Storage and Databases

Target Complete Date	@December 2, 2022
Status	Completed
Reviewed	To review
# Time to complete (Hours)	5
Type	Cloud Guru

- Introduction to Data storage in GCP
  - Horizontal scaling vs vertical scaling
  - Cloud storage
  - Cloud BigTable
  - BigQuery
  - Cloud Spanner
  - Cloud SQL
  - Firestore
  - Cloud Memorystore
- Working with Data
  - Data modelling
- Cloud storage
- Service accounts
  - Keys and access
- Cloud Storage Transfer Service
  - BigQuery Data Transfer Service
  - Transfer appliance
- Cloud SQL
  - High availability
  - Import MySQL
  - Postgres
- MySQL Demo
  - What is secure tunneling
  - Why do we clone databases?
- Cloud firestore
  - SQL vs NoSQL

Introduction to Data storage in GCP

💡 You can decide where to store your data based on a flowchart.

✏️ **Data latency:** The time it takes for data packets to be stored or retrieved.

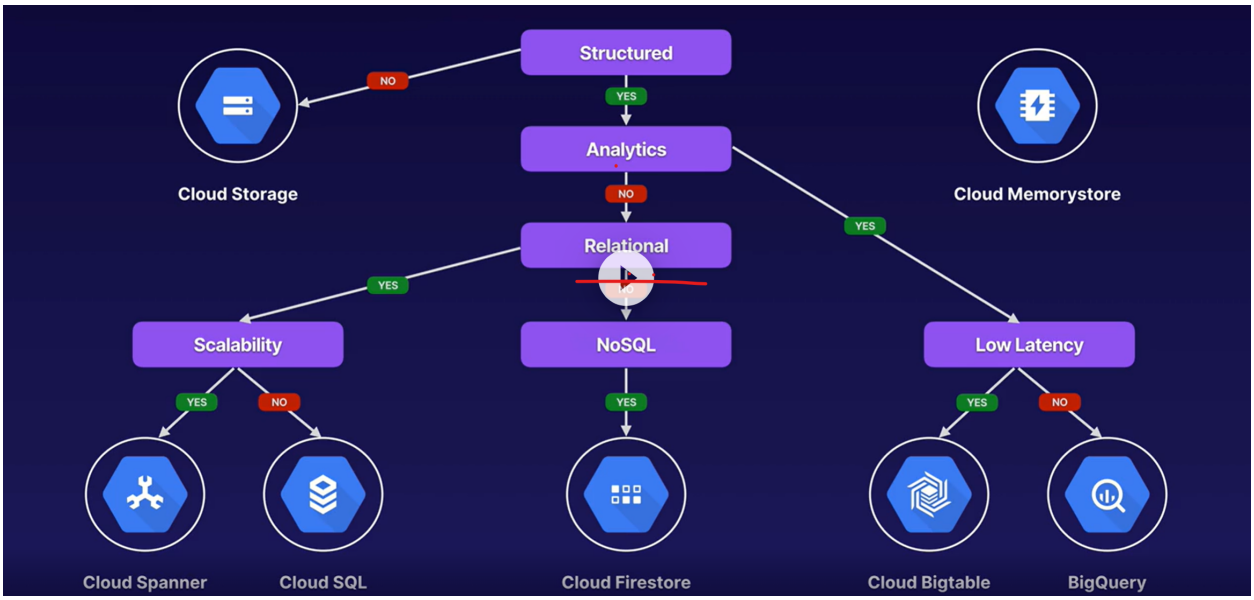
Horizontal scaling vs vertical scaling

💡 **Scalability** in databases is the ability for a database to adjust its resources to meet its demands constantly. As a project grows bigger or traffic increases, the original database server resources, such as RAM, CPU, and hard disks might not suffice. This when you'll need to start to scale your database.

Scaling can be achieved two ways: Horizontally and vertically.

