

# 第八章 磁盘存储器的管理

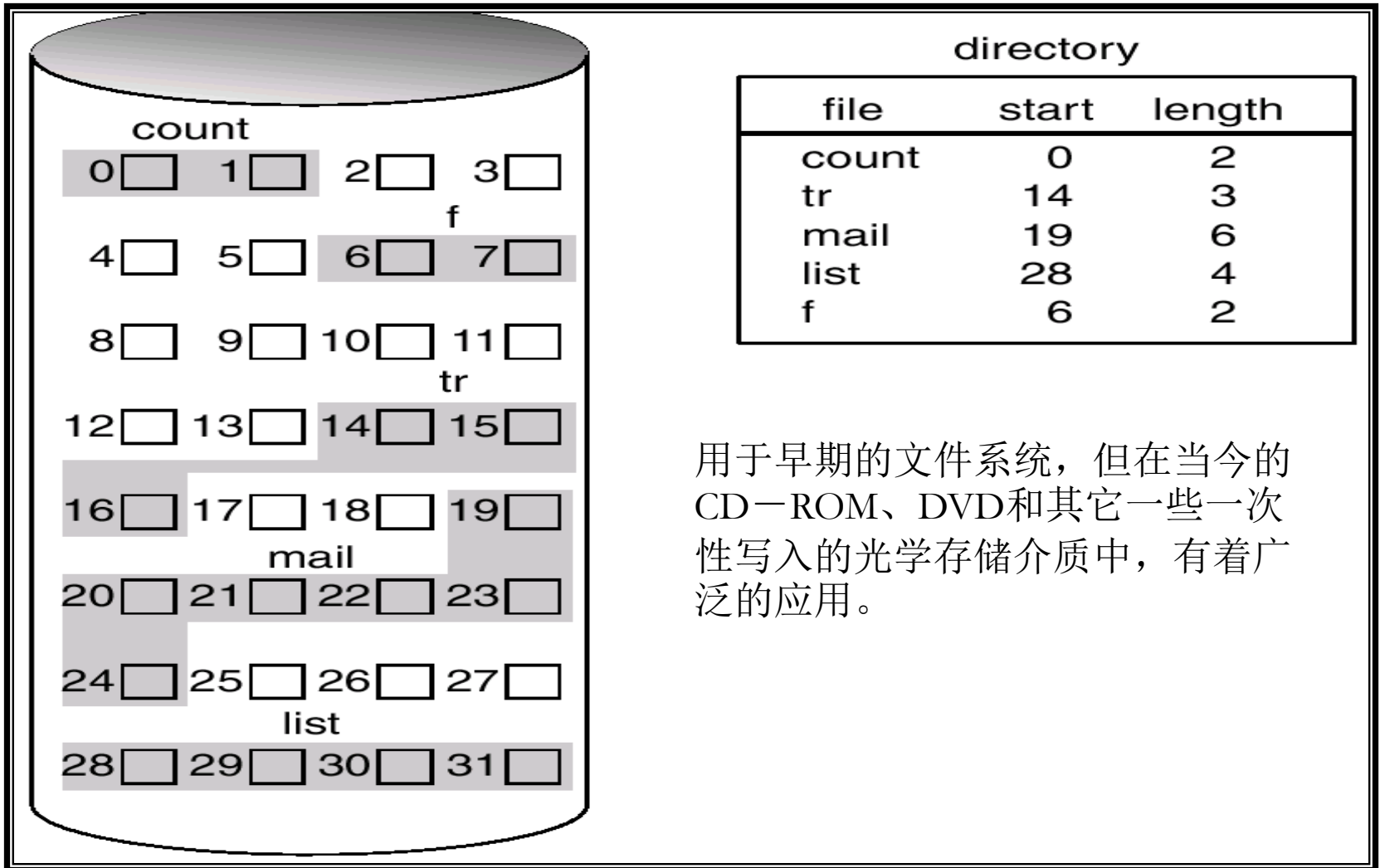
# 8.1 外存的组织方式

有效利用外存空间

提高文件访问速度

连续分配、链接分配、索引分配

# 8.1.1 连续分配



# 8.1.1 连续分配

- 1 连续分配方式
  - 为每个文件分配一组相邻的盘块
  - 分配给文件的首个物理块的地址登记在目录项
- 2 特点
  - 顺序访问容易
  - 要求有连续的存储空间
  - 必须事先知道文件的长度，不利于动态增长
  - 文件增删会产生碎片，需移动文件以利用碎片空间

