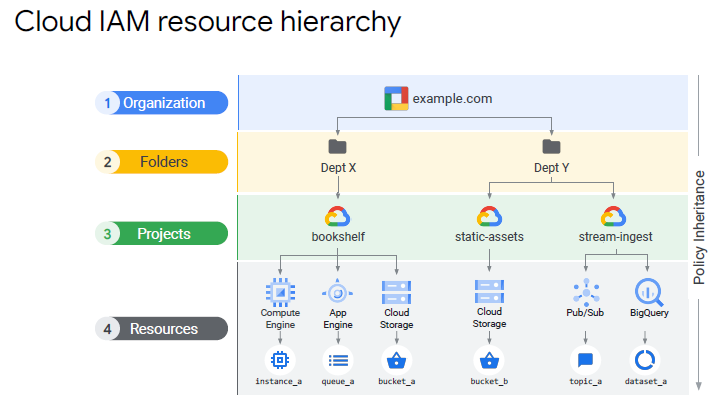
**Essential Google Cloud Infrastructure: Core Services**

**Cloud IAM**

* Identity and Access Management
* Who can do what on which resource
  + Who – person, group or application
  + What – specific privileges or actions
  + Resource – any GC service



Policy

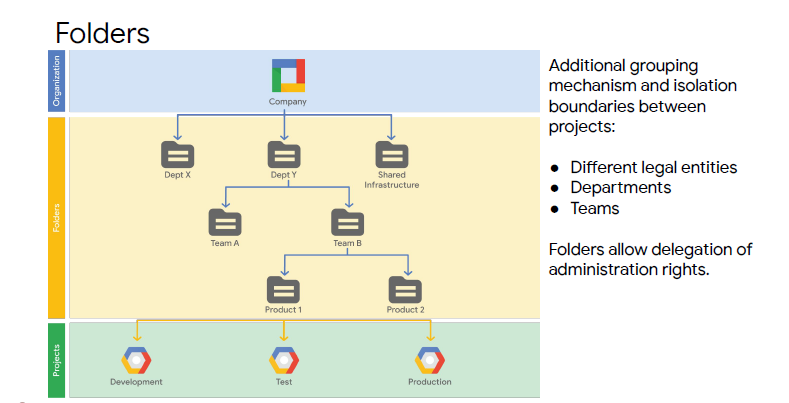
* collection of access statements attached to a resource.
* Each policy contains a set of role and role members, resources inherit policies from their parents.
* Less restrictive parent policy overrises a more restrictive resource policy.
* Child policy cannot restrict access granted at the parent level (if you have editor role at dept X and viewer role at bookshelf project, you still have editor access to bookshelf project)
* Policy of least privilege – select the smallest scope necessary for task to reduce risk exposure.

IAM conditions

* Define and enforce conditional attribute
* Grant resource access to identities if configured conditions are met
* E.g. temporary access in the event of production being down

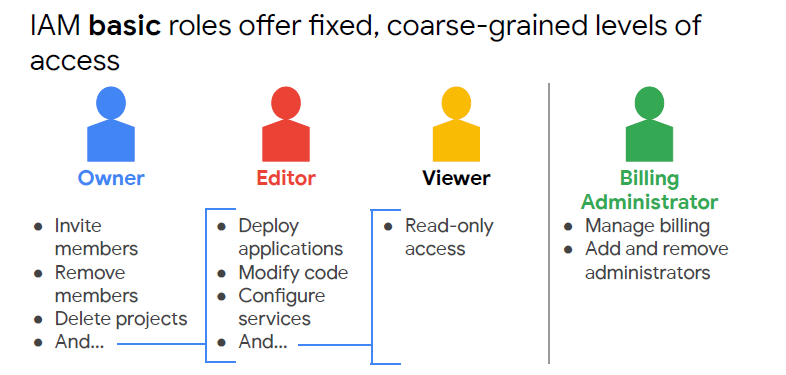
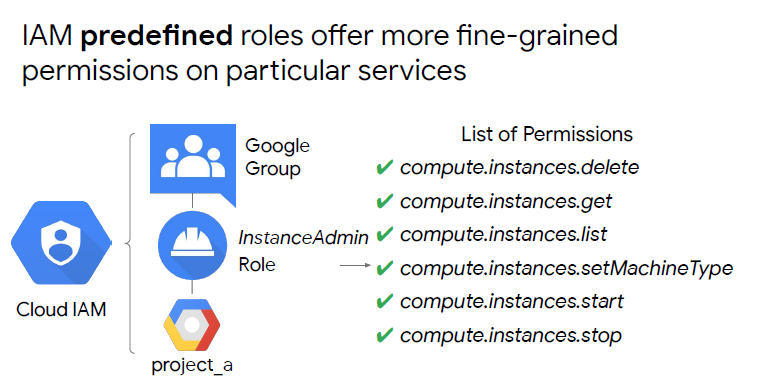
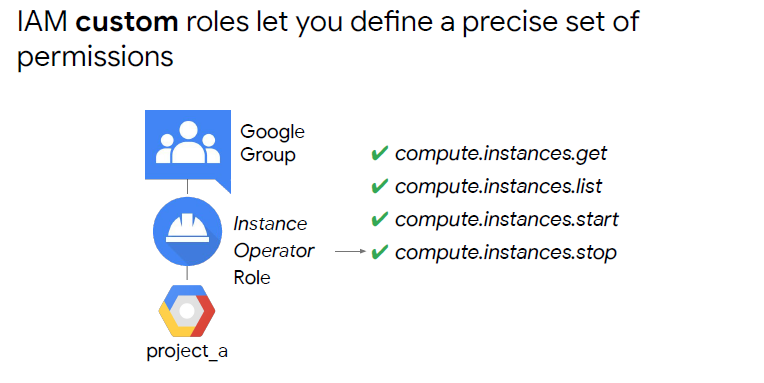
Organisation node

