

OPERATING SYSTEMS

File Management

Suresh Jamadagni

Department of Computer Science

OPERATING SYSTEMS

File System

Suresh Jamadagni

Department of Computer Science

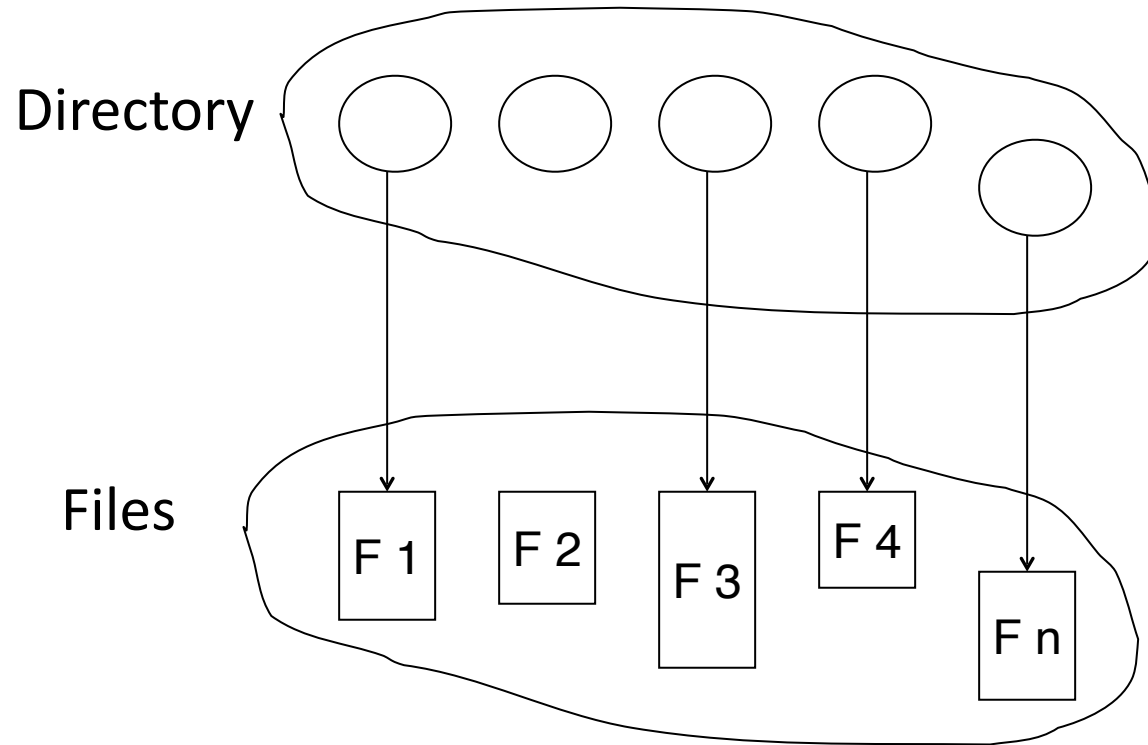
OPERATING SYSTEMS

Slides Credits for all the PPTs of this course



- The slides/diagrams in this course are an **adaptation, combination,** and **enhancement** of material from the following resources and persons:
 1. Slides of Operating System Concepts, Abraham Silberschatz, Peter Baer Galvin, Greg Gagne - 9th edition 2013 and some slides from 10th edition 2018
 2. Some conceptual text and diagram from Operating Systems - Internals and Design Principles, William Stallings, 9th edition 2018
 3. Some presentation transcripts from A. Frank – P. Weisberg
 4. Some conceptual text from Operating Systems: Three Easy Pieces, Remzi Arpaci-Dusseau, Andrea Arpaci Dusseau

