

# File System

## What is a file?

A file can be defined as a data structure that stores a sequence of records. The file is stored in the file system and can reside on a disk or in the main memory. The file can be simple (text only) or complex (specially formatted). A collection of files is called a directory. The collection of directories at different levels is called the file system. Different file attributes give information about the file. These are

## File attributes

The file system stores file attributes in the file's data block. The file system also uses dynamic file attributes, which are automatically appended to or subtracted from a file's data block when modified.

## Stay updated with the latest blogs on online courses and skills

Enter Mobile Number

Register Now

- **Name-** This is the only information stored in a human-readable format. A name is given to each file. One directory cannot contain two files with the same name.
- **Extension-** This attribute is required for systems that support different file types. For example, a text file has a .txt extension and a video file has a .mp4 extension.
- **Location-** Refers to the location of the file on the device. The file system has several places where you can save your files. Each file contains its location as an attribute.
- **Size-File-** size means the number of bytes captured by the file in memory.
- **Identifier-** Each file is identified by a unique tag number within the file system called an identifier.
- **Protection-** This attribute assigns and controls access to read, write, and execute files. Computer administrators may need different protection for each file. Therefore, each file has its own permissions on different user groups.
- **Compression-** It tells the type of compression there is.
- **Created/Modified date-** Each file has a timestamp that includes the date and time the file was last modified.

## Operations on file

- **Creating-** Creating a file involves two steps. First, check if there is free space. You must create a new directory file entry if you have enough space.
- **Opening-** This is a common operation performed on a file. After you create a file, you need to open it before you can perform any file processing operations. If the user

wants to open the file, call a system call to instruct the operating system to open the file and pass the file name to the file

- **Writing-** To write a file, you need to know two things. One is the name of the file and the other is the information or data written to the file
- **Deleting-** To delete a file, first browse the directory of the named file, then free up the file area and delete the directory entry.
- **Truncating-** To truncate a file, delete only the contents of the file, but don't delete its file structure.
- **Close File-** After **editing** the **file**, **you need to close** the file **to free up any permanent** changes made and occupied **resources**. **Finally**, the internal **descriptor** created when the file was **opened is released**.

## File system in an operating system

*A file system is a collection of files and directories used by an operating system to organize the storage of files and to provide a pathway for users to access those files.*

A file system is a software layer that manages files and folders on an electronic storage device, such as a hard disk or flash memory. File systems allow different programs to access and use the same storage locations without knowing each other's internal details.

Your operating system has a file system that helps keep track of your files. The file system organises and manages files and directories (or folders). It also keeps track of each file's space on your hard drive.

There are a few different file system types, each with unique benefits and uses. In this post, we'll look at the different types of file systems and discuss when it's best to use them.



## [Swapping in Operating System](#)

[Swapping is a memory management technique that can increase the operating system's performance. This article describes Swapping in the operating system and how to swap a memory management technique in...read more](#)

## [Read Later](#)



## [Interrupts in Operating System](#)

[This article focuses on operating system interrupts. It also focuses on handling interrupts.](#)



### **Virtual Memory with Real-life analogy**

[Your computer wouldn't be able to do much without virtual memory. This little-known feature of your operating system makes it possible for you to open multiple programs simultaneously. surf the...read more](#)

### **Why do you need a file system?**

Without a file system, storage devices store large blocks of data simultaneously, and the operating system cannot distinguish between them. The main reason for using a file system are:

- To improve performance in terms of response time.
- To eliminate or mitigate the risk of data loss or corruption
- To meet your data management needs and requirements
- Easy access to your data.

Imagine a room with a bunch of paper scattered all over the place. You won't be able to access the papers. So it is better to arrange them in files or folders.



### **Operating System Interview Questions**

[In this article you will find important operating system questions which are likeable to be asked in interview.](#)

### **Functions of file system**

- The purpose of a file system is to provide a way for users to access files and directories. The file system does this by providing a directory structure and a set of rules for how files and directories are stored.
- Provides security against information loss due to system failure.
- If necessary, a file recovery module is also provided.
- The file system provides a pathway for users to access files and directories by storing them in a specific location on the file system. The pathname determines the pathway, a sequence of file system components that specifies where the file or directory should be located on the file system.

