ENGR-E516 Assignment – Map-Reduce with VMs and FaaS

Part 1 – Map Reduce on VMs

Overview

In Part 1, the Map Reduce implementation from Assignment 2 has been deployed on GCP Compute Engine instances. Each mapper/reducer VM is launched/destroyed only from the master. The master also creates the KV store VM for storage. All communication between VMs is TCP socket-based.

Steps to Execute/Test Implementation

- The master VM is already created and it's public IP is 34.140.99.15
  There is a Flask server on port 8081 with below 2 endpoints. It is recommended to use an app like Postman for testing these APIs.

  - **To launch map reduce**
    Note: This process will take roughly 5-8 minutes to complete as the VMs need to be created and the execution environment/dependencies need to set up for all mapper and reducer VMs.

    Send below POST request at endpoint "/launch_map_reduce"

    cURL request

    ```
    curl --location --request POST '34.140.99.15:8081/launch_map_reduce'
    ```

  - **To fetch final output**
    Send below GET request at endpoint "/final_output"

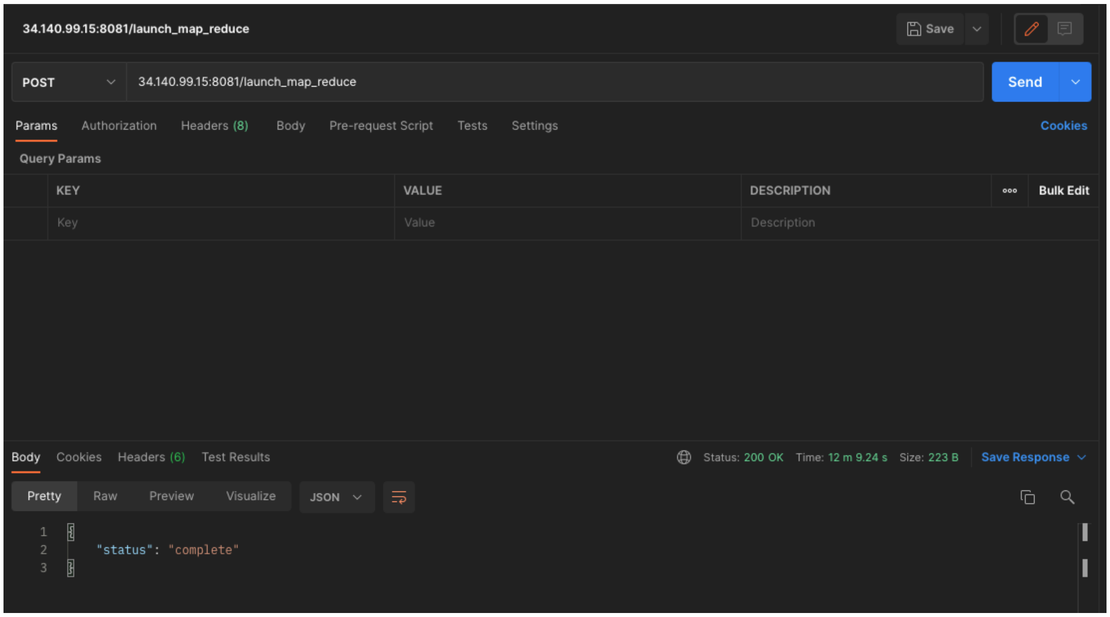
    cURL request

    ```
    curl --location --request GET '34.140.99.15:8081/final_output'
    ```

Test Example using Postman

After the entire map-reducer workflow is complete, the status of “complete” will be seen in the response. Then, one can send a GET request on the “/final_output” API to fetch the result.

