

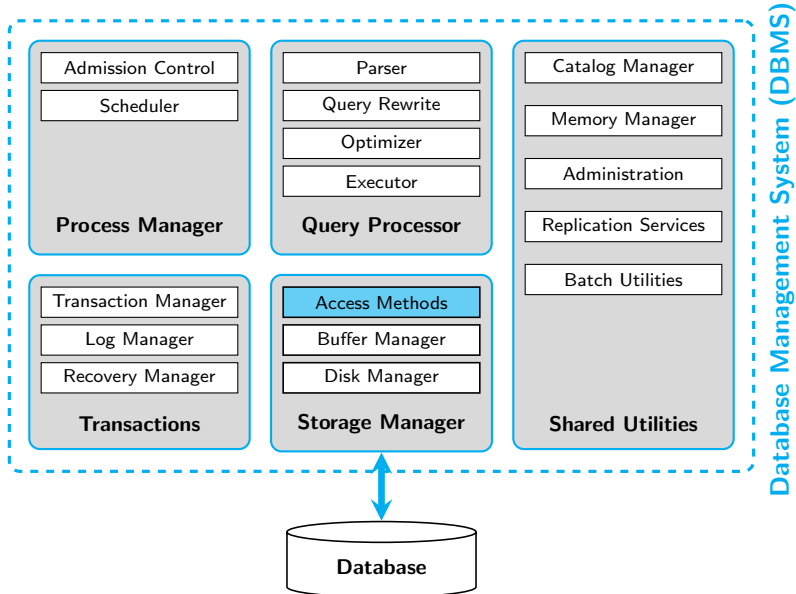
Introducción a índices

Clase 04

IIC 3413

Prof. Cristian Riveros

Métodos de acceso



Métodos de acceso

Procesador de consultas

considera las relaciones como “colección de records”.

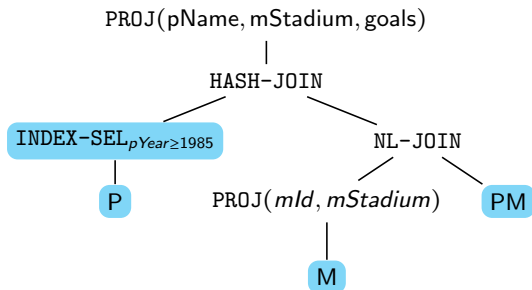
Administrador de almacenamiento provee al procesador de consultas con métodos para acceder estas colecciones de records.

Estos métodos son conocidos como **métodos de acceso**.

Evaluación de consulta y métodos de acceso

```
Q = SELECT  pName, mStadium, goals
      FROM    Players AS P, Matches AS M, Players_Matches AS PM
      WHERE   P.pId = PM.pId AND PM.mId = M.mId AND
              P.pYear ≥ 1985
```

Plan físico para evaluar Q:



Interfaz de acceso a relaciones

1. **Create** o **Destroy**

- Crear o destruir el acceso.

2. **Insert**(record)

- Insertar un nuevo record a la relación.

3. **Delete**(RID)

- Eliminar un record dado su Record ID (RID).

4. **Get**(searchkey)

- Buscar un record dado una “llave de busqueda”.
- El criterio de busqueda puede ser diverso (por valor, por rango, etc).
- En general, **Get** puede estar “sobrecargado”.

5. **Scan**

- Iterar sobre todos los records.

Ejemplo: Heapfile

1. **Create** o **Destroy**: crea un heap-file de la relación.
2. **Insert**(record): inserta un record al final del archivo.
3. **Delete**(RID): busca la página del record y la elimina.
4. **Get**(searchkey)
 - Busca en todas las páginas los records que satisface el searchkey.
5. **Scan**
 - Iterar sobre todas las páginas y sobre todos los records.

