

GitOps with Flux v2



**Kubernetes Melbourne
meetup**

August 2023

Yann Vigara



About me

French

20+ years in Tech

Entrepreneur



<https://linkedin.com/in/yvigara>
yann@celest.io



Coffee, Cheese, Wine, Beer and Tech



What is **GitOps**?

GitOps is a set of practices and tools for managing infrastructure and applications so that the whole system is described declaratively and version controlled.

An automated process ensures that the deployed environment matches the state specified in a repository.

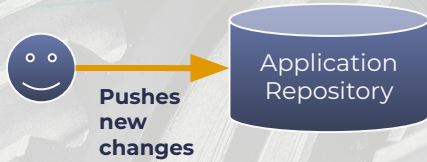


A high-angle, top-down photograph of a person's hands and arms pulling a thick, white, frayed rope across a weathered wooden deck. The person is wearing a dark jacket and dark pants. A green plastic pulley is mounted on a metal bracket on the deck. To the right, a large coil of the same rope is visible. The text 'GitOps promotes a pull-based deployment model' is overlaid in the upper right corner, with 'pull-based' highlighted in a red box.

GitOps promotes a
pull-based
deployment model



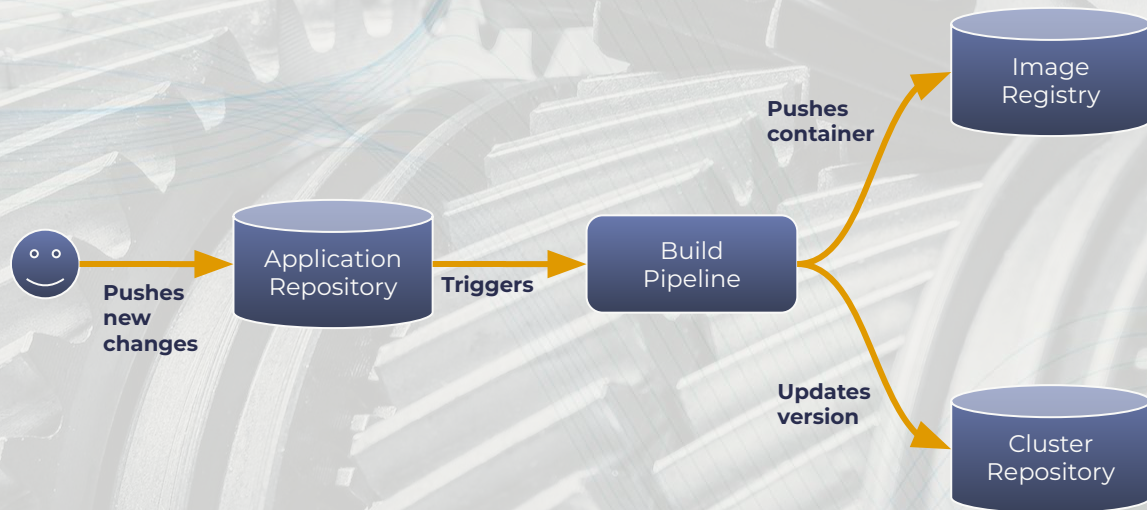
Push-based deployment



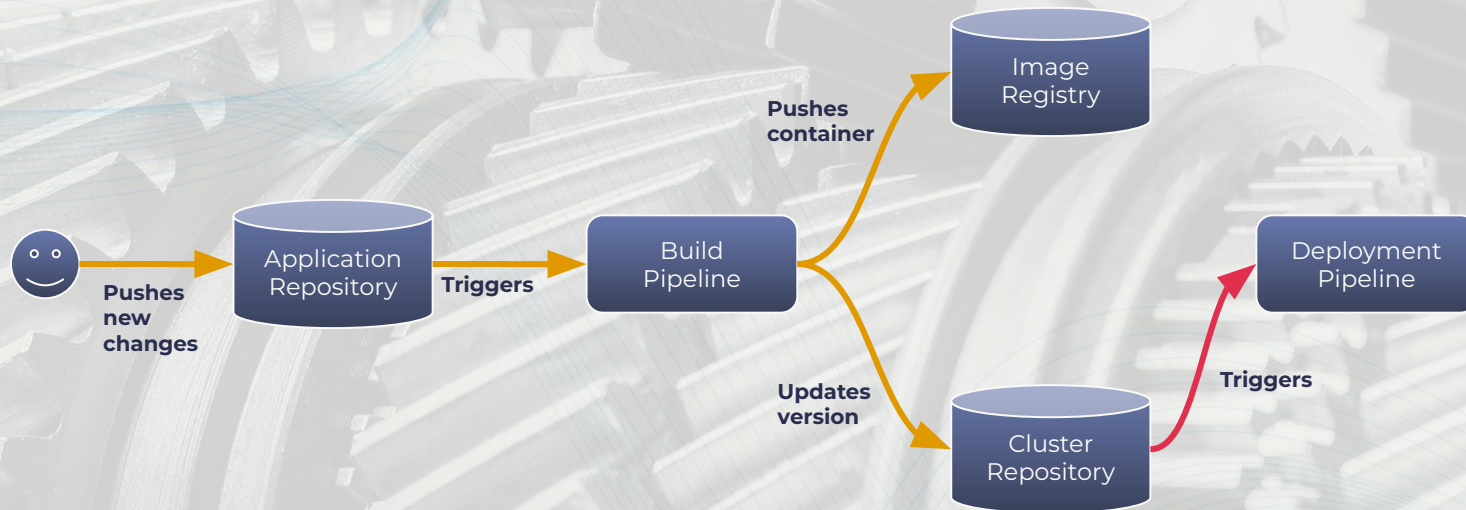
Push-based deployment



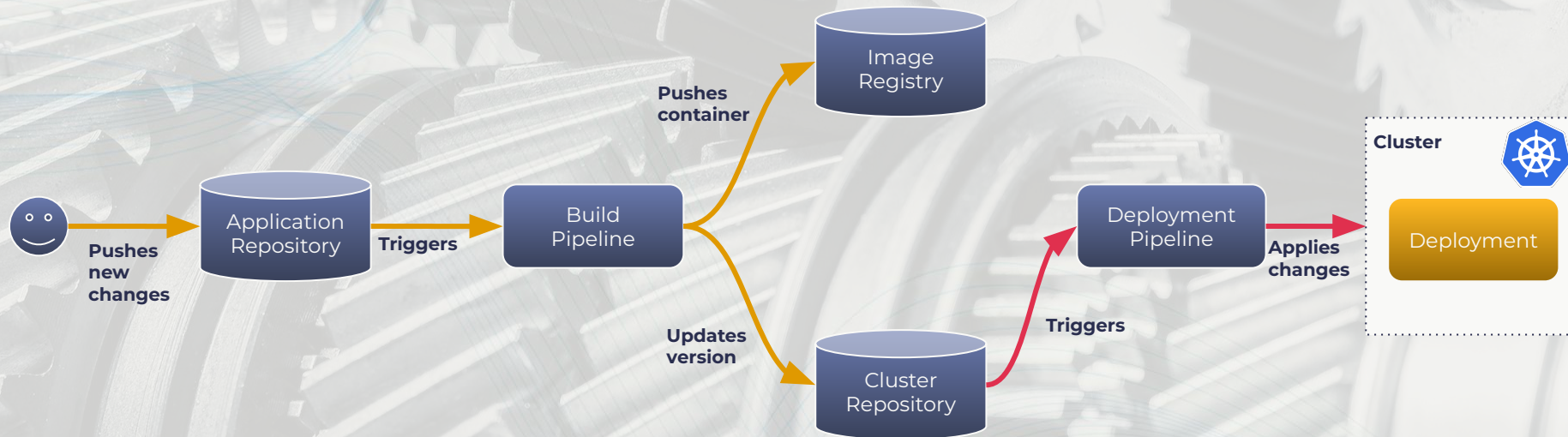
Push-based deployment



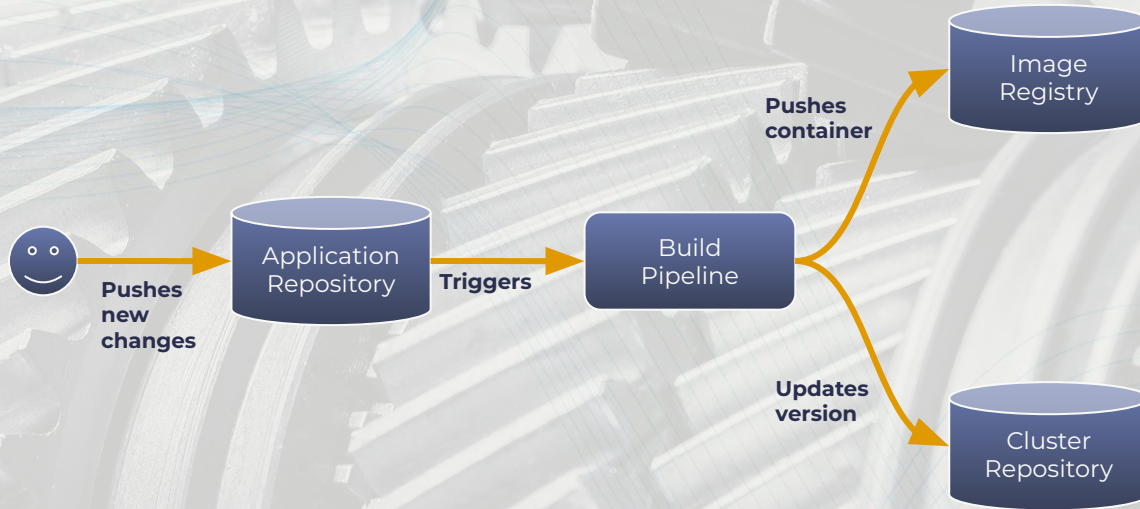
Push-based deployment



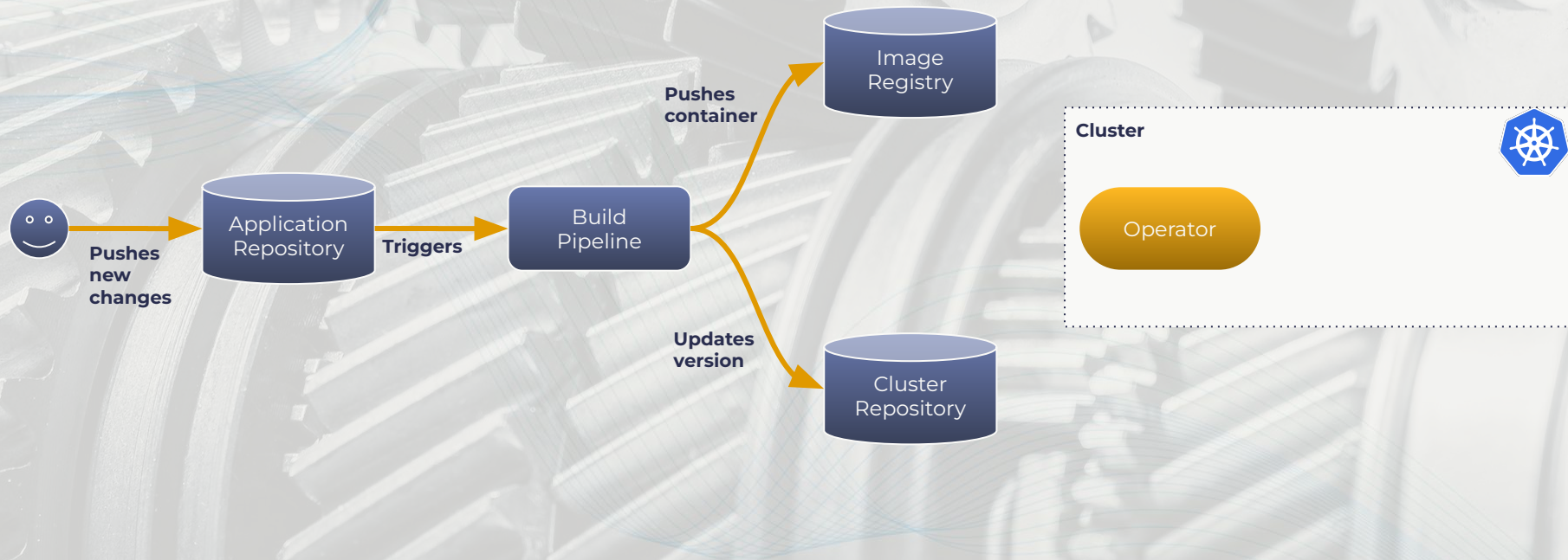
Push-based deployment



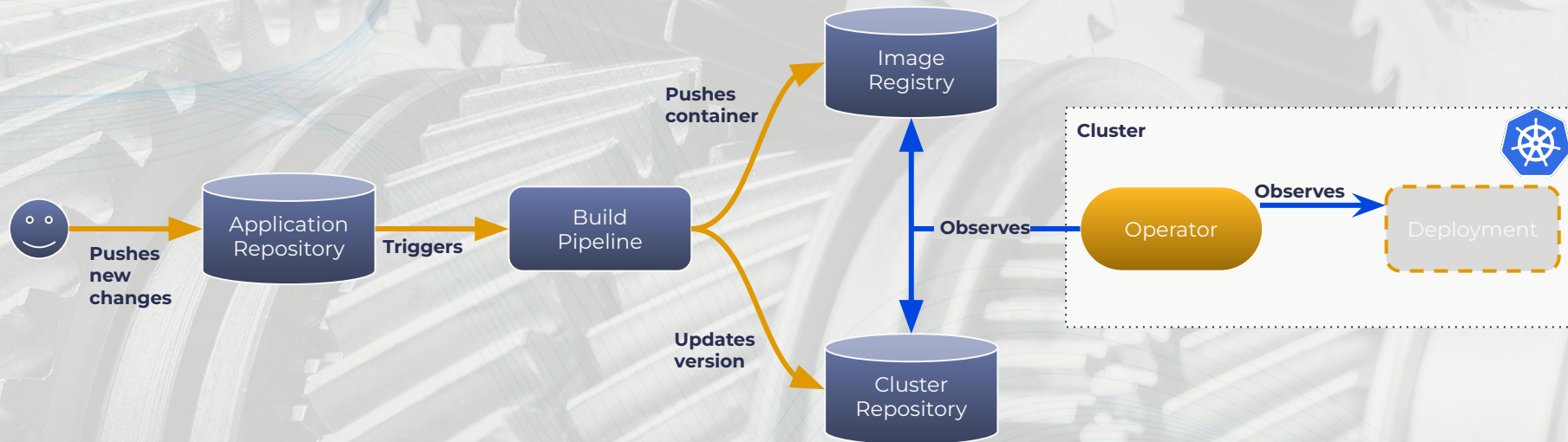
