Persistent Memory Preview Session

Ian Neal

University of Michigan, CSE

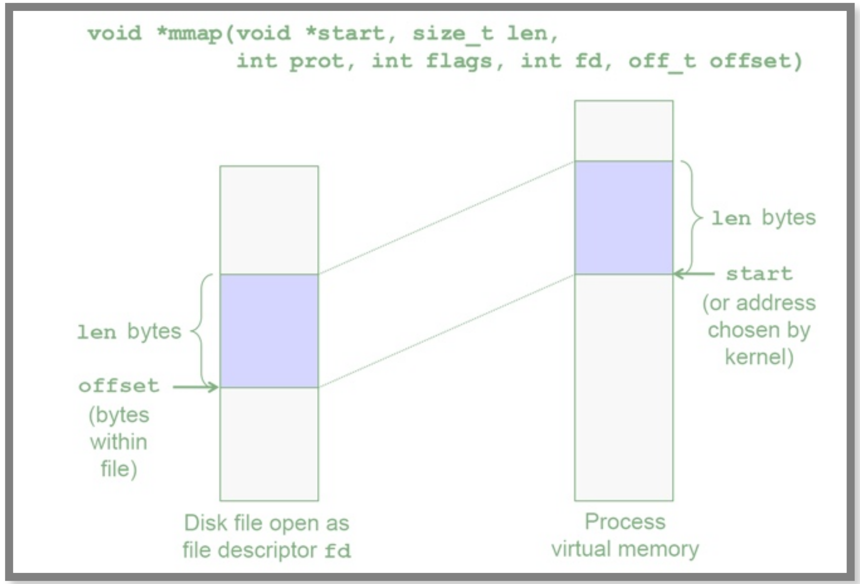
iangneal@umich.edu (about.iangneal.io)

OSDI 21

USENIX ATC '21

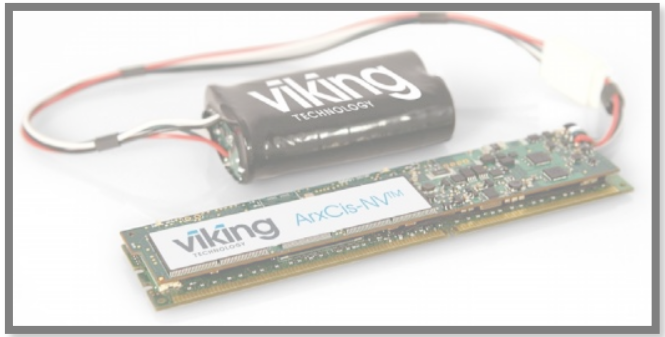
What is Persistent Memory (PM)?

- Like memory, but persistent!
- Retains data without active power



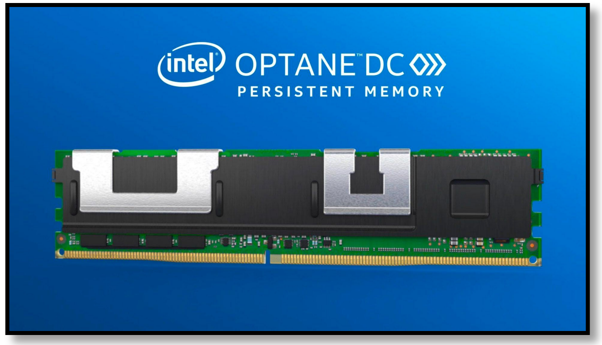
```
void *mmap(void *start, size_t len,
           int prot, int flags, int fd, off_t offset)
```

Abstraction



Powered Backups

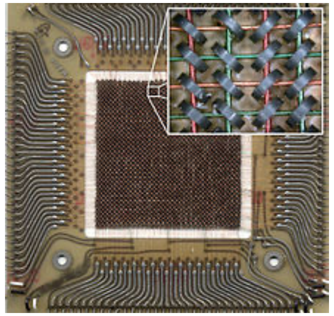
Source: Viking Technology



Non-volatile Media

Source: Intel Corporation

Timeline

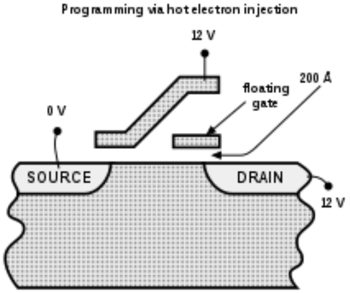


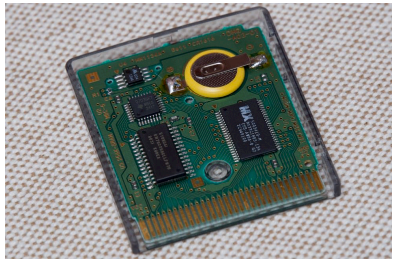
Source: Wikimedia Commons

Magnetic-core memory
(1955—1975)

