

# The Case for DRAM Translation Layer (DTL): Enhancing Datacenter Efficiency, Reliability, and Beyond



Jae W. Lee ([jaewlee@snu.ac.kr](mailto:jaewlee@snu.ac.kr))  
Seoul National University  
10/25/2024

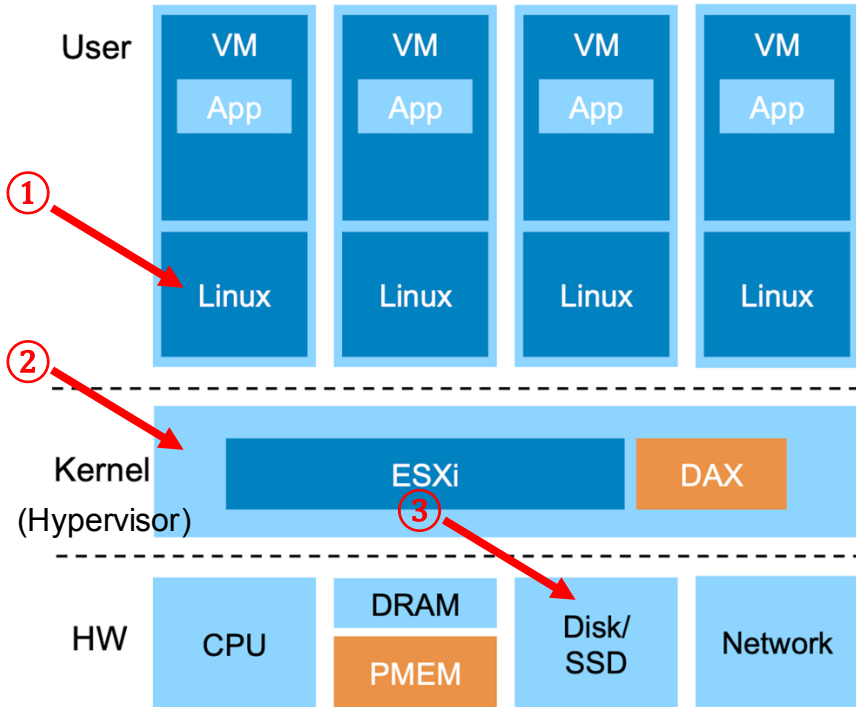
# A Quote

*“Any problem in computer science can be solved with another level of indirection.”*

- Butler Lampson (Turing Award Lecture, 1992)  
attributed to David J. Wheeler



# Levels of Indirection in Virtualized System Stack



Indirection layers are prevalent in today's virtualized system stack.

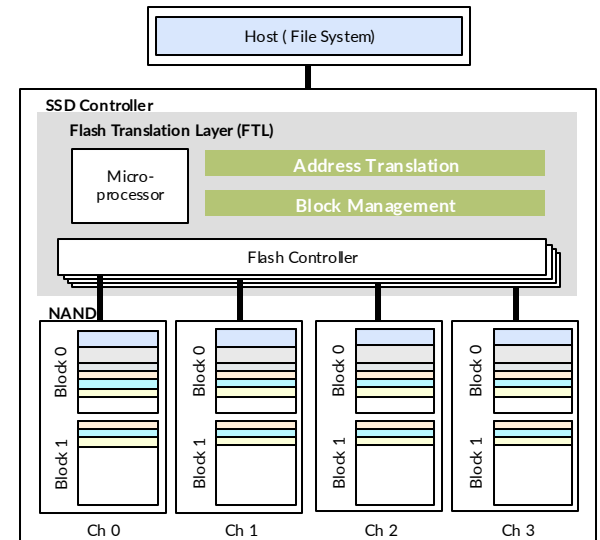
1. OS: Virtual memory  
Virtual address  $\rightarrow$  Physical address (PA)
2. Hypervisor: Memory virtualization  
Guest PA  $\rightarrow$  Host PA
3. SSD: Flash Translation Layer (FTL)  
Logical block address (LBA)  $\rightarrow$  NAND device physical address

Figure from "[Software-Defined Memory: Platform for Virtual Machines](#)" @ memverge.com



