# Types of databases

## Relational database

A relational database is a collection of information that organizes data points with defined relationships for easy access. In the relational database model, the data structures including data tables, indexes and views, remain separate from the physical storage structures, enabling database administrators to edit the physical data storage without affecting the logical data structure. In the enterprise, relational databases are used to organize data and identify relationships between key data points. They make it easy to sort and find information, which helps organizations make business decisions more efficiently and minimize costs. Each row, also called a record or tuple, contains a unique instance of data or *key* for the categories defined by the columns. Each table has a unique primary key that identifies the information in a table. The relationship between tables can be set via the use of foreign keys to a field in a table that links to the primary key of another table.

For example, in a business there is a table like in customer, items, foods, time, phone number, email, gender, address, and so on. so that the user can view the data or information needed for some certain event. When creating a relational database, users define the domain of possible values in a data column and constraints that may apply to that data value.

In addition, relational databases possess physical data independence. This refers to a system's capacity to make changes to the inner schema without altering the external schemas or application programs. Inner schema alterations may include the following:

- the use of new storage devices;
- modifying indexes;
- changing from a specific access method to a different one;
- using different data structures; and
- using various storage structures or file organizations.

## Analytical database

An analytic database, also called an analytical database, is a read-only system that stores historical data on business metrics such as sales performance and inventory levels. Business analysts, corporate executives and other workers run queries and reports against an analytic database. The information is regularly updated to include recent transaction data from an organization's operational systems. Analytical databases store massive amounts of data that organizations use to gain insight into their business, customers and more.

There are five main types of analytic databases:

- Columnar databases

  A columnar database stores data in large contiguous blocks of memory called data columns. This differs from a row-oriented database, where data is stored in tables split across columns and rows. It is the use of columns that makes columnar databases well suited to analytical processing, and able to meet the requirements of data warehouses.

- Massively parallel processing (MPP) databases

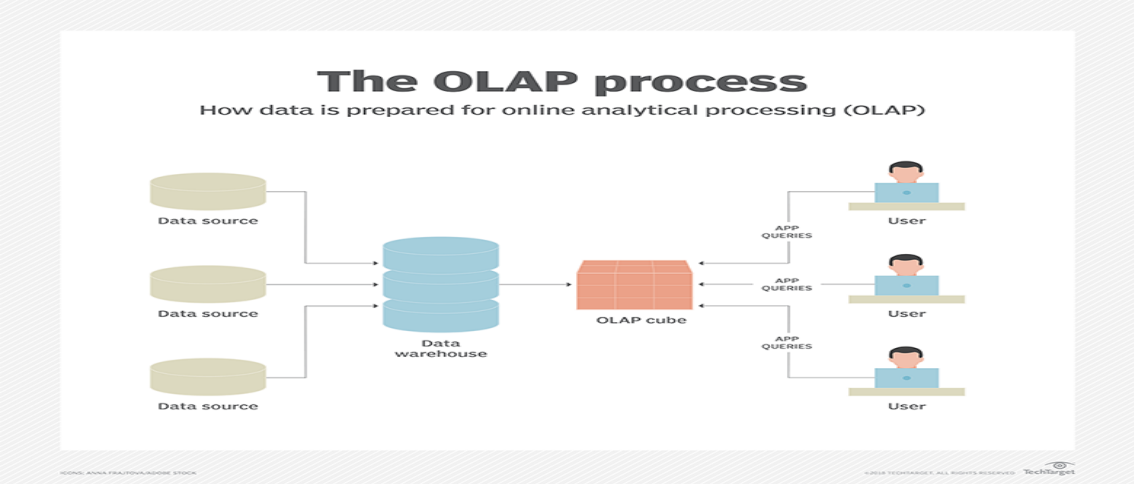
  In an MPP database, data is stored on multiple servers rather than on a single server. Each server has its own set of data and its own computing power. MPP databases typically provide high availability and redundancy by using clusters of computers to serve multiple users.

- In-memory databases.

  In-memory databases are optimized to store data in random-access memory. This contrasts with traditional databases that rely on hard drives to store both data and program instructions. Because data is stored in memory, in-memory databases provide a near-real-time response.

- Online analytical processing (OLAP) databases

An OLAP database is an online database that is used to create reports and analyses that are faster and more efficient than traditional databases. An OLAP database stores multidimensional *cubes* of aggregated data for analyzing information based on multiple data attributes. The data is stored in a structure designed to facilitate analysis of the data




- Data warehouse appliances

Data warehouse appliances are integrated hardware and software tools designed to optimize the performance of data warehouses. A data warehouse is an information repository large organizations use to centralize, store and index vast amounts of data from disparate sources. Data warehouse appliances simplify and expedite data warehouse implementation and management.

## Key-value database

key-value databases do not have a specified structure. Relational databases store data in tables where each column has an assigned data type. Key-value databases are a collection of key-value pairs that are stored as individual records and do not have a predefined data structure. The key can be anything, but seeing that it is the only way of retrieving the value associated with it, naming the keys should be done strategically.

Key names can range from as simple as numbering to specific descriptions of the value that is about to follow. A key-value database can be thought of as a dictionary or a directory. Dictionaries have words as keys and their meanings as values. The choice of which database an organization should apply depends purely on its users and their needs. However, some of the most common use cases of key-value databases are to record sessions in applications that require logins.

This sometimes makes the choice of the right type of database to use obvious. While key-value databases may be limited in what they can do, they are often the right choice for the following reasons:

- Simplicity

  As mentioned above, key value databases are quite simple to use. The straightforward commands and the absence of data types make work easier for programmers. With this feature data can assume any type, or even multiple types, when needed.

- Speed

  This simplicity makes key value databases quick to respond, provided that the rest of the environment around it is well-built and optimized.

