

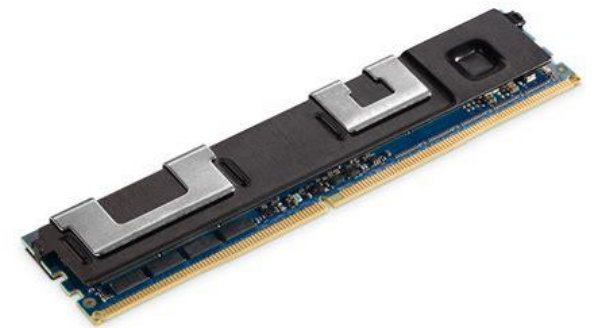
# An Empirical Guide to the Behavior and Use of Scalable Persistent Memory

Jian Yang, Juno Kim, Morteza Hoseinzadeh, **Joseph Izraelevitz**,  
Steven Swanson

*Non-Volatile Systems Laboratory  
Department of Computer Science & Engineering  
University of California, San Diego*

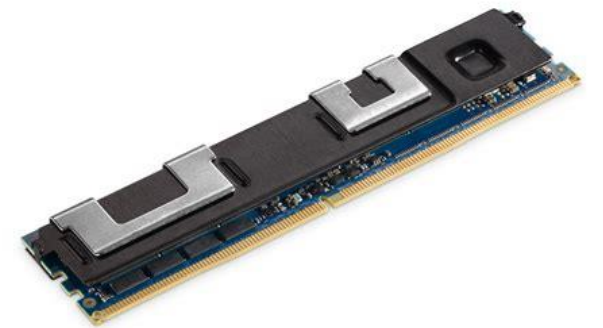
*Department of Electrical, Computer & Energy Engineering  
University of Colorado, Boulder*

# Optane DIMMs are Here!



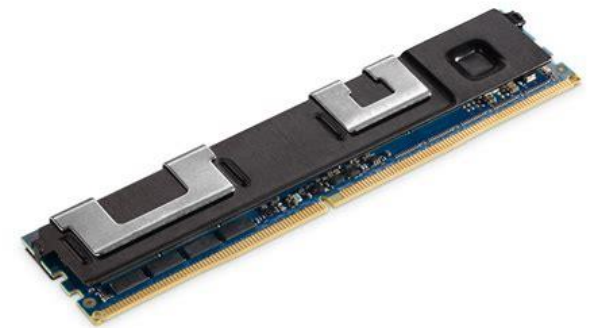
# Optane DIMMs are Here!

Cool.



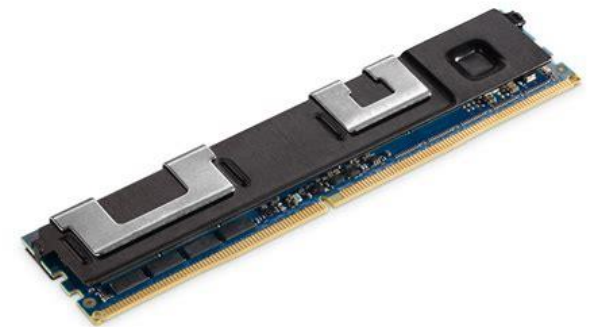
# Optane DIMMs are Here!

- Not Just Slow Dense DRAM™



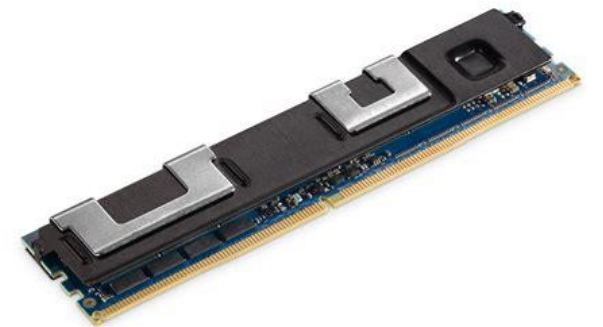
# Optane DIMMs are Here!

- Not Just Slow Dense DRAM™
- Slower media
  - > More complex architecture
  - > Second-order performance anomalies
  - > Fundamentally different



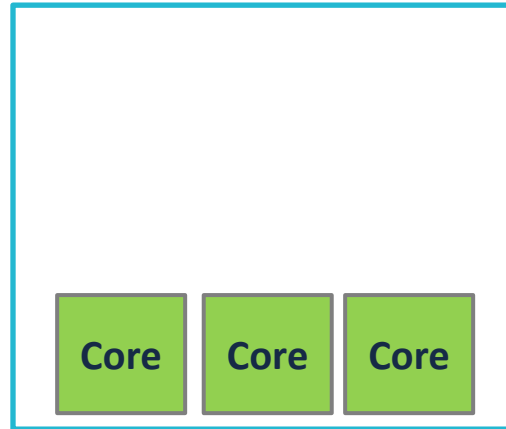
# Outline

- Background
- Basics: Optane DIMM Performance
- Lessons: Optane DIMM Best Practices
  - \*not included: Emulation Study, Best Practices in Macrobenchmarks (see the paper)
- Conclusion