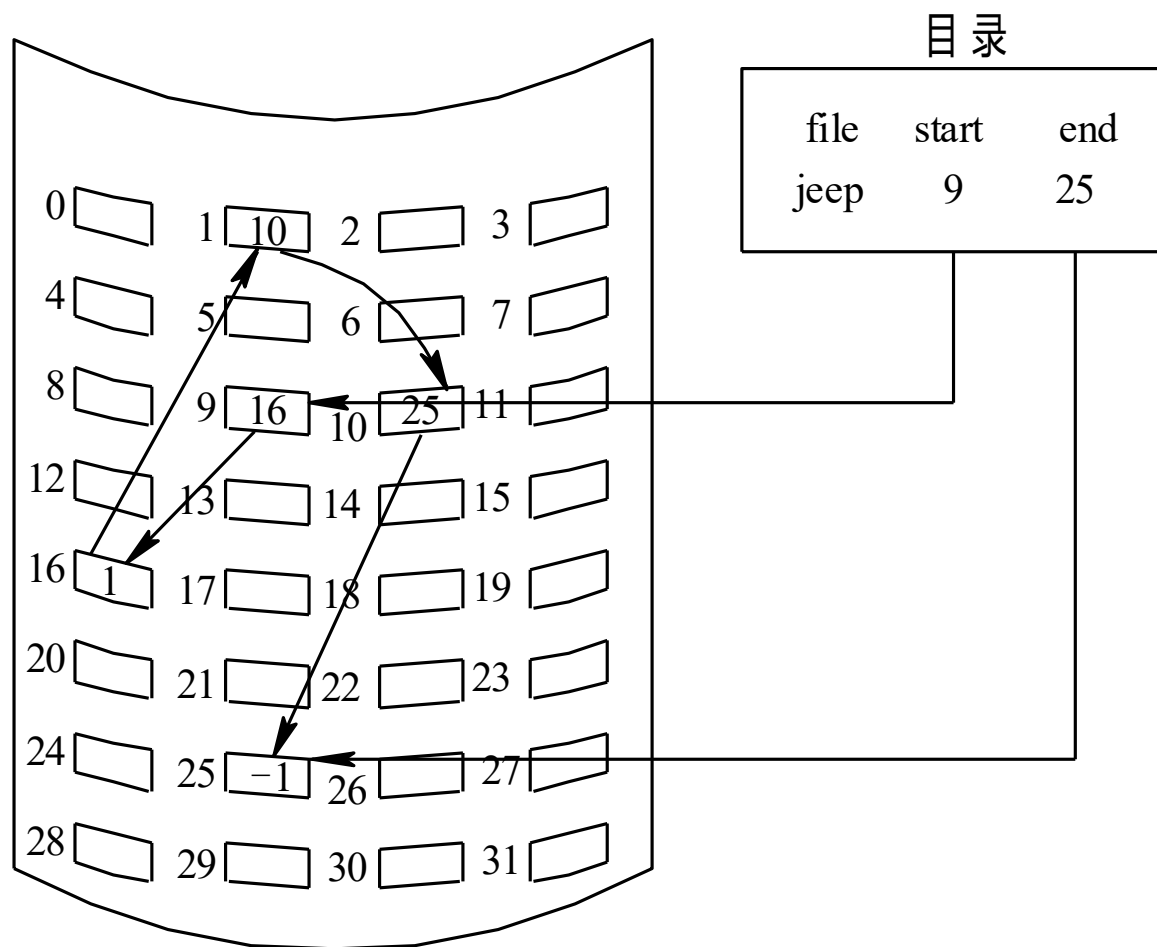
## 8.1.2 链接分配

- 1. 隐式链接

块结构

指针

数据



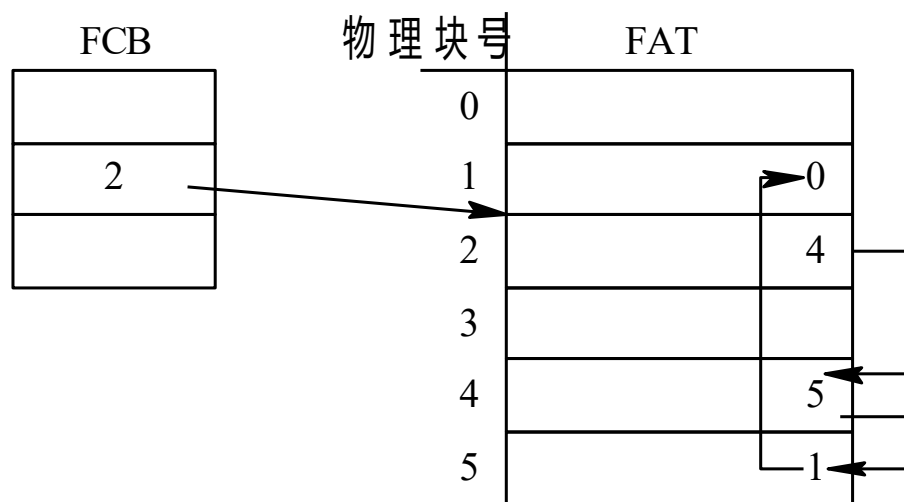
## 8.1.2 链接分配

- 1. 隐式链接
  - 每个盘块中有指向下一个盘块的指针；
  - 目录项有指向链接文件第一个盘块和最后一个盘块的指针；
  - 文件长度可以动态增长、消除了外部碎片
  - 只适合顺序访问、指针错误导致文件损坏，可靠性差
  - 每个物理块上的数据存储空间不再是2的整数次幂（指针占用了若干个字节），从而使文件的访问不太方便。

