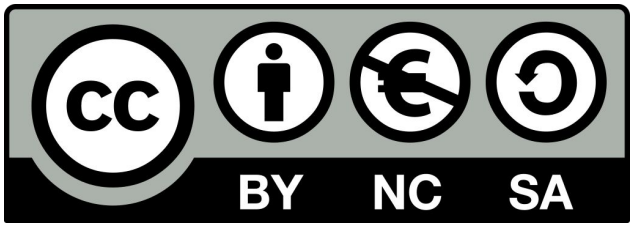
Hands-on GitOps

Weaveworks – https://weave.works – @weaveworks
Lakeside Hackfest – September 2019

Brice Fernandes – brice@weave.works – @fractallambda



Welcome and Introduction

I’m Brice

I work for Weaveworks as a **customer success engineer**

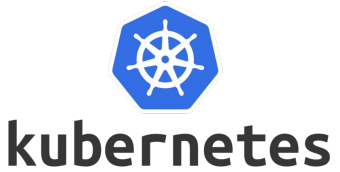
You can find **Weaveworks** at https://www.weave.works or @weaveworks

The team at Weaveworks is behind the GitOps model

You can find me online at @fractallambda

About Weaveworks

- Building cloud-native OSS since 2014
  (Weave Net, Moby, Kubernetes, Prometheus)
- Founding member of CNCF
- Alexis Richardson (Weaveworks CEO) is former chair and member of the CNCF Technical Oversight Committee
- Weave Cloud runs on Kubernetes since 2015



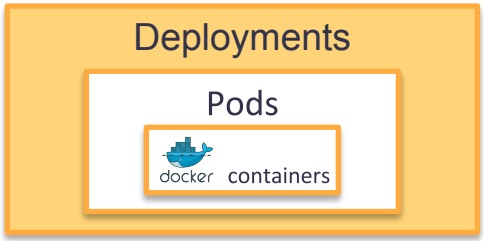
Agenda

14:00	**Welcome & introduction**
14:15	**Getting started with your environment**
14:30	**What is GitOps**
15:00	**Deploying a service the GitOps way**
15:15	**Break (30 minutes)**
15:45	**GitOps in practice**
16:45	**Review and recap**
17:00	**End of session**

Some assumptions

- ➔ You can use the command line.
- ➔ You can use Git.
- ➔ You know what Kubernetes Pods, Deployment, and Services are.
- ➔ You have a modern web browser.

Kubernetes need to know



Containers - Run Docker images, an immutable copy of your application code and all code dependencies in an isolated environment.

Pods - A set of containers, co-scheduled on one machine. Ephemeral. Has unique IP. Has labels.

Deployment - Ensures a certain number of replicas of a pod are running across the cluster.

Service - Gets virtual IP, mapped to endpoints via labels. Named in DNS.

Namespace - Resource names are scoped to a Namespace. Policy boundary.

WIFI: DT-HackFest
Pass: DT#2019LSP

Slides available at:

tinyurl.com/lakeside-gitops-2019

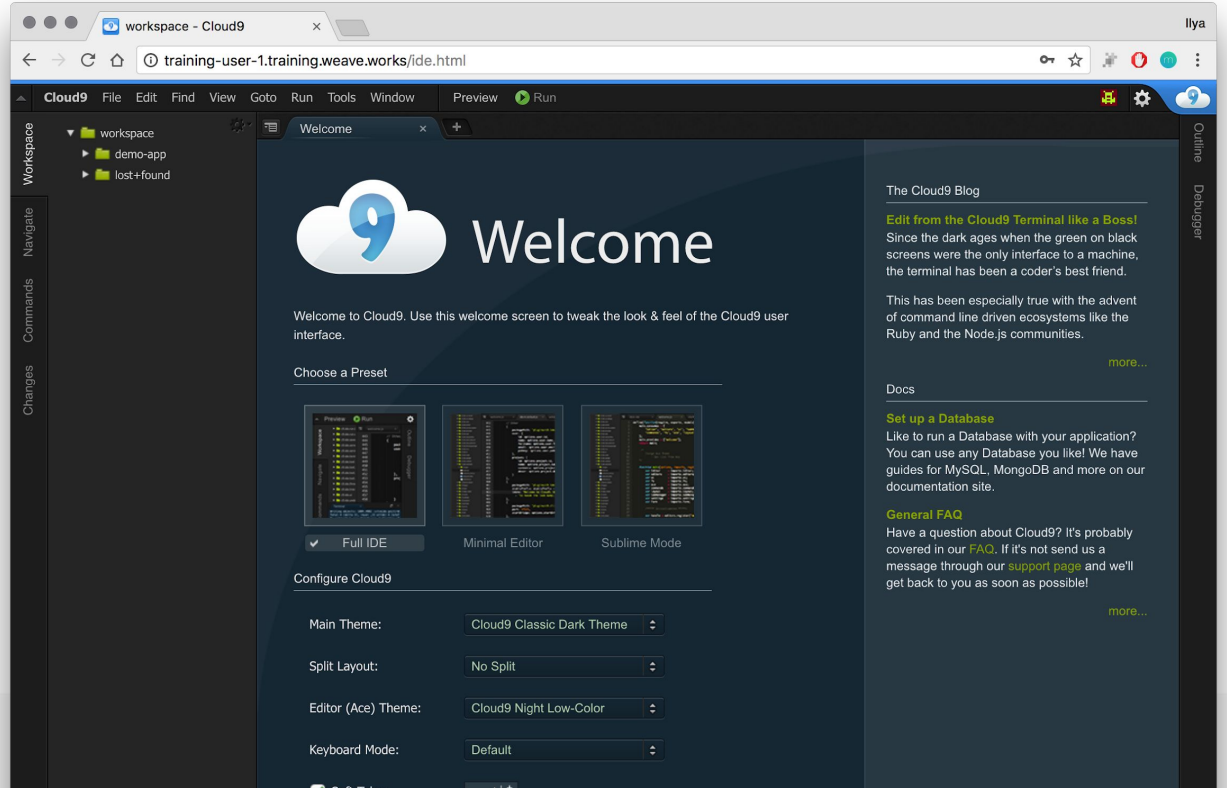
ARE YOU

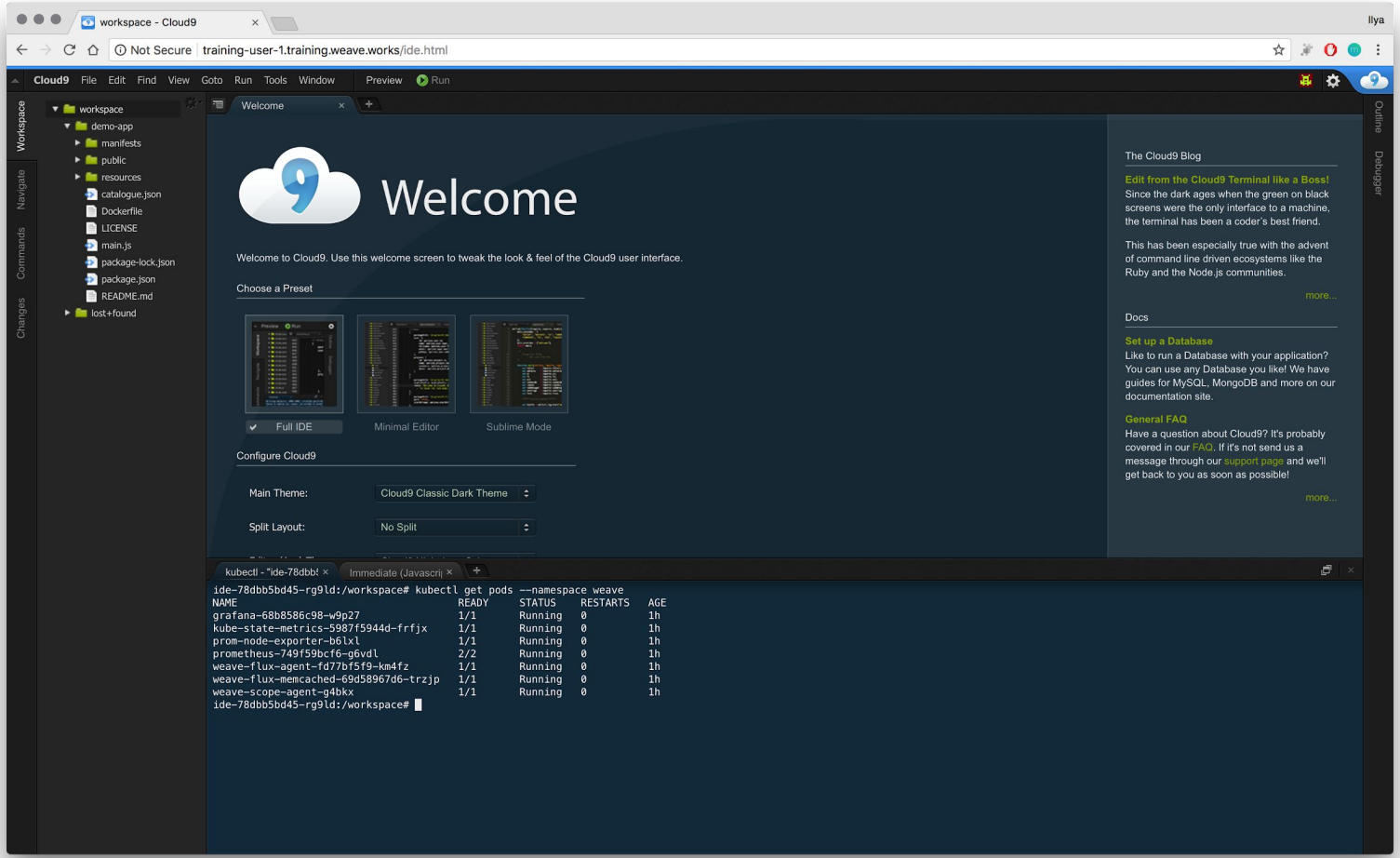
USING KUBERNETES?

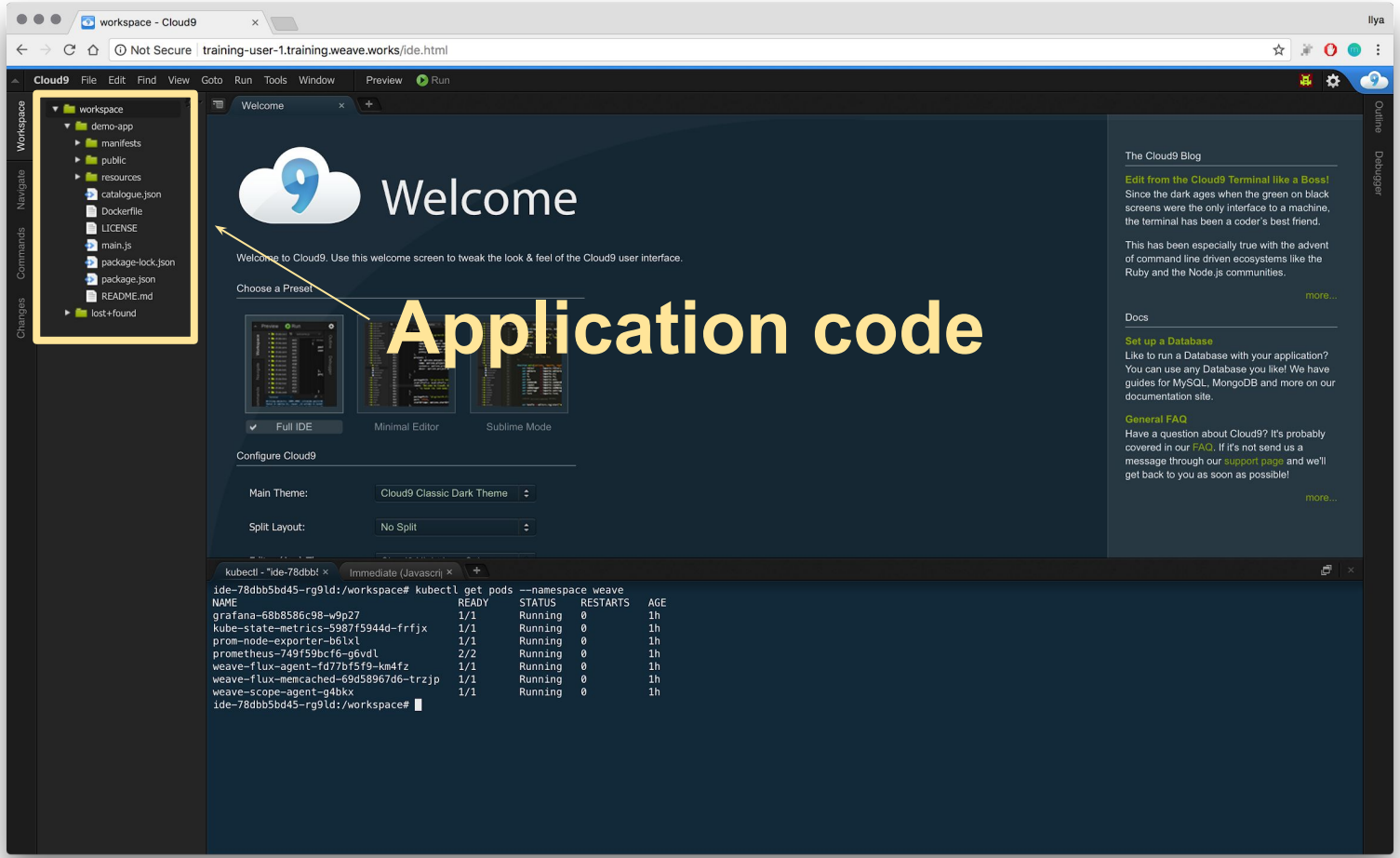
Getting started with your environment

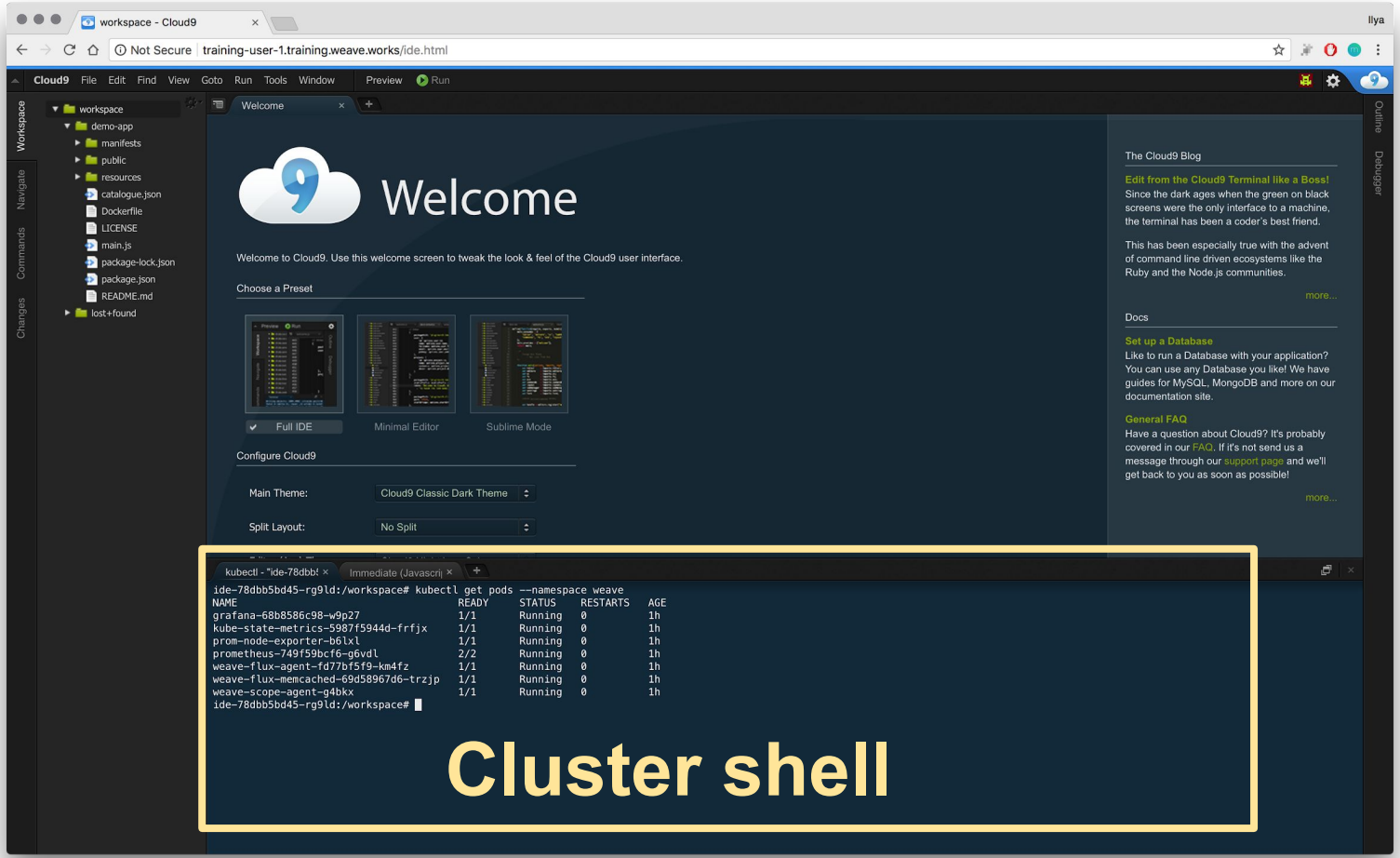
Login to your cluster – Cloud 9 IDE

Pick up a slip with login details. Username, Password and URL on the paper.

