**Oracle To Postgres**

**Introduction:**

We have setup POD for doing the db migration

Which will help to do the schema migration and data migration from oracle to postgres.

apiVersion: apps/v1

kind: Deployment

metadata:

name: ora2pg-v2

namespace: utils

spec:

replicas: 1

selector:

matchLabels:

app: ora2pg-v2

version: v1

template:

metadata:

labels:

app: ora2pg-v2

version: v1

spec:

containers:

- name: ora2pg-v2

image: ${docker-image}

resources:

requests:

cpu: 100m

memory: 128Mi

limits:

cpu: 250m

memory: 256Mi

command:

- sleep

- "infinity"

imagePullPolicy: IfNotPresent

restartPolicy: Always

Deployement.yaml

**Steps to Get into the POD:**

Steps:

1. Check the pods in utils namespace

CMD – kubectl get pods -n utils

Output – POD : ora2pg-v2-5d8dd9554d-htwz5 (identified by ora2pg-v2 prefix)

1. Enter into the POD

CMD - kubectl exec -it ora2pg-v2-5d8dd9554d-htwz5 -n utils -- /bin/bash

**Schema Migration:**

Tools : ora2pg (Ref: <https://ora2pg.darold.net/documentation.html>)

Steps:

**Generate DDL COMMAND**

1. Go to the root directory

CMD – cd $HOME

1. Create the service directory for schema migration

CMD – mkdir $SERVICE\_NAME && cd $SERVICE\_NAME && mkdir schema && cd schema && mkdir logs && mkdir output && mkdir config

1. Create ora2pg.conf file as per your requirement (Ref: <https://github.com/Guy-Incognito/ora2pg/blob/master/config/ora2pg.conf>)

CMD – vi ./config/ora2pg.conf

1. Run

CMD - ora2pg -c ./config/ora2pg.conf >> ./logs/ora2pg.log 2>&1 &

1. Track the progress from top cmd or ps -ef or tracking ./logs/ora2pg.log file

**Execute DDL COMMAND**

1. Once process is done execute the output file through postgres-client

CMD - psql -h ${postgres IP} -d postgres -U ${postgres\_user} -f ./output/output.sql

**ERROR HANDLING**

1. In case of any Error need to rectify from ora2pg.conf than generate output file again or handle it manually

**Note** : Make sure in your ora2pg.conf file contain

LINE : OUTPUT\_DIR $HOME/$SERVICE\_NAME/schema/output

LINE : OUTPUT output.sql

**Data Migration:**

Tools : pgloader (Ref: <https://pgloader.io/about/>)

Steps:

1. Go to the service directory

CMD – cd $HOME/$SERVICE\_NAME

1. Take your all table dump in csv format backup folder

CMD – mkdir backup

1. Create the directory for data migration

CMD – mkdir data && cd data && mkdir $TABLE\_NAME && cd $TABLE\_NAME && mkdir config && mkdir logs && mkdir error

1. Create the pgloader.conf file as per your requirement (Ref: <https://pgloader.readthedocs.io/en/latest/ref/csv.html>)

CMD – vi ./config/pgloader.load

1. RUN

CMD – pgloader ./config/pgloader.load >> ./logs/pgloader.log 2>&1 &

1. Track the progress from top cmd or ps -ef or tracking ./logs/pgloader.log file

**ERROR HANDLING**

1. In case of any error record rectify it from pgloader.load file or manually handle it from ./error/reject.log file

**Note :** Make sure in your pgloader.load file contain within WITH clause

LINE : logfile = '$HOME/$SERVICE\_NAME/data/logs/progress.log’,

LINE : reject log = '$HOME/$SERVICE\_NAME/data/error/reject.log'