High Performance and High Reliable File System for Car Digital Video Recorders

Younghun Kim and Dongkun Shin
Sungkyunkwan University, Suwon, Korea
ehdeoddl@skku.edu, dongkun@skku.edu

***Abstract*— Car digital video recorders store real-time audio/video data at flash memory storage devices. They require high performance and high reliable file systems. However, current file systems are vulnerable to sudden-power-off and flash memory errors. This paper proposes flash-aware cluster allocation techniques that can reduce the probability of file system corruptions and file fragmentation. Experiments on a real system showed that the proposed techniques can prevent the file fragmentation and can improve the write performance and SD card lifetime significantly.**

I. Introduction

Car digital video recorder (DVR), which is also called car black box, records real-time audio/video data at every moment of drive. It can also record audio/video data at every accidental or suspicious event of a parked car. Generally, the captured multimedia data are saved in a secure digital (SD) card, which uses NAND flash memory as storage media. The SD card is formatted by PC-compatible FAT file system. The car DVR should provide a high reliability on recording the data since it can be used for evidence in legal action and the car DVR is exposed to system failures such as sudden-power-off (SPO). However, the current file systems used in car DVRs are not designed considering the required high reliability.

First, the current FAT file system of car DVR system updates metadata frequently. The FAT file system metadata includes the file allocation table (FAT) and the directory entry (DE). The FAT has the cluster chain information for created files. The cluster is the basic unit for space allocation and an entry of FAT has the allocation status of the corresponding cluster. The FAT is modified when any cluster is allocated or freed. The DE has the file information such as the file name, the file size, the first cluster number, the date/time, etc. It is modified when a new file is created or deleted. Since the car DVR makes a new file at every predefined time interval and ceaselessly records the driving data, the metadata are frequently updated. If the metadata are corrupted by SPO or flash memory errors, the recorded data cannot be accessed. The frequent metadata updates increase the probability of file system corruptions.

In particular, there is a mismatch between the size of FAT entry and the size of flash memory page. While an FAT entry is 4 byte, the flash memory page is generally 4 KB. Since the flash memory can be programmed by the unit of page size, all the flash page should be reprogrammed even though the file system wants to modify only a 4-byte FAT entry. Actually, the minimum data transfer unit between file system and storage device is a 512-byte sector. Since flash memory does not permit in-place update, a read-modify-write (RMW) operation is required to update only a partial page, that is, the old flash page is first read into a buffer, several sectors are modified in the buffer, and the modified page is written at a new flash page. The RMW will increase the latency of FAT update operation. In addition, SD card has a limited program/erase (P/E) cycles. The small write requests requiring the RMW operations will exhaust rapidly the P/E cycles of SD card. Some flash memory devices permit the partial page programming, and thus they can reduce the number of RMW operations for updating FAT. However, the partial page programming can corrupt other FAT entries that are previously written at the target flash page when there is an SPO or flash page program error.

The second problem is the file fragmentation. Car DVR generates several different types of data and places them in different folders. The data can be categorized based on when they are generated. For example, continuously recorded video/audio files are placed into the "driving" folder. Vehicle accidents and driving events triggered by the internal gyroscope are placed into the "driving event" folder. The videos triggered by the motion detection and collision detection during parking are stored into the "parking motion" and "parking collision" folders, respectively. Each directory has its limited space and is recycled automatically when the directory space becomes full. The oldest files are overwritten first. Therefore, the files in different directories have different lifetimes and file sizes. While these directories are separated logically, the data of a file can be written at any cluster in the file system irrespective of the file type. Generally, the FAT file system allocates the first free cluster. Since the lifetimes of recorded files are different, the free clusters are fragmented and the newly created file will be written into several fragmented clusters.

The file fragmentation incurs random write operations at SD card. Since flash memory storage show worse performance for random writes compared with sequential writes, the file fragmentation will degrade the file system write performance, and can invoke frame drops at the worst case. In addition, since the random write requests invoke a significant write amplification within flash memory devices, the lifetime of SD card will be reduced by the fragmented write requests. Unfortunately, any defragmentation technique cannot be a solution. Since car DVRs should ceaselessly write the captured data at SD card, it has no time for defragmentation job.

This research was supported by the Center for Integrated Smart Sensors funded by the Ministry of Science, ICT & Future Planning as Global Frontier Project (CISS-2011-0031863).

There have been several optimization techniques for FAT file systems. The all cluster pre-allocation (ACPA) [1] allocates all required clusters at once for a created file to reduce the number of FAT update operations. However, the ACPA does not consider the partial page programming issue and cannot prevent the file fragmentation. The flash-aware extension-based cluster allocation (FECA) [2] allocates a large size of contiguous clusters, called cluster group, in order to alleviate the file fragmentation and frequent FAT updates. However, a large file can be fragmented into several separated cluster groups, and several small files can be mixed within a cluster group. To solve the problems of car DVR file systems, this paper proposes two flash-aware cluster allocation techniques, the partitioned cluster allocation (PCA) and the page-aligned pre-allocation (PAPA).

