

ADMINISTRACIÓN DE LA MEMORIA

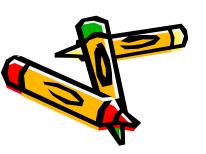
SISTEMAS OPERATIVOS
UNIVERSIDAD DE CARABOBO
FACULTAD EXPERIMENTAL DE CIENCIAS Y TECNOLOGÍA
DEPARTAMENTO DE COMPUTACIÓN
PROF. MIRELLA HERRERA 2010-2011



Administración de la Memoria

Funciones del Manejador de Memoria

- Mantener el recurso memoria. Llevar registro de las partes en uso y desocupadas
- 2. Determinar políticas para decidir cual proceso obtiene memoria, cuando y cuanto
- 3. Asignar memoria cuando el proceso lo requiera
- 4. Reclamar memoria cuando los procesos no la necesiten o hayan terminado



Administración de la Memoria

Dos conflictos

- 1. <u>Separación de los espacios de direcciones</u>: El gestor de memoria debe forzar el aislamiento de espacios de direcciones diferentes para evitar que un proceso activo acceda errónea o maliciosamente y destruya potencialmente los contenidos de los espacios de direcciones de otros procesos
- 2. <u>Compartición de memoria</u>: Permitir que procesos cooperativos accedan a áreas comunes de memoria



Monoprogramación

Se ejecuta sólo un programa a la vez. La memoria se divide entre el programa y el sistema operativo.

User program

Operating system in RAM

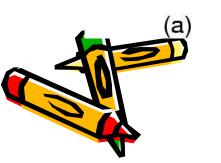
Operating system in ROM

User program Device drivers in ROM

User program

Operating system in RAM

(c)

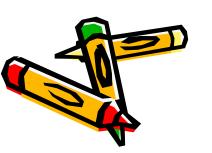


(b)

Multiprogramación

Asignación Contigua: Cada objeto lógico es colocado conjunto de posiciones de memoria con direcciones estrictamente consecutivas. Un método habitual consiste en particionar la memoria física disponible y satisfacer las solicitudes de memoria asignando particiones libres adecuadas, si las hay. Estas particiones pueden ser definidas estática o dinámicamente.

Asignación no contigua: La memoria se asigna de tal modo que partes de un solo objeto lógico pueden ser colocadas en áreas no contiguas de memoria física



Particionamiento Estático: La división de la memoria (número y tamaño de las particiones) permanece fijar y se efectúa durante el proceso de generación del sistema, antes de ejecutar los programas de usuario. Se toma en cuenta la capacidad de la memoria física disponible, el grado deseado de multiprogramación y los tamaños típicos de los procesos frecuentemente ejecutados en la instalación.

Es un método adecuado para entornos estáticos en donde la carga de trabajo es predecible y sus características son conocidas. Ejemplo: entornos de producción en un ambiente bancario, con poco o ningún desarrollo.

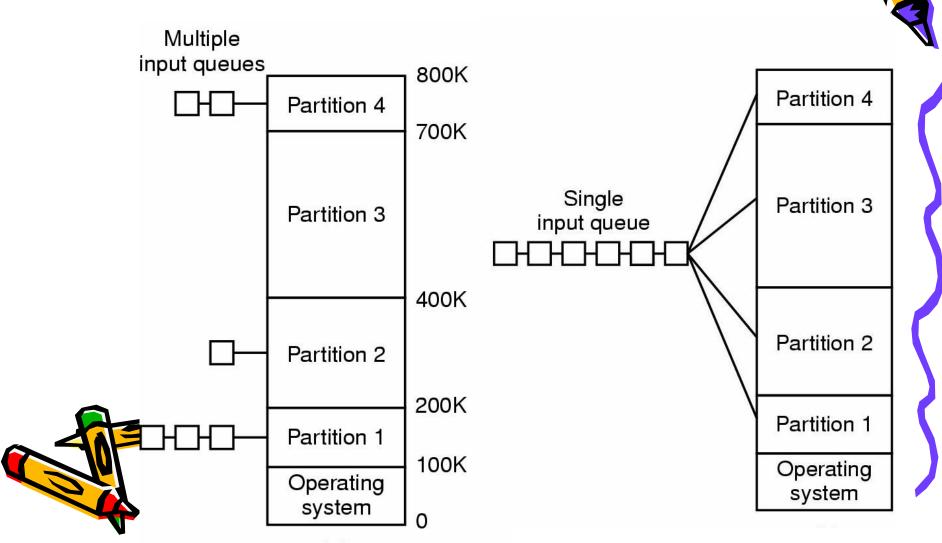
Tabla de Descripción de Particiones TDP

1000k

SO	0K 100K
	400k
A	500k
В	750
C	
	900
	4000:

Nro	Base	Tamaño	Estado
0	Ok	100k	Asig
1	100k	300k	Libre
2	400k	100k	Asig
3	500k	250k	Asig
4	750k	150k	Asig
5	900k	100k	Libre

Asignación Contigua de Memoria Particionamiento Estático:



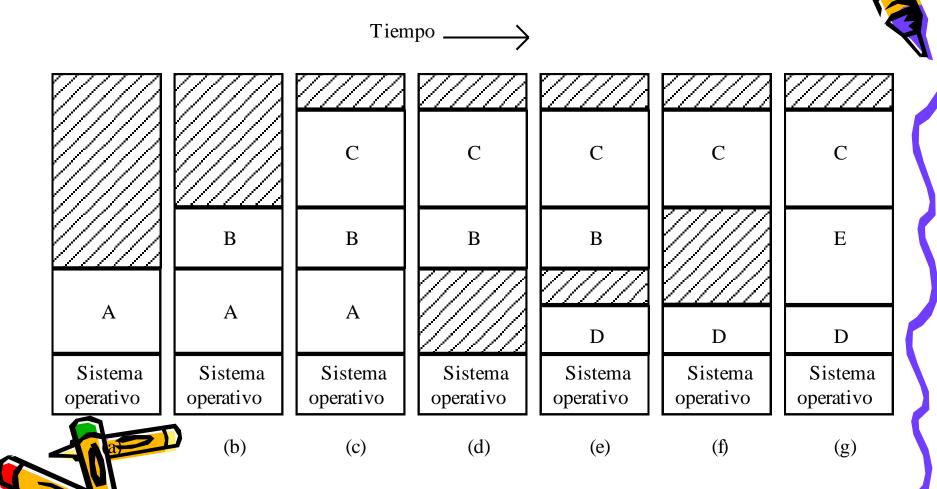
Desventajas de los esquemas:

- Puede suceder que la cola asociada a una partición esté vacía pero la cola de una partición pequeña esté completamente ocupada.
- · Existen varias formas de atender las peticiones:
 - Cada vez que se libera una partición se podría cargar y ejecutar en ellas la tarea que esté de primera en la cola y con tamaño adecuado (primer ajuste).
 - Como no es deseable que se desperdicie una partición de gran tamaño en una tarea pequeña, otra estrategia consiste en buscar en toda la cola de entrada el tamaño que mejor ajuste. En general, las tareas pequeñas son discriminadas para lo cual se puede definir una partición pequeña.
 - Imponer un límite de tiempo para la ejecución de los procesos.



Desventajas del Particionamiento Estático:

- Es inflexible e incapaz de adaptarse a las necesidades cambiantes del sistema
- Presenta fragmentación interna, es decir, se desaprovecha memoria en cada partición cuando se carga un objeto mas pequeño que el tamaño de la partición
- Ningún proceso puede exceder el tamaño de la partición mas grande del sistema
- Limita el grado de multiprogramación (número máximo de procesos activos en el sistema en un momento dado).



Particionamiento Dinámico: Es una asignación adaptativa de la memoria. Se presenta como una respuesta a demandas de usuario. Cada partición tiene el tamaño justo necesario para satisfacer la petición de espacio de un objeto (programa o un área de datos compartida). Es un método adecuado para entornos de desarrollo de programas o donde la carga de trabajo es impredecible.

En general existen tres formas utilizadas por los 50 para administrar el uso de la memoria:

- 1. Mapas de Bits
- Listas enlazadas: Primer Ajuste, Mejor Ajuste y Peor Ajuste.
- 3. Sistema de los asociados o compañeros.

