

# HỆ ĐIỀU HÀNH (OPERATING SYSTEM)

Trình bày: Nguyễn Hoàng Việt  
Khoa Công Nghệ Thông Tin  
Đại Học Cần Thơ

# Chương 10: Cài đặt hệ thống tập tin (File System Implementation)

---

- Cấu trúc hệ thống tập tin
- Cài đặt hệ thống tập tin
- Cài đặt thư mục
- Các phương pháp cấp phát đĩa
- Quản lý không gian trống
- Hiệu suất và hiệu năng

# Cấu trúc hệ thống tập tin (1)

## (File System Structure)

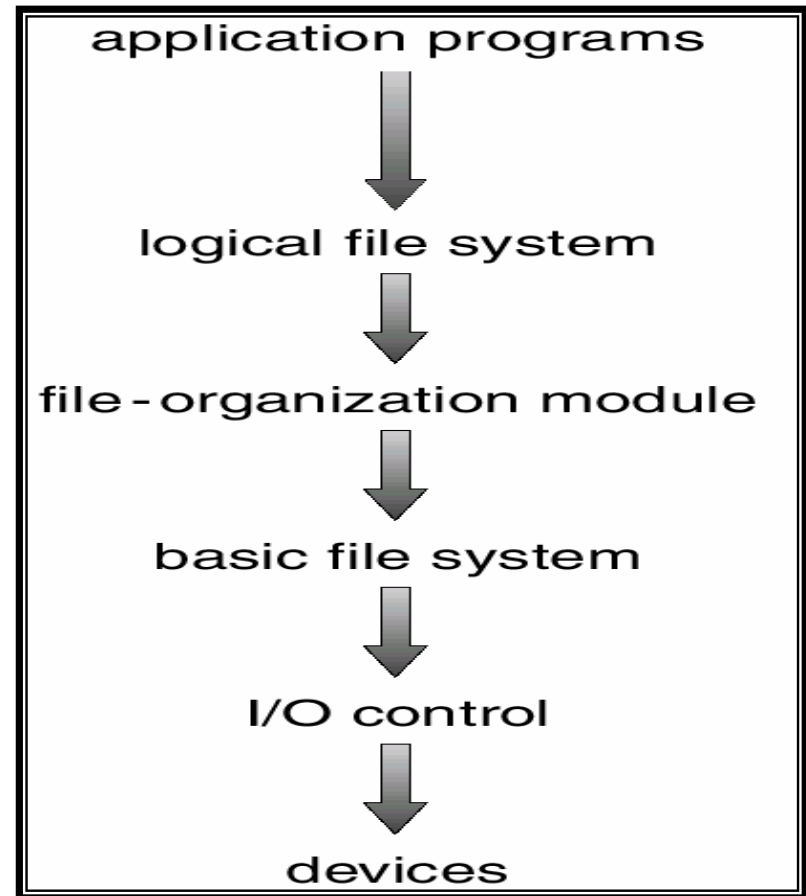
---

- Cấu trúc tập tin:
  - Bao gồm các đơn vị lưu trữ luận lý.
  - Tập tin là tập hợp được đặt tên của các thông tin có liên hệ với nhau.
- Hệ thống tập tin nằm trên thiết bị lưu trữ thứ cấp (đĩa).
  - Đĩa từ là một phương tiện thuận lợi: có thể ghi xóa (rewritable), truy xuất trực tiếp (direct access)
- Hai vấn đề thiết kế:
  - Cách hệ thống tập tin thể hiện đối với người dùng
  - Cách ánh xạ từ hệ thống tập tin luận lý vào thiết bị lưu trữ thứ cấp.
- Hệ thống tập tin được tổ chức thành nhiều tầng.
- **Khối kiểm soát tập tin (File metadata/File control block):** chứa thông tin về một tập tin.

# Cấu trúc hệ thống tập tin (2)

## Hệ thống tập tin phân tầng (Layered File System)

- **I/O control:** gồm device driver và interrupt handler để chuyển thông tin giữa bộ nhớ và đĩa.
- **Basic file system:** các lệnh tương ứng với thiết bị để đọc và viết các khối vật lý.
- **File-organization module:** dịch địa chỉ các khối luận lý thành địa chỉ các khối vật lý.
- **Logical file system:** quản lý cấu trúc hệ thống tập tin, cấu trúc thư mục, bảo vệ và an toàn.

