

**Lunar Way's journey towards Cloud
Native Utopia**
Speaker Kasper Nissen





**Click 'Rate Session'
to rate session
and ask questions.**





Please

**Remember to
rate this session**

Thank you!



Did you **remember**
to rate the previous
session ?



goto;
copenhagen



 Follow us @gotocph

***lunar
way***[®]

@phennex

@lunarway

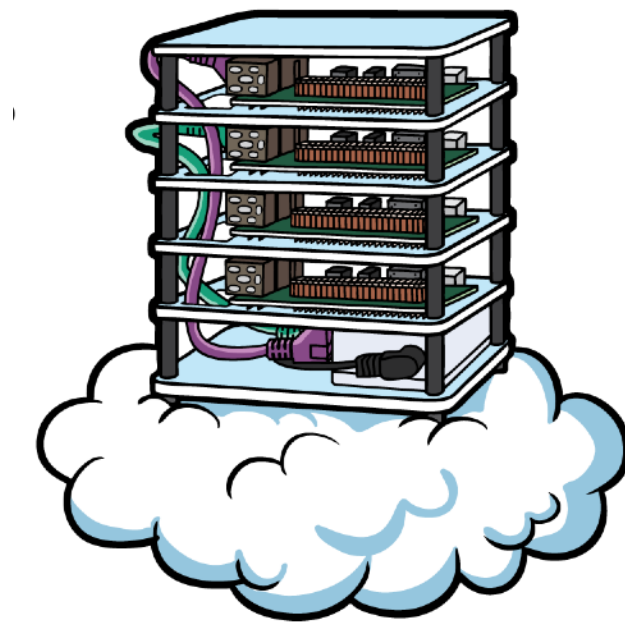


Kasper Nissen

DevOps & Cloud Architecture @ Lunar Way



Cloud Native Aarhus
Cloud Native DK



kubecloud.io





foto: Lars Kruse, Aarhus Universitet

Opret dig og få

Vision

We're living in the era of mobile/digital only
– we believe banking and commerce should to.

Our vision is to ***rethink the interaction with money and defining a completely new category*** - by introducing a new money app.

It's the complex coordination between banking services and commerce use:

- How I save money.
- How I get money.
- How I spend money.



Numbers

12K+

customers

1,3M+

transactions

30+

microservices

35+

employees

700M+ DKK

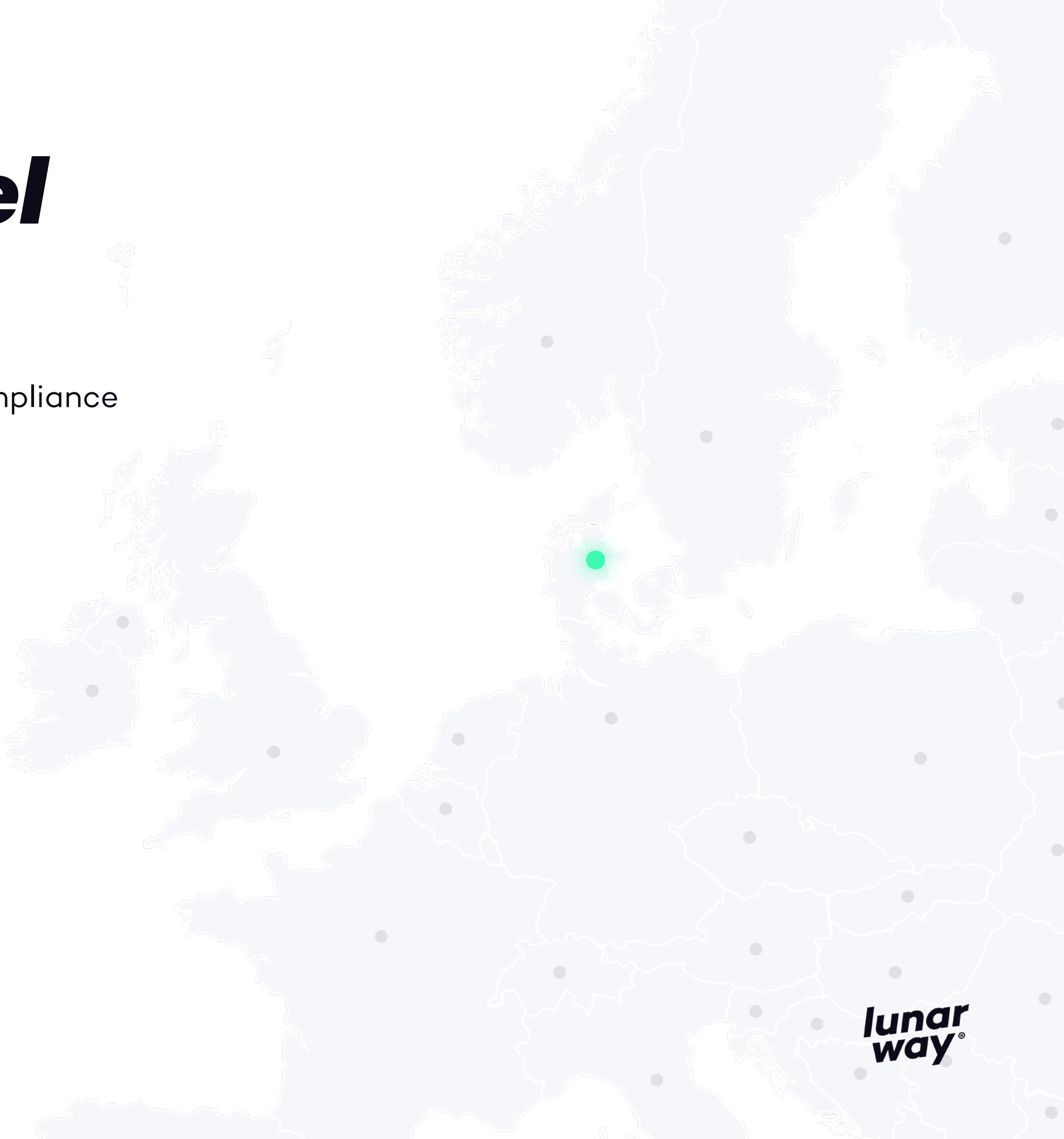
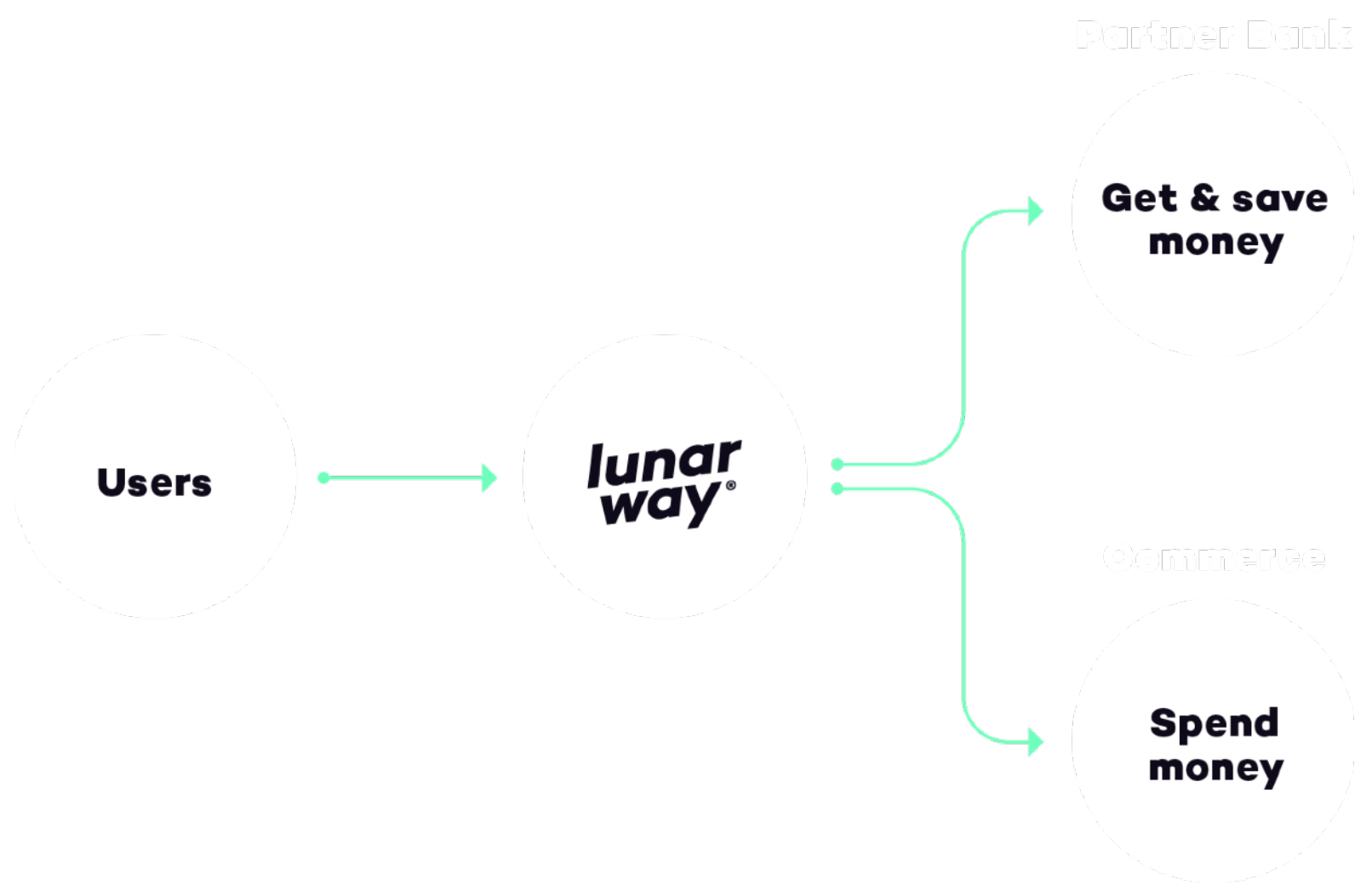
through our system

