

Lab6 File System

2023.12.5

目录

- 缓冲层
- 日志层
- Inode层
- 目录层
- 文件描述符层

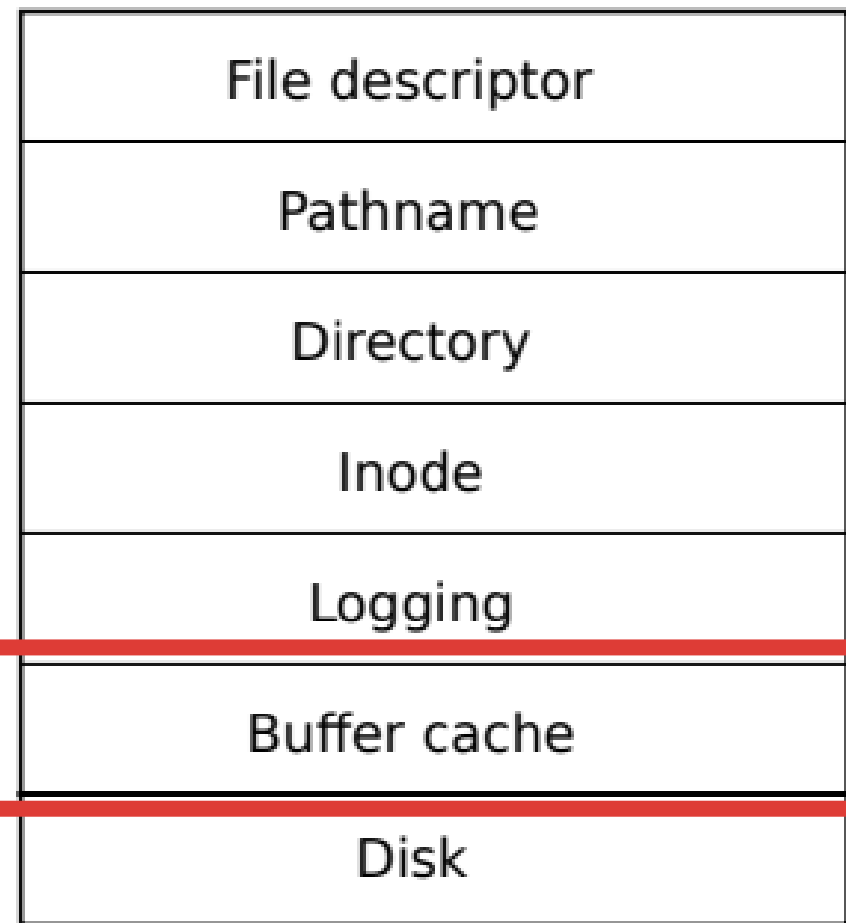
缓冲层

作用：

- 同步对磁盘块的访问，以确保内存中只有一个块的副本，并且一次只有一个内核线程使用该副本
- 缓存流行块，这样它们就不需要从慢速磁盘中重新读取

代码设计：

- 双重链接链表，binit初始化链表。包含两个状态位，B_VALID & B_DIRTY



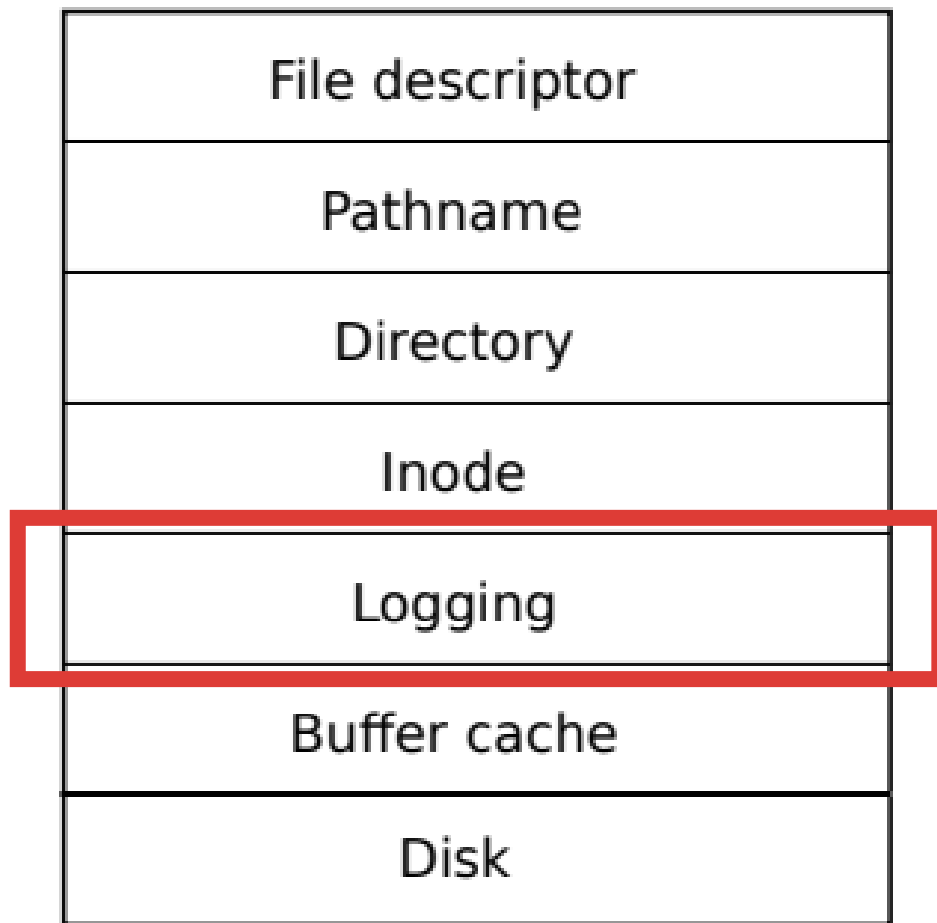
日志层

作用：

- 处理崩溃恢复问题

设计思路：

- XV6系统不直接写入磁盘，会先在磁盘上的日志中放置所有的磁盘写入的描述。记录所有写入后会向磁盘写入提交记录，表明该日志的一共完整的操作。随后系统调用会将所有写入复制到磁盘上。写入完成后系统擦除日志内容。当系统崩溃时，日志被标记为包含完整操作，则恢复代码会将写入复制到磁盘的目标位置，覆盖掉之前的写入内容。日志未被标记为包含完整操作，则崩溃发生在操作提交之前，恢复代码会忽略改日志



