

DevOps bij Alliander

alliander

DevOps bij Alliander

Inhoud

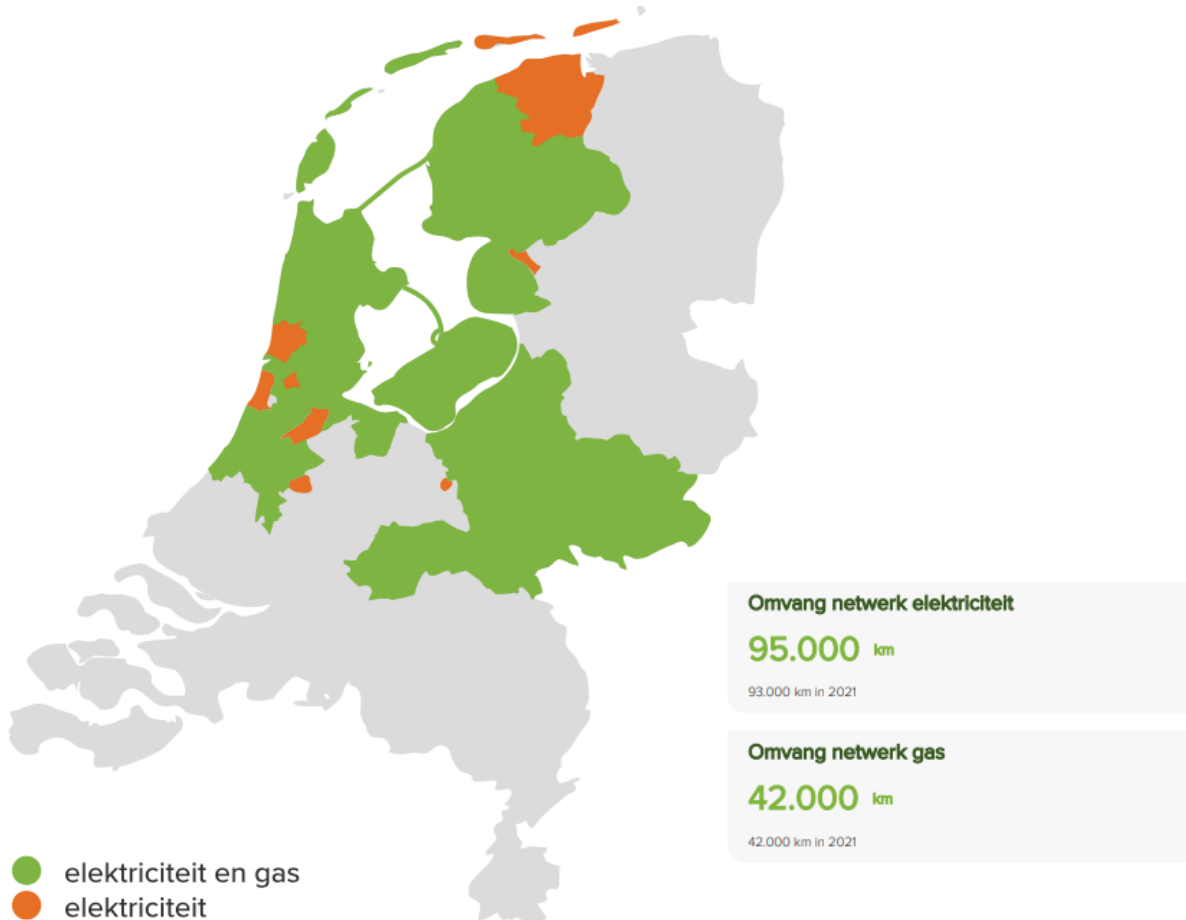


Introductie Alliander & energietransitie
Kubernetes
Helm Charts
GitOps & Operators
CI/CD met Github Actions
Demo

