THE UNIVERSITY *of* EDINBURGH
informatics

Operating Systems (INFR10079)

2022/2023 Semester 2

Secondary-storage

abarbala@inf.ed.ac.uk

Chapter 11

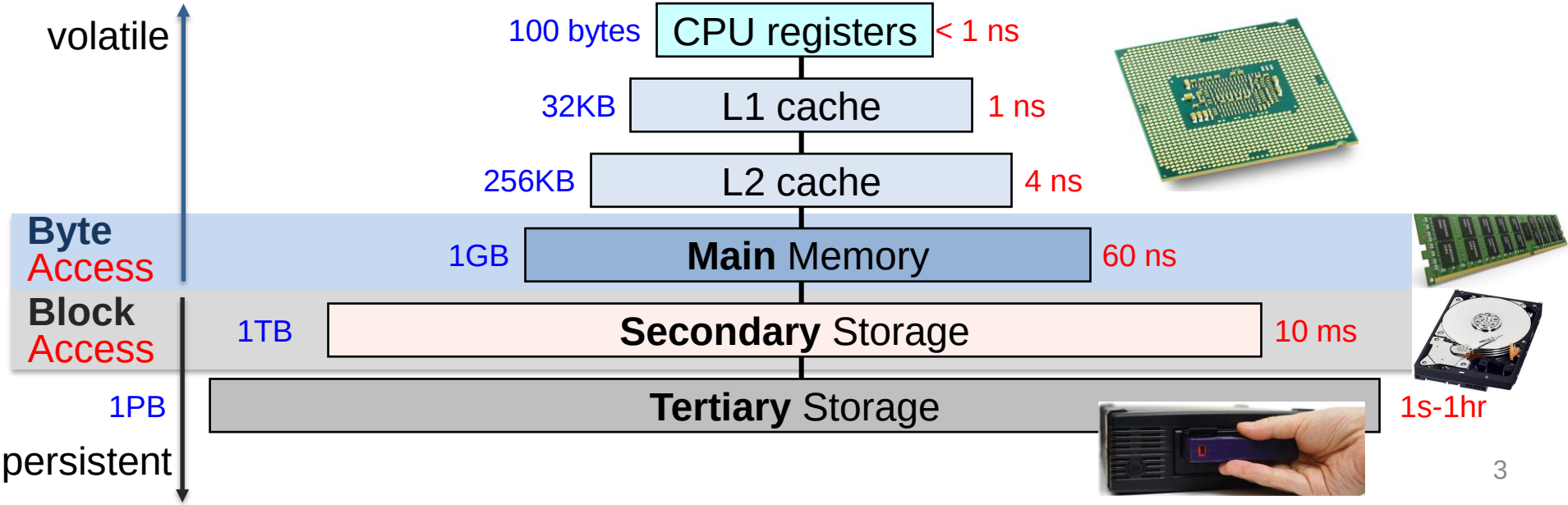
Secondary-storage: Overview

- The Memory Hierarchy
- Magnetic Disks Drives (HDD)
  - Technology
  - Performance
  - Scheduling
  - Scheduling Algorithms
- Solid-state Drives (SSD)
  - Read/write
  - SSD vs HDD
- Technology Update

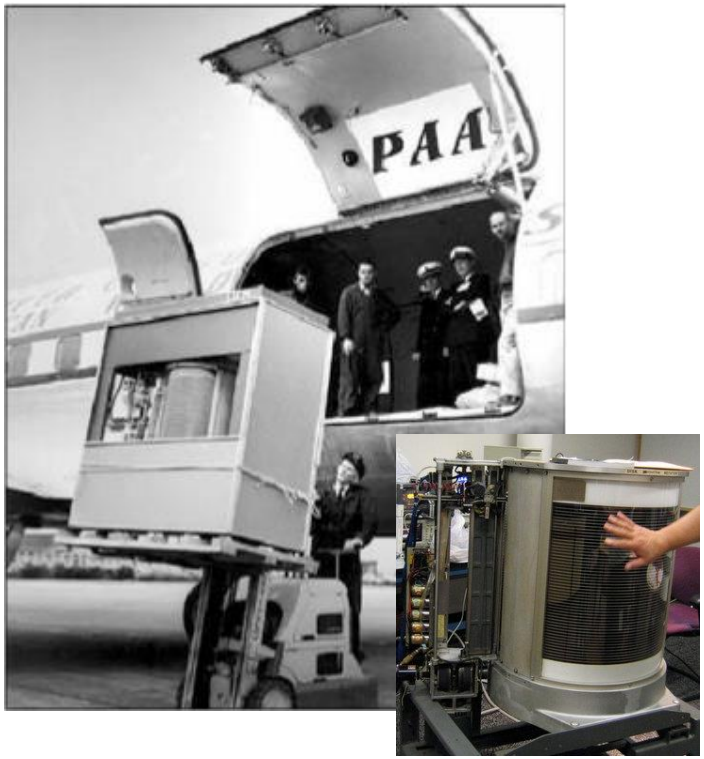
Traditional Secondary Storage

block = multiple bytes (e.g., 512B, 4kB)

- **Block access (vs byte access)**
  - CPU cannot access secondary storage directly
  - CPU accesses primary storage directly (e.g., move instruction)
- Characteristics
  - **Large**: 500 - 8000GB and more (HDD)
  - **Cheap**: 0.035gbp/GB (HDD)
  - **Slow**: millisecond (HDD)
  - **Persistent**: data survives power loss
  - **Fail rarely**
    - Drive dies; Mean Time Between Failure (MTBF) ~3 years
    - 100,000 drives and MTBF is 3 years, 1 “big failure” every 15 minutes!



Early Magnetic Disk Storage Systems



1956
IBM Model 350 disk storage system

5M 6-bit characters (3.75MB)
50 x 24” platters
8,800 character/sec
(part of IBM RAMAC computer)

