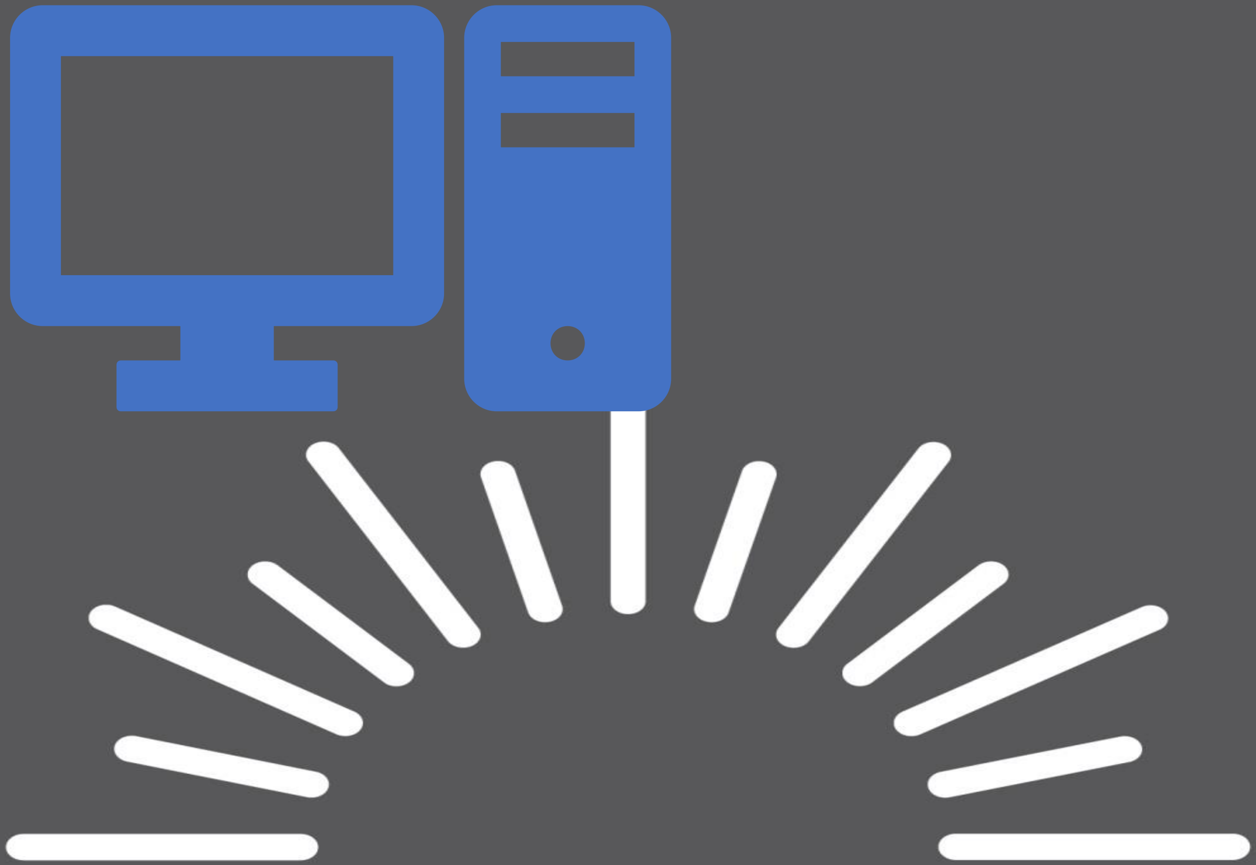
Operating System

(Chapter-5)



Syllabus Content

- 5.1 Overview, File Organization and Access, File Directories, File Sharing

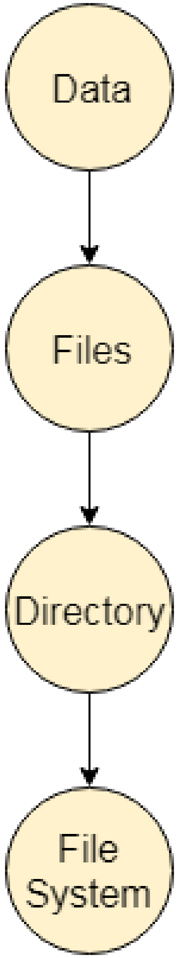
File Management

What is a File ?

- A file can be defined as a data structure which stores the sequence of records.
- Files are stored in a file system, which may exist on a disk or in the main memory.
- Files can be simple (plain text) or complex (specially-formatted).
- The collection of files is known as Directory. The collection of directories at the different levels, is known as File System.

File Management

What is a File ?



File Management

Attributes of the File

1.Name :Every file carries a name by which the file is recognized in the file system. One directory cannot have two files with the same name.

2.Identifier:Along with the name, Each File has its own extension which identifies the type of the file. For example, a text file has the extension .txt, A video file can have the extension .mp4.

3.Type :In a File System, the Files are classified in different types such as video files, audio files, text files, executable files, etc.

4.Location:In the File System, there are several locations on which, the files can be stored. Each file carries its location as its attribute.

5.Size:The Size of the File is one of its most important attribute. By size of the file, we mean the number of bytes acquired by the file in the memory.

6.Protection:The Admin of the computer may want the different protections for the different files. Therefore each file carries its own set of permissions to the different group of Users.

7.Time and Date:Every file carries a time stamp which contains the time and date on which the file is last modified.

File Management

Operations on the File

- A file is a collection of logically related data that is recorded on the secondary storage in the form of sequence of operations.
- The content of the files are defined by its creator who is creating the file.
- The various operations which can be implemented on a file such as read, write, open and close etc. are called file operations.
- These operations are performed by the user by using the commands provided by the operating system.

File Management

Operations on the File

1.Create operation:

This operation is used to create a file in the file system. It is the most widely used operation performed on the file system. To create a new file of a particular type the associated application program calls the file system. This file system allocates space to the file. As the file system knows the format of directory structure, so entry of this new file is made into the appropriate directory.

2. Open operation:

- This operation is the common operation performed on the file. Once the file is created, it must be opened before performing the file processing operations. When the user wants to open a file, it provides a file name to open the particular file in the file system. It tells the operating system to invoke the open system call and passes the file name to the file system.

3. Write operation:

This operation is used to write the information into a file. A system call write is issued that specifies the name of the file and the length of the data has to be written to the file. Whenever the file length is increased by specified value and the file pointer is repositioned after the last byte written.

File Management

Operations on the File

4. Read operation:

This operation reads the contents from a file. A Read pointer is maintained by the OS, pointing to the position up to which the data has been read.

5. Re-position or Seek operation:

The seek system call re-positions the file pointers from the current position to a specific place in the file i.e. forward or backward depending upon the user's requirement. This operation is generally performed with those file management systems that support direct access files.

6. Delete operation:

Deleting the file will not only delete all the data stored inside the file it is also used so that disk space occupied by it is freed. In order to delete the specified file the directory is searched. When the directory entry is located, all the associated file space and the directory entry is released.

File Management

Operations on the File

7. Truncate operation:

Truncating is simply deleting the file except deleting attributes. The file is not completely deleted although the information stored inside the file gets replaced.