Pull-based deployment



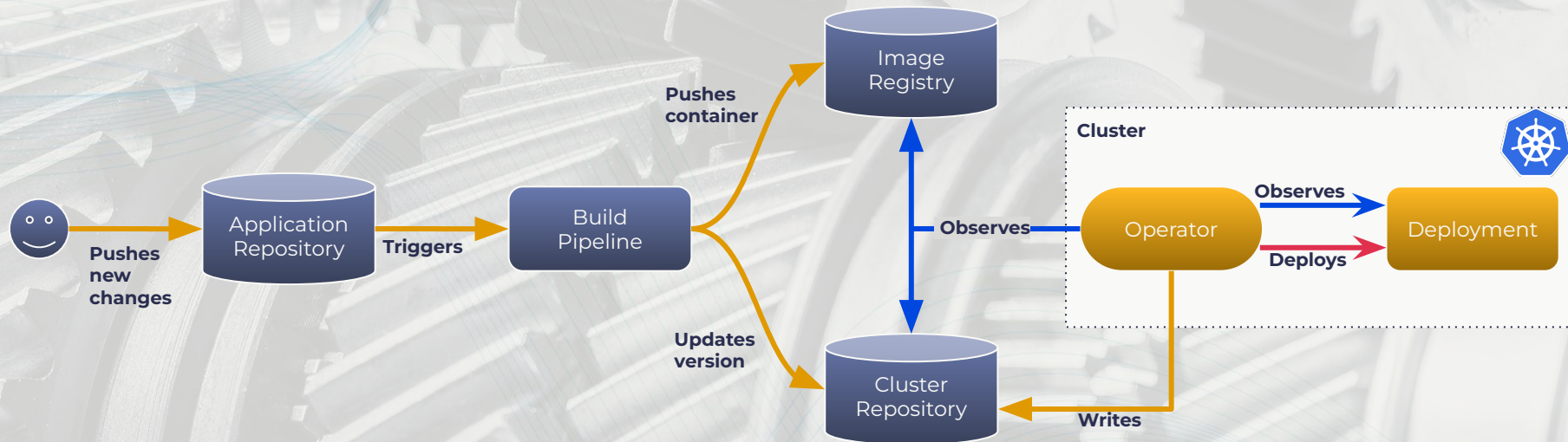
Pull-based deployment



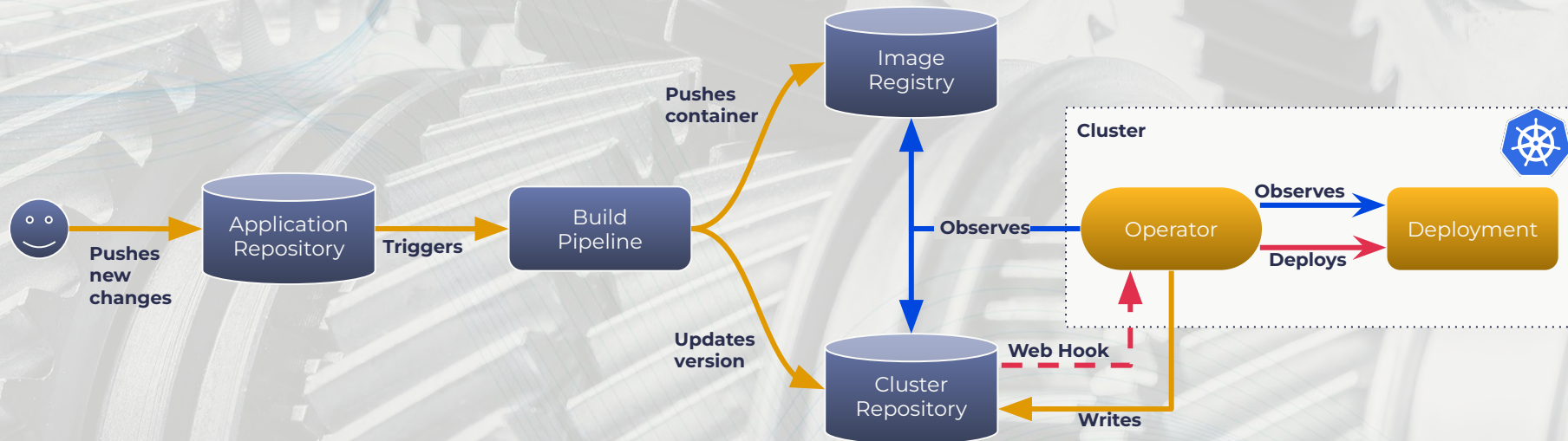
Pull-based deployment



Pull-based deployment



Pull-based deployment



Benefits of GitOps

Same tools developers use

Less drifts in configuration

Increased visibility of changes (Pull Requests)

History and audit logs of changes

Manage multiple environment from the same source



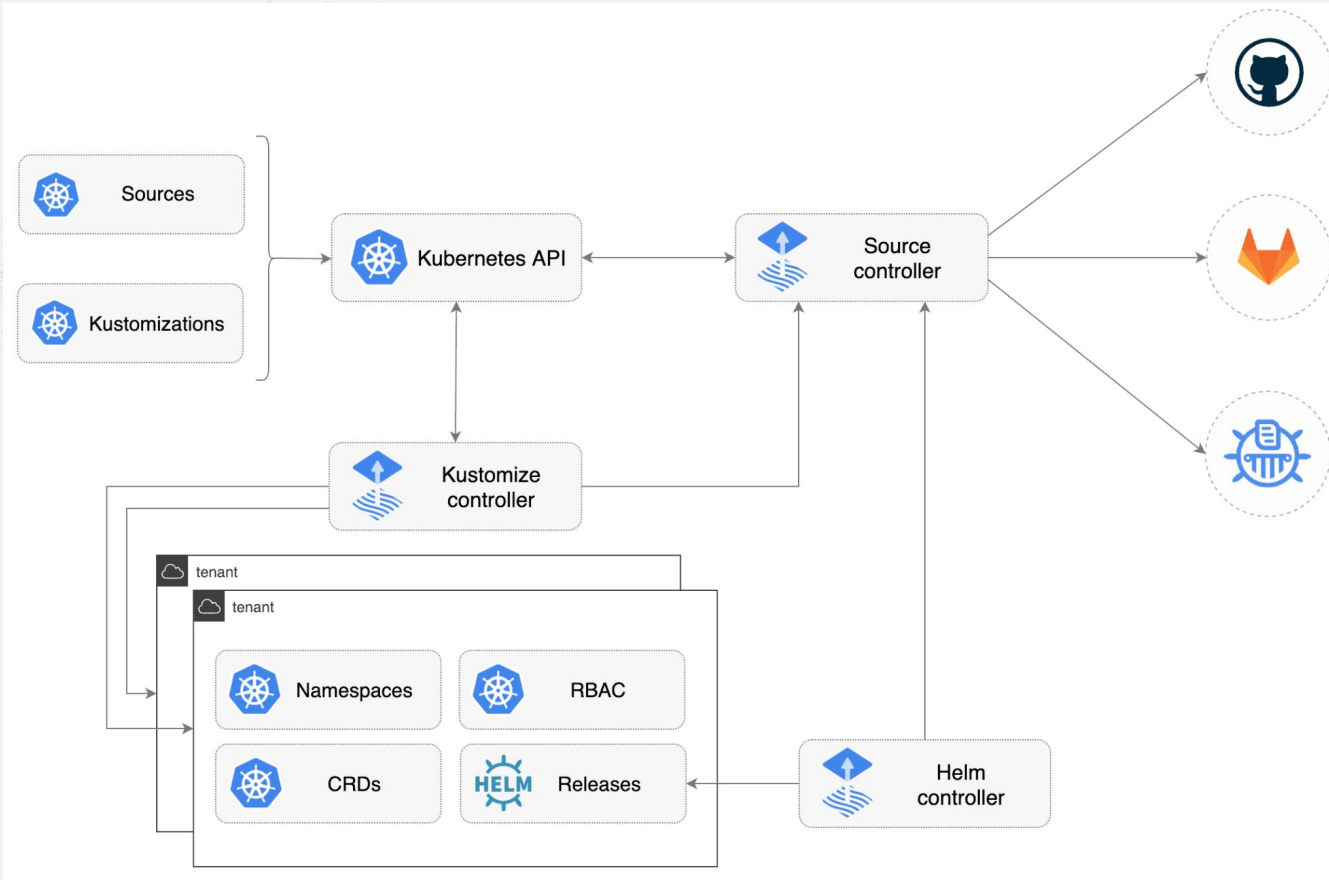
What is **Flux**?

