Report on the First International Workshop on Flash-Based Database Systems (FlashDB 2011)

Xiaofeng Meng[†], Peiquan Jin[‡], Wei Cao[†], Lihua Yue[‡]

[†] School of Information, Renmin University of China, Beijing, China
[‡] University of Science and Technology of China, Hefei, China

1. INTRODUCTION

Recently, new storage media such as flash memory have been developed very quickly, which brings big challenges to the architecture of computer systems as well as the design of system software. In particular, NAND flash (either SLC- or MLC-based) in the form of solid state disks (SSDs) has been an alternative to traditional magnetic disks, both in the home-user environment and in the enterprise computing environment, due to its shock-resistance, low power consumption, non-volatile, and high I/O speed [1]. The special features of flash memory and other new storage media impose new challenges to traditional data management technologies. As a result, traditional database architectures and algorithms designed for magnetic-disk-based storage fail to utilize new storage media efficiently. Meanwhile, the new characteristics of modern storage media, such as not-in-place update and asymmetric read/write/erase latencies of flash memory, also bring great challenges in optimizing database performance, by using new querying algorithms [2], indexes [3], buffer management schemes [4], and new transaction processing protocols. Consequently, exploiting the characteristics of flash memory and other new storage media has become an important topic of database systems research.

In order to make database systems adapt automatically to the characteristics of flash memory and other new storage media, the data management community needs to rethink traditional underlying storage architecture, query processing algorithms, indexing mechanism, buffer management schemes as well as many traditional issues in magnetic-disk-oriented database systems to adapt to the advances in the underlying storage infrastructure.

The First International Workshop on Flash-based Database Systems (FlashDB 2011) was held in conjunction with DASFAA 2011 in Hong Kong on April 22. This full-day event brought together researchers and engineers from academia and industry to discuss and exchange ideas related to flash-based database technologies. The workshop features three invited talks and two research sessions. This summary report gives a concise view of the three invited talks, as well as the novel ideas presented and discussed at the workshop. We hope this report will help the community by conveying the inspiring ideas and topics which form the frontier of this research area.

The workshop began with the opening speech given by Xiaofeng Meng, professor from Renmin University of China. This workshop was co-organized by researchers from Renmin University, University of Science and Technology of China, and Hong Kong Baptist University, and attracted up to twenty attendees from Korea, Germany, France, mainland China, and Hong Kong China in academia as well as in industry. Flash memory has been growing as a new type of storage media with its advantages such as faster IO speed, lower power consumption, better shock resistance etc. compared to magnetic disks. At present, it is still an open issue to utilize the advantages of flash memory to achieve better system performance and higher energy efficiency in current database systems. This workshop aims at serving as a platform to share and exchange ideas, to work together to address flash-based database related problems, and to nurture inspirations of new solutions in this area.

2. INVITED TALKS

The first invited talk was given by Sang-Won Lee, professor at Sungkyunkwan University. The talk was titled as “Some Research Directions in FlashDB” and had four topics. Firstly, the speaker reflected on transactional in-page logging (TIPL) for multi-version read consistency and recovery, the transactional support of in-page logging (IPL[5]) design on NAND flash memory that employs out-of-place update and fast read speed of flash memory. TIPL takes advantage of redo logs dwelt within blocks offering multi-version store and new recovery schemes with nominal overhead. Performance evaluation from event-driven simulators of TPCC traces of multi-version read consistency and fast recovery shows the effectiveness of TIPL. The second topic began with threats and opportunities IPL design faced with the development of flash memory and the emergence of PRAM (i.e., Phase Change Memory, PCM). With this vision, IPL-P (IPL with PRAM) was proposed as a hybrid storage design based on flash memory and PRAM, to keep page-oriented logs on PRAM to utilize the better small-sized-write efficiency of PRAM. IPL-P

outperforms flash-only design in simulated evaluation and in real board evaluation outperforms both flash-only and PRAM-only designs for insert and update operations. In the third topic, Prof. Lee discussed the design consideration of FlashCache, which uses flash-based SSD as extended buffer cache of RAM in the hybrid storage architecture involving both HDD and SSD. He also presented the improved performance when running the TPC-C benchmark on PostgreSQL. Finally, the speaker retrospected the concept of DB machines in the light of SSDs. The breakthrough in flash read interface and the parallelism inside SSDs, data-intensive computing and "bandwidth crisis" confront the host CPU with more burdens. Inspired by current SoC technology such as hardware-based ISP, Prof. Lee rethinks realizing some database computations, e.g. scans, aggregation, joins, and sorting, on embedded CPUs of storage devices to offload the host CPU, a shift from "bring data to computation" to "bring computation to data".

