

# Organización de Computadoras



---

## Clase 10



# Temas de Clase

---

- Memoria Cache
- Memoria Externa



# Memoria Caché

---

- Históricamente CPU han sido más rápidas que las memorias.
- El aumento de circuitos que es posible incluir en un chip
  - ❖ Diseñadores de CPU lo usaron para hacerla más veloz (ej. pipeline).
  - ❖ Los diseñadores de memoria lo usaron para aumentar la capacidad del chip (más memoria, más grandes decodificadores).



## Memoria Caché (2)

---

- Esta diferencia implica: después que la CPU 'emite' una solicitud de lectura a la memoria (bus de direcciones, bus de control) pasan muchos ciclos de reloj antes que reciba la palabra que necesita, por el bus de datos.



# Memoria Caché (3)

---

- En todos los ciclos de instrucción, la CPU accede a memoria al menos una vez, para buscar la instrucción y muchas veces accede a buscar operandos.
- La velocidad a la cual la CPU ejecuta instrucciones está limitada por el tiempo del ciclo de memoria.



# Memoria Caché (4)

---

- El problema no es tecnológico sino económico. Se pueden construir memorias tan rápidas como la CPU, pero para obtener la máxima velocidad tiene que estar dentro del chip de la CPU
- Llegar a la memoria por el bus del sistema es 'lento'.



# Memoria Caché (5)

---

## Solución

- Técnicas para combinar una cantidad pequeña de memoria rápida con una cantidad grande de memoria lenta, para obtener la velocidad de memoria 'casi' rápida.



# Principios (1)

---

- El uso de la memoria caché se sustenta en dos principios ó propiedades que exhiben los programas:

## 1. Principio de localidad espacial de referencia

- cuando se accede a una palabra de memoria, es 'muy probable' que el próximo acceso sea en la vecindad de la palabra anterior.





# Principios (2)

---

## 2. Principio de localidad temporal de referencia

- cuando se accede a una posición de memoria, es 'muy probable' que un lapso de 'tiempo corto', dicha posición de memoria sea accedida nuevamente.



# Localidad espacial

---

- Localidad espacial, se sustenta en:
  - Ejecución secuencial del código
  - Tendencia de los programadores a hacer próximas entre sí variables relacionadas
  - Acceso a estructuras tipo matriz ó pila



# Localidad temporal

---

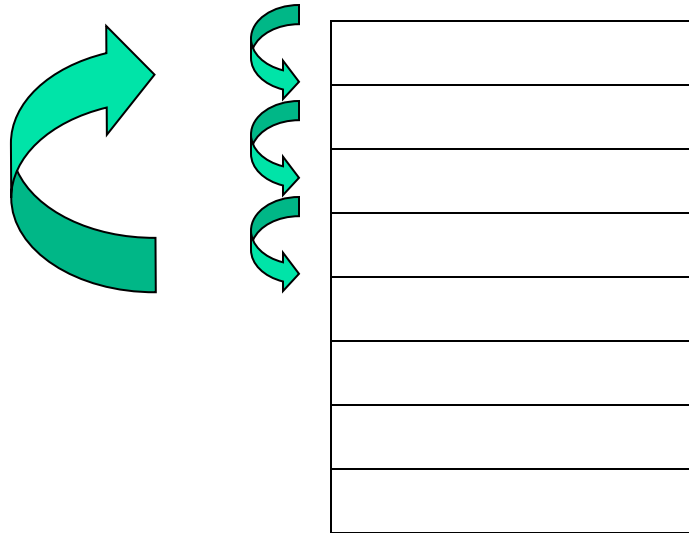
- Localidad temporal, se sustenta en:
  - Formación de ciclos o bucles
  - Subrutinas (Procedimientos o Funciones)
  - Pilas



# Ejemplo

---

- Ej.





## Ejemplo (2)

---

- Estas 2 sentencias exhiben los dos principios antes mencionados:

for  $i=1$  to  $i=10$ , do

$A[i] := 0;$

- ❖ En cada ciclo se consulta cuanto vale  $i$ .
- ❖ Cada asignación  $A[i] := 0$  almacena un 0 en un elemento del arreglo (el siguiente).



