



Everything is Code

Embracing GitOps at Spotify

Tim Hansen, Senior Engineer
[@timbonicus](#)



What is GitOps?

Infrastructure automation using files
stored in source control





Wait isn't that just IaC?

GitOps **applies** infrastructure-as-code (IaC)

- **Must** use source control
- **Must** use code review processes
- Infrastructure changes **must** go through Git



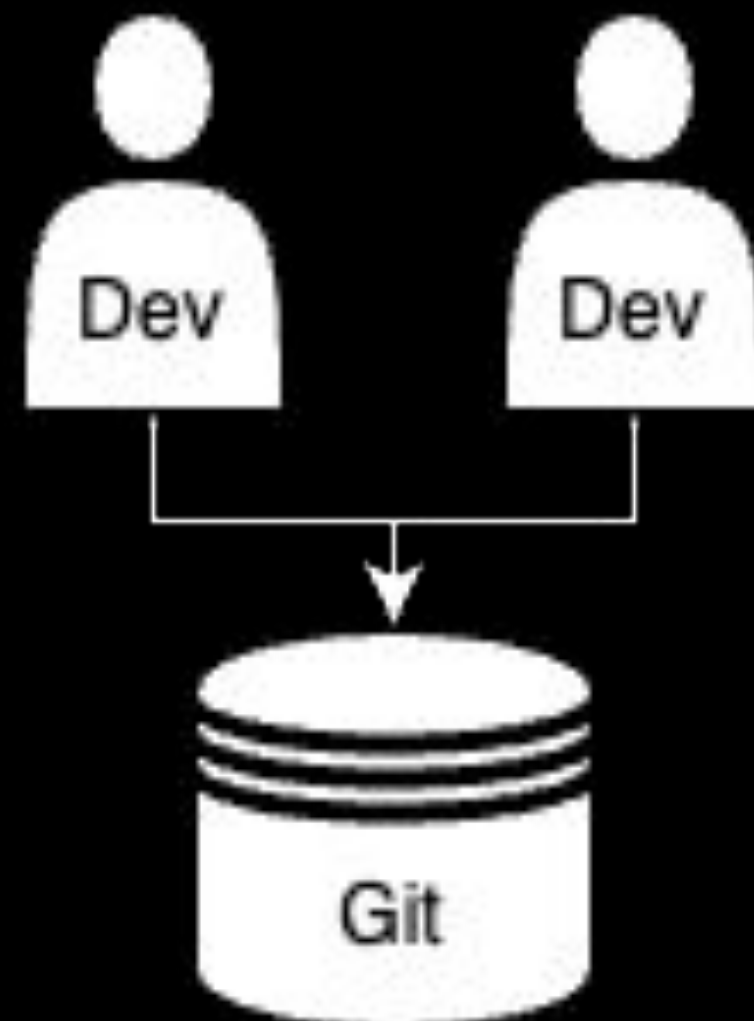
GitOps principles

- Desired state is expressed declaratively
- Declarations are versioned and immutable
- Changes are applied automatically

