RAMinate: Hypervisor-based Virtualization for Hybrid Main Memory Systems

Takahiro Hirofuchi and Ryousei Takano
National Institute of Advanced Industrial Science and Technology

Summary

RAMinate is a hypervisor-based mechanism for hybrid main memory systems composed of DRAM and byte-addressable non-volatile memory (i.e., especially, STT-MRAM in mind).

RAMinate optimizes page locations between DRAM and NVM so as to reduce write traffic to NVM.

To our knowledge, RAMinate is the first work fully implementing a hypervisor-based hybrid memory mechanism.

Introduction

Spin Transfer Torque Magnetoresistive RAM (STT-MRAM, in short, we say MRAM here) will achieve the same level of read/write latency as DRAM around 2016.

MRAM can rewrite memory cells without any practical degradation. MRAM has the potential of serving as the main memory of computer systems like DRAM does today.

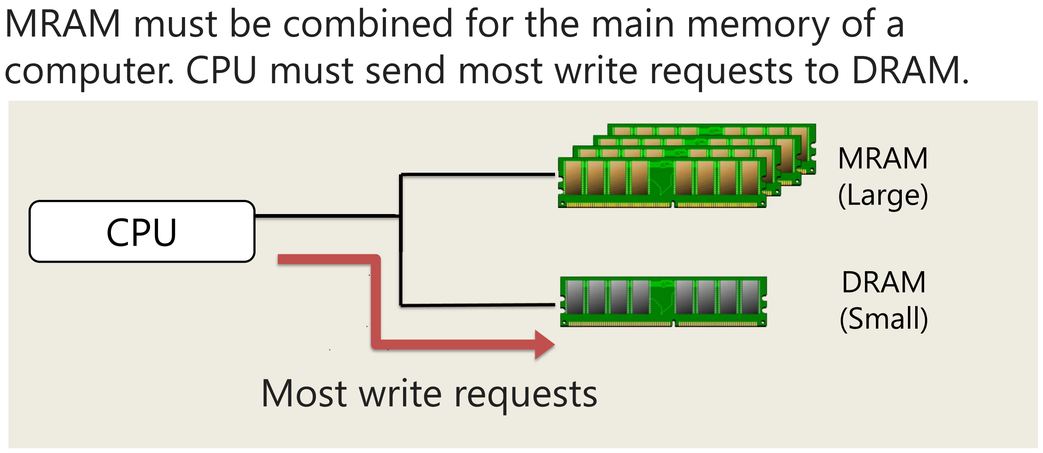
However, the write energy will be 10^2 times larger than that of DRAM.

		2013	2026
Read Time (ns)	DRAM	<10	<10
	STT-MRAM	35	<10
Write/Erasure Time (ns)	DRAM	<10	<10
	STT-MRAM	35	<1
Write Energy (J/bit)	DRAM	4E-15	**2E-15**
	STT-MRAM	2.5E-12	**1.5E-13**

Table 1. Technology roadmap on STT-MRAM and DRAM, according to International Technology Roadmap for Semiconductor 2013

Hybrid Main Memory Systems

To avoid write energy problems of MRAM, both DRAM and MRAM must be combined for the main memory of a computer. CPU must send most write requests to DRAM.



But, past hybrid memory systems do not fit IaaS data centers.

- Past software-based mechanisms need to modify operating systems or application programs.
- Hardware-based mechanisms need to modify the memory subsystem of computers.
- Not transparent to exiting software and hardware components, which will incur huge migration costs to take advantage of emerging NVM technologies.

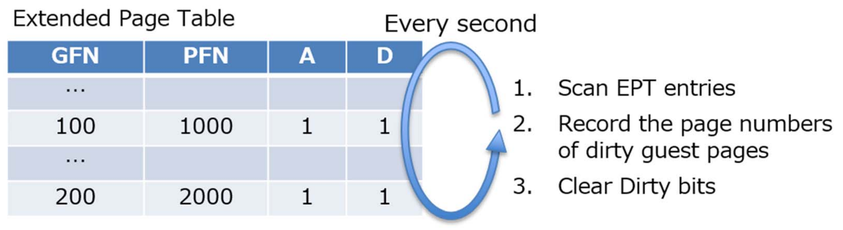
Hybrid Memory Support at Hypervisor

The hypervisor layer, located between guest-OS software and physical hardware, is the most suitable for hybrid memory support. It can provide transparency to guest operating systems without any modification to existing memory controllers.

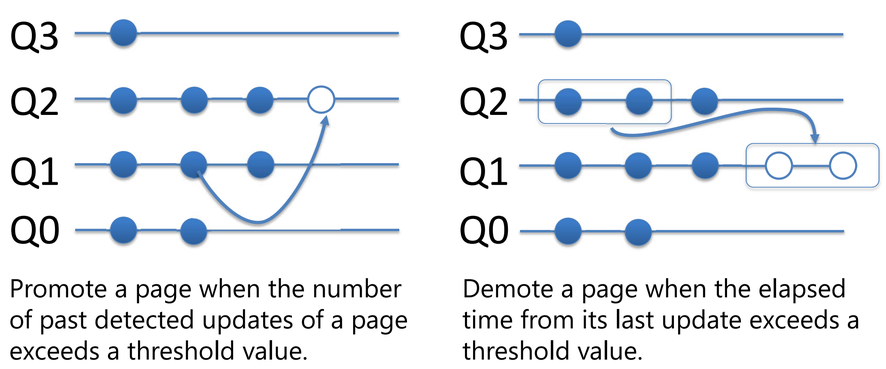
RAMinate is composed of 3 major components:

1. Lightweight trace mechanism of guest memory access
2. Page placement algorithm
3. Page migration mechanism

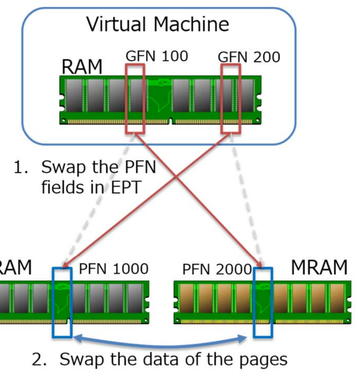
Component 1. RAMinate periodically scans Extended Page Table entries of a VM and finds dirty guest pages.



Component 2. The optimization algorithm, Corked Multi Queue, determines which guest pages must be migrated. The queue level relates to the number of past page updates. MRAM pages in higher levels of queues are candidates for page migration.



Component 3. RAMinate dynamically updates page mapping between guest and physical pages without disrupting the guest operating system.



Swaps a write-intensive MRAM page with a non-write-intensive DRAM page.

Updates the EPT entries of these pages and performs memory copies.

Evaluation

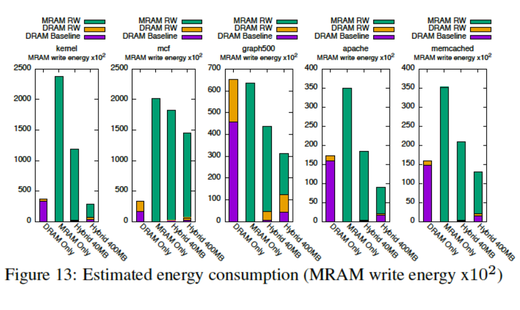


Figure 13: Estimated energy consumption (MRAM write energy x10^2)

Hybrid memory with 10% DRAM reduced energy by 50% for Graph 500 in comparison to DRAM-only memory.

See our SoCC paper for details!