

Primer Exercici d'Access

Enunciat

Donades la taula i les dades al fitxer adjunt, feu el disseny físic de manera que sigui òptima l'execució de la consulta següent:

```
SELECT sum(pressupost) from obres where id = 500;
```

```
create table obres(  
  id number(8,0),  
  zona char(20),  
  tipus number(17,0),  
  pressupost number(17, 0),  
  nom char(100),  
  empreses char(250),  
  descripcio char(250),  
  responsables char(250)  
);
```

Pas previ a l'estudi d'alternatives

Tenint en compte la consulta que es vol optimitzar, i l'esquema de la base de dades sobre la que s'executa.

- Descartar alternatives que ja es veu que no seran les millors, o que no són adequades.
- En aquest exercici:
 - Es descarta *Taula sense cap índex ni hash* perquè tenint en compte que només es busca una única fila de la taula (id és PK), segur que alguna estructura d'accés pot afavorir la consulta.
 - Es descarta *Clustered structure* perquè la consulta es fa sobre una única taula.

Alternativa 1: Table with clustered index, Taula amb un índex cluster

1. Executar script per esborrar objectes
 - El script executat inclou l'esborrat de la paperera
2. Crear la taula que hi ha al fitxer adjunt, afegint la Primary Key (PK), i definint que es crei un índex cluster ple a 2/3. L'índex és per la PK

```
CREATE TABLE obres (... , PRIMARY KEY(id))  
ORGANIZATION INDEX PCTFREE 33;
```

Inserir les files a la taula, usant el script del fitxer adjunt

Reconstruir el índex després dels inserts

```
ALTER TABLE obres MOVE;
```

3. Actualitzar les estadístiques en el catàleg de Oracle usant el script corresponent
4. Comprovar el pla d'execució de la sentència que es vol optimitzar
5. Comprovar les estructures que s'han creat, i quan espai ocupen
6. Guardar informació per comparar amb la resta d'alternatives

Alternativa 2: B+ Tree, Arbre B+

1. Executar script per esborrar objectes
 - El script executat inclou l'esborrat de la paperera
2. Crear la taula que hi ha al fitxer adjunt, sense espai lliure en les blocs que emmagatzemen les files de la taula (PCTFREE 0)

```
CREATE TABLE obres (...) PCTFREE 0 ENABLE ROW MOVEMENT;
```

Inserir les files a la taula, usant el script del fitxer adjunt

```
ALTER TABLE tableName SHRINK SPACE;
```

Crear el índex arbre B+ ple a 2/3

```
CREATE UNIQUE INDEX indexId ON obres (id)  
PCTFREE 33;
```

4. Actualitzar les estadístiques en el catàleg de Oracle usant el script corresponent
5. Comprovar el pla d'execució de la sentència que es vol optimitzar
6. Comprovar les estructures que s'han creat, i quan espai ocupen
7. Guardar informació per comparar amb la resta d'alternatives

Alternativa 3: Table with hash, Taula amb hash

1. Determinar la grandària de la taula sense cap estructura (B)
 - Crear la taula sense cap estructura amb “PCTFREE 0 ENABLE ROW MOVEMENT” i fer “SHRINK SPACE”,
 - Actualitzar les estadístiques i consultar el catàleg per a veure el nombre de blocs que té

```
SELECT table_name, blocks FROM USER_TABLES;
```
2. Executar script per esborrar objectes
 - El script executat inclou l'esborrat de la paperera
3. Crear hash indicant el atribut pel que s'ha de crear ($\lceil 1.25 * B \rceil = 140$)


```
CREATE CLUSTER hashObresId (id number (8,0)) SINGLE TABLE
HASHKEYS 140 PCTFREE 0;
```

Crear la taula:

```
CREATE TABLE obres (... )
CLUSTER hashObresId(id);
```

Inserir les files a la taula, usant el script del fitxer adjunt
4. Actualitzar les estadístiques en el catàleg de Oracle usant el script corresponent
5. Comprovar el pla d'execució de la sentència que es vol optimitzar
6. Comprovar les estructures que s'han creat, i quan espai ocupen
7. Guardar informació per comparar amb la resta d'alternatives

Solució òptima: Taula amb hash

En mirar el pla d'execució en cadascuna de les alternatives es pot veure que la solució amb hash és la que dona un cost més baix

```
SELECT sum(pressupost) from obres where id = 500;
```

En l'exercici no s'estableix un límit en l'espai ocupat per les estructures de dades i per tant no afecta a la decisió

- En cas que dues solucions accedissin al mateix número de blocs (no passa en aquest exercici), s'escolliria la que ocupa menys espai
- En cas que es limités l'espai que es pot ocupar (no passa en aquest exercici), s'escolliria una solució que no superi el límit i que sigui la que millor optimitza la consulta