BUCKNELL UNIVERSITY
Computer Science

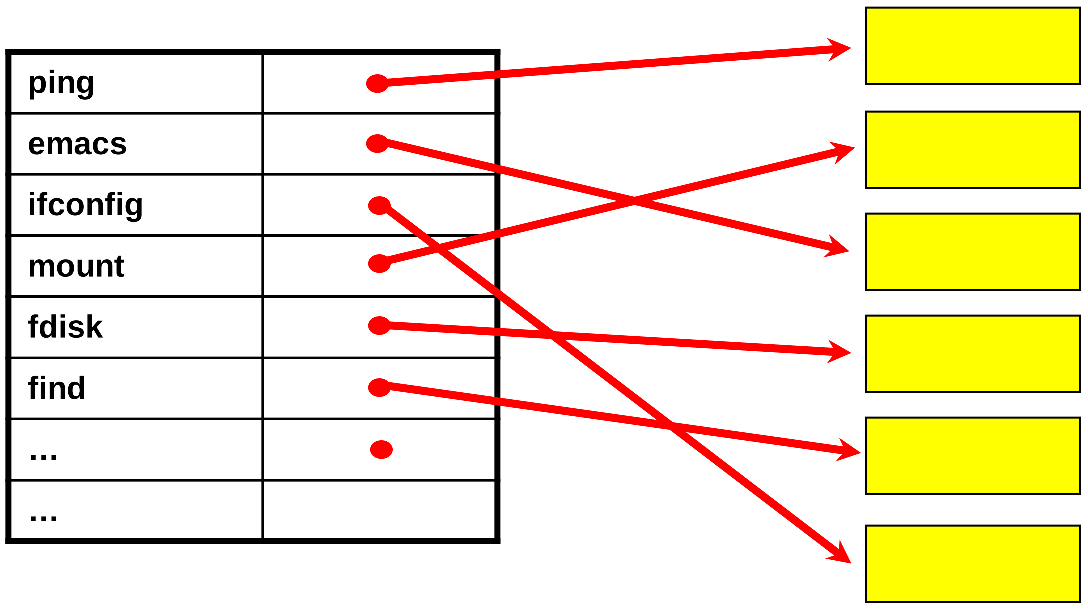
CSCI 315 Operating Systems Design

Directories and Meta-Data

Notice: The slides for this lecture have been largely based on Professor Perrone's notes. Revised by Xiannong Meng.

