

Base de Datos Avanzadas

Bases de Datos para
Inteligencia de Negocio (BI)

Resumen

- ❑ **Contexto** → BI, SSD, BD para SD
- ❑ **Tecnologías Asociadas**
- ❑ **Datawarehouse**
- ❑ **Data-Mart**
- ❑ **Data-Mining**

Inteligencia de Negocio

□ **Inteligencia de Negocios**

- Insumo → Datos / Información
- Producido → Conocimiento.
- Propósito de este Proceso → Ventaja Competitiva,
→ Acciones Concretas

□ **Proceso**



Inteligencia de Negocio

- Se denomina **inteligencia de negocios**, al conjunto de **estrategias y aspectos relevantes enfocados a la administración y creación de conocimiento** sobre el medio, a través del análisis de los **datos** existentes en una organización o empresa, con el **fin obtener ventajas competitivas**.
- Es una arquitectura y un **conjunto integrado** de aplicaciones operacionales, aplicaciones soporte de decisiones (SSD) y bases de datos (BD para SD) que proveen **fácil acceso a los datos del negocio**.

Proceso de Decisión

- ❑ **SSD** atacan una problemática específica y las BD se estructuran de manera diferente.
- ❑ **Procesos de Decisión**
Es un proceso que implica elegir entre varias alternativas o cursos de acción, con el propósito de alcanzar uno o varios objetivos definidos
- ❑ No solo implica **resolver problemas** sino también **investigar oportunidades** de negocio, maximizar resultados.

Proceso de Decisión

Taxonomía de Simon, clasifica a los procesos de decisión en un intervalo continuo entre 2 extremos

❑ **Procesos de Decisión** estructurados
(100 % Automatizables)



❑ **Semiestructurados** franja intermedia

❑ **Procesos de Decisión** NO estructurados
(no son automatizables)



Proceso de Decisión:

- ❑ **No estructurados:**

NO tienen fases estructuradas/automatizables

Se resuelven con la intuición humana



- ❑ **Semi-estructurados:** tienen alguna (o partes de) fase estructurada

Se resuelven combinando procedimientos estándares y juicio humano



- ❑ **Estructurados:** *todas las fases son estructuradas*

Se conocen los procedimientos para obtener la mejor solución



Soporte Informático

para Procesos de decisión Estructurados

Procesos:

- ☐ Repetitivos
- ☐ Alto nivel de automatización
- ☐ Es posible abstraerlos y clasificarlos en "tipos"
- ☐ Se resuelven con fórmulas y modelos cuantitativos



Soporte Informático:

- ☐ Evolución desde 1960, implementan algoritmos desarrollados por disciplinas denominadas **Management Science** y **Operations Research**

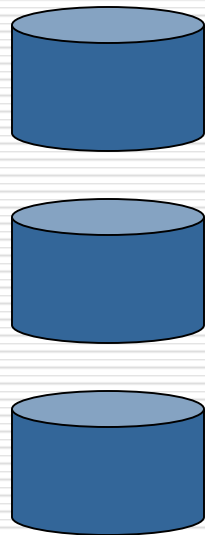
Soporte Informático

para Procesos de decisión No Estructurados y Semi-Estructurados

Sistemas Soporte de Decisiones (SSD)