Netwerkbedrijf Alliander

allliander

Werkgebied



Onze organisatie



Grootste netwerkbedrijf van Nederland

in publieke handen

Actief in 6 provincies

139 gemeenten

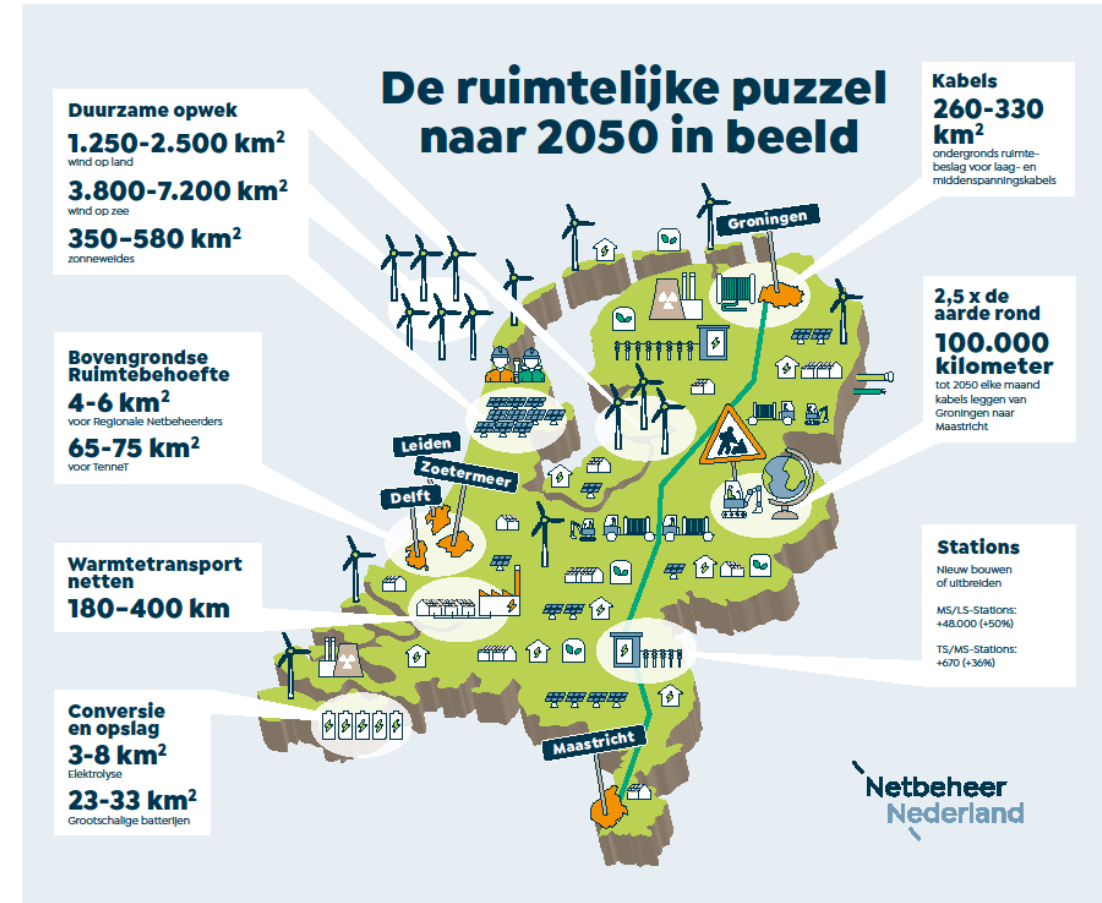
5,9 miljoen aansluitingen

Energietransitie

Een veranderend energiesysteem

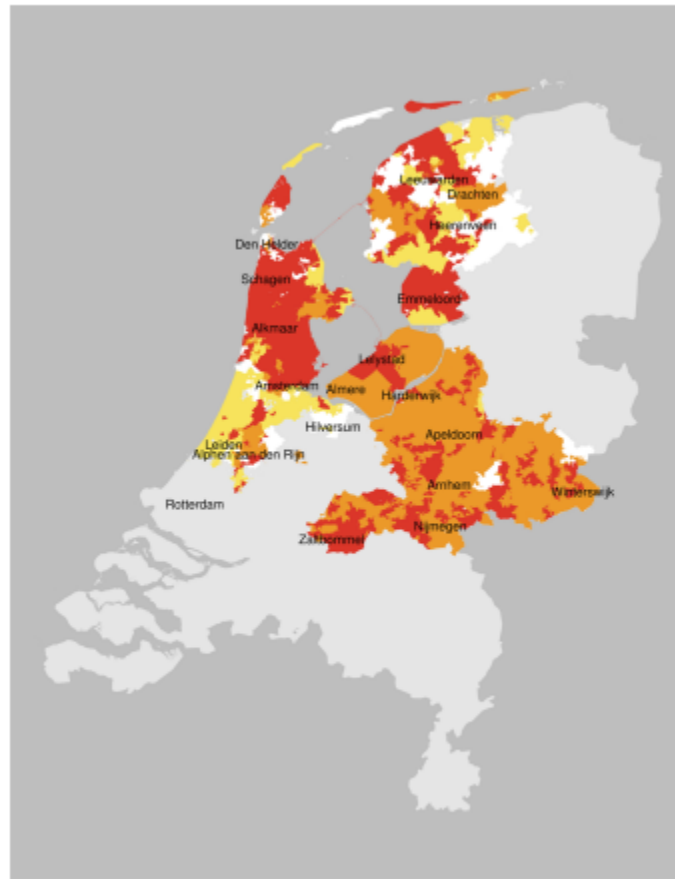
- Verduurzaming, elektrificatie industrie en bouw van woningen
- Vraag naar energie verandert en neemt toe
- Herinrichten van ons netwerk en gedragsverandering
- €8 miljard jaarlijks investeren in de energie-infrastructuur

alliander



Knelpunten in het Alliander net

Overzicht van regionale transportschaarste



Schaarste voor afname

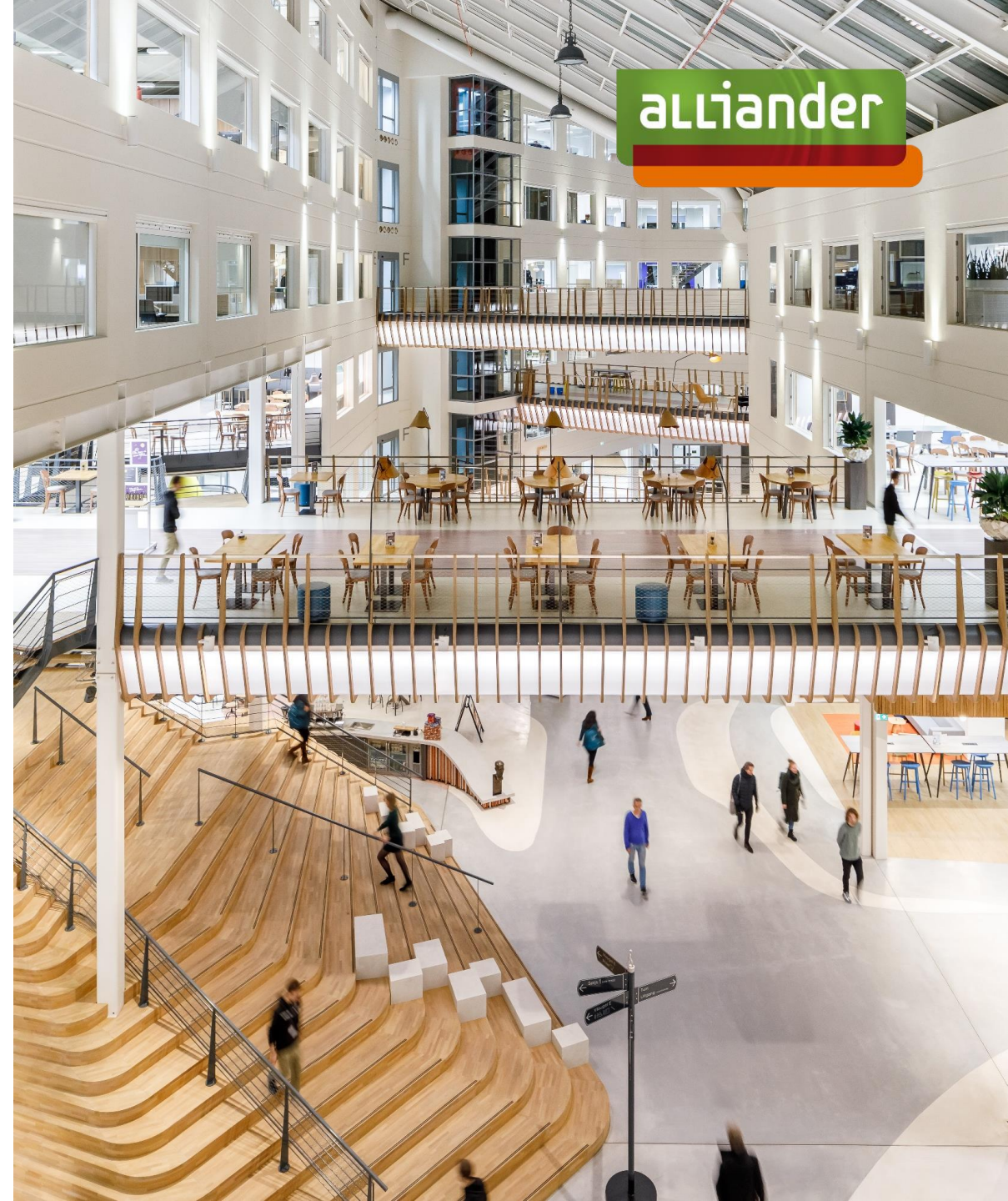


Schaarste voor opwek

Alliander

Organisatie en locaties

- *Samen werken we aan het energienet van vandaag en morgen*
- Ruim 9000 medewerkers
- O.a. Arnhem, Duiven, Amsterdam, Haarlem
- Ongeveer 1200 medewerkers in IT



