



# § 1 基本概念

## 一、文件(file)的定义

文件是由创建者所定义、由文件名标识的一组相关信息集合。

——**文件体**：文件本身的信息元素集合

——**文件属性**：OS为管理文件为之配备的各类说明和描述信息。诸如文件的组织、存储、共享、保密与保护等信息。

### 基本信息类：

- 文件的物理地址
- 文件的长度
- 文件的类型
- 文件的逻辑结构(用户可见结构)
- 文件的物理结构(不可见结构)

### 存取控制类：

- 文件主和其它用户对该文件的访问权限

### 管理信息类：

- 文件创建的日期和时间
- 最近修改该文件的日期和时间
- 当前打开该文件的进程数





# § 1 基本概念

## 二、文件分类

1.按性质和用途:系统文件、用户文件、库文件

2.按文件中数据的形式:源文件、目标文件、可执行文件

3.按存取属性:只执行文件、只读文件、读写文件

4.按组织形式和处理方式(Unix/Linux)

—普通(正规)文件:用户信息或系统信息构成的文件。ls -l命令显示其类型为-

—目录文件:为文件构造的目录信息所形成的文件。ls命令显示其类型为d

—特殊文件:表示I/O设备的文件。ls命令分别显示为c、b、p表示字符设备、块设备和FIFO管道文件



# § 1 基本概念

## 三、文件基本操作(文件类系统调用)

### 1.最基本的文件操作

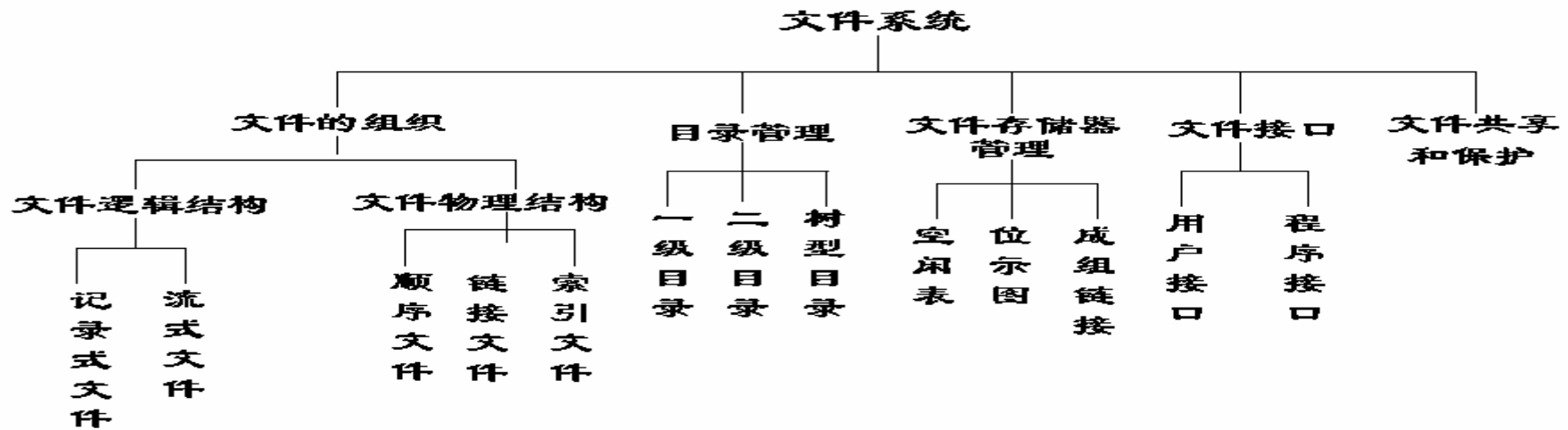
- **创建文件:** 为新文件分配必要外存空间, 并在文件目录中为之建立一个目录项以描述该文件的属性。
- **删除文件:** 先删除指定目录项使其成为空项, 再回收文件空间。
- **读/写文件:** 根据用户给定的文件名查找到目录, 从中获得文件位置指针, 进行读/写文件操作。
- **在文件中重定位:** 设置文件的读/写位置, 实现随机存取。

### 2.文件的“打开”和“关闭”操作

- **打开(open)文件:** 将指名文件的属性从外存拷入内存, 并在用户进程PCB中返回一个编号(称文件描述符或文件句柄)。据此, 在用户进程和指定文件间建立起一个连接, 以提高文件操作速度。
- **关闭(close)文件:** 删除内存文件的属性, 并释放文件描述符。据此, 断开用户进程和指定文件间的连接。



## § 2 文件管理功能



**1.文件的组织：**使信息能从逻辑上构造成文件并加以存放

— 文件逻辑结构:用户按逻辑结构将其信息组织成文件

— 文件物理结构:系统按物理结构存储文件中的信息

**2.目录管理：**建立合理目录结构，实现文件“按名存取”

**3.文件存储器管理：**记录外存使用情况并实现分配与回收

**4.文件接口:**文件操作有关的用户接口、与文件编程相关编程接口

**5.文件共享和保护：**文件的物理存储只有一处，允许用不同路径名(别名)将其打开，并提供文件的保护和保密措施。



## § 3 文件逻辑结构

**文件的逻辑结构**是指从用户角度所观察到的，文件中信息的组织方式。该结构独立于文件的物理存储，是用户可见的文件面貌。由逻辑结构组织形成的信息集合称为**逻辑文件**。

**1.记录式文件：**是指由若干逻辑记录所构成的文件，依次可定义为记录0，记录1.....记录n。记录式文件是**有结构文件**。

—— 逻辑记录是文件系统读写文件的最小单位

**2.流式文件：**是相关字节流的序列，它在逻辑上从0字节到最大字节，其间不再区分记录。是**无结构文件**。

—— 利用读/写指针，指定起始地址和字节长度进行访问

简历文件

学号	姓名	年龄	政治面貌	家庭地址







## § 3 文件逻辑结构

- 例 (西安交大):
- (1) 一个**顺序访问**文件有固定长度为15Byte的记录。假设第1个记录是记录1, 求记录5第1个字节所在的逻辑位置
  - (2) 一个**随机访问**的文件有固定长度为15Byte的记录。假设第1个记录是记录1, 求记录5的第1个字节所在的逻辑位置。
  - (3) 一程序刚从一**直接访问**文件中读取了第1个记录, 接着要读第10个记录, 问: 该程序要读多少个记录才能读入第10个记录?
  - (4) 一程序刚从一**顺序访问**文件中读取了第1个记录, 接着要读第10个记录, 问: 该程序要读多少个记录才能读入第10个记录?
  - (5) 一程序刚从一**顺序访问**文件中读取了第10个记录, 接着要读第6个记录, 则该程序需访问多少个记录才能将第6个记录读入?