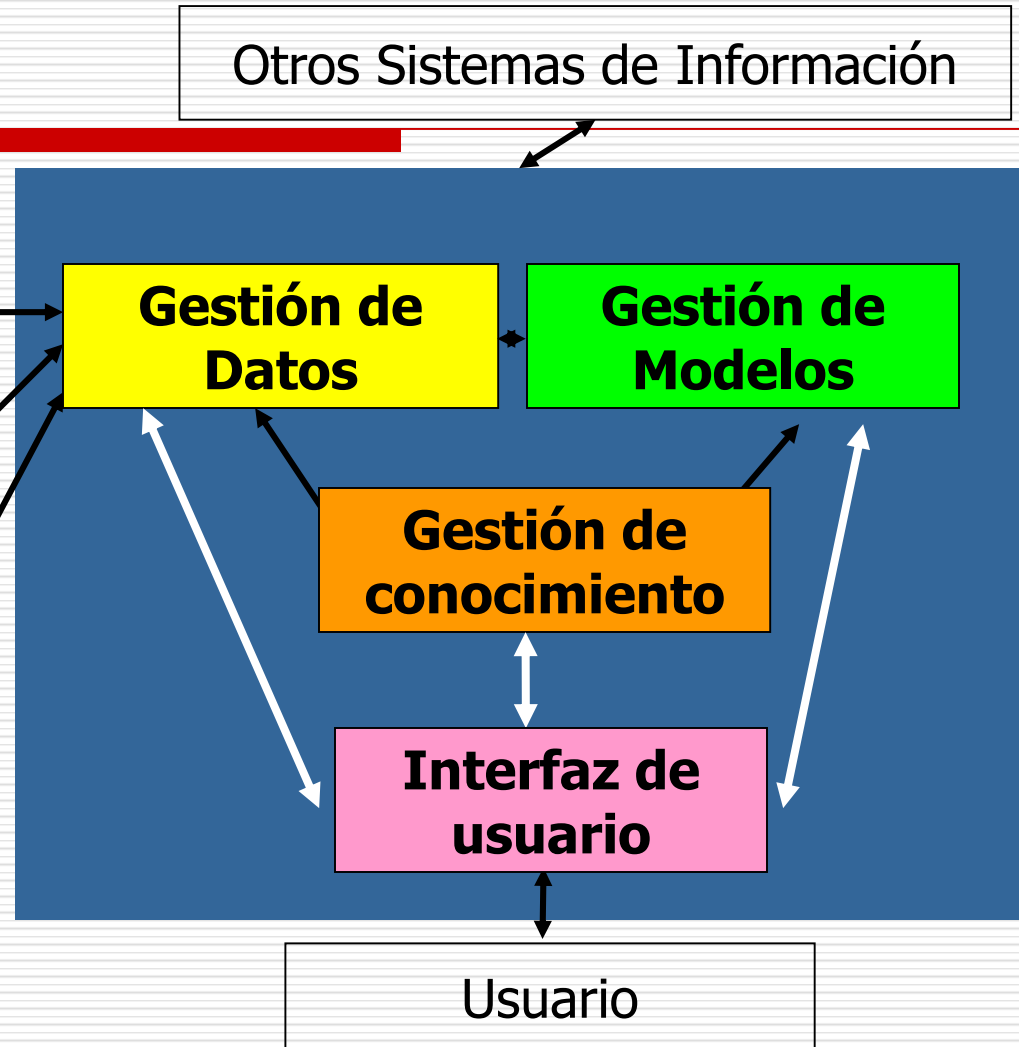
- ❑ Son sistemas de información **interactivos** que **ayudan** al tomador de decisiones a utilizar **datos** y **modelos** para resolver **problemas de decisión no estructurados o semi estructurados**
- ❑ Un SSD no tiene capacidad para resolver problemas por si solo
- ❑ Propósito: ayudar al decisor; NO reemplazarlo

Base de Datos para Soporte de Decisiones



Datos: Externos e internos

Sistemas Soporte de Decisiones



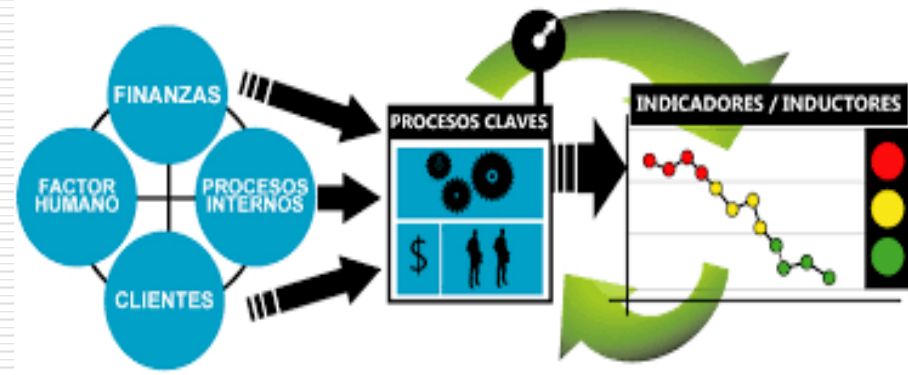
Clasificación de los SSD

□ **Sistemas Soporte de Decisiones**

- Soporte para análisis multidimensional (OLAP)
- Soporte para pronósticos
- Soporte para Balance Scorecard (tablero comando)
- Soporte para procesos de MINING
- Soporte para Gestión de conocimiento

Tableros de Comandos

- Son aplicaciones que provienen del campo de la **administración**, aplicable a cualquier organización y nivel de la misma, cuyo objetivo y utilidad básica es diagnosticar una situación particular de riesgo u oportunidad.



- Se le define como el conjunto de indicadores cuyo seguimiento y evaluación periódica permitirá contar con un mayor conocimiento de la situación de su empresa o sector apoyándose en nuevas TI de toma de decisión.