EEPROM
(1973—today)

- Byte-addressable
- 1:1000 RW latency





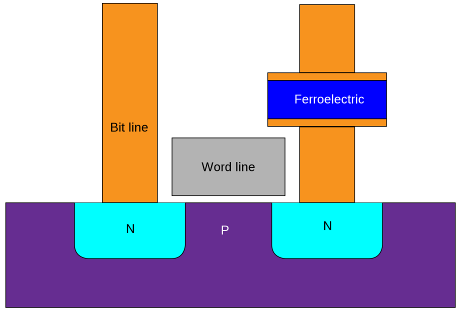
Source: Andrew Cunningham (Ars Technica)

Battery-backed RAM
(1980s—today)

Research on PM abstractions

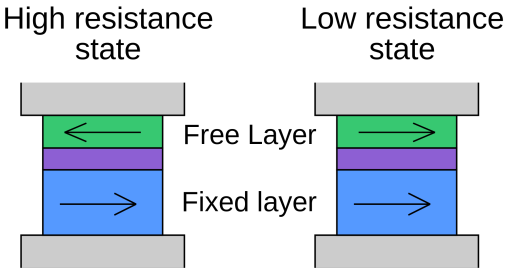
(1980s)

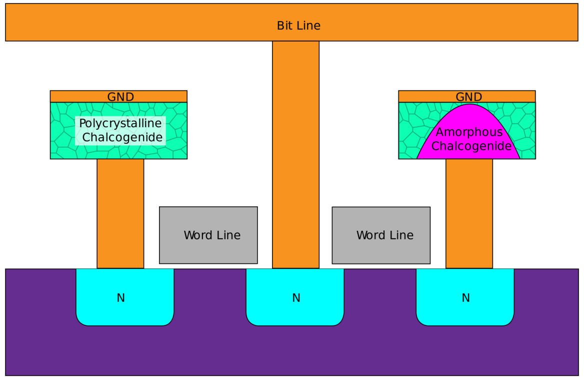
- Abstraction (e.g., PL->DB)
- “Not practical”



FeRAM
(1996—today)

STT-MRAM
(2008—today)

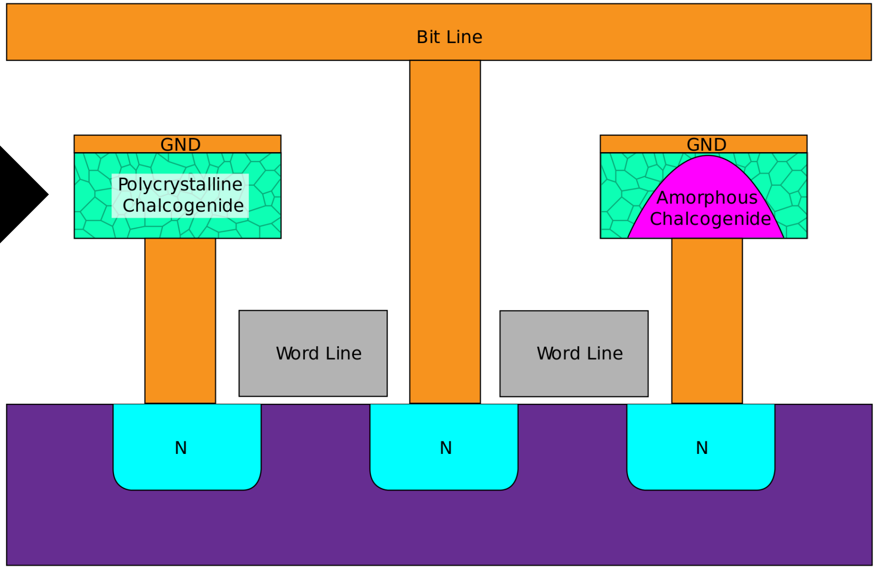




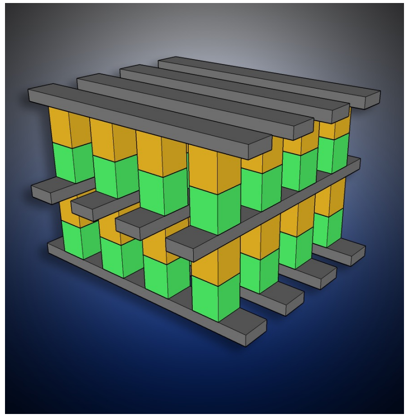
Phase-Change RAM (PCM)
(2008—today)

