WARM

Improving NAND Flash Memory Lifetime with Write-hotness Aware Retention Management

***Yixin Luo**, Yu Cai, Saugata Ghose, Jongmoo Choi*, Onur Mutlu*

*Carnegie Mellon University, *Dankook University*

SAFARI

Carnegie Mellon

Executive Summary

- Flash memory can achieve **50x endurance improvement by relaxing retention time using refresh** *[Cai+ ICCD '12]*
- *Problem*: **Refresh consumes the majority of endurance improvement**
- *Goal*: Reduce refresh overhead to increase flash memory lifetime
- *Key Observation*: **Refresh is unnecessary for *write-hot data***
- *Key Ideas of Write-hotness Aware Retention Management (WARM)*
  - **Physically partition write-hot pages and write-cold pages** within the flash drive
  - **Apply *different* policies** (garbage collection, wear-leveling, refresh) to each group
- *Key Results*
  - WARM w/o refresh **improves lifetime by 3.24x**
  - WARM w/ adaptive refresh **improves lifetime by 12.9x** (1.21x over refresh only)

Outline

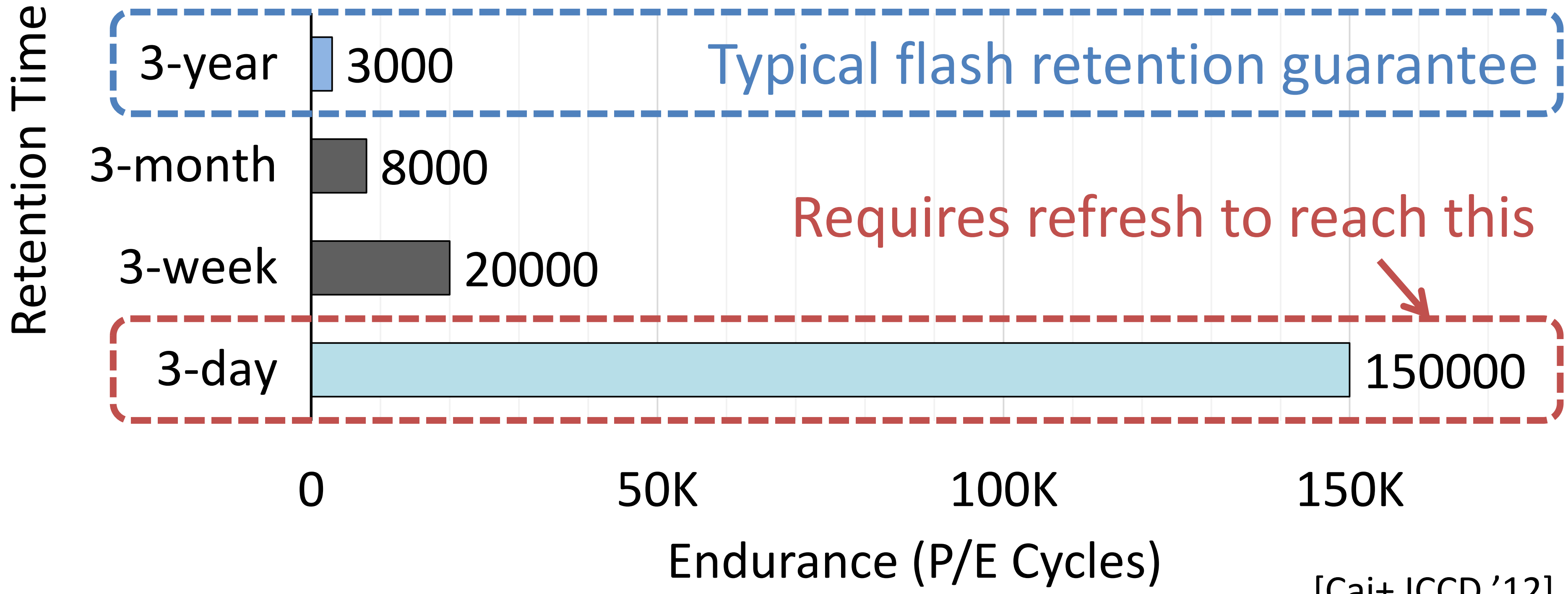
- *Problem and Goal*
- *Key Observations*
- *WARM: Write-hotness Aware Retention Management*
- *Results*
- *Conclusion*

Outline

- *Problem and Goal*
- *Key Observations*
- *WARM: Write-hotness Aware Retention Management*
- *Results*
- *Conclusion*

Retention Time Relaxation for Flash Memory

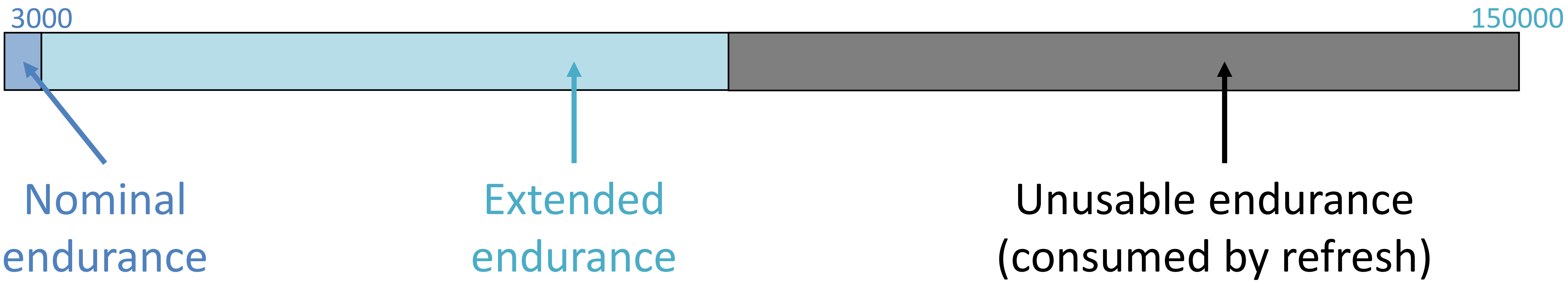
- Flash memory has limited *write endurance*
- *Retention time* significantly affects endurance
  - The duration for which flash memory correctly holds data



