



# Using Azure Kubernetes Service to enable Developer Experience and Regulatory Compliance

2025

# Toni Ylenius

## Expert Developer, Lead Developer

- Work Interests: Open Source Platforms
- Other: Family, Outdoors

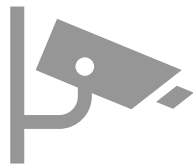


It is all about

# AUTOMATION

to achieve consistency and enabling change,  
that will enable great Developer Experience  
and Secure / Compliant solutions

# What means Regulations and Compliance



Security Regulations (NIS2)



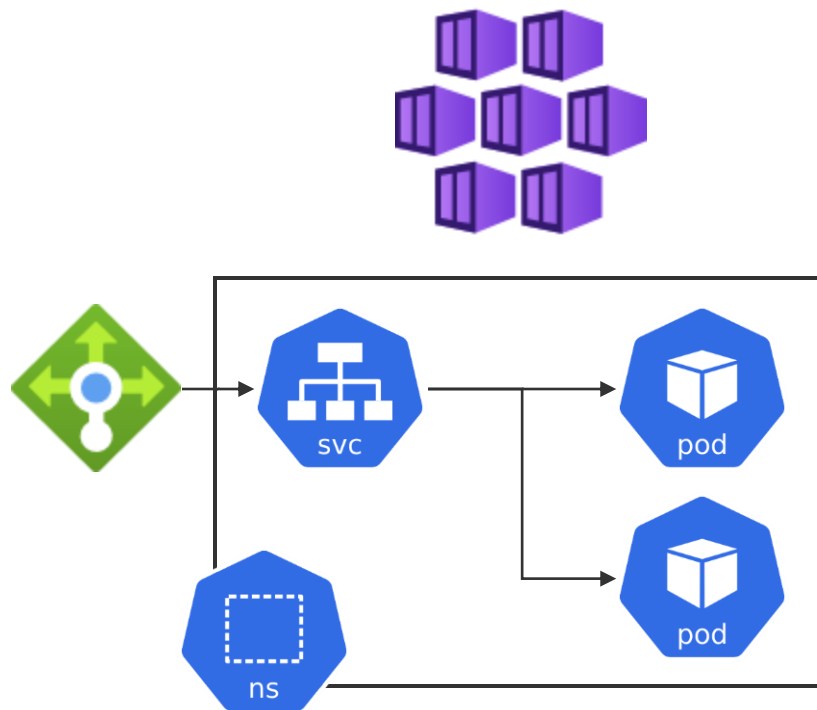
DORA (Digital Operational  
Resilience Act)



Risk Management

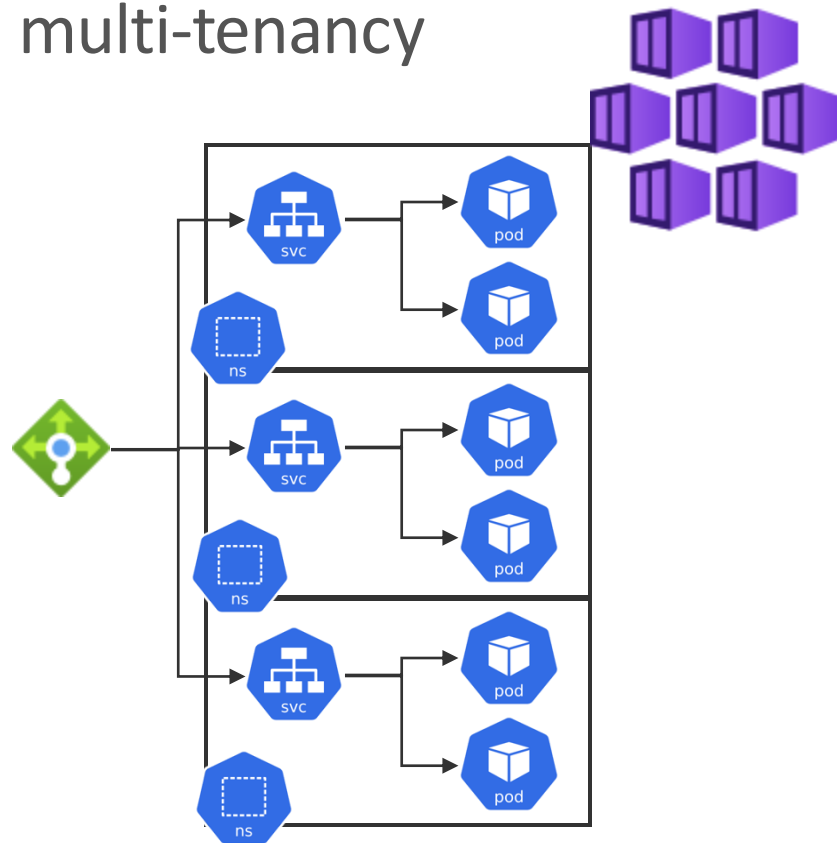
# Azure Kubernetes Service

- Benefits
  - Container based workflow ensures reproducibility in various environments
  - Abstracts Cloud Hardware: VMs, Ingress networking, Scaling