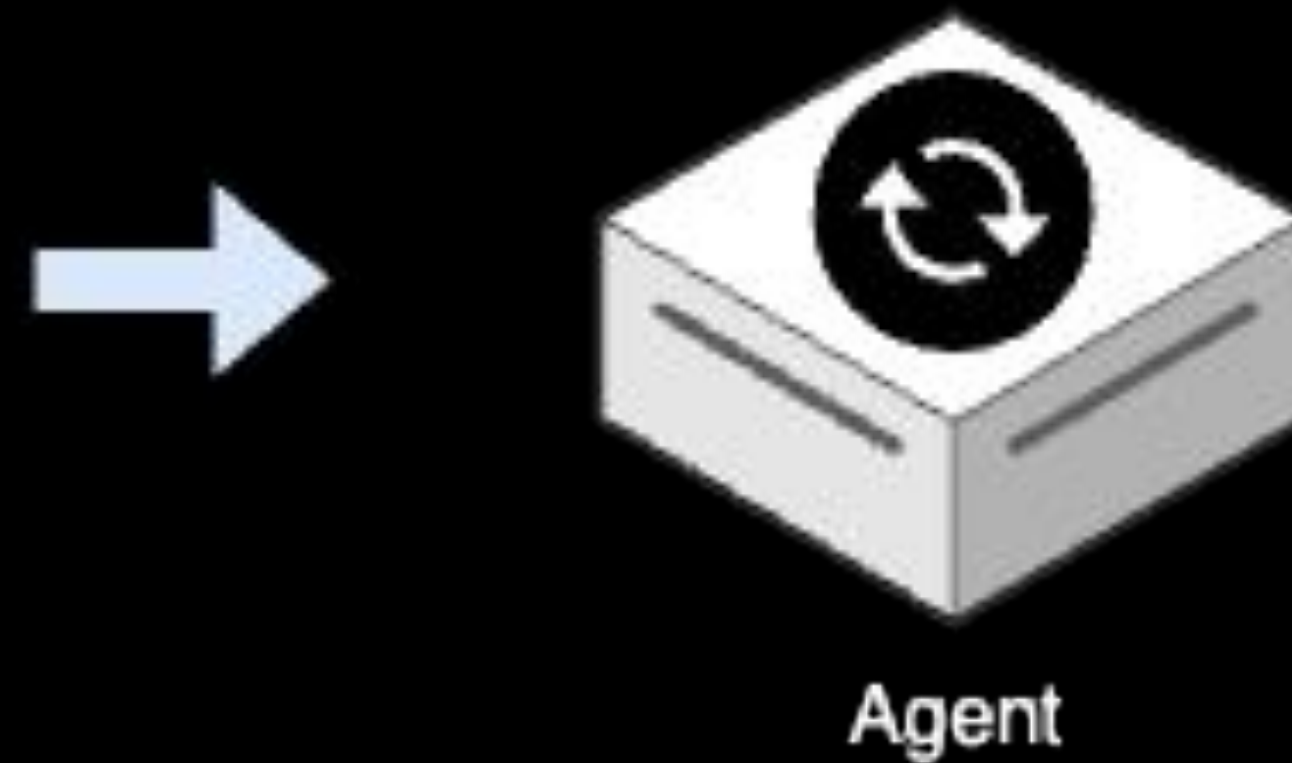


What is GitOps?

