

《操作系统》



主讲教师：翟高寿

联系电话：010-51684177 (办)

电子邮件：gszhai@bjtu.edu.cn

制作人：翟高寿

制作单位：北京交通大学计算机学院

第六章 文件管理

6.1 文件和文件系统

6.2 文件的逻辑结构

6.3 外存分配方式

6.4 文件存储空间管理

6.5 目录管理

6.6 文件共享与保护

6.7 数据一致性控制

6.1 文件和文件系统

6.1.1 文件管理目标及功能

6.1.2 文件、记录和数据项

6.1.3 文件类型

6.1.4 文件系统模型

6.1.5 文件操作

文件管理目标及功能

□ 文件管理目标

- 方便用户、保证安全
- 存取快速、资源利用

□ 文件管理功能

- 管理在外存上的文件
- 提供文件存取、共享和保护手段

□ 文件系统技术手段分析

- 目录、索引结点、“结构”优化
- 存取控制矩阵、容错技术

6.1 文件和文件系统

6.1.1 文件管理目标及功能

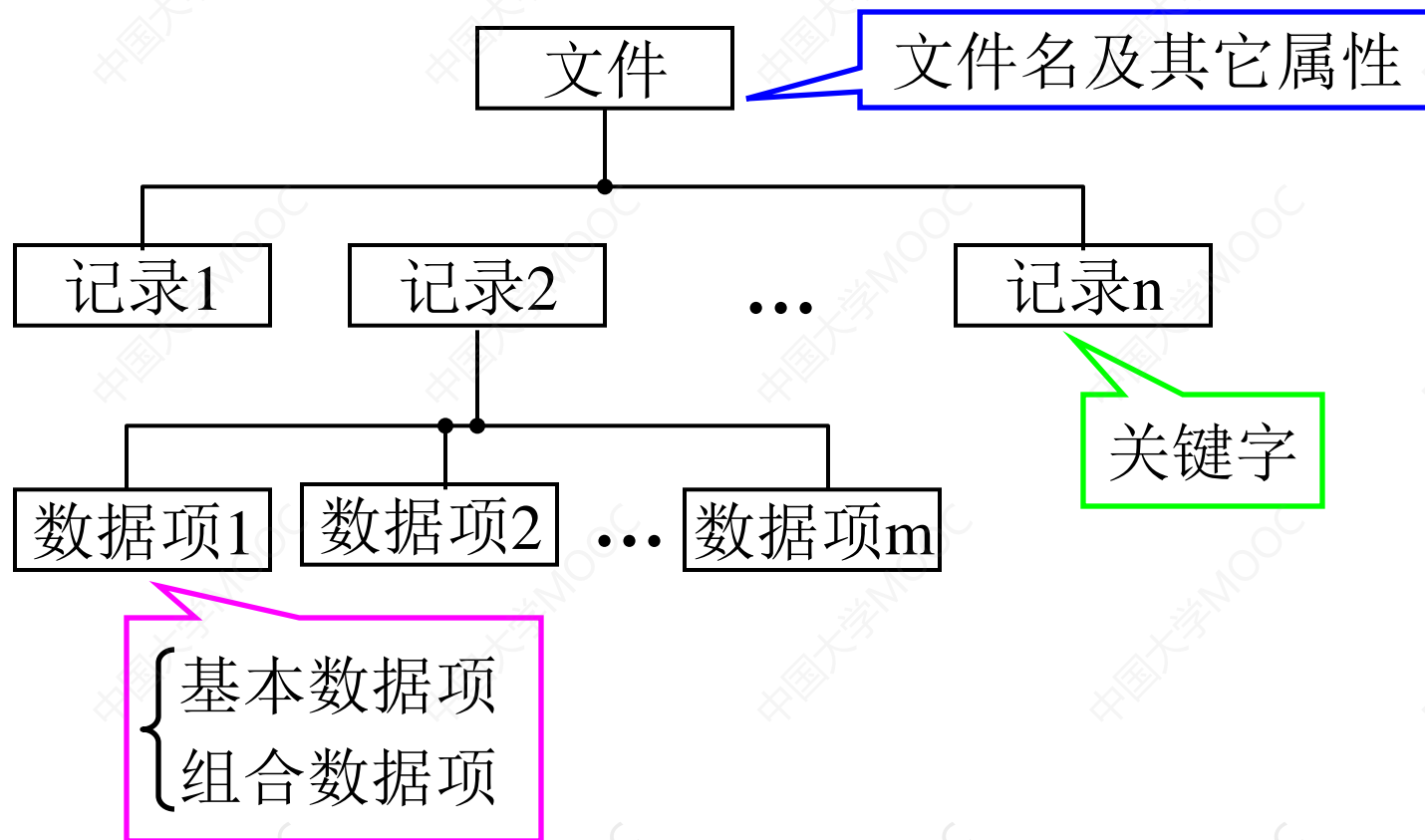
6.1.2 文件、记录和数据项

6.1.3 文件类型

6.1.4 文件系统模型

6.1.5 文件操作

文件、记录和数据项



6.1 文件和文件系统

6.1.1 文件管理目标及功能

6.1.2 文件、记录和数据项

6.1.3 文件类型

6.1.4 文件系统模型

6.1.5 文件操作

文件类型. . .

常对应扩展名

- ❖ 为方便文件存取及管理控制而引入
- 按文件性质与用途分类
 - 系统文件、用户文件、库文件
- 按文件中的数据形式分类
 - 源文件、目标文件、可执行文件
- 按存取控制属性分类
 - 只执行文件、只读文件、读写文件
- 按文件逻辑/物理结构分类

6.1 文件和文件系统

6.1.1 文件管理目标及功能

6.1.2 文件、记录和数据项

6.1.3 文件类型

6.1.4 文件系统模型

6.1.5 文件操作

文件系统模型

文件系统接口（命令/图形化用户接口、程序接口）	
文件管理软件集合	逻辑文件系统层
	基本I/O管理程序层(文件组织模块层)
	基本文件系统层(物理I/O层)
	I/O控制层(设备驱动程序层)
硬件对象（磁盘等辅助存储器）	

常用文件系统格式类型

□ FAT

磁盘卷数据/盘块
组织结构规范

➤ FAT12、FAT16、FAT32

属性	FAT12	FAT16	FAT32
适用场合	小容量[硬/U/软]盘	中等容量硬盘	大容量硬盘
FAT表项大小	12位	16位	28位
最大簇数	4096	65536	>260 000 000
所采用盘块大小	512B~4KB	2KB~32KB	4KB~32KB
最大卷大小（字节）	$16\ 736\ 256 \approx 2^{24}$	$2\ 147\ 133\ 200 \approx 2^{31}$	大约 $2^{41} \approx 2^{43}$

6.1 文件和文件系统

6.1.1 文件管理目标及功能

6.1.2 文件、记录和数据项

6.1.3 文件类型

6.1.4 文件系统模型

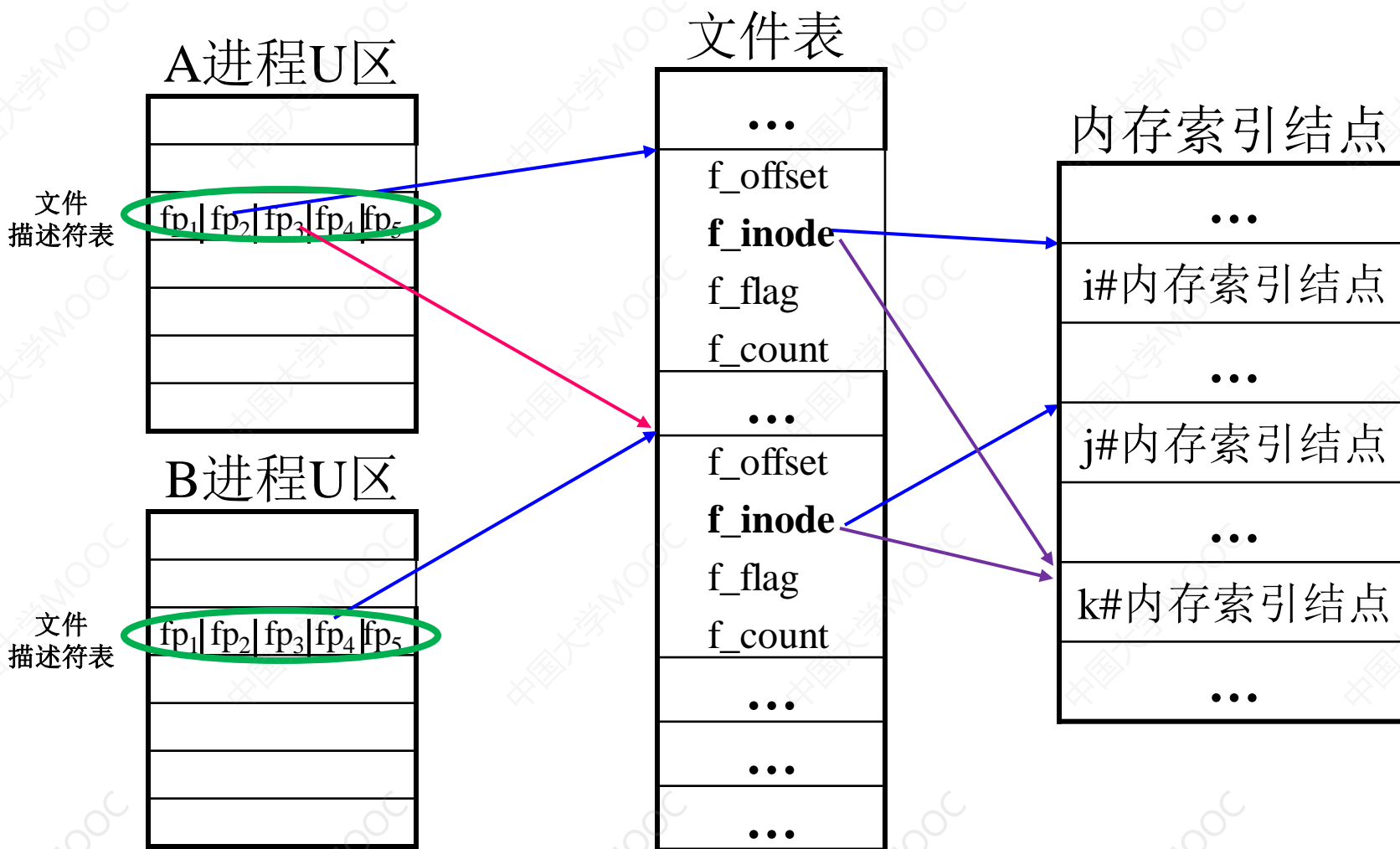
6.1.5 文件操作

目录&文件控制块的支持

文件操作..

- ❑ 创建/删除文件
- ❑ 读/写文件
- ❑ 截断文件
- ❑ 设置文件读/写位置
 - 文件描述符、文件读写指针≠文件指针
- ❑ 打开/关闭文件
 - 内存打开文件表、用户文件描述符表
- ❑ 文件属性设置与获取
- ❑ 目录操作、文件共享及其它操作

UNIX文件描述符表与文件表



6.1 文件和文件系统

6.1.1 文件管理目标及功能

6.1.2 文件、记录和数据项

6.1.3 文件类型

6.1.4 文件系统模型

6.1.5 文件操作

作业题

- ❑ **6.1** 阐述文件管理的目标、功能及技术手段，并就文件系统层次结构模型进行讨论。
- ❑ **6.2** 分别就数据项、记录、关键字和文件的概念进行解释，并就文件的分类展开讨论。
- ❑ **6.3** 通常，文件系统应提供哪些基本文件操作类型？并就其功能实现分别进行简要说明。

第六章 文件管理

6.1 文件和文件系统

6.2 文件的逻辑结构

6.3 外存分配方式

6.4 文件存储空间管理

6.5 目录管理

6.6 文件共享与保护

6.7 数据一致性控制

6.2 文件的逻辑结构

6.2.1 文件逻辑结构及设计要求

6.2.2 文件逻辑结构类型

6.2.3 顺序文件

6.2.4 索引文件

6.2.5 索引顺序文件

文件逻辑结构及设计要求

□ 提高检索效率

- 文件记录或数据检索速度及效率

□ 方便修改

- 文件记录或数据的增、删、修改

□ 降低文件存储成本

- 减少文件占用存储空间
- ~~摒弃存储空间连续性要求~~



6.2 文件的逻辑结构

6.2.1 文件逻辑结构及设计要求

6.2.2 文件逻辑结构类型

6.2.3 顺序文件

6.2.4 索引文件

6.2.5 索引顺序文件

文件逻辑结构的类型

□ 有结构文件（记录式文件）

- ❖ 定长记录和不定长记录

- 顺序文件

- 索引文件

- 分组索引文件（索引顺序文件）

□ 无结构文件

- 基于读写指针和字符进行存取

6.2 文件的逻辑结构

6.2.1 文件逻辑结构及设计要求

6.2.2 文件逻辑结构类型

6.2.3 顺序文件

6.2.4 索引文件

6.2.5 索引顺序文件

顺序文件

❑ 逻辑记录的排序

- 串结构与顺序结构

与关键字次序
一致与否

❑ 顺序文件读写操作

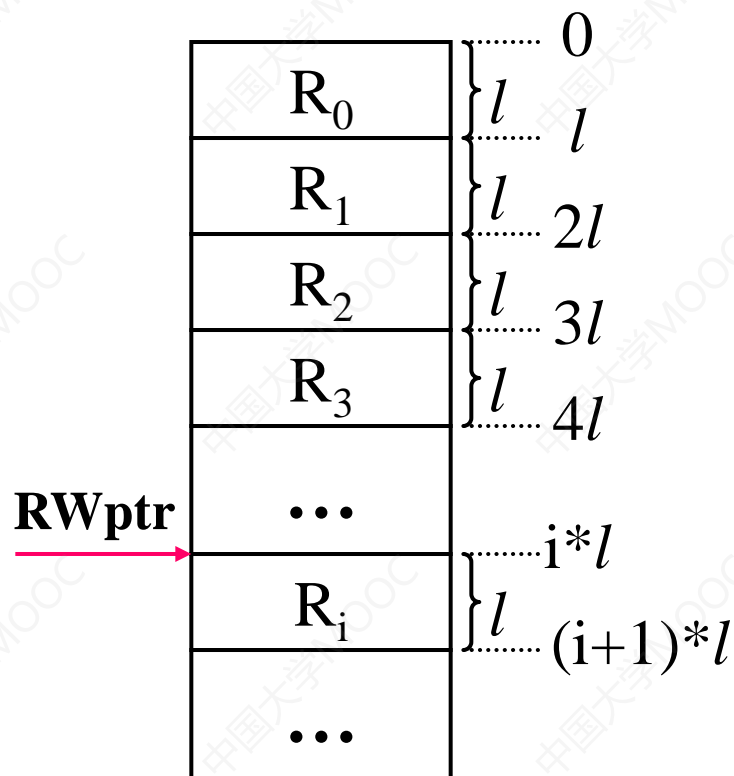
- ❖ 读写指针 $RWptr \Leftrightarrow$ 对应记录逻辑地址
- 定长记录 $RWptr += recordLength$
- 变长记录 $RWptr += currentRecordLength$