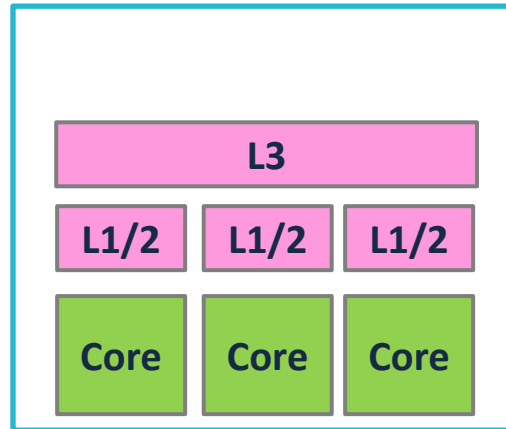
# Background

# Background: Optane in the Machine

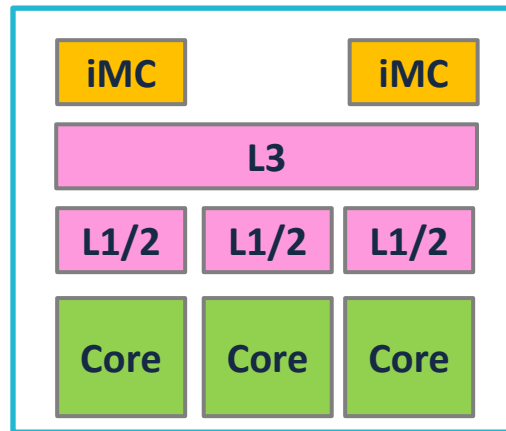




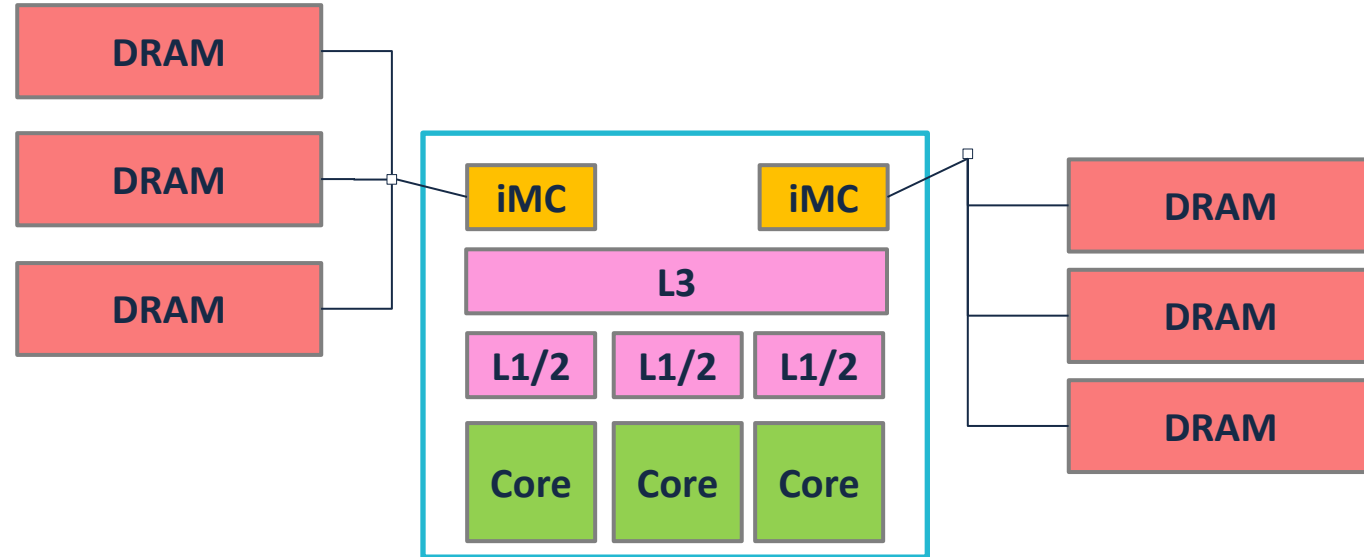
# Background: Optane in the Machine



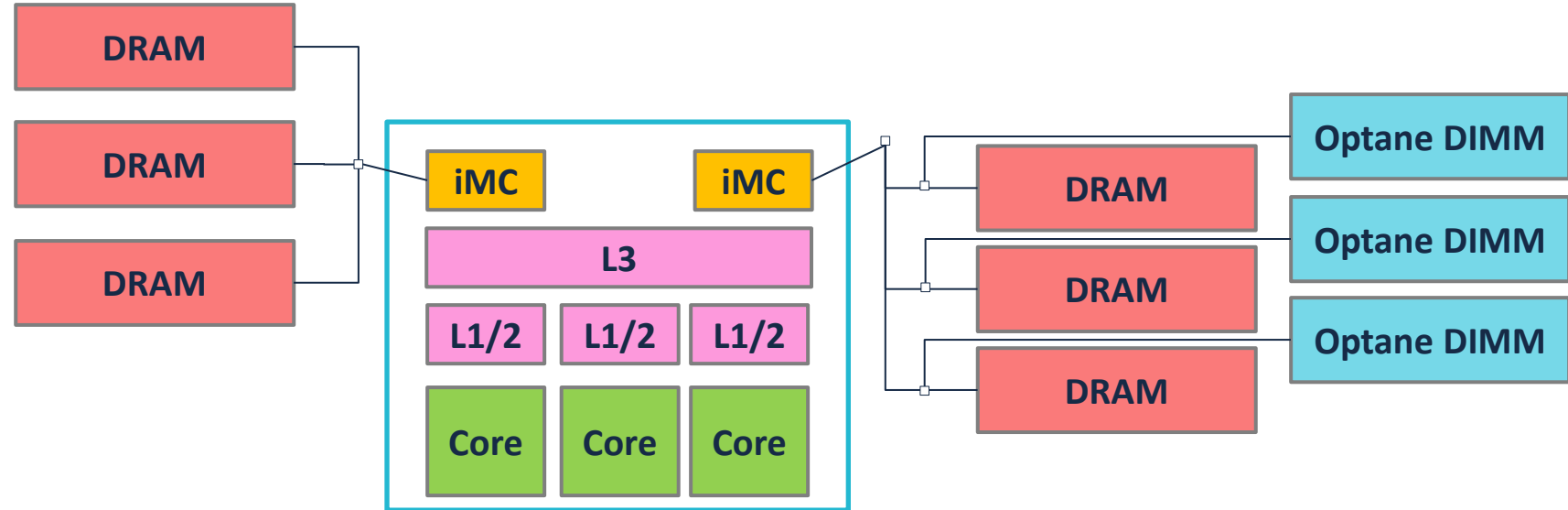
# Background: Optane in the Machine



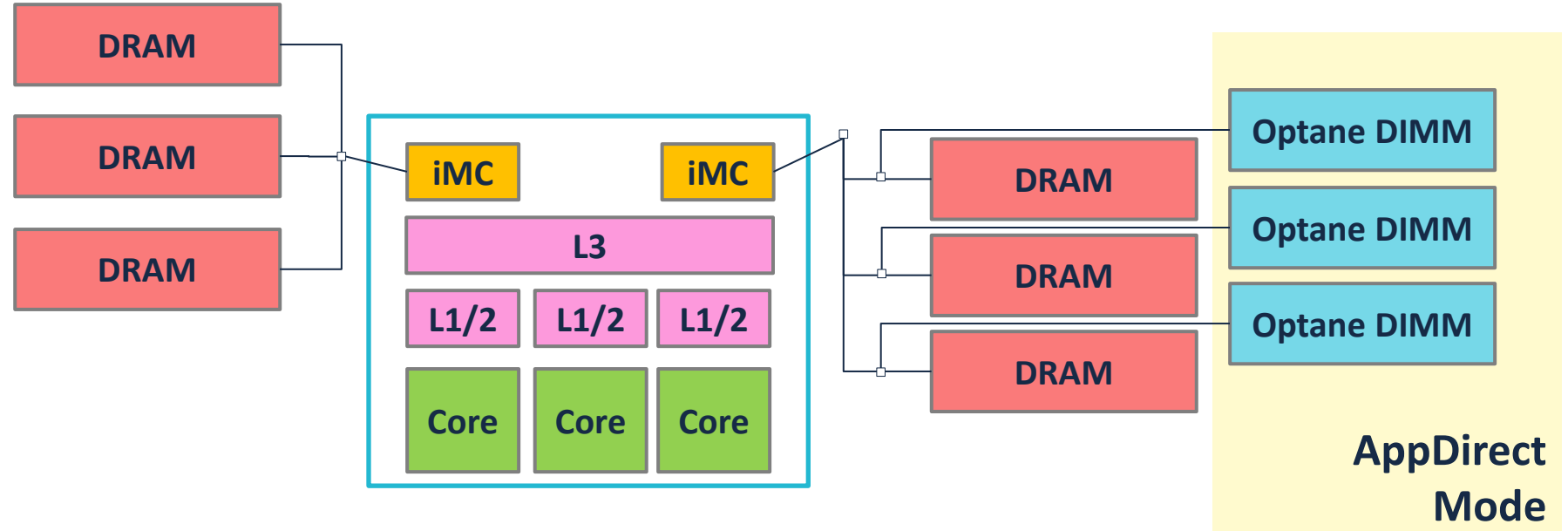
# Background: Optane in the Machine



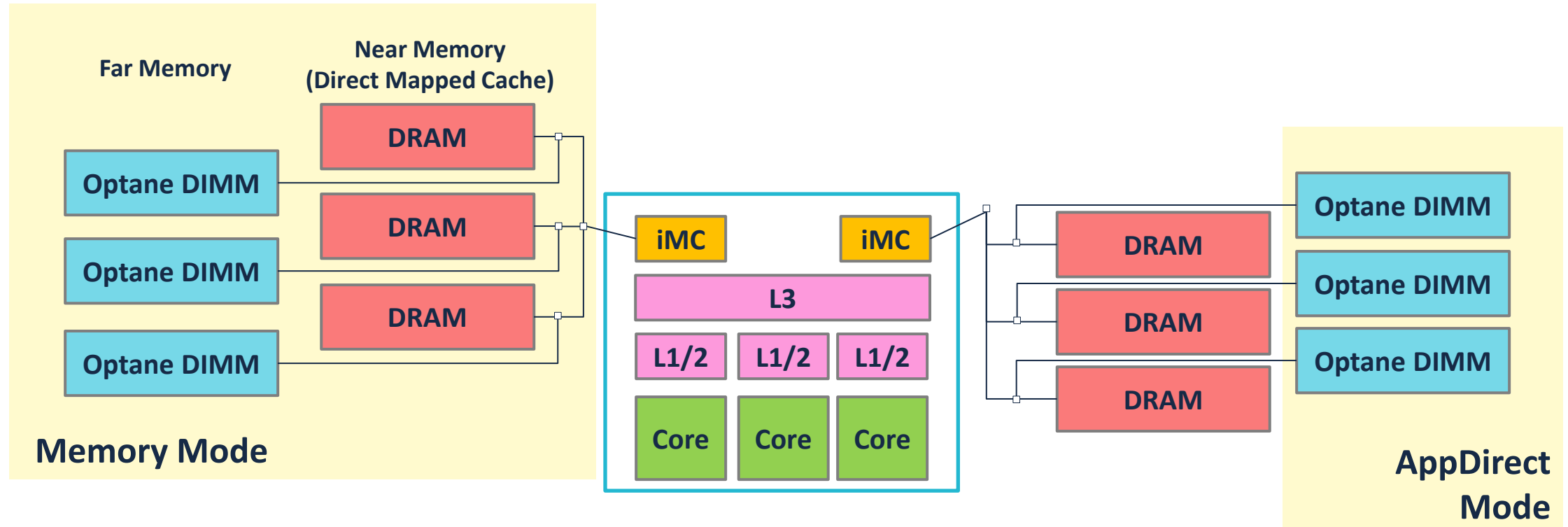
# Background: Optane in the Machine



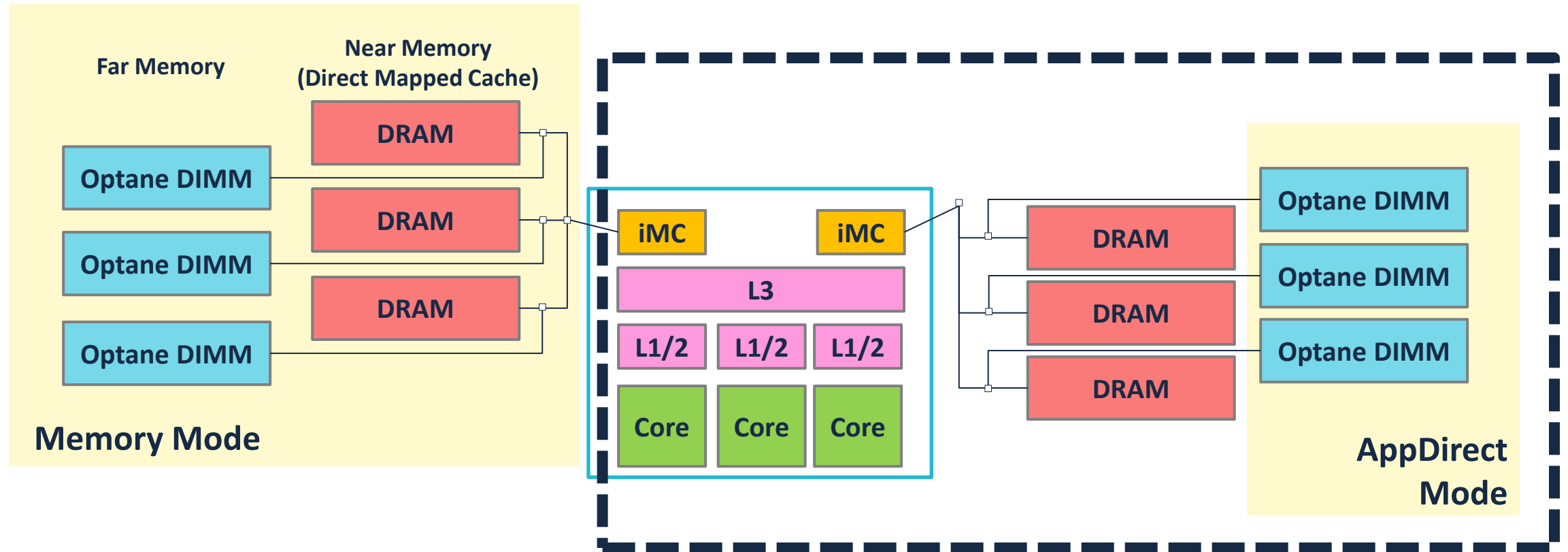
# Background: Optane in the Machine



# Background: Optane in the Machine

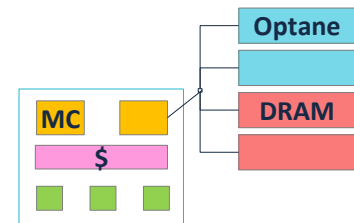


# Background: Optane in the Machine



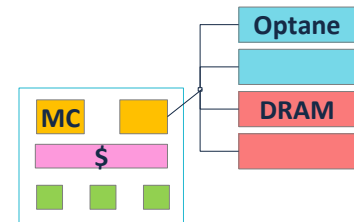
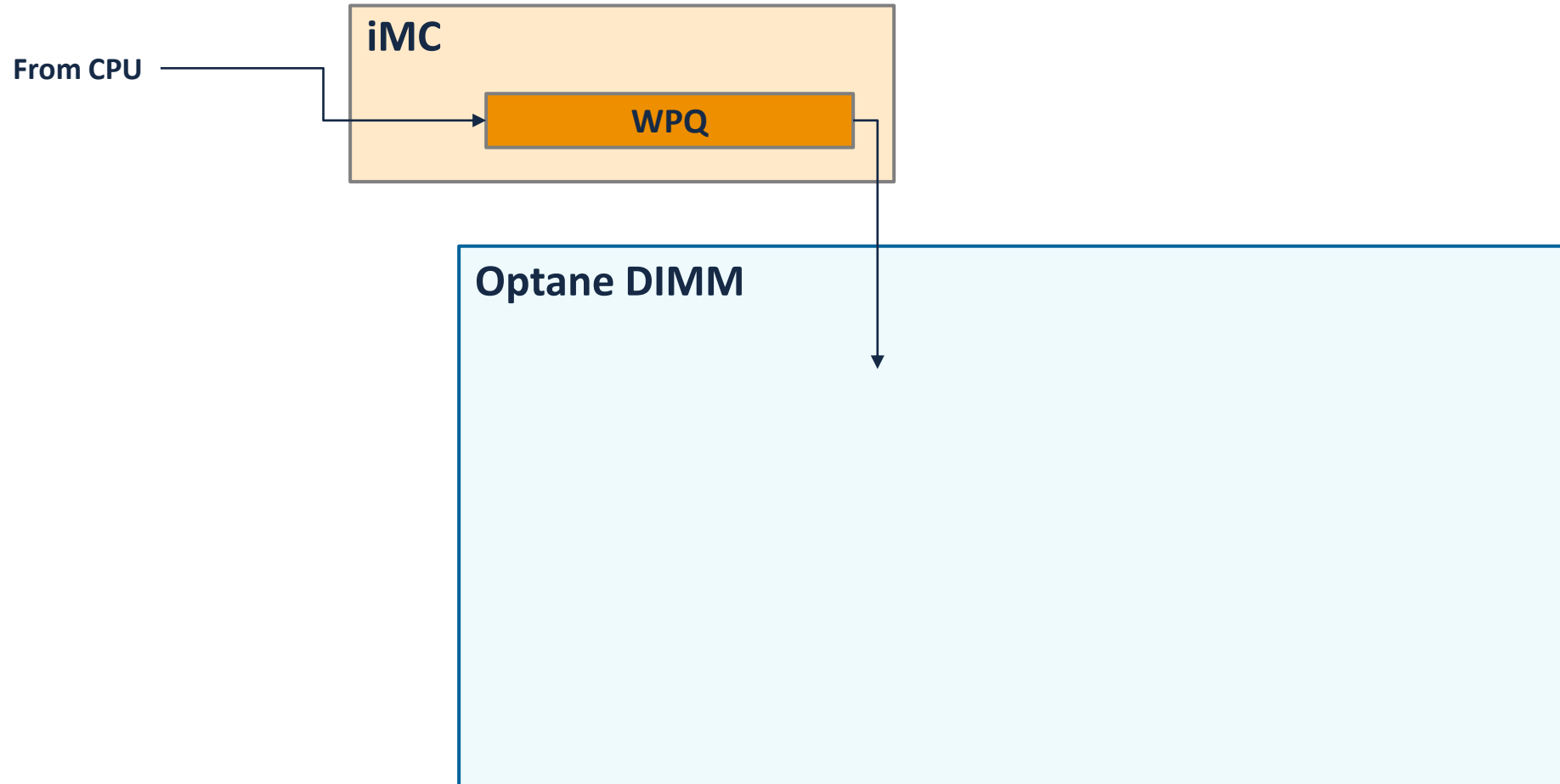
# Background: Optane Internals

Optane DIMM

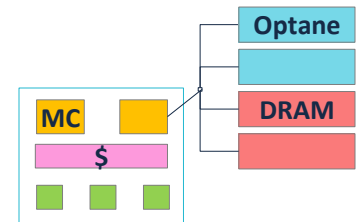
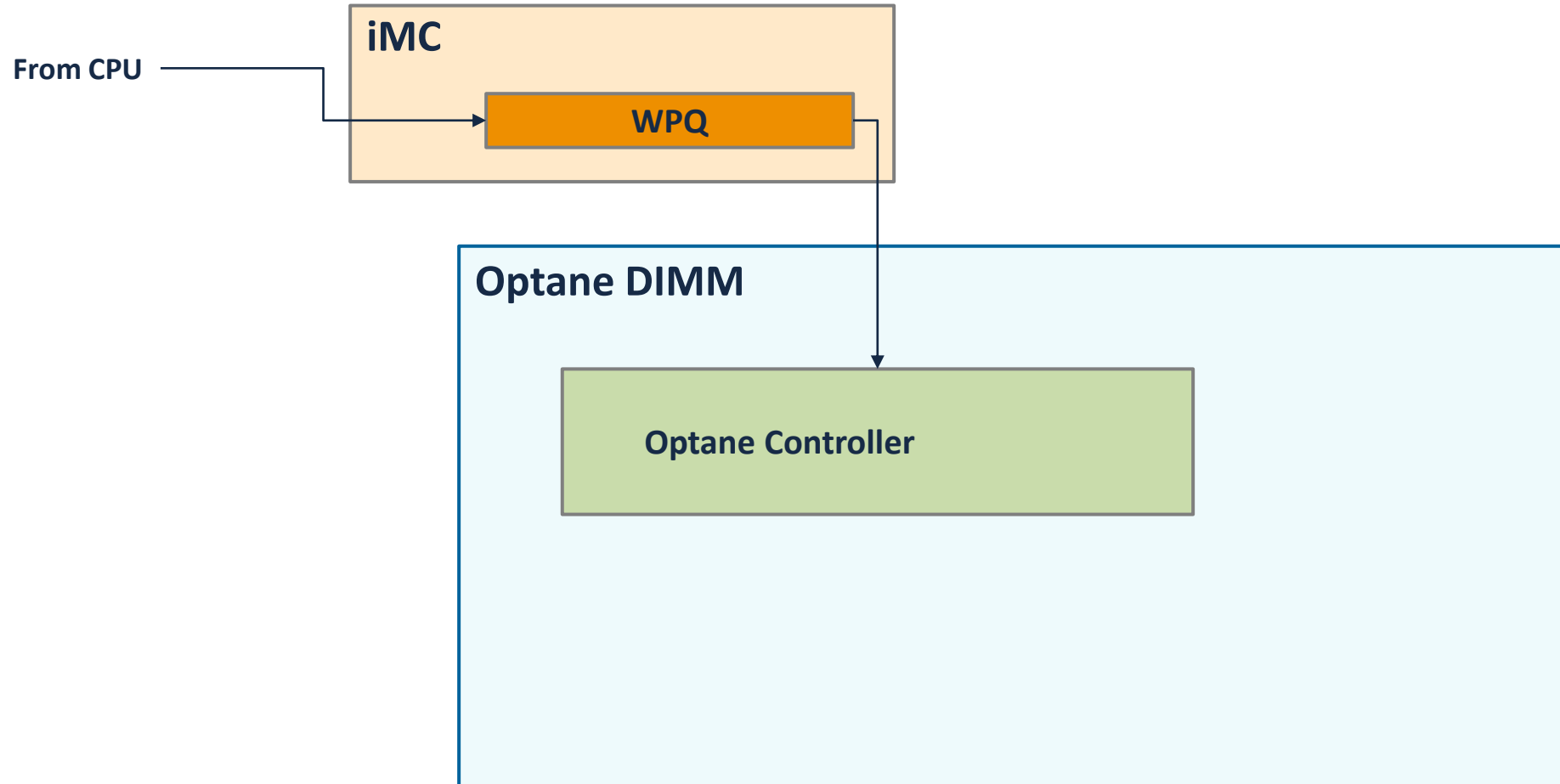




