



FILE MANAGEMENT




File Concept

- ◆ Computers can store information on different storage media, such as disks, tapes, optical disks etc...
- ◆ A file is a named collection of related information that is recorded in secondary storage unit.
- ◆ A file is a sequence of bits, bytes, lines or records whose meaning is defined by the file creator



File Operations

- ◆ Creating a file :- Build a new file
- ◆ Deleting a file :- Remove a file
- ◆ Copy :- Create another version of a file with new name
- ◆ Rename :- Change the name of a file
- ◆ Open :- Prepare a file to be referenced
- ◆ Close :- Prevent further reference to a file
- ◆ List :- Print or display the contents of a file



Individual items can be manipulated by the operations like

- ◆ **Read** :- Input a data item to a process from a file
- ◆ **Write** :- Output a data item from a process to a file
- ◆ **Insert** :- Add new data item to a file
- ◆ **Delete** :- Remove data item from a file



- ◆ Files may be characterized by
- ◆ **Volatility** :- This refers to the frequency with which additions and deletions are made to a file.
- ◆ **Activity** :- This refers to the percentage of a file's records accessed during a given period of time.
- ◆ **Size** :- This refers to the amount of information stored in the file.

File Attributes

- ◆ File Name
- ◆ Type
- ◆ Location
- ◆ Size
- ◆ Protection
- ◆ Time, date, and user identification
 - The information about all files is kept in the directory structure that also resides in the secondary storage





The File System

An important component of the OS is the file system. File system generally contain.

- ◆ **Access Methods** :- These are concerned with the manner in which data stored in files is accessed.
- ◆ **File Management** :- Concerned with allocating space for files to be stored, referenced, shared, and secured.
- ◆ **Auxiliary Storage Management** : - Concerned with allocating space for files on Secondary Storage Devices.
- ◆ **File integrity mechanism** :- These are concerned with guaranteeing that the information in a file is uncorrupted..



File system Functions

- ◆ User should be able to create, modify, and delete files
- ◆ User can share others files in a carefully controlled manner
- ◆ User can structure their files in a manner most appropriate for each application
- ◆ Provides access permissions
- ◆ Provides backup and recovery capabilities
- ◆ Provides security



File Descriptor

- ◆ A file descriptor is a data structure containing information needed to manage a file. A file descriptor may include
 - Symbolic file name
 - Location of file in secondary storage
 - File organization
 - Device type
 - Access control data
 - File type
 - Creation date and time
 - Destroy date
 - Date and time last modified
 - Access activity counts



File Organization

- ◆ File organization refers to the manner in which the records of a file are arranged on secondary storage.
- ◆ Sequential
 - Records are placed in the physical order
 - The *Next* record is the one that physically follows the previous record.
- ◆ Direct
 - Records are directly accessed by their physical addresses.
- ◆ Indexed Sequential
 - Records are arranged in logical sequence according to a key contained in each record.




Allocation Methods


- ◆ The allocation methods deals with how to allocate space to the files so that disk space is utilized effectively and files can be accesses quickly.
- ◆ Three major methods of allocating disk space are
 - Contiguous
 - Linked
 - Indexed
- ◆ Each of these method has its own advantages and disadvantages. Some systems support all three

