Operating System(OS)

BCA IV SEM OS

File system

AS OPERATING SYSTEM

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.
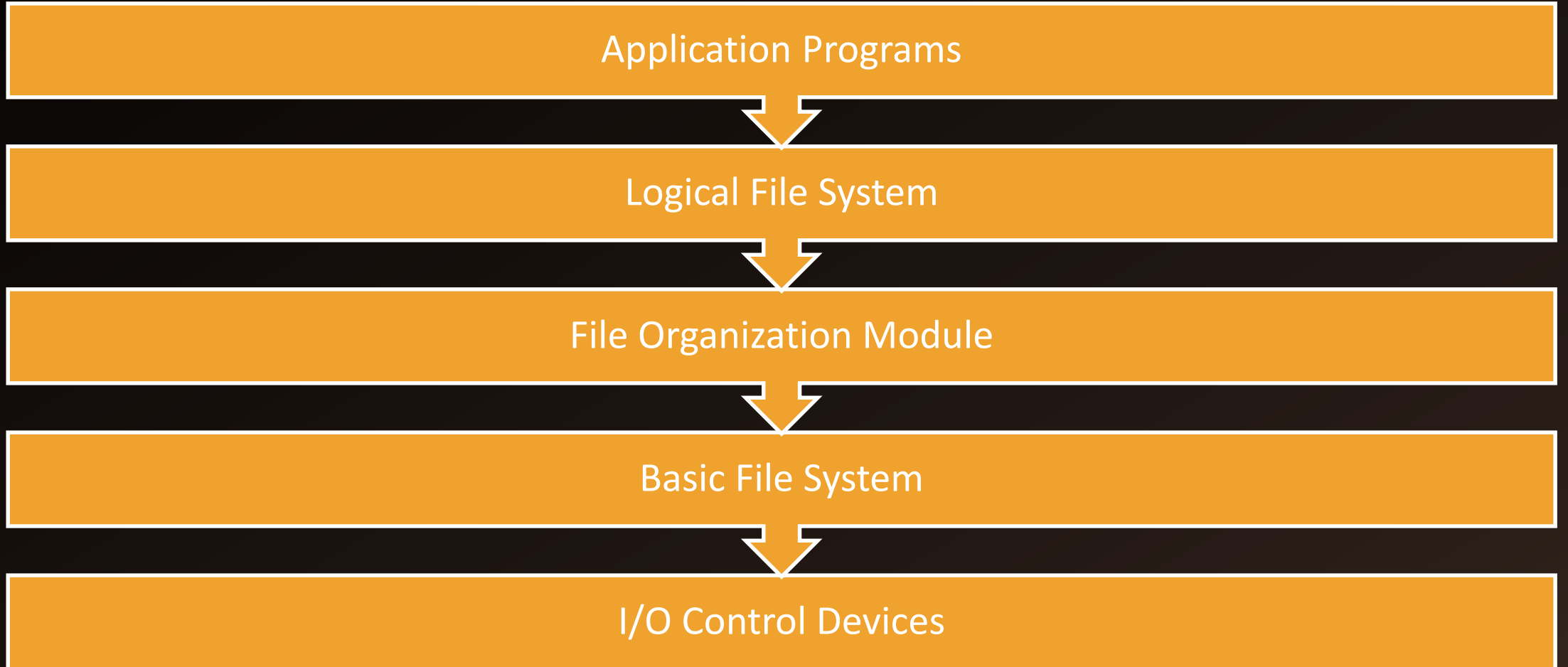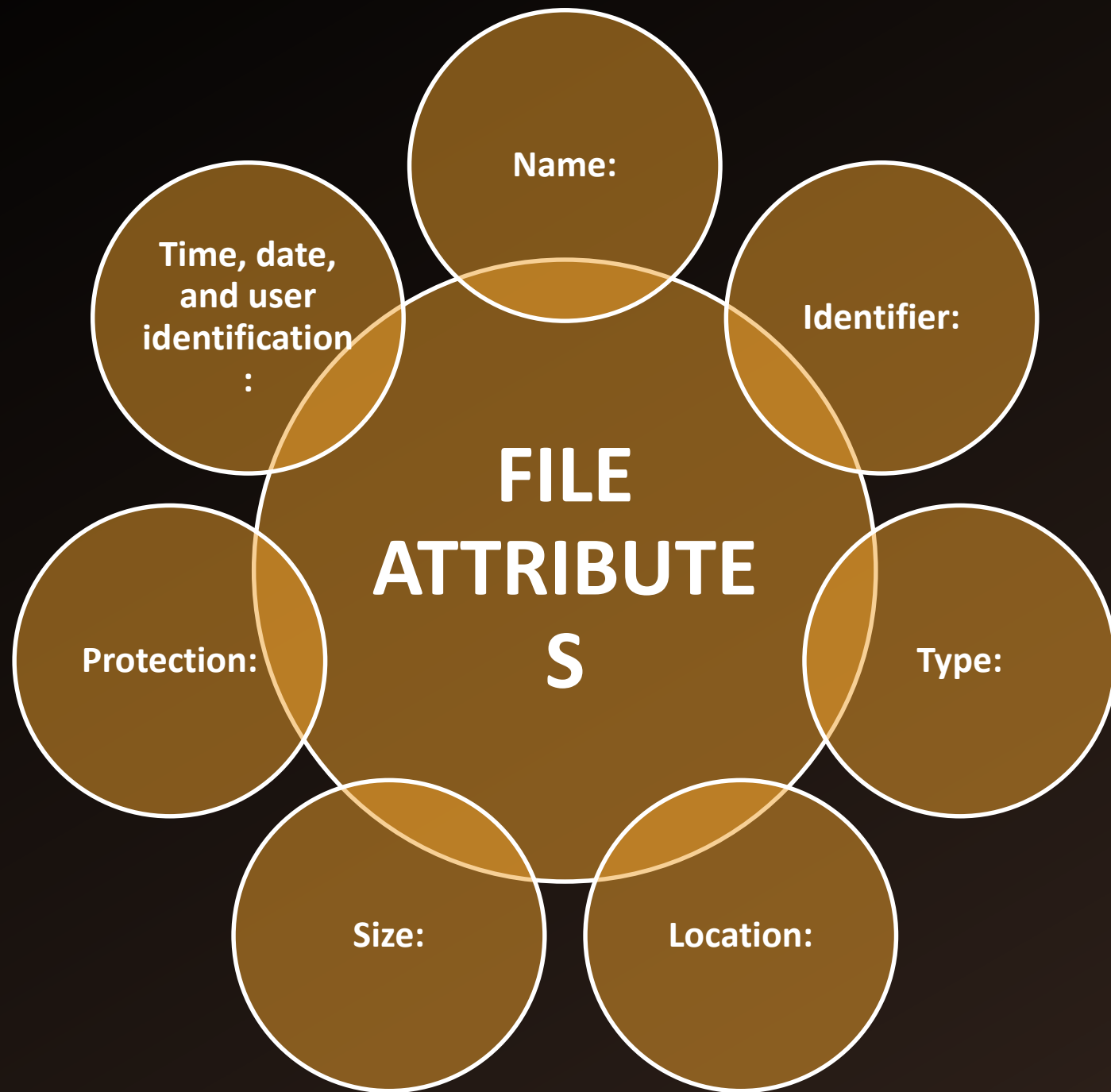
File System Structure