如何改进

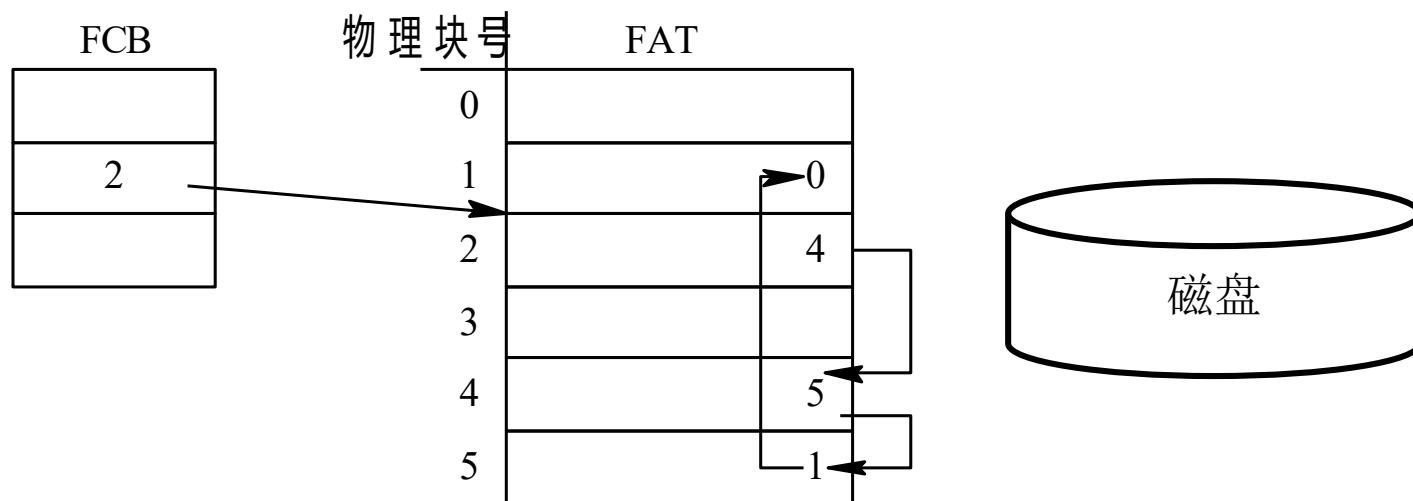
- 2. 显式链接

- 把链接指针统一存放在一张表中；
- 表项的序号为物理盘块号，每个表项中存放链接指针；
- 该表整个磁盘设一张,称为文件分配表 **File Allocation Table**



## 8.1.3 FAT 和 NTFS

- 文件分配表FAT（File Allocation Table）：文件保存在离散的物理磁盘块上，采用显示链法进行文件组织，每张磁盘设置一张文件分配表，表项的序号为物理盘块号，每个表项中存放链接指针。





