



# Memorias



Material Recopilado y Elaborado por el Profesor  
Ricardo González

1



## Area de Almacenamiento Primario

La memoria da al procesador almacenamiento temporal para programas y datos. Todos los programas y datos deben transferirse a la memoria desde un dispositivo de entrada o desde el almacenamiento secundario (disquete), antes de que los programas puedan ejecutarse o procesarse los datos. Las computadoras usan 2 tipos de memoria primaria:

ROM (read only memory), memoria de sólo lectura, en la cual se almacena ciertos programas e información que necesita la computadora las cuales están grabadas permanentemente y no pueden ser modificadas por el programador. Las instrucciones básicas para arrancar una computadora están grabadas aquí y en algunas notebooks han grabado hojas de calculo, basic, etc.

RAM (Random access memory), memoria de acceso aleatorio, la utiliza el usuario mediante sus programas, y es volátil.

2



## Area de Almacenamiento Primario

La memoria del equipo permite almacenar datos de entrada, instrucciones de los programas que se están ejecutando en ese momento, los datos resultados del procesamiento y los datos que se preparan para la salida. Los datos proporcionados a la computadora permanecen en el almacenamiento primario hasta que se utilizan en el procesamiento.

Durante el procesamiento, el almacenamiento primario almacena los datos intermedios y finales de todas las operaciones aritméticas y lógicas. El almacenamiento primario debe guardar también las instrucciones de los programas usados en el procesamiento.

La memoria está subdividida en celdas individuales cada una de las cuales tiene una capacidad similar para almacenar datos.

3



## Area de Almacenamiento Secundario

El almacenamiento secundario es un medio de almacenamiento definitivo (no volátil como el de la memoria RAM). El proceso de transferencia de datos a un equipo de cómputo se le llama procedimiento de lectura. El proceso de transferencia de datos desde la computadora hacia el almacenamiento se denomina procedimiento de escritura. En la actualidad se pueden usar principalmente dos tecnologías para almacenar información:

- 1.- El almacenamiento Magnético.
- 2.- El almacenamiento Óptico.

Algunos dispositivos combinan ambas tecnologías.

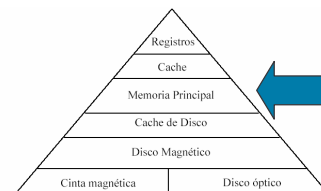
4

# Tipos de Memoria (RAM)

RAM significa Memoria de Acceso Aleatorio (Random Access Memory) y es la que se encarga de almacenar la información mientras el computador se encuentra encendido. Esto quiere decir que cuando el computador arranca ésta se encuentra vacía inicialmente, y entonces se lee información del disco duro y se almacena en ella el sistema operativo (primero), después, cualquier otra cosa que hagamos. Al trabajar en un procesador de palabras, por ejemplo, la información se almacena aquí. La información sólo pasa al disco duro cuando grabamos. Por esto se pierde la información si se apaga el computador sin antes haberla grabado.

5

## Tipos de Memoria (RAM)



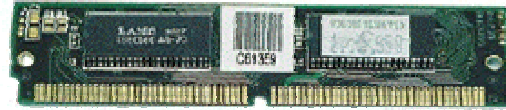
La **DRAM** (*Dynamic Random-Access Memory*), Este tipo de memoria retiene los datos que almacena, al escribirlos continuamente cada pocos milisegundos. La DRAM es el tipo más común de memoria expandible en los PC y las estaciones de trabajo. Viene en módulos “estandar” para ser incluidos en los computadores

La **SRAM** (*Static Random-Access Memory*) Este tipo de memoria es más rápida, confiable y costosa que la DRAM por lo que es más usada como memoria Cache. Las primeras formas de SRAM eran clasificadas como Asyn o Sync RAM si estas estaban sincronizadas o no con el reloj del sistema. La forma más común de SRAM hoy en día es la pipeline burst SRAM, la cual puede operar a mayores velocidades de bus. Puede venir como chips de memoria opcionales y proveen 128Kb, 256Kb o 512Kb de memoria cache, estos chips poseen sus propios zócalos o ranuras

La RAM esta instalada en sets, o bancos. Se agregan mas sets para aumentar la capacidad de la computadora. Los chips de RAM tienen diferentes densidades. Los motherboards de los clones de IBM PC pueden aceptar una variedad de chips de RAM, que incluyen 16Mb, 32Mb, 64Mb, 128Mb, 256Mb, 512Mb y 1Gb.

6

## Tipos de Memoria (DRAM) Según su empaque



**SIMM** (single in-line memory module) Esta forma de memoria se puede ver en sistemas no muy modernos. El primer modulo SIMM tenia 30 contactos o pines y permitia transferencias de datos a una tasa de 8 bits a la vez. Con el desarrollo de los computadores de 32 bits se desarrollaron empaques mas anchos de 72 pines para permitir transferencias de datos de 32 bits a la vez

**DIMM** (dual in-line memory module) Los DIMM se parecen mucho a los SIMM, excepto que los DIMM son mas "largos" (168 pines), y los pines del otro lado del modulo están electricamente aislados entre si. los DIMM de 168 pines permiten tasas de transferencia de 64 bits por vez y son muy comunes en sistemas Pentium y Athlon. Los DIMM más recientes llegan a tener 184 pines. Los DDR2 podrán tener entre 200 y 240 pines.

**RIMM** Es un nombre comercial para un tipo especial de DIMM (a trademarked name for the 184-pin Direct Rambus Memory Module) Estos modulos son parecidos a los DIMM pero generan mas calor, debido a proveen mayores velocidades de acceso y transferencia de datos. los RIMM vienen con un disipador de calor de aluminio que cubre el modulo. A pesar de tener el mismo número de pines que los DIMM sólo pueden ser empleados en tarjetas madres o sistemas específicamente diseñados para RIMMs.

**SODIMM** (small outline) y **MicroDIMM**. Son formatos de memoria para Laptops. Los sub Notebooks emplean MicroDIMM.

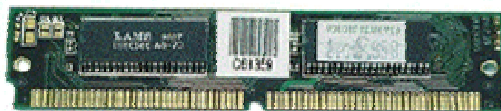
Existen versiones de Rambus llamadas SORIMM.

7

## Tipos de Memoria (DRAM) Según su empaque



SIMM



DIMM



SODIMM



8



## Tipos de Memoria (DRAM)

As we've discovered, most computer memory today uses a DIMM memory module using DRAM. The problem at this stage is that the type of DRAM on the module can greatly affect performance and cause the most confusion when designing or configuring a system.