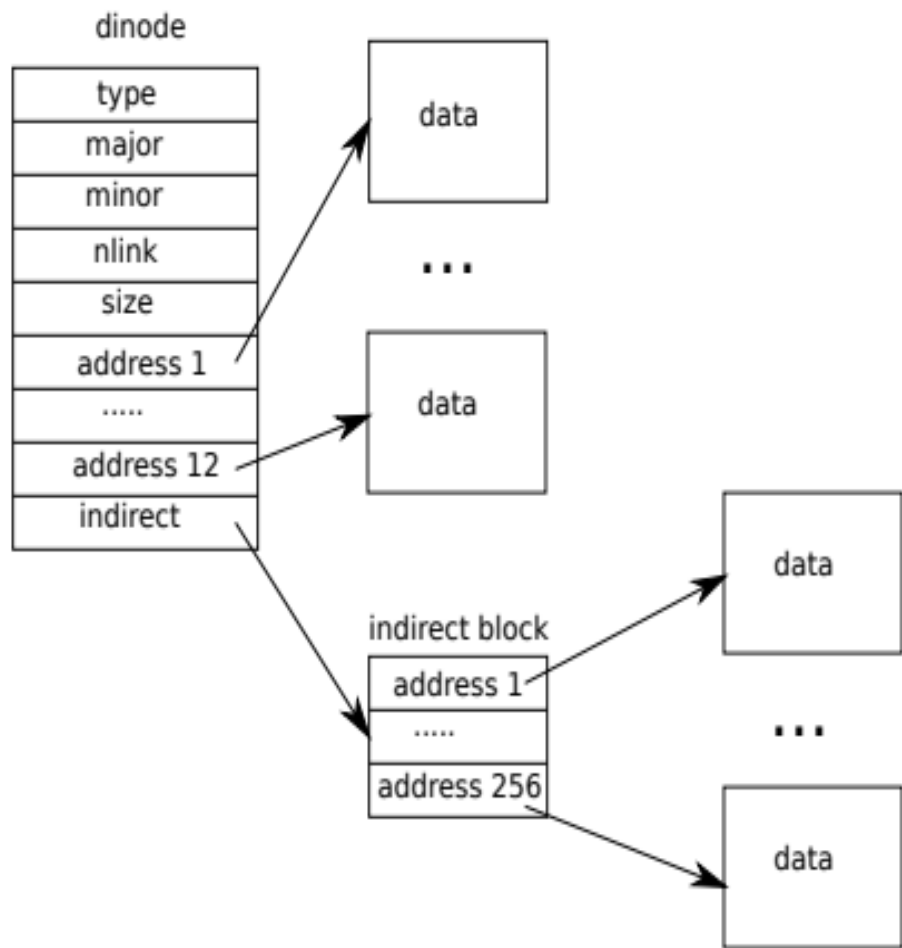
Inode层

作用：

- inode储存文件的元信息，比如文件的创建者、文件的创建日期、文件的大小，文件块的存储位置等等

代码过程：

- 分配inode：XV6调用ialloc
- 获取inode：iget在索引缓存中查询索引节点编号对应的活动条目
- 读写inode：在读取或写入索引节点的元数据或内容前，代码使用ilock锁定索引节点
- 释放inode：iput通过递减引用计数释放索引节点



