Github Setup

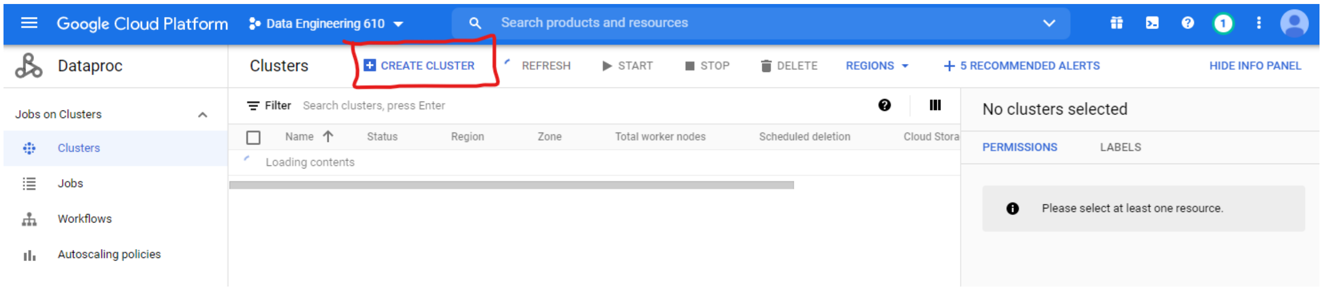
1) Create a github

2) Fork https://github.com/Regis-University-Data-Science/simple_Hadoop_MapReduce_example

Cluster Setup

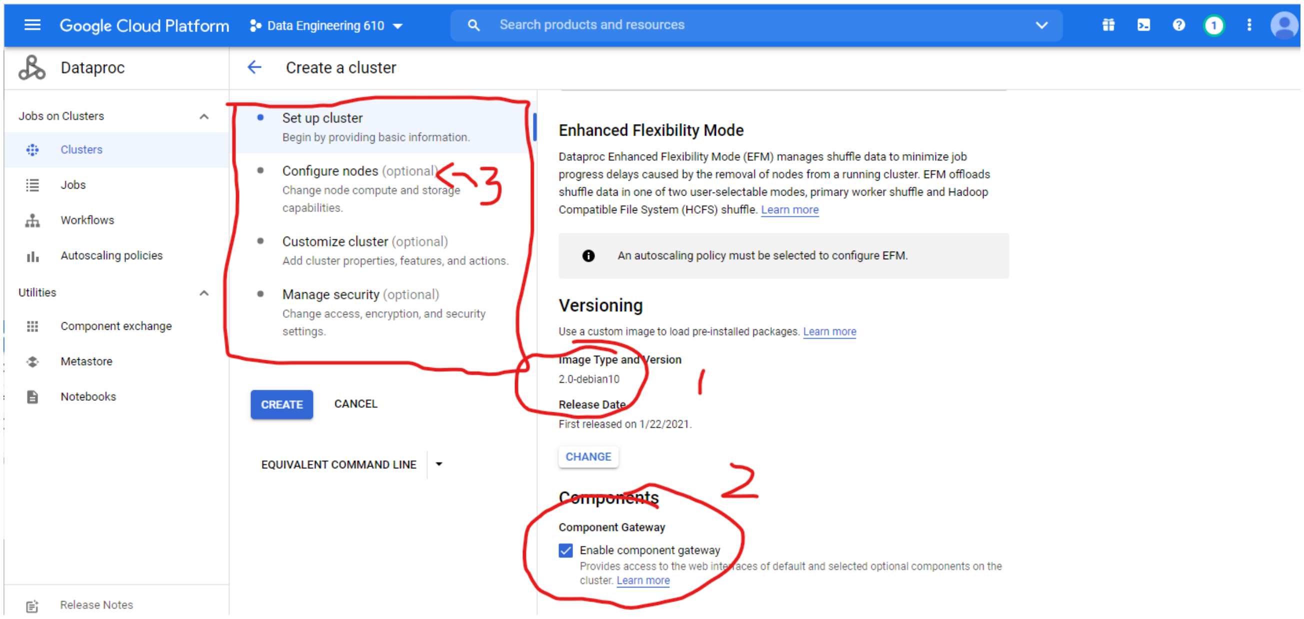
1) Navigate to Dataproc in the GCP environment. You can search for it in the bar, or manually look through the menu.