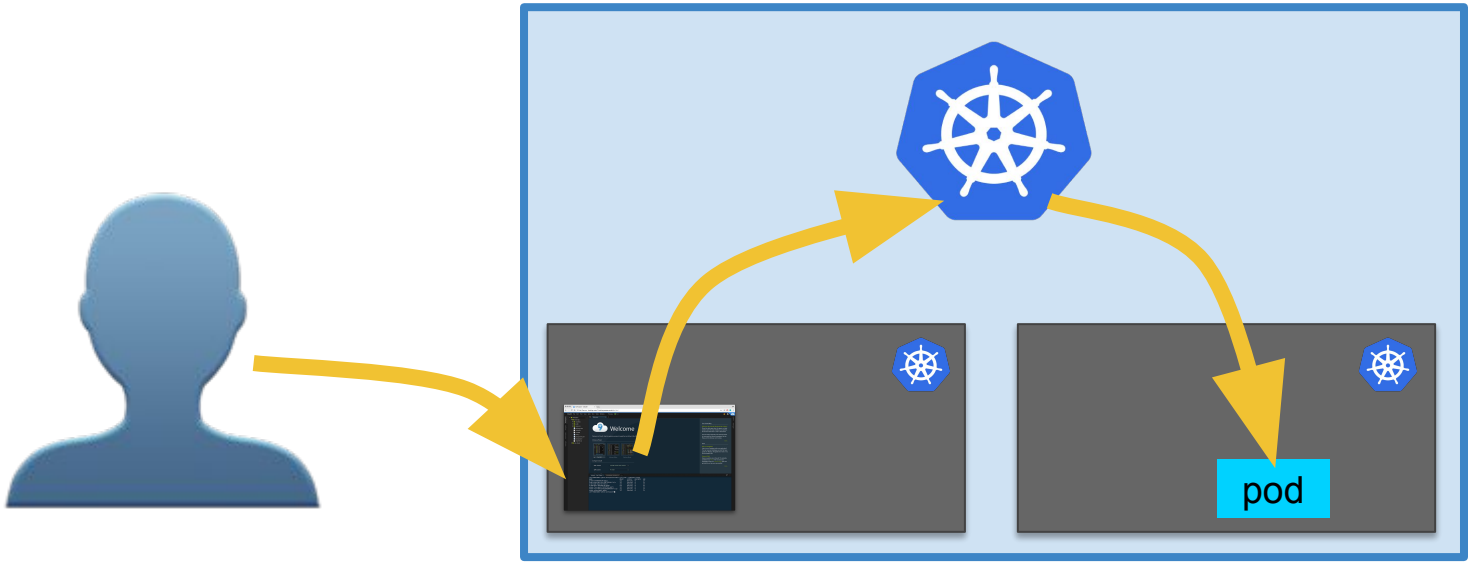



workspace - Cloud9
Ilya
Not Secure
training-user-1.training.weave.works/ide.html
Cloud9 File Edit Find View Goto Run Tools Window
Preview
Run
Workspace
Navigate
Commands
Changes
Outline
Debugger
workspace
demo-app
manifests
public
resources
catalogue.json
Dockerfile
LICENSE
main.js
package-lock.json
package.json
README.md
lost+found
Welcome
Welcome
Welcome to Cloud9. Use this welcome screen to tweak the look & feel of the Cloud9 user interface.
Choose a Preset
Full IDE
Minimal Editor
Sublime Mode
Configure Cloud9
Main Theme:
Cloud9 Classic Dark Theme
Split Layout:
No Split
The Cloud9 Blog
Edit from the Cloud9 Terminal like a Boss!
Since the dark ages when the green on black screens were the only interface to a machine, the terminal has been a coder's best friend.
This has been especially true with the advent of command line driven ecosystems like the Ruby and the Node.js communities.
more...
Docs
Set up a Database
Like to run a Database with your application? You can use any Database you like! We have guides for MySQL, MongoDB and more on our documentation site.
General FAQ
Have a question about Cloud9? It's probably covered in our FAQ. If it's not send us a message through our support page and we'll get back to you as soon as possible!
more...
kubectl - "ide-78dbb!
Immediate (Javascrij
ide-78dbb5bd45-rg9ld:/workspace# kubectl get pods --namespace weave
NAME READY STATUS RESTARTS AGE
grafana-68b8586c98-w9p27 1/1 Running 0 1h
kube-state-metrics-5987f5944d-frfjx 1/1 Running 0 1h
prom-node-exporter-b6lxl 1/1 Running 0 1h
prometheus-749f59bcf6-g6vdl 2/2 Running 0 1h
weave-flux-agent-fd77bf5f9-km4fz 1/1 Running 0 1h
weave-flux-memcached-69d58967d6-trzjp 1/1 Running 0 1h
weave-scope-agent-g4bkx 1/1 Running 0 1h
ide-78dbb5bd45-rg9ld:/workspace#


workspace - Cloud9
Ilya
Not Secure
training-user-1.training.weave.works/ide.html
Cloud9 File Edit Find View Goto Run Tools Window
Preview
Run
Workspace
Navigate
Commands
Changes
workspace
demo-app
manifests
public
resources
catalogue.json
Dockerfile
LICENSE
main.js
package-lock.json
package.json
README.md
lost+found
Welcome
Welcome
Welcome to Cloud9. Use this welcome screen to tweak the look & feel of the Cloud9 user interface.
Choose a Preset
Full IDE
Minimal Editor
Sublime Mode
Configure Cloud9
Main Theme:
Cloud9 Classic Dark Theme
Split Layout:
No Split
The Cloud9 Blog
Edit from the Cloud9 Terminal like a Boss!
Since the dark ages when the green on black screens were the only interface to a machine, the terminal has been a coder's best friend.
This has been especially true with the advent of command line driven ecosystems like the Ruby and the Node.js communities.
more...
Docs
Set up a Database
Like to run a Database with your application? You can use any Database you like! We have guides for MySQL, MongoDB and more on our documentation site.
General FAQ
Have a question about Cloud9? It's probably covered in our FAQ. If it's not send us a message through our support page and we'll get back to you as soon as possible!
more...
Outline
Debugger
kubectl - "ide-78dbb!
Immediate (Javascrij
ide-78dbb5bd45-rg9ld:/workspace# kubectl get pods --namespace weave
NAME READY STATUS RESTARTS AGE
grafana-68b8586c98-w9p27 1/1 Running 0 1h
kube-state-metrics-5987f5944d-frfjx 1/1 Running 0 1h
prom-node-exporter-b6lxl 1/1 Running 0 1h
prometheus-749f59bcf6-g6vdl 2/2 Running 0 1h
weave-flux-agent-fd77bf5f9-km4fz 1/1 Running 0 1h
weave-flux-memcached-69d58967d6-trzjp 1/1 Running 0 1h
weave-scope-agent-g4bkx 1/1 Running 0 1h
ide-78dbb5bd45-rg9ld:/workspace#
Application code


workspace - Cloud9
Ilya
Not Secure
training-user-1.training.weave.works/ide.html
Cloud9 File Edit Find View Goto Run Tools Window
Preview
Run
Welcome
Welcome
Welcome to Cloud9. Use this welcome screen to tweak the look & feel of the Cloud9 user interface.
Choose a Preset
Full IDE
Minimal Editor
Sublime Mode
Configure Cloud9
Main Theme:
Cloud9 Classic Dark Theme
Split Layout:
No Split
The Cloud9 Blog
Edit from the Cloud9 Terminal like a Boss!
Since the dark ages when the green on black screens were the only interface to a machine, the terminal has been a coder's best friend.
This has been especially true with the advent of command line driven ecosystems like the Ruby and the Node.js communities.
more...
Docs
Set up a Database
Like to run a Database with your application? You can use any Database you like! We have guides for MySQL, MongoDB and more on our documentation site.
General FAQ
Have a question about Cloud9? It's probably covered in our FAQ. If it's not send us a message through our support page and we'll get back to you as soon as possible!
more...
workspace
demo-app
manifests
public
resources
catalogue.json
Dockerfile
LICENSE
main.js
package-lock.json
package.json
README.md
lost+found
kubectl - "ide-78dbb!
Immediate (Javascri
ide-78dbb5bd45-rg9ld:/workspace# kubectl get pods --namespace weave
NAME READY STATUS RESTARTS AGE
grafana-68b8586c98-w9p27 1/1 Running 0 1h
kube-state-metrics-5987f5944d-frfjx 1/1 Running 0 1h
prom-node-exporter-b6lxl 1/1 Running 0 1h
prometheus-749f59bcf6-g6vdl 2/2 Running 0 1h
weave-flux-agent-fd77bf5f9-km4fz 1/1 Running 0 1h
weave-flux-memcached-69d58967d6-trzjp 1/1 Running 0 1h
weave-scope-agent-g4bkx 1/1 Running 0 1h
ide-78dbb5bd45-rg9ld:/workspace#
Cluster shell

Your Cluster



Icon by Freepik from www.flaticon.com

Kick the tires on your cluster

1. Start with a simple command:

```
➤ kubectl version
```

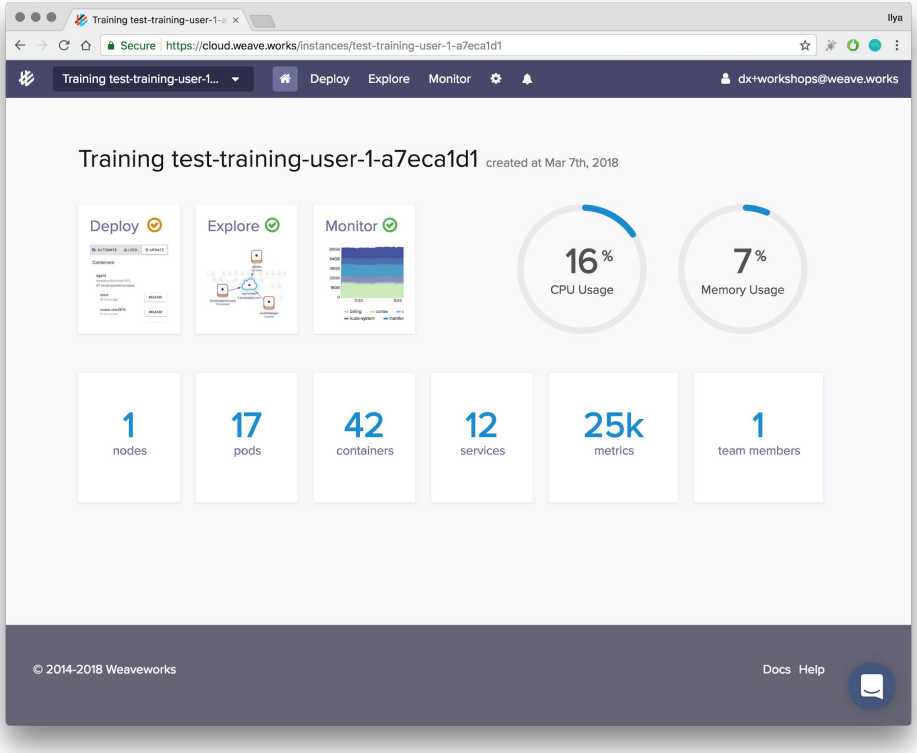
2. Look at what's running on the cluster with

```
➤ kubectl get pods --all-namespaces
```

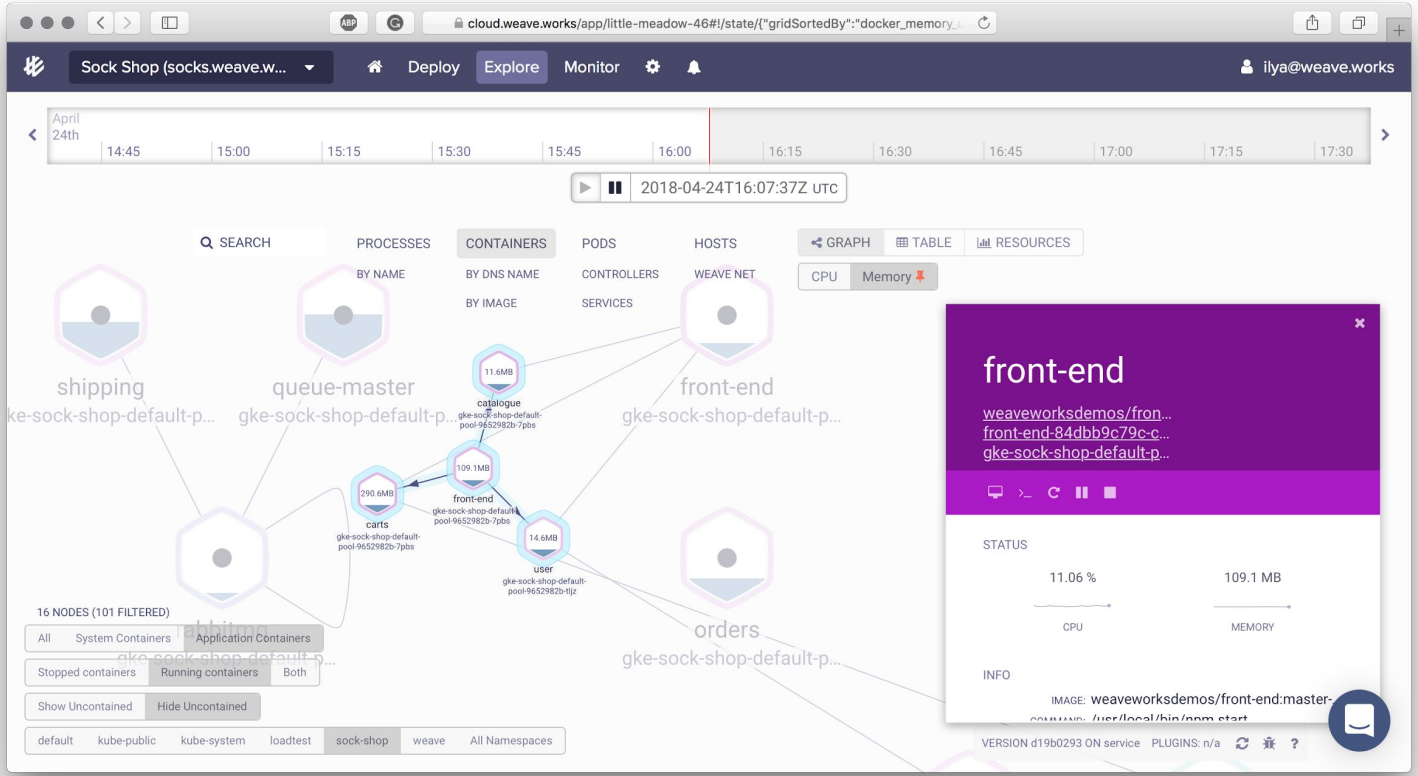
Weave Cloud

"DevOps Console"

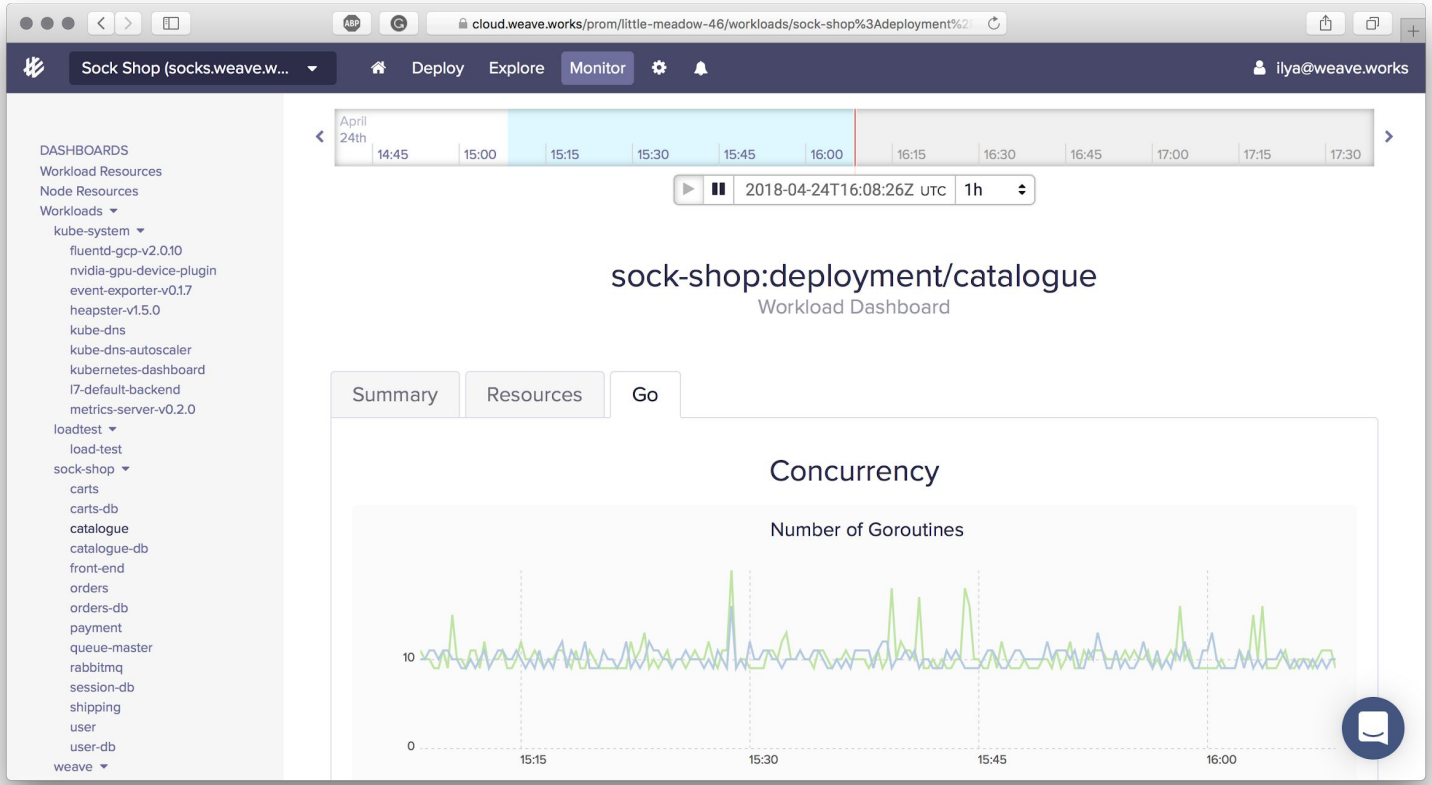
Tooling for deployment, visualisation and observability



Weave Cloud Explore



Weave Cloud Monitor



Install Weave Cloud

https://cloud.weave.works

Follow the install wizard

Run the command in your console

weavecloud

Easily deploy, manage, and monitor container-based applications

Faster deployment

Achieve faster deployment of containerized applications through repeatable and consistent deployment from laptop to production across multiple clusters. Automate it or manually push, all with your preferred CI tool.

