Gyu Sang Choi
Yeungnam University

PRAM and NAND Flash Memory, and B+Tree in PRAM

Outline

- NAND Flash Memory
- PRAM
- Hybrid Storage of PRAM and NAND Flash Memory
- PRAM Translation Layer (PTL)
- B+Tree in PRAM
- Conclusions and Future Work

NAND Flash Memory

- Erase-before-write
- Block and Page
- Out-of-place update
- Asymmetry between read and write latency
- Up to 10^5 erase numbers
- The page size increases as time goes

NAND Flash Type	SLC	MLC
4KB Page Read Latency	~25 us	~60 us
4KB Page Write Latency	~200 us	~800 us
Block Erase Latency	~1.5 ms	~1.5 ms

* K9XXG08XXA **and** K9F8G08UXM Data Sheet

PRAM*

- In-place update
- Byte-addressable
- 10^6 write numbers
- No erase operation
- Asymmetry between read and write latency
- Capacity increases

PRAM	Latency
64Bytes Read Latency	~50ns
64Bytes Write Latency	~1us

- Shimin chen, Phillip B. Gibbons and Suman Nath, Rethinking Database Algorithms for Phase Change Memory, CIDR 2011
- PRAM, PCM, PCRAM are interchangeably used, and PRAM will be only used in this presentation

Hybrid Storage of PRAM and NAND Flash Memory

Prior works

- (PRAM and Flash File System) PFFS* had been proposed (PFFS) to store meta-data into PRAM
  - The meta-data are more frequently updated compared to user data
  - Use a complex wear-leveling mechanism by cold and hot area swapping
- Kim et al# proposed hybrid storage of PRAM and NAND Flash memory which stores meta-data of both file system and NAND Flash memory into PRAM
  - Reduce the write numbers using word-level comparison
  - No result on PRAM's life span

Motivation

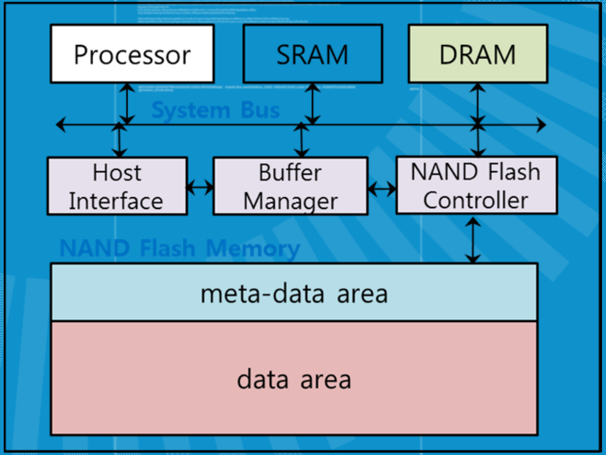
- Solve PRAM's endurance perfectly

*Park et al., Pffs: a scalable flash memory file system for the hybrid architecture of phase-change ram and nand flash. In SAC '08, pages 1498–1503, 2008

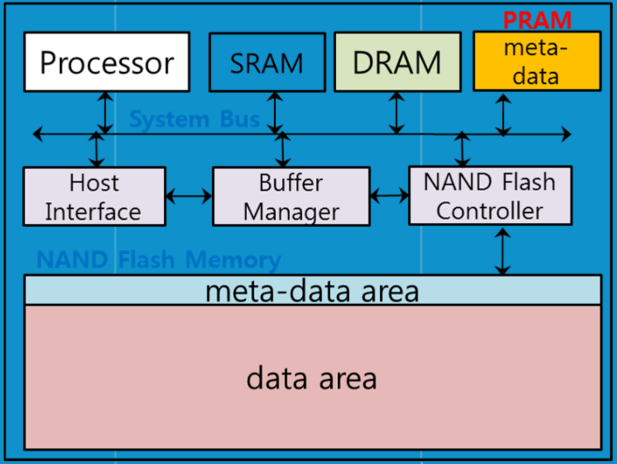
#Kim et al., A PRAM and NAND Flash Hybrid Architecture for High-Performance Embedded Storage Subsystems, in EMSOFT '08, 2008, pp. 31–40.

Hybrid Storage of PRAM and NAND Flash Memory

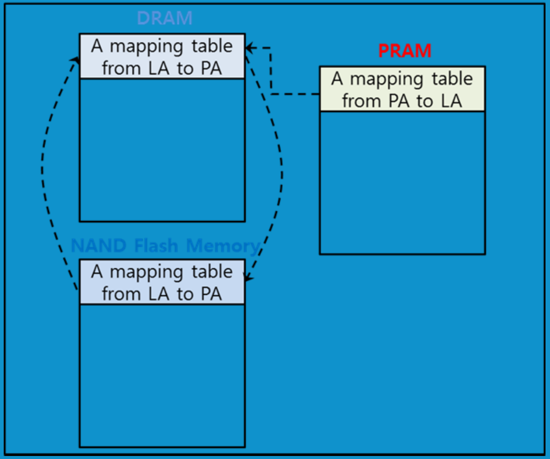
The Proposed Scheme



Internal Architecture of Flash-based SSD



Internal Architecture of the Proposed Hybrid SSD

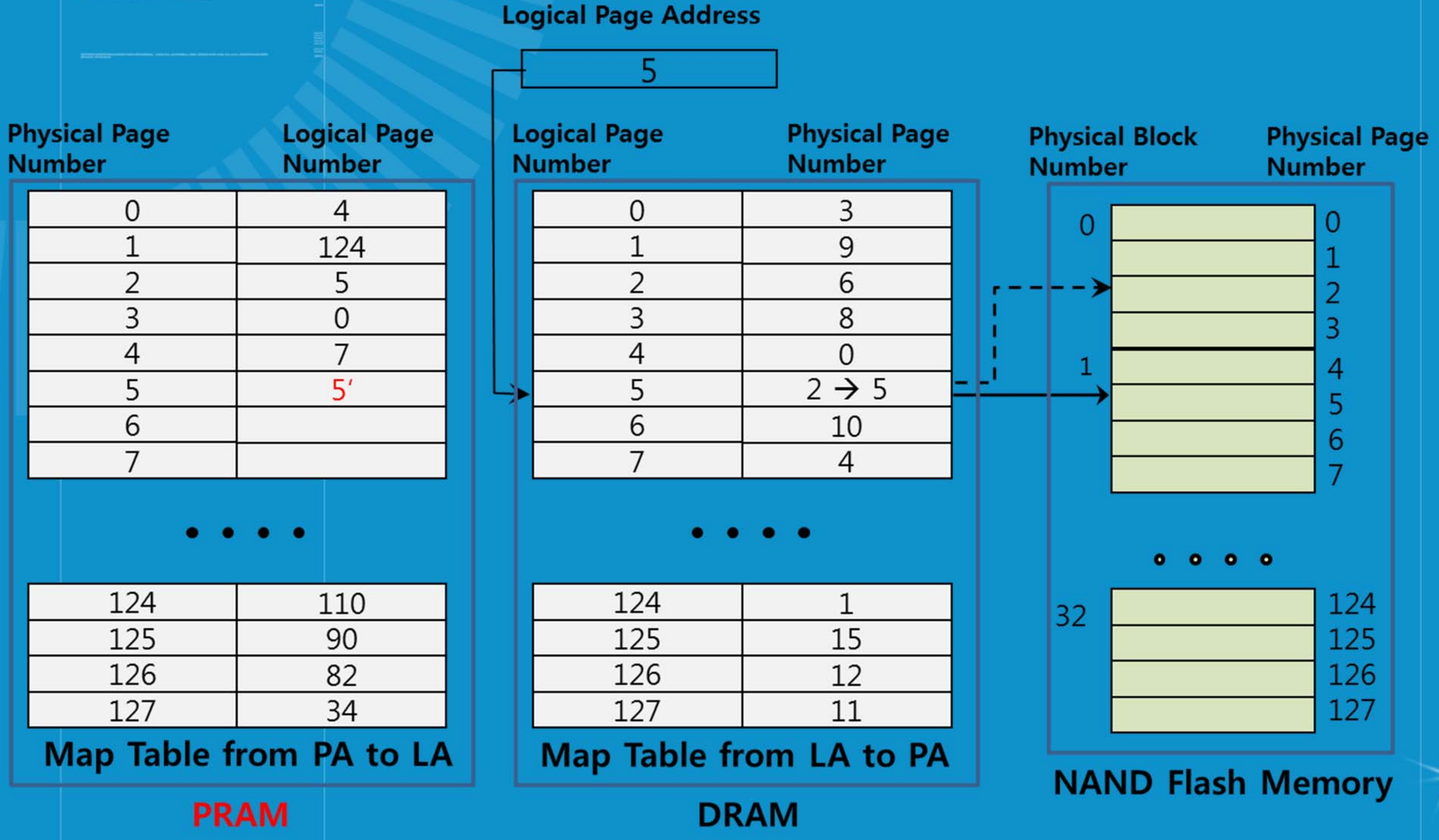


DRAM
A mapping table from LA to PA
PRAM
A mapping table from PA to LA
NAND Flash Memory
A mapping table from LA to PA

Meta-date Management in the Proposed Scheme

Hybrid Storage of PRAM and NAND Flash Memory

The Proposed Scheme



Physical Page Number	Logical Page Number
0	4
1	124
2	5
3	0
4	7
5	5′
6	
7	
• • • •	
124	110
125	90
126	82
127	34

Map Table from PA to LA

PRAM

Logical Page Number	Physical Page Number
0	3
1	9
2	6
3	8
4	0
5	2 → 5
6	10
7	4
• • • •	
124	1
125	15
126	12
127	11