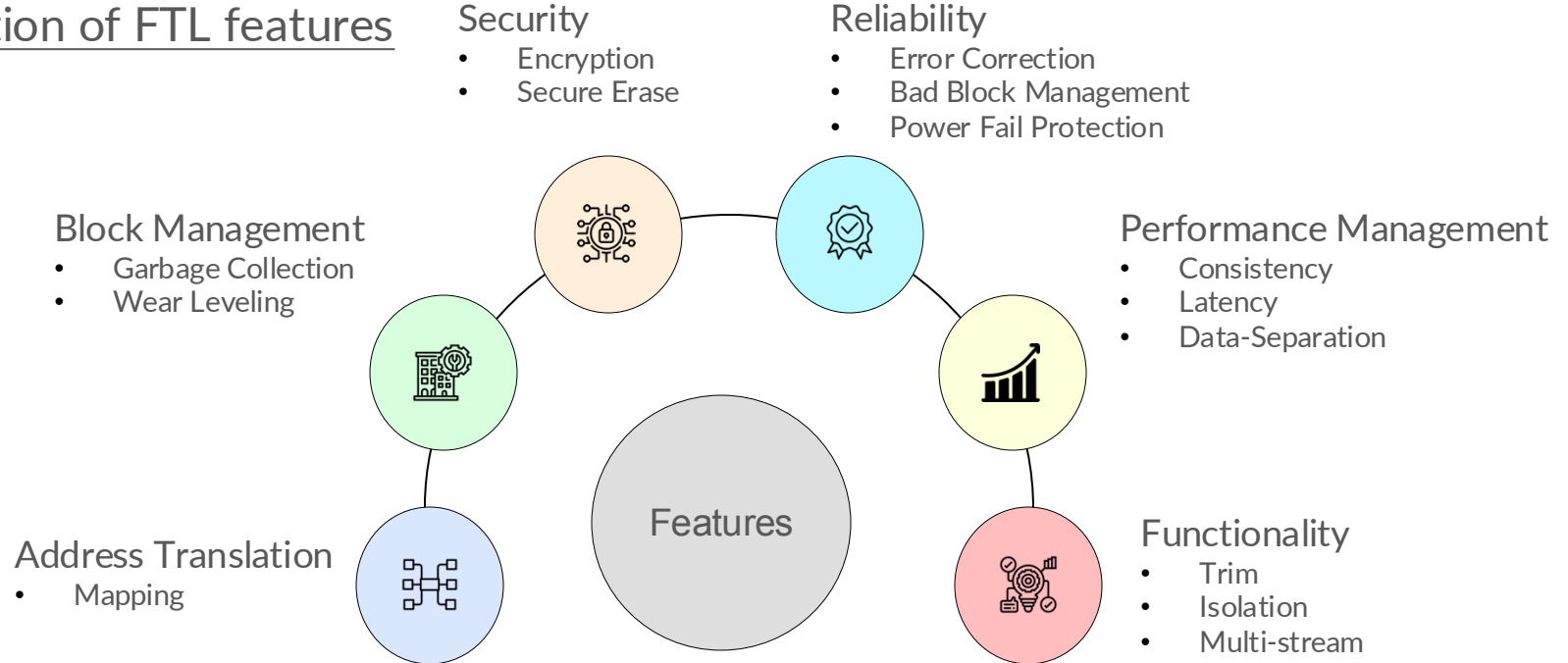
# Example: Flash Translation Layer (FTL)

- A software layer in SSD to make NAND Flash emulate traditional block devices
  - Address Translation
    - Map logical address to physical address
  - Block Management
    - Garbage collection
    - Wear leveling
- NAND Characteristics
  - Erase-Before-Overwrite: No in-place update!
  - Asymmetric units
    - Write & Read: Page (e.g., 8-16 KiB)
    - Erase: Block (e.g., 1-16 MiB)



# Example: Flash Translation Layer (FTL)

## Evolution of FTL features



# CXL and CXL DRAM

- Compute Express Link (CXL) is an open industry interconnect standard enhancing CPU-to-device and CPU-to-memory communication
- CXL utilizes the PCI Express® physical layer for connectivity
- Three protocols on top of that
  - CXL.io: provides discovery, configuration, register access, interrupts, etc.
  - CXL.cache: provides the CXL device access to the processor memory
  - **CXL.mem**: provides the processor access to the CXL device attached memory



# CXL and CXL DRAM

How about **DRAM Translation Layer (DTL)**?

