


# Introducción al Modelamiento Multidimensional (OLAP)

El modelamiento multidimensional es una técnica de diseño de estructuras de datos que permite organizar y consultar información desde múltiples ángulos simultáneamente. Esta metodología es ampliamente utilizada en sistemas OLAP (Online Analytical Processing) y es esencial en soluciones de inteligencia de negocios (BI).

 **por Kibernet Capacitación S.A.**

# ¿Qué es el modelamiento multidimensional?

A diferencia del modelo transaccional OLTP, orientado a la eficiencia de escritura y actualización, el enfoque multidimensional privilegia la lectura analítica: permite realizar consultas complejas, agregaciones, segmentaciones y comparaciones históricas de forma eficiente, incluso cuando se trata de millones de registros.

## Componentes principales del modelo

El modelo se construye a partir de los siguientes elementos:

- Tabla de hechos: Contiene medidas cuantitativas o KPIs del negocio (ej. total\_ventas, unidades\_vendidas).
- Dimensiones: Proveen el contexto del análisis, y están estructuradas jerárquicamente.

Jerarquía típica	Tiempo	Año → Trimestre → Mes → Día
	Producto	Categoría → Subcategoría → SKU
	Geografía	País → Región → Ciudad

Dato clave: Este diseño facilita el análisis tipo "slice and dice" (corte y cruce de datos), esencial en BI.

# Profundización: Ejemplo práctico de cubo OLAP

Supongamos que una empresa de retail quiere analizar las ventas de smartphones durante el primer trimestre de 2023, segmentadas por región.

Este tipo de consulta sería muy costosa y lenta en un modelo OLTP, pero en un modelo multidimensional se resuelve de forma inmediata gracias a la precarga de agregaciones y jerarquías en un cubo OLAP.

## Ejemplo en MDX (lenguaje de consulta para cubos):

El lenguaje MDX (Multidimensional Expressions) se utiliza para consultar cubos OLAP en entornos de inteligencia de negocios. Es el equivalente a SQL, pero para datos multidimensionales, permitiendo ejecutar operaciones como "slice and dice", drill-down, roll-up y más.

Este tipo de lenguaje es común en herramientas como:

- SQL Server Analysis Services (SSAS)
- Excel conectado a cubos OLAP
- Power BI (con modelos multidimensionales)

# Herramientas de BI que soportan conexiones a modelos multidimensionales

```
max analisis_ventas.mdx
1
2 -- Objetivo: Consultar el total de ventas de smartphones por región
3 -- en un cubo OLAP específico, filtrando por el primer trimestre del año 2023.
4 -- para el primer trimestre del año 2023, a partir del cubo 'Cubo_Ventas'
5 |
6 SELECT
7   -- Columna de salida: muestra la medida cuantitativa 'Total Ventas'
8   [Measures].[Total Ventas] ON COLUMNS,
9
10  -- Filas de salida: cada fila representa una región del cubo
11  -- Se obtienen todos los elementos 'hijo' de la jerarquía 'Región' dentro de la dimensión 'Geografía'
12  [Geografía].[Región].Children ON ROWS
13
14 FROM
15   -- Fuente de datos: el cubo OLAP que contiene los hechos y dimensiones relevantes
16   [Cubo_Ventas]
17
18 WHERE
19   (
20     -- Filtro 1: solo incluir productos de la categoría 'Smartphones'
21     [Producto].[Categoría].[Smartphones],
22
23     -- Filtro 2: limitar el análisis al primer trimestre del año 2023
24     [Tiempo].[Año].[2023].[Q1]
25   )
26
```

La imagen muestra un ejemplo práctico de una consulta MDX llamada analisis\_ventas.mdx, diseñada para:

- Mostrar el total de ventas por región.
- Filtrar los datos por la categoría de producto "Smartphones".
- Limitar el análisis al primer trimestre del año 2023.
- Utilizar comentarios paso a paso en forma de pseudocódigo, facilitando la comprensión del script para estudiantes.

REGIÓN	TOTAL VENTAS
Región Metropolitana	1,250,000
Valparaíso	745,000
Biobío	680,000
Antofagasta	510,000
Araucanía	395,000
Magallanes	120,000
TOTAL GENERAL	3,700,000

# Herramientas de implementación más comunes

A continuación, se presentan algunas de las herramientas más utilizadas en la implementación de modelos multidimensionales. Estas plataformas permiten diseñar, consultar y visualizar cubos OLAP, facilitando el análisis eficiente de grandes volúmenes de datos desde múltiples dimensiones.

Herramienta	Características destacadas
SQL Server Analysis Services	Crea cubos OLAP con lenguaje MDX o DAX.
Oracle OLAP	Usa esquemas estrella y facilita agregaciones automáticas.
Power BI / Tableau	Consumen modelos multidimensionales para visualización.

Estas herramientas permiten construir estructuras de análisis robustas para áreas como finanzas, ventas, logística o RRHH.

