

Inteligencia de Negocios

LABORATORIO N° 14

ARQUITECTURA DE DATAWAREHOUSE IV: CONSTELACIÓN DE ESTRELLAS



Alumno(s):	Cutipa Sivincha Maria del Carmen				Nota	
Grupo:	"B"		Ciclo: VI			
Criterio de Evaluación	Excelente (4pts)	Bueno (3pts)	Requiere mejora (2pts)	No acept. (0pts)	Puntaje Logrado	
Identifica los conceptos datawarehouse						
Conoce la estructura del modelo constelación de estrellas						
Identifica las tablas de dimensiones y hechos						
Conoce las etapas de un proceso ETL						
Implementa funciones para la carga de datos ETL						

TEMA: ARQUITECTURA DE DATAWAREHOUSE IV

OBJETIVOS

- Comprender los conceptos de la arquitectura de datawarehouse **constelación de estrellas**
- Comprender las etapas en el proceso ETL (Extracción, Transformación y Carga de datos)
- Implementa un datawarehouse de tipo **constelación de estrellas** y su respectivo ETL

REQUERIMIENTOS

- Leer las diapositivas del tema.

Actividades de desarrollo

1. Elabore un datawarehouse de tipo **constelación de estrellas** de una empresa dedicada a la producción y venta de muebles.

1. Código

```
import pandas as pd
from datetime import date

# Crear las tablas de dimensión como DataFrames de Pandas
def crear_tablas():
    # Tabla de Dimensión Producto
    dim_producto = pd.DataFrame({
        'id_producto': [1, 2, 3],
        'nombre': ["Mesa", "Silla", "Sofá"],
        'categoria': ["Mesa", "Silla", "Sofá"],
        'material': ["Madera", "Metal", "Cuero"],
        'precio': [1500.0, 500.0, 2000.0]
    })

    # Tabla de Dimensión Tiempo
    dim_tiempo = pd.DataFrame({
        'id_tiempo': [1, 2, 3],
```

```
'fecha': [date(2024, 11, 5), date(2024, 11, 6), date(2024, 11, 7)],  
  
'año': [2024, 2024, 2024],  
  
'mes': [11, 11, 11],  
  
'dia': [5, 6, 7],  
  
'trimestre': ["Q4", "Q4", "Q4"]  
  
})  
  
# Tabla de Dimensión Ubicación  
  
dim_ubicacion = pd.DataFrame({  
  
    'id_ubicacion': [1, 2, 3],  
  
    'pais': ["México", "México", "Estados Unidos"],  
  
    'ciudad': ["Ciudad de México", "Guadalajara", "Los Ángeles"],  
  
    'tienda': ["Tienda Centro", "Tienda Sur", "Tienda LA"]  
  
})  
  
# Tabla de Dimensión Empleado  
  
dim_empleado = pd.DataFrame({  
  
    'id_empleado': [1, 2, 3],  
  
    'nombre': ["Carlos", "Ana", "Luis"],  
  
    'puesto': ["Carpintero", "Vendedor", "Supervisor"]  
  
})  
  
# Tabla de Hecho Ventas  
  
hecho_ventas = pd.DataFrame({  
  
    'id_venta': [1, 2, 3],  
  
    'id_producto': [1, 2, 3],  
  
    'id_tiempo': [1, 2, 3],  
  
    'id_ubicacion': [1, 2, 3],  
  
    'cantidad': [5, 10, 2],
```

```
'precio_unitario': [1500.0, 500.0, 2000.0],
'ingresos_totales': [7500.0, 5000.0, 4000.0]
}))

# Tabla de Hecho Producción
hecho_produccion = pd.DataFrame({
    'id_produccion': [1, 2, 3],
    'id_producto': [1, 2, 3],
    'id_tiempo': [1, 2, 3],
    'id_empleado': [1, 2, 3],
    'cantidad_producida': [20, 15, 5],
    'costo_unitario': [1000.0, 300.0, 1500.0],
    'costo_total': [20000.0, 4500.0, 7500.0]
})

    return dim_producto, dim_tiempo, dim_ubicacion, dim_empleado,
hecho_ventas, hecho_produccion

# Guardar las tablas en un archivo Excel y mostrarlas en pantalla
def guardar_y_mostrar():
    # Crear Los DataFrames
    dim_producto, dim_tiempo, dim_ubicacion, dim_empleado, hecho_ventas,
hecho_produccion = crear_tablas()

    # Crear un archivo Excel con cada tabla en una hoja diferente
    with pd.ExcelWriter('datawarehouse_muebles.xlsx') as writer:
        dim_producto.to_excel(writer, sheet_name='DimProducto', index=False)
        dim_tiempo.to_excel(writer, sheet_name='DimTiempo', index=False)
```

```
        dim_ubicacion.to_excel(writer, sheet_name='DimUbicacion',
index=False)

        dim_empleado.to_excel(writer, sheet_name='DimEmpleado', index=False)

        hecho_ventas.to_excel(writer, sheet_name='HechoVentas', index=False)

        hecho_produccion.to_excel(writer, sheet_name='HechoProduccion',
index=False)

# Mostrar las tablas en pantalla

print("DimProducto:")
print(dim_producto)

print("\nDimTiempo:")
print(dim_tiempo)

print("\nDimUbicacion:")
print(dim_ubicacion)

print("\nDimEmpleado:")
print(dim_empleado)

print("\nHechoVentas:")
print(hecho_ventas)

print("\nHechoProduccion:")
print(hecho_produccion)

# Llamar a la función para crear, guardar y mostrar las tablas
guardar_y_mostrar()
```

