Preserving Bi-Modal Utilization for Segment Cleaning in Modern Log-structured Filesystem

Jinsoo Yoo
Hanyang University
jedisty@hanyang.ac.kr

Joontaek Oh
Hanyang University
na94jun@gmail.com

Youjip Won
Hanyang University
yjwon@hanyang.ac.kr

Abstract

Log-structured filesystem buffers all updates in an in-memory segment. When the segment is full, the filesystem writes the updated data to disk in append-only manner. In recent years, the log-structured filesystem has been revaluated because of its out-of-update characteristic which is well aligned with the behavior of SSD. Flash-Friendly Filesystem (F2FS) is designed to fit in flash storage and performs both append-only logging and in-place update depending on the disk utilization.

A segment cleaning process reclaims empty space by discarding old blocks in a victim segment. To reduce the cleaning overhead, it is critical that the victim selection and wear leveling algorithm properly separates the hot blocks and the cold blocks in the filesystem and that the segment utilization forms bi-modal distribution. By classifying hot data from cold data, the log-structured filesystem clusters the hot blocks and cold blocks in the same segments, respectively. As the hot data is easily invalidated, the hot segment has a lower utilization. On the other hand, cold data group together and compose high utilization segments. This bi-modal distribution of the segment utilization enables the log-structured filesystem to select low utilization segment as victim of segment cleaning. With the lower utilization of the victim segment, the log-structured filesystem can mitigate the overhead of copying the live blocks. We find through physical experiment that the modern log-structured filesystem, F2FS, fails to preserve the bimodality. F2FS maintains six types of segments: three hotness levels (hot, warm and cold) for node and data block type, respectively. When the filesystem is clean, the F2FS filesystem successfully seperates the hot blocks. The segment utilization exhibits bimodality. As the segment cleaning proceeds, defficient hotness identification of the F2FS places the hot blocks in the cold segment. The bimodality of the segment utilization gradually disappears. The segment cleaning overhead increases. We examine the utilization of all segments when the segment cleaning is triggered. The workload is designed to exhibit strong temporal locality. In 223Gbyte F2FS partition, we created two files: one for hot access (15 GByte) and the other for cold accesses (135 GByte). The workload generates 4Kbyte random write. The total volume is 2.2 TByte. In the workload, the hot file access accounts for 90% of writes.

During the execution, F2FS performed segment cleaning 20,257 times. We divided the total test time into three stages; the early, interim and late. We checked the utilization of segments selected as a victim of segment cleaning in each stage. Figure 1 shows the frequency distribution of victim segment utilization of each period.

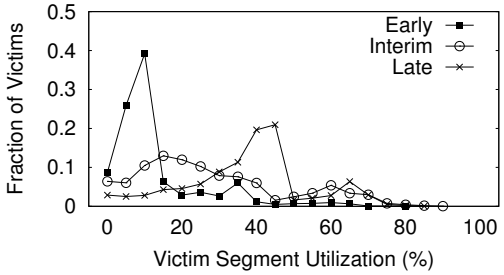

Figure 1: The Frequency Distribution of Victim Segment Utilization

As the filesystem ages, the bimodality of the utilization distribution faded away over time although the test workload generated a clear hot-cold access pattern.

In F2FS, The segment cleaner presumes all live blocks in a victim segment as cold and redirects them to same cold segment. This policy prevents F2FS from classifying hot data separately and blends hot data with cold data in a same segment. As a result, F2FS failed to keep the bi-modal segment utilization and the average utilization of victim segments increases. To address this problem we are developing a new segment cleaning method. The objective of the segment cleaning method is to properly direct the hot blocks in the victim segment. The cleaning method will make F2FS more optimized with flash storage and leverage its ovarall performance.