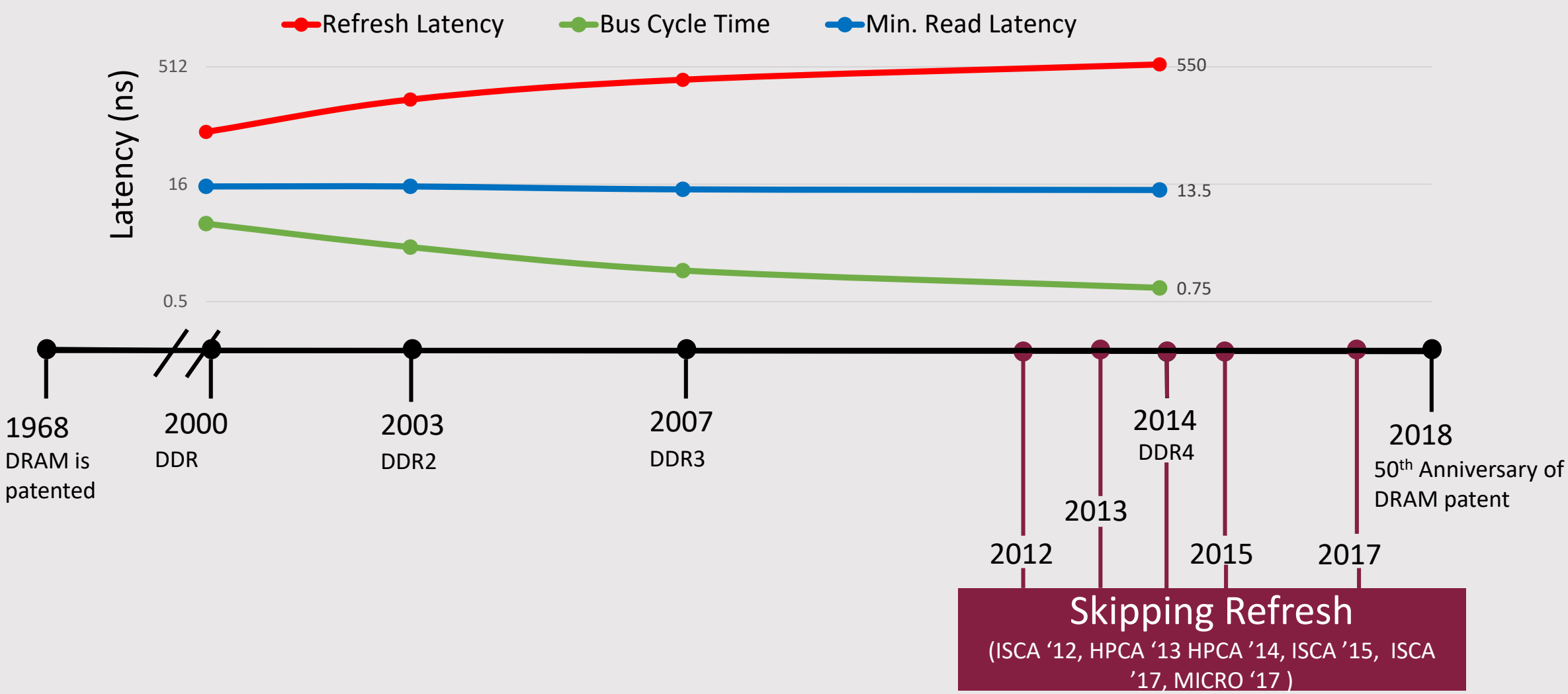
Nonblocking Memory Refresh

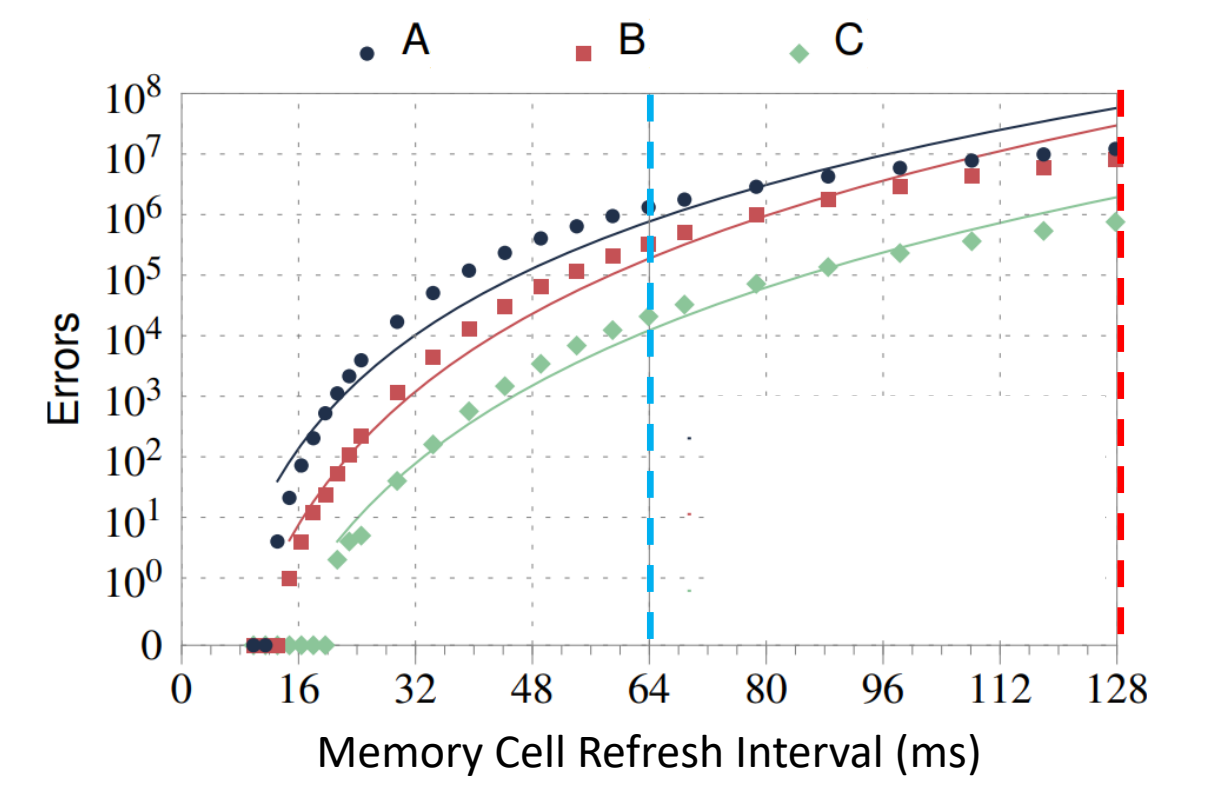
Kate Nguyen, Kehan Lyu, **Xianze Meng**, Vilas Sridharan, Xun Jian

History of DRAM



Issues with Skipping Refresh

Tested DRAM chips from different manufacturers



Y. Kim, R. Daly, J. Kim, C. Fallin, J. H. Lee, D. Lee, C. Wilkerson, K. Lai, and O. Mutlu, "Flipping bits in memory without accessing them: An experimental study of dram disturbance errors," in 2014 ACM/IEEE 41st International Symposium on Computer Architecture (ISCA), pp. 361–372, June 2014.

Data Center ▸ Servers

'Rowhammer' attack flips bits in memory to root Linux

Ouch! Google cracks ...

Once thought safe, DDR4 memory shown to be vulnerable to "Rowhammer"

New research finds "bitflipping" attack ...

