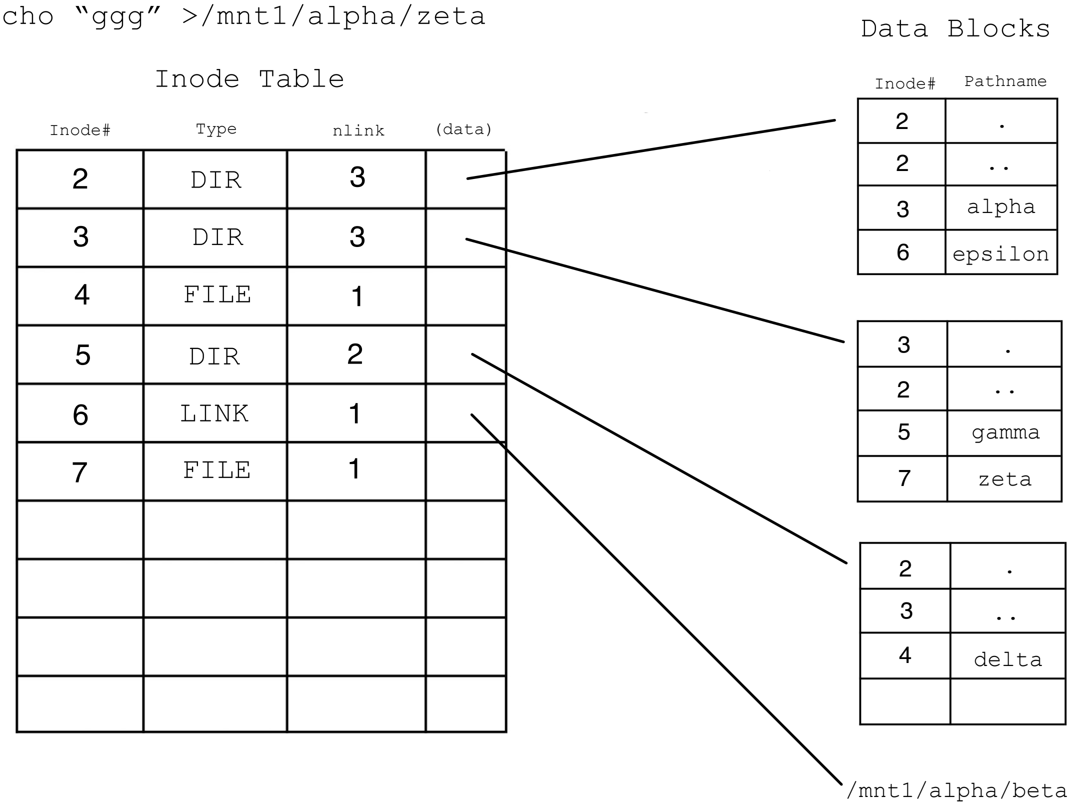
Simon Yoon
ECE357: Computer Operating Systems

1.

```
mkdir /mnt1/alpha/gamma
ln /mnt1/alpha/beta /mnt1/alpha/gamma/delta
ln -s /mnt1/alpha/beta /mnt1/epsilon
rm /mnt1/alpha/beta
echo "ggg" >/mnt1/alpha/zeta
```



Inode Table

Inode#	Type	nlink	(data)
2	DIR	3	
3	DIR	3	
4	FILE	1	
5	DIR	2	
6	LINK	1	
7	FILE	1	

Data Blocks

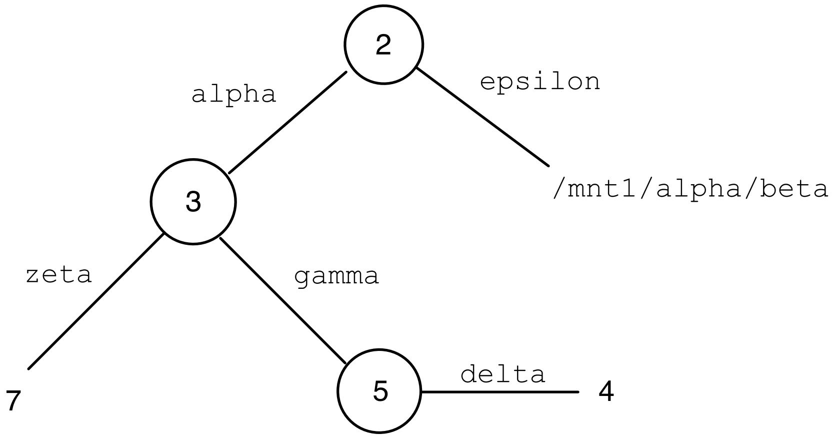
Inode#	Pathname
2	.
2	..
3	alpha
6	epsilon

3	.
2	..
5	gamma
7	zeta

2	.
3	..
4	delta

/mnt1/alpha/beta

User Mode View



2.

A)

The Linux EXT3 filesystem's organization of its data is relevant as an HDD is organized by a CHS coordinate system. The cylinder groups seen in Linux EXT3 and in all modern UNIX filesystems can be advantageous for mechanical hard disks because the moving head must physically move to access sectors on the drive. With the moving head traversing physically, the time it takes to access files stored in 'distant' sectors would be considerable. And with multiple operations on a mechanical hard disk, there would be repeated instances where the inode table, freemap, … etc. need to be accessed. The utility in having these stored in memory close to one another is negligible in an SSD as the displacement of the needle is not necessarily reduced in the SSD's randomly accessed memory.

B)

i)

040755_8 = 100000111101101_2 , so the type is revealed in the first bits of the `mode` field. 04_8 → 0100_2 which is a directory inode, `S_IFDIR`.

ii)

The nlink number refers to the number of pathnames linked to the inode. If `nlink` = 5, then, there are 3 pathnames to inode#6688 as nlink starts at 2. The inode innately has a connection to itself and its parent These 3 pathnames can be attributed to a combination of subdirectories under the inode or hard links to it.

C)

Reasons the practical capacity on a SATA hard disk might be noticeably less is because the user is not watching the videos under the `root` user and therefore, cannot utilize the percentage of the drive accessible only by the `root`, an amount determined by the reserve factor meant for preventing fragmentation. This space in the disk is designated to prevent allocation of memory, placing files directly adjacent to each other. 2^{30} bytes * 4000 ≈ 4TB and with the reserve factor being usually between 5-10% of the disk's capacity, this is a likely case. Another reason might be because the files' attributes such as file size, date created, and a file's author are stored in its metadata, which is not directly contributing to the displayed size of a file. Therefore, the effective space a file takes up is larger than 2^{30} bytes, making 4TB potentially insufficient for 4000+ videos.

D)

`EBUSY`, target could not be unmounted as it is busy which means there are dependencies on the target. i.e an active process. umount could fail if target is not a mount point, `EINVAL`, or if the pathname was to a nonexistent component or empty, `ENONENT`.

E)

The caches that improve the performance of the second iteration of a `ls -l` would be buffer caching, inode caching, and dentry caching. The buffer cache improves performance by caching individual sectors. Having read data from that sector of memory prior, any additional demands that require access to that sector for an allotted time will not require the memory to be read from again. The inode cache stores individual inodes and, in our instance, would improve performance by displaying the relevant information produced by `ls -l`. Dentry caching caches directory entry look ups which are directly required by `ls -l`. This would considerably improve performance as an entire traversal of the inquired directory is not needed for access to entries.