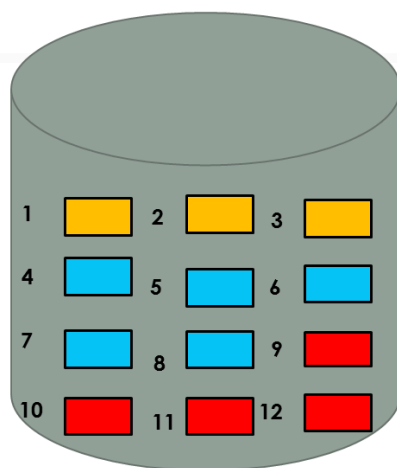
**You May Like – [Difference Between Paging And Segmentation](#)**

## File allocation methods

Allocation refers to allocating memory, file space, and CPU time to running processes. Different types of allocation methods

### 1. Contiguous allocation method

Contiguous allocation is the default allocation scheme in most current file systems. A computer can store data in memory contiguously, meaning that every block of data is located next to each other. This makes it easy for multiple processes to access the same data simultaneously. It is especially convenient when the data is small since it requires fewer memory reads and writes to access it.



File	Start	Length
A	0	3
B	6	5
C	14	4

Suppose we want to store files A, B, and C, then this file is broken into different blocks. All the blocks are stored. If we want to store files A, B, and C, each file is broken into different blocks. All the blocks are stored at different memory locations, which are available contiguously. That means if file A is to be stored, then all the blocks of file A will be stored and then all blocks of file B and then all blocks of file C. Suppose the first block of file A is stored at memory location 500 then the next block B2 will be at 501 and third block will be at 503. And then file B's first block will be 504, and so on.

#### Advantages

- Easy to implement
- Excellent read performance

#### Disadvantages

- Difficult to grow the file. That means if the file size is increased it will be difficult to accommodate that file in memory.
- Suffers internal as well as external fragmentation.
- Memory utilization is inefficient.

**NOTE:** The memory needed to store the file should be known beforehand.

### 2. Non-contiguous allocation

## Linked allocation

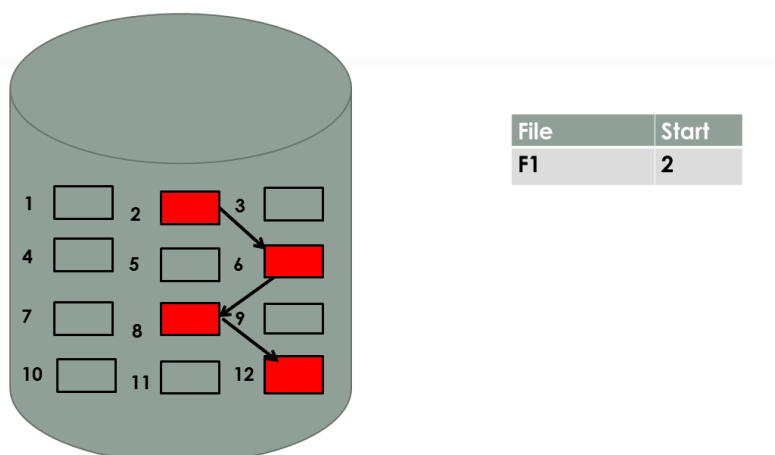
The second type of allocation is non-contiguous allocation. In this type of allocation, files are not stored in one contiguous block on the disk. This can be more efficient because it uses the space on the disk more efficiently, but it's more challenging to manage because files can be scattered all over the disk.

### Advantages

- No external Fragmentation
- File size can increase

### Disadvantages

- Large seek time.
- Pointers overhead
- Memory consumption will be more as pointers will take memory to get stored.
- Random access/direct size is difficult.