(1) 60字节处

(2) 60字节处

(3) 1个, 即记录10

(4) 9个, 即记录2~10

(5) 6个, 即记录1~6



## § 4 目录管理技术

### 一、文件控制块FCB

文件名及其文件属性合称**File Control Block**, FCB是OS为管理文件而设置的数据结构, 是文件存在的标志。

#### FCB的组成

- 文件名
- 文件属性(基本信息类)
  - 文件的物理地址
  - 文件的长度
  - 文件的类型
  - 文件的逻辑结构
  - 文件的物理结构
- 文件属性(存取控制类)
  - 文件主和其它用户对该文件的访问权限
- 文件属性(管理信息类)
  - 文件创建的日期和时间
  - 文件最近修改日期和时间
  - 当前打开该文件的进程数



## § 4 目录管理技术

### 一、文件控制块FCB

文件名及其文件属性合称**File Control Block**, FCB是OS为管理文件而设置的数据结构, 是文件存在的标志。

### 二、文件目录&目录文件

- ▶ 文件系统中所有FCB的有序集合称作该文件系统的文件目录, 一个FCB是其中的一个目录项
- ▶ 将**文件目录**以文件形式保存在外存, 这种实现文件“**按名存取**”的文件称目录文件。

### 三、目录管理的功能

- ▶ **实现文件的“按名存取”**
- ▶ 目录的构造应考虑提高对目录的检索速度
- ▶ 实现文件共享: 文件物理拷贝只有一处, 能从不同目录中访问到
- ▶ 允许重名: 允许不同文件具有相同文件名, 能解决命名冲突





## § 4 目录管理技术

### 四、FCB分解技术

- 由**文件属性**信息单独形成文件的**索引结点**,也称I (Index)结点
- 将FCB分解为文件名、I结点指针和I结点
- 由文件名和I节点指针构成**符号名目录项**。这样,在索引结点机制中,文件目录项分解成符号名目录项和索引结点二部分

例1:某文件系统共有3200个文件, FCB=64Byte,盘块大小为1K。试问在这个文件系统中查找一个FCB, 平均需启动多少次盘块?

**【分析】** 1个盘块可放 $1024/64=16$ 个FCB,  
目录文件占 $3200/16=200$ 个盘块  
查找一个文件的FCB平均启动 $200/2=100$ 次盘块读入

14B		2B	
文件名		索引节点 指针	
文件名1		索引节点指针1	
文件名2		索引节点指针2	
...		...	

UNIX符号名目录文件



## § 4 目录管理技术

### 四、FCB分解技术

- 由**文件属性**信息单独形成文件的**索引结点**,也称I (Index)结点
- 将FCB分解为文件名、I结点指针和I结点
- 由文件名和I节点指针构成**符号名目录项**。这样,在索引结点机制中,文件目录项分解成符号名目录项和索引结点二部分

例1:某文件系统共有3200个文件, FCB=64Byte,盘块大小为1K。试问在这个文件系统中查找一个FCB, 平均需启动多少次盘块?

**【分析】** 1个盘块可放 $1024/64=16$ 个FCB,  
目录文件占 $3200/16=200$ 个盘块  
查找一个文件的FCB平均启动 $200/2=100$ 次盘块读入

例2:对上题文件系统采用FCB分解技术构造目录, 目录由64B变为16B。试问查找一个FCB平均需启动多少次盘块?

**分解前:**查找指定的FCB平均启动100次盘块

**分解后:**1个盘块可放64个FCB,目录文件占50个盘块。  
查找指定的FCB平均启动 $25+1=26$ 次盘块



## § 4 目录管理技术

### 五、一级文件目录

为所有文件建立一张线性目录表，每个文件的FCB在表中占据一个目录项

文件名	状态位	其它	物理地址
Alpha	1	...	_____→
Bata	1	...	_____→
...	...	...	...

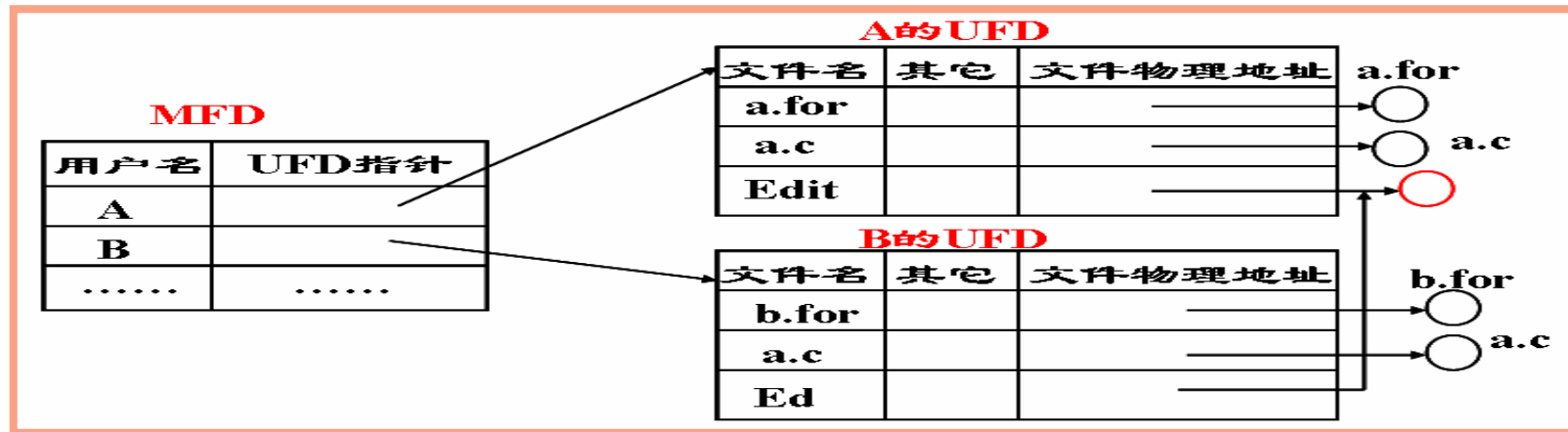
- **能实现目录最基本功能:**文件的“按名存取”
- **目录查找速度慢:** $n$ 个用户,各用户 $m$ 个文件,最坏速度是 $n*m$
- **无法解决命名冲突:**重名(多个文件有相同的文件名) 问题无法解决
- **不便于文件共享:**不能别名(一个文件有多个不同的文件名)共享



## § 4 目录管理技术

### 六、二级文件目录

将目录分成二级:第一级主文件目录Master File Directory, 给出用户名和用户文件目录的指针; 第二级用户文件目录User File Directory, 建立该用户所有文件的FCB目录项。



