**Core Services**

**1.**

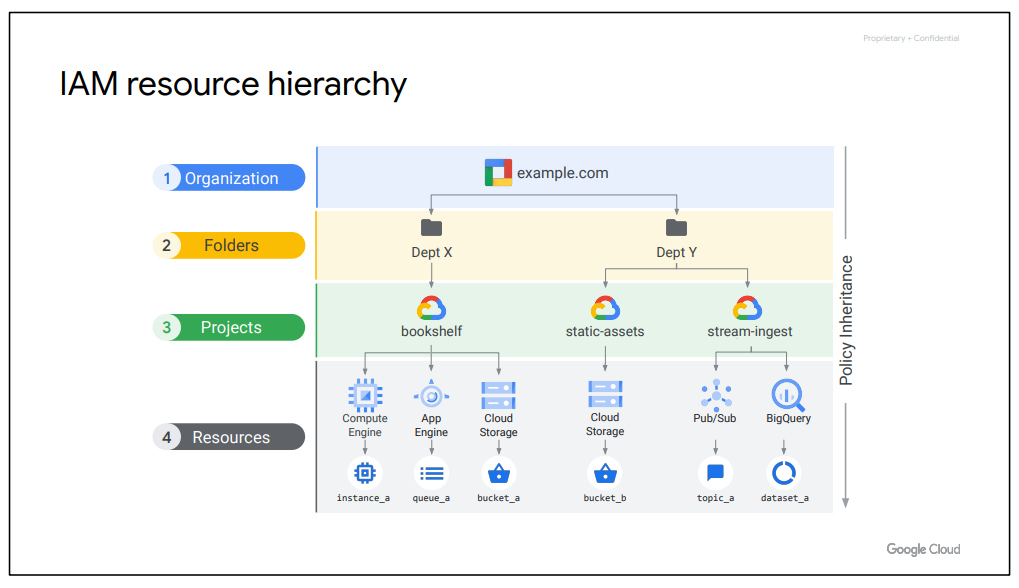
**IAM (Identity and Access Management)**

It basically refers to three basic things :

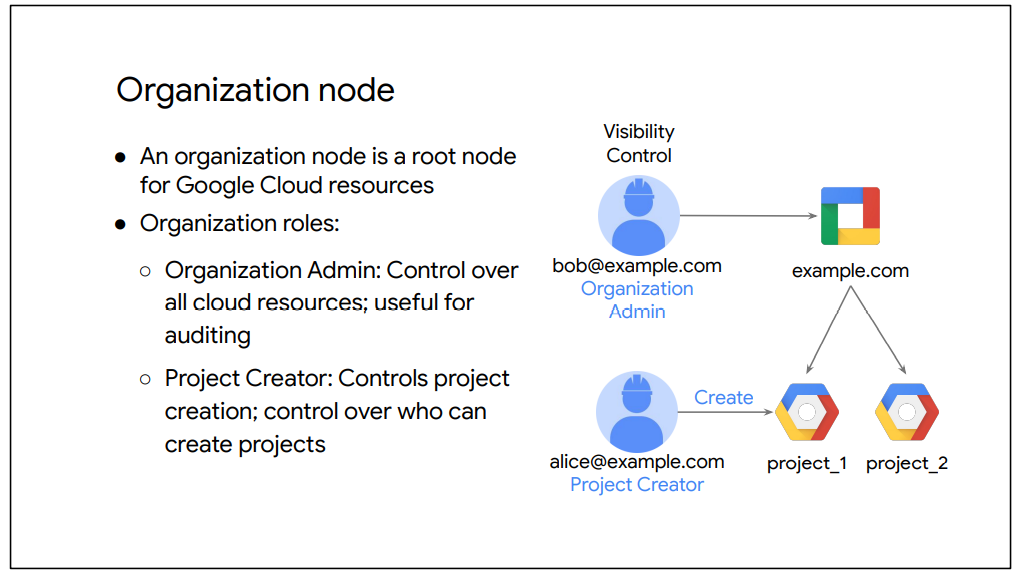
* Who
* Can do what
* On which resource

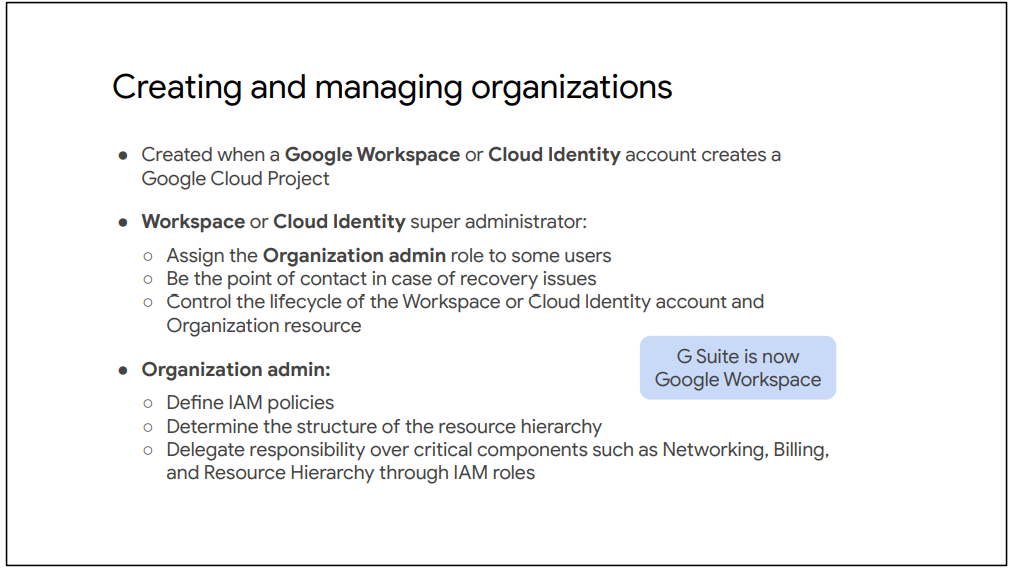
**IAM objects**

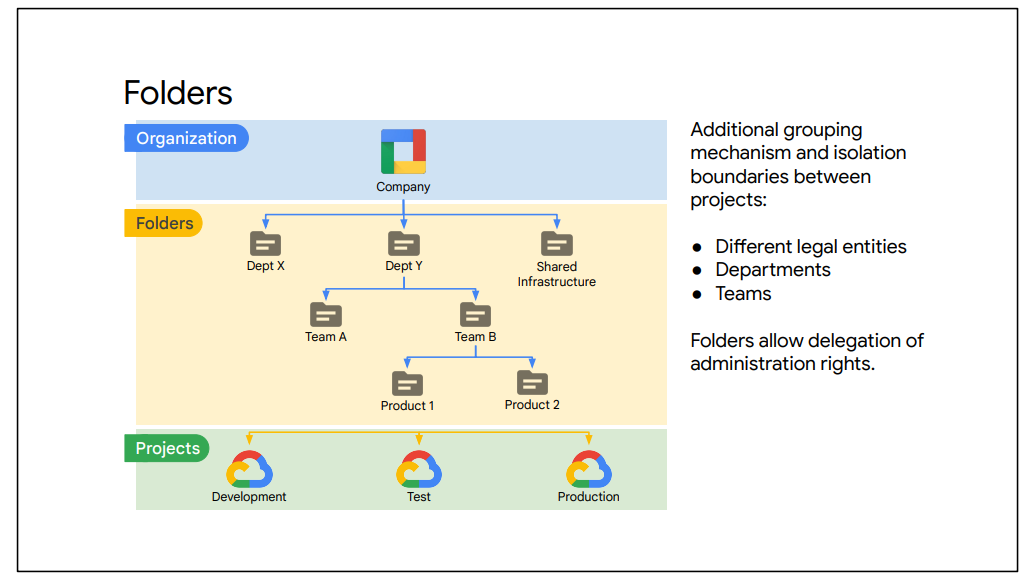
* Organisations
* Folder
* Projects
* Resources
* Roles
* Members

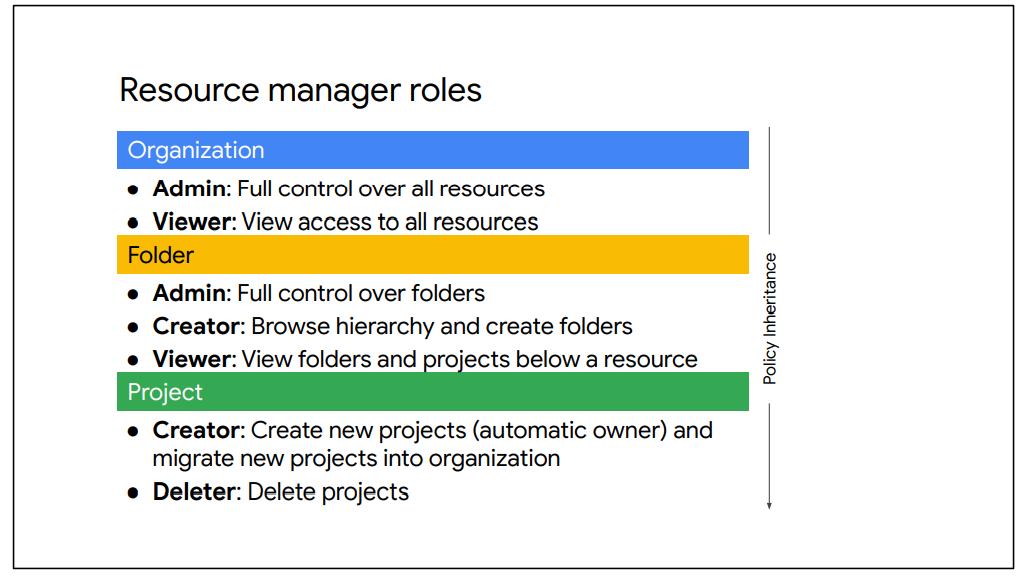


**Organization Node**







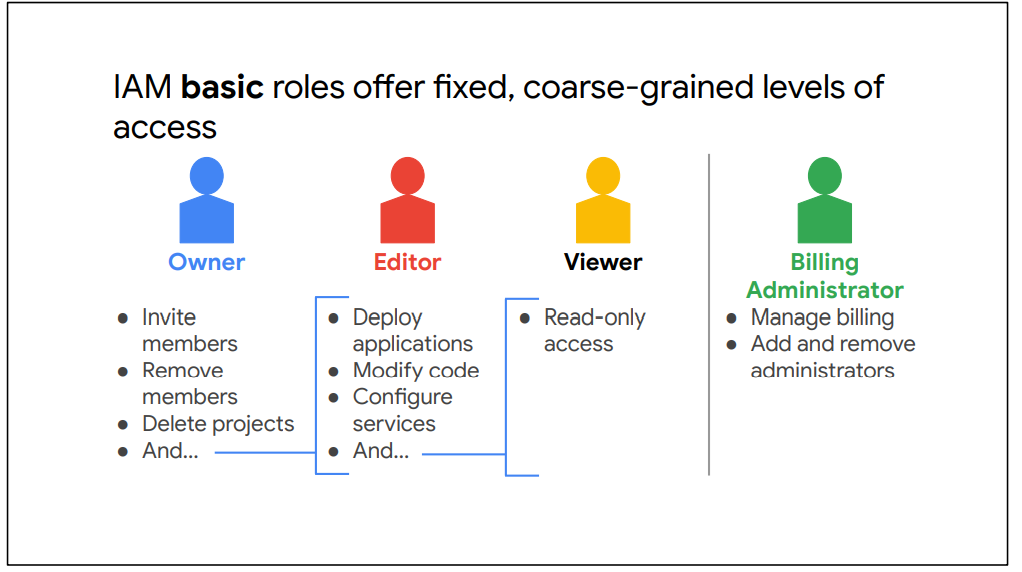


**Roles**

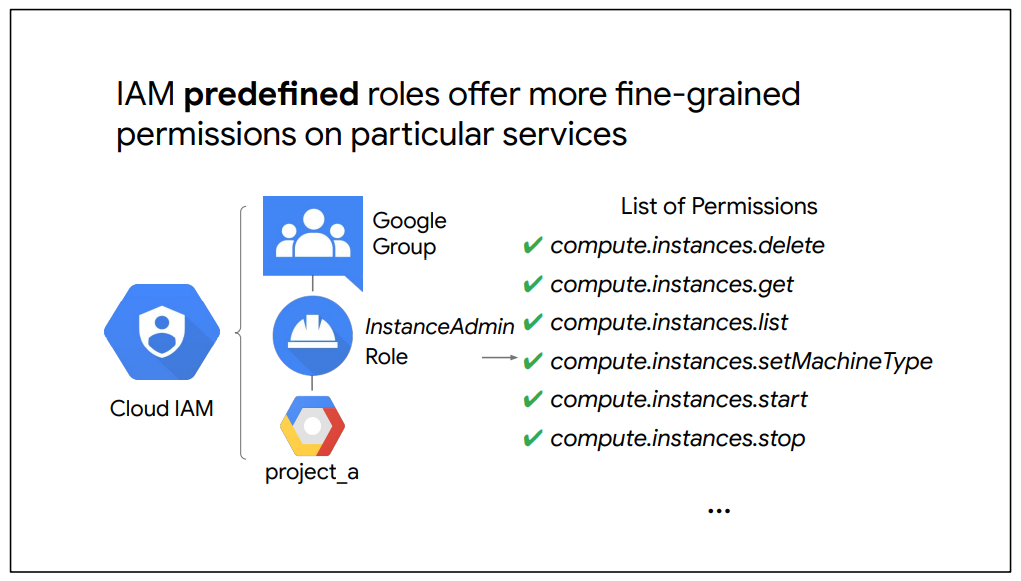
There are three types of IAM Roles

* Basic
* Predefined
* Custom

IAM basic roles apply across all Google Cloud Services in a project



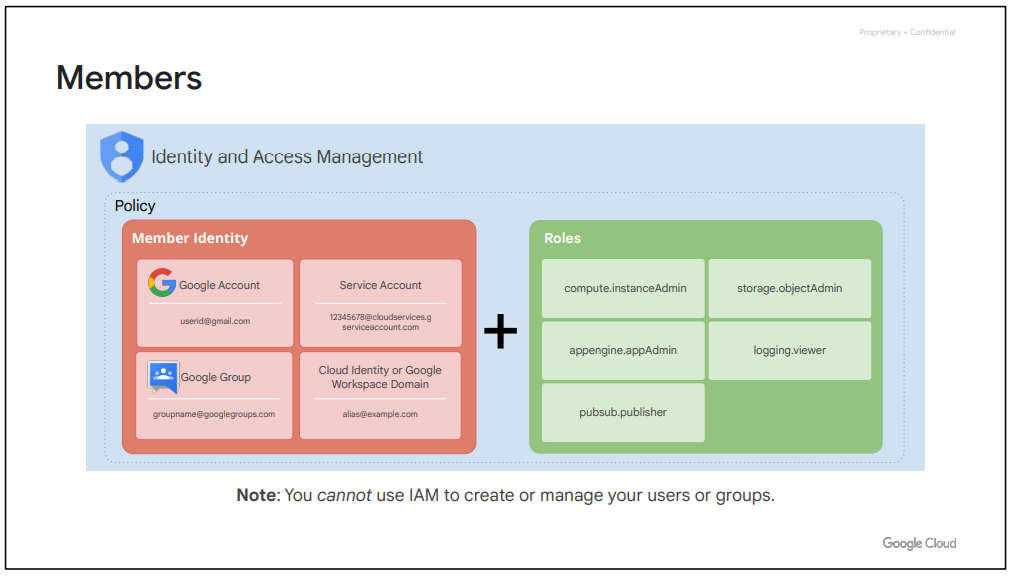
IAM predefined roles apply to a particular GCP Service in a project.



Compute Engine IAM roles

* Compute Admin : Full control of all compute engine resource.
* Network Admin : Permissions to create , modify and delete networking resource , except for firewalls rule and SSL certificates
* Storage Admin : Permissions to create , modify, and delete disks , images and snapshots

IAM custom roles let you define a precise set of permissions.



