

Parcialito V: Optimización de Consultas
Base de Datos 75.15/75.28/95.05 - Cátedra Román
Segundo Cuatrimestre 2025

Melanie García Lapegna - 111848

Ejercicio

La famosa banda de rock “Eruca Sativa” está buscando lugares donde presentar su nuevo hit “Lío”, con el objetivo de estar en las últimas provincias del país en las que no han tocado aún. Para ello utilizan una base de datos relacional con el siguiente esquema:

- Antros(id_antro, nombre, dirección, ciudad, provincia, capacidad)
- Disponibilidad (id_antro, fecha)

Se busca disponibilidad para 50 mil personas en 5 días de enero, con la siguiente consulta:

```
SELECT a.id_antro, nombre
FROM antros a INNER JOIN disponibilidad d USING(id_antro)
WHERE (a.provincia = 'Catamarca' OR a.provincia = 'La
Rioja') AND a.capacidad >= 50,000
AND (d.fecha BETWEEN '2025-01-06' AND '2025-01-10)
```

Se tiene la siguiente información de catálogo:

ANTROS	DISPONIBILIDAD
n(antros) = 10.000	n(disponibilidad) = 100.000
B(antros) = 2.000	B(disponibilidad) = 5.000
V(provincia, antros) = 20	V(id_antro, disponibilidad) = 10.000
	V(fecha, disponibilidad) = 100
H(I(provincia, antro)) = 2	H(I(id_antro, disponibilidad)) = 4
	H(I(fecha, disponibilidad)) = 3

Estime el **costo** de realizar esta consulta y la **cantidad de filas** que serán devueltas, contando la información de catálogo y sabiendo que:

- Se dispone de 100 bloques de memoria disponibles para la operación
- Que los índices indicados no son de clustering
- Que un 10% de los antros tienen capacidad para al menos 50 mil personas.

Evalúe **diferentes alternativas** e indique la que tiene el **menor costo**.

Solución

Estimación de cardinalidad

Comenzaré estimando la cantidad de filas devueltas por la consulta, ya que esta se mantiene invariante independientemente de la metodología empleada para resolver la consulta. Fui en orden de la consulta.

Comienzo estimando la cardinalidad de realizar un **join natural** entre Antros(a) y Disponibilidad(d) por id_antro.

$$\begin{aligned}n(a \bowtie d) &= n(a) * n(d) / \max(\text{Var}(\text{id_antro}, a), \text{Var}(\text{id_antro}, d)) \\&= 10.000 * 100.000 / 10.000 \\&= \mathbf{100.000}\end{aligned}$$

Obs: Considero que $\text{Var}(\text{id_antro}, a) = 10.000$ ya que estimo que id_antro es la PK de la tabla Antros.

Luego, realizó la **selección** que filtra devolverá únicamente las filas que tengan como provincia a Catamarca o La Rioja. (R = resultado de operacion anterior)

Lo pienso como:

$$\begin{aligned}n(\sigma_{\text{provincia} = 'Catamarca'} \text{ OR } \sigma_{\text{provincia} = 'La Rioja'}(R)) &= n(\sigma_{\text{provincia} = 'Catamarca'}(R)) \cup n(\sigma_{\text{provincia} = 'La Rioja'}(R)) \\&= n(\sigma_{\text{provincia} = 'Catamarca'}(R)) + n(\sigma_{\text{provincia} = 'La Rioja'}(R))\end{aligned}$$

$$\begin{aligned}n(\sigma_{\text{provincia} = 'Catamarca'}(R)) &= n(\text{salida_op_anterior}) / V(\text{prov}, a) = \mathbf{100.000} / 20 = 5000 \\ \text{Idem para } n(\sigma_{\text{provincia} = 'La Rioja'}(R)).\end{aligned}$$

$$\therefore n(\sigma_{\text{provincia} = 'Catamarca'} \text{ OR } \sigma_{\text{provincia} = 'La Rioja'}(R)) = 5000 + 5000 = \mathbf{10.000}$$

Continuo con el dato de que únicamente el 10% de los antros tienen capacidad para al menos 50.000 personas

$$n(\sigma_{\text{capacidad} \geq 50.000}(R)) = n(\text{salida_op_anterior}) * 0.1 = \mathbf{10.000} * 0.1 = \mathbf{1.000}$$

Finalmente hago la **selección** por el rango de fecha solicitado.

Se que hay 100 fechas distintas y me piden 5 en particular.

Asumiendo una distribución uniforme de las fechas, voy a plantear para saber cuantas filas tiene una fecha en particular y luego eso lo multiplico por 5(cantidad de fechas en el rango solicitado).

