





Managing Memory Tiers with CXL in Virtualized Environments



Proceedings of the 18th USENIX Symposium on Operating Systems Design and Implementation. July 10–12, 2024 • Santa Clara, CA, USA

Yuhong Zhong  Daniel S. Berger   Carl Waldspurger* Ryan Wee 

Ishwar Agarwal  Rajat Agarwal  Frank Hady  Karthik Kumar 

Mark D. Hill  Mosharaf Chowdhury  Asaf Cidon 

 Columbia University  Microsoft Azure  University of Washington

*Carl Waldspurger Consulting  Intel  University of Wisconsin-Madison  University of Michigan

2025. 01. 21

Presentation by Yeongyu Choi

choiyg@dankook.ac.kr

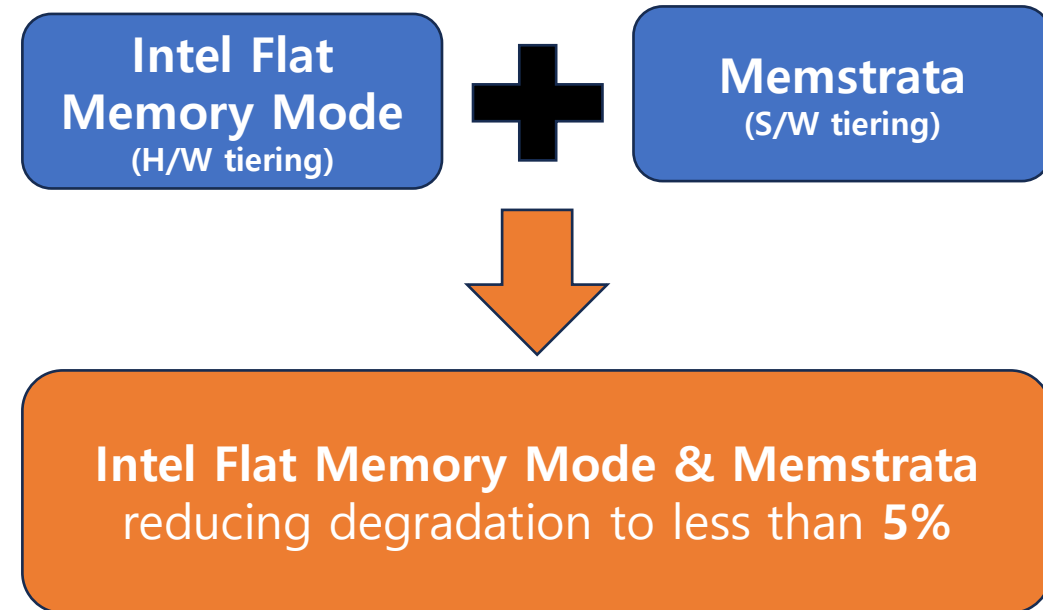
Contents

1. Introduction
2. Background
3. Motivation
4. Intel Flat Memory Mode
5. Memstrata
6. Evaluation
7. Conclusion

Introduction

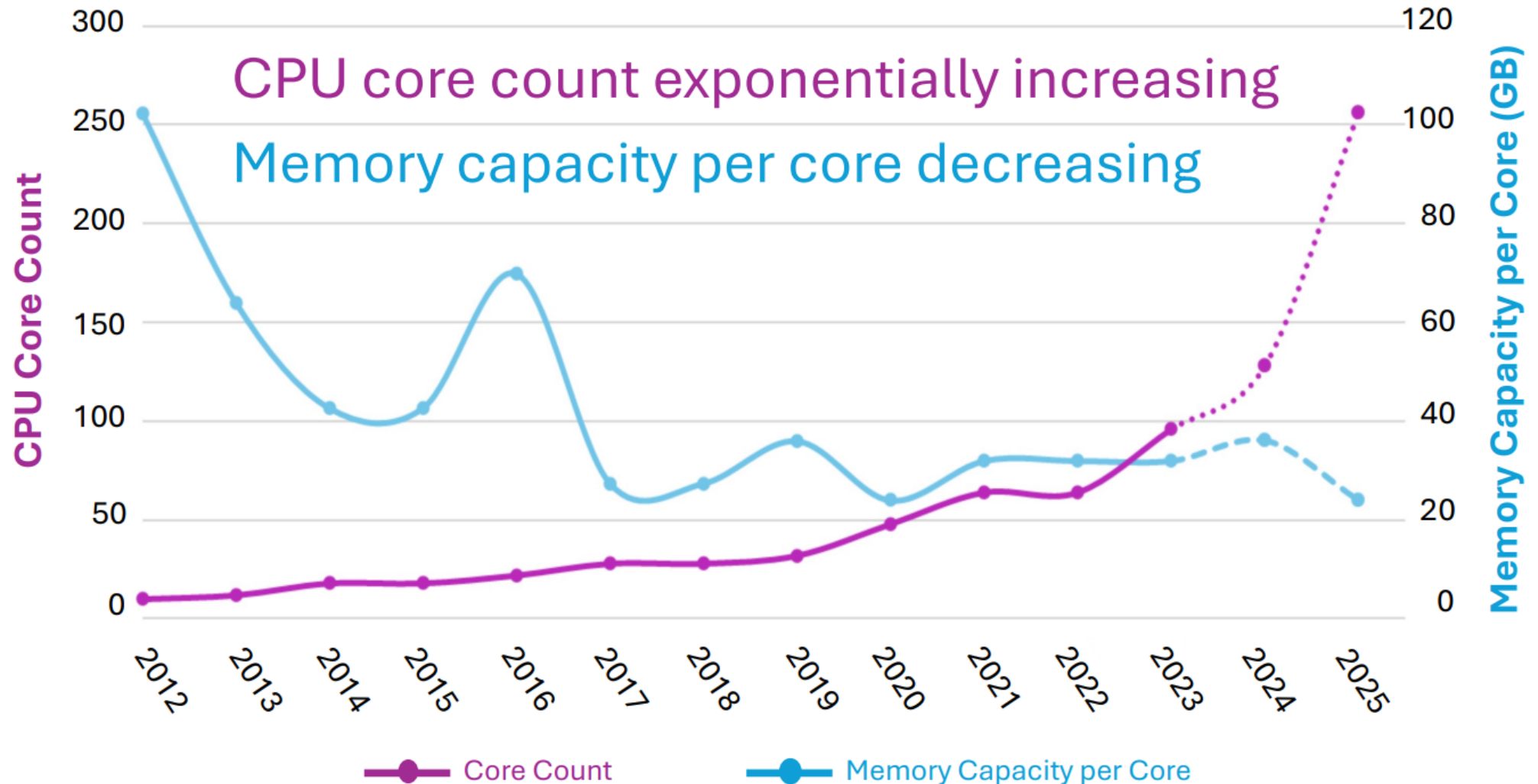
- **CPU core counts scaling faster** than memory capacity
- Insufficient memory capacity in **cloud environments leads to performance bottlenecks**
- **CXL enables second-tier memory** to facilitate core scaling
 - But **CXL adds latency** that hurts performance if not mitigated
- **Through the CXL based memory tiering**
 - Mitigate performance degradation in CXL-based memory tiering
 - Provide performance isolation between Virtual Machines
 - Address the issue of workload outliers in cloud environments

