



# Gestión de memoria



# Administrador de memoria

- El administrador de memoria es la parte del sistema operativo encargada de gestionar de forma óptima la memoria
- Los programas residen en disco, cuando los ejecutamos, se cargan de memoria
- La gestión de memoria consiste en la **asignación de una memoria física de capacidad limitada a los diversos procesos que la soliciten.**

# Administrador de memoria

## ■ ¿Como funciona el administrador?

- Lleva un registro de las partes de la memoria que están en uso y las que no
- Si detecta que hay una parte de memoria que no está en uso, la libera para asignar memoria a los procesos que sí la necesiten
- Si se dispone de mecanismos de intercambio, decidir qué procesos han de abandonar la memoria principal, cuándo y en qué orden, con la finalidad de dejar espacio para otros procesos.
- Proporciona protección a las zonas de memoria de cada proceso y controla que otros procesos no accedan a ella



# Direcciones de memoria

- Las direcciones de memoria permiten identificar el lugar donde se guardan los datos de la memoria
- La CPU busca los datos en la memoria a partir de la dirección indicada en sus registros internos
- La memoria se encuentra compartida por varios procesos



# Asignación de memoria

Una forma de conseguir multiprogramación consiste en dividir la memoria disponible en partes, cada una de las cuales recibirá un proceso o partes de un proceso.

Si todas las partes de un proceso se cargan juntas en la memoria se dice que **la asignación es contigua**,

Si por el contrario, un programa puede estar en espacios físicamente no continuos, se dice que **la asignación es no-contigua**.

Del primer modelo de memoria revisaremos las particiones estáticas y las dinámicas.

Del segundo caso estudiaremos la paginación y segmentación.



# Particiones estáticas

Consiste en la **división de la memoria en zonas de igual o diferente tamaño**, de forma que en cada una de ellas se pueda ejecutar un proceso diferente. Esta división se realiza al principio (en general en el tiempo de arranque), y las particiones quedan fijas a partir de ese momento.



# Fragmentación de la memoria

- Fragmentación: desaprovechamiento de la memoria
- Podemos distinguir entre fragmentación externa e interna

# Fragmentación

Si un trabajo necesita  $N$  bytes de memoria, podrá ejecutarse en cualquier partición de  $M$  bytes tal que  $M \geq N$ , quedando  $(M - N)$  bytes sin utilizar. A esta memoria sin utilizar se le denomina **fragmentación interna**





# Fragmentación

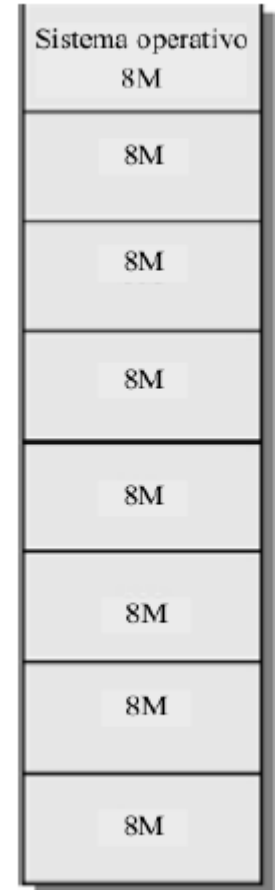
Cuando una partición es demasiado pequeña para contener el programa que está en la cola para ser ejecutado, en cuyo caso el programa no puede ser ejecutado y la partición permanece desocupada. Esto recibe el nombre de **fragmentación externa**.

# Particiones estáticas de igual tamaño

En este caso el tamaño de los bloques es idéntico entre si. Tiene el problema de que **si un proceso ocupa más que el tamaño de la partición no podrá ser ejecutado.**

Además cualquier proceso, por pequeño que sea, ocupará el tamaño de la partición, por lo que **la fragmentación interna es muy alta.**

Como todas las particiones son idénticas, no importa que partición se use. No existe pues ningún algoritmo de asignación para este tipo de particiones estáticas.



