Solid State Drive

Solid-State Disks (SSD)

What is “solid state” storage?

- RAM backed by a battery!
- “NOR flash”
- “NAND flash”
- Newer things

Solid-State Disks (SSD)

What is "solid state" storage?

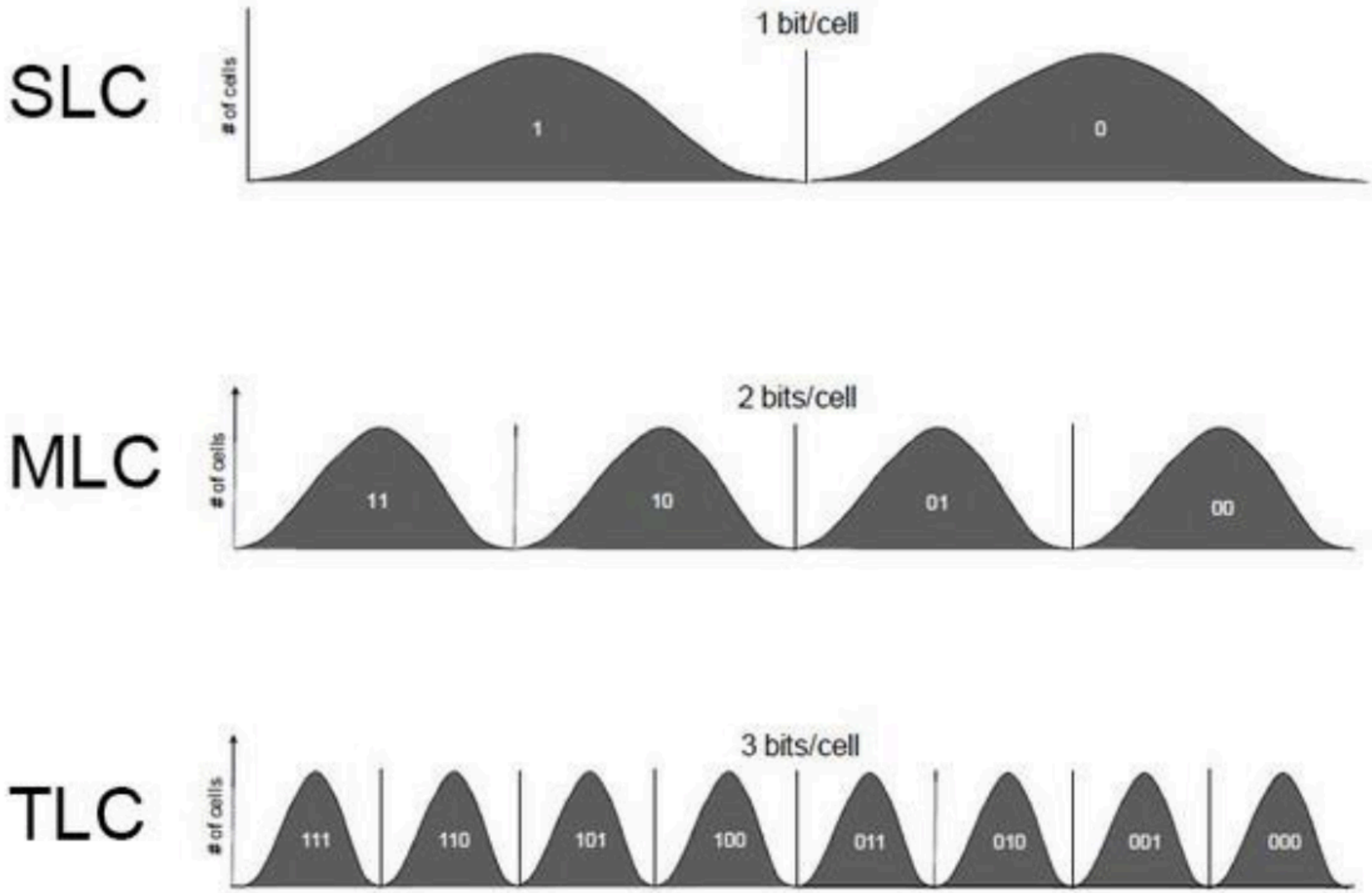
- RAM backed by a battery!
  - Fast
  - Legato "Prestoserve", 1989 ($8,000 for 1 MB)
  - Allowed NFS servers to complete write RPCs without waiting for disk
- "NOR flash"
  - Word-accessible
  - Writes are slow, density is low
  - Used to boot embedded devices, store configuration
- "NAND flash"
  - Read/write "pages" (512 B), erase "blocks" (16 KB)
  - Most SSDs today are NAND flash
- Newer things
  - "Phase-change" memory (melting), magnetic RAM, "Memristor" memory

Newer Things

What is “solid state” storage?

