Imagine, you are using google cloud instances as Kubernetes master and worker nodes

Your requirement is to have persistent mount point inside your pod to store application data

What is the appropriate volume type?

srinathchalla@outlook.com

GCE PersistentDisk

Concept

Udemy

Objectives

Concept

a. gcePersistent Disk

Review Demo

a. Manifest file
b. Create and display
c. Test use case
d. Clean up

srinathchalla@outlook.com

gcePersistentDisk

- Volume mounts a Google Compute Engine (GCE) Persistent Disk into Pod
- Volume data is persisted pods termination
- Read-Write only on one node and Read-Only on many nodes

Restrictions:

- You must create a PD using gcloud or the GCE API or UI before you can use it
- Nodes on which Pods are running must be GCE VMs
- VMs need to be in the same GCE project and zone as the PD

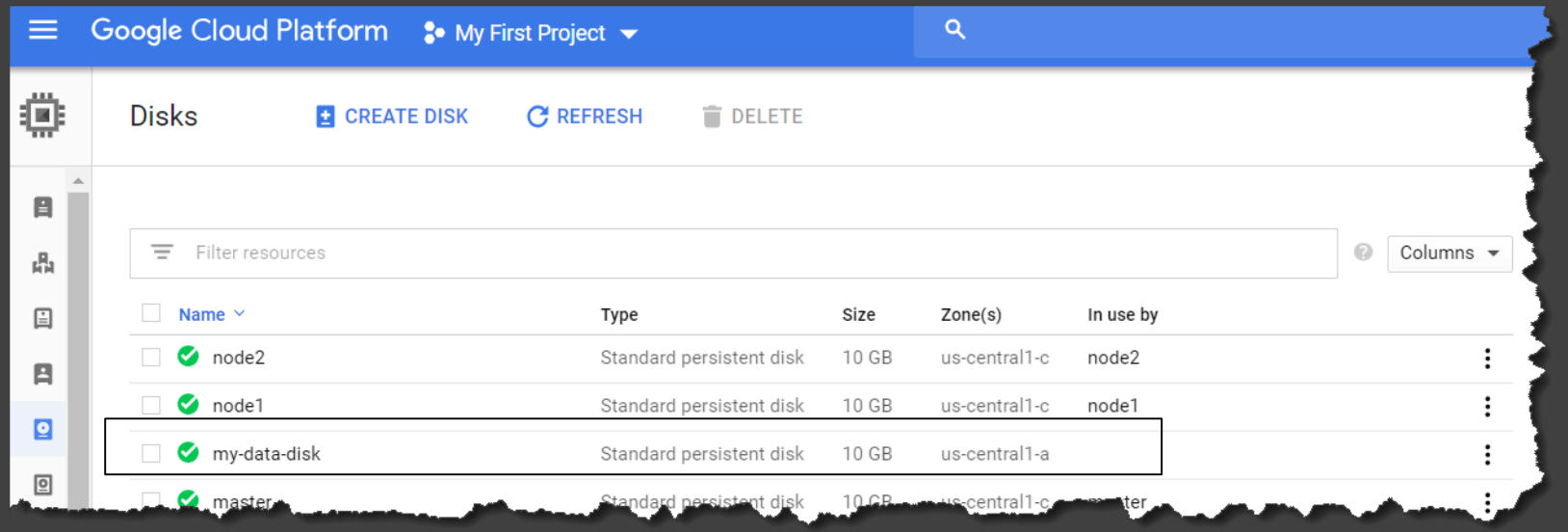
Review
Demo

a. Manifest file
b. Create gcePersistent
d. Display & Validate
e. Clean up

gcePersistentDisk – Create Disk

```
schalla@cloudshell:$ gcloud compute disks create --size=10GB --zone=us-central1-a my-data-disk
WARNING: You have selected a disk size of under [200GB]. This may result in poor I/O performance. For
more information, see: https://developers.google.com/compute/docs/disks#performance.
Created [https://www.googleapis.com/compute/v1/projects/keen-goods-180623/zones/us-central1-
a/disks/my-data-disk].
NAME          ZONE           SIZE_GB  TYPE         STATUS
my-data-disk  us-central1-a  10       pd-standard  READY

New disks are unformatted. You must format and mount a disk before it
can be used. You can find instructions on how to do this at:
https://cloud.google.com/compute/docs/disks/add-persistent-disk#formatting
```



