

Fragmentación con PostgreSQL

Profesor: Heider Sanchez

ACLs: Nicolás Arroyo, Mariana Capuñay, Sebastián Loza

Se le pide crear las siguientes tablas “employees” y “salaries” y cargar los datos del dataset “data2.zip”. Tal vez requiera realizar algunas modificaciones para que la data pueda ser cargado al PostgreSQL correctamente.

P1. Fragmentación con PARTITION BY LIST

- Crear la tabla employees1 indicando que será fragmentado por el atributo dept_no.
- Cargue los datos en la tabla employees1
- Analice los resultados que se obtienen al ejecutar una misma consulta en el atributo dept_no en ambas tablas.
- Use el comando Explain Analyze y coloque los tiempos en una tabla comparativa.

	dept_no = d005	dept_no = d004	dept_no = d007
employees			
employees1			

Usar *SET enable_partition_pruning = on;*

P2. Fragmentación con PARTITION BY RANGE

- Crear la tabla employees2 indicando que será fragmentado por rango sobre el año de la fecha de contrato:

PARTITION BY RANGE (date_part('year', hire_date))

- Realizar la fragmentación con respecto al año (vector: [1988, 1994]).
- Cargue los datos en la tabla employees2.
- Elabore el cuadro de comparación de costos para tres consulta por rango diferentes sobre el atributo hire_date. Cuidar que la query acceda a una sola partición.

	Query 1	Query 2	Query 3
employees			
employees1			

- Aplique el índice btree sobre el atributo hire_date en ambas tablas. Note usted que el índice aplicado sobre la tabla particionada es un índice distribuido. Vuelva a ejecutar las consultas y anote los tiempos.

	Query 1	Query 2	Query 3
employees			
employees1			

P3. Fragmentación con dos atributos.

- Crear la tabla employees3 para ser particionada en dos atributos.
- Agregar la columna “salary” a la tabla employees para guardar el último salario de cada empleado. Obtener el valor del salario desde la tabla “salaries”.
- Considerar un segundo predicado de consulta sobre el atributo salary
 - Proponer un vector de particionamiento lo más equitativo posible.
- Hay dos opciones de fragmentación con dos atributos:
 - Opción 1: Primero fragmentar la tabla en el atributo hire_date. Luego aplicar una sub fragmentación sobre cada partición en el atributo salary.
 - Opción 2: Crear la tabla employees3 indicando ambos atributos en la partición: PARTITION BY RANGE (date_part('year', hire_date), salary
- Mostrar el plan de ejecución con Explain Analyze para tres consultas que incluya ambos atributos, hire_data y salary.

	Query 1	Query 2	Query 3
employees			
employees1			

Entregable: el script y resultado en formato PDF.