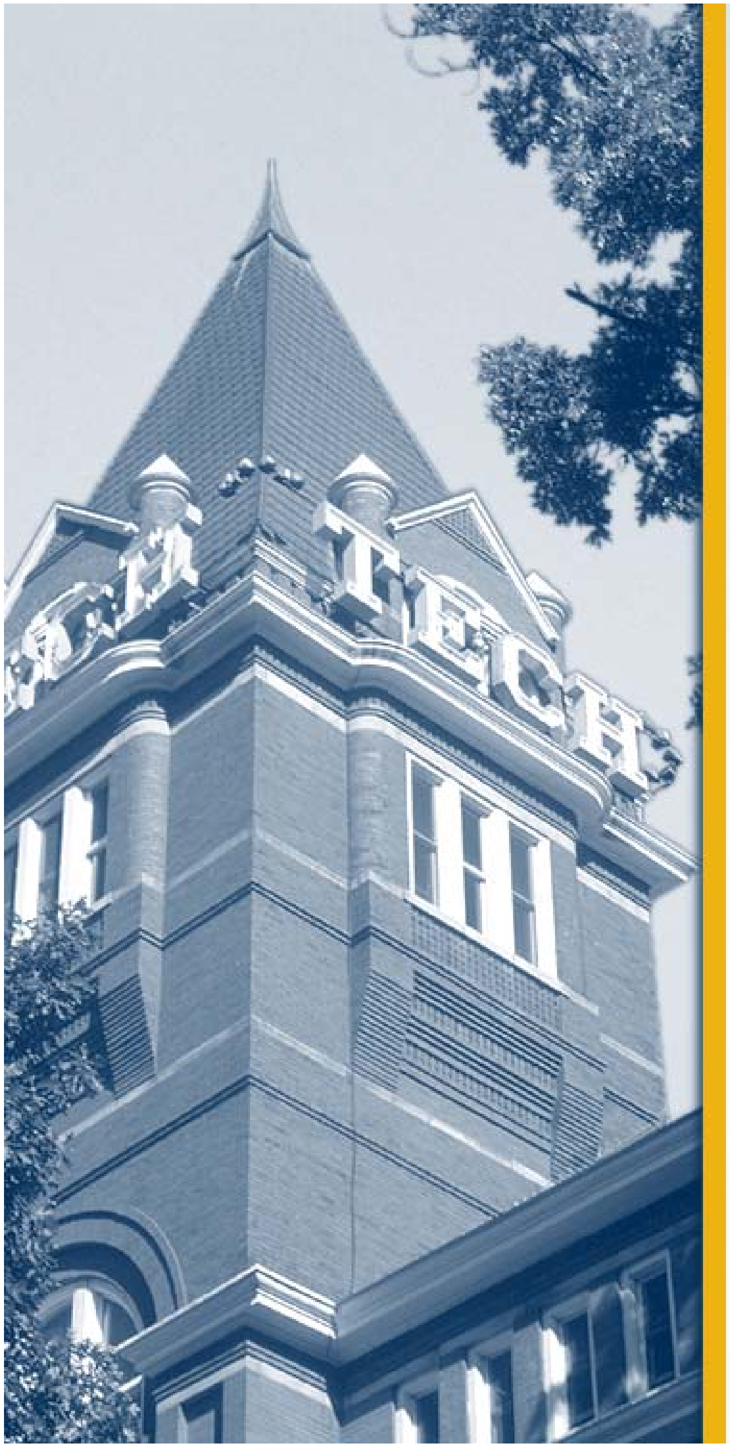


A Semi-Preemptive Garbage Collector for Solid State Drives

Junghee Lee, Youngjae Kim, Galen M. Shipman,
Sarp Oral, Feiyi Wang, and Jongman Kim

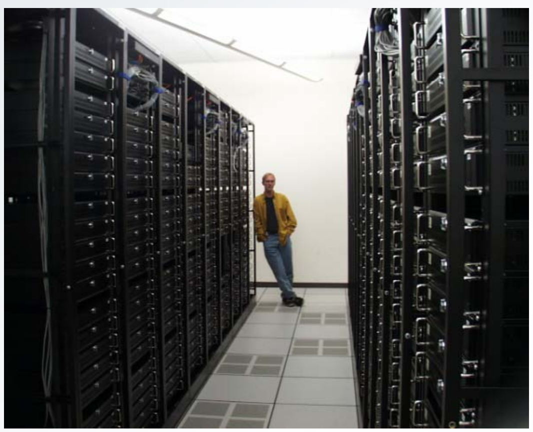
Presented by Junghee Lee

OAK RIDGE NATIONAL LABORATORY
MANAGED BY UT-BATTELLE FOR THE DEPARTMENT OF ENERGY

Georgia Institute of Technology

High Performance Storage Systems

- Server centric services
  - File, web & media servers, transaction processing servers
- Enterprise-scale Storage Systems
  - Information technology focusing on storage, protection, retrieval of data in large-scale environments



High Performance Storage Systems

Storage Unit Hard Disk Drive

Spider: A Large-scale Storage System

- Jaguar
  - Peta-scale computing machine
  - 25,000 nodes with 250,000 cores and over 300 TB memory
- Spider storage system
  - The largest center-wide Lustre-based file system
  - Over 10.7 PB of RAID 6 formatted capacity
    - 13,400 x 1 TB HDDs
  - 192 Lustre I/O servers
    - Over 3TB of memory (on Lustre I/O servers)

