



THE 'OUTER LOOP' IS DEAD: HOW AI AND CLOUD DEVELOPMENT WILL REVOLUTIONISE DEVELOPER EXPERIENCE

LOU BICHARD DX DAY 14.03.24



Introducing GovApp!

The revolutionary new platform for delivering local government services to your citizens.

[GET STARTED](#)



[Watch Intro](#)





```
FROM node:20
WORKDIR /usr/src/app
COPY package*.json ./
RUN npm install
COPY . .
EXPOSE 3000
CMD [ "node", "main.js" ]
```

[Home](#) [Services](#) [Portfolio](#) [Pricing](#) [Team](#) [Contact](#)

Introducing GovApp!

The revolutionary new platform for delivering local government services to your citizens.

[GET STARTED](#)



[Watch Intro](#)



```
● ● ●  
  
version: '3.0'  
services:  
  front-end:  
    image: front-end  
    build:  
      context: ./front-end  
      dockerfile: Dockerfile  
    ports:  
      - "3000:3000"  
  
  back-end:  
    image: back-end  
    build:  
      context: ./back-end  
      dockerfile: Dockerfile  
    ports:  
      - "3001:3001"
```







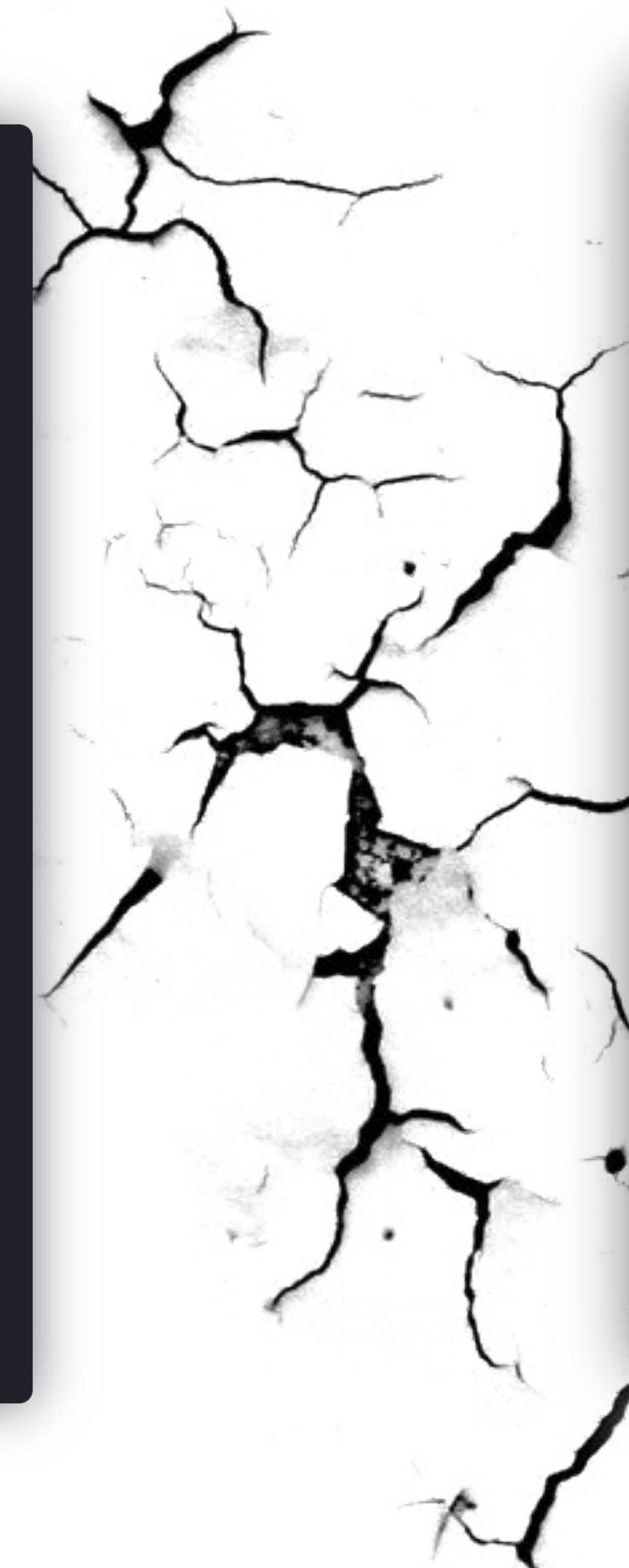
```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: microservicea
spec:
  selector:
    matchLabels:
      app: front-end
  template:
    spec:
      containers:
        - name: front-end
          image: 16335.dkr.ecr.us-east-1.amazonaws.com/front-end
      ports:
        - name: http
          containerPort: 3000
```



New message in: #incident-govapp-production-broken

Have you fixed it, yet?

```
version: '3.0'  
services:  
  front-end:  
    image: front-end  
    build:  
      context: ./front-end  
      dockerfile: Dockerfile  
    ports:  
      - "3000:3000"  
  
  back-end:  
    image: back-end  
    build:  
      context: ./back-end  
      dockerfile: Dockerfile  
    ports:  
      - "3001:3001"
```



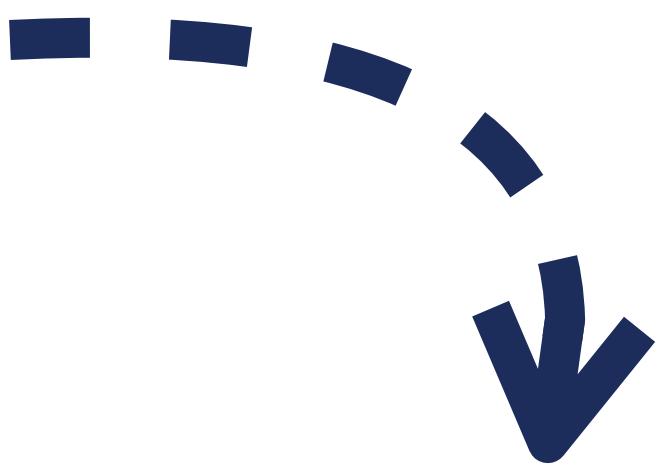
```
apiVersion: apps/v1  
kind: Deployment  
metadata:  
  name: microservicea  
spec:  
  selector:  
    matchLabels:  
      app: front-end  
  template:  
    spec:  
      containers:  
        - name: front-end  
          image: 16335.dkr.ecr.us  
      ports:  
        - name: http  
          containerPort: 3000
```