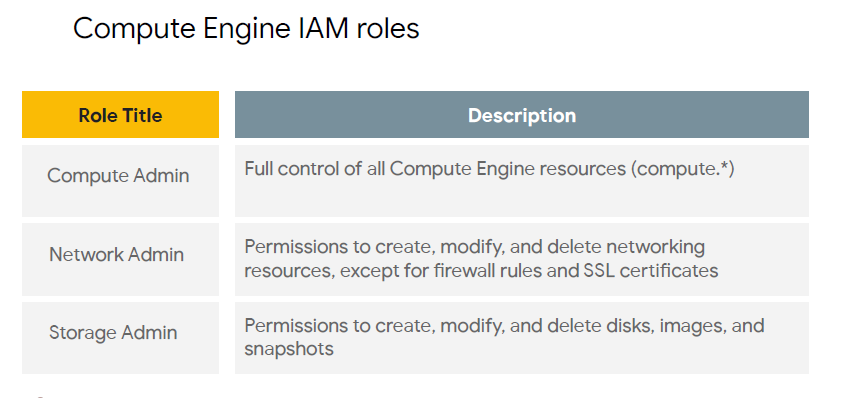
* Root node inf GCP resources
* Organisation Role – control over all cloud resources (useful for auditing)
* Project creator – project creation and control over who can create projects



^Folders can be considered as suborganisations within the organisation

IAM roles

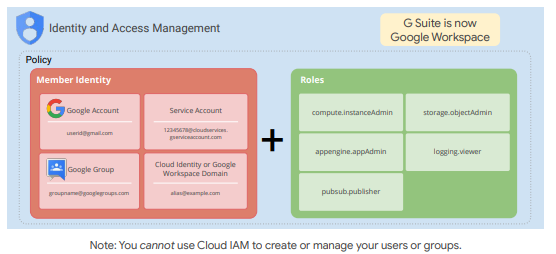
* Basic roles – applied to project and affect all resources in that project (owner, editor, viewer) 
* Predefined roles – defined where the predefined roles can be applied. Provides granular access to specific resources and prevents unwanted access to other resources. Have a collection of permissions
* Custom roles – can define a precise set of permissions



Members

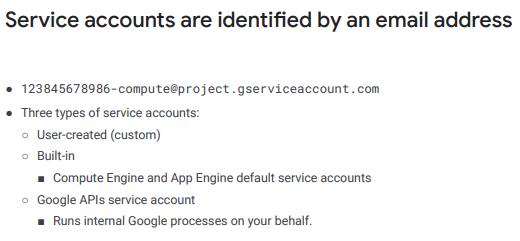
5 types:

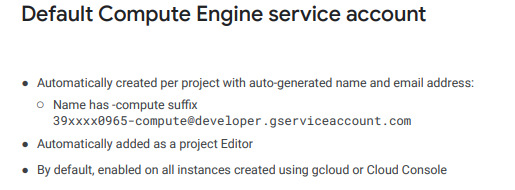
1. Google accounts – person who interacts with GCP
2. Service accounts – account belonging to application
3. Google groups – named collection of google and service accounts
4. Cloud identity domains – manage users or groups using admin console but do not pay for or use gsuite collaboration tools such as gmail, docs, drive, etc. Comes in free or premium, with the premium adding mobile device management
5. Google Workspace domains – organisations internet domain name

