BASES DE DATOS
AVANZADAS

Viernes 13 DE MAYO, 2022

Actividad N°3.

Iván Cáceres,
Sebastián Cornejo,
Tobías Guerrero,
Marcelo Muñoz

Profesor

Juan Ricardo Giadach

Índice

Introducción

En esta entrega, se realizan modificaciones a las tablas Personas1 y Personas2, mediante el archivo proporcionado, que recibe el nombre de quincemil. Para así obtener tiempos de ejecución de las consultas (DELETE - UPDATE), en donde la primera corresponde a la eliminación de ruts que tienen su último dígito par y la segunda corresponde a la actualización de la edad, la cual queda como esta más 15 cuando el rut tiene su último dígito impar, en esta última se toma una restricción que dice que la edad máxima debe ser 99, si la supera, la edad queda igual a 0. Estas consultas se realizan 3 veces, con un reinicio del computador entre cada una, para que la memoria caché no intefiera en los resultados. Luego de todo lo anterior, se realiza un análisis explicando diferencias de cada proceso e indicando como optimizarlo.

▪ Características del entorno de trabajo

Este trabajo se realiza en un computador con las siguientes características (Software/Hardware):

Elemento	Valor
Nombre del SO	Microsoft Windows 10 Pro
Versión	10.0.19044 Compilación 19044
Descripción adicional del SO	No disponible
Fabricante del SO	Microsoft Corporation
Nombre del sistema	DESKTOP-BTF3JK5
Fabricante del sistema	HP
Modelo del sistema	HP EliteBook 830 G5
Tipo de sistema	x64-based PC
SKU del sistema	5FV92EC#ABM
Procesador	Intel(R) Core(TM) i5-8350U CPU @ 1.70GHz, 1896 Mhz, 4 procesadores princi...
Versión y fecha de BIOS	HP Q78 Ver. 01.19.00, 13-01-2022
Versión de SMBIOS	3.1
Versión de controladora integr...	4.109
Modo de BIOS	UEFI
Fabricante de la placa base	HP
Producto placa base	83B3
Versión de la placa base	KBC Version 04.6D.00
Rol de plataforma	Móvil
Estado de arranque seguro	Activado
Configuración de PCR7	Se necesita elevación de privilegios para ver
Directorio de Windows	C:\WINDOWS
Directorio del sistema	C:\WINDOWS\system32
Dispositivo de arranque	\Device\HarddiskVolume1
Configuración regional	México
Capa de abstracción de hardw...	Versión = "10.0.19041.1566"
Nombre de usuario	DESKTOP-BTF3JK5\Ivaaaaaaan
Zona horaria	Hora est. Sudamérica Pacífico
Memoria física instalada (RAM)	16,0 GB

Figura 1: Datos del sistema.

Elemento	Valor
Descripción	Unidad de disco
Fabricante	(Unidades de disco estándar)
Modelo	SAMSUNG MZVLB512HAJQ-000H1
Bytes/sector	512
Medio cargado	Sí
Tipo de medio	Fixed hard disk
Particiones	3
Bus SCSI	0
Unidades lógicas SCSI	0
Puerto SCSI	0
Id. de destino SCSI	0
Sectores/pista	63
Tamaño	476,94 GB (512.105.932.800 bytes)
Nº total de cilindros	62.260
Nº total de sectores	1.000.206.900
Nº total de pistas	15.876.300
Pistas/cilindro	255
Partición	Disco #0, partición #0
Tamaño de partición	260,00 MB (272.629.760 bytes)
Desplazamiento inicial de parti...	1.048.576 bytes
Partición	Disco #0, partición #1
Tamaño de partición	475,76 GB (510.847.352.832 bytes)
Desplazamiento inicial de parti...	290.455.552 bytes
Partición	Disco #0, partición #2
Tamaño de partición	922,00 MB (966.787.072 bytes)
Desplazamiento inicial de parti...	511.137.808.384 bytes

Figura 2: Especificaciones del disco.

Cabe destacar que la unidad de almacenamiento que se utiliza es del tipo ssd M.2, por lo que los resultados esperados son de un tiempo reducido en comparación a una unidad hdd.

