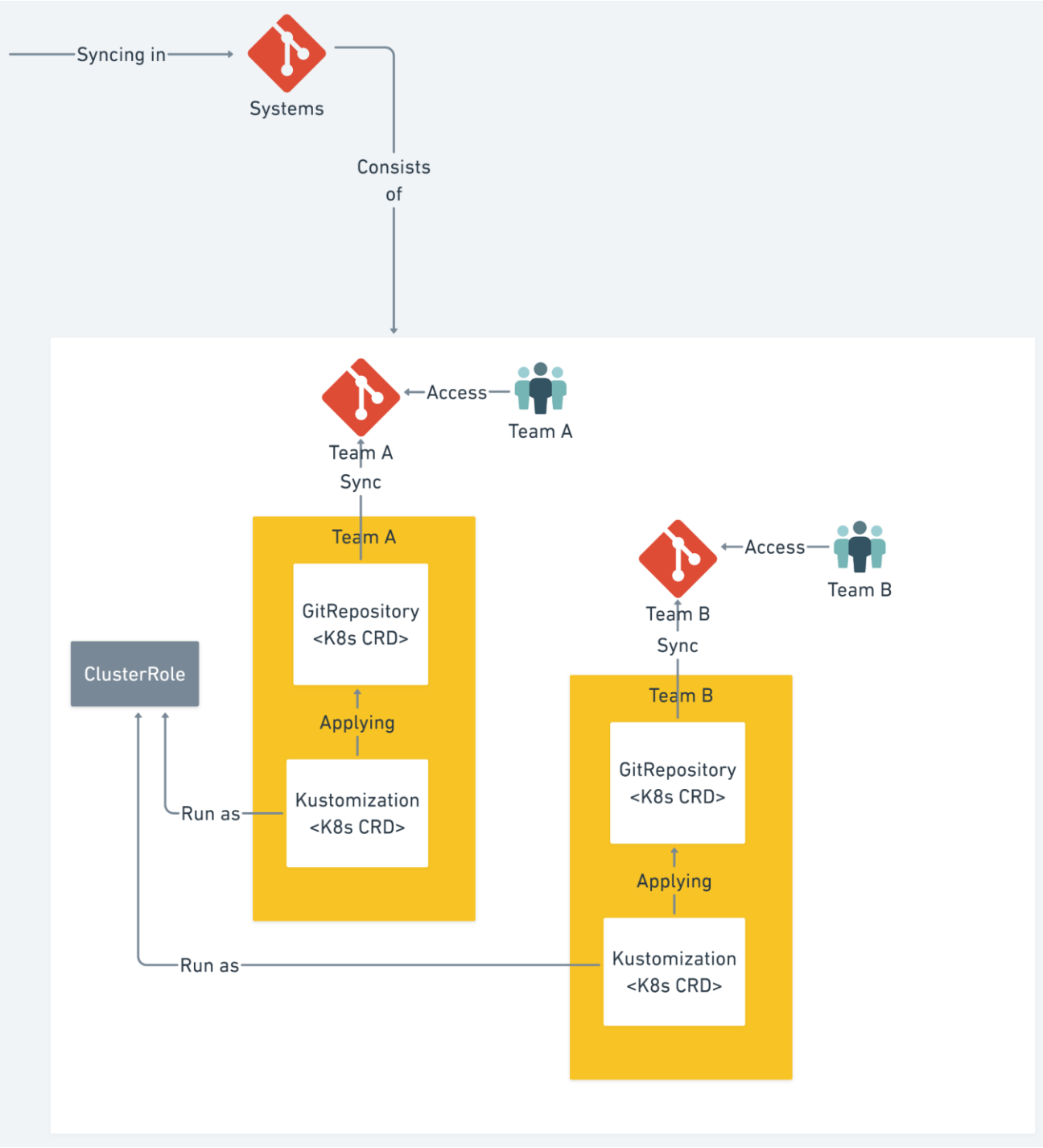
Bitnami Sealed Secret

flux


K8s admin
Commit
K8s admin
Commit
...
K8s admin
Commit
Platform K8s YAML
Pull in changes
Kubernetes cluster
flux
Reconciliate
K8s API Server


