Recitation-7 LFS

Name: ChenJiayang ID: 5140379036

Question1: Why an append-only log might lead to h...?

- Because when we want to write into log, we needn't think about where the log entry should be placed and if the entry is needed to be written, it just adds the end of the logfile and after crashing the system just read from the end of the log, which can improve its performance.
- And why is the question raised? The whole chapter doesn't mention this issue.

Question2: How does the log-structured file syste...?

- LFS first buffers all updates (including metadata!) in an in-memory segment; when the segment is full, it is written to disk in one long, sequential transfer to an unused part of the disk
- LFS uses an ancient technique known as write buffering. Before writing to the disk, LFS keeps track of updates in memory; when it has received a sufficient number of updates, it writes them to disk all at once, thus ensuring efficient use of the disk.

Question3: Reads are likely to cause a seek. How to...?

- First put the inode just at the end of the data block.
- Next put the imap next to the inode.
- Finally place a check point region at the beginning of the disk which contains pointers to (i.e., addresses of) the latest pieces of the inode map, and thus the inode map pieces can be found by reading the CR first

Question4: In the log structured file system all the...?

- To ensure that the CR update happens atomically, LFS actually keeps two CRs, one at either end of the disk, and writes to them alternately. It first writes out a header (with timestamp), then the body of the CR. If the system crashes during a CR update, LFS can detect this by seeing an inconsistent pair of timestamps. LFS will always choose to use the most recent CR that has consistent timestamps, and thus consistent update of the CR is achieved.
- To ensure consisitence when crashing during LFS buffer writes in a segment.LFS can recover by simply reading in the checkpoint region, the imap pieces it points to, and subsequent files and directories; however, the last many seconds of updates would be lost. And when it comes to improve this implement, LFS tries to rebuild many of those segments through a technique known as roll forward in the database community.

Question5: Are there certain file access patterns b...?

- When encountering frequent writes to large files, it might cause the problem that first the system write or update files into new disk and then choose to free old version in order to fit new data content. However if the files are large and the write requests are frequent, then it is busy for the disk and LFS to update and write data and meanwhile free and look for new space for subsequent wirtes. Edit By MaHua