

# COMP 1553

## Computer Forensics 2

---

### Storage Devices and Operating System Intro

Tuan Vuong  
Computing & Information Systems  
University of Greenwich



# Contact detail

---

- Name: Tuan VUONG
- Email: t.p.vuong@gre.ac.uk
  - please use your University email account
- Tel: 020 8331 8388
  - but best to use email
- Office: Queen Mary 362
- Office hours :
  - Monday 2-3pm, Thursday 11am-12pm

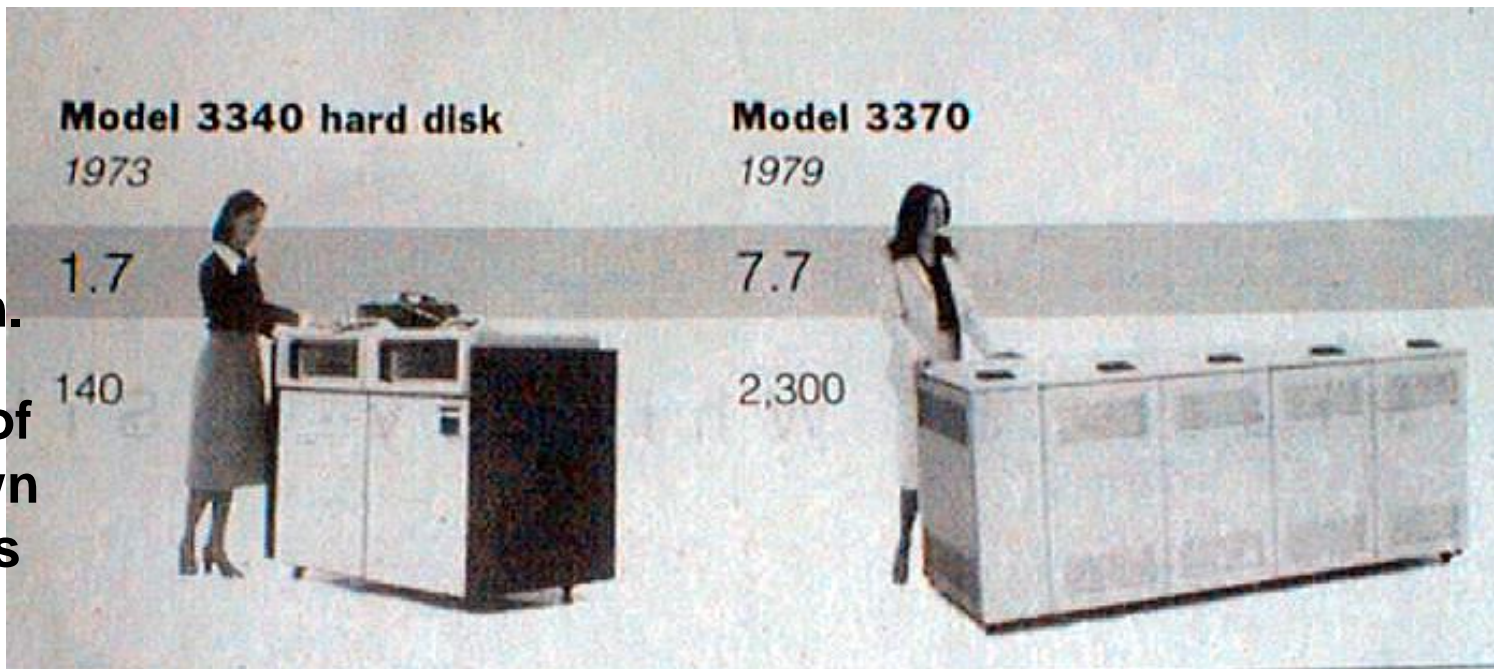
# Contents

---

- Data storage and devices
- Introduction to OSs



# Disk History

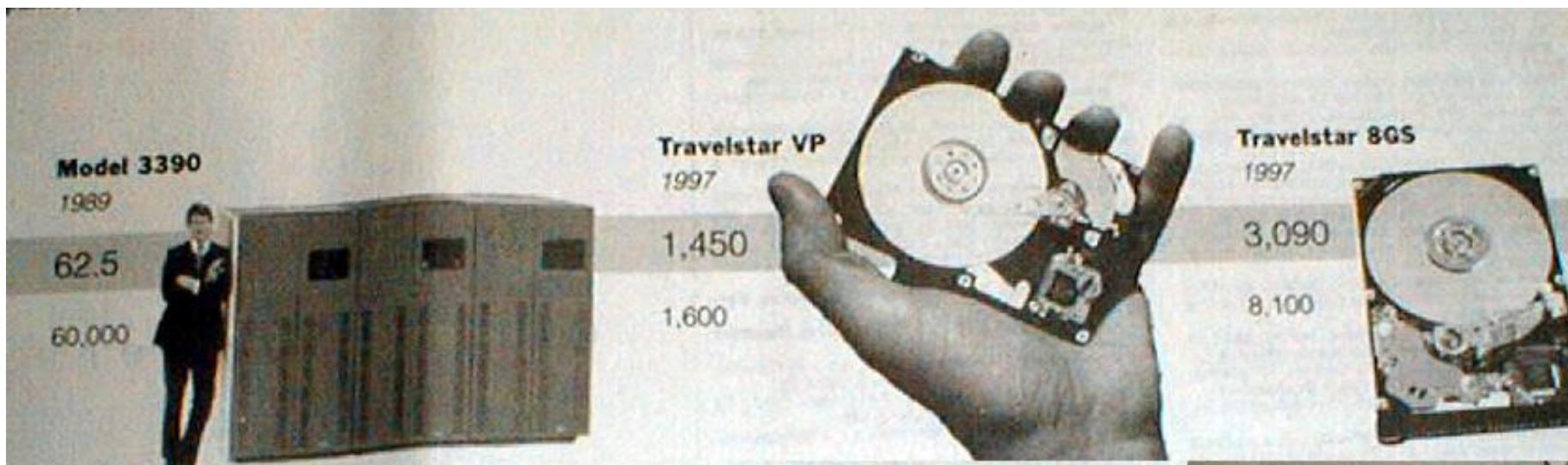


**1973:**  
**1.7 Mbit/sq. in**  
**140 MBytes**

**1979:**  
**7.7 Mbit/sq. in**  
**2,300 MBytes**

# Disk History

---



**1989:**  
**63 Mbit/sq. in**  
**60,000 MBytes**

**1997:**  
**1450 Mbit/sq. in**  
**2300 MBytes**

**1997:**  
**3090 Mbit/sq. in**  
**8100 MBytes**

# Storage Technology Drivers

---

- Driven by the prevailing computing paradigm
  - 1950s: migration from batch to on-line processing
  - 1990s: migration to ubiquitous computing
    - computers in phones, books, cars, video cameras, ...
    - nationwide fiber optical network with wireless tails
- Effects on storage industry:
  - Embedded storage
    - smaller, cheaper, more reliable, lower power
  - Data utilities
    - high capacity, hierarchically managed storage



# A safe place for data

- **Memory:** the storage area in which programs are kept when they are running and which contains the data needed by the running programs.
- Memory on motherboard is **volatile** – it loses power, it forgets.
  - Sometimes called primary or main memory
  - DRAM: common primary memory
- Memory that stores programs between runs is called secondary memory
- Magnetic disks (hard disks): common secondary memory (non-volatile)
- Organized as a collection of platters, rotating on a spindle at constant speed
- Platters are covered with magnetic recording material, similar to that found on cassettes or videotapes.
- To read and write information on a hard disk, a *movable arm* containing a small electromagnetic coil called a *read/write head*
- Hard disk diameters: from 1 to 3.5 inches. The widest disks have the highest performance, the smallest have the lowest cost and the best cost per megabyte is usually a disk in between.
- About 5-15 **ms** for data access in hard disks compared to 40-80 **ns** for DRAMs (due to the use of mechanical components in hard disks).
- However, the cost of a hard disk is much smaller (more than 100 times) than DRAM for the same storage capacity.

# A safe place for data (contd.)

---

- Optical disks including CDs and DVDs
  - discussed in the next slide
- Magnetic tapes
  - slow serial access
  - used mainly for backups
  - replaced recently by duplicate hard drives
- FLASH-based removable memory cards
  - nonvolatile semiconductor memory
  - typically connected by a Universal Serial Bus (USB) connection
- Floppy drives and Zip drives
  - a version of magnetic disk technology with removable disks
- Can you think of any other?