# Particiones estáticas de distinto tamaño

Para intentar solucionar estos problemas los bloques que se crean son de distinto tamaño. Ahora se pueden ejecutar procesos de mayor tamaño y la fragmentación interna se reduce. Pero no lo solucionan todo: **no se podrán ejecutar procesos que sean mayores que la mayor partición**, y sigue existiendo un **alto grado de fragmentación interna** y crea además **bastante fragmentación externa**.



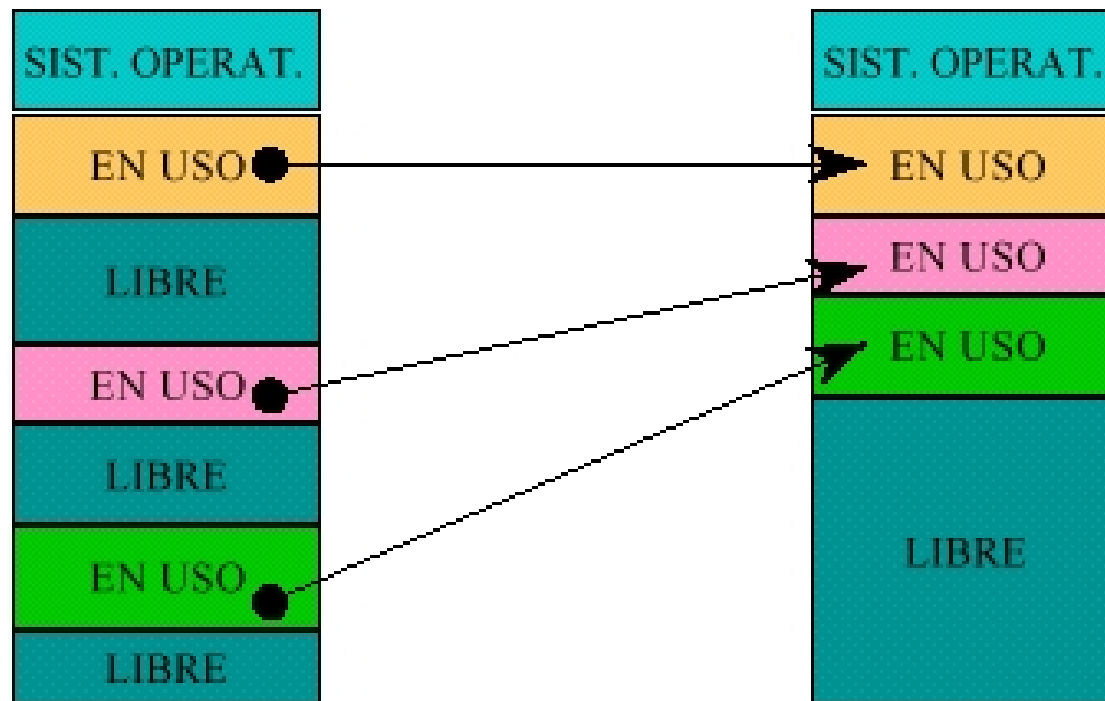


# Particiones dinámicas

- Las particiones dinámicas son variables en número y longitud, esto quiere decir que cuando se carga un proceso a memoria principal se le asigna el espacio que necesita en memoria y no más.
- Esta partición comienza siendo muy buena pero en el transcurso de uso deja un gran número de huecos pequeños en la memoria lo cual se le denomina **fragmentación externa**

# Particiones dinámicas

Se debe usar la **compactación** para evitar esta fragmentación, el sistema operativo desplaza los procesos para que estén contiguos de forma que todos los espacios de memoria libre se agrupen en un bloque, es decir:





# Particiones dinámicas

Existen problemas asociados a la compactación :

- **Problemas de eficiencia** pues el sistema operativo tiene que mover los bloques de memoria y es una tarea que toma una considerable cantidad de tiempo.
- **Las direcciones de memoria cambian** y hay que actualizar todos los punteros a direcciones memoria que se guardan.



# Paginación

La gestión de memoria paginada está basada en dos ideas:

- **La división del espacio lógico de direcciones en trozos del mismo tamaño**
- **La asignación de memoria física no-contigua**, es decir, que los diversos trozos en que ha sido dividido el espacio lógico pueden situarse de forma dispersa a lo largo de la memoria física.



