



INSTITUTO POLITÉCNICO NACIONAL ESCUELA SUPERIOR DE CÓMPUTO

INGENIERÍA EN SISTEMAS COMPUTACIONALES

SISTEMAS OPERATIVOS

David Díaz Araujo



INSTITUTO POLITÉCNICO NACIONAL

Práctica 7

Asignación de memoria por paginación

ESCOM®

○ Alumnos:

- 1) Arcos Hermida Aldo Alejandro (N° lista 5)
- 2) Chávez Becerra Bianca Cristina (N° lista 9)
- 3) Islas Osorio Enrique (N° lista 20)
- 4) Juárez Cabrera Jessica (N° lista 21)
- 5) Palmerin García Diego (N° lista 28)

○ Grupo: 4CM1

○ Materia: Sistemas operativos

○ Profesor: Araujo Díaz David

Índice

Objetivo	3
Descripción.....	3
Memoria.....	3
Administración de memoria	4
Memoria Virtual.....	5
Paginación	5
Conclusiones.....	9
Referencias	9

Objetivo

Describir como se realiza la asignación de memoria por paginación en los sistemas operativos.

Descripción

Memoria

La memoria es un componente crucial para el funcionamiento de un sistema operativo. La memoria esta compuesta por bytes, cada uno con su propia dirección. El procesador extrae instrucciones de la memoria de acuerdo con el valor del contador del programa. La instrucción se extrae de memoria y se decodifica para poder ser ejecutada. La unidad de memoria no sabe cómo se generan las direcciones de memoria ni para qué se utilizan, sólo le interesa la secuencia de direcciones generadas por el programa en ejecución.

La memoria principal (RAM) y los registros integrados del procesador son las únicas áreas del almacenamiento a las cuales tiene acceso directo el procesador. Todas las instrucciones en ejecución y los datos utilizados por esas instrucciones deberán encontrarse almacenados ya sea en la memoria principal o en los registros integrados. En el caso que los datos no se encuentren en memoria, estos deben ser llevados ahí para que el procesador pueda operar con ellos.

Generalmente el acceso a la memoria principal se da por medio del bus de memoria, lo cual puede alentar la operación del procesador. La diferencia de velocidad de memoria entre el procesador y la memoria principal se resuelve mediante una memoria rápida entre estas, un búfer de memoria (caché).

Administración de memoria

El sistema operativo a través del administrador de memoria se encarga de administrar parte de la jerarquía de memoria, llevando un registro de la memoria en uso, asignando memoria a los procesos que la necesiten y desasignándola cuando terminen de requerirla.

Existen diversos esquemas de administración de memoria, sin abstracción o con abstracción de memoria. En el esquema sin abstracción de memoria solo se puede ejecutar un proceso a la vez y la memoria consta únicamente de la memoria física. En el caso de la abstracción de memoria, esta se logra mediante un espacio de direcciones, un conjunto de direcciones que puede utilizar un proceso para direccionar la memoria.

Cada proceso debe disponer de un espacio de memoria, para ello debe haber un rango de direcciones a las cuales el proceso tenga permitido acceder y que efectivamente solo acceda a alguna de estas. Esto se logra a través de dos registros: registro base y registro límite. El registro base almacena la dirección de memoria física legal más pequeña y el registro límite especifica el tamaño del rango. Estos registros sólo pueden ser cargados por el sistema operativo, cualquier intento, por parte de un programa ejecutado en modo usuario, de acceder a la memoria del sistema operativo o de otros usuarios provocaría un error fatal.

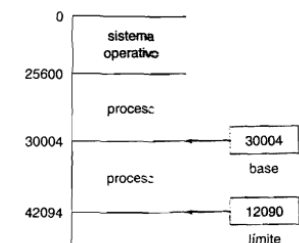
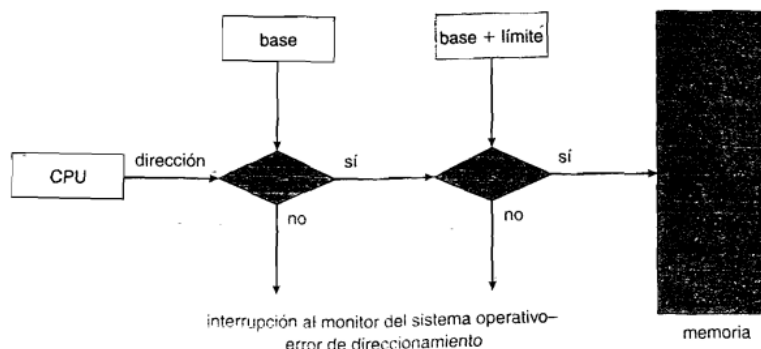


Ilustración 1 Espacio lógico de direcciones definido por el registro base y registro límite



Memoria Virtual

Dada la necesidad de ejecutar programas demasiado grandes para caber en la memoria y de tener sistemas que puedan soportar varios programas ejecutándose a la vez, se buscó la forma de dividir programas grandes en partes modulares más pequeñas que se mantienen en disco, pero pueden pasarse a memoria para ser ejecutadas en caso de ser necesario. Esto se logra con la memoria virtual, la cual libera al usuario de las restricciones de la memoria principal. La idea detrás de la memoria virtual es que cada programa tiene su propio espacio de direcciones, el cual se divide en páginas. Cada página es un rango continuo de direcciones asociadas a la memoria física.

