A story about
Platform Engineering

Niklas Voss
Google Cloud

Hamburg
Sep 12, 2023

Agenda

01 What is Platform Engineering

02 The fictional company

- Application Delivery
- Policies and Guardrails

03 Putting it all together

04 Kubernetes as IDP

Google Cloud

What is Platform Engineering

Google Cloud

A platform is...

"A foundation of **self-service APIs, tools, services, knowledge and support** which are arranged as a ***compelling* internal product**.

Autonomous delivery teams can make use of the platform to **deliver product features at a higher pace**, with **reduced coordination**"

– Evan Bottcher

"A **curated *experience*** for engineers (the customers of the platform)"

"...the purpose of a platform team is to enable stream-aligned teams to deliver work with substantial **autonomy**... The platform team provides internal services to **reduce the cognitive load** that would be required from stream-aligned teams to develop these underlying services"

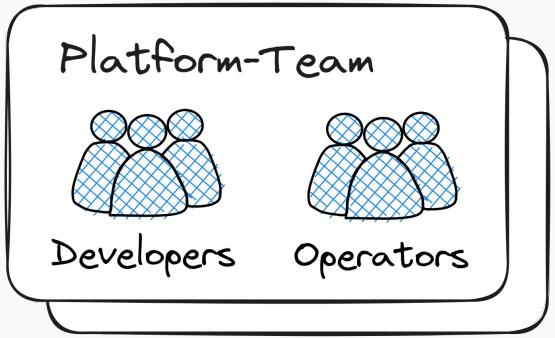
– Matthew Skelton (Team Topologies)

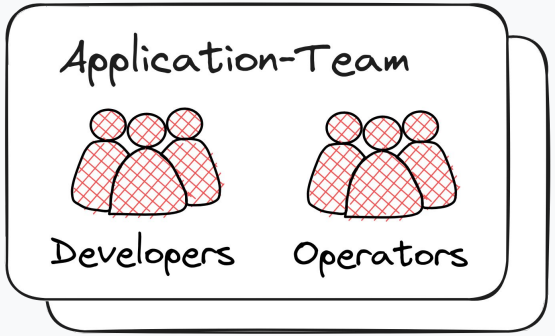
An Emerging Approach to Building Platforms

Platform Engineering Principles

- Developers use a **thinnest viable platform** to build out infrastructure
- **The platform is the product** and application teams are the customers
- Platform teams **own and build the toolchain** and **abstractions around complex infrastructure**
- The platform provides **end to end paved paths** (or golden paths)

Skelton, Matthew, and Manuel Pais. Team Topologies: Organizing Business and Technology Teams for Fast Flow. IT Revolution, 2019.





Google Cloud

Why Platform Engineering?

1

Reduce cognitive load, **lower burnout-risk**

2

Accelerate development and **faster *time to market* with lower costs**

3

Guardrails suited to your organization **improving security and governance**

Google Cloud

Why to build a platform on Kubernetes?

- **Cognitive load** high managing configuration unrelated to value-stream of application team
- **Standardize security, governance und compliance**
- **Specialize** to your own requirements

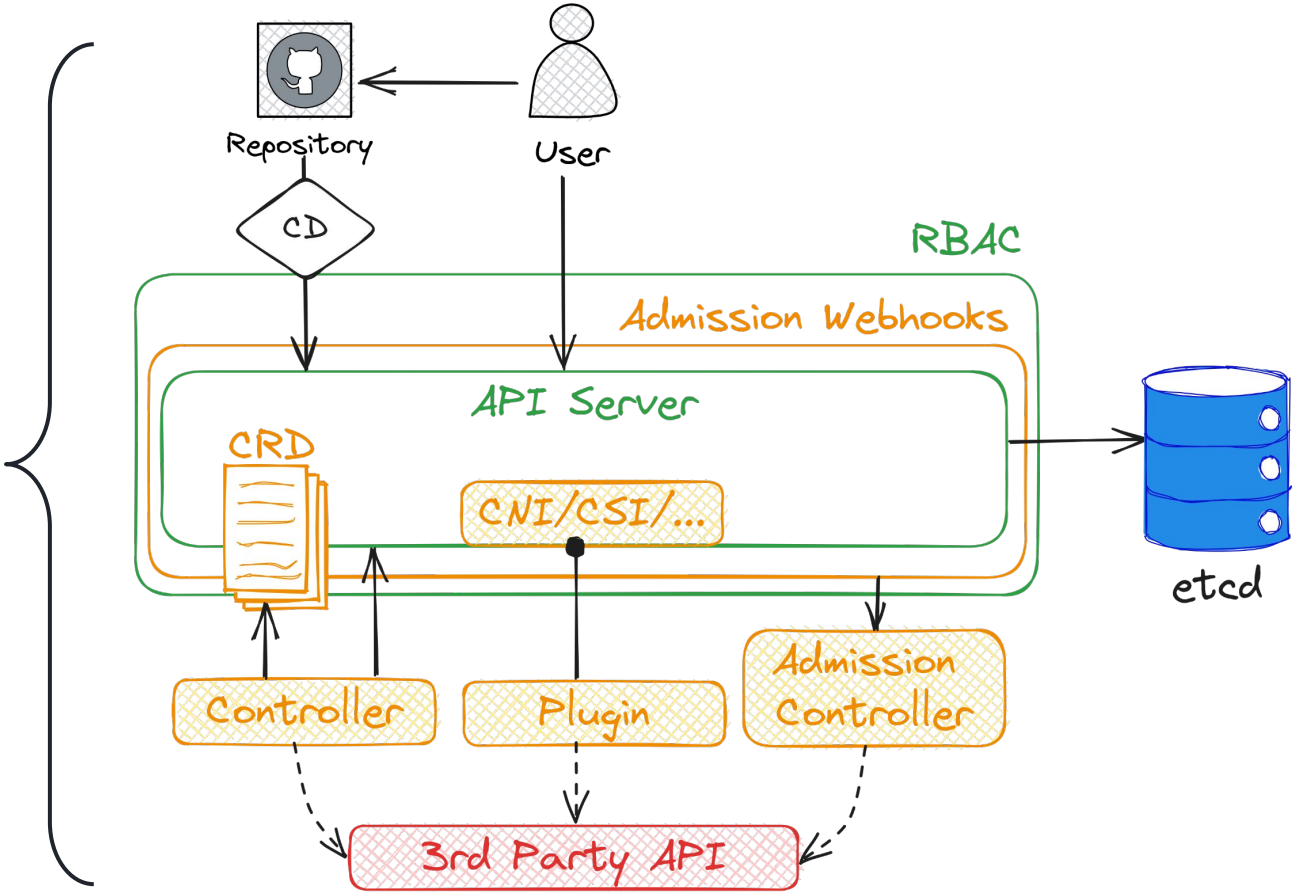
Kelsey Hightower: “Kubernetes is a platform for building platforms. It's a better place to start, and not the endgame.”

