RAMinate: Hypervisor-based Virtualization for Hybrid Main Memory Systems

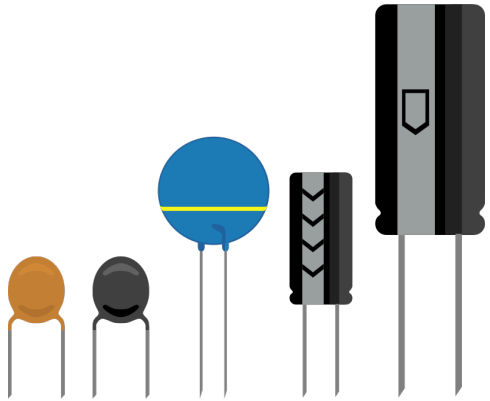
Takahiro Hirofuchi and Ryousei Takano

National Institute of Advanced Industrial Science and Technology (AIST)

ACM Symposium on Cloud Computing 2016, Oct. 2016

Emerging Non-volatile Memory (NVM)

A DRAM cell =~ a capacitor



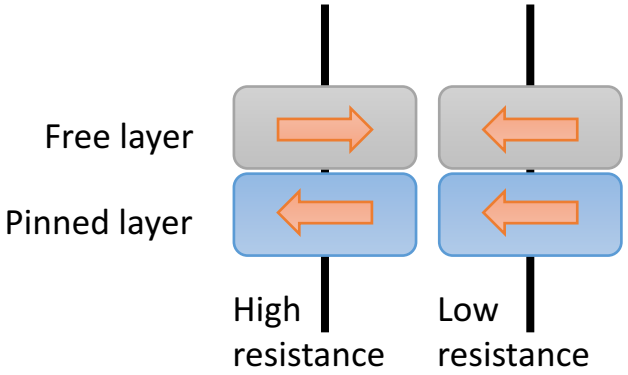
Energy consuming

- Need refresh energy
- More than 30% of energy consumption of a data center

Not scalable anymore

- Due to serious energy dissipation

A MRAM cell =~ ferromagnetism



Energy efficient

- No refresh energy

Scalable in theory

- Higher density will be possible

Technology Roadmap on STT-MRAM

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

		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**

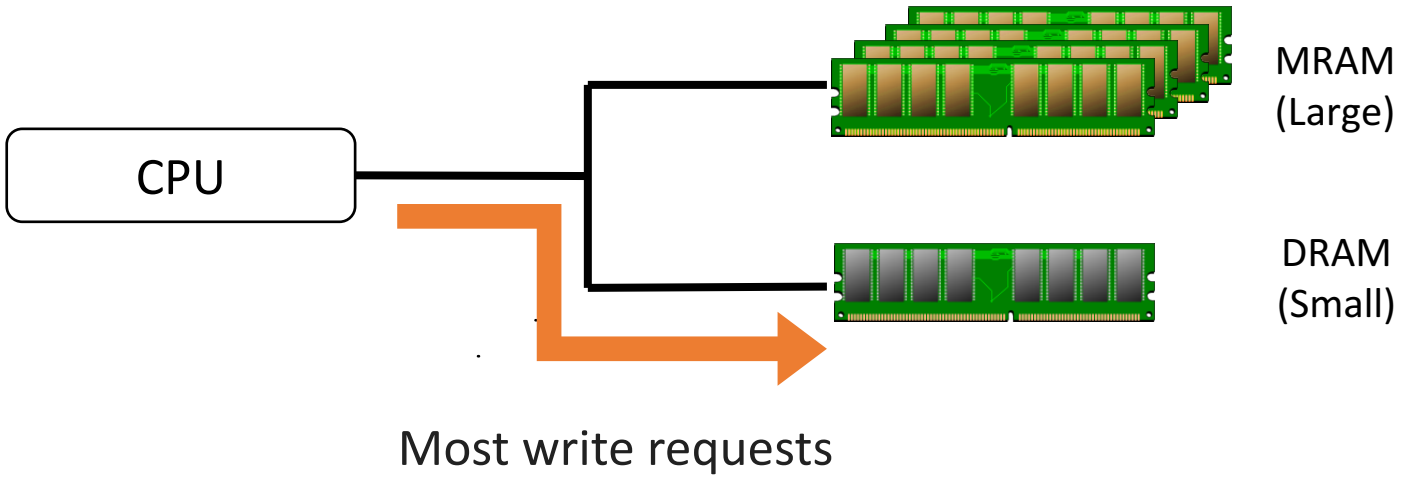
Spin Transfer Torque Magnetoresistive RAM (STT-MRAM, in short, we say MRAM here) will achieve the same level of read/write latency as DRAM around 2016.

