

Code-to-Cloud



Insight DevOps Fellowship, New York

Wen Gong

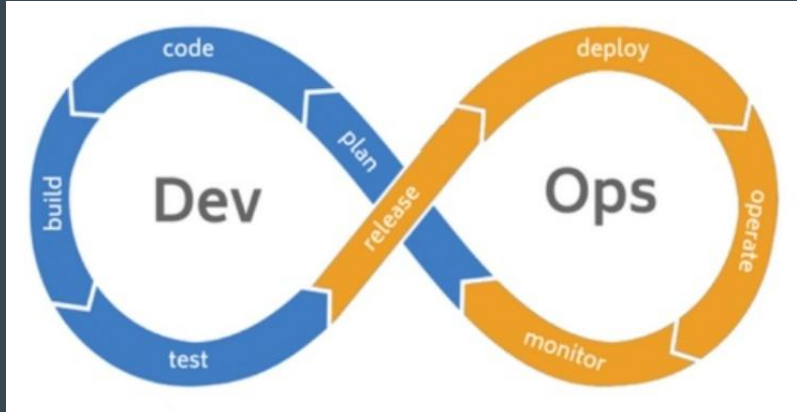
Motivation

Problems

- monolithic app
- dev not test-driven
- lack of tests
- build not automated
- slow approval