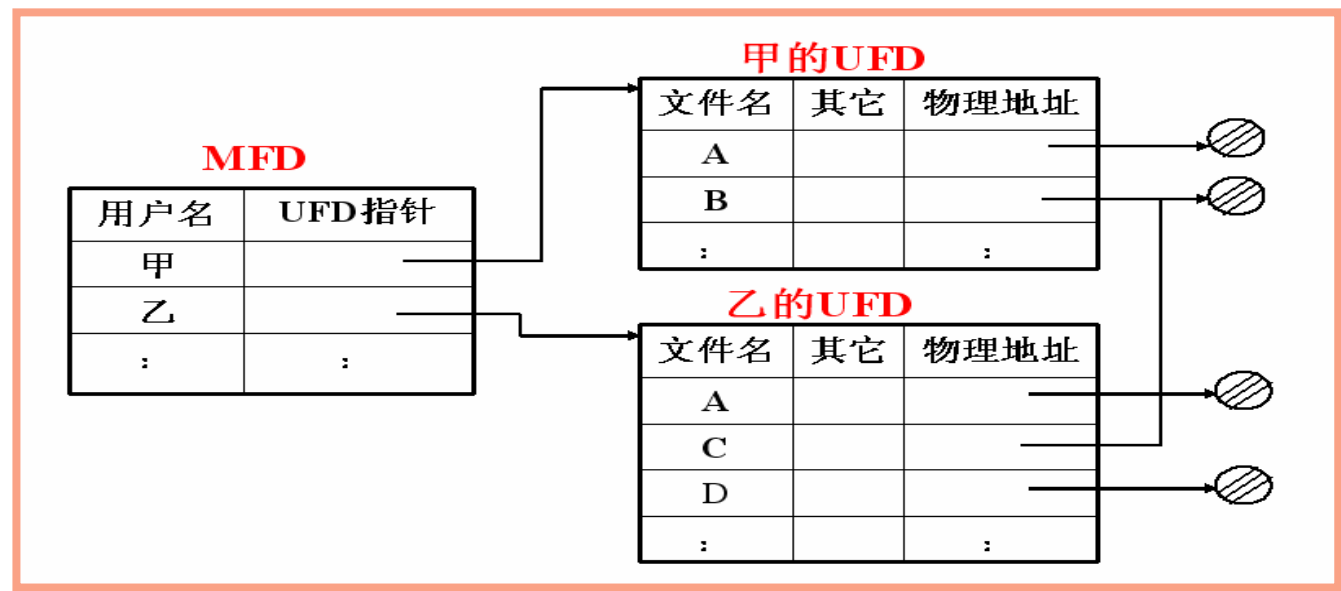
- 实现了文件的“按名存取”: 用户名/文件名
- 提高了目录检索速度: 如n个用户, 每用户最多m个文件, 则最坏速度为 $n+m$ 而非 $n*m$
- 解决了文件的重名问题和文件共享问题



## § 4 目录管理技术

例：有甲、乙两个用户，甲用户有文件A、B，乙用户有文件A、C、D。甲用户的文件A与乙用户的文件A不是同一文件，甲用户的文件B与乙用户的文件C是同一文件。请设计一个目录组织方案，并画图说明。