## FAT12

以盘块（扇区）为基本分配单位；

最大容量

4096（FAT表项数目）

× 512B（扇区大小）

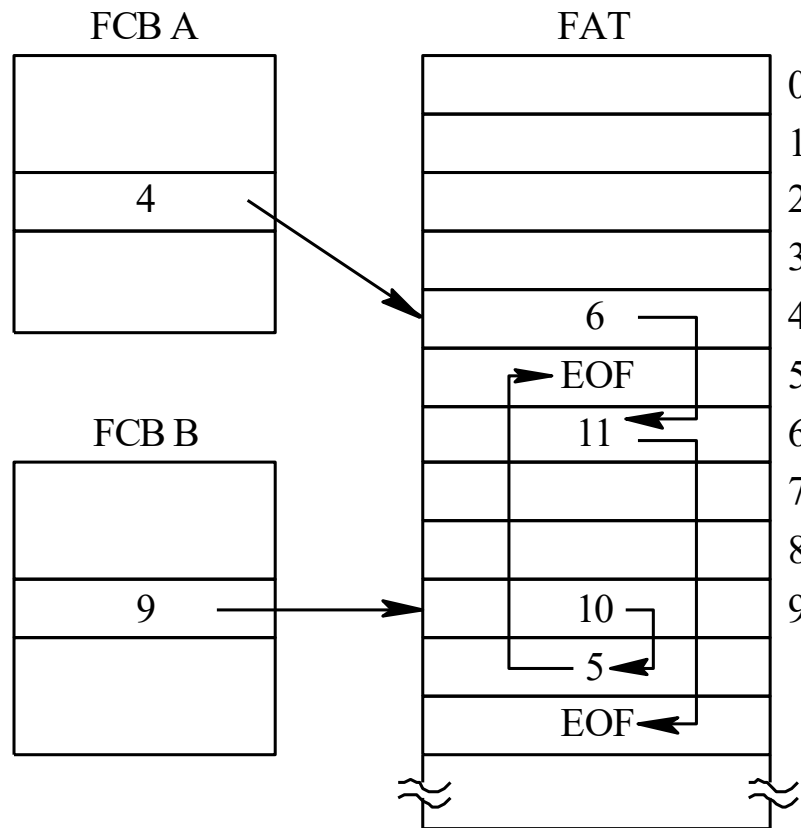
= 2M

簇

一组连续的扇区，作为基本单位

容量有限、碎片； 8+3格式文件名

- MS-DOS的文件物理结构
- 文件A: 4、6、11;
- 文件B: 9、10、5;



- FAT16
  - FAT表16位
  - 最大容量：  $2^{16} * 64$  (每簇的扇区数) \* 512 = 2048MB
  - 簇内碎片大
- FAT32
  - 簇固定大小为4KB。
  - 最大容量：  $2^{32} * 4KB = 2\text{ TB}$
  - FAT大，导致速度慢；不支持小于512M 分区；单个文件不能大于4GB；不能向下兼容

# 8.1.4 NTFS文件

- 4 NTFS
  - 64位磁盘地址、支持长文件名。
  - 磁盘组织
    - 簇为基本单位
    - 簇大小可选择
  - 文件组织
    - 主控文件表

