Author Retrospective for Energy Conservation Techniques for Disk Array-based Servers

Eduardo Pinheiro
Google

Ricardo Bianchini
Rutgers University and Microsoft

Abstract. This is a retrospective on our original paper titled "Energy Conservation Techniques for Disk Array-based Servers", which was published in the Proceedings of the International Conference on Supercomputing in 2004.

Original paper: http://dx.doi.org/10.1145/2591635.2591666

Categories and Subject Descriptors

D.4 [**Operating systems**]: Storage management

Keywords

Energy conservation; disk power; disk arrays

Back then... The early 2000s were an exciting time for systems research. At that time, Internet services' datacenters were becoming larger and more sophisticated. In 2001, researchers at Duke University and our own group at Rutgers University published the first papers [4, 16] arguing that energy management would become critical for datacenters. Until then, energy management had only been considered for battery-operated devices, such as laptop computers.

A little later, researchers also started to consider the energy consumption of the storage subsystem in datacenters and supercomputers. This line of research started in 2002 with the MAID (Massive Array of Idle Disks) system [5], which used hard disk drives for large-scale storage, instead of tapes. To cope with the greater energy consumption of disks, MAID proposed that most of them be kept asleep (i.e., off); only a relatively small number of additional "cache disks" would be active at all times. A cache disk miss would then involve the activation of the (sleeping) disk that contained the desired data. Upon reading the data from this disk, the system would store it on any cache disk assuming that more requests for it would arrive. The activated disk would be transitioned back to sleep after a period of idleness.

Another key advance related to our project appeared in the following year. In 2003, researchers at Penn State University and our own group proposed that disks should be able to perform accesses at one of multiple rotation speeds [7, 3]. The idea was to adjust a disk's rotation speed dynamically, according to the amount of load offered to it. Since the spindle motor consumes the largest fraction of a disk's power and reducing rotation speed has a quadratic impact on power consumption [7], this approach can accrue significant energy savings under low and moderate loads.

Our project and PDC. By early 2003, it had become clear to us that neither MAID nor multi-speed disks would produce the highest energy savings in disk array-based servers. MAID involves the energy (and management) overhead of the cache disks and depends heavily on the number of such disks for a good performance/energy tradeoff. Simply using multi-speed disks often limits savings, as the data layout may force all disks to operate at relatively high speeds.

Given these observations and early evidence of a high data popularity skew in certain server workloads [2, 9], we decided to investigate whether segregating popular and unpopular data onto separate disks via migration would enable higher and more robust energy savings [15]. We called this approach Popular Data Concentration (PDC). PDC would not require extra disks and would enable a subset of disks (those storing unpopular data) to be in a low-power state longer.

To avoid performance degradation for popular data, PDC should avoid overloading the popular disks. Thus, we needed to consider the expected access rate of each disk explicitly. We could do this by estimating the future load (in MBytes/second) on each disk to be the sum of the recent load directed to the data to be stored on it. PDC should then only migrate data onto a disk until the expected load on the disk is close to its maximum throughput for the workload. Keeping track of data popularity and offered load efficiently was a challenge, but we could leverage prior work on second-level storage caching algorithms [20].

The final step was to build a PDC-based system that would enable high performance and energy savings, despite the high energy overhead of data migration. We implemented PDC in a system we call Nomad FS. Nomad FS is an energy-aware file server that we implemented at the user level on top of a local UNIX file system. It explicitly caches data blocks in main memory, again at the user level. Although Nomad FS receives requests for 8-KByte file blocks, it migrates entire files according to PDC. To conserve energy in disk arrays composed of conventional (single-speed) disks, Nomad FS spins disks down after a fixed period of idleness. A spun down disk is automatically reactivated on the next access to it. For servers with (autonomous) two-speed disks [3], Nomad FS does not explicitly effect power

ICS 25th Anniversary Volume. 2014
ACM 978-1-4503-2840-1/14/06.
http://dx.doi.org/10.1145/2591635.2591666.

mode transitions. Furthermore, it does not migrate files out of disks that are already running in low speed to reduce the migration overhead.

