



为什么需要文件系统？

- 所有的计算机应用程序都要存储信息，检索信息
- 三个基本要求：
 - 能够存储大量的信息
 - 长期保存信息
 - 可以共享信息
- ❖ 目的：灵活方便、安全可靠地存取数据

文件系统的引入

- ❖ 早期解决方法
 - 纸带和卡片
- ❖ 磁盘存储器和磁带存储器
 - 程序和数据才开始真正被计算机管理
- ❖ 解决方法
 - 把信息以文件的形式存储在磁盘或其他外部介质上
 - 通过操作系统来管理，包括：
 - 文件的结构，命名，存取，使用，保护和实现方法

文件系统的功能

- ❖ 统一管理文件的存储空间，实施存储空间的分配与回收
- ❖ 实现文件的按名存取
 - 名字空间 → 存储空间
- ❖ 实现文件信息的共享，并提供文件的保护和保密措施
- ❖ 向用户提供一个方便使用的接口
 - 提供对文件系统操作命令
 - 提供对文件的操作命令
 - 信息存取、加工等

从磁盘数据块到文件的抽象

- ❖ 文件是磁盘数据块的抽象
 - 文件名是逻辑标识
 - 物理结构和逻辑结构
 - 文件控制块
- ❖ 目录
 - 文件组织、检索、共享

文件系统

- ❖ 目的与要求
 - 了解文件结构，访问方式，存储结构
 - 掌握文件管理用的文件控制块和文件目录结构
 - 了解文件存储器分区和空间管理，文件系统调用处理及使用，系统的组成和各部分功能
- ❖ 重点与难点
 - 文件的逻辑结构与物理结构
 - 文件目录结构
 - 文件使用和控制
- ❖ 作业
 - 思维导图

课后阅读与思考

- ❖ 教材
 - 第7章
- ❖ Modern Operating System (2nd edition)
(现代操作系统)
 - Chapter 4 File System

文件系统

- ❖ 文件结构
 - 文件概念
 - 文件的逻辑结构
 - 文件的物理存储
 - 文件控制块
- ❖ 文件目录结构
- ❖ 文件存储器空间布局与管理
- ❖ 文件访问接口
- ❖ 文件保护
- ❖ 文件系统的基本模型

文件结构

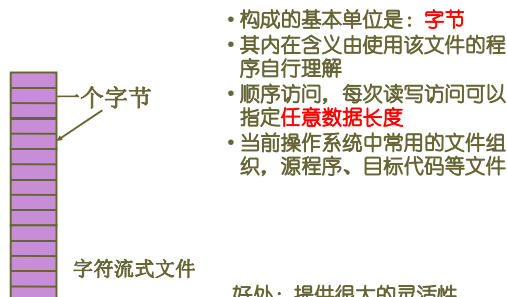
- ❖ 文件概念
 - 一组带标识的，在逻辑上有完整意义的信息项的序列
 - 信息项：构成文件内容的基本单位
 - 长度：单个字节，或多个字节
 - 文件内容的意义：由文件的建立者和使用者解释
- ❖ 文件名
 - 文件的标识符号
- ❖ 文件包括两部分
 - 文件体：文件本身的信息
 - 文件说明：文件存储和管理信息
 - 文件名、文件内部标识、文件存储地址、访问权限、访问时间等

文件逻辑结构

- ❖ 文件的逻辑结构
 - 从用户角度看文件的组织形式
 - 经过抽象
 - 屏蔽信息在物理介质上组织存放的具体方式
- ❖ 设计逻辑结构原则
 - 易于操作
 - 查找快捷
 - 修改方便
 - 空间紧凑

操作系统感知的文件逻辑结构

❖ 流式文件

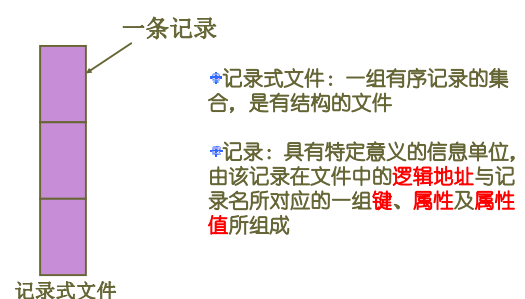


- 构成的基本单位是：字节
- 其内在含义由使用该文件的程序自行理解
- 顺序访问，每次读写访问可以指定任意数据长度
- 当前操作系统中常用的文件组织，源程序、目标代码等文件