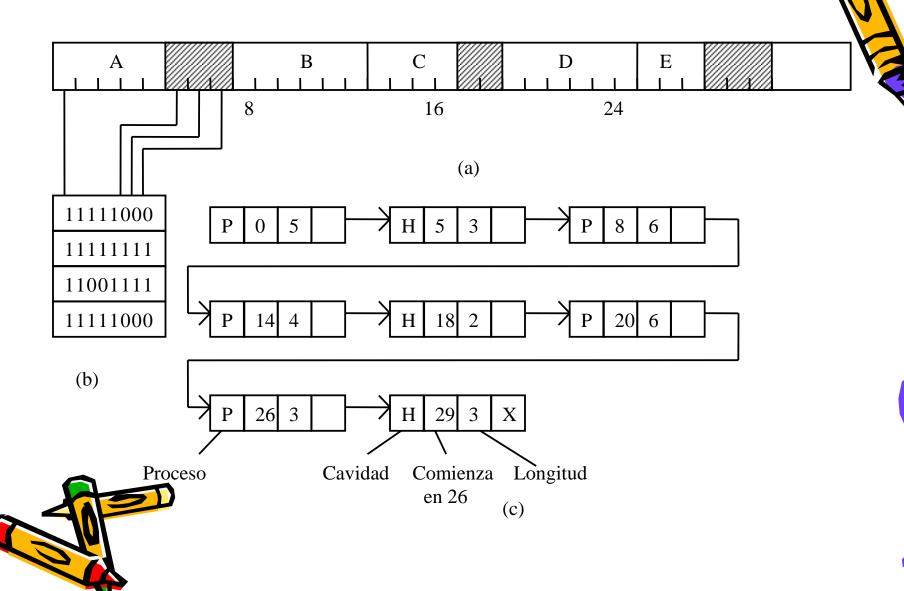
Mapas de Bits: La memoria se divide en unidades de asignación que pueden ser tan pequeñas como unas cuantas palabras o tan grande como unos cuantos kbytes.

A cada unidad de asignación le corresponde un bit en el mapa de bits, el cual toma el valor O si la unidad está libre y 1 si está ocupado.

Mientras más pequeña sea la unidad de asignación más grande será el mapa de bits, si la unidad de asignación es grande el mapa de bits será pequeño pero habrá fragmentación interna

Cuando se requiere un espacio de k unidades, el administrador de la memoria debe recorrer el mapa para encontrar una cadena de k ceros consecutivos.

Desventaja: La búsqueda es una operación lenta por lo que éstos no son utilizados con frecuencia.



Listas Enlazadas:

- ✓ Primer ajuste: el administrador rastrea la lista hasta hallar una cavidad que sea lo suficientemente grande.
- ✓ Siguiente ajuste: igual que el primer ajuste pero empezando en donde se quedó la vez anterior.
- ✓ El que mejor ajusta: busca en toda la lista hasta encontrar la cavidad que ajusta mejor.
- ✓ El que peor ajusta: toma siempre la cavidad más grande disponible.
- ✓ **Desventaja**: Las búsquedas pueden resultar engorrosas y consumir tiempo.

Sistema de los Asociados o Compañeros:

		Memoria						Cav	Cavidades	
	128k 25				56k 384k 512k 640k		768k	896k 1	M	
Inicialmente										1
Solicitud 70	A	1	12	.8	256		512			3
Solicitud 35	A	1	В	64	2.	56	512			3
Solicitud 80	A	1	В	64	C	128		512		
Retorno A	128 B		64	С	128	512			4	
Solicitud 60	D	64	В	64	C	128		512		4
Retorno B	D	64	128		С	128		512		4
Retorno D	256				С	128	512			3
Retorno C	1024								1	

Desventajas del Particionamiento Dinámico

- ✓ La gestión de memoria es más compleja que en el particionamiento estático, consumiento más espacio y tiempo de uso del SO
- ✓ Presenta poca fragmentación interna cuando se usa el sistema de los Asociados o Compañeros.
- ✓ Presenta fragmentación externa, es decir, se desaprovecha memoria entre las particiones, imponiendo una penalidad de tiempo para la compactación.



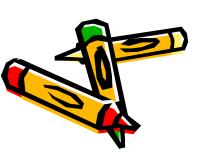
Concepto:

Es un esquema de gestión de memoria en donde puede que sólo una parte del espacio de direcciones virtuales de un proceso "residente" sea cargada realmente en memoria física.

