Characterizing and Modeling Non-Volatile Memory Systems[1]

Zixuan Wang Xiao Liu Jian Yang* Theodore Michailidis Steven Swanson Jishen Zhao
University of California, San Diego

I. Introduction

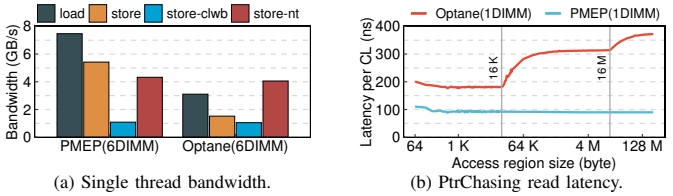
A. The Problem

Intel recently released the first server-grade nonvolatile memory (NVRAM) DIMM product, Optane DC persistent memory (aka. Optane DIMM) [2]. The new NVRAM device introduces numerous research opportunities on exploiting NVRAM in future computing systems, due to the performance discrepancy identified by preliminary profiling [3]. Yet, corresponding research efforts are significantly constrained by the limited accessibility of the new device to the broad research community.

Performance Discrepancy. Many previous NVRAM studies perform experiments using NVRAM emulators and simulators, which assume that NVRAM performs like a slower DRAM. For instance, widely-used NVRAM emulators PMEP [1] and Quartz [6] model NVRAM systems by stalling the CPU for additional cycles. However, Optane DIMM characterization [3], [7] demonstrates that real NVRAM systems have much more complex performance behaviors than conventional DRAM systems. As an example, Figure 1 shows our experiments that illustrate two observations on the performance discrepancy. First, Figure 1a shows that PMEP models both load and store bandwidth higher than the bandwidth of non-temporal stores (*store-nt*). However, with Optane DIMM, using non-temporal stores lead to a higher bandwidth than the others. Second, in Figure 1b, PMEP maintains a stable latency per cache line read, while Optane DIMM read latency increases with the growth of the pointer chasing region size. Both observations show that the real NVRAM system is not simply a slower DRAM DIMM-based system.

Inefficient NVRAM Simulation Models. Previous memory simulators, such as DRAMSim2 [5] and Ramulator [4], model memory architecture based on conventional DRAM systems. We find a large mismatch of bandwidth and latency characteristics between various simulators and our real-system profiling on Optane DIMM. Furthermore, we compare the pointer chasing load latency of Ramulator and Optane DIMM. Similar to PMEP results (Figure 1b), the simulated read latency appears to be stable, whereas Optane DIMM system profiling shows an increasing latency with larger pointer chasing regions.

Insufficient Profiling Tools. In order to develop a memory simulator that models the sophisticated performance behavior and microarchitecture of real NVRAM systems, we need to collect sufficient information about detailed performance characteristics. This requires a comprehensive architecture-level memory system profiling. Most previous performance profiling tools focus on investigating basic memory system performance characteristics, such as latency, bandwidth, and access counts. However, none of the widely-used performance profiling tools allows us to analyze detailed on-DIMM buffering and management schemes of Optane DIMM.



(a) Single thread bandwidth. (b) PtrChasing read latency.

Fig. 1. Comparison between PMEP [1] emulation and Optane DIMM-based real machine measurement. *Store-clwb* are writes followed by `clwb` cache write-back instructions; *store-nt* are non-temporal writes.

B. Contributions

Our paper makes three major contributions which will likely have long-term impact on both industry and academia. (1) Our profiler LENS is the first one that discovers the complex NVRAM microarchitecture design, including on-DIMM buffer (or queue) hierarchy, capacity, and access granularity. (2) Using LENS, we provide the first architecture level profiling of real NVRAM DIMM products in server systems. The observations we discovered help researchers to rethink the NVRAM system design. (3) We propose VANS, the first NVRAM simulator that implements the Optane DIMM microarchitecture design and achieves higher accuracy compared to previous DRAM-based main memory simulators.