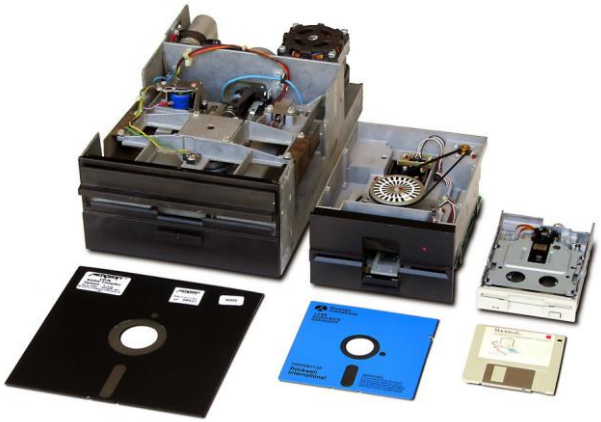
1965
IBM 2314 storage system

29.2M bytes (29.2MB)
8 x 11 platters
310,000 byte/sec

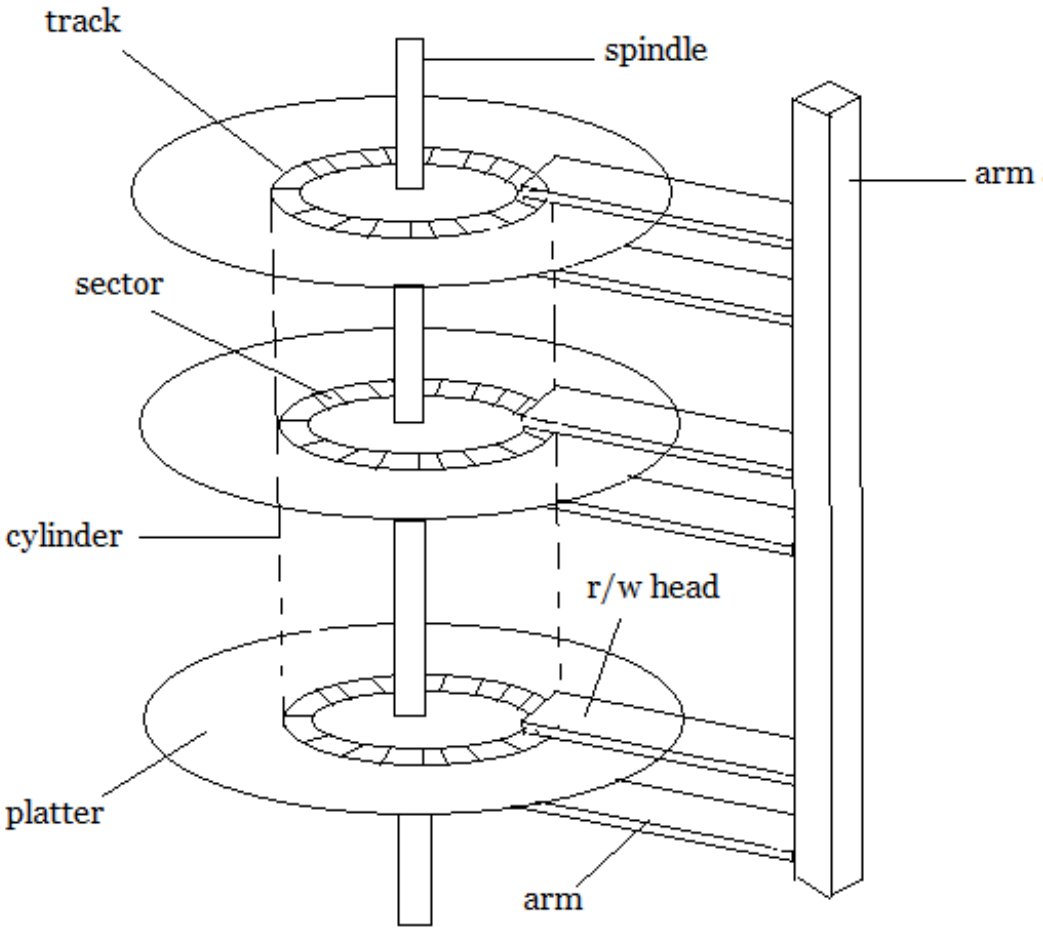
Magnetic Disks #1

Hard Disk Drive (HDD)

Floppy Disk Drive (FDD)



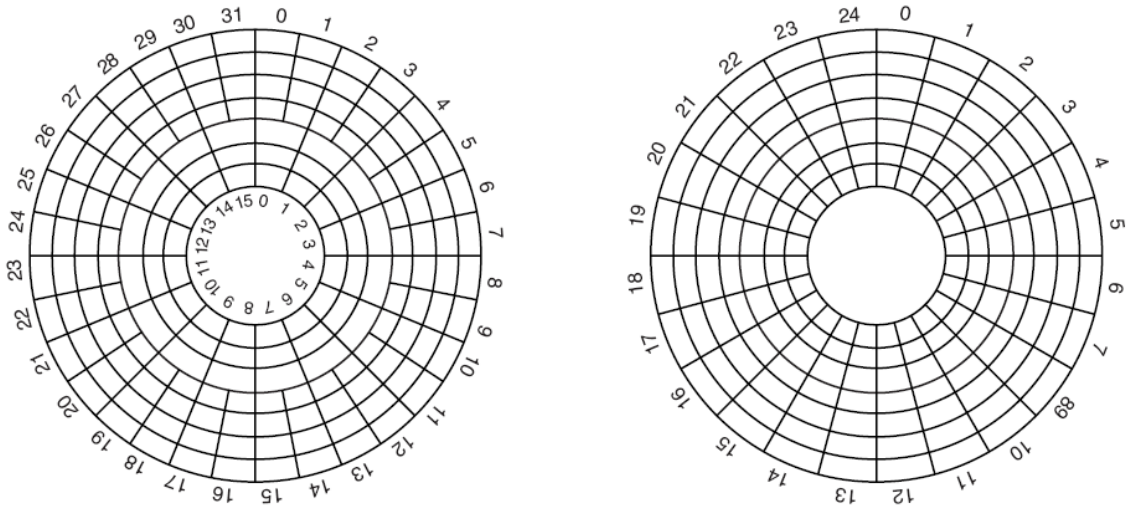
(single arm/head)



https://www.studytonight.com/operating-system/images/secondary-storage-1.png

Magnetic Disk #2

- Read/write errors, bad blocks, missed seeks, etc.
- Physical Geometry vs Addressing
  - **Previously** geometry used for addressing: **head, cylinder, sector**
  - **Now** independent: **Logical Block Address (LBA)**
    - Mapped onto the sectors of the disk sequentially



(**left**) Physical geometry of a disk with two zones. (**right**) A possible virtual geometry (addressing) for this disk

Example: Seagate Barracuda 3.5” Disk Drive

- **35gbp** cost (March 2020)
- **1Terabyte** of storage (1000 GB)
- 4 platters, 8 disk heads
- 63 sectors (512 bytes) per track
- 16,383 cylinders (tracks)
- 7200 rpm
- up to 300 MB/second transfer (SATA)
- 9 ms avg. seek, 4.5 ms avg. rotational latency
- 1 ms track-to-track seek
- 64 MB cache

… in March 2023, **1TB** costs **32gbp**, **4TB** costs **75gbp**, **8TB** costs **125gbp**

Disk Performance

- Depends on …
- **Seek time:** moving the disk arm to the correct cylinder
  - Depends on how fast disk arm can move
    - Not diminishing quickly due to physics
- **Rotation (latency):** waiting for the sector to rotate under head
  - Depends on rotation rate of disk
    - Rates are slowly increasing
- **Transfer time:** transferring data from surface to disk controller
  - Depends on density of bytes on disk
    - Increasing, relatively quickly

