

COP 4610

Operating System Principles

File Systems

1

File Concept

- Contiguous logical address space
- File = collection of related information recorded on secondary storage
- Types:
 - Data
 - Numeric (text, ASCII; "LINE_MAX" bytes)
 - Character (text, ASCII; "LINE_MAX" bytes)
 - Binary (executable, readable by computer)
 - Program

File "interpretation" is up to user/program

2

File Structure

- None - sequence of words, bytes
- Simple record structure
 - Lines
 - Fixed length
 - Variable length
- Complex Structures
 - Formatted document
- Who decides:
 - Operating system
 - Program

COP 4610 – Operating System Principles

3

3

Example Structures

CSV

```
Poellabauer,Christian,5,3.45
Doe,Jane,6,3.98
Bowyer,Kevin,4,4.25
```

XML

```
<Participant FN="Christian" LN="Poellabauer">
  <Publication Year="2008" />
  <Publication Year="2009" />
</Participant>
```

Structured

Poellabauer	Christian	cpoellab
Bualuan	Ramzi	rbualuan
Thain	Doug	dthain

```
struct ProfInfo
{
  char szLast[41];
  char szFirst[31];
  char szNetID[11];
};
```

COP 4610 – Operating System Principles

4

4

Attributes: File Control Block (FCB)

- **Name**
- **Identifier** – unique tag (number)
- **Type**
- **Location**
- **Size**
- **Protection (ACL)** – controls who can do reading, writing, executing
- **Time, date, and user identification** – data for protection, security, and usage monitoring

file permissions
file dates (create, access, write)
file owner, group, ACL
file size
file data blocks or pointers to file data blocks

5

File Types – Name, Extension

file type	usual extension	function
executable	exe, com, bin or none	ready-to-run machine-language program
object	obj, o	compiled, machine language, not linked
source code	c, cc, java, pas, asm, a	source code in various languages
batch	bat, sh	commands to the command interpreter
text	txt, doc	textual data, documents
word processor	wp, tex, rtf, doc	various word-processor formats
library	lib, a, so, dll	libraries of routines for programmers
print or view	ps, pdf, jpg	ASCII or binary file in a format for printing or viewing
archive	arc, zip, tar	related files grouped into one file, sometimes compressed, for archiving or storage
multimedia	mpeg, mov, rm, mp3, avi	binary file containing audio or A/V information

6

File Operations

- File is an **abstract data type**
- **Basic operations on files:**
 - Create
 - Write
 - Read
 - **Reposition within file**
 - Delete
 - Truncate
- *Open(F_i)* – search the directory structure on disk for entry F_i , and move the content of entry to memory
- *Close (F_i)* – move the content of entry F_i in memory back to directory structure on disk
- Opening files: “remember” used files; efficiency; convenience

COP 4610 – Operating System Principles

7

7

Open File Locking

- Provided by some operating systems and file systems (*flock* and *fcntl* system calls)
- Mediates access to a file
- Mandatory or advisory:
 - **Mandatory** – access is denied depending on locks held and requested
 - **Advisory** – processes can find status of locks and decide what to do