8. Close operation:

When the processing of the file is complete, it should be closed so that all the changes made permanent and all the resources occupied should be released. On closing it deallocates all the internal descriptors that were created when the file was opened.

9. Append operation:

This operation adds data to the end of the file.

10. Rename operation:

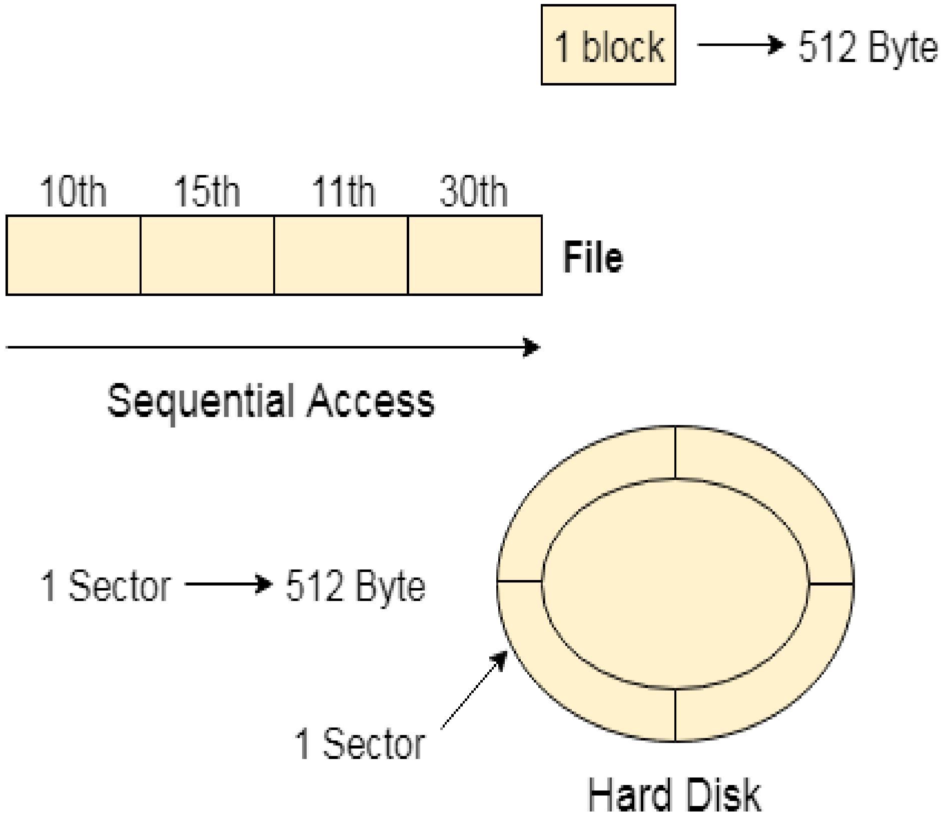
This operation is used to rename the existing file.

File Management

File Access Methods

Sequential Access:

- In sequential access, the OS read the file word by word.
- A pointer is maintained which initially points to the base address of the file.
- If the user wants to read first word of the file then the pointer provides that word to the user and increases its value by 1 word. This process continues till the end of the file.
- Modern word systems do provide the concept of direct access and indexed access but the most used method is sequential access due to the fact that most of the files such as text files, audio files, video files, etc need to be sequentially accessed.



File Management

File Access Methods

Direct Access

- The Direct Access is mostly required in the case of database systems. In most of the cases, we need filtered information from the database. The sequential access can be very slow and inefficient in such cases.

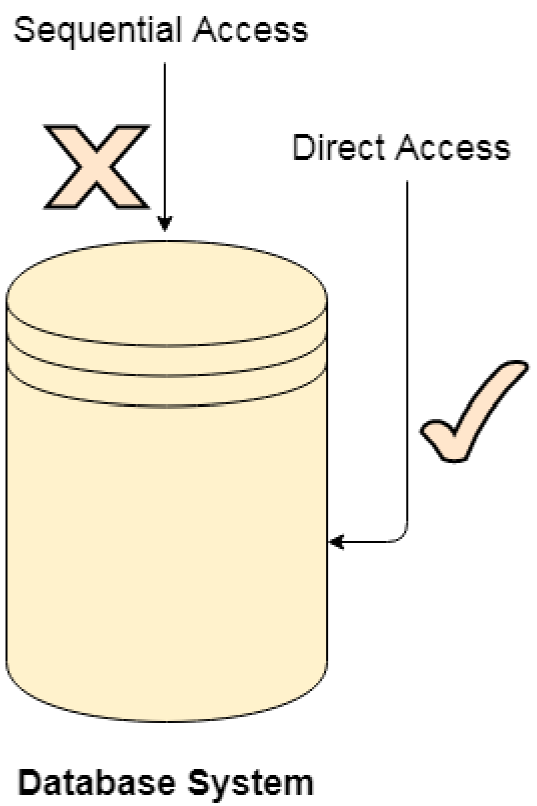
Suppose every block of the storage stores 4 records and we know that the record we needed is stored in 10th block. In that case, the sequential access will not be implemented because it will traverse all the blocks in order to access the needed record.

Direct access will give the required result despite of the fact that the operating system has to perform some complex tasks such as determining the desired block number. However, that is generally implemented in database applications.

File Management

File Access Methods

Direct Access



File Access Methods

Indexed Access

- If a file can be sorted on any of the filed then an index can be assigned to a group of certain records. However, A particular record can be accessed by its index. The index is nothing but the address of a record in the file.
- In index accessing, searching in a large database became very quick and easy but we need to have some extra space in the memory to store the index value.
-