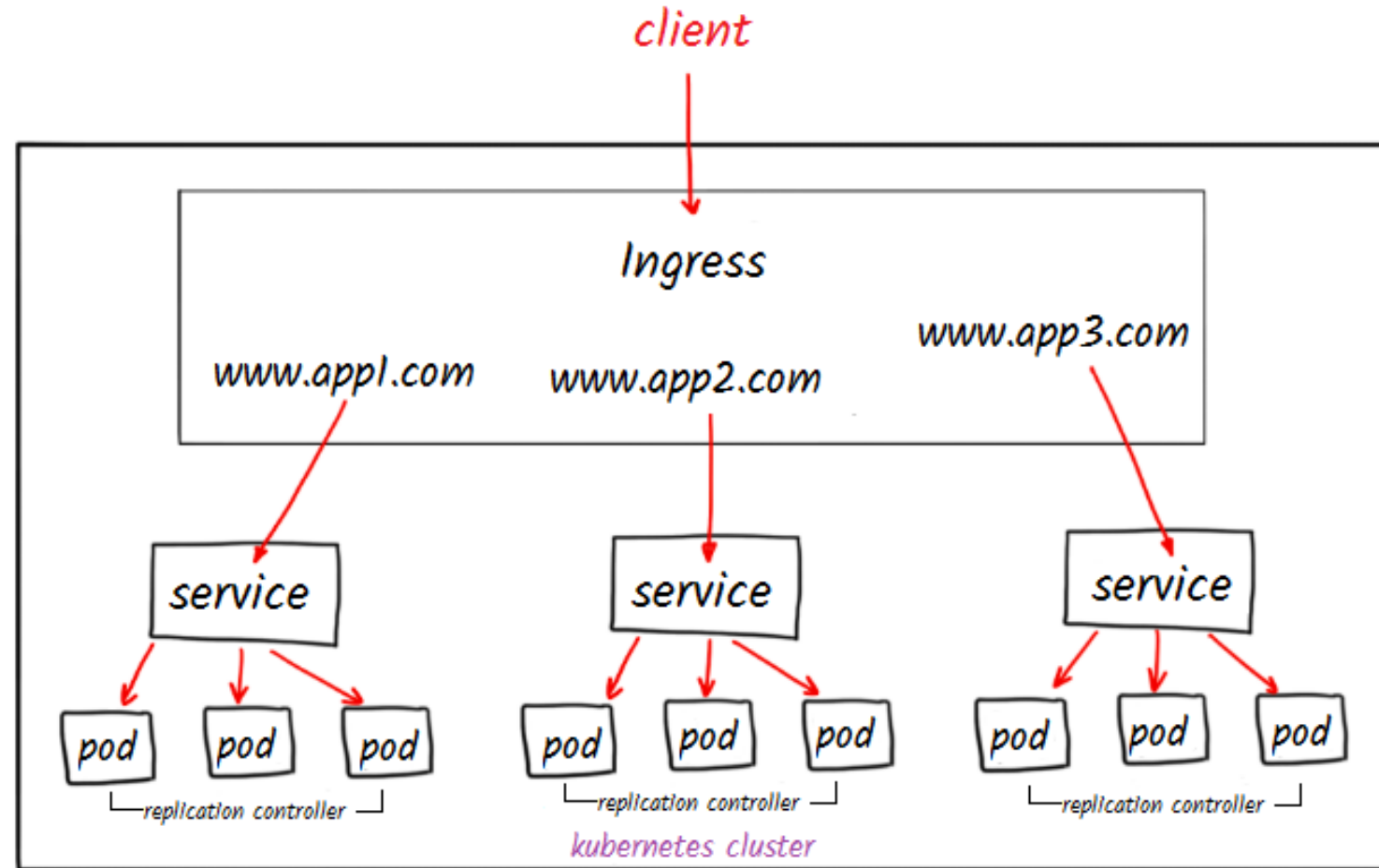


kubernetes

Kubernetes

resources

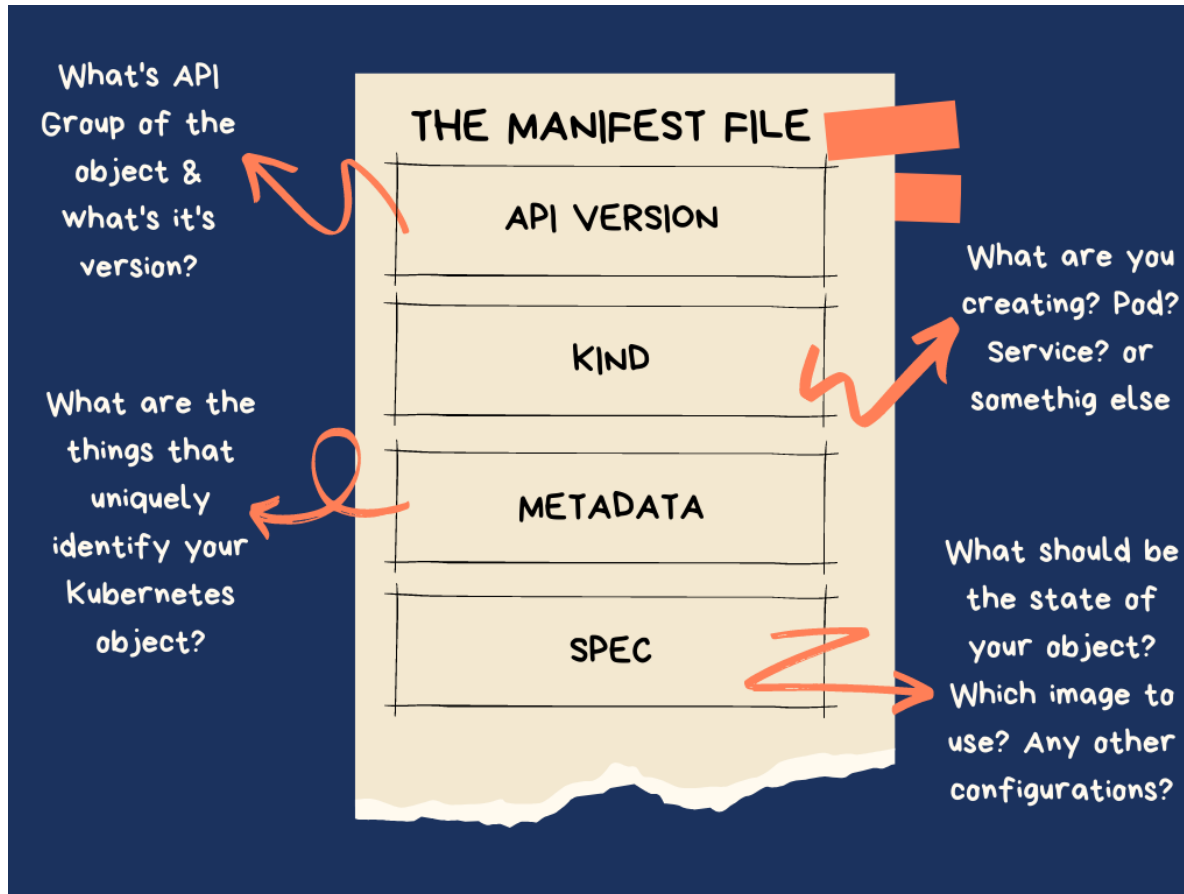
- **Cluster:** A set of Nodes that run containerized applications managed by Kubernetes.
- **Node:** A worker machine in Kubernetes, part of a cluster.
- **Ingress:** manages external access to the services in a cluster, typically HTTP.
- **Service:** a method for exposing a network application that is running as one or more **Pods** in your cluster.
- **Pods:** are the smallest deployable units of computing that you can create and manage in Kubernetes. It runs one or more containers, with shared storage and network resources, and a specification for how to run the containers.