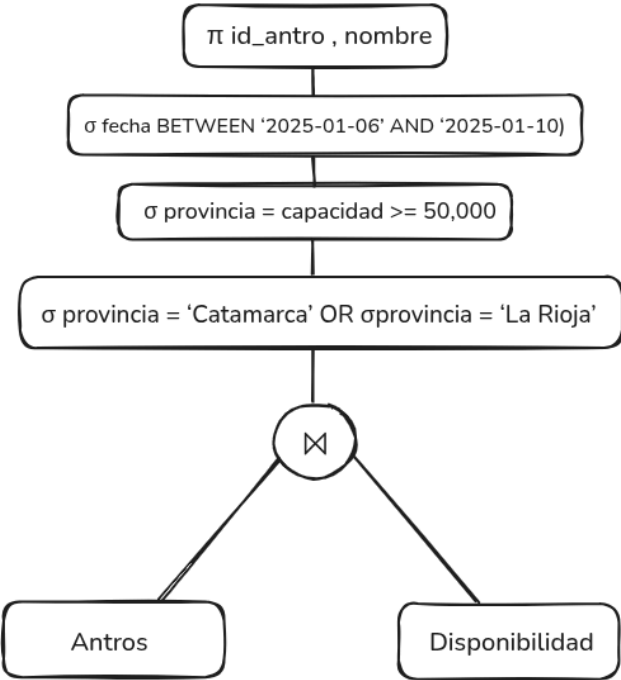
$$n(\sigma_{\text{fecha} = \text{una_dentro_rango}}(R)) = n(\text{salida_op_anterior}) / V(\text{fecha}, d) = \mathbf{1.000} / 100 = \mathbf{10}$$

=> Finalmente la **cantidad de filas** devuelta por la consulta es $\mathbf{10} * 5 = \mathbf{50}$.

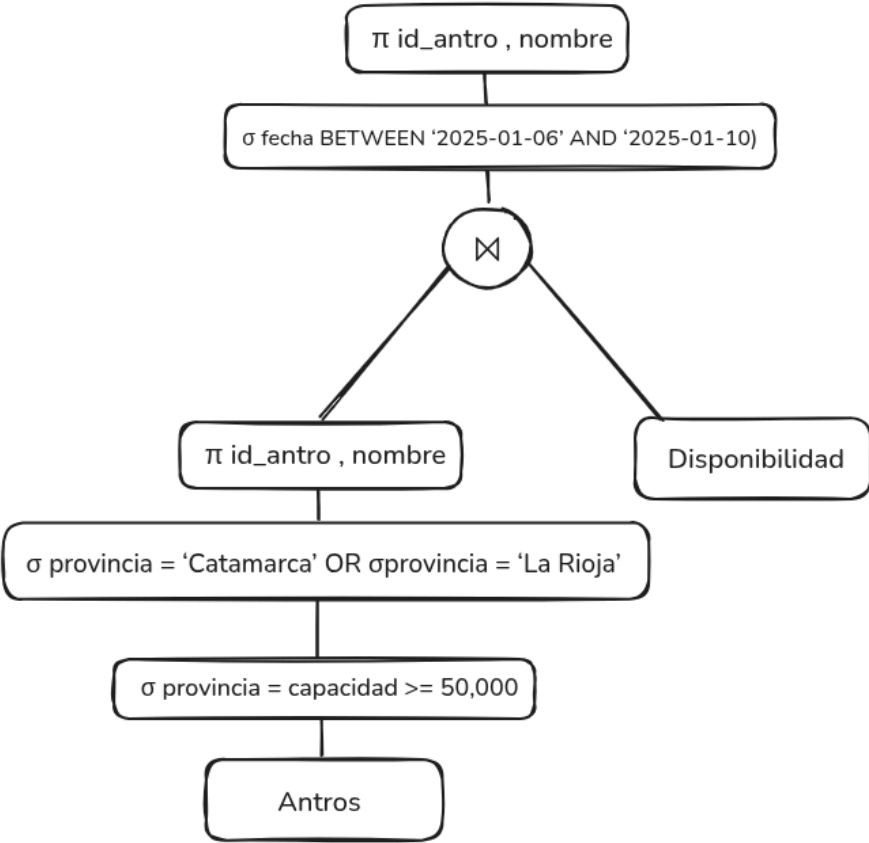
Evaluación de alternativas.

A continuación mostrare las distintas alternativas que plantee y cual fue la de menor costo.