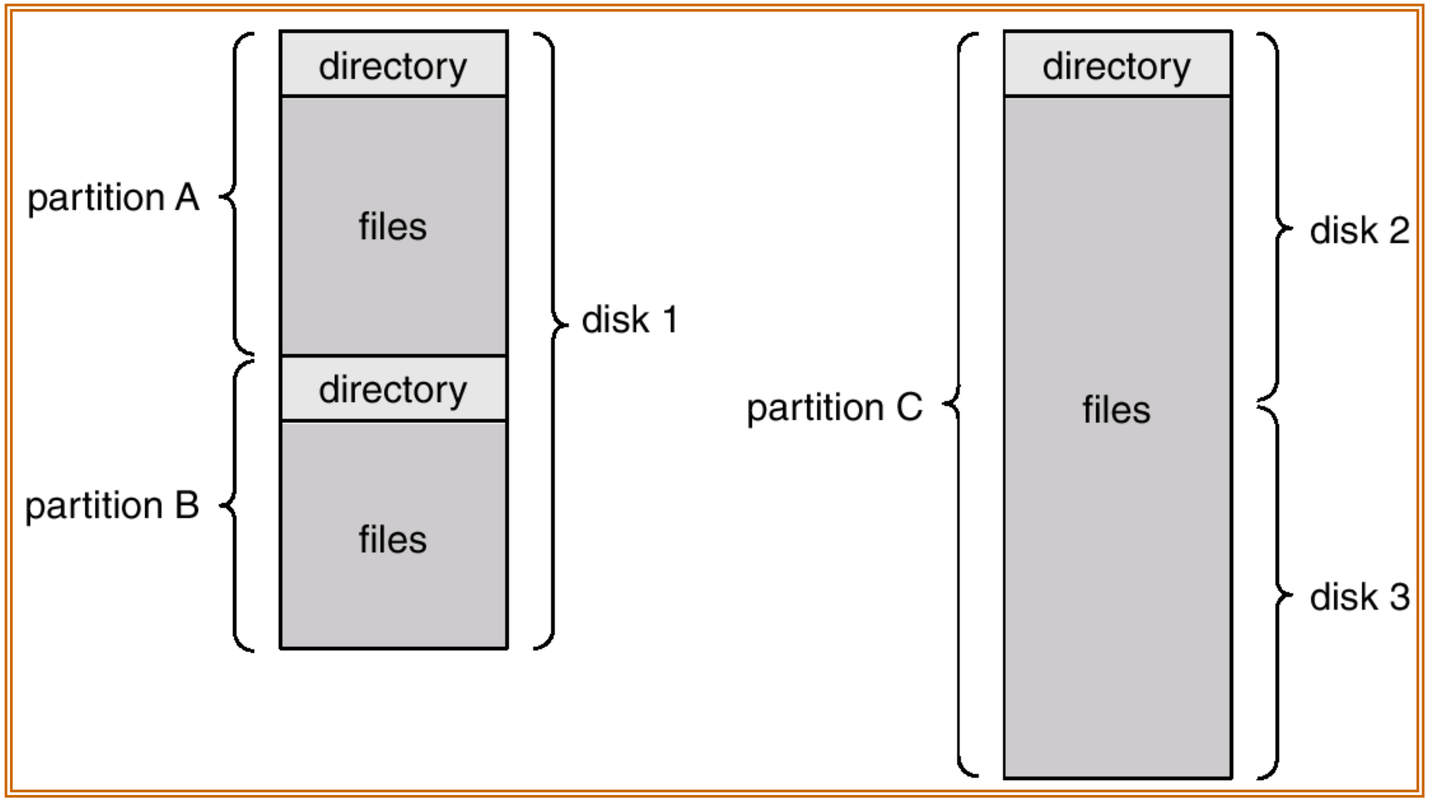
Directory Structure

Directory: a symbol table that maps file names into directory entries.



Both the directory structure and the files reside on disk. Backups of these two structures are kept on back-up storage.

Partitions and Directories
(File system organization)



Operations on Directories

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

Example of Directory Listing

```
#include <stdlib.h>
#include <sys/types.h>
#include <dirent.h>

int main(int argc, char* argv[]) {

  struct dirent *dp;
  DIR *dirp;

  if (argc != 2) {
     fprintf(stderr, "usage %s dir_name\n", argv[0]);
     exit(1);
  }
  char * dname = argv[1];

  dirp = opendir(dname);
  if (dirp != NULL) { // it is a directory

    printf("directory : %s\n",dname);

    for (dp = readdir(dirp); NULL != dp; dp = readdir(dirp)) {
      printf("%s\n", dp->d_name);
    }
    closedir (dirp);
  }

  return 0;
}
```

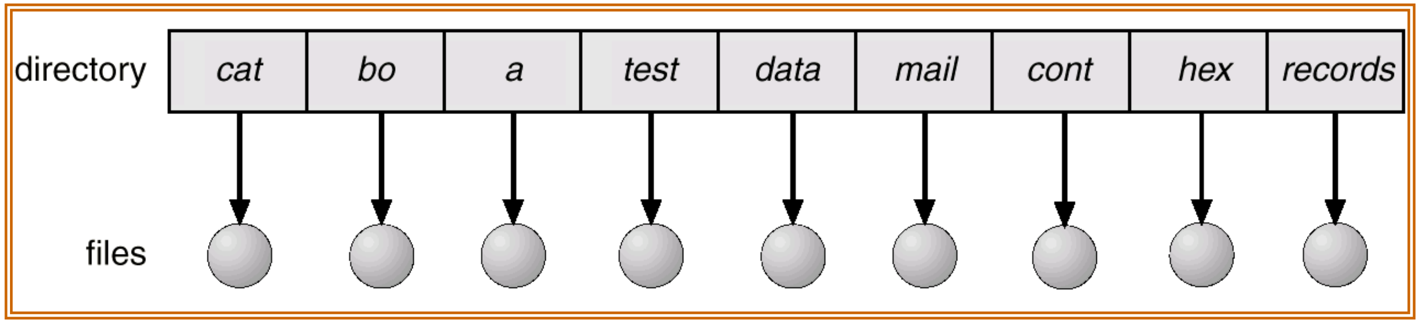
```
[xmeng@linuxremote1 files]$ gcc list_dir.c
[xmeng@linuxremote1 files]$ ./a.out ../
directory : ../
.
..
thread
sync
process
deadlock
scheduling
memory
files
[xmeng@linuxremote1 files]$ ./a.out ./
directory : ./
.
..
file-test.c
a.out
file-test.c~
list_dir.c
hello.txt
list_dir.c~
[xmeng@linuxremote1 files]$
```

Goals of Directory Logical Organization

- **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.



