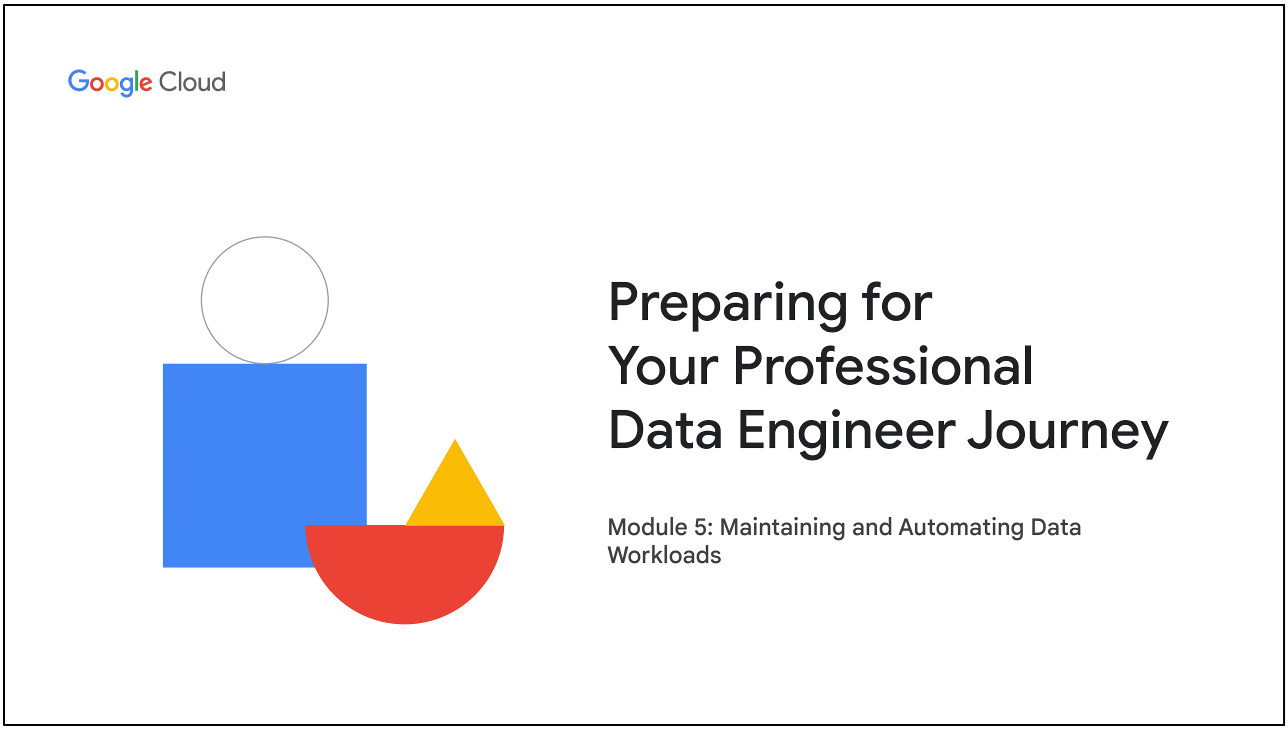
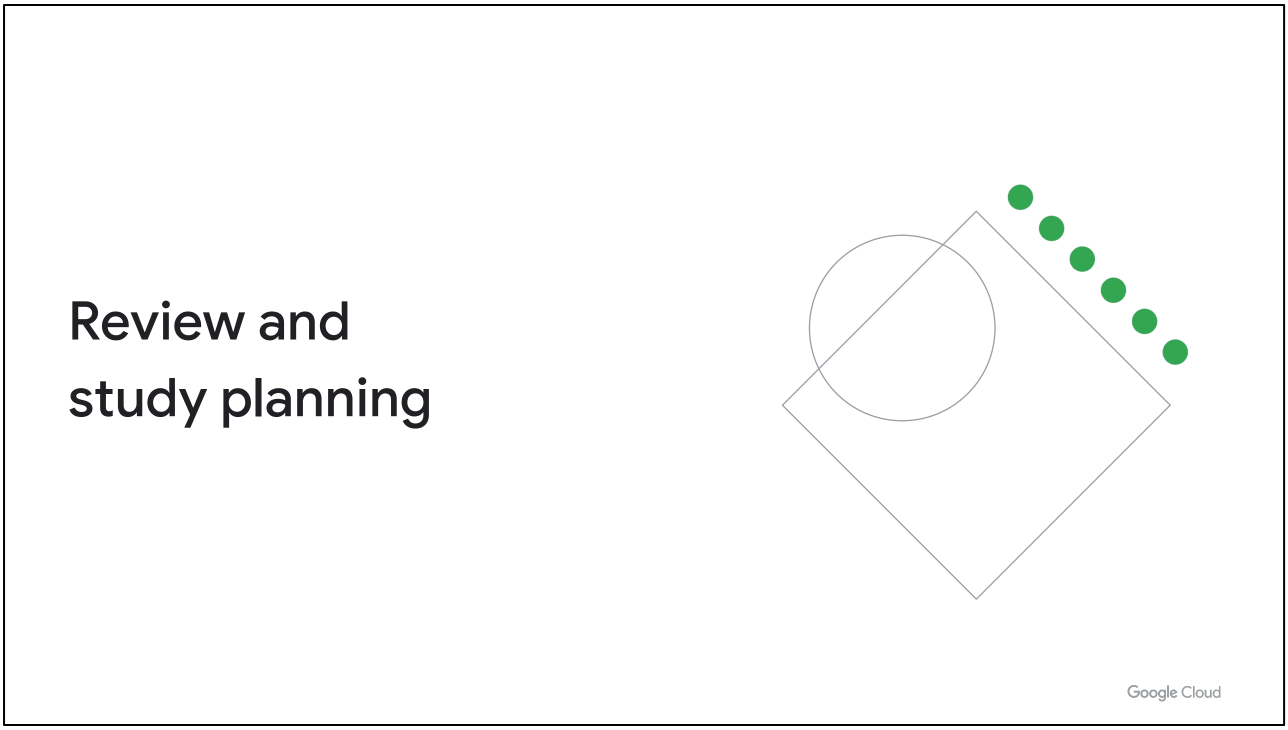


Google Cloud

Preparing for Your Professional Data Engineer Journey

Module 5: Maintaining and Automating Data Workloads

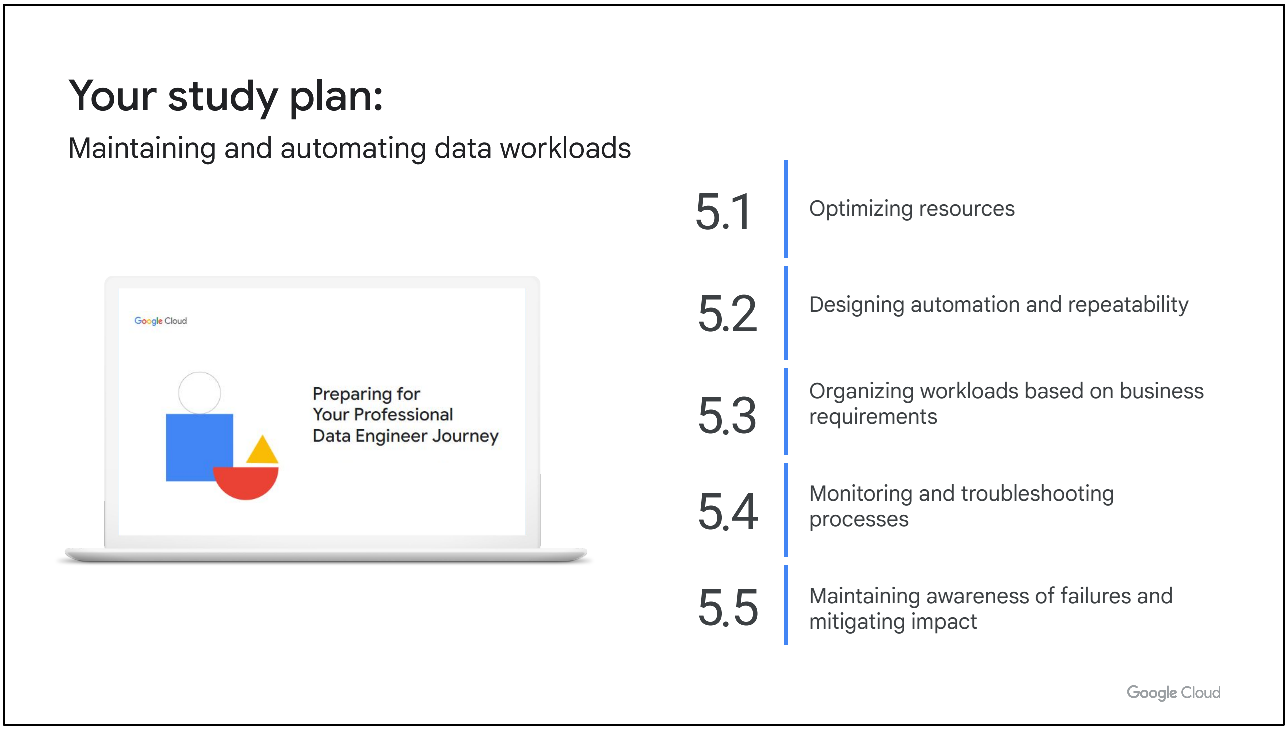
Welcome to Module 5: Maintaining and Automating Data Workloads.



Review and study planning

Now let’s review how to use these diagnostic questions to help you identify what to include in your study plan.

As a reminder - this course isn’t designed to teach you everything you need to know for the exam - and the diagnostic questions don’t cover everything that could be on the exam. Instead, this activity is meant to give you a better sense of the scope of this section and the different skills you’ll want to develop as you prepare for the certification.



Your study plan:

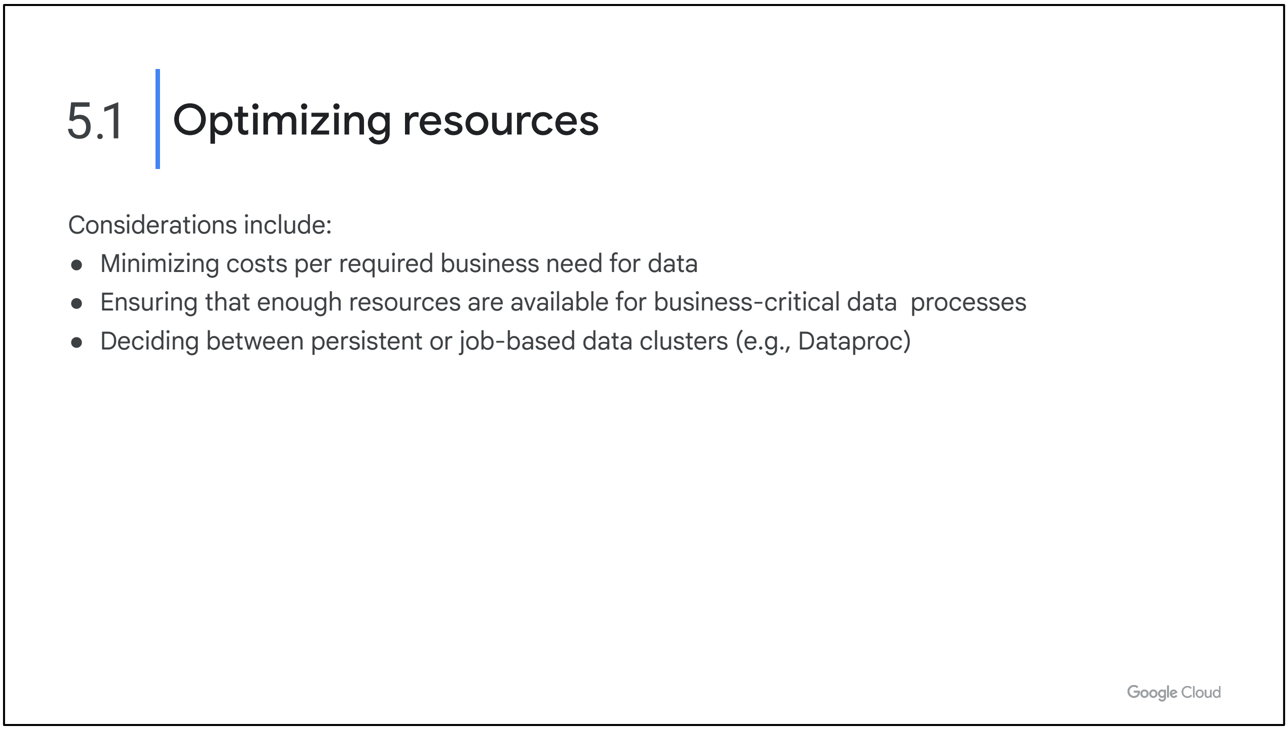
Maintaining and automating data workloads

Google Cloud
Preparing for
Your Professional
Data Engineer Journey

5.1	Optimizing resources
5.2	Designing automation and repeatability
5.3	Organizing workloads based on business requirements
5.4	Monitoring and troubleshooting processes
5.5	Maintaining awareness of failures and mitigating impact

Google Cloud

You'll approach this review by looking at the objectives of this exam section and the questions you just answered about each one. Let's introduce an objective, briefly review the answers to the related questions, then explain where you can find out more in the learning resources and/or in Google documentation. As you go through each section objective, use the page in your workbook to mark the specific documentation, courses, and skill badges you'll want to emphasize in your study plan.