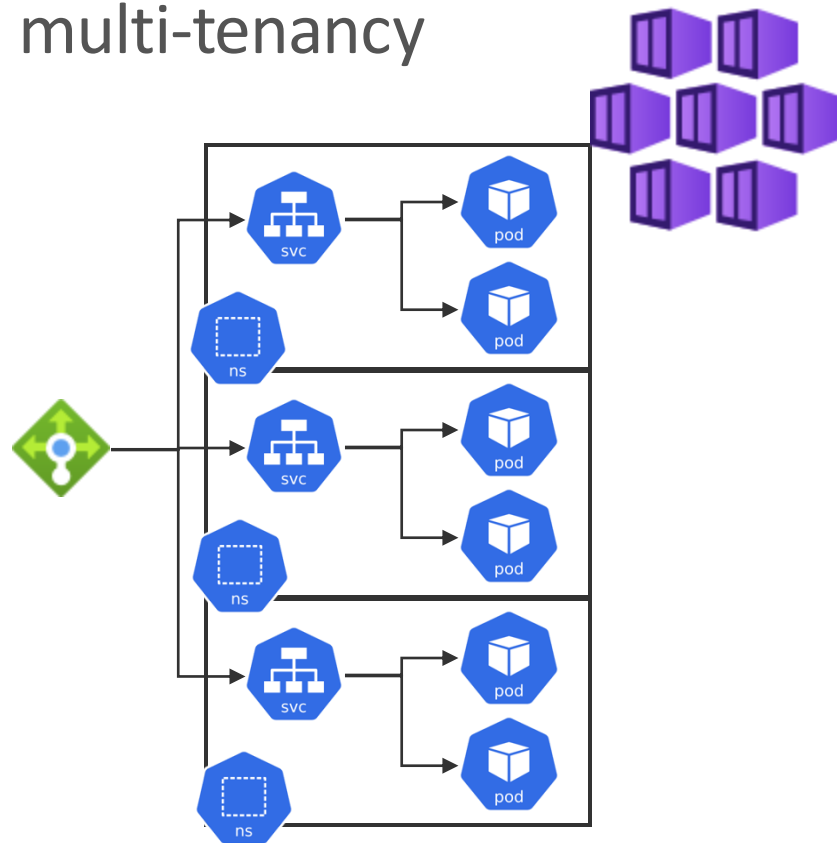
# Azure Kubernetes Service and multi-tenancy

- Benefits
  - Re-use of the Infrastructure Resources
  - Re-use of the Infrastructure Configuration, fewer environments to manage



# Azure Kubernetes Service and multi-tenancy

- Namespaces
- Azure RBAC
- Ingress
- Private Clusters
- Azure Policies
- Quotas
- Network Policies
- ....



# Challenges ahead...

- The list mentioned need to be managed
- Applications need to have best practices in place
- (note, AKS Automatic to the rescue)
- Tooling spree: Kustomize, Helm, Istio, Linkerd, Cilium, ExternalDNS, KEDA, ArgoCD, Flux, cert-manager, External Secrets, Prometheus
- How about other Azure resources and integrations to them



# DevOps vs Platform Engineering

## Standards

- Technology stack
- **Templates, re-usability**
- Platform as Product

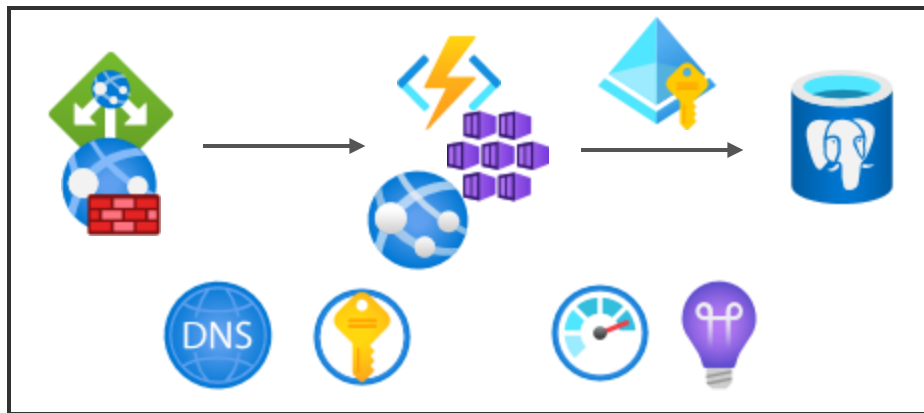
## Guardrails

- Policies
- **Automation**
- Enforced security
- Observability

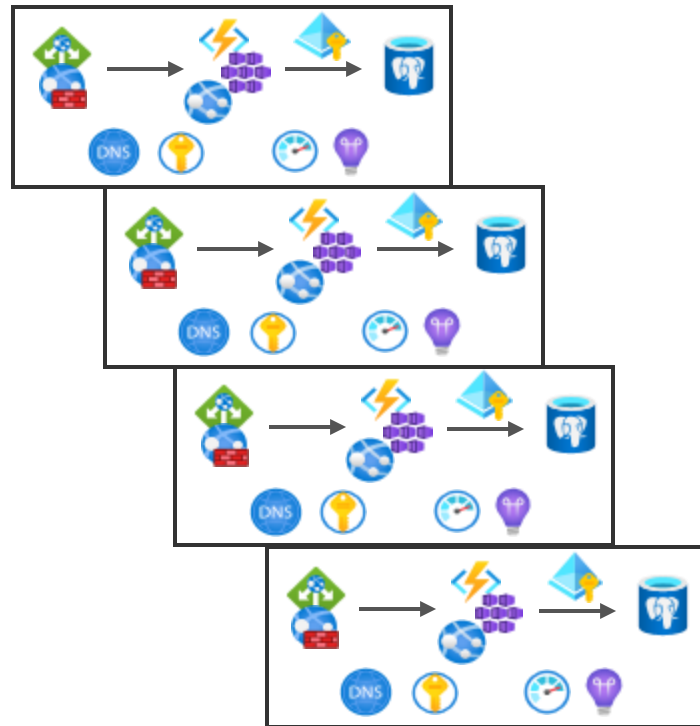
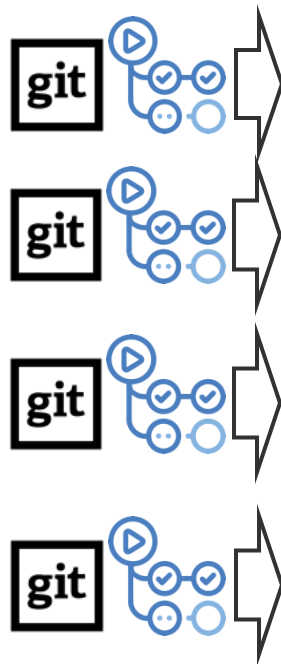
# Infrastructure as Code



