

# Building a Database as a Service on Kubernetes

By Abhi Vaidyanatha and Lucy Burns

# Who are we anyway?



**Abhi Vaidyanatha**

He/Him/His

Software Engineer

PlanetScale



**The PlanetScale Team**



**Lucy Burns**

She/Her/Hers

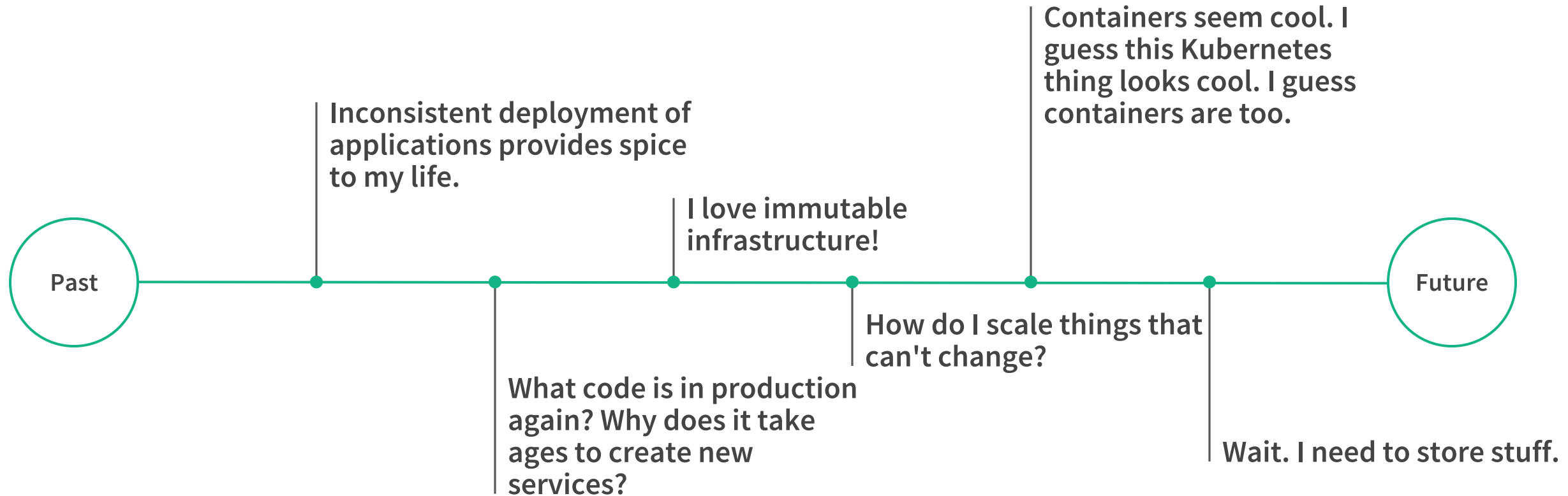
Product Manager

PlanetScale

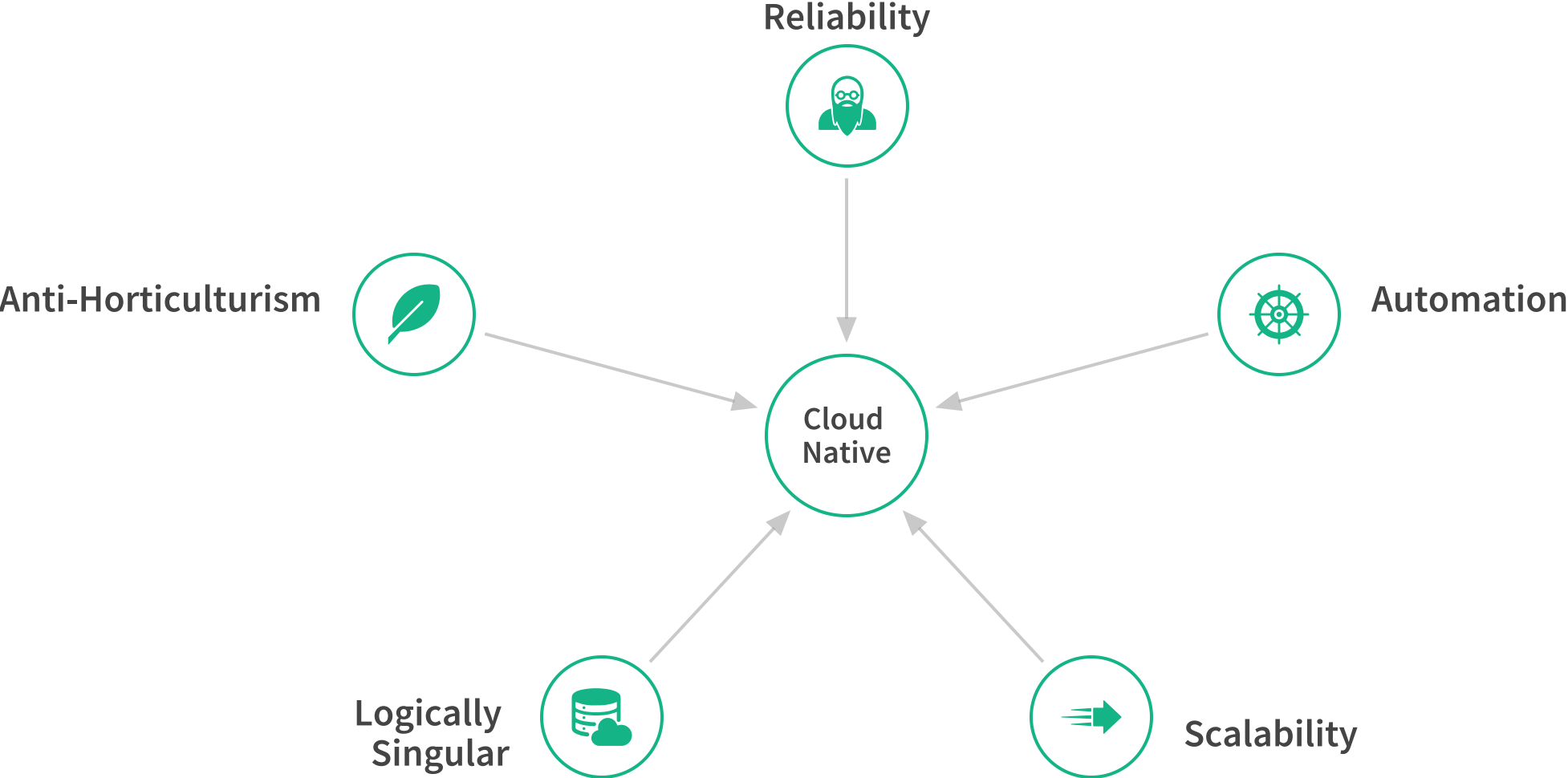
# Agenda

- 1 Defining the problem.
- 2 Choosing the technology.
- 3 Fitting it together.