File Management

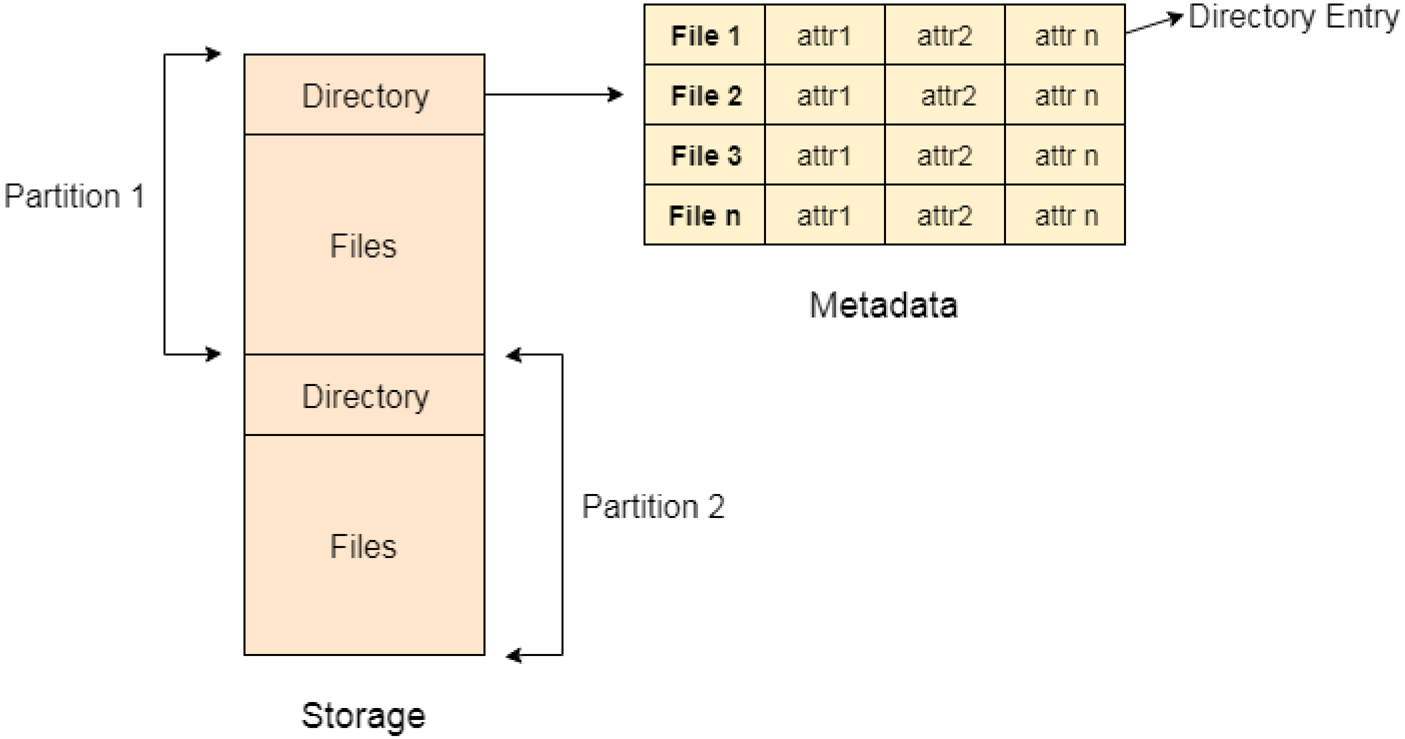
Directory Structure

What is a directory?

- Directory can be defined as the listing of the related files on the disk. The directory may store some or the entire file attributes.
- To get the benefit of different file systems on the different operating systems, A hard disk can be divided into the number of partitions of different sizes. The partitions are also called volumes or mini disks.
- Each partition must have at least one directory in which, all the files of the partition can be listed. A directory entry is maintained for each file in the directory which stores all the information related to that file.

File Management

Directory Structure



File 1	attr1	attr2	attr n
File 2	attr1	attr2	attr n
File 3	attr1	attr2	attr n
File n	attr1	attr2	attr n

File Management

Directory Structure

What is a directory?

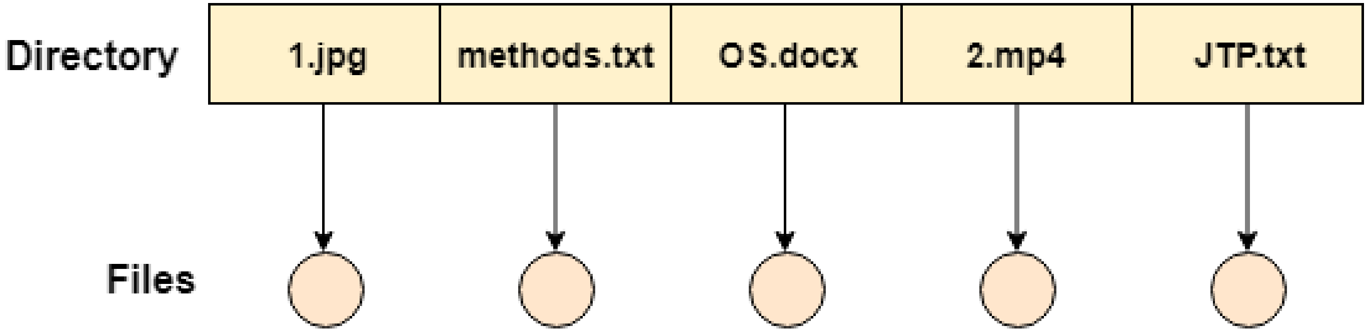
Every Directory supports a number of common operations on the file:

- File Creation
- Search for the file
- File deletion
- Renaming the file
- Traversing Files
- Listing of files

File Management

Single Level Directory

The simplest method is to have one big list of all the files on the disk. The entire system will contain only one directory which is supposed to mention all the files present in the file system. The directory contains one entry per each file present on the file system.



Single Level Directory

File Management

Single Level Directory

Advantages

- Implementation is very simple.
- If the sizes of the files are very small then the searching becomes faster.
- File creation, searching, deletion is very simple since we have only one directory.

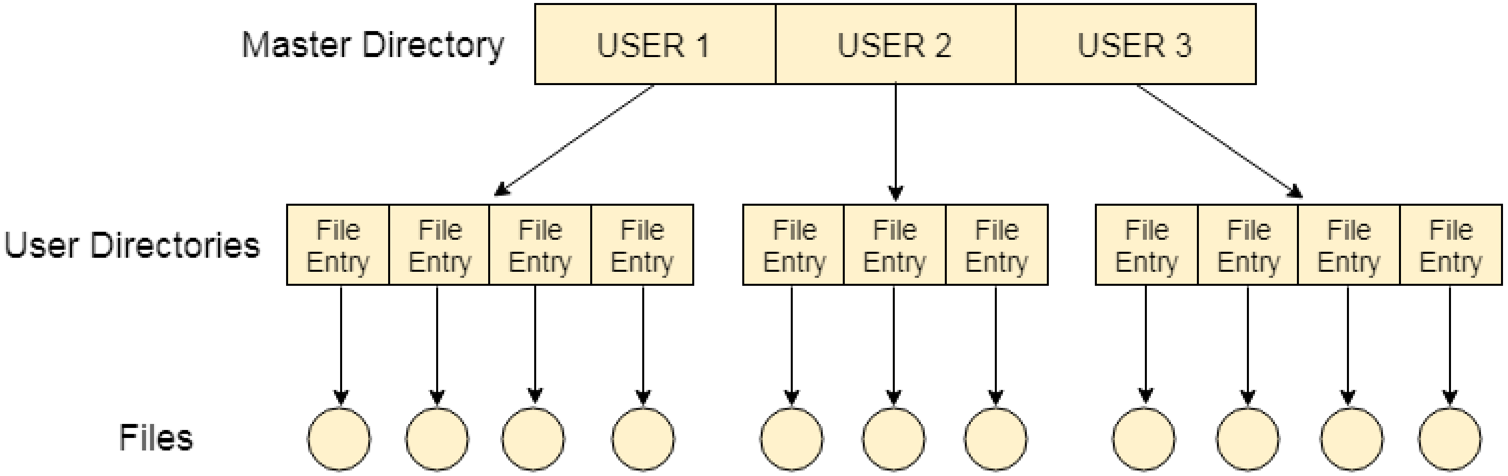
Disadvantages

- We cannot have two files with the same name.
- The directory may be very big therefore searching for a file may take so much time.
- Protection cannot be implemented for multiple users.
- There are no ways to group same kind of files.
- Choosing the unique name for every file is a bit complex and limits the number of files in the system because most of the Operating System limits the number of characters used to construct the file name.

File Management

Two Level Directory

In two level directory systems, we can create a separate directory for each user. There is one master directory which contains separate directories dedicated to each user. For each user, there is a different directory present at the second level, containing group of user's file. The system doesn't let a user to enter in the other user's directory without permission.