As commercial two-speed disks were not available and we wanted to explore a large space of parameters quickly, we also built a simulator of Nomad FS. For comparison with PDC, we implemented MAID in a variation of Nomad FS and in simulation. We validated the simulator against our real implementations of PDC and MAID using conventional disks. Our parameter exploration results were positive. In summary, we found that PDC conserves substantial energy during periods of light load on the file server (at a very small response time penalty) with two-speed disks. We also found that PDC can deal more gracefully with high request rates and low file popularity than MAID or two-speed disks without data movement. In addition, PDC achieved consistent energy savings for most of the parameter space. However, the PDC gains degrade substantially for long migration intervals. In comparison, the behavior of MAID is heavily dependent on the number of cache disks used. Furthermore, the energy overhead of these disks is pronounced in several parts of the space. We also found that using two-speed disks without data movement behaves well when disk loads remain light most of the time. For conventional disks, PDC can only conserve energy when the load on the server is low, but those savings come at a high performance overhead for the (typically small) percentage of requests that get delayed by disk spin-ups.

In a 2006 paper [17], we proposed a new energy management technique for redundant large-scale storage systems (called Diverted Accesses) and combined it with PDC, assuming conventional disks.

Impact of PDC. PDC has had an important impact on the community. As of March 2014, our original PDC paper has been cited by 300+ other papers, according to Google Scholar. More importantly, many published systems have clearly been influenced by PDC. For example, Hibernator [21] combines PDC with RAID (Redundant Array of Inexpensive Disks) and multi-speed disks. Hibernator improved on our migration scheme by grouping the disks into "tiers" (all disks in a tier are organized as a RAID and spin at the same speed). Migrations would then only need to happen across tiers. The SEA (Striping Energy-Aware) data placement scheme [19] took a similar approach in combining PDC, RAID, and two-speed disks.

Many papers have combined PDC with solid-state drives (SSDs). For example, Kim *et al.* proposed an extension of PDC, called Pattern-based PDC [12], which migrates frequently read data to a conventional hard disk and frequently written data to an SSD. Deng *et al.* considered combining an SSD, a conventional hard drive, and popularity-based migrations between them in the design of energy-aware disk drives [6]. Lee and Koh took a similar approach for arrays of conventional disks and called their approach PDC-NH [13].

More broadly, PDC has also influenced research in energy-aware distributed systems and even energy-aware main memory systems. GreenHDFS [10] and Lightning [11] are energy-aware distributed file systems that implement popularity-based data migration between "storage zones" comprising groups of servers with conventional disks. For main memory, Huang *et al.* [8] and Wu *et al.* [18], for example, dynamically place data in memory ranks based on popularity. Not surprisingly, we leveraged PDC-style data placement in our own work on energy-aware memory [14].

Finally, Bostoen *et al.* produced an extensive survey of energy management for storage systems [1]. The paper discusses some of the above systems, and also places PDC at their roots.

Looking forward. In recent years, research on techniques tailored to SSDs (often used as caches for disks) and redundant large-scale disk storage (e.g., Diverted Accesses) has become popular. We expect these topics to remain in evidence for the foreseeable future, as SSD costs per bit keep decreasing and cloud storage demand keeps increasing. Despite these more recent topics, interest in PDC is strong as ever: the two years in which the original PDC paper received the most citations have been last year and the year before. With this in mind, we hope that PDC, the concepts that surround it, and the parameter space exploration in our paper will remain relevant for years to come.

Acknowledgements. We would like to thank Vinicio Carrera and Cezary Dubnicki for their help with many aspects of our energy-aware storage work. We are also indebted to the National Science Foundation for its support of our work via grants CCR-0238182 (CAREER award) and EIA-0224428.

1. REFERENCES

[1] T. Bostoen, S. Mullender, and Y. Berbers. Power-Reduction Techniques for Data-Center Storage Systems. *ACM Computing Surveys*, 45(3), 2013.

[2] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker. Web Caching and Zipf-like Distributions: Evidence and Implications. In *Proceedings of IEEE InfoCom'99*, 1999.

