29th IEEE Conference on Mass Storage Systems and Technologies (MSST)
Long Beach, California, USA, May 6~10, 2013

VSSIM: Virtual Machine based SSD Simulator

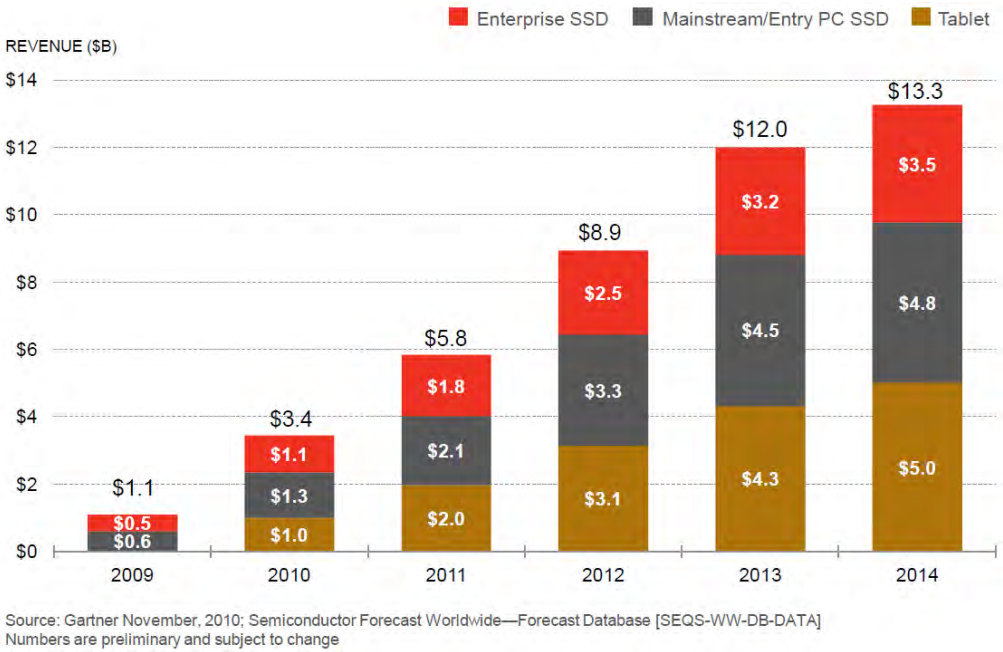
Jinsoo Yoo, Youjip Won, Joongwoo Hwang, Sooyong Kang,
Jongmoo Choi, Sungroh Yoon and Jaehyuk Cha

Dept. of Electronics and Computer Engineering
Hanyang University

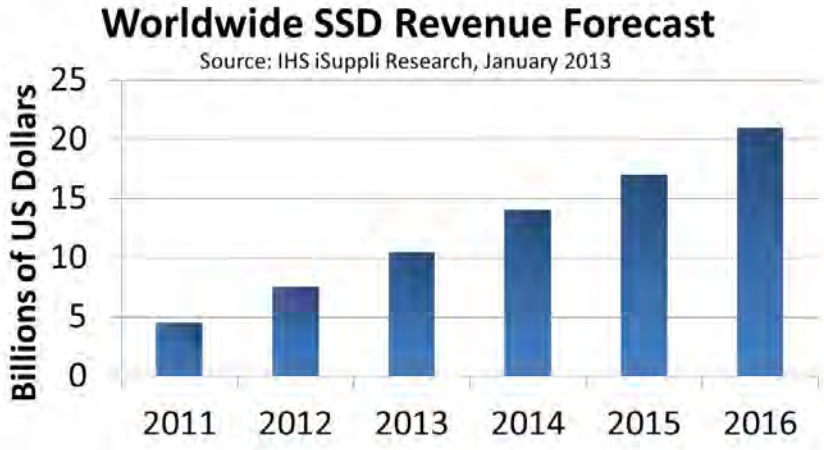
Contents

- Introduction
- VSSIM(Virtual Machine based SSD Simulator)
  - The architecture of VSSIM
  - Algorithm
  - Validation
- Case Studies
- Conclusion
- Demo

SSD Market is Expanding

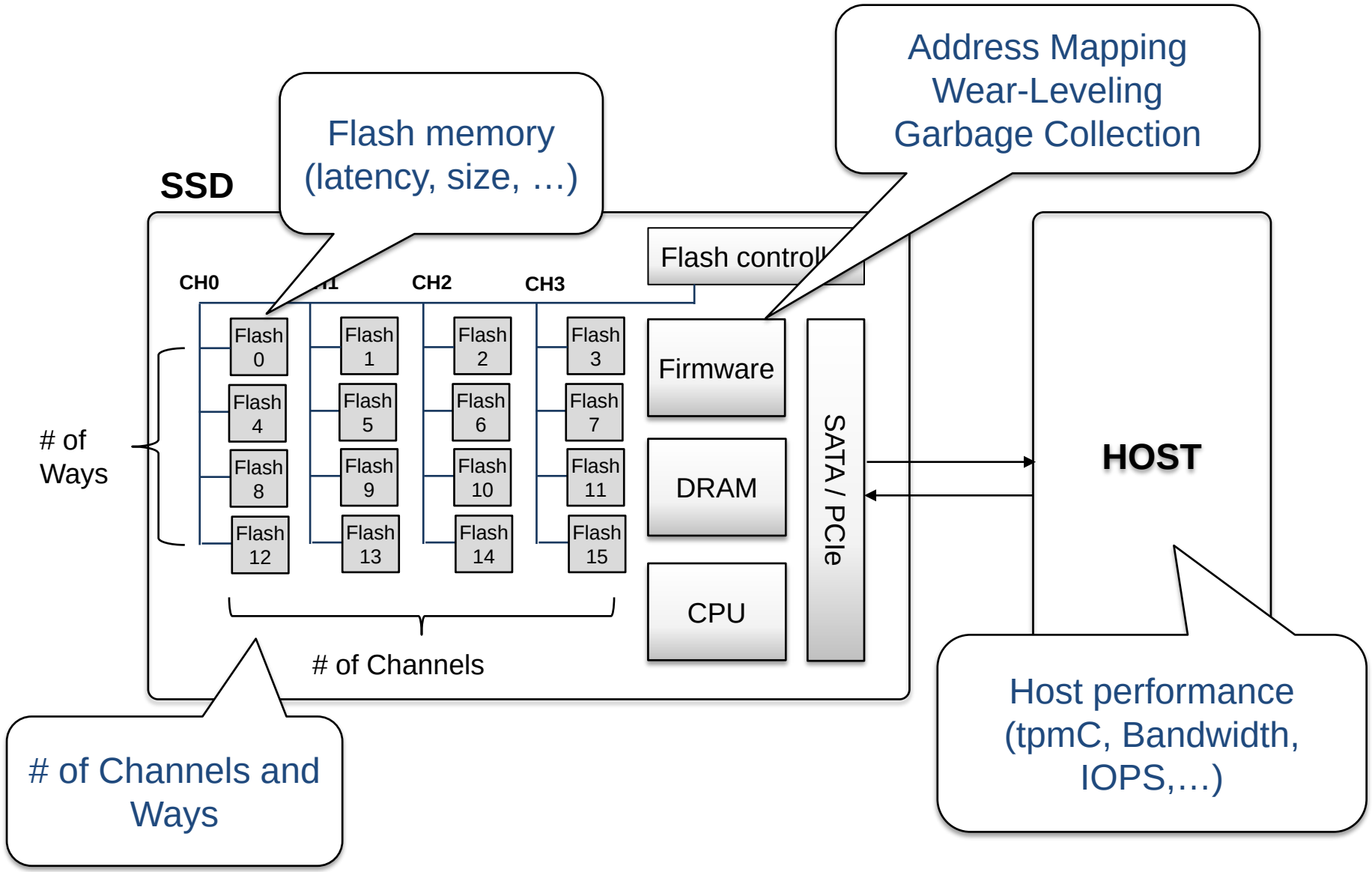


Source: Gartner November, 2010; Semiconductor Forecast Worldwide—Forecast Database [SEQS-WW-DB-DATA]
Numbers are preliminary and subject to change



It is important to **design an SSD Component** for future SSD performance requirements.

Design Components of SSD

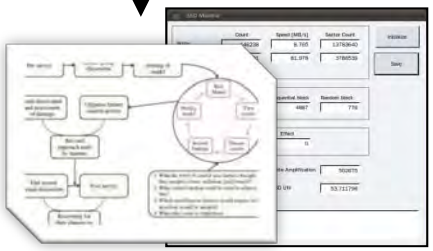


SSD Design

SSD Design of S/W & H/W

- # of Channel / Way
- Mapping Algorithm
- Garbage Collection
- Wear-leveling

⋮



H/W Prototyping

Pros.

- Accurate

Cons.

- Time Consuming
- Expensive
- Inflexible

S/W based Simulation

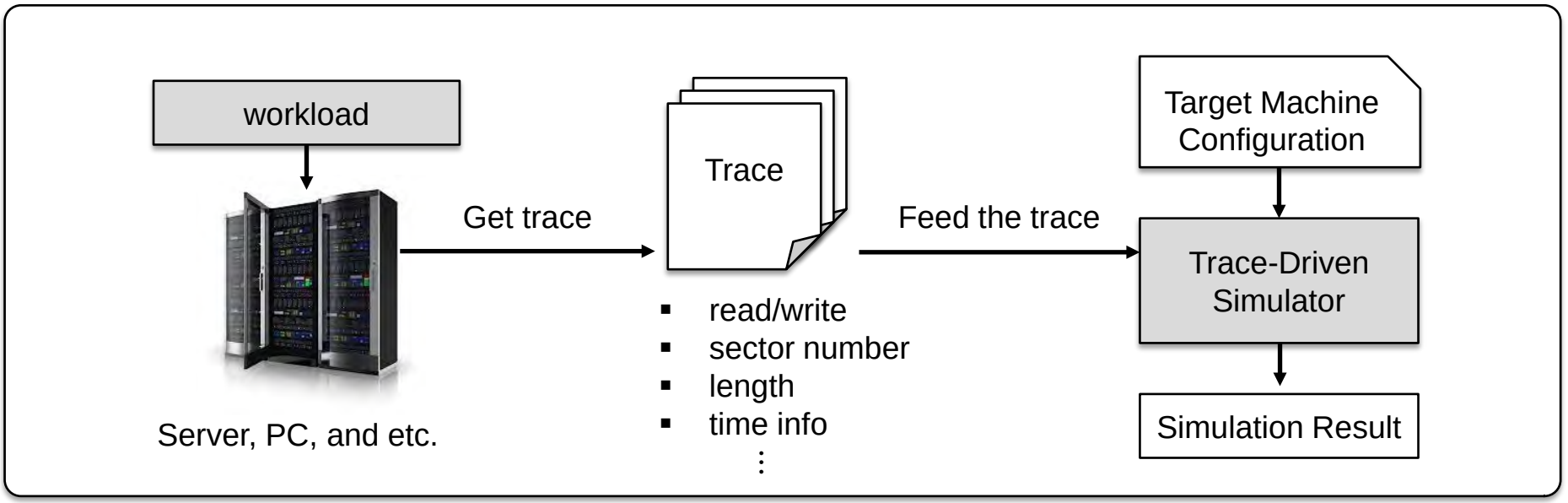
Pros.

- Time saving
- Cost efficient
- Flexible

Cons.

