# Report on Workshop on Operating Systems Support for Next Generation Large Scale NVRAM (NVRAMOS 2009)

Sang-Won Lee Sooyong Kang Youjip Won Jongmoo Choi

Sungkyunkwan University
Suwon, Korea
wonlee@ece.skku.ac.kr

Hanyang University
Seoul, Korea
{sykang, yjwon}@hanyang.ac.kr

Dankook University
Yongin, Korea
choijm@dku.edu

## 1. MOTIVATIONS

NAND Flash memory solid state disk (hereafter SSD) technology is advancing rapidly in capacity and speed. Also, we envision that byte-addressable NVRAM devices, which is being thought as the future replacement of Flash memory technology will be commercially viable for storage within the next 3–7 years. Modern computer system takes hierarchical organization. It consists of CPU, Main Memory and Storage. This hierarchical organization is natural outcome of economic concern. Modern Operating System is designed to effectively exploit the physical characteristics of the device in each system hierarchy. Process management, memory management, and storage management subsystems are all designed to fill the gaps (speed and space) among the layers in different hierarchies while maximizing cost performance ratio. Current operating system paradigm draws clear line between memory and storage and handles them in very different way. Large scale byte-addressable NVRAM combines the physical characteristics of main memory with the non-volatility of storage, presenting new challenges for system designers. From the DBMS and Operating System's point of view, the advancement of this device calls for a reconsideration of legacy operating system architectures that have been designed based upon the notion of system hierarchy: CPU, memory and storage.

NVRAMOS Workshop (Workshop on Systems Support for large Scale NVRAM) is biannual event which invites leading experts from academia as well as industry in the area of large scale NVRAM, Flash Storage, Storage Class Memory, NVRAM device technology and related areas. The workshop specifically focuses on Operating Systems aspect of exploiting new semiconductor technology (FLASH, PRAM, MRAM, FRAM, Solid Electrolyte) as storage device or memory/storage device. The primary goal of this workshop is to form a community in this field and provide a forum where the experts in this area can put forth their vision, share views, exchange ideas and have intensive discussions on the technical challenges we may face (or are facing already) in the next several years. The first NVRAMOS workshop was held in Nov. 2007. NVRAMOS workshop now successfully positions itself as prime forum which achieves the original goal. NVRAMOS workshop has been two day event and has been allocating extensive time slot for each presentation so that speaker and the workshop participants can have interactive and intense technical discussion about the talk theme in in-depth manner.

## 2. WORKSHOP OUTLINE

NVRAMOS 2009 Spring workshop (http://www.dmclab.hanyang.ac.kr/nvramos09) consists of three technical sessions: *DAMO, invited talks and peer-reviewed paper presentations*. Different from past three NVRAMOS Workshops, workshop committee made new session which is dedicated for "casual and informal" discussion on cutting edge technical themes. It is called DAMO session.

### 2.1 DAMO Session

The DAMO session is a kind of *gong show or Birds-Of-a-Feather (BoF) session* to warm up the workshop and to attract the participants' attention. Each speaker in DAMO session is allocated 20 minute time slot. The presenter first gives a quick review of his/her Flash research field, what they are working on, and/or presents his/her view on future research direction and outstanding problems.

The first presentation, by Bumsoo Kim (CEO, Indilinx Co[1]., Korea), reviewed the SSD development history from the 1st generation to the current 3rd generation, predicted that the 4th generation SSD will appear in the market in one or two years. In particular, he predicted that the enterprise market (including OLTP (online transaction processing) applications) would be more promising than PC or laptop market because of the urgent IOPS (IO per seconds) requirements.

The second presentation, by Tae-Sun Chung (Ajou University, Korea), gave a survey on Flash translation layers (called *FTL*), and argued that database systems should be aware of the software stacks from database

[1] it is known for as a controller provider of the OCZ Vertex SSDs

module, file systems down to the lowest FTL layer for better performance.

The third presentation, Sang-Won Lee (Sungkyunkwan University), explained the importance of access density (i.e. IOPS per GB) in OLTP applications and argued that the intrinsic high access density makes OLTP applications one of the most promising killer applications for SSD. In addition, he pointed out that the Flash-aware buffer management policy needs to be investigated because of the asymmetric speed of read and write operations in Flash memory SSD. Another important issue pointed out is the performance variations in transaction throughput when SSD is used as storage device. In particular, depending on the adopted FTL techniques, a commercial SSD shows high fluctuations in performance while the other SSD does relatively uniform throughput. He finished his talk saying that one of the virtues of the enterprise storage devices is the predictable performance and it should be considered in designing FTL techniques.