- **重名问题**——甲的A文件与乙的A文件
- **别名问题**——甲的B文件与乙的C文件





## § 5 文件的物理结构

**文件的物理结构:**是指从系统角度出发, 文件在物理存储器上的存放方法。物理结构是用户不可见的文件结构。

**物理文件:**由文件的物理结构所形成的相关信息集称为物理文件, 物理文件是相关物理盘块的集合。研究侧重点在于提高文件访问速度和有效利用外存空间。





## § 5 文件的物理结构

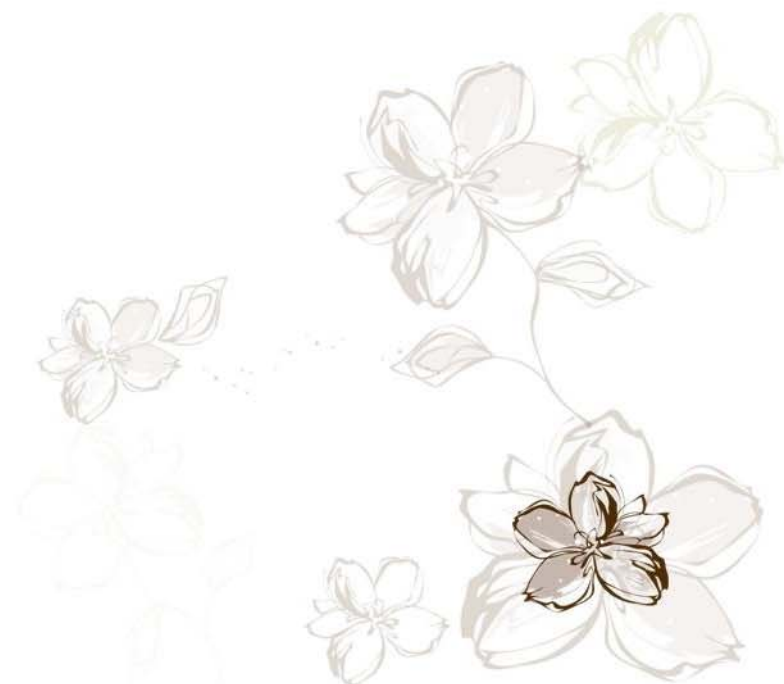
### ❖ 自主学习内容

1. **连续文件**的存储结构及其存取特点

---

2. **链接文件**的存储结构及其存取特点

---





# § 5 文件的物理结构

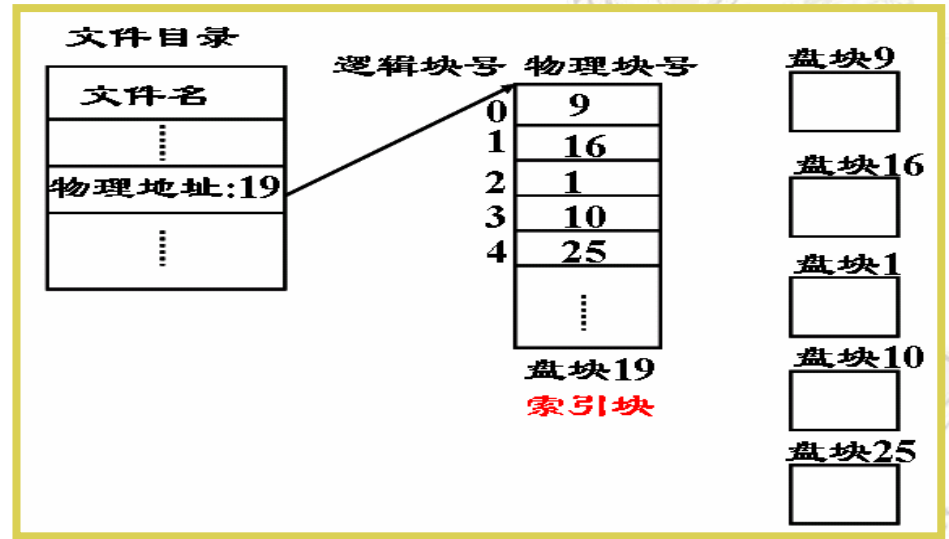
## 三、索引结构

**背景:**盘块大小=1K,盘块号占4字节。一盘块可记录256个盘块指针

**1.单级索引:**为离散分配的文件分配一个索引块,在索引块中记录该文件的所有盘块号。

- 既支持顺序存取, 又支持直接存取
- 对中小型文件, 造成索引块空间浪费
- 寻址能力256盘块, 可寻址的文件长度 $\leq 256K$

例：一个文件有5个记录,  
获得9,16,1,10,25号盘块。  
此外, 为该文件分配了19  
盘块作为索引块。





## § 5 文件的物理结构

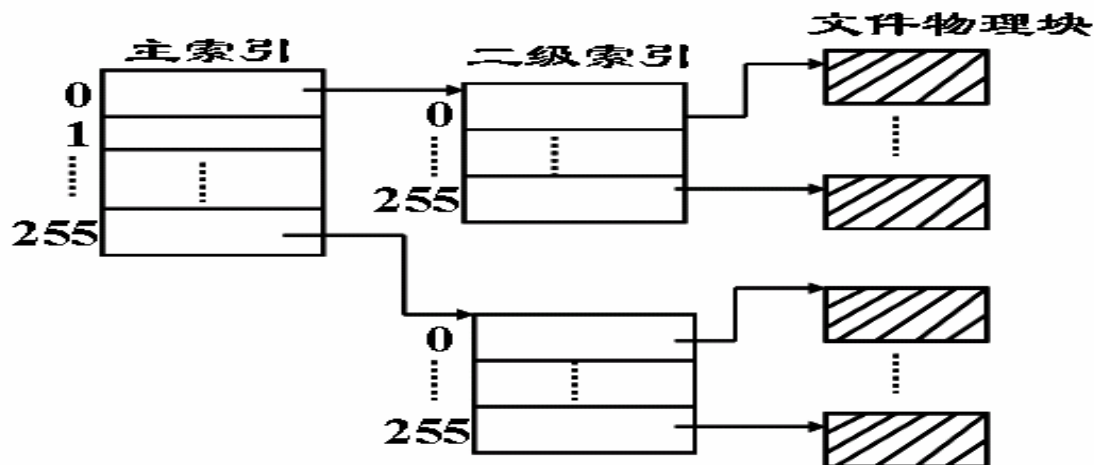
### 三、索引结构

**背景:**盘块大小=1K,盘块号占4字节。一盘块可记录256个盘块指针

**1.单级索引:**为离散分配的文件分配一个索引块,在索引块中记录该文件的所有盘块号。

**2.二级索引:**设立第一级(主)索引,在主索引中填入第二级各索引块的块号,第二级索引块中记录文件的盘块号。

——寻址能力 $256^2=64K$ 块,文件最大长度:  $64K*1K=64M$





## § 5 文件的物理结构

例 (**清华大学**)：文件系统采用多重索引结构搜索文件内容，设块长为512B，每个块号占3B，如果不考虑逻辑块号在物理块中所占位置，分别计算采用二级索引和三级索引时可寻址的文件最大长度。

