

Tema 4. Nivel Interno

Nivel interno → Operaciones sobre unidades mínimas (Págs o bloques de BD)
 ↳ SGBD

Nivel físico → Capa de abstracción sobre hardware
 ↳ SO

Cada registro → RID Record Identifier → Página
 ↳ Offset
 ↳ Cabecera: nº y tipo columnas
 ↳ Datos contenidos

SGBD → Páginas o bloques de BD \Rightarrow Bloques de SO $\xrightarrow{\text{agregados}}$ Archivos de SO

Organización y métodos de acceso

- Organización secuencial: registros almacenados consecutivos \Rightarrow Ayuda estructuras ordenadas por clave física

Indexación → Ficheros indexados cada fichero sea un índice → RID
 ↳ Clave de búsqueda
 ↳ fíjero + pequeño que el de datos

→ Índice primario: clave de búsqueda = física
 → Índice secundario: clave de búsqueda \neq física (Puede clave compuesta)

→ Índices no densos + para mantenerlos en HD → Clave de búsqueda
 ↳ Direc comienzo bloque del registro
 ↳ 1 registro/bloque
 ↳ SOLO sobre clave física

→ Índices jerárquicos índices sobre índices → Primer nivel: sobre datos, denso o no
 ↳ Otros: no densos

→ Árboles B+ valores clave en nodos hoja → Orden M: máx nº de hijos/nodo
 ↳ Balanceado

→ B+ en BD: orden elevado,
 ↳ cada nodo capacidad de un bloque de datos

→ Tablas Organizadas por Índices (COOT) no guarda RID sino tupla

Nodo	Restricción	Min	Max	Ejemplo M=5
Raíz (cuando no es nodo único)	Número de hijos	2	M	2-5
Interno	Número de hijos	$\lceil M/2 \rceil$	M	3-5
Hoja	Número de claves	$\lfloor M/2 \rfloor$	M-1	2-4

→ Árboles B: variante del anterior, algunas claves en nodos intermedios

- 2396 → 6892
- Índices por clave invertida para reducir embastillamientos por bloques
 - ↳ Adecuado consultas por valor
 - ↳ Id para búsquedas por intervalos

 - Índices BITMAP para cada valor de clave \Rightarrow Sec de n bits
 - Bit = 1 \Rightarrow Valor presente en la tupla
 - Bit = 0 \Rightarrow No lo está
 - ↳ Para columnas que tienen pocos valores y consultas con OR
 - ↳ Si muchos valores \Rightarrow B-tree
- $n = n$ de tuplas
-

Métodos de acceso: Acceso directo

- Sin acceso adicional A pelo
- Alg. que nos indique posición del registro

Acceso directo: calcular dirección del registro con alg. o función sobre uno de sus campos

- Funcionamiento: valor clave \rightarrow Alg. de direcc. entre 0 y N → Trans. entreco/RSD \rightarrow Accm. físico
- Mecanismos
 - ↳ Cuadrados centrales
 - ↳ Congruencias
 - ↳ Desplazamiento
 - ↳ Conversión de base
- Problema: huecos y colisiones \Rightarrow Sol. Zona de desbordamiento y listas de colisión / de sinónimos

Si aparecen más colisiones...

Hashing:

- Básico: divide ficheros en cubos. Alg. direcciona cubos + de 1 registro/cubo
 - I zonas + cubos que otras y cubos de desbordamiento

- Dinámico: se parte de config. uniforme de pocos cubos y se van añadiendo cubos
 - Técnica: clave transformada \rightarrow Índice en memoria \rightarrow Direc. cubo
 - Conforme se añaden registros \Rightarrow Nuevos cubos, cuentas índice

 - Algoritmo: $v = \text{valor clave} \Rightarrow hcv = K$ clave de direcc. $\Rightarrow b < d \leq n$
 - $\text{pref. local} \Rightarrow 2^b$ cubos
 - $\text{pref. global} \Rightarrow$ Como max. 2^d cubos
 - n de bits de K en binario
 - Cada cubo pref. local $p \Rightarrow$ Si. primeros p bits de K coinciden con los del cubo, el registro de v va a ese cubo
 - S. un cubo se llena \Rightarrow Se divide en 2 con probabilidad $p+1$ \rightarrow Otro con bit = 0