# CDs and DVDs

- 
- In a CD data is recorded in a spiral fashion with individual bits being recorded by burning small pits (approximately 1 micron in diameter) into the disk surface
  - Disk is read by shining a laser at the CD surface and examining the reflected light to see whether there is a pit or a flat surface
  - DVDs use the same approach, with the addition that there are multiple layers that the laser beam can be focused on
  - Rewritable CDs and DVDs use a different recording surface that has a crystalline reflective material
  - Pits are formed that are not reflective in a manner similar to write-once CD or DVD
  - To erase the CD or DVD, the surface is heated and cooled slowly to restore the surface to its crystalline structure

# Magnetic Disk

---

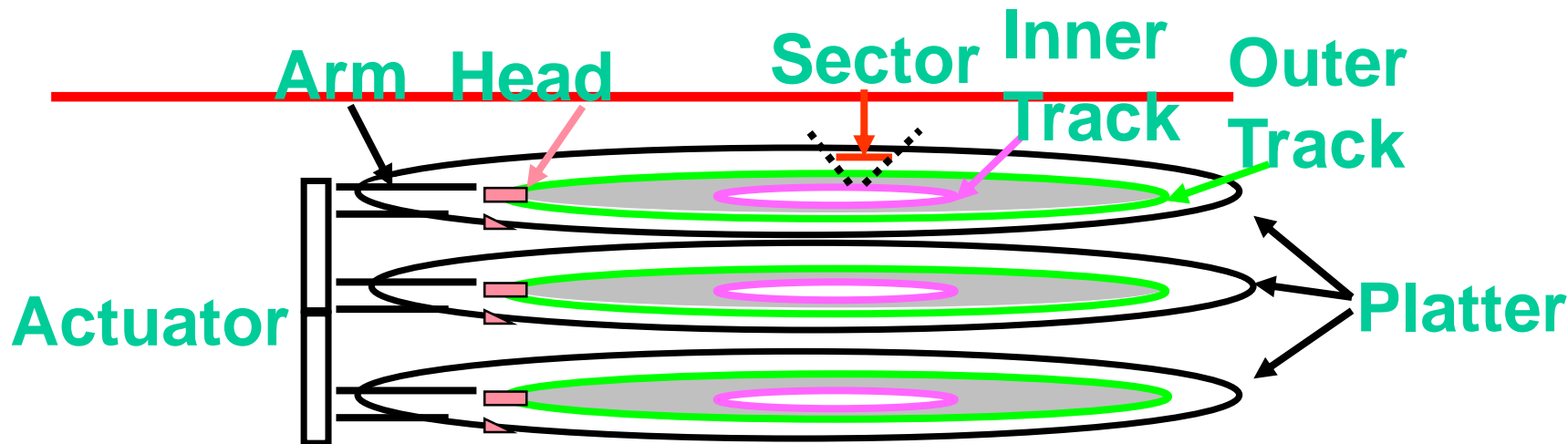
- Purpose:
  - Long term, nonvolatile storage
  - Large, inexpensive, and slow
  - Rely on a rotating platter coated with a magnetic surface
  - Consists of a collection of platters (1-4), each of which has two recordable disk surfaces
  - Each disk surface is divided into concentric circles, called **tracks**.
  - Use a moveable read/write head to access the disk
- Typical numbers (depending on the disk size):
  - Typical diameter ranges from 1 to 3.5 in
  - 10,000 to 50,000 tracks (concentric cycles) per surface
  - 100 to 500 sectors per track
    - A sector is the smallest unit of information that can be read or written

# Characteristics of a Magnetic Disk

---

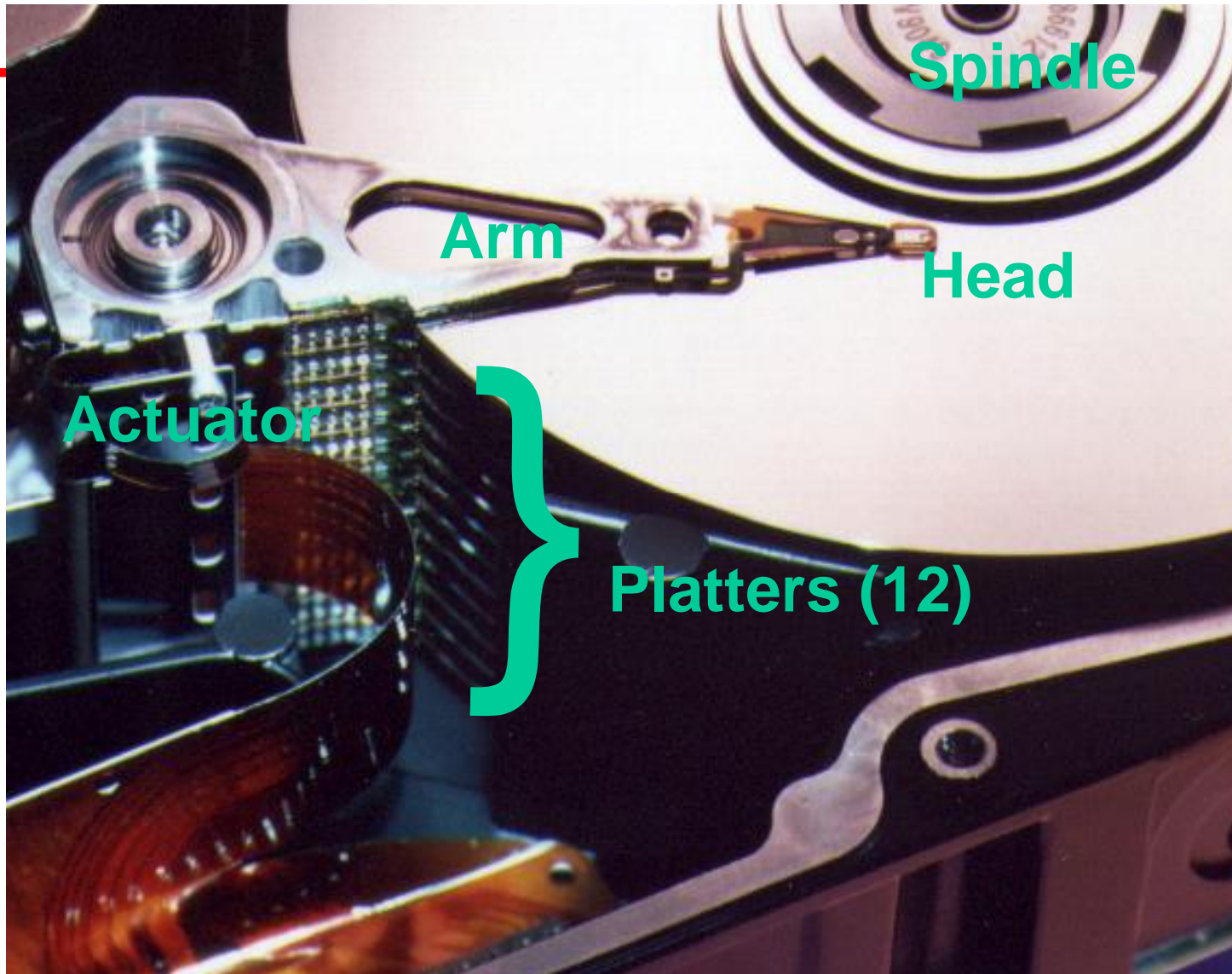
- To access data, the operating system must direct the disk through a 3-stage process.
  - First step: Position the head over the proper track (**seek operation**). The time to move the head to the desired track is called the *seek time* (disk manufacturers report minimum seek time, maximum seek time and average seek time in their manuals)
  - Rotational latency: time for the desired sector to rotate under the read/write head -> ***usually assumed to be equal to half of the total rotation time***
  - Last component of a disk access: Transfer time-> the time to transfer a block of bits (sector) under the read-write head
- Average seek time as reported by the industry:
  - Typically in the range of 8 ms to 12 ms
  - Defined as **(Sum of the time for all possible seeks) / (total # of possible seeks)**
- Due to locality of disk references, actual average seek time may:
  - Only be 25% to 33% of the advertised number

