

TEMA-3-GESTION-DE-MEMORIA.pdf



mrg23



Sistemas Operativos



2º Grado en Ingeniería Informática



Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación
Universidad de Granada



MÁSTER EN

Inteligencia Artificial & Data Management

MADRID

Formamos
talento para un futuro
Sostenible

saber más



Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos Holandés con una garantía de hasta 100.000 euros por depositante. Consulta más información en [ing.es](https://www.ing.es)



TEMA 3: GESTIÓN DE MEMORIA

1. INTRODUCCIÓN

Conceptos:

Tipos de archivos: Se clasifican en distintos tipos según su uso y formato:

- Archivos ordinarios (o regulares): Contienen datos de usuario (texto, binario, multimedia)
- Directorios: Son estructuras que organizan y contienen otros archivos o directorios.
- Archivos especiales:
 - Dispositivos de bloque: Representan dispositivos de almacenamiento como discos.
 - Dispositivos de carácter: Representan dispositivos de E/S como teclados o impresoras.
- Enlaces (link): Referencias simbólicas o físicas a otros archivos.
- Tuberías o FIFO: Permiten la comunicación entre procesos.

Nombres y extensiones en los distintos SO:

- Windows:
 - Usa nombres de archivo con una extensión (.txt, .exe)
 - Longitud máxima: 255 caracteres.
 - Insensible a mayúsculas/minúsculas.
- Linux/Unix:
 - No obliga al uso de extensiones, pero se usan como convención (.sh, .c)
 - Longitud máxima: Generalmente 255 caracteres.
 - Sensible a mayúsculas/minúsculas.
- MacOs:
 - Similar a Linux en estructura.
 - Extensiones opcionales, aunque compatibles con Windows (.app)
- Sistemas antiguos (MS-DOS):
 - Limitación de 8.3 caracteres para nombres/extensiones (archivo.txt).

Estructura de un archivo ejecutable. Depende del sistema operativo:

- Windows (Formato PE - Portable Executable):
 - Encabezado DOS (compatibilidad con MS-DOS).
 - Tabla de secciones (descripción de código, datos y recursos).
 - Información de importación/exportación de funciones.
 - Código máquina:
- Linux (Formato ELF - Executable and Linkable Format):
 - Cabecera ELF (identificación del formato).
 - Tabla de segmentos (instrucciones, datos, pila).
 - Información para enlazadores y cargadores dinámicos.
- MacOs (Formato Mach-O):
 - Cabecera Mach-O (identificación del ejecutable).
 - Comandos de carga (información para inicializar el programa).
 - Secciones de código y datos.

Dirección lógica: Es la dirección generada por el procesador durante la ejecución de un programa. Es independiente del hardware físico.

dirección física: Es la dirección real en la memoria física (RAM). Se obtiene mediante la unidad de gestión de memoria (MMU).

dirección virtual: Es la dirección usada en esquemas de memoria virtual. Los programas operan sobre estas direcciones, que se mapean a direcciones físicas según sea necesario.

Consulta condiciones aquí



do your thing

WUOLAH

OBJETIVOS GENERALES DE LA GESTIÓN DE MEMORIA

Organización: ¿cómo está dividida la memoria?

Gestión: Dado un esquema de organización, ¿qué estrategias se deben seguir para obtener un rendimiento óptimo?

- » Estrategias de asignación de memoria
- » Estrategias de sustitución (determinación de qué zona de memoria elegir para ser sustituida por otra)
- » Estrategia de cuándo buscar/traer procesos a memoria

Protección

INTERCAMBIO (SWAPPING)

Intercambiar procesos entre memoria y un almacenamiento auxiliar

El almacenamiento auxiliar debe ser un disco rápido con espacio para albergar las imágenes de memoria de los procesos de usuario

El factor principal en el tiempo de intercambio es el tiempo de transferencia

El **intercambiador** tiene las siguientes responsabilidades:

- Seleccionar procesos para retirarlos de MP
- Seleccionar procesos para incorporarlos a MP
- Gestionar y asignar el espacio de intercambio

LOCALIZACIÓN DEL ESPACIO DE INTERCAMBIO

Se mantiene un archivo de intercambio global con la información de intercambio de todos los procesos.

Existe un archivo de intercambio para cada proceso.

