



Sistemas Operativos 2021-2022

TEMA 4 (segunda parte) Gestión de memoria virtual

- 4.1. Introducción
- 4.2. Implementaciones de la memoria virtual
- 4.3. Políticas de gestión de la memoria virtual

- La gestión de memoria de los SO de la parte anterior tenía un problema
 - El proceso tenía que estar **completamente cargado** en memoria principal para su ejecución
 - No sería posible ejecutar programas con tamaño mayor que aquella
- Es fácilmente comprobable que **no se necesita todo el proceso** en memoria principal constantemente
 - Presencia de bucles en el código
 - El código de gestión de errores se ejecuta poco
 - Estructuras de datos sobredimensionadas
 - Ciertas opciones de los programas se usan rara vez

- La posibilidad de ejecutar un programa **parcialmente cargado**, tiene varias **ventajas**:
 - El programa puede ser mayor que la memoria física
 $|L| > |R|$
 - Podrá haber más programas en memoria simultáneamente
 - **Aumenta el grado de multiprogramación**
 - La operación de intercambio de un proceso (suspender procesos) consume menos tiempo de E/S
 - Al estar ya la mayor parte del proceso en memoria secundaria.



Introducción

Memoria virtual

- Solución: **memoria virtual** (Fotheringham, 1961)
 - Se mantienen en memoria principal sólo las partes en uso de cada proceso
 - El proceso completo reside además en memoria secundaria
 - Es **transparente** al programador
 - Sólo se tiene que concentrar en el problema que resuelve su programa
 - Algo/alguien tiene que conseguir dicha transparencia
 - El sistema operativo y el hardware



Introducción

Memoria virtual

- Dificultades de la memoria virtual
 - El SO y el hardware tienen que **gestionar la carga y descarga de partes del proceso (páginas)**
 - Necesidad de soporte hardware
 - **MMU específica y más compleja**
 - Reinicio de la ejecución de una instrucción ante una falta de página del proceso ☹️ 😊



Implementaciones de la memoria virtual Índice

4.2.1. Introducción

4.2.2. Paginación



Implementaciones de la memoria virtual

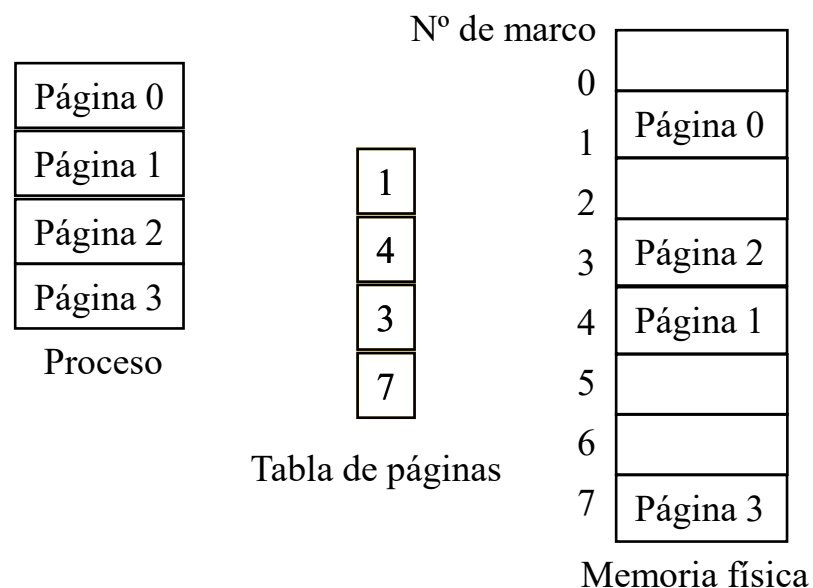
Introducción

- Dependiendo del hardware disponible, se dispone de diferentes modos de implantar una gestión de memoria virtual
 - **Paginación (solo será visto éste en detalle)**
 - Segmentación
 - Segmentación paginada

Implementaciones de la memoria virtual

Paginación

- Los procesos se dividen en bloques de tamaño fijo
 - Páginas
- La memoria física se divide en bloques de tamaño fijo
 - Marcos de página
 - Igual tamaño que el de una página
- **Deben estar en memoria**
 - Páginas que contienen la instrucción en curso
 - Páginas que contienen los datos que maneja ésta





Implementaciones de la memoria virtual

Paginación

- **Organización de procesos y de la memoria principal**
 - Igual que en paginación simple.
- **Organización de las direcciones virtuales y de las direcciones físicas**
 - Igual que en paginación simple.
 - Nótese que las direcciones de los procesos reciben ahora el nombre de *direcciones virtuales*.