El tamaño permisible del espacio de direcciones virtuales de un solo proceso puede exceder a la capacidad máxima de la memoria física instalada en un sistema dado. Esto se consigue manteniendo una imagen del espacio de direcciones virtuales completo de un proceso en memoria secundaria y trayendo a memoria principal partes de esa imagen cuando sean necesarias

Justificación

- 1. Ciertas partes de un programa no son referenciadas durante una ejecución específica. Ej: Rutinas de manejo de errores.
- 2. Existen caminos de ejecución alternativos que no se extienden por todo el espacio de direcciones, según condiciones externas o internas del programa. Ej: Uso de condicionales y datos de entrada.
- 3. Programación estructurada.



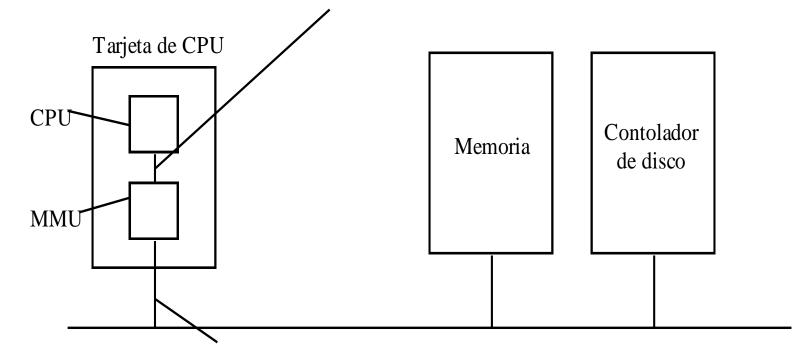
Concepto:

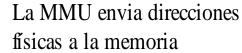
El mecanismo de traducción de direcciones debe ser capaz de asociar nombres virtuales a posiciones físicas. En cualquier instante el hardware de traducción debe realizar la función f: V M tal que

> r si el elemento x está en memoria real en la posición r excepción por elemento ausente si el elemento x no está en memoria real



La CPU envia direcciones virtuales a la MMU

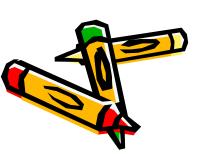






<u>Paginación</u>

- •<u>TMP</u>: <u>Tabla de Mapa de Página</u>: Una por cada proceso activo. Proporciona información sobre la Ausencia/Presencia de una página en memoria principal.
- •TMM: Tabla de Mapa de Memoria. Se usa para llevar la cuenta de las zonas de memoria física que están disponibles para asignación.
- •<u>TMA</u>: <u>Tabla de Mapa de Archivo</u>. Existe una por cada proceso activo. Contiene direcciones de almacenamiento secundario de todas las páginas. Se utiliza para cargar en memoria principal los elementos ausentes. Su base puede ser almacenada en la PCB de los procesos.



<u>Paginación</u>

Solución al problema de la fragmentación externa.

Idea fundamental:

- ·Dividir la memoria física (principal) en bloques iguales de tamaño fijo relativamente pequeños llamados marcos.
- ·La memoria lógica (procesos) se divide en bloques del mismo tamaño llamados páginas.

Ejecución: Las páginas se cargan desde el almacenamiento auxiliar a un marco de memoria que esté disponible.

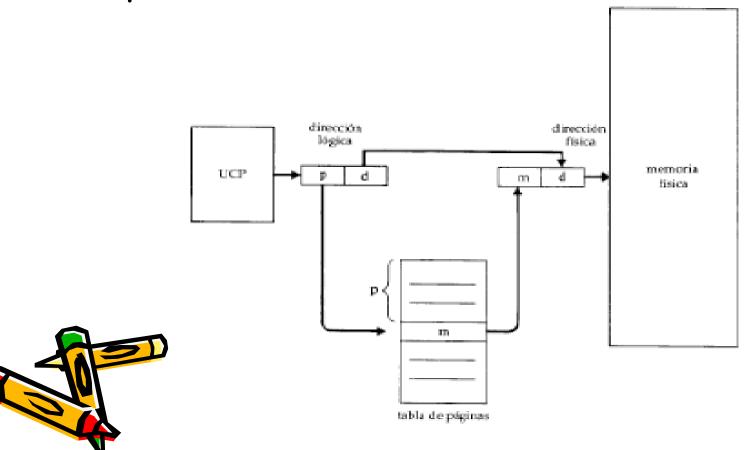
La dirección de la memoria consta de un número de página y de un desplazamiento dentro de la página.