# Typical structure

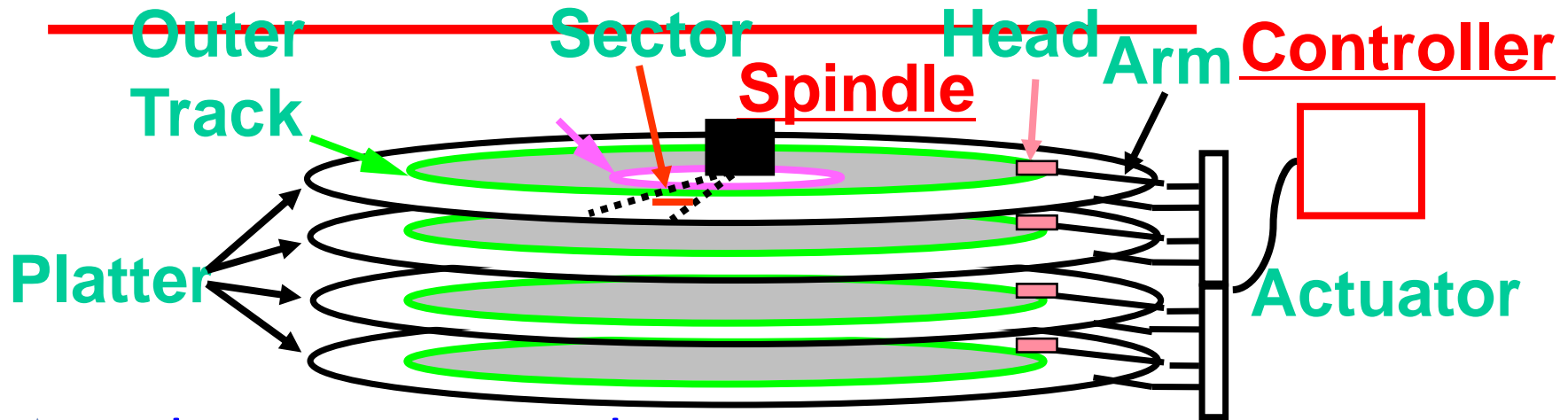


- Several platters, with information recorded magnetically on both surfaces (usually)
- Bits recorded in tracks, which in turn divided into sectors (e.g., 512 Bytes)
- Actuator moves head (end of arm, 1/surface) over track ("seek"), select surface, wait for sector rotate under head, then read or write
  - "Cylinder": all tracks under heads

# What does an HDD looks like?



# Disk Latency?



⦿ **Disk Latency = Seek Time + Rotation Time + Transfer Time + Controller Overhead**

- ⦿ Seek Time? depends no. tracks move arm, seek speed of disk
- ⦿ Rotation Time? depends on speed disk rotates, how far sector is from head
- ⦿ Transfer Time? depends on data rate (bandwidth) of disk (bit density), size of request

# Hard Disks

- All information stored on a hard disk is recorded in tracks, which are concentric circles placed on the surface of each platter, much like the annual rings of a tree. The **tracks are numbered**, starting from zero, **starting at the outside of the platter** and increasing as you go in.
- A modern hard disk has tens of thousands of tracks on each platter.





# Rotational Latency

---

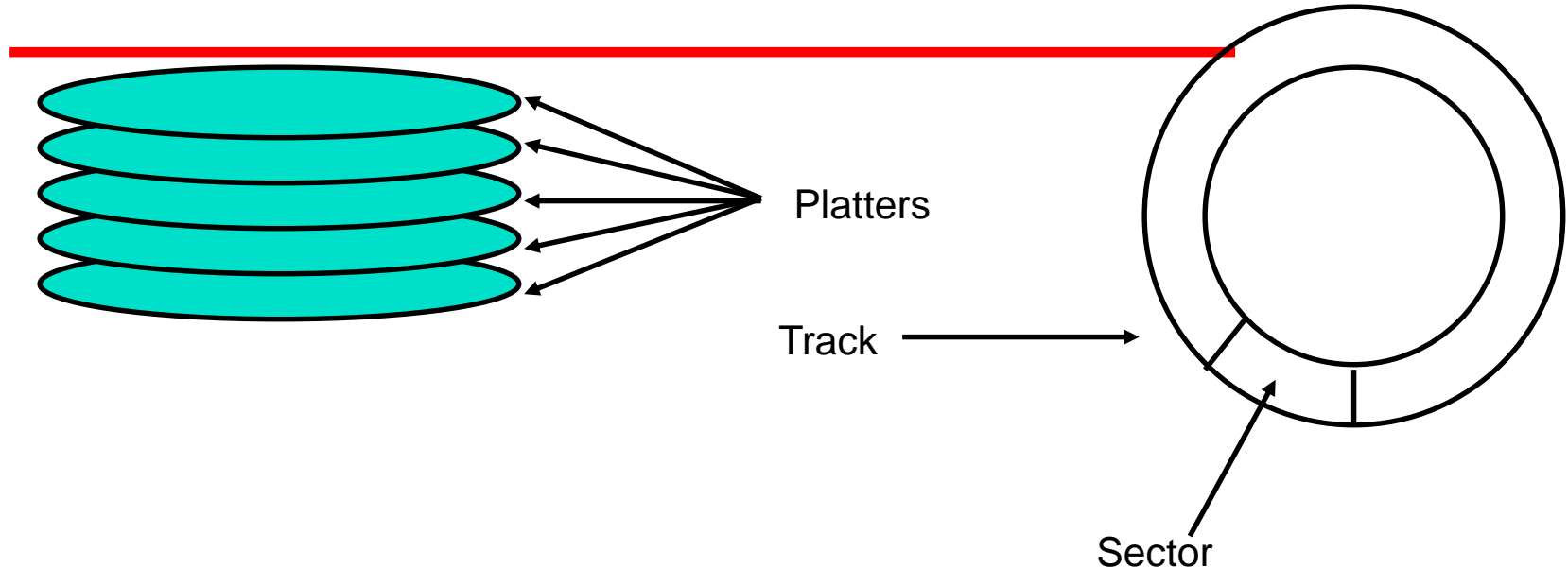
- Rotational Latency:
  - Most disks rotate at 5400 to 15000 RPM (revolutions per minute)
  - Approximately 11.2 ms to 4 ms per revolution, respectively
  - *Average latency to the desired information is assumed to be halfway around the disk:*  
5.6 ms at 5400 RPM, 2 ms at 15000 RPM
- Transfer Time is a function of :
  - Sector size
  - Rotation speed
  - Recording density: bits per inch on a track



# Example

- 
- What is the average time to read or write a 512-byte sector, for a typical disk rotating at 10,000 RPM? Advertised seek time is 6 ms, transfer rate is 50 MB/sec, controller overhead is 0.2 ms, the disk is idle so there is no waiting time
  - Disk Access Time = Seek time + Rotational Latency + Transfer time  
+ Controller Time + waiting Delay
  - 512 bytes = 0.5 KB
  - Disk Access Time = 6 ms + 0.5 rotations / 10000 RPM + 0.5 KB / 50 MB/s + 0.2 ms + 0
  - Disk Access Time = 9.2 ms
  - If real seek time is 25% of the advertised one, then seek time = 1.5 ms and disk access time = 4.7 ms, with rotation delay being the largest component

# Platters Tracks Sectors



- ❁ Traditionally all tracks have the same number of sectors
- ❁ Recently relaxed: varying number of sectors, speed varies with track location

# Availability - MTTF

---

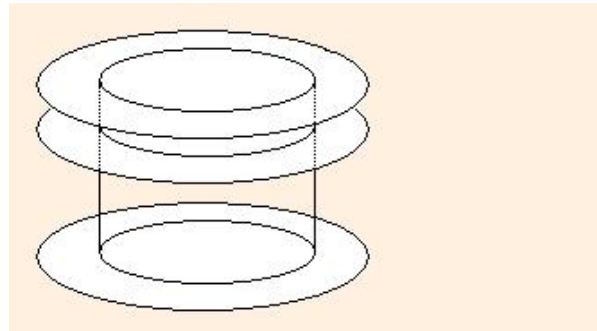
- A system can alternate between two states of delivered service:
  1. Service accomplished: service is delivered as specified
  2. Service interruption: service is different from specification
- Transition from 1 to 2: failure
- Transition from 2 to 1: restoration
- Reliability: a measure of continuous service accomplishment
- Mean time to failure (MTTF) is a reliability measure (e.g., in the specifications of a magnetic disk a manufacturer may state that the MTTF is 600000 hours).
- Service interruption is measured as mean time to repair (MTTR)
- Availability is a measure of the service accomplishment with respect to the alternation between the two states of accomplishment and interruption
- $\text{Availability} = \text{MTTF} / (\text{MTTF} + \text{MTTR})$

# Hard Disks (contd.)

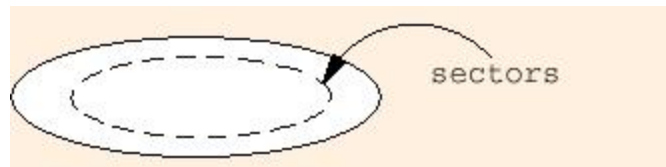
- 
- Remember: Data is accessed **by moving the heads from the inner to the outer part of the disk**, driven by the head actuator.
  - Each track can hold many thousands of bytes of data.
  - Each track is broken into smaller units called **sectors**.
  - Each sector holds 512 bytes of user data, plus as many as a few dozen additional bytes used for internal drive control and for error detection and correction.

