Adaptive Memory and Storage Fusion on Non-Volatile One-Memory System

The 8th IEEE Non-Volatile Memory Systems and Applications Symposium, Hangzhou, China, August 18-21, 2019

OUTLINE

- Background: Non-Volatile Memory and Phase Change Memory
- Introduction: Observations, Motivation, and Objective
- Our Solution: One-Memory Consolidation Layer
- Experiments: Performance and Lifetime Evaluation
- Conclusion

Non-Volatile Memory

- From perfect technology in theory to existing product on market
  - Intel Optane Memory - 3D XPoint[1]
- Byte-addressable
  - Support byte-addressing for processor
- Non-volatile
  - Store information consistently with very little power consumption
- A wide range of variety and possible usages
  - PCM[2], MRAM[3], ReRAM[4], FeRAM[5], etc.

[1] F. T. Hady, A. Foong, B. Veal, and D. Williams, "Platform storage performance with 3D XPoint technology," Proceedings of the IEEE, vol. 105, pp. 1822 – 1833, Sept. 2017.
[2] H.-S. P. Wong, S. Raoux, S. Kim, J. Liang, J. P. Reifenberg, B. Rajendran, M. Asheghi, and K. E. Goodson, "Phase change memory," Proceedings of the IEEE, vol. 98, pp. 2201 – 2227, Dec. 2010.
[3] S. Tehrani, J. M. Slaughter, E. Chen, M. Durlam, J. Shi, and M. DeHerren, "Progress and outlook for MRAM technology," *IEEE Transactions on Magnetics*, vol. 35, pp. 2814 – 2819, Sept. 1999.
[4] H. Akinaga and H. Shima, "Resistive random access memory (ReRAM) based on metal oxides," Proceedings of the IEEE, vol. 98, pp. 2237 – 2251, Dec. 2010.
[5] J. F. Scott, Ferroelectric Memories. Springer Series in Advanced Microelectronics, Springer Berlin Heidelberg, 2013.

Phase Change Memory as Main Memory or Storage

- PCM vs. DRAM
  - Lower leakage power[1]
  - Higher scalability[2]
- PCM vs. NAND Flash
  - Lower R/W latency[3]
  - Higher throughput[4]

	DRAM	PCM	NAND
Idle Power	~W/GB	<0.1W/GB	<0.1W/GB
Cell density	$2/3F^2$	$5F^2$	$4\sim5F^2$
Read latency	10ns	20ns	25 μ s
Write latency	10ns	100ns	200 μ s
Endurance	10^{16}	$10^8\sim10^9$	$10^4\sim10^6$

Table 1: Comparison with current technology[3, 5]

- PCM has the potential to be used as both memory and storage

[1] P. Zhou, B. Zhao, J. Yang, and Y. Zhang, "A durable and energy efficient main memory using phase change memory technology," in International Symposium on Computer Architecture (ISCA), (Austin, TX, USA), June 2009.

[2] B. C. Lee, P. Zhou, J. Yang, Y. Zhang, B. Zhao, E. Ipek, O. Mutlu, and D. Burger, "Phase-change technology and the future of main memory," IEEE Micro, vol. 30, pp. 143 – 143, Jan. 2010.

[3] S. Eilert, M. Leinwander, and G. Crisenza, "Phase change memory: A new memory enables new memory usage models," in IEEE In- ternational Memory Workshop, (Monterey, CA, USA), May 2009.

[4] F. Bedeschi, C. Resta, O. Khouri, E. Buda, L. Costa, M. Ferraro, F. Pellizzer, F. Ottogalli, A. Pirovano, M. Tosi, R. Bez, R. Gastaldi, and G. Casagrande, "An 8Mb demonstrator for high-density 1.8V phase-change memories," in Symposium on VLSI Circuits, (Honolulu, HI, USA, USA), June 2004.

[5] J.-Y. Jung and S. Cho, "Memorage: Emerging persistent RAM based malleable main memory and storage architecture," in ACM International Conference on Supercomputing (ICS), Eugene, Oregon, USA, Jun. 2013.