MRAM can rewrite memory cells without any practical degradation.

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

Hybrid Main Memory

- Achieve large main memory capacity with small energy consumption
- 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.

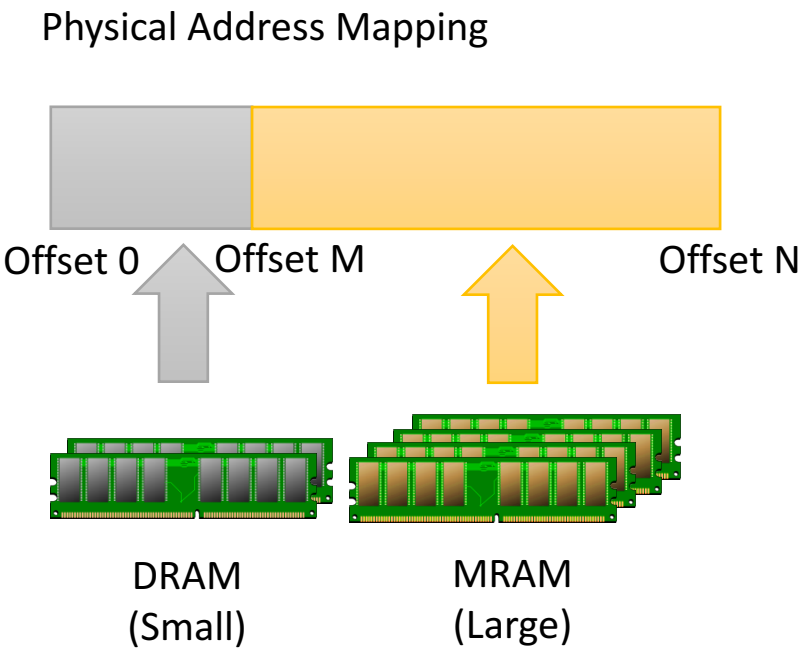


Requirements for Hybrid Memory Systems in IaaS Datacenters

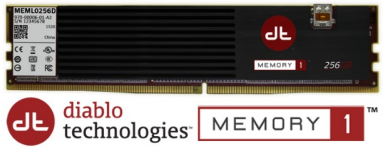
- Transparent to guest operating systems
  - A guest OS is under customer's administrative domain, not service provider's administrative domain.
- Easy to apply the current server systems
  - Extending a memory controller is costly, and should be avoided.

Assumption

Both DRAM and STT-MRAM are byte-addressable.
Both are mapped to different physical address ranges.