The fourth presenter, Jaehyuk Cha (Hanyang University, Korea), stressed the importance of SSD benchmarks. Even though most SSD support the same host interface (e.g. SATA) with hard disks, there are several characteristics to make SSDs distinguishable from hard disks, and also there are diverse design spaces which can affect the performance characteristics of SSD. For this reason, the existing IO benchmarks developed mainly for comparing the hard disks are too plain to reveal the intrinsic performance differences among various SSDs. He emphasized that it is critical to develop new IO benchmarks tailored for SSD. The presenter recommended the uFlip benchmark [2] as a good starting point for further micro benchmark developments which can help expose the performance characteristics between different SSDs.

The last presentation, by Sooyong Kang (Hanyang University, Korea), discussed the write buffer cache issue on SSD controller, which FTL can exploit to improve the write performance. In particular, he raised a question of *whether it is possible to find an optimal write buffer replacement scheme*. Further, when NVRAM (for example, PRAM, FeRAM, and MRAM, which are expected to be commercially available in the market) is used as write buffer. Two specific issues are discussed: 1) whether there exists an optimal off-line buffer replacement algorithm?, and 2) if so, what parameters should be considered for algorithm evaluation? For the second issue, the presenter suggested that the status of log blocks managed by FTL and temporal locality in access patterns as well as the size of page clusters (i.e. basic unit of replacement) should be taken into account. The presentation concluded that NVRAM buffer and the traditional log blocks in FTL should be managed as an integrated write buffer cache.

## 2.2 Invited Presentations

The NVRAMOS 2009 Spring invited three speakers from both industry and academia. The talks are on SSD simulator, Flash memory-based database systems, and SSD performance model.

The first speaker, Eui-Young Chung (Yonsei University, Korea), introduced their effort to develop a SSD simulator which is based on the Cycle-Accurate Model. The simulator not only explored the internal architecture of SSD, but also considered both the H/W (Channel/Way, DRAM cache management, Hybrid Flash memory arrays such as MLC+SLC combination, etc.) and S/W (FTL) features of SSD, simultaneously. During the discussion, many audience members suggested to conduct the simulation using various access patterns that database systems incur and compare the results with those of the previous works about SSD performance in the database systems, which can help not only evaluating the accuracy of the black-box approach but also finding out the bottleneck point inside SSD when it is used in the database system.

The second speaker, Bongki Moon (University of Arizona, USA), presented challenges and opportunities of Flash memory database systems. While the current SSD products still have a random scattered writes issue, Moon said that the SSD can provide a few immediate benefits for some DB operations: 1) reduce commit-time delay by fast logging, 2) reduce read time for multi-versioned data, and 3) Flash-friendly I/O patterns in temporary table spaces. To remedy the random scattered writes issue, he said that besides industries' efforts to increase the random write performance from storage systems' viewpoint, a great deal of efforts need to be also solicited from database systems' point of view. He briefly introduced In-Page Logging (IPL) approach [1] as an example. During the discussion, some audience members argued that two key assumptions in the IPL approach, 1) multiple writes are possible within a page, and 2) non-sequential page writes in a data block, are not valid any more. A participant from from industry added that these assumptions are difficult to accept not only in the current SSDs but (probably) also in the future SSDs, because the characteristics of Flash memory chip (e.g. the allowed number of partial writes per page), are getting worse as its capacity increases.

The last invited speaker, Dongjun Shin (Samsung Electronics), introduced an interesting SSD performance model, which is based on the well-known RISC instruction pipelining performance model. He argued that, because there is parallelism between computation and I/O in SSD, SSD operations can also be pipelined, which led his team to revisit the old model. Based on their experiment, the accuracy of the model was proven to be quite reliable. In modern SSDs, micro-controller issues I/O instructions in pipelined fashion, and there can be multiple outstanding instructions each of which accesses different NAND Flash

chips within SSD. Using the simulation model, they found an interesting fact that firmware overhead, which has been regarded as negligible, becomes rather significant especially when microcontroller of SSD works in deeply pipelined fashion. Their model showed that the performance of random I/O operation in SSD critically relies on the firmware efficiency and is currently being bound by firmware overhead. The audience agreed that it would be very challenging to design application-specific SSDs (e.g. DBMS-aware SSD); especially when we are to migrate part of functionalities of database's storage manager down to SSD controller, which generally increases the complexity of firmware.

