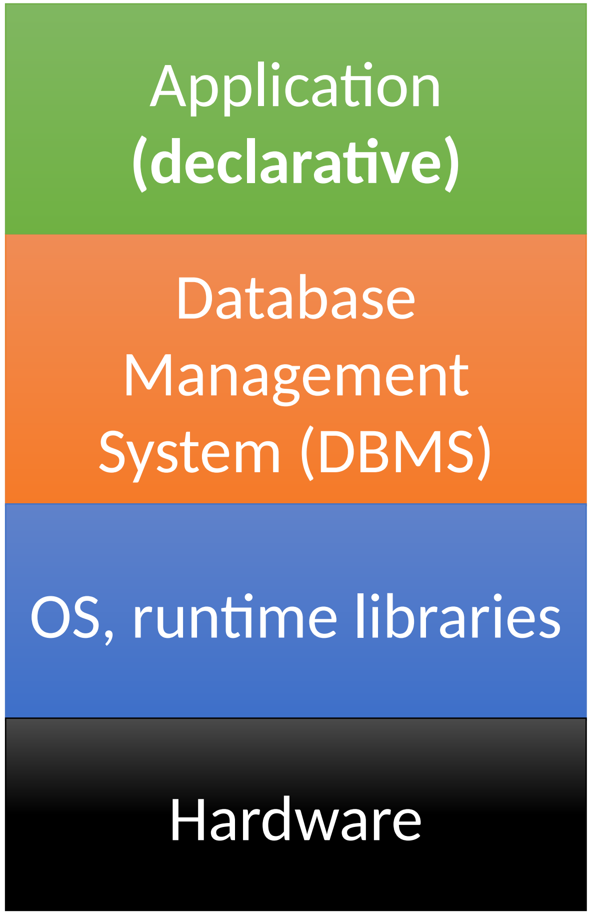
CSE 541: Database Systems I

Memory Hierarchy and Storage Devices

Slides Credit: Tianzheng Wang, Ryan Stutsman

Big Picture



Application **(declarative)**

Application that uses a DBMS to access databases (typically SQL)

Database Management System (DBMS)

Software system that handles access to databases (records, tables) in storage devices

OS, runtime libraries

Tools for building a DBMS: file systems, synchronization primitives…

Hardware

CPU, memory, **storage devices**, network…

Memory & Storage

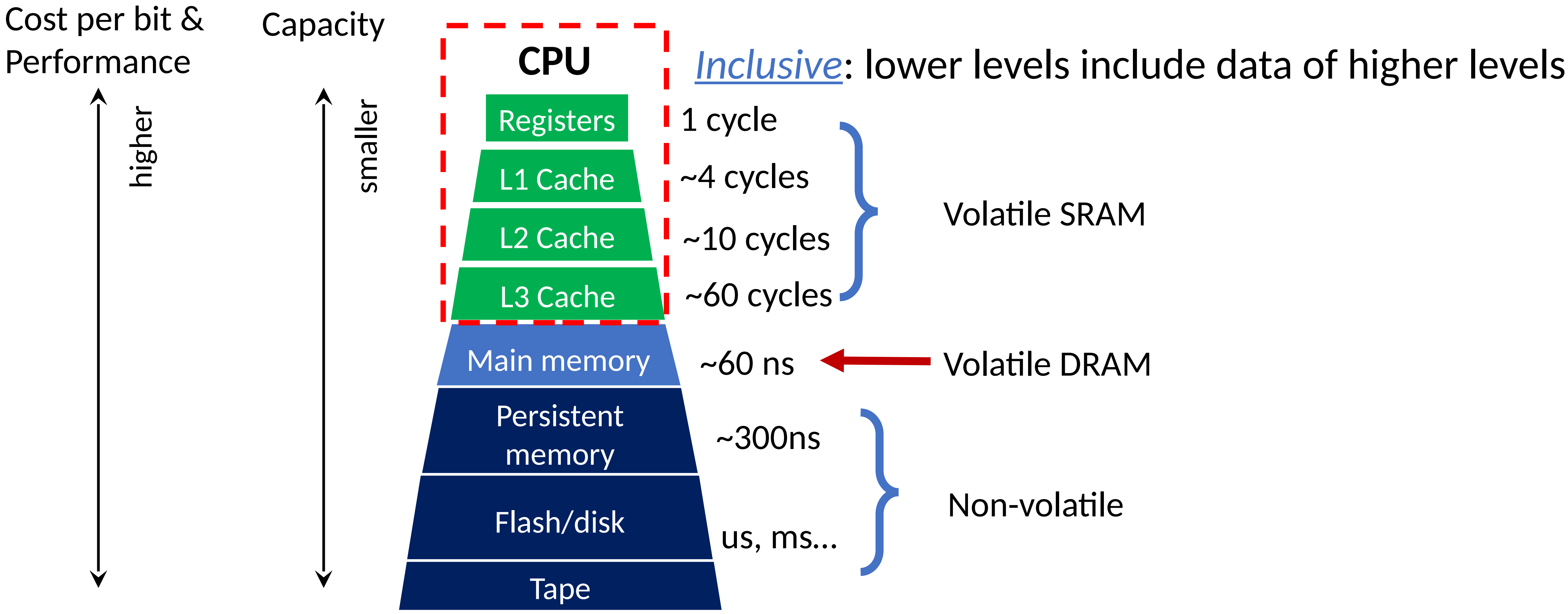
What's the perfect device to store data (in dream)?