OUTLINE

- Background: Non-Volatile Memory and Phase Change Memory
- Introduction: Observations, Motivation, and Objective
- Our Solution: One-Memory Consolidation Layer
- Experiments: Performance and Lifetime Evaluation
- Conclusion

Observations

- PCM is generally believed to enter the market first[1, 2, 3]
- A system using PCM as both memory and storage seems to be reasonable
- A limited number of write-cycle
  - The data retention time of a PCM frame will be shortened after many writes
- The inevitable performance-lifetime tradeoff

[1] J.-Y. Jung and S. Cho, "Memorage: Emerging persistent RAM based malleable main memory and storage architecture," in ACM International Conference on Supercomputing (ICS), Eugene, Oregon, USA, Jun. 2013.
[2] R. F. Freitas and W. W. Wilcke, "Storage-class memory: The next storage system technology," IBM Journal of Research and Development, vol. 52, pp. 439 – 447, July 2008.
[3] S. Raoux, G. W. Burr, M. J. Breitwisch, C. T. Rettner, Y.-C. Chen, R. M. Shelby, M. Salinga, D. Krebs, S.-H. Chen, H.-L. Lung, and C. H. Lam, "Phase-change random access memory: A scalable technology," IBM Journal of Research and Development, vol. 52, pp. 465 – 479, July 2008.

Motivation - Tradeoff Dilemma

- Always in-place execution
  - Best performance
  - Unconditionally performing in-place writes would quickly wear down some PCM frames
- Static configuration of memory and storage
  - Good lifetime
  - Falls back to the traditional dichotomy of separated memory and storage, incurs lots of overhead

Related Work

- Integrating Memory Management with a File System on a Non-Volatile Main Memory System[1]
  - Uses NVM as both memory and storage, in the same physical address space as DRAM
  - NVM blocks can be mapped into virtual memory address space for memory use
- NVM Duet: Unified Working Memory and Persistent Store architecture[2]
  - No advance partitioning of PCM resources between memory and storage
  - Relaxes the retention time constraints for memory
- Memorage: Emerging Persistent RAM Based Malleable Main Memory and Storage Architecture[3]
  - Memory may borrow free space from storage during memory shortage

[1] S. Oikawa, "Integrating memory management with a file system on a non-volatile main memory system," in ACM Symposium on Applied Computing (SAC), Coimbra, Portugal, Mar. 2013.
[2] R.-S. Liu, D.-Y. Shen, C.-L. Yang, S.-C. Yu, and C.-Y. M. Wang, "NVM duet: Unified working memory and persistent store architecture," in International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS), Salt Lake City, Utah, USA, Mar. 2014.
[3] J.-Y. Jung and S. Cho, "Memorage: Emerging persistent RAM based malleable main memory and storage architecture," in ACM International Conference on Supercomputing (ICS), Eugene, Oregon, USA, Jun. 2013.