**MFT Record for a Small File or Directory:**



# 8.1.5 索引分配

- 1. 单级索引分配

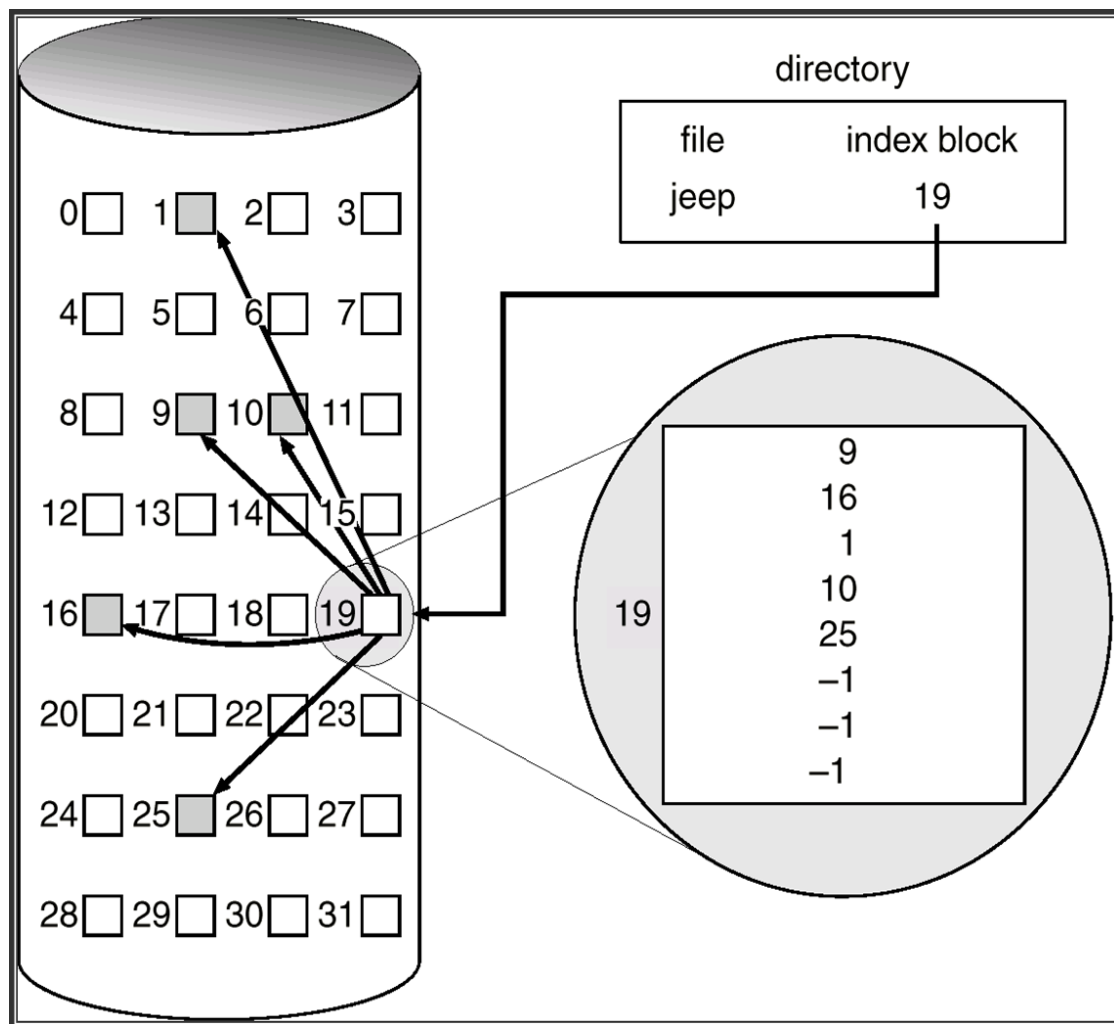
- 链接分配方式

- 1) 不能支持高效的直接存取，要对于一个较大的文件进行直接存取，须首先在FAT中顺序地查找许多盘块号。
    - (2) FAT需占用较大的内存空间。

