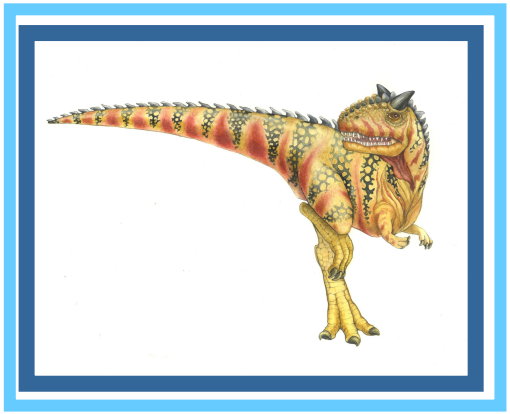
Chapter 10: File-System Interface



Chapter 10: File-System Interface

- File Concept
- Access Methods
- Directory Structure
- File-System Mounting
- File Sharing
- Protection

Objectives

- To explain the function of file systems
- To describe the interfaces to file systems
- To discuss file-system design tradeoffs, including access methods, file sharing, file locking, and directory structures
- To explore file-system protection

File Concept

- Contiguous logical address space
- Types:
  - Data
    - numeric
    - character
    - binary
  - Program

File Structure

- None - sequence of words, bytes
- Simple record structure
  - Lines
  - Fixed length
  - Variable length
- Complex Structures
  - Formatted document
  - Relocatable load file
- Can simulate last two with first method by inserting appropriate control characters
- Who decides:
  - Operating system
  - Program

File Attributes

- **Name** – only information kept in human-readable form
- **Identifier** – unique tag (number) identifies file within file system
- **Type** – needed for systems that support different types
- **Location** – pointer to file location on device
- **Size** – current file size
- **Protection** – controls who can do reading, writing, executing
- **Time, date, and user identification** – data for protection, security, and usage monitoring
- Information about files are kept in the directory structure, which is maintained on the disk

File Operations

- File is an **abstract data type**
- **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 to directory structure on disk

Open Files

- Several pieces of data are needed to manage open files:
  - File pointer: pointer to last read/write location, per process that has the file open
  - File-open count: counter of number of times a file is open – to allow removal of data from open-file table when last processes closes it
  - Disk location of the file: cache of data access information
  - Access rights: per-process access mode information

Open File Locking

- Provided by some operating systems and file systems
- 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

File Types – Name, Extension

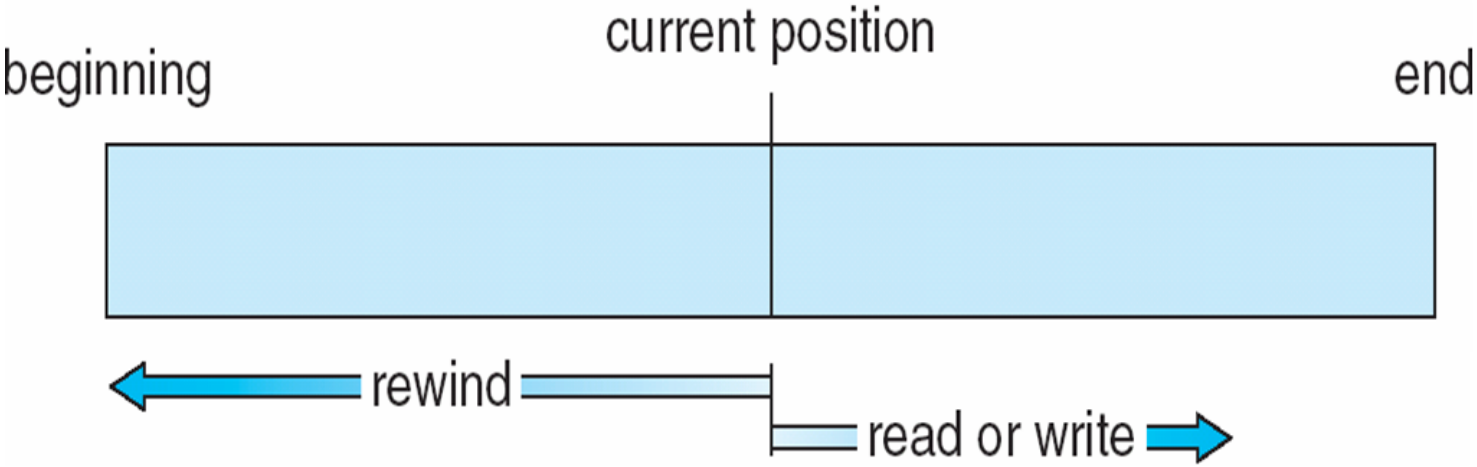
file type	usual extension	function
executable	exe, com, bin or none	ready-to-run machine-language program
object	obj, o	compiled, machine language, not linked
source code	c, cc, java, pas, asm, a	source code in various languages
batch	bat, sh	commands to the command interpreter
text	txt, doc	textual 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

Access Methods

- **Sequential Access**
  - read next
  - write next
  - reset
  - no read after last write
    - (rewrite)
- **Direct Access**
  - read *n*
  - write *n*
  - position to *n*
    - read next
    - write next
  - rewrite *n*