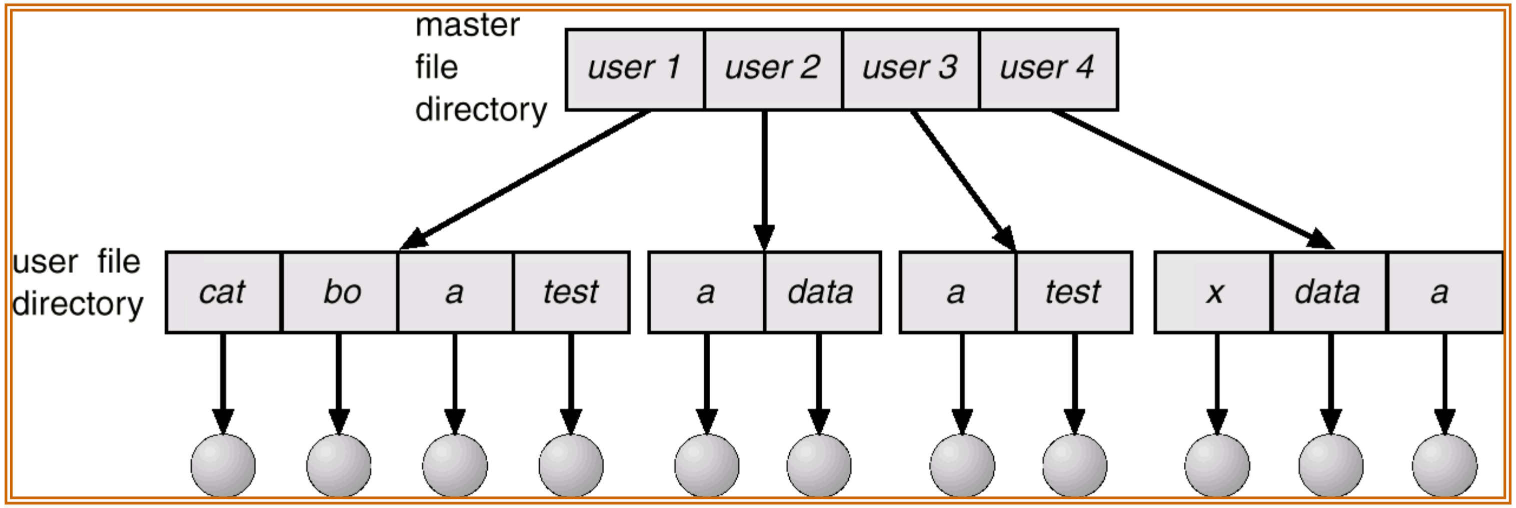
Drawbacks:

Naming problem

Grouping problem

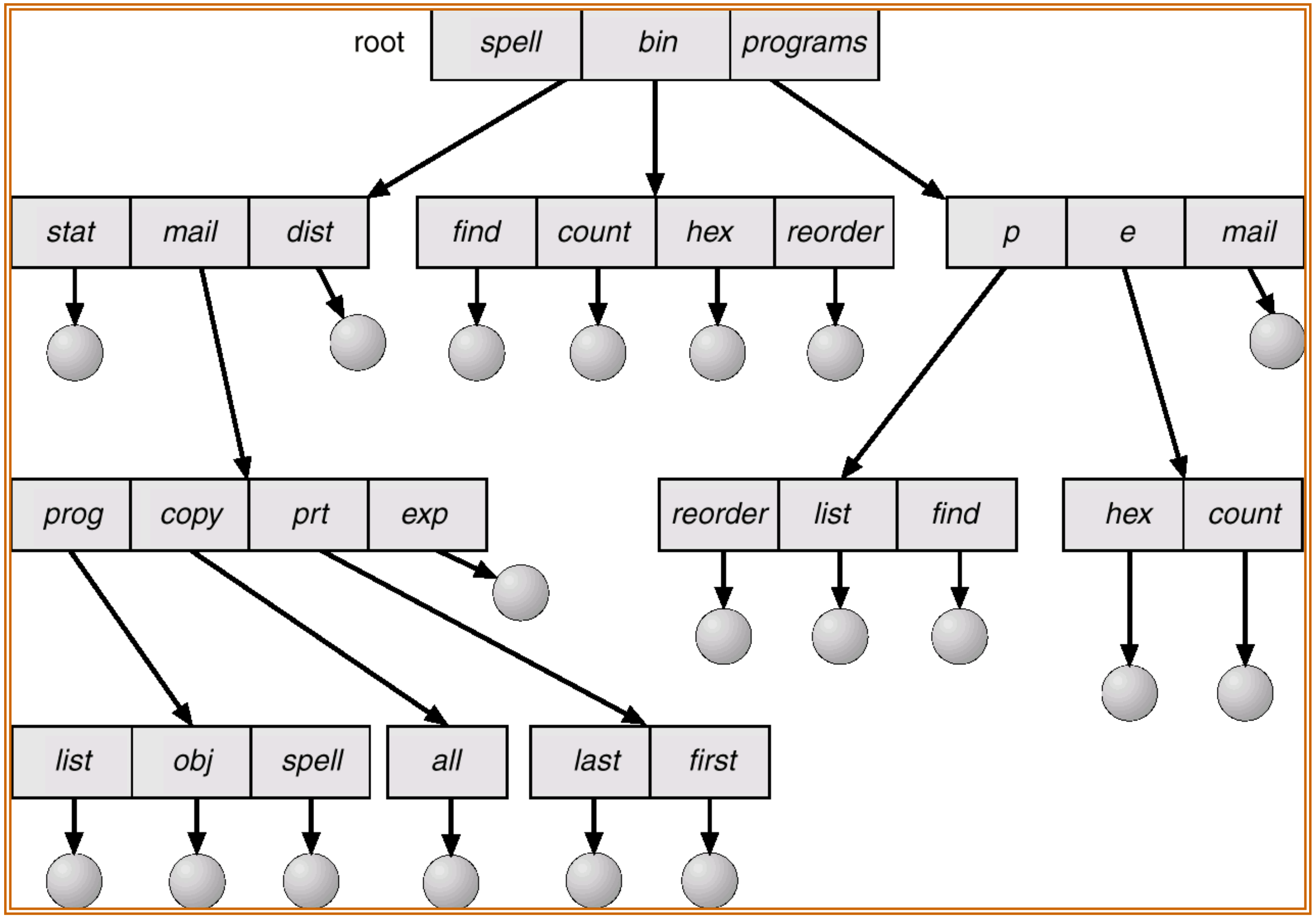
Two-Level Directory

A 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.)

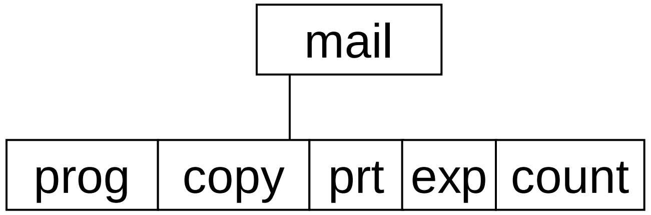
- **Absolute** or **relative** path name.
- Creating a new file is done in current directory by default.
- Delete a file