- “Phase-change” memory (melting)
- Magnetic RAM
- “Memristor” memory
- Intel's new “3D XPoint” / “Optane”
  - How it works isn't widely known
  - Characteristics
    - Word addressable (small random accesses are fast)
    - Slower than RAM, faster than NAND flash
    - Less power than RAM, more power than NAND flash
    - Doesn't have write amplification
    - Wear is less of a threat
    - Price is a multiple of NAND flash
  - Initially packaged as “Optane” SSD
  - Expected to be packaged later as DIMMs
    - Exact usage model unclear

NAND


SLC
1 bit/cell
# of cells
1
0
MLC
2 bits/cell
# of cells
11
10
01
00
TLC
3 bits/cell
# of cells
111
110
101
100
011
010
001
000

Solid-State Disks (SSD)

Architectural features of NAND flash

- No moving parts means no “seek time” / “rotational delay”
- Read is faster than write
- Write and “erase” are different
  - A blank page can be written to (once)
  - A written page must be erased before rewriting
  - But pages can't be individually erased!
    - “Erase” works on multi-page *blocks* (16 KB)
    - “Erase” is very slow
    - “Erase” *damages the block* each time

Implications

- “Write amplification”
- “Wear leveling”

Advantages

- Reliability in portable environments and no noise
  - No moving parts
- Faster start up
  - Does not need spin up
- Extremely low read latency
  - No seek time (25 us per page/4KB)
- Deterministic read performance
  - The performance does not depends on the location of data

