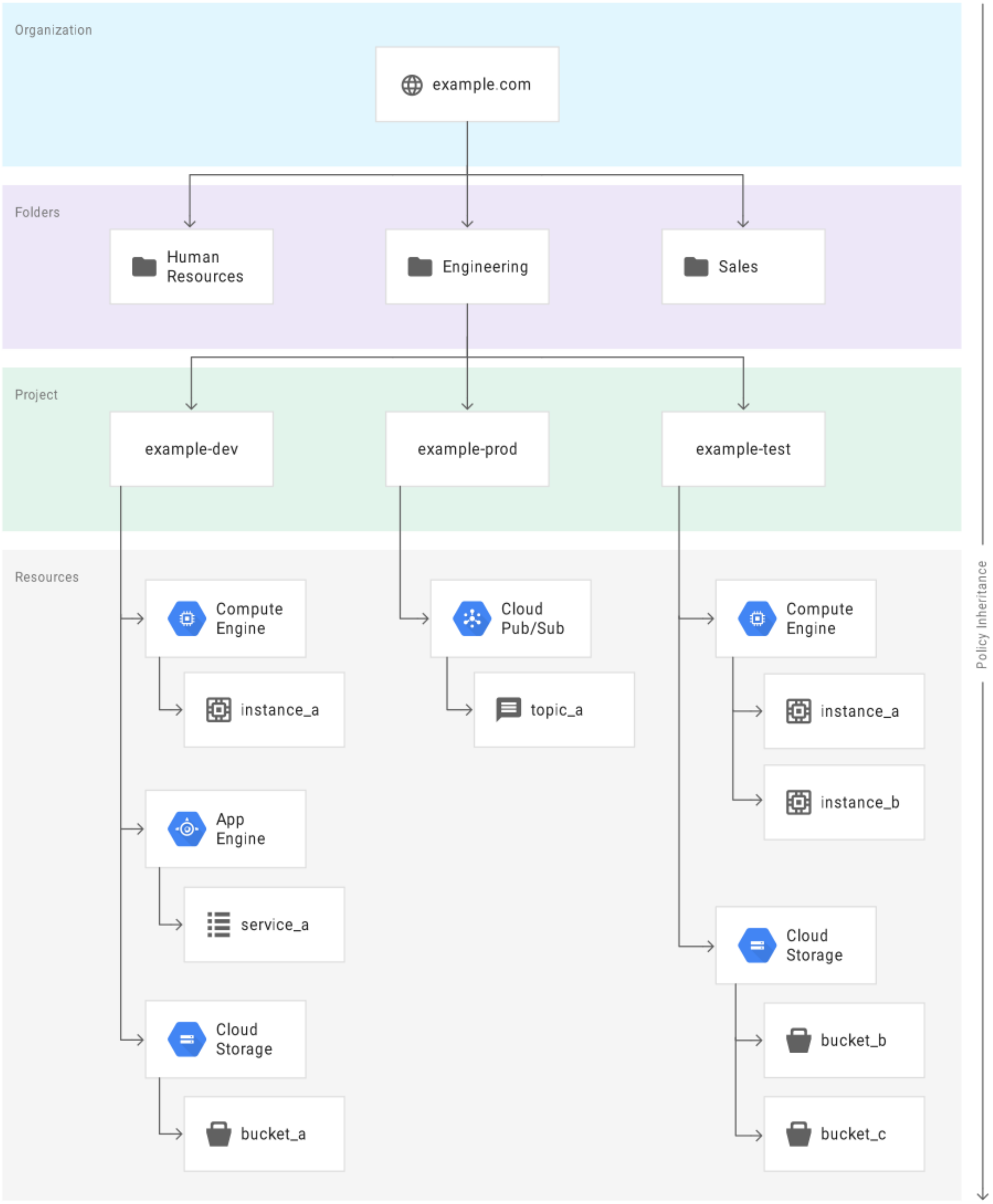
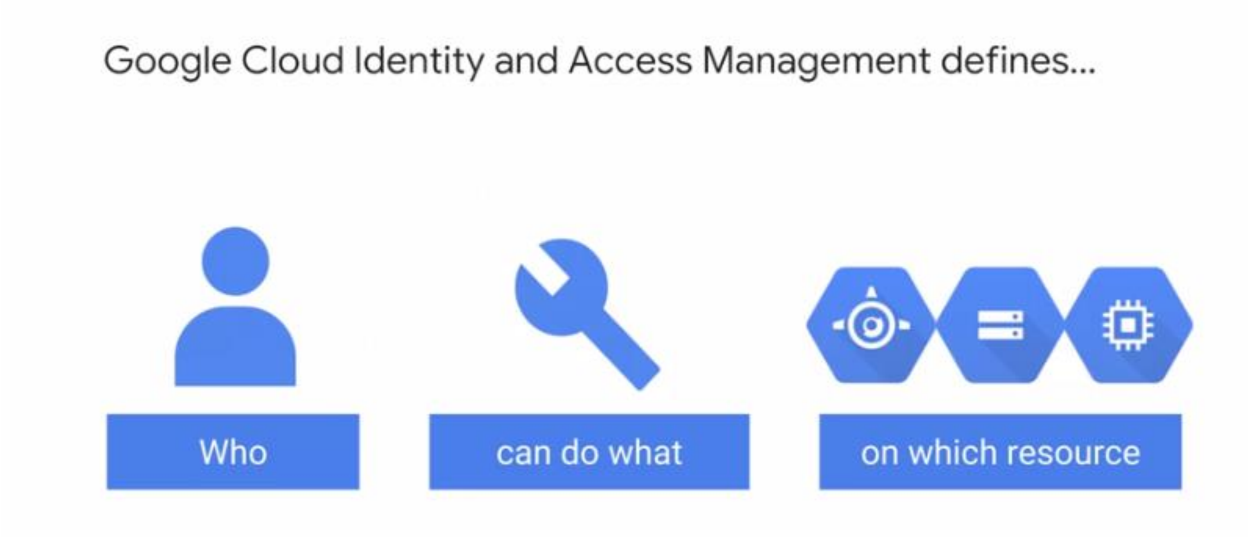
Google Cloud Platform Resource Hierarchy

Courtesy:

https://cloud.google.com/resource-manager/docs/cloud-platform-resource-hierarchy



IAM – Identity Access Management



Google Cloud Identity and Access Management defines...