The second invited talk was given by Theo Härder, professor at the University of Kaiserslautern. The title of his presentation was "Energy Efficiency is not Enough, Energy Proportionality is Needed!"[6]. His talk included four parts. The first part described the characteristics of flash memory and SSDs and showed the differences of different SSD types on the basis of empirical experimental results. A number of issues were explored including whether SSDs suffer from random access, whether SSDs exhibit unstable and fluctuating behavior, whether read/write asymmetry is as bad as commonly expected, whether overwriting of blocks on a full disk is much slower than writing to an empty disk, and whether queue depth has an impact on performance. Energy-consuming experiments have revealed that different SSDs have different power profiles and that the power consumption for idle states and peak loads is considerably lower than for HDD. A critical question concerning energy consumption is whether energy efficiency and energy proportionality observed at the SSD device level can be also expected at the system level. For this reason, the second part compared disk- and SSD-based DBMS buffer management methods, such as CFDC, CFLRU, LRU, LRU-WSR and REF. The CFDC algorithm was generally superior to its competitor algorithms w. r. t. performance and energy efficiency. However, the energy consumption of ATX-, IDE- with a SATA-based disk at the different processing states such as idle, working and peak, revealed that these components are not energy proportional to system utilization. In the third part, Prof. Härder analyzed the relationship between the power use and the system utilization, including CPU, hard disk and SSD, and discussed how energy-proportional computing could be achieved. Ideal energy-proportional computing should consume no energy in idle states; power consumption should linearly increase with system utilization and approach full power usage (100%) in peak load situations (100% system utilization). However, current computer systems are not energy proportional at all, because major components (main memory, parts of ATX, etc.) consume an almost constant amount of energy independent of the degree of system utilization. Therefore, the entire system reaches at idle states often already more than 50% (in case of a very large main memory close to 100%) of the energy consumption needed for peak load. The last part introduced a research project aiming at energy-proportional computing in the context of DBMS use. The system called WattDB uses an architecture where the powerful DB server machine is replaced by a cluster of wimpy shared-nothing computing nodes and some shared-disk storage nodes. By activating the processing nodes on demand, power consumption of the entire system can be decreased to a minimum level and, thus, energy proportionality can be approximated. Starting with a single node in the cluster, additional nodes can be activated on demand without interrupting DB processing. In this way, the cluster is able to scale up to n nodes and smoothly grow and shrink, so WattDB can stepwise approximate an ideal energy-proportional behavior. Each of the individual computing nodes is able to access the entire database via storage nodes. Because of dynamic node fluctuation, frequent DB cluster coordination is necessary to optimally support DB processing and maintenance as well as concurrency control and logging/recovery, DB partitioning, etc. Methods for flexible physiological DB partitioning have to be developed to successfully reach the research objectives of WattDB. Finally, Prof. Härder stated that "In the future, WattDB will be specialized towards differing directions to provide tailor-made support for the application classes OLTP, OLAP, and MapReduce".

Jianliang Xu, professor at Hong Kong Baptist University, was the third invited speaker. In this talk entitled "Flash-based Database Systems: Some Experiences from the FlashDB Project", Prof. Xu first introduced the FlashDB project, an NSFC key project collaboratively carried out by three institutions in mainland China and Hong Kong. The goal of the project is to investigate new architectures and methods to boost database performance, by exploiting unique flash I/O characteristics. The speaker exemplified three case studies towards this goal. In the first case study, DigestJoin is a two-phase flash based join processing method that makes good use of random reads on flash memory devices and reduces writing of intermediate join results [7]. In the first phase, digest tables in the form of <join_key, tuple_id> are generated and then joined. In the second phase, based on the digest join

results, full tuples are reloaded to form the final join results. The second case study addressed a new approach for write performance optimization [8]. Based on the observations that sequential, focused, and partitioned writes are more efficient than general random ones on flash-based storage devices, by setting up a small sized (e.g. 1–16 MB) stable buffer on flash devices, a general random write can be transformed into a focused write to the stable buffer and an efficient flush of pages from the stable buffer to the destination. And the third case study presented that the shadow paging technique well suits out-of-place updates of flash memory devices. Combined with the partial page programming feature in SLC flash memory, a novel flag commit idea was discussed to support transaction recovery. Two specific protocols, Commit-based Flag Commit (CFC) and Abort-based Flag Commit (AFC), were designed to support normal transaction processing, commit/abort, garbage collection, and recovery.