n = relative block number

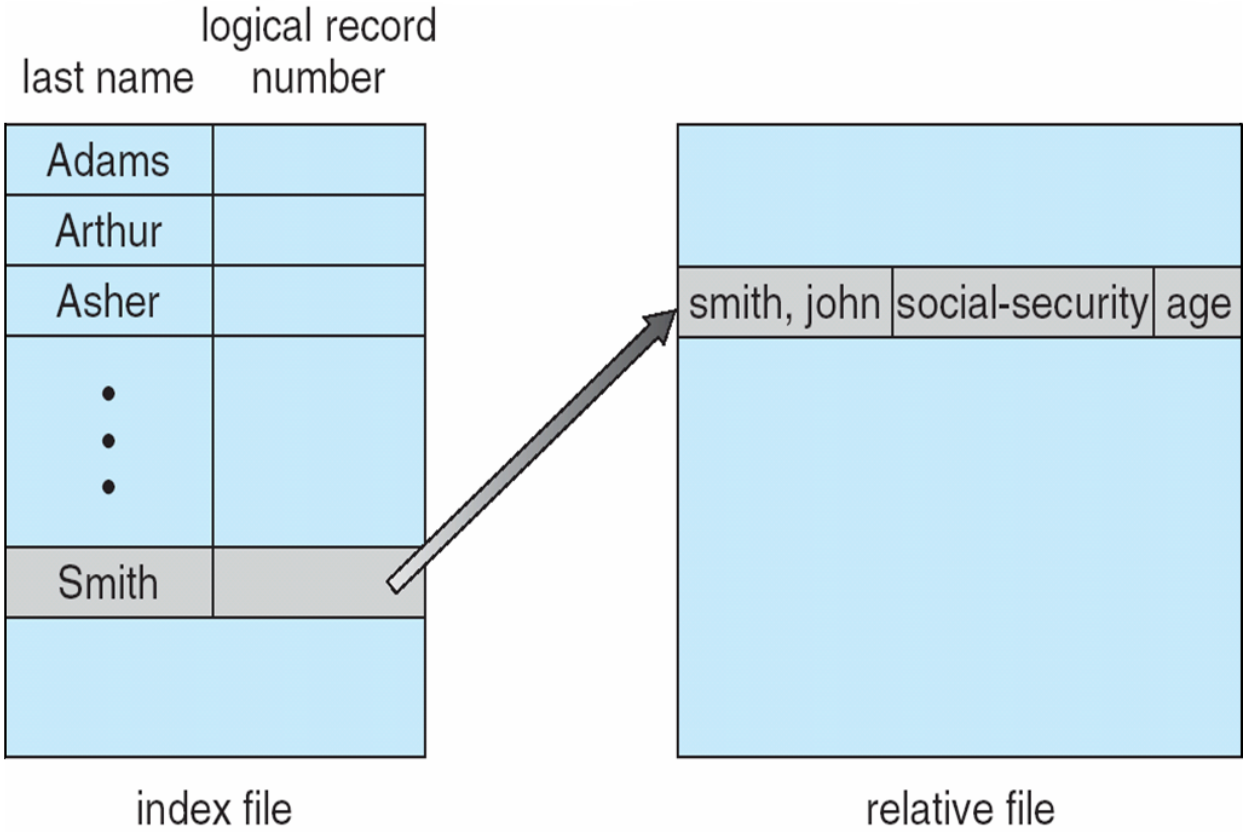
Sequential-access File



Simulation of Sequential Access on Direct-access File

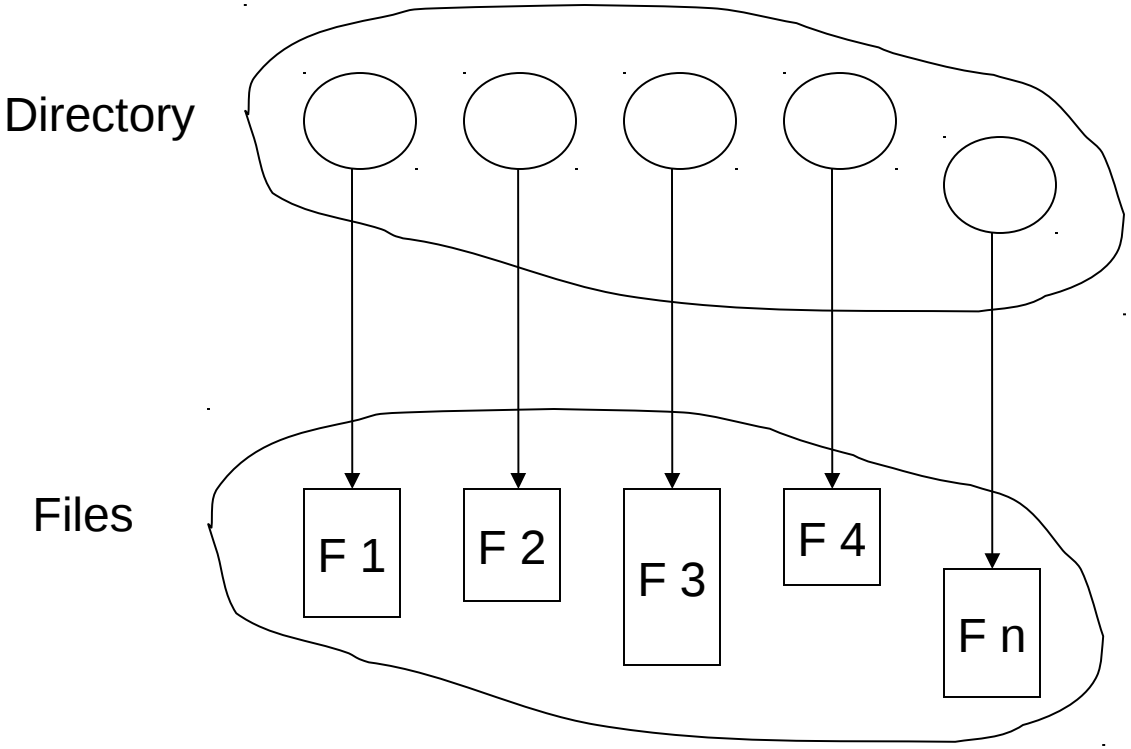
sequential access	implementation for direct access
reset	*cp* = 0;
read next	*read cp*; *cp* = *cp* + 1;
write next	*write cp*; *cp* = *cp* + 1;

Example of Index and Relative Files



Directory Structure

- A collection of nodes containing information about all files

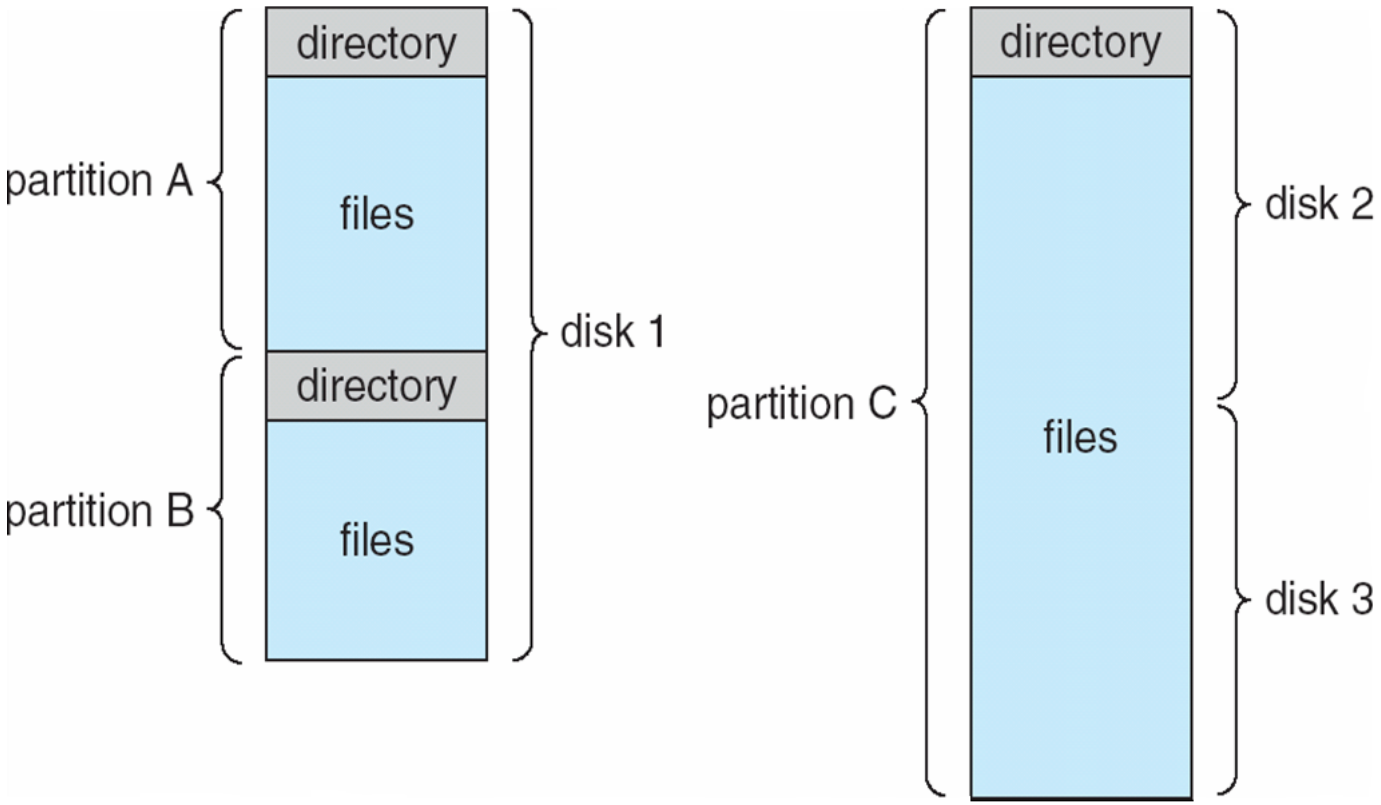


Both the directory structure and the files reside on disk
Backups of these two structures are kept on tapes

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

A Typical File-system Organization



Operations Performed on Directory

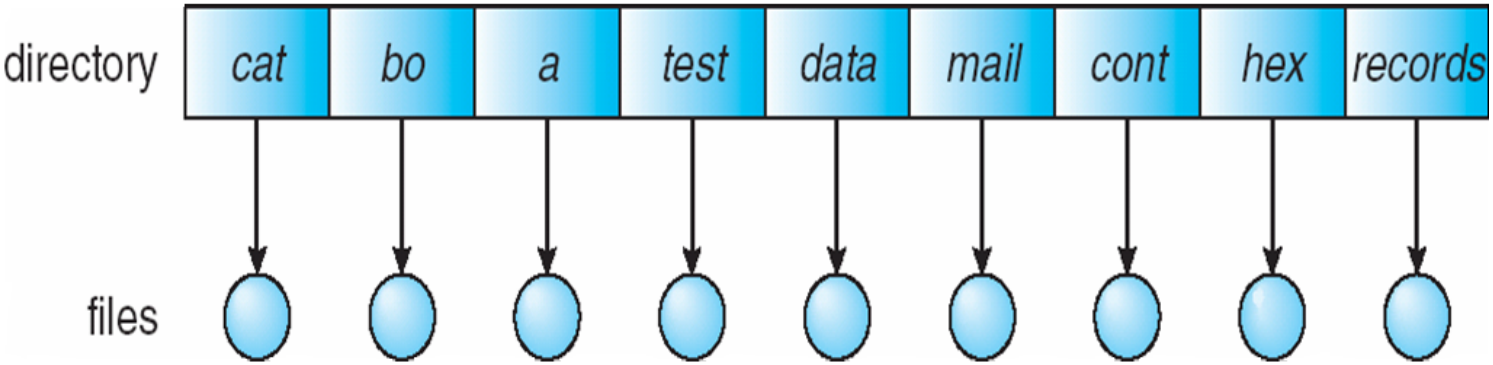
- Search for a file
- Create a file
- Delete a file
- List a directory
- Rename a file
- Traverse the file system