Map Table from LA to PA

DRAM

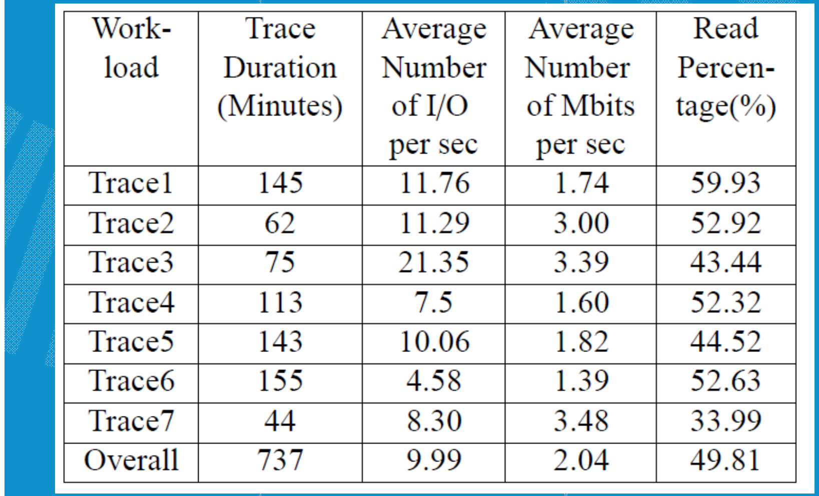
Physical Block Number	Physical Page Number
0	0
	1
	2
	3
1	4
	5
	6
	7
• • • •	
32	124
	125
	126
	127

NAND Flash Memory

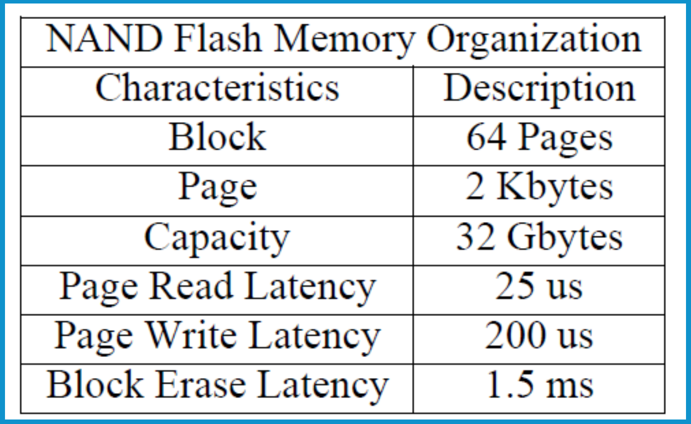
Hybrid Storage of PRAM and NAND Flash Memory

Experimental Setup

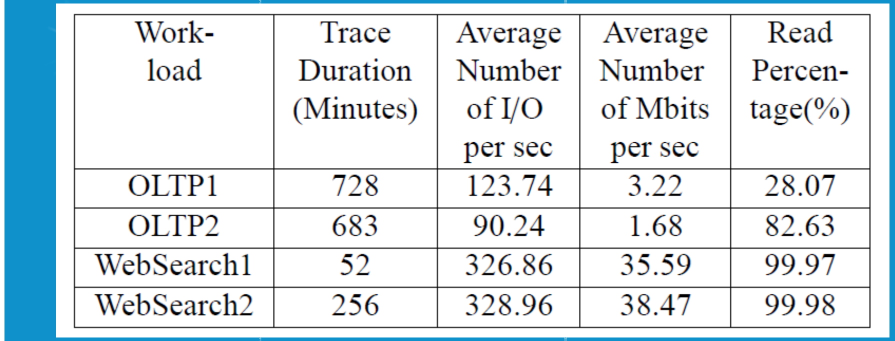
- Modify FlashSim* to support PRAM



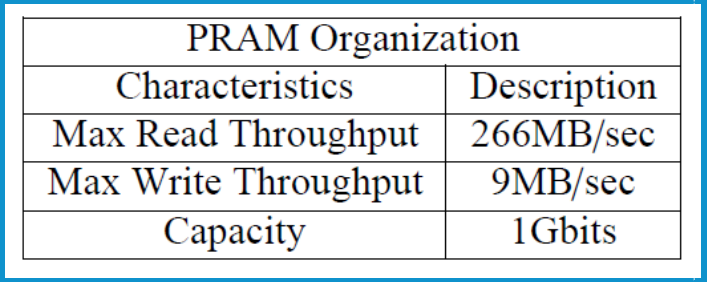
Work-load	Trace Duration (Minutes)	Average Number of I/O per sec	Average Number of Mbits per sec	Read Percen-tage(%)
Trace1	145	11.76	1.74	59.93
Trace2	62	11.29	3.00	52.92
Trace3	75	21.35	3.39	43.44
Trace4	113	7.5	1.60	52.32
Trace5	143	10.06	1.82	44.52
Trace6	155	4.58	1.39	52.63
Trace7	44	8.30	3.48	33.99
Overall	737	9.99	2.04	49.81



NAND Flash Memory Organization	
Characteristics	Description
Block	64 Pages
Page	2 Kbytes
Capacity	32 Gbytes
Page Read Latency	25 us
Page Write Latency	200 us
Block Erase Latency	1.5 ms



Work-load	Trace Duration (Minutes)	Average Number of I/O per sec	Average Number of Mbits per sec	Read Percen-tage(%)
OLTP1	728	123.74	3.22	28.07
OLTP2	683	90.24	1.68	82.63
WebSearch1	52	326.86	35.59	99.97
WebSearch2	256	328.96	38.47	99.98

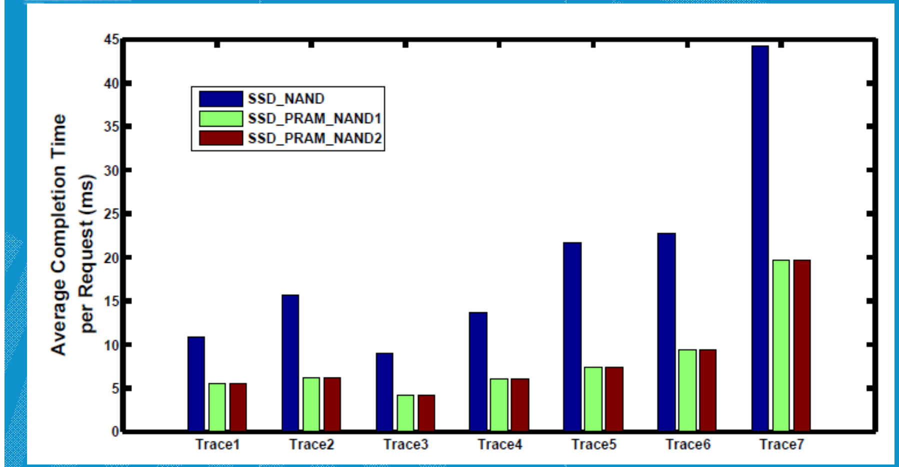


PRAM Organization	
Characteristics	Description
Max Read Throughput	266MB/sec
Max Write Throughput	9MB/sec
Capacity	1Gbits

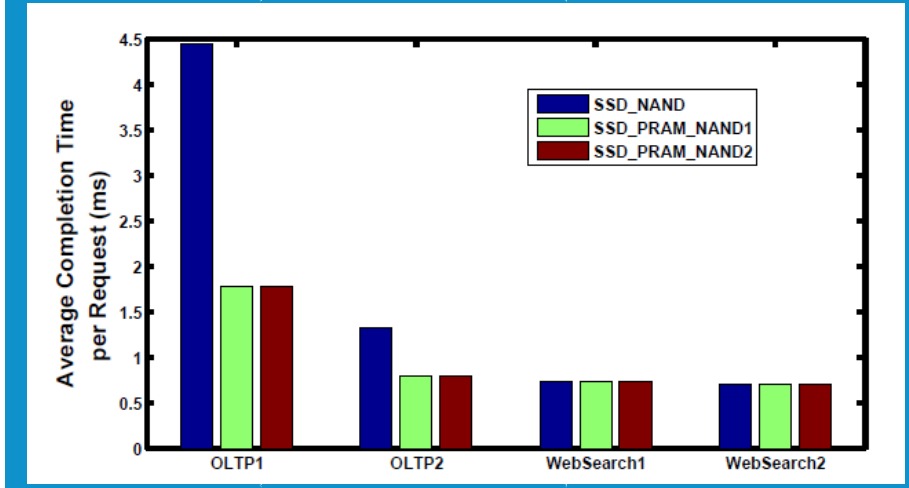
*Y. Kim, B. Tauras, A. Gupta, B. Urgaonkar, Flashsim: A simulator for nand flash-based solid-state drives, Proceedings of International Conference on Advances in System Simulation, 2009, pp. 125–131

Hybrid Storage of PRAM and NAND Flash Memory

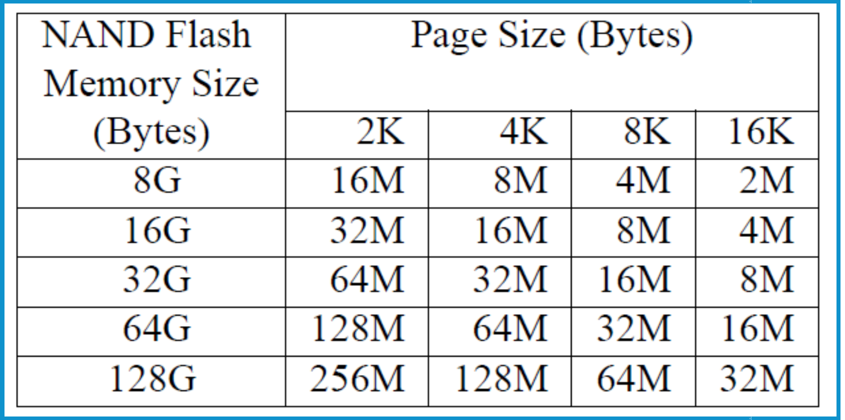
Results



PC Workloads



Server Workloads



NAND Flash Memory Size (Bytes)	Page Size (Bytes)			
	2K	4K	8K	16K
8G	16M	8M	4M	2M
16G	32M	16M	8M	4M
32G	64M	32M	16M	8M
64G	128M	64M	32M	16M
128G	256M	128M	64M	32M