3

kubernetes clusters

The partner model

- All money is in the partner bank
- Leverage the partner banks' infrastructure and compliance

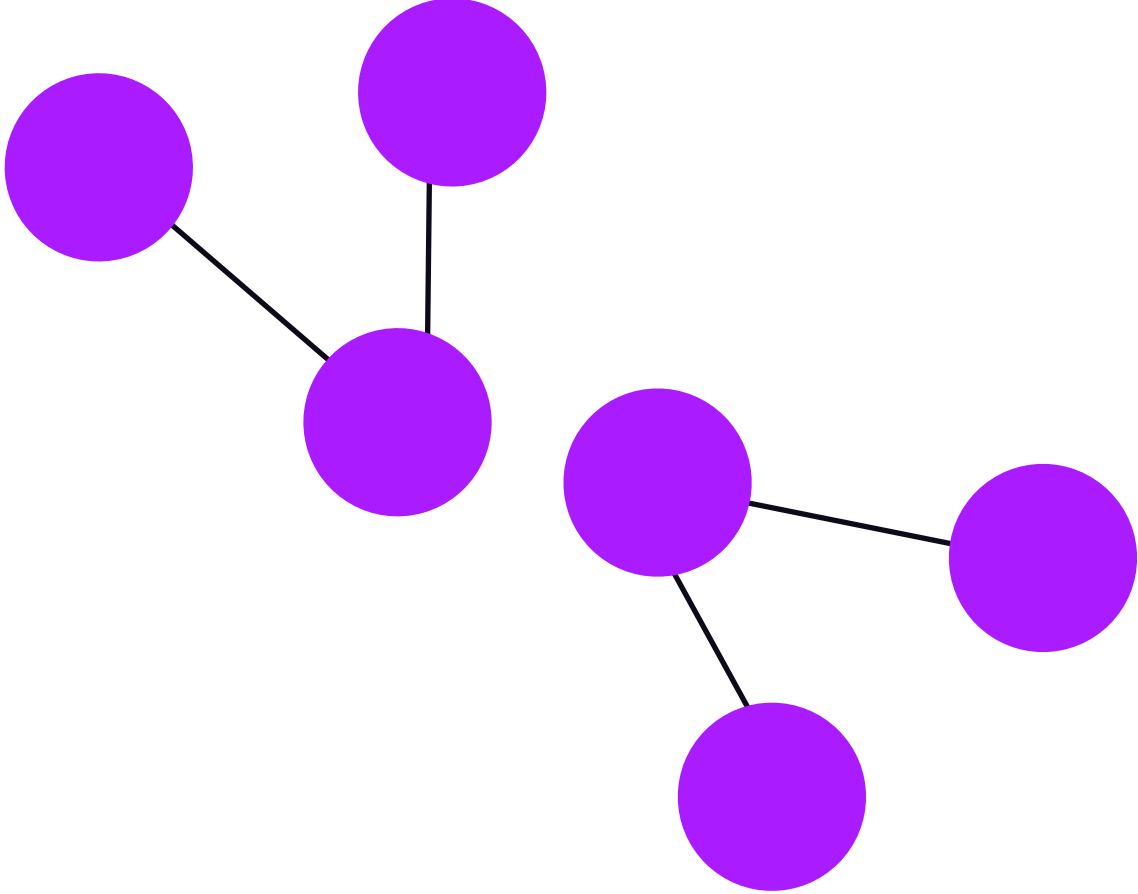


The image features a solid purple background with a network of thin, light purple lines connecting several large, semi-transparent purple circles of varying sizes. The circles are arranged in a way that suggests a global or interconnected network. In the top left corner, the text "lunar way" is written in a white, lowercase, sans-serif font, with a registered trademark symbol (®) to the right of "way".

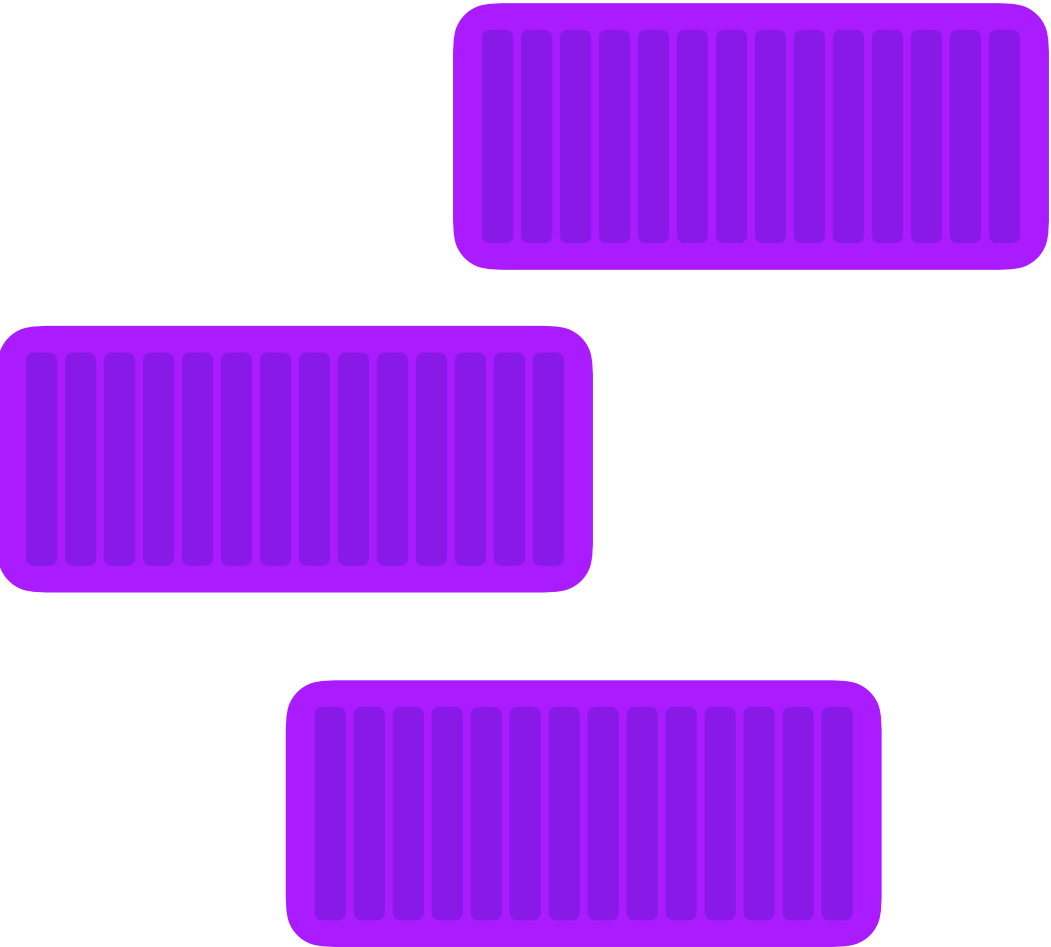
***lunar
way***[®]

***Lunar Way's journey towards
Cloud Native Utopia***

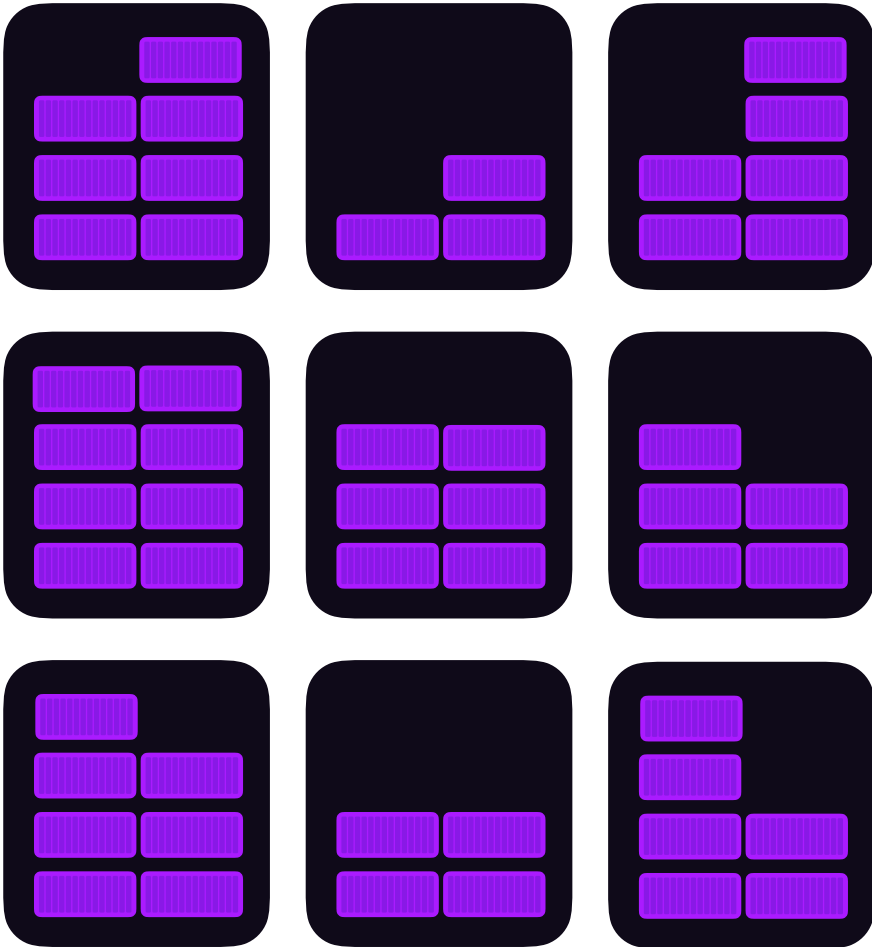
Cloud Native?