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, (e.g., all Java programs, all games, …)

Single-Level Directory

- A single directory for all users

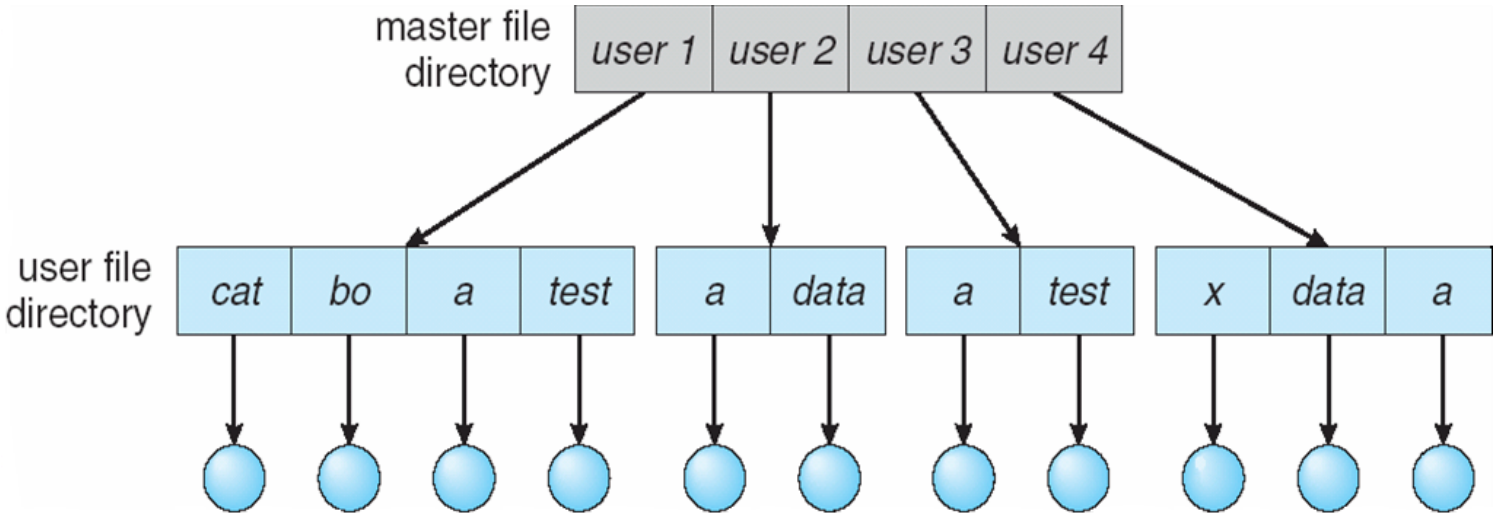


Naming problem

Grouping problem

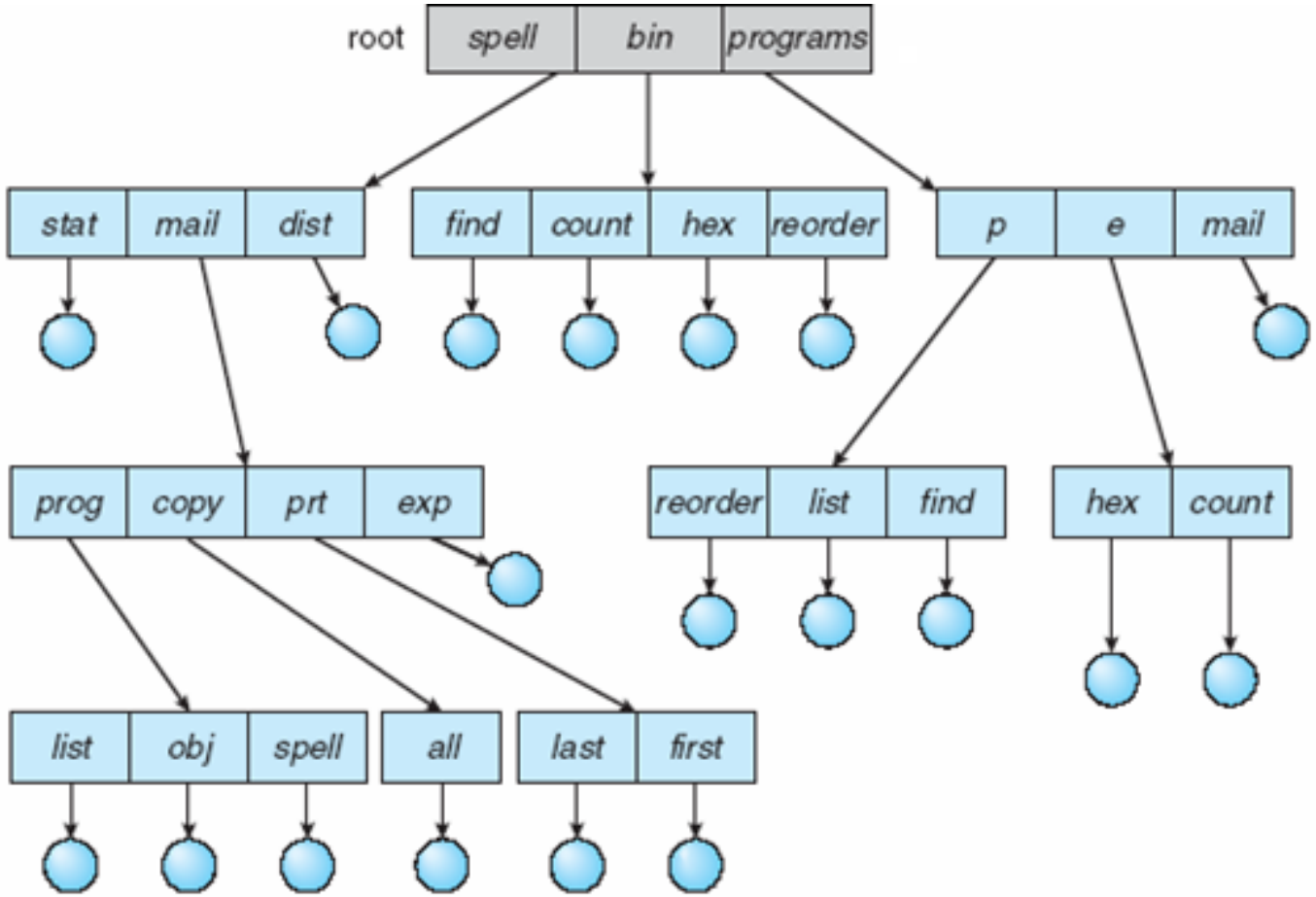
Two-Level Directory

- Separate directory for each user



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

Tree-Structured Directories



Tree-Structured Directories (Cont)

- Efficient searching
- Grouping Capability
- Current directory (working directory)
  - cd /spell/mail/prog
  - type list

Tree-Structured Directories (Cont)