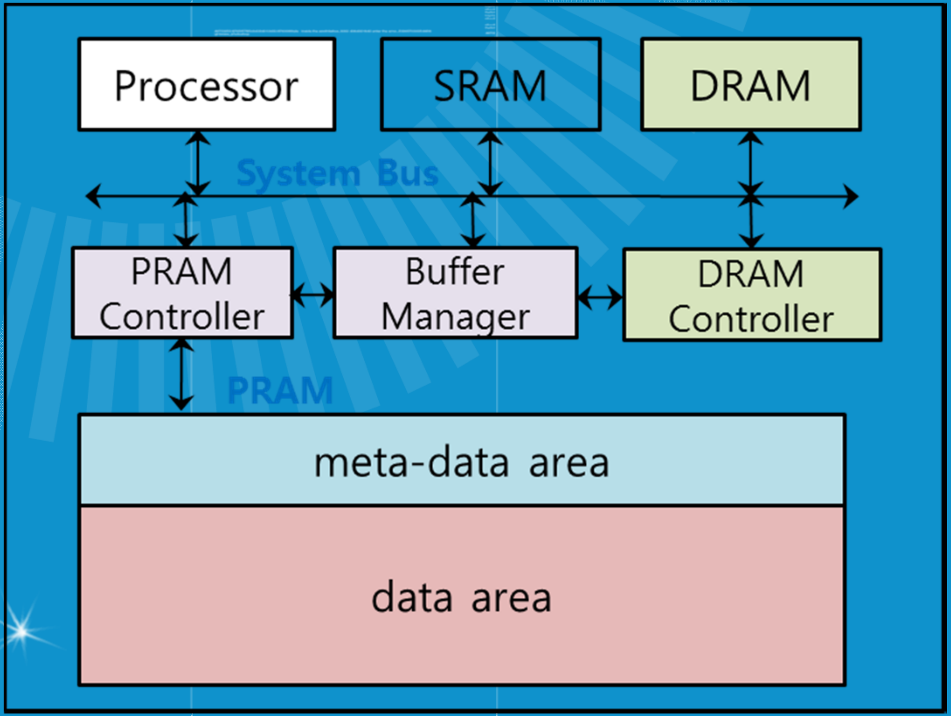
The Required PRAM Size to Manage Meta-data of NAND Flash Memory

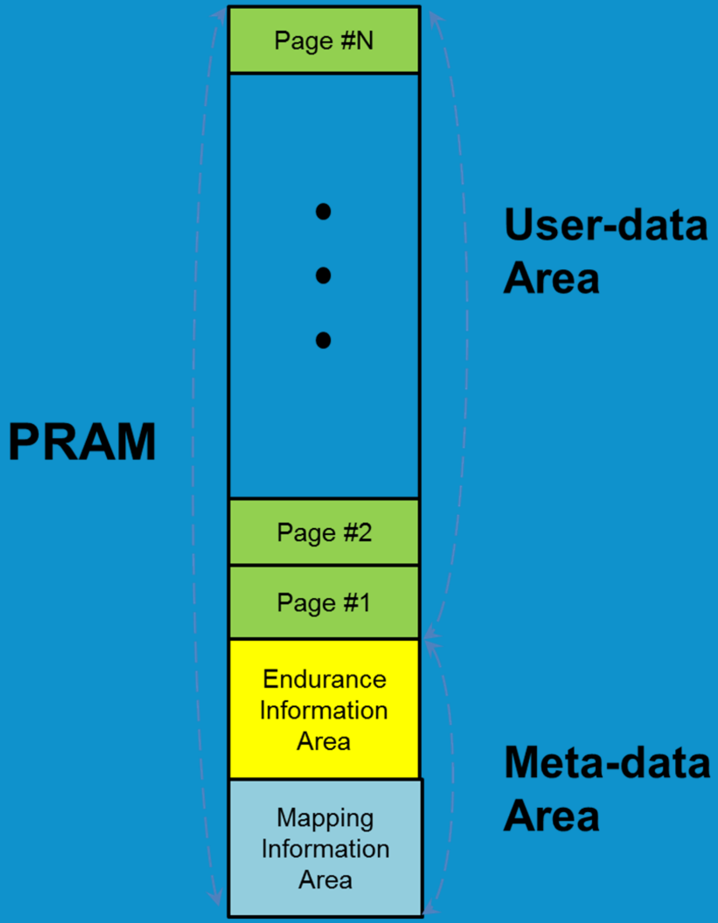
PRAM-only Storage

- Motivation
  - Use PRAM as main memory or storage
  - Need to solve PRAM's endurance problem
    - The prior proposed wear-leveling mechanisms are mainly related to bit-level write reduction
  - Propose a new wear-level mechanism

PRAM-only Storage

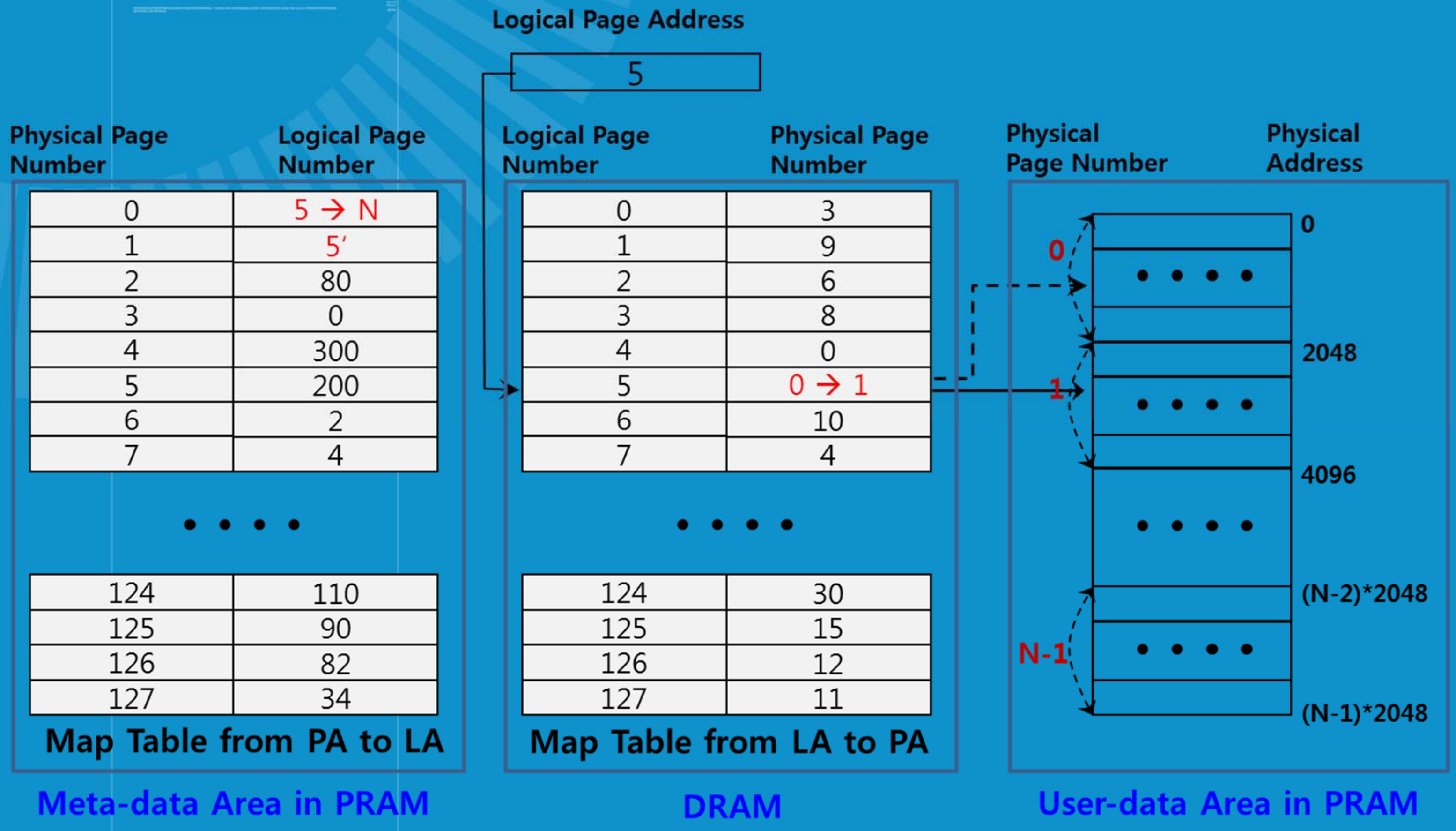
The Proposed Scheme





PRAM-only Storage

The Proposed Scheme



Logical Page Address: 5

Physical Page Number	Logical Page Number
0	5 → N
1	5′
2	80
3	0
4	300
5	200
6	2
7	4
• • • •	
124	110
125	90
126	82
127	34

Map Table from PA to LA

Meta-data Area in PRAM

Logical Page Number	Physical Page Number
0	3
1	9
2	6
3	8
4	0
5	0 → 1
6	10
7	4
• • • •	
124	30
125	15
126	12
127	11