- When the OS uses the disk, **tries to minimize** all such costs
  - Specifically, seeks and rotation

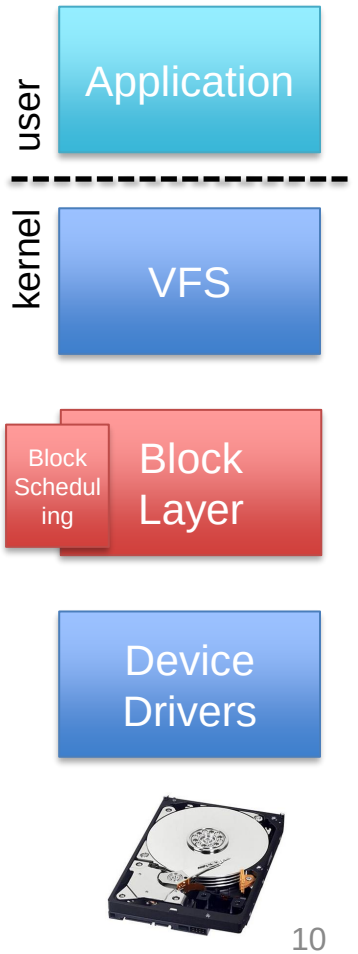
Software Performance

- OS may increase file block size
  - **Reduce seeking**
- OS may aim at co-locate “related” items
  - **Reduce seeking**
    - Blocks of the same file
    - Data and metadata for a file

- OS may keep data or metadata in memory to reduce physical disk access
  - **Avoid** slow disk **accesses**
    - But wasting valuable physical memory
- OS may fetch blocks into memory before requested
  - **Hide** slow disk **accesses**

Performance via Block Scheduling

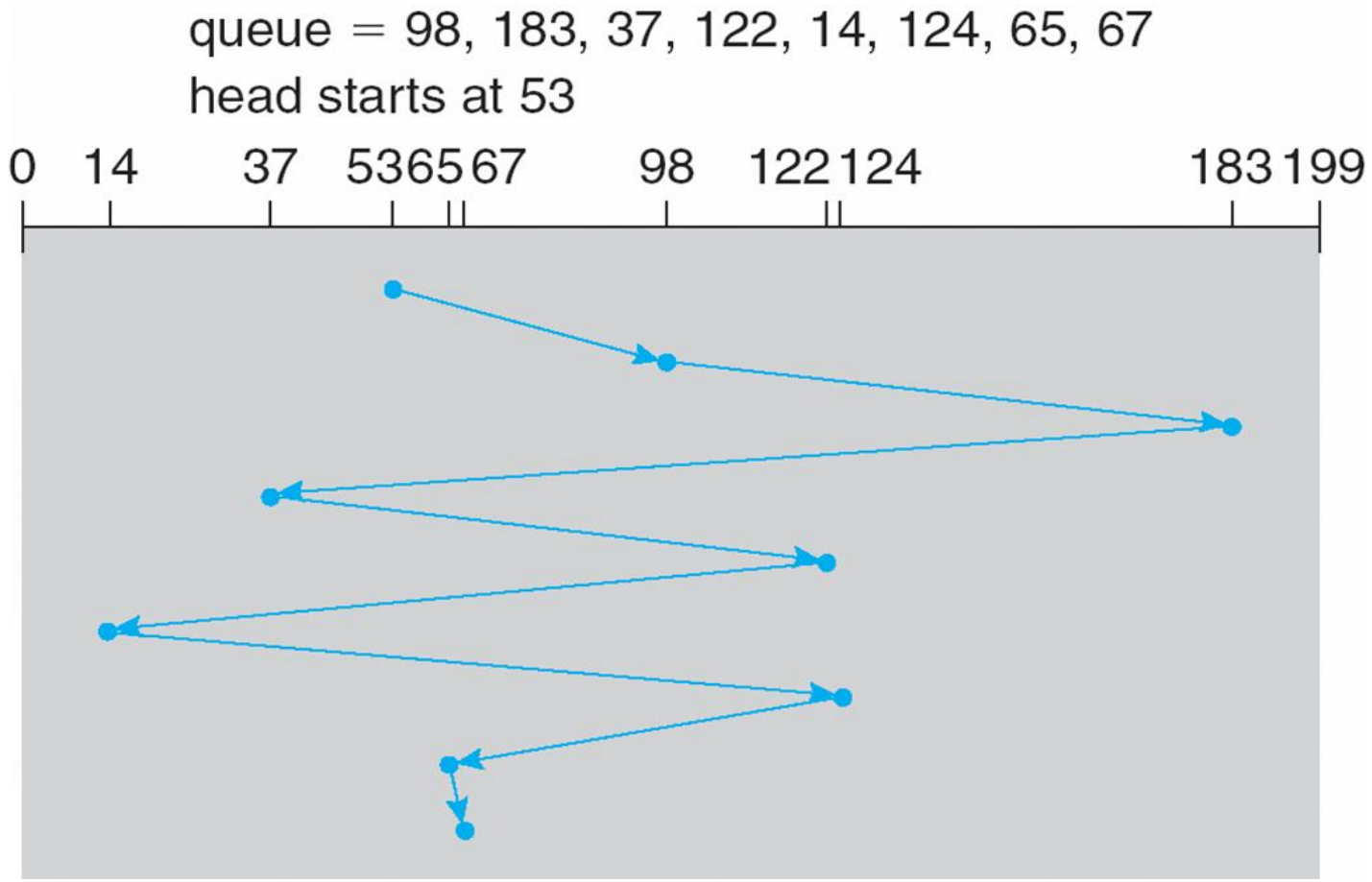
- Applications request data accesses to the OS
  - OS maintains **request queues**
  - OS generates **transfer commands** to/from the disk(s)
    - Imply seeks, waits for rotations, data transfers
- How to **reduce** applications' **waiting time**?
  - OS modifies **order of block requests queued waiting** for the disk
    - Traditionally, based on cylinder #
  - Fairness, timeliness, etc.
- Multiple **block scheduling** algorithms
  - FCFS (first come first served, no scheduling)
  - SSTF (shortest seek time first)
  - SCAN (elevator algorithm)
  - C-SCAN (typewriter)