✏️ **Vertical scaling** is the action of adding more resources to **a server** to handle an increasing load. Including CPUs, RAM, or hard disks. This is typically done by the cloud provider.

✏️ **Horizontal scaling** is adding **more servers** so that the load is distributed across multiple nodes. Usually requires more effect, but it is easier to scale indefinitely. Typically done through clustering and load-balancing.



Cloud storage

Ideal for unstructured object storage. Provides you with buckets that can depend on regional, dual-region or multi-region access. Additionally, it can contain standard, nearline or coldine buckets. It can also trigger storage events.

Cloud BigTable

Petabyte-scale NoSQL database. Designed for wide column high write volume data. Such as time-series, transactional or IoT data. Wide column, key value data.

BigQuery

Petabyte-scale analytics data warehouse. Fast SQL queries across large datasets.

Cloud Spanner

💡 Global SQL based relation database. Provides horizontal scalability and high availability. Along with strong consistency. Used a lot in the financial sector

Cap theorem

💡 The cap theorem dictates that when a system uses a partitioned system relying on network access. The network may fail, subsequently leaving the system with a choice to either maintain high availability or high reliability, but not both.

- This theorem asks how reliable we can build a system
- Formalizes some useful limits on reliability.
- Stands for Consistency, Availability and Partition tolerance
- A consistent design would prevent accessibility as it would not be able to guarantee consistency across all nodes.
- An available design will provide you access however it might come at the expense of consistency.
- If there is a partition you must choose, you either have a consistent design or an available design.
    - You can make a trade-off between the two
- **Consistency.** All reads receive the most recent write or an error.
- **Availability.** All reads contain data, but it might not be the most recent.
- **Partition tolerance.** The system continues to operate despite network failures (ie; dropped partitions, slow network connections, or unavailable network connections between nodes.)

Cloud SQL

💡 Managed MySQL and PostgreSQL instances. Built in backups, replicas and failover. Vertically scalable, SQL server.

Firestore

💡 Fully-managed NOSQL document database. Large collections of small JSON docouments. Realtime database with mobile SDKs. Strong consistency.

- SDK stands for: Software Development Kit

Cloud Memorystore

💡 Managed redis instances. In-memory DB, cache or message broker. Built-in high availability. Vertically scalable.

Working with Data

Key considerations needed to manage data. How to get it where it needs to go, in its desired format.

1. Sources and sinks: Does the data require preparation work? Is the source format appropriate for the data sink?
2. Structured and unstructured: How is data being processed and stored? Does structured data conform to the correct model?
3. Batched and streaming: Are pipelines required to ingest data? Are latency and time-windowing a concern?

Data modelling

💡 **Data modelling**: Structured data requires a consistent model. The model may already be in place. Data may require preparation or transformation.

Stages of modelling:

1. Conceptual: What are the entities in my data. What are their attributes and relationships?
2. Logical: What are the structures of my entities? Can the model be normalized?
3. Physical: How will I implement my database? What keys and indexes do I need?

We must ensure that we have good relational schema design.

- Normalization and reducing waste
- PK and table relationships
- Accuracy and integrity
    - How to handle schema changes

Cloud storage

1. Fully managed object storage
    a. Unstructured data
2. Multiple storage classes: Life cycle management for objects and buckets.
3. Secure and durable

> 💡 A bucket is a logical container for objects. Buckets exist within projects. Has to have a globally unique name.

- Geo-redundancy:
    - Regional
    - Dual-regional
    - Multi-regional
- Storage classes: Each class behaves the same way. Costs are different
    - Standard

        - 99.99 % availability
    - Nearline
        - Stored for 30 days
    - Coldline
        - 90 days minimum storage
    - Archive
        - 365 days minimum storage
- Objects are stored as opaque data.