Two Level Directory

File Management

Two Level Directory

Characteristics of two level directory system

- Each files has a path name as /User-name/directory-name/
- Different users can have the same file name.
- Searching becomes more efficient as only one user's list needs to be traversed.
- The same kind of files cannot be grouped into a single directory for a particular user.
- Every Operating System maintains a variable as PWD which contains the present directory name (present user name) so that the searching can be done appropriately.

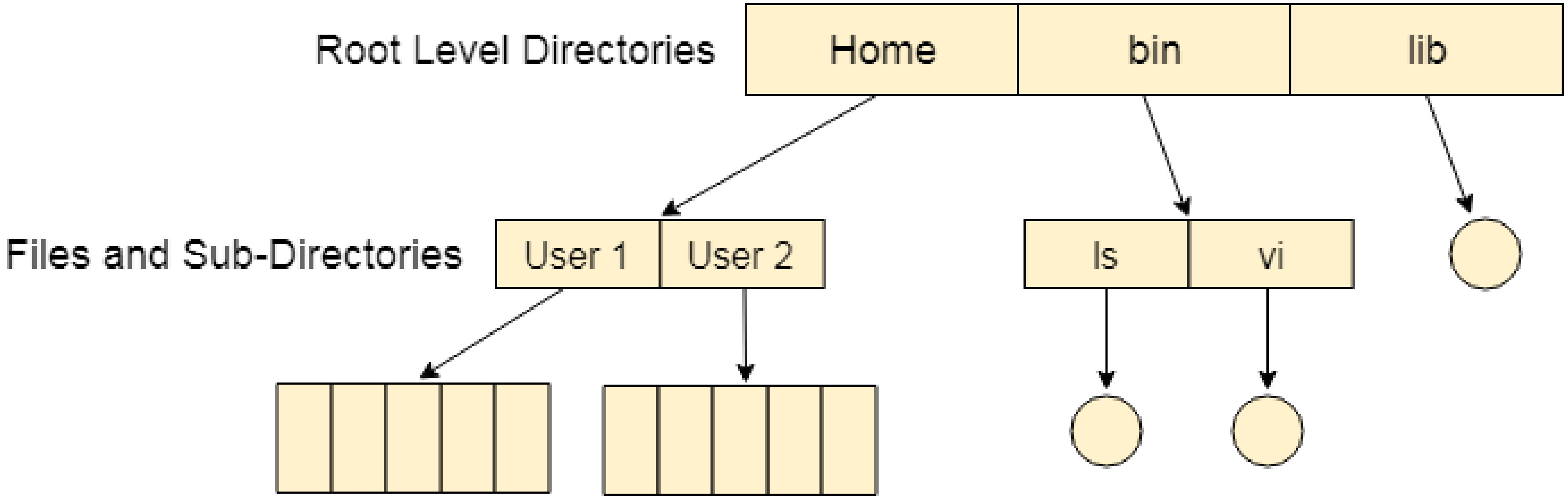
File Management

Tree Structured Directory

- In Tree structured directory system, any directory entry can either be a file or sub directory. Tree structured directory system overcomes the drawbacks of two level directory system. The similar kind of files can now be grouped in one directory.
- Each user has its own directory and it cannot enter in the other user's directory. However, the user has the permission to read the root's data but he cannot write or modify this. Only administrator of the system has the complete access of root directory.
- Searching is more efficient in this directory structure. The concept of current working directory is used. A file can be accessed by two types of path, either relative or absolute.
- Absolute path is the path of the file with respect to the root directory of the system while relative path is the path with respect to the current working directory of the system. In tree structured directory systems, the user is given the privilege to create the files as well as directories.

File Management

Tree Structured Directory



The Structured Directory System

File Management

Tree Structured Directory

Permissions on the file and directory

- A tree structured directory system may consist of various levels therefore there is a set of permissions assigned to each file and directory.
- The permissions are R W X which are regarding reading, writing and the execution of the files or directory. The permissions are assigned to three types of users: owner, group and others.
- There is a identification bit which differentiate between directory and file. For a directory, it is d and for a file, it is dot (.)

File Management

Acyclic-Graph Structured Directories

- The tree structured directory system doesn't allow the same file to exist in multiple directories therefore sharing is major concern in tree structured directory system.
- We can provide sharing by making the directory an acyclic graph. In this system, two or more directory entry can point to the same file or sub directory.
- That file or sub directory is shared between the two directory entries.
- These kinds of directory graphs can be made using links or aliases. We can have multiple paths for a same file. Links can either be symbolic (logical) or hard link (physical).

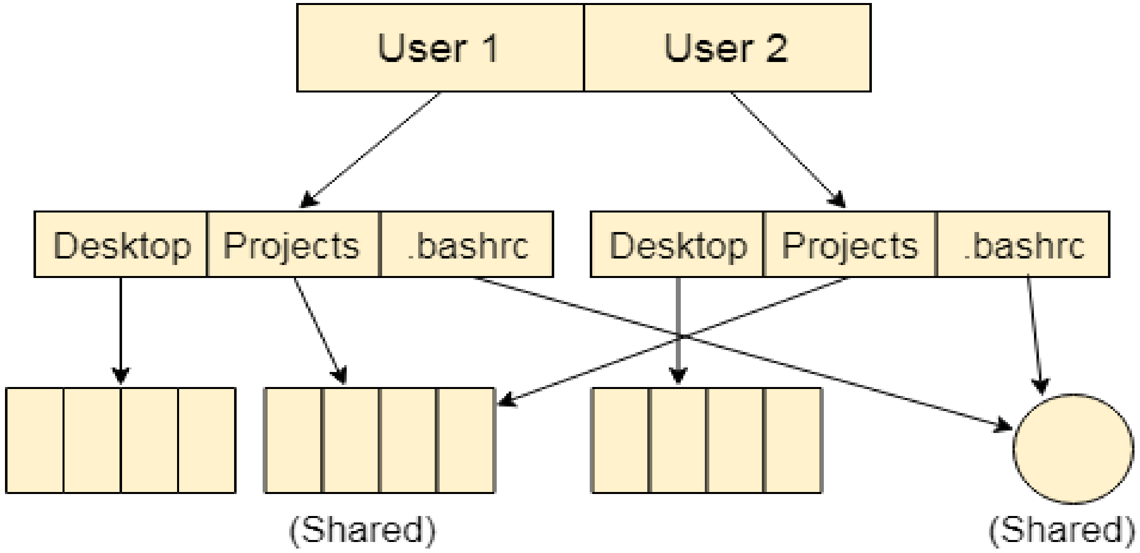
File Management

Acyclic-Graph Structured Directories

- If a file gets deleted in acyclic graph structured directory system, then

1. In the case of soft link, the file just gets deleted and we are left with a dangling pointer.

2. In the case of hard link, the actual file will be deleted only if all the references to it gets deleted.



Acyclic-Graph Structured Directory System

File Management

File Systems

File system is the part of the operating system which is responsible for file management. It provides a mechanism to store the data and access to the file contents including data and programs. Some Operating systems treats everything as a file for example Ubuntu.