- FPM (fast page mode DRAM). Introduced in 1987, this is an early form of DRAM that was once very common because it was slightly faster than DRAM. This type of memory was frequently mounted on a SIMM in 486 and early Pentium computers.
- EDO (extended data output). EDO offers a slight performance boost over FPM DRAM by cutting a few steps when addressing memory. Introduced in 1995, EDO requires a specific chip set and is limited to bus speeds of 66 MHz. Once again, these chips are mounted on SIMM modules.
- BEDO (burst EDO RAM). This is a slightly faster variant of the EDO RAM chip in which read and write operations send bursts of data in batches of four. This type of memory is not widely used.
- SDRAM (synchronous DRAM). SDRAM synchronizes itself with the microprocessor clock speed allowing faster access to memory. These chips are mounted on DIMM memory modules and are classified according to the CPU speed they are designed to support. For example, a PC133 SDRAM DIMM is designed for a Pentium 133 CPU.
- ESDRAM (enhanced SDRAM). This version of SDRAM includes a small SRAM cache in order to reduce latency and speed up operations. This standard is not widely used.
- SLDRAM (synchronous link dynamic RAM). Another version of SDRAM, which was designed as a royalty-free, open-industry standard design alternative to RDRAM. It is not widely used.
- DDR (double data rate SDRAM). DDR allows data transfers on both the rising and falling edges of the clock cycle, which doubles the data throughput. DDR SDRAM chips are mounted on 184-pin DIMM modules and are typically available in 128 Mb to 1 GB capacity. They operate at bus speeds up to 400 MHz. DDR memory is very common, but the technology behind it is at its limits, and it is being replaced by DDR2.
- DDR2. These chips are the next generation of DDR SDRAM memory. They are mounted on 240-pin DIMM modules, can operate at higher bus speeds and have a capacity to hold 256 Mb to 2 GB of memory. DDR2 has twice the latency of DDR but delivers data at twice the speed of DDR, theoretically performing at the same level.
- RDRAM (Rambus DRAM). A proprietary protocol-based, high-speed memory technology developed by Rambus Inc., RDRAM has current frequencies of 800 MHz to 1200 MHz, and planned chip sets can expect to reach 1600 MHz. RDRAM RIMMs can only be used on motherboards or systems specifically designed for them. There are 184 pins for 16-bit RDRAM RIMMs; 32-bit RDRAM RIMMs have 232 pins.



## Conexión de la Ram al CPU

La mayoría de los motherboards están diseñados con **"wait states"**(estados de espera). Estos se usan para adaptar una memoria lenta a un microprocesador rápido: los wait states hacen que el microprocesador espere un poco antes de acceder nuevamente a la memoria. Cero wait state (estado de espera cero) significa que el motherboard no se detiene durante la operación, así no existe pérdida de velocidad. Uno a cuatro estados de espera significa que los cálculos son interrumpidos de una a cuatro periodos de tiempo antes de seguir con la siguiente instrucción.



## Tipos de Memoria (ROM)

Aunque la computadora es operada por el software que se le proporciona a través de los drivers de disco, una pequeña parte de los datos de programación están contenidos en el chip de memoria de sólo lectura (ROM, Read Only Memory). Esta memoria es permanente; no puede ser cambiada y sus contenidos permanecen aún cuando la alimentación eléctrica de la computadora es interrumpida.

El software que reside en el chip de ROM se denomina Basic Input Output System (sistema básico de entrada salida) o ROM BIOS.

11



## Limitaciones en el diseño de la memoria

- ¿qué cantidad? ¿qué velocidad? ¿qué coste?
  - **Cantidad:** según cantidad las aplicaciones la usarán
  - **Velocidad:** mayor rendimiento si  
velocidad memoria == velocidad del procesador
  - **Coste:** razonable en relación con los otros componentes
- Relaciones:
  - A menor tiempo de acceso, mayor coste por bit
  - A mayor capacidad, menor coste por bit
  - A mayor capacidad, mayor tiempo de acceso

12

# Jerarquia de Memorias

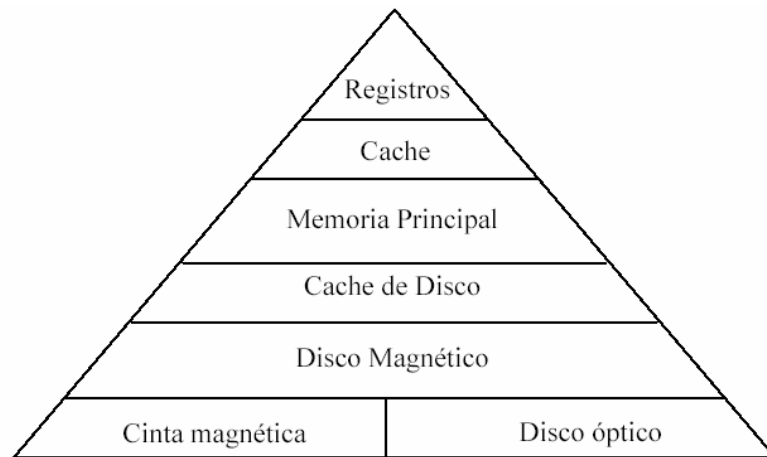


Figura. Jerarquía moderna de memoria

13

# Jerarquia de Memorias

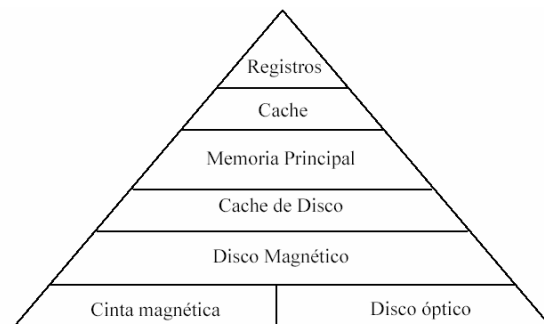


Figura. Jerarquía moderna de memoria

- Bajando por la pirámide...

- ↓ – Disminuye el coste por bit
- ↓ – Se incrementa la capacidad
- ↓ – Se incrementa los tiempos de acceso
- ↓ – Disminuye la frecuencia de acceso a la memoria por el procesador

14



# Jerarquia de Memorias

- Clave del éxito
  - Disminuir la frecuencia de acceso
  - La base: principio “cercanía de referencias”

“Durante el curso de ejecución de un programa, las referencias a memoria por parte del procesador, tanto para instrucciones como datos, tienden a estar agrupadas”

**Motivo:**

Los programas contienen bucles y subrutinas iterativas. Se producen referencias repetitivas a un conjunto de instrucciones. Durante un largo periodo, las agrupaciones cambian, pero en un corto periodo el procesador trabaja con grupos fijos de referencias a memoria

15



# Localidad Espacial y Temporal

El principio de localidad establece que los programas acceden a una parte relativamente pequeña de su espacio de direcciones en cualquier instante de tiempo.

