



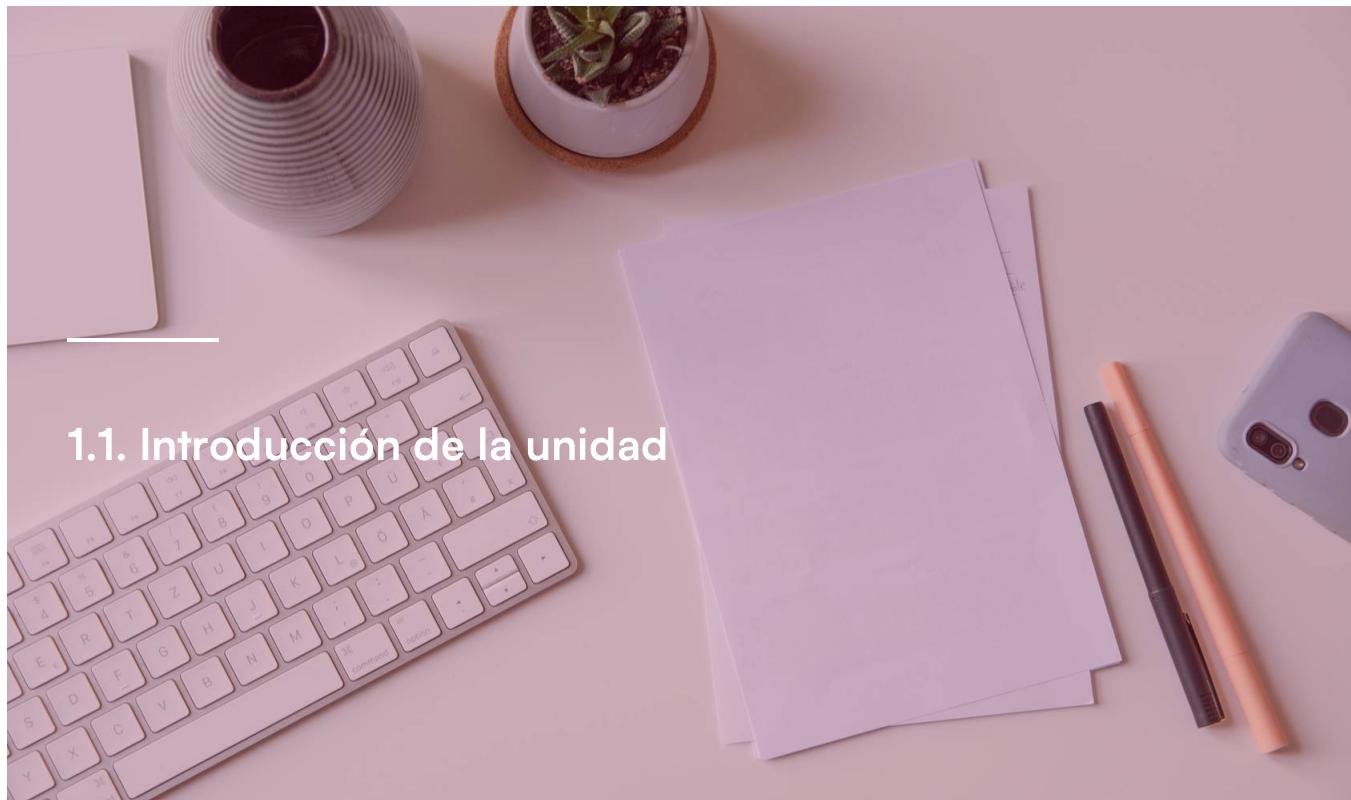
Almacenes de datos y bases de datos analíticas



- ☰ I. Introducción
- ☰ II. Objetivos
- ☰ III. Almacenes de datos
- ☰ IV. Herramientas de análisis de un almacén de datos: OLAP
- ☰ V. Multidimensionalidad y el modelo multidimensional
- ☰ VI. Desnormalización
- ☰ VII. Lenguajes de consulta analíticos: MDX
- ☰ VIII. Resumen
- ☰ IX. Caso práctico con solución

≡ X. Glosario

I. Introducción



1.1. Introducción de la unidad

Tras ver una introducción a la inteligencia de negocio, en esta unidad se va a profundizar en los almacenes de datos que entran en juego en los sistemas analíticos que sirven como sustento para las herramientas de *business intelligence* que funcionan como interfaz para los usuarios. Es decir, se va a ahondar en los sistemas de *back-end* que entran en juego en un entorno de inteligencia de negocio.

Los sistemas de bases de datos tradicionales y transaccionales han constituido la columna vertebral de las organizaciones y aplicaciones de negocio durante más de 20 años. Sin embargo, el incremento progresivo de la volumetría de datos en la organización, así como los actuales escenarios interconectados y los nuevos

requisitos de análisis de información, ha favorecido el desarrollo de nuevas tecnologías, con nuevos modos de almacenamiento, orientadas a dar cobertura a estas necesidades.

Como ya se vio en la unidad anterior de este módulo, los almacenes de datos son una pieza clave en las soluciones de inteligencia de negocio y su aparición supuso una verdadera revolución para la analítica empresarial. Por ello se detallará el concepto de almacén de datos, cuándo y cómo usarlo, así como las características esenciales de cada uno de ellos.

Se presentará el concepto de *data warehouse* más en profundidad para poder conocer el fundamento de las bases de datos analíticas y el papel esencial que juegan como motor del procesamiento analítico para las herramientas de visualización que harán de interfaz con el usuario.

Los almacenes analíticos de datos suelen contar con un carácter multidimensional dentro de su arquitectura. Esto significa que se normalizan los datos para tener una tabla central de hechos, y un grupo de tablas satélite alrededor componiendo las dimensiones por las cuales se van a analizar los hechos. También se detallarán cómo y cuáles son sus lenguajes de consulta y las herramientas que permiten realizar el análisis de los datos que contienen.

Dentro de la presentación de este tipo de repositorios de información se procederá a identificar y exponer las ventajas de las bases de datos analíticas frente a modelos clásicos anteriores.

Se presentarán herramientas y motores de análisis de información a partir de los modelos multidimensionales OLAP, así como las tipologías de las que constan.

Igual que existe el estándar SQL para consultar y operar con las bases de datos relacionales, los almacenamientos de tipo OLAP interactúan mediante un lenguaje denominado MDX (*MultiDimensional eXpressions*). Este es el lenguaje de consulta para bases de datos multidimensionales.

II. Objetivos



2.1. Objetivos de la unidad

- 1 Entender el concepto de almacenes de datos, por qué surgen y sus características esenciales.
- 2 Conocer el fundamento de las bases de datos analíticas y su papel esencial como motor del procesamiento analítico.
- 3 Comprender el carácter multidimensional de su arquitectura, así como de sus lenguajes de consulta y herramientas de análisis.
- 4 Identificar y comprender las ventajas de las bases de datos analíticas frente a modelos clásicos.

5

Conocer las herramientas y motores de análisis de información a partir de los modelos multidimensionales OLAP e identificar los distintos tipos.

6

Conocer MDX como lenguaje analítico de consulta de información en un modelo multidimensional.

III. Almacenes de datos

Introducción 1

A medida que se desarrollan las redes de comunicación, los modos de almacenar los datos, la capacidad de transporte de las redes, y la naturaleza de los medios de comunicación (la adaptación de los medios tradicionales –radio, prensa, tv– a formatos de nueva generación – páginas web, redes sociales y, en general, formatos digitales–), se establece un nuevo ecosistema de datos.

Introducción 2

En esta nueva situación, a las empresas no solo les resulta más fácil y eficiente la digitalización de todos los datos que generan, sino también estar interconectadas a otros orígenes externos de información que, a priori, son interesante para integrar sus sistemas de información para minimizar la curva de entropía de sus sistemas.

Introducción 3

El resultado es que, en las organizaciones, se generan grandes cantidades de datos constantemente, que son almacenados en algún sistema de información con múltiples fuentes de datos, internas o externas a la empresa. Esta multiplicidad de fuentes de distinta naturaleza (documentos de texto, bases de datos, etc.) requiere que se puedan interconectar entre ellas mediante algún mecanismo único, ya que los datos existentes pueden estar almacenados en diferentes plataformas, distintos formatos, lenguajes de acceso o consultas, sistemas *hardware* y *software*, etc.

Introducción 4

Por ello, las organizaciones necesitan la información contenida en sus datos, los datos que generan sus procesos de negocio, así como otras potenciales fuentes externas. Sin embargo, la marcada especialización de los sistemas en una organización provoca que, en la mayoría de los casos, se disponga de varios sistemas de información que gestionan distintos procesos de negocio, lo que implica redundancia y duplicidad de datos en diferentes sistemas, uso de distintas bases de datos y desconexión entre ellas, lo que genera problemas de unicidad de la información, comprobación, validación e, incluso, localización.

Introducción 5

La pregunta es obvia: ¿qué pueden hacer las organizaciones para dar respuesta a estas necesidades y a este nuevo contexto? La solución viene con el *data warehouse*.

Introducción 6

Para una empresa, tener un simple almacén de datos para realizar los procesos pertinentes no es suficiente, ya que lo que realmente necesita es información para formar un conocimiento que le permita tomar decisiones de negocio oportunas.

“La aparición de los *Data Warehouses* o almacenes de datos son la respuesta a las necesidades de los usuarios que necesitan información consistente, integrada, histórica y preparada para ser analizada para poder tomar decisiones. Al recuperar la información de los distintos sistemas, tanto transaccionales como departamentales o externos, y almacenándolos en un entorno integrado de información diseñado por los usuarios, el *Data Warehouses* nos permitirá analizar la información contextualmente y relacionada dentro de la organización.¹”

– J. L. Cano

¹Cano, J. L. *Business intelligence: competir con información*. Banesto, Fundación Cultural; 2007.

En definitiva, un *data warehouse*, o almacén de datos, es una base de datos que almacena de forma especialmente estructurada datos de las organizaciones para proporcionar información y conocimiento, respondiendo a cualquier pregunta compleja relacionada con los datos que almacena.

La principal funcionalidad de un *data warehouse* es facilitar un sistema que provea una **versión única de la verdad** para toda la organización. Esta información permite realizar una **toma de decisiones estratégicas** basadas en datos verídicos. También permite, a su vez, el análisis de tendencias, patrones y correlaciones de los datos. El *data warehouse* es un **almacén de datos históricos y consolidados** de varias fuentes, que simplifica el desarrollo de herramientas analíticas de los entornos que las adoptan.

3.1. Origen de los almacenes de datos

i Las *killer queries* fueron y siguen siendo el motivo de la creación de los almacenes de datos o *data warehouses*. Estas consultas no son más que consultas muy “pesadas” a las bases de datos operacionales de las empresas, que tienen como objetivo crear informes o análisis. Sin embargo, estas consultas pueden llegar a empeorar el rendimiento de la base de datos.

Las empresas y organizaciones necesitan realizar continuas consultas para poder acceder a los datos, explorarlos, analizarlos y, de esta forma, tomar las decisiones pertinentes. Si continuamente, y de forma excesiva, se consultan los datos históricos, es decir, se realizan estas *killer queries* a la base de datos, el sistema operacional de la empresa puede dejar de funcionar.

Por esto, surge la necesidad de englobar los datos en un entorno que no afecte a la operativa y actividad de la empresa y que de alguna forma evite estas consultas.

Así se crearon los almacenes de datos como herramienta de las soluciones de inteligencia de negocio y como pieza fundamental en la planificación y el desarrollo de las empresas.

Otro motivo clave es el poder combinar diversas fuentes de datos en un único repositorio para así comparar información de estas fuentes.

Todo esto permite tener un almacén centralizado de datos que facilite la toma de decisiones de la organización.

CONTINUAR

3.2. Qué es un almacén de datos

Tal y como se acaba de comentar, los almacenes de datos o *data warehouses* se crean para obtener unos datos de negocio bien consolidados y definidos, de fácil acceso y consistentes.

Si se recuerda el esquema global de solución de inteligencia de negocio expuesto en la unidad anterior, se puede comprobar que el almacén de datos se nutre de numerosas fuentes, entre ellas, obviamente, los sistemas operacionales internos, a través de procesos de extracción, transformación y carga (ETL), para posteriormente ser capaces de explorar y analizar la información y, con ello, generar conocimiento.

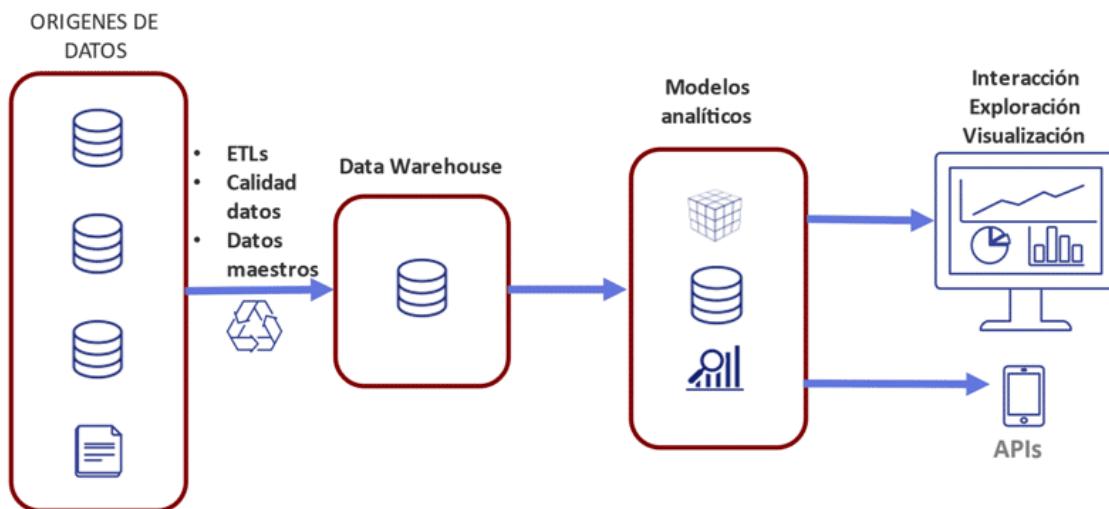


Figura 1. Arquitectura de inteligencia de negocio.

Fuente: elaboración propia.

Hay muchas definiciones de almacén de datos o *data warehouse*. Una primera aproximación es la del profesor Hugh J. Watson, que lo define de la siguiente manera: “Un almacén de datos o *Data Warehouse* es una colección de información creada para soportar las aplicaciones de toma de decisiones”.²

Además, como señala J. L. Cano, “los almacenes de datos se representan habitualmente como una gran base de datos, pero pueden estar distribuidos en distintas bases de datos. El trabajo de construir un almacén de datos corporativo puede generar inflexibilidades o ser costoso y requerir plazos de tiempo que las organizaciones no están dispuestas a aceptar. En parte, estas razones originaron la aparición de los *Data Mart*”.³

²Gray, P.; Watson, H. J. *Decision Support in the Data Warehouse*. Prentice Hall PTR; 1998.

³Cano, *op. cit.*

Por el contrario, un *data mart* está orientado a un grupo específico de usuarios dentro de una compañía u organización, como, por ejemplo, un departamento o un grupo con mismas necesidades de análisis, con objetivos comunes.

Esto permite tener una representación de los datos más acotada, que puede implicar a varias áreas (marketing y ventas, por ejemplo), para dar respuesta a usos muy concretos de análisis sobre la organización.