# Cylinders

- Corresponding tracks on all surfaces on a drive, when taken together, make up a **cylinder**:



- Since an individual data block is one **sector** of a track:



# Hard Disks (contd.)

---

- **IDE** (Intelligent Drive Electronics) disk drives pretend to have a far different geometry than they actually have physically. For instance, a typical hard drive might tell the operating system that it has 16,384 cylinders, 80 heads and 63 sectors per track;
- Since 80 heads implies at least 40 platters, each with 2 surfaces, we see immediately that this cannot represent the physical contents of a hard drive which is just 3/4 of an inch thick! In fact, most hard drives have only one or two platters.
- **Serial Advanced Technology Attachment SATA and SATA, 1.5 – 3 and eSATA** (150MB/s – 300MB/s)



# Hard Disks (contd.)

- 
- The computation of the hard drive capacity then proceeds as follows:
  - Bytes in 1 disk = cylinders \* heads \* tracks \* sectors \* bytes per sector =
  - (16,384 cylinders / disk) \* (80 heads / cylinder) \* (1 track / head) \* (63 sectors / track) \* (512 bytes / sector) = 42,278,584,320 bytes
  - Remember your Bytes! This is a large and rather inconvenient number. We adopt the following "metric" prefixes **when discussing storage space**:
  - 1 **Kilobyte (KB)** =  $2^{10}$  bytes = 1,024 bytes
  - 1 **Megabyte (MB)** =  $2^{20}$  bytes = 1,048,576 bytes = 1,024 KB
  - 1 **Gigabyte (GB)** =  $2^{30}$  bytes = 1,073,741,824 bytes = 1,048,576 KB = 1,024 MB
  - 1 **Terabyte (TB)** =  $2^{40}$  bytes = 1,099,511,627,776 bytes = 1,073,741,824 KB = 1,048,576 MB = 1,024 GB

# Remember

---

- Average distance sector from head?
- 1/2 time of a rotation
  - 10000 Revolutions Per Minute  $\Rightarrow$  166.67 Rev/sec
  - 1 revolution =  $1 / 166.67$  sec  $\Rightarrow$  6.00 milliseconds
  - 1/2 rotation (revolution)  $\Rightarrow$  3.00 ms
- Average no. tracks move arm?
  - Sum all possible seek distances from all possible tracks / # possible
    - Assumes average seek distance is random
  - Disk industry standard benchmark

# Recap: Response time?

## Purpose:

- Long-term, nonvolatile storage
- Large, inexpensive, slow level in the storage hierarchy

## Characteristics:

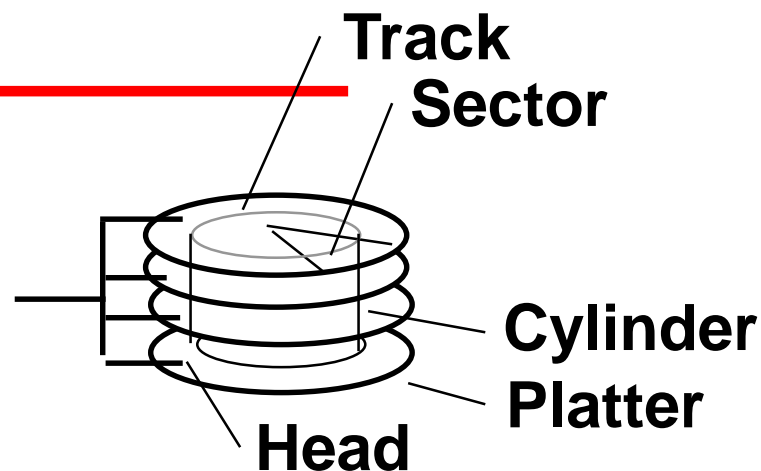
- Seek Time (~8 ms avg)
  - positional latency
  - rotational latency

## Transfer rate

- 10-40 MByte/sec
- Blocks

## Capacity

- Gigabytes
- Quadruples every 2 years



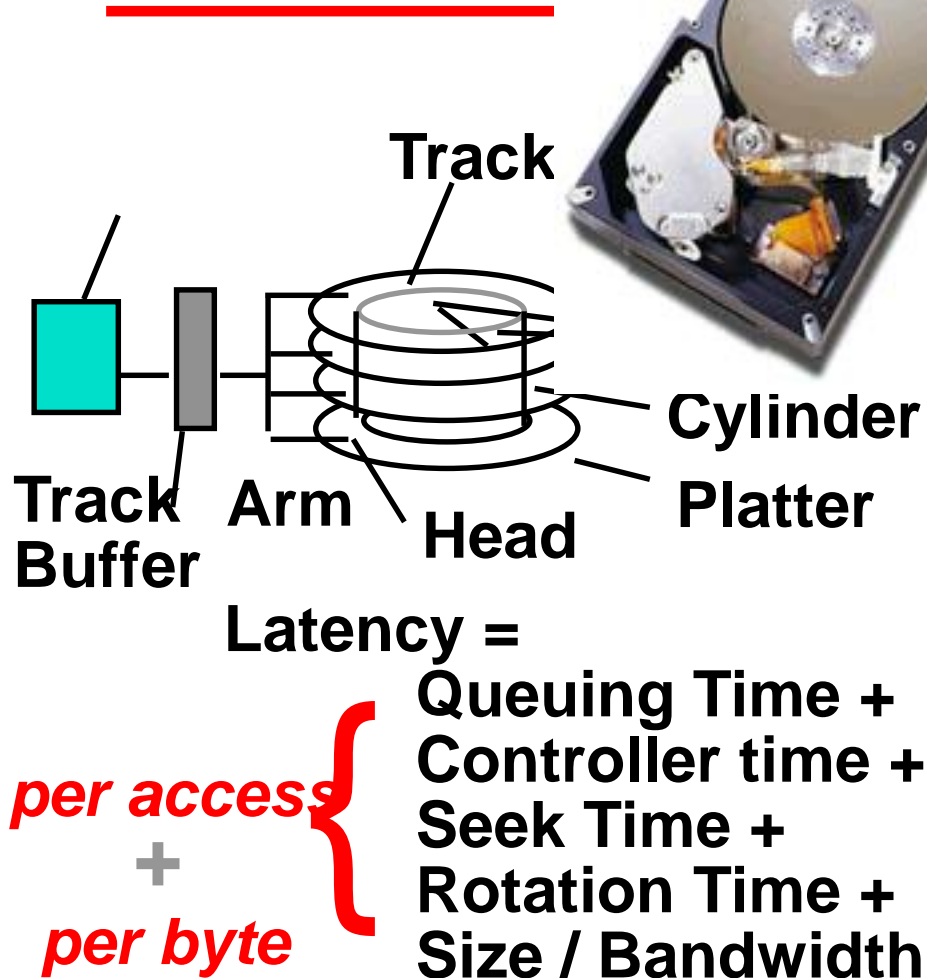
7200 RPM = 120 RPS => 8 ms per rev  
ave rot. latency = 4 ms  
128 sectors per track => 0.25 ms per sector  
1 KB per sector => 16 MB / s

## Response time

= Queue + Controller + Seek + Rot + Xfer

Service time

# A real life example



- 181.6 GB, 3.5 inch disk
- 12 platters, 24 surfaces
- 24,247 cylinders
- 7,200 RPM; (4.2 ms avg. latency)
- 7.4/8.2 ms avg. seek (r/w)
- 64 to 35 MB/s (internal)
- 0.1 ms controller time
- 10.3 watts (idle)

# Test your knowledge

---

- Calculate time to read 64 KB (128 sectors) for Barracuda 180 X using advertised performance; sector is on outer track

Disk latency = average seek time + average rotational delay + transfer time + controller overhead

$$= 7.4 \text{ ms} + 0.5 * 1/(7200 \text{ RPM})$$

$$+ 64 \text{ KB} / (65 \text{ MB/s}) + 0.1 \text{ ms}$$

$$= 7.4 \text{ ms} + 0.5 / (7200 \text{ RPM} / (60000 \text{ ms/M}))$$

$$+ 64 \text{ KB} / (65 \text{ KB/ms}) + 0.1 \text{ ms}$$

$$= 7.4 + 4.2 + 1.0 + 0.1 \text{ ms} = 12.7 \text{ ms}$$

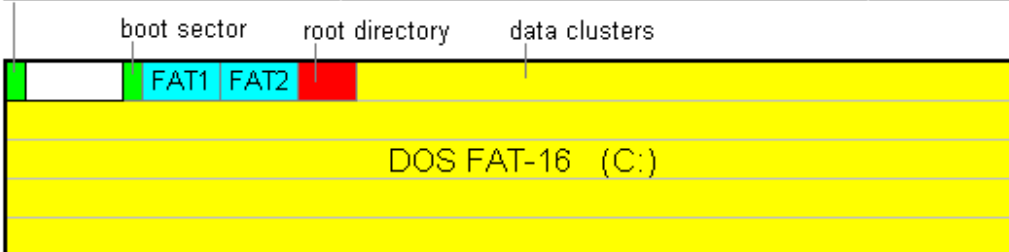
# Hard Disks (contd.)