C. LENS

To address the challenge of insufficient profiling tools, we propose a *Low-level profilEr for Non-volatile memory System* (LENS). LENS consists of a set of NVRAM profiling tools and low-level microbenchmarks. In this paper, we use LENS to profile the detailed architecture design of Optane DIMM.

LENS adopts three key components – *probers* – to examine the following three aspects of NVRAM architecture design, respectively.

- **Buffer prober** – Analyzes on-DIMM buffers' architecture and their properties, including buffer (or queue) hierarchy, capacity, and access granularity.
- **Policy prober** – Analyzes NVRAM control policies on data migration and multi-DIMM interleaving.
- **Performance prober** – Facilitates the above two probers to analyze performance characteristics, including memory bandwidth and the access latency of various buffers and queues in NVRAM DIMMs and iMCs.

[1]"Characterizing and Modeling Non-Volatile Memory Systems", in Proceedings of the 53rd IEEE/ACM International Symposium on Microarchitecture, pp. 496-508, October 2020

*Now at Google.

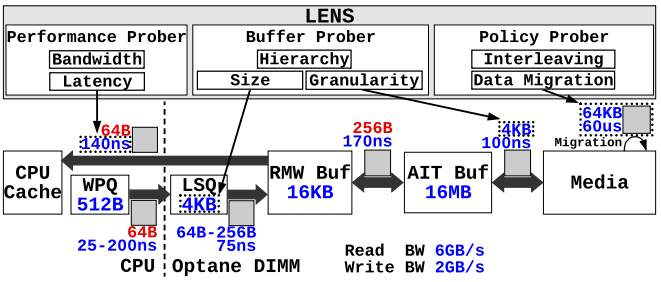


Fig. 2. LENS probers and Optane DIMM parameters. Red numbers are obtained from Intel documents; blue numbers are characterized by LENS.

Each prober employs customized microbenchmarks to trigger specific hardware behaviors, which lead to different latency and bandwidth patterns with different memory architecture properties. By analyzing the performance patterns, we identify the corresponding architecture properties and parameters.

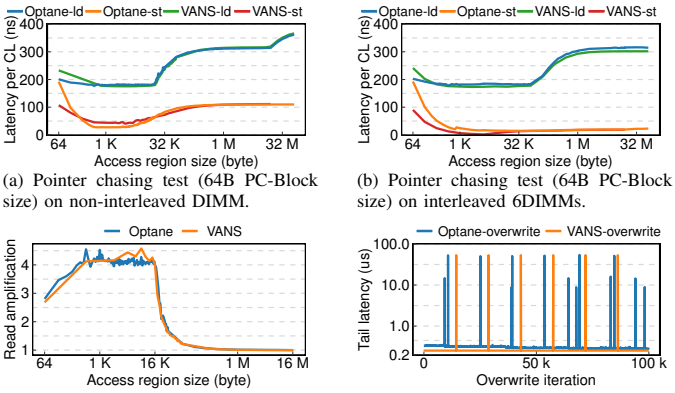
Figure 2 shows an overview of our real-machine characterization using LENS. Each prober identifies certain memory system architecture and performance characteristics, as illustrated by arrows in the figure. We identified the write-pending-queue (WPQ) size and multi-DIMM interleaving scheme in iMC. We also identified two on-DIMM buffers, a 16KB SRAM-based read-modify-write (RMW) buffer and a 16MB DRAM-based address indirection translation (AIT) [2] buffer, with 256B and 4KB access granularity, respectively. We find that these two buffers are organized as a two-level inclusive buffer hierarchy rather than independent buffers. We also identified a 4KB load-store-queue (LSQ) [2], which reorders the incoming requests to perform write combining. Finally, we identified a long tail-latency effect, which may be caused by wear-leveling data migration.

D. VANS

Based on our profiling observations, we build *Validated cycle-Accurate NVRAM Simulator* (VANS) that models Optane DIMM's microarchitectural design and its parameters.