**块长= 512B，块号占3B。**

**因此，一个索引块含 $\text{INT}(512/3)=170$ 个索引项**

**二级索引文件最大长度=170\*170=28900块**

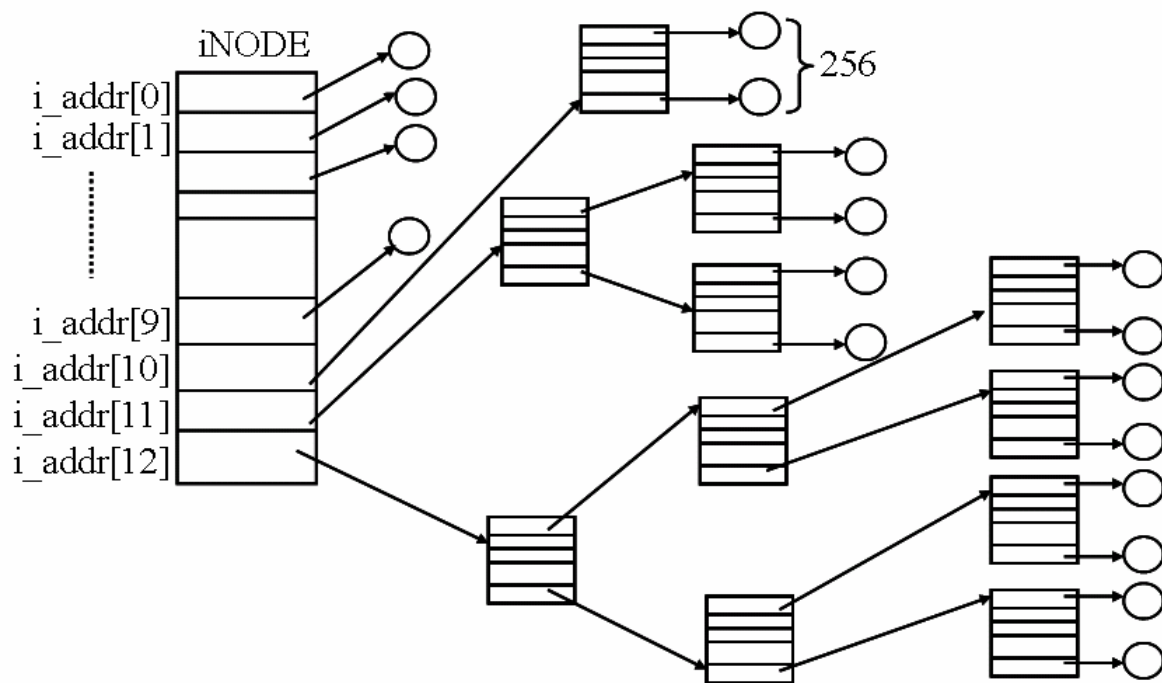
**三级索引文件最大长度=170\*170\*170=491300块**





## § 5 文件的物理结构

### 四、UNIX混合索引机制



设字节偏移用*i*表示,逻辑块号用*LB*表示,  
块内地址用*LW*表示

公式:  $LB = \text{INT}(i / \text{盘块长})$

$LW = i \text{ MOD } \text{盘块长} = i - LB * \text{盘块长}$

(1) 若  $0 \leq LB < 10$ ,

采用直接寻址

*i*字节偏移对应物理地址:  
*i\_addr*[*LB*]盘块的第*LW*  
地址

(2) 若  $10 \leq LB < 10 + 256$ ,

采用一级间址寻址

在一级间址块中的地址  
 $= LB - 10$

(3)  $266 \leq LB < 266 + 256^2$ ,

采用二级间址

在二级间址块中地址  
 $= (LB - 266) / 256$ ,

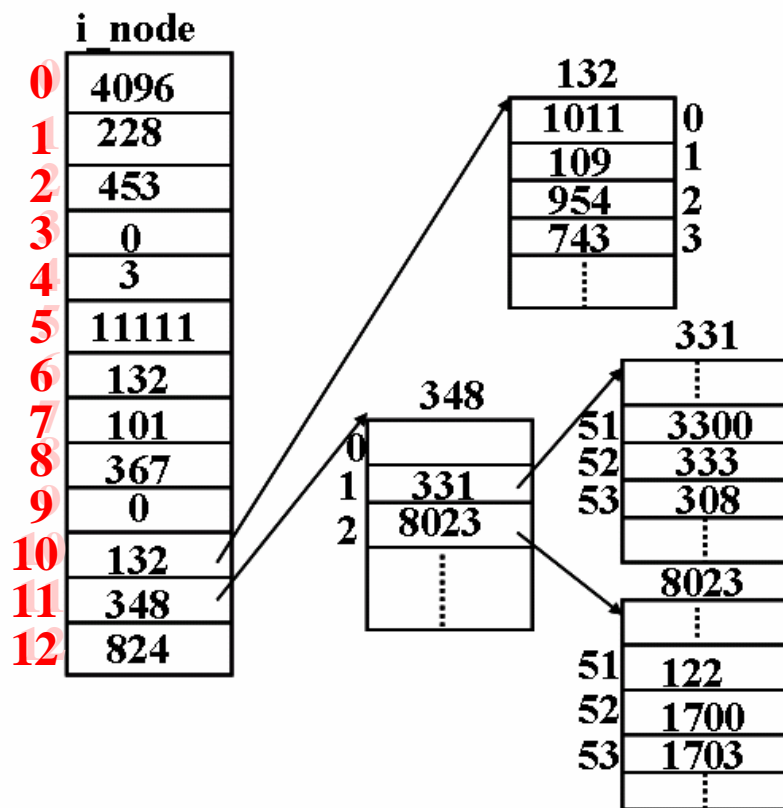
在一级间址块中地址  
 $= (LB - 266) \% 256$



## § 5 文件的物理结构

例1：某文件索引结点如图所示，设盘块大小为1K，每个盘块号占4个字节，试将文件的下列字节偏移量转换为物理地址。

(1) 9000      (2) 14000      (3) 850000



(1) 字节偏移量9000：

逻辑块号=9000/1024=8

块号偏移=9000-1024\*8  
=808

因为逻辑块号小于10，故  
采用直接寻址。

从i\_addr[8]读出物理盘块  
为367，故9000的物理地址  
是367盘块的第808字节。

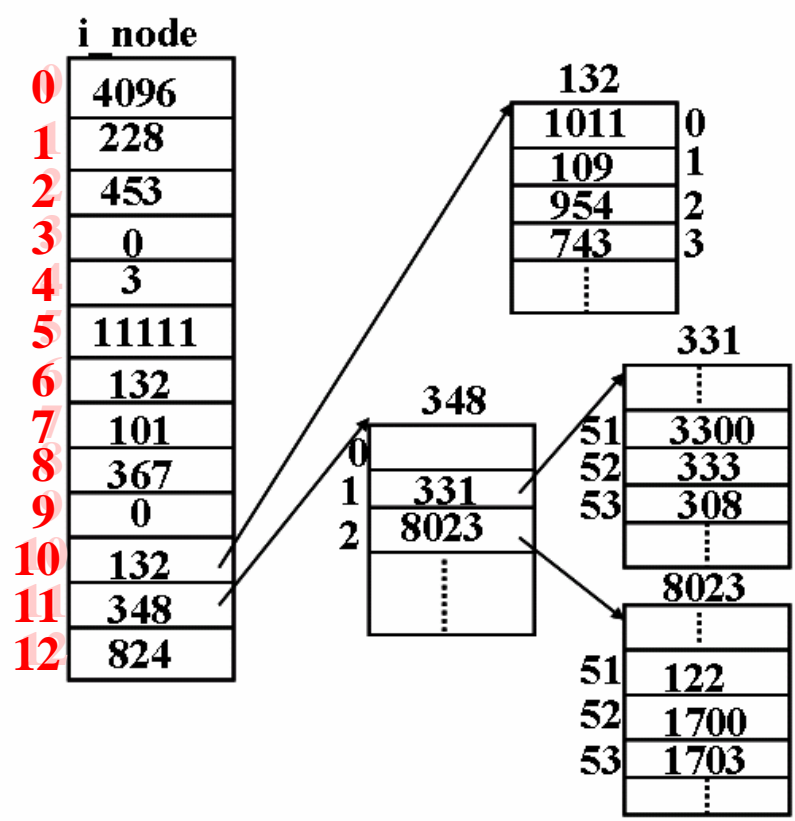




# § 5 文件的物理结构

例1：某文件索引结点如图所示，设盘块大小为1K，每个盘块号占4个字节，试将文件的下列字节偏移量转换为物理地址。

- (1) 9000
- (2) 14000
- (3) 850000



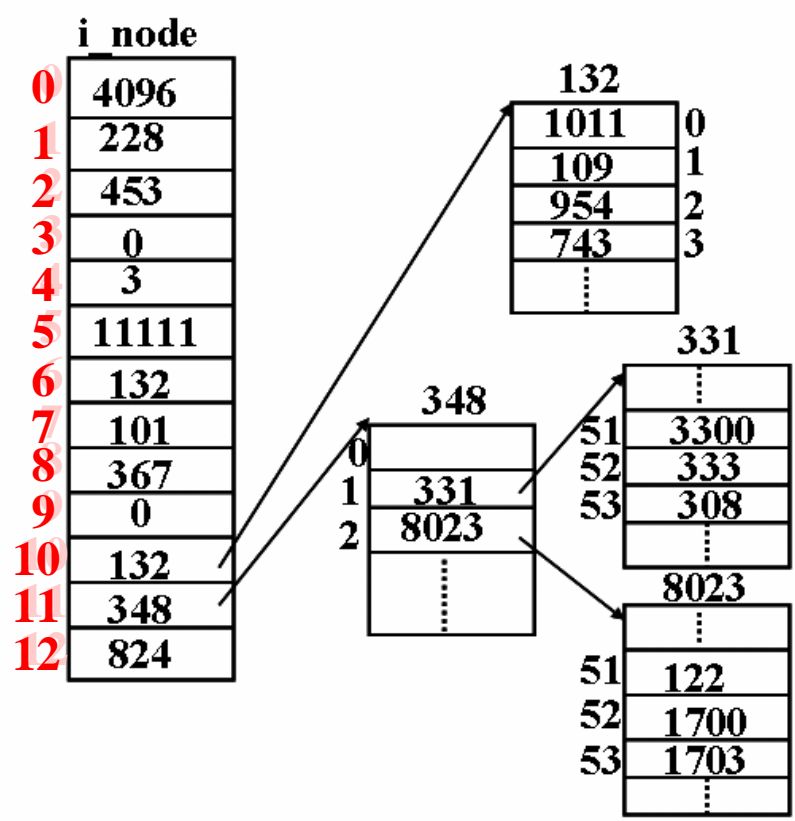
(2) 字节偏移量14000：  
逻辑块号=14000/1024=13  
块号偏移=14000-1024\*13  
=688  
因为10≤13<266，故采用一级间接寻址。  
从i\_addr[10]读出一级间址块为132。  
在一级间址块中的地址=13-10=3。  
故14000的物理地址是743  
盘块的第688字节。