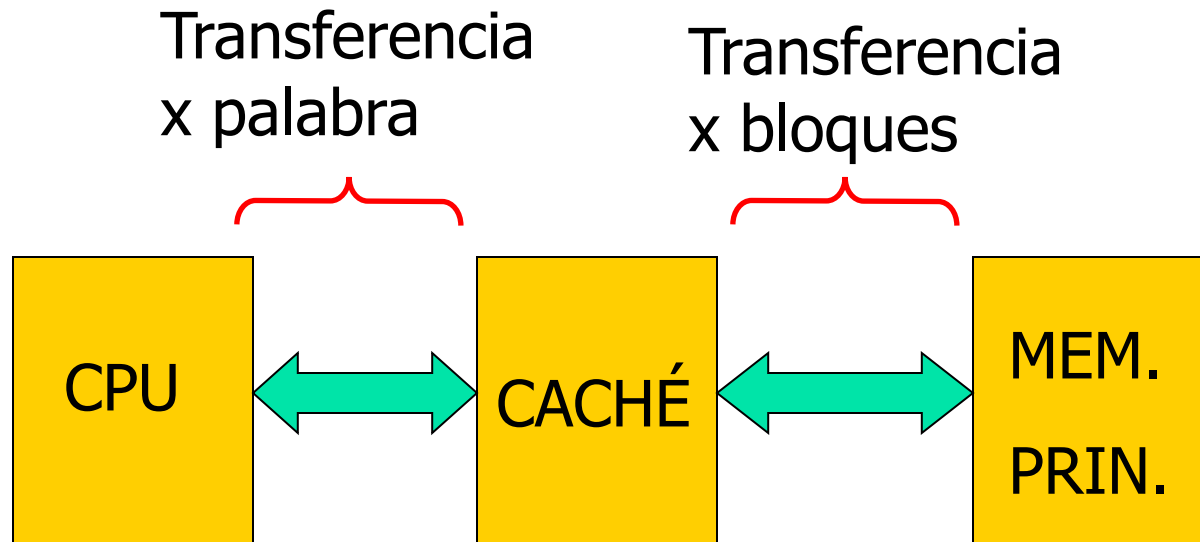
# Caché

---

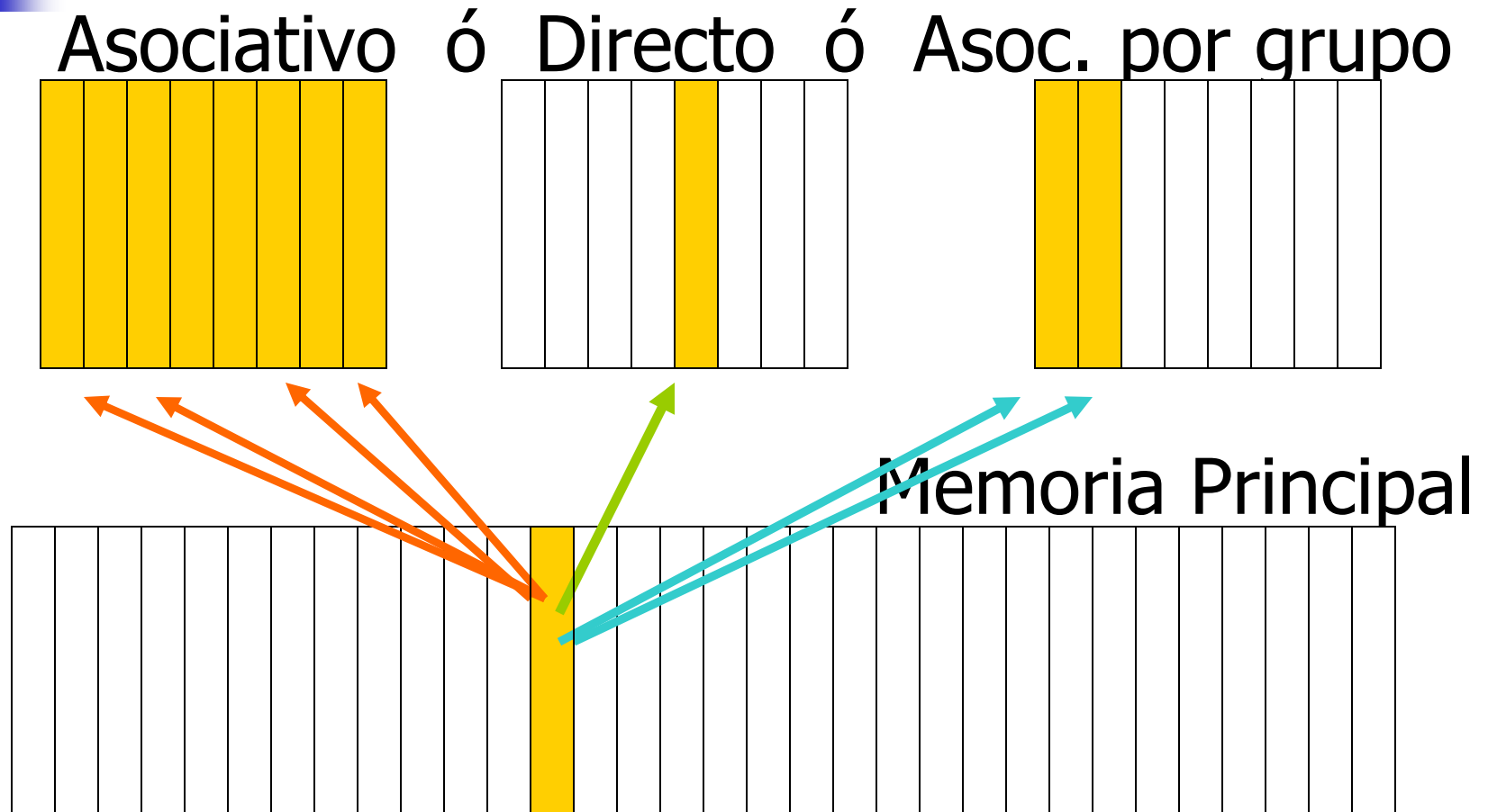
- La idea general es que cuando se hace referencia a una palabra, ella y alguna de las vecinas se traen de la memoria grande y lenta a la caché, para que en el siguiente acceso la palabra buscada se encuentre en el caché.

# Caché

- Ej.



# Mapeo de la memoria







# Aciertos y fallos (1)

---

- La efectividad de la caché se expresa a través de la frecuencia de aciertos: es decir el número de veces que la caché acierta direcciones.
- Un **acierto de caché** sucede cuando los datos que necesita el procesador están almacenados en la caché
  - la CPU obtiene los datos a alta velocidad.



## Aciertos y fallos (2)

---

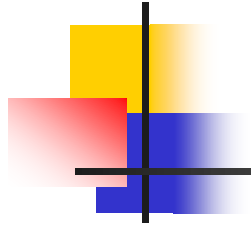
- Un **fallo de caché** ocurre cuando los datos buscados no se encuentran en la caché
  - la CPU tiene que obtenerlos de la memoria principal, a una velocidad menor.



# Caché: L1 y L2

---

- ✓ ¿Por qué hay dos (ó +) niveles de caché?
  - ✓ L1 y L2
- ✓ Porcentaje de aciertos: 90%
- ✓ Porcentaje de fallos: 10%
- ✓ Conviene mejorar el 90% con el mismo razonamiento y no poner esfuerzo en el 10% restante.



# Memoria externa



# Tipos de memoria externa

---

- Discos magnéticos
- Discos ópticos
  - CD-ROM
  - CD-R
  - CD-RW
  - DVD
- Cintas Magnéticas