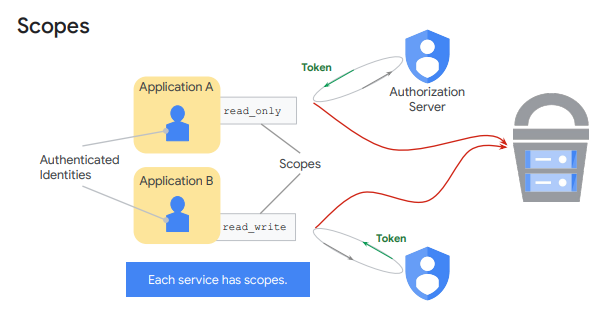


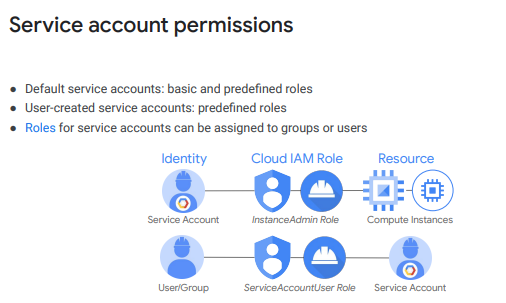
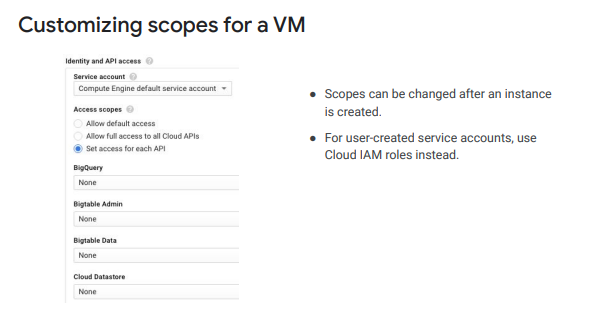
Service Accounts

* Belongs to application
* Identity for carrying out server to server interactions
* Auto acquire tokens with credentials
* Tokens grant access to any service API in your project and other services you have granted access too.

