Security, Policy, and Reliability

Tom Stern

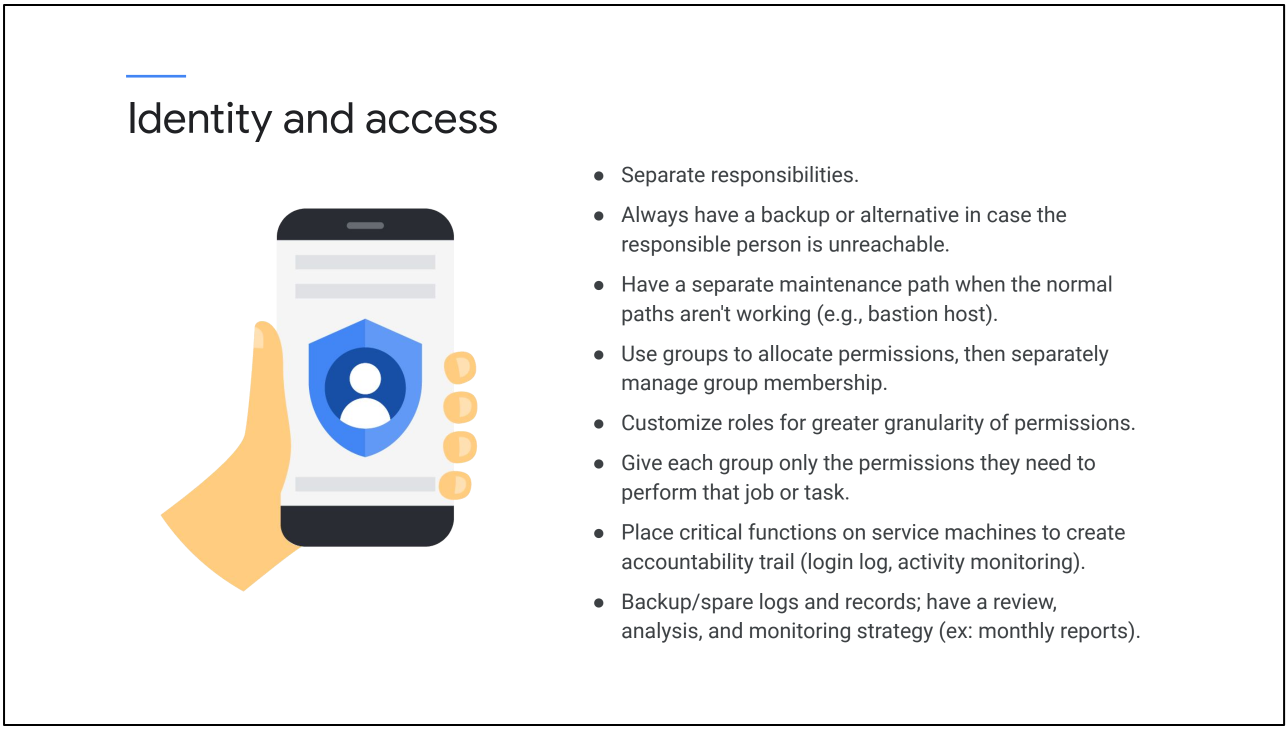
This module covers qualities of the data engineering solution beyond functionality and performance. It addresses security, policy, and reliability.

Designing for Security and Compliance

Designing for security and compliance.

Security is a broad term. It includes privacy, authentication, and authorization, and identity and access management. It could include intrusion detection, attack mitigation, resilience, and recovery. So security really appears across the technologies, and not in just one place.

You need to be aware of the granularity of control for each service. The exercise is, imagine there are two people, one needs accessa and the other must not have access. What is the smallest unit or degree of control the technology supports. Can you distinguish security to an individual field, to a row or record, to columns, to a specific database or entity, or just the kinds of actions that can be performed on the service?



Identity and access

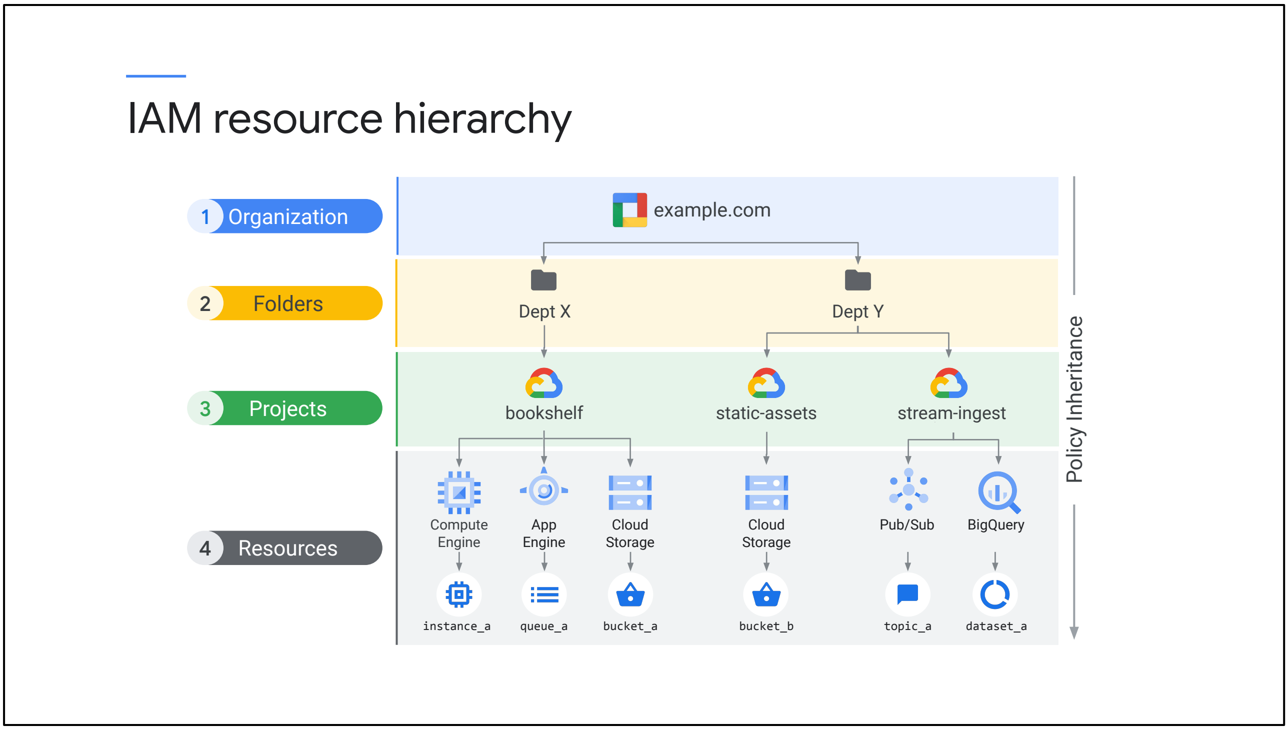
- Separate responsibilities.
- Always have a backup or alternative in case the responsible person is unreachable.
- Have a separate maintenance path when the normal paths aren't working (e.g., bastion host).
- Use groups to allocate permissions, then separately manage group membership.
- Customize roles for greater granularity of permissions.
- Give each group only the permissions they need to perform that job or task.
- Place critical functions on service machines to create accountability trail (login log, activity monitoring).
- Backup/spare logs and records; have a review, analysis, and monitoring strategy (ex: monthly reports).

Commit a security checklist to memory. Sometimes just running down a list will rapidly identify a solution.

A key concept is to assign roles to groups, and use group membership to grant permissions to individuals.

How will the service be monitored or reported, and how often will these items be reviewed?

Finally, you need to know what kinds of logs and reporting are available from each technology.