https://www.thomas-krenn.com/en/wiki/Linux_Storage_Stack_Diagram

FCFS

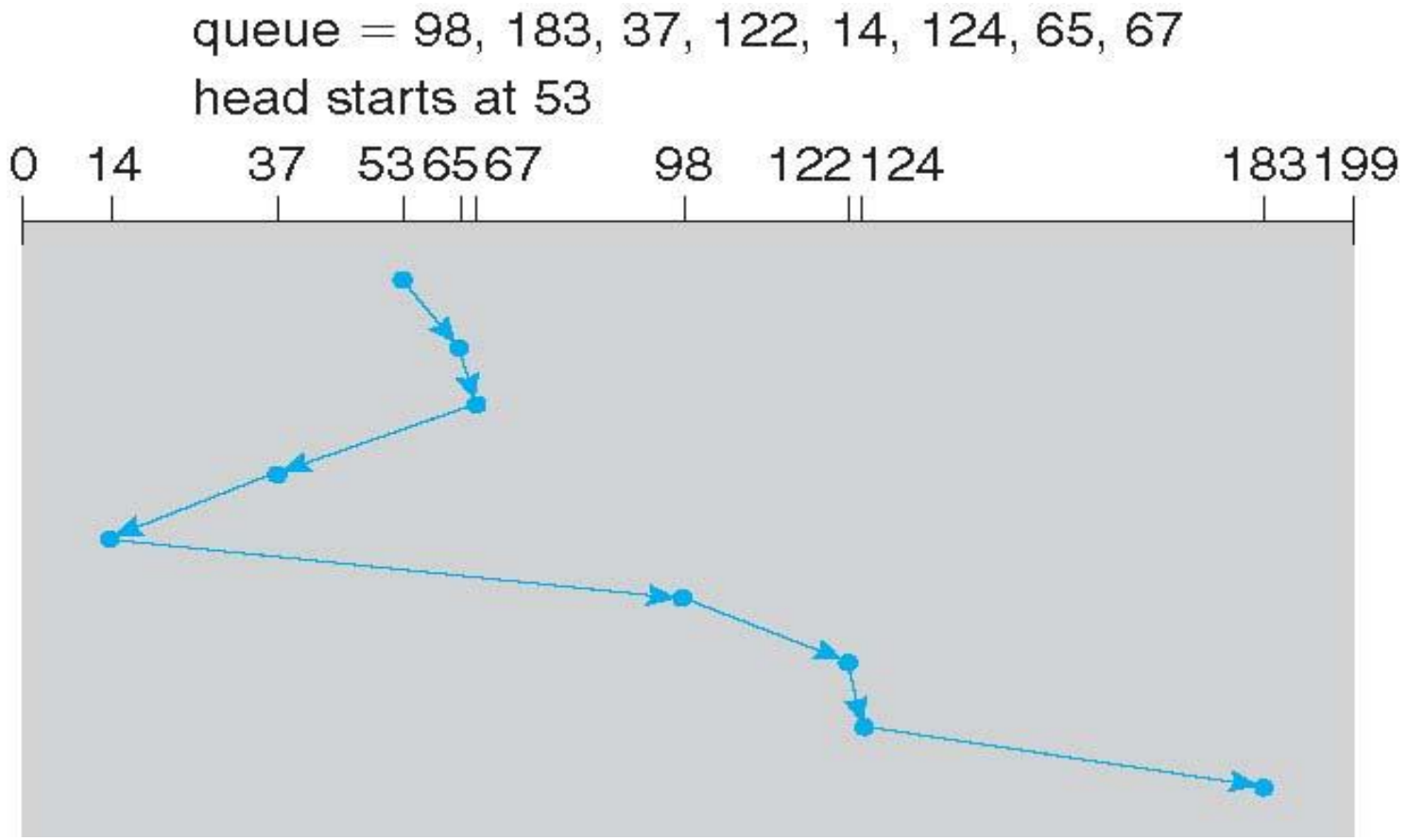
- First come first served



- Reasonable when load is low
- Long waiting time for long request queues

SSTF

- Shortest seek time first



- Minimize arm movement (seek time), maximize request rate
- Unfairly favors middle (clustered) blocks

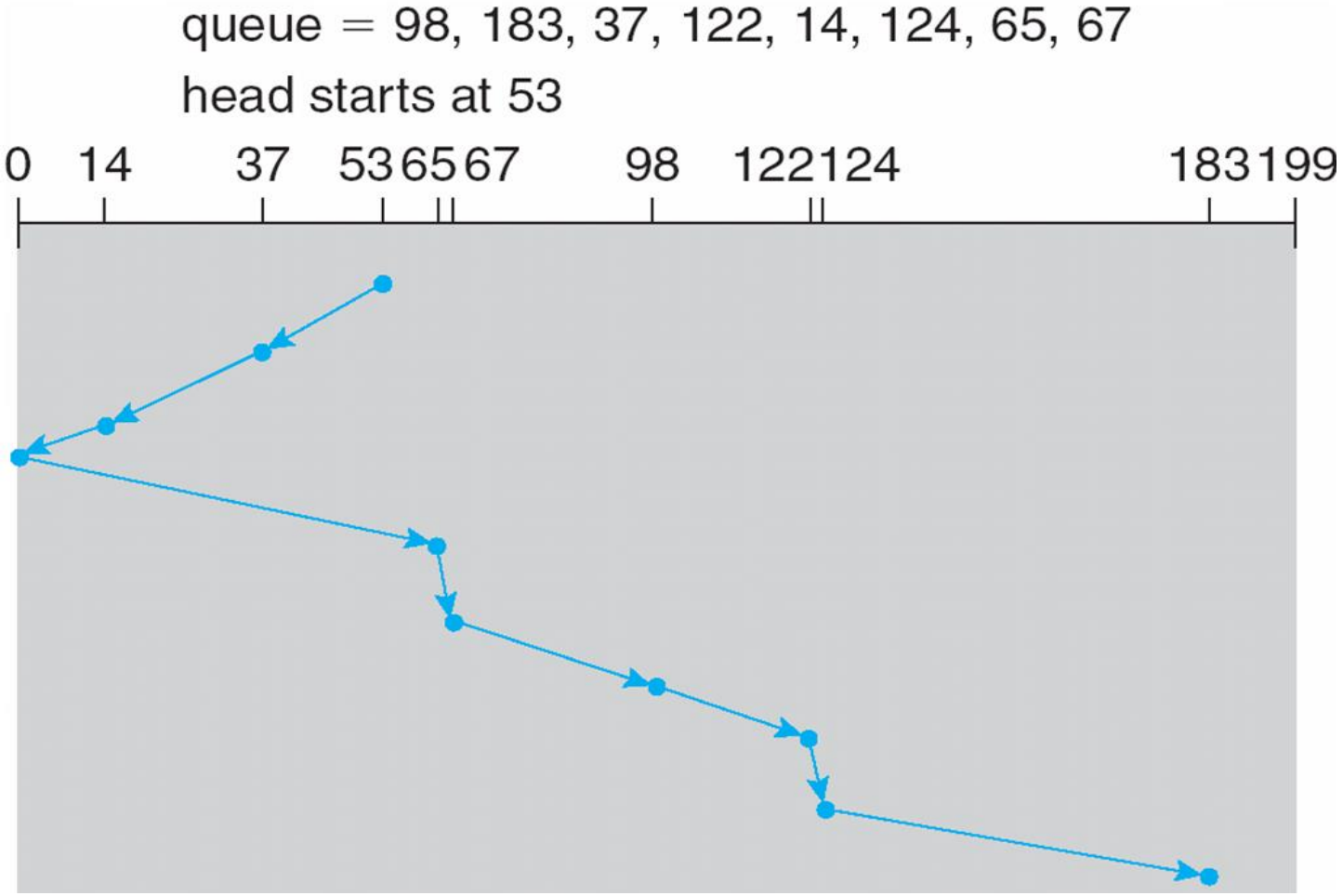
SCAN #1

- Disk arm **starts at one end** of the disk
  - Moves **toward the other end**
- Servicing requests until it gets to the other end of the disk
  - Where the head movement is reversed, and **servicing continues**