2) Create a New Cluster



Please select at least one resource.

3) Enable Component Gateway, choose an operating system (debian 2.0 works with the map reduce commands), and navigate to the configure nodes section.


Google Cloud Platform
Data Engineering 610
Search products and resources
Dataproc
Create a cluster
Jobs on Clusters
Clusters
Jobs
Workflows
Autoscaling policies
Utilities
Component exchange
Metastore
Notebooks
Release Notes
Set up cluster
Begin by providing basic information.
Configure nodes (optional)
Change node compute and storage capabilities.
Customize cluster (optional)
Add cluster properties, features, and actions.
Manage security (optional)
Change access, encryption, and security settings.
CREATE
CANCEL
EQUIVALENT COMMAND LINE
3
1
2

Enhanced Flexibility Mode

Dataproc Enhanced Flexibility Mode (EFM) manages shuffle data to minimize job progress delays caused by the removal of nodes from a running cluster. EFM offloads shuffle data in one of two user-selectable modes, primary worker shuffle and Hadoop Compatible File System (HCFS) shuffle. Learn more

An autoscaling policy must be selected to configure EFM.

Versioning

Use a custom image to load pre-installed packages. Learn more

Image Type and Version: 2.0-debian10

Release Date: First released on 1/22/2021.

CHANGE

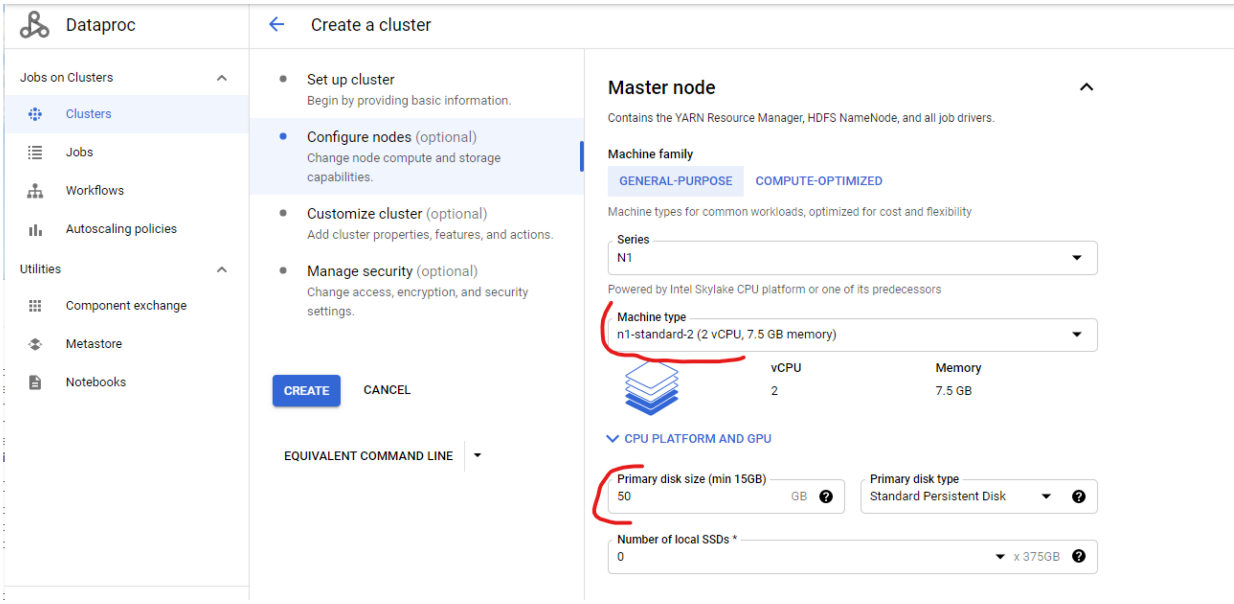
Components

Component Gateway

☑ Enable component gateway

Provides access to the web interfaces of default and selected optional components on the cluster. Learn more