# Discos magnéticos

---

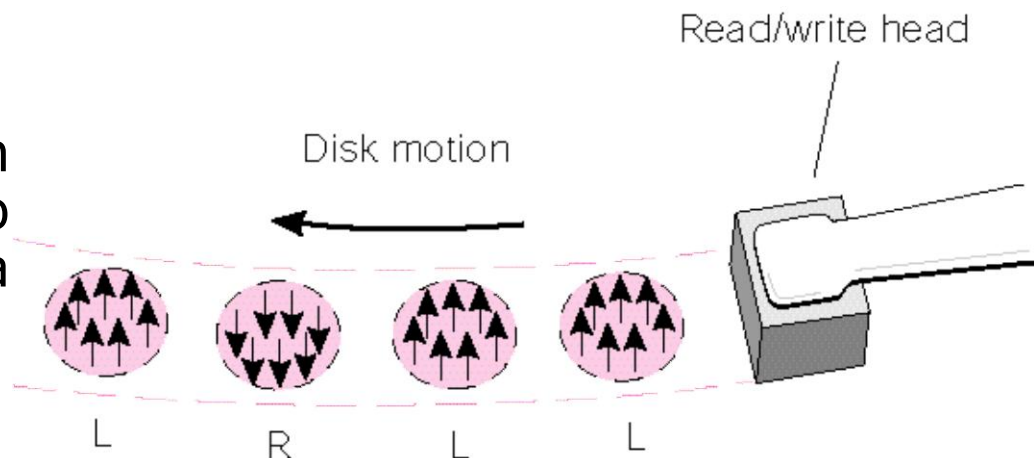
## Platos

- Superficies de Al cubiertos con óxido de Fe, material magnético.
- ✓ Ahora también se usa vidrio
  - Se dilata menos que el Al.
  - Superficie más uniforme.
  - Reducción de defectos superficiales.

# Principios físicos

- ✓ Pequeñas áreas del disco son magnetizadas en diferentes direcciones por un transductor.
- ✓ Debe existir un movimiento relativo entre disco y el transductor al momento de la lectura/escritura.

⌘ Cambios en la dirección de magnetización es lo que se detecta en la lectura





# Mecánica de lectura y escritura

---

- ❖ Lectura y escritura es a través de una cabeza transductora (bobina).
- ❖ Durante lectura/escritura, la cabeza es estacionaria y el plato gira.
- ❖ Se almacenan ceros y unos por medio de la magnetización de pequeñas áreas del material.



