Chương 8
File-System Interface

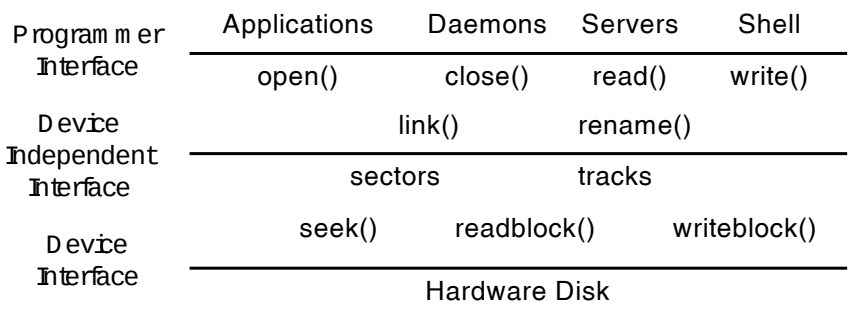
Nội dung

- Khái niệm cơ sở
- Các phương thức truy cập
- Cấu trúc thư mục
- File-System mounting
- Chia sẻ file
- Bảo vệ (Protection)

Khái niệm cơ sở

User Abstraction		Hardware Resource
Process/Thread		CPU
Address Space	⇐ **OS** ⇒	Memory
Files		Disk

Mô hình trừu tượng hóa File System



Những yêu cầu của user

- ***Persistence***: dữ liệu được lưu giữ bền vững
- ***Speed***: truy xuất dữ liệu nhanh
- ***Size***: có thể lưu trữ thật nhiều dữ liệu
- ***Sharing/protection***: user có thể chia sẻ dữ liệu khi cần thiết và có thể bảo vệ dữ liệu riêng tư khi cần thiết
- ***Ease of use***: user có thể dễ dàng tìm kiếm, kiểm tra, hiệu chỉnh, cập nhật dữ liệu

Đặc điểm của phần cứng và OS

- Phần cứng cung cấp sự hỗ trợ sau
  - *Persistence* : các thiết bị lưu trữ bền vững (non-volatile memory)
  - *Speed* : cung cấp khả năng truy xuất ngẫu nhiên (random access), nâng cao tốc độ đĩa (5400 → 7200 → 10K rpm,...)
  - *Size* : dung lượng đĩa ngày càng lớn (40GB, 80GB, 120GB,...)
- Hệ điều hành cung cấp
  - *Persistence* : lưu trữ dư thừa, back-up phục vụ cho recovery (ví dụ: RAID,...).
  - *Ease of user* :
    - Gán tên cho một khối dữ liệu (file)
    - Tổ chức cấu trúc quản lý file: thư mục
    - Thực hiện quá trình ánh xạ file (cái nhìn luận lý của user) vào không gian lưu trữ vật lý (trên đĩa). Quá trình này *trong suốt* đối với user.
  - *Sharing/Protection* : các quyền truy cập file/thư mục

Khái niệm cơ sở (t.t)

- Để quản lý các thiết bị lưu trữ một cách hiệu quả, OS che dấu chi tiết cấp thấp của các thiết bị lưu trữ vật lý, chỉ định nghĩa các khái niệmlưu trữ luận lý
- Các khái niệm luận lý (user view)
  - *File*
  - *File description*
  - *Directory*
  - *File System*

Các thuộc tính của File

- *Name* – tên file (human-readable form)
- *Type* – binary, text, image, …
- *Location* – con trỏ đến vị trí file trên thiết bị lưu trữ
- *Size* – kích thước hiện tại (và mức tối đa cho phép !!!)
- *Protection* – kiểm soát quyền đọc, ghi, thực thi file
- *Time, date, user identification* – các thông tin dùng cho protection, security, và usage monitoring.
- Thông tin về file được giữ trong cấu trúc thư mục. Cấu trúc này được lưu trữ trên thiết bị lưu trữ vật lý.