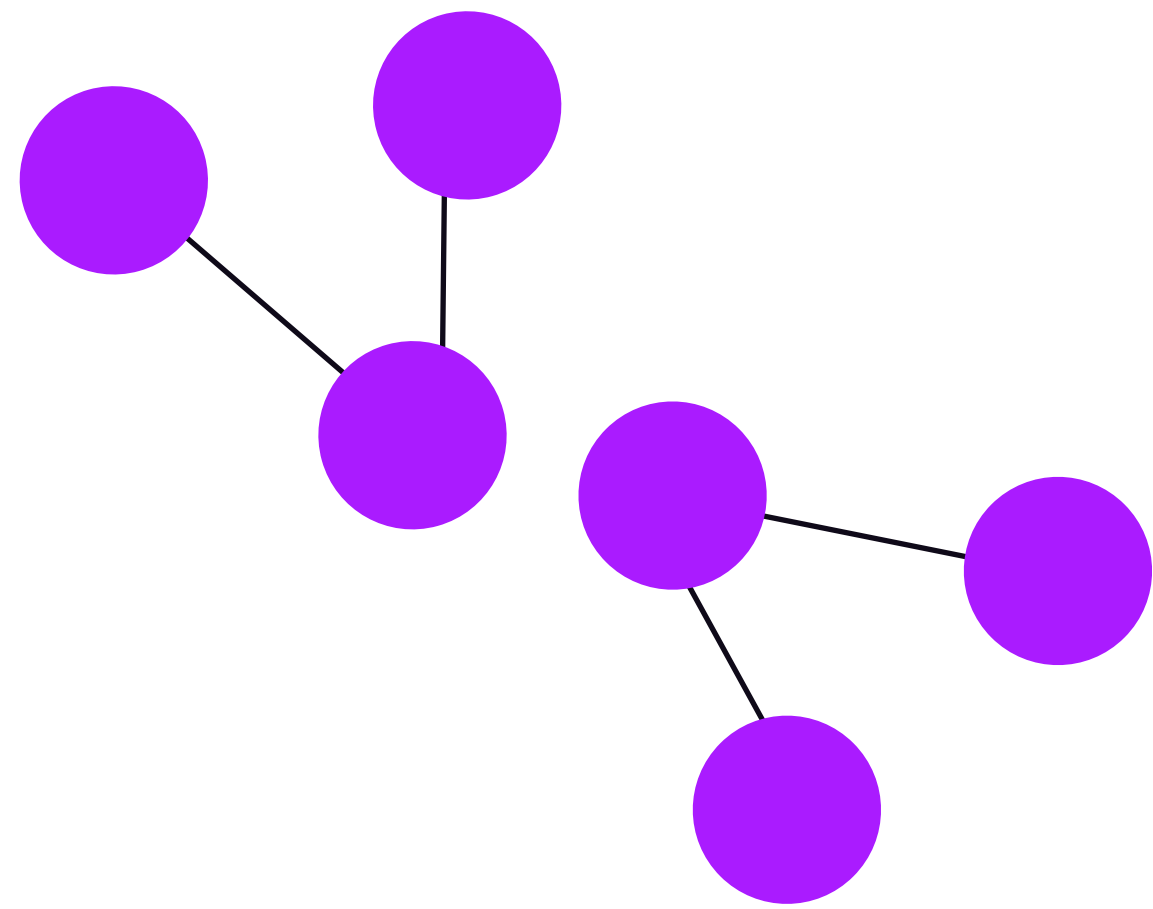
Microservice oriented



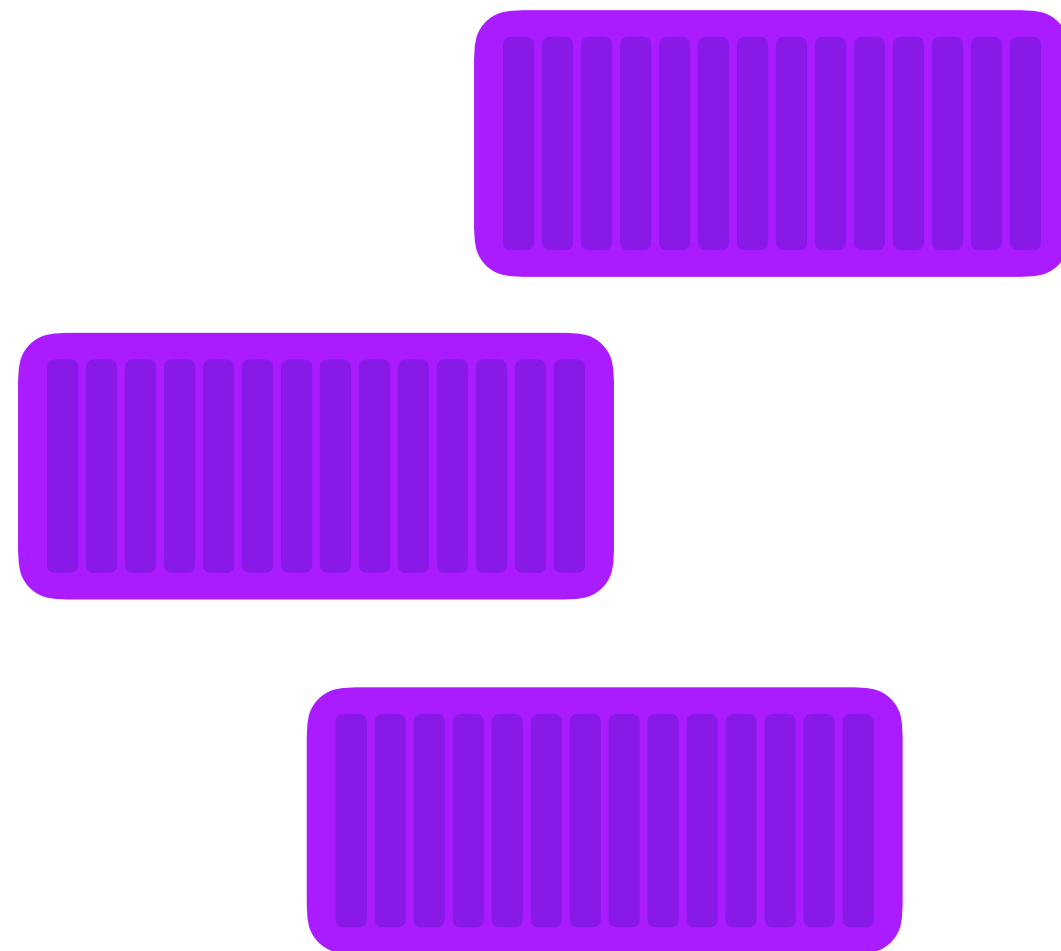
Container packaged



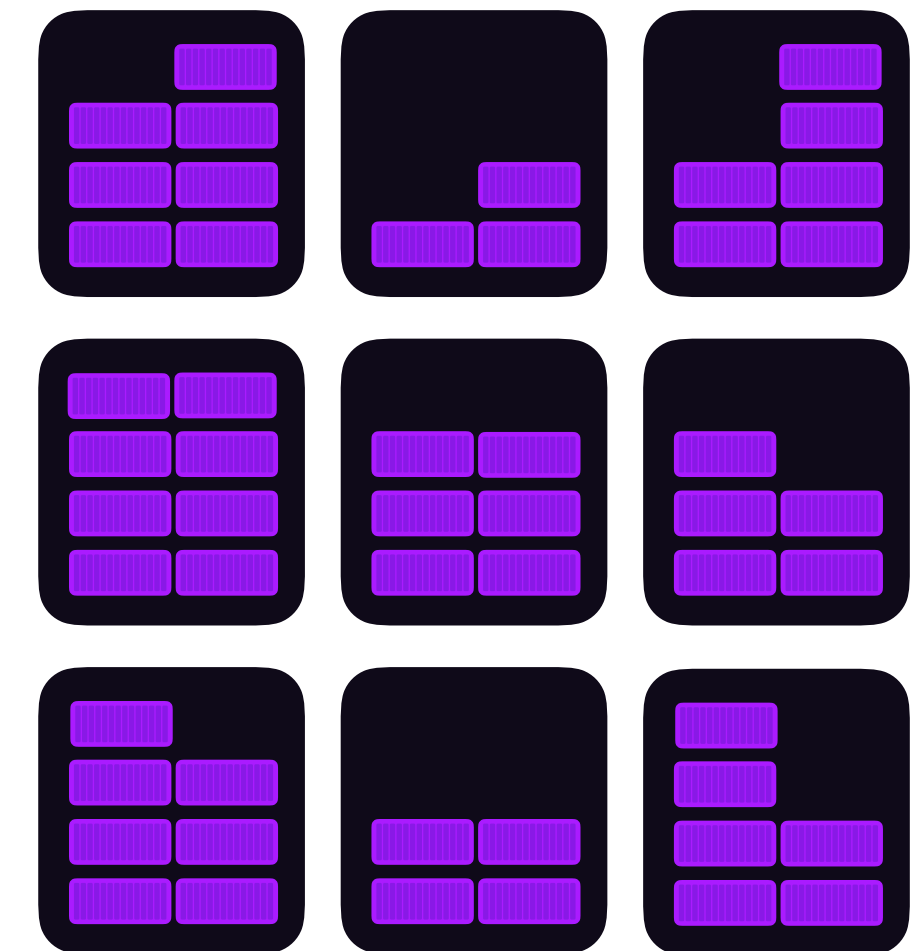
Dynamically scheduled



Microservice oriented



Container packaged



Dynamically scheduled

“

Cloud Native is structuring teams, culture and technology to utilize automation and architectures to manage complexity and unlock velocity.

Joe Beda, CTO at Heptio

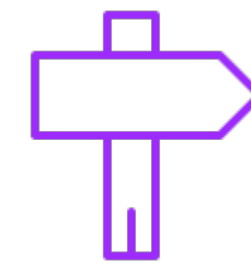
Why go there?