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

  mkdir <dir-name>

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

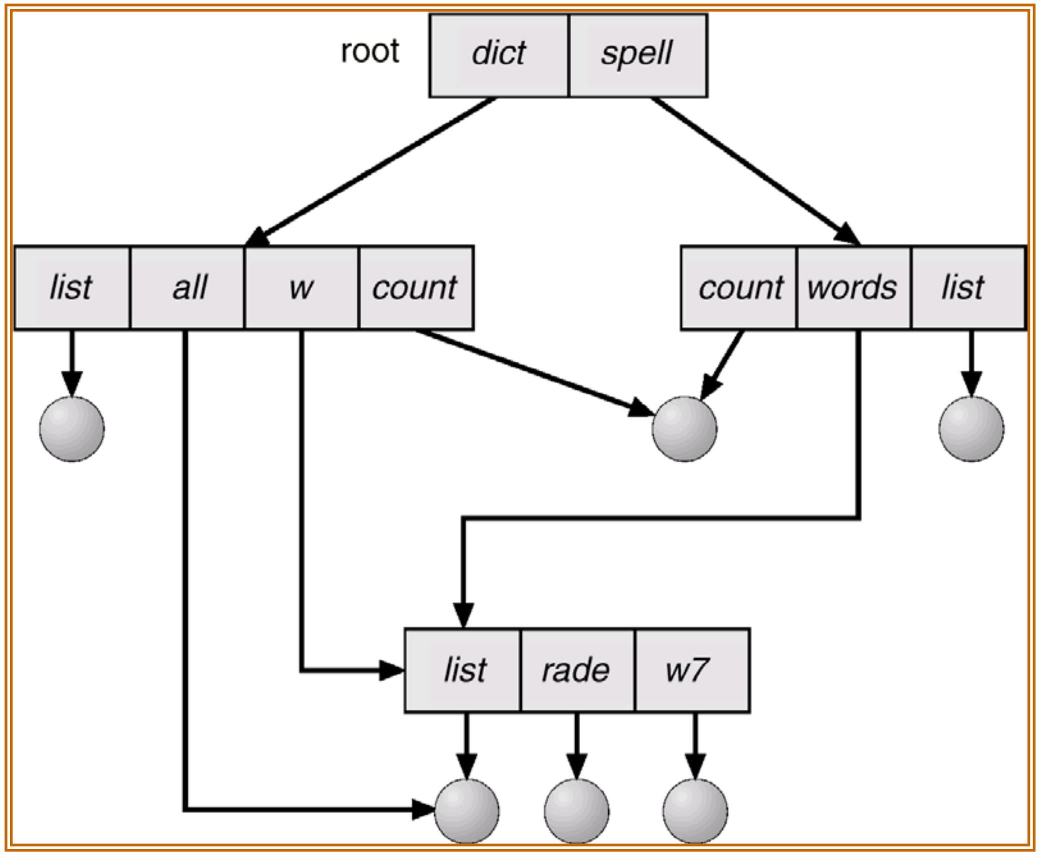
  mkdir count



Deleting "mail" ⇒ deleting the entire subtree rooted by "mail".

Acyclic-Graph Directories

Have shared subdirectories and files.



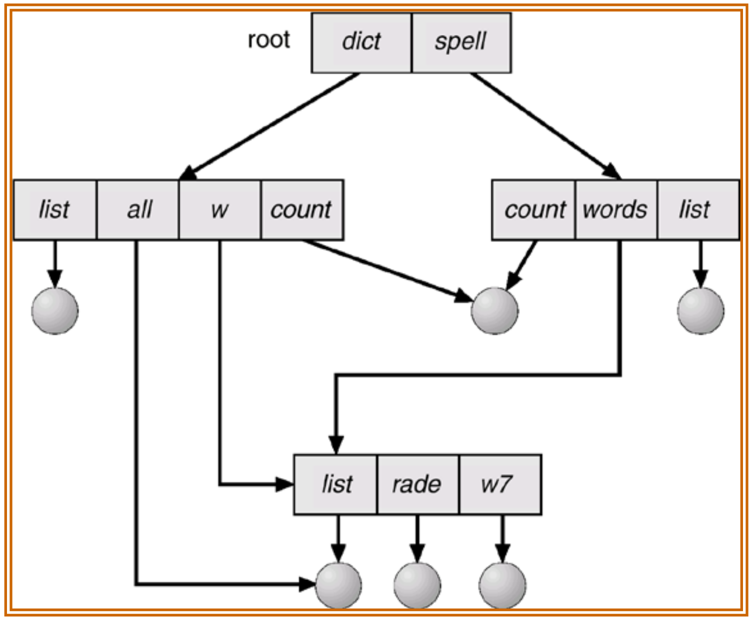
Acyclic-Graph Directories (Cont.)

- Two different names (aliasing).
- If *dict* deletes *list* $\Rightarrow$ 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.

Acyclic-Graph Directories

Have <u>shared</u> subdirectories and files.



links: {soft (symbolic)
hard

Unix: **ln** (read man page);

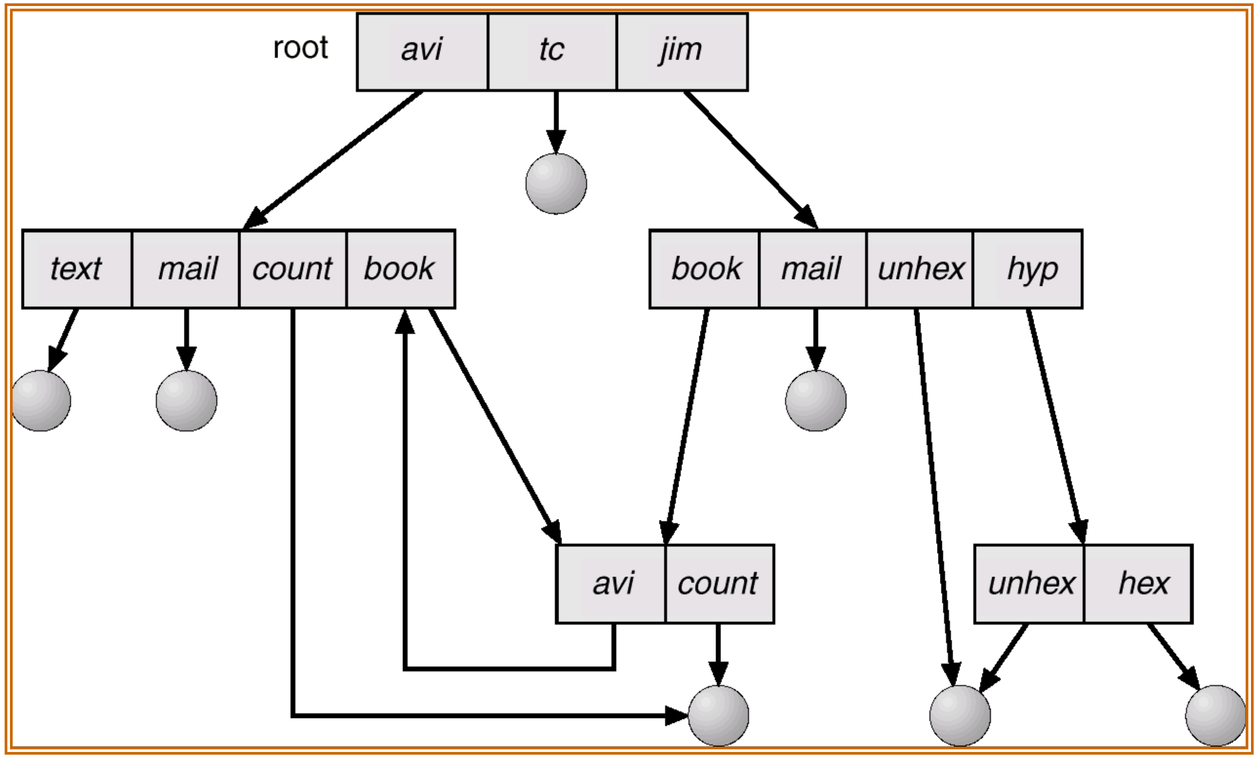
need to keep a reference count on each file or directory.

