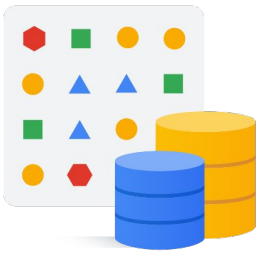
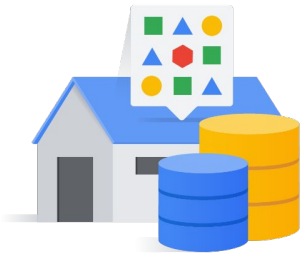
Database, data warehouses, and data lakes

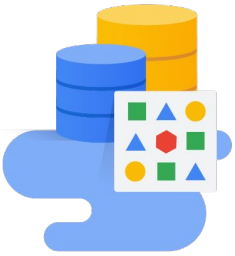
Organizations need a modern approach to managing vast volumes of data



Databases



Data warehouses



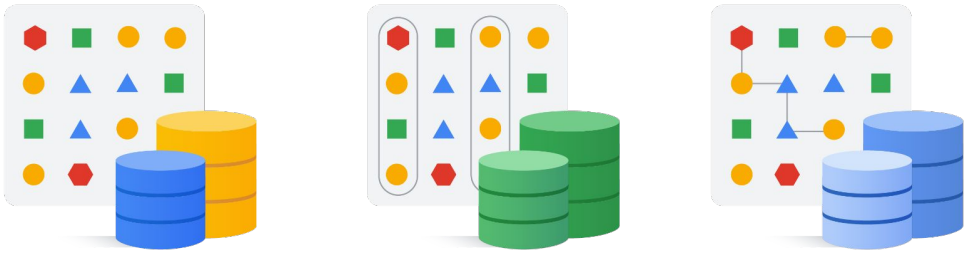
Data lakes

Organizations need a modern approach to enterprise data to manage the vast volumes that are produced. The list of options often includes **databases**, **data warehouses**, and **data lakes**.

Let's explore each of these options; starting with databases.

Database

An organized collection of data **stored in tables** and accessed electronically from a computer system.

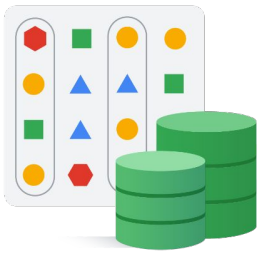


Databases → Relational databases | Non-relational databases

A **database** is an organized collection of data stored in tables and accessed electronically from a computer system.

Let's examine two types of databases: relational and non-relational.

Relational database



Relational databases

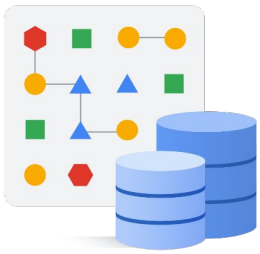
Stores data points in tables, rows, and columns that have a clearly defined schema.

- Is highly consistent and reliable
- Suited for large amounts of structured data.
- Is designed for business data processing.
- Is designed for storing online transactional data.

A **relational database** stores and provides access to data points that are related to one another. This means storing information in tables, rows, and columns that have a clearly defined schema that represents the structure or logical configuration of the database.

A relational database can establish links—or relationships–between information by joining tables, and structured query language, or SQL, can be used to query and manipulate data. Relational databases are highly consistent, reliable, and best suited for dealing with large amounts of structured data. They're designed for business data processing and storing the online transactional data needed to support the daily operations of a company.

Non-relational database



Non-relational databases (NoSQL)

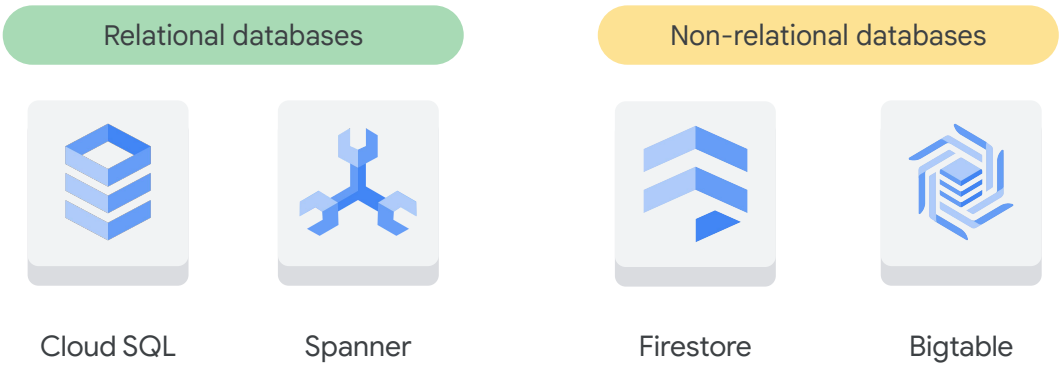
- Doesn't use a tabular format.
- Follows a flexible data model.
- Ideal for data with changing organization.
- Ideal for applications with diverse data types.