A CNCF project providing a set of specialised tools and Kubernetes controllers which:

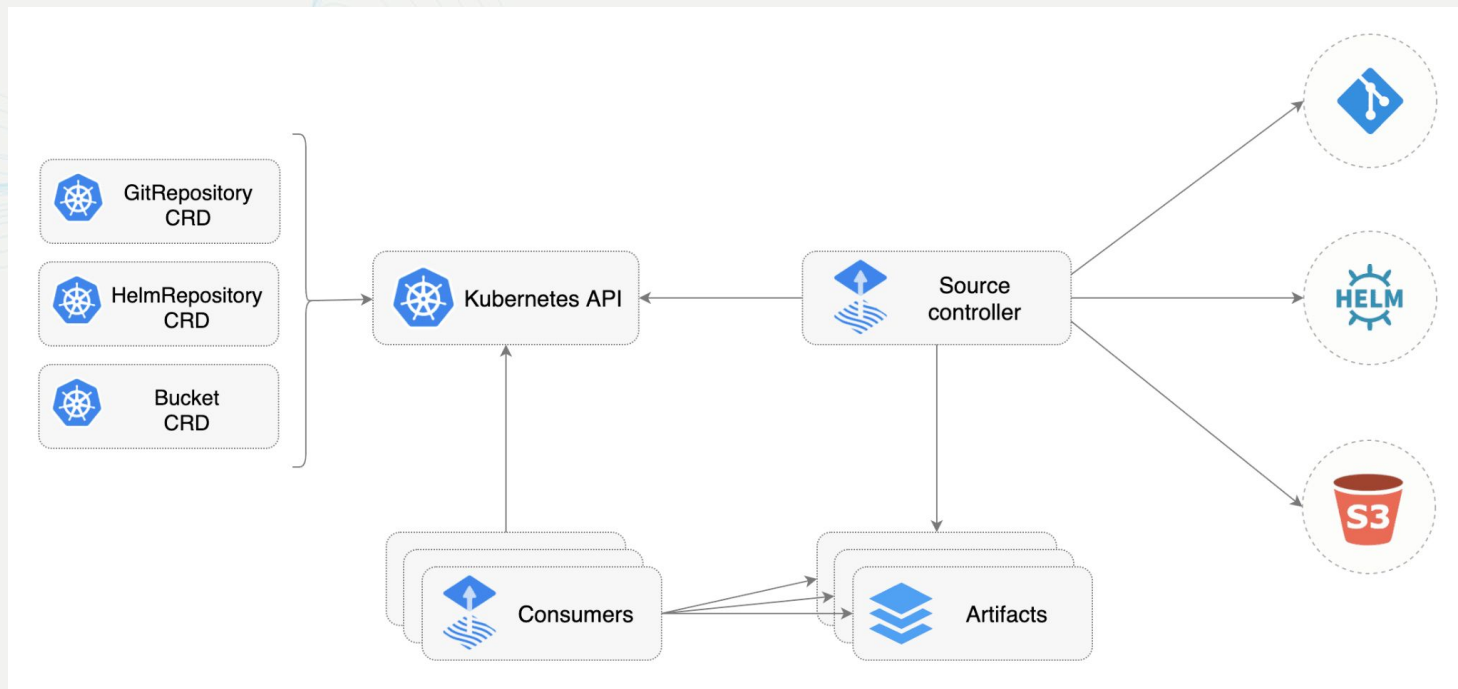
- Pull changes from Sources
- Reconcile the changes in the cluster
- Notify of changes



