ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Data and Parity Storing Techniques:A Survey

Kavitha Sreekala Krishnan*, S. ArunKumar

* PG Scholar, Department of Electronics and Communication Engineering,
PSNA college of Engineering and Technology,
Dindigul -624619, India.
Assistant Professor, Department of Electronics and Communication Engineering,
PSNA college of Engineering and Technology,
Dindigul -624619, India.

Abstract

RAID is a data storage virtualization technology that combines multiple disk drive components into a logical unit for the purpose of data redundancy or for the performance improvement. Data is distributed across the drives in several ways which is referred to as RAID levels, depending on the specific level of redundancy and performance requirement. Solid-state drive RAID is a strategy for improving performance that involves dividing and storing the same data on multiple solid state drives.A group of solid-state drives (SSDs) are connected together to create a redundant array of independent disks (RAID). An operating system sees the RAID as one large disk, but since read and write operations are being spread out over multiple disks, inputs/outputs (I/Os) can be carried out simultaneously, so that performance speed can be improved. In this paper we will discusss about various method by which data can be moved to and from other storage devices and how they overcomes the problem that occur when we store data in disk drives.

Keywords: Solid-state drive, Redundant Array of Independent Disk (RAID), data storage virtualization tehnology.

Introduction

A data storage device is a device for recording information .A storage device is a computing hardware that is used for storing, porting and extracting data files and objects. It can hold and store information both temporarily or permanently, and can be internal or external to a computer, server or any other computing device. A storage device can also be known as a storage medium. A storage device can hold and process information, or both. Device that only holds information is known as a recording medium. Devices that process information can either access a separate portable recording medium or a permanent component to store and retrieve information. There are two different types of storage devices:

- Primary Storage Devices: It is smaller in size, and are designed to hold data temporarily and are internal to the computer.
- Secondary Storage Devices: These have large storage capacity, and can store data permanently. They can be both internal or external to the computer, and they include the hard disk, compact disk drive and USB storage device.

A hard disk drive (HDD) is a data storage device used for storing and retrieving digital information using rapidly rotating disks coated with magnetic material. An HDD retains its data even when powered off. Data is read in a random-access manner, i.e; individual blocks of data can be stored or retrieved in any order rather than sequentially. An HDD consists of one or more rigid rapidly rotating disks with magnetic heads arranged on a moving actuator arm to read and write data to the surfaces. HDDs are connected to systems by standard interface cables such as SATA (Serial ATA), USB or SAS (Serial attached SCSI) cables. The main drawback of HDD is the storage size, low cost,reliability, access time, power consumption and generates noise. To avoid all these drawbacks we have introduced solid state drives.

A solid-state drive (SSD) is a data storage device using integrated circuit assemblies as memory to store data persistently. SSD technology uses electronic interfaces compatible with traditional block input/output (I/O) hard disk drives, thus permitting simple replacement in applications. SSDs uses NAND-based flash memory, that retains data without power. For applications requiring faster access, but not necessary data persistence after power loss, SSDs can be constructed from

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

random-access memory (RAM). Such devices may employ separate power sources to maintain data after power loss. Hybrid drives or solid state hybrid drives (SSHD) combine the features of both SSDs and HDDs in the same unit, containing a large hard disk drive and an SSD cache to improve performance of frequently accessed.