The File system takes care of the following issues

File Structure

We have seen various data structures in which the file can be stored. The task of the file system is to maintain an optimal file structure.

Recovering Free space

Whenever a file gets deleted from the hard disk, there is a free space created in the disk. There can be many such spaces which need to be recovered in order to reallocate them to other files.

disk space assignment to the files

The major concern about the file is deciding where to store the files on the hard disk. There are various disks scheduling algorithm which will be covered later in this tutorial.

tracking data location

A File may or may not be stored within only one block. It can be stored in the non contiguous blocks on the disk. We need to keep track of all the blocks on which the part of the files reside.

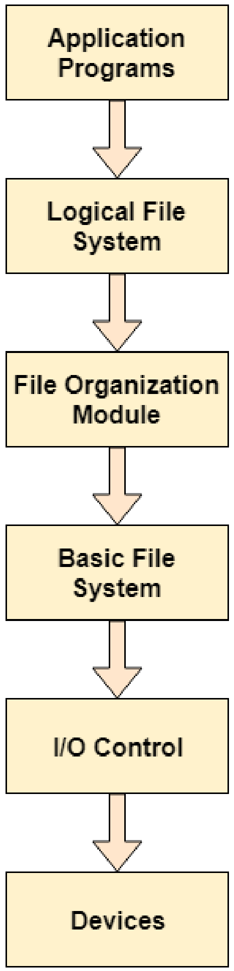
File Management

File System Structure

- File System provide efficient access to the disk by allowing data to be stored, located and retrieved in a convenient way. A file System must be able to store the file, locate the file and retrieve the file.
- Most of the Operating Systems use layering approach for every task including file systems. Every layer of the file system is responsible for some activities.
- The image shown below, elaborates how the file system is divided in different layers, and also the functionality of each layer.

File Management

File System Structure



File Management

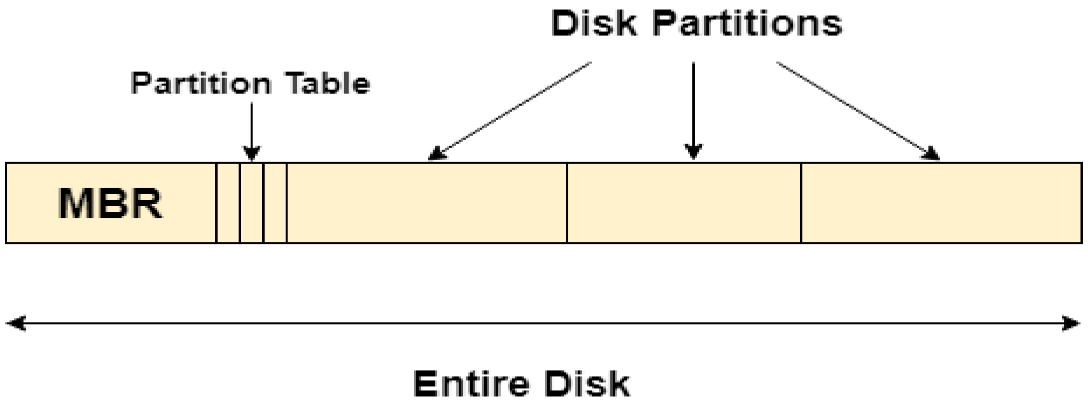
File System Structure

- When an application program asks for a file, the first request is directed to the logical file system. The logical file system contains the Meta data of the file and directory structure. If the application program doesn't have the required permissions of the file then this layer will throw an error. Logical file systems also verify the path to the file.
- Generally, files are divided into various logical blocks. Files are to be stored in the hard disk and to be retrieved from the hard disk. Hard disk is divided into various tracks and sectors. Therefore, in order to store and retrieve the files, the logical blocks need to be mapped to physical blocks. This mapping is done by File organization module. It is also responsible for free space management.
- Once File organization module decided which physical block the application program needs, it passes this information to basic file system. The basic file system is responsible for issuing the commands to I/O control in order to fetch those blocks.
- I/O controls contain the codes by using which it can access hard disk. These codes are known as device drivers. I/O controls are also responsible for handling interrupts.

File Management

Master Boot Record (MBR)

- Master boot record is the information present in the first sector of any hard disk. It contains the information regarding how and where the Operating system is located in the hard disk so that it can be booted in the RAM.
- MBR is sometimes called master partition table because it includes a partition table which locates every partition in the hard disk.
- Master boot record (MBR) also includes a program which reads the boot sector record of the partition that contains operating system.



File Management

What happens when you turn on your computer?

- Due to the fact that the main memory is volatile, when we turn on our computer, CPU cannot access the main memory directly. However, there is a special program called as BIOS stored in ROM is accessed for the first time by the CPU.
- BIOS contains the code, by executing which, the CPU access the very first partition of hard disk that is MBR. It contains a partition table for all the partitions of the hard disk.
- Since, MBR contains the information about where the operating system is being stored and it also contains a program which can read the boot sector record of the partition, hence the CPU fetches all this information and load the operating system into the main memory.

File Management

On Disk Data Structures

There are various on disk data structures that are used to implement a file system. This structure may vary depending upon the operating system.

Boot Control Block

Boot Control Block contains all the information which is needed to boot an operating system from that volume. It is called boot block in UNIX file system. In NTFS, it is called the partition boot sector.

Volume Control Block

Volume control block all the information regarding that volume such as number of blocks, size of each block, partition table, pointers to free blocks and free FCB blocks. In UNIX file system, it is known as super block. In NTFS, this information is stored inside master file table.

File Management

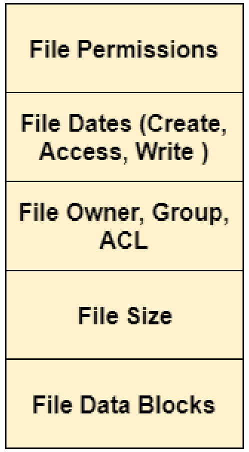
On Disk Data Structures

Directory Structure (per file system)

A directory structure (per file system) contains file names and pointers to corresponding FCBs. In UNIX, it includes inode numbers associated to file names.

File Control Block

File Control block contains all the details about the file such as ownership details, permission details, file size,etc. In UFS, this detail is stored in inode. In NTFS, this information is stored inside master file table as a relational database structure. A typical file control block is shown in the image below..



File Permissions

File Dates (Create, Access, Write)

File Owner, Group, ACL

File Size

File Data Blocks

File Control Block

File Management

In Memory Data Structure

The in-memory data structures are used for file system management as well as performance improvement via caching. This information is loaded on the mount time and discarded on ejection.

In-memory Mount Table

In-memory mount table contains the list of all the devices which are being mounted to the system. Whenever the connection is maintained to a device, its entry will be done in the mount table.

In-memory Directory structure cache

This is the list of directory which is recently accessed by the CPU. The directories present in the list can also be accessed in the near future so it will be better to store them temporally in cache.

System-wide open file table

This is the list of all the open files in the system at a particular time. Whenever the user open any file for reading or writing, the entry will be made in this open file table.

Per process Open file table

It is the list of open files subjected to every process. Since there is already a list which is there for every open file in the system thereforeIt only contains Pointers to the appropriate entry in the system wide table.