srinathchalla@outlook.com

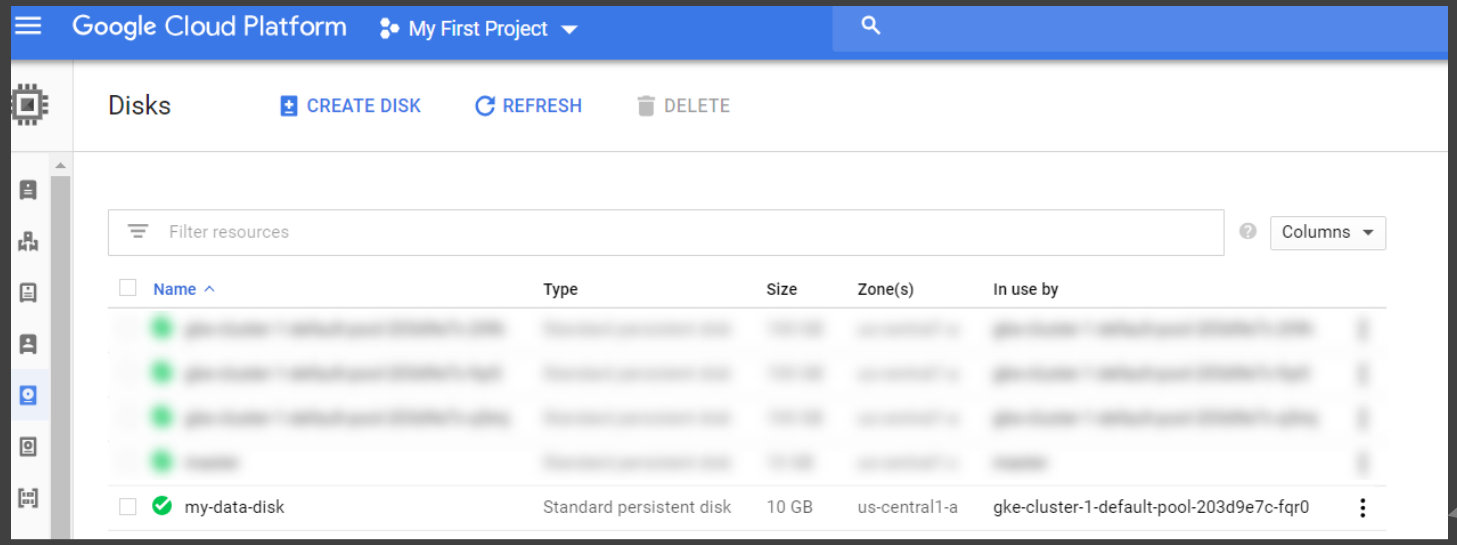
gcePersistentDisk - Config

```yaml
# gcePersistentDisk
apiVersion: v1
kind: Pod
metadata:
  name: gce-pd
spec:
  containers:
  - name: test-container
    image: nginx
    volumeMounts:
    - mountPath: /test-pd
      name: test-volume
  volumes:
  - name: test-volume
    gcePersistentDisk:
      pdName: my-data-disk
      fsType: ext4
```

gcePersistentDisk – Create & Display

```
schalla@cloudshell:$ kubectl create -f test-gcepd.yaml
pod/gce-pd created
```

```
schalla@cloudshell:$ kubectl get po -o wide
NAME      READY     STATUS    RESTARTS   AGE       IP           NODE
gce-pd    1/1       Running   0          1h        10.8.0.29    gke-cluster-1-default-pool-203d9e7c-fqr0
```



srinathchalla@outlook.com

gcePersistentDisk – Describe

```
schalla@cloudshell:$ kubectl describe po gce-pd
Name:         gce-pd
Namespace:    default
Node:         gke-cluster-1-default-pool-203d9e7c-fqr0/10.128.0.6
Start Time:   Thu, 30 Aug 2018 21:57:39 +0530
Labels:       <none>
Annotations:  kubernetes.io/limit-ranger=LimitRanger plugin set: cpu request for container test-
container
Status:       Running
IP:           10.8.0.29
Containers:
  test-container:
    ...
  Mounts:
      /test-pd from test-volume (rw)
      /var/run/secrets/kubernetes.io/serviceaccount from default-token-v6mpn (ro)
Volumes:
  test-volume:
    Type:       GCEPersistentDisk (a Persistent Disk resource in Google Compute Engine)
    PDName:     my-data-disk
    FSType:     ext4
    ...
```

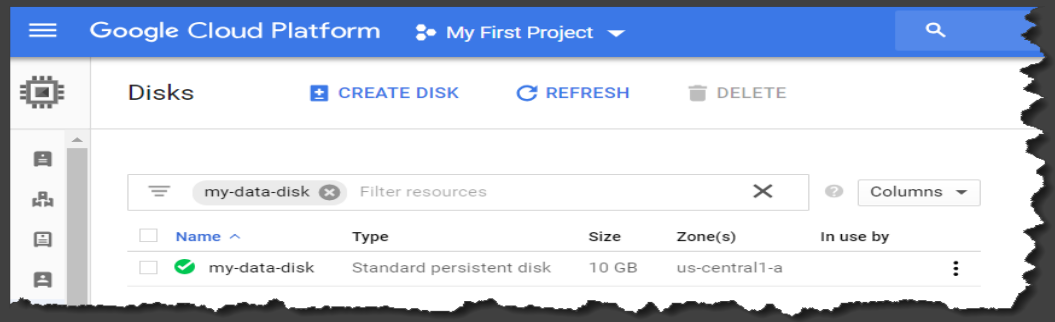
srinathchalla@outlook.com

1. Create a sample test file inside the mount.
2. Delete the Pod
3. Recreate the Pod with same configuration
4. Verify the data created in step-1 is still available?