A **non-relational database**, sometimes known as a NoSQL database, is less structured in format and doesn't use a tabular format of rows and columns like relational databases.

Instead, non-relational databases follow a flexible data model, which makes them ideal for storing data that changes its organization frequently or for applications that handle diverse types of data. This includes when large quantities of complex and diverse data need to be organized, or when the data regularly evolves to meet new business requirements.

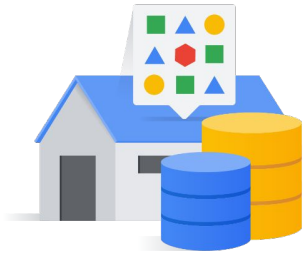
Choosing the right database depends on the use case.

Google Cloud database products



Google Cloud relational database products include Cloud SQL and Spanner, while Firestore and Bigtable are non-relational database products. We'll look at these products in more detail later.

Data warehouse



Data warehouse

An enterprise system used for the analysis and reporting of structured and semi-structured data from multiple sources.

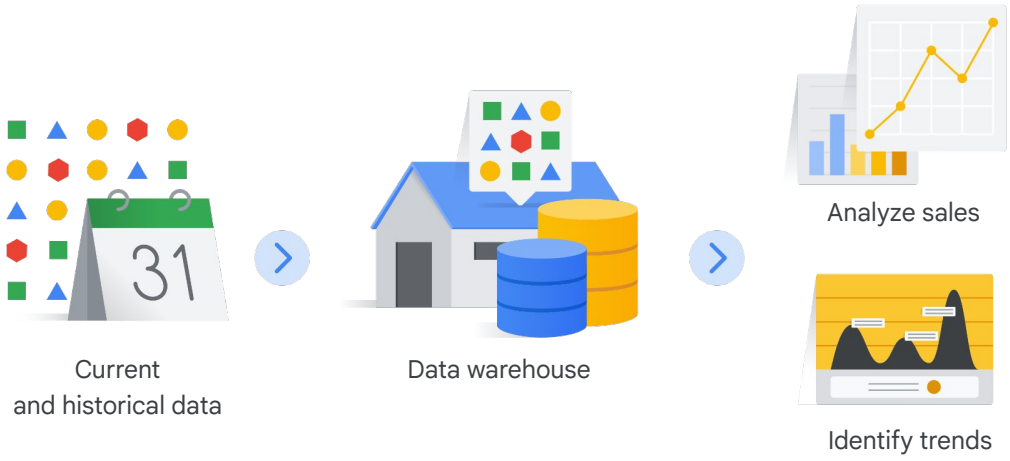
Business data:

- Point of sale
- Marketing automation
- CRM data

Let's explore another data management concept, the **data warehouse**. Like a database, a data warehouse is a place to store data. However, while a database is designed to *capture* data for storage, retrieval, and use, a data warehouse is designed to *analyze* data.

A data warehouse is an enterprise system used for the analysis and reporting of structured and semi-structured data from multiple sources. Think of the data warehouse as the central hub for all business data. Business data might include point-of-sale transactions, marketing automation, or even customer relationship management data.