3. RESEARCH SESSIONS

3.1 Session A: Storage Management for SSD

The fundamental thing to successfully adopt flash devices in database systems is a storage design that is based on the specific storage and access characteristics of flash memory or flash devices. This session features four research papers addressing this problem from different aspects.

The paper entitled Page-Level Log Mapping: From Many-to-Many Mapping to One-to-One Mapping addressed the logical-to-physical page mapping issue in flash-based systems. The authors designed a page-level log mapping method called PLM, which uses backward link technique to support efficient reads and writes, and therefore can yield optimal overall performance. Besides, the authors developed two implementations of PLM incorporating flash-optimized strategies for buffer management, free page allocation and garbage collection. Finally, the proposed algorithm achieved high efficiency across a series of experiments.

In the paper entitled A Novel Method to Extend Memory Lifetime in Flash-based DBMS the authors first analyzed the previous methods for free space management in conventional DBMSs, such as free list and space map, and pointed out that those traditional approaches are not suitable for flash-based database systems. Therefore, the authors proposed to use an Append-Only (AO) scheme to maintain the free space in DBMS. The AO scheme allocates new empty pages as soon as a write request comes and appends it to the tail of the original database files, which avoids useless searching for a page with free space. Furthermore, in order to reduce the number of small write and random write, a stand-alone write buffer was also proposed to collect the inserted and updated records. The experiment result based on a flexible emulator showed this approach enjoys a 74.5% page write decrease.

The paper entitled Log-Compact R-Tree: an Efficient Spatial Index for SSD proposed a novel flash-aware variant of R-Tree, named LCR-Tree, which records the updates of R-Tree as logs to transfer random writes to sequential ones. Distinguished from previous attempts, compacted log was introduced to combine newly arrival logs with the original ones on the same node, which renders great decrement of random writes with at most one additional read for each node access. In this way, although more read overhead is invoked, the write performance is improved significantly. The experimental results on both synthetic and real data sets showed that the LCR-Tree can achieve up to 3X gains over RFTL, an existing flash-based index scheme, and the R-tree

The paper entitled An FTL- agnostic Layer to Improve Random Write on Flash Memory proposed a data placement algorithm specially designed for flash memory to improve the efficiency of random writes. In this paper, the authors first claimed that there is a strong correlation between write performances and spatial locality for FTL-based flash devices, and defined a distance between logical pages to reflect this effect. Based on the concept of page distance, the authors proposed a simple data placement algorithm which aims at transforming random writes into quasi-sequential access patterns trades. The efficiency of such a mechanism was validated by a formal mathematical model. In the experiment, the proposed method improved the random write performance by up to two orders of magnitude.

3.2 Session B: Energy Efficiency & Hybrid Storage

Energy efficiency is one of the key merits of flash memory. How to design a system that keeps high performance while saving energy is a challenging problem. The first presented paper addressed this issue. The other two presented papers discussed hybrid storage architecture of NAND flash and PRAM memory on mobile devices and hybrid storage of HDD and SSD.

The paper entitled Trading Memory for Performance and Energy mainly addressed the problem of tradeoff between performance and power consumption when managing extremely large amounts of data. From the standpoint of architecture, a three-layer database storage system was designed and implemented for reducing the power consumption. The prototype uses flash-based devices as an intermediate caching layer. The memory and disk layers are basically the same as those in the classical two-layer disk-based storage

systems. To manage the flash layer, two algorithms, namely the Local (LOC) algorithm and the Global (GLB) algorithm, were presented as the replacement policy. Both experiments on synthetic and real-life traces were conducted to measure the overall performance and energy efficiency. The results showed that flash-based layer significantly improves the I/O efficiency and then reduces the use of energy-inefficient RAM-based memory without compromising the overall system performance.

The paper entitled Design of embedded database based on hybrid storage of PRAM and NAND flash memory studied the problems of database systems on mobile devices with a single storage media – NAND flash and a single file system, YAFFS2. To overcome the inefficiency of small-sized data read/write operations and frequent updates, hybrid storage architecture of PRAM and NAND flash memory was proposed to take advantage of the specific properties of PRAM, i.e., byte addressability and in-place updates. The proposed architecture replaces NOR flash memory in the conventional architecture by using PRAM memory as boot-up code storage as well as a data storage. Such a hybrid system was implemented on the basis of SQLite and dual file systems (YAFFS2 and PRAMFS). Particularly, the rollback journals of SQLite are stored on PRAM via the file system PRAMFS, while the database files are stored on NAND flash memory through the file system YAFFS2. Evaluation on board with NAND flash and PRAM emulated by UtRAM [9] with software delay showed the proposed architecture reduces the transaction time by 45% compared with systems equipped with only NAND flash memory.