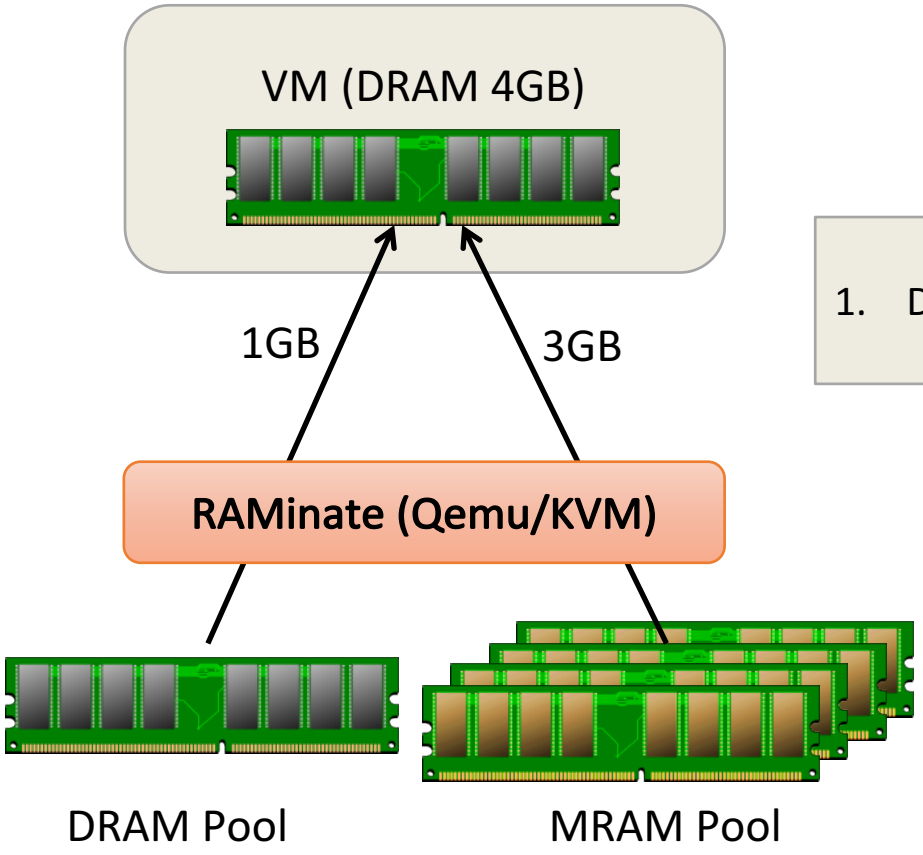


- Future NVM technologies will be also attached to a DIMM interface
- Note: Currently, DIMM-based Flash modules are already on the market.



RAMinate: Hybrid Memory Support at Hypervisor

Provide transparency to guest operating systems without any modification to existing memory controllers.



1. Detect written pages
2. Determine necessary page swap operations
3. Update page mapping

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

EPT maintains mapping between guest frame numbers and physical frame numbers. When a guest page is written, CPU sets the dirty bit.

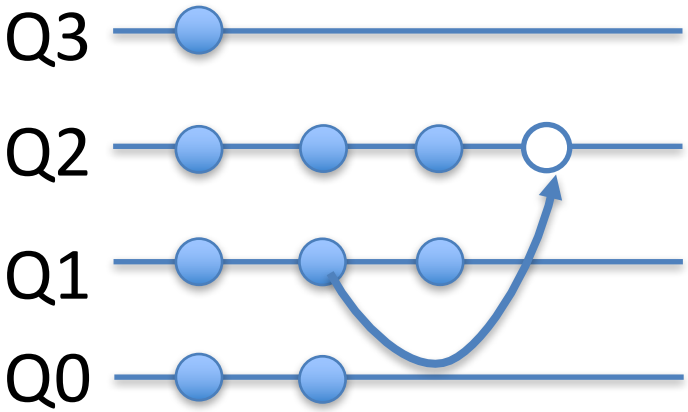
Extended Page Table

GFN	PFN	A	D
…			
100	1000	1	1
…			
200	2000	1	1

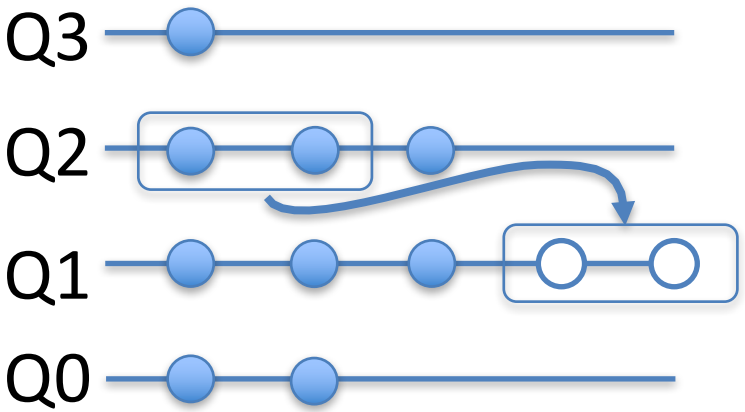
1. Scan EPT entries
2. Record the page numbers of dirty guest pages
3. Clear Dirty bits

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.



Promote a page when the number of past detected updates of a page exceeds a threshold value.