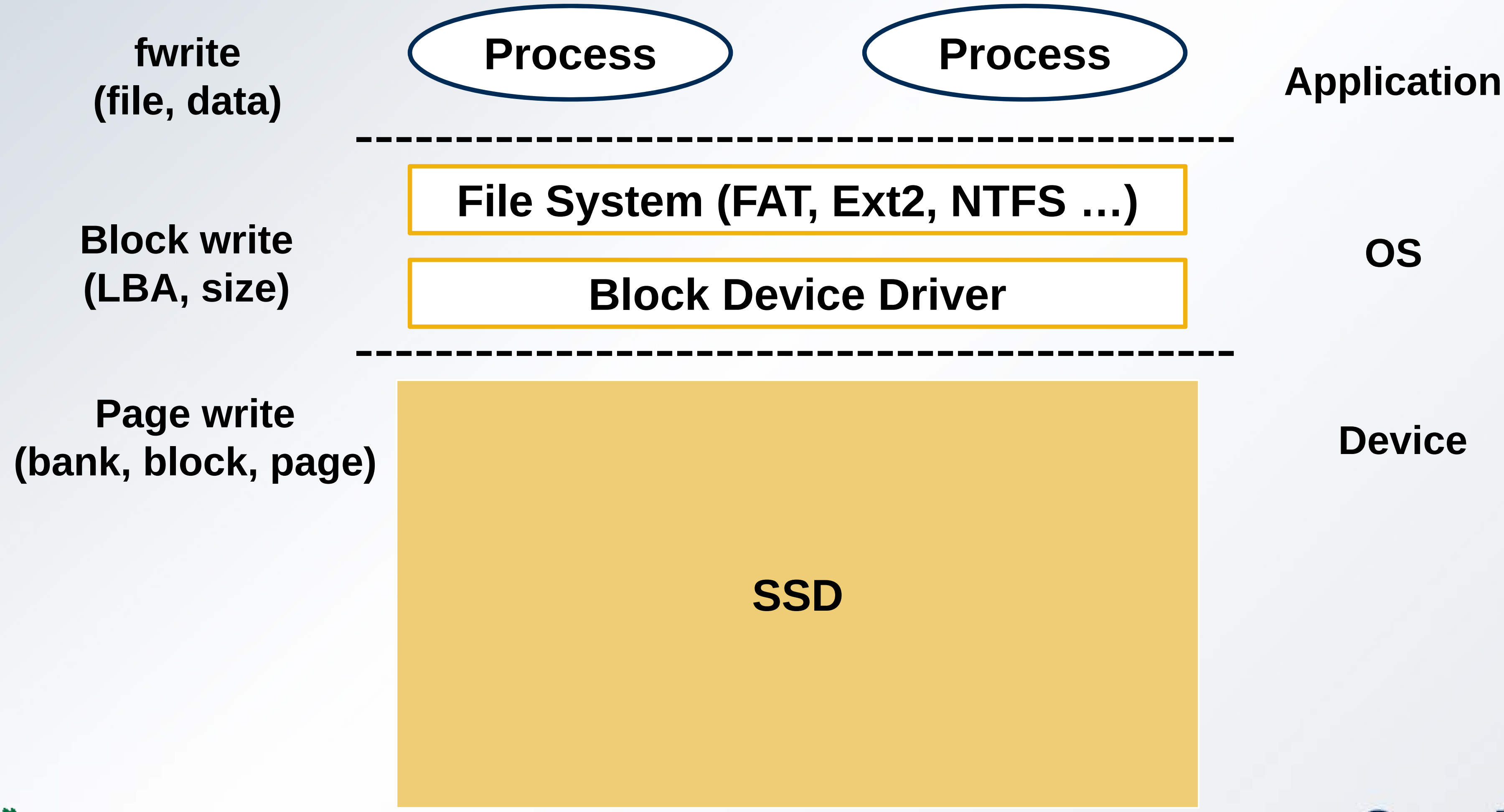
Emergence of NAND Flash based SSD

- NAND Flash vs. Hard Disk Drives
  - Pros:
    - Semi-conductor technology, no mechanical parts
    - Offer lower access latencies
      - *μs* for SSDs vs. *ms* for HDDs
    - Lower power consumption
    - Higher robustness to vibrations and temperature
  - Cons:
    - Limited lifetime
      - 10K - 1M erases per block
    - High cost
      - About 8X more expensive than current hard disks
    - ***Performance variability***