New paper alleges servers, some DDR4 DRAM still vulnerable to critical Rowhammer attack

March 16, 2016 at 4:29 pm | 6 Comments

DRAM Rowhammer vulnerability Leads to Kernel Privilege Escalation

Monday, March 09, 2015 Swati Khandelwal

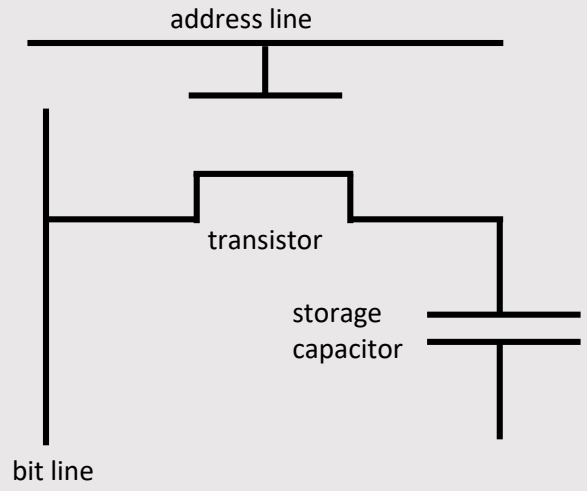
New Rowhammer Attack Bypasses Previously Proposed Countermeasures

By Catalin Cimpanu October 4, 2017

Skipping refresh reduces memory security

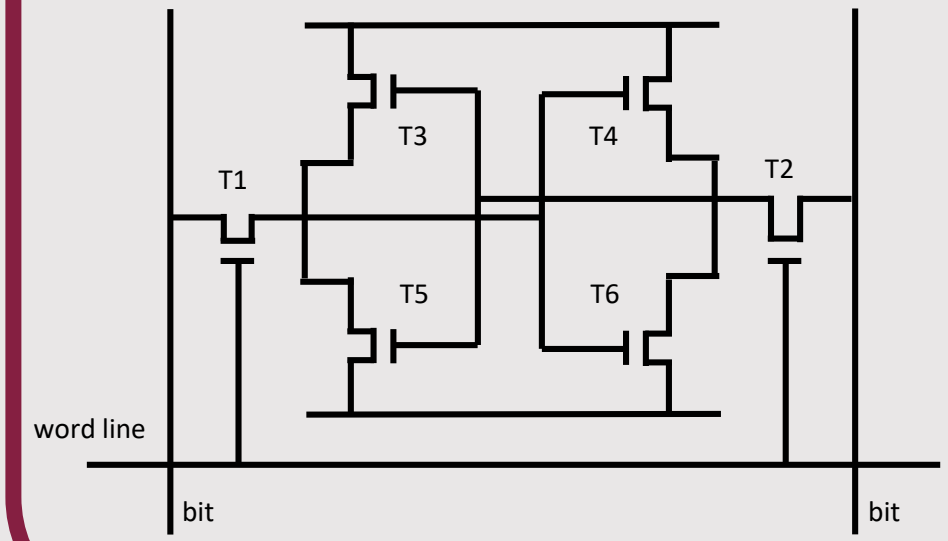
Why DRAM Refresh Hurts Performance

DRAM



Blocking Refresh

SRAM

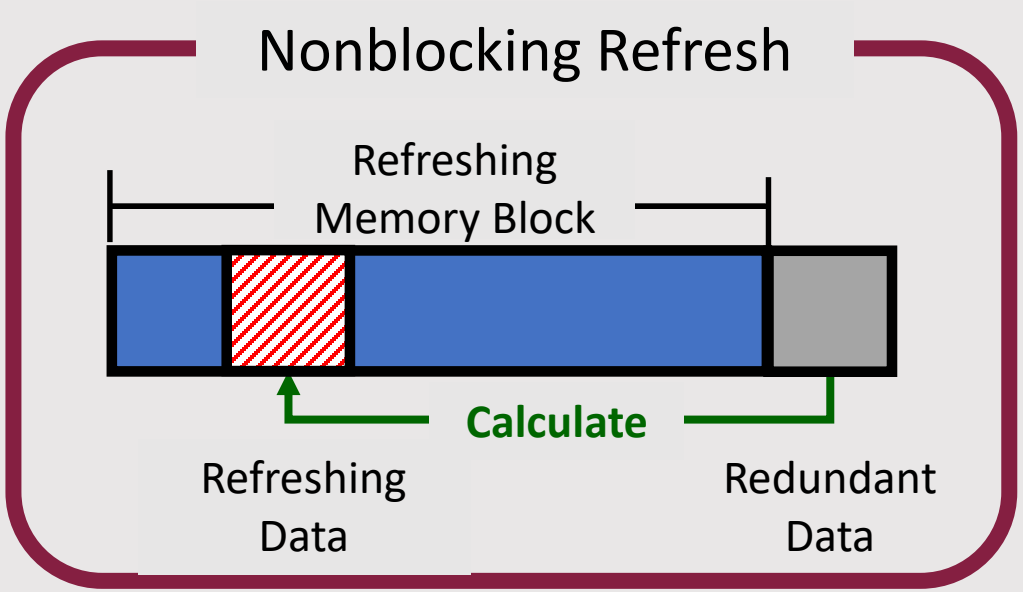


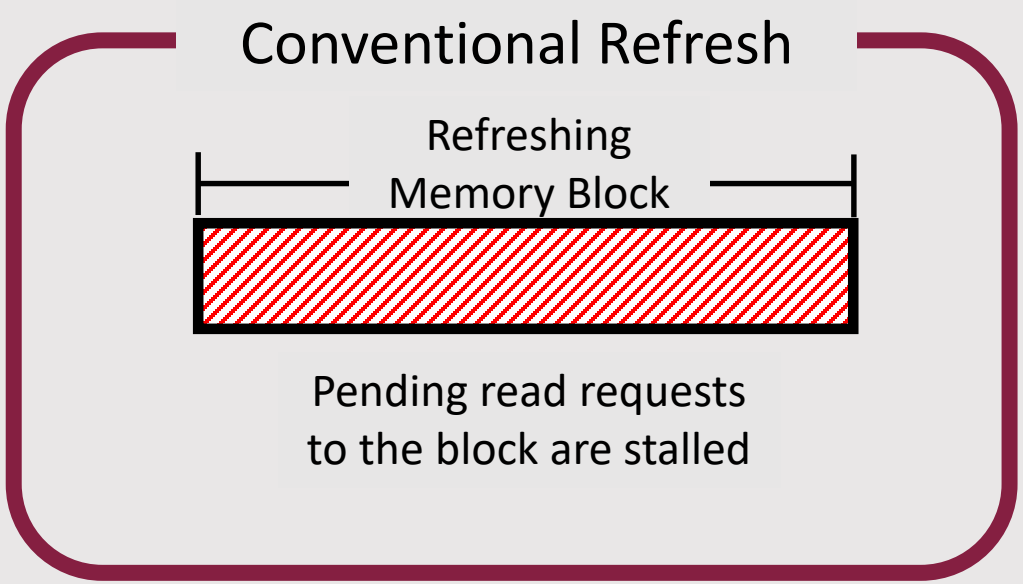
Nonblocking Refresh

Our Proposal: Nonblocking Refresh

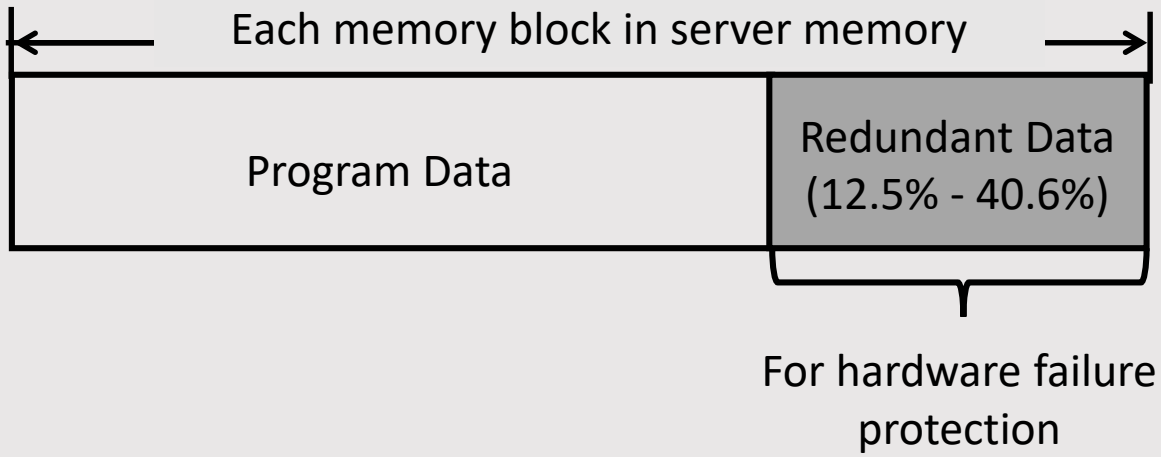
- Improve performance while retaining the same level of security as the conventional baseline.
- Transform DRAM refresh into the static/background refresh in SRAM at the system level.
- Refresh DRAM in the background without stalling read accesses to refreshing memory blocks.