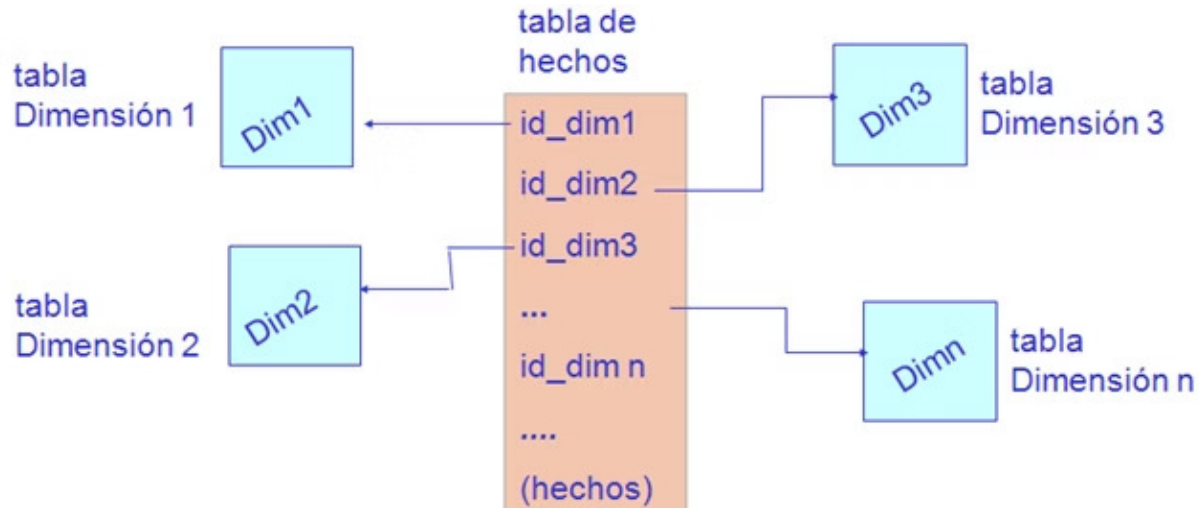
# Beneficios clave del modelamiento multidimensional

El modelamiento multidimensional ofrece ventajas significativas frente a otros enfoques de diseño de datos, especialmente en contextos donde el análisis ágil y la toma de decisiones son críticos. A continuación, se describen sus principales beneficios, los cuales derivan de su estructura orientada a hechos y dimensiones, su soporte para jerarquías y agregaciones, y su optimización para consultas analíticas.

Beneficio	Descripción
Agilidad	Consultas complejas se resuelven en segundos, sin afectar el sistema.
Flexibilidad	Permite combinar dimensiones libremente: tiempo, cliente, producto, etc.
Histórico	Ideal para analizar tendencias de datos durante años sin pérdida de rendimiento.

Estos beneficios son fundamentales para organizaciones que buscan tomar decisiones basadas en datos en entornos altamente competitivos.

# Esquema básico de un modelo multidimensional



Esta imagen representa la estructura lógica de un modelo multidimensional, en la que una tabla de hechos almacena métricas cuantitativas (como ventas o ingresos) y se conecta con múltiples tablas de dimensiones (como producto, tiempo o región) mediante claves foráneas.

Este diseño permite analizar la información desde distintos ángulos, facilitando consultas como:

- Total de ventas por región.
- Ventas de un producto específico en cierto periodo.
- Comparación de rendimiento entre canales de venta.

Observa la imagen y familiarízate con el clásico esquema estrella (star schema), una estructura fundamental en el modelado multidimensional que profundizaremos más adelante.

# Diferencias entre OLTP y OLAP

Para entender la relevancia del modelamiento multidimensional, es fundamental distinguirlo del enfoque tradicional OLTP (Online Transaction Processing), con el que coexiste en los sistemas empresariales.

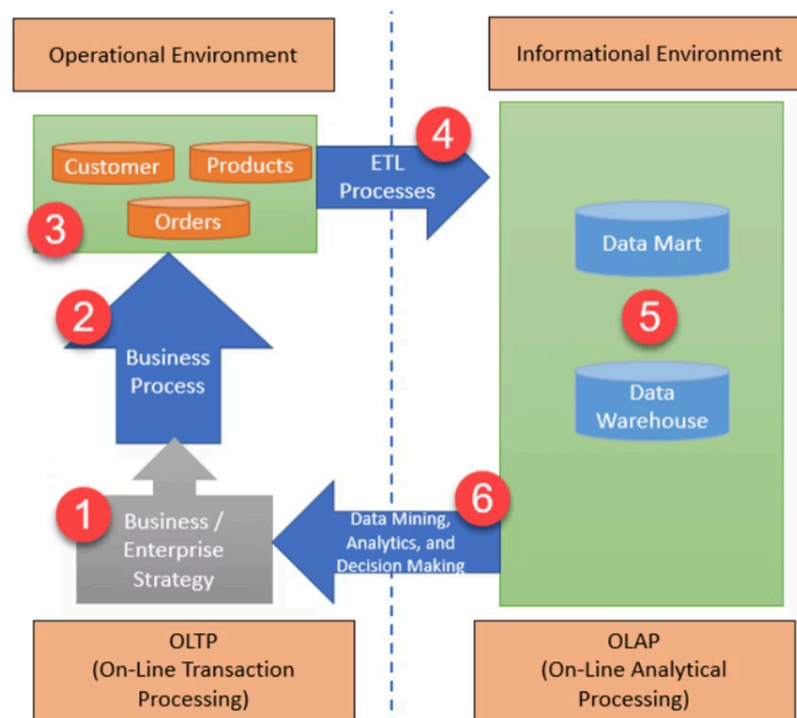
Mientras que OLTP se centra en la gestión eficiente de operaciones del día a día (como ventas, pagos, actualizaciones de inventario), OLAP está diseñado para el análisis estratégico de datos a partir de múltiples fuentes y períodos.