gcePersistentDisk – Testing

```
schalla@cloudshell:$ kubectl exec gce-pd -it -- /bin/sh
# df
Filesystem     1K-blocks    Used Available Use% Mounted on
overlay         98868448 4720928  94131136   5% /
tmpfs            1897156       0   1897156   0% /dev
tmpfs            1897156       0   1897156   0% /sys/fs/cgroup
/dev/sdb        10255636   36888   9678076   1% /test-pd
/dev/sda1       98868448 4720928  94131136   5% /etc/hosts
shm                65536       0     65536   0% /dev/shm
tmpfs            1897156      12   1897144   1% /run/secrets/kubernetes.io/serviceaccount
tmpfs            1897156       0   1897156   0% /sys/firmware
#
# echo "Testing - 1" > /test-ed/test1.html
# exit
```

```
schalla@cloudshell:$  kubectl delete -f gce-pd.yaml
pod "gce-pd" deleted
```

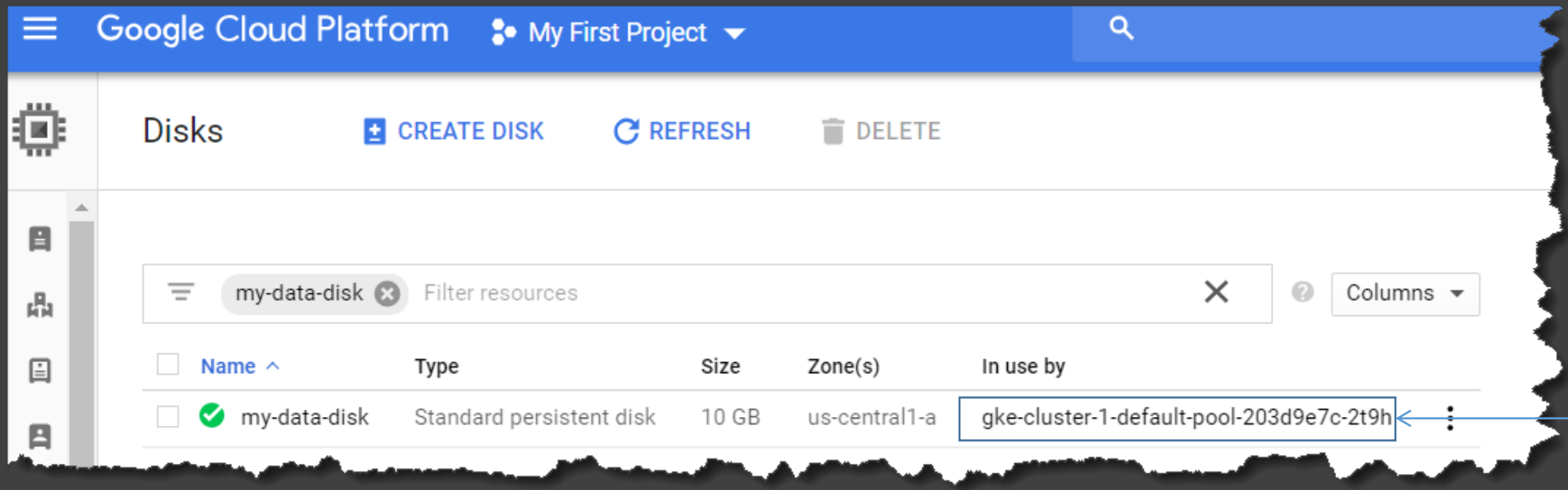


srinathchalla@outlook.com

gcePersistentDisk – Config Create

```
schalla@cloudshell:$ kubectl create -f test-gcepd.yaml
pod/gce-pd created
```

```
schalla@cloudshell:$ kubectl get po -o wide
NAME      READY     STATUS    RESTARTS   AGE       IP           NODE
gce-pd    1/1       Running   0          3m        10.8.1.20    gke-cluster-1-default-pool-203d9e7c-2t9h
```



srinathchalla@outlook.com

gcePersistentDisk – Testing

```
schalla@cloudshell:$ kubectl exec gce-pd df /test-pd
Filesystem     1K-blocks  Used Available Use% Mounted on
/dev/sdb        10255636 36892   9678072   1% /test-pd
```

```
schalla@cloudshell:$ kubectl exec gce-pd ls /test-pd/
lost+found
test1.html
```

```
schalla@cloudshell:$ kubectl exec gce-pd cat /test-pd/test1.html
Testing - 1
```

srinathchalla@outlook.com

Summary

Concept

a. gcePersistent Disk

Review Demo

a. Manifest file
b. Create and display
c. Test use case
d. Clean up

srinathchalla@outlook.com

Demo

gcePersistentDisk