

第6章文件系统

一、文件系统概述

1. 文件系统的引入

- 数据需要长期保存
- 主存储器容量较小
- 存储介质种类繁多
- 发挥数据的信息作用

2. 文件及分类

- 什么是文件(file)
 - ◆ 文件是一组相关数据的有序集合
 - ◆ 这组数据保存在电子存储介质上
 - ◆ 这个集合拥有一个用户可以访问的标识符。
- 文件命名(File Naming)
 - ◆ 文件名=文件主名[.扩展名]
 - ◆ “8.3”规则

➤ 文件分类

- ◆ 按文件的性质分类：系统文件、库文件和用户文件
- ◆ 按文件的存取控制分类：只读文件、可执行文件、读写文件和隐藏文件
- ◆ 按文件的组织分类：目录文件、普通文件和设备文件
- ◆ 按文件内容的表示分类：文本文件(普通文本文件和超文本文件)和二进制文件

3.文件系统及其主要功能

➤ 文件系统(File System)

➤ 文件系统主要功能

- ◆ 文件内容的组织
- ◆ 文件和目录管理
- ◆ 文件存储空间管理
- ◆ 文件系统的接口
- ◆ 文件的共享与安全性

二、文件的逻辑结构

1.流式文件

- 现有文件系统的文件逻辑结构大多数都是采用流式文件
- 没有对文件内容进行结构上的划分
- 文件读写指针

2.记录式文件

- 逻辑记录,简称记录(Record)
- 字段(Field)
- 记录长度
- 主键(Primary key)
- 定长记录文件和变长记录文件

三、文件的物理结构

1.文件存取方式(File Access)

- 顺序存取法(Sequential Access)
- 随机存取法(Random Access)

2.文件存储介质

- 磁盘-随机存取的存储设备
- 磁带-顺序存取的存储设备

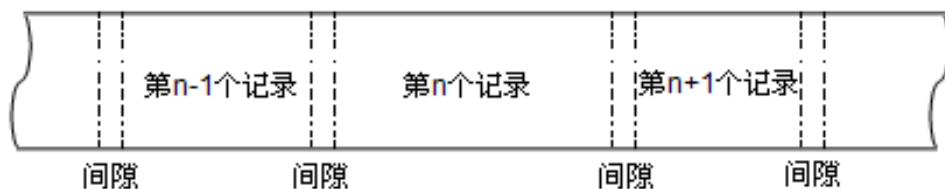


图6-3 磁带结构的例子

文件的物理结构是指文件内容在存储介质上的存放方式

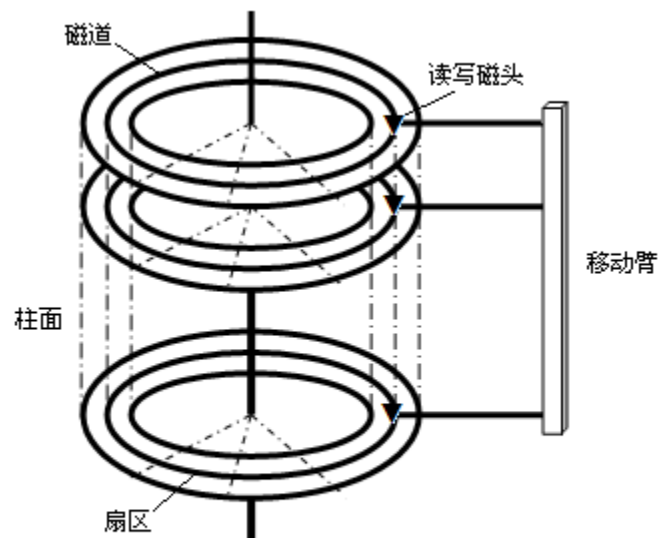


图6-2 磁盘的结构示意图

柱面号C、磁头号H和扇区号S
逻辑地址(LBA)