❑ 顺序文件评析

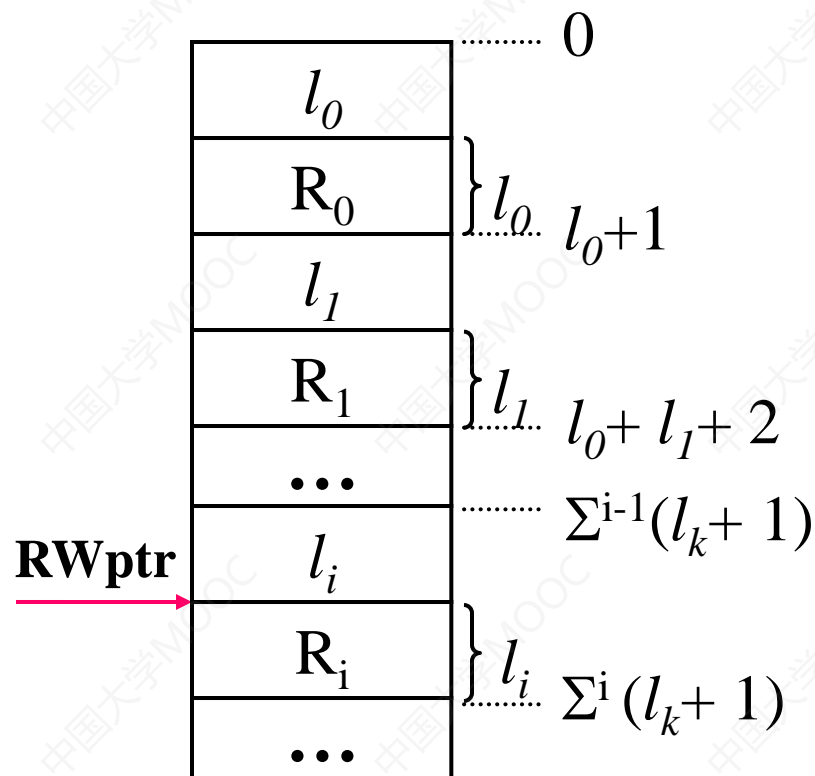
- 适于批量存取及磁带介质
- 交互应用场合单个记录操作低效

解决方案?

顺序文件直接存取分析



定长记录顺序文件



变长记录顺序文件

6.2 文件的逻辑结构

6.2.1 文件逻辑结构及设计要求

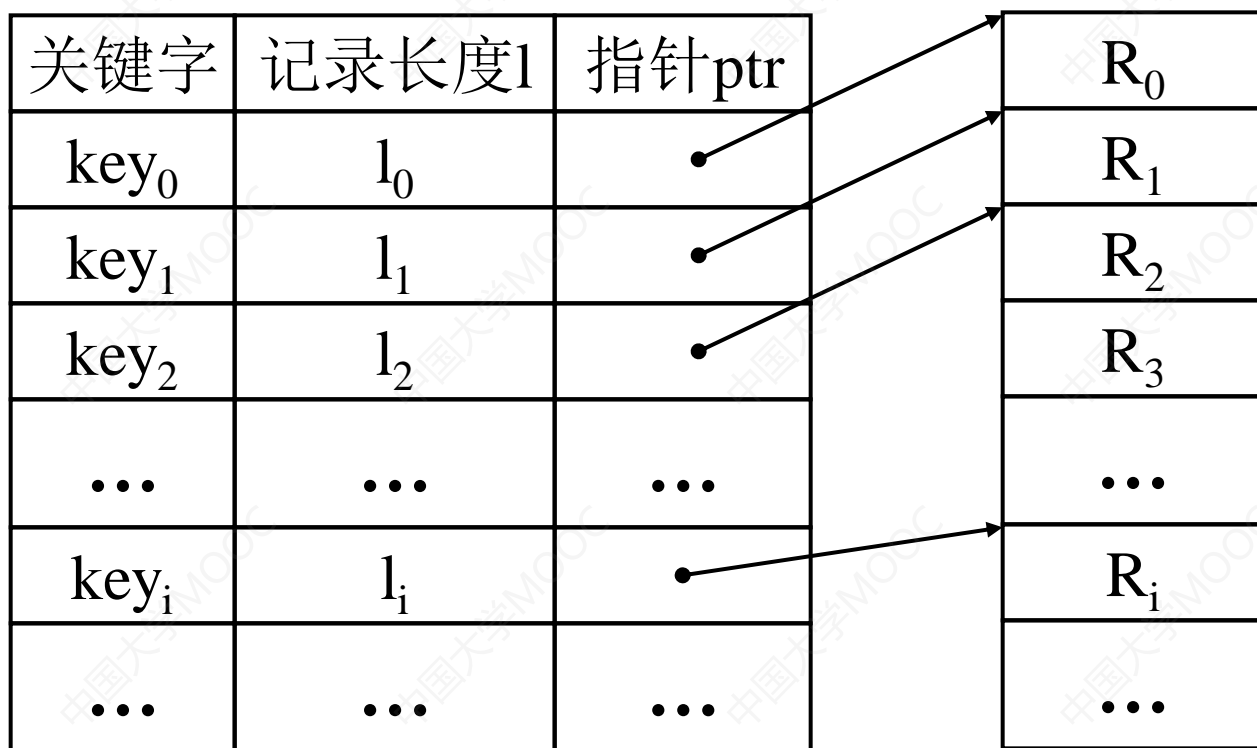
6.2.2 文件逻辑结构类型

6.2.3 顺序文件

6.2.4 索引文件

6.2.5 索引顺序文件

索引文件组织及检索机制



索引表（定长记录顺序文件）

主数据文件

6.2 文件的逻辑结构

6.2.1 文件逻辑结构及设计要求

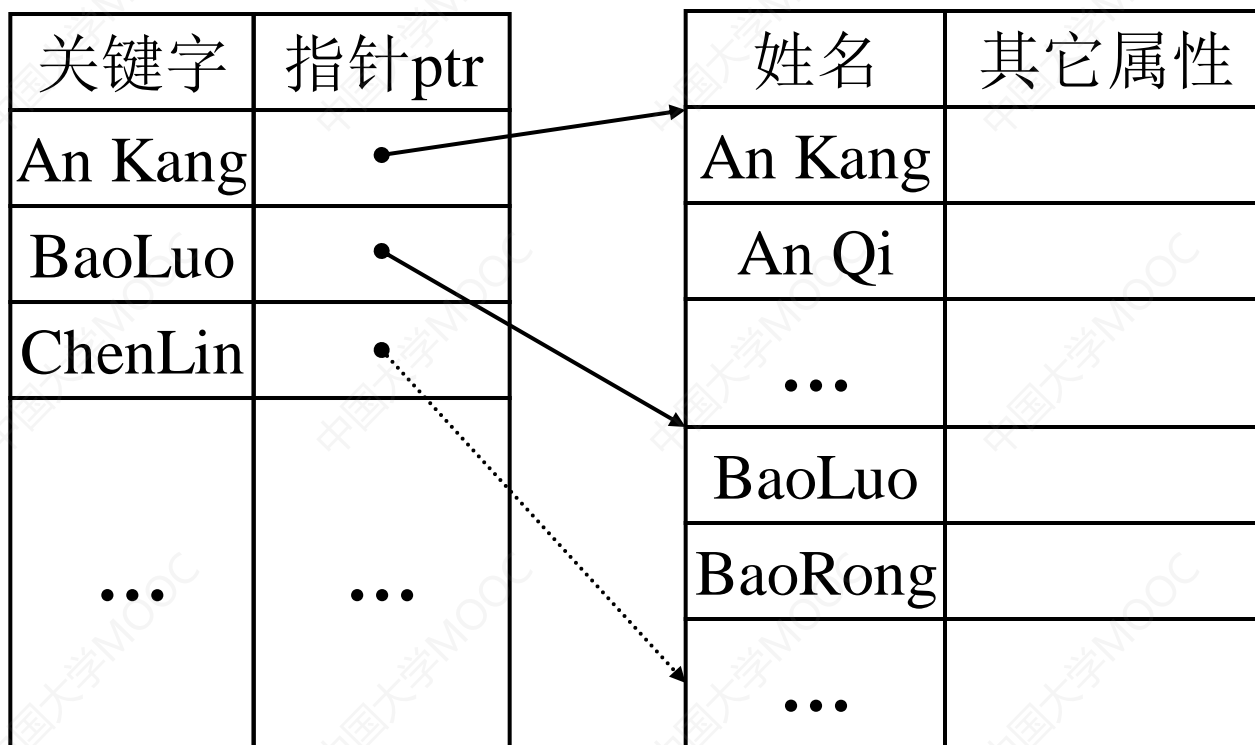
6.2.2 文件逻辑结构类型

6.2.3 顺序文件

6.2.4 索引文件

6.2.5 索引顺序文件

索引顺序文件组织及检索机制



索引表

(定长记录顺序文件)

主数据文件

索引顺序文件检索效率分析

- 对于拥有 N 条记录
基于顺序查找
字的记录，不

$$[1+2+3+\dots+(N-1)+N]/N$$

$$= [N \times (N+1)] / (2 \times N)$$

$$= (N+1)/2$$

系统检索开销

$$[\sqrt{N}+1]/2 + [\sqrt{N}+1]/2$$

$$= \sqrt{N}+1$$

- 顺序文件
- 索引文件组织方式 $(N+1)/2$ 条
- 索引顺序文件组织方式 $\sqrt{N}+1$ 条

❖ 分组大小 \sqrt{N} 条记录

- 举例说明($N=10000$)

主数据文件1000000条记录, 共10000分组;
一级索引顺序文件10000索引项, 共100分组;
二级索引顺序文件100个索引项

$$\begin{aligned} & (100+1)/2 + (100+1)/2 + (100+1)/2 \\ & = 303/2 \\ & = 151.5 \end{aligned}$$

❖ 检索开销 100

➤ 两级索引顺序文件组织方式

❖ 主文件分组大小 **100** 条记录 [$\text{CubicRoot}(N)$]

❖ 一级索引表分组大小 **100** 条记录 [$\text{CubicRoot}(N)$]

❖ 检索开销 **151.5**条 [$1.5 * \text{CubicRoot}(N) + 1.5$]

6.2 文件的逻辑结构

6.2.1 文件逻辑结构及设计要求

6.2.2 文件逻辑结构类型

6.2.3 顺序文件

6.2.4 索引文件

6.2.5 索引顺序文件

作业题

- **6.4** 什么是文件的逻辑结构？其基本设计要求是什么？并就主要的文件逻辑结构类型特别是顺序文件、索引文件、索引顺序文件等从组织结构、检索方法、检索速度和存储费用等方面展开简明扼要的说明和讨论。

第六章 文件管理

6.1 文件和文件系统

6.2 文件的逻辑结构

6.3 外存分配方式

6.4 文件存储空间管理

6.5 目录管理

6.6 文件共享与保护

6.7 数据一致性控制

磁盘：磁盘卷（分区）：
【(磁盘/文件)元数据+文件】

6.3 外存分配方式

6.3.1 文件物理结构与外存分配

6.3.2 连续分配

6.3.3 链接分配

6.3.4 索引分配

6.3.5 直接文件和散列文件

文件物理结构与外存分配

- 文件物理结构（存储结构）
 - 文件在外存上的存储组织形式
- 外存空间分配方法设计目标
 - 空间利用及文件访问速度
- 外存分配方式与文件物理结构
 - 连续分配 \Leftrightarrow 顺序文件结构
 - 链接分配 \Leftrightarrow 链接式文件结构
 - 索引分配 \Leftrightarrow 索引式文件结构
 - 直接文件与散列文件