La siguiente tabla resume sus diferencias más importantes:

Característica	OLTP (Online Transaction Processing)	OLAP (Online Analytical Processing)
Objetivo	Registrar y procesar transacciones	Analizar grandes volúmenes de datos históricos
Modelo de datos	Altamente normalizado	Desnormalizado (modelo estrella o copo de nieve)
Tipo de consultas	Cortas, frecuentes, simples (ej: INSERT, UPDATE en SQL)	Largas, complejas, con agregaciones (ej: GROUP BY, JOIN en SQL / consultas MDX o fórmulas DAX)
Formato de consulta	SQL tradicional (.sql) ejecutado en Workbench, DBeaver, etc.	MDX (.mdx), DAX (.pbix), Excel pivot table, Power BI, SSAS
Tiempo de respuesta	Muy rápido para operaciones individuales	Optimizado para consultas agregadas, jerárquicas o por múltiples dimensiones
Usuarios principales	Operativos (cajeros, aplicaciones móviles, puntos de venta)	Analistas, gerentes, ejecutivos que generan reportes estratégicos
Ejemplo típico	Registrar la compra de un producto en una tienda en tiempo real	Consultar ventas trimestrales por categoría de producto y región
Actualización de datos	En tiempo real o casi inmediato (segundos)	Periódica (diaria, semanal, mensual) mediante procesos de carga (ETL)



# Análisis de la arquitectura OLTP y OLAP



Esta imagen refleja cómo OLTP y OLAP trabajan en conjunto, mostrando que el análisis estratégico (OLAP) nace a partir de los datos transaccionales (OLTP). Esta arquitectura es común en empresas que desean extraer valor de sus operaciones mediante inteligencia de negocios.

Analizando la imagen paso a paso:

1. Estrategia del negocio: Define los objetivos operativos y analíticos de la organización.
2. Procesos de negocio: Actividades diarias que generan transacciones (ventas, pedidos, pagos...).
3. Datos transaccionales (OLTP): Información operativa generada por los sistemas del entorno transaccional. Ejemplos: clientes, productos, órdenes.
4. Procesos ETL: Se extraen, transforman y cargan los datos desde OLTP hacia OLAP. Aquí se limpian, integran y organizan.
5. Entorno OLAP: Los datos llegan al Data Warehouse o Data Mart para ser consultados, agrupados y analizados.
6. Análisis y toma de decisiones: Herramientas de BI permiten realizar minería de datos, análisis comparativos y generar reportes gerenciales.

OLTP y OLAP no son tecnologías excluyentes: se complementan. En una arquitectura moderna de datos, OLTP es la fuente operativa que alimenta el sistema OLAP a través de procesos de integración y transformación (ETL), permitiendo separar la operación del análisis.

# Aproximaciones y enfoques en el modelamiento multidimensional

El modelamiento multidimensional puede abordarse desde diferentes estrategias de diseño, que definen el orden y la forma en que se estructura el entorno analítico dentro de una organización. Las más reconocidas y aplicadas en la industria son las aproximaciones propuestas por Ralph Kimball y Bill Inmon, dos referentes fundamentales en el mundo del Data Warehousing.

## Enfoque de Ralph Kimball (Bottom-up)

El enfoque de Kimball propone construir el almacén de datos desde abajo hacia arriba. Es decir, se comienza por desarrollar Data Marts temáticos (por ejemplo, uno para ventas, otro para inventario, etc.), cada uno con su propio modelo multidimensional. Estos luego se integran a través de una arquitectura común.

Características principales:

- Se basa en esquemas estrella o copo de nieve.
- Alta orientación al usuario de negocio.
- Tiempo de implementación más corto.
- Usa el concepto de conformed dimensions (dimensiones compartidas entre Data Marts).

Por ejemplo, una empresa de retail podría crear primero un cubo de ventas para su equipo comercial, luego uno de logística, y más adelante los integra bajo una visión corporativa común.

## Enfoque de Bill Inmon (Top-down)

Inmon plantea una estrategia más estructurada, donde se comienza por construir un Data Warehouse central altamente normalizado, que actúa como repositorio corporativo de datos históricos. Desde allí se derivan Data Marts que atienden necesidades analíticas específicas.

Características principales:

- Modelo de datos en forma normalizada (3NF) en la capa central.
- Mayor enfoque en calidad de datos y gobierno corporativo.
- Proceso de implementación más largo, pero con mayor control.
- Mejor alineación con procesos ETL complejos y sistemas heterogéneos.

Imagina como una empresa financiera centraliza toda su información histórica en un Data Warehouse corporativo. Desde allí, crea Data Marts para análisis de riesgos, comportamiento de clientes y desempeño por segmento.

# Comparación entre Kimball e Inmon

A continuación, se presentan las principales diferencias entre los enfoques de Kimball e Inmon, las cuales te permitirán comprender sus estrategias, ventajas y aplicaciones según el contexto organizacional.

Característica	Kimball (Bottom-up)	Inmon (Top-down)
Punto de partida	Data Marts individuales	Data Warehouse centralizado
Modelo inicial	Desnormalizado (estrella)	Normalizado (3NF)
Velocidad de implementación	Rápida (por fases)	Lenta (más estructurada)
Orientación	Usuario final / negocio	Gobierno de datos / IT corporativa
Ejemplo de modelo	Cubo de ventas → luego integrar	Almacén central → derivar cubos

# Componentes del modelado multidimensional

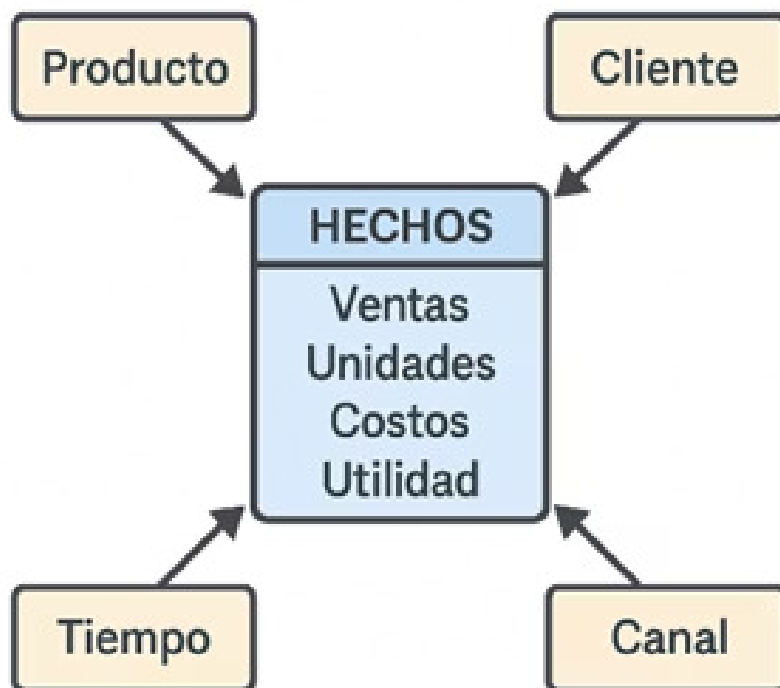
El modelado multidimensional se basa en una estructura clara y modular que permite organizar la información de forma que sea fácilmente consultable desde distintas perspectivas. Esta estructura se compone de los siguientes elementos clave:

## Hechos

Son los datos cuantitativos que queremos analizar. Generalmente, corresponden a indicadores de negocio o KPIs, como:

- Total de ventas
- Unidades vendidas
- Costos operativos
- Utilidad bruta

Estos datos se almacenan en la tabla de hechos (fact table), que actúa como el eje central del modelo.



## Dimensiones

Proveen el contexto de análisis de los hechos. Se almacenan en tablas de dimensiones, y permiten descomponer, agrupar y analizar los datos desde diferentes ángulos, tales como:

- Tiempo (año, mes, día)
- Producto (categoría, marca, SKU)
- Cliente (segmento, ubicación)
- Canal de venta (online, físico, mayorista)

Cada dimensión puede incluir jerarquías y atributos.



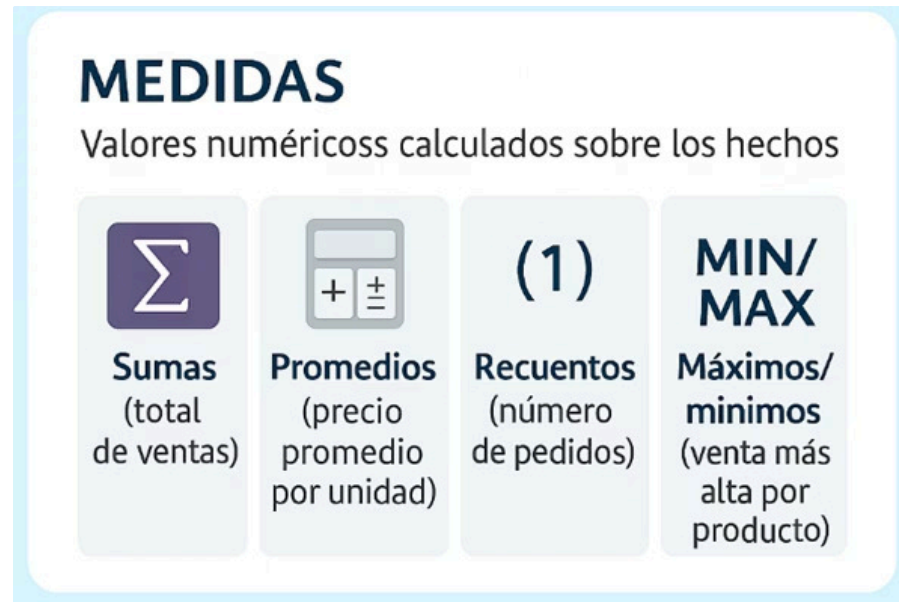
# Medidas y Jerarquías

## Medidas

Son los valores numéricos que se calculan sobre los hechos, como:

- Sumas (total de ventas)
- Promedios (precio promedio por unidad)
- Recuentos (número de pedidos)
- Máximos/mínimos (venta más alta por producto)

Estas medidas se utilizan en los cubos OLAP para generar reportes y dashboards.



## Jerarquías

Permiten navegar entre distintos niveles de detalle dentro de una dimensión. Son fundamentales para operaciones como drill-down (bajar al detalle) o roll-up (agrupar):

Ejemplo:

- Dimensión Tiempo: Año → Trimestre → Mes → Día
- Dimensión Geografía: País → Región → Ciudad

Las jerarquías definen el orden lógico de los niveles de análisis.