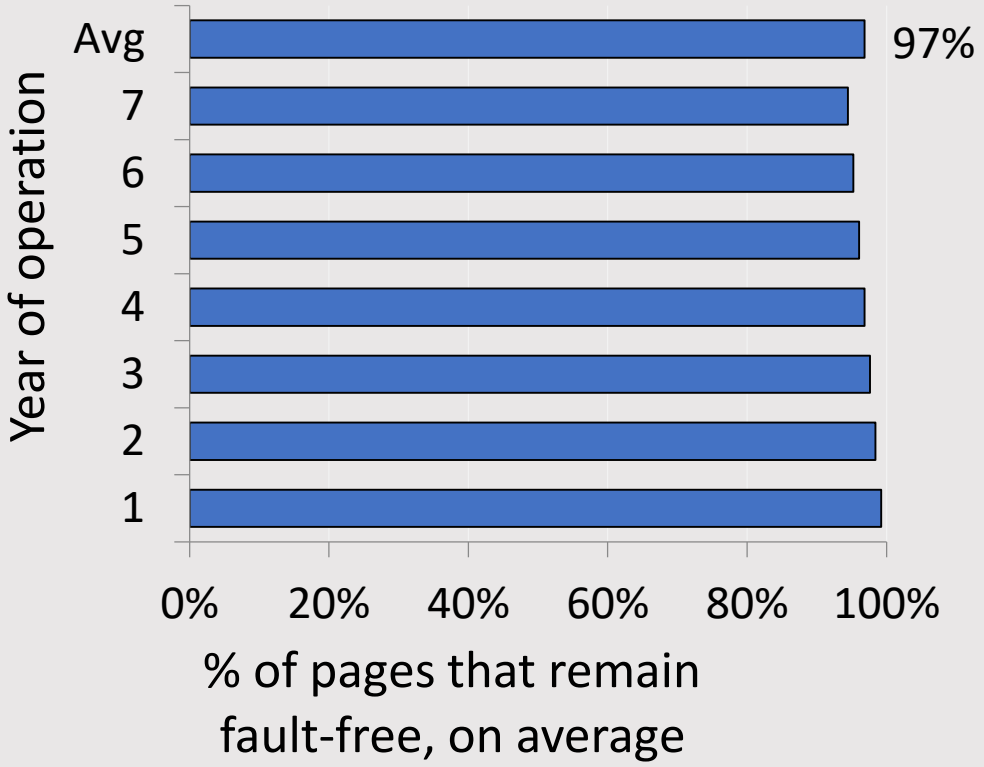
How Nonblocking Refresh Works





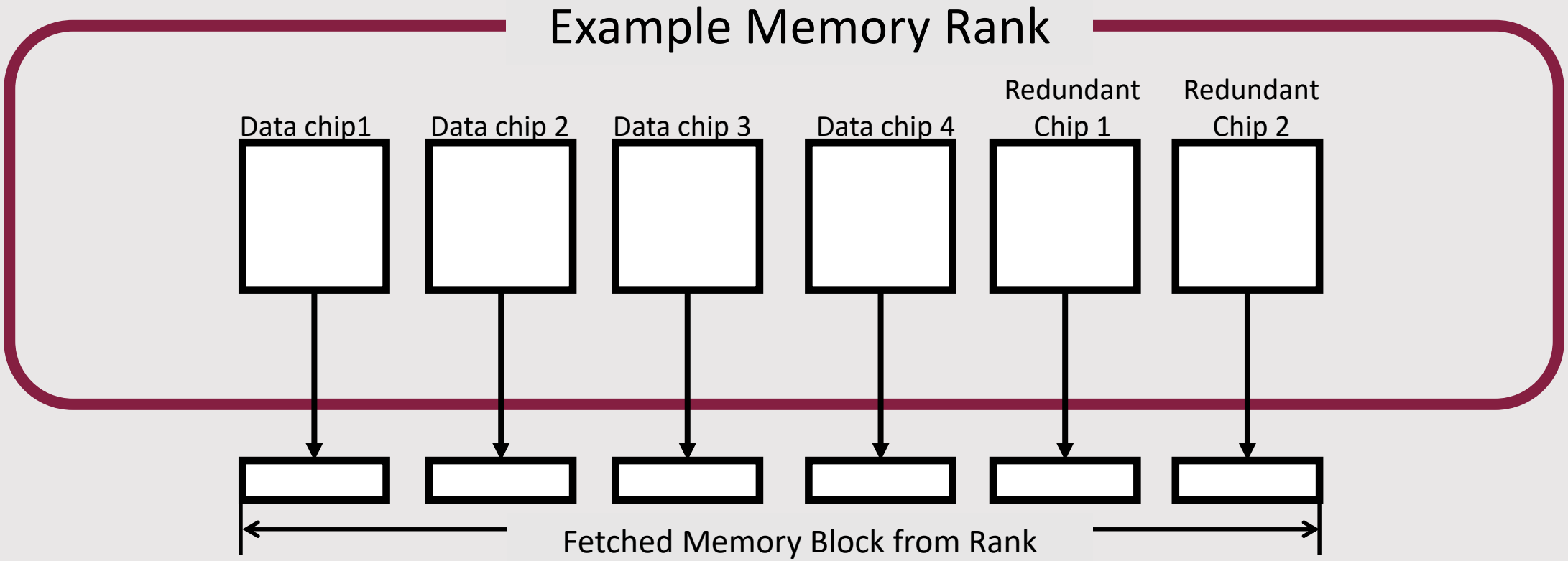
Leveraging Existing Redundant Data for Free



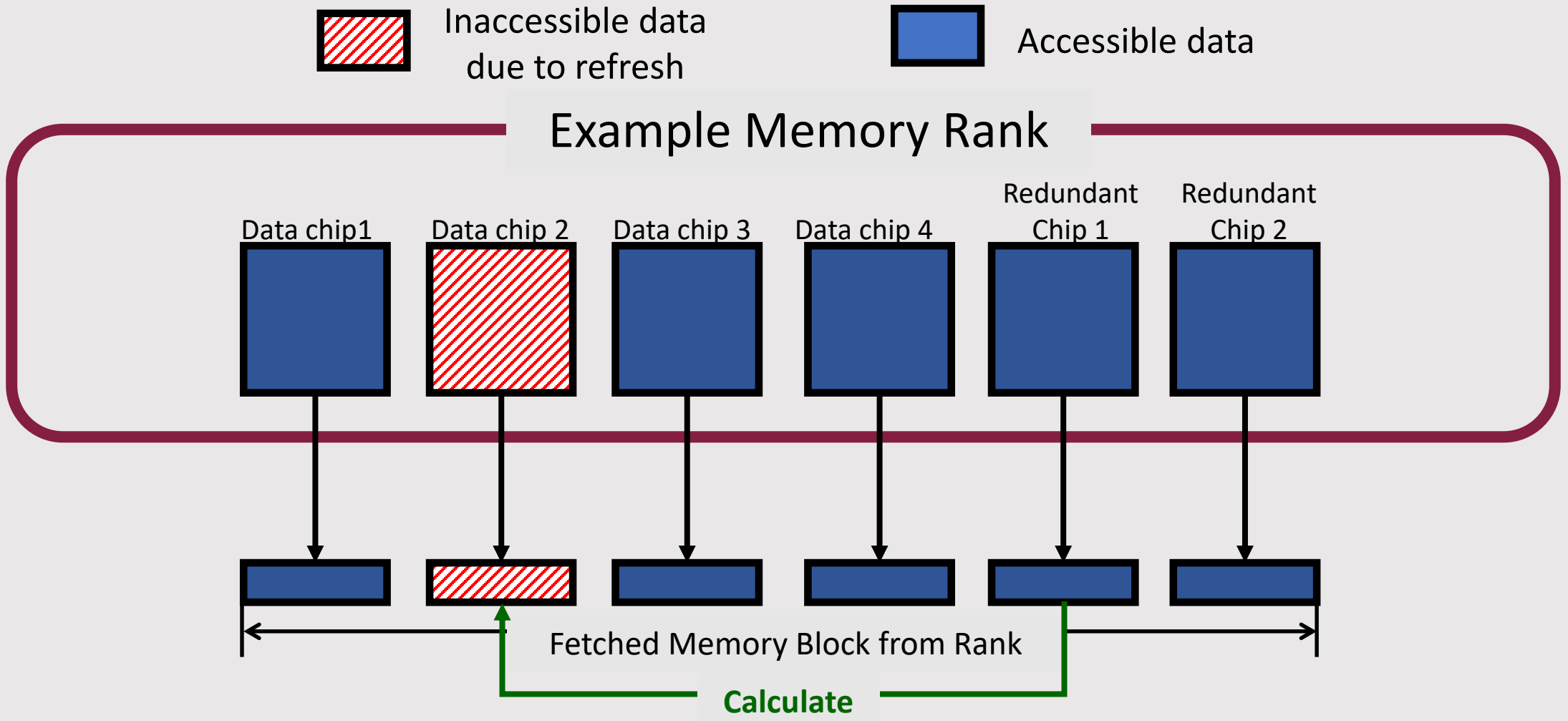


For server systems, Nonblocking Refresh can leverage existing underutilized redundant data ***without storage overheads***.