Launch Map Reduce



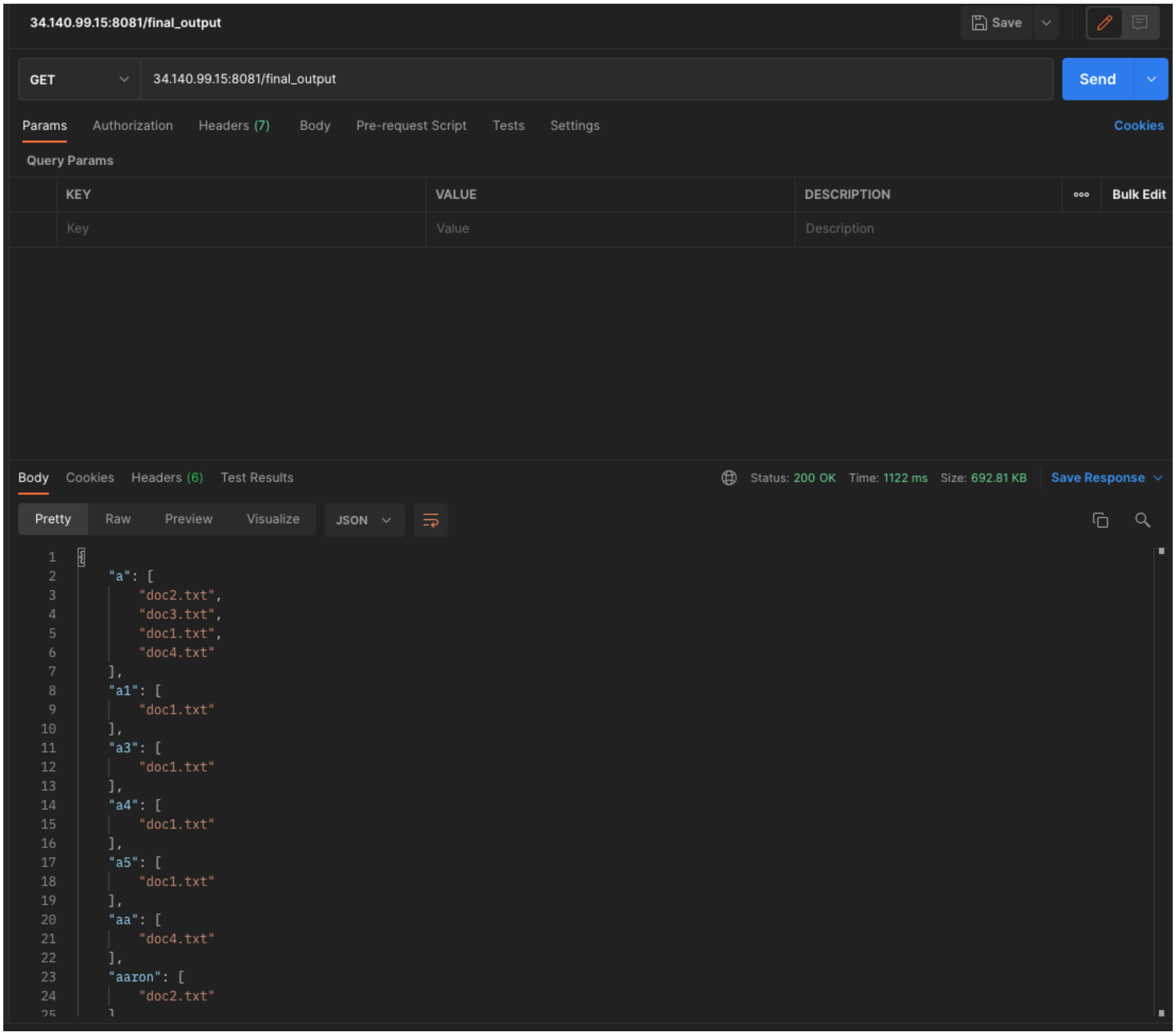
34.140.99.15:8081/launch_map_reduce

POST 34.140.99.15:8081/launch_map_reduce

Status: 200 OK Time: 12 m 9.24 s Size: 223 B

```
{
    "status": "complete"
}
```

Fetch Result



GET 34.140.99.15:8081/final_output

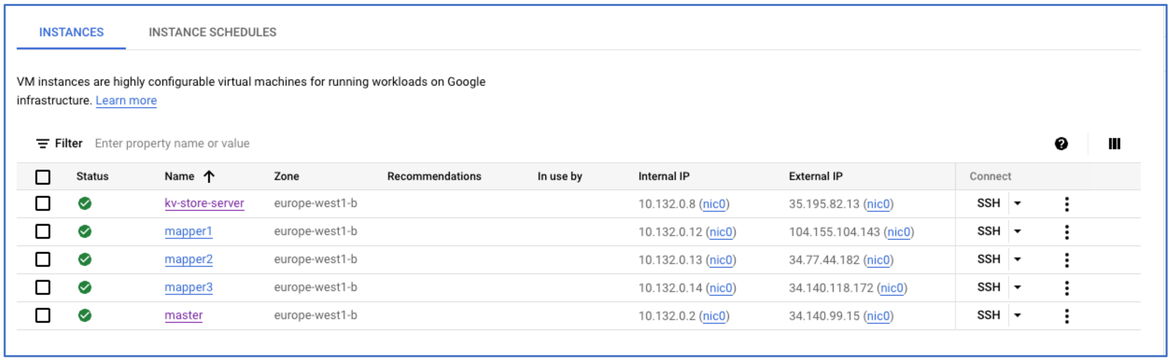
KEY	VALUE	DESCRIPTION
Key	Value	Description

Status: 200 OK Time: 1122 ms Size: 692.81 KB

```
{
    "a": [
        "doc2.txt",
        "doc3.txt",
        "doc1.txt",
        "doc4.txt"
    ],
    "a1": [
        "doc1.txt"
    ],
    "a3": [
        "doc1.txt"
    ],
    "a4": [
        "doc1.txt"
    ],
    "a5": [
        "doc1.txt"
    ],
    "aa": [
        "doc4.txt"
    ],
    "aaron": [
        "doc2.txt"
    ]
```

Implementation

KV Store, Mapper and Reducer VMs are created, set up and triggered by the master VM.



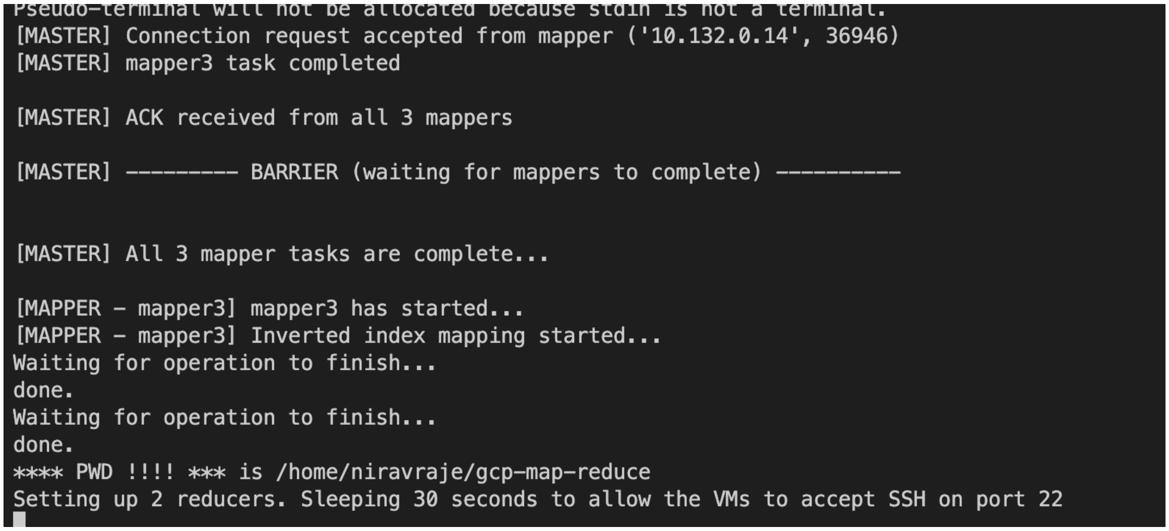
INSTANCES INSTANCE SCHEDULES

VM instances are highly configurable virtual machines for running workloads on Google infrastructure. Learn more