- Less accurate

Trace-driven Simulator

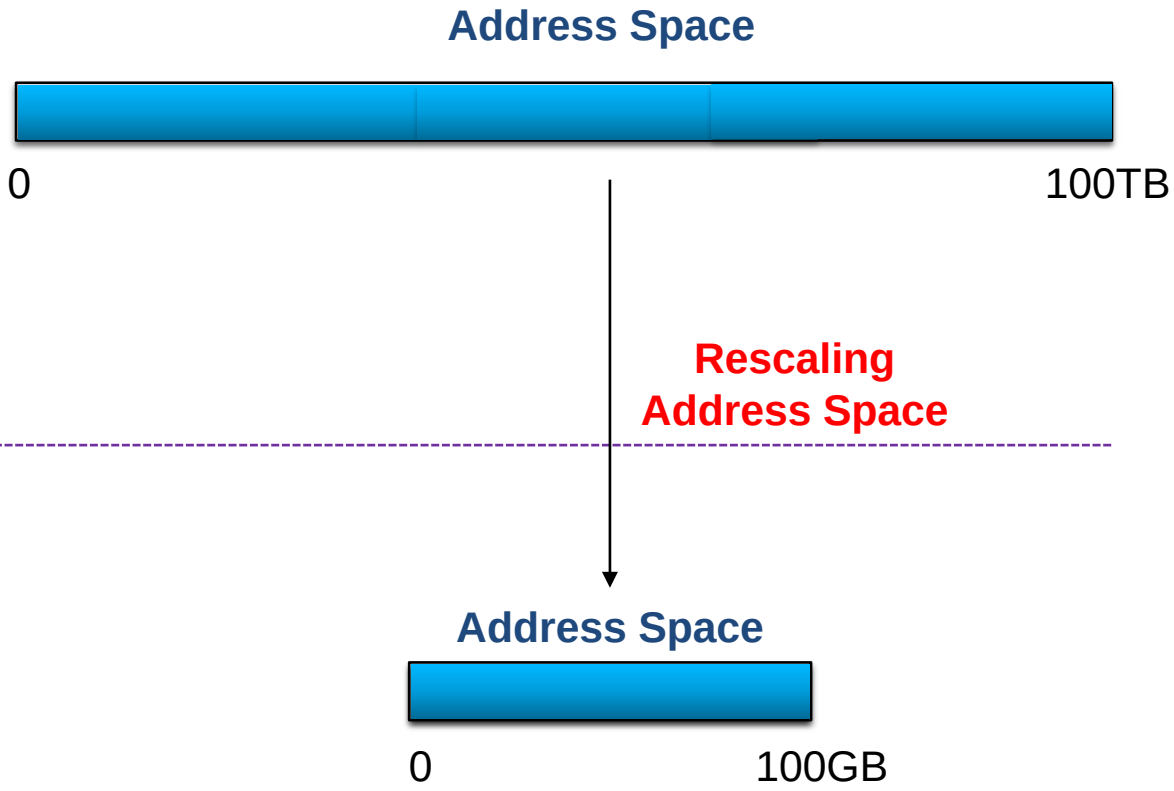


- DISKSIM for SSD [Agrwawal08]
- FLASHSIM [Kim09]
- CPS-SIM [Lee09]

Issues in Trace-driven Simulator: Address Space Rescaling

Server, Mass Storage, etc.

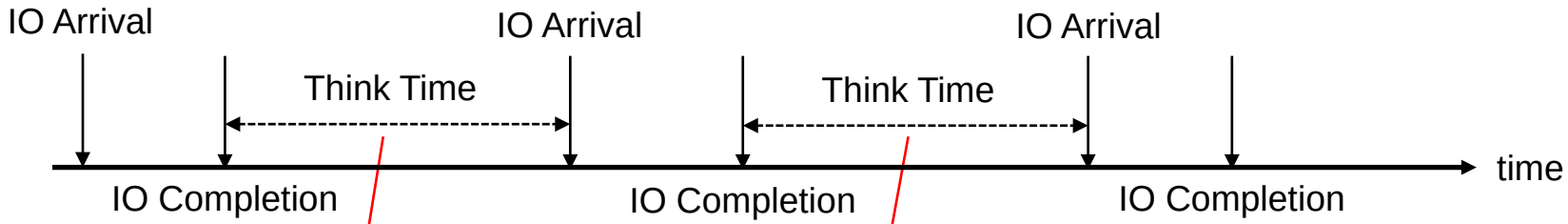
Simulation target



Disintegrates the access locality in the trace

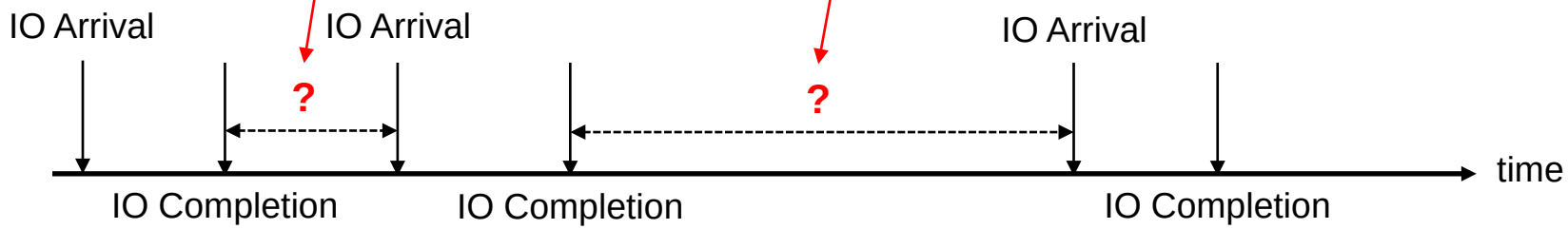
Issues in Trace-driven Simulator: Think Time Rescaling

Original Trace



Think Time Rescaling for closed loop system

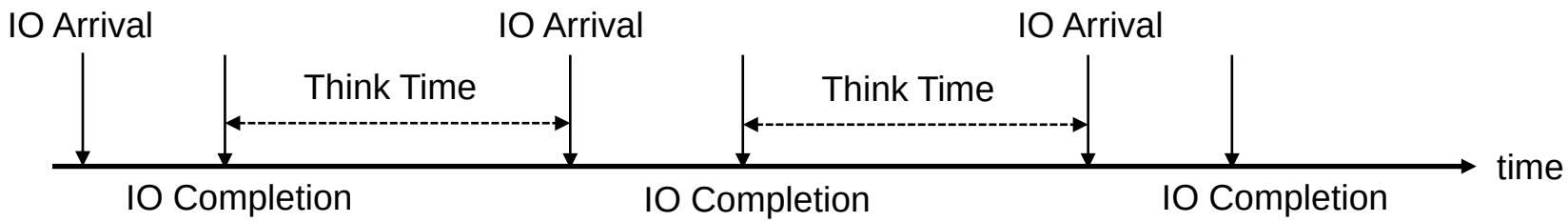
Simulation Target



- //Trace [Mesnier07]
- Buttress [Anderson04]

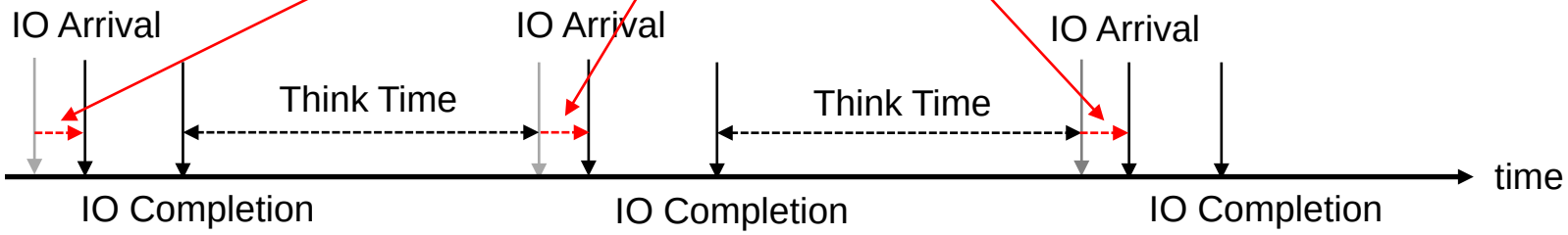
Issues in Trace-driven Simulator: Accurate Replay

Original Trace

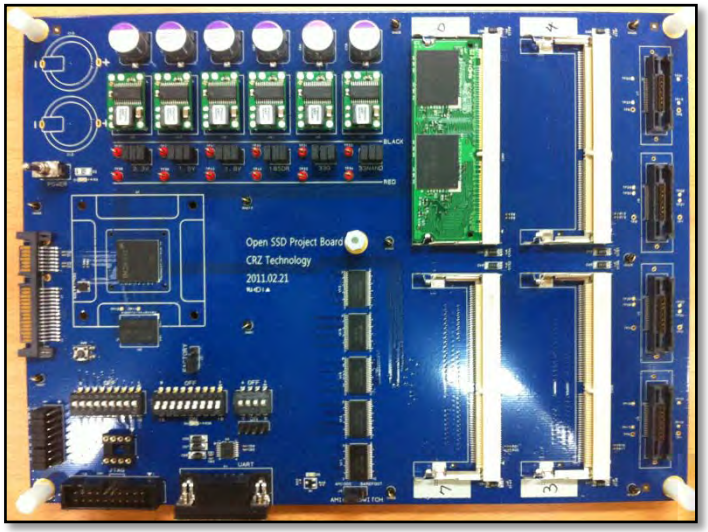


Difficulty in accurate replay

Simulation Target



Hardware-based Emulator



Hardware-based Emulator:

- BlueSSD [Lee10]
- OpenSSD [Lee11]
- FPGA-based solid-state drive prototyping Platform [Cai11]

⋮

Pros.

- Accurate test result

Cons.

Cannot Extend

- # of Flash memories
- # of Channels
- # of Ways

Cannot Change

- NAND IO latency
- NAND page size

Remake is needed to use a new NAND flash memory

Inflexible

Functionality of SSD Simulators

	Type		Host Perf.	Firmware Change	# of Channels/Ways	NAND Latency
DiSKSIM [Agrawal08]	Trace-driven	Offline	No	O	O	O
NANDSIM	Kernel Code	Online	Yes	X	X	O
F						
OpenSSD [Lee11]	Hardware-base	Online	Yes	O	△	X
BlueSSD [Lee10]	Hardware-base	Online	No	O	△	X

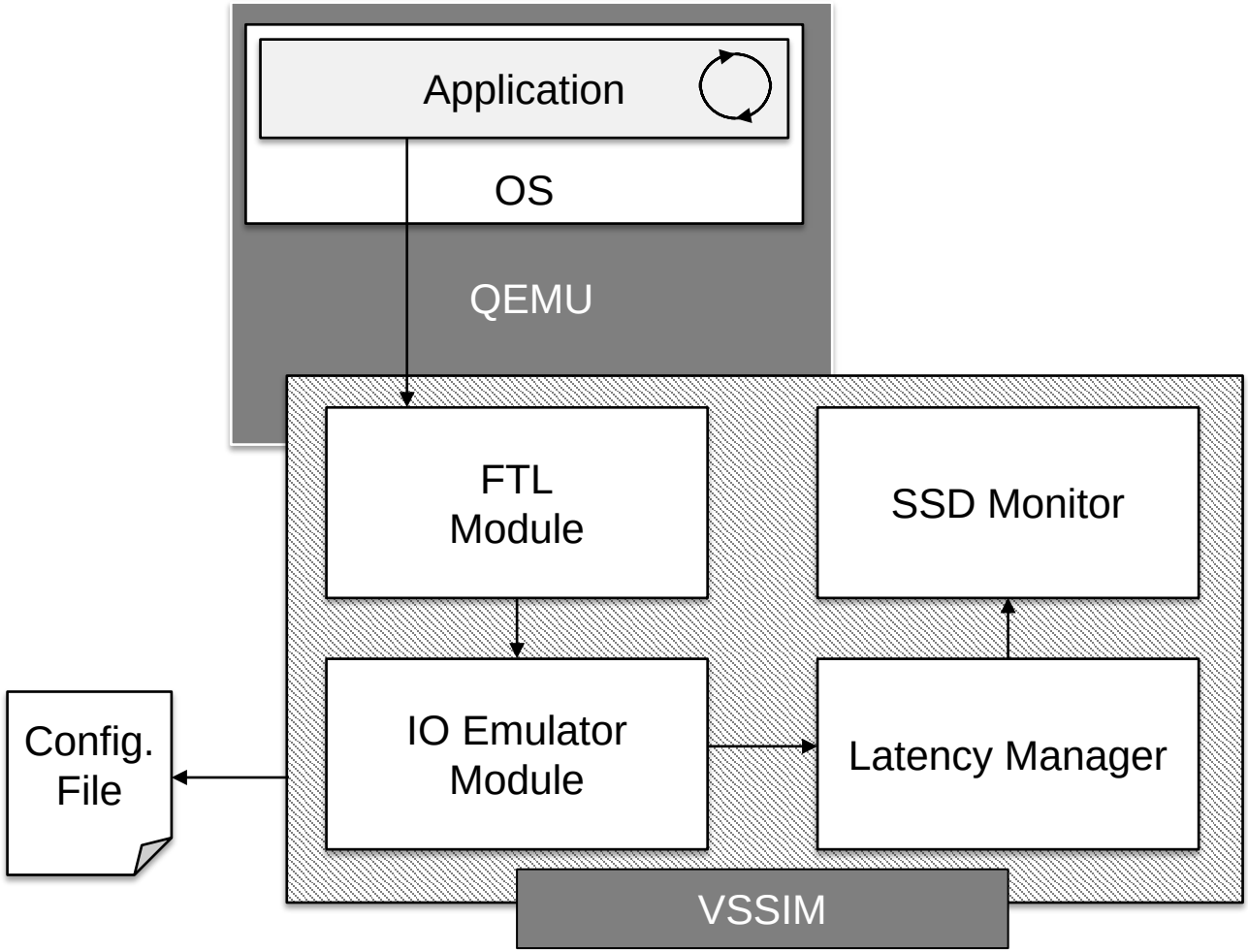
Can an SSD Simulator support **all these functionality**?