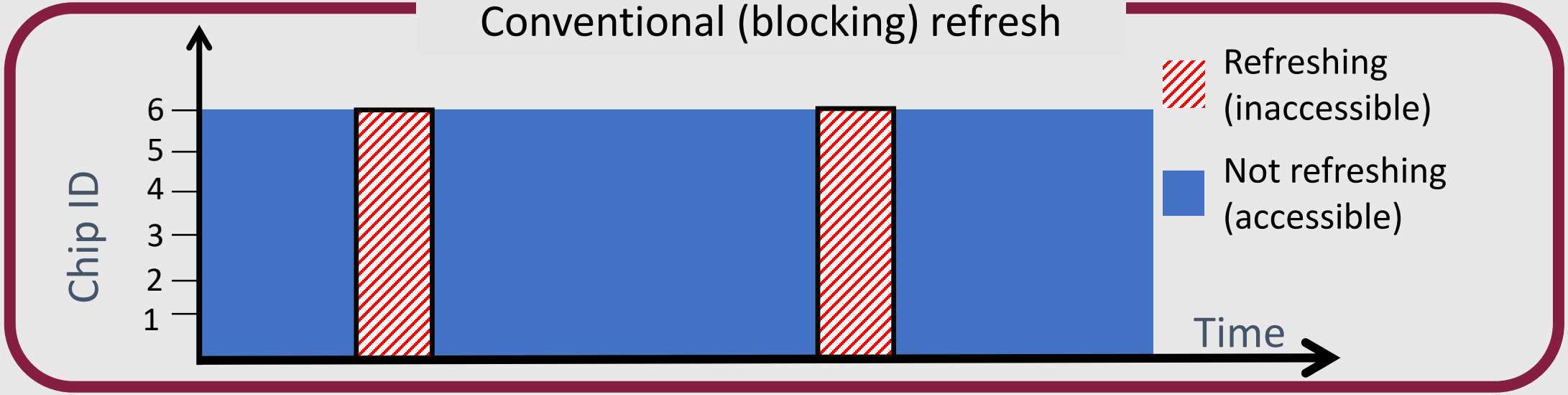
Primer on Server Memory Organization

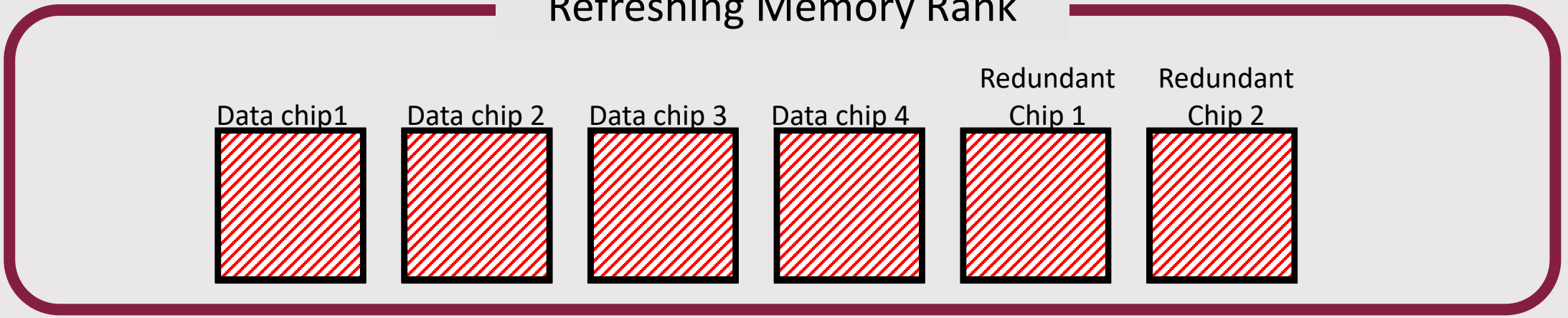


Nonblocking Refresh for Server Memory

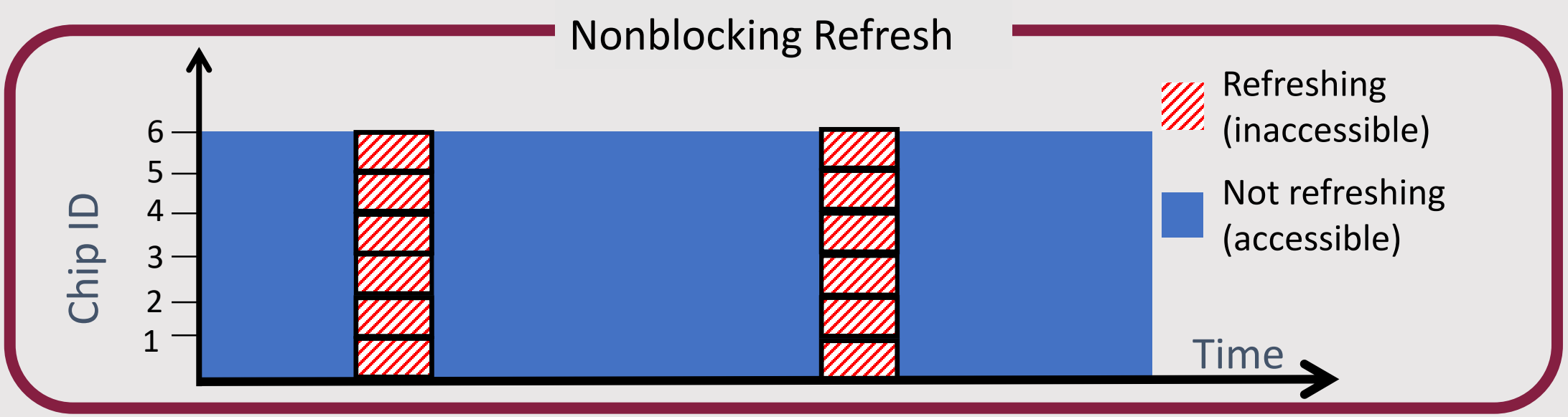


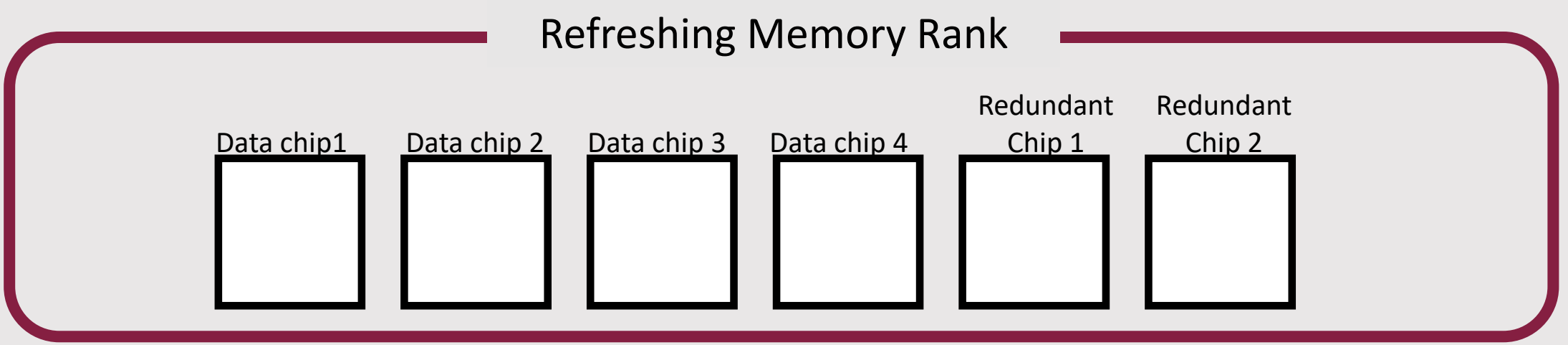
Challenge 1: Ensuring Same Amount of Refresh