## Imaginary Example Application



# Infrastructure as Code

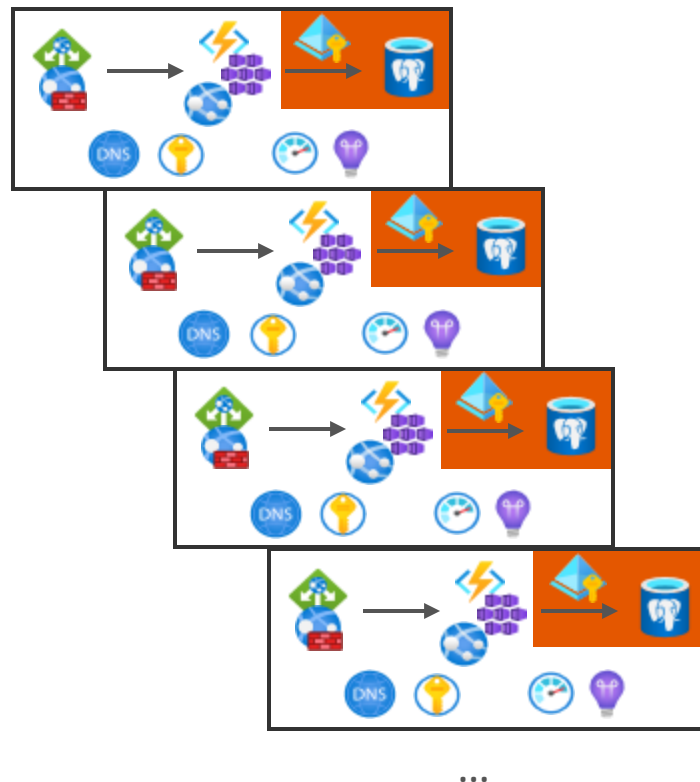
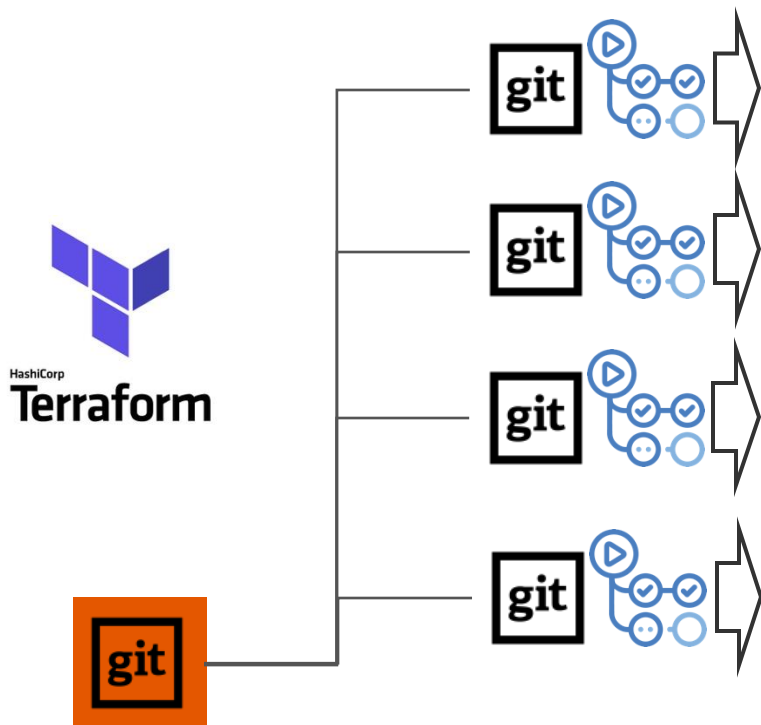


...

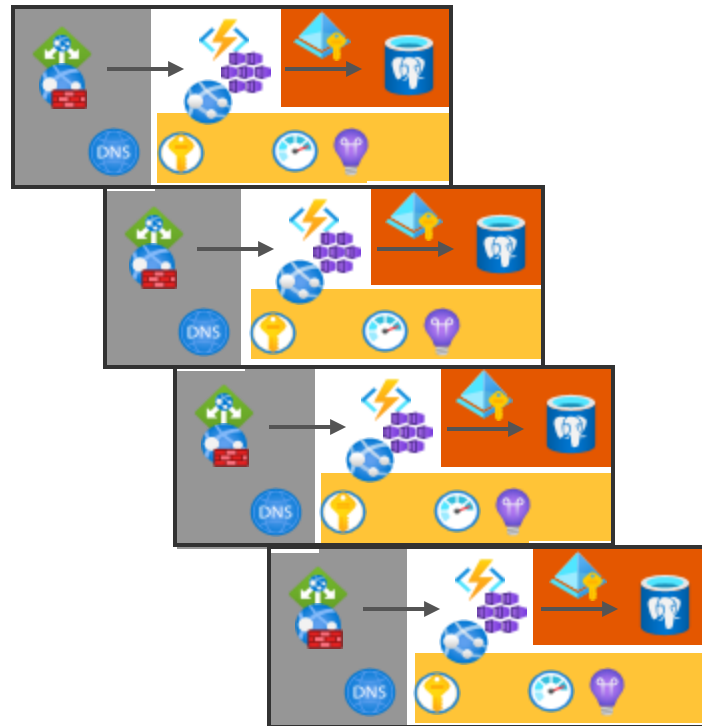
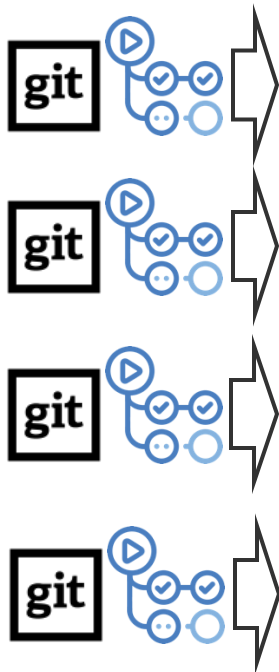
# Changes will eventually happen