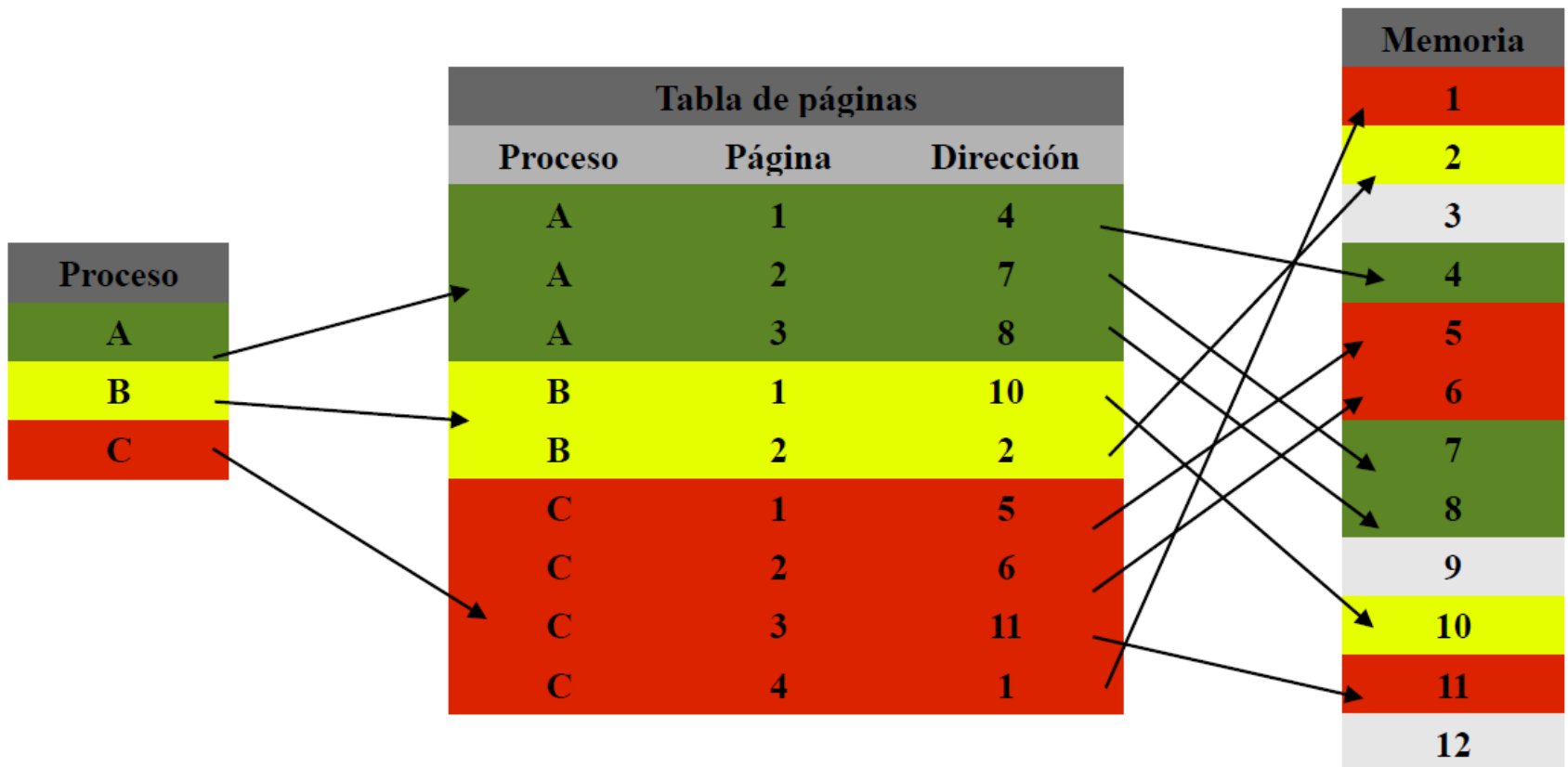
# Paginación

Para dicha gestión se divide el proceso en trozos iguales llamados **páginas**, y la memoria física en zonas de igual tamaño llamados **encuadres, bloques o marcos de página**.