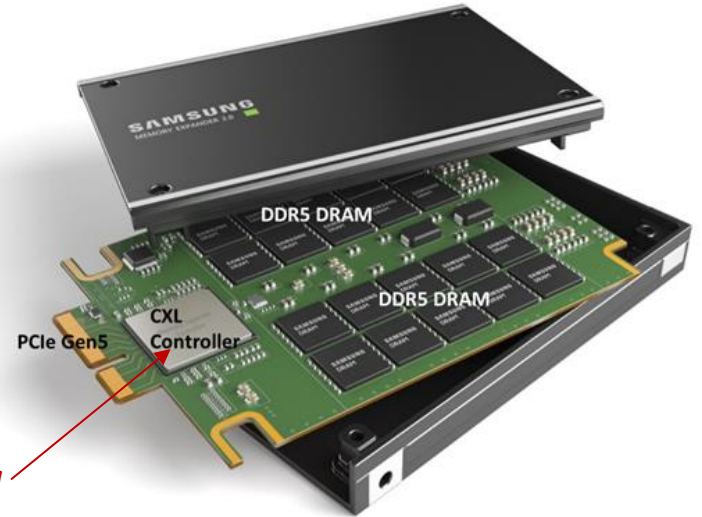
DTL provides capabilities for:

- Host Physical Address (HPA) to Device Physical Address (DPA) translation
- Flexible, host-transparent data migration

Applications?

DRAM power savings, reliability & beyond!

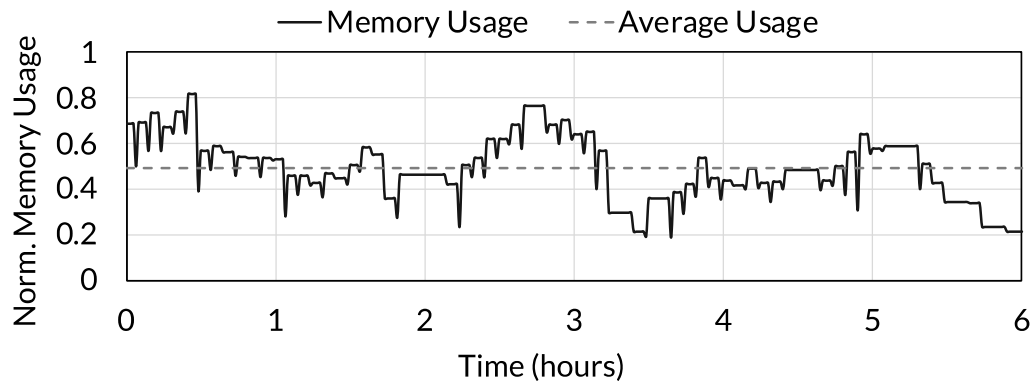
*Place another level of indirection here!*



\* Figure from [Samsung Tech Blog: Expanding the Limits of Memory Bandwidth and Density: Samsung's CXL Memory Expander](#)

# Low Memory Utilization in Datacenters

- **Low utilization of DRAM capacity** in datacenters
  - Only around **40-60%** on average<sup>[1]</sup>



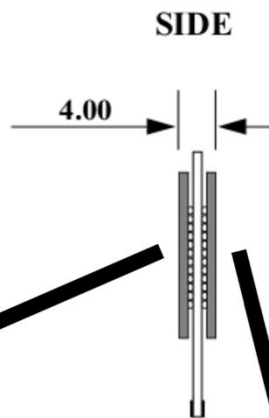
[1] Tirmazi et al., “Borg: the Next Generation”, EuroSys, 2020.