- ## Eliminación de Indice

  En este punto se pide deshacer el índice creado para la tabla personas1 en la actividad anterior, para esto se realiza la instrucción *"DROP INDEX indice;"* donde 'indice' es el nombre otorgado a este en la primera actividad.

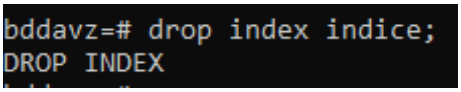

Figura 3: Eliminación Indice

- ## Comprobación existencia llave primaria

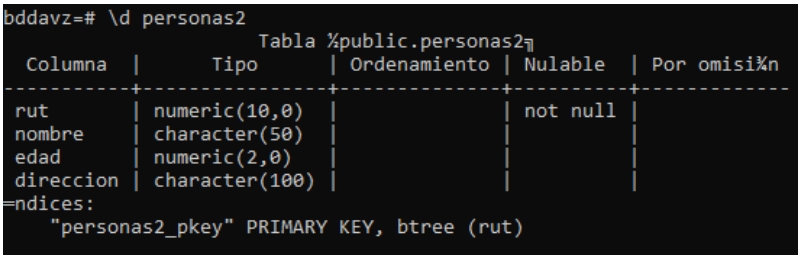

Figura 4: Comprobación existencia llave primaria

Proceso

En el archivo quincemil hay 15.000 RUT con los que hay que hacer el siguiente proceso:

- Si el último dígito del rut es par, se debe eliminar la fila correspondiente de la tabla PERSONASx.
- Si el último dígito es impar, se debe sumar 15 a la edad, teniendo cuidado de no sobrepasar la edad 99. Si eso ocurriera, la edad debe quedar en cero.

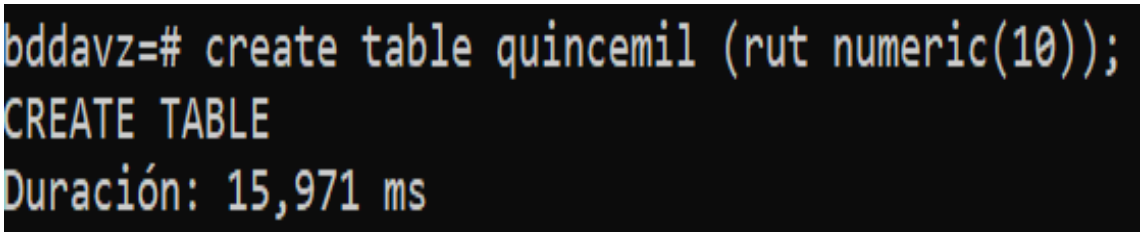

Figura 5: Creacion Tabla quincemil

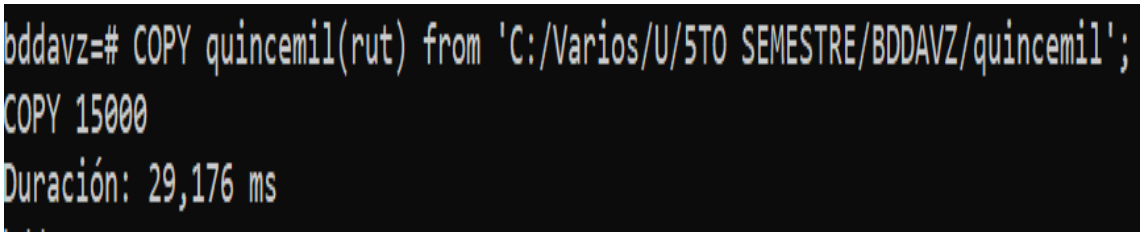

Figura 6: Insercion de Datos en Tabla quincemil

→ Proceso en Personas1

Se pide en PERSONAS1, hacer el proceso descrito previamente, midiendo el tiempo que demora el proceso.