5.1 Optimizing resources

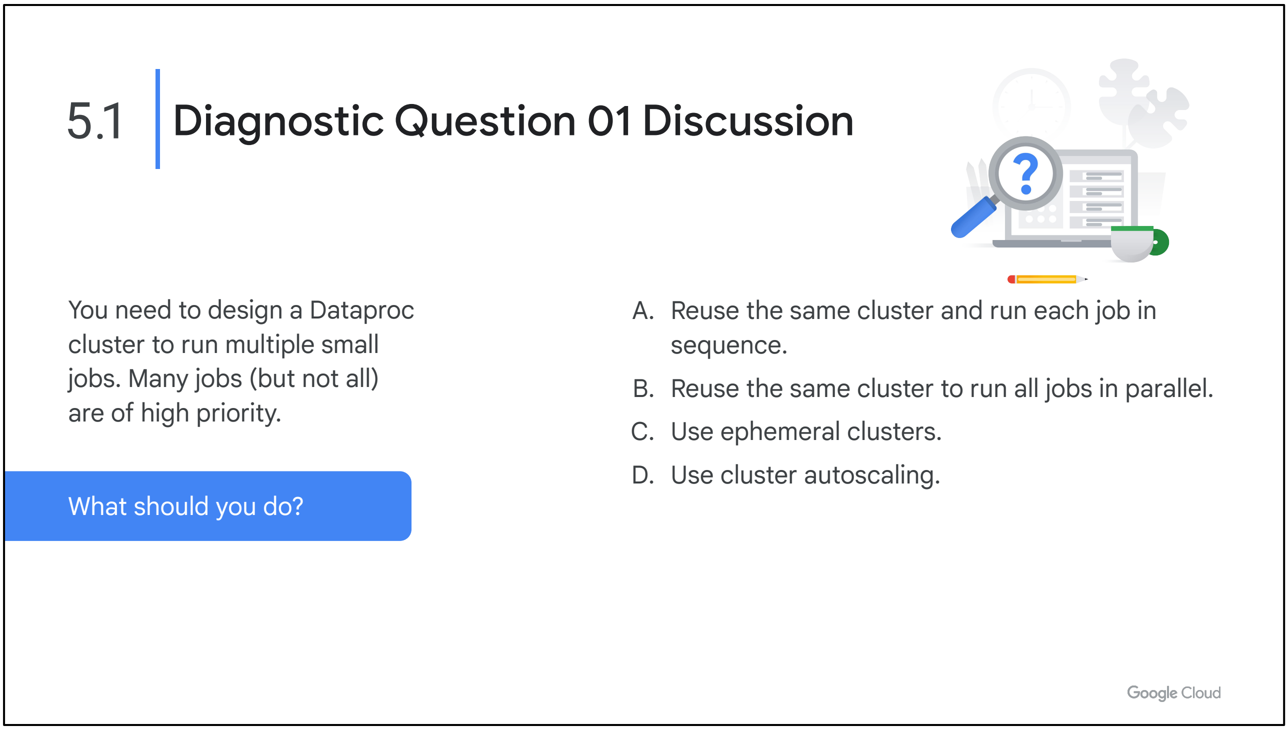
Considerations include:

- Minimizing costs per required business need for data
- Ensuring that enough resources are available for business-critical data processes
- Deciding between persistent or job-based data clusters (e.g., Dataproc)

Google Cloud

As a Professional Data Engineer, you need to provision enough resources to meet the performance requirements for various data processes. But at the same time, you need to minimize the costs that are associated with these resources. One way to minimize the cost is to decide whether you want persistent resources or job-based resources. Persistent resources ensure guaranteed throughput, but you need to pay a fixed price irrespective of the actual usage. On the other hand, job-based resources provide flexibility to create clusters only when they are required and thus pay based on the actual usage.

Question 1 tested your familiarity with options and considerations for deciding between persistent and job-based resources.



5.1 Diagnostic Question 01 Discussion

You need to design a Dataproc cluster to run multiple small jobs. Many jobs (but not all) are of high priority.

What should you do?

A. Reuse the same cluster and run each job in sequence.

B. Reuse the same cluster to run all jobs in parallel.

C. Use ephemeral clusters.

D. Use cluster autoscaling.

Google Cloud

Feedback:

A. Incorrect. Because many jobs are high priority, running them in sequence would not meet the business requirements.

B. Incorrect. With Dataproc, this is not a recommended approach. Configurations and running jobs could interfere with each other.

C. Correct. Jobs can use ephemeral clusters to quickly run the job and then deallocate the resources after use. Multiple jobs can be run in parallel without interfering with each other.

D. Incorrect. Cluster autoscaling is effective within a cluster running individual jobs, but it is not recommended when running multiple jobs because they can interfere with the resource scaling.

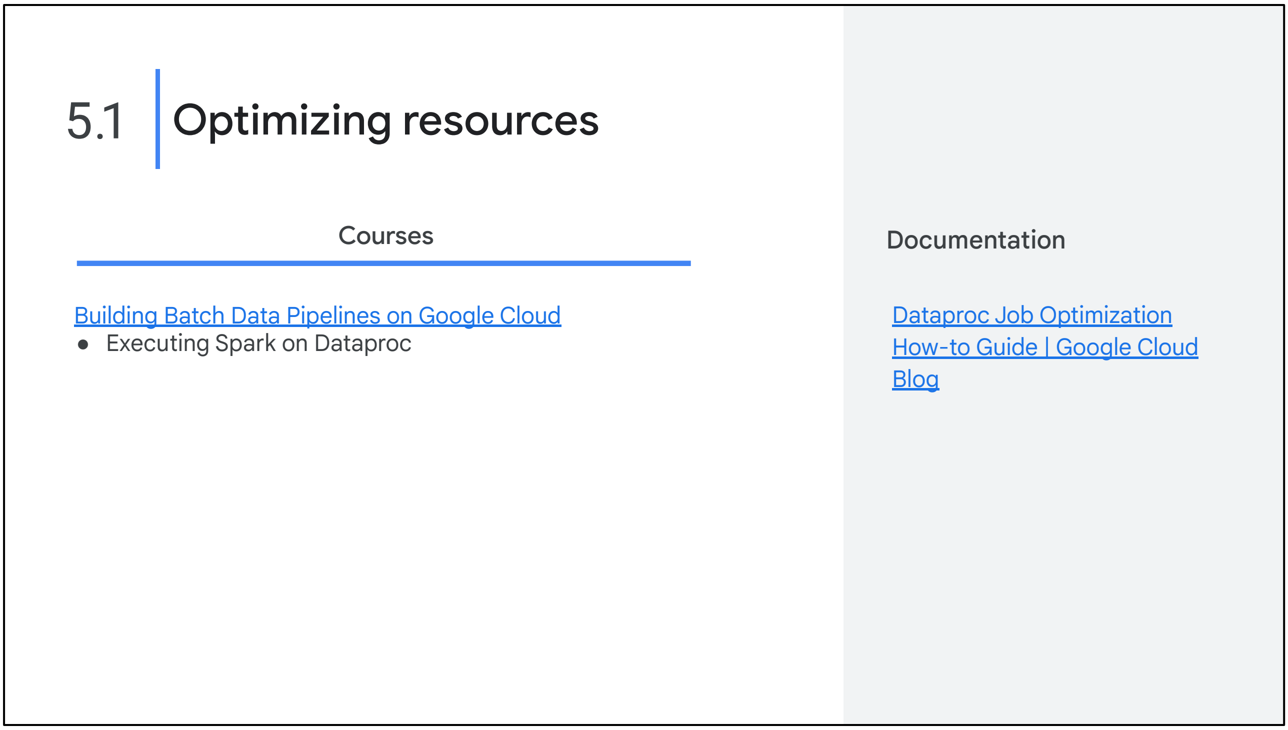
Links:
https://cloud.google.com/blog/products/data-analytics/dataproc-job-optimization-how-to-guide

More information:
Courses:
Building Batch Data Pipelines on Google Cloud

- Executing Spark on Dataproc

Summary:
Google Cloud's elasticity makes it convenient to allocate and deallocate resources

quickly. Each job can be isolated from others to avoid interference. The data engineer or the infrastructure engineer can also offload the resource management to Google Cloud.



5.1 Optimizing resources

Courses

Building Batch Data Pipelines on Google Cloud

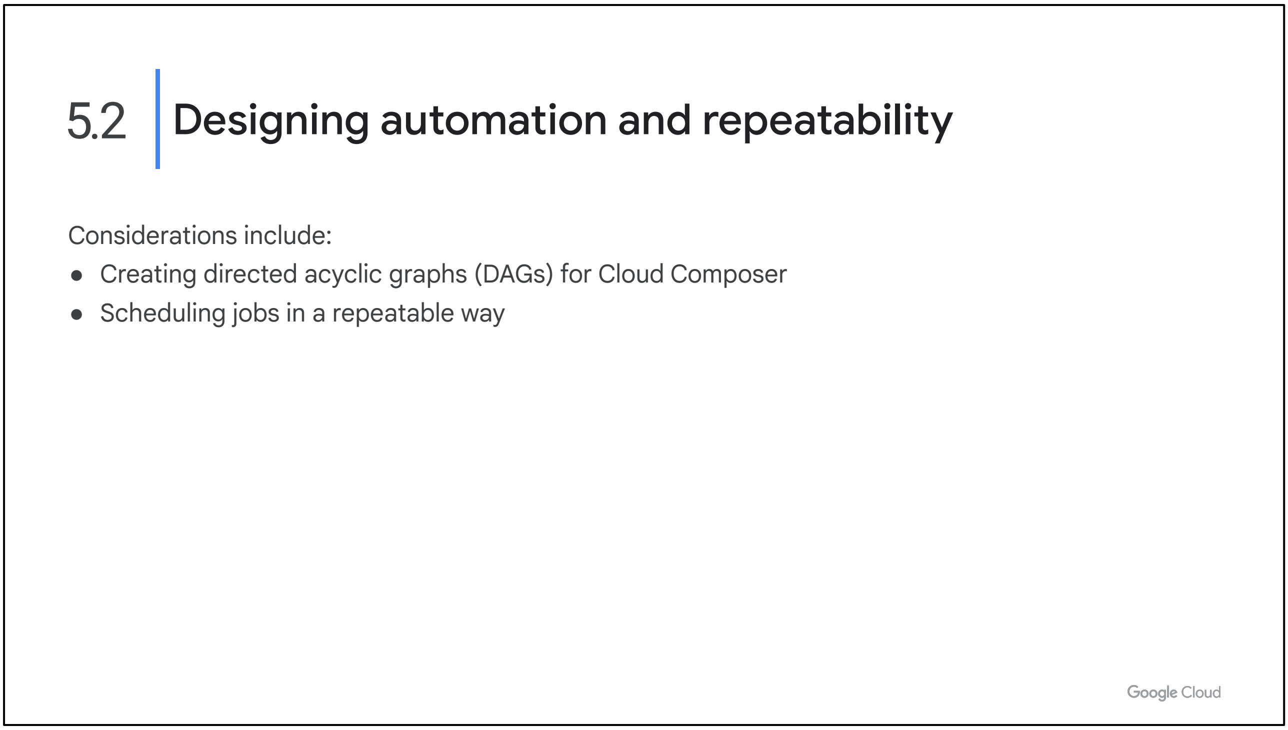
- Executing Spark on Dataproc

Documentation

Dataproc Job Optimization How-to Guide | Google Cloud Blog

You just reviewed a diagnostic question that addressed some aspects of optimizing resources. These are some courses, skill badges, and links to learn more about the concepts in these questions. They provide a starting point to explore Google-recommended practices.