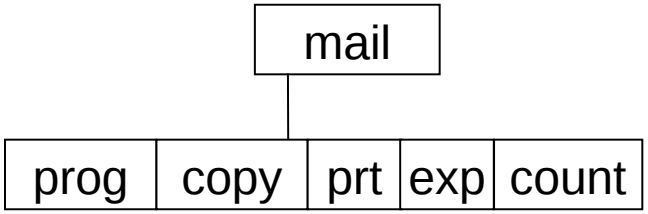
- **Absolute** or **relative** path name
- Creating a new file is done in current directory
- Delete a file

  rm <file-name>
- Creating a new subdirectory is done in current directory

  mkdir <dir-name>

  Example: if in current directory /mail

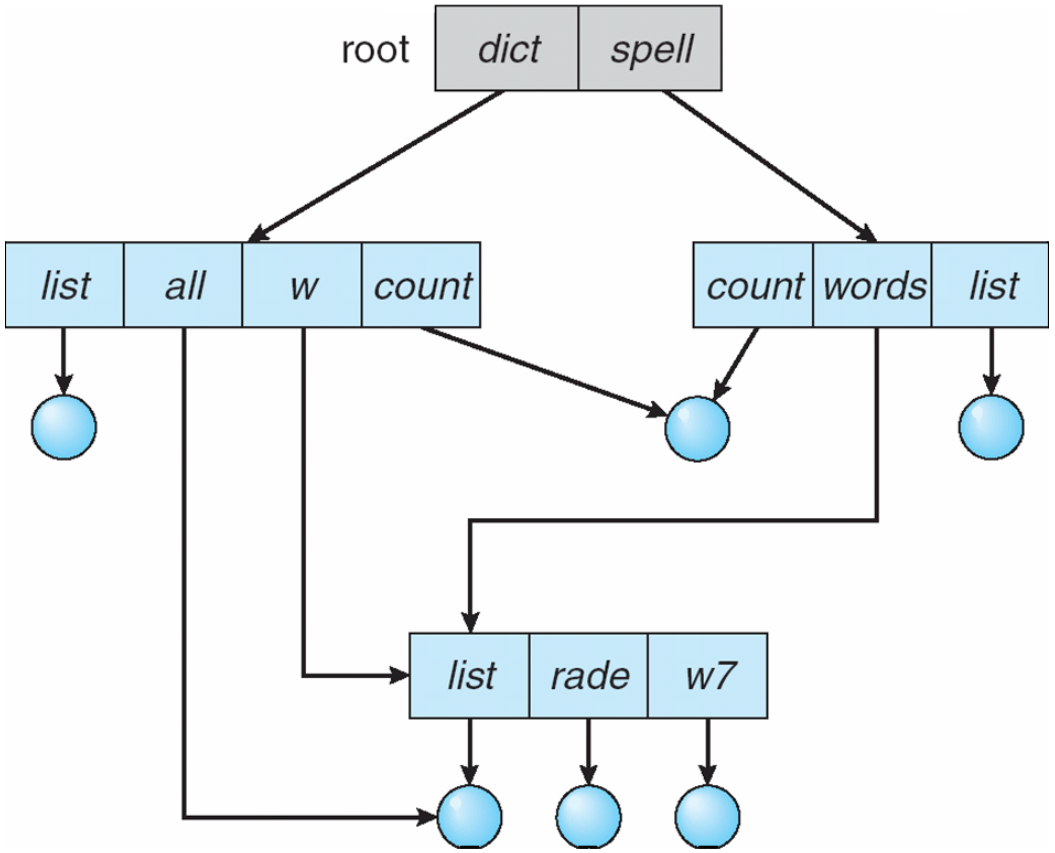
  mkdir count



Deleting “mail” ⇒ deleting the entire subtree rooted by “mail”

Acyclic-Graph Directories

- Have shared subdirectories and files

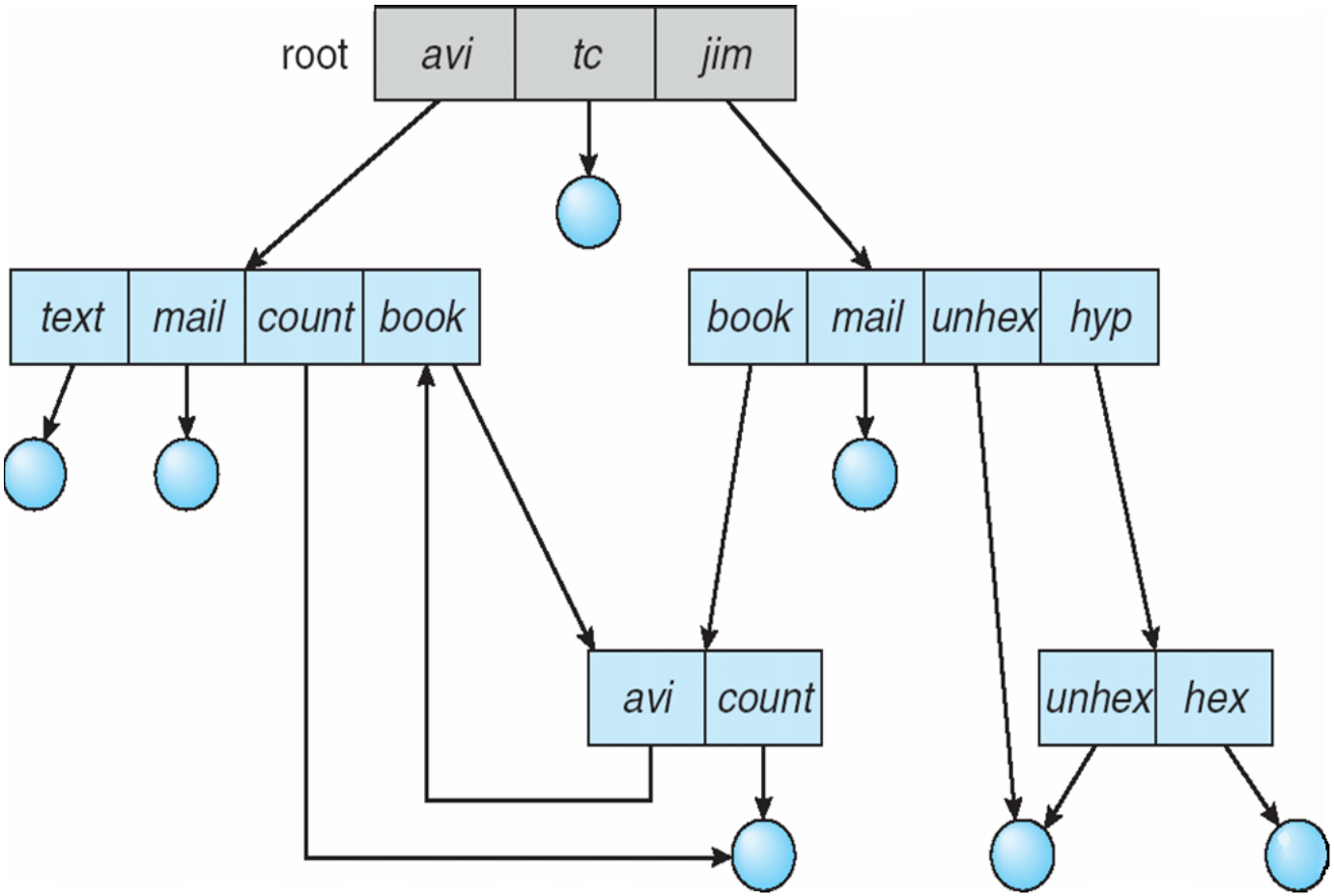


Acyclic-Graph Directories (Cont.)

- Two different names (aliasing)
- If *dict* deletes *list* ⇒ 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