<u>Paginación</u>

Cada dirección generada por la CPU se divide en:

- ·Número de página"p": índice de la tabla de páginas
- ·Desplazamiento dentro de la página "d".







Estructura de la tabla de páginas:

·Denominador común:

Una tabla de páginas para cada proceso.

·¿Cómo localiza el SO la TDP de un proceso?

En la PCB

Contador de instrucciones, registros, información de E/S, etc.

Puntero a la TDP.

·¿Qué ocurre en un cambio de contexto?

El despachador cargará los registros con los valores del nuevo proceso.

A partir de la TDP almacenada, cargará los valores correctos de la TDP en hardware.



Inconveniente de la Paginación: Fragmentación interna.

Sucede cuando las necesidades de memoria de un proceso no coinciden con los tamaños de las páginas.

¿Tamaño de las páginas?

·Pequeño:

- Mejora la fragmentación interna.
- Aumenta el tamaño de la tabla de páginas.

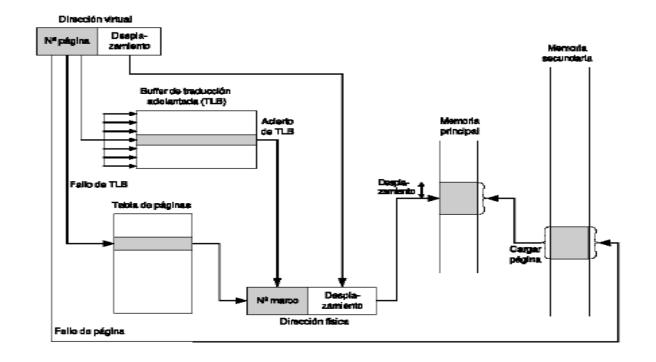
·Grande:

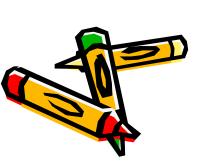
- Peor desde el punto de vista de la fragmentación interna.
- Tamaño de las tablas de páginas menor.
- La E/S de disco es más eficiente cuando la cantidad de datos transferidos es mayor.

·Tendencia en los últimos años:

- Aumentar el tamaño a medida que los procesos, los conjuntos de datos y la memoria principal se han vuelto más grandes.
- 2-4 KB

TLB (Translation Lookahead Buffer, o tabla de registros asociativos): Pequeño caché especial en hardware para el que cada registro se compone de dos partes: clave y valor. Funcionamiento: Se presenta una clave y, si encuentra alguna coincidencia, devuelve el valor correspondiente. Permite búsquedas rápidas pero el hardware es costoso.





TLB (Translation Lookahead Buffer, o tabla de registros asociativos):

Funcionamiento: acceso a la posición i

- · Obtiene el número de página donde se encuentra i.
- Si está en TLB => Obtenemos el marco de página donde se encuentra.
- Sino, acceso a la TMP y actualizar TLB.
- Si TLB llena => Sustitución de una de las existentes.
- Ojo, cambio de contexto:
- Desalojar (borrar) el TLB.
- Tasa de aciertos:
 - Porcentaje de las veces que un número de página se encuentra en los registros asociativos.
 - Buenas tasas de aciertos: 80% -98%
 - Ejemplo: Intel 80486 => TLB de 32 entradas.
 - Sus fabricantes dicen que tiene una tasa de aciertos del 98%

TLB (Translation Lookahead Buffer, o tabla de registros asociativos):

Funcionamiento: acceso a la posición i

- · Obtiene el número de página donde se encuentra i.
- Si está en TLB => Obtenemos el marco de página donde se encuentra.
- Sino, acceso a la TMP y actualizar TLB.
- Si TLB llena => Sustitución de una de las existentes.
- Ojo, cambio de contexto:
- Desalojar (borrar) el TLB.
- Tasa de aciertos:
 - Porcentaje de las veces que un número de página se encuentra en los registros asociativos.
 - Buenas tasas de aciertos: 80% -98%
 - Ejemplo: Intel 80486 => TLB de 32 entradas.
 - Sus fabricantes dicen que tiene una tasa de aciertos del 98%