Google Cloud

How can Kubernetes be extended?

- **Custom resource definitions** (CRDs)
- **Admission webhooks:** Validating and Mutating
- **Plugins:** Device plugins, storage plugins (CSI), network plugins (CNI)

How can Kubernetes be extended?



Google Cloud

The fictional company

Google Cloud

Courtesy of OpenAI's DALL-E 2

- Fictional company is a **FinTech** called ~~Finicorn~~ Flinnifonch
- Organization has needs for **secure workflows** and **guardrails** due to regulation and compliance requirements
- Stance on vendors: **Commodities yes**, but avoid lock-in if sensible

Courtesy of OpenAI's DALL-E 2

- **Application-Teams:**
  - **Little to no knowledge provisioning infrastructure**
  - Intermediate Kubernetes knowledge
- **Platform-Group:**
  - Expert Kubernetes knowledge
  - **Experts in provisioning infrastructure**

Application Delivery

Google Cloud

- Started as Knative Build in 2018
- **Open-Source CI/CD platform** implementation
- **Open API spec** to describe CI/CD pipelines
- Governed by the **Continuous Delivery Foundation** (CDF)
- Vision and Mission: **Composable, Declarative, Reproducible and Cloud-Native**

Flinnifonch likes: Composability, Kubernetes-Native, Open API spec

Google Cloud

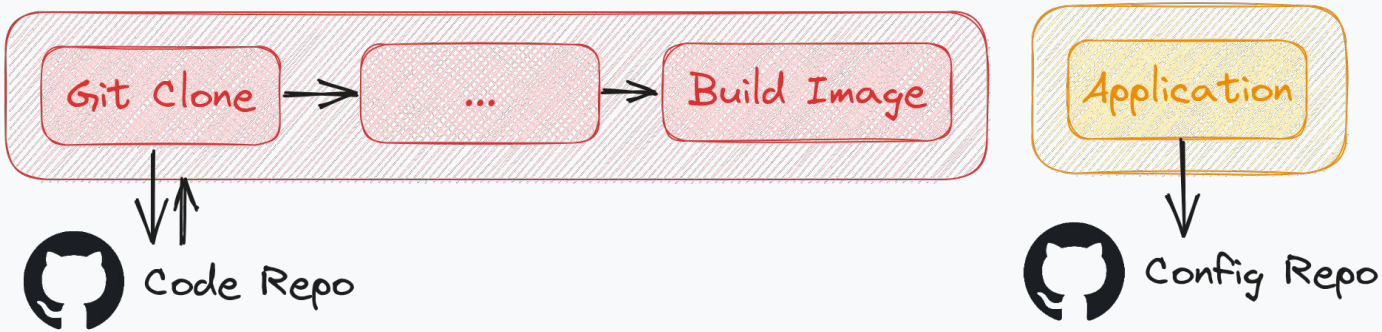
- ArgoCD was created by Intuit and donated to the CNCF in 2020
- **Automates deployment** of applications **to clusters**
- Follows **GitOps** operational framework
- **Enterprise-friendly** (auditability, compliance, security, RBAC, SSO)