[Cai+ ICCD '12]

NAND Flash Refresh

- *Flash Correct and Refresh (FCR), Adaptive Rate FCR (ARFCR) [Cai+ ICCD '12]*



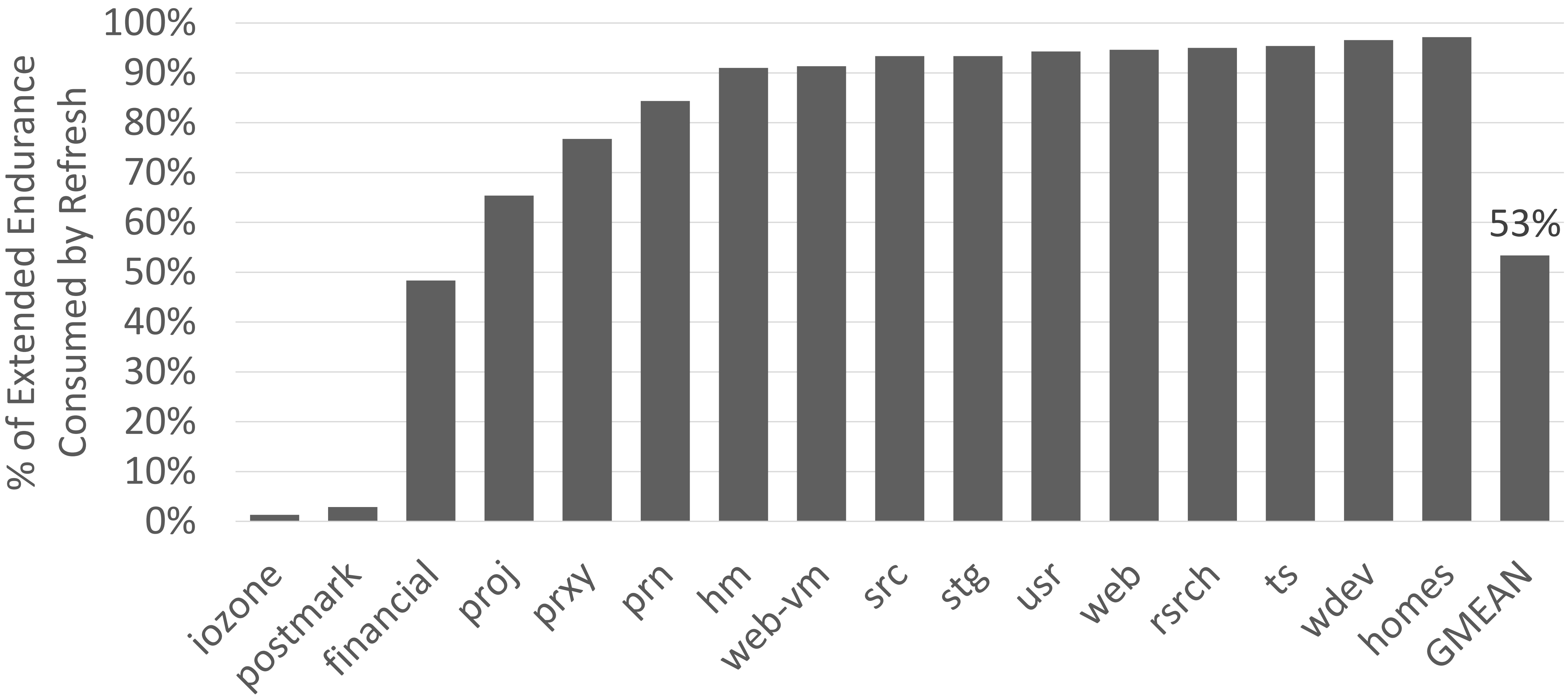
Problem: Flash refresh operations reduce extended lifetime

Goal: Reduce refresh overhead, improve flash lifetime

Outline

- *Problem and Goal*
- *Key Observations*
- *WARM: Write-hotness Aware Retention Management*
- *Results*
- *Conclusion*