> ✏️ **Opaque data type** is a data type whose concrete data structure is not defined in an interface

- Encrypted in flight and at rest
- Objects are immutable, must be overwritten
- Overwrites are atomic operations
- Objects can be versioned

Access to google cloud storage through:

- HTTP API
- GCP console
- SDKs
- gsutil

Supports advanced features:

- Parallel uploads of composite objects
- Integrity checking
- Transcoding
    - Can be compressed and decompressed during downloads

- Requestor pays
  - Insists on charges being made to the individual requesting data
  - Operation charges:
    - Class A: uploading
    - Class: B: Downloading
    - Class C: Deleting
  - Networking charges applies when there is data egress.

> ✏️ **Data Egress**: Refers to data leaving a network in transit to an external location.
>
> - Moving data out of bucket to other places, i.e. other regions.

  - Network retrieval charges

Life cycle management can be applied to lower costs

- Apply a life cycle configuration file
  - GCP will check conditions of a file, and perform an action.

Security and access controls

- IAM for bulk access to buckets
- ACLs for granular access to buckets
- Signed URLs
- Signed policy documents

Pub/Sub will usually be used a way to automate the ingestion the data in cloud storage.

Service accounts

✏️ Identity & Access management in Google cloud are controlled by policies which contain members and roles.

- Members can contain emails or services accounts
- Roles define a list of permissions that are assigned to one of the members
  - Instance admin
  - Pub/Sub publisher
- Members are human users
  - Authenticate with their own credentials, tied to the life cycle of the user.
  - Do not use for non-human operations
    - Access could stop functioning
    - Could leak into source code
- Service accounts are specific accounts created for specific tasks
  - Identity can be assumed by an application/workload
  - Keys can be easily roated
- Google-managed and user-managed SAs
- GCP services assume service account identity

Keys and access

- Google managed keys for google managed SAs
- User-managed keys for user-managed Sas
- User-managed key is a downloaded JSON file
  - Extremely powerful credentials
  - Be very careful with it

Industry standard

- IAM for bulk access to buckets

    - Roles assigned to members
- ACLs for granular access to buckets
    - Access control lists allow you to grant permissions to a scope

Access control lists

Permissions:

- Reader
- Writer
- Owner
- Default: Predefined project-private ACL applied

Cloud Storage Transfer Service

- This automates a transfer of data from a source to a sink.
- The sink is always google storage.

Supported sources:

- AWS S3
- HTTP
- Buckets

You have further use with CSTS.

- Filters based on names and dates
- Scheduled and periodic transfers
- Delete objects in destination and source bucket

You can assign IAM roles to transfer service.

BigQuery Data Transfer Service

- Automates data transfer to Bigquery
- Data loaded on a regular basis

- Backfills can recover from gaps or outages

Supported sources:

- Cloud Storage
- Youtube
- S3 etc..

Transfer appliance

- Physical rackabble storage device
- Ship the full device back to Google

Cloud SQL

Features:

1. Managed SQL instances
    a. Don’t set up servers or softwares, patches
2. Multiple database engines
    a. Postgres, MySQL
3. Scalability and availability
    a. Vertical scaling to 64 cores and 416GB RAM

- You need to define maintenance windows, that will tell Cloud SQL when to perform disruptive maintenance
- Automated backups
- Instance restores
- Point-in-time recovery
    - Requires binary logging
- Unsupported features:
    - User defined functions

    - InnoDB
    - Super privileges
    - Unsupported statements

High availability

> 💡 Spanning a region and two zones, with a primary mySQL instance which replicates to another zone. The client application communicates with the second instance, by changing the database IP address that is linked to client.
>
> Read replicas can also be created, which will provide extra read capacity for application processes that don’t need to write to the database.

