



Applications / **argocd-dev**

APPLICATION DETAILS TREE

APP DETAILS

APP DIFF

SYNC

SYNC STATUS

HISTORY AND ROLLBACK

DELETE

REFRESH

Log out

APP HEALTH  
**Missing**

CURRENT SYNC STATUS  
**OutOfSync** From 3.33.2 (3.33.2)

LAST SYNC RESULT  
**Sync OK** To 3.33.2

Succeeded a minute ago (Thu Feb 03 2022 11:47:24 GMT-0500)



# Do Rancher ao GitOps

Melhorando o fluxo dos Devs



**RANCHER**<sup>®</sup>



# Robert Silva

- DevOps e Instrutor na JLCP
- +10 anos trabalhando com TI atuando como consultor e arquiteto de soluções
- CKA | DCA | ZCE | ZCT
- Aspirante a marombeiro, a breja não deixa o shape vir.

Contatos:

- IG, LinkedIn, Twitter, YouTube:  
@robertsilvatech



# Bernardo Lankheet

- DevOps e Especialista em Monitoramento na JLCP
- +10 anos atuando com TI e a 3 anos como consultor
- CKA | AZ-104 | AZ-500 | ZCP | ZCS
- Ciclista, trilheiro e corredor de rua nas horas vagas

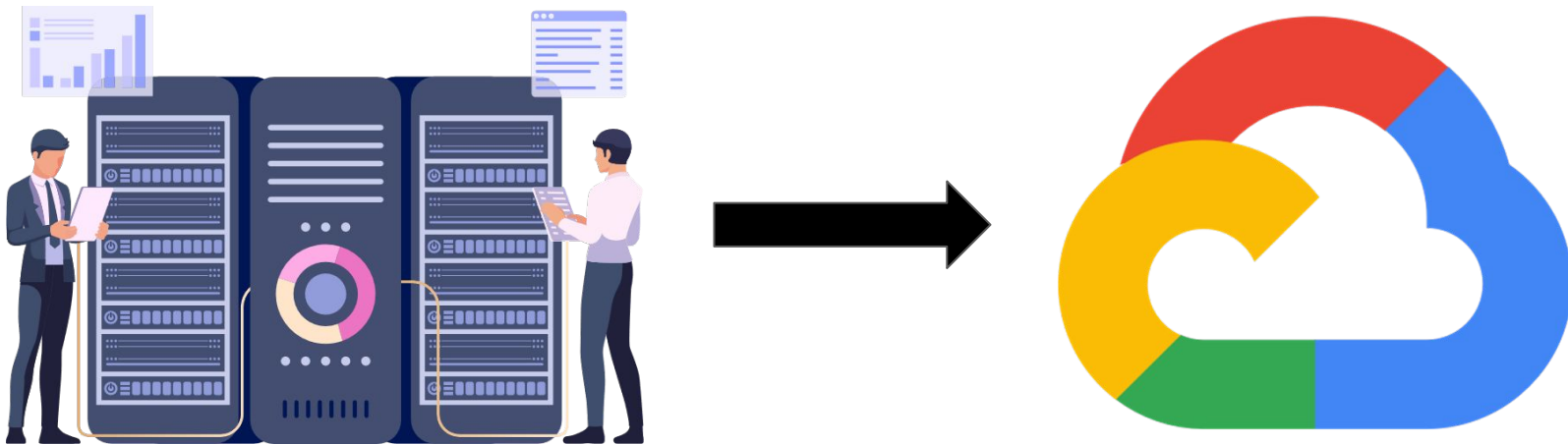
## Contatos:

- IG, LinkedIn, Twitter, YouTube: @bernardolankheet



# O projeto

Principal objetivo, desocupar o datacenter, migrando todas as aplicações para a GCP.



# O ambiente

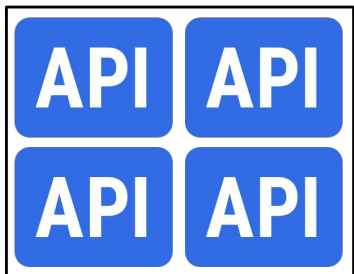
Swarm > Kubernetes



# As aplicações



Escopo  
suporte a várias  
áreas de negócio



Escopo  
automações de  
Tech



# Desafios

- Definir a estratégia de migração de Swarm para Kubernetes;
- Manter um padrão de deploy das aplicações;
- Aplicar melhorias no padrão de deploy das aplicações.