Observation 1: Refresh Overhead is High



Observation 2: Write-Hot Pages Can Skip Refresh

Update

Invalid Page

Invalid Page

Write-Hot Page

Skip Refresh

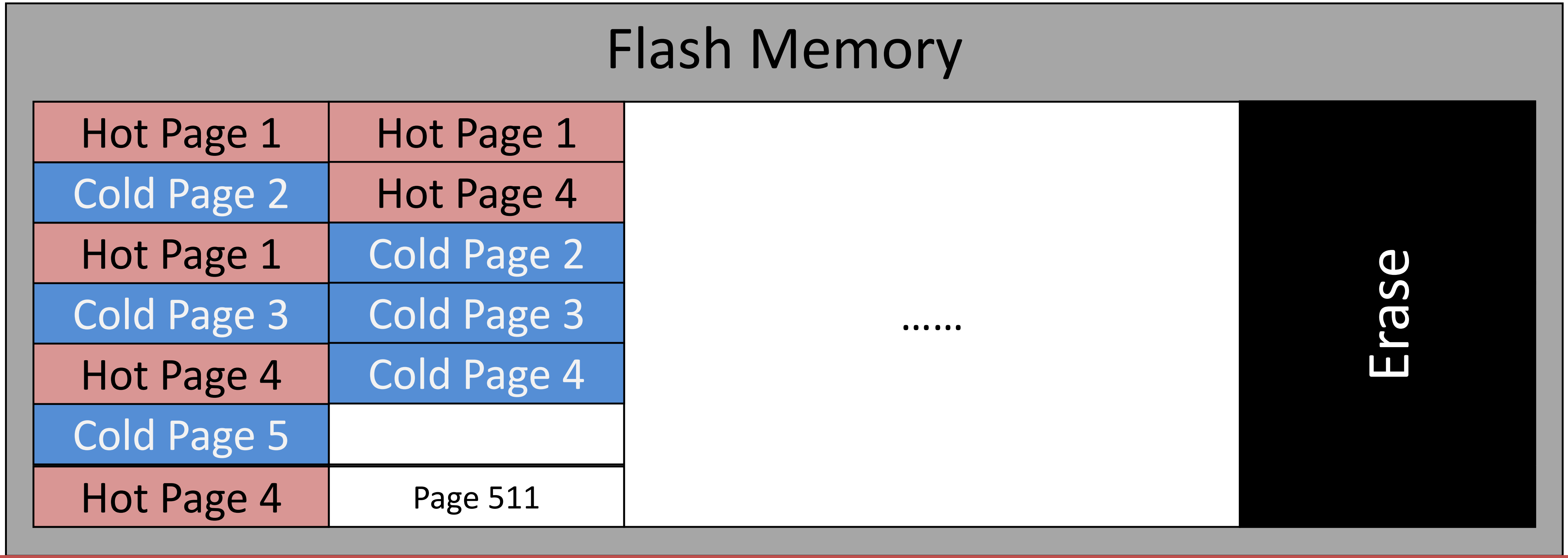
Retention Effect

Invalid Page

Write-Cold Page

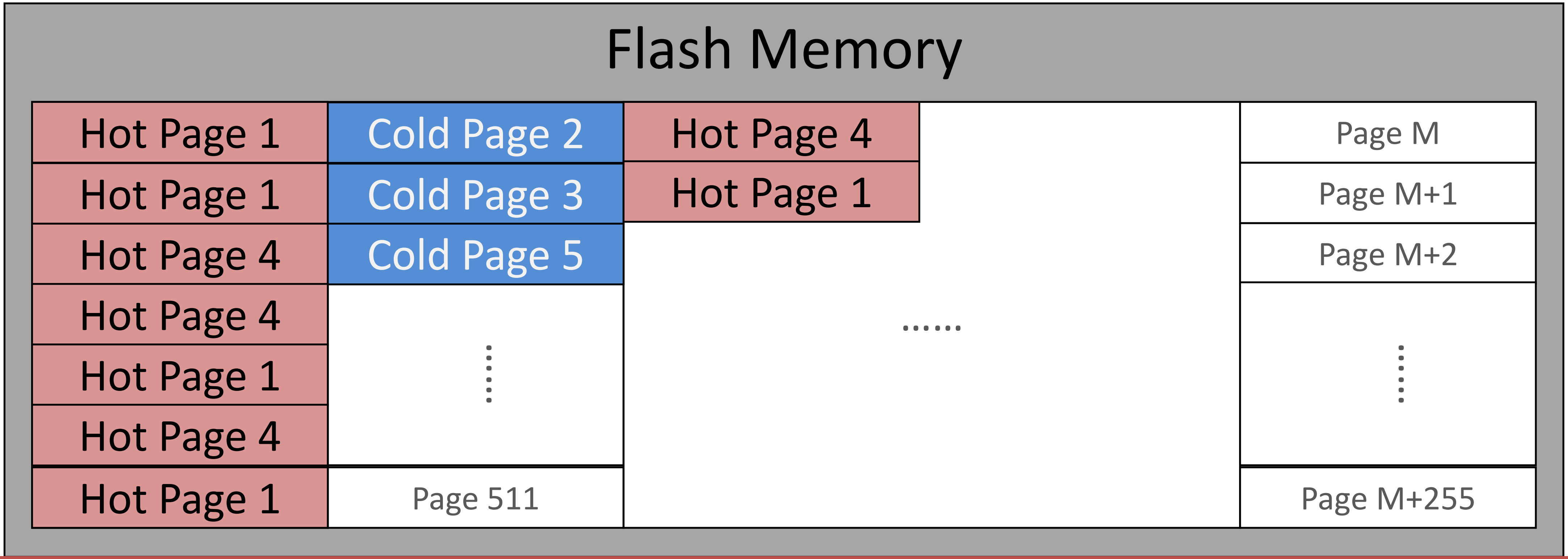
Need Refresh

Conventional Write-Hotness Oblivious Management



Unable to relax retention time for blocks with write-hot and cold pages

Key Idea: Write-Hotness Aware Management



Can relax retention time for blocks with write-hot pages only

Outline

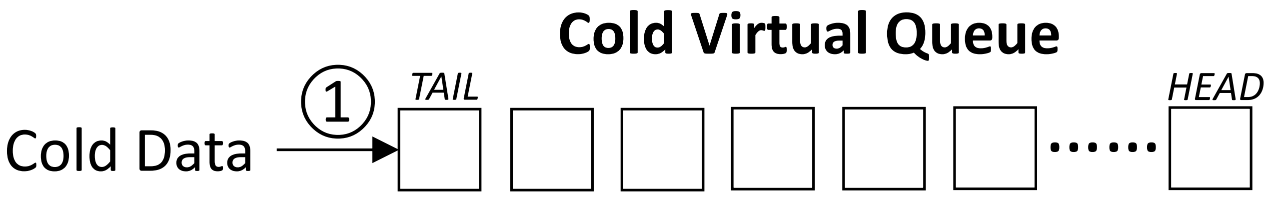
- *Problem and Goal*
- *Key Observations*
- *WARM: Write-hotness Aware Retention Management*
- *Results*
- *Conclusion*

WARM Overview

- Design Goal:
  - Relax retention time w/o refresh for write-hot data only

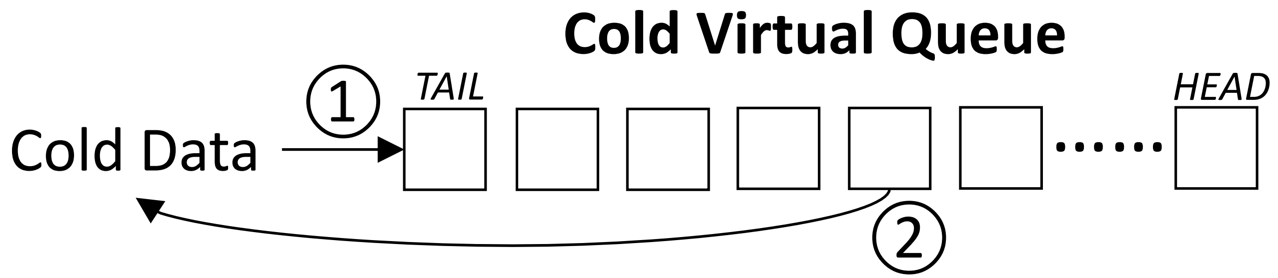
- WARM: Write-hotness Aware Retention Management
  - Write-hot/write-cold data partitioning algorithm
  - Write-hotness aware flash policies
    - *Partition write-hot and write-cold data into separate blocks*
    - *Skip refreshes for write-hot blocks*
    - *More efficient garbage collection and wear-leveling*