- O: Supported
- △: Supported with a H/W Limitation
- X: Not supported

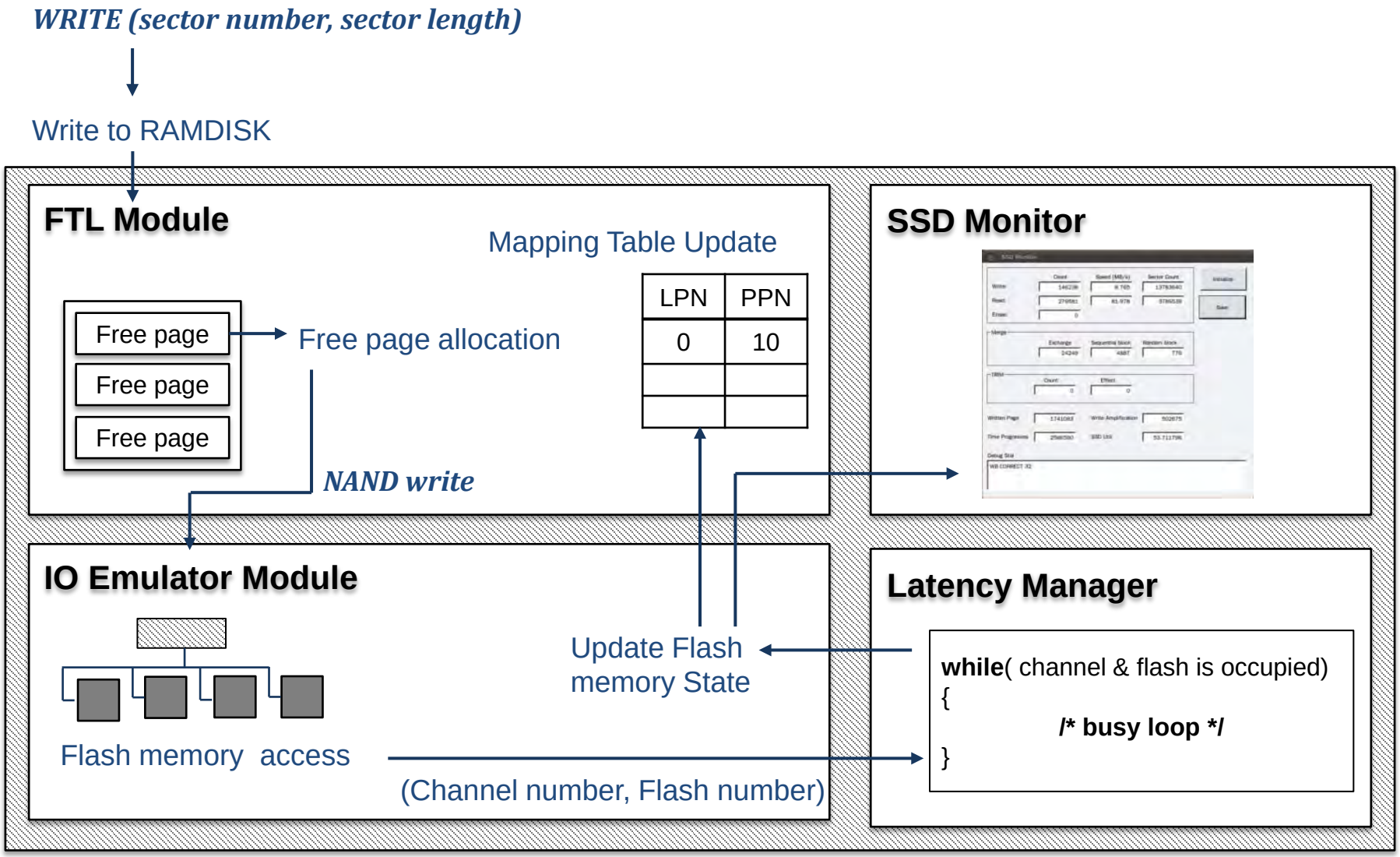
Requirements for a new SSD simulator

- Does not need to use a **trace**
- Can measure **Host** performance in real-time
- Can simulate **various SSD** architecture
  - # of channel/way
  - NAND flash memory
  - Page size
    ⋮
- Easily changes **firmware**
  - Flash Translation Layer (FTL)
  - Write Buffer / Map Cache / TRIM
    ⋮

Virtual Machine based SSD Simulator: VSSIM



VSSIM Mechanism



FTL Module

Maintains **mapping Information** and **block status**

- Maintain logical to physical **address translation.**
- Garbage Collection
- Weal-leveling
- **Issue NAND read/write request** to IO Emulator Module.

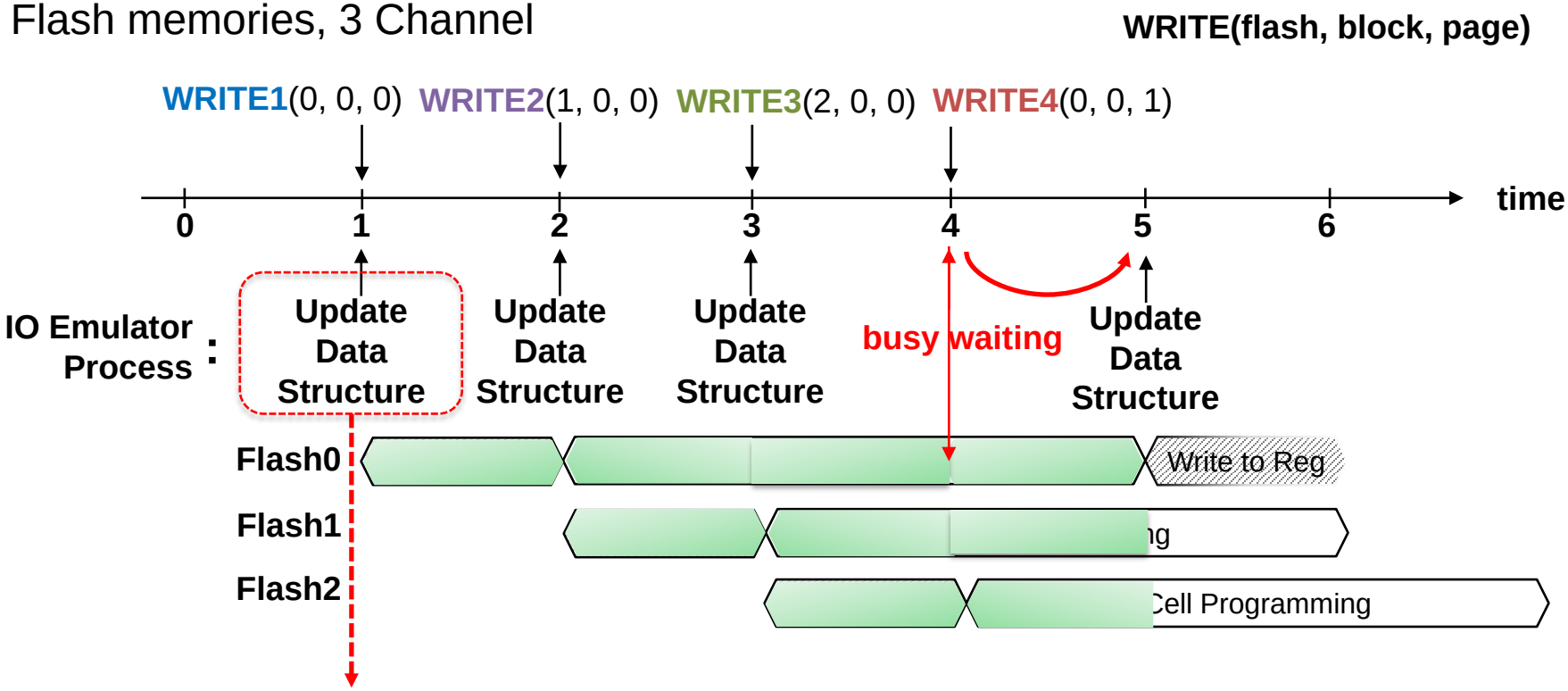
IO Emulator Module

Emulate NAND IO operation

- Perform NAND read/write/erase operation.
- **Introduces** the **appropriate amount of latency** with latency manager.
- Support multi-channel, multi-way operation **using single thread.**

IO Emulator

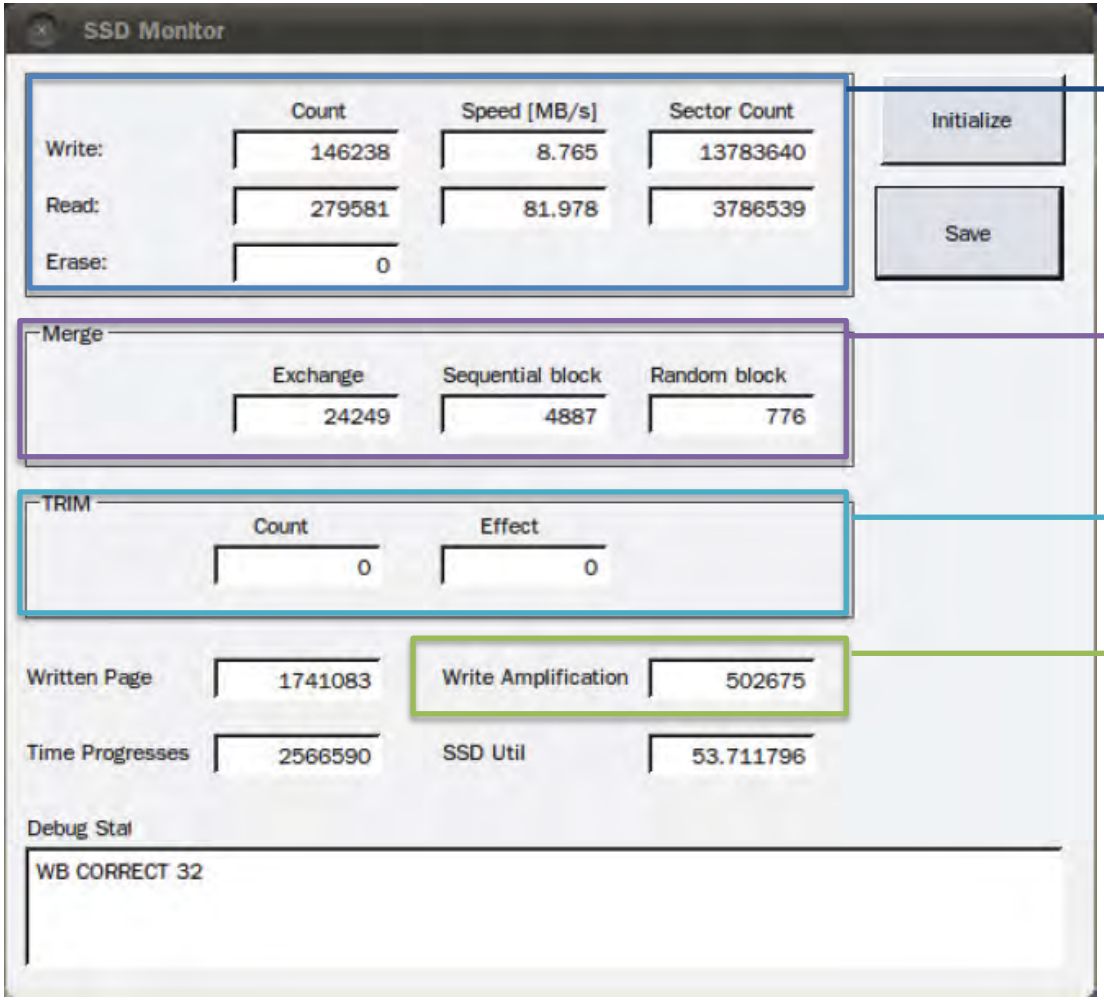
3 Flash memories, 3 Channel



IO Emulator Data Structure

	IO Type	Reg IO Time	Cell IO Time
Flash 0	Write	1	2
Flash 1	Write	2	3
Flash 2	Write	3	4

