

## **I535 Spring 2023 - MGMT - Distributed Computing and File System**

Submit a short report and the completion screenshots that compare two labs and describe your experience with them. The report should include the following:

- A short description of how you understand Dataproc capabilities (feel free to consult GCP documentation). 1 Pt.
- A summary of what you learned in each lab. 1 Pt.
- A brief discussion of how changing the number of worker nodes would affect the performance in your cluster. 1 Pt.

### **Dataproc Capabilities**

Google Cloud Dataproc is a powerful big data processing service that enables users to run popular frameworks such as Apache Hadoop and Apache Spark on the cloud. With Dataproc, users can easily create and manage clusters, scale resources based on demand, and integrate with other Google Cloud services. One of the key benefits of Dataproc is its ease of use. Users can create clusters with just a few clicks, and Dataproc takes care of the underlying infrastructure and configuration. This makes it easy for organizations to focus on their data processing tasks rather than the underlying infrastructure.

Additionally, Dataproc offers flexibility and customization options, allowing users to specify the versions of frameworks they want to use and automate setup tasks. The service also supports data encryption and integrates with Cloud IAM for access control. Overall, Google Cloud Dataproc is a robust and versatile big data processing service that offers many benefits to organizations looking to process and analyze large amounts of data on the cloud.

# Dataproc: Qwik Start - Console

## Learning of Lab 1:

In the Google Cloud Dataproc Qwik Start - Console lab, users learn the basics of creating and managing Dataproc clusters using the Google Cloud Console. This lab provides a hands-on introduction to the key features of Dataproc, such as cluster creation, job submission, and monitoring. Users learn how to create a cluster using the Cloud Console, configure cluster settings such as cluster size and machine type, and submit a job to the cluster using the Cloud Shell command line. The lab also covers how to monitor job progress and view job logs using the Dataproc web interface.

### Task 1: Create Cluster

The screenshot shows the Google Cloud Console interface for the Dataproc service. On the left, there's a sidebar with options like Clusters, Jobs, Workflows, and Autoscaling policies. The main area is titled 'Clusters' and shows a table with one row. The row for 'example-cluster' has a checkbox next to it. The table columns include Name, Status, Region, Zone, Total worker nodes, Scheduled deletion, and Cloud provider. The status is 'Running'. The region is 'us-west4' and the zone is 'us-west4-c'. The total worker nodes are 2. The scheduled deletion is 'Off'. The Cloud provider is 'dataproc' and the ID is '14752'. A message at the bottom right of the table area says 'Please select at least one resource.'

### Task 2:

The screenshot shows the Google Cloud Console interface for the Dataproc service, specifically for a job named 'job-9b182c9e'. The job status is 'Succeeded'. The 'MONITORING' tab is selected, showing a chart of YARN memory usage over time. Below the chart, a note says 'The charts below represent the metrics from the cluster this job ran on, scoped to the time that this job was running. It is possible for more than one job to run on a cluster at a time, so these metrics may not reflect this job's resource usage accurately. Metrics for a job may lag behind the job run by several minutes.' There are buttons for 'SAVE AS DASHBOARD' and 'RESET ZOOM', and a time range selector from '1 hour' to '30 days' with a current selection of '11:45AM - 11:50AM'. The 'CONFIGURATION' tab is also visible. At the bottom, the 'Output' section shows log entries in a text area with a 'LINE WRAP: OFF' option. One entry reads: '23/03/03 16:49:45 INFO org.sparkproject.jetty.server.AbstractConnector: Started ServerConnector@2ec5fb26{HTTP/1.1}{0.0.0.0:46715}'.