Para realizar aquello se realizan 3 pruebas en donde se resume en una tabla los tiempos y se crea un gráfico para ver visualmente el comportamiento de estos. Para cada una de las pruebas se ejecutan los siguientes comandos:

```
1 BEGIN;
2 DELETE from personas1 where rut in (select rut from quincemil) and
      rut%2=0;
3 UPDATE personas1 set edad = CASE WHEN edad <=84 THEN edad+15 WHEN edad>84
      THEN 0 END where rut in (select rut from quincemil) and rut%2!=0;
4 ROLLBACK;
```

Prueba 1

```
bddavz=# BEGIN;
BEGIN
Duración: 0,276 ms
bddavz=*#
bddavz=*# DELETE from personas1 where rut in (select rut from quincemil) and rut%2=0;
DELETE 7527
Duración: 24602,055 ms (00:24,602)
bddavz=*#
bddavz=*# UPDATE personas1 set edad = CASE WHEN edad <=84 THEN edad+15 WHEN edad >84 THEN 0 END where rut in (select rut from quincemil) and rut%2!=0;
UPDATE 7473
Duración: 26075,734 ms (00:26,076)
bddavz=*# ROLLBACK;
ROLLBACK
Duración: 11,800 ms
```

Figura 7: Comando primera prueba

Prueba 2

```
bddavz=# BEGIN;
BEGIN
Duración: 0,425 ms
bddavz=*#
bddavz=*# DELETE from personas1 where rut in (select rut from quincemil) and rut%2=0;
DELETE 7527
Duración: 24269,223 ms (00:24,269)
bddavz=*#
bddavz=*# UPDATE personas1 set edad = CASE WHEN edad <=84 THEN edad+15 WHEN edad >84 THEN 0 END where rut in (select rut from quincemil) and rut%2!=0;
UPDATE 7473
Duración: 28119,220 ms (00:28,119)
bddavz=*#
bddavz=*# ROLLBACK;
ROLLBACK
Duración: 0,898 ms
```

Figura 8: Comando segunda prueba

Prueba 3

```
bddavz=# BEGIN;
BEGIN
Duración: 0,465 ms
bddavz=*#
bddavz=*# DELETE from personas1 where rut in (select rut from quincemil) and rut%2=0;
DELETE 7527
Duración: 24194,368 ms (00:24,194)
bddavz=*#
bddavz=*# UPDATE personas1 set edad = CASE WHEN edad <=84 THEN edad+15 WHEN edad >84 THEN 0 END where rut in (select rut from quincemil) and rut%2!=0;
UPDATE 7473
Duración: 31617,882 ms (00:31,618)
bddavz=*#
bddavz=*# ROLLBACK;
ROLLBACK
Duración: 1,419 ms
```

Figura 9: Comando tercera prueba

Prueba	DELETE (s)	UPDATE (s)	TOTAL (s)
Prueba 1	24,602	26,076	50,678
Prueba 2	24,269	28,119	52,388
Prueba 3	24,194	31,618	59,737
Promedio tiempos	24,355	28,6	54.267

Cuadro 1: Tiempos de ambas consultas en personas1.

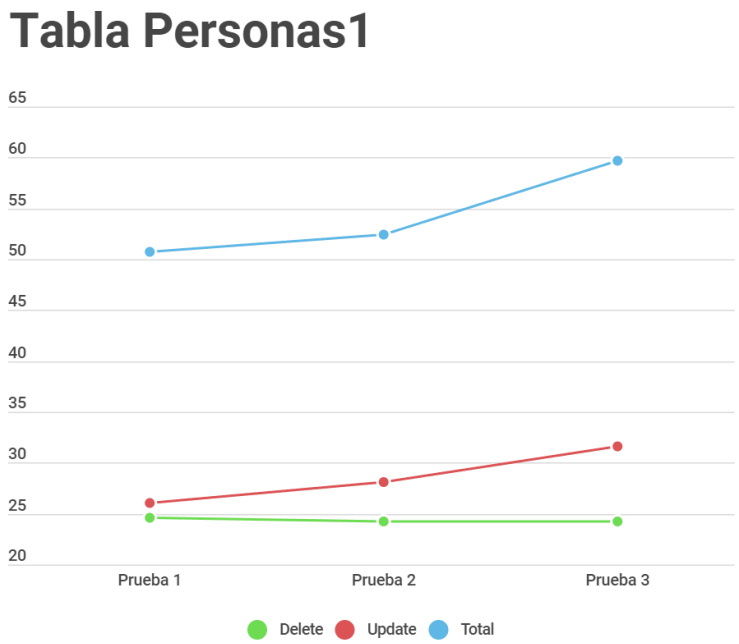

Figura 10: Gráfico pruebas en personas1.