Các tác vụ trên file/thư mục

- ***Create***
  - Cấp phát không gian lưu trữ
  - Thêm entry trong thư mục
- ***Write***
  - Tìm file trong thư mục
  - Thực hiện tác vụ ghi tại vị trí con trỏ ghi (write pointer)
- ***Read***
  - Tìm file trong thư mục
  - Thực hiện tác vụ đọc tại vị trí con trỏ đọc (read pointer)
- ***Reposition – file seek***
  - Thiết lập con trỏ đọc/ghi đến vị trí được chỉ định (tái định vị).
- ***Delete***
  - Tìm thư mục chứa file cần xóa
  - Xóa các disk blocks của file
  - Xóa entry trong thư mục
- ***Truncate***
  - Giữ lại tất cả các thuộc tính file, ngoại trừ kích thước file = 0
- ***Open(Fi)***
  - Tìm file Fi trong cấu trúc thư mục Search trên đĩa
  - Kiểm tra quyền truy cập
  - Nạp nội dung entry Fi trong thư mục vào bộ nhớ
- ***Close(Fi)***
  - Chuyển nội dung entry Fi trong bộ nhớ vào đĩa

Các cấu trúc file

- Không cấu trúc – một chuỗi words, bytes
- Cấu trúc record đơn giản
  - Lines
  - Fixed length
  - Variable length
- Cấu trúc phức tạp
  - Formatted document (Word/Excel document,...)
  - Relocatable load file (executable, dynamic linking library,...).
- Mọi hệ điều hành phải hỗ trợ ít nhất một cấu trúc – đó là cấu trúc của *file thực thi* (DOS/Windows: exe, com,...)
- Ai quyết định cấu trúc của file
  - Operating System ?
  - Program ?

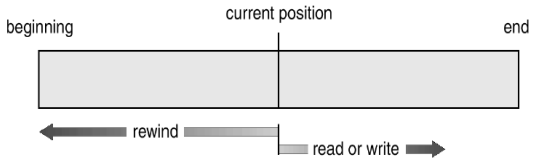
Kiểu file – Name, Extension

file type	usual extension	function
executable	exe, com, bin or none	read 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, rrf, doc	various word-processor formats
library	lib, a, so, dll, mpeg, mov, rm	libraries of routines for programmers
print or view	arc, zip, tar	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	binary file containing audio or A/V information

Access Methods

- ***Sequential Access***: truy xuất thông tin tuần tự
  - *read (đọc và tự động dời con trỏ file đến vị trí kế tiếp)*
  - *write*
  - *trở về đầu file, đến cuối file*
  - *forward/backward n record*

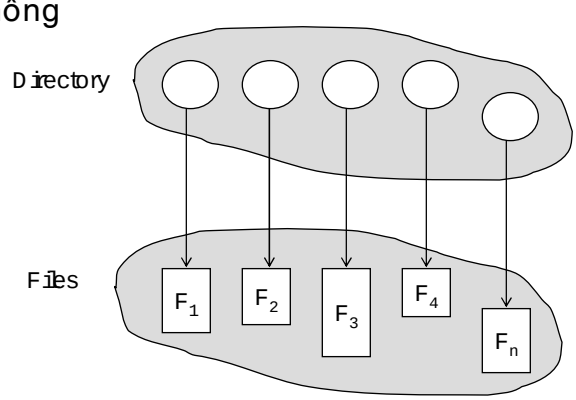
- ***Direct Access*** (relative access)
  - file cấu thành từ các khối, record kích thước cố định
  - *read n*
  - *write n*
  - *position to n*
  - *read next*
  - *write next*
  - *rewrite n*