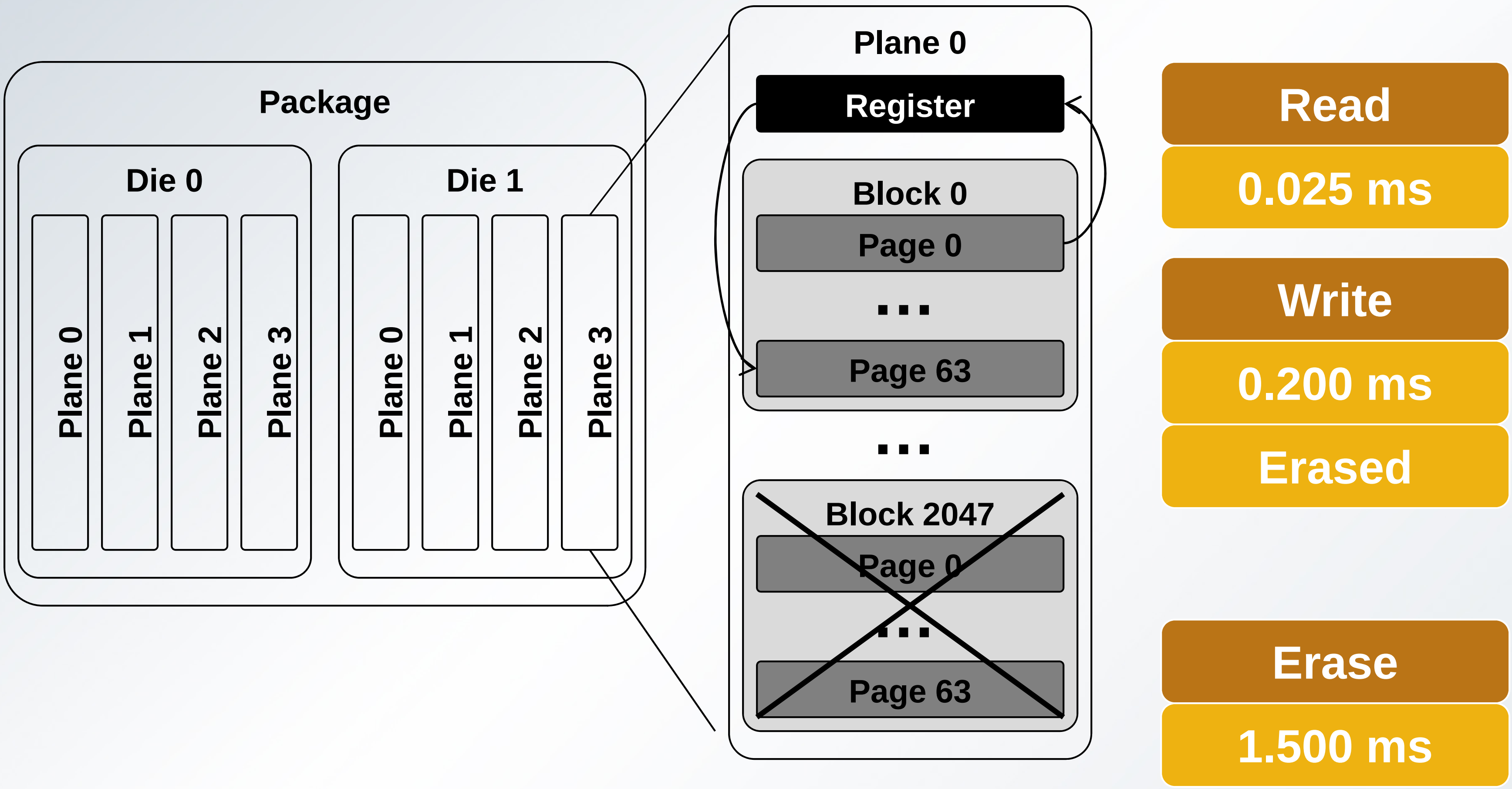
Outline

- Introduction
- Background and Motivation
  - NAND Flash and SSD
  - Garbage Collection
  - Pathological Behavior of SSDs
- Semi-Preemptive Garbage Collection
- Evaluation
- Conclusion

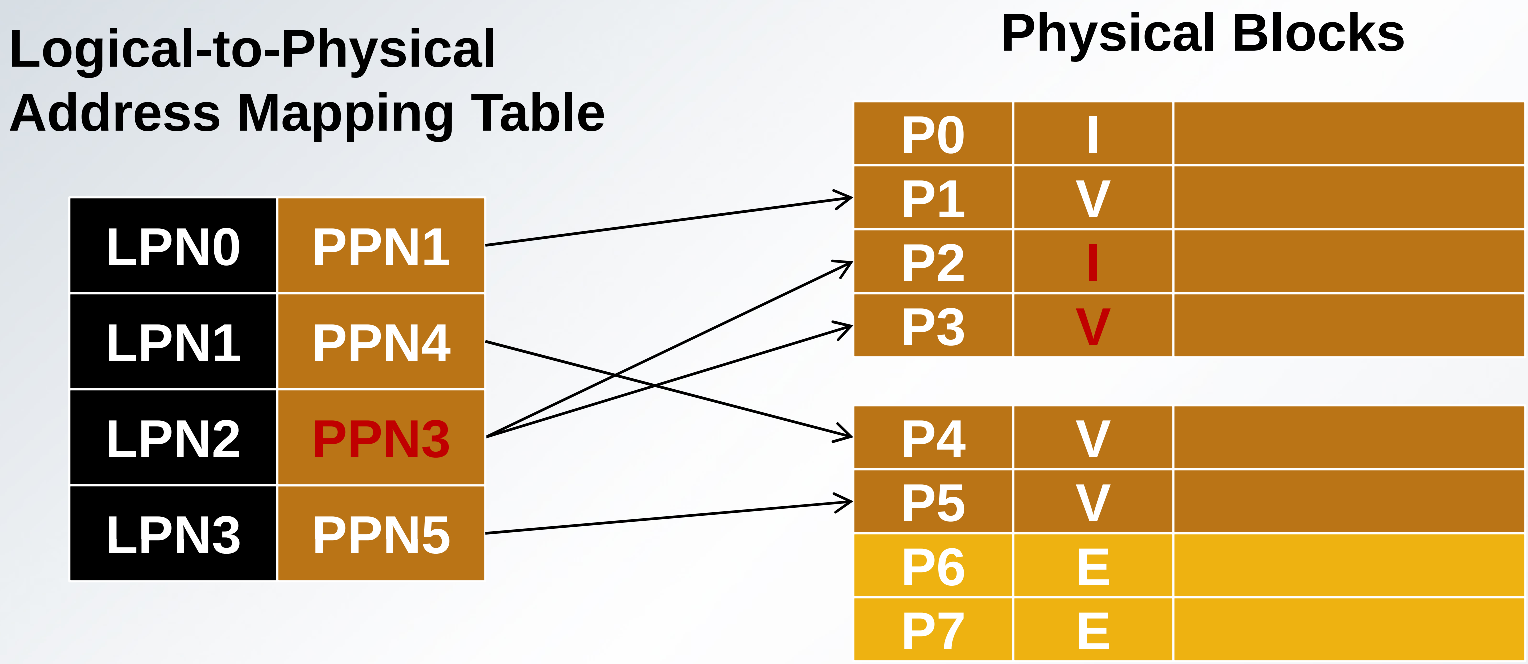
NAND Flash based SSD



NAND Flash Organization



Out-Of-Place Write



Write to LPN2 → Invalidate PPN2 → Write to PPN3 → Update table

Garbage Collection

Select Victim Block

Move Valid Pages

Erase Victim Block

Physical Blocks

P0	E	
P1	E	
P2	E	
P3	E	

P4	V	
P5	V	
P6	V	
P7	V	

2 reads + 2 writes + 1 erase= 2*0.025 + 2*0.200 + 1.5 = 1.950(ms) !!