2. Ejecución

```
PS C:\Users\USER\Documents\Proyectos\Inteligencia de negocios\S14> python datawarehouse_constelacion.py
```

```
DimProducto:
```

	id_producto	nombre	categoria	material	precio
0	1	Mesa	Mesa	Madera	1500.0
1	2	Silla	Silla	Metal	500.0
2	3	Sofá	Sofá	Cuero	2000.0

```
DimTiempo:
```

	id_tiempo	fecha	año	mes	día	trimestre
0	1	2024-11-05	2024	11	5	Q4
1	2	2024-11-06	2024	11	6	Q4
2	3	2024-11-07	2024	11	7	Q4

```
DimUbicacion:
```

	id_ubicacion	país	ciudad	tienda
0	1	México	Ciudad de México	Tienda Centro
1	2	México	Guadalajara	Tienda Sur
2	3	Estados Unidos	Los Ángeles	Tienda LA

```
DimEmpleado:
```

	id_empleado	nombre	puesto
0	1	Carlos	Carpintero
1	2	Ana	Vendedor
2	3	Luis	Supervisor

```
HechoVentas:
```

	id_venta	id_producto	id_tiempo	id_ubicacion	cantidad	precio_unitario	ingresos_totales
0	1	1	1	1	5	1500.0	7500.0
1	2	2	2	2	10	500.0	5000.0
2	3	3	3	3	2	2000.0	4000.0

```
HechoProduccion:
```

	id_produccion	id_producto	id_tiempo	id_empleado	cantidad_producida	costo_unitario	costo_total
0	1	1	1	1	20	1000.0	20000.0
1	2	2	2	2	15	300.0	4500.0
2	3	3	3	3	5	1500.0	7500.0