SSD Monitor: Real-time Monitoring Tool



# of Write/Read/Erase requests
- IO Bandwidth (MB/s)
- Total # of sectors written/read

# of Merge Operations

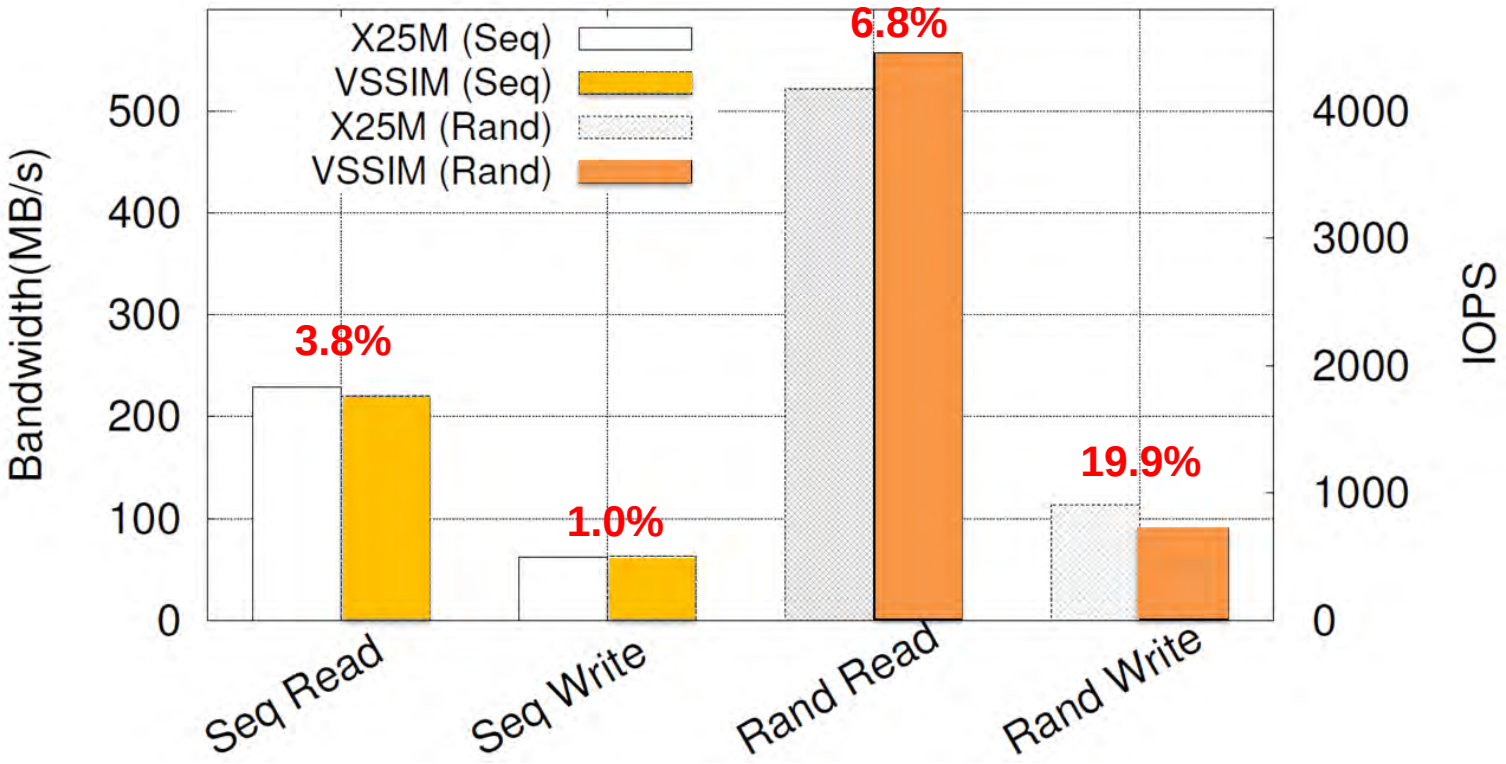
# of TRIM Commands

Write Amplification

VSSIM Validation

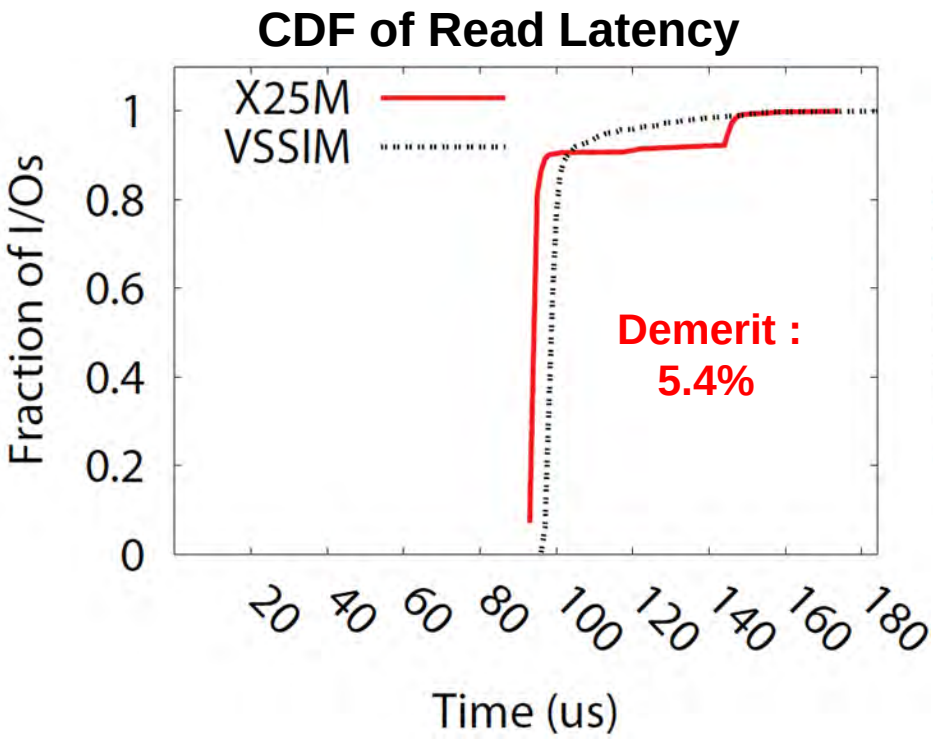
Intel X25M vs. VSSIM (configured as X25M)

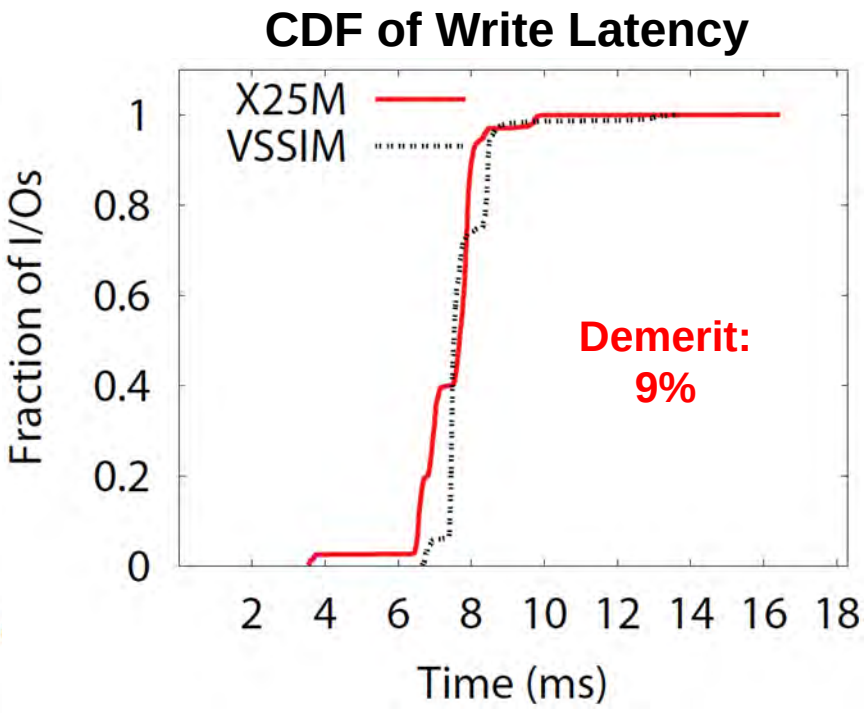
- 10 channel, 2 way, 4KByte page size



- Seq Read / Write : 512MB File size, 512KB record size
- Rand Read / Write : 512MB File size, 4KB record size

VSSIM validation with X25M using Demerit





- 512MB File size, 512KB record size, Sequential Read/Write

Strength of VSSIM

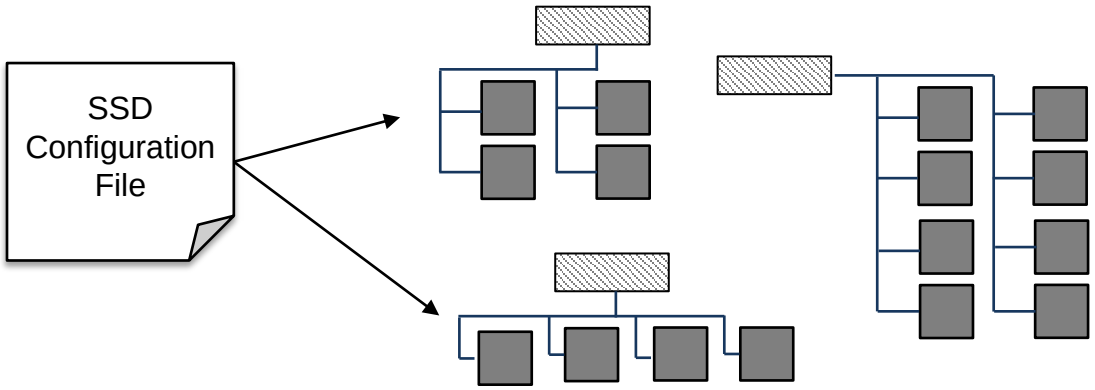
Real-time Execution

- Do not need trace
- Can measure Host performance
- Display SSD behavior in Real-time

Modularize

- Easily change or fix firmware (Mapping Algorithm, GC, W/L, and etc.)

Can simulate various SSD architecture

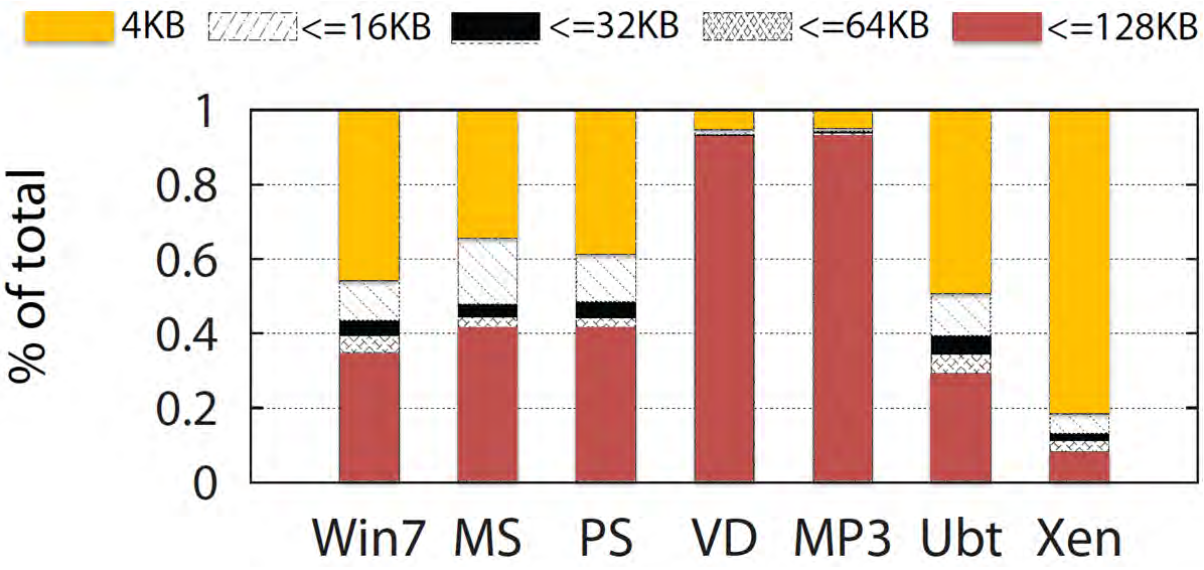


CASE STUDIES

SSD Specification

SSD Label	A	B	C	D	E
Channel	8	4	2	4	2
Way	1	2	4	1	2
# of Flash	8 EA			4EA	
Flash size	2 Gbyte			4 Gbyte	
Page size	2 Kbyte			4 Kbyte	
Read	60 usec			50 usec	
Program	800 usec			900 usec	
Erase	1.5 msec			2 msec	

Workload Characteristics

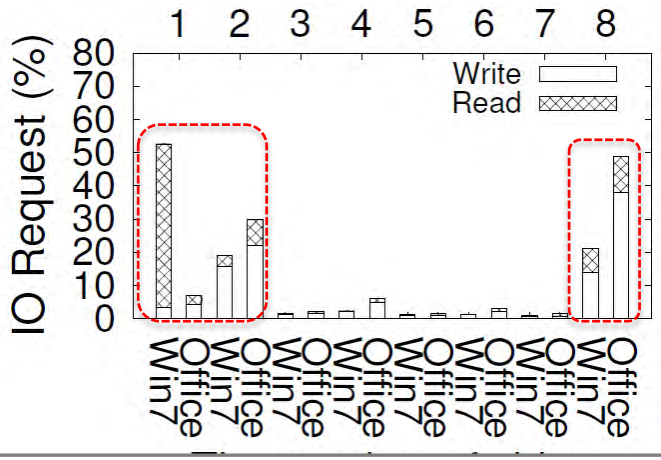


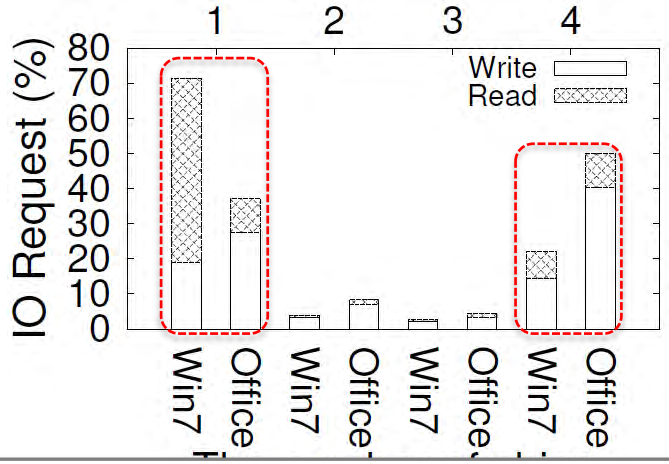
IO Size Distribution (Write, # of Cmd)

- Win7: Windows7 Installation
- MS : MS Office Installation
- PS : Photoshop Installation
- VD : Video File Copy
- MP3 : 100 MP3 File Copy
- Ubt : Ubuntu 10.04 Installation
- Xen : Xen Compile.