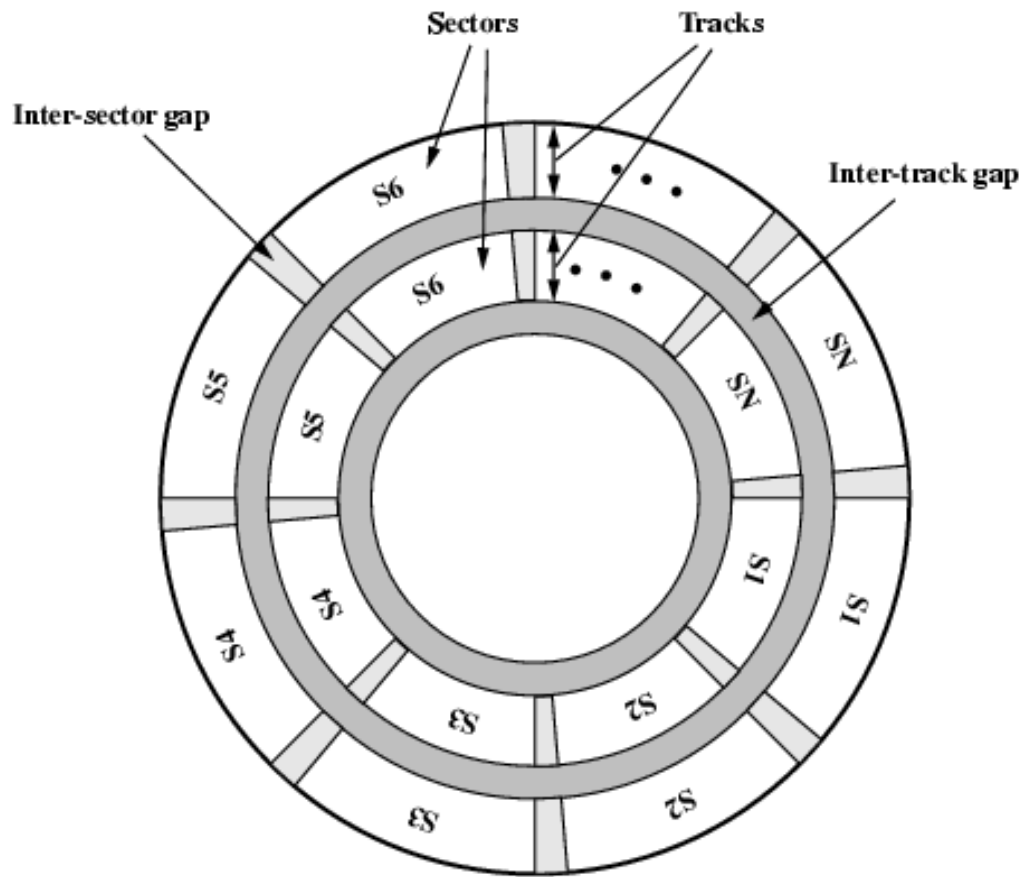


# Organización de los datos

---

- ✓ Anillos concéntricos: **pistas** ó tracks.
  - Espacios (gaps) entre pistas
  - Reducir gaps para aumentar capacidad
  - Mismo N° de bits por pista
  - Velocidad angular constante
- ✓ Pistas divididas en sectores
- ✓ Mínimo tamaño de bloque: **sector**
- ✓ Bloque: más de un sector (**cluster**)

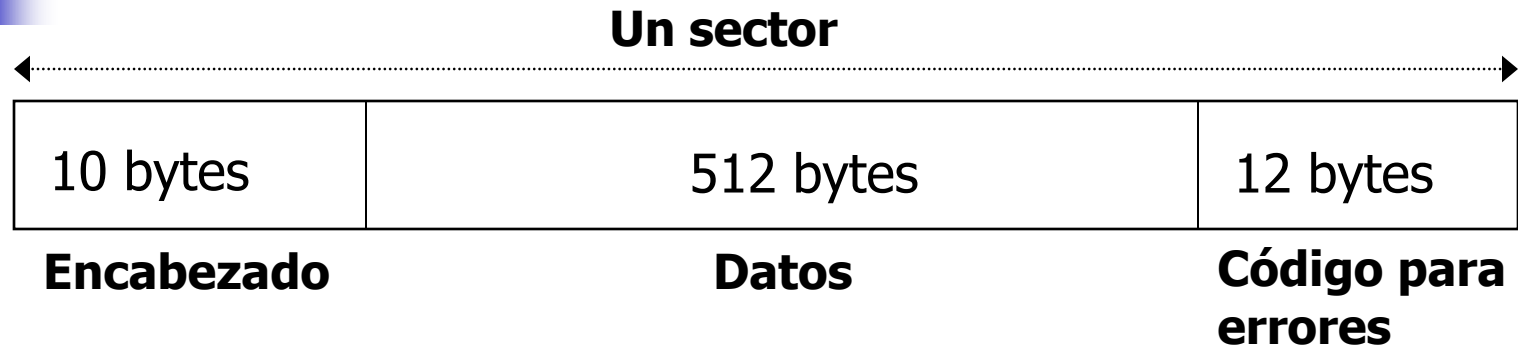
# Pista y Sector



- Un número entero de sectores se graban en una pista.
- El sector es la unidad de transferencia de/hacia el disco.



# Sector típico



- ✓ Sucesión o serie de bits divididos en campos
  - **Encabezado** con información para sincronizar la lectura e identificar el sector.
  - **Datos** con longitud en bytes expresada usualmente como potencia de 2.
  - **Código para errores** con información para detectar y/o corregir posibles errores.



# Características posibles

---

- ✓ Cabeza fija (raro) o móvil.
- ✓ Disco removible o fijo.
- ✓ Simple ó doble lado.
- ✓ Uno ó múltiples platos.
- ✓ Mecanismo de cabeza:
  - Contacto (Floppy)
  - Distancia Fija
  - Aerodinámica (Winchester)