**WELCOME, TO THE
CHASM OF DOOM**

HELLO, I'M LOU



PRODUCT MANAGER

PREV

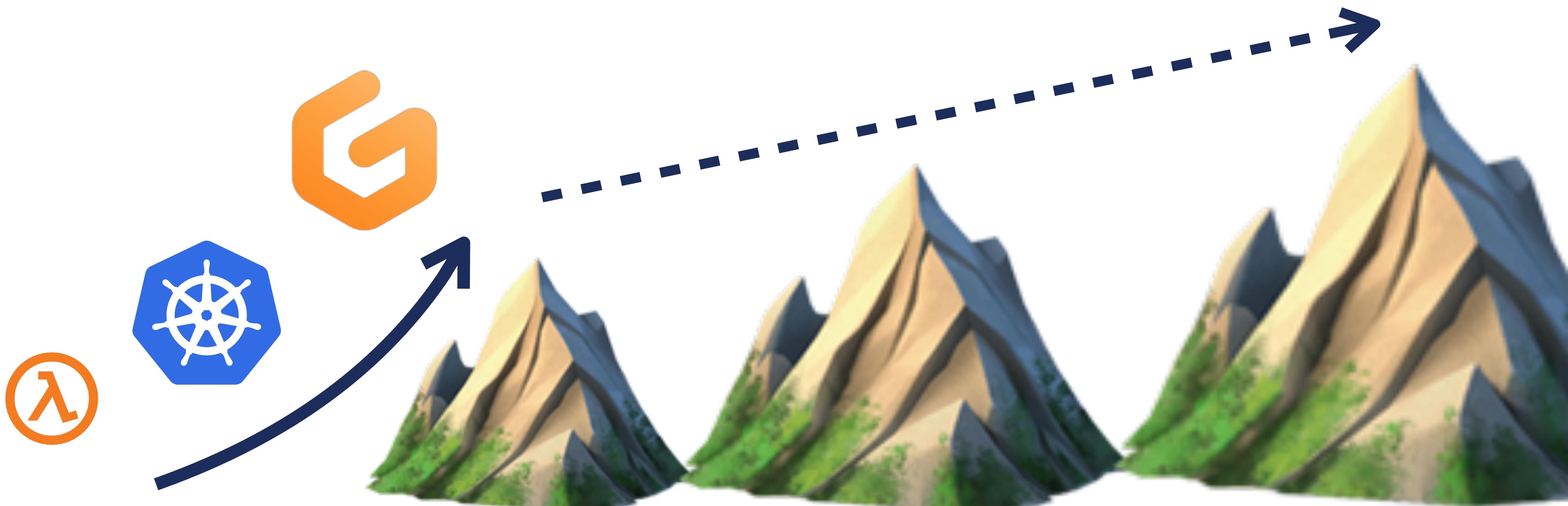
PLATFORM ENGINEER

PREV

SOFTWARE ENGINEER

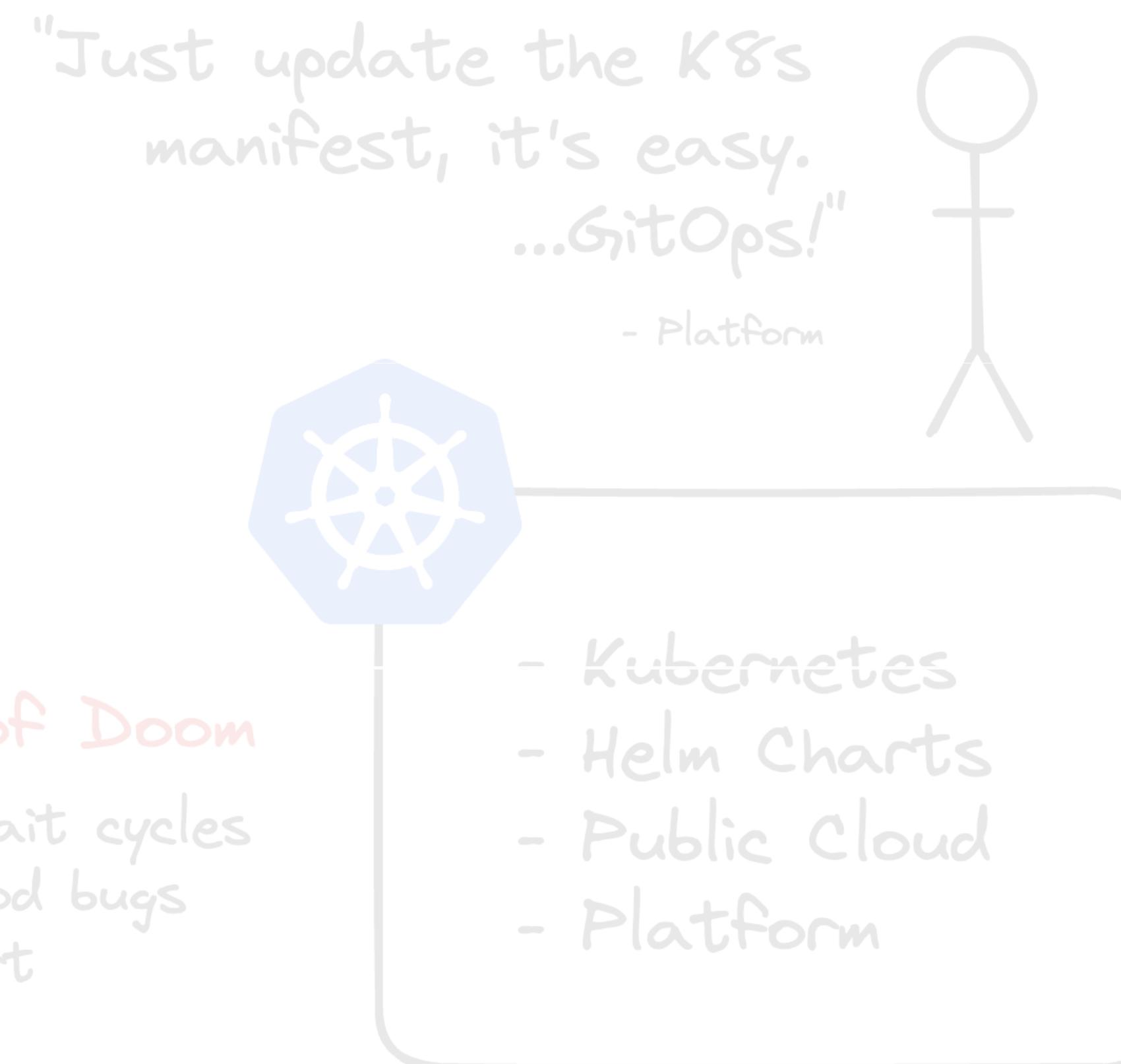
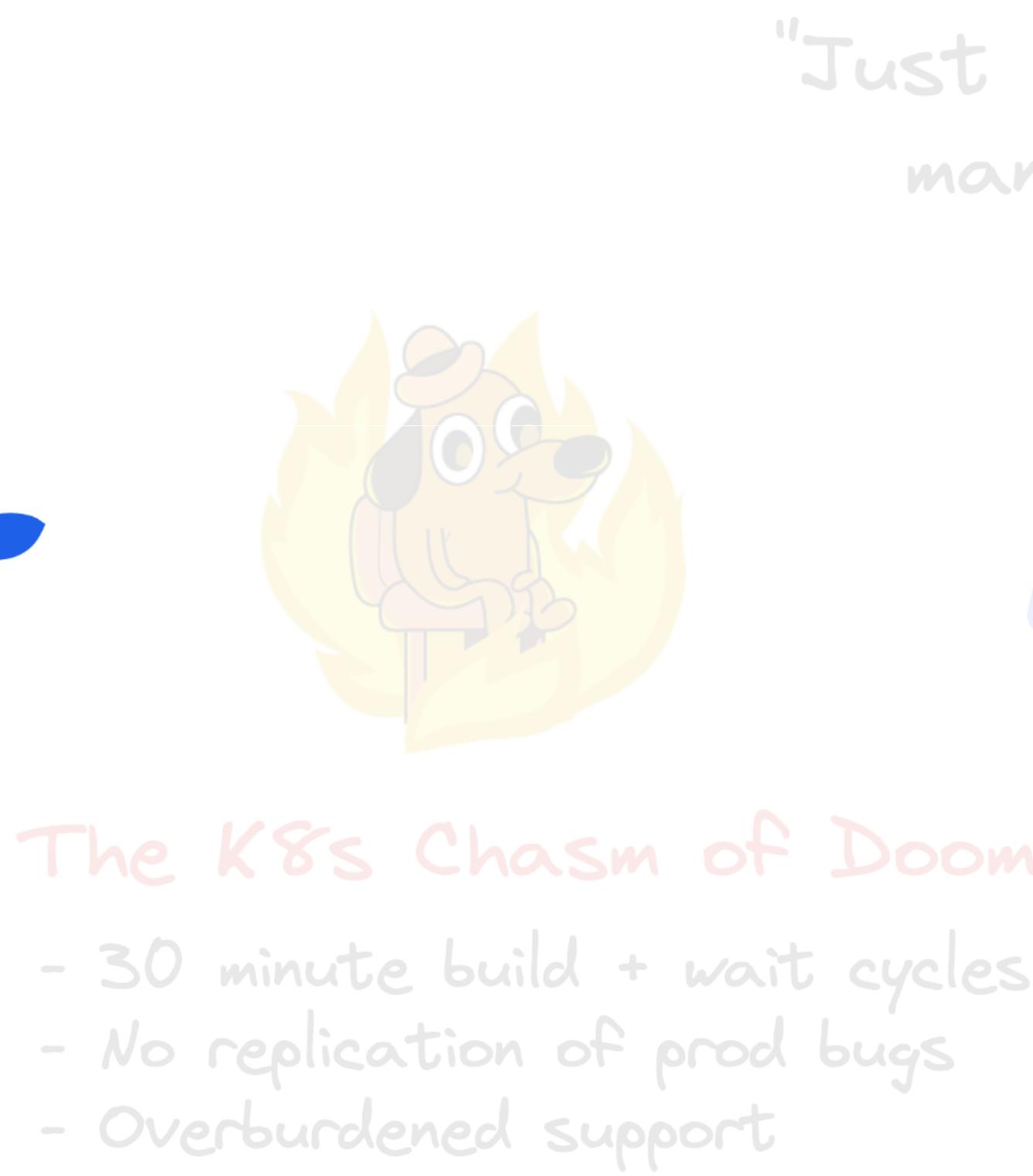
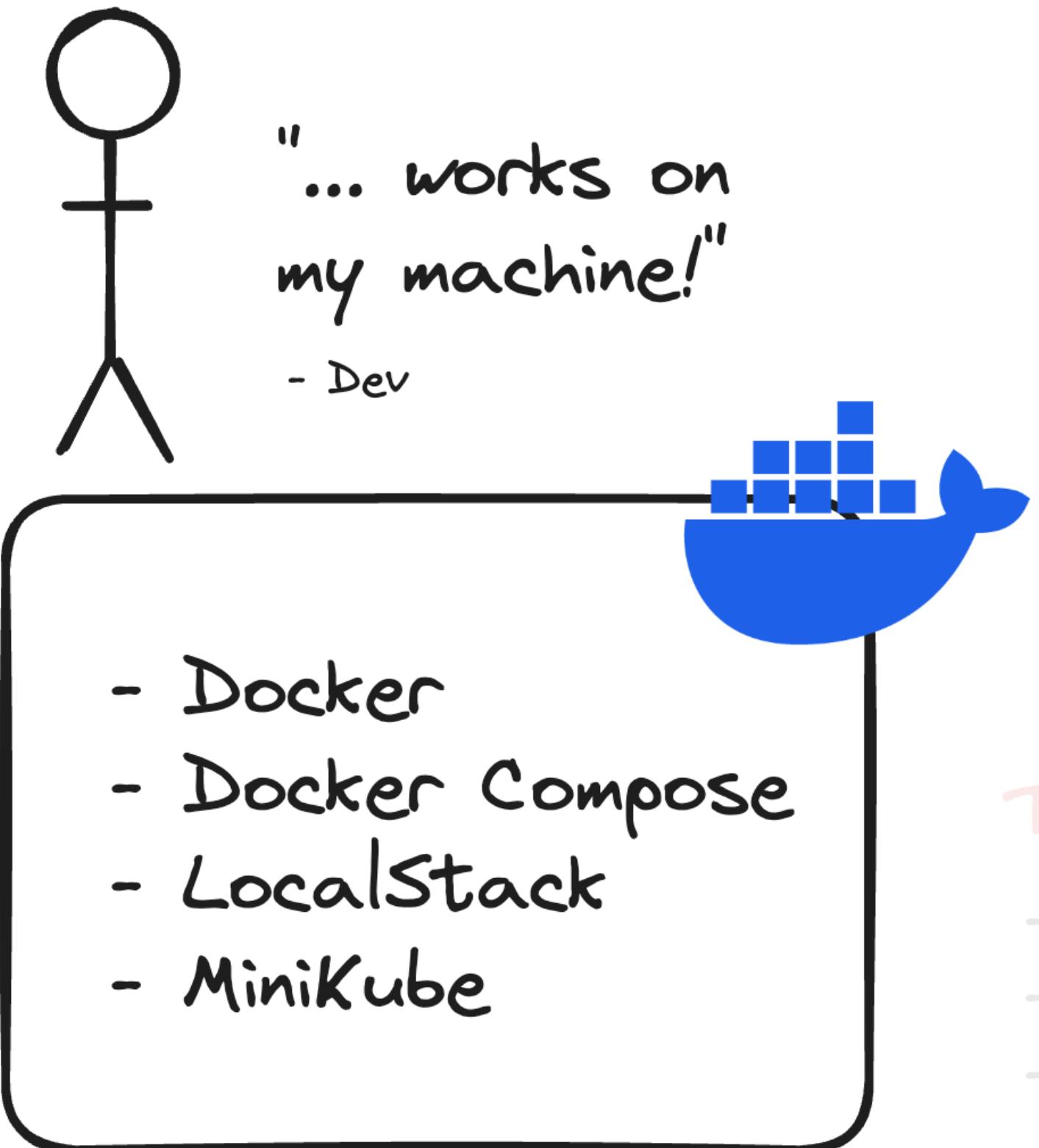