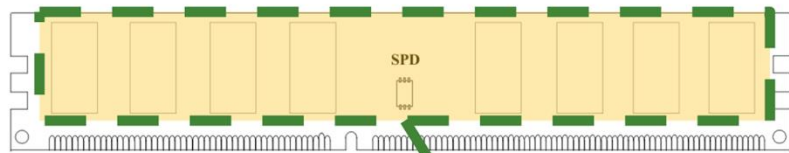
## DIMM (Dual in-line memory module)



Side view



Front of DIMM



Back of DIMM



**CXL** opens up opportunities for localizing power-state management *within* the CXL device, thus makes the DRAM power management *transparent* to the host

# Challenges in Datacenter #2: Memory Reliability

- Memory *density* increase poses challenges to efficiency of **ECC**
- Conventional *memory mirroring*
  - Full mirroring
    - Replicating entire dataset
    - 50% reduction in available memory capacity and significant performance degradation
  - Partial mirroring
    - Predefining the mirrored address range
    - Trading the coverage of protection for higher performance
- **Limitations** in pooled memory system
  - Requiring system reboot or OS intervention
  - Lacking flexibility in managing mirroring space capacity



# Challenges in Datacenter #2: Memory Reliability

- Memory *density* increase poses challenges to efficiency of ECC
- Conventional *memory mirroring*
  - Full mirroring

For *underutilized* memory in datacenters, *CXL* opens up opportunities for localizing power-state management and implementing memory mirroring *within* the CXL devices, thus makes the DRAM power management and reliability enhancement *transparent* to the host

- *Limitations* in pooled memory system
  - Requiring system reboot or *OS* intervention
  - Lacking flexibility in managing mirroring space capacity



# Outline

- Background and Motivation
- **Proposal #1: DRAM Power Savings**
- Proposal #2: Adaptive Partial Mirroring
- Conclusion



# DRAM Translation Layer: Software-Transparent DRAM Power Savings for Disaggregated Memory

Wenjing Jin, Wonsuk Jang, Haneul Park, Jongsung Lee, Soosung Kim, Jae W. Lee

Presented at IEEE/ACM *International Symposium on Computer Architecture (ISCA) 2023*



# Memory Disaggregation Technologies

- RDMA-based memory disaggregation

- Redundant memory copies, OS kernel and network overheads → long access latency ☹️

- CXL-based memory disaggregation 😊

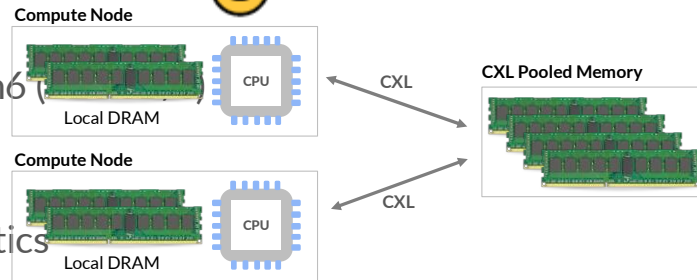
- High-bandwidth

- CXL 3.0 use PCIe Gen6

- Low-latency

-  Compute Express Link™

- Memory semantics



# Our Proposal: DRAM Translation Layer

- DTL is the *first proposal* of address (re)mapping and data migration mechanism for disaggregated memory
  - Transparent to both software stack and memory controller on the host
  - Inspired by Flash Translation Layer (*FTL*) in SSDs, DTL is placed in the CXL memory controller
- Two DRAM power management techniques
  - Rank-level power-down
  - Hotness-aware self-refresh
- Key idea
  - Control address remapping to maximize the number of ranks that can *enter low-power states* while provisioning *sufficient DRAM bandwidth*
- Key results
  - **31.6%** DRAM energy savings for rank-level power-down scheme with **1.6%** of performance cost



# Outline

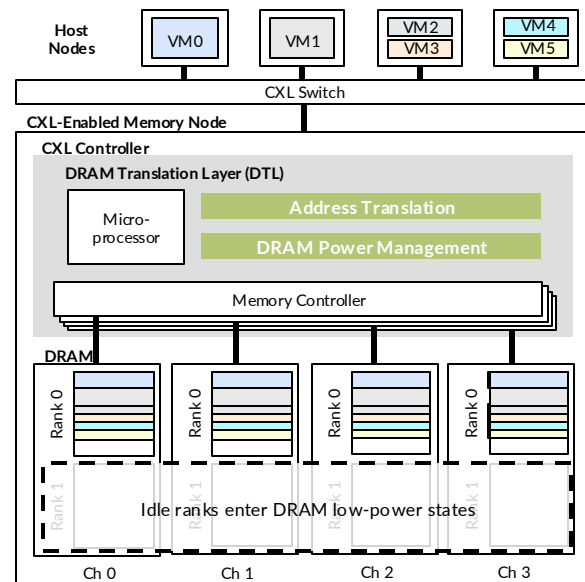
- Overview
- **DRAM Translation Layer**
  - Address translation
  - Two-power management techniques
    - Technique #1: rank-level power-down
    - Technique #2: hotness-aware self-refresh
- Evaluation
- Conclusion