4) Configure Node Sizes



- Set up cluster
  Begin by providing basic information.
- Configure nodes (optional)
  Change node compute and storage capabilities.
- Customize cluster (optional)
  Add cluster properties, features, and actions.
- Manage security (optional)
  Change access, encryption, and security settings.

Master node

Contains the YARN Resource Manager, HDFS NameNode, and all job drivers.

Machine family

Machine types for common workloads, optimized for cost and flexibility

Series: N1

Powered by Intel Skylake CPU platform or one of its predecessors

Machine type: n1-standard-2 (2 vCPU, 7.5 GB memory)

vCPU	Memory
2	7.5 GB

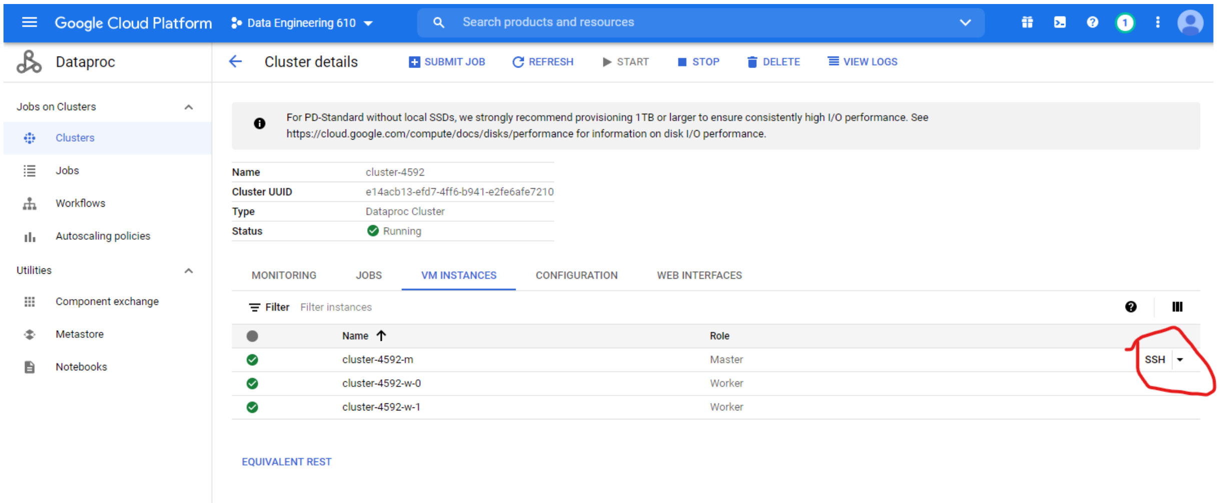
Primary disk size (min 15GB): 50 GB

Primary disk type: Standard Persistent Disk

Number of local SSDs *: 0 x 375GB

5) Create Cluster (It’s the blue button on the above screen shot)

6) Wait for cluster to start, and then ssh into the main node.



For PD-Standard without local SSDs, we strongly recommend provisioning 1TB or larger to ensure consistently high I/O performance. See https://cloud.google.com/compute/docs/disks/performance for information on disk I/O performance.