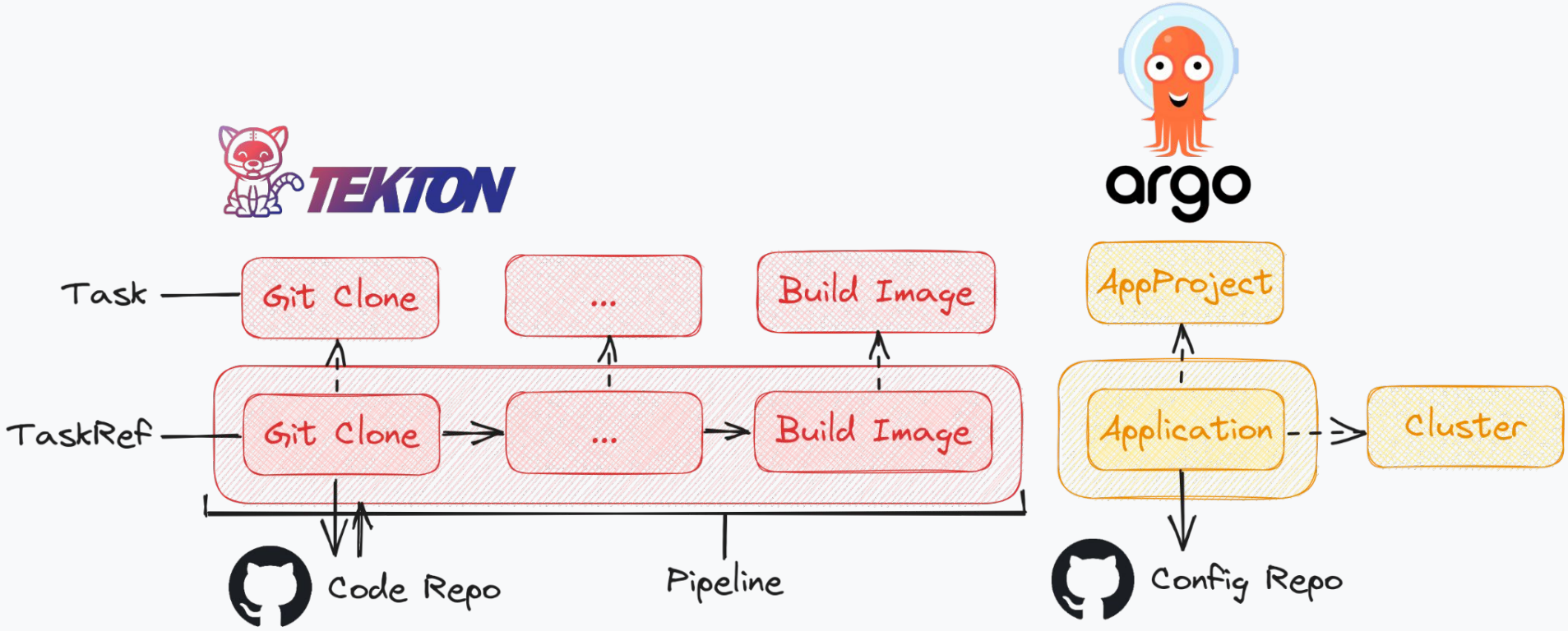
Flinnifonch likes: UI, DX

Google Cloud


TEKTON


argo


Git Clone
...
Build Image
Code Repo
Application
Config Repo



Google Cloud

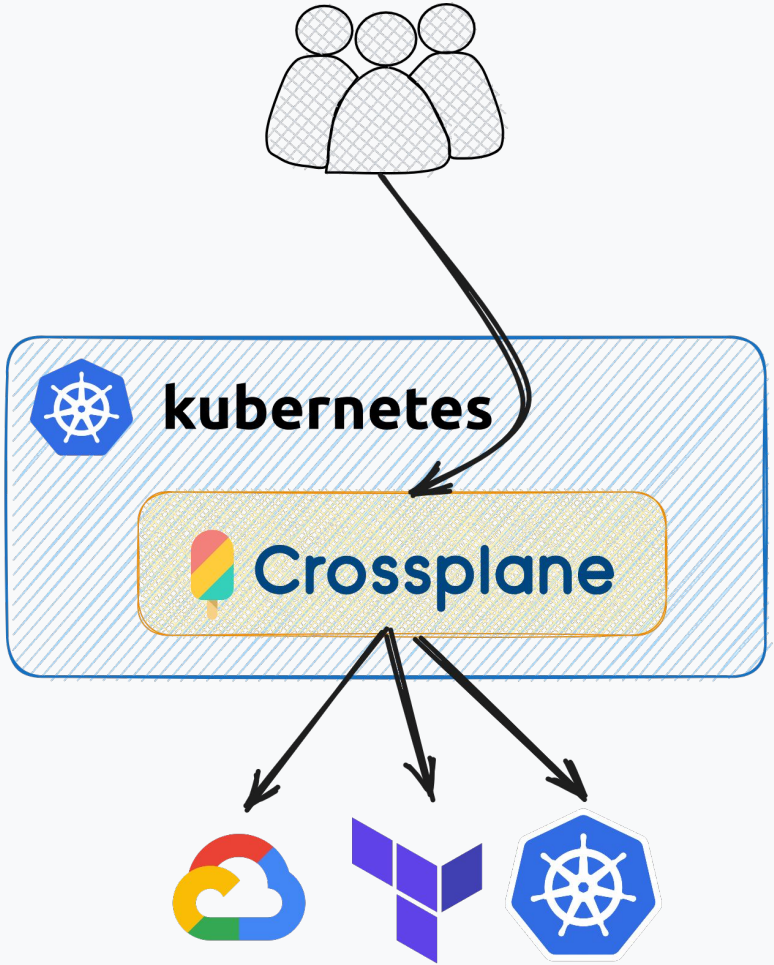
- **ArgoCD** can reconcile changes to **Kubernetes Resources**
- But what about **provisioning infrastructure** for our application?
- Or interacting with **external resources**?