? A collection of nodes containing information about all files



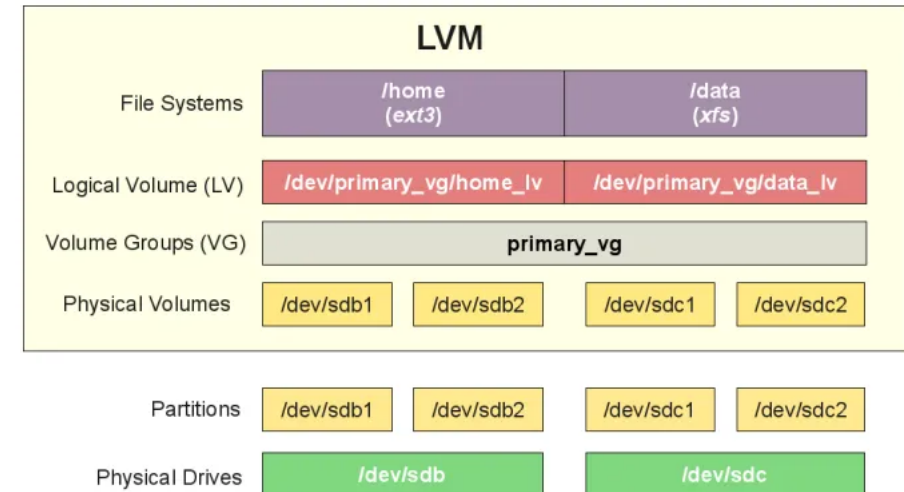
Both the directory structure and the files reside on disk

OPERATING SYSTEMS

Disk Structure



- ? Disk can be subdivided into **partitions**
- ? Disks or partitions can be **RAID** protected against failure
- ? Disk or partition can be used **raw** – without a file system, or **formatted** with a file system
- ? Partitions also known as minidisks, slices
- ? Entity containing file system known as a **volume**
- ? Each volume containing file system also tracks that file system's info in **device directory** or **volume table of contents**
- ? As well as **general-purpose file systems** there are many **special-purpose file systems**, frequently all within the same operating system or computer