Phase Change Memory

- Phase-change RAM (PCM)
- 3D XPoint (***high density*** PCM)
- **2015**: joint Intel+Micron announcement
- **2017**: Optane SSDs available
- **2019**: DIMMs (Series 100) available



A PCM cell in low-resistance state (left) and high-resistance state (right).

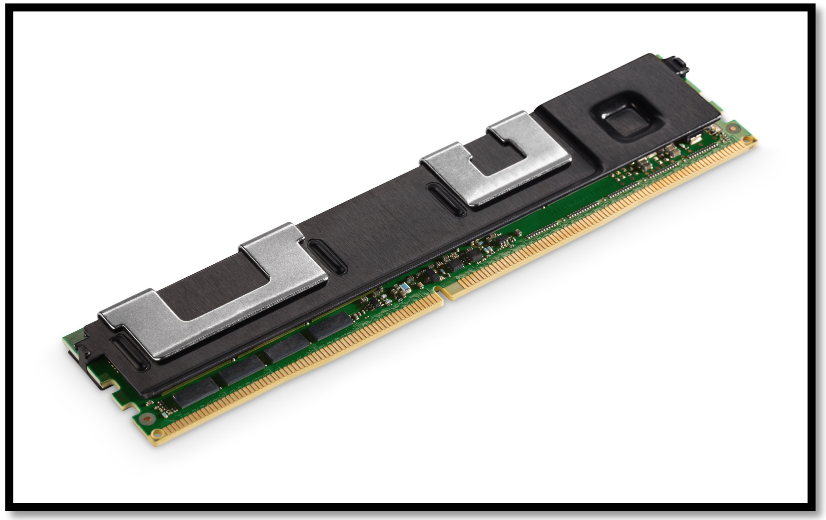


3D XPoint 2 layer diagram.

Image Sources: Wikimedia Commons

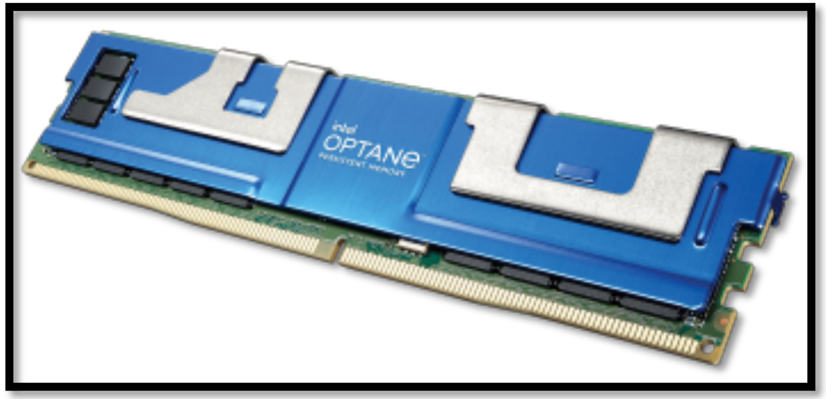
Modern Production

- PM, NVM, SCM ≅ Intel Optane DC Persistent Memory Module



Series 100

- 2-3x slower than DRAM (~300ns)
- Requires explicit cache flushing (CPU cache is volatile, PM isn't)



Series 200

- eADR (extended ADR) auto-persists updates in CPU cache
- ~32% higher bandwidth than Series 100

