3.1.1 FILE ALLOCATION TABLE (FAT)

Definition:

The **File Allocation Table (FAT)** is a data structure used by the operating system to keep track of file locations on a disk.

- It maintains **a list of all clusters** on the disk and records:
  - Which clusters are **free**
  - Which clusters are **allocated to files**
  - The **sequence** of clusters for each file

The File Allocation Table (FAT) is a critical component of computer systems that allows for the organization and management of files on a disk. It helps to keep track of where files are stored on a disk and how much space is available for new files using a table that maps file names to their physical location on the disk.

Structure of FAT

It has a simple and straightforward structure. It consists of a sequence of entries, with each entry representing a cluster on the disk. A cluster is a group of contiguous sectors, which is the smallest unit of disk space that can be allocated to a file. Each entry in the FAT contains information about the status of the corresponding cluster, such as whether it is free or allocated to a file. The entries also contain pointers to the next cluster in a file, allowing the FAT to keep track of the sequence of clusters that make up a file. The first entry in the FAT is reserved for the root directory of the disk, while the remaining entries are used for file and directory clusters. The size and format of the FAT can vary depending on the version of the file system and the size of the disk. For example,

Older versions of FAT: FAT12 and FAT16 have smaller maximum disk sizes.

Newer versions of FAT: FAT32 can support larger disks and use longer entry sizes to accommodate more clusters.

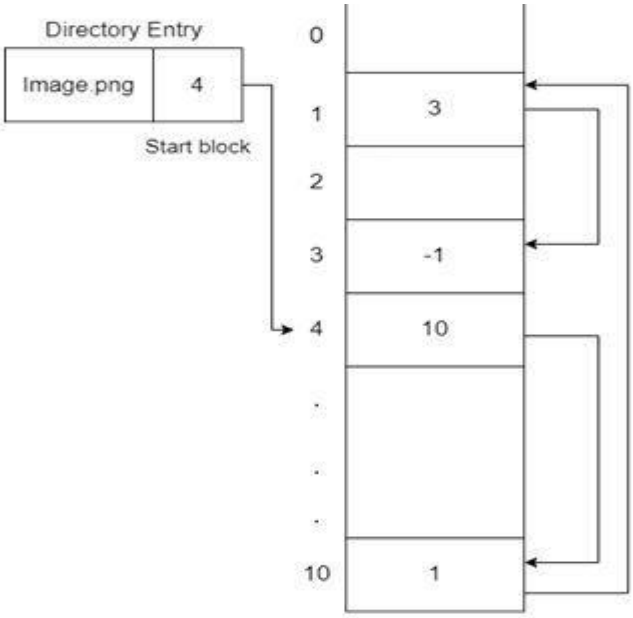

Diagram-1

1. **Directory Entry**
    - Holds file information like **name**, **attributes**, **size**, and importantly, the **start block number** where the file's first cluster is stored.

      File: Image.png

      Start Block: 4

2. **File Allocation Table**
    - An array where **each index represents a block/cluster** on the disk.
    - Each entry stores:
        - **Next block number** → pointer to the next block of the file
        - **-1** → end of file (EOF marker)
        - **0** → free (unused) block

Working:

- **Step 1:**

  The file Image.png starts at **block 4** (from directory entry).
- **Step 2:**

  Look at **FAT[4]** → value is 10.

  This means: after block 4, go to block 10.

- **Step 3:**
  Look at **FAT[10]** → value is 1.
  After block 10, go to block 1.
- **Step 4:**
  Look at **FAT[1]** → value is 3.
  After block 1, go to block 3.
- **Step 5:**
  Look at **FAT[3]** → value is -1.
  This means block 3 is the **last block** of the file.

From the above diagram,

- **Data is not stored in sequential blocks** which is an example of **non-contiguous allocation**.
- FAT makes it possible to store files even if blocks are scattered around the disk.
- The OS only needs the **start block** from the directory entry; the rest is found by following the FAT chain.

Types of File Allocation Table(FAT)

1. **FAT12**
   - It was the original version of the FAT file system.
   - Uses **12-bit** entries for each cluster number.
   - It was designed for small disks, with a maximum size of 16MB and a cluster size of 512 bytes.
   - It is no longer commonly used.
   - Example: Old floppy disks, digital cameras and music players.

2.FAT16

- Uses **16-bit** entries for each cluster.
- Next version of the FAT file system.
- It supports larger disks than FAT12, with a maximum size of 2GB and a cluster size of up to 64KB.
- It is still used on some devices, but it is not as common as it used to be.
- **Example**: USB drives under 2 GB.

3. FAT32

- Uses **28 bits** of a 32-bit entry for cluster numbers.
- most recent version of the FAT file system.
- It was designed to support larger disks than FAT16, with a maximum size of 2TB and a cluster size of up to 32KB.
- FAT32 is still widely used today.
- Example: USB drives and SD cards.

4. exFAT (Extended FAT)

- "Extended File Allocation Table" — introduced by Microsoft.
- **Cluster size** is 4 KB to 32 MB.
- **Max file size**: 16 EB (Exabyte's).
- **Usage**: Flash drives, SDXC cards, Embedded system, USB drives.

Advantages:

- **Simple** structure, easy to implement
- Supported on **almost all OS platforms**
- Good for small disks and removable media
- Allows multiple copies of the FAT for redundancy

Disadvantages:

- **Poor performance** on large disks (large table size)
- **Fragmentation** occurs easily, slowing down file access
- Inefficient space usage if cluster size is large
- No built-in security or journaling