GitOps

Operations by Pull Request

Kasper Nissen (@phennex)

$ whoami

LUNAR®

Kasper Nissen (@phennex)

Cloud Architect / Site Reliability Engineer at Lunar

CNCF Ambassador

Certified Kubernetes Administrator

Cloud Native Aarhus (Cloud Native Copenhagen)

Cloud Native Nordics

Occasional speaker at Meetups, Conferences

Blog: kubecloud.io

kubernetes
Removing failing pod
23
slave2
192.168.1.202
slave3
192.168.1.203
slave4
192.168.1.204

Agenda

What is Gitops?

Flavours of Gitops

GitOps at LUNAR

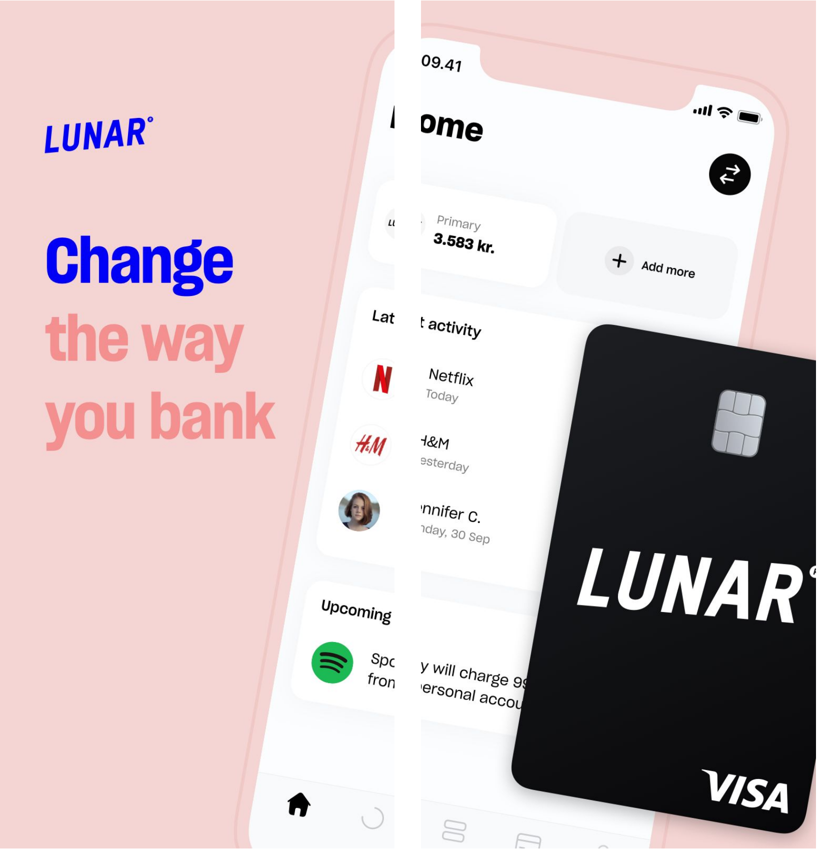
Chaos Engineering

GitOps at Cloud Native Nordics

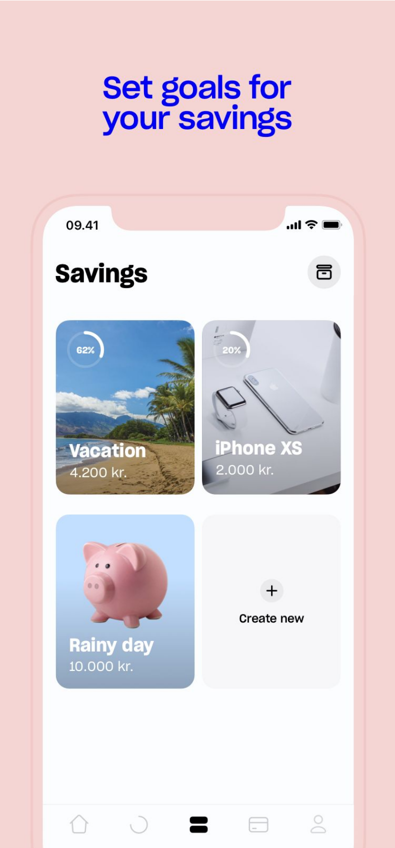
Progressive Delivery

About LUNAR

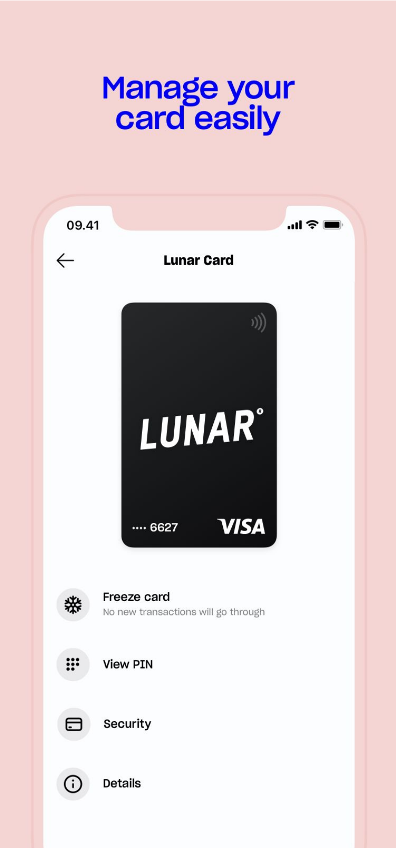
License to build a bank.



Set goals for your savings



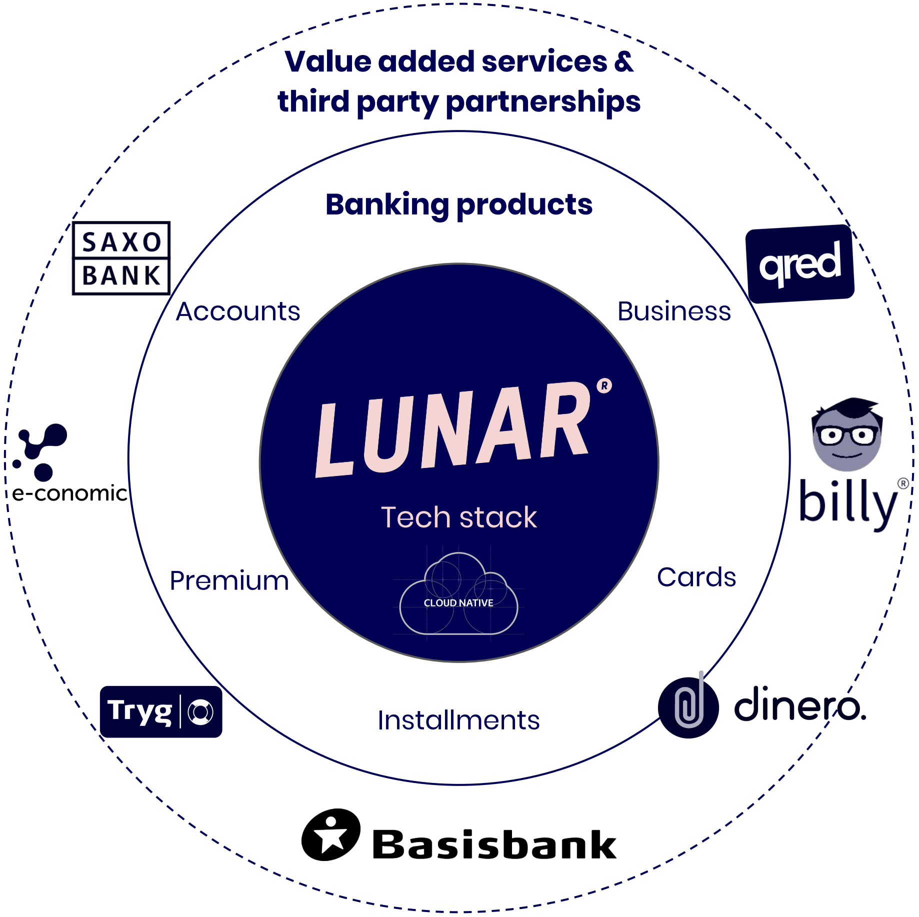
Manage your card easily



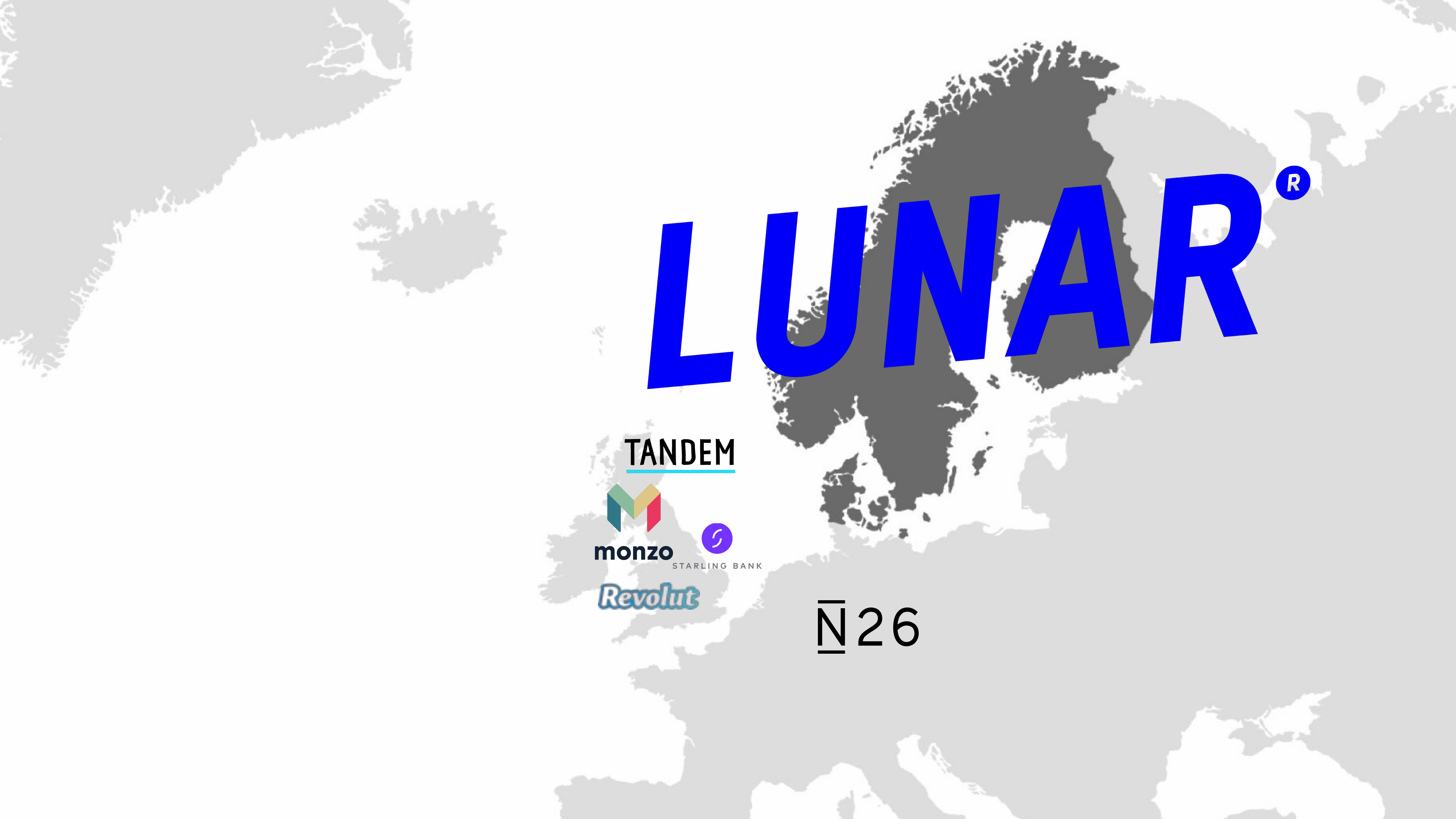
Building a Nordic bank is an enabler to our vision…

… on the back of the bank we are expanding to a Financial Super App

It’s basically a **single portal** to a wide range of products and services from **core banking to lifestyle**, shopping, hospitality and transportation driven by **user experiences**



Rethinking the banking experience


LUNAR®
TANDEM
monzo
STARLING BANK
Revolut
N26

Faster
Better
Stronger

GIT
OPS

What does reconciliation mean?