# DRAM Translation Layer Overview

- Address translation
  - Host physical address (**HPA**) → device physical address (**DPA**)
  - Flexible data migration
- Two device-side power-management techniques
  - Rank-level power-down
  - Hotness-aware self-refresh

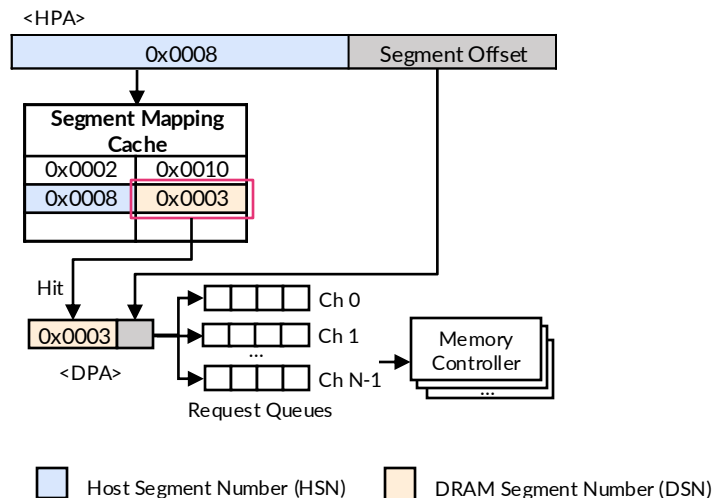


# Address Translation

- Segment is the unit of address mapping in DTL (e.g., 2MB)
- Segment mapping cache (**SMC**)
  - Two-level TLB-like structure for fast HPA-to-DPA translation
  - Hit ratio is **97.7%**

HPA: host physical address    DPA: device physical address

Cache Hit



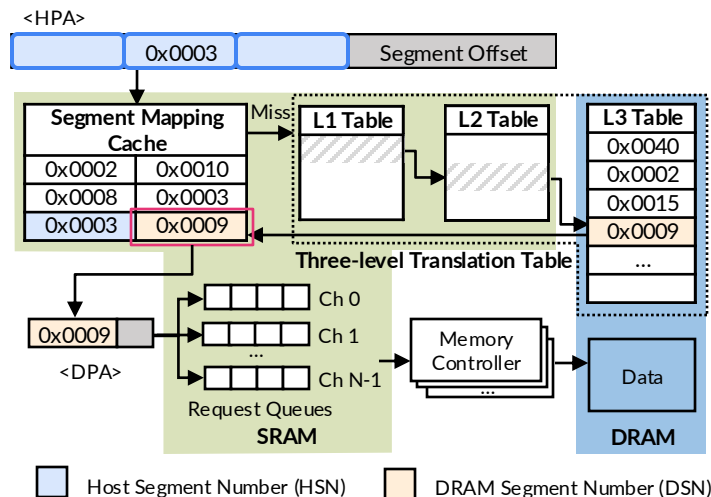
# Address Translation

- Segment is the unit of address mapping in DTL (e.g., 2MB)
- Segment mapping cache (**SMC**)
  - Two-level TLB-like structure for fast HPA-to-DPA translation
  - Hit ratio is **97.7%**