Kubernetes

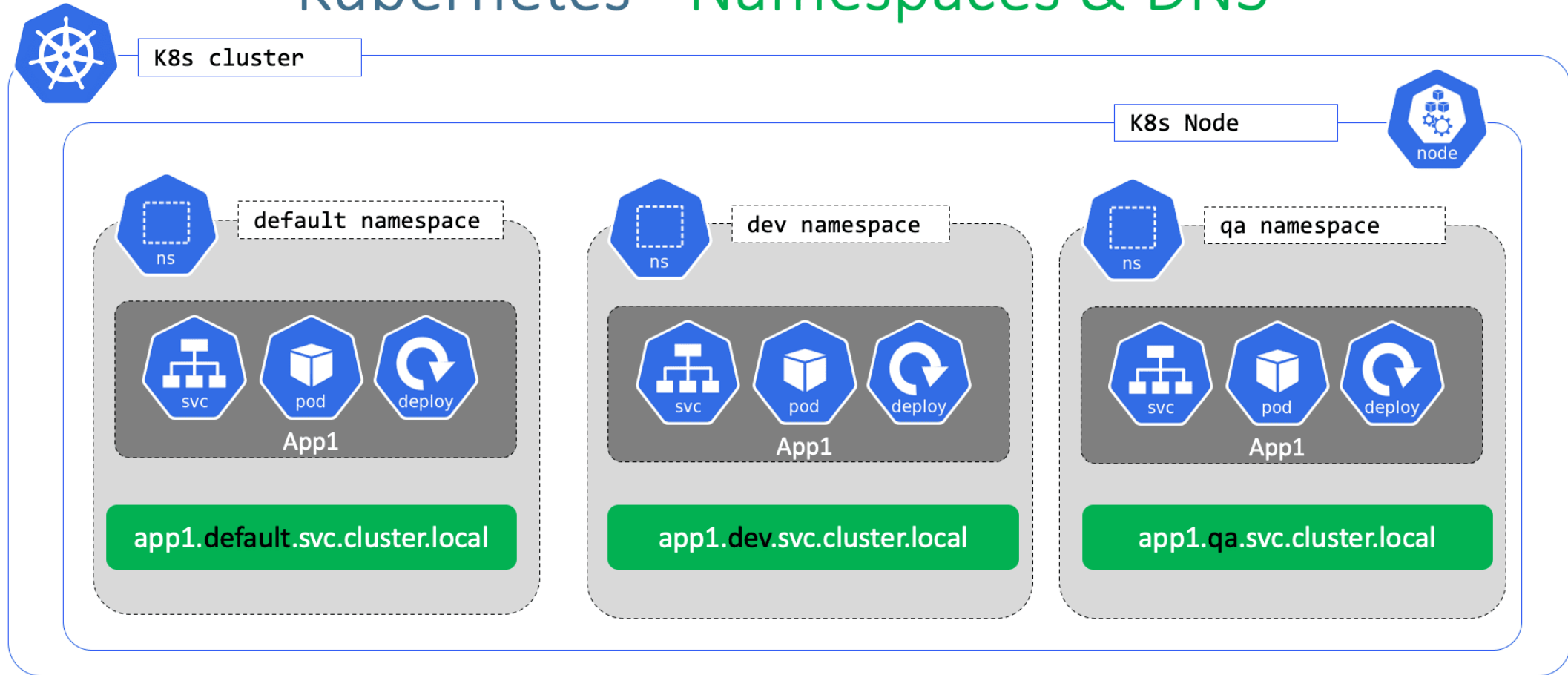
Manifest

- Beschrijving van je Kubernetes object in YAML of JSON



```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: todo-client-app-deploy
spec:
  replicas: 3
  selector:
    matchLabels:
      app: todo-client-app
  template:
    metadata:
      labels:
        app: todo-client-app
    spec:
      containers:
      - image: todo-client-kubernetes-app:latest
        name: container1
        imagePullPolicy: Always
```