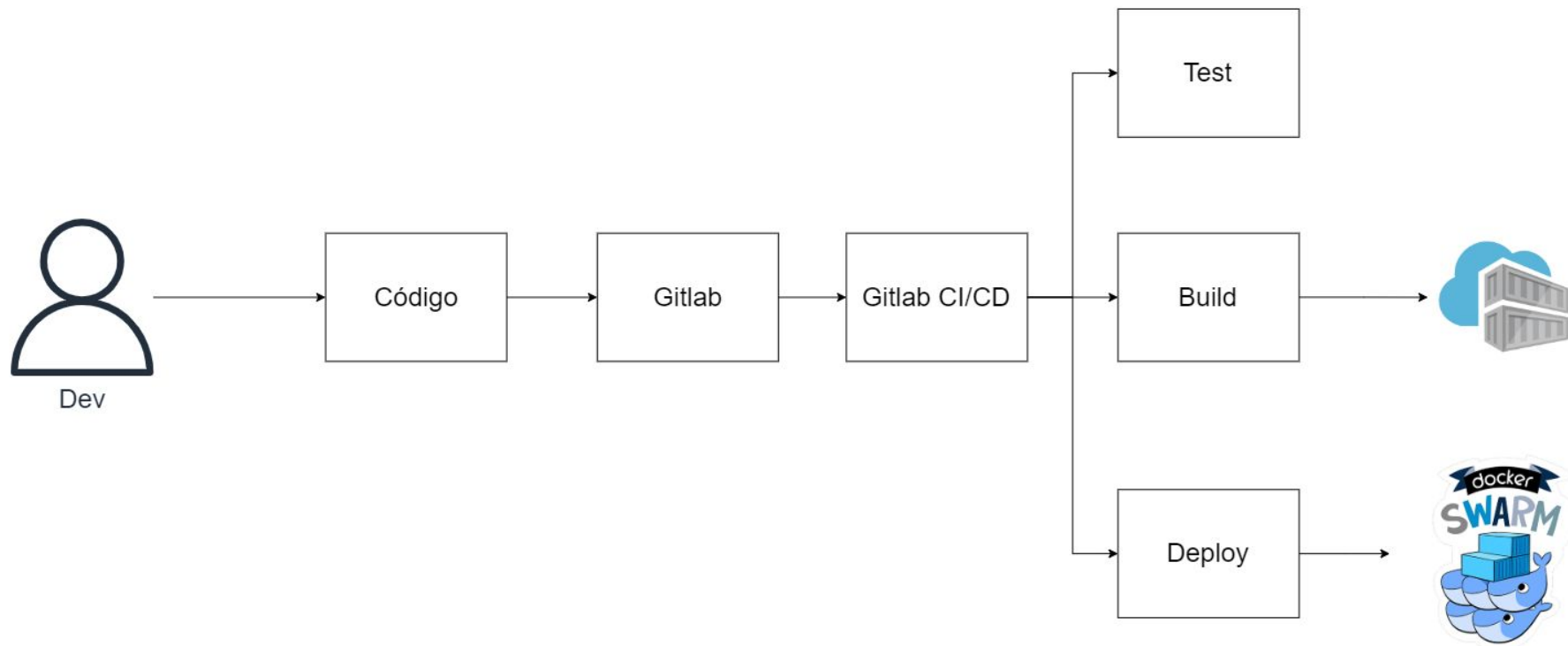
## SURPRESAS

Durante o mapeamento com o time de dev que atuava no Kubernetes:

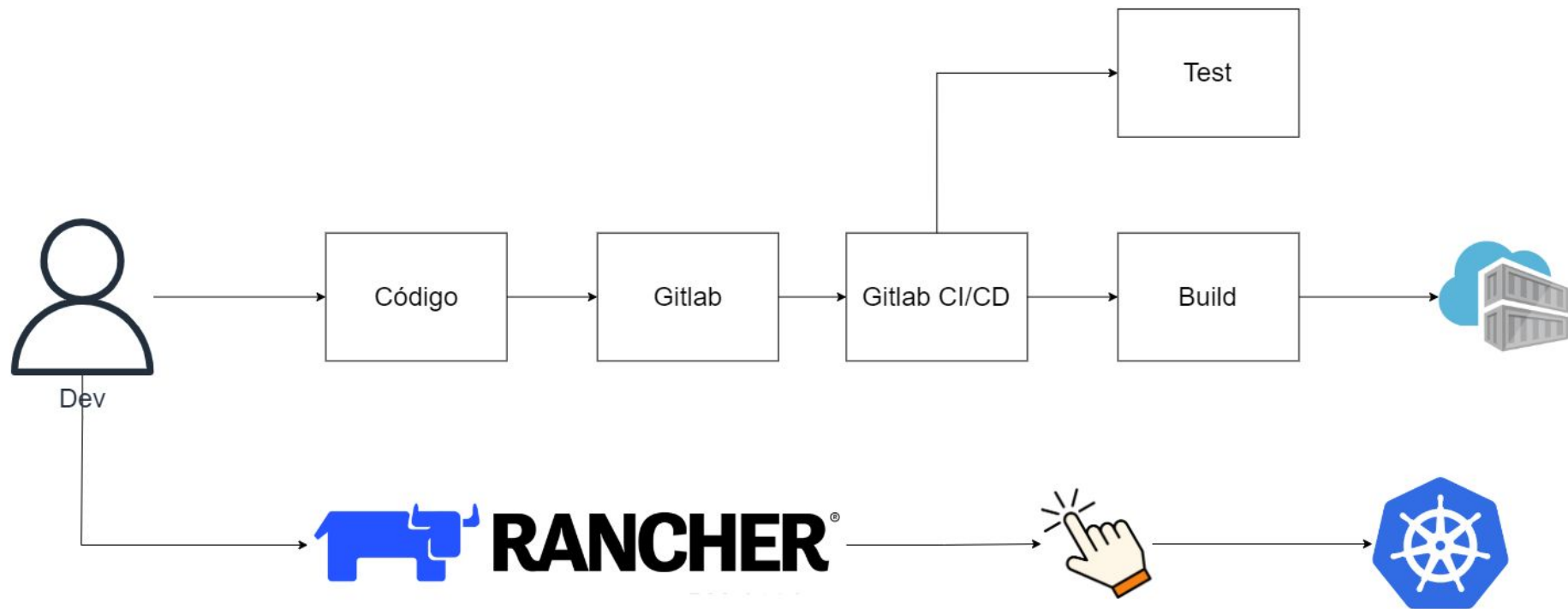
- Objetos do Kubernetes criado direto no Rancher via GUI;
- Processo de deploy manual;
- Após o build, acessar o Rancher e trocar a imagem do deployment.



# O processo de CI/CD para Swarm



# O processo manual dos Devs



# O novo padrão de Deploy



Criação de charts para definir o padrão de deploy das aplicações.

Problemas que encontramos no caminho



- Como integrar o Gitlab com o Kubernetes?
  - Time responsável não homologou a integração.

ArgoCD, um nova ferramenta para compor a estrutura.

- O processo de Deploy muda completamente.



# O ArgoCD

ArgoCD é uma ferramenta declarativa de entrega contínua GitOps para Kubernetes:

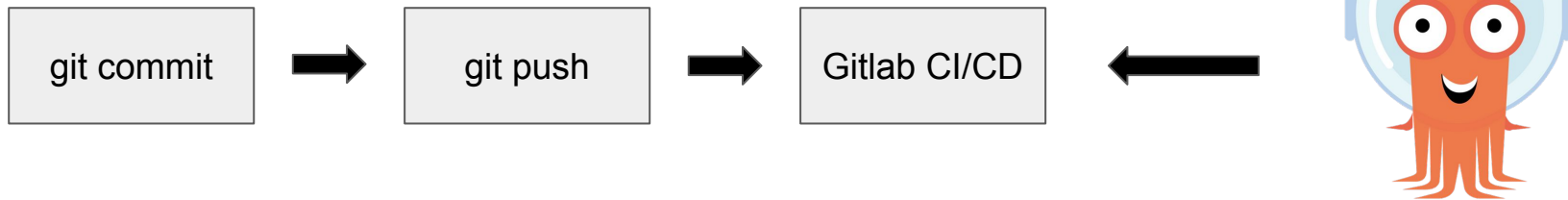
- O ArgoCD segue o padrão GitOps de usar repositórios Git como a fonte da verdade para definir o estado desejado do aplicativo;
- O ArgoCD faz apenas o Continuous Delivery.

