

Conceptos de Sistemas Operativos

Practica 5

Ramiro Cabral

Contents

1	Memoria fisica y Memoria Virtual	2
2	Asignacion de Memoria	2
3	Fragmentacion	3
4	Paginacion	4
4.1	Direccionamiento	4
5	Segmentacion	5
5.1	Similitudes/diferencias con las particiones dinamicas	5
5.2	Similitudes/diferencias con la paginacion	5
6	Memoria Virtual	6
6.1	Fallos de pagina (Page Faults)	6
6.2	Tamaño de pagina	7
7	Reemplazo de paginas	7
7.1	Alcance del reemplazo	8

1 Memoria fisica y Memoria Virtual

- **Direccion Logica**

- Es una direccion que enmascara o abstraee una direccion fisica.
- Referencia a una localidad en memoria.
- Se la debe traducir a una direccion fisica.

- **Direccion Fisica**

- Direccion real. Con ella se accede efectivamente a memoria.
- Representa la direccion absoluta en memoria principal.

- La traduccion de las direcciones virtuales a direcciones fisicas se realiza mediante la **MMU** (Memory Management Unit).

2 Asignacion de Memoria

- **Particiones fijas**

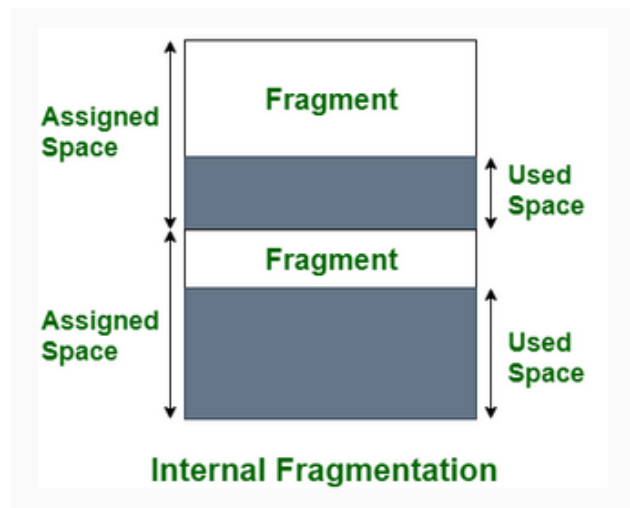
- Se divide la memoria en particiones de tamaño fijo.
- Cada particion aloja un unico proceso.
- Cada proceso se coloca en alguna particion segun a algun criterio.
 - * **First Fit.**
 - * **Best Fit.**
 - * **Worst Fit.**
 - * **Next Fit.**

- **Particiones Dinamicas**

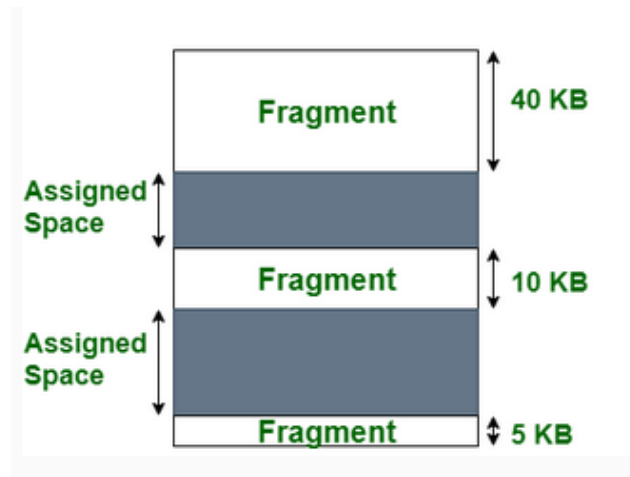
- Las particiones varian en tamaño y en numero.
- Alojan un proceso cada una.
- Cada particion se genera en forma dinamica, y del tamaño justo que necesita el proceso.

3 Fragmentacion

- Se produce cuando una localidad de memoria no puede ser utilizada por no encontrarse en forma contigua.
- **Fragmentacion Interna**
 - Se produce en el esquema de particiones fija.
 - Es interna a la localidad asignada.
 - Es la porcion de la localidad que queda sin utilizar.
 - Diferencia entre la memoria asignada y la solicitada.



- **Fragmentacion Externa**
 - Se produce en el esquema de particiones dinamicas.
 - Son huecos que van quedando en la memoria a medida que los procesos finalizan.
 - Puede darse el caso de que tengamos memoria libre para aloacar un proceso, pero que no la podamos utilizar.
 - Una solucion es la **compactacion**, pero es muy costosa de implementar.



4 Paginacion

- La memoria se divide en porciones de igual tamaño llamadas **marcos(frames)**.
- El espacio de direcciones de los procesos se divide en porciones de igual tamaño llamadas **paginas**.
- Tamaño de pagina = Tamaño de marco (generalmente 512 bytes).
- El SO mantiene una tabla de paginas para cada proceso, que indica en que marco se encuentra cada pagina.

4.1 Direccionamiento

- Un proceso en ejecucion hace referencia a una direccion virtual $v = (p, d)$.
- El SO busca la pagina p en la tabla de paginas del proceso y determina en que marco se encuentra.
- La direccion de almacenamiento real es formada por la concatenacion de la direccion del marco de p y d , donde d es el desplazamiento.
- En este esquema se puede producir fragmentacion interna.