reconciliation

noun

UK /ˌrek.ᵊnˌsɪl.iˈeɪ.ʃᵊn/ US /ˌrek.ᵊnˌsɪl.iˈeɪ.ʃᵊn/

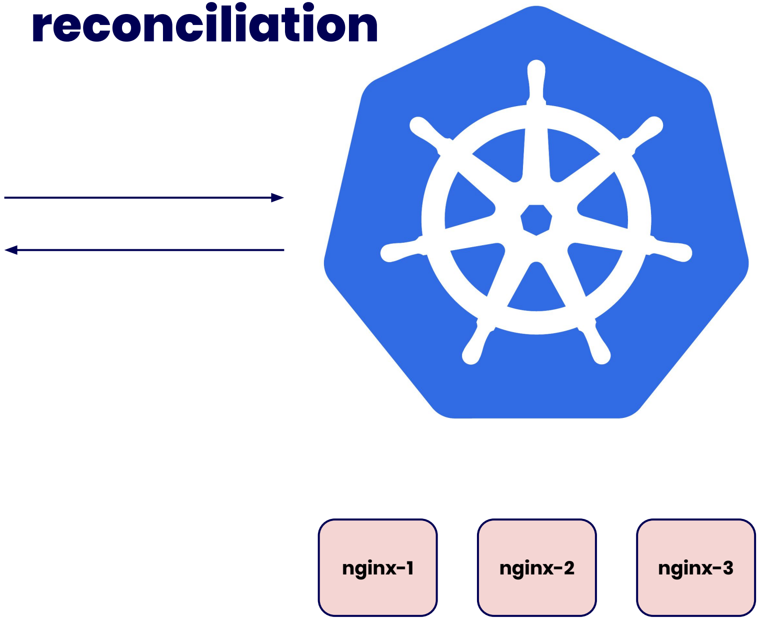
the process of making two people or groups of people friendly again after they have argued seriously or fought and kept apart from each other, or a situation in which this happens

the process of making two opposite beliefs, ideas, or situations agree

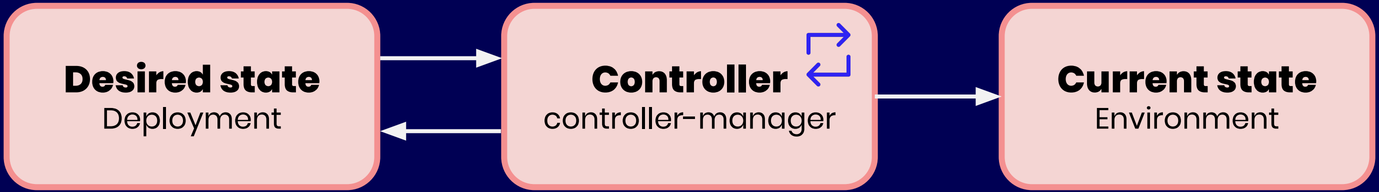
Source: https://dictionary.cambridge.org/dictionary/english/reconciliation

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx:1.17.5
        ports:
          - containerPort: 80
```

application reconciliation



reconciliation


Desired state
Deployment

Controller
controller-manager

Current state
Environment

What is GitOps

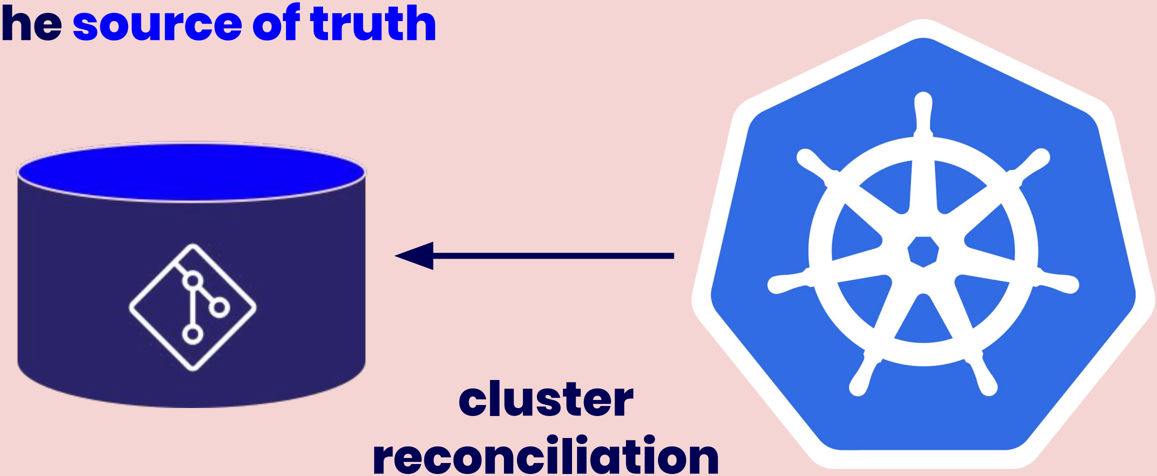
gitops

UK /gɪt/ɒp/ US /git/ɑːp/

GitOps is a way to do Kubernetes cluster management and application delivery. It works by using Git as a single source of truth for declarative infrastructure and applications. With Git at the center of your delivery pipelines, developers can make pull requests to accelerate and simplify application deployments and operations tasks to Kubernetes.

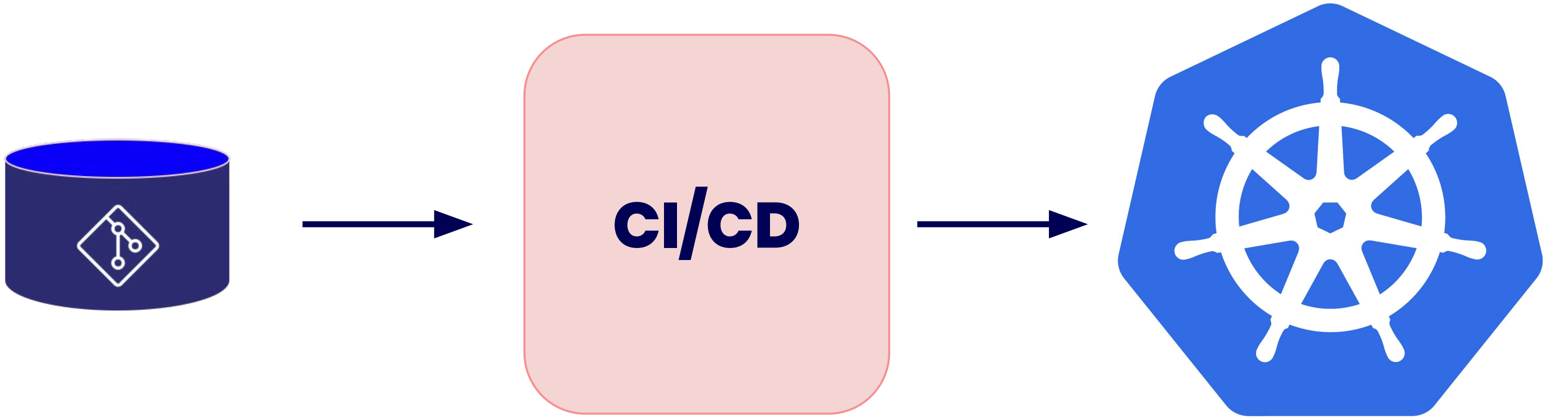
Source : https://www.weave.works/technologies/gitops/

The source of truth



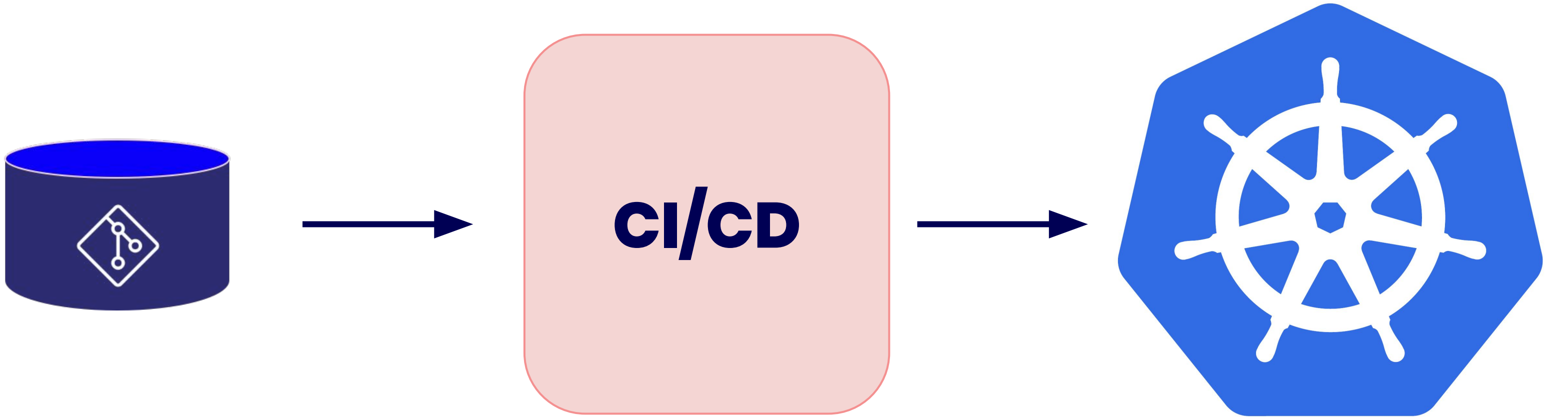
cluster reconciliation

What’s wrong with kubectl apply?



CI/CD