1. Network changes
2. Database hardening need to be applied
3. App Service migration to v4

# Infrastructure as Code Library

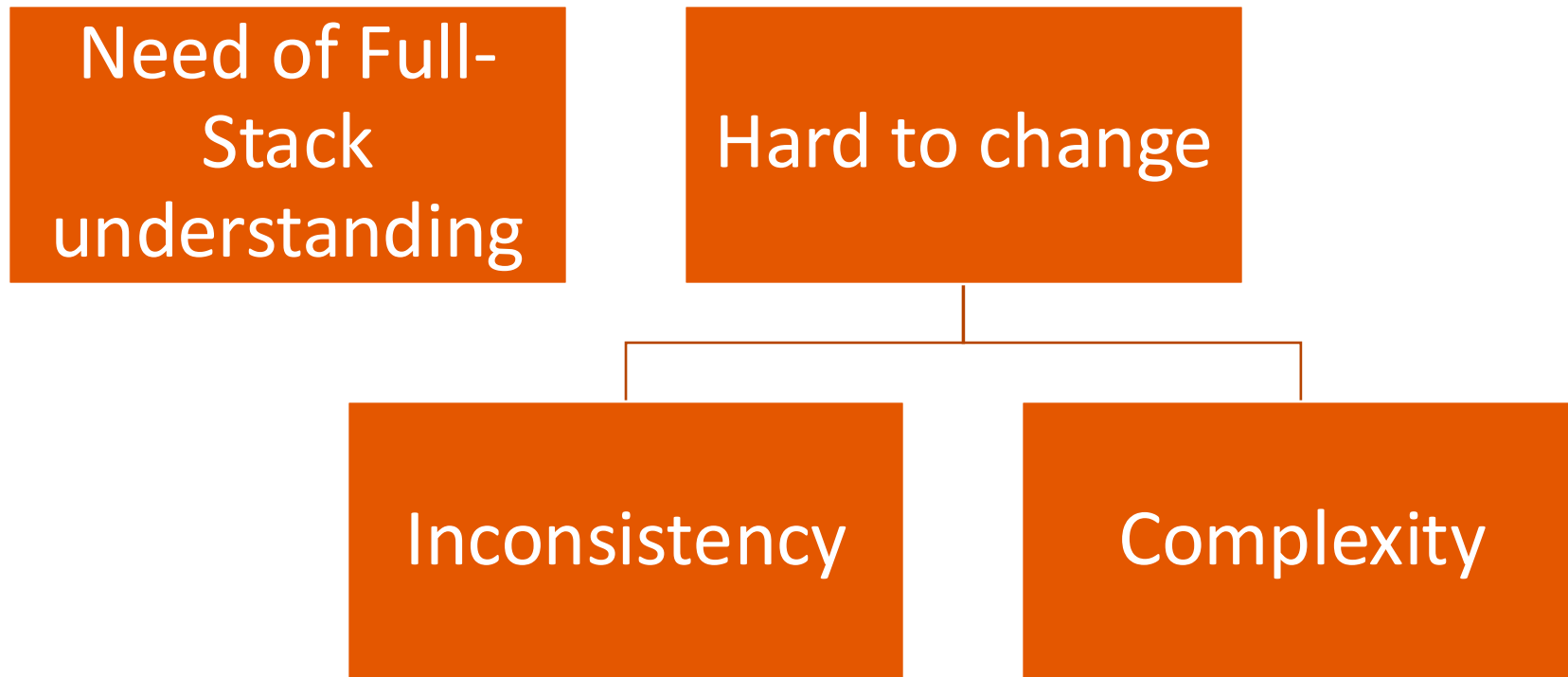


# Infrastructure as Code Library



...

Already lots of challenges solved



# Our Examples

1. Network changes
2. Database hardening need to be applied
3. App Service migration to v4

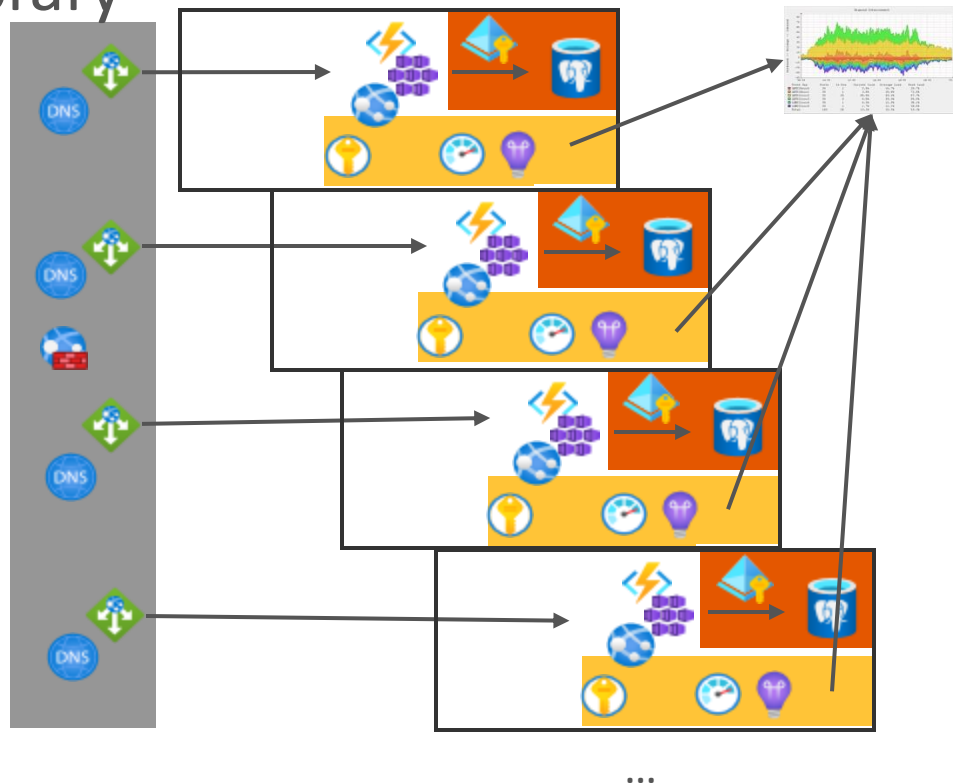
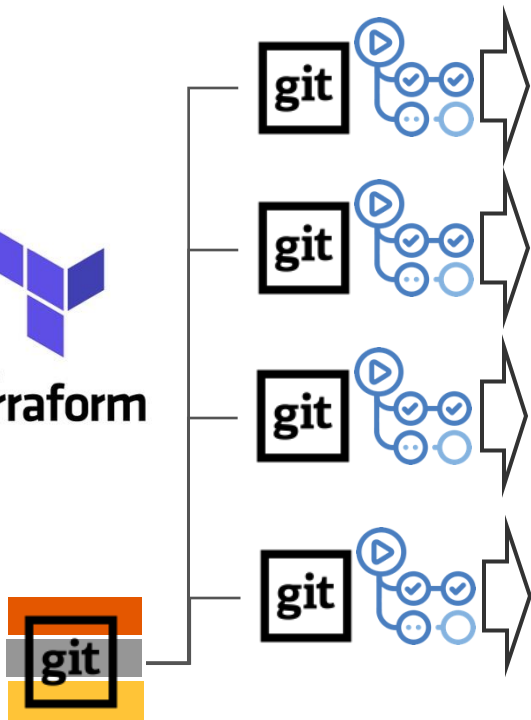
-> Quite Good Developer Experience Achieved