- **SCAN algorithm** called the **elevator algorithm**
  - https://www.popularmechanics.com/technology/infrastructure/a20986/the-hidden-science-of-elevators/

- Note
  - If requests are uniformly dense
    - largest density at other end of disk
    - and those wait the longest

SCAN #2



- Skews wait times non-uniformly

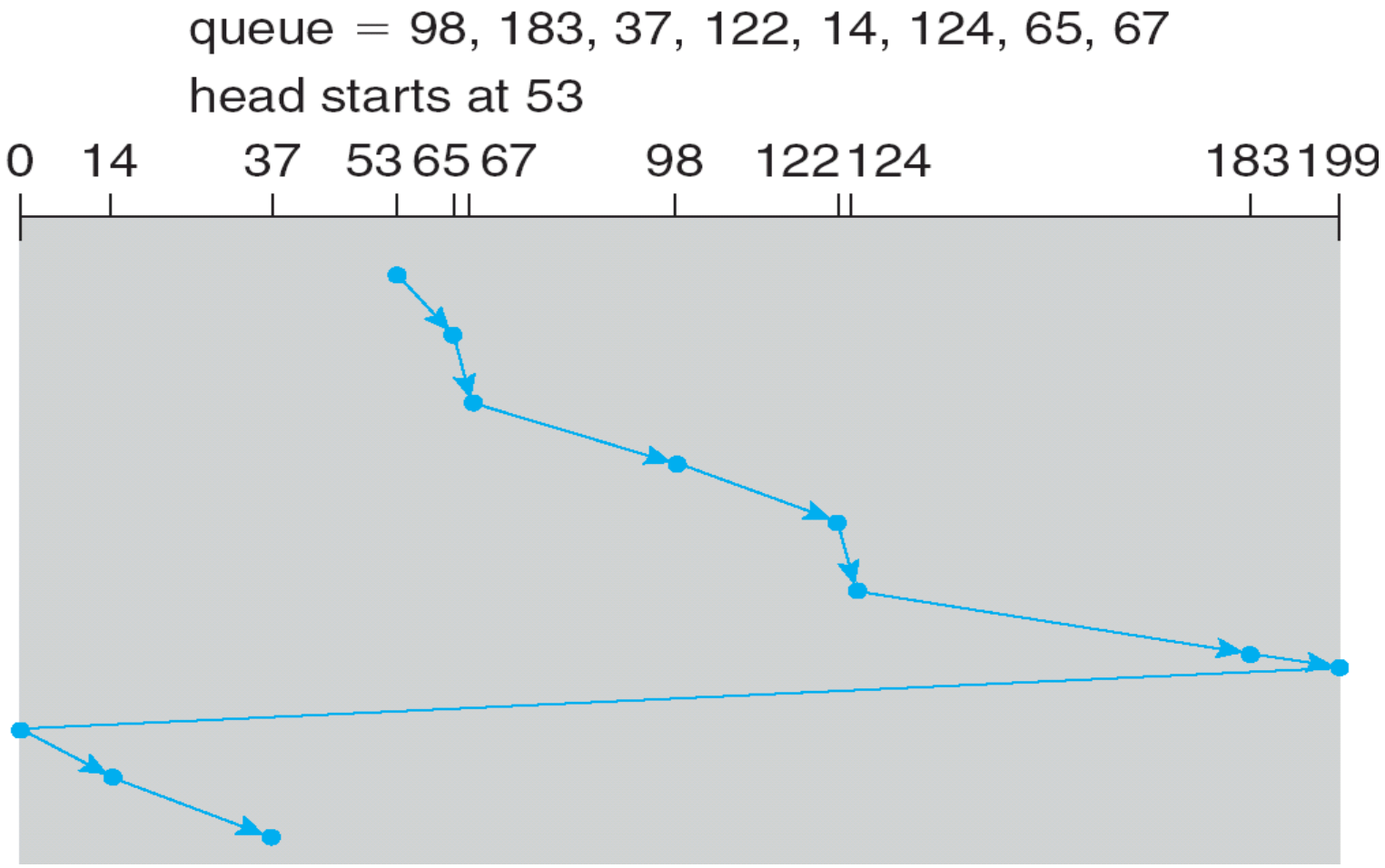
C-SCAN #1

- Provides a **more uniform wait time** than SCAN

- Head moves from **one end** of the disk **to the other**
  - Servicing requests as it goes
- When it reaches the **other end**
  - Immediately returns to the **beginning of the disk**
  - **Without servicing** any requests on the return trip