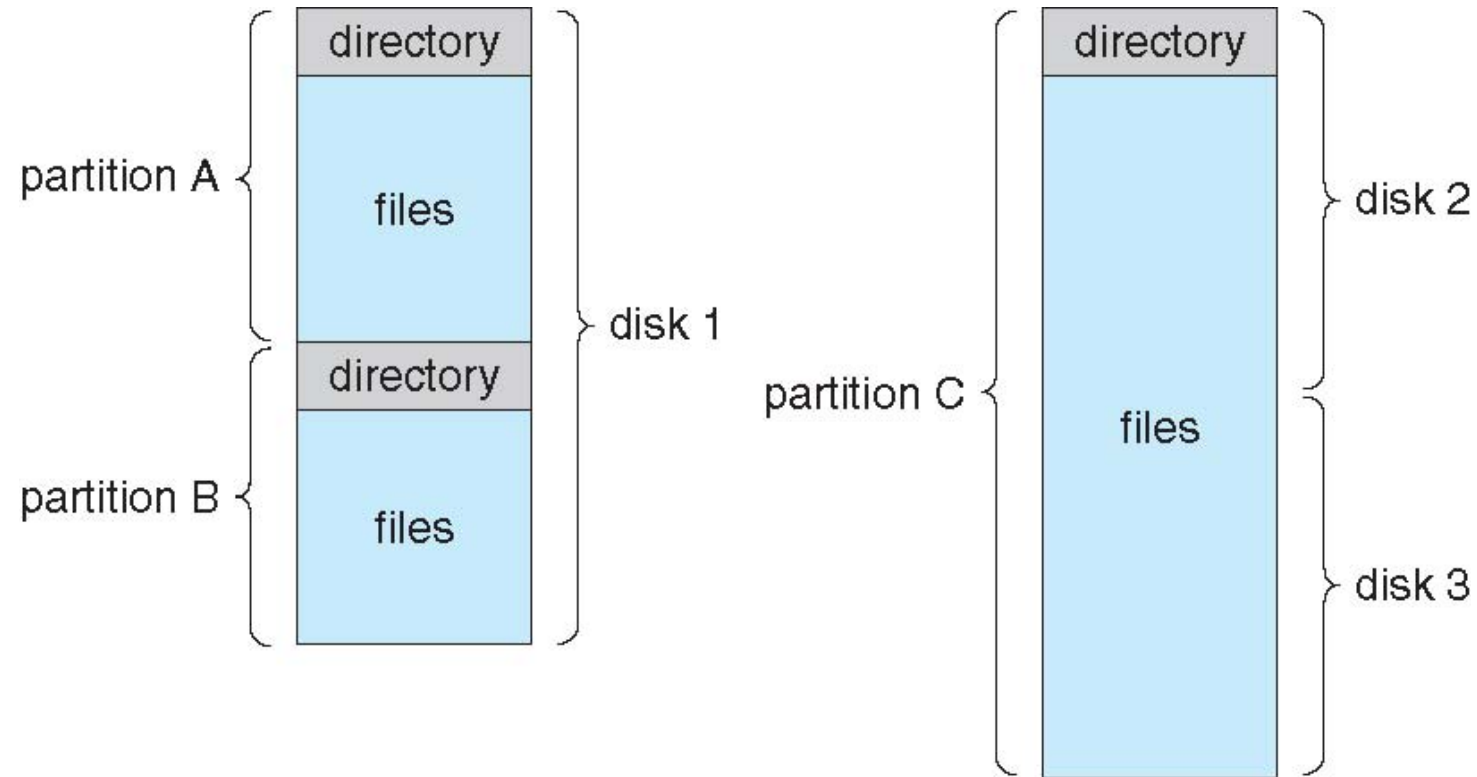


OPERATING SYSTEMS

A Typical File-system Organization



PES
UNIVERSITY
ONLINE



- ❑ We mostly talk of general-purpose file systems
- ❑ But systems frequently have many file systems, some general- and some special- purpose
- ❑ Consider Solaris has
 - ❑ tmpfs – memory-based volatile FS for fast, temporary I/O
 - ❑ objfs – interface into kernel memory to get kernel symbols for debugging
 - ❑ ctfs – contract file system for managing daemons
 - ❑ lofs – loopback file system allows one FS to be accessed in place of another
 - ❑ procfs – kernel interface to process structures
 - ❑ ufs, zfs – general purpose file systems

- ☐ Search for a file
- ☐ Create a file
- ☐ Delete a file
- ☐ List a directory
- ☐ Rename a file
- ☐ Traverse the file system

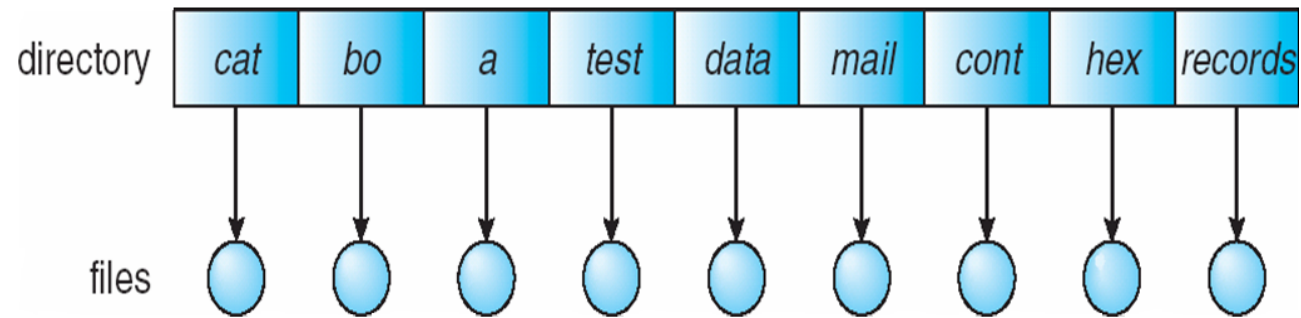
The directory is organized logically to obtain

- ❑ Efficiency – locating a file quickly
- ❑ Naming – convenient to users
 - ❑ Two users can have same name for different files
 - ❑ The same file can have several different names
- ❑ Grouping – logical grouping of files by properties, (e.g., all Java programs, all games, ...)

