SNS COLLEGE OF ENGINEERING

Kurumbapalayam (Po), Coimbatore – 641 107

An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING-IOT Including CS&BCT

COURSE NAME : 19SB504 DATABASE MANAGEMENT SYSTEMS

III YEAR / V SEMESTER

Unit V- CONCURRENCY CONTROL AND RECOVERY SYSTEM

Topic : **STORAGE STRUCTURE, RECOVERY AND ATOMICITY**

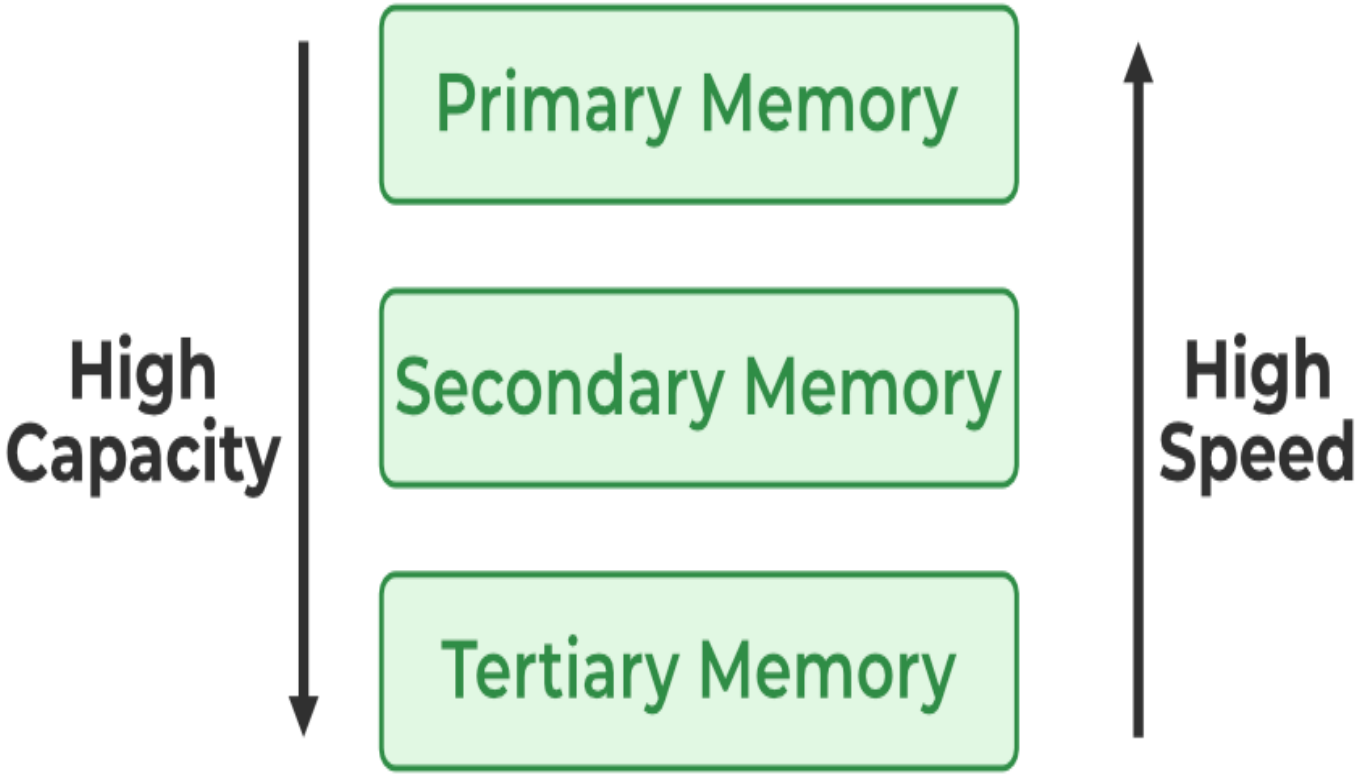
STORAGE STRUCTURE IN DBMS

Objective:

The **records** in databases are stored in **file formats**. Physically, the data is stored in electromagnetic format on a device.