Contribution



Background

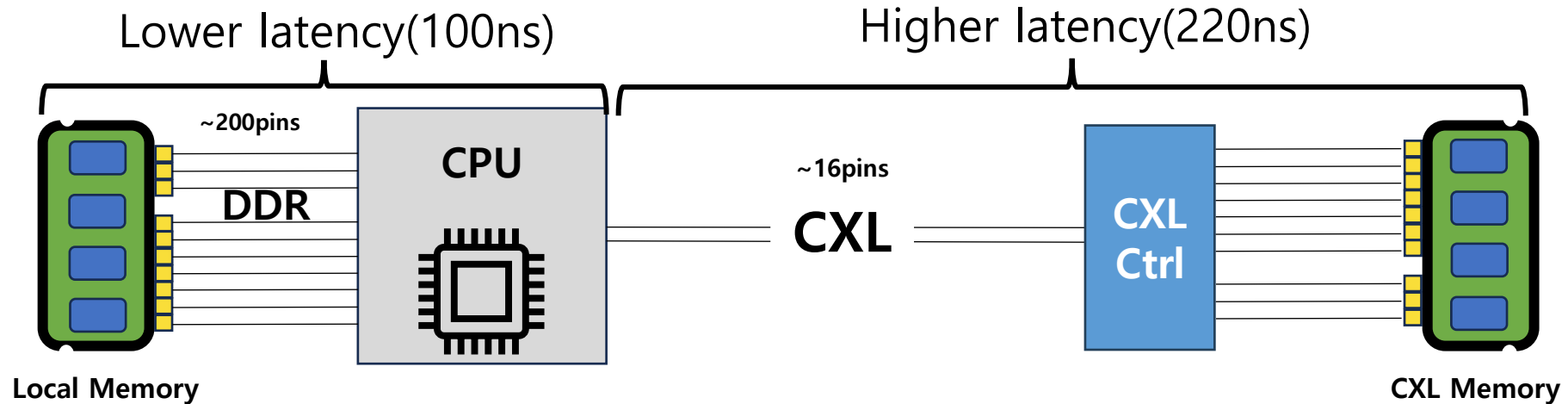
Growth of CPU Cores and Memory



Source: Micron's Perspective on Impact of CXL on DRAM Bit Growth Rate

Background

What's the difference between the CXL and DDR



	DDR	CXL
Protocol Type	Parallel	Serial(PCIe-based)
Pin Count	More pins	Fewer pins
Scalability	Limited	High
Latency	Lower	Higher(2x local DRAM)
Usage Flexibility	Only DRAM modules supported	Supports heterogeneous memory types (e.g., DRAM, NVRAM)

Higher CXL Latency Can Degrade Workloads

- CXL latency (220 ns) \approx 2x local memory latency (100 ns)
- CXL slowdowns workloads by up to 62%
- Memory tiering: place data between local and CXL memory
- Cloud requirements for CXL include:
 - Minimal slowdown
 - Low CPU overhead
 - Huge page friendly

Background & Motivation

Type of Memory Tiering

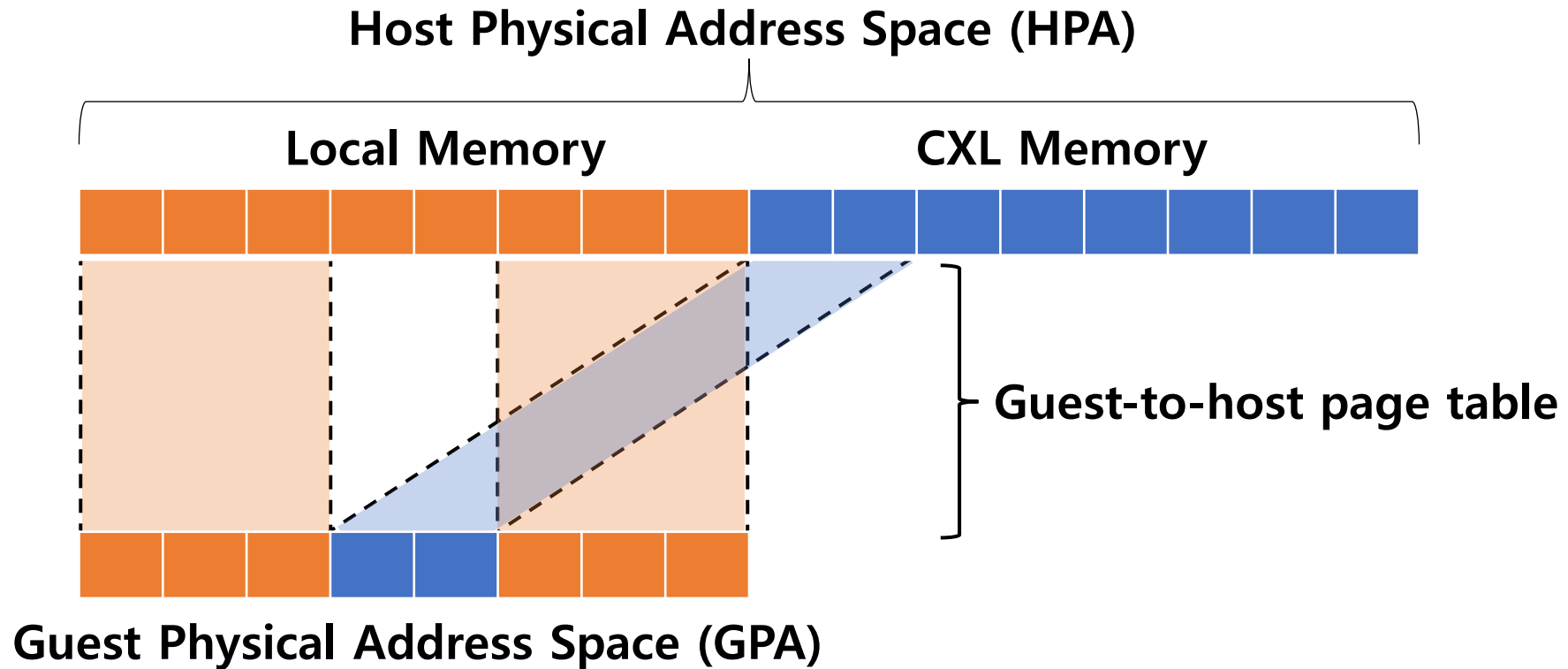
Introduced in this work

	S/W Tiering	H/W Tiering	S/W + H/W Tiering
	HeMem (SOSP'21) TPP (ASPLOS'23) MENTIS (SOSP'23)	Intel Flat Memory Mode	Intel Flat Memory Mode + Memstrata
Minimal slowdown	High tail slowdown	High tail slowdown	Minimal slowdown
CPU overhead	High overhead	Low overhead	Low overhead
Huge page friendly	Unfriendly	Friendly	Friendly

Motivation

Prior Work: Software-Managed Memory Tiering

Use **hypervisor/OS** to **identify popular pages** and **decide page placement**