Import MySQL

- InnoDB
- CSV import
- External replica promotion
    - Binary log retention

Postgres

- Secure IP whitelisting or cloud proxies
- Supports Postgres extensions
    - PostGIS
- Uses regional persistent disk
    - No new standby instance

☑️ ~~What is a cloud proxy?~~

☑️ ~~What is IP whitelisting~~

When you grant access only to certain IP addresses

☑️ ~~What is a backup?~~

- [x] ~~What is point-in-time recovery~~
- [ ] Look into google keep

MySQL Demo

⚙️ In this demo we launch a MySQL instance, and serve up a VM to interact with the MySQL instance. He does something with a cloud SQL proxy, I do not know what that means, and he connected it using a service account.

⚙️ You are able to import SQL files as schemas into cloud SQL databases, as well as import data directly using a CSV file import on the GUI.

Proxies

What is secure tunneling

💡 SSH tunneling is a direct secure connection to another computer so that you can control it. It allows you to forward data to another location securely.

Networking ports: Networking protocols assign each shipment of data specific port numbers.

- [x] ~~Why do we clone databases?~~

Why do we clone databases?

💡 Cloning a database is the process of creating a point-in-time copy of a production database or it’s backup set.

Multiple reasons to do so:
- Recovering data mistakenly deleted

Cloud firestore

SQL vs NoSQL

SQL

- Entities with relationships
- Strict schema
- Complex queries
- Scale vertically

NoSQL

- Hierarchical documents
- Dynamic schema
- Simple queries
- Scale horizontally

Firestore features

1. Fully managed NoSQL database
    a. Integrated with GCP and Firebase
    b. Serverless autoscaling NoSQL document store
2. Realtime DB with mobile SDKs
    a. Android and IOS client libraries
3. Scalability and consistency
    a. Horizontal autoscaling and strong consistency

Firestore Data model

💡 Hierarchical store of documents, with nested collections. Can also use references that point to other documents within another document in a database. Ideal for semi-structured data that has to be highly scalable and available

