CompulsiveFS : Making NVRAM Suitable for Extremely Reliable Storage

Kevin M. Greenan, Ethan L. Miller

Byte-addressable, non-volatile memory (NVRAM) technologies such as magnetoresistive random access memory and phase-change memory have recently emerged as viable competitors to Flash RAM. These new technologies have the ability to improve the performance, reliability and power consumption of current storage systems. NVRAM generally operates—in disk-based storage systems—as a low-latency write cache and improves data reliability during a power loss or a system crash. CompulsiveFS takes advantage of the performance benefits recognized when storing persistent metadata in NVRAM and creates a durable file system which improves the reliability of NVRAM-resident data in the face of software errors, hardware failures and system crashes.

Currently, page protection is used to protect a write cache from software errors in the event of a system crash [1]. Page-protection uses page-level access control to mark a set of pages as read-only. Every write to a read-only page results in a memory exception. Valid writes must mark a page as writable before writing and revert to read-only once the write has completed. If most of the writes are relatively large, then page protection results in very little overhead. In contrast such protection over frequent, small write workloads results in unnecessary overhead; requiring two permission changes per write. Such overhead turns out to be prohibitive for our metadata-centric workload of many small writes. Since CompulsiveFS primarily stores file system metadata in NVRAM, the protection mechanisms must cater to a small-write workload and must have the ability to recover from hardware or software errors in NVRAM.

Existing NVRAM-based file systems do not include features that effectively guard against file system corruption or NVRAM corruption. We are designing CompulsiveFS to address this problem by providing file storage that can survive multiple errors in NVRAM, whether caused by errant operating system writes or by memory corruption. CompulsiveFS uses three mechanisms to maintain file system consistency. First, an erasure-encoded log structure is used to reliably to store persistent metadata. Second, CompulsiveFS checks integrity on *every* file system operation. Finally, NVRAM is periodically scanned to ensure that all NVRAM-resident data is in a consistent state. While CompulsiveFS is designed for reliability, we expect it to have excellent performance, thanks to the ability to do word-aligned reads and writes in NVRAM.

Maintaining log consistency is extremely crucial as the log holds the only copy of the file system metadata. Metadata integrity may be compromised due to file system errors, wild writes, media wear or in the case of a multiple NVRAM banks, media failure. The log structure allows us to periodically verify file system consistency by comparing log transactions to the live state of the system. In order to conserve space, CompulsiveFS will contain algorithms that clean the log as transactions are replayed. The log-based structure also enables CompulsiveFS to perform incremental erasure-encoding and signature computation over the contents of the log, resulting in a highly efficient, fault-tolerant log. By embedding parity and signatures in the log, CompulsiveFS has the ability to periodically scan, check and correct errors in NVRAM.

Preliminary results taken from an erasure-encoded log implementation against page protection show that standard page protection on small-write workloads results in unwanted overhead. Our prototype log achieves throughput rates an order of magnitude faster than page protection during small writes (10-50 bytes). We believe that most of the page protection overhead is caused by fixing up the page tables and flushing regions of the TLB during each protection call. Even when ignoring the overhead, page protection alone does not provide any form of error correction.

As the size of storage systems increase, so does the probability of corruption and the time required to make repairs. We now live in a world where system outages are costly and data loss is unacceptable. In an effort to prevent permanent damage as a result of corruption and minimize system downtime, CompulsiveFS periodically verifies the consistency of file system structures and all NVRAM-resident data. File system structures are checked using transactional information in the file system log, while the consistency of the NVRAM-resident log is maintained using erasure codes. We believe such on-line verification is a requirement for any large-scale storage system.

CompulsiveFS is still in the early stages of research, design and implementation. Our preliminary tests do not give any indication of how effective an erasure-encoded log will be in terms of protection. We hope to show that our mechanisms provide just as much or better protection than page-based access control. Currently, we are working on the structures necessary to efficiently index extents on disk, the algorithms required to clean and check the log, wear-leveling algorithms, log structure across multiple NVRAM banks, the use of CompulsiveFS in a distributed file system, effective error injection techniques and ideal system parameters (check intervals, amount of parity,segment sizes, etc.).

References

[1] P. M. Chen, W. T. Ng, S. Chandra, C. Aycock, G. Rajamani, and D. Lowell. The Rio file cache: Surviving operating system crashes. In *Proceedings of the 7th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, pages 74–83, Oct. 1996.