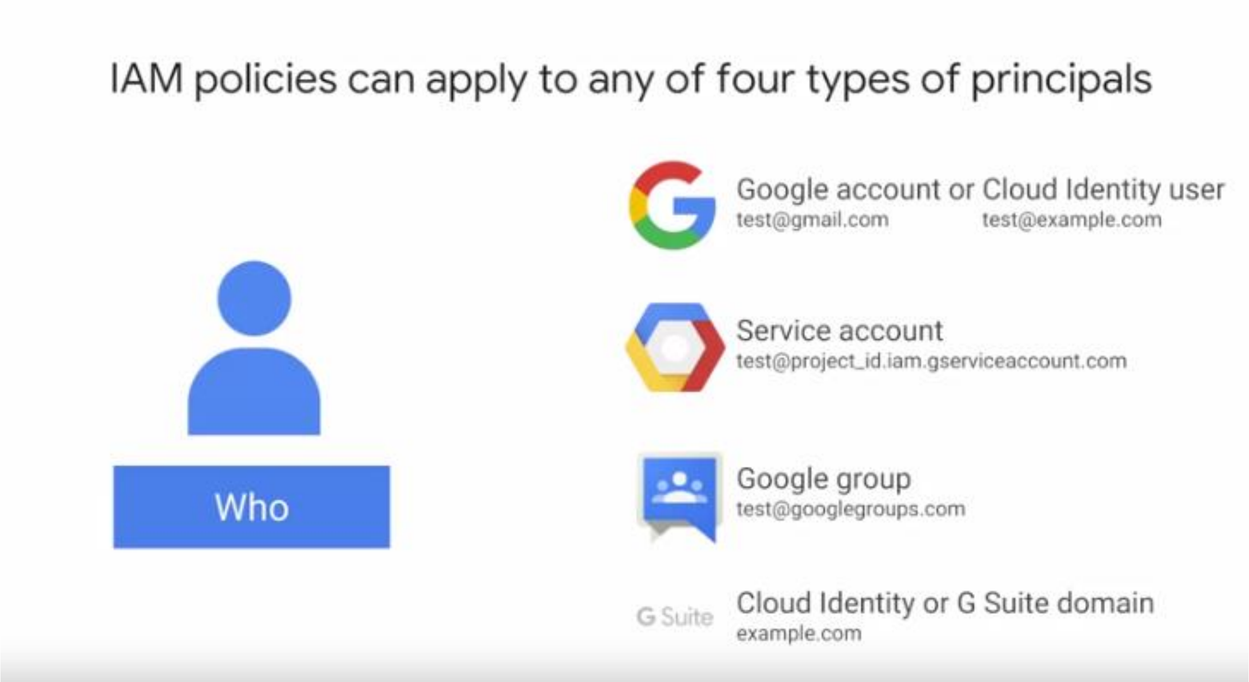
Who

can do what

on which resource

Who has Access (itsbala9701@gmail.com)- Balaji SK

- Google Account/ Cloud Identity User
- Google group Account - Group Activity
- A Service Account
- A G-Suite domain
- Cloud Identity Domain