Kubernetes - Namespaces & DNS

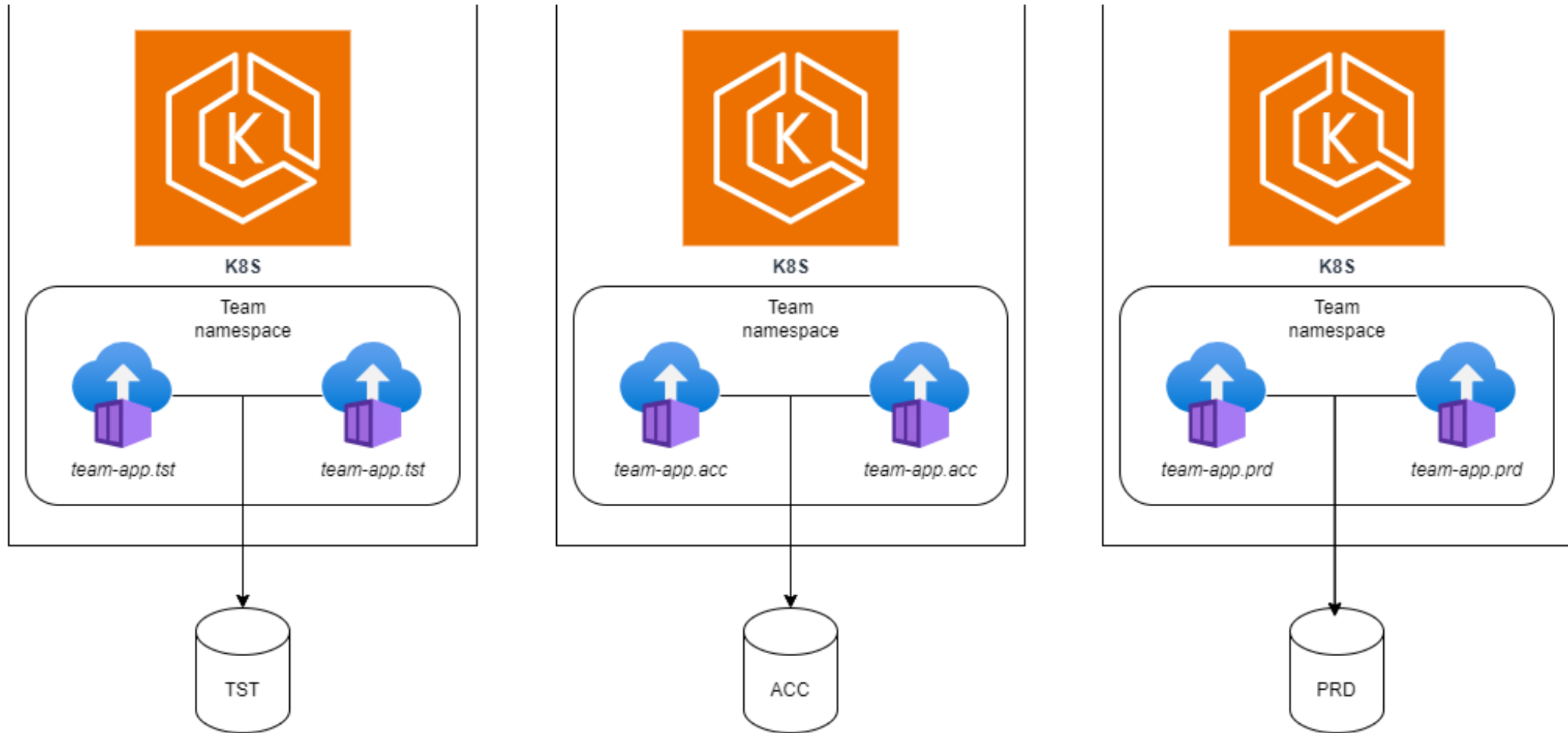


`<service-name>.<namespace-name>.svc.cluster.local`

Kubernetes

Bij Alliander

alllander



HELM

Package manager for Kubernetes

alllander

- Tool voor het managen en deployen van Kubernetes applicaties.
- Verpakken van Kubernetes resources in een Chart
- Een Kubernetes *package* -> *Chart*
 - Chart, alle resources om een instantie van een Kubernetes Applicatie te draaien
 - Configuratie voor applicatie
 - Een release is een instantie van een Chart met specifieke configuratie



**The
package manager
for Kubernetes**

HELM

Charts en templates

- Een helm chart bestaat uit:
 - Templates folder – bevat verschillende Kubernetes Manifests
 - Deployment.yaml, ingress.yaml, service.yaml, etc
 - Chart.yaml – Metadata over applicatie, naam, versie, type
 - Values.yaml – configuratie voor je applicatie, bijv. port, appname, etc
 - <https://github.com/helm/examples/tree/main/charts/hello-world>

HELM

Voordelen

- Helm makes life easy
- Installatie van software met een enkel command, Helm Install **package**
- Versionering van Charts
- Eenvoudig beheer van Kubernetes manifests
- Charts staan los van omgeving en zijn configureerbaar
- <https://artifacthub.io/>

Infrastructure as Code (IaC)

- Infrastructure as code is het beschrijven van je configuratie en inrichting van je cloud infrastructuur door middel van `code`
- YAML, Python, Typescript
- Bijvoorbeeld Helm Charts, maar ook:
 - Amazon Cloud Development Kit, Terraform en Ansible
- Voordelen
 - Sneller en automatisch neerzetten van juiste configuratie
 - Herbruikbaar en versie beheer op je infrastructuur
 - Minder risico op handmatige fouten (ClickOps)
 - Herkennen van *drift* tussen je huidige staat en beschreven staat van je applicatie

Gitops uses Git as a “single source of truth” for CI/CD processes, integrating code changes in a single, shared repository per project and using pull requests to manage infrastructure and deployment.



Gitops

Wat is gitops?

Gitops is een werkwijze voor cloud native applications & Kubernetes waarbij applicatie en infrastructuur code opgeslagen in Git wordt gebruikt om automatisch te deployen.

- Gitops is onderdeel van Devops.
- Gitops is een Continuous Deployment strategie
- Gitops is een uitbreiding op Infrastructure as Code (IaC).
- Gitops zorgt ervoor dat de Cloud infrastructuur van een systeem onmiddellijk reproduceerbaar is op basis van de toestand van een Git-repository
- Met een Gitops Operator synchroniseert je systeem met je infrastructuur code.

Gitops

Voordelen van gitops

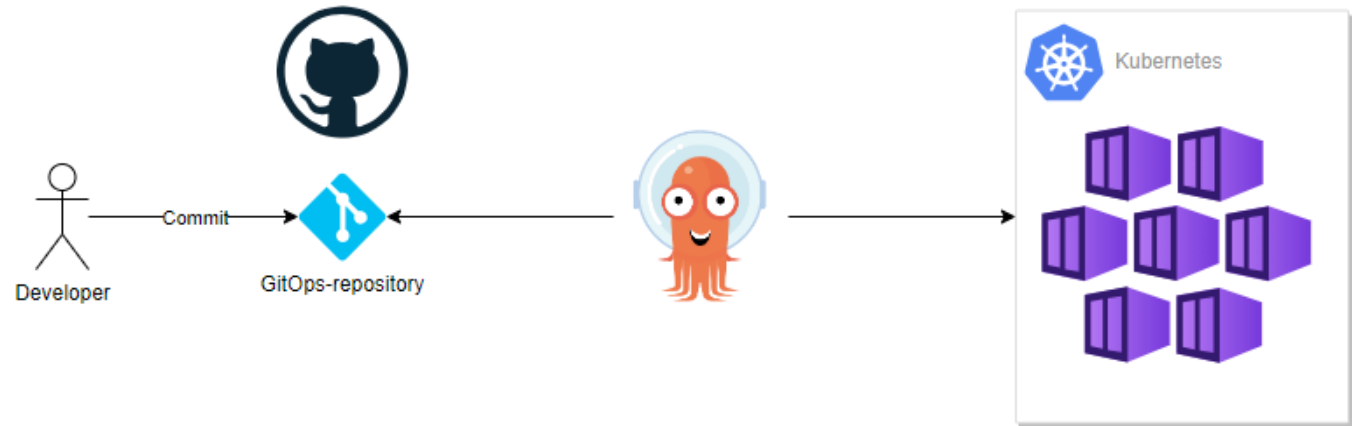
- Git is de single source of truth van de staat van je infrastructuur
- De historie van je gewenste staat van je infrastructuur staat opgeslagen in git
- Git is de enige plek waar je aanpassingen in je infrastructuur aanbrengt -> Security -> Developers hebben geen toegang tot het cluster nodig, enkel tot git.
- Je kan updates aan je infrastructuur automatisch uitrollen naar verschillende omgevingen

Gitops Operators

Hoe werkt Gitops?

De **Gitops Operator** kan de huidige staat van de cloud omgeving observeren (bijvoorbeeld per applicatie) en vergelijkt dit met de gewenste staat van het systeem zoals het in Git beschreven staat.