The electromagnetic devices used in database systems for data storage are classified as follows:

- ✓ Primary Memory
- ✓ Secondary Memory
- ✓ Tertiary Memory


Primary Memory
Secondary Memory
Tertiary Memory
High Capacity
High Speed

Primary Memory

The primary memory of a server is the type of **data storage that is directly accessible by the central processing unit**, meaning that it doesn't require any other devices to read from it.

Cache Memory

- The size of these devices is considerably **smaller and they are volatile.**

- These primary memory devices are **usually more expensive due to their increased speed and performance.**

Secondary Memory

Secondary storage is also called as Online storage.

It is the storage area that allows the user to save and store data permanently.

This type of memory does not lose the data due to any power failure or system crash.

That's why we also call it non-volatile storage.

Flash Memory:

A flash memory stores data in USB (Universal Serial Bus) keys which are further plugged into the USB slots of a computer system.

Magnetic Disk Storage:

This type of storage media is also known as online storage media.

A magnetic disk is used for storing the data for a long time.

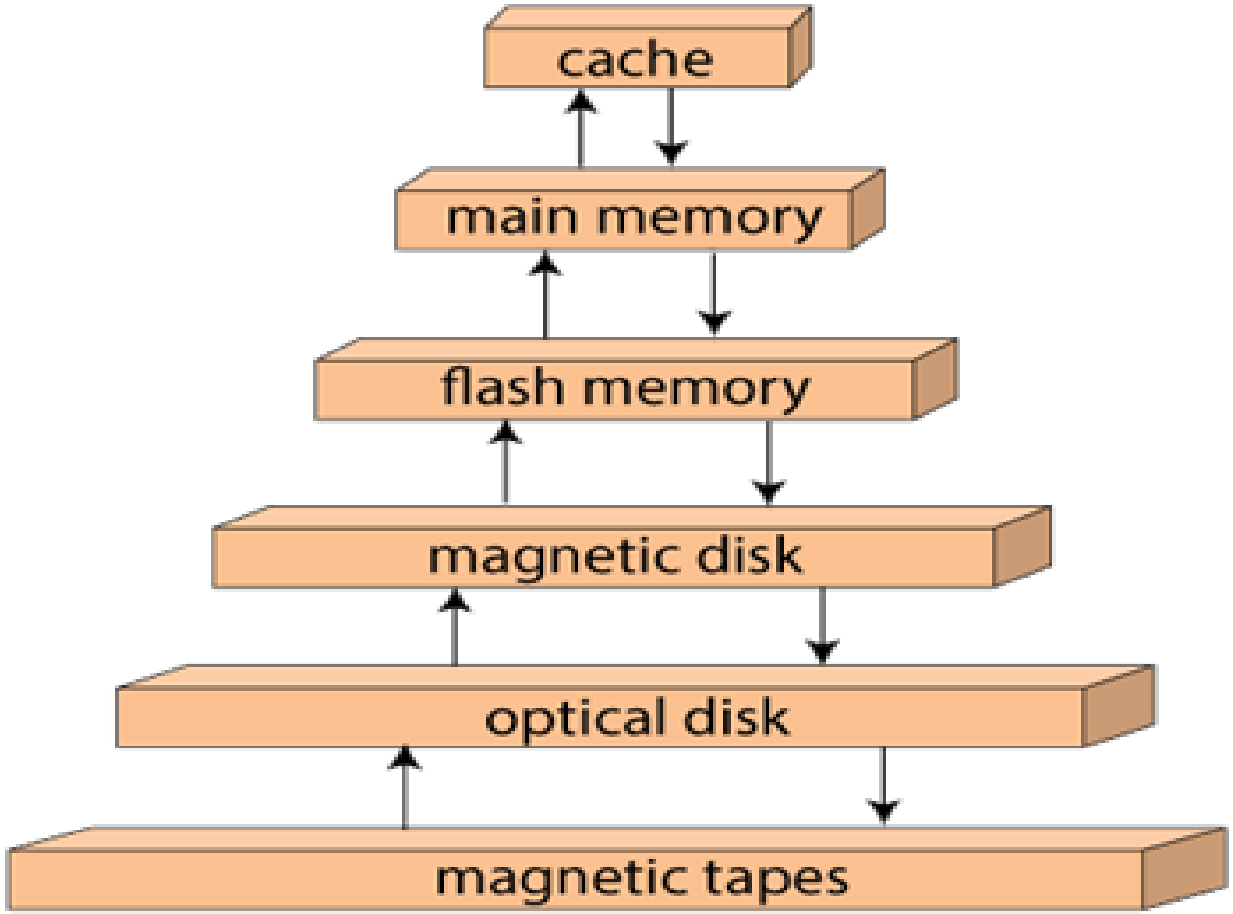
Tertiary Storage

It is the storage type that is **external** from the computer system. It has the **slowest speed**.