Paginación

La paginación es un esquema de gestión de memoria utilizado para conseguir memoria virtual. En el esquema de paginación, se descompone la memoria lógica en unidades de tamaño fijo llamadas páginas y las unidades correspondientes en memoria física se llaman marcos de página. Las páginas y los marcos de página generalmente son del mismo tamaño, normalmente una potencia de 2 variando entre 512 bytes y 16 MB por página, el cual está definido por el hardware. Las transferencias entre la RAM y el disco siempre son en páginas completas. Este esquema requiere una tabla de páginas, en la cual cada entrada de la tabla indique si la correspondiente página está presente (P) o no en memoria principal. Si el bit indica que la página se encuentra en memoria, entonces la entrada también debe indicar el número de marco de dicha página.

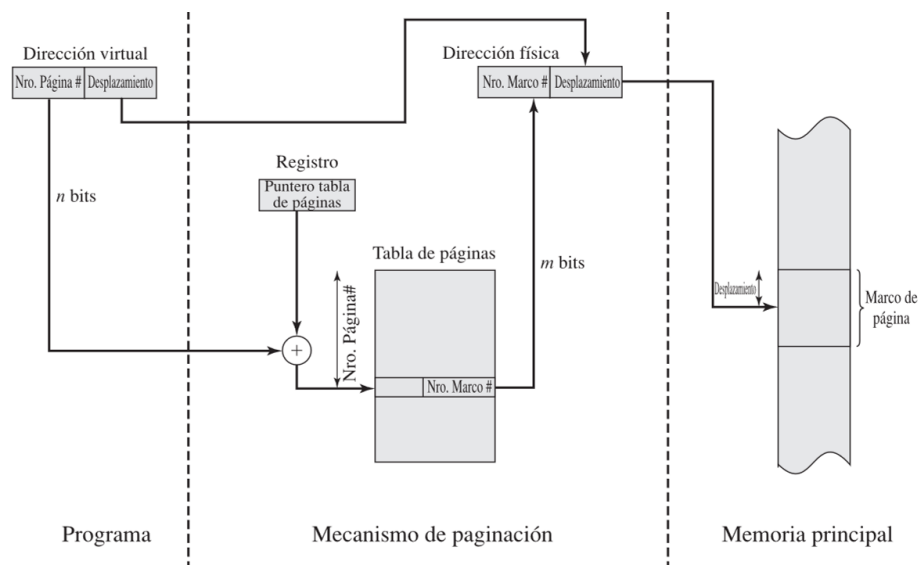


Figura 8.3. Traducción de direcciones en un sistema con paginación.

Práctica 7 : Asignación de memoria por paginación

Cuando se requiere ejecutar un proceso, sus páginas se cargan en los marcos de memoria disponibles. Las direcciones generadas por la CPU consisten en un número de página y un desplazamiento, a la dirección física la cual a su vez consta de un marco y un desplazamiento. El número de página se utiliza como índice para una tabla de páginas. La tabla de páginas contiene la dirección base de cada página en memoria física; esta dirección base se combina con el desplazamiento de página para definir la dirección de memoria física que se envía a la unidad de memoria.