Implementaciones de la memoria virtual

Paginación

- **Estructuras de datos** (se amplían las estructuras de datos de *paginación simple*)
 - **Tabla de marcos de página**
 - Igual que en la paginación simple
 - **Tabla de páginas** por proceso
 - N° de marco de página
 - Bits de protección
 - *Bit de modificación* (a 1 si la página se ha modificado)
 - *Bit de presencia* (a 1 si la página está en memoria principal)
 - *Bit de referencia* (a 1 si la página se ha referenciado)
 - **Tabla del mapa del archivo** por proceso (NUEVA)
 - Direcciones en almacenamiento secundario de las páginas del proceso



Implementaciones de la memoria virtual

Paginación

- **Protección**
 - Igual que paginación simple



Implementaciones de la memoria virtual

Paginación

- **Traducción de direcciones virtuales a físicas**
 - Aparece el concepto de “**fallo de página**”
 - Cuando la CPU genera una dirección virtual ésta se divide en página (**p**) y desplazamiento (**d**) dentro de la página
 - A diferencia de paginación simple ahora la página puede o no estar en memoria principal.
 - Si no está en memoria principal se genera una **excepción de fallo de página** y el SO debe arreglar la situación



Implementaciones de la memoria virtual

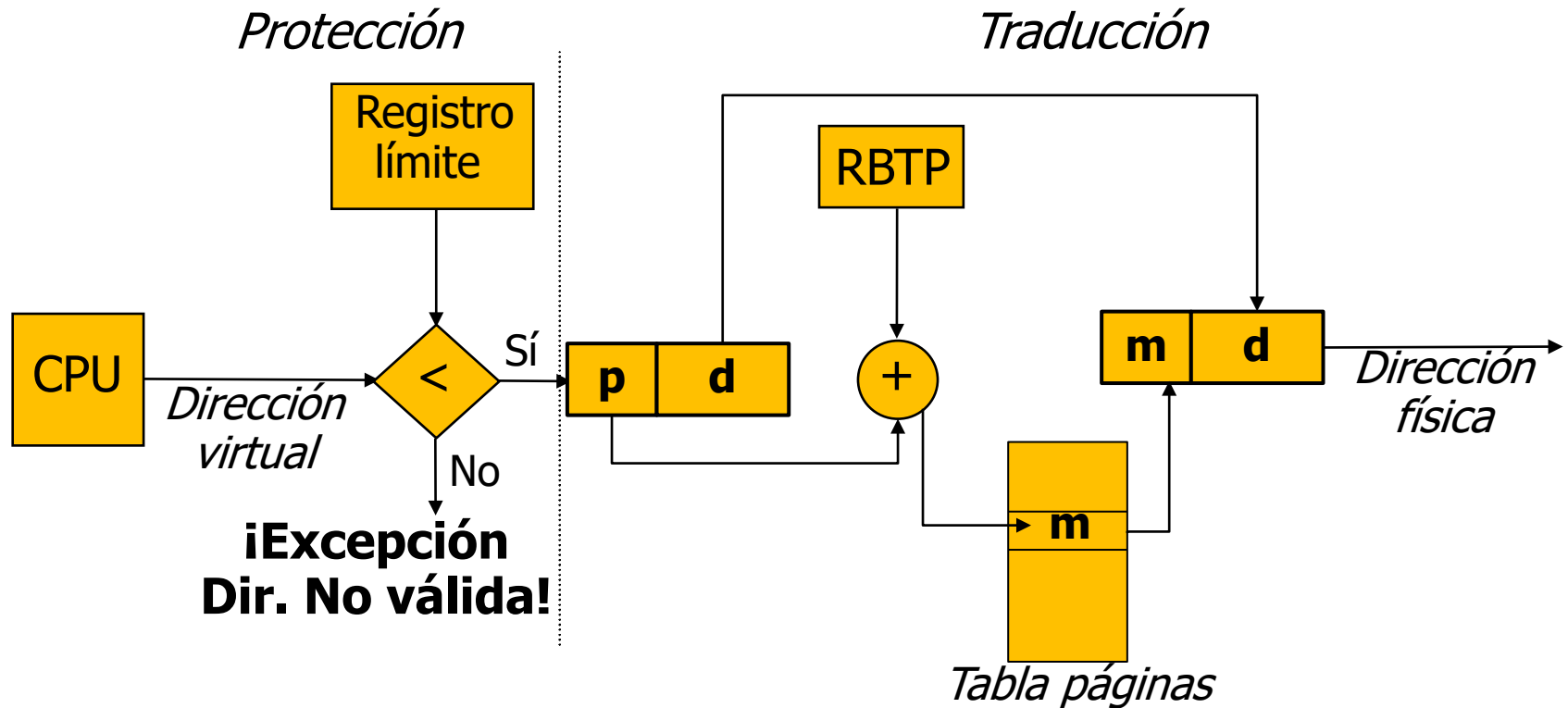
Paginación

- **Traducción de direcciones virtuales a físicas**
(continuación)
 - Hay dos casos posibles
 1. Que NO haya fallo de página (bit de presencia a 1)
 2. Que haya fallo de página (bit de presencia a 0)
 - En el caso 1 la traducción es igual que la de la paginación simple
 - **El caso 2 es más complicado** (se verá con detalle más adelante)
 - Tiene que participar el SO, además de la MMU, para completar la traducción.

Implementaciones de la memoria virtual

Paginación

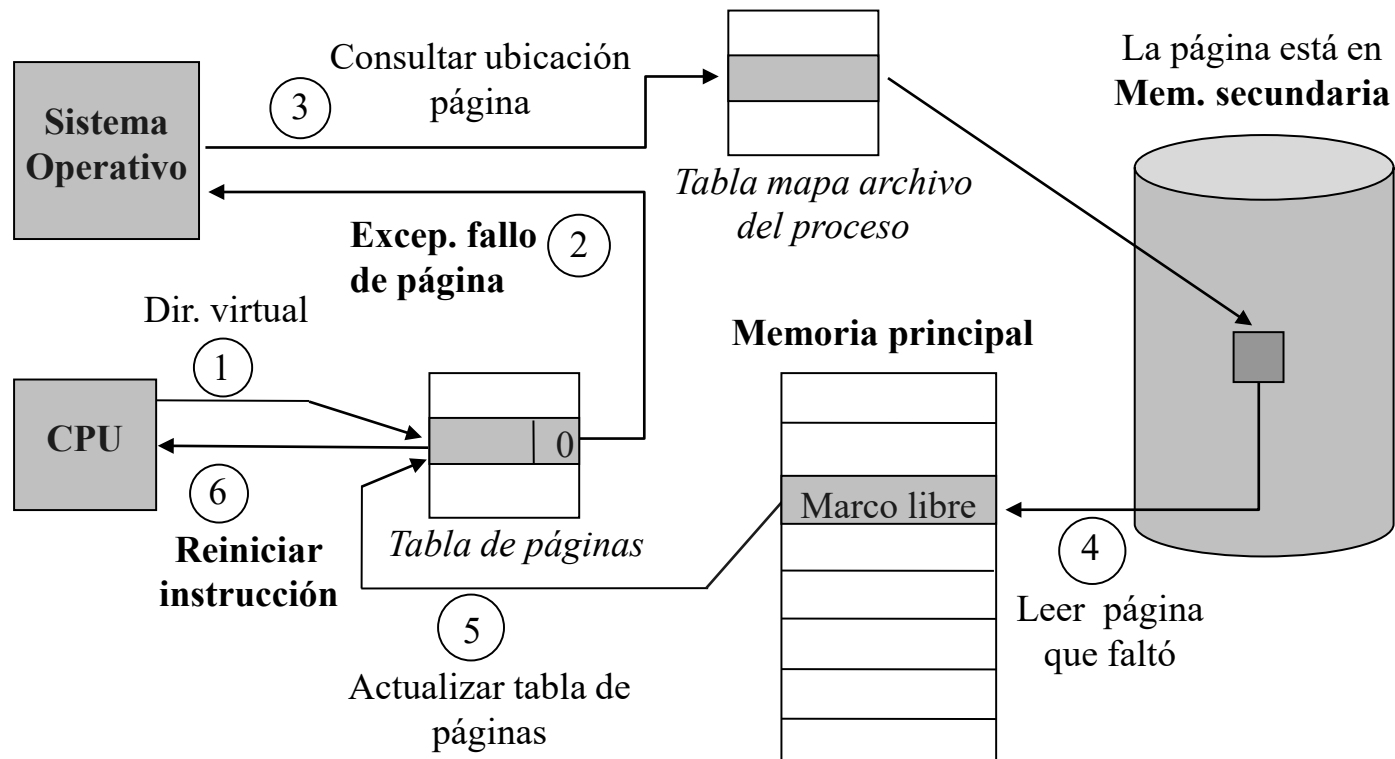
- Caso 1: Traducción de direcciones **SIN fallo de página**