Objective

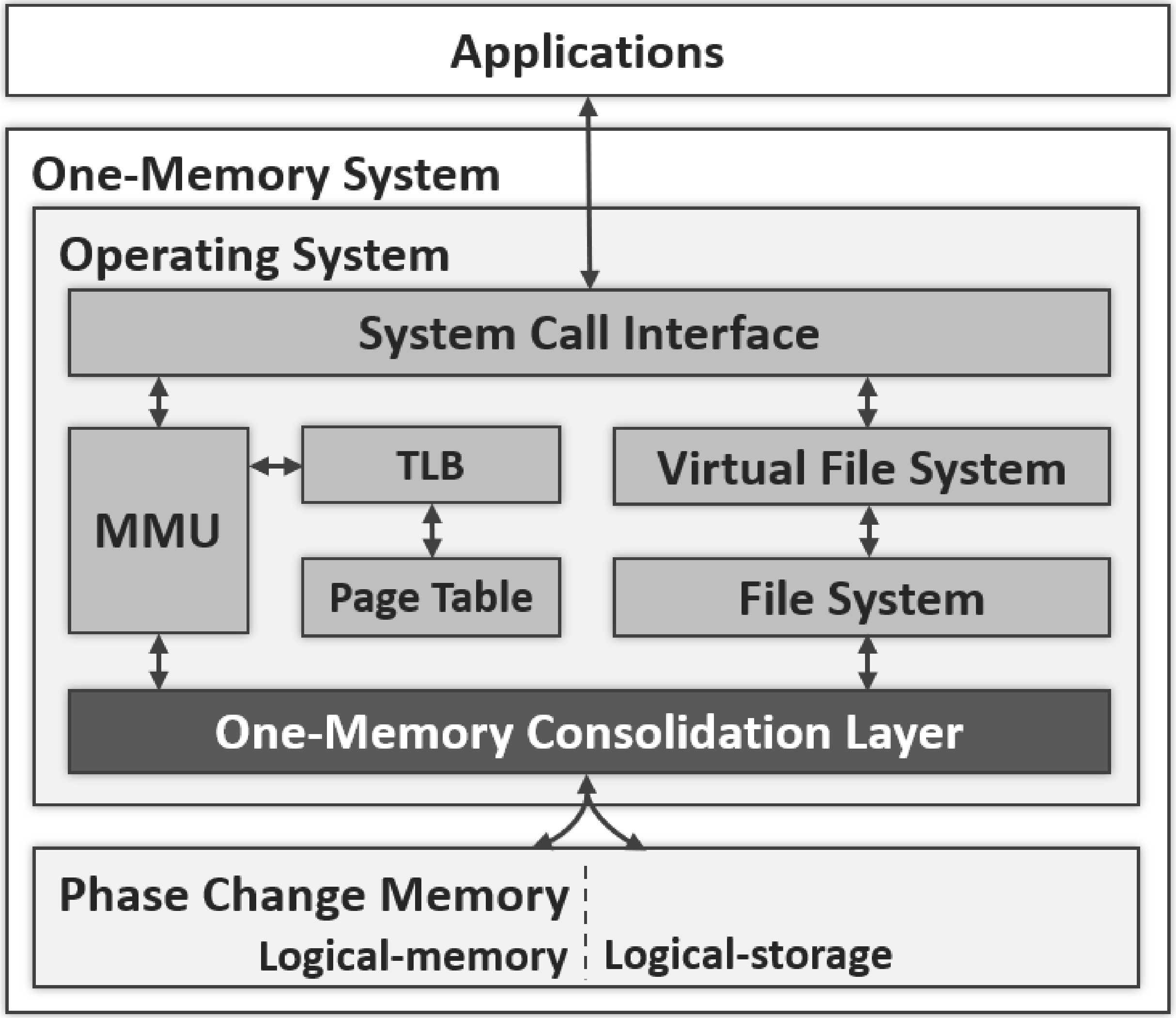
- Leverage PCM as both memory and storage in a system
  - The required data retention time of main memory is much shorter than that of storage
- Design a self-adaptive PCM management strategy for online storage and memory requests
  - Minimize the number of extra writes caused by the data movement between memory and storage for performance optimization
  - Meet the capacity and retention time constraints of the storage with a little lifetime sacrifice

OUTLINE

- Background: Non-Volatile Memory and Phase Change Memory
- Introduction: Observations, Motivation, and Objective
- Our Solution: One-Memory Consolidation Layer
- Experiments: Performance and Lifetime Evaluation
- Conclusion

System Architecture: One-Memory System

- PCM as both memory and storage
  - Located on the main memory bus
  - No longer physically separated
  - Only logically different
- One-Memory Consolidation Layer (OMCL) takes place beneath the OS
  - Be installed as an extension module
  - Judiciously select requests to be performed in-place or out-place