```
kubectl apply -f deploy/
```



CI/CD

What's wrong?

- CI/CD needs write access to your clusters
- How to track rollout failures?
- No audit trail of the kubernetes resources
- No single source of truth of the state in the cluster

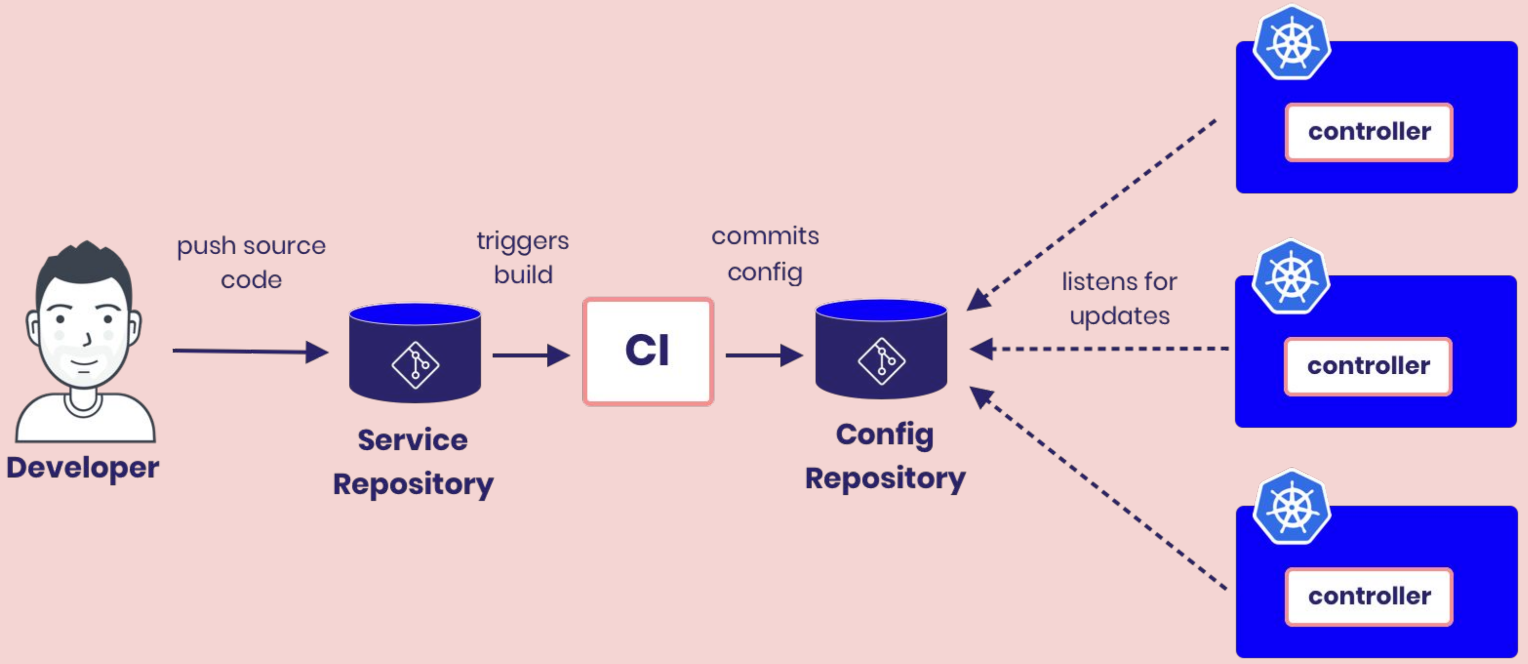
It's imperative. **It should be declarative.**

Flavours of GitOps

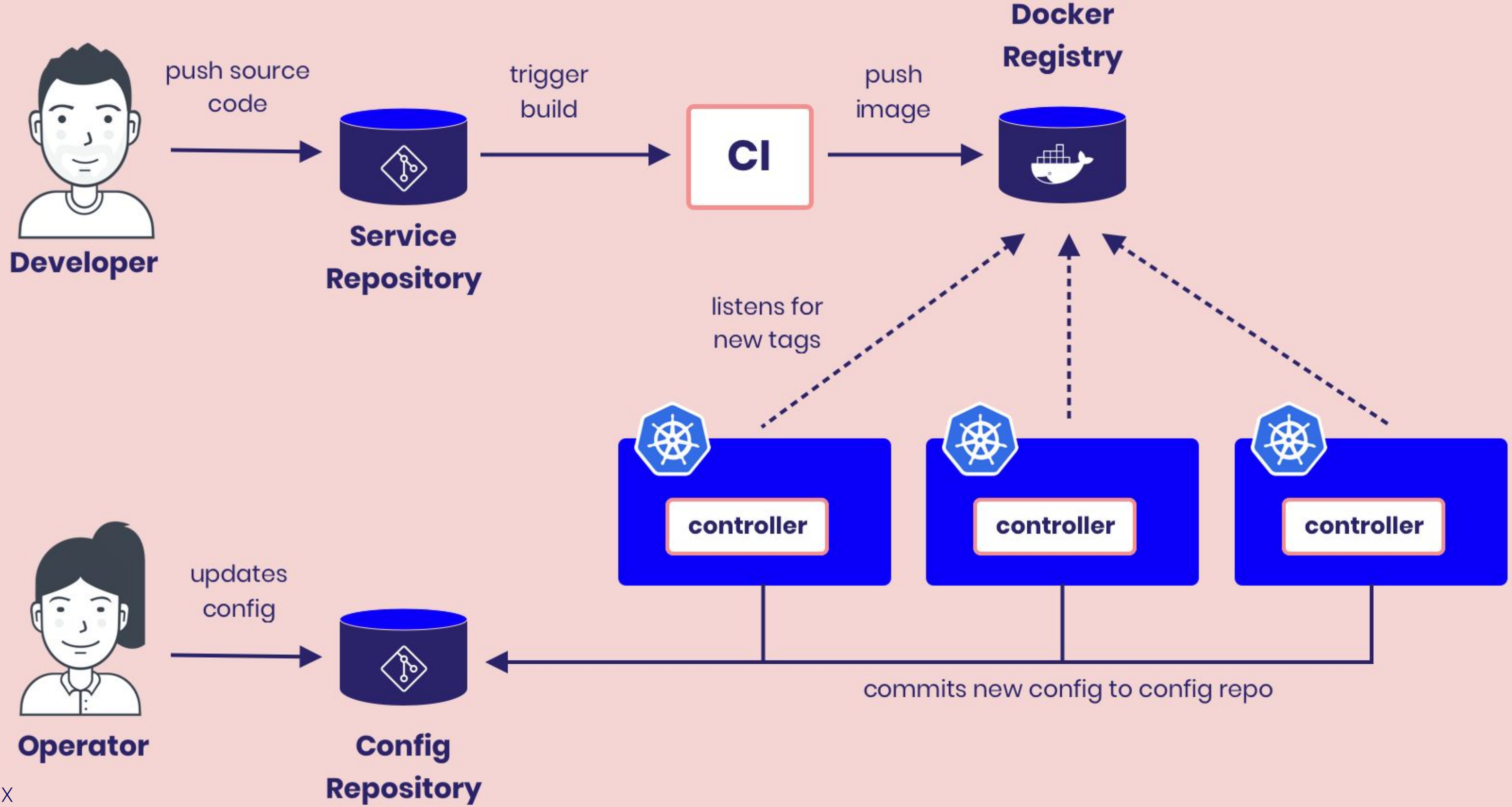
Flavours

- Decentralized One-way flow
- Decentralized Two-way flow
- Centralized flow

Decentralized one-way flow



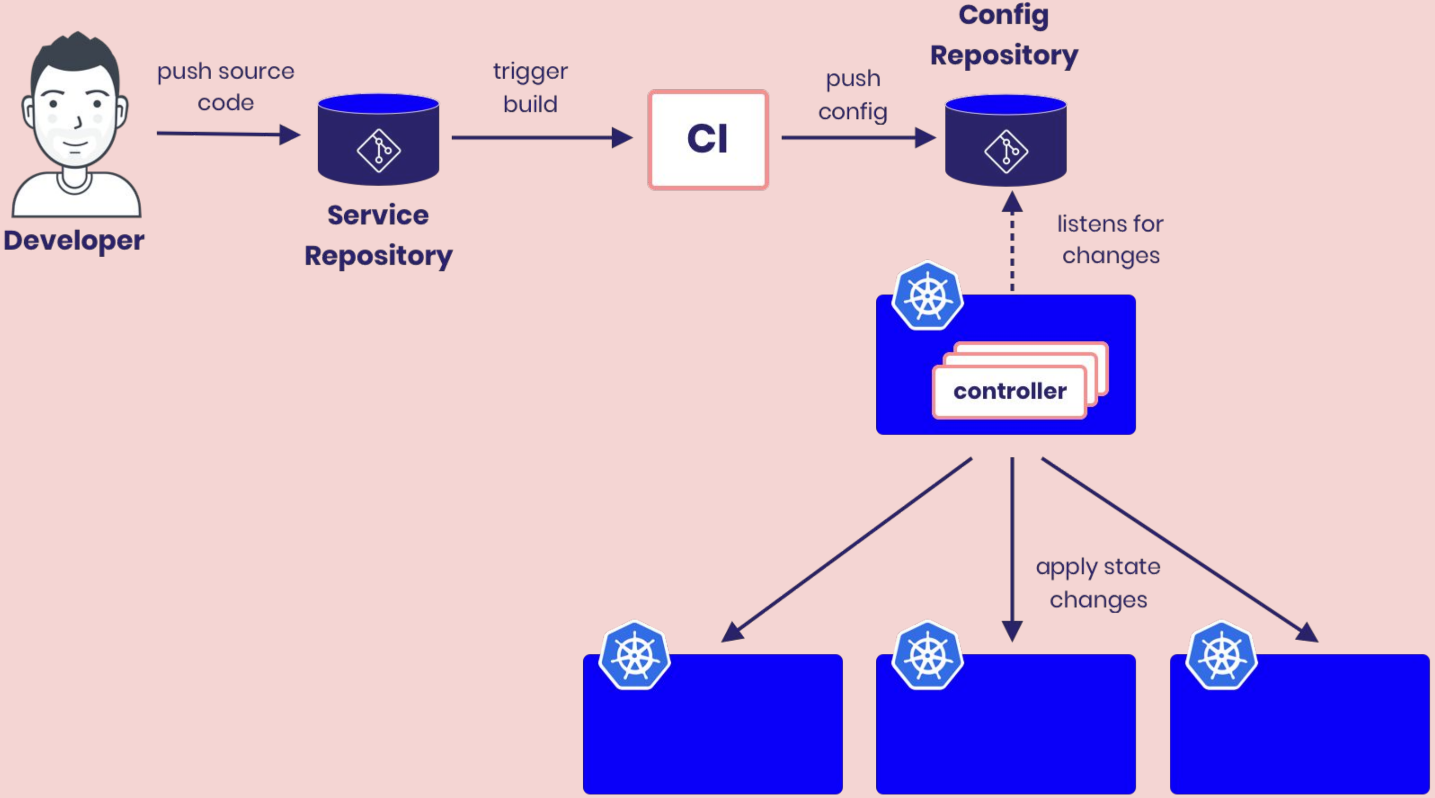
Decentralized two-way flow



@phennex

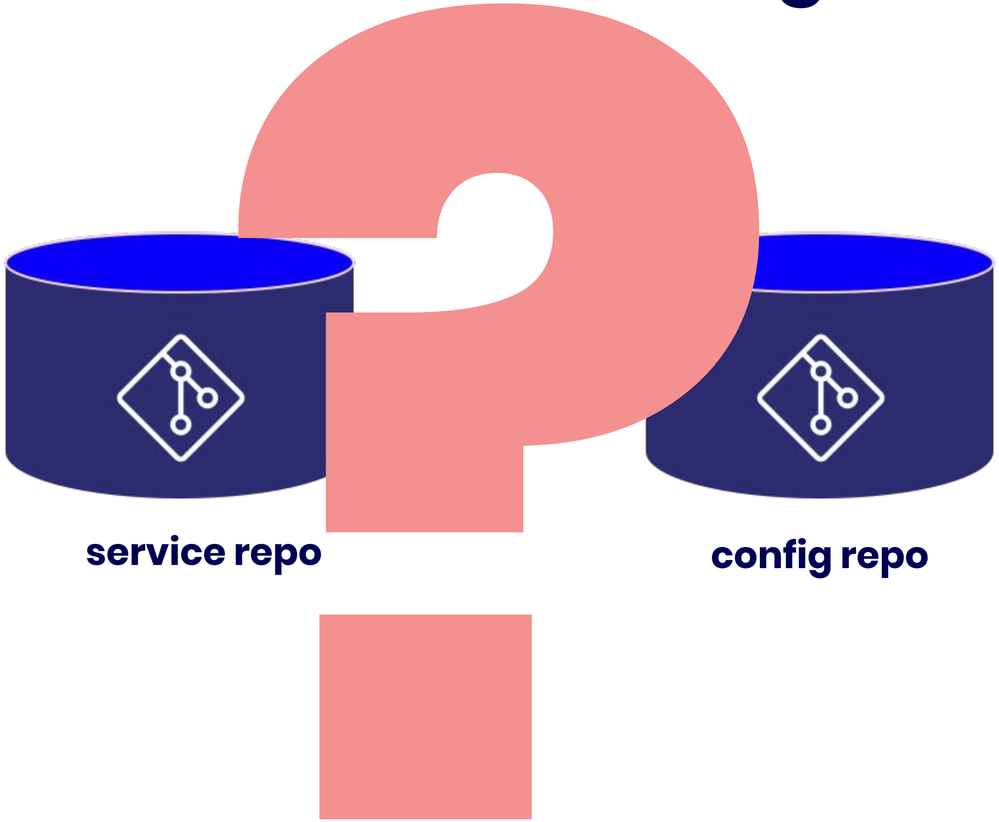
LUNAR

Centralized flow



Where should service config live?

Where should service config live?



Implementations/tooling

originally by weaveworks (now CNCF project)

originally by intuit

Argo CD

a declarative, GitOps continuous delivery tool for Kubernetes

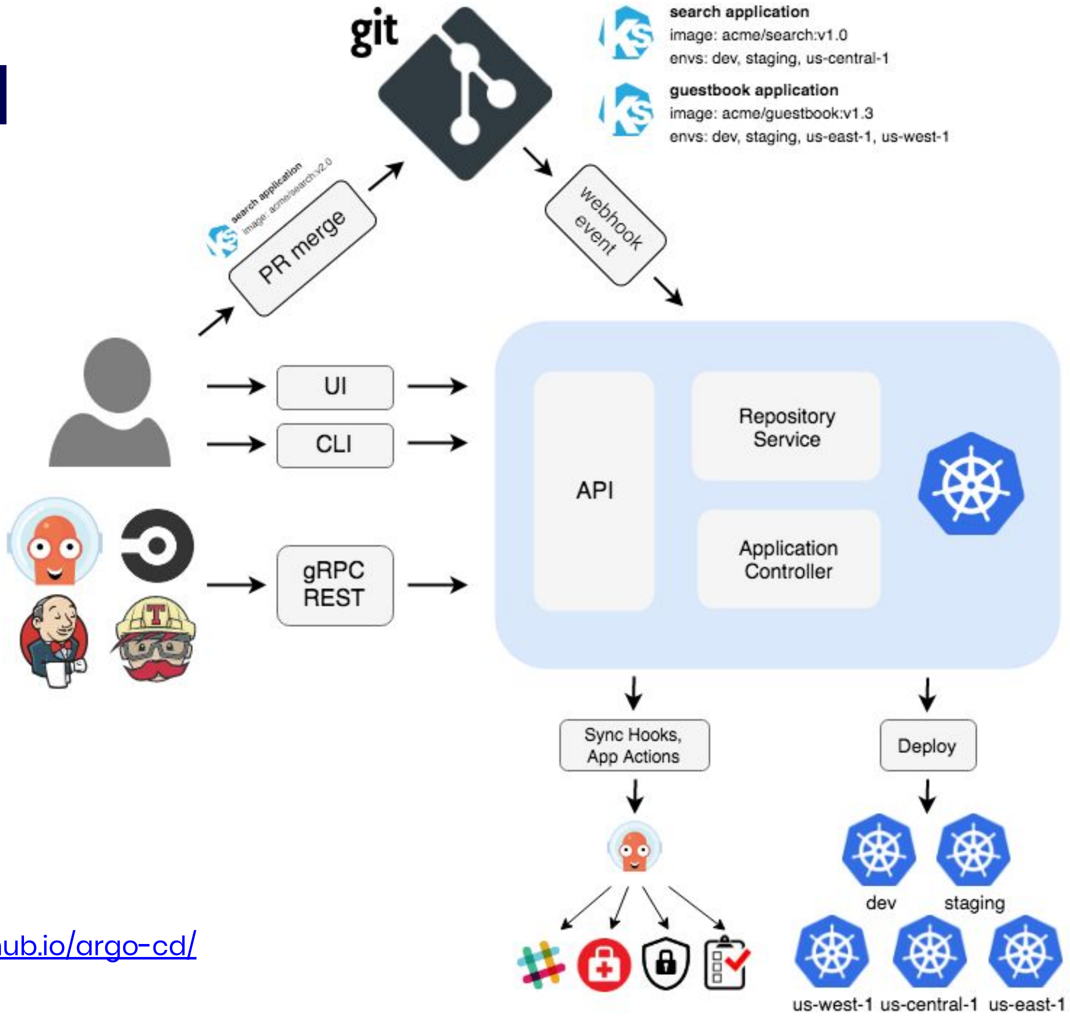
Features

- **Automated deployment** of applications to specified target environments
- Support for multiple config management/templating tools (Kustomize, Helm, Ksonnet, Jsonnet, plain-YAML)
- Ability to **manage** and **deploy** to **multiple clusters**
- SSO Integration (OIDC, OAuth2, LDAP, SAML 2.0, GitHub, GitLab, Microsoft, LinkedIn)
- Multi-tenancy and RBAC policies for authorization
- **Rollback/Roll**-anywhere to any application configuration committed in Git repository
- Health status analysis of application resources
- Automated configuration **drift detection** and **visualization**

Features

- **Automated or manual syncing** of applications to its **desired state**
- **Web UI** which provides real-time view of application activity
- **CLI** for automation and CI integration
- Webhook integration (GitHub, BitBucket, GitLab)
- Access tokens for automation
- PreSync, Sync, PostSync hooks to support complex application rollouts (e.g.blue/green & canary upgrades)
- Audit trails for application events and API calls
- Prometheus metrics
- Parameter overrides for overriding ksonnet/helm parameters in Git

argo-cd



Source: https://argoproj.github.io/argo-cd/

@phennex

Flux CD

The GitOps operator for Kubernetes

What is Flux CD?

"Flux is a tool that automatically **ensures that the state of your Kubernetes cluster matches the configuration you've supplied in Git.** It uses an operator in the cluster to trigger deployments *inside* Kubernetes, which means that you don't need a separate continuous delivery tool."

Source: www.fluxcd.io

Why Flux CD

Declarative

Describe the entire desired state of your system in Git. This includes apps, configuration, dashboards, monitoring, and everything else.

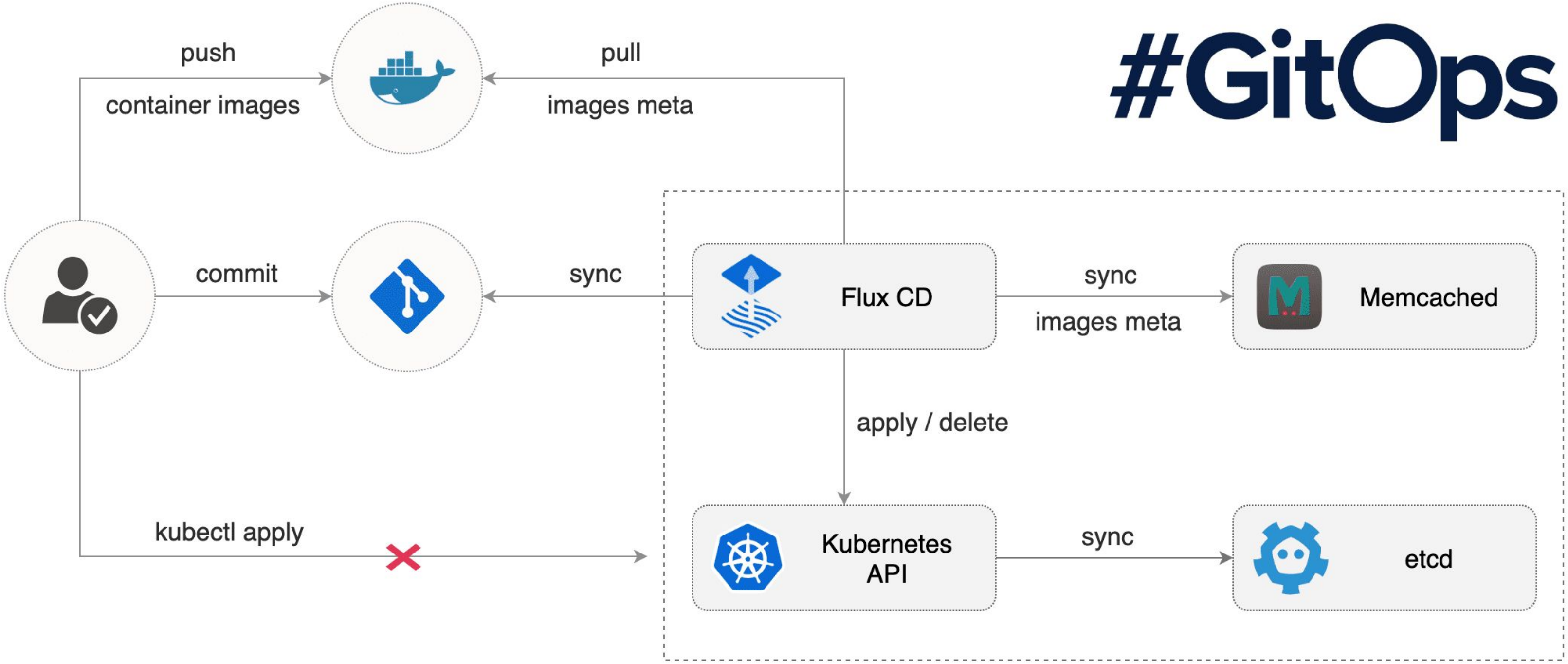
Automated

Use YAML to enforce conformance to the declared system. You don't need to run kubectl because all changes go through Git. Use diffing tools to detect divergence between observed and desired state and receive notifications.

Code, not containers

With Flux, everything is controlled through pull requests, which means no learning curve for new developers. Just use your standard PR process. Your Git history provides a sequence of transactions, allowing you to recover system state from any snapshot. Fix a production issue via pull request rather than making changes to the running system.

The Flux CD workflow



Source: https://fluxcd.io/

@phennex

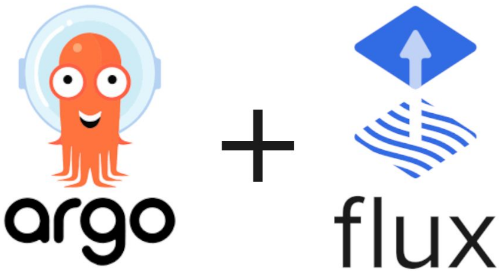
LUNAR

Argo Flux

The two biggest GitOps projects joining forces

Argo Flux

- Extract common functionality into **gitops-engine**
  - Access to Git repositories
  - Kubernetes resource cache
  - Manifest Generation
  - Resources reconciliation
  - Sync Planning



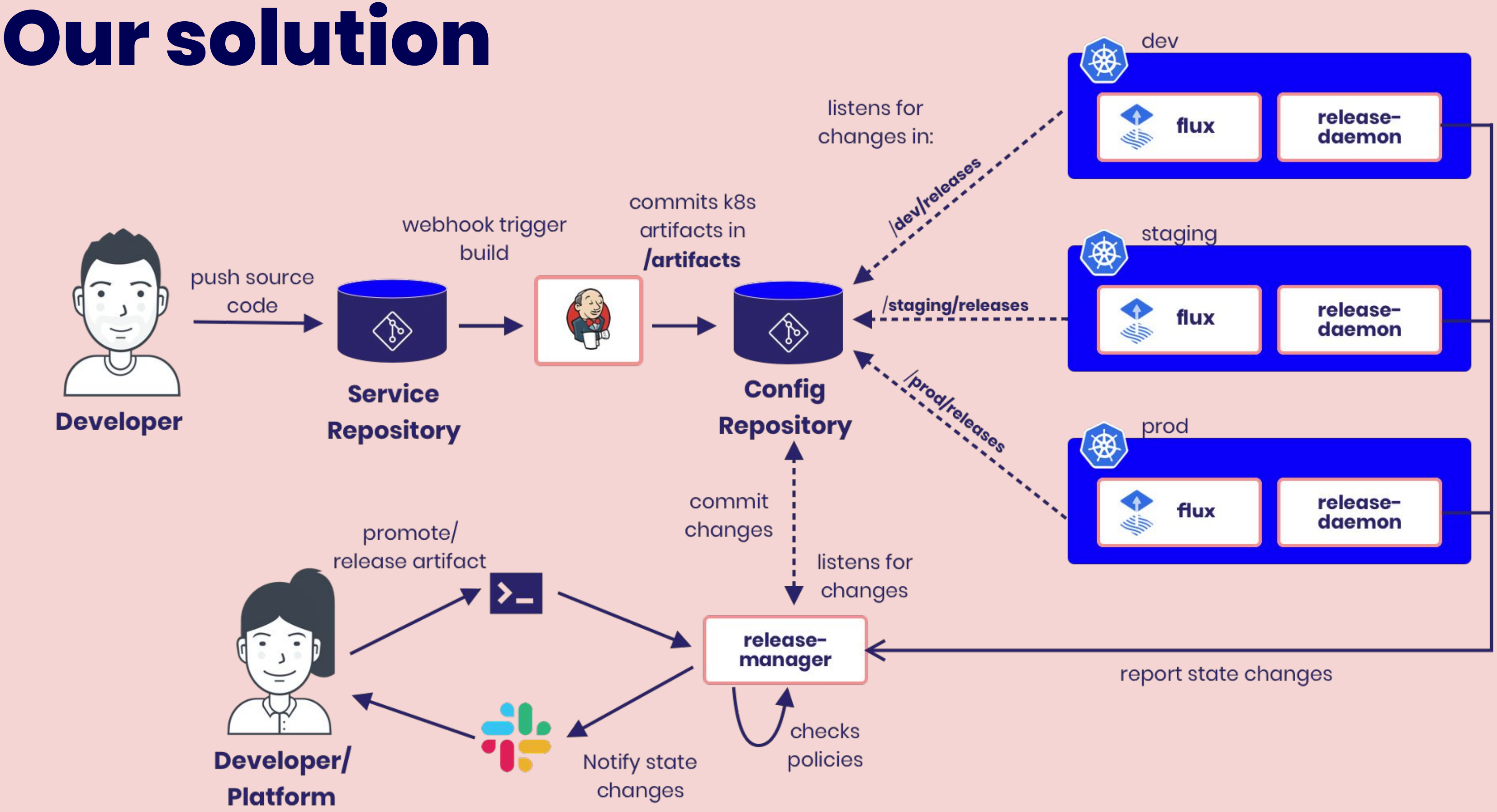
Source: https://github.com/argoproj/gitops-engine

GitOps at Lunar

Why GitOps?

- Audit trail of deployments
- Limit access to clusters
- Make Disaster Recovery an unpainful event

Our solution



@phennex

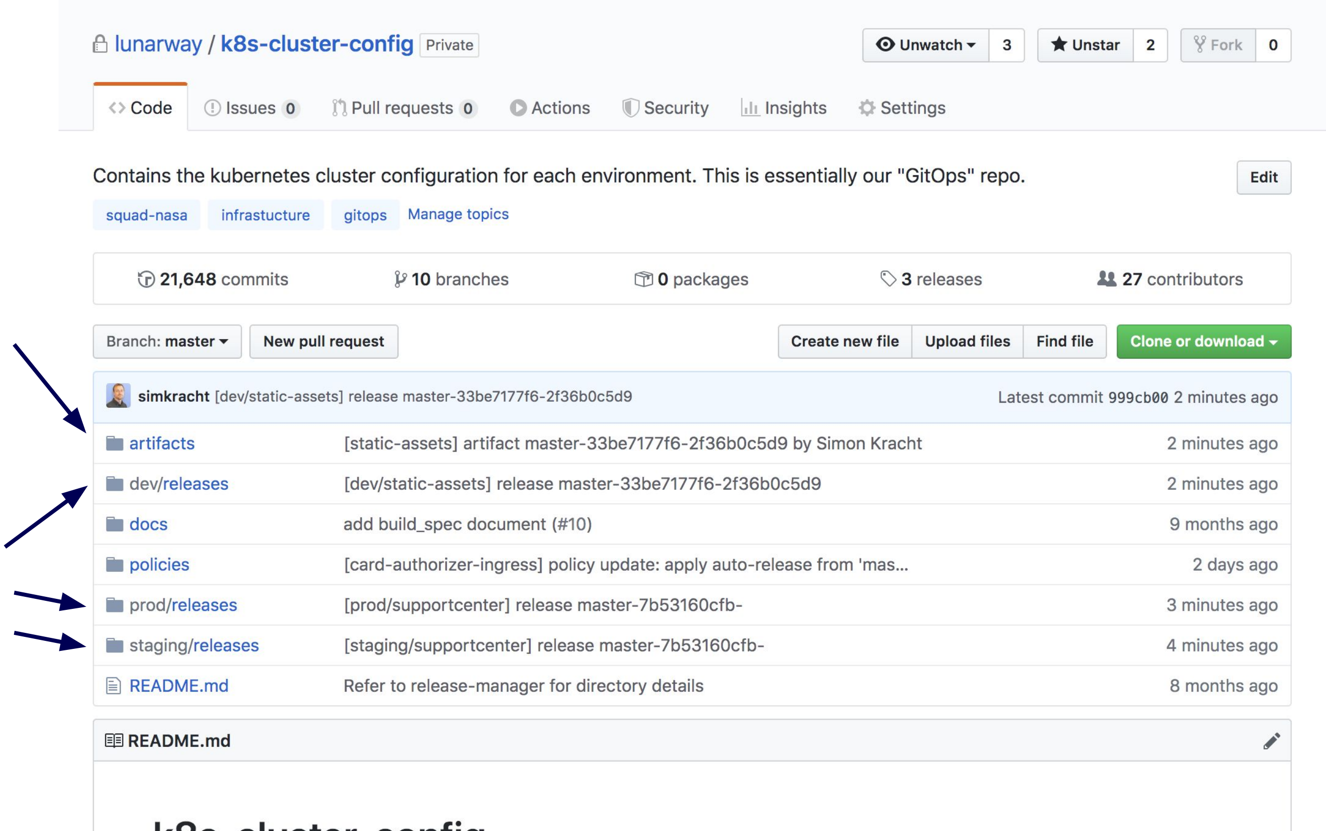
LUNAR

One or more config repos?

Our solution

Artifacts
pushed by Jenkins

Environments
controlled by
release-manager



Contains the kubernetes cluster configuration for each environment. This is essentially our "GitOps" repo.

simkracht [dev/static-assets] release master-33be7177f6-2f36b0c5d9 Latest commit `999cb00` 2 minutes ago

artifacts	[static-assets] artifact master-33be7177f6-2f36b0c5d9 by Simon Kracht	2 minutes ago
dev/releases	[dev/static-assets] release master-33be7177f6-2f36b0c5d9	2 minutes ago
docs	add build_spec document (#10)	9 months ago
policies	[card-authorizer-ingress] policy update: apply auto-release from 'mas...	2 days ago
prod/releases	[prod/supportcenter] release master-7b53160cfb-	3 minutes ago
staging/releases	[staging/supportcenter] release master-7b53160cfb-	4 minutes ago
README.md	Refer to release-manager for directory details	8 months ago

README.md

k8s-cluster-config

@phennex

LUNAR

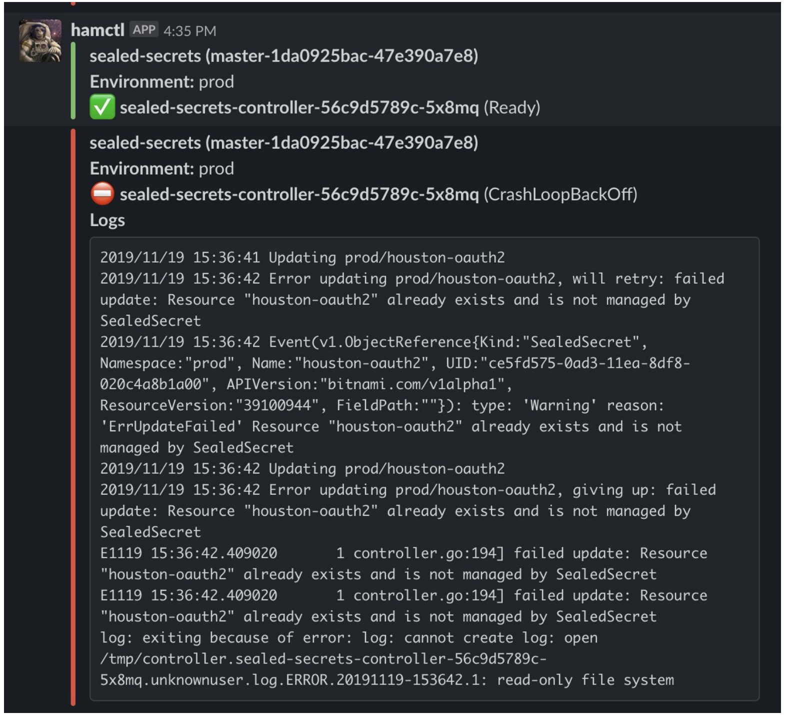
Dealing with multiple environments

release-manager

- 4 components
  - **release-server**
    - The server itself, this is where the magic happens
  - **release-daemon**
    - Kubernetes controller listens for updates on resources and reports back to release-server
  - **hamctl**
    - CLI for developers to control releases
  - **artifact**
    - Tool for generating metadata object; artifact.json
    - Handles slack communication from CI as well

Source: https://github.com/lunarway/release-manager

release-daemon



hamctl APP 4:35 PM

sealed-secrets (master-1da0925bac-47e390a7e8)

Environment: prod

✅ sealed-secrets-controller-56c9d5789c-5x8mq (Ready)

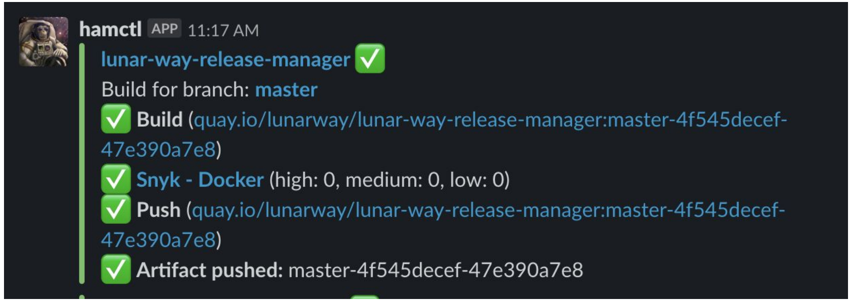
sealed-secrets (master-1da0925bac-47e390a7e8)

Environment: prod

⛔ sealed-secrets-controller-56c9d5789c-5x8mq (CrashLoopBackOff)

Logs