O ArgoCD monitora a pasta de um repositório definido em sua aplicação para identificar mudanças de status para que possa aplicar ou não no cluster K8S.



# Revisando o Fluxo

git push model



- Qualquer push na branch configurada no `.gitlab-ci.yml` vai trigger o pipeline.
- O ArgoCD **identifica uma alteração e valida se houve mudanças na aplicação ou não.**



Então basta configurar o argo apontando pro repositório atual onde temos o código fonte da aplicação + o helm chart?

## Best Practices



### Separating Config Vs. Source Code Repositories

Using a separate Git repository to hold your kubernetes manifests, keeping the config separate from your application source code, is highly recommended for the following reasons:

[https://argo-cd.readthedocs.io/en/stable/user-guide/best\\_practices/](https://argo-cd.readthedocs.io/en/stable/user-guide/best_practices/)

# Cada time com seu repositório

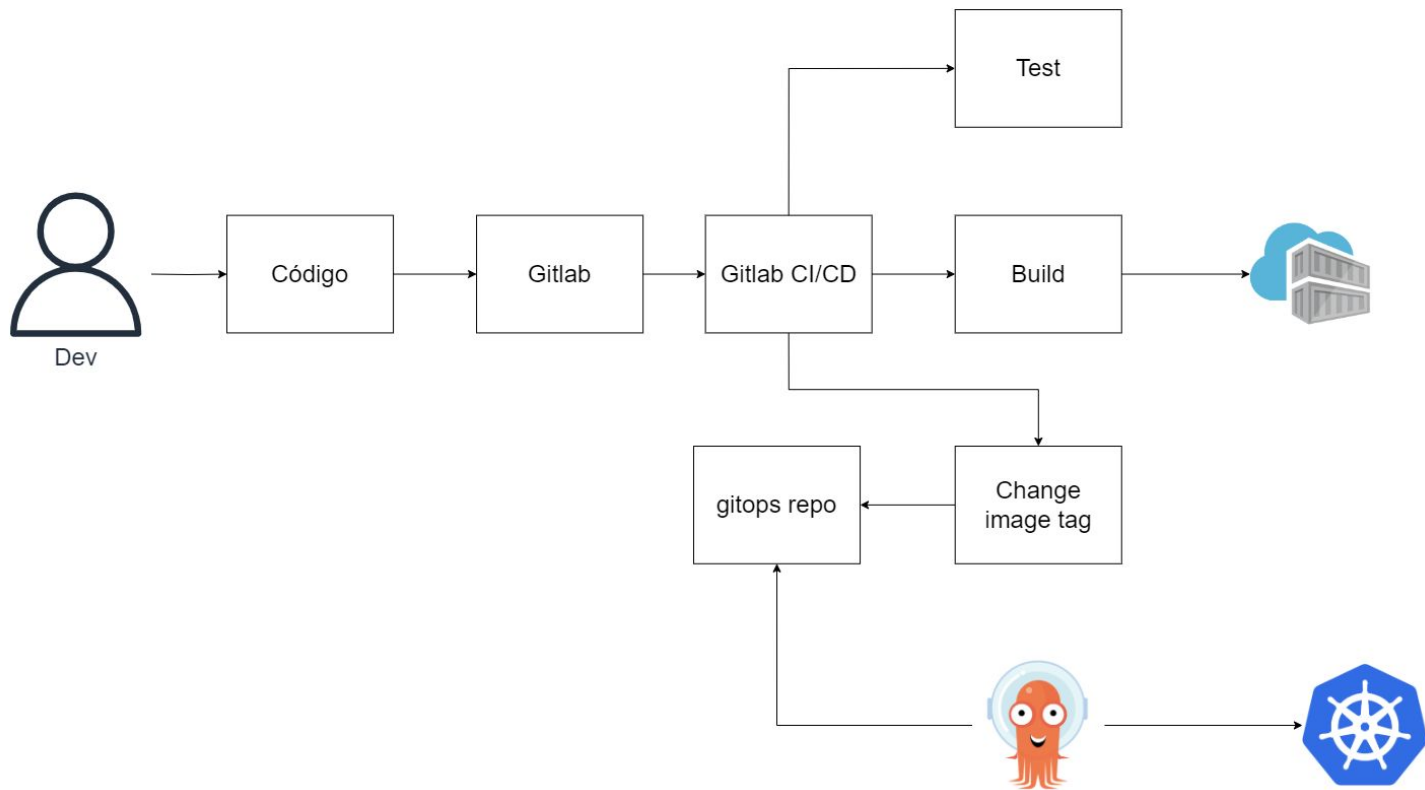


Código fonte da  
aplicação



Objetos  
Kubernetes

# O novo fluxo





Um pouco de configuração

```
image: python:3.8
```

```
stages:
```

- test
- build
- change-image-tag

```
change-image-tag-helm: &change-image-tag-helm
  image:
    name: robertsilvatech/ubuntu-with-git:20.04
    entrypoint: [ '/bin/bash', '-c', 'ln -snf /bin/bash /bin/sh && /bin/bash -c $0' ]
  needs:
    - build
  stage: change-image-tag
  variables:
    CONFIG_REPO_NAME: gitops-aplicacoes
    PROJECT_FOLDER: app-xyz-api
  before_script:
    - git config --global user.email "ci@argo" && git config --global user.name "Gitlab CI"
  script:
    - echo "cloning config repo $CONFIG_REPO_NAME"
    - git clone "https://$GIT_TOKEN_USER:$GIT_TOKEN@gitlab.empresa.com/$CONFIG_REPO_NAME.git"
    - ls && cd $CONFIG_REPO_NAME && pwd
    - git remote -v
    - git checkout $CI_COMMIT_REF_SLUG