Reduce time-to-market



Allow for continuous innovation



Ability to pivot



Our journey so far

Ruby on Rails

Strangling Rails

Kubernetes

2015

2016

2017

2018

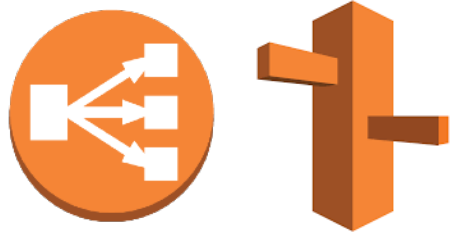
Integrations
First services

Microservices with
monolithic deployment

The end of our
“monolith”

So, what do we have running?

Highlevel overview



Node

RabbitMQ

realm

VPN

Node

NGINX

node

RAILS

node

Node

NGINX

node

RAILS

node

Node

NGINX

node

RAILS

node



***How to navigate the
Cloud Native ecosystem?***



Cloud Native Landscape
v0.9.7

github.com/cncf/landscape

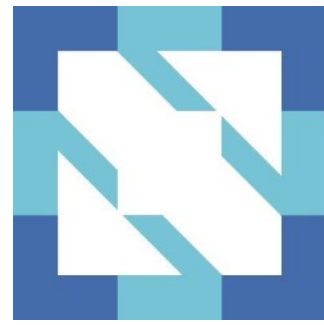
This landscape is intended as a map through the previously uncharted terrain of cloud native technologies. There are many routes to deploying a cloud native application, with CNCF Projects representing a particularly well-traveled path.

CLOUD NATIVE COMPUTING FOUNDATION

Redpoint | Amplify PARTNERS

Greyed logos are not open source





CLOUD NATIVE COMPUTING FOUNDATION

Currently Hosted Projects



Kubernetes

Orchestration



Prometheus

Monitoring



OpenTracing

Distributed Tracing API



Fluentd

Logging



linkerd

Service Mesh



gRPC

Remote Procedure Call



CoreDNS

Service Discovery



containerd

Container Runtime



rkt

Container Runtime



CNI

Networking API



Envoy

Service Mesh



Jaeger

Distributed Tracing



Dynamic Scheduling with Kubernetes



Why do we need an orchestration tool?

- **Scheduling** - where are our containers going to run?
- **Availability** - scale to a desired state
- **Resilience** - if a container dies, we need a new one to spin up
- **Storage** - where do we store our data?
- **Deployments** - we want a way to canary deployments
- **Updates** - how can we update our containers without downtime?
- **Networking** - how are our containers going to communicate?
- **Service Discovery** - how will they find each other?

What is Kubernetes?

Kubernetes is an open-source platform designed to automate deploying, scaling and operating application containers.

- **Portable:** public, private, hybrid, multi-cloud
- **Extensible:** modular, pluggable, hookable, composable
- **Self-healing:** auto-placement, auto-restart, auto-replication, auto-scaling

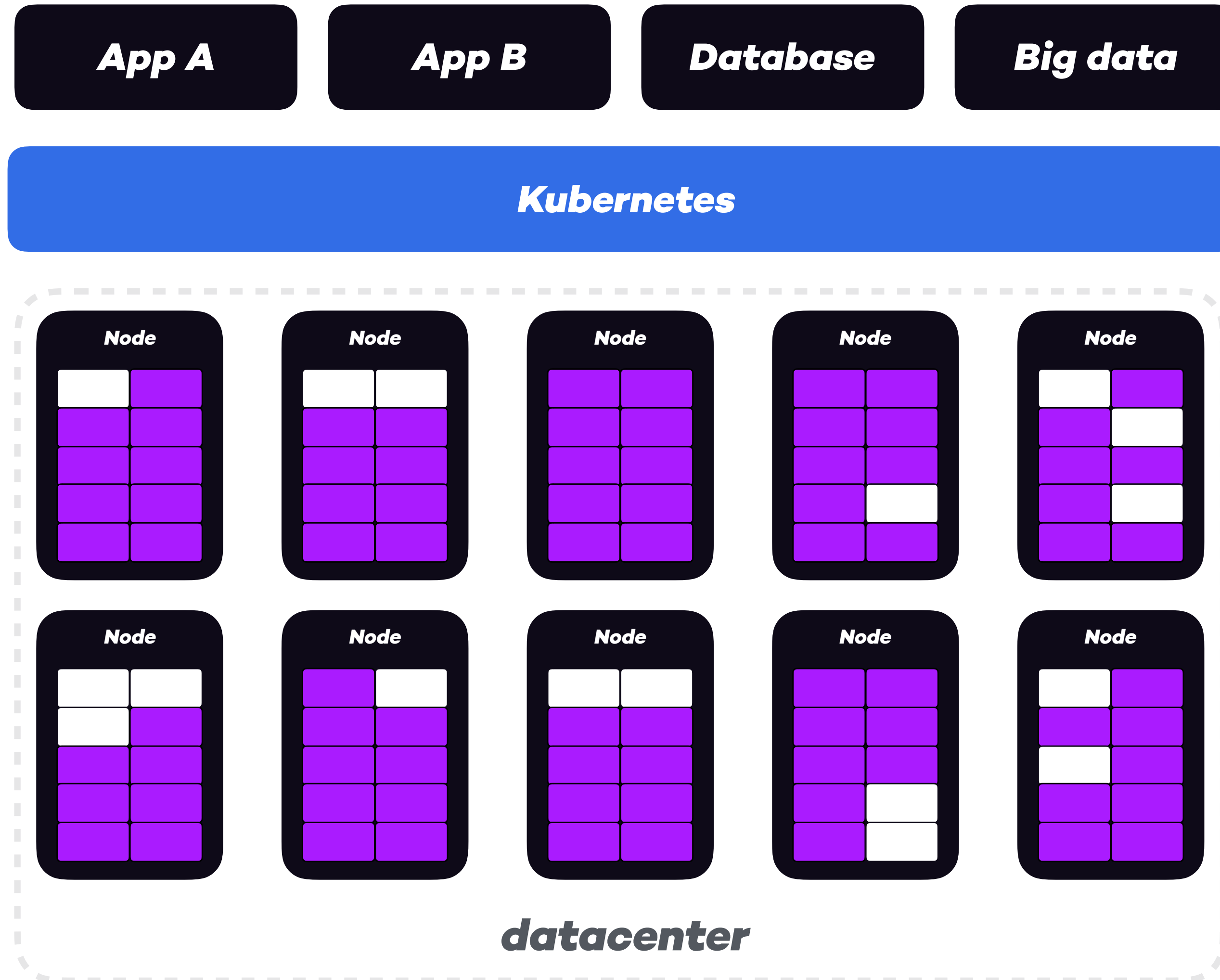
Google started the Kubernetes project in 2014.

Kubernetes builds upon a [decade and a half of experience that Google has with running production workloads at scale](#), combined with best-of-breed ideas and practices from the community.



Source: <https://kubernetes.io/docs/concepts/overview/what-is-kubernetes/>

What does it do?



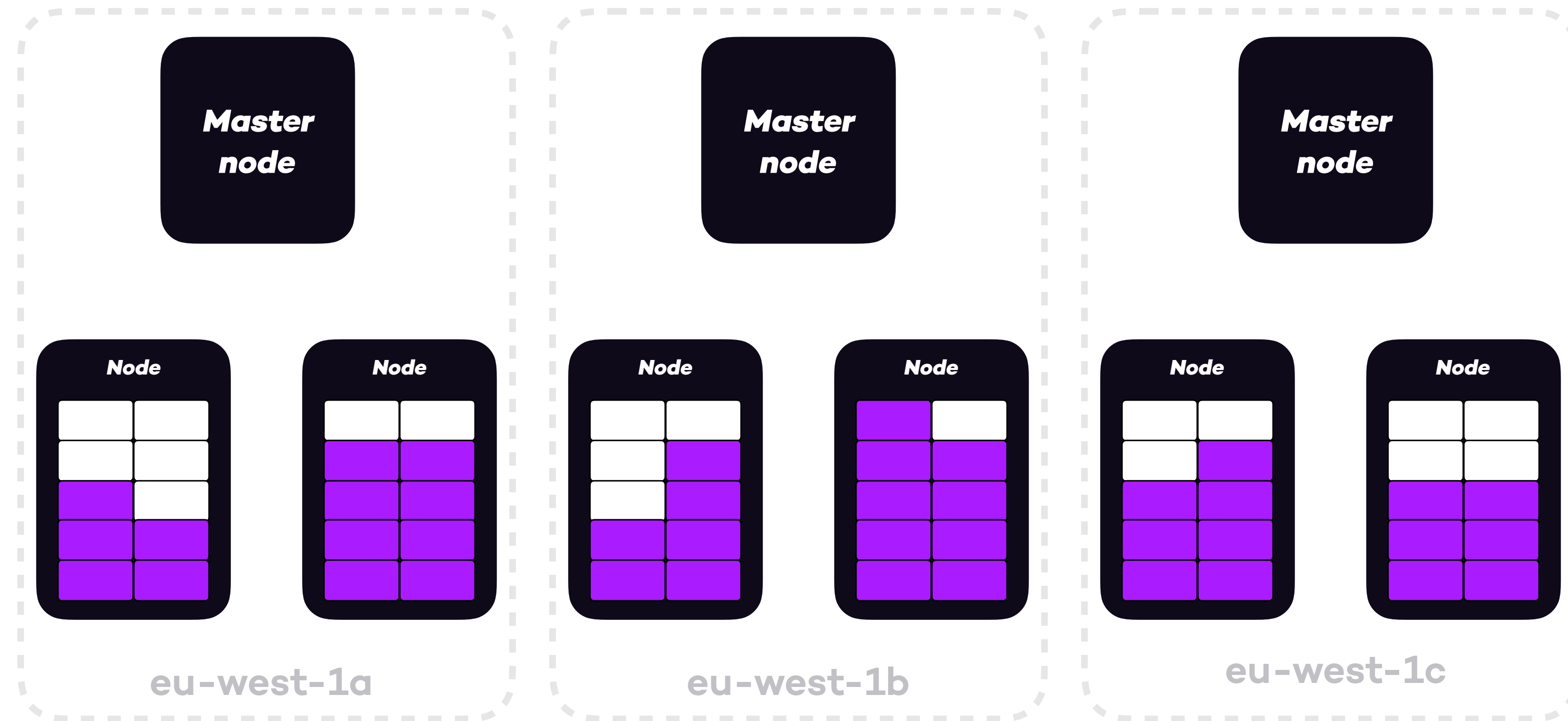
Kubernetes at Lunar Way

**3
Clusters
in AWS**

**Minikube
for local
development**

**KOPS
for
maintaining
cluster**

Highly available



What do we think of it?