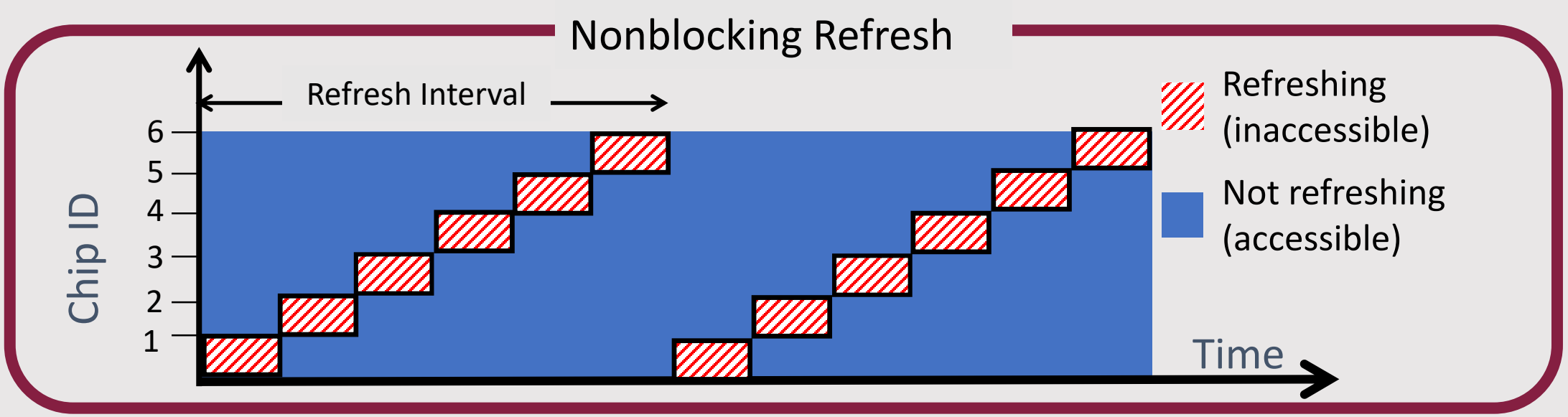


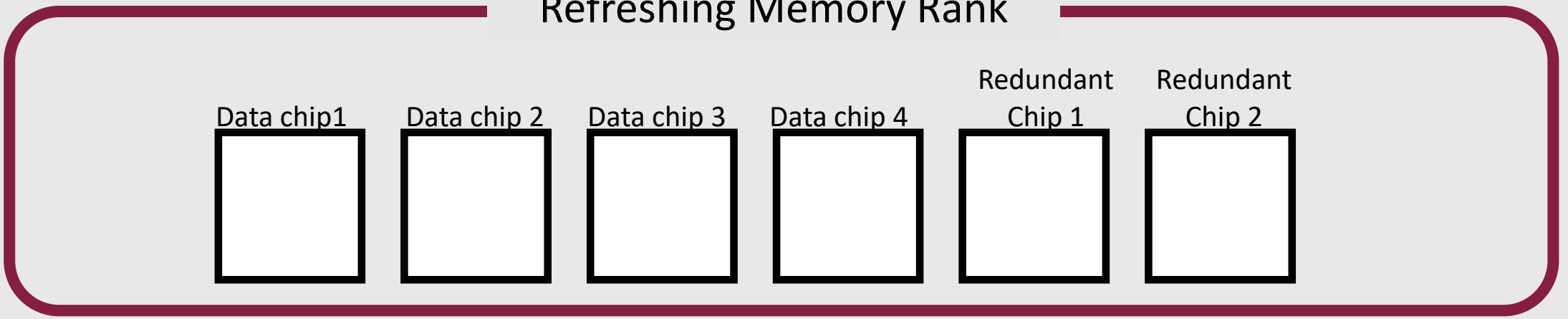
Challenge 1: Ensuring Same Amount of Refresh



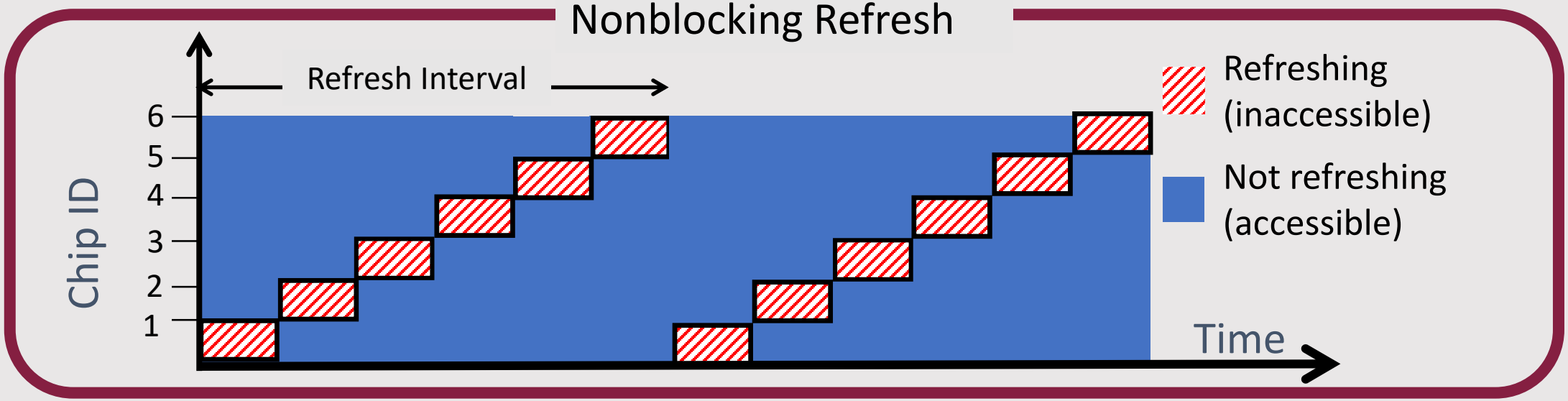


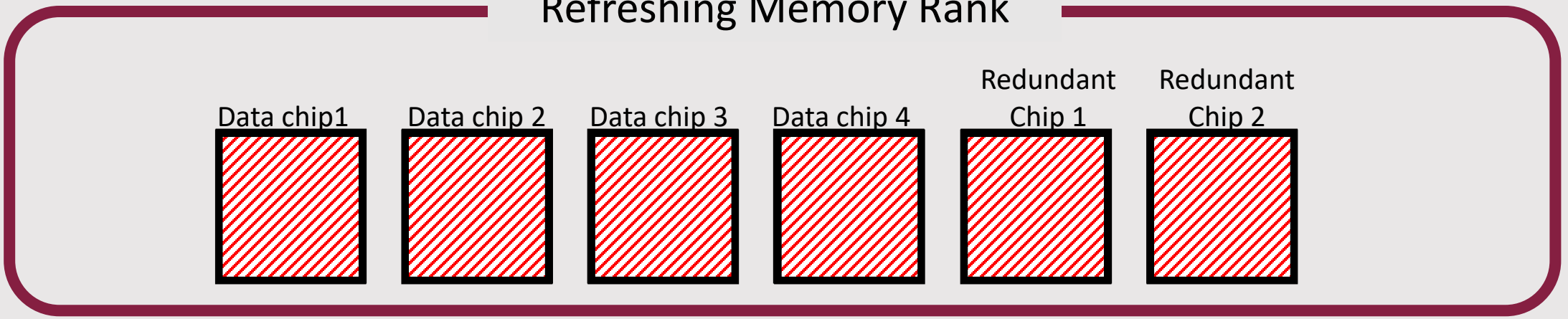
Challenge 1: Ensuring Same Amount of Refresh