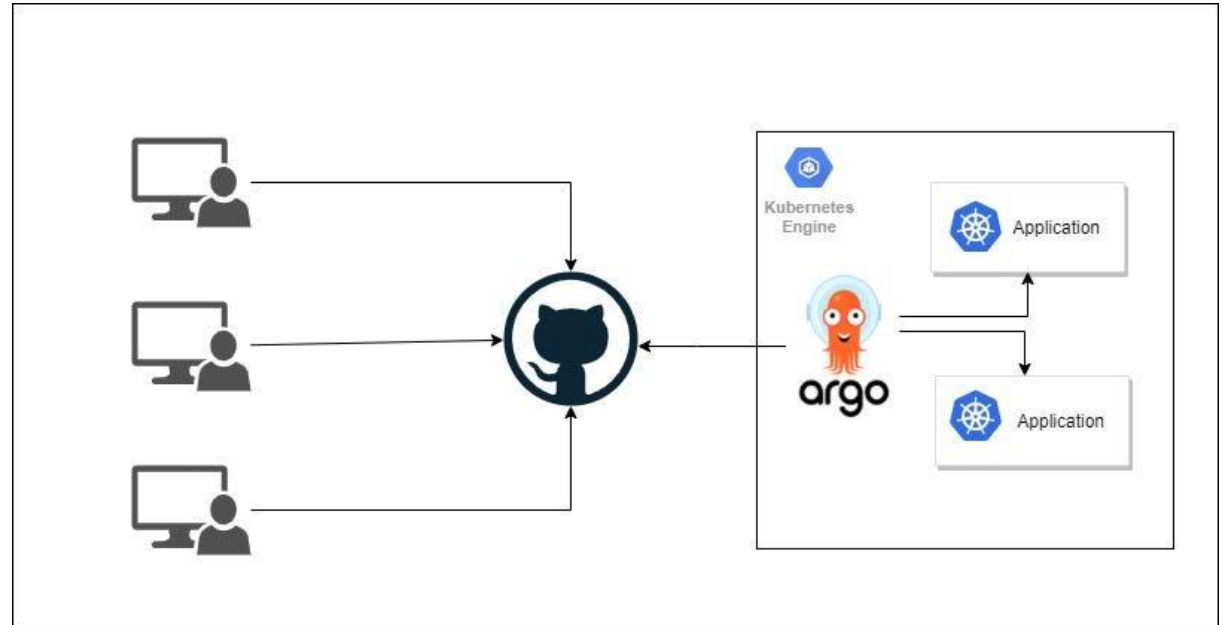
Bij Alliander gebruiken we ArgoCD als Gitops Operator



Gitops operators

Argo CD

- Gitops tool voor Continuous Delivery voor Kubernetes
- Continue monitoring op
 - state van actieve applicaties in het cluster
- Gewenste state gedefinieerd in Git
- OutOfSync / Drift
- Handmatig of automatisch (terug) synchroniseren naar doelstatus
- Health checks op je cluster
 - Healthy, Progressing, Degraded, Suspended