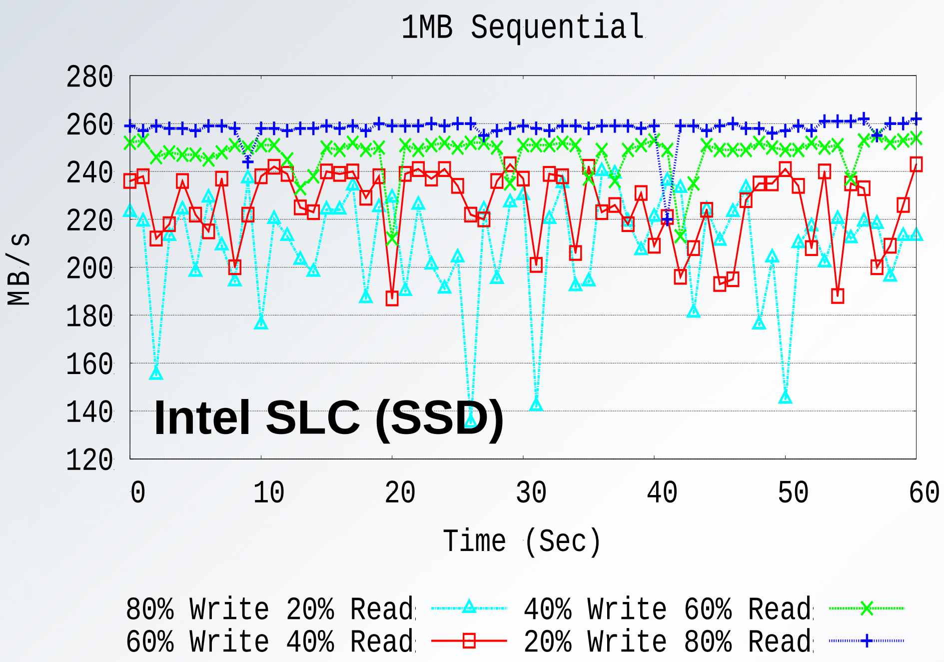
Pathological Behavior of SSDs

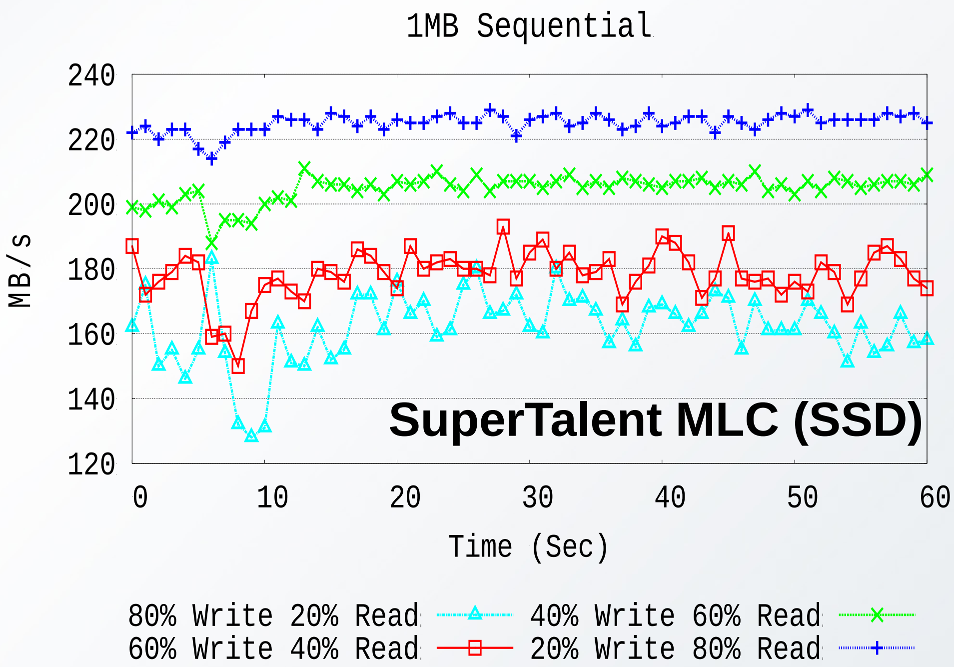
- Does GC have an impact on the foreground operations?
  - If so, we can observe sudden bandwidth drop
  - More drop with more write requests
  - More drop with more bursty workloads

- Experimental Setup
  - SSD devices
    - Intel (SLC) 64GB SSD
    - SuperTalent (MLC) 120GB SSD
  - I/O generator
    - Used *libaio* asynchronous I/O library for block-level testing

Bandwidth Drop for Write-Dominant Workloads

- Experiments
  - Measured bandwidth for 1MB by varying read-write ratio





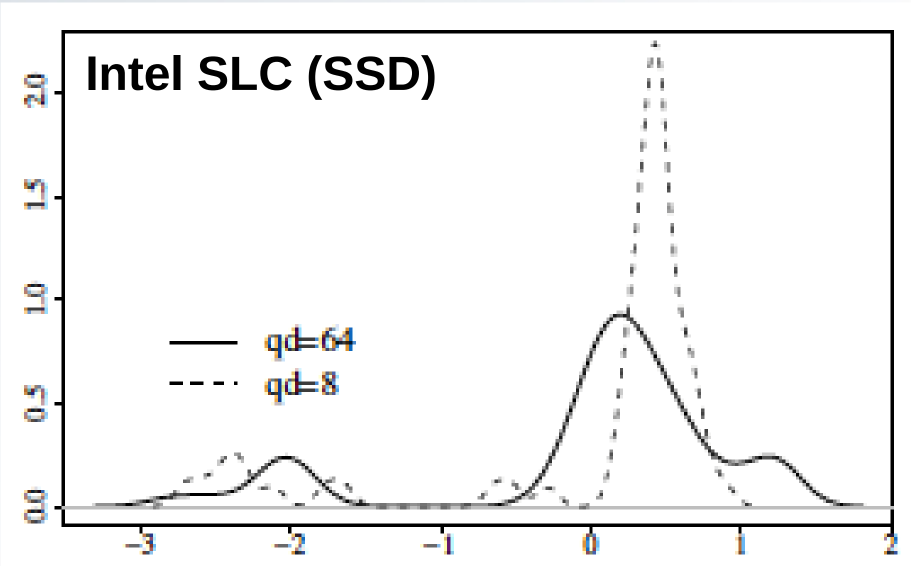
Performance variability increases as we increase write-percentage of workloads.