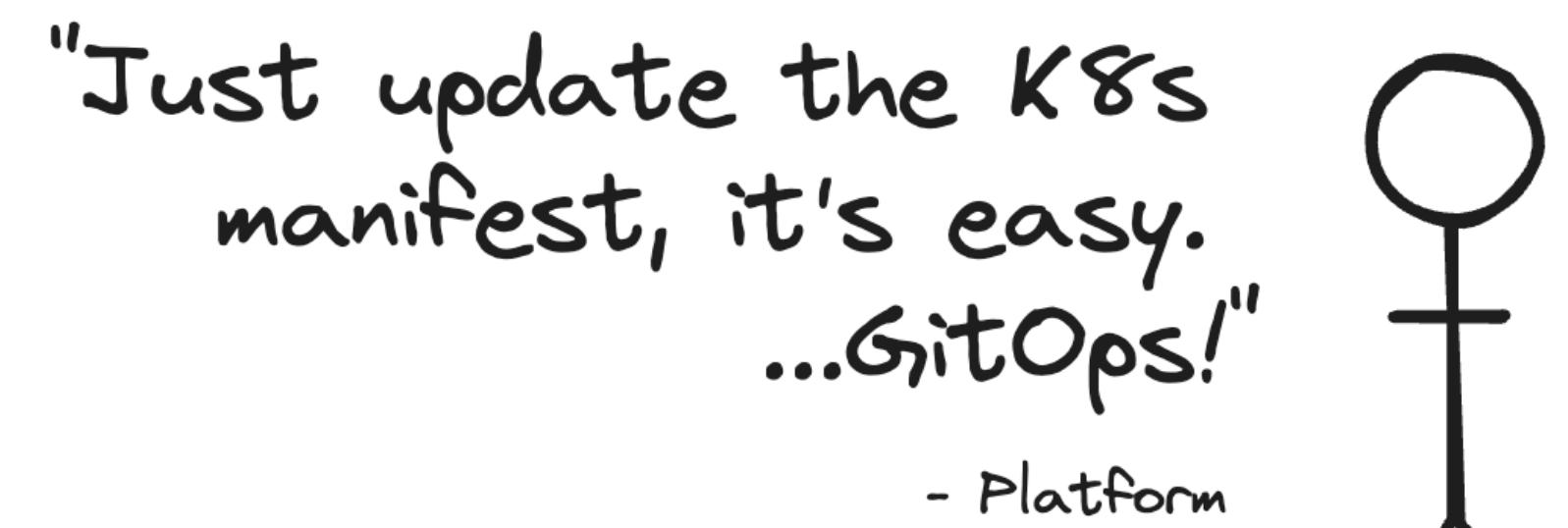
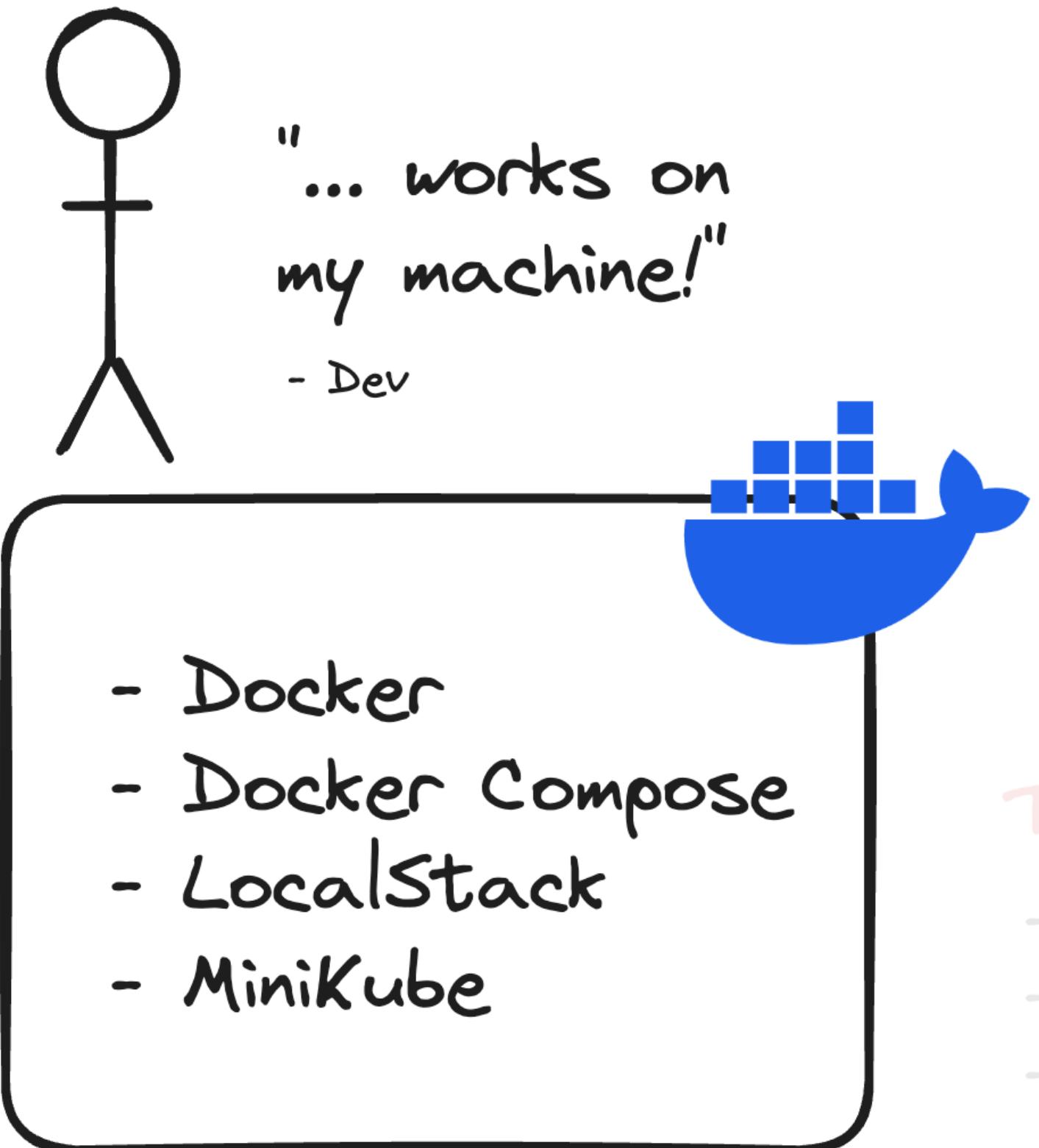
MENTAL MODEL VISION OF THE FUTURE