IAM policies can apply to any of four types of principals

Who

Google account or Cloud Identity user
test@gmail.com test@example.com

Service account
test@project_id.iam.gserviceaccount.com

Google group
test@googlegroups.com

Cloud Identity or G Suite domain
example.com

Can do what

Can perform what activity can be defined by Roles

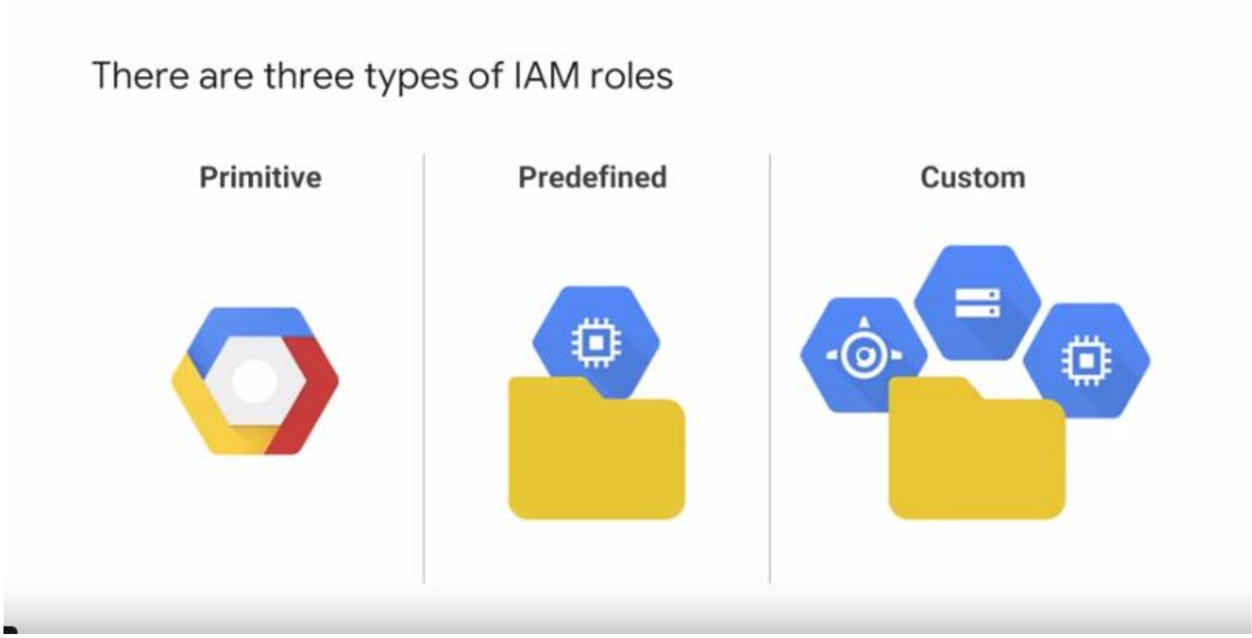
For Eg: Virtual Machine

Create / Start / Stop / delete the VM

A Role in-turn is a Collection of Permissions

Roles

- **Primitive**
- **Pre-defined**
- **Custom**



There are three types of IAM roles

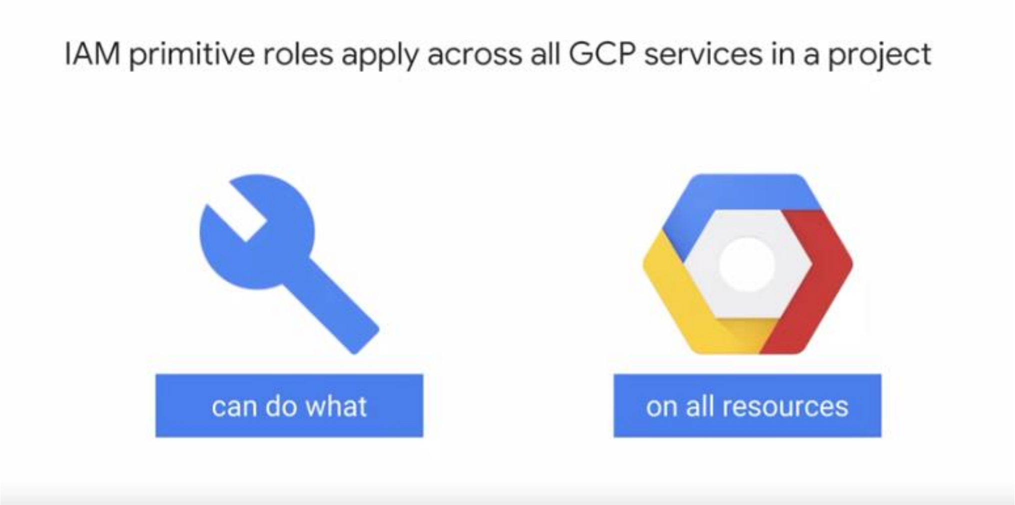
Primitive

Predefined

Custom

Coarse Grained Role -

Primitive Roles:



IAM primitive roles apply across all GCP services in a project

can do what

on all resources

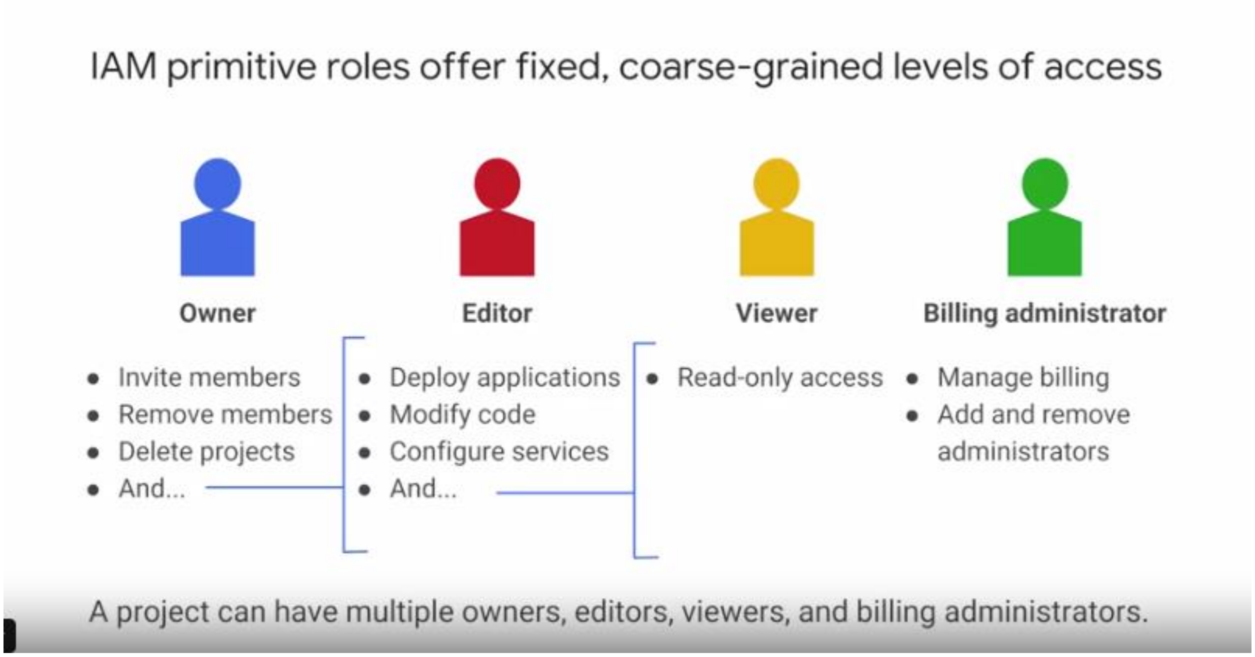
Primitive Roles are applied across all Services in a project

Owner

Editor

Viewer

Fine Grained Roles – Predefined



IAM primitive roles offer fixed, coarse-grained levels of access

Owner

- Invite members
- Remove members
- Delete projects
- And...

Editor

- Deploy applications
- Modify code
- Configure services
- And...

Viewer

- Read-only access

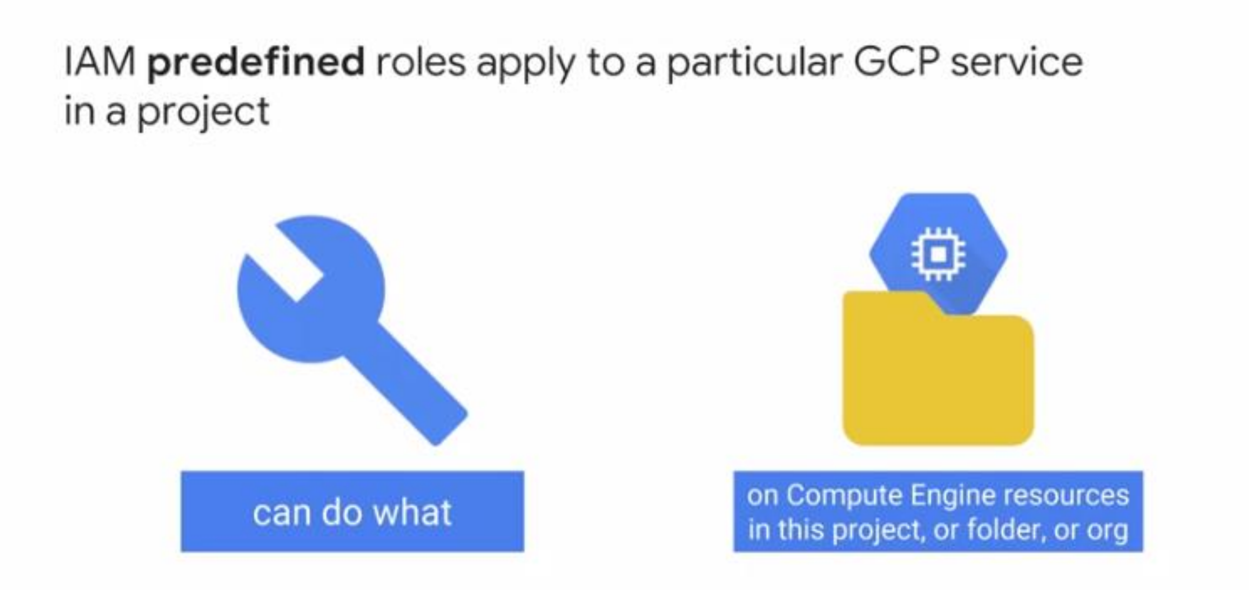
Billing administrator

- Manage billing
- Add and remove administrators

A project can have multiple owners, editors, viewers, and billing administrators.