Dirección virtual

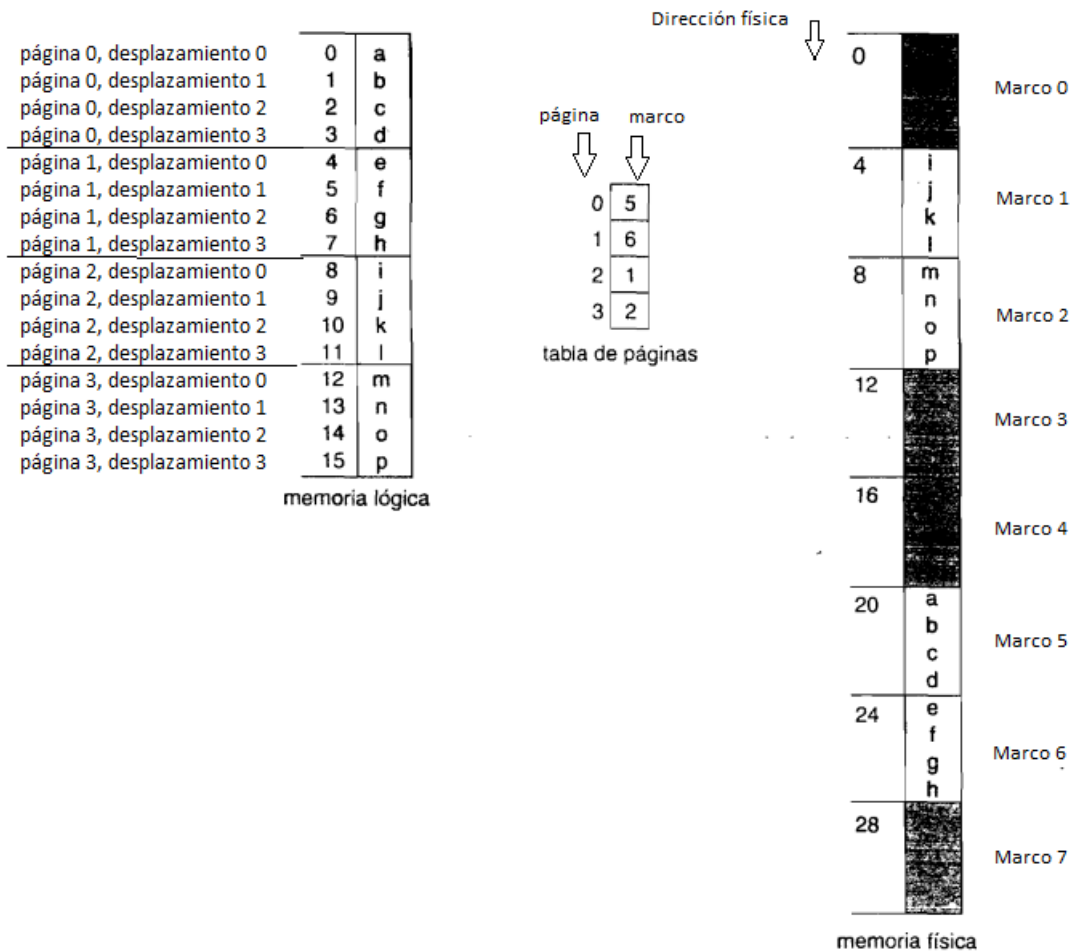
Número de página	Desplazamiento
------------------	----------------

Entrada de la tabla de páginas

P	M	Otros bits de control	Número de marco
---	---	-----------------------	-----------------

P = bit de presente
M = bit de modificado

(a) Únicamente paginación

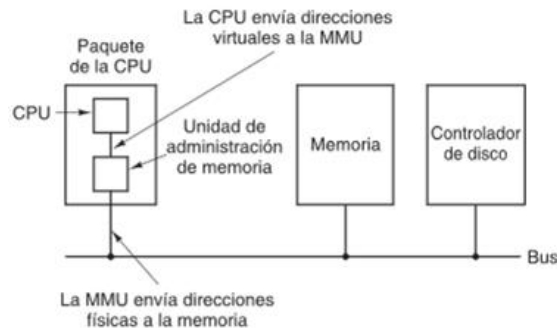


Ejemplo de paginación para una memoria de 32 bytes con páginas de 4 bytes.

La tabla de páginas es de longitud variable, ya que depende del tamaño del proceso. Cuando un proceso está en ejecución, al menos parte de su tabla de páginas debe encontrarse en memoria, incluyendo la entrada de tabla de páginas (ETP) de la página que este ejecutándose. El resto de las tablas de páginas se almacenan en la memoria virtual.

Las direcciones virtuales se envían a la unidad de administración de memoria MMU (memory management unit), que asocia las direcciones virtuales(página) a las direcciones de memoria física (marcos de página).

