GCP

Google Cloud

Professional Cloud Database Engineer

Google Certified Professional Cloud Database Engineer

Database Concepts

BY ANKIT MISTRY

Structure, Semi-structured & unstructured Data

BY ANKIT MISTRY

Structure Data

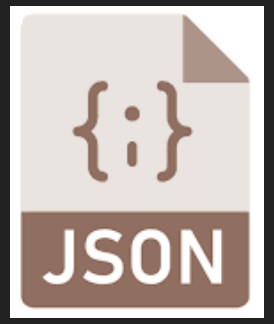
- Fixed Schema
- Stored in rows & column format - Table
- Relation exist between data
- Almost 20% of enterprise data
- Stored in order
- Require less storage
- SQL can be used to interact with data
- Each row has same number of columns
- MySQL, Oracle SQL, PostgreSQL , MSSQL
- In GCP, Cloud SQL, Cloud Spanner

Book_id	Book_name	Author_id
100	C	1
101	Java	1
102	Python	2

Author_id	Author_name
1	John
2	Alice

Semi-structured Data


```
<employees>
  <employee>
    <firstName>John</firstName>
    <lastName>Doe</lastName>
  </employee>
  <employee>
    <firstName>Anna</firstName>
    <lastName>Smith</lastName>
  </employee>
  <employee>
    <firstName>Peter</firstName>
    <lastName>Jones</lastName>
  </employee>
</employees>
```




```
{"employees":[
  { "firstName":"John", "lastName":"Doe" },
  { "firstName":"Anna", "lastName":"Smith" },
  { "firstName":"Peter", "lastName":"Jones" }
]}
```

- No SQL Database
- JSON, XML, Document DB – No Fixed schema
- MongoDB, Cassandra, Redis, Neo4j
- In GCP, BigTable, DataStore, memoryStore

Unstructured Data

- No Fixed Schema
- No Structure within data
- occupies much more space for storage
- Almost 80% of data
- Video, Audio, Binary, Zip file kind of data are unstructured data
- Google Cloud Storage, File store inside GCP to store Unstructured data

OLTP & OLAP

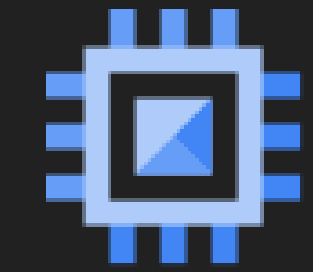
OLTP

- OLTP – Online Transaction Processing
- Simple Query
- Large number of small transaction
- Traditional RDBMS
- Database modification
- Popular Database - MySQL, PostgreSQL, Oracle, MSSQL
- ERP, CRM, Banking application
- GCP - Cloud SQL, Cloud spanner

OLAP

- OLAP – Online Analytical Processing
- Data warehousing
- Data is collected from multiple sources
- Complex Query
- Data analysis
- Google Cloud Big Query – Petabyte Data warehouse
- Reporting Application, Web click analysis, BI Dashboard app

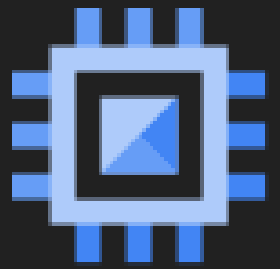
Vertical – Horizontal Scaling



8 VCPU - 16 GB

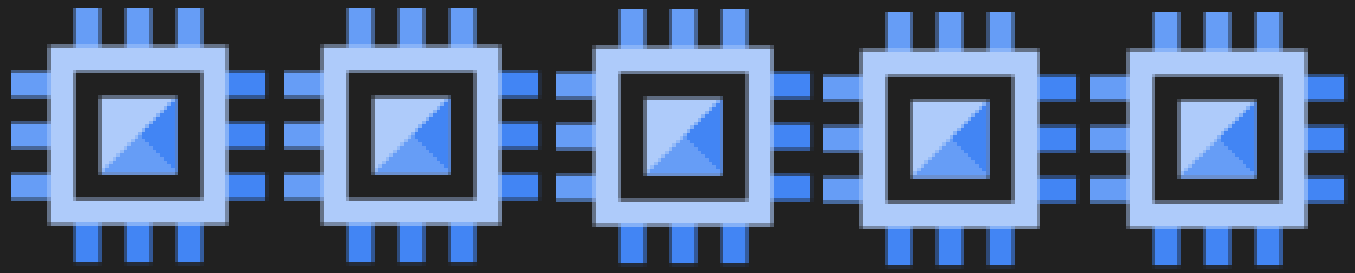
Vertical Scaling

12 VCPU - 24 GB



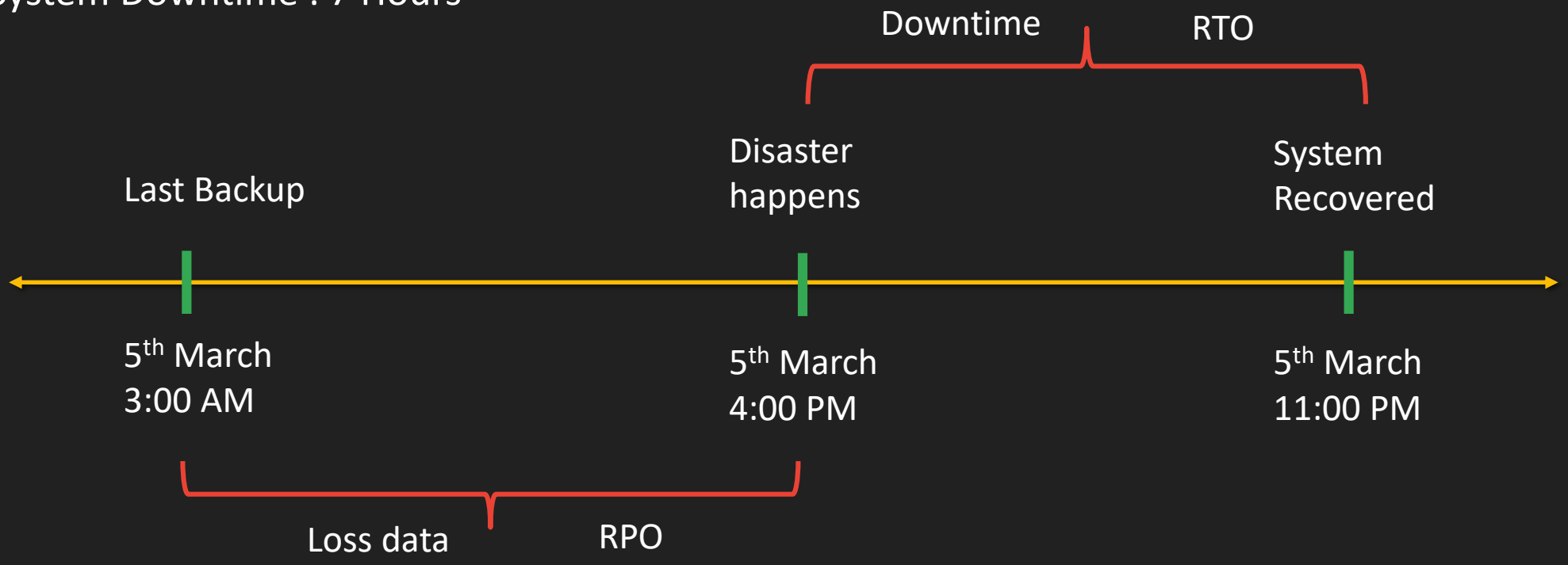
8 VCPU - 16 GB

Horizontal Scaling



RTO & RPO

- Data loss : 13 hours
- System Downtime : 7 Hours



RTO & RPO

- RTO – Recovery Time objective
  - Maximum time for which system can be down
- RPO - Recovery Point objective
  - Maximum time for which organization can tolerate Dataloss

