RAID4S: Adding SSDs to RAID Arrays

Rosie Wacha (student, rwacha@cs.ucsc.edu)*, Scott A. Brandt (scott@cs.ucsc.edu)*,
John Bent (johnbent@lanl.gov)† and Carlos Maltzahn (carlosm@cs.ucsc.edu)*
*University of California, Santa Cruz
†Los Alamos National Laboratory

Please consider for both WiP and poster presentation.

Solid State Drives (SSDs) provide faster random I/O and use less power than hard drives, but are not yet cheap enough to substitute for all of the drives in large-scale storage systems. We present RAID4S, a cost-effective, high-performance technique for integrating SSDs into a RAID array. Our preliminary results demonstrate that an N+1 RAID4S array can perform up to N times better than a similar RAID4 array and up to two times better ($2N/(N+1)$, to be precise) than a similar RAID5 array without any of the complexity that arises from spreading the parity among all of the disks.

Small writes limit RAID performance by requiring 4 I/Os per data block written. RAID5 improves on RAID4 by distributing the parity update load over all of the disks in the array, but they must still handle that extra parity load. RAID4S replaces the parity disk in a RAID4 array with a fast SSD device, speeding up the parity accesses so that multiple small writes to different data disks may proceed in parallel while the set of parity accesses occurs on the SSD. If the SSD is N times faster than disk, N separate single-block I/Os may be performed to N data drives at the same time that all N parity I/Os are performed on the SSD parity device. This allows complete parallelization of the N small writes, which complete as a set in the time it takes to complete a single small write in RAID4 or $\lceil (N+1)/2 \rceil$ small writes in RAID5.

Most workloads include small and large reads and writes. Large writes still happen at the speed of the slowest disk, but with one less disk large writes may see a slight increase in performance. Reads are unaffected. Overall, the performance improvement from RAID4S depends upon the speed of the SSD relative to the disks and the fraction of small writes in the workload. As caches grow and become more effective, large storage workloads are increasingly write-intensive. Some techniques (e.g., WAFL [1]) transform small writes into large writes, but some small writes are inevitable, particularly as the disks fill up. RAID4S addresses this problem. Although large writes see only marginal performance improvement, large writes and reads underutilize the SSD, providing additional time that may be spent performing parity operations for small writes and allowing a RAID4S system to provide additional speedups on mixed workloads with SSDs less than N times faster than disk. Degraded mode operation is also improved by RAID4S over traditional RAID4 and RAID5 by offloading the more frequent parity writes to a single faster device. Finally, the faster random-access performance of the parity device may enable more sophisticated reconstruction techniques.

References

[1] D. Hitz, J. Lau, and M. Malcolm, "File system design for an NFS file server appliance," in *Proceedings of the Winter 1994 USENIX Technical Conference*, San Francisco, CA, Jan. 1994, pp. 235–246.

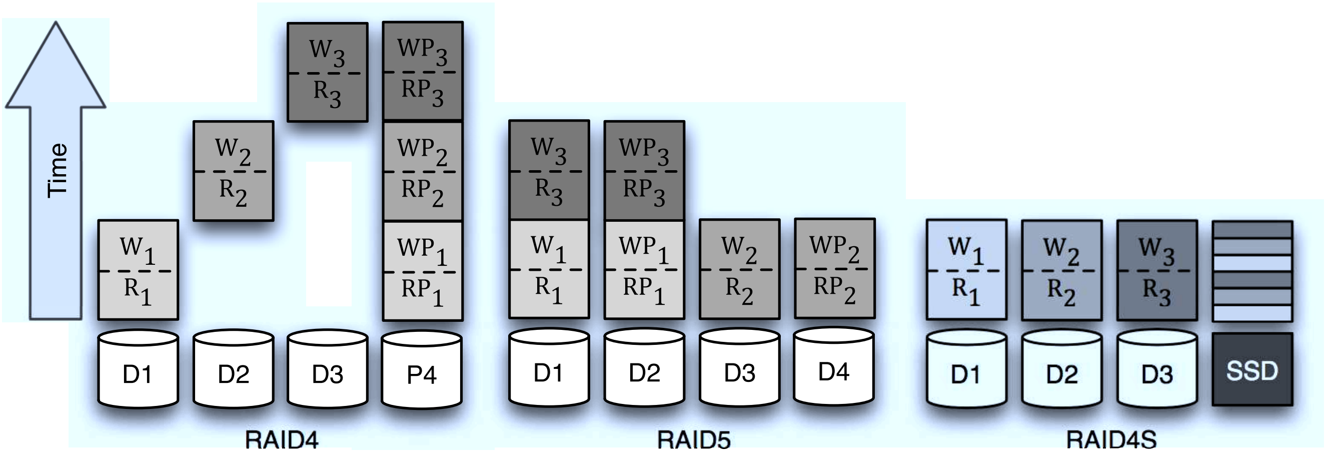

Fig. 1. The time to complete three small writes is compared for RAID4, RAID5, and RAID4S. RAID4S with three disk drives allows the complete parallelization of N=3 accesses to different disks, thereby finishing the work more quickly.