- Scalability

  This is a beloved advantage of NoSQL databases over relational databases in general, and key-value stores in particular. Unlike relational databases, which are only scalable vertically, key-value stores are also infinitely scalable horizontally.

- Easy to move

  The absence of a query language means that the database can be easily moved between different systems without having to change the architecture.

- Reliability

  Built-in redundancy comes in handy to cover for a lost storage node where duplicated data comes in place of what's been lost.

Column family

columnar databases store their data by columns, rather than by rows. These columns are gathered to form subgroups. The keys and the column names of this type of database are not fixed. Columns within the same column family, or cluster of columns, can have a different number of rows and can accommodate different types of data and names.

Relational databases have a set schema and they function as tables of rows and columns. Wide-column databases have a similar, but different schema. They also have rows and columns. However, they are not fixed within a table, but have a dynamic schema. Each column is stored separately. If there are similar (related) columns, they are joined into column families and then the column families are stored separately from other column families.

**Column family types**

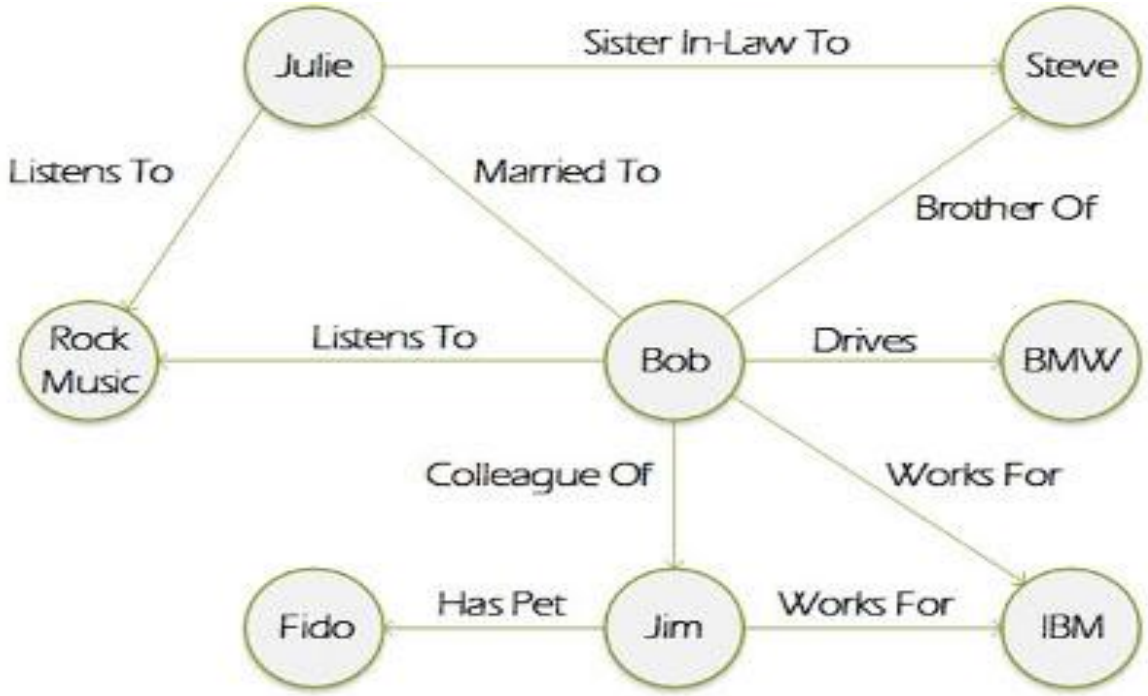
- **Standard column family.** This column family type is similar to a table; it contains a key-value pair where the key is the row key, and the values are stored in columns using their names as their identifiers.

- **Super column family.** A super column represents an array of columns. Each super column has a name and a value mapping the super column out to several different columns. Related super columns are joined under a single row into super column families. Compared to a relational database, this is like a view of several different tables within a database. Imagine you had the view of the columns and values available for a single row -- that is a single identifier across many different tables -- and were able to store them all in one place: That is the super column family.

## Graph

A graph database, also referred to as a semantic database, is a software application designed to store, query and modify network graphs. A network graph is a visual construct that consists of nodes and edges. Each node represents an entity (such as a person) and each edge represents a connection or relationship between two nodes.

graph databases are sometimes referred to as triple stores. That's because this type of database uses a special index that stores information about nodes, edges and the relationship between them in groups of three. graph databases are sometimes referred to as triple stores. That's because this type of database uses a special index that stores information about nodes, edges and the relationship between them in groups of three.



This is the index will look like:

| Row | ?s | ?p | ?o |
|---|---|---|---|
| 1 | :Bob | :marriedTo | :Julie |
| 2 | :Bob | :brotherOf | :Steve |
| 3 | :Bob | :listensTo | :RockMusic |
| 4 | :Julie | :listensTo | :RockMusic |
| 5 | :Julie | :sisterInLawTo | :Steve |
| 6 | :Jim | :worksFor | :IBM |

## Document

Document databases focus on storage and access methods optimized for documents as opposed to rows or records in an RDBMS. The data model is a set of collections of documents that contain key-value collections. In a document store the values can be nested documents or lists, as well as scalar values. The nesting aspect is one important differentiator with the advanced key-value stores we just presented. The

attribute names are not predefined in a global schema, but rather are dynamically defined for each document at runtime.

Document databases store data as structured documents. To some extent, document databases are the further development of key-value stores. On the one hand, document databases employ similar key-value stores and permit nested key-value pairs. These databases enable higher query performance than key-value stores. On the other hand, document databases are regarded as an improvement to scheme-less key-value stores by adopting a self-described document format.