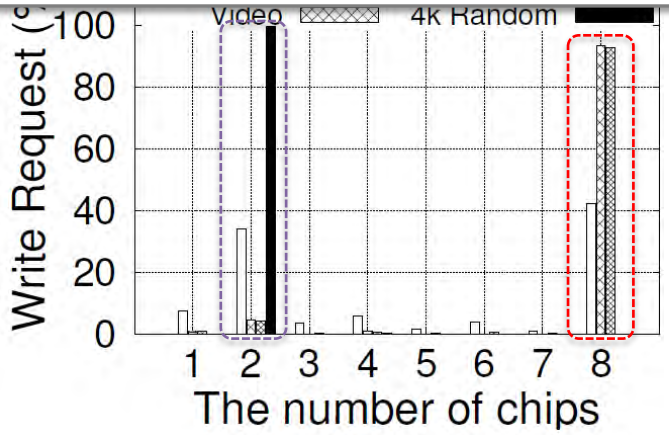
In all workloads, **76%** of write operations are either less than **4KB** or larger than **64KB.**

BI-modality of IO size distribution

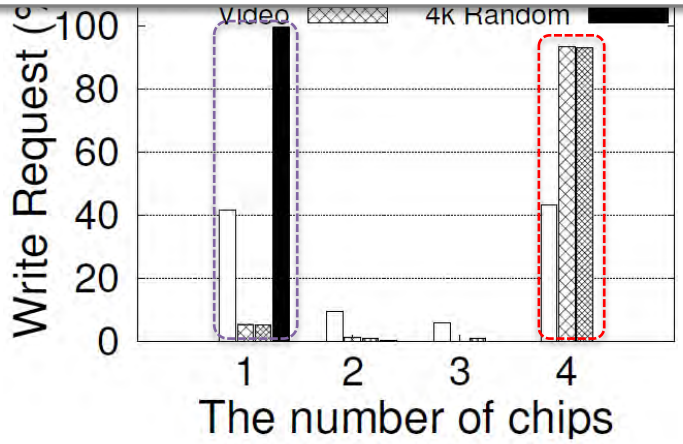




FTL should be designed to handle
multiple mapping granularity.

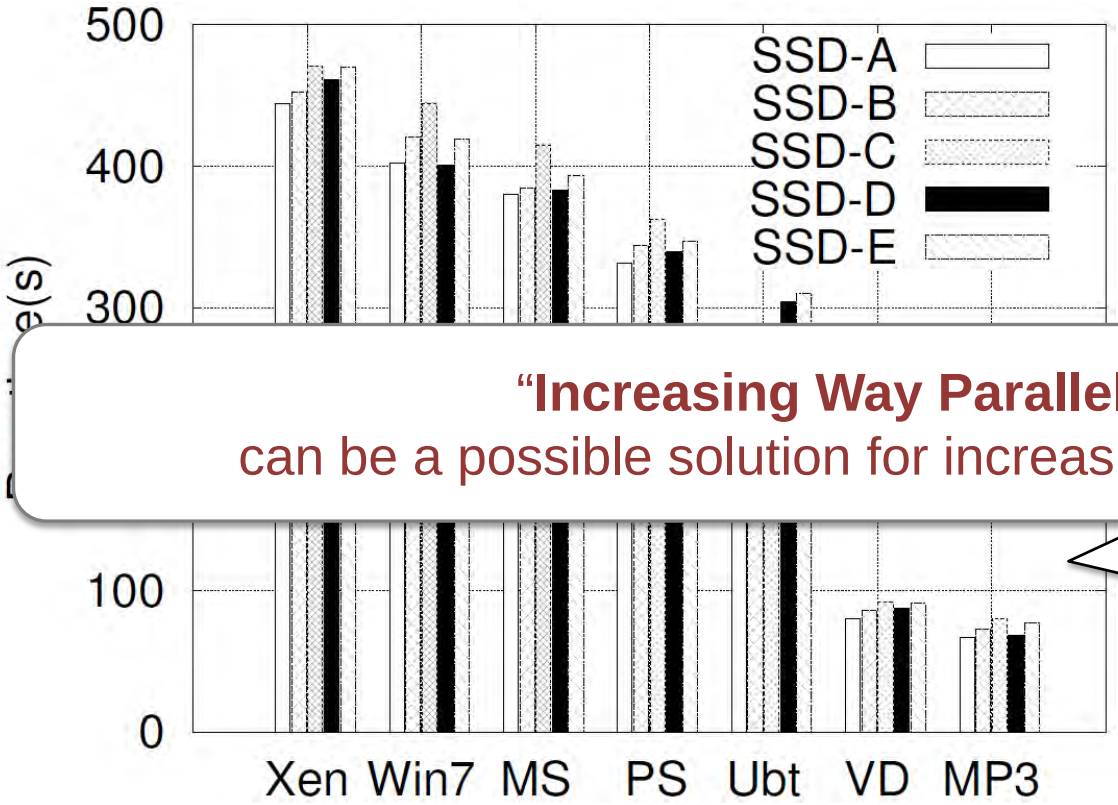


(c) 8 ch: Write only



(d) 4 ch: Write Only

SSD performance and # of channels



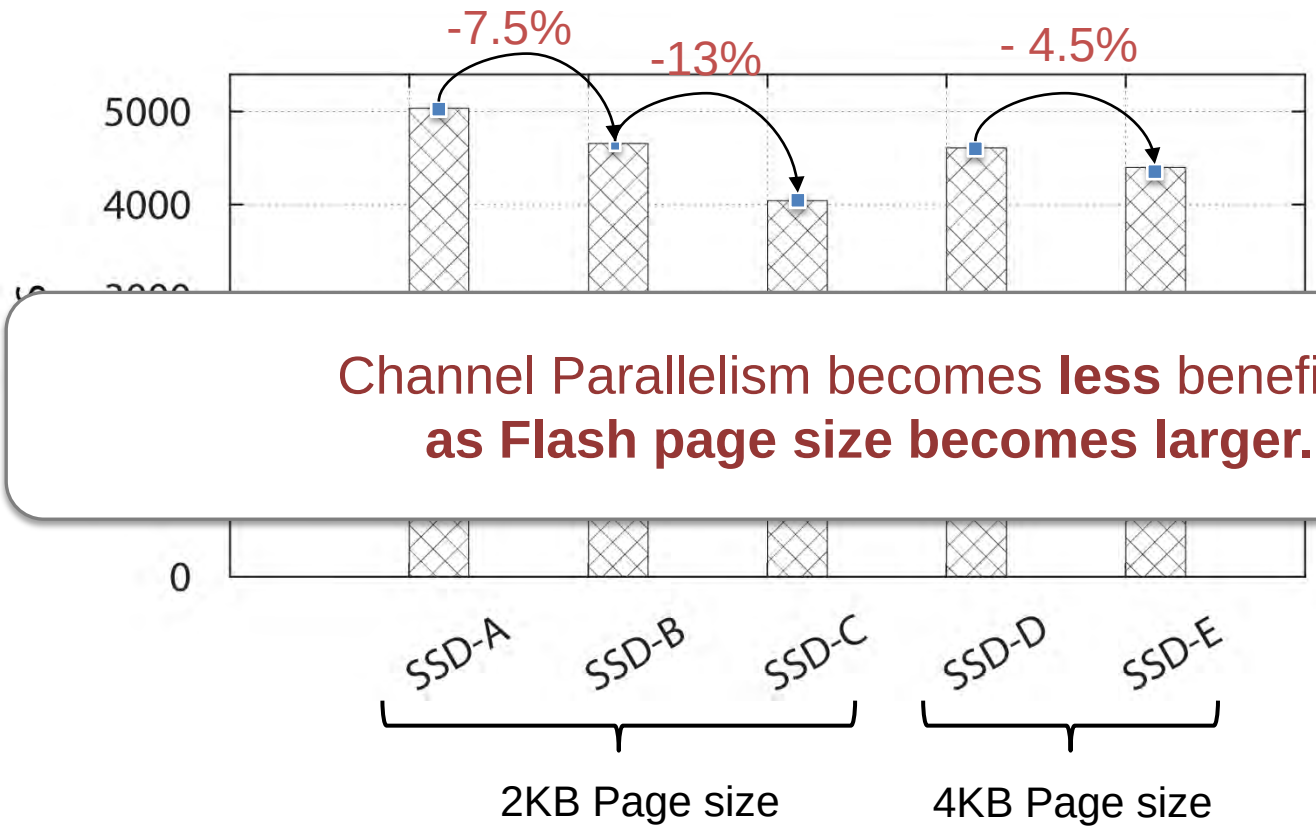
- SSD-A : **8** Ch **1** Way
- SSD-B : **4** Ch **2** Way
- SSD-C : **2** Ch **4** Way
- SSD-D : **4** Ch **1** Way

"**Increasing Way Parallelism**"
can be a possible solution for increasing performance.

As # of **ways** is doubled and # of **channel**s is reduced, the performance decreases by 4~8%

Page Size and Channel Parallelism

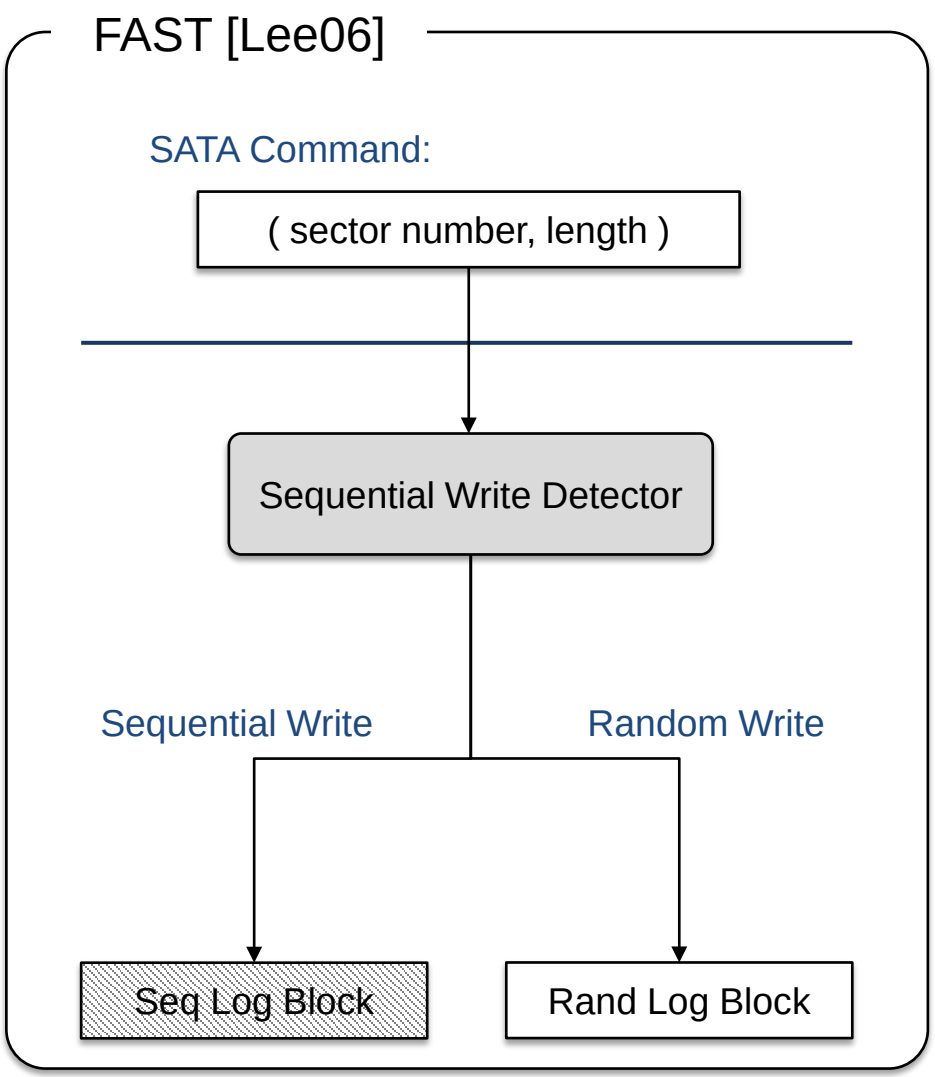
- 512MB File size, 4KB record size, Random Write (IOZONE)