Materials and methods

Serial Attached SCSI Interface

Serial Attached SCSI (SAS) is a point-to-point serial protocol that moves data to and from computer storage devices such as hard drives and tape drives. SAS uses the standard SCSI command set. A typical Serial Attached SCSI system consists of the following basic components: an initiator, a target, a service delivery subsystem and an expander. SAS is the first standard specification to provide an interconnect mechanism for both SCSI and Serial ATA (SATA). SAS meets both enterprise and midrange storage requirements at relatively low costs, providing users and integrators with flexible storage architectures. The SAS bus operates point-to-point. Each SAS device is connected by a dedicated link to the initiator, unless an expander is used. If one initiator is connected to one target, there is no opportunity for contention. SAS has no termination issues and does not require terminator packs. SAS eliminates clock skew. SAS allows up to 65,535 devices through the use of expanders. SAS allows a higher transfer speed (3 or 6 Gbit/s) . SAS achieves these speeds on each initiator-target connection, hence getting higher throughput. SAS use the SCSI command-set. SAS controllers may be connected to SATA devices, either directly using native SATA protocol or through SAS expanders using SATA Tunneled Protocol (STP). The Serial Attached SCSI standard defines several layers (in order from highest to lowest): application, transport, port, link, PHY and physical. Serial Attached SCSI comprises three transport protocols:Serial SCSI Protocol (SSP) ,Serial ATA Tunneling Protocol (STP) and Serial Management Protocol (SMP) . Serial Attached SCSI (SAS), bring more flexible storage solutions to end users and storage/systems integrators in a variety of ways. The SAS protocol provides a tunneling mechanism for delivering SATA frames through SAS connection infrastructure, including physical cabling connections. Serialization of the SCSI interface overcomes the physical and functional limitations of parallel interface technology. SCSI offers scalability, flexibility, and cost-effectiveness for connectivity, data transport, and data storage. Increased bandwidth requirement, as well as challenges presented by clock skew and power consumption, prevent SCSI from moving beyond the Ultra320 specification.

In[6] Cai, Y., Agere Syst., Allentown PA, Fang L, Ratemo R, Liu J proposed that SOCs and hard drive controller ICs with integrated SAS links have extremely high production volume, high profit margin pressure, and a low defect rate requirement. On the other hand, constant need for small die size, low power, fast design cycle, rapid increase in performance specs and price pressure does not normally allow for much design margin.

Hybrid flash file system

In [4] Chul Lee, Sung Hoon Baek and Kyu Ho Park proposed a hybrid flash file system (HFFS) based on both NOR and NAND flash memory. In a conventional NAND based flash file system, there is a trade-off between life span and durability in the frequent writing of small amount of data. Because NAND flash supports only a page-level I/O, atleast one page is wasted in the synchronous writing of small amount of data. The wasting of pages reduces the utilization and life span of the NAND flash. To alleviate the utilization problem, some NAND based flash file systems write small amounts of data asynchronously with RAM buffers, though buffering in RAM decreases the durability of the system. HFFS eliminates the trade-off between life span and durability. It synchronously stores data as a log in the NOR flash, whenever we append small amount of data to a file. The merged logs are then flushed to the NAND flash in a page-aligned fashion. The erase count of the NAND flash is at most one cycle over the whole erase blocks, whereas the erase count of the NOR flash is less than 50 cycles.

In [5] S.H. Lim and K.H. Park proposed the implementation of our HFFS is based on our previous NAND flash-based file system, called CFFS. The experimental results reveal that our HFFS provides a longer life span than a conventional NAND flash-based synchronous flash file system with a similar level of durability. The HFFS does not generate garbage because it merges a small amount of data in the NOR flash. NVMFS, is used to resolve the random write issue of SSDs. First, NVMFS distributes data dynamically between NVRAM and SSD. Hot data can be permanently stored on NVRAM without writing back to SSD, while relatively cold data can be temporarily cached by NVRAM with another copy on SSD. Second, NVMFS absorbs random writes on NVRAM and employs long term data access patterns in allocating space on SSD. As a result, NVMFS experiences reduced erase overheads at SSD. Third, NVMFS explores different write policies on NVRAM and SSD. We do in-place

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

updates on NVRAM and non-overwrite on SSD. The maximum write bandwidth of SSD is exploited by transforming random writes at file system level to sequential ones at SSD level.

Flash-aware RAID Techniques

In [2] Soojun Im and Dongkun Shin proposed that solid-state disks (SSDs), which are composed of multiple NAND flash chips, are replacing hard disk drives (HDDs) in the mass storage market. The performances of SSDs are increasing due to the exploitation of parallel I/O architectures. However, reliability remains as a critical issue when designing a large-scale flash storage. For both high performance and reliability, Redundant Arrays of Inexpensive Disks (RAID) storage architecture is essential to flash memory SSD. However, the parity handling overhead for reliable storage is significant. RAID technique for flash memory SSD for reducing the parity updating cost has already been proposed. To reduce the number of write operations for the parity updates, the proposed scheme delays the parity update which must accompany each data write in the original RAID technique. In addition, by exploiting the characteristics of flash memory, the proposed scheme uses the partial parity technique to reduce the number of read operations required to calculate a parity. We evaluated the performance improvements using a RAID-5 SSD simulator.