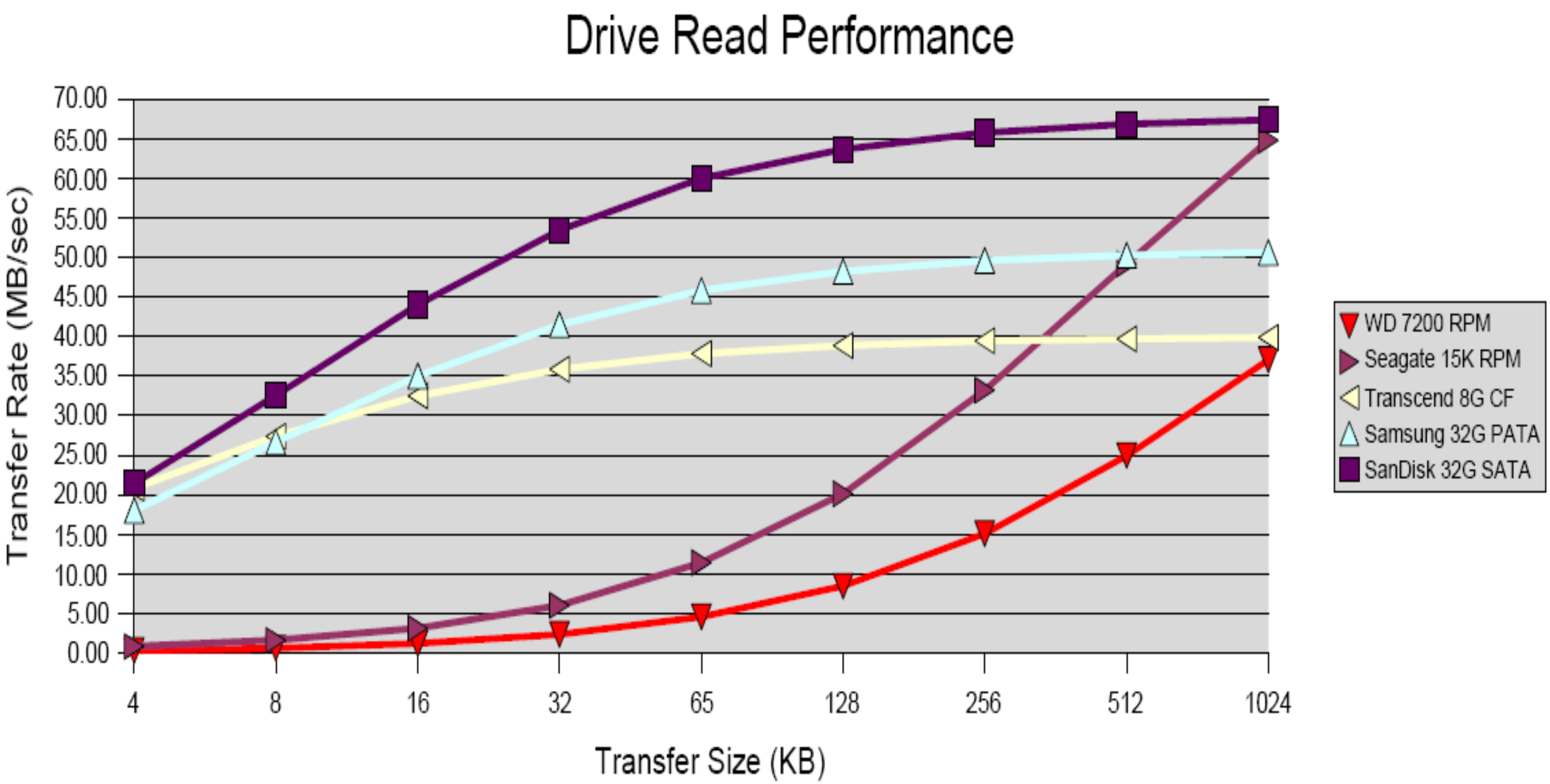
Disadvantage

- Cost significantly more per unit capacity
  - 3$/GB vs. 0.15$/GB
- Limited write erase time
  - 100000 writes for SLC (MLC is even fewer)
  - high endurance cells may have an 1-5 million
  - But some files still need more
  - Weaver leaving to spread writes all over the disk
- Slower write speeds because of the erase blocks are becoming larger and larger(1.5 ms per erase)
- For low capacity flash SSDs, low power consumption and heat production when in active use. High capacity SSDs may have significant higher power requirements

Typical read and write rates

	Drive Model	Description	Seek Time			Latency	Read XFR Rate		Write XFR Rate	
			Track to Track	Average	Full Stroke		Outer Tracks	Inner Tracks	Outer Tracks	Inner Tracks
Hard Drives	Western Digital WD7500AYYS	7200 RPM 3.5" SATA	0.6 ms	8.9 ms	12.0 ms	4.2 ms	85 MB/sec	60 MB/sec*	85 MB/sec	60 MB/sec*
	Seagate ST936751SS	15K RPM 2.5" SAS	0.2 ms	2.9 ms	5.0 ms*	2.0 ms	112 MB/sec	79 MB/sec	112 MB/sec	79 MB/sec
Flash SSDs	Transcend TS8GCF266	8GB 266x CF Card	0.09ms				40 MB/sec		32 MB/sec	
	Samsung MCAQE32G5APP	32G 2.5" PATA	0.14ms				51 MB/sec		28 MB/sec	
	Sandisk SATA5000	32G 2.5" SATA	0.125ms				68 MB/sec		40 MB/sec	