→ Proceso en Personas2

Se pide en PERSONAS2, repetir el proceso anterior, también con su respectiva tabla de tiempos y su gráfico.

```
1 BEGIN;
2 DELETE from personas2 where rut in (select rut from quincemil) and
      rut%2=0;
3 UPDATE personas2 set edad = CASE WHEN edad <=84 THEN edad+15 WHEN edad>84
      THEN 0 END where rut in (select rut from quincemil) and rut%2!=0;
4 ROLLBACK;
```

Prueba 1

```
bddavz=# BEGIN;
BEGIN
Duración: 0,413 ms
bddavz=*#
bddavz=*# DELETE from personas2 where rut in (select rut from quincemil) and rut%2=0;
DELETE 7527
Duración: 4729,626 ms (00:04,730)
bddavz=*#
bddavz=*# UPDATE personas2 set edad = CASE WHEN edad <=84 THEN edad+15 WHEN edad >84 THEN 0 END where rut in (select rut from quincemil) and rut%2!=0;
UPDATE 7473
Duración: 1451,597 ms (00:01,452)
bddavz=*#
bddavz=*# ROLLBACK;
ROLLBACK
Duración: 0,266 ms
```

Figura 11: Comando primera prueba

Prueba 2

```
bddavz=# BEGIN;
BEGIN
Duración: 0,419 ms
bddavz=*#
bddavz=*# DELETE from personas2 where rut in (select rut from quincemil) and rut%2=0;
DELETE 7527
Duración: 4493,428 ms (00:04,493)
bddavz=*#
bddavz=*# UPDATE personas2 set edad = CASE WHEN edad <=84 THEN edad+15 WHEN edad >84 THEN 0 END where rut in (select rut from quincemil) and rut%2!=0;
UPDATE 7473
Duración: 1463,738 ms (00:01,464)
bddavz=*#
bddavz=*# ROLLBACK;
ROLLBACK
Duración: 0,211 ms
```

Figura 12: Comando segunda prueba

Prueba 3

```
bddavz=# BEGIN;
BEGIN
Duración: 0,416 ms
bddavz=*#
bddavz=*# DELETE from personas2 where rut in (select rut from quincemil) and rut%2=0;
DELETE 7527
Duración: 4538,809 ms (00:04,539)
bddavz=*#
bddavz=*# UPDATE personas2 set edad = CASE WHEN edad <=84 THEN edad+15 WHEN edad >84 THEN 0 END where rut in (select rut from quincemil) and rut%2!=0;
UPDATE 7473
Duración: 1499,955 ms (00:01,500)
bddavz=*#
bddavz=*# ROLLBACK;
ROLLBACK
Duración: 0,206 ms
```

Figura 13: Comando tercera prueba

Prueba	DELETE (s)	UPDATE (s)	TOTAL (s)
Prueba 1	4,730	1,452	6,182
Prueba 2	4,493	1,464	5,957
Prueba 3	4,539	1,5	6,039
Promedio tiempos	4,587	1,805	6,059

Cuadro 2: Tiempos de ambas consultas en personas2.