IAM resource hierarchy

1 Organization

2 Folders

3 Projects

4 Resources

example.com

Dept X

Dept Y

bookshelf

static-assets

stream-ingest

Compute Engine

App Engine

Cloud Storage

Cloud Storage

Pub/Sub

BigQuery

instance_a

queue_a

bucket_a

bucket_b

topic_a

dataset_a

Policy Inheritance

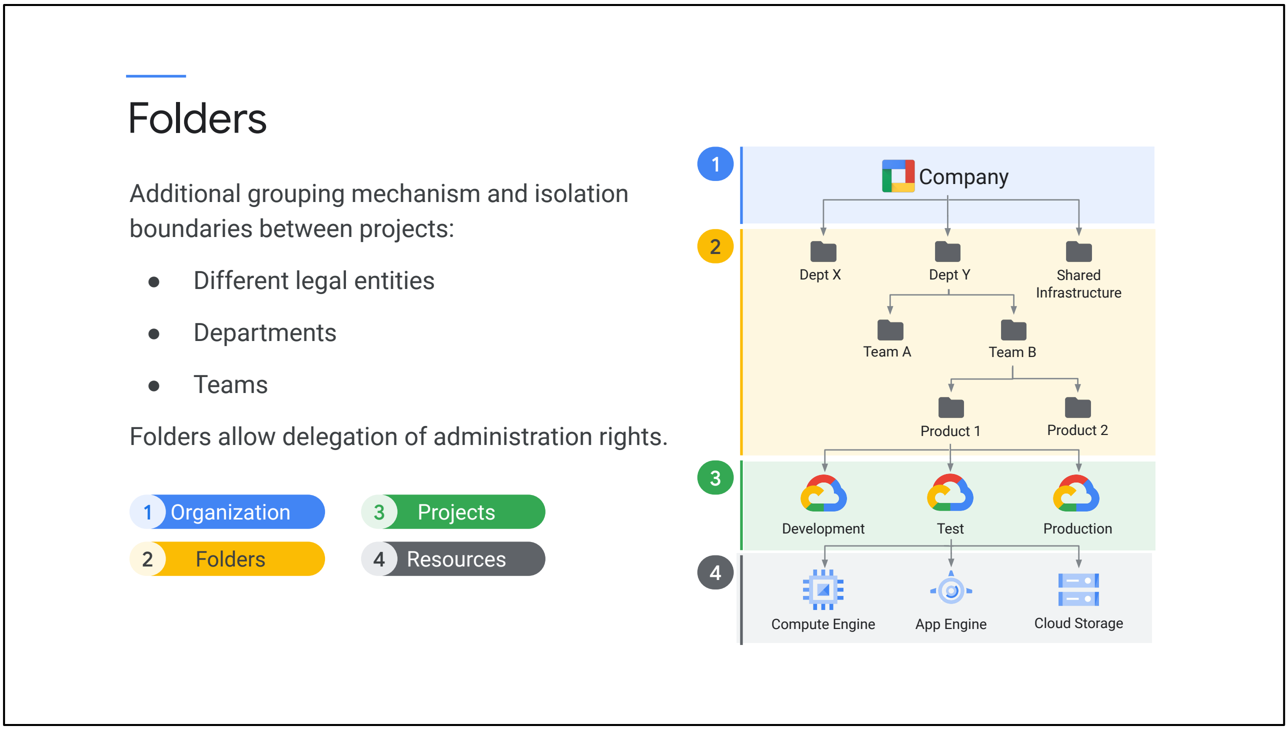
A policy is set on a resource, and each policy contains a set of roles and role members.

Resources inherit policies from parents. So a policy can be set on a resource, for example a service. And another policy can be set on a parent, such as a project that contains that service.

The final policy is the union of the parent policy and the resource policy.

What happens when these two policies are in conflict? What if the policy on the resource only gives access to a single Cloud Storage bucket and restricts access to all other buckets. However, at the project level, a rule exists that grants access to all buckets in the Project? Which rule wins? The more restrictive rule on the resource, or the more general rule on the project?

If the parent policy is less restrictive, it overrides a more restrictive resource policy. So in this case the Project policy wins.



Folders

Additional grouping mechanism and isolation boundaries between projects:

- Different legal entities
- Departments
- Teams

Folders allow delegation of administration rights.

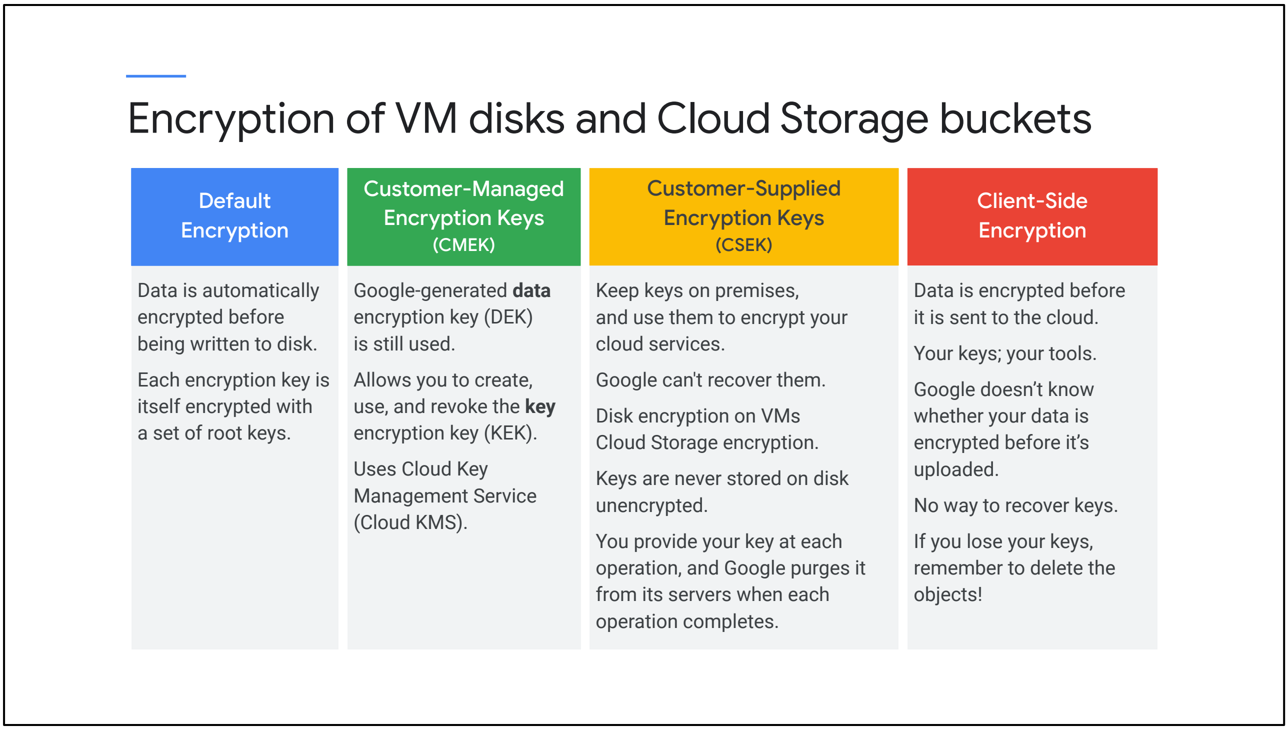
1 Organization

2 Folders

3 Projects

4 Resources

Folders map well to organization structure. It is a way to isolate organizations or users or products while still having them share billing and corporate resources.