Autonomous services

Squads can work independent of other squads.

High availability

Kubernetes takes care of container failures.

Easy maintenance

KOPS to spin up our clusters, and maintain them.

Freedom & Flexibility

We run many different type of workloads in the cluster.
Gives us mobility to become cloud agnostic.

Easy independent deployment

Kubernetes allows us to deploy multiple times a day.

Scalable infrastructure

Scaling the infrastructure is easy, both on node and container level.

“

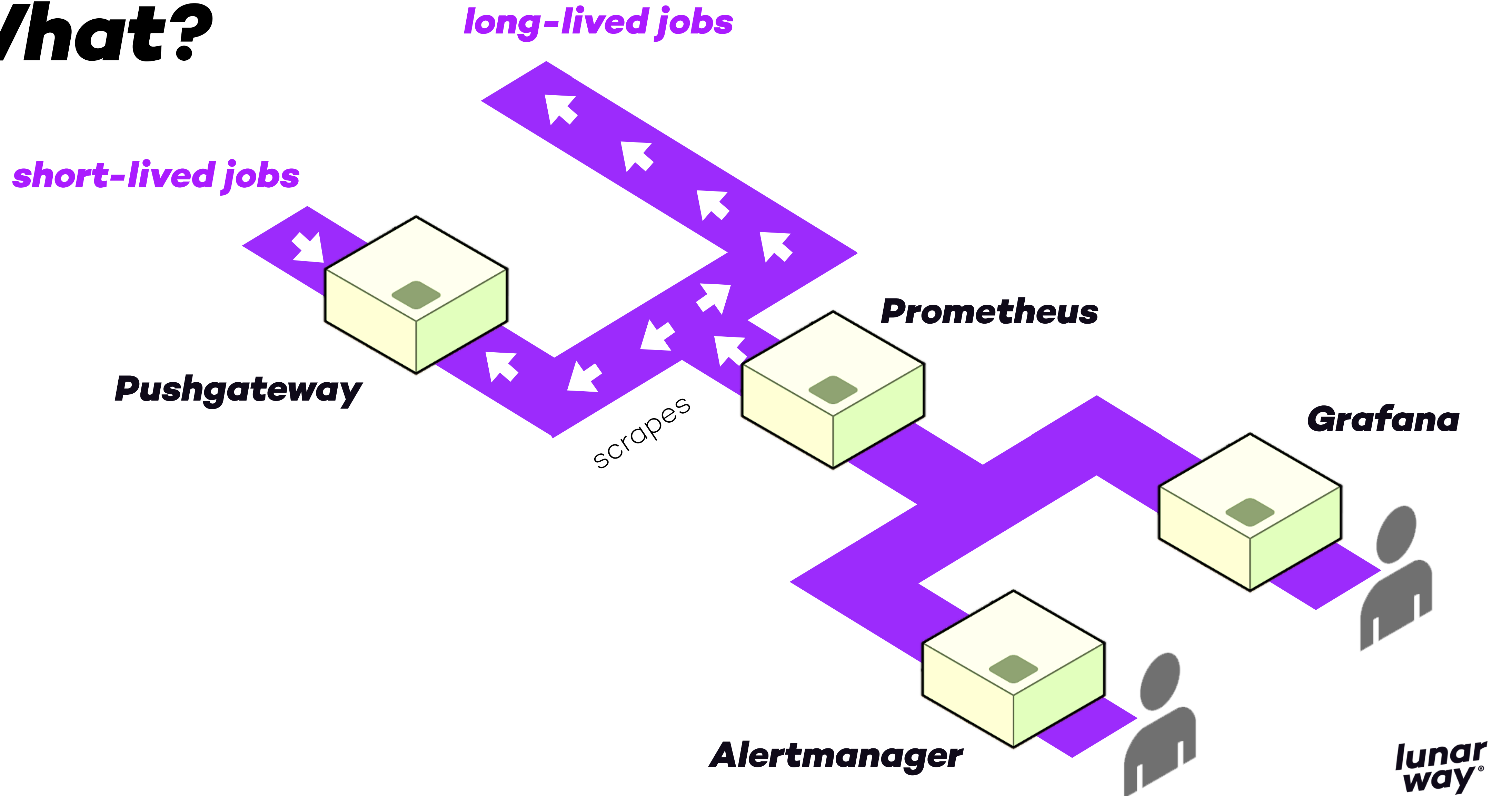
*Containerization transforms the data center from being **machine-oriented** to being **application-oriented***

Burns et al., Borg, Omega, and Kubernetes, 2016

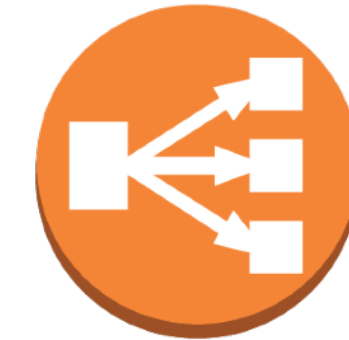
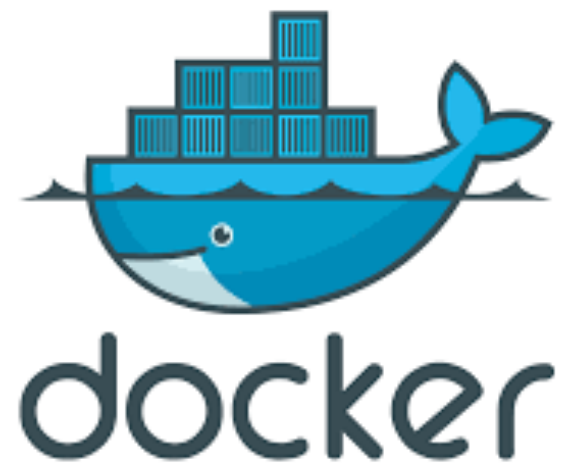
Monitoring with Prometheus



What?



What metrics are we collecting?



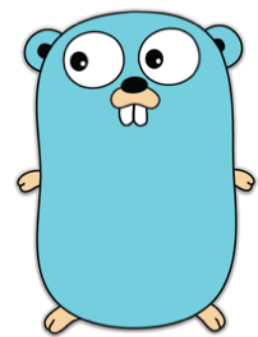
NGINX



Amazon EC2



**Custom
Metrics**



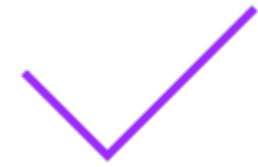


```
annotations:  
  prometheus.io/scrape: 'true'
```

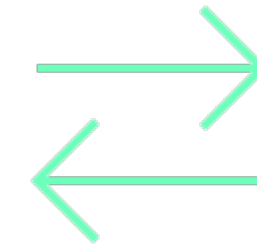
What do we think of it?



