Federico Ceballos Torres
Informe Técnico
A3 – M2 – DS – SoyHenry

Avance 3: Diseño del Data Warehouse y Pipeline ETL en Snowflake

1. Introducción

El objetivo del Avance 3 es transformar la base de datos operacional de FleetLogix en un sistema analítico capaz de soportar reportes históricos, métricas por conductor, vehículo y ruta, y análisis de tendencias. Para esto, se diseñó e implementó un **Data Warehouse en Snowflake**, utilizando un **modelo estrella (star schema)** y un **pipeline ETL automatizado en Python**.

La solución permite pasar de consultas operativas a análisis avanzados, contemplando historización, control de calidad, cálculo de métricas y carga incremental diaria.

---

2. Diseño del Modelo Estrella

El modelo estrella está compuesto por una tabla de hechos central (**fact_deliveries**) y múltiples dimensiones que permiten analizar los datos desde distintas perspectivas.

2.1 Tabla de Hechos – fact_deliveries

Contiene una fila por cada entrega completada.
Incluye todas las métricas analíticas clave:

- Tiempo de entrega
- Retrasos
- Distancia recorrida
- Combustible utilizado
- Eficiencia (km/L)
- Ingresos estimados
- Costo por entrega
- Entregas/hora
- Número de entregas en el viaje

Además, guarda claves foráneas hacia todas las dimensiones SCD (date, time, driver, vehicle, customer, route).

2.2 Dimensiones Implementadas

dim_date

Calendario completo con:

- Año
- Mes
- Día
- Día de la semana
- Indicador de fin de semana

dim_time

Desglosa tiempos en:

- Hora
- Minuto
- Segundo

Para análisis por turnos y patrones horarios.

dim_customer

Información del cliente:

- Nombre
- Ciudad
- Categoría
- Fecha de primera entrega
- Total de entregas

dim_driver ***(SCD Tipo 2)***

Incluye:

- Nombre completo
- Licencia
- Datos de contacto
- Estado laboral
- Columnas **valid_from**, **valid_to**, **is_current**

Permite historizar cambios (por ej., si cambia el número de teléfono o estado del conductor).

dim_vehicle ***(SCD Tipo 2)***

Contiene:

- Tipo de vehículo
- Capacidad
- Combustible
- Estado
- Último mantenimiento
- Columnas SCD para cambios históricos

dim_route

Incluye detalles fijos de ruta:

- Ciudades origen/destino
- Distancia
- Peajes
- Duración estimada

---

3. Implementación en Snowflake

Se creó:

- Base de datos: **FLEETLOGIX_DW**
- Warehouse: **FLEETLOGIX_WH**
- Schema: **ANALYTICS**

Además:

✓ Crecimiento automático

AUTO_RESUME y AUTO_SUSPEND habilitados para optimizar costos.

✓ Time Travel

DATA_RETENTION_TIME_IN_DAYS = 1
Permite consultar el estado de la base 24 horas atrás.

✓ Vistas seguras

Snowflake permite crear vistas con acceso restringido (ej. ventas solo ve sus clientes). Esto queda contemplado en la arquitectura aunque no es obligatorio para el script.

4. Pipeline ETL Automatizado (Python + PostgreSQL + Snowflake)

Se desarrolló un pipeline completo con las etapas:

4.1 Extracción

Desde PostgreSQL:

- Datos de entregas
- Viajes
- Vehículos
- Conductores
- Rutas

La extracción se limita al **día previo** para evitar duplicados y cargar solo cambios.

4.2 Transformación

El ETL calcula:

- delivery_time_minutes
- delay_minutes
- trip_duration_hours
- deliveries_in_trip
- deliveries_per_hour
- fuel_efficiency_km_per_liter
- cost_per_delivery
- revenue_per_delivery

Además realiza validaciones de calidad:

- No tiempos negativos
- No pesos fuera de rango
- Eliminación de registros inconsistentes

Finalmente, prepara columnas SCD para dimensiones históricas.

4.3 Carga de Dimensiones (SCD Tipo 2)

Para **driver** y **vehicle**, el pipeline:

1. Detecta cambios en datos sensibles
2. "Cierra" la versión anterior (valid_to = ayer)
3. Inserta una nueva versión con valid_from = hoy

4.4 Carga de la Fact Table

Todos los registros transformados se insertan en **fact_deliveries**, junto con un etl_batch_id para trazabilidad.

4.5 Cálculo de Totales Pre-Aggregados

Se genera la tabla daily_totals con:

- Total de entregas
- Promedio de tiempo de entrega
- Eficiencia promedio
- Revenue total

Esto acelera dashboards de BI.

4.6 Automatización

El pipline usa **schedule** para ejecutarse a las **02:00 AM** todos los días.

---

5. Conclusión

El Avance 3 implementa un Data Warehouse robusto, con historización completa, métricas avanzadas y pipeline automatizado. La arquitectura permite escalar hacia análisis predictivos, dashboards ejecutivos y modelos de machine learning.

FleetLogix adquiere una base sólida para sus operaciones analíticas y de toma de decisiones.
