

操作系统第 11 章：file system implementation

尹志涛 22307150055

2024 年 12 月 5 日

1 文件的物理结构

文件在 disk 中如何存储的呢？

1.1 文件存储方式

1.1.1 连续分配

file 在 disk 中也是分块然后连续分配。例如一个块是 1KB，那么一个 4KB 的 file 会被分为 4 个块，然后按连续的 PA 存放。

访问 file 的时候也是顺序访问，也就是，访问第 2 块的 file 时，需要先访问 0 和 1 块。

同时为了维护以及快速查找 file 的位置，存在一个 file 表，表项为：

文件名 | 文件开始的物理块 | 文件所占的块长度（块数）

1.1.2 链接分配

还有一种分配方式是，链接分配方式。链接分配方式分为两种：隐式和显式链接。

file table: 文件名 | 起始块号 | 结束块号，每一块的末尾会有一个指针指向下一个属于 file 的块。



以上是隐式链接分配。

显式链接分配：

file table: 文件名 | 起始块号

除此之外还有一个 FAT (文件分配表) 记录了每一个物理块号对应的下一块的物理块号。



1.1.3 对比连续分配和链接分配

连续分配的优点是便于访问，且存储方便。只要有一块连续的地址空间就可以进行分配；缺点是，如果 disk 中连续的空间不足以存储这个 file，就不会分配空间。

链接分配可以实现空间的充分利用。隐式链接的缺点是，不支持 random access，如果需要访问 file 的某一块，必须从起始块开始逐次访问。

显式链接可以随机访问，根据 FAT 就可以实现随机访问。

1.1.4 索引分配

每一个文件存在一个索引表。

索引表项：逻辑块号 | 物理块号

依然存在一个 filetable (针对 disk): filetable 中表项为：文件名 | 索引块的物理块号。索引块中存放了索引表。因此访问时，先根据 filetable 找到对应文件名的索引块，然后找到索引块里面的索引表。



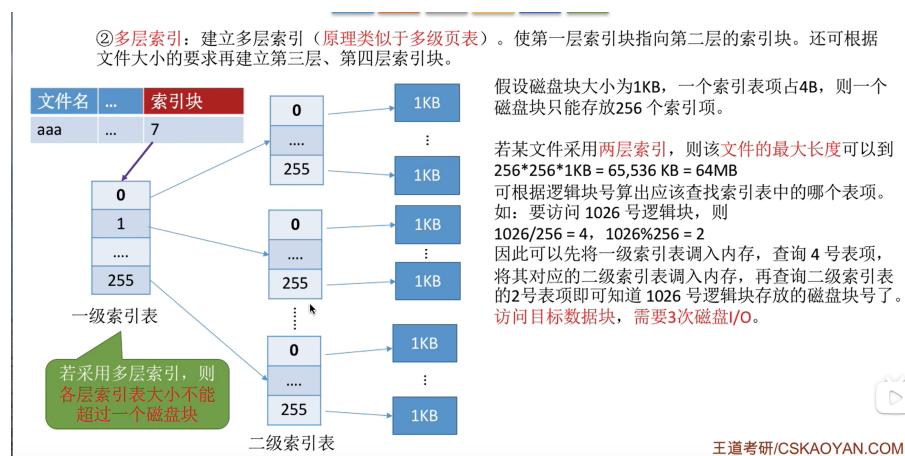
索引分配和链接分配的区别是：索引分配对每个 file 都有一个索引表。链接分配，disk 只需要有一个 FAT。

1.1.5 索引块不够大的情况

如果索引分配时，一个索引块都放不下索引表，用以下方案处理。

1. 链接方案：索引块存在一个指向下一个索引块的指针。
2. 多层索引：类似多级页表，第一层指向第二层，第二层指向第三层，到底层才能找到对应的索引表。

多层次索引加快了访问的速度，没有节省访问的空间。



索引块都是存在 disk 中的，没有调入内存，每一次访问索引表都需要一次 IO 操作。

3. 混合索引：对于一些常用的文件块，在索引表中可以直接写入物理块号，其余的不常用的块，可以根据 visit frequency 考虑用 1 层索引表或者两层索引表...。