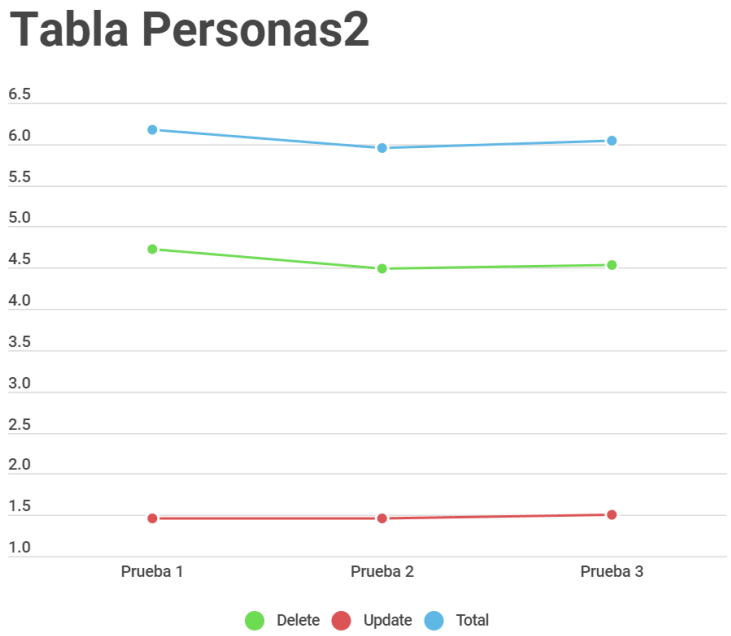

Figura 14: Grafico pruebas en personas2

Análisis de tiempos obtenidos

Explains

A continuación se muestran los planes de ejecución para cada caso:

```
bddavz=*# explain DELETE from personas1 where rut in (select rut from quincemil) and rut%2=0;
                                    QUERY PLAN
-------------------------------------------------------------------------------
 Delete on personas1  (cost=418.50..1967154.44 rows=0 width=0)
   ->  Hash Semi Join  (cost=418.50..1967154.44 rows=75 width=12)
         Hash Cond: (personas1.rut = quincemil.rut)
         ->  Seq Scan on personas1  (cost=0.00..1966080.00 rows=249564 width=14)
               Filter: ((rut % '2'::numeric) = '0'::numeric)
         ->  Hash  (cost=231.00..231.00 rows=15000 width=14)
               ->  Seq Scan on quincemil  (cost=0.00..231.00 rows=15000 width=14)
(7 filas)
```

Figura 15: Explain Delete Personas1.

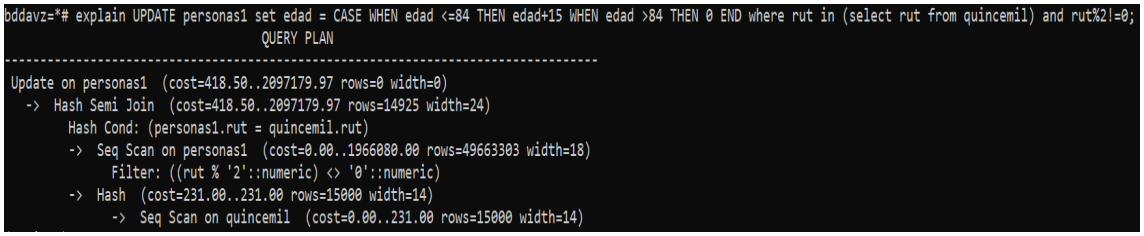

Figura 16: Explain Update Personas1.

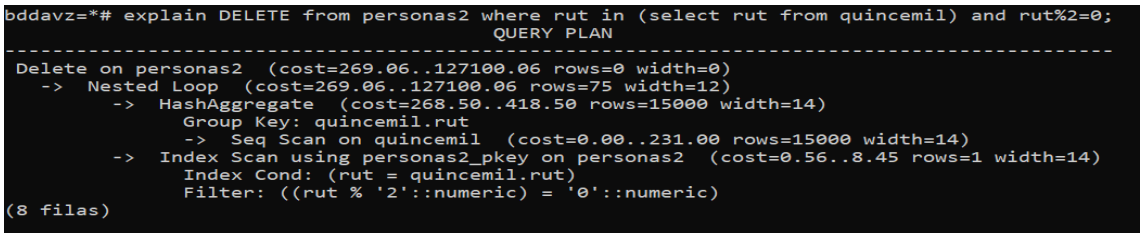

Figura 17: Explain Delete Personas2.

