

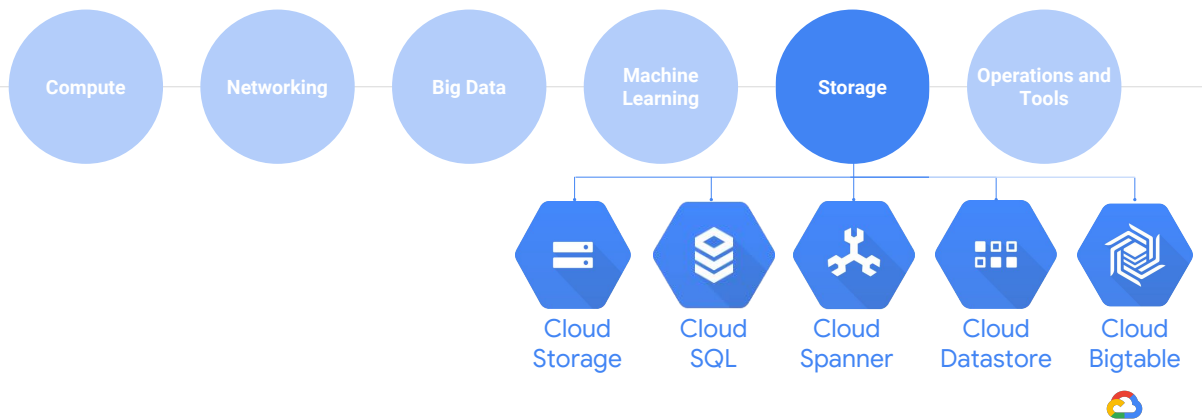


---

GCP Fundamentals: Core Infrastructure

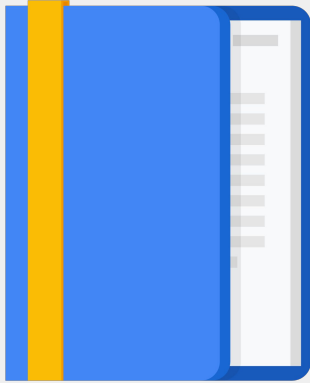
**Storage in the Cloud**

# Google Cloud Platform



Google Cloud Platform has many storage options that satisfy nearly every customer use case. In this module, we turn our attention to the core storage options: Google Cloud Storage, Google Cloud SQL, Google Cloud Spanner, Cloud Datastore, and Google Cloud Bigtable.

# Agenda



- **Cloud Storage**
- Cloud Bigtable
- Cloud SQL and Cloud Spanner
- Cloud Datastore
- Comparing storage options
- Quiz and Lab



## Cloud Storage is binary large-object storage

- High performance, internet-scale
  - Simple administration
- Does not require capacity management
- Data encryption at rest
- Data encryption in transit by default from Google to endpoint
- Online and offline import services are available



Google Cloud Storage offers developers and IT organizations durable and highly available object storage. It assesses no minimum fee; you pay only for what you use. Prior provisioning of capacity isn't necessary.

What's object storage? It's not the same as file storage, in which you manage your data as a hierarchy of folders. It's not the same as block storage, in which your operating system manages your data as chunks of disk. Instead, object storage means this: you say to your storage, "Here, keep this arbitrary sequence of bytes,," and the storage lets you address it with a unique key. In Google Cloud Storage and in other systems, these unique keys are in the form of URLs, which means object storage interacts well with web technologies.

Google Cloud Storage always encrypts your data on the server side, before it is written to disk, at no additional charge. Data traveling between a customer's device and Google is encrypted by default using HTTPS/TLS (Transport Layer Security). In fact, Google was the first major cloud provider to enable HTTPS/TLS by default.

Google Cloud Storage is not a file system, although it can be accessed as one via third-party tools such as Cloud Storage FUSE. The storage objects offered by Google Cloud Storage are "immutable," which means that you do not edit them in place, but instead create a new version. Google Cloud Storage's primary use is whenever binary

large-object storage is needed: online content, backup and archiving, storage of intermediate results in processing workflows, and more.

Offline Media Import/Export is a third-party solution that allows you to load data into Google Cloud Storage by sending your physical media, such as hard disk drives (HDDs), tapes, and USB flash drives, to a third-party service provider who uploads data on your behalf. Offline Media Import/Export is helpful if you're limited to a slow, unreliable, or expensive internet connection.