```
2019/11/19 15:36:41 Updating prod/houston-oauth2
2019/11/19 15:36:42 Error updating prod/houston-oauth2, will retry: failed
update: Resource "houston-oauth2" already exists and is not managed by
SealedSecret
2019/11/19 15:36:42 Event(v1.ObjectReference{Kind:"SealedSecret",
Namespace:"prod", Name:"houston-oauth2", UID:"ce5fd575-0ad3-11ea-8df8-
020c4a8b1a00", APIVersion:"bitnami.com/v1alpha1",
ResourceVersion:"39100944", FieldPath:""}): type: 'Warning' reason:
'ErrUpdateFailed' Resource "houston-oauth2" already exists and is not
managed by SealedSecret
2019/11/19 15:36:42 Updating prod/houston-oauth2
2019/11/19 15:36:42 Error updating prod/houston-oauth2, giving up: failed
update: Resource "houston-oauth2" already exists and is not managed by
SealedSecret
E1119 15:36:42.409020       1 controller.go:194] failed update: Resource
"houston-oauth2" already exists and is not managed by SealedSecret
E1119 15:36:42.409020       1 controller.go:194] failed update: Resource
"houston-oauth2" already exists and is not managed by SealedSecret
log: exiting because of error: log: cannot create log: open
/tmp/controller.sealed-secrets-controller-56c9d5789c-
5x8mq.unknownuser.log.ERROR.20191119-153642.1: read-only file system
```



hamctl APP 11:17 AM

lunar-way-release-manager ✅

Build for branch: master

✅ **Build** (quay.io/lunarway/lunar-way-release-manager:master-4f545decef-47e390a7e8)

✅ **Snyk - Docker** (high: 0, medium: 0, low: 0)

✅ **Push** (quay.io/lunarway/lunar-way-release-manager:master-4f545decef-47e390a7e8)

✅ **Artifact pushed:** master-4f545decef-47e390a7e8

Source: https://github.com/lunarway/release-manager

hamctl

