

Tema 5: Organización Apilo

- 5.1 Introducción
 - 5.2 Estructura de la organización apilada
 - Apilo estructurado
 - Apilo no estructurado
 - 5.3 Ocupación del fichero apilo
 - 5.4 Operaciones con los ficheros apilo
 - 5.5 Resumen
-

Contenidos extraídos del libro:

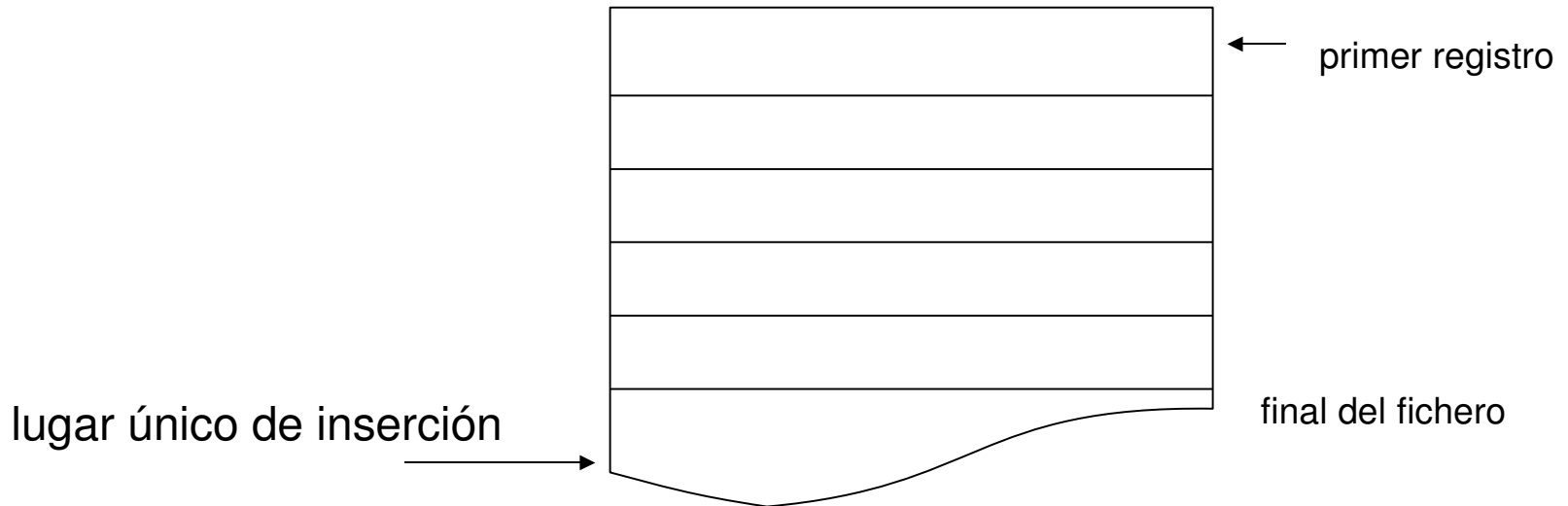
Ficheros. Organizaciones clásicas para el almacenamiento de la información

Autores: *Irene Luque Ruiz. Juan Antonio Romero del Castillo y Miguel Ángel Gómez-Nieto*
Editorial: Servicio de Publicaciones de la Universidad de Córdoba, 1998.
ISBN 84-7801-468-3

5.1 Introducción

El fichero apilo:

- Estructura de pila: FILO
- Inserción rápida de registros al final del fichero
- Se usa como recurso en el que “apilar” datos de forma rápida sin más



5.1 Introducción

Ejemplo: Laboratorio de análisis clínicos. La captura automática de datos

Un laboratorio de análisis clínicos dispone de un autoanalizador que realiza hasta 100 analíticas diferentes sobre muestras de distintos pacientes dejadas en una bandeja porta-muestras. Los datos recogidos son de diferente tipo y tamaño según la muestra realizada... Cuando el autoanalizador llena el buffer interno descarga los datos a través de una conexión serie a un ordenador... Cada registro del fichero almacena los datos de cada análisis realizado con un número de orden que determina la posición de la muestra en el porta-muestras, el parámetro analítico y el resultado...

Registro: NúmeroOrdenMuestra, NombreAnalítica, valorAnalítica, NombreAnalítica, valorAnalítica ...
Registro: NúmeroOrdenMuestra, NombreAnalítica, valorAnalítica, NombreAnalítica, valorAnalítica ...

