

操作系统课程设计：二级文件系统

基本要求和需要提交的报告形式 V2 同济大学计算机系

一、 实验目的

阅读、裁剪操作系统源代码（文件相关部分），深入理解操作系统文件概念和文件系统实现细节，培养剖析大型软件、设计系统程序的能力。

二、 实验内容

- 1、剖析 Unix V6++源代码，深入理解其文件管理模块、高速缓存管理模块和硬盘驱动模块的设计思路和实现技术。
- 2、裁剪 Unix V6++内核，用以管理二级文件系统。

背景知识

二级文件系统用一个宿主机上的大文件模拟磁盘，文件0#~511#字节是磁盘的0#扇区，512#~1023#字节是磁盘的1#扇区。。。通常这个大文件叫做磁盘镜像文件，后缀是.img。

二级文件系统是管理磁盘镜像文件的应用程序，启动时打开磁盘镜像文件，比如：`fd = open ("SecondaryFS.img", RDWR)`。

随后，应客户端请求，它会使用宿主机文件描述符 `fd` 代理客户端读写磁盘镜像文件中保存的数据，比如：需要读`n#`扇区时，它 `lseek(fd, n*512)`，将磁盘镜像文件的读写指针定位至`n#`扇区起始偏移量，随后 `count = read(fd, &内存变量 , 512)`，从镜像文件中同步读出`n#`扇区内容，存入给定的内存变量。再具体些，Unix V6++的同步读 `bp = bread(n)`，二级文件系统实现其核心IO操作应该使用 2个系统调用：

```
lseek( fd, n*512 ); count = read( fd, bp->b_addr , 512 )
```

其中，`bp`指针指向的缓存块管理分配给`n#`扇区的磁盘高速缓存块，`b_addr`是缓存块的首地址。

实验要求

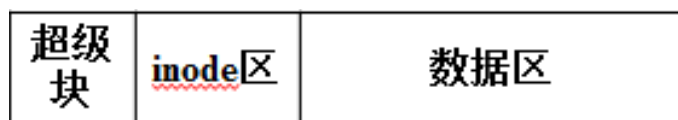
本质上说，内核用以管理（一级）文件系统的所有数据结构和内核子程序均可用来管理二级文件系统。后者管理的所有文件内容及文件系统的元数据存放在镜像文件的确切位置，除 SuperBlock 所在扇区号和根目录的 DiskInode 号之外，一切由文件系统元数据决定。

本设计，基本实验要求如下：

1、 设计满足以下指标的简单二级文件系统 SecondaryFS，宿主操作系统可以是 windows 也可以是 Linux。

1.1 本实验使用磁盘镜像文件 c:\ SecondaryFS.img，存储整个文件卷中的所有信息。一个文件卷实际上就是一张逻辑磁盘，磁盘中存储的信息以块为单位。每块 512 字节。

1.2 c:\ SecondaryFS.img 文件是一个标准UNIX V6++文件卷，格式如下：



1.3 阅读文件系统内核源代码，深刻领会 UNIX V6++ 文件管理系统内核的设计思想。

文件系统（文件卷）的静态结构

(1) inode (DiskInode) 是文件控制块，每个文件实体拥有唯一 inode。二级文件系统保存的文件实体只有两类，普通文件 和 目录文件。

(2) 目录项 (DirectoryEntry) 登记 文件名 和 inode之间的 对应关系。目录项存放在目录文件里。文件卷中，所有目录项构成一棵文件树。

(3) 在磁盘上保存文件数据，需要为其分配磁盘数据块（扇区），管理这些数据块需要使用混合索引树。树根在DiskInode，字段 d_addr。在磁盘上保存文件数据，是数据持久化操作。