3.文件I/O操作

- 文件I/O操作的基本单位是物理块

4.记录成组和分解

- 不仅可以减少内碎片的存储开销
- 可以减少I/O操作的次数

5.物理结构分类

- 连续结构

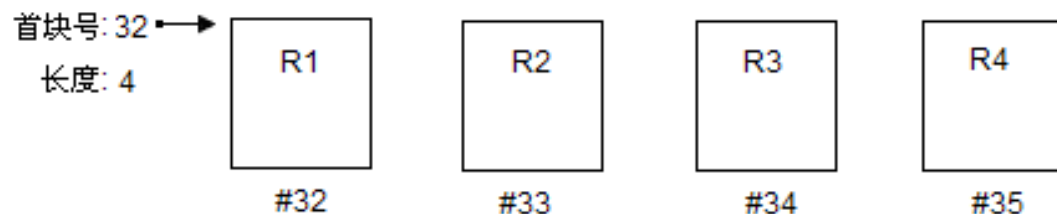


图6-4 连续结构例子

连续结构的特点:

- ①管理简单
- ②存取速度快
- ③存储空间连续分配，存储空间利用率不高
- ④不便于文件内容的增加或删除

➤ 链接结构

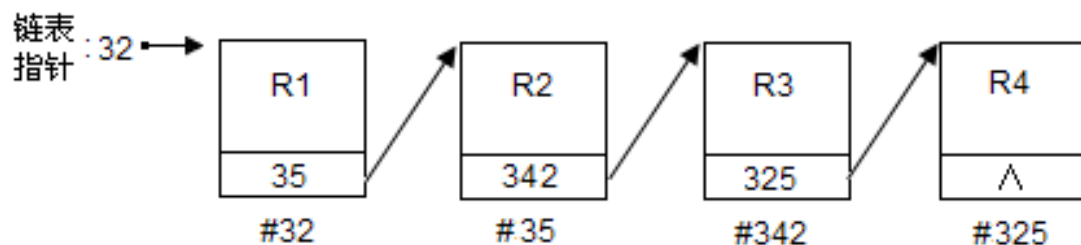


图6-5 链接结构例子

链接结构的特点:

- ①非连续的存储分配，提高了存储空间的利用率
- ②方便文件内容的增加或删除
- ③只适合于顺序存取，存取速度慢
- ④指针信息造成物理块信息不完整，并导致数据无法控制

文件 A →	0	00F8H
	1	293
	2	101
	⋮	
	100	257
	101	301
	⋮	
	255	0000H
文件 B →	256	100
	257	258
	258	FFFFH
	⋮	
	300	0000H
	301	FFFFH
		...

图6-6 FAT16的例子

➤ 索引结构

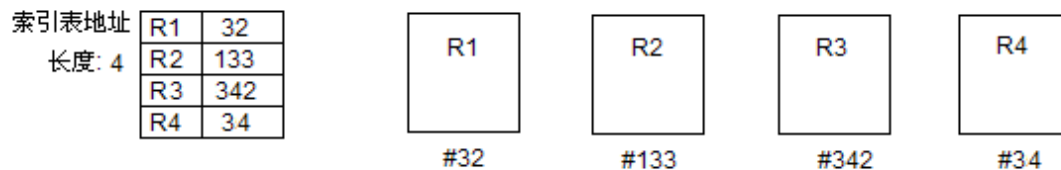


图6-7 索引结构例子

索引结构的特点:

- ①非连续的存储分配，提高了存储空间的利用率
- ②方便文件内容的增加或删除
- ③实现随机存取
- ④索引表占用额外的存储空间
- ⑤增加检索的开销

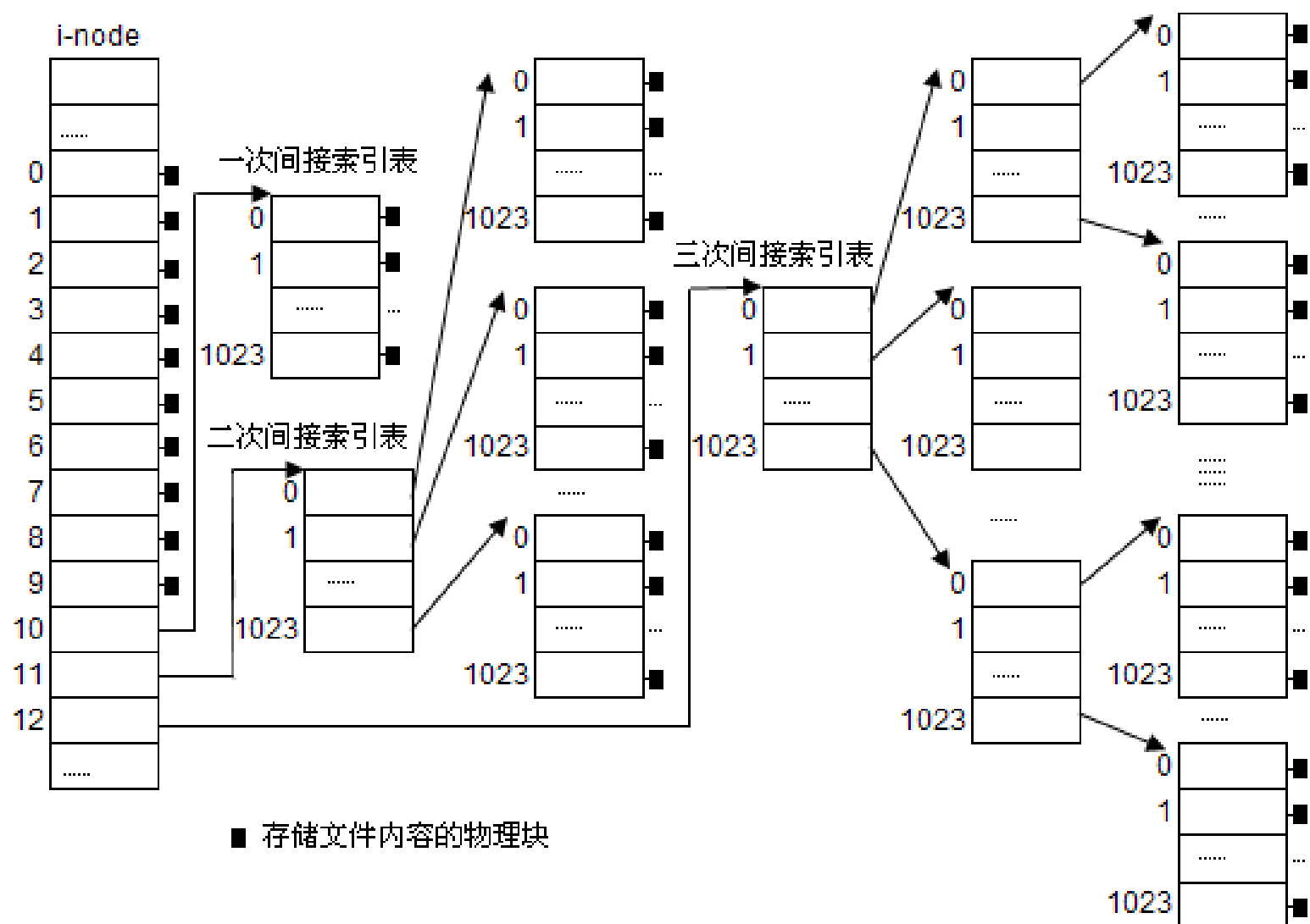


图6-8 UNIX的多级索引结构

四、文件目录管理

1.文件控制块(FCB, File Control Block)

- 描述信息部分
- 管理信息部分

2.文件目录及其结构

- 文件目录
- 目录文件,简称目录(Directory), 或文件夹
- 目录结构的分类
 - ◆ 单级目录

访问文件时,检索范围大,
影响了存取时间

任何两个文件名都不能重名

.....
用户 u1 的文件 A 的 FCB
用户 u2 的文件 B 的 FCB
用户 u2 的文件 C 的 FCB
用户 u3 的文件 D 的 FCB
用户 u1 的文件 E 的 FCB
.....