5 Segmentacion

- Se puede ver como una mejora de la paginacion, en la que no hay fragmentacion interna, sino externa.
- Ahora la tabla de segmentos, ademas de tener la direccion del inicio del mismo, tiene la longitud o limite.
- Las direcciones logicas constan de un numero de segmento S y un desplazamiento D dentro del segmento.
- Todos los segmentos de un programa pueden no tener el mismo tamaño.
- El programa es dividido en partes/secciones, donde en cada seccion se guardan datos similares.

5.1 Similitudes/diferencias con las particiones dinamicas

- **Similitudes:**
 - Ambas pueden generar fragmentacion externa.
 - Ambas se basan en la idea de distribuir espacios de tamaño variable a los procesos
- **Diferencias:**
 - El particionamiento dinamico requiere que todo el proceso se encuentre cargado de forma continua en memoria.
 - La segmentacion subdivide y distribuye las divisiones de forma indiferente a su orden real, aunque si mantiene la secuencia dentro de cada segmento.

5.2 Similitudes/diferencias con la paginacion

- **Similitudes:**
 - Ambas tecnicas permiten la distribucion en sectores no contiguos de la memoria.
 - Se requiere de estructuras adicionales.
- **Diferencias:**

- La paginacion es transparente al programador, mientras que la segmentacion no.
- La paginacion elimina la fragmentacion externa.
- La segmentacion facilita modularidad, estructuras de datos grandes y da mejor soporte a la comparticion y proteccion.

6 Memoria Virtual

- Permite ejecutar procesos que requieren mas memoria que la disponible en el sistema.
- Mantiene en memoria principal solo aquella memoria que el procesos este utilizando y el resto en disco.
- El usuario no debe preocuparse por las limitaciones de la memoria fisica.
- El SO debe apoyarse en el hardware, ya que deber ser capaz de detectar cuando se trata de acceder a una direccion que no se encuentra cargada en memoria, y el SO alli podria actuar para atender el fallo.
- Para implementar esta tecnica con paginacion por demanda, se requiere como minimo un bit que indique la presencia de la pagina en memoria.
- Por otro lado, tambien debe contarse con informacion sobre la presencia de modificaciones en las paginas cargadas, ya que si se descarga una pagina modificada, los datos cargados en el disco deben actualizarse.

6.1 Fallos de pagina (Page Faults)

- Se producen cuando una instruccion ejecutada hace referencia a una direccion logica cuya pagina no se encuentra cargada en memoria principal.
- El Hardware de la maquina es el responsable de detectar un Page Fault, generando una excepcion para que el sistema operativo se encargue de cargar la pagina en memoria principal.
- Acciones del SO al producirse un fallo de pagina:
 1. Se genera el trap.

2. El SO bloquea al proceso actual (la CPU toma otro proceso).
3. El SO busca un marco libre en la memoria y genera una operacion de E/S que le pide al disco, para copiar en dicho marco la pagina deseada.
4. La E/s avisa por interrupcion cuando finaliza.
5. El SO actualiza la tabla de paginas del proceso.
6. El proceso pasa del estado bloqueado a listo para ejecutar.

6.2 Tamaño de pagina

- **Tamaño de pagina pequeño:**
 - Menor fragmentacion interna.
 - Mas paginas requeridas por proceso, por lo que se necesita una tabla de paginas mas grande.
 - Mas paginas pueden residir en memoria.
- **Tamaño de pagina grande:**
 - Mayor fragmentacion interna.
 - Mayor eficiencia al mover paginas hacia la memoria principal.

7 Reemplazo de paginas

- Cuando se produce un fallo de pagina, el SO debe decidir que pagina reemplazar.
- El objetivo es minimizar el numero de fallos de pagina.
- **Algoritmos de reemplazo de paginas:**
 - **FIFO:** Se reemplaza la pagina que esta en memoria hace mas tiempo.
 - **LRU:** Se reemplaza la pagina que no se ha utilizado hace mas tiempo. Favorece a las paginas menos recientemente accedidas.
 - **FIFO (Segunda Chance:)**
 - * Se utiliza un bit adicional, llamado bit de referencia.
 - * Cuando la pagina se carga en memoria, el bit R se pone en 0.

- * Cuando la pagina es referenciada, el bit R se pone en 1.
 - * La victima se busca en orden FIFO. Se selecciona la primer pagina cuyo bit R esta en 0.
 - * Mientras se busca a la victima, cada bit R que tiene el valor 1, se cambia a 0.
- Cuando una pagina que fue modificada es reemplazada, se reserva uno o varios marcos para la descarga asincronica de paginas.
 1. La pagina que provoco el fallo se coloca en un frame designado a la descarga asincronica.
 2. El SO envia la orden de descargar asincronicamente la pagina modificada mientras continua la ejecucion de otro proceso.
 3. El frame de descarga asincronica pasa a ser el que contiene la pagina victima que ya se descargo correctamente.

7.1 Alcance del reemplazo

• Reemplazo local

- El fallo de pagina de un proceso solo puede reemplazar sus propias paginas.
- No cambia la cantidad de frames asignados.
- El SO puede determinar cual es la tasa de page-faults de cada proceso.
- Un proceso puede tener frames asignados que no usa, y no pueden ser usador por otros procesos.

• Reemplazo global

- El fallo de pagina de un proceso puede reemplazar la pagina de cualquier proceso.
- El SO no controla la tasa de page-faults de cada proceso.
- Puede tomar frames de otro proceso, aumentando la cantidad de frames asignados a el.
- Un proceso de alta prioridad podria tomar los frames de un proceso de menor prioridad.