This activity can be performed on which Resource ..

Pre-defined Roles are applied to particular service :



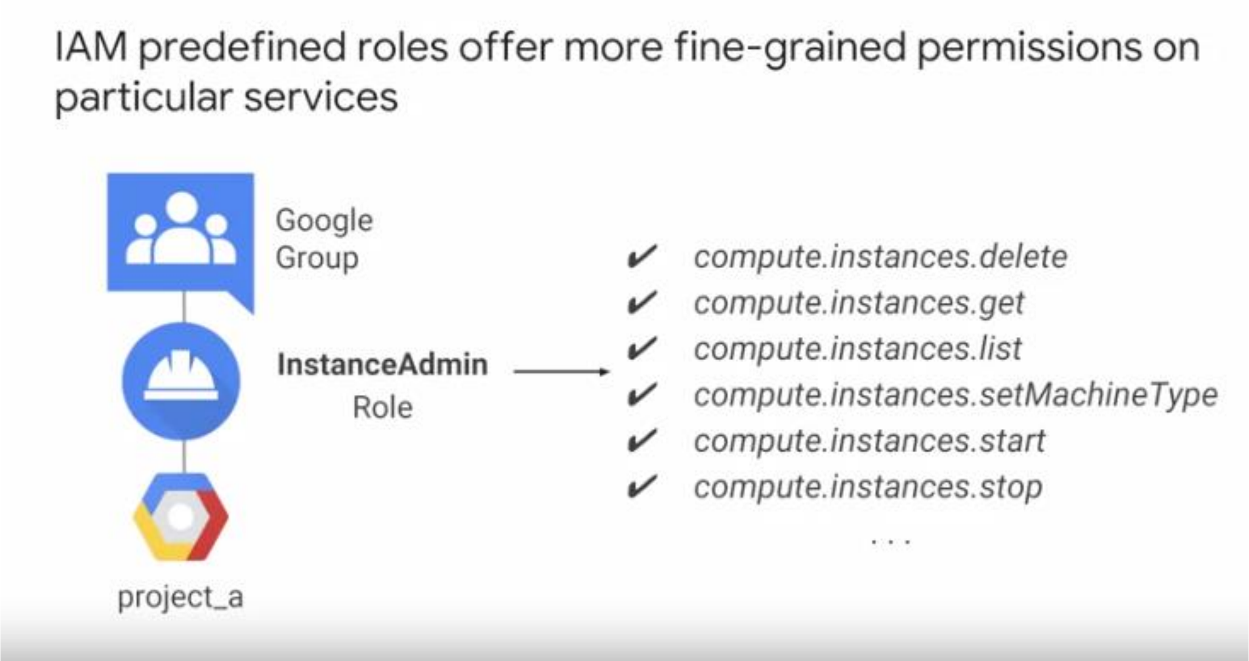
IAM **predefined** roles apply to a particular GCP service in a project

can do what

on Compute Engine resources in this project, or folder, or org

Pre-defined roles are defined on top of a GCP Service offered by GCP

Is a larger set



IAM predefined roles offer more fine-grained permissions on particular services

Google Group

InstanceAdmin Role →

- ✔ *compute.instances.delete*
- ✔ *compute.instances.get*
- ✔ *compute.instances.list*
- ✔ *compute.instances.setMachineType*
- ✔ *compute.instances.start*
- ✔ *compute.instances.stop*

. . .

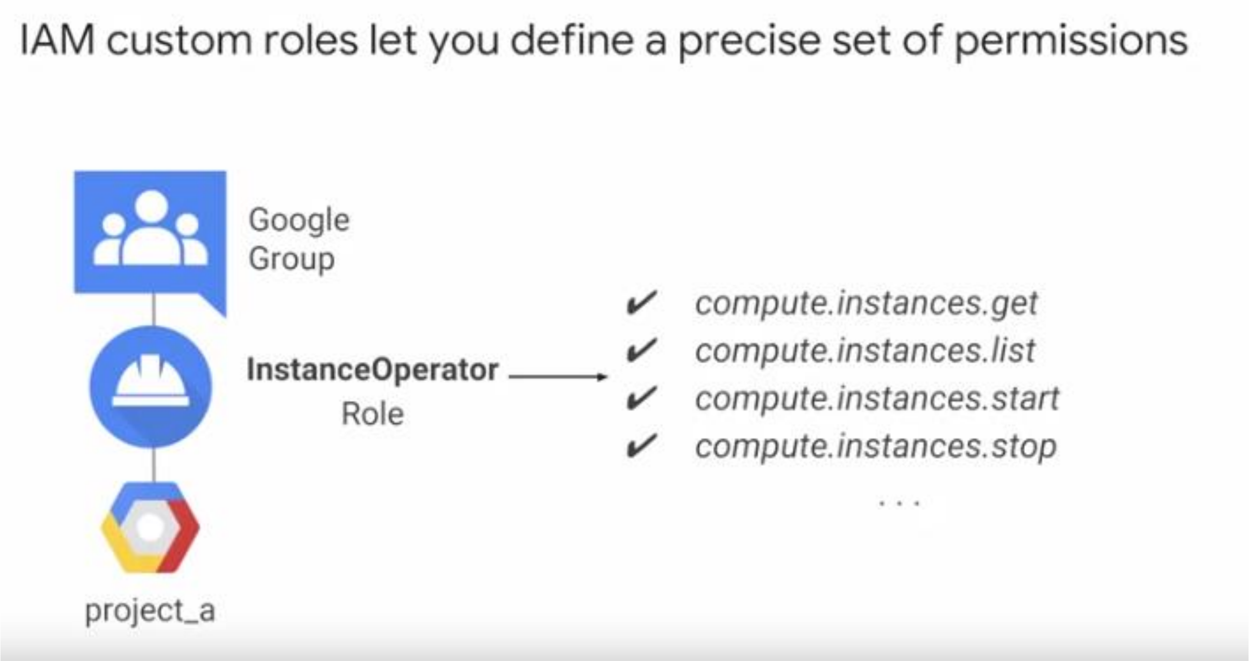
project_a

You can have more granular Role by defining a Custom role:

Granting the Principle of least Privilege for a user

Custom roles can be used only at

Project Level and Organizational level and not at the Folder level



IAM custom roles let you define a precise set of permissions

Google Group

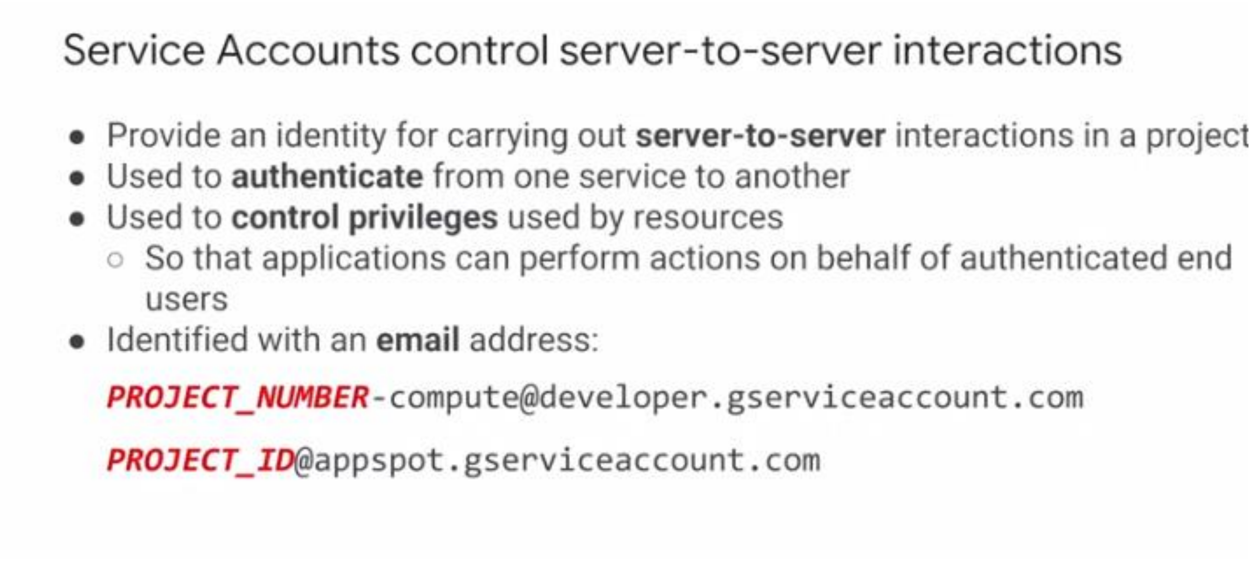
InstanceOperator Role

- ✓ *compute.instances.get*
- ✓ *compute.instances.list*
- ✓ *compute.instances.start*
- ✓ *compute.instances.stop*

. . .

project_a

Now instead of a Person I would like to grant it to Virtual Machine /App Engine what should I do then ?



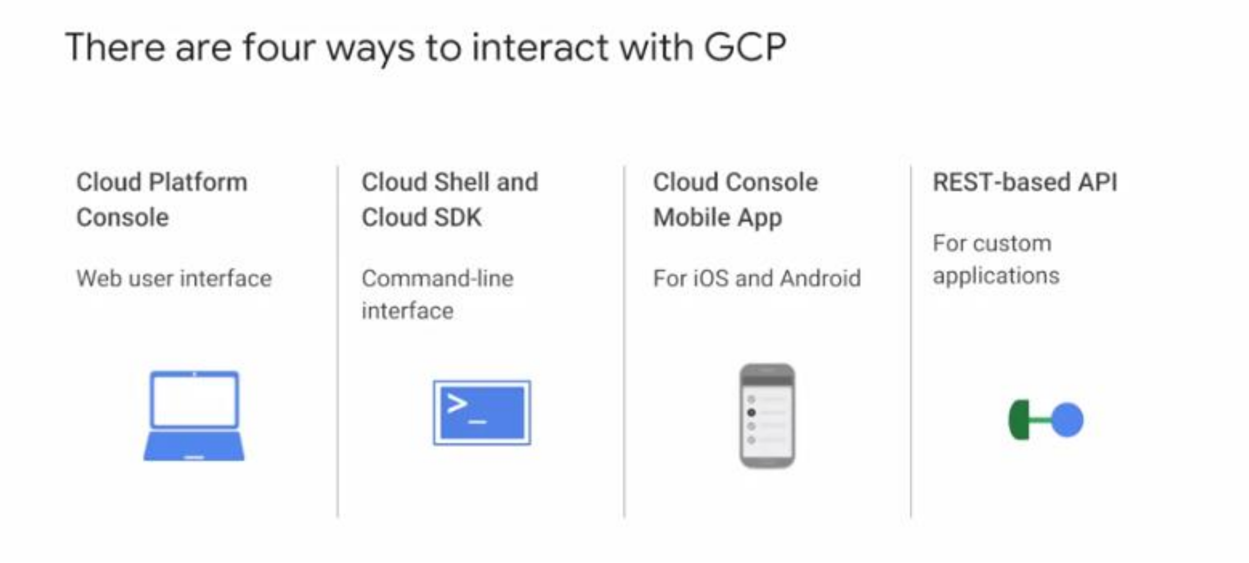
Service Accounts control server-to-server interactions

- Provide an identity for carrying out **server-to-server** interactions in a project
- Used to **authenticate** from one service to another
- Used to **control privileges** used by resources
  - So that applications can perform actions on behalf of authenticated end users
- Identified with an **email** address:

```
PROJECT_NUMBER-compute@developer.gserviceaccount.com

PROJECT_ID@appspot.gserviceaccount.com
```

There are 4 different ways to interact with GCP:



There are four ways to interact with GCP

Cloud Platform Console

Web user interface

Cloud Shell and Cloud SDK

Command-line interface

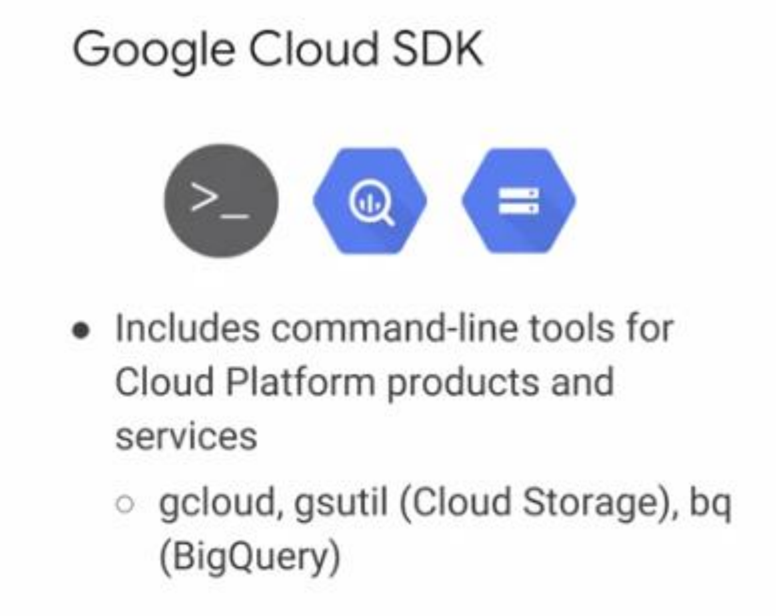
Cloud Console Mobile App

For iOS and Android

REST-based API

For custom applications

SDK good examples:



Google Cloud SDK

- Includes command-line tools for Cloud Platform products and services
  - gcloud, gsutil (Cloud Storage), bq (BigQuery)

Google Cloud Storage

Object Storage

- Unlimited storage with no minimum object size.
- Worldwide accessibility and worldwide storage locations.
- Low latency (time to first byte typically tens of milliseconds).

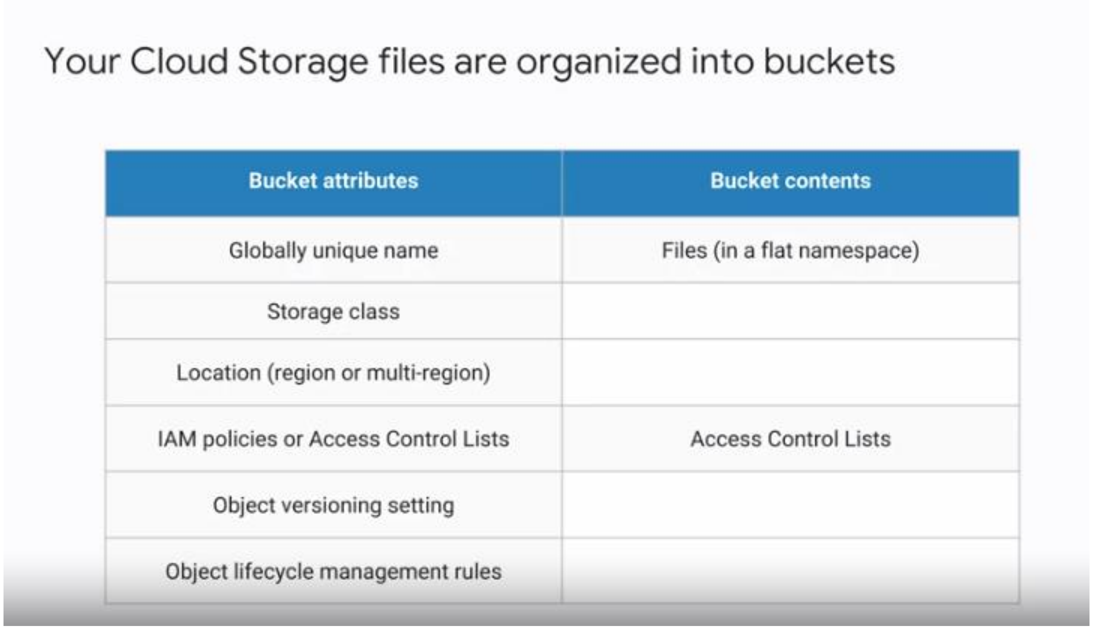
- High durability (99.999999999% annual durability).
- Geo-redundancy if the data is stored in a multi-region or dual-region.
- A uniform experience with Cloud Storage features, security, tools, and APIs.

11 9's durability and 99.99%

Bucket to store objects and these are identified as a key value pair

Objects stored are immutable (You cannot modify objects stored)

Hence you store different versions of the same object



Your Cloud Storage files are organized into buckets

Bucket attributes	Bucket contents
Globally unique name	Files (in a flat namespace)
Storage class	
Location (region or multi-region)	
IAM policies or Access Control Lists	Access Control Lists
Object versioning setting	
Object lifecycle management rules	

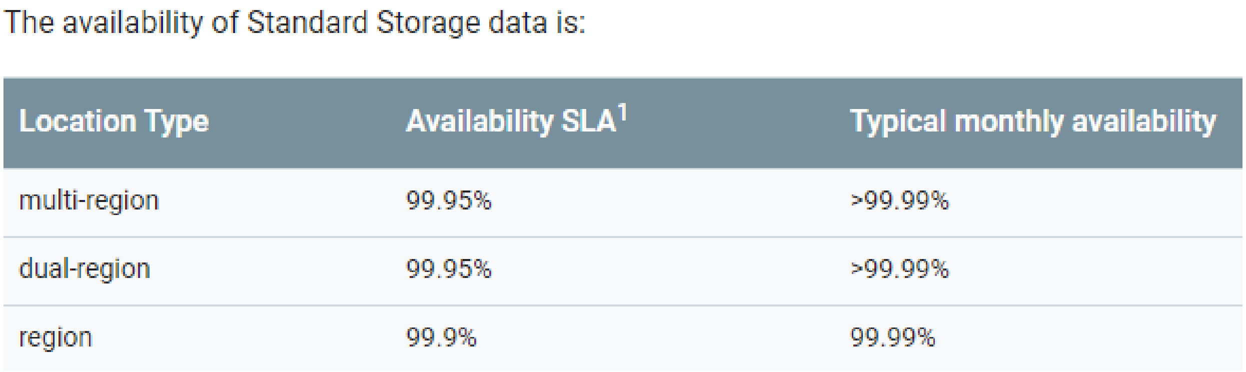
Use cases:

- Store the objects and access them thro' an URL
- Use it for archival
- Use the data stored for DR (Disaster Recovery)
- Store large objects and stream them ..

Storage Classes:

Standard Storage

Standard Storage is best for data that is frequently accessed ("hot" data) and/or stored for only brief periods of time.