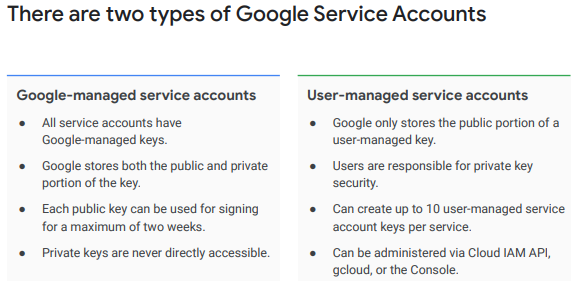


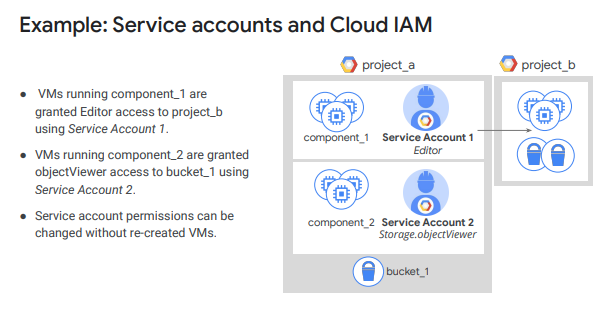


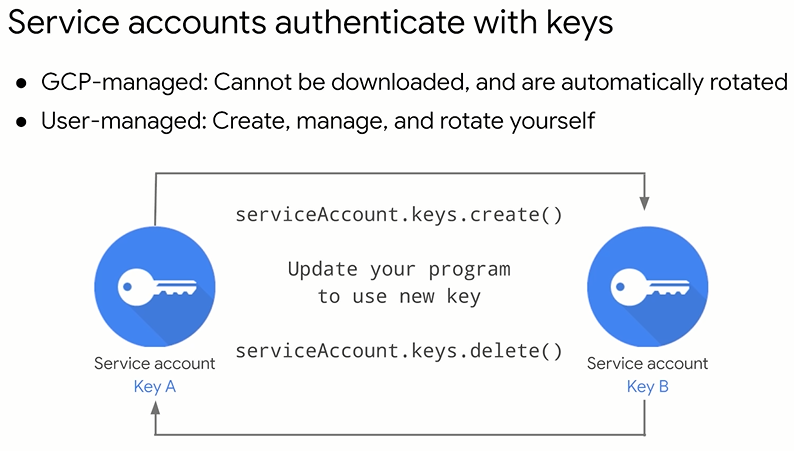


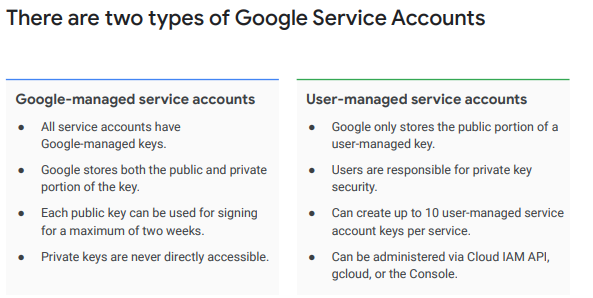


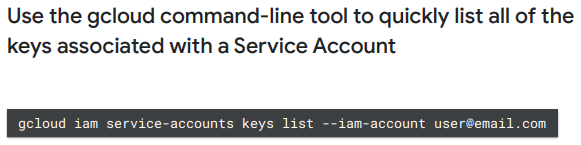
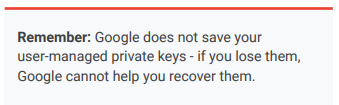
Can set up a service account and then use it as a resource by assigning users ServiceAccountUser role. The user can then act as that service account. The user can access all the resources that the service account has access to.



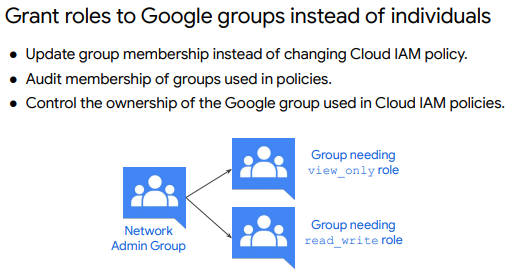








Cloud IAM best practices

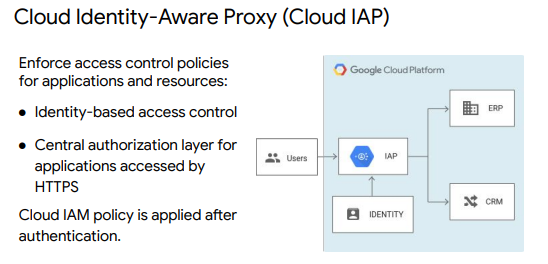
* Use projects to group resources that share the same trust boundary
* Check the policy granted on each resources and make sure you understand the inheritance
* Use “principles of least privilege” when granting roles
* Audit policies in Cloud Audit logs: *setiamploicy*
* Audit membership of groups used in policies
* Grant roles to Google Groups instead of individuals

Service Account Best Practices

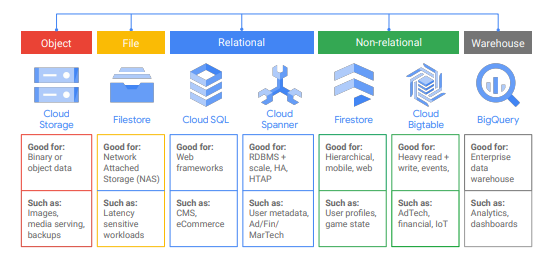
* Be careful granting serviceAccountUser role
* Give service accounts a display name that clearly identifies its purpose
* Establish naming convention for service accounts
* Establish key rotation policies and method
* Audit with serviceAccount.keys.list()

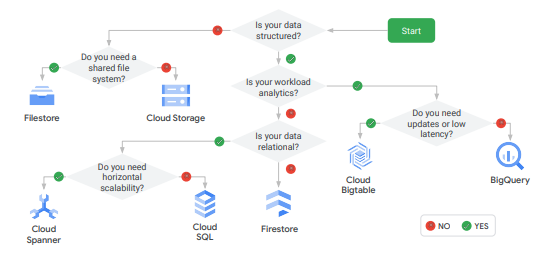
Cloud Identity-Aware Proxy (Cloud IAP)

* Central authorisation layer for applications accessed by https
* Can then use access control model instead of relying on network-level firewalls
* Applications and resources protected by Cloud IAP can only be accessed through the proxy by users and groups with the correct cloud IAM role