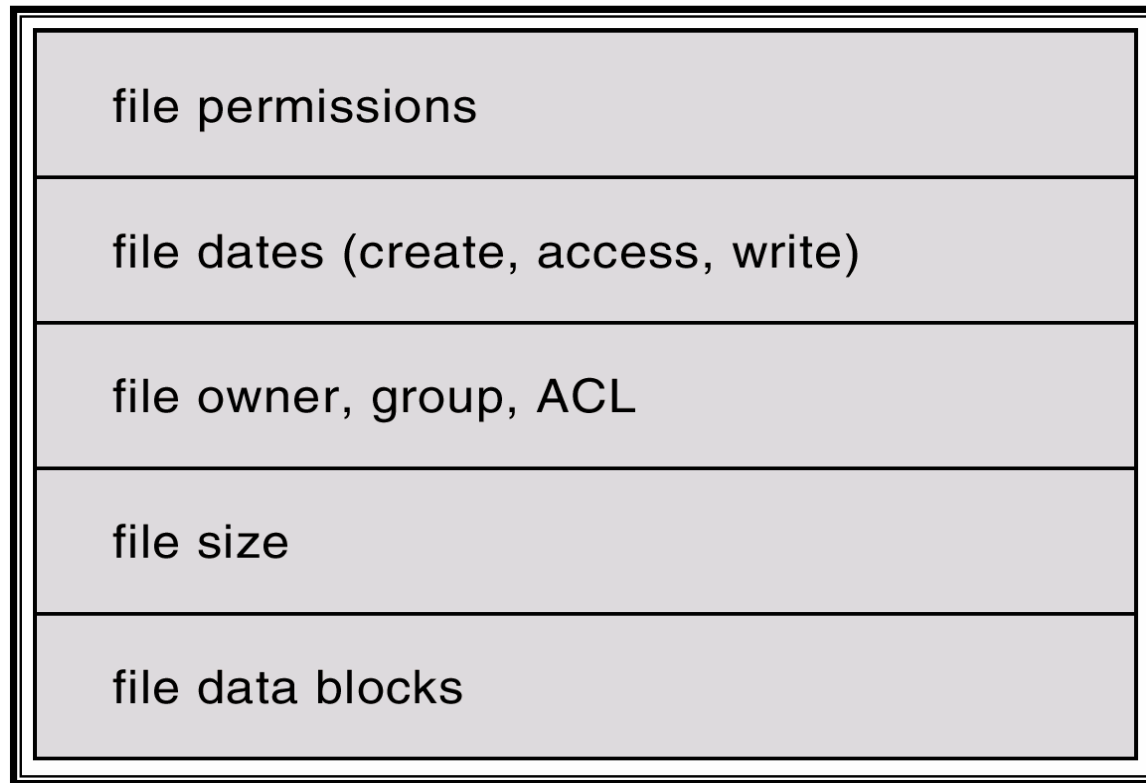


# Cài đặt hệ thống tập tin (1)

## Một khối kiểm soát tập tin (File Control Block)

---

- Cấu trúc lưu trữ, bao gồm các thông tin về một tập tin

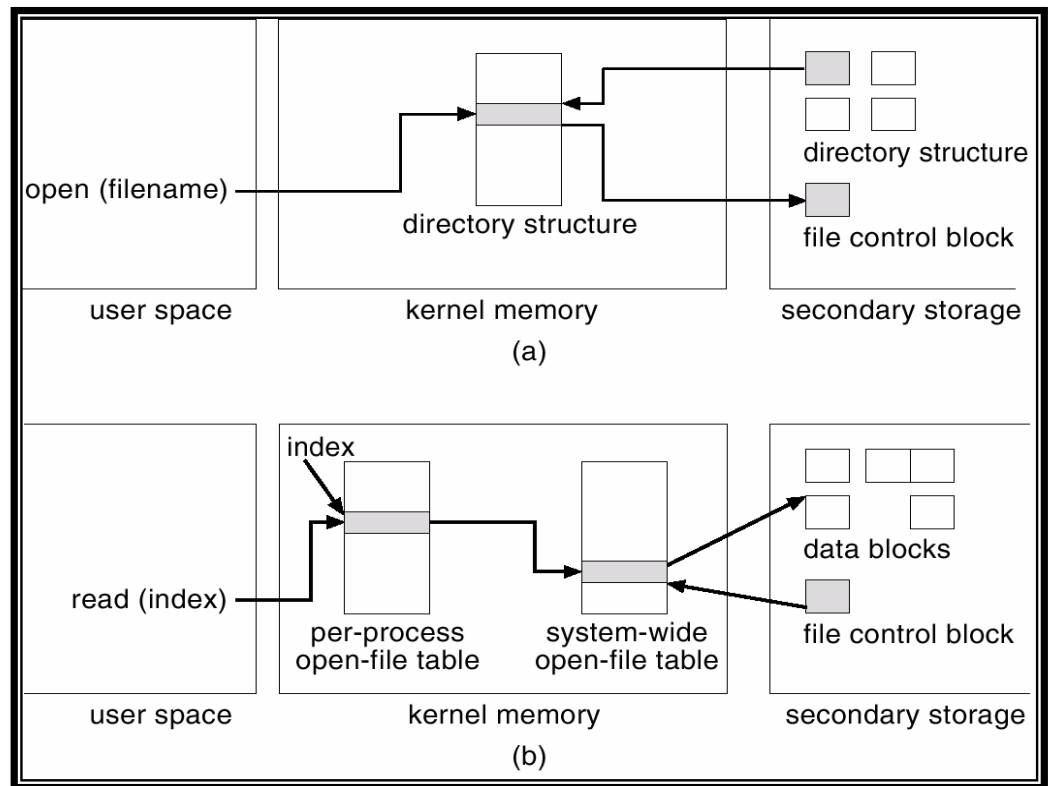


# Cài đặt hệ thống tập tin (2)

## Cấu trúc của hệ thống tập tin trong bộ nhớ

Các ví dụ sau sẽ mô tả sự cần thiết của các cấu trúc hệ thống tập tin được cung cấp bởi hệ điều hành.

- Hình (a) chỉ ra thao tác mở một file.
- Hình (b) chỉ ra việc đọc một file.



# Cài đặt hệ thống tập tin (3)

## Cấu trúc trên đĩa

---

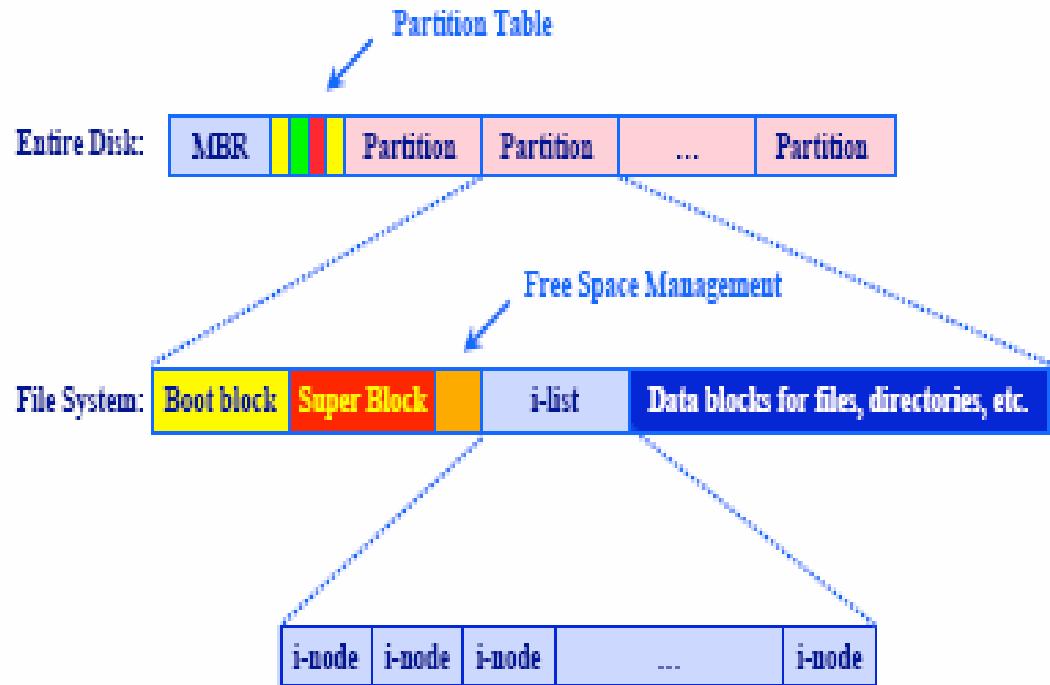
Cấu trúc trên đĩa của hệ thống tập tin bao gồm:

- Khối điều khiển khởi động (Boot control block)
  - Chứa thông tin về cách khởi động hệ điều hành
- Khối điều khiển phân khu (Partition control block)
  - Các chi tiết về phân khu: thông tin về các khối, các con trỏ đến khối và tập tin,
- Cấu trúc thư mục (Directory structure)
  - Cách tổ chức của các tập tin
- Khối điều khiển tập tin (File control block)
  - Chi tiết về tập tin
  - Inode trong UFS