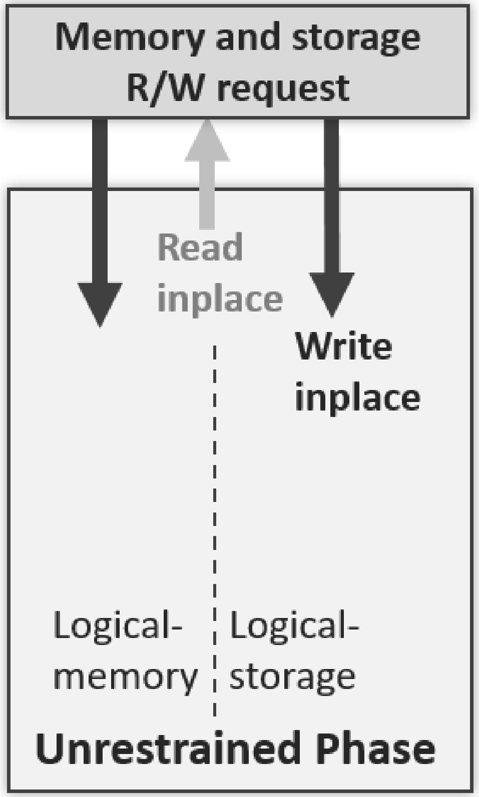


Definitions of PCM Frames

- Retention-time-qualified frames
  - Can be used for both storage and memory purpose
- Retention-time-unqualified frames
  - Can only be used for memory purpose
  - Retention time relaxation
- Frames used for storage purpose: logical-storage frames
- Frames used for memory purpose: logical-memory frames

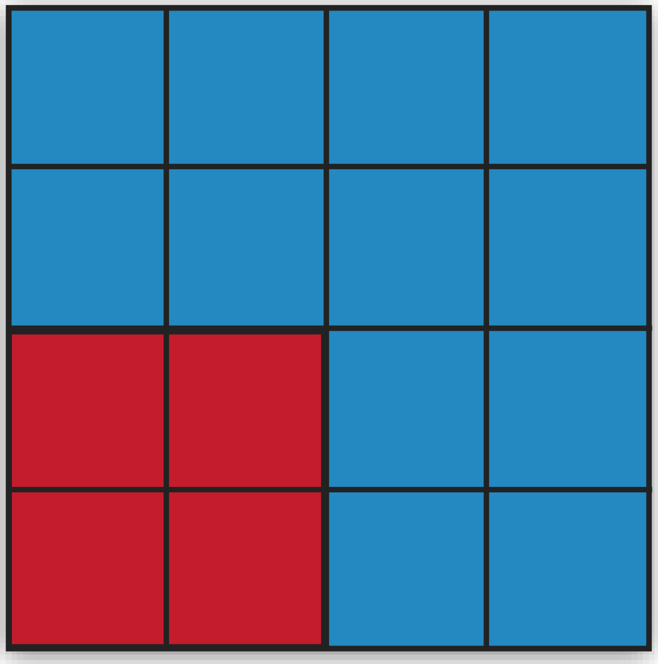
Strategy Overview

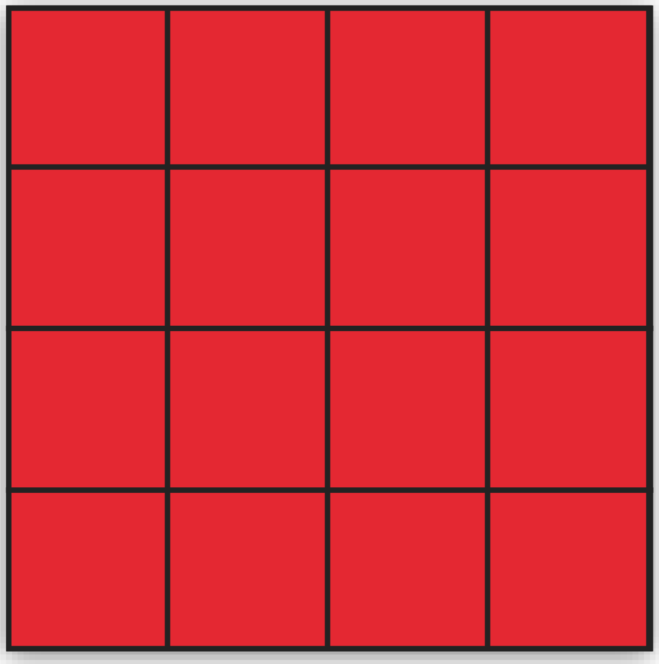
- Read requests are always performed in-place
- Write requests
  - Unrestrained Phase
    - Always write in-place



Unrestrained Phase (1/2)

- All the frames are retention-time-qualified for storage at first
  - All the frames can be used as logical-storage and logical-memory frames
- Requests are all performed in-place
  - Optimize the performance
  - An anti-wear leveling approach


Anti-wear leveling


Wear leveling

Unrestrained Phase (2/2)