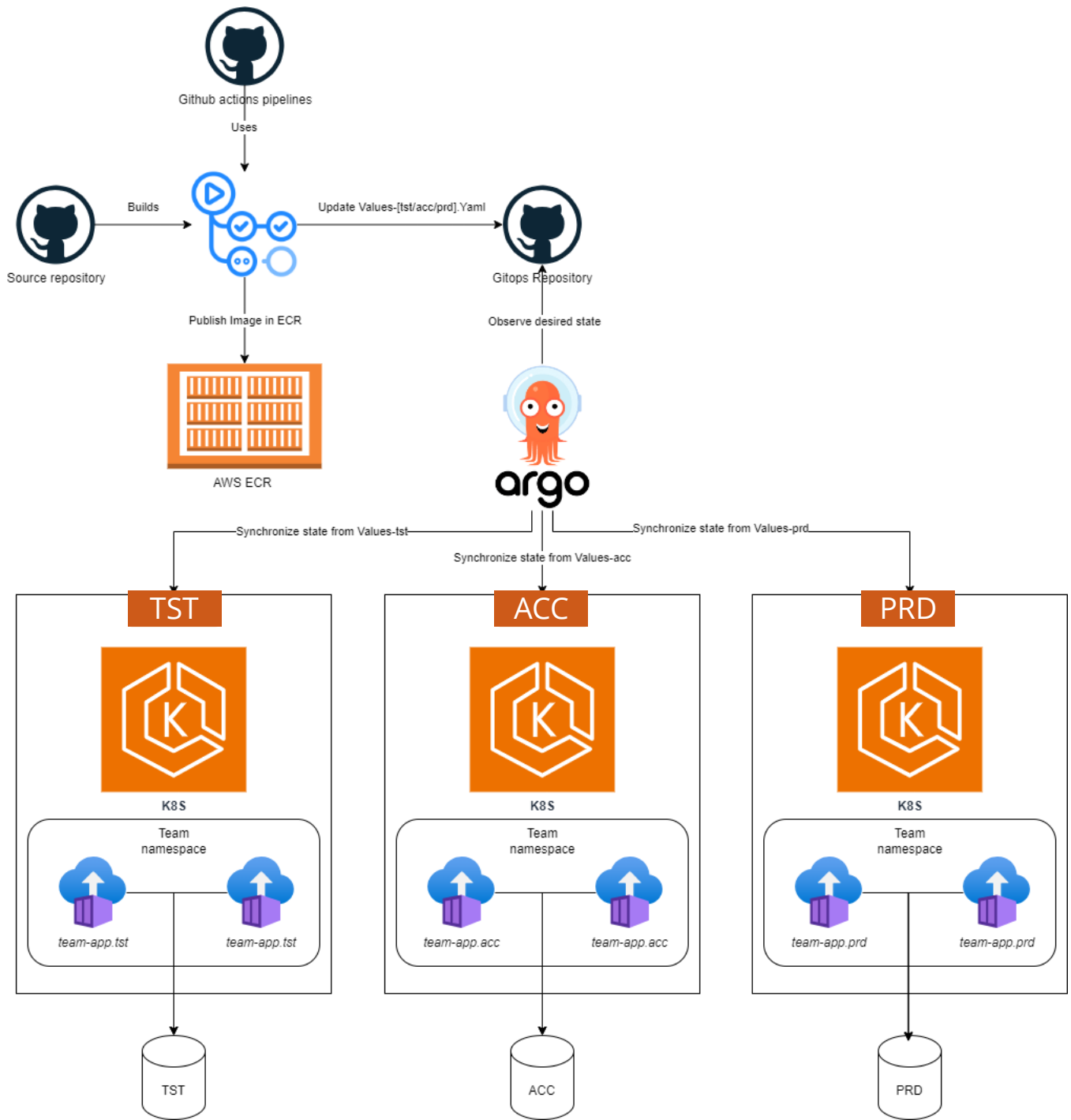
AWS ECR – Elastic Container Registry

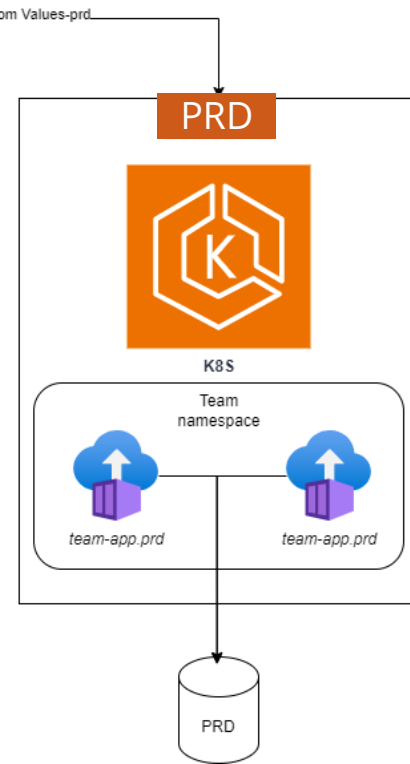
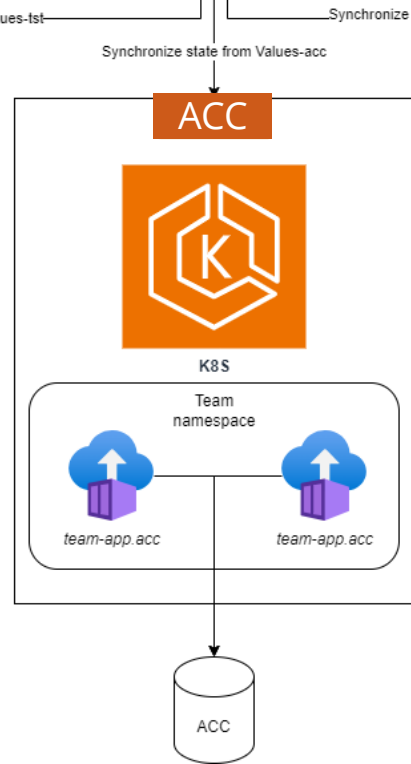
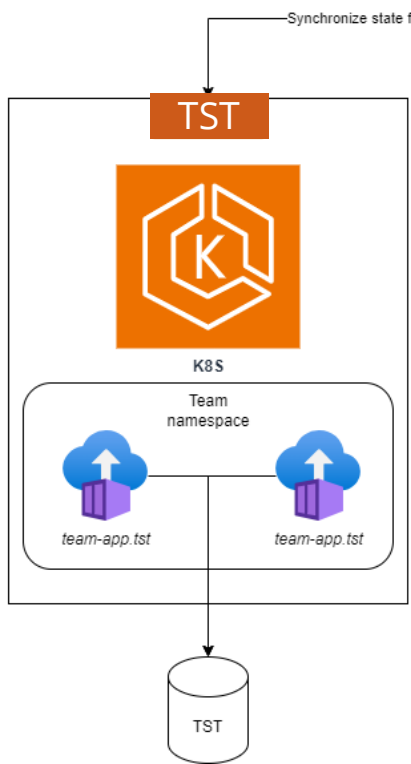
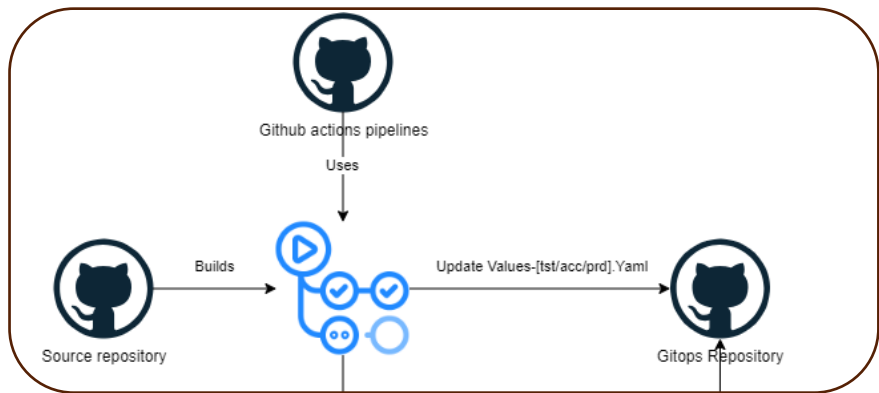
Register voor images

- Verzamelbak voor alle images die gebuild worden door GH Actions
- [repo naam]:[branch]-[github commit hash]-[datum build]
- Python-api:develop-4fd6-a8d6-f86a-5g9a-20240930
- Image retention policy, regels voor het bewaren van images obv tag
- Wekelijkse automatische build van images -> Package updates

```
1 - name: Set datetime variable
2   id: datetime
3   run: echo "datetime=$(date +%Y%m%d%H%M%S)" >> $GITHUB_OUTPUT
4 - name: Build, tag and push image to Amazon ECR
5   id: build
6   env:
7     ECR_REGISTRY: <your registry>
8     ECR_REPOSITORY: <your repo>
9   run: |-
10    if [ "$GITHUB_REF" == "refs/heads/main" ]; then
11      TAG="main-$GITHUB_SHA-${ steps.datetime.outputs.datetime }"
12    else
13      TAG="develop-$GITHUB_SHA-${ steps.datetime.outputs.datetime }"
14    fi
15    docker build -t $ECR_REGISTRY/$ECR_REPOSITORY:$TAG .
16    docker push $ECR_REGISTRY/$ECR_REPOSITORY:$TAG
17
```

```
1 {
2   "rules": [
3     {
4       "rulePriority": 1,
5       "description": "Keep the newest 5 main images",
6       "selection": {
7         "tagStatus": "tagged",
8         "tagPrefixList": ["main"],
9         "countType": "imageCountMoreThan",
10        "countNumber": 5
11      },
12      "action": {"type": "expire"}
13    },
14    {
15      "rulePriority": 2,
16      "description": "Keep the newest 10 develop images",
17      "selection": {
18        "tagStatus": "tagged",
19        "tagPrefixList": ["develop"],
20        "countType": "imageCountMoreThan",
21        "countNumber": 10
22      },
23      "action": {"type": "expire"}
24    }
25  ]
26 }
```