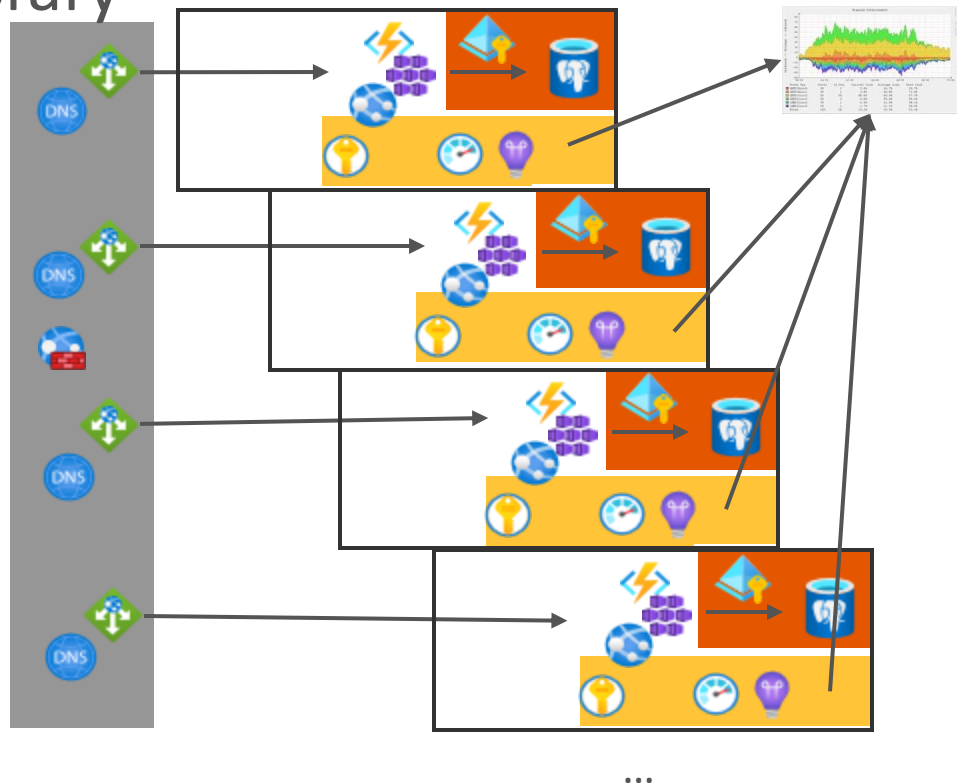
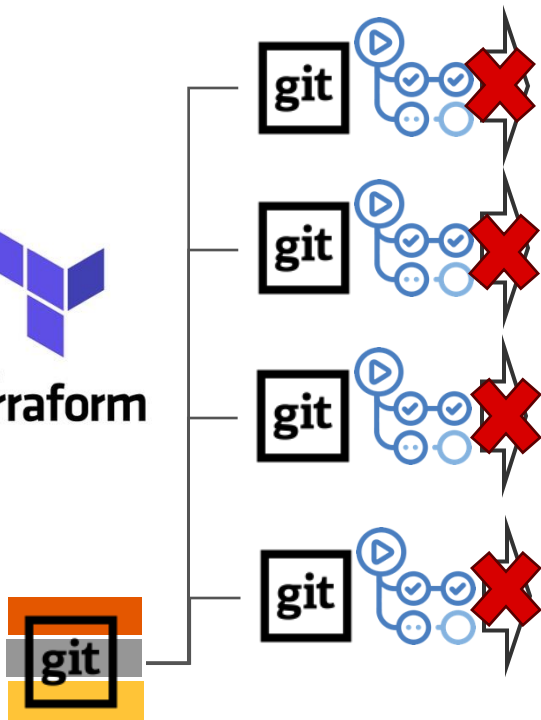
# Welcome to an Enterprise Environment

- Lots of teams, and changes are slow
- Cannot change everything at once. Need to integrate old systems that are not Cloud Native
- And you will always have some SaaS services
- Regulations and rules (and people) might require approvals to be added
- Knowhow need to be centralized