There are five different types of members :

* Google Accounts : A Google account represents a developer, an administrator, or any other person who interacts with Google Cloud. Any email address that is associated with a Google account can be an identity, including gmail.com or other domains. New users can sign up for a Google account by going to the Google account signup page, without receiving mail through Gmail.
* Service Account : A service account is an account that belongs to your application instead of to an individual end user. When you run code that is hosted on Google Cloud, you specify the account that the code should run as. You can create as many service accounts as needed to represent the different logical components of your application.
* Google Group : A Google group is a named collection of Google accounts and service accounts. Every group has a unique email address that is associated with the group. Google groups are a convenient way to apply an access policy to a collection of users. You can grant and change access controls for a whole group at once instead of granting or changing access controls one-at-a-time for individual users or service accounts.
* Workspace domains : A Workspace domain represents a virtual group of all the Google accounts that have been created in an organization's Workspace account. Workspace domains represent your organization's internet domain name, such as example.com, and when you add a user to your Workspace domain, a new Google account is created for the user inside this virtual group, such as [username@example.com](mailto:username@example.com).
* Cloud Identity : Google Cloud customers who are not Workspace customers can get these same capabilities through Cloud Identity. Cloud Identity lets you manage users and groups using the Google Admin Console, but you do not pay for or receive Workspace’s collaboration products such as Gmail, Docs, Drive, and Calendar.

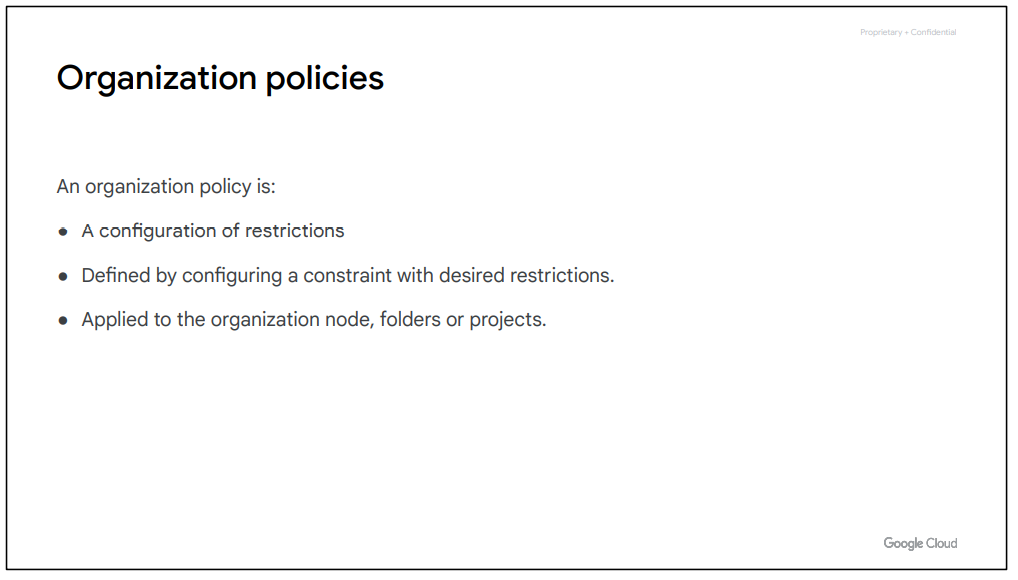
**IAM policies**

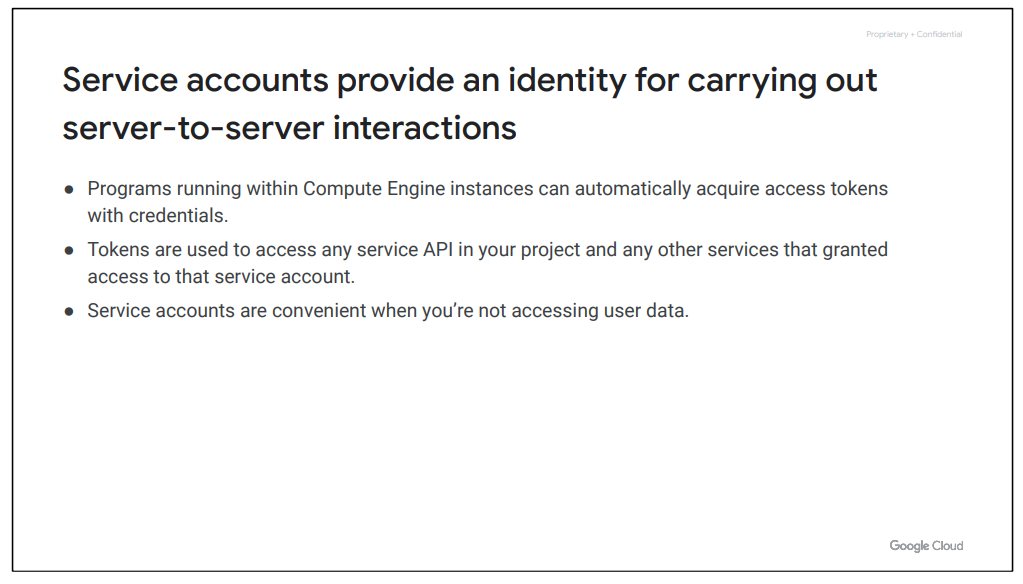
* A policy consists of list of bindings.
* A binding binds a list of members to a role.

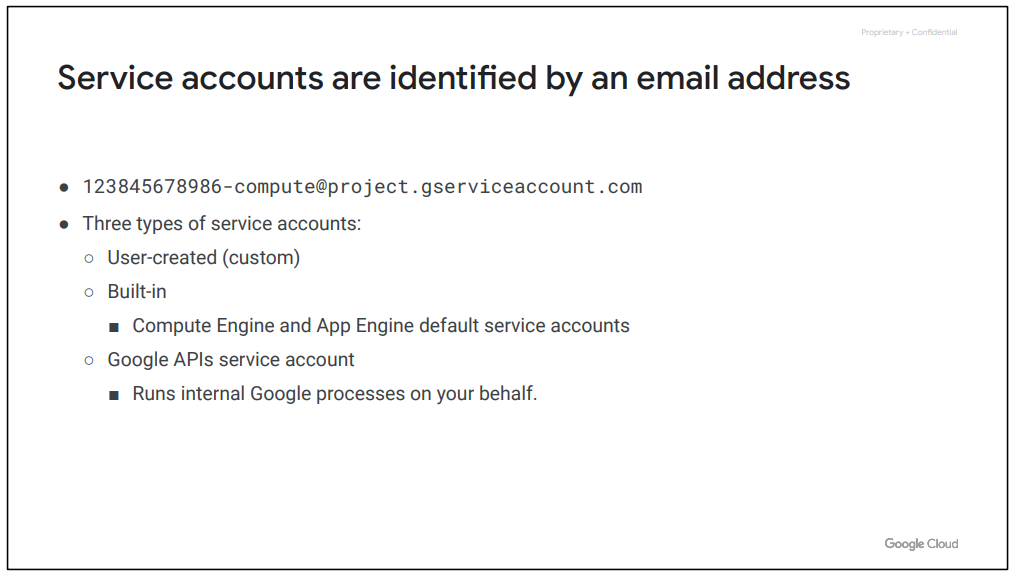
IAM Conditions allow you to define and enforce conditional, attribute-based access control for Google Cloud resources.

With IAM Conditions, you can choose to grant resource access to identities (members) only if configured conditions are met. For example, this could be done to configure temporary access for users in the event of a production issue or to limit access to resources only for employees making requests from your corporate office.

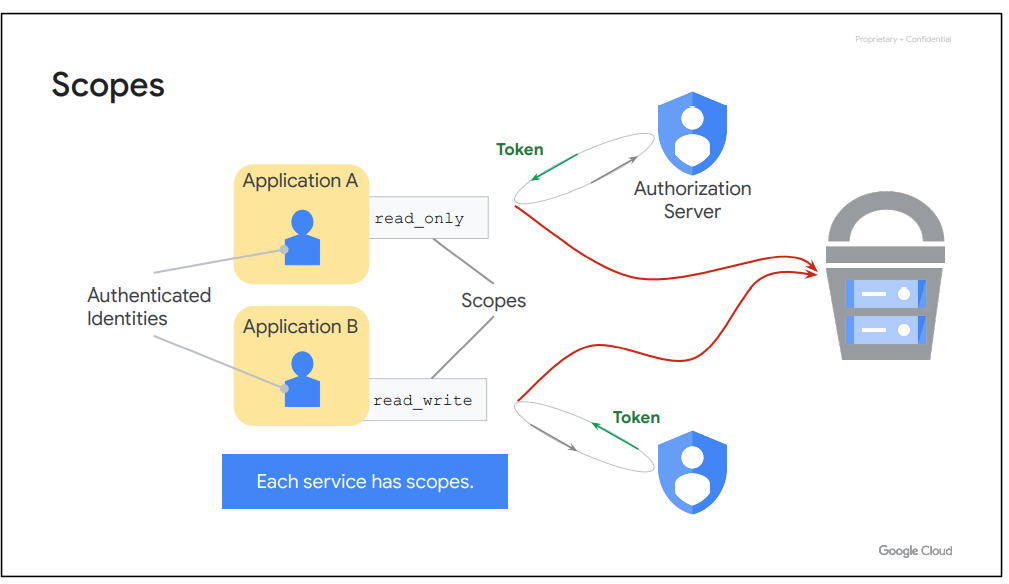
Conditions are specified in the role bindings of a resource's IAM policy. When a condition exists, the access request is only granted if the condition expression evaluates to true. Each condition expression is defined as a set of logic statements allowing you to specify one or more attributes to check.





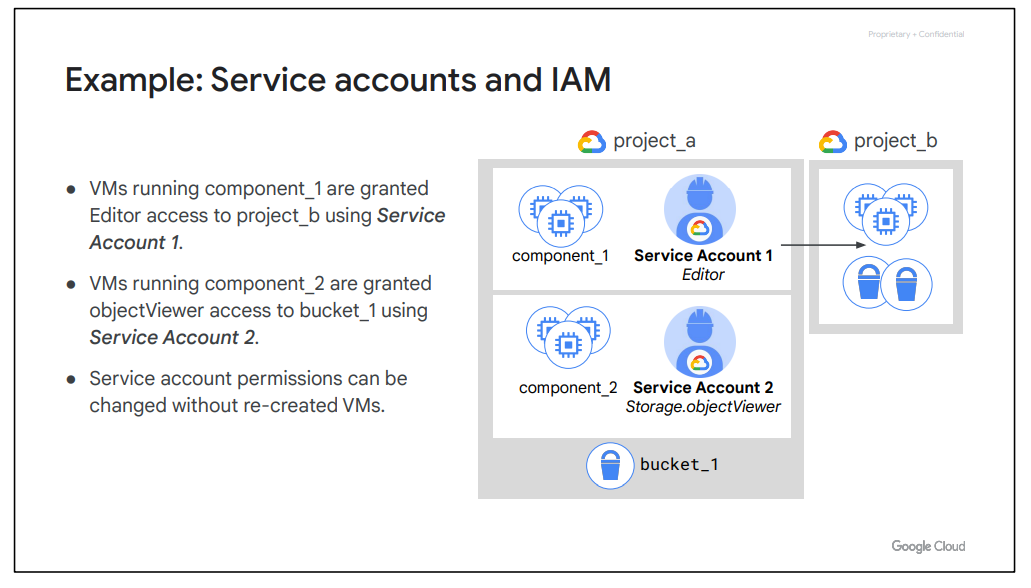


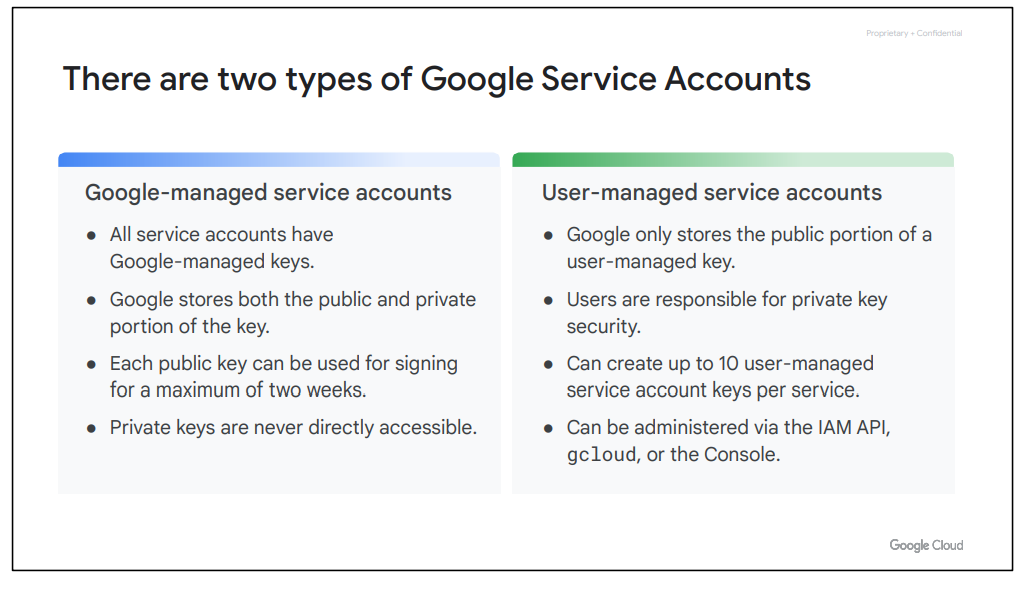




Now, authorization is the process of determining what permissions an authenticated identity has on a set of specified resources. Scopes are used to determine whether an authenticated identity is authorized.

In the example shown here, Applications A and B contain Authenticated Identities (or service accounts). Let’s assume that both applications want to use a Cloud Storage bucket. They each request access from the Google Authorization server, and in return they receive an access token. Application A receives an access token with read-only scope, so it can only read from the Cloud Storage bucket. Application B, in contrast, receives an access token with read-write scope, so it can read and modify data in the Cloud Storage bucket.