Normalmente, y dependiendo de la estrategia de diseño que se siga, los *data marts* son más pequeños que los almacenes de datos y serán dependientes de estos. Tienen menos cantidad de información, menos modelos de negocio y son utilizados por un número inferior de usuarios.

Un *data mart* es un **subconjunto** importante de un *data warehouse* orientado a un grupo específico de usuarios y áreas de la empresa. Los *data marts* son más rápidos de desarrollar, y su explotación resulta más sencilla por la variedad y cantidad de datos más acotada.

Las principales diferencias respecto un *data warehouse* son las siguientes:

- Un *data warehouse* es un repositorio central que almacena un gran volumen de datos recopilados de muchos orígenes, y que intenta abarcar todos los departamentos de la organización. El *data mart* está orientado y centrado a un área específica.
- El diseño y desarrollo de un *data mart* es más sencillo por estar más acotado.
- En un *data mart* el volumen de datos es menor y la velocidad de procesamiento, mayor.
- El *data mart* se usa como apoyo en la toma de decisiones tácticas.

En la figura 2 se muestra cómo han ido surgiendo una serie de alternativas a las bases de datos tradicionales que van desde los denominados *appliances de teradata, exadata*, etc., hasta los *data warehouses* en la nube, como Snowflake, pasando por soluciones de procesamiento distribuido, como Hadoop, y diferentes opciones en la nube ofrecidas por Amazon (Redshift), Microsoft (Azure) y Google (BigQuery).

Todas estas soluciones siguen estando vigentes; a la hora de apostar por una o por otra se han de evaluar los factores propios y los requisitos de cada situación.

Data Driven Data Warehouse Evolution

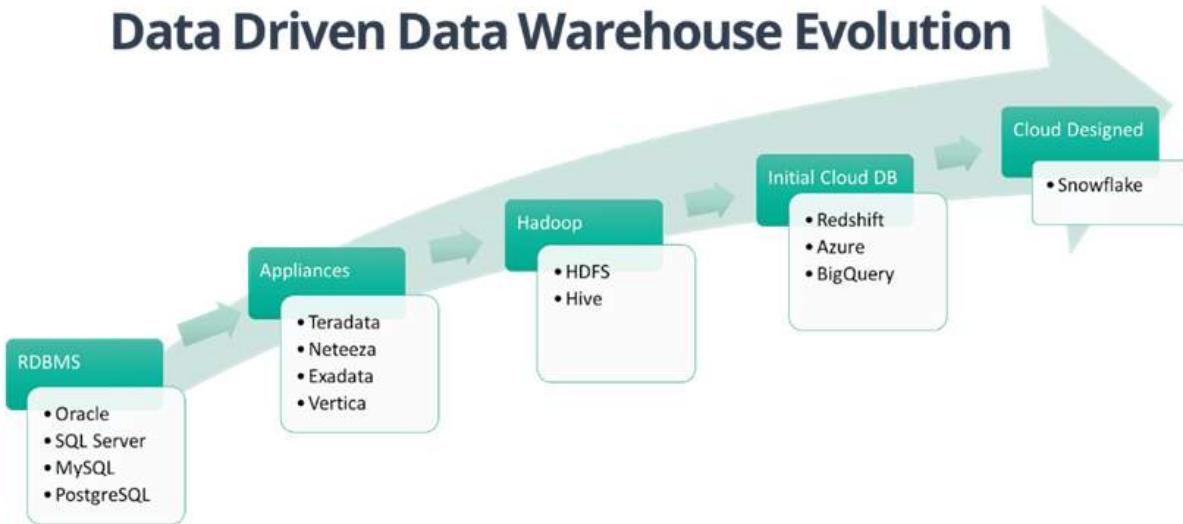


Figura 2. Evolución de *data warehouse*.

Fuente: Data Driven Investor™ (medium.datadriveninvestor.com).

CONTINUAR

3.3. Propiedades o características de un almacén de datos

Bill Inmon, el precursor del concepto de almacén de datos, definió este término de la siguiente manera: “La colección de datos, orientados a tema, integrados, cambiantes con el tiempo y no volátiles, para la ayuda al proceso de toma de decisiones de la dirección de una empresa”.⁴

Los DW contienen datos corporativos granulares. Los datos pueden utilizarse para distintos propósitos, incluso para requisitos futuros desconocidos en el momento.

A continuación, se expondrán las características de los almacenes de datos que engloban esta definición:

⁴Inmon, W. H. *Building the Data Warehouse*. John Wiley & Sons Inc.; 1992.

Orientados a tema

Los datos son almacenados y organizados para que cada elemento registrado esté relacionado a un mismo evento del mundo real.

Por tanto, se organiza en torno a temas importantes, tales como clientes, proveedor, producto y ventas.

En lugar de concentrarse en las operaciones cotidianas y el procesamiento de transacciones de una organización, se centra en el modelado y análisis de datos para la toma de decisiones.

Los almacenes de datos proporcionan una visión sencilla y concisa alrededor de temas específicos, excluyendo los datos que no son útiles en el proceso de la toma de decisiones.

Integrados

Deben estar diseñados para almacenar todos los datos empresariales. Esto quiere decir que un almacén de datos suele construirse integrando múltiples fuentes heterogéneas, como bases de datos relacionales, archivos planos y registros de transacciones procedente de los sistemas operacionales.

Se aplican técnicas de limpieza de datos, integración y unificación para garantizar la coherencia en la nomenclatura de fuentes, estructuras de codificación, medidas de atributo, etc.

Cambiantes con el tiempo

Los datos se almacenan para proporcionar información desde una perspectiva histórica (por ejemplo, los últimos cinco-diez años), pero, a su vez, todos los cambios producidos en los datos deben ser registrados para poder reflejar todas las variaciones en el tiempo.

Como consecuencia, cada estructura de datos en el almacén contiene, implícita o explícitamente, un elemento temporal o, mejor dicho, una referencia al eje temporal.

No volátiles

Un almacén de datos está siempre separado de los datos de aplicación encontrados en una organización.

Debido a esta separación, un almacén de datos no requiere procesamiento de transacciones, recuperación y mecanismos de control de concurrencia al dato.

Por lo general, solo requiere dos operaciones de acceso a datos: la carga inicial de datos y acceso a datos. Por lo que, una vez que los datos son registrados, estos no deben ser modificados ni eliminados.

3.4. Estrategias de diseño del almacén de datos

Aunque existen muchas metodologías de desarrollo de DW, generalmente, se utilizan dos modelos que han probado tener gran éxito en el desarrollo de soluciones BI. A estas propuestas se las conoce como **bus data warehouse**, de Kimball, y **data warehouse empresarial** o **repositorio central de datos**, de Inmon.

3.4.1. Data warehouse de Inmon: diseño top-down

"Un Data Warehouse es un conjunto integrado de bases de datos, con orientación temática, diseñado para el apoyo a la toma de decisiones, donde cada unidad de datos es relevante en algún momento del tiempo".⁵ Inmon establece, según su arquitectura, que los *data marts* surgen a partir del *data warehouse empresarial* (EDW).

También se denominan **data marts dependientes** de un *data warehouse central*; extraen información resumida del *data warehouse* para mejorar el rendimiento y la seguridad del acceso a los datos.

De forma esquemática, se puede representar de la siguiente manera:

⁵Inmon, op. cit.

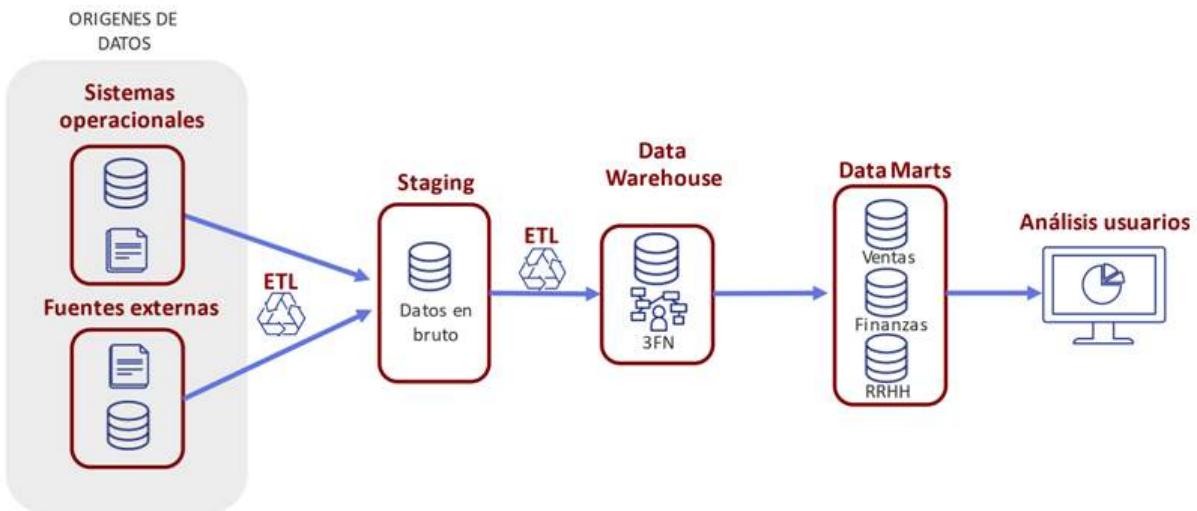


Figura 3. Data warehouse de Inmon.

Fuente: elaboración propia.

CONTINUAR

3.4.2. Data warehouse de Kimball: diseño *bottom-up*

Kimball, sin embargo, define el *data warehouse* como una “copia de las transacciones de datos específicamente estructurada para consultas y análisis”.⁶

Kimball, a diferencia de Inmon, establece una visión más práctica, asumiendo que el almacén de datos está compuesto por la unión de todos los *data marts* corporativos que estén relacionados entre sí, a través de sus **dimensiones**. Para Kimball, el almacén de datos no es más que una **constelación** de *data marts*.

⁶Kimball, R. *The Data Warehouse Toolkit*. John Wiley; 1996.

También se denominan **data marts independientes** que se crean a partir de fuentes externas mediante un proceso ETL. Es una solución rápida y eficiente, pero lleva a la duplicidad de datos. Posteriormente, se pueden llevar estos datos al *data warehouse* central. Los *data marts* se relacionan entre sí mediante las dimensiones generando el modelo en estrella, y es una de las formas más extendidas a la hora de afrontar un sistema de inteligencia de negocio. En este caso el *data warehouse* es opcional.

Se muestra, a continuación, una representación gráfica de su modelo:

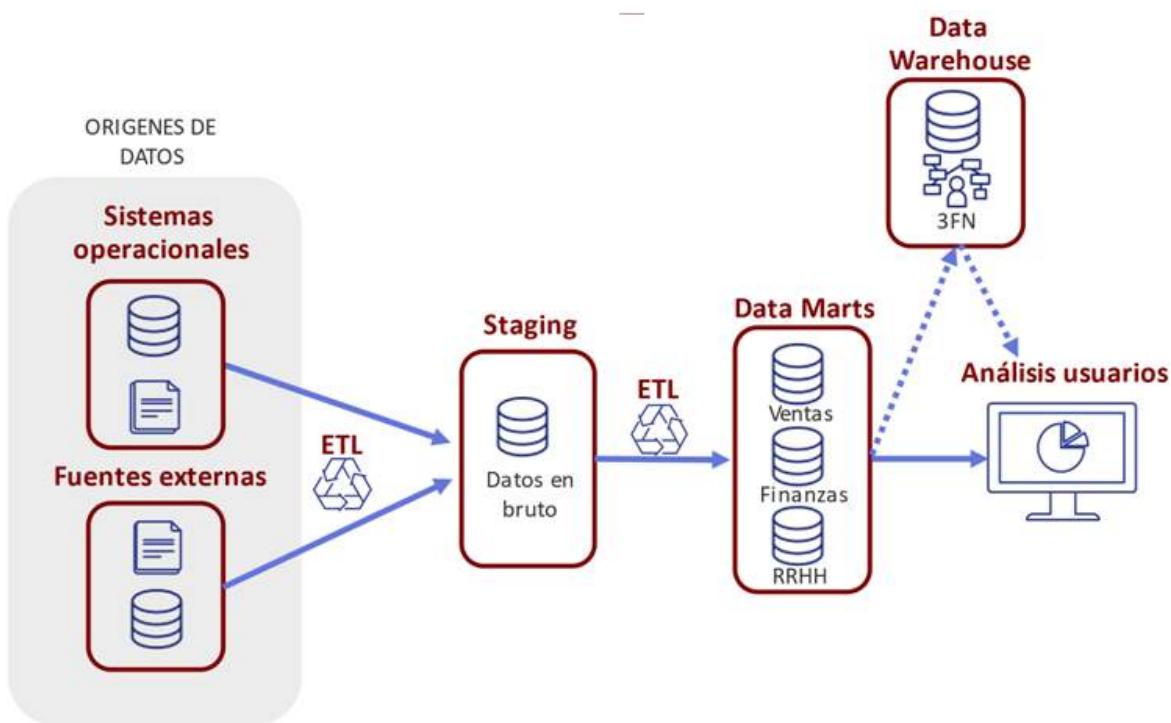


Figura 4. Data warehouse de Kimball.

Fuente: elaboración propia.

Kimball propone un diseño de *data marts* independientes, es decir, los datos son alimentados directamente de los orígenes de información; mientras que Inmon propone *data marts* dependientes, lo que quiere decir que se alimentan desde el almacén de datos corporativo.

Sin embargo, es importante señalar que una estrategia de arquitectura basada en *data marts* independientes, en determinados contextos, puede mantener, e incluso acentuar, el problema de "silos de información", lo que puede dar lugar a entornos difícilmente escalables, con elevados costes de mantenimiento, donde los datos reflejan inconsistencia y falta de unicidad.

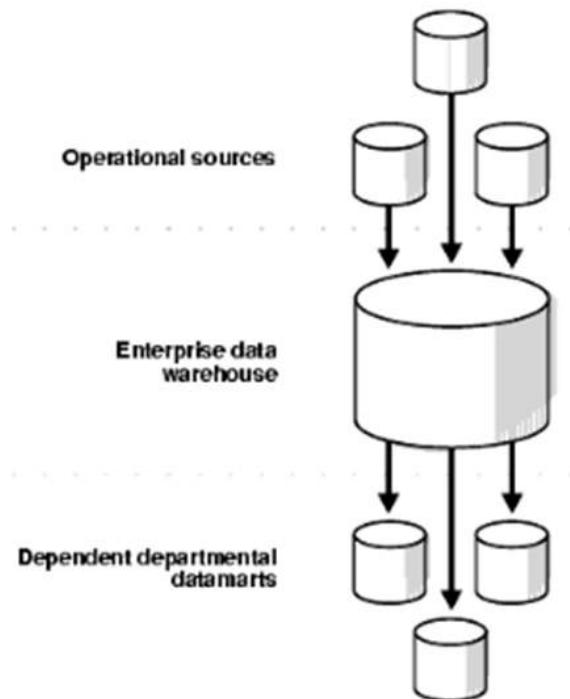
Los *data marts* dependientes extraen datos de un almacén de datos central ya creado, lo que implica que se asume que ya existe un almacén de datos. En cambio, los *marts* de datos independientes son sistemas autónomos construidos al extraer datos directamente de fuentes operacionales o fuentes externas de datos o por una combinación de ambos.

Inmon y los *data marts* dependientes

Un *data mart* dependiente permite unir los datos de su organización en un almacén de datos, lo que posibilita la centralización y unificación de la información.

Figura 5. *Data mart* dependiente.