File Management

Directory Implementation

There are mainly two algorithms which are used in these days.

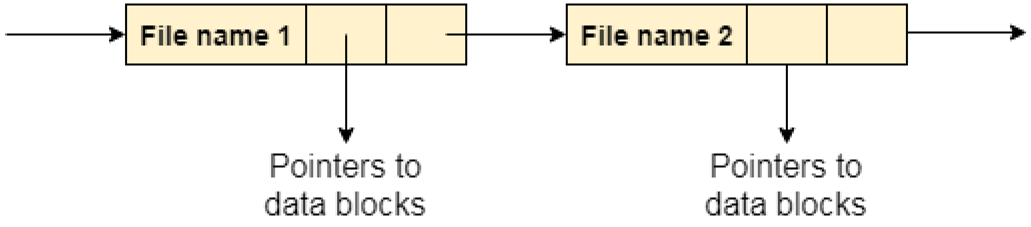
1. Linear List

In this algorithm, all the files in a directory are maintained as singly lined list. Each file contains the pointers to the data blocks which are assigned to it and the next file in the directory.

Characteristics

When a new file is created, then the entire list is checked whether the new file name is matching to a existing file name or not. In case, it doesn't exist, the file can be created at the beginning or at the end. Therefore, searching for a unique name is a big concern because traversing the whole list takes time.

The list needs to be traversed in case of every operation (creation, deletion, updating, etc) on the files therefore the systems become inefficient.



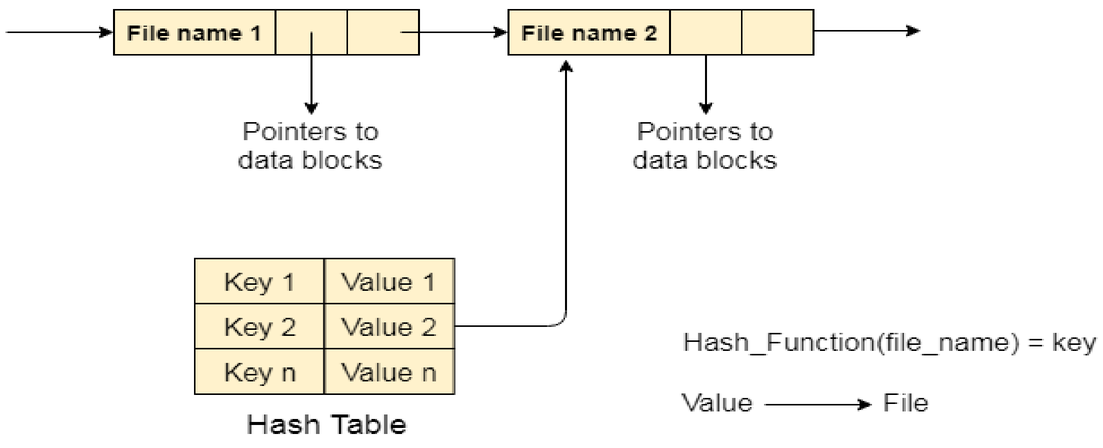
Linear List

File Management

Directory Implementation

2. Hash Table

- To overcome the drawbacks of singly linked list implementation of directories, there is an alternative approach that is hash table. This approach suggests to use hash table along with the linked lists.
- A key-value pair for each file in the directory gets generated and stored in the hash table. The key can be determined by applying the hash function on the file name while the key points to the corresponding file stored in the directory.
- Now, searching becomes efficient due to the fact that now, entire list will not be searched on every operating. Only hash table entries are checked using the key and if an entry found then the corresponding file will be fetched using the value.



File Management

Allocation Methods

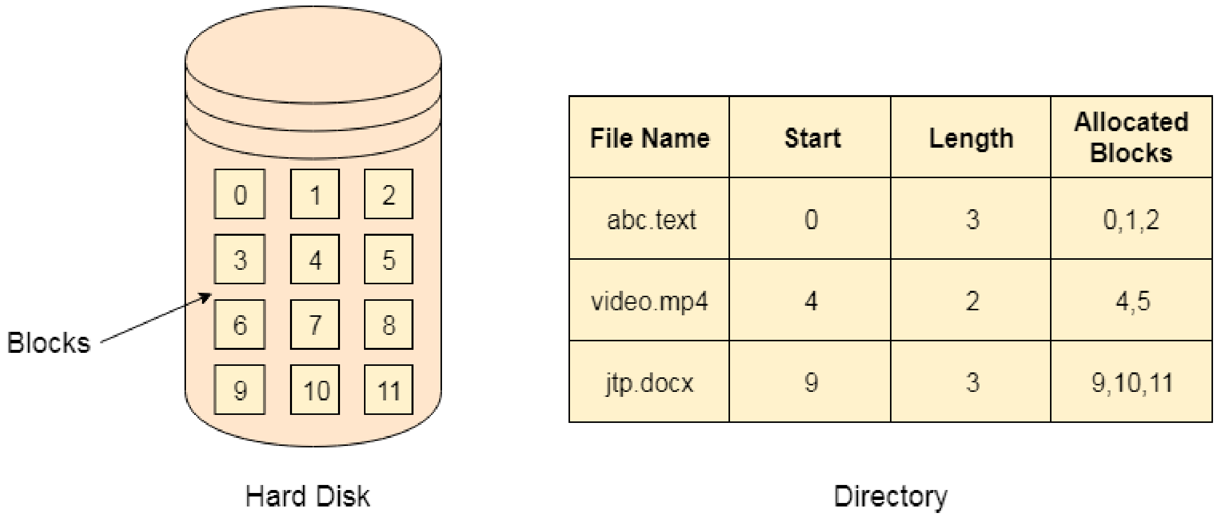
- There are various methods which can be used to allocate disk space to the files. Selection of an appropriate allocation method will significantly affect the performance and efficiency of the system. Allocation method provides a way in which the disk will be utilized and the files will be accessed.
- There are following methods which can be used for allocation.

1. Contiguous Allocation.
2. Extents
3. Linked Allocation
4. Clustering
5. FAT
6. Indexed Allocation
7. Linked Indexed Allocation
8. Multilevel Indexed Allocation
9. Inode

File Management

Allocation Methods

- **Contiguous Allocation**
- If the blocks are allocated to the file in such a way that all the logical blocks of the file get the contiguous physical block in the hard disk then such allocation scheme is known as contiguous allocation.
- In the image shown below, there are three files in the directory. The starting block and the length of each file are mentioned in the table. We can check in the table that the contiguous blocks are assigned to each file as per its need.



File Name	Start	Length	Allocated Blocks
abc.text	0	3	0,1,2
video.mp4	4	2	4,5
jtp.docx	9	3	9,10,11

Contiguous Allocation

File Management

Allocation Methods

- **Contiguous Allocation**

Advantages

- It is simple to implement.
- We will get Excellent read performance.
- Supports Random Access into files.