# Cài đặt hệ thống tập tin (4)

## Cấu trúc trên đĩa (tt)

- **MBR (Master Boot Record):** dùng boot máy tính.
- **Partition Table:** địa chỉ bắt đầu và kết thúc của mỗi partition.
- **Boot block:** chứa chương trình đặc biệt phần cứng được gọi một cách tự động để nạp Unix lúc khởi động.
- **Super Block:** dạng và số khối trong hệ thống tập tin.
- **Free space management:** chứa 2 danh sách, số hiệu các khối đĩa dữ liệu trống và số hiệu các i-node trống.
- **i-list/i-node table:** chứa thông tin về tập tin (metadata)



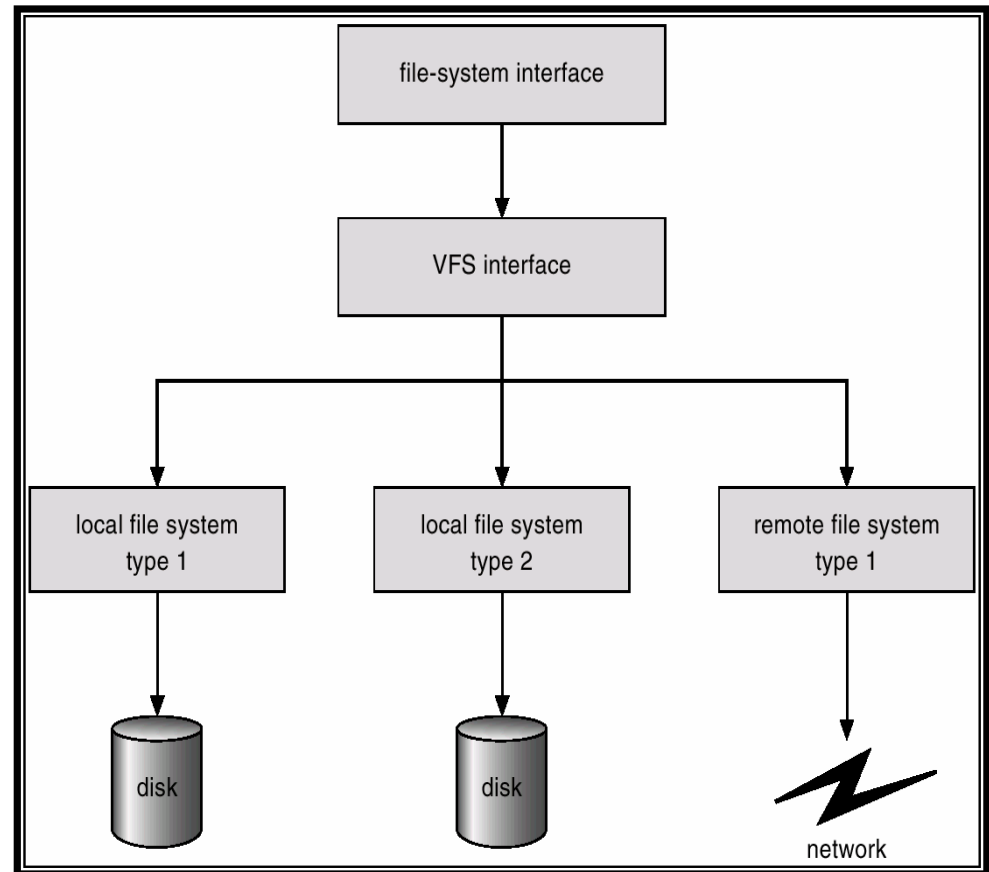
**Cấu trúc đĩa trong Unix FS**



# Cài đặt hệ thống tập tin (5)

## Hệ thống tập tin ảo (Virtual File System)

- Hệ thống tập tin ảo (VFS - Virtual File Systems): là một phương thức cài đặt hệ thống tập tin hướng đối tượng, cho phép cài đặt nhiều hệ thống tập tin.
- VFS cho phép sử dụng cùng một giao diện lời gọi hệ thống (API – Application Programming Interface) cho nhiều loại hệ thống tập tin khác nhau.
- Các hàm API được dùng cho giao diện VFS, hơn là cho một kiểu hệ thống tập tin cụ thể nào đó.



# Cài đặt thư mục (Directory Implementation)

---

- Danh sách tuyến tính (linear list): danh sách tuyến tính của các tên tập tin với con trỏ chỉ đến các khối dữ liệu của tập tin.
  - Dễ lập trình
  - Thực thi mất nhiều thời gian
- Bảng băm (hash table): danh sách tuyến tính của các đầu thư mục (directory entry), nhưng với cấu trúc dữ liệu băm.
  - Giảm thời gian tìm kiếm trên thư mục
  - **Sự đụng độ**: là tình huống mà hai tên tập tin được băm vào cùng một vị trí
  - Kích thước cố định

# Các phương pháp cấp phát đĩa (1)

## (Allocation Methods)

---

- Phương pháp cấp phát đĩa chỉ cách thức mà chúng ta lưu nội dung tập tin vào các khối đĩa như thế nào:
  - Cấp phát kề nhau (Contiguous Allocation)
  - Cấp phát kiểu liên kết (Linked Allocation)
  - Cấp phát kiểu chỉ mục (Indexed Allocation)