Fuente: Darshan. Institute of Engineering and Technology.

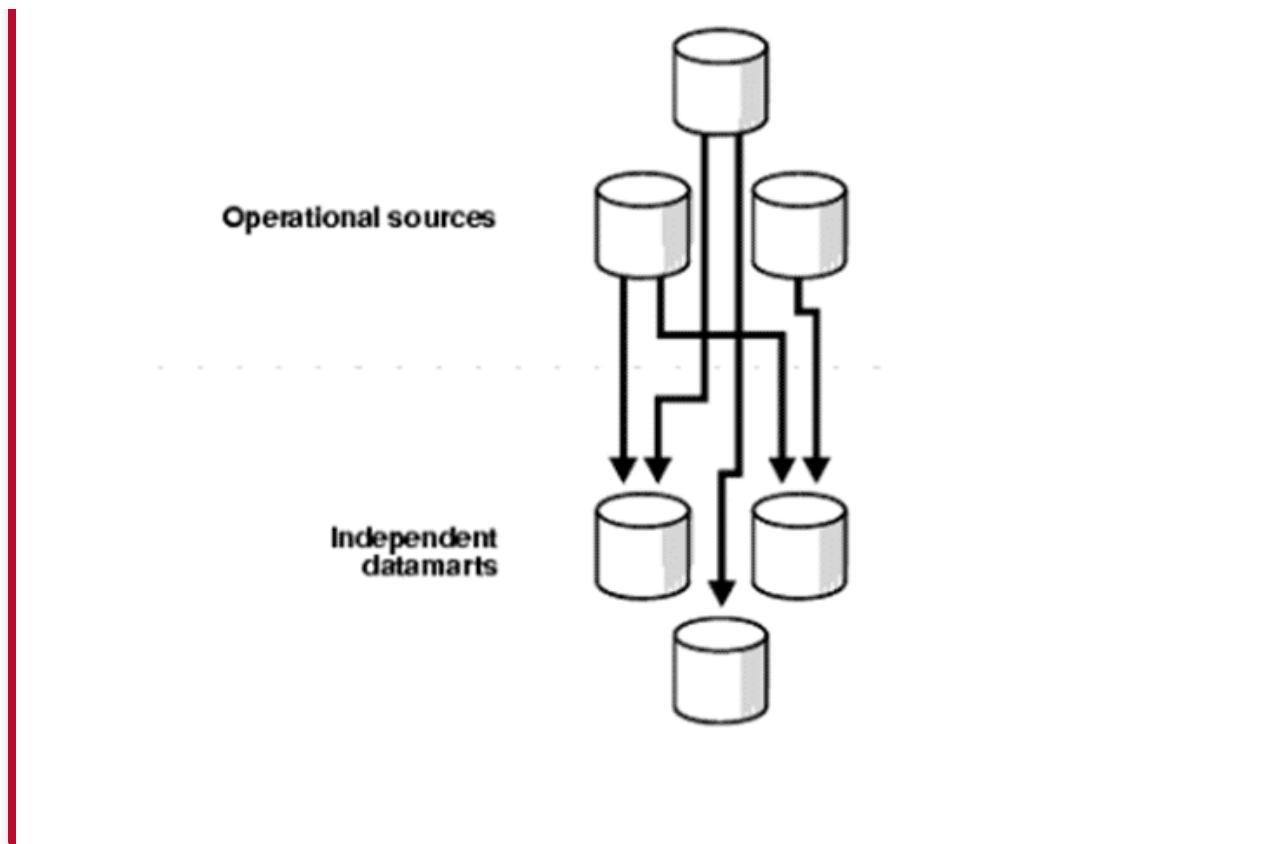


Kimball y los data marts independientes

Se crea un *data mart* de datos independiente sin el uso de un almacén de datos central. Esto podría ser útil para grupos más pequeños de usuarios dentro de una organización.

Figura 6. *Data mart* independiente.

Fuente: Darshan. Institute of Engineering and Technology.



Las diferencias entre estos dos modelos de desarrollo de los almacenes de datos se podrían resumir de la siguiente manera:

	Inmon	Kimball
Presupuesto	Coste inicial alto	Coste inicial bajo
Plazos	Requiere más tiempo de desarrollo	Tiempo de desarrollo inferior
Expertise	Equipo con especialización alta	Equipo con especialización media
Alcance	Toda la compañía	Departamentos individuales
Mantenimiento	Fácil mantenimiento	Mantenimiento más complejo

Figura 7. Diferencias entre el *data warehouse* de Inmon y el de Kimball.

Fuente: elaboración propia.

Por último, hay que recalcar que existen los **data marts híbridos**, que combinan datos de un *data warehouse* central y de fuentes externas.

CONTINUAR

3.5. Detalle de un *data mart*

Un *data mart* es una estructura concreta de datos que tiene como objetivo proveer de información que responda a preguntas de una temática específica. Las principales características son las siguientes:

- Es una aplicación del almacén de datos.
- Está enfocado a satisfacer necesidades de áreas de negocio (departamentos, divisiones, filiales, etc.).
- Permite una implementación rápida, ya que provee de productos funcionales en corto periodo de tiempo.
- Está construido para soportar una línea de negocio.
- Es muy apropiado para sumarizar (crear superredes) grandes cantidades de datos.
- Está construido en un modelo dimensional que usa un esquema de estrella.

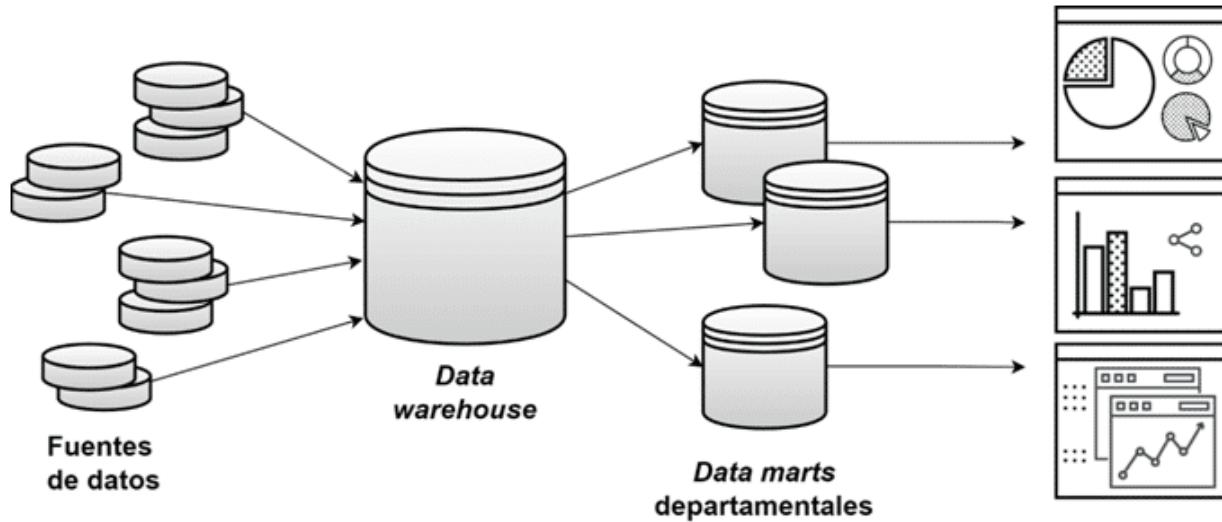


Figura 8. Arquitectura de un DW con *data marts*.

Fuente: elaboración propia.

Por tanto, los *data marts* contienen un subconjunto de datos de toda la entidad que son valiosos para grupos específicos de personas en una organización. Esto permite a los usuarios finales focalizar los esfuerzos de análisis en su área de datos.

i Otra importante razón por la que se recomienda el uso de los *data marts* es porque mejoran el tiempo de respuesta del usuario final, ya que permiten que los usuarios tengan acceso frecuente al tipo específico de datos que necesitan ver y proporcionan datos que facilitan la visión colectiva de un grupo de usuarios.

A continuación, se exponen las ventajas del uso de *data marts*:

- Reducen el volumen de datos que se van a explorar/analizar, lo que acelera las consultas.

- Partición de datos para imponer estrategias de control de acceso.
- Permiten segmentar datos en diferentes plataformas de *hardware*.
- Fácil acceso a los datos que se necesitan consultar con mucha frecuencia.
- Mejoran el tiempo de respuesta del usuario final.
- Menor coste que implementar un almacén de datos completo.
- Contienen solo los datos esenciales del negocio útiles para cada grupo de usuarios.

3.6. Esquema de diseño de un almacén de datos

Un almacén de datos puede ser construido usando un enfoque descendente, *top-down*, o ascendente, *bottom-up*.

Enfoque top-down —

Hace referencia a los *data marts* dependientes.

- El enfoque descendente comienza con el diseño y la planificación general.
- Es útil en los casos en que la tecnología es madura y bien conocida, y cuando los problemas empresariales que deben ser resueltos son claros y se comprenden bien.

Enfoque bottom-up —

Hace referencia a los *data marts* independientes.

- El enfoque ascendente comienza con pilotos, pruebas y prototipos.
- Esto es útil en la primera etapa de modelado de negocio y desarrollo tecnológico.
- Permite a una organización avanzar a un coste considerablemente menor y evaluar los beneficios de la tecnología antes de asumir compromisos significativos.

En todo proceso de diseño de almacén de datos se pueden identificar las siguientes fases o **pasos**:

Paso 1

Elegir un **proceso de negocio** para modelar, por ejemplo, pedidos, facturas, envíos, inventario, administración de cuentas o ventas.

Paso 2

Si el proceso empresarial que modelar es organizativo e implica múltiples colecciones de objetos complejos que afectan a toda la organización, debe seguirse un modelo de **almacén de datos dependiente** (Inmon). Sin embargo, si el proceso es departamental y se centra en el análisis de un tipo de proceso de negocio, debe elegirse un modelo basado en ***data marts* independientes** (Kimball).

Paso 3

Hay que determinar la **granularidad** del proceso de negocio. La granularidad es el nivel fundamental, atómico, de los datos que representar en la tabla de **hechos** para este proceso, por ejemplo, transacciones individuales, instantáneas diarias individuales, etc.

Paso 4

Hay que determinar las **dimensiones** que se aplicarán a cada registro de tabla de hechos. Las dimensiones más comunes son tiempo, artículo, cliente, proveedor, almacén, tipo de transacción y estado.

Paso 5

Hay que determinar las medidas que llenarán cada registro de tabla de hechos. Las **medidas** más frecuentes son cantidades numéricas aditivas, como los dólares vendidos, y unidades vendidas.

Pero ¿en qué consisten los conceptos de **dimensión, medidas y tabla de hechos**? Estos conceptos vienen ligados al concepto de multidimensionalidad, en el cual se profundizará más adelante, pero es necesario ir adelantando una breve definición:

Tabla de hechos

La tabla de hechos contiene todos los datos que son relevantes para la unidad de negocio que se desea analizar.

En ella se almacenan, registro a registro, los hechos y métricas: aquello que se desea medir o cuantificar, es decir, el hecho que se quiere cuantificar. Se entiende por métrica aquellos valores sensibles de ser cuantificados o medibles, normalmente numéricos, y que sirven para evaluar el progreso de un factor de la organización.

Cada registro de hechos está compuesto por un conjunto de claves foráneas –una clave por cada dimensión– y uno o varios valores numéricos, que son los propios hechos en sí.

Los valores vienen determinados por los objetivos de la organización y pueden incluir medidas como número de habitantes, importe de las subvenciones, número de inspecciones, edad media de los cargos electos, etc.

Dimensiones y tabla de dimensiones

Las dimensiones describen hechos, las tablas de dimensiones contienen numerosos atributos que permiten describir un hecho con mayor detalle y en función de este eje.

Las dimensiones son los conceptos por los que se requiere analizar los hechos ocurridos en el sistema. En estadística se denominaría una **variable categórica**. Este tipo de campos se asocian, principalmente, con aquellos que no tienen un valor numérico. Suelen ser valores que **responden a preguntas** típicas como dónde, cuándo, quién y qué. Por lo tanto, se puede decir que las dimensiones más comunes son de carácter geográfico (dónde), temporal (cuándo), de personas (quién) y productos (qué). Por ejemplo, la dimensión tiempo permite describir y analizar el hecho a través de esta dimensión o eje de análisis. Da una idea de cómo se quiere analizar la información.

IV. Herramientas de análisis de un almacén de datos: OLAP

Existen distintas tecnologías para analizar la información que reside en un almacén de datos, pero la más extendida es OLAP. Los sistemas OLAP son bases de datos orientadas al procesamiento analítico.

Los analistas buscan poder analizar la información de diferente manera y sin partir de consultas predefinidas. Demandan flexibilidad para poder analizar y explorar la información, navegando a través de ella, a lo largo de diferentes dimensiones o ejes, y saltando de mayor a menor nivel de detalle, para poder apreciar tanto los grandes números como el detalle que causa cualquier variación en cualquier instante de tiempo.

Como respuesta a estas necesidades, surge OLAP: ofrece la capacidad de realizar un análisis inmediato de datos a lo largo de varias dimensiones —como ventas a lo largo del tiempo, coste por geografía, etc.—, de una forma dinámica, y permitiendo que el usuario final realice estas configuraciones, lo que da lugar a lo que se conoce como análisis multidimensional.

Esta es la forma natural que los analistas aplican para analizar la información, ya que los modelos de negocio normalmente son multidimensionales. La visualización de la información es independiente con respecto a cómo se haya almacenado.

Según lo planteado por Nigel Pendse,⁷ las herramientas OLAP deben pasar la prueba FASMI (análisis rápido de información multidimensional compartida). Por lo tanto, deberían ser:

⁷Pendse, N. *The OLAP Report. What is OLAP?* Business Application Research Center; 2007.

- Lo suficientemente rápidas como para permitir consultas interactivas.
- Deberían ayudar a la tarea de análisis, ya que proporcionan flexibilidad en el uso de herramientas estadísticas (en función del tipo de estudio que se esté realizando).
- Deberían proporcionar mecanismos de seguridad –confidencialidad e integridad– para permitir el intercambio de datos.
- Deberían proporcionar una visión multidimensional para que los usuarios puedan usar la metáfora del cubo de datos.
- Deberían poder administrar grandes volúmenes de datos y metadatos.