图6-9 单级目录结构例子

◆ 二级目录

用户文件目录(UFD)和系统主目录(MFD)

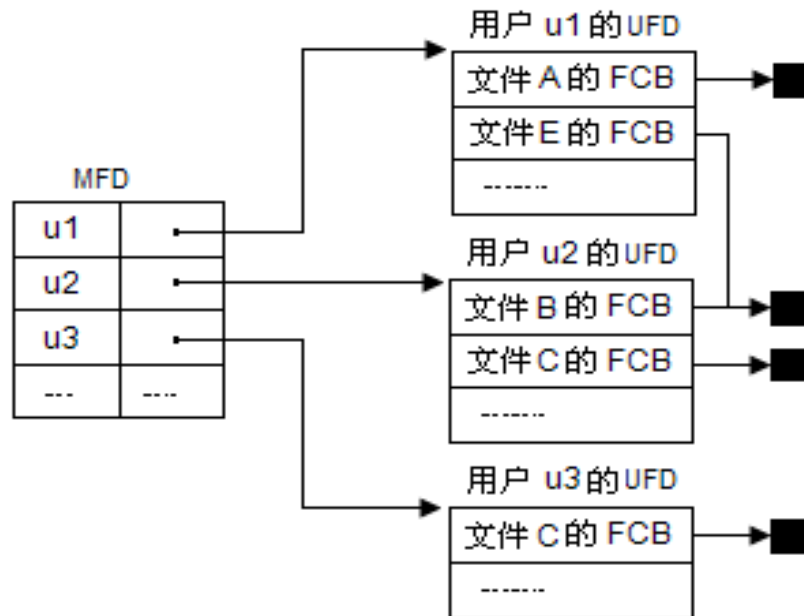


图6-10 二级目录结构例子

二级目录也具有结构简单、容易实现的优点，同时，减少了检索的开销。

二级目录结构还解决不同用户之间的重名问题

在二级目录结构中，容易实现用户之间的文件共享

◆ 多级目录-树状目录结构

根目标(Root)

父目录/子目录

绝对路径(Absolute Path)

当前目录(Current Directory)

相对路径(Relative Path)

两个特殊的子目录：“.”和“..”