好处：提供很大的灵活性

操作系统感知的文件逻辑结构

❖ 记录式文件



❖ 记录式文件：一组有序记录的集合，是有结构的文件

❖ 记录：具有特定意义的信息单位，由该记录在文件中的逻辑地址与记录名所对应的一组键、属性及属性值所组成

文件访问方式

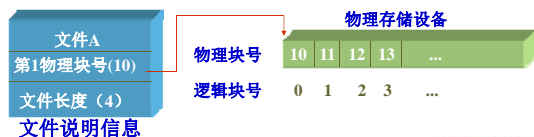
- ❖ 顺序访问
 - 程序**依次**访问文件的数据，操作系统自动记录文件访问的当前位置
- ❖ 直接访问
 - **随机**
 - 程序读/写时直接给出要访问数据的**逻辑位置**（如第几个字节或第几个记录）及长度，由OS将逻辑位置转换成**物理位置**并访问之

文件的物理存储

- ❖ 从**系统的角度**来看文件，研究如何在物理介质上存放文件
- ❖ **物理记录与逻辑记录**的关系
 - 物理块划分
- ❖ 文件的物理组织方法
 - 顺序结构
 - 链接结构
 - 索引结构

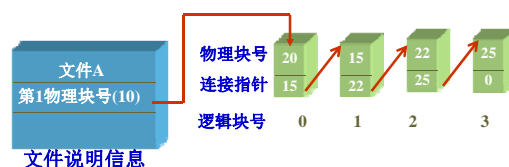
顺序结构

- ❖ 文件的信息存放在若干**连续的物理块**中
- ❖ 优点：简单
 - 支持顺序存取和随机存取
 - 所需的磁盘寻道次数和寻道时间最少
- ❖ 缺点：
 - 文件不能动态增长
 - 不利于文件插入和删除
 - 存储碎片问题——存储压缩技术



链接结构

- ❖ 用**非连续的物理块**存放文件信息
- ❖ 通过**指针**链接成串联队列

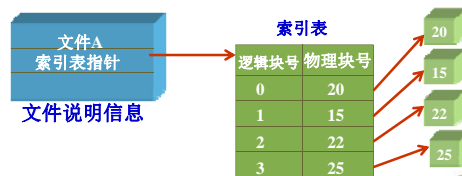


链接结构

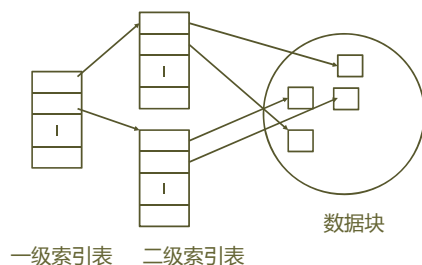
- 优点：
 - 提高了磁盘空间利用率, 不存在碎片问题
 - 有利于文件插入和删除
 - 有利于**文件动态扩充**
- 缺点：
 - 存取速度慢, 不适于随机存取
 - 可靠性问题, 如指针出错
 - 更多的寻道次数和寻道时间
 - 链接指针占用一定的空间

索引结构

- ❖ 一个文件的信息存放在若干不连续物理块中，系统为每个文件建立一个专用数据结构—**索引表**，并将这些**块的块号**存放在一个索引表中
- ❖ 一个索引表就是**磁盘块地址数组**，其中第*i*个条目指向文件的第*i*块



二级索引结构



多级索引示意图



文件控制块

❖ 文件控制块 (FCB)

- 操作系统为管理文件而设置的**数据结构**，存放了为管理文件所需的所有信息
- 文件控制块是**文件存在的标志**
- 文件控制块的内容：
 - 文件名，文件号，用户名，文件地址，文件长度，文件类型，文件属性，共享计数，文件的建立日期，保存期限，最后修改日期，最后访问日期，口令，文件逻辑结构，文件物理结构等