Motivation

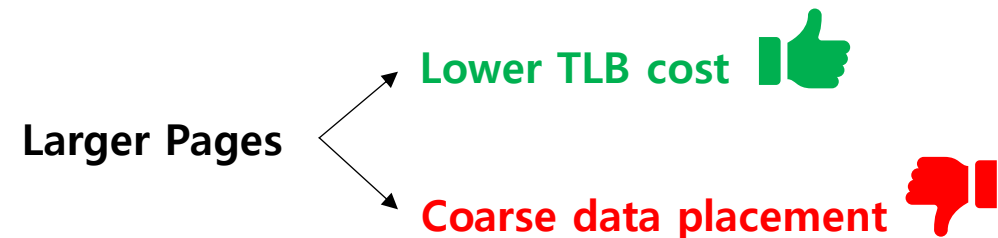
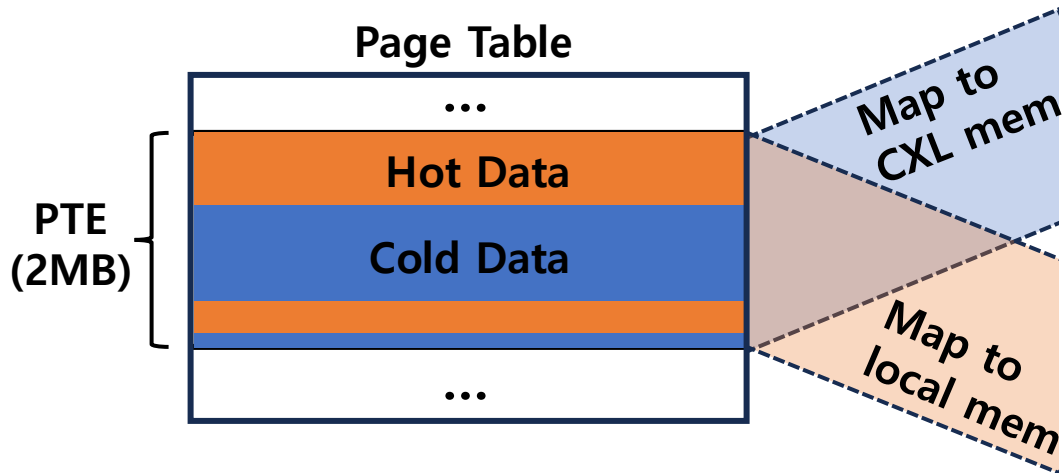
Software Tiering at Odds With Virtualization

Issue 1: High CPU overhead

- Instruction sampling(PEBS,IBS) is disabled in clouds
- Frequent page table scans incur excessive CPU overhead

Issue 2: Huge page penalty

- Virtualization uses larger page sizes(2MB, 1GB) to reduce TLB cost



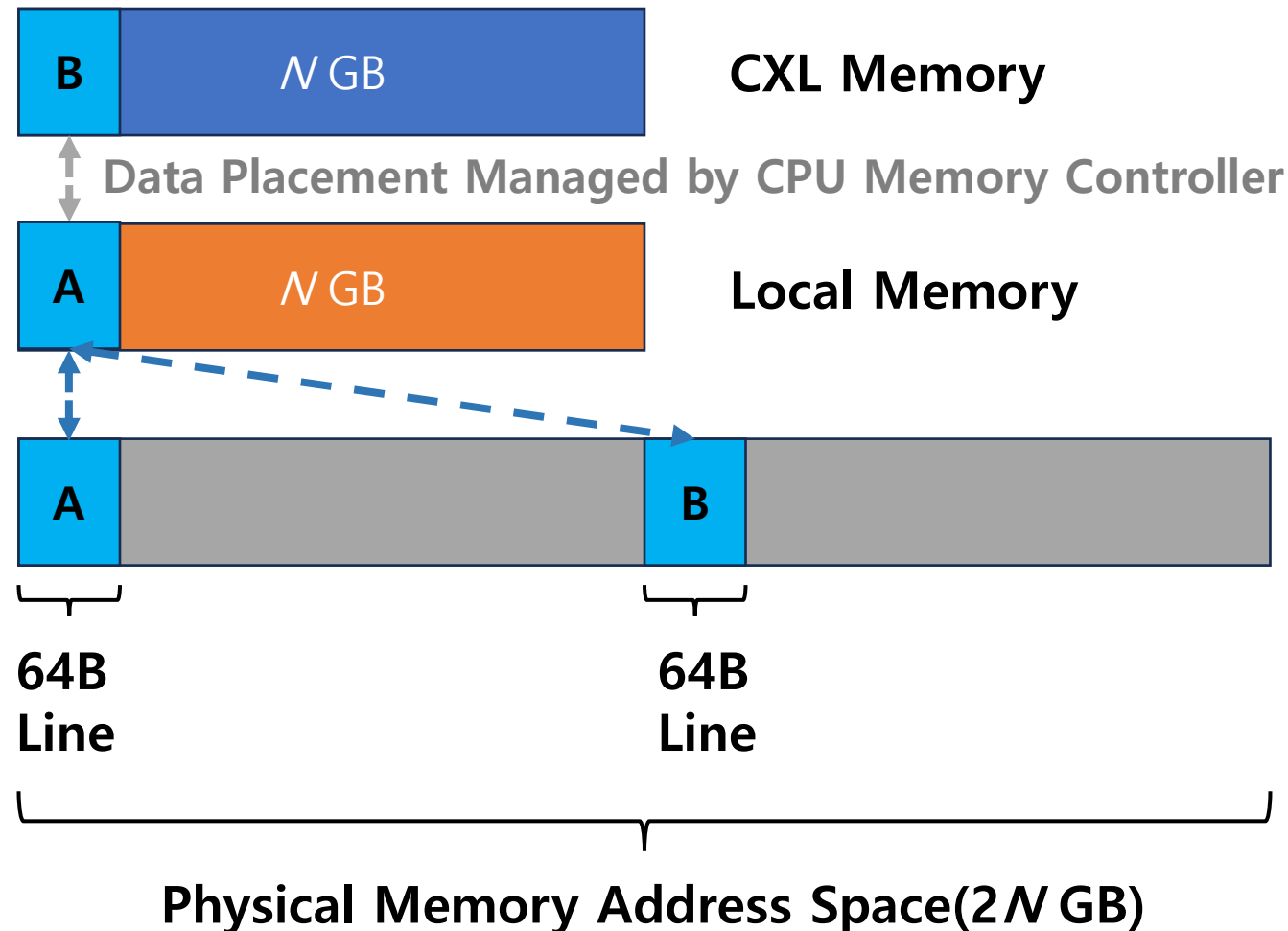
Intel Flat Memory Mode

Introducing Hardware Tiering for CXL

- First hardware-managed cacheline-granular memory tiering for CXL
- Data placement managed by the CPU memory controller
 - Zero CPU over head
 - Huge page friendly