- Becoming retention-time-unqualified frame
    - Logical-memory
        - Keep using
    - Logical-storage
        - Copy the data to a free retention-time-qualified frame
        - Original frame will be used for logical-memory frame in priority

Vigilant Phase (1/3)

- Shifts to Vigilant Phase depending on two factors
- α , the number of retention-time-unqualified frames over total frames minus logical-storage frames
  - Frames holding hot data continue to wear out
- β , the average write-cycle of retention-time-qualified frames
  - Frames are equally written, causing a false appearance that there are still plenty of retention-time-qualified frames left

Vigilant Phase (2/3)

- Use retention-time-unqualified frames for logical-memory in priority
- Monitor the in-place write frequency on logical-storage frames
    - A short-term measurement to detect bursts of writes
    - A long-term measurement to identify relatively low-frequent but continuous writes
- Write the storage data to logical-memory frames if it exceeds the limit

Vigilant Phase (3/3)

- The data retention time of logical-memory frames is still an issue
- No write request is issued to a logical memory frame within a timeout interval
  - Memory data
    - Conduct in-place data refresh
  - Cached storage data
    - Write the storage data back to its original logical-storage frame

Protected Phase

- Shift to Protected Phase if α or β exceeds the limit of the next level
- In-place writes on logical-storage frames are forbidden
  - Move data to logical-memory frame for write operation
- Protect the last remaining retention-time-qualified frames

OUTLINE

- Background: Non-Volatile Memory and Phase Change Memory
- Introduction: Observations, Motivation, and Objective
- Our Solution: One-Memory Consolidation Layer
- Experiments: Performance and Lifetime Evaluation
- Conclusion

Experiment Setup: Input Request Trace[1, 2]

- Storage request
  - Read: 40%, write: 20%
  - Generated randomly in a normal distribution
  - Data footprint set as 50% of the total PCM capacity
- Memory request
  - Write: 40%
  - Generated randomly in a uniform distribution
  - Data footprint set as 10% of the total PCM capacity
- Maximum consecutive writes to the same frame: 10

[1] Live Optics Basics: Read/Write Ratio. https://support.liveoptics.com/hc/en-us/articles/229590547-Live-Optics-Basics-Read-Write-Ratio

[2] D. Arteaga, M. Zhao, P. V. Riezen, and L. Zwart. A trace-driven analysis of solid-state caching in cloud computing systems. http://www.cloudvps.com.

Experiment Setup: Test Models (1/2)

- OMCL
  - The maximum write-cycle of a PCM frame is set to 10^3
  - Limit of α and β
    - Unrestrained Phase: 0.3 and 0.4*PCM frame' s maximum write-cycle
    - Vigilant Phase: 0.9 and 0.6*PCM frame' s maximum write-cycle
  - Limit of write frequency monitors
    - Short-term: 6 writes within 10 requests
    - Long-term: 300 writes within 1000 requests