Encryption of VM disks and Cloud Storage buckets

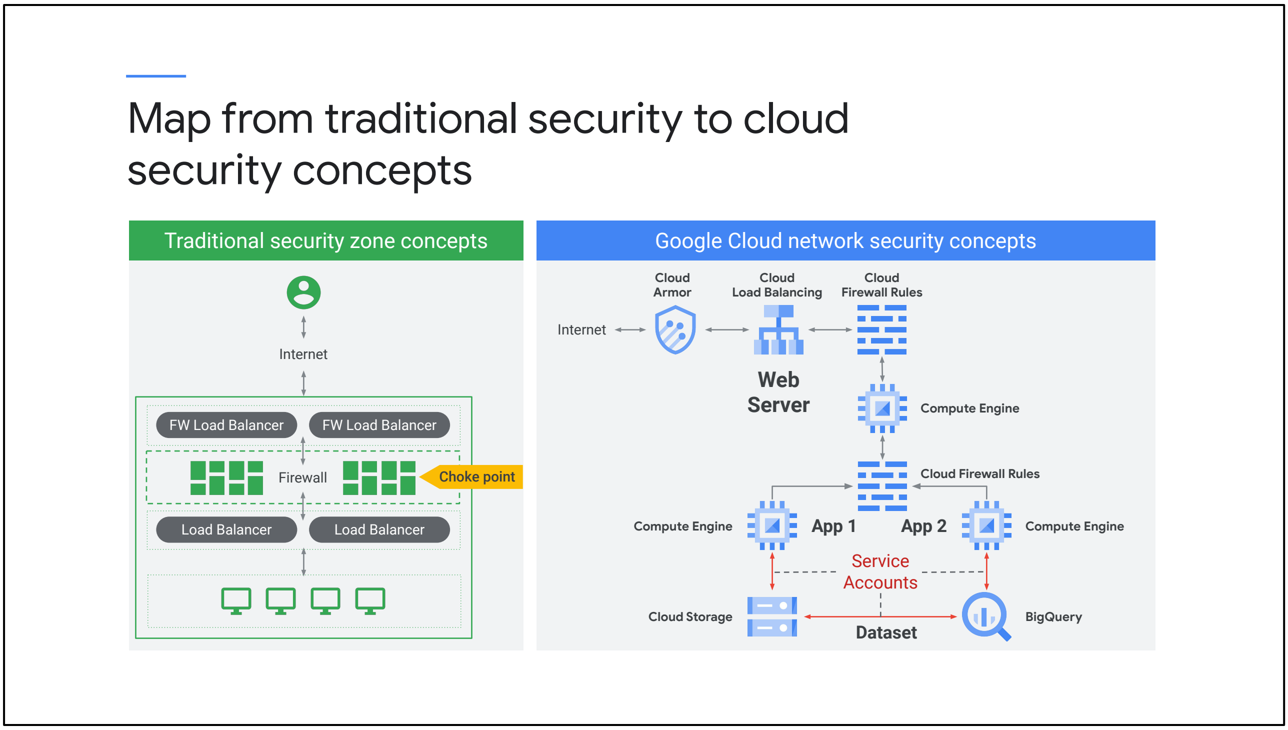
Default Encryption	Customer-Managed Encryption Keys (CMEK)	Customer-Supplied Encryption Keys (CSEK)	Client-Side Encryption
Data is automatically encrypted before being written to disk. Each encryption key is itself encrypted with a set of root keys.	Google-generated **data** encryption key (DEK) is still used. Allows you to create, use, and revoke the **key** encryption key (KEK). Uses Cloud Key Management Service (Cloud KMS).	Keep keys on premises, and use them to encrypt your cloud services. Google can't recover them. Disk encryption on VMs Cloud Storage encryption. Keys are never stored on disk unencrypted. You provide your key at each operation, and Google purges it from its servers when each operation completes.	Data is encrypted before it is sent to the cloud. Your keys; your tools. Google doesn't know whether your data is encrypted before it's uploaded. No way to recover keys. If you lose your keys, remember to delete the objects!

There are many encryption options for data at rest and in storage. Default encryption at rest uses the Key Management System (KMS) to generate KEKs which are key encryption keys and DEKs the data encryption keys.

When you use Dataproc, cluster and job data is stored on Persistent Disks (PDs) associated with the Compute Engine VMs in your cluster and in a Cloud Storage bucket. This PD and bucket data is encrypted using a Google-generated data encryption key (DEK) and key encryption key (KEK).

Customer Managed Encryption Keys, CMEK, is a feature that allows you to create, use, and revoke the key encryption key (KEK). Google still controls the data encryption key (DEK).

Client-Side Encryption simply means that you encrypt the data or file before you upload it to the cloud.



Map from traditional security to cloud security concepts

Traditional security zone concepts

Internet

FW Load Balancer

FW Load Balancer

Firewall

Choke point

Load Balancer

Load Balancer

Google Cloud network security concepts

Cloud Armor

Cloud Load Balancing

Cloud Firewall Rules

Internet

Web Server

Compute Engine

Cloud Firewall Rules

Compute Engine

App 1

App 2

Compute Engine

Service Accounts

Cloud Storage

Dataset

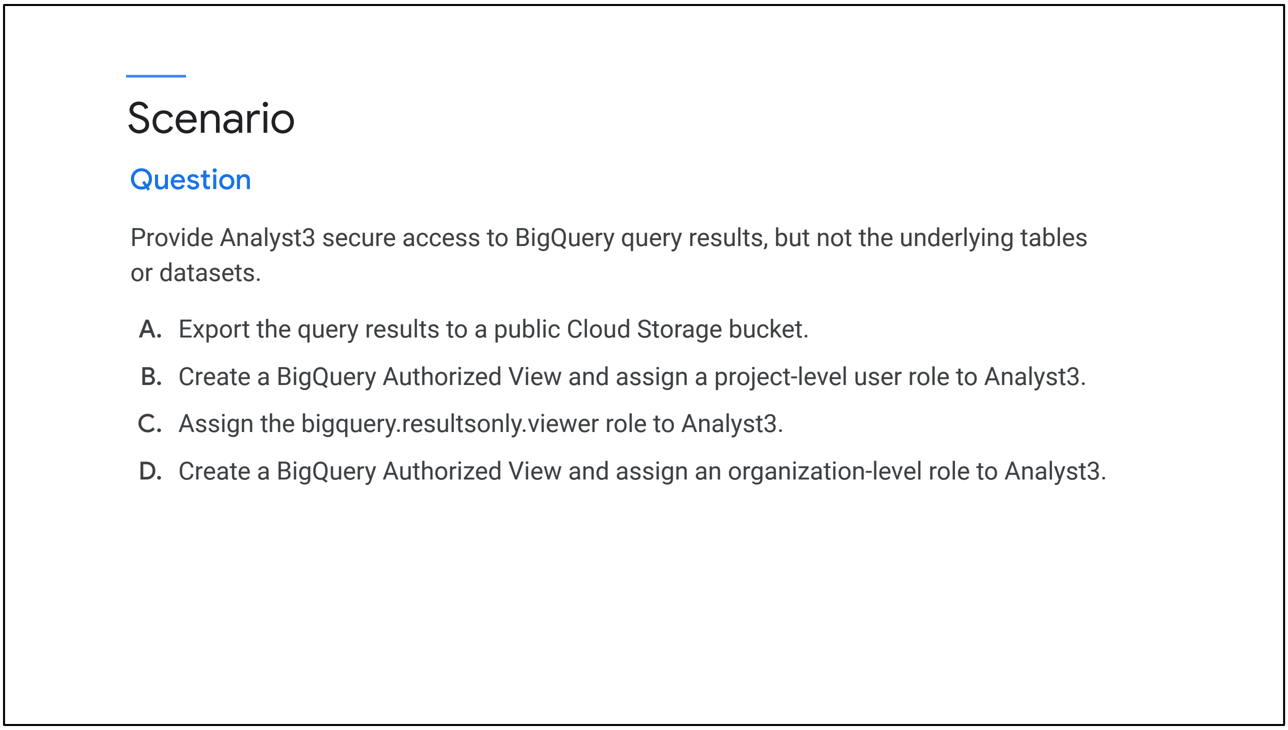
BigQuery

Key concepts: Cloud Armor, Cloud Load Balancing, Cloud Firewall Rules, Service Accounts, separation into front-end and back-end, isolation of resources using separate service accounts between services.