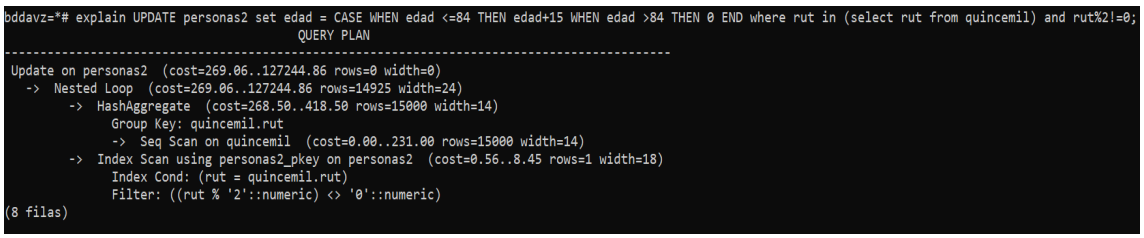

Figura 18: Explain Update Personas2.

Utilizando los explains se observa que en ambas tablas, tanto para la instrucción DELETE como UPDATE se recorre la tabla quincemil de manera secuencial, sin embargo al realizar la búsqueda de estos 15000 rut para verificar su paridad en personas1 se realiza de manera secuencial, mientras que en personas2 esta instrucción es por índices (logarítmica). Es por esto que en todas las consultas la asociación de ruts que coinciden en ambas tablas considerando su paridad serán más rápidas en la tabla personas2, ya que esta cuenta con llave primaria en rut, la cual es utilizada en esta instrucción.

Resumenes Tiempos

Ahora se realiza una tabla con los totales de cada consulta en cada prueba, para personas1 y personas2.

Prueba	personas1	personas2
Prueba 1	50,678	6,182
Prueba 2	52,388	5,957
Prueba 3	59,737	6,039
Promedio tiempos	54,267	6,059

Cuadro 3: Tiempos totales.

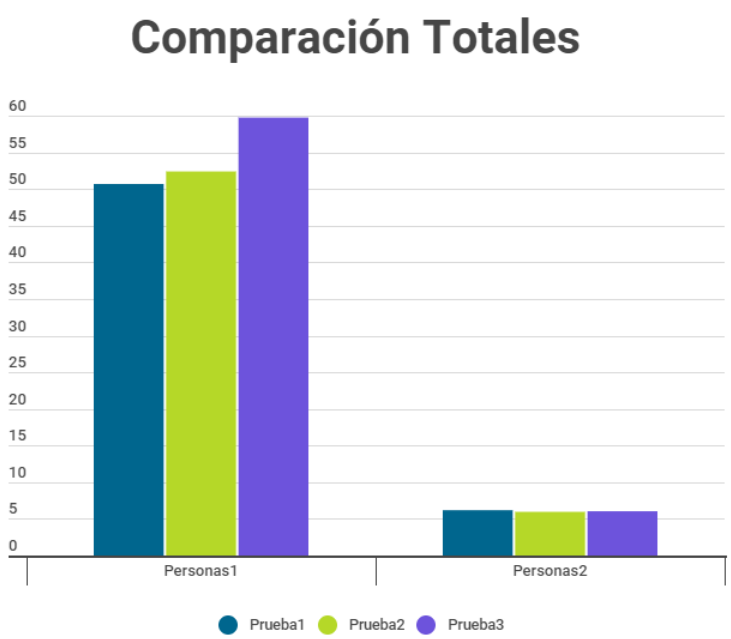

Figura 19: Gráfico tiempos totales.

Consulta	personas1	personas2
Tiempo Promedio (s)	54,268	6,0594

Cuadro 4: Promedio de tiempos totales de cada consultas.