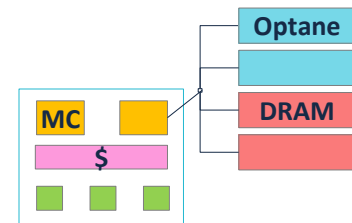
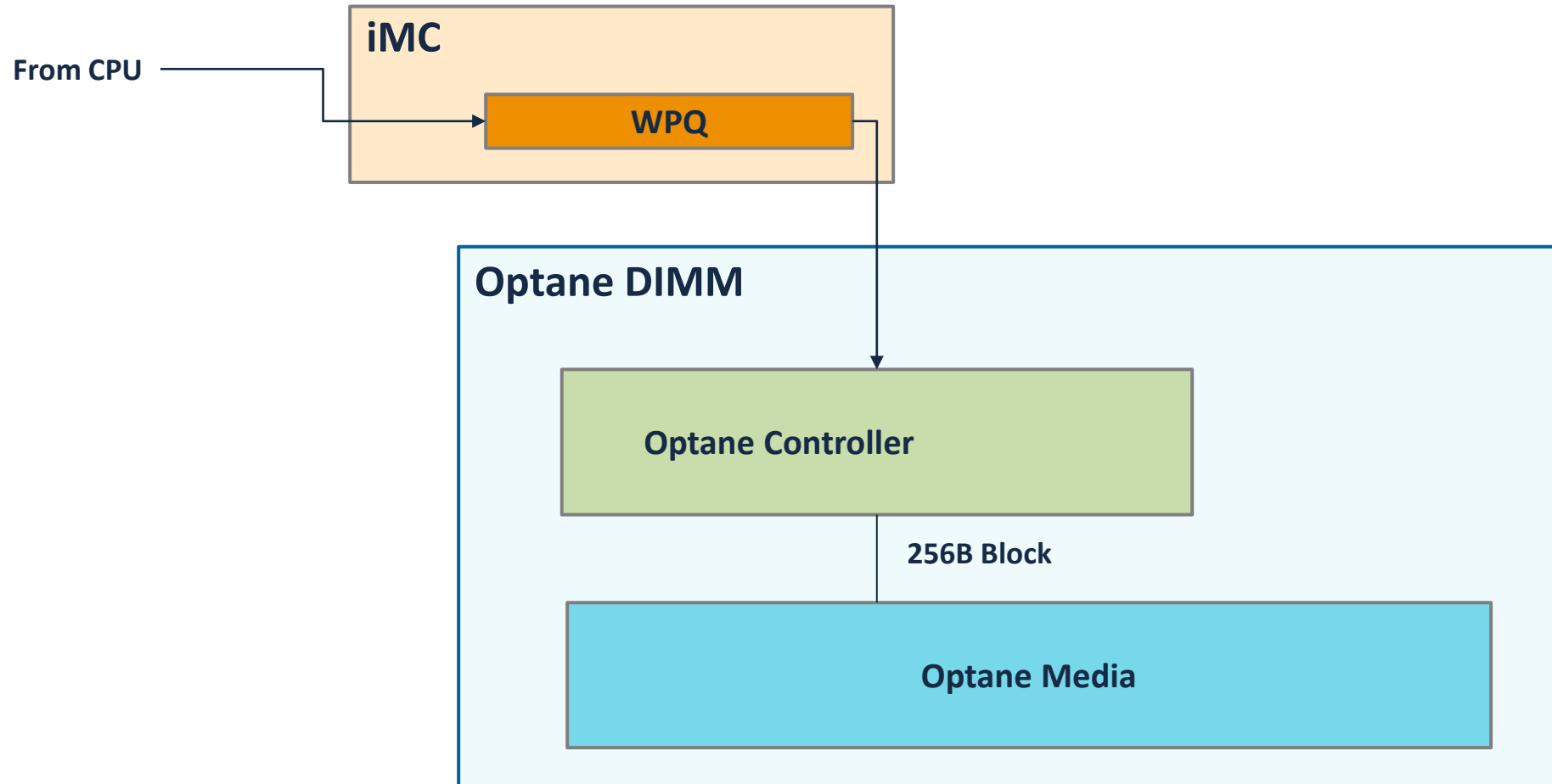
# Background: Optane Internals



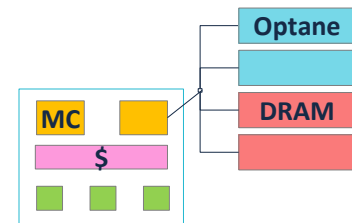
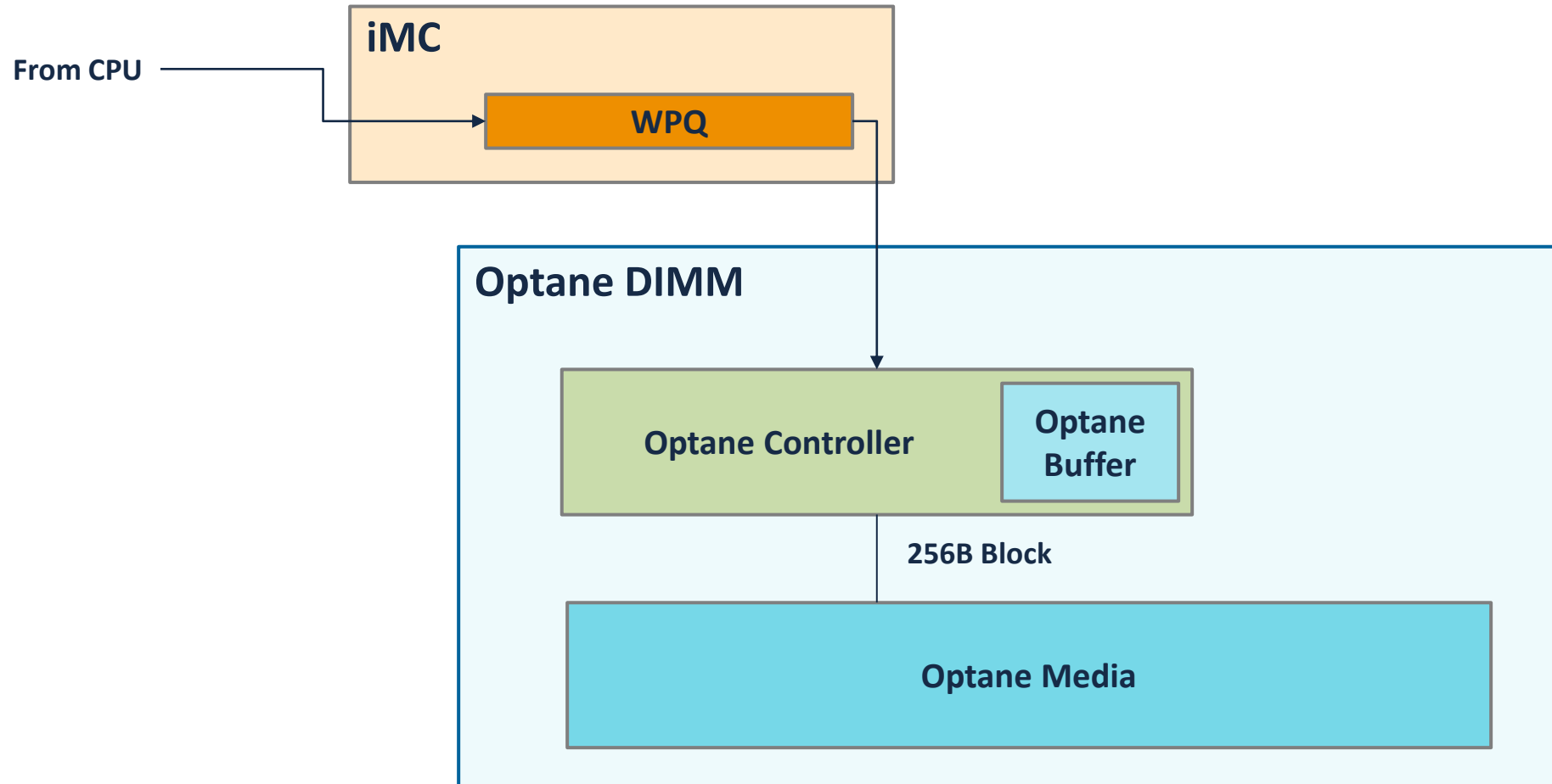
# Background: Optane Internals



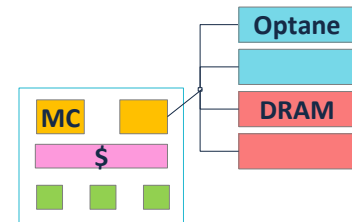
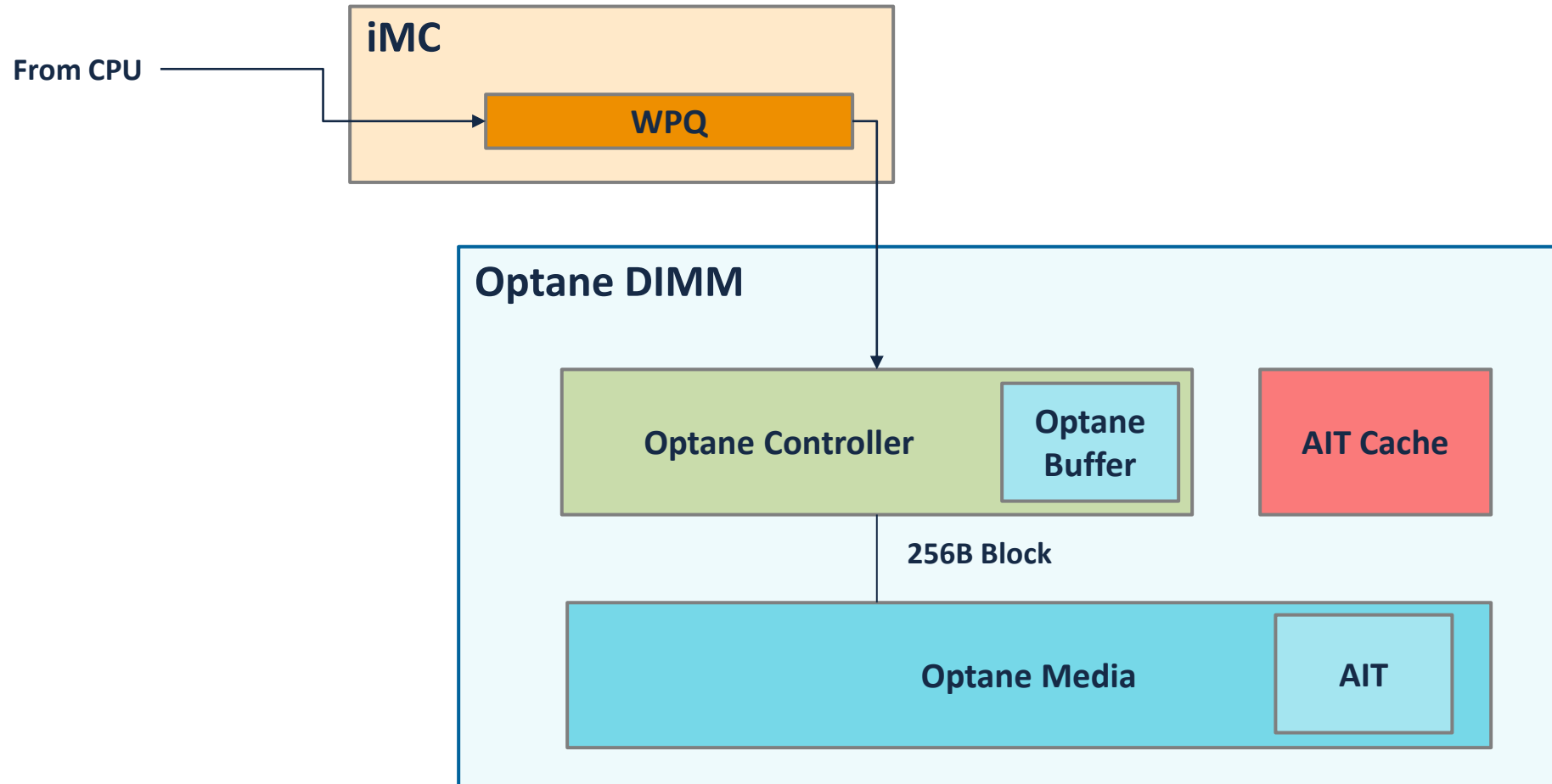
# Background: Optane Internals



