Crossplane & GitOps in Action:

Unlocking The Kubernetes-Native Control Plane for GCP

devfest
Istanbul 2024

WAIT
LET ME TAKE A SELFIE

Hi 👋,

I'm **Koray Oksay**, a Kubernetes Consultant & Instructor at KUBERMATIC

Find me on X at: **@korayoksay**

CNCF Ambassador | Kubestronaut

Kubernetes Contributor `#k8s-sig-infra`

GitHub Profile: **@koksay**

Hi 👋,

I’m **Batuhan Apaydın**, a Platform Engineer at trendyol.com

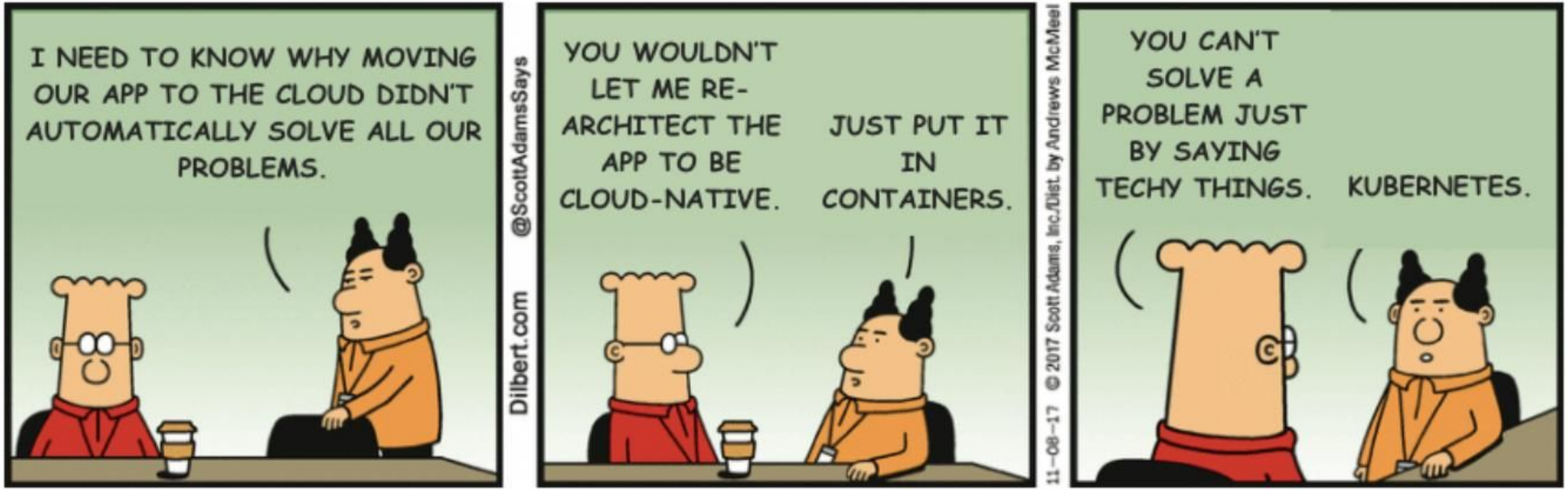
Find me on X at: **@developerguyba**

CNCF Ambassador | Kubestronaut

First ever docker CAPTAIN from Turkey 🇹🇷

GitHub Profile: **@developer-guy**

All your problems are solved. You're welcome!


I NEED TO KNOW WHY MOVING OUR APP TO THE CLOUD DIDN'T AUTOMATICALLY SOLVE ALL OUR PROBLEMS.
@ScottAdamsSays
Dilbert.com
YOU WOULDN'T LET ME RE-ARCHITECT THE APP TO BE CLOUD-NATIVE.
JUST PUT IT IN CONTAINERS.
11-08-17 © 2017 Scott Adams, Inc./Dist. by Andrews McMeel
YOU CAN'T SOLVE A PROBLEM JUST BY SAYING TECHY THINGS.
KUBERNETES.

[Gitpod: We're leaving Kubernetes]

devfest
Istanbul 2024

Containers: revolutionized the way we build, ship, and run applications!