文件管理

❖ 文件结构

❖ 文件目录结构

- 一级目录结构
- 二级目录结构
- 树形目录结构
- 无环图目录结构

❖ 文件存储器空间布局与管理

❖ 文件访问接口

❖ 文件保护

❖ 文件系统的基本模型

文件目录与目录结构

❖ 对文件的**按名存取**

- 为了有效地利用存储空间及迅速准确地完成**文件名到文件物理块的转换**

❖ 引入**文件目录**

- 文件的**名字**及其存放的**辅存**空间地址信息的**目录表**

文件目录结构

❖ 目录项

- 代表一个文件，是FCB或者包含指向FCB的指针

❖ 文件目录

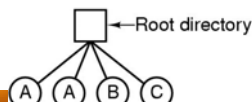
- 目录项的有序集合

❖ 目录文件

- 为了实现对文件目录的管理，通常将文件目录以文件的形式保存在外存，这个文件就叫**目录文件**

一级目录结构

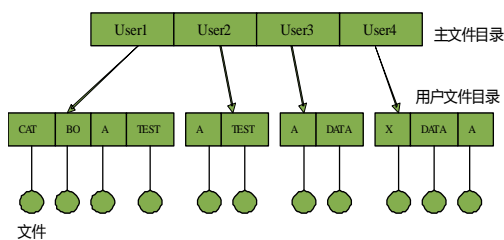
- ❖ 目录的组织结构，设计目标是**检索效率**
 - 整个目录组织是一个**线性结构**，系统中的所有文件都建立在一张目录表中
 - 主要用于单用户操作系统
 - 文件系统通过该目录表提供的信息，对文件进行创建、搜索、读写和删除等操作
 - 文件系统可实现对**文件空间的管理和按名存取**
- ❖ 特点：
 - 结构简单；
 - 文件多时，目录检索时间长；
 - 有**命名冲突**：如重名（多个文件有相同的文件名）或别名（一个文件有多个不同的文件名）



二级目录结构

- ❖ 在根目录下，每个**用户对应一个目录**（第二级目录）
 - 在用户目录下是该用户的文件，而不再有下级目录
 - 适用于多用户系统，各用户可有自己的专用目录
 - 第一级：主文件目录，给出用户名和用户子目录所在的物理位置
 - 第二级：用户文件目录，给出该用户所有文件的FCB
- ❖ 解决文件**重名问题**，实现用户间的**文件共享**，降低查找时间
- ❖ 缺点：增加系统开销

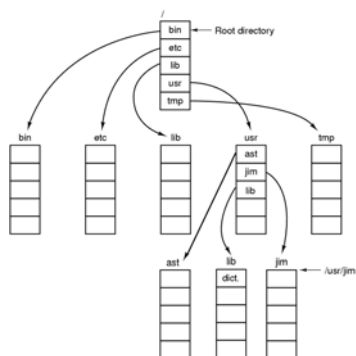
文件目录——二级目录



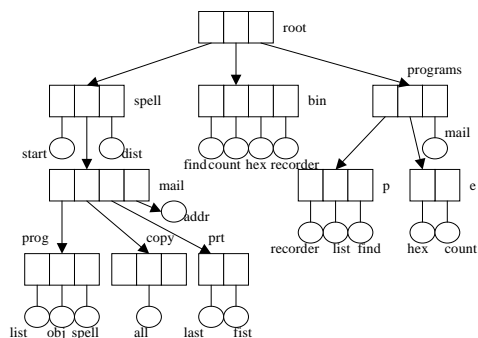
树型目录结构

- ❖ tree-like
 - 在文件数目较多时，便于系统和用户将文件分散管理
 - 适用于较大的文件系统管理
 - 目录级别太多时，会增加路径检索时间
- ❖ 目录树
 - 中间结点是目录，叶子结点是目录或文件**
 - 目录的上下级关系**
 - 当前目录(current directory, working directory)、父目录(parent directory)、子目录(subdirectory)、根目录(root directory)等；
- ❖ 路径(path)
 - 由**根目录或当前目录开始依次经由的各级目录名**，加上最终的目录名或文件名形成的字符串
 - 绝对路径
 - 相对路径

树型目录结构



树型目录结构



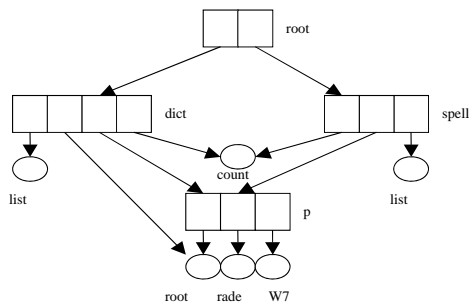
树型目录结构

- ❖ 树型目录特点
 - 层次清楚
 - 不同性质、不同用户的文件可以构成不同子树，便于管理
 - 不同用户的文件可以被赋予不同的存取权限，有利于文件的保护
 - 解决文件重名问题
 - 从根开始到文件名为止，不会由文件重名而引发混乱
 - 查找速度快
 - 可为每类文件建立一个子目录，由于对多级目录的查找每次只查找目录的一个子集，所以搜索速度快于一级和二级目录
- ❖ 目前大多数操作系统：UNIX, Linux, Windows等多采用多级目录结构