2 文件存储空间管理

以上提到的是文件的物理结构，也就是文件占用多个物理块的情况，可能在多个磁盘内存放。讨论的是如何快速高效的访问文件摆放的位置。

现在讨论的是找哪些 disk 来存放 file。也就是记录 free disk block 的算法

2.1 空闲表法

空闲表法

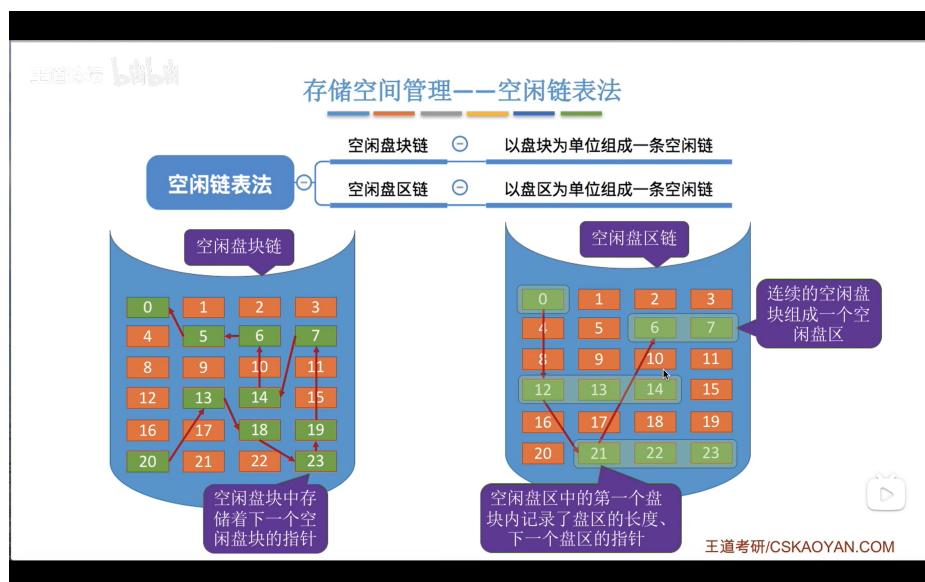
存在一个空闲盘块表，表项为：第一个空闲盘块号 | 连续的空闲盘块数量



2.2 空闲链表法 chained free portion

链表法分为两种，空闲盘块链和空闲盘区链。

空闲盘块链就是指，每一个块之间按照链表连接起来。空闲盘区链就是指，(连续的块合起来叫做一个区)，每一个区按链表连起来



空闲盘区的第一个盘块，需要记录盘区的长度以及下一个盘的指针。

2.3 free-space management 位示图法

在一个磁盘中，有 k 个块，那么你就找 k 个 bit 组成一个很长的二进制数字 a ，然后按 32bit 分为 k 个字，称为字号。 $a[i]$ 指第 i 个字， $a[i][k]$ 指第 i 个字的第 k 个 bit（从左到右，不是指低位高位）。如果 $a[i][k]=1$ ，说明不空闲， $=0$ 就空闲。