Implementaciones de la memoria virtual

Paginación

■ Pasos en la gestión de un fallo de página





Implementaciones de la memoria virtual

Paginación

- Responsabilidades de la rutina de gestión de la excepción de fallo de página (forma parte del S.O.)
 - Obtener un marco de página libre (varias políticas)
 - Pasar al proceso al estado “Bloqueado”
 - Durante la operación de E/S correspondiente a la lectura de la página que produjo la falta
 - Permite dejar libre el procesador para la ejecución de otro(s) proceso(s) → Multiprogramación
 - Tomar de nuevo el control cuando finalice la E/S
 - Actualizar estructuras de datos
 - Pasar al proceso al estado “Listo para ejecución”



Implementaciones de la memoria virtual

Paginación

- Estructuras modificadas en el esquema de la transparencia 15
 - En (5) el SO modifica:
 - **La tabla de páginas del proceso** que produjo el fallo de página
 - Se indica en la entrada de la tabla de páginas asociada a la página solicitada que dicha página está cargada en el marco elegido, y además:
 - Bit de presencia de la página solicitada a 1
 - Bit de modificación de la página solicitada a 0
 - Bit de referencia de la página solicitada a 0
 - **La tabla de marcos del sistema** para marcar que el marco donde se cargo la página está ahora ocupado



Políticas de gestión de la memoria virtual

Índice

4.3.1. Introducción

4.3.2. Políticas de lectura

4.3.3. Políticas de ubicación

4.3.4. Gestión del conjunto residente

4.3.4.1 Políticas de asignación

4.3.4.2 Políticas de reemplazo



Políticas de gestión de la memoria virtual

Introducción

- Aparte de la técnica de memoria virtual a utilizar (paginación, segmentación, segmentación paginada), necesitamos **software de gestión de memoria virtual**
 - **Forma parte del sistema operativo**
- Existe una serie de **algoritmos** empleados para diferentes aspectos de la gestión de memoria virtual
 - La elección de unos u otros va encaminada a maximizar el rendimiento
 - Los estudios de los algoritmos se aplican, de manera general (en toda la literatura), a la memoria virtual con paginación



Políticas de gestión de la memoria virtual

Políticas de lectura

- Determinan **cuándo** se tiene que cargar una página en memoria
 - **Paginación por demanda**
 - **Sólo se carga una página cuando se referencia**
 - El principio de localidad reduce los fallos de página
 - **Paginación por demanda pura**
 - Los procesos inician su ejecución sin páginas en memoria
 - **Paginación previa**
 - Cuando se produce un fallo de página se carga la página que lo produce y alguna más
 - Mayor eficiencia de carga a costa de más espacio.
 - Difícil precisar qué paginas adicionales cargar
 - **Su utilidad no ha sido demostrada**



Políticas de gestión de la memoria virtual

Políticas de ubicación

- Ya las conocemos de los otros modelos de memoria vistos anteriormente
- Determinan **dónde** va a residir en memoria principal la parte del proceso que esté cargada
 - Para paginación
 - La ubicación carece de importancia



Políticas de gestión de la memoria virtual

Gestión del conjunto residente

- Conjunto residente de un proceso
 - Conjunto (identidad) de páginas del mismo cargadas en memoria principal
- El SO tiene que decidir no solo qué paginas mantener en MP sino **también cuántas** → Tamaño del conjunto residente de cada proceso



Políticas de gestión de la memoria virtual

Gestión del conjunto residente

- Políticas de asignación

- Determinan el **número de marcos de página** asignados a cada proceso → Máximo número de paginas del proceso en MP
- Por el principio de localidad, en cada instante existe un número de marcos asignados ideal
 - Si la asignación real es menor que la ideal, se incrementa la tasa de fallos de página
 - Peligro de **hiperpaginación**
 - Si la asignación real es muy superior a la ideal se reduce esta tasa pero también se reduce el grado de multiprogramación



Políticas de gestión de la memoria virtual

Gestión del conjunto residente

- Políticas de asignación (*continuación*)
 - **Número mínimo** de marcos asignados
 - Número máximo de referencias a páginas de una instrucción máquina + 1
 - Definido por la arquitectura hardware
 - Si no se asignara este mínimo habría instrucciones máquina reiniciándose continuamente