Powerful insights

No matter how dynamic your microservices environment, Weave Cloud's hosted, scalable Prometheus service lets you query across hosts, services, and metrics, and quickly identify issues with your app.

Control through visibility

Observe and understand how your applications and microservices are connected together in containers, making complex troubleshooting simple.

Sign up and get a 14-day free trial

Log in with GitHub

Log in with Google

Or sign up with email

example@domain.com

first name

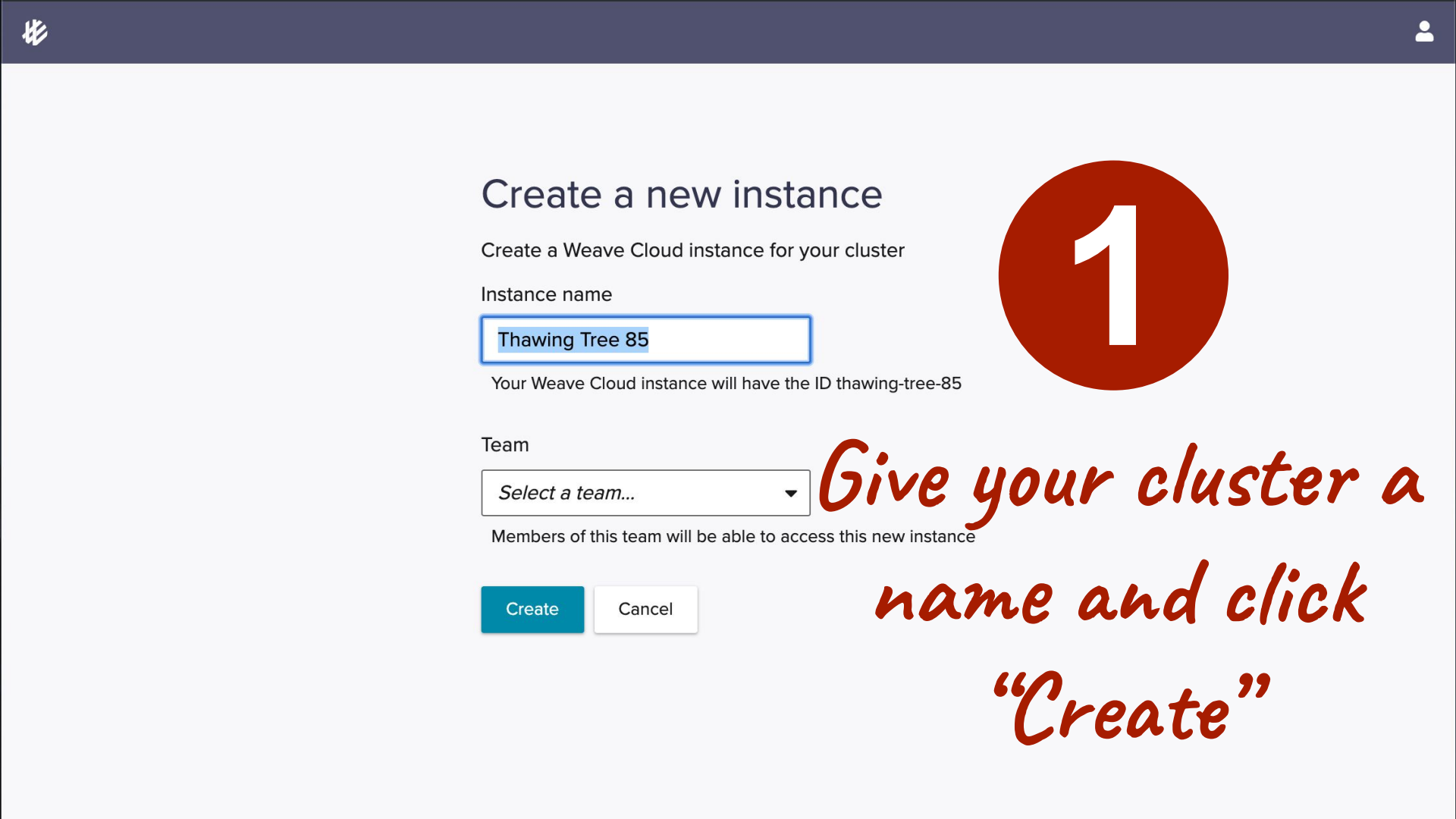
last name

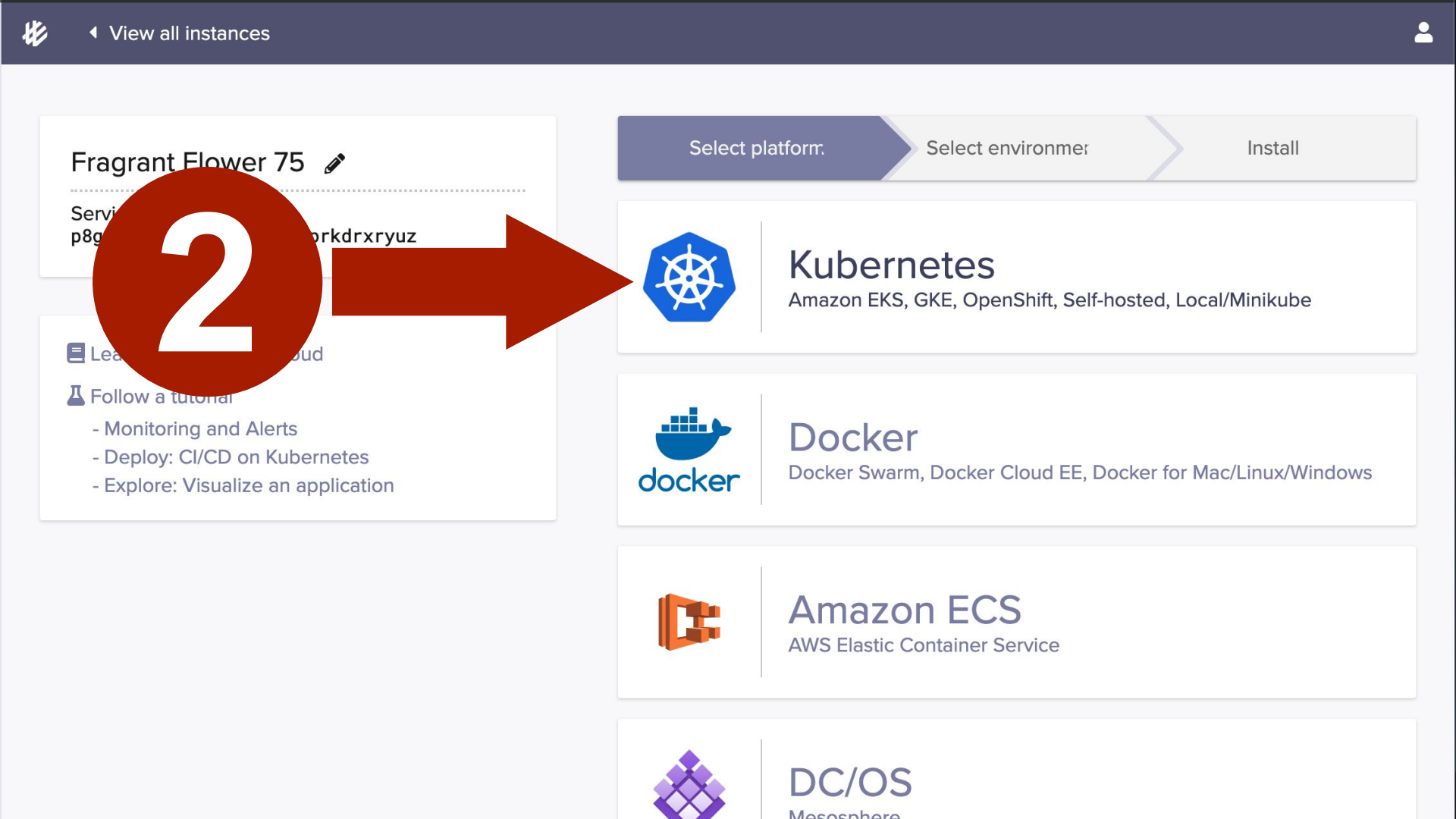
company

Email me a login link

Already have an account? Log in

By clicking on the buttons above, you are agreeing to our Terms of Service and Privacy Policy.


Create a new instance
Create a Weave Cloud instance for your cluster
Instance name
Thawing Tree 85
Your Weave Cloud instance will have the ID thawing-tree-85
Team
Select a team...
Members of this team will be able to access this new instance
Create
Cancel
1
Give your cluster a name and click "Create"


View all instances
Fragrant Flower 75
2
Select platform
Select environment
Install
Kubernetes
Amazon EKS, GKE, OpenShift, Self-hosted, Local/Minikube
Follow a tutorial
- Monitoring and Alerts
- Deploy: CI/CD on Kubernetes
- Explore: Visualize an application
docker
Docker
Docker Swarm, Docker Cloud EE, Docker for Mac/Linux/Windows
Amazon ECS
AWS Elastic Container Service
DC/OS

View all instances

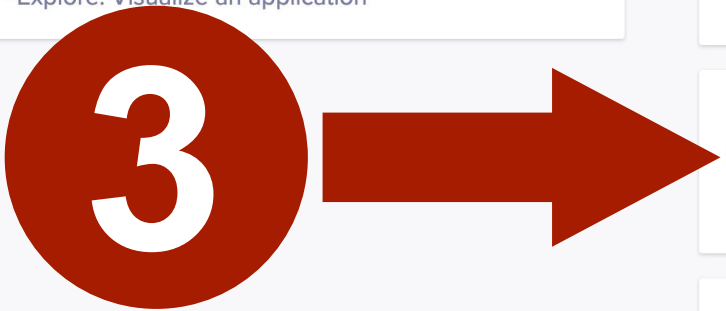
Brice

lakeside-hackfest

Service token:
5987p5dc4ajf7rjyx1coin674598mjmw

Learn about Weave Cloud

Follow a tutorial
- Monitoring and Alerts
- Deploy: CI/CD on Kubernetes
- Explore: Visualize an application



Platform
Kubernetes

Select environment

Install

Minikube
I use minikube to manage a local Kubernetes VM

Amazon EKS
I am deploying to Amazon Elastic Container Service for Kubernetes

Google Container Engine
I am deploying to GKE

Generic Kubernetes
I have a cluster of servers running Kubernetes

View all instances

Brice

lakeside-hackfest

Service token:
5987p5dc4ajf7rjyx1coin674598mjmw

Learn about Weave Cloud

Follow a tutorial

- Monitoring and Alerts
- Deploy: CI/CD on Kubernetes
- Explore: Visualize an application

Platform: Kubernetes | Environment: GKE | Install

Install the Weave Cloud Agents

Run the following command against your cluster to deploy all Weave Cloud agents:

```
curl -Ls https://get.weave.works |
  sh -s -- --token=5987p5dc4ajf7rjyx1coin674598mjmw --gke
```

You can uninstall the Weave Cloud agents at any time with `$ kubectl delete namespac  eave`

Started installing agents

Waiting for Weave Cloud agents to connect

4

Copy the command into your Cloud 9 terminal

lakeside-hackfest | View all instances | Brice

lakeside-hackfest

Service token:
`5987p5dc4ajf7rjyx1coin674598mjmw`

- Learn about Weave Cloud
- Follow a tutorial
  - Monitoring and Alerts
  - Deploy: CI/CD on Kubernetes
  - Explore: Visualize an application

Platform: Kubernetes | Environment: GKE | You are connected!

Install the Weave Cloud Agents

Run the following command against your cluster to deploy all Weave Cloud agents:

```
curl -Ls https://get.weave.works |
  sh -s -- --token=5987p5dc4ajf7rjyx1coin674598mjmw --gke
```

You can uninstall the Weave Cloud agents at any time with `$ kubectl delete namespace weave`

- Started installing agents on Kubernetes cluster v1.13.7-gke.8
- All Weave Cloud agents are connected!

5 Wait for

View your cluster

Check what was installed

Look at what's running on the cluster with

```
➤ kubectl get pods --all-namespaces
```

Checkpoint

Flux running in the cluster

```
ide-7f649d7745-h2c99:/workspace$ kubectl get pods --all-namespaces
NAMESPACE     NAME                                                    READY     STATUS    RESTARTS   AGE
ide           git-server-b8977c48-x6zfl                               1/1       Running   0          55m
ide           ide-7f649d7745-h2c99                                    1/1       Running   0          55m
kube-system   cert-manager-59cffdc65d-vd8x2                           1/1       Running   0          55m
kube-system   external-dns-controller-75b88ccfb-n5krb                 1/1       Running   0          55m
kube-system   heapster-v1.6.1-7bbf896c75-zjb6p                        3/3       Running   0          55m
kube-system   kube-dns-6987857fdb-zmqvx                               4/4       Running   0          55m
kube-system   kube-dns-autoscaler-bb58c6784-k2sgl                     1/1       Running   0          55m
kube-system   kube-proxy-gke-training-user-1-default-pool-4619e880-pz0x  1/1       Running   0          55m
kube-system   l7-default-backend-fd59995cd-lwqx8                      1/1       Running   0          55m
kube-system   metrics-server-v0.3.1-57c75779f-fcn7m                   2/2       Running   0          55m
kube-system   prometheus-to-sd-47w58                                  1/1       Running   0          55m
kube-system   traefik-ingress-controller-bb6gs                        1/1       Running   0          55m
weave         kube-state-metrics-64c8cdd4f9-f4bfm                     1/1       Running   0          10m
weave         prom-node-exporter-vn9jt                                1/1       Running   0          10m
weave         prometheus-5c4dcb7bb6-cnng5                             2/2       Running   0          10m
weave         weave-agent-799c9b6db-bwjl2                             1/1       Running   0          10m
weave         weave-flux-agent-6cddd4c885-5d2qq                       1/1       Running   0          10m
weave         weave-flux-memcached-57db5b87f7-bc457                   1/1       Running   0          10m
weave         weave-scope-agent-sqdq8                                 1/1       Running   0          10m
weave         weave-scope-cluster-agent-56c9f6b866-qhtwv              1/1       Running   0          10m
```

What is GitOps

GitOps is...

GitOps is...

An operation model

GitOps is...

An operation model

Derived from CS and operation knowledge

GitOps is...

An operation model

Derived from CS and operation knowledge

Technology agnostic (name notwithstanding)

GitOps is...