Write-Hot/Write-Cold Data Partitioning Algorithm



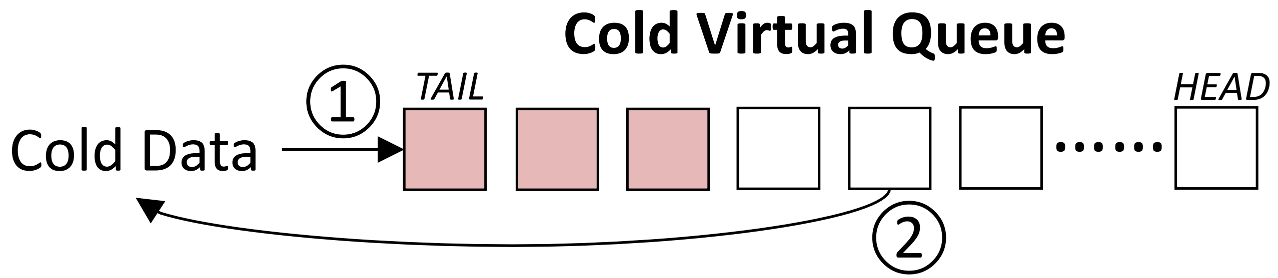
1. Initially, all data is cold and is stored in the cold virtual queue.

Write-Hot/Write-Cold Data Partitioning Algorithm



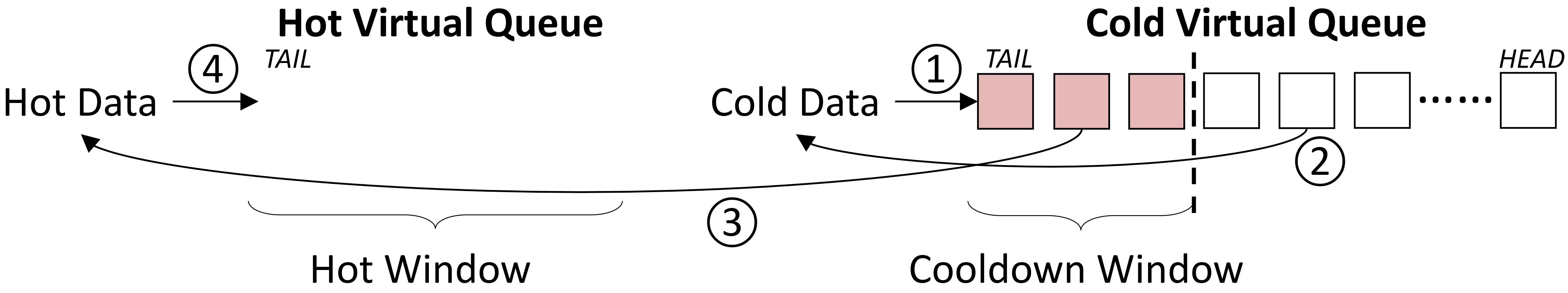
2. On a write operation, the data is pushed to the tail of the cold virtual queue.

Write-Hot/Write-Cold Data Partitioning Algorithm



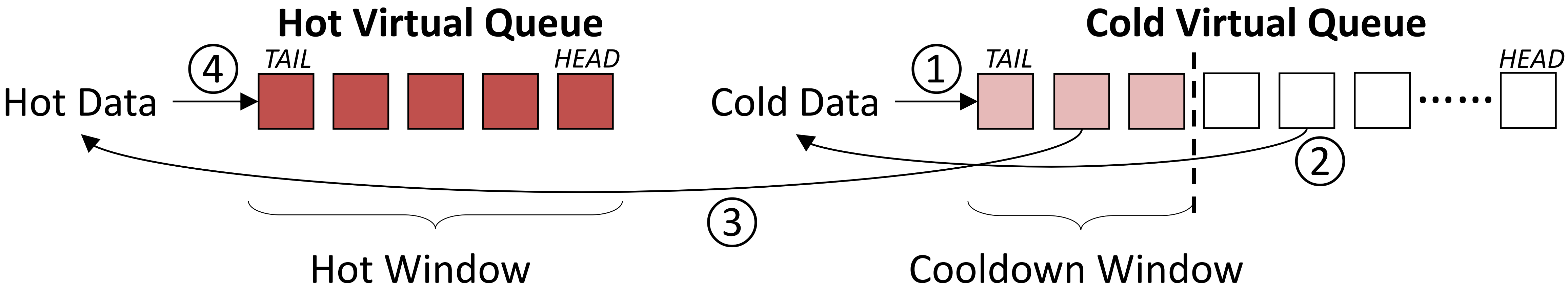
Recently-written data is at the tail of cold virtual queue.

Write-Hot/Write-Cold Data Partitioning Algorithm



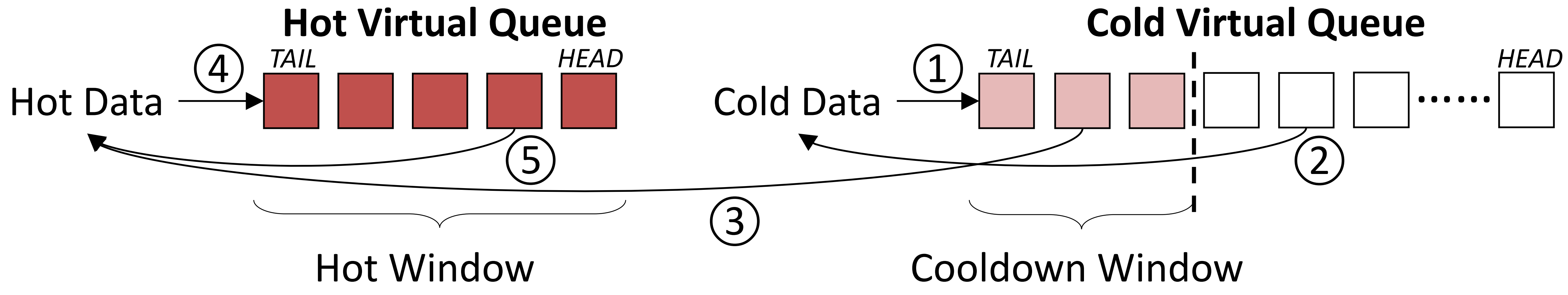
3, 4. On a write hit in the cooldown window,
the data is promoted to the hot virtual queue.