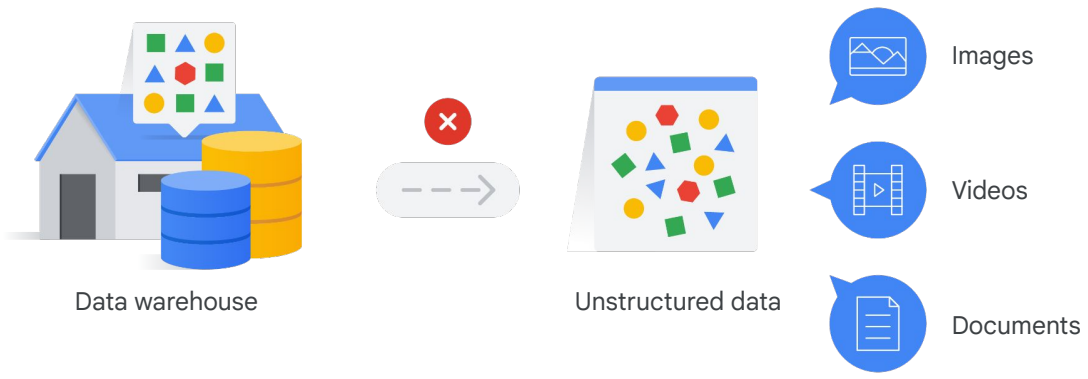
A data warehouse provides a long-range view of data over time



Suited for both ad hoc analysis and custom reporting, a data warehouse can help analyze sales and identify trends, because it can store both current and historical data in one place.

This capability can provide a long-range view of data over time, which makes a data warehouse a primary component of business intelligence.

A data warehouse is not the answer for unstructured data

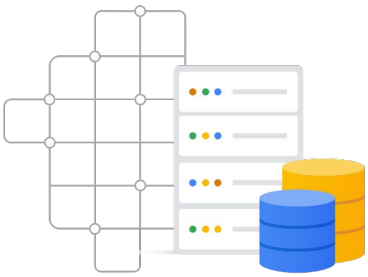


Although data warehouses handle structured and semi-structured data, they're not typically the answer for how to handle large amounts of available unstructured data, like images, videos, and documents.

Unstructured data, which doesn't conform to a well-defined schema, is often disregarded in traditional analytics.

BigQuery: a fully-managed data warehouse

"Fully managed" means that BigQuery takes care of the underlying infrastructure



Terabytes and petabytes of data gathered from a wide range of sources

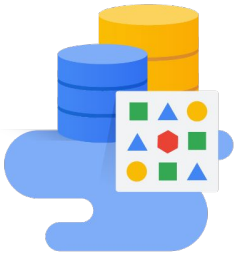
BigQuery is a fully-managed data warehouse.

A data warehouse is a large store, containing **terabytes** and **petabytes** of data gathered from a wide range of sources within an organization, that's used to guide management decisions.

Being fully managed means that BigQuery takes care of the underlying infrastructure, so you can focus on using SQL queries to answer business questions–without worrying about deployment, scalability, and security.

At this point, it's useful to consider what the main difference is between a data warehouse and a data lake. A data lake is just a pool of raw, unorganized, and unclassified data, which has no specified purpose. A data warehouse on the other hand, contains structured and organized data, which can be used for advanced querying.

Data lake



Data lake

A repository designed to ingest, store, explore, process, and analyze any type or volume of raw data

- Operational systems
- Web sources
- Social media
- IoT

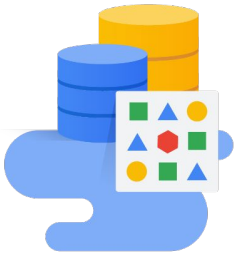
It can store **different types of data**:

- In its original format
- By ignoring size limits
- Without much preprocessing
- Without adding structure

A **data lake** is a repository designed to ingest, store, explore, process, and analyze any type or volume of raw data, regardless of the source, like operational systems, web sources, social media, or Internet of Things, or IoT.

It can store different types of data in its original format; ignoring size limits, and without much pre-processing or adding structure.

Data lake



Data lake

Raw data

- Prevents data contamination.
- Prevents adding bias.
- Can be enriched with other data.

Having this unprocessed, raw data available for analysis prevents unintentionally contaminating the data or adding bias. It also means that the raw data can be enriched by merging it with other data at the same time.

This differs from a data warehouse that contains structured data that has been cleaned and processed, ready for strategic analysis based on predefined business needs.