Acyclic-Graph Directories (Cont.)

- Different names (aliasing) for the same file or directory.
- 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.

General Graph Directory

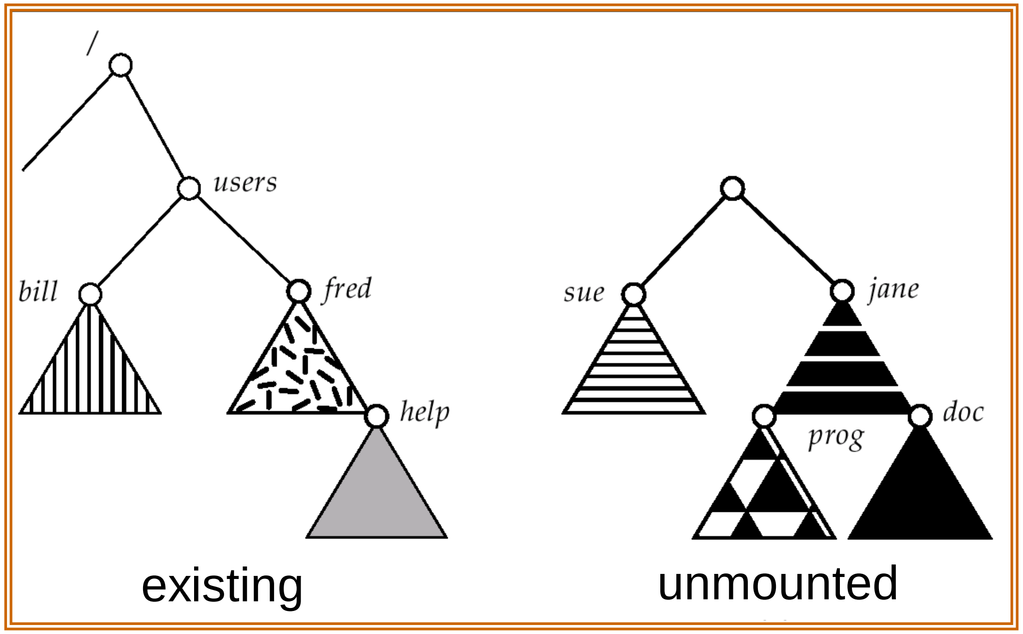


General Graph Directory (Cont.)

- **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 (partition) must be **mounted** before it can be accessed. Mounting allows one to attach the file system on one device to the file system on another device.
- A unmounted file system needs to be attached to a **mount point** before it can be accessed.



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.

Protection

- **File owner/creator should be able to control:**
  - what can be done,
  - by whom.

Discretionary Access Control (DAC)

- **Types of access:**
  - Read,
  - Write,
  - Execute,
  - Append,
  - Delete,
  - List.

Protection

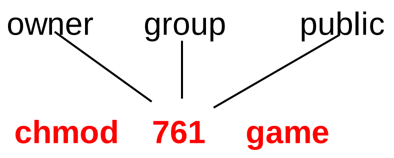
- **Mandatory Access Control (MAC):**
  - **System policy:** files tied to access levels = (public, restricted, confidential, classified, top-secret).
  - Process also has access level: can read from and write to all files at same level, can only read from files below, can only write to files above.
- **Role-Based Access Control (RBAC):**
  - **System policy:** defines **“roles”** (generalization of the Unix idea of groups).
  - Roles are associated with access rules to sets of files and devices.
  - A process can change roles (in a pre-defined set of possibilities) during execution.