} 离散分配

6.3 外存分配方式

6.3.1 文件物理结构与外存分配

6.3.2 连续分配

6.3.3 链接分配

6.3.4 索引分配

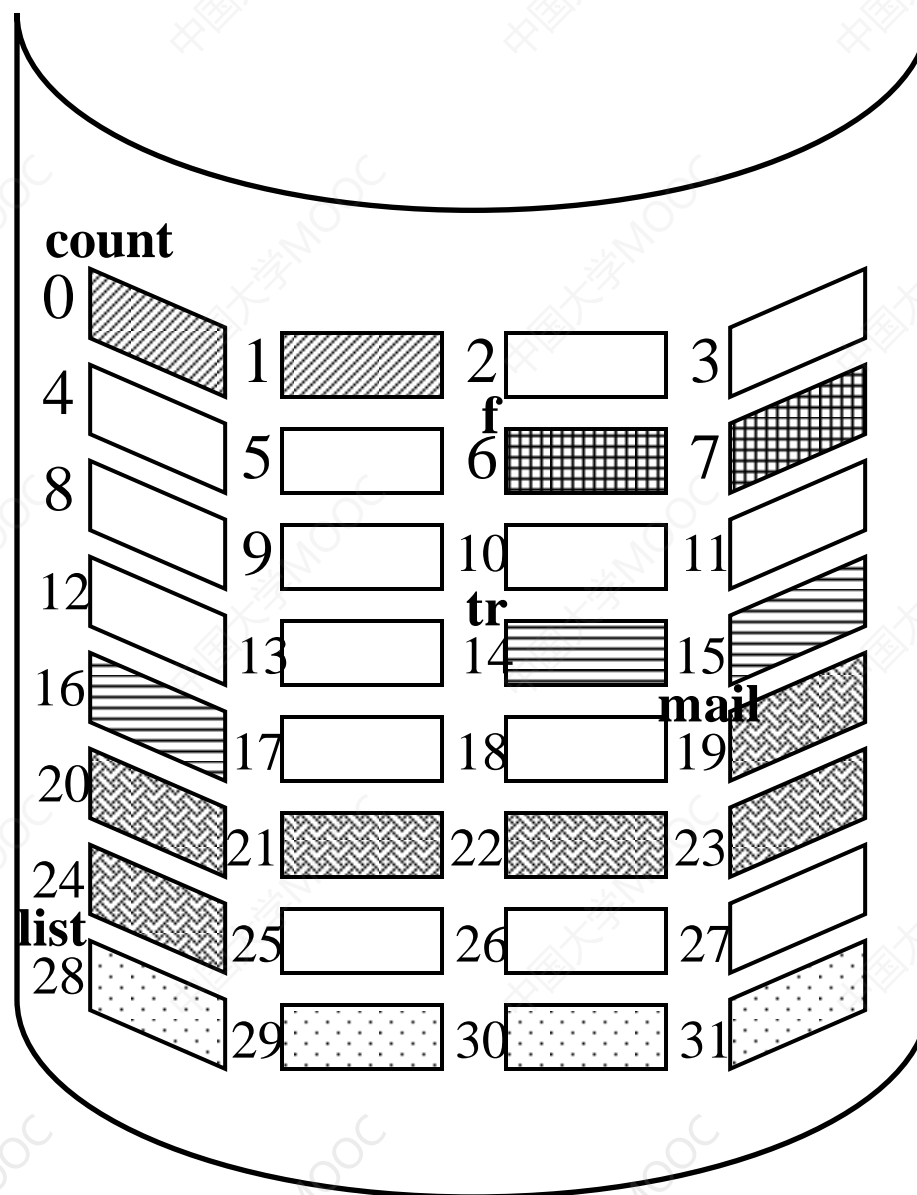
6.3.5 直接文件和散列文件

连续分配

示意图

文件目录

文件名	起始盘块号	盘块数
count	0	2
tr	14	3
mail	19	6
list	28	4
f	6	2



连续分配评析

❑ 外部碎片问题

- 磁盘空间分割后形成的较小的无法存储文件的连续区
- 紧凑方法



磁盘整理工具

❑ 主要优点

- 顺序访问容易且速度快
- 支持直接存取

❑ 主要缺点

- 要求有连续的存储空间，空间利用率低
- 必须事先知道文件的长度

6.3 外存分配方式

6.3.1 文件物理结构与外存分配

6.3.2 连续分配

6.3.3 链接分配

6.3.4 索引分配

6.3.5 直接文件和散列文件

链接分配

□ 基本思想

- 支持离散分配方式，通过盘块上的指针实现同一文件多个离散盘块的链接

□ 优点评析

- 消除了外部碎片，外存空间利用率高
- 按需分配，且无需事先知道文件长度
- 支持文件动态增长，方便文件增删改

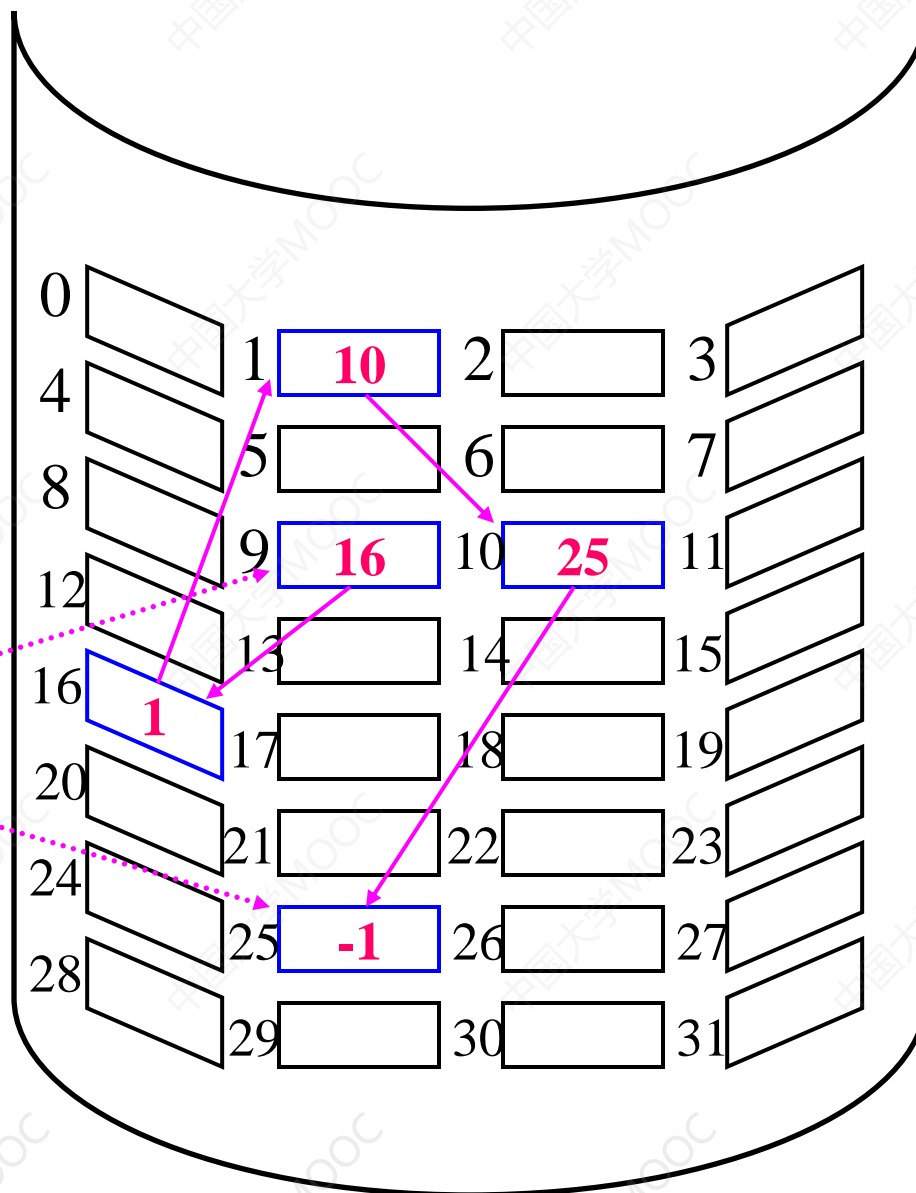
□ 链接方式

- 隐式链接/显式链接

隐式链接 示意图

文件目录

文件名	起始盘块号	末尾盘块号
jeep	9	25



隐式链接问题及对策

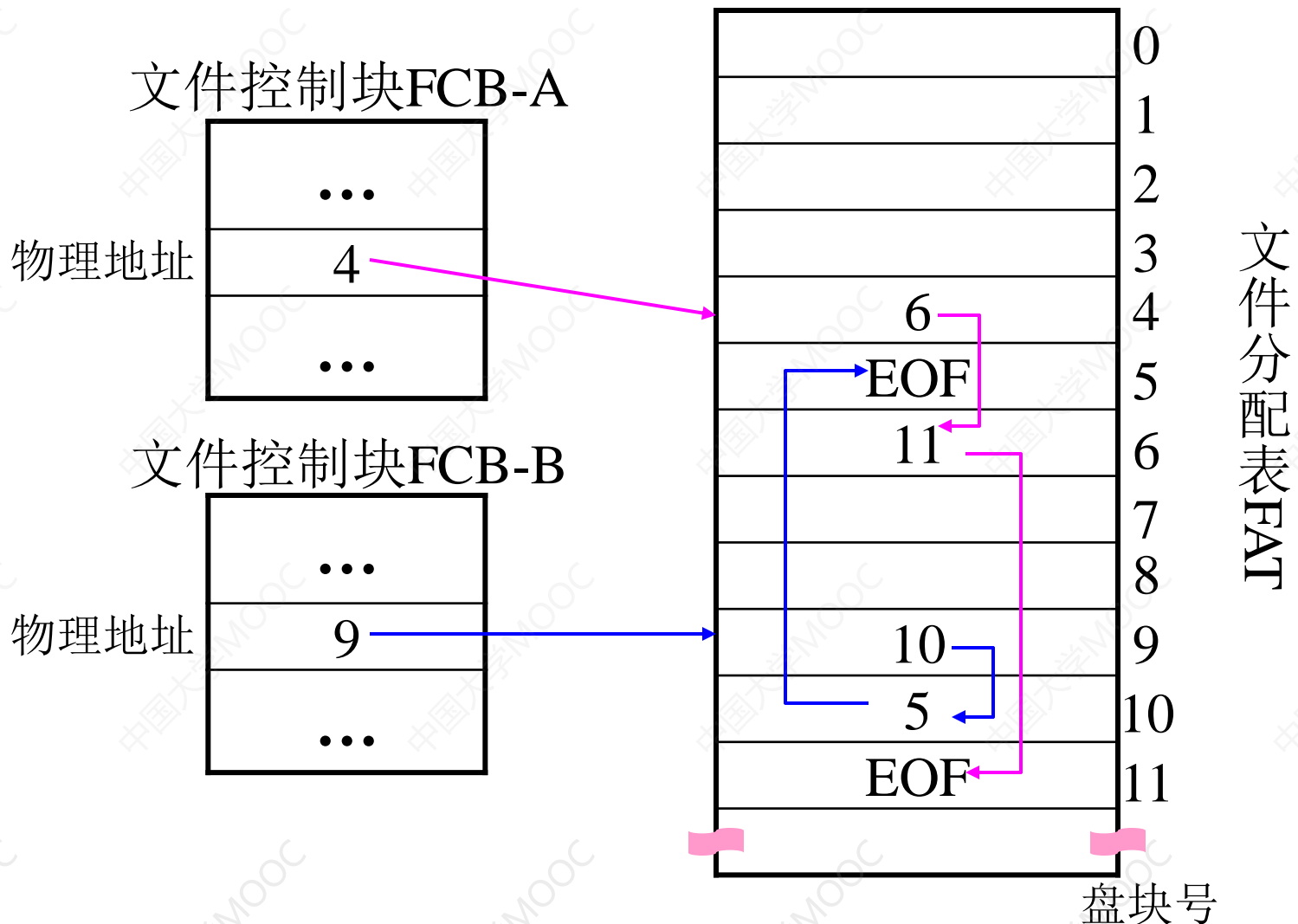
□ 主要问题

- 只适合顺序访问，对随机存取极其低效
- 仅通过链接指针实现离散各盘块的链接，只要其中任何一个指针出现问题，都会导致整条链的断开，所以可靠性较差

□ 为了提高检索速度和减少指针所占用的存储空间，可将几个盘块组成一个簇，以簇为单位进行盘块分配

- 缺点：内部碎片增大

显式链接与文件分配表FAT



文件分配表空间开销计算

□ 设定盘块大小为1KB

➤ 对于1.2MB的软盘，共有盘块

$$1.2\text{MB}/1\text{KB} = 1.2\text{K} \in (2^8, 2^{12})$$

故文件分配表表项取12位即1.5B

所以FAT共需空间 $1.2\text{K} \times 1.5\text{B} = 1.8\text{KB}$

➤ 对于200MB的硬盘，共有盘块

$$200\text{MB}/1\text{KB} = 200\text{K} \in (2^{16}, 2^{20})$$

故采用FAT32格式，文件分配表表项取32位即4字节，所以FAT共需空间 $200\text{K} \times 4\text{B} = 800\text{KB}$

6.3 外存分配方式

6.3.1 文件物理结构与外存分配

6.3.2 连续分配

6.3.3 链接分配

6.3.4 索引分配

6.3.5 直接文件和散列文件

索引分配的引入及基本思想

❑ 链接分配方式的缺陷

- 不能支持高效地直接存取
 - ❖ 若对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
- FAT占用内存空间较大及文件所占盘块号随机

❑ 基本思想

- 文件打开仅须把**该文件**所占用盘块编号调入内存即可，故可将每个文件所对应的盘块号集中地存放一个所谓的**索引块**中，形成一个索引表
- 索引块 \Leftrightarrow 文件 \Leftrightarrow 文件目录项 \Leftrightarrow 索引块指针
- 对于小文件，索引块利用率极低

索引分配

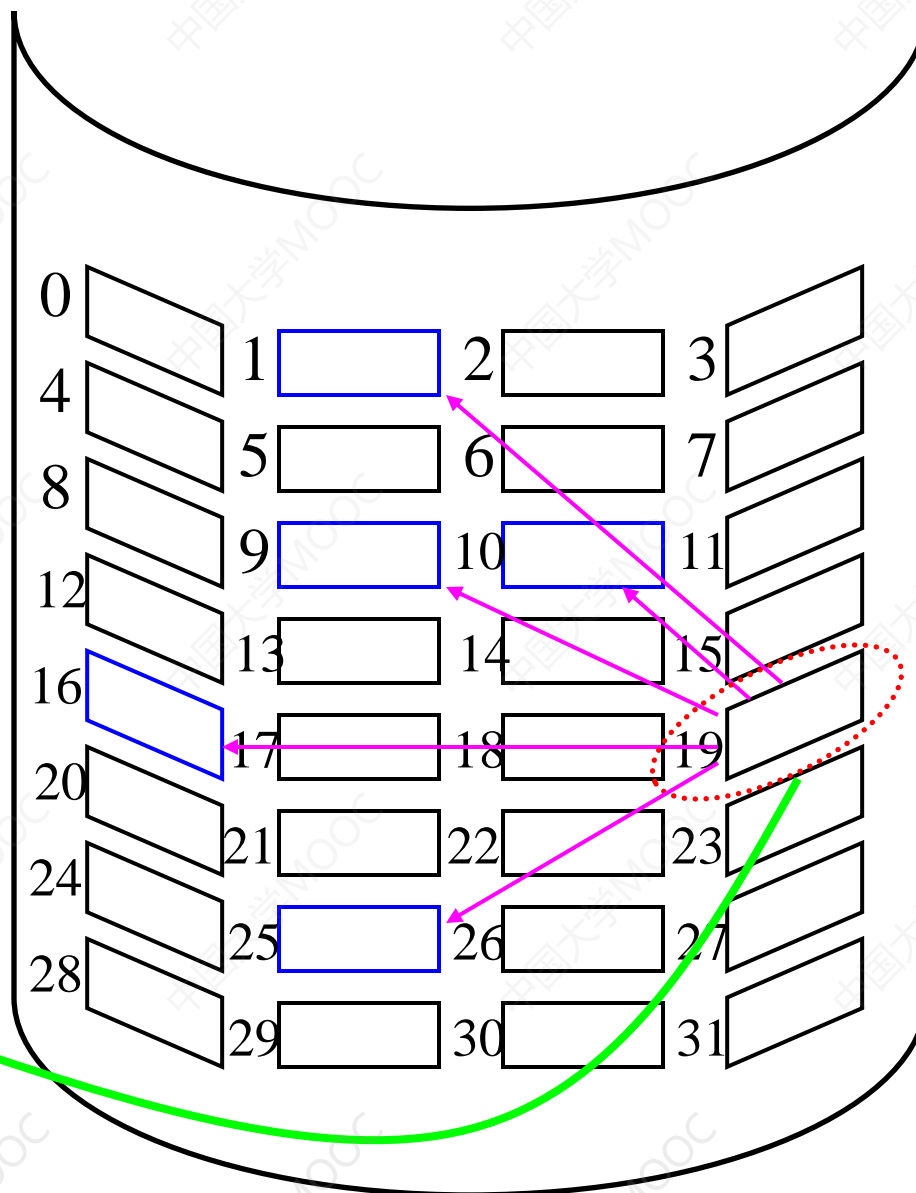
示意图

文件
目录

文件名 索引盘块号
jeep 19

19#
盘块
(索引盘块)