```
# implicit order between envs: master -> dev -> staging -> prod
$ hamctl promote --env prod
```

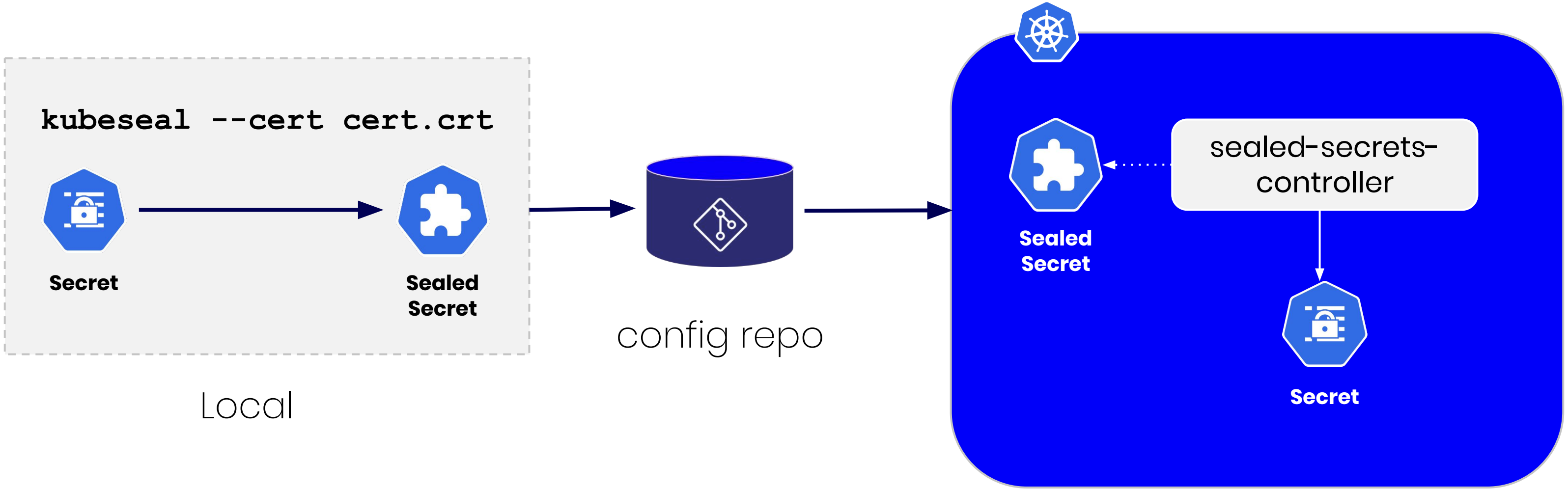
```
# choose which brand to deploy where
$ hamctl release --branch hotfix --env dev
```