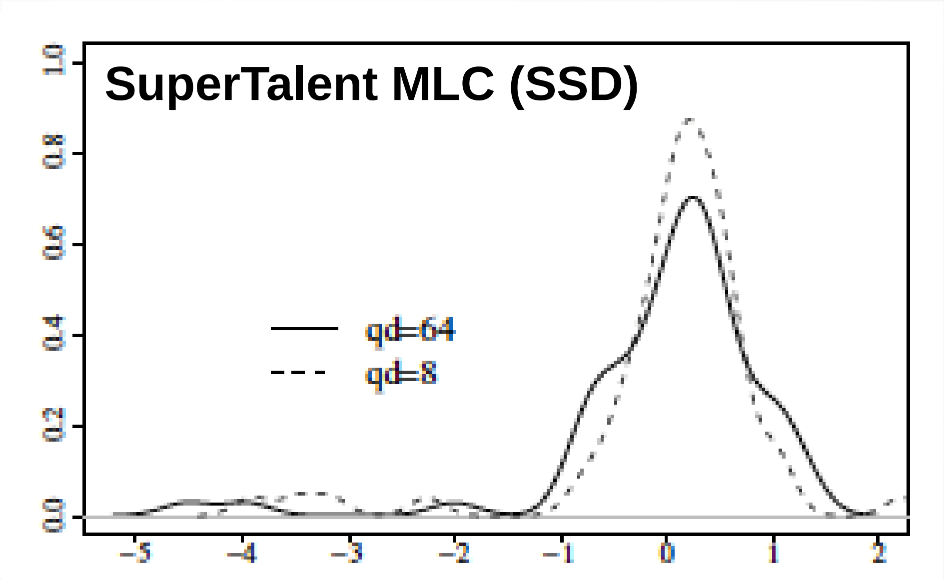
Oak Ridge National Laboratory
MANAGED BY UT-BATTELLE FOR THE DEPARTMENT OF ENERGY

Georgia Tech

Performance Variability for Bursty Workloads

- Experiments
  - Measured SSD write bandwidth for queue depth (qd) is 8 and 64
  - Normalized I/O bandwidth with a Z distribution





Performance variability increases as we increase the arrival-rate of requests (bursty workloads).

OAK RIDGE NATIONAL LABORATORY
MANAGED BY UT-BATTELLE FOR THE DEPARTMENT OF ENERGY

Georgia Tech

Lessons Learned

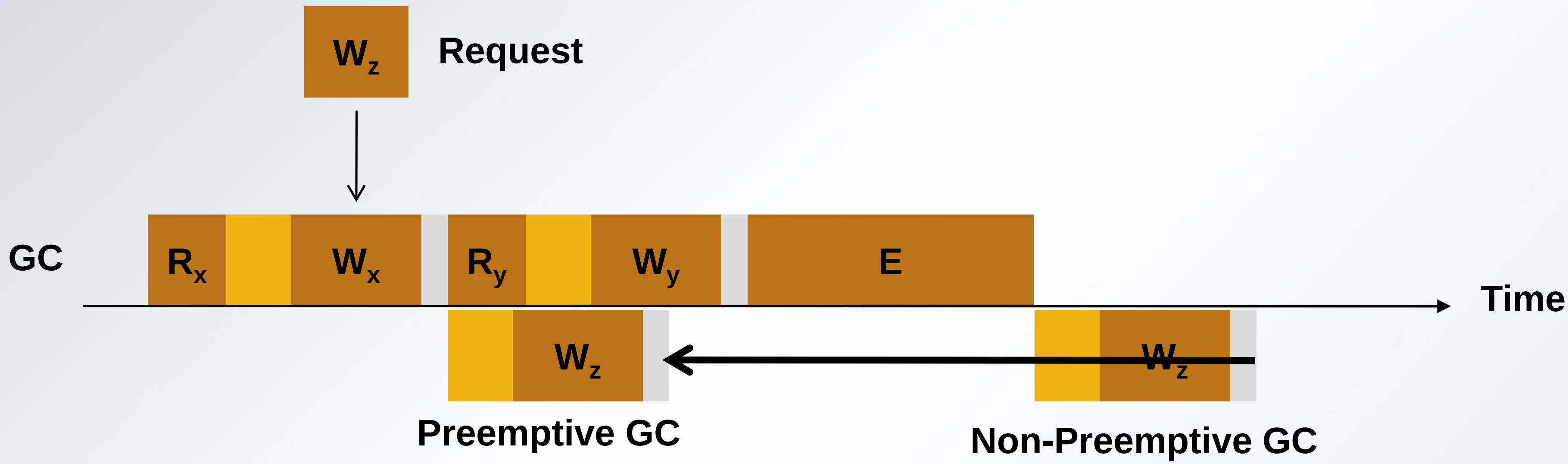
- From the empirical study, we learned:
  - Performance variability increases as the percentage of writes in workloads increases.
  - Performance variability increases with respect to the arrival rate of write requests.

