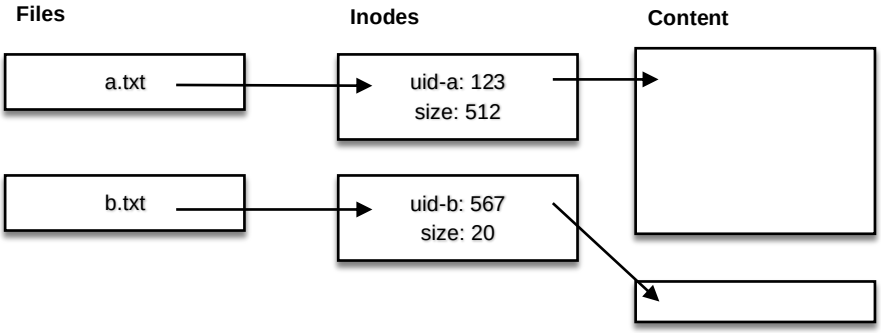
File System Management
Practice Exercises

COMP-310/ECSE-427 Winter 23

Question 1: Hard-links vs Soft-links

A. Consider the schema below, which shows two files, a.txt, and b.txt, their corresponding inodes, and the content of the files. We create a hard link to the file a.txt, called hard-link.txt and a soft link to b.txt, called soft-link.txt. Complete the schema to reflect the state of the system after soft-link.txt and hard-link.txt have been created. What is the content of the hard-link and of the soft-link?

B. We now remove the files a.txt and b.txt. Modify the schema you created in point 2 to reflect the state of the system after a.txt and b.txt have been removed. What is the content of the hard-link and of the soft-link?



Question 2: File System Indexing I

Assume we have a disk with 512-byte disk sectors and 8-byte disk addresses. The file system uses inodes with 10 direct block pointers, 3 indirect block pointer, and 2 double-indirect block pointer, and 1 triple-indirect block pointer. The block size is identical to the sector size of the disk.

A. What is the maximum file size that can be represented by this file system?

B. A user process opens a file of length 1GB, and issues read requests for the following blocks of this file: 50, 137, 51, 138, 139, 4100, 1. What is the total number of disk sector accesses? Explain your answer by drawing a diagram illustrating which disk accesses occur.

Assume that prior to the open, the inode of the file and the blocks belonging to the file are *not* in the cache, but that there is space in the cache to store all of them. You may also assume that there is no file system activity going on, other than the accesses to this particular file.

C. The number of direct, and (single/double/triple) indirect pointers in this file system was chosen to optimize for fast access in the common case. Knowing this, what can you infer about the typical file size that is served by this file system?

Question 3: File System Indexing II

What is the size in bytes of the biggest file that can be represented with an inode that contains PD direct pointers (disk addresses), PI pointers to an indirect block, and PDI pointers to double-indirect blocks? Assume that a disk block is S bytes and P pointers (disk addresses) fit into a disk block.

Question 4: Disk Scheduling

Consider a disk drive which has 500 cylinders (indexed 0 to 499). The disk head is currently at 100. The queue of pending requests, in FIFO order, is:

21, 87, 172, 450, 45, 217, 300, 5, 98, 113.

Out of SCAN (moving up), C-LOOK (moving up, serving requests on the way up), and SSTF, which disk scheduling policy minimizes the total hard disk arm motion?

Question 5: Extending a File

Consider a file currently consisting of 200 blocks. Assume that the file is open and that its inode and the free space bitmaps are already in memory.

Calculate how many disk I/O operations are required for linked, FAT, and indexed allocation strategies, if, for one block, the following conditions hold. Assume indexed allocation with 10 direct pointers, 1 indirect pointer, and 1 double-indirect pointer, where each pointer block can hold 100 pointers. Also assume that the block information to be added is stored in memory.

a) The block is added at the beginning.
b) The block is added in the middle.
c) The block is added at the end.

Question 6: Random File Access

Assume that the file with filename bigfile in the current directory has a size of 1 Gigabyte (2^30 bytes). Assume furthermore that you are given a function randomfilelocation() that returns a random value between 0 and 2^30 – 1. Using Linux primitives, write pseudo-code for a program that does 100 iterations of a loop in which each loop iteration draws a random number R using randomfilelocation(), and printfs to stdout 10 bytes of the file bigfile starting at location R. You

need to be explicit about what Linux calls you are using, and about the parameters to those calls that matter for the correctness of your pseudo-code, but you can omit any parameters that are irrelevant (i.e., permission flags and the like).

Question 7: RAID

A. What is the main advantage that RAID 5 has over RAID 4? Why?

B. What is the minimum number of drives you need to configure a RAID 5 array? Briefly explain your answer.

Question 8: Log-structured File System

In a log-structured file system (LFS) how does the system determine that a block is "old", i.e., logically overwritten by a later write to the same block?

A. Explain what information LFS maintains in memory to make this determination.

B. Explain what information LFS maintains on disk to make this determination.

C. Given your answers to A and B, describe how the determination is made.