Provides great insights to all of our services



Makes it easy for developers to instrument their services

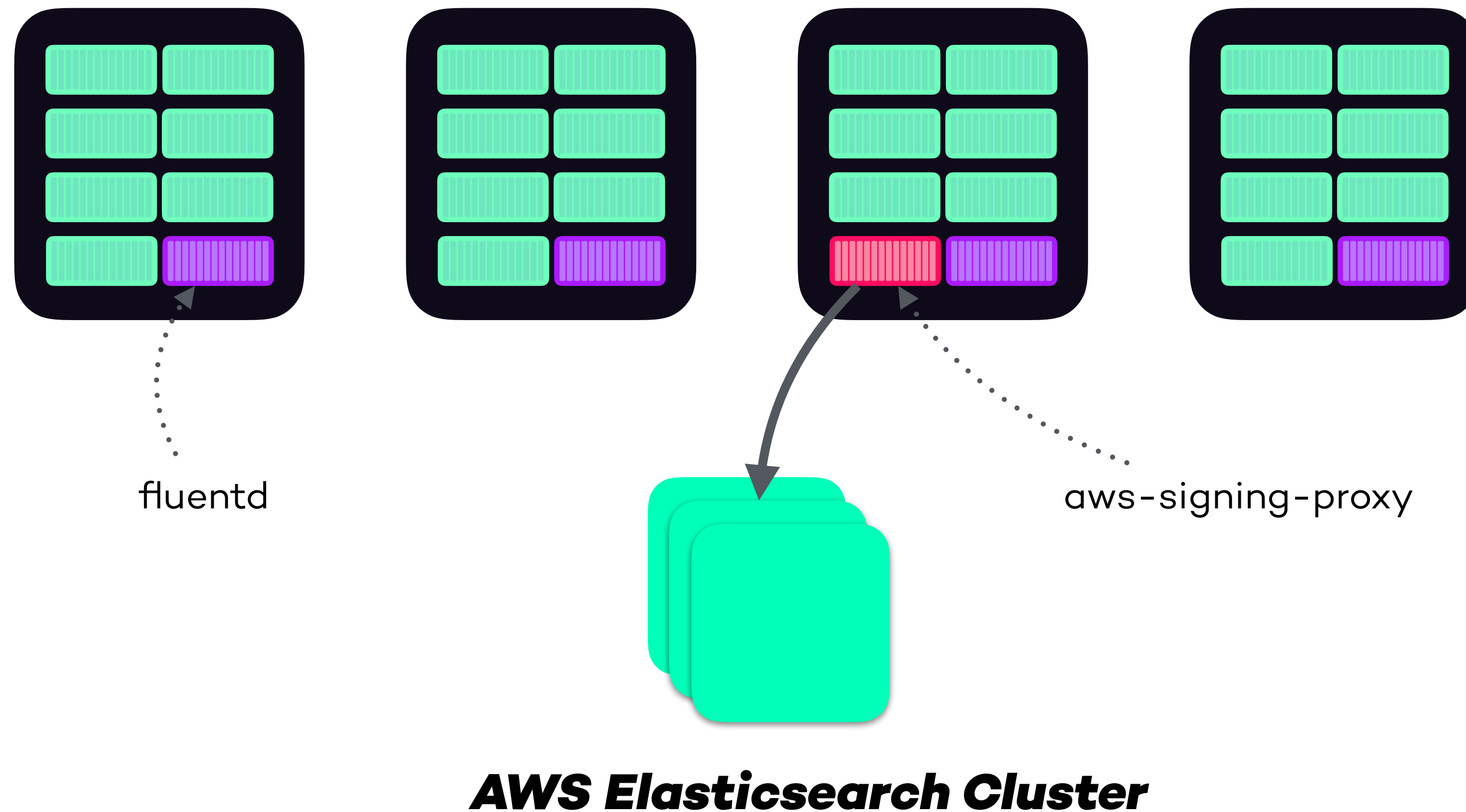


Integrates well with many different services

Log Collection with Fluentd



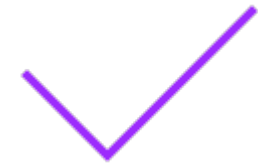
Logging setup



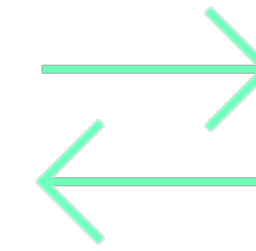
What do we think of it?



Works great with Kubernetes.
Deployed as a DaemonSet



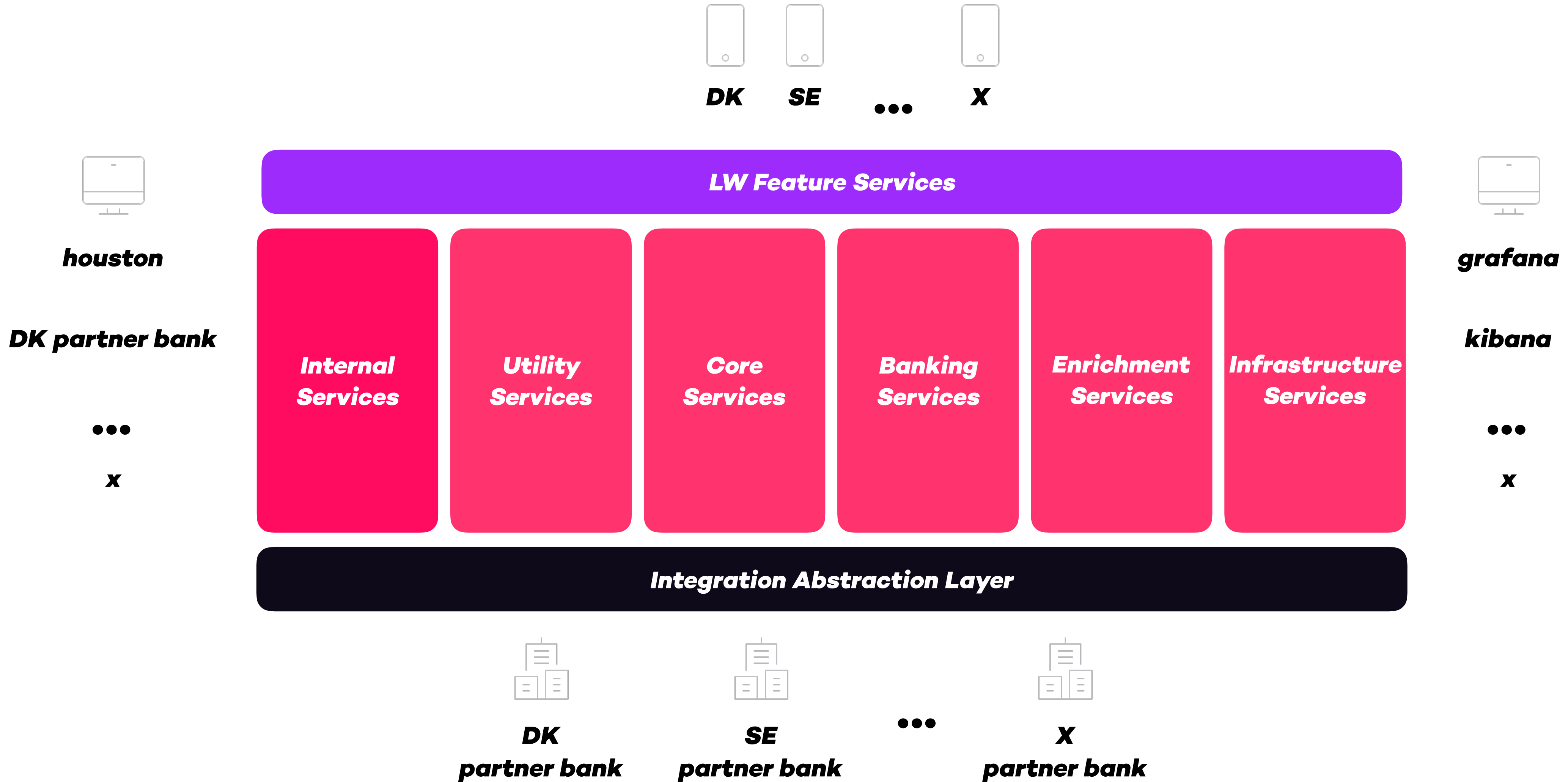
Small memory footprint



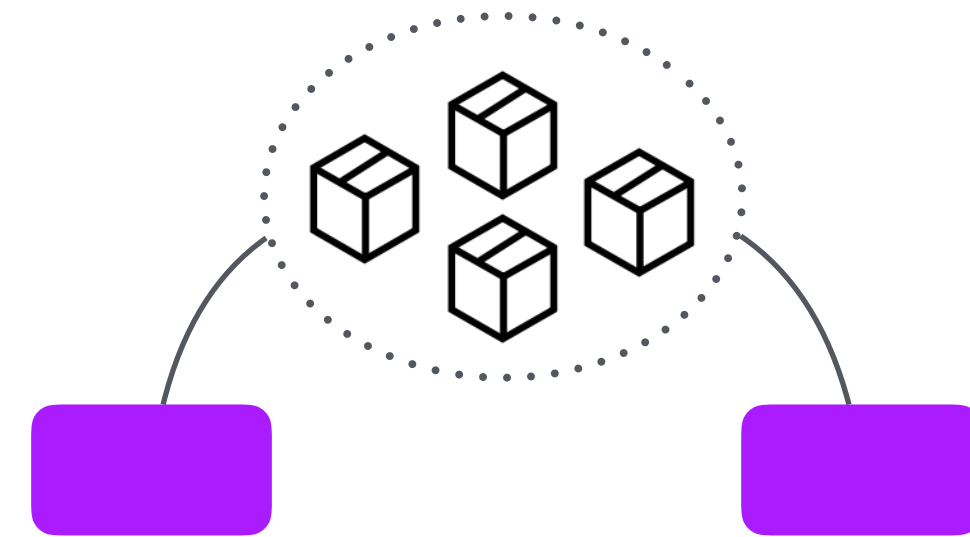
Proven reliability and
performance.

***Back to what it is we are
trying to do...***

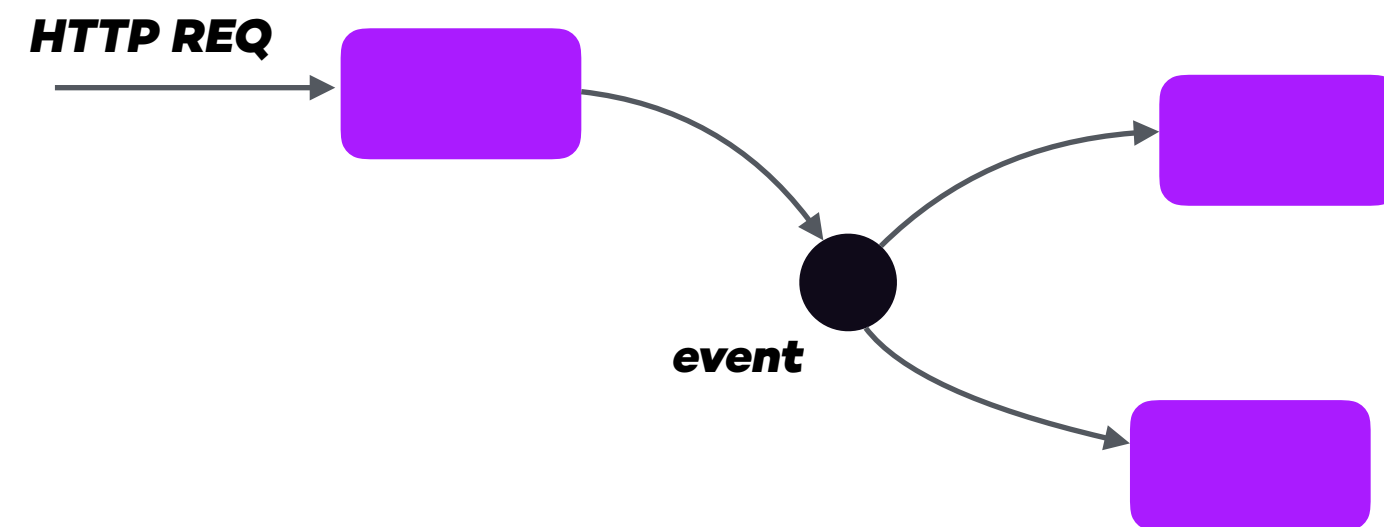
Building a scalable architecture



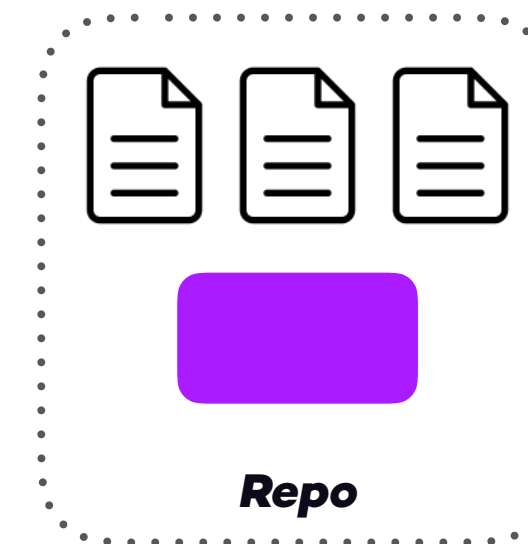
How are we building our services?