Because of pervasive availability of firewall rules, you don't have to install a router in the network at a particular location to get firewall protection. That means you can layer the firewalls as shown in this example.

Because if pervasive support for Service accounts, you can "lock down" connections between components.

When faced with a security question on an exam (or in practice), determine which of the specific technologies/services is being discussed (authentication, encryption) for example. Then determine exactly what the goals are for sufficient security. Is it deterrence? Is it meeting a standard for compliance? Is the goal to eliminate a particular risk or vulnerability? This will help you define the scope of a solution, whether on an exam or in application.



Scenario

Question

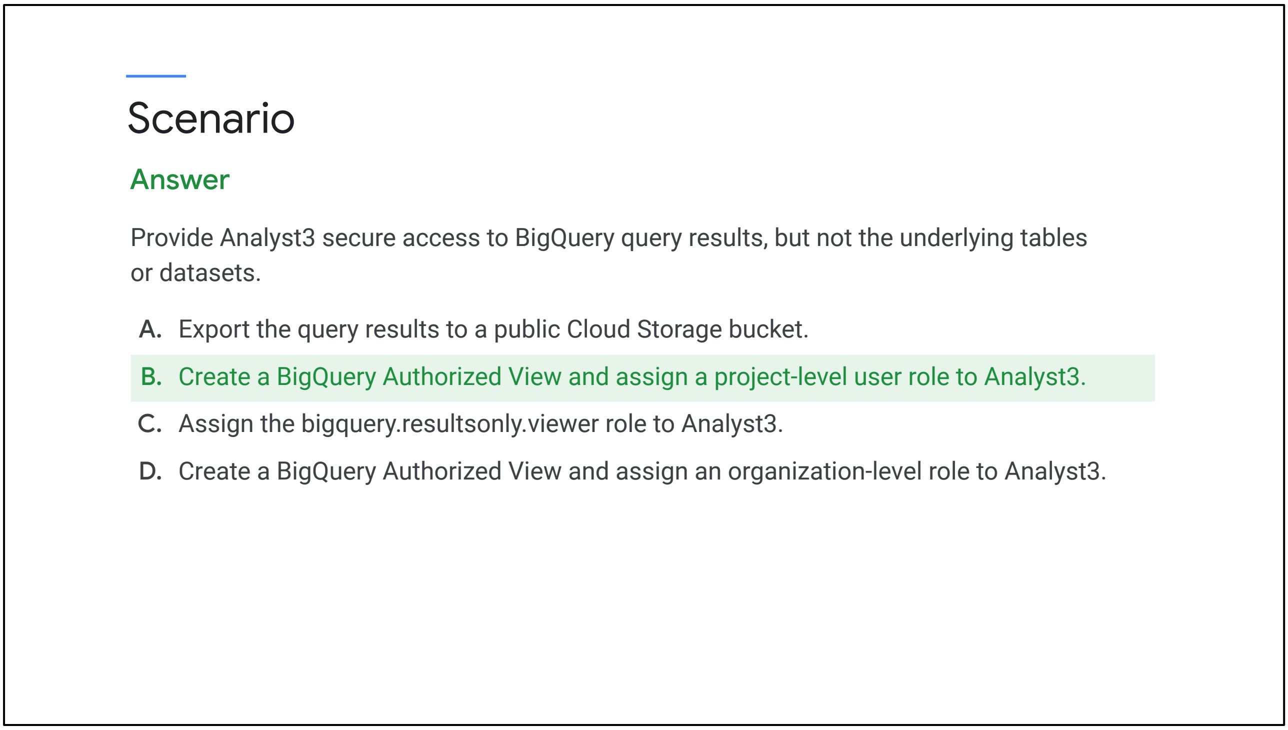
Provide Analyst3 secure access to BigQuery query results, but not the underlying tables or datasets.

A. Export the query results to a public Cloud Storage bucket.
B. Create a BigQuery Authorized View and assign a project-level user role to Analyst3.
C. Assign the bigquery.resultsonly.viewer role to Analyst3.
D. Create a BigQuery Authorized View and assign an organization-level role to Analyst3.

Let's try another sample exam question.

Provide Analyst3 secure access to BigQuery query results, but not the underlying tables or datasets.

A. Export the query results to a public Cloud Storage bucket.
B. Create a BigQuery Authorized View and assign a project-level user role to Analyst3
C. Assign the bigquery.resultsonly.viewer role to Analyst3
D. Create a BigQuery Authorized View and assign an organization-level role to Analyst3.



Scenario

Answer

Provide Analyst3 secure access to BigQuery query results, but not the underlying tables or datasets.

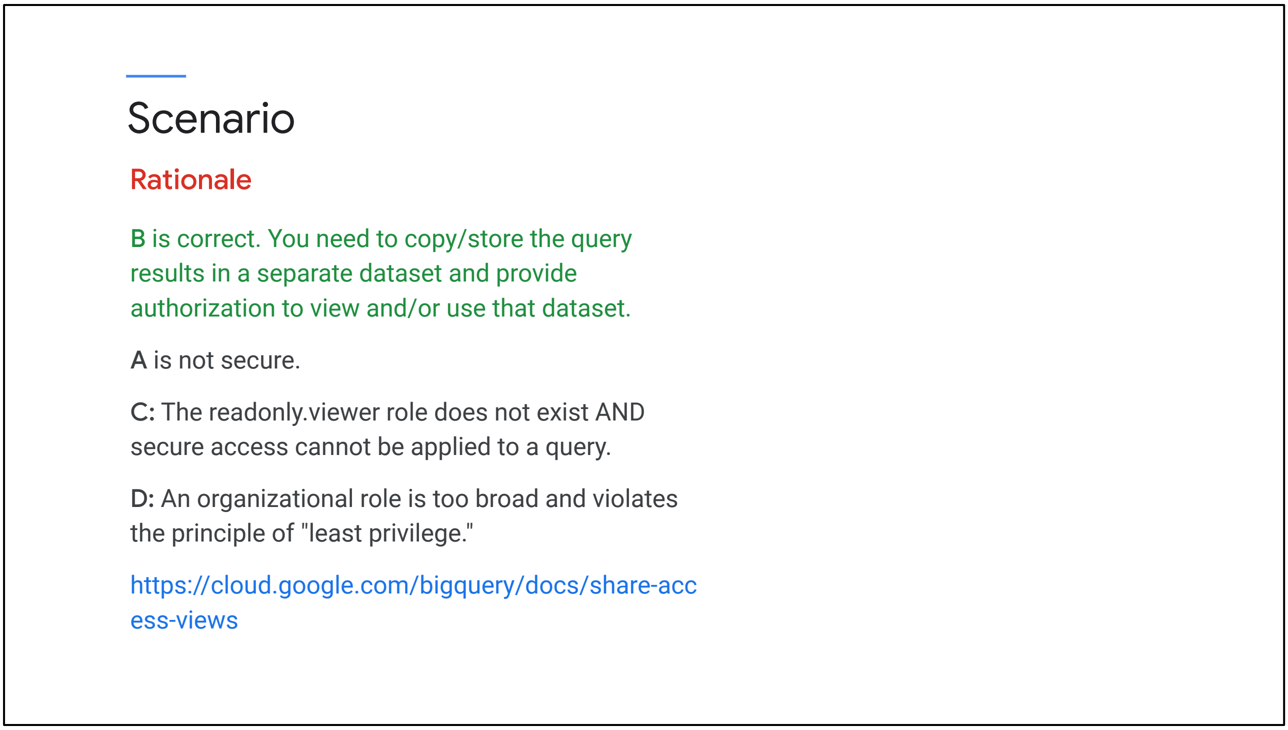
A. Export the query results to a public Cloud Storage bucket.