Políticas de gestión de la memoria virtual

Gestión del conjunto residente

- Políticas de asignación (*continuación*)
 - Tipos de políticas de asignación
 - Asignación fija
 - El nº de marcos asignados a cada proceso se decide cuando se crea y no varía en toda su ejecución
 - Poco eficiente ☹
 - Asignación variable
 - El nº de marcos asignados varía dinámicamente
 - Dicho nº debería poder deducirse dinámicamente en base al comportamiento del proceso
 - Más eficiente porque es una asignación adaptativa ☺ → Si el proceso tiene asignados demasiados marcos se le quitan y al contrario.



Políticas de gestión de la memoria virtual

Gestión del conjunto residente

■ Políticas de reemplazo

- Determinan **qué página (víctima) es elegida** para ser reemplazada por otra que se va a cargar
- Esta política solo interviene si hay fallo de página
- La página víctima elegida puede ser O NO del mismo proceso que provocó el fallo de página
- Ante la ocurrencia de un fallo de página, se elige página víctima cuando
 - No hay marcos de página libres
 - Hay marcos de página libres, pero el proceso que produjo la falta no tiene ninguno libre → Alcanzó el máximo determinado por la política de asignación vista anteriormente



Políticas de gestión de la memoria virtual

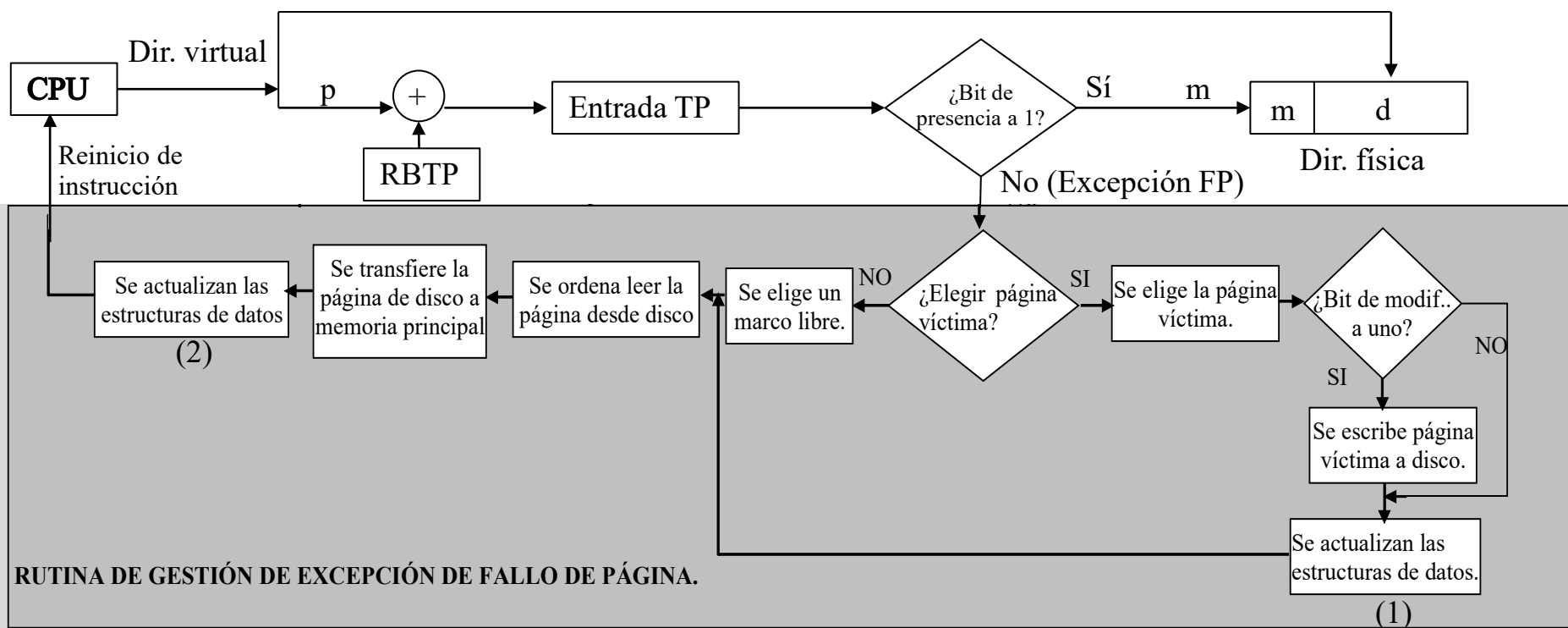
Gestión del conjunto residente

- Políticas de reemplazo
 - Alcance del reemplazo
 - Local
 - La página víctima se elige SIEMPRE de entre las que forman parte del conjunto residente del proceso que produjo el fallo de página
 - Global
 - La página víctima elegida puede ser O NO del mismo proceso que provocó el fallo de página