2. ORGANIZACIÓN DE LA MEMORIA VIRTUAL

a. CONCEPTO MEMORIA VIRTUAL

El tamaño del programa, los datos y la pila puede exceder la cantidad de memoria física disponible para él.

Se usa un almacenamiento a dos niveles:

- **Memoria Principal:** partes del proceso necesarias en un momento dado.
- **Memoria Secundaria:** espacio de direcciones completo del proceso.

Es necesario:

- Saber qué se encuentra en memoria principal.
- Una política de movimiento entre MP y MS.

Además, la memoria virtual

- Resuelve el problema del crecimiento dinámico de los procesos.
- Puede aumentar el grado de multiprogramación.

UNIDAD DE GESTIÓN DE MEMORIA

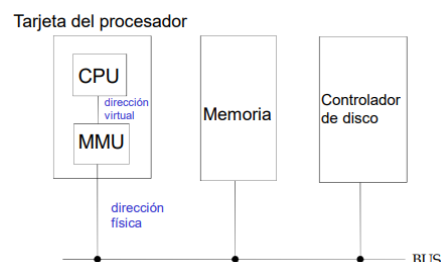
La **MMU** (Memory Management Unit) es un dispositivo hardware que traduce direcciones virtuales a direcciones físicas. Este dispositivo está gestionado por el SO.

En el esquema MMU más simple, el valor del registro base se añade a cada dirección generada por el proceso de usuario al mismo tiempo que es enviado a memoria

El programa de usuario trata sólo con direcciones lógicas; éste nunca ve direcciones reales

Además de la traducción, el MMU deberá:

- Detectar si la dirección aludida se encuentra o no en MP
- Generar una excepción si no se encuentra en MP.



b. PAGINACIÓN

El espacio de direcciones físicas de un proceso puede ser no contiguo.

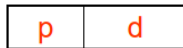
La memoria física se divide en bloques de tamaño fijo, denominados **marcos de página**.

El tamaño es potencia de dos, de 0.5 a 8 Kb.

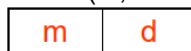
El espacio lógico de un proceso se divide en bloques del mismo tamaño, denominados **páginas**.

Los marcos de páginas contendrán páginas de los procesos

Las **direcciones lógicas**, que son las que genera la CPU se dividen en número de página (**p**) y desplazamiento dentro de la página (**d**)



Las **direcciones físicas** se dividen en número de marco (**m**, dirección base del marco donde está almacenada la página) y desplazamiento (**d**)



Cuando la CPU genere una dirección lógica será necesario traducirla a la dirección física correspondiente, la **tabla de páginas** mantiene información necesaria para realizar dicha traducción. Existe una tabla de páginas por proceso

- **Tabla de ubicación en disco** (una por proceso) ubicación de cada página en el almacenamiento secundario
- **Tabla de marcos de página**, usada por el S.O. y contiene información sobre cada marco de página

CONTENIDO DE LA TABLA DE PÁGINAS

Una entrada por cada página del proceso:

- **Número de marco** (dirección base del marco) en el que está almacenada la página si está en MP
- **Bit de presencia** o bit válido
- **Bit de modificación**
- **Modo de acceso** autorizado a la página (bits de protección)

Nº de página	nº marco	presencia	modificación	protección
46		1	0	01

Ejemplo: Contenido de la tabla de páginas

Espacio de direcciones virtuales

Pag1
Pag2
Pag3
Pag4
Pag5
Pag6
Pag7
Pag8
Pag9
Pag10
Pag11
Pag12

Num. Pagina →

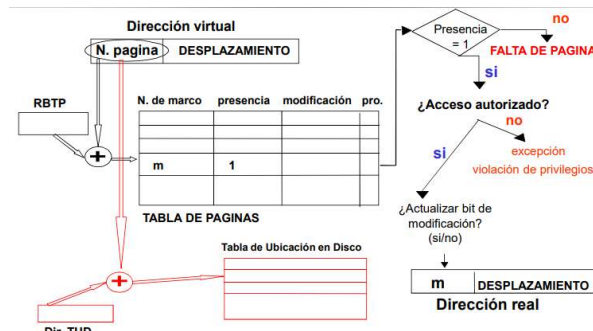
Tabla de Páginas

Nº de Marco	Bit de presencia	
1	8	1
2	-	0
3	1	1
4	2	1
5	-	0
6	-	0
7	3	1
8	3	0
9	-	0
10	5	1
11	-	-
12	10	1