Ejemplo: Sortedfile

1. **Create** o **Destroy**: crea un sortedfile de la relación.
2. **Insert**(record): inserta un record en la **posición** que le corresponde.
3. **Delete**(RID): busca la página del record y la elimina.
4. **Get**(searchkey)
 - **Busqueda binaria** del record que satisface el searchkey.
5. **Scan**.
 - Iterar sobre todos las páginas y sobre todos los records.

¿qué es un índice?

Definición

Método de acceso que **optimiza** el acceso a los datos para **una consulta o conjunto de consultas** en particular.

Ejemplos

- Índice de un libro.
- Orden alfabético en un diccionario.
- Número de páginas de un libro.
- Secciones de un diario.

Algunos hechos importantes sobre un índice

1. Un índice optimiza un subconjunto de consultas.
2. Un índice optimiza ciertas consultas pero puede hacer otras más costosas.
3. Es posible sacrificar espacio por tiempo, pero **idealmente** un índice debe ser mantenido en RAM.

¿qué consultas nos gustaría optimizar con índices?

Busqueda por **valor** (*value query*).

```
SELECT  *  
FROM    table  
WHERE   attribute = 'value'
```

Busqueda por **rango** (*range query*).

```
SELECT  *  
FROM    table  
WHERE   attribute ≥ 'value'
```

Busqueda por **match** (*pattern matching*).

```
SELECT  *  
FROM    table  
WHERE   attribute LIKE 'patrón'
```

¿qué otra consulta nos gustaría optimizar?

Evaluación de la eficiencia de índices

Evaluación con respecto a:

- Tipo de acceso.
- Tiempo de acceso.
- Tiempo de inserción.
- Tiempo de eliminación.
- Sobrecarga de espacio.

¿qué parametro es el más importante?

Componentes de un índice

Definición

Search key = parámetros de búsqueda.

Index entry = valor o puntero
de la estructura interna de un índice

Data entry = record mismo, o
dirección donde se almacena un record.

Para un search key k , un **data entry** puede ser:

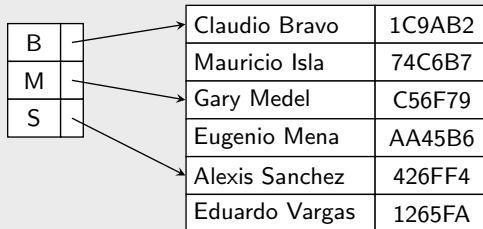
1. Un record. (que satisface el search key k)
2. (k, RID) .
3. $(k, \text{lista de } RID)$.

Componentes de un índice

Ejemplos

Índice sobre la relación *Players*(*ID*, *name*, *number*):

B		Claudio Bravo	1C9AB2
M		Mauricio Isla	74C6B7
S		Gary Medel	C56F79
		Eugenio Mena	AA45B6
		Alexis Sanchez	426FF4
		Eduardo Vargas	1265FA



The diagram illustrates a B-tree index structure. On the left, a B-tree node is shown with three slots containing the letters 'B', 'M', and 'S'. Arrows point from the right side of this node to specific rows in a data table on the right. The arrow from 'B' points to the row for Claudio Bravo. The arrow from 'M' points to the row for Mauricio Isla. The arrow from 'S' points to the row for Gary Medel. The data table contains six rows, each with a player's name and a hexadecimal ID.

¿cuáles son los **search keys**, **index** y **data entries** de este índice?

Componentes de un índice

Ejemplos

INDEX	
ALCOHOL CAN BE A GAS!	
U.S. corn, 27, 27f, 31–32, 31f, 39–40	forms of, 437
<i>Agrocybe aegerita</i> (mushroom), 314f	generator using, 444
Agrol, 18	household power use of, 446–448
Agrol Company, 17, 18	industrial-grade, 206
air conditioners	leakage of, 268
cogenerators as, 445	lighting with, 447
heat pumps compared to, 218, 219	liquid, 210
household cogenerated, 445	off-road uses for, 196–197, 339–341, 444, 462
ice block, 447	oxygen content of, 347
air pollution, 34–35, 56	phase separation of, 225–226
catalytic converters and, 379	prairie v. corn, 42
coal and, 57–58	proof requirement and, 196–197
exhaust, 425	reforming, 431
neat ethanol reducing, 350	sources for, 119–180
small engine, 421	storage of, 232, 268–274, 268f, 271f
stoichiometric ratios and, 379–380	sugar, 136
two-stroke engines and, 425	vaporized, 66, 331f, 332–333, 418
wood smoke, 224, 339	

¿cuáles son los **search keys**, **index** y **data entries** de este índice?

