Important Commands

**To set up GOOGLE\_APPLICATION\_CREDENTIALS**

*Linux - export GOOGLE\_APPLICATION\_CREDENTIALS=/home/shrutijoshi/eternal-sylph-336115-ecfcba713fbc.json*

*Windows - set GOOGLE\_APPLICATION\_CREDENTIALS=C:\Documents\Learning\SVC\fabric-1333-29edd96ad1f1.json*

**To Configure service account with google cloud SDK**

*gcloud auth activate-service-account svc-rw@fabric-1333.iam.gserviceaccount.com --key-file=C:\Documents\Learning\SVC\fabric-1333-29edd96ad1f1.json --project=fabric-1333*

*gcloud auth activate-service-account svc-dezyre@eternal-sylph-336115.iam.gserviceaccount.com --key-file=eternal-sylph-336115-ecfcba713fbc.json --project=eternal-sylph-336115*

**Install Python modules for apache beam**

pip install apache-beam[gcp]

**To Run Batch Dataflow Job from Console using Direct Runner**

*python3 flights\_batch\_templated.py --input\_path=gs://dezyre-bucket/data/flights\_sample.csv -–side\_input\_path=gs://dezyre-bucket/data/airlines.csv --table=eternal-sylph-336115:dataset1.flights\_side --error\_table=eternal-sylph-336115:dataset1.error --staging\_location gs://dezyre-bucket/dataflow/staging --temp\_location gs://dezyre-bucket/dataflow/temp --project eternal-sylph-336115 --job\_name flight-test-batch-bq*

**To create template of dataflow job**

*python3 flights\_batch\_templated.py --runner DataflowRunner --project eternal-sylph-336115 --region us-west1 --staging\_location gs://dezyre-bucket/dataflow/staging --temp\_location gs://dezyre-bucket/dataflow/temp --num\_workers 1 --max\_num\_workers 2 --template\_location gs://dezyre-bucket/batchtemplate --save\_main\_session True*

**Install Python api client**

pip3 install google-api-python-client

pip3 install fastapi==0.62.0

pip3 install uvicorn==0.13.1

**To run dataflow template from python code**

python3 dataflow\_template\_call.py

**To create microservice for running dataflow template from python code**

uvicorn dataflow\_template\_service:app --proxy-headers --host localhost --port 8080 –reload

**To create a temporary tunnel**

gcloud compute ssh instance-1 --project eternal-sylph-336115 --zone us-central1-c -- -NL 8080:localhost:8080

Install Docker on Debian

<https://docs.docker.com/engine/install/debian/>

sudo apt-get remove docker docker-engine docker.io containerd runc

sudo apt-get update

sudo apt-get install ca-certificates curl gnupg lsb-release

curl -fsSL https://download.docker.com/linux/debian/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg

echo "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/debian $(lsb\_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

sudo apt-get update

sudo apt-get install docker-ce docker-ce-cli containerd.io

sudo docker run hello-world

**Configure docker with gcloud**

sudo gcloud auth configure-docker

I**nstall Kubectl with Gcloud SDK**

gcloud components install kubectl or sudo apt-get install kubectl

**Check the Kubectl version**

kubectl version --client

**Docker Understanding**

<https://docs.docker.com/get-started/>

**Docker reference commands**

<https://docs.docker.com/engine/reference/run/>

**To build Docker Image from DockerFile**

sudo docker build --no-cache -t dezyreimage .

**To delete docker image**

sudo docker image rm dezyreimage

**To run container from image**

sudo docker run -d --name dezyrecontainer -p 80:80 dezyreimage

**To see all the containers**

sudo docker ps -a

**To kill all the stopped containers**

sudo docker container prune

**To get bash terminal of the container**

sudo docker exec -it dezyrecontainer /bin/bash

**To see logs of container**

sudo docker logs dezyrecontainer

**Image push to google container registry**

docker tag dezyreimage:tag1 gcr.io/eternal-sylph-336115/dezyreimage:tag1

docker push gcr.io/eternal-sylph-336115/dezyreimage:tag1

**Common Issues**

1. Permission issue while pushing first image to gcr.io

Resolution – Make sure your service account has storage admin role assigned. Run below commands.

*export GOOGLE\_APPLICATION\_CREDENTIALS=/home/shrutijoshi/eternal-sylph-336115-1740a20541fe.json*

*sudo gcloud auth activate-service-account svc-dezyre-admin@eternal-sylph-336115.iam.gserviceaccount.com --key-file=eternal-sylph-336115-1740a20541fe.json --project=eternal-sylph-336115*

*sudo gcloud auth configure-docker*

**ENABLE below API from cloud console**

**Dataflow API**

**containerregistry.googleapis.com**

**container.googleapis.com**

**Get list of GKE clusters available**

*gcloud container clusters list*

*gcloud config set compute/zone us-central1*

*gcloud container clusters get-credentials autopilot-cluster-1*

**Create a namespace and use it**

kubectl create namespace dev

kubectl get namespace

kubectl config set-context --current --namespace=dev

**See list of nodes available**

kubectl get nodes

**To create the deployment using file**

Kubectl apply -f deploy.yaml

Or

kubectl create deployment data-pipeline --image=gcr.io/eternal-sylph-336115/dezyreimage:tag1

**To see the deployments**

Kubectl get deploy

Kubectl get pods

**To create the service to expose outside**

Kubectl apply -f service.yaml

Or

kubectl expose deployment data-pipeline --type LoadBalancer --port 80 --target-port 80

**To see the service**

Kubectl get service

**Scaling a deployment**

kubectl autoscale deployment data-pipeline --max 3 --min 2 --cpu-percent 50

kubectl get hpa

kubectl get hpa data-pipeline -o yaml

kubectl get pods

kubectl delete hpa

*Extra Documentation*

[*https://kubernetes.io/docs/concepts/workloads/controllers/deployment*](https://kubernetes.io/docs/concepts/workloads/controllers/deployment)

[*https://kubernetes.io/docs/reference/kubectl/cheatsheet/*](https://kubernetes.io/docs/reference/kubectl/cheatsheet/)

[*https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#rollout*](https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#rollout)

[*https://cloud.google.com/dataflow/docs/reference/rest/v1b3/projects.jobs/get*](https://cloud.google.com/dataflow/docs/reference/rest/v1b3/projects.jobs/get)

[*https://cloud.google.com/kubernetes-engine/docs/how-to/scaling-apps*](https://cloud.google.com/kubernetes-engine/docs/how-to/scaling-apps)

[*https://cloud.google.com/kubernetes-engine/docs/how-to/cluster-autoscaler*](https://cloud.google.com/kubernetes-engine/docs/how-to/cluster-autoscaler)

[*https://www.bluematador.com/blog/kubernetes-deployments-rolling-update-configuration*](https://www.bluematador.com/blog/kubernetes-deployments-rolling-update-configuration)

[*https://learnk8s.io/kubernetes-rollbacks*](https://learnk8s.io/kubernetes-rollbacks)