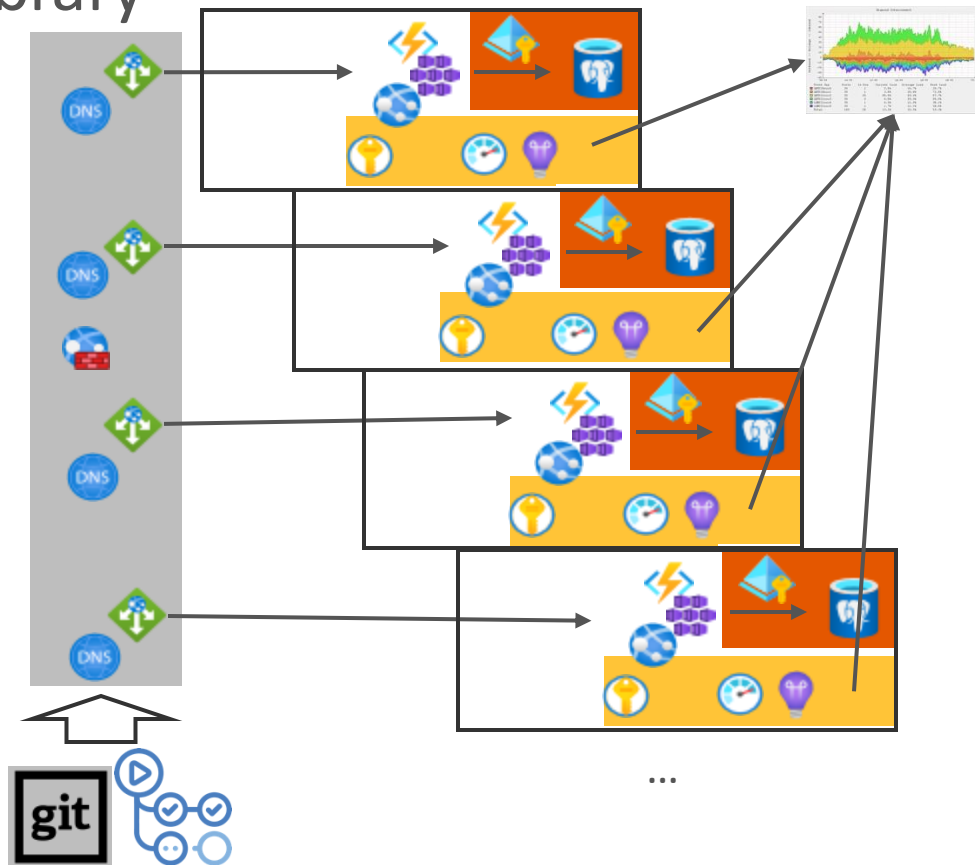
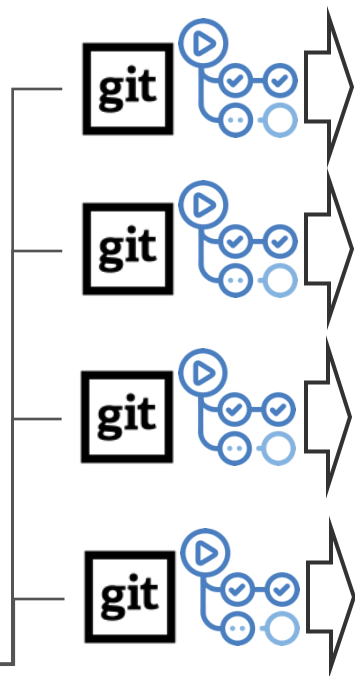
# Infrastructure as Code Library