- Also known as **typewriter** algorithm

C-SCAN #2



- Uniform wait times

Selecting a Disk-Scheduling Algorithm

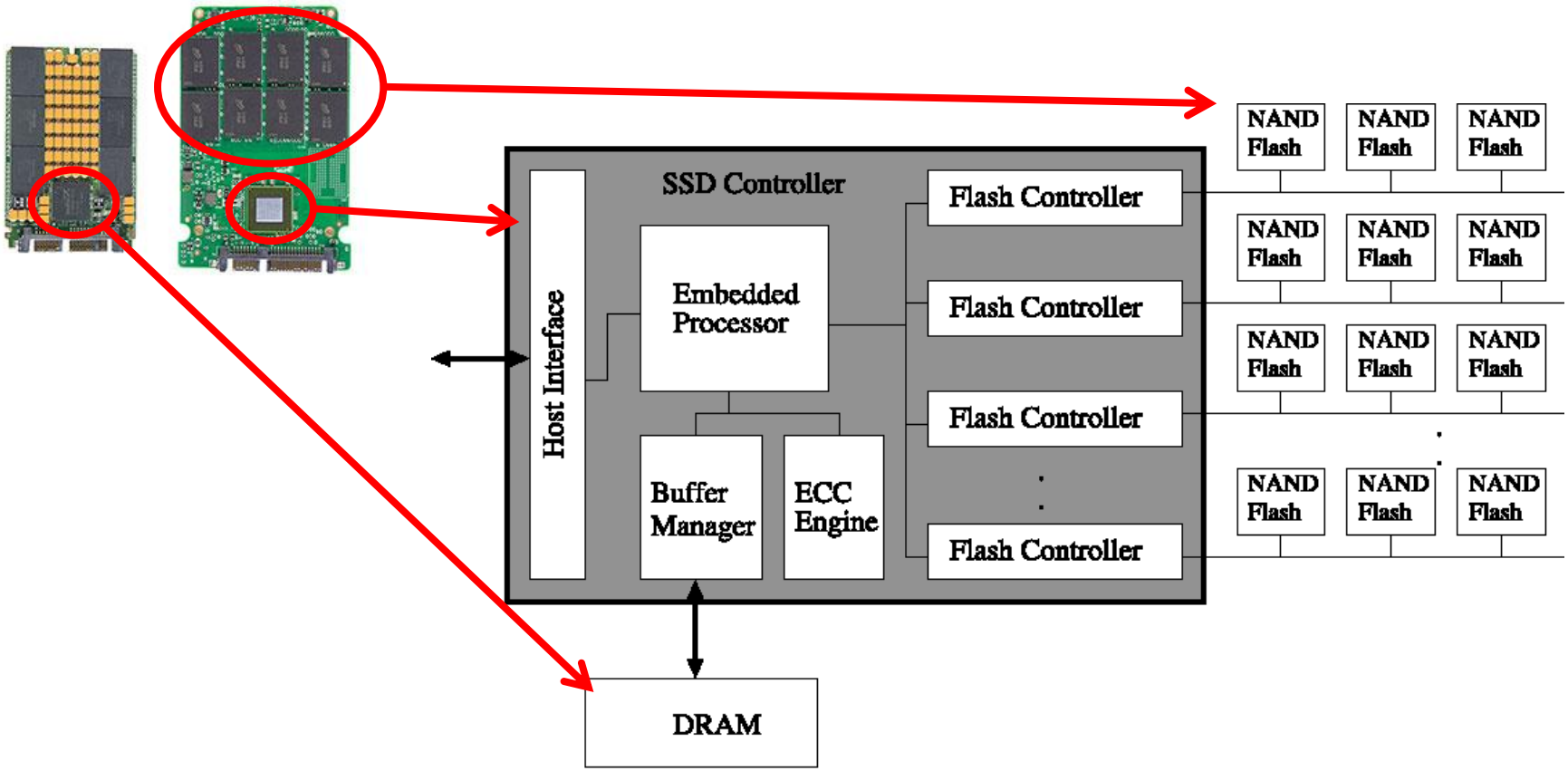
- When there is **one request** all algorithms behave like FCFS
- SCAN and C-SCAN perform better for systems with **heavy load** on the disk (less starvation)

- Performance depends on the **number and types of requests**
- Requests for disk service can be **influenced by**
  - File-allocation method
  - Metadata layout

- OS block-scheduling algorithm
  - **Module** of the OS, ease replacement
- **Linux**
  - **Deadline:** variation of C-SCAN with two queues
  - **NOOP:** variation of FCFS
  - **CFQ:** uses the concept of timeslices

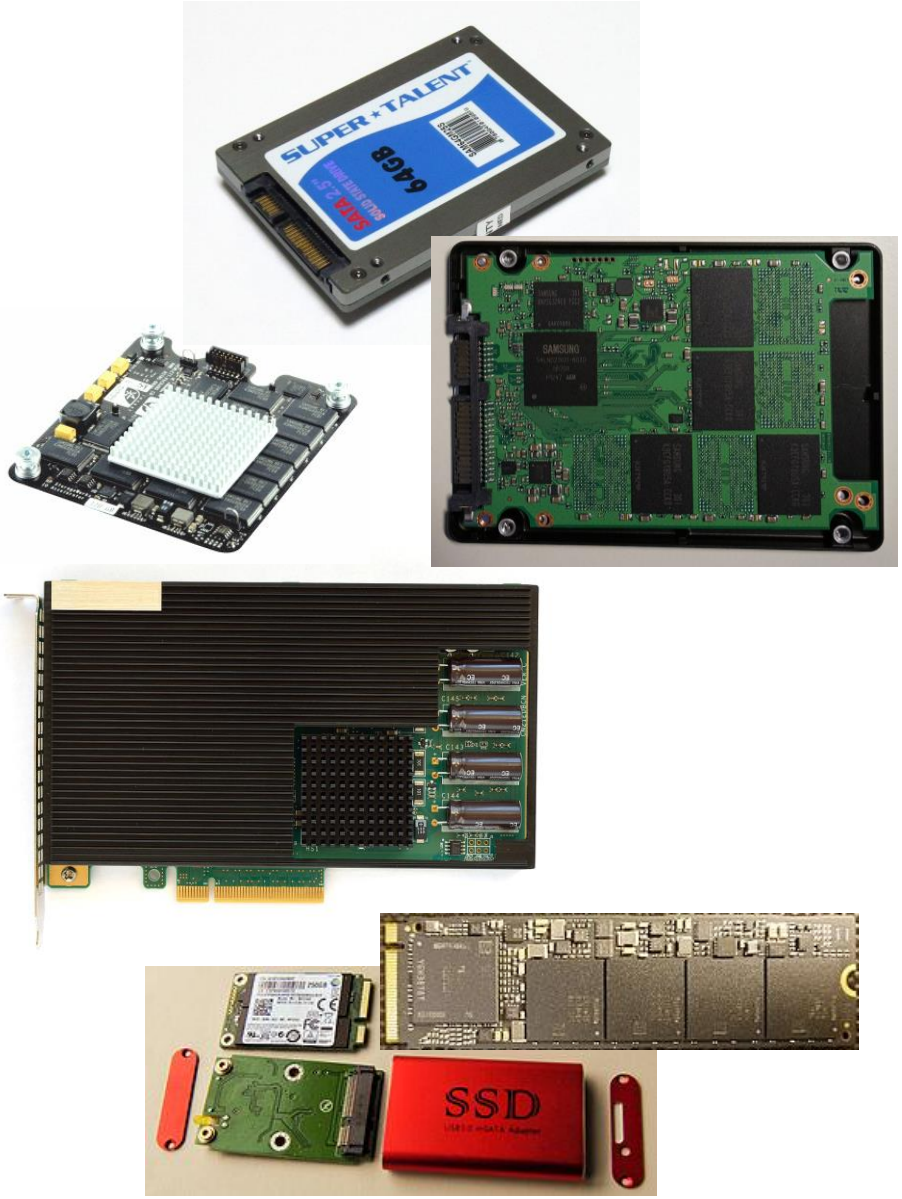
Solid-State Drives #1

Flash Disks



Solid-State Drives #2

- Different technologies
  - NOR
  - **NAND**
  - **3D XPoint**
  - Memristor
  - …
- Multiple interfaces
  - USB
  - SATA, mSATA
  - NVMe (M.2, PCIe)
  - …