OPERATING SYSTEMS

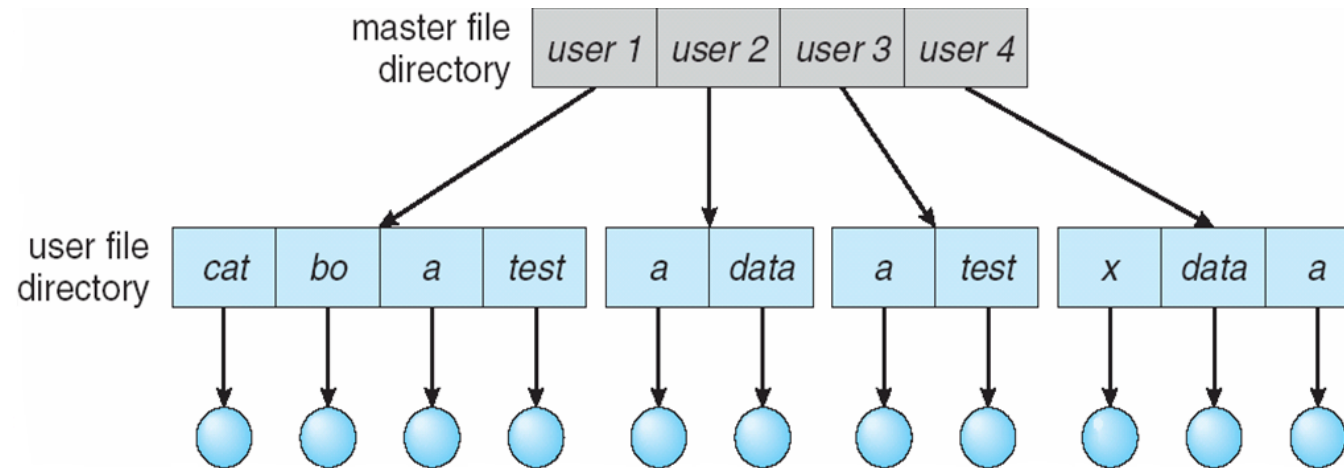
Single-Level Directory

- ? A single directory for all users



- ? Naming problem
- ? Grouping problem

? Separate directory for each user

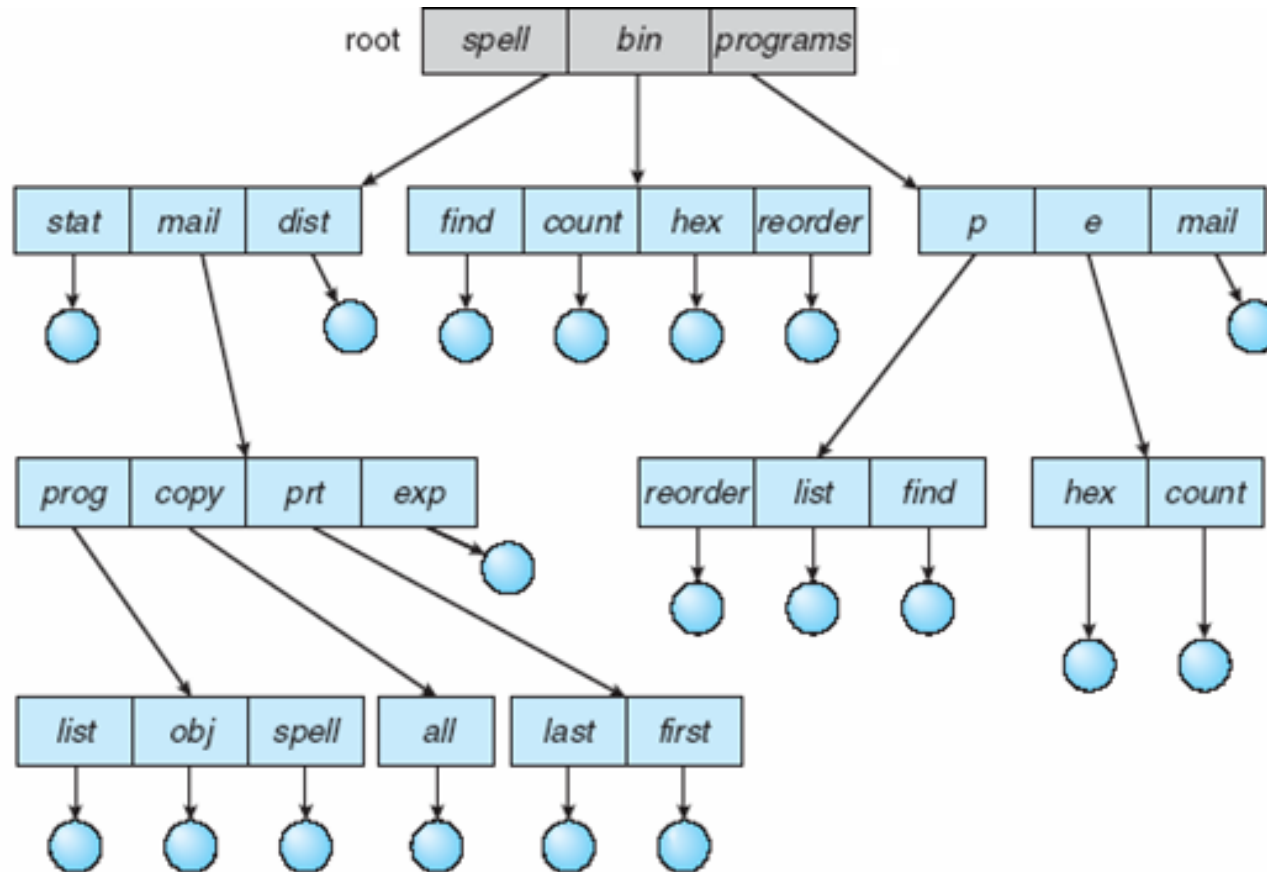


? Path name

? Can have the same file name for different user

? Efficient searching

? No grouping capability



- A tree structure is the most common directory structure.
- The tree has a root directory, and every file in the system have a unique path.

- ❑ Efficient searching
- ❑ Grouping Capability
- ❑ Current directory (working directory)
 - ❑ **cd /spell/mail/prog**
 - ❑ **type list**
- ❑ We cannot share files

? **Absolute** or **relative** path name