Map Table from LA to PA

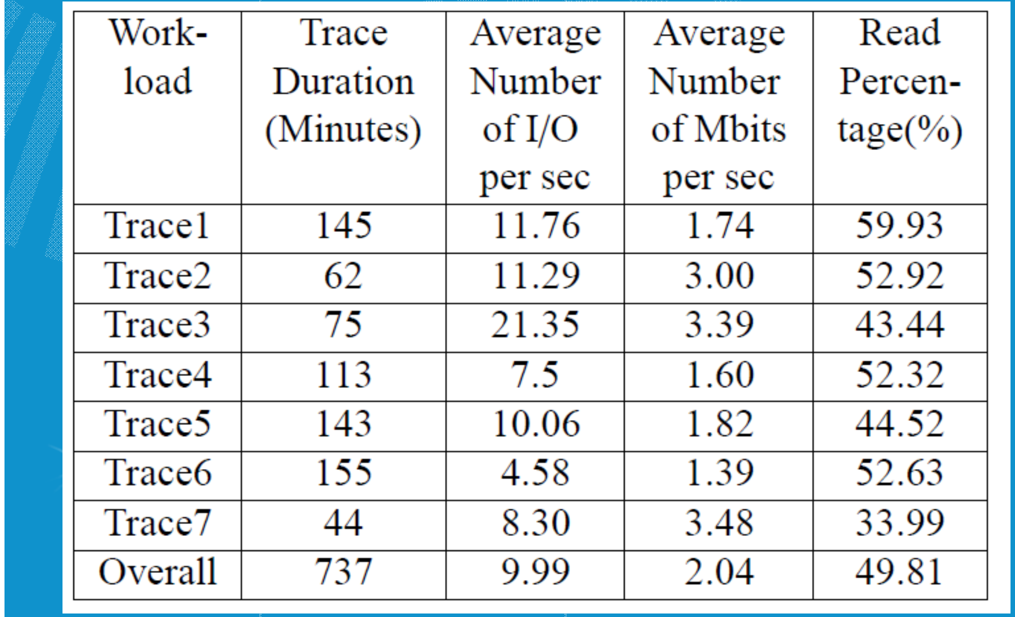
DRAM

User-data Area in PRAM

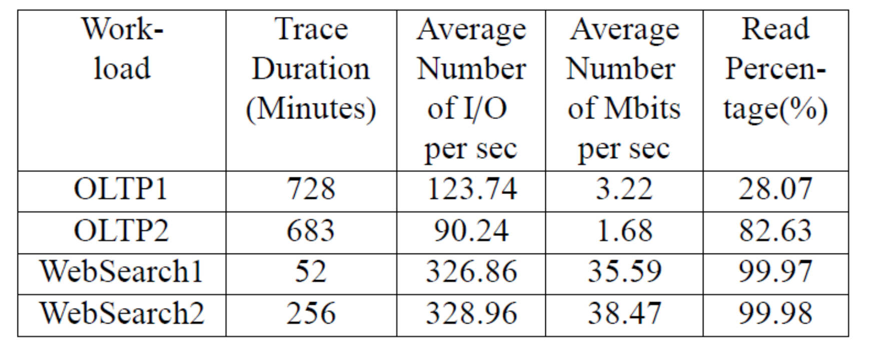
PRAM-only Storage

Experimental Setup

- Modify FlashSim* to support PRAM



Work-load	Trace Duration (Minutes)	Average Number of I/O per sec	Average Number of Mbits per sec	Read Percen-tage(%)
Trace1	145	11.76	1.74	59.93
Trace2	62	11.29	3.00	52.92
Trace3	75	21.35	3.39	43.44
Trace4	113	7.5	1.60	52.32
Trace5	143	10.06	1.82	44.52
Trace6	155	4.58	1.39	52.63
Trace7	44	8.30	3.48	33.99
Overall	737	9.99	2.04	49.81

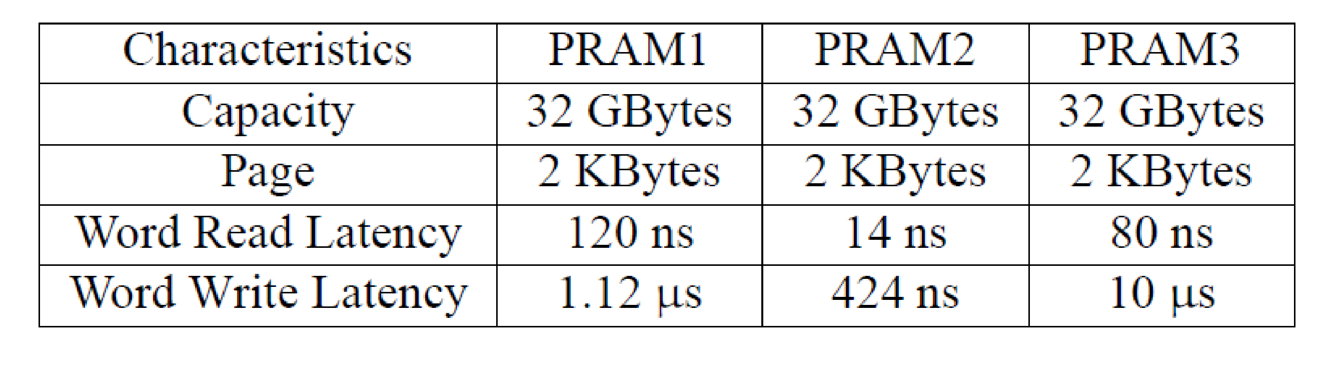


Work-load	Trace Duration (Minutes)	Average Number of I/O per sec	Average Number of Mbits per sec	Read Percen-tage(%)
OLTP1	728	123.74	3.22	28.07
OLTP2	683	90.24	1.68	82.63
WebSearch1	52	326.86	35.59	99.97
WebSearch2	256	328.96	38.47	99.98

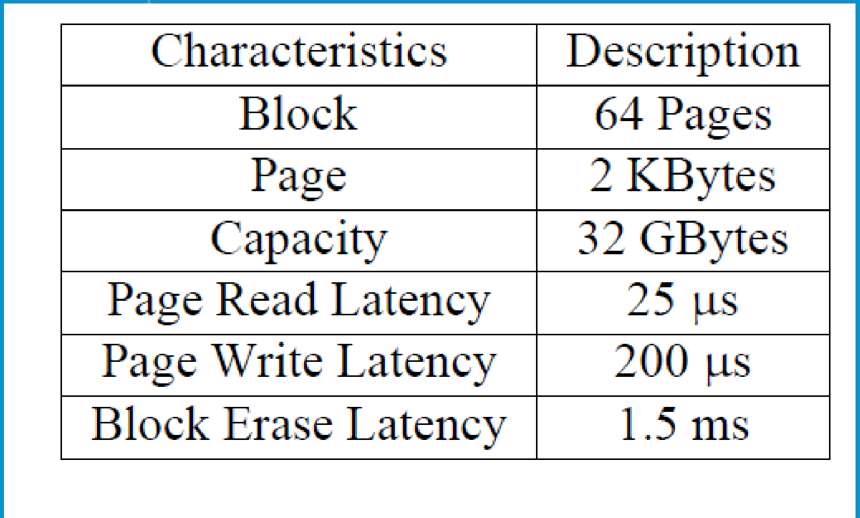
*Y. Kim, B. Tauras, A. Gupta, B. Urgaonkar, Flashsim: A simulator for nand flash-based solid-state drives, Proceedings of International Conference on Advances in System Simulation, 2009, pp. 125–131

PRAM-only Storage

Experimental Setup



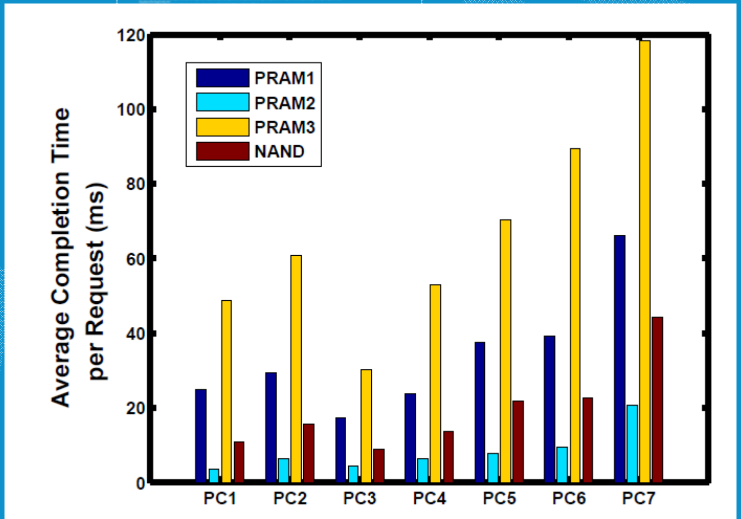
Characteristics	PRAM1	PRAM2	PRAM3
Capacity	32 GBytes	32 GBytes	32 GBytes
Page	2 KBytes	2 KBytes	2 KBytes
Word Read Latency	120 ns	14 ns	80 ns
Word Write Latency	1.12 μs	424 ns	10 μs



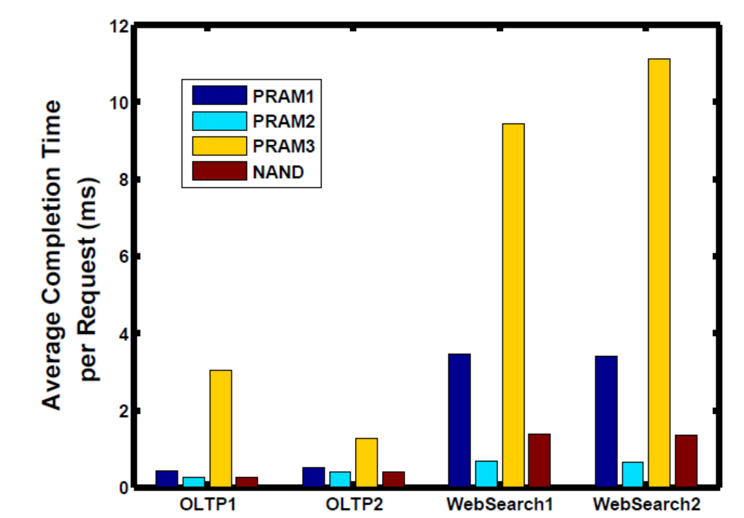
Characteristics	Description
Block	64 Pages
Page	2 KBytes
Capacity	32 GBytes
Page Read Latency	25 μs
Page Write Latency	200 μs
Block Erase Latency	1.5 ms

PRAM-only Storage

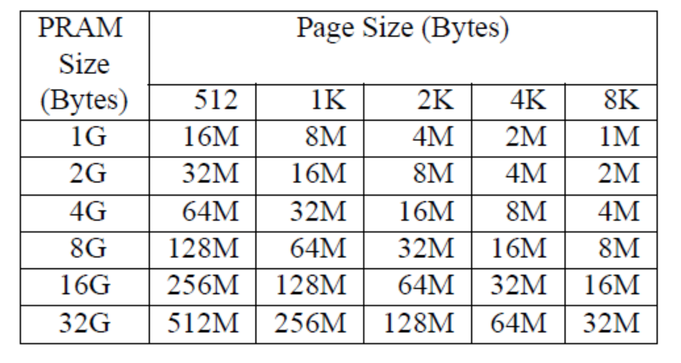
Results



PC Workloads



Server Workloads



PRAM Size (Bytes)	Page Size (Bytes)				
	512	1K	2K	4K	8K
1G	16M	8M	4M	2M	1M
2G	32M	16M	8M	4M	2M
4G	64M	32M	16M	8M	4M
8G	128M	64M	32M	16M	8M
16G	256M	128M	64M	32M	16M
32G	512M	256M	128M	64M	32M

