My PROJECT\_ID : chrome-courage-155503

**$ gcloud auth list**

Credentialed Accounts:

- huang.shaow@husky.neu.edu ACTIVE

To set the active account, run:

$ gcloud config set account `ACCOUNT`

**$ gcloud config list project**

[core]

project = chrome-courage-155503

Your active configuration is: [cloudshell-12918]

**$ gcloud config set compute/zone us-central1-f**

Updated property [compute/zone].

**$ nano server.js**

var http = require('http');

var handleRequest = function(request, response) {

response.writeHead(200);

response.end('Hello World!');

};

var www = http.createServer(handleRequest);

www.listen(8080);

**$ node server.js**

Ctrl-C to exit

**$ nano Dockerfile**

FROM node:6.9.2

EXPOSE 8080

COPY server.js

CMD node server.js

**$ docker build -t gcr.io/chrome-courage-155503/hello-node:v1 .**

ending build context to Docker daemon 18.84 MB

Step 1 : FROM node:6.9.2

6.9.2: Pulling from library/node

75a822cd7888: Pull complete

57de64c72267: Pull complete

4306be1e8943: Pull complete

871436ab7225: Pull complete

0110c26a367a: Pull complete

1f04fe713f1b: Pull complete

ac7c0b5fb553: Pull complete

Digest: sha256:2e95be60faf429d6c97d928c762cb36f1940f4456ce4bd33fbdc34de94a5e043

Status: Downloaded newer image for node:6.9.2

---> faaadb4aaf9b

Step 2 : EXPOSE 8080

---> Running in a9b54242785c

---> affed67c168f

Removing intermediate container a9b54242785c

Step 3 : COPY server.js

COPY requires at least one argument

FROM node:6.9.2

EXPOSE 8080

COPY server.js .

CMD node server.js

**$ docker build -t gcr.io/chrome-courage-155503/hello-node:v1 .**

Sending build context to Docker daemon 18.84 MB

Step 1 : FROM node:6.9.2

---> faaadb4aaf9b

Step 2 : EXPOSE 8080

---> Using cache

---> affed67c168f

Step 3 : COPY server.js .

---> 100638749f4c

Removing intermediate container 3070521d5e58

Step 4 : CMD node server.js

---> Running in f81e067fbd7b

---> 4936fbbe967b

Removing intermediate container f81e067fbd7b

Successfully built 4936fbbe967b

**$ docker run -d -p 8080:8080 gcr.io/chrome-courage-155503/hello-node:v1**

a3168d1fbeb1853b2e38f1350ed20b5aa9febef27e02a36cdf83f1bea9d4b11f

**$ curl http://localhost:8080**

Hello World!

**$ docker ps**

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

a3168d1fbeb1 gcr.io/chrome-courage-155503/hello-node:v1 "/bin/sh -c 'node ser" About a minute ago Up About a minute 0.0.0.0:8080->8080/tcp suspicious\_brattain

**$ docker stop a3168d1fbeb1**

a3168d1fbeb1

**$ gcloud docker -- push gcr.io/chrome-courage-155503/hello-node:v1**

The push refers to a repository [gcr.io/chrome-courage-155503/hello-node]

51cde2fbe0cb: Pushed

381c97ba7dc3: Pushed

604c78617f34: Pushed

fa18e5ffd316: Pushed

0a5e2b2ddeaa: Pushed

53c779688d06: Pushed

60a0858edcd5: Pushed

b6ca02dfe5e6: Pushed

v1: digest: sha256:5bce4a00c1b11198ab5dde7f5fe598a86be2bc411cfd7249a67f25db50d62006 size: 2002

**$ gcloud config set project chrome-courage-155503**

Updated property [core/project].

**$ gcloud container clusters create hello-world \**

**> --num-nodes 2 \**

**> --machine-type n1-standard-1 \**

**> --zone us-central1-f**

ERROR: (gcloud.container.clusters.create) ResponseError: code=403, message=Insufficient regional quota to satisfy request for resource: "CPUS". The request requires '2.0' and is short '1.0'. The regional quota is '8.0' with '1.0' available.

I have created some cluster instances on GKE, so the space is not enough. Thus, I DELETE existing Container Engine cluster and try again.

**$ gcloud container clusters create hello-world**

**--num-nodes 2**

**--machine-type n1-standard-1**

**--zone us-central1-f**

Creating cluster hello-world...done.

Created [https://container.googleapis.com/v1/projects/chrome-courage-155503/zones/us-central1-f/clusters/hello-world].

kubeconfig entry generated for hello-world.

NAME ZONE MASTER\_VERSION MASTER\_IP MACHINE\_TYPE NODE\_VERSION NUM\_NODES STATUS

hello-world us-central1-f 1.5.6 35.184.135.60 n1-standard-1 1.5.6 2 RUNNING

**$ kubectl run hello-node \**

**> --image=gcr.io/chrome-courage-155503/hello-node:v1 \**

**> --port=8080**

deployment "hello-node" created

**$ kubectl get deployments**

NAME DESIRED CURRENT UP-TO-DATE AVAILABLE AGE

hello-node 1 1 1 1 1m

**$ kubectl get pods**

NAME READY STATUS RESTARTS AGE

hello-node-3932933177-frxsz 1/1 Running 0 2m

**$ kubectl cluster-info**

Kubernetes master is running at https://35.184.135.60

GLBCDefaultBackend is running at https://35.184.135.60/api/v1/proxy/namespaces/kube-system/services/default-http-backend

Heapster is running at https://35.184.135.60/api/v1/proxy/namespaces/kube-system/services/heapster

KubeDNS is running at https://35.184.135.60/api/v1/proxy/namespaces/kube-system/services/kube-dns

kubernetes-dashboard is running at https://35.184.135.60/api/v1/proxy/namespaces/kube-system/services/kubernetes-dashboard

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.

**$ kubectl expose deployment hello-node --type="LoadBalancer"**

service "hello-node" exposed

**$ kubectl get services**

NAME CLUSTER-IP EXTERNAL-IP PORT(S) AGE

hello-node 10.11.251.120 35.188.108.63 8080:31125/TCP 57s

kubernetes 10.11.240.1 <none> 443/TCP 14m