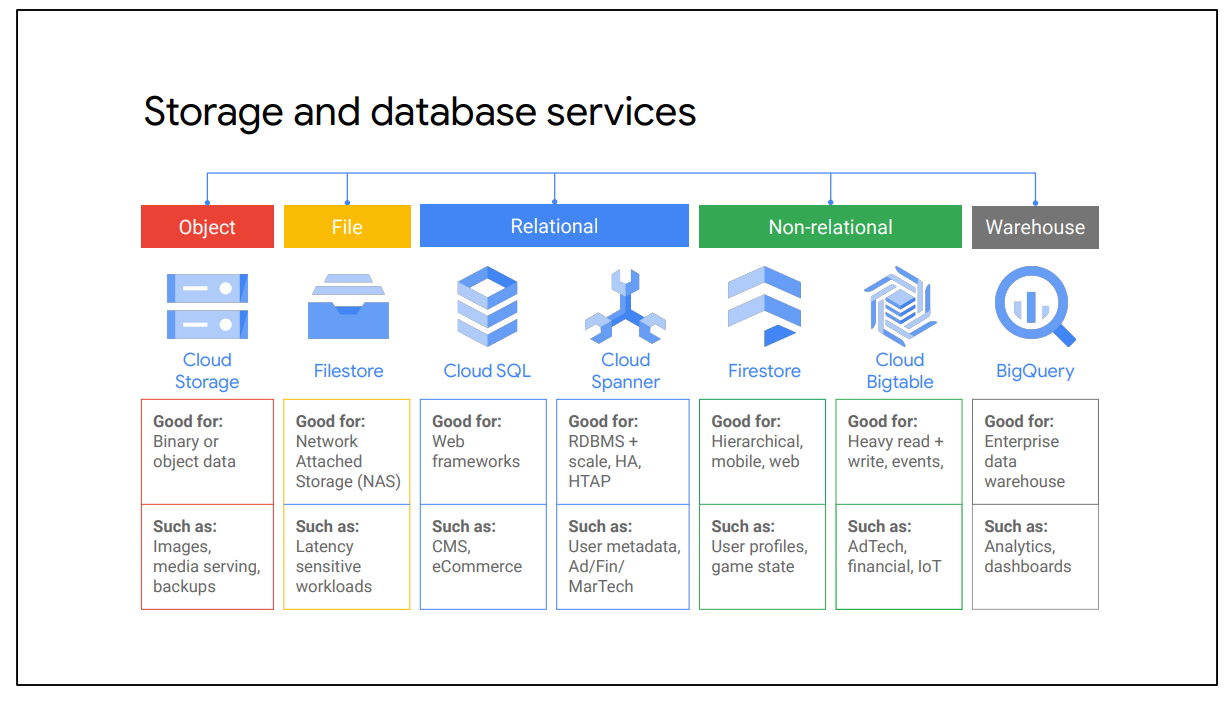


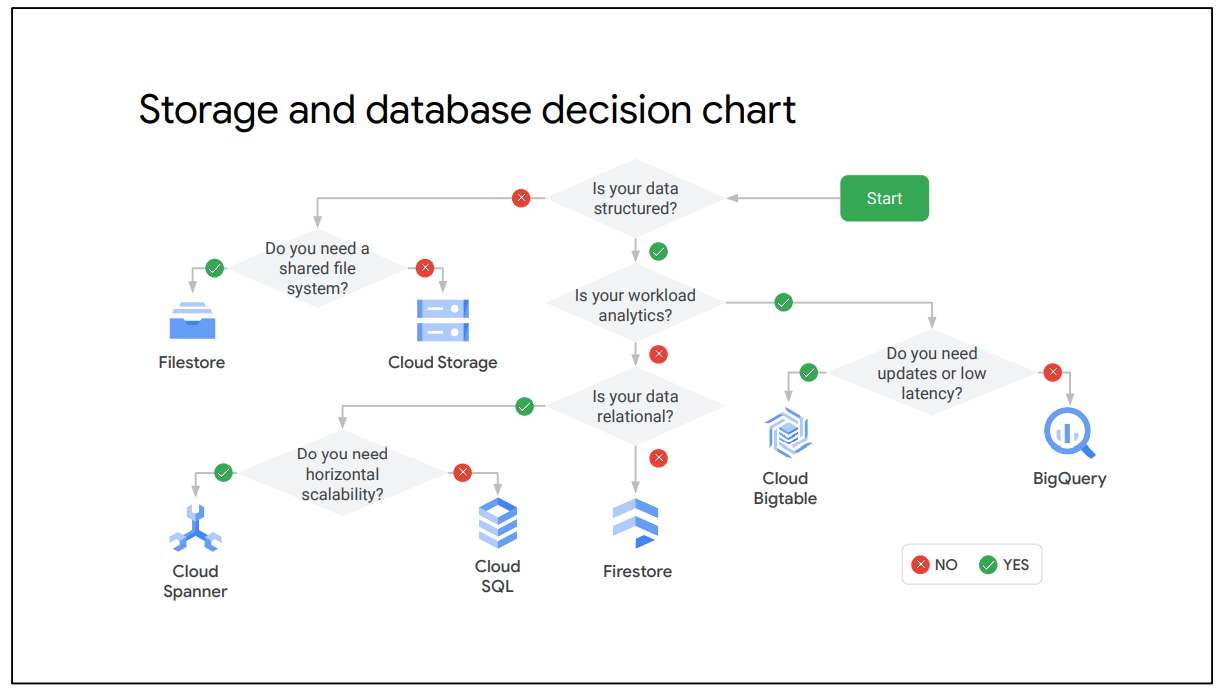
HH

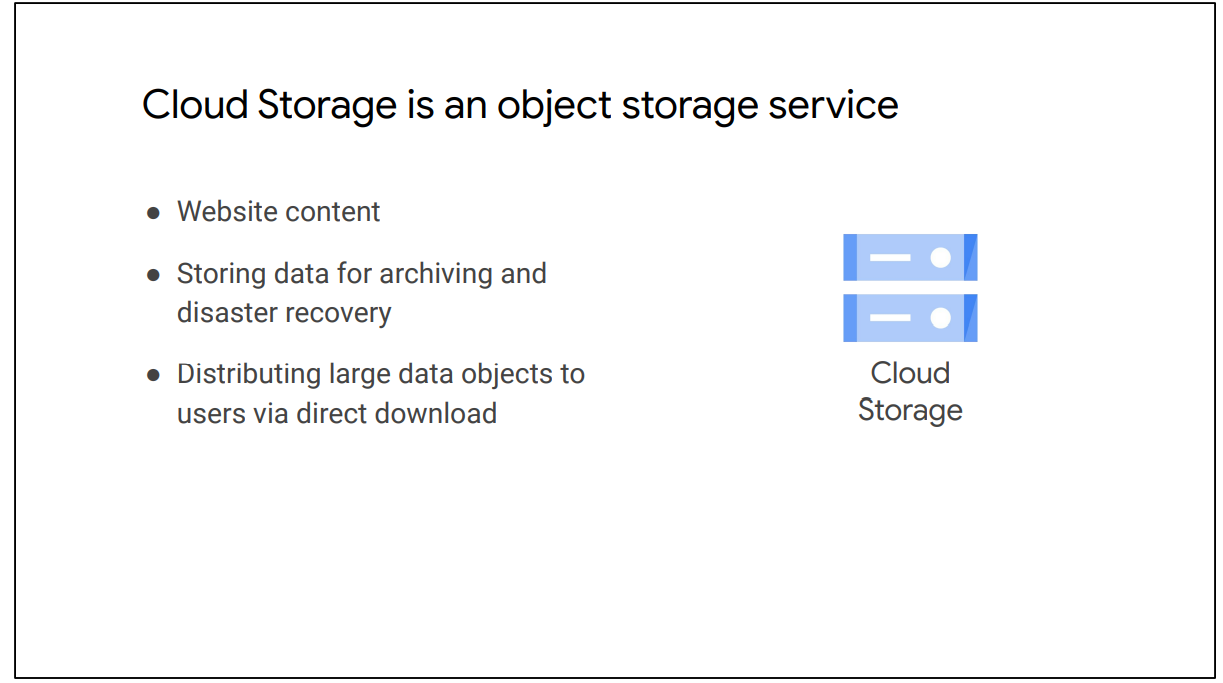


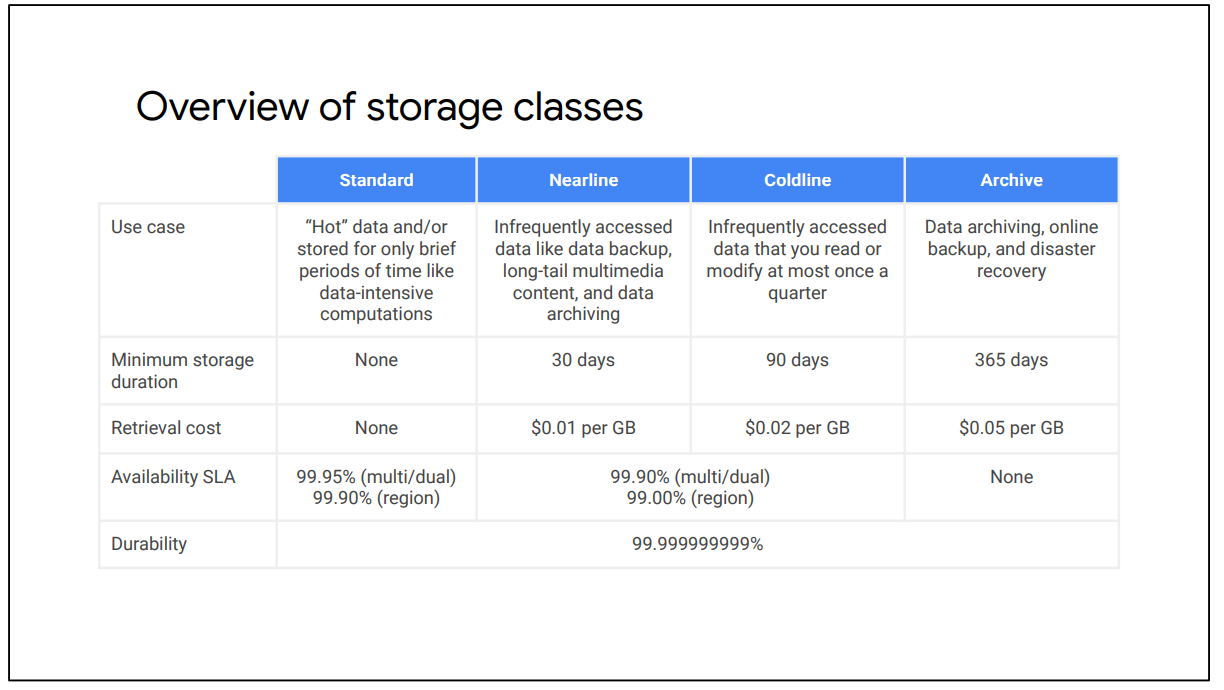
Applications and resources protected by IAP can only be accessed through the proxy by users and groups with the correct IAM role. When you grant a user access to an application or resource by IAP, they’re subject to the fine-grained access controls implemented by the product in use without requiring a VPN. IAP performs authentication and authorization checks when a user tries to access an IAP-secured resource, as shown on the right.

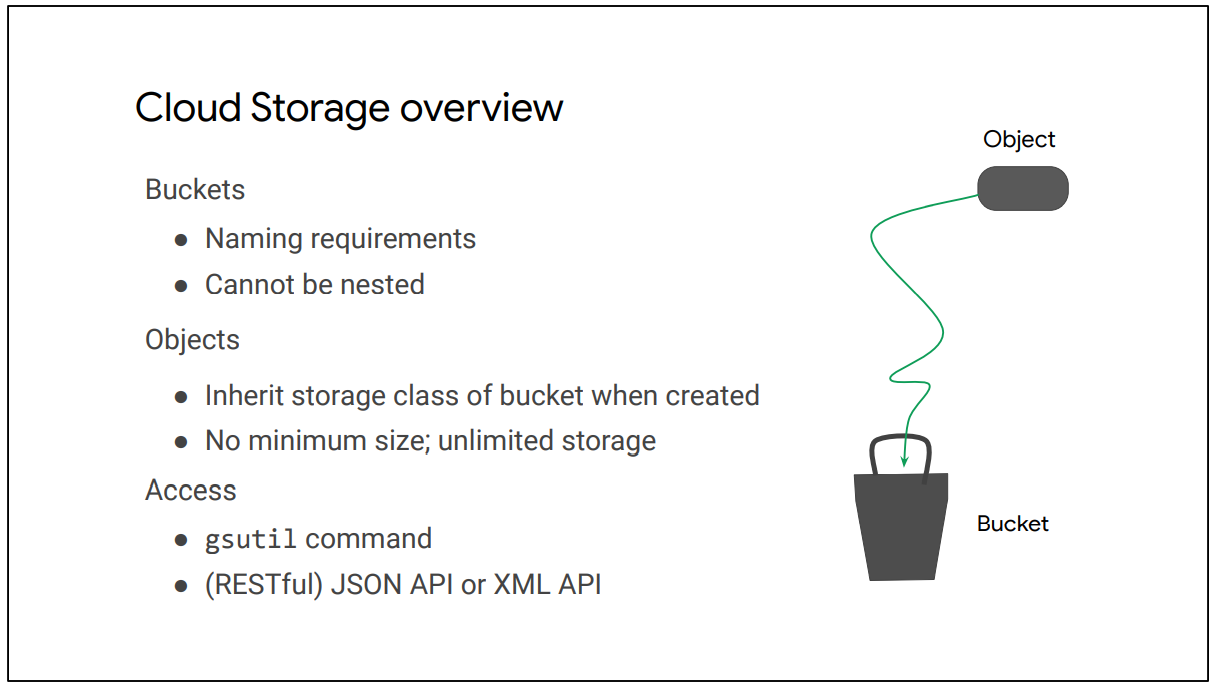
**2.**





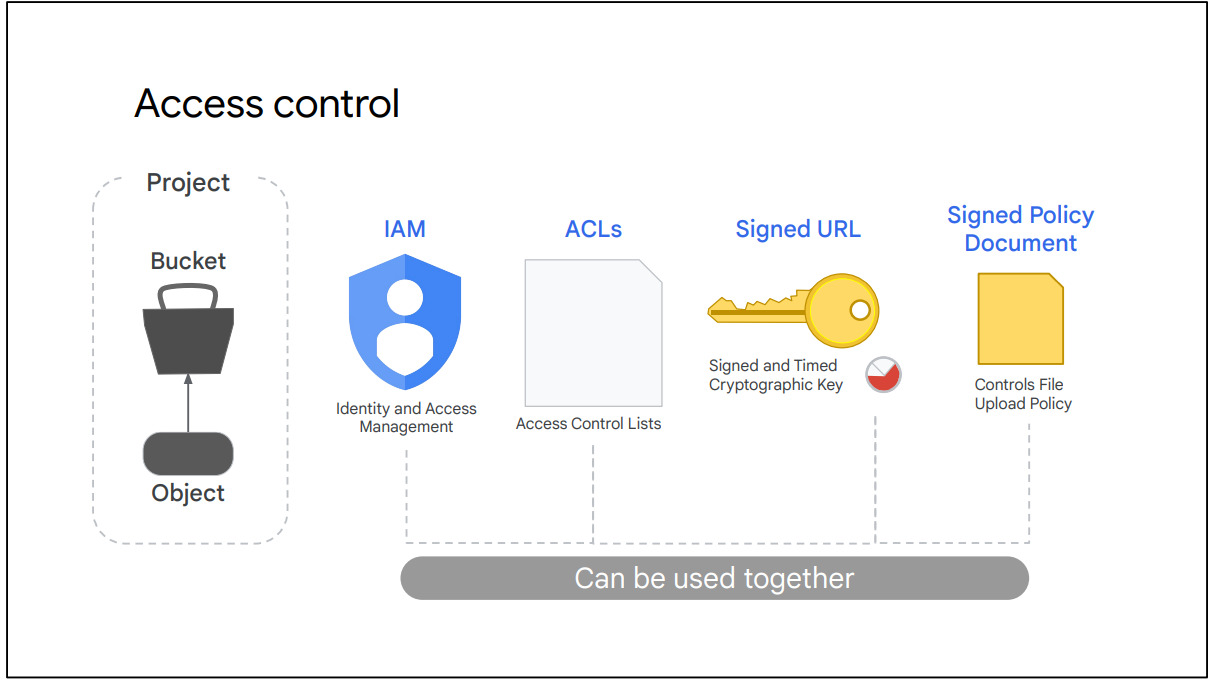






**Changing Default Storage Class**

* Default class is applied to new objects.
* Regional bucket can never be changed to multi-region/dual buckets
* Multi-region bucket can never be changed to regional.
* Objects can be moved from bucket to bucket.
* Object Lifecycle Management can manage the classes of objects.



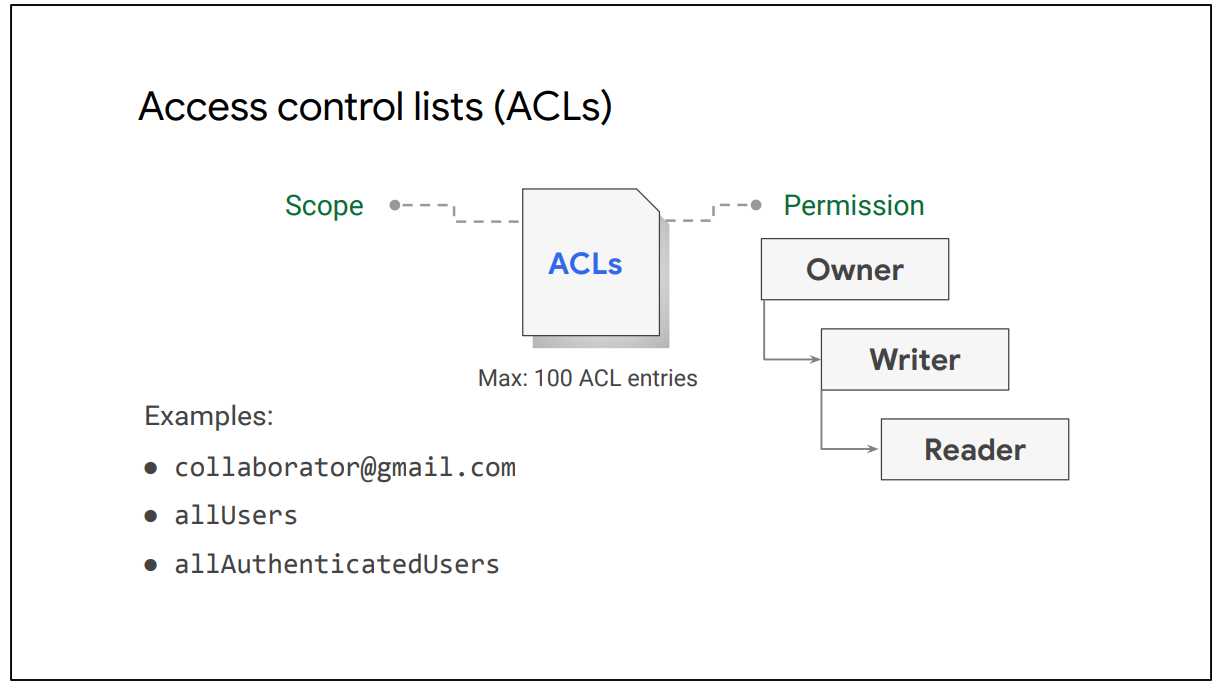
Let’s look at access control for your objects and buckets that are part of a project.