[3] E. V. Carrera, E. Pinheiro, and R. Bianchini. Conserving Disk Energy in Network Servers. In *Proceedings of the 17th International Conference on Supercomputing*, 2003.

[4] J. Chase, D. Anderson, P. Thackar, A. Vahdat, and R. Boyle. Managing Energy and Server Resources in Hosting Centers. In *Proceedings of the 18th Symposium on Operating Systems Principles*, 2001.

[5] D. Colarelli and D. Grunwald. Massive Arrays of Idle Disks For Storage Archives. In *Proceedings of the 15th High Performance Networking and Computing Conference*, 2002.

[6] Y. Deng, F. Wang, and N. Helian. EED: Energy Efficient Disk Drive Architecture. *Information Sciences*, 178(22), 2008.

[7] S. Gurumurthi, A. Sivasubramaniam, M. Kandemir, and H. Franke. DRPM: Dynamic Speed Control for Power Management in Server Class Disks. In *Proceedings of the International Symposium on Computer Architecture*, 2003.

[8] H. Huang, K. G. Shin, C. Lefurgy, and T. Keller. Improving Energy Efficiency by Making DRAM Less Randomly Accessed. In *Proceedings of the 2005 International Symposium on Low Power Electronics and Design*, 2005.

[9] S. Jin and A. Bestavros. GISMO: A Generator of Internet Streaming Media Objects and Workloads. *ACM SIGMETRICS Performance Evaluation Review*, 29(3), 2001.

[10] R. Kaushik and M. Bhandarkar. GreenHDFS: Towards an Energy-Conserving, Storage-Efficient, Hybrid Hadoop Compute Cluster. In *Proceedings of the USENIX Annual Technical Conference*, 2010.

[11] R. Kaushik, L. Cherkasova, R. Campbell, and K. Nahrstedt. Lightning: Self-Adaptive, Energy-Conserving, Multi-Zoned, Commodity Green Cloud Storage System. In *Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing*, 2010.

[12] Y.-J. Kim, K.-T. Kwon, and J. Kim. Energy-efficient File Placement Techniques for Heterogeneous Mobile Storage Systems. In *Proceedings of the 6th ACM/IEEE International Conference on Embedded Software*, 2006.

[13] D. Lee and K. Koh. PDC-NH: Popular Data Concentration on NAND Flash and Hard Disk Drive. In *Proceedings of the 10th IEEE/ACM International Conference on Grid Computing*, 2009.

[14] V. Pandey, W. Jiang, Y. Zhou, and R. Bianchini. DMA-Aware Memory Energy Management. In *Proceedings of the 12th Symposium on High-Performance Computer Architecture*, 2006.

[15] E. Pinheiro and R. Bianchini. Energy Conservation Techniques for Disk Array-Based Servers. In *Proceedings of the 18th International Conference on Supercomputing*, 2004.

[16] E. Pinheiro, R. Bianchini, E. V. Carrera, and T. Heath. Load Balancing and Unbalancing for Power and Performance in Cluster-Based Systems. In *Proceedings of the Workshop on Compilers and Operating Systems for Low Power*, September 2001.

[17] E. Pinheiro, R. Bianchini, and C. Dubnicki. Exploiting Redundancy to Conserve Energy in Storage Systems. In *Proceedings of SIGMETRICS*, 2006.

[18] D. Wu, B. He, X. Tang, J. Xu, and M. Guo. Ramzzz: Rank-aware DRAM Power Management with Dynamic Migrations and Demotions. In *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*, 2012.

[19] T. Xie. SEA: A Striping-Based Energy-Aware Strategy for Data Placement in RAID-Structured Storage Systems. *IEEE Transactions on Computers*, 57(6), 2008.

[20] Y. Zhou, P. Chen, , and K. Li. The Multi-Queue Replacement Algorithm for Second-Level Buffer Caches. In *Proceedings of the 2001 USENIX Annual Technical Conference*, 2001.

[21] Q. Zhu, Z. Chen, L. Tan, Y. Zhou, K. Keeton, and J. Wilkes. Hibernator: Helping Disk Arrays Sleep Through the Winter. In *Proceedings of the 20th ACM Symposium on Operating Systems Principles*, 2005.