HPA: host physical address    DPA: device physical address

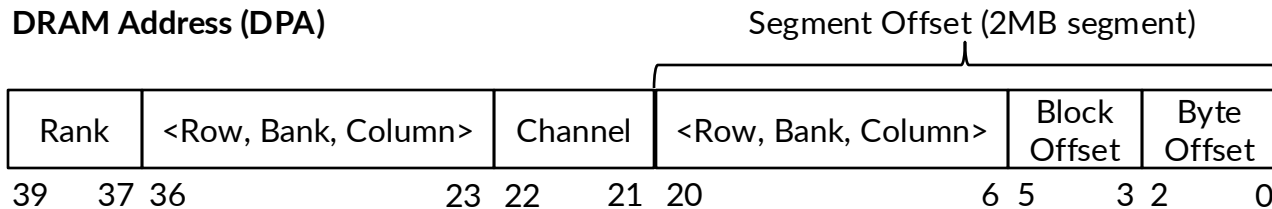
- Translation overhead
  - 4.2ns on average
  - (min/max 0.67ns/123.7ns)
  - 0.18% slowdown on Cloudsuite benchmarks

Cache Miss



# DRAM Physical Address Bit Mapping

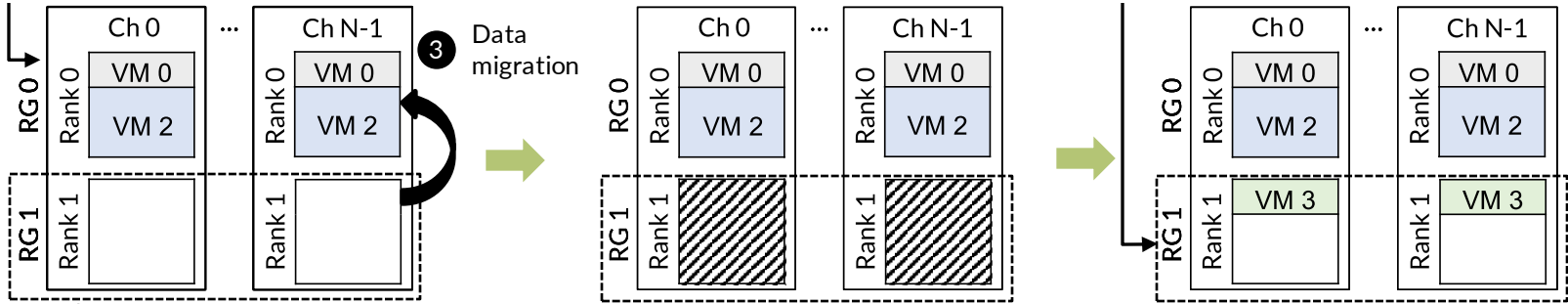
- Performance benefits of rank-interleaving are relatively small for cloud services
  - Only **1.7%** of performance loss for local memory when we do not exploit rank-interleaving
  - The cost further reduced to **1.4%** for remote CXL memory
- DRAM physical address bit mapping for CXL memory
  - Channels are interleaved at segment granularity



# Technique #1: Rank-level Power-down

- **Consolidating unallocated DRAM segments** to subset of ranks at VM deallocation