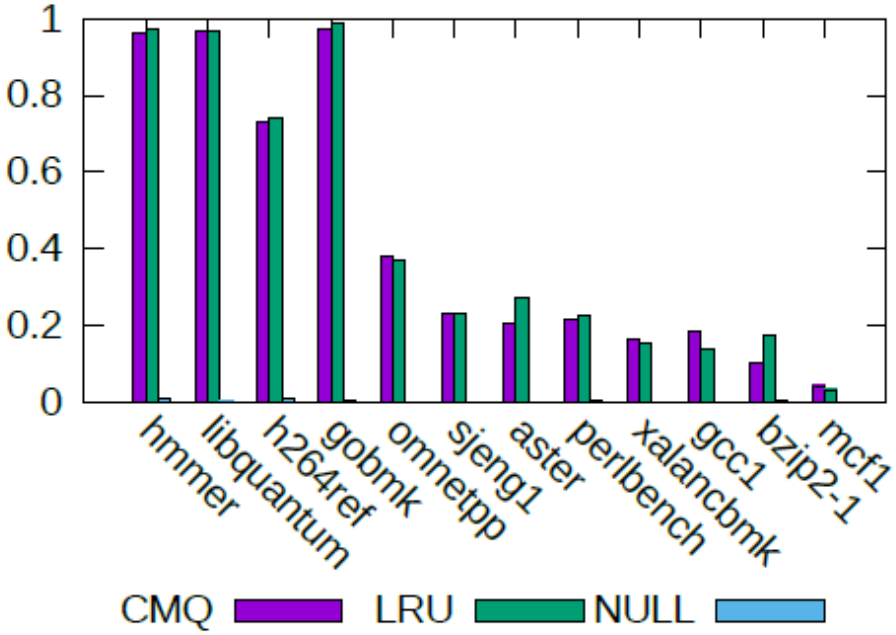


Demote a page when the elapsed time from its last update exceeds a threshold value.