An operation model

Derived from CS and operation knowledge

Technology agnostic (name notwithstanding)

A set of principles (Why instead of How)

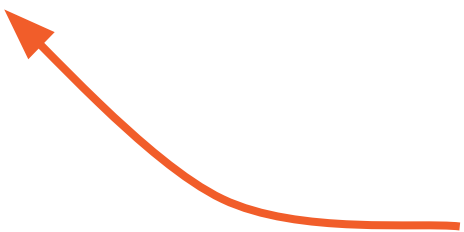
GitOps is...

An operation model

Derived from CS and operation knowledge

Technology agnostic (name notwithstanding)

A set of principles (Why instead of How)



Although Weaveworks can help with how

GitOps is...

An operation model

Derived from CS and operation knowledge

Technology agnostic (name notwithstanding)

A set of principles (Why instead of How)

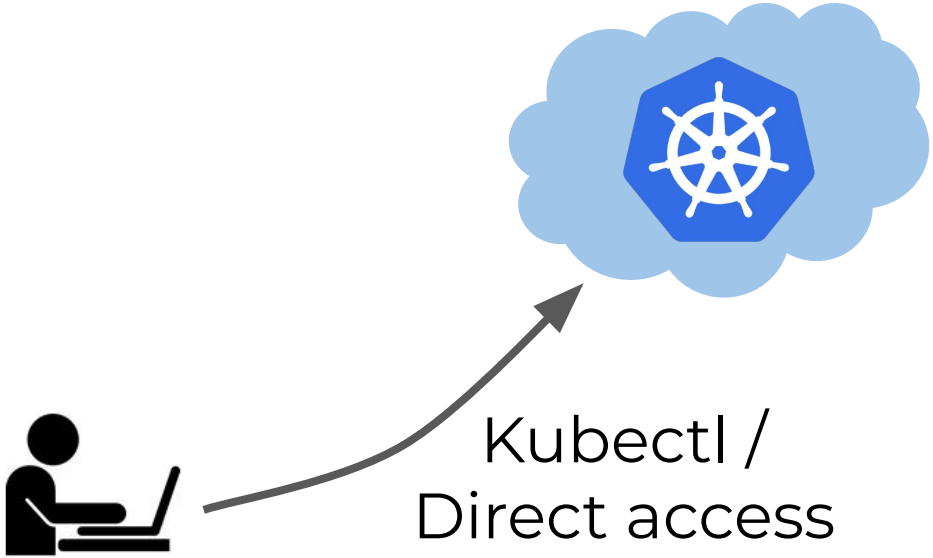
A way to speed up your team

The GitOps Model

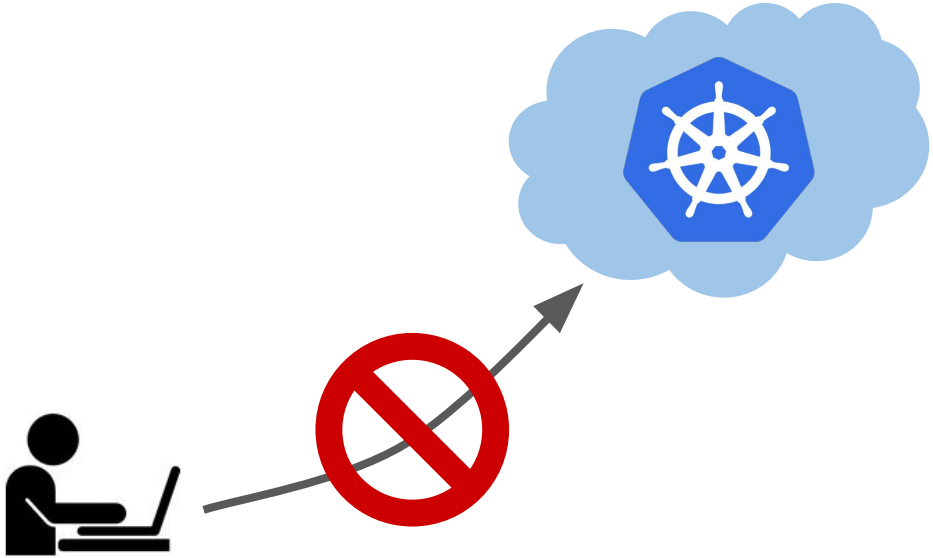
GitOps ON Kubernetes

Kubernetes
Cluster

GitOps ON Kubernetes

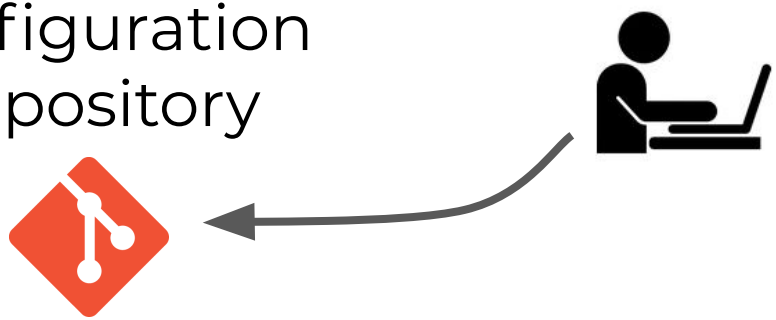


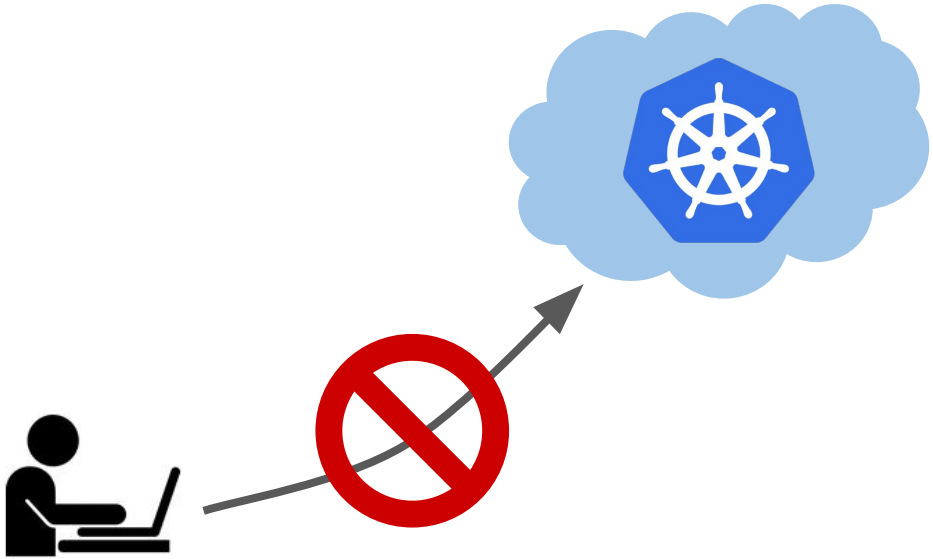
GitOps ON Kubernetes



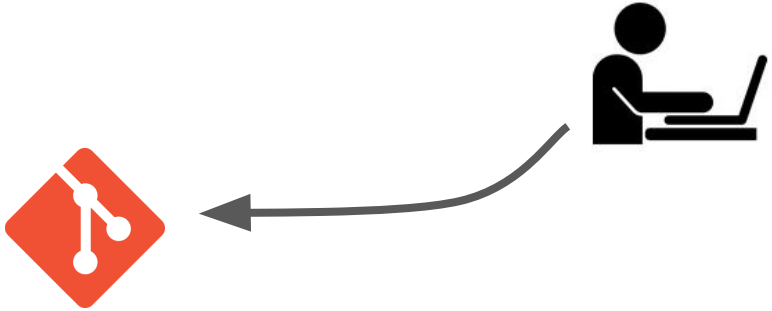
GitOps ON Kubernetes

Configuration Repository

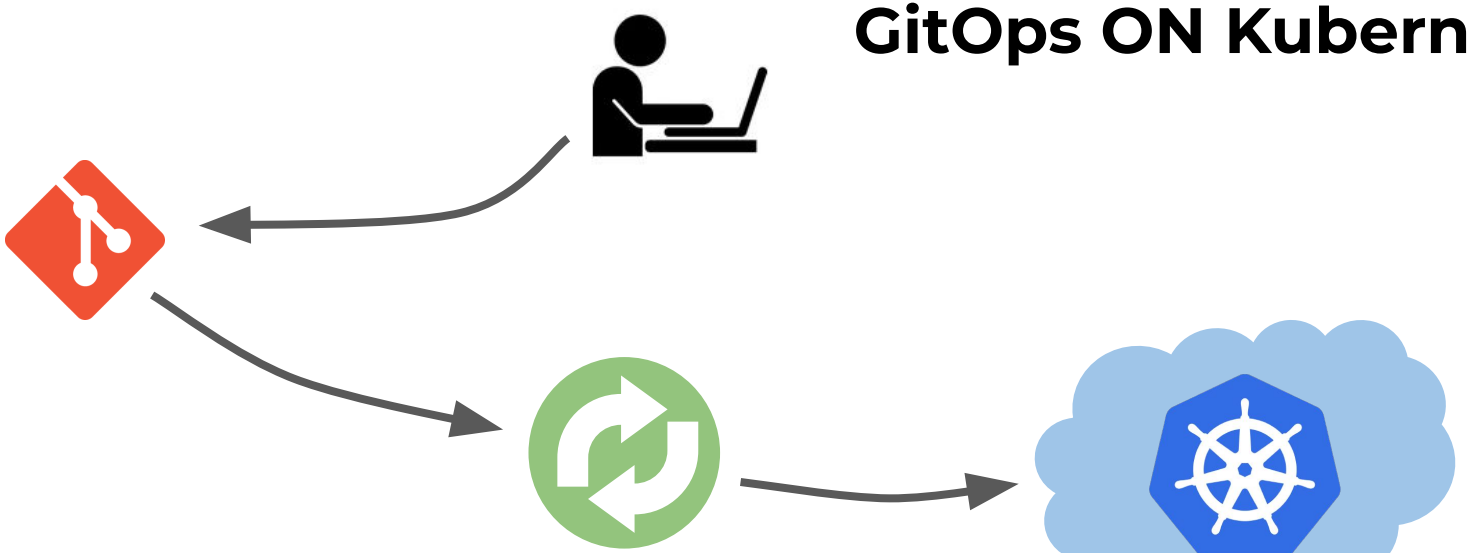




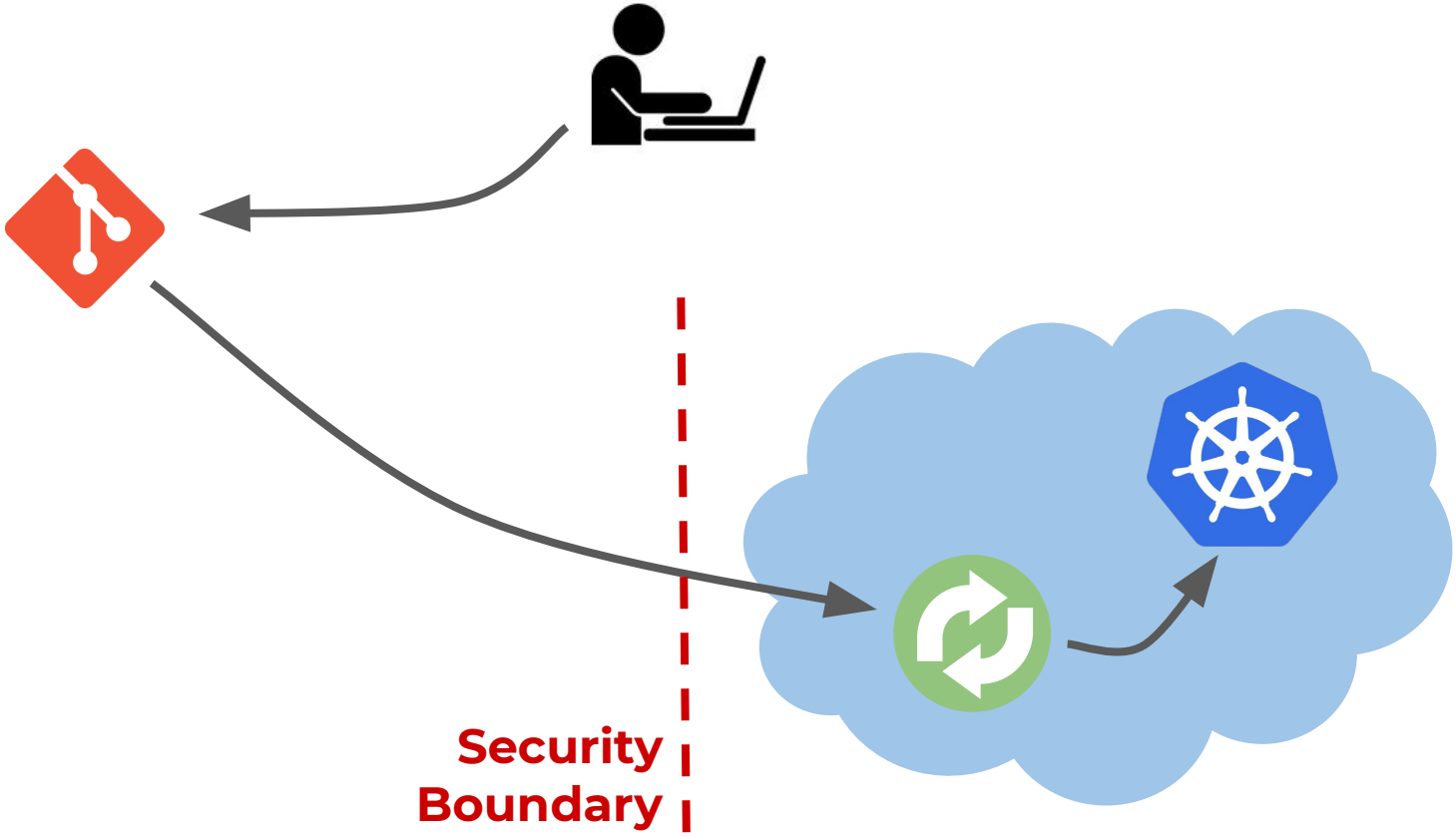
GitOps ON Kubernetes



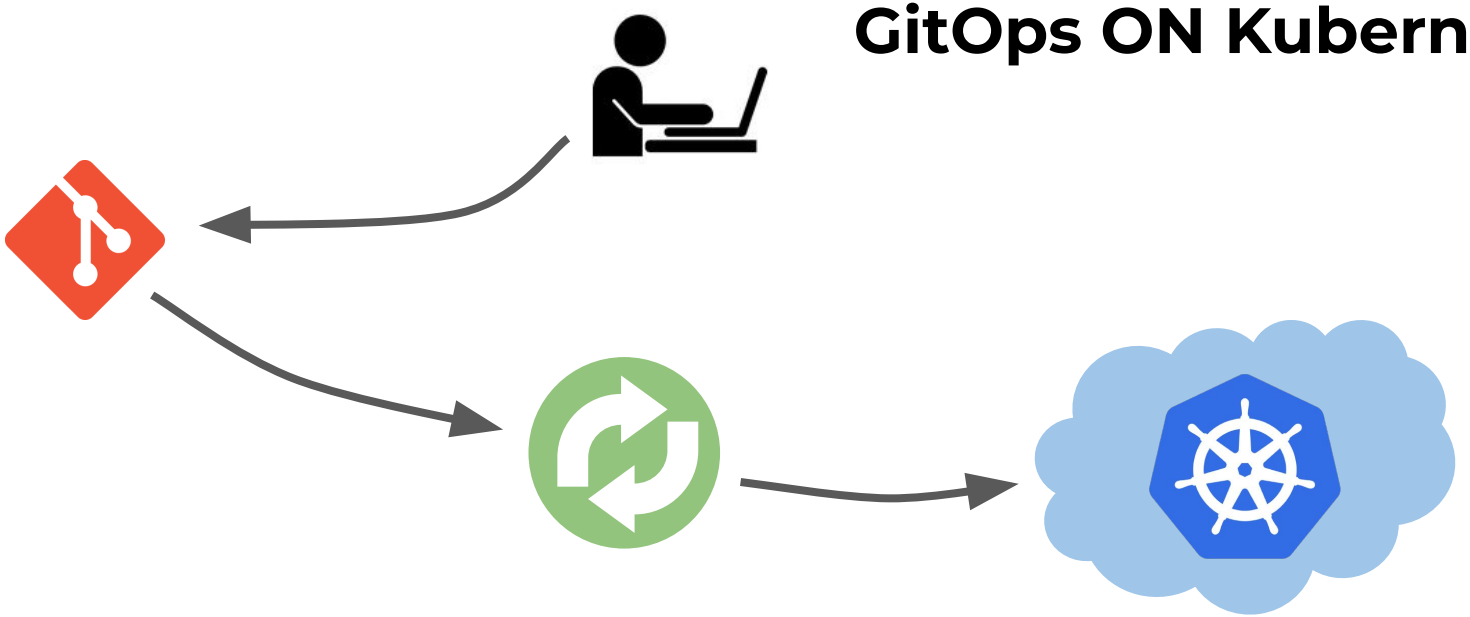
GitOps ON Kubernetes



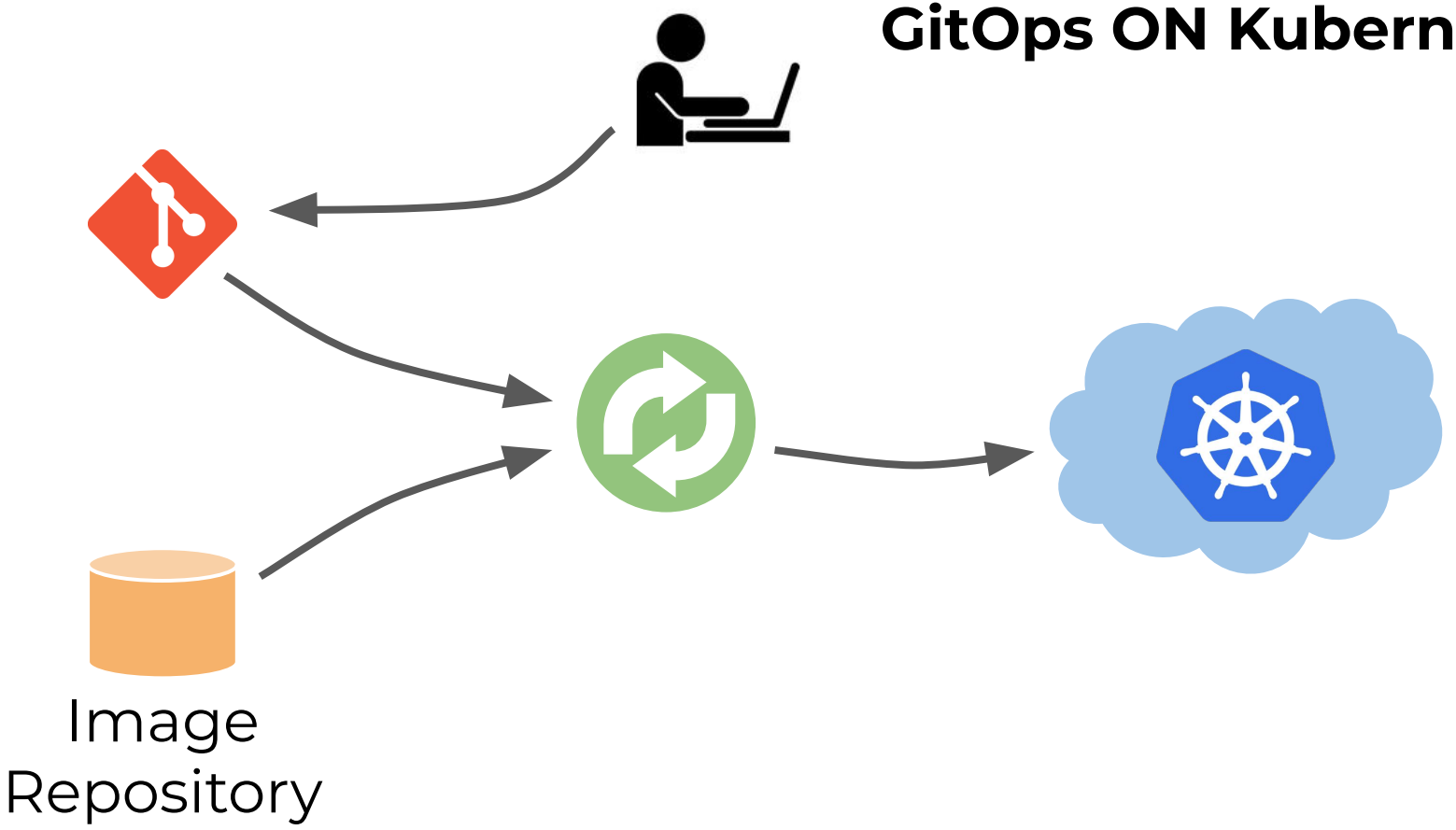
*


Security
Boundary

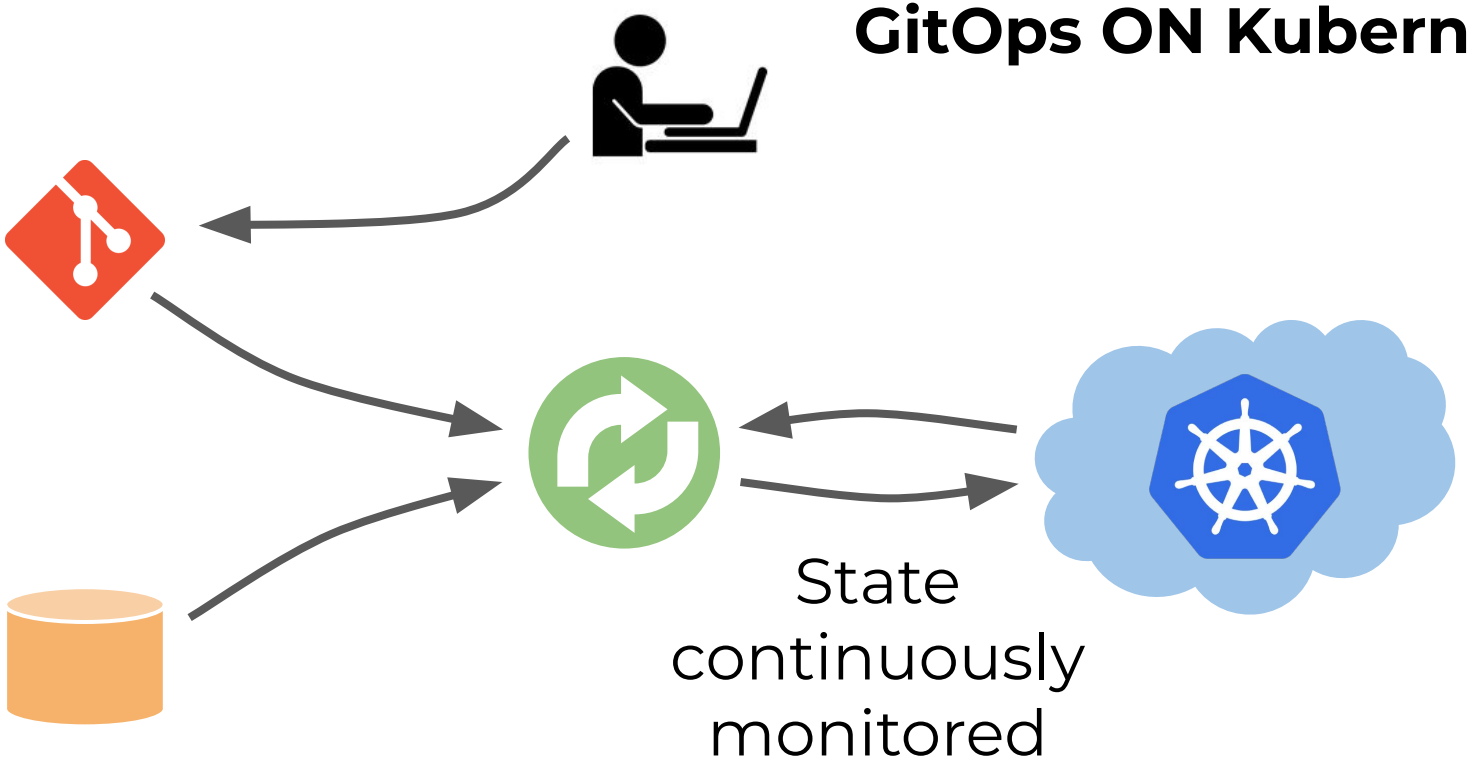
GitOps ON Kubernetes



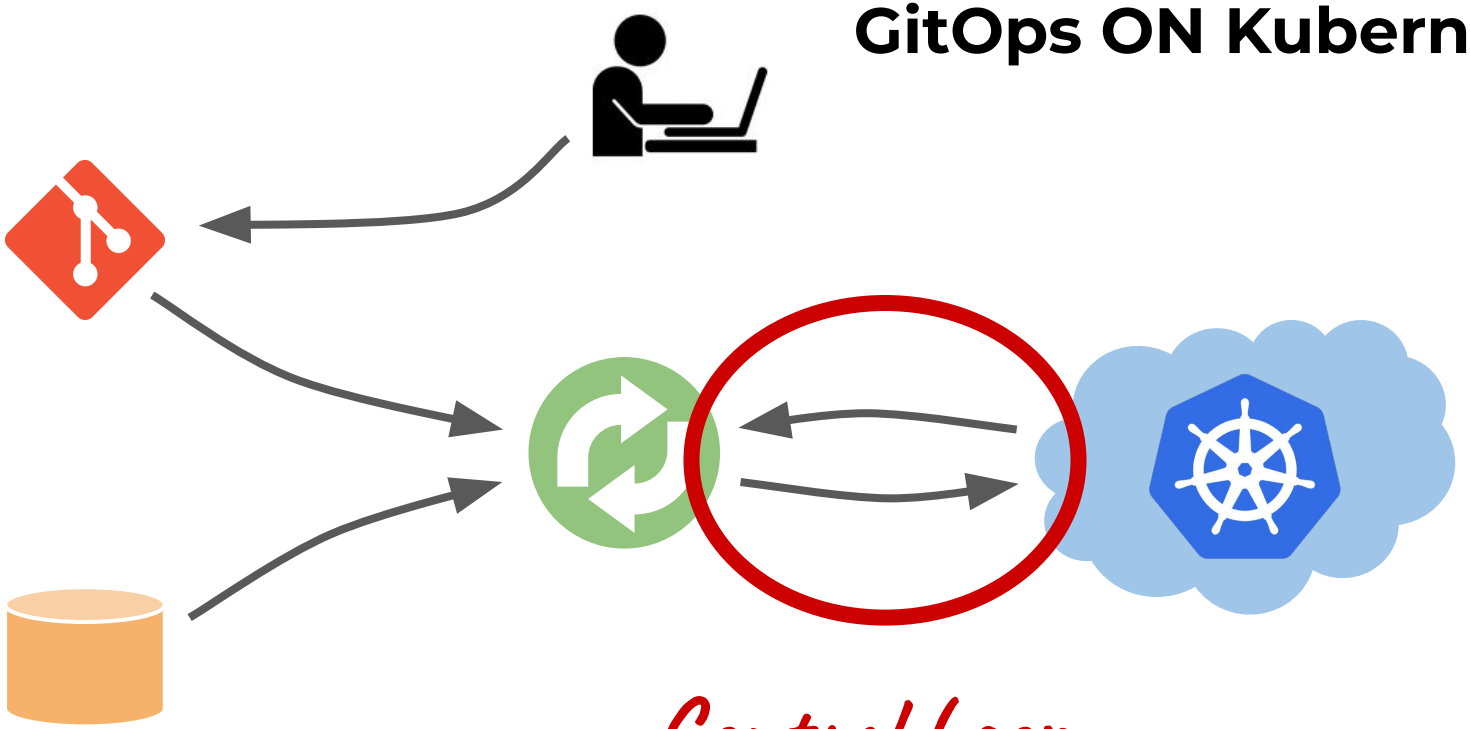
GitOps ON Kubernetes



GitOps ON Kubernetes



GitOps ON Kubernetes



Principles of GitOps

1. The entire system is described declaratively.
2. The canonical desired system state is versioned (with Git)
3. Approved changes to the desired state are automatically applied to the system
4. Software agents ensure correctness and alert on divergence

1. The entire system is described declaratively.

1 The entire system is described declaratively.