Drive read performance



Mixed writes and reads

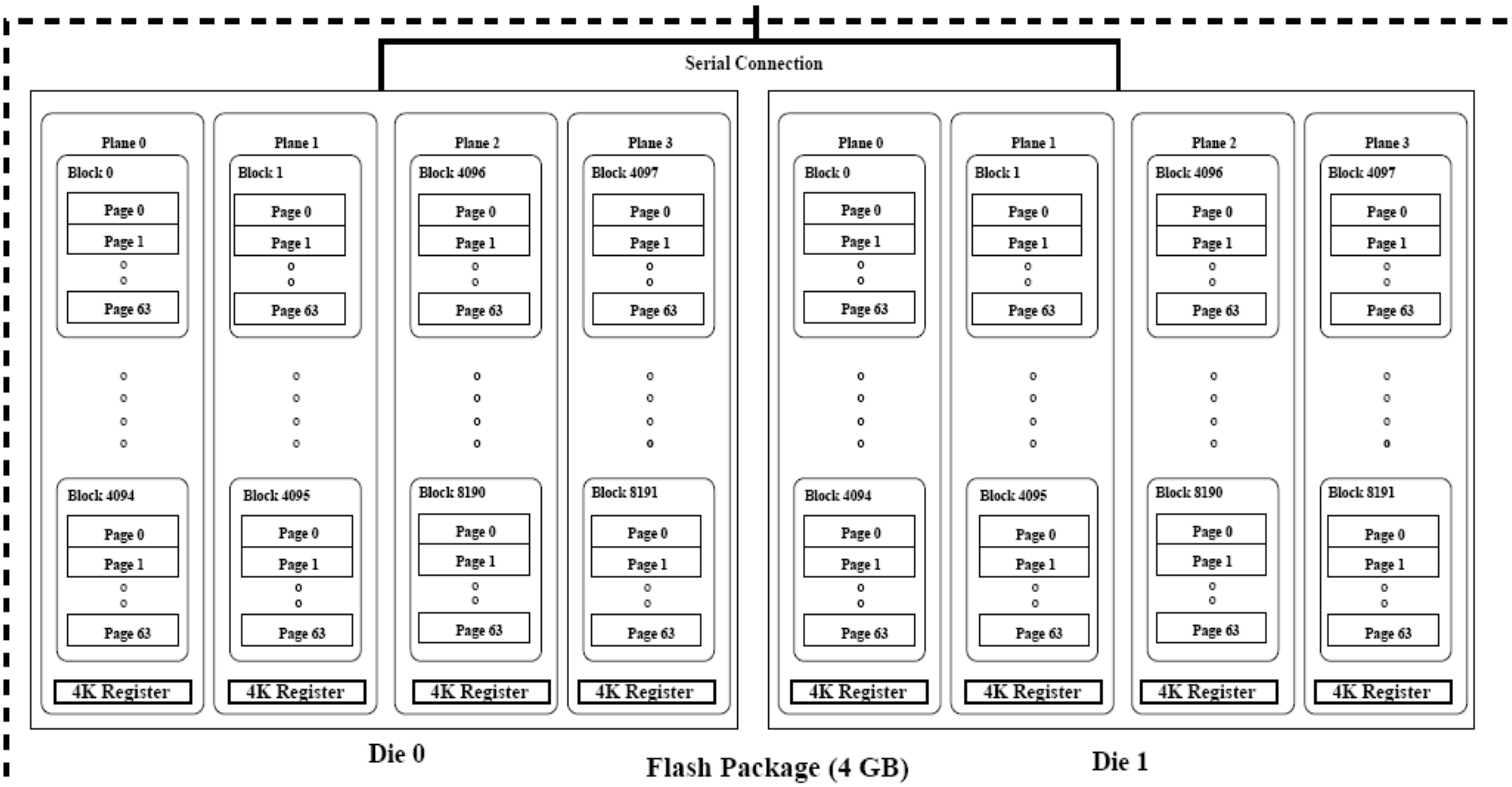
% Writes	Total IOPS	Performance vs 15K SAS Hard Drive
0%	5400	20x better
5%	252	1.25x better
10%	130	1.5x worse
20%	65	3x worse
50%	26	8x worse
100%	13	16x worse

	Sequential		Random 4K	
	Read	Write	Read	Write
USB	11.7 MB/sec	4.3 MB/sec	150/sec	<20/sec
MTron	100 MB/sec	80 MB/sec	11K/sec	130/sec
Zeus	200 MB/sec	100 MB/sec	52K/sec	11K/sec
FusionIO	700 MB/sec	600 MB/sec	87K/sec	Not avail

Power consumption

Device	Approximate power consumption
DRAM DIMM module (1 GB)	5W
15,000-RPM drive (300 GB)	17.2W
7200-RPM drive (750 GB)	12.6W
High-performance flash SSD (128 GB)	2W

Samsug flash internals



Bandwidth and interleave

- Without interleaving
  - For read: 25+100 us per page
    - 8000 reads/s = 32MB/s
  - For write: 200+100 us per page
    - 3330 writes/s = 13 MB/s
- With interleaving
  - For read
    - 10000 reads/s = 40MB/s
  - For write
    - 5000 writes/s = 20 MB/s

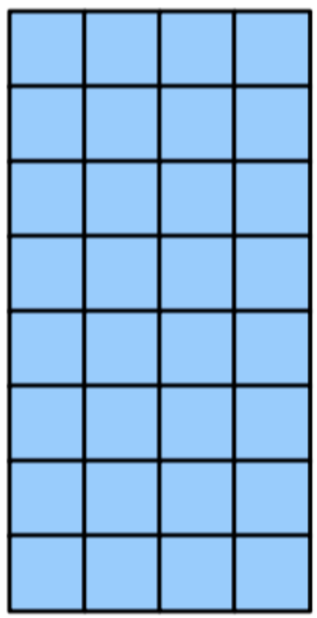
An example of SSD

- 1 die = 4 planes
- 1 plane = 2048 blocks
- 1 block = 64 pages
- 1 page = 4KB
- Dies can operate independently
- Reading and programming is performed on a page basis, erasure can only be performed on a block basis.