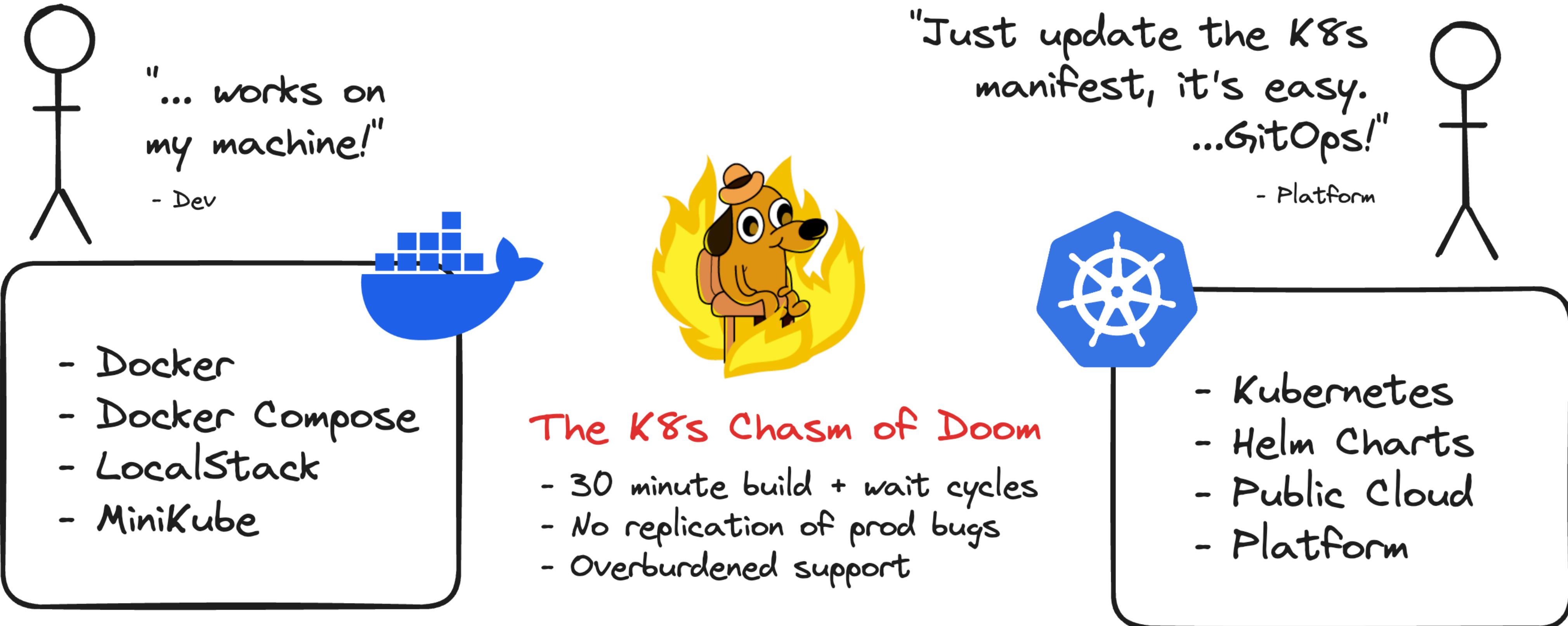


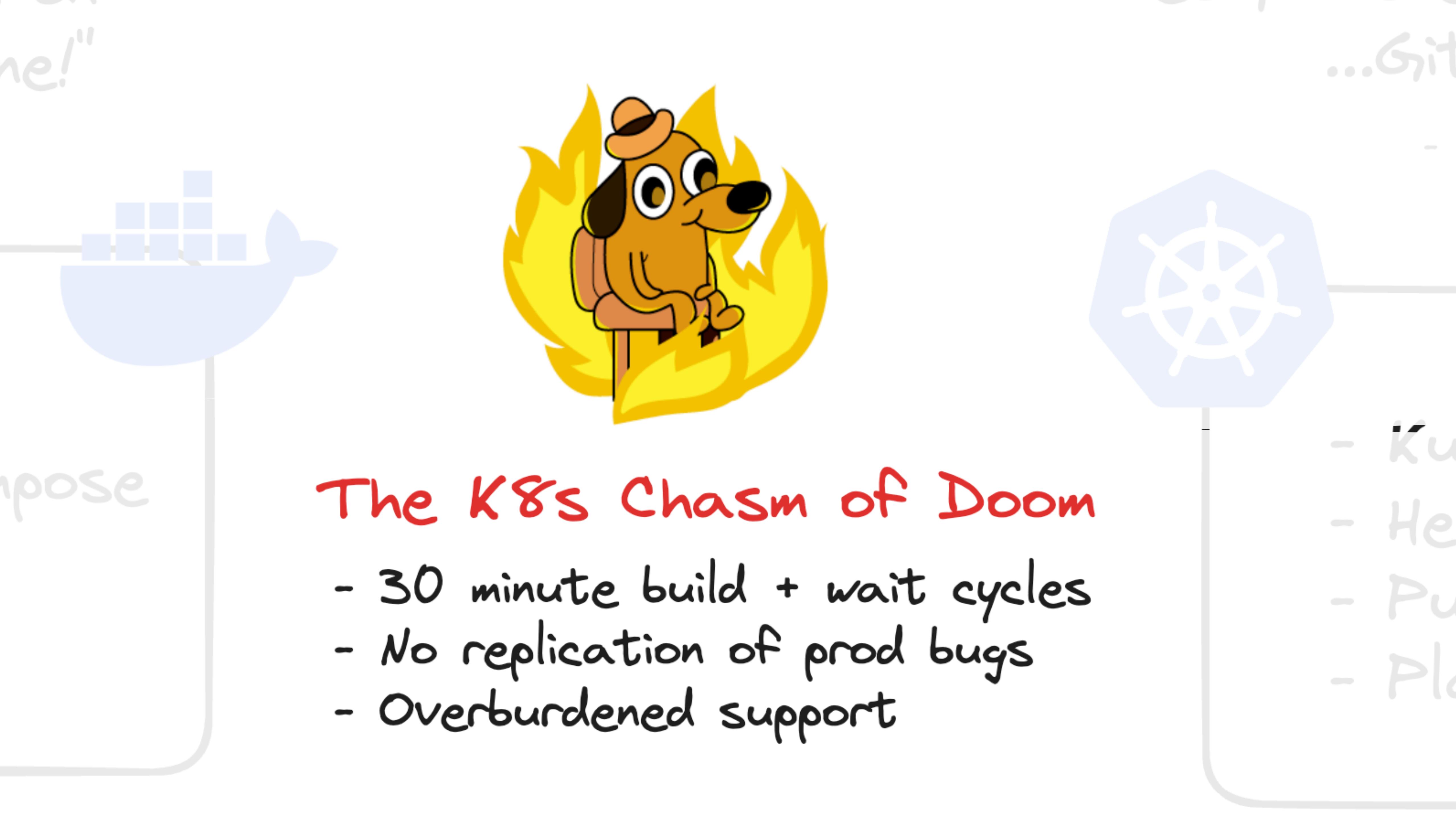


**WELCOME, TO THE
CHASM OF DOOM**









The K8s Chasm of Doom

- 30 minute build + wait cycles
- No replication of prod bugs
- Overburdened support

**ABSTRACTING APPS
FROM INFRA
IS REALLY HARD**

WHAT WE RUN



ABSTRACTING APPS
FROM INFRA
IS REALLY HARD

WHAT WE RUN

ABSTRACTING APPS

FROM INFRA

IS REALLY HARD

FROM WHERE IT RUNS



COMPILER, IDE

ABSTRACTING APPS

FROM INFRA

IS REALLY HARD

CLOUD, DISTRIBUTED

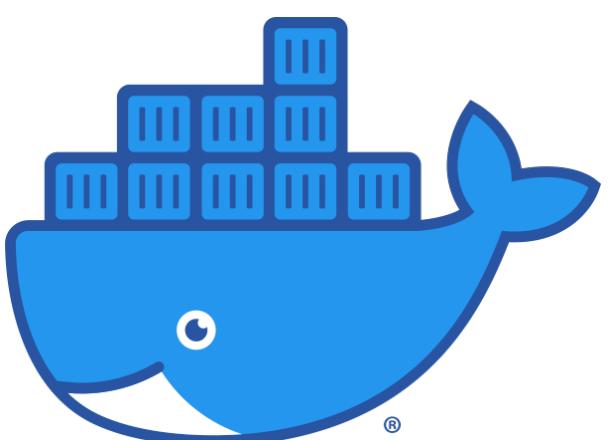


INNER LOOP

**ABSTRACTING APPS
FROM INFRA
IS REALLY HARD**

OUTER LOOP

INNER LOOP

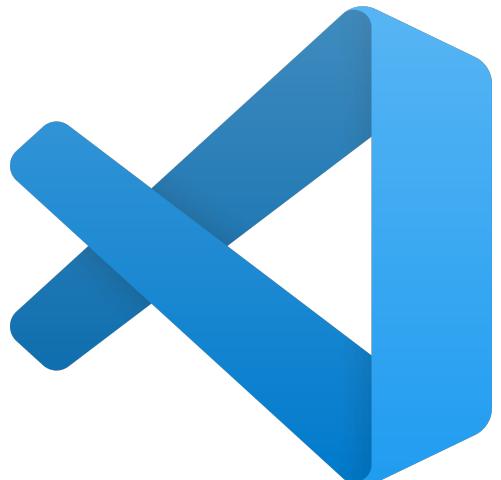


OUTER LOOP



DRIFT

INNER LOOP



OUTER LOOP



**“BUT... I’M NOT
USING KUBERNETES”**



"Just mock the API
it's easy
...Serverless!"

- Serverless Dev



- NPM
- LocalStack
- SLS
- CDK



The Serverless Chasm of Doom

- 30 minute build + wait cycles
- No replication of prod bugs
- Need a PhD in AWS

"You don't worry about scaling! Just learn 15 new services"

- Serverless Dev



- AWS Lambda
- DynamoDB
- API Gateway
- Step Functions





"Just mock the API
it's easy
...Serverless!"

- Serverless Dev



- NPM
- LocalStack
- SLS
- CDK



The Serverless Chasm of Doom

- 30 minute build + wait cycles
- No replication of prod bugs
- Need a PhD in AWS

"You don't worry about scaling! Just learn 15 new services"

- Serverless Dev



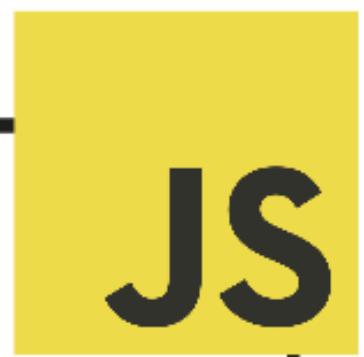
- AWS Lambda
- DynamoDB
- API Gateway
- Step Functions





"Just mock the API
it's easy
...Serverless!"

- Serverless Dev



- NPM
- LocalStack
- SLS
- CDK



The Serverless
Chasm of Doom

- 30 minute build + wait cycles
- No replication of prod bugs
- Need a PhD in AWS

"You don't worry about
scaling! Just learn 15
new services"

- Serverless Dev



- AWS Lambda
- DynamoDB
- API Gateway
- Step Functions





"Just mock the API
it's easy
...Serverless!"