In the paper entitled Hybrid Storage with Disk Based Write Cache, the authors proposed using HDDs as the write cache for SSDs to exert the better sequential write performance of HDDs while avoiding random writes on SSDs. In this hybrid storage architecture, pages are read from both HDD and SSD, while updated pages are all written to HDD once evicted from buffer. To take advantage of the high read speed of SSDs, the authors presented an approach to migrating read-mostly pages into SSD, in case that they are first located in HDD. Those migrated pages are organized as blocks and all migrations are performed according to a block unit, which aims at making use of the high sequential-write performance of SSD and also reducing the erase times of flash memory. Experiments were performed on several synthetic traces, and the results showed that the hybrid scheme ensures most read operations are performed on SSD and most write operations are focused on HDD. Meanwhile, it has less runtime than the single-disk-based mechanism.

4. DISCUSSIONS

At the end of the workshop, some open questions were identified by Prof. Jianliang Xu in his presentation concerning some potential research directions, which are summarized as follows:

(a) What are the main challenges and issues for flash-based enterprise database applications? While SSDs have been adopted as an alternative storage for enterprise database applications, architectures, data structures, and algorithms optimized for such applications should be developed in accordance with the performance objective such as information access speed, energy efficiency, or even endurance of flash devices.

(b) Which storage hierarchy will prevail in the future? With the advent of flash memory and the coexistence of magnetic disks, will flash memory serve as an extension of main memory or an extension of magnetic disks? Many schemes for hybrid storage have been proposed; but which one will prevail in the future remains to be seen.

(c) What is the impact of new NVRAM storage technologies such as Phase Change Memory (PCM)? PCM emerges with better I/O bandwidth, longer write endurance, bit-alterability, and byte-addressability, compared to flash memory. How to utilize the advantageous features of NVRAM to complement current storage systems is a very promising and interesting research problem.

5. ACKNOWLEDGEMENTS

This work was supported by the National Natural Science Foundation of China under the grant No. 60833005 and No. 61073039.

6. REFERENCES

[1] J. Gray and B. Fitzgerald, Flash Disk Opportunity for Server Applications, ACM Queue, vol. 6(4), pp.18–23, 2008.

[2] D. Tsirogiannis, S. Harizopoulos, M. Shah, J. Wiener, Goetz Graefe. Query Processing Techniques for Solid State Drives, In Proc. of SIGMOD'09, pp. 59-72, 2009.

[3] S. Yin, P. Pucheral, X. Meng, PBFilter: Indexing Flash-Resident Data Through Partitioned Summaries, In Proc. of CIKM'08, pp. 1333-1334, 2008.

[4] Z. Li, P. Jin, X. Su, K. Cui, L. Yue, CCF-LRU: A New Buffer Replacement Algorithm for Flash Memory, IEEE Trans. on Consumer Electronics, Vol.55(3), pp.1351-1359, 2009.

[5] S. –W. Lee, B. Moon, Design of flash-based DBMS: an In-Page Logging Approach, In Proc. of SIGMOD'07, pp. 55-66, 2007.

[6] T. Härder, V. Hudlet, Y. Ou, D. Schall, Energy Efficiency is not Enough, Energy Proportionality is Needed!, In Proc. of DASFAA'11 Workshops, LNCS 6637, pp. 226-239, 2011.

[7] Y. Li, S. T. On, J. Xu, B. Choi, H. Hu, DigestJoin: Exploiting Fast Random Reads for Flash-based Joins, In Proc. of MDM'09, pp. 152-161, 2009
[8] Y. Li, J. Xu, B. Choi, H. Hu, StableBuffer : Optimizing Write Performance for DBMS Applications on Flash Devices, In Proc. of CIKM'10, pp. 339-348, 2010
[9] Y. Park, S.-H. Lim, C. Lee, K. H. Park, PFFS: A Scalable Flash Memory File System for the Hybrid Architecture of Phase-Change RAM, In Proc. of SAC '08, pp. 1498–1503, 2008