### Task 3: Task 3. View the job output

The screenshot shows the Google Cloud Dataproc interface. On the left, there's a sidebar with sections like Jobs on Clusters, Metastore Services, Utilities, and Workbench. The main area is titled 'Job details' and shows the 'MONITORING' tab selected. Below it, the 'Output' tab is active, displaying a log of Spark tasks. A message at the top of the log says: 'Spark jobs take ~60 seconds to initialize resources.' The log itself contains several lines of Java code, likely Spark logs, indicating the initialization of various components like MapOutputTracker, BlockManagerMaster, and OutputCommitCoordinator. A modal window at the bottom asks 'Request to update cluster example-cluster submitted'.

```
23/03/03 16:49:44 INFO org.apache.spark.SparkEnv: Registering MapOutputTracker
23/03/03 16:49:45 INFO org.apache.spark.SparkEnv: Registering BlockManagerMaster
23/03/03 16:49:45 INFO org.apache.spark.SparkEnv: Registering BlockManagerMasterHeartbeat
23/03/03 16:49:45 INFO org.apache.spark.SparkEnv: Registering OutputCommitCoordinator
23/03/03 16:49:45 INFO org.sparkproject.jetty.util.log.Logging initialized @563ms to org.sparkproject.jetty.util.log.Slf4jLog
23/03/03 16:49:45 INFO org.sparkproject.jetty.server.Server: jetty-9.4.40.v20210413; built: 2021-04-13T20:42:42.668Z; git: b881a572662e1943a14ae12e7e1287989f218b74; jvm 1.8.0_362-b09
23/03/03 16:49:45 INFO org.sparkproject.jetty.server.Server: Started @5819ms
23/03/03 16:49:45 INFO org.sparkproject.jetty.server.AbstractConnector: Started ServerConnector@2ec5fb26{HTTP/1.1, (http/1.1)}{0.0.0.0:46715}
23/03/03 16:49:47 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceManager at example-cluster-m/10.182.0.3:8032
23/03/03 16:49:47 INFO org.apache.hadoop.yarn.client.AHSProxy: Connecting to Application History server at example-cluster-m/10.182.0.3:10200
23/03/03 16:49:48 INFO org.apache.hadoop.conf.Configuration: resource-types.xml not found
23/03/03 16:49:48 INFO org.apache.hadoop.util.resource.ResourceUtils: Unable to find 'resource-types.xml'.
23/03/03 16:49:50 INFO org.apache.hadoop.yarn.client.api.impl.YarnClientImpl: Submitted application application_1677862038877_0001
23/03/03 16:49:51 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceManager at example-cluster-m/10.182.0.3:8030
23/03/03 16:49:54 INFO com.google.cloud.hadoop.repackaged.gcs.com.google.cloud.hadoop.Gcsio.GoogleCloudStorageImpl: Ignoring exception of type GoogleJsonResponseException; verified object already exists with desired state.
Pi is roughly 3.1403757918785902
23/03/03 16:50:04 INFO org.sparkproject.jetty.server.AbstractConnector: Stopped Spark@2ec5fb26{HTTP/1.1, (http/1.1)}{0.0.0.0:0}
```

### Task 4: Update Clusters

The screenshot shows the Google Cloud Dataproc interface. The left sidebar has sections like Jobs on Clusters, Metastore Services, Utilities, and Workbench. The main area is titled 'Cluster details' and shows the configuration for a cluster named 'example-cluster'. A modal window titled 'Editing cluster' is open, allowing changes to worker node counts. The 'Worker nodes' field is set to 4, and the 'Secondary worker nodes' field is set to 0. To the right of the modal, there's a 'Labels' section with three key-value pairs: Key 1 (goog-dataproc-cluster-name) is example-cluster, Key 2 (goog-dataproc-cluster-uuid) is a50d4c8a-c107-49c5-bcdf-1abf, and Key 3 (goog-dataproc-location) is us-west4. At the bottom of the modal, there are 'SAVE', 'CANCEL', and 'EQUIVALENT REST' buttons.

Label	Value
Key 1	goog-dataproc-cluster-name
Value 1	example-cluster
Key 2	goog-dataproc-cluster-uuid
Value 2	a50d4c8a-c107-49c5-bcdf-1abf
Key 3	goog-dataproc-location
Value 3	us-west4

## Task 5: Completion

The screenshot shows a browser window with a tab titled "example-cluster - Configuration". The main content area is titled "Task 5. Test your understanding". It contains two questions:

- Which type of Dataproc job is submitted in the lab?**  
The correct answer, "Spark", is selected with a green checkmark. Other options include Hadoop, SparkSQL, PySpark, and Pig. A "Submit" button is visible.
- Dataproc helps users process, transform and understand vast quantities of data.**  
The correct answer, "True", is selected with a green checkmark. A "Submit" button is visible.