- More formal definition:
  - A container image is a lightweight, stand-alone, executable package of a piece of software that includes everything needed to run it: code, runtime, system tools, system libraries, settings - https://www.docker.com/what-container
- A Linux container is nothing more than a process that runs on Linux
- The Open Container Initiative defines open industry standards around container formats and runtimes
  - Established in June 2015 by Docker and other leaders in the container industry - https://specs.opencontainers.org
  - Docker's original image format has become the **OCI Image Specification**
  - Docker donated its runtime, **runc**, to the OCI to serve as the first implementation of the standard

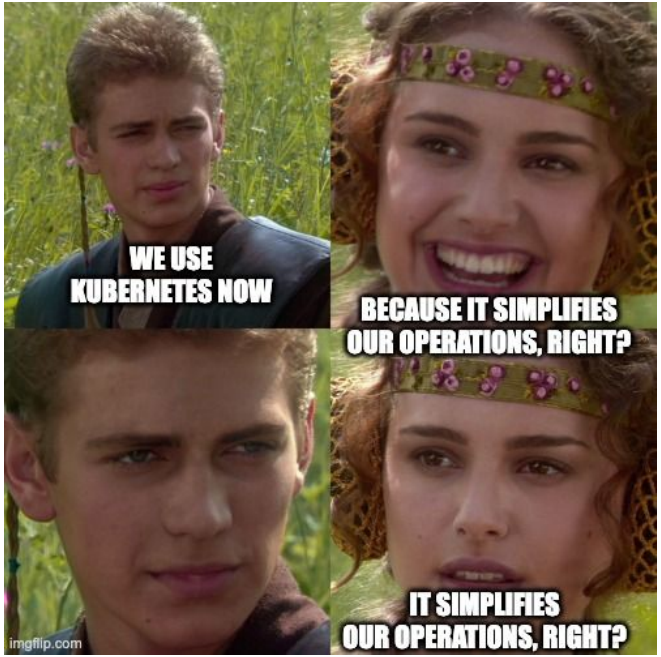
devfest
Istanbul 2024

[A giant container ship the length of four football pitches has become wedged across Egypt's Suez Canal, blocking one of the world's busiest trade routes.]

Kubernetes: a de-facto standard for managing containers at SCALE!

Greek for "Helmsman" or "Pilot"

- Developed by Google engineers Joe Beda, Brendan Burns, and Craig McLuckie in 2014 and open sourced shortly thereafter!
  - Kubernetes now has more than 2,300 contributors, and has been widely adopted by companies large and small, including half of the Fortune 100.
  - Kubernetes was the first graduated project for the CNCF!
- Often described as the "operating system for the cloud"
- Inspired and informed by Google's experiences and internal systems - Kubernetes was sort of a spiritual successor to Borg," describes Kubernetes co-creator Joe Beda in a 2021 interview with Increment.
- Honeypot put together a cool documentary series (part 1 and part 2).

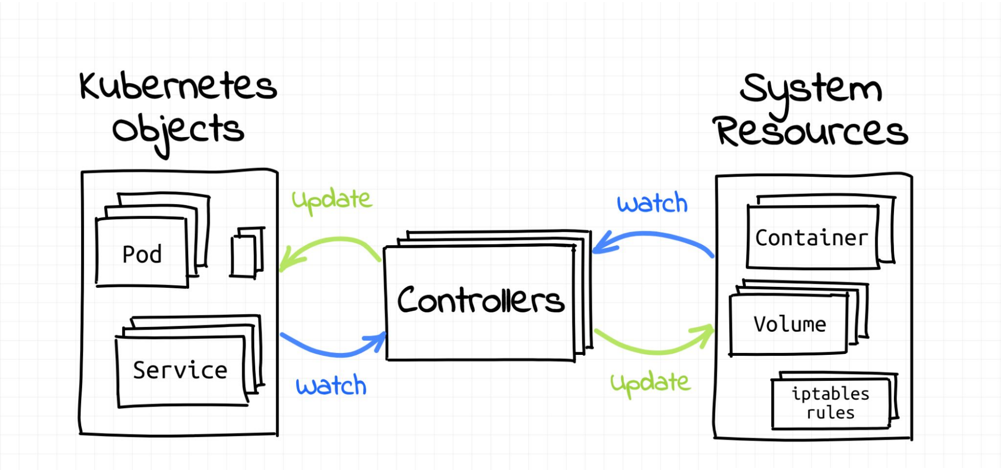

WE USE KUBERNETES NOW
BECAUSE IT SIMPLIFIES OUR OPERATIONS, RIGHT?
IT SIMPLIFIES OUR OPERATIONS, RIGHT?
imgflip.com

devfest
Istanbul 2024

Kubernetes Operators: extend Kubernetes API by adding custom types!

