OS – INFORMATION MANAGEMENT

J. FATHIMA FOUZIA

Assistant Professor

Department Of CS & IT

Jamal Mohamed College (Autonomous)

Trichy - 620020

FILE SYSTEM MODEL

- In a computer, a file system is the way in which files are named and where they are placed logically for storage and retrieval.
- Without a file system, stored information wouldn't be isolated into individual files and would be difficult to identify and retrieve.
- File systems can differ between operating systems (OS), such as Microsoft Windows, macOS and Linux-based systems.
- Some file systems are designed for specific applications.
- Major types of file systems include distributed file systems, disk-based file systems and special purpose file systems.

FILE SYSTEM MODEL - Continued

- File systems can also restrict read and write access to a particular group of users. Passwords are the easiest way to do this.
- Major file systems include the following:
  - File allocation table (FAT)
  - Global file system (GFS)
  - Hierarchical file system (HFS)
  - The New Technology file system (NTFS)
  - Universal disk format (UDF)

SYMBOLIC FILE SYSTEMS

- An SFS is a managed directory which is initialized with the command: sfs init.
- All commands to be executed in the context of an individual SFS must be run from within the SFS directory tree.
- Files are added using the command sfs add-col my_collection /path/to/source (add collection).

BASIC FILE SYSTEM

- FILE
  - A **file** is a container that holds information. Most of the files you use contain information (data) in some particular format--a document, a spreadsheet, a chart.
- FOLDERS
  - A **folder** is a container for files, similar to a folder in a file cabinet. In fact, File Manager uses a folder icon to represent a folder. A folder can contain other folders--sometimes called subfolders.

BASIC FILE SYSTEM

- OBJECTS
  - Since files and folders are both represented in File Manager as icons, the term **object** is used to describe them both.
- PATHS
  - The location of a file is often specified by listing the folders and subfolders that lead to the file--this list is called a **path**. A file's path is visible in two places in File Manager.

BASIC FILE SYSTEM

- PATHS AND PATH NAMES
  - The path to an object is a way to specify where the object is located in the file system. There are two ways to specify the path: absolute path and relative path.
- ABSOLUTE PATHS
  - A path is an **absolute path** if it begins at the root folder. The **root folder** is the single common folder on your system where the hierarchy begins. If a path begins with a slash (/), it is an absolute path specified from the root folder.

BASIC FILE SYSTEM

- RELATIVE PATHS
  - A path is **relative** if it describes the location of a file or folder as it relates to the current folder. If you are in a folder and you want to move down the folder tree, you don't need to type the absolute path name.

LOGICAL FILE SYSTEM

- Logical files do not contain data.
- They contain a description of records that are found in one or more physical files.
- A logical file is a view or representation of one or more physical files.
- Logical files that contain more than one format are referred to as multi-format logical files.

PHYSICAL FILE SYSTEM

- Physical file contains the actual data that is stored in the system.
- It describes how the data is to be presented .
- It contains one popular format for presenting data that is record format.
- Other formats are also available.
- The record format contains one or more field names and corresponding data.
- A field name can be given first either in column wise or row wise.
- A file has one or more records.
- A database has one or more files.

TYPES OF PHYSICAL FILE SYSTEM

- Pile
- Indirect or Sequential Access
- Direct or Random Access
- Indexed Access

TYPES OF PHYSICAL FILE SYSTEM

- Pile
  - In this method, data is stored in the file in the order they arrive.
  - Records may have different fields.
  - Record search is difficult.
- Indirect or Sequential Access
  - In this method, data is stored in the file in the order they arrive.
  - Records may have different fields.
  - Record search is difficult.

TYPES OF PHYSICAL FILE SYSTEM

- Direct or Random Access
  - Records are stored in any order and the accessing is direct.
  - When a random record or last record needs to be accessed, no need to access from the beginning or previous pointer value.
  - Accessing becomes easy but still searching is difficult.
- Indexed Access
  - A file is created an index which contains all the information about each records and their location.
  - With that, searching and accessing becomes easy.

FILE ALLOCATION METHODS

- Files are allocated disk storage using 3 ways:
  - Contiguous Allocation
  - Linked Allocation
  - Indexed Allocation

CONTIGUOUS ALLOCATION

- Each file occupies a contiguous address space on a disk and is Easy to implement.
- External fragmentation problem arises.
- ADVANTAGES
  - Simplicity
  - Easy to maintain and configure.
- DISADVANTAGES
  - Finding space for a new file is difficult.
  - External fragmentation problem arises.
  - Needs compaction,
  - Compaction computation time.

LINKED ALLOCATION

- Each file carries data and a link to other block for taking next record.
  - No external fragmentation.
  - Effectively sequential access.
  - Not suitable for random access.
- ADVANTAGES
  - Easy reference of data.
  - File data is scattered in the disk.
  - Searching is easy.
  - Avoids external fragmentation.
- DISADVANTAGES
  - Inserting pointer link in each data of the file is work overhead.

INDEXED ALLOCATION

- Indexed allocation contains a block for holding all pointer values.
- The index block address will be given. This index block contains all the pointer links referring which block to take the data next.
- Searching and accessing becomes easy here.
- ADVANTAGES
  - Searching and accessing of data is easy.
  - Solves external fragmentation problem.

Reference

- S.E. Madnick & J.J.Donovan, Operating Systems, McGraw Hill International Book Co., New Delhi 2017.
- James L. Peterson & Abaraham Silbertschatz, An Introduction To Operating Systems, Addison-wesley Publishing Co., New York, 1987.