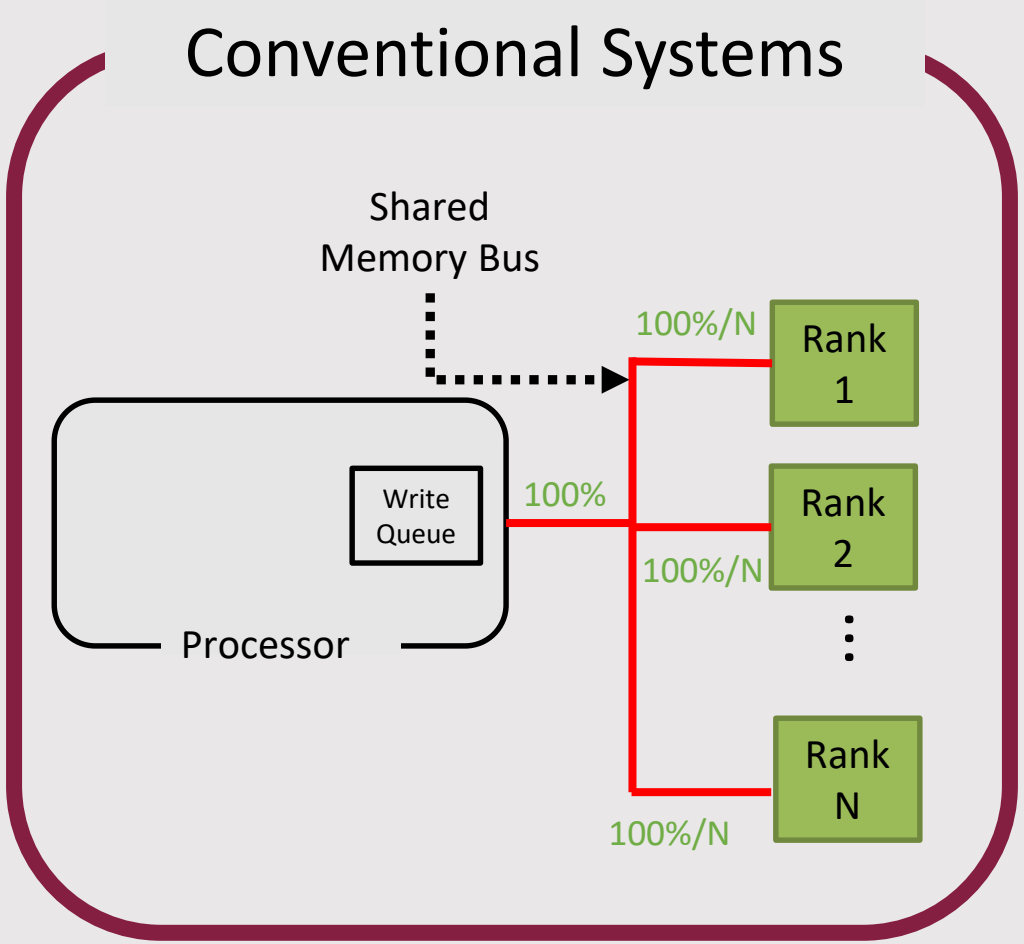


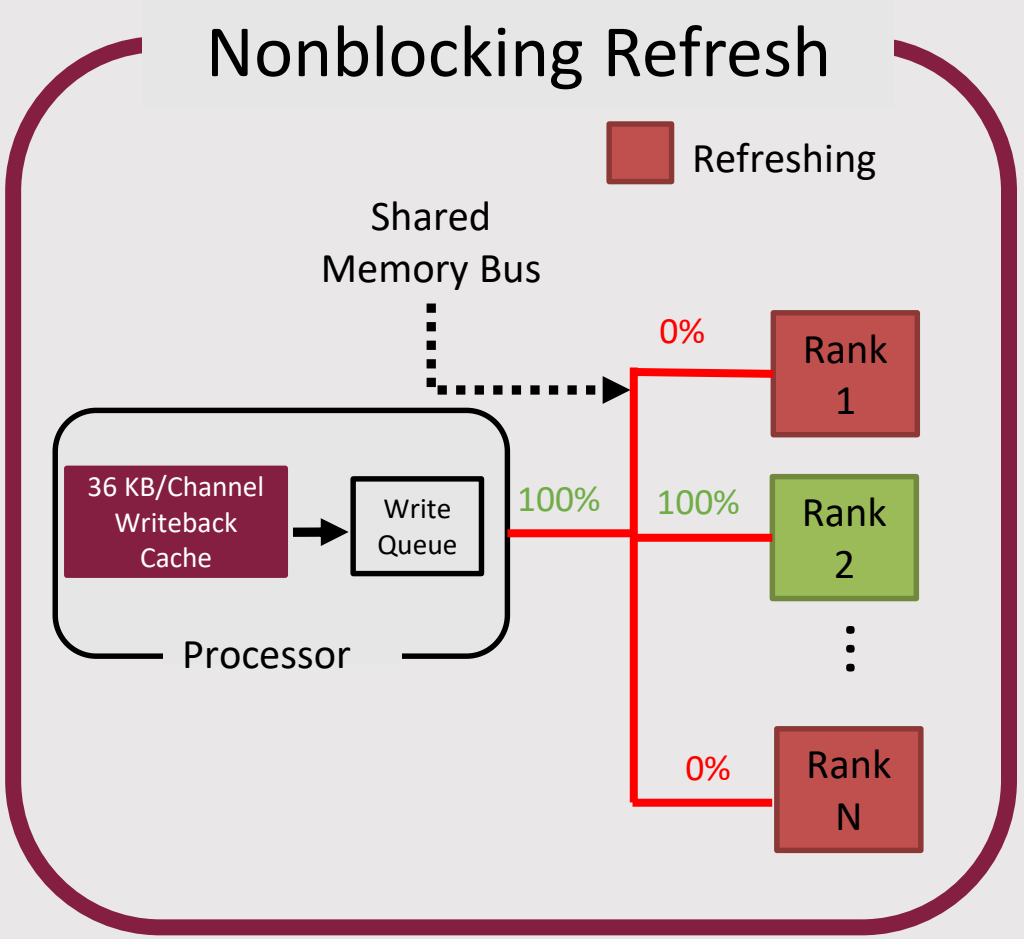
Challenge 1: Ensuring Same Amount of Refresh