Associativity and Mapping of Intel Flat Memory Mode

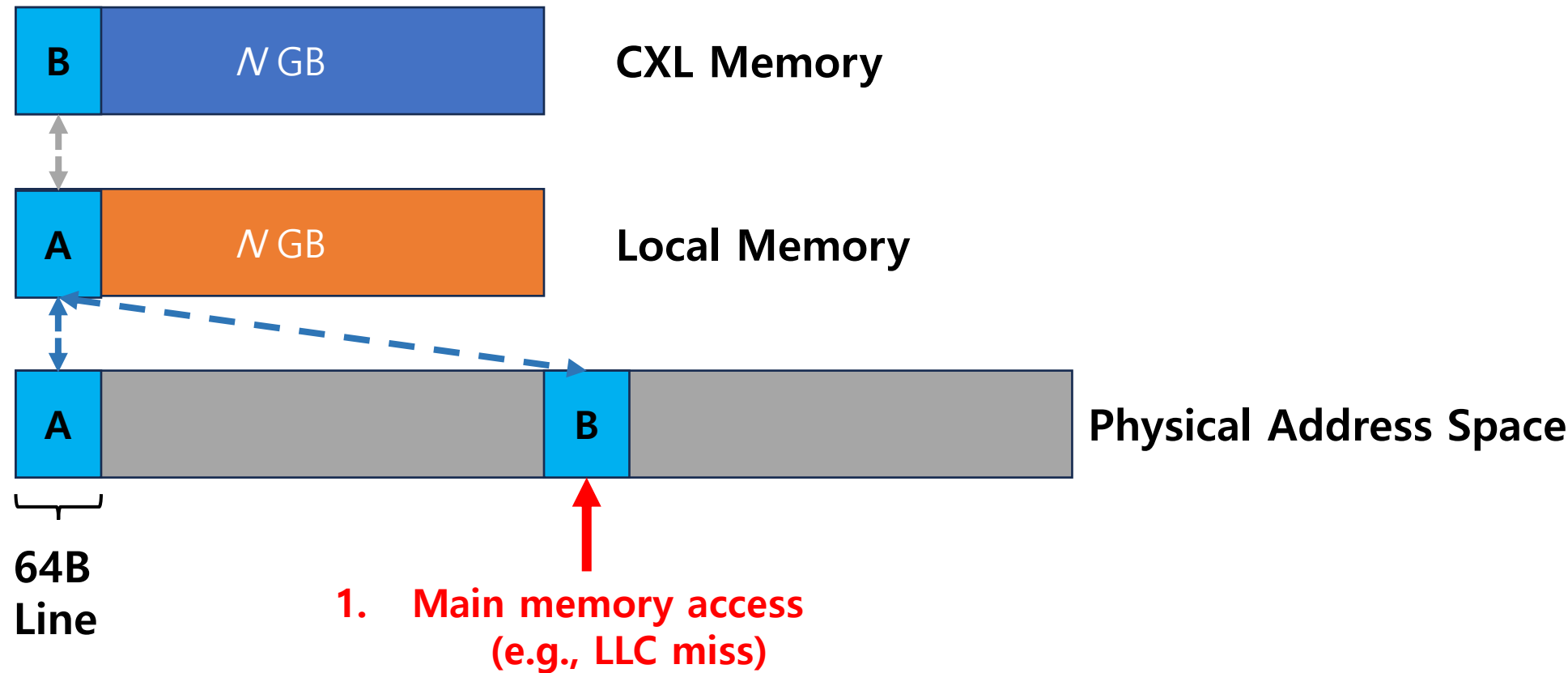
Local memory as a **direct-mapped, exclusive** cache of CXL memory



Intel Flat Memory Mode

Local Memory Miss in Intel Flat Memory Mode

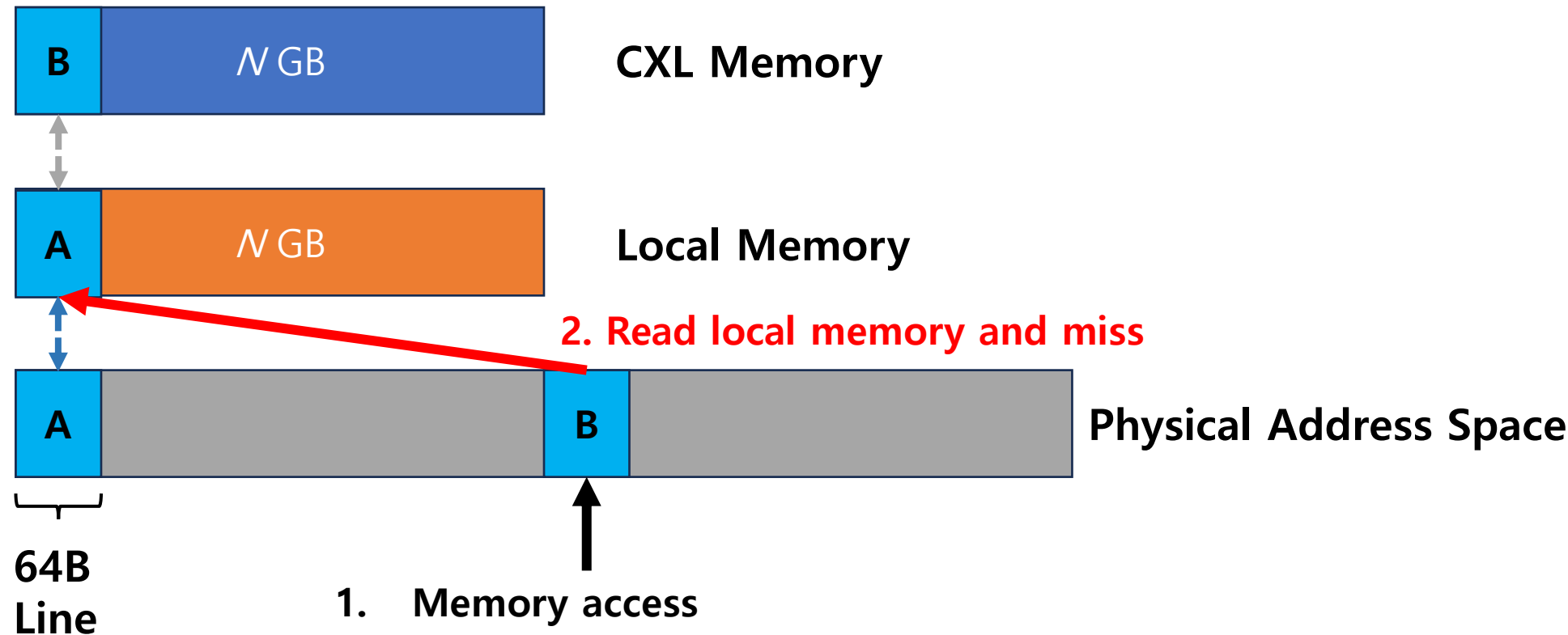
When a main memory access misses in local memory, the hardware will “**swap**” the two cache lines