Filter Enter property name or value

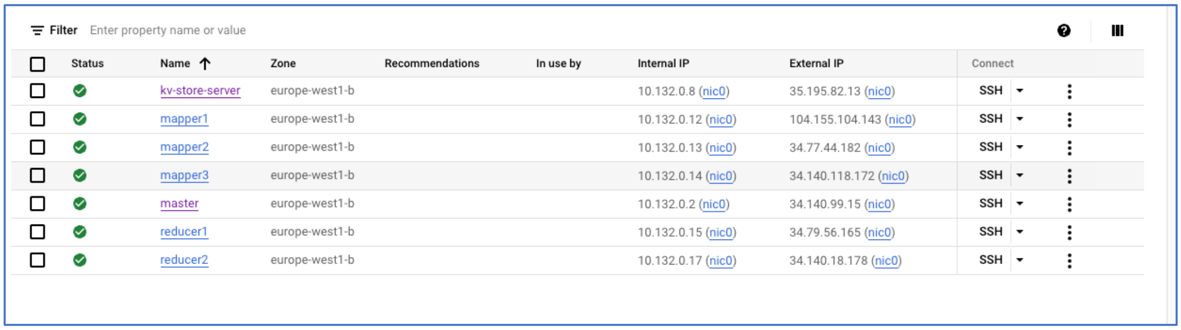
□	Status	Name ↑	Zone	Recommendations	In use by	Internal IP	External IP	Connect
□	✓	kv-store-server	europe-west1-b			10.132.0.8 (nic0)	35.195.82.13 (nic0)	SSH ▾ ⋮
□	✓	mapper1	europe-west1-b			10.132.0.12 (nic0)	104.155.104.143 (nic0)	SSH ▾ ⋮
□	✓	mapper2	europe-west1-b			10.132.0.13 (nic0)	34.77.44.182 (nic0)	SSH ▾ ⋮
□	✓	mapper3	europe-west1-b			10.132.0.14 (nic0)	34.140.118.172 (nic0)	SSH ▾ ⋮
□	✓	master	europe-west1-b			10.132.0.2 (nic0)	34.140.99.15 (nic0)	SSH ▾ ⋮

Barrier: Reducer VMs wait for Mapper VMs to complete their task.



```
Pseudo-terminal will not be allocated because stdin is not a terminal.
[MASTER] Connection request accepted from mapper ('10.132.0.14', 36946)
[MASTER] mapper3 task completed

[MASTER] ACK received from all 3 mappers

[MASTER] --------- BARRIER (waiting for mappers to complete) ----------


[MASTER] All 3 mapper tasks are complete...

[MAPPER - mapper3] mapper3 has started...
[MAPPER - mapper3] Inverted index mapping started...
Waiting for operation to finish...
done.
Waiting for operation to finish...
done.
**** PWD !!!! *** is /home/niravraje/gcp-map-reduce
Setting up 2 reducers. Sleeping 30 seconds to allow the VMs to accept SSH on port 22
```

Reducers are launched only after the mappers are complete.