- Maximizing opportunities for entering **MPSM** mode at rank group granularity



RG: rank group

2 Select rank group to deactivate

4 Power down rank group

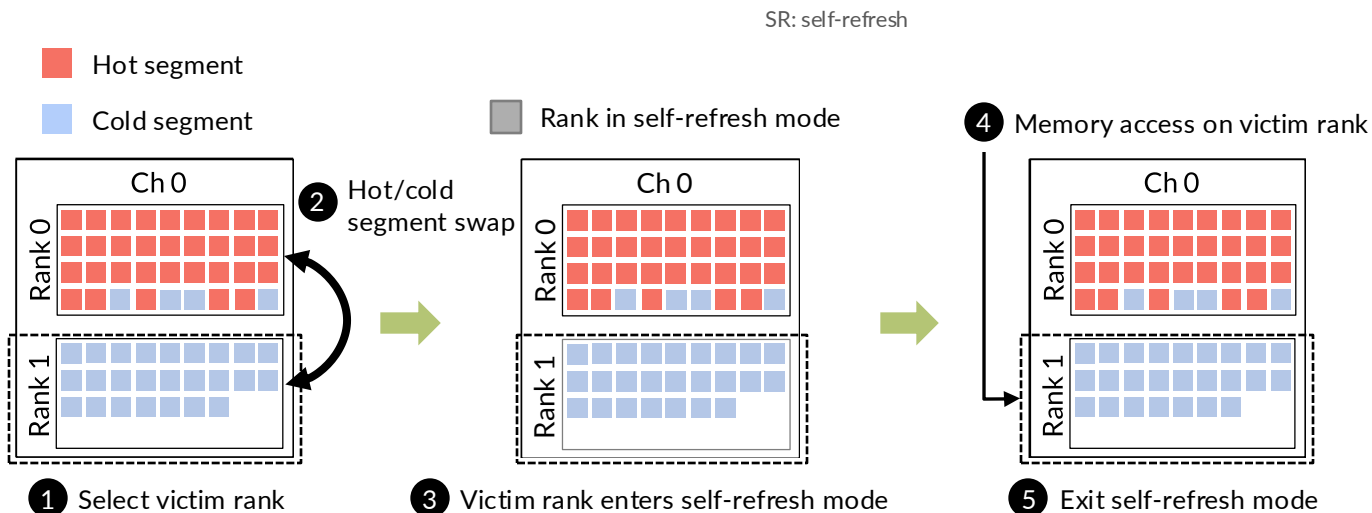
6 Exit power-down mode

Rank group in power down mode

Rank group indicates a set of ranks with the same rank index across all memory channels

# Technique #2: Hotness-aware Self-refresh

- Periodically *swapping cold-hot segments* across ranks
- Maximizing the rank idle time for entering **SR** mode

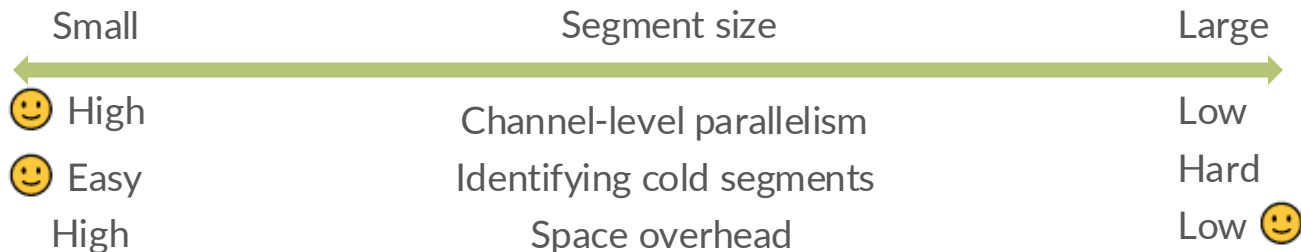


➡ Additional *migration table* to track segment access

# Trade-off in Address Mapping Granularity

- Smaller segment size (address mapping granularity)
  - Higher channel-level parallelism
    - DTL requires channel-interleaving at segment granularity
    - Placing channel bits in the lower bits enhances channel-level parallelism

We choose **2MB** for the address mapping granularity




# Outline

- Overview
- DRAM Translation Layer
  - Address translation
  - Two-power management techniques
    - Technique #1: rank-level power-down
    - Technique #2: hotness-aware self-refresh
- **Evaluation**
- Conclusion





# Methodology

- **Real machine** measurement for rank-level power-down scheme
  - 400 VMs from Microsoft Azure VM traces<sup>[1]</sup>
  - Measures background and active power consumption with different number of **active ranks**
- Custom **trace-driven simulation** for hotness-aware self-refresh scheme
  - Collects memory traces by Intel Pintool<sup>[2]</sup>
  - Generates post-cache traces via cache simulation
  - Randomly mixes post-cache traces
- Workloads
  - Cloudsuite 4.0<sup>[3]</sup>  CloudSuite