The Required PRAM Size to Manage Meta-data of NAND Flash Memory

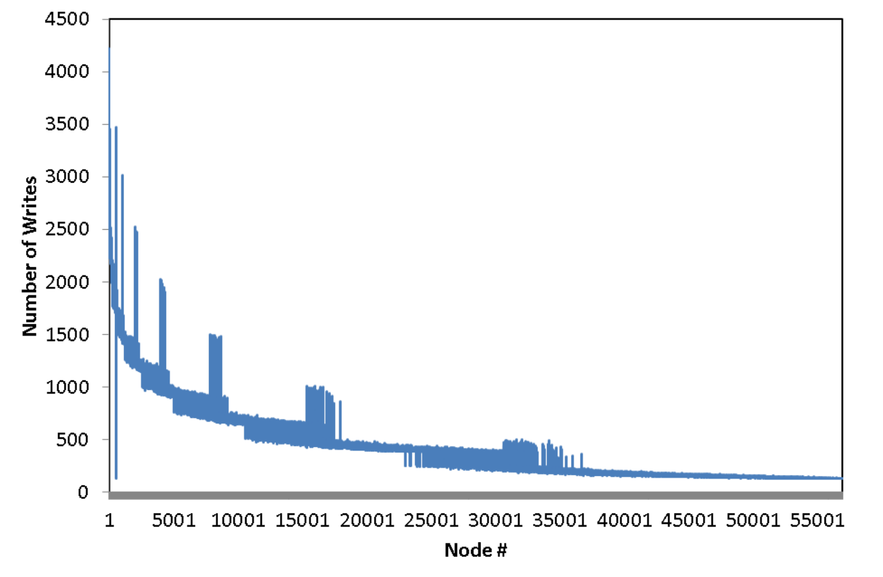
B+Tree in PRAM

B+Tree

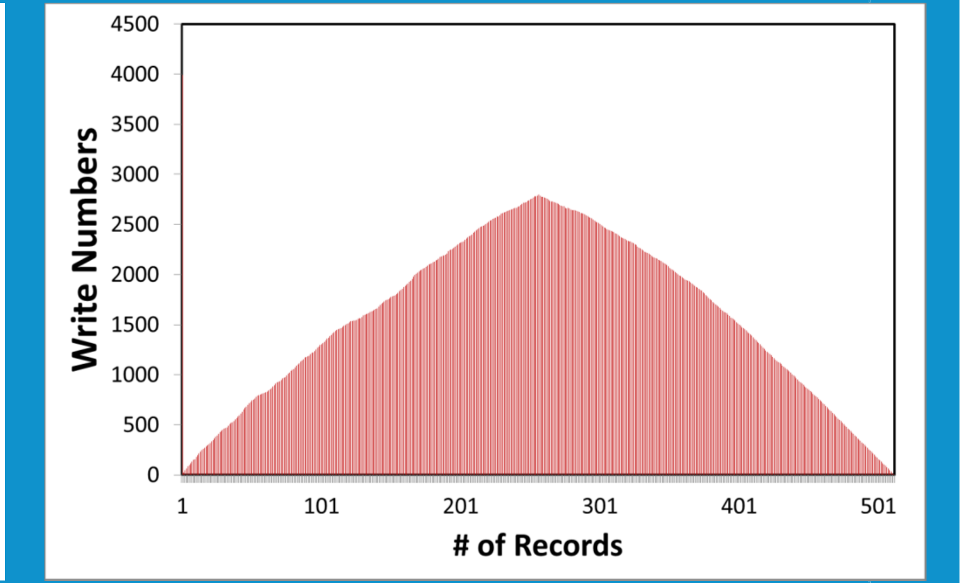
- All paths from root to leaf are of the same length
- Each node has between $\lceil n/2 \rceil$ and n pointers. Each leaf node stores between $\lceil (n-1)/2 \rceil$ and $n-1$ values.
- n is called fanout (it corresponds to the maximum number of pointers/children). The value $\lceil (n-1)/2 \rceil$ is called order (it corresponds to the minimum number of values).
- Special cases:
  - If the root is not a leaf, it has at least 2 children.
  - If the root is a leaf (that is, there are no other nodes in the tree), it can have between 0 and $(n-1)$ values.

B+Tree in PRAM

Motivation



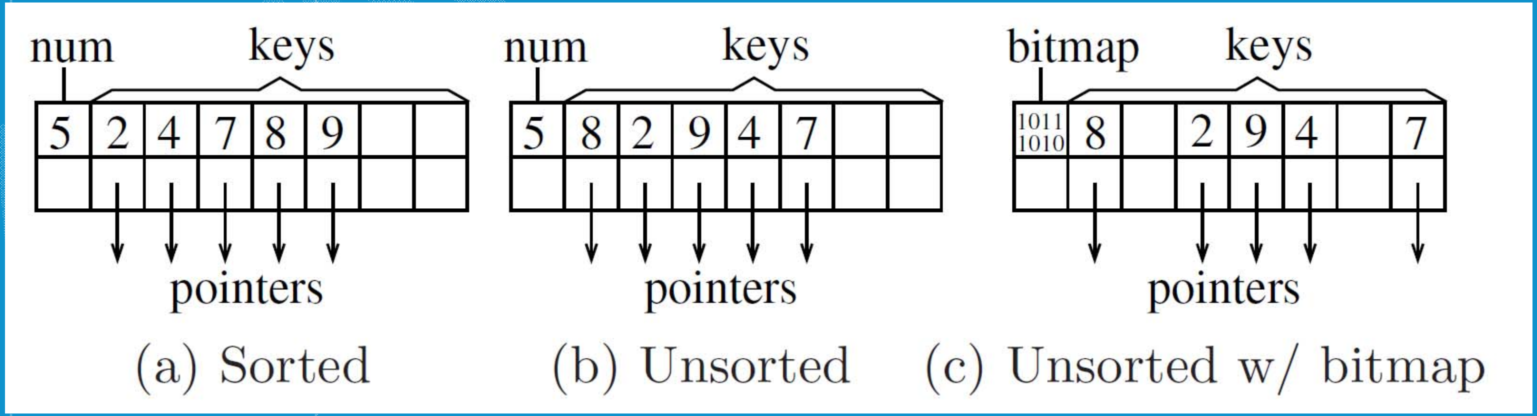
Write Numbers per Node



Write Number per Record within a Node

B+Tree in PRAM

- Prior Works
  - B+Tree with unsorted leaf nodes*



* Shimin chen, Phillip B. Gibbons and Suman Nath, Rethinking Database Algorithms for Phase Change Memory, CIDR 2011

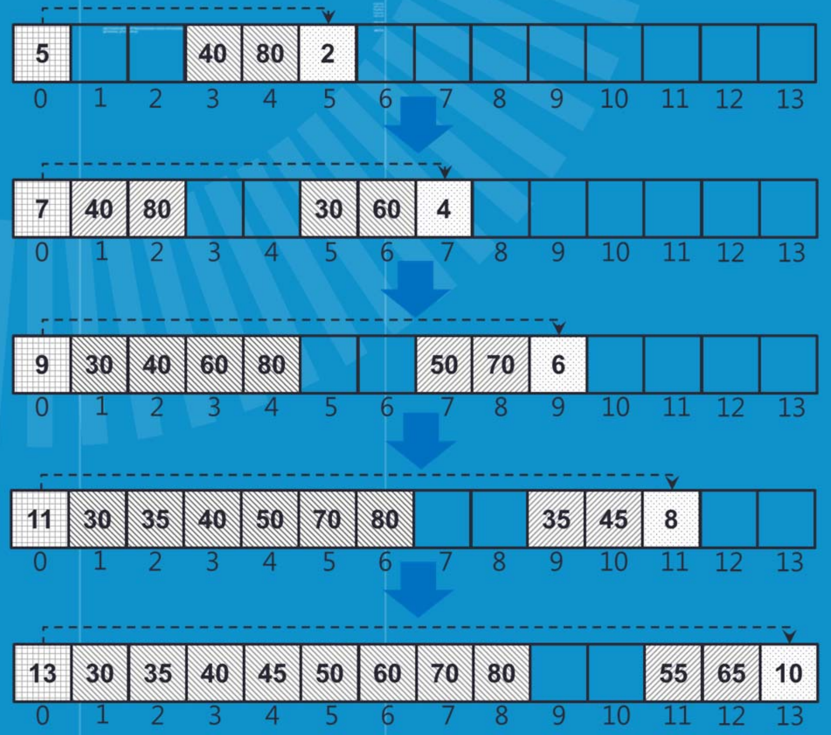
B+Tree in PRAM: P+TREE

- The Proposed Scheme
  - Divide a node into Area1 and Area2
  - Area2 is much smaller than Area1
  - A new key/record is first inserted into Area2
  - If Area2 is full, merge Area1 and Area2 into Area1
  - Search Both Area1 and Area2