Intel Flat Memory Mode

Local Memory Miss in Intel Flat Memory Mode

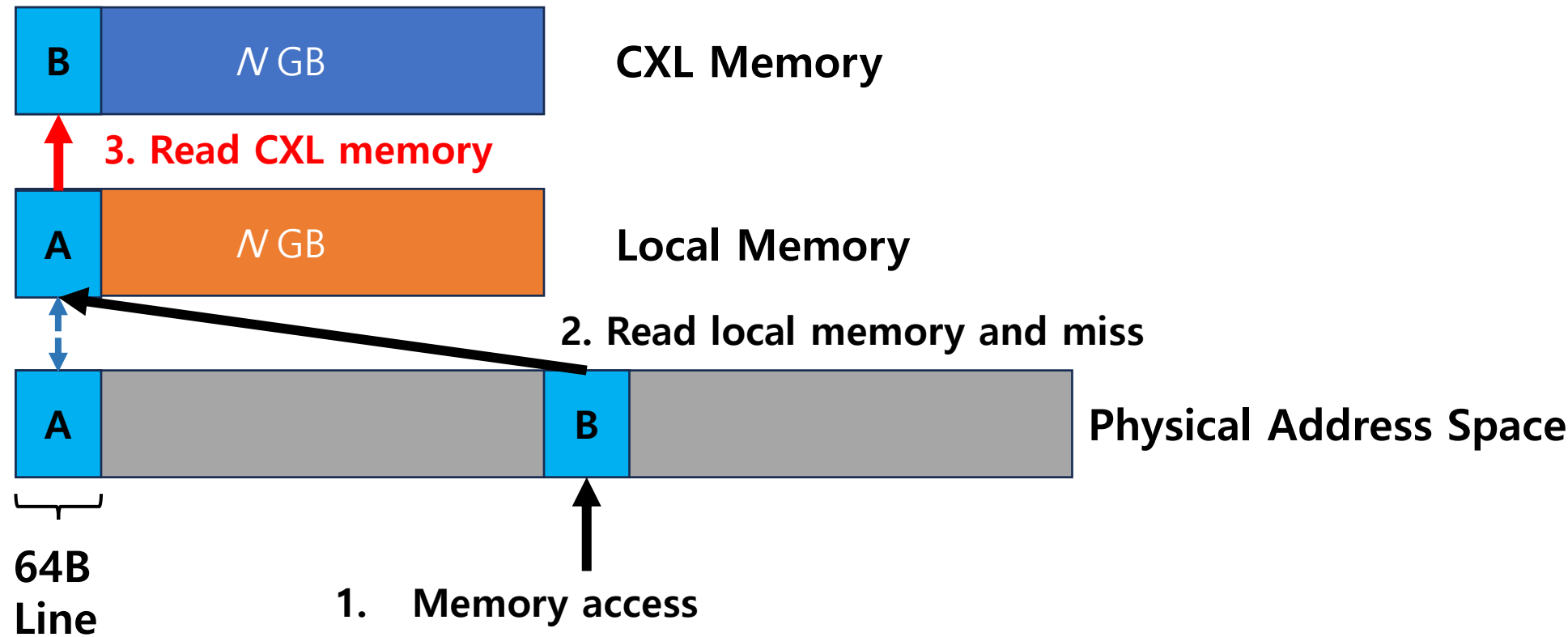
When a main memory access misses in local memory, the hardware will “**swap**” the two cache lines



Intel Flat Memory Mode

Local Memory Miss in Intel Flat Memory Mode

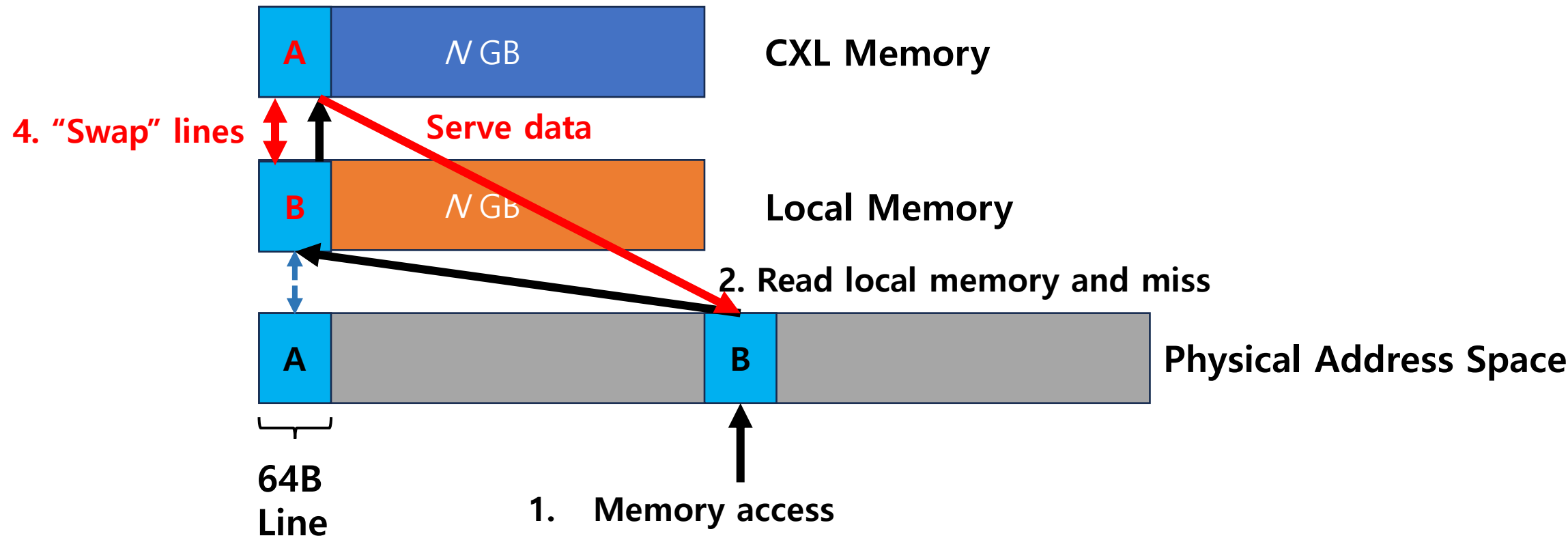
When a main memory access misses in local memory, the hardware will “**swap**” the two cache lines



Intel Flat Memory Mode

Local Memory Miss in Intel Flat Memory Mode

When a main memory access misses in local memory, the hardware will “**swap**” the two cache lines



Limitation of Intel Flat Memory Mode

Hardware Tiering Alone Still has Limitations

- Challenge 1: Some workloads have **heavy local memory misses**

26% workloads have >5% slowdown
("outlier" workloads)