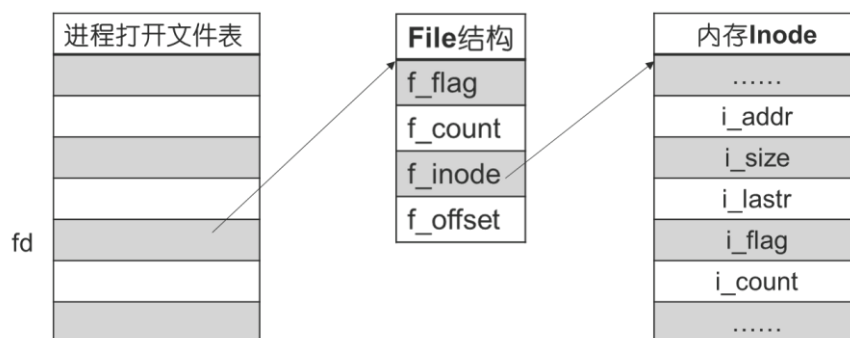
文件系统的使用，就 3 件事儿

(1) 按名查找，这需要访问目录文件、搜索文件树。

(2) 读写普通文件，要用打开文件结构。

(3) 新建、删除 文件 和 目录，这是在更新目录树。

使用文件期间，文件系统需要频繁访问其DiskInode，后者驻留内存直至使用完毕。Unix V6++在内存中设置Inode表，每个元素缓存一个DiskInode，记录该文件的使用状态。此外，还设置有 (1) File 表，每个元素登记单次文件访问会话的状态 (2) 进程的打开文件表，每个元素是一根File结构的指针，文件描述符fd是其数组下标，进程用它引用File结构和内存Inode，指定目标数据在文件中的起始偏移量（文件的读写指针 f_offset），定位其在磁盘上的起始物理地址（用f_offset 和 i_addr 数组，求得数据所在的扇区号 和 它在扇区内的偏移量）。



图：Unix 的打开文件结构

访问普通文件需要使用打开文件结构。open, create系统调用建立打开文件结构；close关闭打开文件结构。前两者会分配 File 和 Inode，并且递增其引用计数器；后者递减计数器，为 0 时释放File 和Inode结构。read, write 和 lseek系统调用使用打开文件表，为进程提供文件数据访问服务。

访问目录文件不需要使用打开文件结构，但会用到内存Inode。目录搜索期间，使用到的每个目录文件，用前将其DiskInode读入内存 Inode，用完关闭释放内存Inode，需要时写回磁盘。使用指遍历目录文件。

1.4 基于上述思想设计实现文件系统内核，实现以下系统调用：

针对普通文件：open, close, lseek, read, write, create

针对目录树：link, unlink, ls, mkdir 新建一个子目录

2、编写 2 个应用程序：

(1) Initialize程序，格式化二级文件系统 c:\SecondaryFS.img。

(2) SecondaryFS程序，接收客户端输入的文件操作命令，访问二级文件系统。

3、为二级文件系统SecondaryFS设计一个简单的测试用用户接口。

(1) 提供文件系统访问命令：每个命令行对应一个系统调用。命令行的第一个字符串是命令 f****，对应 1.4节介绍的一个系统调用 ****。

(2) 命令cd，改变文件系统会话的当前工作目录。文件系统会话指用户使用文件系统的整个过程。

(3) 命令 fin [extername] [intername]，将外部名为 extername 的文件存入二级文件系统，内部文件名为 intername。外部指windows或Linux系统。

(4) 命令 fout [intername] [extername]，将内部文件名为 intername 的文件写入外部名为 extername的文件。

(5) 命令shutdown，安全关闭二级文件系统SecondaryFS。将脏缓存写回镜像文件。

(6) 命令 exit，关闭程序SecondaryFS。相当于断电，文件系统会丢数据。

二级文件系统的使用方法和用户界面设计请参考《范文 用户级文件系统 柴松耀》P28，代码测试。

4、课程设计需要完成的基本功能，此处满分90分。

(1) 格式化一个Unix V6++文件卷。

(2) 启动二级文件系统SecondaryFS。

(3) 在已有目录home下新建子目录：txt, photos 和 reports。

(4) txt 子目录下，fin一个纯文本文件，对它实施 顺序读写，随机读写操作。photos 子目录下，fin一张图片。reports子目录下，fin你的课设报告。要求，稍后可以无损fout。

备注：从 UNIX V6++源代码的 fs 文件夹中，可以找到拥有现代程序设计风格的 UNIX V6 文件系统模块。dev 文件夹中有用于块设备驱动和高速缓存管理的算法。裁剪现成的代码，用以管理存储在镜像文件 c:\ SecondaryFS上的Unix V6++文件卷。

5、需要提交的材料：

(1) 源文件

(2) 课程设计报告

注意：无需提交可执行程序。推荐Linux环境开发，写一个makefile文件，把它和所有的源文件放入一个子目录。进入该子目录后，执行 make 命令生成 SecondaryFS程序。

作业 1：提交程序Initialize。格式化一个Unix V6++文件卷，供二级文件系统使用。截止时间，第三周周六。参考我们做过的格式化文件卷的作业。

作业 2：完成二级文件系统SecondaryFS。不拖拉，推荐第六周周六完成。

注意：前四周不推荐创新。用最基本的

Windows/Linux read/write/lseek 系统调用 完成作业。

第六周，周四下午 1点半 至 3点，我们集中。

申优同学要参加质疑，完成系统设计、实现，写好课设报告后，随时质疑、验收，我们约时间。会开周一~周三上午，我没空。

所有同学，第六周周六后，可以交作业。加油，同学们。

邓蓉 2024/3/3