JUSTUS-LIEBIG-UNIVERSITÄT GIESSEN

de.NBI
GERMAN NETWORK FOR BIOINFORMATICS INFRASTRUCTURE

CLOUD USER 24 MEETING

Advanced Kubernetes
Day 1: GitOps

Sebastian Beyvers | Bioinformatics & Systems Biology | JLU Giessen
sebastian.beyvers@cb.jlug.de

02.12.2024

Timetable

Day 1: Introduction to GitOps

- What is GitOps ?
- Why GitOps?
- Tool comparison (ArgoCD, Flux, ..)
- Security considerations
- Hands on!

Day 2: Running Workflows on Kubernetes

- Container Native Workflows
  - Architectural Patterns
  - Resource management
- Nextflow on Kubernetes
- (Snakemake)

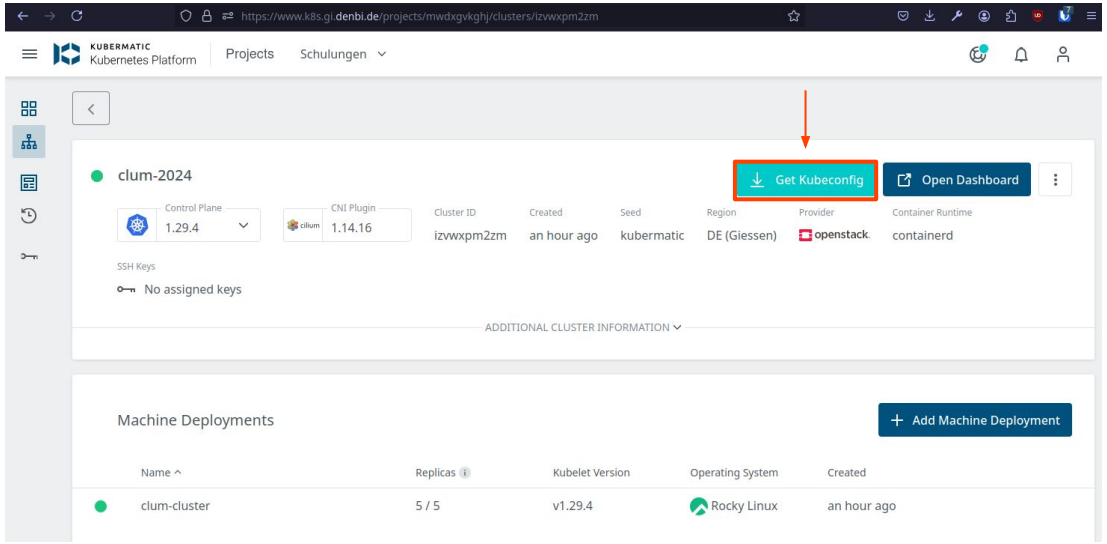
Day 3: Emerging topics

- Gateway API
- Data on Kubernetes
- Kueue
- (Q & A)

Setup

- Login @ https://www.k8s.gi.denbi.de
- Send me your LS login:
  - sebastian.beyvers@cb.jlug.de
- Download .kube/config
- Make sure that Kubectl is working