Write-Hot/Write-Cold Data Partitioning Algorithm



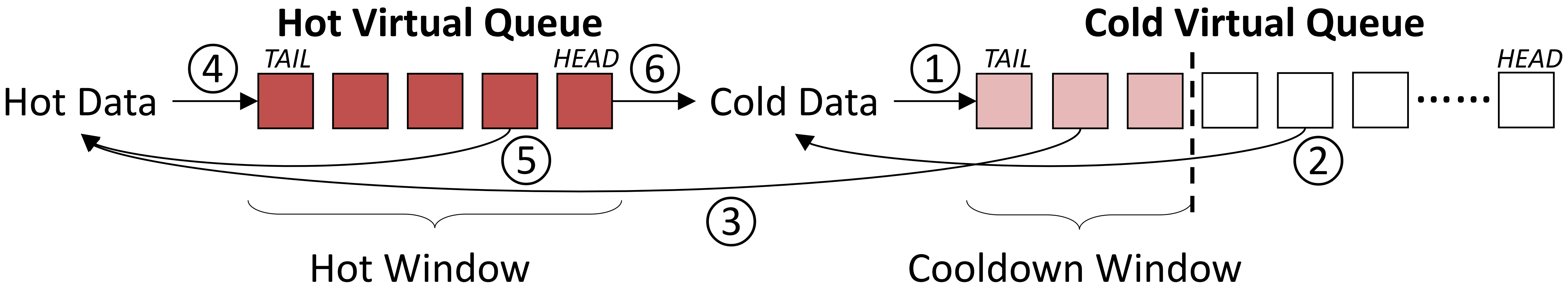
Data is sorted by write-hotness in the hot virtual queue.

Write-Hot/Write-Cold Data Partitioning Algorithm



5. On a write hit in hot virtual queue, the data is pushed to the tail.

Write-Hot/Write-Cold Data Partitioning Algorithm

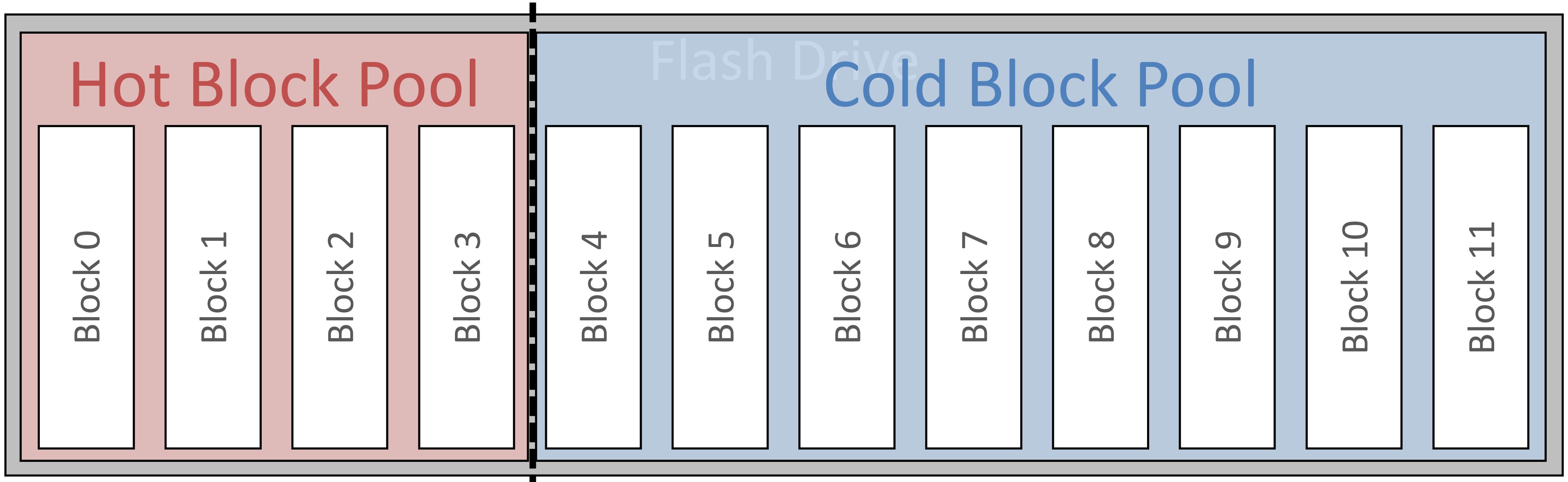


6. Unmodified hot data will be demoted to the cold virtual queue.

Conventional Flash Management Policies

- *Flash Translation Layer (FTL)*
  - Map data to erased blocks
  - Translate logical page number to physical page number
- *Garbage Collection*
  - Triggered before erasing a victim block
  - Remap all valid data on the victim block
- *Wear-leveling*
  - Triggered to balance wear-level among blocks

Write-Hotness Aware Flash Policies



- Write-hot data → naturally relaxed retention time
- Program in block order
- Garbage collect in block order
- All blocks naturally wear-leveled

- Write-cold data → lower write frequency, less wear-out
- Conventional garbage collection
- Conventional wear-leveling algorithm

Dynamically Sizing the Hot and Cold Block Pools

All blocks are divided between the hot and cold block pools

1. Find the maximum hot pool size
2. Reduce hot virtual queue size to maximize cold pool lifetime
3. Size the cooldown window to minimize ping-ponging of data between the two pools