Google Cloud

Crossplane

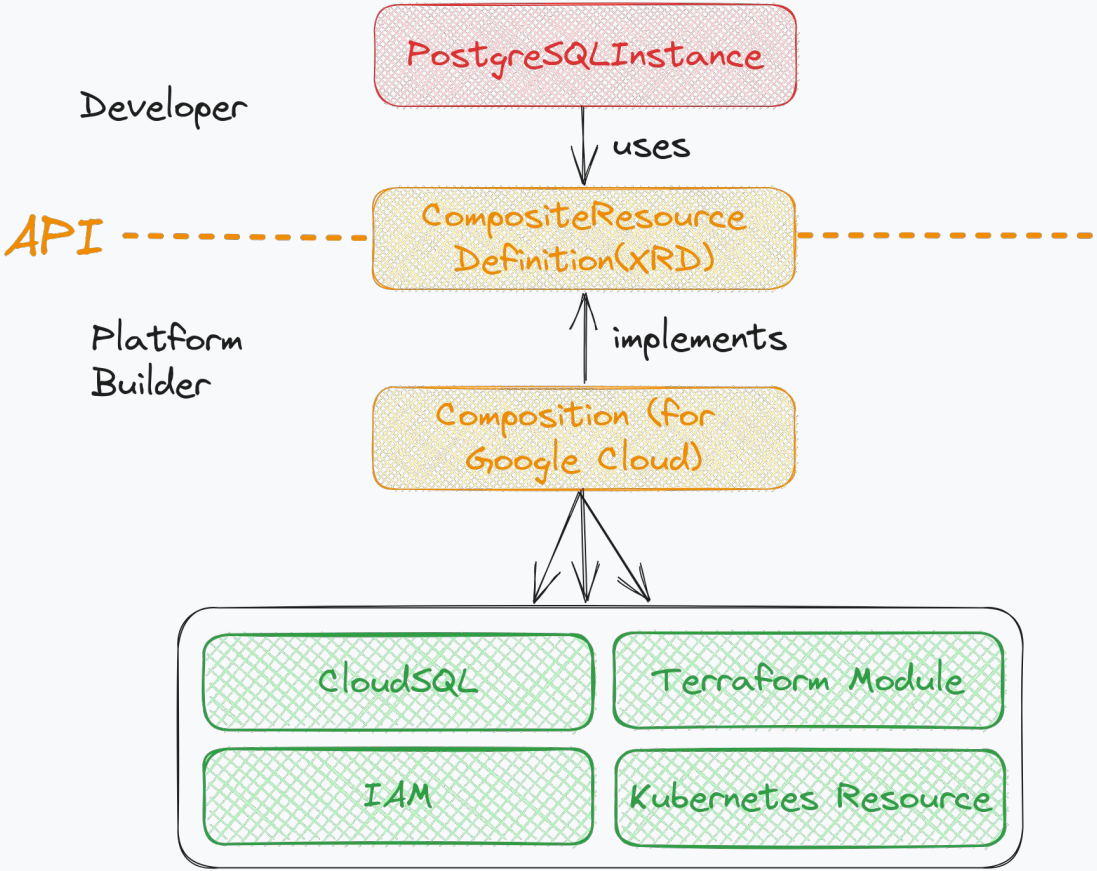
"**Crossplane is an open source, CNCF project built on the foundation of Kubernetes to orchestrate anything.** Encapsulate policies, permissions, and other guardrails behind a custom API line to enable your customers to self-service without needing to become an infrastructure expert."

– crossplane.io



Google Cloud

Crossplane



Flinnifonch verdict: ❤️

Google Cloud

Policies and Guardrails

Google Cloud

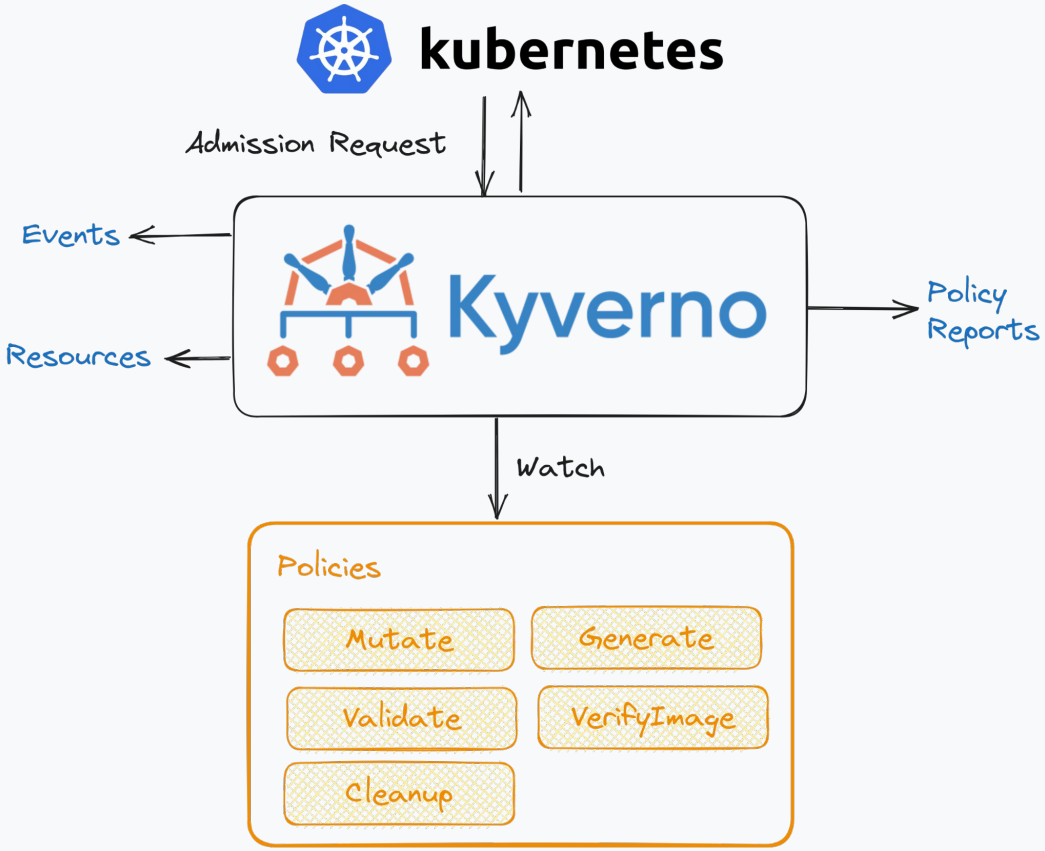
- **Policies** are managed **as Kubernetes resources** (no new language)
- Policies can **validate**, **mutate**, **generate** and **cleanup** Kubernetes resources
- Can also **verify image signatures** and artifacts
- CLI can be used to test policies or validate resources as part of CI/CD