General Graph Directory



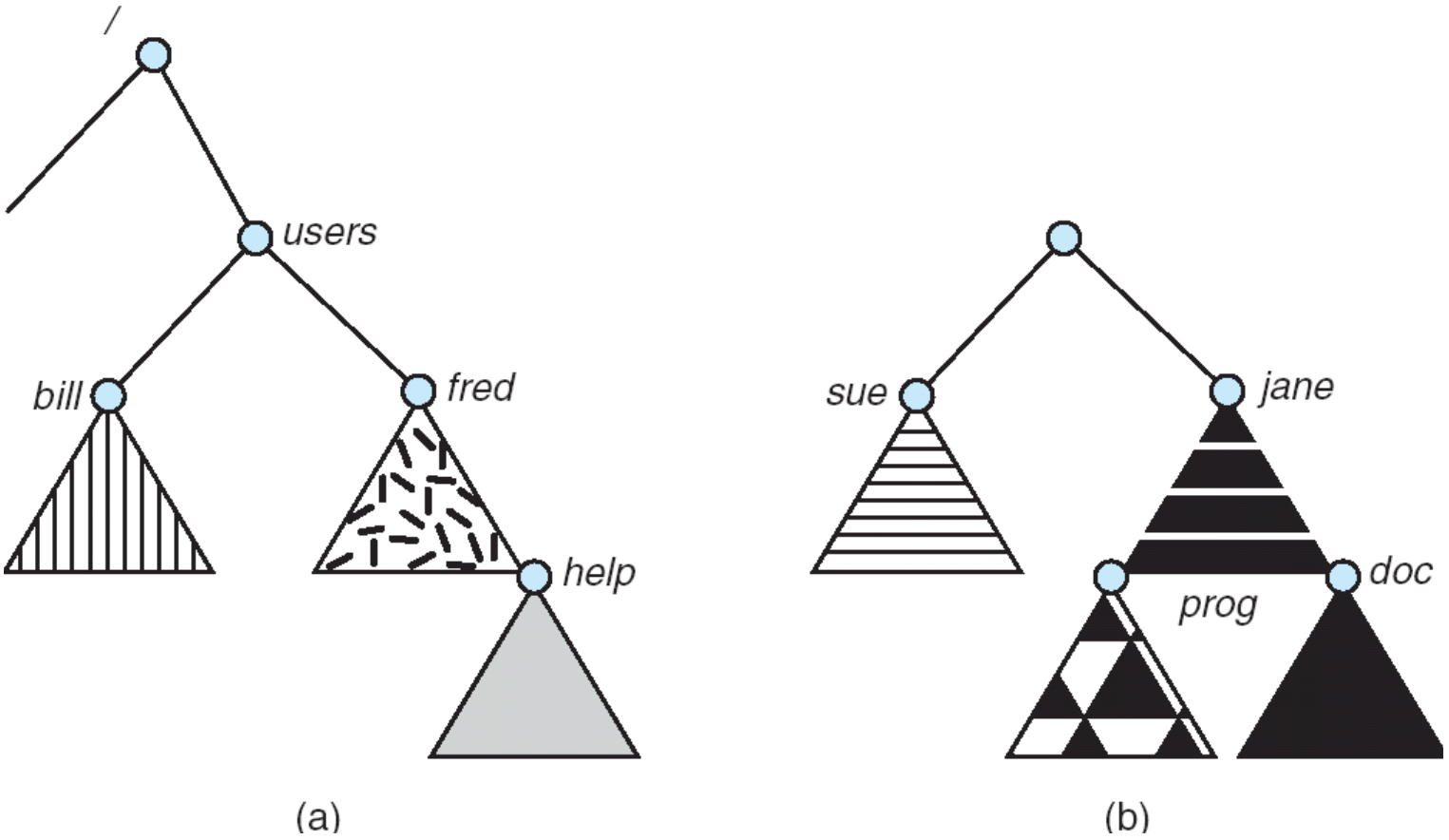
General Graph Directory (Cont.)

- 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

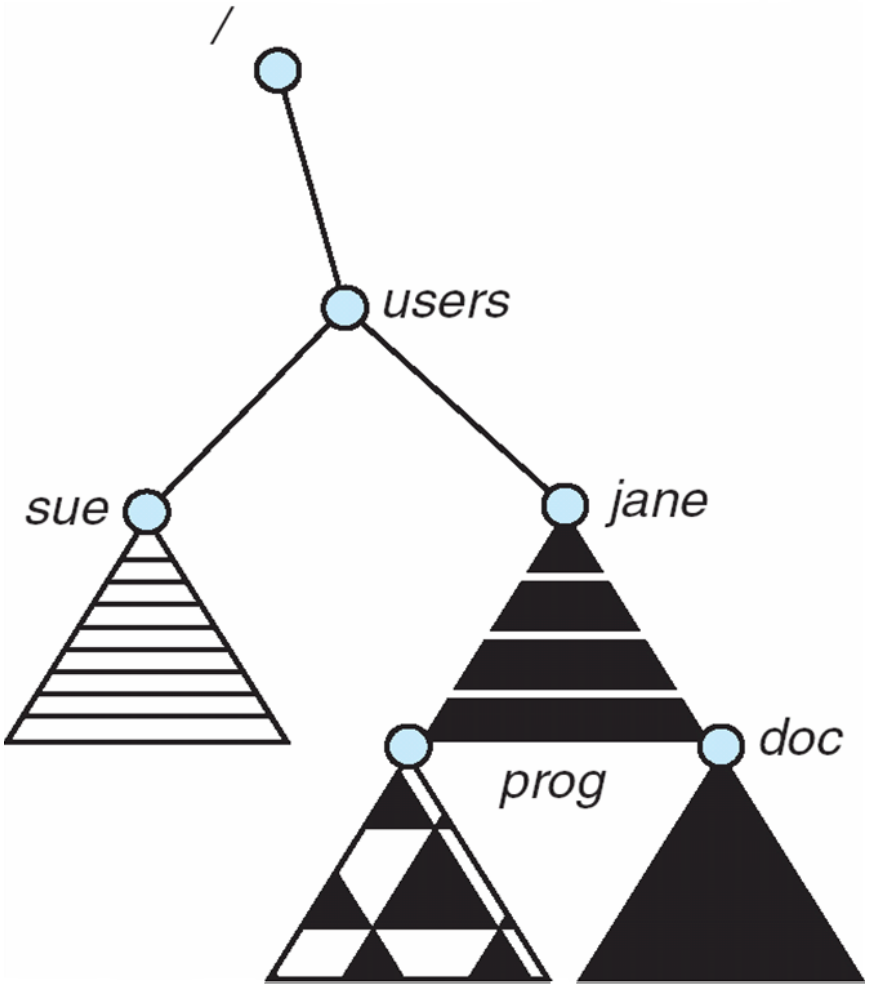
File System Mounting

- A file system must be **mounted** before it can be accessed
- A unmounted file system (i.e. Fig. 11-11(b)) is mounted at a **mount point**

(a) Existing. (b) Unmounted Partition



Mount Point



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

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

File Sharing – Remote File Systems

- Uses networking to allow file system access between systems
  - Manually via programs like FTP
  - Automatically, seamlessly using **distributed file systems**
  - Semi automatically via the **world wide web**
- **Client-server** model allows clients to mount remote file systems from servers
  - Server can serve multiple clients
  - Client and user-on-client identification is insecure or complicated
  - **NFS** is standard UNIX client-server file sharing protocol
  - **CIFS** is standard Windows protocol
  - Standard operating system file calls are translated into remote calls
- Distributed Information Systems **(distributed naming services)** such as LDAP, DNS, NIS, Active Directory implement unified access to information needed for remote computing

File Sharing – Failure Modes

- Remote file systems add new failure modes, due to network failure, server failure
- Recovery from failure can involve state information about status of each remote request
- Stateless protocols such as NFS include all information in each request, allowing easy recovery but less security

File Sharing – Consistency Semantics

- **Consistency semantics** specify how multiple users are to access a shared file simultaneously
  - The problem here is smilar to process synchronization of Chap 5
    - Acceptable solutions tend to be less complex due to disk I/O and network latency (for remote file systems
  - Andrew File System (AFS) implemented complex remote file sharing semantics
  - Unix file system (UFS) implements:
    - Writes to an open file become visible immediately (after save) to other users of the same open file
    - Users can elect to be notified when the file has been changed externally
    - Allows for sharing file pointer in order to allow multiple users to read and write simultaneously
  - AFS has **session semantics**
    - Writes only visible to sessions starting after the file is saved and closed