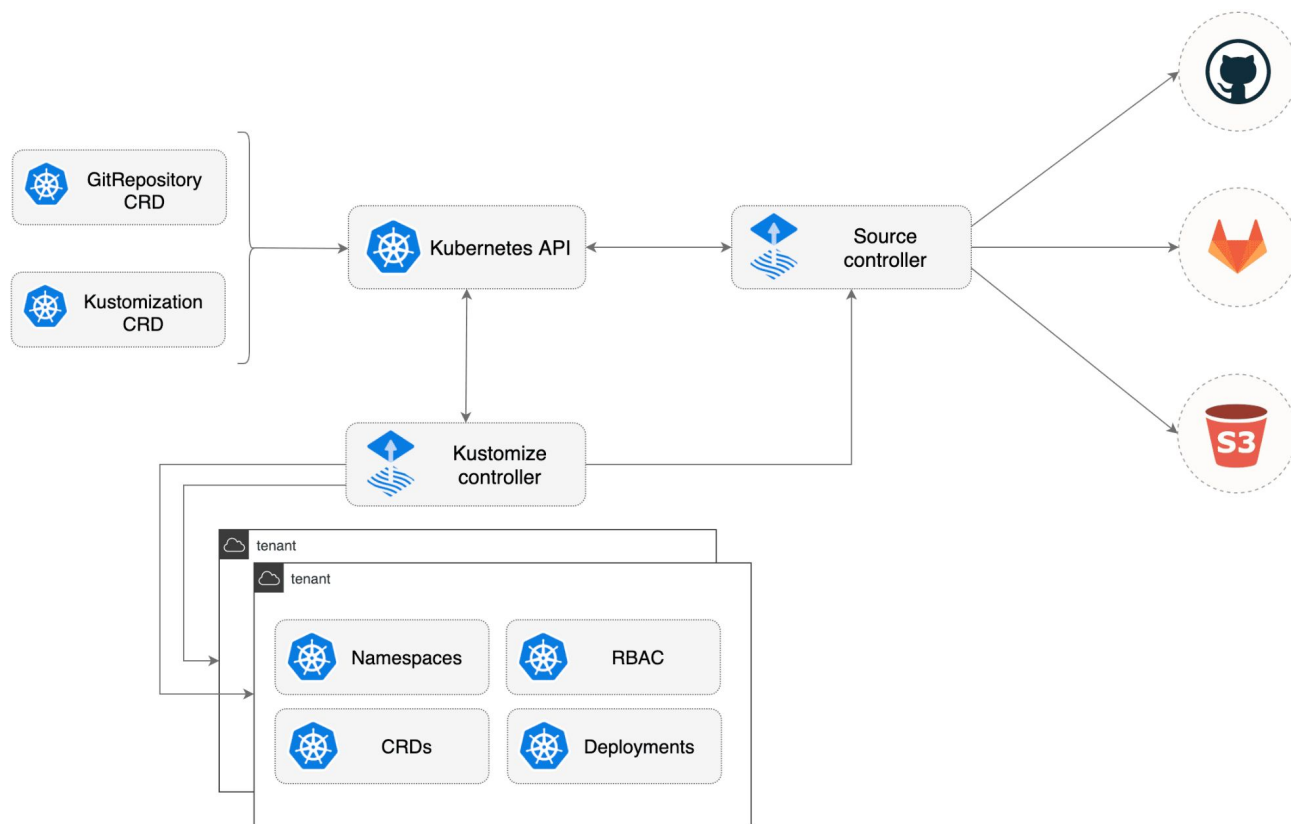
Flux Components



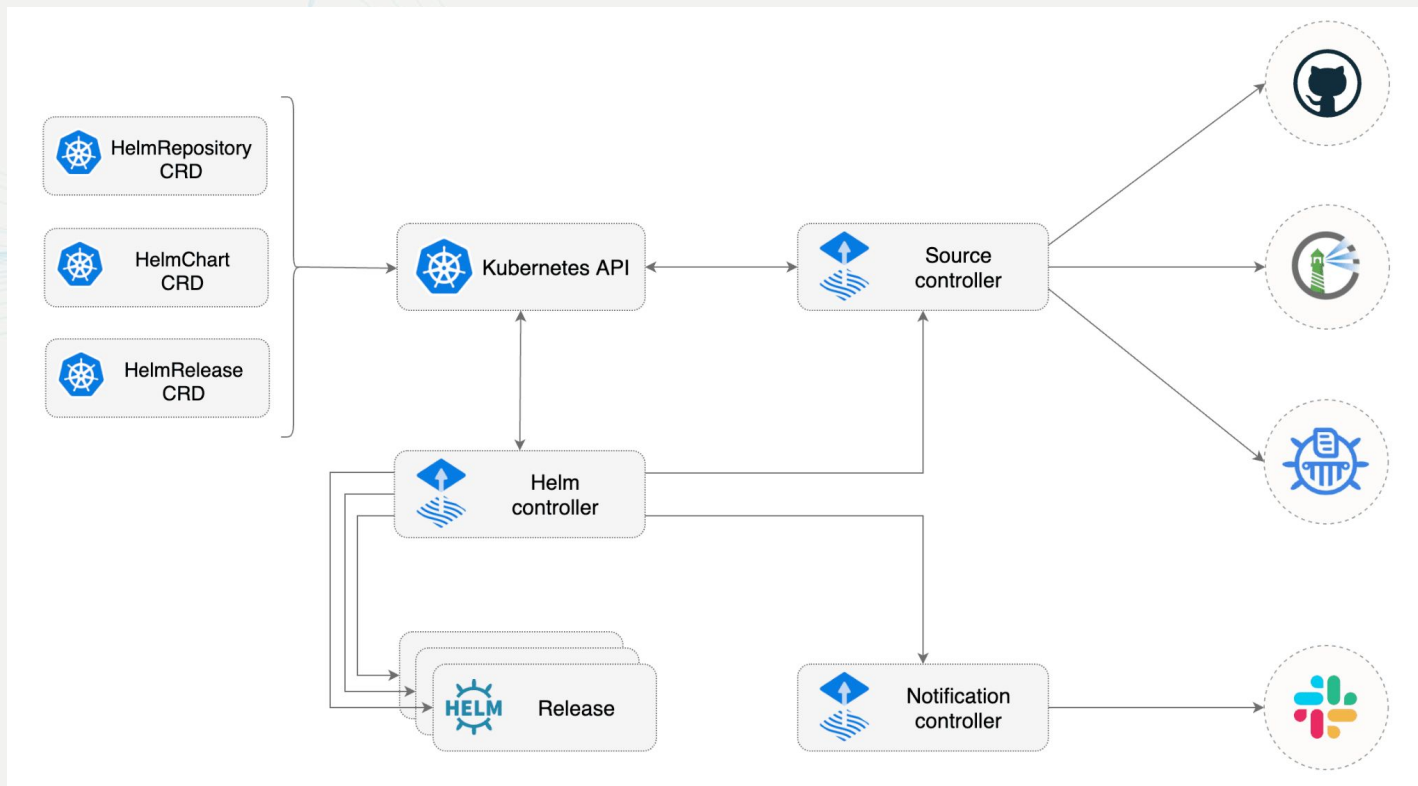
Flux Components | Source Controller



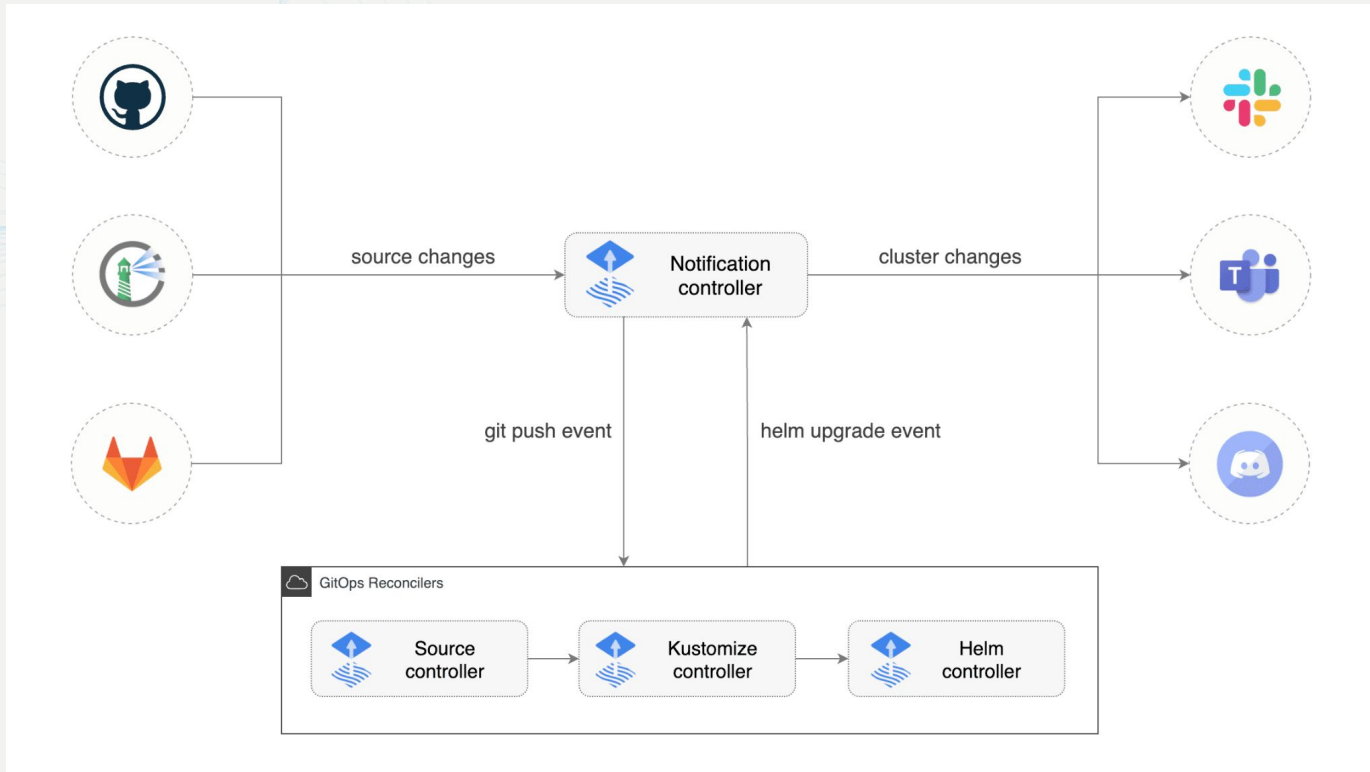
Flux Components | **Kustomize Controller**