Flinnifonch likes: No new language

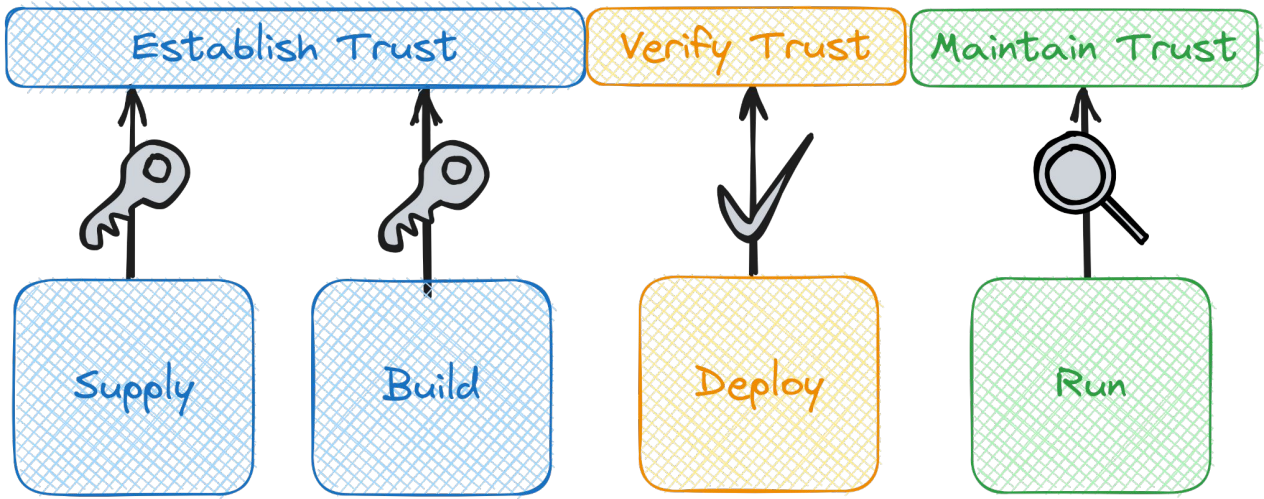
Google Cloud

Kyverno

Google Cloud



Software Supply Chain Security



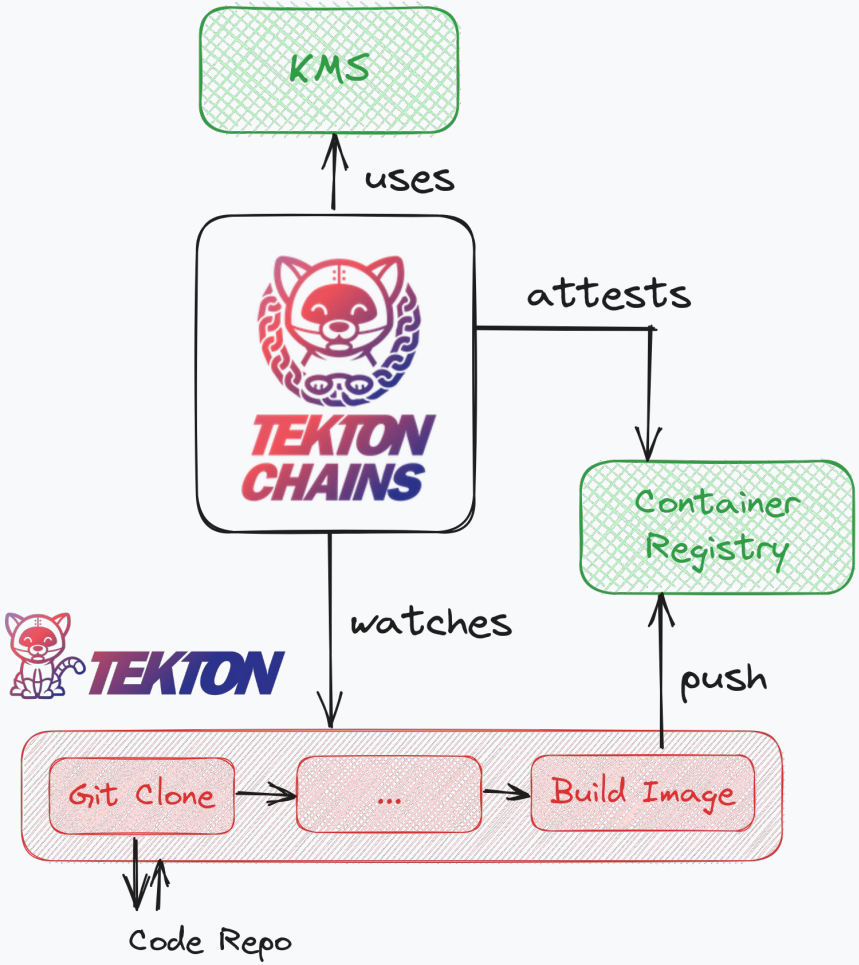
- Create rules and guardrails to follow best-practices and ensure compliance
- Verify that rules are being followed
- Continuously monitor to maintain trust