B. Create a BigQuery Authorized View and assign a project-level user role to Analyst3.

C. Assign the bigquery.resultsonly.viewer role to Analyst3.

D. Create a BigQuery Authorized View and assign an organization-level role to Analyst3.

The answer is B.
Create a BigQuery Authorized View and assign a project-level user role to Analyst3.



Scenario

Rationale

B is correct. You need to copy/store the query results in a separate dataset and provide authorization to view and/or use that dataset.

A is not secure.

C: The readonly.viewer role does not exist AND secure access cannot be applied to a query.

D: An organizational role is too broad and violates the principle of "least privilege."

https://cloud.google.com/bigquery/docs/share-access-views

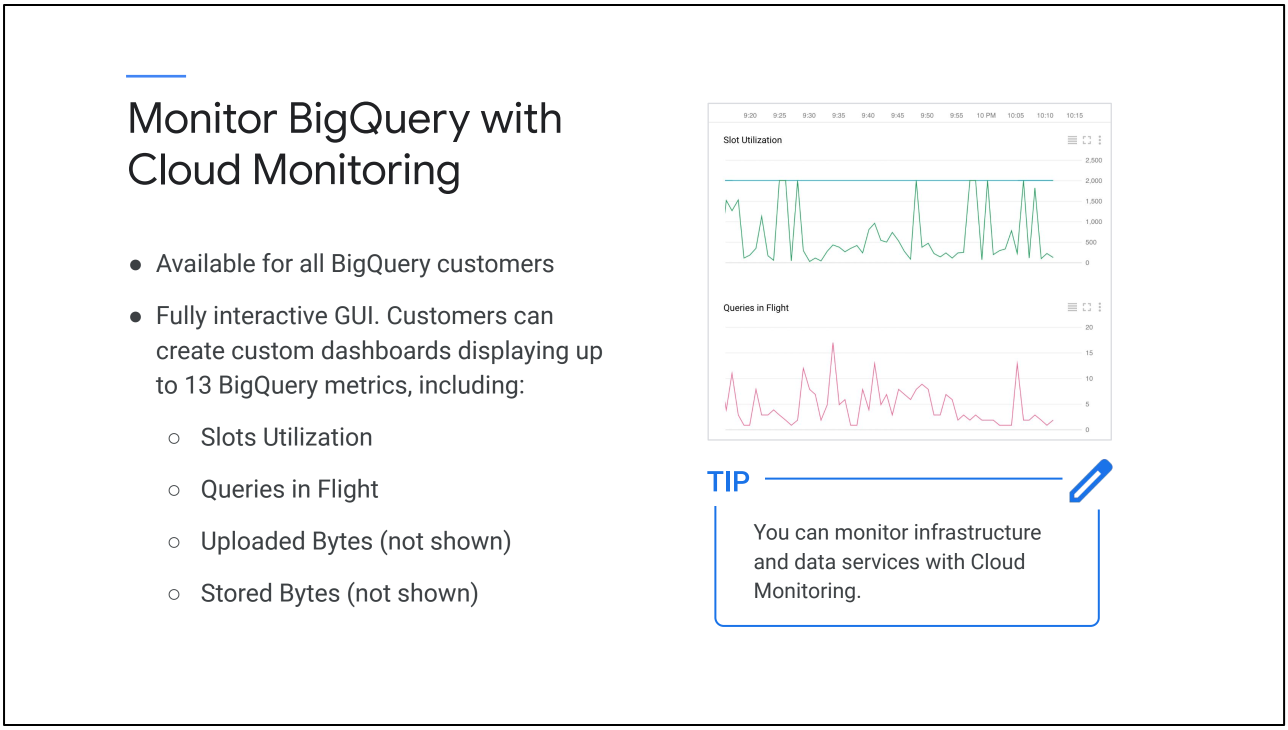
You need to copy/store the query results in a separate dataset, and provide authorization to view and/or use that dataset.

The other solutions are not secure.

Performing Quality Control

Tom Stern

The next section of the exam guide refers to "Performing quality control". This is part of the reliability section of the exam guide. So it is referring to how you can monitor the quality of your solution.



Monitor BigQuery with Cloud Monitoring

- Available for all BigQuery customers
- Fully interactive GUI. Customers can create custom dashboards displaying up to 13 BigQuery metrics, including:
  - Slots Utilization
  - Queries in Flight
  - Uploaded Bytes (not shown)
  - Stored Bytes (not shown)

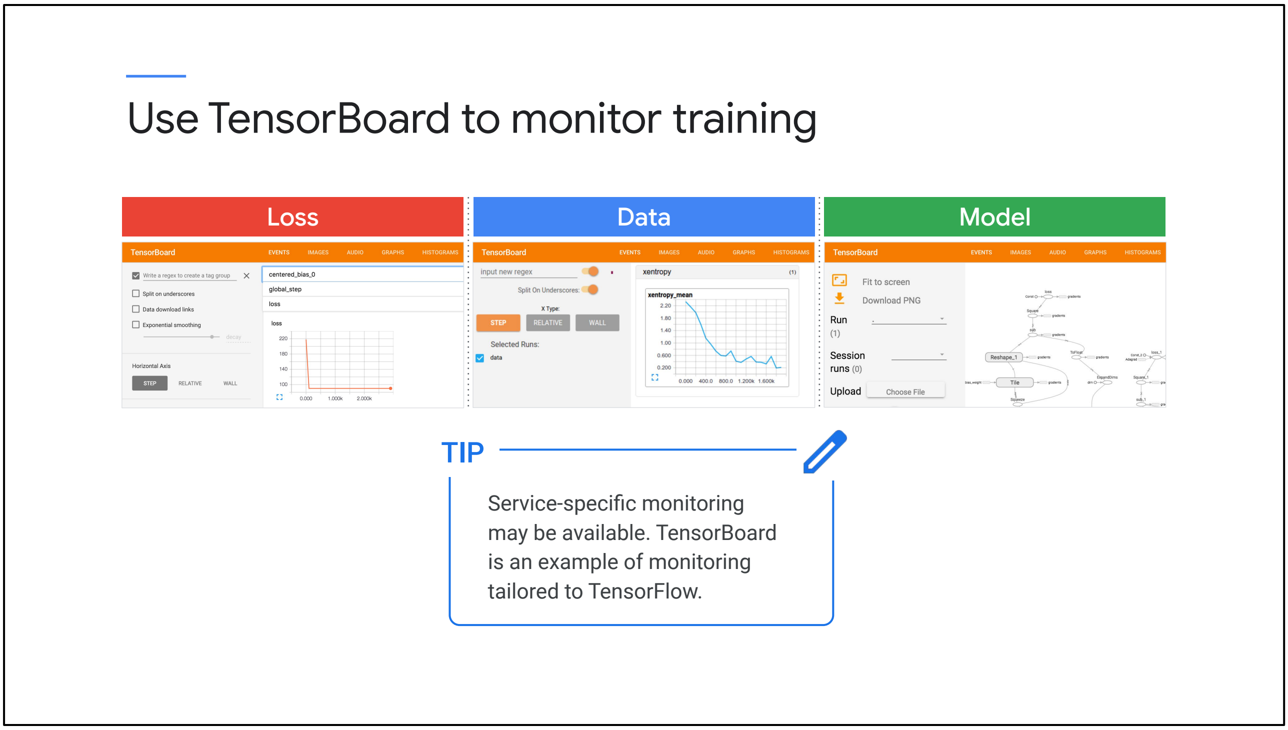
TIP

You can monitor infrastructure and data services with Cloud Monitoring.