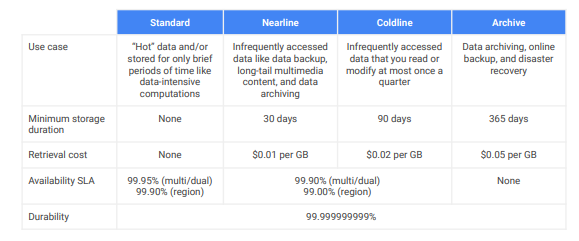
**Storage and Database Services**

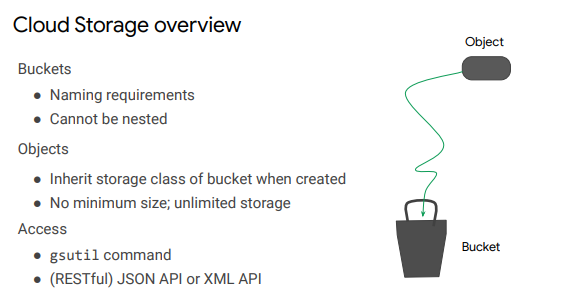


****

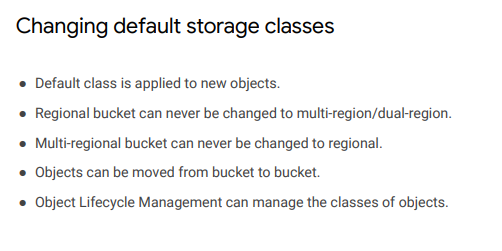
Cloud storage

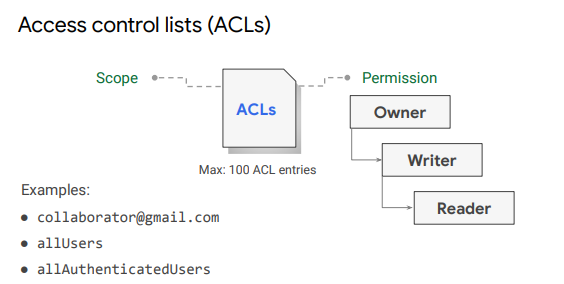
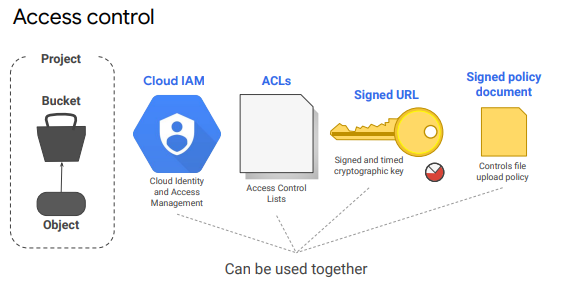
* Object storage service
* World wide storage and retrieval of any amount of data at any time
* Examples, website content, storing data for archiving and disaster recovery, distributing large data objects to users via direct download etc
* Scalable to exabytes
* Time to first byte in miliseconds
* Very high availability across all storage classes
* Single API across storage classes



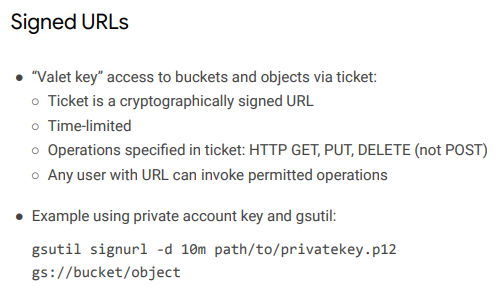


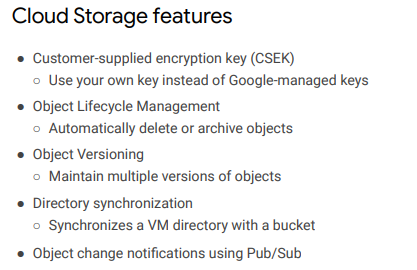
^ as such isn’t a file system so hard to index files. Objects are accessed via a specific url

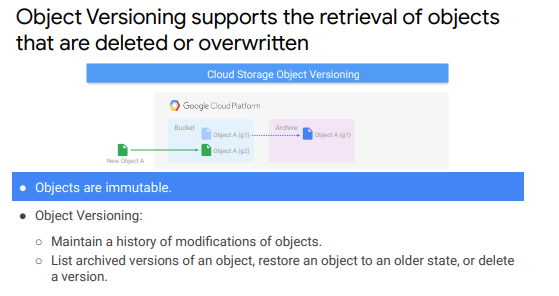




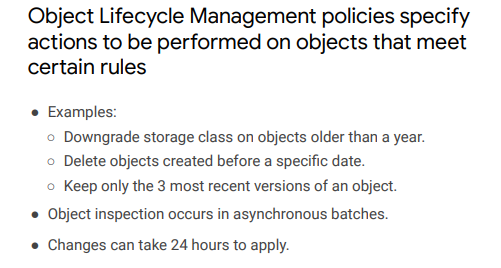
^ who has access and what level of access. allUsers is anyone on the internet (with or without Google Account). allAuthenticatedUsers anyone authenticated with Google account