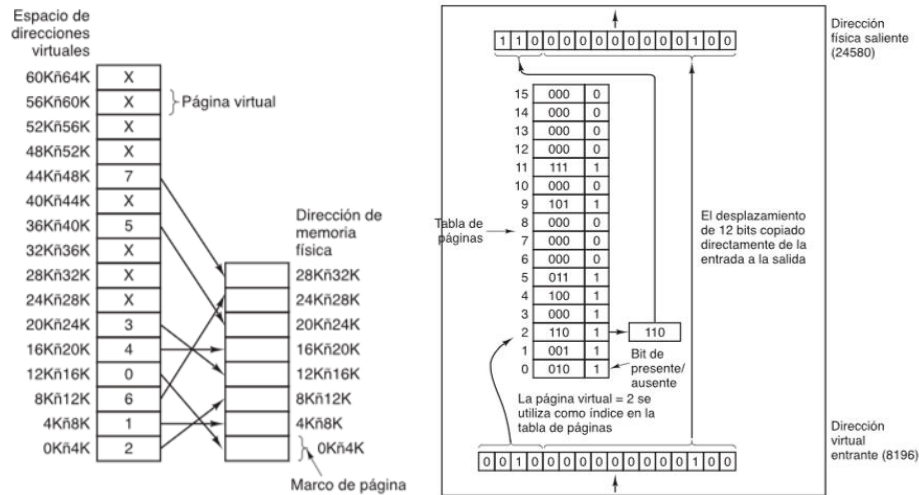
La MMU transforma la dirección a la correspondiente del marco de página y la envía al bus de memoria, para lo cual la memoria solo recibe la instrucción de leer o escribir en esa dirección y lo cumple.



No todas las páginas virtuales se asocian a memoria física, por lo cual es necesario llevar un registro de las páginas que se encuentran físicamente presentes en la memoria. Esto se lleva a cabo a través de un bit de presente/ausente.

Así si la MMU detecta una página que no esta asociada, hace que la CPU haga un fallo de página(trap). Así el sistema operativo selecciona un marco de página que se utilice poco y escribe su contenido de vuelta al disco (si no esta ahí). Después obtiene la página que se acaba de referenciar en el marco de página que se acaba de liberar, cambia la asociación y reinicia la operación que originó el fallo(trap).

Práctica 7 : Asignación de memoria por paginación



Cuando llega un proceso al sistema para ser ejecutado, se examina su tamaño expresado en páginas y cada una de estas necesitará un marco. Si el proceso requiere n páginas, debe haber disponibles al menos n marcos en memoria. Si hay disponibles n marcos, se asignan al proceso que acaba de llegar. La primera página del proceso se carga en uno de los marcos asignados y se incluye el número de marco en la tabla de páginas para este proceso. La siguiente página se carga en otro marco y su número de marco se coloca en la tabla de páginas, y así sucesivamente.

Dado a que el sistema operativo es el que gestiona la memoria física, este debe estar consciente del número total de marcos, aquellos que han sido asignados, cuales están disponibles, etc. Esta información se mantiene en una tabla de marcos, la cual tiene una entrada por cada marco físico que indica si está libre o asignado, en caso de estar asignado, a qué página de qué proceso o procesos ha sido asignado.

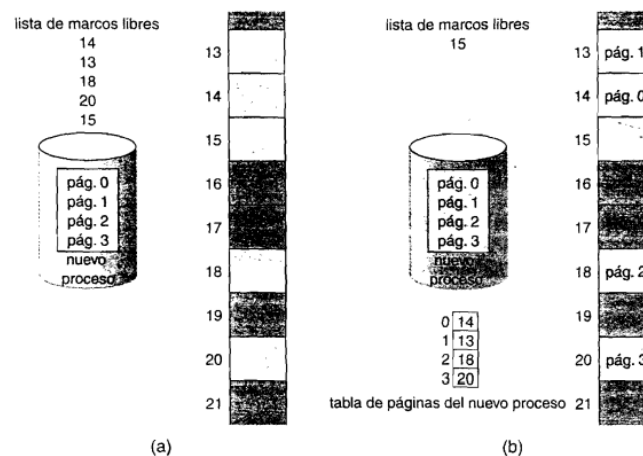


Figura 8.10 Marcos libres (a) antes de la asignación y (b) después de la asignación.

Conclusiones

La memoria es un componente crucial para el funcionamiento de un sistema operativo, un recurso que se asigna y comparte entre diversos procesos. De ahí la importancia de su adecuada gestión para que permita mantener tantos procesos en memoria RAM como sea posible con el fin de mejorar el funcionamiento del procesador. La paginación es un esquema de gestión de memoria utilizado para lograrlo. Este esquema permite que la memoria de un proceso no sea contigua y que un programa se pueda comenzar a ejecutar con tan solo una parte del mismo en memoria y el resto se puede ir cargando conforma se va requiriendo.

Referencias

- Hernandez, A. (2016, 29 de abril). Paginación y segmentación. Recuperado de: <https://sistemasoperativosax.wordpress.com/2016/04/29/paginacion-y-segmentacion/>
- Silberschatz, A. G. (2005). *Fundamentos de sistemas operativos. Séptima edición*. Madrid: McGraw-Hill.
- Stallings, W. (2005). *Sistemas operativos: Aspectos internos y principio de diseño. Quinta edición.* . Madrid: Pearson Educación.
- Tanenbaum, A. S. (2009). *Sistemas Operativos Modernos. Tercera edición*. México: Pearson Educación.