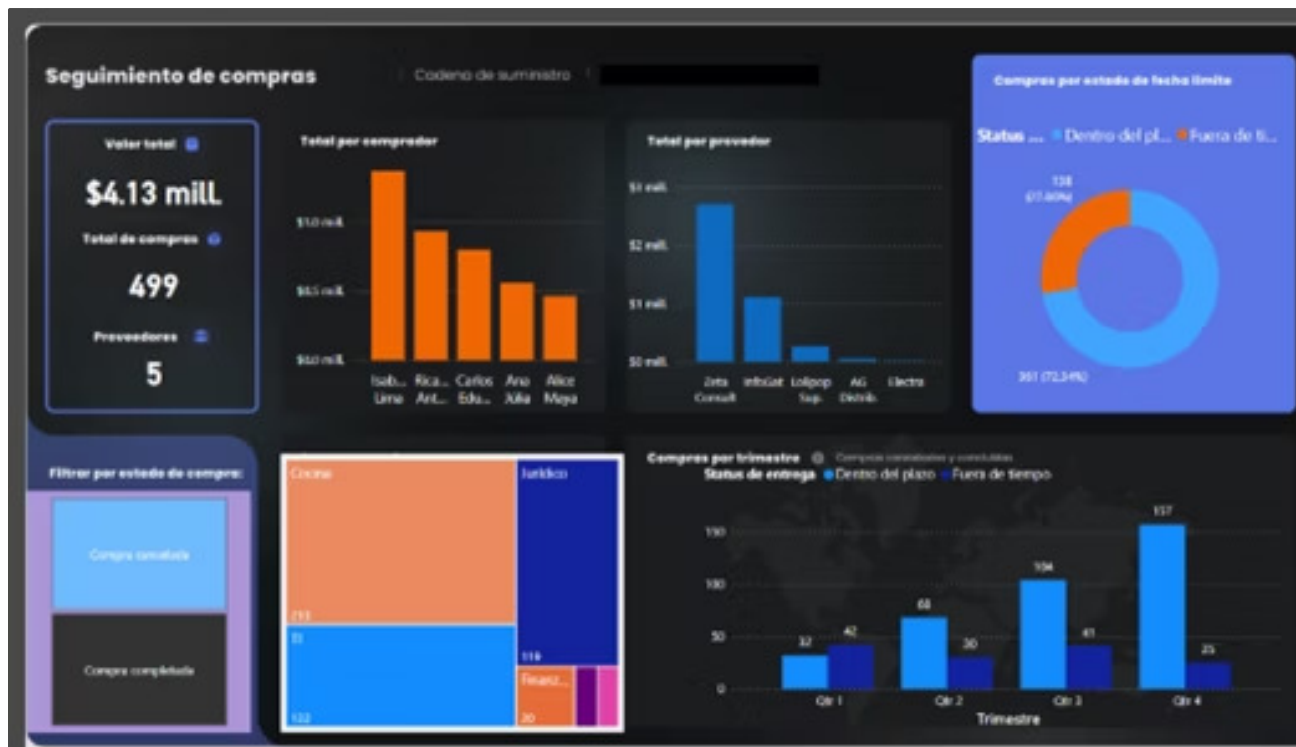
# Agregaciones

Son resúmenes de datos generados a partir de las medidas, organizados según los niveles de jerarquía. Gracias a ellas, se pueden realizar consultas rápidas sin tener que recorrer millones de registros.

Ejemplo:

- Total de ventas por mes (agregado desde el nivel de día).
- Promedio de ventas por región (agregado desde las ciudades).

Estas agregaciones se pre-calculan en los cubos para optimizar el rendimiento analítico.



Resumen de Métricas en un Dashboard: Las medidas como ventas, número de pedidos y proveedores se agregan por dimensiones como comprador, proveedor, trimestre y estado. Estas agregaciones permiten visualizar el comportamiento del negocio desde múltiples ángulos y niveles de detalle.

# Tipo de esquemas: Estrella y Copo de Nieve

Dentro del modelamiento multidimensional, la forma en que se estructuran las relaciones entre la tabla de hechos y las tablas de dimensiones da origen a distintos tipos de esquemas. Los más utilizados son el Esquema Estrella (Star Schema) y el Esquema Copo de Nieve (Snowflake Schema). Ambos cumplen la función de organizar la información para facilitar el análisis, pero lo hacen con diferentes grados de normalización y complejidad estructural, profundicemos un poco en cada uno de ellos.

## Esquema Estrella (Star Schema)

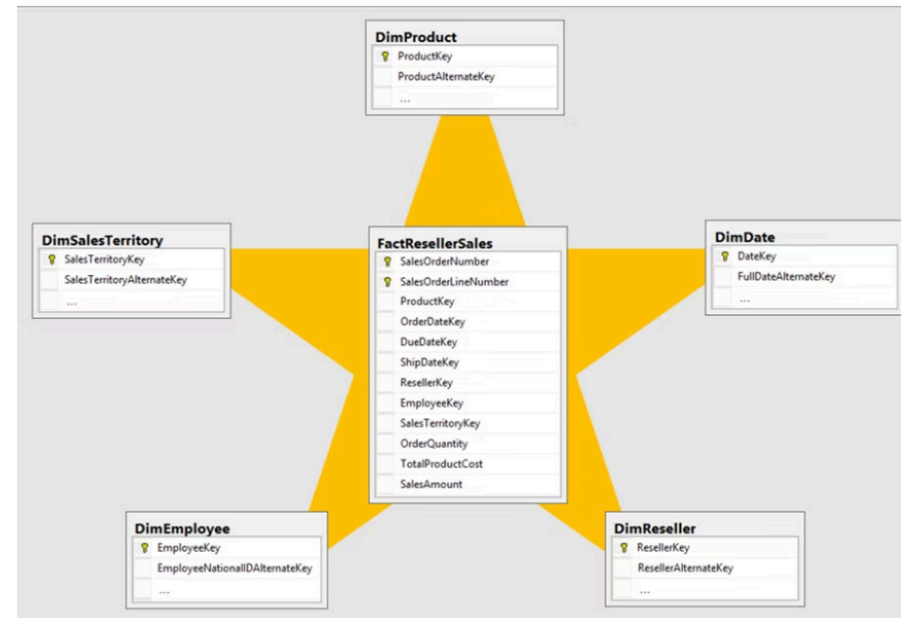
El esquema estrella es el más común y sencillo. Consiste en una tabla de hechos central que se conecta directamente con múltiples tablas de dimensiones, las cuales contienen descripciones de los contextos analíticos como tiempo, producto, cliente, etc.

Características principales:

- Las dimensiones están desnormalizadas: toda la información relevante está contenida en una sola tabla por dimensión.
- Las relaciones son simples y directas, lo que facilita la navegación y mejora el rendimiento de las consultas.
- Es muy utilizado en herramientas OLAP y BI debido a su simplicidad.

Ventajas:

- Consultas más rápidas.
- Estructura visualmente intuitiva.
- Fácil de entender para analistas de negocio.



Modelado de datos - Esquema estrella

# Esquema Copo de Nieve (Snowflake Schema)

El esquema copo de nieve es una extensión del esquema estrella en el que las dimensiones están parcialmente normalizadas, dividiéndose en sub-tablas relacionadas.