RTO & RPO

- RTO – Recovery Time objective
  - Maximum time for which system can be down
- RPO - Recovery Point objective
  - Maximum time for which organization can tolerate Dataloss

Database Design Consideration

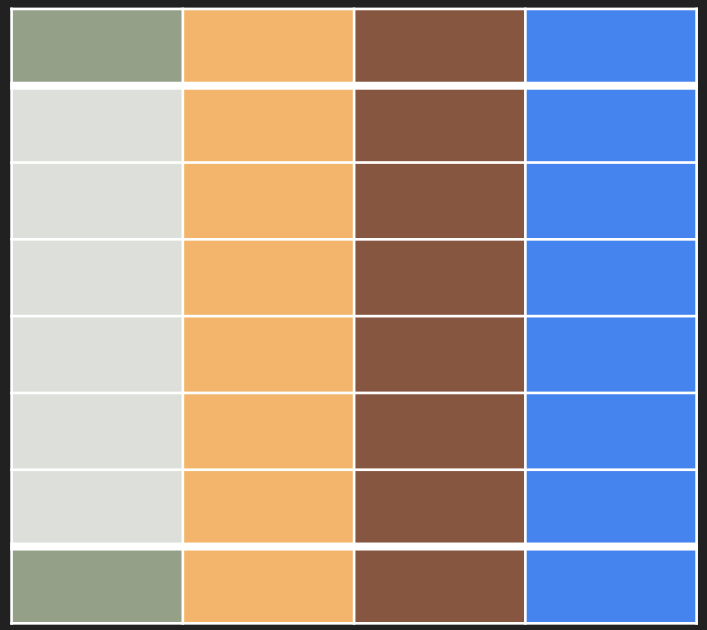
- Latency
- IOPS
- Types of Data
- Size of Data
- Schema (Fixed vs Flexible)