Beyond code, data ⇒

Implementation independent

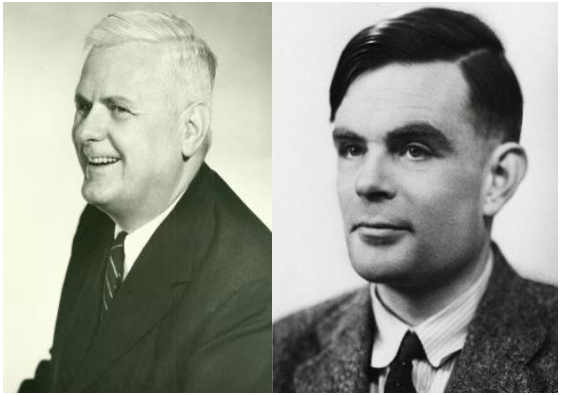
Easy to abstract in simple ways

Easy to validate for correctness

Easy to generate & manipulate from code

1 The entire system is described declaratively.

Beyond code, data ⇒



Implementation independent

Easy to abstract in simple ways

Easy to validate for correctness

Easy to generate & manipulate from code

How is that different from Infrastructure as code?

How is that different from Infrastructure as code?

It’s about consistency in the failure case.

It’s about consistency in the failure case.

When imperative systems fail, the system ends up in an **unknown, inconsistent state**.

an **unknown, inconsistent state**.

Declarative changes let you think of changes as transactions.

Declarative changes let you think of changes as transactions.

This is a very good thing.

2 The canonical desired system state is versioned (with Git)

2 The canonical desired system state is versioned (with Git)

Canonical Source of Truth (DRY)

With declarative definition, trivialises rollbacks

Excellent security guarantees for auditing

Sophisticated approval processes (& existing workflows)

Great Software ↔ Human collaboration point

3 Approved changes to the desired state are automatically applied to the system

3 Approved changes to the desired state are automatically applied to the system

Significant velocity gains

Privileged operators don’t cross security boundaries

Separates **What** and **How**.

4 Software agents ensure correctness and alert on divergence

4 Software agents ensure correctness and alert on divergence

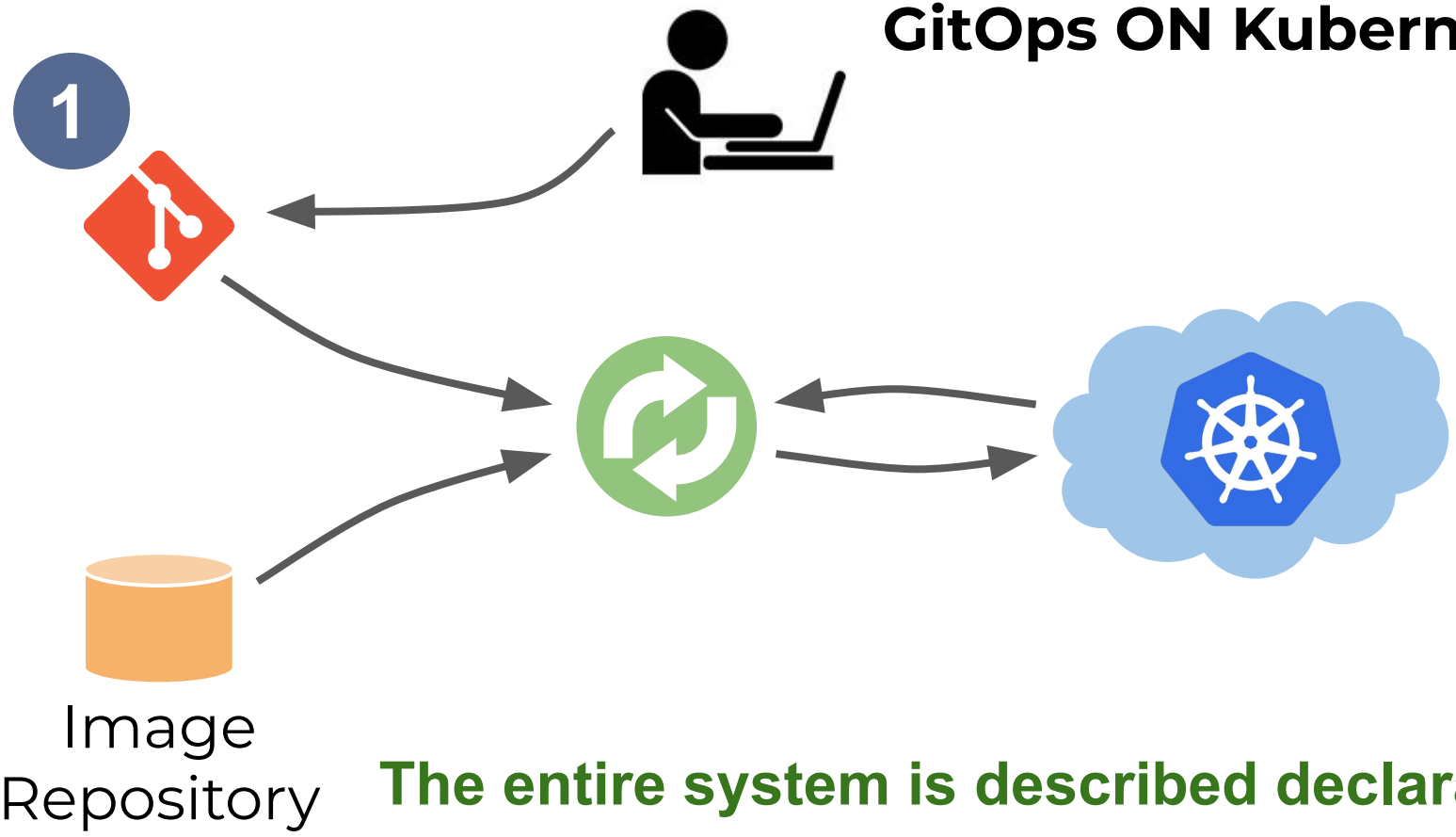
Continuously checking that desired state is met