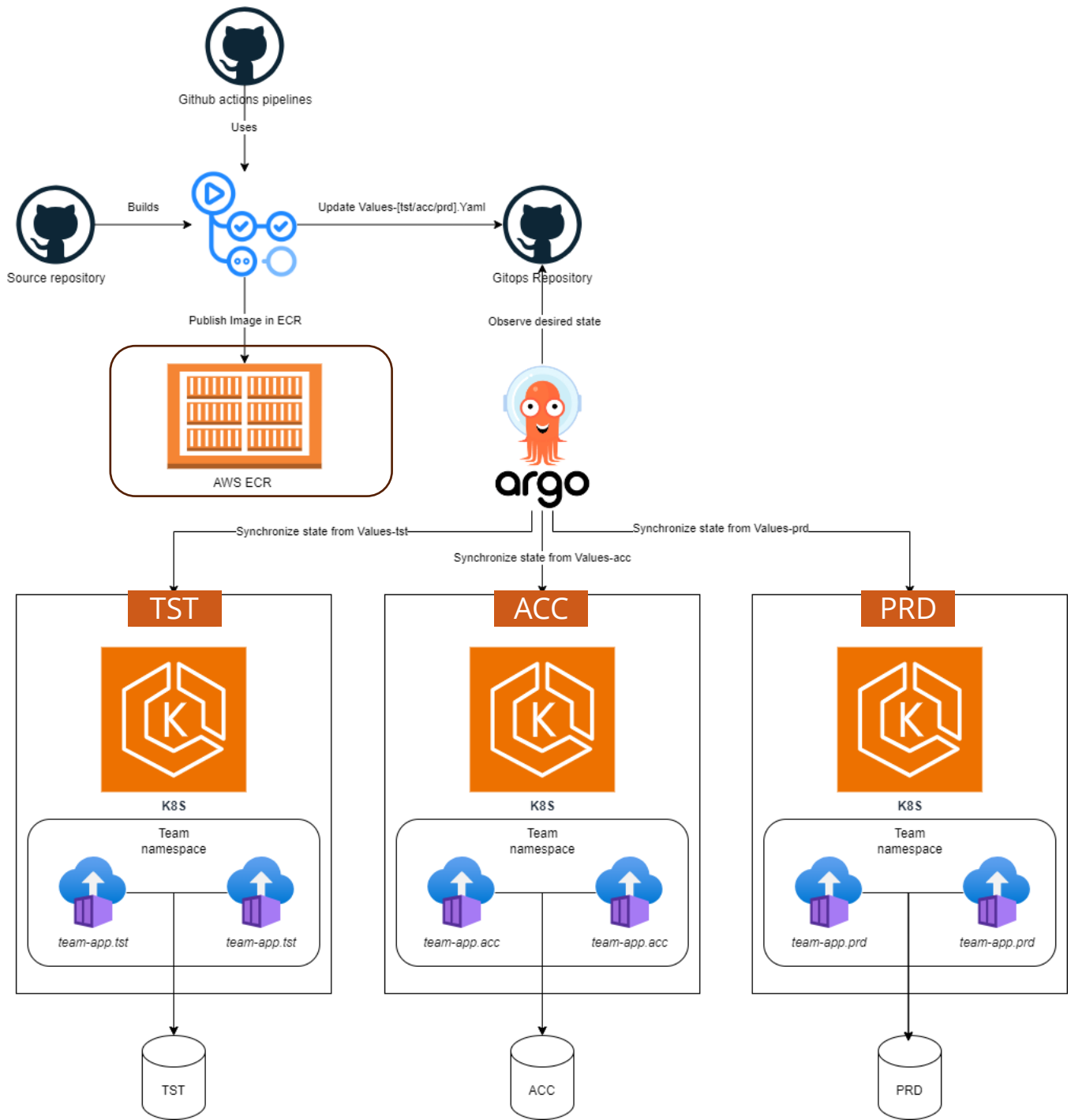


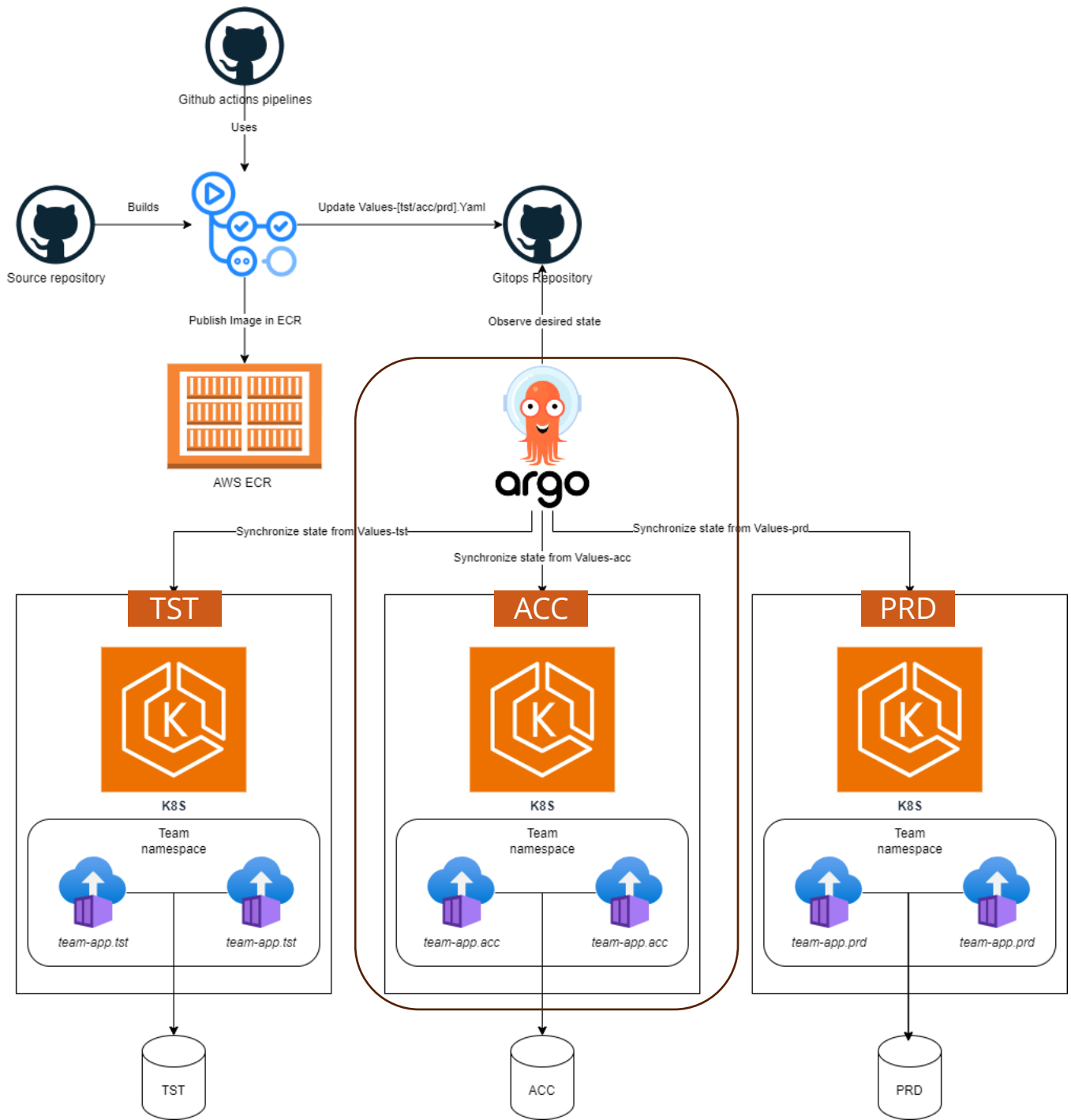


Summary

Jobs

- ✓ prepare
- ✓ build-image
- ✓ push-to-ecr
- ✓ update-gitops-repo
- ✓ refresh-argocd





Applicatie deployen op Kubernetes

Ci/CD met Github Actions en ArgoCD

- Aantal dingen vooraf:
 - Kale versie van een python api aangemaakt
 - Rechten ingesteld voor: ArgoCD, AWS en PAX
 - Container register in AWS aangemaakt
- Wat gaan we doen in de demo:
 - [CI] Github actions pipeline aanroepen en analyseren
 - [CD] GitOps repository configureren
 - [CD] ArgoCD applicatie aanmaken
 - [Config] Omgevings variabele configureren in aws
 - [Config] Applicatie controleren in Kubernetes