Por ejemplo, la dimensión Producto podría separarse en:

- Tabla Producto → con ID, nombre, categoría ID.
- Tabla Categoría → con ID, nombre de categoría.

Características principales:

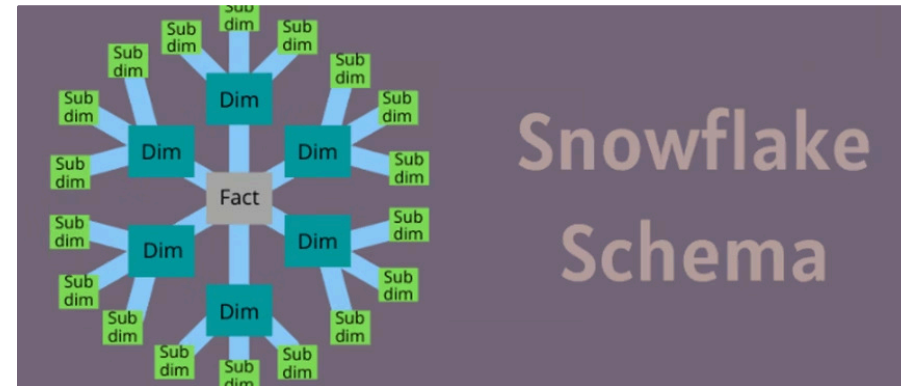
- Reducción de redundancia en datos de dimensiones.
- Estructura más normalizada y flexible para ciertos escenarios.
- Consultas más complejas, ya que requieren múltiples JOIN.

Ventajas:

- Menor duplicación de datos.
- Mantenimiento más eficiente en dimensiones jerárquicas grandes.

Desventajas:

- Consultas más lentas que en el esquema estrella.
- Mayor complejidad en los JOIN.



Modelado de datos - Esquema copo de nieve



# Comparación entre Esquema Estrella y Copo de Nieve

Aspecto	Esquema Estrella	Esquema Copo de Nieve
Estructura	Simple, desnormalizada	Compleja, parcialmente normalizada
Rendimiento de consulta	Alto (menos JOIN)	Medio o bajo (más JOIN)
Facilidad de mantenimiento	Menor (por redundancia)	Mayor (mejor integridad de datos)
Uso en herramientas BI	Muy compatible	Compatible, pero puede requerir más configuración
Facilidad de comprensión	Alta	Media o baja para usuarios no técnicos

Ambos esquemas tienen su lugar en el diseño de soluciones OLAP. El esquema estrella es ideal para consultas rápidas y simples, mientras que el copo de nieve es preferido cuando se prioriza la integridad y consistencia de los datos en entornos más complejos.

En muchos proyectos reales, se utiliza una combinación de ambos: dimensiones más simples en estrella, y otras más complejas en copo de nieve, según las necesidades del negocio.

# Tablas de hechos

En el modelamiento multidimensional, la tabla de hechos es el componente central del esquema (ya sea estrella o copo de nieve). Representa los eventos medibles del negocio y se enlaza con todas las dimensiones relevantes mediante claves foráneas.

## ¿Qué contiene una tabla de hechos?

- Métricas o medidas numéricas del negocio, como:
  - Total de ventas
  - Cantidad de unidades vendidas
  - Costos o ingresos
  - Tiempo invertido, margen, beneficio neto, etc.
- Claves foráneas (FK) que conectan con las dimensiones asociadas:
  - Producto
  - Cliente
  - Tiempo
  - Región / canal / empleado (según el caso)

Cada fila en la tabla de hechos representa un evento o transacción específica, como una venta individual, un pedido, un clic o una interacción de usuario.

# Ejemplo básico en SQL para tabla de hechos

A continuación, se muestra un ejemplo de cómo crear, mediante SQL, una tabla de hechos llamada ventas, elemento central en un modelo multidimensional. Esta tabla está diseñada para registrar eventos medibles del negocio y establecer conexiones con distintas dimensiones analíticas.

ventas.sql U X

modulo-5 > sesion-5 > ventas.sql

```
1 CREATE TABLE ventas (  
2     id_venta SERIAL PRIMARY KEY,  
3     id_producto INT,  
4     id_cliente INT,  
5     id_tiempo INT,  
6     total_venta DECIMAL  
7 );  
8
```

Podemos notar en esta consulta que:

- id\_venta es el identificador único del hecho.
- Los campos id\_producto, id\_cliente, id\_tiempo son claves foráneas a las respectivas dimensiones.
- total\_venta es la medida que se desea analizar.

## Tipos de hechos más comunes

Existen diferentes tipos de hechos según cómo se comportan al ser agregados a través de las dimensiones. La siguiente tabla resume los más comunes, junto con su descripción y ejemplos típicos en entornos analíticos.

Tipo de hecho	Descripción	Ejemplo
Hechos aditivos	Pueden ser sumados en todas las dimensiones.	Total de ventas, unidades vendidas
Hechos semiaditivos	Se pueden sumar en algunas dimensiones, pero no en otras.	Saldos, inventario
Hechos no aditivos	No se pueden sumar directamente. Requieren agregaciones específicas.	Porcentajes, ratios, tasas

# Buenas prácticas al definir tablas de hechos

- Utilizar claves numéricas enteras como FK para optimizar el rendimiento.
- Asegurar la calidad de los datos agregados (evitar duplicados o errores de conteo).
- Determinar claramente la granularidad del hecho:
  - ¿Cada fila representa una venta?
  - ¿Una línea de pedido?
  - ¿Una visita al sitio?
- Incluir columnas de fecha de carga y fuente del dato para trazabilidad y auditoría.

## Relación con dimensiones

La tabla de hechos no tiene valor por sí sola: solo al estar conectada con dimensiones permite análisis como:

- ¿Cuánto vendimos en enero de 2023 por canal?
- ¿Cuál fue la utilidad por cliente VIP en la región norte?
- ¿Qué producto tuvo mayor volumen de venta en el último trimestre?

Estas preguntas se responden al navegar por las dimensiones y agregar medidas desde la tabla de hechos.

El diseño correcto de la tabla de hechos es fundamental para la eficiencia analítica. Una mala definición (por ejemplo, sin clave única, con medidas ambiguas o duplicadas) puede generar errores en los resultados, dashboards inexactos y decisiones mal fundamentadas.

En términos prácticos, la tabla de hechos es el "corazón" del modelo OLAP, y su diseño define el alcance del análisis posible.

# Tablas de dimensiones

Las tablas de dimensiones son componentes fundamentales del modelo multidimensional, ya que proporcionan el contexto necesario para interpretar y analizar los hechos. Si la tabla de hechos responde a la pregunta "¿cuánto?", las dimensiones responden a "¿cuándo?", "¿dónde?", "¿qué producto?", "¿para qué cliente?", etc.

## ¿Qué contienen las tablas de dimensiones?

Las tablas de dimensiones están compuestas por:

- Una clave primaria (PK) que se conecta con la tabla de hechos como clave foránea.
- Un conjunto de atributos descriptivos que permiten filtrar, agrupar o clasificar los hechos.

En muchos casos, jerarquías naturales que organizan la información en niveles (por ejemplo: país → región → ciudad).

Las dimensiones permiten aplicar operaciones OLAP como slice, dice, drill-down, roll-up y pivot.

# Ejemplo básico en SQL para dimensiones

A continuación, se presenta un ejemplo básico de cómo definir una tabla de dimensión en SQL. En este caso, se modela la dimensión producto, con sus atributos principales:

- id\_producto es la clave primaria de la dimensión.
- nombre es un atributo descriptivo (ej. "Notebook HP 15").
- categoria permite agrupar productos en niveles superiores (ej. "Electrónica").

## Ejemplos de dimensiones comunes y sus atributos

Estas son algunas de las dimensiones más comunes en un modelo multidimensional, junto con ejemplos de los atributos que suelen contener:

Dimensión	Ejemplos de atributos
Tiempo	Año, trimestre, mes, semana, día
Producto	Nombre, categoría, marca, precio base
Cliente	Nombre, edad, género, segmento, ubicación
Geografía	País, región, ciudad, zona comercial
Canal	Tipo de canal, canal físico/online, distribuidor asociado

# Estructura y vinculación de dimensiones

## Estructura general de una tabla de dimensión

- Tiene una fila por entidad única (un cliente, un producto, un día).
- Generalmente, su número de registros es menor que en la tabla de hechos.
- Puede incluir columnas derivadas o enriquecidas con información externa (por ejemplo: cliente + historial de compras).

## ¿Cómo se vinculan con la tabla de hechos?

Cada dimensión se enlaza con la tabla de hechos a través de su clave primaria, permitiendo acceder a información contextual.

```
ventas.sql U X
modulo-5 > sesion-5 > ventas.sql
1  SELECT
2      f.total_venta,
3      p.nombre AS producto,
4      p.categoria,
5      c.nombre AS cliente,
6      t.mes
7  FROM
8      ventas f
9  JOIN producto p ON f.id_producto = p.id_producto
10 JOIN cliente c ON f.id_cliente = c.id_cliente
11 JOIN tiempo t ON f.id_tiempo = t.id_tiempo;
12
```

Este tipo de consulta permite analizar ventas combinando múltiples dimensiones, como producto, cliente y fecha.

Output o simulación de salida de datos al ejecutar la consulta de la

total_venta	producto	categoria	cliente	mes
125000	Notebook HP 15	Electrónica	Camila Rojas	Enero
82000	Celular Galaxy A10	Electrónica	Juan Soto	Enero
53000	Silla Ergonómica	Muebles	Carla Fuentes	Febrero
210000	Smart TV LG 50"	Electrónica	Camila Rojas	Marzo
43000	Escritorio Roble	Muebles	Juan Soto	Marzo

# Buenas prácticas al diseñar tablas de dimensiones

- Usar claves enteras numéricas como identificadores.
- Evitar ambigüedad en los atributos (por ejemplo, evitar usar "nombre" sin contexto).
- Incluir jerarquías para facilitar el análisis agregado.
- Permitir que los nombres de atributos sean legibles y consistentes.
- Considerar campos adicionales para etiquetado, segmentación y filtrado.

Las dimensiones son la puerta de entrada al análisis multidimensional. Son las que permiten a los usuarios del negocio entender el contexto y explorar los datos desde diferentes ángulos. Un modelo puede tener múltiples dimensiones, pero deben ser bien diseñadas para que las consultas sean claras, rápidas y útiles.

En resumen: los hechos dicen qué pasó, las dimensiones explican cómo, cuándo, dónde y con quién.



# Jerarquías y agregaciones

En el modelamiento multidimensional, las jerarquías y agregaciones permiten organizar los datos de manera lógica y flexible para facilitar su análisis a distintos niveles de detalle. Estas estructuras son esenciales para realizar operaciones OLAP como drill-down (descender al detalle) y roll-up (agrupar en niveles superiores).

## ¿Qué es una jerarquía?

Una jerarquía es una estructura de niveles dentro de una dimensión, que representa una relación de dependencia o agregación natural entre sus atributos. Permite navegar de forma progresiva desde un nivel general hacia uno más específico, o viceversa.