Access Lists and Groups

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

		RWX
a) **owner access**	7 $\Rightarrow$	1 1 1
		RWX
b) **group access**	6 $\Rightarrow$	1 1 0
		RWX
c) **public access**	1 $\Rightarrow$	0 0 1

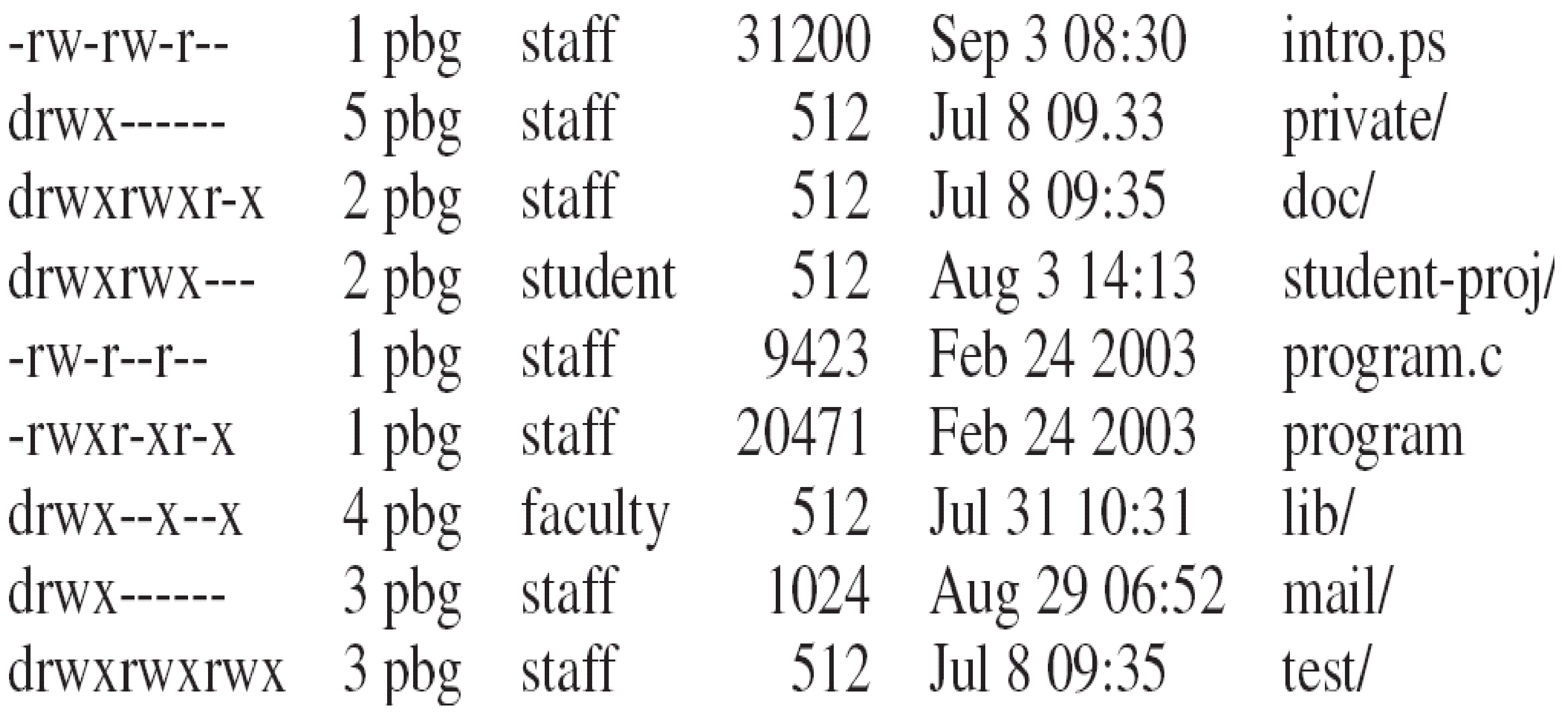
- Ask manager to create a group (unique name), say G, and add some users to the group.
- For a particular file (say *game*) or subdirectory, define an appropriate access.



```
chmod 761 game
```