- All hard disks on all IBM compatible computers have the same way of partitioning.
- First sector of the disk, called MBR (Master Boot Record), contains the partition table.
- This table has four records, each of them can describe one partition.
- In the simplest case we would have all disk space assigned to one partition, like in the following example

**Example 1**      Hard disk 340M [ 665 cyl x 16 heads x 63 sects ]

**MBR**

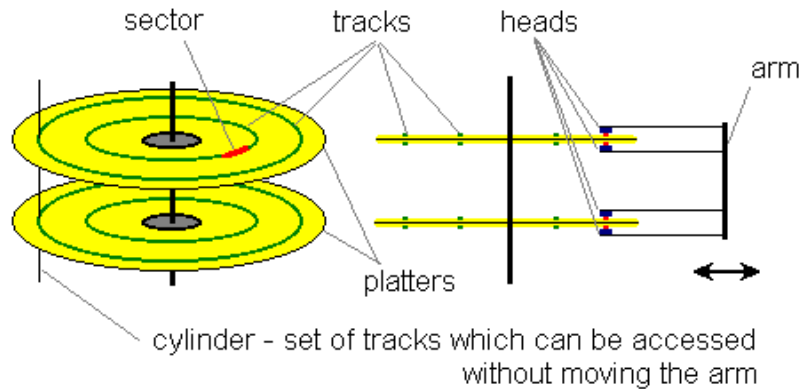
#	Partition Type	Starting			Ending			Size[K]
		Cyl	Side	Sect	Cyl	Side	Sect	
1	DOS FAT-16	0	1	1	664	15	63	335,128
2	Unused	0	0	0	0	0	0	
3	Unused	0	0	0	0	0	0	
4	Unused	0	0	0	0	0	0	

The diagram illustrates the disk layout. It shows a horizontal bar representing the disk. The first sector is the boot sector (green). This is followed by FAT1 (cyan) and FAT2 (cyan). Then comes the root directory (red). The rest of the disk is filled with data clusters (yellow). The label 'DOS FAT-16 (C:)' is centered over the yellow area.

Note that MBR occupies one sector at cylinder 0, side 0, sector 1 and partition starts on the cylinder 0, side 1, sector 1. The 62 sector gap between them was left unused, because we want all partitions to start at the cylinder boundary or, at least, on the side boundary.

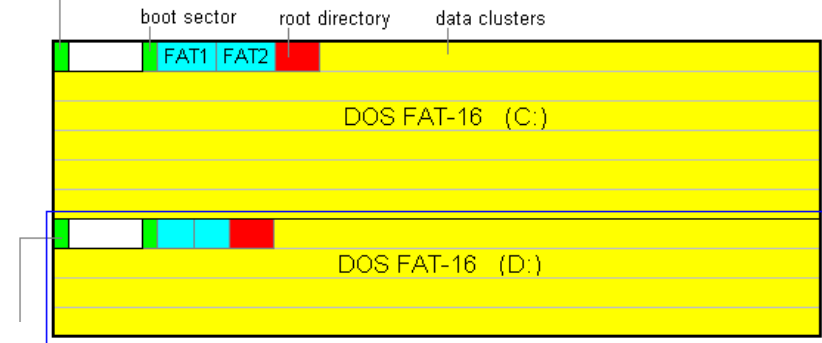
# Hard Disks and Operating Systems



**Example 2** Hard disk 340M [ 665 cyl x 16 heads x 63 sects ]

## MBR

#	Partition Type	Starting			Ending			Size[K]
		Cyl	Side	Sect	Cyl	Side	Sect	
1	DOS FAT-16	0	1	1	399	15	63	200,568
2	Unused	0	0	0	0	0	0	
3	Unused	0	0	0	0	0	0	
4	DOS Extended	400	0	1	664	15	63	133,560

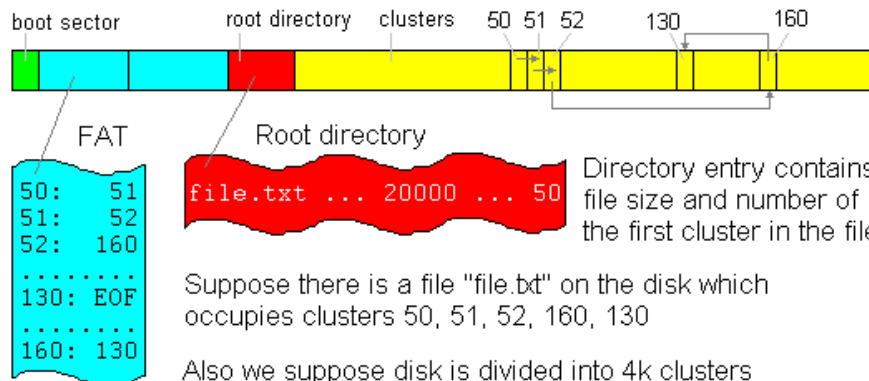


## Extended partition

## EMBR 1

#	Partition Type	Cyl	Side	Sect	Cyl	Side	Sect	Size[K]
1	DOS FAT-16	400	1	1	664	15	63	133,528
2	Unused	0	0	0	0	0	0	

## FAT-16





# Operating system and File System

---

- **Operating System** and **File System** are different things, which many people unfortunately use interchangeably.
- So a FAT-16 could have MS-DOS 6.22 installed on it, it could be Windows 95 or NT, or it could be all three of them installed in the different directories in the same partition. We could even have Linux in the same partition (not advised to do so!).
- **FAT-16? FAT-32?** The name of the file system is FAT-16 because it has FAT (File Allocation Table) and also because each entry in FAT is 16-bits long (2 bytes). This means that FAT-16 partition cannot have more than 65,535 clusters ( $2^{16} = 65,536$ ). Similarly FAT-32 has 32-bit entries and could address up to  $2^{32}$  clusters. (Actually they use only 28 bits).


# Clusters and capacity

Cluster	File system type		
size	FAT-12	FAT-16	FAT-32
2K	8M	128M	512G
4K	16M	256M	1024G
8K	32M	512M	2048G
16K	64M	1G	2048G
32K	128M	2G	2048G
Partition size	Recommended file system / cluster size		
1-16M	FAT-12 / 4K		
16-256M	FAT-16 / 4K		
256-512M	FAT-16 / 8K		
512M-1G	FAT-16 / 16K or FAT-32 / 4K		
1-8G	FAT-32 / 4K		
8G and up	FAT-32 / 8K		

# Flash disks

---

## Compact Flash Cards

- Intel Strata Flash
  -  32 Mb in 1 square cm. (.6 mm thick)
- 100,000 write/erase cycles.
- Standby current = 100uA, write = 45mA
- Compact Flash 256MB~£12 512MB~£24
- Transfer @ 3.5MB/s (USB dependant)

## IBM Microdrive 1G~370

- Standby current = 20mA, write = 250mA
- Efficiency advertised in wats/MB

## VS. Disks

- Nearly instant standby wake-up time
- Random access to data stored
- Tolerant to shock and vibration (1000G of operating shock)

# Tapes

- 
- • Longitudinal tape uses same technology as
    - hard disk; tracks its density improvements
  - Disk head flies above surface, tape head lies on surface
  - Disk fixed, tape removable
  - • Inherent cost-performance based on geometries:
    - fixed rotating platters with gaps
    - (random access, limited area, 1 media / reader)
  - vs.
    - removable long strips wound on spool
    - (sequential access, "unlimited" length, multiple / reader)
  - • Helical Scan (VCR, Camcoder, DAT)
    - Spins head at angle to tape to improve density

# Tape drawbacks

---

- Tape wear out:
  - Helical 100s of passes to 1000s for longitudinal
- Head wear out:
  - 2000 hours for helical
- Both must be accounted for in economic / reliability model
- Bits stretch
- Readers must be compatible with multiple generations of media
- Long rewind, eject, load, spin-up times;  
not inherent, just no need in marketplace
- Designed for archival

# Replace large disks with smaller disks

---

- RAID
- Files are "striped" across multiple disks
- Redundancy yields high data availability
  - Availability: service still provided to user, even if some components failed
- Disks will still fail
- Contents reconstructed from data redundantly stored in the array
  - ⇒ Capacity penalty to store redundant info
  - ⇒ Bandwidth penalty to update redundant info

# RAID systems

---

- **A. RAID 0: Striped Disk Array**

Raid 0 is not a fault tolerant RAID solution, if one drive fails, all data within the entire array is lost. It is used where raw speed is the only (or major) objective. It provides the highest storage efficiency of all array types.

- **B. RAID 1: Mirrored Disk Array**

RAID 1 provides complete protection and is used in applications containing mission critical data. It uses paired disks, where one physical disk is partnered with a second physical disk. Each physical disk contains the same exact data to form a single virtual drive.



# RAID systems

---

- **C. RAID 10: Mirroring and Striping**

RAID 10 consists of multiple sets of mirrored drives. These mirrored drives are then striped together to create the final virtual drive. The result is an extremely scalable mirror array, capable of performing reads and writes significantly faster (since the disk operations are spread over more drive heads).

- **D. RAID (0+1): Striping and Mirroring**

Not to be confused with RAID 10 (they are very different). Raid 0+1 flips the order of RAID 10. Drives are first striped, then these drives are mirrored. Typically, two or more disks are striped to create one segment and an equal number of drives are striped to form an additional segment. These two striped segments are then mirrored to create the final virtual drive.