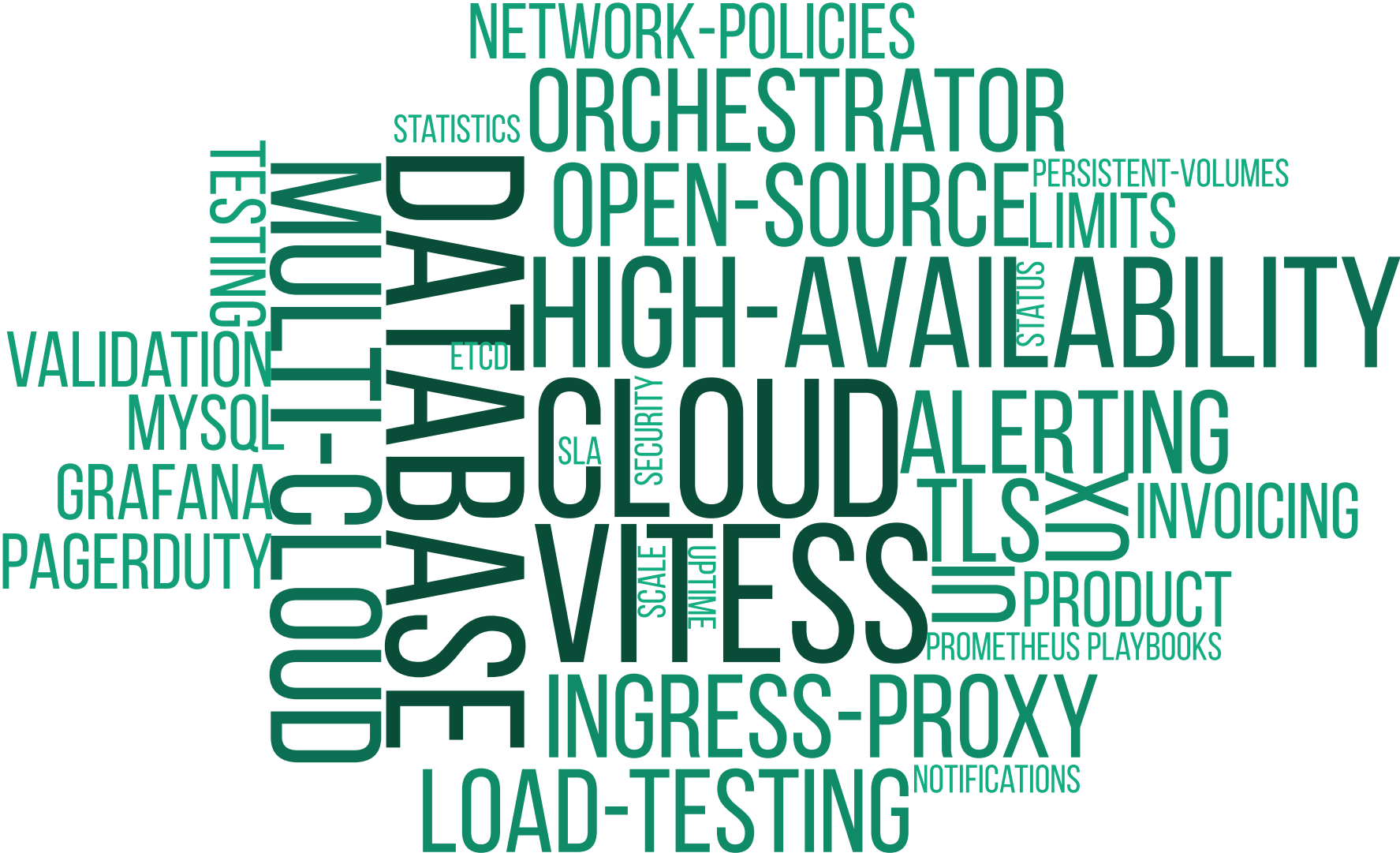
# Where are we? The Kubernetes Journey



# What do we want from a database?



# What is the MVP of a Database?



# Why Open Source?

- **Pros**

- Ideal for container ecosystem
- Motivated contributors, bleeding edge technology
- Participation in the community
- Why reinvent the wheel?

- **Cons**

- Feature Prioritization
- Longevity
- Support

Cool. Now what?

# What's our platform?



**kubernetes**

- Enables flexibility of deployment
- Scheduling of Jobs across heterogeneous compute clusters
- Services for network connectivity



# The Bread and Butter.

- ViteSS is a great database management system.
- But... ViteSS is a challenge to configure
- ViteSS is designed to manage one large distributed system.
- Network Policies and more!
- What's the best way to bring up ViteSS?



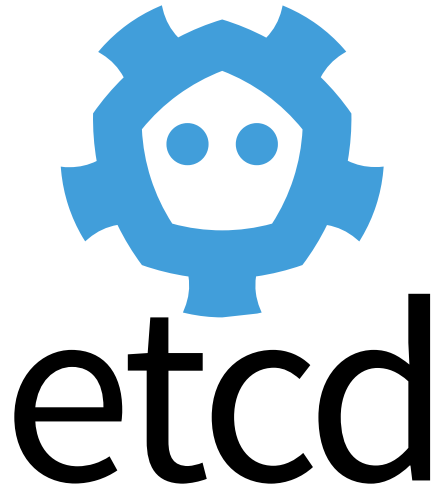
# Kubernetes Operators



**OPERATOR  
SDK**

- 1 Why are operators cool?
- 2 They exist for etcd and Prometheus... but how about our database?
- 3 Let's make one!
- 4 Yeah we did that all wrong; let's make another one.

# How do we manage configuration?



- Etcd stores configuration for our distributed database.
- Can't we just use the operator?
- Storing state in a stateless object doesn't really work that well.
- How do we recover from loss of quorum?

# If something is broken, how will we know?

- Installing Prometheus is mostly becoming a plumbing engineer.
- Most software knows about it, so metrics endpoints are very common.
- Why is transparent configuration cool?
- Don't use cospypasta when CoreOS has taken care of all the hard work for you.



Prometheus

# We want pretty colors!



- You're not a complete product without offering some sort of insights package.
- Grafana can be auto-provisioned, so we baked it into our Vitess operator.
- How do we manage ingress to our dashboards?