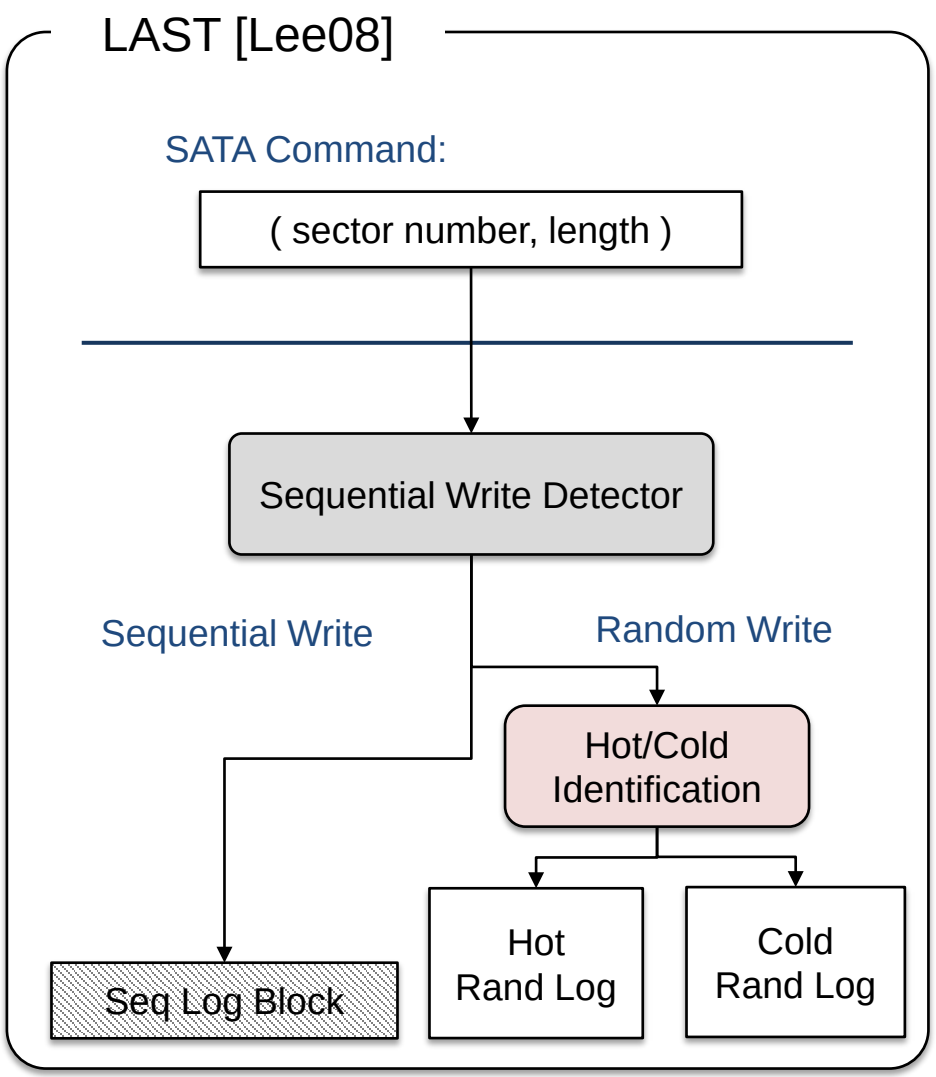


- SSD-A : **8** Ch **1** Way
- SSD-B : **4** Ch **2** Way
- SSD-C : **2** Ch **4** Way

Channel Parallelism becomes **less** beneficial
as Flash page size becomes larger.

The Basics of Hybrid Mapping FTL



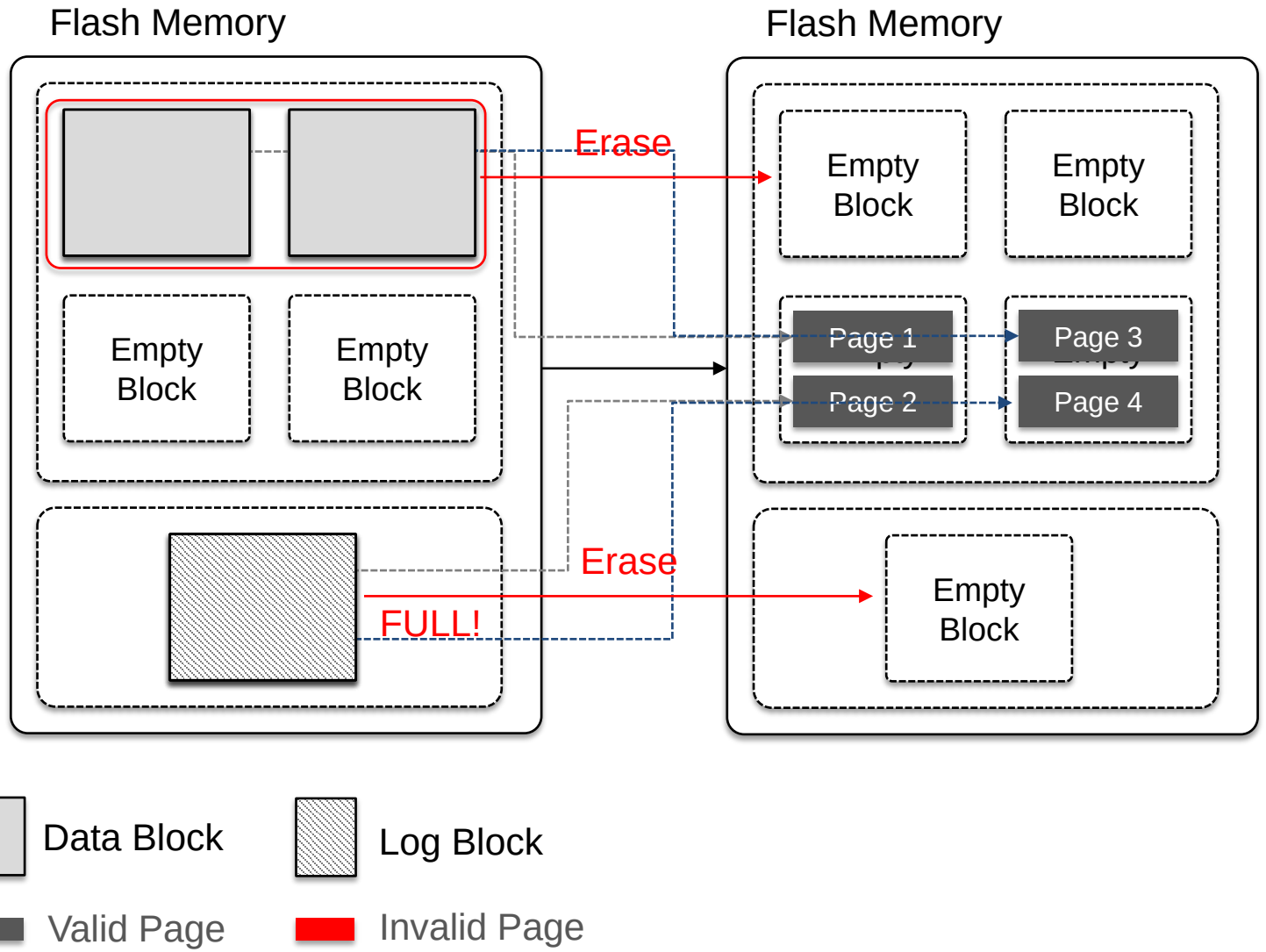


Merge Operation in Hybrid Mapping

Write Request:

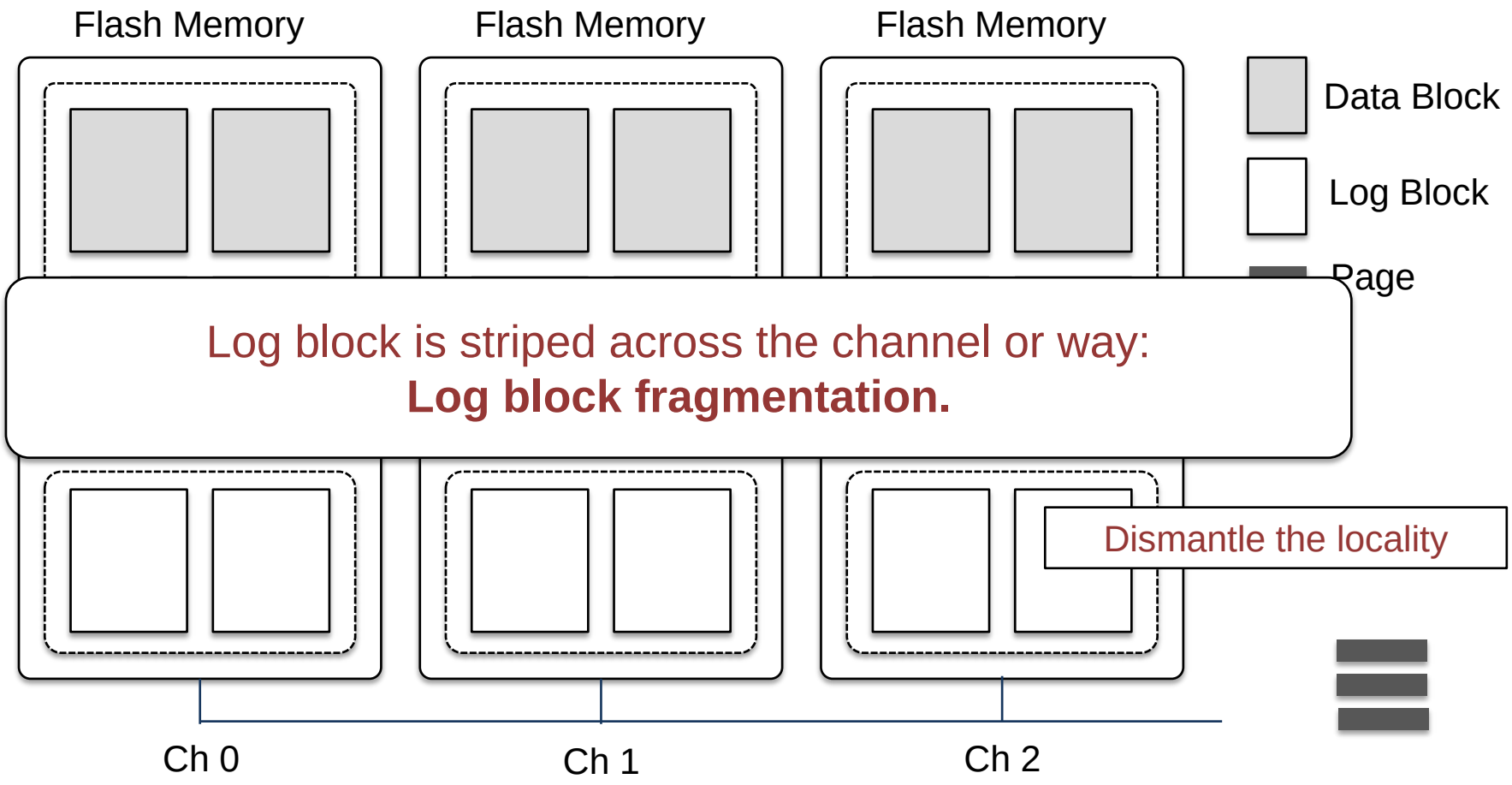
Write Page 2

Write Page 4



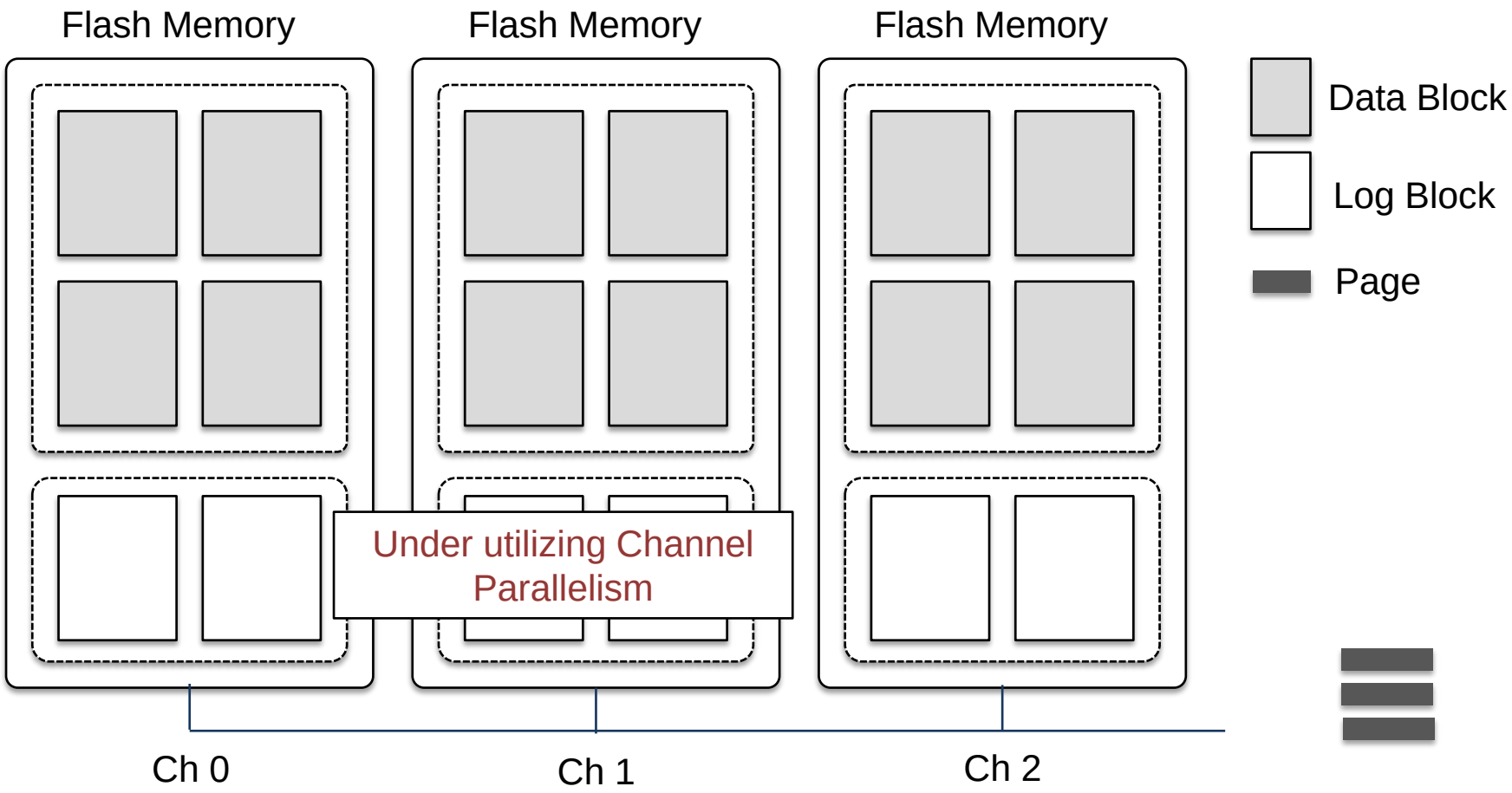
Log Block Fragmentation in Hybrid Mapping

Interleaving log block writes



Log Block Fragmentation in Hybrid Mapping

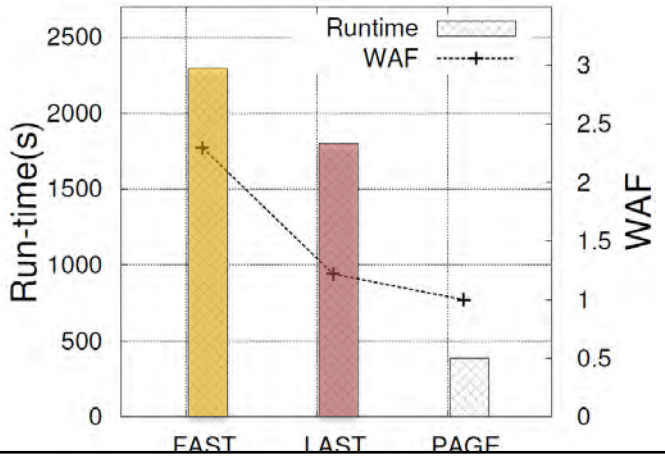
Not interleaving log block writes



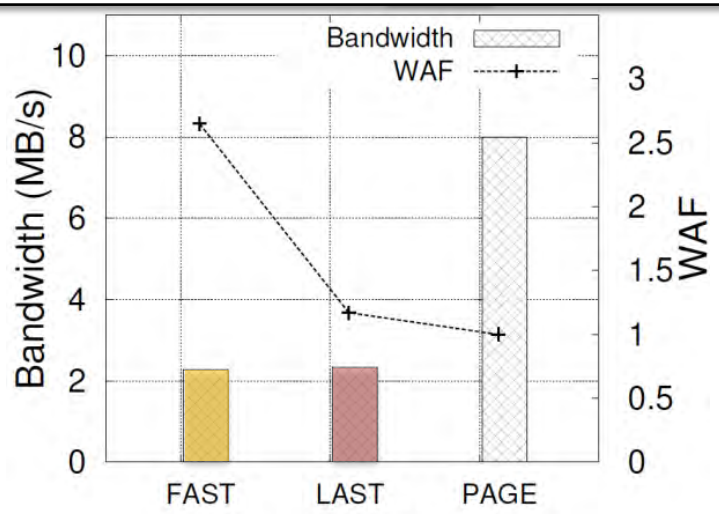
Hybrid Mapping and Multi-channel / multi-way SSD

4 Channel 1 Way SSD

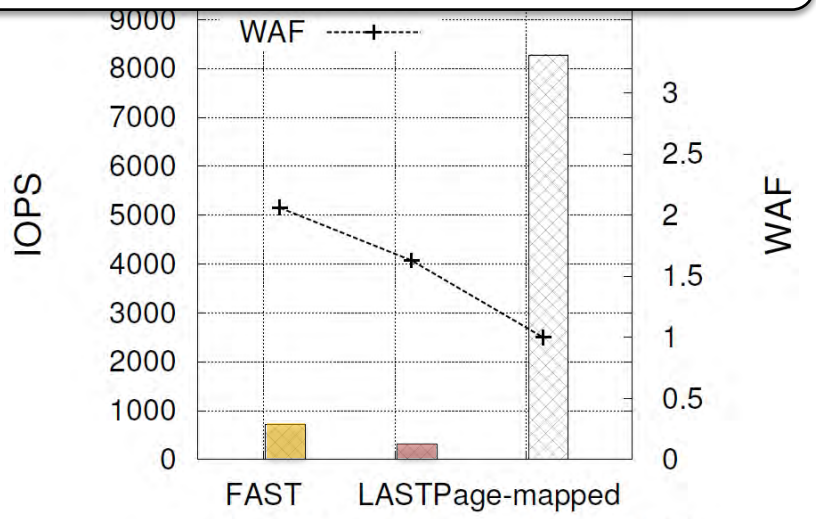
- **FAST: Interleaves only random log blocks.**
- **LAST: Interleaves only cold random blocks.**



Hybrid mapping does not fit in **Multi-ch/Multi-way** SSD.



(b) Copying 100 MP3 files



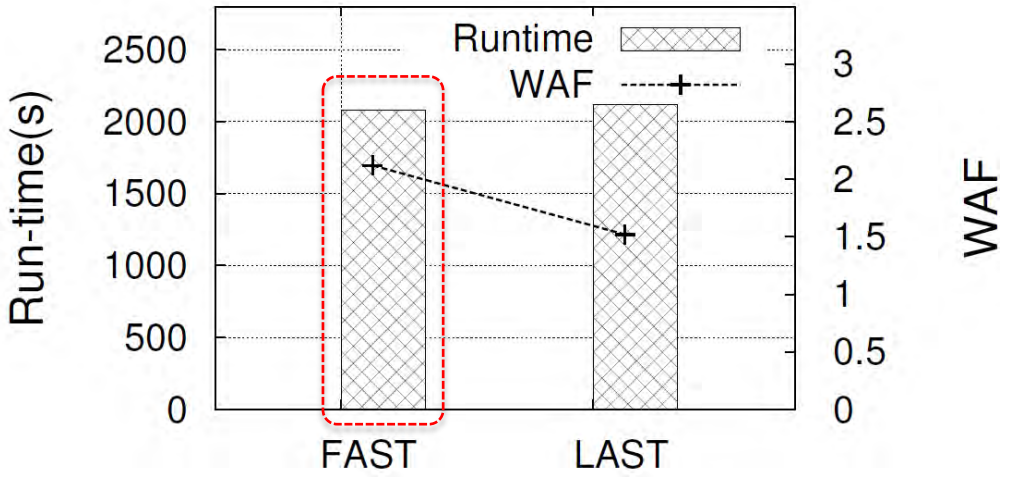
(c) Random Write (IOZONE)

Write Amplification Factor as a Performance Metric

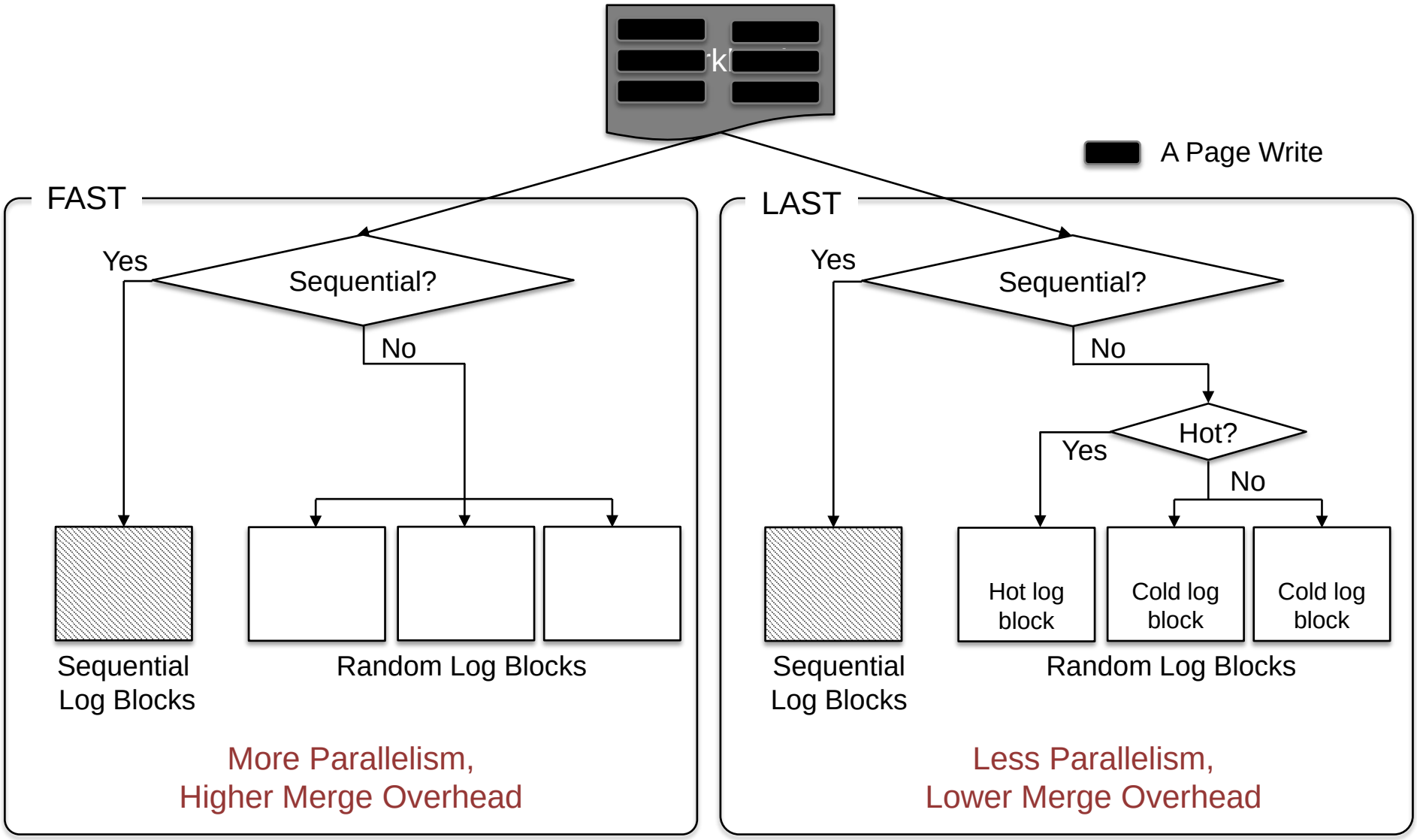
$$WAF = \frac{\text{\# of page writes which actually happen into flash memory}}{\text{\# of page writes from the host}}$$

WAF is a fair metric for SSD performance and endurance.