Ejemplo de jerarquía en la dimensión Tiempo:

- Año
  - └── Mes
- - └── Día

Ejemplo en la dimensión Geografía:

- País
  - └── Región
- - └── Ciudad

Estas jerarquías son clave para generar reportes que permitan cambiar el nivel de análisis sin rediseñar la consulta.

## ¿Qué es una agregación?

Una agregación es un resumen estadístico de los datos en la tabla de hechos, calculado con base en los niveles definidos por las jerarquías. Las operaciones de agregación más comunes incluyen:

- Suma (ej: total de ventas por mes)
- Promedio (ej: ventas promedio por producto)
- Conteo (ej: número de transacciones por cliente)
- Máximo/Mínimo (ej: valor máximo de venta en un trimestre)

Sin jerarquías, estas agregaciones serían planas y no escalables; con jerarquías, se transforman en análisis multicapas.

Jerarquía: Tiempo → Año → Mes → Día

# Consulta agregada y su importancia

Mostrar el total de ventas por mes y luego por año:

ventas.sql U X

modulo-5 > sesion-5 > ventas.sql

```
1  SELECT
2    EXTRACT(YEAR FROM fecha) AS anio,
3    EXTRACT(MONTH FROM fecha) AS mes,
4    SUM(total_venta) AS total_mensual
5  FROM ventas
6  GROUP BY anio, mes
7  ORDER BY anio, mes;
8
9
```

Este tipo de agregación es lo que permite analizar tendencias temporales, estacionalidades o acumulados históricos con facilidad.

Output o simulación de salida de datos al ejecutar la consulta de la imagen 16

anio	mes	total_mensual
2023	1	215000
2023	2	173000
2023	3	289500
2024	1	245300
2024	2	198000
2024	3	223750

Este tipo de consulta permite obtener totales mensuales a partir de la tabla de hechos, apoyándose en la jerarquía de la dimensión tiempo. Es una operación típica en entornos OLAP para realizar análisis históricos y comparativos.

## ¿Por qué son importantes las jerarquías y agregaciones?

Beneficio	Descripción breve
Navegación flexible	Permiten cambiar entre niveles de análisis sin redefinir consultas.
Comparación por niveles	Facilitan comparaciones jerárquicas (ej: región vs. ciudad).
Optimización de rendimiento	Muchas herramientas OLAP pre-calculan agregaciones para acelerar el análisis.
Soporte a decisiones	Entregan insights claros tanto a nivel macro como micro.

# Buenas prácticas al definir jerarquías

- Asegúrate de que los niveles sean lógicos y consistentes.
- Evita niveles redundantes o poco informativos.
- Define jerarquías explícitamente en las herramientas OLAP (SSAS, Power BI, etc.).
- Utiliza nombres descriptivos para facilitar la navegación de los usuarios.

Las jerarquías y agregaciones son las llaves para escalar el análisis: permiten observar un fenómeno desde lo general hasta lo específico. Diseñarlas bien es lo que convierte a un modelo multidimensional en una herramienta útil para la exploración, comparación y toma de decisiones.

En la práctica, estas agregaciones se construyen mediante funciones como SUM() o AVG() aplicadas sobre medidas en la tabla de hechos, agrupadas por niveles jerárquicos. Así, una jerarquía bien diseñada potencia el uso de consultas SQL para responder preguntas complejas con una simple línea de código.

En resumen: las jerarquías ordenan, las agregaciones resumen, y juntas empoderan el análisis.

# Diseño y desarrollo de modelos multidimensionales

Diseñar un buen modelo multidimensional no se trata solo de unir tablas: es crear una estructura que hable el lenguaje del negocio, que responda con agilidad a las preguntas clave de quienes toman decisiones y que sea comprensible para quienes analizan los datos día a día. Este proceso requiere combinar técnica, claridad y empatía hacia el usuario final. Un modelo bien diseñado no solo mejora el rendimiento de las consultas, sino que también facilita el análisis exploratorio, evita ambigüedades y promueve una toma de decisiones basada en evidencia.

## Buenas prácticas recomendadas:

- Incluir solo las dimensiones necesarias: evita sobrecargar el modelo con información que no será usada.
- Crear jerarquías explícitas y bien definidas: permiten navegar entre niveles de análisis con claridad (por ejemplo: Año → Mes → Día).
- Usar esquemas estrella cuando el rendimiento sea prioritario: simplifican las consultas y mejoran la velocidad en herramientas de BI.
- Documentar todas las relaciones y claves: facilita la comprensión del modelo y su mantenimiento a futuro.
- Validar el modelo con los usuarios de negocio: asegúrate de que las dimensiones y medidas reflejen cómo realmente se analizan los datos en la organización.

# Resumen de conceptos clave



## Modelamiento multidimensional

Técnica de diseño de datos que organiza la información en hechos y dimensiones para facilitar el análisis desde múltiples perspectivas.



## Tabla de hechos

Contiene las métricas cuantitativas del negocio y se conecta con las dimensiones mediante claves foráneas.



## Dimensiones

Proporcionan el contexto para analizar los hechos, permitiendo filtrar, agrupar y segmentar la información.



## Jerarquías y agregaciones

Permiten navegar entre diferentes niveles de detalle y resumir datos para análisis más eficientes.

# Conclusión

A lo largo de esta sesión, exploramos los fundamentos del modelamiento multidimensional, comprendiendo cómo las tablas de hechos, dimensiones, jerarquías y agregaciones trabajan en conjunto para construir modelos analíticos potentes, escalables y orientados al negocio. Analizamos diferentes tipos de esquemas, entendimos la importancia de un buen diseño y descubrimos cómo transformar datos complejos en información valiosa para la toma de decisiones.

Este conocimiento representa un paso esencial para cualquier ingeniero o ingeniera de datos que trabaje con soluciones de inteligencia de negocios, ya que permite estructurar los datos de manera que realmente generen valor para las organizaciones.