Offline import is available through third-party providers:

<https://cloud.google.com/storage/docs/offline-media-import-export>

Cloud Storage Transfer Service enables you to import large amounts of online data into Google Cloud Storage quickly and cost-effectively. To use Cloud Storage Transfer Service, you set up a transfer from a data source to data sink. Data sources can be an Amazon Simple Storage Service (Amazon S3) bucket, an HTTP/HTTPS location, or another Google Cloud Storage bucket. Data sinks are always a Google Cloud Storage bucket.

Example uses of Cloud Storage Transfer Service include:

Backing up data to a Google Cloud Storage bucket from other storage providers.

Moving data from a Standard Storage bucket to a Nearline Storage bucket to lower your storage costs.

Your Cloud Storage files are organized into **buckets**

Bucket attributes	Bucket contents
Globally unique name	Files (in a flat namespace)
Storage class	
Location (region or multi-region)	
IAM policies or Access Control Lists	Access Control Lists
Object versioning setting	
Object lifecycle management rules	



Your Cloud Storage files are organized into buckets. When you create a bucket: you give it a globally-unique name; you specify a geographic location where the bucket and its contents are stored; and you choose a default storage class. Pick a location that minimizes latency for your users. For example, if most of your users are in Europe, you probably want to pick a European location: a GCP region in Europe, or else the EU multi-region.

There are several ways to control users' access to your objects and buckets. For most purposes, Cloud IAM is sufficient. Roles are inherited from project to bucket to object. If you need finer control, you can create access control lists ("ACLs") that offer finer control, ACLs define who has access to your buckets and objects, as well as what level of access they have. Each ACL consists of two pieces of information: A scope, which defines who can perform the specified actions (for example, a specific user or group of users). And a permission, which defines what actions can be performed (for example, read or write).

Remember that Cloud Storage objects are immutable. You can turn on object versioning on your buckets if you want. If you do, Cloud Storage keeps a

history of modifications—that is, overwrites or deletes—of all objects in the bucket. You can list the archived versions of an object, restore an object to an older state, or permanently delete a version, as needed. If you don't turn on object versioning, new always overwrites old.

Cloud Storage also offers lifecycle management policies. For example, you could tell Cloud Storage to delete objects older than 365 days, or to delete objects created before January 1, 2013; or to keep only the 3 most recent versions of each object in a bucket that has versioning enabled.

## Choosing among Cloud Storage classes

	Multi-regional	Regional	Nearline	Coldline
Intended for data that is...	Most frequently accessed	Accessed frequently within a region	Accessed less than once a month	Accessed less than once a year
Availability SLA	99.95%	99.90%	99.00%	99.00%
Access APIs	Consistent APIs			
Access time	Millisecond access			
Storage price	Price per GB stored per month			
Retrieval price	Total price per GB transferred			
Use cases	Content storage and delivery	In-region analytics, transcoding	Long-tail content, backups	Archiving, disaster recovery



Cloud Storage lets you choose among four different types of storage classes: Regional, Multi-regional, Nearline and Coldline. Multi-regional and Regional are high-performance object storage, whereas Nearline and Coldline are backup and archival storage. All of the storage classes are accessed in analogous ways using the Cloud Storage API, and they all offer millisecond access times.

**Regional Storage** lets you store your data in a specific GCP region, us-central1, europe-west1 or asia-east1. It's cheaper than multi-regional storage, but it offers less redundancy.

**Multi-Regional Storage** costs a bit more, but it's geo-redundant. That means you pick a broad geographical location, like United States, the European Union, or Asia, and Cloud Storage stores your data in at least two geographic locations separated by at least 160 kilometers.

Multi-Regional Storage is appropriate for storing frequently accessed storing data: website content, interactive workloads, or data that's part of mobile and gaming applications. People use regional storage, on the other hand, to store data close to their Compute Engine virtual machines or their Kubernetes Engine clusters. That gives better performance for data-intensive computations.



**Nearline storage** is a low-cost, highly durable storage service for storing infrequently accessed data. This storage class is a better choice than Multi-Regional Storage or Regional Storage in scenarios where you plan to read or modify your data on average once a month or less. For example, if you want to continuously add files to Cloud Storage and plan to access those files once a month for analysis, Nearline Storage is a great choice.

**Coldline Storage** is a very-low-cost, highly durable storage service for data archiving, online backup, and disaster recovery. Coldline Storage is the best choice for data that you plan to access at most once a year, due to its slightly lower availability, 90-day minimum storage duration, costs for data access, and higher per-operation costs. For example, if you want to archive data or have access in the event of a disaster recovery event.