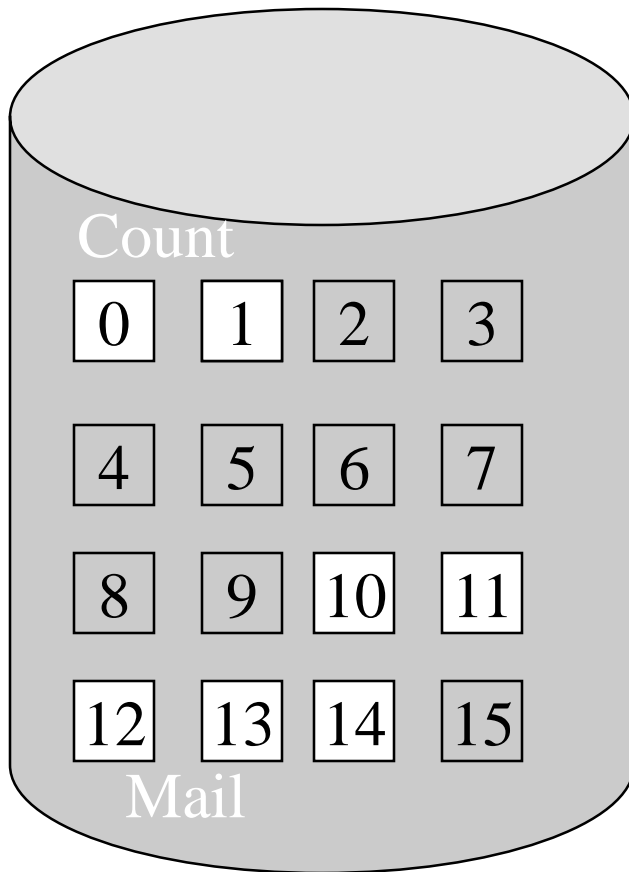


Contiguous Allocation

- ◆ The contiguous allocation method requires each file to occupy a set of contiguous block on the disk
- ◆ Contiguous allocation of a file is defined by the disk address and length of the first block. If the file is n blocks long, and starts at location b then it occupies blocks $b, b+1, b+2, \dots, b+n-1$.
- ◆ The directory entry for each file indicates the address of the starting block and the length of area allocated for this file

- 
- ◆ Accessing a file is easy
 - ◆ It supports sequential and direct access
 - ◆ For sequential access the file system remembers the address of the last block and when necessary reads the next block
 - ◆ For direct access to block I with starting block b , we can immediately access block $B+i$

- 
- ◆ Difficulties in contiguous allocation
 - Finding free space for a new file
 - External fragmentation
 - Determining how much space is needed for a file
 - Chance of occurring internal fragmentation



Directory

File	Start	Length
Count	0	2
Mail	10	5



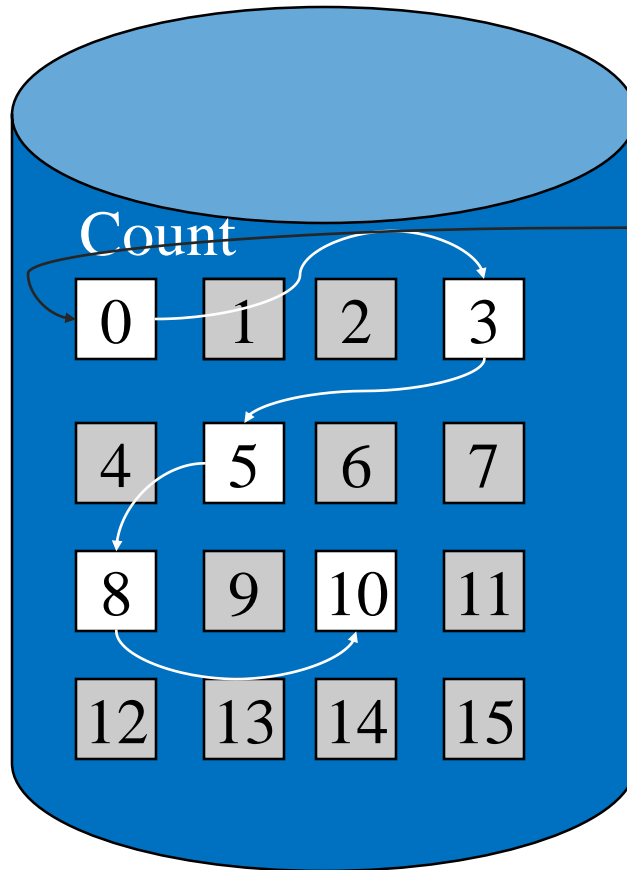
Linked Allocation

- ◆ In this method each file is a linked list of disk blocks
- ◆ The disk blocks can be scattered any where in the disk
- ◆ The directory contains a pointer to the first and last block of the file
- ◆ To create a new file make corresponding entry in the directory list with the pointer initialized to null. A write to file causes a free block to be found and this new block is then written to, and is linked to the end of the file
- ◆ There is no external fragmentations

Linked Allocation

Directory

File	Start	End
Count	0	10





◆ Drawbacks

- Allows sequential access only
- Space required for the pointers
- Reliability



Indexed Allocation

- ◆ Each file has its own index block, Which is an array of disk block addresses
- ◆ The directory list contains the address of the index block of that file
- ◆ The i^{th} entry in the index block points to the i^{th} block of the file
- ◆ To read the i^{th} block , use the pointer in the i^{th} block entry to find and read the desired disk block
- ◆ It allows direct access with no external fragmentation



Directory

File	Index block
Count	14

