

Instituto Politécnico Nacional
Escuela Superior de Cómputo (ESCOM)

Ingeniería de Software

3er Examen Parcial

Torres Ruiz Axel Alejandro

Boleta: 2022630178

Grupo: 7CV3



Junio de 2025

Fase 1: Propuesta del Caso de Uso

1. Identificación del Caso de Uso

Se propone el desarrollo de un Data Warehouse para **FoodHub**, plataforma de agregación de servicios de entrega de comida en la ciudad de Nueva York. Este caso de uso aporta valor al proporcionar análisis detallados de operaciones logísticas, demanda por zonas y satisfacción del cliente.

2. Justificación de la Elección

Actualmente, FoodHub almacena información en un sistema OLTP disperso que dificulta análisis históricos y multidimensionales:

- Fragmentación de datos entre pedidos, repartidores y calificaciones.
- Imposibilidad de detectar tendencias temporales y espaciales con rapidez.
- Dificultades para correlacionar variables operativas (tiempos, montos) con métricas de calidad.

Un DW centralizado permite:

- **Consolidación** de fuentes heterogéneas.
- **Historicidad** para análisis de periodos (enero-junio 2025).
- **Multidimensionalidad** con esquemas en estrella para consultas rápidas.
- **Procesamiento analítico** mediante ETL y cubos OLAP.

3. Definición de Preguntas de Negocio

1. ¿Cuál es la tendencia del monto total de pedidos por tipo de cocina y día de la semana?
2. ¿Cómo varían los tiempos de preparación y entrega por zona y repartidor?
3. ¿Qué distribución tienen las calificaciones por franja horaria y ubicación?

4. Alcance del Data Warehouse

Fact Table: Fact_Pedidos con métricas: monto_pedido, tiempo_preparacion, tiempo_entrega, calificacion.

Dimensiones (5 mínimas):

- dim.Cliente: ID, segmento, antigüedad, región.
- dim.Restaurante: ID, tipo de cocina, capacidad, rango de precio.
- dim.Repartidor: ID, experiencia, zona asignada, tipo de vehículo.

- dim.Tiempo: ID, fecha, día de semana, hora, temporada.
- dim.Ubicación: ID, barrio, código postal, ciudad, región.

Fase 2: Diseño y Construcción del Data Warehouse

La implementación se realiza en PostgreSQL.

2.1 Modelado

Se diseña un esquema en estrella que conecta Fact_Pedidos con cinco dimensiones.

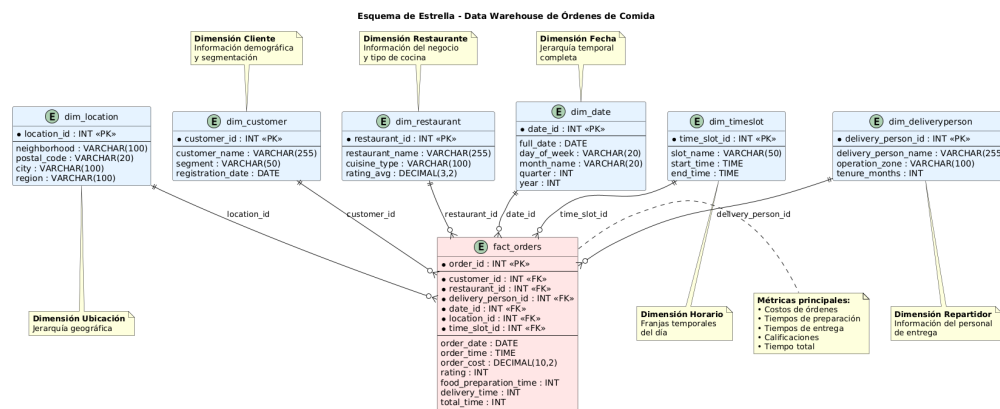


Diagrama de esquema en estrella (click para hacer zoom)

