

Parcialito N° 7: Costos

EJ 1

T título = "LICENCIADO", género = "M" (profesores)

① Como el atributo título tiene un índice secundario asociado se aplica primero esta condición y luego se selecciona del resultado los tuplos que cumplen con la selección de género.

Por índice secundario el costo resulta:

$$\text{cost}(\text{prtit}) = \text{Height}(I(\text{título}, \text{profesores})) + \frac{n(\text{profesores})}{V(\text{título}, \text{profesores})}$$

Reemplazamos con los datos del enunciado y obtenemos:

$$\text{cost}(\text{prtit}) = 2 + \frac{500}{10} = 52$$

Ahora bien, siguiendo con la condición restante, por ser un atributo simple sin índice se realiza una búsqueda de tipo full scan de donde el costo resulta de leer todos los bloques y aplicar la condición. En este sentido, si se aplica la segunda condición después de cada acceso a memoria se aplicará el filtro para cada uno de los bloques obtenidos.

Finalmente el costo de la consulta es 52

Por otra parte vemos que ocurre cuando la primera condición aplicada es la del atributo simple, donde por full scan resulta que el costo es:

$$\text{cost}(\text{género}) = B(\text{profesores}) = \frac{n(\text{profesores})}{f(\text{profesores})} = \frac{500}{25} = 20$$

Si aplicamos la condición del atributo con índice vemos que:

$$\text{cost}(\text{prtit}) = 2 + 20 = 22$$

Finalmente el costo de la consulta es 22.

De ambos análisis llegamos a la conclusión de que el método de acceso más eficiente accede a 22 bloques y resulta de aplicar file scan para el atributo genero e index scan para el atributo título en ese orden.

② Asumido que los datos son independientes, la cantidad de tuplas devueltas por la consulta resulta:

$$n(\text{licenciados}) = \frac{n(\text{profesores})}{V(\text{título, profesores}) \cdot V(\text{genero, profesores})} = \frac{500}{10 \cdot 2} = 25 \quad \text{2ra}$$

ET 2 $\text{me gusta} \bowtie \text{id_usuario} \neq \text{id_usuario}' \wedge \text{id_publicacion} = \text{id_publicacion}' \text{ me gusta}'$

① Puesto que no existen atributos con índices asociados, aplicamos el método de loops anidados por bloque, restringiendo la cantidad de memoria disponible según los datos del enunciado.

Vemos que con 999 bloques de memoria disponibles y donde la cantidad de bloques de la relación es 100.000 podemos calcular el costo como:

$$\begin{aligned} \text{cost}_L(\text{me gusta} \bowtie \text{me gusta}') &= B(\text{me gusta}) + \text{upper}\left(\frac{B(\text{me gusta})}{999}\right) B(\text{me gusta}') \\ &= 10 \cdot 100.000 \end{aligned}$$

Por otra parte, podríamos calcular el costo de la junta mediante sort-merge. Vemos que como los bloques no entran en memoria, se deberá utilizar un sort externo, por lo cual el costo resultará:

$$\begin{aligned} \text{cost}_S(\text{me gusta} \bowtie \text{me gusta}') &= B(\text{me gusta}) + B(\text{me gusta}') + 2B(\text{me gusta}) [\log_{1000}(B(\text{me gusta}))] + \\ &+ 2B(\text{me gusta}') [\log_{1000}(B(\text{me gusta}'))] \end{aligned}$$

Puesto que $B(\text{me gusta}) = B(\text{me gusta}')$ podemos reducir el cálculo a:

$$\begin{aligned} \text{cost}_S(\text{me gusta} \bowtie \text{me gusta}') &= 2B(\text{me gusta}) + 4B(\text{me gusta}) [\log_{1000}(B(\text{me gusta}))] \\ &\approx 866.667 \end{aligned}$$

Por último, calculamos el costo usando Hash Grace que resulta:

$$\text{cost}_G (\text{me gusta} \text{ y } \text{me gusta}') = 3 (B(\text{me gusta}) + B(\text{me gusta}')) \\ = 600.000$$

Finalmente, el método de acceso más eficiente para resolver la consulta se obtiene con Hash Grace (o junto Hash) y tiene un costo de 600.000.

② la cantidad de tuplas devueltas se puede calcular como:

$$n (\text{me gusta} \text{ y } \text{me gusta}') = \frac{n(\text{me gusta}) \cdot n(\text{me gusta}')}{\max(V(\text{id-usuario}, \text{me gusta}), V(\text{id-publicacion}, \text{me gusta}))} \\ = \frac{100.000.000 * 100.000.000}{10.000.000} = \boxed{1.000.000.000} \quad \text{RTA}$$