9
16
10
25
1
-1
-1



两/多级索引分配

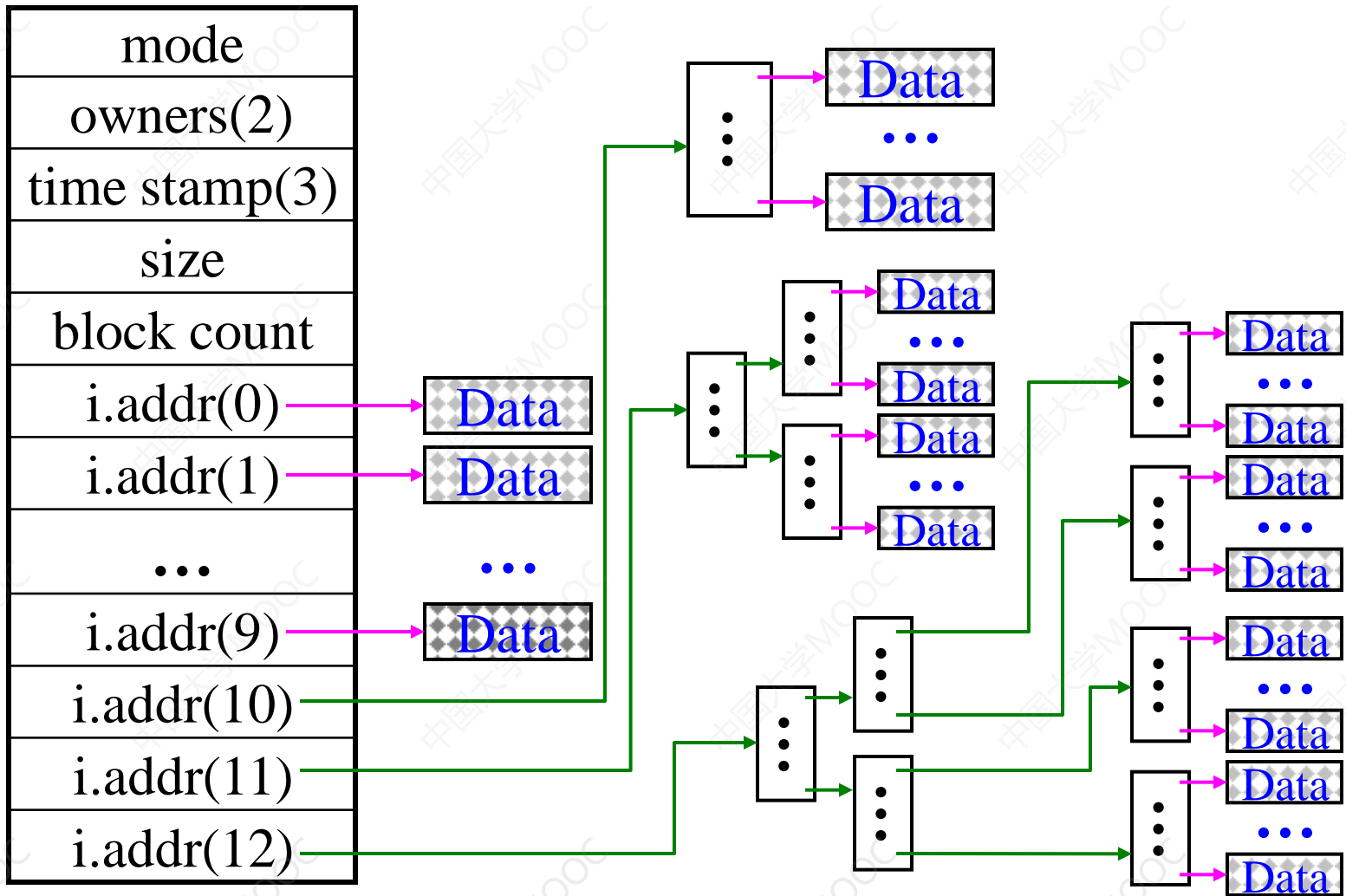
□ 基本思想

- 对于太大的文件和索引块太多时，直接用链接指针来链接索引块的方法显然是低效的，为此应引入多级索引分配方式

□ 允许文件最大长度

- 两级索引、盘块大小1KB、盘块号占4B
则一个索引块可含 $1\text{KB}/4\text{B}=256$ 个盘块号，
于是两级索引最多可含 $256 \times 256 = 64\text{K}$
个盘块号，允许文件最大长度为64MB

混合分配方式示意图



混合分配方式(UNIX系统)

□ 直接寻址

- ❖ 直接地址项存放对应文件数据的盘块的盘块号
- 盘块大小4KB、盘块号占4B，则支持长度在 $4\text{KB} \times 10 = 40\text{KB}$ 以内的文件

□ 一次间接寻址

- ❖ $i.\text{addr}(10)$ 指向对应文件的一级索引块
- 一级索引块可含 $4\text{KB}/4\text{B} = 1\text{K}$ 个盘块号，故支持长度在 $(4\text{KB} \times 1\text{K} = 4\text{MB}) + 40\text{KB}$ 以内的文件

□ 多次间接寻址

- ❖ $i.\text{addr}(11)$ 、 $i.\text{addr}(12)$ 分别指向对应文件的两级索引块和三级索引块，所以支持文件长度可达 $(4\text{KB} \times 1\text{K} \times 1\text{K} \times 1\text{K} = 4\text{TB}) + (4\text{KB} \times 1\text{K} \times 1\text{K} = 4\text{GB}) + 4\text{MB} + 40\text{KB}$

UNIX文件操作地址转换过程

I. 将字节偏移量转换为文件逻辑盘块号

➤ $\text{LogicBlk\#} = \text{Offset} / \text{SizeOfBlk}$

II. 将逻辑盘块号转换为物理盘块号

A. 确定物理盘块号所在地址项或索引盘块

B. 确定物理盘块号所在索引盘块及位置

□ 举例说明（假定盘块大小1KB）

➤ Offset 9000B

➤ Offset 14000B

6.3 外存分配方式

6.3.1 文件物理结构与外存分配

6.3.2 连续分配

6.3.3 链接分配

6.3.4 索引分配

6.3.5 直接文件和散列文件

直接文件和散列文件

□ 直接文件

- 记录键值 => 物理地址

□ 散列文件（Hash文件）

- 散列函数及目录表



6.3 外存分配方式

6.3.1 文件物理结构与外存分配

6.3.2 连续分配

6.3.3 链接分配

6.3.4 索引分配

6.3.5 直接文件和散列文件

作业题

- **6.5** 什么是文件的物理结构？其基本设计要求是什么？并就主要的文件物理结构类型及外存分配方式展开简明扼要的说明和讨论。
- **6.6[必做]** 假定盘块的大小为4KB，对于10GB的硬盘，其文件分配表需占用多少存储空间？当硬盘容量为40GB时，文件分配表又需占用多少存储空间？
- **6.7[必做]** 假如盘块的大小为4KB，每个盘块号占4个字节，在两级索引分配时，允许的最大文件是多少？

第六章 文件管理

6.1 文件和文件系统

6.2 文件的逻辑结构

6.3 外存分配方式

6.4 文件存储空间管理

6.5 目录管理

6.6 文件共享与保护

6.7 数据一致性控制

6.4 文件存储空间管理

6.4.1 管理目标与技术要领

6.4.2 空闲表法

6.4.3 空闲链表法

6.4.4 位示图法

6.4.5 成组链接法

管理目标与技术要领

- 文件存储空间管理目标
 - 外存空间利用与文件访问速度
- 文件存储空间管理技术要领
 - 连续分配与离散分配优劣各异
 - 文件物理结构与存储组织方式
 - 分配以掌握空间使用情况为前提
 - A. 存储空间使用情况数据结构
 - B. 存储空间的分配与回收
 - C. 基本分配单位为磁盘块

6.4 文件存储空间管理

6.4.1 管理目标与技术要领

6.4.2 空闲表法

6.4.3 空闲链表法

6.4.4 位示图法

6.4.5 成组链接法

空闲表法

□ 属于连续分配方式。

□ 空闲盘区的分配

- 首次适应算法
- 循环首次适应算法
- 最佳适应算法
- 最坏适应算法

□ 空闲盘区的回收

- 考虑回收区域插入点前后盘块区是否相邻接，相邻则予合并

适用场合？

空闲盘块（区）表

序号	第一空闲盘块号	空闲盘块数
0	3	3
1	8	5
2	16	2
3

6.4 文件存储空间管理

6.4.1 管理目标与技术要领

6.4.2 空闲表法

6.4.3 空闲链表法

6.4.4 位示图法

6.4.5 成组链接法

空闲链表法

□ 将所有空闲盘块（区）拉成一条空闲链

➤ 空闲盘块链

- ❖ 以盘块为基本元素链接
- ❖ 分配与回收

➤ 空闲盘区链

- ❖ 以盘区为基本元素链接，且每个盘区除含有指示下一个盘块区的指针外，还应标有指明本盘区大小的信息
- ❖ 分配与回收
- ❖ 为提高对空闲盘区的检索速度，可采用显式链接方式

6.4 文件存储空间管理

6.4.1 管理目标与技术要领

6.4.2 空闲表法

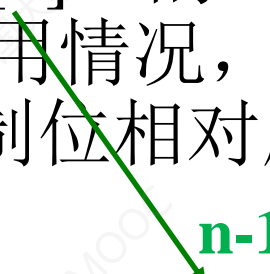
6.4.3 空闲链表法

6.4.4 位示图法

6.4.5 成组链接法

位示图

- 利用位示图（即二维数组Map[m][n]）的一位（0/1）来表示一个磁盘盘块的使用情况，使磁盘上所有盘块）都与一个二进制位相对应



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	1	1	0	0	0	1	0	0	1	0	1	0	0	1	1	0
1	1	0	0	1	0	0	0	1	0	1	0	0	1	1	1	1
2	1	1	0	1	1	0	0	1	0	1	0	0	1	0	1	0
...															
m-1	1	1	0	1	0	0	1	0	1	0	1	0	0	0	1	0

物理块（盘块）的分配

Var Map: array [m][n] of bit;

- 顺序扫描位示图，找出一个或一组其值均为空闲的二进制位
- 将所找到的一个或一组二进制位 Map[i][j] 的行/列号转换为与之对应的盘块号 b : $b = n \times i + j$
- 按盘块号分配盘块，同时修改位示图

物理块（盘块）的回收

- I. 将回收盘块的盘块号b转换为位示图中的行号i和列号j:
 - A. $i = b \text{ DIV } n$;
 - B. $j = b \text{ MOD } n$;
- II. 按盘块号回收盘块
- III. 根据回收盘块对应二进制位的行/列号修改位示图

6.4 文件存储空间管理

6.4.1 管理目标与技术要领

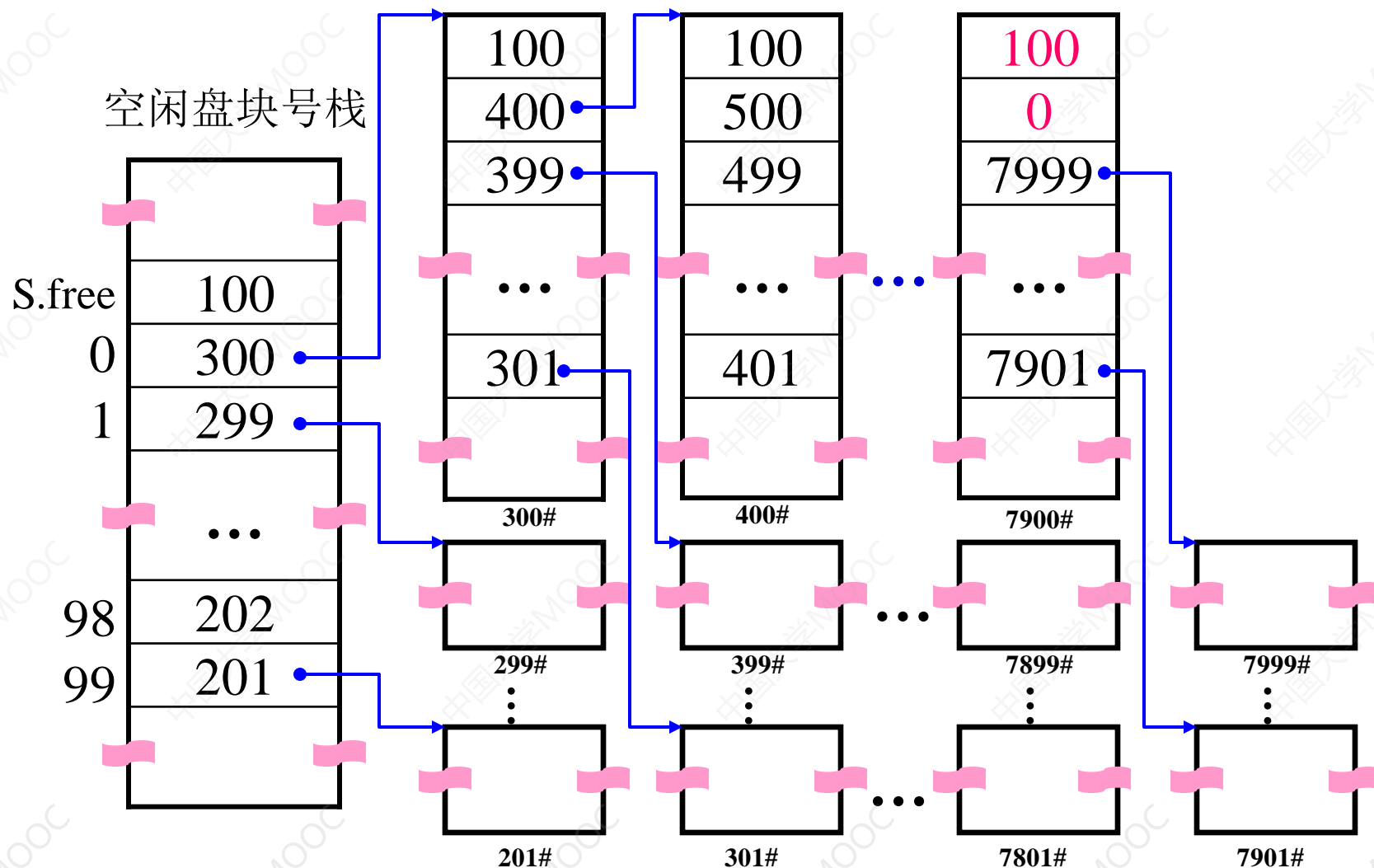
6.4.2 空闲表法

6.4.3 空闲链表法

6.4.4 位示图法

6.4.5 成组链接法

成组链接法



空闲盘块的分配

I. 检查空闲盘块号栈是否上锁:

- a) 如未上锁，则从栈顶取出一空闲盘块号，
 - 1) 若其不是栈底 $S.free(0)$ 即空闲盘块号数 $N \neq 1$ ，则将其对应盘块分配给用户，然后将栈顶指针下移一格，亦即做 N 的减1操作
 - 2) 若其是栈底 $S.free(0)$ [即 $N=1$]且其值为0，则分配失败返回；否则调用磁盘读过程，将栈底盘块号所对应盘块的内容读入栈中作为新的空闲盘块号栈的内容，同时将该盘块号对应的盘块分配给用户
- b) 如上锁，则等待