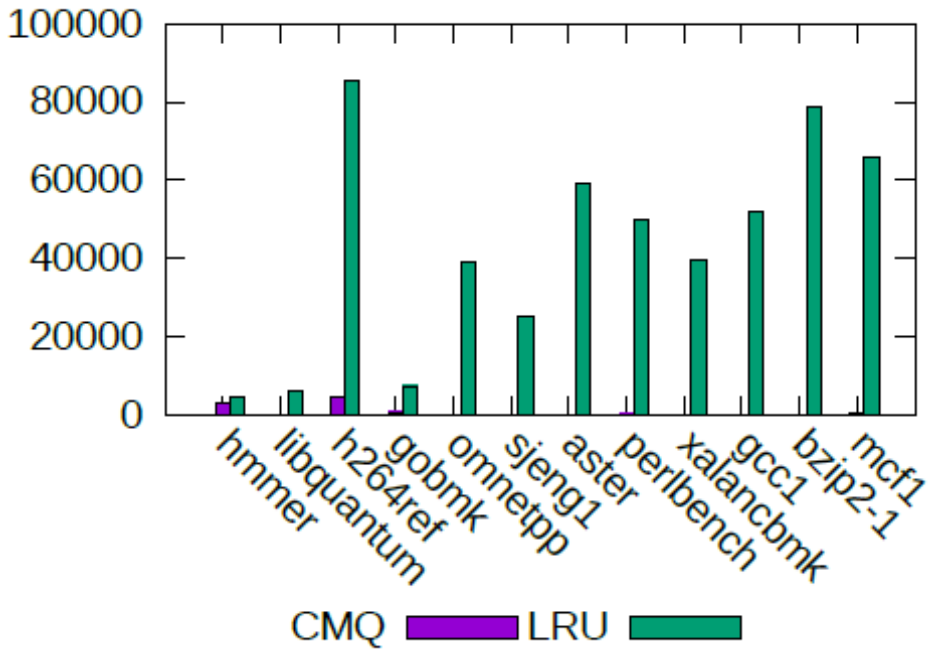
Simulation

- Compare CMQ, LRU and nothing
- Use trace data during SPEC CPU 2006

DRAM hit ratio

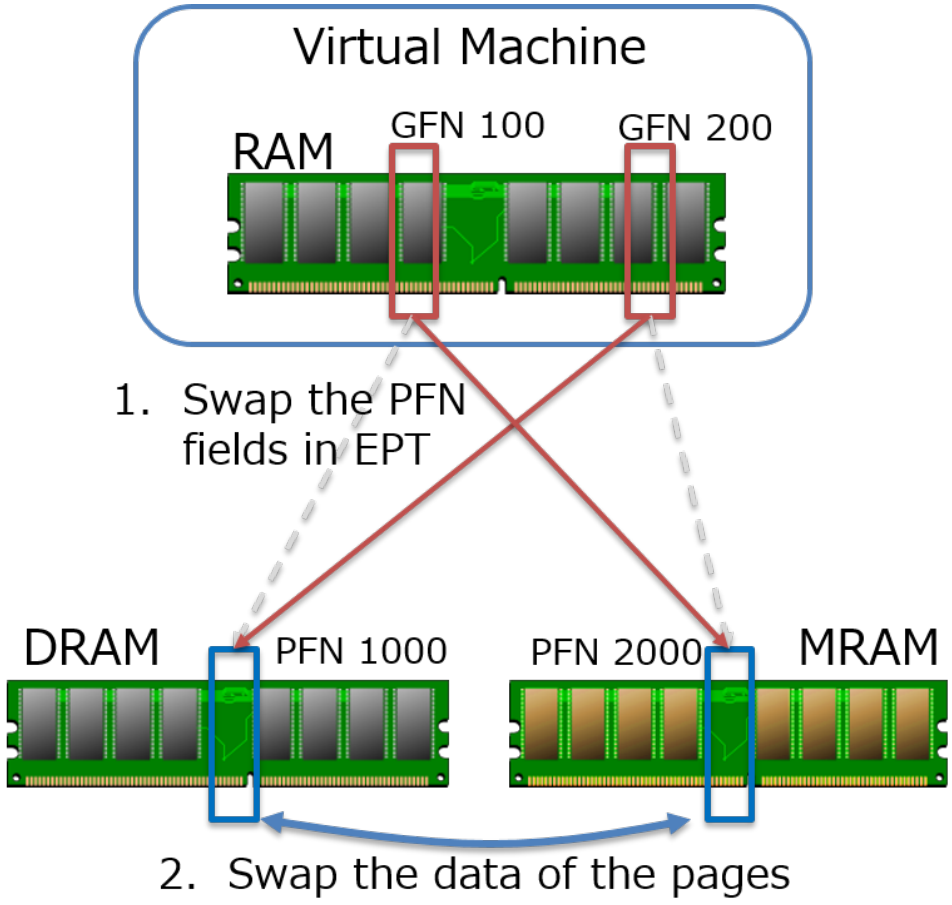


The number of swapped pages



CMQ drastically reduces the number of necessary page swaps.

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



Page Swap in Detail

- Temporary stop the VM during optimization
  1. Stop the VM
  2. Repeat page swap operations
     1. Swap the PFNs of 2 EPT entries
     2. Swap the data of the memory pages
  3. Restart the VM

Page Swap Overhead

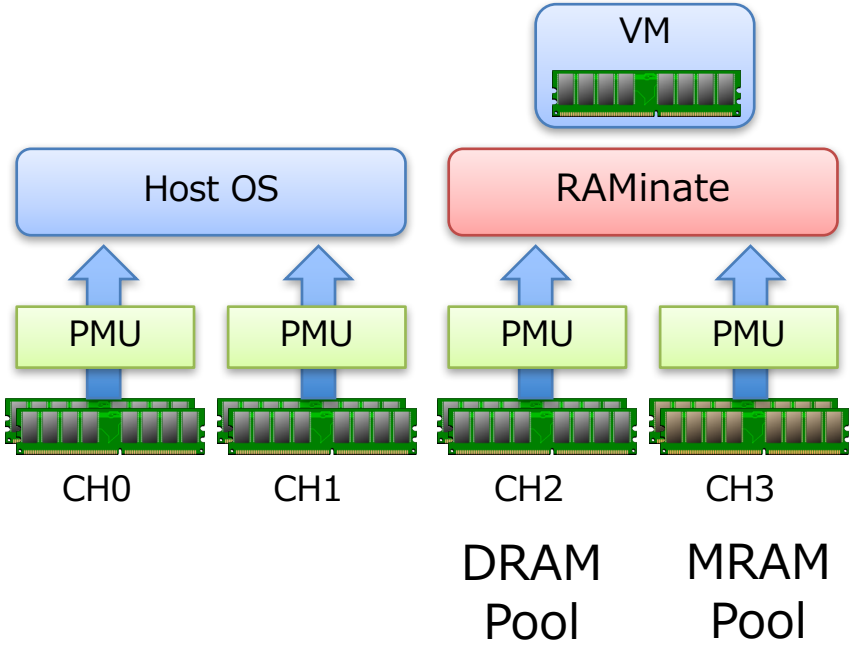
- Downtime is approximately 30ms for 1000 swap operations.
- Short enough, acceptable
  - Page mapping optimization is performed every 5 second in the default setting.

