

Pipeline ETL FleetLogix - Data Warehouse

Descripción

Pipeline automático que extrae datos diarios de PostgreSQL (base operacional), los transforma con métricas de negocio, y los carga en Snowflake (Data Warehouse) para análisis.

Archivos

`avance3_dw.py` - Pipeline ETL Principal

Script de Python que ejecuta el proceso ETL automáticamente todos los días a las 2:00 AM.

¿Qué hace?

1. **Extracción**: Saca los datos del último día desde PostgreSQL (deliveries, vehículos, conductores, rutas)
2. **Transformación**: Calcula métricas importantes:
 - Tiempo de entrega y retrasos
 - Eficiencia de combustible
 - Costos y revenue por entrega
 - Indicador de entregas a tiempo
3. **Carga**: Inserta todo en Snowflake en un modelo estrella:
 - 6 Dimensiones: fecha, tiempo, vehículo, conductor, ruta, cliente
 - 1 Tabla de Hechos: fact_deliveries con todas las métricas

Script de Snowflake (SQL)

Define el esquema del Data Warehouse en Snowflake.

¿Qué crea?

- **Warehouse y Base**: `FLEETLOGIX_WH` y `FLEETLOGIX_DW`

- ****6 Tablas Dimensiones****:

- ``dim_date``: Calendario 2020-2030 con atributos (día, mes, trimestre, etc.)
- ``dim_time``: 48 slots de media hora con turnos
- ``dim_vehicle``, ``dim_driver``, ``dim_route``, ``dim_customer``: Datos maestros con claves autoincrementales
- ****1 Tabla de Hechos****: ``fact_deliveries`` con métricas y foreign keys a dimensiones
- ****Vistas Seguras****: Para diferentes roles (ventas, operaciones)

****Importante****: Las claves surrogadas (vehicle_key, driver_key, etc.) se generan automáticamente con ``IDENTITY``.

Modelo Dimensional

Modelo estrella con 6 dimensiones + 1 hecho. Las dimensiones se cargan una sola vez, los hechos se cargan diariamente.

****Transformaciones implementadas:****

- ☒ Cálculo de métricas (7 métricas calculadas)
- ☒ Validación de calidad (filtro de pesos)
- ☒ Tabla de totales pre-calculados
- ☒ SCD Type 2 en conductores (histórico de cambios)

Automatización

El script usa la librería ``schedule`` para ejecutarse automáticamente a las 2:00 AM todos los días. Mantiene un log en ``etl_pipeline.log``.

```
244
245   select * from dim_customer;|
246   select * from dim_date;
247   select * from dim_driver;
248   select * from dim_route;
249   select * from dim_time;
250   select * from dim_vehicle;
251   select * from fact_deliveries;
252
253
```

Results (just now)

Table Chart

404 rows 47ms

	# CUSTOMER_KEY	# CUSTOMER_ID	A CUSTOMER_NAME	A CUSTOMER_TYPE	A CITY	FIRST_DELIVER
1	159	159	Harold Sanabria Ángel	Individual	Bogotá	2026-02-23
2	160	160	Germán Héctor Moreno	Individual	Cartagena	2026-02-23
3	249	249	Mauricio Vera	Individual	Cali	2026-02-23
4	161	161	Jorge Johan Hernández C	Individual	Bogotá	2026-02-23

```
244
245   select * from dim_customer;
246   select * from dim_date;
247   select * from dim_driver;
248   select * from dim_route;
249   select * from dim_time;
250   select * from dim_vehicle;
251   select * from fact_deliveries;
252
253
```

Results (just now)

Table Chart

4,018 rows 50ms

	# DATE_KEY	FULL_DATE	# DAY_OF_WEEK	A DAY_NAME	# DAY_OF_MONTH	# DAY_OF_YEAR	# W
1	20200101	2020-01-01	2	Wednesday	1	1	
2	20200102	2020-01-02	3	Thursday	2	2	
3	20200103	2020-01-03	4	Friday	3	3	
4	20200104	2020-01-04	5	Saturday	4	4	

Home

avance_03.sql

+

Settings

avance 3 > avance_03.sql

▶

▼

👤

🏠

📄

Share

...