- This is because:
  - Any incoming requests during the GC should wait until the on-going GC ends.
  - ***GC is not preemptive***

Outline

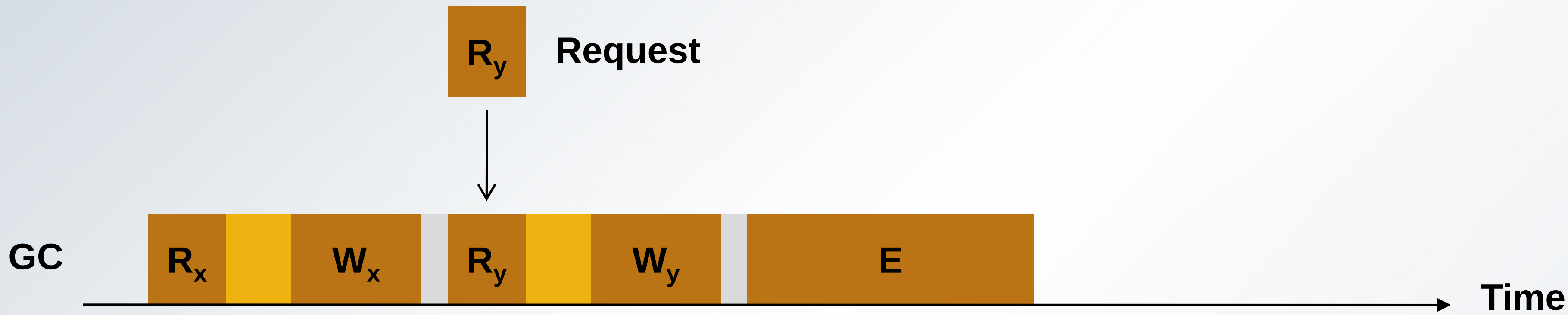
- Introduction
- Background and Motivation
- Semi-Preemptive Garbage Collection
  - Semi-Preemption
  - Further Optimization
  - Level of Allowed Preemption
- Evaluation
- Conclusion

Technique #1: Semi-Preemption

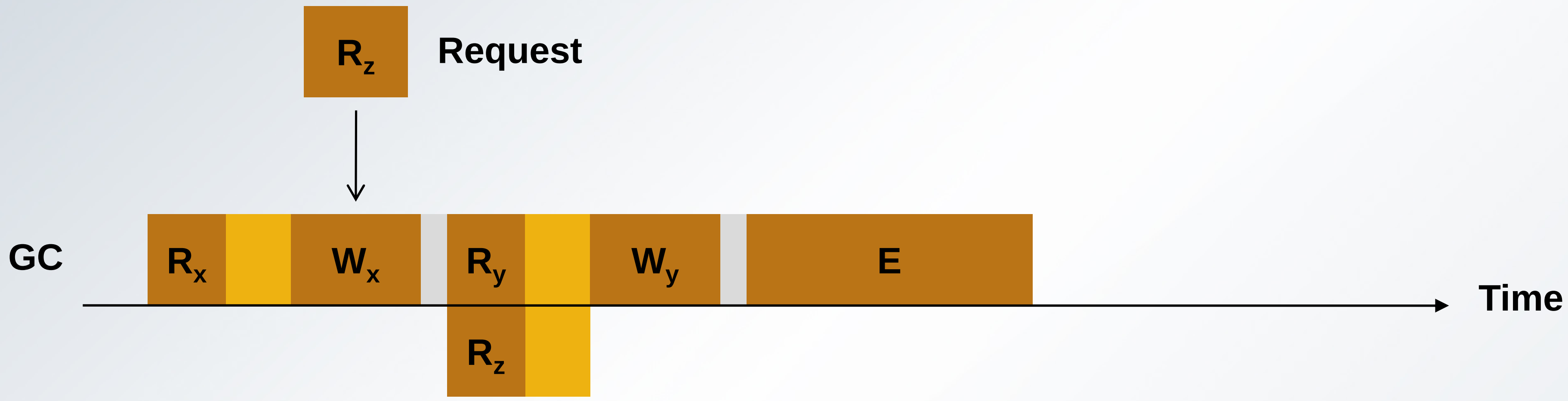


- R_x: Read page x
- W_x: Write page x
- E: Erase a block
- Data transfer
- Meta data update

Technique #2: Merge



Technique #3: Pipeline



Level of Allowed Preemption

- Drawback of PGC
  : The completion time of GC is delayed
  → May incur lack of free blocks
  → Sometimes need to prohibit preemption
- States of PGC

	Garbage collection	Read requests	Write requests
State 0	X		
State 1	O	O	O
State 2	O	O	X
State 3	O	X	X

Outline

- Introduction
- Background and Motivation
- Semi-Preemptive Garbage Collection
- Evaluation
  - Setup
  - Synthetic Workloads
  - Realistic Workloads
- Conclusion