Protection

The remainder of this deck is optional for all students

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

Access Lists and Groups

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

  RWX

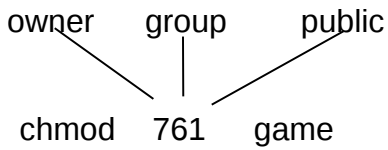
  a) **owner access** 7 ⇒ 1 1 1

  RWX

  b) **group access** 6 ⇒ 1 1 0

  RWX

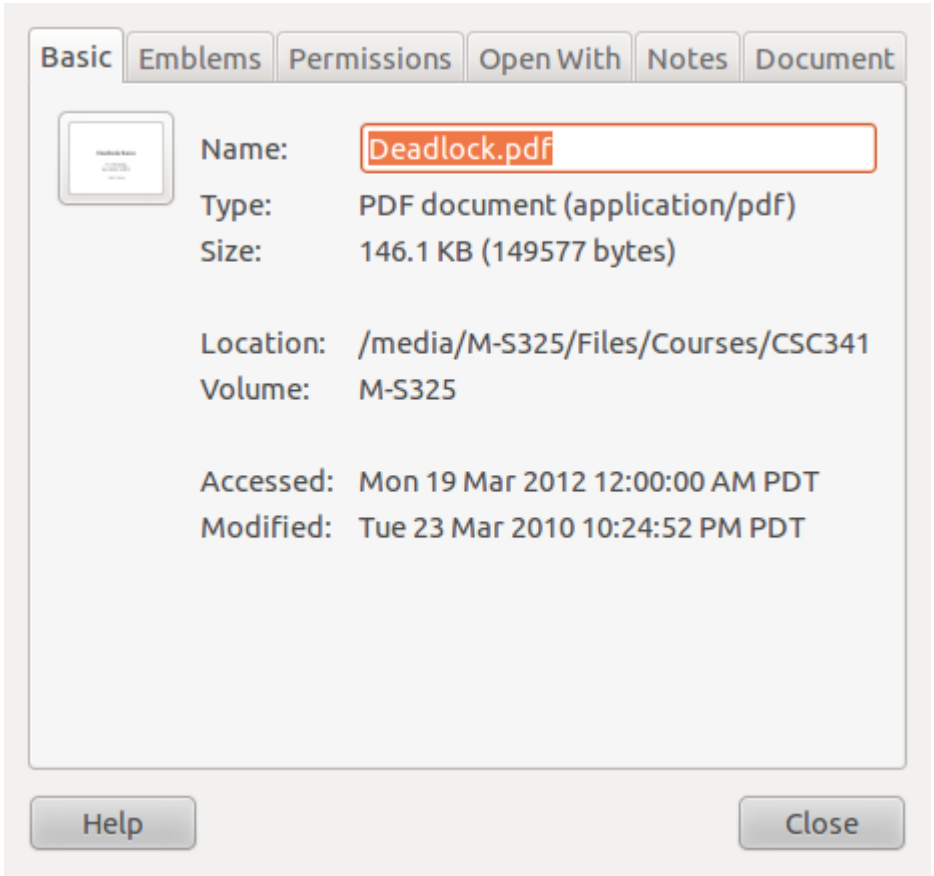
  c) **public access** 1 ⇒ 0 0 1
- Ask manager to create a group (unique name), say G, and add some users to the group.
- For a particular file (say *game*) or subdirectory, define an appropriate access.