https://cloud.google.com/blog/products/data-analytics/dataproc-job-optimization-how-to-guide



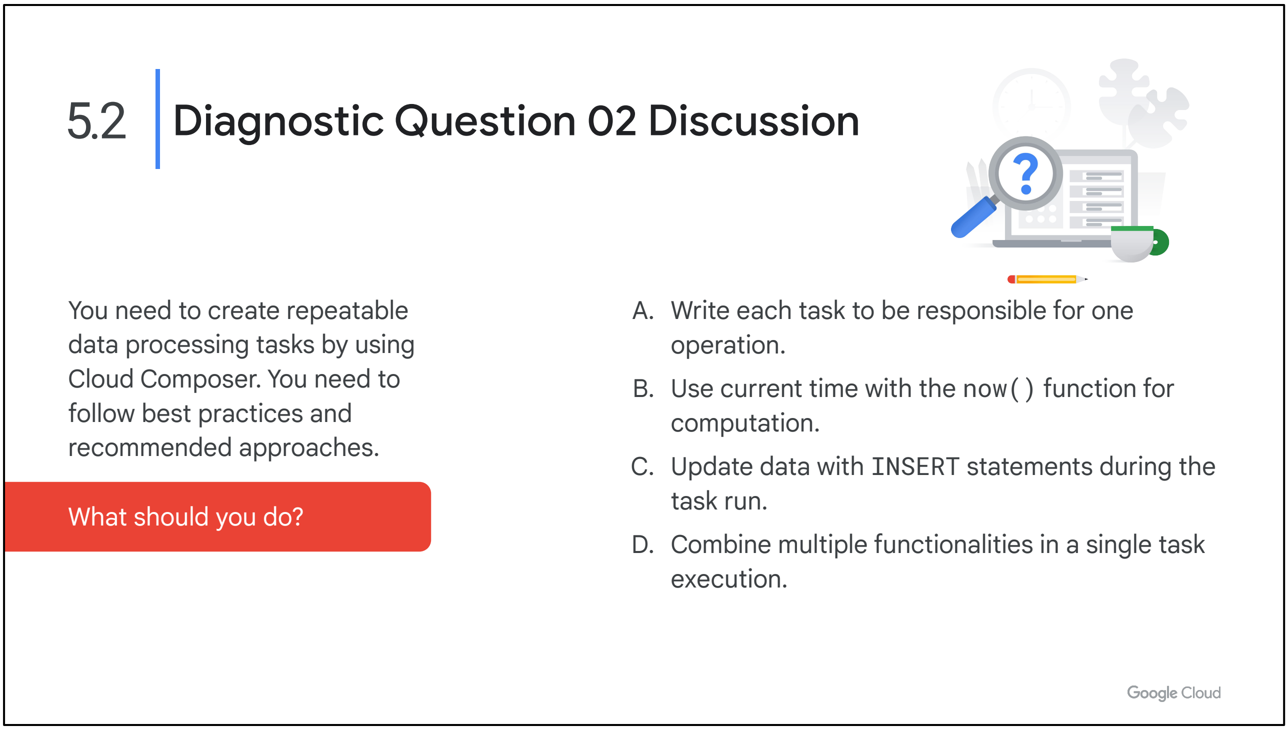
5.2 Designing automation and repeatability

Considerations include:

- Creating directed acyclic graphs (DAGs) for Cloud Composer
- Scheduling jobs in a repeatable way

As a Professional Data Engineer, you will work on increasing productivity by automating the repeatable workloads. You need to be fully familiar with Cloud Composer, a fully managed workflow orchestration service built on Apache Airflow that lets you author, schedule, and monitor your workflows. Cloud Composer lets you configure your pipelines as directed acyclic graphs (DAGs) using Python.

Question 2 asked you to describe how directed acyclic graphs (DAGs) support automation and repeatability of workloads.



5.2 Diagnostic Question 02 Discussion

You need to create repeatable data processing tasks by using Cloud Composer. You need to follow best practices and recommended approaches.

What should you do?

A. Write each task to be responsible for one operation.

B. Use current time with the `now()` function for computation.

C. Update data with `INSERT` statements during the task run.

D. Combine multiple functionalities in a single task execution.

Google Cloud

Feedback:

A. Correct. To run repeatable tasks, it is recommended to use atomic tasks that have a single responsibility. Many of these tasks can be combined in sequence to achieve a desired end result.

B. Incorrect. Using current time for data processing results in different and non-repeatable task runs.

C. Incorrect. Insert statements in tasks creates new entries. Rerunning the same tasks will insert new entries again. Instead of INSERT, UPSERT should be used.

D. Incorrect. It is recommended that a single task should take up a single functionality.

Links:

https://cloud.google.com/composer/docs/how-to/using/writing-dags
https://airflow.apache.org/docs/apache-airflow/stable/core-concepts/dags.html
https://docs.astronomer.io/learn/dag-best-practices

More information:

Courses:

Building Batch Data Pipelines on Google Cloud

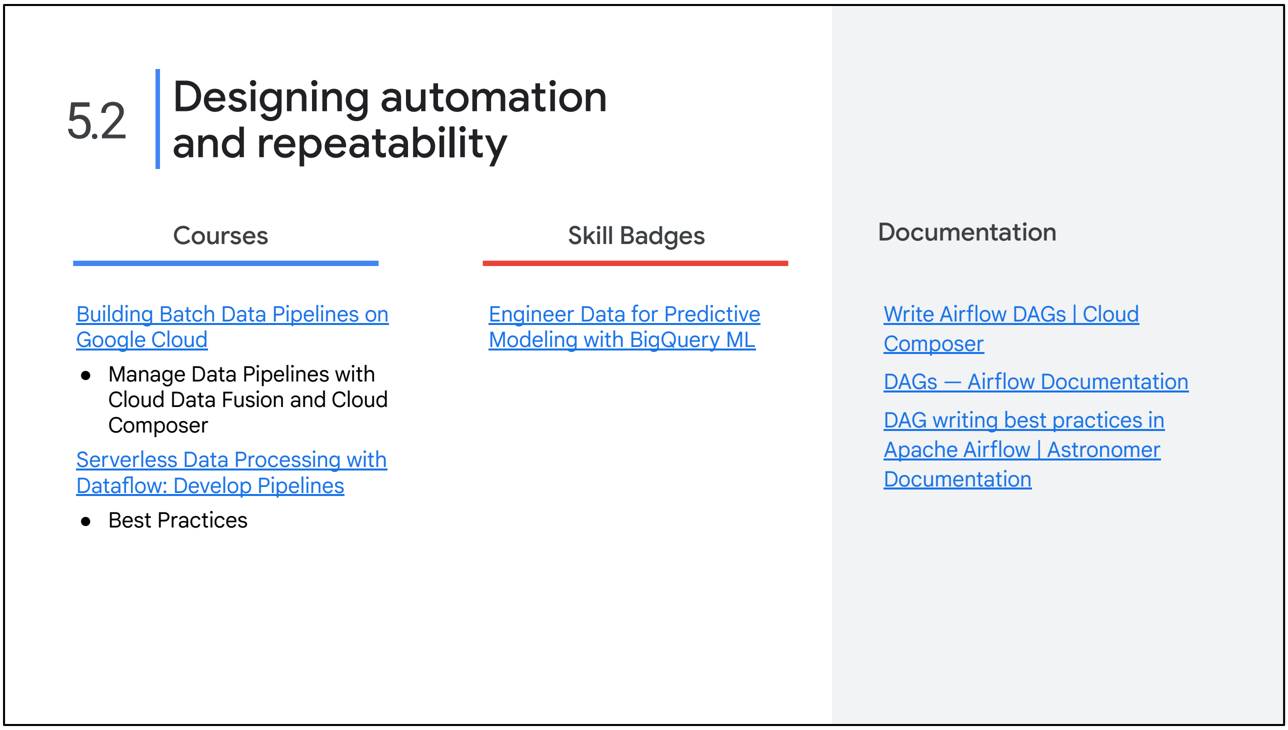
- Manage Data Pipelines with Cloud Data Fusion and Cloud Composer

Serverless Data Processing with Dataflow: Develop Pipelines

- Best Practices

Skill badge:
Engineer Data for Predictive Modeling with BigQuery ML

Summary:
Creating repeatable, idempotent tasks is preferable in data pipelines. The data engineer should design the entire data pipeline as a series of composable atomic tasks. For Cloud Composer, you can create DAGs (directed acyclic graphs) that link the atomic, idempotent tasks in a processing pipeline which itself will then become repeatable.



5.2 Designing automation and repeatability

Courses

Building Batch Data Pipelines on Google Cloud

- Manage Data Pipelines with Cloud Data Fusion and Cloud Composer

Serverless Data Processing with Dataflow: Develop Pipelines

- Best Practices

Skill Badges

Engineer Data for Predictive Modeling with BigQuery ML

Documentation

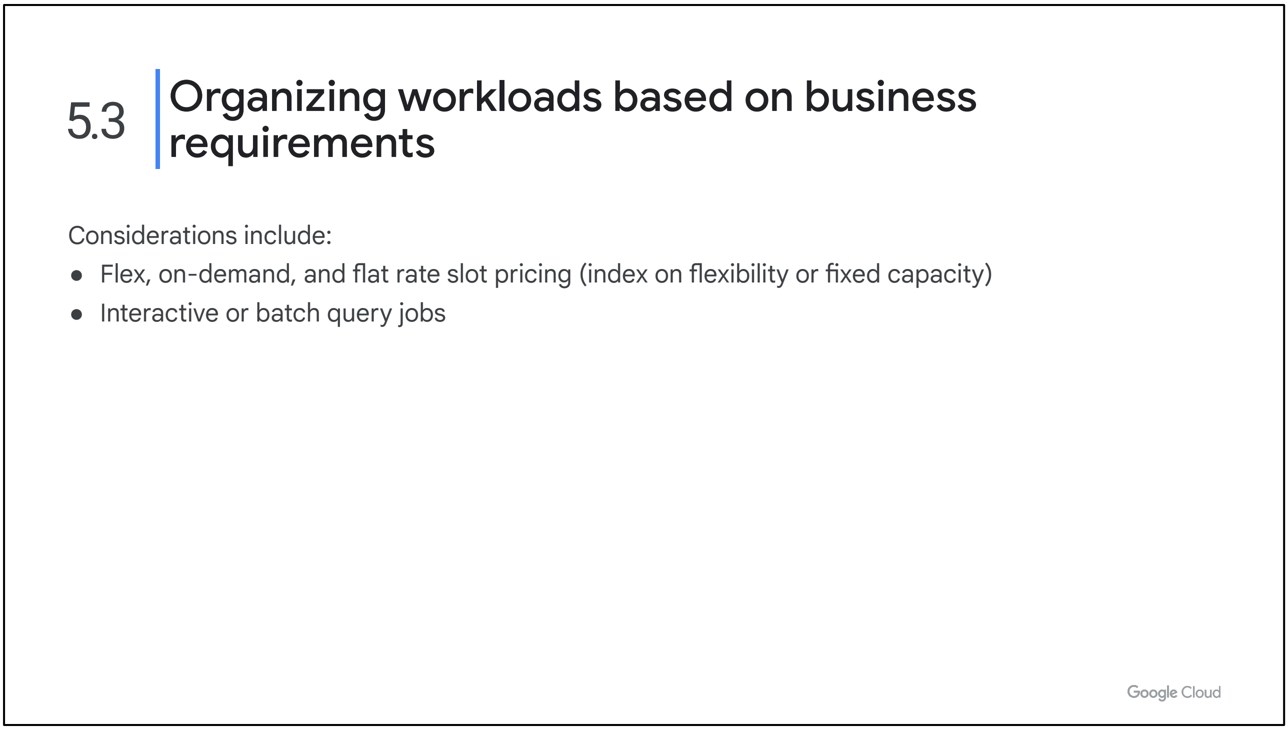
Write Airflow DAGs | Cloud Composer

DAGs — Airflow Documentation

DAG writing best practices in Apache Airflow | Astronomer Documentation

The diagnostic question you just reviewed explored some aspects of designing automation and repeatability. These are some courses and links to learn more about the concepts in these questions. They provide a starting point to explore Google-recommended practices.