B+Tree in PRAM: P+TREE

The Proposed Scheme: Insert Operation

Insert 40, 80, 60, 30, 70, 50, 35, 45, 55 and 65 sequentially



0	1	2	3	4	5	6	7	8	9	10	11	12	13
5			40	80	2								

0	1	2	3	4	5	6	7	8	9	10	11	12	13
7	40	80			30	60	4						

0	1	2	3	4	5	6	7	8	9	10	11	12	13
9	30	40	60	80			50	70	6				

0	1	2	3	4	5	6	7	8	9	10	11	12	13
11	30	35	40	50	70	80			35	45	8		

0	1	2	3	4	5	6	7	8	9	10	11	12	13
13	30	35	40	45	50	60	70	80			55	65	10

- Node Size: 14
- Fanout: 10
- Area2's size: 2

▦ The pointer to the location to hold the number of records in the node

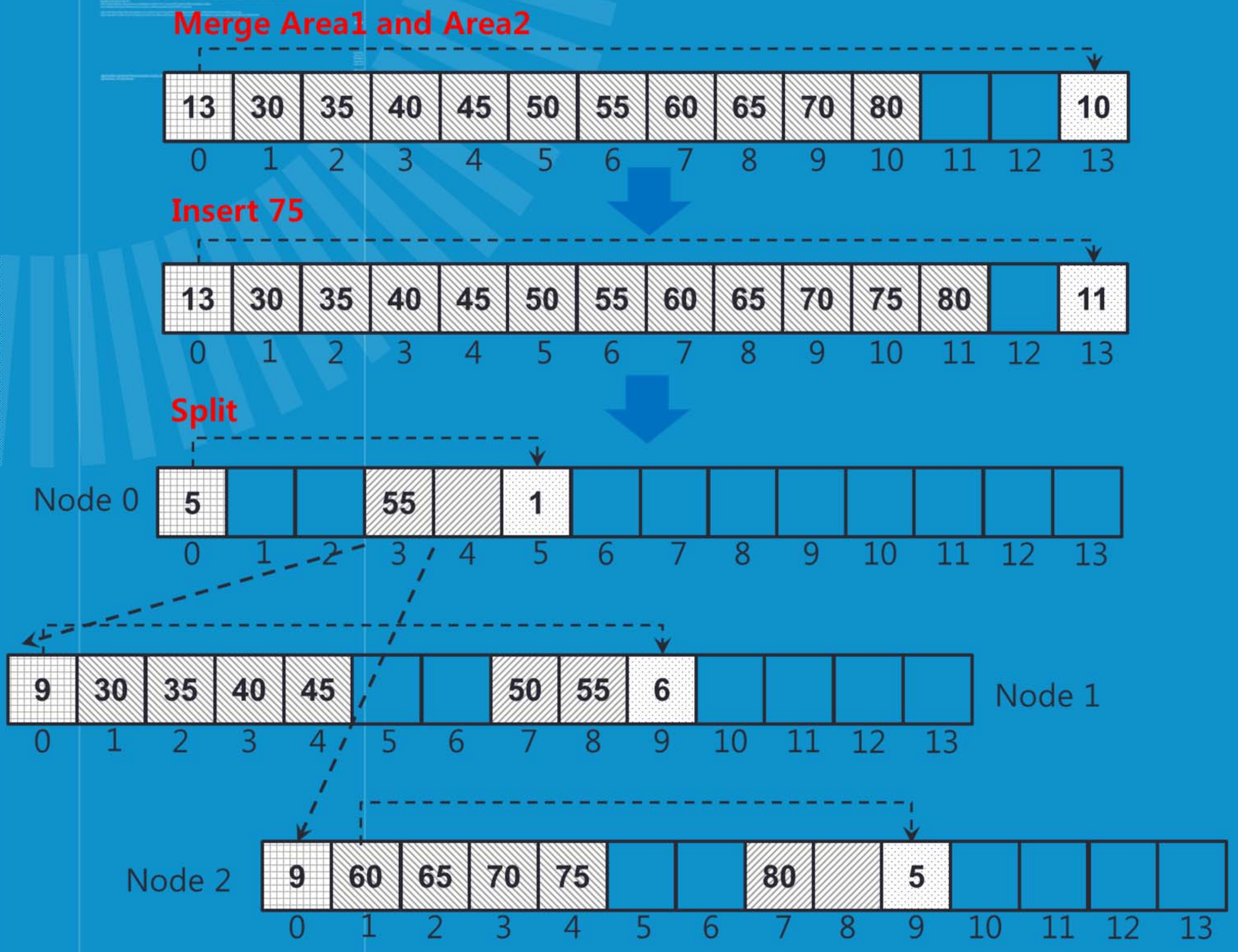
▧ Area 1

▨ Area 2

░ The number of records

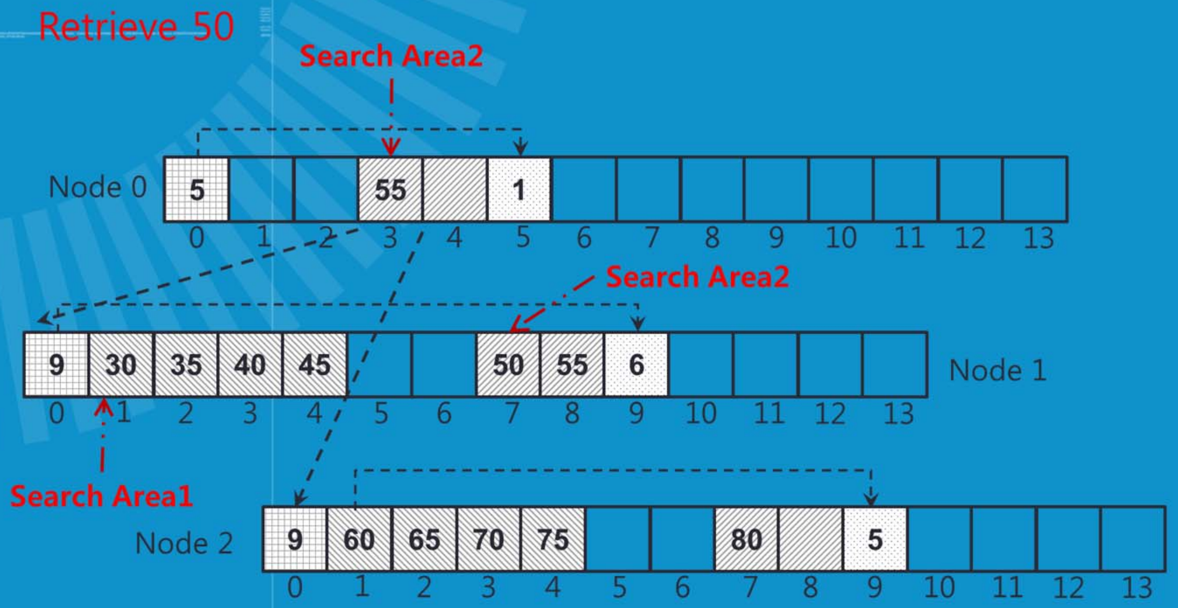
B+Tree in PRAM: P+TREE

The Proposed Scheme: Split Operation



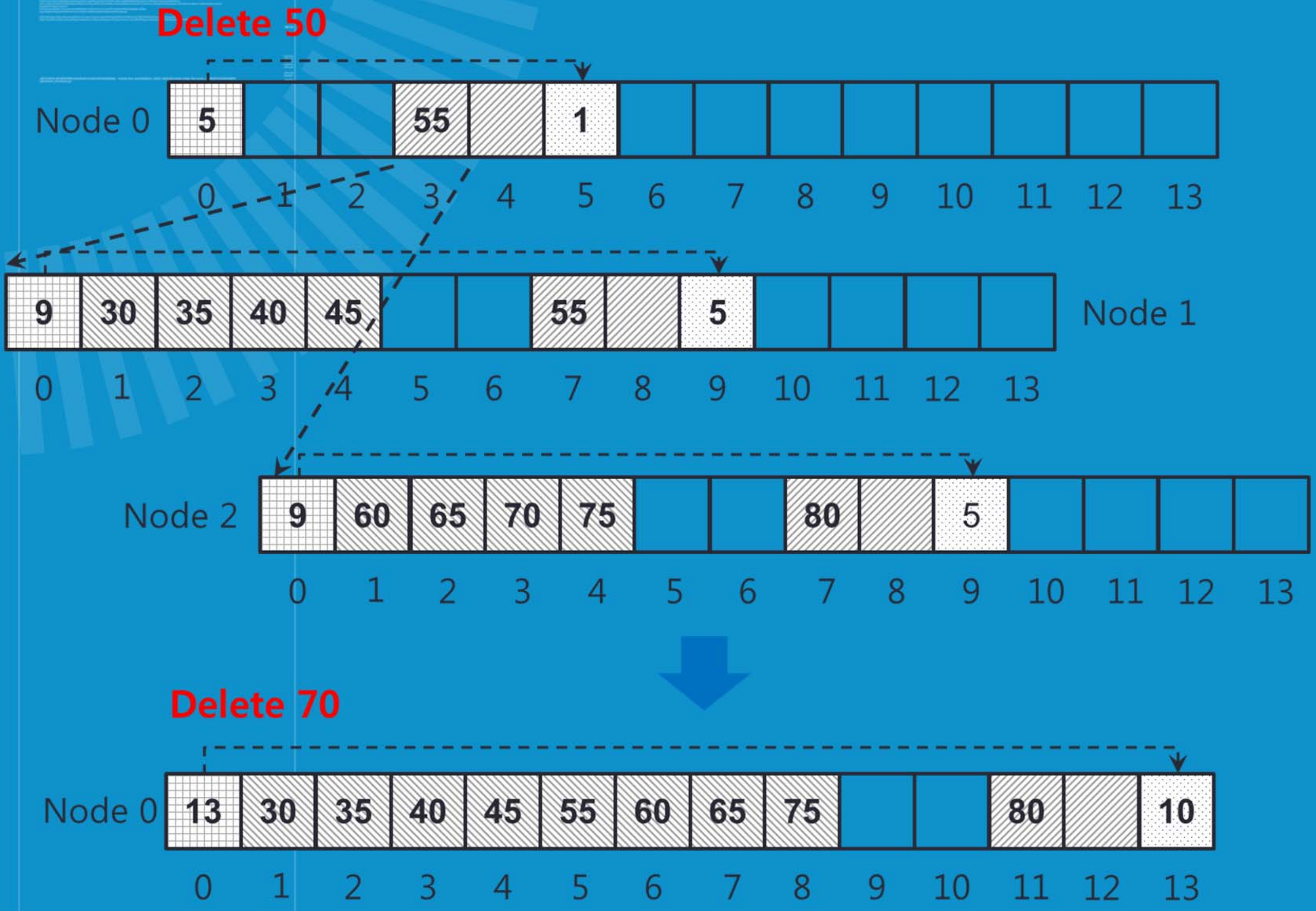
B+Tree in PRAM: P+TREE

- The Proposed Scheme: Retrieve Operation



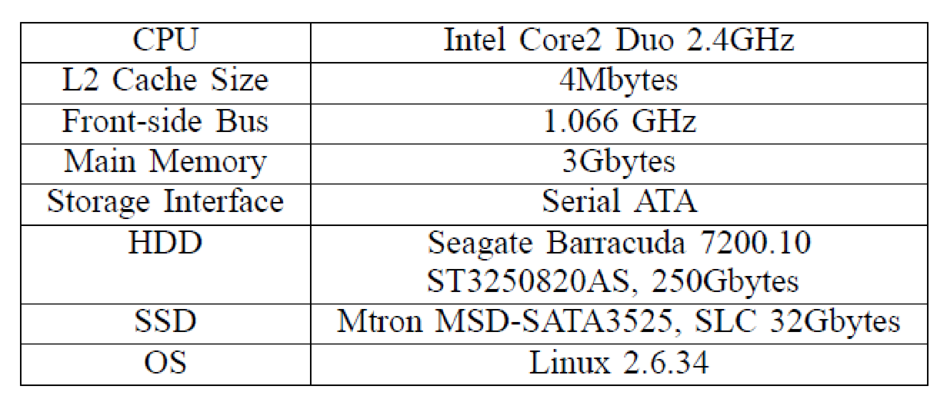
B+Tree in PRAM: P+TREE

The Proposed Scheme: Delete Operation

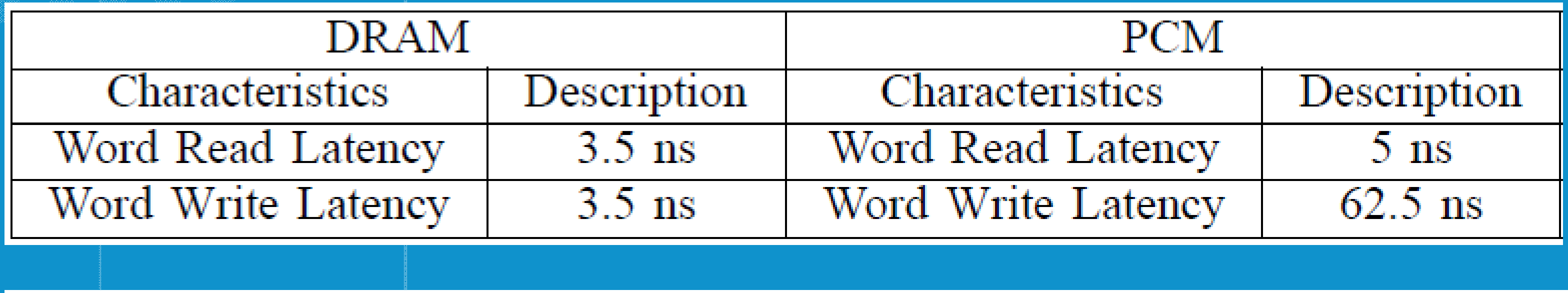


B+Tree in PRAM: P+TREE