- Read
  - 25μs from page to data register
  - 100μs transfer in the serial line
- Write
  - Page granularity
  - Sequentially with in a block
  - Block must be erased before writing
  - 200μs from register into flash cells

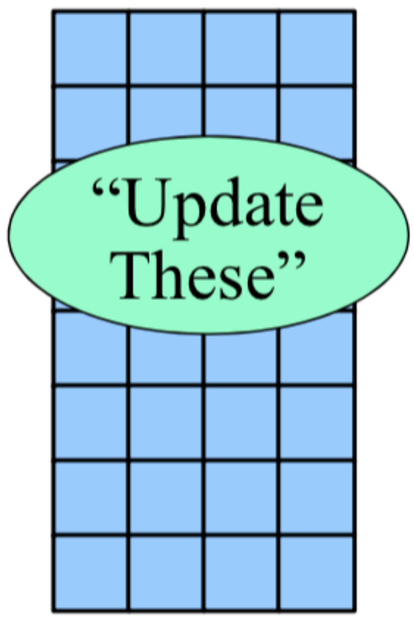
"Write Amplification"

Goal: update 8 pages (4 KB) in a block (16 KB)



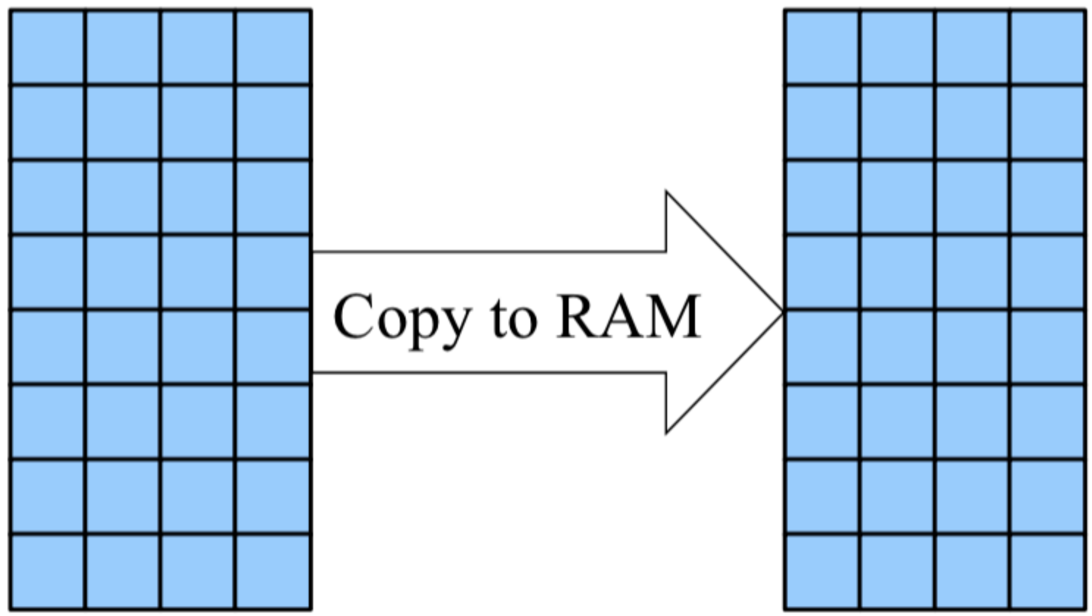
“Write Amplification”

Goal: update 8 pages (4 KB) in a block (16 KB)



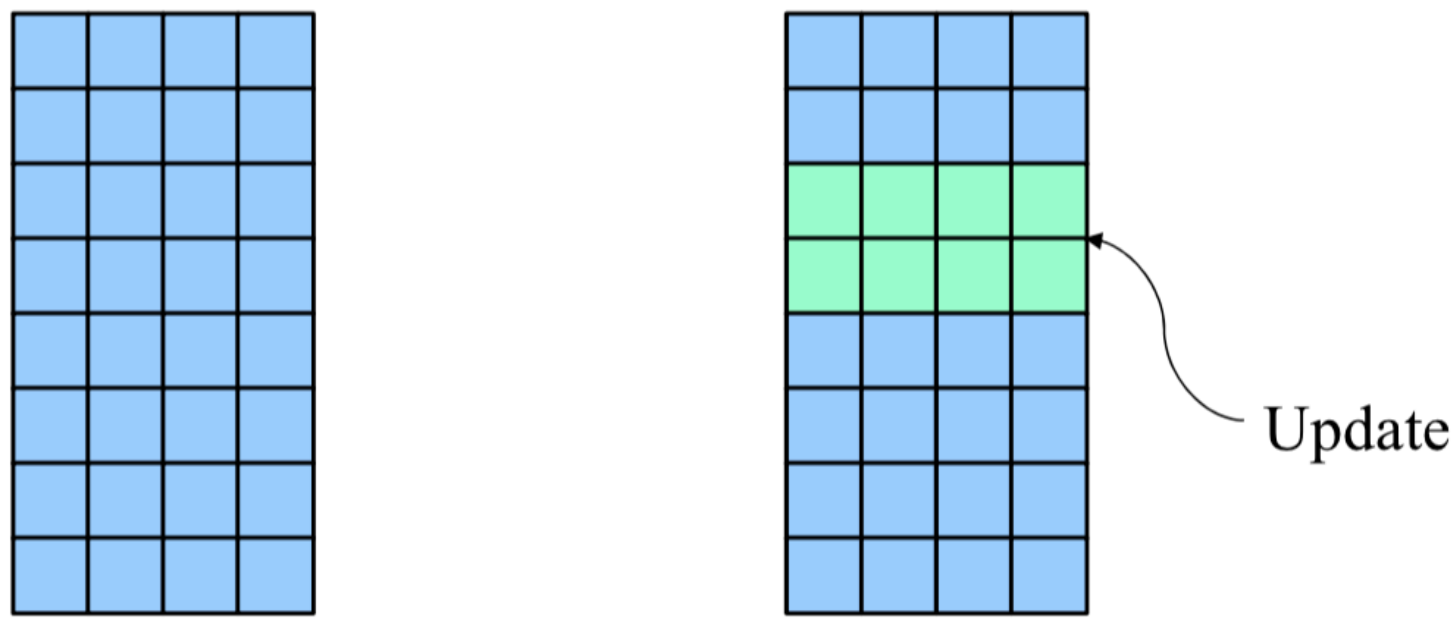
“Write Amplification”