5.1 Introducción

Ejemplo: Laboratorio de análisis clínicos. La captura automática de datos



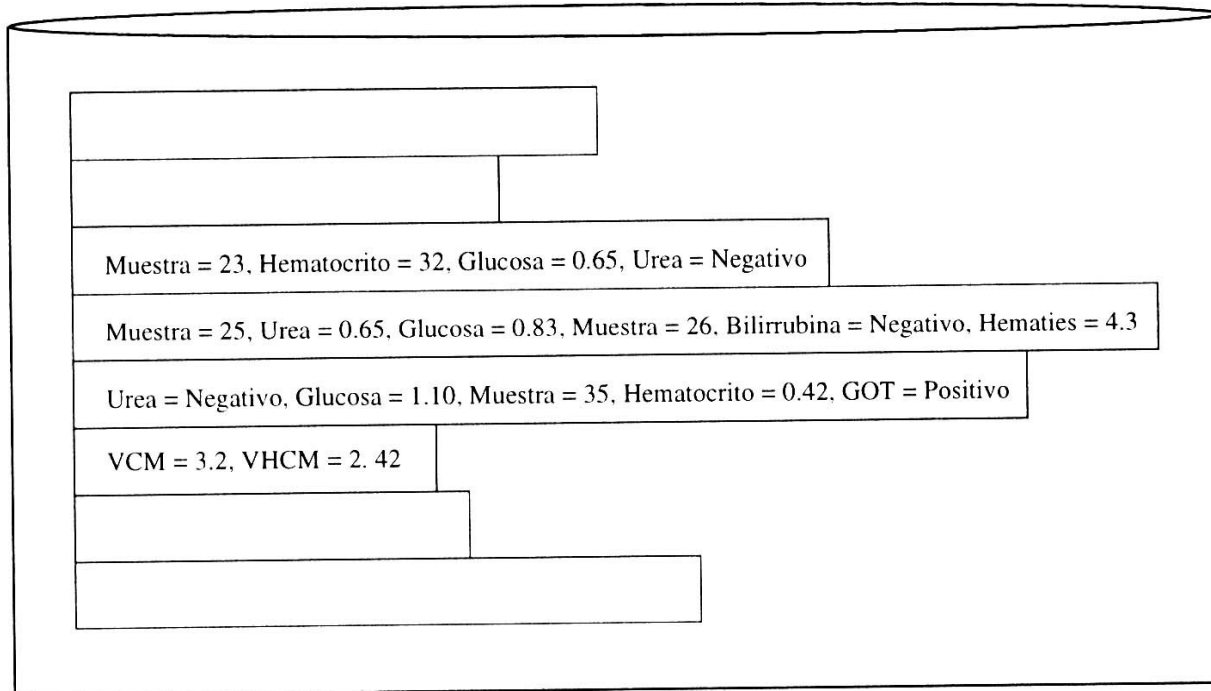
Registro: NúmeroOrdenMuestra, NombreAnalítica, valorAnalítica, NombreAnalítica, valorAnalítica ...

Registro: NúmeroOrdenMuestra, NombreAnalítica, valorAnalítica, NombreAnalítica, valorAnalítica ...

5.2 Estructura del fichero apilo

El fichero apilo no estructurado:

- Registros de longitud variable
- Datos identificados y con separadores
- No existe ningún tipo de orden
- Procesamiento serial de los datos



5.2 Estructura del fichero apilo

El fichero apilo estructurado:

- Registros de longitud fija o variable
- Registro estructurado: sin identificadores ni separadores
- No existe ningún tipo de orden
- Procesamiento serial de los datos

Número-muestra, Parámetro-analítico, Resultado		
4	Urea	Negativo
99	Bilirrubina	Negativo
7	Glucosa	1.1

(a)

4	Urea	Negativo
99	Bilirrubina	Negativo
7	Glucosa	1.1
12	Hematocrito	36
78	Hemáticos	4.23

(b)

5.3 Ocupación

Apilo no estructurado

El tamaño del fichero vendría dado por la siguiente expresión:

$$\Gamma = \sum_{i=1}^r \sum_{j=1}^k (L_{I_{ij}} + L_{V_{ij}} + L_{S_j})$$

$L_{I_{ij}}$: tamaño del **identificador** del atributo j del registro i

$L_{V_{ij}}$: tamaño del **valor** del atributo j del registro i

L_{S_j} : suma de los tamaños de todos los **separadores**

r : número de registros

k : número de atributos

Podríamos estimar también la ocupación esperada

$$\longrightarrow \hat{\Gamma} = \bar{R} r_e$$

r_e : número esperado de registros

5.3 Ocupación

Apilo estructurado

longitud variable, tamaño medio de registro (no hay identificadores)