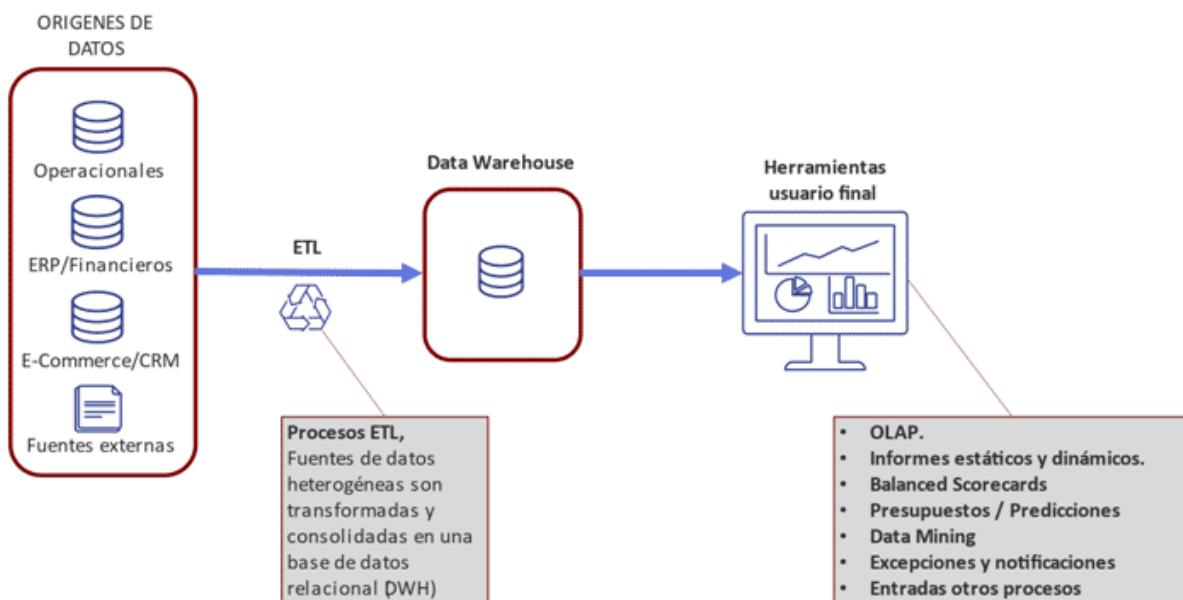


Figura 9. Diagrama de un entorno *data warehouse*.

Fuente: elaboración propia.

CONTINUAR

4.1. OLAP: procesamiento analítico

Como se ha citado en módulos y unidades anteriores, la finalidad de los sistemas de bases de datos operacionales consiste en la realización de transacciones y la resolución de consultas, por esa razón se denominan sistemas OLTP (*online transaction processing*).

Pero, si estos sistemas gestionan y mantienen la operativa de datos de la organización de forma eficiente, ¿por qué surge OLAP?

4.2. ¿Por qué implementar otra base de datos aparte de la OLTP?

Los sistemas OLAP se imponen por diversas razones:

- El *data warehouse* usa modelos más simples y orientados a negocio, fácilmente comprensibles, a diferencia de los modelos OLTP.
- OLAP proporciona un óptimo rendimiento en consulta de grandes volúmenes de información, en contraposición al bajo rendimiento demostrado por los OLTP.
- Los OLTP no están diseñados con el propósito de dar respuesta a grandes consultas de datos, por lo que, cuando asumen dicha función, consumen recursos que entran en conflicto con los dimensionados y asignados para su propósito transaccional.

- Las consultas son más complejas y extensas como consecuencia de la normalización y de los numerosos cruces de tablas que han de realizarse para extraer los datos.

4.3. OLAP vs. OLTP

Si se analizan las características de estas dos tecnologías, se ven claramente las diferencias sustanciales entre ambas:

	OLTP	OLAP
Características	Procesamiento operativo.	Procesamiento de información.
Orientación	Transacciones.	Análisis.
Tipo de usuarios	Empleados de los procesos operativos.	Ejecutivos.
Función	Operaciones día a día.	Requisitos de información a largo plazo, apoyo a las decisiones.
Diseño BB. DD.	Modelo relacional. Orientado a la aplicación.	Modelo en estrella o copo de nieve. Orientado a la información.
Datos	Actuales: garantizados hasta la fecha.	Históricos: precisión mantenida a través del tiempo.
Resumen	Primitivo, muy detallado.	Resumido, consolidado.
Tipo de vista	Relación detallada, plana.	Resumida, multidimensional.

Unidad de trabajo	Transacciones simples y rápidas.	Consultas complejas.
Acceso	Lectura y escritura.	Mayoritariamente, lectura.
Focalizado en	Entrada de datos.	Lectura de datos.
Operaciones	Índice/hash en clave primaria.	Muchos escaneos.
Número de registros a los que se accede	Decenas.	Millones.
Número de usuarios	Miles.	Cientos.
Tamaño BB. DD.	100 MG a GB.	100 GB a TB.
Prioridad	Alto rendimiento, alta disponibilidad.	Alta flexibilidad, autonomía del usuario final.
Métrica	Rendimiento de transacción.	Rendimiento de la consulta, tiempo de respuesta.

Tabla 1. Comparación entre los sistemas OLAP y OLTP.

Fuente: elaboración propia a partir de Jiawei, H.; Kamber, M. *Data Mining: Concepts and Techniques* (2.a ed.). Morgan Kaufmann Publishers; 2001.

Los sistemas OLAP se actualizan con poca frecuencia, tradicionalmente, una vez al día o una vez a la semana, pero este factor puede ser necesario para acceder a millones de filas de datos para devolver un conjunto de resultados. Están diseñados para mejorar la velocidad de consulta. Es por esto por lo que las arquitecturas OLAP tratan de almacenar los datos de forma que sea más efectivo el análisis dinámico.

Estos análisis suelen implicar, generalmente, la lectura de grandes cantidades de datos para llegar a extraer algún tipo de información útil (tendencias, patrones de comportamiento, generación de informes, etc.).

En los sistemas OLAP, como un almacén de datos, los datos se suelen estructurar en un modelo de datos especial, denominado modelo multidimensional o cubo multidimensional, que proporciona mejores resultados de explotación de datos, comparado con otros modelos. Además, su diseño sigue un estándar intuitivo, cuya meta es la fácil comprensión y el buen rendimiento.

CONTINUAR

4.4. Tipos de OLAP: ROLAP vs. MOLAP

Es importante conocer las diferentes arquitecturas OLAP existentes, sus ventajas e inconvenientes.

4.4.1. ROLAP: ventajas, inconvenientes y usos

R-OLAP o ROLAP (*relational online analytical processing*) permite realizar análisis multidimensional dinámico a partir de los datos almacenados en una base de datos relacional. En el modelo R-OLAP, los datos se almacenan como filas y columnas de forma relacional.

Con el fin de ocultar la estructura de almacenamiento al usuario y de presentar datos de forma multidimensional, se crea una capa semántica de metadatos. La capa de metadatos admite la asignación de dimensiones a las tablas relacionales. Los metadatos adicionales admiten las summarizaciones o creaciones de superredes y agregaciones. Se pueden almacenar los metadatos en bases de datos relacionales.⁸

La información se almacena en las tablas de la base de datos relacional. El sistema de gestión de bases de datos relacional (RDBMS) dispone de una tabla de hechos, la cual almacena la actividad para analizar de la empresa, y esta se relaciona con otras tablas que conforman las diferentes dimensiones (productos, tiempo, etc.).⁹

Al modelo conformado por la unión de la tabla de hechos y las tablas de dimensiones se le denomina modelo en estrella. Un modelo en estrella es la realización más simple de un *data mart* y siempre requiere de un proceso de desnormalización.

⁸Ponniah, P. *Data Warehousing Fundamentals. A Comprehensive Guide for IT Professionals*. John Wiley & Sons; 2004.

⁹Reddy, M. S.; Khan, V. *Comparative Analysis of On-Line Analytical Processing Tools*. Göteborg: IT University of Göteborg; 2007.

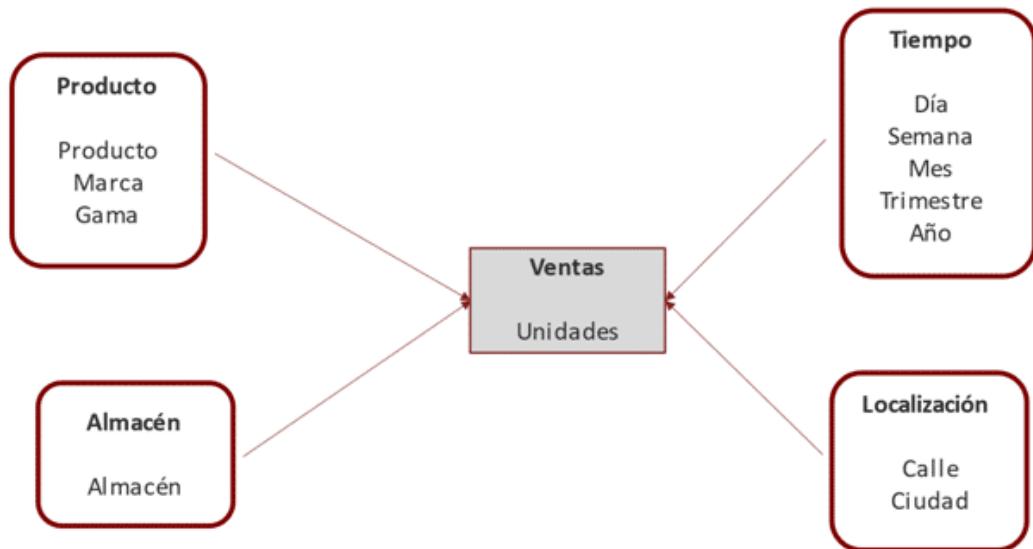


Figura 10. Ejemplo de modelo en estrella.

Fuente: elaboración propia.

Atendiendo a la arquitectura de una solución de inteligencia de negocio estudiada en la unidad anterior, se puede decir que los sistemas ROLAP se suelen ajustar a una arquitectura de datos de tres niveles: datos, aplicación y presentación.

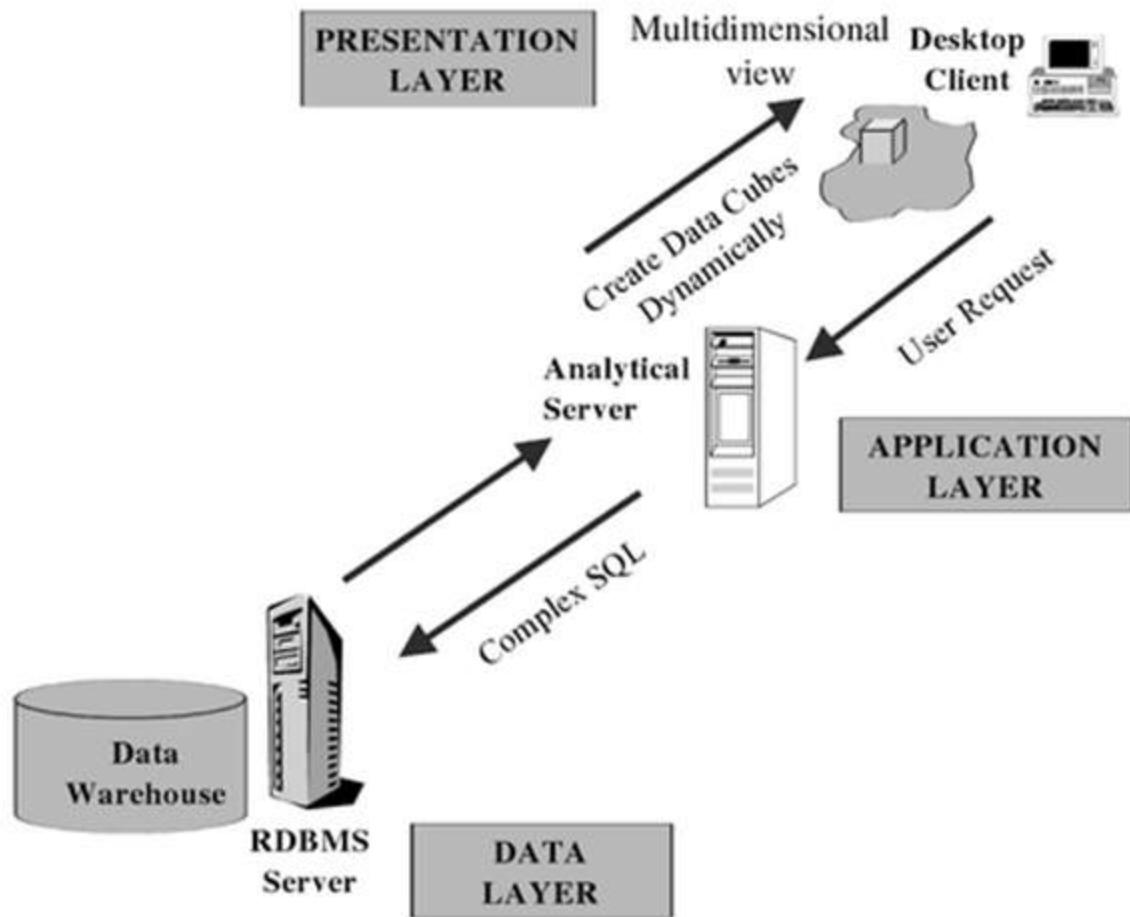


Figura 11. Arquitectura de modelo ROLAP.

Fuente: Ponniah, P. *Data Warehousing Fundamentals. A Comprehensive Guide for IT Professionals*. John Wiley & Sons; 2004.

Por tanto, el sistema OLAP relacional (ROLAP) aprovecha las ventajas de esta nueva tecnología analítica y la aplica sobre bases de datos relacionales tradicionales. Pero, para ello, para poder aplicar y realizar este análisis haciendo uso de una base de datos tradicional con un motor OLAP, es necesaria una capa intermedia que traduzca el lenguaje OLAP (multidimensional) al lenguaje comprensible para estas bases de datos: SQL.

Esto quiere decir que, para implementar estos sistemas, es necesario implementar un modelo físico en la base de datos compuesto por tablas, con una estructura concreta, y un modelo lógico, definiendo modelos multidimensionales, especialmente pensados para ser explotados por herramientas de visualización.

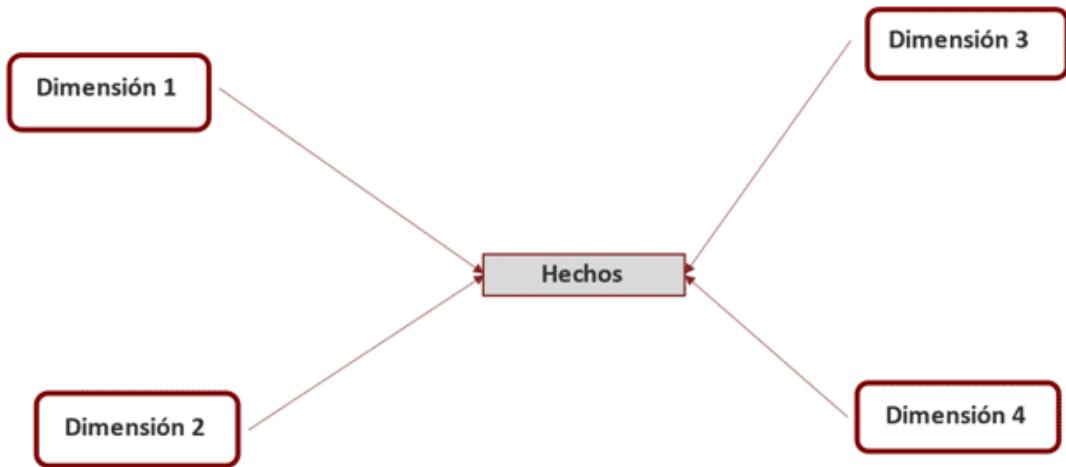
Modelo físico: existen varios modelos válidos para implementar la capa física, como los modelos en estrella, en constelación y modelos copo de nieve.

El modelo de estrella

Es la arquitectura de almacenamiento de datos más simples, cuyo esquema está definido por una sola tabla de hechos central, rodeada de tablas de dimensiones. En este caso, el esquema multidimensional no incluye ninguna jerarquía física, a nivel de base de datos relacional, entre los atributos de las diferentes dimensiones del esquema, de forma que cada una de ellas almacena todos sus atributos. Este es el modelo que se asocia a los *data marts*.

Figura 12. Ejemplo de modelo en forma de estrella.

Fuente: elaboración propia.