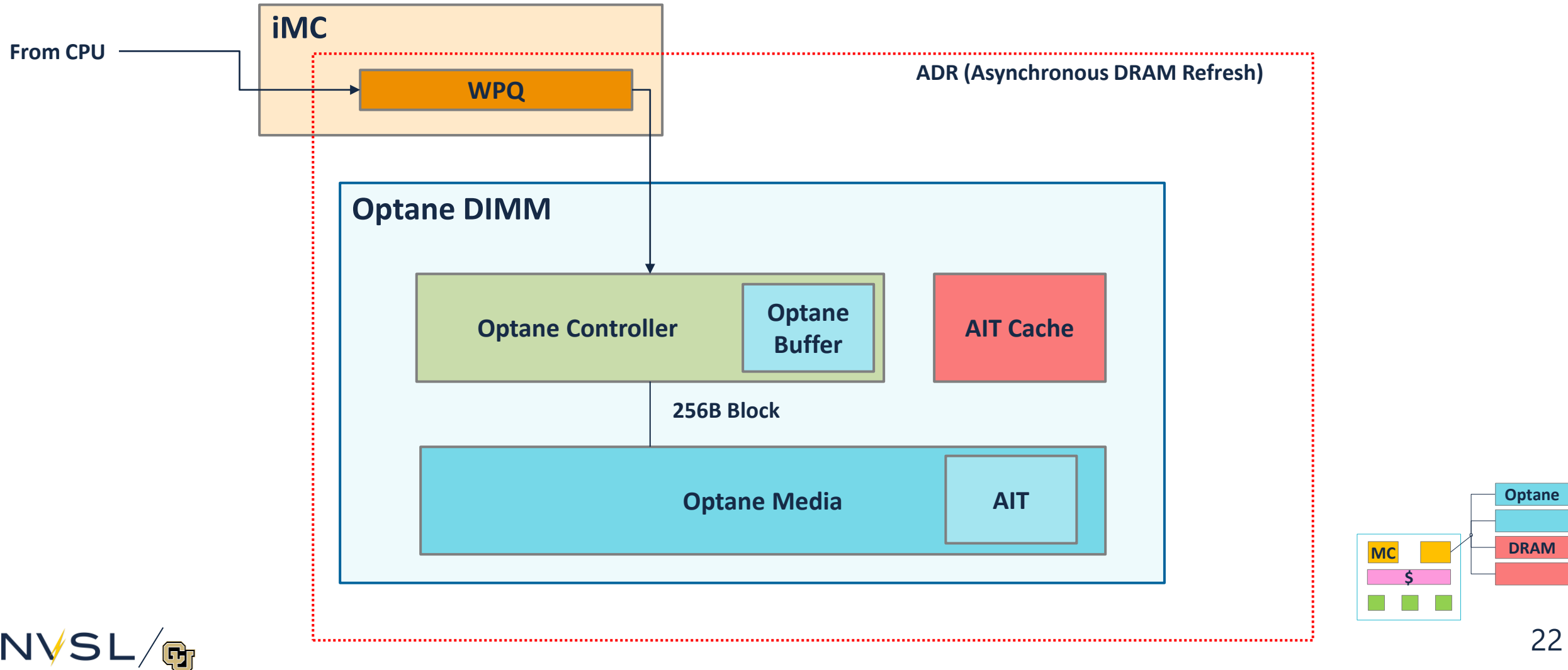
# Background: Optane Internals



# Background: Optane Internals

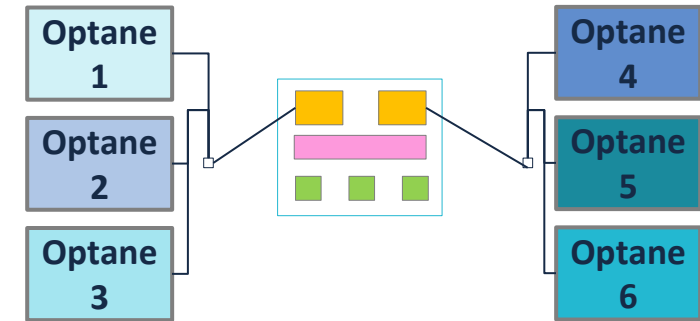


# Background: Optane Internals



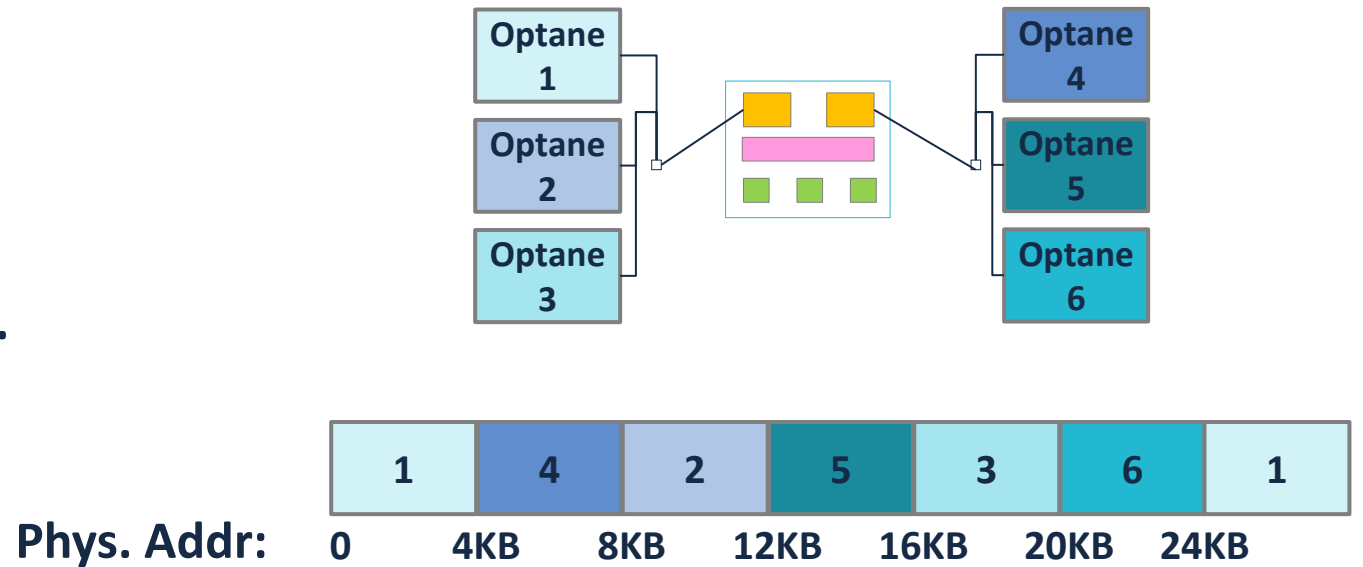
# Background: Optane Interleaving

- Optane is interleaved across NVDIMMs at 4KB granularity.



# Background: Optane Interleaving

- Optane is interleaved across NVDIMMs at 4KB granularity.





# Basics: How does Optane DC perform?

# Basics: Our Approach

- Microbenchmark sweeps across state space
  - Access Patterns (random, sequential)
  - Operations (read, ntstore, clflush, etc.)
  - Access/Stride Size
  - Power Budget
  - NUMA Configuration
  - Address Space Interleaving
- Targeted experiments
- Total: 10,000 experiments
  - <https://github.com/NVSL/OptaneStudy>

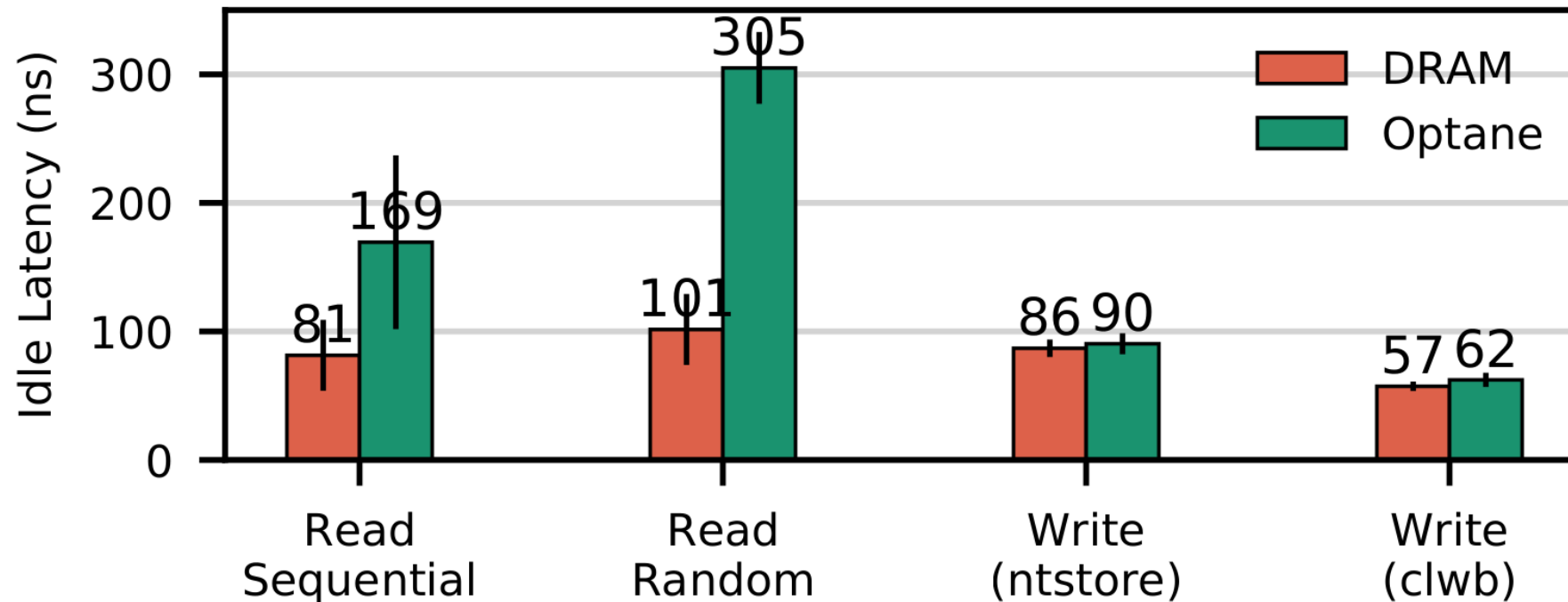


# Test Platform

- CPU
  - Intel Cascade Lake, 24 cores at 2.2 GHz in 2 sockets
  - Hyperthreading off
- DRAM
  - 32 GB Micron DDR4 2666 MHz
  - 384 GB across 2 sockets w/ 6 channels
- Optane
  - 256 GB Intel Optane 2666 MHz QS
  - 3 TB across 2 sockets w/ 6 channels
- OS
  - Fedora 27, 4.13.0

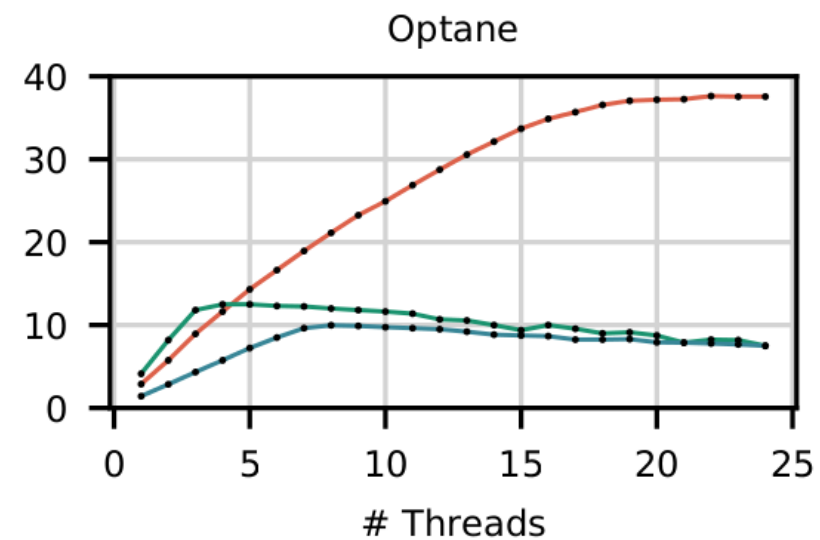
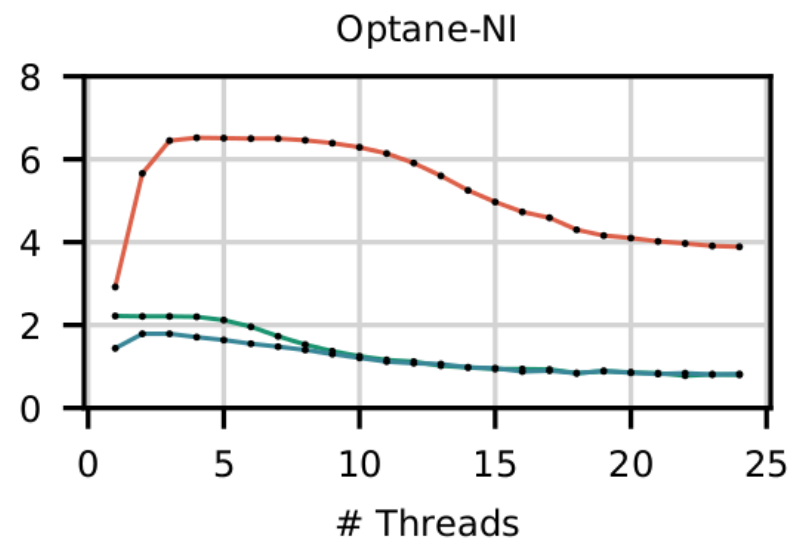
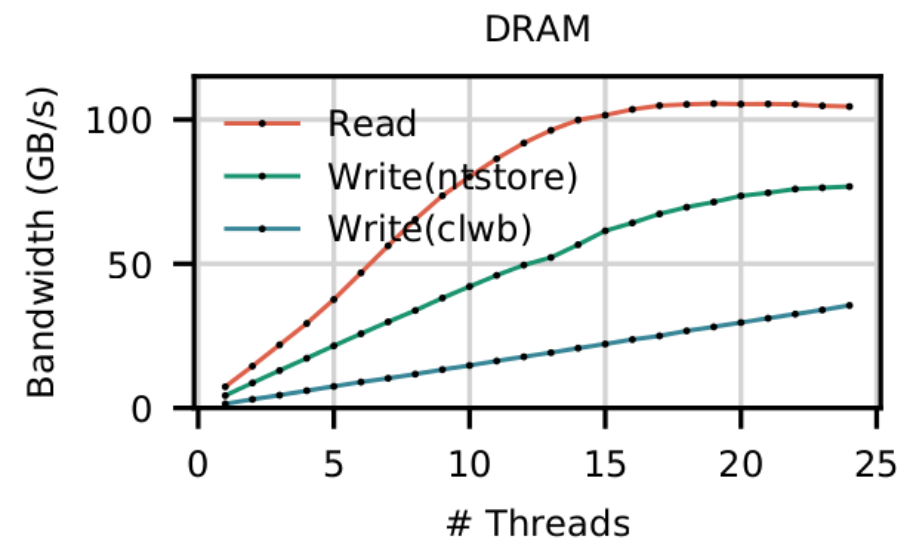
# Basics: Latency