# Paginación

Puesto que cada página lógica se puede ubicar en cualquier página física es necesaria una **tabla de paginas** que establezca, para cada página lógica, cual es el marco de página físico dónde esta localizada.



# Paginación

Cuando hay que asignar un proceso de tamaño  $T$ , el sistema operativo debe asignar  $T/p$  (entero por exceso) encuadres libres, siendo  $p$  el tamaño de página y de encuadre. Observe que se puede producir una **pequeña fragmentación interna en la última página** llamada en este caso fragmentación de página.

El modelo es muy efectivo en cuanto a la **mejora en la utilización de la memoria** (se elimina la fragmentación externa y la fragmentación interna se reduce a la última página del proceso).



# Segmentación

- Técnica similar a la paginación que permite definir los bloques de memoria de tamaño variable
- Con lo cual los segmentos pueden tener longitudes distintas



# Segmentación

La segmentación de memoria es un esquema de manejo de memoria mediante el cual la estructura del programa refleja su división lógica. Llevándose a cabo una agrupación lógica de la información en bloques de tamaño variable denominados segmentos, cada uno de ellos tienen información lógica del programa:

Subrutina, array, etc.



# Segmentación

- Como en la paginación, existe una tabla de segmentos, para convertir las direcciones físicas a lógicas y viceversa
- Funciona de la misma forma que en la paginación
- Con la diferencia de que cada entrada de la tabla de segmentos contiene:
  - el número de segmento
  - la dirección donde comienza el segmento
  - su tamaño