## 2.3 Paper Presentations

Four accepted papers have been presented in the paper session. Two of them deal with the characteristics and performance issues of SSD, while the others explore the impacts of Storage Class Memory (hereafter SCM) on file systems, databases, and operating systems. SCM, also known as Non-volatile RAM (NVRAM) such as Ferroelectric RAM (FeRAM), Magnetoresistive RAM (MRAM), and Phase-change RAM (PRAM) has characteristics of both random accessibility and non-volatility [5]. It can be used not only as main memory for dynamic program execution but also as secondary storage for file systems.

The first presentation, by Youjip Won and Minsuk Choi (Hanyang University, Korea), evaluated the performance characteristics of various SSDs under a diversity of file systems and I/O workloads. Recently, SSDs have started to replace hard disks in laptop computer and even in server storage [3]. These trends raise an interesting question: "Are the existing file systems and I/O workloads primarily optimized for hard disks still valid for SSDs?" To answer the question, the authors conducted a quantitative analysis: 1) using three different SSDs, namely Intel X25-M, Samsung MMCRE64G5MPP, and Mtron MSD-SATA3525, and 2) mounting three different file systems, namely EXT3, REISER, XFS, NILFS, and FAT32 on the SSDs, and 3) applying four different I/O workloads with various file/record sizes, and 4) measuring throughput for each combinational case. Experimental results showed that SSD performance is very sensitive to I/O workloads and file system, and there is much room to develop efficient SSD-aware algorithms both for file systems and FTLs. The results triggered active discussions among the audience and some strange results were clarified, which were derived from the special features of FTL in SSDs and/or from the anomalous behaviors of file systems in Linux. Also, the audience agreed that the characteristics of SSDs are quite different from those of hard disks, which requires new studies for file systems and I/O workloads appropriate to SSDs.

The second presentation, by Junkil Ryu and Chanik Park (POSTECH, Korea), devised a black box approach to investigate the internal structures and algorithms inside SSDs. The performance of SSD heavily depends on the internal structures such as bus interleaving, chip interleaving, and the size of DRAM write buffer, and the FTL algorithms, e.g. mapping scheme and a garbage collection policy [4]. By applying the devised black-box approach, the authors retrieved an important performance parameter, called SSD management block, from commercially available SSD products, which can then be used to optimally configure OS policies in buffer cache and prefetching. During the discussion, the distinction and similarity of performance model between hard disks and SSDs were elaborated. Some participants described that the degree of parallelism and garbage collection policy are the most critical parameters in SSD as the seek time in hard disks.

The third presentation, by In Hwan Doh, Young Jin Kim, Eunsam Kim, Sam H. Noh (Hongik University), and Donghee Lee (University of Seoul), exploited Storage Class Memory as a reliable write buffer and analyzed the impacts of SCM on the performance and dependability of computer systems. With the SCM write buffer, the authors enhanced system reliability by providing transactional supports and improved system performance by applying delayed writes and reducing write requests to the disks. The audience suggested various ideas about how to make use of SCM to resolve the traditional problems of databases and file systems such as small random writes handling, intent logging and recovery, and data-intensive application supports.

The final presentation, by Seungjae Baek and Jongmoo Choi (Dankook University), examined how to apply SCM on embedded systems. They conjectured that there are three possible organizations in using SCM in the legacy computer organization, namely SCM as storage, SCM as main memory, and SCM as both. Their conjecture is in accordance with those suggested by M. K. Qureshi et al. [6], but the former mainly focused on the SCM as both organization, while the latter concentrating on the SCM as main memory one. The authors proposed a novel operating system for SCM as both organizations, which contain a new SCM manager, supporting not only file interfaces but also memory interfaces, simultaneously. In other words, file objects and memory objects are co-located in SCM, which makes it feasible to exchange between the objects without copy overheads. Some of the audiences recommended other possible organizations such as hybrid storage with SCM and SSD, and discussed tradeoffs among them in terms of energy, cost, and performance.

## 3. WORKSHOP CONCLUSIONS

Many participants agreed that the research on Flash memory SSD and its related system software technologies is one of the main-stream topics in current computer science field. Some participants carefully envision that SCM will be a next wave for storage when the large scale manufacturing technology is possible.

