

Paginación

En sistemas operativos de computadoras, los sistemas de paginación de memoria dividen los programas en pequeñas partes o páginas.

Del mismo modo, la memoria es dividida en trozos del mismo tamaño que las páginas llamados **marcos de página**. De esta forma, la cantidad de memoria desperdiciada por un proceso es el final de su última página, lo que minimiza la fragmentación interna y evita la externa.

En un momento cualquiera, la memoria se encuentra ocupada con páginas de diferentes procesos, mientras que algunos marcos están disponibles para su uso. El sistema operativo mantiene una lista de estos últimos marcos, y una tabla por cada proceso, donde consta en qué marco se encuentra cada página del proceso. De esta forma, las páginas de un proceso pueden no estar contiguamente ubicadas en memoria, y pueden intercalarse con las páginas de otros procesos.

En la tabla de páginas de un proceso, se encuentra la ubicación del marco que contiene a cada una de sus páginas. Las direcciones lógicas ahora se forman como un número de página y de un desplazamiento dentro de esa página (conocido comúnmente como *offset*). El número de página es usado como un índice dentro de la tabla de páginas, y una vez obtenida la dirección del marco de memoria, se utiliza el desplazamiento para componer la dirección real o dirección física. Este proceso se realiza en una parte del computador específicamente diseñada para esta tarea, es decir, es un proceso hardware y no software.

De esta forma, cuando un proceso es cargado en memoria, se cargan todas sus páginas en marcos libres y se completa su tabla de páginas.

El único inconveniente de paginación pura es que todas las páginas de un proceso deben estar en memoria para que se pueda ejecutar. Esto hace que si los programas son de tamaño considerable, no puedan cargarse muchos a la vez, disminuyendo el grado de multiprogramación del sistema. Para evitar esto, y aprovechando el principio de cercanía de referencias donde se puede esperar que un programa trabaje con un conjunto cercano de referencias a memoria (es decir con un conjunto residente más pequeño que el total de sus páginas), se permitirá que algunas páginas del proceso sean guardadas en un espacio de intercambio (en memoria secundaria) mientras no se necesiten.

Cuando la paginación se utiliza junto con memoria virtual, el sistema operativo mantiene además el conocimiento sobre qué páginas están en memoria principal y cuáles no, usando la tabla de paginación. Si una página buscada está marcada como no disponible (tal vez porque no está presente en la memoria principal, pero sí en el área de intercambio), cuando la CPU intenta referenciar una dirección de memoria en esa página, la MMU responde levantando una excepción (comúnmente llamada fallo de página).

Si la página se encuentra en el espacio de intercambio, el sistema operativo invocará una operación llamada *intercambio de página*, para traer a memoria principal la página requerida. La operación lleva varios pasos. Primero se selecciona una página en memoria, por ejemplo una que no haya sido usada recientemente (para más detalles ver algoritmos de reemplazo de páginas). Si la página fue modificada, se escribe la misma en el espacio de intercambio. El siguiente paso en el proceso es leer la información en la página necesitada desde el espacio de intercambio. Cuando esto sucede, las tablas para traducción de direcciones virtuales a reales son actualizadas para reflejar los contenidos de la memoria física. Entonces el intercambio de página sale, y el programa que usó la dirección que causó la excepción es vuelto a ejecutar desde el punto en que se dio la misma y continúa como si nada hubiera pasado. También es posible que una dirección virtual sea marcada como no disponible porque no fue localizada previamente. En estos casos, una página de memoria es localizada y llenada con ceros, la tabla de paginación es modificada para mostrar los cambios y el programa se reinicia como en el otro caso.

Si la página que ha provocado el fallo de página no se encuentra en el espacio de intercambio, habrá que ir a buscarla a la unidad de disco. El proceso, por lo demás sería análogo.

Segmentación

Sistema de gestión de memoria en un sistema operativo.

La segmentación es una técnica de gestión de memoria que pretende acercarse más al punto de vista del usuario. Los programas se desarrollan, generalmente, en torno a un núcleo central (principal) desde el que se bifurca a otras partes (rutinas) o se accede a zonas de datos (tablas, pilas, etc).

Desde este punto de vista, un programa es un conjunto de componentes lógicos de tamaño variable o un conjunto de segmentos, es decir, el espacio lógico de direcciones se considera como un *conjunto de segmentos*, cada uno definido por un identificador, y consistente de un punto de inicio y el tamaño asignado.

La segmentación de un programa la realiza el compilador y en ella cada dirección lógica se expresará mediante dos valores: *Número de segmento (s)* y *desplazamiento dentro del segmento (d)*.

Una de las implementaciones más obvias y directas de un espacio de memoria segmentado es asignar un segmento distinto a cada una de las secciones del espacio en memoria de un proceso.

La segmentación también ayuda a incrementar la *modularidad* de un programa: Es muy común que las bibliotecas enlazadas dinámicamente estén representadas en segmentos independientes.

Paginación y Segmentación Combinadas

Tanto la paginación como la segmentación tienen sus ventajas:

La paginación elimina la fragmentación externa y de este modo, aprovecha la memoria principal de forma eficiente. Además, puesto que los fragmentos que se cargan y descargan de la memoria principal son de tamaño fijo e iguales, es posible construir algoritmos de gestión de memoria sofisticados que aprovechen mejor el comportamiento de los programas.

La segmentación tiene las ventajas antes citadas, incluida la capacitación de gestionar estructuras de datos que puedan crecer, la modularidad y el soporte de la compartición y la protección. Para combinar las ventajas de ambas, algunos sistemas están equipados con hardware del procesador y software del S.O.

En un sistema con paginación y segmentación combinadas, el espacio de direcciones de un usuario se divide en varios segmentos según el criterio del programador. Cada segmento se vuelve a dividir en varias páginas de tamaño fijo, que tienen la misma longitud que un marco de memoria principal. Si el segmento tiene menor longitud que la página, el segmento ocupará sólo una página. Desde el punto de vista del programador, una dirección lógica todavía está formada por un número de segmento y un desplazamiento en el segmento. Desde el punto de vista de sistema, el desplazamiento del segmento se ve como un número de página dentro del segmento y un desplazamiento dentro de la página.

La entrada de la tabla de segmentos contiene la longitud del segmento. Los bits de presencia y modificación no son necesarios, puesto que estos elementos se gestionan en la página. Pueden usarse otros bits de control para compartición y protección. La entrada de la tabla de páginas es, la misma que se usa en un sistema de paginación pura. Cada número de página se convierte en el número de marco correspondiente si la página está presente en la memoria. El bit de modificación indica si se necesita escribir la página en el disco cuando se asigna el marco a otra página.