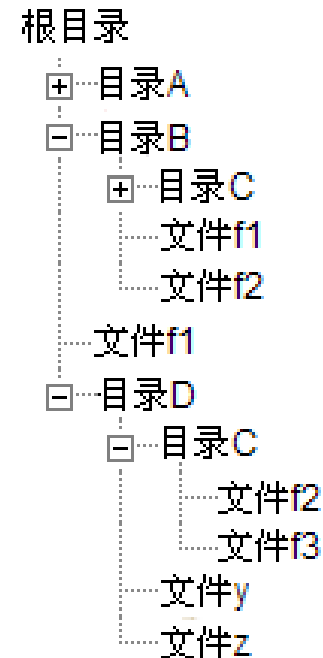


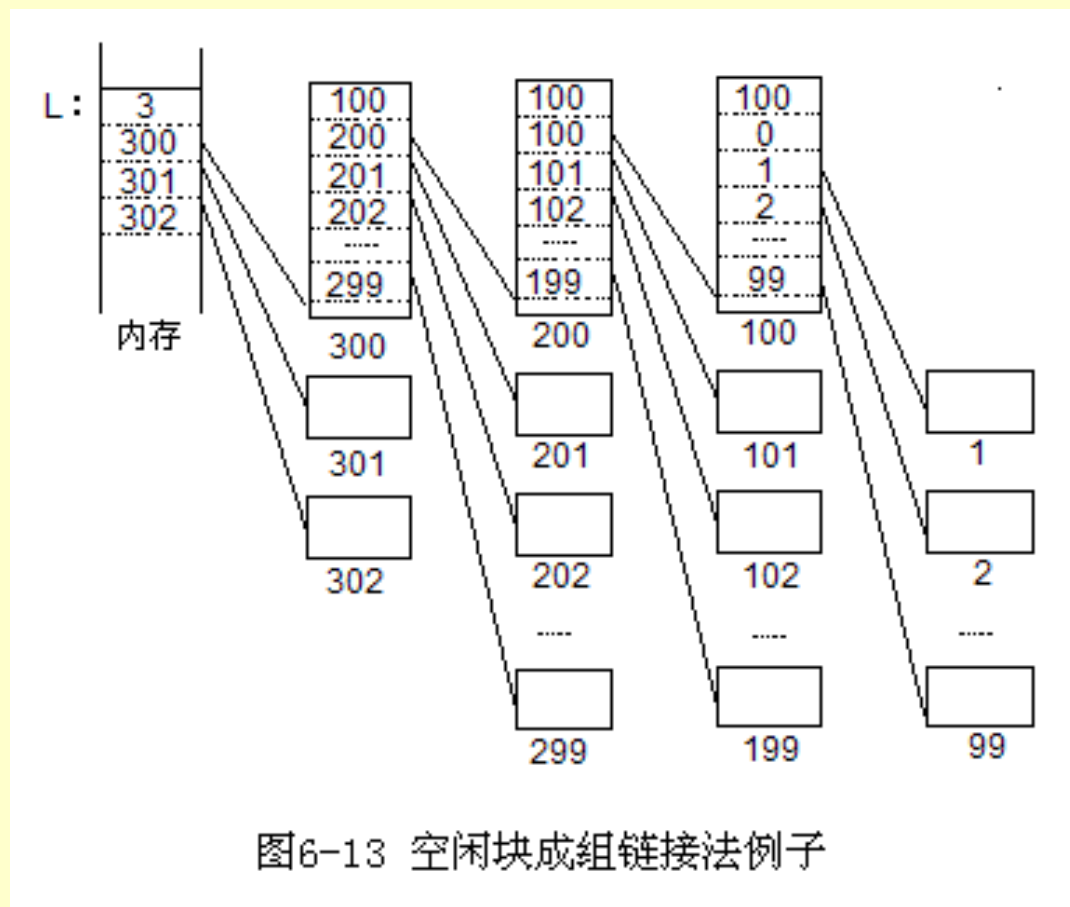
图6-11 树型目录结构例子

五、文件存储空间管理

1.管理磁盘空闲物理块的数据结构

- 位示图
- 空闲块表
- 空闲块链表

2.空闲块成组链接法



分配算法

当一个文件需要1个空闲块时，分配算法如下：

```
i=L[0];           //专用组空闲块数
free=L[i];        //取专用组空闲块列表末尾的一个空闲块，用于分配
if(free==0){
    报告：磁盘没有空闲块，不能保存；
    算法结束； }
if( i==1){        //专用组的第1个块，该块含有链表的指针信息
    启动读I/O操作，读取专用组的第1个块(即块号free)的信息；
    并存入内存L开始的区域中，下一个组作为专用组；
} else {          //专用组的空闲块数大于1，只须空闲块数减1
    i=i-1;
    L[0]=i; }
块号free，分配给文件。 ■
```

在图6-13。假定用户新建一个文件A，其文件内容需要5个空闲块，那么，按上述算法分配时，文件A依次得到的块号.....

回收算法

当回收一个文件占用的1个空闲块时，假定回收的空闲块号为**free**，算法描述如下：

```
i=L[0];           //专用组空闲块数
if(i<100)         //专用组空闲数目小于规定有的分组数
    i=i+1;        //空闲块数量加1;
else{             //专用组空闲块数达到分组的数目，可以构成一个组
    启动写I/O操作，将内存L开始的101个存储单元的数据写入新回收的块free中；
    新回收的块作为专用组，置i=1;
}
L[i]=free;        //新回收的空闲块加入空闲块号列表的末尾;
L[0]=i。 ■
```