Image Sources: Intel Corporation

Modern PM Research

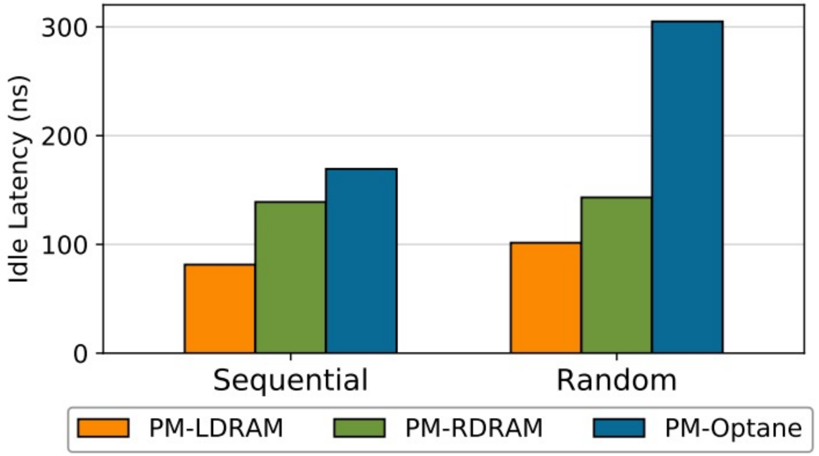
1. Persistent Data Structures and Storage Systems

2. Volatile Use Cases

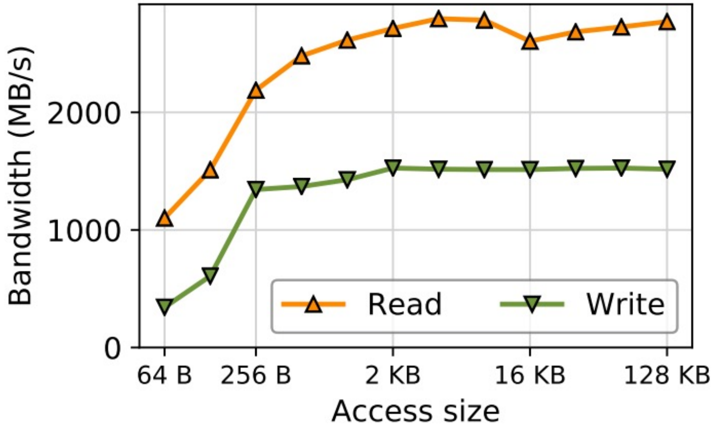
3. Developer Tools

PM Storage Systems (Area 1)

- Only 2-3x slower than DRAM
- ~8x denser (128-512GB devices)
- Different read/write, seq/rand perf
- Software overhead easily exposed



Read latency of Optane DC memory on a cache miss.



Optane DC memory bandwidth.

Graph Sources: **Basic Performance Measurements of the Intel Optane DC Persistent Memory Module.** *Izraelevitz et al.*

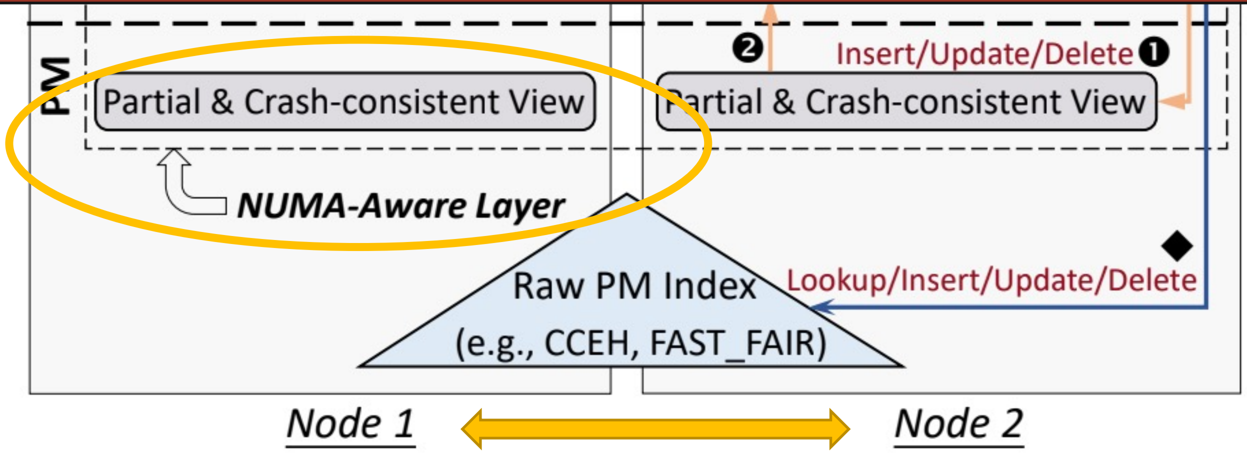
PM Storage Systems Papers

Nap: A Black-Box Approach to NUMA-Aware Persistent Memory Indexes [OSDI]

Qing Wang, Youyou Lu, Junru Li, and Jiwu Shu

Location of PM accesses must be accounted for!