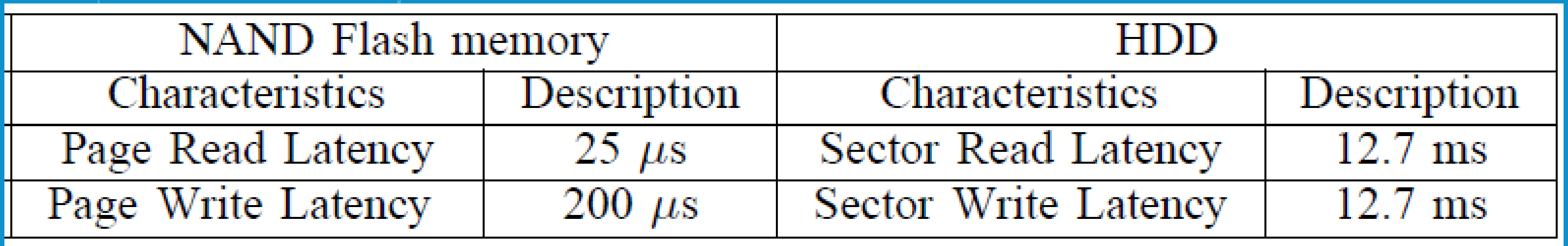
Experimental Setup



CPU	Intel Core2 Duo 2.4GHz
L2 Cache Size	4Mbytes
Front-side Bus	1.066 GHz
Main Memory	3Gbytes
Storage Interface	Serial ATA
HDD	Seagate Barracuda 7200.10 ST3250820AS, 250Gbytes
SSD	Mtron MSD-SATA3525, SLC 32Gbytes
OS	Linux 2.6.34



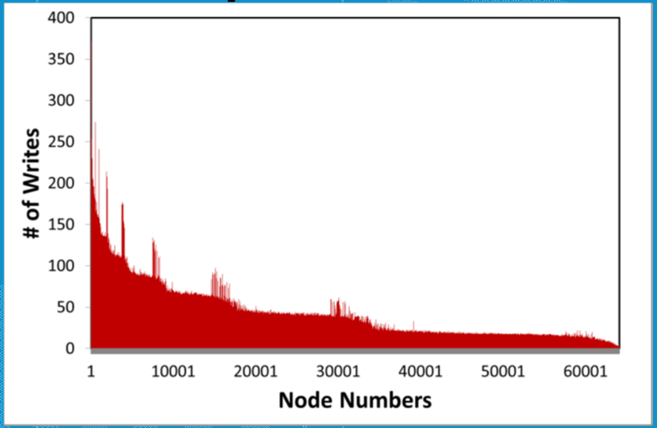
DRAM		PCM	
Characteristics	Description	Characteristics	Description
Word Read Latency	3.5 ns	Word Read Latency	5 ns
Word Write Latency	3.5 ns	Word Write Latency	62.5 ns



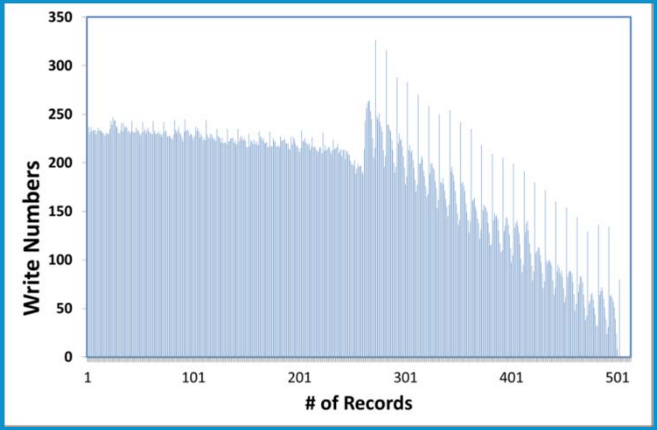
NAND Flash memory		HDD	
Characteristics	Description	Characteristics	Description
Page Read Latency	25 μs	Sector Read Latency	12.7 ms
Page Write Latency	200 μs	Sector Write Latency	12.7 ms

B+Tree in PRAM: P+TREE

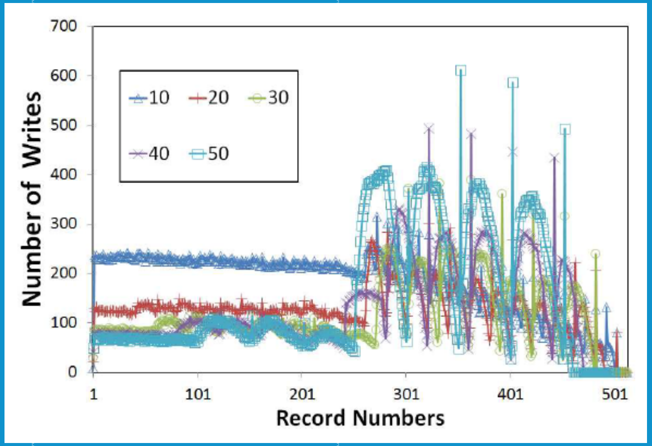
- The Proposed Scheme: Area2's size and Endurance



Write Numbers per Node



Write Number per Record within a Node



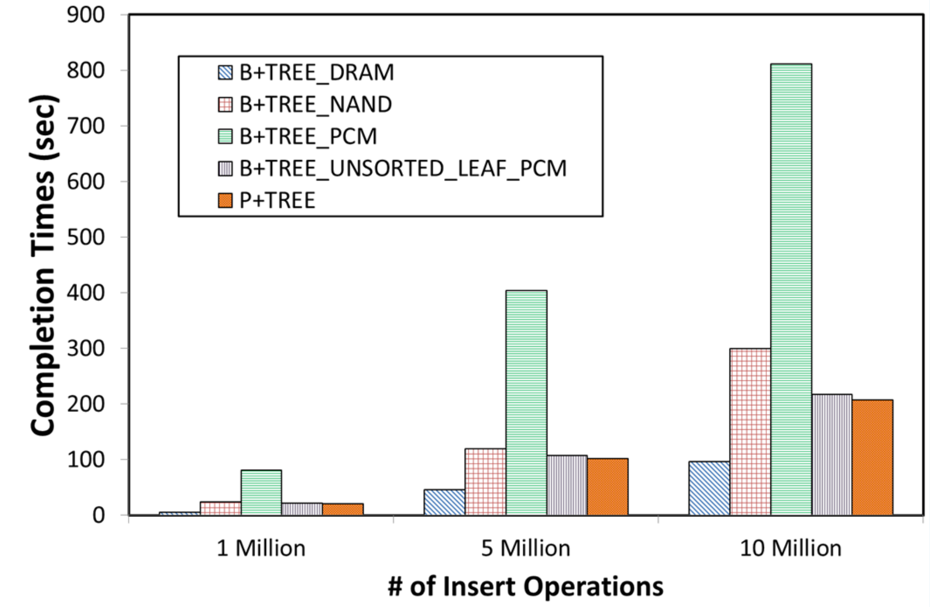
The Comparison of Write Numbers by Varying the Size of Area2