Integrated monitoring across services can simplify the activity of monitoring a solution. You can get graphs for multiple values in a single dashboard. It is possible to surface application values as custom metrics in Cloud Monitoring.

These charts show Slot Utilization, Slots available and queries in flight for a 1 hr period of BigQuery.

The exam tip here is that you can monitor infrastructure and data services with Cloud Monitoring.



Use TensorBoard to monitor training

TIP

Service-specific monitoring may be available. TensorBoard is an example of monitoring tailored to TensorFlow.

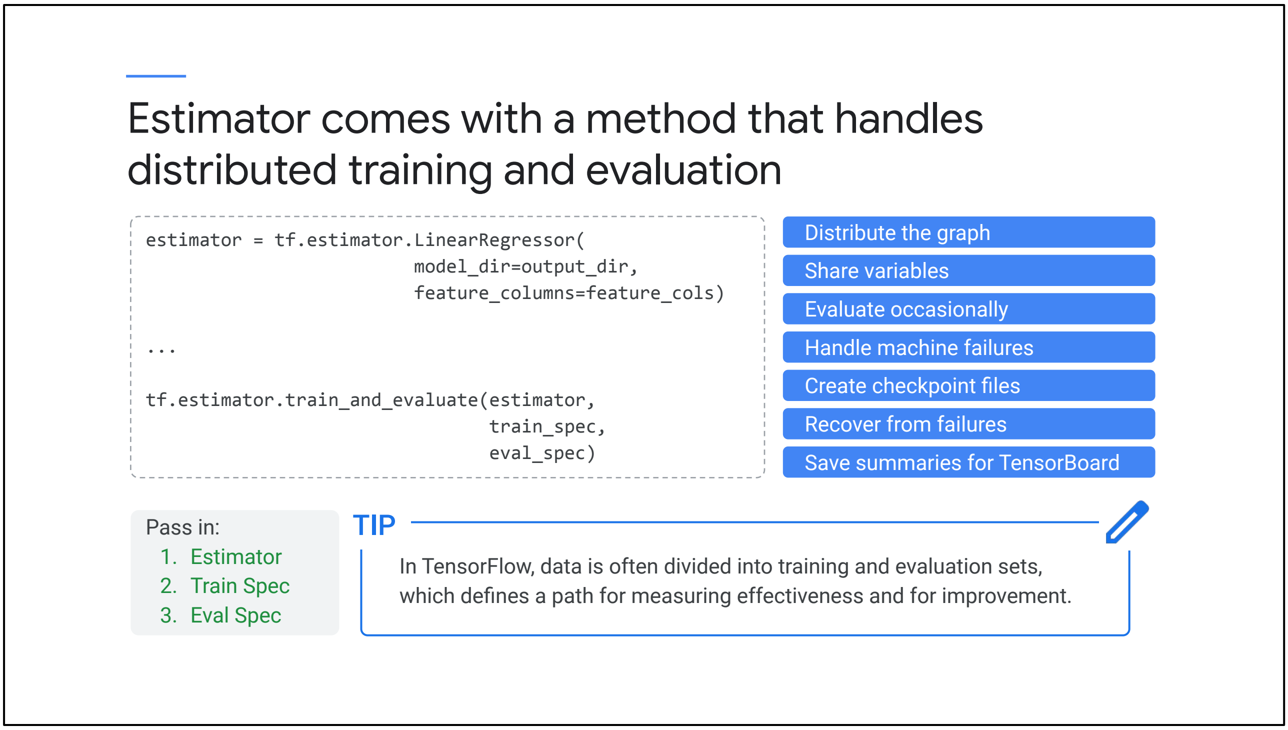
TensorBoard is a collection of visualization tools designed specifically to help you visualize TensorFlow.

- TensorFlow graph
- Plot quantitative metrics
- Pass and graph additional data

Events at THE top left shows “loss”.

The other graphs show the linear model graph as built by TensorFlow.

The exam tip here is that Service-specific monitoring may be available. TensorBoard is an example of monitoring tailored to TensorFlow.



Estimator comes with a method that handles distributed training and evaluation

```
estimator = tf.estimator.LinearRegressor(
                         model_dir=output_dir,
                         feature_columns=feature_cols)

...

tf.estimator.train_and_evaluate(estimator,
                                train_spec,
                                eval_spec)
```

- Distribute the graph
- Share variables
- Evaluate occasionally
- Handle machine failures
- Create checkpoint files
- Recover from failures
- Save summaries for TensorBoard

Pass in:
1. Estimator
2. Train Spec
3. Eval Spec

TIP

In TensorFlow, data is often divided into training and evaluation sets, which defines a path for measuring effectiveness and for improvement.

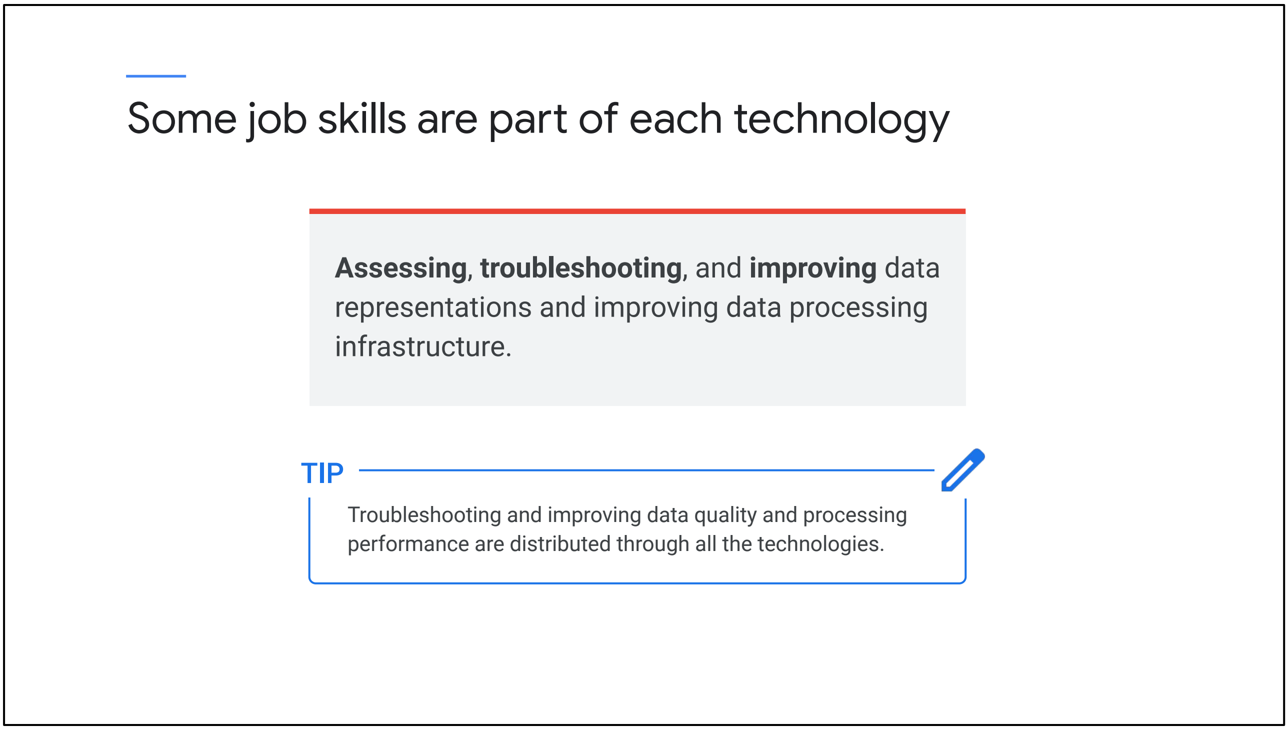
Here are some tips for reliability with Machine Learning.

