Exploring the Design Space of Page Management for Multi-Tiered Memory Systems

Jonghyeon Kim, Wonkyo Choe, Jeongseob Ahn

AJOU UNIVERSITY

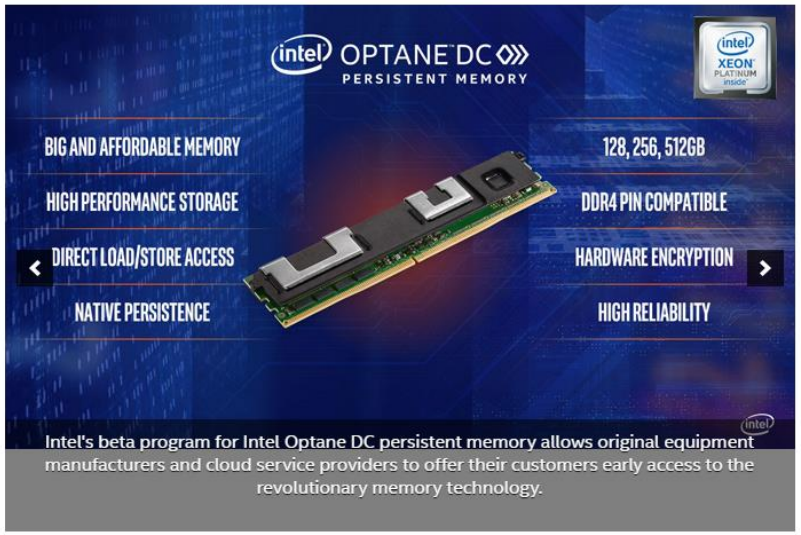
2021 USENIX Annual Technical Conference

Large Memory Systems

intel newsroom | Top News Sections | News By Category | All News | Search Newsroom

News Byte
October 30, 2018
Contact Intel PR

Intel Optane DC Persistent Memory Readies for Widespread Deployment



Samsung Unveils Industry-First Memory Module Incorporating New CXL Interconnect Standard

Korea on May 11, 2021 Audio Share

DDR5 DRAM-based memory module is designed to meet the high-performance demands of data-intensive applications including AI and HPC

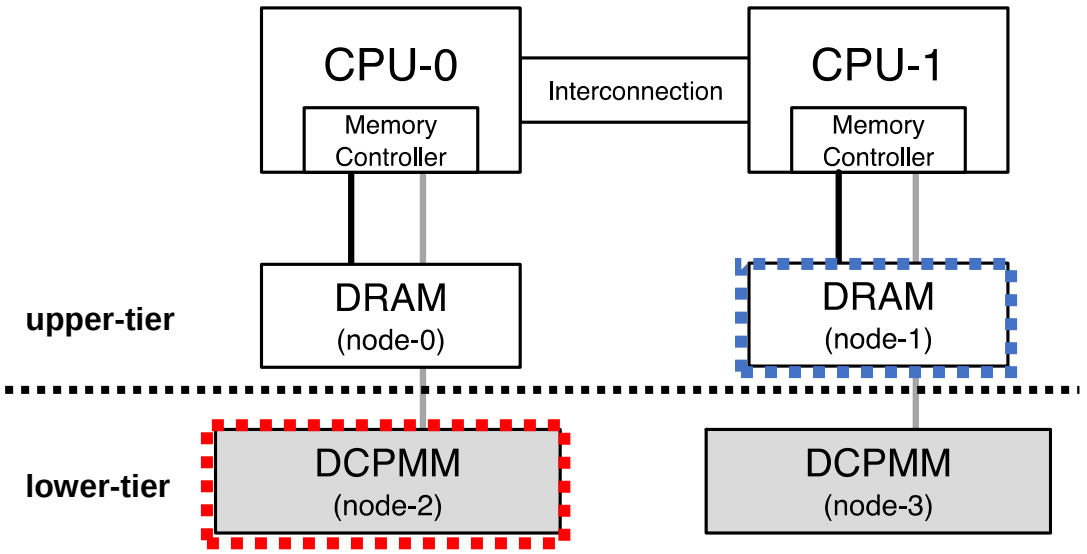
CXL interface enables memory capacity to scale to the terabyte level and substantially reduces system latency