https://cloud.google.com/composer/docs/how-to/using/writing-dags
https://airflow.apache.org/docs/apache-airflow/stable/core-concepts/dags.html
https://docs.astronomer.io/learn/dag-best-practices



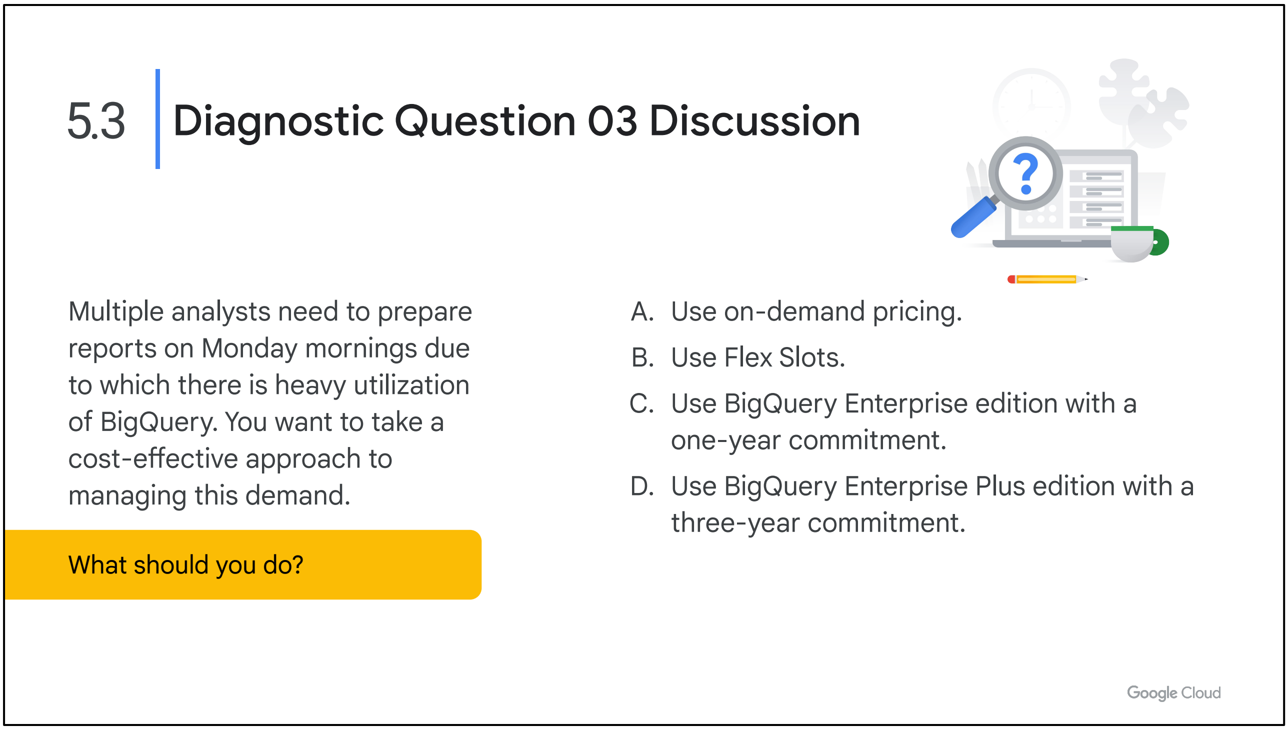
5.3 Organizing workloads based on business requirements

Considerations include:

- Flex, on-demand, and flat rate slot pricing (index on flexibility or fixed capacity)
- Interactive or batch query jobs

As a Professional Data Engineer, you need to be familiar with different pricing models for various Google Cloud services and should be able to select the appropriate pricing model based on the business use case. For example, BigQuery allows you to select between on-demand, flat-rate, and Flex pricing models. WIth the on-demand pricing model, you pay only for what you consume. The flat-rate pricing model provides dedicated resources and Flex pricing allows you to augment the existing capacity for a short period of time.

Question 3 asked you to differentiate between BigQuery slot options to organize and execute workloads based on business requirements. Question 4 asked you to differentiate between interactive and batch query jobs in BigQuery to organize and execute workloads based on business requirements.



5.3 Diagnostic Question 03 Discussion

Multiple analysts need to prepare reports on Monday mornings due to which there is heavy utilization of BigQuery. You want to take a cost-effective approach to managing this demand.

What should you do?

A. Use on-demand pricing.
B. Use Flex Slots.
C. Use BigQuery Enterprise edition with a one-year commitment.
D. Use BigQuery Enterprise Plus edition with a three-year commitment.

Google Cloud

Feedback:

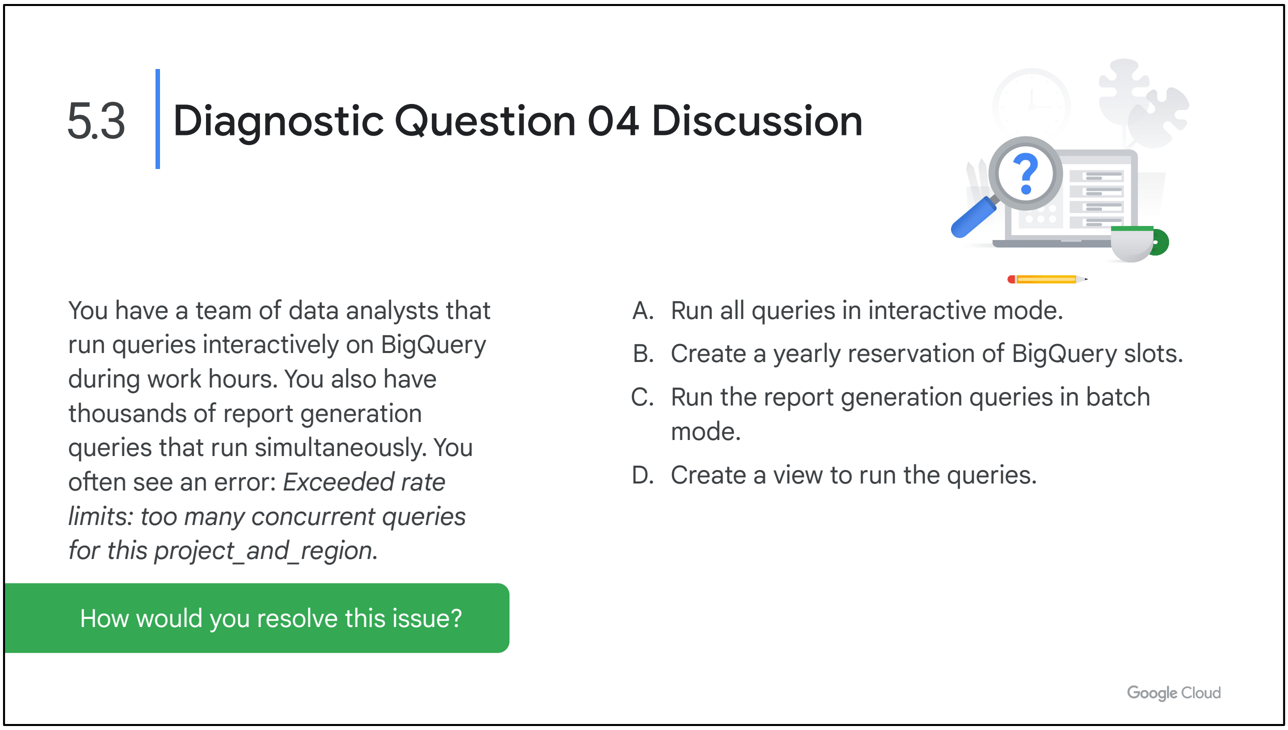
A. Incorrect. On-demand pricing is not cost-effective for this scenario.
B. Correct. Flex Slots let you reserve BigQuery slots for short durations.
C. Incorrect. A year-long commitment is unnecessary given the duration of the demand.
D. Incorrect. A multi year-long commitment is unnecessary given the duration of the demand.

Links:
https://cloud.google.com/blog/products/data-analytics/introducing-bigquery-flex-slots
https://cloud.google.com/bigquery/docs/reservations-intro
https://cloud.google.com/bigquery/docs/editions-intro

More information:
Courses:
Building Resilient Streaming Analytics Systems on Google Cloud

- Advanced BigQuery Functionality and Performance

Summary:
BigQuery has a variety of options for query pricing, including long-term commitments, short-term reservations, and on-demand pricing. A Professional Data Engineer can control costs by choosing the appropriate option for the duration of the analytics demand.



5.3 Diagnostic Question 04 Discussion

You have a team of data analysts that run queries interactively on BigQuery during work hours. You also have thousands of report generation queries that run simultaneously. You often see an error: *Exceeded rate limits: too many concurrent queries for this project_and_region.*

How would you resolve this issue?

A. Run all queries in interactive mode.
B. Create a yearly reservation of BigQuery slots.
C. Run the report generation queries in batch mode.
D. Create a view to run the queries.

Google Cloud

Feedback:

A. Incorrect. Running all queries in interactive mode will lead to the same error.
B. Incorrect. Reserving BigQuery slots will not avoid the error because concurrent query limits still apply.
C. Correct. Offloading the report generation queries to batch mode reduces the number of concurrent queries.
D. Incorrect. Queries run on views also count towards the concurrent queries limit, therefore, they do not solve this issue.