Políticas de gestión de la memoria virtual

Gestión del conjunto residente

- Esquema de funcionamiento **global** de la paginación virtual (**Solo se refleja la traducción**)





Políticas de gestión de la memoria virtual

Gestión del conjunto residente

- Estructuras modificadas en el esquema anterior:
 - En (1) se modifican
 - **Tabla de páginas del proceso al que pertenece la página víctima**
 - Bit de presencia de la página víctima a 0
 - En (2) se modifican
 - **Tabla de páginas del proceso que produjo la falta de página**
 - Se indica en la entrada de la tabla de páginas asociada a la página solicitada que dicha página está cargada en el marco elegido, y además:
 - Bit de presencia de la página solicitada a 1
 - Bit de modificación de la página solicitada a 0
 - Bit de referencia de la página solicitada a 0
 - **La tabla de marcos del sistema** para marcar que el marco donde estaba la página víctima tiene ahora otra página



Políticas de gestión de la memoria virtual

Gestión del conjunto residente

- Estructuras modificadas en el esquema anterior (*continuación*):
 - El bit de referencia de una página es puesta a 1 por la propia MMU cada vez que se tiene éxito en la correspondencia y se accede a la MP (no se produce una falta de página) y se pone a 0 por el S.O. cada cierto tiempo y siempre que se carga la página a MP.



Políticas de gestión de la memoria virtual

Gestión del conjunto residente

- Políticas de reemplazo
 - Se basan en el principio de localidad de referencias.
 - Algoritmo FIFO (Asignación fija y alcance local)
 - Página víctima: la que lleva más tiempo en memoria
 - Trata los marcos de página asignados al proceso como un buffer circular
 - Ventajas
 - Fácil de entender e implementar
 - Inconvenientes
 - Rendimiento pobre



Políticas de gestión de la memoria virtual

Gestión del conjunto residente

- Políticas de reemplazo (*continuación*)
 - Algoritmo óptimo (Asignación fija y alcance local)
 - Página víctima: la que se va a tardar más tiempo en utilizar
 - Requiere conocer acontecimientos futuros
 - Ventajas
 - Genera el menor número de fallos de página → Solución óptima
 - Inconvenientes
 - No es posible implementarlo
 - Por ello, se utiliza habitualmente como herramienta de comparación



Políticas de gestión de la memoria virtual

Gestión del conjunto residente

- Políticas de reemplazo (*continuación*)
 - Algoritmo de la página menos recientemente usada (LRU) (Asignación fija y alcance local)
 - Página víctima: la que lleva más tiempo sin referenciar
 - Aproximación del algoritmo óptimo
 - Ventajas
 - Se aproxima al óptimo
 - Inconvenientes
 - Implementación costosa (sobrecarga)
 - Porque es necesario recordar la edad de cada página
 - Necesita soporte hardware (**bit de referencia**)



Políticas de gestión de la memoria virtual

Gestión del conjunto residente

- Políticas de reemplazo (*continuación*)
 - Estrategia del conjunto de trabajo (Asignación variable y alcance local)
 - El conjunto de trabajo con parámetro Δ (tamaño de ventana) de un proceso en el instante de tiempo virtual t es el conjunto de páginas referenciadas por el proceso en las últimas Δ unidades de tiempo virtual
 - Aproximaciones del tiempo virtual
 - 1 unidad de tiempo virtual = 1 ciclo de instrucción
 - 1 unidad de tiempo virtual = 1 referencia
 - El conjunto de trabajo es una aproximación de la localidad del proceso
 - A medida que se ejecuta un proceso
 - Varía el tamaño de su conjunto de trabajo
 - Varía el conjunto de páginas que lo forman



Lecturas recomendadas

- Stallings, “Sistemas Operativos”, 5ª edición
 - Capítulo 8, “Memoria virtual”
- Silberschatz, “Fundamentos de Sistemas Operativos”, 7ª edición
 - Capítulo 9, “Memoria virtual”