● We can use IAM for the project to control which individual user or service account can see the bucket, list the objects in the bucket, view the names of the objects in the bucket, or create new buckets. For most purposes, IAM is sufficient, and roles are inherited from project to bucket to object.

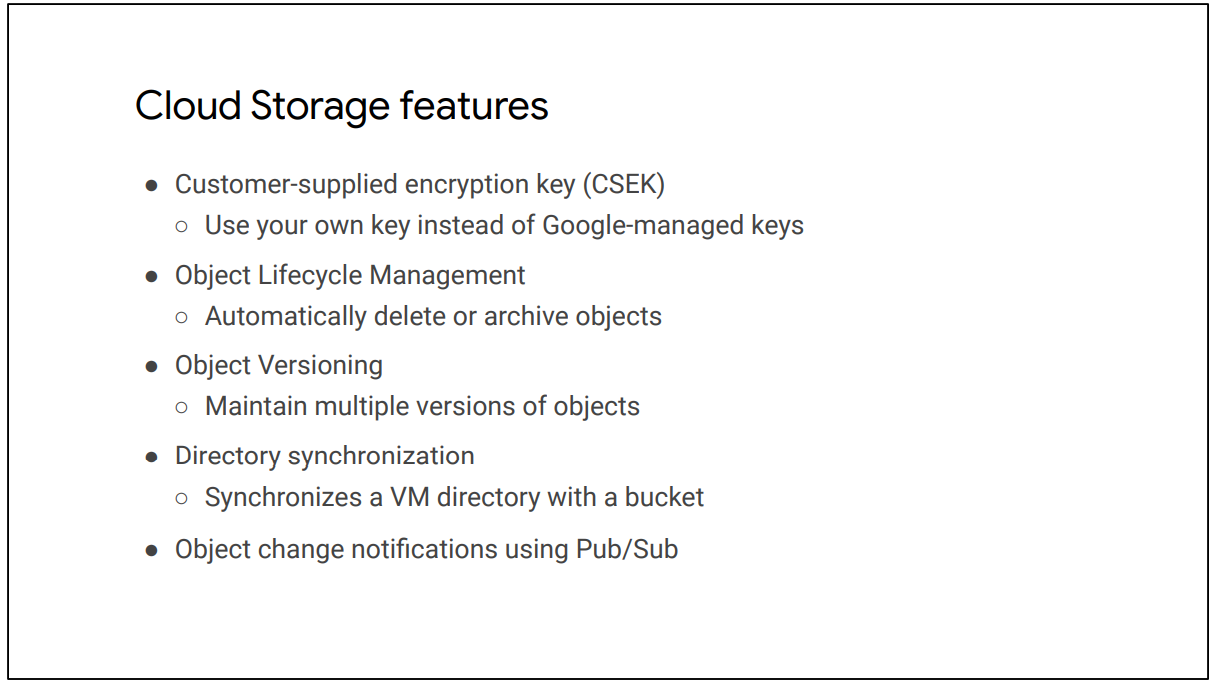
● Access control lists or ACLs offer finer control.

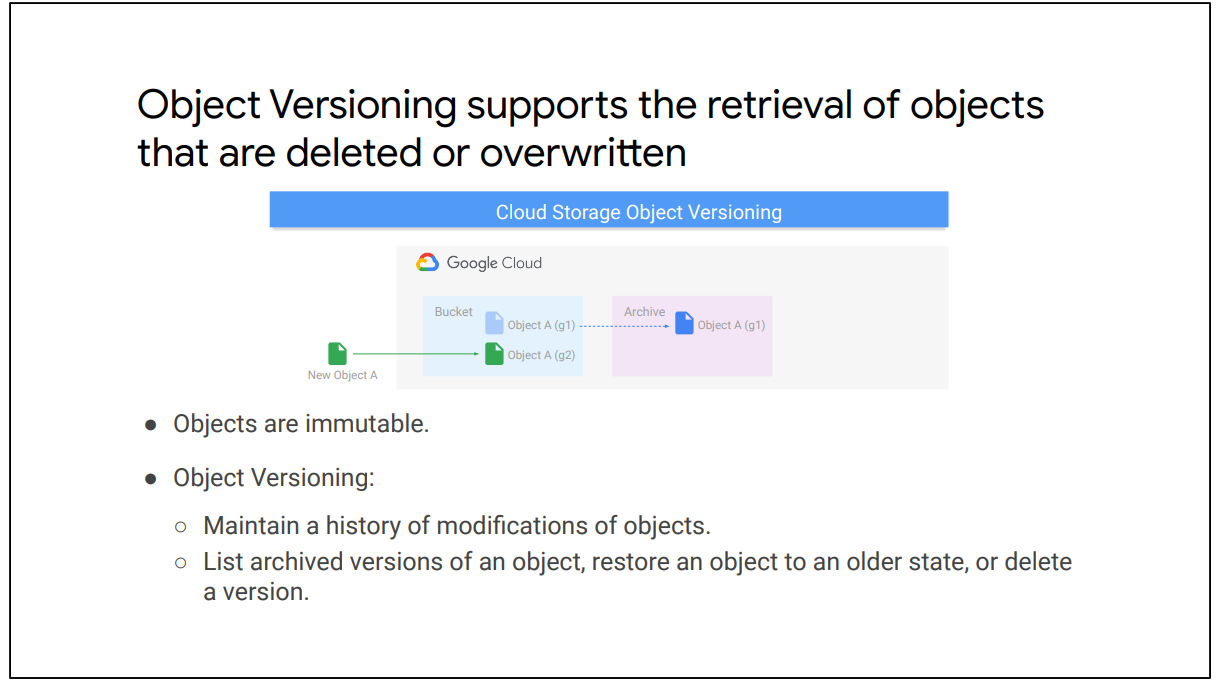
● For even more detailed control, signed URLs provide a cryptographic key that gives time-limited access to a bucket or object.

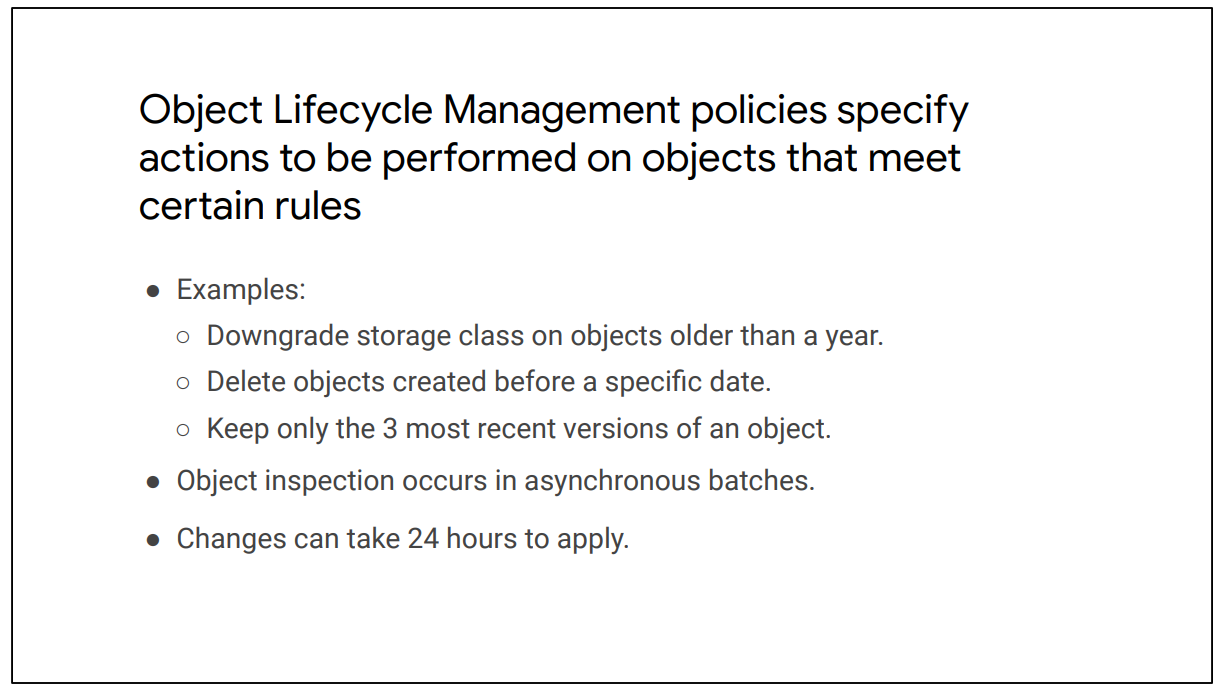
● Finally, a signed policy document further refines the control by determining what kind of file can be uploaded by someone with a signed URL. Let’s take a closer look at ACLs and signed URLs.