Memoria Física

1	Pag3
2	Pag4
3	
4	
5	Pag10
6	Pag7
7	
8	Pag1
9	
10	Pag12

Esquema de traducción



FALTA DE PÁGINA

1. Bloquear proceso.
2. Encontrar la ubicación en disco de la página solicitada (tabla de ubicación en disco).
3. Encontrar un marco libre. Si no hubiera, se puede optar por desplazar una página de MP.
4. Cargar la página desde disco al marco de MP.
5. Actualizar tablas (bit presencia=1, nº marco, ...).
6. Desbloquear proceso.
7. Reiniciar la instrucción que originó la falta de página.

Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 5/5 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos Holandés con una garantía de hasta 100.000 euros por depositante. Consulta más información en ing.es



IMPLEMENTACIÓN DE LA TABLA DE PÁGINAS

La tabla de páginas se mantiene en memoria principal.

El registro base de la tabla de páginas (RBTP) apunta a la tabla de páginas (suele almacenarse en el PCB del proceso).

En este esquema:

- » Cada acceso a una instrucción o dato requiere **dos accesos a memoria**: uno a la tabla de páginas y otro a memoria.
- » Un problema adicional viene determinado por el **tamaño** de la tabla de páginas.

TAMAÑO DE LA TABLA DE PÁGINAS

Ejemplo:

- » Dirección virtual: 32 bits.
- » Tamaño de página = 4 Kbytes (212 bytes).

Tamaño del campo desplazamiento = 12 bits.

Tamaño número de página virtual = 20 bits.

Nº de páginas virtuales = $2^{20} = 1,048,576!$

Solución: para reducir el tamaño de la TP:

- » Paginación multinivel.

PAGINACIÓN MULTINIVEL

Esta solución opta por "paginar las tablas de páginas".

La partición de la tabla de páginas permite al SO dejar particiones no usadas sin cargar hasta que el proceso las necesita. Aquellas porciones del espacio de direcciones que no se usan no necesitan tener tabla de página

PAGINACIÓN A DOS NIVELES

Lo que hacemos es paginar la tabla de páginas.

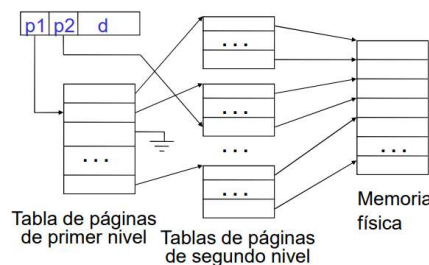
La dirección lógica se divide en:

- » número de página (n bits):
 - un número de página **p1** (=k)
 - desplazamiento de página **p2** (=n-k)
- » desplazamiento de página **d** (m bits).

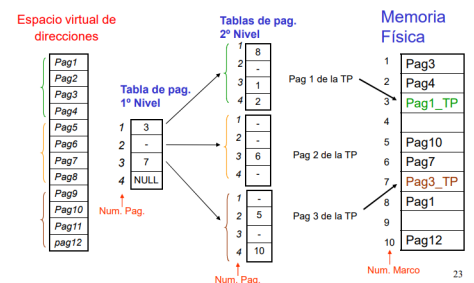
Así una dirección lógica es de la forma:

p1	p2	d
----	----	---

Esquema de paginación a dos niveles



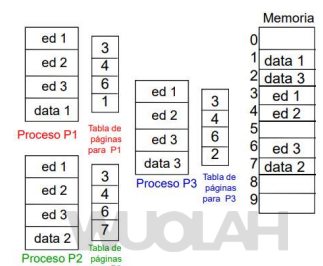
Ejemplo: Esquema de paginación a 2 niveles



PÁGINAS COMPARTIDAS

Una copia de código de solo lectura compartido entre varios procesos.

Ej. editores, compiladores, sistemas de ventanas



c. SEGMENTACIÓN

Esquema de organización de memoria que soporta mejor la visión de memoria del usuario: un programa es una colección de unidades lógicas - segmentos-
Por ejemplo: procedimientos, funciones, pila, tabla de símbolos, matrices, etc.