# Các phương pháp cấp phát đĩa (2)

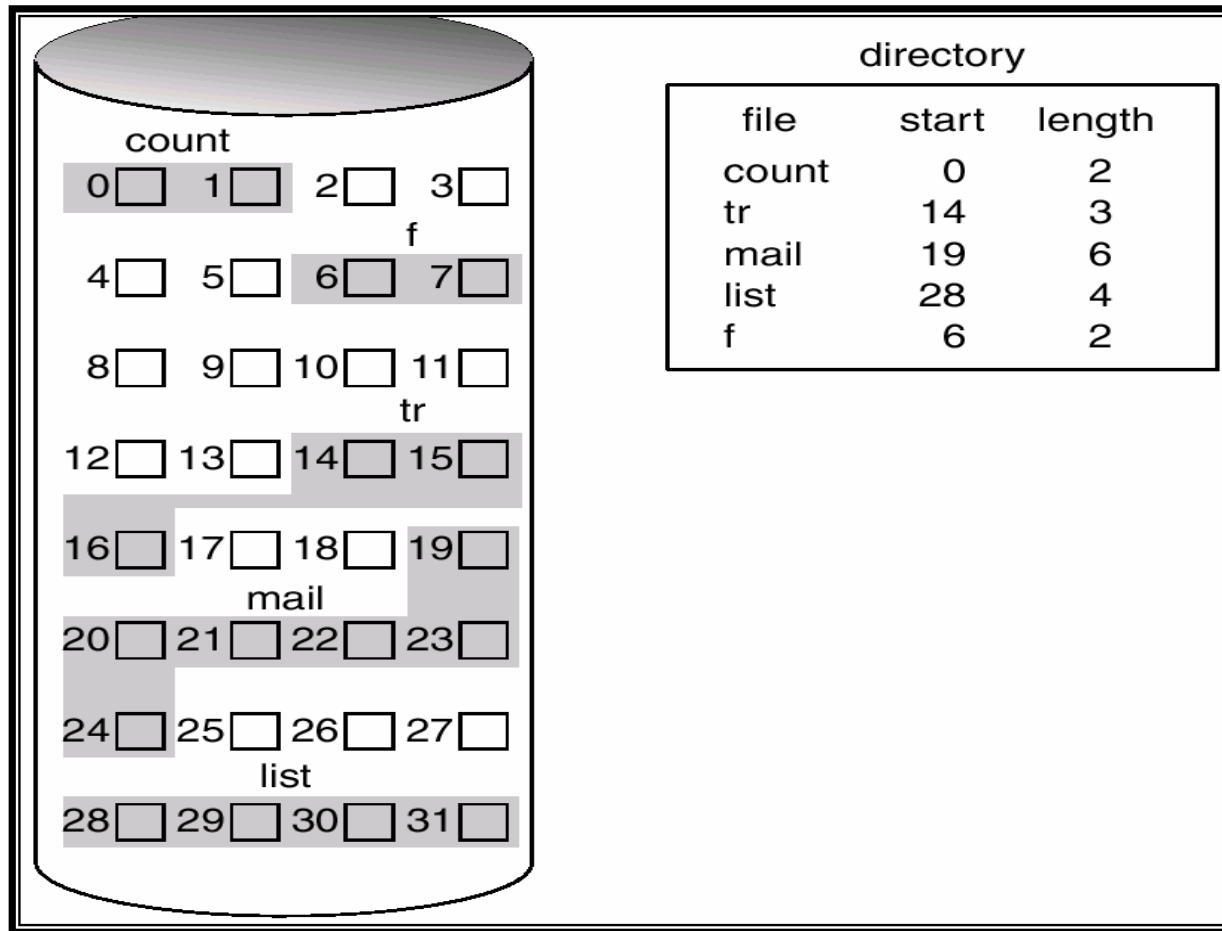
## Cấp phát kề nhau (Contiguous Allocation)

---

- Mỗi file chiếm một tập các khối đĩa liên kề nhau trên đĩa.
- Đơn giản – chỉ cần biết vị trí đầu tiên của file và chiều dài của nó (số lượng khối đĩa).
- Cho phép truy cập cả tuần tự và ngẫu nhiên.
- Lãng phí không gian đĩa (vấn đề cấp phát động), dễ gây phân mảnh ngoài.
- Không thể mở rộng kích thước tập tin.

# Các phương pháp cấp phát đĩa (3)

## Cấp phát kề nhau không gian đĩa



# Các phương pháp cấp phát đĩa (4)

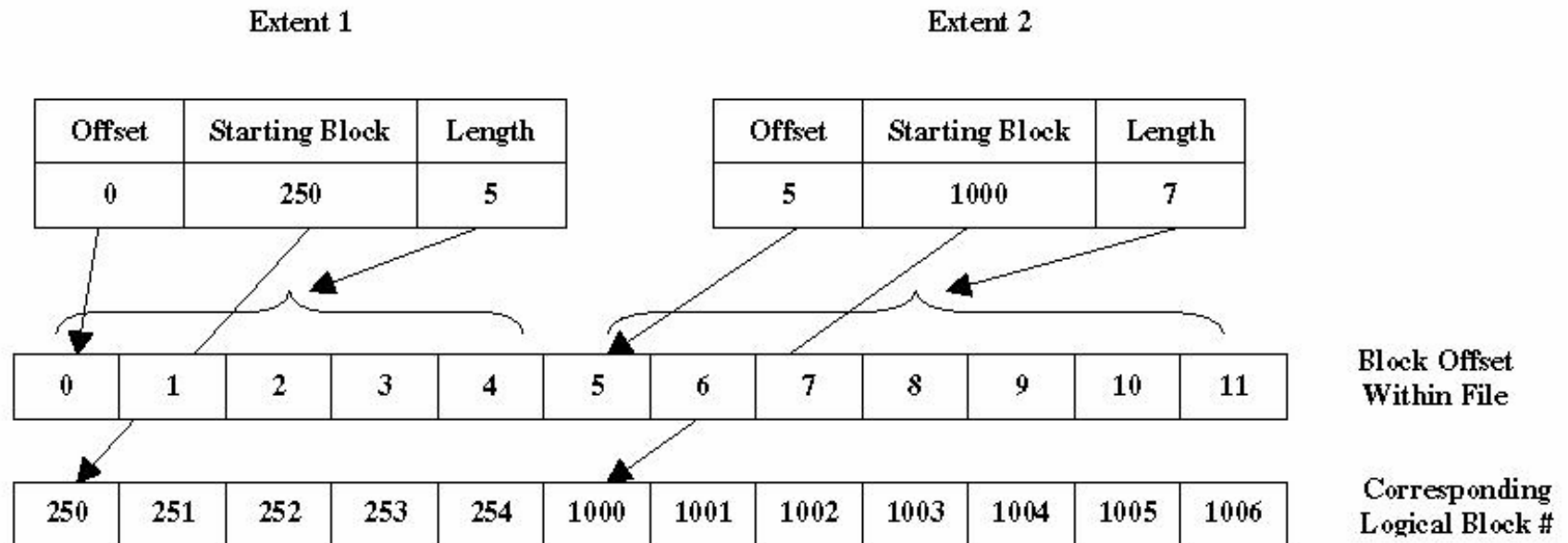
## Hệ thống dựa trên phần mở rộng (Extent-Based System)

---

- Nhiều hệ thống tập tin mới (XFS, Veritas FS (VxFS), NTFS) sử dụng sơ đồ cấp phát kề nhau có sửa đổi
  - Thay vì cấp phát dựa trên các khối đĩa, các hệ thống này cấp phát dựa trên các phần mở rộng, gọi là **extent**.
- Một **extent** chứa nhiều khối đĩa kề nhau
  - Các extent có kích thước (số khối đĩa) khác nhau
  - Một file có thể có một hoặc nhiều extent.
- Vị trí các khối đĩa của file được ghi nhận như vị trí và số khối đĩa (block count), sau đó là liên kết đến khối đĩa đầu tiên của extent kế tiếp.
- Vẫn có thể có phân mảnh trong và phân mảnh ngoài.