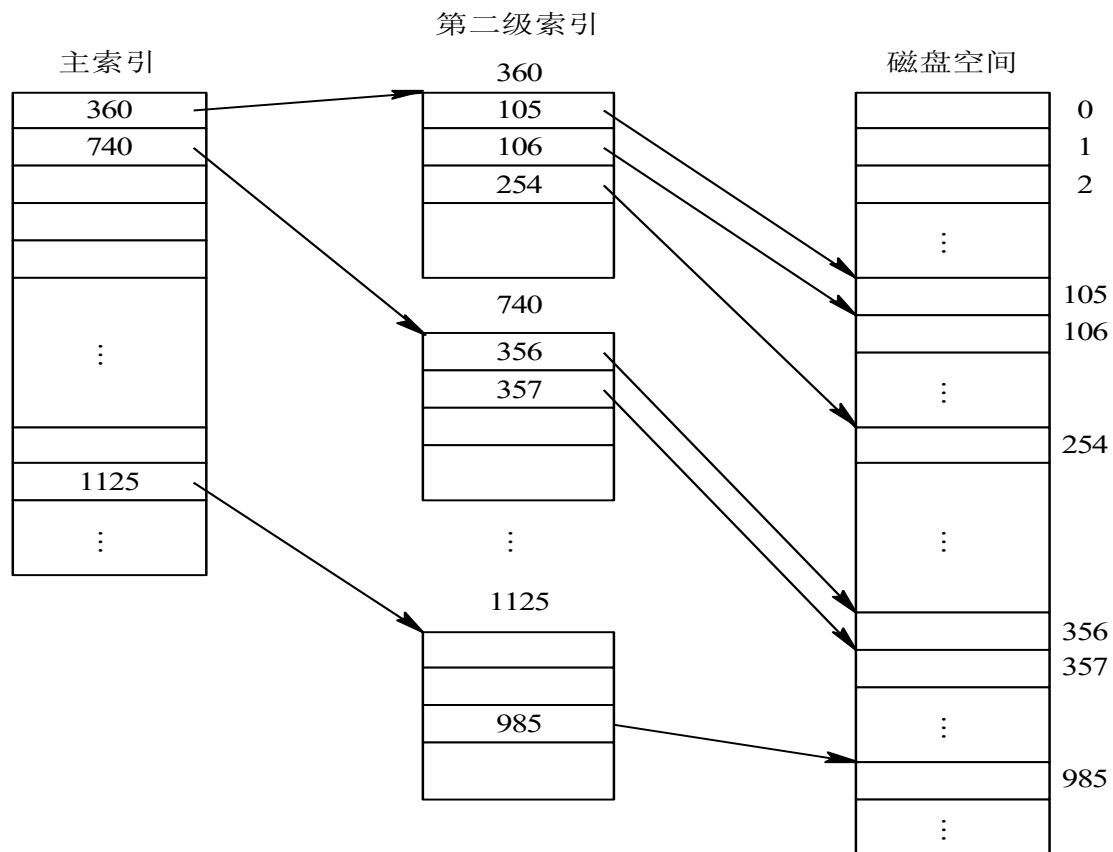
- 基本思想

- 每个文件有一个索引块（表），将分配给该该文件的所有盘块号都记录在该索引块
    - 索引表的指针记录在文件目录

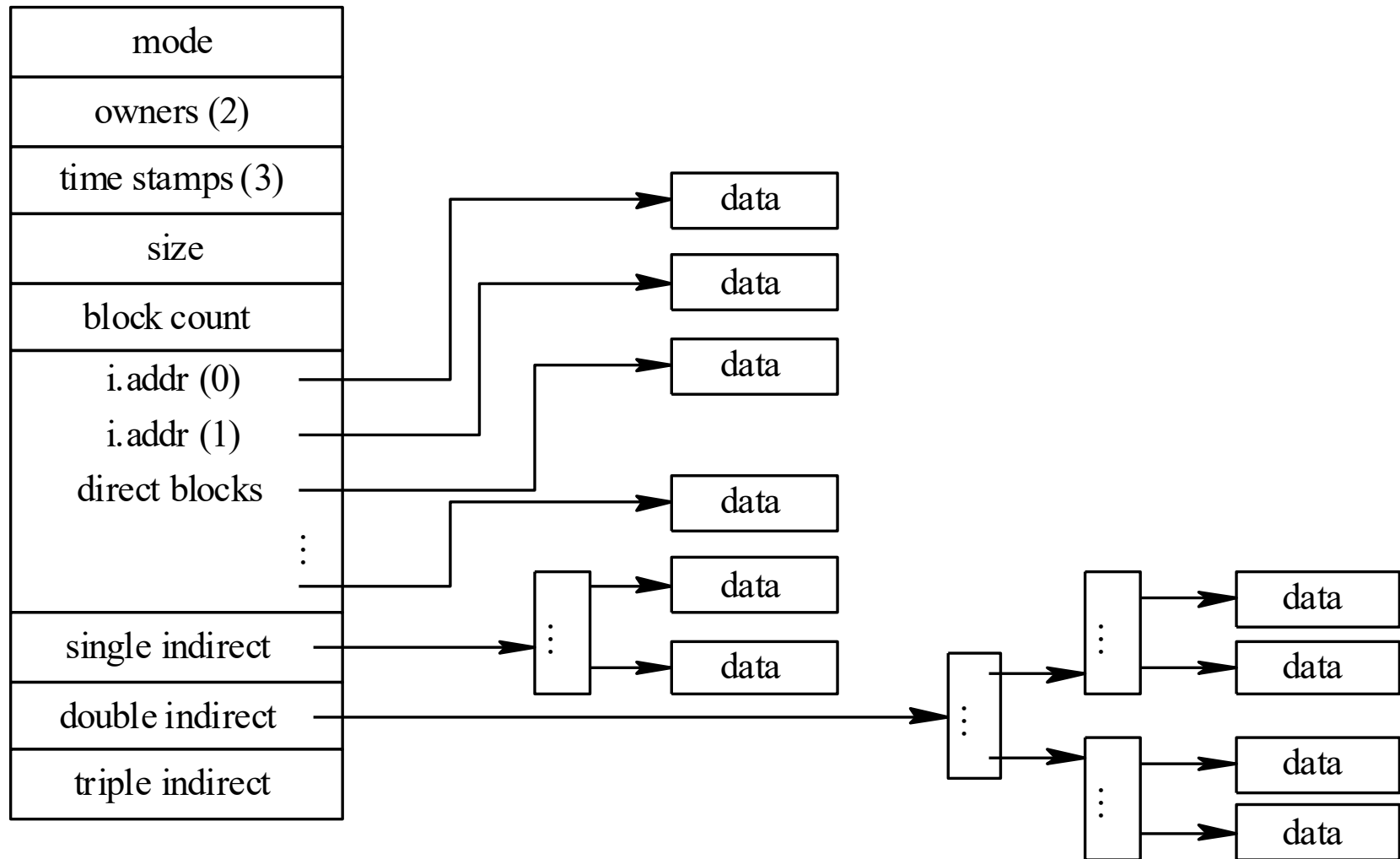
- 支持直接访问
- 需要外存空间



- 2. 多级索引分配
  - 大文件使用多个索引块，为索引块建立索引



# 混合索引(增量索引)





## 8.2 文件存储空间的管理

记录存储空间的使用情况