VANS models both released Optane DIMM hardware components (e.g., WPQ in iMC) and undocumented microarchitectural components (e.g., LSQ, multi-level buffers). The iMC in VANS supports multi-DIMM control, which controls multiple DRAM and NVRAM DIMMs to provide both Memory and App Direct modes. We implement the WPQ in iMC with Asynchronized DRAM Refresh (ADR). The NVRAM DIMM model consists of an LSQ, an RMW Buffer, and an AIT. The LSQ serves as the highest-level storage in the DIMM, directly queuing load/store requests from the iMC. During each scheduling epoch, the LSQ performs write combining to reduce the number of read-modify-write operations.

The RMW Buffer receives read and write requests from the LSQ, and accesses the AIT Buffer at certain granularity (by default 256B based on our Optane DIMM analysis). The AIT consists of a translation table and a data buffer (AIT Buffer) in the on-DIMM DRAM. The translation table stores the records of CPU address to media address translation. It also stores the media wear-leveling records in each table entry. If one media block is wearing out, AIT stalls the inflight CPU writes to this



(a) Pointer chasing test (64B PC-Block size) on non-interleaved DIMM.

(b) Pointer chasing test (64B PC-Block size) on interleaved 6DIMMs.

(c) RMW Buffer's read amplification in Figure 3a test.

(d) Overwrite (256B region), each iteration is one 256B write.

Fig. 3. VANS performance validation with microbenchmarks: (a-c) pointer chasing and (d) overwrite.

block, migrates the data into another media block, updates the translation record, and then resumes the CPU write. The AIT Buffer stores data from the media to accommodate read and write requests from RMW Buffer.

Verification and Validation. To verify VANS DRAM model, we run SPEC2006 and SPEC2017 benchmarks on VANS and capture internal DRAM command traces. Then we employ Micron's DDR4 verification model with Cadence toolchain to test these command traces. The results demonstrate that our model does not generate any illegal DDR4 command, showing that our on-DIMM DRAM model complies with DDR4 specification.

To validate VANS NVRAM model, we run LENS microbenchmarks on VANS and compare the simulated load-/store latency, bandwidth and read amplification factors with Optane DIMM-based real-machine results. Figure 3(a-b) show the pointer chasing latency on a single DIMM and interleaved 6DIMMs. The simulated load/store latency curves match the curves of real-system profiling. Furthermore, VANS's read amplification (Figure 3c) and overwrite latency (Figure 3d) match the Optane DIMM's. Overall, VANS achieves over 86.5% accuracy across all metrics.

REFERENCES

[1] S. R. Dulloor, S. Kumar, A. Keshavamurthy, P. Lantz, D. Reddy, R. Sankaran, and J. Jackson, "System software for persistent memory," in *EuroSys*, 2014.

[2] Intel, "Intel® Optane™ DC Persistent Memory," 2019. [Online]. Available: https://www.intel.com/content/www/us/en/architecture-and-technology/optane-dc-persistent-memory

[3] J. Izraelevitz, J. Yang, L. Zhang, J. Kim, X. Liu, A. Memaripour, Y. J. Soh, Z. Wang, Y. Xu, S. R. Dulloor, J. Zhao, and S. Swanson, "Basic performance measurements of the Intel Optane DC persistent memory module," *arXiv*, 2019.

[4] Y. Kim, W. Yang, and O. Mutlu, "Ramulator: A fast and extensible DRAM simulator," *CAL*, 2016.

[5] P. Rosenfeld, E. Cooper-Balis, and B. Jacob, "DRAMSim2: A cycle accurate memory system simulator," *CAL*, 2011.

[6] H. Volos, G. Magalhaes, L. Cherkasova, and J. Li, "Quartz: A lightweight performance emulator for persistent memory software," in *Middleware*, 2015.

[7] J. Yang, J. Kim, M. Hoseinzadeh, J. Izraelevitz, and S. Swanson, "An empirical guide to the behavior and use of scalable persistent memory," in *FAST*, 2020.