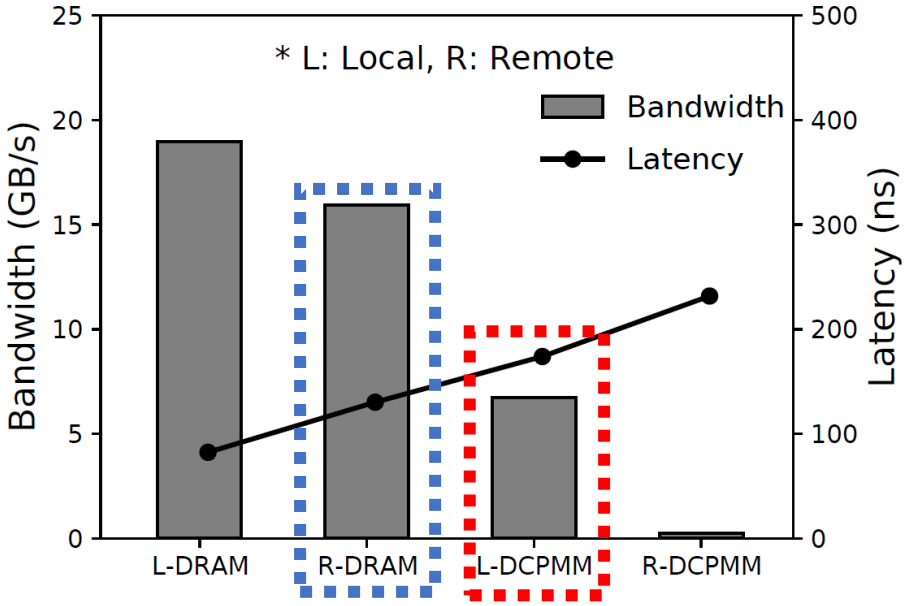
This study utilizes Intel Optane DC Persistent Memory Module (DCPMM) as slow memory

Large Memory Systems

- Emerging memory technologies have been widely used to build large memory systems
- However, such storage class memory (SCM) provide lower performance than DRAM
  - High latency and low bandwidth, compared to DRAM
- Future large memory systems will offer a form of tiered memory architecture with DRAM and SCM

Multi-tiered Memory Systems

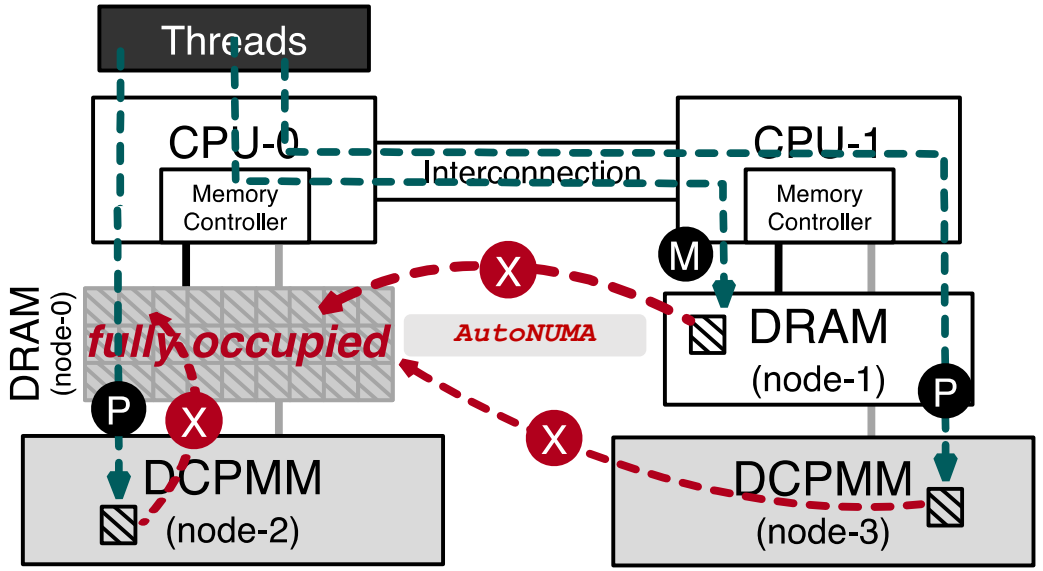




- DCPMM can be exposed as main memory
  - Since Linux kernel v5.0
  - Treated as a CPU-less NUMA node