Minh họa Sequential Access

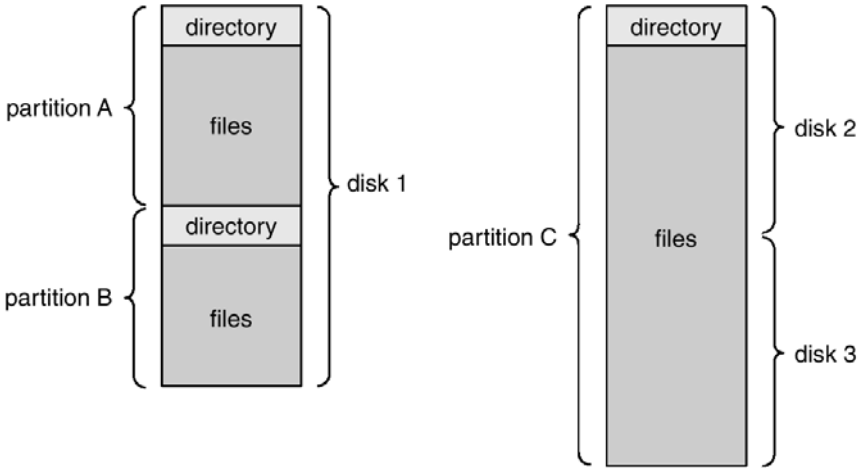
Cấu trúc thư mục

- Là tập hợp các node lưu giữ thông tin của các file trong hệ thống



Cả cấu trúc thư mục và các file đều nằm trên đĩa

Ví dụ về cấu trúc File-System



Thông tin của thư mục

- Name
- Type: file, directory, symbolic link, ...
- Address: địa chỉ của khối lưu trữ file
- Current length
- Maximum length
- Date last accessed
- Date last updated
- Owner ID
- Protection information
- ...

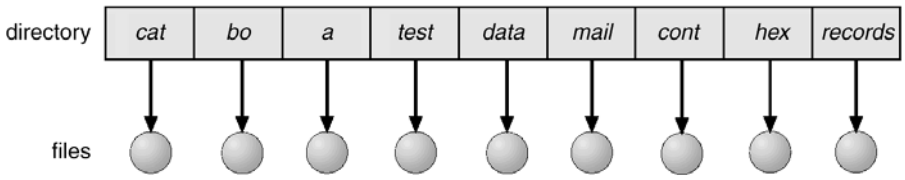
Các tác vụ trên thư mục

- ***Search***: tìm trong cấu trúc thư mục để lấy được thông tin trong entry của một file
- ***Create***: tạo một file mới và thêm vào cấu trúc thư mục để quản lý
- ***Delete***: xóa một file
- ***List***: hiển thị nội dung một thư mục
- ***Rename***: đổi tên file
- ***Traverse***: duyệt toàn bộ hệ thống file
- Thực chất, thư mục cũng là một ***file đặc biệt***, cung cấp sự ánh xạ tương ứng giữa định danh của file và tất cả thông tin, dữ liệu khác của file.

Mục tiêu tổ chức cấu trúc thư mục

- ***Hiệu quả*** (efficiency) – định vị nhanh chóng một file nào đó.
- ***Đặt tên*** (naming) – thuận tiện cho user. Ví dụ
  - Hai users có thể có file trùng tên (nhưng khác nhau)
  - Một file có thể có nhiều tên khác nhau.
- ***Nhóm*** (grouping) – tổ chức thành các nhóm file luận lý dựa trên các thuộc tính của file, (ví dụ: file chương trình C, file tài liệu MS Word, file chương trình games, …)