Shared Core Dependencies

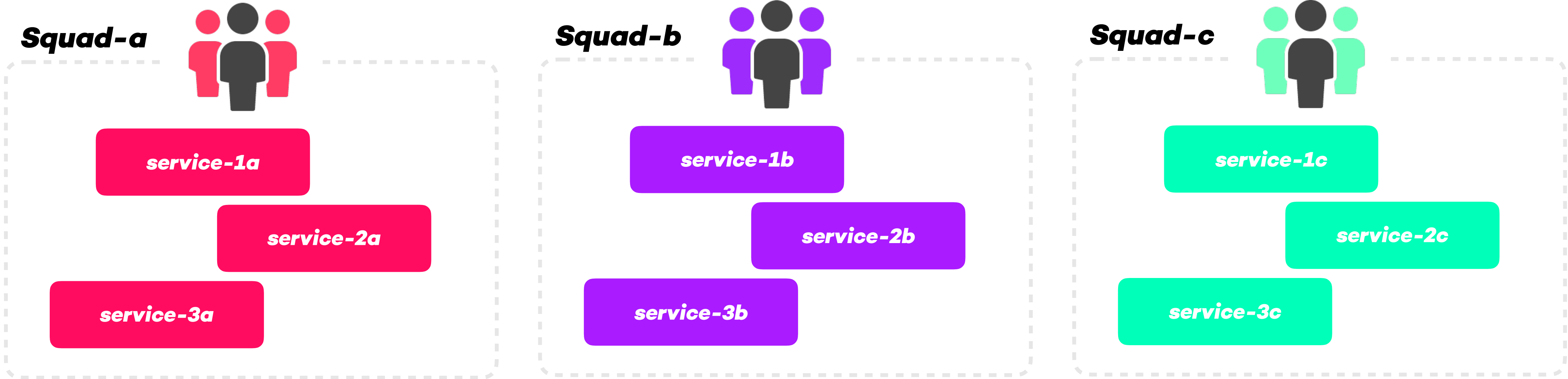


Asynchronous first



Self-contained

Organization & Culture



#squad-core

Environments

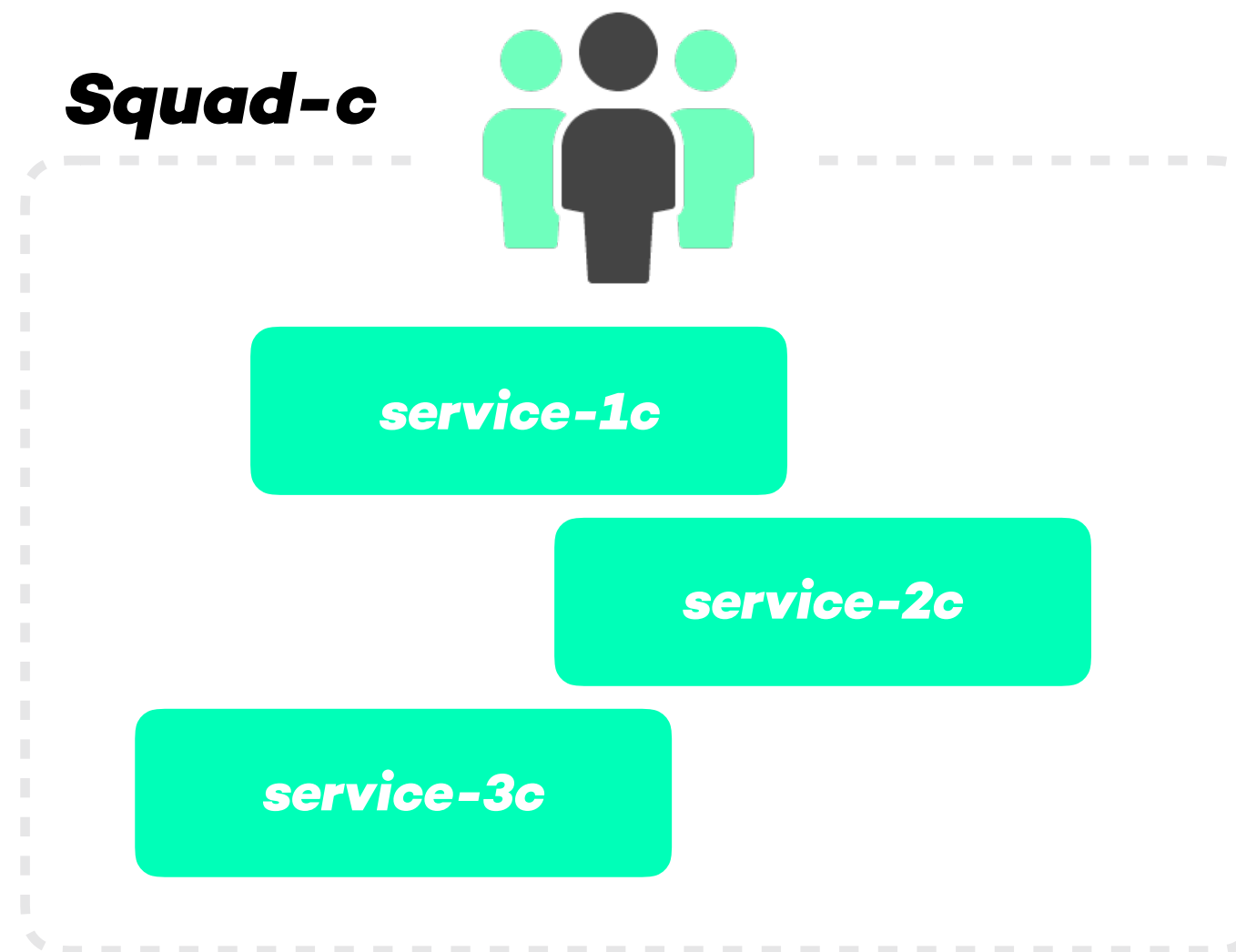
Deployment

Logging

Pipeline

Monitoring

Alerting



Communication

#squad-core

***Provide feature teams with tools and services,
that allows them to move faster and build more
quality in.***

***Did we find Cloud Native
utopia?***

We are on the right path!

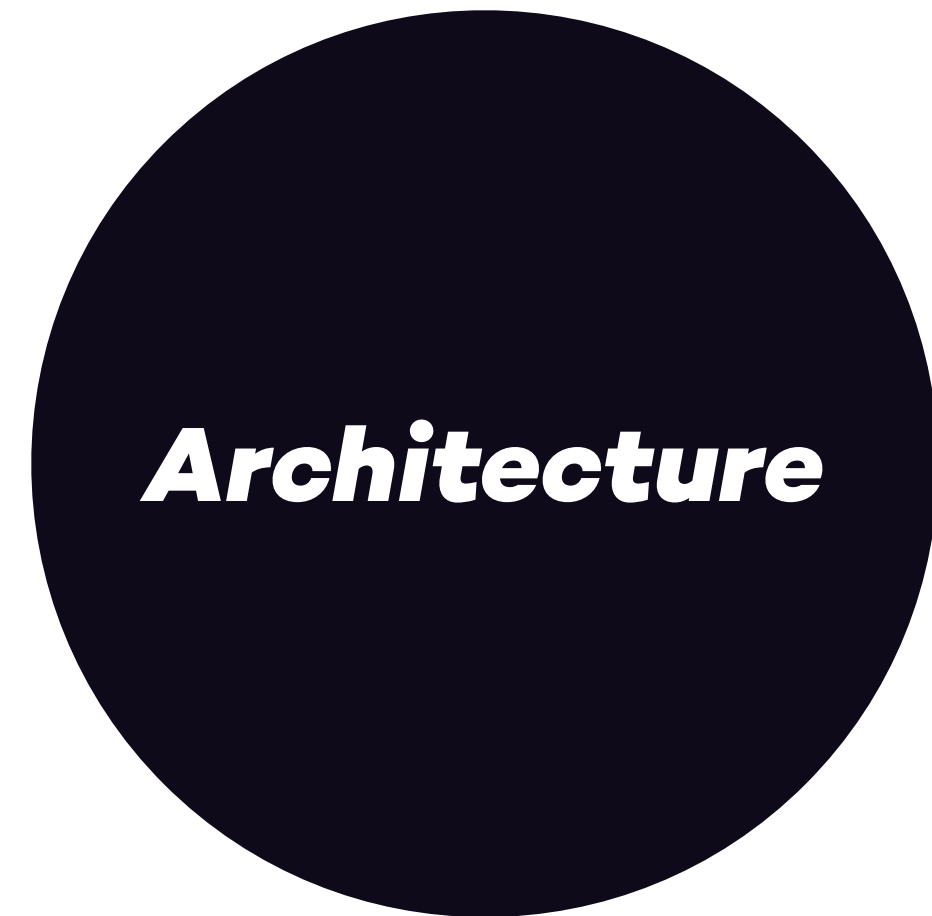
We are doing microservices.