# Các phương pháp cấp phát đĩa (5)

## Hệ thống dựa trên phần mở rộng (Extent-Based System)

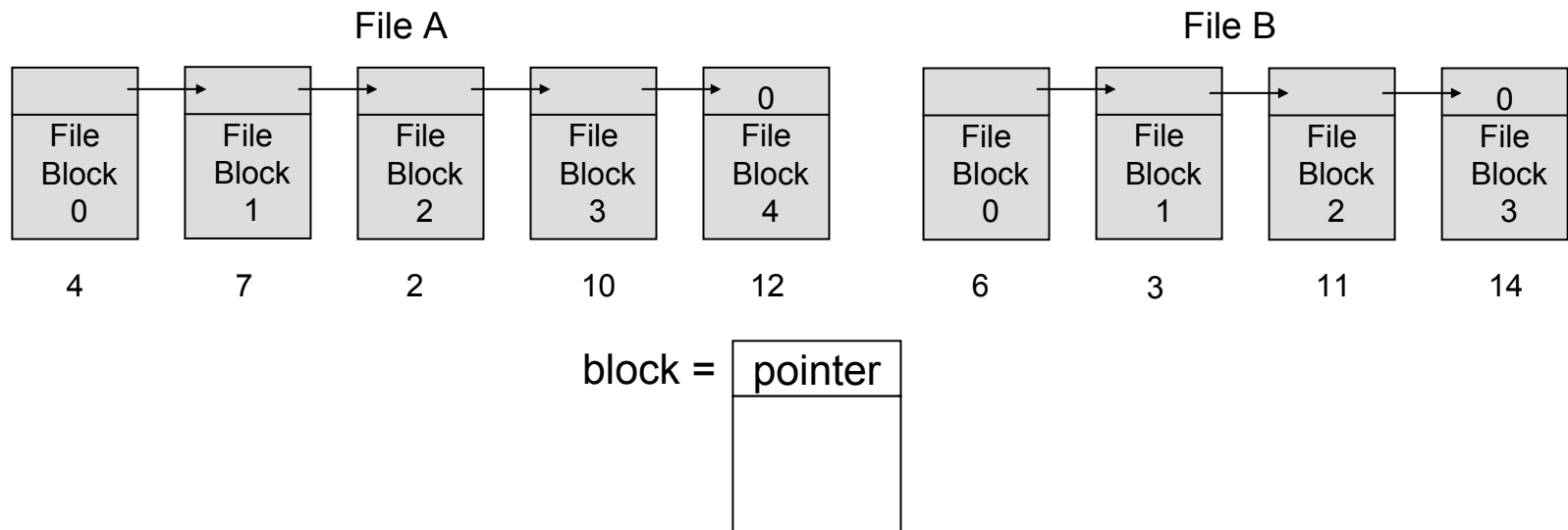


**XFS extent descriptors**

# Các phương pháp cấp phát đĩa (6)

## Cấp phát kiểu liên kết (Linked Allocation)

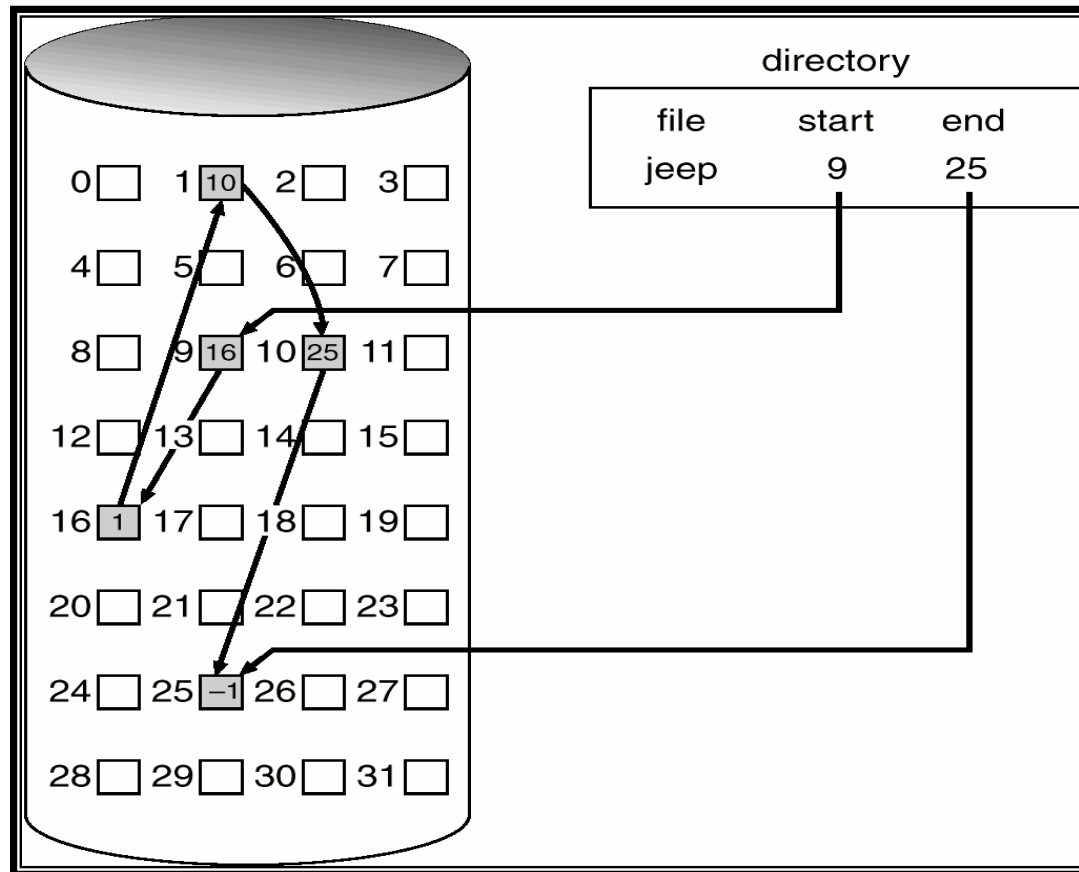
- Một file là một danh sách liên kết các khối đĩa: các khối có thể trải ra ở bất cứ nơi nào trên đĩa.
  - Đơn giản – chỉ cần địa chỉ bắt đầu.
  - Không có lãng phí không gian, tránh phân mảnh.
  - Không thể truy cập ngẫu nhiên.
  - Mất không gian cho pointer trên mỗi khối đĩa.