Data warehouses and data lakes are complementary tools

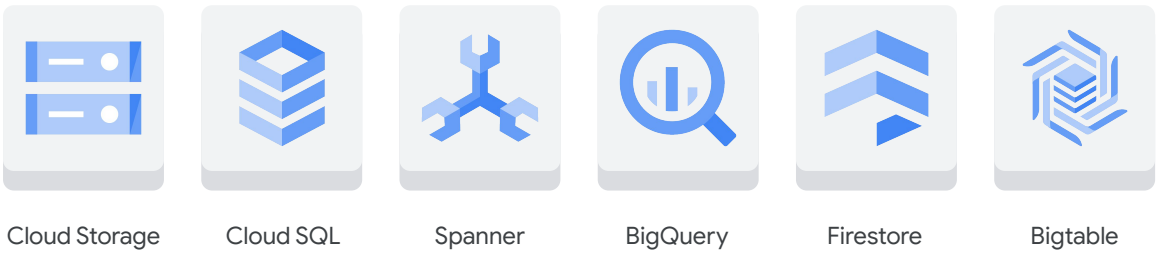
Data warehouses and data lakes should be considered complementary instead of competing tools. Although both store data in some capacity, each is optimized for different uses.

- Traditional data warehouse users are business intelligence analysts who are closer to the business and focus on driving insights from data. These users traditionally use the data to answer questions.
- Data lake users, and also analysts, include data engineers and data scientists. They're closer to the raw data with the tools and capabilities to explore, mine, and experiment with the data. These users find answers in the data, but they also find questions.

As enterprises are increasingly focused on data-driven decision making, data warehouses and data lakes play a critical role in an organization's digital transformation journey. Democratization of data lets users gain a deeper understanding of business situations because they have more context than ever before. Today, organizations need a 360-degree real-time view of their businesses to gain a competitive edge.

Choosing the right storage product

What's the right scenario for each storage option?

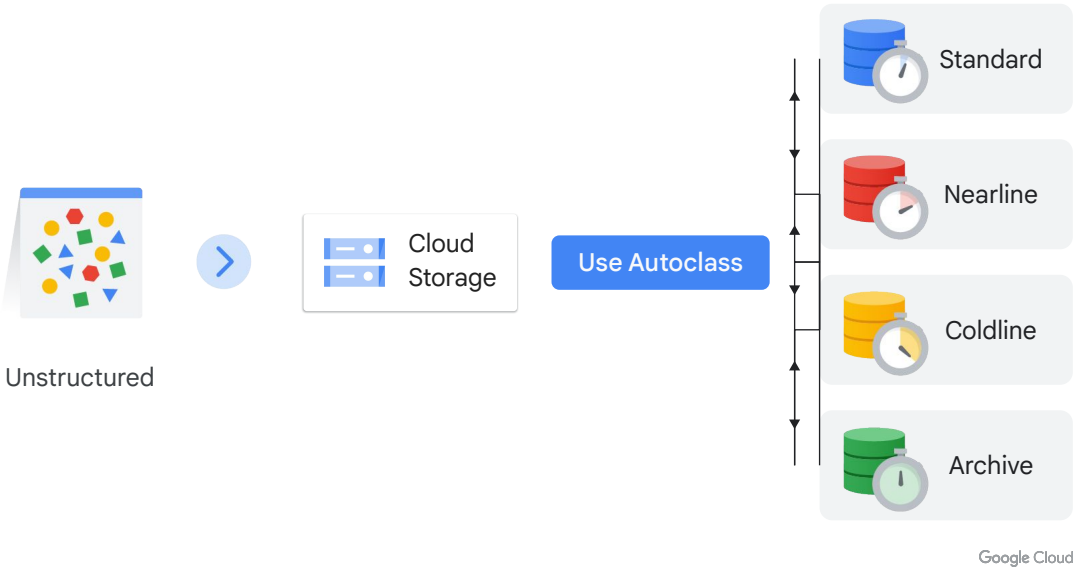


Data type + Business need

So, you've learned about the different storage options that Google Cloud offers, but in what scenarios should you use each one?

Ultimately, it's a combination of the **data type** that needs to be stored and the **business need**.

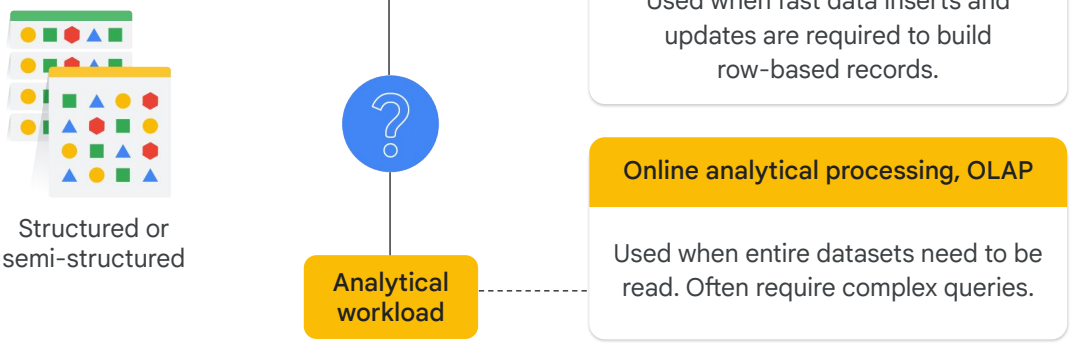
Use Cloud Storage for unstructured data



If data is unstructured, then **Cloud Storage** is the most appropriate option.