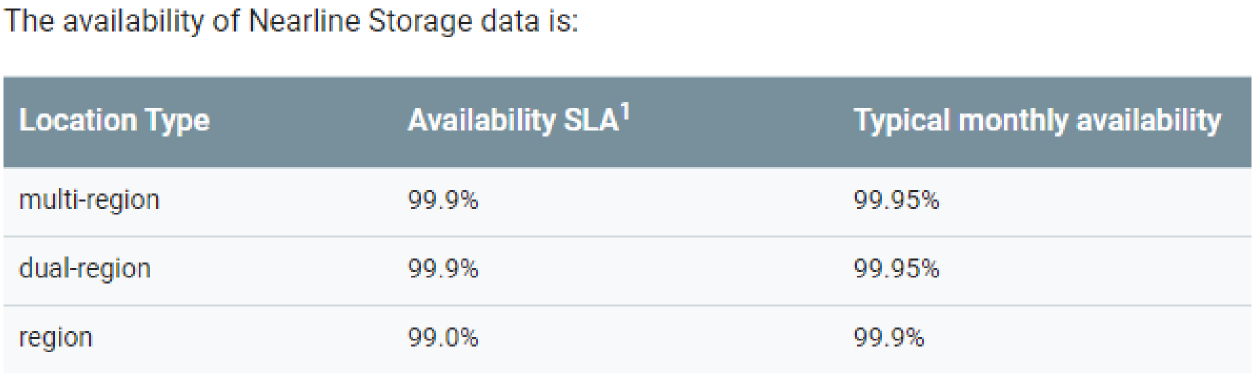
The availability of Standard Storage data is:

Location Type	Availability SLA[1]	Typical monthly availability
multi-region	99.95%	>99.99%
dual-region	99.95%	>99.99%
region	99.9%	99.99%

Near-line Storage

Near-line Storage is a low-cost, highly durable storage service for storing infrequently accessed data.

Near-line Storage is a better choice than Standard Storage in scenarios where slightly lower availability, a 30-day minimum storage duration, and costs for data access are acceptable trade-offs for lowered at-rest storage costs.



The availability of Nearline Storage data is:

Location Type	Availability SLA[1]	Typical monthly availability
multi-region	99.9%	99.95%
dual-region	99.9%	99.95%
region	99.0%	99.9%

Cold-line Storage

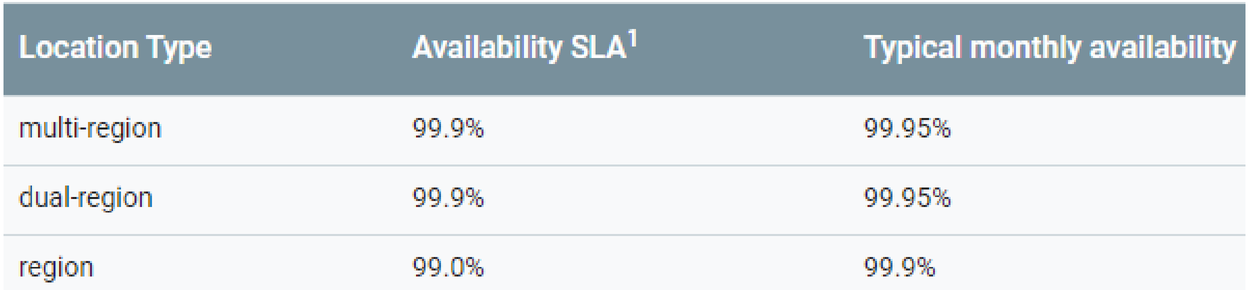
Cold-line Storage is a very-low-cost, highly durable storage service for data archiving, online backup, and disaster recovery.

Unlike other "cold" storage services, your data is available within milliseconds, not hours or days.

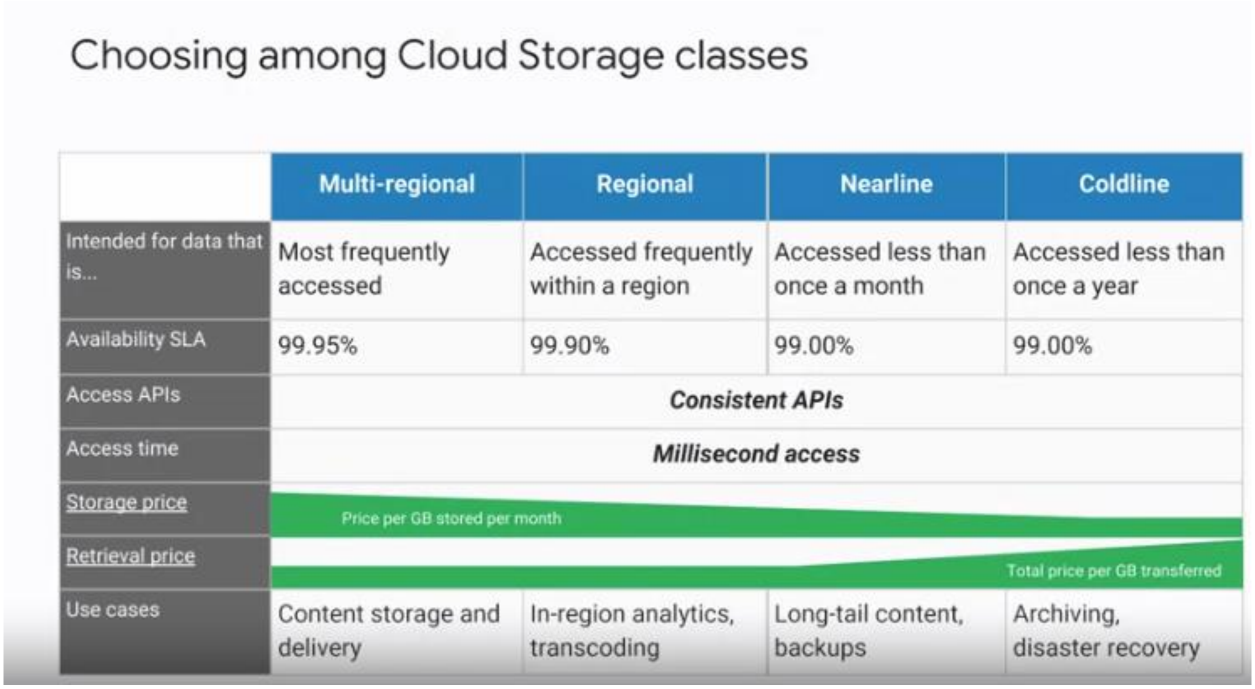
For example:

- Cold data storage - Infrequently accessed data, such as data stored for legal or regulatory reasons, can be stored at low cost as Coldline Storage and be available when you need it.
- Disaster recovery - In the event of a disaster recovery event, recovery time is key. Cloud Storage provides low latency access to data stored as Coldline Storage.