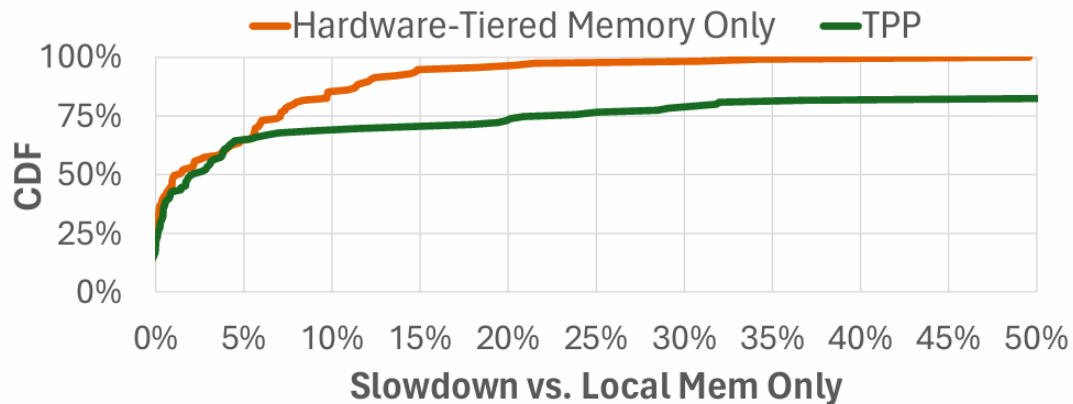
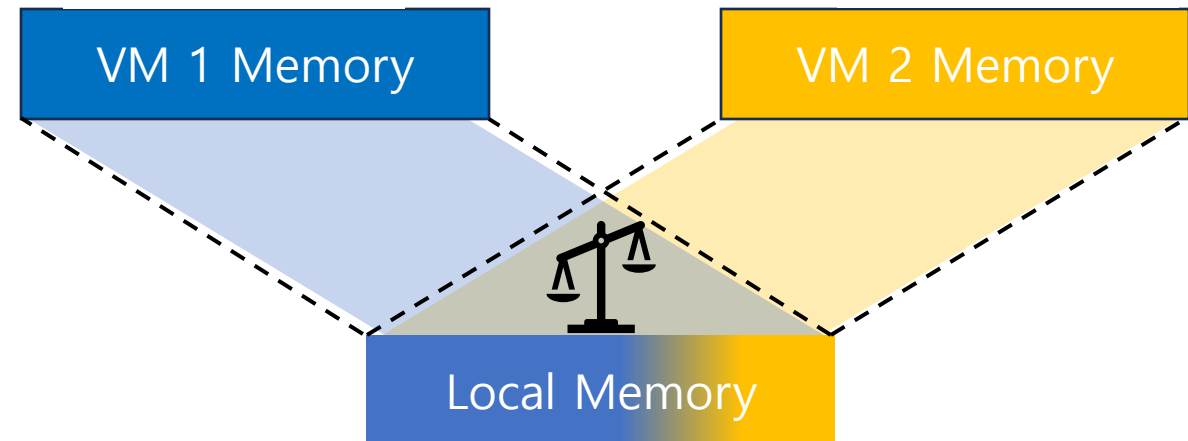


Figure 11: Slowdown distribution of hardware-tiered memory and TPP. TPP is configured with 50% local memory to match the local memory ratio of hardware-tiered memory.

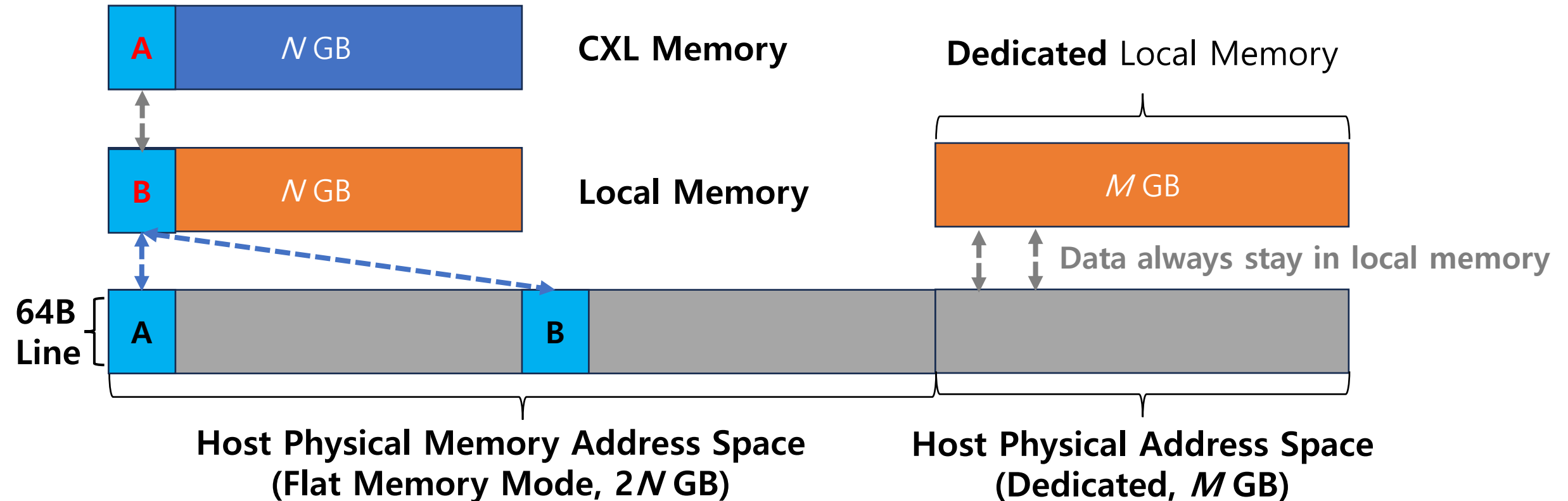
- Challenge 2: **No performance isolation** across VMs

Local memory contention across VMs
(more than 50% slowdown)



Adding Dedicated Local Memory for Outliers

How to allocate dedicated local memory across VMs?



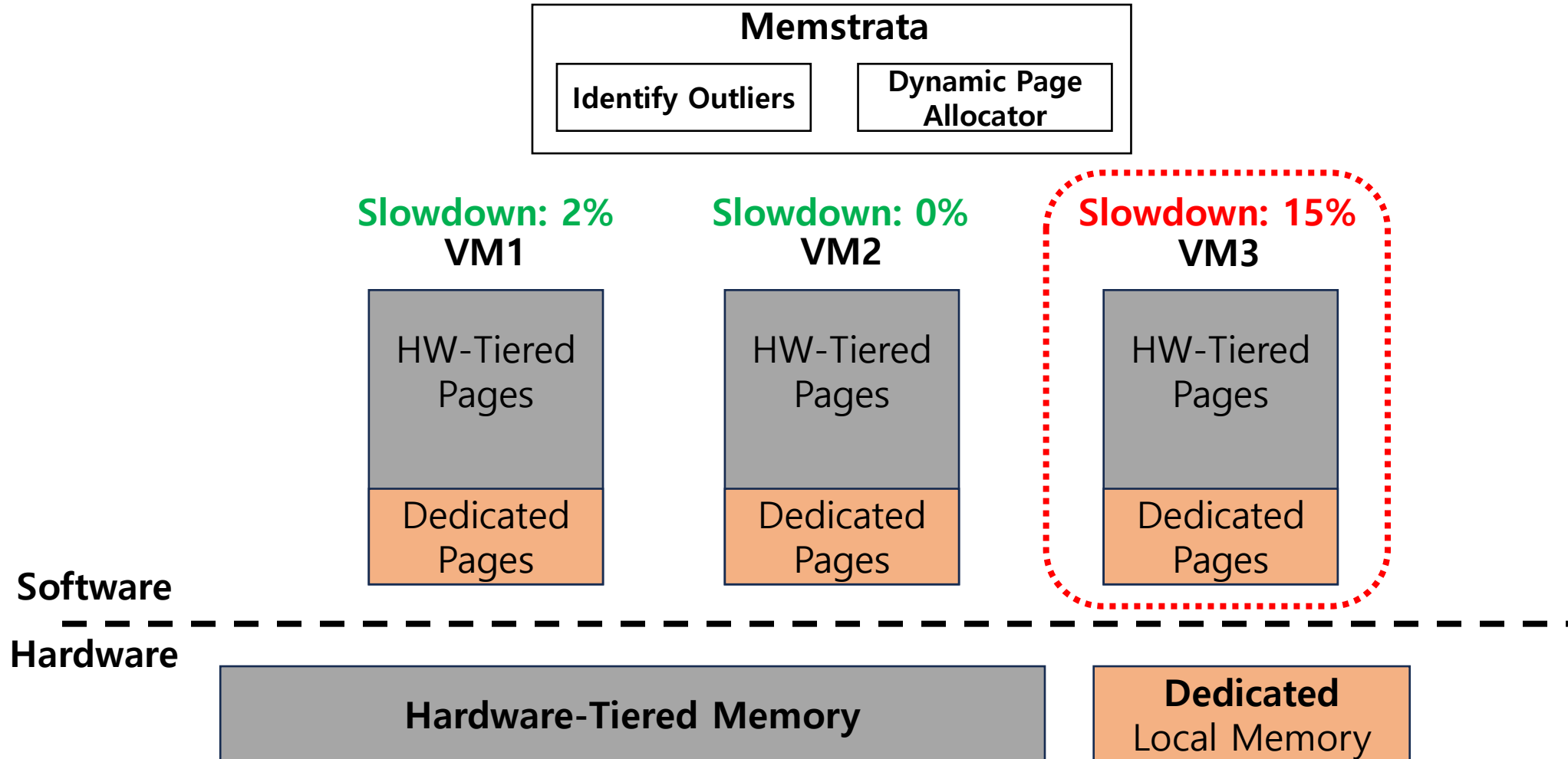
Memstrata: Memory Allocator for HW Tiering