Table 1: Elapsed time for simultaneous page swaps

# of Page swaps	Elapsed time in ms (mean ± SD)
0	28.3 ± 10.1
100	27.9 ± 11.9
1000	29.8 ± 10.0
10000	44.3 ± 14.5

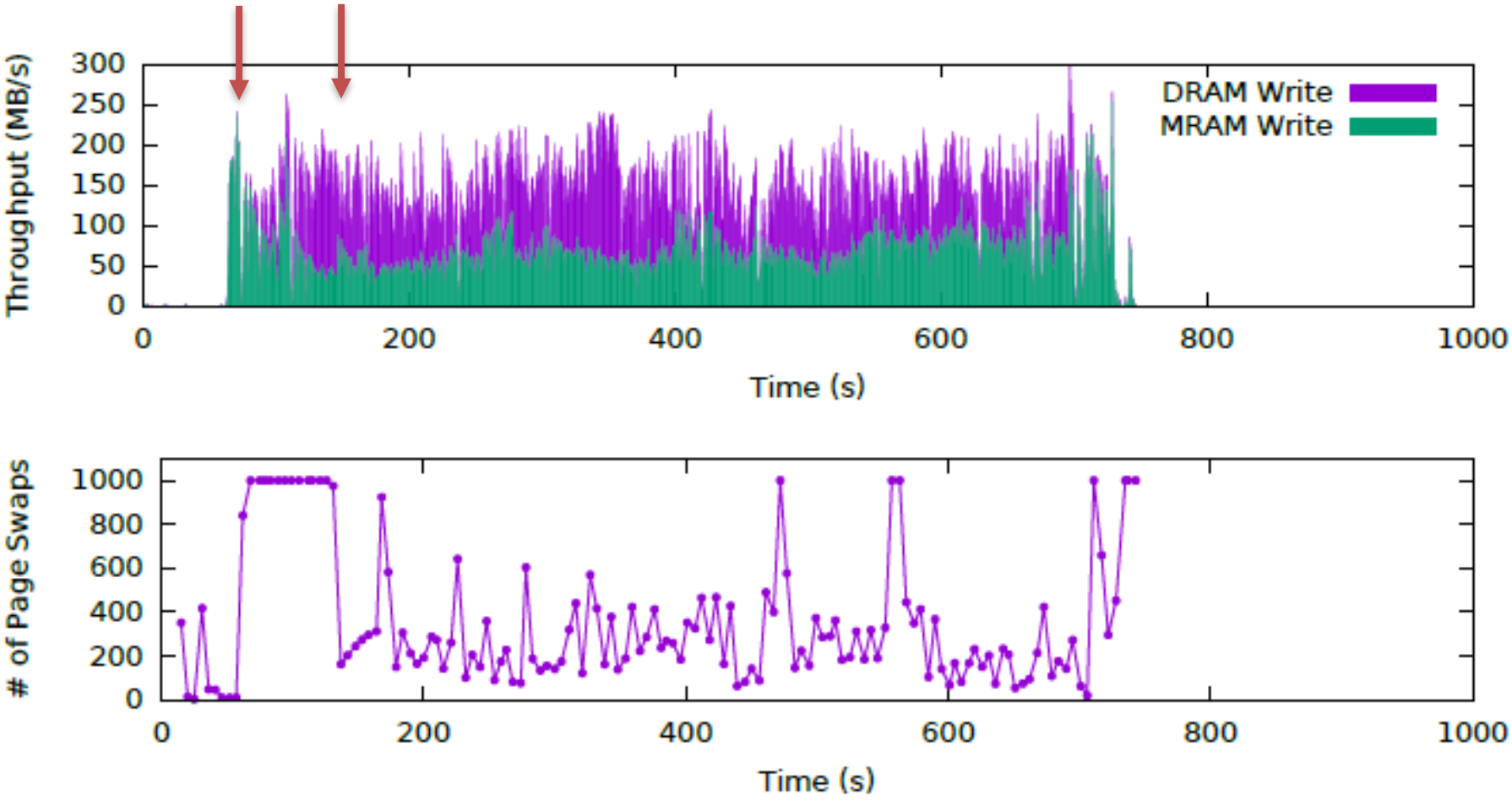
Experiments

- Use one memory channel for MRAM
- Disable memory channel interleaving
- Measure read/write traffic per memory channel by PMU
  - CAS_COUNT.RD Counter
  - CAS_COUNT.WR Counter