II. 分配盘块缓冲，然后返回

空闲盘块的回收

I. 检查空闲盘块号栈，若未上锁：

- a) 若栈中空闲盘块号数 $N < 100$ 即栈未满，则执行空闲盘块号数 N 的加1操作，并将回收盘块的盘块号记入空闲盘块号栈的栈顶
- b) 否则，若栈中空闲盘块号数 N 已达100即栈已满，应将空闲盘块号栈的当前内容包括盘块数及所有盘块号记入新回收的盘块中，同时将新回收盘块的盘块号作为 $S.free(0)$ 即栈底，并置空闲盘块号栈的空闲盘块数 N 为1
- c) 返回

II. 否则已上锁，等待

6.4 文件存储空间管理

6.4.1 管理目标与技术要领

6.4.2 空闲表法

6.4.3 空闲链表法

6.4.4 位示图法

6.4.5 成组链接法

作业题

- **6.8** 文件存储空间管理的基本目标是什么？关于空闲存储空间的管理通常可基于哪些数据结构及方法进行描述和管理？并逐一进行简要说明。
- **6.9[必做]** 计算机系统利用**位示图**（**详参作业手册**）来管理空闲盘块，盘块大小为1KB，现要为某文件分配两个盘块，试说明盘块分配的具体过程。
- **6.10[必做]** 某计算机系统磁盘容量为520MB，盘块大小为1KB。其中前4MB用于存放索引结点等，后10MB用作对换区，采用成组链接法管理外存空间，每组100个盘块。试画出外存尚未使用的成组链接图。

第六章 文件管理

6.1 文件和文件系统

6.2 文件的逻辑结构

6.3 外存分配方式

6.4 文件存储空间管理

6.5 目录管理

6.6 文件共享与保护

6.7 数据一致性控制

6.5 目录管理

6.5.1 目录管理基本要求

6.5.2 文件控制块

6.5.3 索引结点

6.5.4 目录结构

6.5.5 目录查询技术

目录管理功能及要求

□ 目录引入理由

- 文件的有效管理与组织要求基于文件名便能快速、准确地找到指定文件

□ 文件目录管理的功能及要求

- 实现“按名存取”（文件名 \Rightarrow 外存地址）
- 提高目录检索速度及文件存取速度
- 文件共享（外存保留一份文件副本）
- 允许文件重名，以便于文件使用

6.5 目录管理

6.5.1 目录管理基本要求

6.5.2 文件控制块

6.5.3 索引结点

6.5.4 目录结构

6.5.5 目录查询技术

文件控制块

□ 基本概念

- 文件↔文件控制块↔文件目录项
- 文件目录↔文件控制块有序集合↔目录文件

□ 主要信息内容

- 基本信息（文件名、物理位置、结构类型）
- 存取控制信息（各类用户存取权限）
- 使用信息（建立/上次修改日期及时间）
 - ❖ 当前使用信息（当前已打开该文件的进程数、是否被其它进程锁住、文件在内存是否已修改但尚未拷贝到盘上）

MS

览

文件控制块即目录共占据
 $112 * 32 / 512 = 112 / 16 = 7$ 个盘块（此
处即扇区），于是假设每次目录读
取仅拷贝一个目录盘块到内存，则
目录检索平均需启动软盘驱动器
 $(7+1)/2 = 4$ 次

文件长度
数

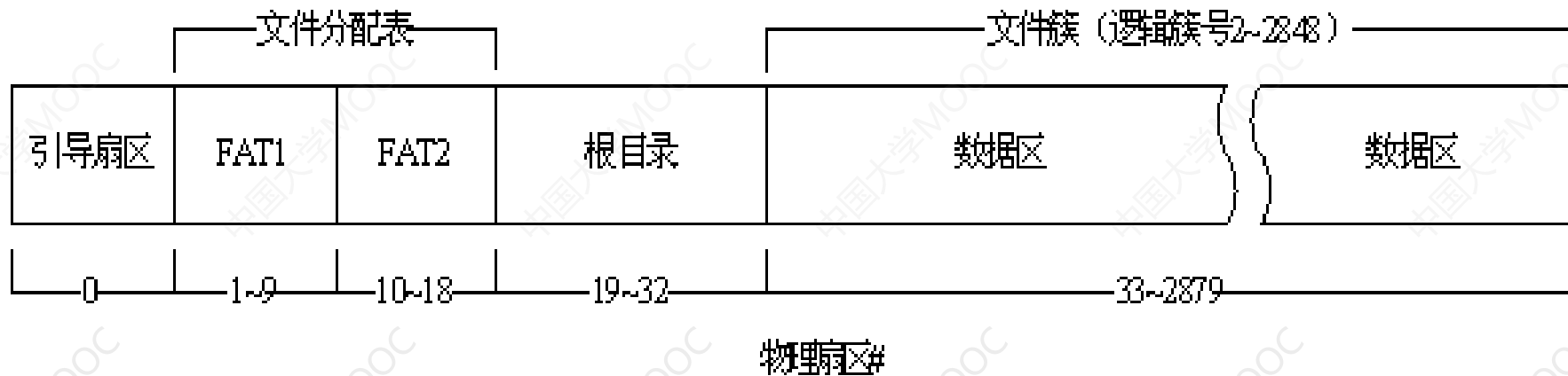
- 文件控制块长度
 ➤ 文件名及文件扩展名共11个字符
- 对于360KB的软盘，总共含有112个FCB，
故占用 **$7 * 512B$** 的存储空间。目录检索开销？

FAT12文件系统标准

- ❑ 引导扇区
- ❑ 文件分配表
- ❑ 目录项
- ❑ 逻辑簇号 \leftrightarrow 物理扇区号
- ❑ 磁盘分区组织结构

不同FAT文件系统支持的
磁盘空间大小及文件数？

➤ 1.44MB软盘组织结构



6.5 目录管理

6.5.1 目录管理基本要求

6.5.2 文件控制块

6.5.3 索引结点

6.5.4 目录结构

6.5.5 目录查询技术

索引结点的引入

□ 必要性

- 文件目录存放在磁盘上，且可能要占用大量盘块 N ，检索开销很大（盘块调入次数 $[N+1]/2$ ）

□ 可行性

- 只有文件名对目录检索有用

文件名	索引结点编号
文件名1	
文件名2	
文件名3	

UNIX文件目录

文件描述控制信息

索引结点（i结点）集合

索引结点检索开销分析

□ 举例说明

- 盘块大小1KB，文件目录共3200个FCB

□ 引入索引结点前（FCB占64B）

- 每盘块包含□个FCB，文件目录共需占用□个盘块，故查找一个文件平均需启动磁盘□次

□ 引入索引

假设每次目录读取仅拷贝一个目录盘块到内存，则目录检索平均需启动磁盘驱动器 $(1+2+3+\dots+200)/200$
 $= [200*(1+200)/2]/200 = 100.5$ 次

磁盘索引结点与内存索引结点

□ 磁盘索引结点

- 文件 ⇔ 磁盘索引结点
- 主要内容（文件主标识、文件类型、文件存取权限、文件长度、文件存取时间、文件物理地址、文件连接计数）

□ 内存索引结点

- 文件打开时对磁盘索引结点在内存的拷贝
- 专有内容（索引结点编号、索引结点状态、索引结点访问计数、文件所属文件系统逻辑设备号、空闲链表/散列队列指针）

6.5 目录管理

6.5.1 目录管理基本要求

6.5.2 文件控制块

6.5.3 索引结点

6.5.4 目录结构

6.5.5 目录查询技术

单级目录结构

□ 文件创建

- 目录检索⇒申请空目录项⇒属性设置

□ 文件删除

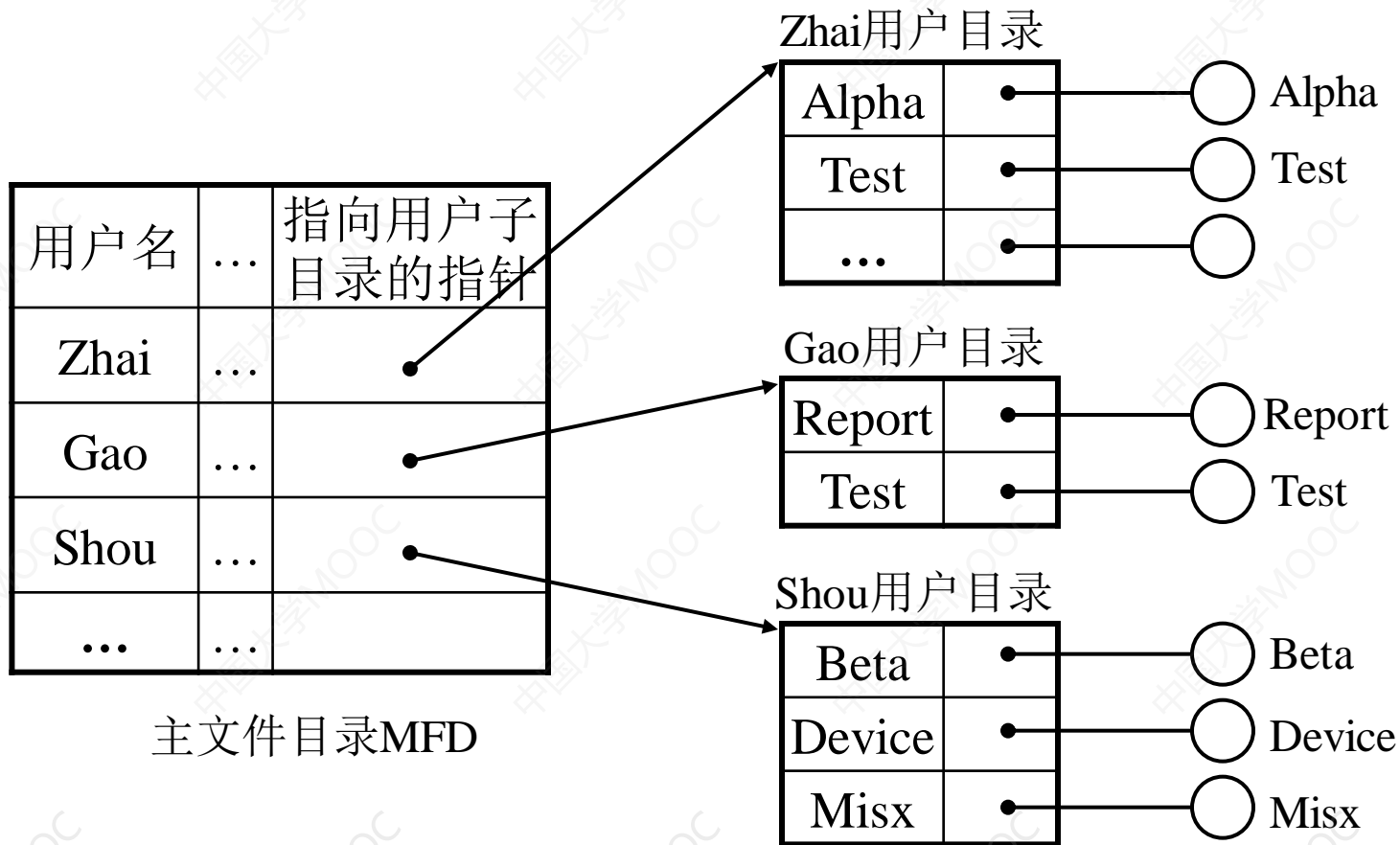
- 目录检索⇒外存空间回收⇒目录项回收

□ 缺点

- 查找速度慢、不允许重名、不便于文件共享

文件名	状态位	物理地址	文件其它属性
Alpha.wps			
Report.ppt			
Text.dat			
...			