There are a number of things you can do to improve reliability. For example, you can recognize machine failures, create checkpoint files, and recover from failures.

You can also control how often evaluation occurs to make the overall process more efficient.

The tip shown is that "In TensorFlow data is often divided into training and evaluation sets, defining a path for measuring effectiveness and for improvement."

So the overall exam tip is that there might be quality processes or reliability processes built in to the technology, such as this demonstrates.



Some job skills are part of each technology

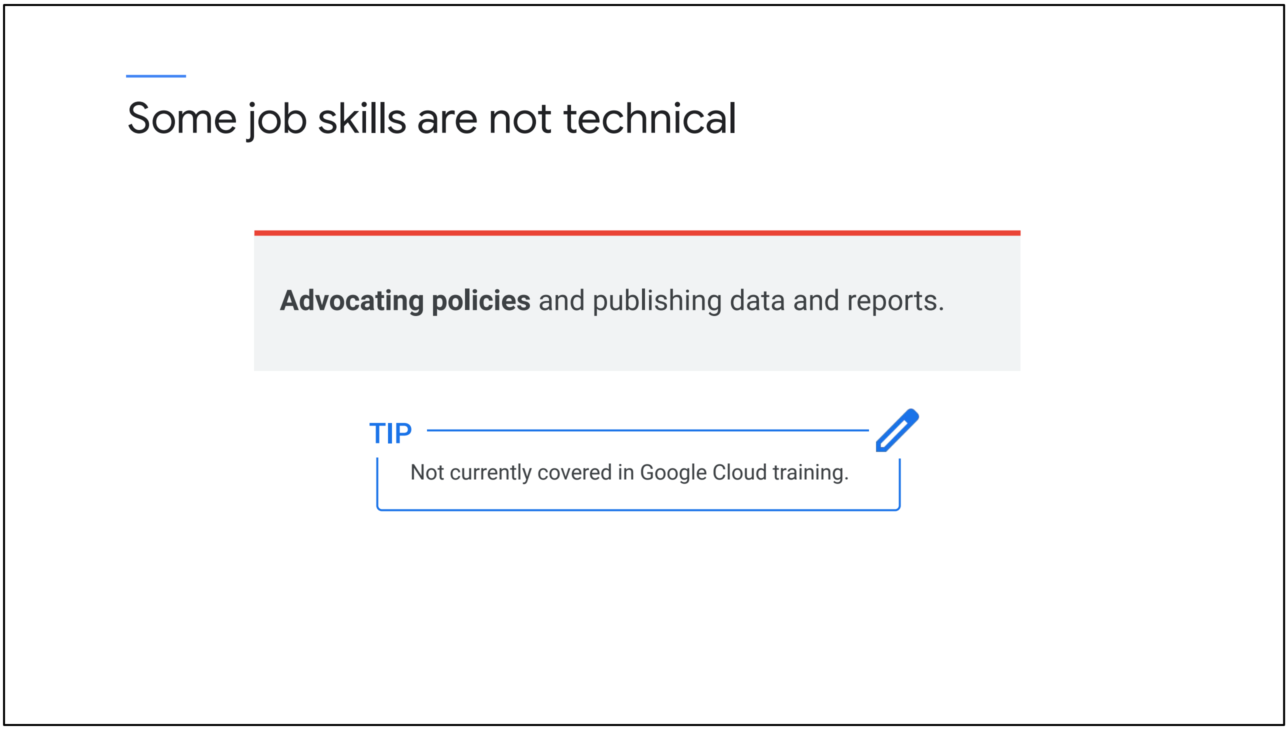
Assessing, **troubleshooting**, and **improving** data representations and improving data processing infrastructure.

TIP

Troubleshooting and improving data quality and processing performance are distributed through all the technologies.

The exam tip here is that Troubleshooting and improving data quality and processing performance is distributed through all the technologies. It would be a good idea to make sure you know the main troubleshooting methods for each data engineering technology and service.

Security and troubleshooting are the lateral subjects that cut across all technologies. You need to look for them in each technology area and dig into the documentation as needed.



Some job skills are not technical

Advocating policies and publishing data and reports.

TIP

Not currently covered in Google Cloud training.

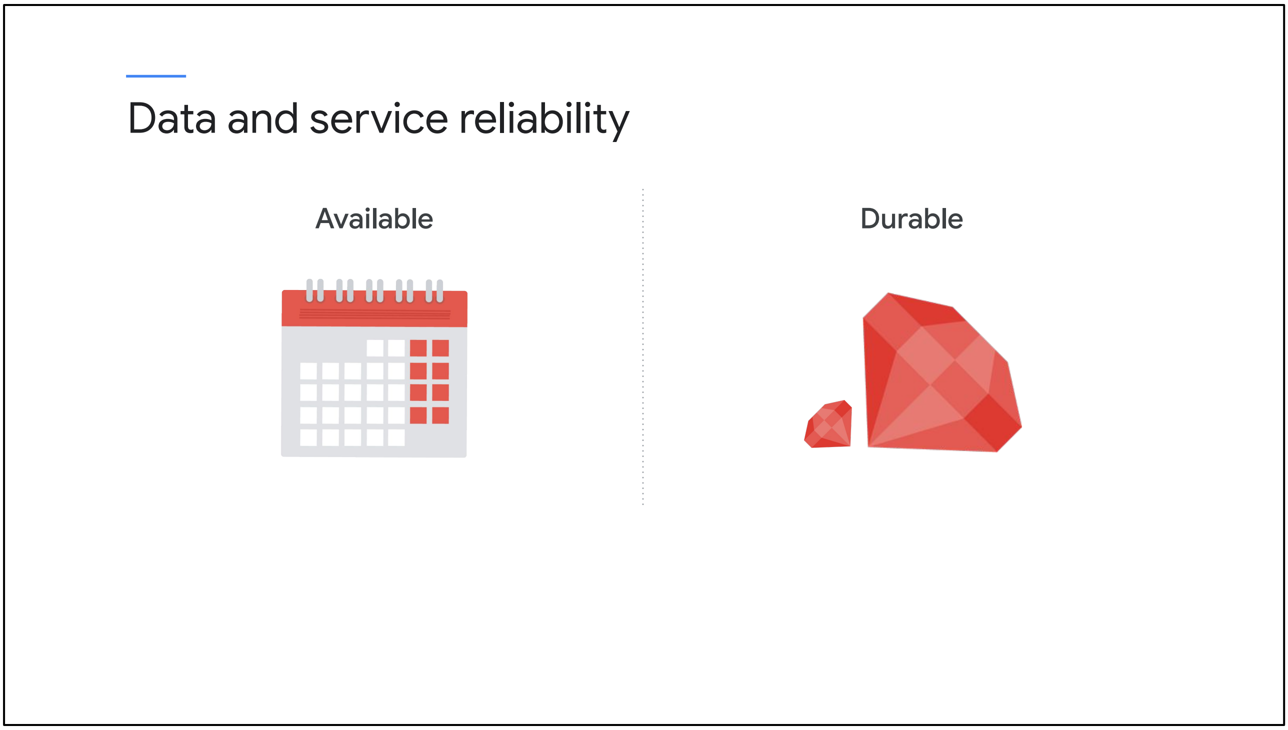
The training *does* cover the mechanics of generating and reports. But not explicitly how to present and advocate for policies.

This subject is in the exam outline. It is a general Job skill rather than a technical skill and not specifically covered in Google technical training. Nevertheless, it IS part of the job and could be on the exam. So this is one of the items I suggest you make sure you know, even though it is not in our training.

Ensuring Reliability

Tom Stern

Reliable means that the service produces consistent outputs and operates as expected. If we were to quantify it, it would be a measure of how long the service performs its intended function.