- **E. RAID 3: Striping with Dedicated Parity Disk**

RAID 3 is a fault tolerant version of RAID 1 (Striping). Fault tolerance is achieved by adding an extra disk to the array and dedicating it to storing parity information. Parity information is generated and written during write operations and checked on reads. It requires a minimum of three drives and provides data protection.

# RAID systems

---

- **F. RAID 5: Striping and Parity**

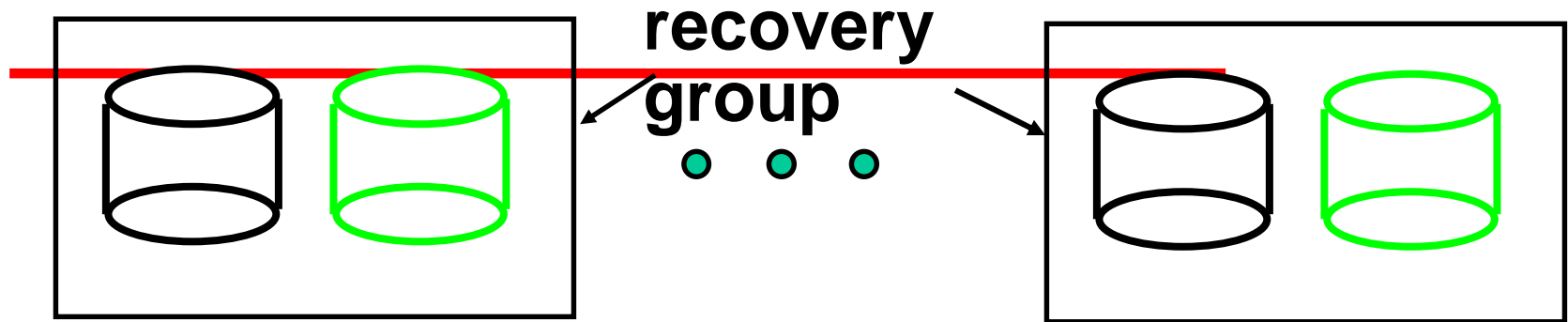
Raid 5 is similar to RAID 3 but the parity is not stored on one dedicated drive, instead parity information is interspersed across the drive array. RAID 5 requires a minimum of 3 drives. One drive can fail without affecting the availability of data. In the event of a failure, the controller regenerates the lost data of the failed drive from the other surviving drives.

By distributing parity across the arrays member disks, RAID Level 5 reduces (but does not eliminate) the write bottleneck. The result is asymmetrical performance, with reads substantially outperforming writes. To reduce or eliminate this intrinsic asymmetry, RAID level 5 is often augmented with techniques such as caching and parallel multiprocessors.

Pro's:

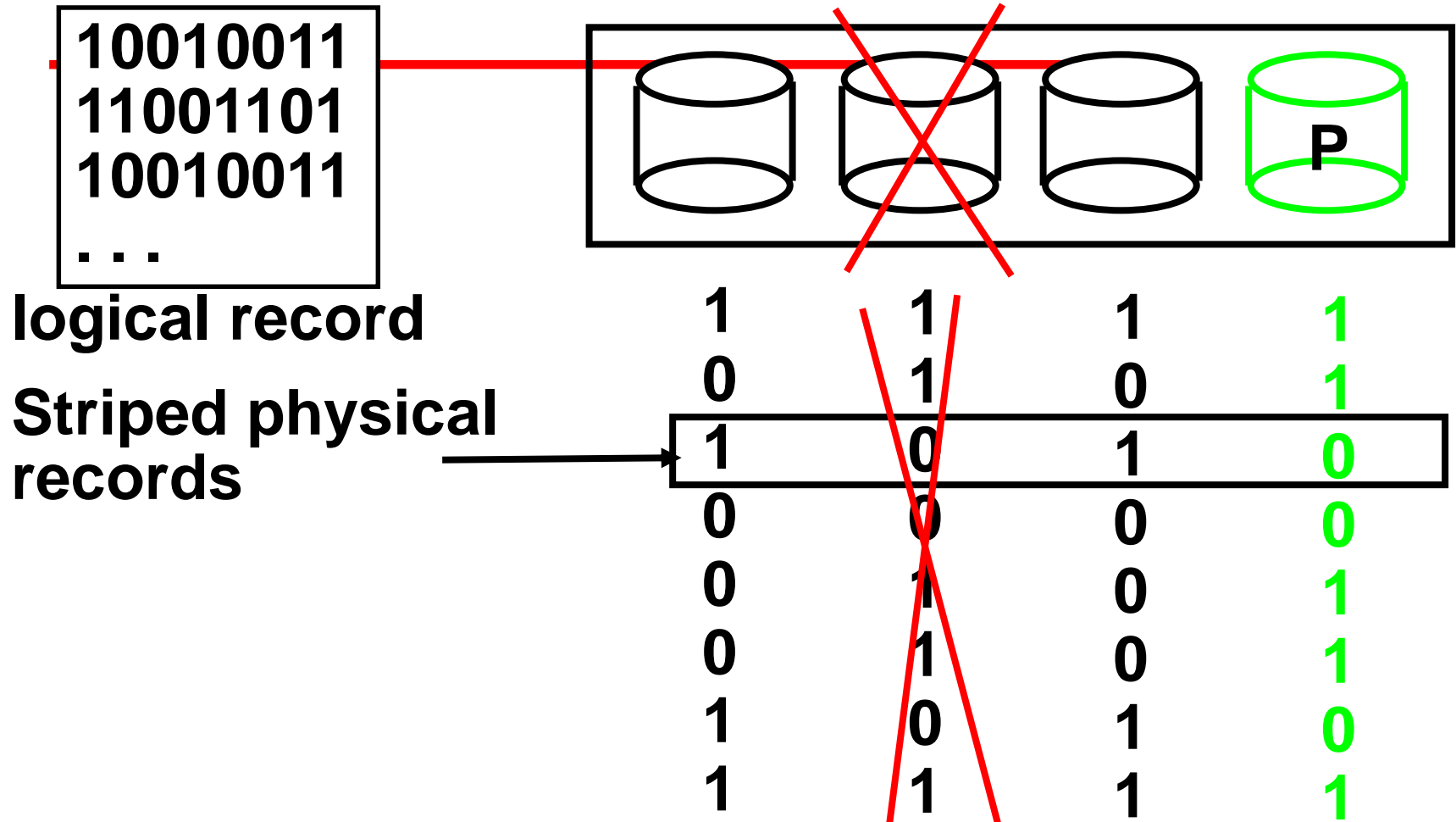
- Best suited for heavy read applications.
  - The amount of useable space is the number of physical drives in the virtual drive minus 1.
- Con's
- A single disk failure reduces the array to RAID 0

# RAID1



- Each disk is fully duplicated onto its “mirror”  
Very high availability can be achieved
- Bandwidth sacrifice on write:  
Logical write = two physical writes
  - Reads may be optimized
- Most expensive solution: 100% capacity overhead
- (RAID 2 not interesting, so skip)