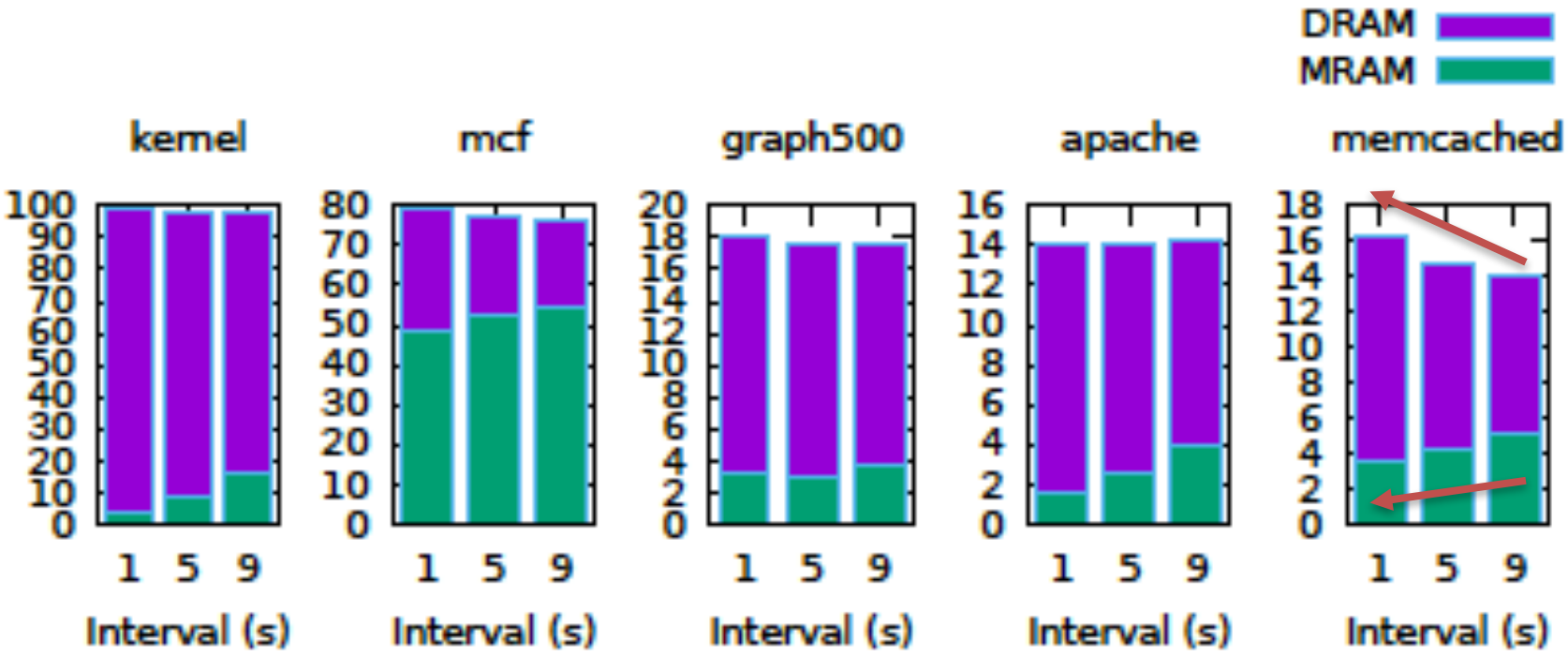


Kernel Compile (DRAM 40MB, MRAM 3960MB)

First, most write traffic went to MRAM. After optimized, only 50%.



The Total Amount of Written Data to DRAM and MRAM in GB



The DRAM size is 10% of VM memory (400MB).

A smaller interval of optimization reduced the ratio of MRAM traffic, but increased the total of written data because of memory copy operations for page swaps.
(Also, there is more performance overhead in a smaller interval. See the paper.)

Energy Estimation

- Explore how much energy can be saved (or wasted) if RAMinate is applied to future STT-MRAM devices
- First, develop energy models of DRAM and STT-MRAM
- Next, estimate energy consumption by applying obtained the data through the experiments to the models

Energy Models

- Assume a simple linear correlation between read/write throughput and energy consumption
- Sufficient for discussing orders of magnitudes larger write energy

- DRAM
  - X: Read data size (MB)
  - Y: Written data size (MB)
  - P: Idle Power (W) per VM
  - T: Duration of an experiment (s)
  - Energy (J) = a*X + a*Y + P*T
- STT-MRAM with 100x write energy
  - X: Read data size (MB)
  - Y: Written data size (MB)
  - Energy (J) = a*X + (100*a)*Y