Support Data types

- String, Reference, Bytes, Maps, Arrays

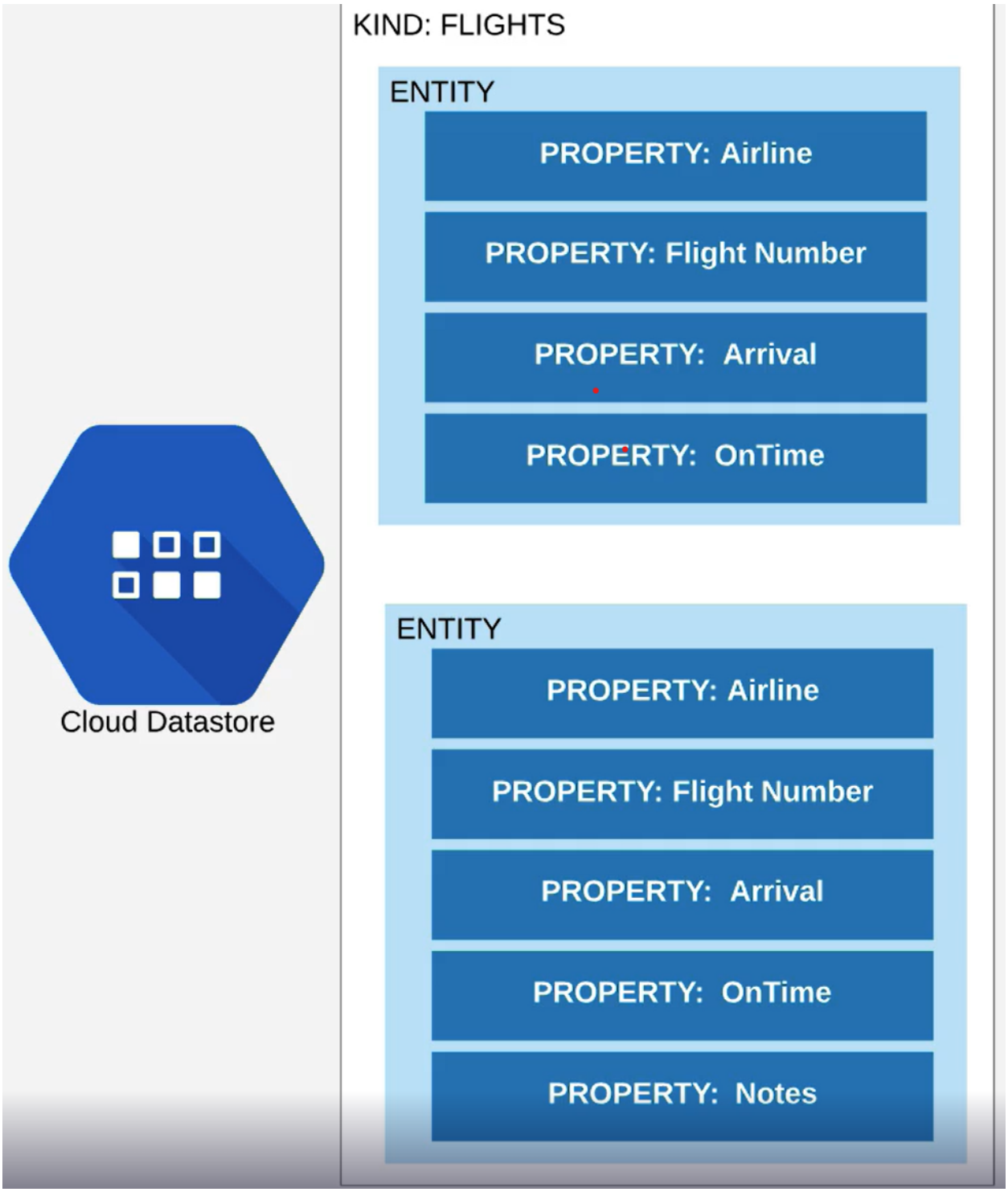
Working with Data

- Create a new document with a unique identifier
- Each document needs a unique identifier to work
- Retrieve a document using a Unique ID or a all documents within a given collection
    - Can also use a WHERE clause
        - Array contains etc…
- Automatically creates single-field indexes, it means it already has an index for most types of queries. So it is quick outside of the box.

☐ What is a composite index

Indexes

Cloud Datastore Demo


KIND: FLIGHTS
ENTITY
PROPERTY: Airline
PROPERTY: Flight Number
PROPERTY: Arrival
PROPERTY: OnTime
Cloud Datastore
ENTITY
PROPERTY: Airline
PROPERTY: Flight Number
PROPERTY: Arrival
PROPERTY: OnTime
PROPERTY: Notes

✏️ **Entity**: Equivalent to a standard db row
Properties: Equivalent field
Kind: Equivalent to a relational database table

⚙️ To query data in firestore, we use GQL (Google Query Language). A field has to be indexed for you to be able to use the where clause for it.