COP 4610 – Operating System Principles

8

8

Access Methods

- **Sequential Access**

read next
write next
reset
skip forward

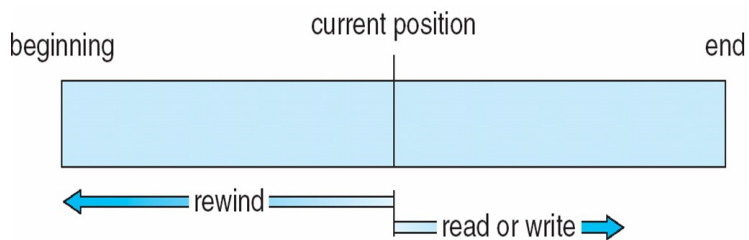
- **Direct Access**

read n
write n
position to n
read next
write next
rewrite n

n = relative block number

9

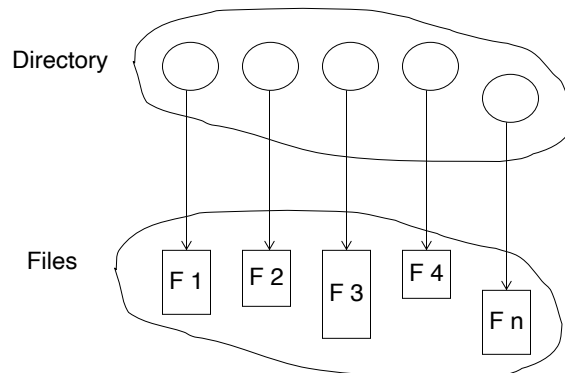
Sequential-Access File



10

Directory Structure

- A collection of nodes containing information about all files



Both the directory structure and the files reside on disk

COP 4610 – Operating System Principles

11

11

Disk Structure

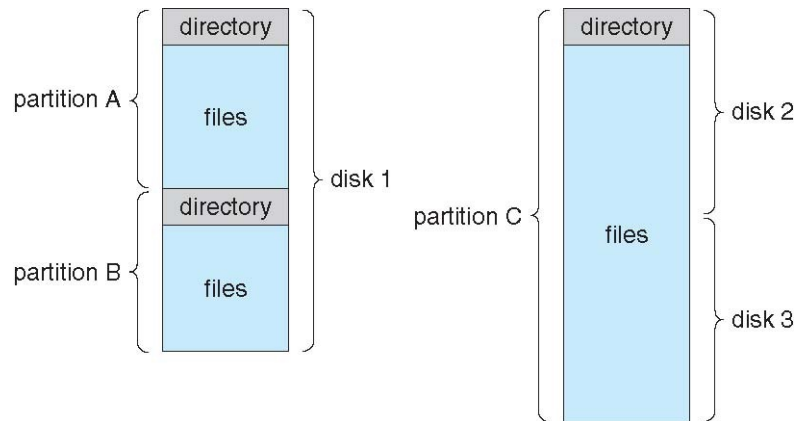
- Disk can be subdivided into **partitions**
- Disk or partition can be used **raw** – without a file system, or **formatted** with a file system
- Entity containing file system known as a **volume**
- Each volume containing file system also tracks that file system's info in **device directory** or **volume table of contents**
- In addition to **general-purpose file systems**, there can be many **special-purpose file systems**