Review and commit code



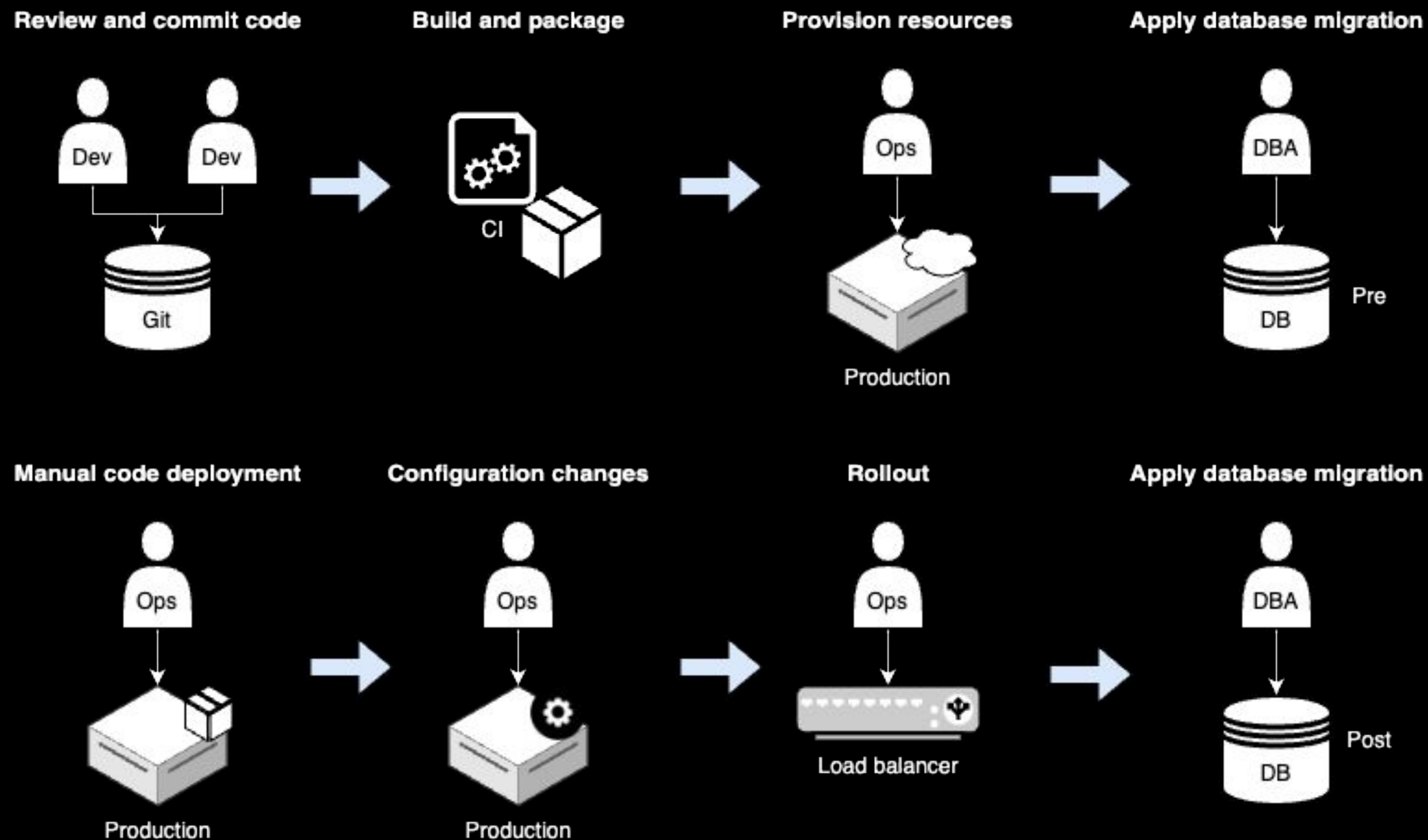
Compare state



Changes applied



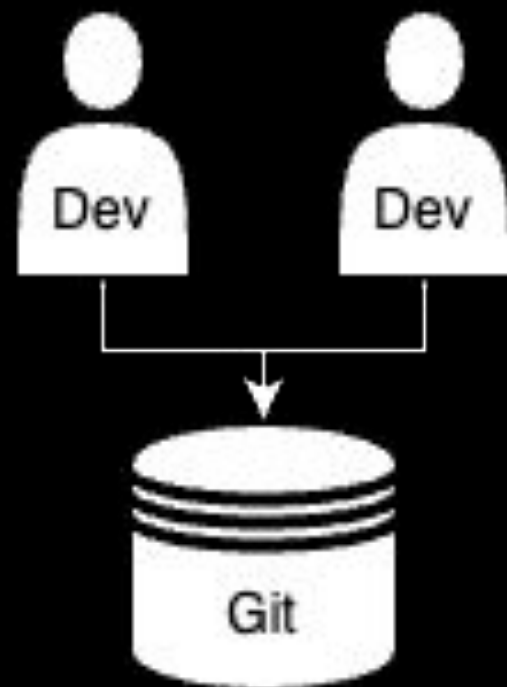
Typical release at company X





Typical release at Spotify

Review and commit code



Build and package



Changes synced



Automatic migration



New pods created



Automated canary rollout





GitOps benefits

- Automation, scale, speed
- Change review process
- Version history
- Repository permissions
- Easy rollback





GitOps at Spotify





GitOps: Builds

Tingle

- Central build system
- Jenkins pipelines + action containers
- Declarative language
- Triggered by GitHub webhooks



GitOps: Builds

```
version: "2"
useBuildIdentity: true
pipelines:
- pipeline: "Review build"
  ref: "pr/./merge"
  stages:
  - stage: "Build"
    tasks:
    - task: "maven"
      steps:
      - action:
"action-containers/mvn-java17:latest"
        args:
        - "clean"
        - "verify"
- pipeline: "Master build"
  ref: "master"
  stages:
```

```
- stage: "Build"
  tasks:
  - task: "maven"
    steps:
    - action:
"action-containers/mvn-java17:latest"
      args:
      - "clean"
      - "deploy"
- stage: "Deployment - production"
  tasks:
  - task: "deploy"
    steps:
    - action: "action-containers/deploy:latest"
      args:
      - "deploy example-app"
--entity-id=example-app\
  \ --path=. --git-commit=$COMMIT_SHA"
  entrypoint: "/bin/bash"
```



GitOps: Builds

```
# build-info.yaml
version: 2
template:
  type: javaBackend
```

Tingle 🤔 ✨

```
version: "2"
useBuildIdentity: true
pipelines:
- pipeline: "Review build"
  ref: "pr/./merge"
  stages:
  - stage: "Build"
    tasks:
    - task: "maven"
      steps:
      - action: "action-containers/mvn-java17:latest"
        args:
        - "clean"
        - "verify"
- pipeline: "Master build"
  ref: "master"
  stages:
  - stage: "Build"
    tasks:
    - task: "maven"
      steps:
      - action: "action-containers/mvn-java17:latest"
        args:
        - "clean"
        - "deploy"
  - stage: "Deployment - production"
    tasks:
    - task: "deploy"
      steps:
      - action: "action-containers/deploy:latest"
        args:
        - "deploy example-app --entity-id=example-app\
          \ --path=. --git-commit=$COMMIT_SHA"
      entrypoint: "/bin/bash"
```




GitOps: Builds

```
# build-info.yaml
version: 2
template:
  type: javaBackend
  javaBackend:
    javaVersion: 21
    parallelize: true
```