2.4 成组链接法

很难用文字描述。这是 UNIX 系统的 disk 分配法。

用的是 super block 超级块。

3 operations in file

3.1 open file table 打开文件表

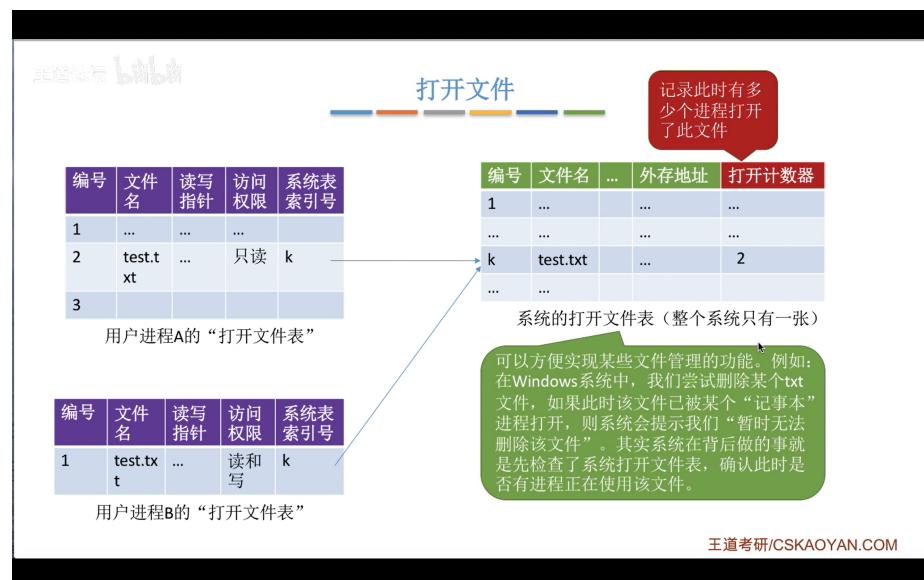
操作系统处理 open 系统调用时，做了两件事

- 根据文件存放路径在对应目录表里查找对应目录项，以确定该用户是否有对应权限。
- 将目录项复制到 **main memory** 的打开文件表中，并将对应表目的编号返回给用户。之后用户使用打开文件表中的编号来指定要操作的文件。

打开文件表：有两种，一种是 OS 系统记录的打开文件表，称之为 open file table，一种是进程自己持有的打开文件表，被称为 channel table。

OS 系统的 Openfile table 记录的是：全编号 | 文件名 | 权限... | 外存地址 | 文件打开的次数。

进程的 openfile table 记录的是：进程内的编号 | 文件名 | 读写指针 | 访问权限 | 系统表编号



3.2 close file 关闭文件

关闭文件也是对 openfile table 操作：

- 删除进程打开文件表中的表项
- 回收分配给该文件的内存资源等等。并且在系统的打开文件表中把次数 count-;

- 如果 count==0，系统打开文件表中删除这个文件。并且取消给这个文件分配的内存资源。

3.3 读写权限

因为 write 和 read 前都必须已经有 open 这个系统调用。

因此进程打开文件表里面已经有了这个文件。因此 write 和 read 只需要注明 1. 进程内编号 2. 要读/写多少 byte 3. 外存位置和内存位置。

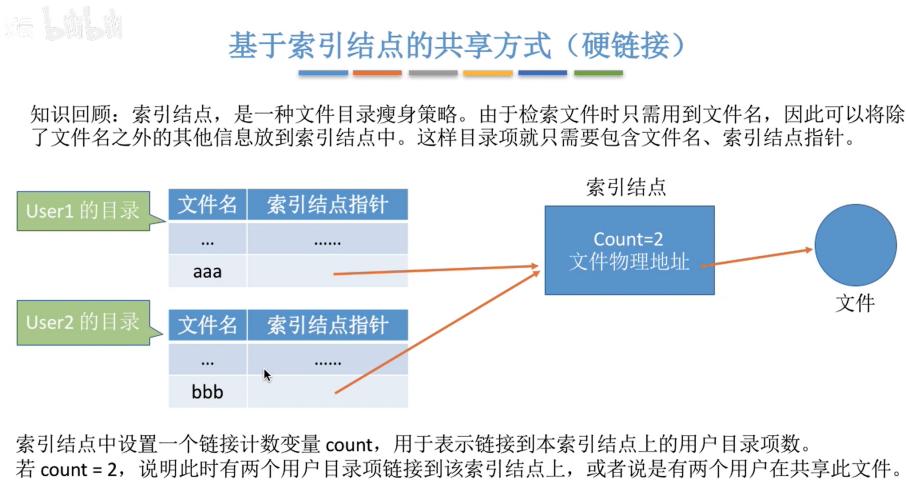
如果是读的话，标注内存位置以方便数据读到内存里面。如果是写的话，标注外存位置，把数据写到外存里。