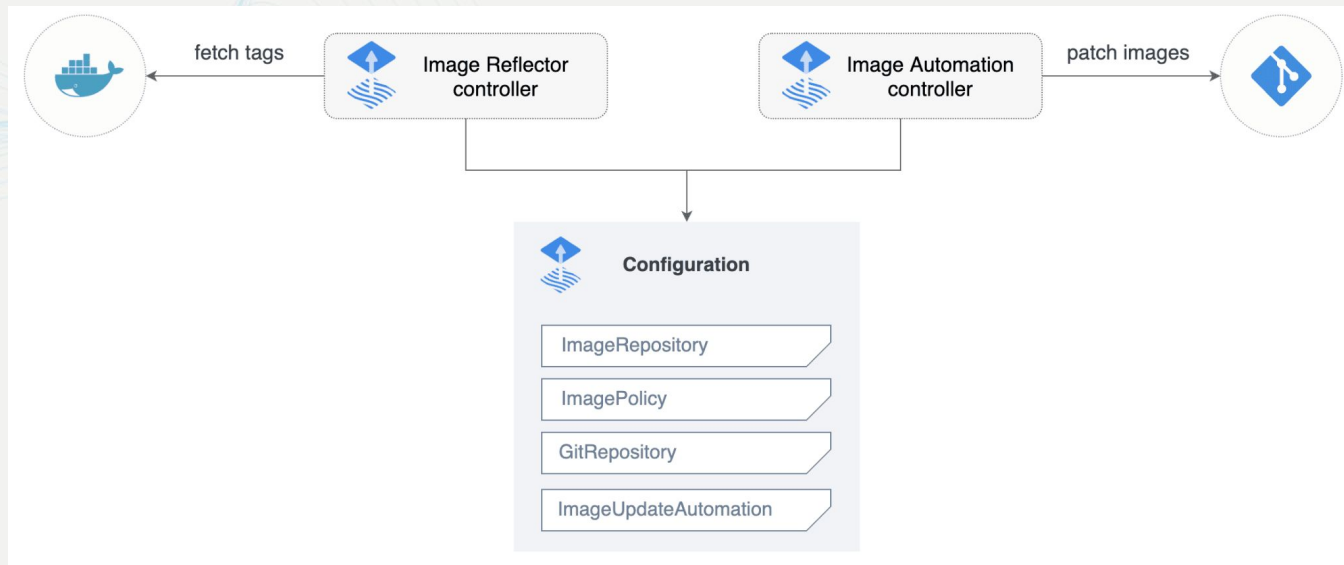
Flux Components | Helm Controller



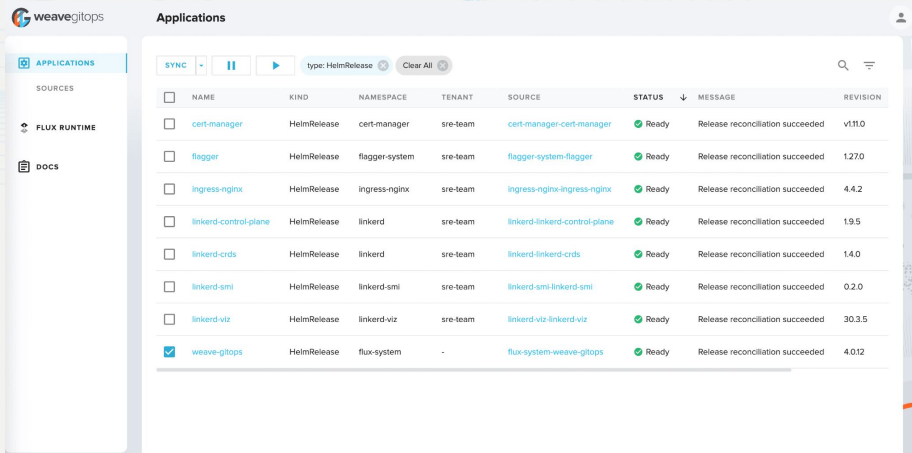
Flux Components | Notification Controller