两级目录结构示意图



两级目录结构评析

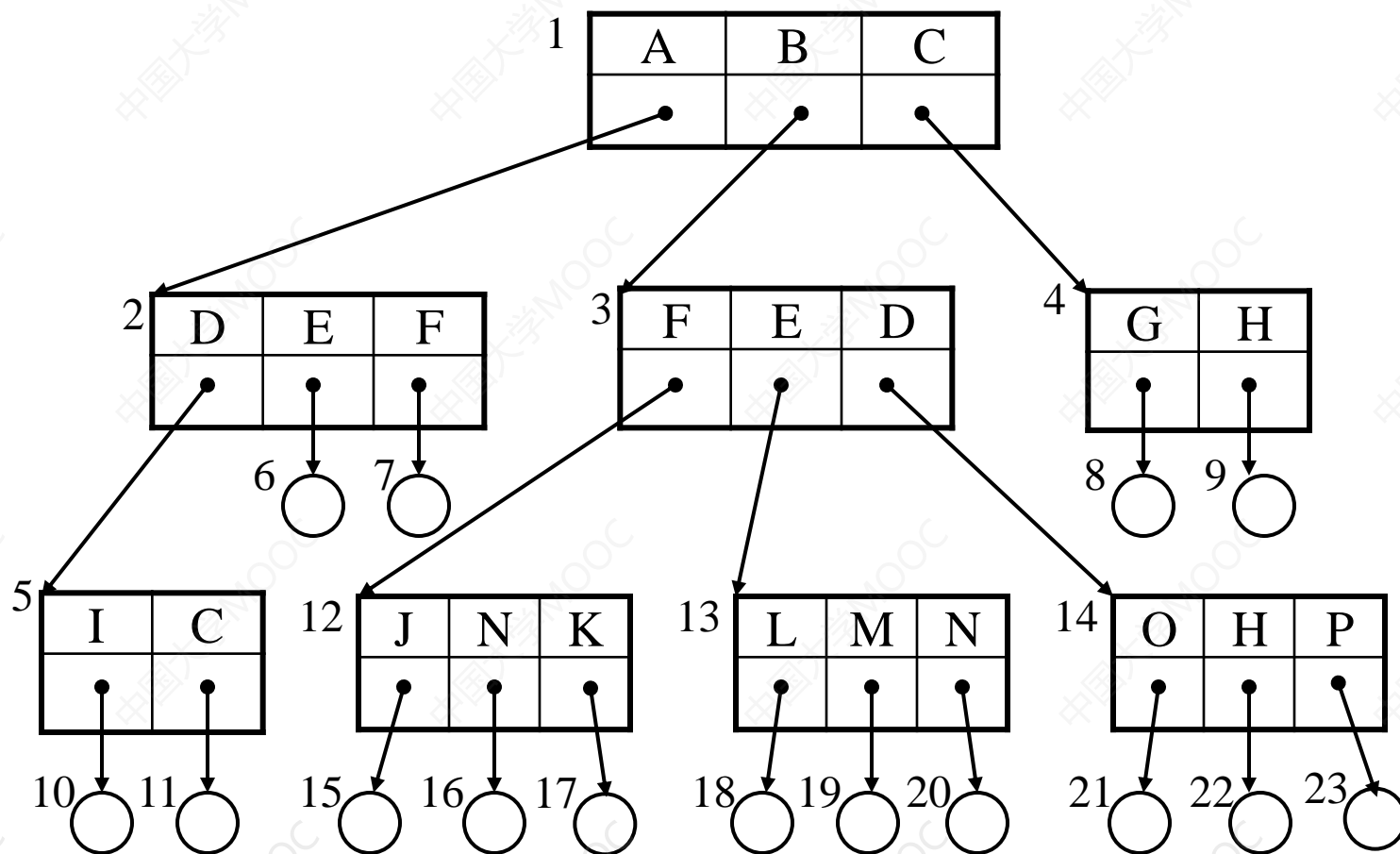
□ 优点

- 提高了目录检索速度
 - ❖ MFD含 n 个UFD，UFD最多有 m 个目录项，则为找到指定文件（用户+文件名）最多需检索 个目录项；对于单级目录结构，最多需检索 个目录项
- 不同用户目录下可使用相同的文件名
- 不同用户可用不同名称访问同一共享文件

□ 缺点

- 用户隔离不便于文件共享和用户间协作

树型目录结构示意图



树型目录结构实现机制及评析

□ 路径名（绝对路径名）

- 从根目录到各数据文件之间只有唯一的通路，该路径上的全部目录文件名与数据文件名用“/”连接形成特定数据文件的路径名

□ 相对路径名

- 当前目录（工作目录）
- 从当前目录开始逐级通过中间目录文件最后到达所访问数据文件的路径名称为相对路径名

□ 目录增删

- 目录检索作为第一步
- 目录删除处理方法（是否允许删除非空目录）

□ 满足目录管理的所有要求

6.5 目录管理

6.5.1 目录管理基本要求

6.5.2 文件控制块

6.5.3 索引结点

6.5.4 目录结构

6.5.5 目录查询技术

文件存取及目录查询技术

□ 实现文件按名存取的基本步骤

1. 系统根据用户提供的文件名，对文件目录进行查询，找出该文件的文件控制块或索引结点
2. 按照对应文件控制块或索引结点中所记录的文件物理地址（盘块号），计算出文件在磁盘上的物理地址
3. 启动磁盘驱动程序，将所存取的文件读入内存进行具体读写操作

冲突处理？

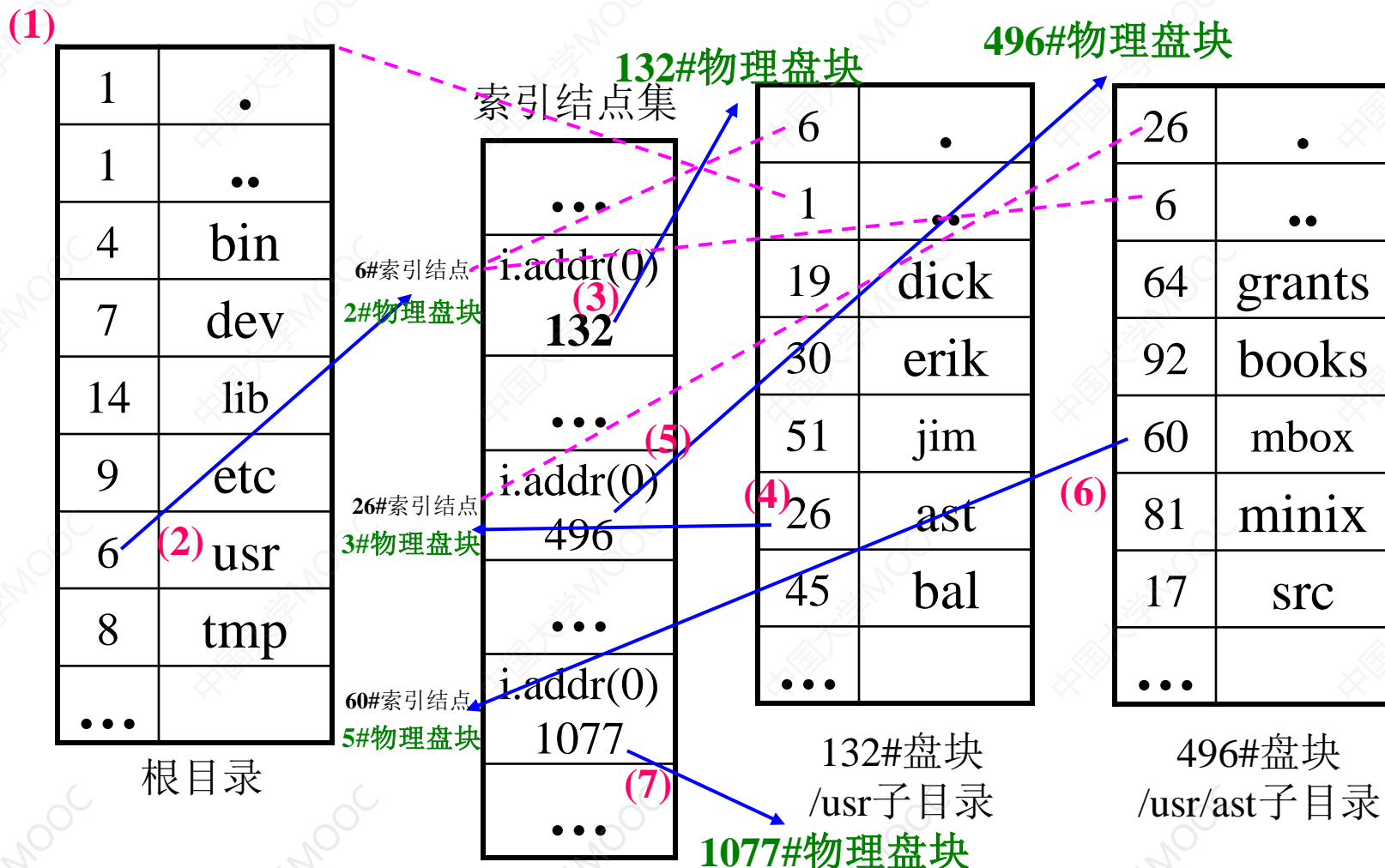
□ 目录查询技术

- 线性检索法（顺序检索法）和散列方法

UNIX文件卷组织结构

- ❑ 系统索引
 - 假设盘块大小为1KB，索引结点占64字节，则i#索引结点应位于的物理盘块号为：
- ❑ 超级块
 - 文件系统管理块及磁盘索引结点)
- ❑ 磁盘索引结点块 2# ~ K#
 - 存放磁盘索引结点
- ❑ 文件数据块 (K+1)# ~ N#
 - 存放文件数据

目录查询举例说明: /usr/ast/mbox



6.5 目录管理

6.5.1 目录管理基本要求

6.5.2 文件控制块

6.5.3 索引结点

6.5.4 目录结构

6.5.5 目录查询技术

作业题

- **6.11** 文件目录管理的基本要求有哪些？
从索引结点和目录结构等两方面阐述目录管理技术的演化发展，并就树形目录结构中文件的按名存取和目录检索技术举例展开详细说明。

重点问题解析

——文件存取全过程

文件存取操作过程剖析

- 在传统的操作系统中，设流式文件yourfile已经打开，且读写指针offset移至2700字节处，根据要求说明系统完成用户程序读盘请求read(fd, 100)的整个过程
 - 基于符号文件目录与索引结点的目录组织方式
 - 物理文件组织采用混合索引方式
 - A. 直接寻址可寻址10个物理块
 - B. 一级索引可索引128个物理块
 - C. yourfile文件体存放于从200号开始的连续物理块中
 - 文件的逻辑块长等于物理块长，均为512字节
 - 读操作通过单缓冲进行
 - 本次是第一次读操作

用户程序读盘请求的整个过程

- 1、请求者进程从用户空间进入核态
- 2、设备无关性软件执行
- 3、磁盘驱动进程运行
- 4、CPU响应通道发来的完成中断请求
- 5、磁盘驱动进程运行
- 6、请求者进程返回用户空间

1、请求者进程从用户空间进入核态

1.1 用户空间执行read函数，
准备系统调用参数

1.2 通过系统调用进入核态

2、设备无关性软件执行

2.1 调用逻辑文件系统求要读的数据所在的逻辑块号 **【5】**（十进制）

2.2 确定youfile所在的设备、把逻辑块号转换为物理块号 **【205】**（十进制）

2.3 申请分配输入缓冲区

2.4 把读操作的参数（文件名、所在的设备、物理块号、缓冲区地址）通知磁盘驱动进程

2.5 唤醒磁盘驱动进程

2.6 请求者进程阻塞

3、磁盘驱动进程运行

3.0 检查确认输入缓冲区无所读数据

3.1 根据读操作的参数将一维物理块号转换为三维物理地址（柱面号、磁头号、扇区号）

3.2 组织通道程序

3.3 启动通道与设备

3.4 磁盘驱动进程自行阻塞（如果设备请求队列无其它请求）

4、CPU响应通道发来的中断请求

4.1 正在执行其它进程的CPU响应设备完成中断

4.2 通过外中断进入核态

4.3 再次唤醒磁盘驱动进程

4.4 分析中断原因，进行磁盘中断处理

4.5 从中断返回

5、磁盘驱动进程运行

5.1 设备驱动进程把输入缓冲区中的数据分离出来并传送到请求者进程的数据区，即从缓冲区内偏移地址

【 140 】（十进制）读100字节送用户数据区，修改读写指针offset为

【 2800 】（十进制）

5.2 唤醒请求者进程，磁盘驱动进程自行阻塞

6、请求者进程返回用户空间

6.1 释放输入缓冲区

6.2 请求者返回用户态

磁盘调度算法

第四、五、六章 重难点习题讲解

作业题

□ 4.10 某虚拟存储器的用户空间共有32个页面，每页1KB，主存16KB。假定某时刻系统为用户的第0、1、2、3页分别分配的物理块号为5、10、4、7，试将虚拟地址0A5C和093C变换为物理地址。

□ 4.11 某请求分页系统采用局部置换算法，但
法，但
4、
的物
程中所及
得结果。

置换
2、1、
作业
访问过
并比较所

为此，页号为2，对应物理块号4。
所以，物理地址为
 $(0001\ 00\ 01)_2(3C)_{16} = (113C)_{16}$

$(093C)_{16} = (0000\ 10\ 01)_2(3C)_{16}$

作业题

□ 6.5 什么是索引？并就分配方式展开

□ 6.6[必做] 假如硬盘容量为10GB，空间？当硬盘容量为10GB时，需占用多少存储空间？

□ 6.7[必做] 假如盘块的大小为4KB，每个盘块号占4个字节，在两级索引分配时，允许的最大文件是多少？

盘块数 = $10\text{GB}/4\text{KB} = 10\text{M}$ ，故最大盘块号为10M-1，需要至少24个二进制位来表示盘块号。

每个盘块中可包含的盘块号数 = $4\text{KB}/4 = 1\text{K}$ ，故两级索引分配最多可包含盘块号数为 $1\text{K} \times 1\text{K} = 1\text{M}$ ，所以，两级索引支持的最大文件大小为 $1\text{M} \times 4\text{KB} = 4\text{GB}$

作业题

□ 6.9

成组链接法所管理外存空间为4MB~ (520MB-10MB-1B),即4MB~(510MB-1B),共506MB。对应盘块号范围为4MB/1KB~[(510MB-1B)/1KB-1],即4K~(510K-1),也就是4096~522239。按每组100 盘块, 即 4096~4195, 4196~4295, 4296~4395, ..., 522096~522195, 522196~522239。最末一组拥有44个盘块,但由于需填补0作为栈底,所以实际记作45。

某文件

主要为
盘块分配的具体过程。

□

6.10[必做]

某计算机系统磁盘容量为520MB,盘块大小为1KB。其中前4MB用于存放索引结点等,后10MB用作对换区,采用成组链接法管理外存空间,每组100个盘块。试画出外存尚未使用的成组链接图。