You have to decide a storage class: Standard, Nearline, Coldline, or Archive. Or whether to let the Autoclass feature decide that for you.

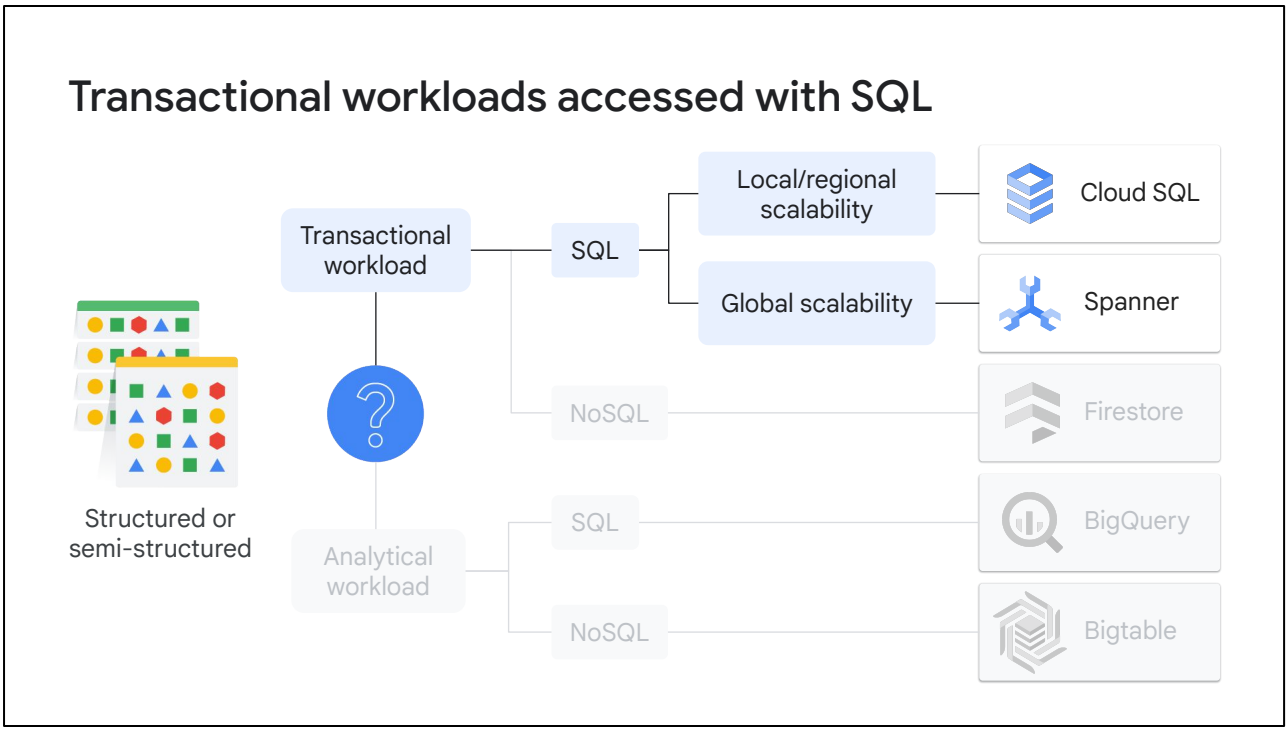
Structured or semi-structured data



If data is structured or semi-structured, choosing a storage product will depend on whether workloads are **transactional** or **analytical**.