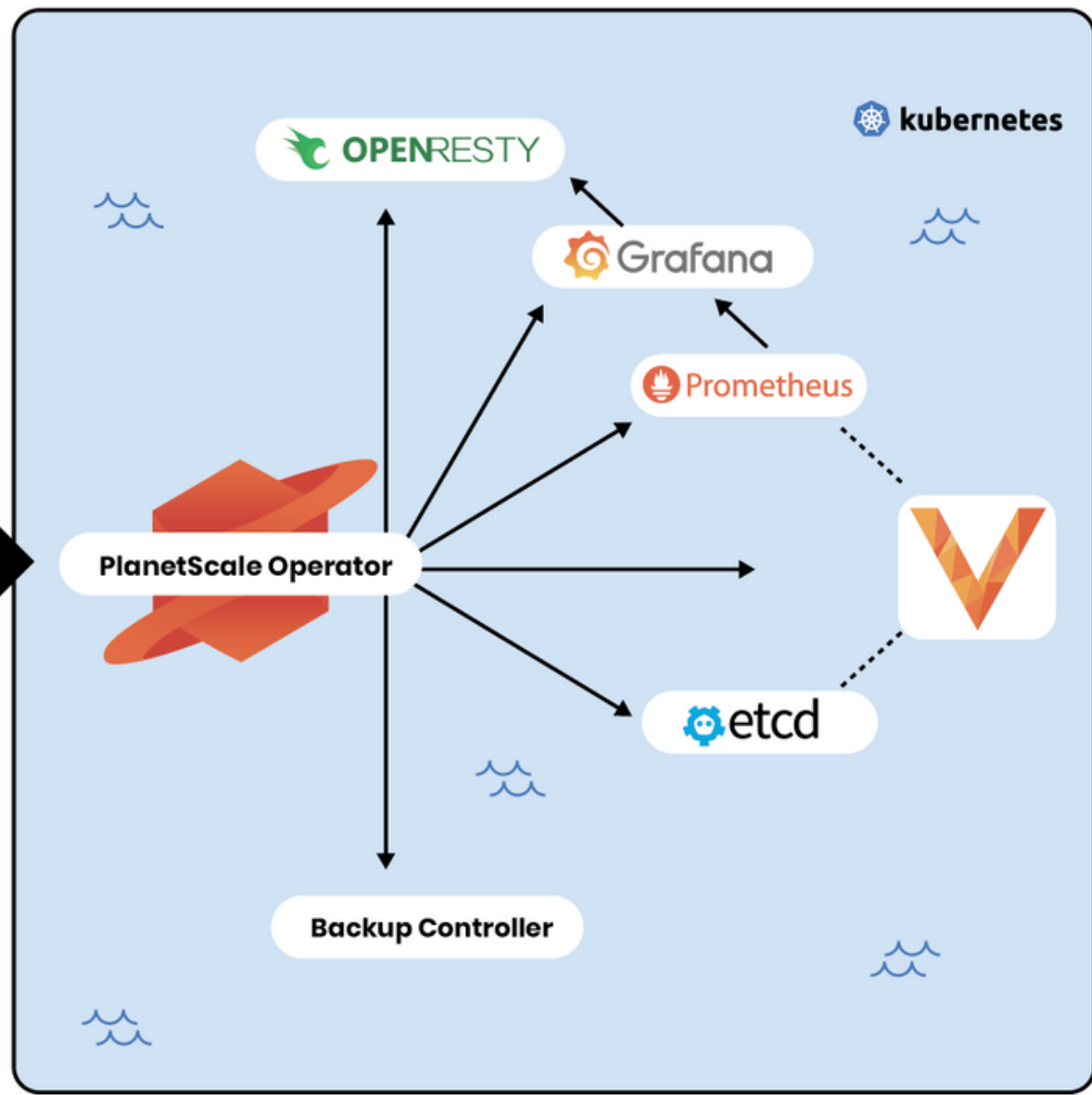
# How do we handle internal connectivity?

- **We have a lot of requirements here.**
  - Proxy our services [Main function]
  - Lightweight
  - Concurrent
  - Integrate with the Kubernetes API
- **OpenResty retains all NGINX functionality.**





K8s API request



# Hey, we built something!

- **PlanetScale CNDb Demo**

[www.console.planetscale.com](http://www.console.planetscale.com)

- **Wait. Was this actually that hard?**

Well. Yeah. But not for the reasons we thought it would be.



# What's next?

- ① Building trust in cloud infrastructure
- ② BYOK - Bring your own Kubernetes
- ③ Multi-Cloud Clusters
- ④ Predictive scaling

Come talk to us at Booth SE39!

# Learn more about Vitess at KubeCon

- ① **Tuesday 3:20 - 3:55pm** Vitess: Stateless Storage in the Cloud - Sugu Sougoumarane, PlanetScale
- ② **Wednesday 2:25 pm - 3:00pm** Geo-partitioning with Vitess - Deepthi Sigireddi, PlanetScale & Jitendra Vaidya, PlanetScale
- ③ **Thursday 2:25 - 3:00pm** Gone in 60 Minutes: Migrating 20 TB from AKS to GKE in an Hour with Vitess - Derek Perkins, Nozzle

Come talk to us at Booth SE39!