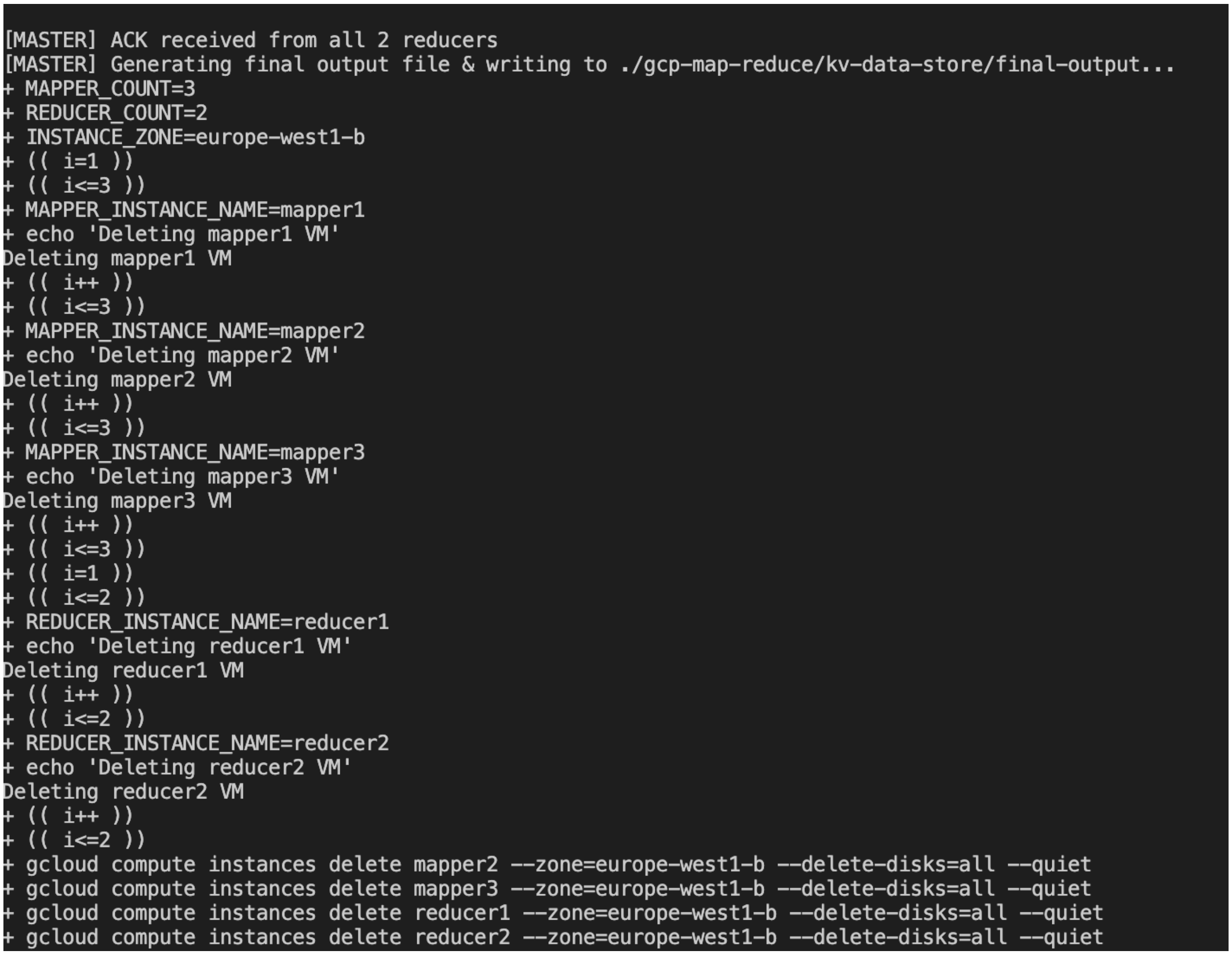
Filter Enter property name or value

□	Status	Name ↑	Zone	Recommendations	In use by	Internal IP	External IP	Connect
□	✓	kv-store-server	europe-west1-b			10.132.0.8 (nic0)	35.195.82.13 (nic0)	SSH ▾ ⋮
□	✓	mapper1	europe-west1-b			10.132.0.12 (nic0)	104.155.104.143 (nic0)	SSH ▾ ⋮
□	✓	mapper2	europe-west1-b			10.132.0.13 (nic0)	34.77.44.182 (nic0)	SSH ▾ ⋮
□	✓	mapper3	europe-west1-b			10.132.0.14 (nic0)	34.140.118.172 (nic0)	SSH ▾ ⋮
□	✓	master	europe-west1-b			10.132.0.2 (nic0)	34.140.99.15 (nic0)	SSH ▾ ⋮
□	✓	reducer1	europe-west1-b			10.132.0.15 (nic0)	34.79.56.165 (nic0)	SSH ▾ ⋮
□	✓	reducer2	europe-west1-b			10.132.0.17 (nic0)	34.140.18.178 (nic0)	SSH ▾ ⋮

Final output is stored in the KV store itself. The /final_output API is provided for external users to fetch the output.

The cleanup script is run by the master after process is complete to delete all mappers and reducers.

The KV Store is not deleted by the script in order to allow GET operations on final output.



```
[MASTER] ACK received from all 2 reducers
[MASTER] Generating final output file & writing to ./gcp-map-reduce/kv-data-store/final-output...
+ MAPPER_COUNT=3
+ REDUCER_COUNT=2
+ INSTANCE_ZONE=europe-west1-b
+ (( i=1 ))
+ (( i<=3 ))
+ MAPPER_INSTANCE_NAME=mapper1
+ echo 'Deleting mapper1 VM'
Deleting mapper1 VM
+ (( i++ ))
+ (( i<=3 ))
+ MAPPER_INSTANCE_NAME=mapper2
+ echo 'Deleting mapper2 VM'
Deleting mapper2 VM
+ (( i++ ))
+ (( i<=3 ))
+ MAPPER_INSTANCE_NAME=mapper3
+ echo 'Deleting mapper3 VM'
Deleting mapper3 VM
+ (( i++ ))
+ (( i<=3 ))
+ (( i=1 ))
+ (( i<=2 ))
+ REDUCER_INSTANCE_NAME=reducer1
+ echo 'Deleting reducer1 VM'
Deleting reducer1 VM
+ (( i++ ))
+ (( i<=2 ))
+ REDUCER_INSTANCE_NAME=reducer2
+ echo 'Deleting reducer2 VM'
Deleting reducer2 VM
+ (( i++ ))
+ (( i<=2 ))
+ gcloud compute instances delete mapper2 --zone=europe-west1-b --delete-disks=all --quiet
+ gcloud compute instances delete mapper3 --zone=europe-west1-b --delete-disks=all --quiet
+ gcloud compute instances delete reducer1 --zone=europe-west1-b --delete-disks=all --quiet
+ gcloud compute instances delete reducer2 --zone=europe-west1-b --delete-disks=all --quiet
```

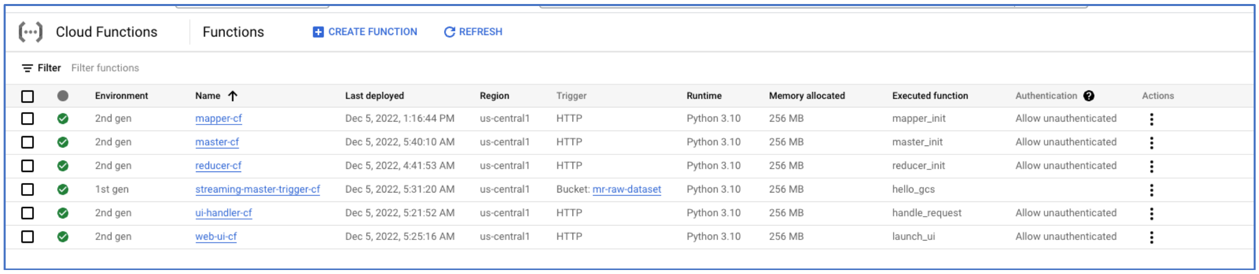
Part 2 – Map Reduce using FaaS (Google Cloud Functions)

Overview

In this part, the entire map reduce workflow is processed using FaaS.