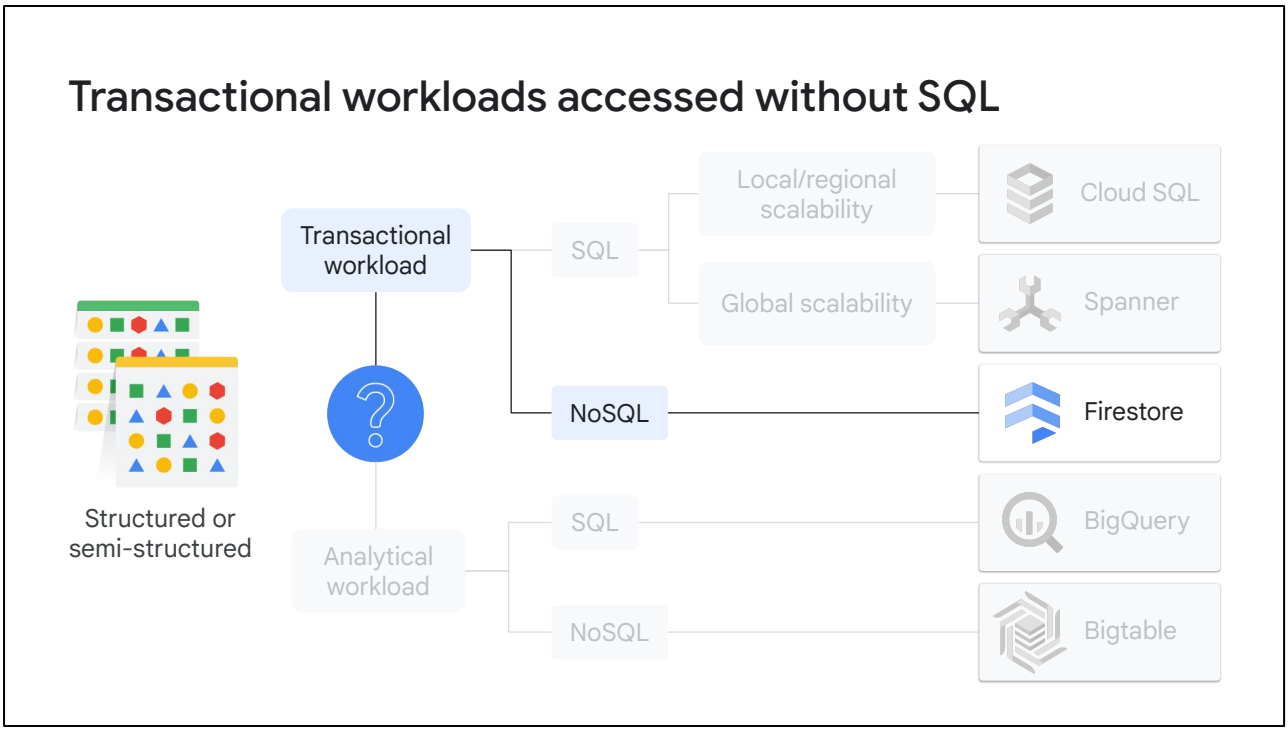
Transactional workloads stem from online transaction processing, or OLTP, systems, which are used when fast data inserts and updates are required to build row-based records. An example of this is point-of-sale transaction records.

Then there are analytical workloads, which stem from online analytical processing, or OLAP systems, which are used when entire datasets need to be read. They often require complex queries, for example, aggregations. An example here would be analyzing sales history to see trends and aggregated views.



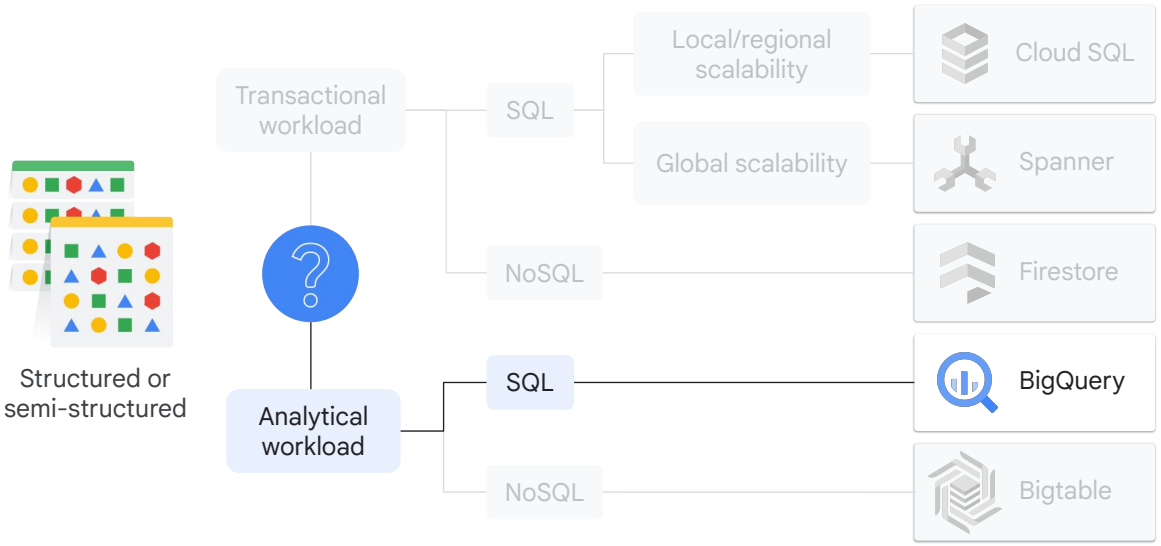
After you determine if the workloads are transactional or analytical, you must determine whether the data will be accessed by using SQL.