Application Programs

Logical File System

File Organization Module

Basic File System

I/O Control Devices

FILE ATTRIBUTES

Name:	Identifier:	Type:	Location:	Size:	Protection:	Time, date, and user identification:
File name is the name given to the file. A name is usually a string of characters.	Identifier is a unique number for a file. It identifies files within the file system. It is not readable to us, unlike file names.	Type is another attribute of a file which specifies the type of file such as archive file (.zip), source code file (.c, .java), .docx file, .txt file, etc.	Specifies the location of the file on the device (The directory path). This attribute is a pointer to a device.	Specifies the current size of the file (in Kb, Mb, Gb, etc.) and possibly the maximum allowed size of the file.	Specifies information about Access control (Permissions about Who can read, edit, write, and execute the file.) It provides security to sensitive and private information.	This information tells us about the date and time on which the file was created, last modified, created and modified by which user, etc.

OS File Operations

File Creation and Manipulation

1. Creating Files
2. Creating Directories
3. Opening Files
4. Reading Files
5. Writing Files
6. Renaming Files and Directories
7. Deleting Files and Directories

File organization and search

1. Copying Files
2. Moving Files
3. Searching for Files

File Security and Metadata

1. File Permissions
2. File Ownership
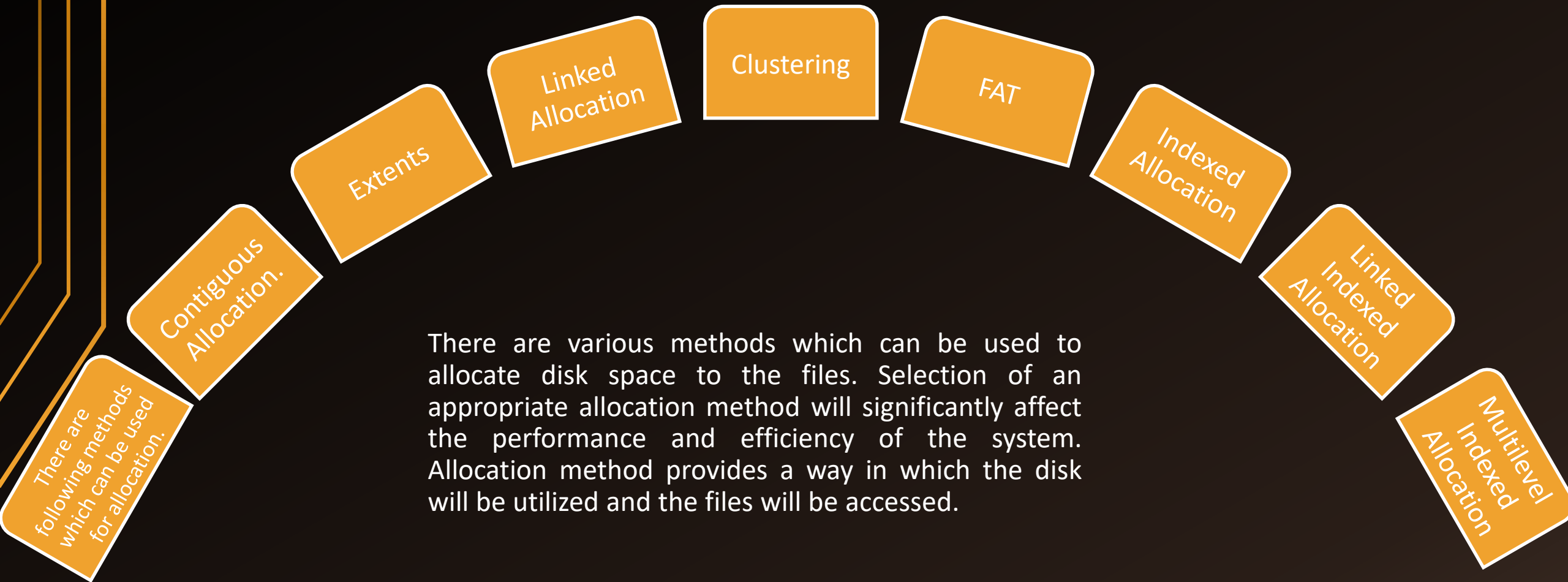3. File Metadata

File Compression and Encryption

1. File Compression
2. File Encryption

Operations on the File

1.Create operation :	2. Open operation :	3. Write operation :	4. Read operation :	5. Re-position or Seek operation:	6. Delete operation :	7. Truncate operation :	8. Close operation :	9. Append operation :	10. Rename 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.	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.	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.	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.	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.	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.	Truncating is simply deleting the file except deleting attributes. The file is not completely deleted although the information stored inside the file gets replaced.	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.	This operation adds data to the end of the file.	This operation is used to rename the existing file.

FILE ALLOCATION METHODS

There are following methods which can be used for allocation.

Contiguous Allocation.

Extents

Linked Allocation

Clustering

FAT

Indexed Allocation

Linked Indexed Allocation

Multilevel Indexed Allocation

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.

A file system is responsible to allocate the free blocks to the file therefore it has to keep track of all the free blocks present in the disk. There are mainly two approaches by using which, the free blocks in the disk are managed.

Free Space Management

1. Bit Vector

2. Linked List

In this approach, the free space list is implemented as a bit map vector. It contains the number of bits where each bit represents each block.

If the block is empty then the bit is 1 otherwise it is 0. Initially all the blocks are empty therefore each bit in the bit map vector contains 1.

LAs the space allocation proceeds, the file system starts allocating blocks to the files and setting the respective bit to 0.

It is another approach for free space management. This approach suggests linking together all the free blocks and keeping a pointer in the cache which points to the first free block.

Therefore, all the free blocks on the disks will be linked together with a pointer. Whenever a block gets allocated, its previous free block will be linked to its next free block.

Thanks

@KAMALNAINX
@THEKAMALNAIN