$$\bar{R} = \sum_{j=1}^k (\bar{L}_{V_j} + L_S)$$

longitud variable, ocupación del fichero

$$\Gamma = \sum_{i=1}^r \sum_{j=1}^k (L_{V_{ij}} + L_{S_j})$$

longitud fija, tamaño de registro

$$R = \sum_{j=1}^k L_{V_j}$$

longitud fija, ocupación del fichero

$$\Gamma = R r$$

5.4 Operaciones con ficheros apilo

Inserción: T_I

- Desplazamiento del brazo del disco hasta el cilindro t_l
- Rotación del dispositivo hasta el bloque buscado t_r
- Transferencia del bloque donde se insertará el nuevo registro a memoria T_{tB}
- Procesamiento del bloque para empaquetar el nuevo registro T_P
- Reescritura del bloque modificado T_{RE}

$$T_I = t_l + t_r + T_{tB} + T_{RE} + T_P$$

T_{tB} : tiempo de transferencia de bloque a memoria

T_{RE} : tiempo de reescritura de un bloque

T_P : tiempo de procesamiento en memoria para el empaquetamiento del nuevo registro

5.4 Operaciones con ficheros apilo

Inserción: T_I

- Considerando que $T_{RE} = 2t_r$
- Podemos reducir:

$$T_I = t_l + 3t_r + T_{tB}$$

Este será el menor tiempo de inserción de todas las organizaciones de ficheros que estudiemos

5.4 Operaciones con ficheros apilo

Lectura: T_L

- Siempre en base a un criterio de búsqueda
- Puede suponer la lectura de todo el fichero en busca del registro deseado
- Por término medio se leerán la mitad de los bloques buscando un registro concreto

$$\Omega_b \approx \frac{1}{2} b$$

suponemos los bloques en el mismo cilindro

$$\bar{T}_L = t_l + \Omega_b \times (t_r + T_{tB}) + T_P$$

Ω_b : número promedio de bloques leídos en la operación de lectura

5.4 Operaciones con ficheros apilo

Lectura: T_L

Si se desprecia: tiempo de localización, latencia rotacional y tiempo de procesamiento:

$$\bar{T}_L = \frac{1}{2} b T_{tB} = \frac{1}{2} r T_{tR}$$

T_{tR} : tiempo de transferencia de registro

!! La mitad de los bloques (o registros) !!

5.4 Operaciones con ficheros apilo

Lectura Consecutiva: T_{LC}

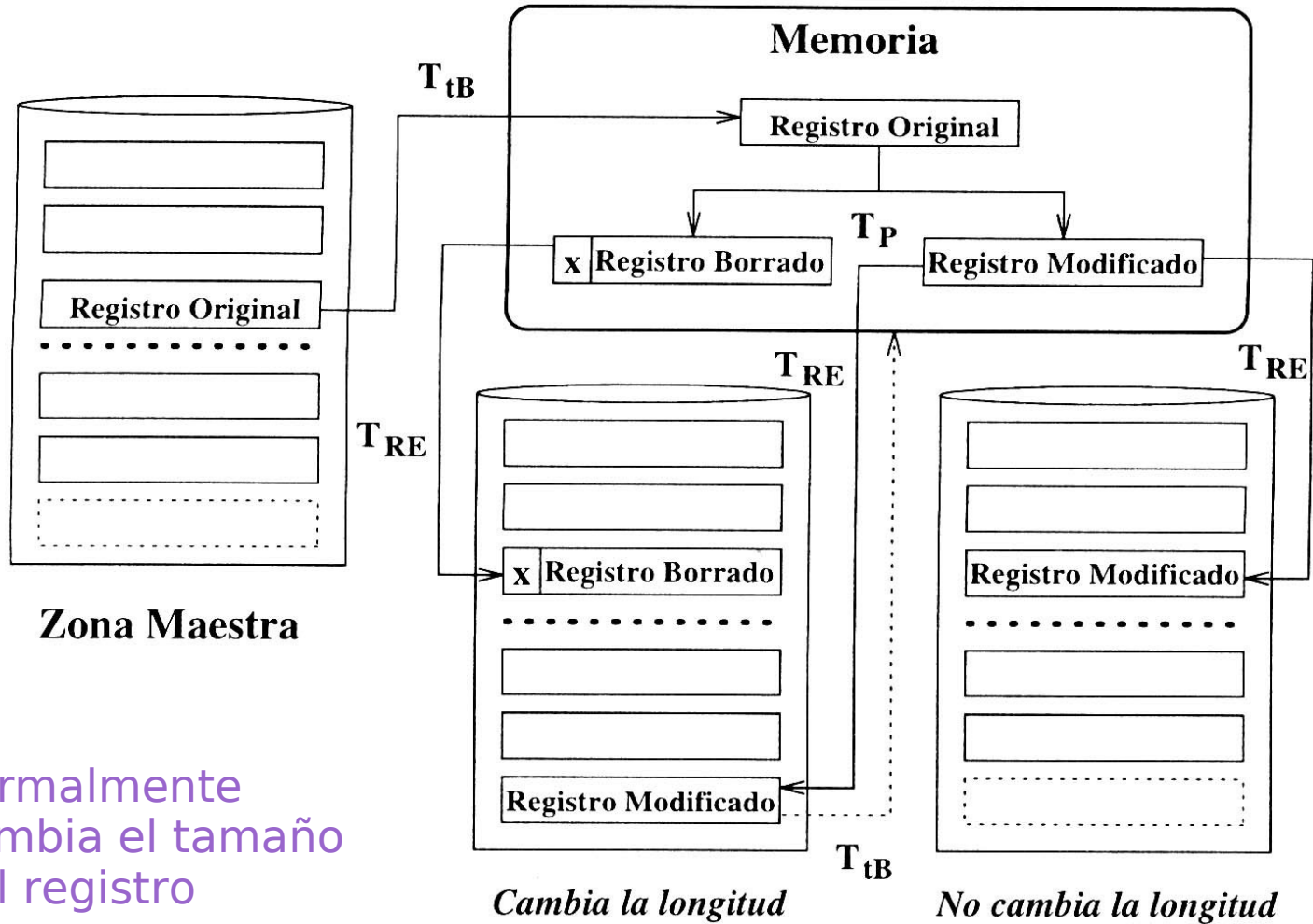
En esta caso, el acceso a un registro no facilita el acceso al siguiente: ¡ no están ordenados !