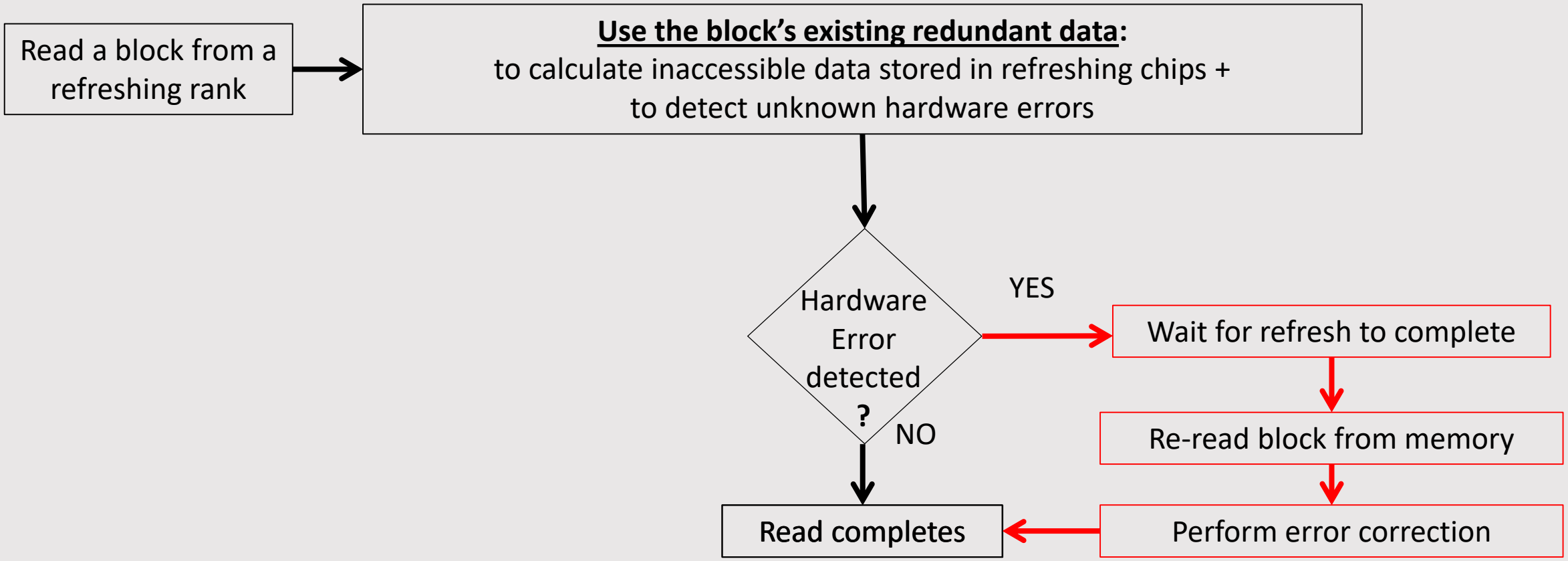


Challenge 2: Ensuring Memory Write Bandwidth





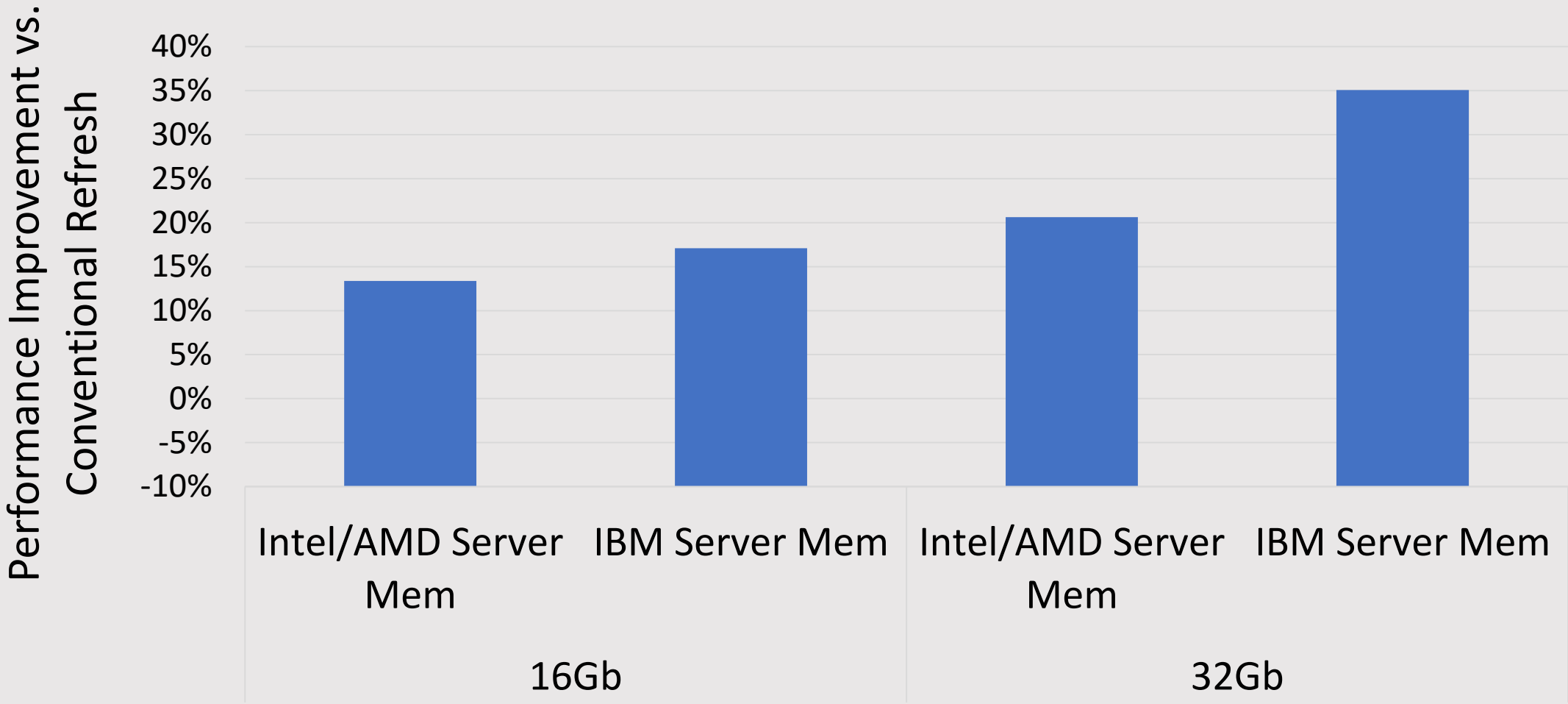
Challenge 3: Preserving Baseline Hardware Failure Protection



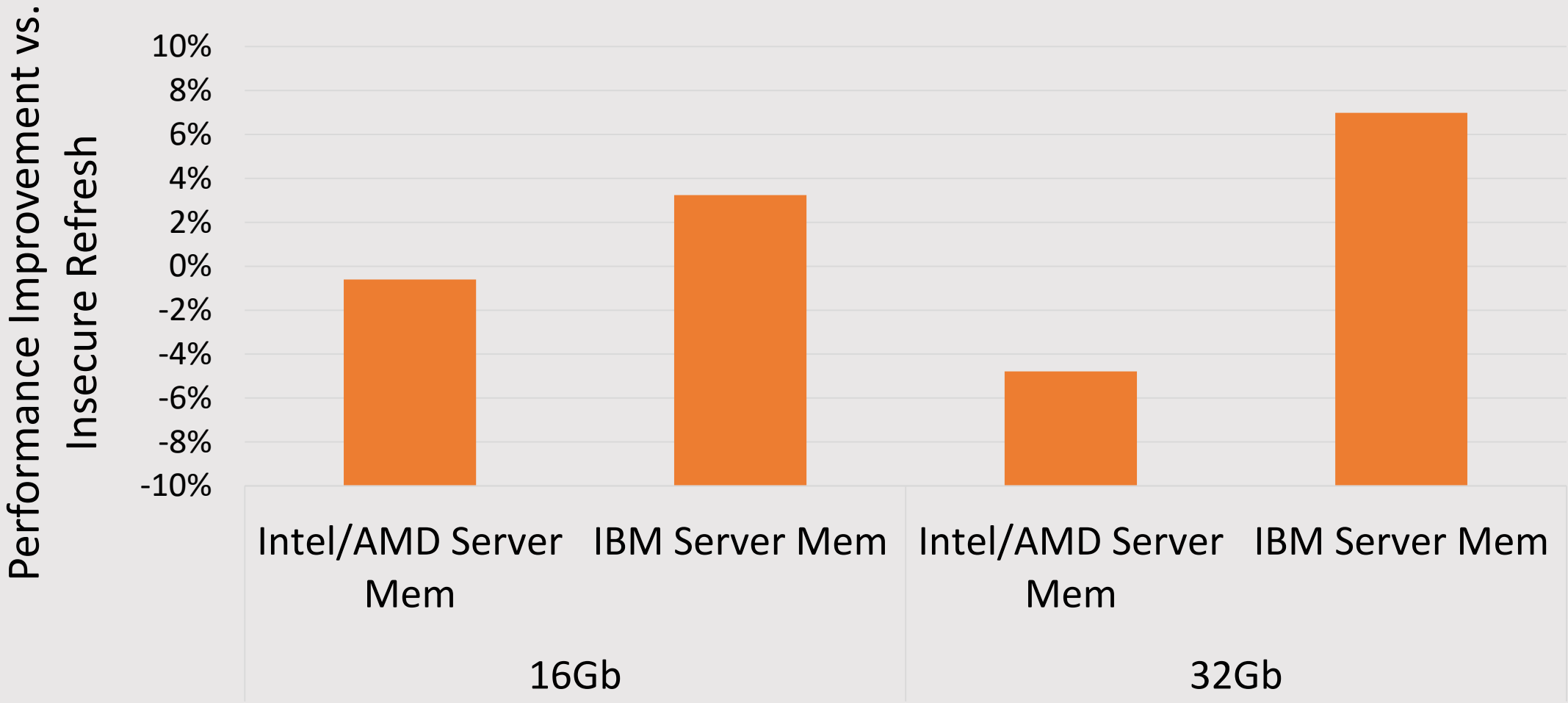
Methodology

- Two Memory Systems:
  - Intel/AMD Server Memory Systems
  - IBM Server Memory System
- Baseline:
  - Conventional Refresh: fully compliance with manufacturer specification
  - Insecure Refresh: skips 75% of refresh operations
- Evaluated 7 multi-threaded and 7 multi-program workloads
- 16gb and future 32gb DRAM
- 4 memory channels with 4 ranks per channel