2.3x, 1.6x higher throughput for write, read-intensive workloads!


PM
❷
Insert/Update/Delete ❶
Partial & Crash-consistent View
Partial & Crash-consistent View
NUMA-Aware Layer
◆
Raw PM Index
(e.g., CCEH, FAST_FAIR)
Lookup/Insert/Update/Delete
Node 1
Node 2

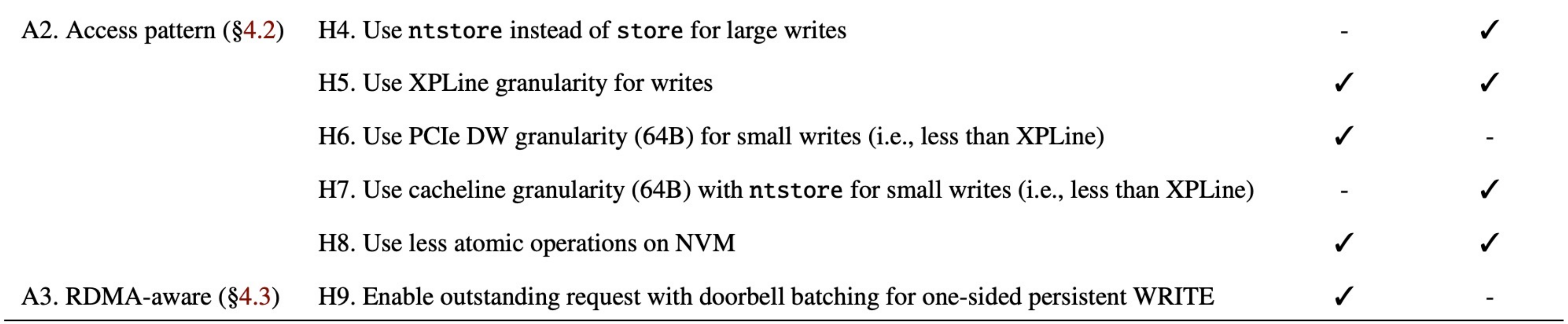
Accesses across nodes only have 59% bandwidth!

PM Storage Systems Papers

Characterizing and Optimizing Remote Persistent Memory with RDMA and NVM [ATC]

Xingda Wei, Xiating Xie, Rong Chen, Haibo Chen, Binyu Zang

Systems build on ***emulated*** NVM perform poorly on ***real*** NVM!



A2. Access pattern (§4.2)	H4. Use `ntstore` instead of `store` for large writes	-	✓
	H5. Use XPLine granularity for writes	✓	✓
	H6. Use PCIe DW granularity (64B) for small writes (i.e., less than XPLine)	✓	-
	H7. Use cacheline granularity (64B) with `ntstore` for small writes (i.e., less than XPLine)	-	✓
	H8. Use less atomic operations on NVM	✓	✓
A3. RDMA-aware (§4.3)	H9. Enable outstanding request with doorbell batching for one-sided persistent WRITE	✓	-

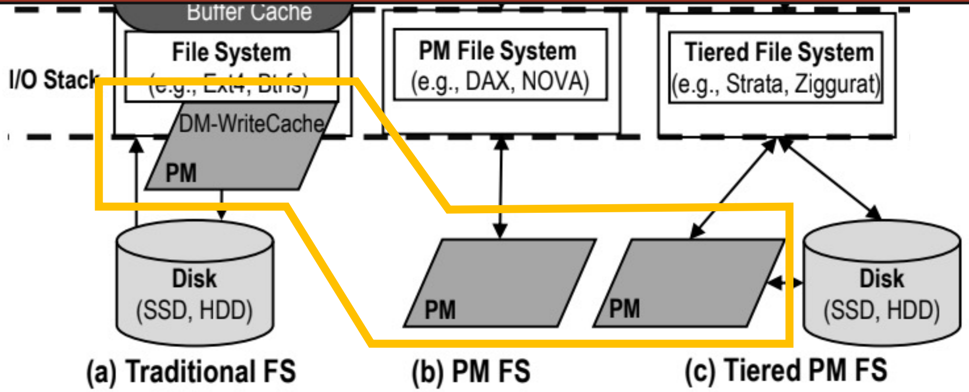
Up to 2.4x better performance!