用户对文件A的内容作了删减和修改，在删减过程，依次回收299、301、302和298，成组链表的结构？

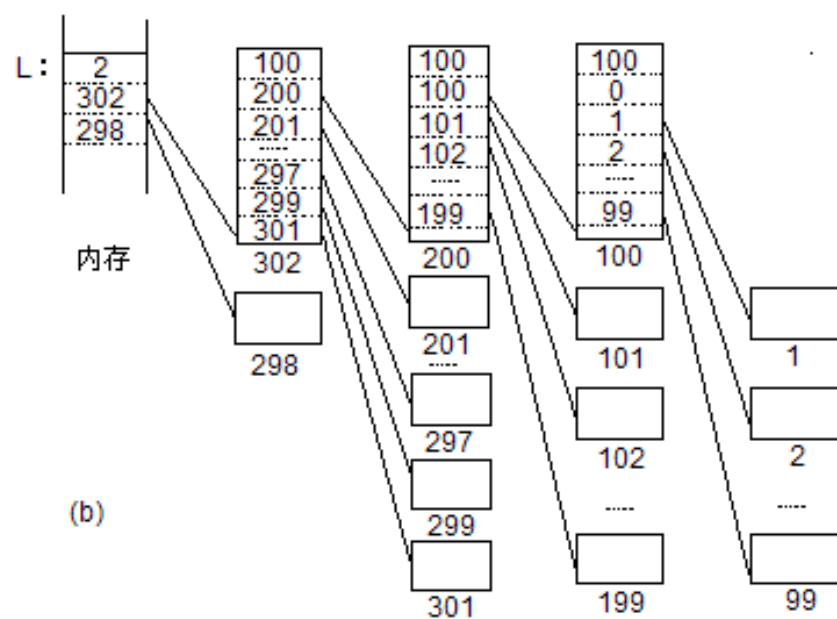
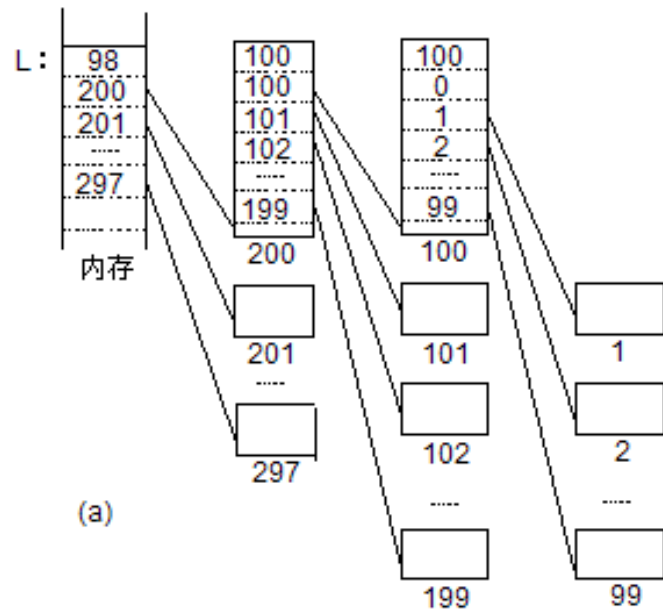