We package services in containers.

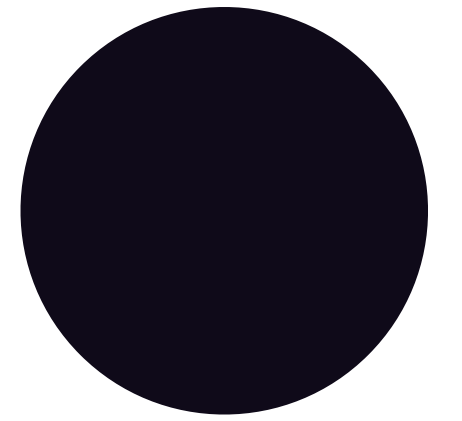
We deploy these in a dynamically scheduled environment.

But, there's still room for improvement...

Challenges



Architecture



Avoid building a distributed monolith

Use bounded context pattern to avoid cross cutting concerns.

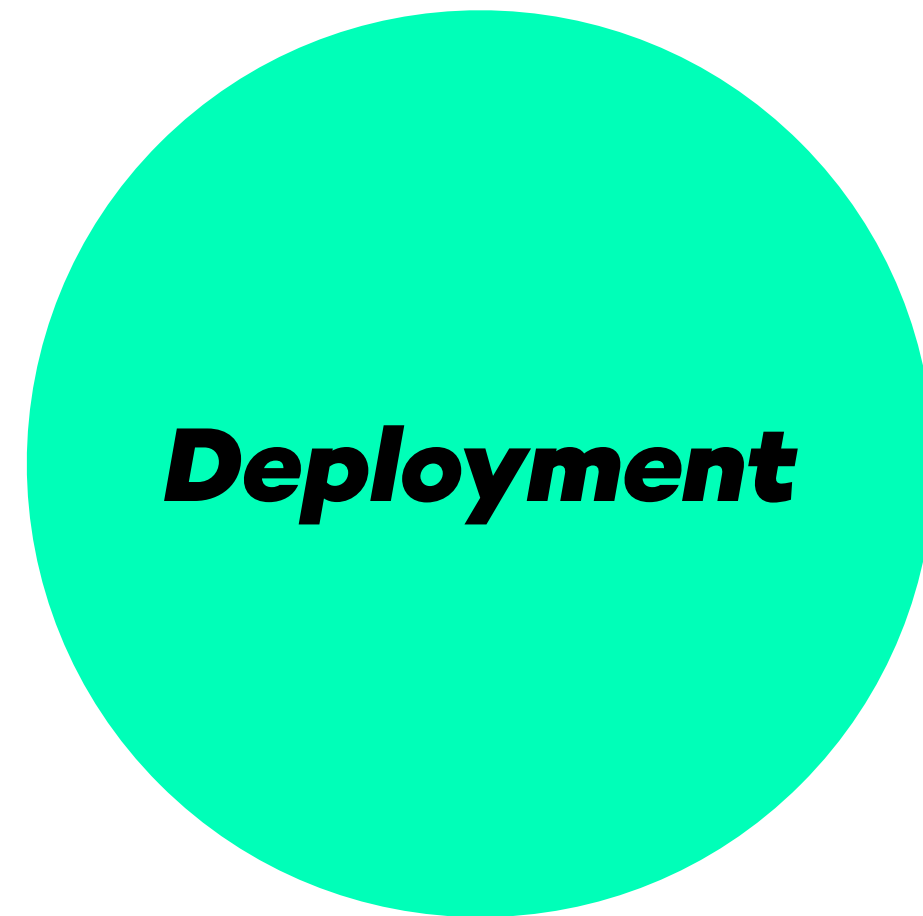
Strangler Application Pattern

Building new functionality as new services

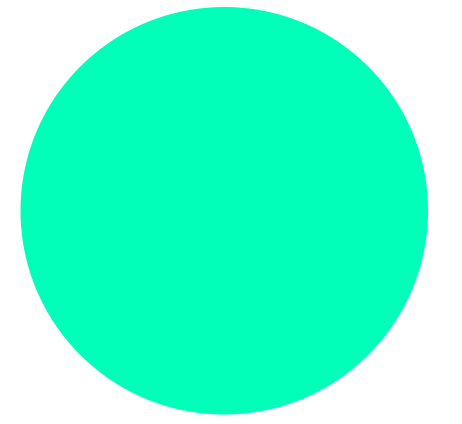
Asynchronous vs Synchronous

Problems when integrating with external partners.
Synchronous calls from app.

Challenges



Deployment



Monolithic deployments

A lot of risk involved, and less frequent deployment.

Cloud Native maturity of CI/CD

Good old Jenkins and scripts to the rescue

Configuration follows Image

Container registry just stores the image.



Challenges



Architecture

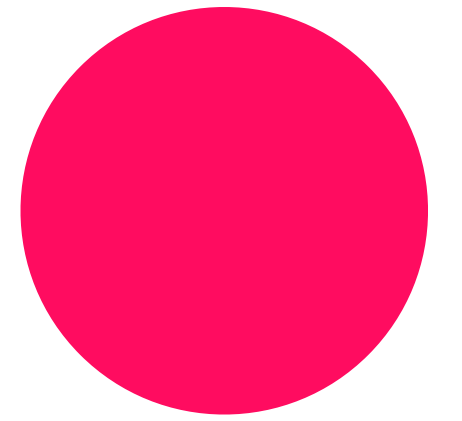


Deployment



***Development
Environment***

Development Environment



Minikube is great in the beginning

What about when running 30 services?

Local cluster boot time is a pain

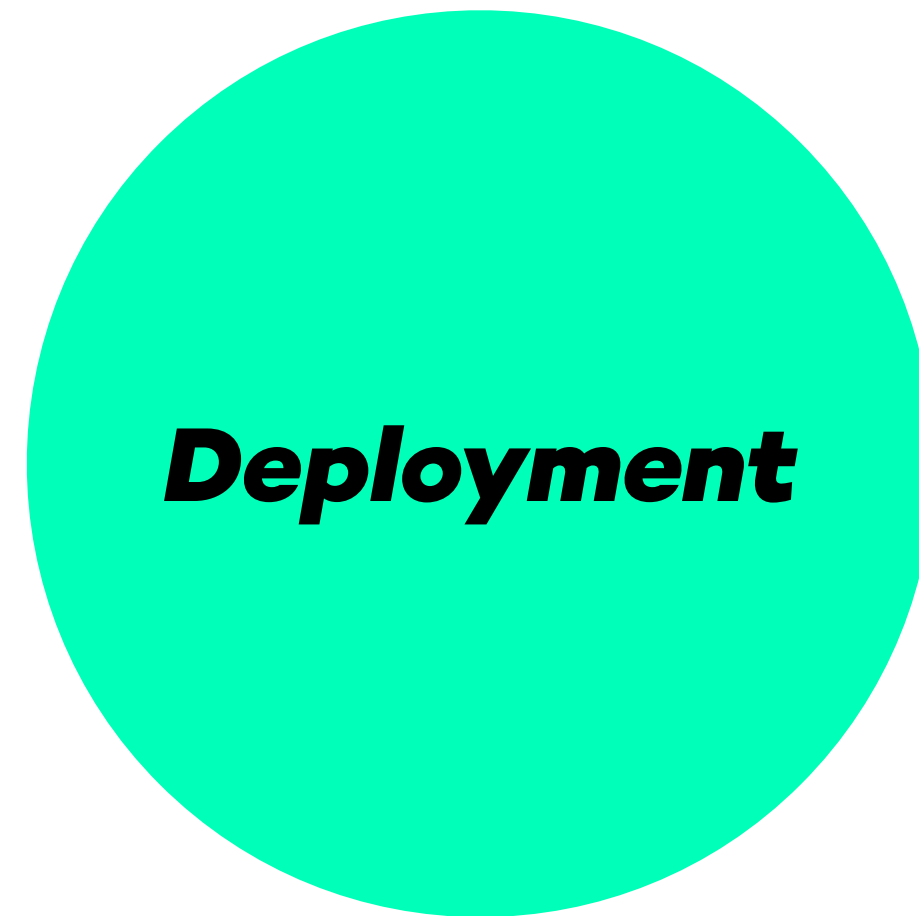
Fetching services over the internet everytime is slow.

Proxy into a cloud environment instead?

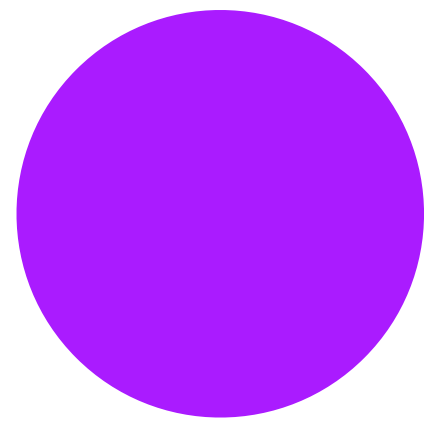
We are looking at a project called telepresence.io



Challenges



Operations



No more SSH'ing into machines

We use kubectl for management.

Kubernetes is moving fast

Keeping up is time-consuming

Kops helps us manage our clusters

Kops makes it fairly easy to update and maintain.



What else are we looking at



***lunar
way***[®]

***THANK
YOU!***

kni@lunarway.com
@phenex

We are hiring!

goto;
copenhagen



 Follow us @gotocph