# § 5 文件的物理结构

例1：某文件索引结点如图所示，设盘块大小为1K，每个盘块号占4个字节，试将文件的下列字节偏移量转换为物理地址。

- (1) 9000      (2) 14000      (3) 850000



(3)字节偏移量850000：

逻辑块号=830,块内偏移80

因为 $266 < 830 < 266 + 256^2$ ，

故为二级间接寻址。

从i\_addr[11]读出二级间址块为348。

$(830 - 266) / 256 = 2$

$(830 - 266) \% 256 = 52$

故850000的物理地址是1700盘块的第80字节。



## § 5 文件的物理结构

**例2 (南航)：** UNIX混合索引结构，设盘块大小为1K，每个间址放256个盘块地址，问：

(1) 一个2M的文件要占用多少盘块？注意：占用的盘块空间包括文件本身和间址块两部分。

(2) 在该文件系统中，文件最大可为多少字节？

$$2\text{M}/1\text{K}=2048\text{块}$$

$$=10+256+1782$$

$$=10+256+6*256+246$$

因此，2M文件占用盘块数为：

$$10+1+256+1+7+1782=2057\text{块}$$

$$\text{文件最大长度} = (10+256+256^2+256^3) \times 1\text{KB}$$

$$\approx 1.725 \times 10^{10}\text{B}$$



## § 5 文件的物理结构

例3 (10年)：设文件索引节点中有7个地址项，其中4个地址项为直接地址索引，2个地址项是一级间接索引，1个地址项是二级间接地址索引，每个地址项大小为4字节，若磁盘索引块和磁盘数据块大小均为256字节，则可表示的单个文件的最大长度是 C。

A.33KB

B.519KB

C.1057KB

D.16516KB

1个索引块含有索引项 $=256/4=64$ 个

文件最大长度 $=(4+2*64+64*64)*256B$

$=4228*256B=1082368/1024K=1057K$



## § 6 文件存储器管理

### ❖ 自主学习内容：

#### 位示图技术

1. 如何用位示图表示磁盘空间中盘块使用情况？
2. 以位示图作为数据基，如何进行盘块分配与回收？



#### 空闲表技术

1. 在空闲表中如何表示磁盘空间中存在哪些连续的空闲盘块？
2. 如何查找空闲表实现盘块分配？有哪些常用的分配算法？
3. 为减轻外存碎片，在盘块回收时，应作何处理（合并）？





## § 6 文件存储器管理

### 三、UNIX成组链接法

#### 1. 空闲盘块组织

- 将文件存储器中所有空闲盘块,分成若干组
- 每组第一块记录下一组的块数及各块的块号

eg:一文件系统的文件存储空间计有7799盘块, 盘块号为201#~7999#, 以100块为一组进行分组:

201#~300#

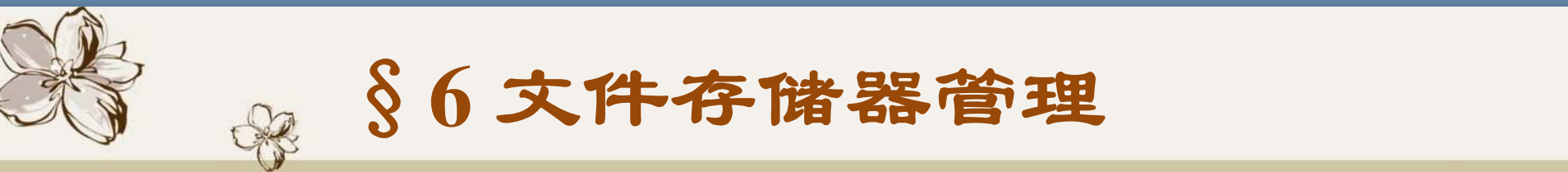
301#~400#

...