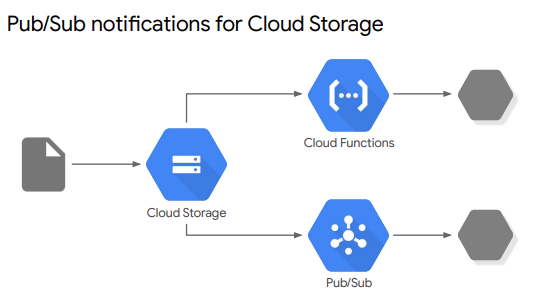




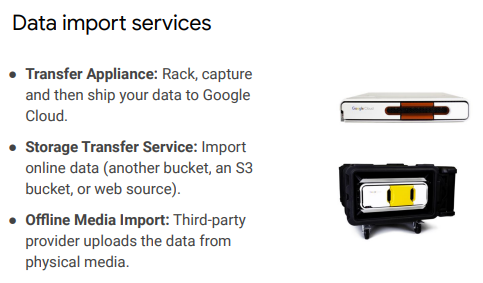


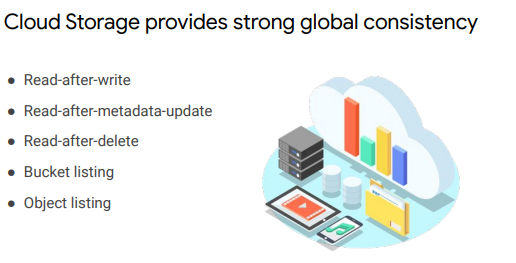
^archived object given a generation number (g1, g2) to identify its history. Turning versioning off leaves existing archived objects in place



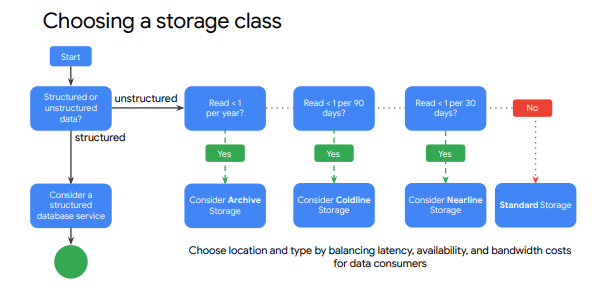


^pub/sub is the recommended way to track changes to object in Cloud Storage because they’re faster, more flexible, easier to setup and more cost effective



^Transfer Appliance is up to 1 petabyte of data

^ After uploading an object and receiving a success response, the object is immediately available for download and metadata operations. Prevent getting a 404 error. The opposite applied for deletion.

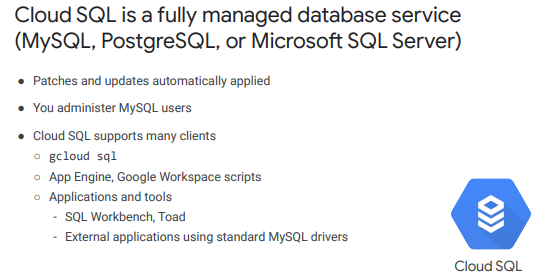


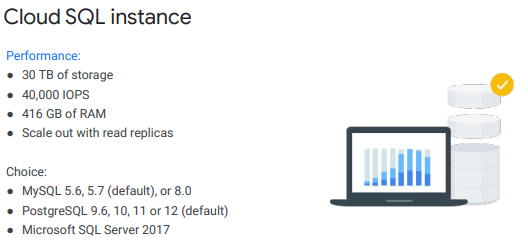
* Use a region to help optimize latency and network bandwidth for data consumers, such as analytics pipelines, that are grouped in the same region
* Use a dual-region when you want similar performance advantages as regions, but also want the higher availability that comes with being geo-redundant.
* Use a multi-region when you want to serve content to data consumers that are outside of the Google network and distributed across large geographic areas, or when you want the higher data availability that comes with being geo-redundant.