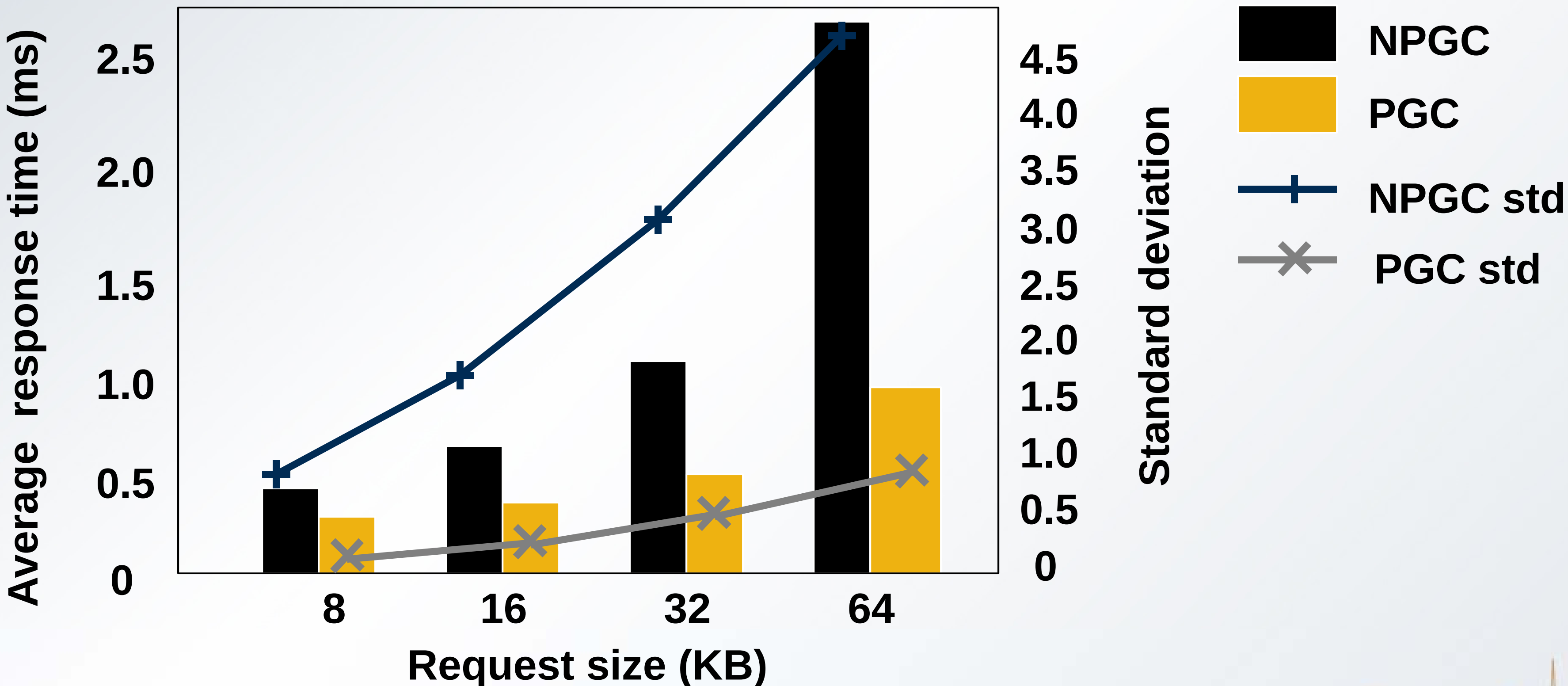
Setup

- Simulator
  - MSR's SSD simulator based on DiskSim
- Workloads
  - Synthetic workloads
    - Used the synthetic workload generator in DiskSim
  - Realistic workloads

	Workloads	Average request size (KB)	Read ratio (%)	Arrival rate (IOP/s)
Write dominant	Financial	7.09	18.92	47.19
	Cello	7.06	19.63	74.24
Read dominant	TPC-H	31.62	91.80	172.73
	OpenMail	9.49	63.30	846.62

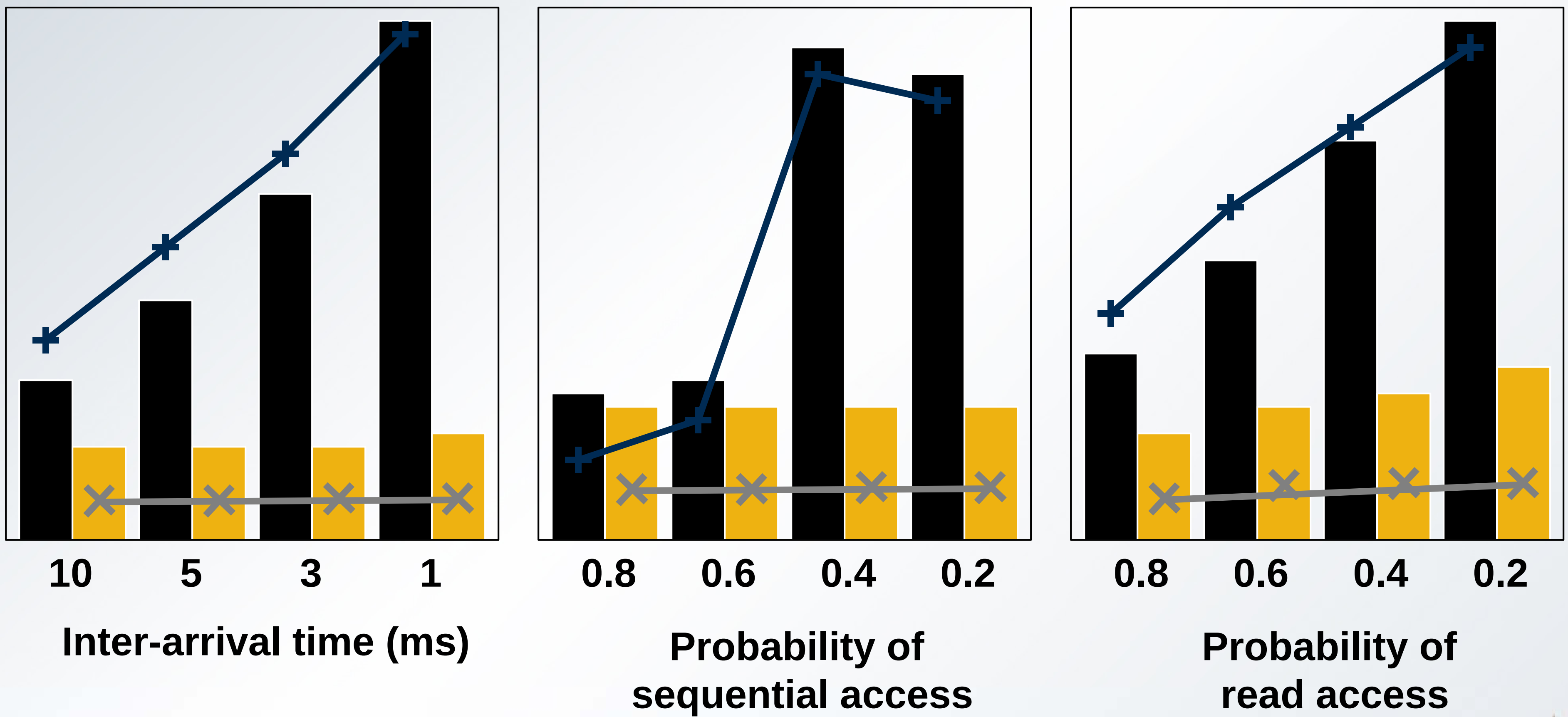
Performance Improvements for Synthetic Workloads