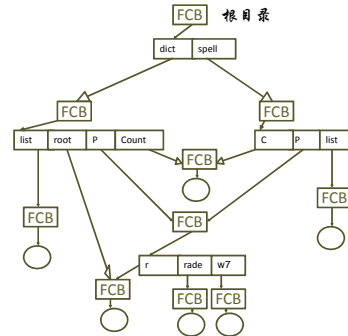
无环图目录结构

- ❖ 当一个文件副本可以同时分到两个不同目录（类别）时，即同一个文件有二条路径名，或多条路径名时出现无环图目录结构
- ❖ 特点
 - 方便文件共享，分类
 - 两个或多个FCB的一致性难保证。
 - 如删除文件时，当文件修改而引起FCB内容变化时

无环图目录结构示意图



将FCB独立存放的无环图目录结构



- 无环图目录结构一种变通的实现方法 - 符号链接
 - 建立多个符号链文件，该文件内容为要访问文件的路径名
 - 当访问符号链文件时，读出文件中的路径名，再重新从根查找路径名代表的文件FCB

目录操作

- ❖ 创建目录项
 - 建立文件或目录时调用
 - mkdir
- ❖ 删除目录项
 - 删除文件或目录时调用
 - rm -r
- ❖ 查找目录项
 - 打开文件或目录时调用
 - opendir, closedir, readdir
- ❖ 修改目录项
 - 文件或目录改名
 - rename, move

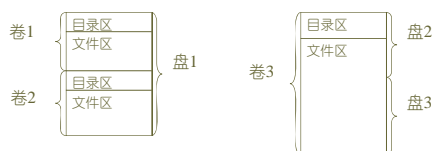
文件管理

- ❖ 文件结构
- ❖ 文件目录结构
- ❖ 文件存储器空间布局与管理
- ❖ 文件访问接口
- ❖ 文件保护
- ❖ 文件系统的基本模型

文件存储器空间布局与管理

❖ 文件存储空间的划分与初始化

- **文件卷**可以是一个物理盘，也可以是一个物理盘的一部分，一个支持超大型文件的文件卷也可以由多个物理盘组成



文件存储空间管理

■ 文件区和目录区自由空间表示

- 将盘空间等分成物理块

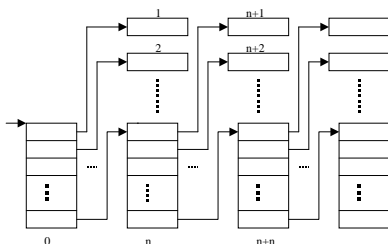
■ **位向量** (bitmap表)

- 每一物理块由bitmap表中的一位表示，1表示占用，0表示空闲

■ 空闲扇区的**成组链接**

- 把所有空闲块链成一个链
- 对空间的申请和释放时以块为单位的，申请时从链首取空闲块，释放时将物理块接入链尾

空闲扇区的成组链接



文件管理

- ❖ 文件结构
- ❖ 文件目录结构
- ❖ 文件存储器空间布局与管理
- ❖ **文件访问接口**
 - 传统文件系统调用的实现
 - 存储映像文件访问
- ❖ 文件保护
- ❖ 文件系统的基本模型

传统文件系统调用的实现

- ❖ 在文件系统中提供对文件的各种操作
- ❖ 方便、灵活地使用文件及文件系统
- ❖ 形式为：**系统调用或命令**
 - 主要功能
 - 创建、打开、读、写、关闭、删除文件
 - 目录
 - 当做一种特殊的文件
 - 目录项
 - 当做文件中的记录

文件的建立与删除

- ❖ CREATE、DELETE系统调用参数包括
 - 文件名（路径名）
 - 设备名（卷名）
 - 其它信息（FCB中要用户提供的信息）

文件的建立与删除

❖ CREATE系统调用大致处理过程

- 检查参数合法性
- 在文件目录结构中的适当位置建立一个**文件控制块FCB**
- 将参数填入FCB
- **分配文件所存放的外存空间**（也可lazy分配即在写数据时分配），建立**索引表**，填入FCB中

