Operating Systems Lab 10

210010032(Nithin Sabu), 210010019(Himanshu Kumar)

Problem:

We implement immediate files in the Minix file system, for files up to 32 bytes. Implement only for the file system mounted at /home.

● File creation: you can start by creating the file as an immediate one. When a file grows beyond 32B, then you can make it a regular file.

● File read: if it is an immediate file, you can respond with the inode structure contents. If not, you can follow the default behavior of looking up zones.

● File write: similar to read. You must take care to ensure that if you want to write to the inode structure, then the new file size is still within 32B. When a regular file shrinks to less than 32 bytes, there is no need to come back to immediate mode.

● File delete: deleting immediate files does not require any handling of zones.

Immediate Files:

Inodes include pointers to the disk block where a file's data is kept as well as data pertaining to the metadata of files, including permissions, file size, last access and modification times, and last access time. These pointers either point directly to a disk block or, in a single-indirect manner, to a list of additional pointers to data blocks. Regular files have the drawback of requiring the allocation of an entire disk block, even for relatively short files, which wastes disk space.
Data can be directly stored in the inode rather than on the disk thanks to immediate files. In Minix3, an inode has a length of 64B, of which 40B are utilized to store references to data blocks. These 40B can be utilized to store the file content directly in cases where no data blocks

are used. Therefore, instantaneous files can be used for files up to 40B in size. As a result, shorter files require fewer disk accesses, which shortens access times. We deal with instantaneous files here, up to 32 byte sizes.

Modifications:

Following files have been modified:
/minix/fs/mfs/read.c
/minix/fs/mfs/write.c
/minix/servers/vfs/link.c
/minix/servers/vfs/open.c
/minix/servers/vfs/read.c
/minix/include/minix/const.h
/minix/lib/libc/gen/fslib.c

Output: