**Data Warehouse de Transporte - Guía Rápida**

**Resumen del Modelo**

Este es un **esquema estrella** (star schema) para analizar operaciones de transporte y logística. Tiene una tabla de hechos central rodeada de dimensiones que proporcionan contexto.

**Tabla de Hechos: F\_Shipment**

**¿Qué representa?** Cada fila es un envío completado.

**Métricas principales:**

* ingresos / costo\_total → Rentabilidad
* distancia\_km / peso\_kg / volumen\_m3 → Capacidad utilizada
* tiempo\_transito\_min / retraso\_min → Eficiencia operativa
* cantidad\_paradas / cantidad\_items → Complejidad del envío

**Claves:**

* shipment\_key (PK): ID único del DW
* shipment\_number: Número de guía (dimensión degenerada)
* 8 Foreign Keys a dimensiones

**Dimensiones**

**📅 D\_Fecha - La más importante**

**Propósito:** Análisis temporal

**Atributos clave:**

* Jerarquía: año → trimestre → mes → semana → día
* Descriptivos: dia\_semana, nombre\_mes
* Flags: es\_fin\_semana, es\_festivo

**Uso:** La tabla de hechos la referencia DOS veces:

* fecha\_salida\_key: Cuándo comenzó el envío
* fecha\_llegada\_key: Cuándo se completó

**Por qué es rica:** Permite análisis como "ventas por trimestre", "retrasos los lunes vs viernes", "estacionalidad"

**👥 D\_Cliente (con historial SCD Tipo 2)**

**Propósito:** ¿Quién contrata el servicio?

**Atributos:**

* segmento: Bronze, Silver, Gold, Platinum
* tipo\_cliente: Corporativo, PYME, Individual
* region, pais, industria

**SCD Tipo 2 explicado:** Cuando un cliente cambia de segmento (ej: Bronze → Gold), NO actualizamos el registro, creamos uno nuevo:

cliente\_key | codigo\_cliente | segmento | es\_actual | fecha\_inicio | fecha\_fin

1001 | CLI-001 | Bronze | FALSE | 2023-01-01 | 2024-02-29

1002 | CLI-001 | Gold | TRUE | 2024-03-01 | NULL

**Beneficio:** Los envíos antiguos mantienen el segmento que tenía el cliente EN ESE MOMENTO. Puedes responder: "¿Cuánto facturamos cuando estos clientes ERAN Bronze?"

**🚛 D\_Vehiculo (con historial SCD Tipo 2)**

**Propósito:** Características de la flota

**Atributos:**

* tipo\_vehiculo, marca, modelo
* capacidad\_kg, capacidad\_m3
* estado\_vehiculo: Operativo, Mantenimiento, Fuera de Servicio
* proveedor\_transporte: Propio vs Tercerizado

**Igual que Cliente:** Tracking de cambios (cambio de estado, cambio de proveedor)

**🛣️ D\_Ruta**

**Propósito:** Información de las rutas planificadas

**Atributos:**

* codigo\_ruta, nombre\_ruta (ej: "Buenos Aires - Córdoba")
* tipo\_ruta: Urbana, Interurbana, Internacional
* distancia\_planificada\_km, tiempo\_estimado\_min
* nivel\_dificultad, tiene\_peajes, cantidad\_peajes

**Análisis posible:** Comparar distancia planificada vs real, rutas más rentables, impacto de peajes

**📍 D\_Lugar**

**Propósito:** Geografía de origen y destino

**Jerarquía:** continente → pais → provincia\_estado → ciudad

**Atributos adicionales:**

* latitud, longitud (para mapas)
* zona\_horaria
* tipo\_ubicacion: Centro de Distribución, Puerto, Aeropuerto, Cliente

**Uso:** La tabla de hechos la referencia DOS veces:

* origen\_key: De dónde sale
* destino\_key: A dónde va

**Análisis posible:** Matriz origen-destino, corredores más rentables, drill-down de país a ciudad

**🏷️ D\_Junk - Dimensión especial**

**Propósito:** Consolidar múltiples atributos de baja cardinalidad en una sola dimensión

**Atributos consolidados:**

* tipo\_servicio: Express, Estándar, Económico
* estado\_envio: Completado, En Tránsito, Cancelado
* prioridad: Alta, Media, Baja
* tipo\_carga: Paquetería, Paletizado, Granel
* Flags: requiere\_refrigeracion, es\_fragil, requiere\_seguro

**¿Por qué junk?** En lugar de tener 7 dimensiones pequeñas (7 JOINs), tenemos 1 con todas las combinaciones pre-calculadas (~500 filas). Queries más rápidas, modelo más simple.

**Consultas Típicas**

**Rentabilidad por segmento:**

SELECT

c.segmento,

SUM(f.ingresos - f.costo\_total) as margen,

COUNT(\*) as envios

FROM F\_Shipment f

JOIN D\_Cliente c ON f.cliente\_key = c.cliente\_key

WHERE c.es\_actual = TRUE

GROUP BY c.segmento;

**Top rutas con retrasos:**

SELECT

r.nombre\_ruta,

AVG(f.retraso\_min) as retraso\_promedio

FROM F\_Shipment f

JOIN D\_Ruta r ON f.ruta\_key = r.ruta\_key

GROUP BY r.nombre\_ruta

ORDER BY retraso\_promedio DESC

LIMIT 10;

**Análisis temporal:**

SELECT

d.año,

d.mes,

d.dia\_semana,

COUNT(\*) as total\_envios,

AVG(f.ingresos) as ingreso\_promedio

FROM F\_Shipment f

JOIN D\_Fecha d ON f.fecha\_salida\_key = d.fecha\_key

GROUP BY d.año, d.mes, d.dia\_semana;

**Conceptos Clave Aplicados**

✅ **Esquema Estrella:** 1 tabla de hechos en el centro, dimensiones alrededor ✅ **Granularidad atómica:** Un registro = un envío (nivel más detallado) ✅ **Surrogate Keys:** IDs artificiales (no usamos códigos de negocio como PK) ✅ **SCD Tipo 2:** Para mantener historial de cambios en Cliente y Vehículo ✅ **Dimensión Junk:** Para consolidar flags y atributos de baja cardinalidad ✅ **Dimensión Degenerada:** shipment\_number vive en la tabla de hechos ✅ **Role-Playing Dimension:** D\_Fecha y D\_Lugar usadas múltiples veces con roles diferentes

**Beneficios del Diseño**

**Para Analistas:**

* Queries simples con pocos JOINs
* Fácil de entender (modelo intuitivo)
* Drill-down natural (año → mes → día, país → ciudad)

**Para el Negocio:**

* Respuestas rápidas a preguntas complejas
* Análisis histórico confiable
* Identificación de oportunidades y problemas

**Para TI:**

* Performance optimizado para lecturas
* Escalable (particionamiento por fecha)
* ETL predecible y automatizable

**Métricas Derivadas (Calculadas en Queries)**

No almacenamos estas métricas, las calculamos:

* **Margen:** ingresos - costo\_total
* **Margen %:** (margen / ingresos) \* 100
* **Desviación distancia:** distancia\_km - distancia\_planificada\_km
* **Utilización capacidad:** peso\_kg / capacidad\_vehiculo\_kg \* 100
* **Índice puntualidad:** COUNT(retraso\_min = 0) / COUNT(\*) \* 100

**Resumen Visual**

D\_Fecha

|

+--------------+--------------+

| |

(salida) (llegada)

| |

+-------- F\_SHIPMENT ---------+

| | |

D\_Cliente D\_Vehiculo D\_Ruta

| | |

D\_Lugar D\_Lugar D\_Junk

(origen) (destino)

**Estrella = Modelo simple pero poderoso**

erDiagram

F\_Shipment {

int shipment\_key PK

string shipment\_number "Dim Degenerada"

int fecha\_salida\_key FK

int fecha\_llegada\_key FK

int cliente\_key FK

int ruta\_key FK

int vehiculo\_key FK

int origen\_key FK

int destino\_key FK

int junk\_key FK

decimal ingresos

decimal costo\_total

decimal distancia\_km

decimal peso\_kg

decimal volumen\_m3

int tiempo\_transito\_min

int retraso\_min

int cantidad\_paradas

int cantidad\_items

}

D\_Fecha {

int fecha\_key PK

date fecha\_completa

int año

int trimestre

int mes

int semana

int dia\_mes

int dia\_año

string dia\_semana

string nombre\_mes

string trimestre\_desc

boolean es\_fin\_semana

boolean es\_festivo

string temporada

}

D\_Cliente {

int cliente\_key PK

int version

string codigo\_cliente

string nombre

string segmento

string tipo\_cliente

string region

string pais

string industria

date fecha\_alta

date fecha\_inicio\_vigencia

date fecha\_fin\_vigencia

boolean es\_actual

}

D\_Vehiculo {

int vehiculo\_key PK

int version

string matricula

string tipo\_vehiculo

string marca

string modelo

int año\_fabricacion

decimal capacidad\_kg

decimal capacidad\_m3

string estado\_vehiculo

string proveedor\_transporte

date fecha\_inicio\_vigencia

date fecha\_fin\_vigencia

boolean es\_actual

}

D\_Ruta {

int ruta\_key PK

string codigo\_ruta

string nombre\_ruta

string tipo\_ruta

decimal distancia\_planificada\_km

int tiempo\_estimado\_min

string nivel\_dificultad

boolean tiene\_peajes

int cantidad\_peajes

string zona\_geografica

}

D\_Lugar {

int lugar\_key PK

string codigo\_postal

string ciudad

string provincia\_estado

string pais

string region

string continente

decimal latitud

decimal longitud

string zona\_horaria

string tipo\_ubicacion

}

D\_Junk {

int junk\_key PK

string tipo\_servicio

string desc\_tipo\_servicio

string estado\_envio

string desc\_estado

string prioridad

string tipo\_carga

boolean requiere\_refrigeracion

boolean es\_fragil

boolean requiere\_seguro

}

F\_Shipment }o--|| D\_Fecha : "fecha\_salida"

F\_Shipment }o--|| D\_Fecha : "fecha\_llegada"

F\_Shipment }o--|| D\_Cliente : "cliente"

F\_Shipment }o--|| D\_Vehiculo : "vehiculo"

F\_Shipment }o--|| D\_Ruta : "ruta"

F\_Shipment }o--|| D\_Lugar : "origen"

F\_Shipment }o--|| D\_Lugar : "destino"

F\_Shipment }o--|| D\_Junk : "atributos\_transaccionales"