Some assembly may be required. Ideas included!

Kubernetes is a platform for building platforms!



devfest
Istanbul 2024

Current state of the Art Infrastructure As Code

HashiCorp Terraform

Pulumi

ANSIBLE

OpenTofu

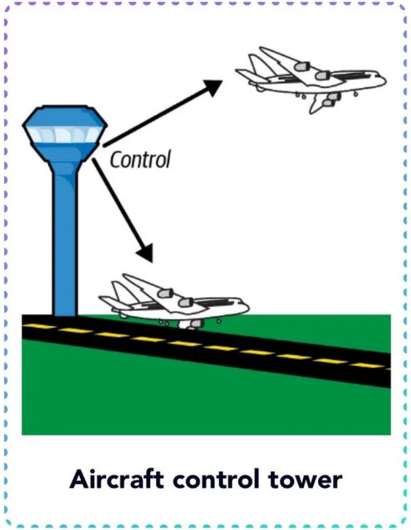
The Result:

Use the same API across bare metal and every cloud provider, not only for container orchestration, but also for any object you need to manage!

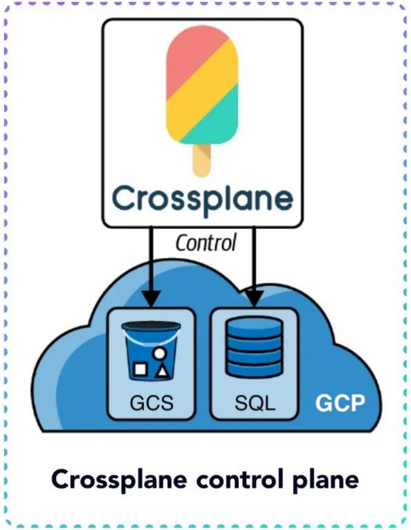
devfest
Istanbul 2024

Crossplane: the cloud native control plane framework!

“It’s a bridge between cloud resources and Kubernetes”



Aircraft control tower



Crossplane control plane

What’s a Control Plane?

Control planes create and manage the lifecycle of resources.
Control planes constantly check that the intended resources exist, report when the intended state doesn’t match reality and act to make things right.

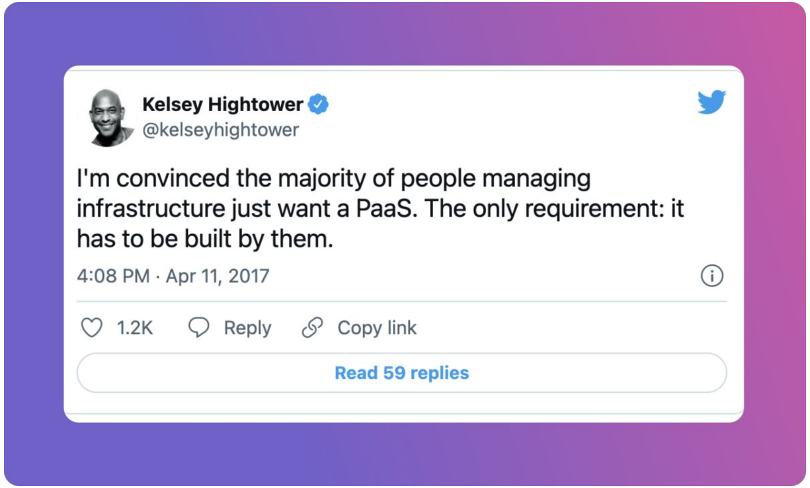
https://crossplane.io

devfest
Istanbul 2024

Internal Developer Platform (IDP): self-service platform built by engineers for engineers!

"An Internal Developer Platform (IDP) is built by a platform team to build golden paths and enable developer self-service.

An IDP consists of many different techs and tools, glued together in a way that lowers cognitive load on developers without abstracting away context and underlying technologies."



Kelsey Hightower @kelseyhightower

I'm convinced the majority of people managing infrastructure just want a PaaS. The only requirement: it has to be built by them.