```
# setup auto-release policy
$ hamctl policy apply auto-release --branch master
--env dev
```

Source: https://github.com/lunarway/release-manager

What about secrets?

sealed-secrets



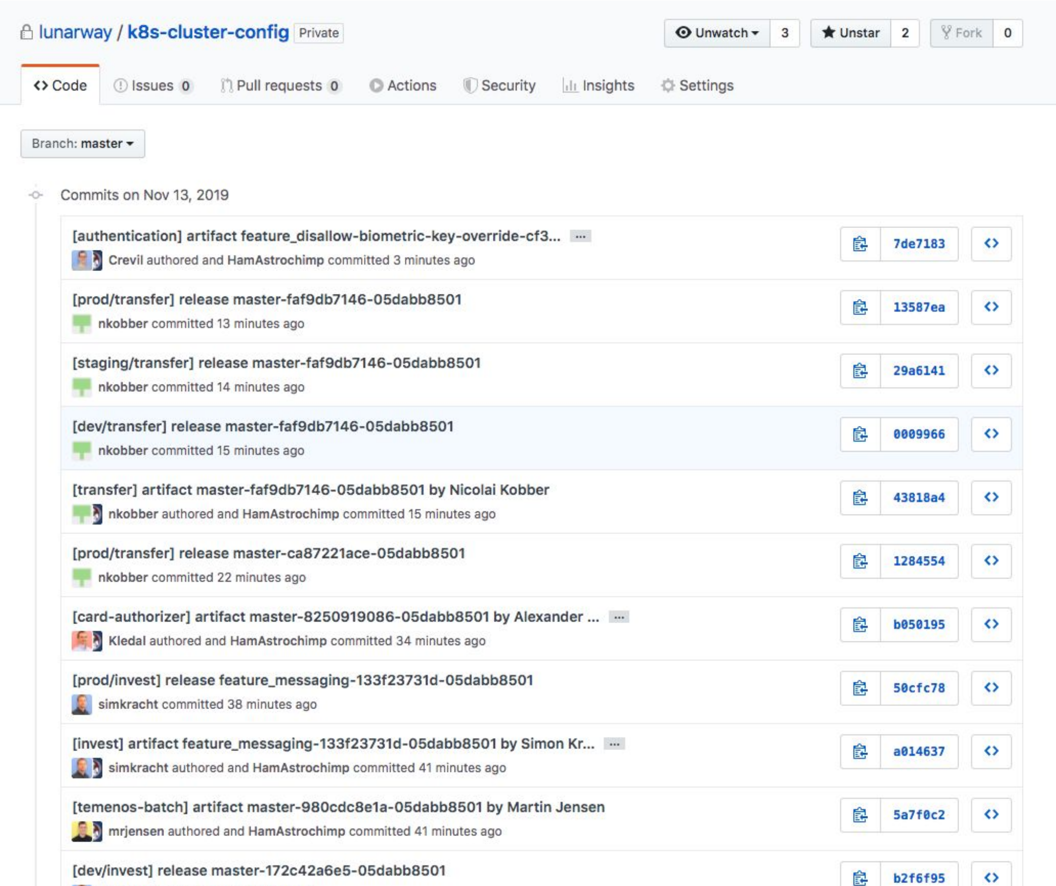
Source: https://github.com/bitnami-labs/sealed-secrets

@phennex

LUNAR

Why is this useful?

Audit



Commits on Nov 13, 2019

[authentication] artifact feature_disallow-biometric-key-override-cf3... …
Crevil authored and HamAstrochimp committed 3 minutes ago — 7de7183

[prod/transfer] release master-faf9db7146-05dabb8501
nkobber committed 13 minutes ago — 13587ea

[staging/transfer] release master-faf9db7146-05dabb8501
nkobber committed 14 minutes ago — 29a6141

[dev/transfer] release master-faf9db7146-05dabb8501
nkobber committed 15 minutes ago — 0009966

[transfer] artifact master-faf9db7146-05dabb8501 by Nicolai Kobber
nkobber authored and HamAstrochimp committed 15 minutes ago — 43818a4

[prod/transfer] release master-ca87221ace-05dabb8501
nkobber committed 22 minutes ago — 1284554

[card-authorizer] artifact master-8250919086-05dabb8501 by Alexander ... …
Kledal authored and HamAstrochimp committed 34 minutes ago — b050195

[prod/invest] release feature_messaging-133f23731d-05dabb8501
simkracht committed 38 minutes ago — 50cfc78

[invest] artifact feature_messaging-133f23731d-05dabb8501 by Simon Kr... …
simkracht authored and HamAstrochimp committed 41 minutes ago — a014637

[temenos-batch] artifact master-980cdc8e1a-05dabb8501 by Martin Jensen
mrjensen authored and HamAstrochimp committed 41 minutes ago — 5a7f0c2

[dev/invest] release master-172c42a6e5-05dabb8501 — b2f6f95

Forklifting the entire platform

@phennex

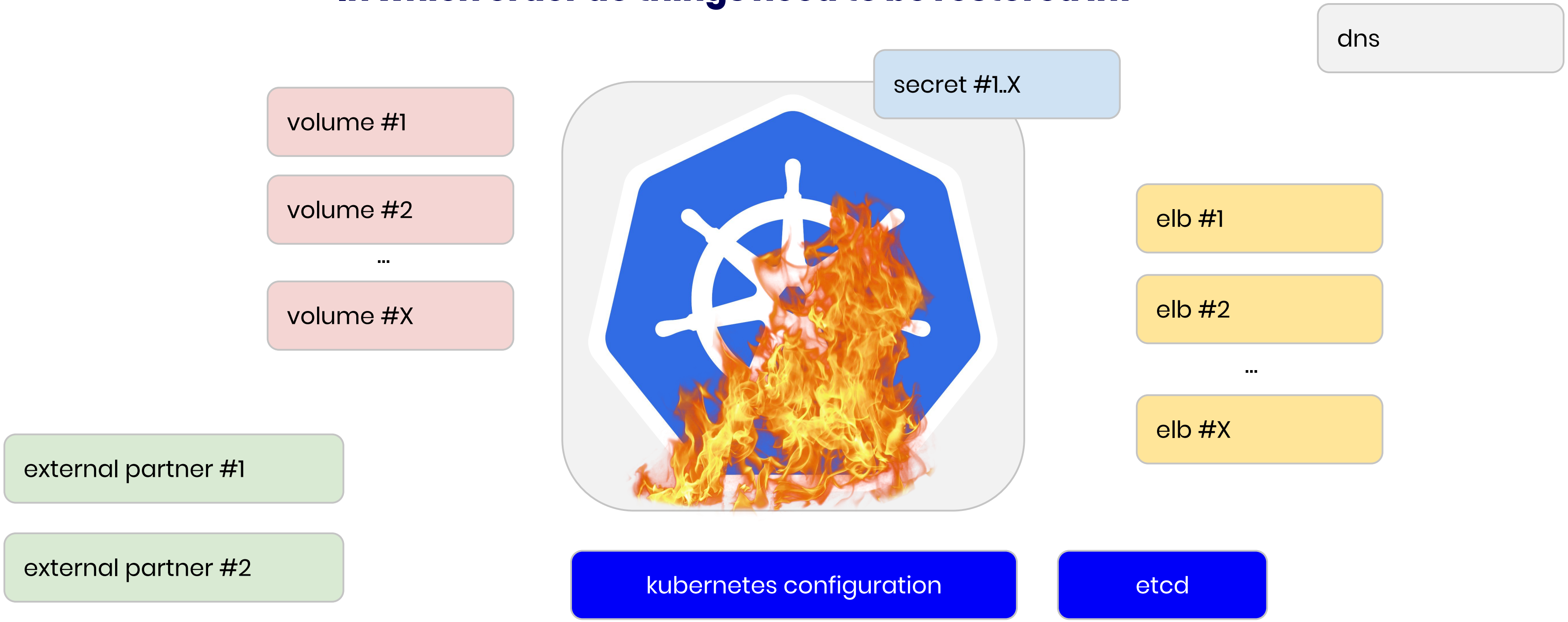
Disaster Recovery Failover Gamedays

Gameday

How fast can you recover from a cluster failure?

How to restore the environment?

In which order do things need to be restored in?



dns

secret #1..X

volume #1

volume #2

...

volume #X

elb #1

elb #2

...

elb #X

external partner #1

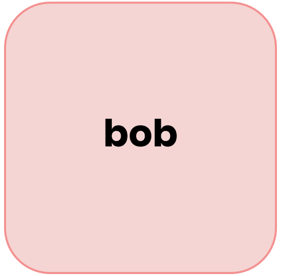
external partner #2

kubernetes configuration

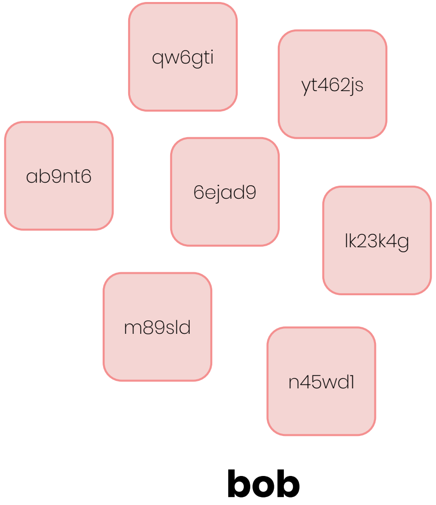
etcd

Clusters as ~~cattle~~ herds instead of pets

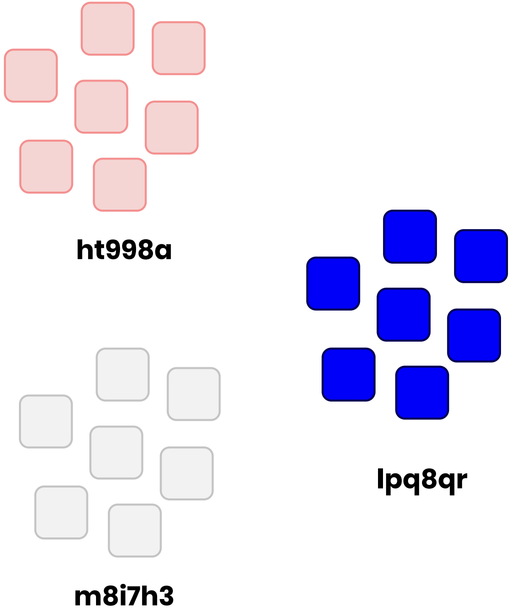
Clusters as ~~cattle~~ herds instead of pets



pet instance



pet clusters



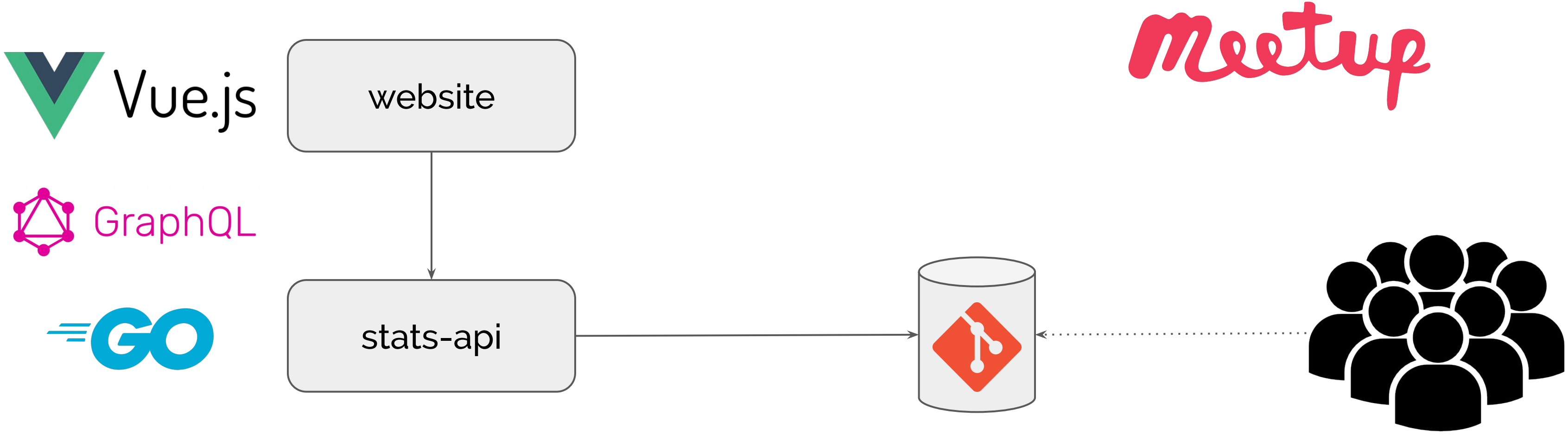
Chaos Engineering offers a dialogue with your system

CLOUD NATIVE
NORDICS

GitOps at Cloud Native Nordics

What? Don’t they just run a static website?

Overview



website

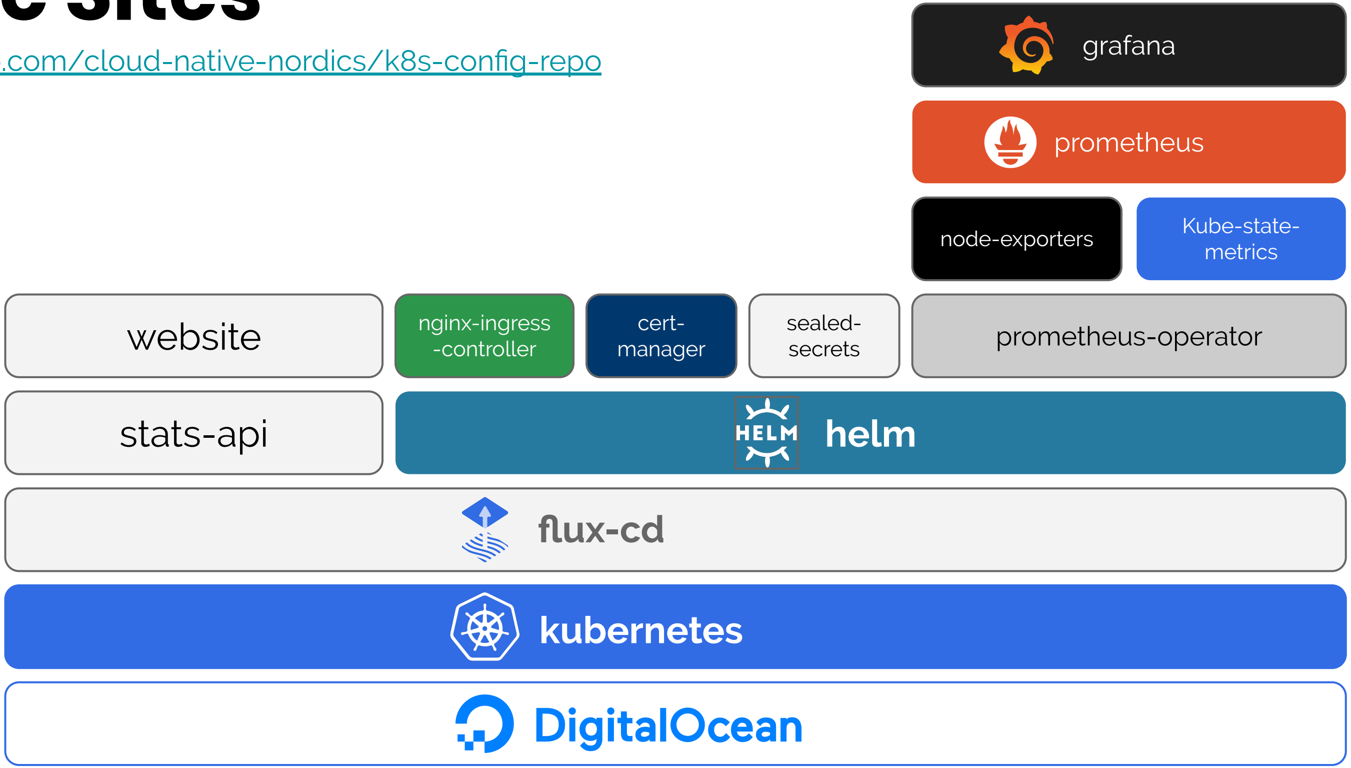
stats-api

cloud-native-nordics/meetups
config.json

Organizers

Kubernetes-based Infrastructure for Static Sites

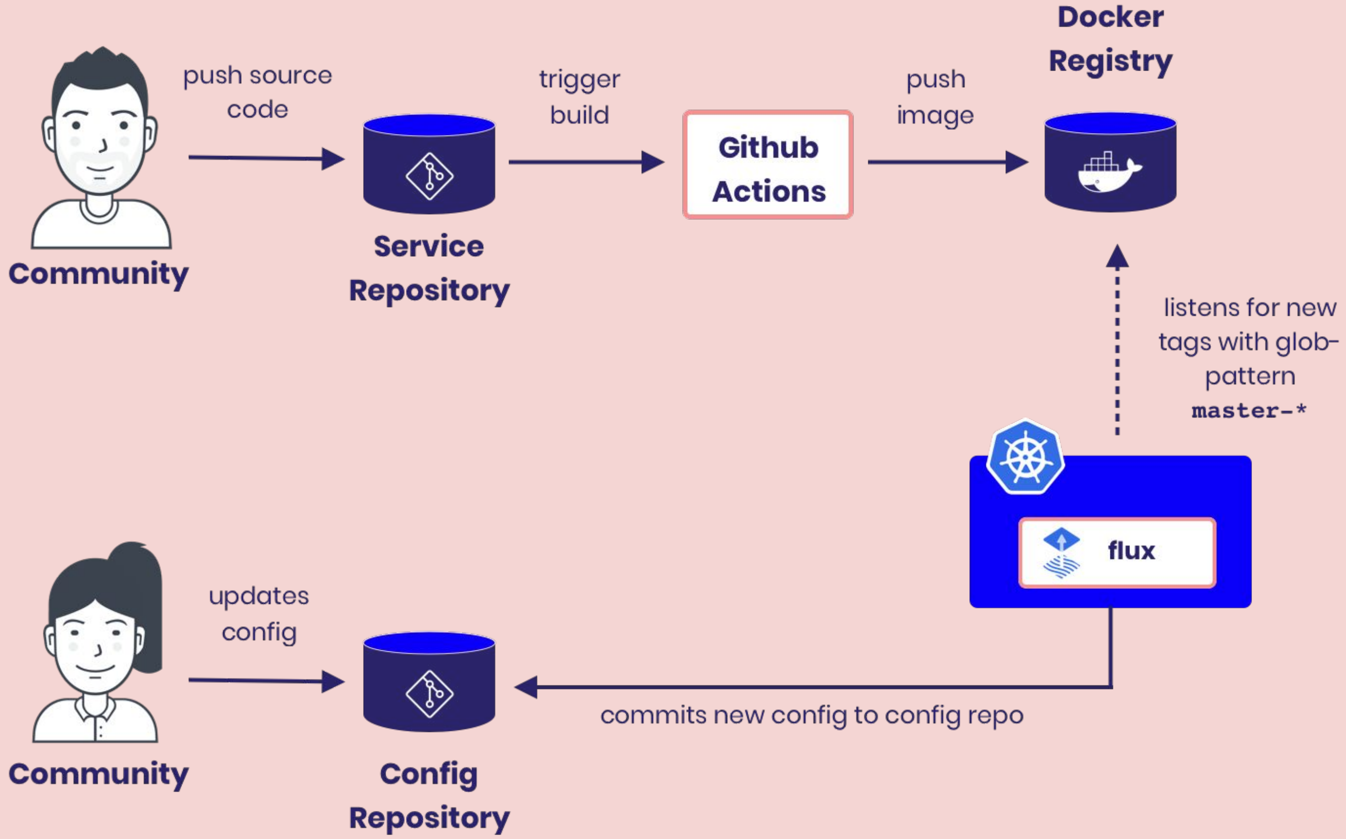
https://github.com/cloud-native-nordics/k8s-config-repo



@phennex

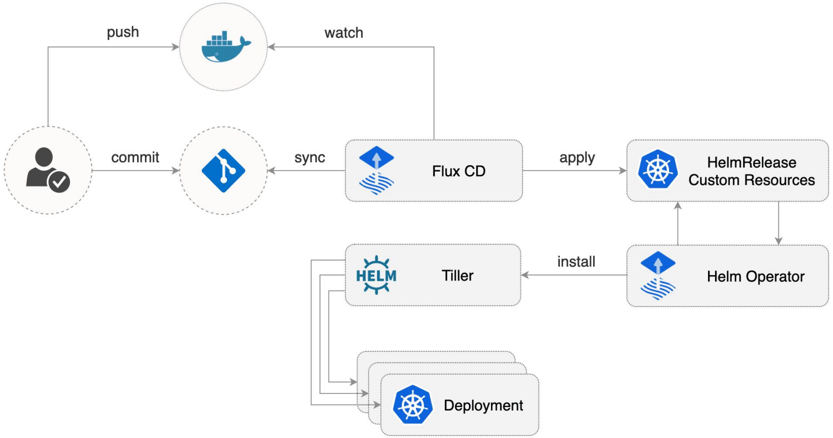
LUNAR

Our solution



HelmRelease

Custom Resource + Operator to support helm

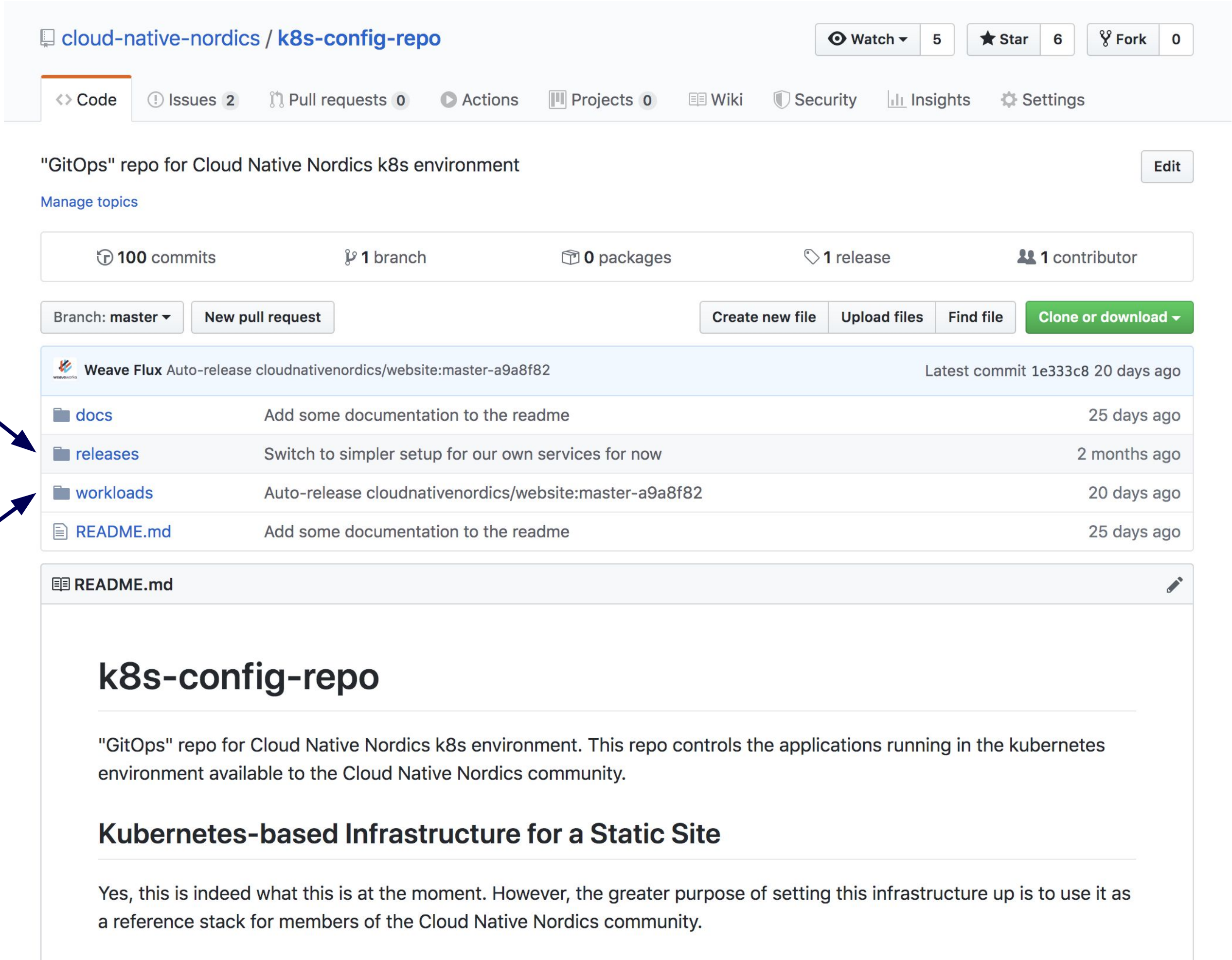