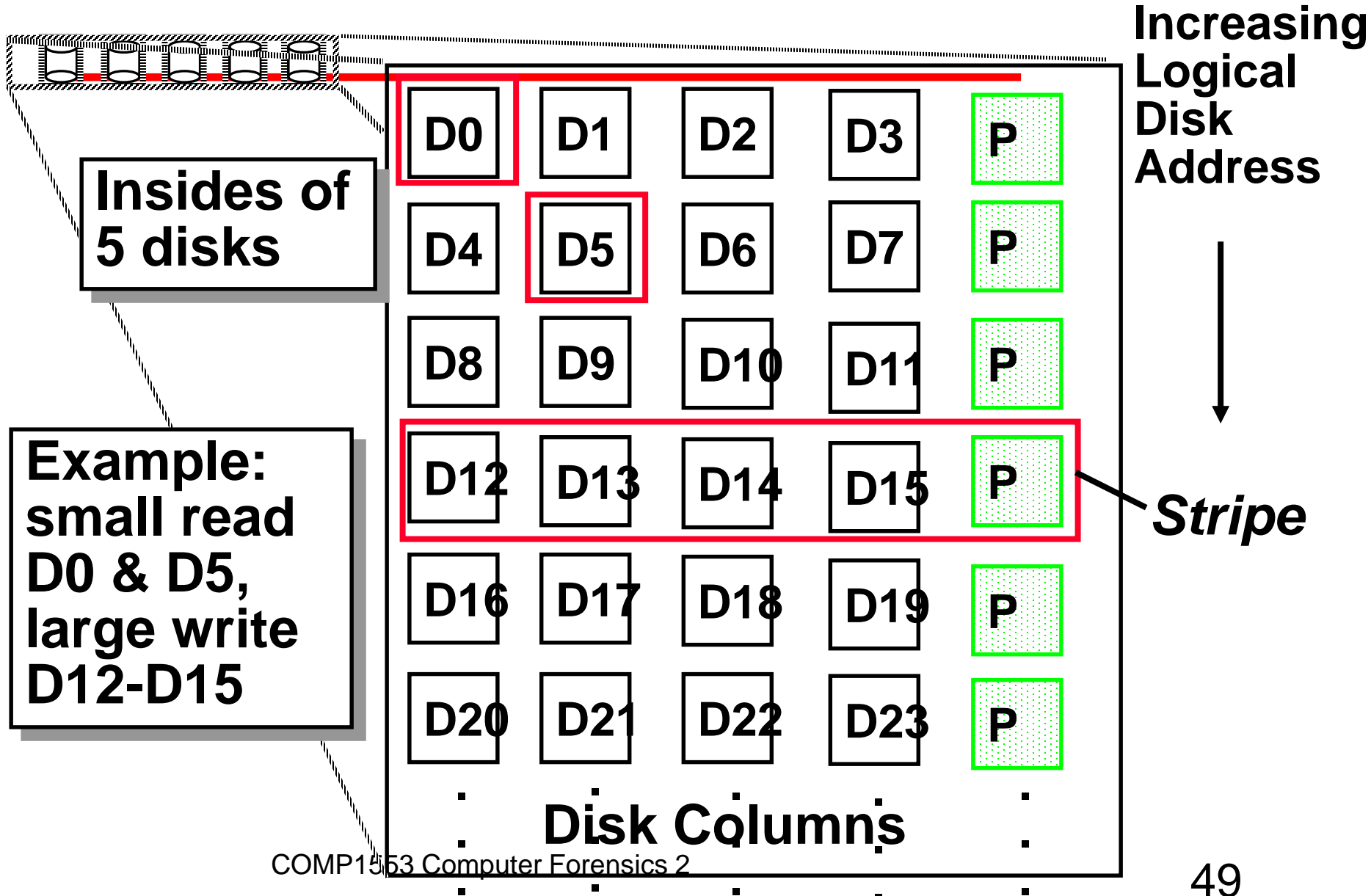
# RAID 3 – Parity disk



# RAID 4

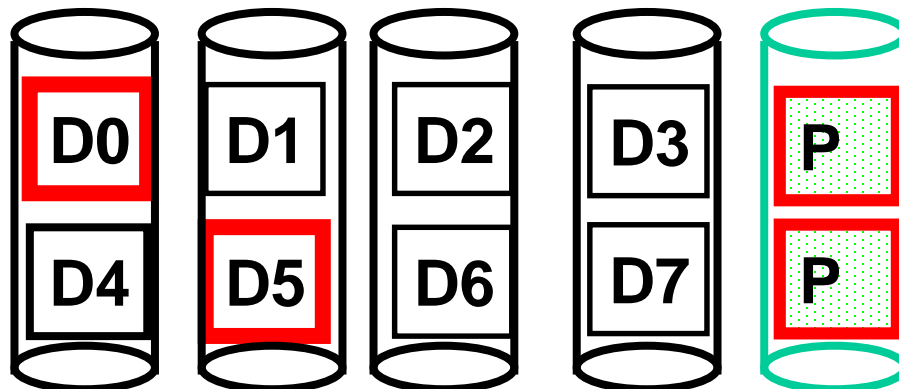
- 
- RAID 3 relies on parity disk to discover errors on Read
  - But every sector has an error detection field
  - Rely on error detection field to catch errors on read, not on the parity disk
  - Allows independent reads to different disks simultaneously

# RAID 4



# RAID 5

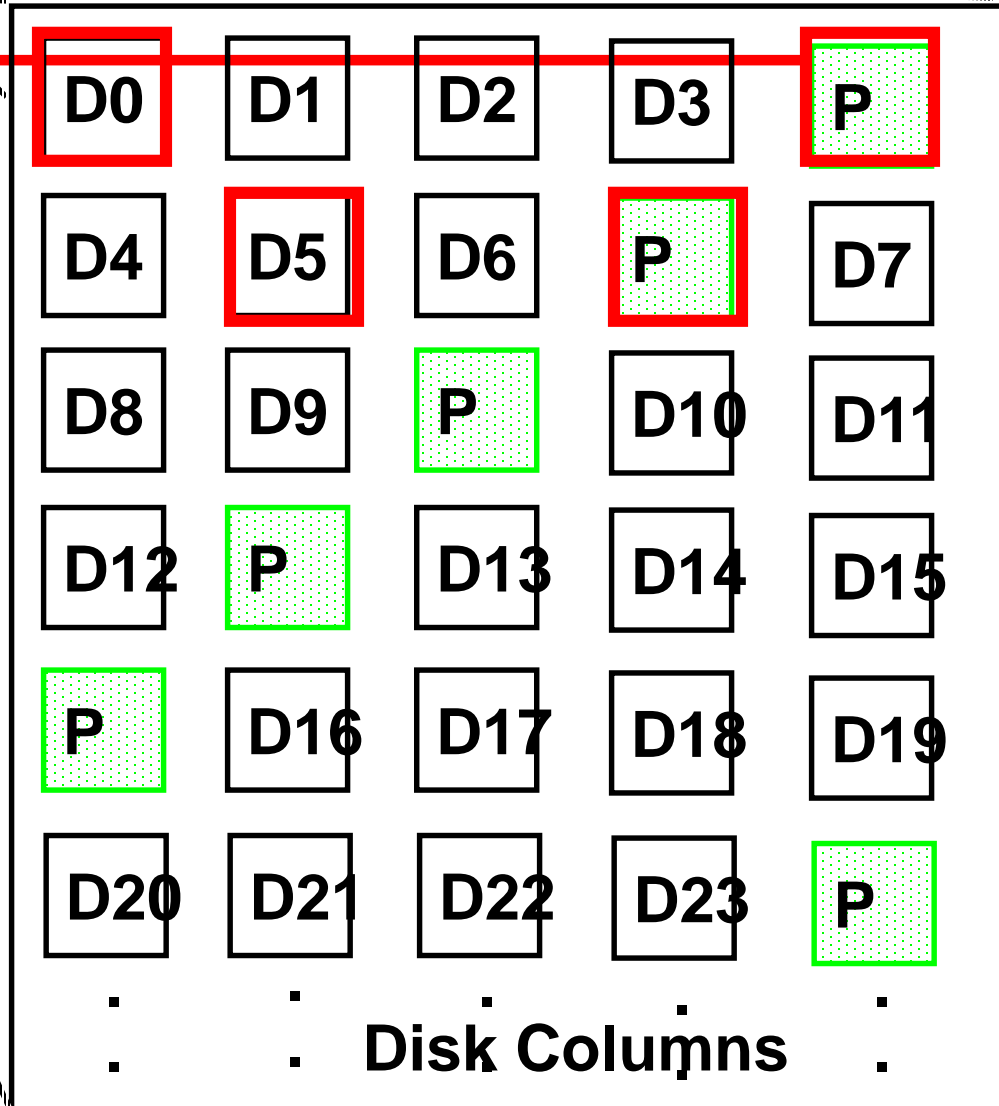
- RAID 4 works well for small reads
- Small writes (write to one disk):
  - Option 1: read other data disks, create new sum and write to Parity Disk
  - Option 2: since P has old sum, compare old data to new data, add the difference to P
- Small writes are limited by Parity Disk: Write to D0, D5 both also write to P disk



# RAID 5

Independent writes possible because of interleaved parity

Example:  
write to  
D0, D5  
uses disks  
0, 1, 3, 4





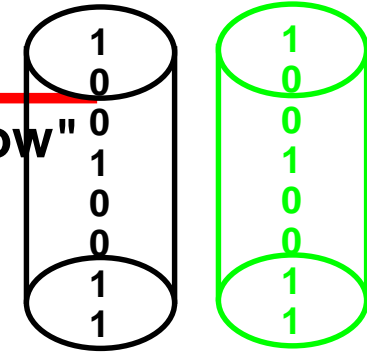
# RAID Summary

- ***Disk Mirroring, Shadowing (RAID 1)***

Each disk is fully duplicated onto its "shadow"