- 2x -3x as slow as DRAM
- Write latency masked by ADR



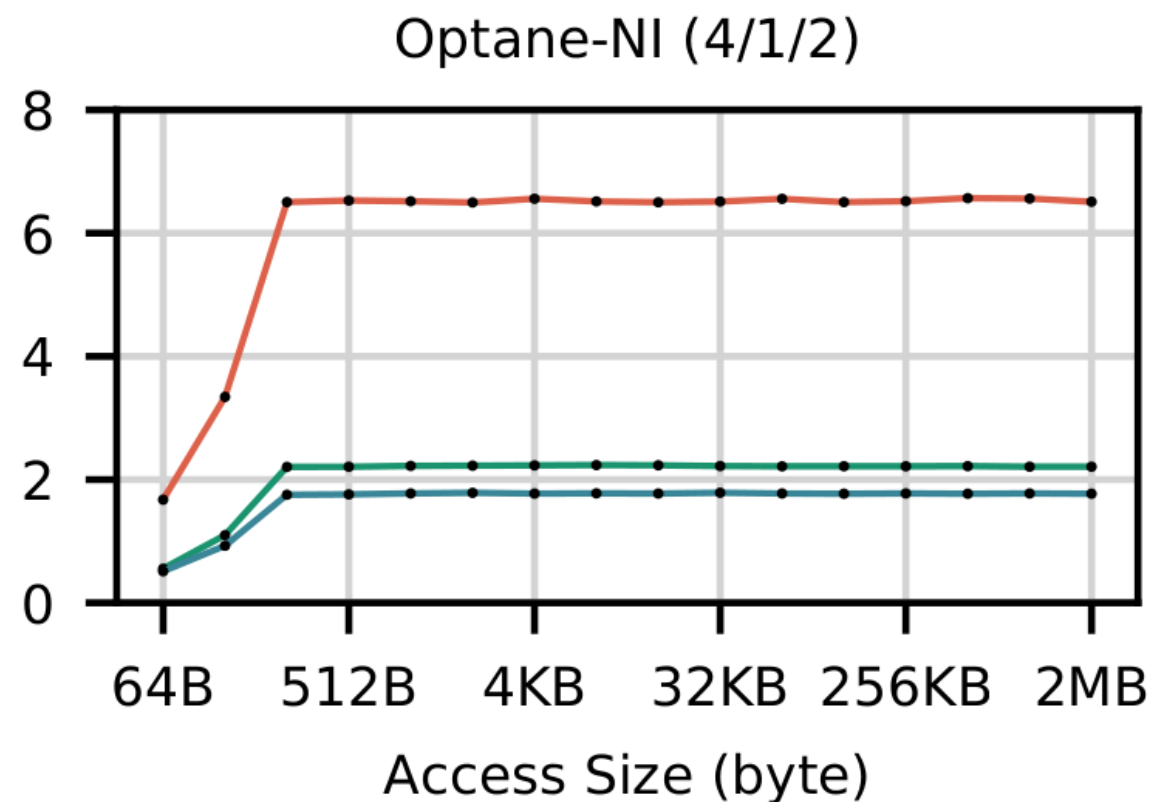
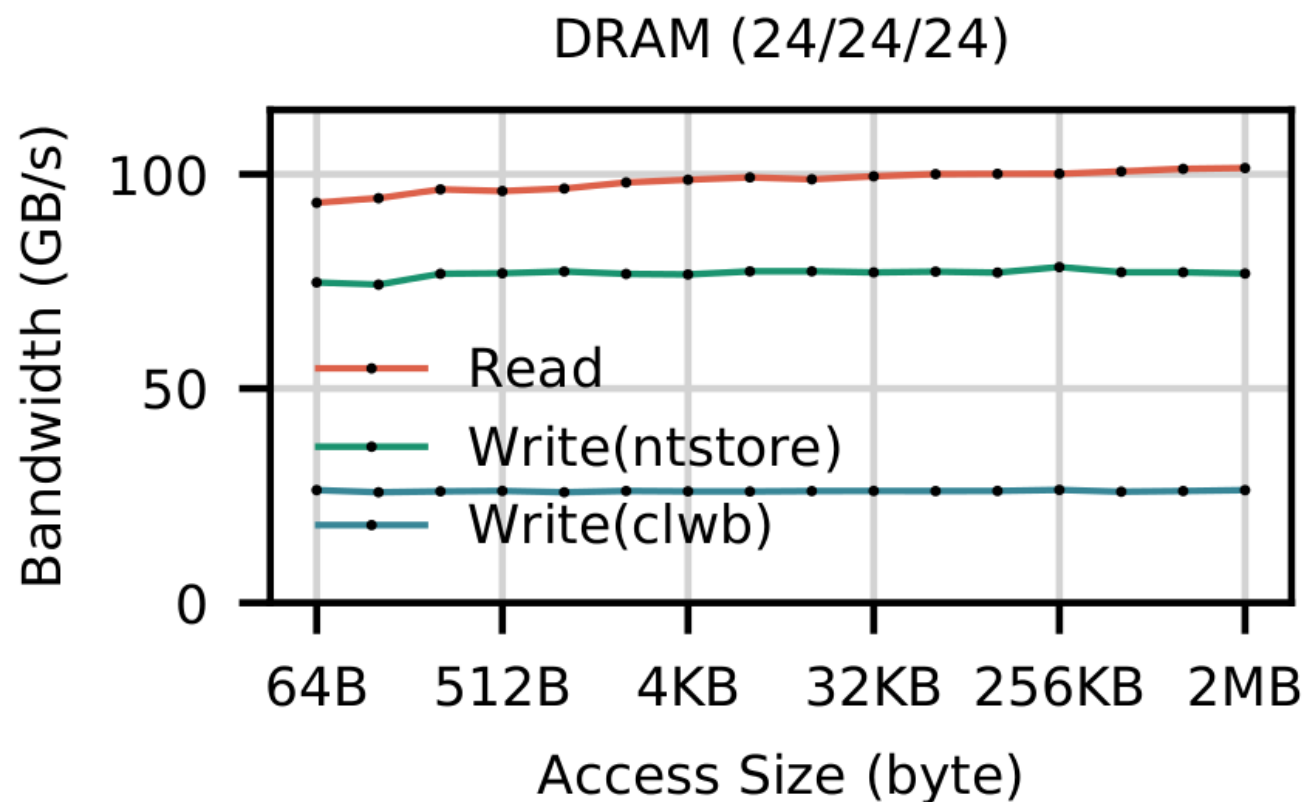
# Basics: Bandwidth

- ~Scalable reads, Non-scalable writes



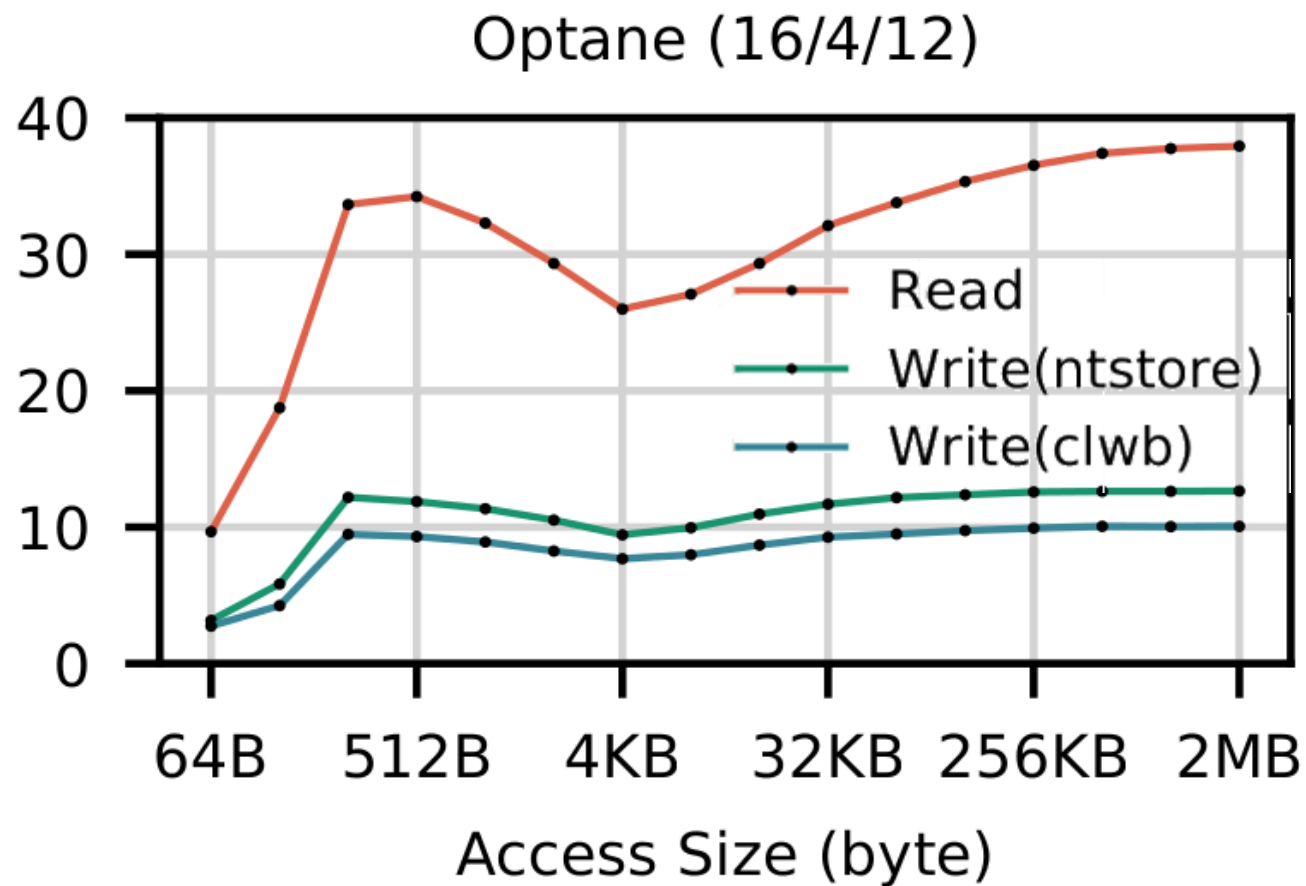
# Basics: Bandwidth

- Access size matters



# Basics: Bandwidth

- A mystery!