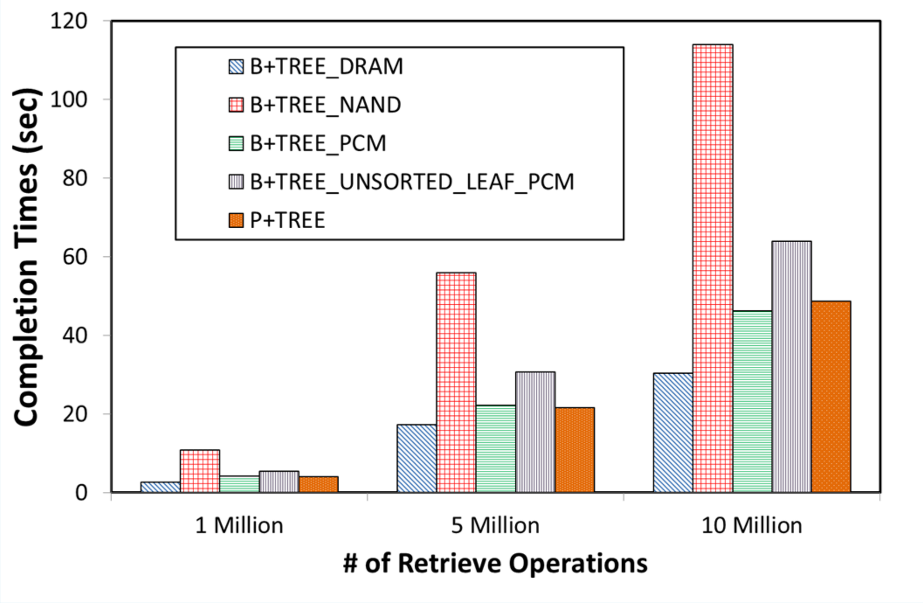
B+Tree in PRAM: P+TREE

- Results:

B+Tree in PRAM

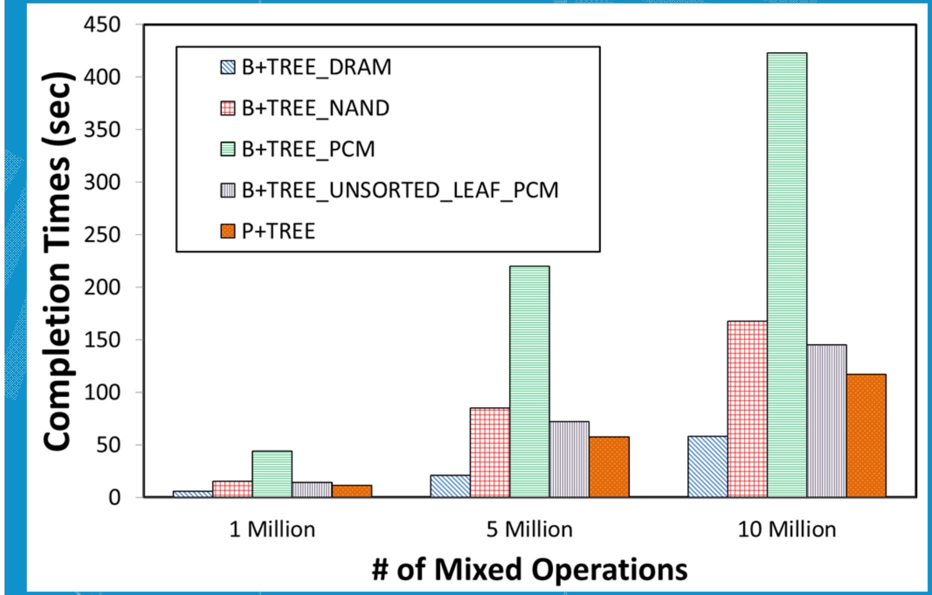
Results: Insert and Retrieve Operations

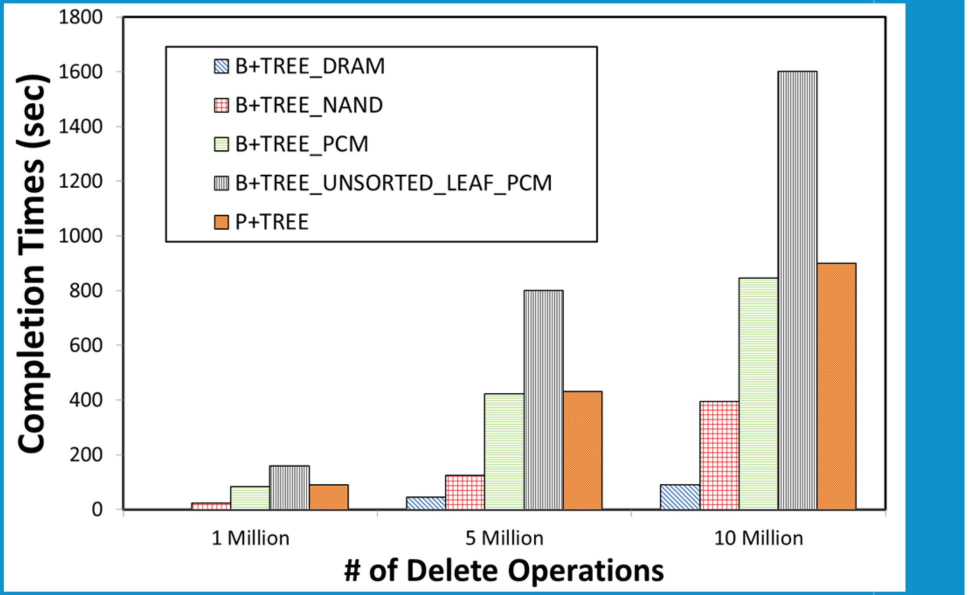




B+Tree in PRAM

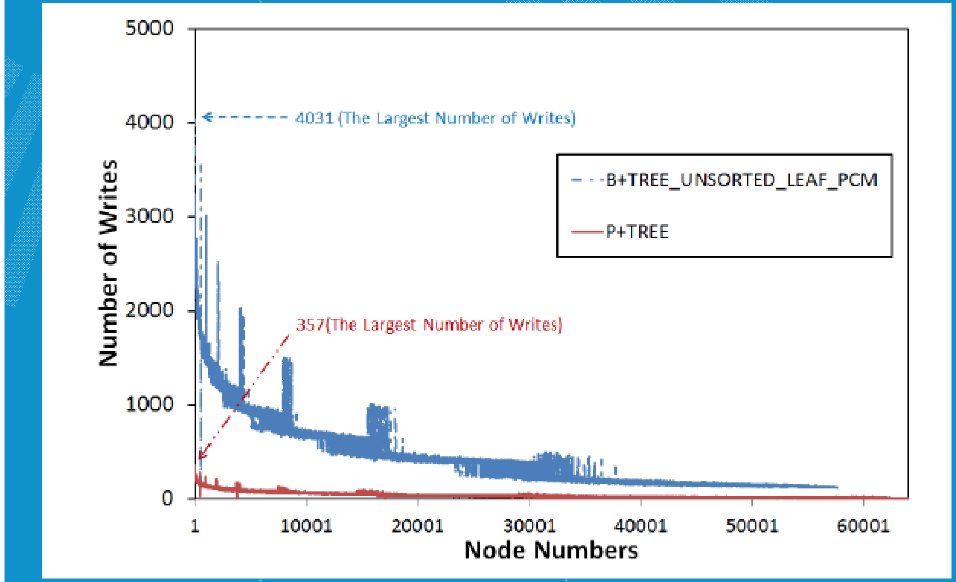
Results: Mixed and Delete Operations

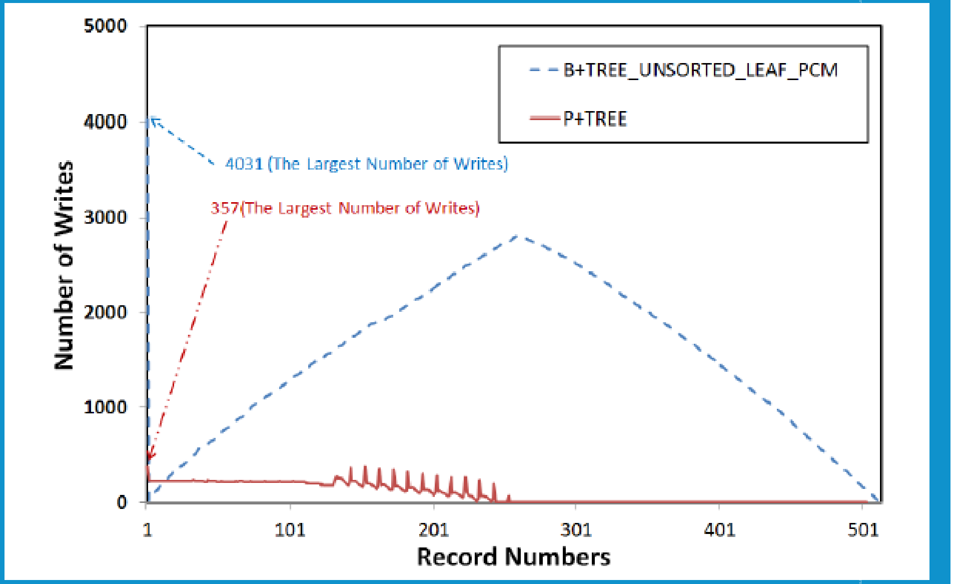




B+Tree in PRAM

Results: Endurance





Conclusions and Future Works

- The proposed hybrid storage of PRAM and NAND solve the PRAM's endurance problem perfectly
- PTL could be used in a storage device and even a memory controller
- P+TREE shows better performance and endurance compared to the original scheme

- Will propose new data structures for PRAM
- Evaluate the proposed schemes in real PRAM board.

THANKS!
QUESTIONS?