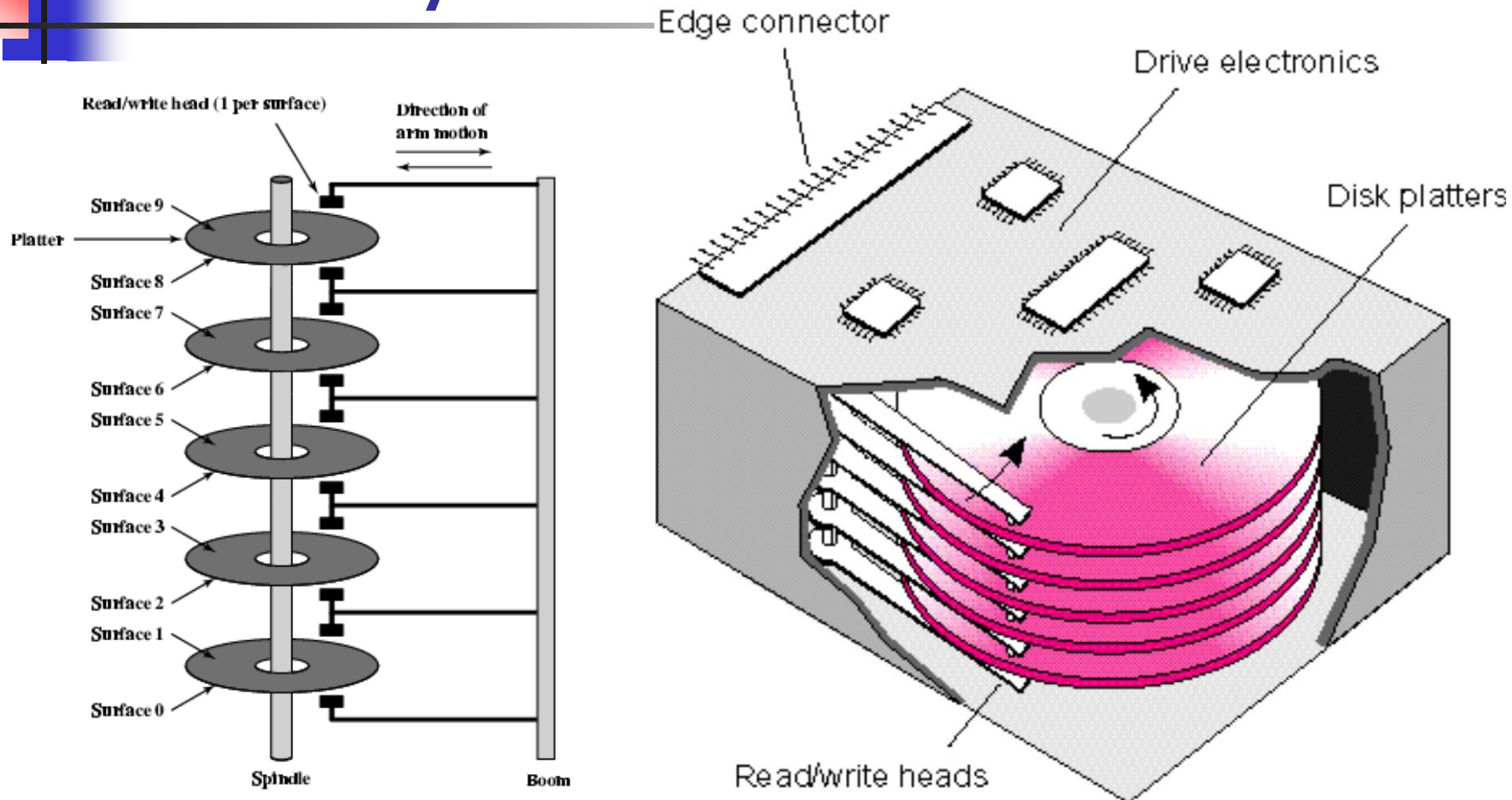


# Estructura de un disco

---

- Múltiples platos
- Una cabeza por cara
- Todas las cabezas se mueven solidariamente
- Pistas alineadas en cada plato forman cilindros
- Datos son almacenados por cilindros
  - Reduce movimientos de cabezas
  - Aumenta velocidad de respuesta