\*answer in ~10 minutes

# Lessons: What are Optane Best Practices?

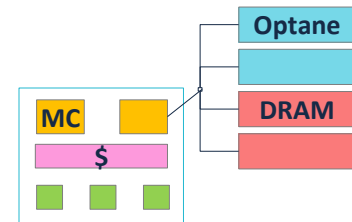
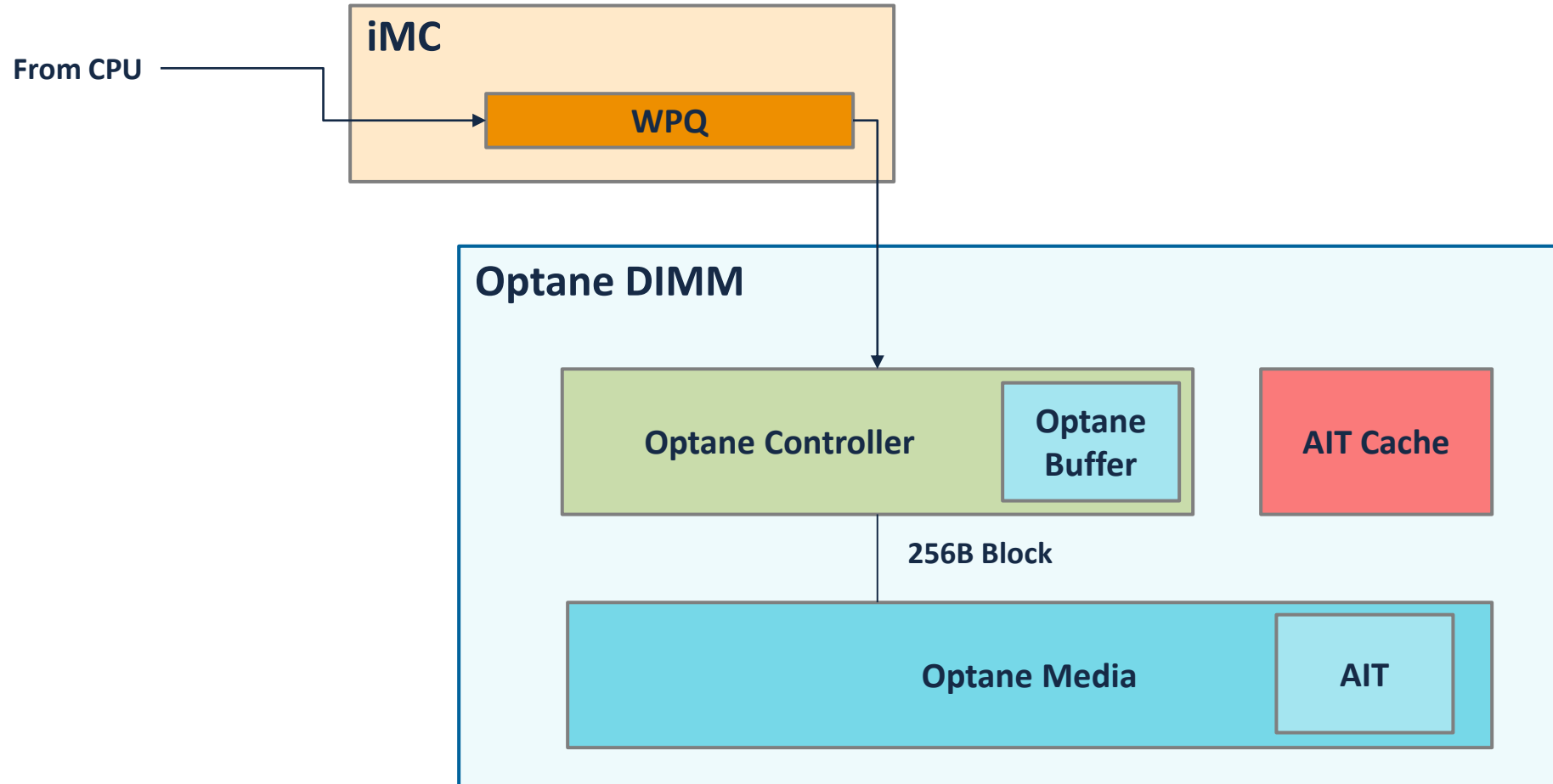


# Lessons: What are Optane Best Practices?

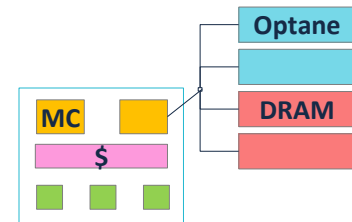
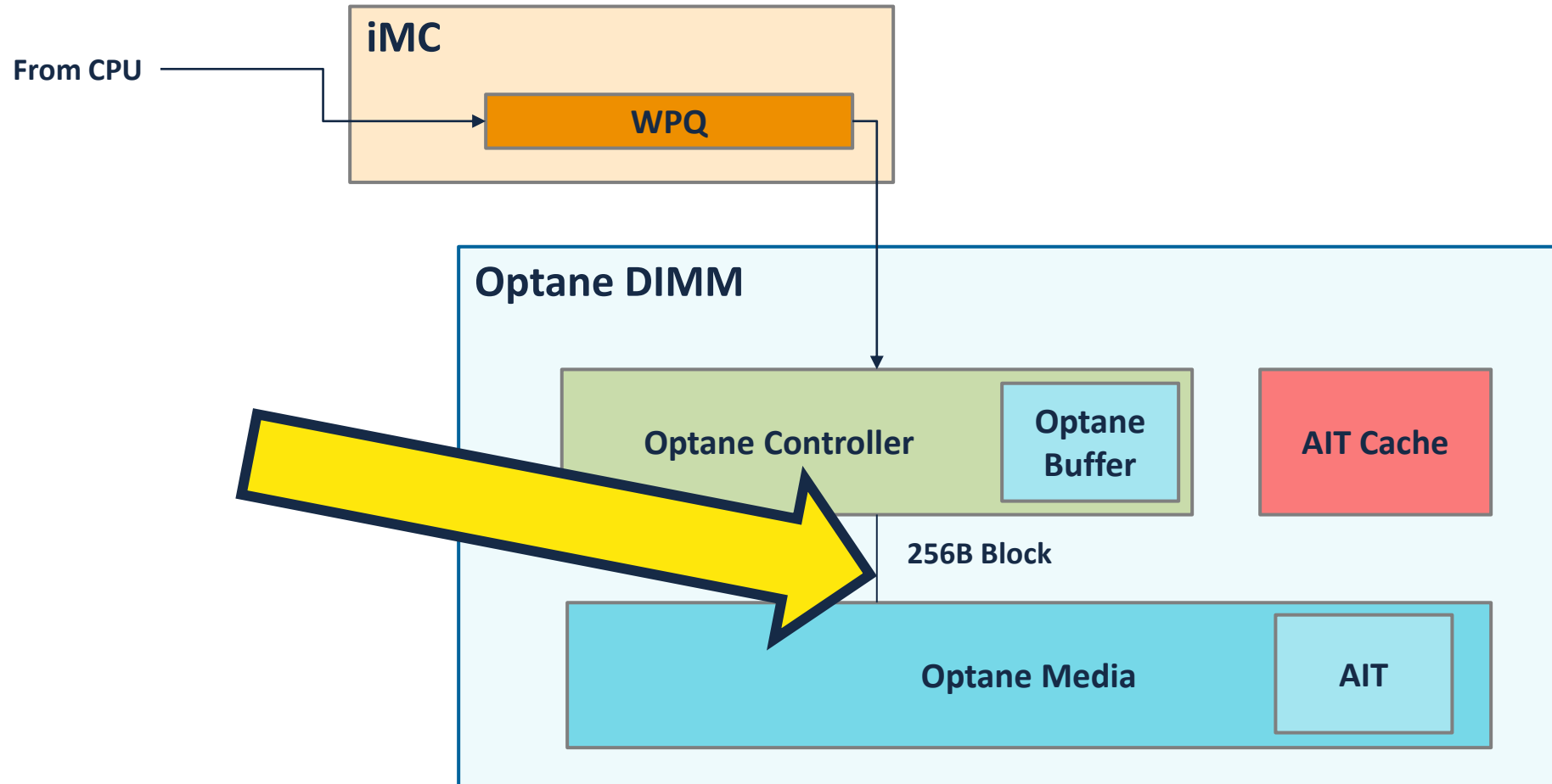
- Avoid small random accesses
- Use ntstores for large writes
- Limit threads accessing one NVDIMM
- Avoid mixed and multi-threaded NUMA accesses

# Lesson #1: Avoid small random accesses

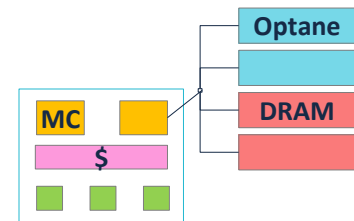
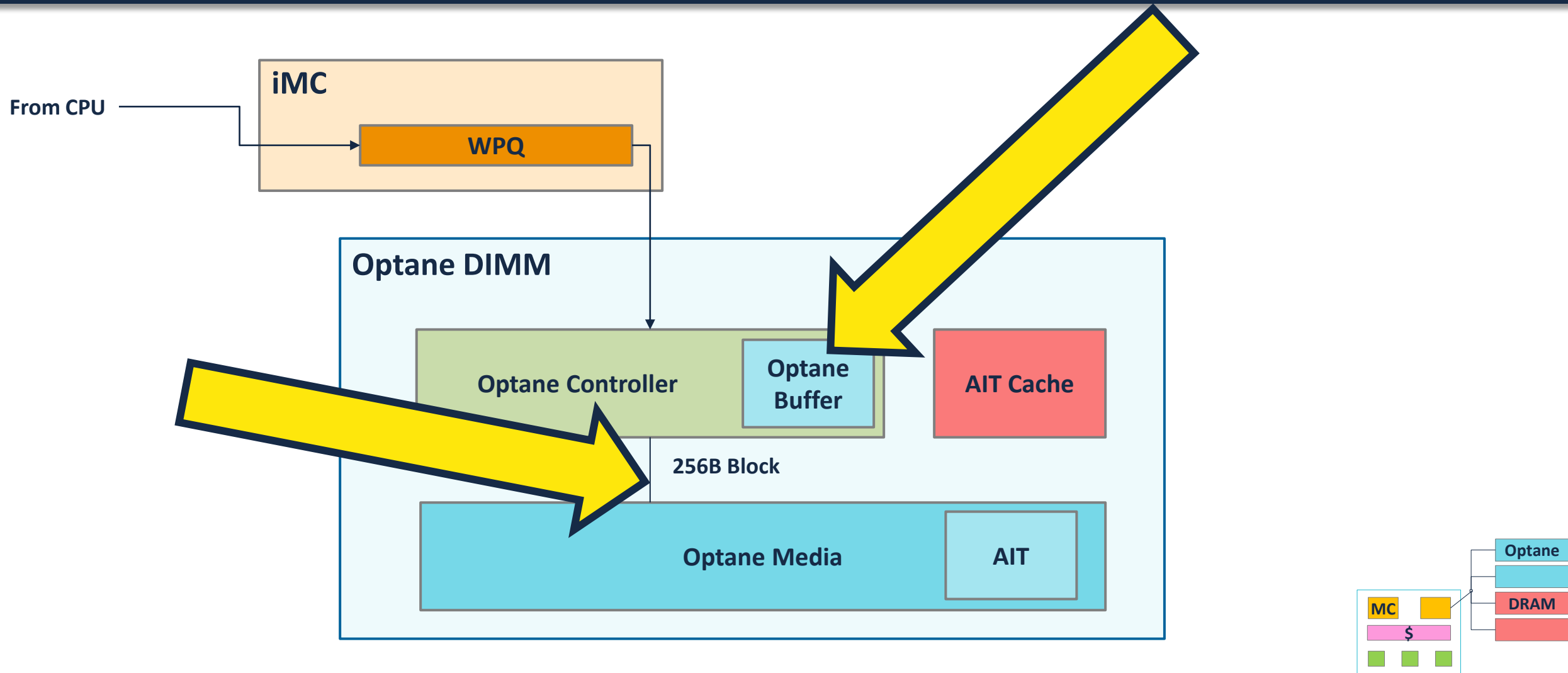
# Lesson #1: Avoid small random accesses



# Lesson #1: Avoid small random accesses

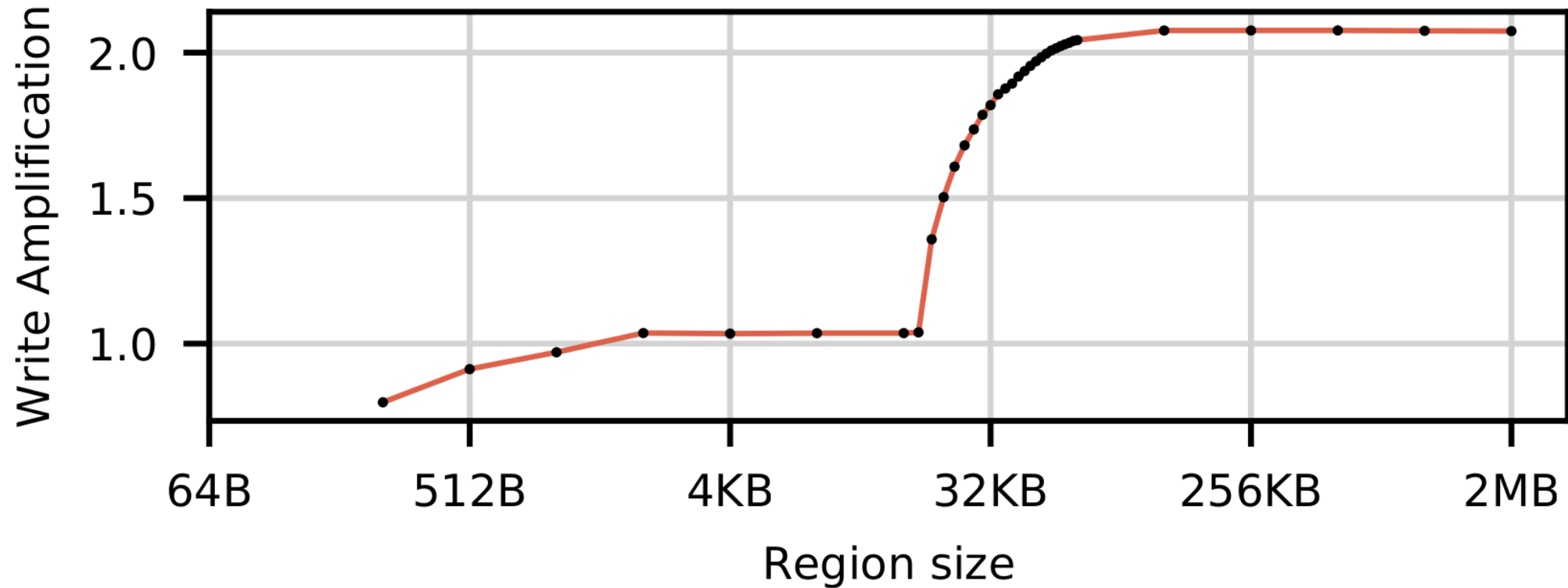


# Lesson #1: Avoid small random accesses



# Lessons: Optane Buffer Size

- Write amplification if working set is larger than Optane Buffer



# Lesson #1: Avoid small random accesses

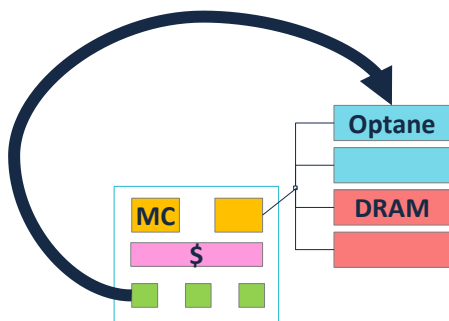
- Bad bandwidth with:
  - Small random writes (<256B)
  - Not tiny working set / NVDIMM (>16KB)
- Good bandwidth with
  - Sequential accesses