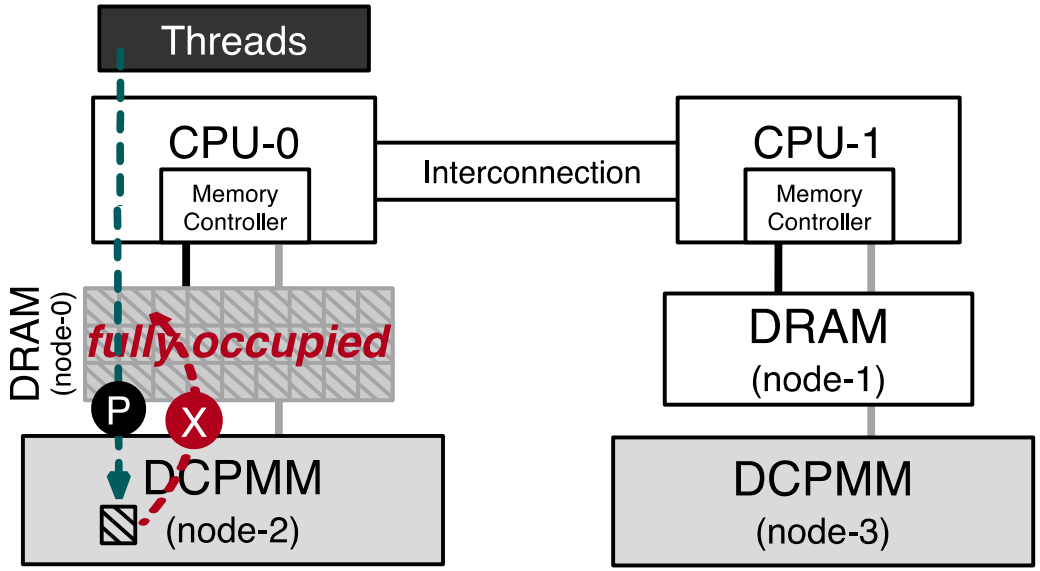
Does the memory management in Linux work well?

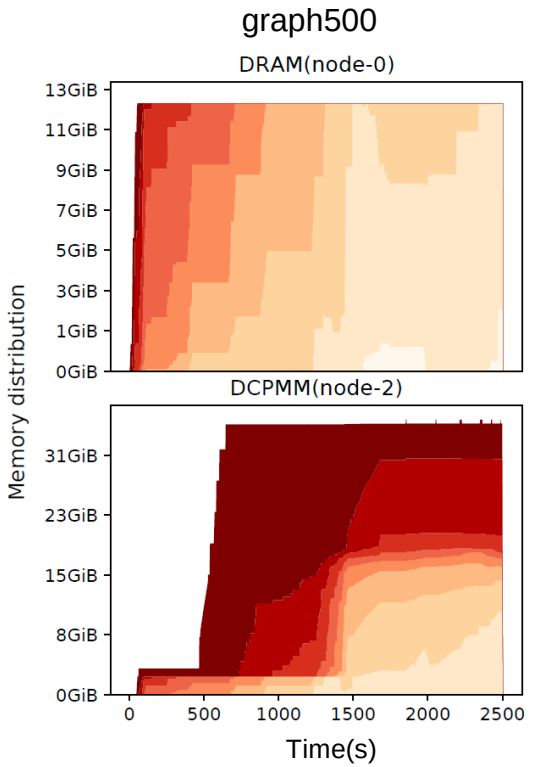
Need for Page Placement in Multi-Tiered Memory



Problem 1.
In current Linux, page movement (promotion or migration) is not allowed when the target node has no free space

Need for Page Reclamation in Multi-Tiered Memory





less effectiveness of upper-tier memory!