- **Supply chain security addon** for Tekton
- **Observes all TaskRun executions** in cluster and takes snapshots
- Converts snapshots into **standard payload formats** (e.g. in-toto)
- **Signs** (x509, KMS) and **stores** them (e.g. OCI)

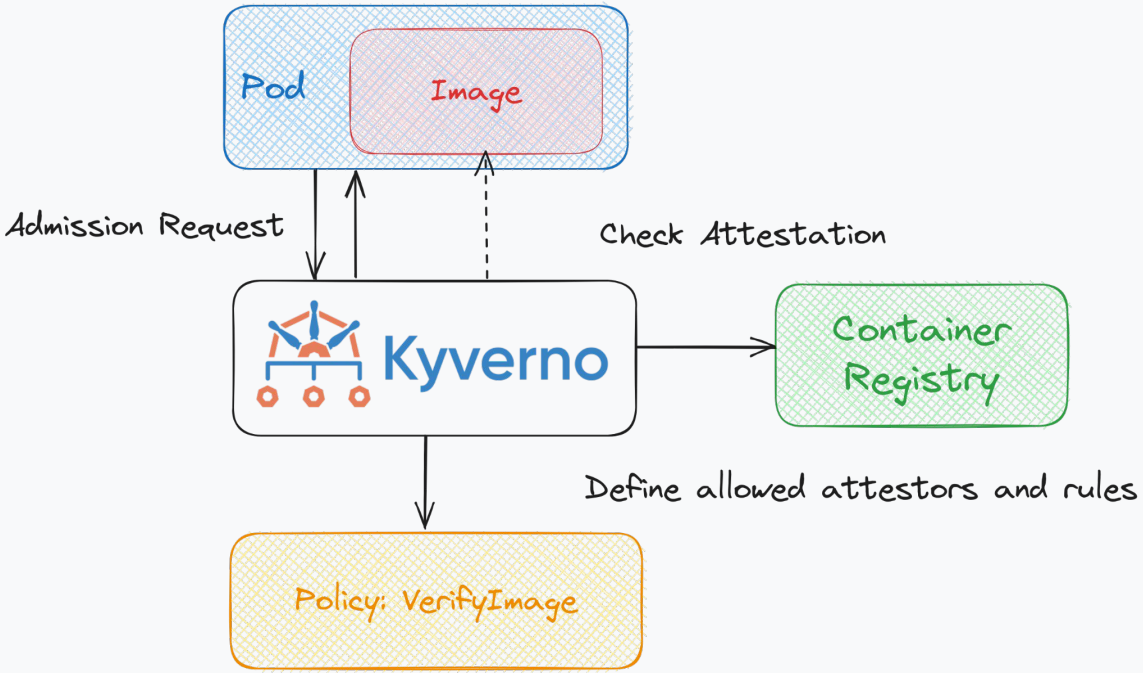
Flinnifonch likes: No changes to DX

Google Cloud

Google Cloud



Kyverno

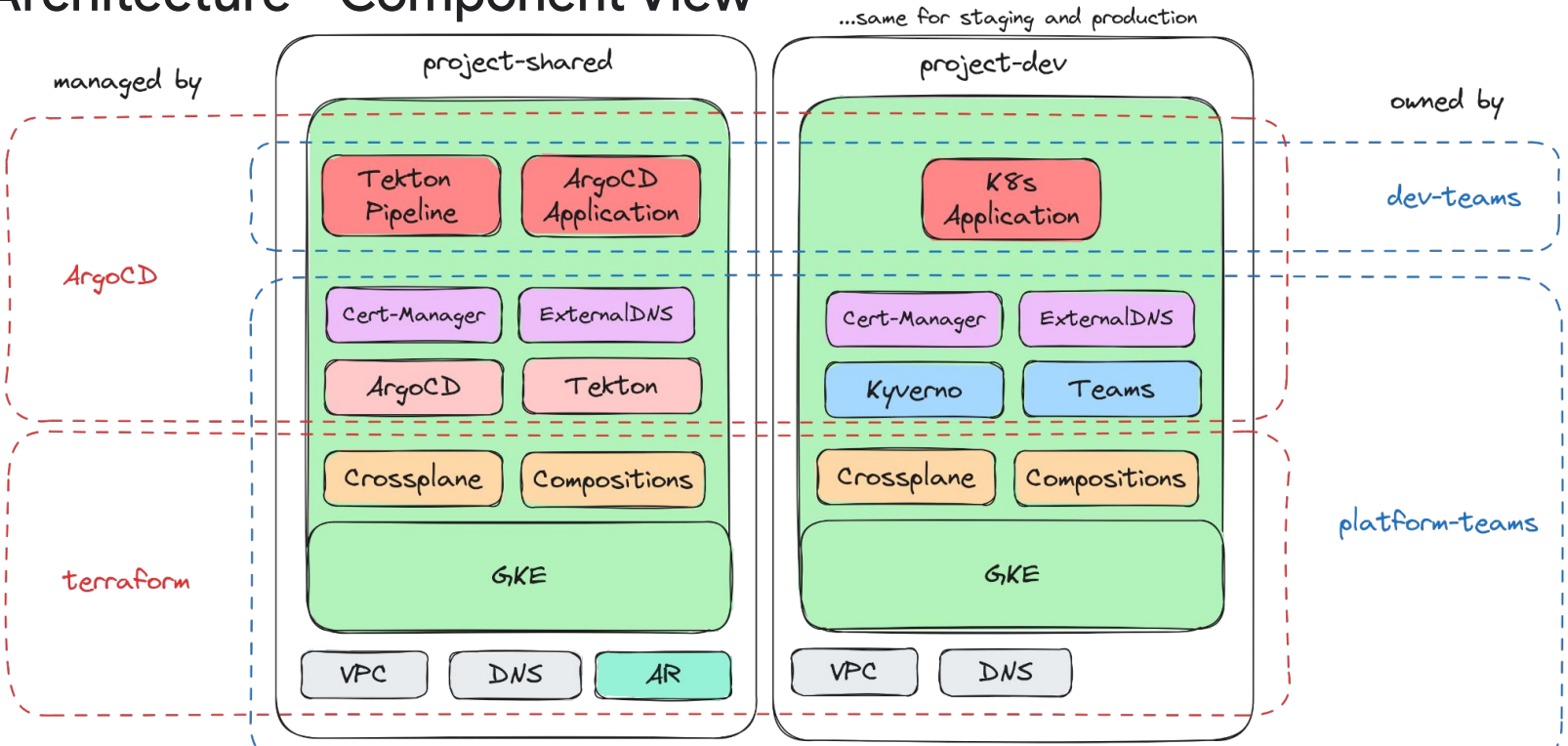


Google Cloud

Putting everything together

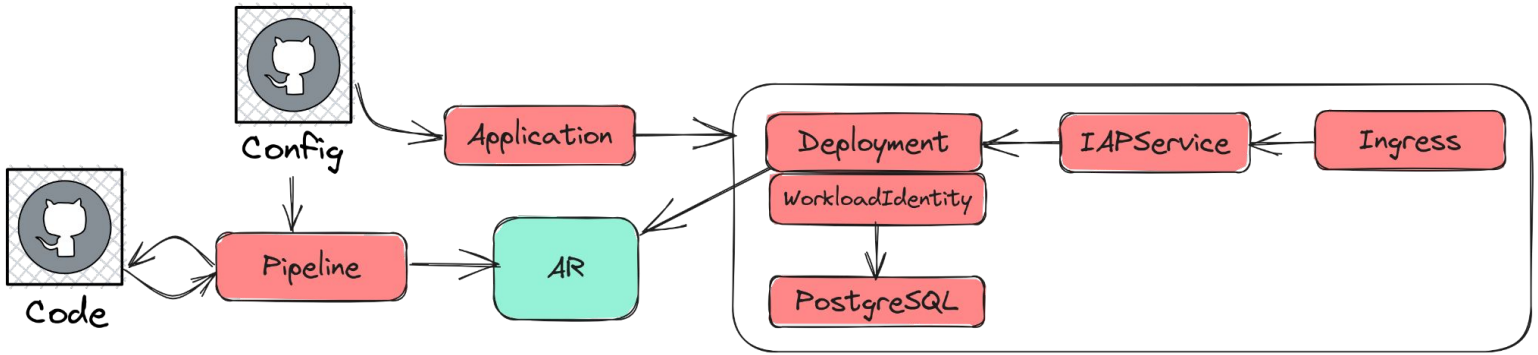