目录层

作用：

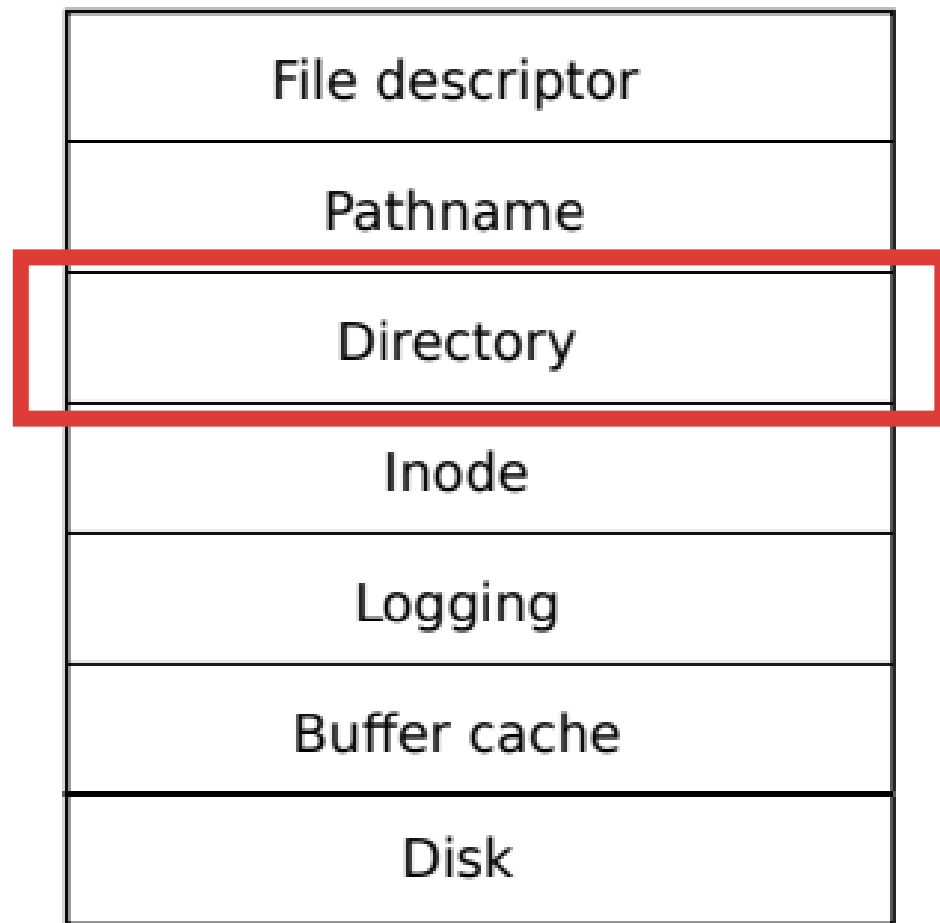
- 实现文件名索引文件

设计思路：

- 将目录作为一种文件，设置一共inode，内容时一系列目录条目，每个条目时一个struct dirent，其包含文件名和索引节点编

过程：

- dirlookup在目录中搜索特定的条目，dirlink将文件名和对应的编号组成的条目写入目录，完成链接



文件描述符层

作用：

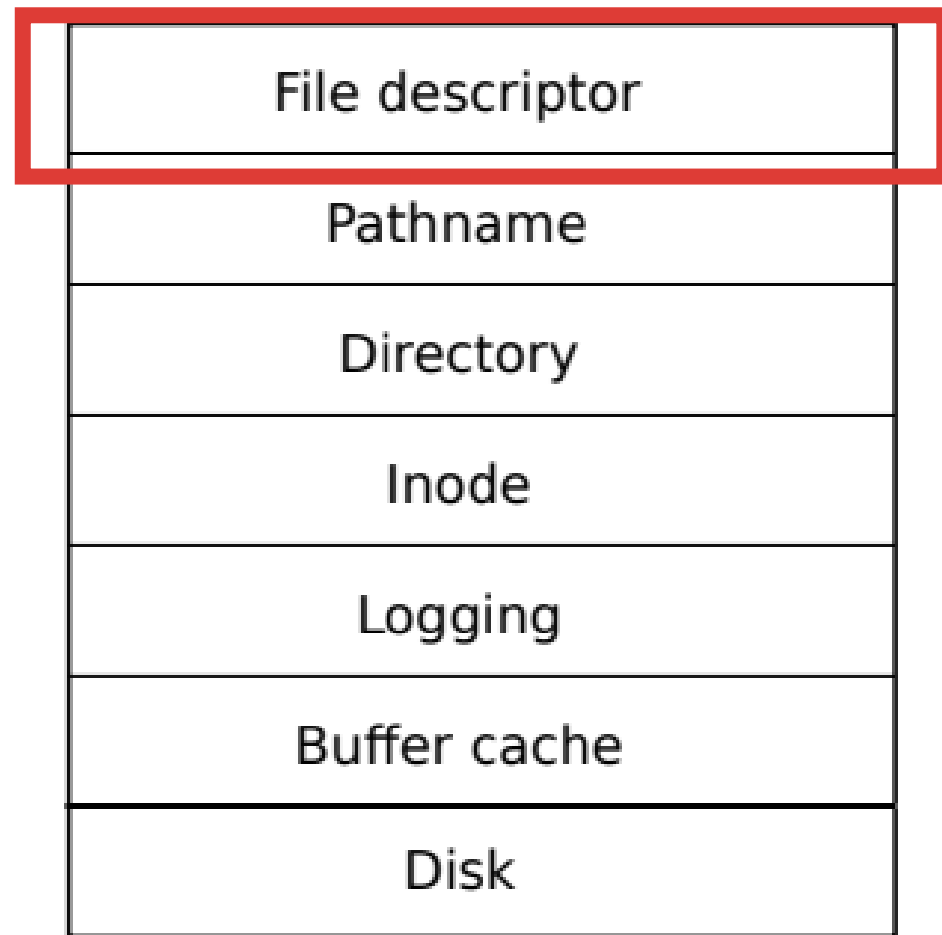
- 将系统中大多数资源表示为文件

设计思路：

- 每个打开的文件（即文件描述符）都由一个struct file表示，该结构是一个inode或管道的包装，加上一个i/o偏移量。Xv6为每个进程提供了自己的打开文件表

过程：

- 系统中所有打开的文件都保存在一个全局文件表ftable中。文件表具有分配文件、创建重复引用、释放引用以及读取和写入数据（fileread和filewrite）的功能



实验准备

启动实验，请切换到 fs 分支

- git fetch
- git checkout fs
- make clean

实验1: Large Files

在此实验中，你需要增加 xv6 文件的最大大小，原先 xv6 的 inode 包含 12 个直接数据块号和 1 个间接数据块号，其中间接数据块包含 256 个块号（BSIZE/4），因此一个 xv6 的文件最多只能含有 268 个数据块的数据