Below are the cloud functions created:

- Master
- Mapper
- Reducer
- Streaming master trigger – Used for monitoring any uploads to the “raw-dataset” bucket. Any new file upload triggers the master process.
- Web UI – A cloud function serving the inverted index search UI
- UI Handler – Accepts the word entered by the user on the Web UI, searches for the word in the final output and returns the result to display on the UI.

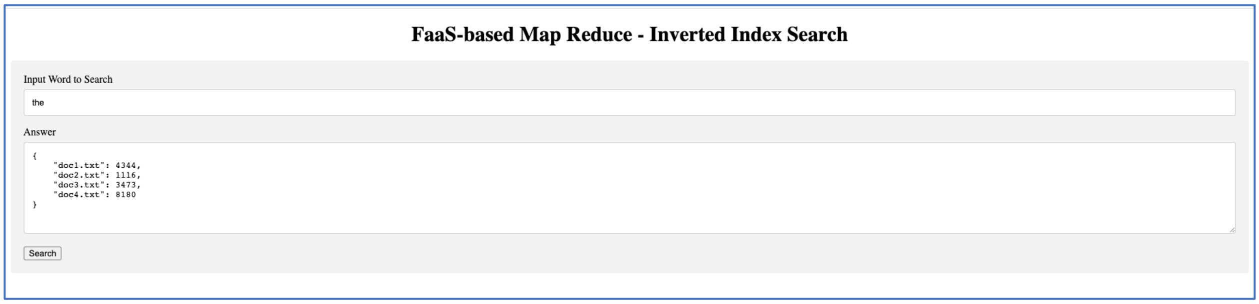


□	●	Environment	Name ↑	Last deployed	Region	Trigger	Runtime	Memory allocated	Executed function	Authentication	Actions
□	✓	2nd gen	mapper-cf	Dec 5, 2022, 1:16:44 PM	us-central1	HTTP	Python 3.10	256 MB	mapper_init	Allow unauthenticated	⋮
□	✓	2nd gen	master-cf	Dec 5, 2022, 5:40:10 AM	us-central1	HTTP	Python 3.10	256 MB	master_init	Allow unauthenticated	⋮
□	✓	2nd gen	reducer-cf	Dec 5, 2022, 4:41:53 AM	us-central1	HTTP	Python 3.10	256 MB	reducer_init	Allow unauthenticated	⋮
□	✓	1st gen	streaming-master-trigger-cf	Dec 5, 2022, 5:31:20 AM	us-central1	Bucket: mr-raw-dataset	Python 3.10	256 MB	hello_gcs		⋮
□	✓	2nd gen	ui-handler-cf	Dec 5, 2022, 5:21:52 AM	us-central1	HTTP	Python 3.10	256 MB	handle_request	Allow unauthenticated	⋮
□	✓	2nd gen	web-ui-cf	Dec 5, 2022, 5:25:16 AM	us-central1	HTTP	Python 3.10	256 MB	launch_ui	Allow unauthenticated	⋮

Implementation

Web UI

The UI can be accessed at: https://web-ui-cf-gf2co4zquq-uc.a.run.app



FaaS-based Map Reduce - Inverted Index Search

Input Word to Search

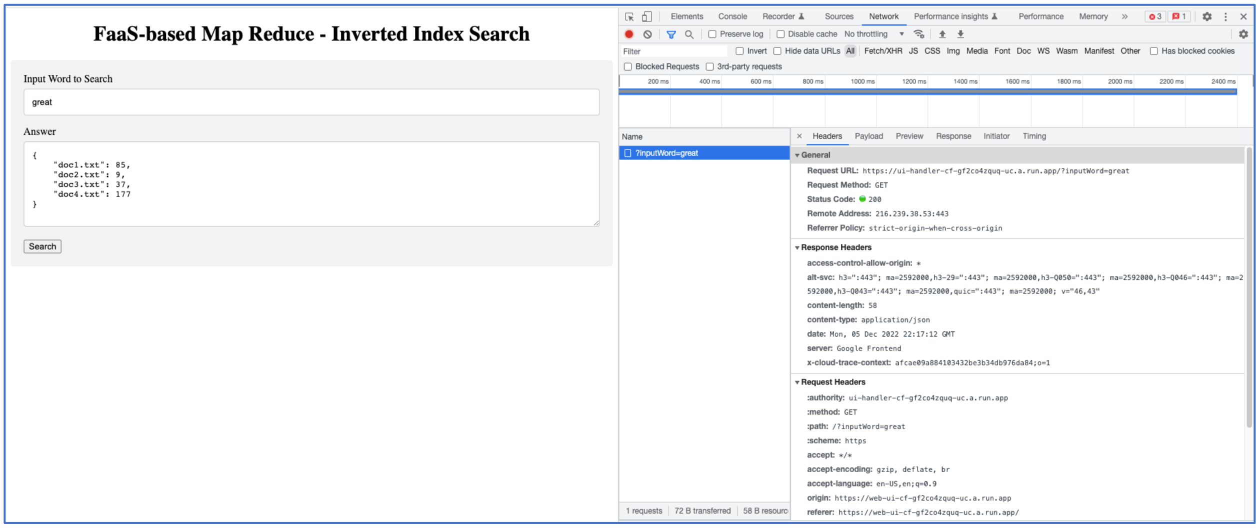
the

Answer

```
{
    "doc1.txt": 4344,
    "doc2.txt": 1116,
    "doc3.txt": 3473,
    "doc4.txt": 8180
}
```

Currently there are 4 documents in the raw-dataset bucket.

To search another word, please refresh the page. This is a basic UI to test implementation and can be improved further to indicate “Word not found” or show a progress indicator.