# Lesson #2: Use ntstores for large writes

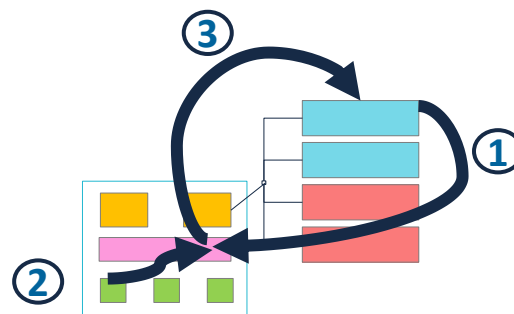


# Lessons: Store instructions

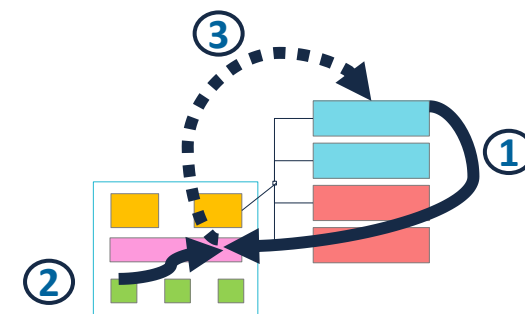
ntstore



store + clwb

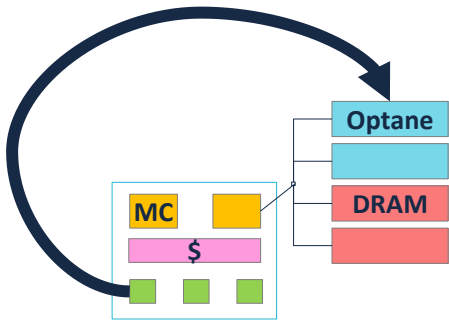


store

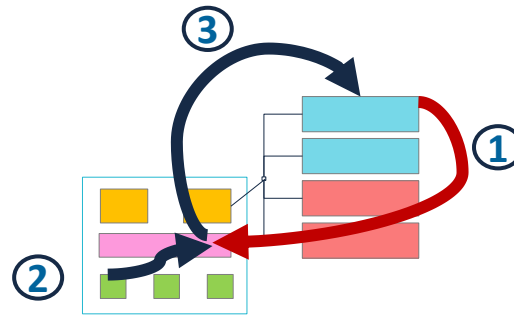


# Lessons: Store instructions

ntstore

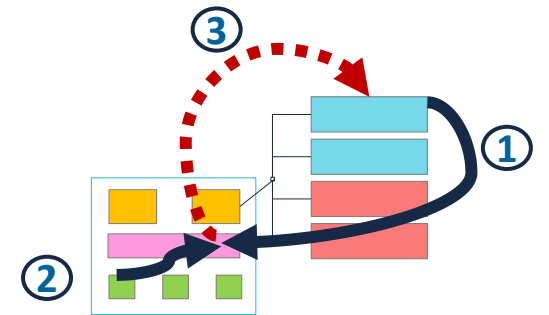


store + clwb



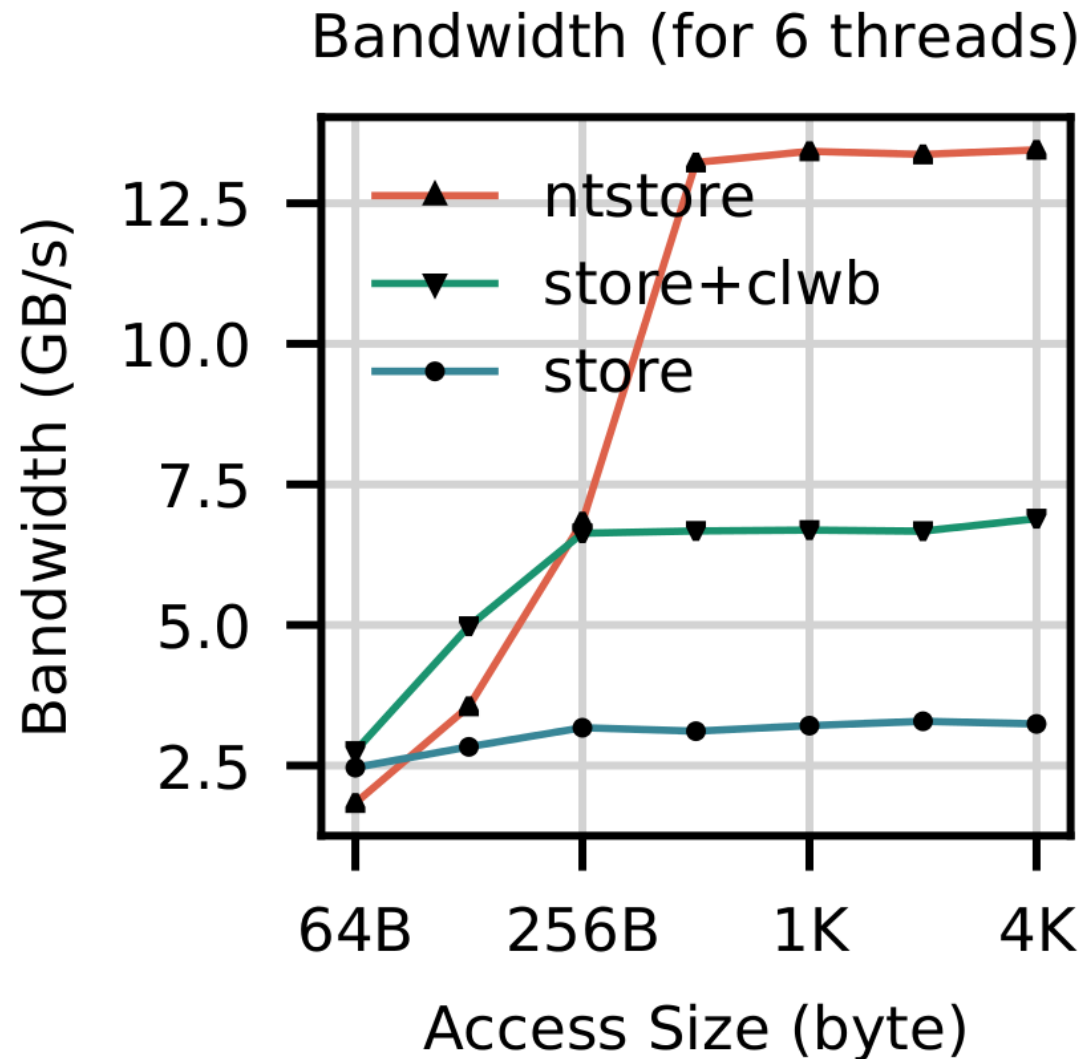
\*lost bandwidth

store



\*lost locality

# Lesson #2: Use ntstores for large writes



# Lesson #2: Use ntstores for large writes

- Non-temporal stores bypass the cache
  - Avoid cache-line read
  - Maintain locality

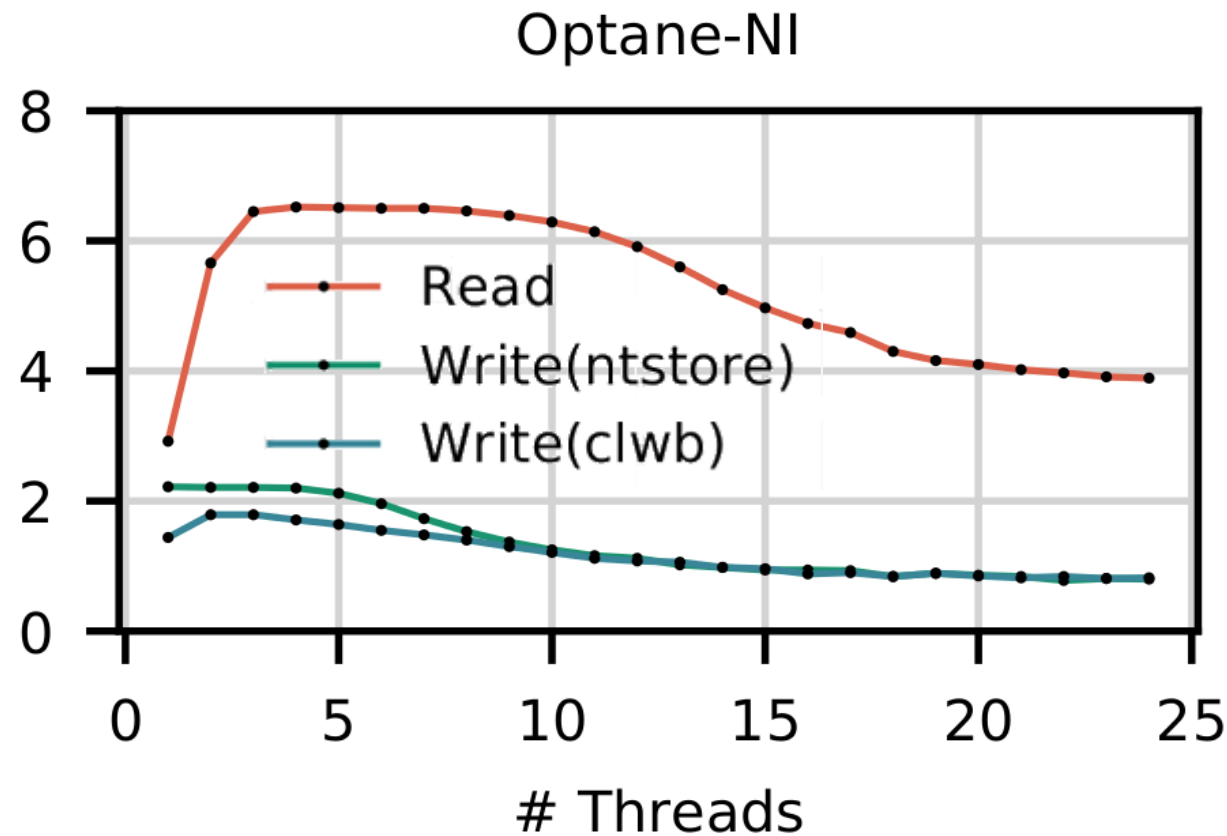
# Lesson #3: Limit threads accessing one NVDIMM

# Lesson #3: Limit threads accessing one NVDIMM

- Contention at Optane Buffer
- Contention at iMC

# Lessons: Contention at Optane Buffer

- More threads = access amplification = lower bandwidth



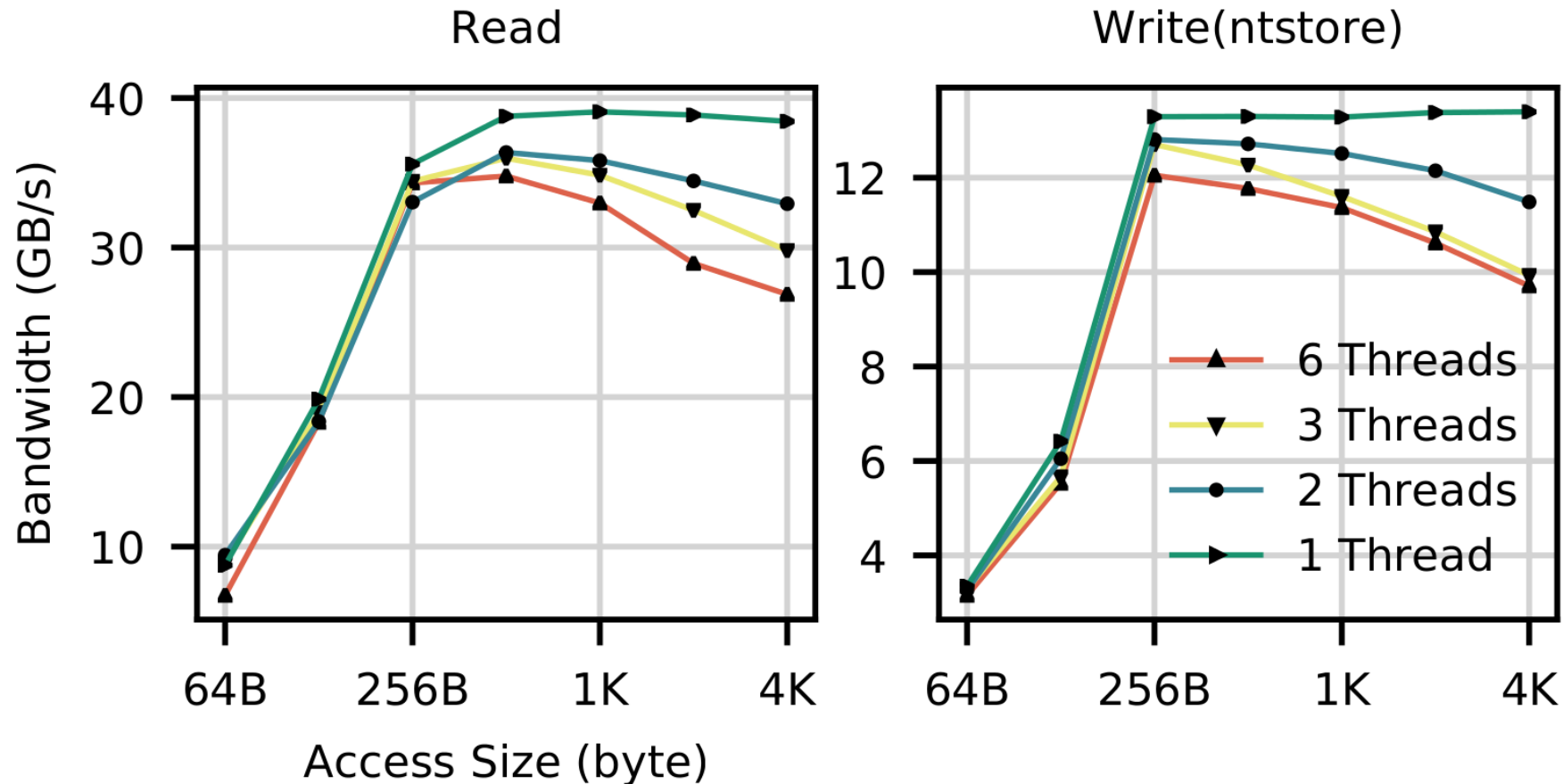
# Lessons: Contention at iMC

- iMCs aren't designed for slow, variable latency accesses
  - Short queues end up clogged