- Serverless Dev



- NPM
- LocalStack
- SLS
- CDK



The Serverless Chasm of Doom

- 30 minute build + wait cycles
- No replication of prod bugs
- Need a PhD in AWS



"You don't worry about scaling! Just learn 15 new services"

- Serverless Dev



- AWS Lambda
- DynamoDB
- API Gateway
- Step Functions

easy
serverless!"

ess Dev

ack

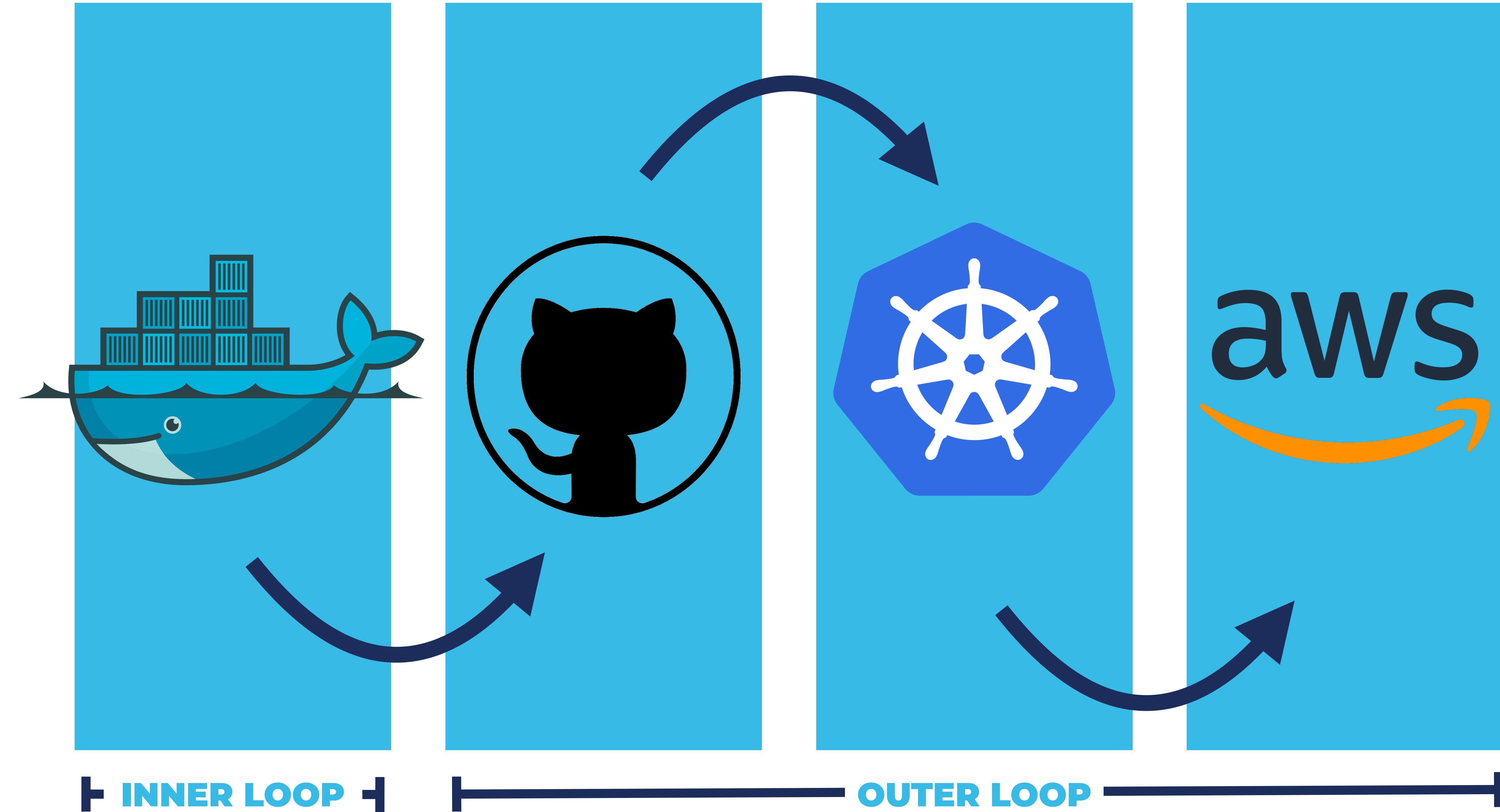


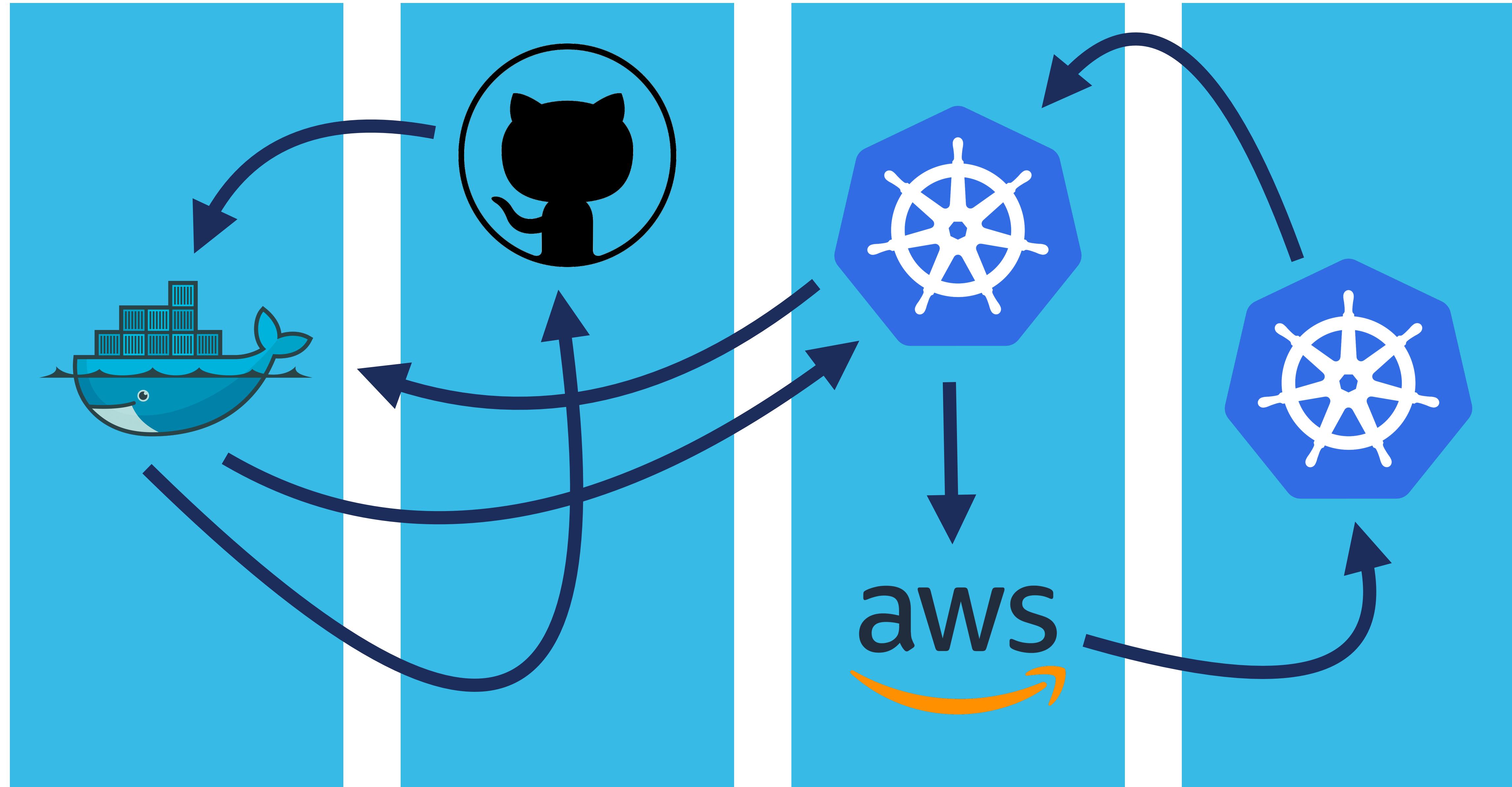
The Serverless Chasm of Doom

- 30 minute build + wait cycles
- No replication of prod bugs
- Need a PhD in AWS