Hybrid Parity-based Disk Array

In [3] B. Mao, H. Jiang, D. Feng, S. Wu, J. Chen, L. Zeng, and L. Tian a Hybrid Parity-based Disk Array architecture, (HPDA) combines a group of SSDs and two hard disk drives (HDDs) to improve the performance and reliability of SSD-based storage systems. In HPDA, the SSDs (data disks) and part of one HDD (parity disk) compose a RAID4 disk array. The second HDD and the free space of the parity disk are mirrored to form a RAID1-style write buffer that temporarily absorbs the small write requests and acts as a surrogate set during recovery when a disk fails. The write data is reclaimed back to the data disks during the lightly loaded or idle period of the system. Reliability analysis shows that the reliability of HPDA, in terms of MTTDL (Mean Time To Data Loss), is better than that of pure HDD-based or SSD-based disk array. Disadvantages of HPDA include that capacity will be limited by the capacity of the smallest drive and also that if the SSD side of the mirror fails, the performance levels will drop off a cliff.

Delayed partial parity scheme

In [12] Soojun Im and Dongkun Shin has proposed that the I/O performances of flash memory solid-state disks (SSDs) are increasing by exploiting parallel I/O architectures.The reliability problem is a critical issue in building a large-scale flash storage. To overcome this problem a Redundant Arrays of Inexpensive Disks (RAID) architecture which uses the delayed parity update and partial parity caching techniques for reliable and high-performance flash memory SSDs has been proposed. The proposed techniques improve the performance of the RAID-5 SSD by 38% and 30% on average in comparison to the original RAID-5 technique and the previous delayed parity update technique, respectively.

In [9] Yi Qin, Dan Feng, Jingning Liu and Wei Tong has proposed that all SSDs have to employ error correcting code (ECC) technique to ensure the reliability of flash memory at page level.The data loss may be caused by bad block or chip failure of flash memory. To solve this problem, this paper proposes a flash memory redundant array technique, which is similar to that of RAID-4. In this method, we utilize built-in NVRAM to cache the parity data update for minimal write to flash memory in parity channel.

In [18] Y. Lee, S. Jung, and Y. H. Song proposed that in the RAID technology, multi-bit burst failure in the page, block or device can be easily detected and corrected so that the reliability can be significantly enhanced. However the existing RAID-5 method for the flash-based storage has delayed response time for parity updating. To overcome this problem, we have proposed a novel approach using a RAID technique in flash storage called Flash-aware Redundancy Array. In this approach, parity updates are postponed so that they are not included in the critical path of read and write operations. Instead, they are scheduled when the device becomes idle.

Wear Levelling Scheme

Wear leveling is a technique for prolonging the service life of erasable computer storage medium, such as flash memory used in solid-state drives (SSDs) and USB flash drives. There are very few different wear leveling mechanisms used in flash memory system, each with varying levels of flash memory longevity enhancement. There are three basic types of wear leveling mechanisms used in flash memory storage devices: No wear leveling, Dynamic wear leveling and Static wear leveling.

In [16] Jianwei Liao, Fengxiang Zhang, and Guoqiang Xiao shows that wear-leveling mechanism can reduce total erasure counts and yield uniform erasure counts among all blocks at the late lifetime of the storage devices. Wear

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

leveling is an indispensable technique to even out wear caused by the writes.

In [8] Jie Fan , Song Jiang , Jiwu Shu , Long Sun and Qingda Hu Ho proposed State-of-the-art wear-leveling schemes, such as Start-Gap and Security Refresh, which start to function once even when a single block failure occurs because their designs require persistent writ able address space for wear leveling operations.

In [1] Chundong Wang and Weng-Fai Wong proposed that Operating System-Assisted Wear leveling (SAW) algorithm can significantly improve the wearevenness. SAW takes advantage of OS's knowledge about files at a higher level of abstraction, and provide useful hints to the lower-level FTL to accommodate data. A prototype based on the file system and an FTL has been developed to verify the effectiveness of SAW.

