Bases de Datos a Grande Escala

Entrega 3

**ANDRÉS CAMPOS CUIÑA**

**FECHA DE ENTREGA: 10/12/2021**

ÍNDICE

[1 Ejercicio 1 1](#_Toc88306410)

[2 Ejercicio 2 6](#_Toc88306411)

# Ejercicio 1

**Probar el uso de la replicación. Para esto, borramos todas la tablas (en cascada) y las creamos de nuevo. Antes de crearlas modificar el script para eliminar las claves foráneas cláusulas *references*, ya que no se puede usar replicación en CITUS con claves foráneas. Antes de distribuir los datos, asignar el valor 2 a la variable de factor de replicación ("SET citus.shard\_replication\_factor = 2"). Cargar de nuevo los datos. Probar una de las consultas anteriores. Pausar una de las máquinas *worker*. Probar de nuevo la consulta. Intentar varias veces si es necesario, para que el coordinador se dé cuenta que uno de los *worker* no responde.**

En primer lugar, debemos borrar todas las tablas (en cascada). Mediante el siguiente código de SQL podemos obtener todas las instrucciones para borrar las tablas:



Tras la ejecución del código SQL anterior obtenemos el siguiente código, que nos permitirá borrar todas las tablas:



Una vez hemos eliminado todas las tabas, las volvemos a crear mediante el script que se nos proporciona en el enunciado de esta entrega, modificándolo para eliminar las claves foráneas (en la cláusula *references*), ya que no se puede usar replicación en CITUS con claves foráneas. Este script quedaría de la siguiente manera:





Tras ejecutar este script ya tendríamos las tablas creadas de nuevo. A continuación, establecemos el factor de replicación al valor 2 mediante la siguiente instrucción:



Después de ejecutar esta instrucción, debemos distribuir las tablas y ejecutar las instrucciones de inserción de datos tal y como se muestra en el enunciado de esta entrega.

Ahora probamos la ejecución de la consulta:



Esta consulta nos da el siguiente resultado en un tiempo de 1,557s :

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Ahora pausamos la máquina **cluster2** y volvemos a probar la ejecución de la misma consulta. Después de que el coordinador se de cuenta de que uno de los nodos *worker* no responde, la consulta se puede ejecutar de forma correcta en el nodo que sigue activo, aunque el tiempo de ejecución es mayor (de 4,413s) debido a que no se puede paralelizar la consulta entre los dos nodos *worker*:

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

# Ejercicio 2

**Buscar información sobre el formato de almacenamiento columnar en Citus (**[**https://docs.citusdata.com/en/v10.1/admin\_guide/table\_management.html#columnar-storage**](https://docs.citusdata.com/en/v10.1/admin_guide/table_management.html#columnar-storage)**). Probar a crear la tabla "peliculas" en formato columnar y comprobar si hay ganancia en tiempo de ejecución en las consultas anteriores.**

Ejecutamos las dos consultas anteriores con los dos nodos *worker* activados y con un factor de replicación igual a 2. Para la consulta siguiente se obtiene un tiempo de ejecución de 2,441s:



Mientras que para la consulta siguiente se obtiene un tiempo de ejecución de 2,913s:



A continuación, borramos las tablas como se hizo en el ejercicio anterior y las volvemos a crear con la tabla “películas” en formato columnar, mediante el siguiente código SQL:



La creación del resto de tablas es igual que antes. Después, seguimos los mismos pasos que en el ejercicio anterior para distribuir las tablas y para llevar a cabo la inserción de los datos. Una vez tenemos esto hecho ya podemos ejecutar las dos consultas anteriores y comparar los tiempos de ejecución.

Con la tabla “peliculas” en formato columnar la primera de las dos consultas anteriores se ejecuta en un tiempo de 761ms mientras que la segunda de las consultas anteriores se ejecutó en un tiempo de 1,625s.

Como se puede ver en la siguiente tabla:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **No Columnar** | **Columnar** | **Mejora** |
| **Consulta 1** | 2,441s | 0,761s | x3,2 |
| **Consulta 2** | 2,913s | 1,625s | X1,8 |

En ambas consultas, se ejecutó de forma más rápida la consulta que se corrió sobre la tabla “películas” en formato columnar. Esto se debe a que las consultas sobre esta tabla columnar son más rápidas y más eficientes que sobre una tabla que no lo sea. Esto se debe a que a que estas consultas sólo acceden a algunas de estas columnas, pero a todas las filas (debido a las agregaciones) por lo que al no tener que obtener las filas de diferentes nodos *worker* las consultas son más eficientes.