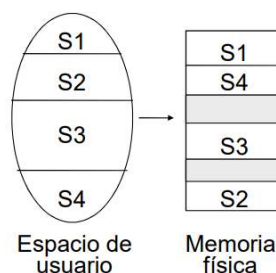


TABLA DE SEGMENTOS

Una dirección lógica es una tupla: **<número_de_segmento, desplazamiento>**

La **Tabla de Segmentos** aplica **direcciones bidimensionales** definidas por el usuario en direcciones físicas de una dimensión. Cada entrada de la tabla tiene los siguientes elementos (aparte de presencia, modificación y protección):

- » **base** - dirección física donde reside el inicio del segmento en memoria.
- » **tamaño** - longitud del segmento.

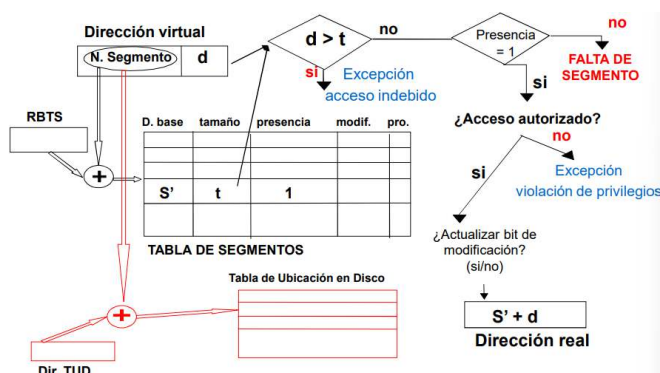
IMPLEMENTACIÓN DE LA TABLA DE SEGMENTOS

La tabla de segmentos se mantiene en memoria principal.

El **Registro Base de la Tabla de Segmentos** (RBTS) apunta a la tabla de segmentos (suele almacenarse en el PCB del proceso).

El **Registro Longitud de la Tabla de Segmentos** (STLR) indica el número de segmentos del proceso; el n° de segmento s , generado en una dirección lógica, es legal si $s < \text{STLR}$ (suele almacenarse en el PCB del proceso)

Esquema de traducción



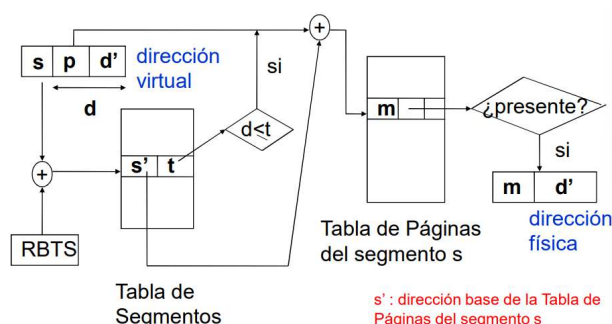
d. SEGMENTACIÓN PAGINADA

La variabilidad del tamaño de los **segmentos** y el requisito de memoria contigua dentro de un segmento **complica la gestión** de MP y MS.

Por otro lado, la **paginación** simplifica eso pero complica más los **temas de compartición y protección** (éstos van mejor en segmentación).

Algunos sistemas combinan ambos enfoques, obteniendo la mayoría de las ventajas de la segmentación y eliminando los problemas de una gestión de memoria compleja

Esquema de traducción



1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos Holandés con una garantía de hasta 100.000 euros por depositante. Consulta más información en ing.es

Que te den **10 € para gastar**
es una fantasía.
ING lo hace realidad.

Abre la **Cuenta NoCuenta** con el código
WUOLAH10, haz tu primer pago y llévate 10 €.

Quiero el cash

[Consulta condiciones aquí](#)



do your thing

3. GESTIÓN DE LA MEMORIA VIRTUAL

a. INTRODUCCIÓN

- Gestión de Memoria Virtual con paginación
- Criterios de clasificación respecto a:
 - Políticas de asignación:
 - Fija
 - Variable
 - Políticas de búsqueda:
 - Paginación por demanda
 - Paginación anticipada
 - Políticas de sustitución:
 - Sustitución local
 - Sustitución global

Independientemente de la política de sustitución utilizada, existen ciertos criterios que siempre deben cumplirse:

- Páginas “limpias” frente a “sucias”: se pretende minimizar el coste de transferencia.
- Páginas compartidas: se pretende reducir el nº de faltas de página.
- Páginas especiales: algunos marcos pueden estar bloqueados (ejemplo: buffers de E/S mientras se realiza una transferencia)