图6-14 空闲块成组链接分配、回收

六、文件使用

1.命令接口

➤ 上机实践

2.程序接口

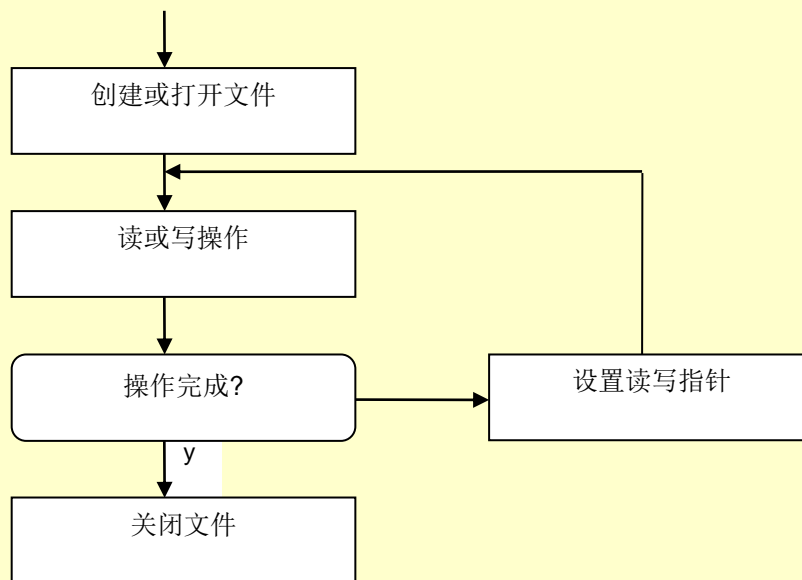


图6-15 文件使用的基本步骤

七、文件的共享

1.文件共享(File Sharing)的目的

- 避免因每个用户或进程各自保存一份，造成的外存储器的开销
- 用户或进程之间的任务协作

2.文件共享(File Sharing)的基本方法

- 绕道法
- 链接法
- 基本文件目录法(BFD)

3.基本文件目录法(BFD)

在文件的按名存取过程中，首先根据文件名查找文件目录，得到文件的**FCB**；然后，从**FCB**中得到文件的物理位置，再启动**I/O**操作进行文件的读/写操作。

- ◆ 符号文件目录(SFD, Symblc Files Directory)
- ◆ 基本文件目录(BFD, Binary Files Directory)
- ◆ 系统主目录(MFD, Main Files Directory)