4 文件共享 file sharing

4.1 基于索引节点的 file sharing-硬链接 hard linking

索引节点：存储了文件除文件名外的其他信息，如读写权限，文件大小，链接计数器 count 等等。这样目录项就可以只包含两个：文件名 | 索引节点指针

其中 count 指，链接到这个索引节点的用户目录项数。count 的意思可以说是：有多少个用户都在用这个文件。



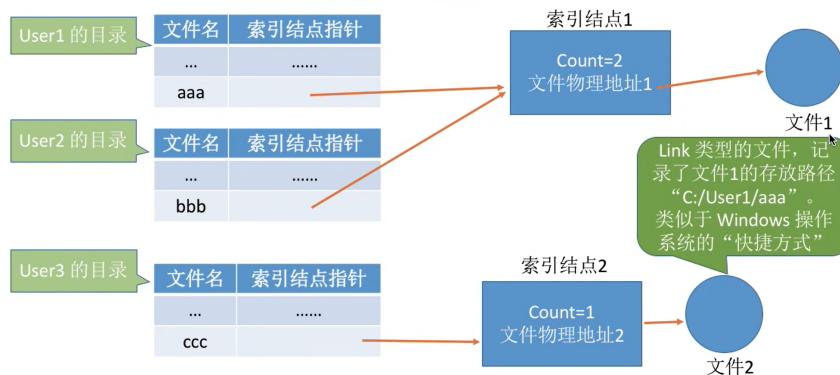
如图，aaa 和 bbb 其实是同一个文件，只是在用户不同的情况下名字不一样而已。其文件信息是一样的。

4.2 基于符号链的 file sharing: 软链接

当建立软链接时，在 user 的目录下先创建一个文件，然后这个文件里面放的是需要 sharing 的 file 的物理地址。和间接访问差不多。

这时 user 的目录项是：文件名 | 索引节点，索引节点指向自己创建的那个文件的信息。然后那个文件里面再存 sharing file 的 PA。

基于符号链的共享方式（软链接）



当 User3 访问 “ccc” 时，操作系统判断文件 “ccc” 属于 Link 类型文件，于是会根据其中记录的路径层层查找目录，最终找到 User1 的目录表中的 “aaa” 表项，于是就找到了文件1的索引结点。

5 文件保护 protection

5.1 口令保护

为文件设置一个口令，访问文件必须提供口令。口令一般放在 FCB 或者索引结点 (i-node) 中。

这个方法很快捷，但是不安全。因为口令是放在 os 中的。

5.2 加密保护

这个加密有很多种方法，这里的加密是指对文件做加密，例如，假如 file 中是 0101000101001101100

你可以设定一个 11111 作为密码，然后密码和 file 全部做“异或”或者“或”才得到正确的 file。

5.3 访问控制

在每个文件的 FCB 中，增加一个访问控制表 ACL(Access-Control-list)，该表中记录了各个用户可以对该文件做哪些操作。

某文件的访问控制列表

用户	读	写	执行	添加	删除	列表 清单
father	1	1	1	1	1	1
mother	1	0	1	0	0	1
son	0	0	0	0	0	0

或者你可以按 groups 设定功能，group 的定义是：把 user 分成一组一组，每一组都是一个 group。

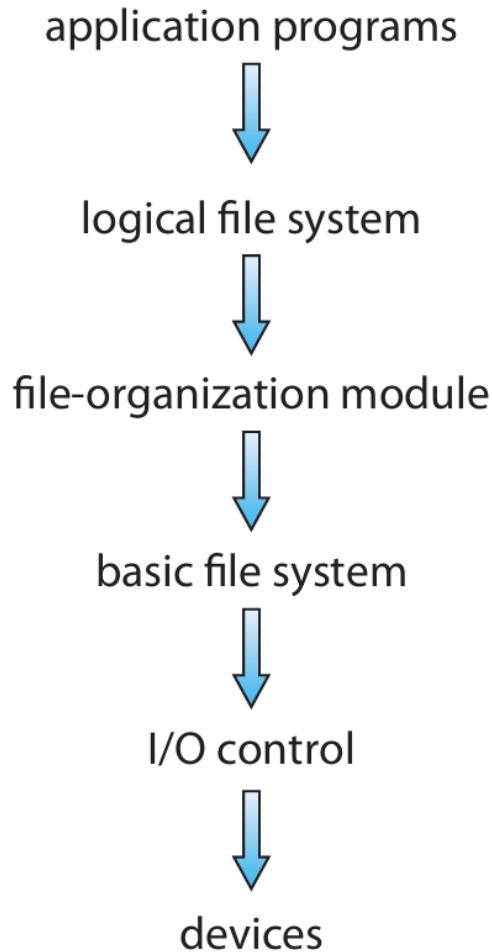
精简的访问列表：以“组”为单位，标记各“组”用户可以对文件执行哪些操作。
如：分为系统管理员、文件主、文件主的伙伴、其他用户几个分组。
当某用户想要访问文件时，系统会检查该用户所属的分组是否有相应的访问权限。