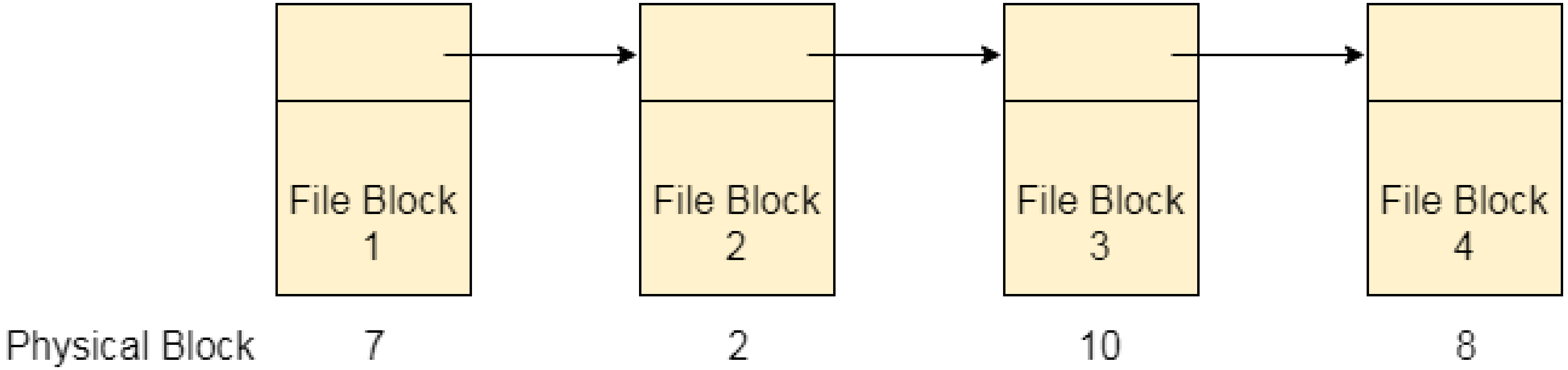
Disadvantages

- The disk will become fragmented.
- It may be difficult to have a file grow.

File Management

Allocation Methods

- **Linked List Allocation**
- **Linked List allocation solves all problems of contiguous allocation. In linked list allocation, each file is considered as the linked list of disk blocks. However, the disks blocks allocated to a particular file need not to be contiguous on the disk. Each disk block allocated to a file contains a pointer which points to the next disk block allocated to the same file.**



Linked List Allocation

File Management

Allocation Methods

- Linked List Allocation

Advantages

- There is no external fragmentation with linked allocation.
- Any free block can be utilized in order to satisfy the file block requests.
- File can continue to grow as long as the free blocks are available.
- Directory entry will only contain the starting block address.

Disadvantages

- Random Access is not provided.
- Pointers require some space in the disk blocks.
- Any of the pointers in the linked list must not be broken otherwise the file will get corrupted.
- Need to traverse each block.

File Management

Allocation Methods

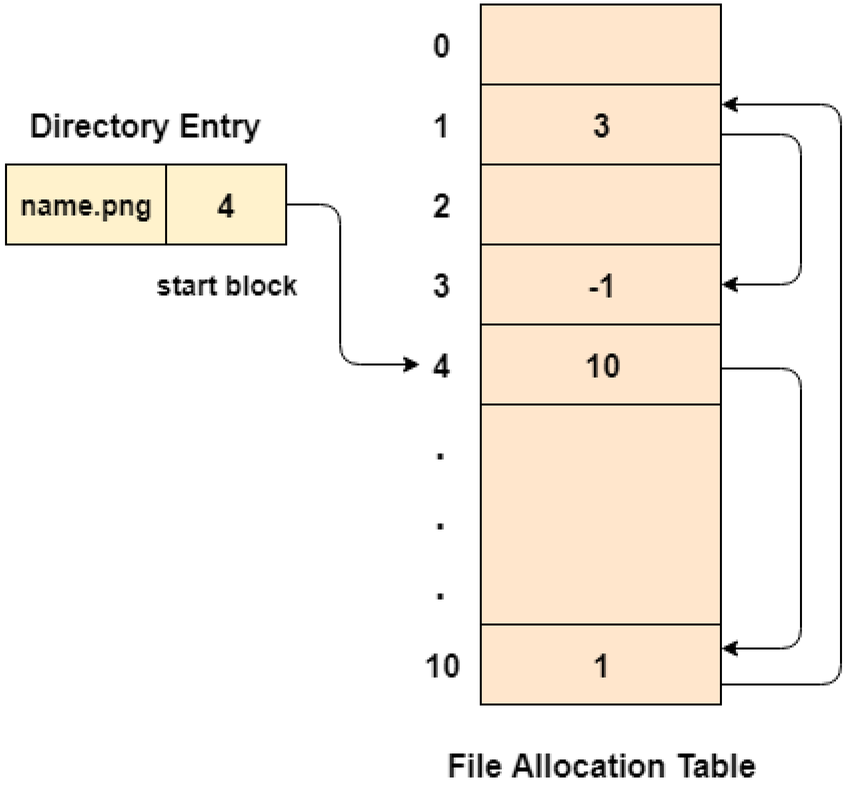
File Allocation Table

- The main disadvantage of linked list allocation is that the Random access to a particular block is not provided. In order to access a block, we need to access all its previous blocks.
- File Allocation Table overcomes this drawback of linked list allocation. In this scheme, a file allocation table is maintained, which gathers all the disk block links. The table has one entry for each disk block and is indexed by block number.
- File allocation table needs to be cached in order to reduce the number of head seeks. Now the head doesn't need to traverse all the disk blocks in order to access one successive block.
- It simply accesses the file allocation table, read the desired block entry from there and access that block. This is the way by which the random access is accomplished by using FAT. It is used by MS-DOS and pre-NT Windows versions.

File Management

Allocation Methods

File Allocation Table



File Management

Allocation Methods

Indexed Allocation

Limitation of FAT

Limitation in the existing technology causes the evolution of a new technology. Till now, we have seen various allocation methods; each of them was carrying several advantages and disadvantages.

File allocation table tries to solve as many problems as possible but leads to a drawback. The more the number of blocks, the more will be the size of FAT.