Outline

- *Problem and Goal*
- *Key Observations*
- *WARM: Write-hotness Aware Retention Management*
- *Results*
- *Conclusion*

Methodology

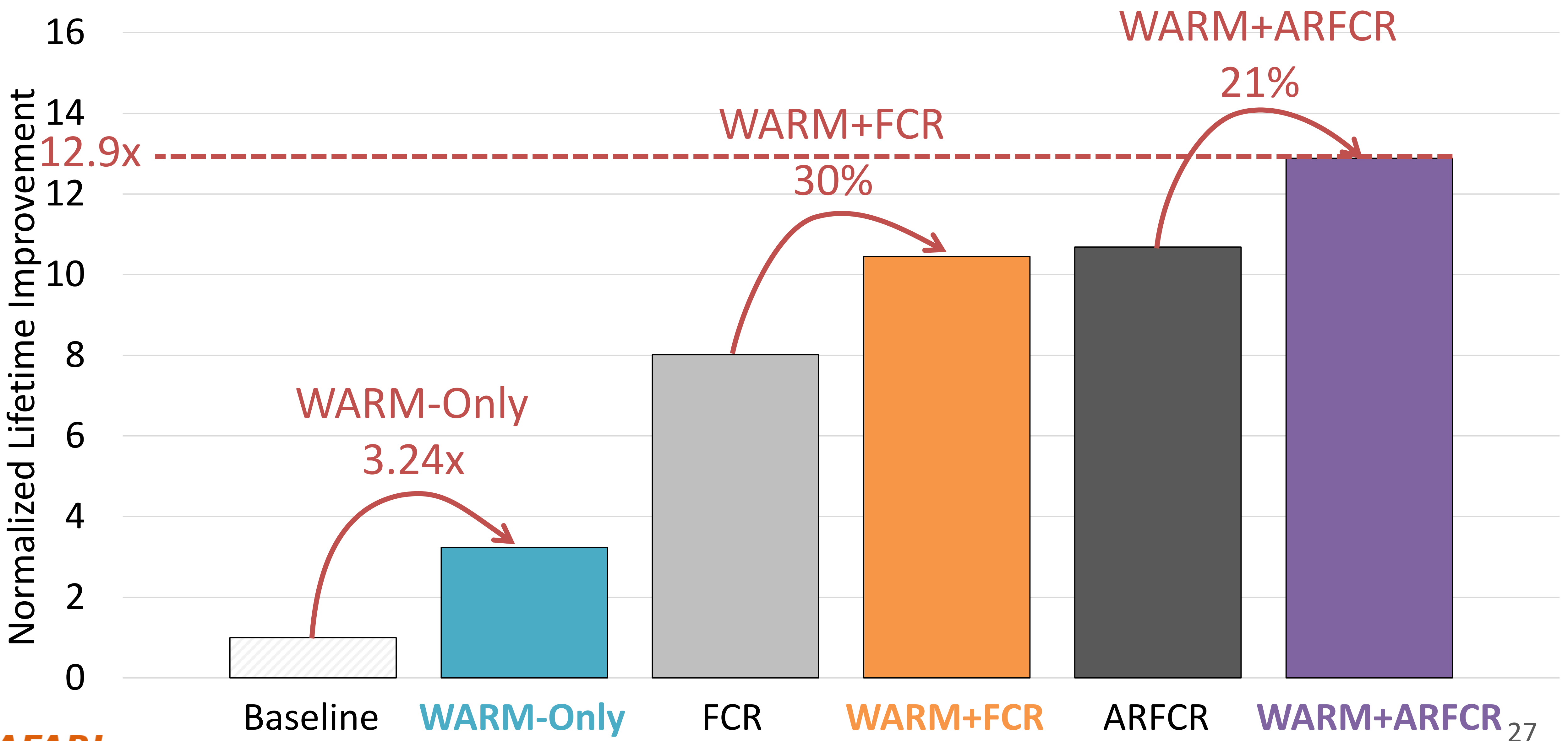
- *DiskSim 4.0 + SSD model*

Parameter	Value
Page read to register latency	25 µs
Page write from register latency	200 µs
Block erase latency	1.5 ms
Data bus latency	50 µs
Page/block size	8 KB/1 MB
Die/package size	8 GB/64 GB
Total capacity	256 GB
Over-provisioning	15%
Endurance for 3-year retention time	3,000 PEC
Endurance for 3-day retention time	150,000 PEC