FaaS-based Map Reduce - Inverted Index Search

Input Word to Search: great

Answer

```
{
    "doc1.txt": 85,
    "doc2.txt": 9,
    "doc3.txt": 37,
    "doc4.txt": 177
}
```

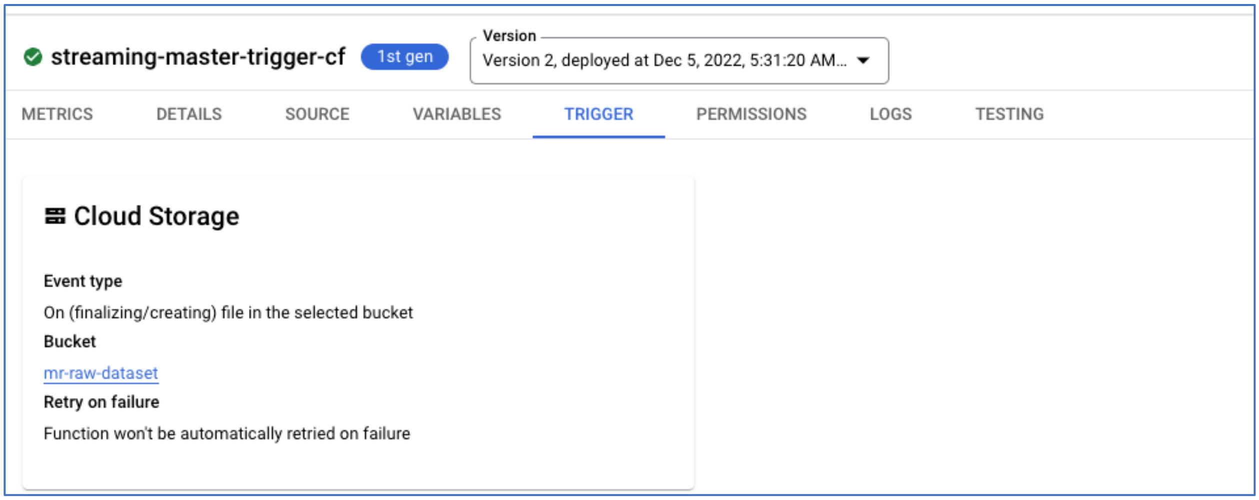
?inputWord=great

```
▾ General
  Request URL: https://ui-handler-cf-gf2co4zquq-uc.a.run.app/?inputWord=great
  Request Method: GET
  Status Code: ● 200
  Remote Address: 216.239.38.53:443
  Referrer Policy: strict-origin-when-cross-origin
▾ Response Headers
  access-control-allow-origin: *
  alt-svc: h3=":443"; ma=2592000,h3-29=":443"; ma=2592000,h3-Q050=":443"; ma=2592000,h3-Q046=":443"; ma=2
  592000,h3-Q043=":443"; ma=2592000,quic=":443"; ma=2592000; v="46,43"
  content-length: 58
  content-type: application/json
  date: Mon, 05 Dec 2022 22:17:12 GMT
  server: Google Frontend
  x-cloud-trace-context: afcae09a884103432be3b34db976da84;o=1
▾ Request Headers
  :authority: ui-handler-cf-gf2co4zquq-uc.a.run.app
  :method: GET
  :path: /?inputWord=great
  :scheme: https
  accept: */*
  accept-encoding: gzip, deflate, br
  accept-language: en-US,en;q=0.9
  origin: https://web-ui-cf-gf2co4zquq-uc.a.run.app
  referer: https://web-ui-cf-gf2co4zquq-uc.a.run.app/
```

1 requests | 72 B transferred | 58 B resourc

Streaming Search

On creating a new file or uploading one for the “mr-raw-dataset” bucket, the below CF will trigger master as shown below.



✓ streaming-master-trigger-cf 1st gen

Version: Version 2, deployed at Dec 5, 2022, 5:31:20 AM...