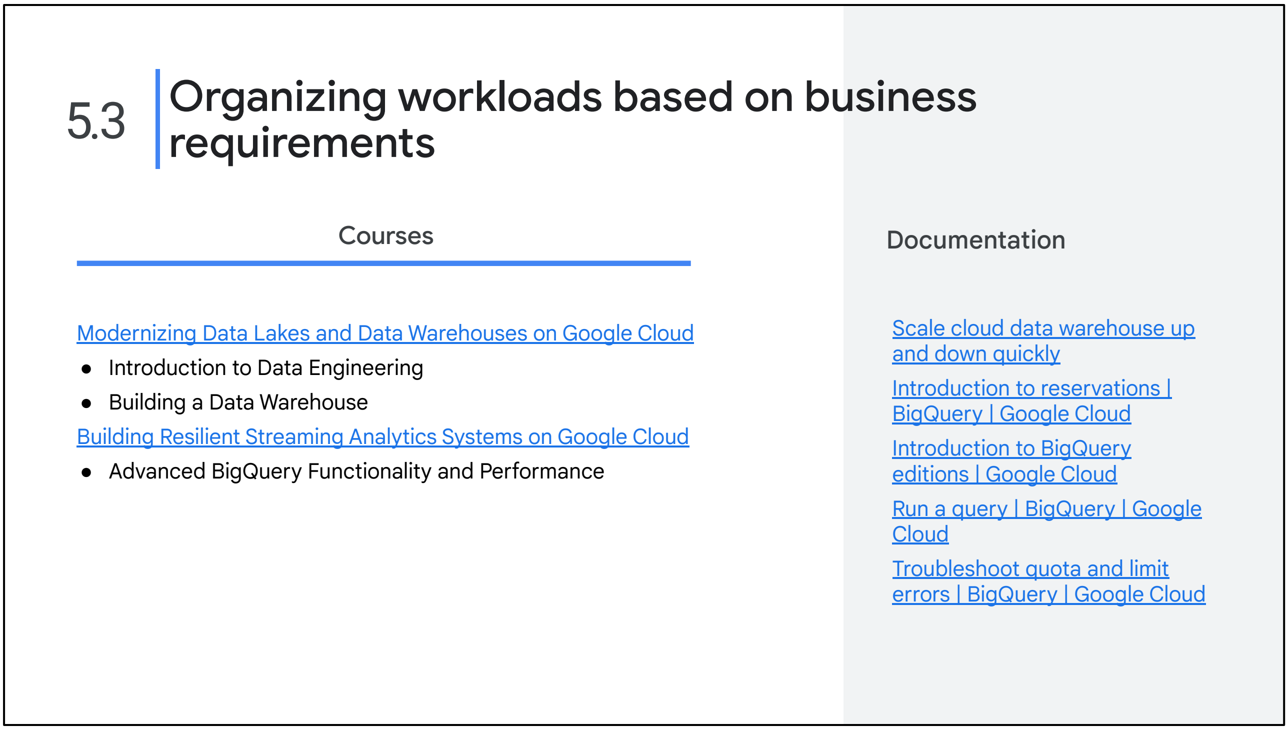
Links:
https://cloud.google.com/bigquery/docs/running-queries
https://cloud.google.com/bigquery/docs/troubleshoot-quotas#ts-concurrent-queries-quota

More information:
Courses:
Modernizing Data Lakes and Data Warehouses on Google Cloud

- Introduction to Data Engineering
- Building a Data Warehouse

Summary:
Google Cloud has quotas and imposes limits on the usage of certain resources. In some cases, your team can request increases in these quotas and limits. In other

cases, they are fixed. A Professional Data Engineer needs to know these limits to ensure availability of the services to their internal teams. Unlike interactive queries, batch queries don't count toward the concurrent limit. BigQuery runs the batch queries when idle resources are available, which will typically happen in a few minutes.



5.3 Organizing workloads based on business requirements

Courses

[Modernizing Data Lakes and Data Warehouses on Google Cloud]

- Introduction to Data Engineering
- Building a Data Warehouse

[Building Resilient Streaming Analytics Systems on Google Cloud]

- Advanced BigQuery Functionality and Performance

Documentation

Scale cloud data warehouse up and down quickly

Introduction to reservations | BigQuery | Google Cloud

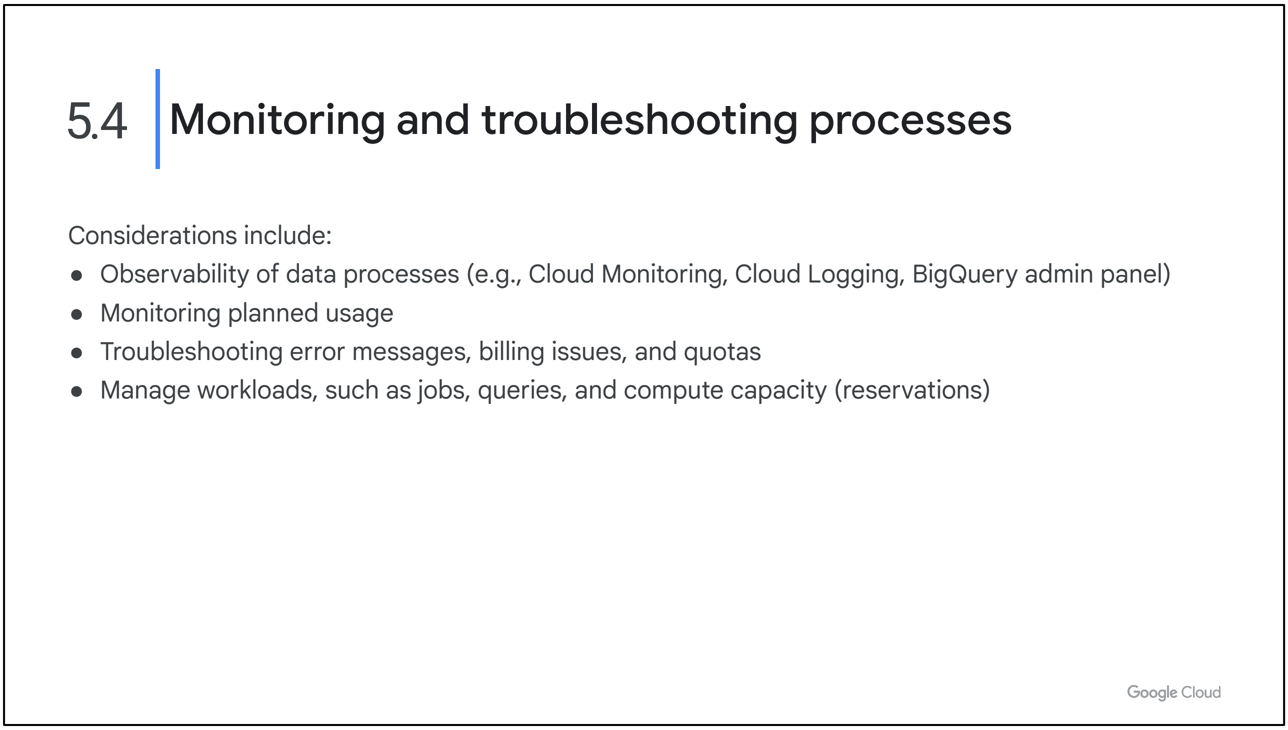
Introduction to BigQuery editions | Google Cloud

Run a query | BigQuery | Google Cloud

Troubleshoot quota and limit errors | BigQuery | Google Cloud

You just reviewed diagnostic questions that addressed considerations related to organizing your workloads based on business requirements. These are some courses, skill badges, and links to learn more about the concepts in these questions. They provide a starting point to explore Google-recommended practices.

https://cloud.google.com/blog/products/data-analytics/introducing-bigquery-flex-slots
https://cloud.google.com/bigquery/docs/reservations-intro
https://cloud.google.com/bigquery/docs/editions-intro



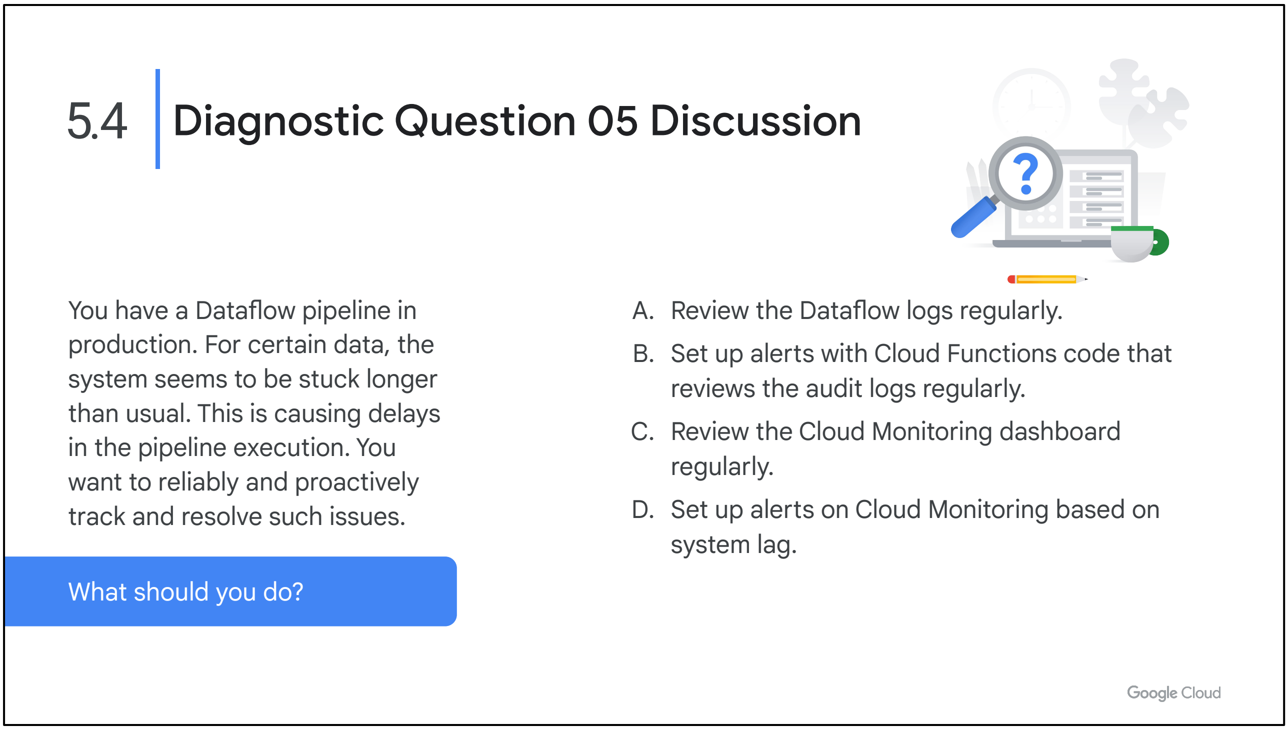
5.4 Monitoring and troubleshooting processes

Considerations include:

- Observability of data processes (e.g., Cloud Monitoring, Cloud Logging, BigQuery admin panel)
- Monitoring planned usage
- Troubleshooting error messages, billing issues, and quotas
- Manage workloads, such as jobs, queries, and compute capacity (reservations)

Once you have automated your workloads, it is very important to constantly monitor them and take a quick corrective action if some error arises. Google Cloud offers Cloud Monitoring, a service to gain visibility into the performance, availability, and health of your applications and infrastructure. It offers automatic out-of-the-box metric collection dashboards and ability to configure alerts using the complex rules. You also need to be familiar with other tools for monitoring and troubleshooting such as Cloud Logging and hte BigQuery admin panel.

Question 5 tested your knowledge of monitoring active and planned data workflows. Question 6 tested your knowledge of how to monitor and troubleshoot errors resulting from data workloads. Question 7 tested your familiarity with options for managing data workload jobs and compute capacity.



5.4 Diagnostic Question 05 Discussion

You have a Dataflow pipeline in production. For certain data, the system seems to be stuck longer than usual. This is causing delays in the pipeline execution. You want to reliably and proactively track and resolve such issues.

What should you do?

A. Review the Dataflow logs regularly.

B. Set up alerts with Cloud Functions code that reviews the audit logs regularly.

C. Review the Cloud Monitoring dashboard regularly.

D. Set up alerts on Cloud Monitoring based on system lag.

Google Cloud

Feedback:

A. Incorrect. Watching for delays in the Dataflow logs is not a viable or reliable solution.

B. Incorrect. Audit logs do not show information such as Dataflow pipeline system lag. Writing code to review logs and send alerts is cumbersome and is not recommended.