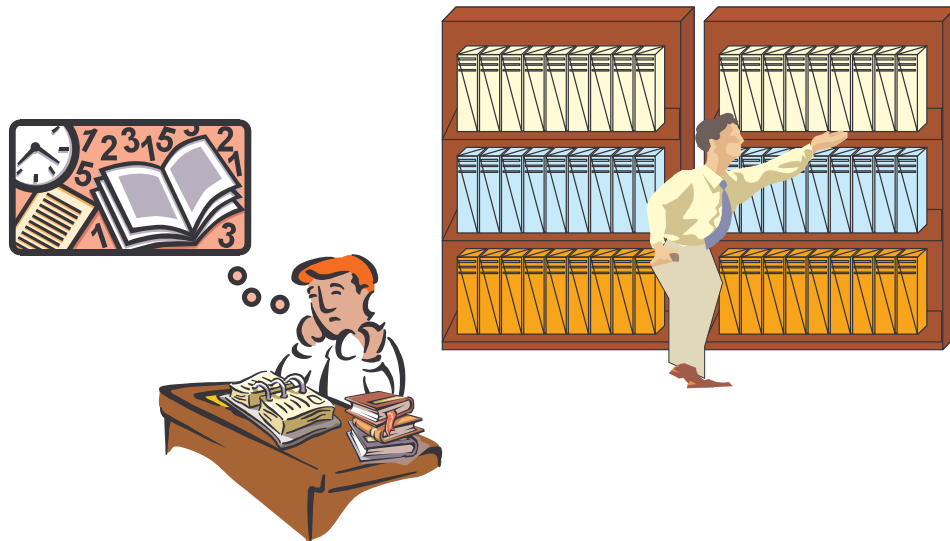
**Localidad Temporal:** Si un elemento es referenciado en un instante de tiempo dado, volverá a ser referenciado pronto.

**Localidad Espacial:** Si un elemento es referenciado, los elementos cuyas direcciones están próximas tenderán a ser referenciados pronto.

16



# Tener más con menos



17

# Memoria Cache

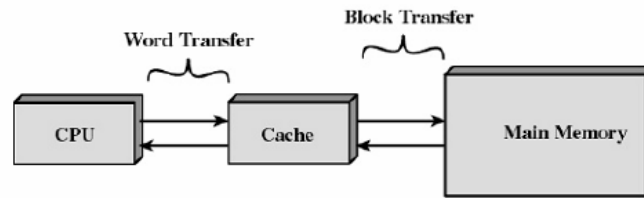
Cache: Un sitio seguro para almacenar cosas

Nuevo diccionario Mundial de Webster de la lengua americana 1988

- Memoria Cache
  - Invisible para el sistema operativo
  - Aumenta la velocidad de la memoria
  - La velocidad del procesador es mayor que la velocidad de la memoria

18

# Memoria Cache



- Contiene una porción de la memoria principal
- El procesador primero comprueba si el dato se encuentra en la cache
- Si no lo encuentra en la cache, el bloque de memoria que contiene la información requerida se mueve a la cache. El procesador lo lee de la cache

19

# Memoria Cache

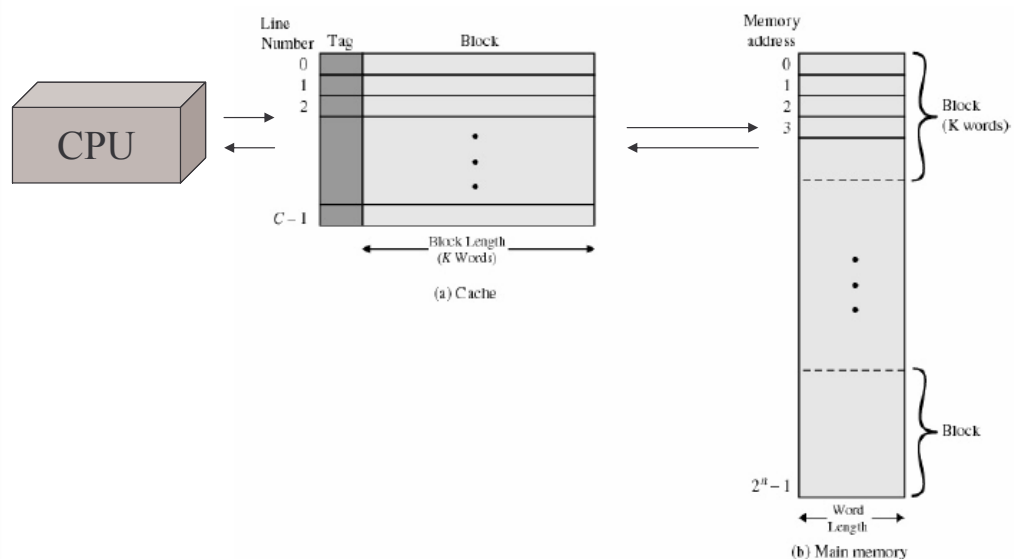
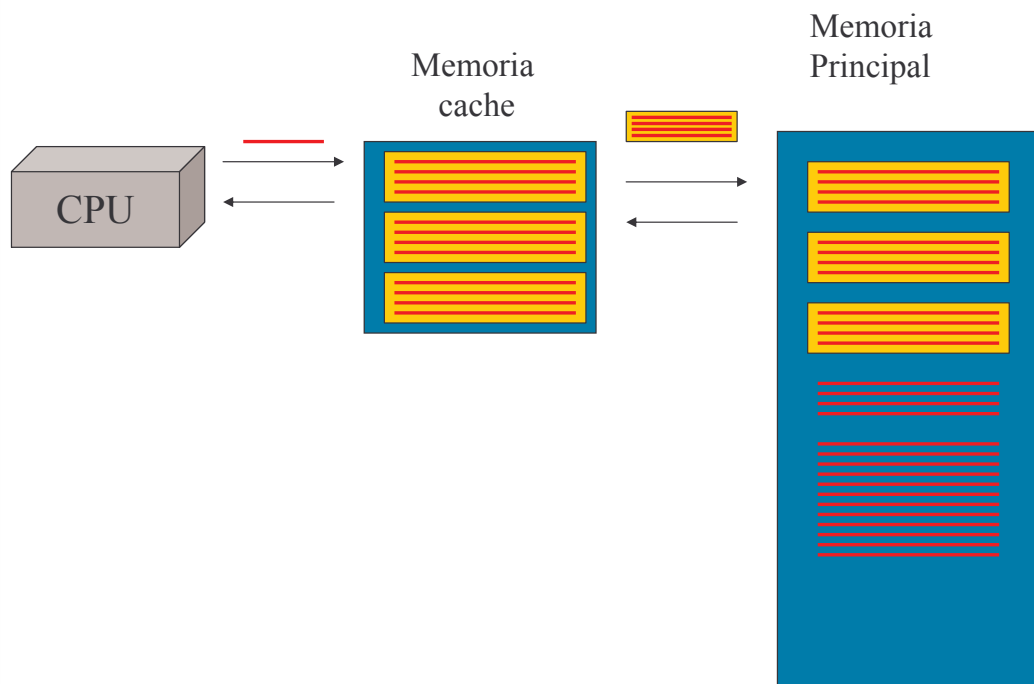