SSD Reads

- Reads
  - Unit **of read is a *page***, typically 4kB
- COTS SSD handles
  - **~100k reads/s**
  - **10-100us** latency
    - 50-1000x better than magnetic disks
  - **50-5000 MB/s** read throughput
    - At least 1-10x better than magnetic disks
- Read access time is (mostly) **independent** of the device geometry
  - Block scheduler **is not needed**

SSD Writes

- Writes
  - Unit of **write is a page**
    - Lower writes/s than reads/s
    - Higher write latency than read latency
    - Lower throughput than read

- Flash media must be ***erased*** **before it can be written**
  - **Unit of erase is a block**, typically 64-256 pages
    - Takes ~1ms to erase a block (depends on manufacturing technology)
    - Can only be erased a certain number of times before unusable
      - Typically 10,000 – 1,000,000 times
  - Write amplification

- To extend lifetime require **Flash Translation Layer** (FTL)
  - Implemented in firmware
  - Wear leveling

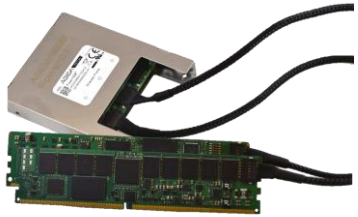
SSD vs HDD

- **Capacity** (March 2020, in March 2023 is half this price)
  - Flash SSD costs at min 0.1gbp/GB
    - 1TB drive costs around 100gbp (cheap models)
  - 1TB hard drive costs around 35gbp
- **Energy**
  - SSD is typically more energy efficient than a hard drive
    - 1-2 watts to power an SSD
    - ~10 watts to power a hard disk drive
- **Physical resistance**
  - SSD has no moving parts
  - Hard disk drive cannot work correctly if subject to physical acceleration

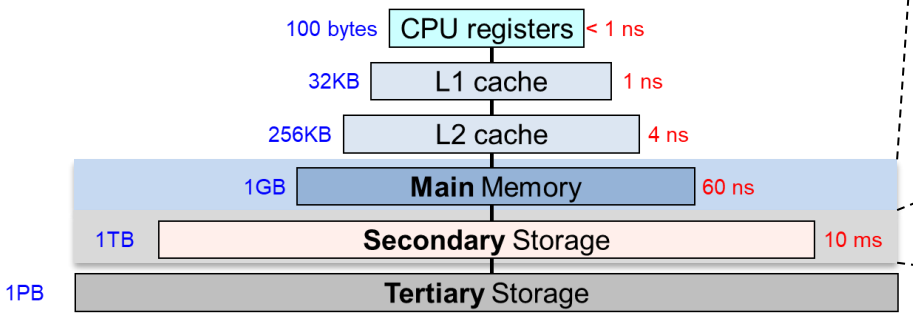
Technology Update

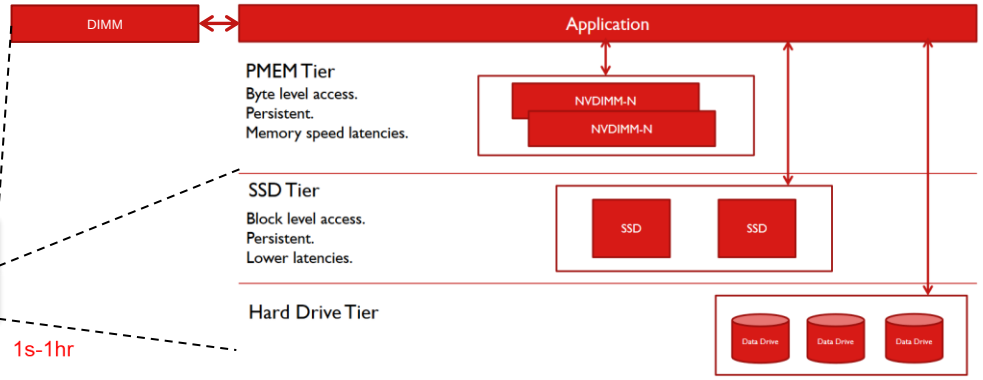
- Everyone **wants**
  - **Faster** (secondary) storage, as fast as main memory
  - **Larger** main memory, as large as (secondary) storage
  - **Persistent**

- **Non-volatile Memory** (NVM)/**Persistent Memory** (PM, PMem)
  - Non-volatile/persistent, i.e., survives reboots
  - Main memory form factor, looks like main memory (NVDIMM-x)
  - Technology used: battery-backed, 3D XPoint, PCM, etc.
  - Examples
    - Intel Optane Persistent Memory (NVDIMM-P) https://www.intel.co.uk/content/www/uk/en/architecture-and-technology/optane-dc-persistent-memory.html
    - Dell NVDIMM (NVDIMM-N) https://downloads.dell.com/solutions/general-solution-resources/White%20Papers/NVDIMM-N%20on%20Dell%20PowerEdge%20servers%20and%20VMware%20ESXi.pdf