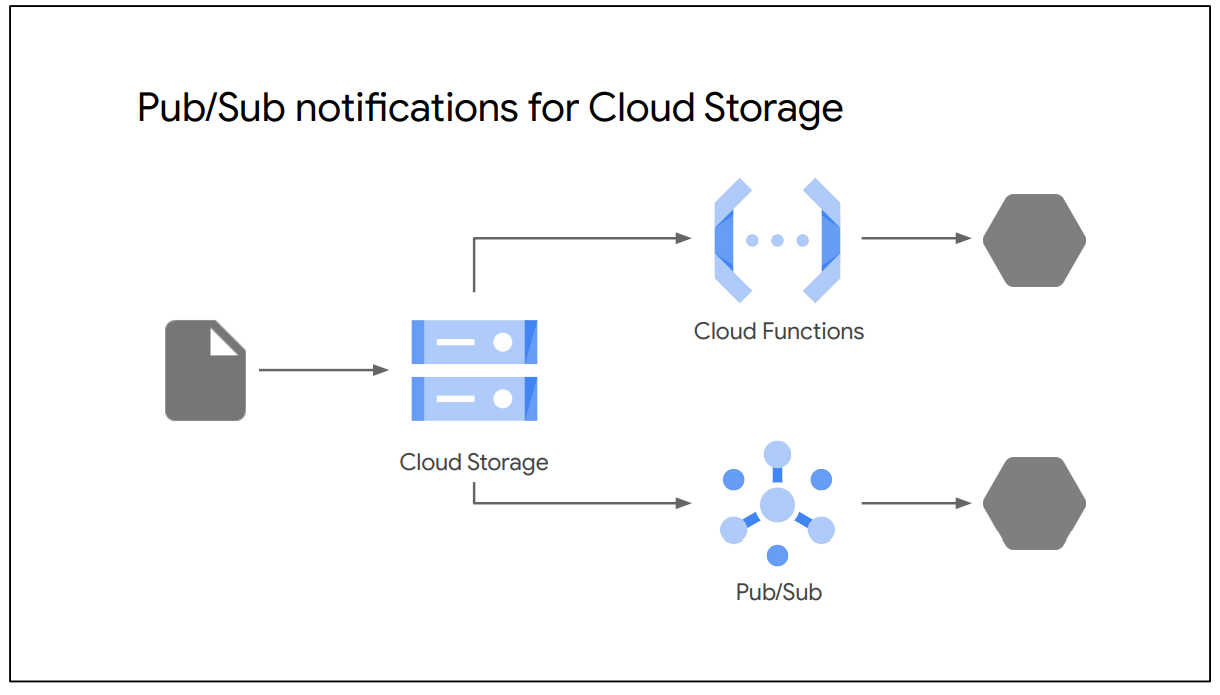


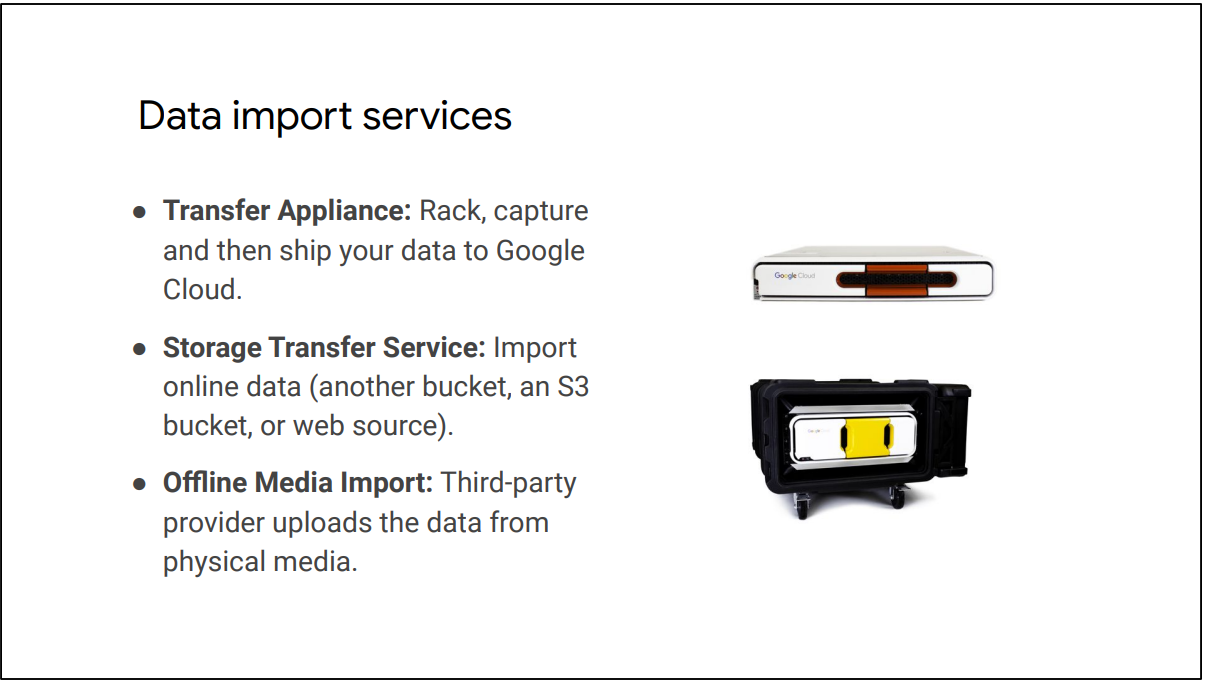


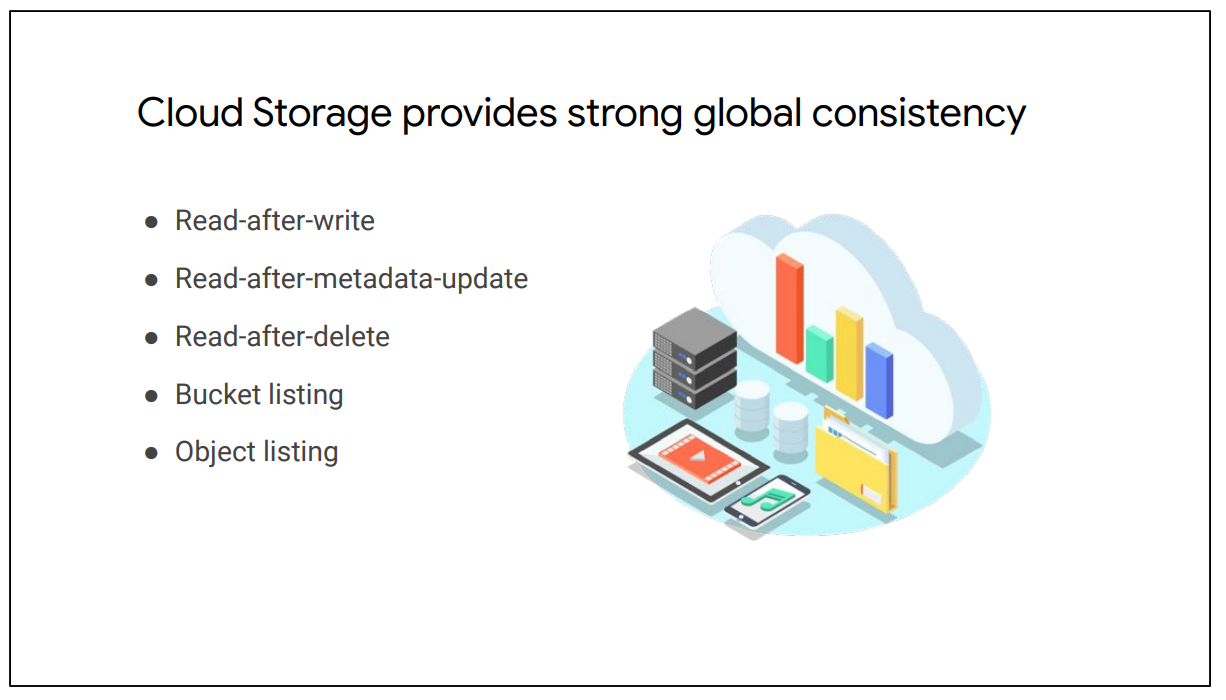


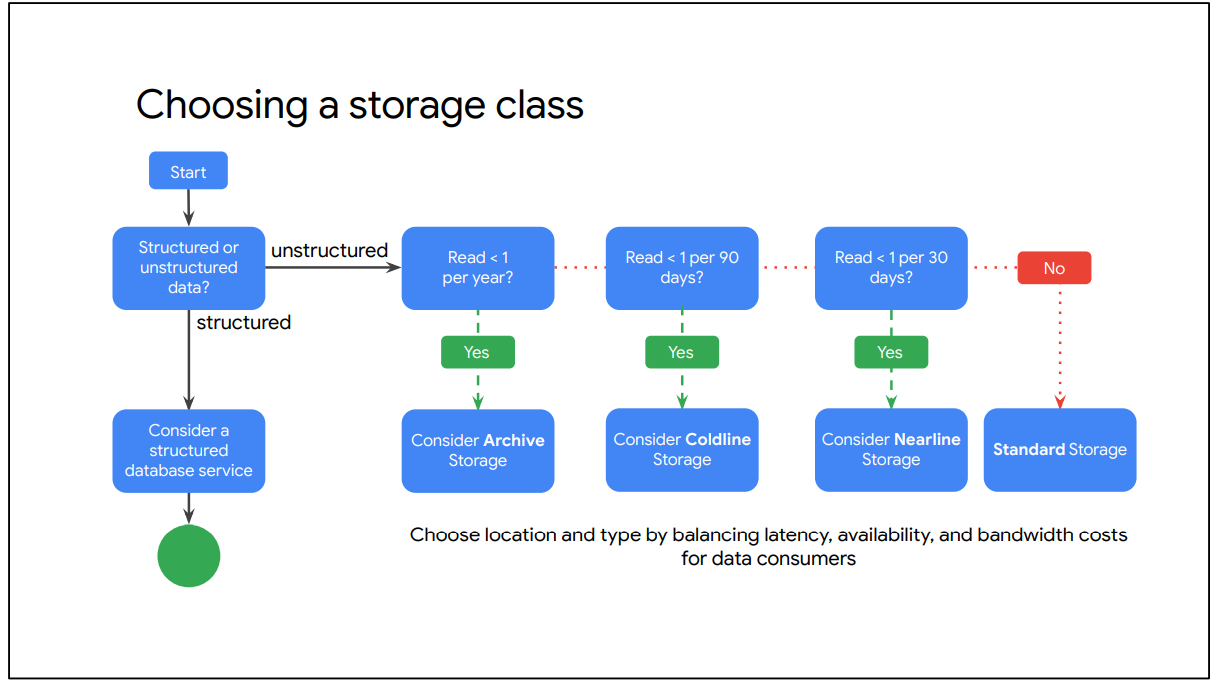


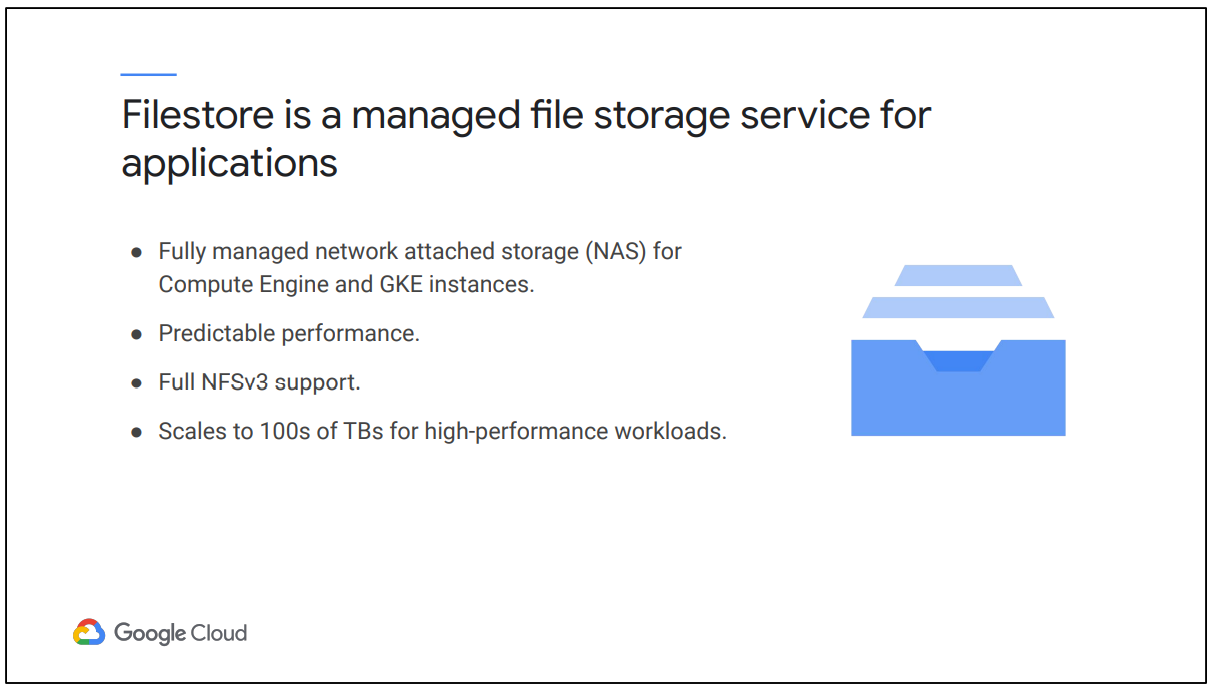












Filestore has many use cases.

● Using Filestore, you can expedite migration of enterprise applications. Many on-premises applications require a filesystem interface to data. As these applications continue to migrate to the cloud, Filestore can support a broad range of enterprise applications that need a shared filesystem. ● For media rendering, you can easily mount Filestore file shares on Compute Engine instances, enabling visual effects artists to collaborate on the same file share. As rendering workflows typically run across fleets (“render farms”) of compute machines, all of which mount a shared filesystem, Filestore and Compute Engine can scale to meet your job’s rendering needs.

● Electronic Design Automation (EDA) is all about data management. It requires the ability to batch workloads across thousands of cores and has large memory needs. Filestore offers the necessary capacity and scale to meet the needs of manufacturing customers doing intensive EDA and also makes sure files are universally accessible.

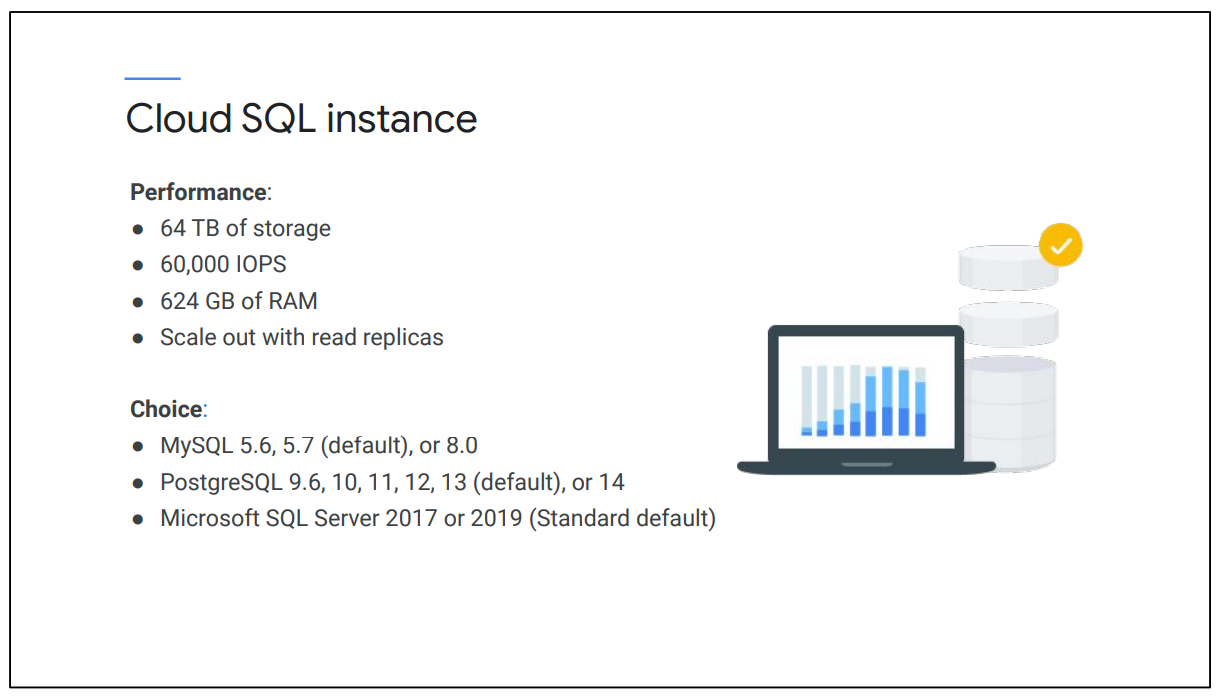
● Data analytics workloads include compute complex financial models or analysis of environmental data. These workloads are latency sensitive. Filestore offers low latency for file operations and, as capacity or performance needs change, you can easily grow or shrink your instances as needed. As a persistent and shareable storage layer, Filestore enables immediate access to data for high-performance, smart analytics without the need to lose valuable time on loading and off-loading data to clients’ drives.

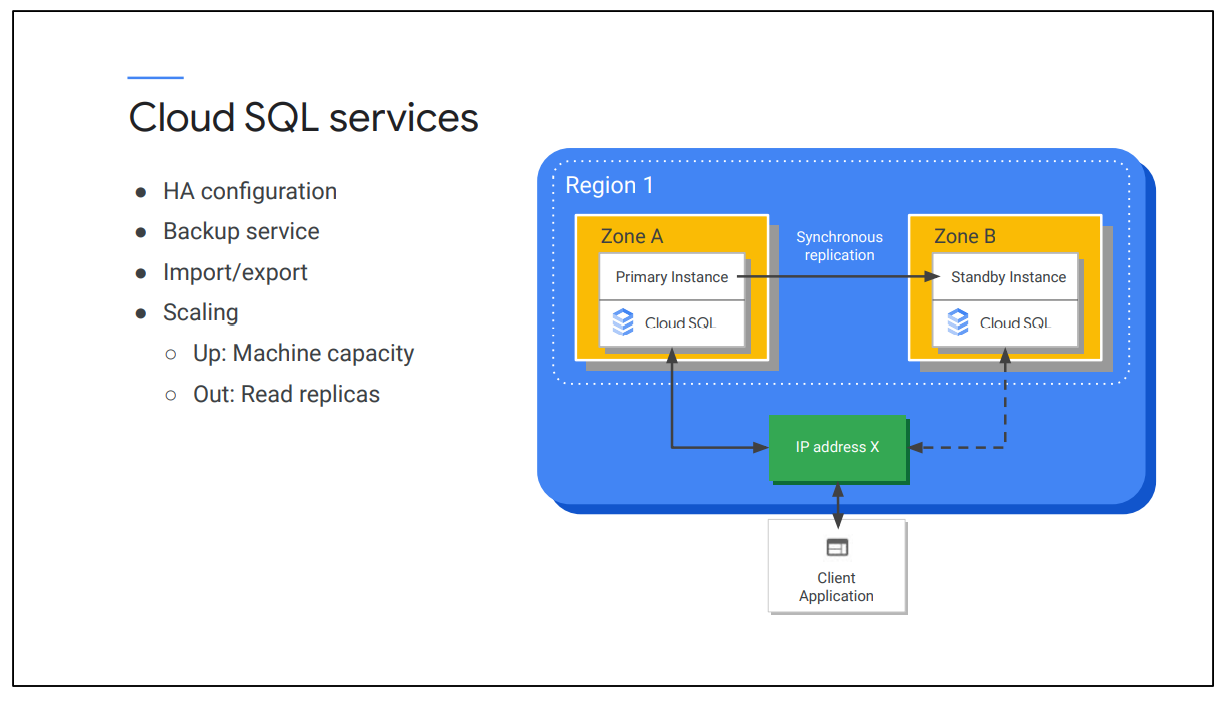
● Genome sequencing requires an incredible amount of raw data, on the order

● of billions of data points per person. This type of analysis requires speed, scalability, and security. Filestore meets the needs of companies and research institutions performing scientific research, while also offering predictable prices for the performance.

● Web developers and large hosting providers also rely on Filestore to manage and serve web content, including needs such as WordPress hosting.







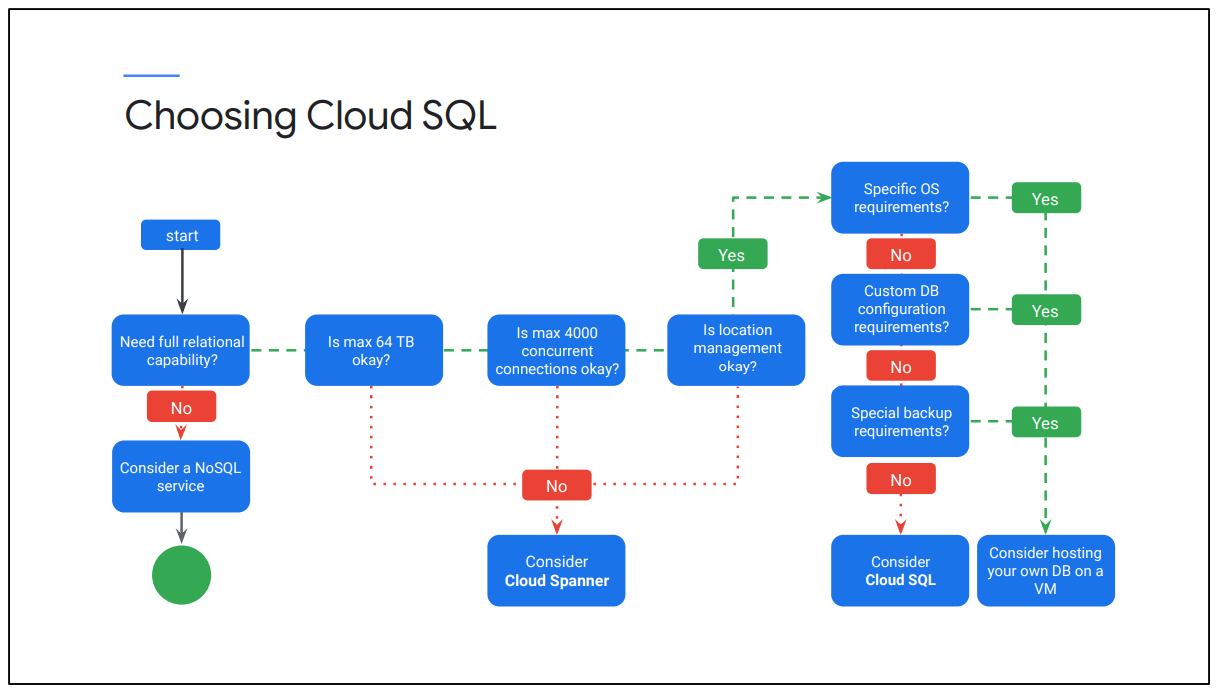
Let’s focus on some other services provided by Cloud SQL:

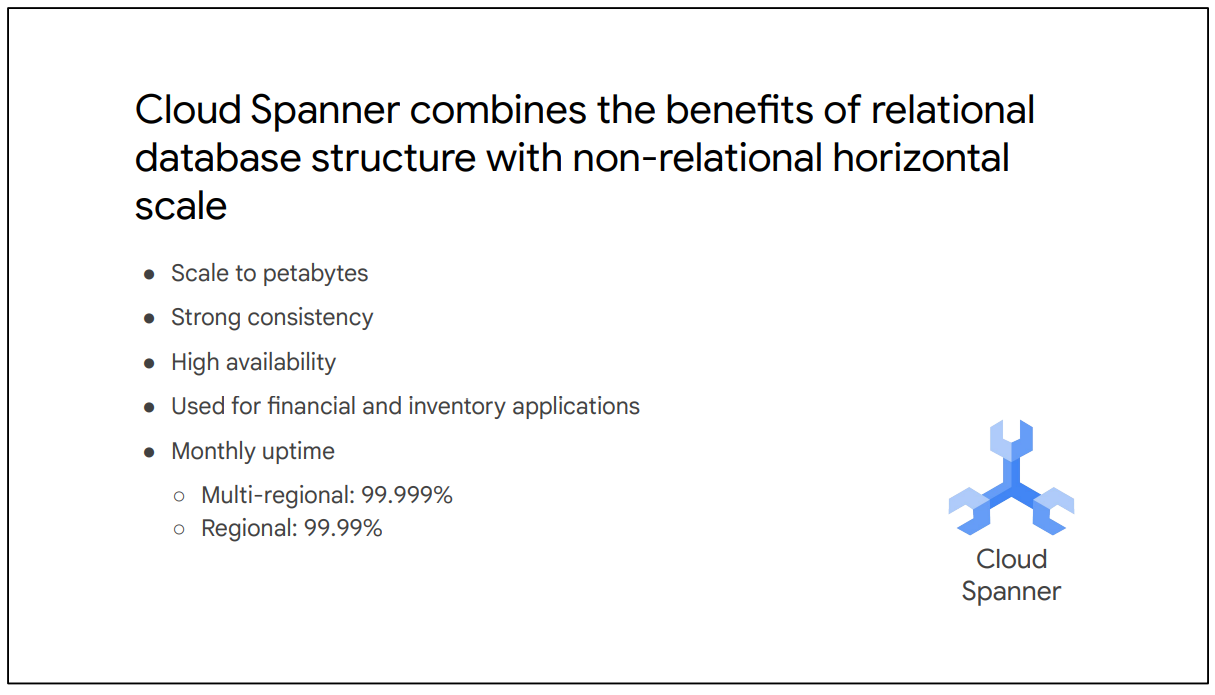
● In HA configuration, within a regional instance, the configuration is made up of a primary instance and a standby instance. Through synchronous replication to each zone's persistent disk, all writes made to the primary instance are replicated to disks in both zones before a transaction is reported as committed. In the event of an instance or zone failure, the persistent disk is attached to the standby instance, and it becomes the new primary instance. Users are then rerouted to the new primary. This process is called a failover.