El modelo copo de nieve

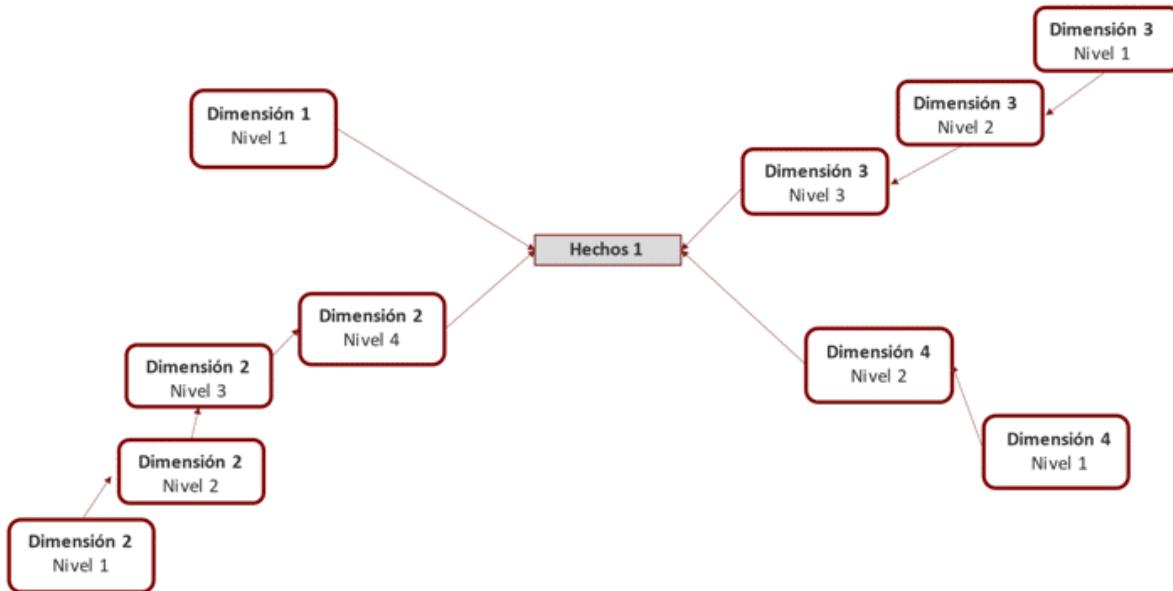
Es una arquitectura de almacenamiento de datos que recibe su nombre por su similitud a un copo de nieve. Al igual que el modelo de estrella, consta de una o muchas tablas de hechos rodeadas de tablas de dimensiones, donde cada nivel de la dimensión puede ser definido en otra tabla de dimensión y se conectan entre sí con una relación (n:1).

Es una arquitectura algo más compleja que el modelo de estrella. Es importante resaltar que esta arquitectura obtiene un rendimiento menor que el modelo anterior, ya que necesita incrementar los

joins para consultar la información. Pero permite una mayor reutilización de las tablas, ya que no todas las tablas de hechos tienen el mismo nivel de detalle para todas las jerarquías de dimensiones.

Figura 13. Ejemplo de modelo en forma de copo de nieve.

Fuente: elaboración propia.



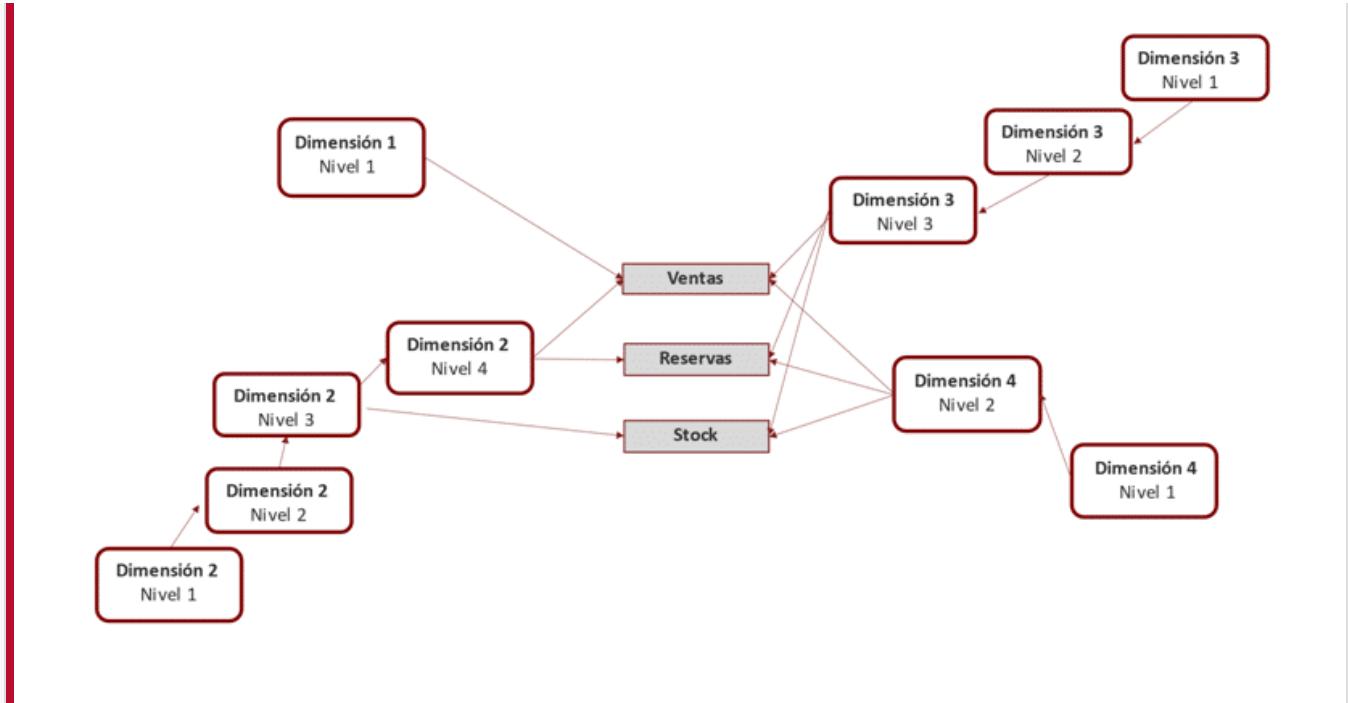
Modelo constelación de hechos

Este modelo consiste en la unión de varios *data marts*, ya sean modelos de copo de nieve o de estrella. El hecho principal es que comparten dimensiones, lo que implica su principal ventaja: la optimización del espacio.

A continuación, se muestra un modelo en forma de constelación formado por dos *data marts*: ventas y transportes y sus respectivas tablas de hechos. Este modelo suele estar más asociado a los *data warehouses* empresariales más tradicionales.

Figura 14. Ejemplo de modelo de constelación de hechos.

Fuente: elaboración propia.



La característica fundamental de estos modelos es que están orientados a la consulta, es decir, optimizan la velocidad de consulta y hacen más eficiente el análisis y la aplicación de grandes consultas, frente al modelo tradicional de relaciones. Estos modelos implican la definición e implementación de tablas de dimensiones y tablas de hechos.

Modelo lógico: sobre el modelo físico de tablas definidas en base de datos, se definirá un modelo lógico, que es el propio modelo multidimensional, ya que la base de datos en sí no tiene tal naturaleza. En este modelo lógico, se definirá todo lo necesario para crear un cubo multidimensional de análisis y su correspondiente mapeo con el modelo lógico.

De esta forma, un usuario realizará una consulta avanzada, navegación o análisis multidimensional a través del motor ROLAP, cuya función será realizar consultas al vuelo multidimensionales y a la capa lógica, para verificar el mapeo entre capas.

ROLAP hace uso de resultados precalculados y genera dinámicamente resultados de consultas a partir de los datos de mayor nivel de detalle, si fuera preciso. Las arquitecturas ROLAP aplican estrategias de optimización de accesos para acelerar consultas como el particionado de los datos, la desnormalización y múltiples *joins*.

VENTAJAS	DESVENTAJAS	USOS
----------	-------------	------

Las principales **ventajas** de los sistemas ROLAP son las siguientes:

- Su uso es muy eficaz cuando se requiere acceso a alto nivel de detalle de los datos o se requiere mucha flexibilidad en la generación de análisis.
- Permiten el uso de bases de datos relacionales, lo que implica que el dato es accesible para aplicaciones tercera de forma estándar.
- Soportan mayor tipo de datos y métricas no aditivas.
- Mejor rendimiento en modelos de grandes volúmenes de datos, sin imponer ninguna limitación en cuanto a volumetría de datos.
- Arquitectura muy escalable.
- No requieren copia de datos, se accede a ellos tal y como se tienen en el almacén de datos.
- Hacen uso de herramientas ETL para procesos de carga, lo que permite su integración y posibilita personalizar dichos procesos.
- Son más flexibles en entornos más dinámicos, permiten la redefinición y ampliación de modelos lógicos a nivel de metadatos, son menos pesados que con otros modelos.

VENTAJAS	DESVENTAJAS	USOS
----------	-------------	------

Algunas de sus **desventajas** son las siguientes:

- Aunque son sistemas diseñados para que las consultas puedan ser resueltas de forma muy rápida, siempre serán algo menos rápidos que los sistemas MOLAP.
- No hay indexación implícita.

- Hay que incluir una capa de metadatos adicional para el mapeo OLAP-relacional.

VENTAJAS	DESVENTAJAS	USOS

Viendo sus ventajas y desventajas, los **usos** frecuentes de estos sistemas son los siguientes:

- En entornos que demanden grandes volúmenes de datos con accesibilidad hasta el más mínimo detalle.
- En entornos donde los usuarios quieren realizar consultas *ad hoc* de cualquier atributo.

CONTINUAR

4.4.2. MOLAP: ventajas, inconvenientes y usos

Los sistemas MOLAP son los sistemas OLAP multidimensionales propiamente dichos, es decir, aquellos que implementan un almacenamiento de datos multidimensional para proporcionar el análisis analítico.

En los sistemas ROLAP, los datos se almacenan en bases de datos relacionales no multidimensionales y esta multidimensionalidad se define en una capa lógica. En los MOLAP, esto no es necesario, porque el dato ya se guarda en una estructura multidimensional directamente. Esta estructura, en la mayoría de los casos, es propiedad de los fabricantes.

Tradicionalmente, en los sistemas MOLAP se dan los siguientes **flujos de datos**:

1

Rutinas *batch*: cargan los datos desde los orígenes hasta el sistema MOLAP.

2

Una vez cargados los datos en bruto, se aplican y calculan datos agregados a las dimensiones definidas, llenando toda la estructura multidimensional de datos. En resumen, se precalcula cualquier

Posteriormente, se realiza

un proceso de indexado

para mejorar tiempos de
acceso y consulta.

Tal y como se puede comprobar, no es necesario realizar una definición de metadatos, puesto que no es imprescindible mapear la base de datos con el modelo multidimensional.

VENTAJAS

DESVENTAJAS

USOS

Las principales **ventajas** del sistema MOLAP son las siguientes:

- Es un sistema puramente multidimensional.
- Posee una mayor velocidad de análisis. Se consigue el mejor rendimiento y acceso a los datos durante el análisis, principalmente cuando se accede a información agregada.
- El alto rendimiento en la resolución de consultas se debe al previo almacenamiento optimizado del dato, a la indexación multidimensional de los datos y a la memoria caché.
- Los cubos multidimensionales MOLAP están optimizados para las operaciones nativas *slicing* y *dicing* de OLAP.
- “Pueden realizar cálculos complejos, pues la mayoría de ellos se han precalculado durante la construcción del cubo multidimensional, por eso los datos se pueden devolver tan rápido”.¹⁰
- No hay que definir capa de mapeo.

VENTAJAS	DESVENTAJAS	USOS
----------	-------------	------

Algunas de sus **desventajas** son las siguientes:

- Estos sistemas tienen dificultades para generar y realizar consultas en modelos con numerosas dimensiones.
- Su capacidad para crear dinámicamente nuevas agregaciones o hacer cálculos de ratios o métricas, que hayan sido precalculadas previamente, es limitada. Y, en general, la navegación y, por tanto, el análisis están limitados a la definición previa realizada.
- Es el sistema que más espacio en disco requiere y no existe un estándar fijo de modelo de almacenamiento, lo que implica que cada fabricante define su propia tecnología.
- Se introduce redundancia de datos, ya que realiza copia de los datos iniciales.
- Hay que tener en cuenta que precalcular o preconsolidar los datos transaccionales proporcionan velocidad de consulta, pero preconsolidar totalmente los datos de entrada puede requerir una enorme cantidad de espacio en disco y tiempo de procesamiento.
- Se necesitan expertos para implementar y mantener estos sistemas.
- Cada fabricante y cada producto definen un lenguaje específico.

VENTAJAS	DESVENTAJAS	USOS
----------	-------------	------

Viendo sus ventajas y desventajas, los **usos** frecuentes de estos sistemas son los siguientes:

- Se recomienda MOLAP para modelos que son consultados con mucha frecuencia y cuya respuesta debe darse lo más rápidamente posible, pero sin necesitar que el cubo cambie constantemente. Es decir, entornos donde los modelos están claramente definidos y no cambien en el tiempo.
- Entornos donde no se requiera acceso a los datos de manera muy detallada.
- Adecuados para dar soluciones departamentales y con un número de dimensiones limitado.

¹⁰ Rojas Bartomeu, P. *Estudio comparativo de bases de datos analíticas*. TFM Máster en Tecnologías de la Información, Especialidad en Ingeniería del Software y Sistemas de Información. Everis, Facultat d'Informàtica de Barcelona, UPC; 2009.

4.4.3. HOLAP

Los sistemas HOLAP son los sistemas OLAP híbridos, que no son más que la combinación de sistemas ROLAP y MOLAP.

Tres modos de almacenamiento:

- Multidimensional OLAP (MOLAP)
- Relacional OLAP (ROLAP)
- Híbrido OLAP (HOLAP)

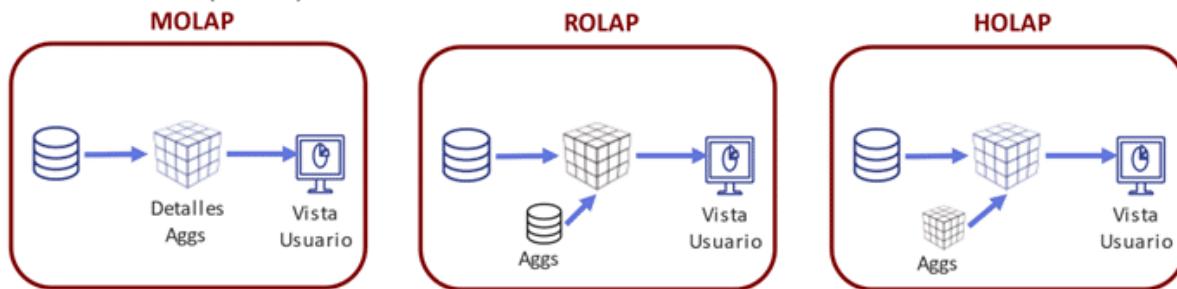


Figura 15. Modos de almacenamiento.

Fuente: elaboración propia.

V. Multidimensionalidad y el modelo multidimensional

5.1. Modelo multidimensional

Hasta ahora, se ha citado en varias ocasiones los términos “multidimensional” y “modelo multidimensional” para hacer referencia al modelo de datos usado por los sistemas analíticos OLAP. A continuación, se estudiará en detalle el modelo multidimensional.