# Các phương pháp cấp phát đĩa (7)

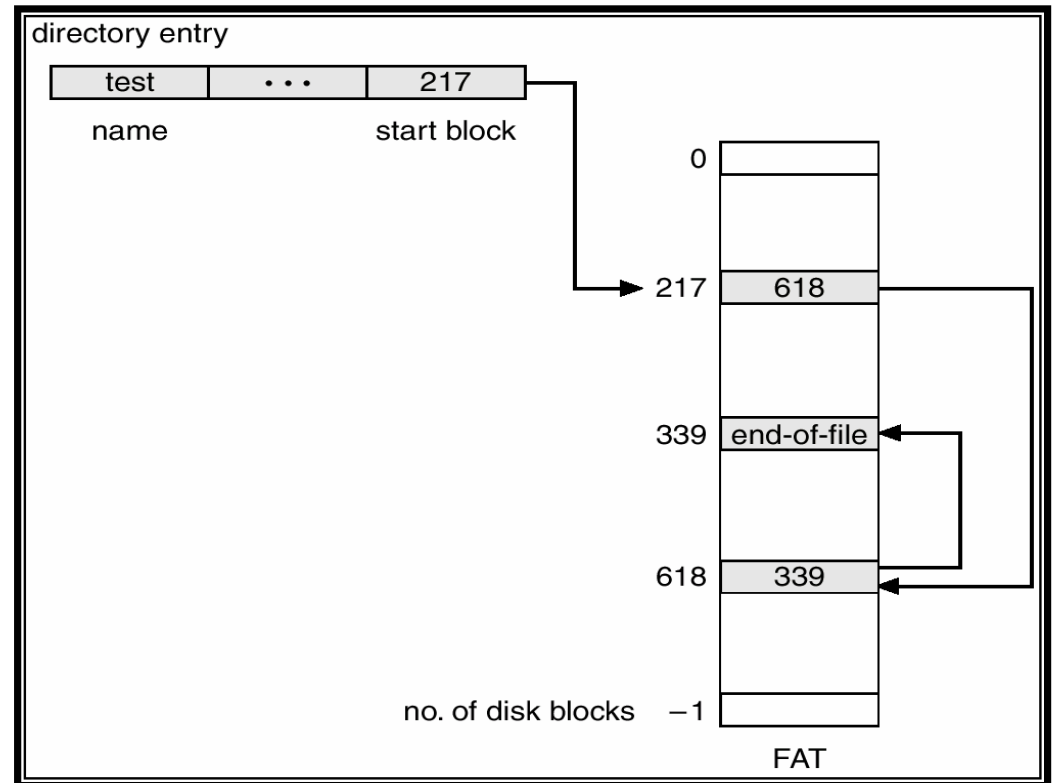
## Cấp phát kiểu liên kết (tt)



# Các phương pháp cấp phát đĩa (8)

## Cấp phát liên kết dùng bảng chỉ mục (File Allocation Table)

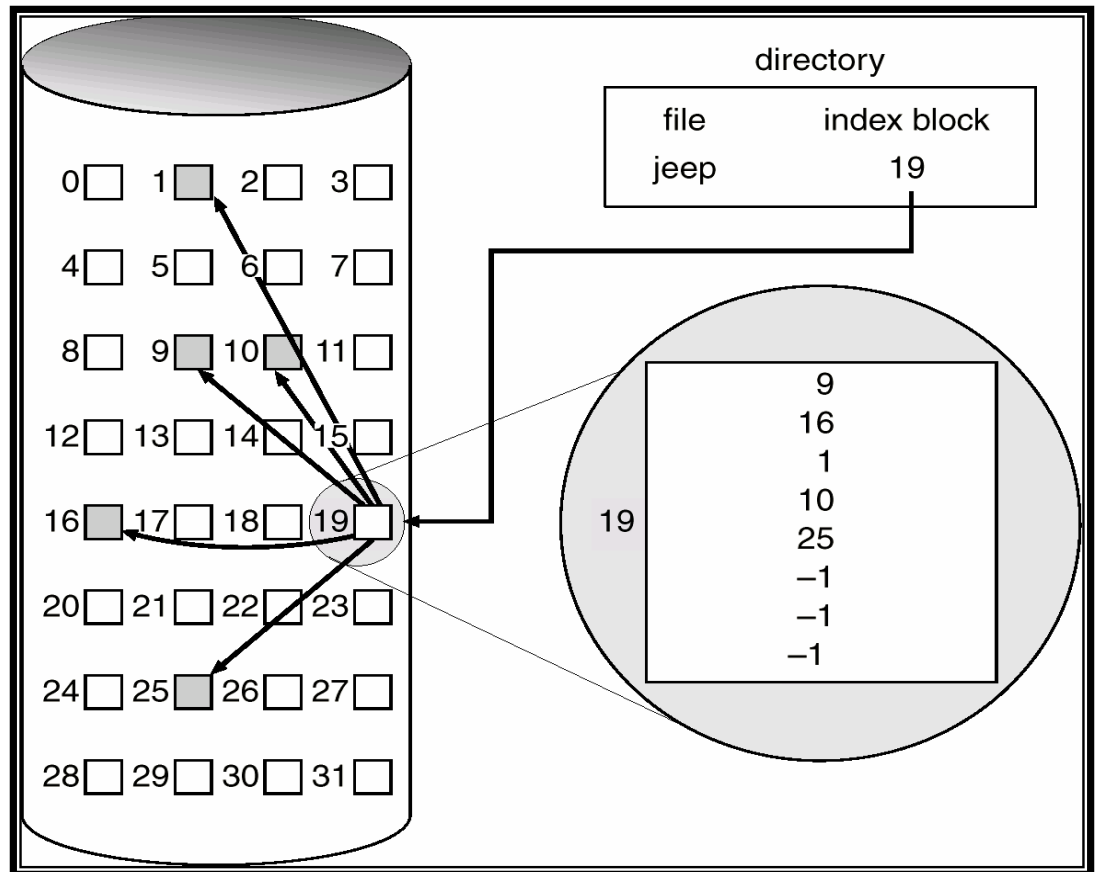
- Các con trỏ đến các khối đĩa của các file được đưa vào một bảng chỉ mục và nạp vào bộ nhớ.
- Bảng có 1 entry cho mỗi khối đĩa, được chỉ mục bằng số hiệu khối đĩa.
- Entry chứa số hiệu khối đĩa của khối kế tiếp trong file
- File-allocation table (FAT) – giải pháp cấp phát đĩa được sử dụng bởi MS-DOS và OS/2