```yaml
apiVersion: flux.weave.works/v1beta1
kind: HelmRelease
metadata:
  name: nginx-ingress
  namespace: default
  annotations:
    flux.weave.works/automated: "false"
spec:
  releaseName: nginx-ingress
  chart:
    repository: https://kubernetes-charts...googleapis.com/
    name: nginx-ingress
    version: 1.22.1
  values:
    controller:
      publishService:
        enabled: true
```

@phennex

HelmRelease managed using Helm

Regular YAML kubernetes deployments, etc



"GitOps" repo for Cloud Native Nordics k8s environment

Weave Flux Auto-release cloudnativenordics/website:master-a9a8f82 Latest commit 1e333c8 20 days ago

docs	Add some documentation to the readme	25 days ago
releases	Switch to simpler setup for our own services for now	2 months ago
workloads	Auto-release cloudnativenordics/website:master-a9a8f82	20 days ago
README.md	Add some documentation to the readme	25 days ago

README.md

k8s-config-repo

"GitOps" repo for Cloud Native Nordics k8s environment. This repo controls the applications running in the kubernetes environment available to the Cloud Native Nordics community.

Kubernetes-based Infrastructure for a Static Site

Yes, this is indeed what this is at the moment. However, the greater purpose of setting this infrastructure up is to use it as a reference stack for members of the Cloud Native Nordics community.

@phennex

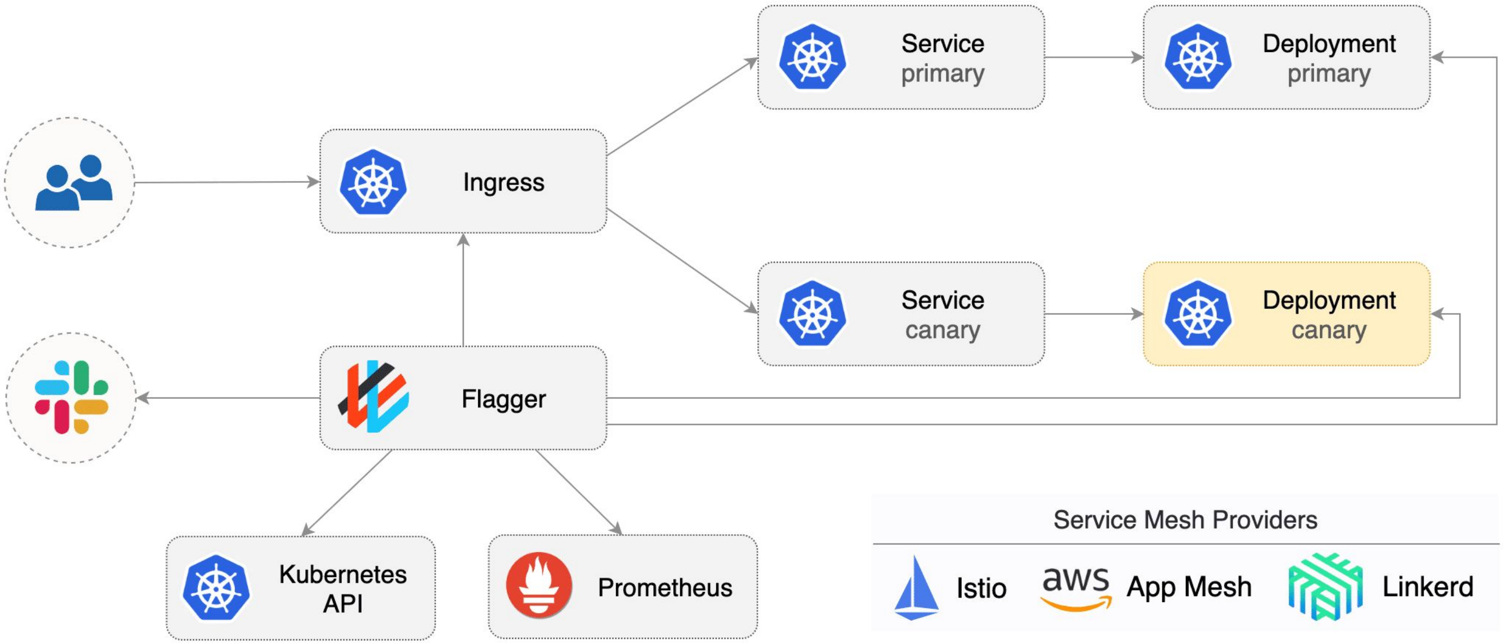
Progressive Delivery

Progressive Delivery

Progressive delivery is the process of pushing changes to a product iteratively, first to a small audience and then to increasingly larger audiences to maintain quality control (QC). The goal of progressive delivery is to improve delivery times for new product features and mitigate risk by controlling who is able to see them.

Source: https://searchitoperations.techtarget.com/definition/progressive-delivery

Progressive Delivery



Source: https://github.com/weaveworks/flagger

@phennex

Canary

Canary release is a technique to reduce the risk of introducing a new software version in production by slowly rolling out the change to a small subset of users before rolling it out to the entire infrastructure and making it available to everybody.

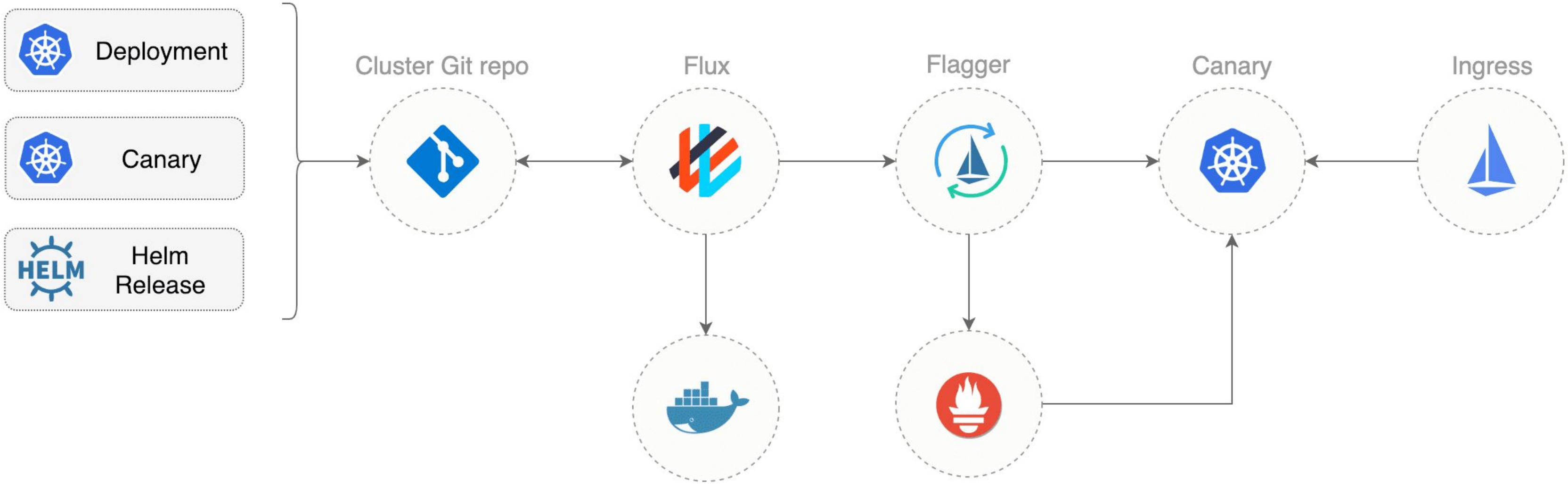
COAL MINE
WANT TO REALLY MESS WITH THESE GUYS? LAY DOWN AND TAKE A NAP.

©Creators Syndicate

@phennex