Cloud Storage

Event type

On (finalizing/creating) file in the selected bucket

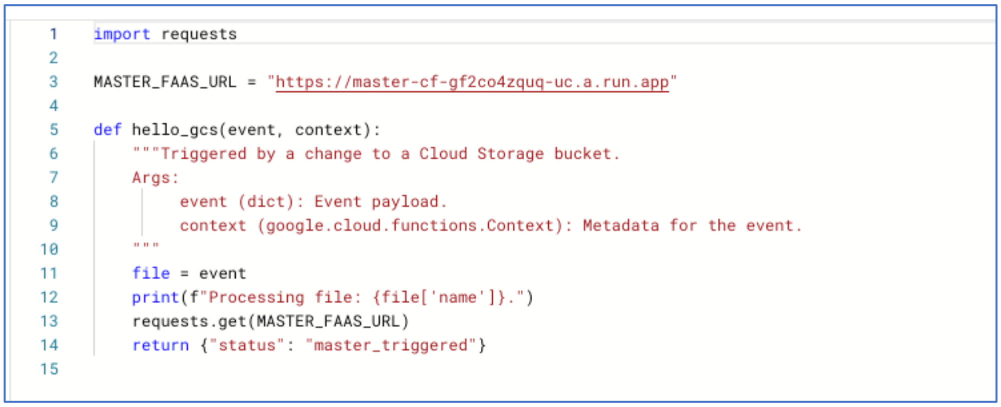
Bucket

mr-raw-dataset

Retry on failure

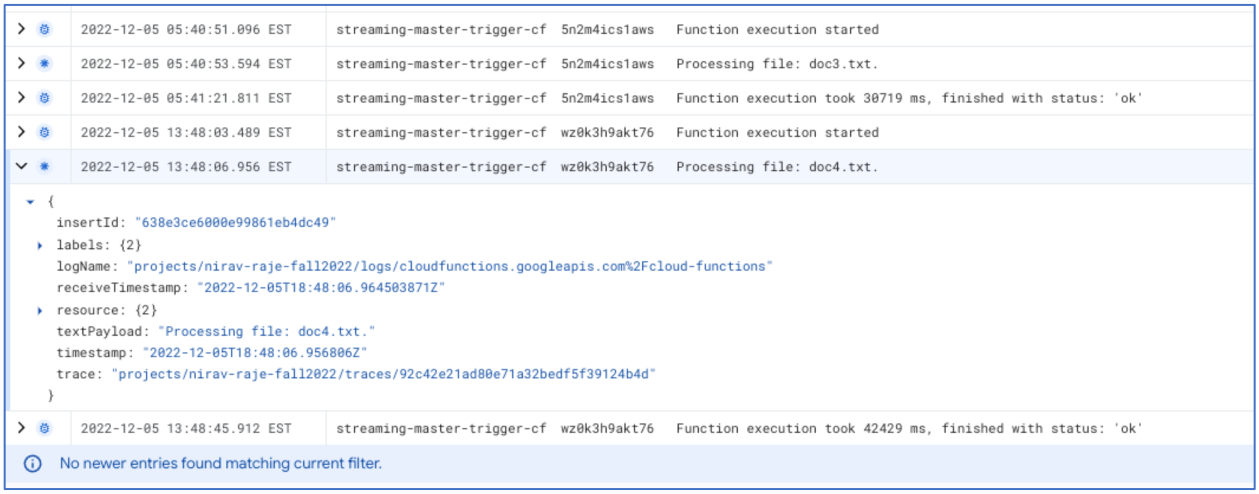
Function won't be automatically retried on failure

Source code



```python
import requests

MASTER_FAAS_URL = "https://master-cf-gf2co4zquq-uc.a.run.app"

def hello_gcs(event, context):
    """Triggered by a change to a Cloud Storage bucket.
    Args:
        event (dict): Event payload.
        context (google.cloud.functions.Context): Metadata for the event.
    """
    file = event
    print(f"Processing file: {file['name']}.")
    requests.get(MASTER_FAAS_URL)
    return {"status": "master_triggered"}

```

As we can see below, when “doc3.txt” and “doc4.txt” were uploaded, the master CF was triggered.

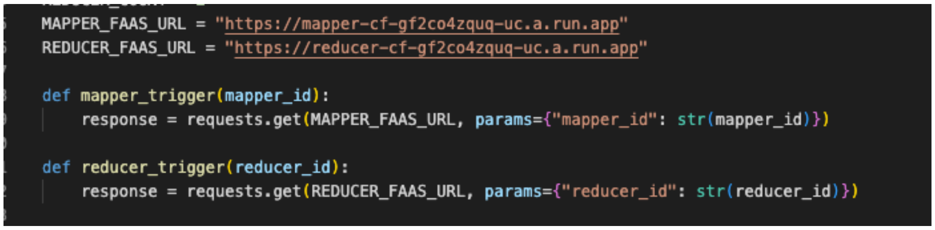


```
2022-12-05 05:40:51.096 EST   streaming-master-trigger-cf  5n2m4ics1aws   Function execution started
2022-12-05 05:40:53.594 EST   streaming-master-trigger-cf  5n2m4ics1aws   Processing file: doc3.txt.
2022-12-05 05:41:21.811 EST   streaming-master-trigger-cf  5n2m4ics1aws   Function execution took 30719 ms, finished with status: 'ok'
2022-12-05 13:48:03.489 EST   streaming-master-trigger-cf  wz0k3h9akt76   Function execution started
2022-12-05 13:48:06.956 EST   streaming-master-trigger-cf  wz0k3h9akt76   Processing file: doc4.txt.
 {
   insertId: "638e3ce6000e99861eb4dc49"
  labels: {2}
   logName: "projects/nirav-raje-fall2022/logs/cloudfunctions.googleapis.com%2Fcloud-functions"
   receiveTimestamp: "2022-12-05T18:48:06.964503871Z"
  resource: {2}
   textPayload: "Processing file: doc4.txt."
   timestamp: "2022-12-05T18:48:06.956806Z"
   trace: "projects/nirav-raje-fall2022/traces/92c42e21ad80e71a32bedf5f39124b4d"
 }
2022-12-05 13:48:45.912 EST   streaming-master-trigger-cf  wz0k3h9akt76   Function execution took 42429 ms, finished with status: 'ok'
```

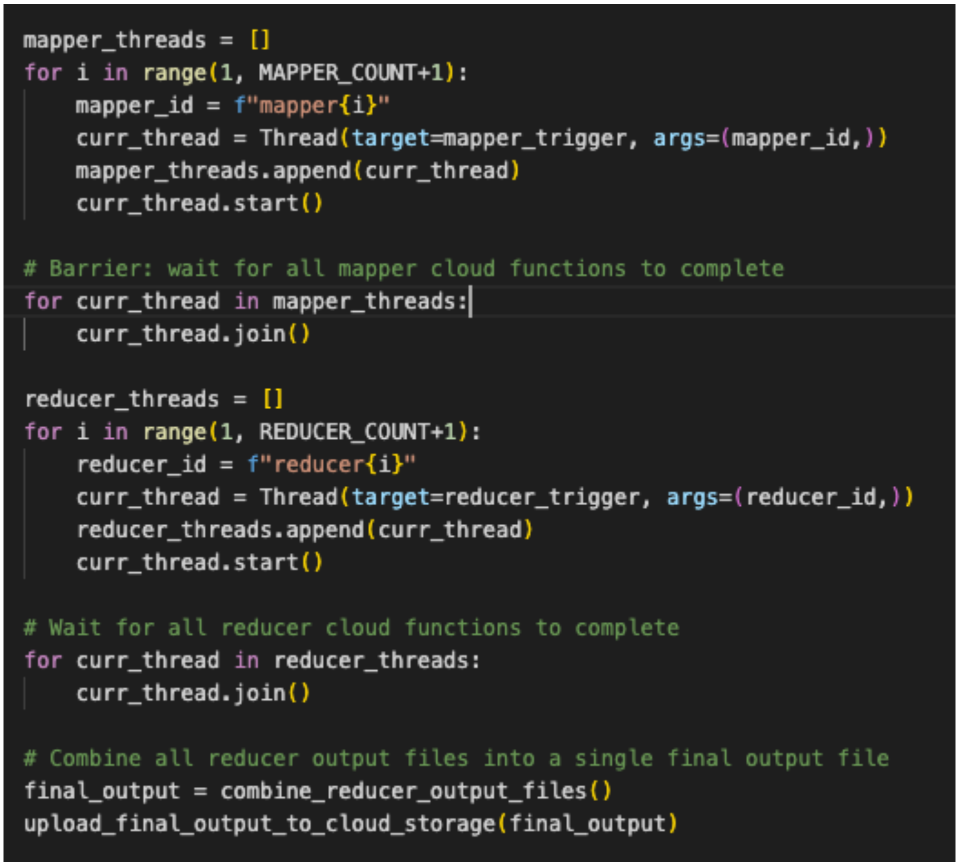
No newer entries found matching current filter.