Name	cluster-4592
Cluster UUID	e14acb13-efd7-4ff6-b941-e2fe6afe7210
Type	Dataproc Cluster
Status	✓ Running

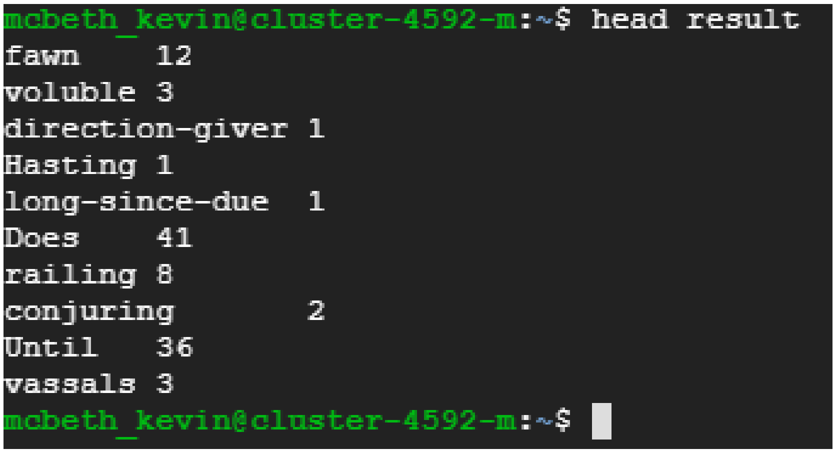
●	Name ↑	Role	
✓	cluster-4592-m	Master	SSH ▾
✓	cluster-4592-w-0	Worker	
✓	cluster-4592-w-1	Worker	

Run Map Reduce

1) Run the following commands, make sure that the $$ sections are changed for you:

```
git clone https://github.com/$GITHUB_PROFILE_NAME_HERE$/simple_Hadoop_MapReduce_example

wget http://norvig.com/ngrams/shakespeare.txt

hdfs dfs -mkdir /shakespeare

hdfs dfs -mkdir /shakespeare/input

hdfs dfs -copyFromLocal shakespeare.txt /shakespeare/input

hdfs dfs -ls /shakespeare/input

cd simple_Hadoop_MapReduce_example
```

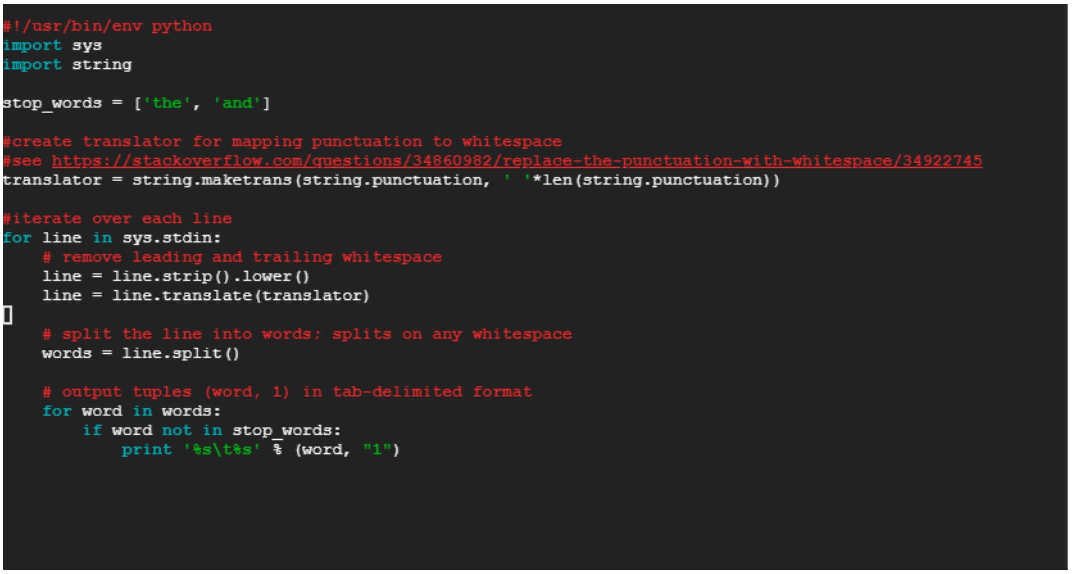
```
mapred streaming -file mapper.py -mapper mapper.py -file reducer.py -reducer reducer.py -input /shakespeare/input -output /shakespeare/output

hdfs dfs -ls /shakespeare/output

hdfs dfs -getmerge /shakespeare/output/ /home/$CLUSTER_USER_NAME_HERE$/result

head /home/$CLUSTER_USER_NAME_HERE$/ result
```



```
mcbeth_kevin@cluster-4592-m:~$ head result
fawn    12
voluble 3
direction-giver 1
Hasting 1
long-since-due  1
Does    41
railing 8
conjuring       2
Until   36
vassals 3
mcbeth_kevin@cluster-4592-m:~$
```