Cloud spanner

Features

1. Managed SQL-compliant DB
2. Horizontally scalable
3. Highly available

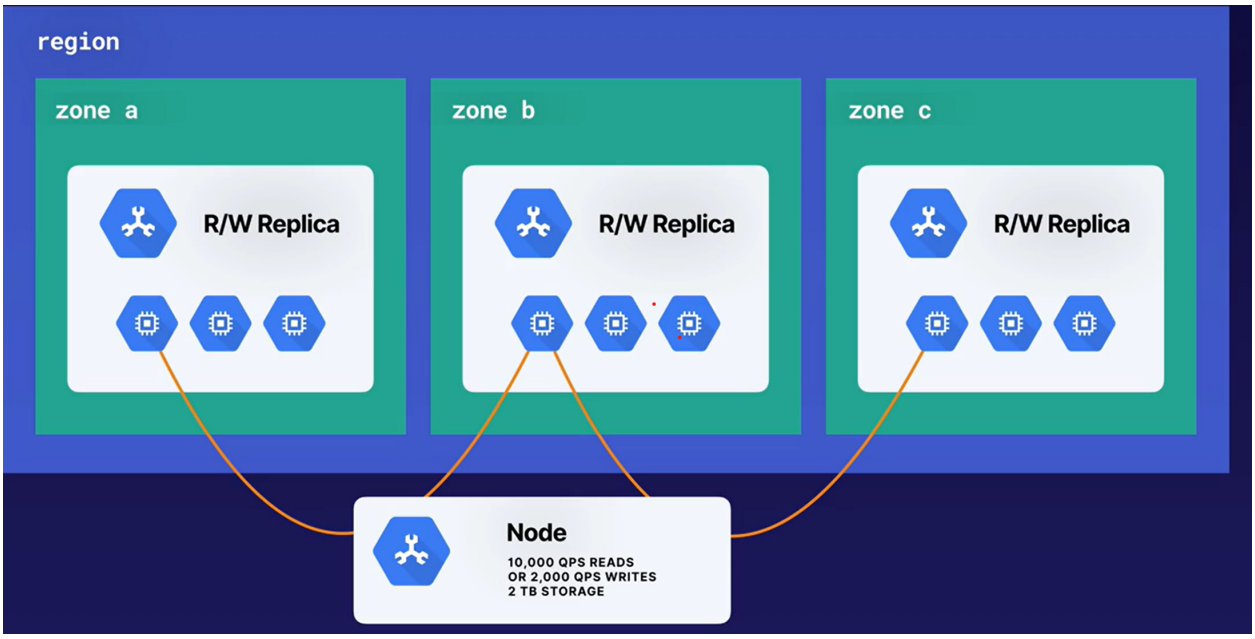
Cap theorem

💡 In a distributed system that stores data, there are three principles: Consistency, Availability, Partition tolerance.
Consistency means db transactions must only change data according to specific rules, **Availability**: Always available to serve queries
Partition tolerance: Tolerant to the loss of any partition, system needs to tolerate failures if part goes offline.

- Global private network allows it to provide 5 9s of availability.

Instances

> 💡 A spanner instance is an allocation of resources. It's configuration simplifies to two things: regional availability, and the number of nodes.



- Spanner places R/W replica in each zone
- Node count of 1, means each replica will be powered by 1 virtual machine
- The number of replicas always stays the same, but you can scale the node count up and down
- Each node can serve 2TB , and serve many R/Ws