第六章 文件管理

6.1 文件和文件系统

6.2 文件的逻辑结构

6.3 外存分配方式

6.4 文件存储空间管理

6.5 目录管理

6.6 文件共享与保护

6.7 数据一致性控制

6.6 文件共享与保护

6.6.1 文件共享概念与发展

6.6.2 基于索引结点的共享方式

6.6.3 利用符号链实现文件共享

6.6.4 文件系统安全保护

6.6.5 磁盘容错技术

文件共享概念及必要性

□ 概念

- 指系统应允许多个用户（进程）共享同一份文件，从而在系统中只需保存该共享文件的一个副本即可

□ 必要性

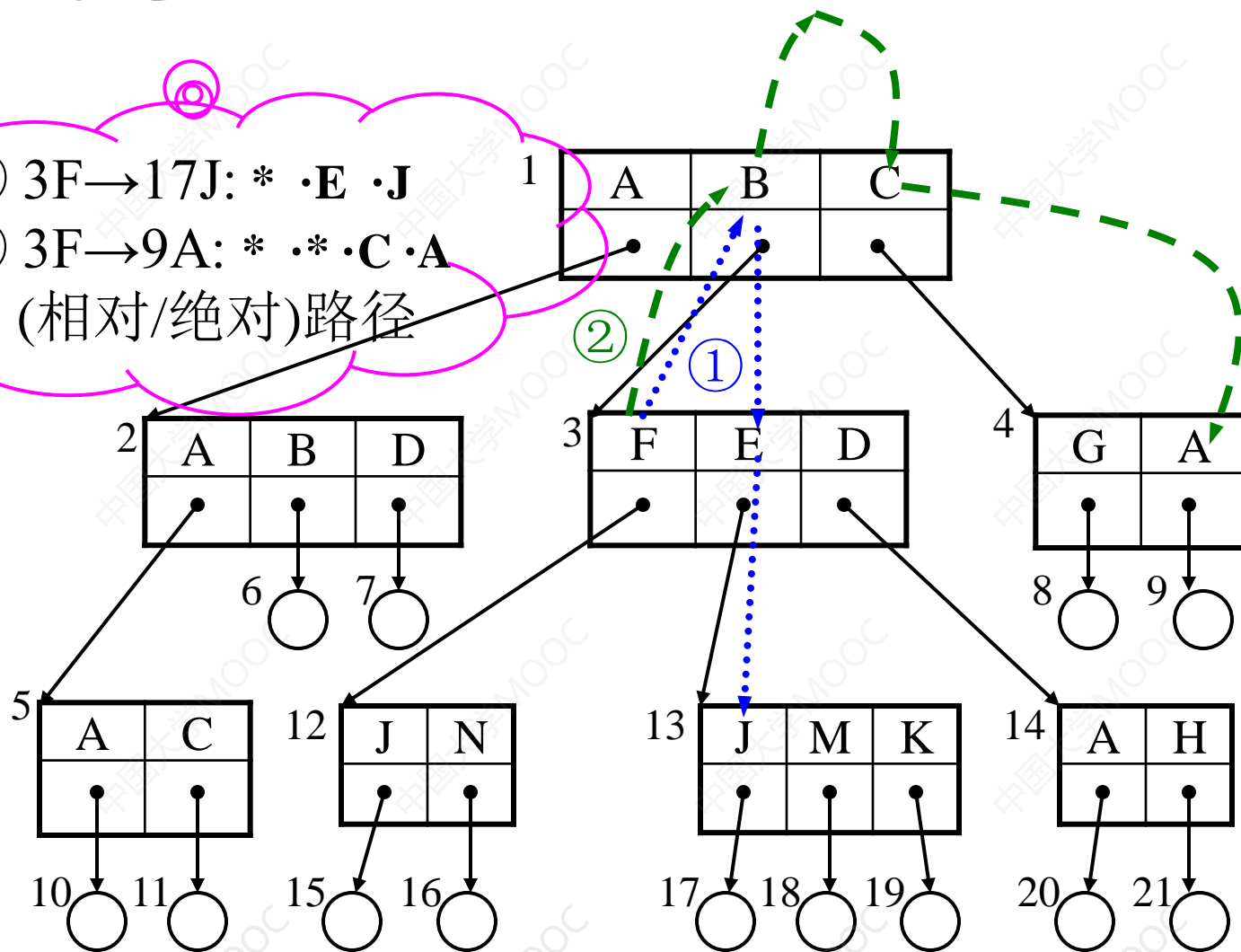
- 如果系统不能提供文件共享功能，就意味着凡是需要该文件的用户，都须各自备有此文件的副本，因此必然会造成存储空间的极大浪费

绕弯路法

① 3F→17J: * ·E ·J

② 3F→9A: * ·* ·C ·A

□ (相对/绝对)路径

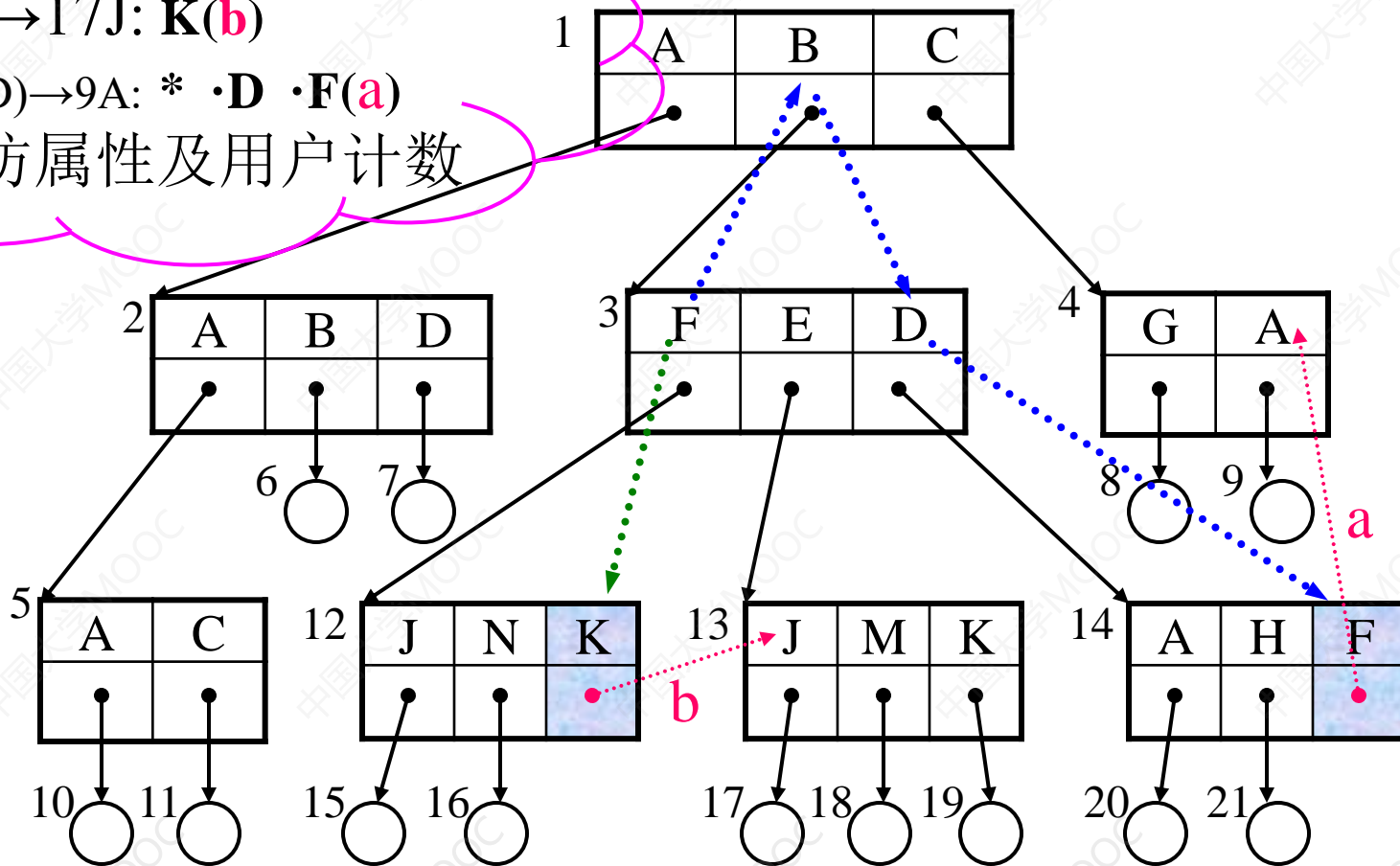


连访法

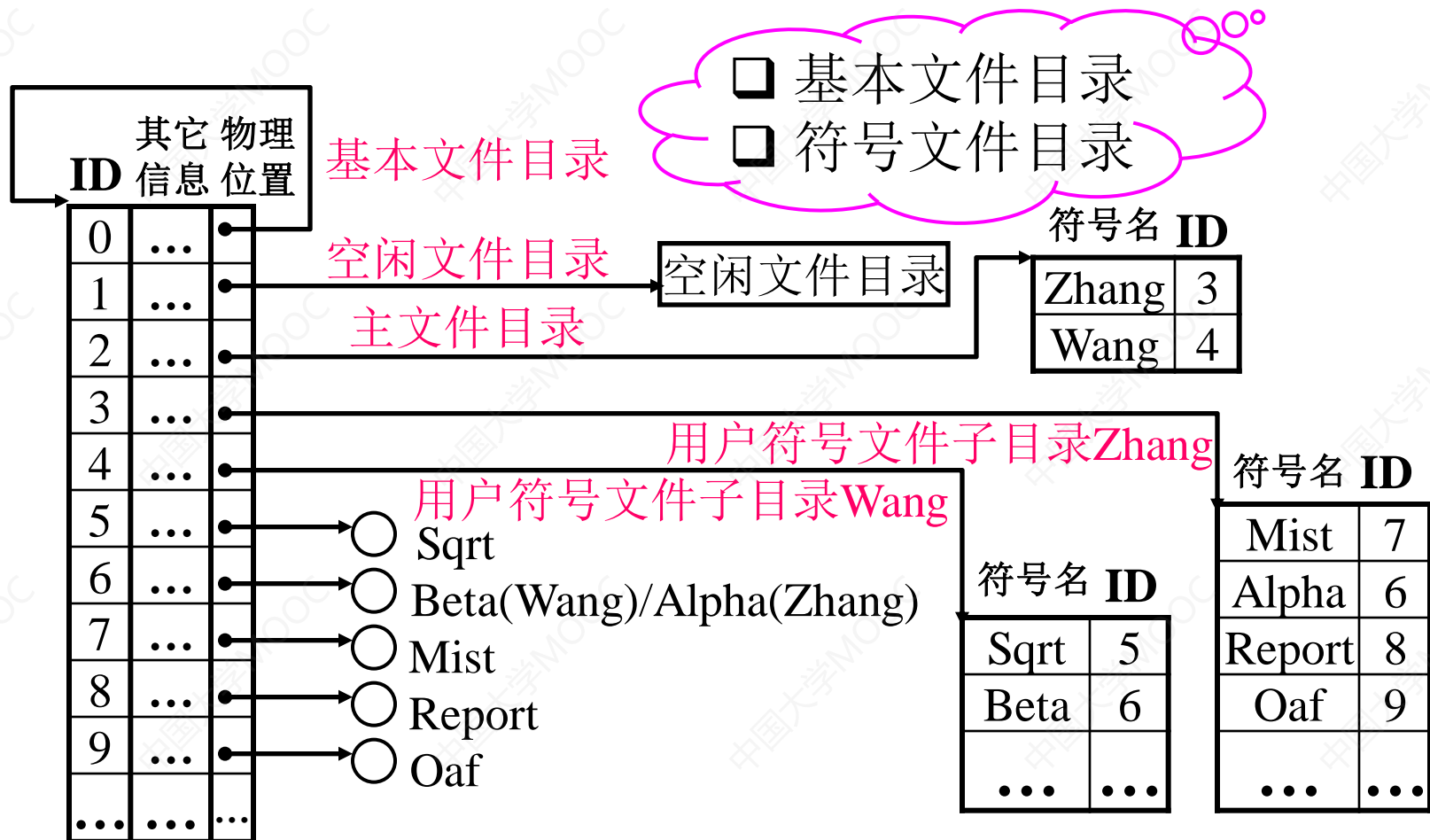
① 3F→17J: **K(b)**

② 3F(D)→9A: * ·**D** ·**F(a)**

□ 连访属性及用户计数



基于基本文件目录实现文件共享



基本文件目录

6.6 文件共享与保护

6.6.1 文件共享概念与发展

6.6.2 基于索引结点的共享方式

6.6.3 利用符号链实现文件共享

6.6.4 文件系统安全保护

6.6.5 磁盘容错技术

基于索引结点的文件共享

□ 基于文件目录建立文件

Linux硬链接及处理方案举例：

➤ 法1：FCP

In **DataFile.txt** **YLJFile**

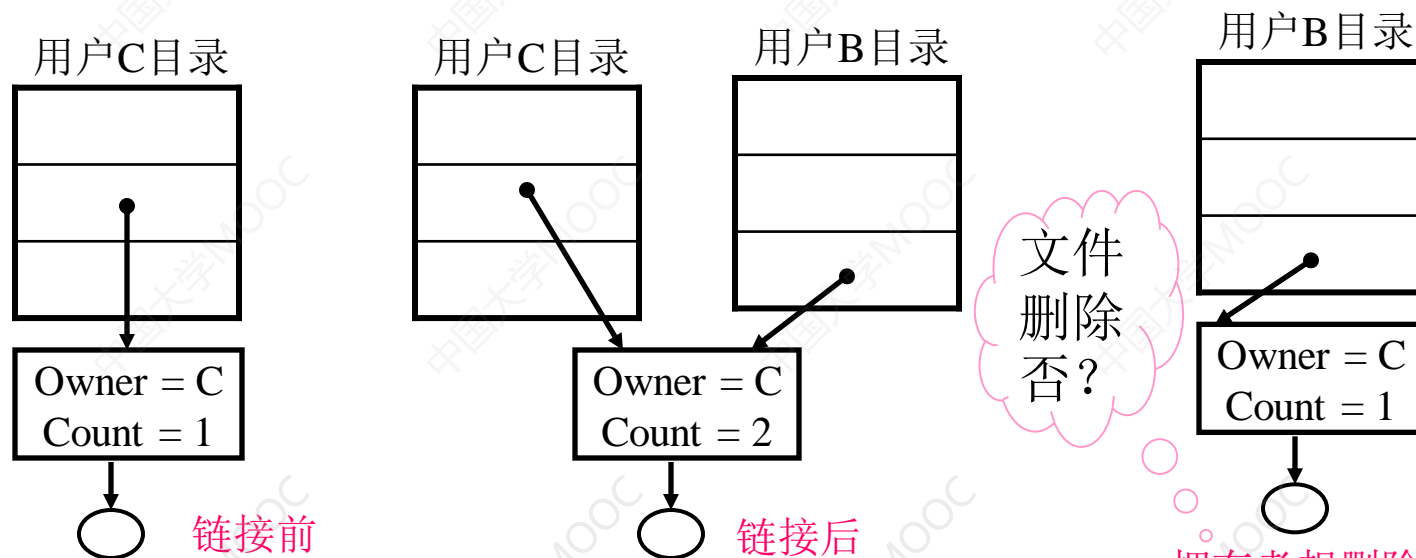
❖ 一旦文件

rm **DataFile.txt**

➤ 法2：符

cat **YLJFile**

❖ 指针悬空问题



6.6 文件共享与保护

6.6.1 文件共享概念与发展

6.6.2 基于索引结点的共享方式

6.6.3 利用符号链实现文件共享

6.6.4 文件系统安全保护

6.6.5 磁盘容错技术

基于符号链的文件共享

□ LINK类型文件

Linux软链接举例：

`ln -s DataFile.txt RLJFile`

`rm DataFile.txt`

`cat RLJFile`

指针，从而避免

了指针悬空

- 可用于链接网络中任何地方计算机中的文件
- 系统开销问题
 - ❖ 文件操作多次读盘与磁盘启动
 - ❖ 符号链索引结点及文件空间开销
 - ❖ 整个文件系统遍历操作的复杂度和工作量加大