System can self heal

Recovers from errors without intervention (PEBKAC)

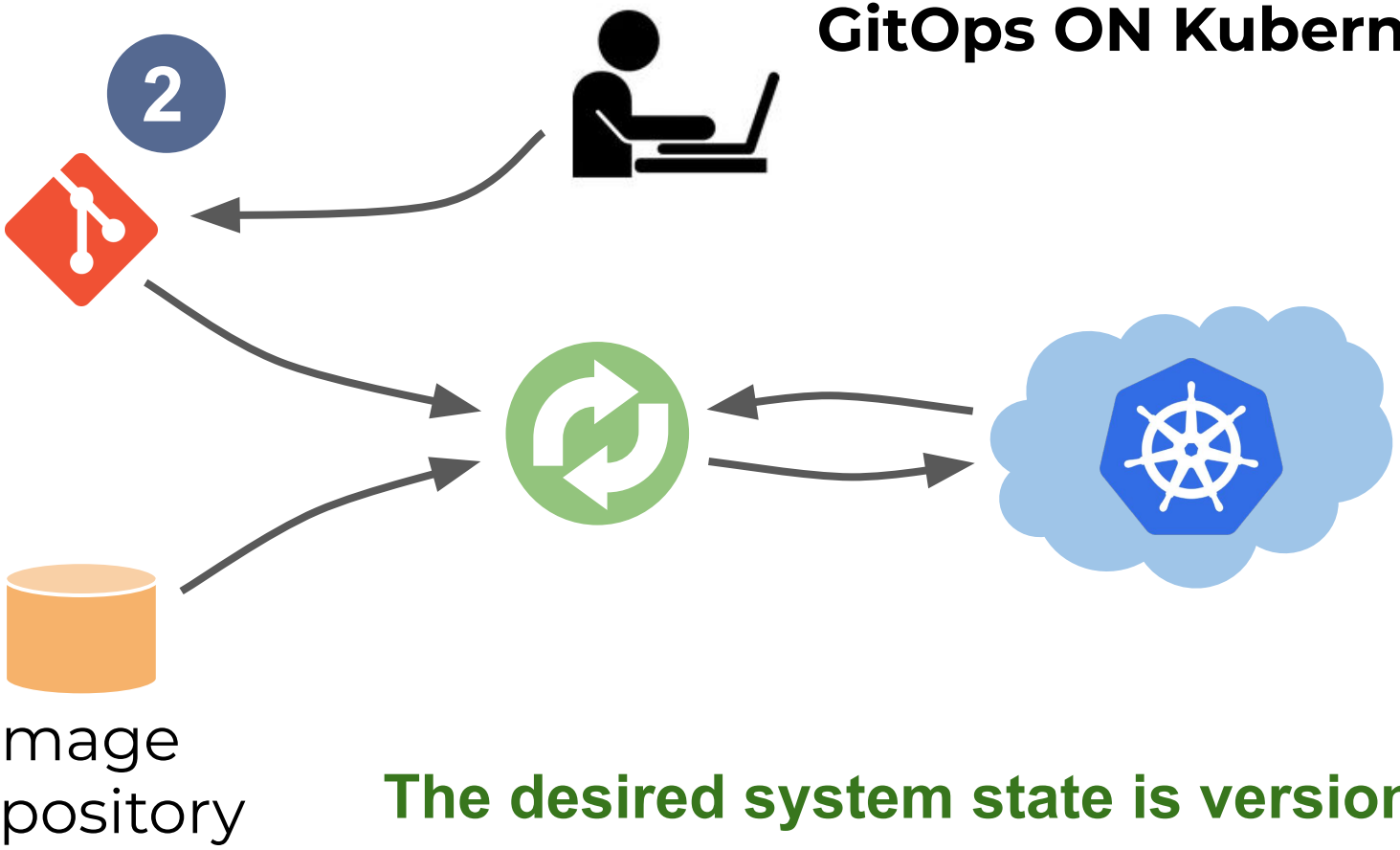
It’s the control loop for your operations

GitOps ON Kubernetes



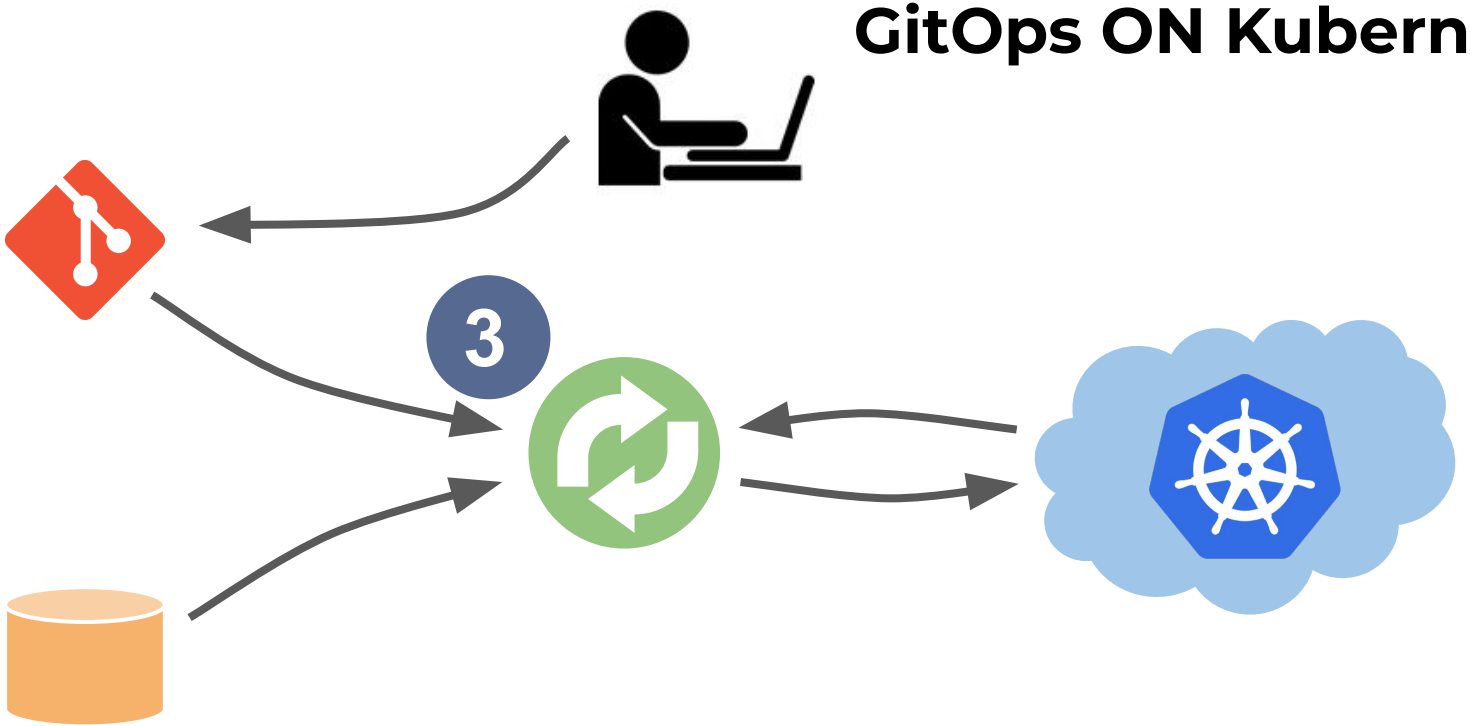
The entire system is described declaratively.

GitOps ON Kubernetes



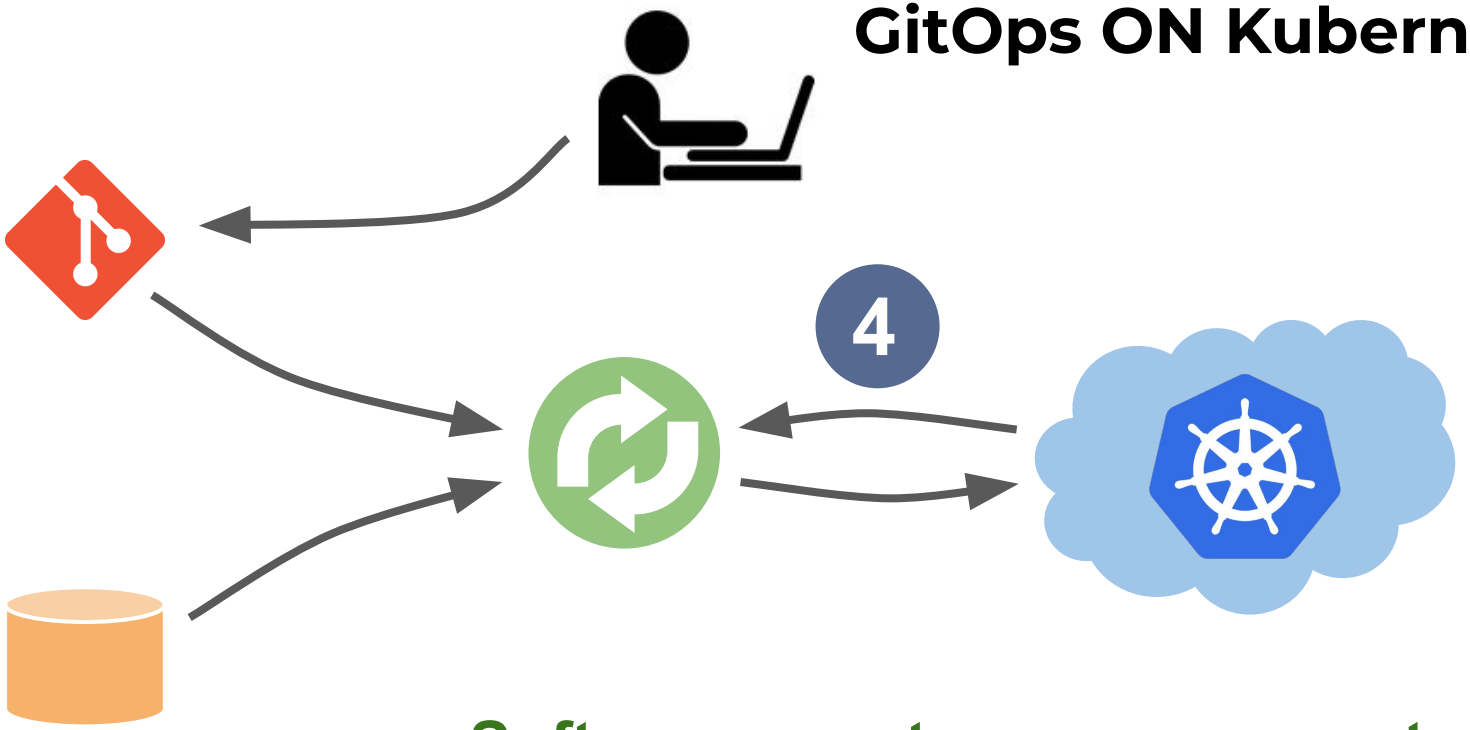
The desired system state is versioned

GitOps ON Kubernetes



Approved changes to the desired state are automatically applied to the system

GitOps ON Kubernetes



Software agents ensure correctness and alert on divergence

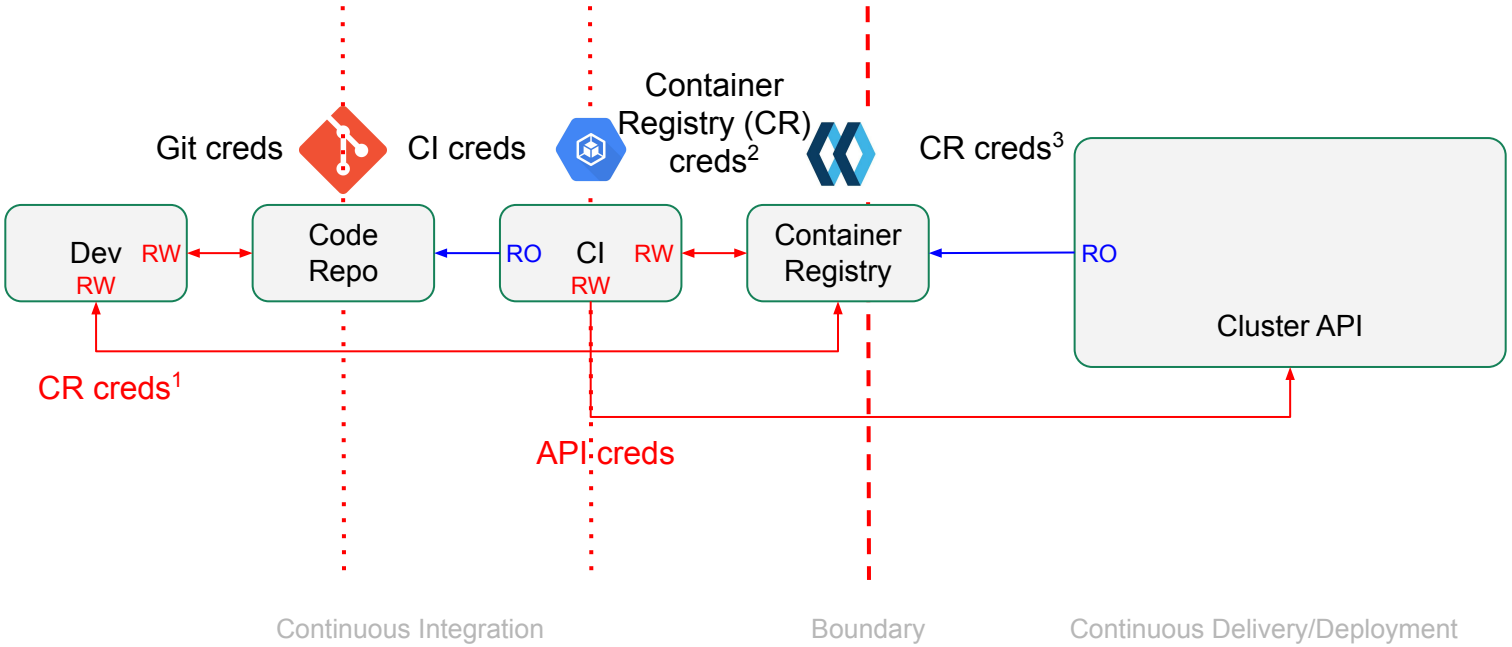
Gitops is Functional Reactive Programming…

...for your infrastructure.

Like React, but for servers and applications.

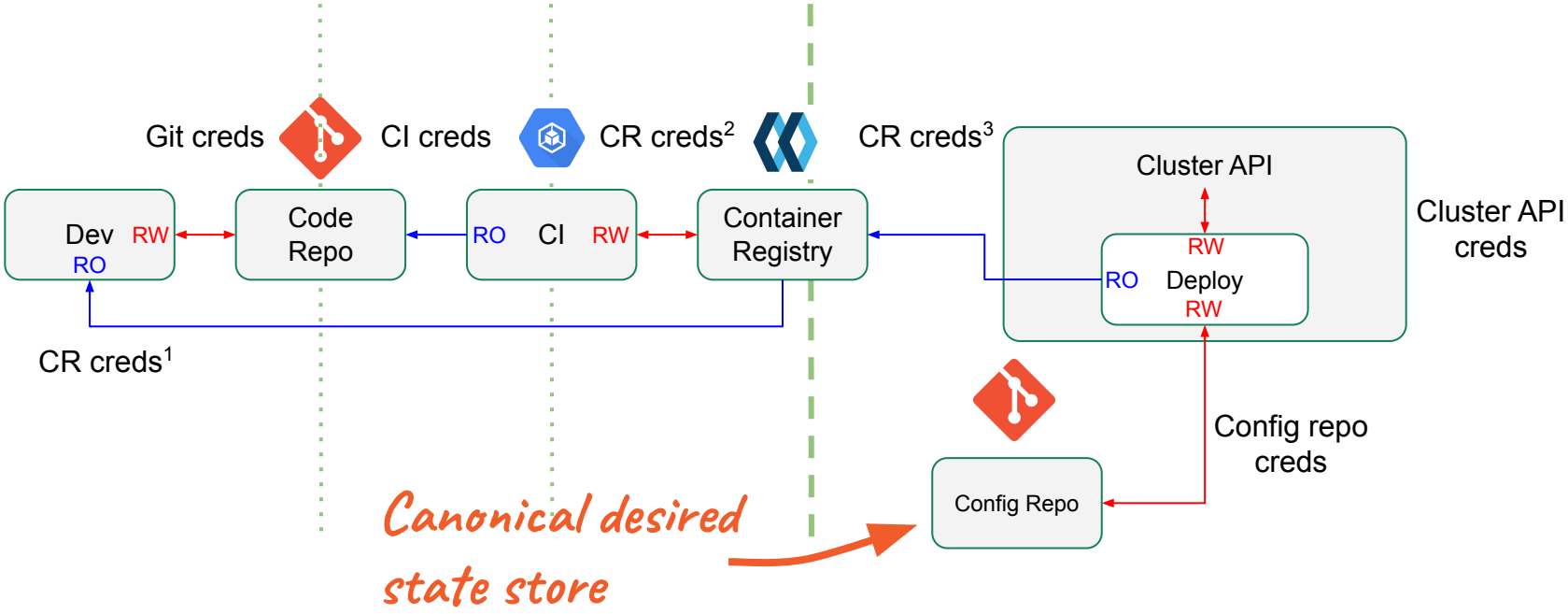
Typical CICD pipeline

Shares credentials cross several logical security boundaries.