GitOps: Deployments

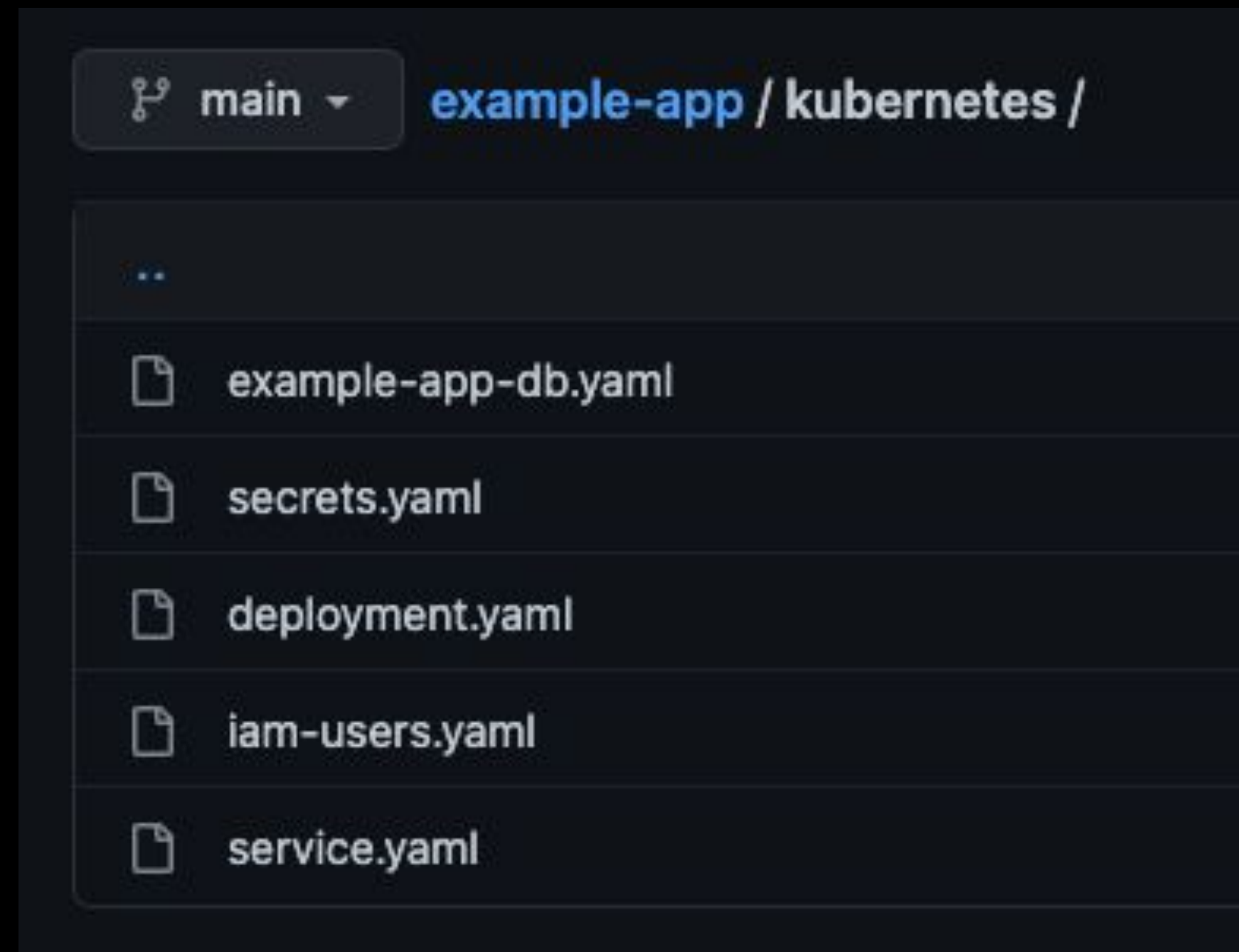
Tugboat

- Central deployment system
- Declarative language
- Triggered by Tingle



GitOps: Deployments

```
# deployment-info.yaml
version: 1
components:
- componentId: example-app
  tugboat:
    backends:
      - backend: kubernetes
```





GitOps: Deployments

```
# example-app-db.yaml

apiVersion: sql.cnrm.cloud.google.com/v1beta1
kind: SQLInstance
metadata:
  name: "example-app-db"
  namespace: "example-app"
  annotations:
    cnrm.cloud.google.com/project-id: my-project
  databaseVersion: "POSTGRES_14"
  region: "europe-west1"
  settings:
    diskType: "PD_SSD"
    availabilityType: "REGIONAL"
    backupConfiguration:
      backupRetentionSettings:
        retainedBackups: 30
```

```
# iam-users.yaml

apiVersion: iam.cnrm.cloud.google.com/v1beta1
kind: IAMPartialPolicy
metadata:
  name: "example-app-db.iam-user-auth"
  namespace: "example-app"
spec:
  resourceRef:
    apiVersion: resourcemanager.cnrm.cloud.google.com/v1beta1
    kind: Project
    external: "my-project"
  bindings:
    - role: "roles/cloudsql.instanceUser"
      members:
        - member: "serviceAccount:example-sa@gserviceaccount.com"
```




GitOps: Deployments

```
# deployment-info.yaml
version: 1
components:
- componentId: example-app
  tugboat:
    backends:
    - backend: kubernetes
    automatedCanary:
      canaryService: example-canary
      steps:
      - analysis:
          templates:
            - templateName: javaBackend
```