SQL & NOSQL Type

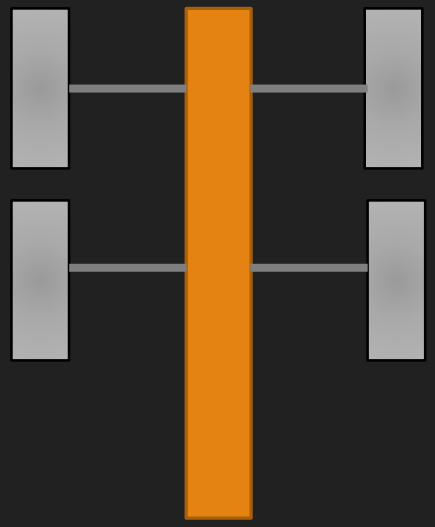
BY ANKIT MISTRY

SQL Type

Relational - OLTP

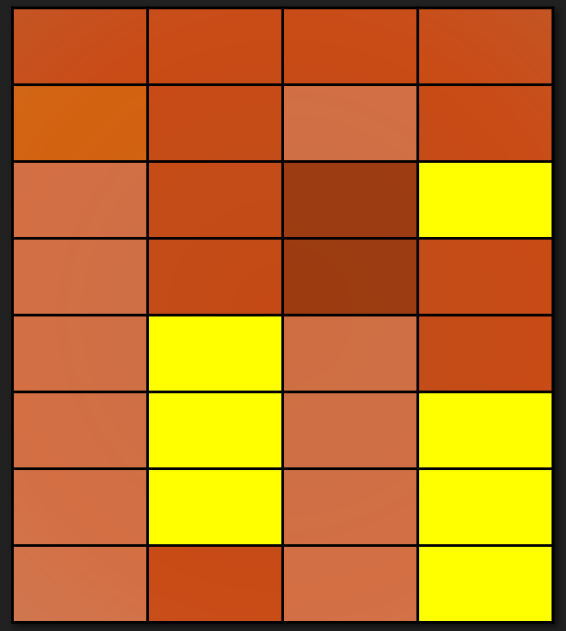


OLAP

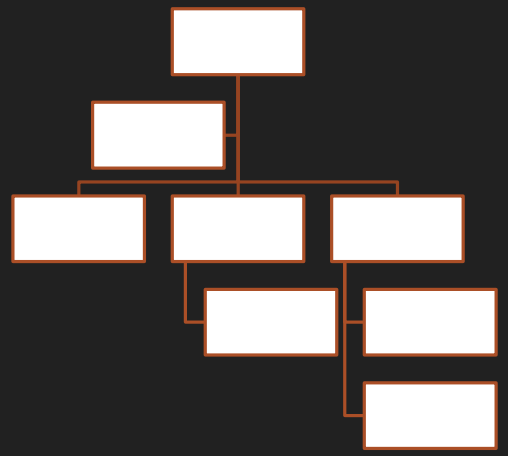


NoSQL Type

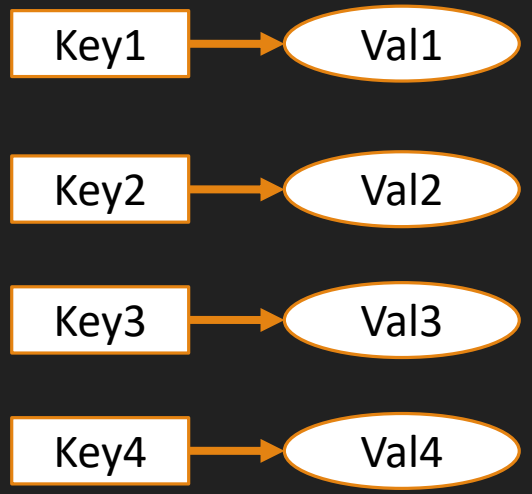
Column



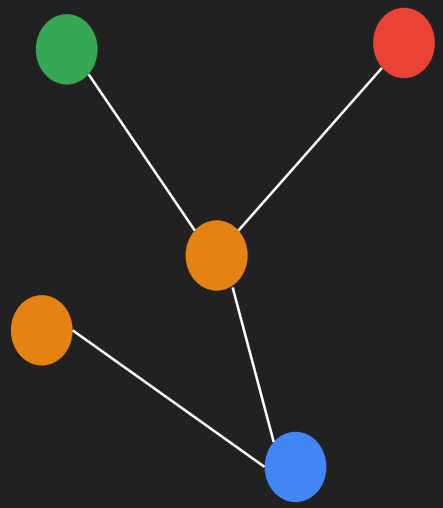
Document



Key-value



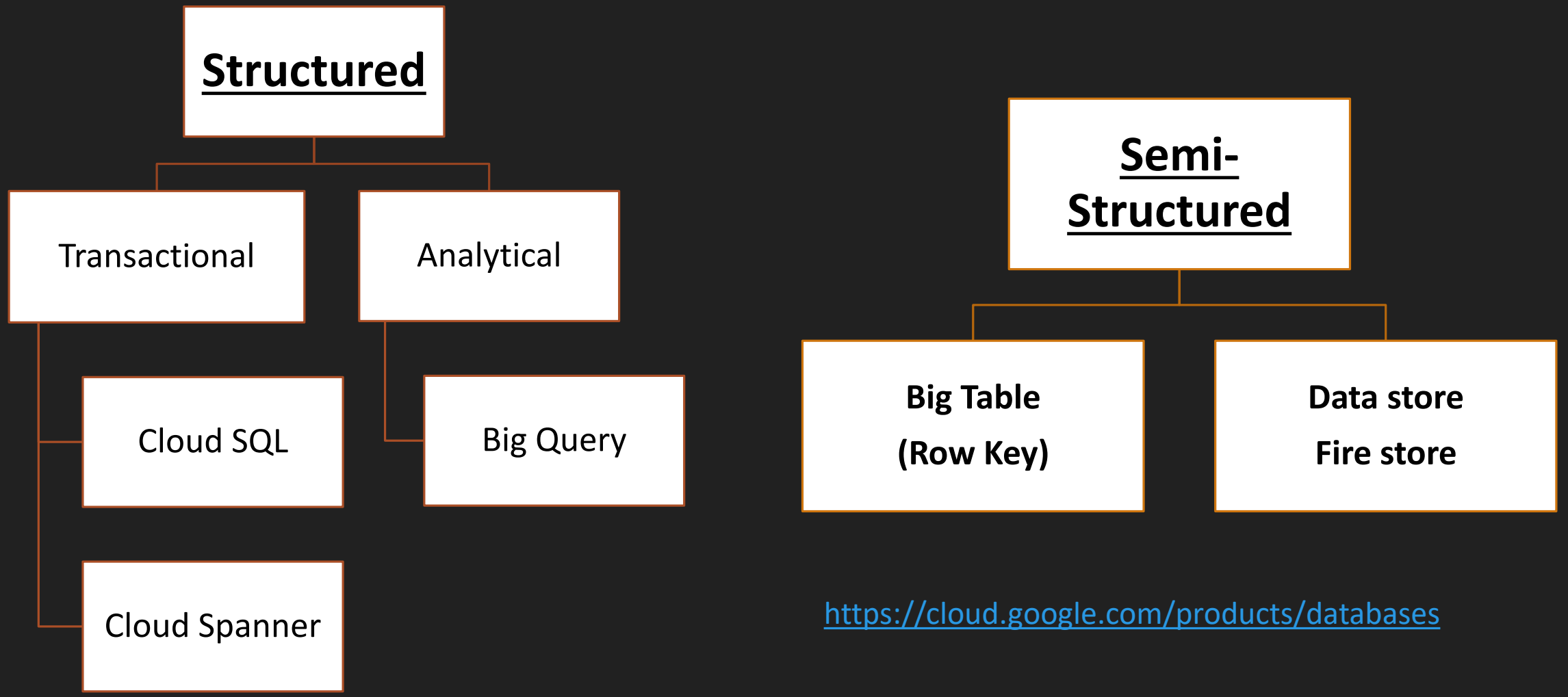
Graph



Different GCP Database Product

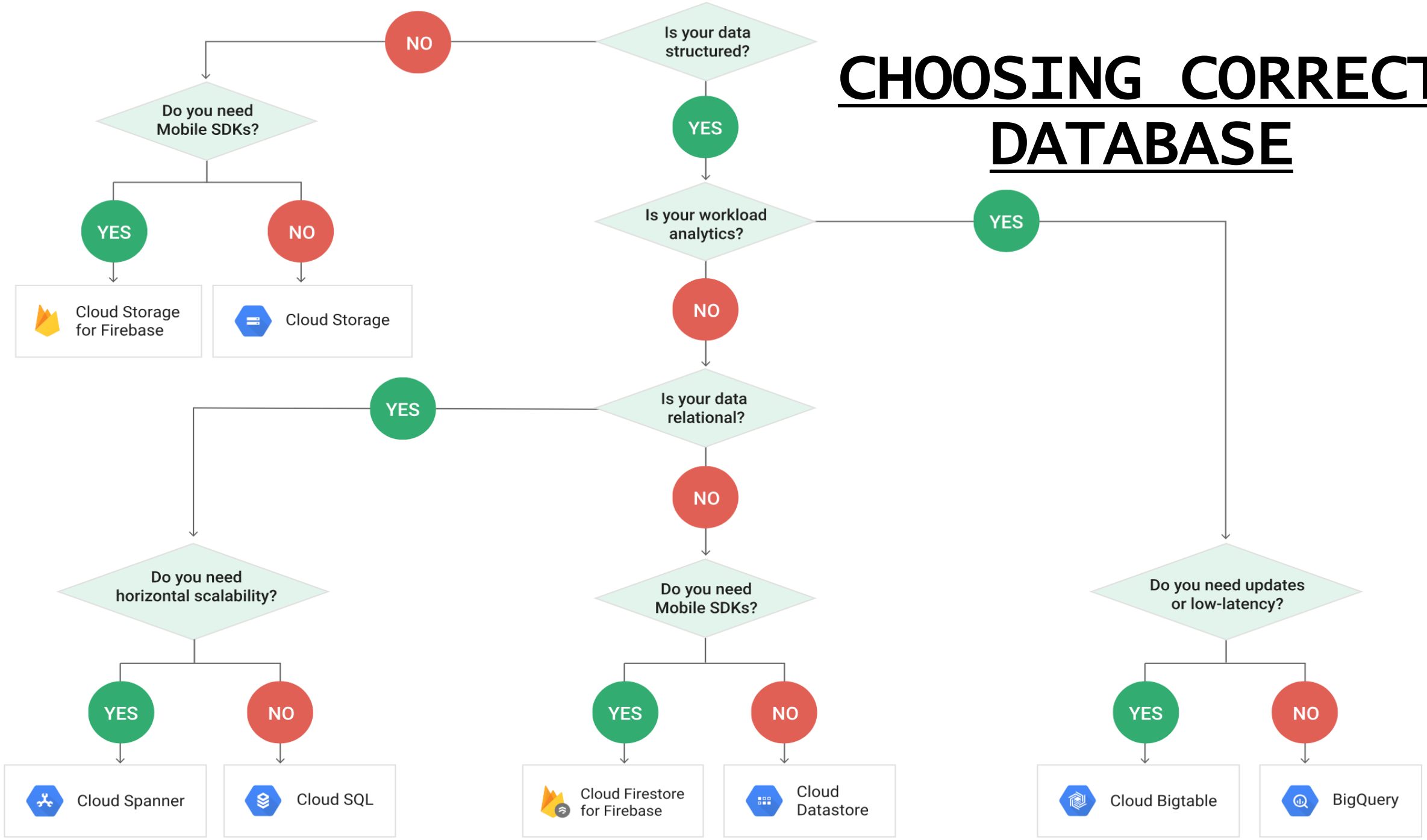
BY ANKIT MISTRY

Different Database product



https://cloud.google.com/products/databases

CHOOSING CORRECT DATABASE



RDBMS in GCloud

BY ANKIT MISTRY

RDBMS in GCloud

- You need relational databases as backend for your application
- Which RDBMS system you want to use
  - MySQL
  - Postgress
- How will you setup in Google Cloud
- Install MySQL on Google Compute Engine
- But It will not be managed services
- What other solutions
- Managed MySQL services – Cloud SQL

Google Cloud SQL

- Fully managed Relational database services for MySQL, PostgreSQL & SQL Server
- Lift & shift above database
- Regional Database with 99.95% SLA
- Storage up to 30 TB
- Vertical Scaling - Scale up to 96 core & 624 GB Memory
- Horizontal Scaling with Read Replicas - To transfer workload to other instance
- Data is encrypted with Google managed key or CMEK
- Cloud SQL can be accessed from anywhere like – App Engine, Compute Engine…
- Used for storing Transactional database
- Ecommerce, CRM kind application backend.