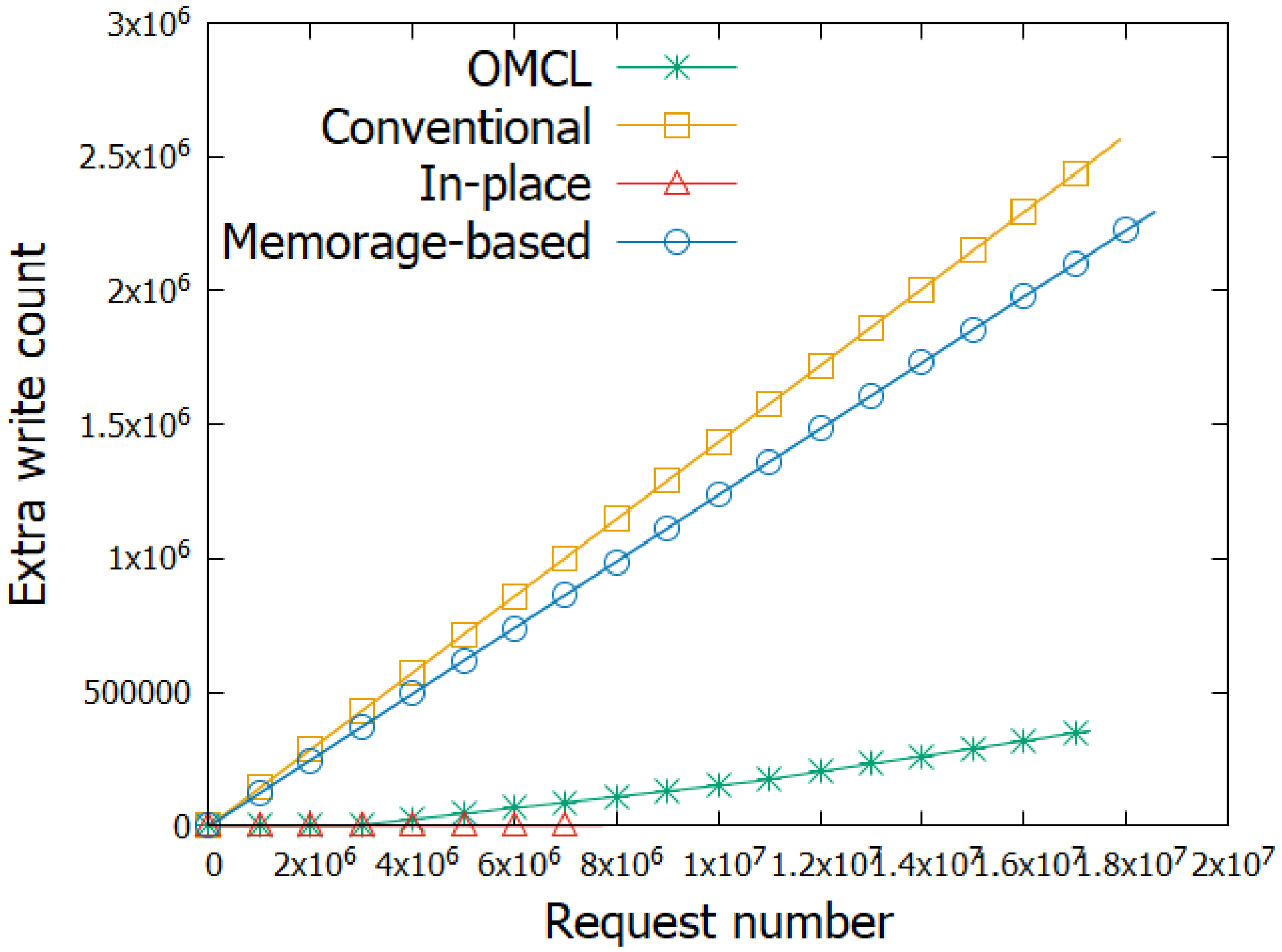
Experiment Setup: Test Models (1/2)

- In-place
  - Perform all read and write requests in-place
- Conventional
  - Separate the PCM into 10% memory and 90% storage
  - Cache storage data into memory space if the data are going to be written
- Memorage-based[1]
  - Same with Conventional
  - Memory may borrow an extra 5% of the PCM space from storage

[1] J.-Y. Jung and S. Cho, "Memorage: Emerging persistent RAM based malleable main memory and storage architecture," in ACM International Conference on Supercomputing (ICS), Eugene, Oregon, USA, Jun. 2013.