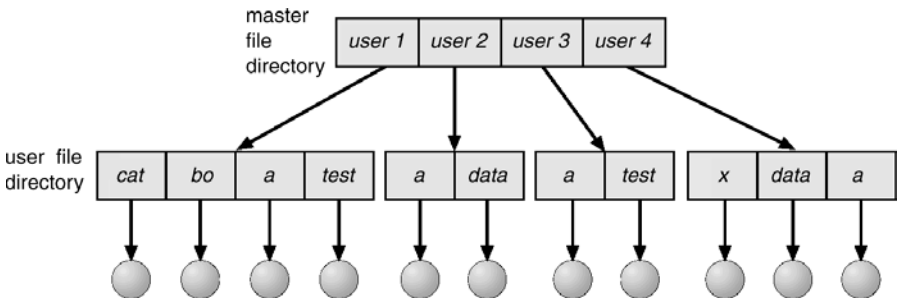
Cấu trúc single-level



☹ Naming

☹ Grouping

Cấu trúc two-level



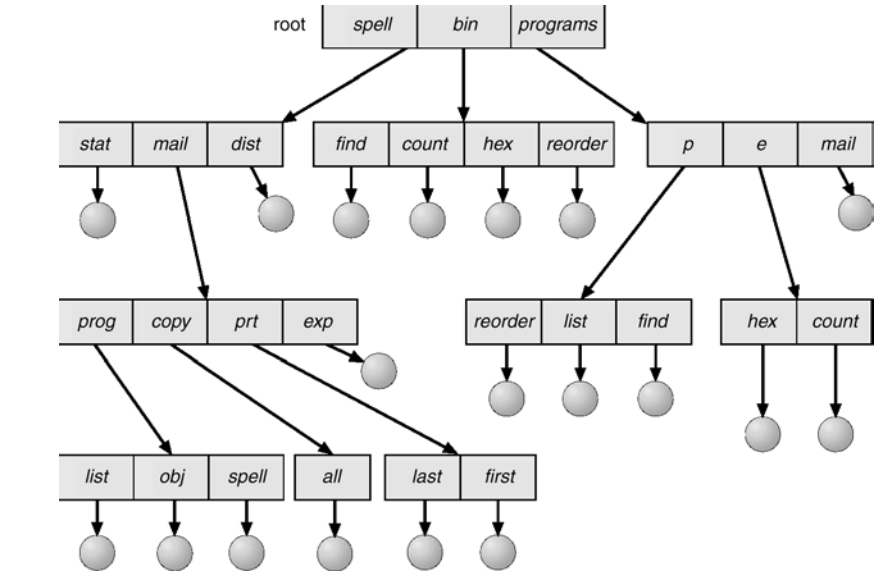
☺ Khái niệm đường dẫn - path name: không có ở single-level

☺ Hai users có thể có file trùng tên

☺ Tìm kiếm hiệu quả hơn

☹ Không hỗ trợ khả năng grouping

Cấu trúc cây thư mục



Cấu trúc cây thư mục (t.t)

☺ Tìm kiếm hiệu quả

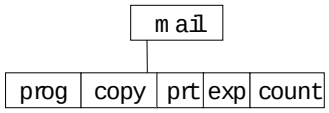
☺ Cung cấp khả năng grouping (bằng cách tổ chức cấu trúc thư mục con)

☺ Có thể dùng đường dẫn tuyệt đối hoặc tương đối

☺ Khái niệm thư mục làm việc (working directory). Ví dụ

cd /spell/mail/prog

cat list



☺ Xóa cây thư mục

xóa thư mục mail

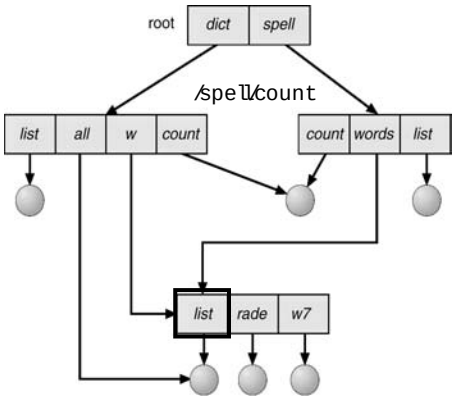
☹ Tổ chức cấu trúc dữ liệu quản lý khá phức tạp (phần hiện thực hệ thống file)