$$T_{LC}^{-} = T_L = \frac{1}{2} b T_{tB} = \frac{1}{2} r T_{tR}$$

¡¡ La mitad de los bloques (o registros) !!

5.4 Operaciones con ficheros apilo

Actualización: T_A



5.4 Operaciones con ficheros apilo

Actualización: T_A

- 1.- Recuperar el registro
- 2.- Reescribirlo marcado para borrado
- 3.- Insertar el registro modificado

$$T_A = \bar{T}_L + T_{RE} + T_I$$

5.4 Operaciones con ficheros apilo

Lectura Exhaustiva (total): T_{LT}

Supone leer sistemáticamente todos los datos sin importar el orden

$$T_{LT} = 2 \times \bar{T}_L = bT_{tB} = rT_{tR}$$

5.4 Operaciones con ficheros apilo

Lectura Ordenada: T_{LO}

- Leer todos los datos en orden
- Basamos esta operación en la Lectura Consecutiva

$$T_{LO} = \bar{T}_L + (r - 1) \times \bar{T}_{LC} = r \bar{T}_L$$

¡ coste de tiempo enorme !



$$\bar{T}_L = \frac{1}{2} b T_{tB} = \frac{1}{2} r T_{tR}$$

5.4 Operaciones con ficheros apilo

Lectura Ordenada: T_{LO}

Podríamos ordenar primero + leer ordenado después

$$T_{LO} = T_C(r) + T_{LT}(r')$$

T_C : tiempo de ordenación del fichero

r' : número de registros que satisfacen el criterio de ordenación

5.4 Operaciones con ficheros apilo

Reorganización: T_{RO}

- Las operaciones de actualización y borrado generan registros marcados para borrado
- Volatilidad de los datos: cuando los datos almacenados dejan de ser útiles debido a una operación (borrado, actualización, etc.)
- El proceso incluye:
 1. Lectura de todos los registros
 2. Análisis de cada registro para desestimar los no borrados
 3. Escritura de los registros en un nuevo fichero
 4. Borrar fichero original y renombrar el nuevo

$$T_{RO} = r \times T_{tR} + (r - r_B) \times T_{tR}$$

5.5 Resumen

- Organización primitiva y con pocas ventajas en el acceso a los datos.
- Es la de inserción más rápida ya que solo se añade el registro al final.
- Utilizados en sistemas automáticos de adquisición de datos, donde lo importante es recoger los datos más que procesarlos en algún modo.
- Altos costes de mantenimiento.
- Se usa como fichero de paso de los datos hacia otras organizaciones donde el control es más fácil.

Fin