- Capacity: *unlimited*
- Bandwidth: *unlimited*
- Access speed: *instant*
- Price: *free*
- Data retention: *forever*
- Reliability: *never fails*

Nah, impossible…….

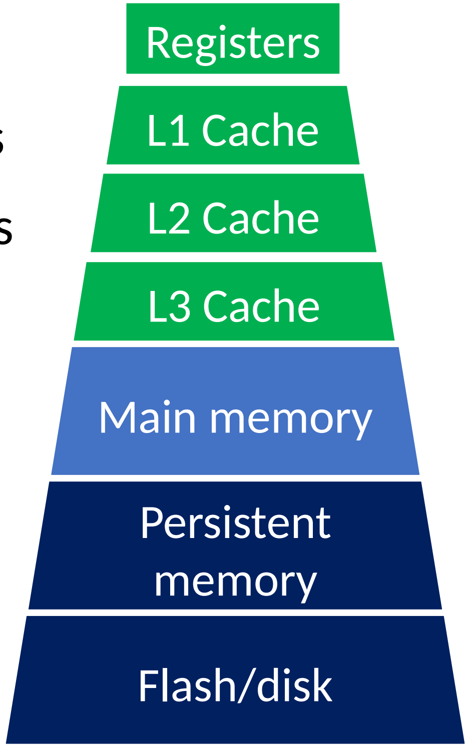
*Tradeoffs among **performance**, **capacity**, **endurance** and **cost***

Memory/Storage Hierarchy



Must bring data from storage to memory for CPU to access

Relative Speed – an Analogy

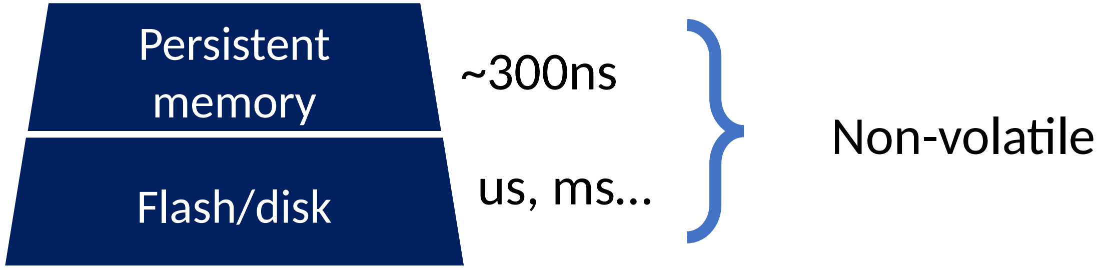


1 cycle	Registers	
~4 cycles	L1 Cache	Get a piece of paper from desk
~10 cycles	L2 Cache	Get a book from a bookshelf
~60 cycles	L3 Cache	Get a book from the next room
~60 ns	Main memory	Walk down the hall to grab a coffee
~300ns	Persistent memory	
us, ms…	Flash/disk	**Walk from Penn State to Rio in Brazil!**

