Cloud Storage, Cloud Datastore, Cloud Bigtable, Cloud SQL, and Cloud Spanner

Applications can have various types of data. For example, if your application is a social image sharing site, you have to store image files, high volumes of use of messages, and transactional data. You'll likely need to cache frequently access data as well. And of course, you'll want to collect query and analyze all the data to gather business intelligence about your users as well as product usage patterns.

Google Cloud platform offers a number of managed services that you can use for each type of data. In the module, data storage options, you'll learn about services such as**, Cloud Storage, Cloud Datastore, Cloud Bigtable, Cloud SQL, Spanner and BigQuery**. You'll learn the ideal use cases for each data storage option as well as use cases for which the option might not be suitable. Armed with this knowledge, you can choose the data storage option that meets the specific use case in your application. A common problem that application developers face is handling database connections in a secure manner. You'll learn how to connect your Cloud SQL second generation instance without having to whitelist IP addresses or configure SSL.

This is an overview of data storage options available to you in GCP. You have a full suite of cost effective storage services to choose from when developing with Google Cloud platform. No one size fits all. And your choice of storage and database solutions will depend on your application and workload.

Cloud Storage. Cloud storage is a unified object storage for developers and enterprises that allows you to serve, analyze, and archive data. Objects are accessed via HTTP requests including ranged gets to retrieve portions of the data. The only key is the object name. There is object metadata, but the object itself is treated as just bytes with no structure. The scale of the system allows for serving static content or accepting user uploaded content like photos and videos. Cloud Storage is built for availability, durability, scalability, and consistency. It is an ideal solution for storing images and videos, objects and blobs, and any unstructured data. For more information see the cloud storage documentation available in the download panes below.

Bigtable is a high performance, NoSQL database service. It is a sparsely populated table that can scale to billions of rows and thousands of columns. Bigtable can store terabytes to petabytes of data. Similar to Apache Hbase, Bigtable is built for fast **key-value** look up and scanning over a defined key range. It is similar to a spreadsheet that gives you access to any set of columns from contiguous rows by searching only the value in the first column, the key. Updates to individual rows are atomic. Due to the fast look up and right speed of Bigtable, it's great for user behavior. Bigtable supports operational and analytical applications and is ideal for storing large amounts of single-keyed data and MapReduce operations.

Cloud Datastore is GCP's fully managed, NoSQL document database. It is designed to automatically scale to very large datasets. So your application can maintain high performance as they receive more traffic. Datastore adds many features on top of Bigtable. Like managing multiple indexes over each entity, crossroad transactions, and regional replication. This often makes it a better choice for high value application data like user profiles or shopping carts and orders. Datastore is an ideal solution when your application data is semi-structured or hierarchical and when you need to store durable key-value data.

Cloud SQL is GCP's relational database service. It is managed service that lets Google manage replication, fail over, and backups of your databases so you can focus on your MySQL or PostgreSQL compatible applications. Cloud SQL lets you easily configure replication and backups to protect your data. You can replicate a master instance to one or more read replicas. A read replica is a copy of the master that reflects changes to the master instance in almost real time. You can enable automatic fail over to make your database highly available. Backups allow you to restore your Cloud SQL instance to recover lost data or recover from a problem with your instance. You can enable automated backups for any instance that contains data that you need to protect from loss or damage. The Cloud SQL Proxy works by having a local client called the proxy running in the local environment. Your application communicates with the proxy with a standard database protocol used by your database. The proxy uses a secure channel to communicate with its companion process running on the server.

Cloud SQL Proxy provides secure access to your Cloud SQL second generation instances without you having to whitelist IP addresses or configure SSL. The proxy uses the Cloud SQL API to authenticate with the Google Cloud platform. You must enable the API before using the proxy, and you must provide the proxy with a valid user account. Cloud SQL is ideal for web frameworks, applications requiring structured data, and online transaction processing, OLTP workloads. It is ideal for applications using MySQL, PostgreSQL with minimal refactoring required for migration to GCP.

Cloud Spanner is GCP's fully managed relational database service offering both strong consistency and horizontal scalability. It is designed for mission critical OLTP applications. Cloud Spanner provides automatic synchronous replication for high availability. Spanner is built for multi-region replication and offers one of the highest SLAs in the industry, 99.999 percent. Spanner is ideal for applications with relational structured and semi-structured data that require high availability, strong consistency, and transactional reads and writes. Cloud Spanner and Cloud SQL differ in that, Spanner uses real primary keys and has the notion of interleaved child records instead of foreign keys. Some considerations to keep in mind when working from Spanner include the following. Avoid creating hotspots in your database in Cloud Spanner when choosing a primary key. Hotspots can be created by choosing a column whose value monotonically increases as the first key part because it results in all inserts occurring at the end of your key space. Cloud Spanner divides data among servers by key ranges. So all your inserts will be directed at a single server. This is pictured in the first diagram. Possible fixes to this situation includes, swapping the order of the keys or hashing the key and spreading the rights among n-shards. For more information please see the linked resource available in the download pane below. An interleaved table is a table that you declare to be a child of another table because you want the rows of the child table to be physically stored together with the associated parent row. The prefix of the primary key of a child table must be the primary key of the parent table. The example shows how to define a table albums as a child of singers or singers is at the root of the database hierarchy. When loading data in Clouds Spanner, make sure that rights are well distributed and load the data using multiple workers. Avoid creating non-interleaved indexes on columns with monotonically increasing or decreasing keys. And create indexes after you bulk-load your data.