PM Storage Systems Papers

First Responder: Persistent Memory Simultaneously as High Performance Buffer Cache and Storage [ATC]

Hyunsub Song, Shean Kim, J. Hyun Kim, Ethan JH Park, Sam H. Noh

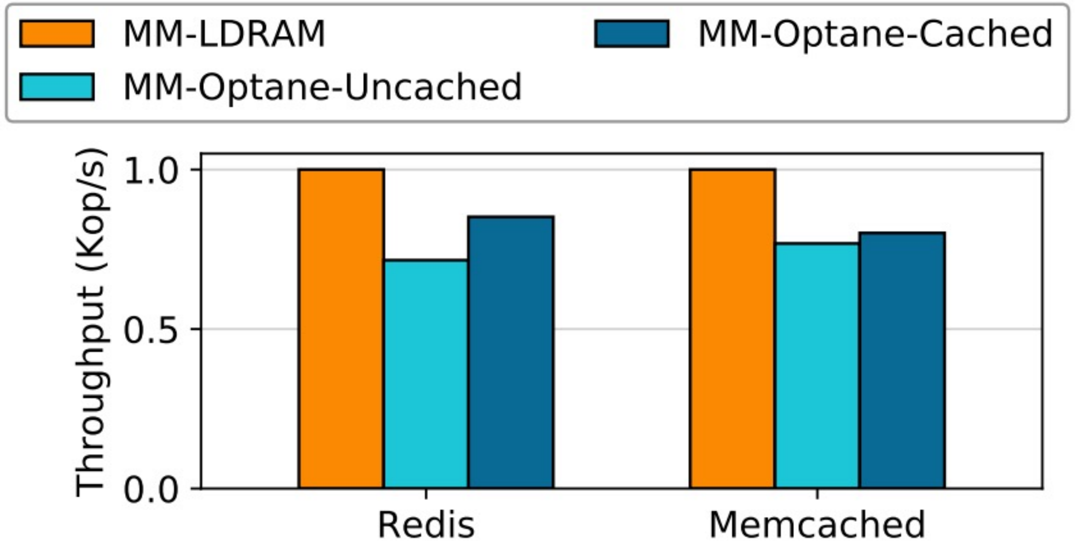
PM can easily accelerate existing systems!



PM as Volatile Memory (Area 2)

+ Larger pools of memory per node
+ Lower energy costs
+ Faster than SWAP

- Higher latency than DRAM
- Lower memory bandwidth



Throughput of memory-caching services using DRAM (**MM-LDRAM**), NVM with DRAM as a cache (**MM-Optane-Cached**), and NVM directly (**MM-Optane-Uncached**).

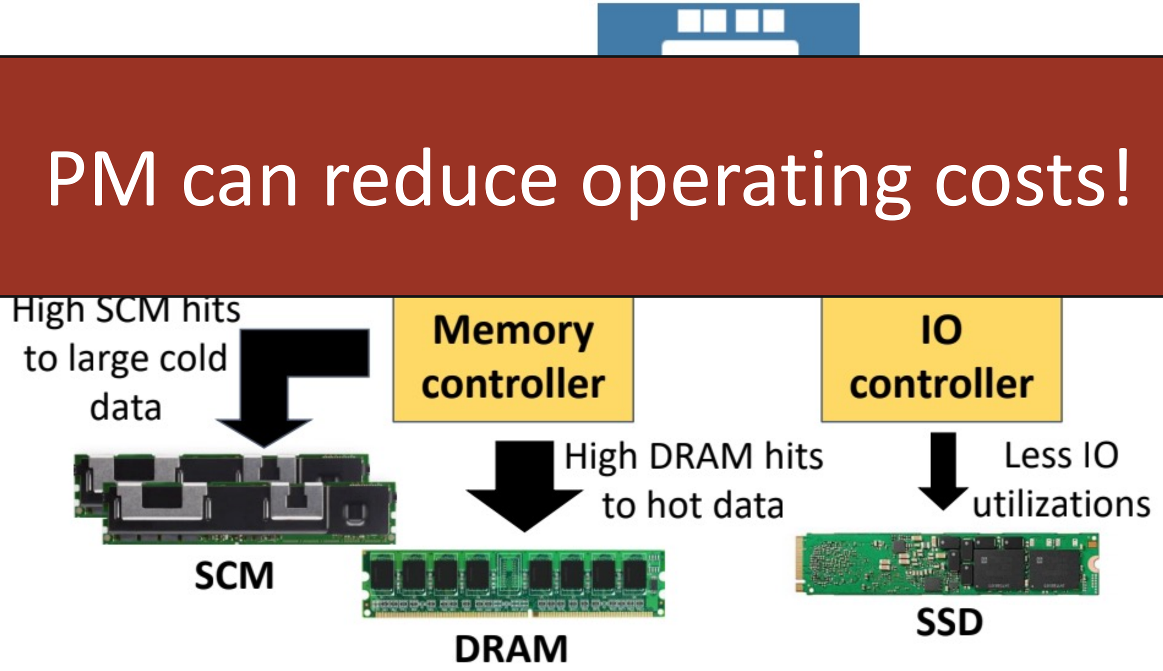
Graph Sources: **Basic Performance Measurements of the Intel Optane DC Persistent Memory Module.** *Izraelevitz et al.*