Platform
bootstrap
1)
2)
3) Syncing environment
specific branch, e.g., dev
Sealed
Secret
Controller
Flux
Controller
GitRepository
<K8s CRD>
Syncing in
Platform
Components
Applying
Kustomization
<K8s CRD>
Bootstrap


Syncing in
Platform Components
Consists of
Platform Components
Linkerd
Grafana
...
Prometheus
Systems Sync
GitRepository <K8s CRD>
Applying
Kustomization <K8s CRD>
Syncing in
Systems


Syncing in
Systems
Consists of
Access
Team A
Team A
Sync
Team A
GitRepository <K8s CRD>
Applying
Kustomization <K8s CRD>
ClusterRole
Run as
Access
Team B
Team B
Sync
Team B
GitRepository <K8s CRD>
Applying
Kustomization <K8s CRD>
Run as

# Disaster recovery

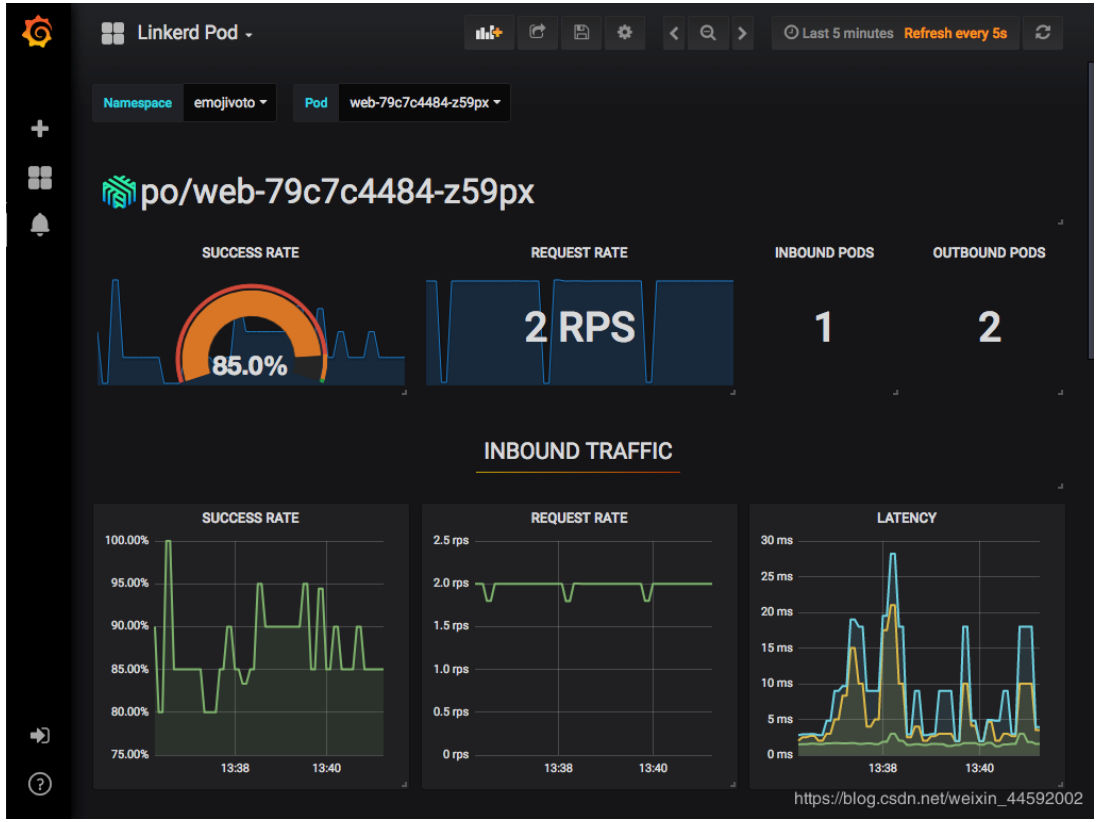
- Working all night story
- Train for it
- First training session ≠ first live disaster

# Service Mesh…really?

- Is K8s not complicated enough?
- We wanted to solve:
  - Network traffic encryption
  - Automatic insights
- Why did we choose Linkerd?

# The biggest benefits

- Network insights
- Metrics
- Main source of alarms



# Onboarding of teams

- Sharp separation between CI and CD
- Rethink monitoring, move towards metrics
- Rethink number of environments
- Scalability and disaster recovery
- Stateful vs stateless

# QA