Storage

Permanent home of data

- Hard disk
- SSD (solid state drive)
- Storage is much cheaper that memory
    - 3GB memory or 2000GB disk about the same cost
    - Access time for disk is at least one order of magnitude slower than for memory



Persistent memory

~300ns

Flash/disk

us, ms…

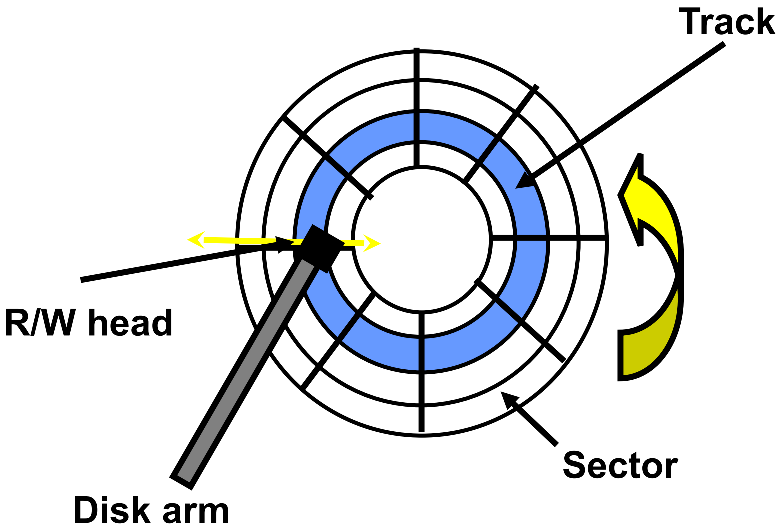
Non-volatile

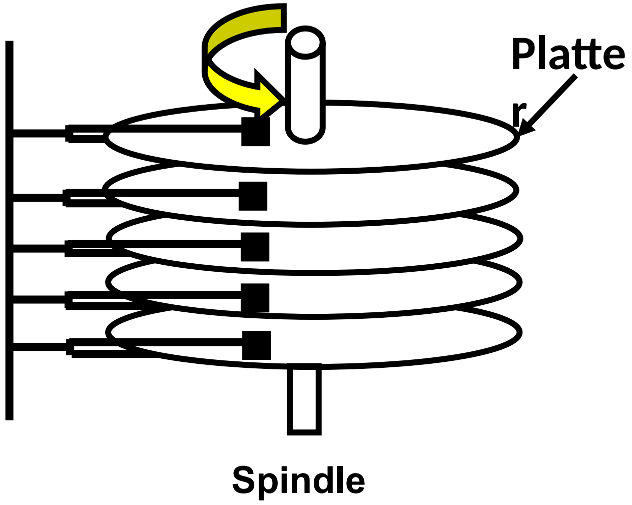
Block Devices

- Most storage devices used today are block devices.
- Disks has a **sector**-addressable address space.
  - Array of sectors (Typically 512B or 4096B)
  - Sector is the **unit of atomicity**.
- Main operations: read + write to sectors
- The nature of its “slower” access makes management “interesting”.

HDD Organization

- Coated with magnetic material that encodes bits
  - Capacity increases come from improvements in bit density
- Logically divided into:
  - Track: ring on a platter
  - Sector: unit of r/w, portion of a track
  - Cylinders: stacks of tracks
- Read/write data (overview):
  - Position disk head over track
    - Seek time
  - Wait for sector to rotate under head
    - Rotational delay
  - Read/write data from/to sector
    - Transfer time
  - Total Delay =
    Seek + Rotation + Transfer