Adjust Mapper File

1) Do something to augment the mapper.py file. I chose to do stop words and get rid of the punctuation. I chose to replace punctuation with white space to handle strings like “run,on,sentence” => “run on sentence”.



```
#!/usr/bin/env python
import sys
import string

stop_words = ['the', 'and']

#create translator for mapping punctuation to whitespace
#see https://stackoverflow.com/questions/34860982/replace-the-punctuation-with-whitespace/34922745
translator = string.maketrans(string.punctuation, ' '*len(string.punctuation))

#iterate over each line
for line in sys.stdin:
    # remove leading and trailing whitespace
    line = line.strip().lower()
    line = line.translate(translator)

    # split the line into words; splits on any whitespace
    words = line.split()

    # output tuples (word, 1) in tab-delimited format
    for word in words:
        if word not in stop_words:
            print '%s\t%s' % (word, "1")
```

2) rerun the map / reduce program

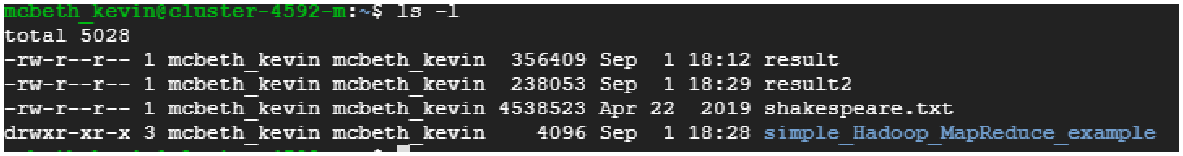
mapred streaming -file mapper.py -mapper mapper.py -file reducer.py -reducer reducer.py -input /shakespeare/input -output /shakespeare/output2

hdfs dfs -ls /shakespeare/output2

hdfs dfs -getmerge /shakespeare/output2/ /home/mcbeth_kevin/result2

head /home/mcbeth_kevin/result2

3) check output to make sure it’s different. You can see that my changes dropped 120k bytes by removing stop words, punctuation.



```
mcbeth_kevin@cluster-4592-m:~$ ls -l
total 5028
-rw-r--r-- 1 mcbeth_kevin mcbeth_kevin  356409 Sep  1 18:12 result
-rw-r--r-- 1 mcbeth_kevin mcbeth_kevin  238053 Sep  1 18:29 result2
-rw-r--r-- 1 mcbeth_kevin mcbeth_kevin 4538523 Apr 22  2019 shakespeare.txt
drwxr-xr-x 3 mcbeth_kevin mcbeth_kevin    4096 Sep  1 18:28 simple_Hadoop_MapReduce_example
```

Push to GitHub

1) Follow RTM guidance from week 2 assignment and rtm 2.

As of 2 weeks ago you can no longer use password access. To get around this, you have to generate a token and maybe set up 2 factor authentication (I did them out of order so I'm not sure if the latter is necessary). Follow the guides below.

https://docs.github.com/en/github/authenticating-to-github/keeping-your-account-and-data-secure/creating-a-personal-access-token

In place of your password you will use this token.

https://docs.github.com/en/github/authenticating-to-github/securing-your-account-with-two-factor-authentication-2fa/configuring-two-factor-authentication

Minimum Deliverables

1) Running cluster (GCP home page cluster running screenshot)

2) Running Commands (e.g. command success, or files in your hdfs system)

3) github upload proof, either your link to your github project or the git push success screenshot.