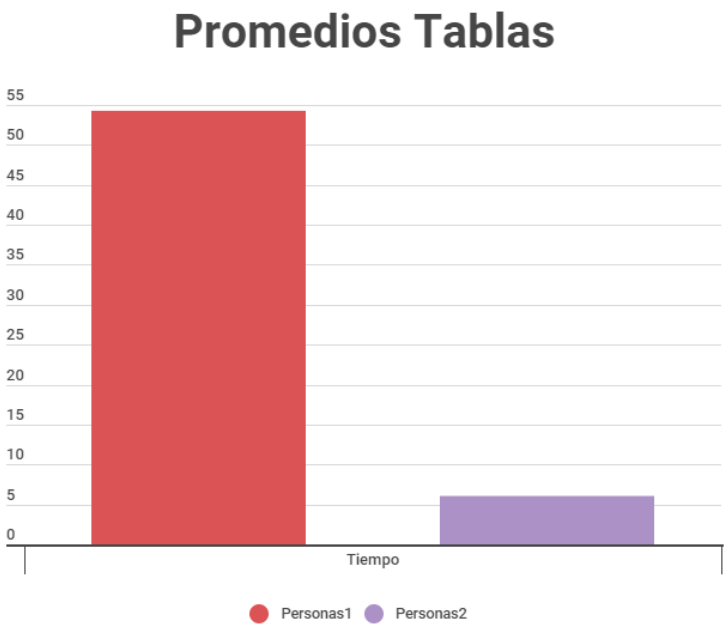

Figura 20: Promedio tiempos.

Es posible observar que los tiempos en personas2 son mucho menores a los de personas1, esto es posiblemente responsabilidad del índice de personas2, ya que al crear un índice se realiza un búsqueda logarítmica, lo cual lo hace más eficiente para la cantidad de datos usada.

También según la teoría la acción de actualizar, debería tomar más tiempo que eliminar por la cantidad de pasos a realizar, pero lo que se observa en la práctica es lo contrario, esto se puede deber a que postgresql es inteligente y probablemente dentro las cosas que hace, una de ellas es guardar en caché datos que se podrían usar en el futuro, etc.

Aun así aunque los costos sean menores en eliminar, lo que menos tiempo toma es actualizar, y tal como se dijo anteriormente, probablemente debido a la "inteligencia" de postgresql para realizar consultas, modificaciones, etc.

Para optimizar el proceso, quizás se puede crear un trigger, o en la tabla Personas1 añadir un índice haciendo mucho más rápido el proceso en esta, ya que el join con la tabla quincemil se realiza de forma logarítmica en vez de secuencial, lo cual para este volumen de datos es eficiente.

Comparación tiempos con actividades anteriores

Operación/Tabla	Actividad 1 Inserción	Actividad 2 Recuperación	Actividad 3 Eliminación	Actividad 3 Actualización
personas1 (s)	298,33	10,32	24,355	28,6
personas2 (s)	1824,58	4,16	4,587	1,805

Cuadro 5: Tiempos en las actividades realizadas.

En estas 3 actividades se han realizado cuatro operaciones, inserción, recuperación, eliminación y actualización de datos en tablas de un sistema de bases de datos. De estas operaciones, la recuperación, eliminación y actualización son mas eficientes en la tabla personas2, mientras que la inserción lo es en la tabla personas1. En teoría la inserción, eliminación y actualización de una tabla en un sistema de bases de datos toma más tiempo en una tabla con índice, sin embargo se observa que la eliminación y actualización demora menos en la tabla personas2 que cuenta con llave primaria, esto se debe a que se realiza un join con la tabla quincemil, al tener llave primaria en rut este join se realiza de manera logarítmica, acelerando el proceso, mientras que en personas1 este join se ejecuta secuencialmente, al ser 15000 filas, secuencialmente buscarlar en una tabla con 50000000 de filas es mucho mas lento, es por esto que demora menos en personas2.

Conclusión

En el presente trabajo se hacen 2 acciones, actualizar y eliminar, para esto se utiliza logica. También posterior a esto se realiza un análisis sobre los tiempos y diferencias de rendimiento obtenidos y se indica como es posible optimizarlos.

También se comparan los resultados obtenidos con los tiempos de las actividades anteriores y se explican/analizan las diferencias.

Viendo esta y las anteriores actividades es posible observar que solo para la inserción es mucho mejor una tabla sin índice, ya que en recuperación, eliminación y actualización la tabla con índice es superior en rendimiento/tiempo, esto se debe a que al realizar el join con la tabla quincemil, la tabla sin índice lo hace secuencial mientras que la con índice lo realiza de manera logarítmica, acelerando la consulta.