COP 4610 – Operating System Principles

12

12

A Typical File-system Organization



COP 4610 – Operating System Principles

13

13

Operations Performed on Directory

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

COP 4610 – Operating System Principles

14

14

Organize the Directory (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

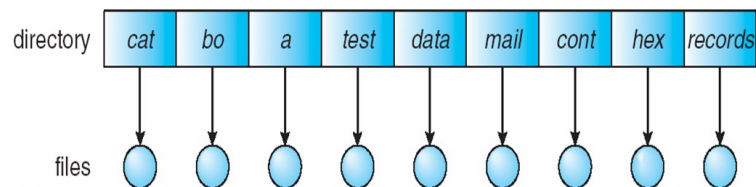
COP 4610 – Operating System Principles

15

15

Single-Level Directory

- A single directory for all users



Naming problem

Grouping problem

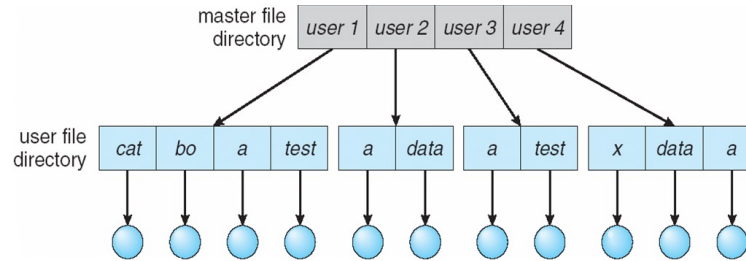
COP 4610 – Operating System Principles

16

16

Two-Level Directory

- Separate directory for each user



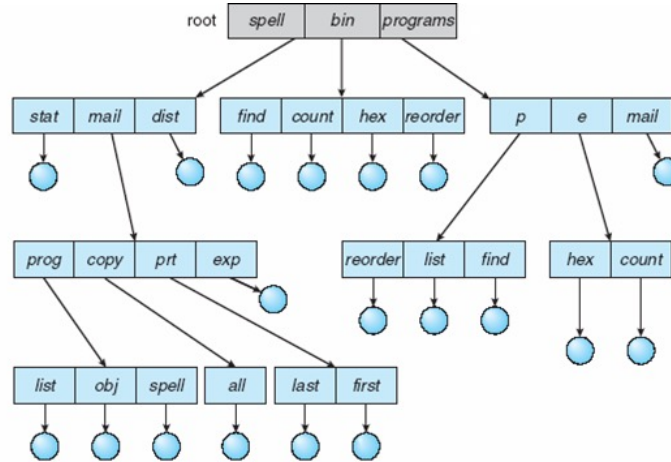
- Path name
- Can have the same file name for different user
- Efficient searching
- No grouping capability

COP 4610 – Operating System Principles

17

17

Tree-Structured Directories



COP 4610 – Operating System Principles

18

18

Tree-Structured Directories (Cont.)

- Efficient searching
- Grouping Capability
- Current directory (“working directory”)
 - cd /spell/mail/prog
 - cd ~
 - cd .
 - cd ..

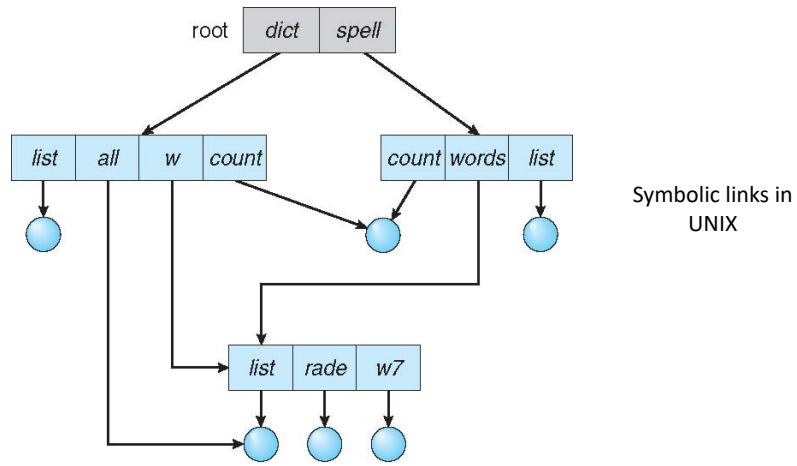
19

Tree-Structured Directories (Cont)

- **Absolute** or **relative** path name
 - ./mydir/myfile
 - /usr/cpoellab/mydir/myfile
- Creating a new file is done in current directory
- Delete a file
 - rm myfile
- Creating a new subdirectory is done in current directory
 - mkdir newdir

20

Acyclic-Graph Directories

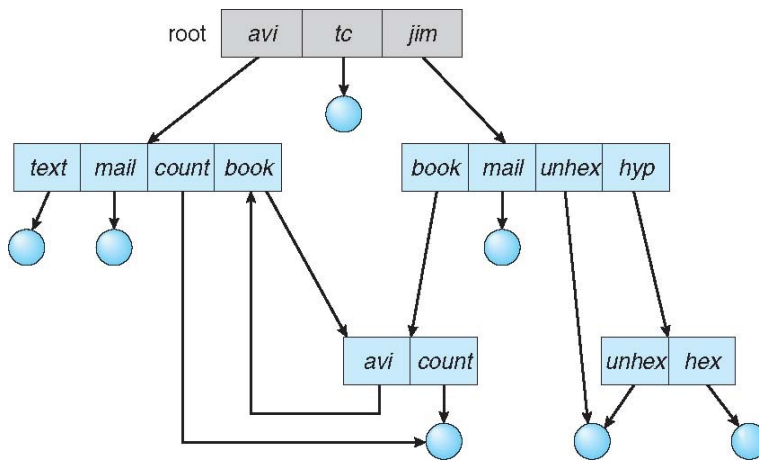


COP 4610 – Operating System Principles

21

21

Acyclic-Graph Directories



COP 4610 – Operating System Principles

22

22

General Graph Directory (Cont.)