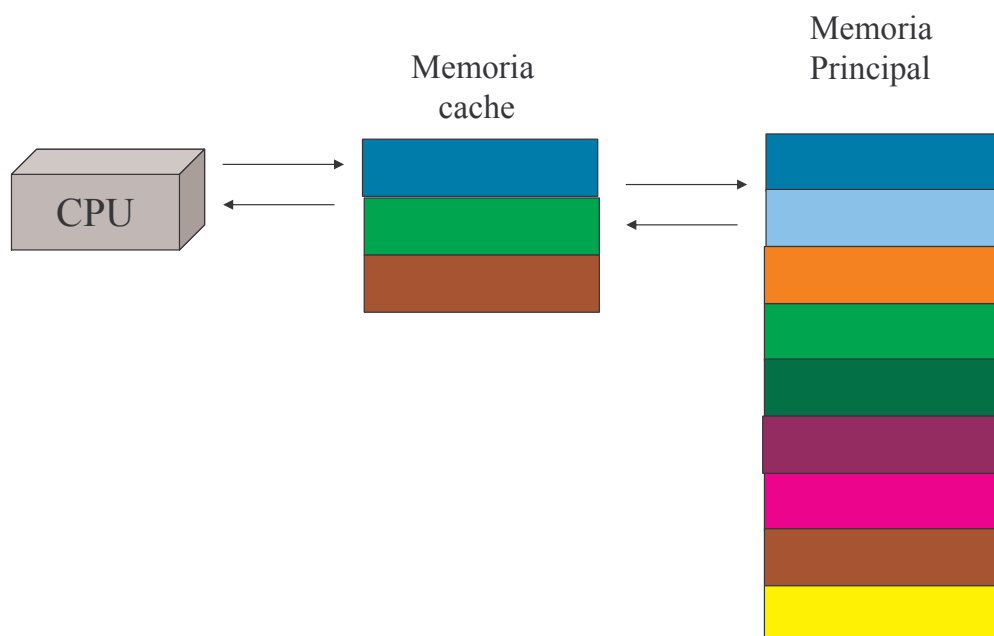
Figure 1.17 Cache/Main-Memory Structure

20

# Memoria Cache



# Memoria Cache



## Tiempo de acceso de un dato

Tiempo acceso de un dato

$$= \text{Tiempo de acceso Cache} * \text{Tasa de aciertos} + \text{Tiempo de acceso RAM} * \text{Tasa de fallos}$$

Tasa de aciertos + Tasa de fallos = 1

Tasa de aciertos (Hit Rate) es la probabilidad de que el dato buscado este en la cache.

Tasa de fallos (Miss Rate) es la probabilidad de que el dato buscado no este en la cache sino sólo en la memoria principal.

Tiempo acceso a cache = 5 nseg,

Tiempo acceso a Mem Principal = 10 nseg,

Tasas de aciertos = 30 %

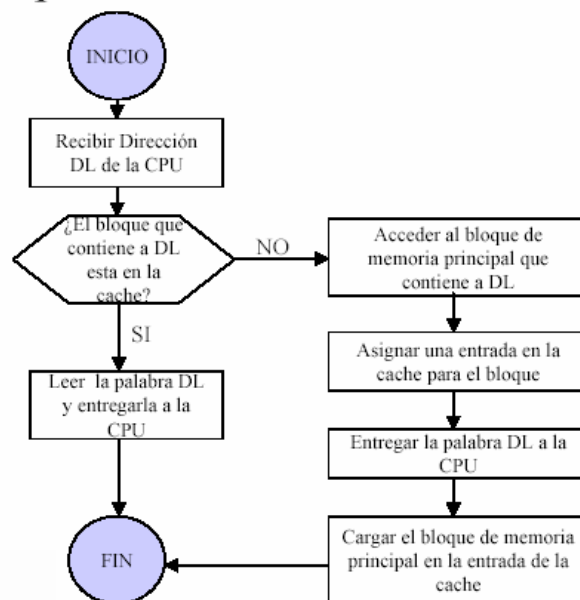
Tiempo promedio de acceso a memoria =

$$5 * 0.3 + 10 * 0.7 = 1.5 + 7.0 = 8.5 \text{ nseg}$$

23

## Memoria Cache

- Funcionamiento. Operación lectura/escritura



# Memoria Cache

- Funcionamiento. Operación lectura/escritura

(1) Recibir dirección DL desde CPU

(2) ¿está el bloque que contiene a DL en Cache?

SI:

(3) Leer DL de la cache y enviar a procesador

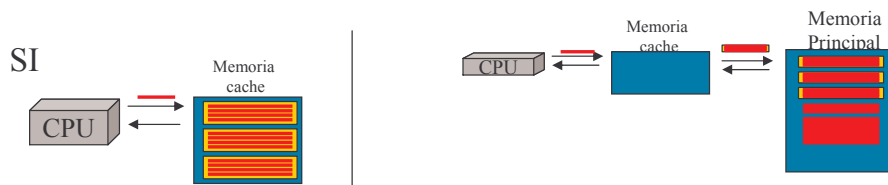
NO:

(3) Acceder a memoria principal en el bloque conteniendo DL

(4) Asignar un grupo en Cache para el bloque

(5) entregar DL al procesador

(6) Cargar el bloque de mem. principal en entrada de la Cache



25

# Memoria Cache

- Diseño de la cache. Aspectos a tener en cuenta

- Tamaño de la cache

- Pequeñas caches tienen un impacto significativo en el rendimiento

- Tamaño de bloque

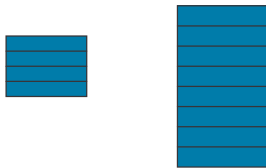
- La unidad de datos utilizada en el intercambio entre la cache y la memoria principal
- Hit (acierto) significa que la información fue encontrada en la cache
- Cuanto más grande es el tamaño de bloque de la cache, pasan a la cache más datos útiles. Sin embargo, la tasa de aciertos disminuye, dado que la probabilidad de que un bloque necesario se saque de la cache es mayor