Google Cloud SQL

- Due to managed Service No maintenance & auto update
- Back-up Database
  - On-demand Backup
  - Schedule backup
- Database migration service (DMS)
  - migrate data from different SQL system to Cloud SQL
- Point-in Time Recovery

Google Cloud SQL

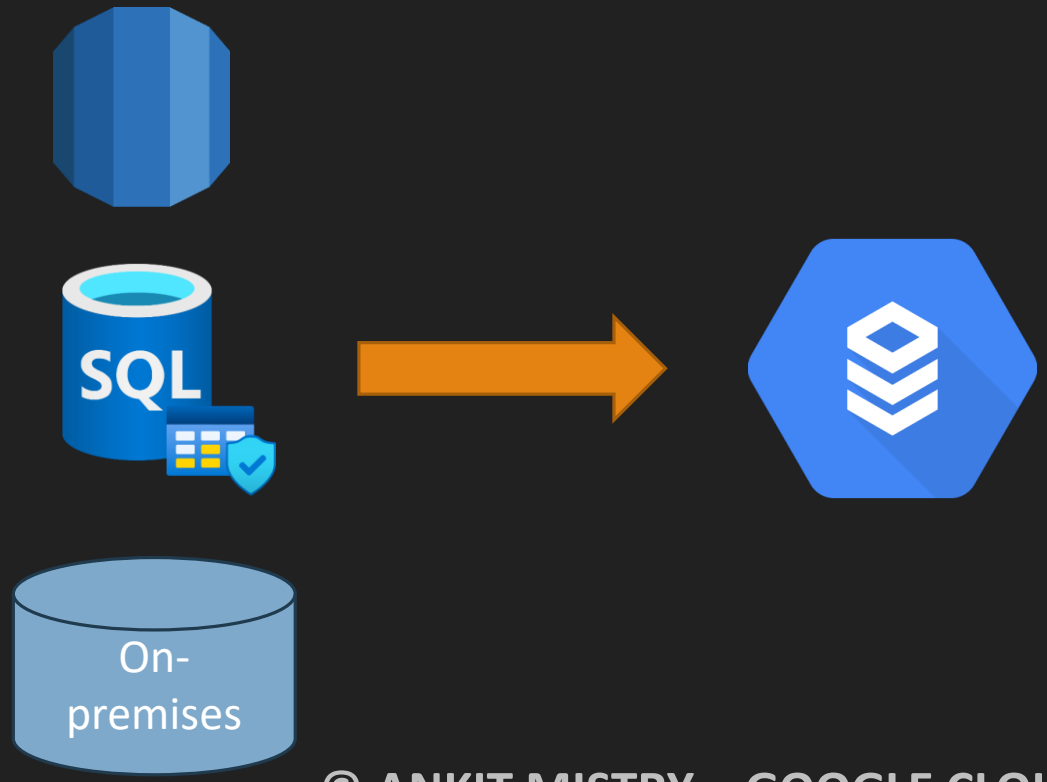
- Create Google Cloud SQL Instances
  - MySQL managed
  - Postgre – Later
- Explore Cloud SQL Features & Properties

Connect to Cloud SQL

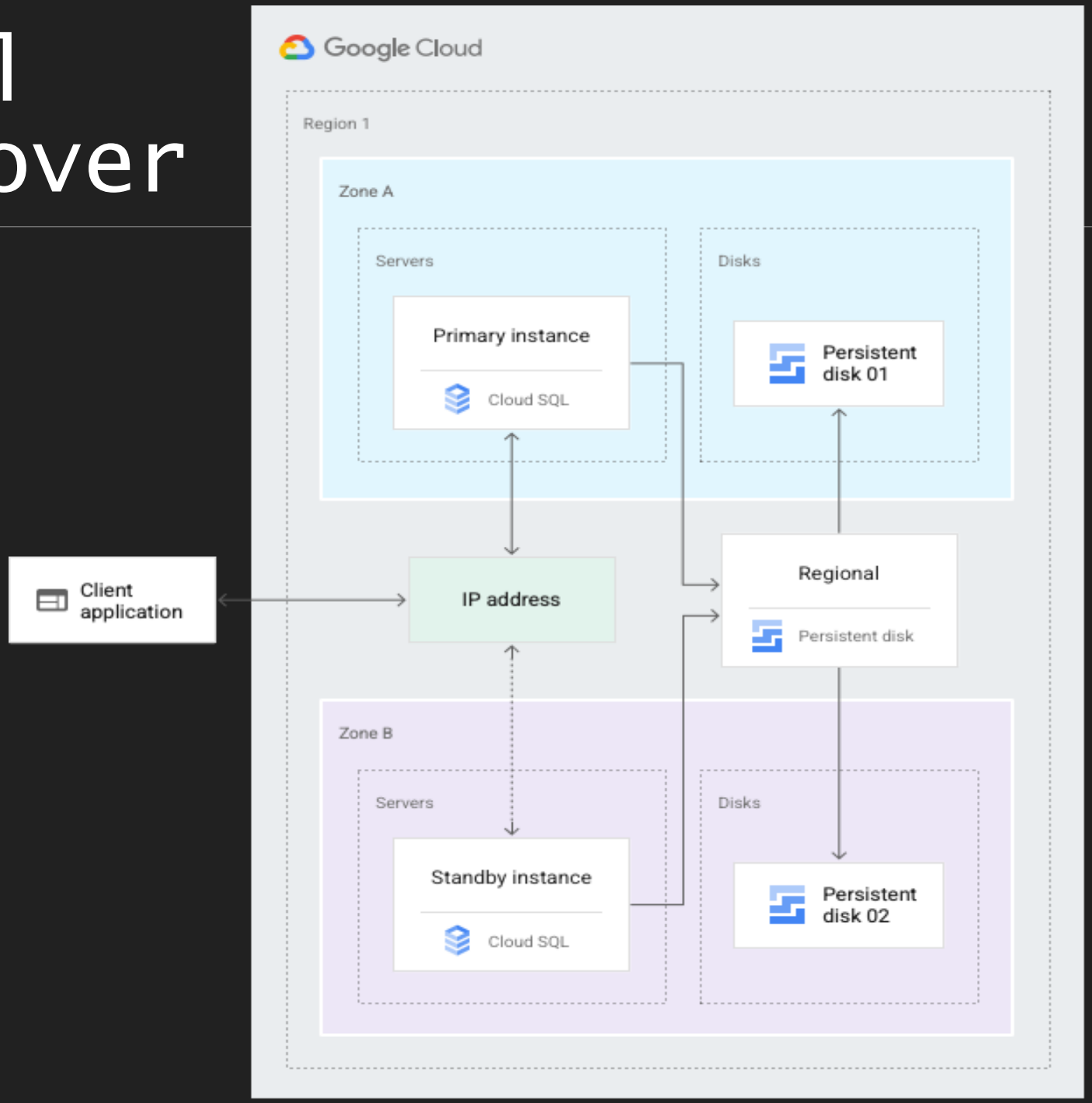
- With Public IP – Easiest one
- With Cloud Proxy
- With Private IP address
- Adding Users
  - Built-in users
  - IAM users
- Connect with above two kind of users
- Secure Connect using SSL certificate
- Some Database Operations