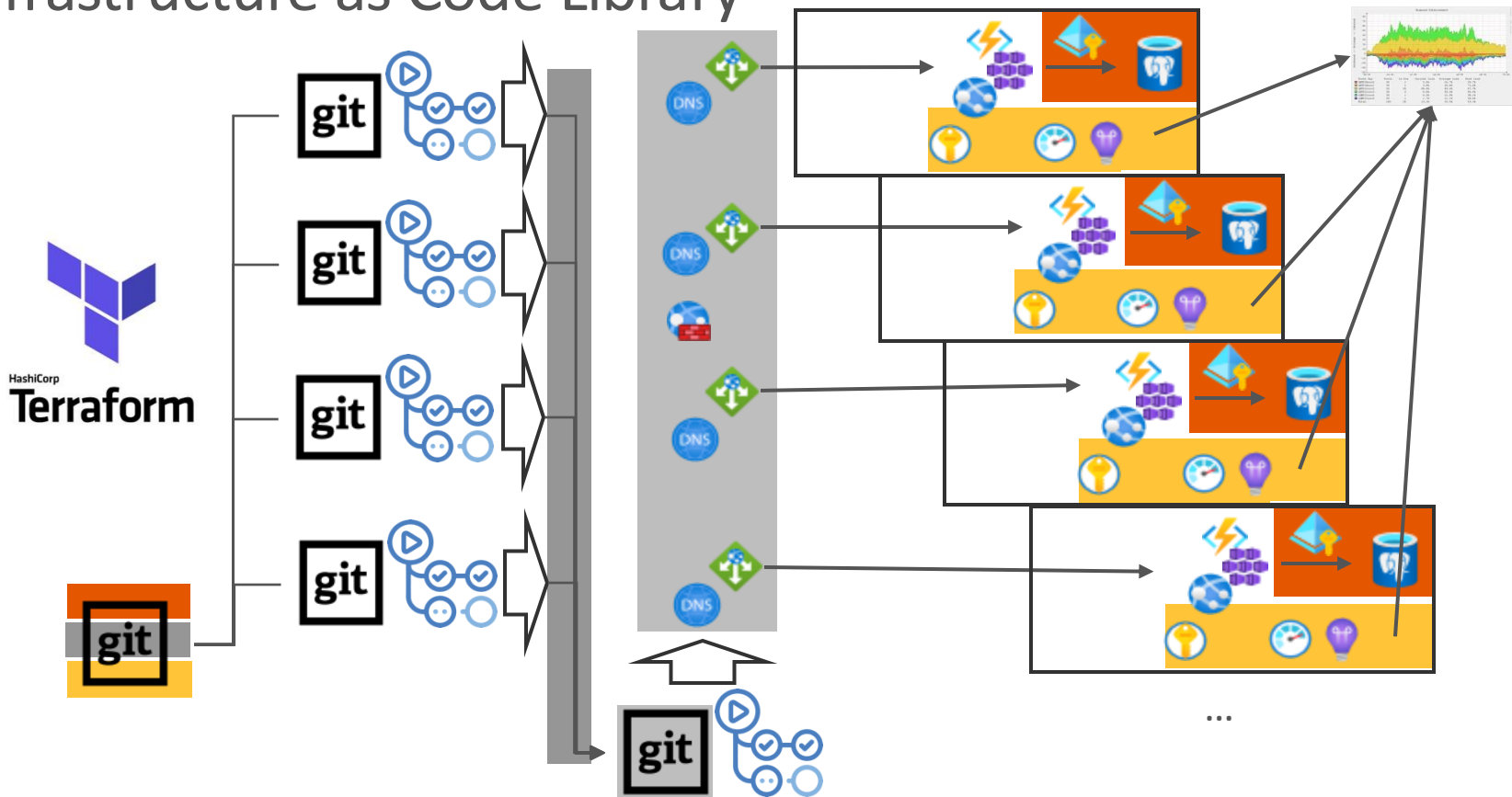
# Infrastructure as Code Library



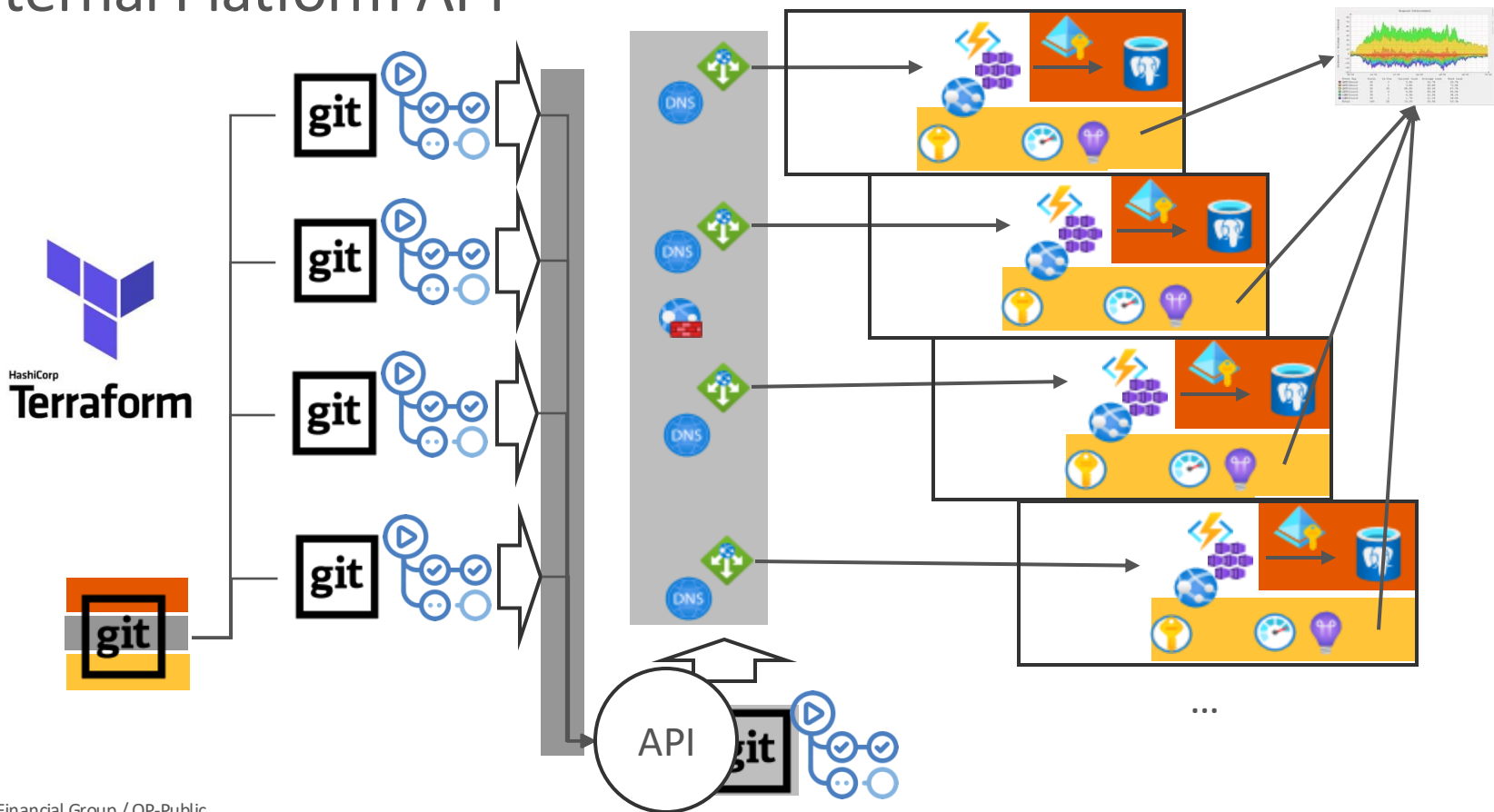
# Infrastructure as Code Library



# Infrastructure as Code Library



# Internal Platform API



# You start building Internal Platform APIs

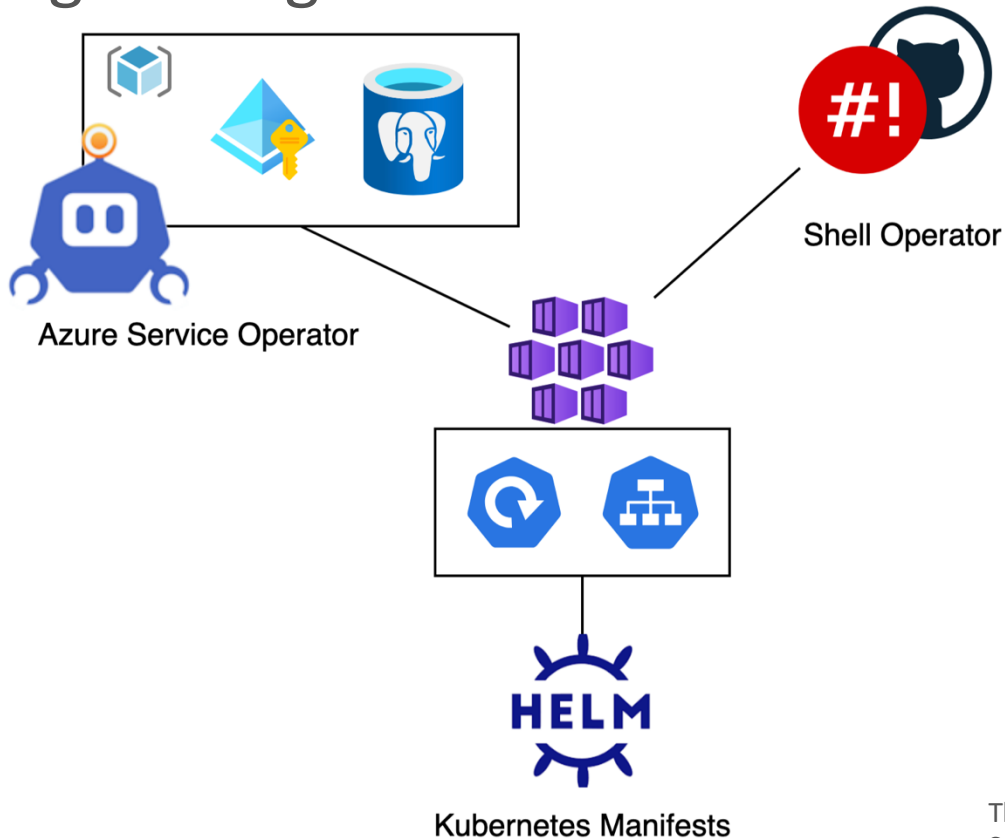


To retain DX and Automation level, but still ensuring Controlled architecture, you start building internal APIs



Time for a Demo

# Platform Engineering with Kubernetes DEMO



The diagram is from the demo repository.  
See the end of the presentation



# Platform Engineering with AKS (Kubernetes)

Container is good enough abstraction for Applications

Extensible API with CRDs

- Provides DB, RBAC, Tenants, Runtime

There is the ecosystem of tools, eg. Crossplane

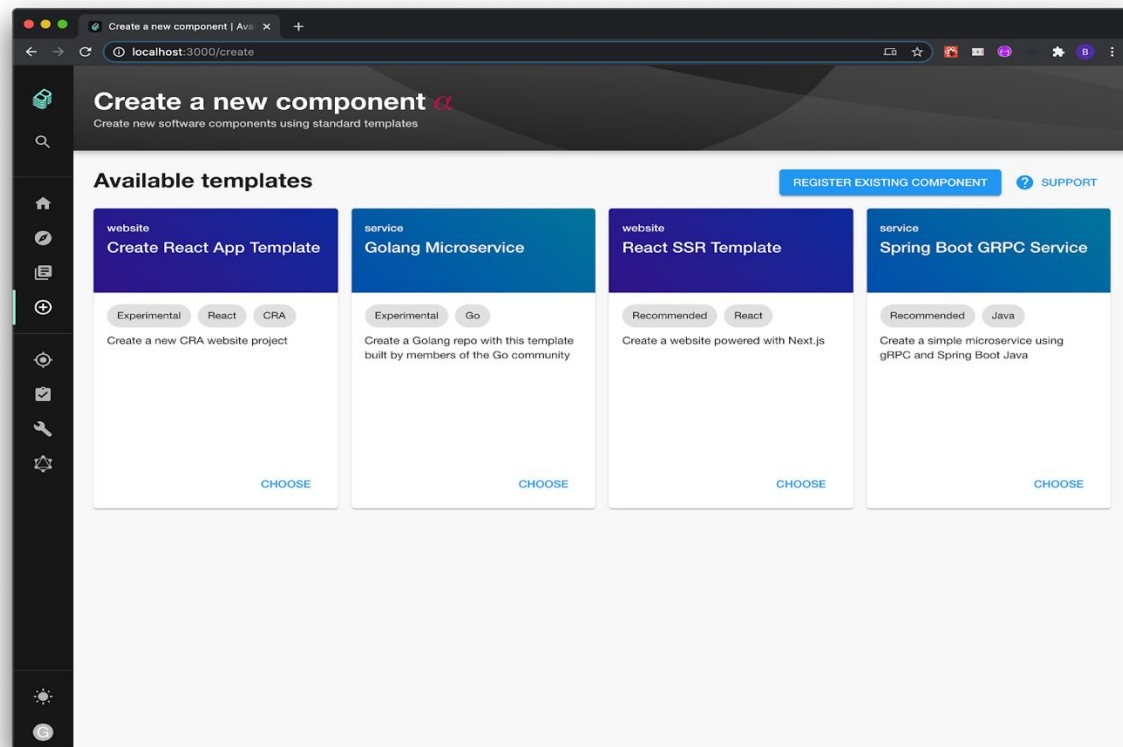
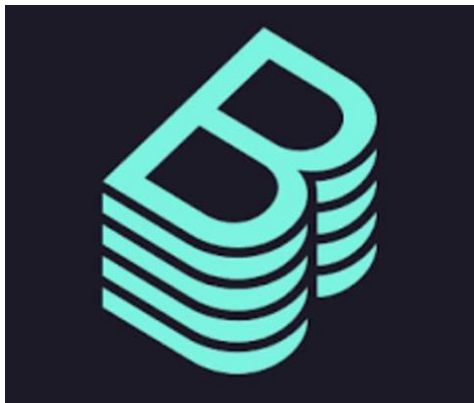
## Compared to old Internal Platforms (eg. JVM Farms)



## Compared to PaaS

- Either one need to adapt, the platform or the company
- There is no single size fit all, or the platform must be very flexible
- Remember the Enterprise Environment and slowness

# Platform as Product: Internal Developer Portal



It is all about

# AUTOMATION

to achieve consistency and enabling change,  
that will enable great Developer Experience  
and Secure / Compliant solutions

# Some References

- Demo <https://github.com/toddnai/pe-automation-demo-2024>
- Platform Engineering: Creating Scalable and Resilient Systems  
<https://build.microsoft.com/en-US/sessions/05883e89-2458-4a3b-94a8-c4472d6e8a2a>
- AWS and Platform Engineering  
<https://www.youtube.com/watch?v=TBzy0QPOyY4&list=PLj6h78yzYM2OyAZIMbJPOsamT2aLKZX6b&index=15>
- H&M's journey: <https://build.microsoft.com/en-US/sessions/a7e28d11-9427-4695-8738-3f6a8c8252df>
- Reddit's PE journey (in k8s context)  
<https://www.youtube.com/watch?v=ruto5Sak-jl>
- Daniel Bryant's KubeCon 2024 take on PE  
<https://www.youtube.com/watch?v=qhfQfQmnNd4>





Thank you