Data and service reliability

Available

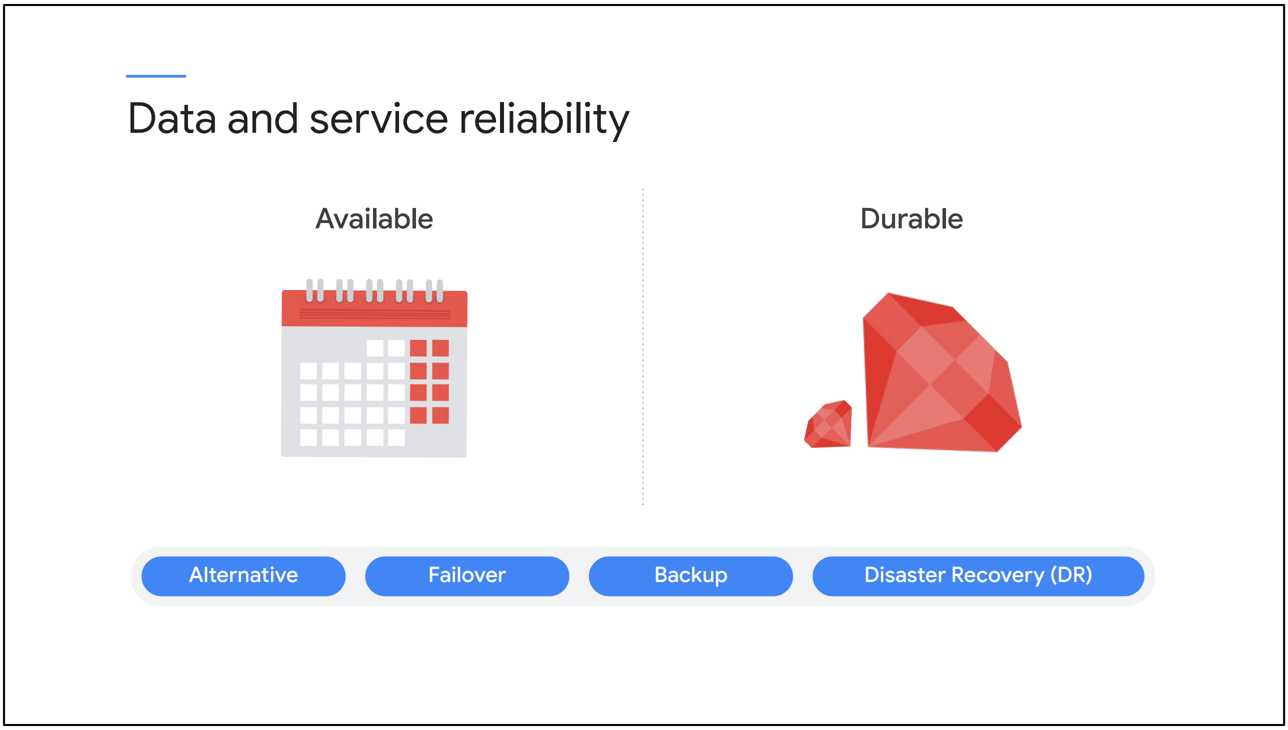
Durable

Available and durable are real-world values and usually are not 100%.

Available means that the service is accessible on demand. A measure of the % of time the item is in an operable state.

Durable has to do with data loss. It means the data does not disappear, and information is not lost over time. More accurately, it is a measure of the rate at which data is lost.

These qualities are related. If a service fails -- has an outage -- then it is not producing reliable results during that period.



Data and service reliability

Available

Durable

Alternative

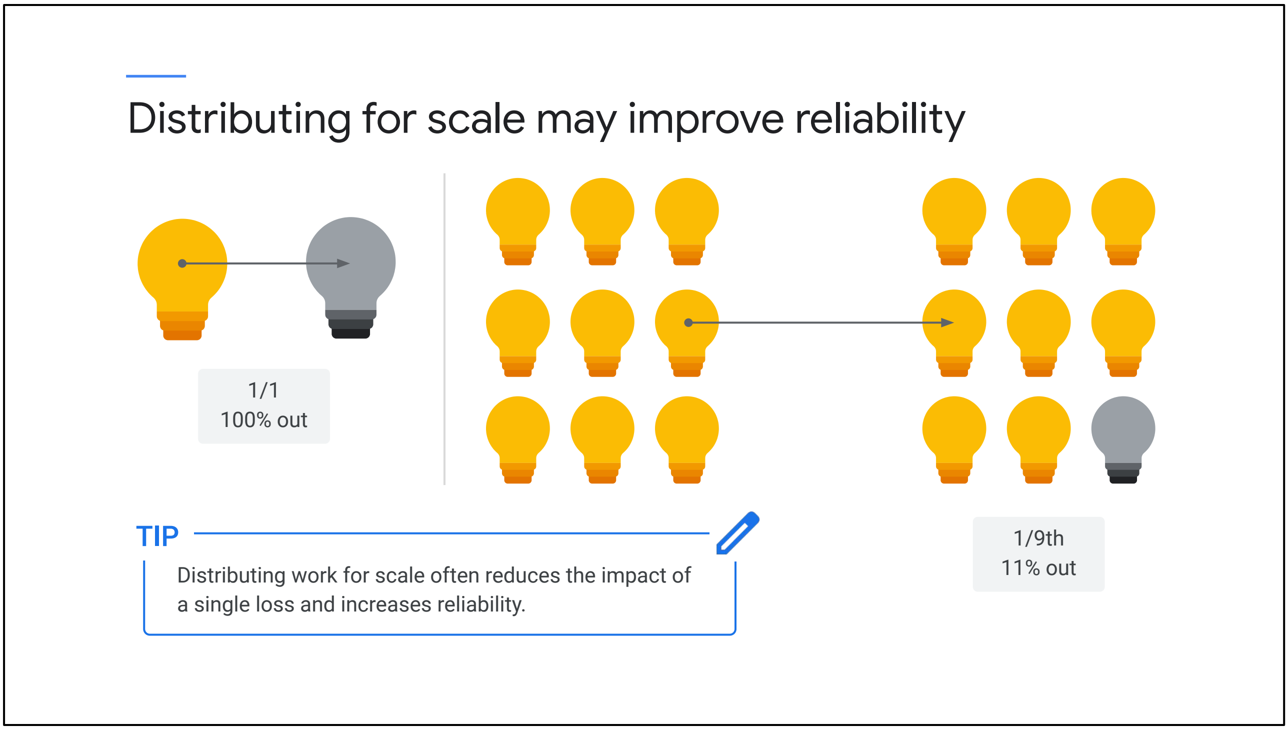
Failover

Backup

Disaster Recovery (DR)

An alternate service or failover might bring the service back online and make it available again. Typically, an outage that causes a loss of data requires more time to recover. If it is recovered from backup or from a disaster recovery plan. But notice that if you have an alternate service such as a copy that can be rapidly turned on, there might be little or no loss of data or time to recover.

So the important thing to consider is what are the business requirements to recover from different kinds of problems, and how much time is allowed for each kind of recovery. For example, disaster recovery of a week might be acceptable for flood damage to a store front. On the other hand, loss of a financial transaction might be completely unacceptable so the transaction itself needs to be atomic, backed up, and redundant.



Distributing for scale may improve reliability

1/1
100% out

1/9th
11% out

TIP

Distributing work for scale often reduces the impact of a single loss and increases reliability.

Simply scaling up may improve reliability. If the solution is designed to be fault tolerant, increasing scale might improve reliability.

In this example, if the service is running on one node, and that node goes out, the service is 100% down. On the other hand, if the service has scaled up and is running on nine nodes, and one goes out, the service is only 11% down.

Google Cloud