Master Function – Orchestration & Barrier Implementation

The mapper and reducer cloud functions are simply called in parallel from the master. Parallel invocation has been achieved using multithreading in Python. Barrier has been implemented using thread.join() in a loop.

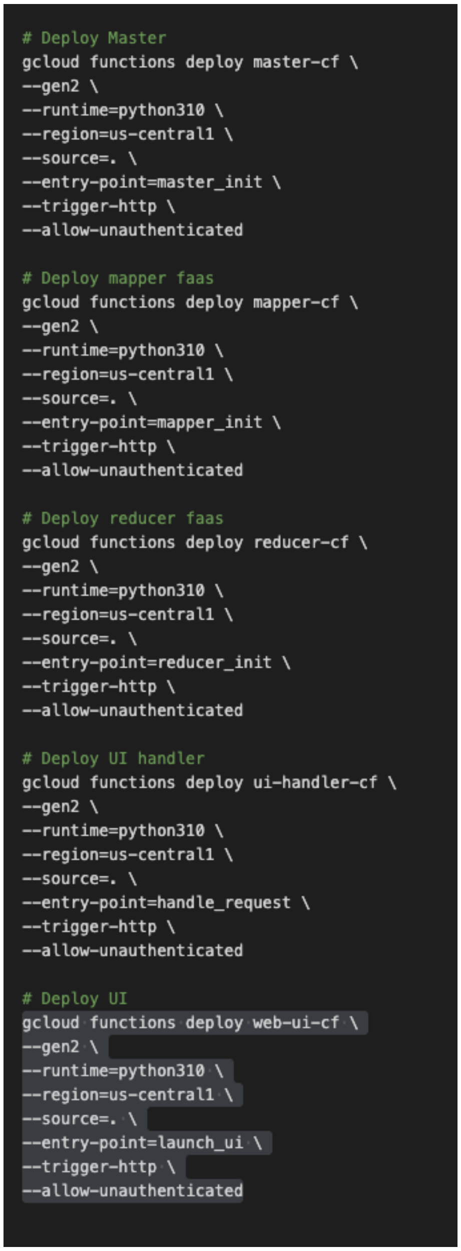


```python
MAPPER_FAAS_URL = "https://mapper-cf-gf2co4zquq-uc.a.run.app"
REDUCER_FAAS_URL = "https://reducer-cf-gf2co4zquq-uc.a.run.app"

def mapper_trigger(mapper_id):
    response = requests.get(MAPPER_FAAS_URL, params={"mapper_id": str(mapper_id)})

def reducer_trigger(reducer_id):
    response = requests.get(REDUCER_FAAS_URL, params={"reducer_id": str(reducer_id)})
```



```
mapper_threads = []
for i in range(1, MAPPER_COUNT+1):
    mapper_id = f"mapper{i}"
    curr_thread = Thread(target=mapper_trigger, args=(mapper_id,))
    mapper_threads.append(curr_thread)
    curr_thread.start()

# Barrier: wait for all mapper cloud functions to complete
for curr_thread in mapper_threads:
    curr_thread.join()

reducer_threads = []
for i in range(1, REDUCER_COUNT+1):
    reducer_id = f"reducer{i}"
    curr_thread = Thread(target=reducer_trigger, args=(reducer_id,))
    reducer_threads.append(curr_thread)
    curr_thread.start()

# Wait for all reducer cloud functions to complete
for curr_thread in reducer_threads:
    curr_thread.join()

# Combine all reducer output files into a single final output file
final_output = combine_reducer_output_files()
upload_final_output_to_cloud_storage(final_output)
```

The below script snippets can be used to create the cloud functions from the gcloud CLI instead of Console. Note: Each snippet needs to be run from the appropriate folder (please see attached codebase for reference)



```
# Deploy Master
gcloud functions deploy master-cf \
--gen2 \
--runtime=python310 \
--region=us-central1 \
--source=. \
--entry-point=master_init \
--trigger-http \
--allow-unauthenticated

# Deploy mapper faas
gcloud functions deploy mapper-cf \
--gen2 \
--runtime=python310 \
--region=us-central1 \
--source=. \
--entry-point=mapper_init \
--trigger-http \
--allow-unauthenticated

# Deploy reducer faas
gcloud functions deploy reducer-cf \
--gen2 \
--runtime=python310 \
--region=us-central1 \
--source=. \
--entry-point=reducer_init \
--trigger-http \
--allow-unauthenticated

# Deploy UI handler
gcloud functions deploy ui-handler-cf \
--gen2 \
--runtime=python310 \
--region=us-central1 \
--source=. \
--entry-point=handle_request \
--trigger-http \
--allow-unauthenticated

# Deploy UI
gcloud functions deploy web-ui-cf \
--gen2 \
--runtime=python310 \
--region=us-central1 \
--source=. \
--entry-point=launch_ui \
--trigger-http \
--allow-unauthenticated
```