The availability of these storage classes varies, with multi-regional having the highest availability of 99.95%, followed by regional with 99.9% and nearline and coldline with 99.0%.

As for pricing, all storage classes incur a cost per gigabyte of data stored per month, with multi-regional having the highest storage price and coldline the lowest storage price. Egress and data transfer charges may also apply.

In addition to those charges, Nearline storage also incurs an access fee per gigabyte of data read, and Coldline storage incurs a higher fee per gigabyte of data read.

There are several ways to bring data into Cloud Storage



### Online transfer

Self-managed copies using command-line tools or drag-and-drop



### Storage Transfer Service

Scheduled, managed batch transfers



### Transfer Appliance <sup>Beta</sup>

Rackable appliances to securely ship your data



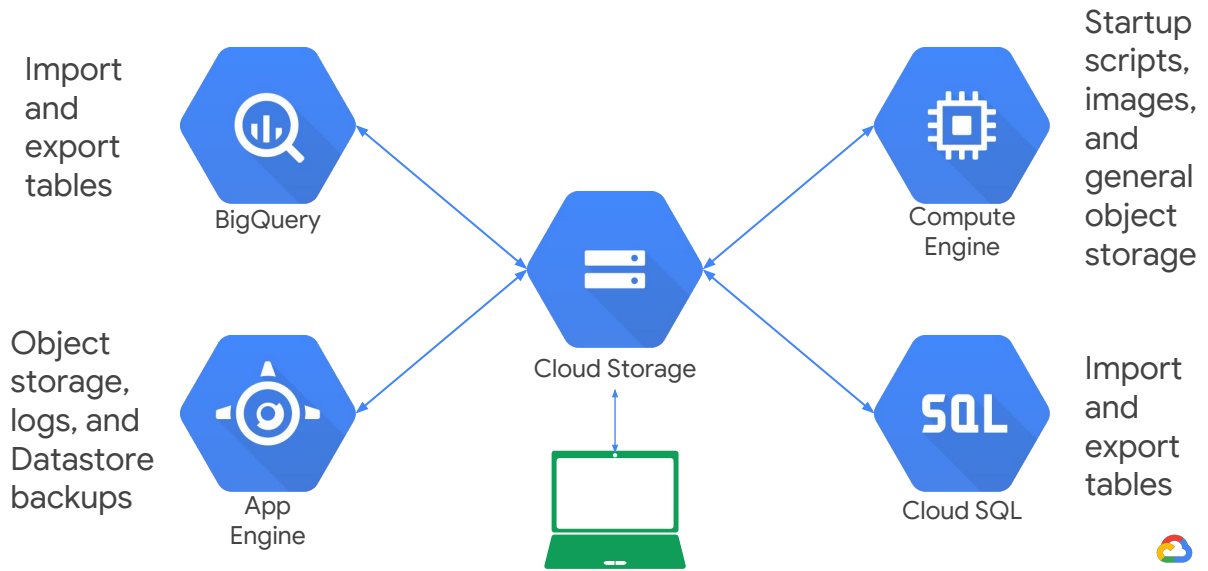
Regardless of which storage class you choose, there are several ways to bring data into Cloud Storage.

Many customers simply use `gsutil`, which is the Cloud Storage command from the Cloud SDK. You can also move data in with a drag and drop in the GCP Console, if you use the Google Chrome browser. But what if you have to upload terabytes or even petabytes of data? Google Cloud Platform offers the online Storage Transfer Service and the offline Transfer Appliance to help.

The Storage Transfer Service lets you schedule and manage batch transfers to Cloud Storage from another cloud provider, from a different Cloud Storage region, or from an HTTP(S) endpoint.

The Transfer Appliance is a rackable, high-capacity storage server that you lease from Google Cloud. You simply connect it to your network, load it with data, and then ship it to an upload facility where the data is uploaded to Cloud Storage. The service enables you to securely transfer up to a petabyte of data on a single appliance. As of this recording, it's still beta, and it's not available everywhere, so check the website for details.

