Universidad de La Sabana
Maestría en Analítica Aplicada

Taller 2: Contenerización con Docker y PostgreSQL

Esteban Bernal - Informatics Engineering - 271938
Juan Montes - Informatics Engineering - 272113
Juan Gómez - Informatics Engineering - 286774

Profesor: Hugo Franco, Ph.D.
Herramientas de Big Data

1. Problema

El objetivo de este taller fue contenerizar una base de datos **PostgreSQL** usando **Docker**, cargar datos de población, esperanza de vida y PIB per cápita, y realizar consultas y comparaciones de rendimiento entre inserción fila por fila (INSERT) y carga masiva (COPY).

2. Método de solución

La solución se abordó mediante:

- Creación de un contenedor Docker con PostgreSQL mediante `docker-compose`.
- Desarrollo de un script en Python para cargar y transformar los datos.
- Inserción de datos en la base usando dos métodos distintos: INSERT y COPY.
- Ejecución de consultas SQL de análisis comparativo.

2.1. Algoritmo propuesto

Listing 1: Fragmento del código en Python

```
conn = psycopg2.connect(
    host="localhost",
    port=5433,
    database="bigdatatools1",
    user="psqluser",
    password="psqlpass"
)
```

3. Resultados

Se obtuvieron los siguientes tiempos de ejecución:

- **INSERT fila por fila:** 15.7 segundos
- **COPY en bloque:** 0.17 segundos

La diferencia muestra que COPY es considerablemente más eficiente para cargas masivas.

4. Discusión

Los resultados permiten concluir:

- COPY es el método recomendado para carga inicial de grandes volúmenes.
- El uso de Docker facilita la portabilidad del entorno de base de datos.
- PostgreSQL combinado con Python y Pandas permite integrar análisis avanzados de datos.

Referencias

- Documentación oficial de PostgreSQL: `https://www.postgresql.org/docs/`
- Psycopg2: `https://www.psycopg.org/`
- Pandas: `https://pandas.pydata.org/`