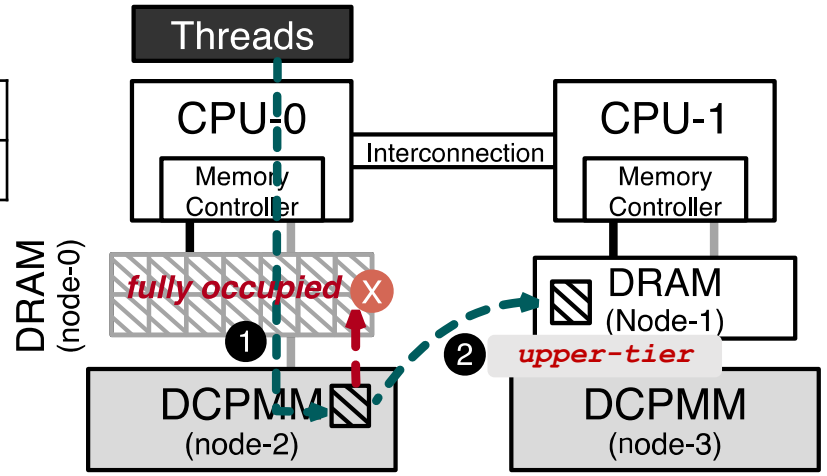
Infrequently accessed

frequently accessed

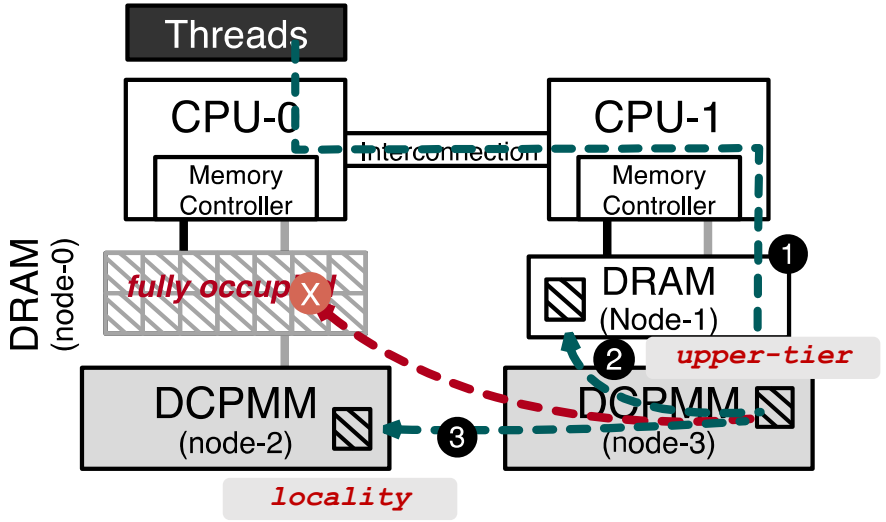
Problem 2.
Infrequently accessed pages can reside in the upper-tier memory while more frequently access pages stay in the lower-tier memory

Exploiting Access Tier and Locality

Tier	O
Locality	X



<Page promotion>



<Page promotion or migration>

Tier	O
Locality	X

OR

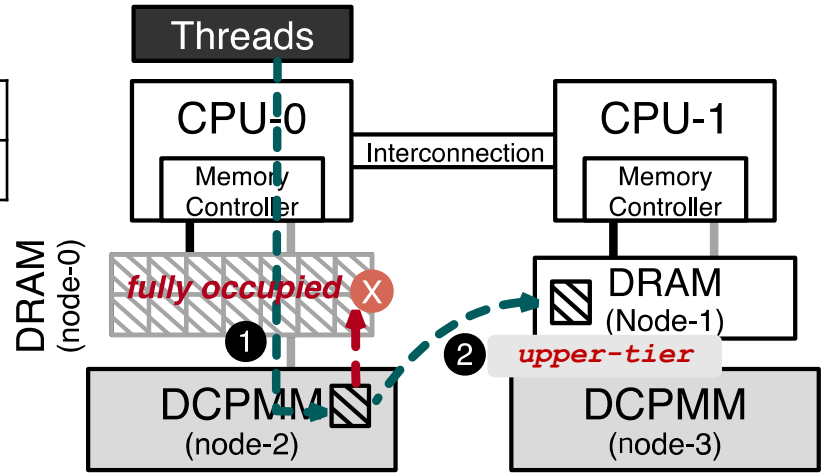
Tier	X
Locality	O

Conservative Promotion or Migration → AutoTiering-CPM

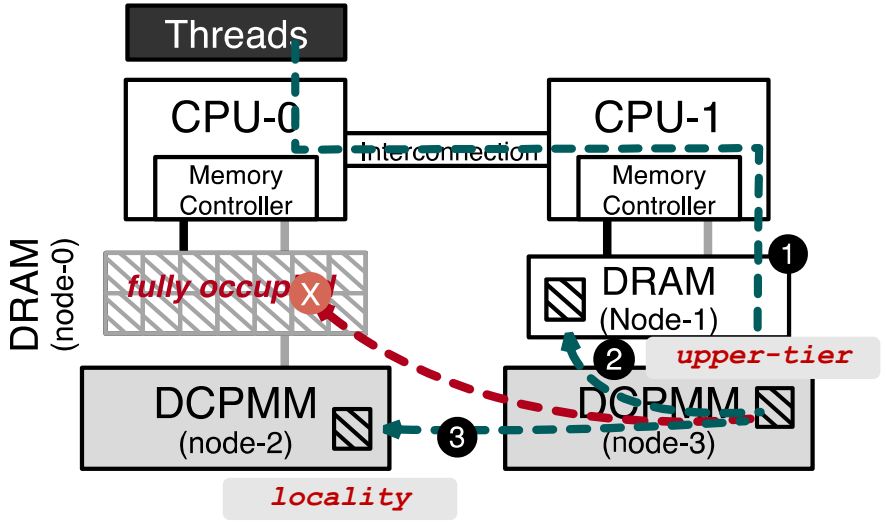
We provide alternatives for page migration failure due to fully occupied target memory node