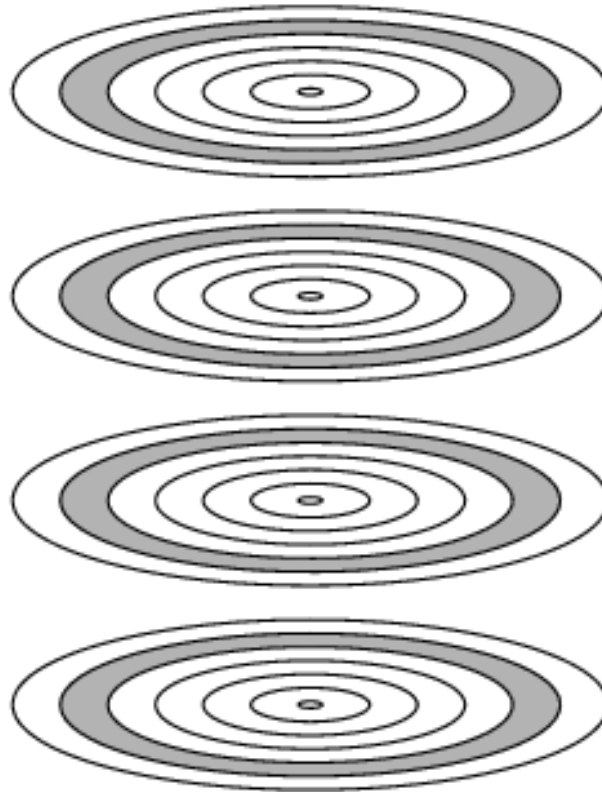
# Platos y cabezas





# Cilindros

---





# Velocidad de giro del disco

---

- ✓ Rota a velocidad angular constante - CAV (rpm)
  - ✓ Un bit más cercano al centro gira más lento que un bit en la periferia.
  - ✓ Los bits (sectores) ocupan distinto espacio en diferentes pistas.
- ✓ En lectura o escritura se requiere sólo mover la cabeza hasta una pista y esperar un sector.
- ✓ Determina la velocidad de transferencia de datos (bits/seg).





# Tiempos

---

- Tiempo de seek (búsqueda)
    - Mover al cilindro (o pista) correcto
  - Tiempo de latencia (por rotación)
    - Esperar que el sector “pase” por debajo de la cabeza
  - Tiempo de Acceso:  $T.\text{seek} + T.\text{latencia}$
- Tiempo Total:
- $T.\text{ de Acceso} + T.\text{ de Transferencia de datos}$



# Capacidad del disco: cálculo

---

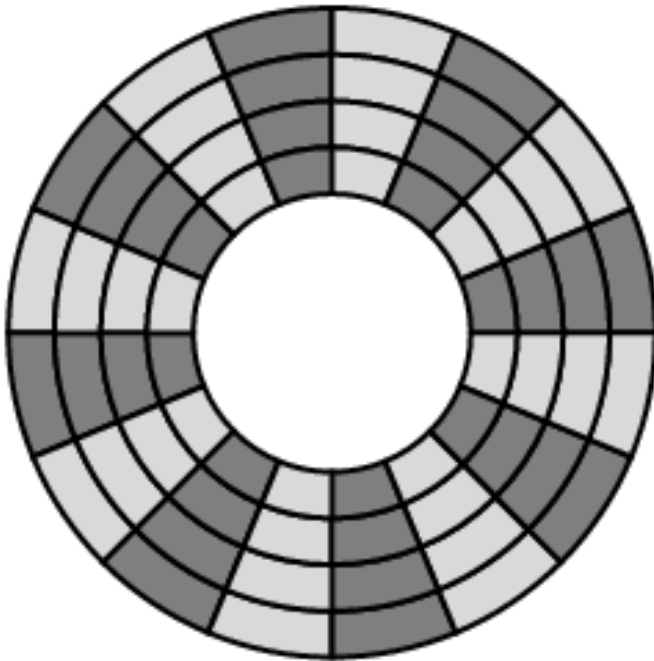
$$\text{Capacidad} = \frac{\text{bytes}}{\text{sector}} \times \frac{\text{sectores}}{\text{pista}} \times \frac{\text{pistas}}{\text{superficie}} \times \# \text{ de superficies}$$

- ✓ Se desperdicia espacio en pistas externas.
- ✓ Hoy en día se usan zonas para incrementar la capacidad
  - c/zona tiene fija la cantidad de bits/pista.
  - requieren circuitos más complejos.