26

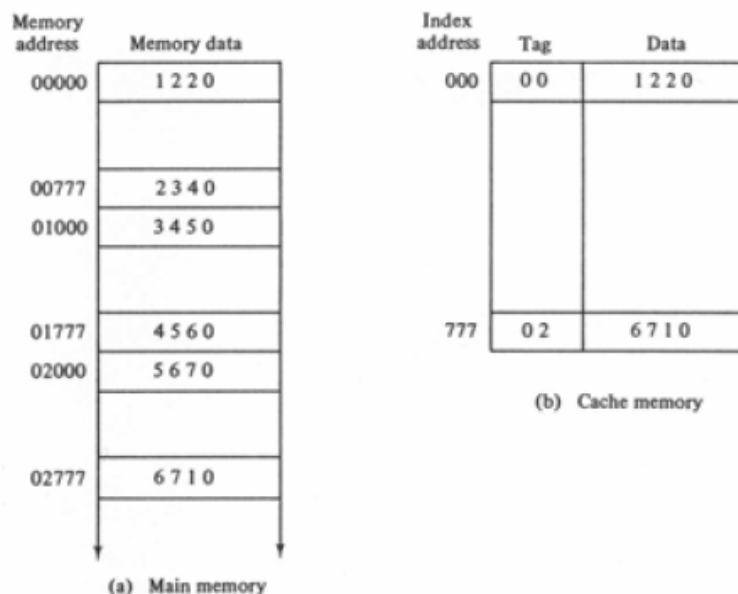
# Memoria Cache

- Diseño de la cache. Aspectos a tener en cuenta
  - Función de mapeo
    - Se determina qué posición de la cache ocupará el bloque
    - Otro bloque puede ser reemplazado. Conviene que el bloque a reemplazar no tenga muchas probabilidades de ser usado
  - Algoritmo de reemplazo
    - Determina el bloque a reemplazar
    - Algoritmo LRU (Least-Recently-Used)
  - Políticas de escritura
    - Cuando la memoria escribe con los datos de la cache
      1. Cada vez que un bloque es actualizado
      2. Cada vez que un bloque es reemplazado
        - » Minimiza operaciones de memoria
        - » Libera la memoria
        - » Deja la memoria en un estado obsoleto



27

## Direct Mapping Cache



28

## Direct Mapped

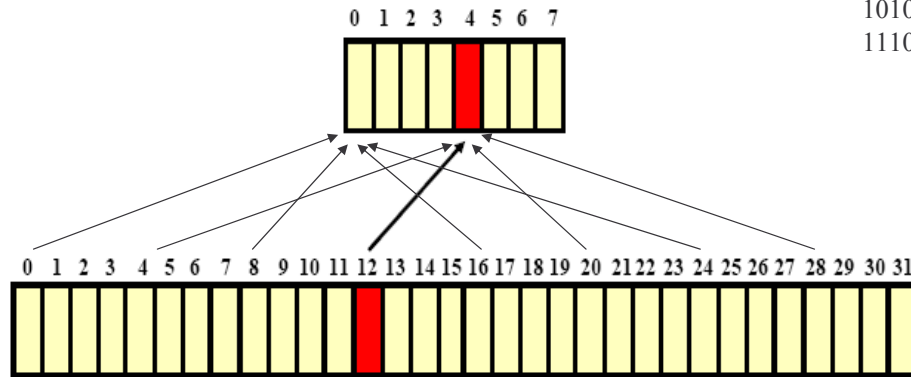
Correspondencia Directa: Correspondencia única para cada entrada de memoria a la Cache

- Each block mapped to exactly 1 cache location

Cache location = (block address) MOD (# blocks in cache)

00000 0  
01000 8  
10000 16  
11000 24

00100 4  
01100 12  
10100 20  
11100 28

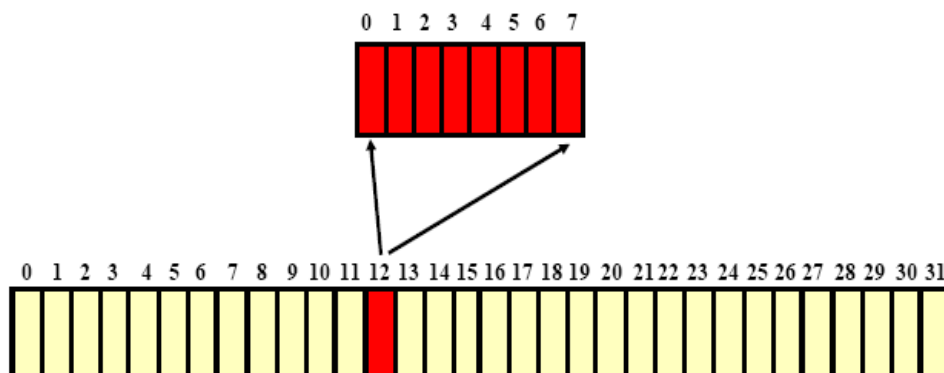


## Fully Associative

Totalmente Asociativa: Cualquier bloque que busquemos podrá estar en cualquier bloque del cache.