Exploiting Access Tier and Locality

Tier	O
Locality	X



<Page promotion>



<Page promotion or migration>

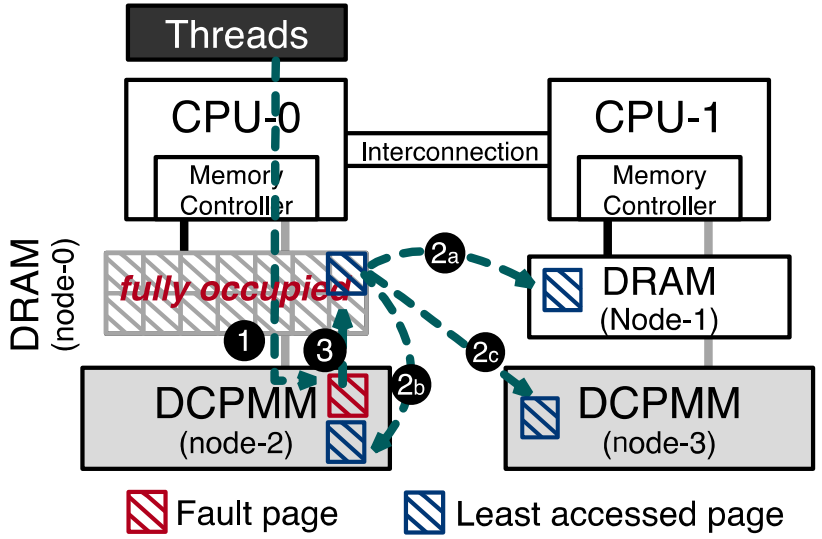
Tier	O
Locality	X

OR

Tier	X
Locality	O

However, the upper-tier (DRAM) memory can still hold infrequently accessed data while frequently used pages reside in the lower-tier (DCPMM) memory