GitOps: Monitoring

Grafana

- Central monitoring system
- Declarative language
- Triggered by Tingle



GitOps: Monitoring

```
# monitoring-info.yaml
metadata:
  schemaVersion: 5

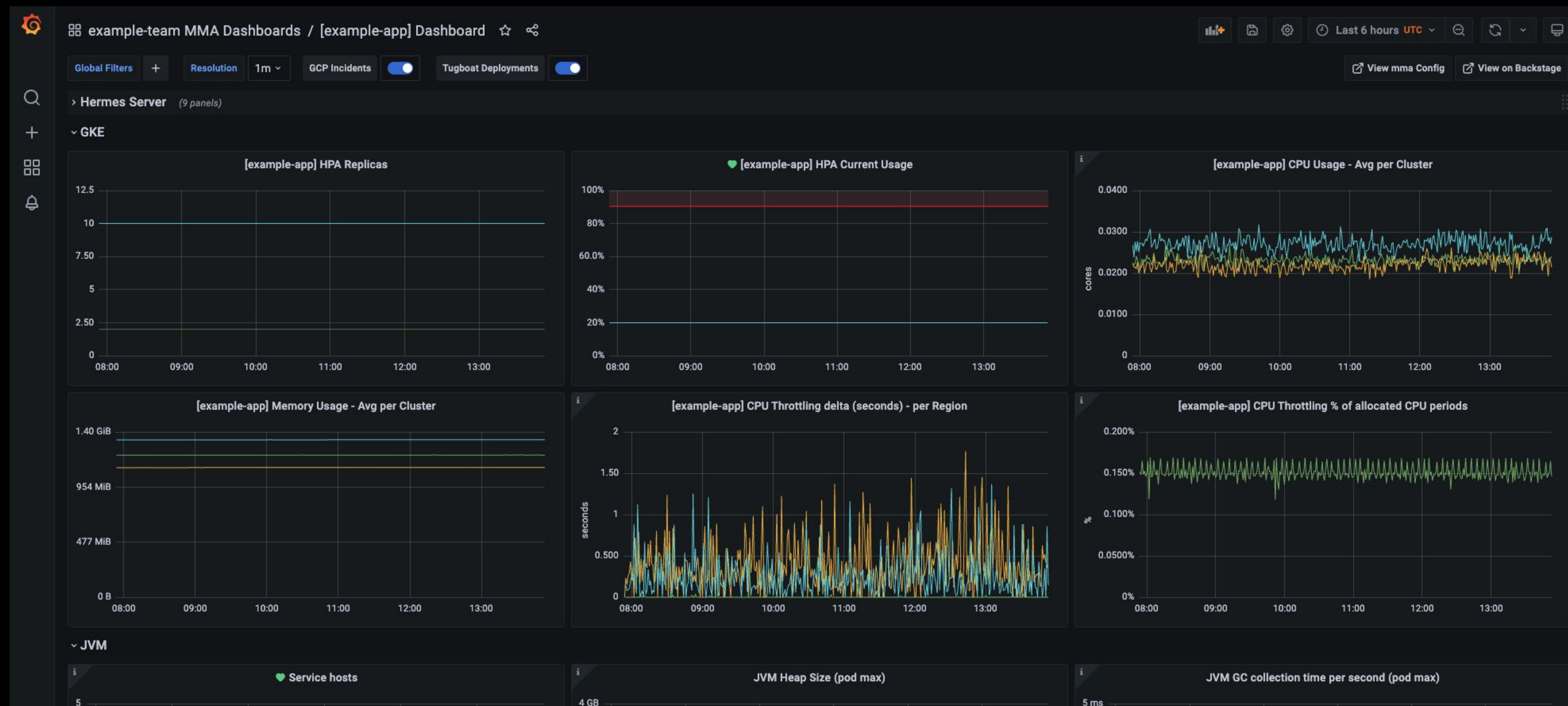
components:
  example-app:
    - name: "[example-app] Dashboard"
      panels:
        - template: javaBackend
          type: bundle
```

MMA 🤔 ✨

```
{
  "queries": {
    "A": {
      "filter": [
        "and",
        [
          "=",
          "application",
          "example-app"
        ],
        [
          "=",
          "stat",
          "1m"
        ],
        [
          "=",
          "env",
          "production"
        ],
        [
          "=",
          "grpc-component",
          "server"
        ],
        [
          "=",
          "what",
          "rpc-error-ratio"
        ]
      ],
      "aggregators": [
        {
          "type": "group",
          "of": [
            "grpc-service",
```



GitOps: Monitoring





GitOps: Monitoring

```
# monitoring-info.yaml
```

```
metadata:
```

```
  schemaVersion: 5
```

```
components:
```

```
  example-app:
```

```
    - name: "[example-app] Dashboard"
```

```
      panels:
```

```
        - template: javaBackend
```

```
          type: bundle
```

```
          exclude_hm: false
```

```
          override:
```

```
            grpc/server/rpc-latency:
```

```
              threshold_ms: 2000
```

```
        - template: http-client
```

```
          type: bundle
```

```
# custom panel
```

```
- template: custom/metric
```

```
  title: Hydration Errors
```

```
  what: hydrate.exception
```

```
  aggregation:
```

```
    category: Group By
```

```
    type: sum
```

```
    of:
```

```
      - pod
```

```
  alias: "[[tag_pod]]"
```

```
  threshold: 10
```


```
  threshold_type: gt
```

```
  window_unit: m
```


```
  window_value: 10
```

```
  action: pagerduty
```


GitOps: Monitoring



mma-feedback-... commented on Oct 1 • edited ▾



 **mma generated 1 dashboards**

Changed dashboards

- [\[example-app\] Dashboard \(preview\)](#)

Dashboard generated for commit [439ac5d](#).