The availability of Coldline Storage data is:

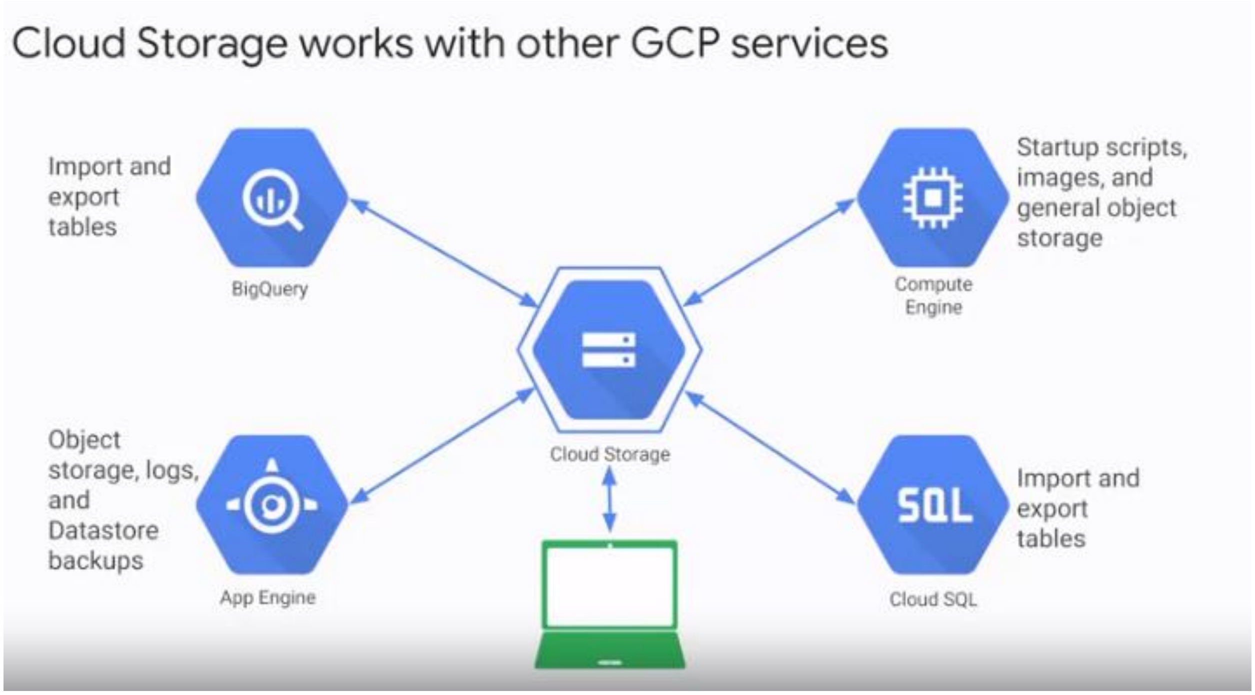


Location Type	Availability SLA[1]	Typical monthly availability
multi-region	99.9%	99.95%
dual-region	99.9%	99.95%
region	99.0%	99.9%



Choosing among Cloud Storage classes

	Multi-regional	Regional	Nearline	Coldline
Intended for data that is...	Most frequently accessed	Accessed frequently within a region	Accessed less than once a month	Accessed less than once a year
Availability SLA	99.95%	99.90%	99.00%	99.00%
Access APIs	*Consistent APIs*			
Access time	*Millisecond access*			
Storage price	Price per GB stored per month			
Retrieval price	Total price per GB transferred			
Use cases	Content storage and delivery	In-region analytics, transcoding	Long-tail content, backups	Archiving, disaster recovery



Cloud Storage works with other GCP services

Demos and Assignments on IAM and Storage

hsbc-2019-<urname>-proj2

create 3 google groups

add atleast 2 gmail accounts / any valid email accounts

dev-usrs-hsbc-urname-grp1 - editor(

Add predefined roles to the group and you may ask the team to create cloud resources and access

app engine admin

Compute admin

Storage admin

custom roles as applicable and add them)

compute instance admin

storage editor

Appengine codeviewer

test-usrs-hsbc-urname-grp2 - readonly ((predefined roles, custom roles as applicable and add them)

devops-usrs-hsbc-urname-grp3 - owner((predefined roles, custom roles as applicable and add them)

Demo 1: GCS buckets

1.Create a user with storage object creator permission

2. Create a bucket in the user 1 login

3. Switch to user2 and try uploading objects

4. success!

Note: user 2 does not have any other access except storage object creator.

```
gsutil cp seconduser.txt gs://[bucket-name]

gsutil ls gs://[bucket-name]

gsutil rm -r gs://[bucket-name]

gsutil ls gs://[bucket-name]

gsutil rm -r gs://[bucket-name]

gsutil mb -l us-east1 gs://[bucket-name]
```

Demo 2: Versioning

```
gsutil versioning get gs://[BUCKET_NAME]

gsutil versioning set on gs://[BUCKET_NAME]

gsutil versioning set off gs://[BUCKET_NAME]

gsutil versioning get gs://[BUCKET_NAME]

gsutil stat gs://[BUCKET_NAME]/[OBJECT_NAME]

gsutil ls -a gs://[BUCKET_NAME]

gsutil setmeta -h "[METADATA_KEY]:[METADATA_VALUE]" gs://[BUCKET_NAME]/[OBJECT_NAME]
```

Example:

```
gsutil setmeta -h "x-goog-meta- color:blue" gs://mytestbuck1232/kitten.png
```

Demo 3: Lifecycle policy

To transfer the objects across different storage classes automatically, a life cycle policy can be set up on a bucket.

Demo 4: Bucket policy only and requester pays

Enabling "bucket policy only" feature allows you to control access using IAM alone. If not enabled, permissions can be controlled by either IAM or by bucket policy (object level permissions are enabled).

Requester pays:

Enable this option to transfer the billing for the current data transfer out job to the requester.

Demo 7: Storage transfer service (Amazon S3 to GCS)

Q) When to use Storage Transfer Service? vs When to use gsutil or console to transfer objects between buckets?

A) Provide an answer based on your learning

Additional Assignment to Try out:

Code lab:

https://codelabs.developers.google.com/codelabs/gcp-aws-gsutil/index.html?index=..%2F..index#0