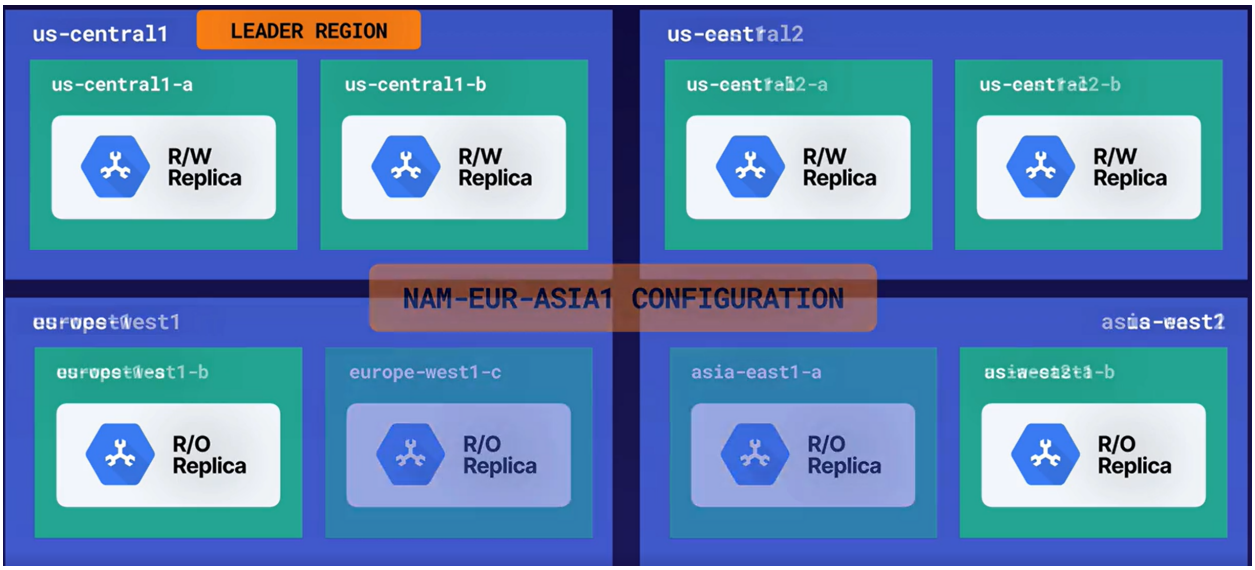
☐ Replicated databases uses quorem

☐ Learn more about distributed systems

Regional Configuration

- Design a performant schema
  - ☐ Avoid hotspots (what does this mean)
- Colocate compute workloads in the same region
- Provision enough nodes to keep average CPU util under 65%

Multiple-regional configurations



- There is a leader region, this is related to distributed systems theory
- Multi-regional benefits
  - 5 9s SLA
  - Reduce latency with distributed data
  - External consistency
    - Concurrency control for tx, that is strict
    - Financial transactions in banks
- Every mutation requires a write quorem

Data model cloud spanner

- Uses relational database tables
- Data is Strongly typed
  - [ ] What does strongly typed mean?
- Strict schema
- Offers Parent-child relationships, rows in same table can be put in same spanner node, for quicker queries.

Transactions

- Locking read-write
  - [ ] What does this mean?
- Read-only
- Partitioned DML
- Ansi SQL best practices

Connecting to Cloud spanner via public API

Cloud spanner demo

⚙️ INTERLEAVE IN PARENT means that lookups on children table will be much more efficient.

⚙️ Cloud spanner allows you to export your data to buckets using dataflow. It will send your data as a binary avro file to your chosen bucket destination.

Cloud memorystore

💡 Cloud memorystore is a fully managed redis instance

- [ ] What is Redis?
- [ ]

Redis

Features:

1. Fully manged redis instance
2. Basic tier

    a. Efficient cache that can withstand a cold restart and full data flush
3. Standard tier
    a. Adds cross-zone replication and automatic failover

Benefits:

- There is no need to provision VMs
- Scale instances with minimal impact
- Private IP and IAM
- Automatic replication and failover
- Choose service tier and region
- Memory capacity
    - Determines network throughput
- Add configuration parameters

Connecting to instances

- Compute engine
- Kubernetes engine
- App engine
- Cloud functions
    - May require serverless VPC connector
    - [ ] What is VPC connector?

Import and export

- Export to RDB backup
- Import from RDB backup
    - Overwrites all current instance data
    - Instance unavailable during import process