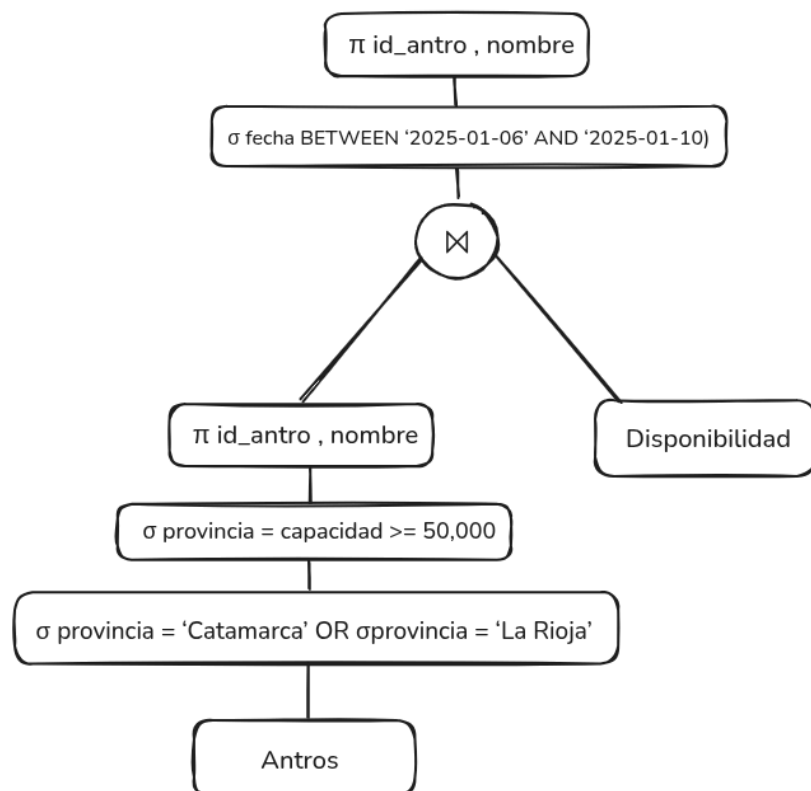
Alternativa 1



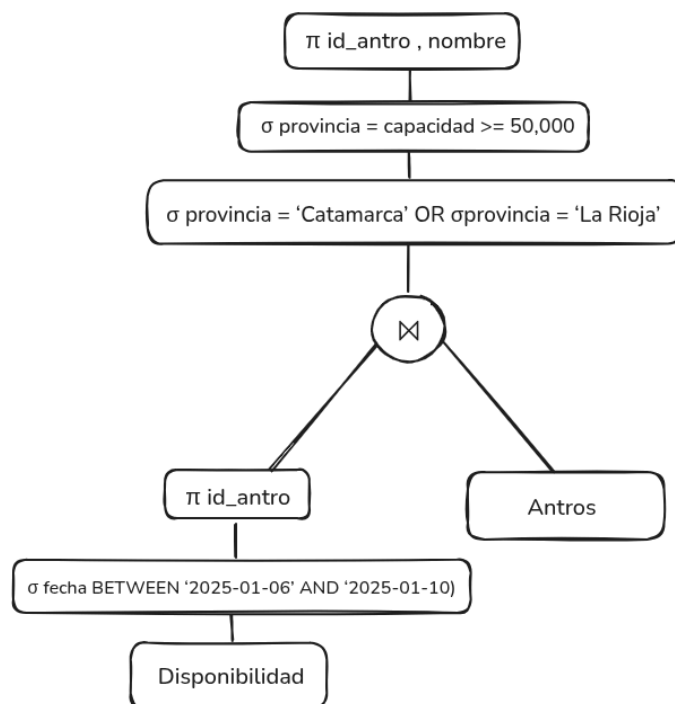
Alternativa 2



Alternativa 3



Alternativa 4



Comencé estimando el costo de la alternativa 3 ya que en esta arranco haciendo un select en la tabla con menor cantidad de filas y además aprovecho el indice que la tabla de Antros tiene sobre el atributo provincia.

Inicialmente realice un **Index Scan** por el índice de provincia. Como hice al estimar la cardinalidad de la consulta, también pense al OR como un AND de las dos condiciones de esta sobre el atributo provincia.

$$\begin{aligned}
 &\text{Costo}(\sigma_{\text{provincia} = \text{'Catamarca'} \text{ OR } \sigma_{\text{provincia} = \text{'La Rioja'}}(\text{Antros})) = \dots \\
 &= \text{Costo}(\sigma_{\text{provincia} = \text{'Catamarca'}}(\text{Antros})) + \text{Costo}(\sigma_{\text{provincia} = \text{'La Rioja'}}(\text{Antros})) \\
 &= \text{Height}(\text{I}(\text{provincia}, a)) + n(\text{Antros}) / V(\text{provincia}, \text{antros}) + \text{Height}(\text{I}(\text{provincia}, a)) + n(\text{Antros}) / \\
 &V(\text{provincia}, \text{antros}) \\
 &= 2 + (10.000/20) + 2 + (10.000/20) \\
 &= \mathbf{1004}
 \end{aligned}$$

Luego aprovecharé y utilizaré pipelines para hacer la selección de la capacidad.