## Cloud Storage works with other GCP services



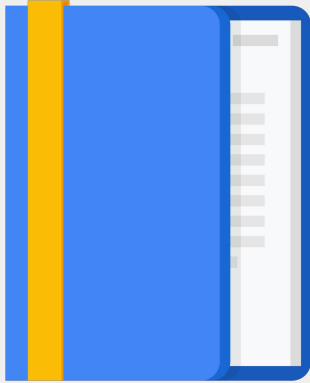
There are other ways of getting your data into Cloud Storage, as this storage option is tightly integrated with many of the Google Cloud Platform products and services.

For example, you can import and export tables from and to BigQuery, as well as Cloud SQL.

You can also store App Engine logs, Cloud Datastore backups, and objects used by App Engine applications like images. Cloud Storage can also store instance startup scripts, Compute Engine images, and objects used by Compute Engine applications.

In short, Cloud Storage is often the ingestion point for data being moved into the cloud, and is frequently the long-term storage location for data.

# Agenda



- Cloud Storage
- **Cloud Bigtable**
- **Cloud SQL** and **Cloud Spanner**
- Cloud Datastore
- Comparing storage options
- Quiz and Lab



## Cloud Bigtable is managed NoSQL

- Fully managed NoSQL, wide-column database service for terabyte applications
- Integrated
  - Accessed using HBase API
  - Native compatibility with big data, Hadoop ecosystems



Cloud Bigtable is Google's NoSQL big data database service. It's the same database that powers many core Google services, including Search, Analytics, Maps, and Gmail.

## Why choose Cloud Bigtable?

- Replicated storage
- Data encryption in-flight and at rest
- Role-based ACLs
- Drives major applications such as Google Analytics and Gmail



Customers frequently choose Bigtable if the data is:

### *Big*

- Large quantities (>1 TB) of semi-structured or structured data

### *Fast*

- Data is high throughput or rapidly changing

### *NoSQL*

- Transactions, strong relational semantics not required

And especially if it is:

### *Time series*

- Data is time-series or has natural semantic ordering

### *Big data*

- You run asynchronous batch or real-time processing on the data

### *Machine learning*

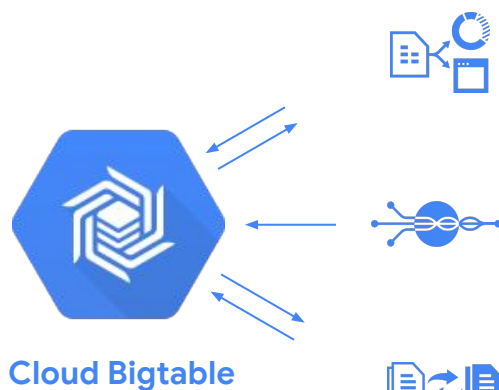
- You run machine learning algorithms on the data

Bigtable is designed to handle massive workloads at consistent low latency and high throughput, so it's a great choice for both operational and analytical applications, including IoT, user analytics, and financial data analysis.

For more information on Cloud Bigtable, see

<https://www.google.com/url?q=https://cloudplatform.googleblog.com/2015/05/introducing-Google-Cloud-Bigtable.html>

## Bigtable Access Patterns



### Application API

Data can be read from and written to Cloud Bigtable through a data service layer like Managed VMs, the HBase REST Server, or a Java Server using the HBase client. Typically this will be to serve data to applications, dashboards, and data services.

### Streaming

Data can be streamed in (written event by event) through a variety of popular stream processing frameworks like Cloud Dataflow Streaming, Spark Streaming, and Storm.

### Batch Processing

Data can be read from and written to Cloud Bigtable through batch processes like Hadoop MapReduce, Dataflow, or Spark. Often, summarized or newly calculated data is written back to Cloud Bigtable or to a downstream database.



As Cloud Bigtable is part of the GCP ecosystem, it can interact with other GCP services and third-party clients.

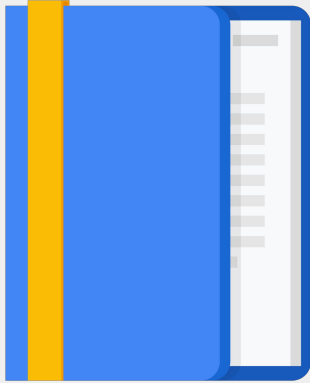
From an application API perspective, data can be read from and written to Cloud Bigtable through a data service layer like Managed VMs, the HBase REST Server, or a Java Server using the HBase client. Typically this will be to serve data to applications, dashboards, and data services.

Data can also be streamed in through a variety of popular stream processing frameworks like Cloud Dataflow Streaming, Spark Streaming, and Storm.