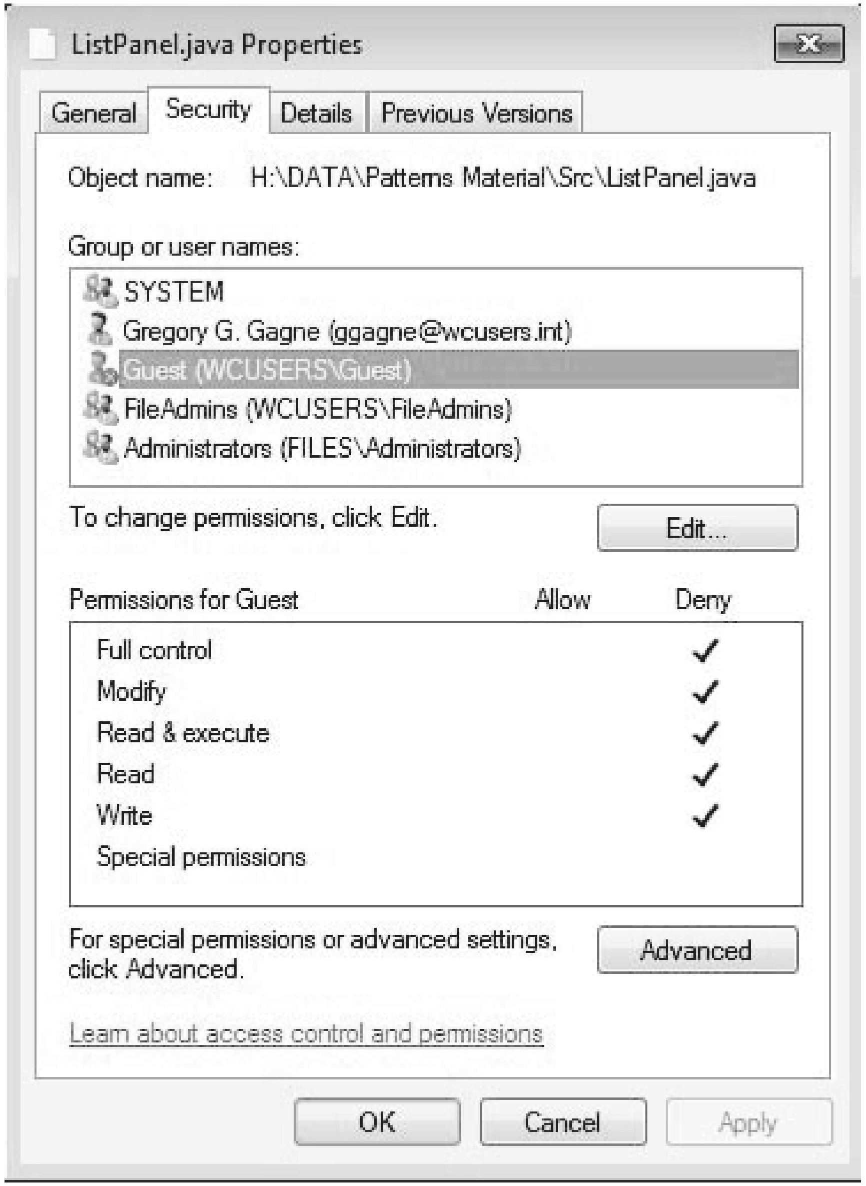
Associate a group with a file: `chgrp G game`

A Sample UNIX Directory Listing



```
-rw-rw-r--    1 pbg   staff     31200   Sep 3 08:30     intro.ps
drwx------    5 pbg   staff       512   Jul 8 09.33     private/
drwxrwxr-x    2 pbg   staff       512   Jul 8 09:35     doc/
drwxrwx---    2 pbg   student     512   Aug 3 14:13     student-proj/
-rw-r--r--    1 pbg   staff      9423   Feb 24 2003     program.c
-rwxr-xr-x    1 pbg   staff     20471   Feb 24 2003     program
drwx--x--x    4 pbg   faculty     512   Jul 31 10:31    lib/
drwx------    3 pbg   staff      1024   Aug 29 06:52    mail/
drwxrwxrwx    3 pbg   staff       512   Jul 8 09:35     test/
```

Windows 7 Access-Control List Management



ListPanel.java Properties

General | Security | Details | Previous Versions

Object name: H:\DATA\Patterns Material\Src\ListPanel.java

Group or user names:

- SYSTEM
- Gregory G. Gagne (ggagne@wcusers.int)
- Guest (WCUSERS\Guest)
- FileAdmins (WCUSERS\FileAdmins)
- Administrators (FILES\Administrators)

To change permissions, click Edit. Edit...

Permissions for Guest	Allow	Deny
Full control		✓
Modify		✓
Read & execute		✓
Read		✓
Write		✓
Special permissions		

For special permissions or advanced settings, click Advanced. Advanced

Learn about access control and permissions

OK Cancel Apply