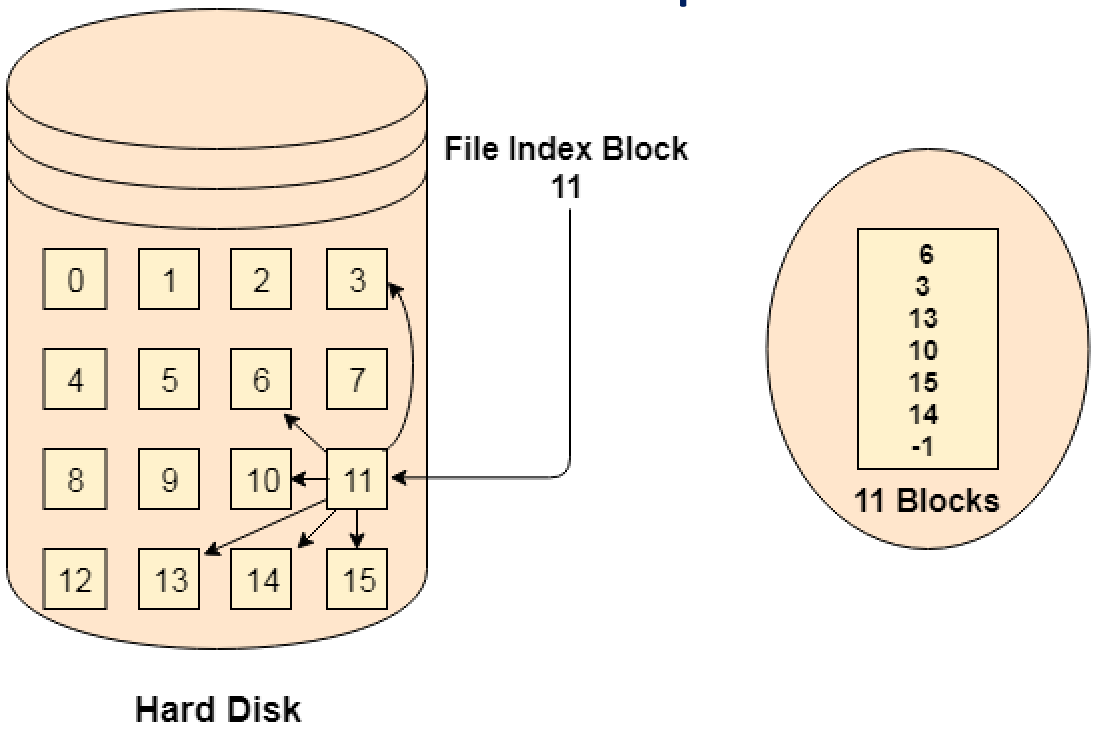
Therefore, we need to allocate more space to a file allocation table. Since, file allocation table needs to be cached therefore it is impossible to have as many space in cache. Here we need a new technology which can solve such problems.

File Management

Allocation Methods

Indexed Allocation Scheme

Instead of maintaining a file allocation table of all the disk pointers, Indexed allocation scheme stores all the disk pointers in one of the blocks called as indexed block. Indexed block doesn't hold the file data, but it holds the pointers to all the disk blocks allocated to that particular file. Directory entry will only contain the index block address.



File Management

Allocation Methods

Indexed Allocation Scheme

Advantages

- **Supports direct access**
- **A bad data block causes the lost of only that block.**

Disadvantages

- **A bad index block could cause the lost of entire file.**
- **Size of a file depends upon the number of pointers, a index block can hold.**
- **Having an index block for a small file is totally wastage.**
- **More pointer overhead**

File Management

Allocation Methods

Linked Index Allocation

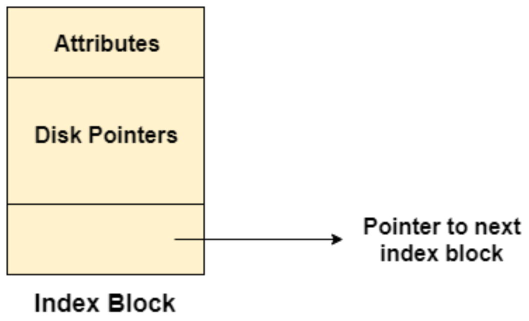
Single level linked Index Allocation

In index allocation, the file size depends on the size of a disk block. To allow large files, we have to link several index blocks together. In linked index allocation,

Small header giving the name of the file

Set of the first 100 block addresses

Pointer to another index block



For the larger files, the last entry of the index block is a pointer which points to another index block. This is also called as linked schema.

File Management

Allocation Methods

Linked Index Allocation

Advantage: It removes file size limitations

Disadvantage: Random Access becomes a bit harder

File Management

Allocation Methods

In Multilevel index allocation

In Multilevel index allocation, we have various levels of indices. There are outer level index blocks which contain the pointers to the inner level index blocks and the inner level index blocks contain the pointers to the file data.

The outer level index is used to find the inner level index.

The inner level index is used to find the desired data block.

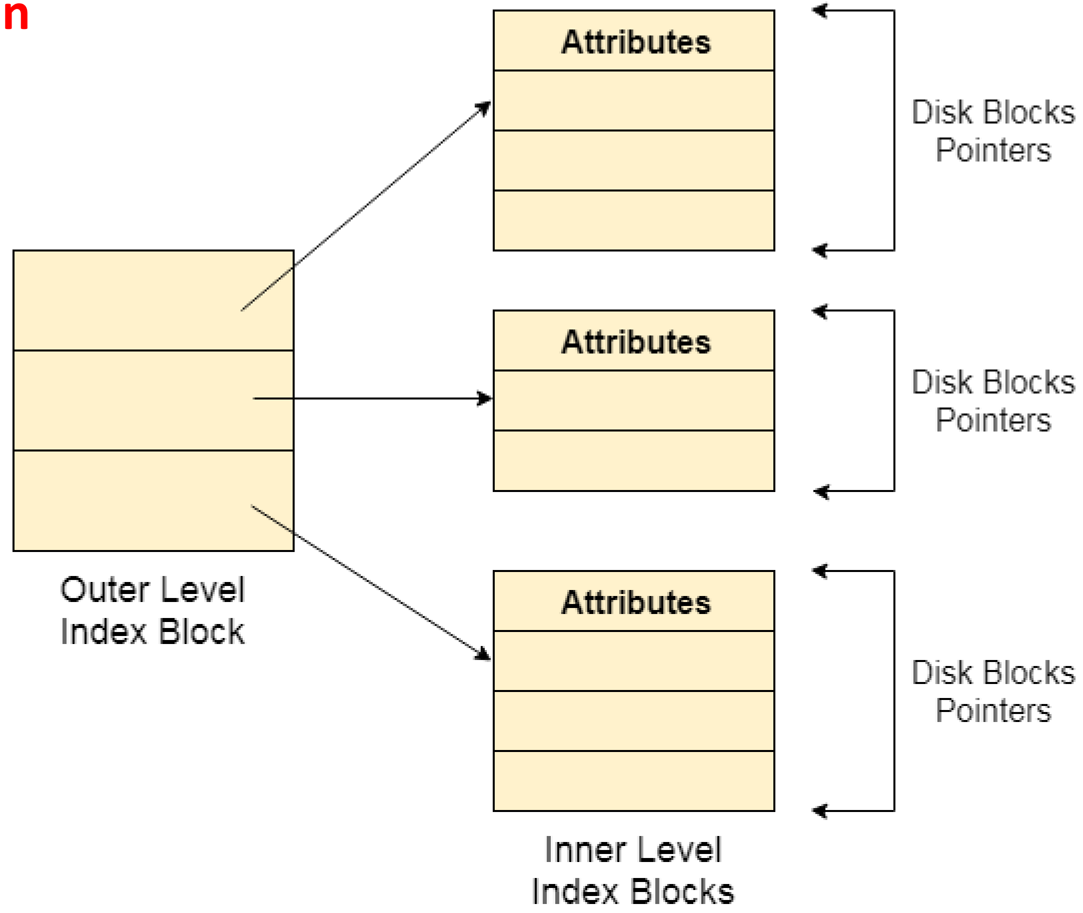
Advantage: Random Access becomes better and efficient.

Disadvantage: Access time for a file will be higher.

File Management

Allocation Methods

In Multilevel index allocation


Attributes
Attributes
Attributes
Disk Blocks Pointers
Disk Blocks Pointers
Disk Blocks Pointers
Outer Level Index Block
Inner Level Index Blocks