Flux Components | Image Controller

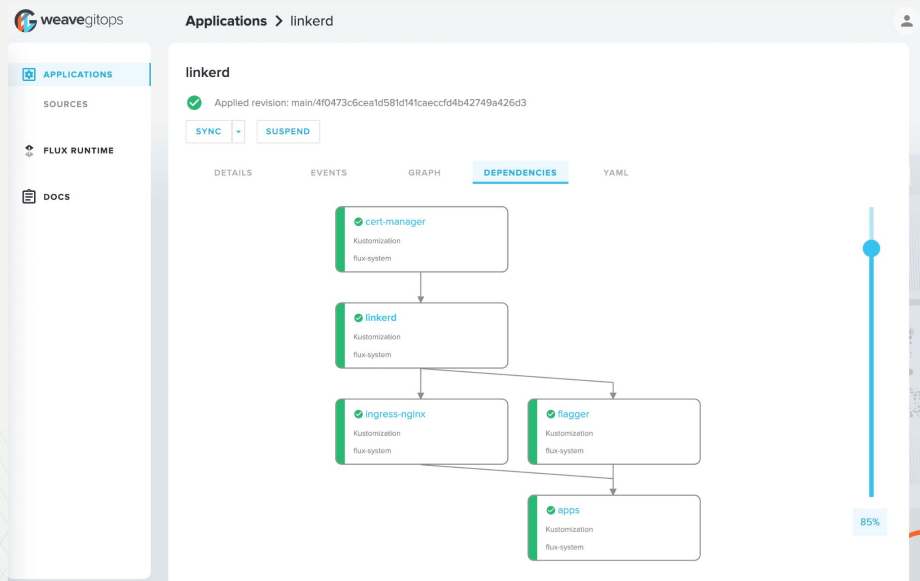


Flux Components | Web UI



The screenshot shows the 'Applications' page in the Weave GitOps web UI. The left sidebar contains navigation links for 'APPLICATIONS', 'SOURCES', 'FLUX RUNTIME', and 'DOCS'. The main area displays a table of installed Helm releases, filtered by 'type: HelmRelease'. The table includes columns for NAME, KIND, NAMESPACE, TENANT, SOURCE, STATUS, MESSAGE, and REVISION. The 'weave-gitops' release is highlighted with a blue checkmark in the first column.

| <input type="checkbox"/> | NAME | KIND | NAMESPACE | TENANT | SOURCE | STATUS | MESSAGE | REVISION |
|-------------------------------------|-----------------------|-------------|----------------|----------|-------------------------------|--------|----------------------------------|----------|
| <input type="checkbox"/> | cert-manager | HelmRelease | cert-manager | sre-team | cert-manager-cert-manager | Ready | Release reconciliation succeeded | v1.11.0 |
| <input type="checkbox"/> | flagger | HelmRelease | flagger-system | sre-team | flagger-system-flagger | Ready | Release reconciliation succeeded | 1.27.0 |
| <input type="checkbox"/> | ingress-nginx | HelmRelease | ingress-nginx | sre-team | ingress-nginx-ingress-nginx | Ready | Release reconciliation succeeded | 4.4.2 |
| <input type="checkbox"/> | linkerd-control-plane | HelmRelease | linkerd | sre-team | linkerd-linkerd-control-plane | Ready | Release reconciliation succeeded | 19.5 |
| <input type="checkbox"/> | linkerd-crdts | HelmRelease | linkerd | sre-team | linkerd-linkerd-crdts | Ready | Release reconciliation succeeded | 14.0 |
| <input type="checkbox"/> | linkerd-smi | HelmRelease | linkerd-smi | sre-team | linkerd-smi-linkerd-smi | Ready | Release reconciliation succeeded | 0.2.0 |
| <input type="checkbox"/> | linkerd-viz | HelmRelease | linkerd-viz | sre-team | linkerd-viz-linkerd-viz | Ready | Release reconciliation succeeded | 30.3.5 |
| <input checked="" type="checkbox"/> | weave-gitops | HelmRelease | flux-system | - | flux-system-weave-gitops | Ready | Release reconciliation succeeded | 4.0.12 |



The screenshot shows the 'linkerd' application details page in the Weave GitOps web UI. The left sidebar contains navigation links for 'APPLICATIONS', 'SOURCES', 'FLUX RUNTIME', and 'DOCS'. The main area displays the 'linkerd' application details, including the applied revision and a dependency graph. The dependency graph shows a flow from 'cert-manager' to 'linkerd', which then branches to 'ingress-nginx' and 'flagger', both of which point to 'apps'.

linkerd

Applied revision: main/4f0473c6cea1d581d141caeccfd4b42749a426d3

SYNC SUSPEND

DETAILS EVENTS GRAPH DEPENDENCIES YAML

```
graph TD; cert-manager[cert-manager] --> linkerd[linkerd]; linkerd --> ingress-nginx[ingress-nginx]; linkerd --> flagger[flagger]; ingress-nginx --> apps[apps]; flagger --> apps;
```



Bootstrapping Flux

Manual process:

- Deploy the Controllers in the cluster
- Create a GitRepository resource pointing at a Git repo
- Create a Secret resource to access the Git repo
- Committing the resource in the Git repo

Or

Install FluxCD Cli and run the **flux bootstrap** command

Or

Use the Flux Terraform Provider



A photograph of three men in a server room, seen from behind, performing a celebratory gesture with their hands raised and palms facing each other. They are standing in front of several tall server racks filled with electronic equipment and blue cables. The man on the left is wearing a dark blue shirt, the middle man a black shirt, and the right man a light blue shirt. A red banner with the text "Demo Time" is overlaid on the left side of the image.

Demo Time



Summary

- Bootstrapped Flux on a fresh Kubernetes Cluster
- Deployed a simple application: Podinfo
- Upgraded Podinfo
- Configured an application using Variable substitution
- Enabled Helm Chart drift detection
- Deployed Cert-Manager and used components dependency
- Deployed an Ingress controller (Traefik) and exposed Podinfo
- Bootstrapped Flux on a cluster using the Flux Terraform Provider
- Deployed everything in one commit



Next Steps

- Flux Get Started guide: <https://fluxcd.io/flux/get-started/>
- Home labs K8s + Flux projects: <https://nanne.dev/k8s-at-home-search/>
- Integrate Renovate for automatic version update
- Configure a GitHub WebHook receiver
- Configure Notifications to send alerts to Slack and update Git Commit Status
- Deploy **tf-controller** to apply Terraform scripts using Flux
- Deploy Weave GitOps WebUI
- Collect metrics and deploy Grafana dashboard
- Integrate **flagger** with Ingress controller for progressive deployment



Questions



<https://linkedin.com/in/yvigara>
yann@celest.io