### 3.1 Implications on Database Researches

From the workshop presentations and discussions, we can learn several implications which Flash memory SSD and next generation NVRAM can have on database fields, and we summarize them in the followings:

First of all, despite the tardy "erase-before-overwrite" characteristics of Flash memory, the modern third generation high-end SSDs, relentlessly and drastically improved in the last few years and now are fast enough to compete with the enterprise class hard disks in terms of performance and even the cost effectiveness in OLTP applications. Moreover, among many potential applications for SSD, online transaction processing (OLTP) is considered as one of the most promising ones because its IO patterns are mainly comprised of random reads and writes, for which SSDs can considerably outperform hard disks. In particular, the database community has two popular macro benchmarks, TPC-C and TPC-H, both of which are very IO-intensive. In addition, the IO patterns from the benchmarks are further complicated by concurrently executing processes. Therefore, they could be good testbeds for comparing different SSDs. These macro benchmarks, in combination with micro benchmarks such as uFlip, could be more realistic than the traditional IO benchmarks for hard disks.

Second, if we take into account the rapid growth of SSD market, it is time to revisit all the major IO-related database techniques, including storage layout, join algorithms, buffer management, and index technology, from the perspectives of SSD, because most of them has assumed the hard disks as its secondary storage [7]. For example, while a certain algorithm could choose sequential scans when it is expected to work on hard disks, index-based version of the algorithm could be much better on SSDs. One thing to note is that the availability of fast SSD does not mean that database side optimizations such as in-page logging [1] are not valid or necessary any more. Instead, there would be many opportunities from database communities. In fact, since 2008, many diverse works on Flash-based DBMSs are appearing in major database conferences and workshops, including SIGMOD, VLDB, ICDE and Damon.

Finally, there are a few opportunities for DBMS architectural changes to properly exploit moderate size NVRAM (e.g. of several tens or hundreds megabytes). One example is to write the log tail directly and persistently in SCM, instead of writing the log buffer in DRAM and flushing it to the hard disk or Flash memory whenever every commit command is issued. By doing this, the response time and throughput can be improved. Another example is to exploit the NVRAM inside SSD controller. The FTL module needs to keep the mapping information between logical and physical address either in the unit of block or page. When the mapping information is changed, the update should be persistently stored in Flash memory. This can be a performance bottleneck in SSD. However, this overhead can be significantly relegated if we manage the mapping information (usually less than hundred megabytes) in NVRAM.

## 4. ACKNOWLEDGMENTS

NVRAMOS Workshops since its inception in Nov. 2007 have been sponsored by National Research Lab Grant(ROA 2007-000-20114-0) from Korean Science and Engineering Foundation(KOSEF).

## 5. REFERENCES

[1] Sang-Won Lee and Bongki Moon, Design of FlashFlash-based DBMS: An In-Page Logging Approach, Proceedings of the ACM SIGMOD conference, June, 2007

[2] Luc Bouganim, Bjorn Tor Jonsson, and Philippe Bonnet uFLIP: Understanding FlashFlash IO Patterns, CIDR 2009, January, 2009

[3] D. Narayanan, E. Thereska, A. Donnelly, S. Elinikety, and A. Rowstron, "Migrating Server Storage to SSDs: Analysis of Tradeoffs", Proceeding of the ACM EuroSys Conference, March 31-April 3, 2009.

[4] N. Agrawal, V. Prabhakaran, T. Wobber, J. D. Davis, M. Manasse, and R. Panigrahy, "Design Tradeoffs for SSD Performance", Proceedings of the 2008 USENIX Annual Technical Conference, pages 57–70, June 22-27, 2008.

[5] G. W. Burr, B. N. Kurdi, J. C. Scott, C. H. Lam, K. Gopalakrishnan, and R. S. Shenoy. "Overview of Candidate Device Technologies for Storage-Class Memory", IBM Journal of Research and Development, 52(4):449–464, 2008.

[6] M. K. Qureshi, V. Srinivasan, and J. A. Rivers, "Scalable High Performance Main Memory System Using Phase-Change Memory Technology", Proceedings of the 36th International Symposium on Computer Architecture (ISCA 2009), June 20-24, 2009.

[7] Sang-Won Lee and Won Kim, On Flash-based DBMS: Issues for Architectural Re-examination, Journal of Object Technology, September, 2007