In [13] Jalil Boukhobza, Pierre Olivier and Stéphane Rubini proposed to reduce the flash memory wear out problem and improve its performance by absorbing the erase operations throughout a dual cache system replacing FTL wear leveling and garbage collection services. This idea is justified by proposing a first performance evaluation of an exclusively cache based system for embedded flash memories. Unlike wear leveling schemes, this proposed cache solution reduces the total number of erase operations reported on the media by absorbing them in the cache for workloads expressing a minimal global sequential rate.

In [15] K. Park, D.-H. Lee, Y. Woo, G. Lee, J.-H. Lee, and D.-H. Kim present reliability and performance enhancement technique on new RAID system based on SSD. First, we must analyze the existing RAID mechanism in the environment of SSD array and then develop a new RAID methodology adaptable to SSD array storage system. Via trace-driven simulation, we evaluate the performance of our new optimized SSD array storage using RAID mechanism. This proposed method enhances the reliability of SSD array 2% higher than that of existing RAID system and improves the I/O performance of SSD array 28% higher than that of existing RAID system.

Parity Cache Technique

In [10] Huh, Jungho proposed adaptive parity cache technique on cache to solve "small write problem" of RAID 5 especially in OLTP environment.This configuration provides a cache management method that adds information on the read/write characteristics of the file when the user process in RAID 5 makes a request to kernel of the file. This information is used for management of disk cache by reading the parity. On write request, this method reduces additional access problems compared to conventional methods by using write related information of the file system to store data and parity together in a cache. So, we can enhance the cache utilization and improve the disk request response time.

In [14] Akcicek D, Koyuncu S. , Sen H. and Kadayif I proposed a solution to the problem of designing a reliable data cache without trading reliability for performance and area, which is a typical characteristic of the conventional parity and ECC based protection techniques. Although parity is simple and fast, it can detect only odd number of errors without correcting any of them. The ECC techniques are more complex and time-consuming, and also have the capability of correcting some of the errors. This technique enhances data cache reliability by storing the replica of data items in active use into cache lines which hold data not likely to be reused. The information about replicas is maintained in a small fully associative cache called shadow cache. By exploiting the replicas to correct the soft errors it enhances the data reliability. Because we keep the replicas in potentially dead blocks, the performance loss is negligible with an extra chip area requirement for the shadow cache. This our technique, is more effective for enhancing the L1 data cache reliability in modern Superscalar machines with only negligible degradation in performance.

In [17] Lee J., Kim Y. , Kim J. and Shipman G. examine write cache algorithm for the array of disks, and proposes a synchronous independent write cache (SIW) algorithm. A pre-parity computation technique is also presented for the RAID of SSDs with parity computation, which calculates parities of blocks in advance before they are stored in the write cache. With the new technique, we propose a complete paradigm shift in the design of write cache. In this paper, large write requests dominant workloads show up to about 50% and 20% improvements in average response times on RAID-0 and RAID-5 respectively as compared to the state-of-the-art write cache algorithm.

In [11] Miremadi S.G. and Zarandi H.R. have analyzed the problem of transient-error recovery of several protecting techniques used in fault-tolerant cache memory. In this paper, reliability and mean-time-to-failure (MTTF) equations for several protecting techniques are derived and estimated. The result of the considered techniques are

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

compared with those of cache memories without redundancies and with only parity codes in both tag and data array of caches. Depending on the error rate under which a cache memory will operate and the size of the cache memory, we can analyze one of the cases used. If the transient error rate is very small or the size of cache memory is small, then protection with only a single parity code is adequate. But for large cache memory or for noisy environment, with high transient error rate, cache scrubbing or single error correction - double error detection technique (SEC-DED) will become essential.

In [7] Wei Zhang, proposed a cost-effective solution to enhance data reliability significantly with minimum impact on performance. The idea is to add a very small fully associative cache to store the replica of every write to the L1 data cache. Due to the data locality and its full associativity, the replication cache can be kept very small while providing replicas for a small fraction of read hits in L1, which can be used to enhance data integrity against soft errors. The replication cache with eight blocks will provide replicas for 97.3 percent of read hits in L1 on average.

