山东大学 软件 学院

操作系统课程设计 课程实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 学号：201600301079 | 姓名： 崔玉峰 | | 班级： 2016级软件4班 |
| 实验题目： **Laboratory 5: Extendable Files：** | | | |
| 实验学时：4 | | 实验日期： 2018/10/30 | |
| 实验目的：  **Laboratory 5: Extendable Files：**  The purpose of this lab session is to help you start to work on extending the Nachos file system, the programming task of Assignment 3. The work required in this laboratory session itself is part of Assignment 3.  The Nachos file system is a simple file system with many restrictions. One of them is that the size of the file is not extendable: once you specify the size of a file upon its creation, the size of the file  is fixed throughout its lifetime. In this laboratory session, you are going to extend the Nachos file system to allow the size of files to be extended. In particular, we want the new Nachos file system to have the following features:  • When a file is created, its initial size can be set to 0.  • The size of a file can be increased if more data are written to the file. | | | |
| 硬件环境：  PC | | | |
| 软件环境：  虚拟机：VMWARE  操作系统：ubuntu 16.04.5 32位 | | | |
| 实验步骤与内容：  **Laboratory 5：**  Laboratory 5 主要实验目的事实现文件的追加功能。主要的修改文件在FileHeader和OpenFile中。   1. **源码分析：** 2. 通过对整个nachos文件系统的分析，在更改文件大小后，肯定要更改文件头FileHeader的信息，因为文件头存储着文件的大小，位置信息，所以要添加修改FileHeader的方法。其次对于OpenFile要增加WriteBack()方法，其主要作用是将修改后的FileHeader写回。 3. 查看fstest.cc中的Append()方法，通过调用OpenFile的Write方法将新增的数据追加到,nachos文件中。 4. 查看OpenFile的Write调用WriteAt方法进行写入。   通过上述两个判断可以看出来，WriteAt方法不能超出文件长度写数据，所以要在调用WriteAt方法之前修改FileHeader的大小，在必要时分配更多的扇区。   1. **具体实现**     **⑴ 在FileHeader中增加Append方法修改文件大小信息。**  FileHeader:: bool Append(int newSize);    传入文件新的大小newSize，根据newSize进行判断更改Fileheader中文件大小信息。  ①如果新的大小newSize小于等于原来大小numBytes ，则不需要修改  ②当newSize所占的扇区数等于原来的扇区数，则不需要更改扇区数，只需将numBytes = newSize;  ③当newSize所占的扇区数大于原来的扇区数，则需要为文件分配新的扇区，所以要加载文件系统的BitMap分配新的扇区，如果扇区不够则return FALSE。否则通过freemap->Find()为他分配新的扇区。最后将freeMap写回完成修改。  **⑵ 在OpenFile中增加WriteBack()方法**  OpenFile写回操作主要是将修改后的FileHeader写回，  所以需要OpenFile增加一个成员变量hdrsectorNum来记录FileHeader的扇区，在构造函数赋值。       1. 更改**OpenFile中的WriteAt()方法**   **需要在写之前调用hdr->Append(position+numBytes)更改FileHeader的信息。**    **其他代码不用改动，就可以正常分配.**   1. **修改fstest.cc**   跟据题目要求将openfile->WriteBack() 的注释去掉。       1. MakeFile文件修改。   将filesys的Makefile文件和Makefile.local文件拷到lab5文件夹中。  Makefile中添加include Makefile.local  Makefile.local添加main.cc\fstest.cc  修改INCPATH 为-I../lab5       1. **实验运行** 2. 修改完毕后make编译 3. 执行 ./nachos -f 创建nachos文件系统 4. 执行./nachos -cp test/small small 将linux下的small文件拷贝到nachos 下。执行./nachos -p small 查看： 5. 执行./nachos -ap test/small small 将linux下的small文件追加nachos 下。执行./nachos -p small 查看：        1. 执行./nachos -cp test/empty empty 将linux下的empty 文件追加nachos 下。执行./nachos -p empty 查看：      1. 执行./nachos -ap test/medium empty 将linux下的medium 文件追加nachos 下的empty文件。执行./nachos -p empty 查看：      1. 执行./nachos -D查看文件情况   **Ubuntu32-2018-11-02-12-51-36**   1. **测试./nachos -hap**   **执行./nachos -nap small empty:** | | | |
| 结论分析与体会：    实验完成了题目要求的所有内容，并测试通过，源码已在报告中附上，对于所有的内容皆有本人独自完成。  通过本次实验学到了nachos文件系统的整体实现，整体目录系统，文件头FileHeader,通过Bitmap管理扇区，目录实现文件名和FileHeader扇区地址的映射，以及OpenFile对于文件的读写操作。  一开始查看nachos源码会比较困难，但是通过老师的讲解会发现，nachos的实现方式与课本上的实现一样单层结构目录，索引分配的文件系统。所以通过实验更能使我了解操作系统的相关知识，巩固课堂所学的知识。  因为对于c++的语法不是很熟悉，尤其对于Makefile的语法不是很熟悉，走了很多弯路，不过最后还是完成了感觉自己学到了很多东西，对操作系统也产生了更多的兴趣 | | | |