文件的建立与删除

❖ DELETE文件系统调用过程

- 检查参数，得到文件名（路径名）
- 按名查找文件目录结构，找到文件的FCB
- 按FCB中的索引表**释放文件所占外存空间**
- 从文件目录结构中**删除FCB**

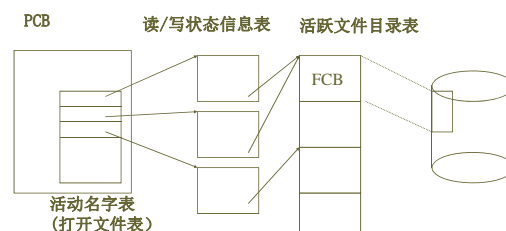
文件的打开与关闭

❖ OPEN系统调用处理过程

- 检查参数，获得文件名（路径名）
- 按名查找文件目录结构，找到文件的FCB
- 将FCB拷贝入内存，存入**内存活跃文件目录表**中
- 建立一张本次**打开的文件读写状态信息表**，将**当前访问指针**指向文件首
- 将文件读写状态信息表的地址存于进程PCB中的资源描述区中“**活动名字表**”中，返回其**索引号**（存入fd）作为本次调用的返回值

文件的打开与关闭

❖ 活跃名字表、文件读写状态信息表、活跃文件目录表的关系



文件的打开与关闭

❖ CLOSE系统调用管理过程

- 检查参数，获得活动名字表索引（fd）
- 按索引在PCB中的“**活跃文件表**”中得到**文件读写状态信息表**的指针
- 如果使用了文件缓冲区，要把改写过的页面写回辅存
- **释放**文件读写状态信息表
- 如果活动文件目录表中FCB不再使用，则释放该FCB所占的空间

文件的读写

❖ 读写参数包括

- fd
- 起始逻辑地址（省缺时即当前访问地址，一般存于读写状态信息表）
- 长度
- 读写信息用户区地址

文件的读写

- Read/Write系统调用大致处理过程
 - 核实参数,按fd获得文件读写状态信息表,获得活跃文件目录表中的FCB
 - 核实操作许可
 - 按FCB中定位信息将要读写的逻辑地址转化成物理地址
 - 如写,则将数据从用户区拷入系统区
 - 如果采用文件直接写技术
 - 将物理地址,内存地址,长度等参数填好,调用外存驱动程序进行I/O操作
 - 如果采用文件延迟写技术
 - 则由操作系统定期地将放在页缓存中的数据写到辅存
 - 如读,则将系统区数据拷入用户区

存储映像文件访问

- ❖ Memory-Mapped File
- ❖ 两个系统调用
 - map: 通过该系统调用将一个文件映射到一段进程地址空间
 - unmap: 将文件与指定进程虚空间段解除映射
- ❖ 文件访问方法
 - Open
 - Map文件到一片虚空间
 - 读写这片虚空间
 - Unmap文件

文件管理

- ❖ 文件结构
- ❖ 文件目录结构
- ❖ 文件存储器空间布局与管理
- ❖ 文件访问接口
- ❖ 文件保护
 - 文件访问保护
 - 文件备份
- ❖ 文件系统的基本模型

文件访问保护

- ❖ 口令
 - 创建文件时提供一个口令,存于FCB中,要对文件访问时,系统核对口令
- ❖ 加密保护
 - 将数据写入文件时,对写入数据加密,读出时进行解密,访问者必须提供密钥

文件访问保护

- ❖ 访问控制
 - 检查用户对文件的访问权限与本次访问是否一致
 - 访问控制矩阵
 - 一维列出全部用户,二维列出所有文件, C_{ij} 值是i用户对j文件的访问权限
 - 简化访问表
 - 在文件中针对文件主,同组用户和其它用户给出访问权限

用户访问权限表

用户类	访问权限
owner	R W E
同组	R E
其它	E

文件保护

❖ 文件可能发生二种破坏情形

- 系统死机等导致文件卷数据破坏
 - 可以通过**备份文件**避免损失
- 不正确的访问方式或非授权访问
 - 可进行**访问控制**

文件备份

❖ 批量备份

- 全量转储
 - 把全部文件定期复制到磁带上
- 增量转储
 - 定期把上次转储以来改过的文件和新文件转储
- 文件系统恢复（死机后恢复）
 - fsck等实用程序将文件目录的数据结构理清
 - 可能丢失文件数据和文件