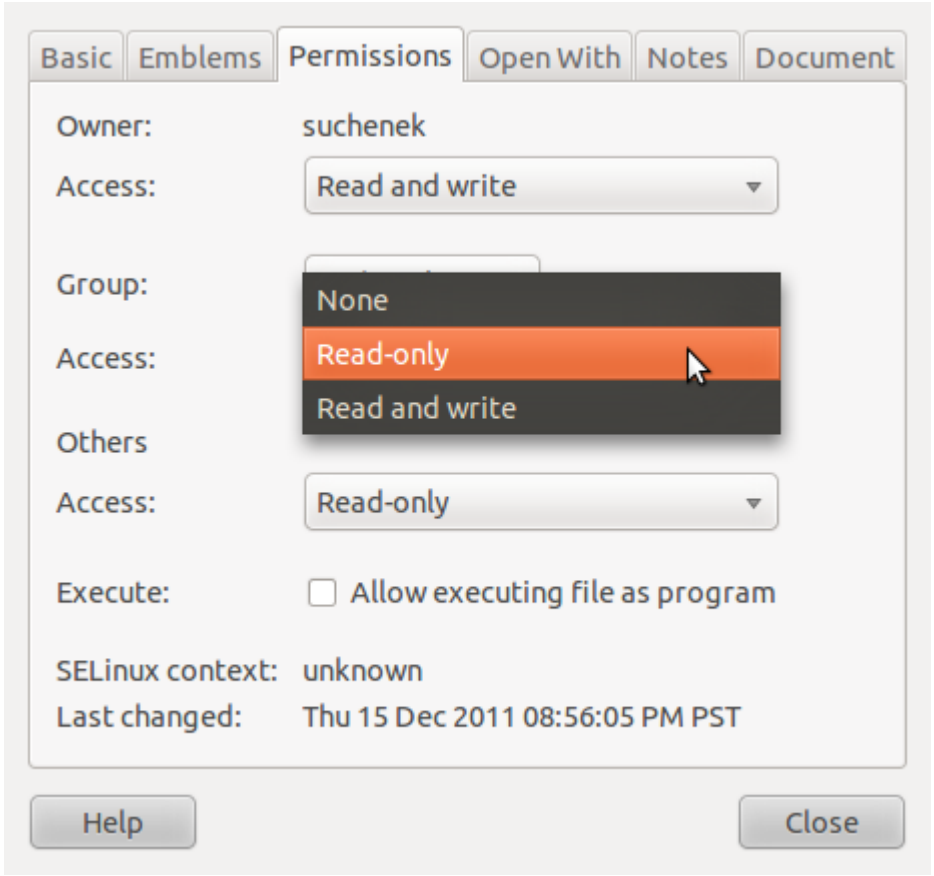
Attach a group to a file

chgrp G game

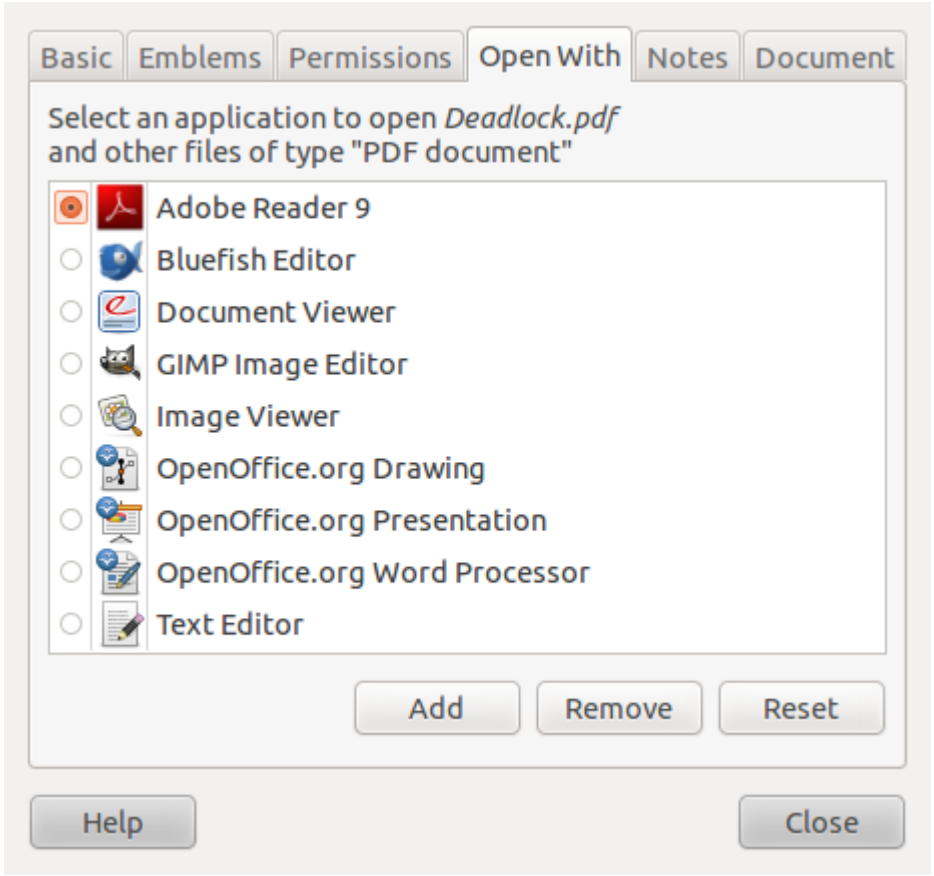
LINUX Access-control List Management



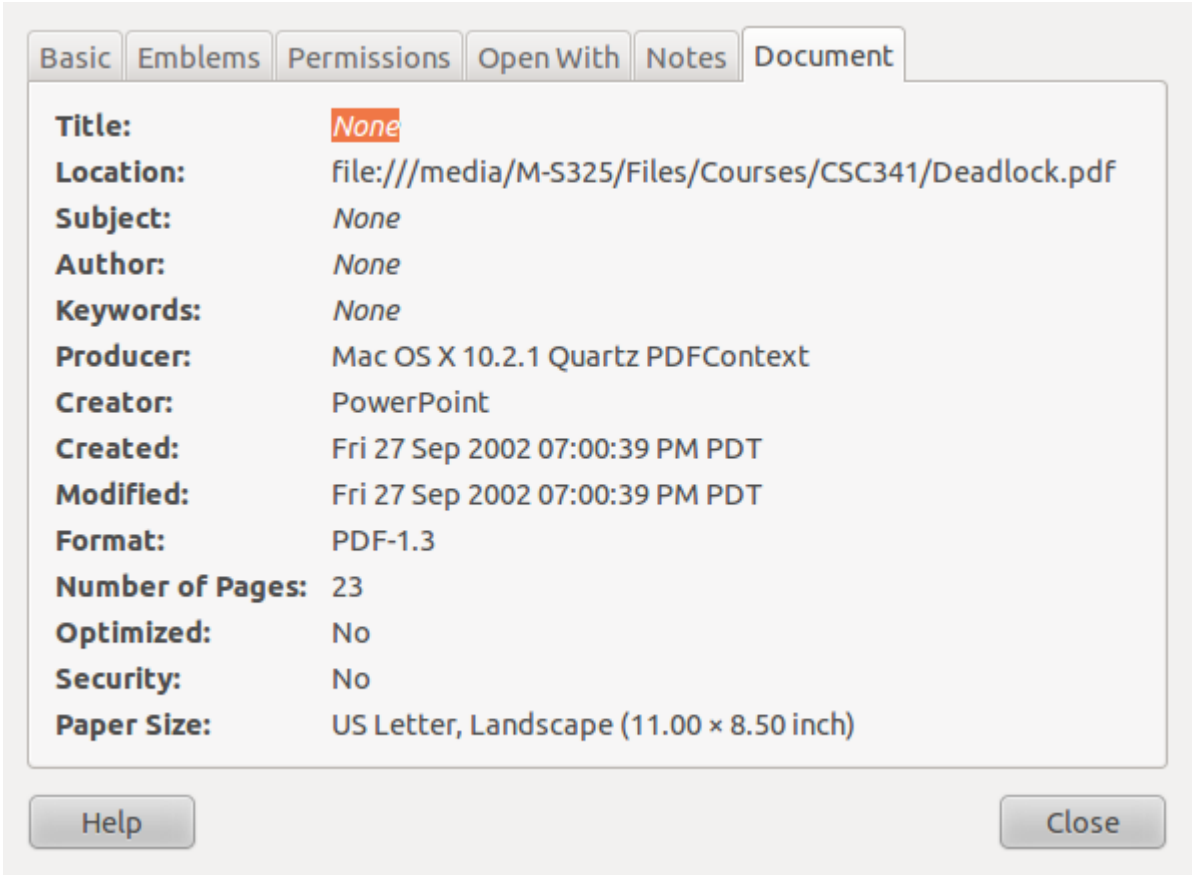
LINUX Access-control List Management



LINUX Access-control List Management



LINUX Access-control List Management



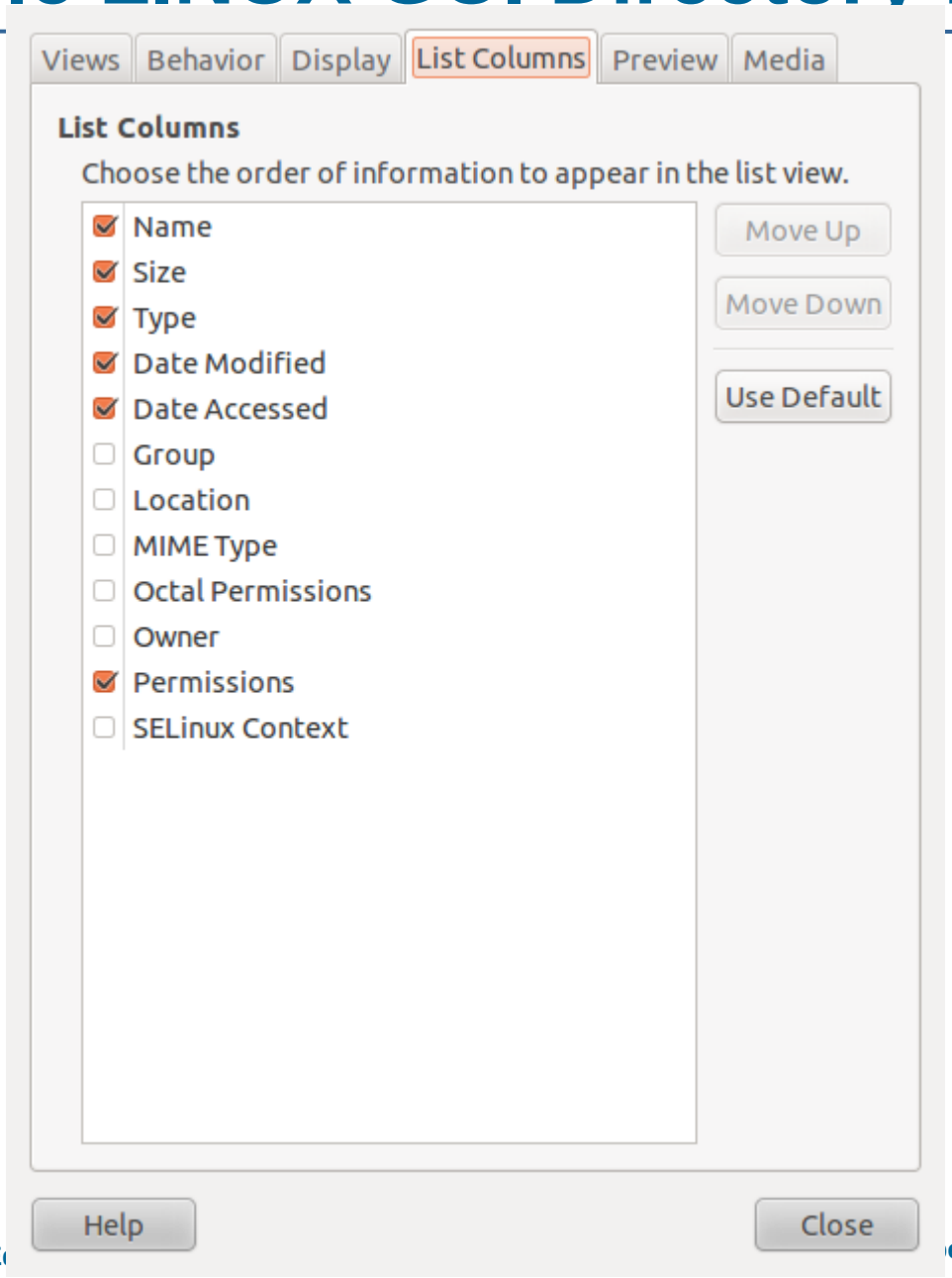
A Sample LINUX Directory Listing

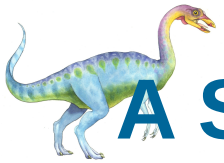
```
File  Edit  View  Search  Terminal  Help
suchenek@nsma131-ms:/media/M-S325/Files/Courses/CSC341$ ls -o -h
total 1.7M
-rw-r--r-- 1 suchenek 1.5K 2010-01-22 22:34 Add_numbers_CSC341_Sp2010~
-rw-r--r-- 1 suchenek 1.4K 2010-01-22 22:42 Add_numbers_CSC541_Sp2010~
-rw-r--r-- 1 suchenek  90K 2010-03-23 22:24 Architecture.pdf
-rw-r--r-- 1 suchenek 4.1K 2010-03-23 22:24 Art of Operating Systems (Peter J. Denning).html
drwx------ 2 suchenek 8.0K 2010-03-23 22:21 CS571 - Operating Systems -- Spring 2002 -- P. J. Denning_files
-rw-r--r-- 1 suchenek  16K 2010-03-23 22:21 CS571 - Operating Systems -- Spring 2002 -- P. J. Denning.html
-rw-r--r-- 1 suchenek 147K 2010-03-23 22:24 Deadlock.pdf
drwx------ 2 suchenek 8.0K 2012-02-22 14:02 handouts
drwx------ 2 suchenek 8.0K 2011-12-15 19:46 Hard_disk_drive_files
-rw-r--r-- 1 suchenek 228K 2010-09-14 05:38 Hard_disk_drive.html
drwx------ 2 suchenek 8.0K 2011-12-15 19:46 HW
drwx------ 3 suchenek 8.0K 2012-04-09 15:17 Images
drwx------ 2 suchenek 8.0K 2012-02-13 20:39 JavaCode
-rw-r--r-- 1 suchenek   24 2010-02-27 00:04 Link_to_companion_website.txt
drwx------ 2 suchenek 8.0K 2012-01-19 14:43 Literature
-rw-r--r-- 1 suchenek 159K 2011-02-09 18:32 Memory_access_time.pdf
-rw-r--r-- 1 suchenek 306K 2010-03-23 22:26 MemPolicy.pdf
drwx------ 4 suchenek 8.0K 2011-12-15 19:46 news_files
-rw-r--r-- 1 suchenek  67K 2010-03-24 04:48 news.html
-rw-r--r-- 1 suchenek 105K 2010-03-23 22:19 OS_Denning.pdf
drwx------ 4 suchenek 8.0K 2012-01-16 17:51 Rosters
drwx------ 7 suchenek 8.0K 2012-04-25 11:23 Slides
drwx------ 2 suchenek 8.0K 2012-02-07 20:12 Solutions
-rw-r--r-- 1 suchenek 201K 2010-03-23 22:24 Synchronization.pdf
drwx------ 3 suchenek  48K 2012-04-25 10:03 TESTS
drwx------ 3 suchenek 8.0K 2012-03-14 11:08 Website
drwx------ 2 suchenek 8.0K 2012-01-19 11:30 Willey_rep_files
-rw-r--r-- 1 suchenek 112K 2012-01-19 11:30 Willey_rep.html
suchenek@nsma131-ms:/media/M-S325/Files/Courses/CSC341$
```

A Sample LINUX Directory Listing

```
File  Edit  View  Search  Terminal  Help
suchenek@nsma131-ms:/media/M-S325/Files/Courses/CSC341/Slides/ProducerConsumer$
ls -o -h -R
.:
total 40K
-rw-r--r-- 1 suchenek 3.6K 2012-03-18 14:49 build.xml
-rw-r--r-- 1 suchenek   82 2012-03-18 14:49 manifest.mf
drwx------ 3 suchenek 8.0K 2012-03-18 14:49 nbproject
drwx------ 3 suchenek 8.0K 2012-03-18 14:49 src
drwx------ 2 suchenek 8.0K 2012-03-18 14:49 test

./nbproject:
total 80K
-rw-r--r-- 1 suchenek  47K 2012-03-18 14:49 build-impl.xml
-rw-r--r-- 1 suchenek  467 2012-03-18 14:49 genfiles.properties
drwx------ 2 suchenek 8.0K 2012-03-18 19:00 private
-rw-r--r-- 1 suchenek 2.3K 2012-03-18 14:49 project.properties
-rw-r--r-- 1 suchenek  509 2012-03-18 14:49 project.xml

./nbproject/private:
total 16K