On the right side, there is a sidebar with the following information:

- GSP103
- Overview (highlighted)
- Setup and requirements
- Task 1. Create a cluster
- Task 2. Submit a job
- Task 3. View the job output
- Task 4. Update a cluster
- Task 5. Test your understanding** (highlighted)
- Congratulations!

A progress bar at the top right indicates "100/100".

### Q. A brief discussion of how changing the number of worker nodes would affect the performance in your cluster.

The number of worker nodes in a Dataproc cluster can significantly affect cluster performance. By dispersing the burden across more machines, increasing the number of worker nodes can enhance performance and allow for faster data processing. Adding extra nodes, however, raises the cost of running the cluster.

Reducing the number of worker nodes, on the other hand, can reduce performance by limiting the cluster's available processing capacity. With large datasets, this can result in slower job completion times and longer processing durations.

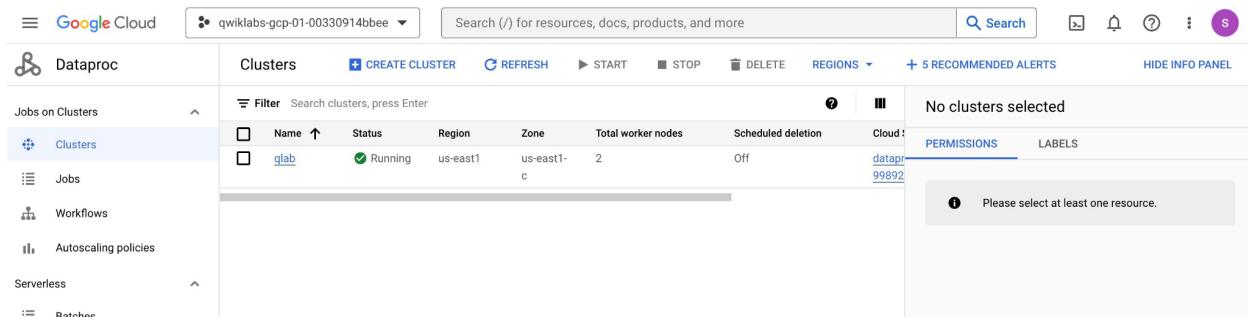
When altering the number of worker nodes in a Dataproc cluster, it is critical to carefully assess the workload and processing requirements. For example, adding more worker nodes may yield considerable performance increases for workloads with high parallelism, whereas adding more worker nodes may not enhance performance as much for tasks with low parallelism.