Cấu trúc acyclic-graph

- Có thể chia sẻ thư mục con và file.
  - Một file chia sẻ có thể có nhiều tên (aliasing)
- Hiện thực chia sẻ
  - ***Symbolic link***: một con trỏ đến file hoặc thư mục khác

    ln -s /spell/count /dict/count
  - ***Hard link***: nhân bản thông tin trong thư mục chia sẻ

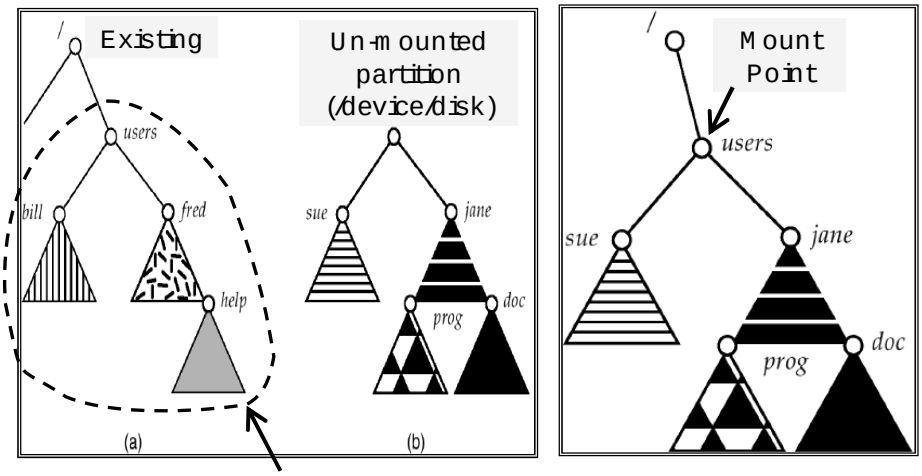
- Nếu file /dict/w/list bị xóa ⇒ dangling pointer
- Giải pháp ???



File system mounting

- Để có thể truy xuất, file system phải được gán lắp (mount)
- Một file system được gắn tại một thư mục, gọi là ***mount point***.
- Các bước tiến hành gán lắp: *mount /device/dsk /users*
  - Cung cấp tên thiết bị và mount point
  - Kiểm tra xem thiết bị có chứa một file system hợp lệ không?
  - Gán lắp file system vào vị trí *mount point*
- Nếu mount-point có chứa file?
  - Các file của thư mục *mount point* sẽ không nhìn thấy được, trừ phi un-mount file system.
- Có thể gắn lắp các file system khác nhau tại các *mount point* khác nhau.

File system mounting (t.t)



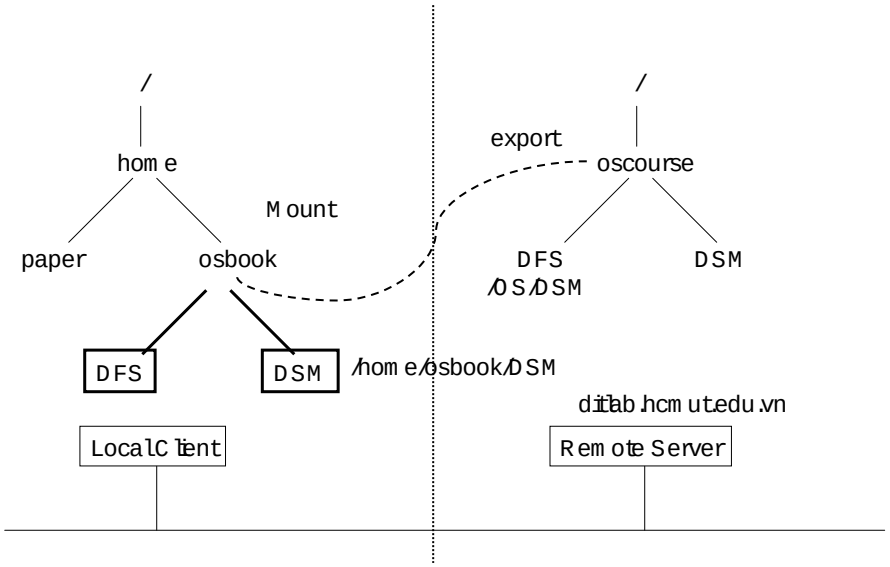
Vùng không nhìn thấy khi /device/disk được mount