2. Implemente su proceso ETL de las dimensiones y hechos.

1. Código

```
import pandas as pd

from datetime import date

# Datos originales (extracción)

def extraer_datos():

    # Dimensión Producto

    dim_producto = pd.DataFrame({

        'id_producto': [1, 2, 3],

        'nombre': ["Mesa", "Silla", "Sofá"],

        'categoria': ["Mesa", "Silla", "Sofá"],

        'material': ["Madera", "Metal", "Cuero"],

        'precio': [1500.0, 500.0, 2000.0]

    })

    # Dimensión Tiempo

    dim_tiempo = pd.DataFrame({

        'id_tiempo': [1, 2, 3],

        'fecha': [date(2024, 11, 5), date(2024, 11, 6), date(2024, 11, 7)],

        'año': [2024, 2024, 2024],

        'mes': [11, 11, 11],

        'dia': [5, 6, 7],

        'trimestre': ["Q4", "Q4", "Q4"]

    })

    # Dimensión Ubicación

    dim_ubicacion = pd.DataFrame({

        'id_ubicacion': [1, 2, 3],

        'pais': ["México", "México", "Estados Unidos"],
```

```
'ciudad': ["Ciudad de México", "Guadalajara", "Los Ángeles"],  
'tienda': ["Tienda Centro", "Tienda Sur", "Tienda LA"]  
})  
  
# Dimensión Empleado  
  
dim_empleado = pd.DataFrame({  
    'id_empleado': [1, 2, 3],  
    'nombre': ["Carlos", "Ana", "Luis"],  
    'puesto': ["Carpintero", "Vendedor", "Supervisor"]  
})  
  
# Hecho Ventas  
  
hecho_ventas = pd.DataFrame({  
    'id_venta': [1, 2, 3],  
    'id_producto': [1, 2, 3],  
    'id_tiempo': [1, 2, 3],  
    'id_ubicacion': [1, 2, 3],  
    'cantidad': [5, 10, 2],  
    'precio_unitario': [1500.0, 500.0, 2000.0],  
    'ingresos_totales': [7500.0, 5000.0, 4000.0]  
})  
  
# Hecho Producción  
  
hecho_produccion = pd.DataFrame({  
    'id_produccion': [1, 2, 3],  
    'id_producto': [1, 2, 3],  
    'id_tiempo': [1, 2, 3],  
    'id_empleado': [1, 2, 3],  
    'cantidad_producida': [20, 15, 5],
```



```
'costo_unitario': [1000.0, 300.0, 1500.0],  
'costo_total': [20000.0, 4500.0, 7500.0]  
})  
  
    return dim_producto, dim_tiempo, dim_ubicacion, dim_empleado,  
hecho_ventas, hecho_produccion  
  
# Transformación de Los datos  
def transformar_datos(dim_producto, dim_tiempo, dim_ubicacion, dim_empleado,  
hecho_ventas, hecho_produccion):  
    # Ejemplo de transformación: incrementar el precio de cada producto un  
10%  
    dim_producto['precio'] = dim_producto['precio'] * 1.10  
  
    # Cambiar nombres de tiendas para incluir la cadena "Sucursal"  
    dim_ubicacion['tienda'] = "Sucursal " + dim_ubicacion['tienda']  
  
    # Cambiar el formato de fecha a cadena  
    dim_tiempo['fecha'] = dim_tiempo['fecha'].astype(str)  
  
    # Transformar ingreso total para mostrar un incremento de 5% por  
comisiones  
    hecho_ventas['ingresos_totales'] = hecho_ventas['ingresos_totales'] *  
1.05  
  
    # Actualizar costos de producción incrementando un 8%  
    hecho_produccion['costo_total'] = hecho_produccion['costo_total'] * 1.08
```

```
        return dim_producto, dim_tiempo, dim_ubicacion, dim_empleado,
hecho_ventas, hecho_produccion

# Carga de datos (simulada mostrando Los datos transformados)
def cargar_datos(dim_producto, dim_tiempo, dim_ubicacion, dim_empleado,
hecho_ventas, hecho_produccion):

    print("=== Datos Transformados ===")

    print("\nDimProducto (Transformado):")
    print(dim_producto)

    print("\nDimTiempo (Transformado):")
    print(dim_tiempo)

    print("\nDimUbicacion (Transformado):")
    print(dim_ubicacion)

    print("\nDimEmpleado (Transformado):")
    print(dim_empleado)

    print("\nHechoVentas (Transformado):")
    print(hecho_ventas)

    print("\nHechoProduccion (Transformado):")
    print(hecho_produccion)

# Proceso ETL completo
def proceso_etl():

    # Extracción

    dim_producto, dim_tiempo, dim_ubicacion, dim_empleado, hecho_ventas,
hecho_produccion = extraer_datos()

    # Mostrar Los datos antes de La transformación

    print("=== Datos Originales ===")

    print("\nDimProducto:")
```

```
print(dim_producto)

print("\nDimTiempo:")

print(dim_tiempo)

print("\nDimUbicacion:")

print(dim_ubicacion)

print("\nDimEmpleado:")

print(dim_empleado)

print("\nHechoVentas:")

print(hecho_ventas)

print("\nHechoProduccion:")

print(hecho_produccion)

# Transformación

    dim_producto, dim_tiempo, dim_ubicacion, dim_empleado, hecho_ventas,
hecho_produccion = transformar_datos(

    dim_producto, dim_tiempo, dim_ubicacion, dim_empleado, hecho_ventas,
hecho_produccion

)

# Carga (imprimir Los datos transformados)

    cargar_datos(dim_producto, dim_tiempo, dim_ubicacion, dim_empleado,
hecho_ventas, hecho_produccion)

# Ejecutar el proceso ETL

proceso_etl()
```

2. Ejecución

```
PS C:\Users\USER\Documents\Proyectos\Inteligencia de negocios\S14> python ETL.py
=== Datos Originales ===
```

DimProducto:

	id_producto	nombre	categoria	material	precio
0	1	Mesa	Mesa	Madera	1500.0
1	2	Silla	Silla	Metal	500.0
2	3	Sofá	Sofá	Cuero	2000.0

DimTiempo:

	id_tiempo	fecha	año	mes	dia	trimestre
0	1	2024-11-05	2024	11	5	Q4
1	2	2024-11-06	2024	11	6	Q4
2	3	2024-11-07	2024	11	7	Q4

DimUbicacion:

	id_ubicacion	pais	ciudad	tienda
0	1	México	Ciudad de México	Tienda Centro
1	2	México	Guadalajara	Tienda Sur
2	3	Estados Unidos	Los Ángeles	Tienda LA

DimEmpleado:

	id_empleado	nombre	puesto
0	1	Carlos	Carpintero
1	2	Ana	Vendedor
2	3	Luis	Supervisor