6.6 文件共享与保护

6.6.1 文件共享概念与发展

6.6.2 基于索引结点的共享方式

6.6.3 利用符号链实现文件共享

6.6.4 文件系统安全保护

6.6.5 磁盘容错技术

文件安全性影响因素及对策

□ 人为因素

➤ 存取控制机制

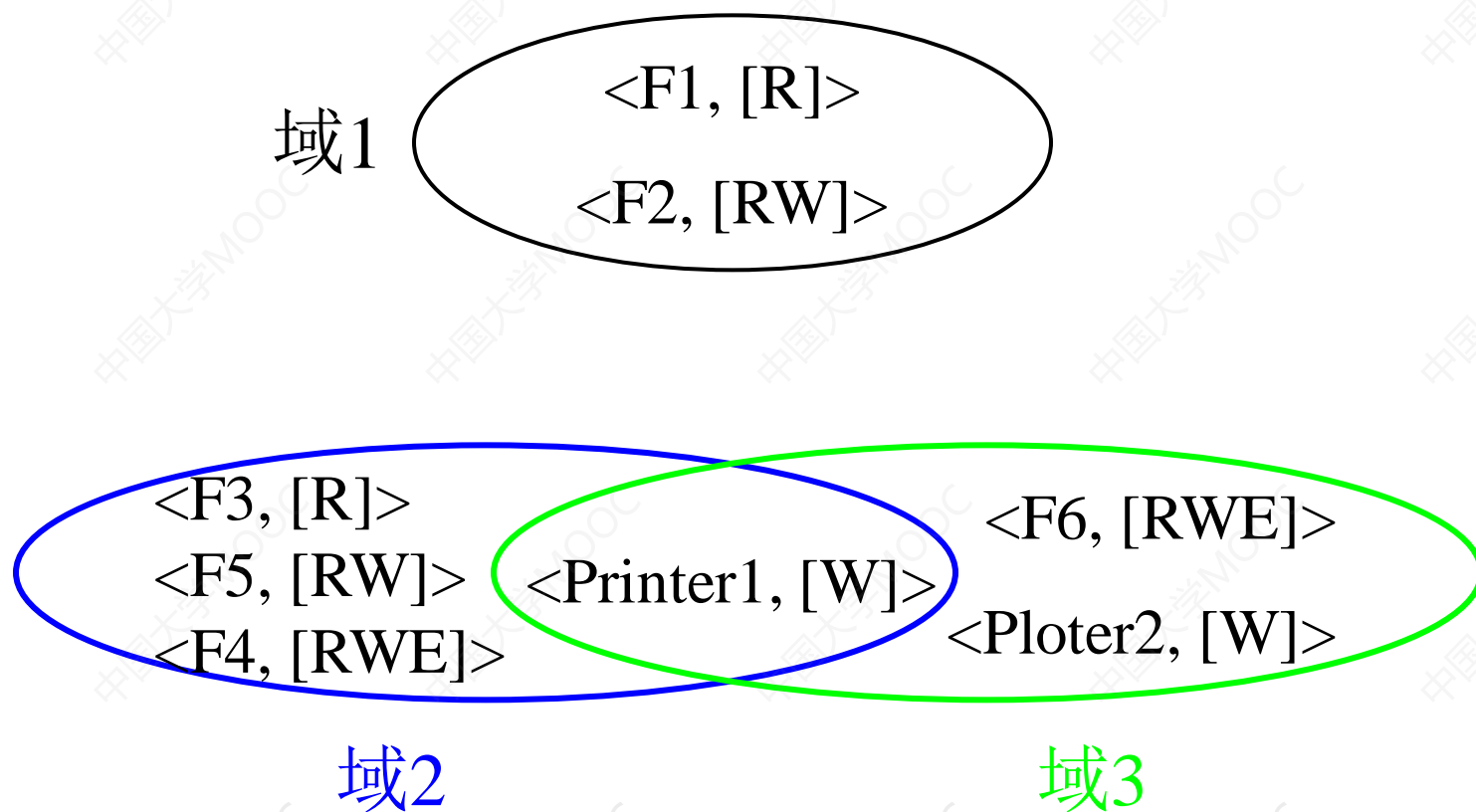
□ 系统因素

➤ 系统容错技术

□ 自然因素

➤ 后备系统

保护域及与进程间联系方式



访问矩阵及切换权引入

域 \ 对象	文件F1	文件F2	文件F3	文件F4	文件F5	文件F6	打印机1	绘图仪2
D1	R	RW						
D2			R	RWE	RW		W	
D3						RWE	W	W



域 \ 对象	文件F1	文件F2	文件F3	文件F4	文件F5	文件F6	打印机1	绘图仪2	D1	D2	D3
D1	R	RW								S	
D2			R	RWE	RW		W				S
D3						RWE	W	W			

拷贝权与所有权

域 \ 对象	文件F1	文件F2	文件F3
D1	E		W*
D2	E	R*	E
D3	E		



域 \ 对象	文件F1	文件F2	文件F3
D1	E		W*
D2	E	R*	E
D3	E	R	W

域 \ 对象	文件F1	文件F2	文件F3
D1	OE		W
D2		R*O	R*OW
D3	E		



域 \ 对象	文件F1	文件F2	文件F3
D1	OE		
D2		R*OW*	R*OW
D3		W	W

控制权（保护域之间）

域 \ 对象	文件 F1	文件 F2	文件 F3	文件 F4	文件 F5	文件 F6	打印机1	绘图仪2	D1	D2	D3
D1	R	RW									
D2			R	RWE	RW		W				C
D3						REW	W	W			

访问矩阵的简化策略

□ 必要性与可行性

- 访问矩阵存储开销及其稀疏性特征

□ 简化对策

- 访问控制表
- 访问权限表
- 兼有式实现机制

索引结点编号

	对象类型	访问权限	对象指针
0	文件	R—	ptrF3
1	文件	RWE	ptrF4
2	文件	RW—	ptrF5
3	打印机	—W—	ptrPrn

文件系统安全管理体系

- ❑ 系统级安全管理
 - 注册与登陆（注册用户表）
- ❑ 用户级安全管理
 - 用户分类及访问权设定
- ❑ 目录级安全管理
 - 目录的读/写/执行许可权
- ❑ 文件级安全管理
 - 文件属性及有效访问权

6.6 文件共享与保护

6.6.1 文件共享概念与发展

6.6.2 基于索引结点的共享方式

6.6.3 利用符号链实现文件共享

6.6.4 文件系统安全保护

6.6.5 磁盘容错技术

磁盘容错技术

□ 基本概念

- 设置冗余部件以提高系统可靠性

□ 低级磁盘容错技术SFT-I

- 双份目录与双份文件分配表
- 热修复重定向、写后读校验

□ 中级磁盘容错技术SFT-II

- 磁盘镜像与磁盘双工

6.6 文件共享与保护

6.6.1 文件共享概念与发展

6.6.2 基于索引结点的共享方式

6.6.3 利用符号链实现文件共享

6.6.4 文件系统安全保护

6.6.5 磁盘容错技术

作业题

- **6.12** 阐述文件共享的概念及必要性，并从基本思想和优缺点评析等角度就绕弯路法、连访法以及基于基本目录、索引结点、符号链的文件共享方式展开讨论。
- **6.13** 影响文件系统安全的主要因素有哪些？如何构建完备的文件系统安全管理体系？并着重就存取控制机制的实现及访问矩阵的优化、磁盘容错技术及所涉关键概念等展开讨论。

第六章 文件管理

6.1 文件和文件系统

6.2 文件的逻辑结构

6.3 外存分配方式

6.4 文件存储空间管理

6.5 目录管理

6.6 文件共享与保护

6.7 数据一致性控制

6.7 数据一致性控制

6.7.1 数据一致性问题及技术

6.7.2 事务概念及恢复算法

6.7.3 检查点及恢复算法改进

6.7.4 并发控制技术

6.7.5 重复数据的一致性问題

数据的一致性问题

- ❑ 当一个数据被分别存储到多个文件中时，便会出现数据的一致性问题
 - 商品进价（流水账、付费帐、分类账、总帐），修改中系统发生故障
- ❑ 硬件支持-稳定存储器
 - 理论上不会出现故障和错误而实际上高度可靠的存储器系统
 - 采用冗余技术，即将一份信息同时驻留在多个独立的非易失性的存储器上

6.7 数据一致性控制

6.7.1 数据一致性问题及技术

6.7.2 事务概念及恢复算法

6.7.3 检查点及恢复算法改进

6.7.4 并发控制技术

6.7.5 重复数据的一致性问题

事务概念及恢复算法

□ 事务的定义

- 用于访问和修改各种数据项的一个程序单位
- 可分散在多个文件中
- “原子性”特征（提交操作/夭折操作）

□ 事务记录表（运行记录）

- 事务名、数据项名、旧值、新值
- $\langle T_i \text{开始} \rangle$ 、 $\langle T_i \text{提交} \rangle$

□ 恢复算法（两个核心过程）

- 已完成事务Redo(T_i) / 夭折事务Undo(T_i)
- ❖ 发生故障时，清理在此之前所发生的事务

6.7 数据一致性控制

6.7.1 数据一致性问题及技术

6.7.2 事务概念及恢复算法

6.7.3 检查点及恢复算法改进

6.7.4 并发控制技术

6.7.5 重复数据的一致性问题的

检查点及恢复算法改进

□ 检查点作用

- 事务记录清理工作经常化（事务记录表和已修改数据输出到稳定存储器），减少恢复开销
- 每当出现一条<检查点>记录时，执行恢复操作

□ 恢复算法改进

- 查找事务记录表，确定在最近检查点以前开始执行的最后事务 T_i
- 针对 T_i 以后开始执行的事务集 T 中的事务 T_k 区别不同情况分别执行恢复操作Redo(T_k) /Undo(T_k)
- ❖ 发生故障或检查点时执行恢复算法

6.7 数据一致性控制

6.7.1 数据一致性问题及技术

6.7.2 事务概念及恢复算法

6.7.3 检查点及恢复算法改进

6.7.4 并发控制技术

6.7.5 重复数据的一致性问题

并发控制

- ❑ 用于实现事务顺序性的技术
- ❑ 利用互斥锁来实现顺序性
 - 共享对象 \Leftrightarrow 互斥锁
 - 简单易行，但效率不高
- ❑ 利用互斥锁和共享锁来实现顺序性
 - 共享文件具有只允许一个事务去写但却允许多个事务同时读的特点
 - 类似于读者与写者问题解决方案

6.7 数据一致性控制

6.7.1 数据一致性问题及技术

6.7.2 事务概念及恢复算法

6.7.3 检查点及恢复算法改进

6.7.4 并发控制技术

6.7.5 重复数据的一致性问题的

重复文件的一致性

- 为保证文件系统的可用性，应为系统中的关键文件设置多个重复拷贝
 - 目录项（文件名） \Leftrightarrow 多个索引结点号
 - 当一个文件拷贝发生修改时，其它的文件拷贝也应做同样的修改，以保证文件中数据的一致性
 - 对策：1、直接根据索引结点找到所有拷贝位置并进行修改；2、为新修改的文件建立同样多份拷贝和替换原有文件拷贝

盘块号一致性的检查

□ 盘块使用情况

- 空闲盘块表（链）与文件分配表
- 对策：构建基于盘块号的两个计数器，分别就空闲盘块号和数据盘块号进行计数，正常情况下，对应每个盘块号的空闲盘块号计数值和数据盘块号计数值应当互补，否则出错

正常情况与盘块丢失举例

计数器组 \ 盘块号	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
空闲盘块计数器组	1	1	0	1	1	0	1	1	1	0	0	1	1	1	0	0
数据盘块计数器组	0	0	1	0	0	1	0	0	0	1	1	0	0	0	1	1

计数器组 \ 盘块号	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
空闲盘块计数器组	1	1	0	1	0	1	1	1	1	0	0	1	1	1	0	0
数据盘块计数器组	0	0	0	0	1	0	0	0	0	1	1	0	0	0	1	1

空闲/数据盘块号重复出现^{举例}

计数器组 \ 盘块号	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
空闲盘块计数器组	1	1	0	1	2	1	1	1	1	0	0	1	1	1	0	0
数据盘块计数器组	0	0	1	0	0	0	0	0	0	1	1	0	0	0	1	1

计数器组 \ 盘块号	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
空闲盘块计数器组	1	1	0	1	1	0	1	1	1	0	0	1	1	1	0	0
数据盘块计数器组	0	0	1	0	0	2	0	0	0	1	1	0	0	0	1	1

链接数一致性的检查

□ Unix类型文件目录

- 目录项 \leftrightarrow 索引结点号
- 共享文件索引结点号在目录中出现的次数应当与其索引结点中的共享用户（进程）计数值相同
- 对策：通过目录遍历构造一张计数器表，为每个文件建立一个表项以记录其索引结点号的计数值

6.7 数据一致性控制

6.7.1 数据一致性问题及技术

6.7.2 事务概念及恢复算法

6.7.3 检查点及恢复算法改进

6.7.4 并发控制技术

6.7.5 重复数据的一致性问題

第六章 文件管理

6.1 文件和文件系统

6.2 文件的逻辑结构

6.3 外存分配方式

6.4 文件存储空间管理

6.5 目录管理

6.6 文件共享与保护

6.7 数据一致性控制

作业题

- **6.14** 谈谈你对数据的一致性问题以及事务、检查点等概念与恢复算法和并发控制技术的认识与理解，并着重就几种典型的重复数据一致性问题及其解决方案进行讨论。
- **6.15** 综合分析可通过哪些途径改善文件系统的性能？
- **6.16** 内存管理和外存管理有哪些异同？试展开深入的分析和说明。

实验课题15

文件系统模拟实现：✎

- I. 模拟实现特定类型的文件系统（如 **FAT** 文件系统）。设计和实现相应文件系统格式的模拟磁盘卷及其 **I/O** 系统的文件存取操作基本功能函数，深入领会和理解文件系统的体系结构、工作原理和设计要领。✎
- II. 撰写实验报告，阐述开发/运行/测试环境、实验步骤、技术难点及解决方案、关键数据结构和算法流程、编译运行测试过程及结果截图、结论与体会等；✎
- III. 要求提交实验报告、源程序及 Makefile 文件（如果有的话）。✎
- IV. 可选提交材料（动画或视频演示）✎

实验课题16

Linux 文件系统设计探析：✎

I. 研读 Linux 内核源码（任意版本均可），探析特定文件系统

（EXT2/EXT3/EXT4/FAT/HPFS/JFS/NTFS/OMFS/PROC/RAMFS/ROMFS/UDF/XFS 任选其一）标准规范及系统实现机制。✎

II. 提交对应源码分析报告、Linux 内核源码及相关参考资料。

III. 可选提交材料（动画或视频演示）✎



**同学们，
再见！**

2021年4月27日星期二