Demotion of Least Accessed Page

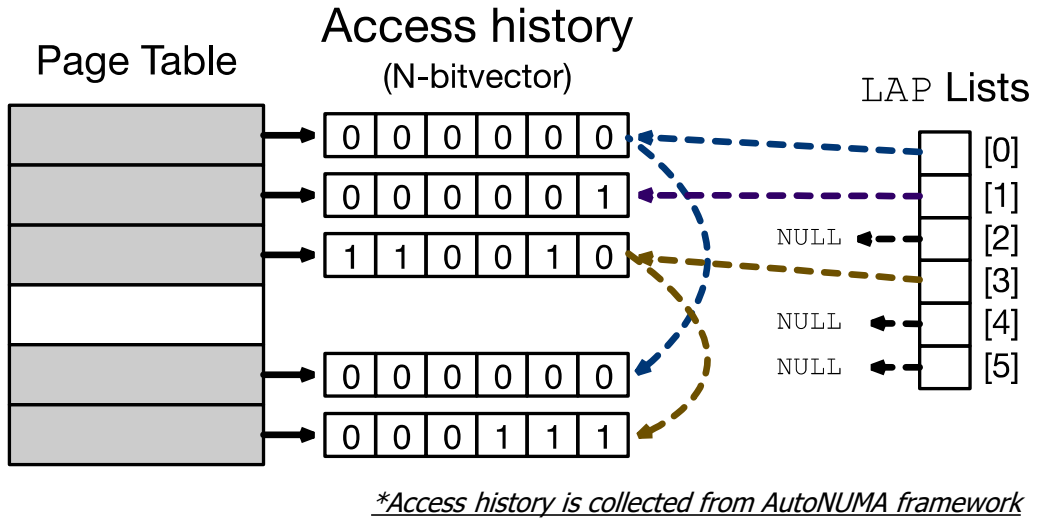


<Page demotion and promotion>

Opportunistic Promotion or Migration → `AutoTiering-OPM`

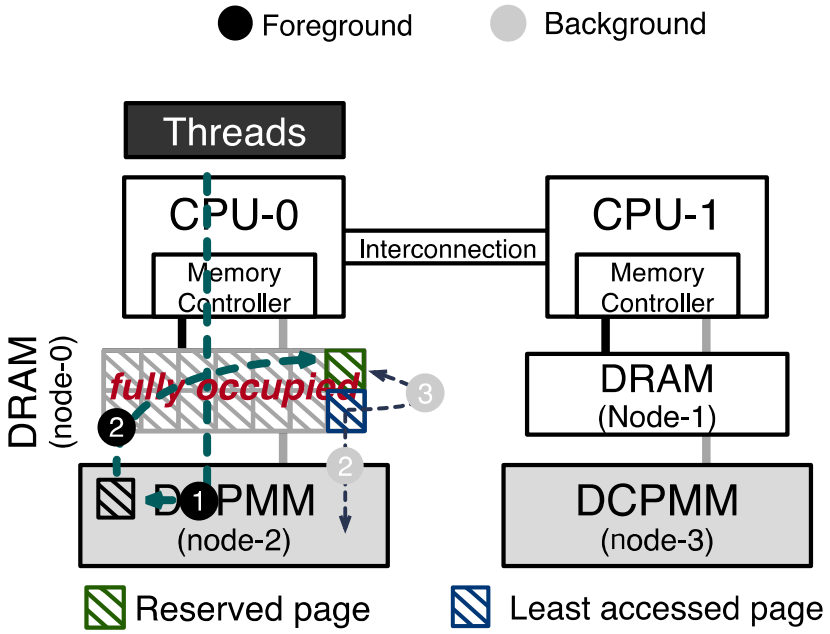
Estimating Least Accessed Page

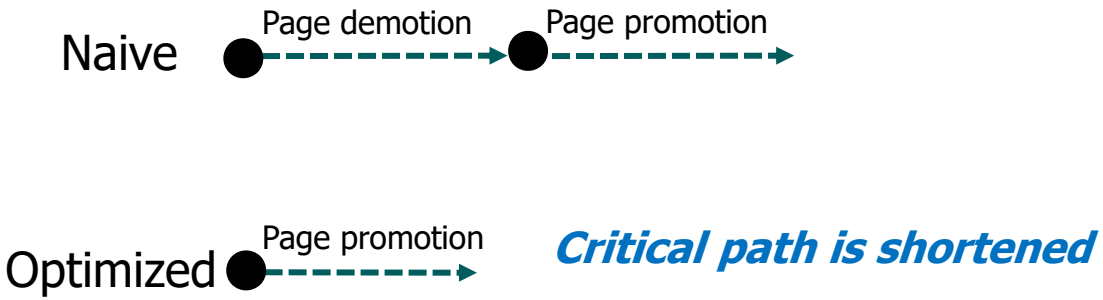
- Finding the Least Accessed Page (`LAP`)
  1. Inactive page from `file-backed` region