Lệnh gán lắp: ***mount /device/dsk /users***

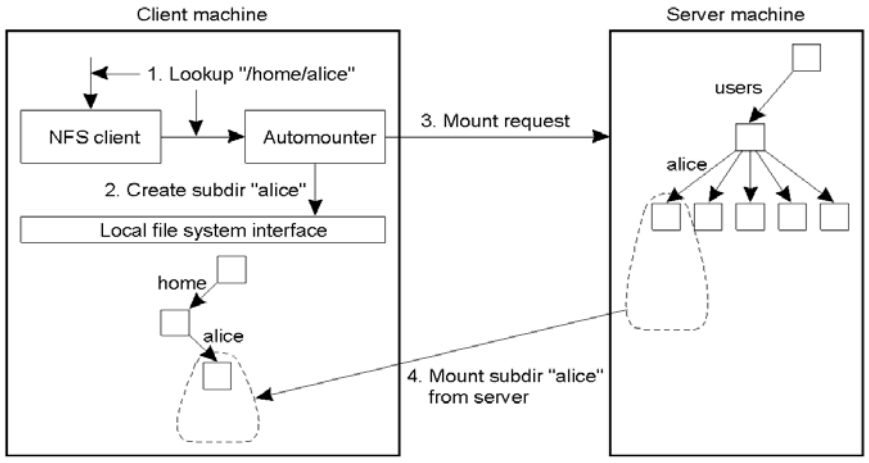
Chia sẻ file

- Nhu cầu chia sẻ file trong hệ thống multi-user rất lớn
- Việc chia sẻ cũng có thể thực hiện qua cơ chế bảo vệ, kiểm soát nghiêm ngặt
- Trên hệ thống phân bố (distributed systems), các file có thể được chia sẻ qua mạng máy tính
- *Network File System* (NFS) là phương pháp chia sẻ file rất phổ biến dùng cơ chế RPC (mô hình client-server)
- Một file system từ xa cũng phải được mount trước khi sử dụng
  - ***mount*** ditlab.hcmut.edu.vn:/oscourse /home/osbook
  - Khi file system đã được mount, mọi truy xuất đến file/thư mục trên đó đều như bình thường, không phân biệt là local hay remote nữa.
  - Vấn đề: naming (DNS, NIS, LDAP, Active Directory, ...)

Ví dụ mount remote file-system



Automounter trên NFS



Bảo vệ (protection)

- Chủ nhân của một file phải được cung cấp khả năng kiểm soát, điều khiển quá trình truy cập file
  - Các tác vụ có thể thực hiện trên file?
  - Những ai được quyền thực hiện thao tác trên file?
- Các kiểu truy xuất file
  - Read
  - Write
  - Execute
  - Append
  - Delete
  - List
- Để kiểm soát quá trình truy cập, cách tiếp cận thông thường là gán kèm theo mỗi file một access-control list (ACL)

Access-control list và group

- Các chế độ truy cập: read, write, execute
- Ba nhóm user: owner, group, universe (public)
  - User name và user identifier (user ID)
  - Group name và group identifier (group ID)
- Ví dụ trong UNIX: có 3 field, mỗi field có 3 bits
  - Field : read, write, execute
  - Bit : r(4), w(2), x(1)

			RWX
owner access	7	⇒	1 1 1
			RWX
group access	6	⇒	1 1 0
			RWX
public access	1	⇒	0 0 1