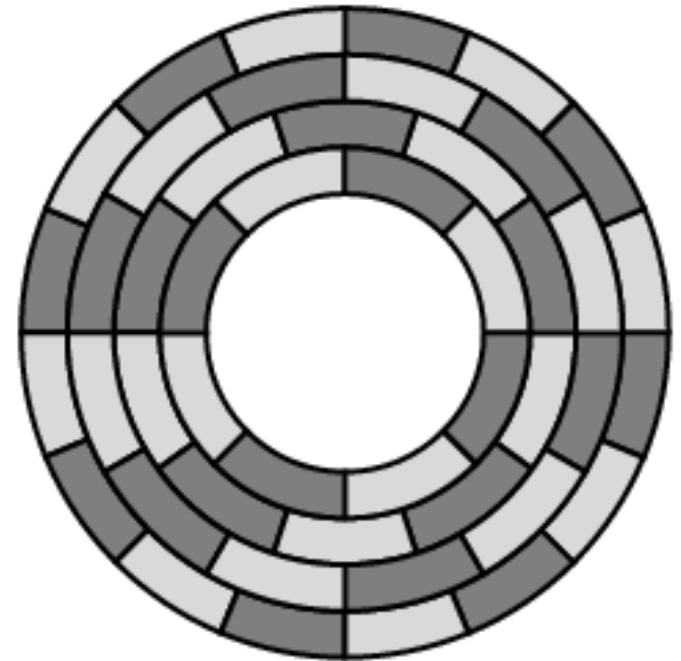


# Aumento de capacidad

---



**Grabación en CAV**



**Grabación en zonas**



# Formato

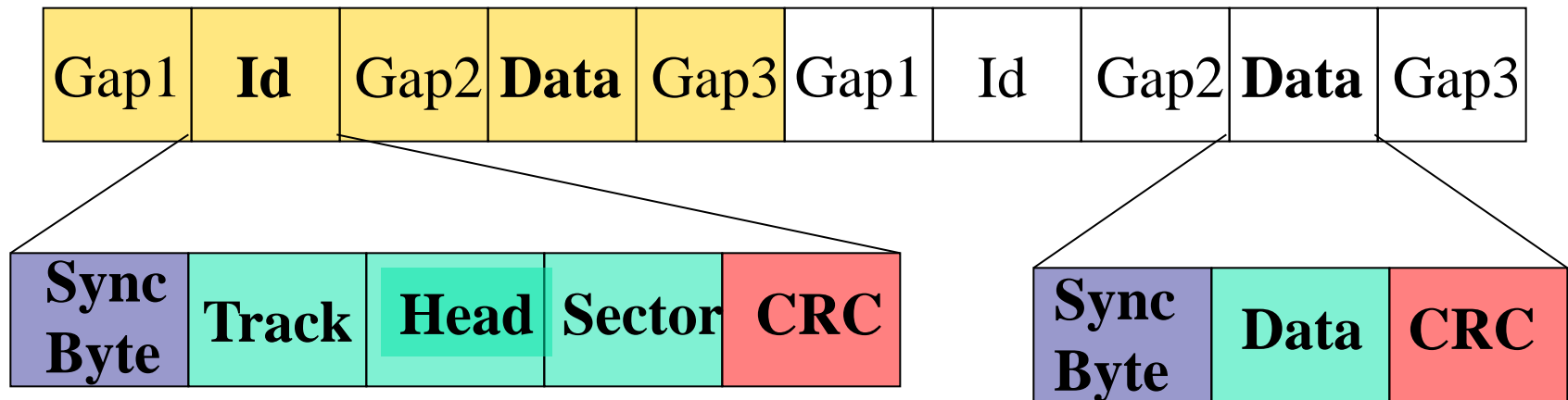
---

- ✓ Define cantidad, tamaño y función de distintos campos en cada pista
  - Hardware: tamaño de sector fijo por marcas físicas.
  - Software: tamaño de sector determinado por S.O.



# Formato ST506 (antiguo!)

---





mas información ...

---

- Capítulo 5: Memoria Externa
  - Stallings, W., 5º Ed.
- Links de interés
  - <http://www.pctechguide.com/02Storage.htm>