-rw-r--r-- 1 suchenek  88 2012-03-18 14:49 private.properties
-rw-r--r-- 1 suchenek 416 2012-03-22 11:41 private.xml

./src:
total 8.0K
drwx------ 2 suchenek 8.0K 2012-03-18 14:49 producerconsumer

./src/producerconsumer:
total 8.0K
-rw-r--r-- 1 suchenek 1.9K 2012-03-18 19:00 Main.java

./test:
total 0
suchenek@nsma131-ms:/media/M-S325/Files/Courses/CSC341/Slides/ProducerConsumer$
```

A Sample LINUX GUI Directory Listing





A Sample LINUX GUI Directory Listing

File Edit View Go Bookmarks Help

Back Forward 50% List View

Places M-S325 Files Courses CSC341

Name	Size	Type	Date Modified	Date Accessed	Permissions
JavaCode	6 items	folder	Mon 13 Feb 2012 08:39:44 PM PST	Sun 12 Feb 2012 11:00:00 PM PST	drwx------
Literature	1 item	folder	Thu 19 Jan 2012 02:43:10 PM PST	Wed 18 Jan 2012 11:00:00 PM PST	drwx------
news_files	59 items	folder	Thu 15 Dec 2011 07:46:48 PM PST	Wed 14 Dec 2011 11:00:00 PM PST	drwx------
Rosters	2 items	folder	Mon 16 Jan 2012 05:51:10 PM PST	Sun 15 Jan 2012 11:00:00 PM PST	drwx------
Slides	16 items	folder	Wed 25 Apr 2012 11:59:22 AM PDT	Wed 25 Apr 2012 11:59:22 AM PDT	drwx------
Solutions	10 items	folder	Tue 07 Feb 2012 08:12:32 PM PST	Mon 06 Feb 2012 11:00:00 PM PST	drwx------
TESTS	478 items	folder	Wed 25 Apr 2012 10:03:49 AM PDT	Wed 25 Apr 2012 10:03:49 AM PDT	drwx------
Website	46 items	folder	Wed 14 Mar 2012 11:08:38 AM PDT	Wed 14 Mar 2012 12:00:00 AM PDT	drwx------
Willey_rep_files	8 items	folder	Thu 19 Jan 2012 11:30:52 AM PST	Wed 18 Jan 2012 11:00:00 PM PST	drwx------
Architecture.pdf	89.8 KB	PDF document	Tue 23 Mar 2010 10:24:32 PM PDT	Mon 19 Mar 2012 12:00:00 AM PDT	-rw-r--r--
Art of Operating Systems (Peter J. Denning).html	4.1 KB	HTML document	Tue 23 Mar 2010 10:24:20 PM PDT	Thu 19 Apr 2012 12:00:00 AM PDT	-rw-r--r--
CS571 - Operating Systems -- Spring 2002 -- P. J. Denning...	15.7 KB	HTML document	Tue 23 Mar 2010 10:21:08 PM PDT	Thu 19 Apr 2012 12:00:00 AM PDT	-rw-r--r--
Deadlock.pdf	146.1 KB	PDF document	Tue 23 Mar 2010 10:24:52 PM PDT	Wed 25 Apr 2012 11:56:09 AM PDT	-rw-r--r--
Hard_disk_drive.html	227.5 KB	HTML document	Tue 14 Sep 2010 05:38:42 AM PDT	Thu 19 Apr 2012 12:00:00 AM PDT	-rw-r--r--
Link_to_companion_website.txt	24 bytes	plain text document	Sat 27 Feb 2010 12:04:28 AM PST	Thu 19 Apr 2012 12:00:00 AM PDT	-rw-r--r--
Memory_access_time.pdf	158.7 KB	PDF document	Wed 09 Feb 2011 06:32:56 PM PST	Wed 04 Apr 2012 12:00:00 AM PDT	-rw-r--r--
MemPolicy.pdf	305.5 KB	PDF document	Tue 23 Mar 2010 10:26:50 PM PDT	Wed 04 Apr 2012 12:00:00 AM PDT	-rw-r--r--
news.html	66.4 KB	HTML document	Wed 24 Mar 2010 04:48:46 AM PDT	Thu 19 Apr 2012 12:00:00 AM PDT	-rw-r--r--
OS_Denning.pdf	104.7 KB	PDF document	Tue 23 Mar 2010 10:19:24 PM PDT	Mon 19 Mar 2012 12:00:00 AM PDT	-rw-r--r--
Synchronization.pdf	200.8 KB	PDF document	Tue 23 Mar 2010 10:24:42 PM PDT	Mon 19 Mar 2012 12:00:00 AM PDT	-rw-r--r--
Willey_rep.html	111.7 KB	HTML document	Thu 19 Jan 2012 11:30:52 AM PST	Thu 19 Apr 2012 12:00:00 AM PDT	-rw-r--r--

GiftOf... NetBe... Materi... Mathe... CSC311 Website Slides Mathe... TESTS Spring12 CSC341 Website Slides TESTS Spring... CSC401 Website Mathe... Slides TESTS Spring12 CSC501 Website CSC541 Website

26 items, Free space: 7.7 GB

End of Chapter 10