HDD Organization

- Disk physics:
  - Modern disks spin at 5400, 7200, 10000, and 15000 rpm
  - Outside edge of 3.5” disk spins at over 150 mph
  - Disk head “floats” on very thin cushion of air above platter
    - Bernoulli effect used to “fly” as close as possible
    - Head crash is exactly that → disk head contacts the surface
- Disks organized as stacks of platters:
  - Disk heads mounted on “combs” → often heads on both sides
  - Disk arms/heads have a single actuator; they must move together
- Disk controller
  - Managing arm/head movements
  - Contains RAM to cache disk contents from/to disk
  - Accepts commands from CPU → responds using DMA/interrupts

CC BY-SA 3.0 Eric Gaba, Wikimedia Commons user Sting

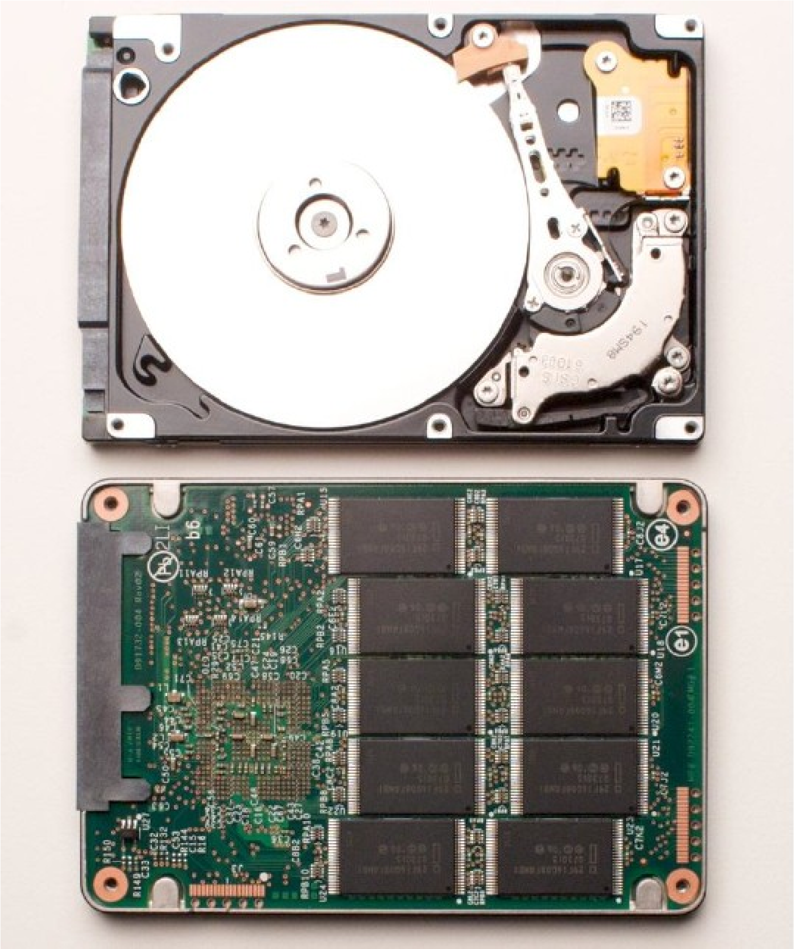
HDD Performance Model

- Seek: Move head to the right track.
  - Slow, 4-10ms.
- Rotate: Rotate to the starting sector.
  - Slow, depends on RPM, ~4ms
- Transfer: Transfer the data out through I/O bus.
  - Fast, depends on RPM.