ASIGNACIÓN FIJA

- Asignación por igual
 - » Se asignan el mismo número de marcos a todos los procesos
 - » Si hay m marcos, y n procesos. A cada proceso se se asignan m/n marcos
- Asignación según tamaño
 - » Proporcional al tamaño del proceso
 - » $si = \text{tamaño de } p_i$
 - » $S = \sum si$
 - » $m = \text{número total de marcos}$
 - » la asignación, a_i , para p_i es:
$$a_i = (si / S) * m$$

PAGINACIÓN POR DEMANDA FRENTE A ANTICIPADA

Las ventajas de la paginación por demanda son:

- » Se garantiza que en MP solo están las páginas necesarias en cada momento.
- » La sobrecarga de decidir qué páginas llevar a MP es mínima.

Las ventajas de la paginación anticipada son:

- » Se puede optimizar el tiempo de respuesta para un proceso pero los algoritmos son más complejos y se consumen más recursos.

RENDIMIENTO DE LA PAGINACIÓN POR DEMANDA

Sea p la tasa de falta de página; $p=0$ no hay faltas de páginas, ó $p=1$, toda referencia es una falta. Por tanto, $0 \leq p \leq 1$

$$TAE = (1 - p) * \text{acceso_a_memoria} + p * (\text{sobrecarga_falta_de_página} + [\text{sacar_fuera_una_página}] + \text{traer_la_página} + \text{sobrecarga_de_rearranque})$$

INFLUENCIA DEL TAMAÑO DE PÁGINA

Cuanto más pequeñas:

- » Aumento del tamaño de las tablas de páginas.
- » Aumento de nº transferencias entre MP y Disco.
- » Reducen la fragmentación interna.

Cuanto más grandes:

- » Grandes cantidades de información que no serán usadas están ocupando MP.
- » Aumenta la fragmentación interna.

Búsqueda de un equilibrio.

Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 5/5 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos Holandés con una garantía de hasta 100.000 euros por depositante. Consulta más información en ing.es



b. ALGORITMOS DE SUSTITUCIÓN

Podemos tener las siguientes combinaciones:

- » Asignación fija y sustitución local.
- » Asignación variable y sustitución local.
- » Asignación variable y sustitución global.

Veremos distintos algoritmos de sustitución y nos basaremos (por simplicidad) en que se utiliza una política de asignación fija y sustitución local.

Cadena de referencia, $\omega = r_1, r_2, r_3, \dots, r_i, \dots$: secuencia de números de páginas referenciadas por un proceso durante su ejecución.

I. ALGORITMO ÓPTIMO

Se sustituye la página que no será objeto de ninguna referencia posterior o que se reference más tarde.

- » 4 marcos de página.
- » **Problema**: debemos tener un “conocimiento perfecto” de la cadena de referencia.
- » Se utiliza para medir cómo de bien se comportan otros algoritmos.
- » Faltas de páginas: 6.

1,2,3,4,1,2,5,1,2,3,4,5

1	1	1	1	1	1	1	1	1	4	4
2	2	2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3	3	3
4	4	4	5	5	5	5	5	5	5	5

* * * * *

II. ALGORITMO FIFO

Se sustituye la página por orden cronológico de llegada a MP (la página más antigua)

- » 4 marcos de página
- » Faltas de página: 10
- » Sufrir de la **Anomalía de Belady**: “más marcos no implican menos faltas de páginas”.

1,2,3,4,1,2,5,1,2,3,4,5

1	1	1	1	1	1	5	5	5	5	4	4
2	2	2	2	2	2	1	1	1	1	5	5
3	3	3	3	3	3	2	2	2	2	2	2
4	4	4	4	4	4	3	3	3	3	3	3

* * * * *

III. ALGORITMO LRU

Se sustituye la página que fue objeto de la referencia más antigua (**Least Recently Used**)

- » 4 marcos de página
- » Faltas de página: 8
- » Implementación del algoritmo:
 - Con contadores
 - Con pila
- » Mayor coste

1,2,3,4,1,2,5,1,2,3,4,5

1	1	1	1	1	1	1	1	1	1	5	5
2	2	2	2	2	2	2	2	2	2	2	2
3	3	3	3	5	5	5	5	4	4	4	4
4	4	4	4	4	4	3	3	3	3	3	3

* * * * *

IMPLEMENTACIONES DEL LRU

LRU con contador:

- » Cada entrada de la tabla de páginas tiene un contador. Cada vez que se referencia la página, se copia el tiempo del reloj en el contador.
- » Cuando necesitamos cambiar una página, se miran todos los contadores y se elige la que tiene el menor tiempo.