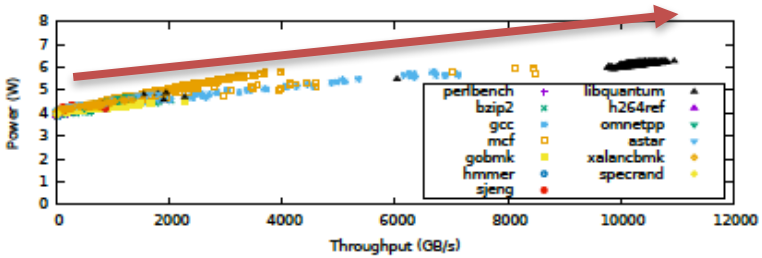
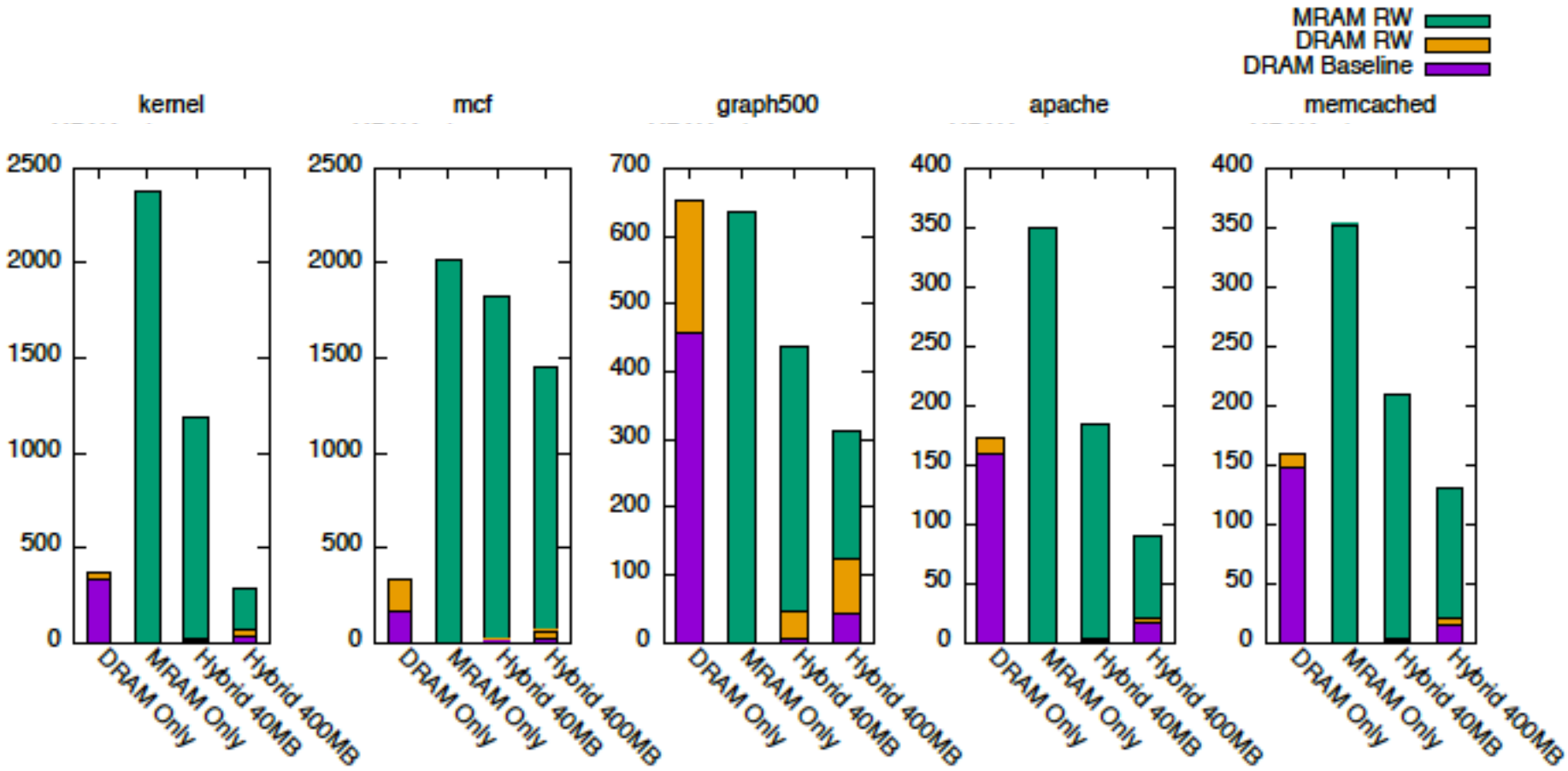


Figure 11: The relationship between memory I/O throughput and power consumption during the SPEC CPU benchmarks. Values were obtained via the performance monitoring unit of the memory controller of the used PM.

Estimated Energy Consumption



- Assume MRAM write energy x10^2 than that of DRAM.
  Other values are discussed in the paper.
- Hybrid memory with 10% DRAM outperformed DRAM-only memory in most cases.

Conclusion

- RAMinate: Hypervisor-based hybrid memory mechanism
  - Transparent to existing applications
  - No modification to existing memory controllers
  - AFAIK, the first study at the hypervisor layer

- Reduce energy consumption of a server with DRAM and future STT-MRAM
  - For tested workloads,
    - 70% reduction in write traffic to STT-MRAM
    - 50% reduction in total energy consumption

- **Poster Today!**