Investigation into RAID Front External Journaling with SSD

Benjamin Young
Virginia Commonwealth University
youngbm3@vcu.edu

Modern Hard Disk Drives (HDD) are large and cheap, but generally slow relative to the rest of the memory hierarchy. Yet, these drives are necessary for long term storage and reliability due to their cost per GB, but recently different memory types are becoming popular by compromising between the storage size and cost of HDDs, with better performance. These memory types, such as Solid State Drives (SSD) and Phase Change Memory (PCM) may not immediately aim to replace HDDs in their entirety, but they do aim to displace some of the unnecessary workload given to the disk drives. One such workload is Journaling; the necessary act of logging any changes that are queued to the File System (FS) memory hierarchy such that, in the event of a system failure, corrupted transactions can be replayed, thus sparing the persistent storage from error checking operations. In a system where many FS are writing to a single storage unit, such as a RAID of HDDs, which does not have dedicated memory for journaling, the act of journaling incoming data causes additional head seeks and parity recalculations, thus reducing performance. In software RAIDs it is possible to disable the RAID journaling feature altogether to maintain performance.

However, even though the RAID controller may not be journaling data that is sent to it before committing to disk, the FS of each node journals the sent data on its own local storage. This project aims to increase parallelism by outsourcing the journal of each FS to a centralized SSD, which can also be used as a cache front for the RAID system. In this manner, the RAID will have access to the journals, and will be able to see all changes queued; in the event of a system failure, the RAID will have direct access to the journal log for recovery. SSDs are a desired choice for this project due to their decreasing costs and high performance. Research has shown that externally journaling increases performance drastically due to parallelism, reduced head movement, and the ability to force journal-to-disk flush operations to be hidden in the background.

This project requires modifications to the way that the journaling is conducted, as well as its means of storage. One such modification is the access pattern that all the FS aggregately contribute to; an SSD works well with a long depth queue so that it can write as much data to a block as possible, since the entire block will need to be copied and updated to a new location under a page write operation. Additionally, data within each journal will need to be marked and traced to indicate its write status; this way, when the RAID needs to recover, it can search the marked data throughout all of the journals, instead of processing every log entry of each journal.

The performance gains from simply outsourcing the journal to a separate device than that which holds the FS shows that performance gains should be seen for not only the RAID systems (employing journaling), but also for the nodes themselves. Additionally, the reliability of the RAID system is increased.