If streaming is not an option, data can also be read from and written to Cloud Bigtable through batch processes like Hadoop MapReduce, Dataflow, or Spark. Often, summarized or newly calculated data is written back to Cloud Bigtable or to a downstream database.



# Agenda



- Cloud Storage
- Cloud Bigtable
- **Cloud SQL and Cloud Spanner**
- Cloud Datastore
- Comparing storage options
- Quiz and Lab



## Cloud SQL is a managed **RDBMS**

- Offers **MySQL** and **PostgreSQL** databases as a service
- Automatic replication
- Managed backups
- **Vertical scaling (read and write)**
- **Horizontal scaling (read)**
- Google security



Cloud SQL is an easy-to-use service that delivers fully managed relational databases. Cloud SQL lets you hand off to Google the mundane, but necessary and often time-consuming tasks—like applying patches and updates, managing backups, and configuring replications—so you can put your focus on building great applications.

Every Cloud SQL instance includes a network firewall, allowing you to control network access to your database instance by granting access.

Cloud SQL is easy to use: it doesn't require any software installation or maintenance.

Easily scale up to 64 processor cores and more than 100 GB of RAM. Quickly scale out with read replicas.

### *Automatic replication*

Google Cloud SQL supports the following read replica scenarios:

- Cloud SQL instances replicating from a Cloud SQL master instance  
Replicas are other instances in the same project and location as the master instance. This feature is in Beta.
- Cloud SQL instances replicating from an external master instance  
The master instance is external to Google Cloud SQL. For example, it

- can be outside the Google network or in a Google Compute Engine instance. This feature is in Beta.
- External MySQL instances replicating from a Cloud SQL master instance  
External replicas are in hosting environments, outside of Cloud SQL.

### *Managed backups*

Cloud SQL takes care of securely storing your backed-up data and makes it easy for you to restore from a backup and perform a point-in-time recovery to a specific state of an instance. Cloud SQL retains up to 7 backups for each instance, which is included in the cost of your instance.

Cloud SQL customer data is encrypted when on Google's internal networks and when stored in database tables, temporary files, and backups.

(MySQL instances are available in either First Generation or Second Generation. Google recommends the use of Second Generation instances for most use cases. First Generation instances are recommended primarily when MySQL 5.5 compatibility is required. Also, First Generation instances may be cost-effective for infrequently used or test/dev database instances, because of their available Per-Use billing plan and the available ON DEMAND activation policy, which causes your instance to automatically shut itself off after 15 minutes of inactivity.)

## Cloud SQL can be used with other GCP services



Cloud SQL can be used with App Engine using standard drivers.

You can configure a Cloud SQL instance to follow an App Engine application.



Compute Engine instances can be authorized to access Cloud SQL instances using an external IP address.

Cloud SQL instances can be configured with a preferred zone.



Cloud SQL can be used with external applications and clients.

Standard tools can be used to administer databases.

External read replicas can be configured.



Another benefit of Cloud SQL instances is that they are accessible by other GCP services and even external services. You can use Cloud SQL with App Engine using standard drivers like Connector/J for Java or MySQLdb for Python.

You can authorize Compute Engine instances to access Cloud SQL instances and configure the Cloud SQL instance to be in the same zone as your virtual machine.

Cloud SQL also supports other applications and tools that you might be used to, like SQL Workbench, Toad and other external applications using standard MySQL drivers.

Cloud **Spanner** is a **horizontally scalable RDBMS**

Cloud Spanner supports:

- Automatic replication
- **Strong global consistency**
- Managed instances with **high availability**
- SQL (ANSI 2011 with extensions)



Cloud Spanner supports strong consistency, including strongly consistent secondary indexes, SQL, and managed instances with high availability through synchronous and built-in data replication. Battle tested by Google's own mission-critical applications and services, Spanner powers Google's \$80 billion business.

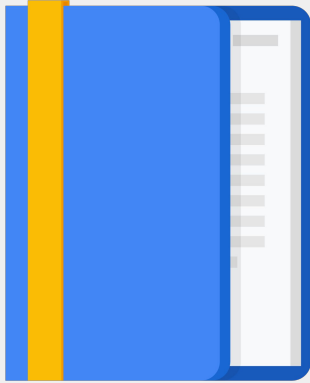
Cloud Spanner is especially suited for applications requiring:

- A SQL RDBMS, with joins and secondary indexes
- Built-in high availability
- Strong global consistency
- Database sizes exceeding ~2 TB
- Many IOPS (Tens of thousands of reads/writes per second or more)

For a technical overview of Cloud Spanner, see

<https://cloudplatform.googleblog.com/2017/02/inside-Cloud-Spanner-and-the-CAP-Theorem.html>.