C. Incorrect. Watching for delays in the monitoring dashboard is not a viable or reliable solution.

D. Correct. Setting up alerts proactively notifies users about issues or metrics that need to be tracked.

Links:
https://cloud.google.com/dataflow/docs/guides/using-cloud-monitoring

More information:
Courses:
Modernizing Data Lakes and Data Warehouses on Google Cloud

- Introduction to Data Engineering

Building Batch Data Pipelines on Google Cloud

- Executing Spark on Dataproc

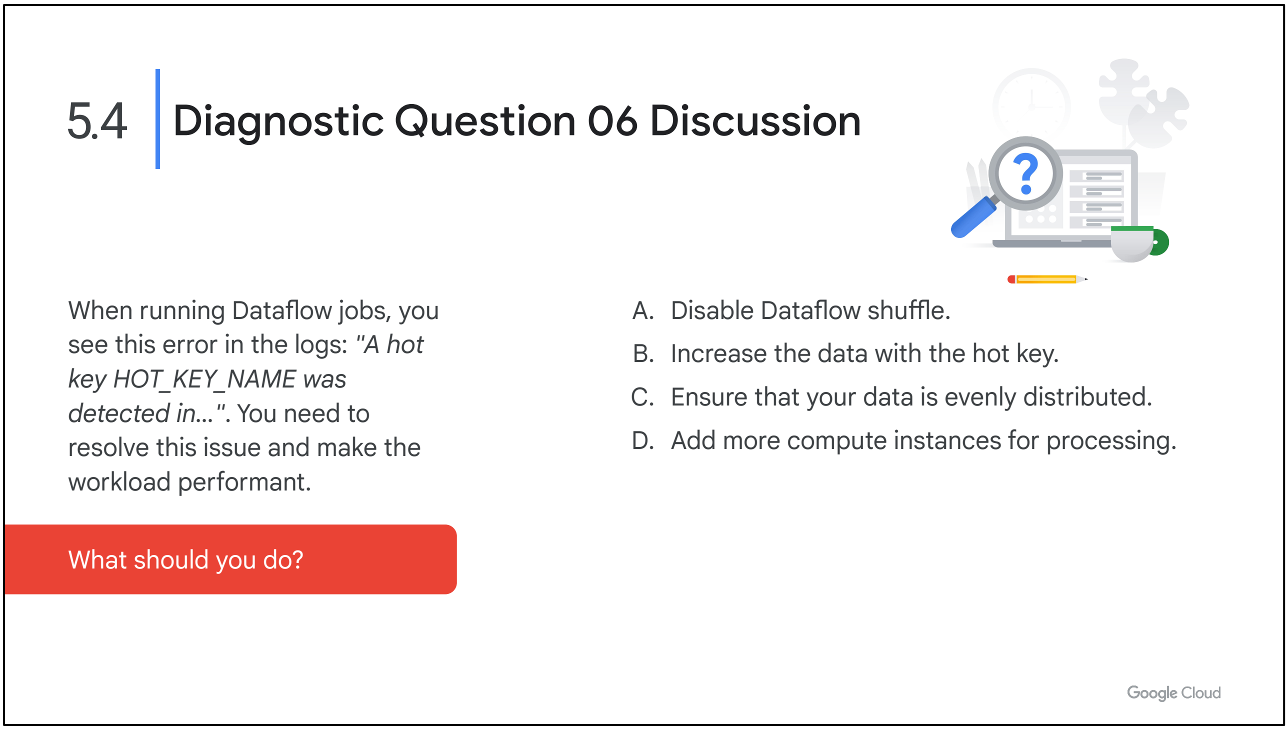
Building Resilient Streaming Analytics Systems on Google Cloud

- Serverless Messaging with Pub/Sub
- Advanced BigQuery Functionality and Performance

Serverless Data Processing with Dataflow: Operations

- Monitoring
- Troubleshooting and Debug

Summary:
There could be many more resources in your cloud environment. Tracking all of them or watching for changes in the logs or the dashboard is not a practical option. Instead, the data engineer can set up critical notifications that will inform the user of reached thresholds.



5.4 Diagnostic Question 06 Discussion

When running Dataflow jobs, you see this error in the logs: *"A hot key HOT_KEY_NAME was detected in…"*. You need to resolve this issue and make the workload performant.

What should you do?

A. Disable Dataflow shuffle.
B. Increase the data with the hot key.
C. Ensure that your data is evenly distributed.
D. Add more compute instances for processing.

Feedback:

A. Incorrect. It is recommended to enable Dataflow Shuffle because it partitions and groups data by key in a scalable, efficient, fault-tolerant manner.
B. Incorrect. Increasing the amount of data with the same hot key will increase hotspots, making the data processing less efficient.
C. Correct. The Dataflow transformations are more performant with an evenly distributed key.
D. Incorrect. Adding more compute instances does not automatically resolve hot spots in the data.

Links:
https://cloud.google.com/dataflow/docs/guides/common-errors#hot-key-detected
https://cloud.google.com/dataflow/docs/guides/troubleshoot-stragglers

More information:
Courses:
Serverless Data Processing with Dataflow: Foundations

- IAM, Quotas, and Permissions

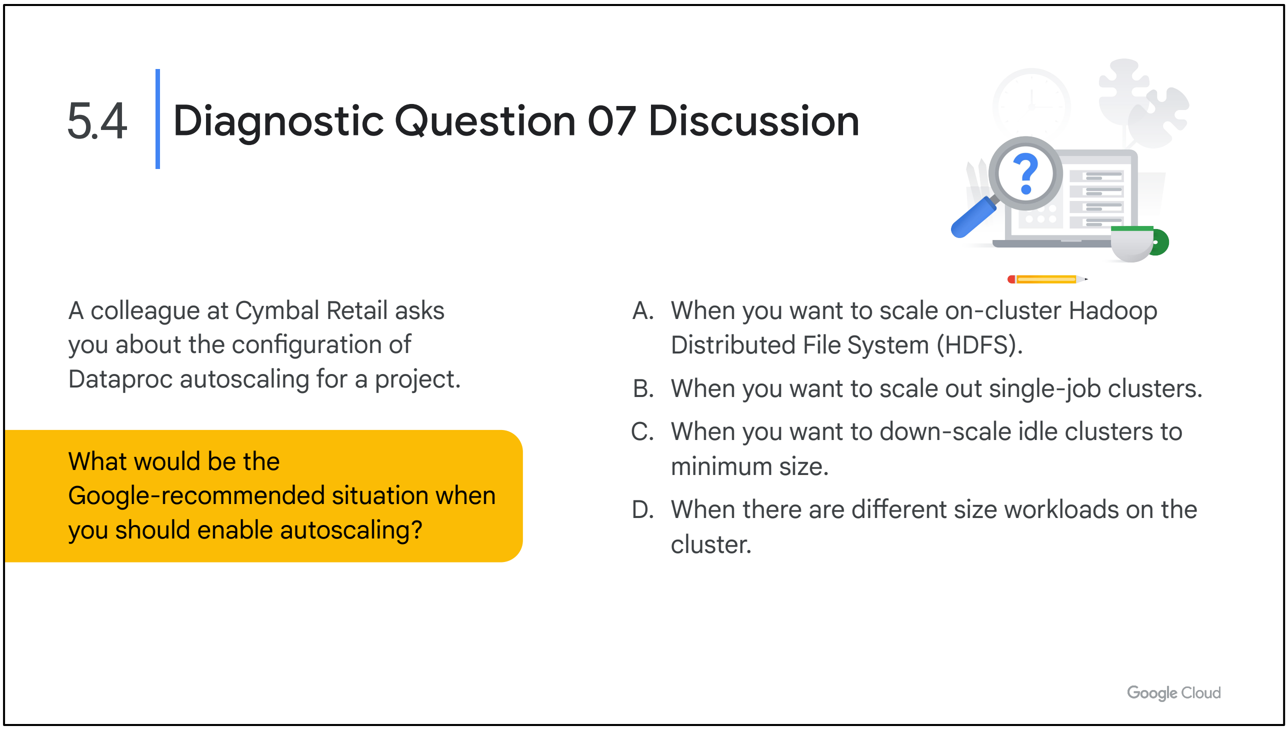
Serverless Data Processing with Dataflow: Develop Pipelines

- State and Timers
- Best Practices

Serverless Data Processing with Dataflow: Operations

- Troubleshooting and Debug
- Reliability

Summary:
The performance of large data processing systems and algorithms is usually negatively affected by keys that are bunched together in a small lexicographic range. A Professional Data Engineer should be able to troubleshoot issues by interpreting errors, examining logs, and monitoring cloud environments. In this case, one can deduce from the error that this is a hot key issue and can improve processing performance by re-keying the data and ensuring that it is more evenly distributed.



5.4 Diagnostic Question 07 Discussion

A colleague at Cymbal Retail asks you about the configuration of Dataproc autoscaling for a project.

What would be the Google-recommended situation when you should enable autoscaling?

A. When you want to scale on-cluster Hadoop Distributed File System (HDFS).
B. When you want to scale out single-job clusters.
C. When you want to down-scale idle clusters to minimum size.
D. When there are different size workloads on the cluster.

Google Cloud

Feedback:

A. Incorrect. Since HDFS utilization is not a signal for autoscaling, this would not be a good use case.
B. Correct. Single job clusters are well suited for autoscaling because there won't be any overlap with scaling of other jobs.
C. Incorrect. On Dataproc, it is recommended that you delete idle clusters instead of scaling down to minimum size because it is quick and cost-efficient.
D. Incorrect. Running different size workloads on the same cluster can cause interference. Long-running jobs might interfere with and delay the downscaling of smaller jobs.

Links:
https://cloud.google.com/dataflow/docs/guides/common-errors#hot-key-detected
https://cloud.google.com/dataflow/docs/guides/troubleshoot-stragglers

More information:
Courses:
Building Batch Data Pipelines on Google Cloud

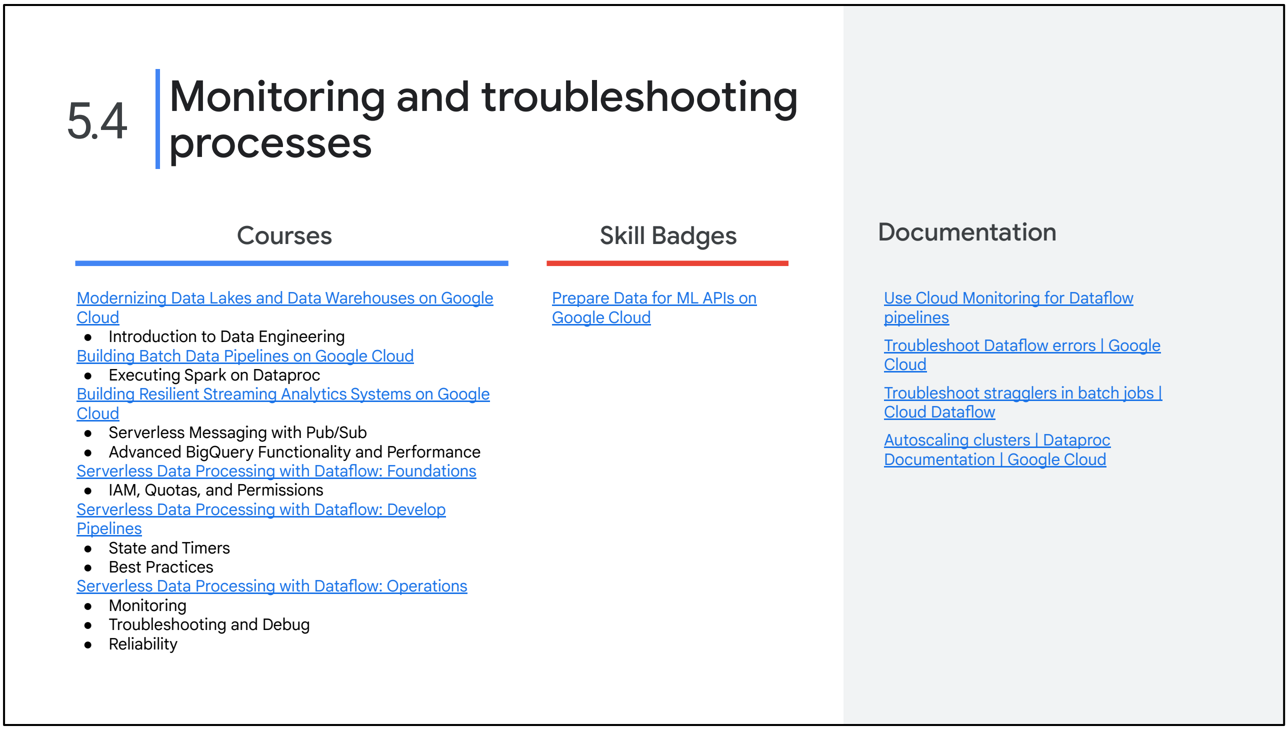
- Executing Spark on Dataproc

Serverless Data Processing with Dataflow: Develop Pipelines

- Best Practices

Skill badge:
Prepare Data for ML APIs on Google Cloud

Summary:
Autoscaling in Google Cloud lets you offload the task of managing infrastructure scaling to Google Cloud. However, autoscaling comes with certain caveats. Certain kinds of Dataproc jobs are more suitable for autoscaling than others. In certain cases, it might be preferable to disable autoscaling and use other approaches to spin down and delete clusters.