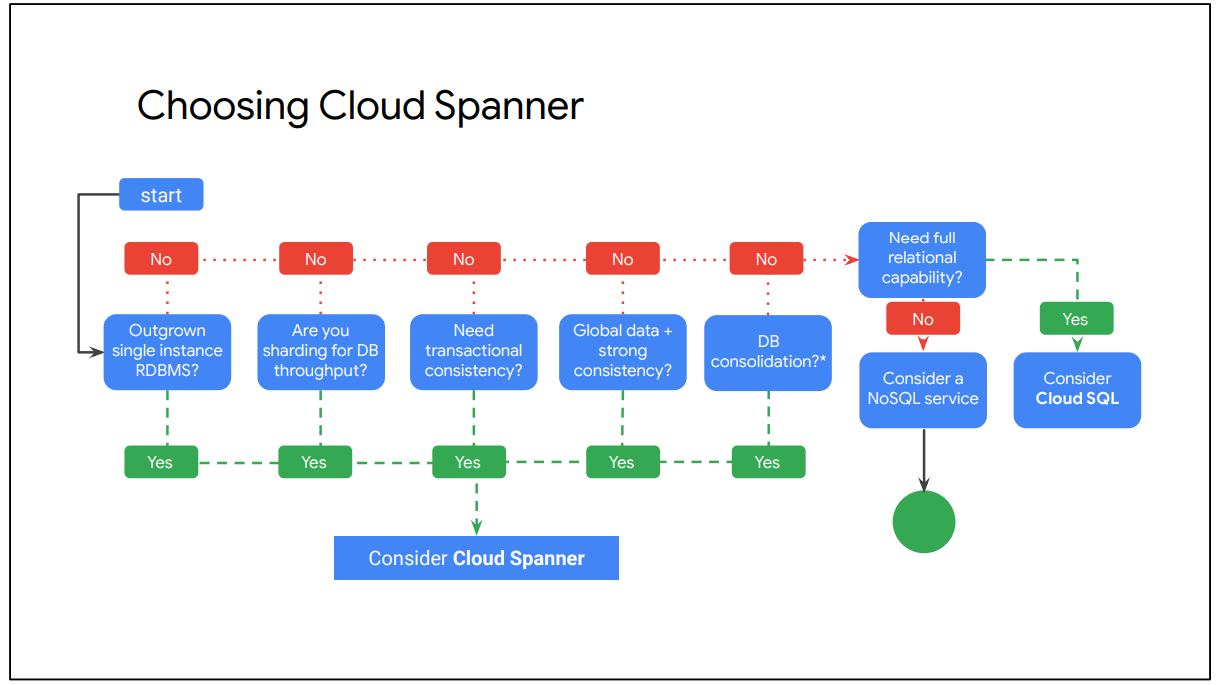
● Cloud SQL also provides automated and on-demand backups with point-in-time recovery.

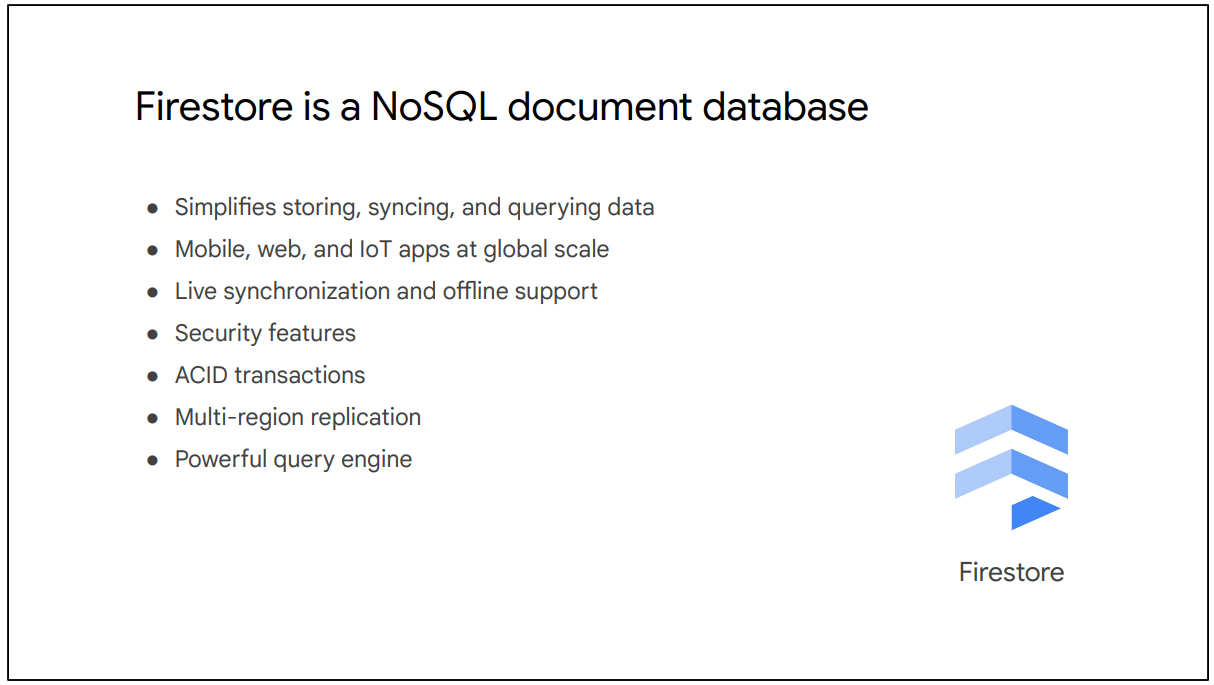
● You can import and export databases using mysqldump, or import and export CSV files.

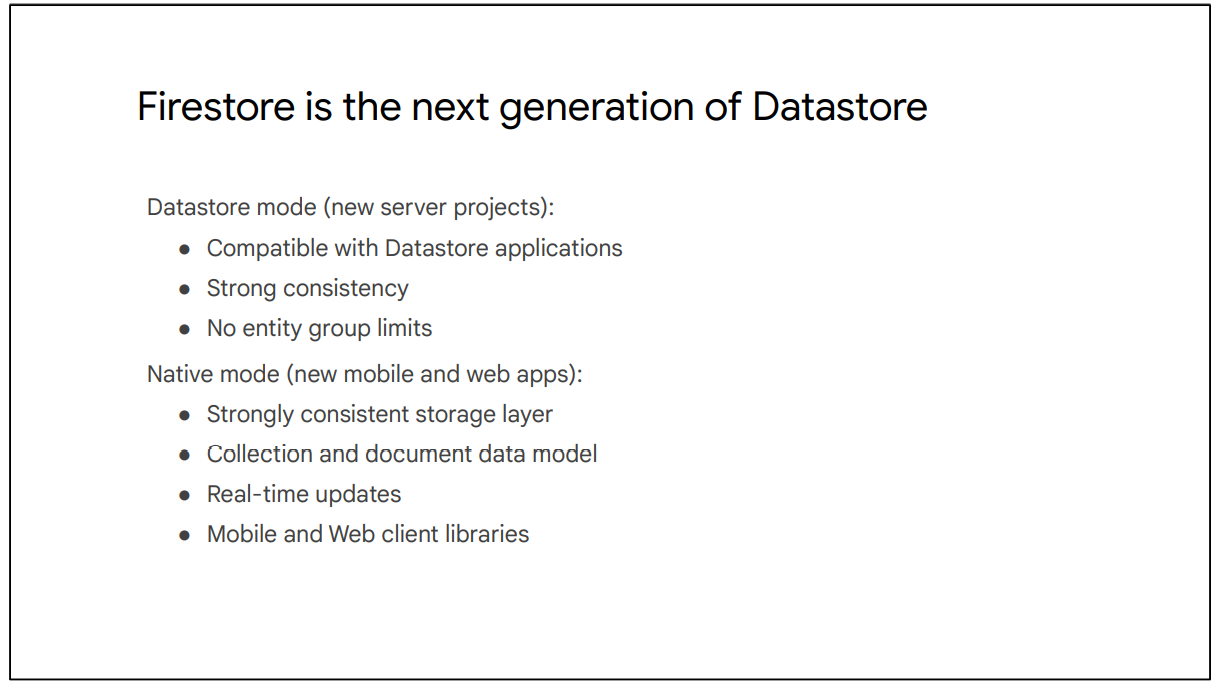
● Cloud SQL can also scale up, which does require a machine restart or scale out using read replicas.