Results and discussion

In [4] Chul Lee, Sung Hoon Baek and Kyu Ho Park result shows that the erase count of the NAND flash is at most one cycle over the whole erase blocks, whereas the erase count of the NOR flash is less than 50 cycles. In [5] S.H. Lim and K.H. Park shows that NVMFS experiences reduced erase overheads at SSD.The maximum write bandwidth of SSD is exploited by transforming random writes at file system level to sequential ones at SSD level. In [2] Soojun Im and Dongkun Shin shows that the proposed scheme uses the partial parity technique to reduce the number of read operations required to calculate a parity. The performance is evaluated using a RAID-5 SSD simulator.In [3] B. Mao, H. Jiang, D. Feng, S. Wu, J. Chen, L. Zeng, and L. Tian shows that capacity will be limited by the capacity of the smallest drive and also that if the SSD side of the mirror fails, the performance levels will drop off a cliff.In [12] Soojun Im and Dongkun Shin shows that the proposed techniques improve the performance of the RAID-5 SSD by 38% and 30% on average in comparison to the original RAID-5 technique and the previous delayed parity update technique, respectively.In [9] Yi Qin, Dan Feng, Jingning Liu and Wei Tong will utilize built-in NVRAM to cache the parity data update for minimal write to flash memory in parity channel.In [18] Y. Lee, S. Jung, and Y. H. Song shows that the parity updates are postponed so that they are not included in the critical path of read and write operations. Instead, they are scheduled when the device becomes idle.In [16] Jianwei Liao, Fengxiang Zhang, and Guoqiang Xiao shows that wear-leveling mechanism can reduce total erasure counts and yield uniform erasure counts among all blocks at the late lifetime of the storage devices. Wear leveling is an indispensable technique to even out wear caused by the writes.In [8] Jie Fan , Song Jiang , Jiwu Shu , Long Sun and Qingda Hu Ho shows that the State-of-the-art wear-leveling schemes, will start to function once even when a single block failure occurs because their designs require persistent writ able address space for wear leveling operations.In [1] Chundong Wang and Weng-Fai Wong uses a prototype based on the file system and an FTL that has been developed to verify the effectiveness of SAW.In [13] Jalil Boukhobza, Pierre Olivier and Stéphane Rubini proposed cache solution that reduces the total number of erase operations reported on the media by absorbing them in the cache for workloads expressing a minimal global sequential rate. In [15] K. Park, D.-H. Lee, Y. Woo, G. Lee, J.-H. Lee, and D.-H. Kim shows that the reliability of SSD array 2% higher than that of existing RAID system and improves the I/O performance of SSD array 28% higher than that of existing RAID system. In [10] Huh, Jungho reduces additional access problems compared to conventional methods by using write related information of the file system to store data and parity together in a cache. So, that we can enhance the cache utilization and improve the disk request response time. In [14] Akcicek D, Koyuncu S. , Sen H. and Kadayif I shows that the ECC technique is more effective for enhancing the L1 data cache reliability in modern Superscalar machines with only negligible degradation in performance. In [17] Lee J., Kim Y. , Kim J. and Shipman G. shows that large write requests dominant workloads show up to about 50% and 20% improvements in average response times on RAID-0 and RAID-5 respectively as compared to the state-of-the-art write cache algorithm. In [11] Miremadi S.G. and Zarandi H.R. . If the transient error rate is very small or the size of cache memory is small, then protection with only a single parity code is adequate. But for large cache memory or for noisy environment, with high transient error rate, cache scrubbing or single error correction - double error detection technique (SEC-DED) will become essential. In [7] Wei Zhang shows that the replication cache with eight blocks will provide replicas for 97.3 percent of read hits in L1 on average.

Conclusion

In this survey, it can be concluded that due to storing data in hard disk drives the power

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

consumption is high and storing parity bit in a seperate storage device in SSD causes difficulty in retriving the data if the disk crashes. To overcome this problem we must design a a partial parity cache and data cache management method for reducing the parity updating cost of a solid-state disk (SSD) based redundant array of inexpensive disk (RAID) system, so that the input/output (I/O) performance of the RAID system can be improved. This method also reduces the number of read and write operations for generating parities in the RAID system.

Acknowledgements

We take this oppurtunity to express our profound gratitude and deep regards to the Management and Principal of PSNA College of Engineering and Technology for encouraging us throughout the project.