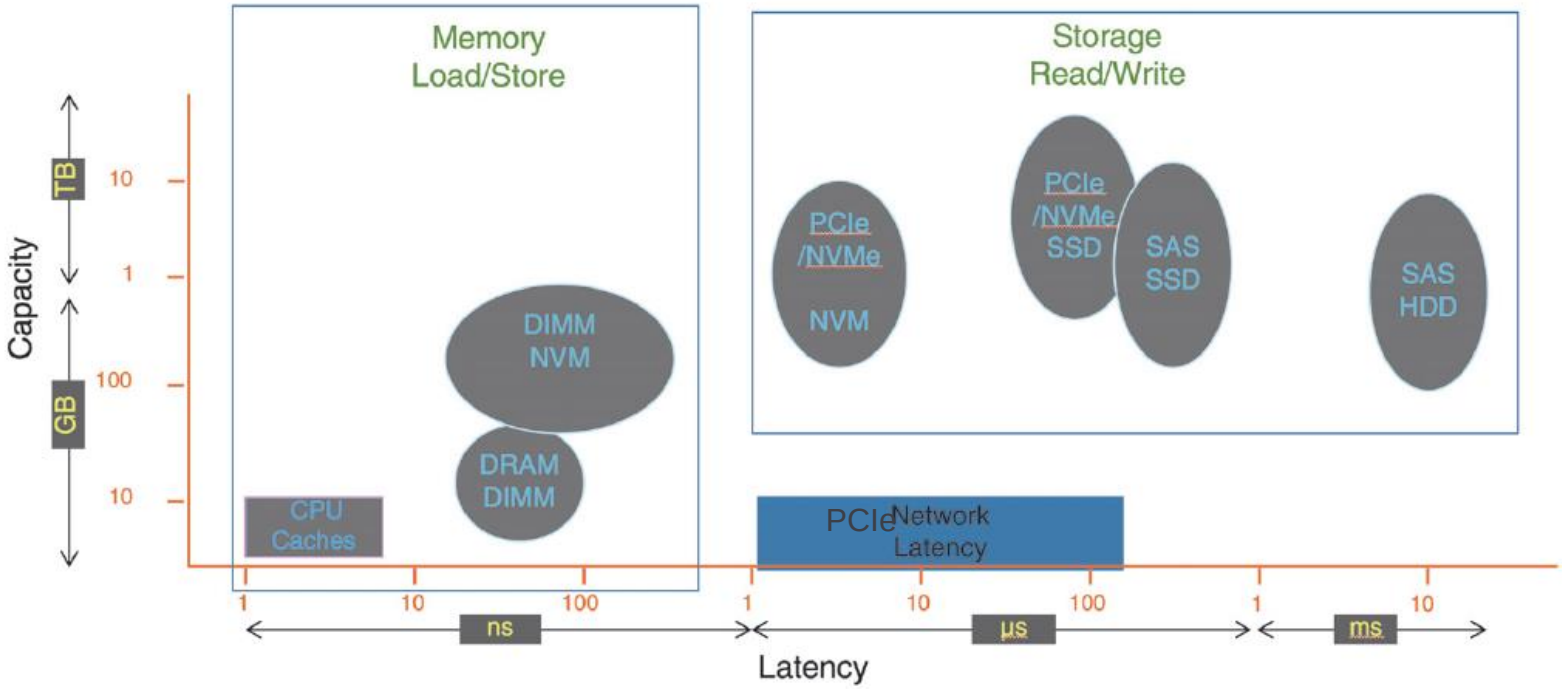


New and Old **Secondary Storage** vs **Primary Storage**



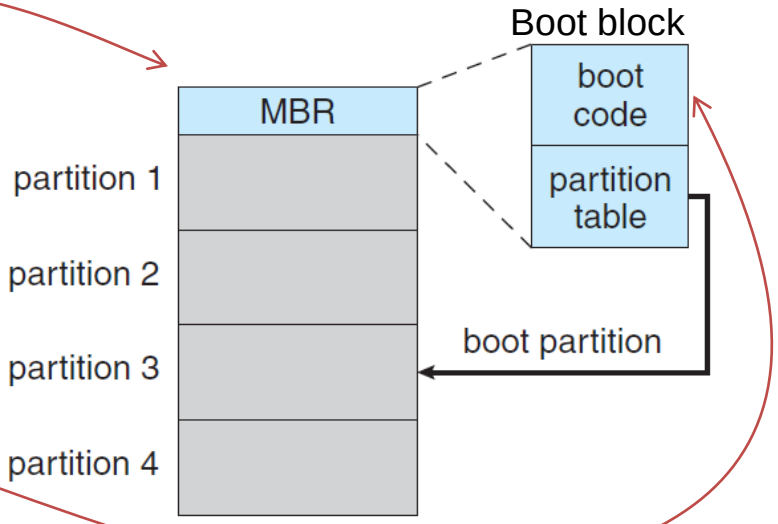


https://www.snia.org/sites/default/files/SSSI/NVDIMM%20-%20Changes%20are%20Here%20So%20What's%20Next%20-%20final.pdf



Storage Device Management #1

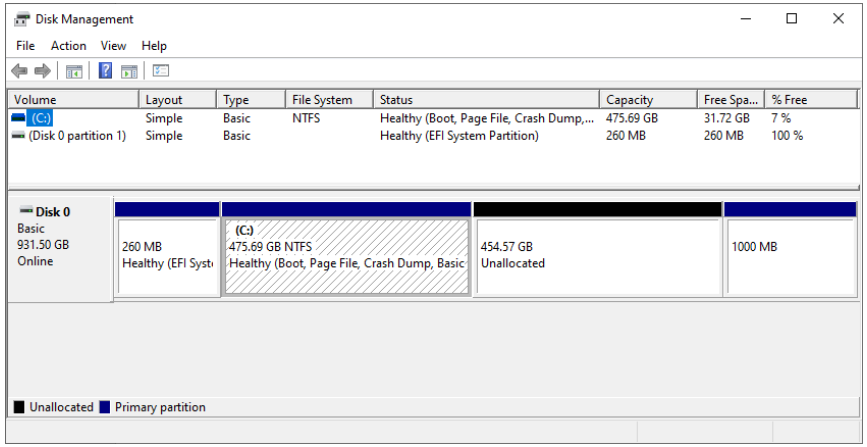
- **Storing the data** on the device is not enough
  - Need metadata
- Before storing the data, device **needs to be initialized**
  - Low-level formatting
  - For each partition
    - Volume creation (lvm2)
    - Logical formatting (file system)



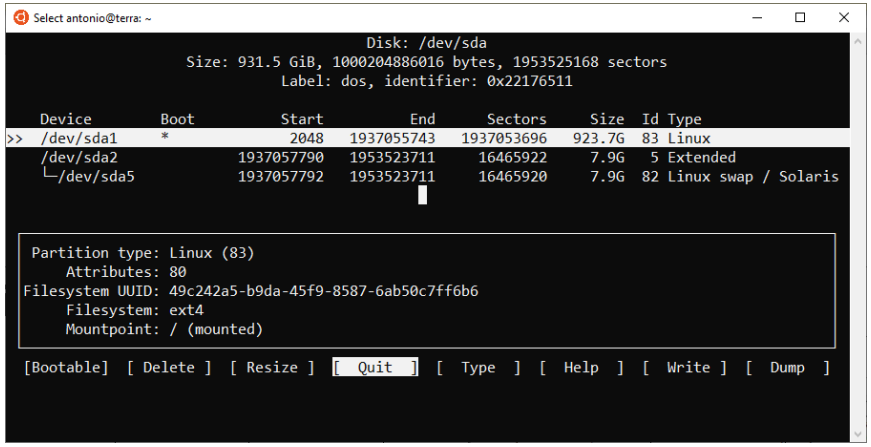
- **Booting**
  1. Firmware, or BIOS
  2. Reads code in MBR
  3. MBR also contain partition table
  4. Code in MBR reads boot sector of the selected partition
  5. Pass control to code in selected partition

Storage Device Management #2

- Windows Disk Management



- Linux cfdisk



- This is the content of the MBR
  - The space MBR occupies not shown
- Different type of partitions
  - Primary (e.g., /dev/sda1)
    - **A single** logical partition
  - Extended (e.g., /dev/sda2)
    - **Multiple** logical partitions
- Each **logical partition** includes either
  - File system
  - Special use (e.g., swap)

https://www.howtogeek.com/184659/beginner-geek-hard-disk-partitions-explained/