Google Cloud

Architecture - Component View

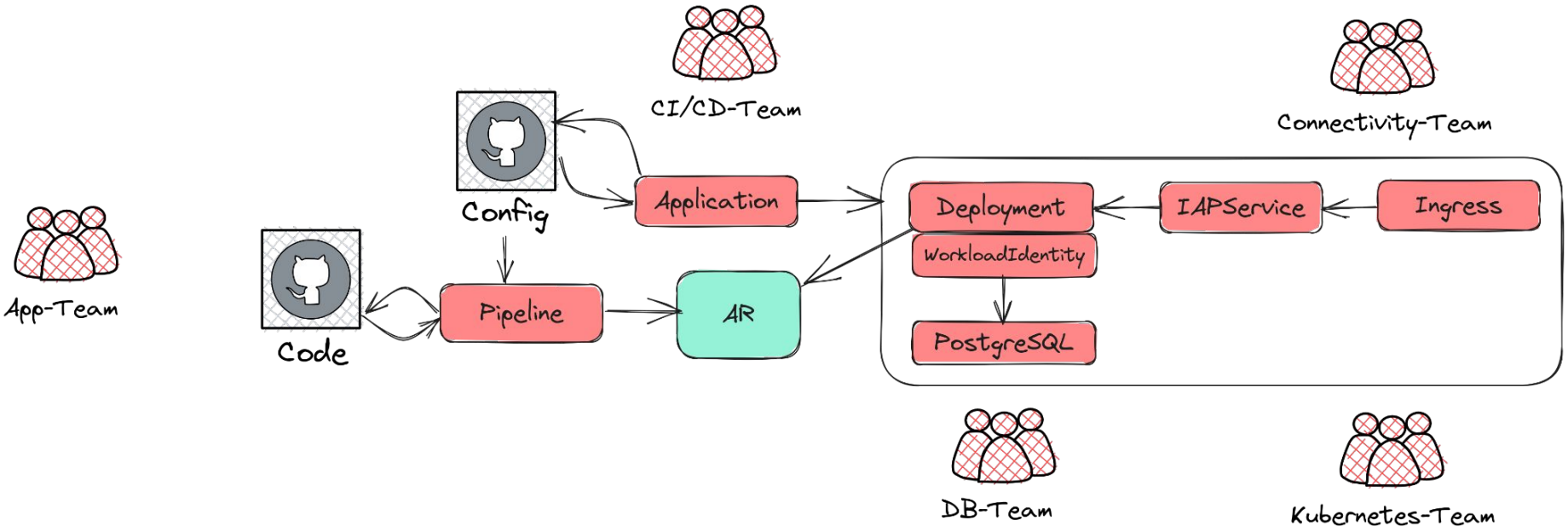


Google Cloud

Architecture - Application View

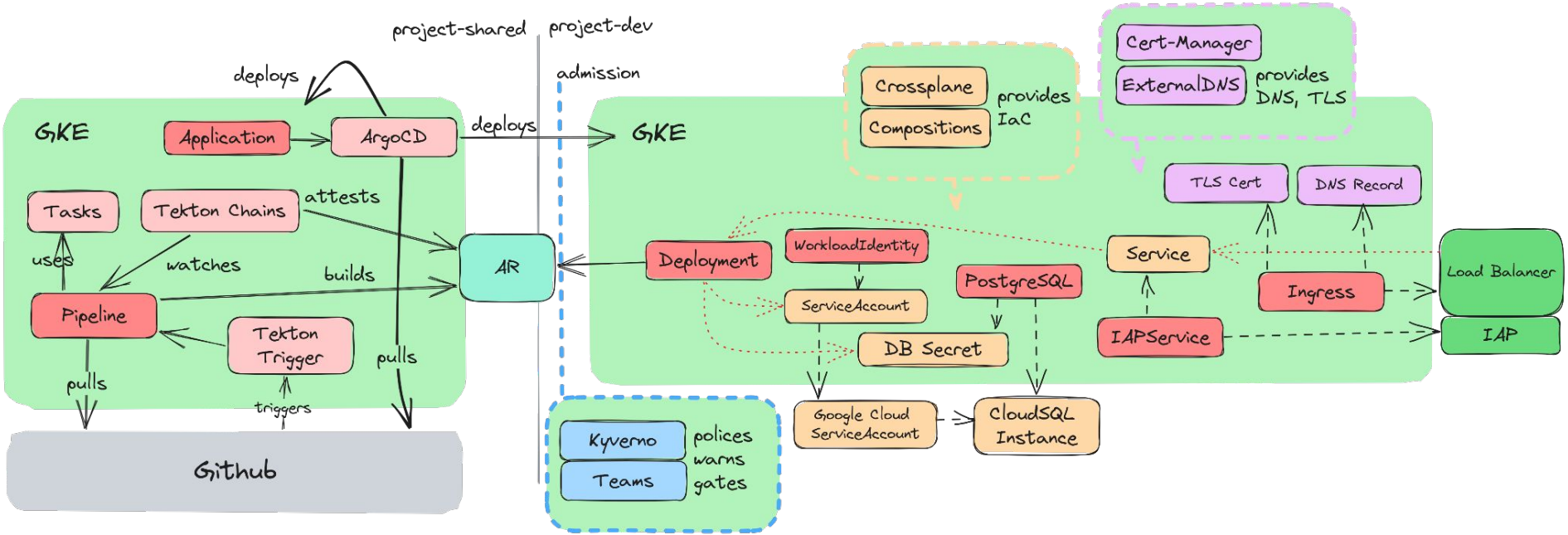


Architecture - Application View



Google Cloud

Architecture - Application View



Google Cloud

Kubernetes as IDP

Google Cloud

- Kubernetes is an **extendable API**
- **Developers are familiar with APIs**
- Kubernetes as IDP ⇒ **developer control plane**
- Only one tools to learn, avoids introducing more paradigms

- Kubernetes **allows decoupling** by introducing separate APIs
- If application teams **lose momentum by coupling** so does a platform ⇒ **avoid a platform monolith**

Conclusion: Design platforms to reduce toil and increase joy (Crystal Hirschorn, Snyk)

Link to repository:

https://github.com/trevex/dogcat

https://github.com/trevex/dogcat-applications

Future of Kubernetes-based platforms: kcp.io?

Thank you!
Questions?

Niklas Voss
Google Cloud

Google Cloud