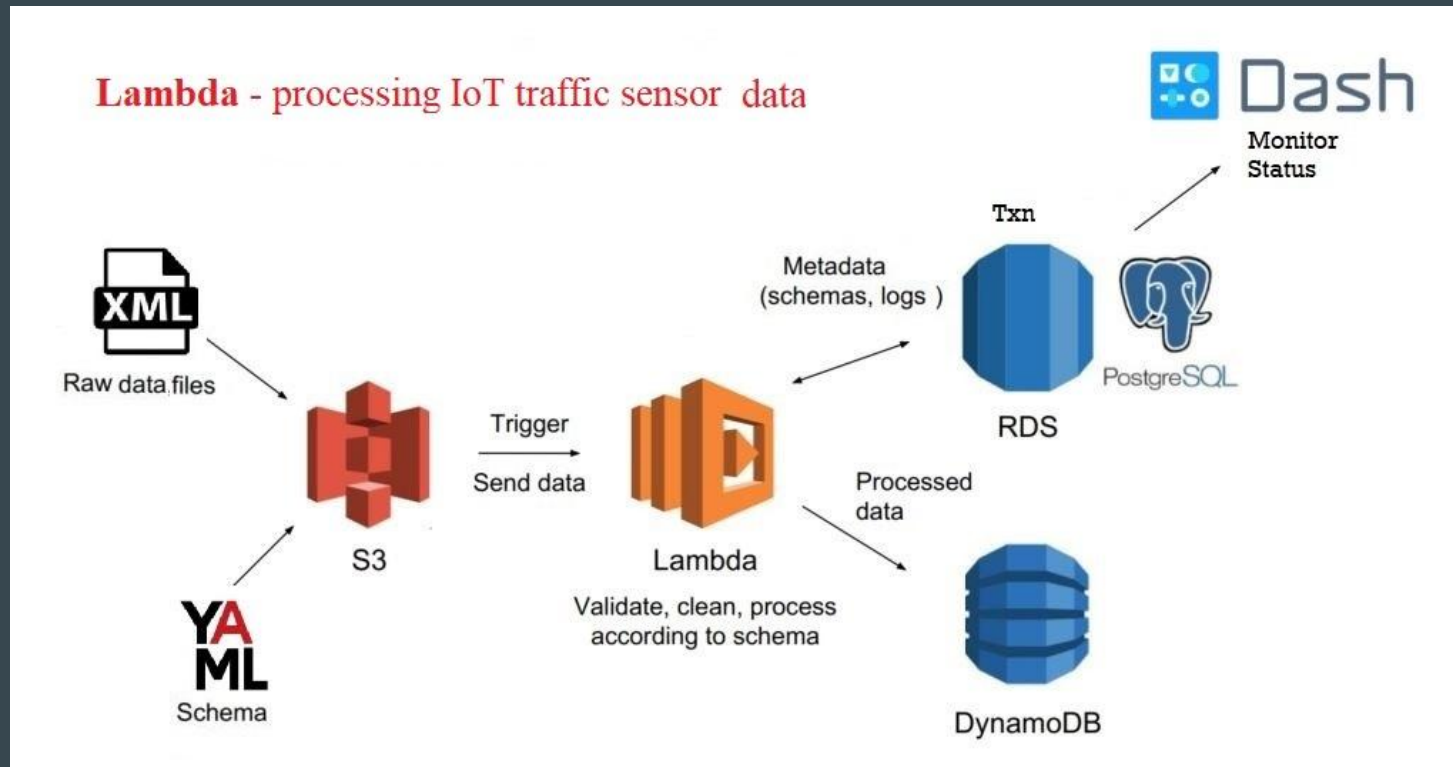
Promise

- faster time-to-market cycle
- productivity gain
- cost savings

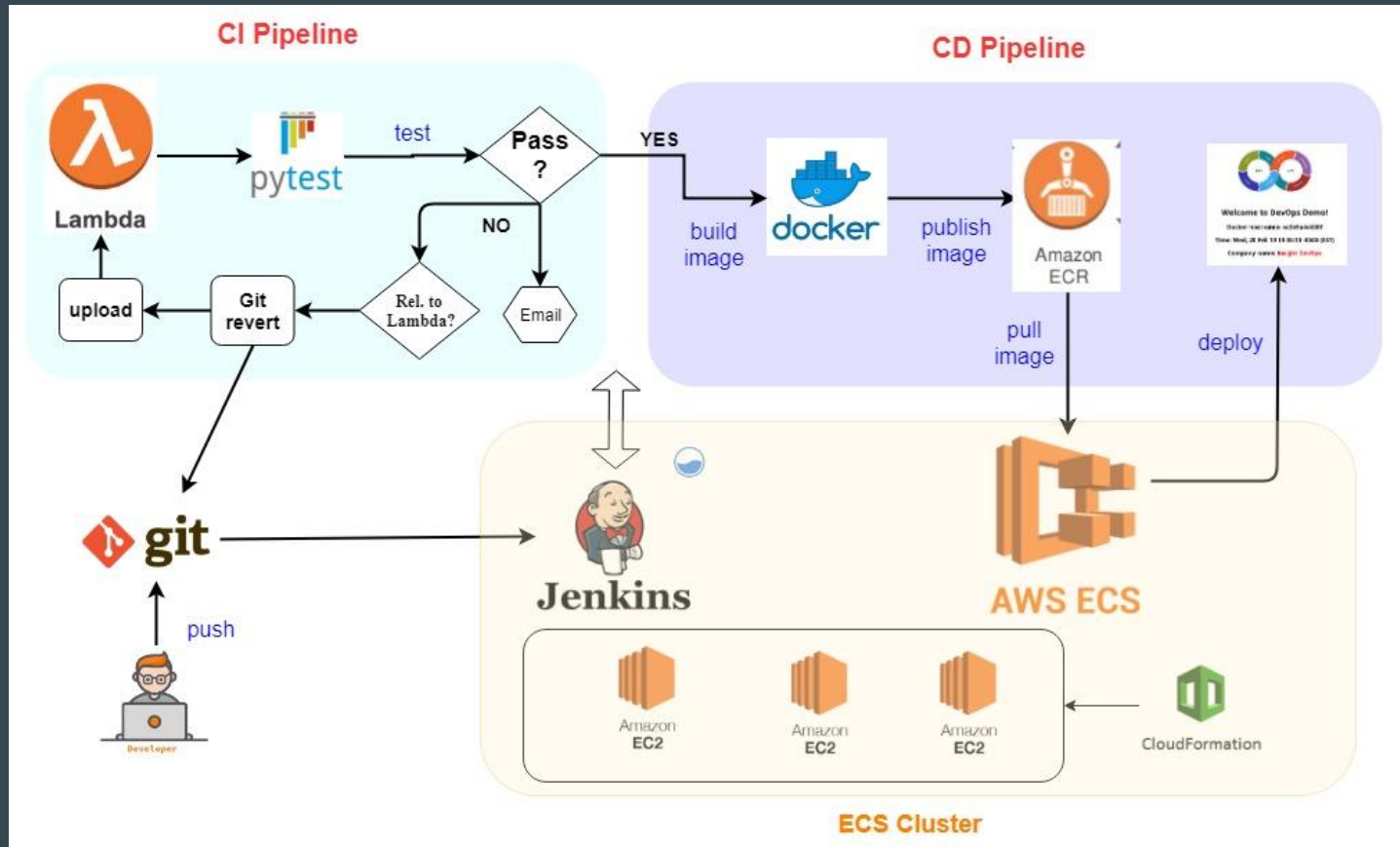


Data Eng. Pipeline

- 1) enhanced logging
- 2) added E2E test
- 3) included in CI



DevOps Pipeline: CI + CD



Challenge - Improved DE pipeline

- added pytest: `test_lambda_function_xml.py`
- enhanced logging by adding new table: `xml_txns`
- improved XML parsing
- fixed issue when uploading 2 files within the same minute