4:08 PM · Apr 11, 2017

Backstage

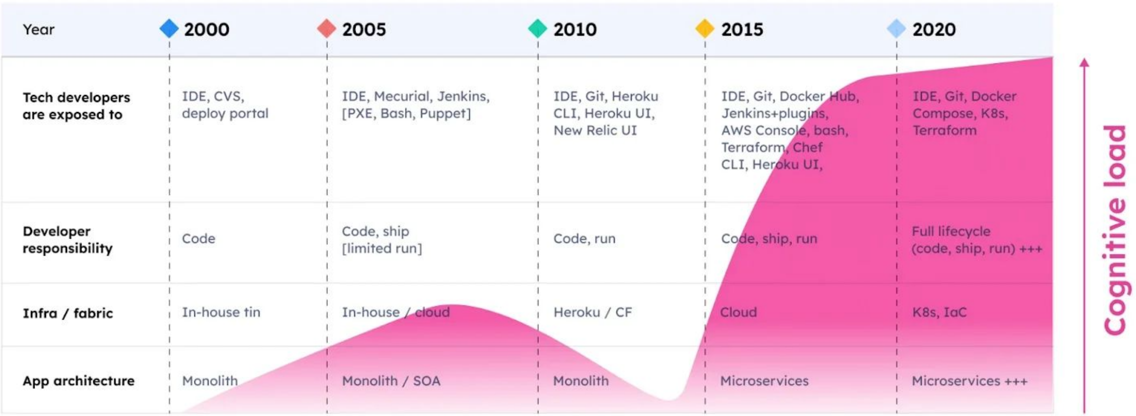
An open source framework for building developer portals

https://internaldeveloperplatform.org

devfest Istanbul 2024

Platform Engineering: empower your DevOps teams to be more productive!

Platform engineers provide an integrated product most often referred to as an “Internal Developer Platform” covering the operational necessities of the entire lifecycle of an application.



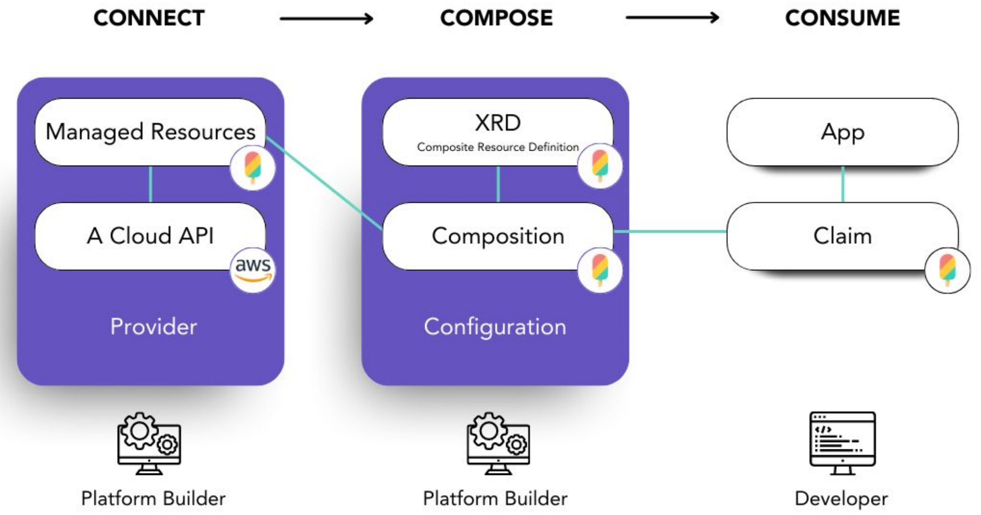
Year	2000	2005	2010	2015	2020
Tech developers are exposed to	IDE, CVS, deploy portal	IDE, Mecurial, Jenkins, [PXE, Bash, Puppet]	IDE, Git, Heroku CLI, Heroku UI, New Relic UI	IDE, Git, Docker Hub, Jenkins+plugins, AWS Console, bash, Terraform, Chef CLI, Heroku UI,	IDE, Git, Docker Compose, K8s, Terraform
Developer responsibility	Code	Code, ship [limited run]	Code, run	Code, ship, run	Full lifecycle (code, ship, run) +++
Infra / fabric	In-house tin	In-house / cloud	Heroku / CF	Cloud	K8s, IaC
App architecture	Monolith	Monolith / SOA	Monolith	Microservices	Microservices +++