But it is capable of storing a large amount of data. It is also known as Offline storage.

Tertiary storage is generally used for data backup. There are following tertiary storage devices available:

Optical Storage: An optical storage can store **megabytes or gigabytes** of data.
Tape Storage: It is the **cheapest storage** medium than disks. Generally, tapes are used for archiving or backing up the data.



Storage device hierarchy

Recovery and Atomicity in DBMS

A database possesses atomicity, consistency, isolation, and durability qualities.

The ability of a system to preserve data and changes made to data defines its durability.

A database could fail for any of the following reasons:

- ✓ System breakdowns occur as a result of hardware or software issues in the system.
- ✓ Transaction failures arise when a certain process dealing with data updates cannot be completed.
- ✓ Disk crashes may occur as a result of the system's failure to read the disc.

Physical damages include issues such as power outages or natural disasters.

The data in the database must be recoverable to the state they were in prior to the system failure, even if the database system fails.

In such situations, database recovery procedures in DBMS are employed to retrieve the data.

The recovery procedures in DBMS ensure the database's atomicity and durability.

If a system crashes in the middle of a transaction and all of its data is lost, it is not regarded as durable.

Data recovery procedures in DBMS make sure that the data is always recoverable to protect the durability property and that its state is retained to protect the atomic property.

The procedures listed below are used to recover data from a DBMS,

- ✓ Recovery based on logs.
- ✓ Recovery through Deferred Update
- ✓ Immediate Recovery via Immediate Update

Log-Based Recovery

Every DBMS has its own system logs, which record every system activity and include timestamps for the event's timing.

Databases manage several log files for operations such as errors, queries, and other database updates.

The log is saved in the following file formats:

- ✓ The structure [start transaction, T] represents the start of transaction T execution.
- ✓ [write the item, T, X, old value, new value] indicates that the transaction T changes the value of the variable X from the old value to the new value.
- ✓ [read item, T, X] indicates that the transaction T reads the value of X.
- ✓ [commit, T] signifies that the modifications to the data have been committed to the database and cannot be updated further by the transaction. There will be no errors after the database has been committed.
- ✓ [abort, T] indicates that the transaction, T, has been cancelled.

Shadow Paging

In shadow paging, a database is divided into n- multiple pages, each of which represents a fixed-size disc memory.

Similarly, shadow pages, which are replicas of the original database, are created.

The database state is copied to the shadow pages at the start of a transaction.

Only the original database will be changed during the transaction, not the shadow pages.

The updates to the shadow pages are made when the transaction reaches the commit step.

The modifications are done so that if the i-th section of the hard disc is changed, the i-th shadow page is also changed.

In the event that the system fails, recovery procedures are carried out after comparing the database's true pages to its shadow pages.

In the logical memory of the Caching/Buffering method is a collection of buffers known as DBMS buffers.

Throughout the process, all logs are kept in buffers, and the main log file is updated once the transaction reaches the commit stage.

Atomicity:

A set of **ideas used to ensure the integrity** of database transactions is known as the ACID model, which stands for Atomicity, Consistency, Isolation, and Durability in database management systems.

Atomicity is **achieved mostly by complex processes** such as journaling or logging or through operating-system calls.

In a database management system, an atomic transaction is defined as an indivisible and irreducible series of database actions in which either everything or nothing happens.

Atomicity Examples:

Atomicity in Online Ticket Booking Systems:

Using an online ticket booking system as an example, a booking may consist of two separate acts that combine to form a transaction:
The first is the payment for the ticket, and the second is to reserve the seat for the person who just paid. According to business logic, these two distinct and separate actions must occur concurrently.

If one develops without the other, problems may arise. The system might reserve the same seat for two separate consumers, for instance.

Thank you .....