? Creating a new file is done in current directory

? Delete a file

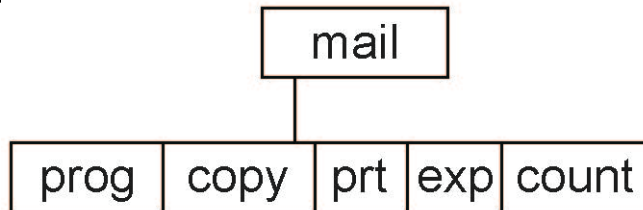
rm <file-name>

? Creating a new subdirectory is done in current directory

mkdir <dir-name>

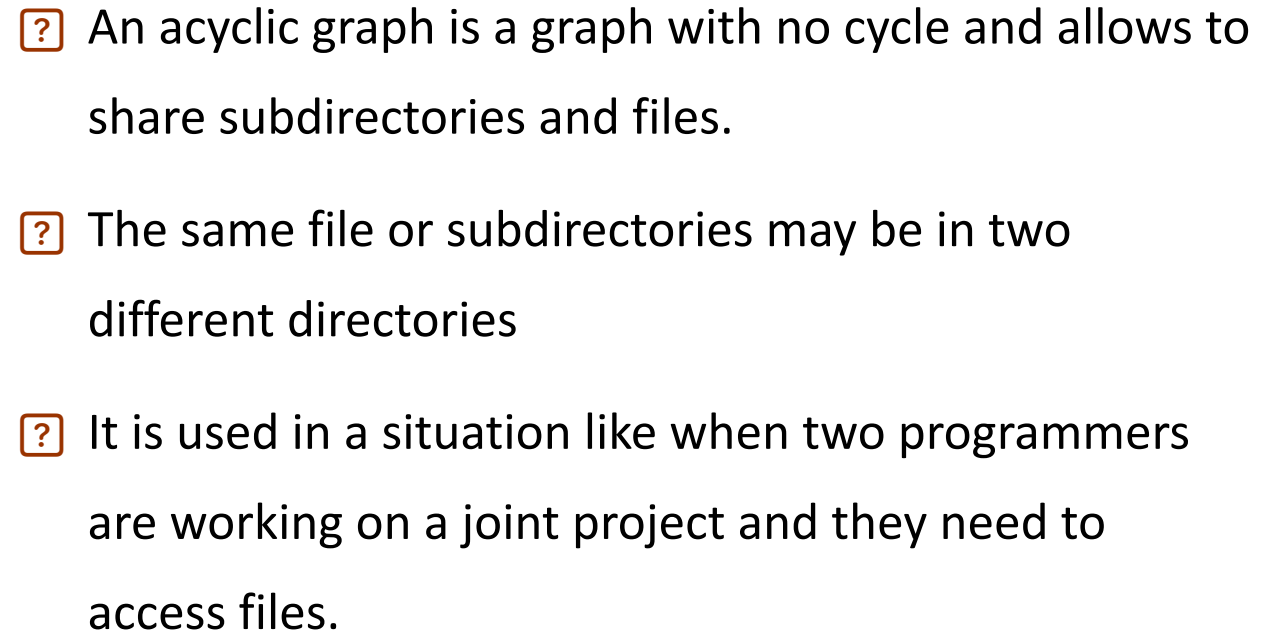
Example: if in current directory **/mail**

mkdir count



Deleting “mail” \Rightarrow deleting the entire subtree rooted by “mail”

Acyclic-Graph Directories



- ❓ An acyclic graph is a graph with no cycle and allows to share subdirectories and files.
- ❓ The same file or subdirectories may be in two different directories
- ❓ It is used in a situation like when two programmers are working on a joint project and they need to access files.