# Agenda



- Cloud Storage
- **Cloud Bigtable**
- Cloud SQL and Cloud Spanner
- **Cloud Datastore**
- Comparing storage options
- Quiz and Lab



**Cloud Datastore** is a horizontally scalable NoSQL DB

- [NoSQL](#) designed for application backends
- Fully managed
  - Uses a distributed architecture to automatically manage scaling
- Built-in redundancy
- Supports [ACID](#) transactions



Cloud Datastore is a highly-scalable NoSQL database for your applications. Like Cloud Bigtable, there is no need for you to provision database instances. Cloud Datastore uses a distributed architecture to automatically manage scaling. Your queries scale with the size of your result set, not the size of your data set.

Cloud Datastore runs in Google data centers, which use redundancy to minimize impact from points of failure. Your application can still use Cloud Datastore when the service receives a planned upgrade.

The total size of Cloud Datastore databases can grow to terabytes and more.

## Google Cloud Datastore: benefits

- Schemaless access
  - No need to think about underlying data structure
- Local development tools
- Includes a free daily quota
- Access from anywhere through a [RESTful interface](#)



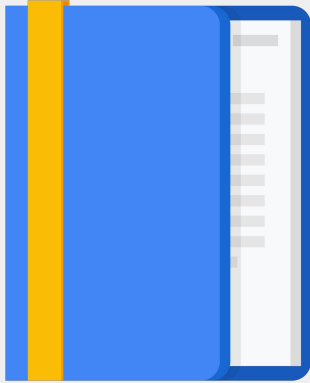
### Cloud Datastore features:

- *Atomic transactions*  
Datastore can execute a set of operations where either all succeed, or none occur.
- *High availability of reads and writes*  
Datastore runs in Google data centers, which use redundancy to minimize impact from points of failure.
- *Massive scalability with high performance*  
Datastore uses a distributed architecture to automatically manage scaling. Datastore uses a mix of indexes and query constraints so your queries scale with the size of your result set, not the size of your data set.
- *Flexible storage and querying of data*  
Datastore maps naturally to object-oriented and scripting languages and is exposed to applications through multiple clients. It also provides a SQL-like [query language](#).
- *Balance of strong and eventual consistency*  
Datastore ensures that entity lookups and ancestor queries always receive strongly consistent data. All other queries are eventually consistent. The consistency models allow your application to deliver a great user experience while handling large amounts of data and users.



- *Encryption at rest*  
Datastore automatically encrypts all data before it is written to disk and automatically decrypts the data when read by an authorized user. For more information, see [Server-Side Encryption](#).
- *Fully managed with no planned downtime*  
Google handles the administration of the Datastore service so you can focus on your application. Your application can still use Datastore when the service receives a planned upgrade.

# Agenda



- Cloud Storage
- Cloud Bigtable
- Cloud SQL and Cloud Spanner
- Cloud Datastore
- **Comparing storage options**
- Quiz and Lab



## Comparing storage options: technical details

	Cloud Datastore	Bigtable	Cloud Storage	Cloud SQL	Cloud Spanner	BigQuery
Type	NoSQL document	NoSQL wide column	Blobstore	Relational SQL for OLTP	Relational SQL for OLTP	Relational SQL for OLAP
Transactions	Yes	Single-row	No	Yes	Yes	No
Complex queries	No	No	No	Yes	Yes	Yes
Capacity	Terabytes+	Petabytes+	Petabytes+	Up to ~10 TB	Petabytes	Petabytes+
Unit size	1 MB/entity	~10 MB/cell ~100 MB/row	5 TB/object	Determined by DB engine	10,240 MiB/row	10 MB/row



Now that we covered GCP's core storage options, let's compare them to help you choose the right service for your application or workflow.

This table focuses on the technical differentiators of the storage services. Each row is a technical specification and each column is a service. Let me cover each service from left to right.

Consider using Cloud Datastore, if you need to store structured objects, or if you require support for transactions and SQL-like queries. This storage services provides terabytes of capacity with a maximum unit size of 1 MB per entity.