系统需要管理分组的信息

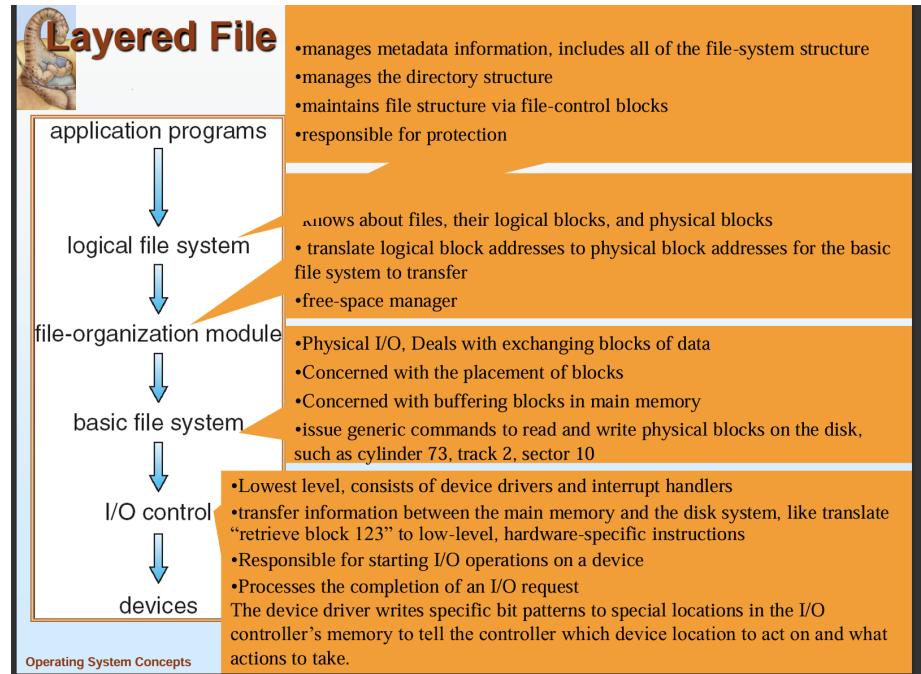
	完全控制	执行	修改	读取	写入
系统管理员	1	1	1	1	1
文件主	0	1	1	1	1
文件主的伙伴	0	1	0	1	0
其他用户	0	0	0	0	0