From Kubernetes to PaaS to … err, what’s next? • Daniel Bryant • PlatformCon 2022

devfest
Istanbul 2024

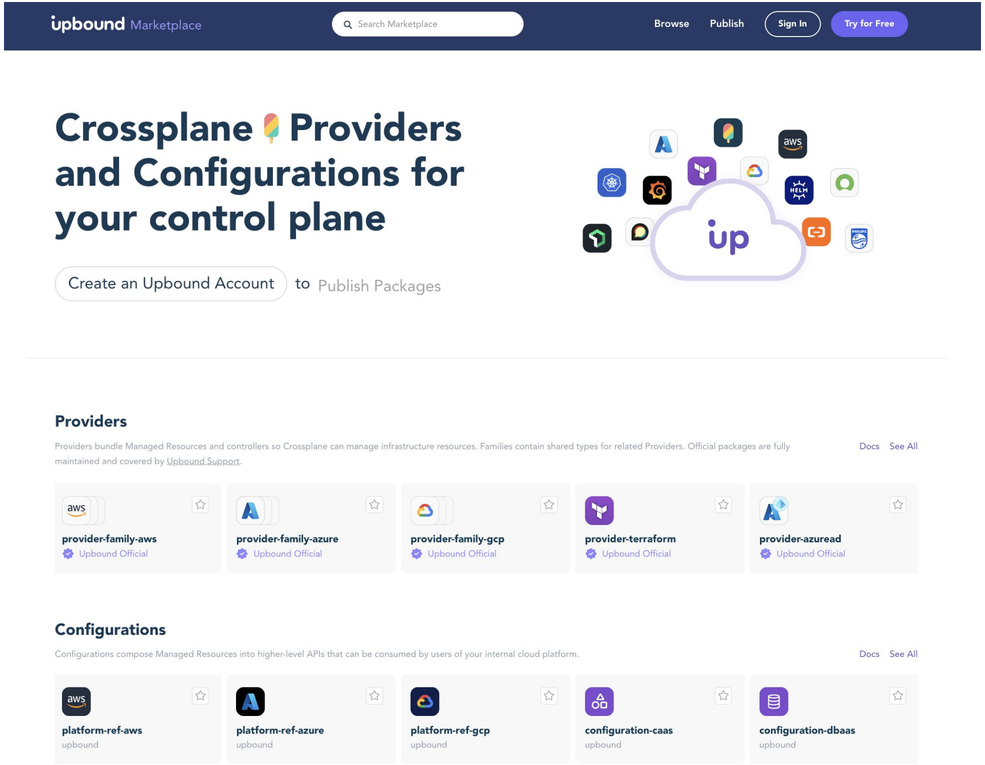
Crossplane: in a nutshell!

- Provider creates new Kubernetes Custom Resource Definitions for an external service.
- Managed Resource is a Provider resource created and managed by Crossplane inside the Kubernetes cluster.
- Composition is a template for creating multiple *managed resources* at once.
- Composite Resources uses a *Composition* template to create multiple *managed resources* as a single Kubernetes object.
- CompositeResourceDefinitions defines the API schema for *Composite Resources* and *Claims*



https://blog.upbound.io/composing-a-platform-by-patching-crossplane-resources

Crossplane: Providers, Providers, Providers!



Crossplane Providers and Configurations for your control plane

Create an Upbound Account to Publish Packages

Providers

Providers bundle Managed Resources and controllers so Crossplane can manage infrastructure resources. Families contain shared types for related Providers. Official packages are fully maintained and covered by Upbound Support.

Docs See All

- provider-family-aws — Upbound Official
- provider-family-azure — Upbound Official
- provider-family-gcp — Upbound Official
- provider-terraform — Upbound Official
- provider-azuread — Upbound Official

Configurations

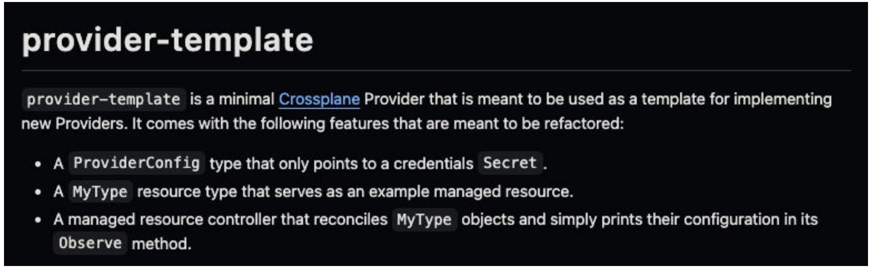
Configurations compose Managed Resources into higher-level APIs that can be consumed by users of your internal cloud platform.

Docs See All

- platform-ref-aws — upbound
- platform-ref-azure — upbound
- platform-ref-gcp — upbound
- configuration-caas — upbound
- configuration-dbaas — upbound

https://marketplace.upbound.io

Crossplane: writing YOUR OWN Provider is easy!