PM as Volatile Memory Paper

Improving Performance of Flash Based Key-Value Stores Using Storage Class Memory as a Volatile Memory Extension [ATC]

Hiwot Tadese Kassa, Jason Akers, Mrinmoy Ghosh, Zhichao Cao, Vaibhav Gogte, Ronald Dreslinski

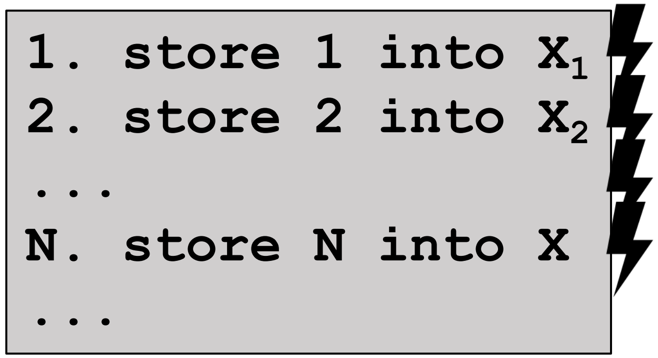
PM can reduce operating costs!



Up to 80% higher throughput at 43-48% lower cost!

PM Developer Tools (Area 3)

- Many crash states to reason about
- Adding cache flushes can be tedious/error prone (Series 100)
- Even more challenging for legacy code
  - Reason about entire code base at once
  - May restructure to remove FS calls
  - May add durability to volatile structures



1. store 1 into X_1

2. store 2 into X_2

...

N. store N into X

...

Updating a PM data structure.

PM Developer Tool Papers

Ayudante: A Deep Reinforcement Learning Approach to Assist Persistent Memory Programming [ATC]

Hanxian Huang, Zixuan Wang, Juno Kim, Steven Swanson, Jishen Zhao

Automatic conversion tools can do just as well as developers!

PM Storage Systems Papers

TIPS: Making Volatile Index Structures Persistent with DRAM-NVMM Tiering [ATC]

R. Madhava Krishnan, Wook-Hee Kim, Xinwei Fu, Sumit Kumar Monga, Hee Won Lee, Minsung Jang, Ajit Mathew, Changwoo Min

Reusing highly optimized volatile indices is possible and ***preferable***!

3-10x higher performance!

Session Information

OSDI 2021

- **Storage**
  - Nap: A Black-Box Approach to NUMA-Aware Persistent Memory Indexes

ATC 2021

- **Peeking over the Fence: RDMA**
  - Characterizing and Optimizing Remote Persistent Memory with RDMA and NVM
- **Friends Fur-Ever: Persistent Memory and In-Memory Computing**
  - Ayudante: A Deep Reinforcement Learning Approach to Assist Persistent Memory Programming
  - TIPS: Making Volatile Index Structures Persistent with DRAM-NVMM Tiering
  - Improving Performance of Flash Based Key-Value Stores Using Storage Class Memory as a Volatile Memory Extension
  - First Responder: Persistent Memory Simultaneously as High Performance Buffer Cache and Storage