Tal y como ya sabe el alumno, el modelo entidad-relación usado por los sistemas operacionales tiene como finalidad optimizar dichos procesos, eliminando redundancia en el almacenamiento de datos, de forma que los procesos de inserción y mantenimiento del dato y su consistencia sean óptimamente eficientes. La intención en estos modelos es que el dato esté representado las mínimas veces posible.

Sin embargo, esto implica que la ejecución (y definición) de simples consultas de datos pueda resultar muy costosa. Sin embargo, este planteamiento no es el más apropiado para los entornos de inteligencia de negocio, donde el objetivo principal es analizar de forma eficiente y ágil la información, realizando numerosas consultas y accesos a los datos.

El modelo multidimensional surge como solución a este problema. Su objetivo fundamental es permitir realizar consultas de forma eficiente, aunque no se conozca la definición de dichas consultas.

El modelo multidimensional, comúnmente llamado cubo multidimensional, es, en definitiva, una arquitectura concreta de cómo definir y almacenar los datos. Permite organizarlos en dimensiones, hechos y medidas.

Hechos

Los **hechos** representan aquello que se desea analizar (compras, ventas, préstamos, unidades, etc.).

Dimensiones

Las **dimensiones** indican los ejes de análisis o aquello por lo que se desea analizar los hechos (tiempo, geografía, etc.).

Medidas

Las **medidas** (o atributos de hecho) son la información relevante sobre el hecho.

Atributo de dimensión

También es importante definir el **atributo de dimensión**, frecuentemente usado, que sirve para aportar información descriptiva de cada dimensión.

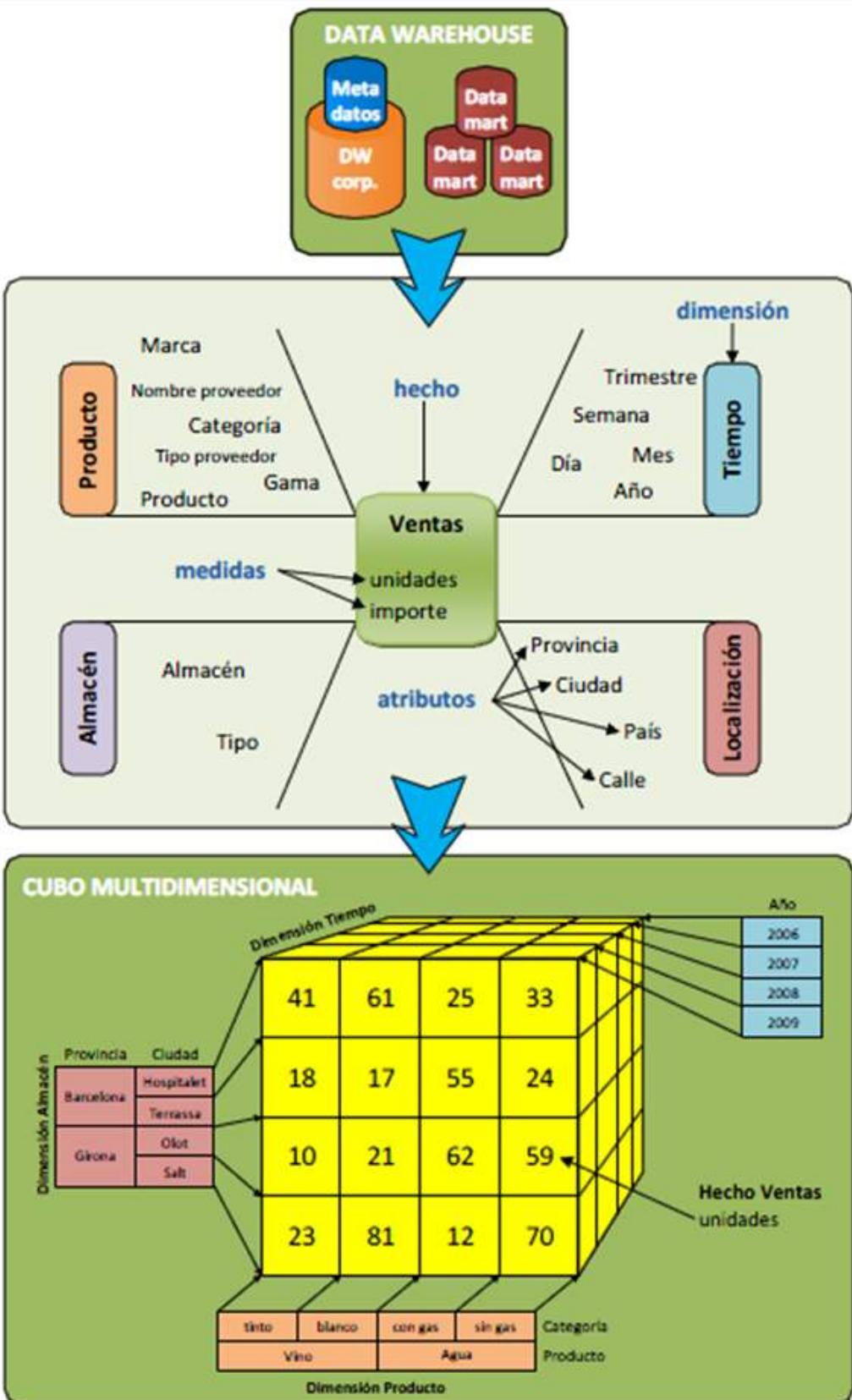


Figura 16. Ejemplo de modelo multidimensional.

Fuente: Ibermática. *Business Intelligence, el conocimiento compartido*. 2007.

La figura 16 muestra un ejemplo de modelo multidimensional muy sencillo. En este caso, se quiere analizar el hecho “ventas”, que contiene dos medidas: unidades e importe. Para describirlo, se utilizan cuatro dimensiones: producto, tiempo, localización y almacén, con sus correspondientes atributos de dimensión.

Por último, hay que destacar la relación existente entre almacén de datos, *data mart*, modelo multidimensional y modelo de estrella:

- Como se habrá podido observar gracias a la lectura de los apartados anteriores, un *data mart* es un submodelo de datos en un almacén de datos. Este, a su vez, puede contener uno o varios modelos multidimensionales, que pueden estar soportados por uno o varios modelos de estrella en el plano físico.
- También se puede afirmar que el modelo de estrella es la realización física más atómica de un modelo multidimensional y de un *data mart*.

CONTINUAR

5.2. ¿En qué consiste el modelo lógico? ¿Cuáles son las estructuras básicas necesarias?

Como se vio anteriormente, todo modelo físico multidimensional requiere de dimensiones, hechos y modelos de estrella.

El modelo lógico no es más que una definición estructurada, ordenada y jerarquizada de dichos elementos. Así, los componentes más importantes de un esquema son cubos, medidas y dimensiones. Se recuerda lo siguiente:

CUBO	MEDIDA	DIMENSIÓN
------	--------	-----------

Un **cubo** es un conjunto de dimensiones y medidas en un área de negocio determinada.

CUBO	MEDIDA	DIMENSIÓN
------	--------	-----------

Una **medida** es una cantidad cuya medición se desea conocer, por ejemplo, las ventas de unidades de un producto o el precio de coste de los artículos del inventario.

CUBO	MEDIDA	DIMENSIÓN
------	--------	-----------

Una **dimensión** es un atributo, o conjunto de atributos, con el que se pueden dividir las medidas en subcategorías. Por ejemplo, su color, el sexo del cliente, el almacén en el que se vende el producto... Las funciones principales de las dimensiones son las siguientes:

- **Filtrar** información para reducir el conjunto de datos que analizar.
- **Agrupar** la información para poder analizar totales y subtotales de los valores numéricos.
- **Etiquetar** los valores.

Estos conceptos ya se estudiaron anteriormente para diseñar el modelo físico. Para el modelo lógico, hay que introducir también los siguientes conceptos:

MIEMBRO	JERARQUÍA	NIVEL	DIMENSIÓN
---------	-----------	-------	-----------

Un **miembro** es un punto dentro de una dimensión, determinado por un conjunto específico de valores de atributos. O lo que es lo mismo: cada uno de los valores que puede tomar un elemento del nivel que sea. Por ejemplo, la construcción cultural de género se compone de dos miembros: masculino y femenino.

MIEMBRO	JERARQUÍA	NIVEL	DIMENSIÓN
---------	-----------	-------	-----------

Una **jerarquía** es un conjunto de miembros organizados en una estructura de varios niveles para el análisis práctico. Como se ha visto anteriormente las dimensiones son clave para explorar el *data warehouse*, pero para ordenar las dimensiones conceptualmente existen las jerarquías de dimensiones. Estas facilitan el análisis por distintas dimensiones relacionadas por su significado. Por ejemplo, la jerarquía tiempo contará con varias dimensiones, como año, trimestre, mes, semana y día. Las jerarquías de dimensiones tienen que garantizar la integridad referencial entre los distintos niveles de la jerarquía.

Las jerarquías se pueden dividir en dos grandes clases:

- **Jerarquías naturales:** todas aquellas que son genéricas para todo el mundo y que no cambian normalmente. Ejemplos de este tipo de jerarquías son las jerarquías de fechas o la geografía.
- **Jerarquías propias o de negocio:** todas aquellas específicas de la organización que está desarrollando la jerarquía. Ejemplos de este tipo de jerarquías son la jerarquía de productos o de fuerza de ventas. Cada empresa puede organizar esto de la manera que le convenga a su negocio.

MIEMBRO	JERARQUÍA	NIVEL	DIMENSIÓN
---------	-----------	-------	-----------

Un **nivel** es un conjunto de miembros cuya distancia a la raíz de la jerarquía es la misma.

MIEMBRO	JERARQUÍA	NIVEL	DIMENSIÓN
---------	-----------	-------	-----------

Una **dimensión** es una colección de las jerarquías que se discriminan sobre el mismo atributo en la misma tabla de hechos (por ejemplo, el día en que se produjo una venta).

Por tanto, ¿cuáles son las estructuras (constructores) necesarias para crear un modelo lógico (esquema)?

- Todo esquema o modelo lógico debe contener al menos un cubo.
- Un cubo debe contener, al menos, una dimensión de análisis y una dimensión de métricas, con una o varias métricas definidas.
- Toda dimensión de análisis debe contener una o varias jerarquías.
- Toda jerarquía debe tener uno o varios niveles.
- Todo cubo o modelo lógico debe tener su correspondiente mapeo físico o tabla de hechos.

De forma gráfica, el esquema más sencillo que implementar sería el que se muestra en la figura siguiente:



Figura 17. Ejemplo de esquema.

Fuente: elaboración propia.

VI. Desnormalización

6.1. Proceso de normalización

Para entender la importancia del proceso de desnormalización en los sistemas OLAP, se debe comenzar recordando el proceso de normalización de las bases de datos en su diseño y su justificación:

- Como se ha visto anteriormente, en toda arquitectura de inteligencia de negocio se extrae la información de numerosos y diferentes orígenes de datos, en su mayoría residentes en sistemas OLTP operacionales. Estos sistemas están diseñados para aportar las aplicaciones transaccionales que soportan el negocio minuto a minuto, por lo que están normalizadas para realizar actualizaciones de forma eficiente y consistente.
- El objetivo final de estos sistemas es obtener alta eficiencia y procesar un alto volumen de transacciones cortas.
- Los sistemas OLTP y modelos E-R definen y realizan un proceso intensivo de normalización.
- Este proceso de normalización busca de manera única la organización de forma eficiente de los datos en la base de datos.

En este punto, ya se debería tener claro qué persigue un sistema OLTP, así como el motivo de su diseño. La pregunta ahora es clara: ¿persigue lo mismo un sistema OLAP?

Ya se han descrito las causas que motivaron la aparición de los sistemas OLAP: necesidad de mayor análisis, consultas más complejas y pesadas, mayor volumetría de datos y velocidad de consulta.

i Puesto que en un sistema ROLAP los datos están almacenados en bases de datos relacionales OLTP, es evidente que se debe realizar un proceso de conversión de estas arquitecturas y, de esta forma, optimizar el objetivo principal de todo sistema OLAP: la velocidad de consulta.

Para conseguir esto, no solo se debe disponer/diseñar la base de datos en función de modelos concretos (estrellas y copos de nieve), sino que también hay que **desnormalizar** la información.

CONTINUAR

6.2. Proceso de desnormalización

En definitiva, en el diseño de sistemas OLAP/ROLAP, cuando se diseña una estrella, y una vez definida la tabla de hechos, hay que identificar e implementar todas las dimensiones del modelo. Los datos de las dimensiones deben ser almacenados de forma **desnormalizada** y, puesto que proceden de sistemas OLTP normalizados, es necesario aplicar un proceso de desnormalización, de forma que, como resultado final, se obtenga un modelo de estrella con una tabla de hechos y toda la información de dimensiones esté desnormalizada en tablas.

El proceso en sí es sencillo. Lo más importante es saber el motivo de aplicar este proceso y en qué consiste.

Ventajas

Sus ventajas principales son las siguientes:

- Optimización de la velocidad de las consultas.
- Simplificación del modelo.

Penalizaciones

Es evidente que deshacer el diseño basado en E/R y el proceso de desnormalización genera penalizaciones en otros aspectos:

- Ocupa mayor espacio en disco. Información redundante.
- Mayor complejidad en el mantenimiento de la información. Hay que replicar en varios sitios.
- Se penalizan acciones de inserción, actualización y borrado.
- Se pierde la información del modelo de procesos de la organización que contempla el modelo E/R.

6.3. Tablas de hechos agregadas

Además de la desnormalización de las dimensiones, otra técnica típica que se usa siempre en búsqueda de un mayor rendimiento de las consultas es la generación de tablas agregadas. Estas tablas son copias de las tablas de hechos que se almacenan con un detalle de la información menor. Es decir, se reduce el detalle de la información para reducir el número de filas de las tablas de hechos, y así conseguir un menor tiempo de respuesta.

Ventajas

Su ventaja principal es la siguiente:

- Optimización de la velocidad de las consultas.

Inconvenientes

Sus inconvenientes son los siguientes:

- Modelo más complejo, más tablas de hechos.
- Ocupa mayor espacio en disco. Información redundante en distintas tablas con distinto nivel de almacenamiento.
- Mayor complejidad en el mantenimiento de la información. Hay que replicar en varios sitios.
- Se penalizan acciones de inserción, actualización y borrado.

VII. Lenguajes de consulta analíticos: MDX

Así como las bases de datos relacionales tienen su lenguaje de consulta propio, es lógico pensar que las bases de datos analíticas definan el suyo propio, más acorde con su carácter multidimensional.

En este tema, se introducirá uno de los lenguajes multidimensionales más extendidos: MDX.

7.1. MDX: conceptos básicos

MDX es el estándar definido por Microsoft Analysis Services para realizar consultas OLAP.

A primera vista, puede parecer similar a SQL. Sin embargo, MDX es un lenguaje completamente nuevo. SQL fue diseñado para consultar tablas de datos que son estructuras planas, donde se organizan los datos en filas y columnas. En OLAP, los datos se organizan en torno a múltiples medidas, dimensiones, jerarquías y niveles.

MDX es un idioma que se utiliza para realizar cálculos y análisis en torno a las estructuras OLAP. Además, MDX incluye un amplio conjunto de funciones para realizar análisis estadístico. MDX es puramente para el análisis y la lectura de datos.