- A lightweight memory allocator in the hypervisor
- Dynamically allocates dedicated memory to **eliminate outliers**
- Provides **performance isolation** between VMs using page coloring

Memstrata + H/W tiering reduces slowdown from 34% to ~5% across all workloads

Memstrata: Memory Allocator for HW Tiering

Memstrata Dynamically Allocates Dedicated Pages



Identifying Outliers in Hypervisor Is Challenging

Challenges:

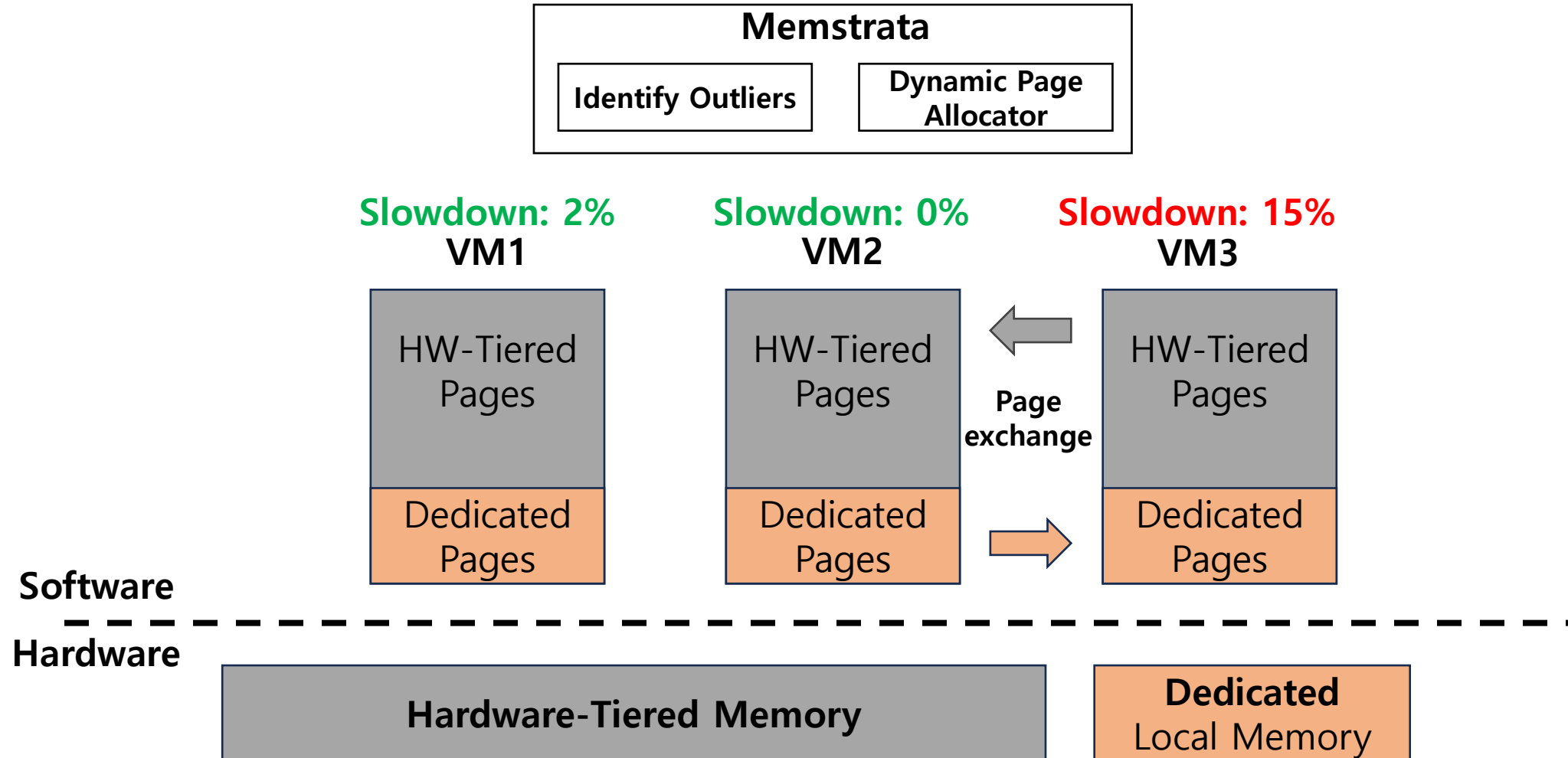
- Hypervisor is unaware of VM workloads
- Hardware tiering only provides system-wide local memory miss rate

We build a **lightweight prediction model** to identify outliers using low-level performance metrics