So, if your data is transactional and you need to access it by using SQL, then **Cloud SQL** and **Spanner** are two options. Cloud SQL works best for local to regional scalability, and Spanner is best to scale a database globally.



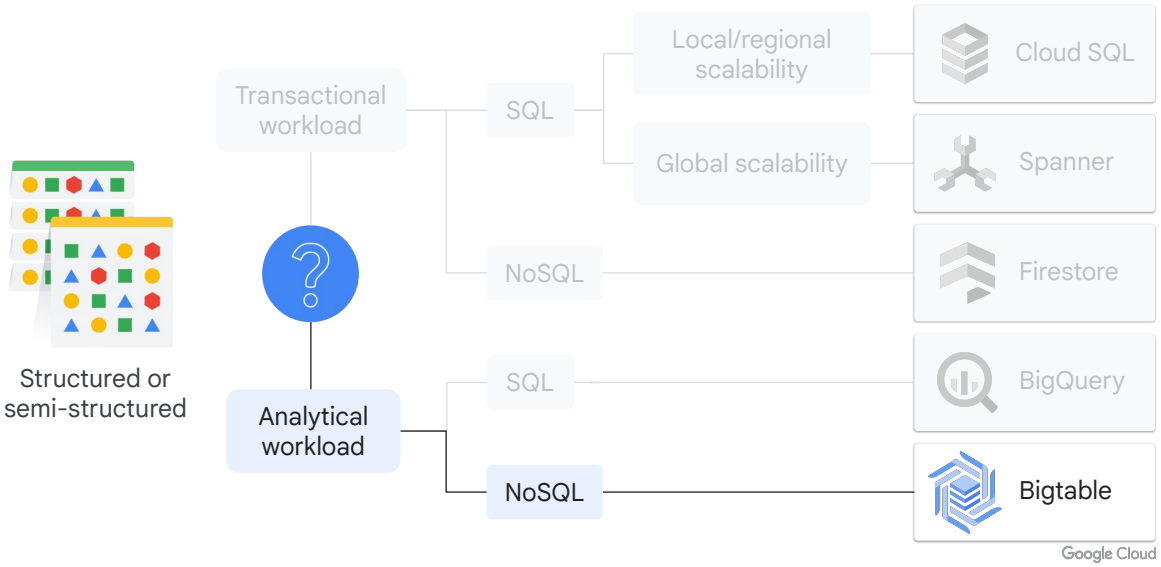
If the transactional data will be accessed without SQL, **Firestore** might be the best option. Firestore is a transactional NoSQL, document-oriented database. (less rigid then a schema; doesn't use SQL query, more programmatic language)

Analytical workloads that require SQL commands



If you have analytical workloads that require SQL commands, **BigQuery** might be the best option. BigQuery, Google's data warehouse solution, lets you analyze petabyte-scale datasets.

Analytical workloads that don't require SQL commands



Alternatively, **Bigtable** provides a scalable NoSQL solution for analytical workloads. It's best for real-time, high-throughput applications that require only **millisecond latency**.

You can use Bigtable to store and query all of the following types of data:

- Time-series data, such as CPU and memory usage over time for multiple servers
- Marketing data, such as purchase histories and customer preferences
- Financial data, such as transaction histories, stock prices, and currency exchange rates
- Internet of Things data, such as usage reports from energy meters and home appliances
- Graph data, such as information about how users are connected to one another