Lift-shift

- Migrate from
  - AWS RDS or Azure SQL or On-premises to Cloud SQL
  - Database Migration or manual Migration



Zonal Failover

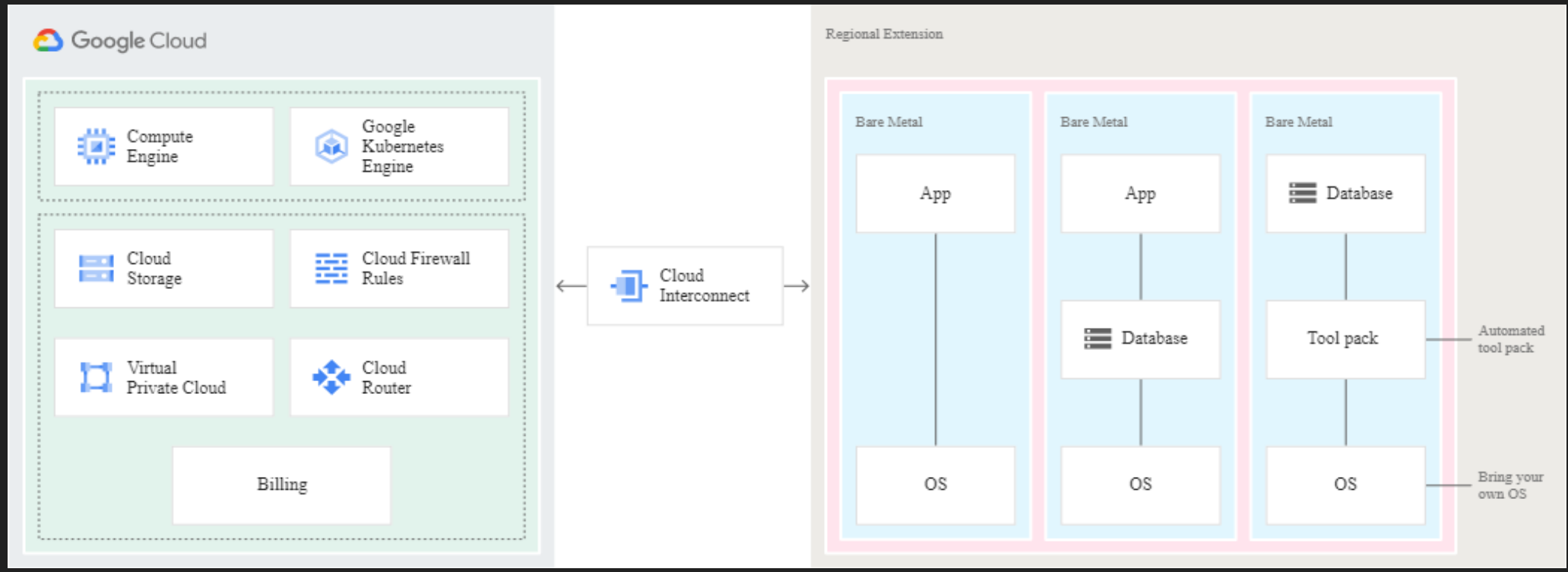


More Feature of Cloud SQL

- Backup demo & restore
- Export database
- Create Read replicas
- More instance operation start stop delete clone restart
- Cloud SQL IAM role
- Create Postgres Account & connect
- Cloud SQL Pricing calculator
- SQL instance management from command line gcloud

Bare Metal

- What about database other than MySQL, Postgres, SQL Server
- Specialized Database like Oracle, SAP HANA, DB2
- GCP doesn't provide any managed service for above Database



Few Question

- Suppose you want to store 500 TB of Relational Database
- Want to write into Read Replica kind of Instances
- Want Horizontal Scalability
- Strong Consistency between all instance
- Distributed Global Scale Database
- Cloud Spanner is the Solution.

Google Cloud Spanner

BY ANKIT MISTRY

Google Cloud Spanner

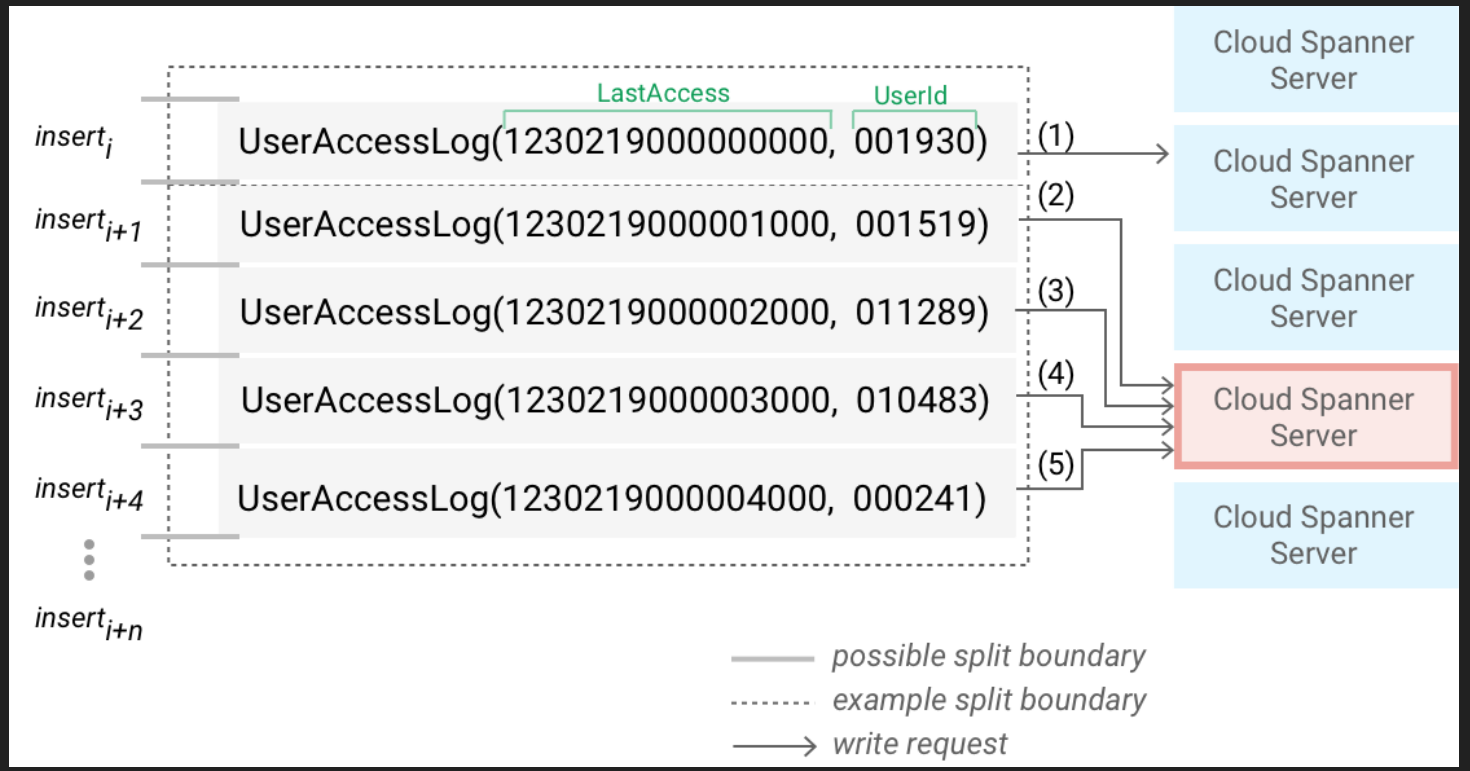
- Distributed & scalable solution for RDBMS in GCP
- Fully managed, Mission critical application
- Horizontal Scalable
- use when Data volume > 2 TB
- Costlier than Cloud SQL
- Cloud SQL has just Read replicas,
  - where as in cloud spanner horizontal read/write across region
- Highly scalable, Petabyte scale
- Data is strongly typed.
  - Must define schema database
  - Datatype for each column of each table must be defined.
- 99.999% availability
- Cloud native solution – specific to GCP
  - Lift & Shift not possible, Not recommended
- Spanner = Cloud SQL + Horizontal Scalable
- Scale to petabyte
- Regional/ Multi-region level instance can be created
- Built on Google Cloud network
- Backup and Restore, point-in-time recovery
- Data export
  - can not export with gcloud
  - Cloud Console or Cloud Dataflow Job