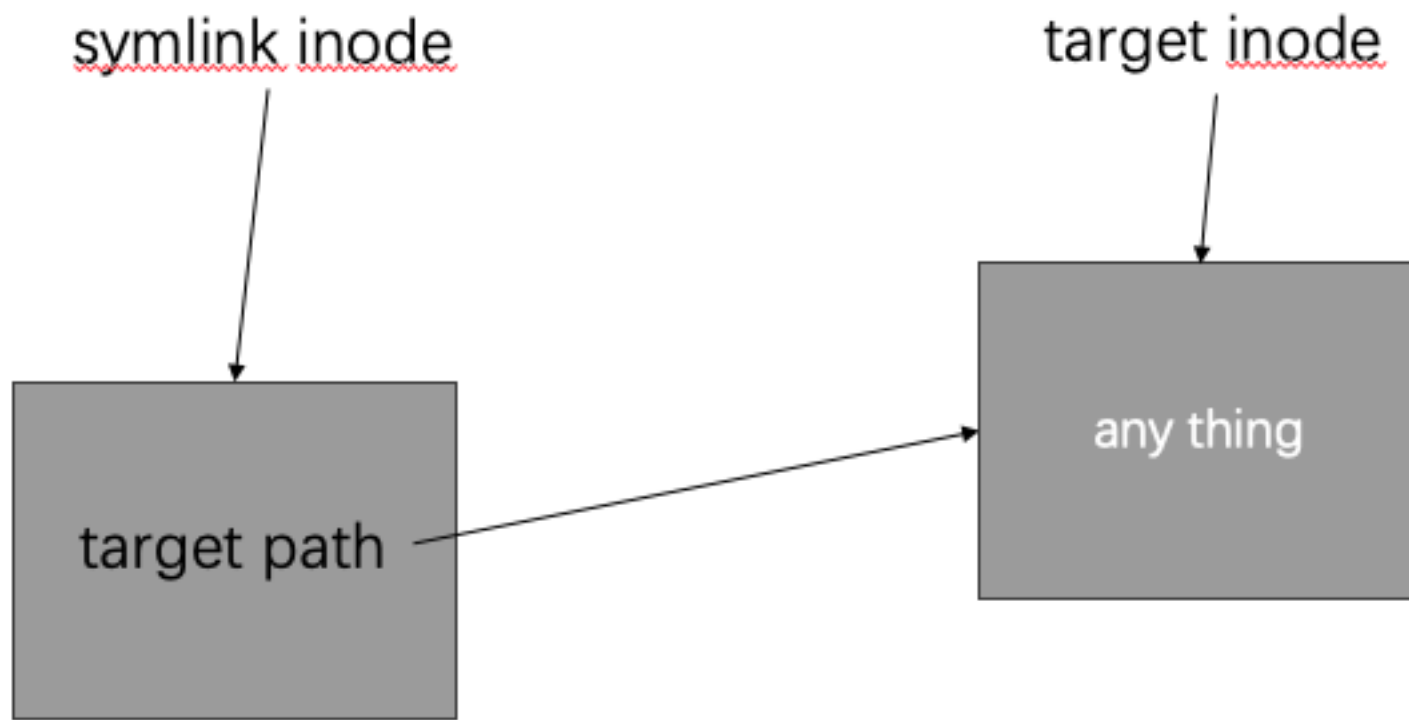
实验目的： 将一个直接数据块号替换成一个两层间接数据块号，两层间接数据块包含 256 个单层间接数据块号地址，每个单层间接数据块包含 256 个数据块地址，结果是一个文件最多可以包含 65803 个块，或 $256*256+256+11$ 个块

实验步骤：

- 修改 fs.h 中 `NDIRECT` 的数量为 11，增加 `NDINDIRECT` 的定义
- 修改 `dinode`、`inode` 的块地址分配大小
- 修改 `bmap` 函数，它会将指定 `inode` 的第 `bn` 个数据块（相对位置）映射到磁盘的数据块号

实验2: Symbolic links

实验内容：实现符号链接 `symlink` 系统调用



实验2: Symbolic links

实验步骤:

- 在kernel/sysfile.c 中实现一个空的 sys_symlink
- 修改 user/usys.pl、user/user.h、kernel/syscall.h、kernel/syscall.c 以添加一个 system call
- 向 kernel/stat.h 中添加新的文件类型 T_SYMLINK
- 向 kernel/fcntl.h 中添加新标志 O_NOFOLLOW
- 实现 symlink(target, path) 系统调用
- 修改 open 系统调用

实验提交

- 提交到邮箱**fduos2023lab36@163.com** :
 - 命名为：**学号-姓名-授课教师-lab6.pdf**，报告内容包含：
 1. 概述题标清练习题编号；
 2. 实验题标清练习题编号，说明实验思路，实验过程（必须包括但不限于实验代码、系统调用流程等）实验效果截图；
 3. 实验过程中遇到的问题及解决方案；
- 截止日期：**2023年12月16日23时**
- 注意：请各位同学独立完成实验，参考代码需注明

参考资料

- **xv6 Lab: File System**
 - <https://pdos.csail.mit.edu/6.828/2023/labs/fs.html>
- **xv6 book: Chapter3**
 - <https://pdos.csail.mit.edu/6.828/2022/xv6/book-riscv-rev3.pdf>