Use cases

- Session cache
  - Persistent session cache
- Message queue
- Pub/Sub

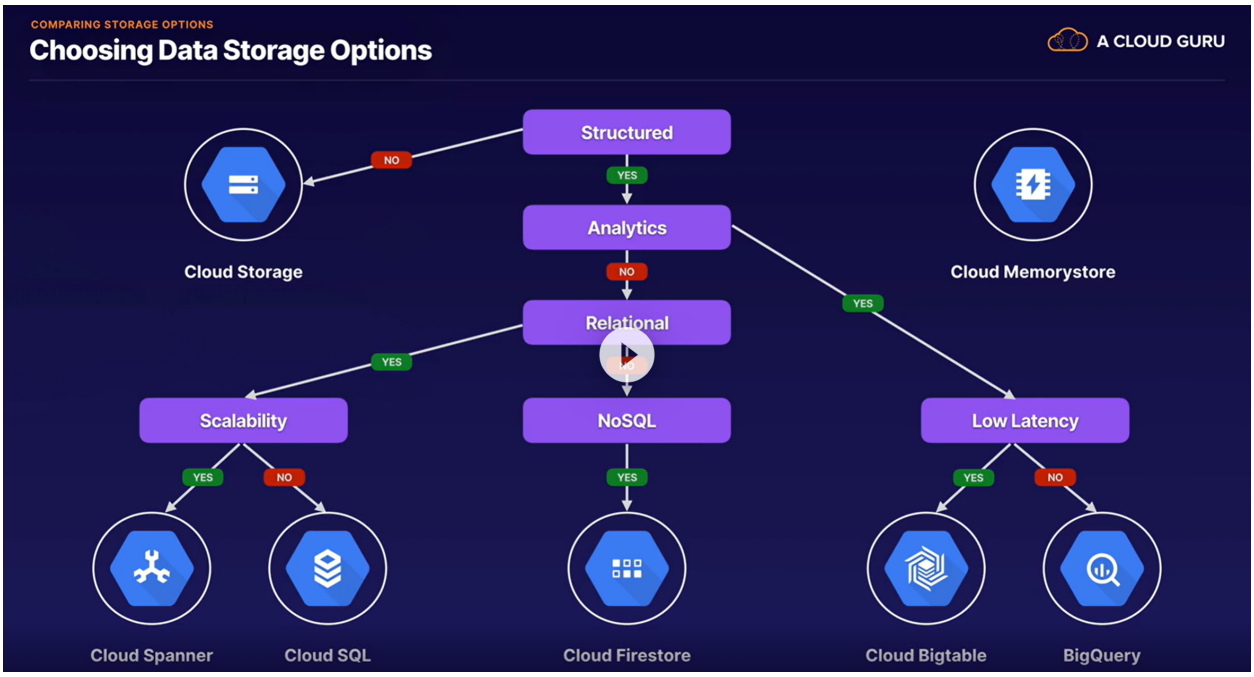
Cloud memory store demo

> ⚙️ In this demo we make a twitter clone. Each post is stored as a key value entry, and each entry is being written to the Redis instance. Using Reddis cli we can connect to the memory store instance. In this example, we exported the database from mem store to a bucket, and then re-imported it, to replicate a point in time recovery.

- We need to know when to use reddis
- Accessing memory store is the same as accessing any redis instance

Comparing storage options

> 💡 The whole point of this chapter has been to enable a data engineer to make decisions as to which services to use in regards to data storage.



- [ ] Review cloud Firestore

Structured vs unstructured

Structured

- SQL data
- NoSQL
- Analytics data
- Keys and values

Low latency vs warehouse

> 💡 When we think about analytics in google terms, we are referring to huge volumes of data over time that we want to use to make predictions or feed machine learning models. This is different from relational data.

Low latency

- Petabyte scale
- Single-key rows
- Time series or IoT data

Warehouse

- Petabyte scale
- Analytics warehouse
- SQL queries

Horizontal vs Vertical scaling

Horizontal

- ANSI SQL
- Global replication
- High availability and consistency
- Expensive: Cloud Spanner

Vertical

- MySQL or PostgreSQL
- Managed service
- High availability

NoSQL vs Key/Value

NoSQL

- Fully managed document database
- Strong consistency
- Mobile SDKs and offline data

Key value

- Redis

☐ Look at Firestore use cases

Exam types

- Choose the right product
  - Memorize the flow chart
- Consider the business requirements
  - Do not pick the best technical solution
  - Several will satisfy technical requirements but only one will fit the business requirement as well
- Get to know the ecosystem
  - GCP data engineers need to help customers move from on-premise to GCP products
- Do the tutorials

  ☐ GCP documentation, designing schemas, creating indexes and tuning performance
- Don’t overlook security
  - Separate users who can write data from users than can just read data

  ☐ Look into regional persistent disk, and how Cloud SQL maintains availability