Spanner vs RDBMS-SQL

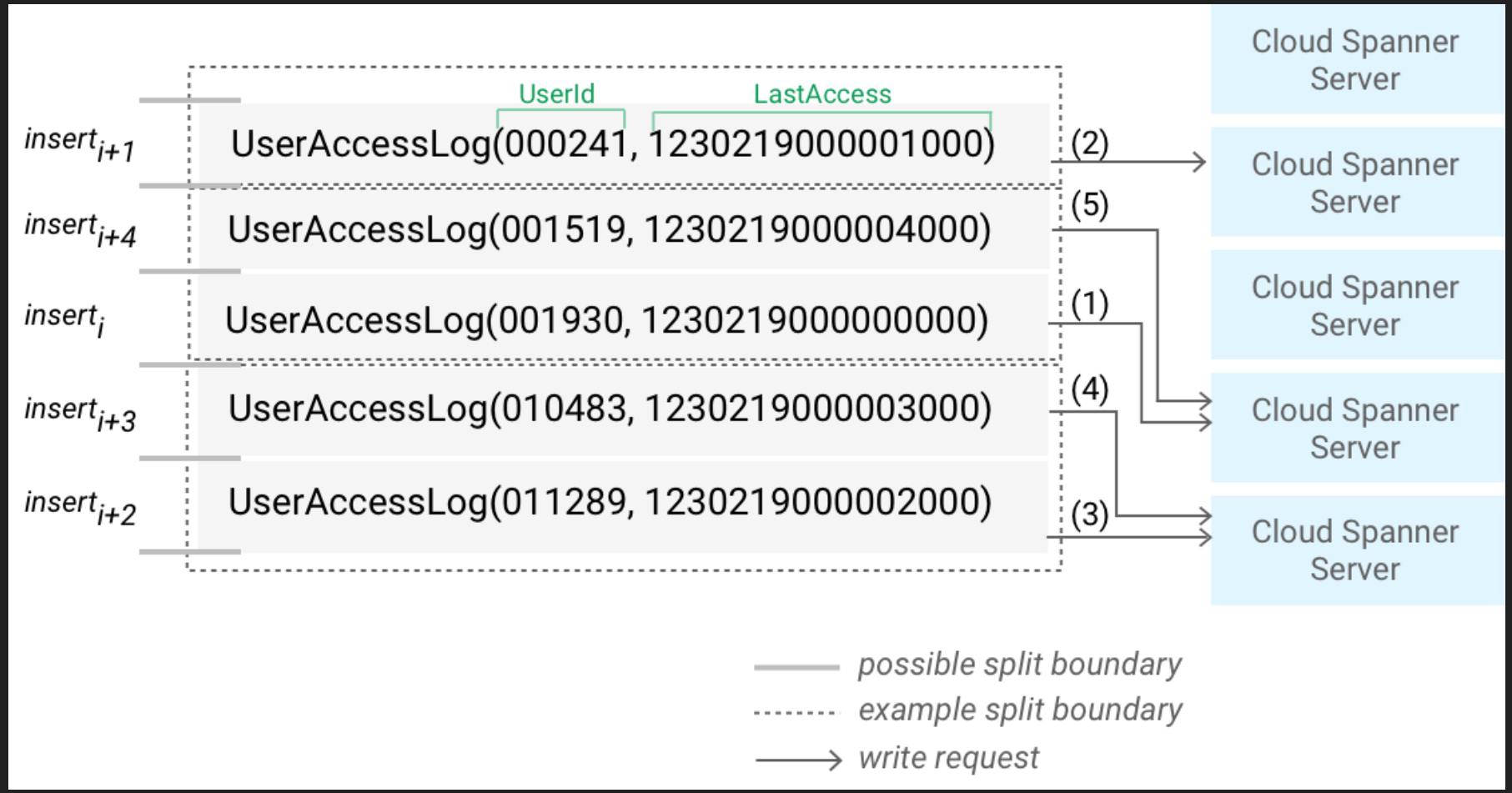
	Spanner	Cloud SQL
Availability	High	During failover little downtime
Scalable	Horizontal	vertical
Price	Costly	Cheaper than spanner
SQL/Schema Support	yes	yes
Replication	High	Only Read Replica