# Các phương pháp cấp phát đĩa (9)

## Cấp phát kiểu chỉ mục (tt)

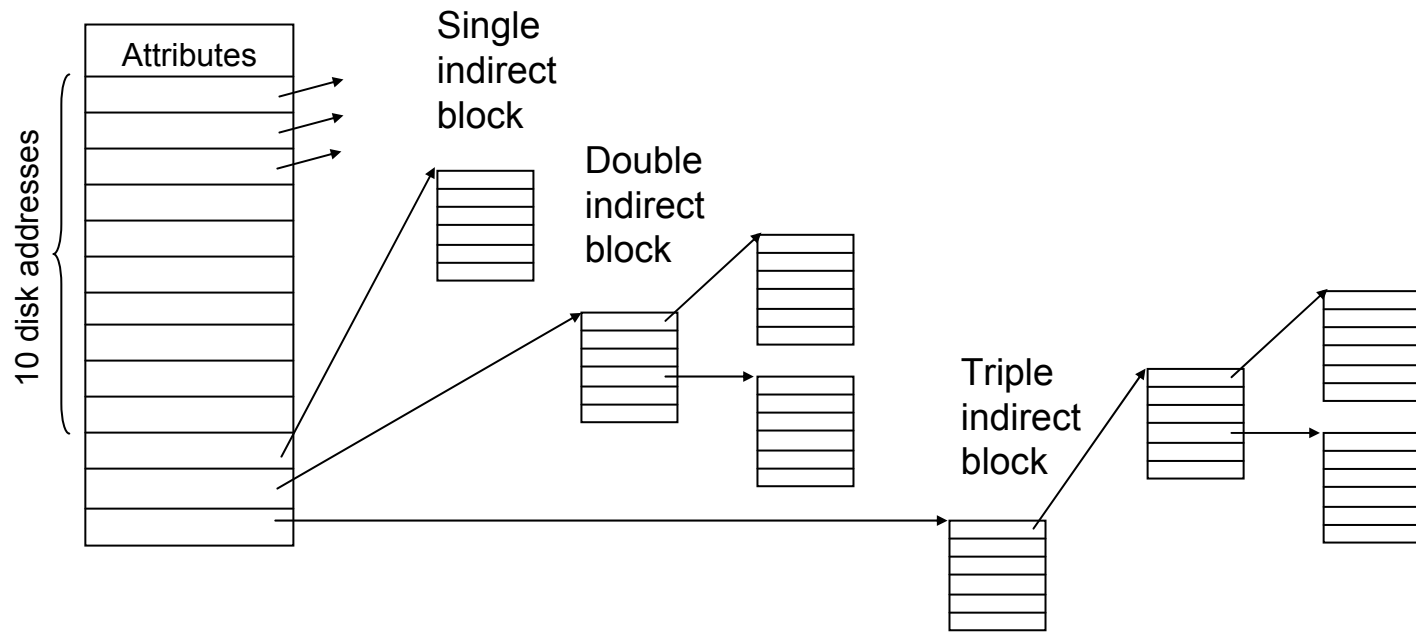
- Mang tất cả các con trỏ (địa chỉ khối) vào một khối, gọi là khối chỉ mục (index block).
- Mỗi tập tin có một khối chỉ mục
- Entry thứ  $i^{\text{th}}$  trong khối chỉ mục chỉ đến khối thứ  $i^{\text{th}}$  của tập tin.
- Directory chứa địa chỉ của khối chỉ mục.



# Các phương pháp cấp phát đĩa (10)

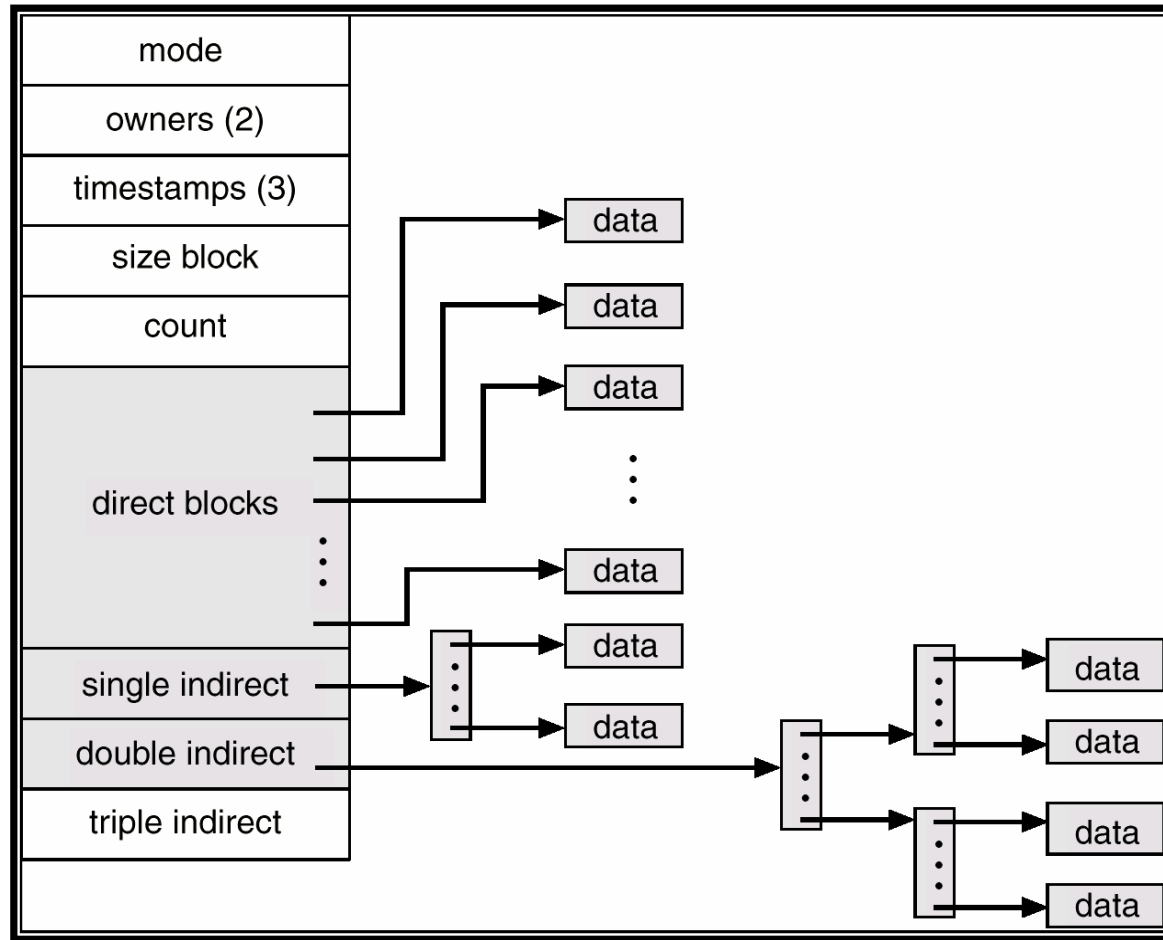
## Cấp phát theo kiểu chỉ mục – Đặc điểm

- Cần bảng chỉ mục
- Truy xuất ngẫu nhiên
- Không bị phân mảnh ngoài, nhưng tốn thêm khối chỉ mục.
- Hiện dùng kiểu kết hợp: bảng chỉ mục chứa khối direct và indirect.