  2. `LAP` page from `anonymous` region
  - Only evict `LAP` if its access frequency is lower than the page to be promoted



Access history is collected from AutoNUMA framework

Hiding Latency of Page Demotion

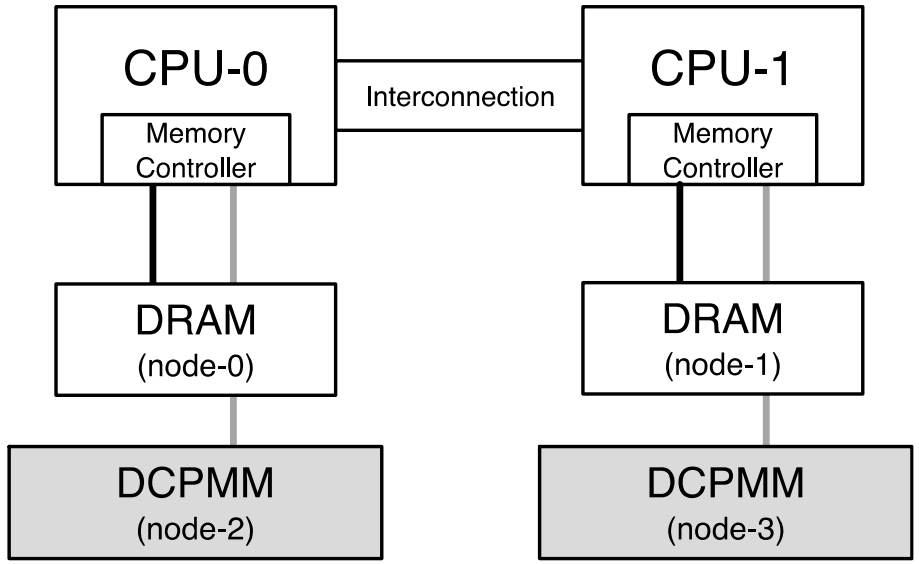




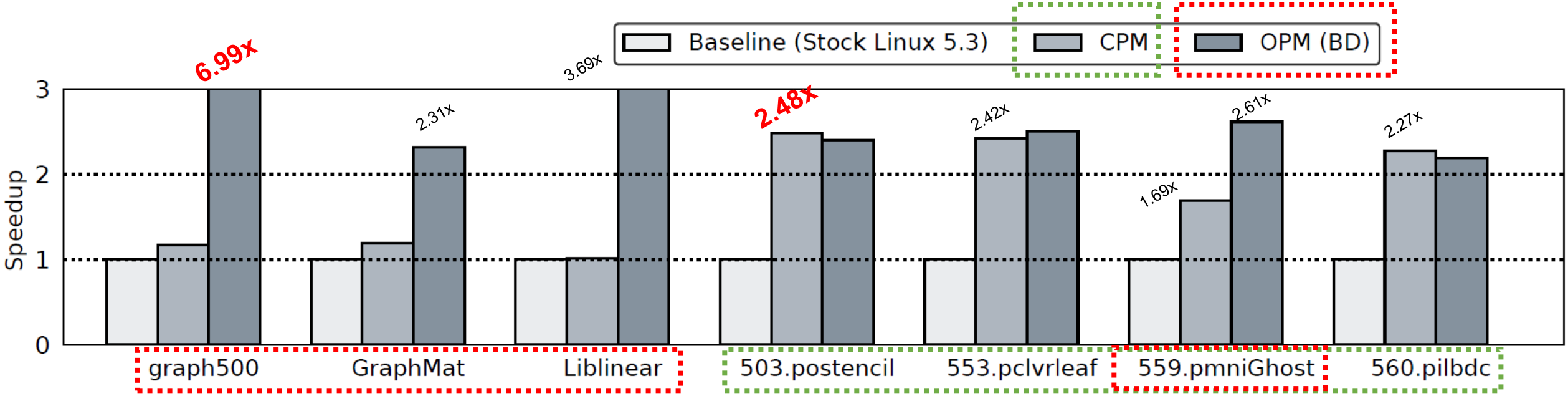
For more details, please refer to our paper