Challenge - Worked around Jenkins plugin error

```
# login
$(aws ecr get-login --no-include-email --region=us-east-1)

#Store the repositoryUri as a variable
REPOSITORY_URI=`aws ecr describe-repositories --repository-names ${RE}

# workaround to fix Docker build and publish plugin issue
TAG_NAME=${REPOSITORY_NAME}:v_${BUILD_NUMBER}
IMG_NAME=${REPOSITORY_URI}:v_${BUILD_NUMBER}
docker build --tag=${TAG_NAME} .
docker tag ${TAG_NAME} ${IMG_NAME}
docker push ${IMG_NAME}

#Replace the build number and repository URI placeholders with the c
sed -e "s;%BUILD_NUMBER%;${BUILD_NUMBER};g" -e "s;%REPOSITORY_URI%;${
#Register the task definition in the repository
aws ecs register-task-definition --family ${FAMILY} --cli-input-json
SERVICES=`aws ecs describe-services --services ${SERVICE_NAME} --clus
#Get latest revision
REVISION=`aws ecs describe-task-definition --task-definition ${NAME}
```

Challenge - DB secrets in Lambda/Docker

- used python client for Vault (hvac)

```
import hvac # python client for Vault
client = hvac.Client(url='http://127.0.0.1:8200/', \
    token='s.1gXcsuHjPZKH3qecpqc6QtTd')
db_secrets = client.read('kv/postgresql_dev')
db_host,db_port,table_name,db_user,db_pwd = \
    db_secrets['data']['hostname'], \
    db_secrets['data']['port'], \
    db_secrets['data']['db_name'], \
    db_secrets['data']['user'], \
    db_secrets['data']['pwd']
```

Demo

CD	walk through from git push to service deployed in AWS (https://youtu.be/dxnUyEwXZhw)
CI	walk through from git push to docker build ready (https://youtu.be/rKY6u4Z6AzM)
DE	walk through DE pipeline (https://youtu.be/qpPLTKGVnkc)

Wen Gong



- Ph.D. in nuclear physics
- 20 years of experience as Software Engineer & Consultant

CRM & master database applications
data quality
integration bridges
ETL

- Passionate about STEM
teach py4kids

ORACLE®

Questions

Links

- <https://github.com/wgong/code2cloud>
- <http://jenkins.s8s.cloud/>

PyTest

README.md

test_lambda_function_xml.py

```
56
57 db_connection_string = f"dbname='{db_name}' user='{db_user}' host='{db_host}' password='{password}'"
58 connection = psycopg2.connect(db_connection_string)
59
60 # download traffic data
61 tmpfile="/tmp/trafficspeed.xml.gz"
62 cmd = f"wget -O {tmpfile} http://opendata.ndw.nu/trafficspeed.xml.gz"
63 wget=subprocess.run(cmd.split(), stdout=subprocess.PIPE).stdout.decode('utf-8')
64
65 cmd = f"env TZ=Europe/Amsterdam date +%Y-%m-%d:%H%M"
66 dt=subprocess.run(cmd.split(), stdout=subprocess.PIPE).stdout.decode('utf-8')
67
68 s3_filename = f"Traffic/{dt[:10]}/{dt[11:15]}_Trafficspeed.gz"
69 cmd = f"aws s3 cp {tmpfile} s3://{s3bucket}/{s3_filename}"
70 aws_s3_cp=subprocess.run(cmd.split(), stdout=subprocess.PIPE).stdout.decode('utf-8')
71
72 cd,msg = get_txn_status(connection, s3_filename, poll_freq, poll_timeout)
73 connection.close()
74
75 return cd,msg
76
77 def test_lambda_xml():
78     status_cd, status_msg = lambda_xml()
79     print(status_cd, status_msg)
80
81     if BREAK_BUILD and status_cd != 0:
82         cmd = f"git revert HEAD --no-edit"
83         git_revert=subprocess.run(cmd.split(), stdout=subprocess.PIPE).stdout.decode('utf-8')
84
85         cmd = f"git push origin HEAD:master"
86         git_revert=subprocess.run(cmd.split(), stdout=subprocess.PIPE).stdout.decode('utf-8')
87
88     assert status_cd == 0
89
```

Jenkins: Pipeline/Jenkinsfile

File Edit Selection View Go Debug Terminal Help Jenkinsfile - vscode-ws (Workspace) - Visual Studio Code

EXPLORER

OPEN EDITORS

- aws-cli.README.md README
- aws-setup.README.md REA...
- TODO.README.md README
- INSTALL.md insight-proj... 9+
- Jenkinsfile insight-project M
- app.py insight-project • src/dash
- lambda_function_xml.py... 1
- logs.py insight-project • sr... 1
- retrieve-realtime-data.sh ins...