Logical write = two physical writes

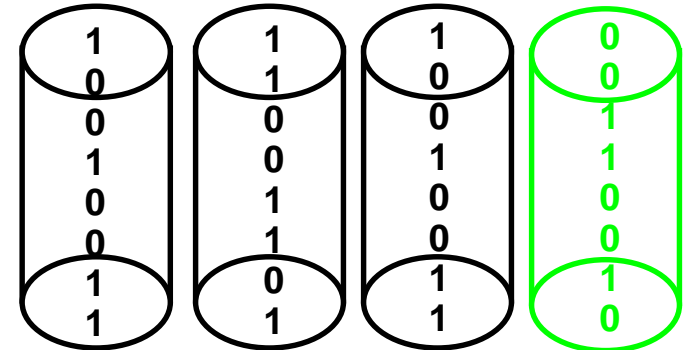
100% capacity overhead



- ***Parity Data Bandwidth Array (RAID 3)***

Parity computed horizontally

Logically a single high data bw disk



- ***High I/O Rate Parity Array (RAID 5)***

Interleaved parity blocks

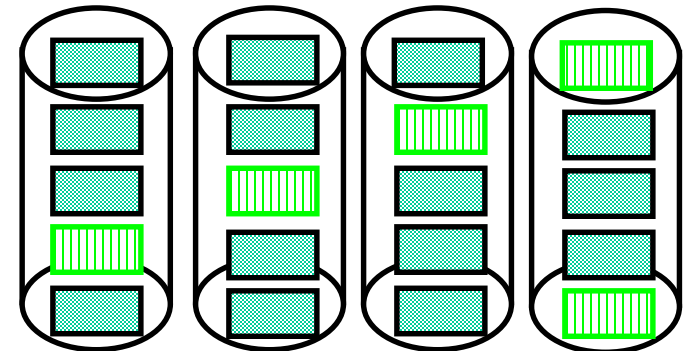
Independent reads and writes

Logical write = 2 reads + 2 writes

Data 00000000

Peven 00000000

Podd 10000000



# What is an Operating System?

---

**“is an interface between hardware and user”**

**“is the program which is responsible for coordinating the activities of the various processes that a PC is executing”**

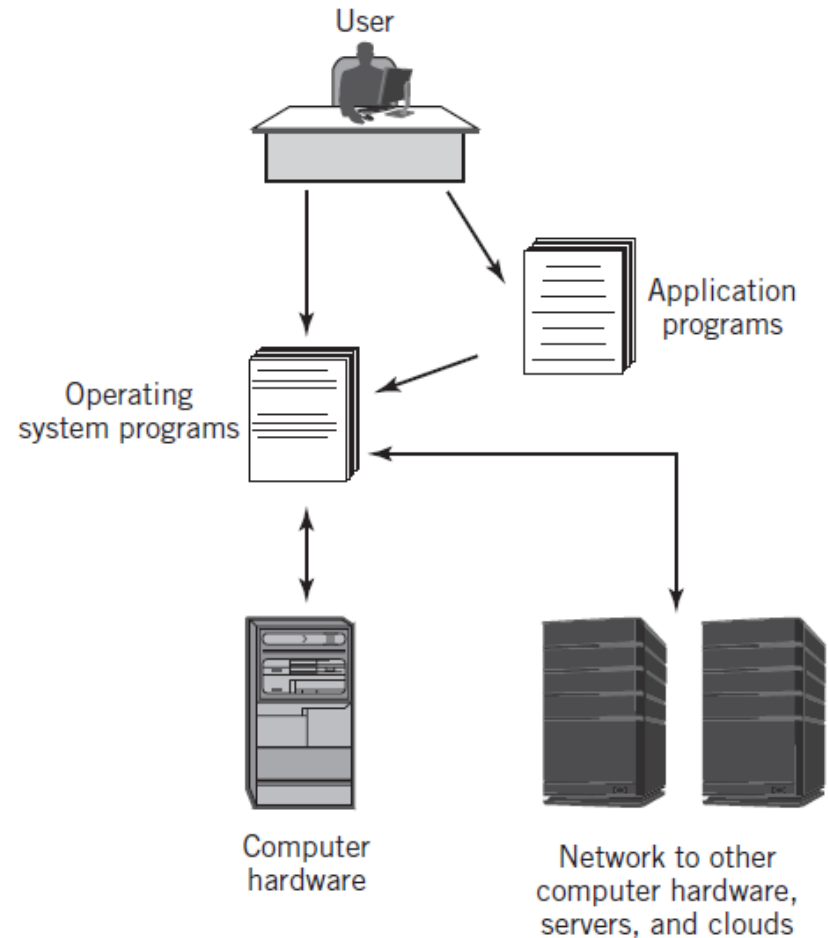
**“is a software that controls the allocation and usage of hardware resources such as memory, CPU time, disk space, and input and output devices”**



**“software that controls the execution of computer programs and may provide various services”**

# Integrated Computer Environment

- The relationship between the various components of a computer system



# Operating System

---

- Operating system (e.g., Windows, Linux, MacOS): interfaces between a user's program and the hardware and provides a variety of services and supervisory functions such as:
  - handling input and output operations
  - allocating storage and memory
  - providing for sharing the computer among multiple applications using it simultaneously
- A program that controls the execution of
- application programs
- An interface between applications and hardware
- Makes the computer more convenient to use
- Manages the resources of a computer and
- controls the way they are used
- Allows resources to be used in an efficient manner
- Examples of OS's?

# Where do we use them?

---

- In more and more places!
- Desktop and Server Computers
- DOS + Windows 95/98/ME
- Windows NT/2000/XP
- Free Unix variants: Linux, FreeBSD, NetBSD, etc.
- Commercial Unix variants: Solaris, HP-UX, AIX, etc.
- MacOS
- Some Game Consoles
- Xbox: Cut-down Windows 2000

# Where do we use them (contd.)

---

- Personal Digital Assistants (PDAs)
- PalmOS
- Windows CE Windows Mobile
- Embedded Linux
- Mobile Phones
- Symbian OS
- Windows Mobile
- Cars (fancy ones)
- Digital Cameras (fancy ones)
- MP3 Players (iPods, etc.)
- Refrigerators!

# Definition of an Operating System

---

- “A collection of computer programs that integrate the hardware resources of the computer and make those resources available to a user and the user’s programs, in a way that allows the user access to the computer in a productive, timely, and efficient manner.”
- The operating system has two fundamental purposes:
  - To control and operate the hardware in an efficient manner
  - To allow the “users” powerful access to the facilities of the machine by providing a variety of facilities and services
- In addition, the operating system expands the capability of the computer system to allow for the concurrent processing of multiple programs and support for multiple users (local and networked)

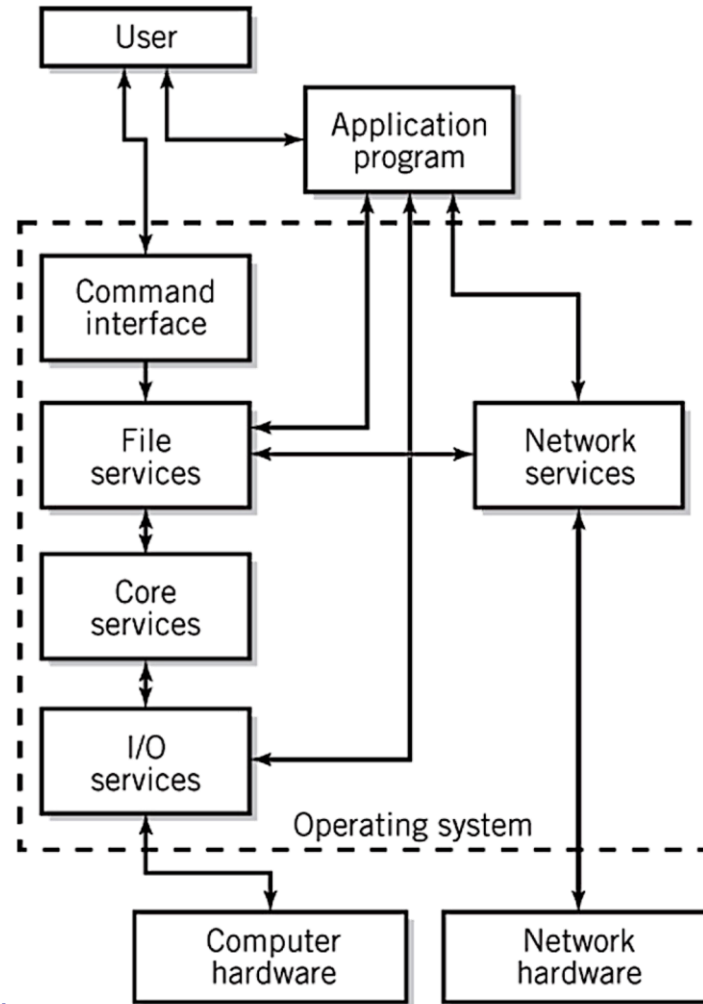
# Operating System – Basic Services

---

- Accepts and processes commands and requests from users and users' programs and presents appropriate output results
  - Manages, loads, and executes programs
  - Manages hardware resources of the computer including interfaces to networks and other external parts of the system
- 
- Note: The operating system itself consists of hundreds or thousands of programs, each specialised for particular OS tasks
  - Upon program completion control returns to the OS, enabling users to operate without restarting



# Simplified Diagram of Operating System Services



# Summary

---

- More on OSs next week!
- Tasks for you: review what we covered today and do some research on the topics!

Any questions?