This is an automated message, please see [mma-feedback](#) to learn more.



timbonicus approved these changes on behalf of **backstage/imaginary-goats** on Oct 1

[View changes](#)



GitOps: Software catalog

```
# catalog-info.yaml
apiVersion: backstage.io/v1alpha1
kind: Component
metadata:
  name: example-app
  annotations:
    spotify.com/service-tier: 4
spec:
  type: service
  lifecycle: production
  owner: team-a
  system: metadata-system
  dependsOn: ['resource:example-app-db']
  apiConsumedBy: ['component:www-artist']
```





GitOps: Software catalog

COMPONENT — SERVICE

example-app ☆

Owner
team-a

Lifecycle
experimental

OVERVIEW

CI/CD

API

DEPLOYMENT

MONITORING

DOCS

About

VIEW
SOURCE

VIEW
TECHDOCS

DESCRIPTION

Artist Lookup

OWNER

team-a

LIFECYCLE

production

SYSTEM

metadata-system

TYPE

service

SERVICE TIER

4: Low criticality

Relations

Links

PagerDuty



GitOps: Software templates

Create a new component

Create new software components using standard templates in your organization

Available Templates

PERSONAL

★ Starred

0

MY COMPANY

All

5

CATEGORIES

TAGS

service

★

Spring Boot gRPC Service

Create a simple microservice using gRPC and Spring Boot Java

recommended

java

grpc

service@example.com

CHOOSE

website

★

React SSR Template

Create a website powered with Next.js

recommended

react

web@example.com

CHOOSE

documentation

★

Documentation Template

Create a new standalone documentation project

recommended

techdocs

mkdocs

backstage/techdocs-core

CHOOSE

website

★

Create React App Template

service

★

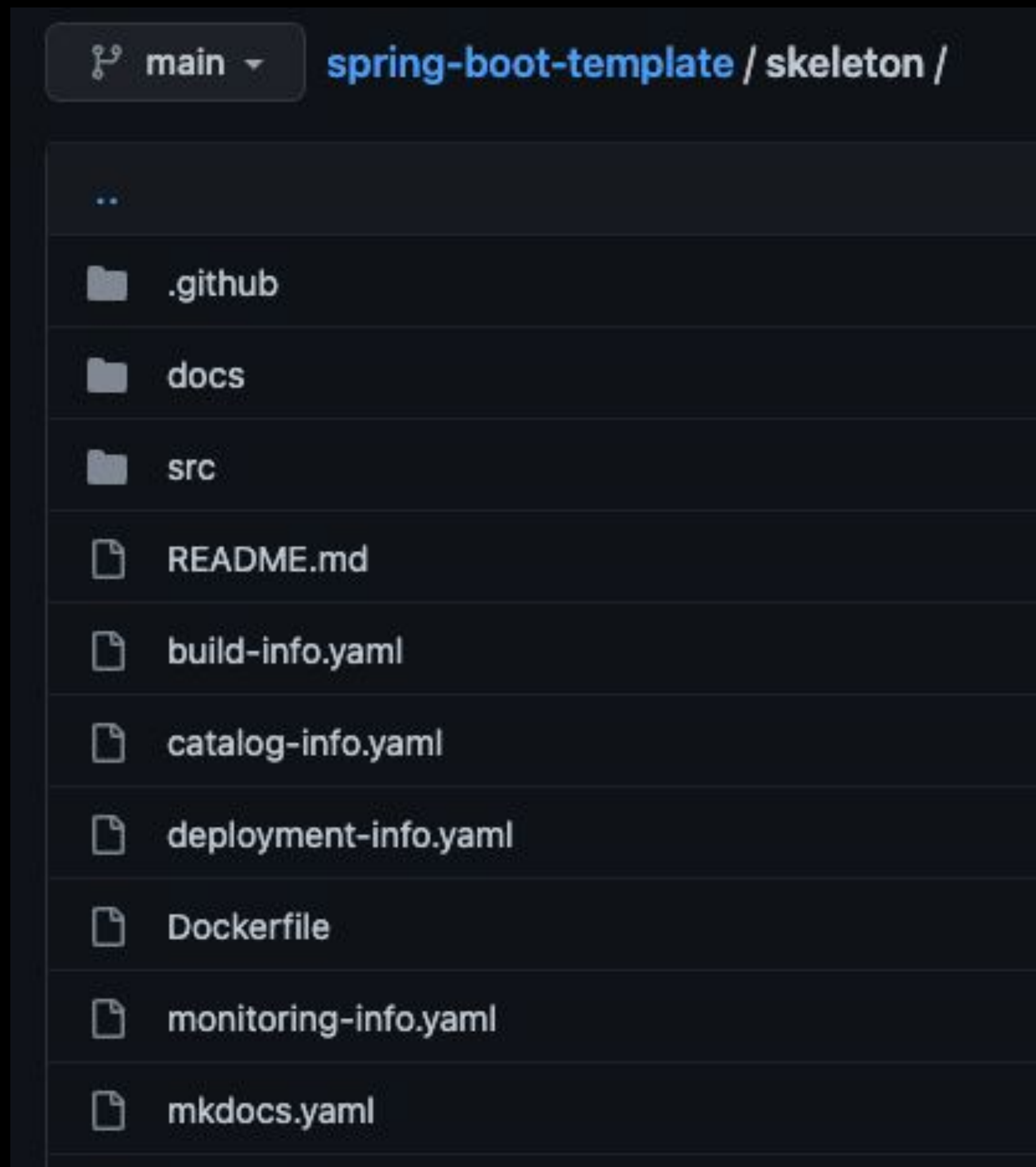
Node Service

REGISTER EXISTING COMPONENT

🔗 SUPPORT



GitOps: Software templates

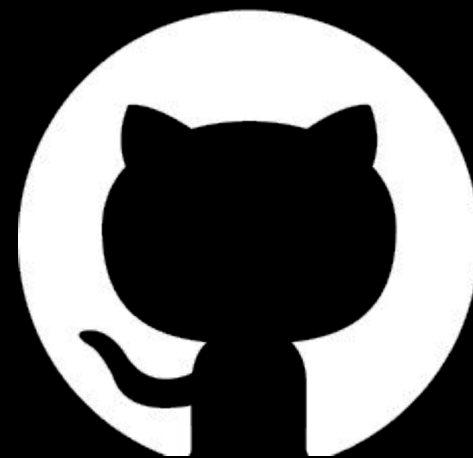




GitOps
outside Spotify

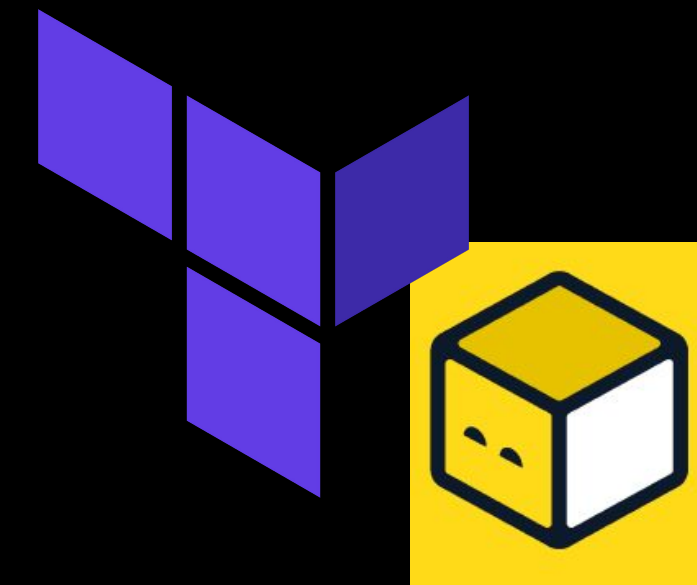
GitOps: Builds

- GitHub Actions
- GitLab CI
- Google Cloud Build
- Tekton Pipelines
- Argo Workflows
- Dagger

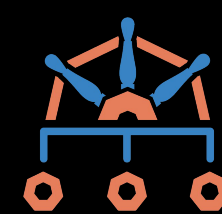


GitOps: Infrastructure

- Terraform (kinda) / OpenTofu
- Google Config Connector
- Crossplane
- Pulumi

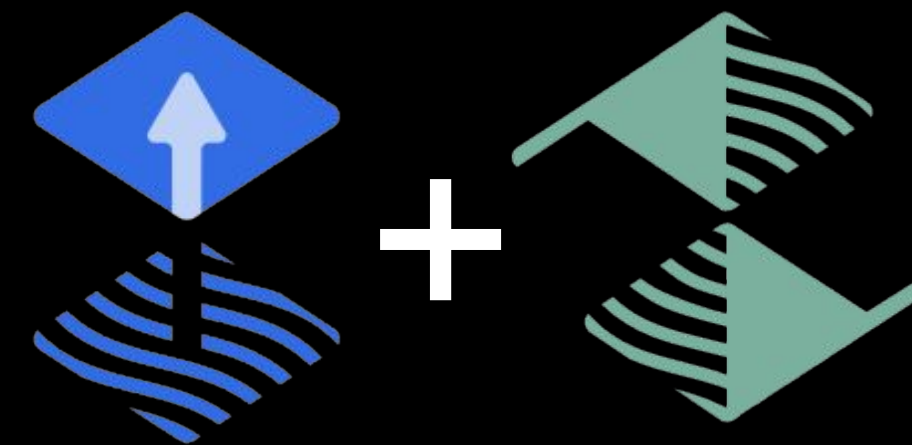


TALK TOMORROW!
BACK stack



GitOps: Deployments

- Argo CD
- Flux + Flagger
- Spinnaker (kinda)