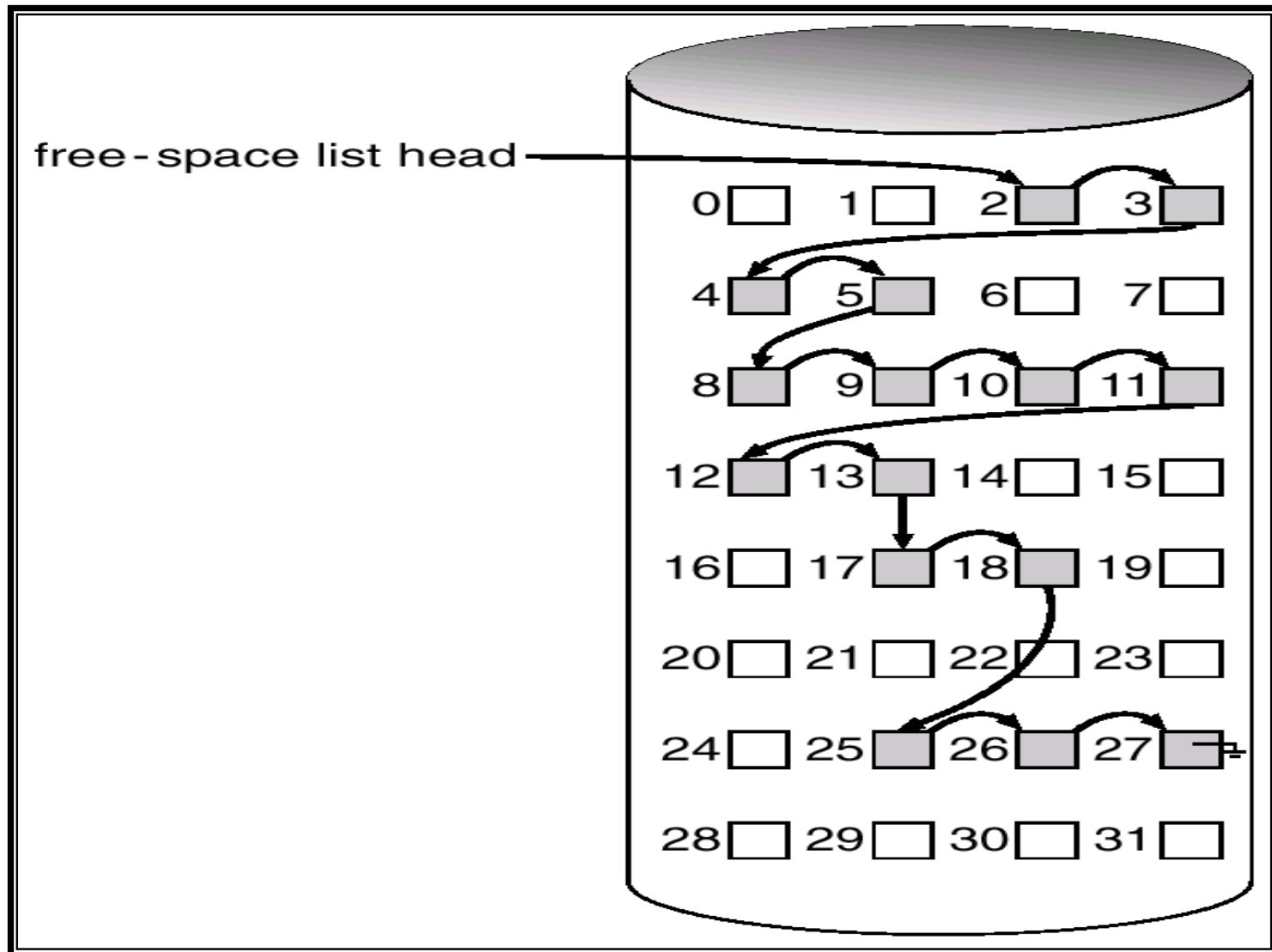
提高对存储空间分配和回收的手段

## 8.2.1 空闲表法和空闲链表法

- 1. 空闲表法
  - 为文件分配连续的存储空间
  - 类似内存的动态分配方式
  - 系统为外存上的所有空闲区建立一张空闲表
- 2. 空闲链表法
  - 空闲盘区链接在一起
    - 空闲盘块链
    - 空闲盘区（可能包括若干盘块）链