Goal: update 8 pages (4 KB) in a block (16 KB)



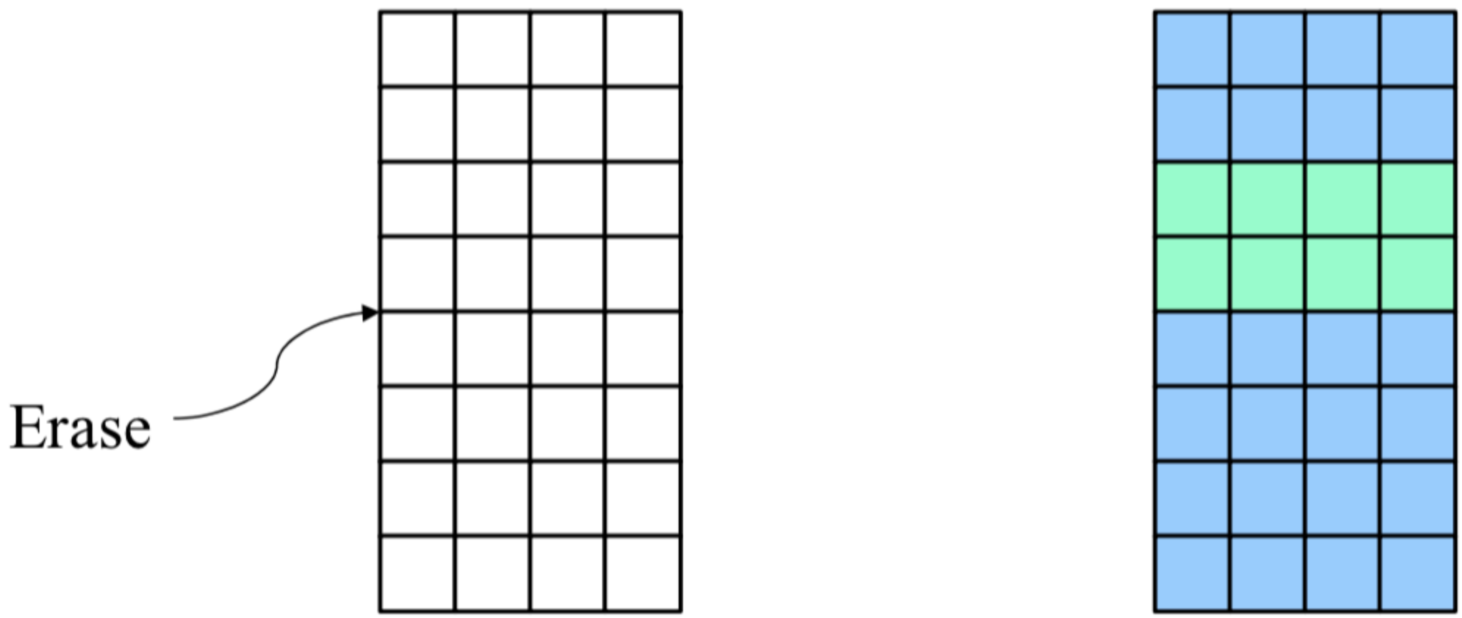
“Write Amplification”

Goal: update 8 pages (4 KB) in a block (16 KB)