- Varied four parameters: request size, inter-arrival time, sequentiality and read/write ratio
- Varied one at a time fixing others



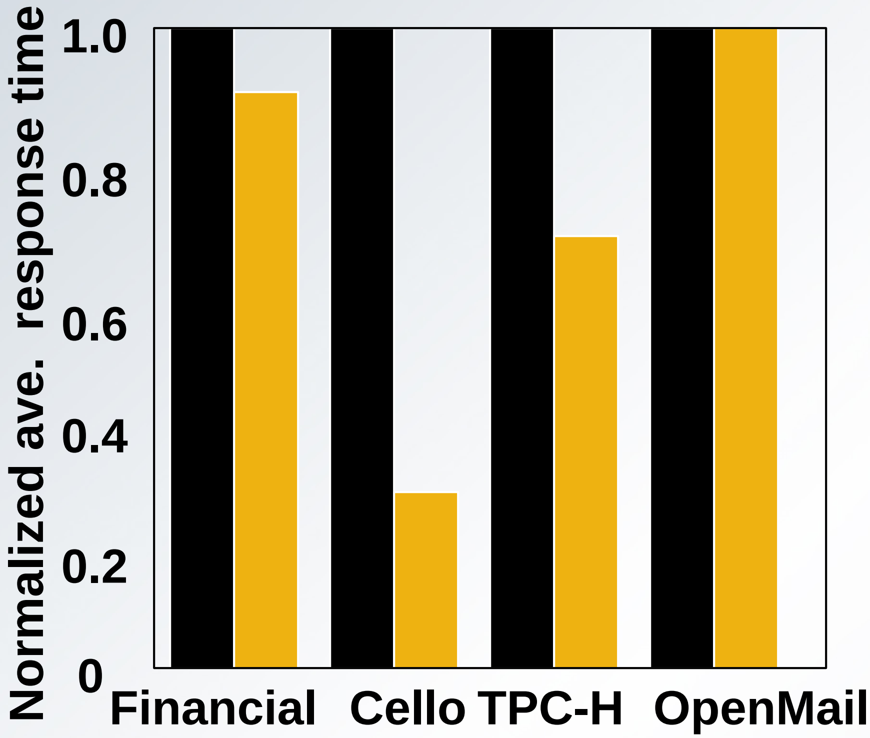
Performance Improvement for Synthetic Workloads (con't)

Bursty | Random dominant | Write dominant



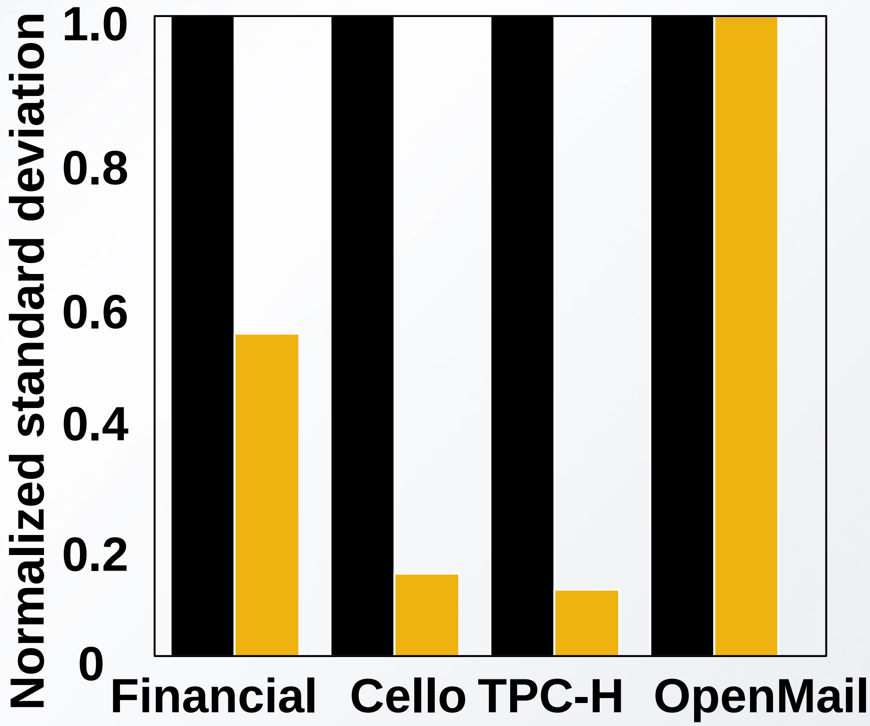
Performance Improvement for Realistic Workloads

- Average Response Time



Improvement of average response time by 6.5% and 66.6% for Financial and Cello.

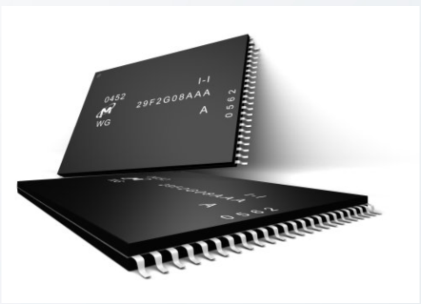
- Variance of Response Times



Improvement of variance of response time by 49.8% and 83.3% for Financial and Cello.

Conclusions

- Solid state drives
  - Fast access speed
  - Performance variation ← garbage collection
- Semi-preemptive garbage collection
  - Service incoming requests during GC
- Average response time and performance variation are reduced by up to 66.6% and 83.3%



Questions?

Contact info

Junghee Lee
junghee.lee@gatech.edu
Electrical and Computer Engineering
Georgia Institute of Technology

Thank you!