- Each block mapped to any cache location

Cache location = any

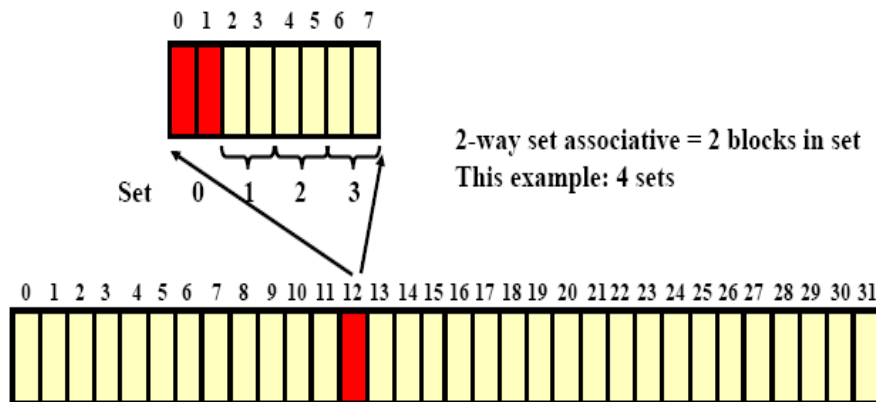


# Set Associative

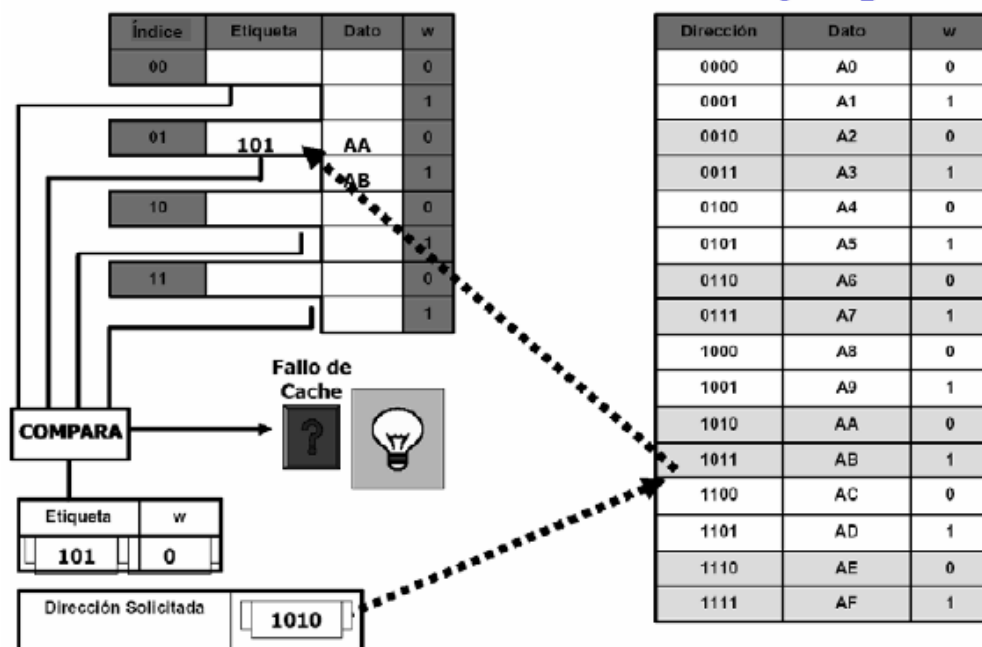
Asociativas por conjunto: existen L entradas que puedan contener la posición de memoria solicitada.

- Each block mapped to subset of cache locations

$$\text{Set selection} = (\text{block address}) \text{ MOD } (\# \text{ sets in cache})$$

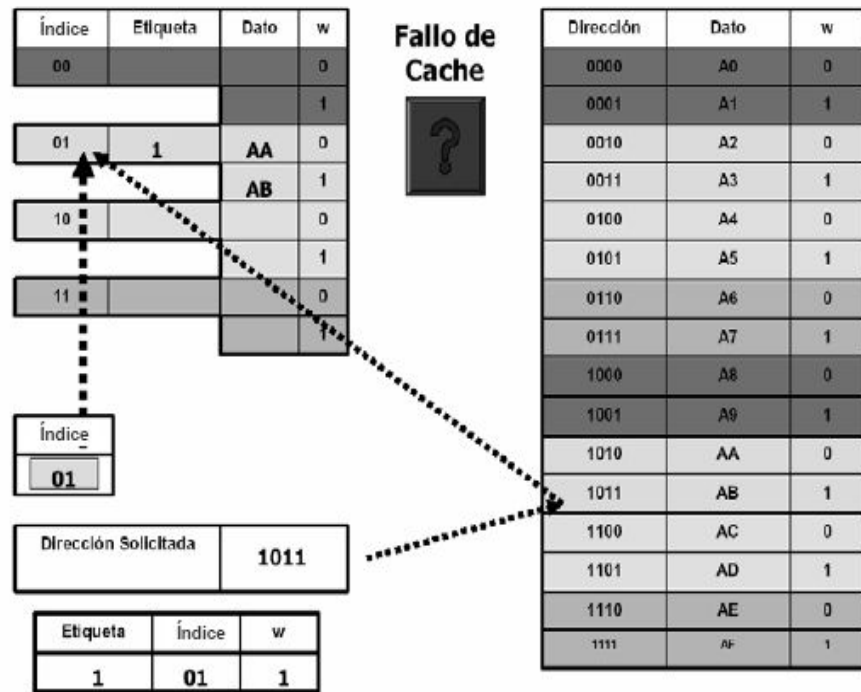


## Caches totalmente asociativas. Ejemplo





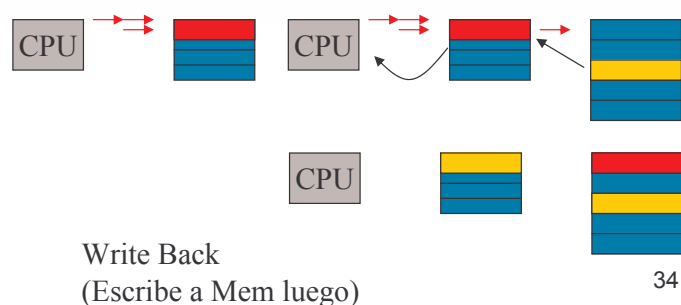
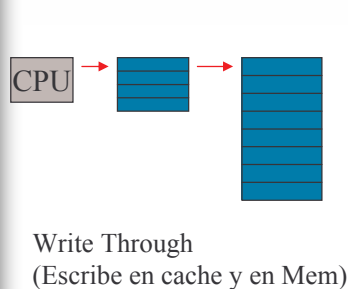
## Caches de mapeado directo. Ejemplo



Arquitectura de Computadores (28) 3

## Memoria Cache

- Diseño de la cache. Aspectos a tener en cuenta
  - Políticas de escritura
    - ¿Cuando la memoria escribe con los datos de la cache?
      1. Cada vez que un bloque es actualizado
      2. Cada vez que un bloque es reemplazado
        - Minimiza operaciones de memoria
        - Libera la memoria
        - Deja la memoria en un estado obsoleto





# Memoria Cache

Mechanism	How It Works	Performance	Coherency Issues
Write Back	Write modified data from cache to memory only when necessary.	Good, because doesn't tie up memory bandwidth.	Can have problems with various copies containing different values.
Write Through	Write modified data from cache to memory immediately.	Not so good - uses a lot of memory bandwidth.	Modified values always written to memory; data always matches.

35



## Memoria Cache (tipos de cache)

La memoria cache forma parte de la tarjeta madre y del procesador (Hay dos tipos) y se utiliza para acceder rápidamente a la información que utiliza el procesador.

Existen cache primario (L1) y cache secundario (L2).

El cache primario esta definido por el procesador y no lo podemos quitar o poner. En cambio el cache secundario (L2) se puede añadir a la tarjeta madre. La regla de mano es que si se tienen 8 Megabytes (Mb) de memoria RAM se debe tener 128 Kilobytes (Kb) de cache. Si se tiene 16 Mb son 256 Kb y si se tiene 32 Mb son 512 Kb. Parece que en adelante no se observa mucha mejoría al ir aumentando el tamaño del cache. Los Pentium II tienen el cache secundario incluido en el procesador y este es normalmente de 512 Kb.