- How do we guarantee no cycles?
 - Allow only links to files, not subdirectories
 - Every time a new link is added use a cycle detection algorithm to determine whether it is OK
 - Ignore links for activities such as recursive search/delete/etc.

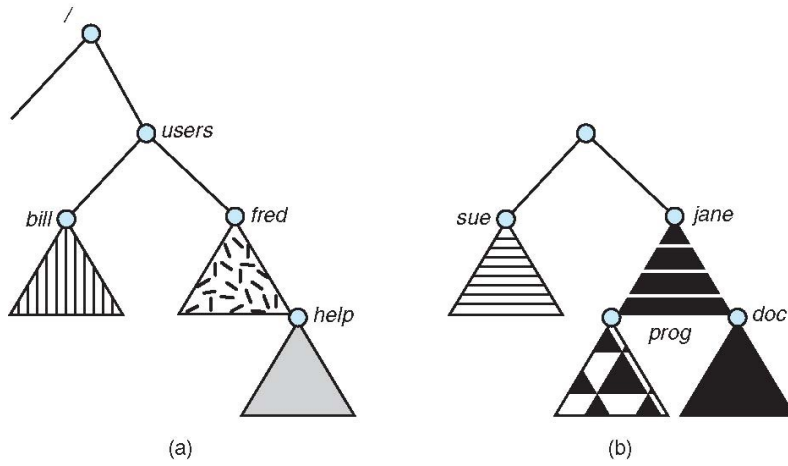
23

File System Mounting

- A file system must be **mounted** before it can be accessed
- A unmounted file system is mounted at a **mount point**

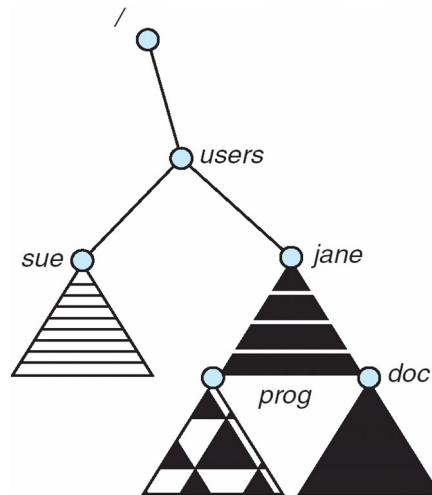
24

(a) Existing (b) Unmounted Partition



25

Mount Point



26

File Sharing

- Sharing of files on multi-user systems is desirable
- Sharing may be done through a **protection** scheme
- On distributed systems, files may be shared across a network
- Network File System (NFS) is a common distributed file-sharing method

COP 4610 – Operating System Principles

27

27

File Sharing – Multiple Users

- **User IDs** identify users, allowing permissions and protections to be per-user
- **Group IDs** allow users to be in groups, permitting group access rights

COP 4610 – Operating System Principles

28

28

Protection

- File owner/creator should be able to control:
 - what can be done
 - by whom
- Types of access
 - **Read**
 - **Write**
 - **Execute**
 - **Append**
 - **Delete**
 - **List**

29

Access Lists and Groups

- Mode of access: read, write, execute
- Three classes of users

a) owner access	7	⇒	RWX 1 1 1
b) group access	6	⇒	RWX 1 1 0
c) public access	1	⇒	RWX 0 0 1

chmod 761 file
chgrp G file

30

A Sample UNIX Directory Listing

```

-rw-rw-r-- 1 pbq staff 31200 Sep 3 08:30 intro.ps
drwx----- 5 pbq staff 512 Jul 8 09:33 private/
drwxrwxr-x 2 pbq staff 512 Jul 8 09:35 doc/
drwxrwx--- 2 pbq student 512 Aug 3 14:13 student-proj/
-rw-r--r-- 1 pbq staff 9423 Feb 24 2003 program.c
-rwxr-xr-x 1 pbq staff 20471 Feb 24 2003 program
drwx--x--x 4 pbq faculty 512 Jul 31 10:31 lib/
drwx----- 3 pbq staff 1024 Aug 29 06:52 mail/
drwxrwxrwx 3 pbq staff 512 Jul 8 09:35 test/

```