# Introduction to Cloud Dataproc: Hadoop and Spark on Google Cloud

## Learning of Lab 2:

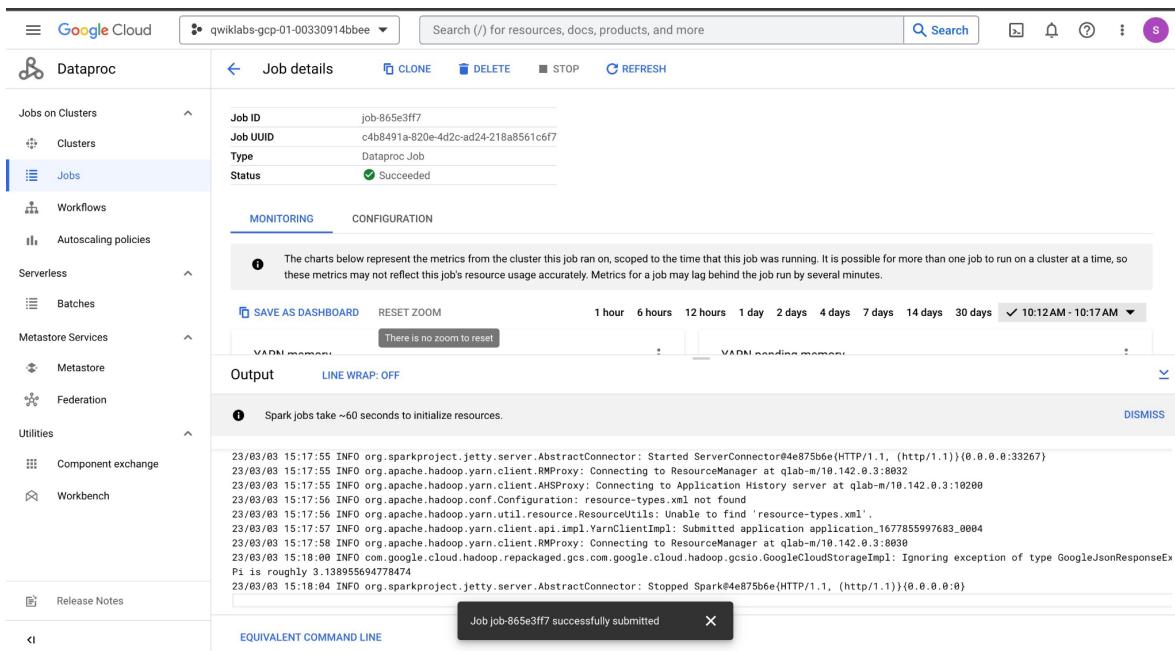
In the Introduction to Cloud Dataproc: Hadoop and Spark on Google Cloud lab, users build on their knowledge of Dataproc by learning how to create a cluster using the Cloud SDK command line. This lab also covers how to submit Hadoop and Spark jobs to the cluster and monitor their progress using the Dataproc web interface. Additionally, users learn how to use the Hadoop Distributed File System (HDFS) to store and manage large amounts of data, and how to run a simple Spark job using the PySpark library. This lab provides a more in-depth look at Dataproc's capabilities and shows users how to use its features to process and analyze large datasets.

## Task 1: Create a Cloud Dataproc cluster



The screenshot shows the Google Cloud Dataproc Clusters page. The left sidebar has sections for Clusters, Jobs, Workflows, Autoscaling policies, Serverless, and Batches. The main area shows a table for 'Jobs on Clusters'. A single cluster named 'glab' is listed, showing it is 'Running' in the 'us-east1' region with 2 worker nodes. The 'Cloud' dropdown is set to 'dataproc-99892'. There are tabs for 'PERMISSIONS' and 'LABELS'. A message at the bottom right says 'Please select at least one resource.'

## Task 2: Submit a Spark job to your cluster



The screenshot shows the Google Cloud Dataproc Job details page for a job with ID 'job-865e3ff7'. The left sidebar includes sections for Clusters, Jobs, Workflows, Autoscaling policies, Serverless, Batches, Metastore Services, Federation, Utilities, Component exchange, and Workbench. The main area displays monitoring charts for YARN memory and pending memory, and a log output window. The log shows the submission of a Spark job, with the final line indicating success: 'Job job-865e3ff7 successfully submitted'. A message at the bottom right says 'DISMISS'.

### Task 3: Shut down your cluster

Clusters

Name	Status	Region	Zone	Total worker nodes	Scheduled deletion
qlab	Deleting	us-east1	us-east1- c	2	Off

PERMISSIONS

ADD PRINCIPAL

Show inherited permissions

Role / Principal

- Dataproc Service Agent (1)
- Editor (3)
- Owner (3)
- Viewer (1)

### Task 4: Completion

End Lab

00:08:59

Caution: When you are in the console, do not deviate from the lab instructions. Doing so may cause your account to be blocked.

Learn more

Open Google Console

Username: student-04-179b4d3efda1

Password: adm1tv30L2Zuo

GCP Project ID: quiklabs-gcp-01-00330914bbbe

✓ Spark

Submit

Dataproc helps users process, transform and understand vast quantities of data.

✓ True

False

GSP123

Overview

Setup and requirements

Task 1. Create a Cloud Dataproc cluster

Task 2. Submit a Spark job to your cluster

Task 3. Shut down your cluster

Task 4. Test your understanding

Congratulations!

Next steps / learn more

10/10

### Congratulations!

You learned how to create a Dataproc cluster, submit a Spark job, and shut down your cluster!

Overall, these labs provide users with a solid foundation for working with Google Cloud Dataproc. They cover the basics of cluster creation and management, job submission, and monitoring, as well as more advanced topics such as HDFS and PySpark. By completing these labs, users will be well-equipped to use Dataproc to process and analyze big data on the cloud.