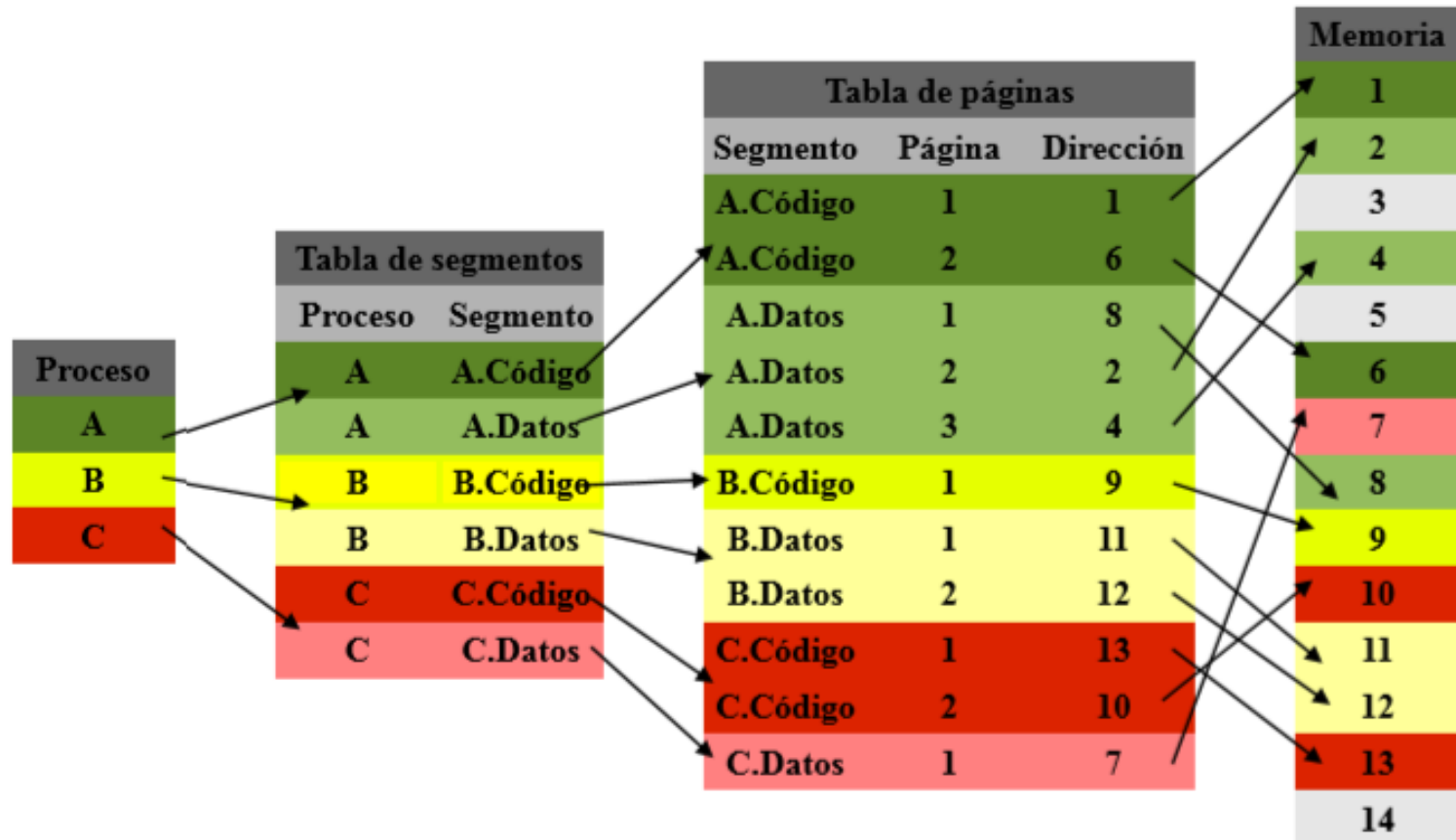


# Paginación segmentada

Una solución más general que intenta aprovechar las ventajas de la segmentación y de la paginación consiste en los modelos de segmentación paginada, en la cual, cada segmento es a su vez dividido en páginas (para facilitar el encaje).

# Paginación segmentada

A cambio, la traducción es más cara puesto que hay que traducir el segmento y a continuación la página dentro del segmento.



# Memoria virtual

- El programa que se va a ubicar en memoria, puede ser excesivamente grande para el tamaño de ésta o para la memoria que queda libre
- Solución:
  - El programa se divide en capas o bloques
  - Las capas o bloques del programa que se están ejecutando permanecen en memoria
  - El resto de bloques del programa que no se estén ejecutando en ese momento se ubican en una zona del disco duro dedicada: memoria virtual





# Memoria virtual

Cuando una instrucción hace referencia a un bloque que no está cargado en memoria, se produce un fallo de página.

El gestor de memoria buscara dicho bloque en disco y cargara el bloque correspondiente en memoria.

Si no existe hueco tendremos que elegir una de las páginas de memoria para sustituirla por la que se solicita. Existen varios algoritmos de reemplazamiento de páginas.

- FCFS Según este algoritmo, cuando hay que dejar libre un marco de memoria, se debe elegir la página que lleva más tiempo en memoria.
- LRU Selecciona la menos recientemente usada
- LFU Selecciona la menos frecuentemente usada.

# Memoria virtual

- Para el usuario el programa está cargado en RAM, aunque realmente sólo está cargado en RAM la parte que se esté ejecutando en ese momento
- Si una parte que no está en memoria se necesita ejecutar, se carga en memoria esta parte del programa que se encuentra almacenada en disco.
- Ventajas: siempre se tiene RAM liberada para realizar otros cálculos o programas

# Memoria virtual

- En sistemas Microsoft se asigna un valor comprendido entre 2,5% y 5% del tamaño del disco duro como memoria virtual.
- SWAPPING Esta zona de intercambio se suele utilizar en sistemas operativos como UNIX y Linux. Está formada por un espacio físico del disco.
- La diferencia entre la gestión de memoria virtual y el intercambio de memoria radica en que mediante la primera puede llegar a ocurrir que el disco duro esté tan lleno que la gestión sea difícil, en el intercambio de memoria no puede ocurrir esto, ya que esta zona siempre está disponible para el intercambio de programas con la memoria principal, puesto que este espacio se gestiona como si fuese una partición del disco duro.