IIC2413 - Bases de Datos

Guía Índices

Problema

Sea la relación $R(a, b, c, d)$ cuyo tamaño es de 1 millón de tuplas, en que cada página contiene P tuplas. Las tuplas de R están ordenados de manera aleatoria. El atributo a es además un candidato a llave primaria, cuyos valores van del 0 al 999.999 (distribuídos uniformemente). Para cada una de las consultas a continuación, diga el número de I/O que se harán en cada uno de los siguientes casos:

- Analizar R sin ningún índice.
- Usar un *B+Tree unclustered* sobre el atributo a. El árbol es de altura h y cada página contiene M punteros ($M > P$).
- Usar un *B+Tree clustered* sobre el atributo a. El árbol es de altura h y cada página de hoja está ocupada al 60 %.
- Usar un *Hash Index unclustered* con 1 millón de buckets. Cada página del indice contiene M punteros ($M > P$).
- Usar un *Hash Index clustered* con 1 millón de buckets.

Las consultas son:

1. Encontrar todas las tuplas de R.
2. Encontrar todas las tuplas de R tal que $a < 50$.
3. Encontrar todas las tuplas de R tal que $a = 50$.
4. Encontrar todas las tuplas de R tal que $a > 50$ y $a < 100$.

Solución

Los costos son los siguientes:

Query	Sin índice	B+Tree - u	B+Tree - c	Hash Índex - u	Hash Índex - c
R	$\frac{10^6}{P}$	$(h-1) + (\frac{10^6}{M}) + 10^6$	$(h-1) + \frac{10^6}{0{,}6P}$	$2 \cdot 10^6$	10^6
$a < 50$	$\frac{10^6}{P}$	$(h-1) + (\frac{50}{M}) + 50$	$(h-1) + (\frac{50}{0{,}6P})$	100	50
$a = 50$	$\frac{10^6}{P}$	$h + 1$	h	2	1
$50 < a < 100$	$\frac{10^6}{P}$	$(h-1) + (\frac{49}{M}) + 49$	$(h-1) + (\frac{49}{0{,}6P})$	98	49

Sin índice

Para el caso cuando no se tiene ningún índice, para todas las consultas debemos acceder a todas las páginas de disco donde se almacenan las tuplas. Si tenemos 10^6 tuplas, y capacidad P en cada página, necesitamos $\frac{10^6}{P}$ páginas para almacenar toda la información. Por lo tanto, para todas las consultas tendremos un costo I/O de $\frac{10^6}{P}$.

B+Tree Clustered

1. En este caso, para encontrar todas las tuplas, primero tenemos que bajar por el árbol hasta la primera hoja (esto suma un costo I/O de h). Ahora, nos dicen que cada página está ocupada en un 60 % por lo que la cantidad de páginas que utilizaremos para guardar todas las tuplas es $\frac{10^6}{0{,}6 \cdot P}$. Cómo ya tenemos la primera página (al bajar hasta la primera hoja), le restamos 1 a la suma.

2. Como queremos las tuplas donde $a < 50$, la cantidad de tuplas que tenemos son 50 (recordemos que los valores empiezan en 0). Con esto, la cantidad de páginas donde están almacenadas estas 50 tuplas, siguiendo la lógica de la consulta anterior son $\frac{50}{0{,}6 \cdot P}$. Aquí también debemos sumar el costo de bajar por el árbol de h, y el -1 por tener la primera página necesaria al bajar por el árbol.

3. Como queremos la tupla donde $a = 50$, solo debemos bajar por el árbol (h).

4. La idea es similar a la consulta 2, solo que en vez de 50 tuplas, tendremos 49 (que son la cantidad de tuplas entre 50 y 100 no incluidos). Entonces, se suma el costo de bajar por el árbol (h) y la cantidad de páginas $\frac{49}{0{,}6 \cdot P}$. Se resta -1 por la misma razon cómo allá.

B+Tree Unclustered

1. Para los casos unclustered, sabemos que cada página tiene M punteros a páginas de disco. Entonces, la cantidad de páginas para almacenar todas las tuplas será $\frac{10^6}{M}$ (ya que $10^6 = paginas \cdot M$). En el caso de obtener todas las tuplas, el costo I/O será el bajar por el árbol (h), más la cantidad de páginas a recorrer ($\frac{10^6}{M}$), más la cantidad de tuplas (10^6). En el factor $\frac{10^6}{M}$ ya está incluida la primera página qué obtuvimos bajando por el árbol (-1).

2. Se sigue la misma lógica que para la consulta anterior, pero considerando solo 50 tuplas (entre 0 y 49).

3. En el caso de obtener un valor particular, debemos bajar por el árbol (h) hasta la página con el puntero en, y luego acceder a la tupla (+1).

4. La misma lógica que para las dos primera consultas, pero considerando solo las 49 tuplas (entre 51 y 99).

Hash Índex Clustered

1. Como tenemos 10^6 buckets, no tendremos colisiones (cada uno corresponde a una tupla). Si queremos acceder a todas las tupla de R, vamos a tener que "hacer una llamada O(1)" por cada tupla, lo que sería costo I/O de 10^6

2. La misma lógica aplica en esta consulta, pero con 50 tuplas.

3. Aquí simplemente tenemos que acceder al valor, por lo que el costo sería 1.

4. La misma lógica que consultas 1 y 3, pero con 49 tuplas.

Hash Índex Unclustered

1. Cuando queremos todas las tuplas, vamos a tener que ir a buscar la página del índice (+1) y luego la página indicada por el puntero (+1), esto para cada tupla. Entonces el costo será $2 \cdot 10^6$
2. Seguimos la misma lógica de la primera consulta, pero con 50 tuplas ($2 \cdot 50 = 100$).
3. La misma lógica pero con una sola tupla ($2 \cdot 1 = 2$).
4. La misma lógica de la primera consulta, pero con 49 tuplas ($2 \cdot 49 = 98$).