7801#~7900#

7901#~7999#, 0#

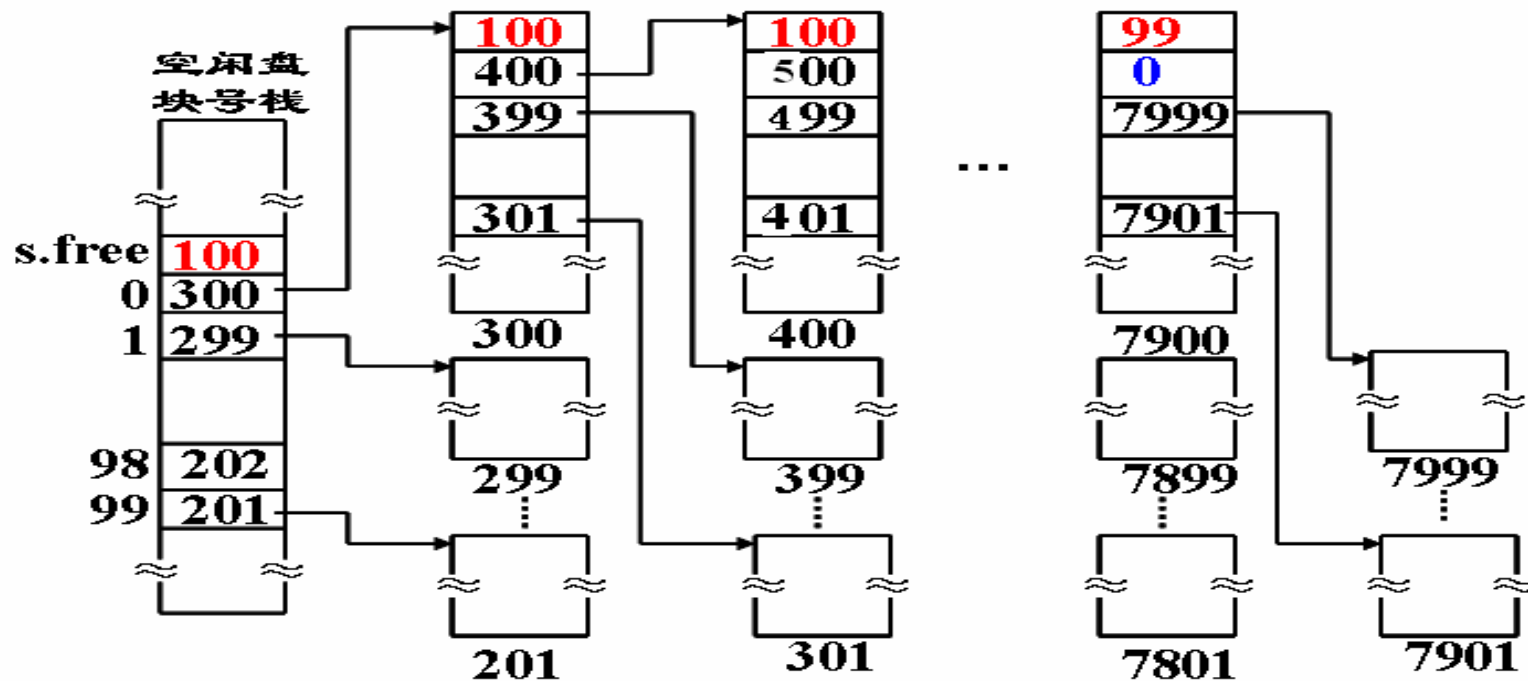




### 三、UNIX成组链接法

#### 1.空闲盘块组织

- 将文件存储器中所有空闲盘块,分成若干组
- 每组第一块记录下一组的块数及各块的块号
- 将第一组的盘块总数及盘块号计入空闲盘块栈s(卷资源表)中



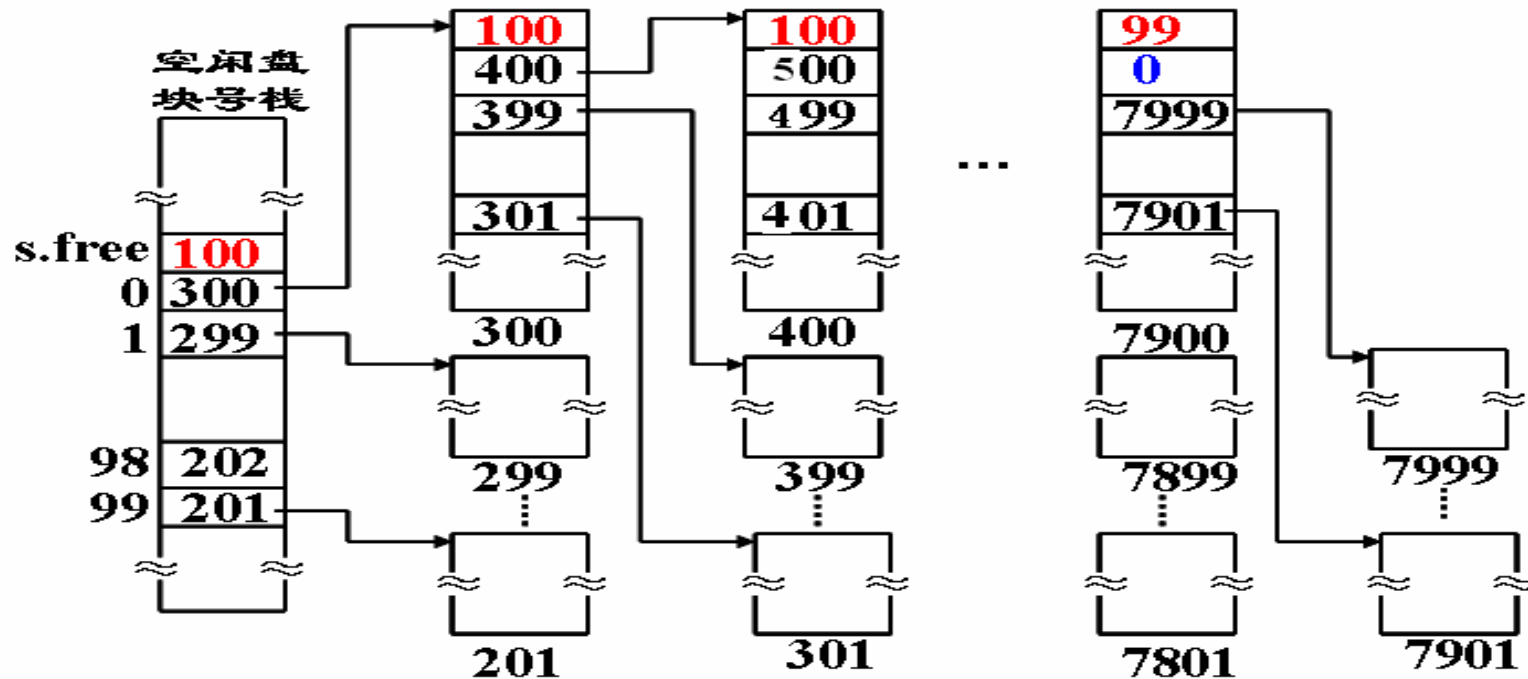


# § 6 文件存储器管理

## 三、UNIX成组链接法

### 2. 盘块分配算法

- $s.free := s.free - 1$ , 将  $s[s.free]$  中所填盘块分配出去
- 当  $s.free = 0$  时“**边界处理**”: 得到  $s[0]$  中的盘块号  $b$ , 将  $b$  中内容复制到栈中作为新的卷资源表内容, 再将  $b$  块分配出去





