

Fragmentación con PostgreSQL

Profesor Heider Sanchez

Se le pide crear la siguiente tabla y cargar los datos que va adjunto en la tarea.

```
CREATE TABLE employees
( employee_id  INTEGER
, first_name   VARCHAR(20)
, last_name    VARCHAR(25) NOT NULL
, email        VARCHAR(25) NOT NULL
, phone_number VARCHAR(20)
, hire_date    TIMESTAMP NOT NULL
, job_id       VARCHAR(10) NOT NULL
, salary       NUMERIC(8,2)
, commission_pct NUMERIC(2,2)
, manager_id   INTEGER
, department_id INTEGER
, CONSTRAINT emp_salary_min CHECK (salary
> 0)
)
```

En la carpeta compartida hay dos datasets: data1.zip, data2.zip. El segundo dataset contiene muchos más datos que el primero, pero debe realizar las modificaciones necesarias para que la data pueda ser cargado al PostgreSQL.

P1. Fragmentación con PARTITION BY LIST

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

	department_id ==50	department_id ==80	department_id ==100
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: [1995, 1998]).

- Crear un índice para el atributo hire_date en cada partición.
- Cargue los datos en la tabla employees2.
- Elabore el cuadro de comparación de costos al ejecutar tres consultas sobre el atributo hire_date.

	Query 1	Query 2	Query 3
employees			
employees2			

P3. Fragmentación con dos atributos.

- Considerar un segundo predicado de consulta sobre el atributo salary (vector: [30k,70k]).
- Hay dos opciones de fragmentación con dos atributos:
 - Sobre la fragmentación anterior aplicar una sub fragmentación sobre cada partición con el atributo salary.
 - Crear otra 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 las tres consultas que usted ha definido en el ejercicio anterior.

Entregable: el script y resultado en formato PDF.