Procesadores más recientes como los Pentium 4 e Itanium llegan a tener 32 kbytes de Cache L1 y hasta 2 Mbytes de L2 y L3

36

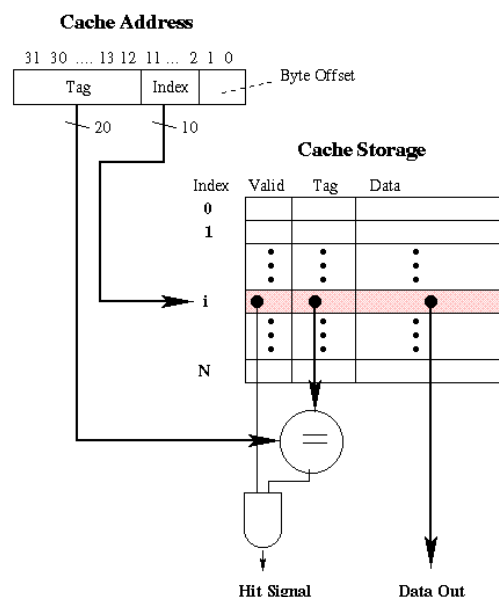
# Memoria Cache (tipos de cache)

La memoria caché de segundo nivel (L2) es una memoria muy rápida llamada SRAM (RAM estática) que se coloca entre la memoria principal y el CPU y que almacena los últimos datos transferidos.

El procesador, como en los casos de caché de disco, primero consulta a dicha memoria intermedia para ver si la información que busca está allí, en caso afirmativo podemos trabajar con ella sin tener que esperar a la más lenta memoria principal.

37

## Memoria Cache



38

# Jerarquía de Memorias

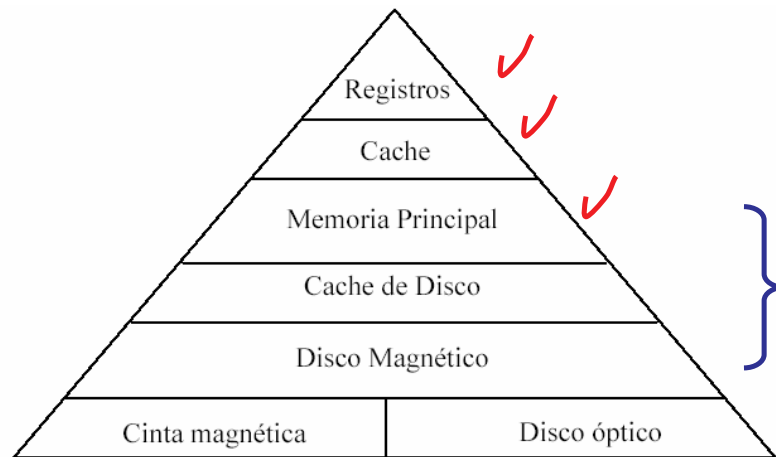


Figura. Jerarquía moderna de memoria

39

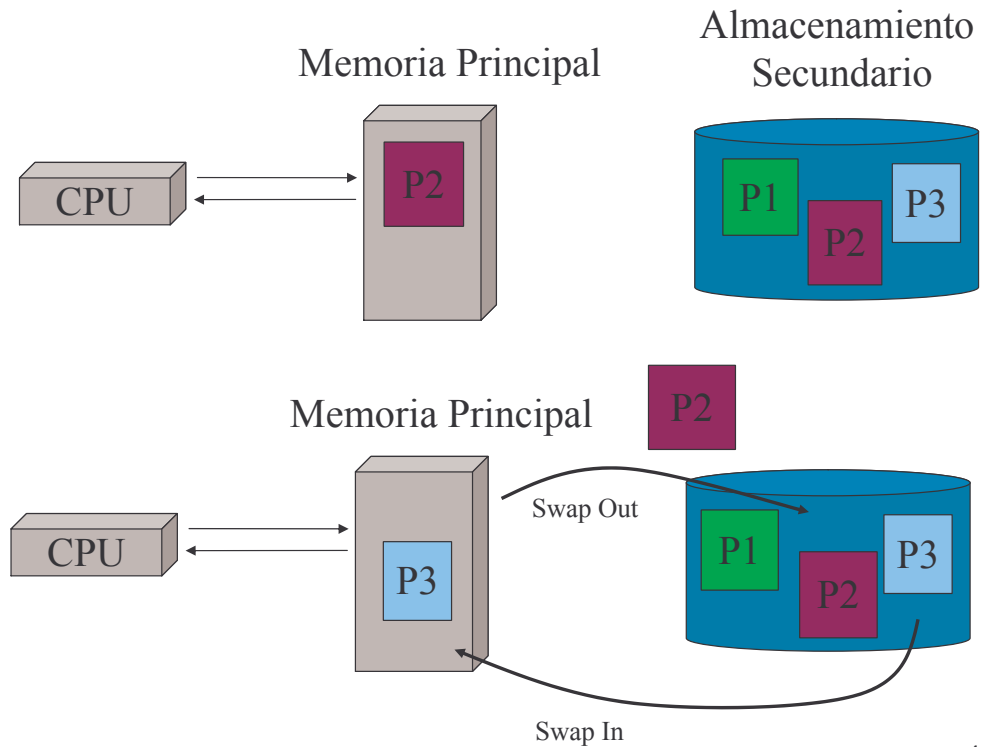
## Memoria Virtual

Es un mecanismo implementado por la mayoría de los sistemas de operación actuales, en la que se da la ilusión a los usuarios de que se cuenta con más memoria Principal de la que realmente existe en el computador.

Se crea para permitir una compartición eficiente de memoria entre múltiples programas y para eliminar los inconvenientes de una memoria principal pequeña y limitada.

40

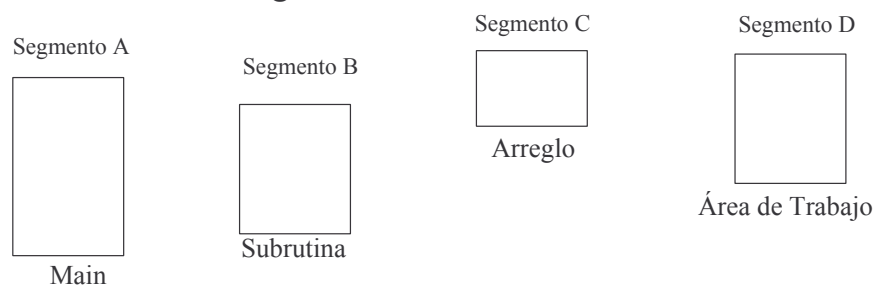
# Memoria Virtual



41

## Memoria Virtual (Segmentación)

Un segmento puede ser definido como una agrupación lógica de información, tal como una subrutina, un arreglo o un área de datos. Cada espacio de direcciones de un trabajo (Job) consiste de una colección de segmentos.