# Các phương pháp cấp phát đĩa (11)

Cấp phát kiểu chỉ mục – Sơ đồ kết hợp UNIX (4K bytes/khối)

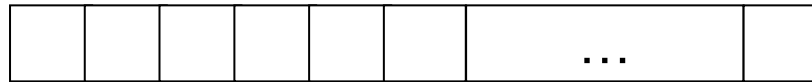


# Quản lý không gian trống (1)

## Bản đồ bit (Bit Map)

---

- Vector bit ( $n$  khối)



$$\text{bit}[i] = \begin{cases} 0 \Rightarrow \text{khối}[i] \text{ trống} \\ 1 \Rightarrow \text{khối}[i] \text{ bị chiếm} \end{cases}$$

- Bản đồ bit yêu cầu thêm không gian lưu trữ.  
Ví dụ:

block size =  $2^{12}$  bytes

disk size =  $2^{30}$  bytes (1 gigabyte)

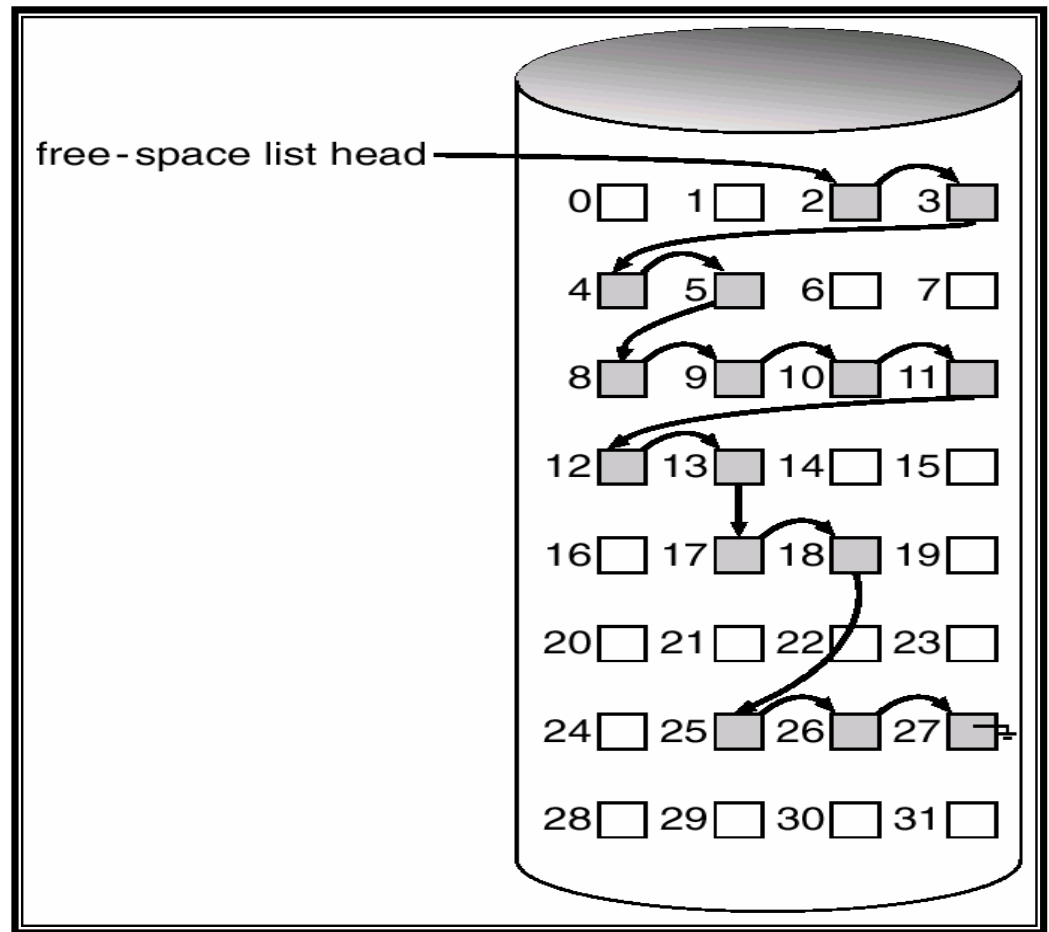
$n = 2^{30}/2^{12} = 2^{18}$  bits (or 32K bytes)

- Dễ dàng để cấp phát không gian liên tục

# Quản lý không gian trống (2)

## Danh sách liên kết (Linked List/Free List)

- Danh sách liên kết (danh sách trống)
  - Nối các khối trống lại với nhau bằng danh sách liên kết.
  - Giữ con trỏ đến khối đĩa trống đầu tiên tại một vị trí trên đĩa và được nạp vào bộ nhớ.
  - Không thể cấp phát liên tục một cách dễ dàng
  - Không có lãng phí không gian



# Quản lý không gian trống (3)

## Nhóm và Đếm danh sách trống

---

- Nhóm (Grouping)
  - Một biến thể của danh sách trống.
  - Lưu giữ danh sách liên kết của các **khối chỉ mục (index blocks)**
  - Mỗi khối chỉ mục lưu địa chỉ của n khối trống trong khối đầu tiên và một con trỏ đến khối chỉ mục kế tiếp.
  - Cho phép cung cấp địa chỉ của một số lượng lớn các khối trống 1 cách nhanh chóng.
- Đếm (Counting): dùng cho các khối trống kề nhau
  - Mỗi mục từ của danh sách lưu địa chỉ khối trống đầu tiên và số lượng n khối trống kế tiếp.



# Hiệu suất và hiệu năng (1)

---

- Hiệu suất phụ thuộc vào:
  - Các giải thuật cấp phát đĩa và thư mục
  - Kiểu dữ liệu được giữ trong đầu mục của thư mục
- Hiệu năng
  - Disk cache – một khu vực bộ nhớ riêng biệt trong bộ nhớ chính dùng trữ tạm các khối đĩa được sử dụng thường xuyên nhất
  - Free-behind và Read-ahead – các kỹ thuật để tối ưu hóa việc truy xuất tuần tự.
  - Ram disk: cho phép thực hiện các thao tác trên đĩa, nhưng thực hiện trên bộ nhớ, thường được dùng để làm kho lưu trữ tạm cho các file.
  - Disk cache được quản lý bởi người dùng, ram disk bởi OS.

# Hiệu suất và hiệu năng (2)

## Nhiều vị trí để trữ tạm nội dung đĩa