References

1. *Chundong Wang and Weng-Fai Wong , 2013, "SAW: System-assisted wear leveling on the write endurance of NAND flash devices," in DAC 2013 50th ACM / EDAC / IEEE, pp. 1-9.*
2. *S. Im and D. Shin, 2011, "Flash-aware RAID Techniques for dependable andhigh-performance flash memory SSD," IEEE Trans. Comput., vol.60, no. 1, pp. 80–92.*
3. *B. Mao, H. Jiang, D. Feng, S. Wu, J. Chen, L. Zeng, and L. Tian, 2010,"HPDA: A hybrid parity-based disk array for enhanced performance and reliability," in Proc. IEEE IPDPS, pp. 1–12.*
4. *C. Lee, S. H. Baek, and K. H. Park, 2008, "A hybrid flash file system based on NOR and NAND flash memories for embedded devices," IEEE Trans. Comput., vol. 57, no. 7, pp. 1002–1008.*
5. *S.H. Lim and K.H. Park, 2006, "An Efficient NAND Flash File System for Flash" in IEEE Trans. Comput., vol. 55, pp. 906 - 912.*
6. *Cai, Y. , Fang L., Ratemo R. , Liu J. , Gross K. , Kozma M., 2005, "A test case for 3Gbps serial attached SCSI (SAS)" in Test Conf. 2005. Proc. ITC 2005. IEEE International, 9pp. 660.*
7. *Wei Zhang, 2005, "Replication Cache: A Small Fully Associative Cache to Improve Data Cache Reliability," in IEEE Trans. Comput , vol. 54, pp. 1547-1555.*
8. *Jie Fan, Song Jiang , Jiwu Shu and Long Sun, 2014, "WL-Reviver: A Framework for Reviving any Wear-Leveling Techniques in the Face of Failures on Phase Change Memory," in DSN 2014 44th Annual IEEE/IFIP International Conf., pp. 228 – 239.*
9. *Yi Qin, Dan Feng, Jingning Liu, Wei Tong, Yang Hu and Zhiming Zhu, 2012, "A Parity Scheme to Enhance Reliability for SSDs" in Networking, Architecture and Storage (NAS), 2012 IEEE 7th International Conf., pp. 293-297.*
10. *Huh and Jungho, 2005, "Design and performance evaluation of an adaptive parity cache technique," in Systems Engineering 2005 ICSEng 2005. 18th International Conf., pp. 58-63.*
11. *Miremadi S.G. and Zarandi H.R., 2005, "Reliability of protecting techniques used in fault-tolerant cache memories," in Electrical and Computer Engineering 2005 Canadian Conf., pp. 820-823.*
12. *Soojun Im and Dongkun Shin, 2010, "Delayed partial parity scheme for reliable and high-performance flash memory SSD," in Mass Storage Systems and Technologies (MSST) 2010 IEEE 26th Symposium, pp. 1-6.*
13. *Boukhobza J. , Olivier P. and Rubini S., 2011, "A Cache Management Strategy to Replace Wear Leveling Techniques for Embedded Flash Memory," in SPECTS 2011 International Symposium, pp. 1-8.*
14. *Akcicek D., Koyuncu S., Sen H. and Kadayif I., 2007, "Exploiting potentially dead blocks for improving data cache reliability against soft errors," in Proc. ISCIS 2007 22nd international symposium, pps. 1-6.*
15. *K. Park, D.-H. Lee, Y. Woo, G. Lee, J.-H. Lee, and D.-H. Kim, 2009, "Reliability and performance enhancement technique for SSD array storage system using RAID mechanism," in Proc. ISCIT, pp. 140–145.*
16. *Liao J. , Zheng F. , Li L. and Xia G., 2014, "Adaptive Wear-leveling in Flash-based Memory," in Computer Architecture Letters, pp. 1.*
17. *Lee J., Kim Y., Kim J. and Shipman G., 2014, "Synchronous I/O Scheduling of Independent Write Caches for an Array of SSDs," in Computer Architecture Letters, pp. 1.*
18. *Y. Lee, S. Jung, and Y. H. Song, 2009, "FRA: A flash-aware redundancy Array of flash storage devices," in Proc. Hardw./Softw. Codesign Syst. Synthesis, pp. 163–172.*