Avoid HotSpots

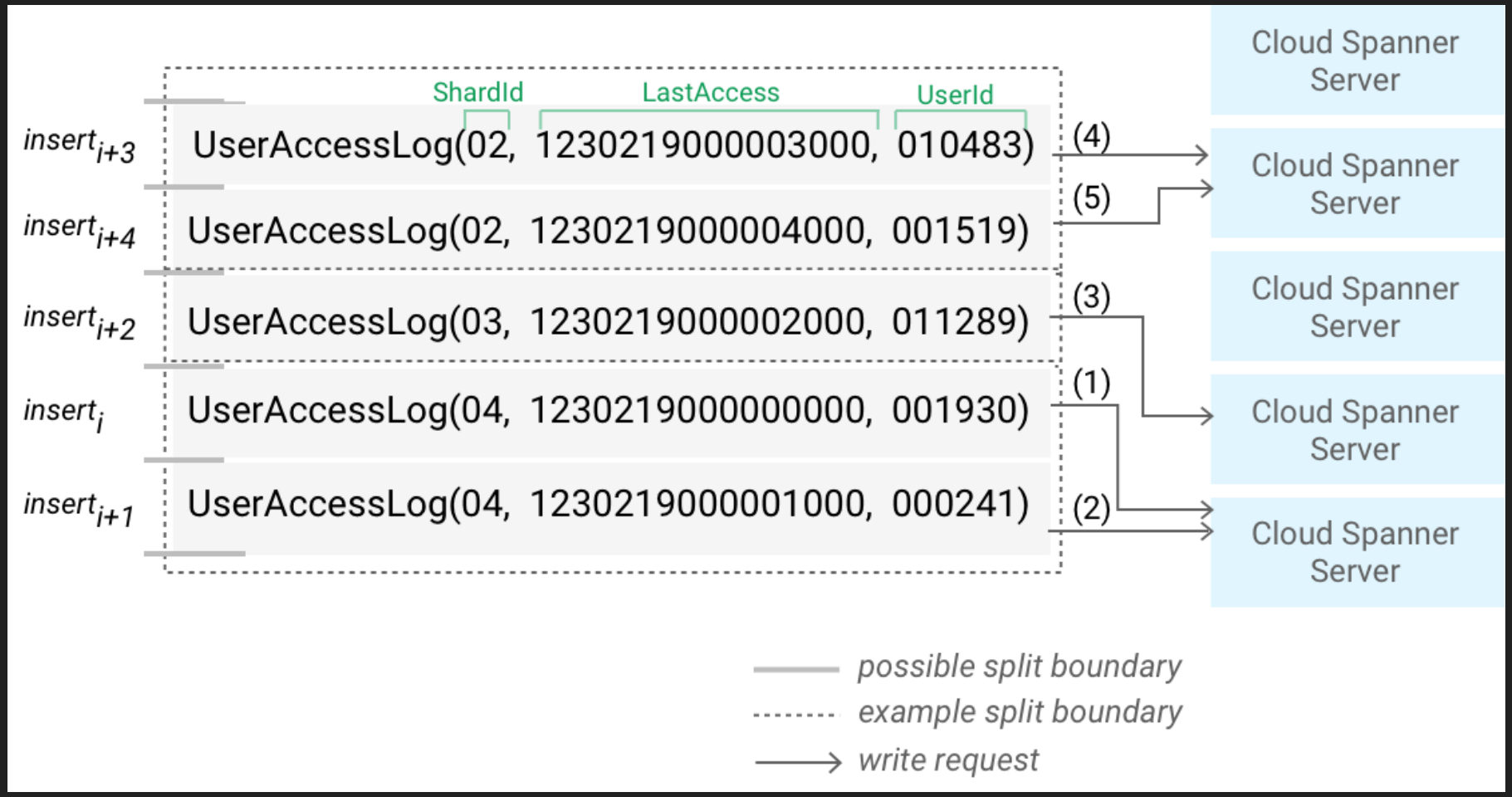
- Choosing Primary is very important.
- https://cloud.google.com/spanner/docs/schema-design



Avoid HotSpots



Avoid HotSpots



[Hands-on] Cloud Spanner

- Create Spanner Instance
- Create database lib_db
- Create 2 Table
  - Author
    - AId
    - AName
  - Book
    - AId
    - BId
    - BTitle
- Insert records in Table
- Fetch same record with Python code

After Job Done make sure to delete Spanner Instance

AlloyDB for PostgreSQL

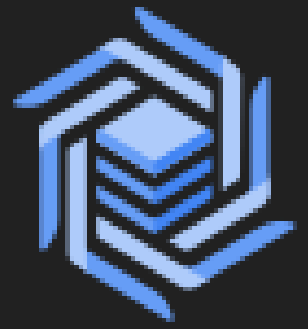
AlloyDB for PostgreSQL

- Fully managed PostgreSQL compatible database
- 4X faster than standard PostgreSQL for transactional load
- 100X faster analytical queries than standard PostgreSQL
- Works for OLTP & OLAP application
- Simplified management with machine learning-enabled autopilot systems
- AlloyDB offers a 99.99% uptime SLA
- Disaggregation of compute and storage
- In-built ML functionality
- https://cloud.google.com/blog/products/databases/alloydb-for-postgresql-intelligent-scalable-storage
- Let’s create AlloyDB instance

NoSQL In GCP

Datastore

Firestore



BigTable

MemoryStore

Datastore & FireStore

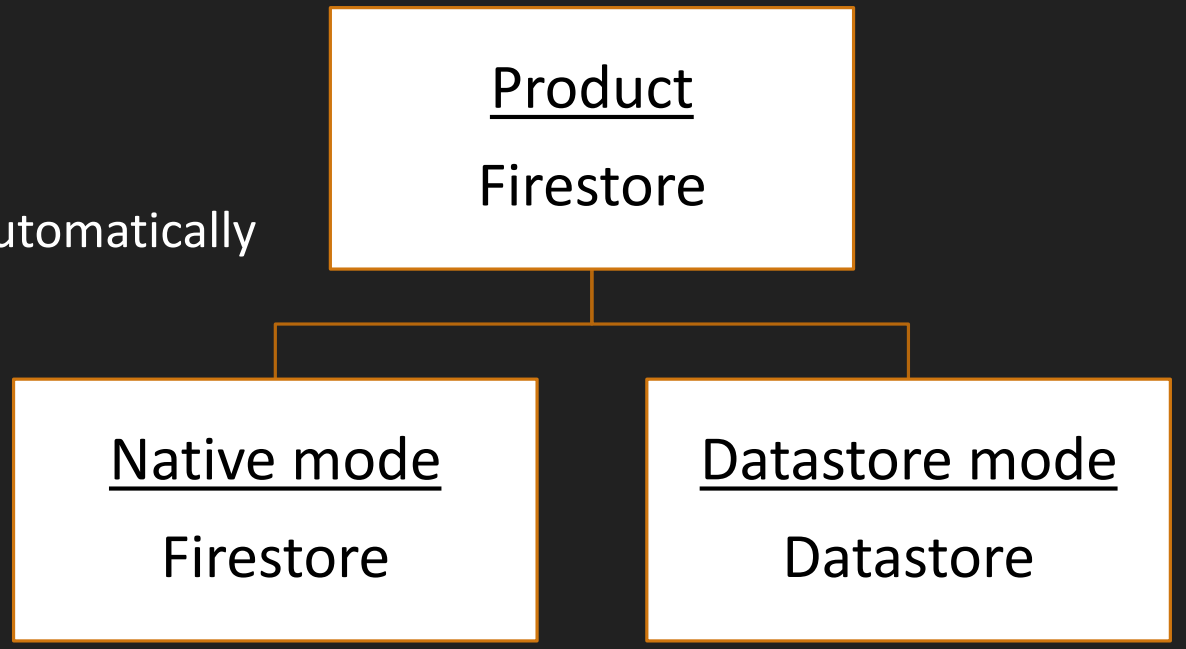
BY ANKIT MISTRY

History

- Datastore is one feature with Google App Engine in 2008
- Cloud Datastore was announced as a standalone product in 2013
- In October 2014, Firebase was acquired by Google – Web Mobile backend Database
- In October 2017, Firebase launched Cloud Firestore
- In 2018, the second-generation Firestore database was opened to general availability.

Naming in GCP

- In Single Project one can use either Datastore or Firestore
- You can not use both in single project
- Once you have selected mode + inserted data, you can not go back
- Still you want to use, need to create another project
- Once region is selected, can not be changed later
- In the future, all existing Datastore databases will be automatically upgraded to Firestore in Datastore mode.