```
apiVersion: flagger.app/v1alpha3
kind: Canary
metadata:
  name: podinfo
spec:
  targetRef:
    kind: Deployment
    name: podinfo
  progressDeadlineSeconds: 60
  service:
    targetPort: 9898
  canaryAnalysis:
    interval: 1m
    threshold: 10
    maxWeight: 50
    stepWeight: 5
    metrics:
    - name: request-success-rate
      threshold: 99
      interval: 1m
    - name: request-duration
      threshold: 500
      interval: 30s
    webhooks:
      - name: load-test
        metadata:
            cmd: "hey -z 1m -q 10 -c 2
  http://podinfo.test:9898/"
```

Progressive Delivery



Source: https://github.com/weaveworks/flagger
@phennex

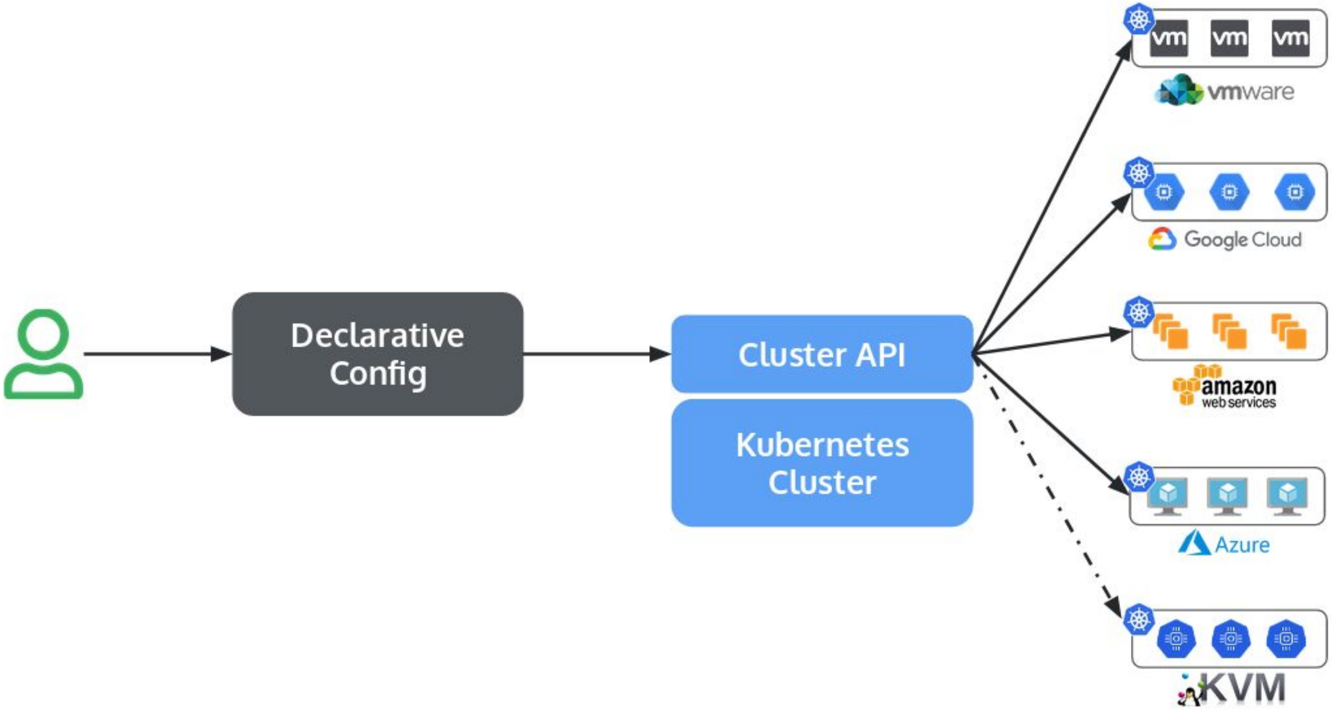
Everything in Git

Everything in Git

- CRDs
    - postgresql-controller
        - manage users, databases, hosts using Kubernetes objects
        - store them in git!
    - AWS Resources
- Infrastructure
    - cluster-api

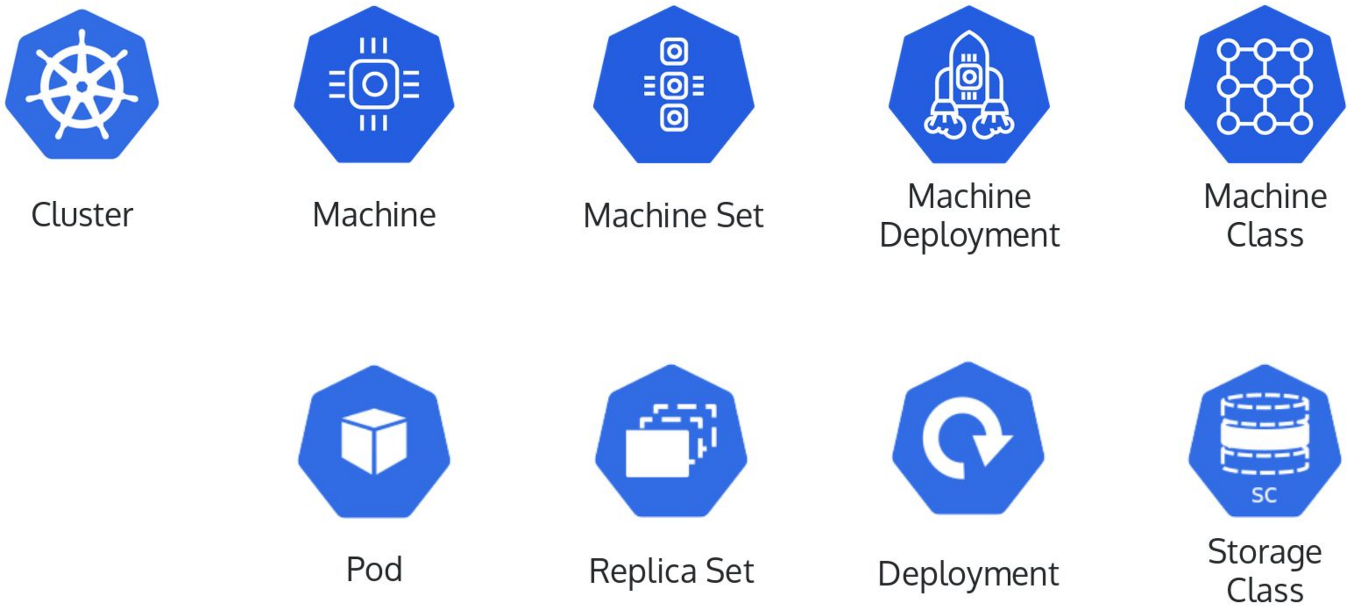
```yaml
apiVersion: lunar.bank/v1
kind: PostgreSQLUser
metadata:
 name: kni
spec:
 name: kni
 read:
  - host:
    value: some.host.com
   reason: "I am a developer"
 write:
  - host:
    valueFrom:
     configMapKeyRef:
      name: database
      key: db.host
   database:
    value: user
   schema:
    value: user
   reason: "Related to support ticket LW-1234"
   start: 2019-09-16T10:00:00Z
   end: 2019-09-16T14:00:00Z
```

cluster-api



Source: https://itnext.io/kubernetes-cluster-creation-on-baremetal-host-using-cluster-api-1c2373230a17

cluster-api



Source: https://itnext.io/kubernetes-cluster-creation-on-baremetal-host-using-cluster-api-1c2373230a17

cluster-api

Use kubernetes to manage kubernetes.

Again....

Using Git!

```yaml
apiVersion: "cluster.k8s.io/v1alpha1"
kind: MachineDeployment
metadata:
  name: nodes
  namespace: kube-system
spec:
  replicas: 5
  selector:
    matchLabels:
      foo: bar
  template:
    metadata:
      labels:
        foo: bar
    spec:
      providerSpec:
        value:
          cloudProvider: "aws"
          cloudProviderSpec:
            region: "eu-central-1"
            availabilityZone: "eu-central-1a"
            vpcId: "vpc-819f62e9"
            subnetId: "subnet-2bff4f43"
            instanceType: "t2.micro"
            instanceProfile: "kubernetes-v1"
            diskSize: 50
            ..
          operatingSystem: "coreos"
          operatingSystemSpec:
            disableAutoUpdate: true
```

Things change…
CI/CD should too

Things change…
Banks should too

@phennex

Questions?

Contact:

Twitter: @phennex
Github: kaspernissen
E-mail: kni@lunarway.com

LUNAR®

We are hiring!

jobs.lunarway.com