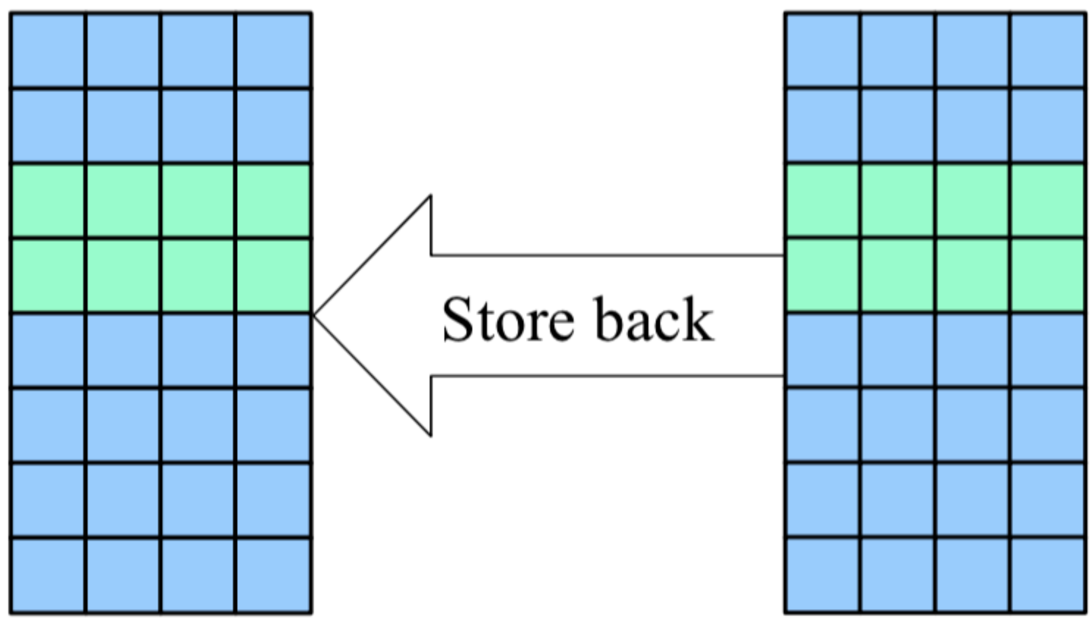
“Write Amplification”

Goal: update 8 pages (4 KB) in a block (16 KB)



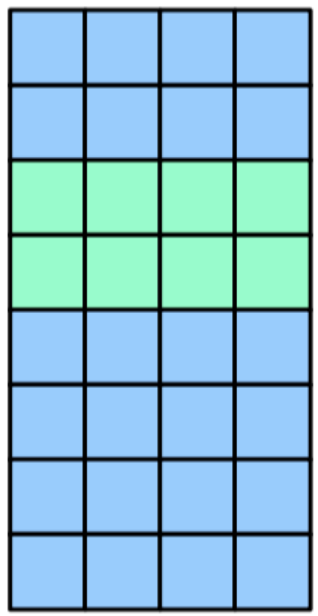
“Write Amplification”

Goal: update 8 pages (4 KB) in a block (16 KB)



“Write Amplification”

Goal: update 8 pages (4 KB) in a block (16 KB)



Result

- Logical: wrote 4 KB
- Physical: erased and write 16 KB
- “Amplification factor”: 4
  - Why do we care? Device will wear out 4X faster!

Hot-Spot Wear

The bad case

- File systems like to write the same block repeatedly
- Erasing damages part of the flash
  - ~10,000 erases destroys a block

Strategy: ?

Managing - Wear Leveling

The bad case

- File systems like to write the same block repeatedly
- Erasing damages part of the flash
  - ~10,000 erases destroys a block

Strategy: lie to the OS!

- Host believes it is writing to specific “disk blocks” - LBA
- Store the information somewhere else!
  - Secretly re-map host address onto NAND address
  - FTL - “flash translation layer”

Managing - Wear Leveling

The bad case

- File systems like to write the same block repeatedly
- Erasing damages part of the flash
  - ~10,000 erases destroys a block