244

245

246

247

248

249

250

251

252

253

select * from dim_customer;

select * from dim_date;

select * from dim_driver;

select * from dim_route;

select * from dim_time;

select * from dim_vehicle;

select * from fact_deliveries;

Results (just now)

×

Table

Chart

🔍

🔗

400 rows ⓘ

34ms

📄

🕒

🔍	# DRIVER_KEY	# DRIVER_ID	EMPLOYEE_CODE	FULL_NAME	LICENSE_NUMBER	LICENSE_EXPIRY
1	601	1	EMP0001	Javier Bustos	2785736751	2027-10-20
2	602	2	EMP0002	Ana Barrios	7767538527	2028-10-12
3	603	3	EMP0003	Sandra Perdomo	8884394611	2029-01-08
4	604	4	EMP0004	Orlando Ospina	6467215542	2026-04-15

🏠

📄

🏠

📄

📖

Feedback

Home

avance_03.sql

+

Settings

avance 3 > avance_03.sql

▶

▼

👤

🏠

📄

Share

...

244

245

246

247

248

249

250

251

252

253

select * from dim_customer;

select * from dim_date;

select * from dim_driver;

select * from dim_route;

select * from dim_time;

select * from dim_vehicle;

select * from fact_deliveries;

Results (just now)

×

Table

Chart

🔍

🔗

48 rows ⓘ

35ms

📄

🕒

🔍	# ROUTE_KEY	# ROUTE_ID	ROUTE_CODE	ORIGIN_CITY	DESTINATION_CITY	DISTANCE_KM
1	313	1	R001	Bogotá	Medellín	439.06
2	314	2	R002	Bogotá	Medellín	484.11
3	315	3	R003	Bogotá	Medellín	457.97
4	316	4	R004	Bogotá	Cali	418.57

🏠

📄

🏠

📄

📖

Feedback

Home

SQL avance_03.sql

+

avance 3 > avance_03.sql

▶

▼

🔍

🏠

📄

Share

...

244

245

246

247

248

249

250

251

252

253

select * from dim_customer;

select * from dim_date;

select * from dim_driver;

select * from dim_route;

select * from dim_time;

select * from dim_vehicle;

select * from fact_deliveries;

Results (just now)

×

Table

Chart

🔍

🔍

48 rows ⓘ

52ms

⬇

🕒

🔍	# TIME_KEY	# HOUR	# MINUTE	# SECOND	⌵ TIME_OF_DAY	⌵ HOUR_24	⌵ HOUR_12	⌵ AM_PM
1	0	0	0	0	Noche	00:00	12:00	AM
2	30	0	30	0	Noche	00:30	12:30	AM
3	100	1	0	0	Noche	01:00	1:00	AM
4	130	1	30	0	Noche	01:30	1:30	AM

⬅

➡

🏠

🔍

📄

📖

Feedback

Home

SQL avance_03.sql

+

avance 3 > avance_03.sql

▶

▼

🔍

🏠

📄

Share

...

244

245

246

247

248

249

250

251

252

253

select * from dim_customer;

select * from dim_date;

select * from dim_driver;

select * from dim_route;

select * from dim_time;

select * from dim_vehicle;

select * from fact_deliveries;

Results (just now)

×

Table

Chart

🔍

🔍

200 rows ⓘ

71ms

⬇

🕒

🔍	# VEHICLE_KEY	# VEHICLE_ID	⌵ LICENSE_PLATE	⌵ VEHICLE_TYPE	# CAPACITY_KG	⌵ FUEL_TYPE	🔍
1	401	1	UDA125	Van	1500.00	gasolina	20
2	402	2	HHE242	Camión Grande	5000.00	diesel	20
3	403	3	VXR658	Van	1500.00	gasolina	20
4	404	4	SNB132	Camión Mediano	3000.00	diesel	20

⬅

➡

🏠

🔍

📄

📖

Feedback

avance 3 > avance_03.sql



```
244
245     select * from dim_customer;
246     select * from dim_date;
247     select * from dim_driver;
248     select * from dim_route;
249     select * from dim_time;
250     select * from dim_vehicle;
251 | select * from fact_deliveries;
252
253
```

Results (just now)



Table Chart

🔍 📄 1,215 rows ⓘ 36ms 🔽 ⌚

#	# DELIVERY_KEY	# DATE_KEY	# SCHEDULED_TIME_KEY	# DELIVERED_TIME_KEY	# VEHICLE_KEY	# DRIVER_KEY
1	1	20260107	1430	1430	549	
2	2	20260107	1130	1100	471	
3	3	20260107	1030	1030	597	
4	4	20260107	1400	1400	563	