5.4 Monitoring and troubleshooting processes

Courses

Modernizing Data Lakes and Data Warehouses on Google Cloud

- Introduction to Data Engineering

Building Batch Data Pipelines on Google Cloud

- Executing Spark on Dataproc

Building Resilient Streaming Analytics Systems on Google Cloud

- Serverless Messaging with Pub/Sub
- Advanced BigQuery Functionality and Performance

Serverless Data Processing with Dataflow: Foundations

- IAM, Quotas, and Permissions

Serverless Data Processing with Dataflow: Develop Pipelines

- State and Timers
- Best Practices

Serverless Data Processing with Dataflow: Operations

- Monitoring
- Troubleshooting and Debug
- Reliability

Skill Badges

Prepare Data for ML APIs on Google Cloud

Documentation

Use Cloud Monitoring for Dataflow pipelines

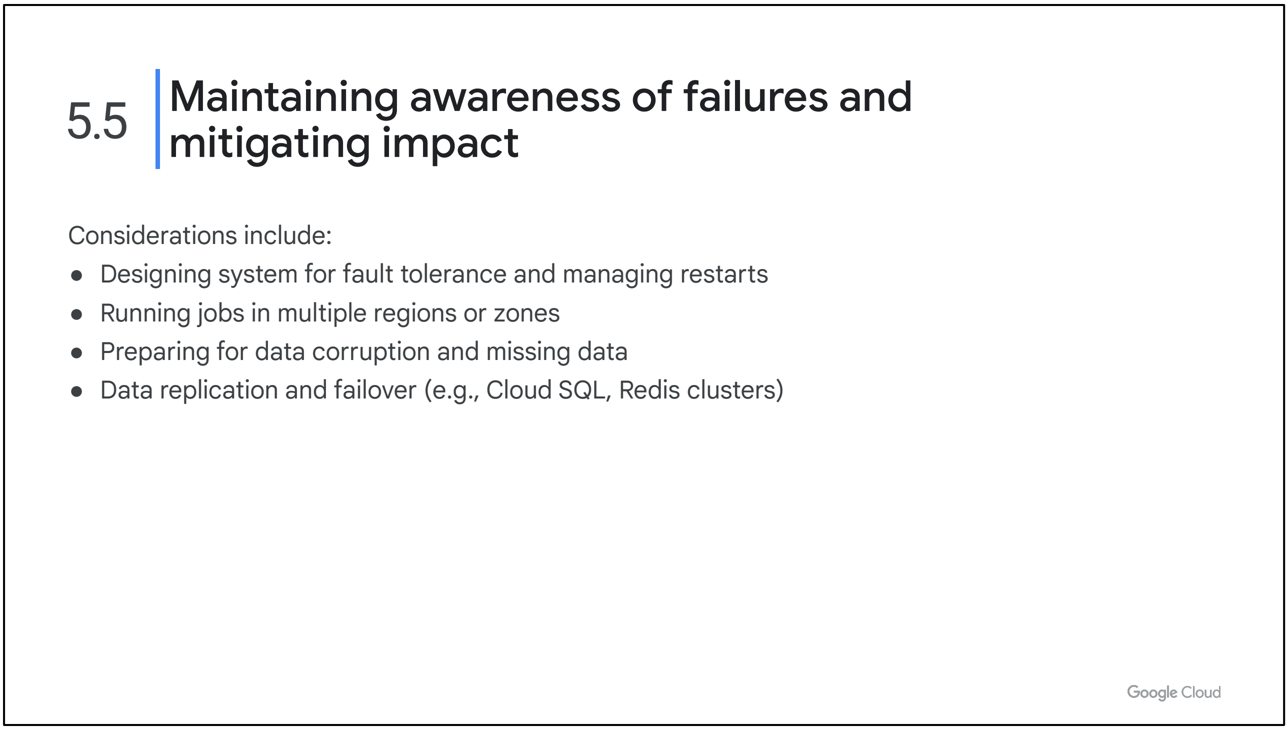
Troubleshoot Dataflow errors | Google Cloud

Troubleshoot stragglers in batch jobs | Cloud Dataflow

Autoscaling clusters | Dataproc Documentation | Google Cloud

You just reviewed several diagnostic questions that addressed considerations related to monitoring and troubleshooting. These are some courses, skill badges, and links to learn more about the concepts in these questions. They provide a starting point to explore Google-recommended practices.

https://cloud.google.com/dataflow/docs/guides/using-cloud-monitoring
https://cloud.google.com/dataflow/docs/guides/common-errors#hot-key-detected
https://cloud.google.com/dataflow/docs/guides/troubleshoot-stragglers
https://cloud.google.com/dataproc/docs/concepts/configuring-clusters/autoscaling



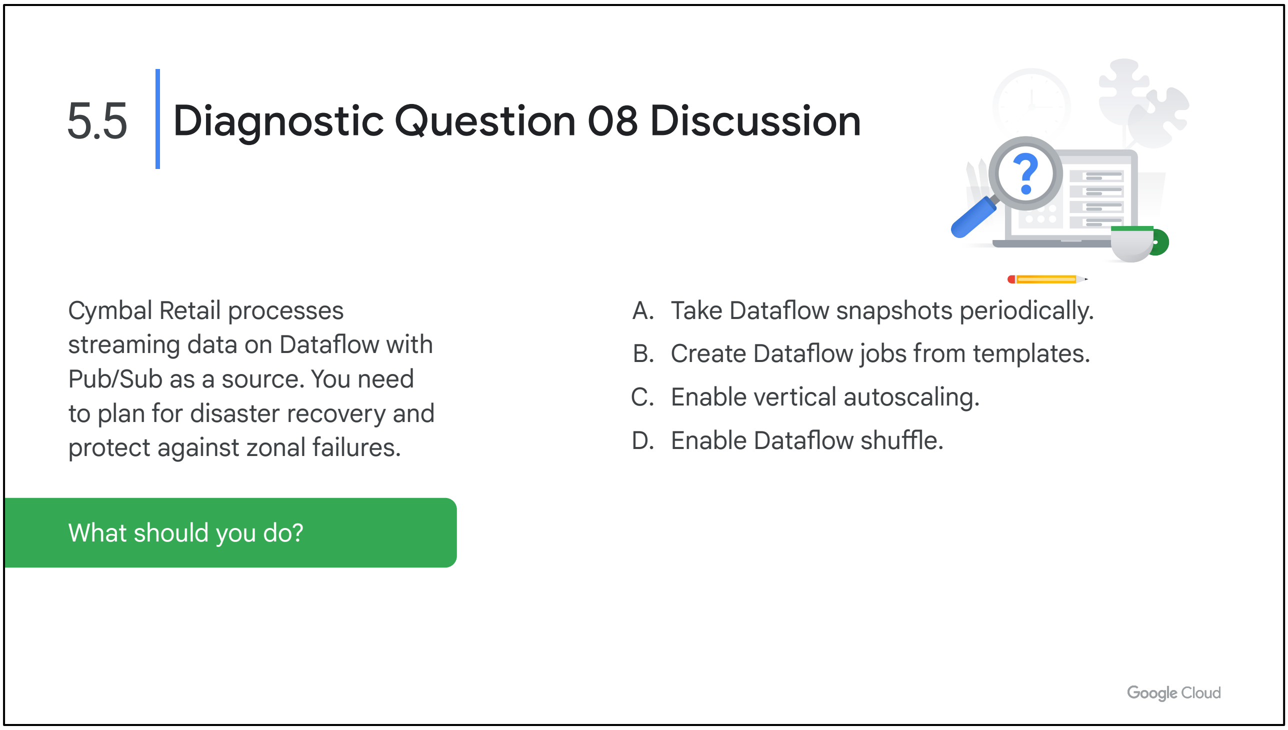
5.5 Maintaining awareness of failures and mitigating impact

Considerations include:

- Designing system for fault tolerance and managing restarts
- Running jobs in multiple regions or zones
- Preparing for data corruption and missing data
- Data replication and failover (e.g., Cloud SQL, Redis clusters)

As a Professional Data Engineer, you should be able to ensure process continuity even when some failure happens. Google Cloud offers multiple options that use redundancy to build fault tolerant systems. For example, you can store your data across multiple zones and regions and thus improve data availability. Or, when using a storage service like Cloud SQL, you can turn on data replication and automatic failover.

Question 8 tested your knowledge of how to run jobs in multiple regions and zones to support fault tolerance. Question 9 tested your ability to outline strategies for data replication and failover. Question 10 tested your familiarity with options for handling corrupt and missing data.



5.5 Diagnostic Question 08 Discussion

Cymbal Retail processes streaming data on Dataflow with Pub/Sub as a source. You need to plan for disaster recovery and protect against zonal failures.

What should you do?

A. Take Dataflow snapshots periodically.
B. Create Dataflow jobs from templates.
C. Enable vertical autoscaling.
D. Enable Dataflow shuffle.

Google Cloud

Feedback:

A. Correct. When running streaming pipelines, Dataflow snapshots can save the current state. You can then start a new job based on the saved state. This allows you to recover from failures and also migrate jobs.
B. Incorrect. Creating jobs from templates makes it easy to start new jobs by changing parameter values. However, since templates do not retain any state, they are not useful for disaster recovery.
C. Incorrect. Vertical autoscaling can dynamically allocate compute capacity based on worker utilization, but it is not relevant for disaster recovery.
D. Incorrect. Horizontal autoscaling can automatically change the number of workers allocated to your job, but it is not relevant for disaster recovery.