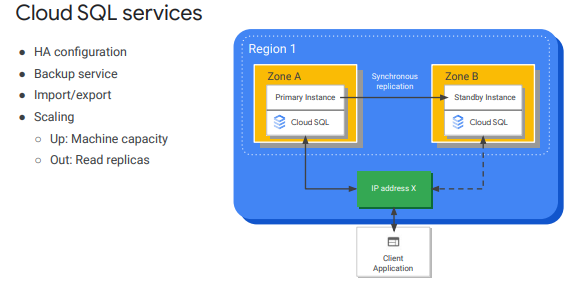
Filestore

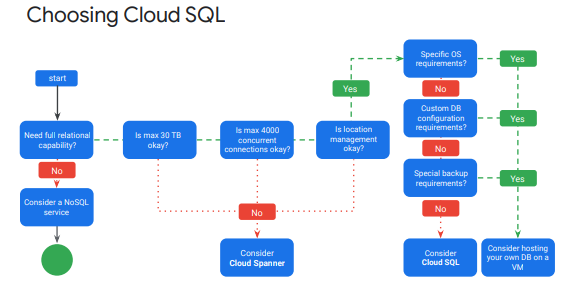
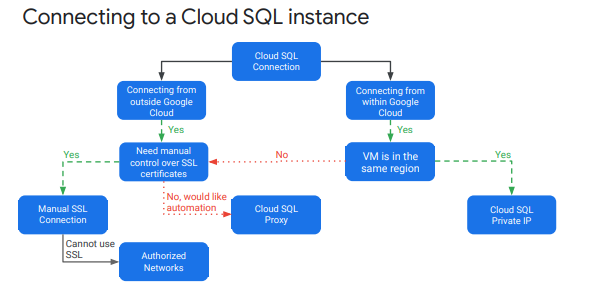
* Managed file storage service for applications
* Fully managed network attached storage (NAS) for Compute Engine and GKE instance (Google Kubernetes Engine)
* Predictable performance
* Full NFSv3 support
* Scales to 100s of TBs
* Examples, application migration, media rendering, Electronic Design Automation (EDA), data analytics, genomics processing, web content management

Cloud SQL

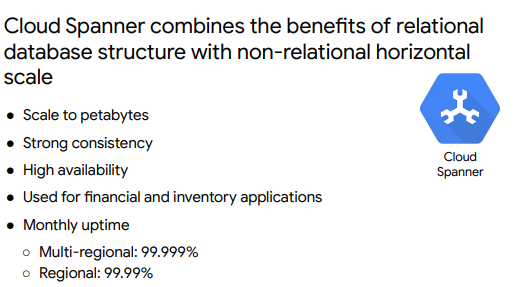


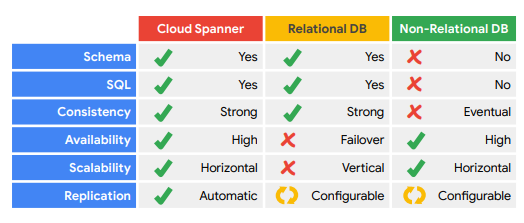


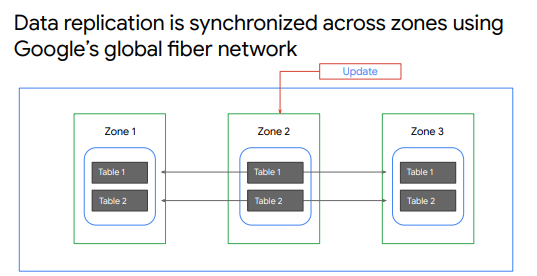




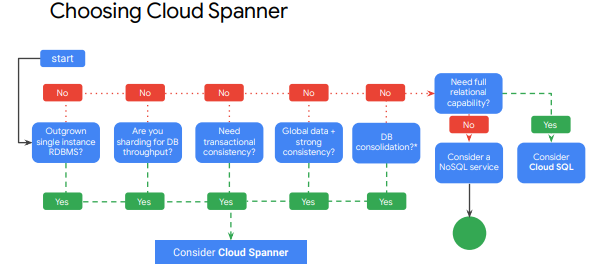
Cloud Spanner



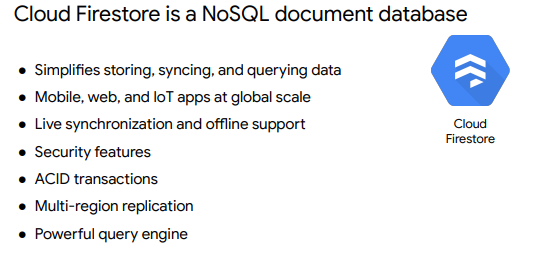


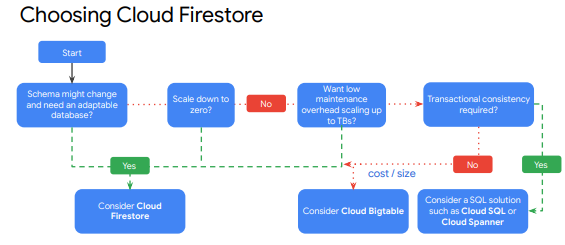


^replicates data in N cloud zones, within one region or several regions. Placement is configurable (choose which region to put data in), allows for high availability and global placement