A cada fila que llega de la operación anterior se la filtra con la condición de la capacidad.

Como asumo uniformidad en la distribución de provincias, me quedo con el 10% de las filas devueltas por la operación anterior.

Calculo la cardinalidad de la operación anterior y sobre aplico la selección que me devolvería el 10% de estas filas.

$$\begin{aligned}
 n(\text{operacion_anterior}) &= n(\text{Antros}) / V(\text{provincia}, \text{antros}) + n(\text{Antros}) / V(\text{provincia}, \text{antros}) \\
 &= 10.000/20 + 10.000/20 = \mathbf{1.000}
 \end{aligned}$$

Entonces, fila por fila se irá aplicando la condición de la capacidad y se descartarán a las que no cumplan con esta.

$$n(\text{operacion_anterior}) = \mathbf{1.000} * 0.1 = \mathbf{100}$$

Y sobre este retorno se vuelve a usar un pipeline para quedarse únicamente con los atributos id_antro y nombre, el cual no tendrá costo ya que únicamente lo hace sobre las filas que le llegan del pipeline. Además, no se pide quitar repetidos (aunque al id_antro ser PK tampoco tendríamos ese problema).

Sobre ese retorno se debe hacer un join con la tabla de disponibilidad, mi **primera opción** es aprovechar que esta tiene un índice en id_antro y utilizarlo para realizar un **loop con único índice**.

El costo de esta operación es

$$\begin{aligned}
 &\text{Costo}(\text{retorno_pipeline} \bowtie \text{Disponibilidad}) = \dots \\
 &= n(\text{filas_pipeline}) * (\text{Height}(\text{I}(\text{id_antro}, \text{disponibilidad})) + n(\text{disponibilidad}) / V(\text{id_antro}, \text{disponibilidad})) \\
 &= \mathbf{100} * (4 + 100.000/10.000) = \mathbf{1400}
 \end{aligned}$$

Esto ya que por cada fila que le llega del pipeline se realiza un **Index Scan** en la tabla disponibilidad con el índice id_antro.

Sobre estos **1400** bloques, por cada fila se vuelve a proyectar sobre los atributos id_antro y nombre, lo cual nuevamente es una operación sin costo por la misma razón anteriormente mencionada.

Por lo tanto, siguiendo esta alternativa el costo total es de **2404 bloques**.

Analizo la **alternativa 2**, sabiendo ya cual es el costo de lo que estime iba a ser el de menor. En esta alternativa deberia arrancar realizando un **File Scan** para quedarme con los antros que cuenten con la capacidad solicitada.

El costo de esta operacion es de **2000** bloques, luego cuando realice el **loop con único índice** como en la anterior alternativa me daría un costo mayor a 2404 bloques. ($3400 > 2404$)

Analizo la **alternativa 1**, si realizo al inicio un loop anidado por bloque, primero que nada no estaria cumpliendo con la heuristica de primero realizar las selecciones, pero si lo quiero hacer a pesar de esto, el costo seria:

Considerando que $M=100$ y que utilizo la tabla de antros como chunk.

$$\begin{aligned}\text{Costo} &= B(\text{antros}) + B(\text{antros})/M - 2 * B(\text{disponibilidad}) \\ &= 2.000 + 2.000/98 * 5.000 > \mathbf{2404}.\end{aligned}$$

Por lo tanto esta alternativa no es conveniente.

Si lo realizara aprovechando el indice en la tabla de disponibilidades y utilizara loop con unico indice:

$$\begin{aligned}\text{Costo} &= B(\text{antros}) + (n(\text{antros}) * (H(I(\text{id_antro}, \text{disponibilidad})) + n(\text{disponibilidad})/V(\text{id_antro}, \\ &\text{disponibilidad}))) \\ &= 2.000 + 10.000 * (4 + (100.000/10.000)) > \mathbf{2404}.\end{aligned}$$

Por lo tanto esta alternativa tampoco es conveniente.

Analizo la **alternativa 4**, en esta aprovecharia el indice que tiene la tabla disponibilidades sobre fechas.

Para utilizar este indice como minimo deberia acceder a uno de estos

$$\begin{aligned}\text{Costo} &= H(I(\text{fecha}, \text{disponibilidad})) + n(\text{disponibilidad})/V(\text{fecha}, \text{disponibilidad}) \\ &= 3 + 100.000/100 = 1003\end{aligned}$$

Ademas, de la altura del indice, se estiman unos 1000 bloques por fecha, por lo que de base deberia acceder a $1000 * 5 = 5000$ bloques sin contar operaciones extras.

Por lo tanto la mejor alternativa es la 3.