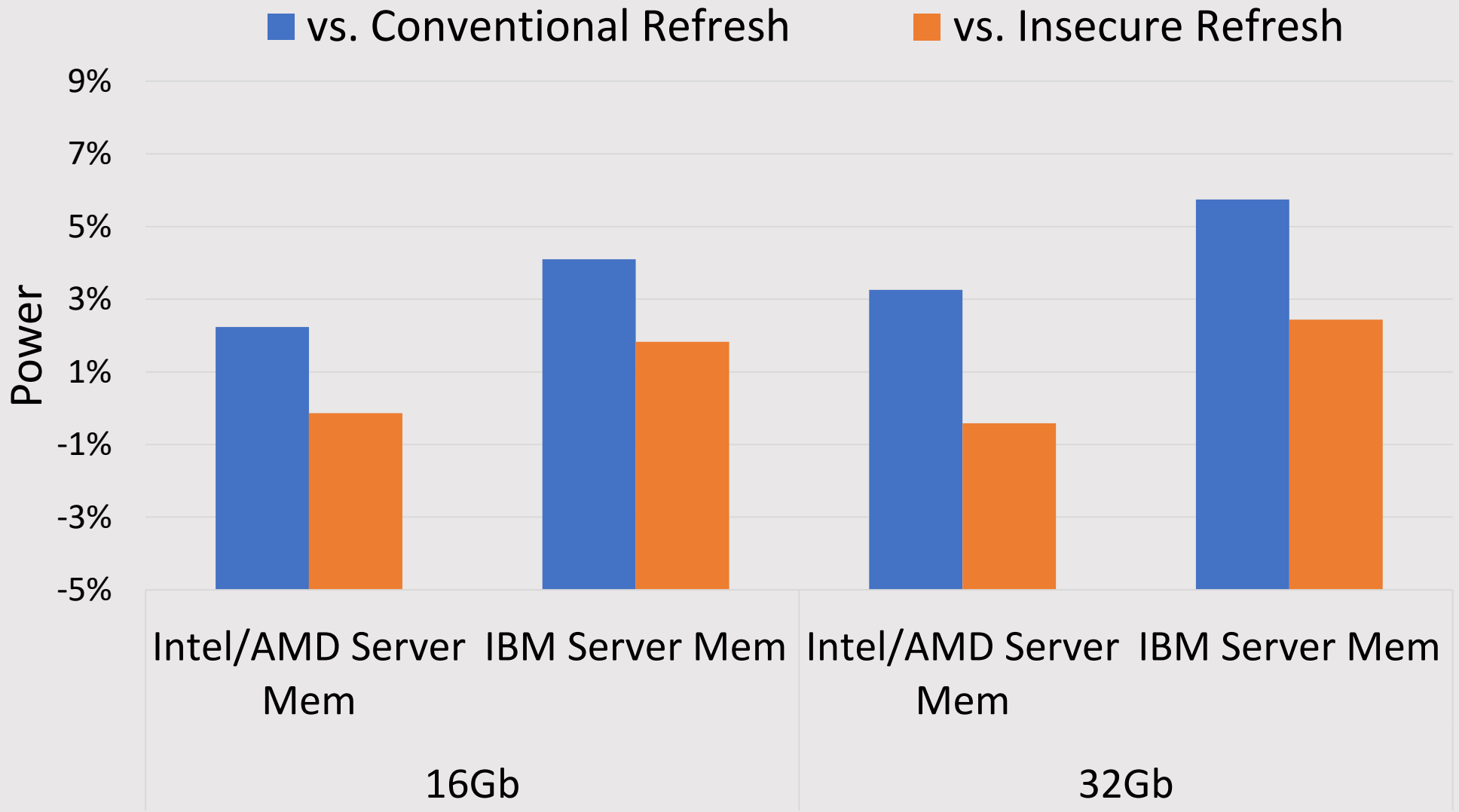
Performance Improvement



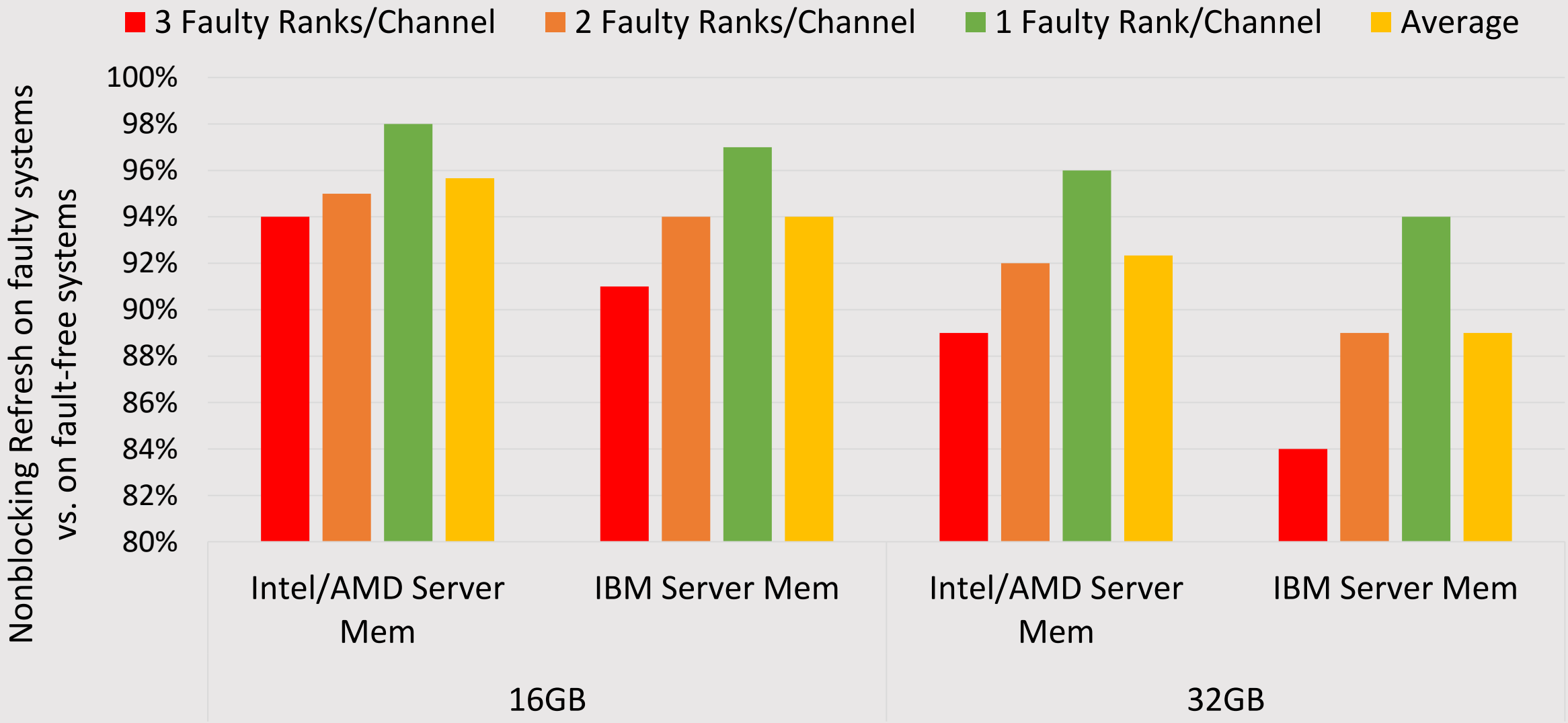
Performance Improvement



Power Consumption



Performance of Systems with Faulty Chips



Conclusion

- Since its invention 50 years ago, DRAM has always required expensive refresh operations that stall accesses to refreshing data.
- We propose Nonblocking Refresh to refresh data in DRAM without stalling read accesses to refreshing data.
- For server memory systems, Nonblocking Refresh improves average performance by 16.2% and 30.3% for 16gb and 32gb chips, respectively.
- Nonblocking Refresh preserves conventional baseline level of security by ensuring the same amount of refresh.

Questions?