Well, I think the WAF may not be a right performance indicator.

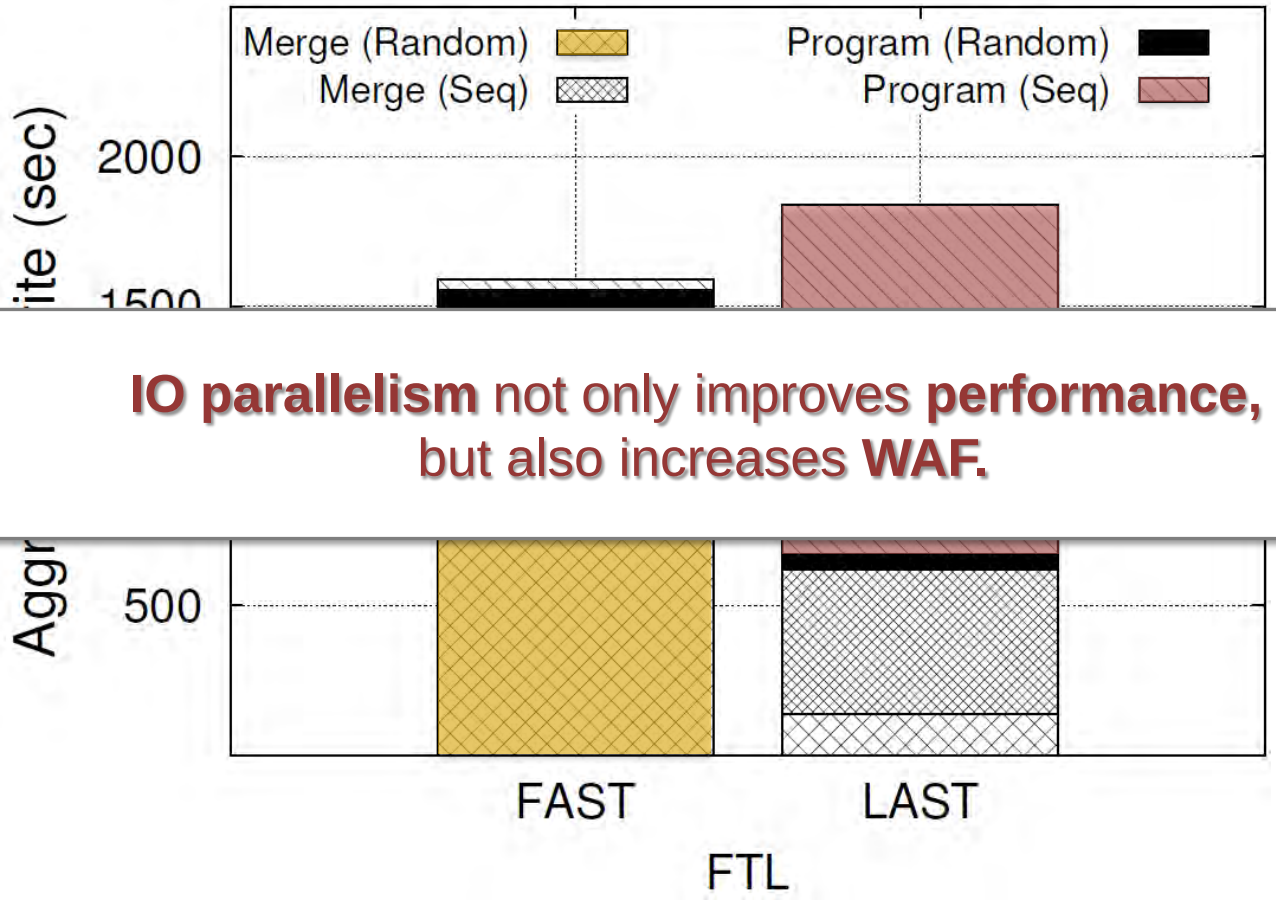


Sequentiality Detection & Performance



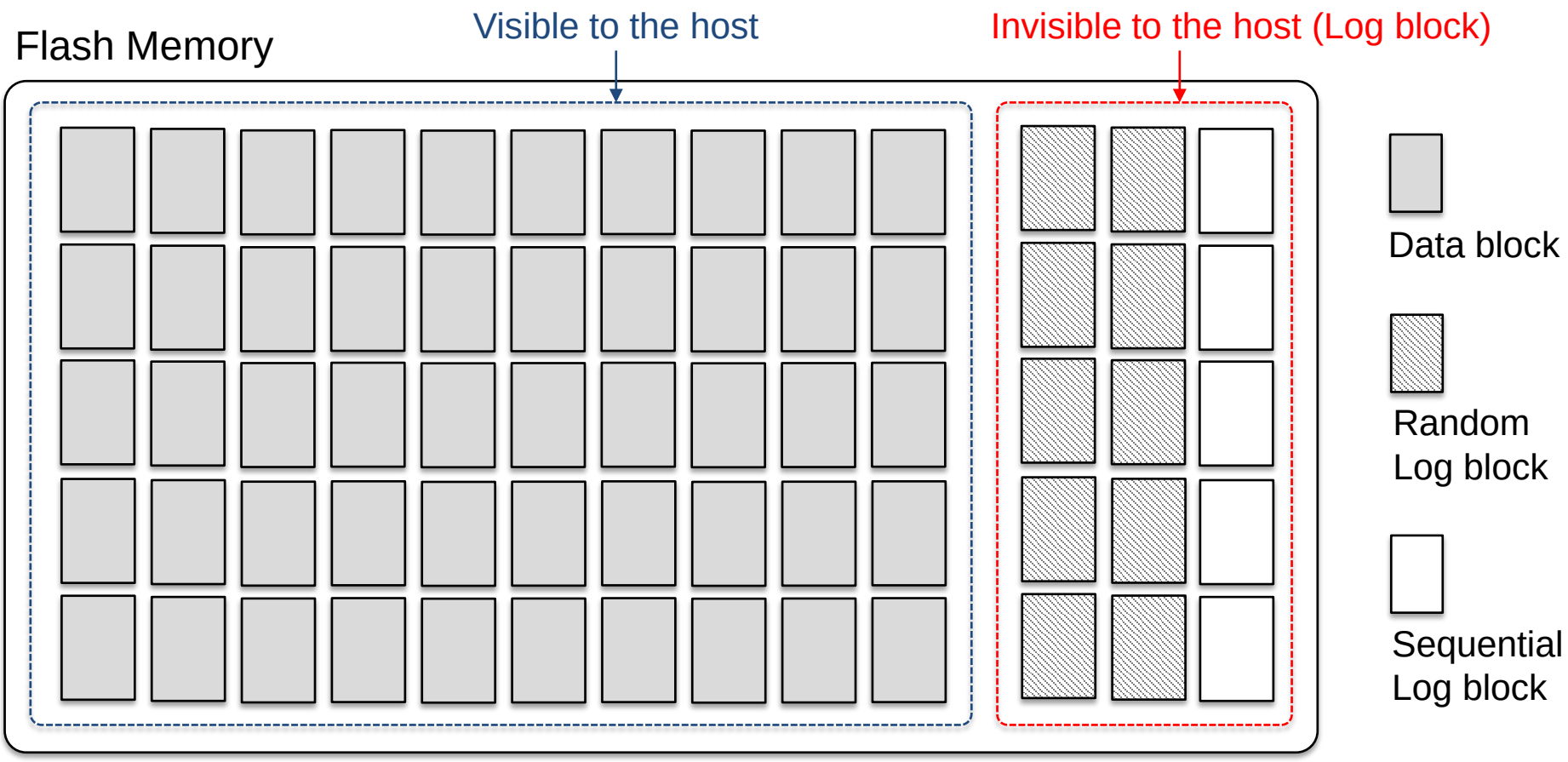
More Parallelism, Higher Merge Cost

- Windows 7 Installation
- 4 Channel 1 Way SSD
  - 64 random log block, 16 sequential log block(OP=0.5%)



IO parallelism not only improves **performance,** but also increases **WAF.**

Over-provisioning Factor



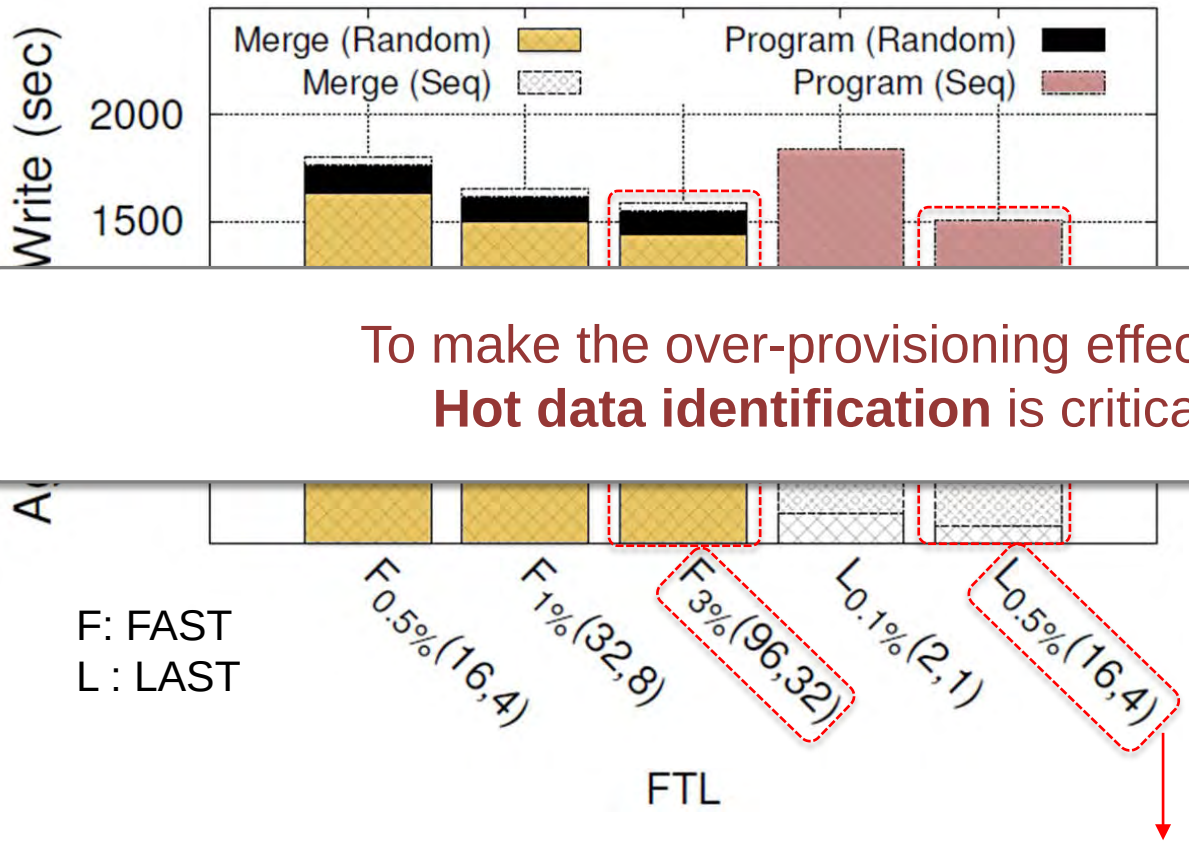
$$\text{Over-provisioning Factor} = \frac{\text{\# of Log blocks}}{\text{\# of Data blocks}}$$

$FTL_{30\%}(10, 5)$

(# of Rand log blocks, # of Seq log blocks)

Importance of Hot/Cold Identification

- Windows 7 Installation



- # of blocks / Flash
  = 4,096 blocks
- # of pages / Block

To make the over-provisioning effective,
Hot data identification is critical.

Same performance with 1/6 of OP

Conclusion

VSSIM: A novel SSD Simulation Tool

- Can measure **Host performance**
- Display SSD behavior in **Real-time**
- Can simulate **various SSD** architecture
- **Easily change** or fix firmware
- **Validate** VSSIM with Real-SSD

VSSIM is publicly available at http://esos.hanyang.ac.kr/vssim

Demo

Thank you

Q & A