Suppose we want to store file F1 then this file is broken into different blocks B1, B2, B3, B4. All the blocks are stored at different memory locations, which are available in a **non-contiguous manner**. And each block will have information and the address of the next block. That means random blocks can be accessed. All blocks can be utilized. So no external fragmentation will be there.

## Indexed allocation

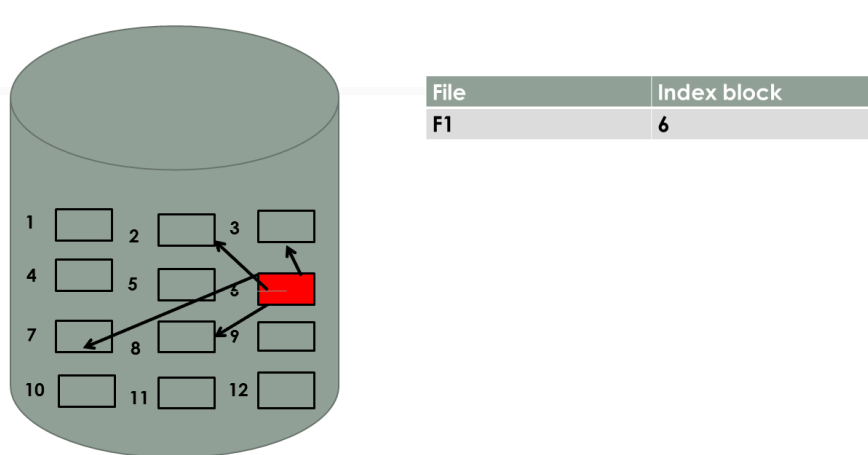
The third type of allocation is index allocation. This creates a special index file that stores information about all the files on the disk. This is the most efficient way to store files but also the most complicated and challenging to manage.

### Advantages

- Supports direct access
- No external fragmentation

### Disadvantages

- Pointer overhead
- Multilevel index
- Every file has individual index blocks to store the addresses for disk space.



Suppose we want to store file F1 then this file is broken into different blocks B1, B2, B3, B4, and B5. All the blocks are stored at different memory locations which are available in a **non-contiguous manner**. The thing to note here is that Every file will have an indexed block which will guide which block is guided to different blocks and where they are present. Like in this case the index block is 6. ***It is the same as the index of a book. Using this you come to know which topic is on which page.*** So index block is like a reference for all blocks. It supports random access. That means random blocks can be accessed. All blocks can be utilized. So no external fragmentation.

## Types of File System

Windows also use the Inode File System to store files and directories not stored on the disk drive. The file system that is used on a particular computer depends on the type of computer that the computer is. For example, a computer used in a business environment might have a different file system than a computer used in a home office.

Regarding the different types of file systems, there are three main categories: FAT, NTFS, and Ext2. Let's take a closer look at each one.

### 1. FAT

FAT is the oldest and most commonly used type of file system. It was developed in the early 1980s and is found in most Windows operating systems. FAT is a fairly basic file system without many securities or features. The FAT directory structure is unorganized, and the files are given the first open location on the drive. In addition, FAT only supports read-only, hidden, system, and archive file attributes.

### 2. NTFS

NTFS is the newer and more advanced type of file system. It was introduced with Windows NT and offered improved security, performance, and flexibility over FAT. NTFS is the default file system for Windows 7 and later versions. NTFS is a recoverable file system because it keeps track of transactions against the file system. So, backup and security mechanism is there in it. More storage capacity

### **3. HPFS**

The HPFS file system, first introduced in OS / 2 1.2, provides better access to larger hard drives that have since been on the market. In addition, new file systems were needed to add naming systems, organizations, and security to meet the growing network server market demands. HPFS maintains the FAT directory organization but adds automatic directory sorting based on filenames.

Thus, there are many different file system types, each with its unique uses. You should choose the file system that is best suited for your needs, and you should also be aware of the pros and cons of each type.

The more complex the file system, the more options you have for storing and accessing your data. However, complex file systems can also be more challenging to use and manage.