HechoVentas:

	id_venta	id_producto	id_tiempo	id_ubicacion	cantidad	precio_unitario	ingresos_totales
0	1	1	1	1	5	1500.0	7500.0
1	2	2	2	2	10	500.0	5000.0
2	3	3	3	3	2	2000.0	4000.0

HechoProduccion:

	id_produccion	id_producto	id_tiempo	id_empleado	cantidad_producida	costo_unitario	costo_total
0	1	1	1	1	20	1000.0	20000.0
1	2	2	2	2	15	300.0	4500.0
2	3	3	3	3	5	1500.0	7500.0

```
=== Datos Transformados ===
```

DimProducto (Transformado):

	id_producto	nombre	categoria	material	precio
0	1	Mesa	Mesa	Madera	1650.0
1	2	Silla	Silla	Metal	550.0
2	3	Sofá	Sofá	Cuero	2200.0

DimTiempo (Transformado):

	id_tiempo	fecha	año	mes	dia	trimestre
0	1	2024-11-05	2024	11	5	Q4
1	2	2024-11-06	2024	11	6	Q4
2	3	2024-11-07	2024	11	7	Q4

DimUbicacion (Transformado):

	id_ubicacion	pais	ciudad	tienda
0	1	México	Ciudad de México	Sucursal Tienda Centro
1	2	México	Guadalajara	Sucursal Tienda Sur
2	3	Estados Unidos	Los Ángeles	Sucursal Tienda LA

DimEmpleado (Transformado):

	id_empleado	nombre	puesto
0	1	Carlos	Carpintero
1	2	Ana	Vendedor
2	3	Luis	Supervisor

```
HechoVentas (Transformado):
id_venta id_producto id_tiempo id_ubicacion cantidad precio_unitario ingresos_totales
0 1 1 1 1 5 1500.0 7875.0
1 2 2 2 2 10 500.0 5250.0
2 3 3 3 3 2 2000.0 4200.0
```

```
HechoProduccion (Transformado):
0 1 Carlos Carpintero
1 2 Ana Vendedor
2 3 Luis Supervisor
```

```
HechoVentas (Transformado):
id_venta id_producto id_tiempo id_ubicacion cantidad precio_unitario ingresos_totales
0 1 1 1 1 5 1500.0 7875.0
0 1 Carlos Carpintero
1 2 Ana Vendedor
2 3 Luis Supervisor
```

```
0 1 Carlos Carpintero
0 1 Carlos Carpintero
1 2 Ana Vendedor
0 1 Carlos Carpintero
1 2 Ana Vendedor
0 1 Carlos Carpintero
0 1 Carlos Carpintero
1 2 Ana Vendedor
2 3 Luis Supervisor
```

```
HechoVentas (Transformado):
id_venta id_producto id_tiempo id_ubicacion cantidad precio_unitario ingresos_totales
0 1 1 1 1 5 1500.0 7875.0
1 2 2 2 2 10 500.0 5250.0
2 3 3 3 3 2 2000.0 4200.0
```

```
HechoProduccion (Transformado):
id_produccion id_producto id_tiempo id_empleado cantidad_producida costo_unitario costo_total
0 1 1 1 1 20 1000.0 21600.0
1 2 2 2 2 15 300.0 4860.0
2 3 3 3 3 5 1500.0 8100.0
```

Conclusiones:

Indicar las conclusiones que llegó después de los temas tratados de manera práctica en este laboratorio.

- La arquitectura de constelación de estrellas brinda una estructura flexible y detallada para analizar tanto las ventas como la producción de muebles, lo que permite examinar estos procesos desde múltiples perspectivas, como el producto, el tiempo, la ubicación y los empleados, facilitando así el análisis multidimensional.
- La modificación que incrementa el precio de los productos en un 10% ilustra cómo el sistema puede adaptarse a ajustes de precios en respuesta a decisiones estratégicas o fluctuaciones en el mercado, lo que proporciona una capacidad de análisis más precisa y adaptable.
- El incremento del 5% en los ingresos totales de ventas habilita el cálculo de métricas adicionales, como comisiones e incentivos, lo cual permite una visión integral de las estrategias de compensación y su impacto en los resultados comerciales.
- Al aplicar un ajuste del 8% en los costos de producción, se explora cómo las variaciones de costos afectan la rentabilidad del negocio, ofreciendo así una herramienta para la toma de decisiones informadas en términos de costos y eficiencia.
- Incorporar datos detallados de ubicación permite realizar un análisis exhaustivo del desempeño de cada tienda o centro de distribución, facilitando la identificación de áreas con mejores resultados o que requieran mejoras, y optimizando así las estrategias de venta y producción en función de la ubicación.