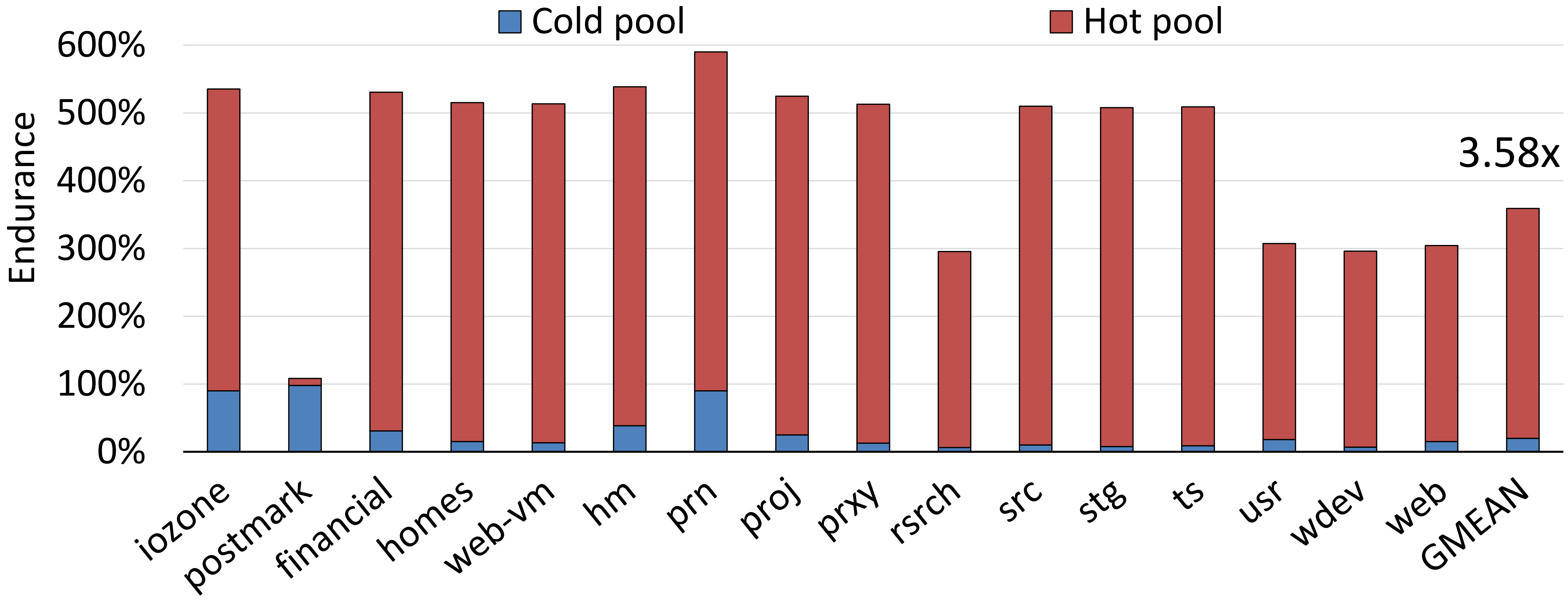
WARM Configurations

- *WARM-Only*
  - Relax retention time in hot block pool only
  - No refresh needed
- *WARM+FCR*
  - First apply **WARM-Only**
  - Then *also* relax retention time in cold block pool
  - Refresh cold blocks every 3 days
- *WARM+ARFCR*
  - Relax retention time in both hot and cold block pools
  - Adaptively increase the refresh frequency over time