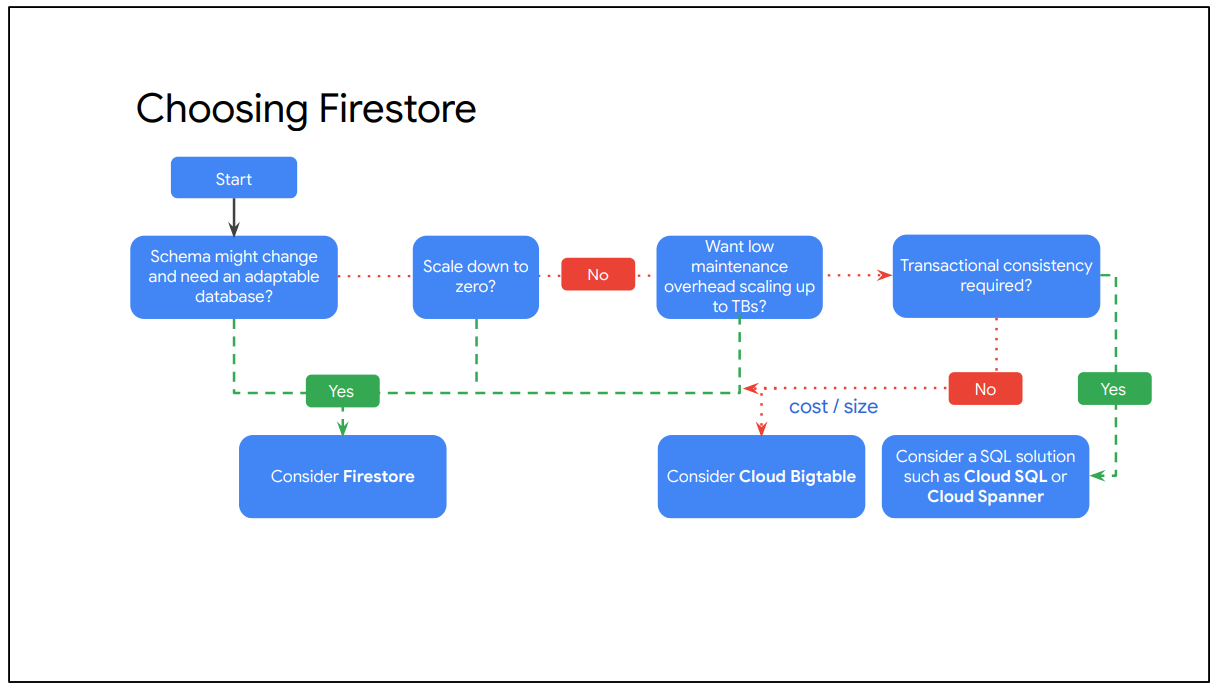


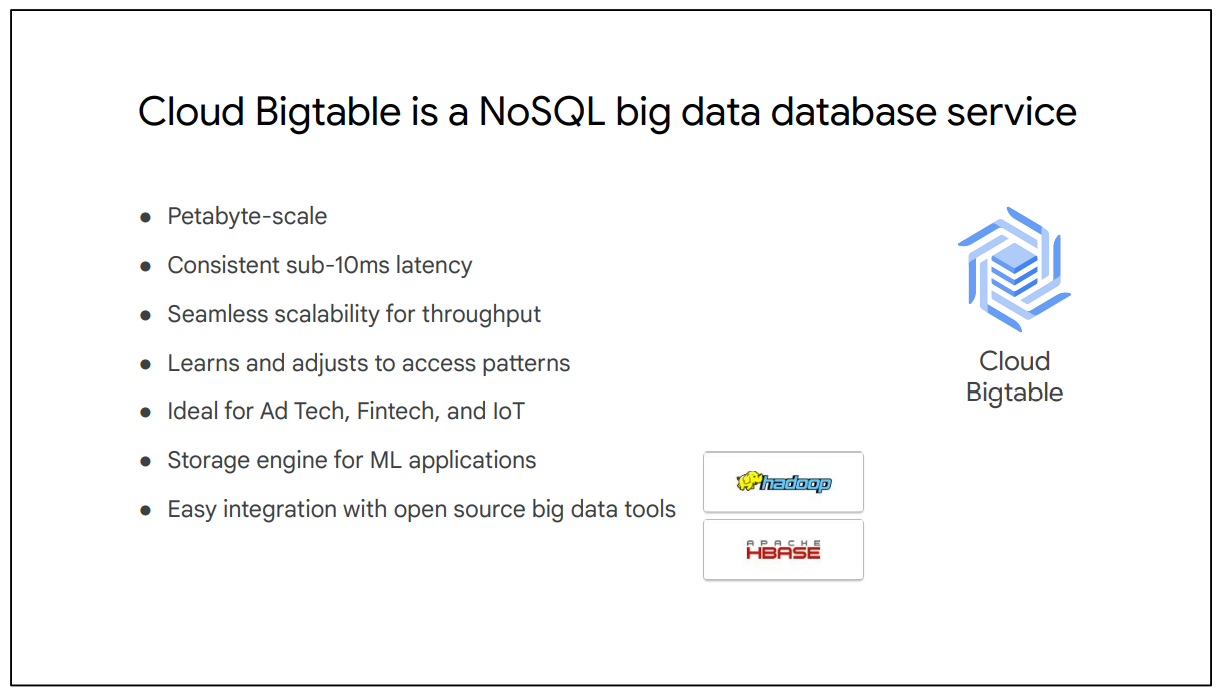


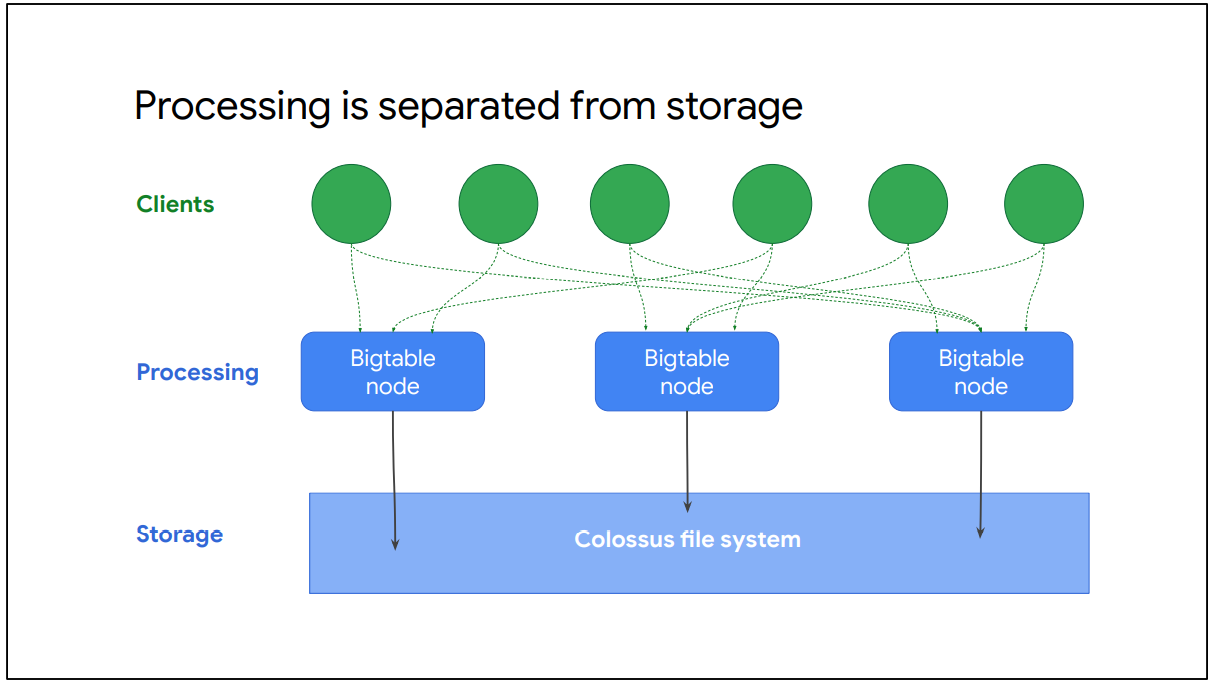




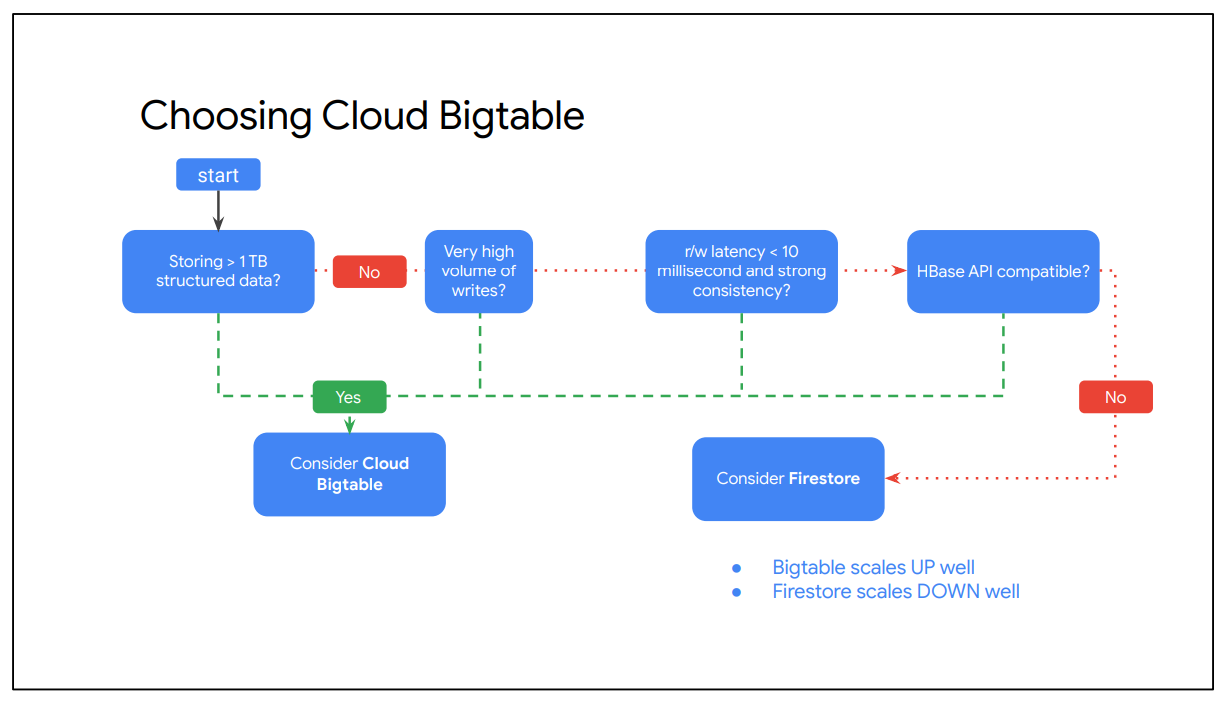


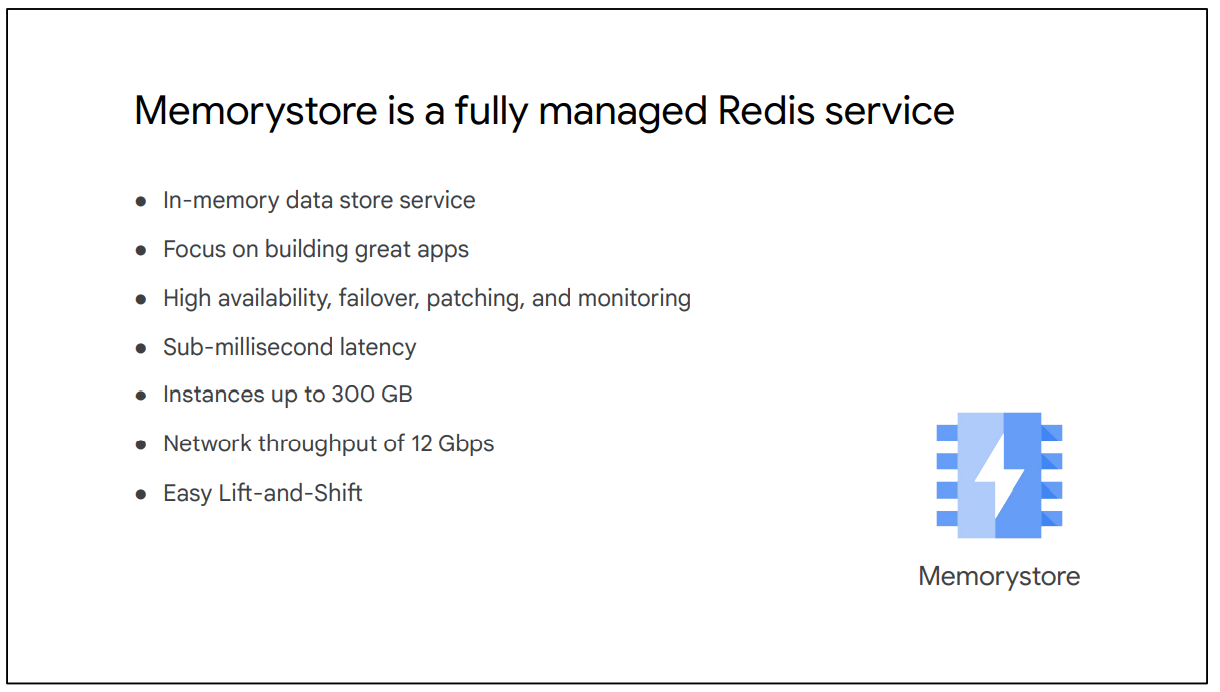






This diagram shows a simplified version of Cloud Bigtable’s overall architecture. It illustrates that processing, which is done through a front-end server pool and nodes, is handled separately from the storage. A Cloud Bigtable table is sharded into blocks of contiguous rows, called tablets, to help balance the workload of queries. Tablets are similar to HBase regions, for those of you who have used the HBase API. Tablets are stored on Colossus, which is Google's file system, in SSTable format. An SSTable provides a persistent, ordered immutable map from keys to values, where both keys and values are arbitrary byte strings.

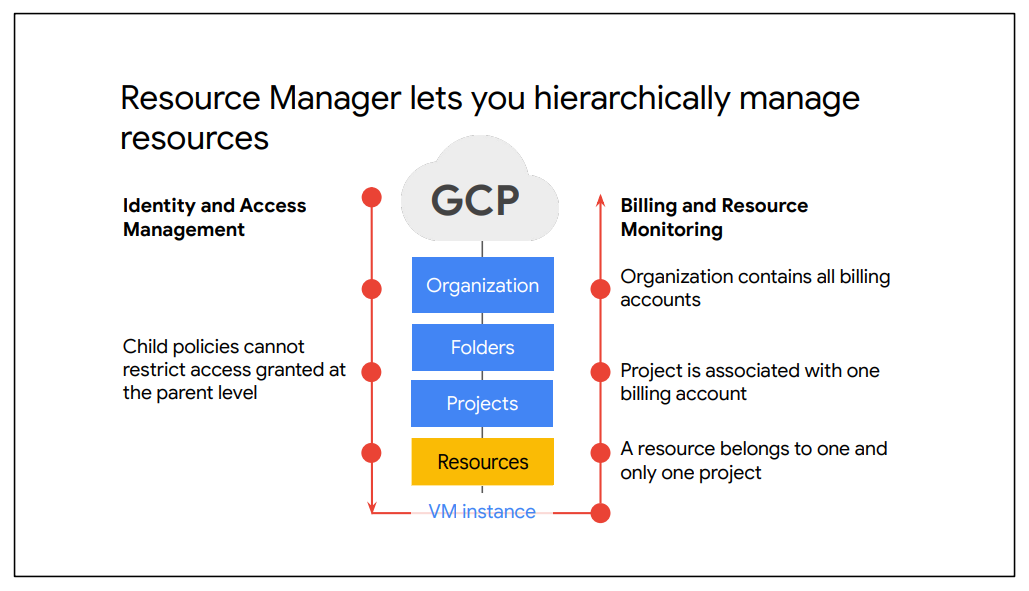


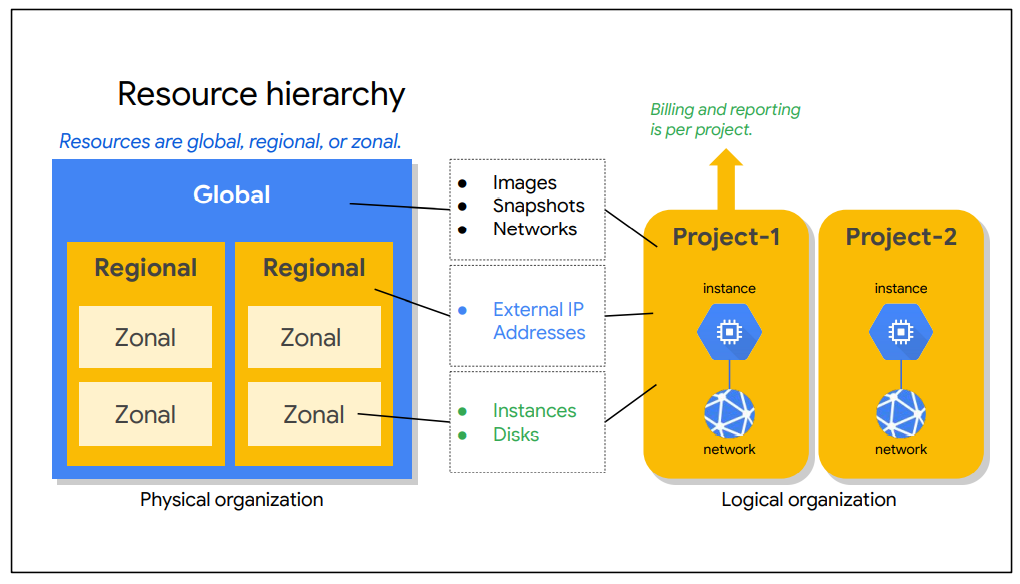


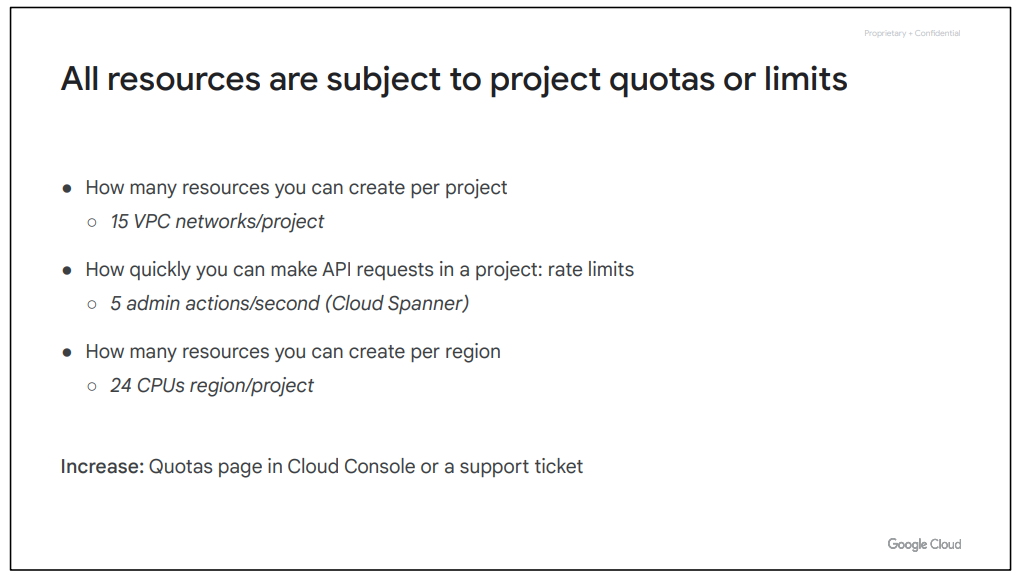
**3.**

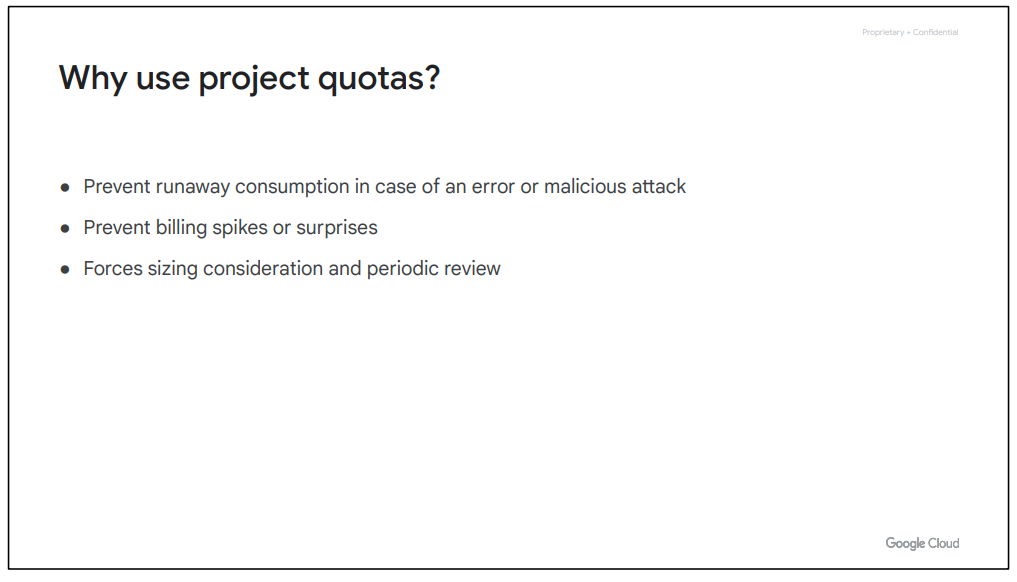
**Resource Manager**

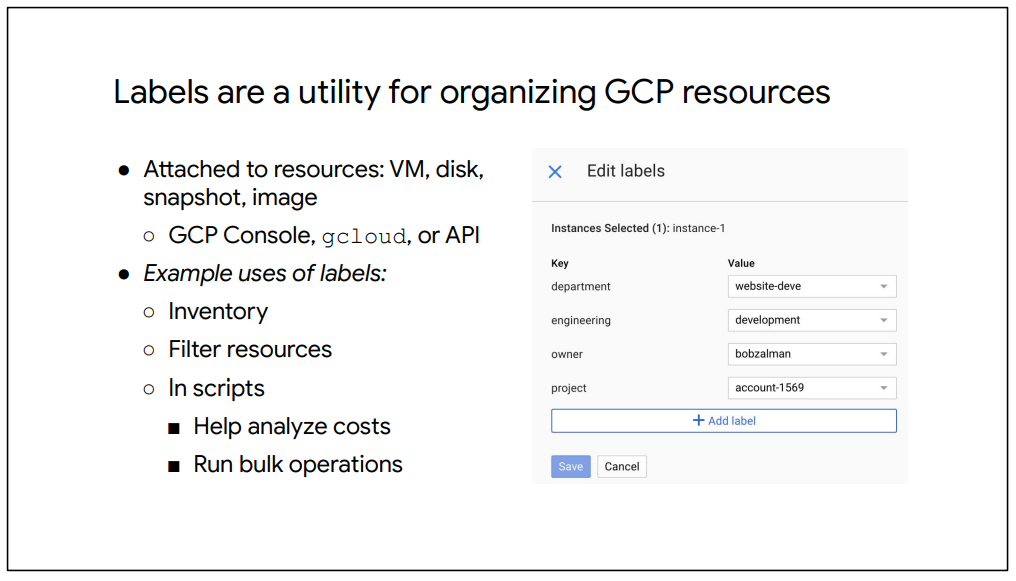
The resource manager lets you hierarchically manage resources by project, folder, and organization. This should sound familiar because we covered it in the Cloud IAM module. Let me refresh your memory: Policies contain a set of roles and members, and policies are set on resources. These resources inherit policies from their parent, as we can see on the left. Therefore, resource policies are a union of parent and resource.