- ❑ Two different names (aliasing)
- ❑ If **dict** deletes **list** \Rightarrow dangling pointer

Solutions:

- ❑ Backpointers, so we can delete all pointers
Variable size records a problem
- ❑ Backpointers using a daisy chain organization
- ❑ Entry-hold-count solution
- ❑ New directory entry type
 - ❑ **Link** – another name (pointer) to an existing file
 - ❑ **Resolve the link** – follow pointer to locate the file

OPERATING SYSTEMS

General Graph Directory

Advantages:

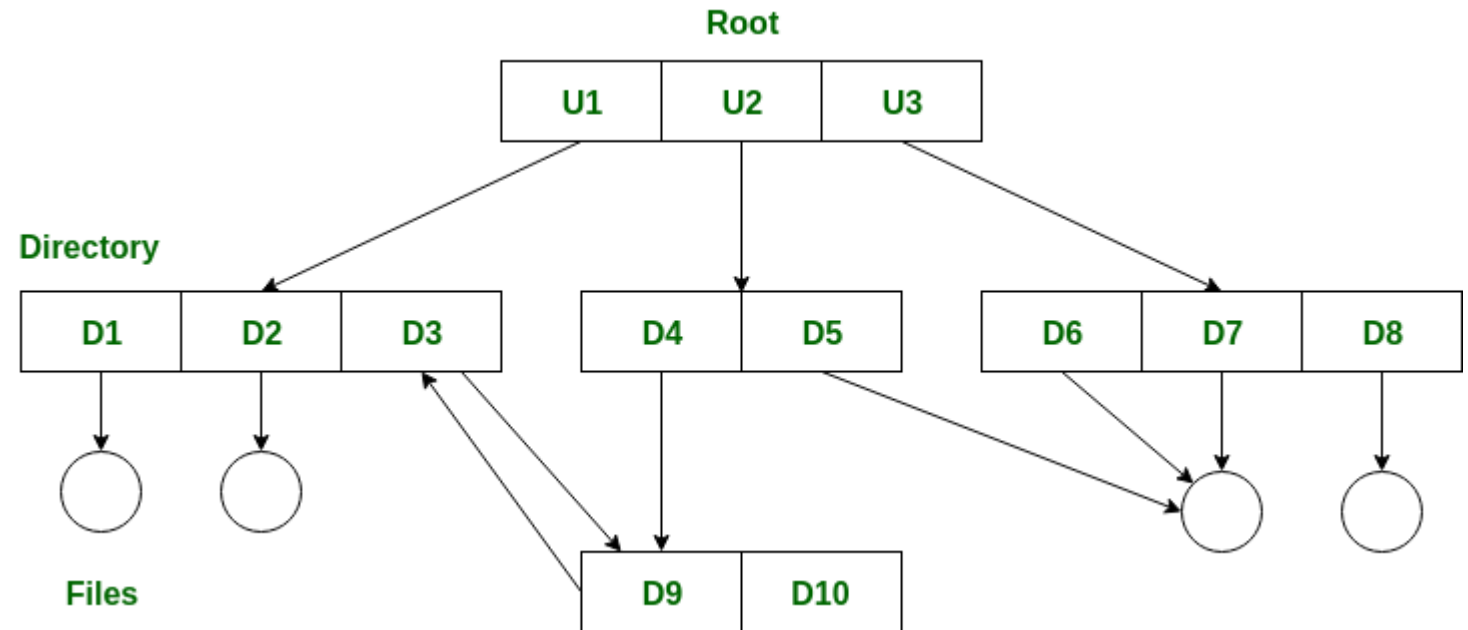
It allows cycles within a dir structure.