VS CODE - WORKSPACE

- go
- hello-aws-docker
- insight-project
 - images
 - jenkins
 - src
 - test
 - sample_Trafficspeed.xml.gz
 - Traffic.yml
 - .gitignore
 - INSTALL.md 9+
 - Jenkinsfile M

OUTLINE

```
1 pipeline {
2   agent any
3   stages {
4     stage('Build') {
5       steps {
6         sh 'echo "upload Lambda function"'
7         sh '''
8           ./upload-lambda-xml.sh
9         '''
10      }
11    }
12    stage('Test') {
13      steps {
14        sh 'echo "run integration test"'
15        sh '''
16          source ~/.aws/env_var
17          echo ${AWS_S3_BUCKET}
18          echo ${AWS_PG_DB_HOST}
19          echo "activate py3"
20          source ~/py36/bin/activate
21          which python3
22          which pytest
23          echo "run pytest"
24          cd src/lambda_xml
25          pytest -q test_lambda_function_xml.py
26        '''
27      }
28    }
29  }
```

master* Python 3.6.5 64-bit ('base': conda) 1 177 Ln 8, Col 43 Spaces: 4 UTF-8

Jenkins: Build Config

← → ↻ 🏠 ⓘ Not secure | ec2-52-3-227-246.compute-1.amazonaws.com/job/insight-project/configure 🔍 ☆ 📦 ⚙️ K 🔄 🌐 📄 Upd. 🗨️

Jenkins ▶ insight-project ▶

General

Branch Sources

Build Configuration

Scan Repository Triggers

Orphaned Item Strategy

Health metrics

Properties

JIRA

Pipeline Libraries

Vault Plugin

Pipeline Model Definition

Build Configuration

Mode

by Jenkinsfile ▼

Script Path

Jenkinsfile ?

Scan Repository Triggers

☐ Monitor Docker Hub/Registry for image changes ?

☒ Periodically if not otherwise run ?

Interval

2 hours ▼ ?

Jenkins: Dashboard

✓ insight-project
< 56

Pipeline

Changes

Tests

Artifacts

Logout

Branch: master

24s

No changes

Commit: —

9 hours ago

Push event to branch master

Start

Build

Test

Deploy

End

Test - 15s

Restart Test

✓ > echo "run integration test" — Shell Script <1s

✓ > pwd whoami which git git --version source ~/.aws/env_var # echo \${AWS_S3_BUCKET} # ... — Shell Script 15s

Docker-build-and-publish

```
29 https://docs.docker.com/engine/reference/commandline/login/#credentials-store
30
31 Login Succeeded
32 Sending build context to Docker daemon 2.107MB
33 Step 1/7 : FROM alpine:3.4
34 ----> b955fa398a69
35 Step 2/7 : RUN apk --update add nginx php5-fpm && mkdir -p /var/log/nginx && touch
/var/log/nginx/access.log && mkdir -p /src && mkdir -p /run/nginx
36 ----> Using cache
37 ----> 17a108601ae9
38 Step 3/7 : ADD www /www
39 ----> d4e48ae60670
40 Step 4/7 : ADD nginx.conf /etc/nginx/
41 ----> defe512ec041
42 Step 5/7 : ADD php-fpm.conf /etc/php5/php-fpm.conf
43 ----> a469f6ae4d54
44 Step 6/7 : EXPOSE 80
45 ----> Running in e10388e9c253
46 Removing intermediate container e10388e9c253
47 ----> 367b6cf55307
48 Step 7/7 : CMD (php-fpm -d variables_order="EGPCS") && (tail -F /var/log/nginx/access.log &) && exec nginx -
g "daemon off;"
49 ----> Running in 2972b4bab732
50 Removing intermediate container 2972b4bab732
51 ----> 787b8c9ef055
52 Successfully built 787b8c9ef055
53 Successfully tagged hello-aws-docker:v_44
54 The push refers to repository [629309645488.dkr.ecr.us-east-1.amazonaws.com/hello-aws-docker]
55 28d322e53d2b: Preparing
```