Experimental Environments

- System
  - Intel(R) Xeon Gold 5218 CPU @ 2.30GHz x 2
  - 16GB DRAM x 2
  - 128GB Intel Optane DCPMM x 2
  - Linux kernel 5.3 with Ubuntu 18.04
- Benchmarks
  - SPECAccel (OpenMP)
  - GraphMat (PageRank)
  - Graph500 (BFS)
  - Liblinear

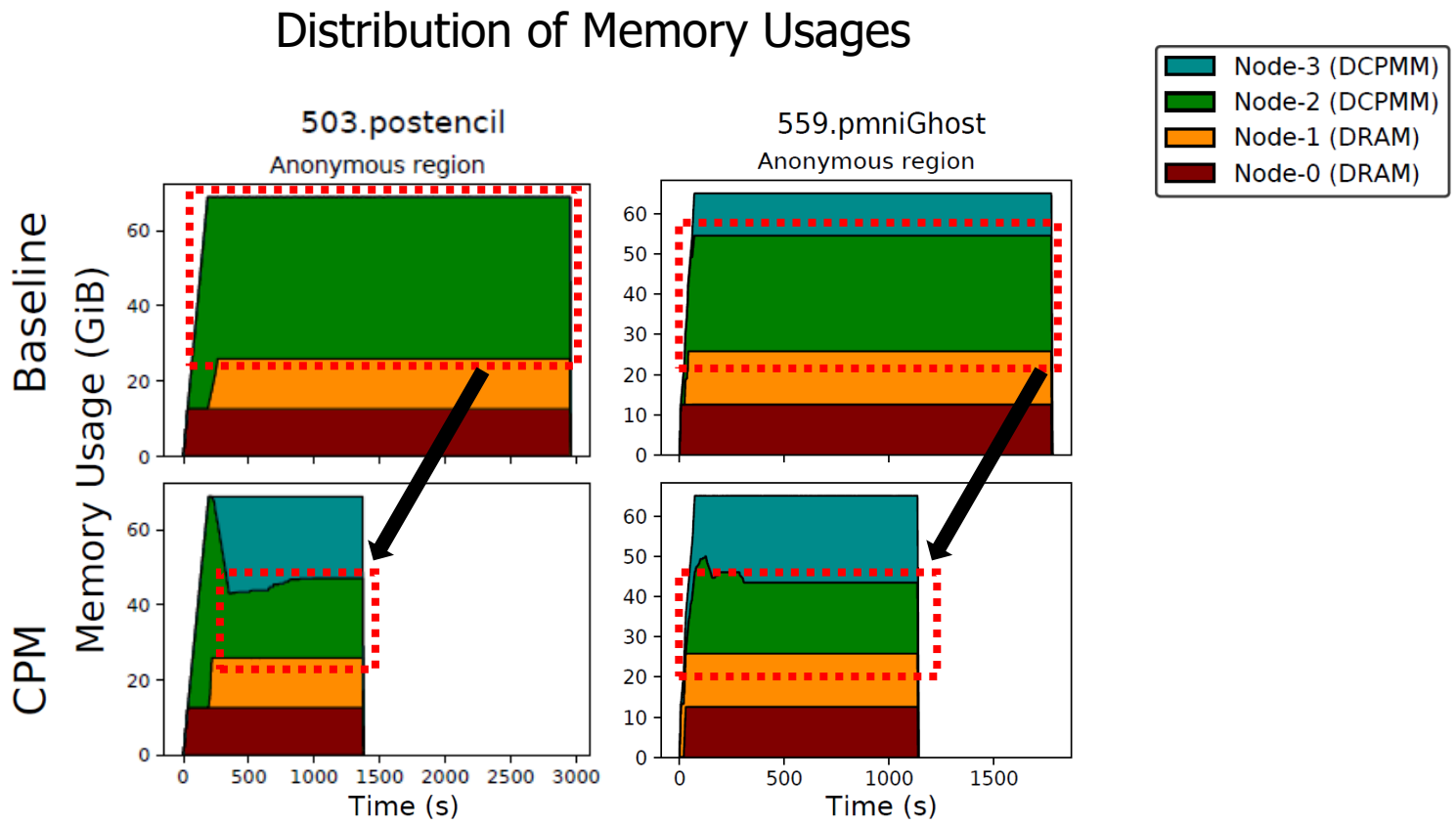


Performance Evaluation



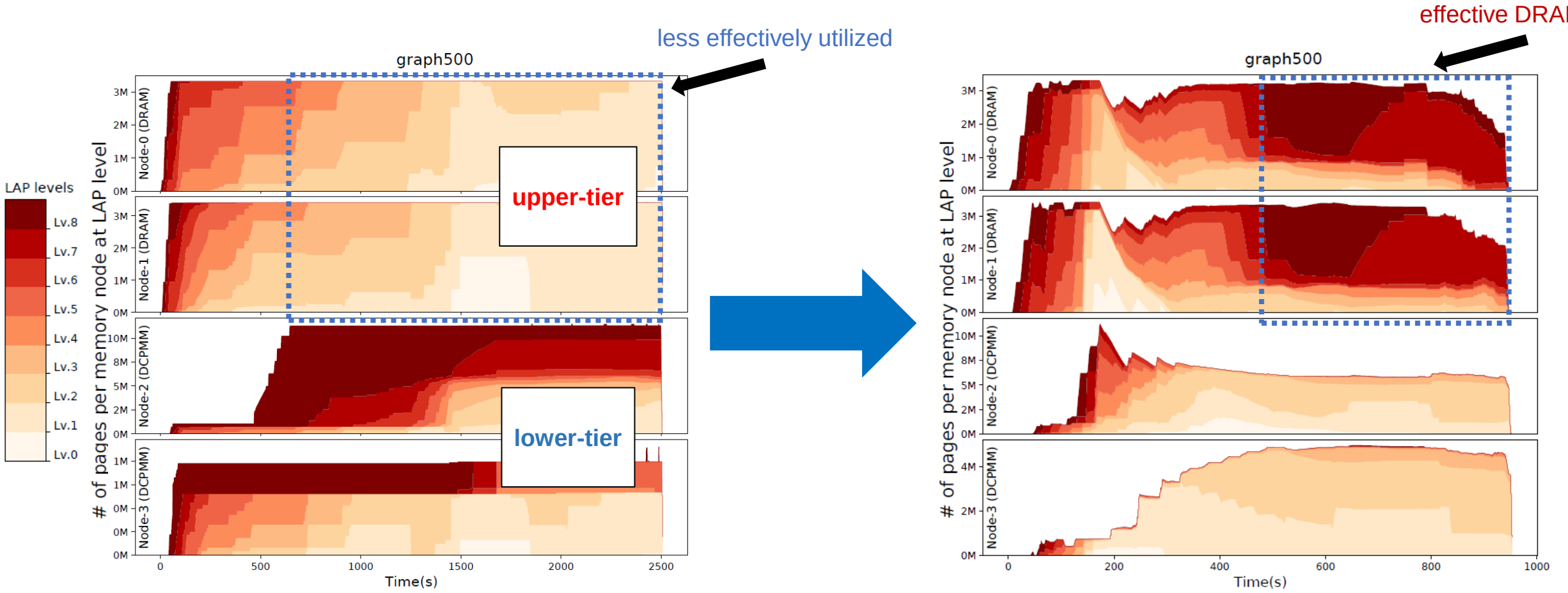
- Most benchmarks are improved by AutoTiering
- In CPM, speedup is up to *2.48x* at 503.postencil
- In OPM(BD), speedup is up to *6.99x* at graph500

Effectiveness of `AutoTiering-CPM`



AutoTiering-CPM makes better use of multi-tiered memory

Effectiveness of LAP classification



AutoTiering-OPM can promote frequently accessed pages while demoting least accessed pages

Conclusion

- Commodity OSes are not mature enough to support multi-tiered memory systems
- We explored new page placement schemes to extract the full benefits of multi-tiered memory systems
- Future work
  - Performance overhead of page migration between DRAM and DCPMM
  - Multi-tenancy issues

Thank You!

tome01@ajou.ac.kr

Exploring the Design Space of Page Management for Multi-Tiered Memory Systems

Source code: http://github.com/csl-ajou/autotiering

Jonghyeon Kim, Wonkyo Choe, Jeongseob Ahn

AJOU UNIVERSITY