II. Flash-Aware Cluster allocation

The PCA is an anti-defragmentation technique. It divides the storage space into several partitions as shown in Fig. 1. Each partition is used for only single-type files. The type of file can be identified from the directory information of the file. Since the oldest files are deleted when the directory space becomes full, the cluster allocation can be performed in a circular manner and each file can use a contiguous cluster region without fragmentation. The PCA manages the partition information such as the start cluster, the last cluster, and the first free cluster of each partition. The information is managed in memory at run time and is saved at a special file when the file system is unmounted. Even for abnormal shutdown, the recovery operation can rebuild the partition information by scanning the storage space. The PCA partitions are logically managed by the file system. While the physical partitions cannot be reconfigured without formatting, the proposed logical partitions can be easily reconfigured at run time.

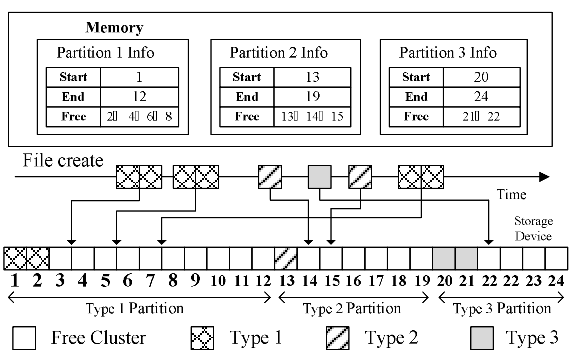


Fig. 1 Partitioned cluster allocation.

The PAPA uses a pre-allocation technique to modify the FAT as the unit of flash memory page. In Fig. 2, when the clusters for 'File 1' are allocated, the original FAT file system will modify only four FAT entries in the logical page number (LPN) 1 of SD card. When the 'File 2' is created, the LPN 1 should be rewritten invoking the RMW operation or the error-prone sub-page programming. However, the proposed PAPA pre-allocates the following five clusters for the next file to avoid the overwrite operation on the flash page with LPN 1.

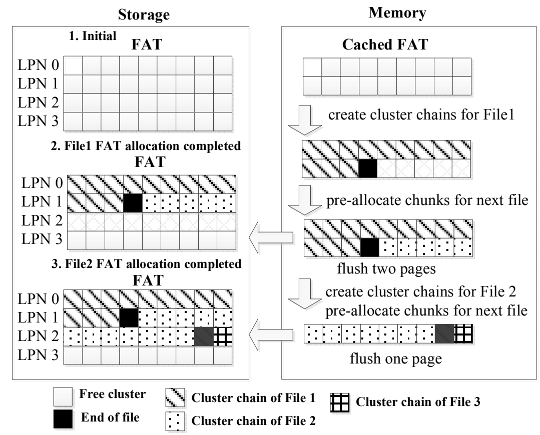
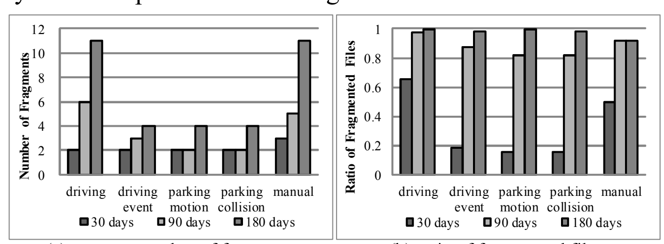


Fig. 2 Page-aligned pre-allocation.

III. Experiments

With a real car DVR platform equipped with a class 10 SD card, the effects of the PCA and PAPA techniques are evaluated. The PCA technique permits only two fragmented cluster regions at maximum when the clusters are allocated across the partition boundary. However, the original FAT file system suffers from the fragmentation problem as shown in Fig. 3. Most of the files are fragmented after 180 days, and one file can be fragmented into more than 10 separated cluster regions. The fragmented files show significant low write performances. The file fragmented into 11 regions increased the write latency by 75% and decreased the SD card lifetime by 36% compared to a non-fragmented file.



(a) average number of fragments (b) ratio of fragmented files

Fig. 3 Fragmentation of the original FAT file system

As shown in Tab. 1, the PAPA technique reduces the file system overhead, the device driver overhead, and the storage write latency. This is because it reduces the number of free cluster search operations and the number of write requests.

Tab. 1 Performance improvement by PAPA.

	Original	PAPA
FS overhead (ms)	72.1	6.3
D/D overhead (ms)	116.9	5.8
Storage write latency (ms)	84.7	7.6
Number of free cluster searches	2500	0
Number of write requests	21	1

Reference

[1] S. Park and S.Y. Ohm, "New Techniques for Real-Time FAT File System in Mobile Multimedia Devices," IEEE Transactions on Consumer Electronics, Vol. 52, No. 1, pp. 1-9, 2006.

[2] S. Ryu, C. Lee, S. Yoo, and S. Seo, "Flash-aware cluster allocation method based on filename extension for FAT file system," Proc. of the 2010 ACM Symposium on Applied Computing, pp. 502-509, 2010.