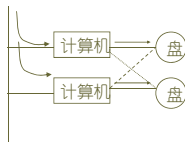
同步备份

■ 镜像盘支持

- 在对磁盘写操作的同时，对称地写其镜像盘

■ 双机动态文件备份

- 有两台机器在文件写操作时完全对称地工作，保证当一台机器出错时，另一台机器还可以接着往下做
- 有了双机动态文件备份，加上双份盘即可防止来自处理机和存储介质两方面损坏对文件系统引起的破坏



文件管理

- ❖ 文件结构
- ❖ 文件目录结构
- ❖ 文件存储器空间布局与管理
- ❖ 文件访问接口
- ❖ 文件保护
- ❖ **文件系统的基本模型**

文件系统基本模型

❖ 文件系统

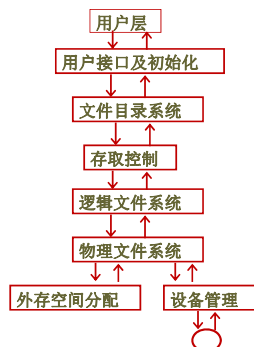
- 负责文件的建立，删除，读/写，修改等以及对文件目录结构和存储介质实施管理
- 一般地，OS的文件系统只提供读/写文件信息的支持而不解释其信息内涵，内涵由各读/写文件的应用程序去解释

文件系统基本模型

❖ 文件系统的优点

- 方便灵活
 - 用户不用管文件在物理存储器上的存放
- 安全可靠
 - 可进行访问限制，可进行文件数据备份与恢复
- 方便共享使用，可按名访问

文件系统基本模型



用户调用接口及初始化模块

- ❖ 对用户进程发出的文件类系统调用进行处理，检查参数合法性
- ❖ 补充省缺参数
- ❖ 把系统调用转化为对相应子程序的调用
- ❖ 负责进程用户空间与OS内核的数据传输交换

文件目录系统

- ❖ 管理与组织文件目录结构
 - 如建立、删除、查找文件目录等
 - 管理“活跃文件目录表”，管理“文件读写状态信息表”和“活动名字表”
 - 打开、关闭及读写时要涉及上述数据结构
- ❖ 上下级的转换
- ❖ 存取控制验证
 - 涉及文件保护的程序属于此模块

逻辑文件系统与文件信息缓冲区

- ❖ 根据文件的逻辑结构，将用户欲读写的逻辑记录转换成文件逻辑块号及块内偏移（offset）
 - 在UNIX流式文件情形下，将文件开始字节数除以块长，取整得逻辑块号，取余得块内偏移

文件系统与文件信息缓冲区

- ❖ 文件缓冲
 - 在内存设立文件数据信息缓冲区，以提高文件访问的速度，减少与外存储器数据交换的次数
 - 系统将文件的某些“相对块号”的数据存放于缓冲区中，系统将这些存有文件数据信息的缓冲区按照Hash队列形式链接起来
 - 当要读写某个文件的某“相对块号”时，按照文件内部号和“相对块号”首先到Hash队列中查找数据是否已经在内存，如果已经在缓冲区中，立即访问之

物理文件系统

- ❖ 功能
 - 把逻辑记录所在的逻辑块号转换成物理块号，利用文件控制块内的索引表信息即可转换
 - 负责对下层模块函数的调用，如外存空间分配及I/O驱动程序
 - 有些操作系统在这里设立磁盘块缓冲
- ❖ 分配模块
 - 负责分配和回收外存空间

设备管理程序

❖ 具有的功能

- 分配设备
- 分配读写缓冲区
- 磁盘调度
- 启动外设
- 处理外设中断
- 释放读写缓冲区
- 释放设备等

小结

❖ 文件结构

- 文件概念，文件的逻辑结构，文件的物理存储
- 文件控制块

❖ 文件目录结构

- 树形目录结构的恩典

❖ 文件存储器空间布局与管理

- 空闲块组织

❖ 文件访问系统调用

- 打开、关闭、读、写

❖ 文件保护

- 访问权限控制

❖ 文件系统的基本模型

- 层次模型，各层的功能

FAT 文件系统磁盘布局

引导扇区	FAT1	FAT2(副本)	根文件夹	其他文件夹及所有文件	剩余扇区
1扇区	由FAT记录项组成，每项代表一个簇	FAT1的副本	32个扇区	簇(从2开始编号)	不足一簇

FAT表和FCB之间的关系



*按照文件逻辑块顺序把所存物理簇链起来，是一种链式索引结构

Q&A