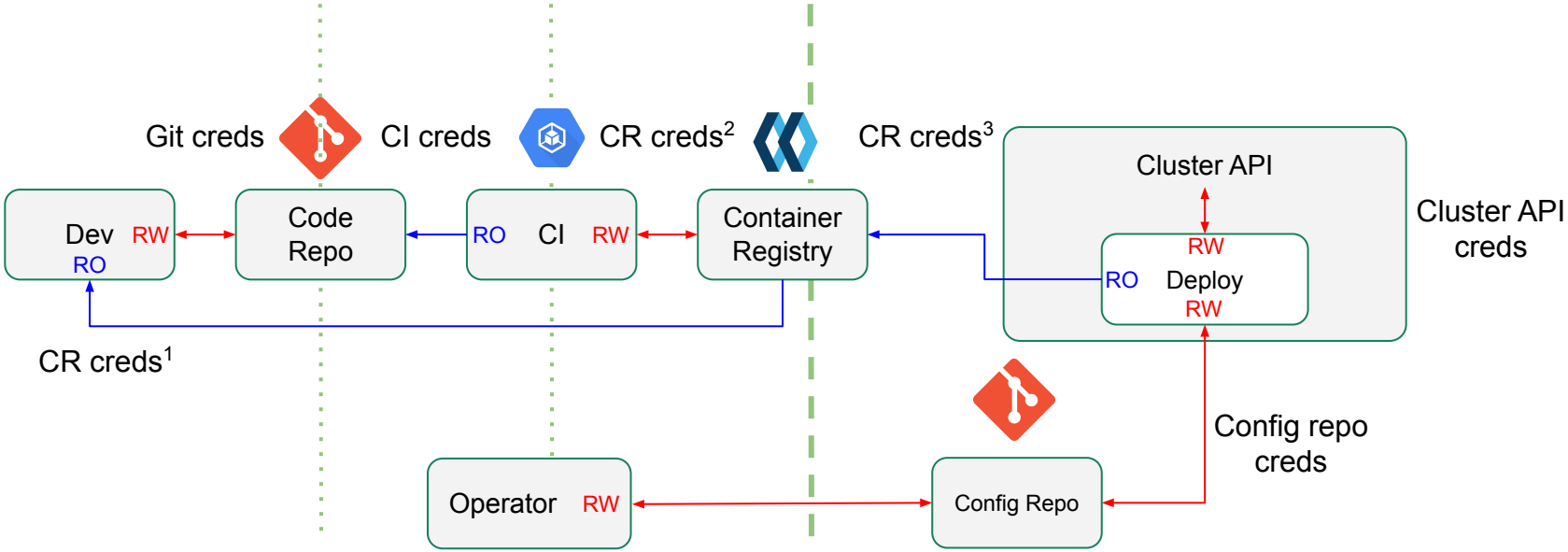
GitOps pipeline

Credentials are never shared across a logical security boundary.



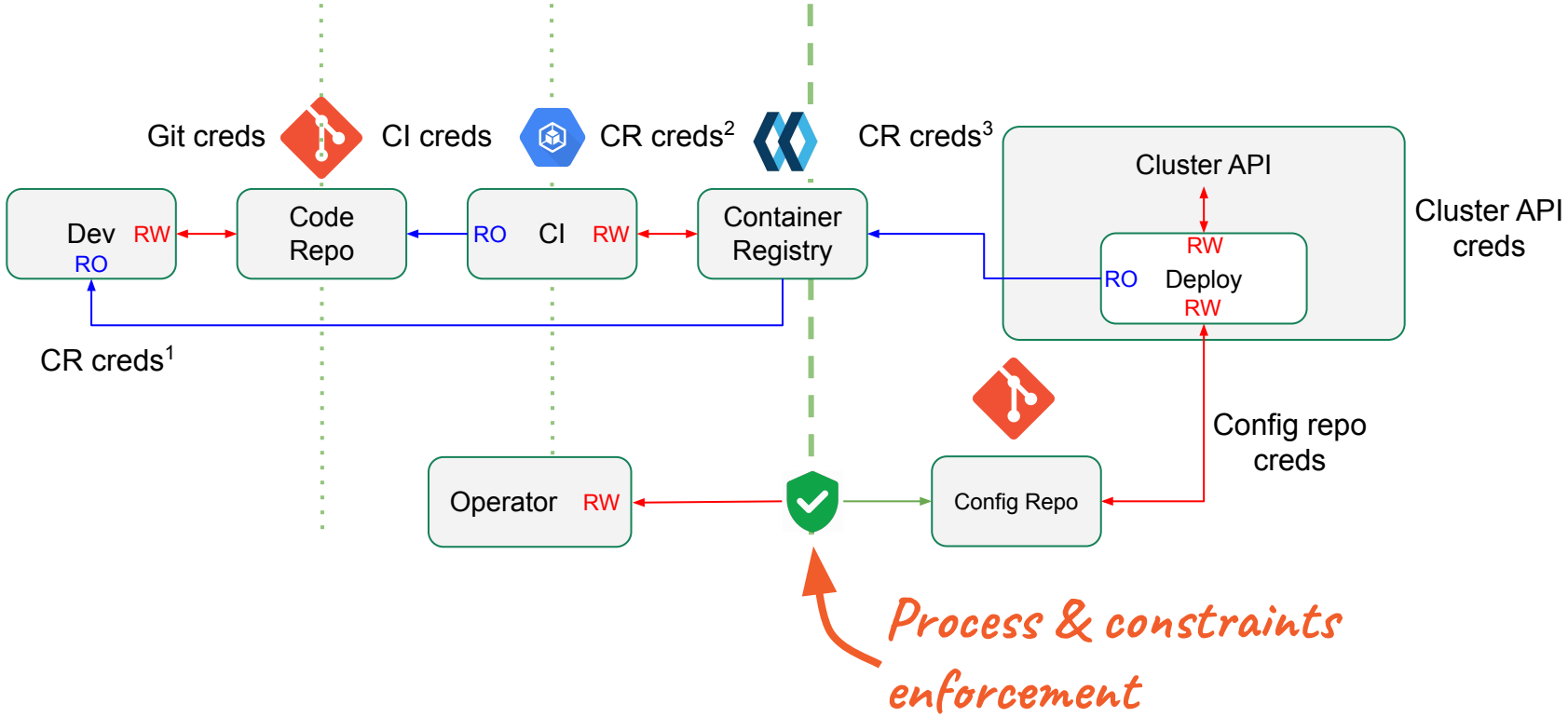
GitOps pipeline

Credentials are never shared across a logical security boundary.



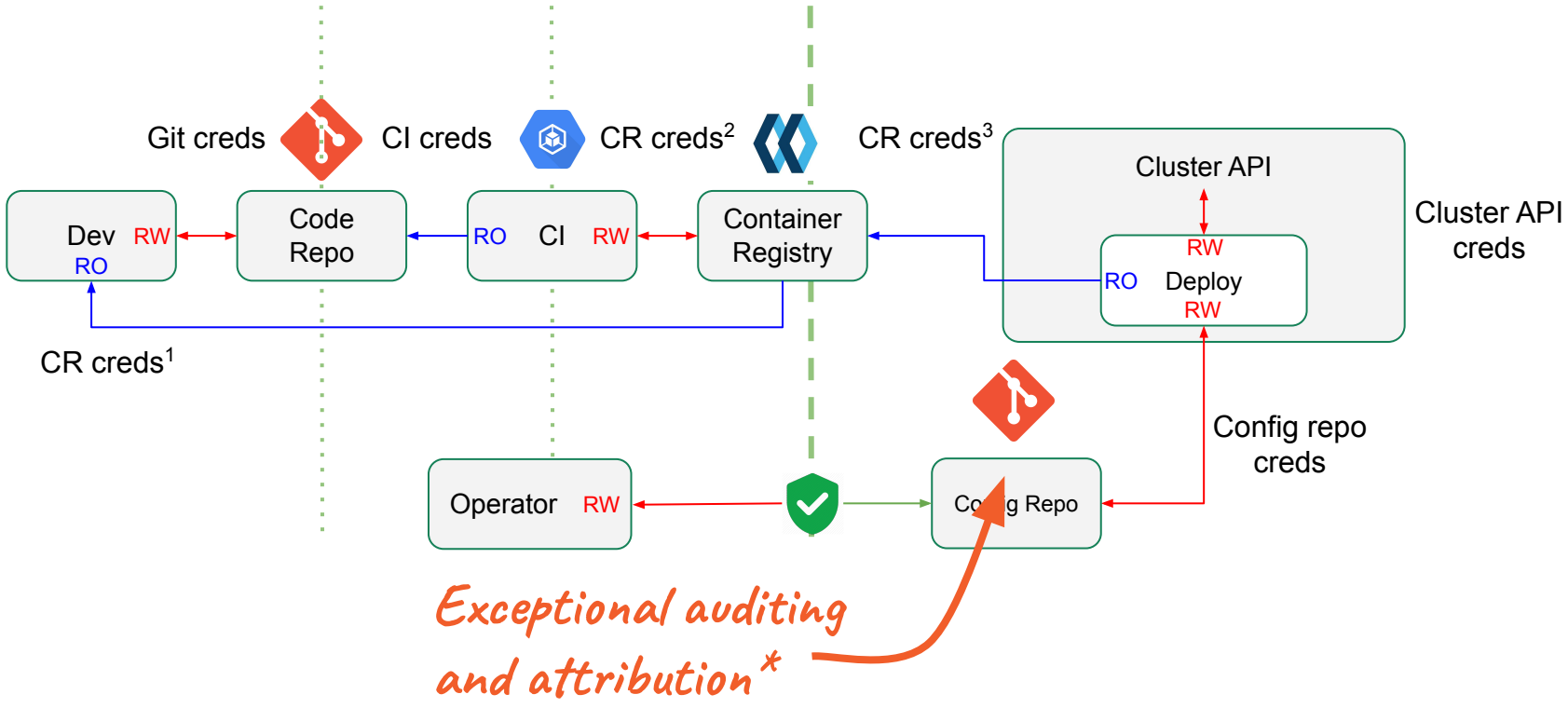
GitOps pipeline

Credentials are never shared across a logical security boundary.



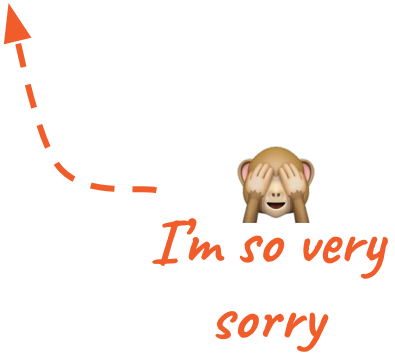
GitOps pipeline

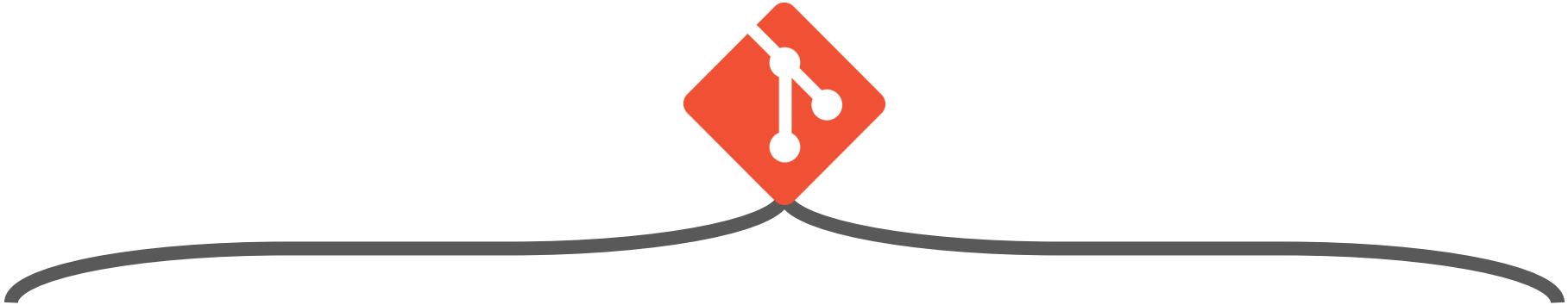
Credentials are never shared across a logical security boundary.



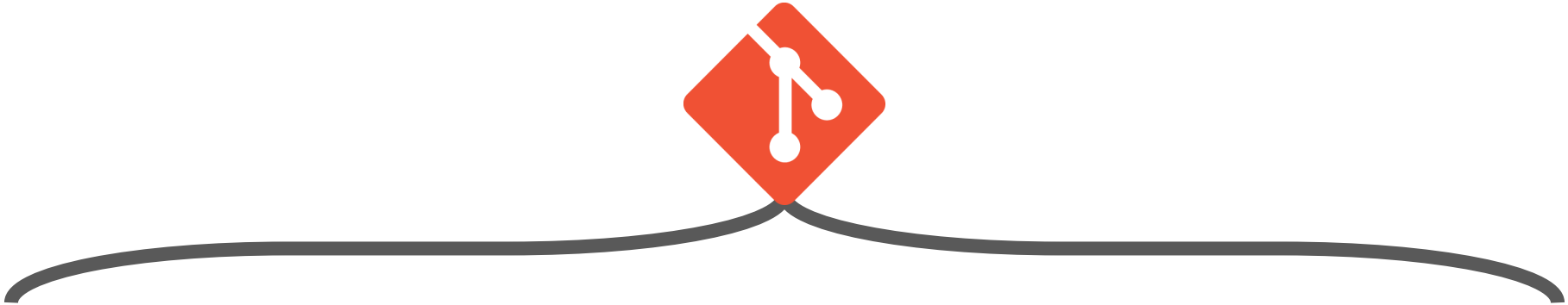
What should be GitOps'ed?

What should be GitOps'ed?





?



Kubernetes Manifests Dashboards Application checklists

Application configuration Alerts Recording Rules

Provisioning scripts Playbook Sealed Secrets

jml Fix corp alert rules 488f056 on 2 Aug

1 contributor

36 lines (32 sloc) | 1.15 KB Raw Blame History

```
# Rules for alerting on stuff related to corporate IT.
#
# All of these alerts must have 'team = "corp"' in their labels. This routes
# them to the #corporate-it channel and away from the production on-call.

# Alert when our terradiff for corp doesn't match the environment.
ALERT CorpTerradiff
  IF max(terradiff_plan_exit_code) == 2
  FOR 1h
  LABELS {
    severity = "warning",
    team = "corp",
  }
  ANNOTATIONS {
    summary = "weaveworks/corp terraform config differs from reality",
    impact = "Someone has lost access to something, or someone has access without approval",
    dashboardURL = "$${base_url}/admin/corp-terradiff",
    containerName = "terradiff",
  }
```

```
571
572  resource "github_membership" "bricef" {
573    username = "bricef"
574    role     = "member"
575  }
576
```

14:30 **AlertManager** APP

> **CorpTerradiff - weaveworks/corp terraform config differs from reality**
> Someone has lost access to something, or someone has access without approval
> Dashboard

Why should we care?

- Trivialises rollbacks
- Exceptional auditing and attribution*
- Separation of concerns
- No crossing security boundary
- Process & constraints enforcement
- Great Software ↔ Human collaboration point
- Easy to validate for correctness (Policies)
- System can self heal

Why should we care?