序号	第一空闲盘块号	空闲盘块数
1	2	4
2	9	3
3	15	5
4	—	—

# 空闲盘块链



### 8.2.2 位示图法

- 用一个二进制位来表示磁盘中一个盘块的使用情况

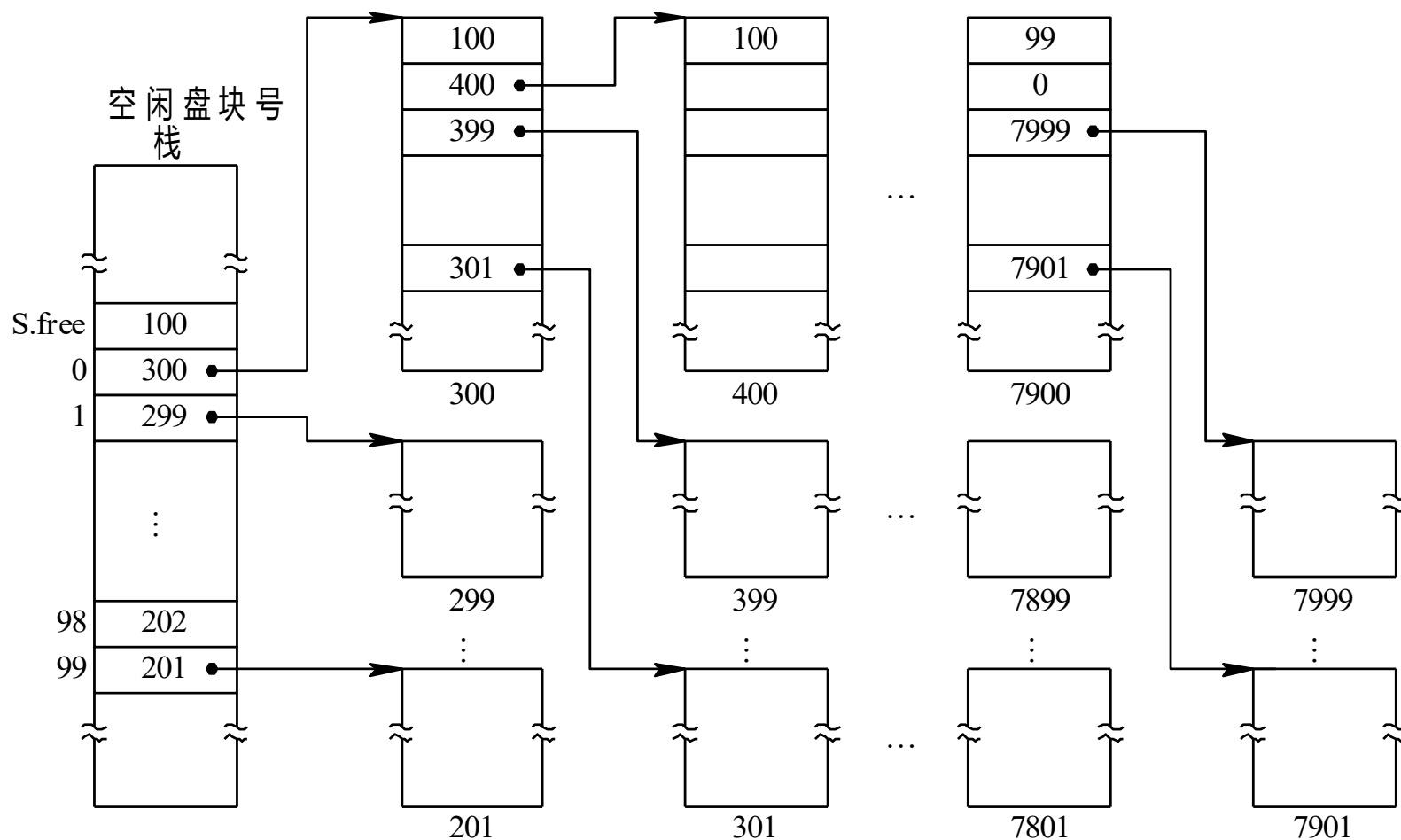
[illegible]

## 6.5.2 位示图法

- 盘块数量:  $m*n$
- 盘块号  $b=n(i-1)+j$  //位示图位置到盘块号
- 位置:  
$$i = (b-1) / n + 1$$
$$j = (b-1) \% n + 1$$

## 8.2.3 成组链接法

结合使用空闲表法和空闲链表法  
适合大型文件，Unix



空闲盘块的成组链接法

## 8.3 提高磁盘I/O速度的途径

# 1 磁盘高速缓存(Disk Cache)

- 利用内存暂存从磁盘中读出的信息。
- 高速缓存是一组在逻辑上属于磁盘，而物理上是驻留在内存中的盘块。



# 1 磁盘高速缓存(Disk Cache)

- 数据交付方式
  - (1) 数据交付。这是直接将高速缓存中的数据，传送到请求者进程的内存工作区中。
  - (2) 指针交付。只将指向高速缓存中某区域的指针， 交付给请求者进程。
    - 所传送的数据量少，节省了数据从磁盘高速缓存存储空间到进程的内存工作区的时间

# 1 磁盘高速缓存(Disk Cache)

- 置换算法

- 磁盘中盘块数据欲调入高速缓存，如缓存满，需要将缓存中数据换出（LRU 及类似算法）
- (1) 访问频率低于请求调页的联想存储器
- (2) 可预见性高于请求调页的联想存储器
- (3) 数据的一致性

- 周期性地写回磁盘

- 程序周期性地强制性地将所有在高速缓存中已修改的盘块数据写回磁盘,减少风险

## 2 提高磁盘I/O速度的其它方法

- 1.提前读(Read-Ahead)
  - 顺序访问文件
- 2. 延迟写
- 3. 优化物理块的分布
- 4. 虚拟盘

# 3 廉价磁盘冗余阵列

- Redundant Array of Inexpensive Disk
  - 用一台磁盘阵列控制器统一管理和控制一组磁盘驱动器，组成可靠快速的大容量磁盘系统
- 1. 并行交叉存取
  - 多台磁盘驱动器
  - 每一盘块的数据分为若干个子盘块
  - 每一个子盘块的数据分别存储在不同磁盘
  - 传输某盘块数据到内存时，各盘块中的子盘块同时向内存传输

# RAID

- 2. RAID的分级
  - (1)RAID 0级
    - 并行交叉存取
  - (2) RAID 1级
    - 磁盘镜像
  - (3) RAID 3级
    - 一台奇偶校验盘完成数据校验
  - (4) RAID 5级
    - 每个驱动器都有自己独立的数据通路
  - (5) RAID 6级和RAID 7级
    - 设置专用可快速访问的异步校验盘

- 3. RAID的优点
  - (1) 可靠性高。
  - (2) 磁盘I/O速度高
  - (3) 性能/价格比高