Flash Lifetime Improvements

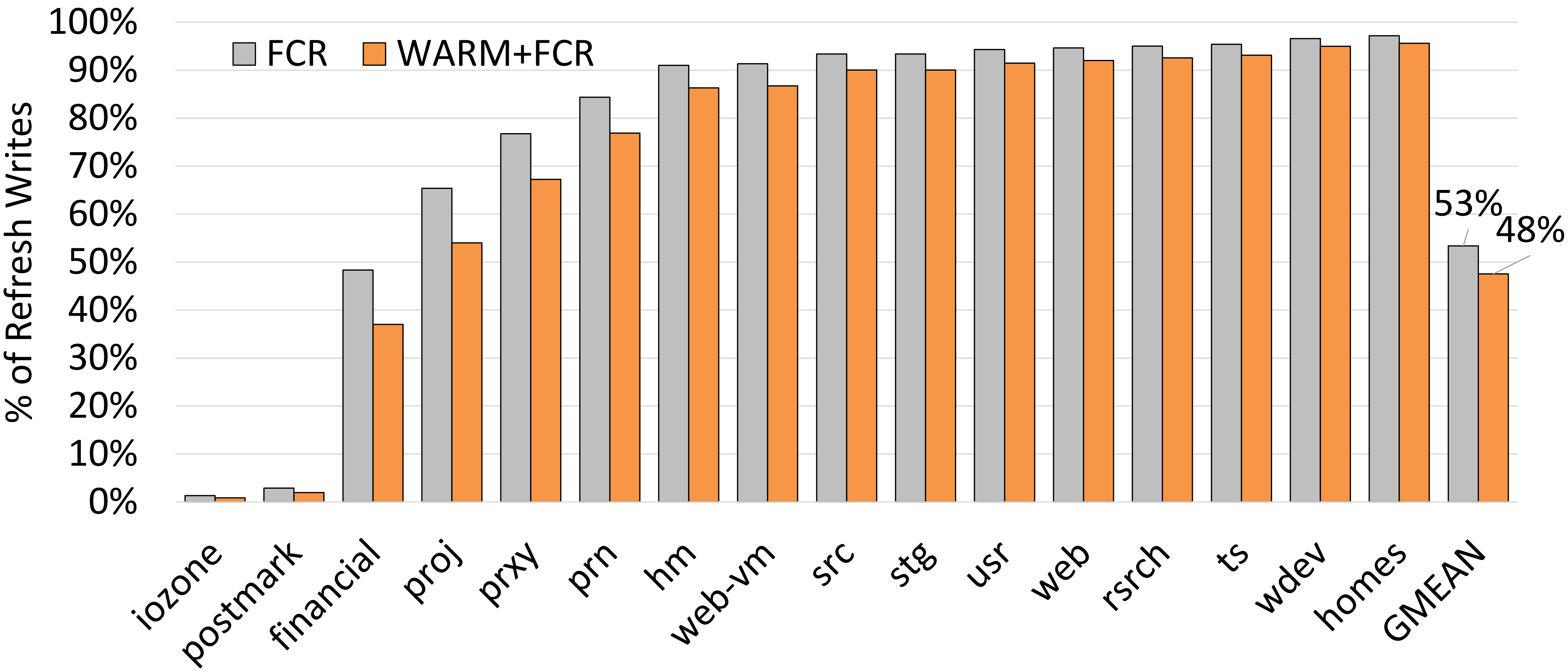


WARM-Only Endurance Improvement

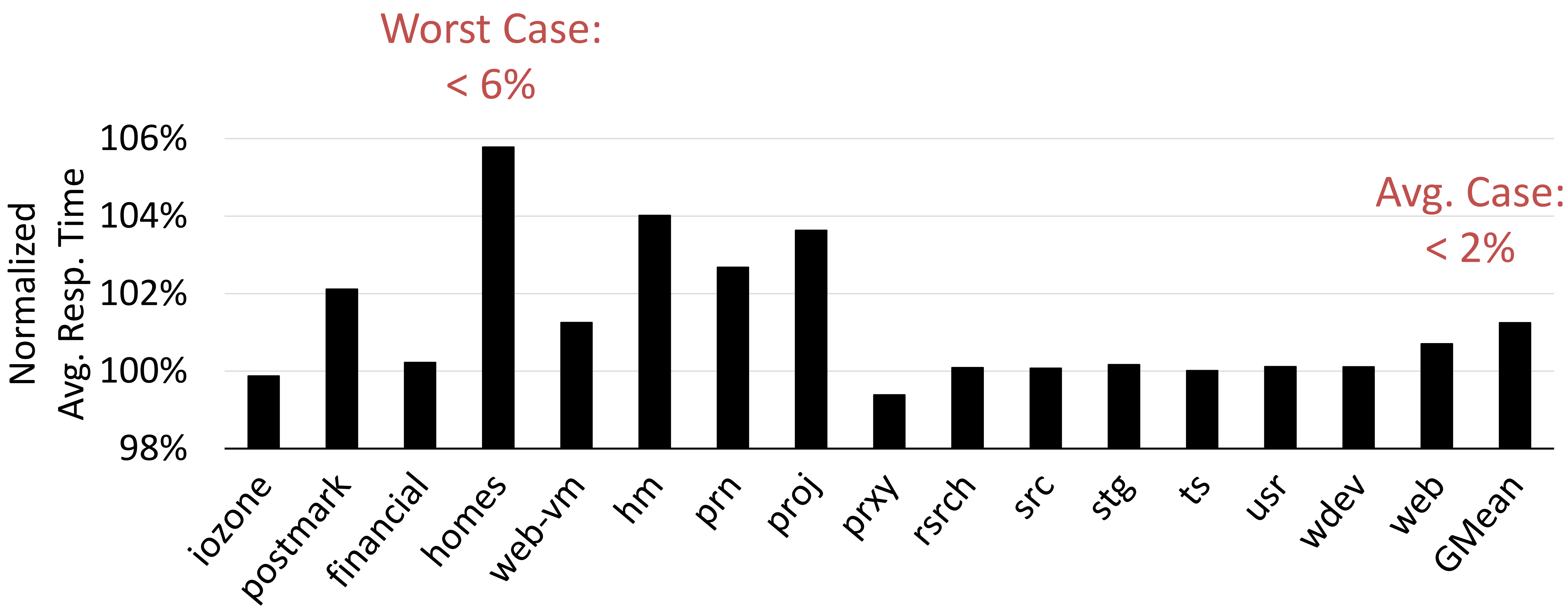


SAFARI

WARM+FCR Refresh Operation Reduction



WARM Performance Impact



Other Results in the Paper

- *Breakdown of write frequency into host writes, garbage collection writes, refresh writes in the hot and cold block pools*
  - WARM reduces refresh writes significantly while having low garbage collection overhead

- *Sensitivity to different capacity over-provisioning amounts*
  - WARM improves flash lifetime more as over-provisioning increases

- *Sensitivity to different refresh intervals*
  - WARM improves flash lifetime more as refresh frequency increases

Outline

- *Problem and Goal*
- *Key Observations*
- *WARM: Write-hotness Aware Retention Management*
- *Results*
- *Conclusion*

Conclusion

- Flash memory can achieve **50x endurance improvement by relaxing retention time using refresh** *[Cai+ ICCD '12]*
- *Problem*: **Refresh consumes the majority of endurance improvement**
- *Goal*: Reduce refresh overhead to increase flash memory lifetime
- *Key Observation*: **Refresh is unnecessary for *write-hot data***
- *Key Ideas of Write-hotness Aware Retention Management (WARM)*
  - **Physically partition write-hot pages and write-cold pages** within the flash drive
  - **Apply *different* policies** (garbage collection, wear-leveling, refresh) to each group
- *Key Results*
  - WARM w/o refresh **improves lifetime by 3.24x**
  - WARM w/ adaptive refresh **improves lifetime by 12.9x** (1.21x over refresh only)

Other Work by SAFARI on Flash Memory

- J. Meza, Q. Wu, S. Kumar, and O. Mutlu. *A Large-Scale Study of Flash Memory Errors in the Field*, SIGMETRICS 2015.
- Y. Cai, Y. Luo, S. Ghose, E. F. Haratsch, K. Mai, O. Mutlu. *Read Disturb Errors in MLC NAND Flash Memory: Characterization and Mitigation*, DSN 2015.
- Y. Cai, Y. Luo, E. F. Haratsch, K. Mai, O. Mutlu. *Data Retention in MLC NAND Flash Memory: Characterization, Optimization and Recovery*, HPCA 2015.
- Y. Cai, G. Yalcin, O. Mutlu, E. F. Haratsch, O. Unsal, A. Cristal, K. Mai. *Neighbor-Cell Assisted Error Correction for MLC NAND Flash Memories*, SIGMETRICS 2014.
- Y. Cai, O. Mutlu, E. F. Haratsch, K. Mai. *Program Interference in MLC NAND Flash Memory: Characterization, Modeling, and Mitigation*, ICCD 2013.
- Y. Cai, G. Yalcin, O. Mutlu, E. F. Haratsch, A. Cristal, O. Unsal, K. Mai. *Error Analysis and Retention-Aware Error Management for NAND Flash Memory*, Intel Technology Jrnl. (ITJ), Vol. 17, No. 1, May 2013.
- Y. Cai, E. F. Haratsch, O. Mutlu, K. Mai. *Threshold Voltage Distribution in MLC NAND Flash Memory: Characterization, Analysis and Modeling*, DATE 2013.
- Y. Cai, G. Yalcin, O. Mutlu, E. F. Haratsch, A. Cristal, O. Unsal, K. Mai. *Flash Correct-and-Refresh: Retention-Aware Error Management for Increased Flash Memory Lifetime*, ICCD 2012.
- Y. Cai, E. F. Haratsch, O. Mutlu, K. Mai. *Error Patterns in MLC NAND Flash Memory: Measurement, Characterization, and Analysis*, DATE 2012.

WARM

Improving NAND Flash Memory Lifetime with Write-hotness Aware Retention Management

***Yixin Luo**, Yu Cai, Saugata Ghose, Jongmoo Choi*, Onur Mutlu*

*Carnegie Mellon University, *Dankook University*

SAFARI

Carnegie Mellon