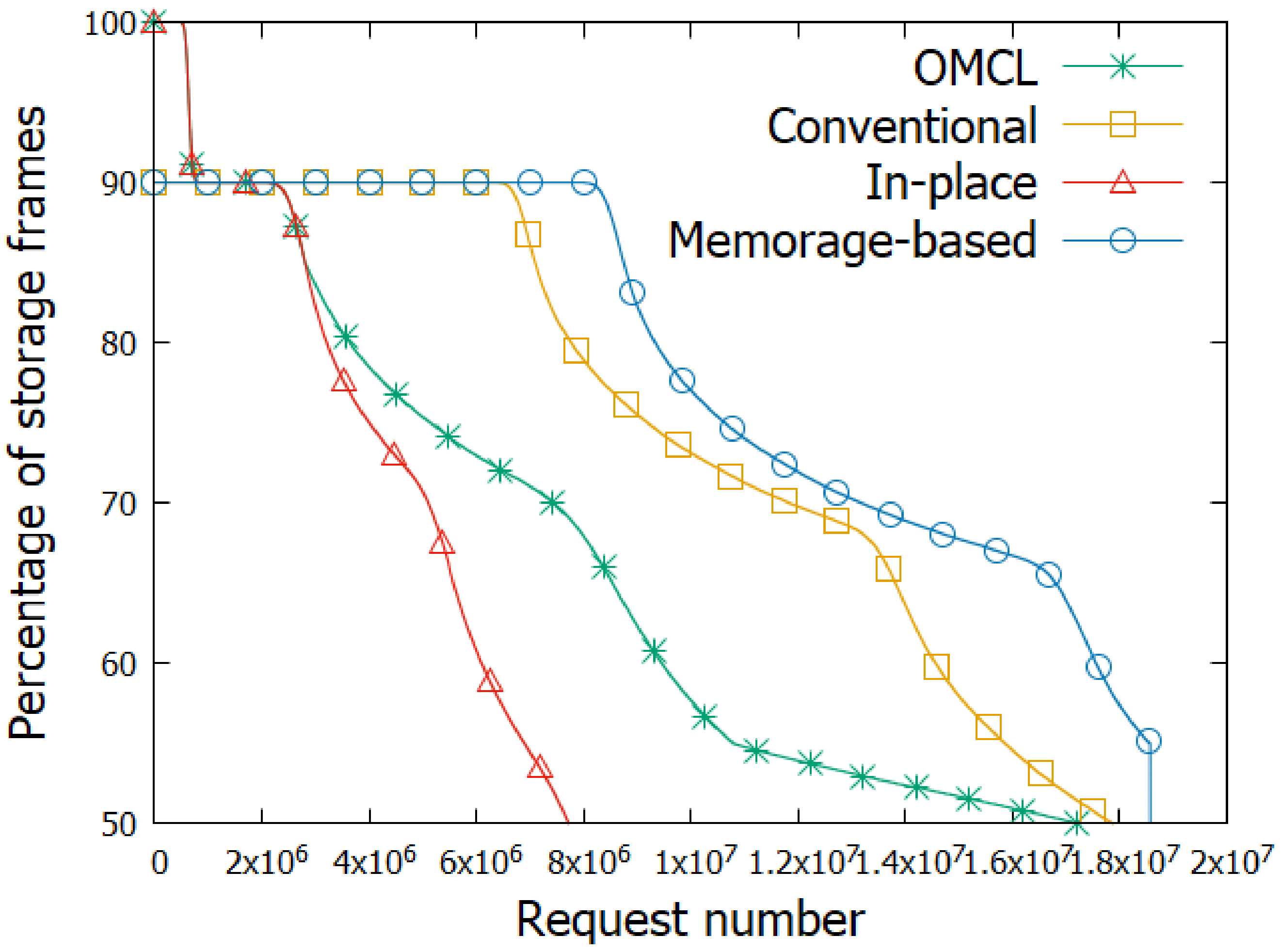
Number of Extra Writes

- The Conventional model has the highest number of extra writes, while the In-place model has the lowest
- The Memorage-based model does reduce its extra writes in comparison with the Conventional model
- OMCL reduces the extra writes by up to 86.1% when compared with the Conventional model



Lifetime – Percentage Retention-Time-Qualified Frames

- The In-place model has the shortest system lifetime, while the Memorage-based model has the longest
- OMCL sacrifices 3.4% of its lifetime when compared to the Conventional model, in return for the tremendous advantage of performance in terms of extra write reduction



OUTLINE

- Background: Non-Volatile Memory and Phase Change Memory
- Introduction: Observations, Motivation, and Objective
- Our Solution: One-Memory Consolidation Layer
- Experiments: Performance and Lifetime Evaluation
- Conclusion

Conclusion

- We propose OMCL with a Phase-Shift mechanism to manage different priority consideration on system performance and lifetime as the system ages
- We aim to optimize the performance of One-Memory System by minimizing the overhead of extra writes, while sacrificing the PCM's lifetime as little as possible
- The experimental results show that OMCL improves the system performance by reducing the overhead of extra writes up to 86.1%, while only 3.4% of the system lifetime is sacrificed