- Trivialises rollbacks
- Exceptional auditing and attribution*
- Separation of concerns
- No crossing security boundary
- Process & constraints enforcement
- Great Software ↔ Human collaboration point
- Easy to validate for correctness (Policies)
- System can self heal

* If you've secured your Git repositories properly

Deploying a service the GitOps Way

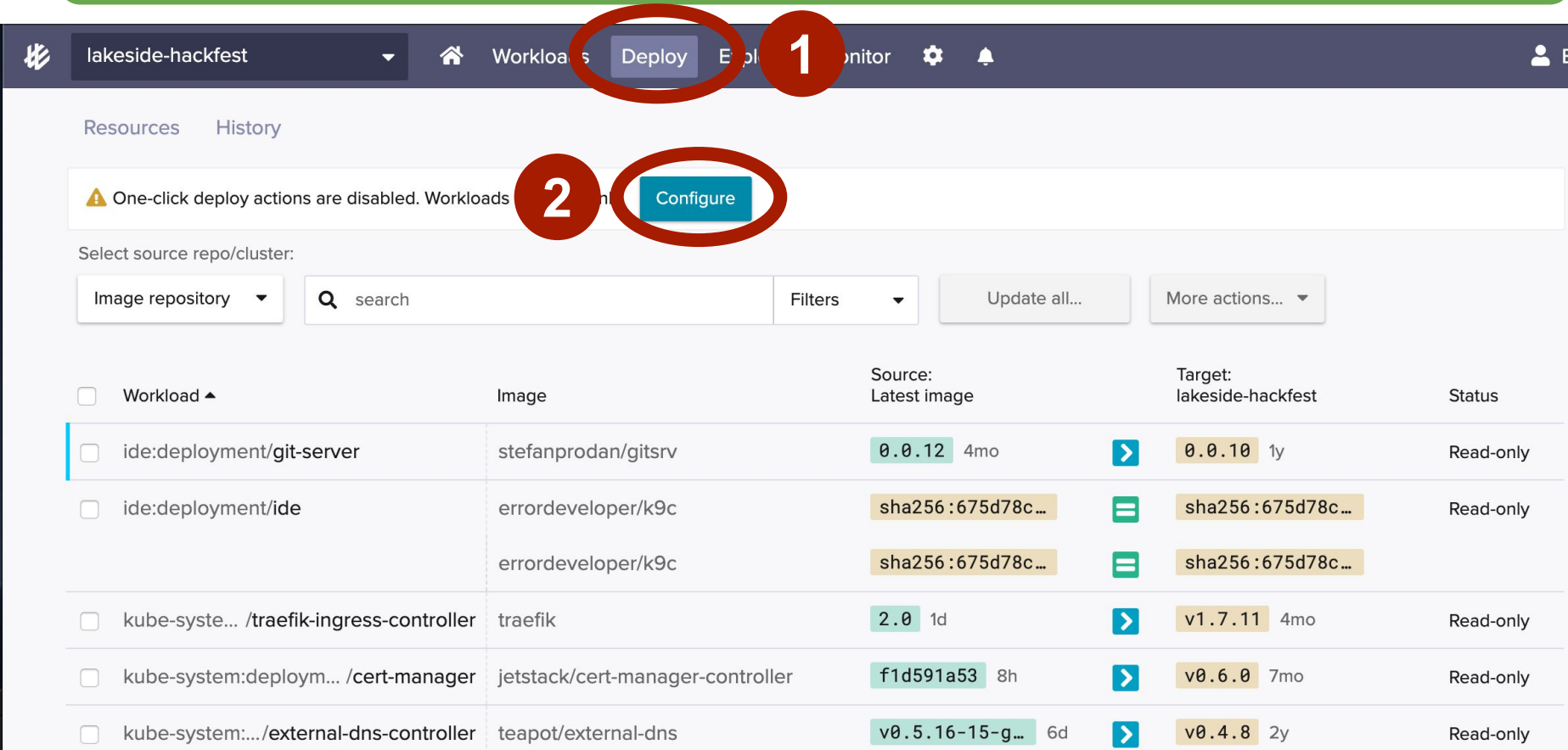
1 – Fork the sample repository

Fork the configuration repository

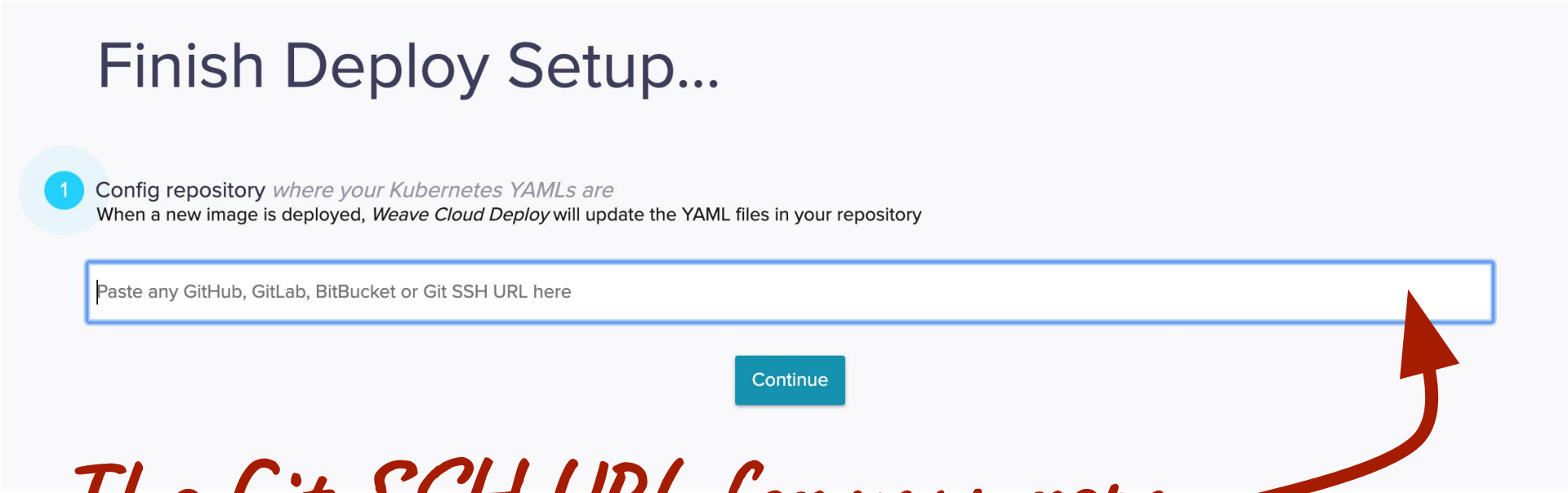
https://github.com/bricef/podinfo-deploy

To your own Github or Bitbucket

2 – Configure Flux to point to your repo



3 – Configure the agents to watch the repo



The Git SSH URL for your repo

4 – Configure the agents to watch the repo

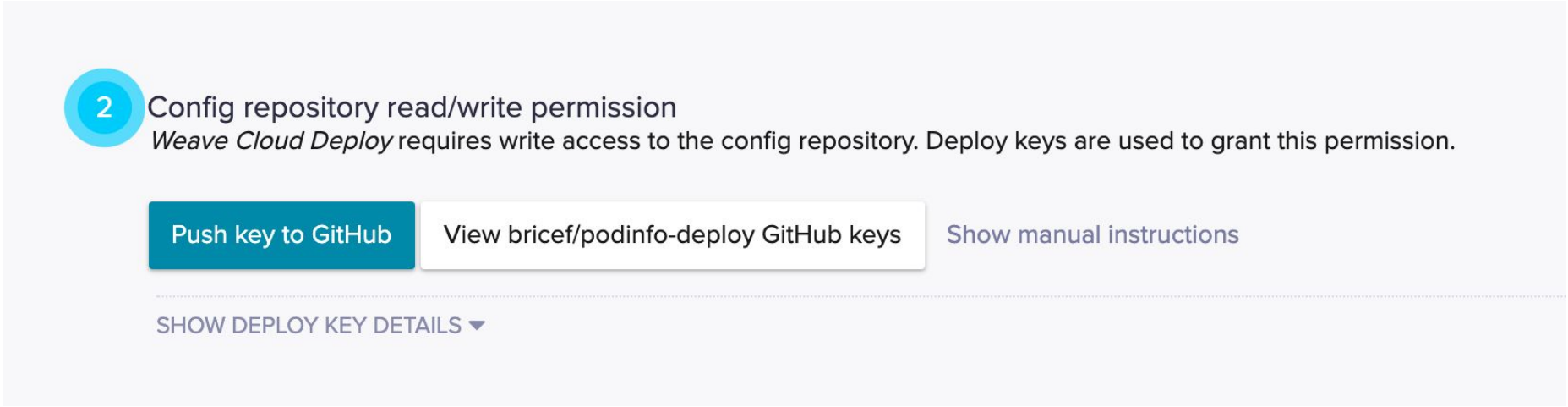
Finish Deploy Setup...

1 Config repository *where your Kubernetes YAMLs are*
When a new image is deployed, *Weave Cloud Deploy* will update the YAML files in your repository

SSH URL		Path to YAML files *optional*		Branch *optional*
git@github.com:bricef/podinfo-deploy	/	deploy	:	master

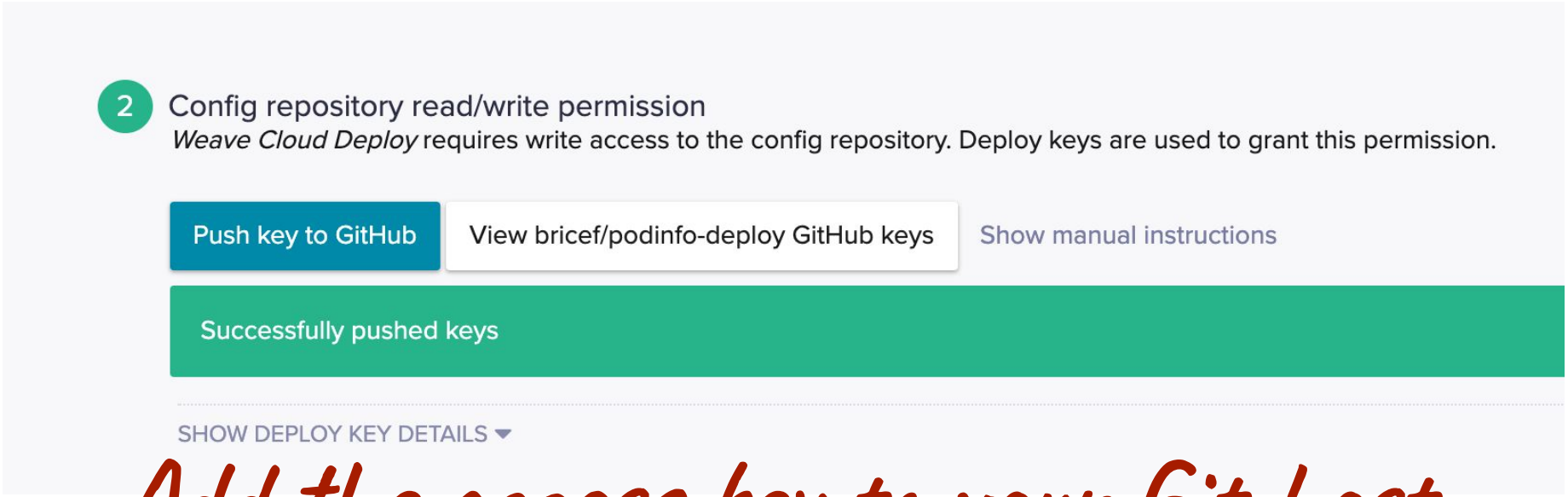
The path 'deploy'

5 – Configure the agents to watch the repo



Add the access key to your Git host

5 – Configure the agents to watch the repo



Add the access key to your Git host

6 – Configure the agents to watch the repo

3 Update cluster

To save the above changes to the config repository, apply it to your cluster:

```
kubectl apply -f "https://cloud.weave.works/k8s/flux.yaml?k8s-version=$(kubectl version | base64 | tr -d '\n')&
t=5987p5dc4ajf7rjyx1coin674598mjmw&flux-version=%5E1&git-label=flux-twilight-lake-63&git-path=deploy&
git-url=git%40github.com%3Abricef%2Fpodinfo-deploy"
```

Copy to clipboard

Copy and run the command into your Cloud 9 terminal

3 – Check that the service is running

Look at what's running on the cluster with

```
➤ watch kubectl get -n dev all
```

The podinfo application should be running in your cluster in the dev namespace

Checkpoint:

Podinfo running in the cluster

So what actually happened?

1. We installed the ‘Flux’ agent on our cluster
2. We added the agent key to our repository (so it can read and write the configuration)
3. We configured the agent to watch to our repository
4. The agent noticed some manifests in the repository, and applied them
5. Kubernetes deployed the manifests

* If you’ve secured your Git repositories properly

Flux

github.com/fluxcd/flux

Speed everything up with webhooks

By default, Flux will poll the Git repository

Things are faster with Webhooks. ***If you're using Gtihub***, configure webhooks.

Coffee Time

Gitops in practice

Exercise: Revert back to a previous version

Either by committing directly to the repository, or by using the deploy UI in Weave Cloud, deploy a previous version of **podinfo**.

How does this compare with your current rollback time?

Is this a realistic rollback time for a production system?

What could slow the rollback down?

What could make it impossible?

Exercise: Automate a deployed service

In the Weave Cloud UI, automate the release of **podinfo**.

What commit does this create in your control repository?

Deautomate **podinfo** using the UI**.**

Now, without using the UI, configure **podinfo** to be continuously deployed with a git commit.

Exercise: Deploy a new service

A developer has made an image available to deploy on the cluster:

quay.io/brice/metrics-demo

Using the podinfo configuration as a template, expose this service to the outside world, *without using kubectl*. Also make sure to use the **v4-subscription-service** version.

You do not need to specify a command or startup arguments.

You will need to know that this image exposes a service on **port 3000**.

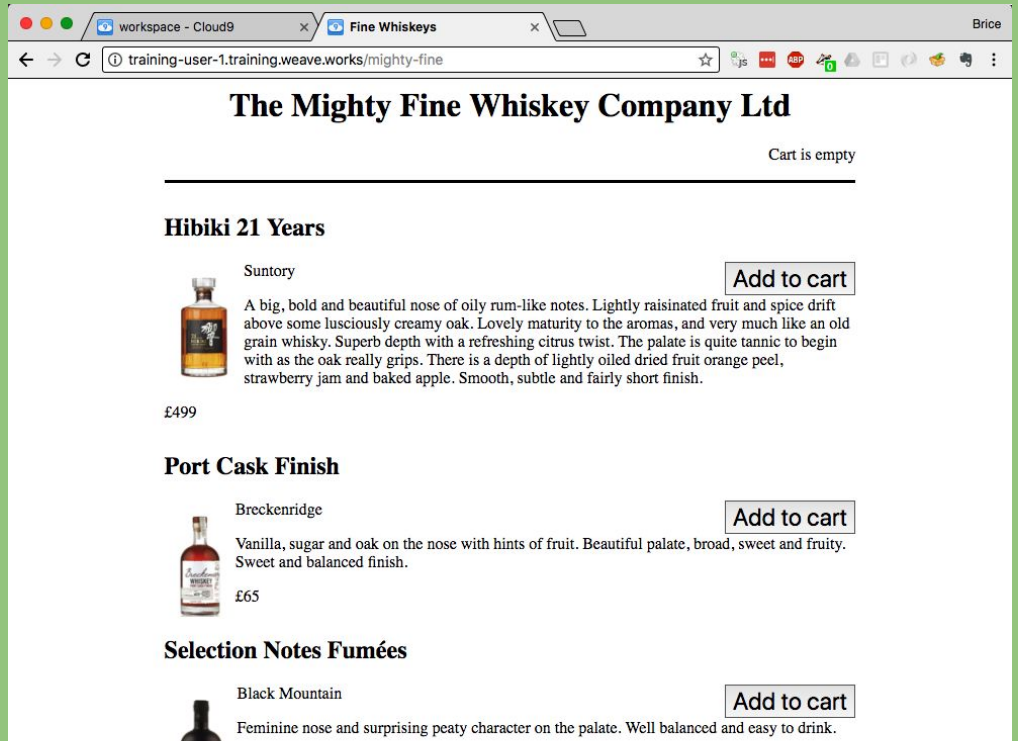
Exercise: Deploy a new service (HINTS)

```
├── README.md
└── deploy
    ├── dev-cert-issuer.yaml
    ├── dev-ns.yaml
    ├── dev-tls-cert.yaml
    └── podinfo
        ├── podinfo-dep.yaml
        ├── podinfo-ingress.yaml
        └── podinfo-svc.yaml
```

```
├── README.md
└── deploy
    ├── dev-cert-issuer.yaml
    ├── dev-ns.yaml
    ├── dev-tls-cert.yaml
    ├── mighty-fine
    │   ├── mf-dep.yaml
    │   ├── mf-ingress.yaml
    │   └── mf-svc.yaml
    └── podinfo
        ├── podinfo-dep.yaml
        ├── podinfo-ingress.yaml
        └── podinfo-svc.yaml
```

Add the new configuration in its own directory

Checkpoint: Service exposed to outside world



End of Day Recap

Agenda

14:00	**Welcome & introduction**
14:15	**Getting started with your environment**
14:30	**What is GitOps**
15:00	**Deploying a service the GitOps way**
15:15	**Break (30 minutes)**
15:45	**GitOps in practice**
16:45	**Review and recap**
17:00	**End of session**

1. The entire system is described declaratively.
2. The canonical desired system state is versioned (with Git)
3. Changes to the desired state are automatically applied to the system
4. Software agents ensure correctness and alert on divergence

Recap: GitOps CI/CD

- Having separate pipelines for CI and CD enables better security
- It's also easier to deal with if a deployment goes wrong
- We built a few versions of a simple app, using a demo CI pipeline
- Deployed those versions to Kubernetes using Weave Cloud
- Automated the deployment
- Deployments, rollback and lock are all done via git
- Git is our single source of truth.

Questions?

Weaveworks
https://weave.works
@weaveworks
@fractallambda
brice@weave.works

BBQ Time