Consider using Cloud Bigtable, if you need to store a large amount of structured objects. Cloud Bigtable does not support SQL queries, nor does it support multi-row transactions. This storage service provides petabytes of capacity with a maximum unit size of 10 MB per cell and 100 MB per row.

Consider using Cloud Storage, if you need to store immutable blobs larger than 10 MB, such as large images or movies. This storage service provides petabytes of capacity with a maximum unit size of 5 TB per object.

Consider using Cloud SQL or Cloud Spanner if you need full SQL support for an online transaction processing system. Cloud SQL provides up to Up to 10,230 GB, depending on machine type, while Cloud Spanner provides petabytes. If Cloud SQL does not fit your requirements because you need horizontal scalability, not just

through read replicas, consider using Cloud Spanner.

We didn't cover BigQuery in this module as it sits on the edge between data storage and data processing, but you will learn more about it in the "Big Data and Machine Learning in the Cloud" module. The usual reason to store data in BigQuery is to use its big data analysis and interactive querying capabilities. You would not want to use BigQuery, for example, as the backing store for an online application.

## Comparing storage options: use cases

	Cloud Datastore	Cloud Bigtable	Cloud Storage	Cloud SQL	Cloud Spanner	BigQuery
Type	NoSQL document	NoSQL wide column	Blobstore	Relational SQL for OLTP	Relational SQL for OLTP	Relational SQL for OLAP
Best for	Semi-structured application data, durable key-value data	"Flat" data, Heavy read/write, events, analytical data	Structured and unstructured binary or object data	Web frameworks, existing applications	Large-scale database applications (> ~2 TB)	Interactive querying, offline analytics
Use cases	Getting started, App Engine applications	AdTech, Financial and IoT data	Images, large media files, backups	User credentials, customer orders	Whenever high I/O, global consistency is needed	Data warehousing



Considering the technical differentiators of the different storage services helps some people decide which storage service to choose, others like to consider use cases. Let me go through each service one more time.

Cloud Datastore is best for semi-structured application data that is used in App Engine applications.

Bigtable is best for analytical data with heavy read and write events, like AdTech, financial or IoT data.

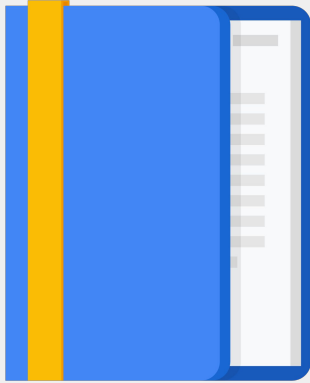
Cloud Storage is best for structured and unstructured binary or object data, like images, large media files and backups.

Cloud SQL is best for web frameworks and existing applications, like storing user credentials and customer orders.

Cloud Spanner is best for large-scale database applications that are larger than 2 TB. For example, for financial trading and e-commerce use cases.

As I mentioned at the beginning of the module, depending on your application you might use one or several of these services to get the job done.

# Agenda



- Cloud Storage
- Cloud Bigtable
- Cloud SQL and Cloud Spanner
- Cloud Datastore
- Comparing storage options
- **Quiz and Lab**



# Question #1

---

Your application transcodes large video files. Which storage service should you consider first?



# Question #1

---

Your application transcodes large video files. Which storage service should you consider first?

Google Cloud Storage





## Question #2

---

You stream huge amounts of data from devices with sensors. Which storage service should you consider first?



## Question #2

---

You stream huge amounts of data from devices with sensors. Which storage service should you consider first?

Google Cloud Bigtable



# Lab

---

In this lab you will store an image in a Cloud Storage bucket and configure GCE to use a Cloud SQL database to reference the image.

## Objectives

- Create a Cloud Storage bucket and place an image into it
- Create a Cloud SQL instance and configure it
- Connect to a Cloud SQL instance from a web server
- Use an image stored in a Cloud Storage bucket in a web page



In this lab you will create a Google Cloud Storage bucket and place an image in it. You'll also configure an application running in Google Compute Engine to use a database managed by Google Cloud SQL and to reference the image in the Cloud Storage bucket.

# More resources

---

Overview of Cloud Storage <https://cloud.google.com/storage/>

Getting started with Google Cloud SQL <https://cloud.google.com/sql/docs/quickstart>

Cloud Bigtable <https://cloud.google.com/stackdriver/docs/>

Cloud Spanner <https://cloud.google.com/spanner/docs/>

Cloud Datastore <https://cloud.google.com/datastore/docs/>