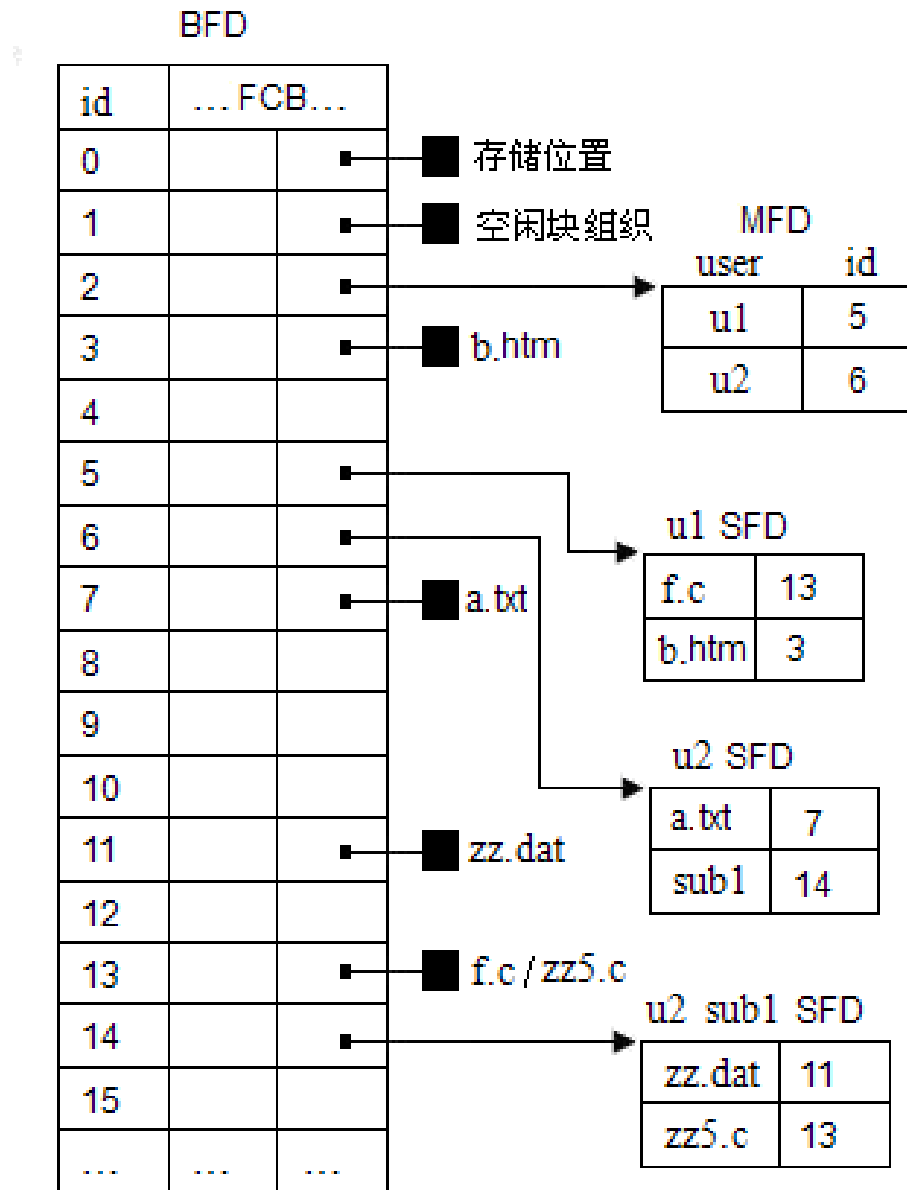


图6-16 基本文件目录法 (BFD)

4.文件共享语义

- ◆ UNIX语义(UNIX Semantics)
- ◆ 会话语义(Session Semantics)
- ◆ 永久文件(Immutable File)

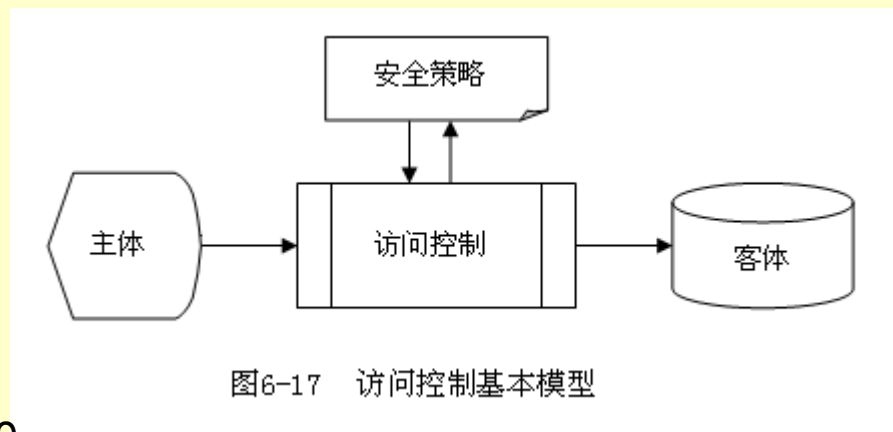
八、文件的安全性

1.文件保护及主要方法

- 文件保护(Protection)含义
- 文件保护的措施：文件的备份(Backup)和恢复(Restore)

2.文件保密及主要方法

- 文件保密含义
- 存取控制的主体、客体、访问属性
- 访问控制模块



➤ 访问控制模块本身应该满足3个条件

◆完备性(Complete)

◆防护性(Protected)

◆行为正确性(Proper Behavior)

➤ 基于主体权限的存取控制方式

◆存取控制矩阵(PM, Protection Matrix)

◆存取控制表(ACL, Access Control List)

◆权能表(CL, Capability List)

	file ₁	file ₂	file ₃	file ₄
u ₁	r		rx	rw
u ₂		wx		r
u ₃	rx	w	x	rwX

图6-18 存取控制矩阵例子

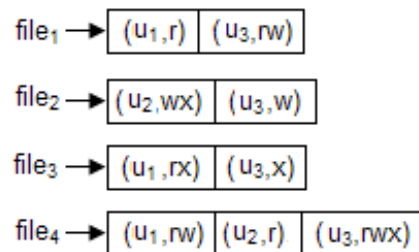


图6-19 存取控制表例子

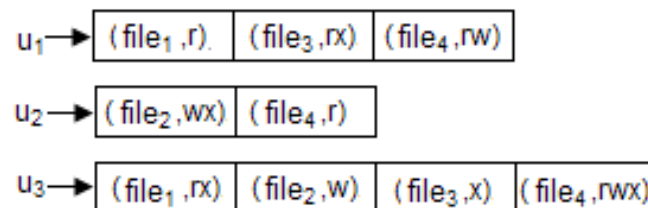


图6-20 权能表例子