Al igual que en SQL, las tablas y columnas son los elementos fundamentales. En MDX son fundamentales las dimensiones, jerarquías y niveles, ya que los sistemas ROLAP utilizan principalmente bases de datos relacionales, como fuentes de datos. En ocasiones, se utilizarán conceptos de SQL para describir la funcionalidad.

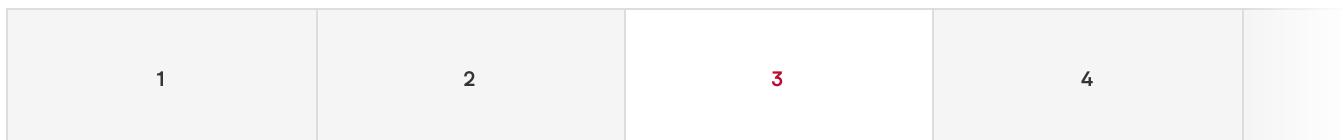
La forma más natural de explicar estos conceptos es con un ejemplo muy sencillo. Partiendo del cubo de ventas definido en la figura 16, se tiene lo siguiente:



Dimensión temporal de las ventas, con niveles de año, trimestre, mes y día (solo se toma una jerarquía).



Dimensión de productos vendidos, con niveles de categoría, marca, gama y producto.



Dimensión de almacén, con niveles de tipo y almacén.



Dimensión de localización, con niveles de país, provincia, ciudad y calle.

1	2	3	4
---	---	---	---

Dimensión de medidas, con importe de las ventas y unidades vendidas.

La primera consulta básica sería la siguiente —

```
SELECT  
{[Medidas].[importe ventas]}  
on columns,  
{[Tiempo].[2008]}  
on rows  
FROM [cubo ventas]
```

Su estructura es la siguiente:

- EjesSets {}
- Filas y columnas
- FROM Cubo
- Where (otras dimensiones, all)

Todas las dimensiones son iguales, incluidas las medidas. El resultado de esta consulta sería el acumulado de todo el importe de ventas para el año 2008. Y es importante destacar que MDX expresa el set de datos de la forma {<set>}.

Si se avanza un poquito más, una segunda consulta sería la siguiente

```
{[Medidas].[importe ventas]}\non columns,\n{[Tiempo].[2008]}\non rows\nFROM [cubo ventas]\nWHERE ([Marca].[NombreMarca])
```

¿Qué devuelve esta consulta? Las ventas acumuladas en 2008, pero que correspondan únicamente a la marca “NombreMarca”.

CONTINUAR

7.2. Expresiones predefinidas en MDX

MDX admite el uso de funciones predefinidas asociadas a cómo está estructurada la información en el modelo multidimensional, permitiendo, de una forma muy simple, referirse a niveles contiguos, distintas jerarquías, *subset* de datos, etc.

Ejemplo

Members, Children, Descendants

{[Tiempo].[Mes].Members}

¿Qué devolverá? Enero, febrero, marzo...

Y {[Tiempo].[2006].Children}? Enero, febrero, marzo.

Hasta ahora solo se ha usado una dimensión por eje, es decir, dos dimensiones. Ahora se va a ver una tupla, como:

- Una combinación de miembros de una o más dimensiones.
- Un miembro multidimensional.

Y se van a generar consultas más apropiadas para un modelo multidimensional:

Ejemplo

```
select NON EMPTY {[Año].[2006], [Measures].[Importe Ventas]},  
([Año].[2007],[Measures].[Número Ventas]) ON COLUMNS,  
NON EMPTY {[Almacén].[Tipo].Children} ON ROWS  
from [Ventas]
```

¿Qué muestra la consulta? El importe de ventas de 2006 y el número de ventas de 2007 por cada uno de los tipos de almacén definidos en el modelo. Se han especificado dos tuplas para las columnas, que añaden una dimensión más a la consulta.

MDX permite simplificar, en gran medida, muchas consultas más complejas, con operadores o expresiones sencillas. Por ejemplo, algo muy usado en MDX es el producto cartesiano (*CrossJoin*). A continuación, se muestra un ejemplo:

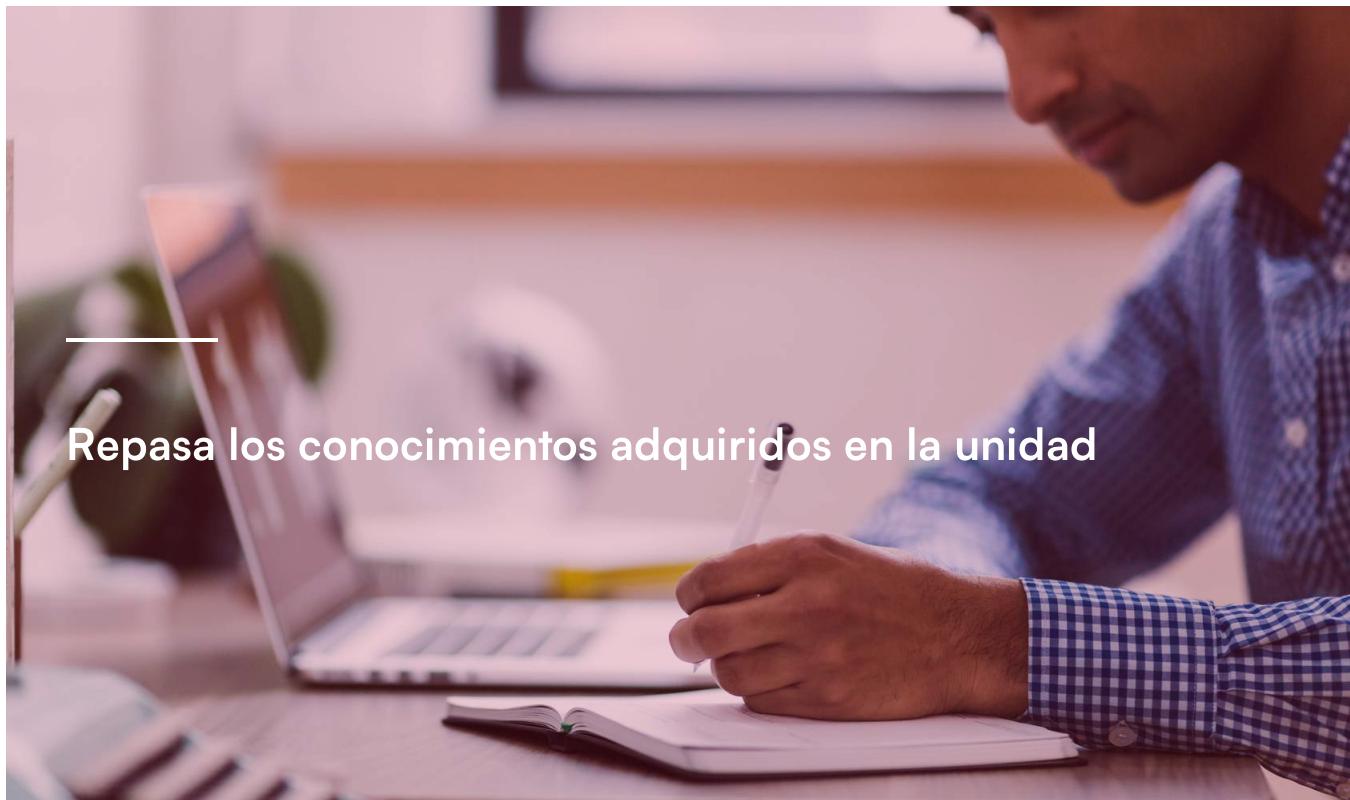
Ejemplo

```
select NON EMPTY Crossjoin({[Año].[2007], [Año].[2008]},  
{[Measures].[Importe Ventas], [Measures].[Número Ventas]}) ON COLUMNS,  
NON EMPTY {[Pais].[España], [Marca].[BMW]}) ON ROWS  
from [Ventas]
```

Muestra el producto cartesiano (no vacío) de los años 2007 y 2008 con las métricas importe de ventas y número de ventas, distribuidos en España para la marca BMW (en filas, no es filtro).

El lenguaje MDX es bastante extenso y arduo de comprender y profundizar en él. No obstante, constituye el core de la comunicación multidimensional y, por tanto, es necesario para cualquier analista tener unas nociones básicas.

VIII. Resumen



Repasa los conocimientos adquiridos en la unidad

En esta unidad se ha hecho una introducción al concepto de almacén de datos o *data warehouse* como motor de una solución de inteligencia de negocio. Se han descrito sus principales cualidades, la causa de su aparición y las principales tecnologías.

Se han especificado las diferentes formas que existen para organizar un sistema analítico y la estructura del almacén de datos, repasando las principales corrientes definidas por Ralph Kimball y William H. Inmon, quienes defendían sistemas totalmente opuestos, desde el divide y vencerás de Kimball hasta los proyectos de inteligencia de negocio faraónicos de Inmon.

La eterna duda en este tipo de sistemas siempre es si realizar un *data warehouse* corporativo o una serie de *data marts* que se intercomuniquen entre ellos. O incluso tener ambos sistemas a la vez. Lo que está comprobado es que las opciones de afrontar proyectos más pequeños y sencillos, aportando un valor al sistema analítico de forma incremental, suelen tener mayor éxito que los procesos de desarrollo muy grandes.

La arquitectura típica de un sistema de inteligencia de negocio constará de cuatro capas, generalmente. Entre ellas están los orígenes de datos, que se almacenarán en una base de datos inicial denominada *staging*. Esta primera base de datos es una copia de los datos de origen en bruto.

Los datos en bruto se procesan para poderlos integrar en el área de explotación, que puede ser un *data warehouse*, varios *data marts* o un cubo OLAP. Estos almacenes serán explotados por usuarios y herramientas analíticas.

Nunca hay que perder de vista las tareas que afectan a las buenas prácticas, ya que garantizan una buena escalabilidad y mantenimiento, claves para el éxito a largo plazo del proyecto.

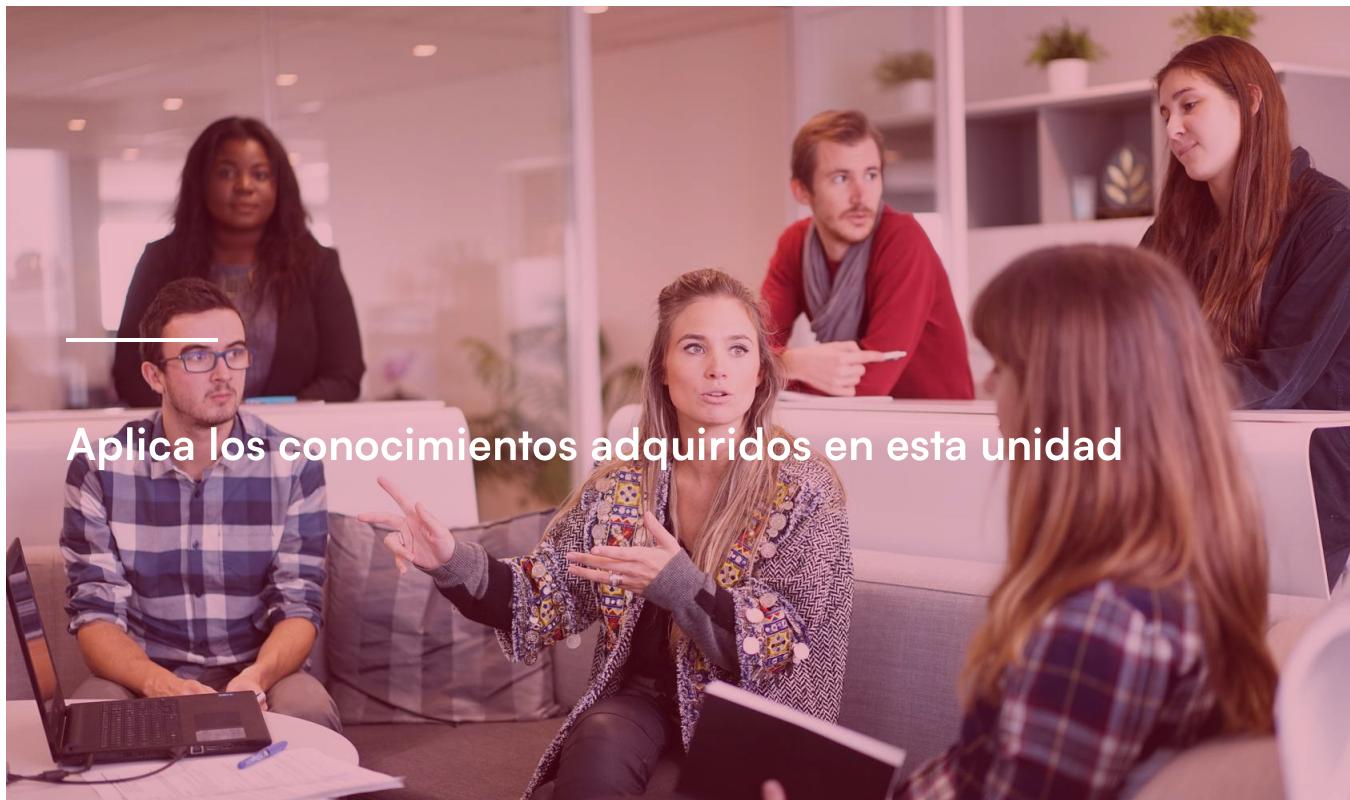
De igual forma, se ha hecho un repaso de los principales conceptos asociados al almacén de datos y *data mart*, destacando la multidimensionalidad y el análisis multidimensional que habilita, así como dimensiones, hechos, atributos, métricas y procesos de desnormalización.

Se han especificado los tipos de almacenes OLAP que existen como ROLAP (*relational online analytical processing*), que permite realizar análisis multidimensional dinámico a partir de los datos almacenados en una base de datos relacional. También se definieron los sistemas MOLAP, que son sistemas multidimensionales propiamente dichos, es decir, aquellos que implementan un almacenamiento de datos multidimensional. Y, por último, los sistemas HOLAP, son los sistemas OLAP híbridos, que no son más que la combinación de sistemas ROLAP y MOLAP.

También se ha repasado el concepto de almacén de datos analíticos, tanto multidimensional como propio del mundo del *data warehouse*. Con ello se detallaron conceptos como la desnormalización y la gestión de tablas agregadas en la búsqueda de un mejor rendimiento a la hora de solucionar las consultas de datos.

La última parte de la unidad se ha dedicado a introducir el lenguaje de consulta analítico MDX. Como ya se ha comentado, es bastante extenso, pero básico y necesario para cualquier analista, por lo que se anima a todos los alumnos a profundizar un poco más en él.

IX. Caso práctico con solución



ENUNCIADO

Ya se ha comentado la necesidad de generar un modelo multidimensional para satisfacer las exigencias de análisis actuales por parte de las organizaciones. Este modelo permite realizar cualquier tipo de análisis de la información, de forma dinámica y ágil, sin penalización de velocidad. Un modelo multidimensional identifica, como se ha visto previamente, dimensiones, métricas, atributos y hechos.

Pero ¿cómo se diseña?

Se partirá de un sistema ROLAP. En una arquitectura MOLAP, una vez identificados dimensiones, hechos, métricas y atributos, se precalculan y se cargan físicamente de forma multidimensional. Sin embargo, no ocurre así en ROLAP, donde los datos están almacenados físicamente en una base de datos relacional. Este hecho requiere que los datos se almacenen en unas estructuras concretas, aunque sea sobre una base de datos relacional.