Componentes de un índice

Ejemplos

Relación *Players*(*ID*, *name*, *number*), ordenado por ID:

232	Claudio Bravo	1
335	Gary Medel	17
481	Eugenio Mena	2
520	Mauricio Isla	4
555	Eduardo Vargas	11
630	Alexis Sanchez	7

¿cuáles son los **search keys**, **index** y **data entries** de este índice?

Clustered indexes

Definición

Clustered index = índice para el cual el orden de sus data entries es el mismo orden de los records en disco.

Unclustered index = índice que NO es clustered.

Clustered indexes

Ejemplos

Índice sobre la relación *Players*(*ID*, *name*, *number*):

B		Claudio Bravo	1C9AB2
M		Mauricio Isla	74C6B7
S		Gary Medel	C56F79
		Eugenio Mena	AA45B6
		Alexis Sanchez	426FF4
		Eduardo Vargas	1265FA

Suponiendo que la relación esta almacenada por orden de ID,
¿es este índice **clustered** o **unclustered**?

Clustered indexes

Ejemplos

Índice sobre la relación *Players*(*ID*, *name*, *number*):

B		232	Claudio Bravo	1
M		520	Mauricio Isla	4
S		335	Gary Medel	17
		481	Eugenio Mena	2
		630	Alexis Sanchez	7
		555	Eduardo Vargas	11

¿es este índice **clustered** o **unclustered**?

¿qué tipo de **data entry** tiene este índice?

Clustered indexes

- En general, por cada relación es posible mantener un solo clustered index. (¿por que?)
- Unclustered index tiene data entries del tipo 2 o 3.
- Unclustered index son ineficientes cuando el output es numeroso.
 - range queries.
- Usualmente, clustered y unclustered indexes son conocidos como:
 - clustered index = índice **primario**.
 - unclustered index = índice **secundario**.

Índices densos o dispersos

Definición

Los índices pueden ser:

Denso (dense) = un index entry por cada record de la relación.

Disperso (sparse) = no todos los records están mencionadas en los index entries.

Índices densos o dispersos

Ejemplos

Índice sobre la relación *Players*(*ID*, *name*, *number*):

B		Claudio Bravo	1C9AB2
M		Mauricio Isla	74C6B7
S		Gary Medel	C56F79
		Eugenio Mena	AA45B6
		Alexis Sanchez	426FF4
		Eduardo Vargas	1265FA

¿es este índice **denso** o **disperso**?

Índices densos o dispersos

Ejemplos

Índice sobre la relación *Players*(ID, name, number):

Bravo		→	232	Claudio Bravo	1
Isla		→	520	Mauricio Isla	4
Medel		→	335	Gary Medel	17
Mena		→	481	Eugenio Mena	2
Sanchez		→	630	Alexis Sanchez	7
Vargas		→	555	Eduardo Vargas	11

¿que ventaja puede tener este índice?

Resumen de la clasificación de índices

Clustered vs unclustered

- Clustered: el orden de sus data entries es el mismo que los records.
- Unclustered: data entries no mantienen el mismo orden de los datos.

Denso vs disperso

- Denso: un index entry por cada record.
- Disperso: no todos los records están mencionadas en los index entries.

Dos tipos de índices básicos

1. Índices basados en **árboles**:

uso del **orden** de los valores para organizar los records.

- ISAM.
- B+ trees.

2. Índices basados en **hashing**:

uso de una distribución uniforme de los valores sobre distintos grupos.

- Extendable Hashing.