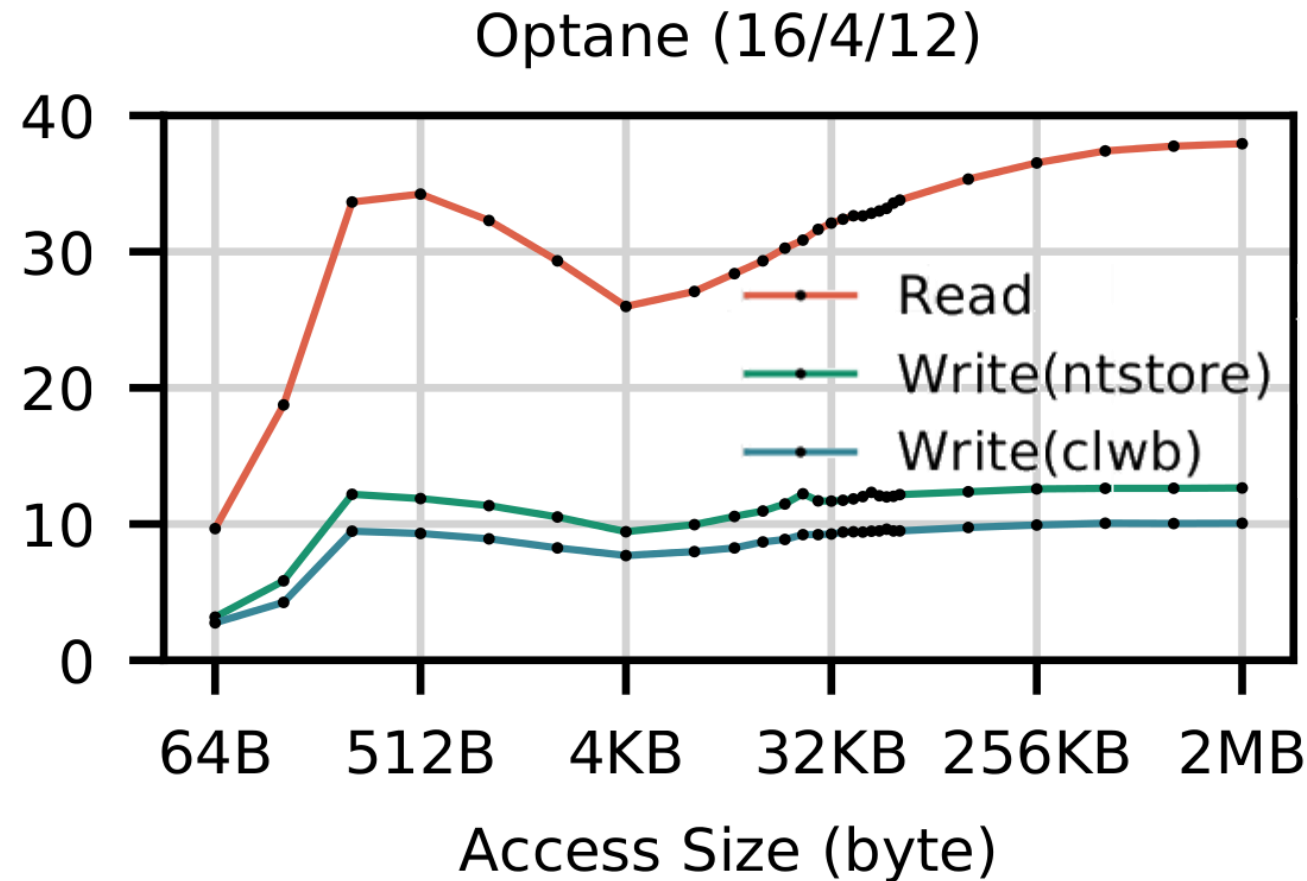
# Lessons: Contention at iMC

- Vary #threads/NVDIMM (total threads = 6)



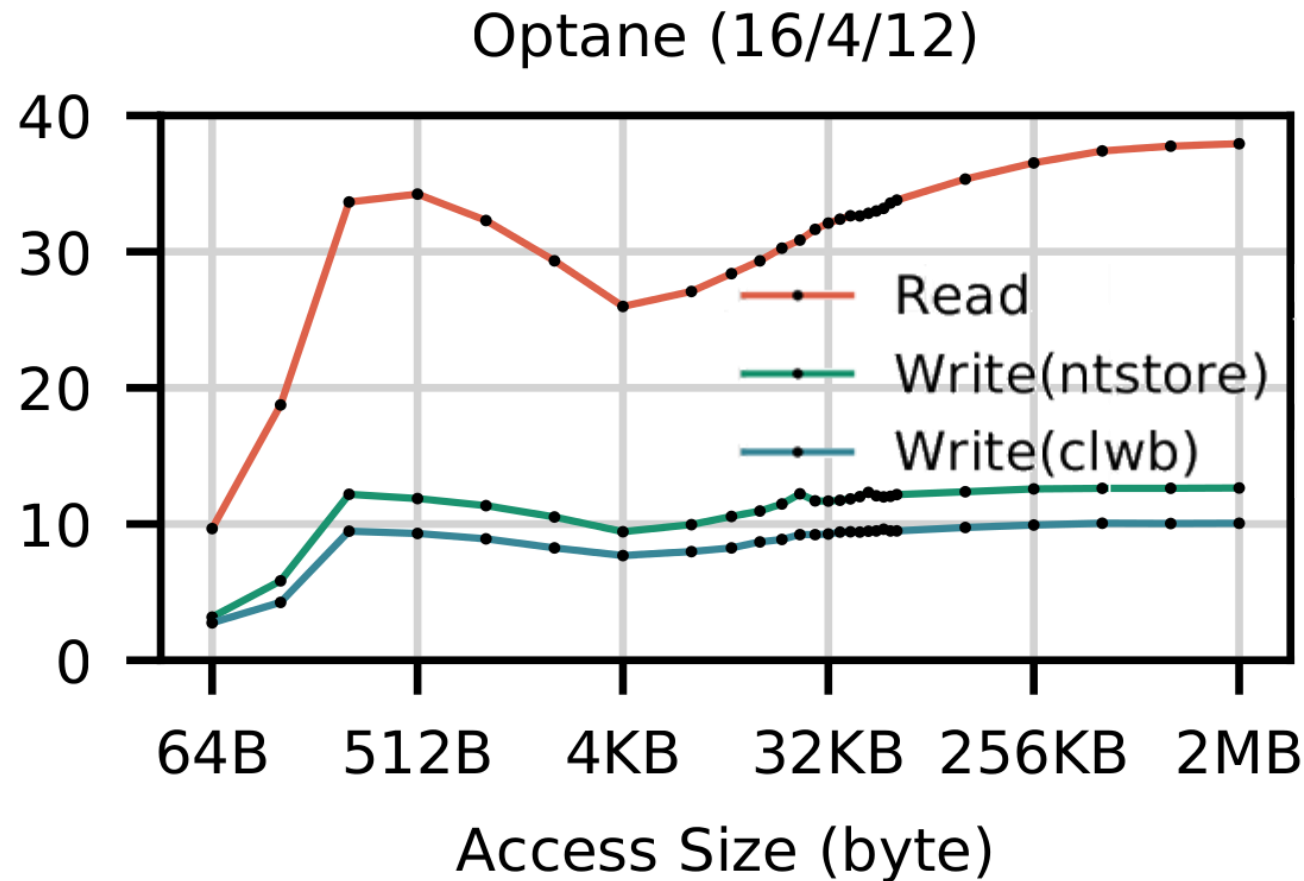
# Lesson: Contention at iMC

- iMC contention is largest when random access size = interleave size (4KB)



# Lesson: Contention at iMC

- iMC contention is largest when random access size = interleave size (4KB)
- iMC load is fairest when access size = #DIMMs x interleave size (24KB)



# Lesson #3: Limit threads accessing one NVDIMM

- Contention at Optane Buffer
  - Increase access amplification
- Contention at iMC
  - Lose bandwidth through uneven NVDIMM load
  - Avoid interleave aligned random accesses

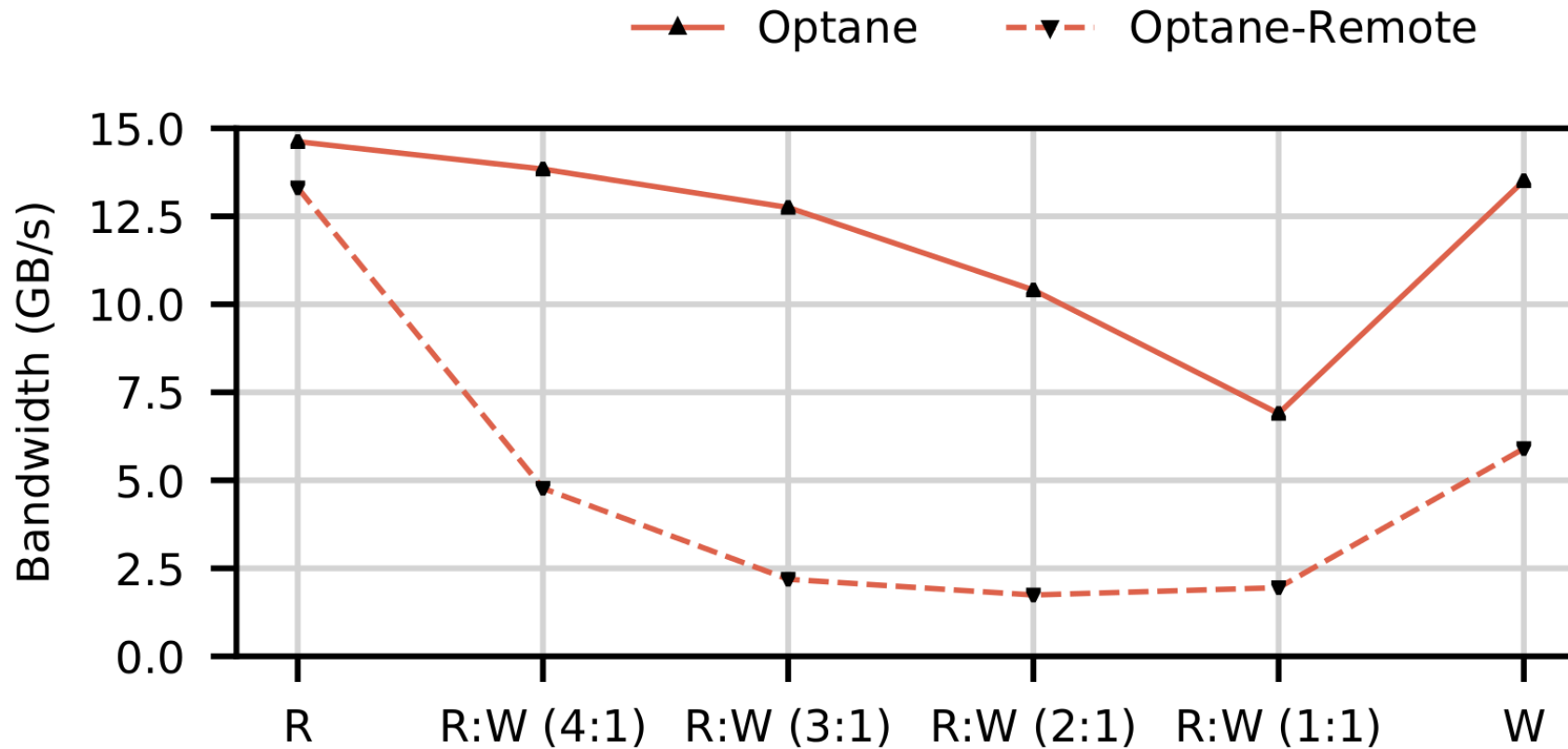
# Lesson #4: Avoid mixed and multi-threaded NUMA accesses

## Lesson #4: Avoid mixed and multi-threaded NUMA accesses

- NUMA effects impact Optane more than DRAM

# Lesson #4: Avoid mixed and multi-threaded NUMA accesses

- NUMA effects impact Optane more than DRAM
- R/W ratio is more important



# Lesson #4: Avoid mixed and multi-threaded NUMA accesses



# Conclusion

# Conclusion

- Not Just Slow Dense DRAM
- Slower media
  - > More complex architecture
  - > Second-order performance anomalies
  - > Fundamentally different
- Max performance is tricky
  - Avoid small random accesses
  - Use ntstores for large writes
  - Limit threads accessing one NVDIMM
  - Avoid mixed and multi-threaded NUMA accesses