La justificación de esto es clara: el modelo multidimensional y sus operaciones deben ser mapeados a relaciones —como en toda base de datos relacional—, para que el lenguaje específico multidimensional (MDX) y sus consultas puedan, a su vez, ser mapeados a consultas basadas en SQL.

A continuación, se aplicará lo comentado en un ejemplo para fijar los conceptos.

SE PIDE

Una organización dispone de una entrada de datos con las siguientes columnas de información:

- Producto
- Marca
- Gama
- Categoría
- Nombre proveedor
- Tipo proveedor
- Día
- Semana

- Mes
- Trimestre
- Año
- Almacén
- Tipo de almacén
- Calle
- Ciudad
- Provincia
- País
- Unidades
- Importe

Partiendo de esta información, plantear el modelo multidimensional (modelo lógico) de ventas, definiendo lo siguiente:

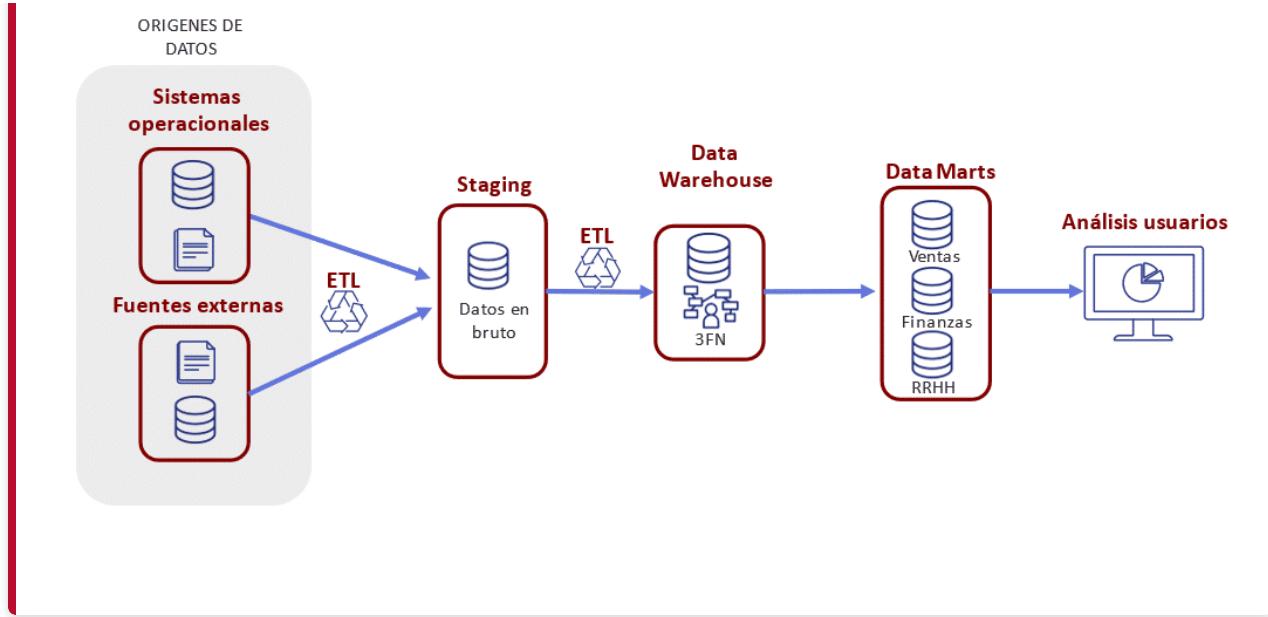
1. Identificación de dimensiones.
2. Jerarquías y niveles de jerarquía.
3. Métricas y atributos.

De la misma forma, y partiendo ahora de la figura que se muestra a continuación, definir los modelos físicos que permitan la implementación del modelo anterior, particularizando para ROLAP. En concreto:

- Modelo de estrella.
- Modelo copo de nieve.

Figura 3. Data warehouse de Inmon.

Fuente: elaboración propia.



[VER SOLUCIÓN](#)

SOLUCIÓN

Anteriormente, se vio un ejemplo de modelo multidimensional en el que se quería analizar el hecho ventas.

Figura 10. Ejemplo de modelo en estrella.

Fuente: elaboración propia.

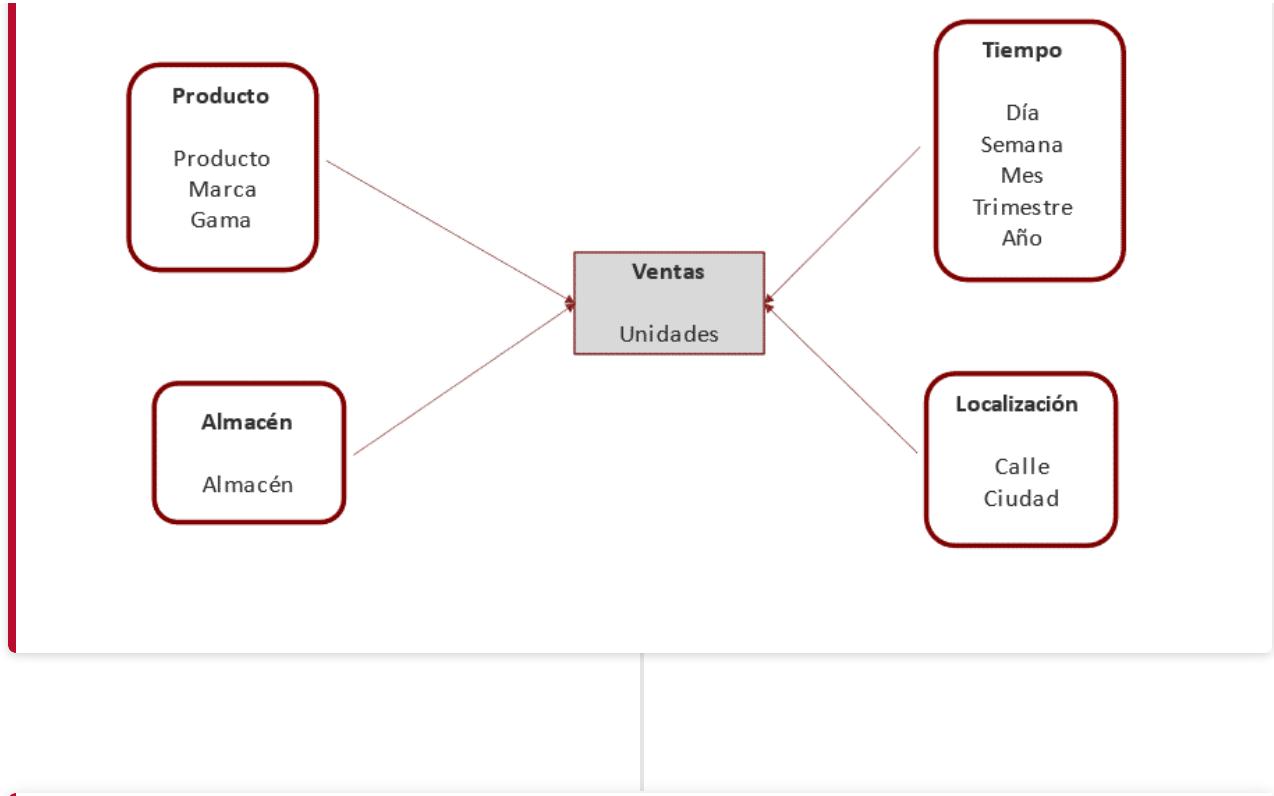


Figura 17. Ejemplo de esquema.

Fuente: elaboración propia.



De la misma manera, se deben identificar cuatro dimensiones:

- Producto, tiempo, almacén y localización.

Los hechos son las ventas, tal y como transmite el enunciado.

Las métricas directas que se pueden definir, como mínimo, son las siguientes:

- Unidades vendidas, cuya función de agregación sería la suma.
- Importe de ventas, cuya función de agregación sería la suma.

Lo primero que se debe introducir es el concepto de **jerarquización de la información**. Los almacenes de datos están preparados para contener grandes cantidades de datos, de granularidad pequeña, capacitados para cualquier tipo de análisis y obtener información agregada y resumida del modelo de negocio. Todo esto desemboca en la necesidad de la jerarquización de la información, que no es más que la definición de una estructura jerárquica dentro de los datos que permite navegar a través de ellos.

El ejemplo claro sería la jerarquización de la dimensión “tiempo”. Dentro de la dimensión temporal, se definen tres niveles:

- Año.
- Mes.
- Día.

La jerarquía de estos niveles es fácil de entender. El nivel superior, “año”, engloba meses y días. El nivel “mes” engloba días, y el nivel “día” define la granularidad de la dimensión temporal. Como máximo detalle, se analizará la dimensión a nivel de “día”.

A cada una de estas estructuraciones de datos se le llama **jerarquía de una dimensión**, pudiendo tener la misma dimensión varias jerarquías.

En el ejemplo que se comentaba anteriormente, se pueden ver las dimensiones con sus jerarquías y comprobar que la dimensión “tiempo” posee dos jerarquías:

- Producto: Categoría -> Gama -> Marca -> Producto.
- Almacén: Tipo -> Almacén.
- Localización: País-> Provincia -> Ciudad -> Calle.
- Tiempo: Año -> Trimestre -> Mes->
- Tiempo: Mes -> Semana -> Día.

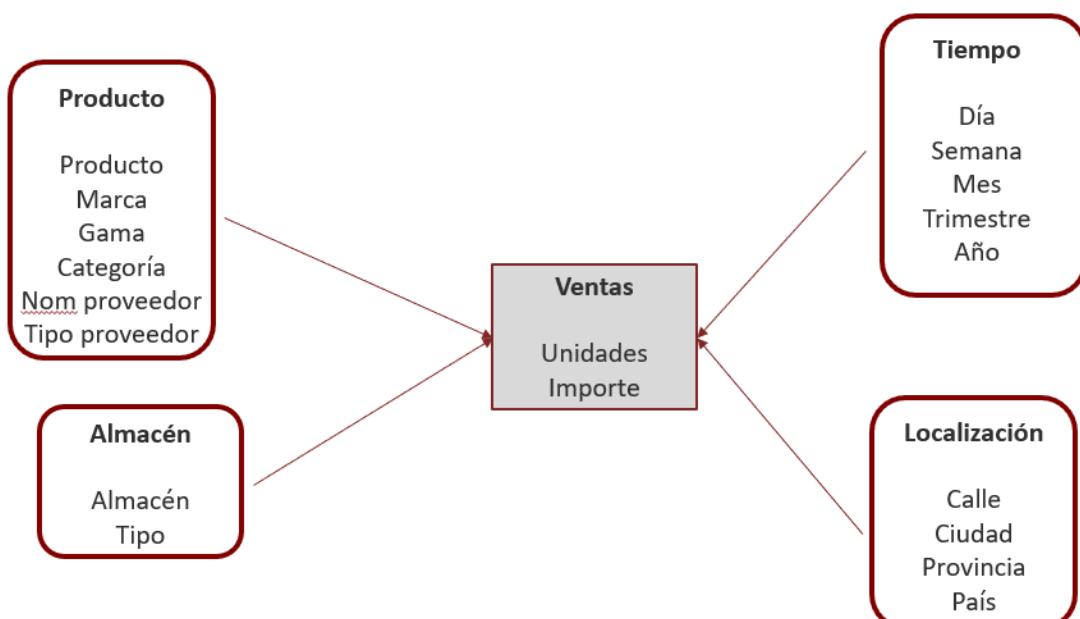
Pero esta estructura de la información debe reflejarse en la base de datos relacional que conforma la base del sistema ROLAP.

Para saber cómo implementarlo, se atenderá a la arquitectura de datos en ROLAP.

Se particularizará para sistema ROLAP. Capa física.

Si se deseara representar el modelo de ventas visto anteriormente, su modelo de estrella sería el

Para poder representar el modelo de ventas visto anteriormente, su modelo de estrella sería el siguiente:

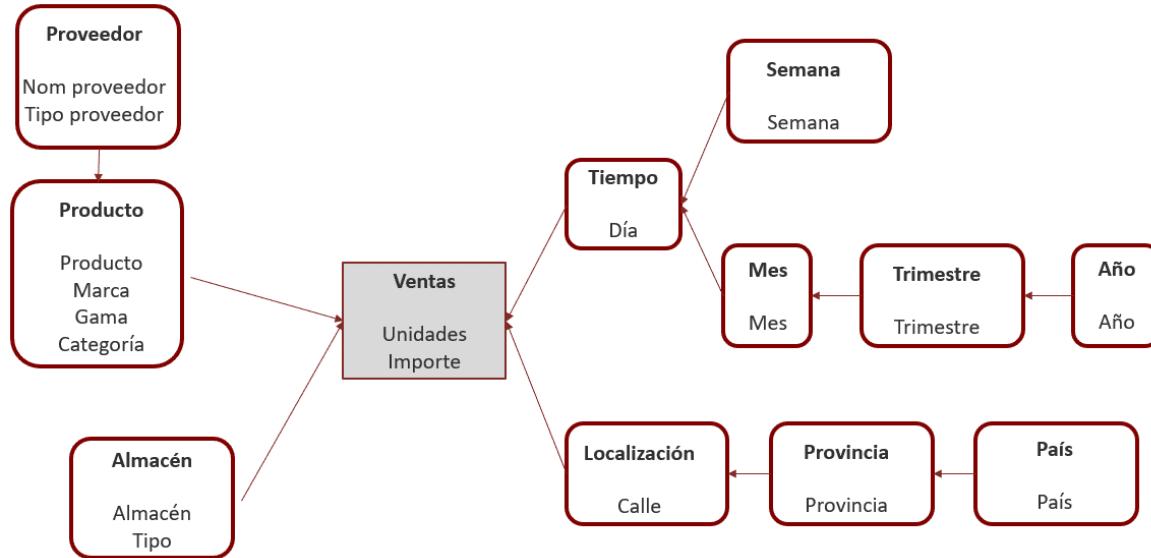


En este modelo, se identifican dos estructuras muy importantes:

- **Dimensiones o ejes de análisis:** cada una de las tablas situadas alrededor de la **tabla de hechos**.

Se llama estrella por su parecido físico a una estrella real, debido a que la tabla de hechos está en el centro y es más grande que las tablas más pequeñas que lo rodean.

El mismo modelo de ventas anterior, en arquitectura copo de nieve, sería el siguiente:



X. Glosario



El glosario contiene términos destacados para la comprensión de la unidad

Almacén de datos o data warehouse (DW) —

Colección de información creada como soporte de las aplicaciones que ayudan en la toma de decisiones de las empresas. Los almacenes de datos se representan habitualmente como una gran base de datos.

Data marts

Almacenes de datos departamentales. Son más pequeños que los almacenes de datos y serán dependientes de estos.

Granularidad

El nivel fundamental, atómico, de los datos que representar en la tabla de hechos.

MDX

Lenguaje de consulta analítico, uno de los lenguajes multidimensionales más extendidos. Es el estándar definido por Microsoft Analysis Services para realizar consultas OLAP.

MOLAP (multidimensional online analytical processing)

Sistemas OLAP multidimensionales propiamente dichos, es decir, aquellos que implementan un almacenamiento de datos multidimensional para proporcionar el análisis analítico.

OLAP (online analytical processing)

Los sistemas OLAP son bases de datos orientadas al procesamiento analítico.

ROLAP (relational online analytical processing)

Permite realizar análisis multidimensionales dinámicos, a partir de los datos almacenados en una base de datos relacional.

Tabla de hechos

Contiene todos los datos que son relevantes para la unidad de negocio que se desea analizar.