Links:
https://cloud.google.com/dataflow/docs/guides/using-snapshots

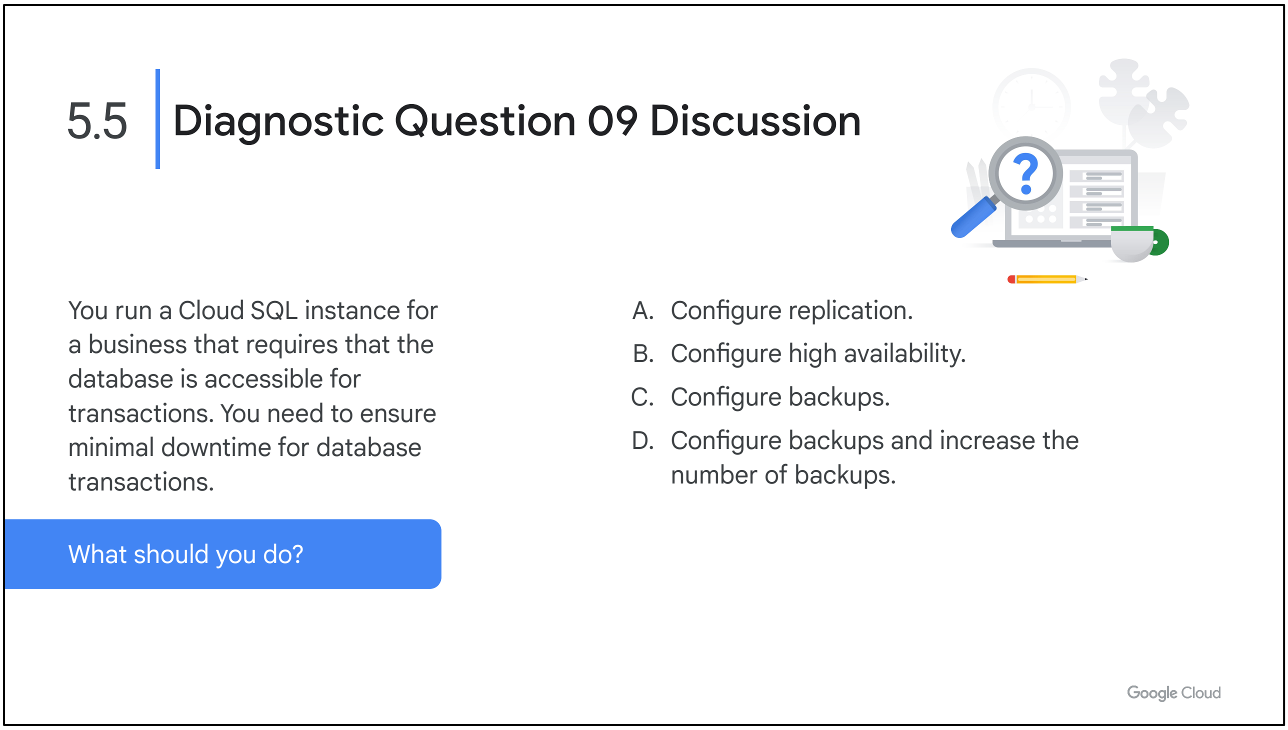
More information:
Courses:
Serverless Data Processing with Dataflow: Develop Pipelines

- Best Practices

Serverless Data Processing with Dataflow: Operations

- Reliability

Summary:
Many types of failures occur in any system. The data engineer not only has to design systems to mitigate those failures but also plan for catastrophic failure. Disaster recovery planning takes into consideration your business objectives like RPO (recovery point objective), RTO (recovery time objective), and cost. In Dataflow, taking periodic snapshots can help you recover from catastrophic failure and restart the jobs from saved state.



5.5 Diagnostic Question 09 Discussion

You run a Cloud SQL instance for a business that requires that the database is accessible for transactions. You need to ensure minimal downtime for database transactions.

What should you do?

A. Configure replication.
B. Configure high availability.
C. Configure backups.
D. Configure backups and increase the number of backups.

Google Cloud

Feedback:

A. Incorrect. Replication on Cloud SQL is effective to offload reads. However, it does not support a full read-write database.
B. Correct. Configuring high availability on Cloud SQL will automatically switch to the secondary instance when the primary instance goes down, thus reducing downtime for the database's users.
C. Incorrect. Backups are useful to retrieve older data in case of catastrophic failures such as accidental data deletion. However, they don’t provide continuous availability of the database, because restoring the database takes time.
D. Incorrect. Increasing the number of backups does not impact or improve availability. Although there are more backups, restoring the database in case of failure incurs downtime.

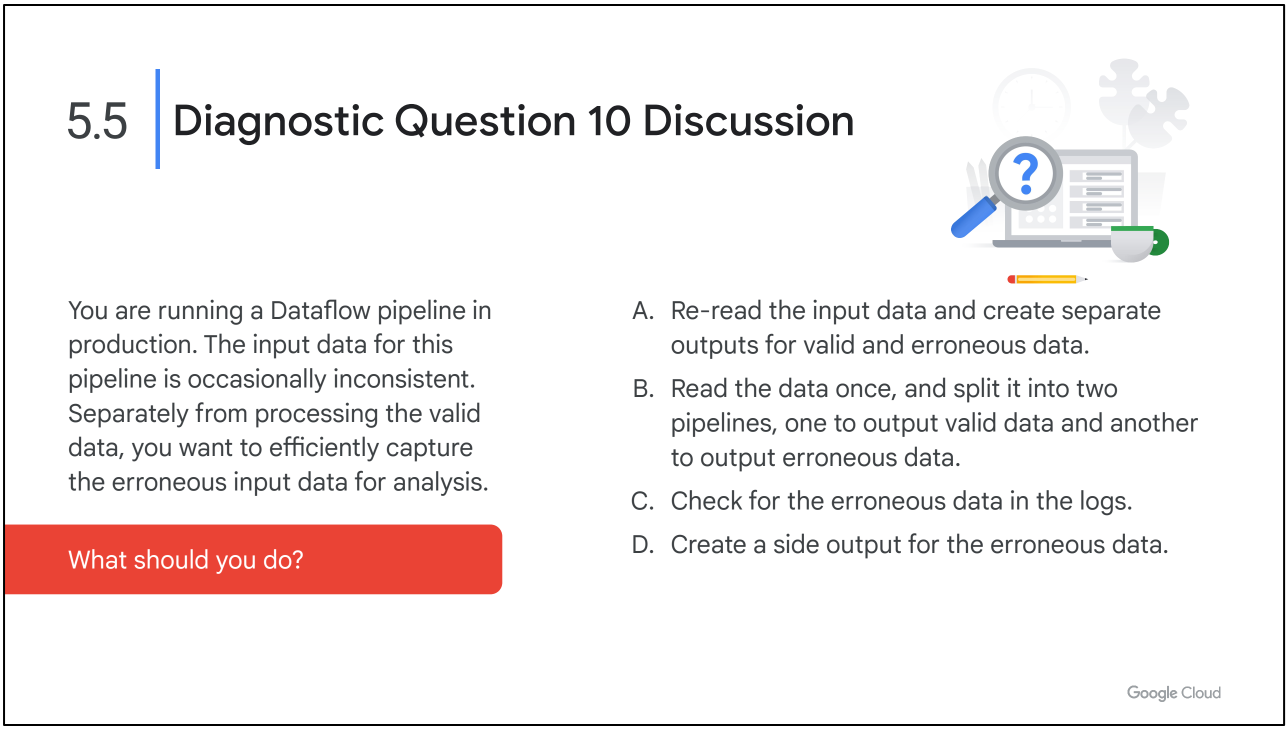
Links:
https://cloud.google.com/sql/docs/mysql/high-availability

More information:
Courses:
Modernizing Data Lakes and Data Warehouses on Google Cloud

- Building a Data Lake

Summary:

Cloud SQL high availability configuration has a primary and secondary instance. Data updates are visible from both instances. In the event of a database failure, the database engineer does not have to manually restore the database instance. Google Cloud will automatically switch the secondary instance to primary, thus maintaining database availability.



5.5 Diagnostic Question 10 Discussion

You are running a Dataflow pipeline in production. The input data for this pipeline is occasionally inconsistent. Separately from processing the valid data, you want to efficiently capture the erroneous input data for analysis.

What should you do?

A. Re-read the input data and create separate outputs for valid and erroneous data.

B. Read the data once, and split it into two pipelines, one to output valid data and another to output erroneous data.

C. Check for the erroneous data in the logs.

D. Create a side output for the erroneous data.

Google Cloud

Feedback:

A. Incorrect. Re-reading the input data for processing only erroneous data is not efficient.

B. Incorrect. Using separate pipelines for the same data (one to compute good data and the other for erroneous data) is not efficient.

C. Incorrect. Erroneous data is not automatically available in the logs.

D. Correct. Using side outputs can collect the erroneous data efficiently and is a recommended approach.

Links:

https://beam.apache.org/documentation/pipelines/design-your-pipeline/#a-single-transform-that-produces-multiple-outputs

More information:

Courses:

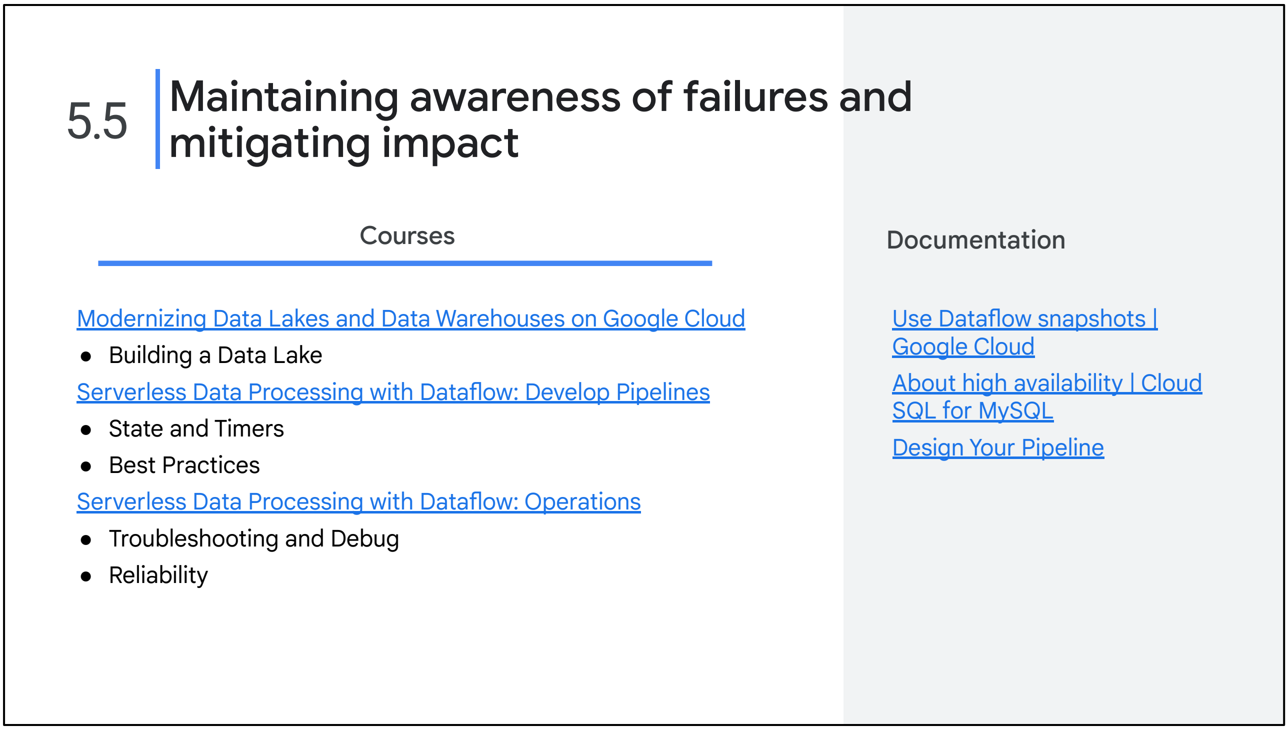
Serverless Data Processing with Dataflow: Develop Pipelines

- State and Timers
- Best Practices

Serverless Data Processing with Dataflow: Operations

- Troubleshooting and Debug
- Reliability

Summary:
The Professional Data Engineer can create Dataflow pipelines to efficiently branch a single transform to output to multiple PCollections. Batch data volumes might be large and streaming data pipelines would be continuous, so re-reading input data is often inefficient or impossible. In such cases, processing can produce multiple outputs.



5.5 Maintaining awareness of failures and mitigating impact

Courses

[Modernizing Data Lakes and Data Warehouses on Google Cloud]

- Building a Data Lake

[Serverless Data Processing with Dataflow: Develop Pipelines]

- State and Timers
- Best Practices

[Serverless Data Processing with Dataflow: Operations]

- Troubleshooting and Debug
- Reliability

Documentation

[Use Dataflow snapshots | Google Cloud]

[About high availability | Cloud SQL for MySQL]

[Design Your Pipeline]

You just reviewed diagnostic questions that addressed considerations related to maintaining awareness of failures and mitigating their impact. These are some courses, skill badges, and links to learn more about the concepts in these questions. They provide a starting point to explore Google-recommended practices.

https://cloud.google.com/dataflow/docs/guides/using-snapshots
https://cloud.google.com/sql/docs/mysql/high-availability
https://beam.apache.org/documentation/pipelines/design-your-pipeline/#a-single-transform-that-produces-multiple-outputs