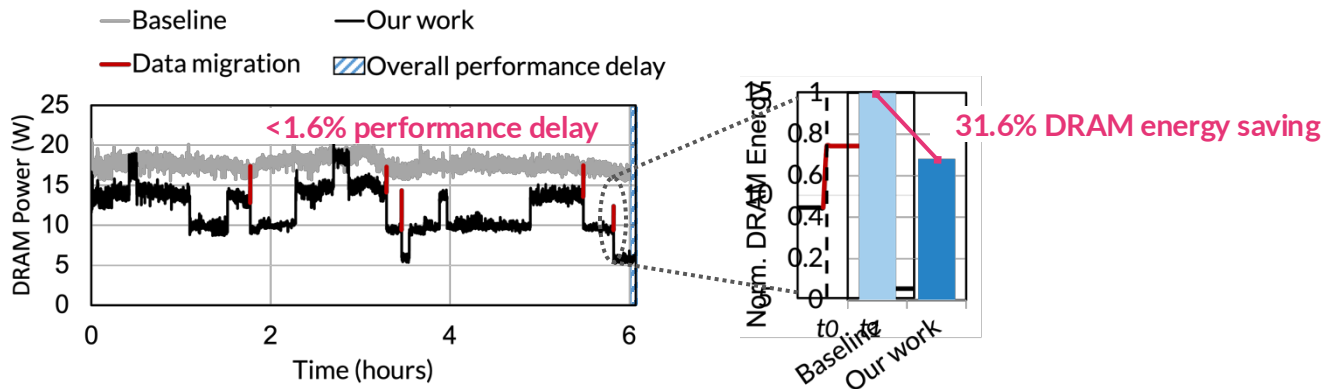
[1] Cortez et al., “Resource Central: Understanding and Predicting Workloads for Improved Resource Management in Large Cloud Platforms”, SOSP, 2017.

[2] Luk et al., “Pin: Building Customized Program Analysis Tools with Dynamic Instrumentation”, ACM SIGPLAN Notices, 2015

[3] <https://github.com/parsa-epfl/cloudsuite>.



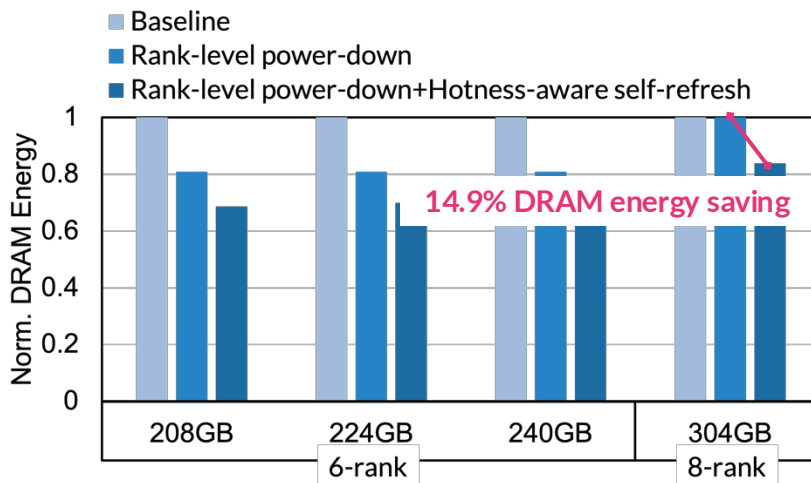
# Rank-level Power-down



- **31.6%** DRAM energy savings with **<1.6%** performance loss
- Data migration delays MPSM entry
  - Migration opportunistically utilize the unused bandwidth by foreground
  - 24GB data migration (~1.3s)

# Total Energy Savings

- Applying hotness-aware self-refresh scheme after rank-level power-down results in an additional reduction in DRAM energy consumption by up to **14.9%** with negligible performance loss



*N-rank* configuration refers to a configuration in which *N* active ranks are utilized to provide enough memory capacity



# The Case for DRAM Translation Layer (DTL): Enhancing Datacenter Efficiency, Reliability, and Beyond



Wenjing Jin

✉ [wenjing.jin@snu.ac.kr](mailto:wenjing.jin@snu.ac.kr)

Jongsung Lee

✉ [leitia@snu.ac.kr](mailto:leitia@snu.ac.kr)

Wonsuk Jang

✉ [skylark0827@snu.ac.kr](mailto:skylark0827@snu.ac.kr)

Soosung Kim

✉ [soosungkim@snu.ac.kr](mailto:soosungkim@snu.ac.kr)

Haneul Park

✉ [skyp0714@snu.ac.kr](mailto:skyp0714@snu.ac.kr)

Jae W. Lee

✉ [jaewlee@snu.ac.kr](mailto:jaewlee@snu.ac.kr)

# Address Mappings: Allocation Unit (AU)

- HSN part is further divided into HostID|AU ID|AU Offset
  - AU is the minimum memory allocation unit to each VM
  - Default AU size: 2GB (minimum vMemory allocation size per instance in top-three cloud vendors)