IV. ALGORITMO DEL RELOJ

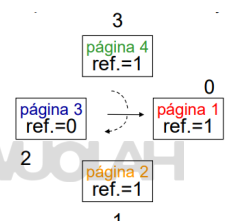
Cada página tiene asociado un bit de referencia R.

Cuando una página llega a memoria se establece a 1; en cada referencia el hardware lo pone a 1.

Los marcos de página se representan por una lista circular y un puntero a la página visitada hace más tiempo.

Selección de una página: Consultar bit R del marco actual ¿R=1?

- Si: R=0; ir al siguiente marco y volver a consultar.
- No: seleccionar para sustituir e incrementar posición.



Consulta condiciones aquí

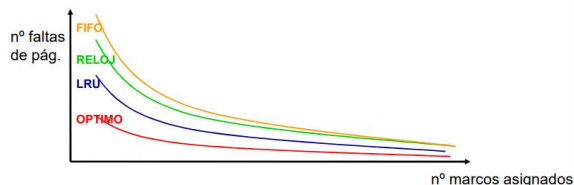


do your thing

COMPARACIÓN

Conclusión:

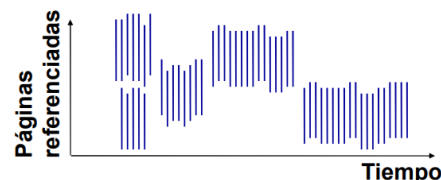
- » Infiuye más la cantidad de MP disponible que el algoritmo de sustitución usado.



c. COMPORTAMIENTO DE LOS PROGRAMAS

Viene definido por la secuencia de referencias a página que realiza el proceso.

Importante para maximizar el rendimiento del sistema de memoria virtual (TLB, alg. sustitución, ...).



PROPIEDAD DE LOCALIDAD

Distintos tipos

- » **Temporal**: Una posición de memoria referenciada recientemente tiene una probabilidad alta de ser referenciada en un futuro próximo (ciclos, rutinas, variables globales, ...).
- » **Espacial**: Si cierta posición de memoria ha sido referenciada es altamente probable que las adyacentes también lo sean (array, ejecución secuencial, ...).



CONJUNTO DE TRABAJO

Observaciones:

- » Mientras el conjunto de páginas necesarias puedan residir en MP, el nº de faltas de página no crece mucho.
- » Si eliminamos de MP páginas de ese conjunto, la activación de paginación crece mucho.

Conjunto de trabajo (*Working Set*) de un proceso es el conjunto de páginas que son referenciadas frecuentemente en un determinado intervalo de tiempo.

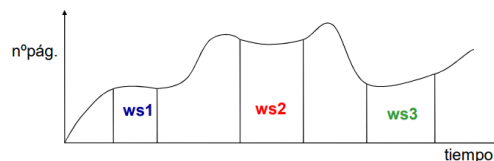
Se define el conjunto de trabajo en el momento t con "ancho de ventana" Δ como:

$$WS(t, \Delta) = \text{páginas referenciadas en el intervalo de tiempo } (t - \Delta, t]$$

CONJUNTO DE TRABAJO: PROPIEDADES

Propiedades

- » Los conjuntos de trabajo son transitorios
- » No se puede predecir el tamaño futuro de un conjunto de trabajo.
- » Difieren unos de otros sustancialmente.



TEORÍA DEL CONJUNTO DE TRABAJO

Un proceso sólo puede ejecutarse si su conjunto de trabajo está en memoria principal. Una página no puede retirarse de la memoria principal si está dentro del conjunto de trabajo del proceso en ejecución.

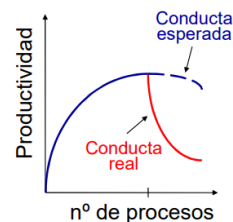
d. HIPERPAGINACIÓN (THRASHING)

Al aumentar el nº de procesos existe un valor a partir del cual se produce una caída brusca de las prestaciones:

- se observa que ningún proceso adquiere tiempo de CPU.
- hay un fuerte aumento del nº de intercambio de págs.