    - 'ls $PROJECT_FOLDER/helm/values-$CI_COMMIT_REF_SLUG*.yaml | xargs sed -i -r "s/tag: (.*)/tag: $CI_COMMIT_REF_SLUG-$CI_COMMIT_SHORT_SHA/g"'
    - git status
    - git add .
    - git commit -m "Update image tag from CI - $CI_COMMIT_REF_SLUG-$CI_COMMIT_SHORT_SHA"
    - git push origin $CI_COMMIT_REF_SLUG
  only:
    - staging
  tags:
    - gcp-shared-runner
```

APP HEALTH ⓘ

**Healthy**

CURRENT SYNC STATUS ⓘ

**OutOfSync**

Author:

Gitlab CI <ci@argo> -

Comment:

Update image tag from CI - staging-2046c97f

MORE

LAST SYNC RESULT ⓘ

**Sync OK**

Succeeded 8 days ago (Thu Apr 27 2023 10:04:22 GMT-0300)

Author:

Gitlab CI <ci@argo> -

Comment:

Update image tag from CI - staging-2046c97f

MORE

APP HEALTH ⓘ

**Healthy**

CURRENT SYNC STATUS ⓘ

**Synced**

Author:

Gitlab CI <ci@argo> -

Comment:

Update image tag from CI on zabbix-server-mysql - staging-6d7b9

MORE

LAST SYNC RESULT ⓘ

**Sync OK**

Succeeded 2 months ago (Tue Mar 07 2023 17:31:29 GMT-0300)

Author:

bernardo.lankheet\_jkcp@protonmail.com -bernardo.lankheet\_jkcp@protonmail.com

Comment:

Alterado logica para query rule id 2

MORE

FILTERS

NAME

NAME

KINDS

KINDS

SYNC STATUS

☐ Synced 3

80%

[SUMMARY](#)[PARAMETERS](#)[MANIFEST](#)[DIFF](#)[EVENTS](#)

Compact diff



Inline diff

### apps/Deployment/staging/servicenow-publisher-events-staging



121	spec:	121	spec:
122	progressDeadlineSeconds: 600	122	progressDeadlineSeconds: 600
123	replicas: 0	123	replicas: 1
124	revisionHistoryLimit: 10	124	revisionHistoryLimit: 10
125	selector:	125	selector:

🕒 Deployed At:  
a month ago (Tue Mar 28  
2023 17:11:42 GMT-0300)

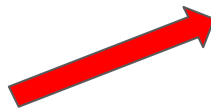
Revision: 049197e



Rollback

⌚ Time to deploy:  
00:02 min

Active for:  
38 days 00:54 hours



🕒 Deployed At:  
2 months ago (Mon Mar  
13 2023 17:57:41 GMT-  
0300)

Revision: 441094b



⌚ Time to deploy:

# Obrigado!