  > kubectl get pods
- Git setup



GitOps Schematic Overview

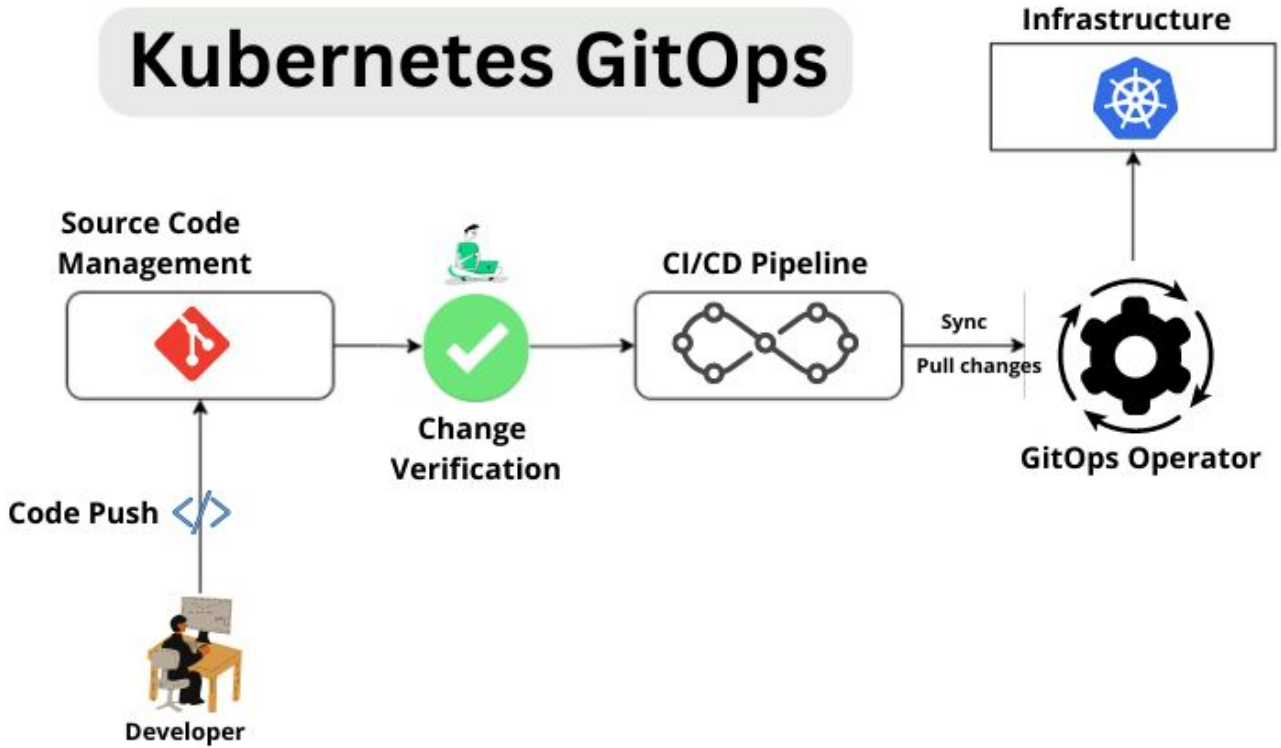


Fig 2: Schematic of a GitOps workflow: source

Why ?

- Version control of declarative manifests (Pods, Deployments etc.)
- Reproducible cluster independent deployment
- Advanced rollback functionalities
- Automated updates
- Reduced yaml management overhead
  → reusable “blueprints” across instances

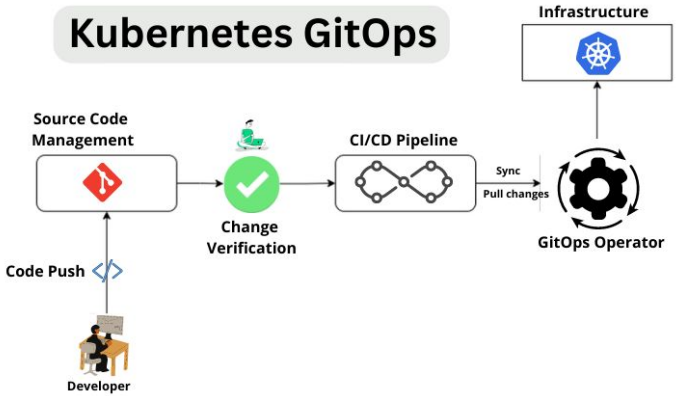


Fig 2: Schematic of a GitOps workflow: source

Tool comparison

vs.

- Cli first approach
- Easier to bootstrap
- Simpler configuration
- No built-in web-ui
- Better Helm integration

- Web & Cli first approach
- Sophisticated plugin system
- More widespread adoption
- Better multi-tenancy
- Better cross-cluster support
- SSO integration

Tool comparison

flux

```yaml
apiVersion: kustomize.toolkit.fluxcd.io/v1
kind: Kustomization
metadata:
  name: apps
  namespace: flux-system
spec:
  interval: 10m0s
  dependsOn:
    - name: apps-configs
  sourceRef:
    kind: GitRepository
    name: flux-system
  path: ./clusters/prod/giessen/foobar
  wait: true
  prune: false
  timeout: 3m0s
```

argo

```yaml
apiVersion: argoproj.io/v1alpha1
kind: Application
metadata:
  name: bakta
  namespace: argocd
spec:
  project: bakta
  source:
    repoURL: >-
      https://git.computational.bio.uni-giessen.de/
    targetRevision: HEAD
    path: projects/bakta
    plugin:
      name: argocd-vault-plugin
  destination:
    server: 'https://kubernetes.default.svc'
    namespace: bakta
  syncPolicy:
    automated:
      allowEmpty: true
      selfHeal: true
    syncOptions:
      - CreateNamespace=true
```

Secret Management

SOPS: Secrets OPerationS

- User specific secrets
  - Secrets will reside in cluster!
- Local encryption / decryption
- No K8s operator
- Built-in support in Flux / Argo
- Support for external secret manager (Vault, KMS, AKV)

Sealed Secrets

- Cluster specific operator
- Private keys only in cluster
- Built-in key-rotation
- No GPG etc. needed

Hands on !