**Relational Database Management System**

**Relational Database**

A relational database is a collection of data organized into a table structure, where the tables can be linked, or related, based on data common to each. Tables are made of rows and columns, where rows are the “records”, and the columns the “attributes”.

Example:

Let’s take the example of a customer table that maintains data about each customer in a company. The columns, or attributes, in the customer table are the Company ID, Company Name, Company Address, and Company Primary Phone; and Each row is a customer record. Now let’s understand what we mean by tables being linked, or related, based on data common to each. Along with the customer table, the company also maintains transaction tables that contain data describing multiple individual transactions pertaining to each customer. The columns for the transaction table might include the Transaction Date, Customer ID, Transaction Amount, and Payment Method. The customer table and the transaction tables can be related based on the common Customer ID field. You can query the customer table to produce reports such as a customer statement that consolidates all transactions in a given period.

This capability of relating tables based on common data enables you to retrieve an entirely

new table from data in one or more tables with a single query. It also allows you to understand the relationships among all available data and gain new insights for making better decisions.

Relational databases use structured query language, or SQL, for querying data. Relational databases build on the organizational principles of flat files such as spreadsheets, with data organized into rows and columns following a well-defined structure and schema.

Relational databases, by design, are ideal for the optimized storage, retrieval, and processing of data for large volumes of data, unlike spreadsheets that have a limited number of rows and columns. Each table in a relational database has a unique set of rows and columns and relationships can be defined between tables, which minimizes data redundancy. Moreover, you can restrict database fields to specific data types and values, which minimizes irregularities and leads to greater consistency and data integrity. Relational databases use SQL for querying data, which gives you the advantage of processing millions of records and retrieving large amounts of data in a matter of seconds. Moreover, the security architecture of relational databases provides controlled access to data and ensures that the standards and policies for governing data can be enforced.

Relational databases range from small desktop systems to massive cloud-based systems.

They can be either:

* open-source and internally supported,
* open-source with commercial support, or
* commercial closed-source systems.

IBM DB2, Microsoft SQL Server, MySQL, Oracle Database, and PostgreSQL are some of the popular relational databases.

Cloud-based relational databases, also referred to as Database-as-a-Service, are gaining wide

use as they have access to the limitless compute and storage capabilities offered by the cloud.

Some of the popular cloud relational databases include Amazon Relational Database Service

(RDS), Google Cloud SQL, IBM DB2 on Cloud, Oracle Cloud, and SQL Azure.

Relational Database Management System

RDBMS is a mature and well-documented technology, making it easy to learn and find qualified

talent. One of the most significant advantages of the relational database approach is its ability

to create meaningful information by joining tables. Some of its other advantages include:

* Flexibility: Using SQL, you can add new columns, add new tables, rename relations, and make other changes while the database is running and queries are happening.
* Reduced redundancy: Relational databases minimize data redundancy.

For example, the information of a customer appears in a single entry in the customer table, and the transaction table pertaining to the customer stores a link to the customer table.

* Ease of backup and disaster recovery: Relational databases offer easy export and import options, making backup and restore easy. Exports can happen while the database is running, making restore on failure easy. Cloud-based relational databases do continuous mirroring, which means the loss of data on restore can be measured in seconds or less.
* ACID-compliance: ACID stands for Atomicity, Consistency, Isolation, and Durability. And ACID compliance implies that the data in the database remains accurate and consistent despite failures, and database transactions are processed reliably.

Now we’ll look at some use cases for relational databases:

1. Online Transaction Processing:

OLTP applications are focused on transaction-oriented tasks that run at high rates. Relational databases are well suited for OLTP applications because they can accommodate a large number of users; they support the ability to insert, update, or delete small amounts of data; and they also support frequent queries and updates as well as fast response times.

1. Data warehouses:

In a data warehousing environment, relational databases can be optimized for online analytical

processing (or OLAP), where historical data is analysed for business intelligence.

1. IoT solutions:

Internet of Things (IoT) solutions require speed as well as the ability to collect and process data from edge devices, which need a lightweight database solution.

This brings us to the limitations of RDBMS:

* RDBMS does not work well with semi-structured and unstructured data and is, therefore, not suitable for extensive analytics on such data.
* For migration between two RDBMSs, schemas and type of data need to be identical between the source and destination tables.
* Relational databases have a limit on the length of data fields, which means if you try to enter more information into a field than it can accommodate, the information will not be stored.
* Despite the limitations and the evolution of data in these times of big data, cloud computing, IoT devices, and social media, RDBMS continues to be the predominant technology for working with structured data.