GitOps: Monitoring

- Grafana (kinda)





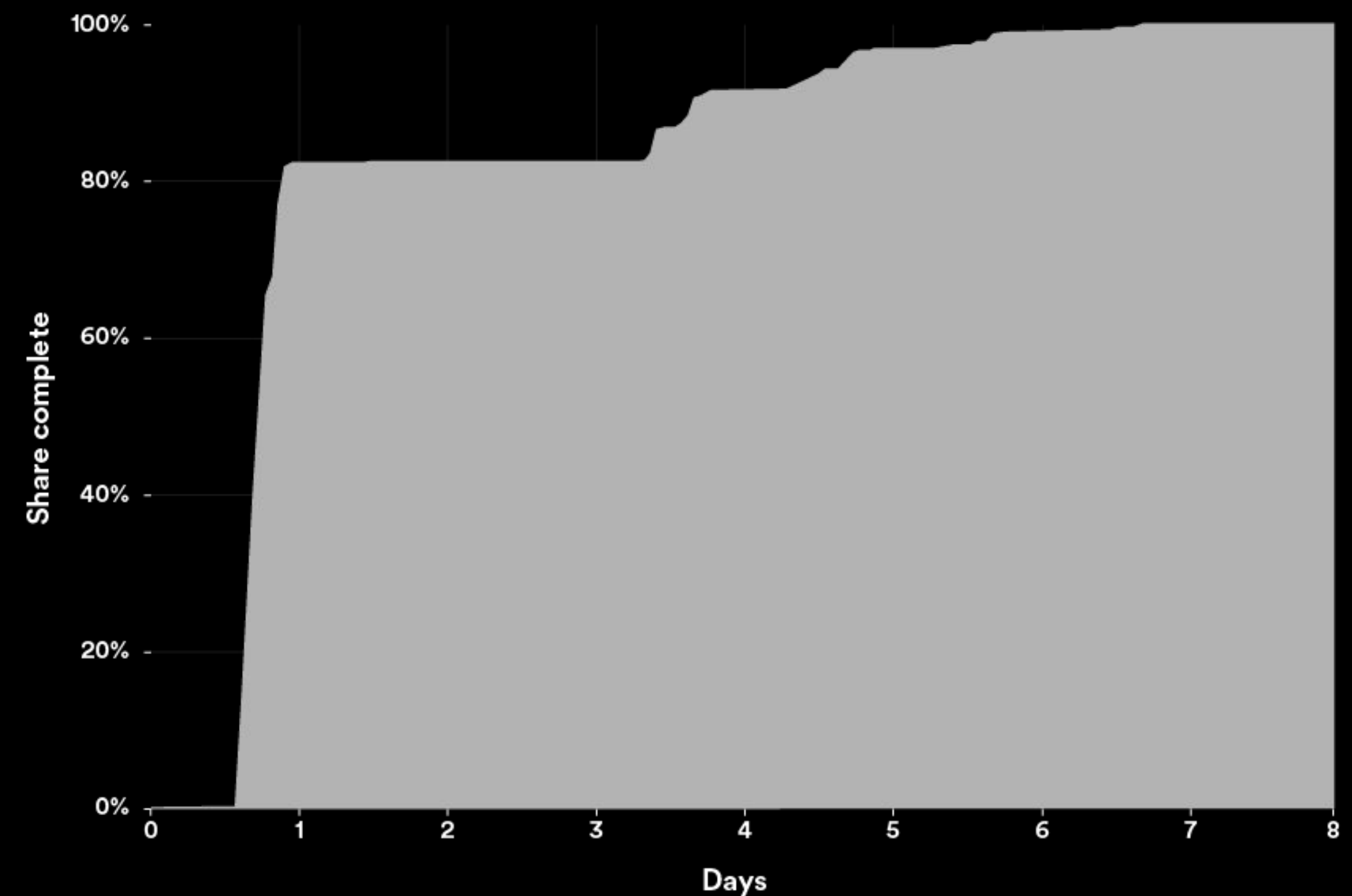
Bonus section





Fleetshift

- Apply code transformations across the entire software fleet
- Smart repository targeting
- Gradual rollouts
- Automerging
- Centralized dependency management





GitOps resources

- opengitops.dev
- gitops.tech
- github.com/weaveworks/awesome-gitops
- Tons of other talks!