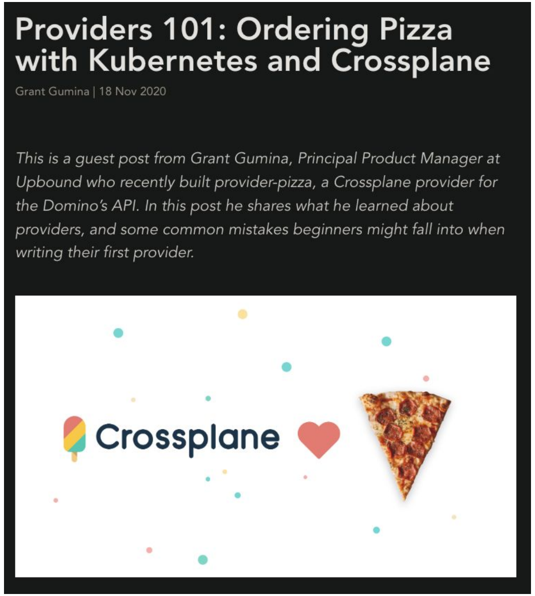


provider-template

`provider-template` is a minimal Crossplane Provider that is meant to be used as a template for implementing new Providers. It comes with the following features that are meant to be refactored:

- A `ProviderConfig` type that only points to a credentials `Secret`.
- A `MyType` resource type that serves as an example managed resource.
- A managed resource controller that reconciles `MyType` objects and simply prints their configuration in its `Observe` method.

https://github.com/crossplane/provider-template



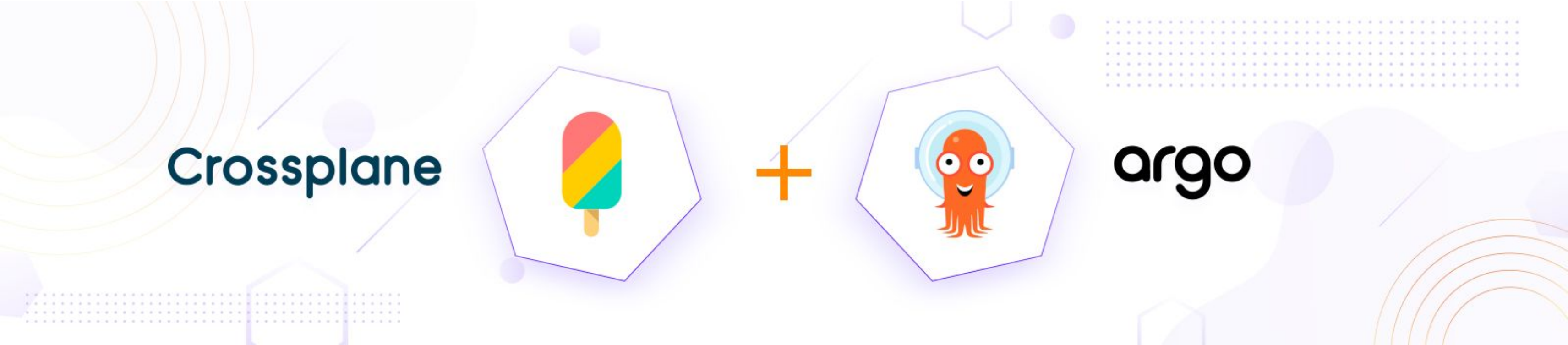
Providers 101: Ordering Pizza with Kubernetes and Crossplane

Grant Gumina | 18 Nov 2020

This is a guest post from Grant Gumina, Principal Product Manager at Upbound who recently built provider-pizza, a Crossplane provider for the Domino's API. In this post he shares what he learned about providers, and some common mistakes beginners might fall into when writing their first provider.

https://blog.crossplane.io/providers-101-ordering-pizza-with-kubernetes-and-crossplane/

GitOps: missing ingredient for managing Infrastructure by automation!



GitOps Principles

OpenGitOps

Declarative

A system managed by GitOps must have its desired state expressed declaratively.

Versioned and Immutable

Desired state is stored in a way that enforces immutability versioning and retains a complete version history.

Pulled Automatically

Software agents automatically pull the desired state declarations from the source.

Continuously Reconciled

Software agents continuously observe actual system state and attempt to apply the desired state.

https://opengitops.dev

devfest
Istanbul 2024

Q&A / Feedback

THANKS

@developerguyba

@korayoksay

devfest
Istanbul 2024