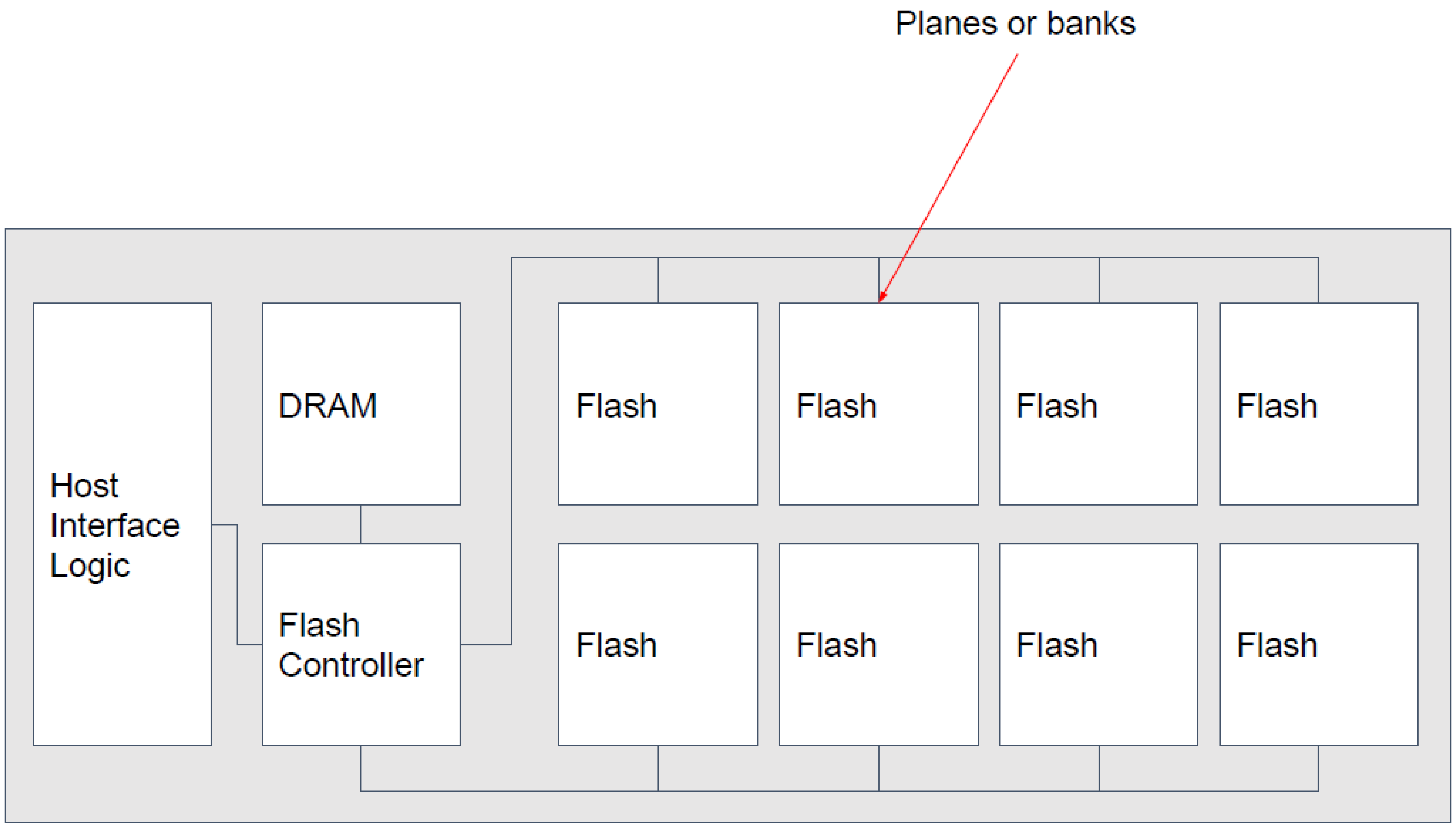
- **Sequential** vs. **Random** Access.

Flash Memory / SSDs

- Solid State
- Non-volatile
- No moving parts
  - More reliable
  - More shock-resistant
- Much faster than HDD
- Much slower than DRAM