Cloud Datastore

- Highly scalable NoSQL database for web and mobile
- Firestore is the next generation of Datastore.
- Serverless Fully managed – No management required
- Select Region & start putting your data
- Schemaless
- ACID transactions support - so can be used for transactional app like Banking
- For Query - SQL-like queries, GQL
- Multiple indexes Support
- Nicely integrated with App Engine
- Use case
  - Session Info, Product catalog

Datastore	RDBMS
Kind	Table
Entity	Row
Property	Column
Key	Primary Key

[Hands-on] Cloud Datastore

- Select Region
- Create some kind & entity from Console
- Execute GQL Query on top
- Demo on indexing
- Access Datastore Operation from Python
  - pip install google-cloud-Datastore
  - https://cloud.google.com/docs/samples?language=python&product=datastore

Cloud Firestore

- Firestore is the next generation of Datastore
- Highly scalable NoSQL database
- Collection & Document Model
- Two mode
  - Native Mode
  - Datastore mode
- Real-time updates
- Mobile and Web client libraries
- Let’s see in Action

Firestore	RDBMS
Collection group	Table
Document	Row
Field	Column
Document ID	Primary Key

Datastore & Firestore Pricing

Google Cloud BigTable

BY ANKIT MISTRY

Cloud BigTable

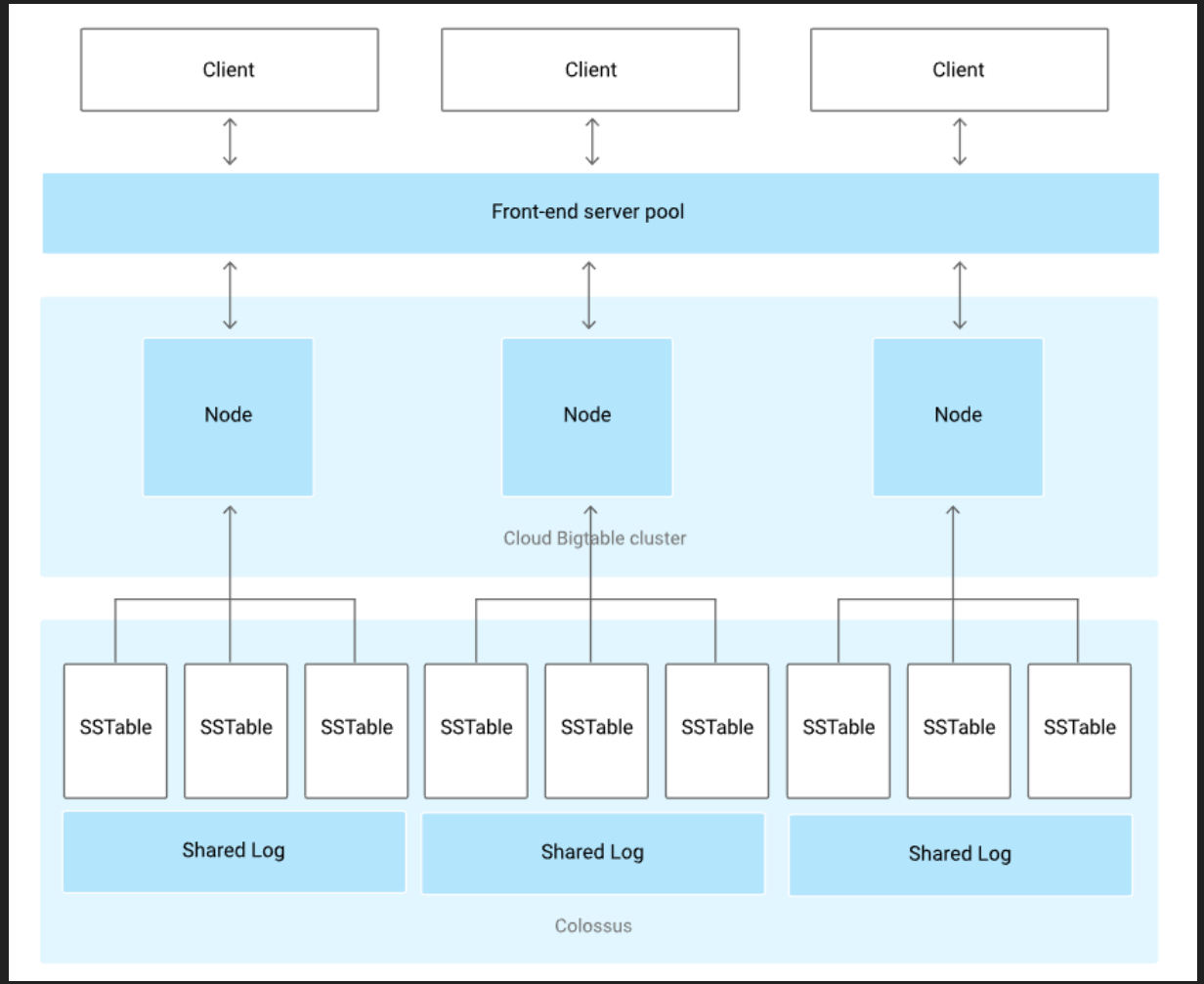
- Wide column NOSQL database
  - Sparse Multi-dimensional array
- Fully managed But Not serverless
- Scale horizontally with Multiple Node
- Scale to huge Volume of data - Petabyte scale
- Data stored at column wise
- Column are grouped into column family
- Low latency
- Handles millions of request per second
- How to access
  - cbt – command line (part of Google cloud SDK)
  - Hbase API
- No Multi column index
  - Only Row key based indexing – key based reading
- Seamless integration with
  - Warehouse – BigQuery, ML Product
- Equivalent Open Source DB - Cassandra, Hbase
- Some use case in
  - Financial, IOT
  - Time series Data

Cloud BigTable Storage

Row Key	Personal_data_cf		Professional_data_cf		
	name	age	salary	designation	company
1					
2					
3					

Professional_data_cf:salary

BigTable Architecture



Row key Design

- Design Row Key is very important
- Row key is equivalent to primary key in Table in RDBMS
- No Join, No Foreign Key support
- Rows are sorted by rowkeys
- Design Row key by keeping in your mind
  - which is your frequent query in application
  - No Hot spotting
  - Don’t use monotonically increasing key
  - Low cardinality attributes
  - Join multiple attribute for key
  - Design such way data distributed evenly across node
- Bigtable uses Colossus file system for data storage
- Bigtable instance store only metadata

[Hands-on]
Google Cloud BigTable

BigTable Pricing

Google Cloud Memorystore

Cloud Memorystore

- Fully managed In-memory database for Redis and Memcached
- sub-millisecond data access – very very low latency
- Two engine supported
  - Redis
  - Memcached
- zero code change require when you lift & shift to cloud
- Protected via Internal IP
- Highly scalable
- Highly available with 99.9% SLA
- Import/Export data from Cloud Storage to memory store
- Used when real-time responses required ,
  - finance, telecom, Health care, Gaming, leaderboard

[Hands-on] Memorystore - Redis

- Create Memorystore – Redis instance
- Create Client VM to connect in same network
- Install Redis client
  - sudo apt install redis-tools
- Redis command to store & retrieve data
- Export data to GCS
- Importing Data from GCP to another Memorystore Redis instance

Memorystore – Memcached

- Create Memorystore – Memcached instance
- Create Client VM to connect in same network
- Install telnet client
  - sudo apt install telnet
- Memchached command to store & retrieve data
- Export & import data to GCS

Database Migration

THANK YOU