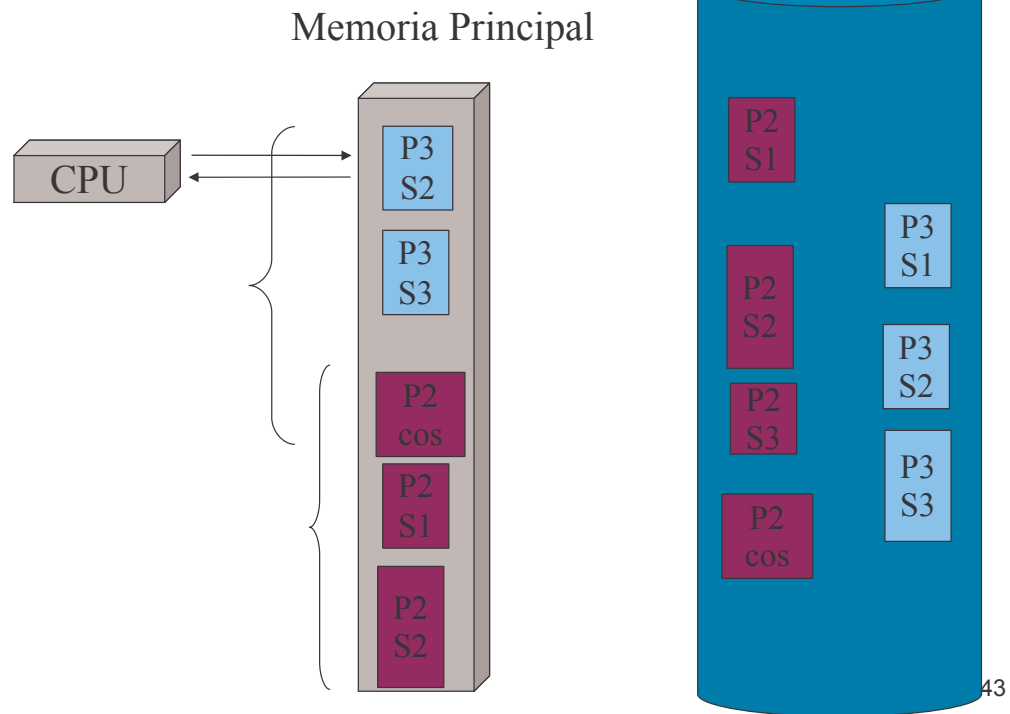


Direcciones

Id del Segmento	desplazamiento en el segmento
-----------------	-------------------------------

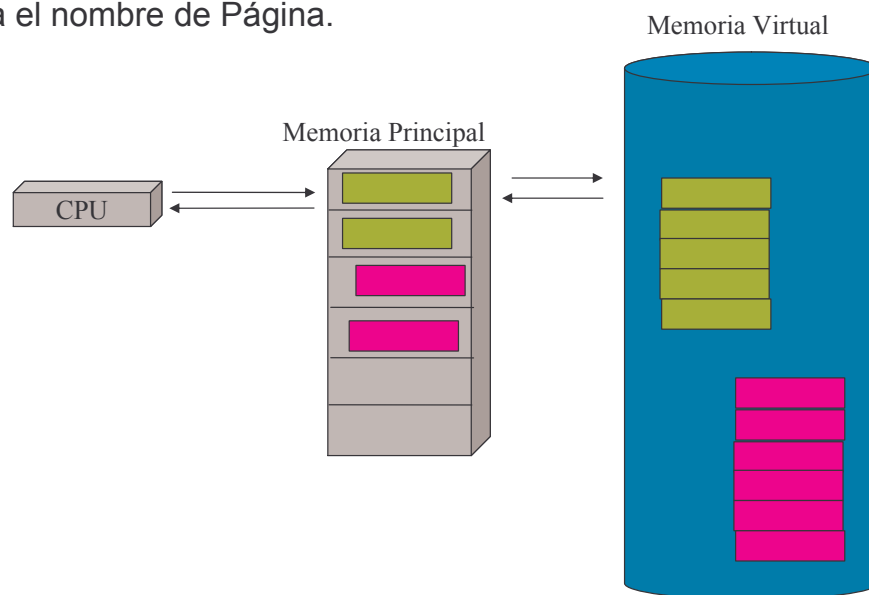
42

# Memoria Virtual (Segmentación)



# Memoria Virtual (Paginación)

Los espacios de direcciones de los programas son divididos en elementos de tamaño fijo, a cada uno de estos elementos se le da el nombre de Página.





# Memoria Virtual (Paginación)

## Dirección Virtual

31 30 29 28 27 ..... 15 14 13 12 11 10 9 8 ..... 3 2 1 0

Número de página	Desplazamiento en la página
------------------	-----------------------------

Traducción

Número de página	Desplazamiento en la página
------------------	-----------------------------

28 27 ..... 15 14 13 12 11 10 9 8 ..... 3 2 1 0

## Dirección Física

45

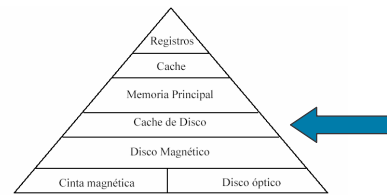


# Políticas de remplazo de página

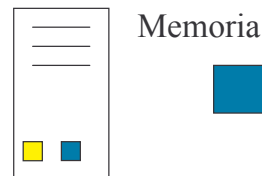
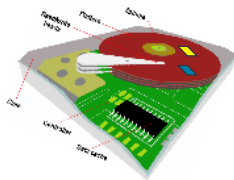
- LRU (Last Recently Used)
- LFU (Last Frequently Used)
- FIFO (First In First Out)

46

# Cache de Disco



- Cache de Disco
  - Una parte de la memoria principal es usada como buffer para almacenar temporalmente datos
  - Una parte de la memoria principal puede ser usado como buffer para almacenar temporalmente datos que deben ser escritos en disco
  - Las escrituras en disco son agrupadas (clustered)
  - Algunos datos escritos fuera pueden ser referenciados otra vez. Los datos son recuperados rápidamente por el software de cache en vez de lentamente del disco



47

# Referencias

- Sistemas Operativos <http://www.monografias.com/trabajos5/sisope/sisope.shtml>
- David y Hennessy, John "Organización y Diseño de Computadores". McGraw Hill. 1995
- Diego Martínez Pomares "Estructuras de los sistemas de computación" <http://www.tlm.unavarra.es/asignaturas/ac/teoria/material/ac.tema2.pdf>
- <http://www.monografias.com/trabajos/acmother/acmother.shtml>
- Donovan John, Madnick Stuart. "Operating Systems". McGraw Hill. 1978.
- Bernie Klinder "Memory Type: Static vs. Dynamic" SearchWinSystems.com. 07 Apr 2005. [http://searchwinsystems.techtarget.com/originalContent/0,289142,sid68\\_gci1073865,00.html](http://searchwinsystems.techtarget.com/originalContent/0,289142,sid68_gci1073865,00.html)

48