- **Per-core** metrics: L3 miss latency correlates with miss ratio

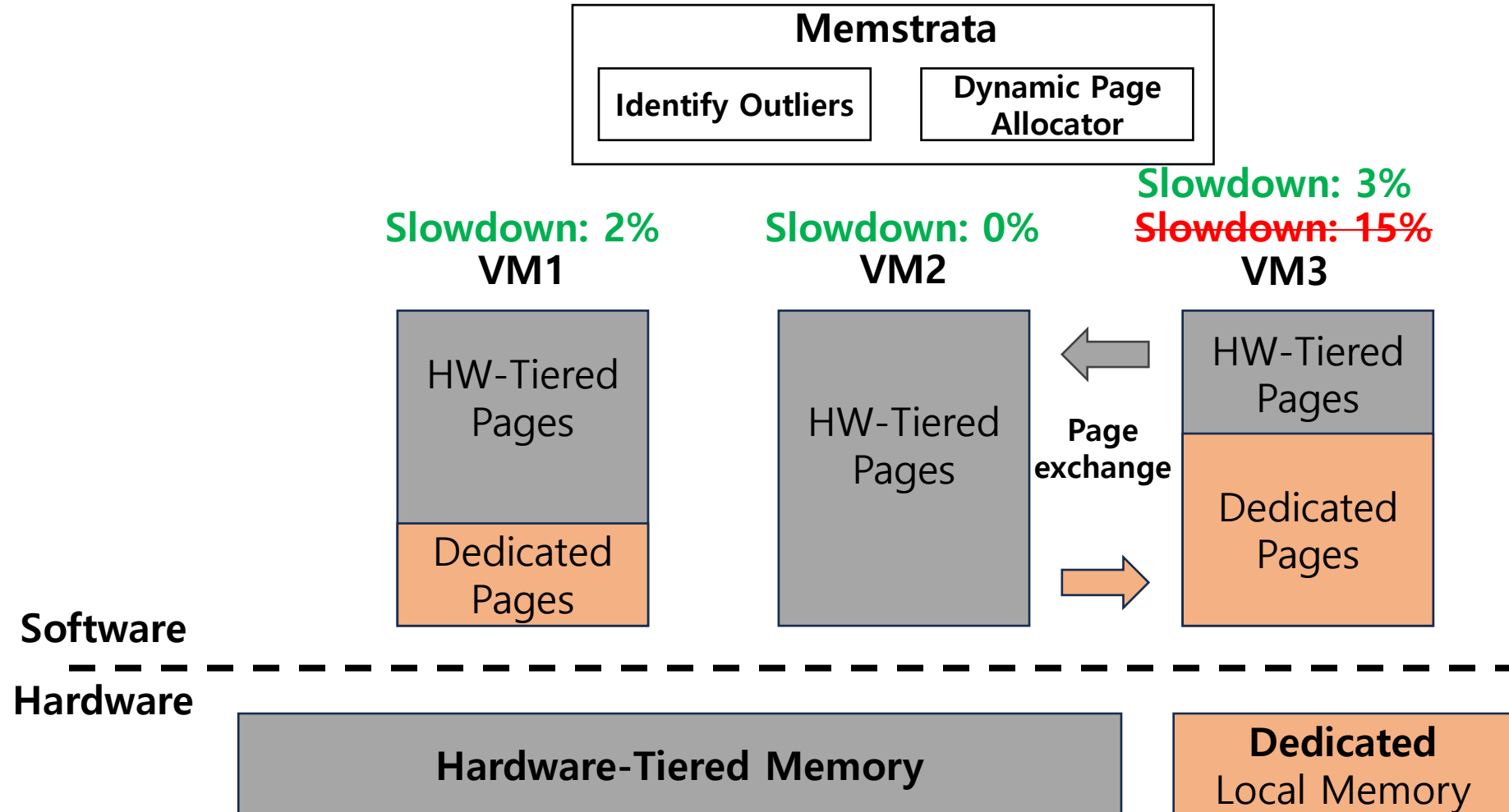
Memstrata: Memory Allocator for HW Tiering

Memstrata Dynamically Allocates Dedicated Pages



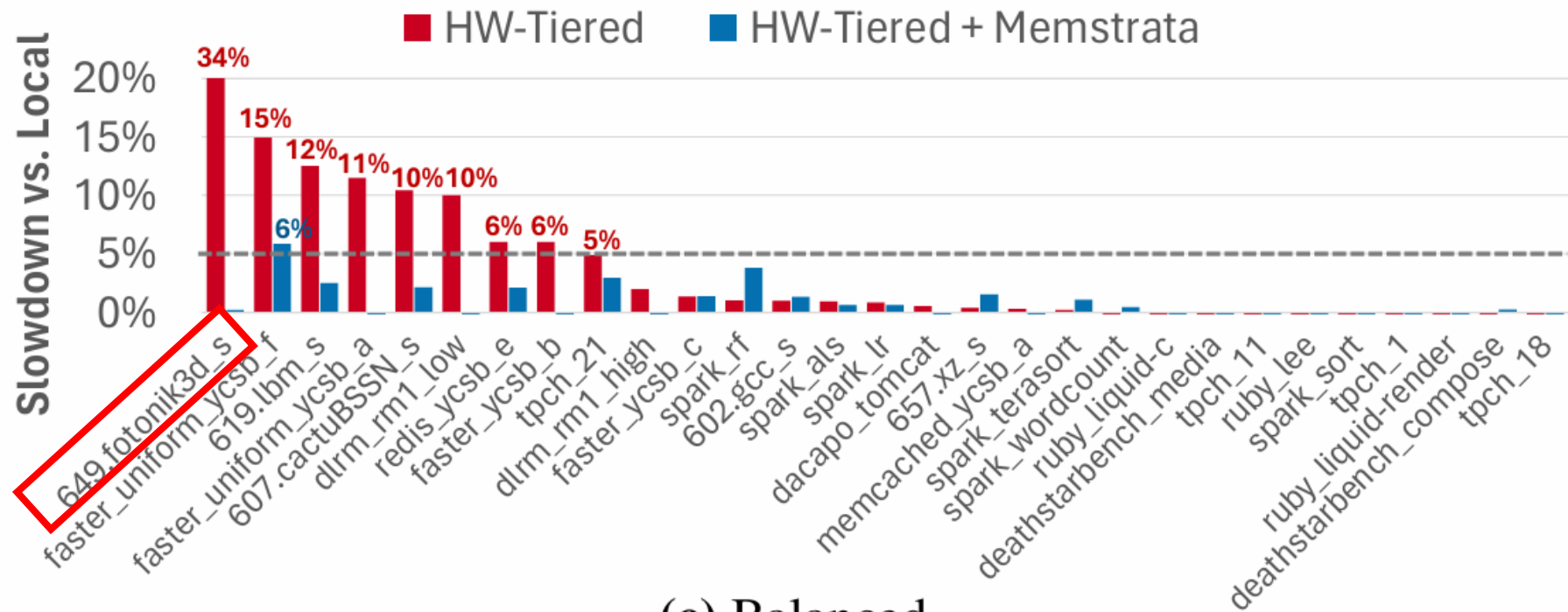
Memstrata: Memory Allocator for HW Tiering

Memstrata Dynamically Allocates Dedicated Pages



Evaluation

Processor: Intel® Xeon® 6 Processor
Local Memory: 128GB DDR5 DRAM
CXL Memory: 128GB DDR5 DRAM via 3 CXL cards
(each hold two DDR5-4800 DIMMs and offer an x16 PCIe5 CXL connection)
CXL Controller: Astera Labs Leo CXL Smart Memory Controller
OS: Ubuntu 22.04 / **Kernel:** Linux 5.19(Modified) / **QEMU/KVM 6.2**



(c) Balanced

Conclusion

- This paper presents Intel® Flat Memory Mode and Memstrata as a combined hardware-software solution to address the challenges of memory tiering in virtualized environments using CXL
- The evaluation showed that the proposed approach achieved less than **5% slowdown for 82% of workloads**, and Memstrata **reduced outlier slowdowns from 30% to 6%**
- This provides a scalable and cost-effective solution to meet the growing memory demands of modern data centers

Thank you