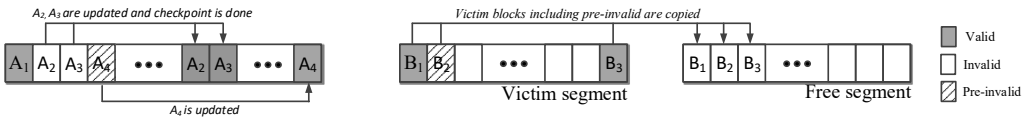
Reducing metadata management overhead of LFS with adaptive checkpoint

Hyunho Gwak, Dongkun Shin
Department of Electrical and Computer Engineering, Sungkyunkwan University, Suwon; Korea
E-mail: dongkun@skku.edu

The log-structured file system (LFS) writes all modifications to storage sequentially with out-of-place update scheme. LFS has been widely used for flash storage since the flash memory does not permit in-place overwrite. However, LFS has a high garbage collection (GC) overhead. During the GC process, valid blocks should be moved to reclaim invalid blocks. This migration of blocks induces metadata update. Eventually, it degrades the performance of LFS. Thus, GC journaling (GCJ) technique was proposed [1]. By removing the checkpoint during GC, the proposed GCJ scheme can significantly reduce the number of metadata flushed during GC. However, it still increases the cost and the processing time of GC. The additional cost is caused by *pre-invalid blocks* which mean invalidated blocks before the last checkpoint. Fig. 1 (a) shows an example. The invalid blocks, A_2 and A_3, were updated and invalidated before the last checkpoint. On the other hand, the pre-invalid block, A_4, was updated after the checkpoint and in-storage metadata refer to the old locations of the pre-invalid blocks. Therefore, if the pre-invalid blocks are overwritten, the file system consistency can be broken at system crash. Thus, LFS treats them as valid blocks until the checkpoint occurs. Fig. 1 (b) shows an example of the increasing GC cost due to copy pre-invalid blocks. LFS copies not only valid blocks but also pre-invalid blocks in the GC procedure. Thus, we propose *adaptive checkpoint* scheme that performs the checkpoint only if the benefit of the checkpoint is bigger than its cost. It calculates the benefit of checkpoint by counting the number of pre- invalid blocks, and the cost by counting the number of cached metadata. Because the cached metadata will be flushed during the checkpoint, it can be calculated as the cost of checkpoint. As a result, the adaptive checkpoint can reduce the number of writes for the GC by selecting checkpoint adaptively. We implemented the adaptive checkpoint in F2FS [2], and evaluated it using LFS GC workload. We observed that it improves performance by 25% which is 8% more than GCJ.



(a) Invalid and Pre-invalid block (b) Garbage Collection with metadata journaling

Fig 1. The additional cost of GC by copying the pre-invalid block

Acknowledgments This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the SW Starlab support program(IITP-2017-0-00914) supervised by the IITP(Institute for Information & communications Technology Promotion.

References [1] Gwak, Hyunho, et al., "Reducing garbage collection overhead of log-structured file systems with GC journaling," ISCE, 2015. [2] Lee, Changman, et al., "F2FS: A New File System for Flash Storage," FAST, 2015.