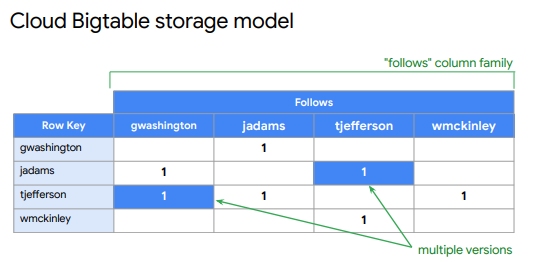
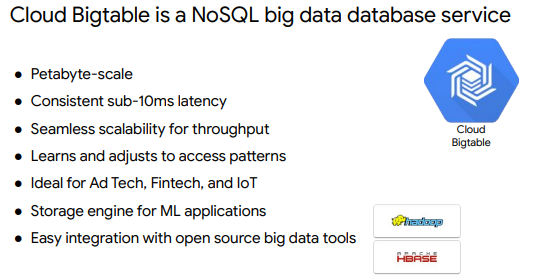


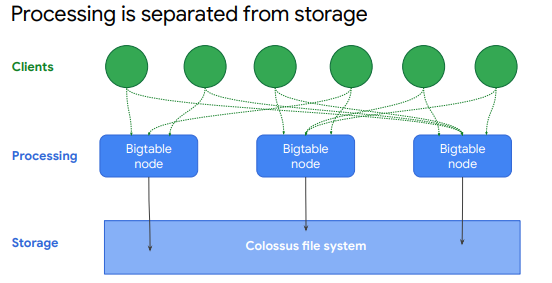
Cloud Firestore



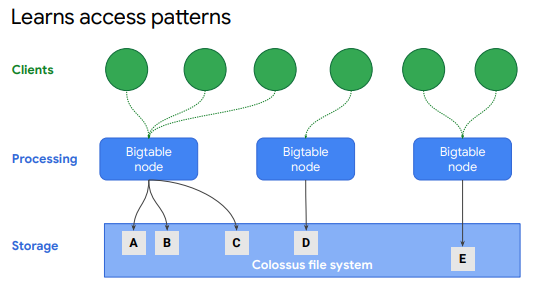


Cloud Bigtable

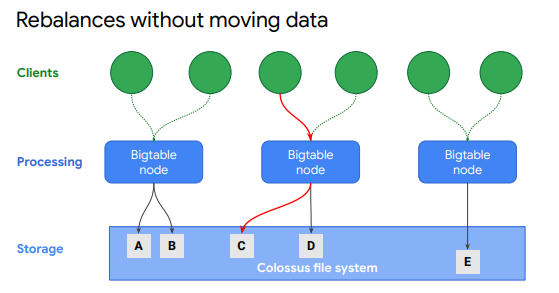


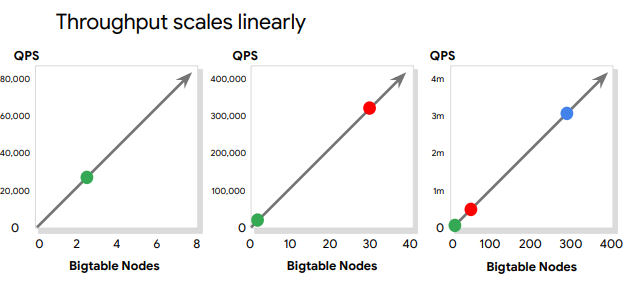


^processing is handled separately from storage

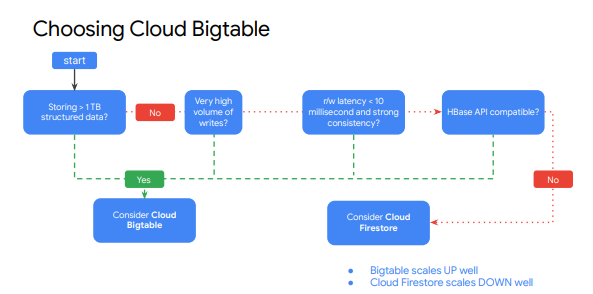


^ if a certain node frequently accessed a certain subset of data, BigTable updates indexes so other nodes can distribute that workload evenly (shown below)



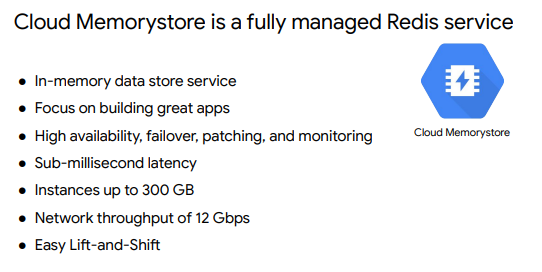


^for every node added, you see linear scale of throughput performance (up to 100s of nodes)



^smallest Bigtable cluster is 3 nodes that handle 30,000 operations per second. Nodes are paid for while operational, regardless of if the application is using them

Cloud Memorystore



**Resource Management**