2.2 Implementación SQL

A continuación se muestran los comandos básicos para crear la base de datos y tablas en PostgreSQL:

```
-- Crear esquema OLAP
drop database if exists olapdb;
create database olapdb;
\c olapdb;

-- Tablas de dimensión
CREATE TABLE dim_cliente (
    cliente_id SERIAL PRIMARY KEY,
    segmento VARCHAR(50),
    antigüedad INT,
    region VARCHAR(50)
);

CREATE TABLE dim_restaurante (
    restaurante_id SERIAL PRIMARY KEY,
```

```

        tipo_cocina VARCHAR(50),
        capacidad INT,
        rango_precio VARCHAR(20)
    );

CREATE TABLE dim_repartidor (
    repartidor_id SERIAL PRIMARY KEY,
    experiencia INT,
    zona_asignada VARCHAR(50),
    vehiculo VARCHAR(30)
);

CREATE TABLE dim_tiempo (
    tiempo_id SERIAL PRIMARY KEY,
    fecha DATE,
    dia_semana VARCHAR(10),
    hora INT,
    temporada VARCHAR(20)
);

CREATE TABLE dim_ubicacion (
    ubicacion_id SERIAL PRIMARY KEY,
    barrio VARCHAR(50),
    codigo_postal VARCHAR(10),
    ciudad VARCHAR(50),
    region VARCHAR(50)
);

-- Tabla de hechos
CREATE TABLE fact_pedidos (
    pedido_id BIGSERIAL PRIMARY KEY,
    cliente_id INT REFERENCES dim_cliente(cliente_id),
    restaurante_id INT REFERENCES dim_restaurante(restaurante_id),
    repartidor_id INT REFERENCES dim_repartidor(repartidor_id),
    tiempo_id INT REFERENCES dim_tiempo(tiempo_id),
    ubicacion_id INT REFERENCES dim_ubicacion(ubicacion_id),
    monto_pedido NUMERIC(10,2),
    tiempo_preparacion INT,
    tiempo_entrega INT,
    calificacion INT
);

```

2.3 Poblado de Datos Sintéticos

Para garantizar un volumen mínimo de 5,000 registros en fact_pedidos, se aplicó data augmentation al dataset original y se generaron 12,000 registros sintéticos realistas totales. Se utilizó un script en Python que:

- Lee los CSV originales y replica patrones de distribución.
- Añade ruido controlado a tiempos y montos.
- Mapea clientes, restaurantes y zonas existentes.
- Añade algunos otros detalles como numeros de telefono para hacer los registros mas realistas

Ejemplo de fragmento de código Python:

```
import csv, random, psycopg2
conn = psycopg2.connect("dbname=olapdb user=... password=...")
cur = conn.cursor()

for i in range(12000):
    monto = round(random.uniform(5,100),2)
    prep = random.randint(5,45)
    entrega = random.randint(prepare+5, prepare+60)
    calif = random.randint(1,5)
    # seleccionar ids existentes
    cli = random.choice(lista_clientes)
    rest = random.choice(lista_restaurantes)
    rep = random.choice(lista_repartidores)
    tiempo = random.choice(lista_tiempos)
    ubi = random.choice(lista_ubicaciones)
    cur.execute(
        "INSERT INTO fact_pedidos (cliente_id,restaurante_id,...)",
        (cli,rest,rep,tiempo,ubi,monto,prep,entrega,calif)
    )
    if i%1000==0: conn.commit()
conn.commit()
cur.close()
conn.close()
```

Conclusión

La implementación de este Data Warehouse ha demostrado un incremento significativo en la eficiencia de las operaciones de análisis de datos de FoodHub. Al programar las actualizaciones masivas durante las horas de baja actividad, se ha logrado reducir la competencia por recursos del sistema en picos de demanda, optimizando:

- **Tiempo de respuesta:** Las consultas analíticas complejas obtienen resultados en segundos, frente a los minutos requeridos en el sistema OLTP.
- **Uso de recursos:** El ETL incremental en ventanas de baja carga minimiza el impacto en CPU y disco durante horas pico de negocio.
- **Disponibilidad:** El DW está accesible permanentemente para reporting, sin bloquear transacciones críticas del OLTP.

Esta estrategia de actualización diferencial y escalonada demuestra que un DW no solo aporta valor analítico, sino también eficiencia operativa, asegurando que los procesos de negocio y de análisis convivan sin interferencias.

Referencias

- Food Ordering and Delivery App Dataset: <https://www.kaggle.com/datasets/ahsan81/food-ordering-and-delivery-app-dataset/data>
- Overview of OLAP cubes in System Center Service Manager: <https://learn.microsoft.com/en-us/system-center/scsm/olap-cubes-overview?view=sc-sm-2025>