Multiple directories can be derived from more than one parent dir.

Disadvantages:

It is more costly than others.

It needs garbage collection (traversing the entire file system, marking everything that can be accessed)

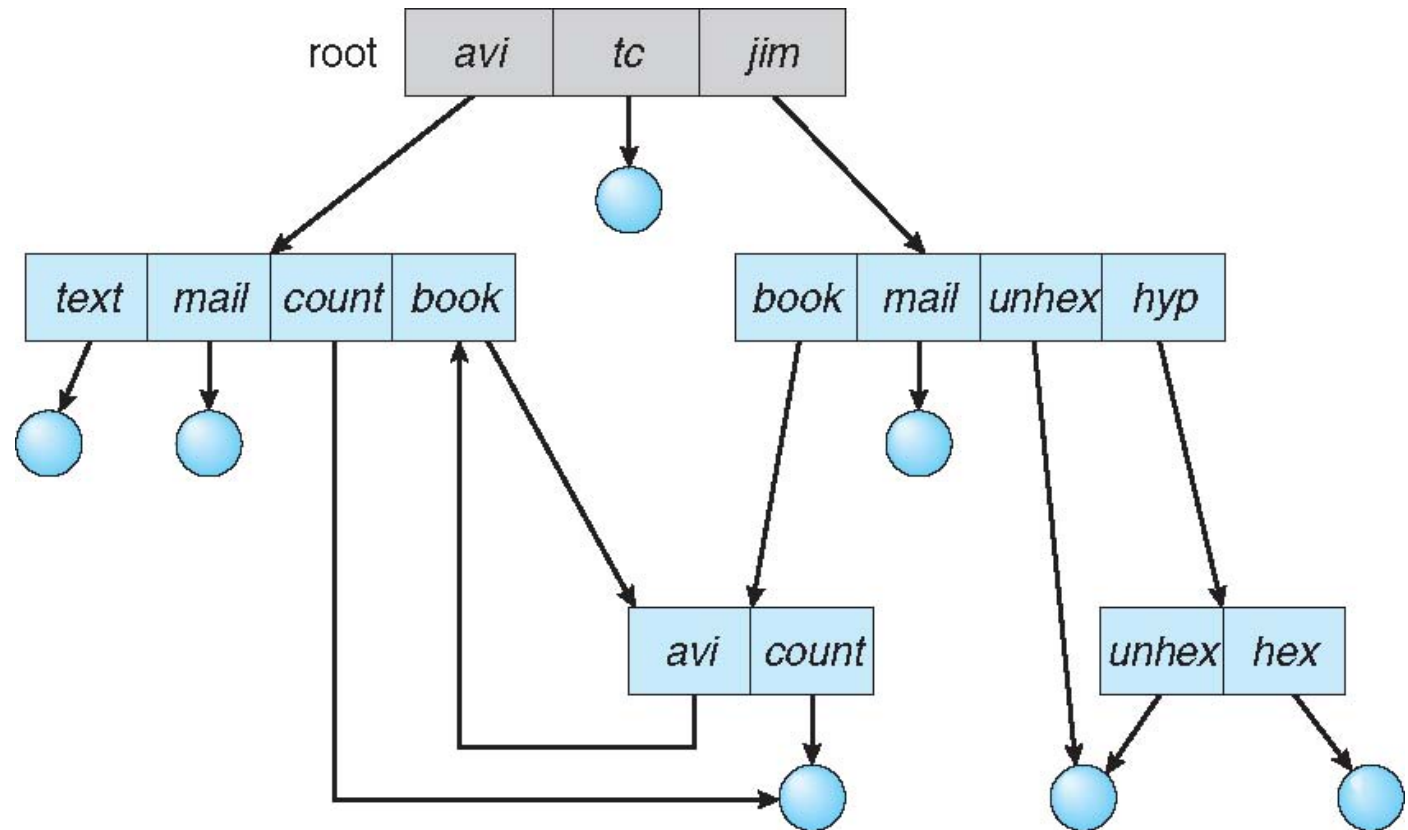


? How do we guarantee no cycles?

? Allow only links to file not subdirectories

? **Garbage collection**

? Every time a new link is added use a cycle detection algorithm to determine whether it is OK





THANK YOU

Suresh Jamadagni

Department of Computer Science Engineering

sureshjamadagni@pes.edu