精简的访问控制列表

6 file system layout 文件系统的层次结构



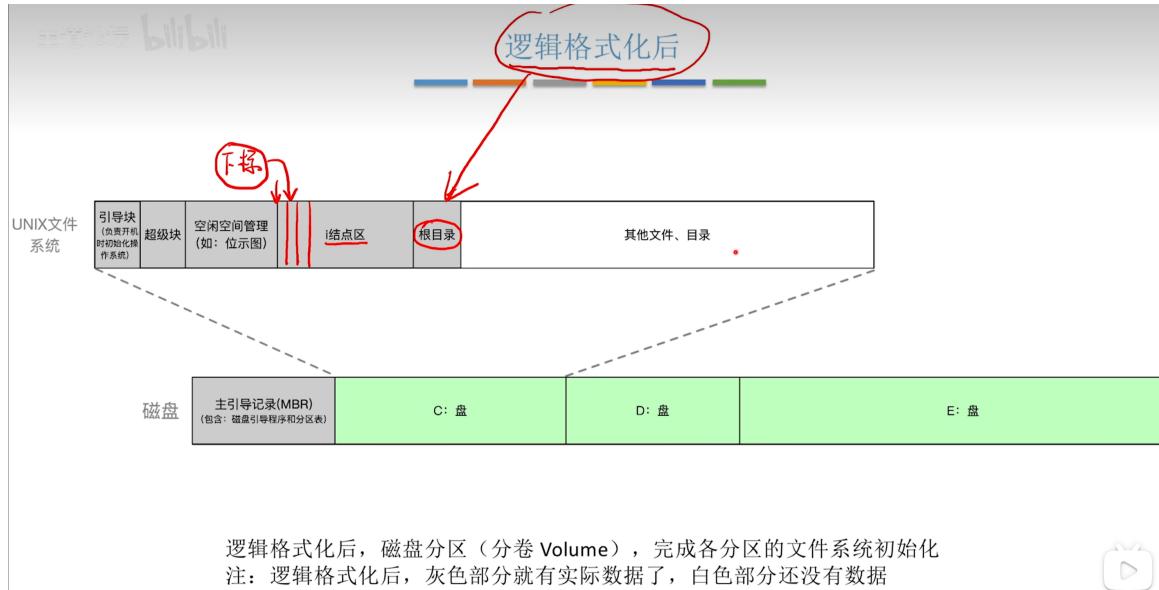
对应功能是：



7 UNIX file management

当磁盘出厂时，没有做任何修改。磁盘首先会对存储空间做物理格式化，也就是把空间分成一个个扇区，并且检测坏扇区，用备用扇区代替坏扇区。

然后根据 user 的操作，会出现逻辑格式化：比如你的电脑有 C 盘，D 盘，这就是逻辑格式化。C 盘这种，叫做分卷 (Volume)。



引导块：Boot block

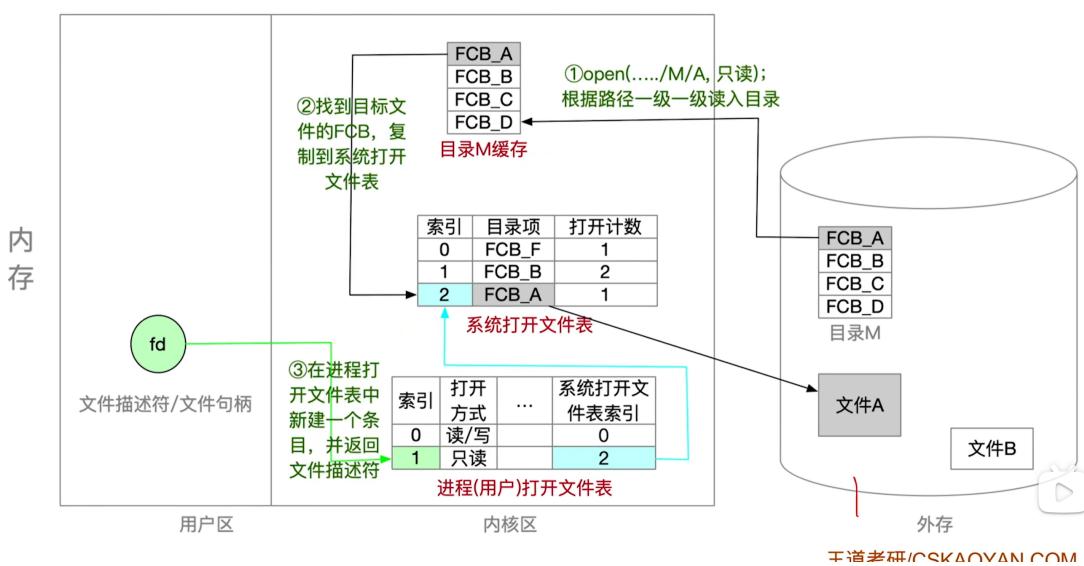
超级块：super block

位式图：free space management

i 节点区：i-node (索引结点)

根目录：chapter10 mentioned

open系统调用打开文件的背后过程



王道考研/CSKAONVAN.COM