- AWS
- DynamoDB
- API Gateway
- Step Functions

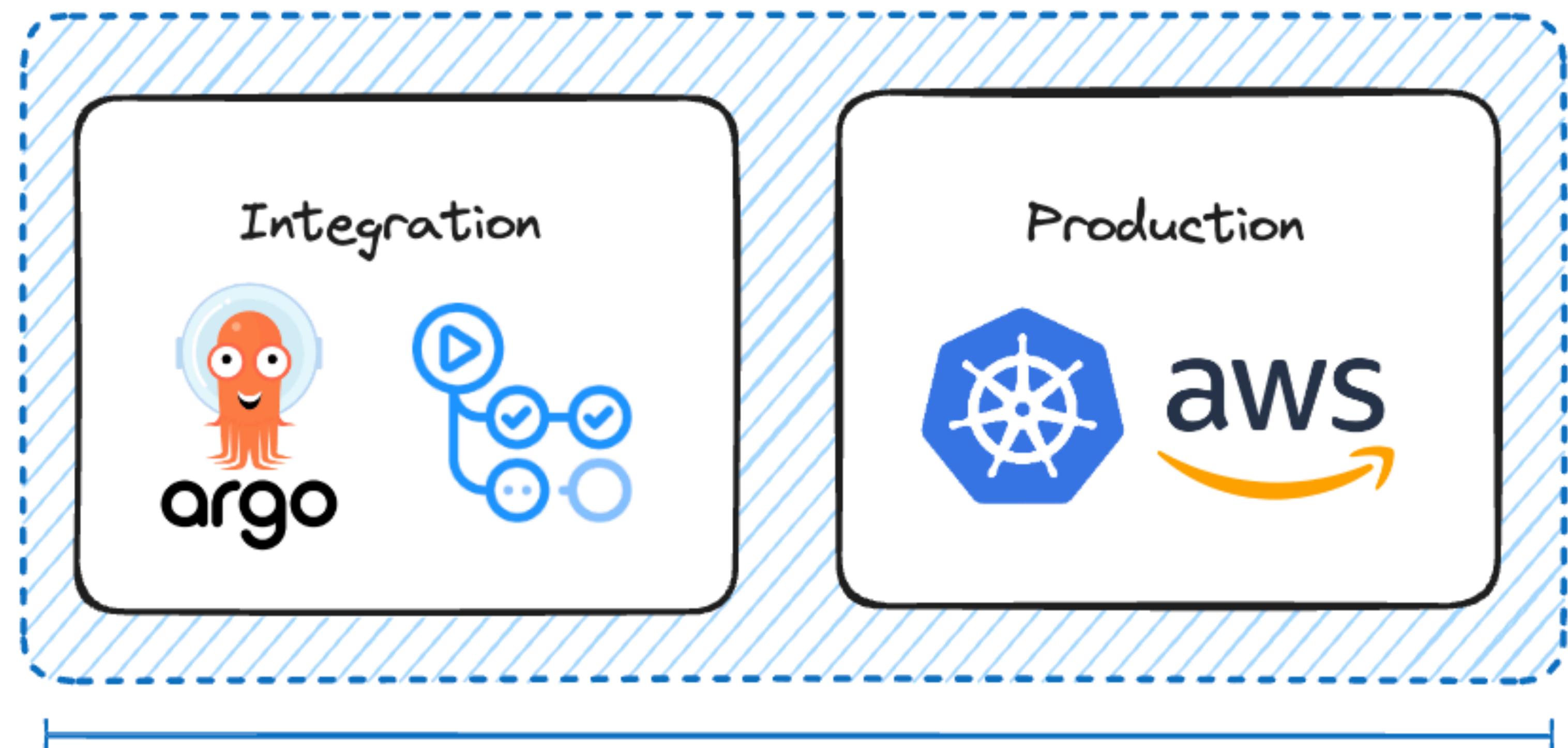
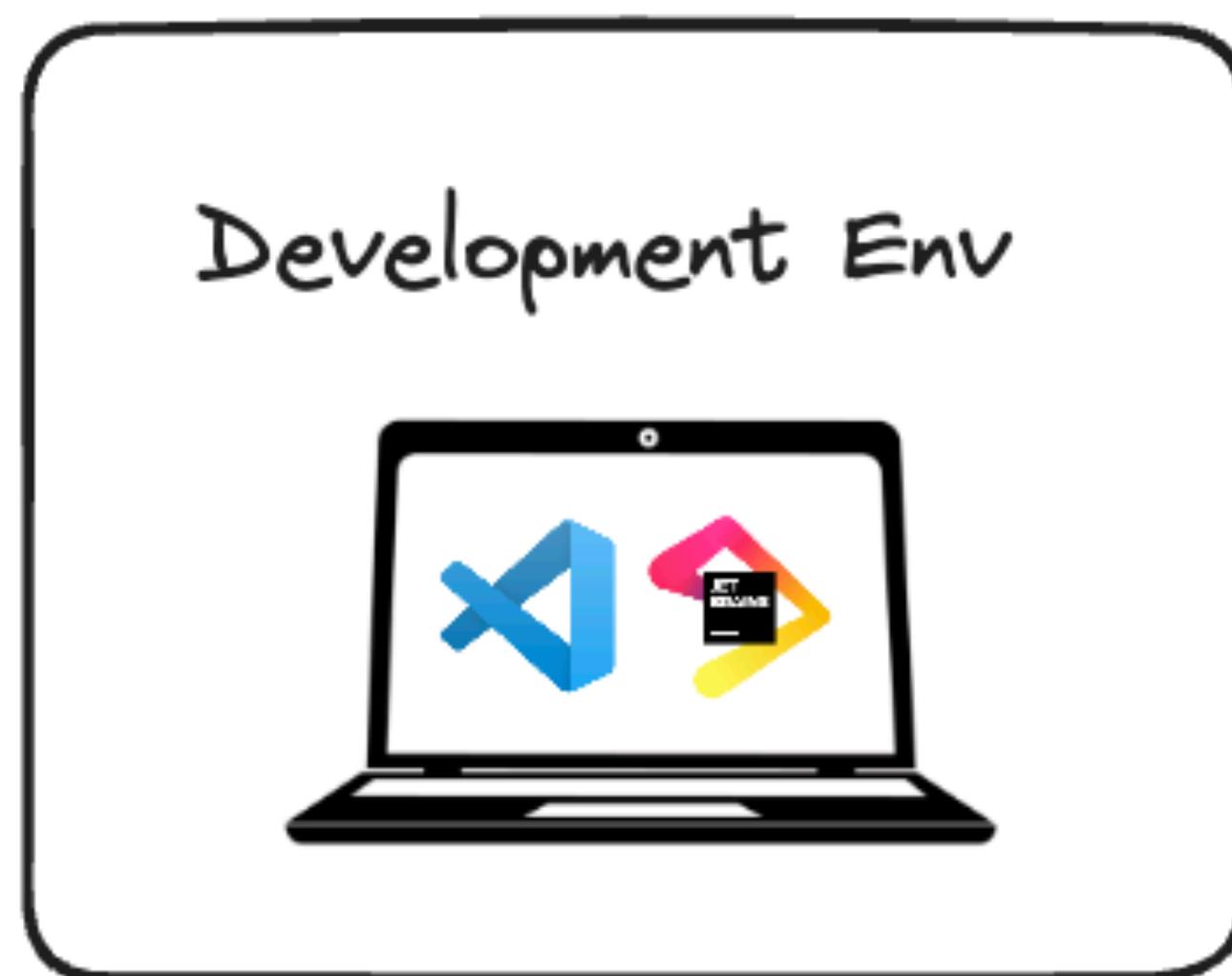




← INNER LOOP →

← OUTER LOOP →

Sphere of influence of most Platform Teams



outer loop emphasis

WHEN DEVS ARE IN CHARGE

INNER LOOP BIAS



WHEN DEVS ARE IN CHARGE

INNER LOOP BIAS



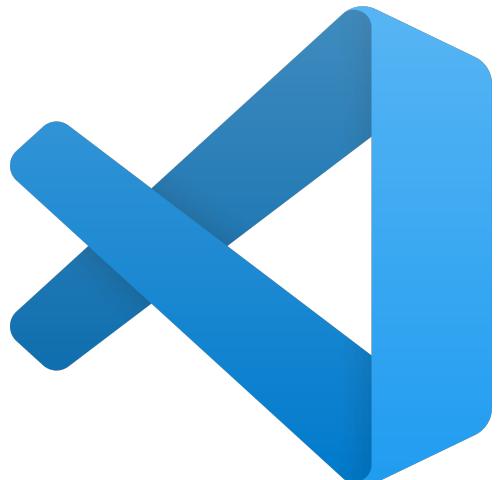
WHEN INFRA ARE IN CHARGE

OUTER LOOP BIAS



DRIFT

INNER LOOP

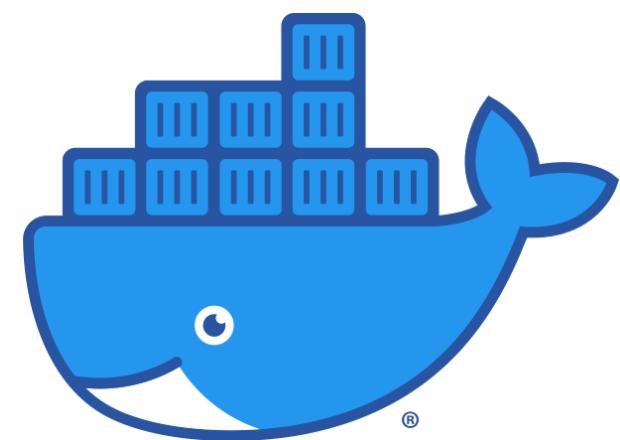
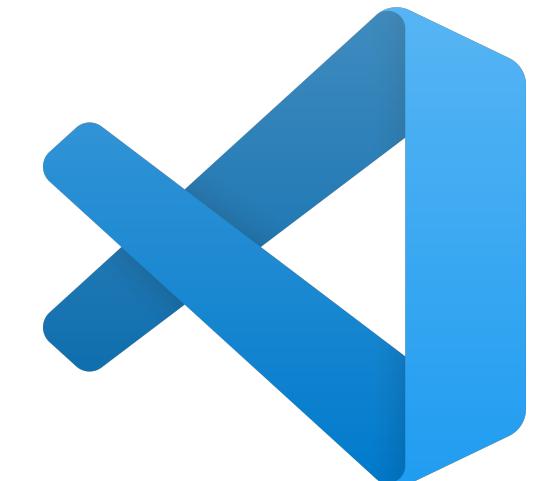


OUTER LOOP



INNER LOOP

aws



WHEN DEVS ARE IN CHARGE

INNER LOOP BIAS



WHEN INFRA ARE IN CHARGE

OUTER LOOP BIAS



WHEN DEVS ARE IN CHARGE

INNER LOOP BIAS



WHEN INFRA ARE IN CHARGE

OUTER LOOP BIAS



**THE COMMON VISION:
GET DEVS BACK
TO THEIR EDITOR**



HOT RELOAD FILE SYNC

The screenshot shows the Tilt web interface with the following details:

- Resources:** 3/3 (backend, frontend, Tiltfile)
- Logs:** Localhost:5734/ (Initial Build) - Shows the command history and logs for the deployment process.
- Tiltfile:** Shows the Tiltfile code used for deployment.

```
# Deploy: tell Tilt what YAML to deploy
k8s_yaml('app.yaml')

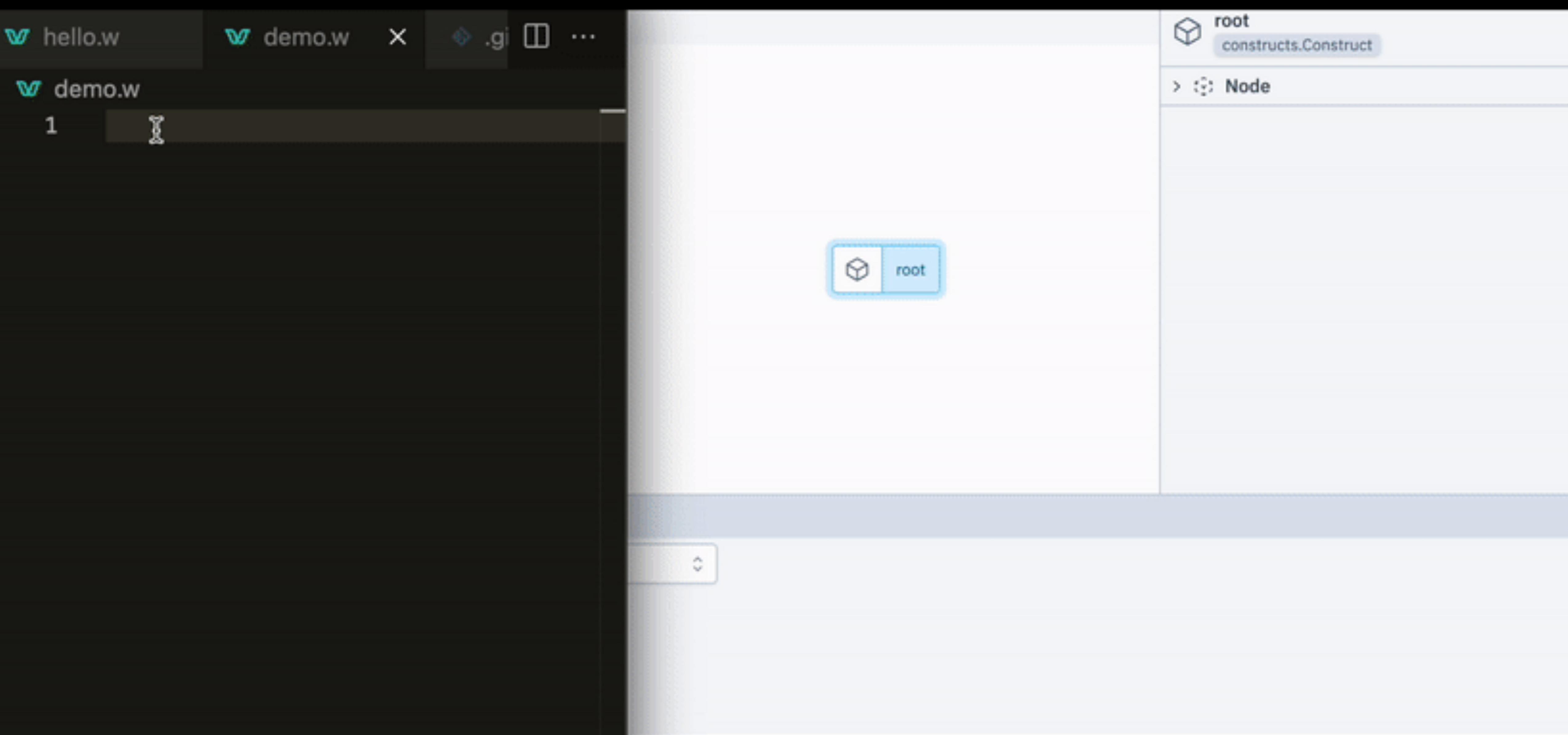
# Build: tell Tilt what images to build from which directories
docker_build('companyname/api', 'api')
docker_build('companyname/web', 'web')
# ...

# Watch: tell Tilt how to connect locally (optional)
k8s_resource('api', port_forwards="5734:5000", labels=["backend"])
```

Attaching to existing pod (api-69fdd74f9d-w62gw). Only new logs will be streamed.

WING

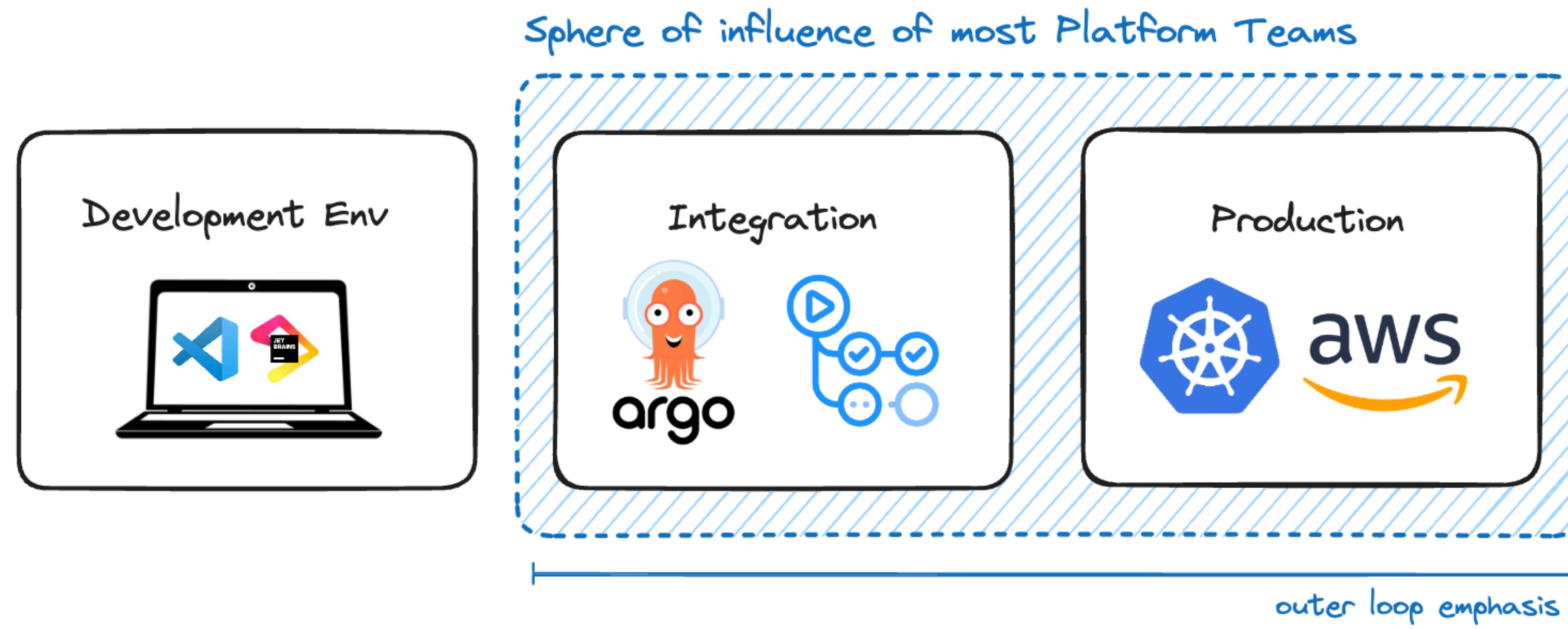


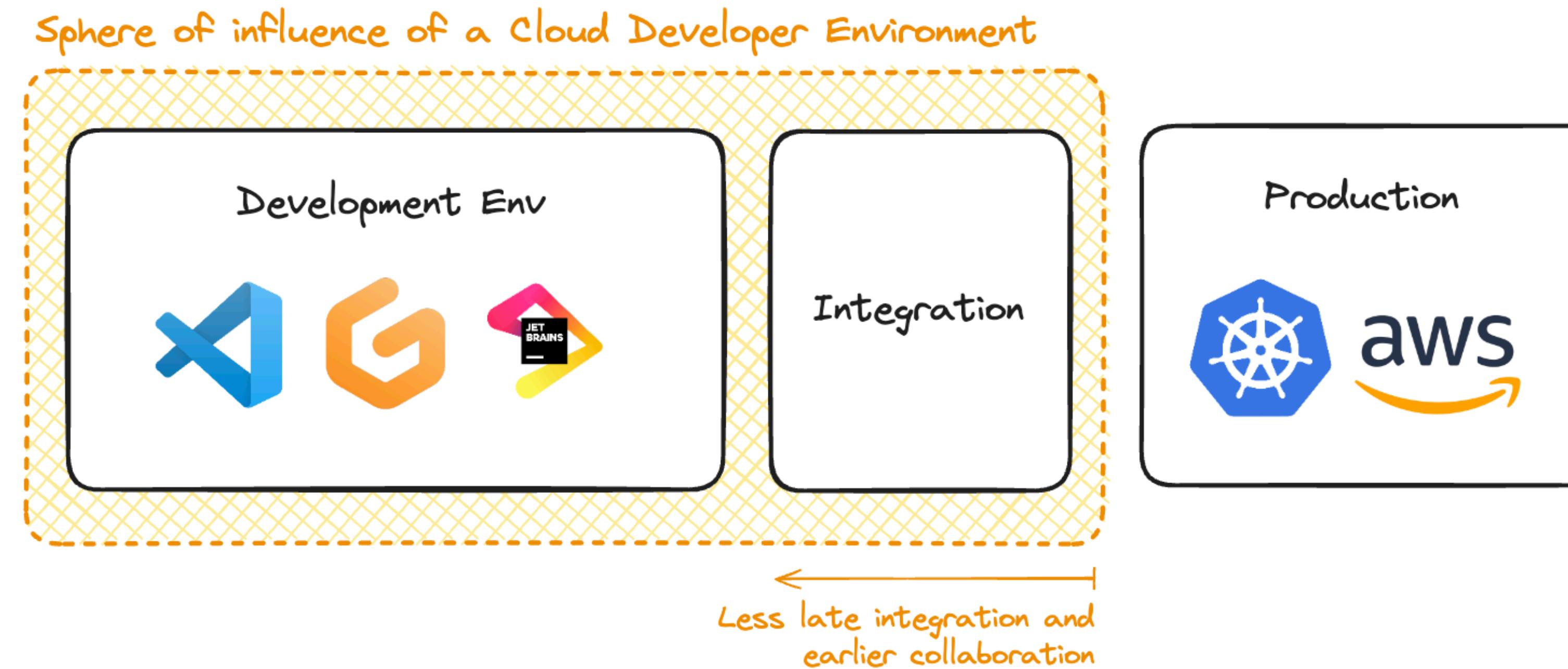
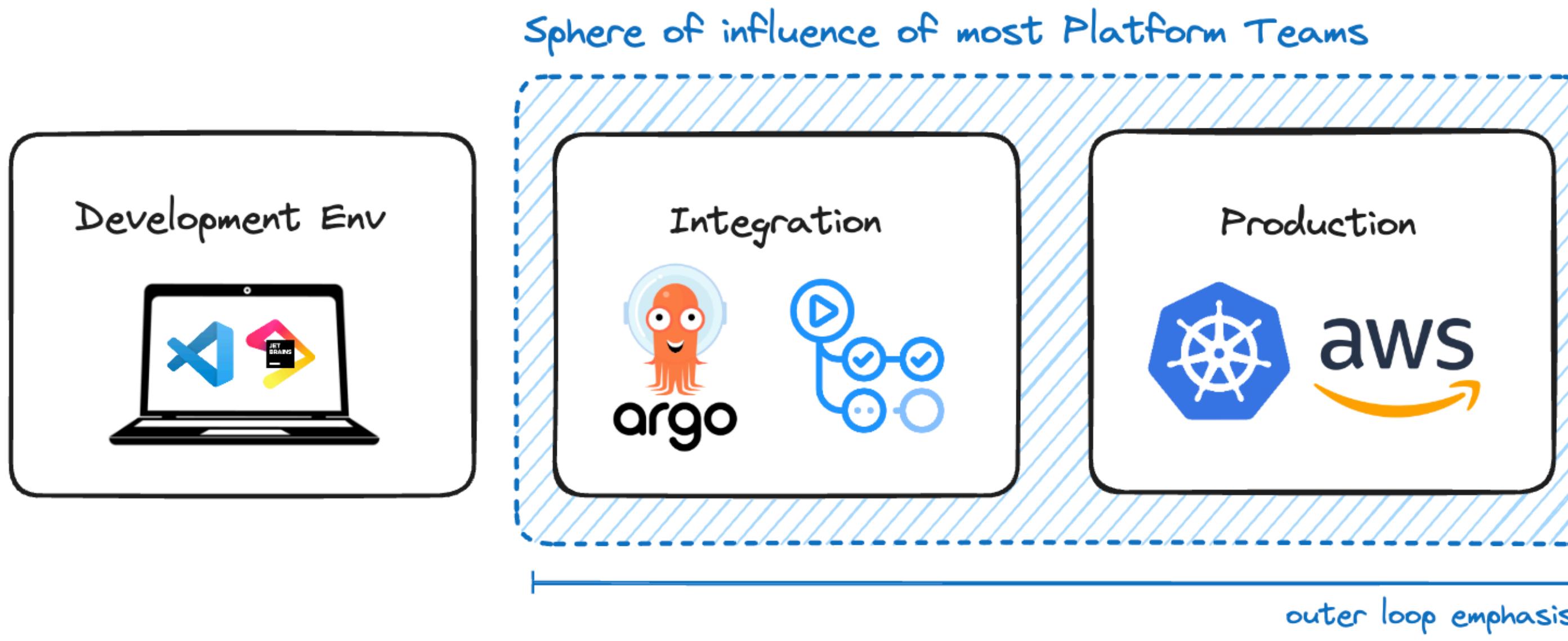