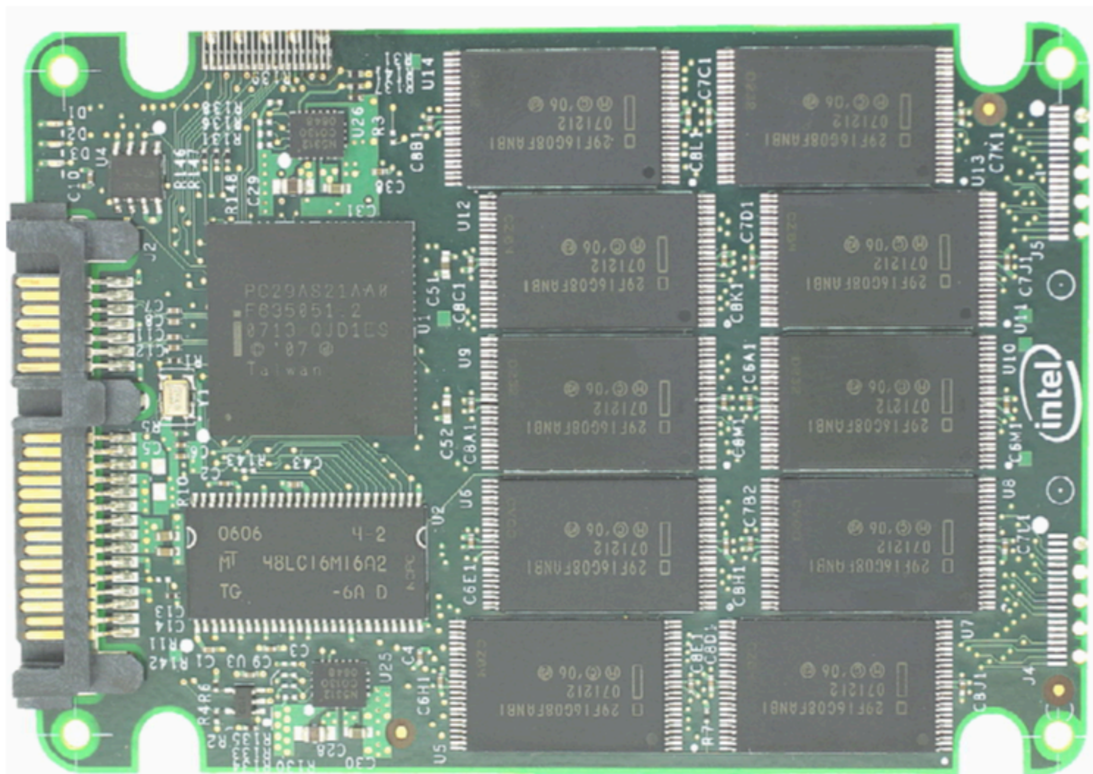
Strategy: lie to the OS!

- Host believes it is writing to specific “disk blocks” - LBA
- Store the information somewhere else!
  - Secretly re-map host address onto NAND address
  - FTL - “flash translation layer”
- Each part of the “disk” moves from one part of the flash to another over time
- “Over-provision”
  - Advertise less space than there really is
  - Use spare space to replace worn-out blocks
- Use up overprovisioning as blocks wear out

Wear Leveling - FTL

FTL is a *computer*

- **CPU, RAM**
- **Access to lots of flash for code & data structures & user data**



Managing - Write Amplification

The bad case

- Small random writes

Strategy: lie to the OS!

- Host believes it is writing to specific “disk blocks” - LBA
- Store the information somewhere else!
  - Secretly re-map host address onto NAND address
  - FTL - “flash translation layer”
- Group multiple small writes into full blocks
  - Write at sequential write rates
- To update a “disk block”, store a new copy *somewhere else*
  - Leaves “holes” in other blocks (stale old block versions)
  - At some point, “clean out” the holes by reading a bunch of old blocks and writing back a smaller number of whole pages
- Rate of cleaning depends amount of unallocated space
  - Controller reserves X% hidden space (ie. 10, 20, 50%)

SSD Summary

SSD vs. disk

- ☺ SSD's implement "regular disk" model
  - LBA sectors
  - Write-sector, read-sector, "park heads", etc.
- ☺ Read operations are extremely fast (100X faster), no "seek time" or "rotational delay" (every sector is "nearby")
- ? Write operations "vary widely" (maybe 100X faster, maybe not faster at all)
- ☺ SSD's use less power than actual disks (~1/5?)
- ☺ SSD's are shock-resistant
- ☹ Writing to an SSD wears it out much faster than a disk
- ☹ SSD's are *expensive* (20X or more)

SSD Summary

Opportunity & threat

- "TRIM" command speeds up writes!
  - "Dear FTL, logically zero-fill these blocks"
- "Securely erase disk" may or may not be possible

The future?

- Lots more SSD's
- Lots more disks too
- Hybrid systems to take advantage of best features of both

What You Should Know

Storage is *slow*

- Whatever you want to do may take *milliseconds*

Storage lies

- You get some number of “disk blocks”
- There is no way to know where on the “disk” they are
- LBA is a faint approximation of proximity

Failure model

- Sometimes a read fails (sorry!)
- Writing to that block will cause the device to re-map
  - Both spinning-disk and SSD
- When re-map space is exhausted, device refuses to write

Security

- Actually erasing information from flash is uncertain