Formas de evitar la hiperpaginación:

- » Asegurar que cada proceso existente tenga asignado un espacio en relación a su comportamiento → **Algoritmos de asignación variables**
- » Actuar directamente sobre el grado de multiprogramación → **Algoritmos de regulación de carga.**



e. ALGORITMOS DE ASIGNACIÓN Y SUSTITUCIÓN

I. ALGORITMO BASADO EN EL MODELO DEL WS

Parámetro del algoritmo: ancho de ventana V .

En cada momento t en que se hace referencia a una posición de memoria, se determina el conjunto de trabajo de la siguiente forma:

$WS = \text{páginas referenciadas en el intervalo } (t - V, t]$

solo estas páginas se mantienen en memoria principal

II. ALGORITMO FFP

Definición: Intervalo entre dos faltas de página = Tiempo actual – tiempo de la falta de página anterior

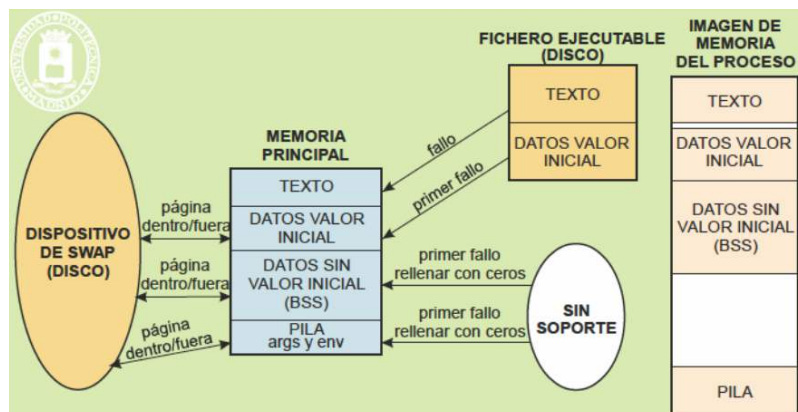
IF intervalo > valor umbral L // L es un parámetro del algoritmo

THEN //el proceso tiene demasiados marcos de página, hay que quitarle. Se retiran de memoria principal todas las páginas no referenciadas entre el momento actual y el momento de la falta de página anterior

ELSE // demasiadas faltas de página, el proceso tiene pocos marcos. se asigna al proceso un marco de página más para albergar la página que ha originado esta falta de página.

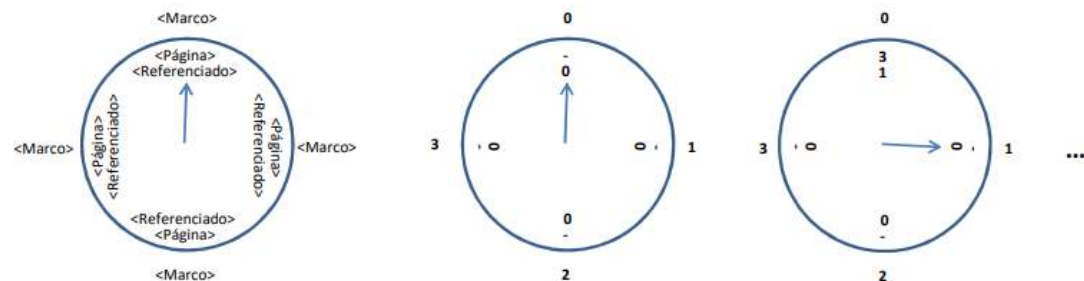
LA VELOCIDAD DE FP SIRVE PARA CONOCER SI EL N° DE MARCOS ASIGNADOS AL PROCESO ES GRANDE O PEQUEÑO

CICLO DE VIDA DE LAS PÁGINAS DE UN PROCESO



Evolución del soporte físico de las páginas de la imagen de memoria de un proceso.

EJEMPLO SIMPLE: Con representación alternativa del algoritmo del reloj



<Marco> = Número de Marco

<Página> = Número de Página almacenada en el marco

<Referenciado> = Bit referenciado puede tomar valor 0 ó 1