**“WITH THE CDK WE DIDN’T MANAGE TO
ENCAPSULATE BOTH THE APPLICATION
AND THE INFRASTRUCTURE TOGETHER
IN A SINGLE MODEL”**











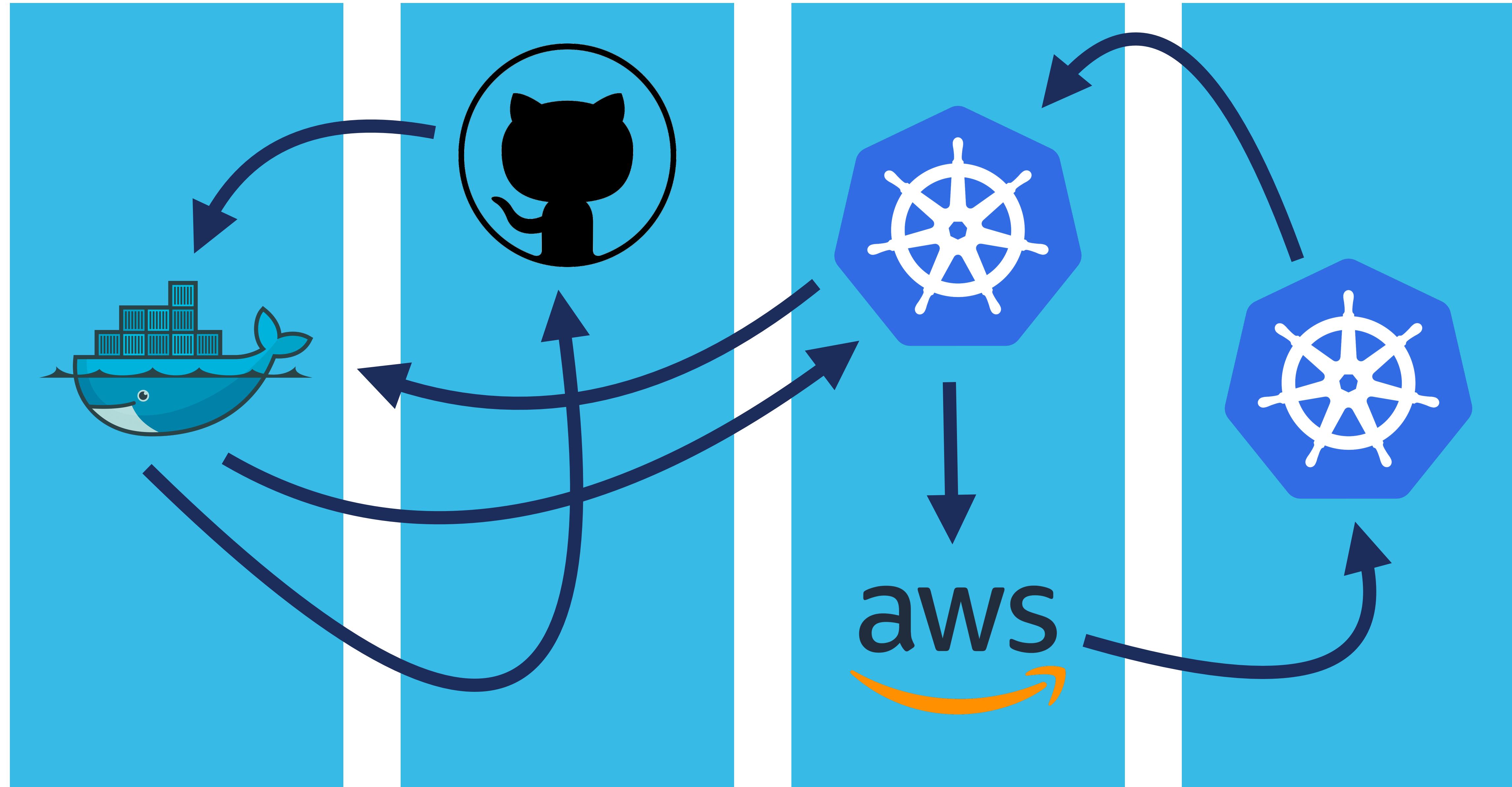
loubichard — fish /workspace/terminal-output — ssh - gitpod workspace ssh loujaybee-terminaloutpu-or5e1jm38yr — 85x23

loujaybee-terminaloutpu-or5e1jm38yr

/w/terminal-output

main ±



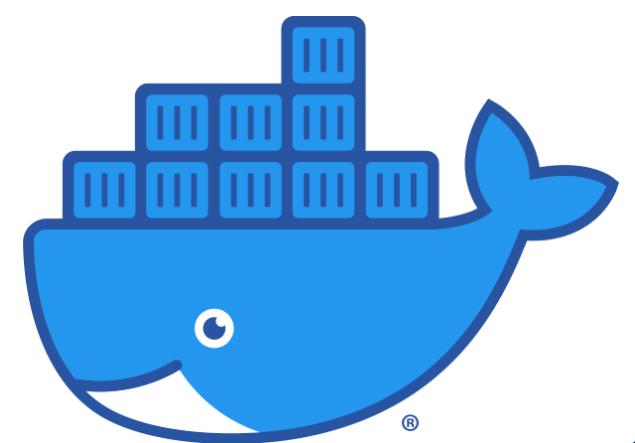
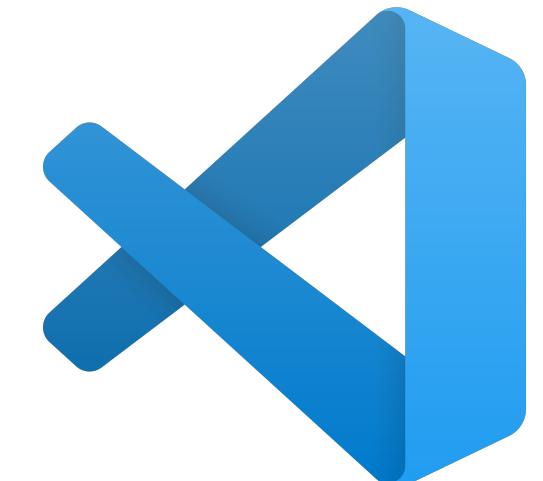


← INNER LOOP →

← OUTER LOOP →

INNER LOOP

aws



COMPILER, IDE

ABSTRACTING APPS

FROM INFRA

IS REALLY HARD

CLOUD, DISTRIBUTED



THERE ARE OTHERS



S K A F F O L D



Dagger



:!> darklang

TO BUILD GREAT DEVELOPER EXPERIENCE

GO BEYOND INNER &

OUTER LOOP THINKING

COLLABORATE WITH

PLATFORM TEAMS

MEET DEVELOPERS

WHERE THEY'RE AT

THANK YOU

@LOUJAYBEE



Gitpod