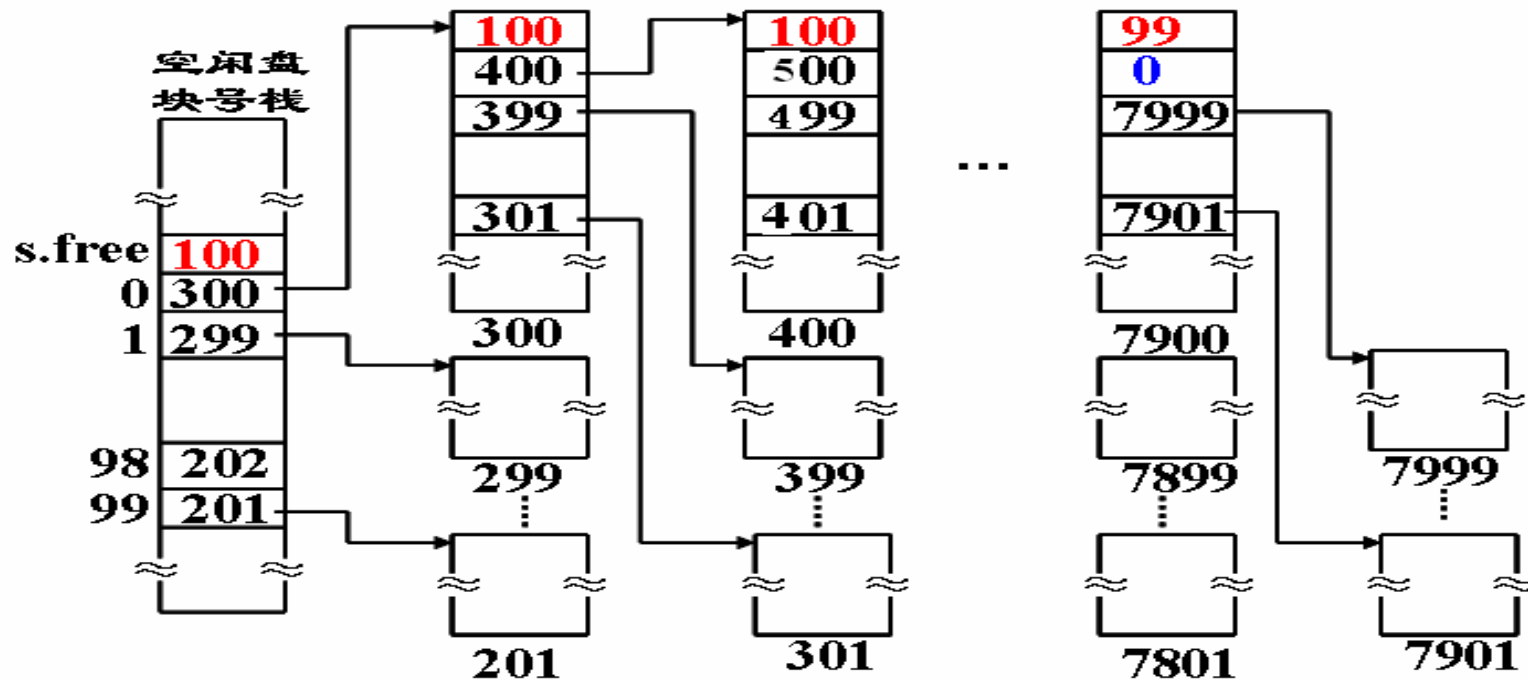
## § 6 文件存储器管理

### 三、UNIX成组链接法

#### 3. 盘块回收算法

— $s[s.free] := b$ ,  $s.free := s.free + 1$

— $s.free = 100$ 时**边界处理**：将卷资源表内容填入欲回收块 $b$ 中，清空卷资源表；置 $s.free$ 为0， $s[s.free] := b$ ,  $s.free := s.free + 1$

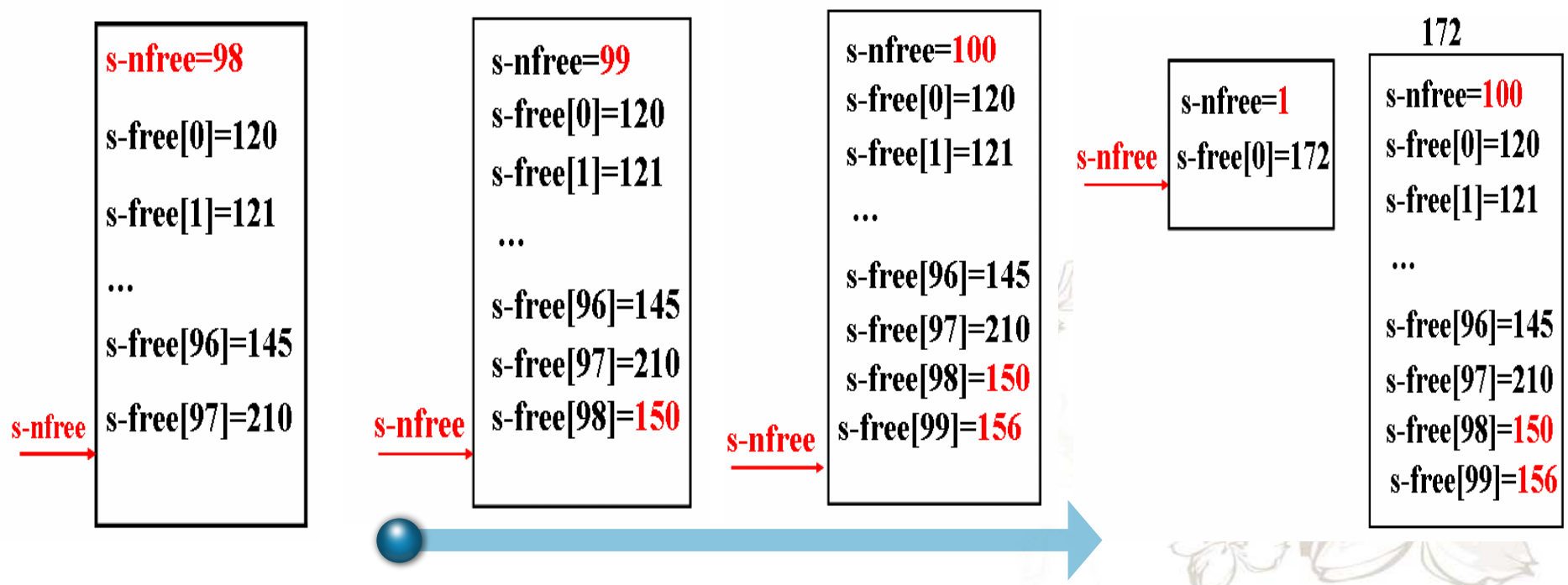




## § 6 文件存储器管理

例：在UNIX系统中有卷资源表如下所示：

- (1) 现有进程要释放四个物理块,其块号为150#,156#, 172#, 177#, 试画出卷资源表的变化。
- (2) 在(1)的基础上, 假定一进程要求5个空闲块, 试画出分配后的卷资源表。





# § 6 文件存储器管理

例：在UNIX系统中有卷资源表如下所示：

- (1) 现有进程要释放四个物理块,其块号为150#,156#, 172#, 177#, 试画出卷资源表的变化。
- (2) 在(1)的基础上, 假定一进程要求5个空闲块, 试画出分配后的卷资源表。