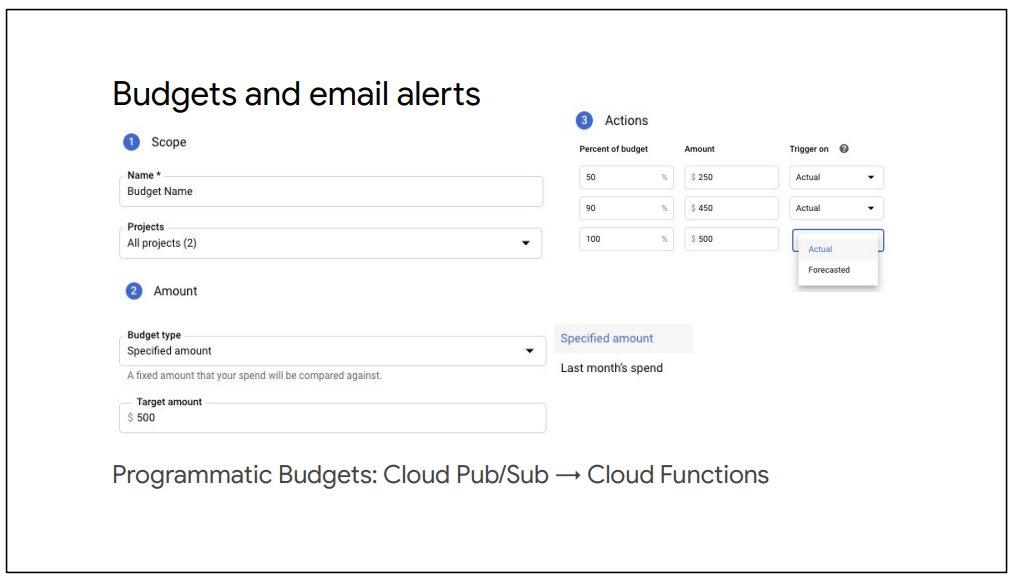


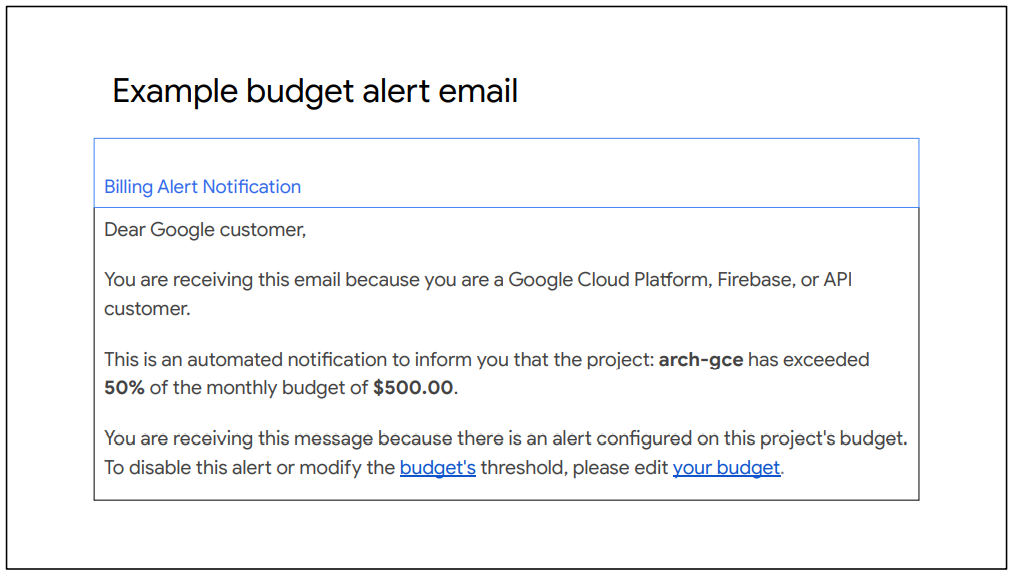


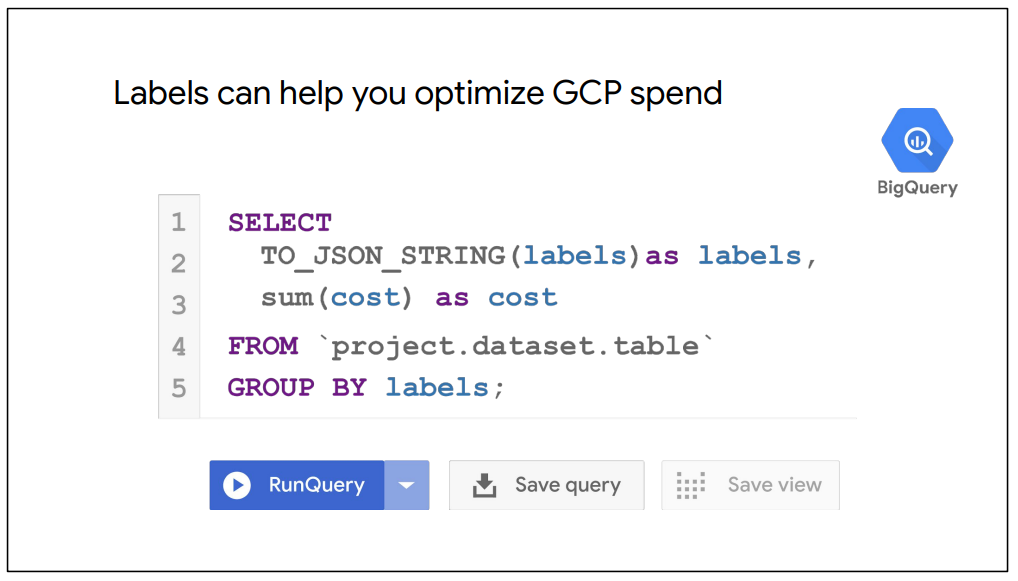




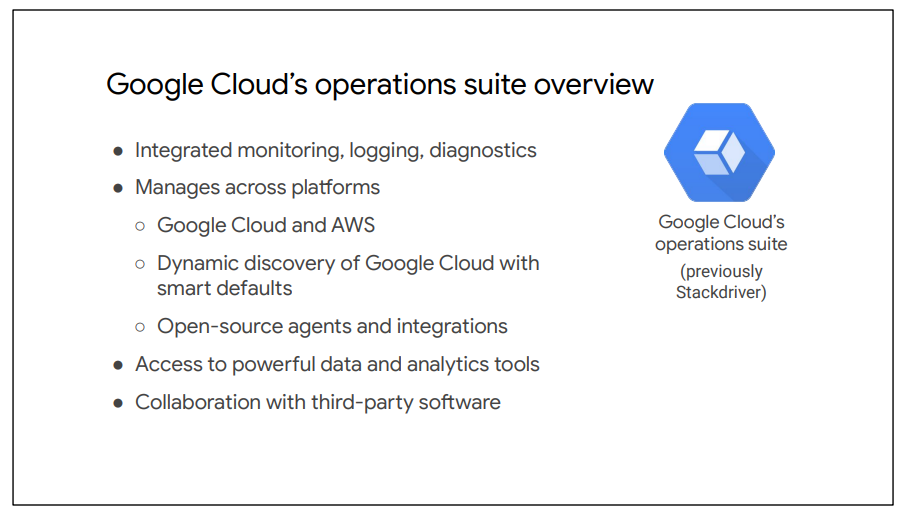


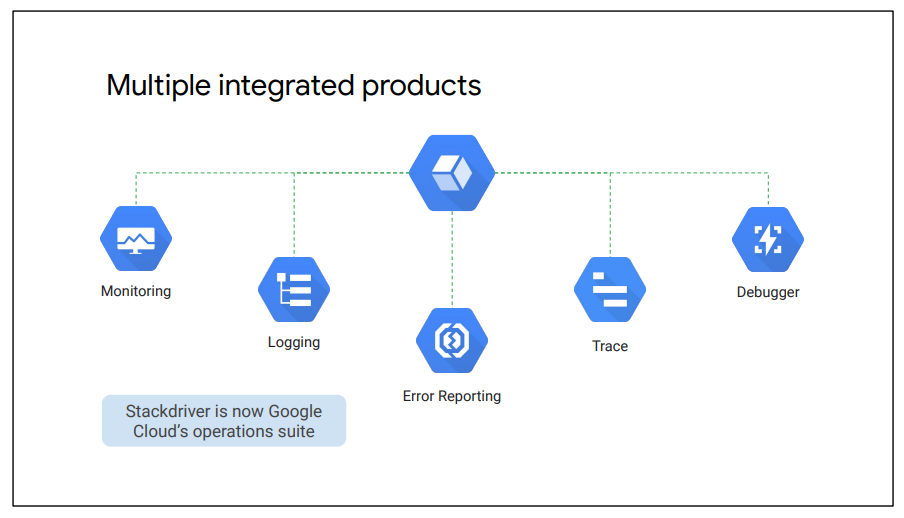


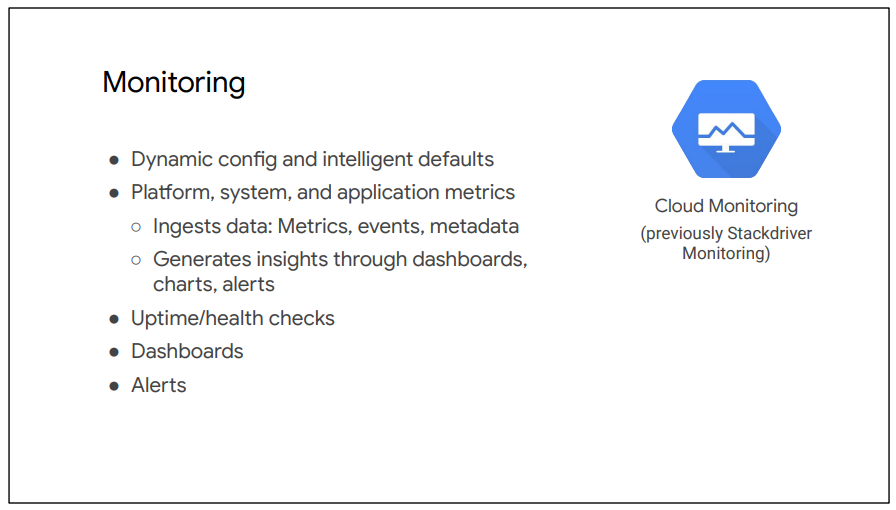


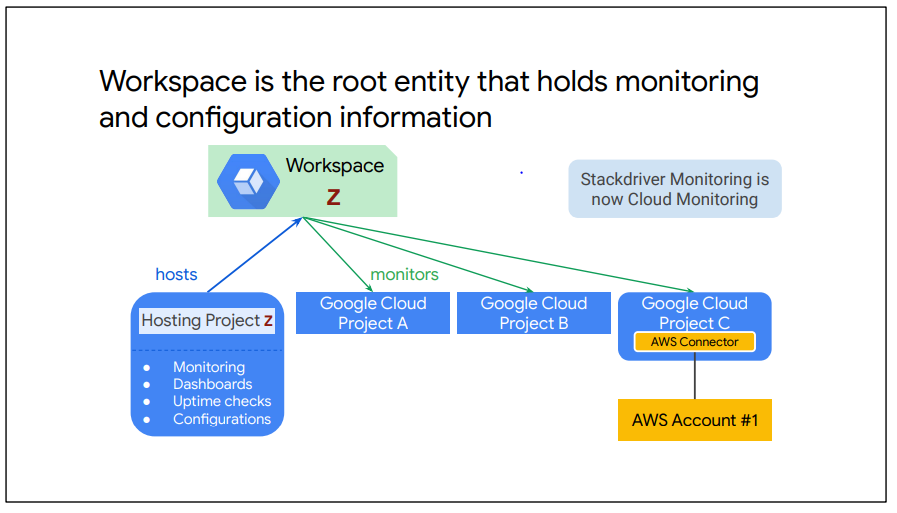


**4.**









Cloud Monitoring allows you to create custom dashboards that contain charts of the metrics that you want to monitor. For example, you can create charts that display your instances’ CPU utilization, the packets or bytes sent and received by those instances, and the packets or bytes dropped by the firewall of those instances.

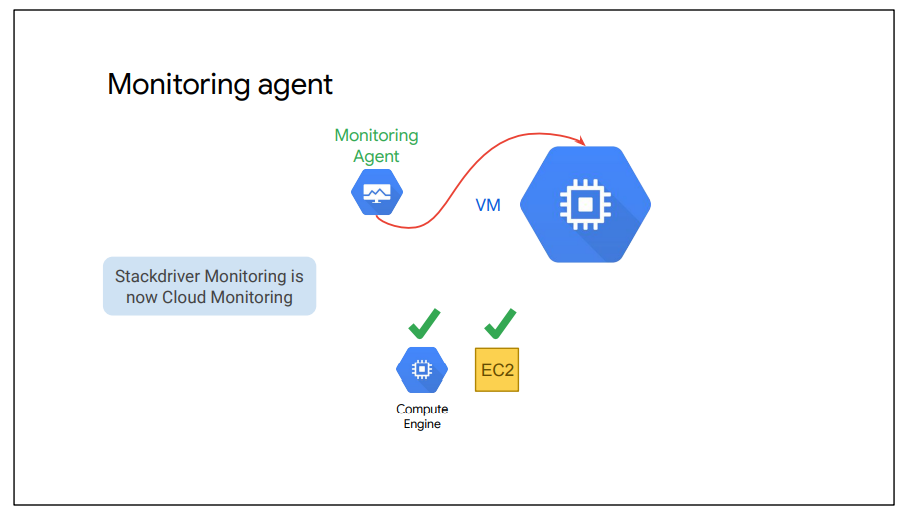
Now, although charts are extremely useful, they can only provide insight while someone is looking at them. But what if your server goes down in the middle of the night or over the weekend? Do you expect someone to always look at dashboards to determine whether your servers are available or have enough capacity or bandwidth?

If not, you want to create alerting policies that notify you when specific conditions are met.

For example, as shown on this slide, you can create an alerting policy when the network egress of your VM instance goes above a certain threshold for a specific timeframe. When this condition is met, you or someone else can be automatically notified through email, SMS, or other channels in order to troubleshoot this issue.

Uptime checks can be configured to test the availability of your public services from locations around the world, as you can see on this slide. The type of uptime check can be set to HTTP, HTTPS, or TCP. The resource to be checked can be an App Engine application, a Compute Engine instance, a URL of a host, or an AWS instance or load balancer.

For each uptime check, you can create an alerting policy and view the latency of each global location.



Cloud Monitoring can access some metrics without the Monitoring agent, including CPU utilization, some disk traffic metrics, network traffic, and uptime information.

However, to access additional system resources and application services, you should install the Monitoring agent. The Monitoring agent is supported for Compute Engine and EC2 instances.