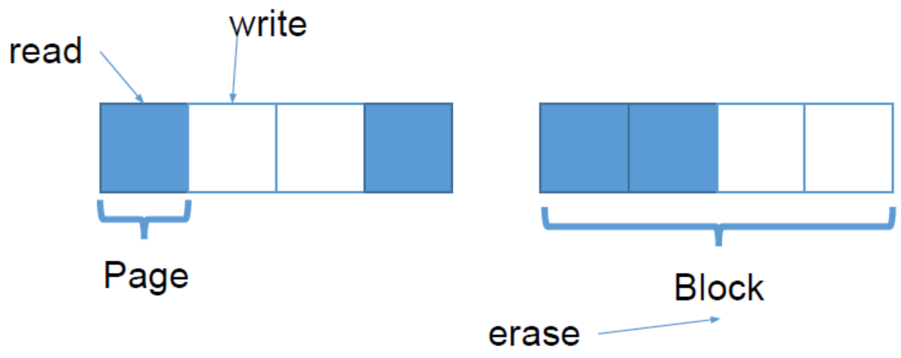


Flash SSD Internals



Flash/SSD Operations

- Read, Write and **Erase**
- Typical ~4KB pages for read/write
- ~256 blocks for erase (64 pages per block)
- Erase: reset all bits in a **block** to 1s
- Write: clear some bits in a **page** to 0s
  - Cannot change a 0 to a 1(requires erase first!!!)
- Read: read a single 4KB **page**



- 4KB Page Read
  - 20 to 100µs (> 100µs from application point of view)
  - > 500MB/s bandwidth (parallel access across banks)
- 4KB Page Write
  - 200µs
  - > 500MB/s bandwidth
- 256KB Block Erase
  - 2ms
  - Ever write takes 2ms?!

Flash Wear

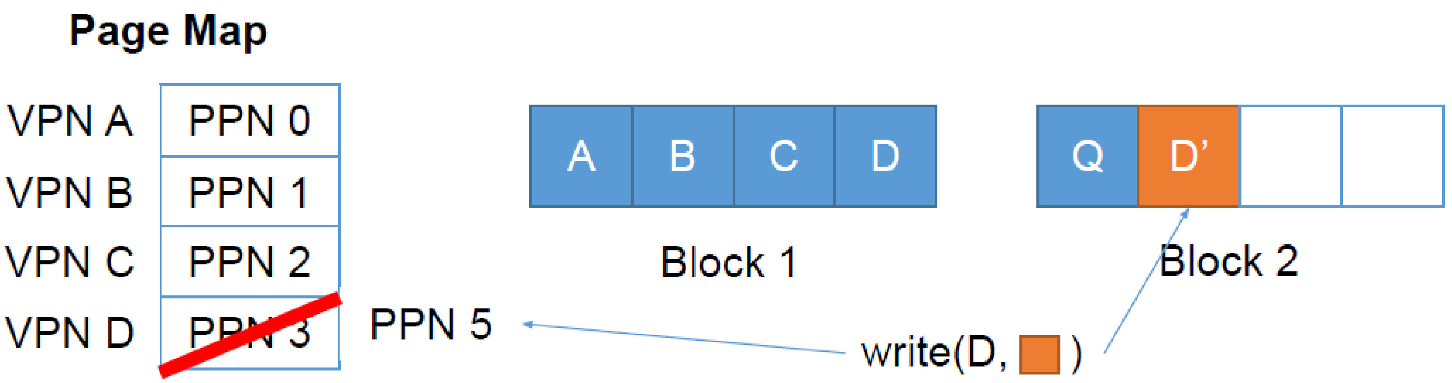
- Cells become unreliable after certain number of erase cycles
- About 100K erases for SLC NAND
- About 10K for MLC NAND
- About 3K for TLC, about 1K for QLC

- Some blocks are written more frequently than others
- What can we do about high churn blocks? (e.g. file system bitmap? FAT?)

- Don’t want a device that only last as long as its weakest cell

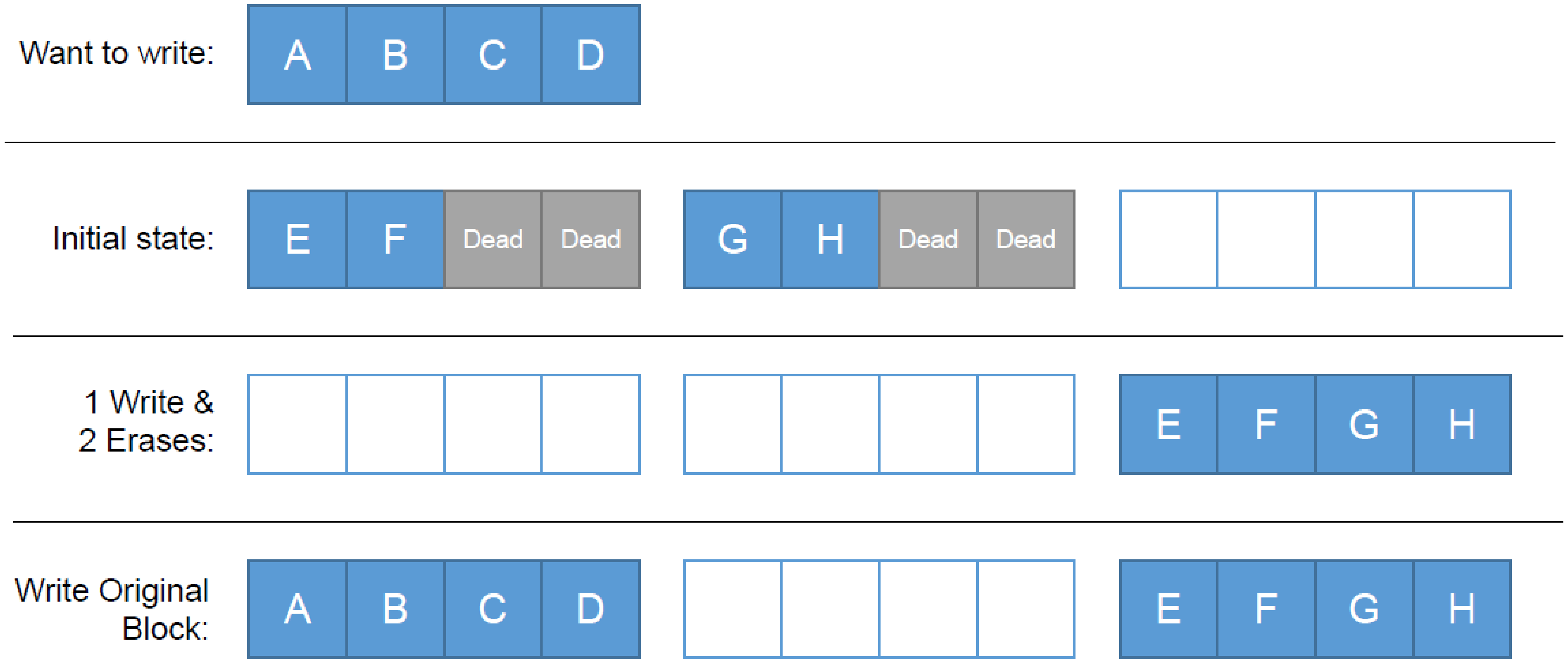
Flash Translation Layers (FTL)

- Idea: can’t rewrite a block in place quickly, instead write new contents to a pre-erased page
- How: indirection, map virtual page numbers to physical pages
- While at it, recycle blocks with the lowest erase count first
  - Rapid updates to single VPN backed by different PPNs



Write Amplification

(Device-level Block Writes) / (Host-level Block Writes)



Want to write: A B C D

Initial state: E F Dead Dead | G H Dead Dead | (empty)

1 Write & 2 Erases: (empty) | (empty) | E F G H

Write Original Block: A B C D | (empty) | E F G H

WA = 2 blocks / 1 blocks = 2.0x

(Flash-based) SSD Performance Model

- Parallel Access: Queue Depth (can up to 64)
- No seek time: Better Random I/O → 10k – 1M IOPS
- Higher Bandwidth: up to 7GB on PCIe 4.0, ~500MB on SATA.

- Asymmetric Read/Write
- Aging in Performance

- Recent PCM based SSD (Intel Optane) is even more astonishing.

Main Memory

DRAM – dynamic random access memory

- Need constant current to retain data
- Volatile: contents lost when power is out
- Current street price: ~$7-30/GB
- Lifetime: can be programmed > 10^{16} times

Media Comparison

- HDD
  - Highest density
  - Millisecond scale latency
  - 100s of IOPS, ~100MB/s bandwidth
- SSD
  - Moderate density
  - Several hundred microsecond scale latency
  - 100Ks of IOPS, up to 7GB/s bandwidth
- Main Memory
  - Low density
  - ~100ns latency
  - Bandwidth can easily hit > 40GB/s

* Let’s forget persistent memory (storage-class memory) for now.