Repositorios para SSD

☐ **Base de Datos para Soporte de Decisiones**

- Data Marts (Departamentales)
- Data Warehouse
- BD para Data Mining

OLTP - OLAP

- ❑ **OLTP (On-Line Transaction Processing): Define el comportamiento habitual de un entorno operacional de gestión:**
 - Altas/Bajas/Modificaciones/Consultas
 - Consultas estructuradas, rápidas y directas.
 - Poco volumen de información
 - Transacciones de negocio, Tiempo Real
 - Gran nivel de concurrencia

OLTP- OLAP

- ❑ **OLAP: On-Line Analytical Process:**
Define el comportamiento de un sistema de análisis de datos y elaboración de información:
 - Sólo Consulta, Operación Lenta
 - Consultas pesadas y no predecibles
 - Gran volumen de información histórica
 - No hay Transacciones de negocio, No son Tiempo real

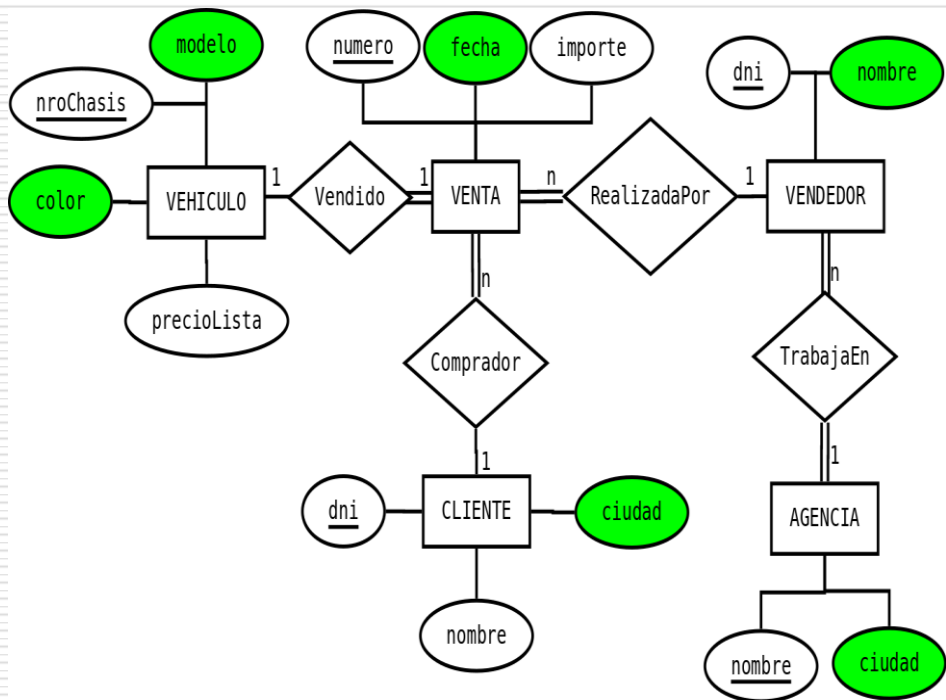
OLTP - OLAP

OLAP representa la **vista multidimensional** de datos de la organización.

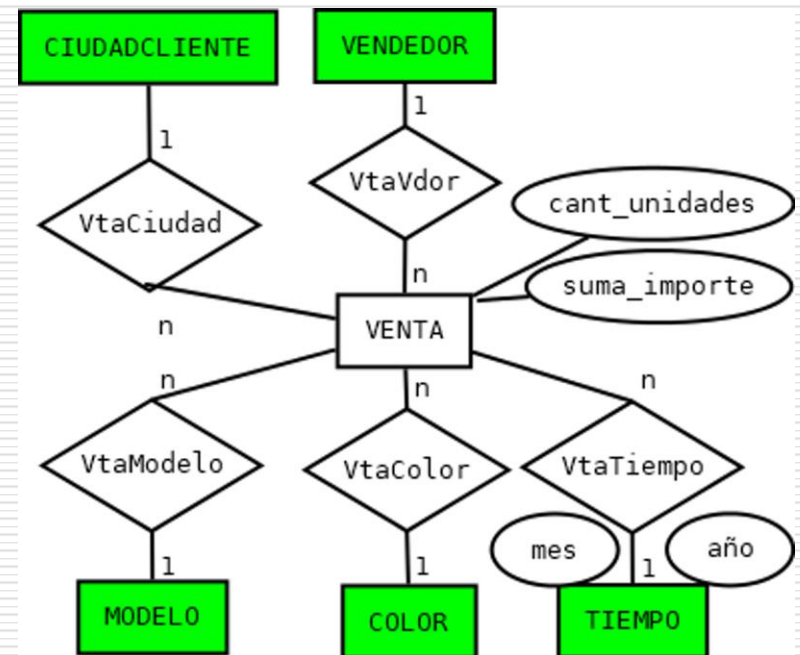
- Para que un dato pueda ser visto desde múltiples dimensiones, es necesario que sea **multidimensional**.
-

Idea de ambos diseños

OLTP



OLAP



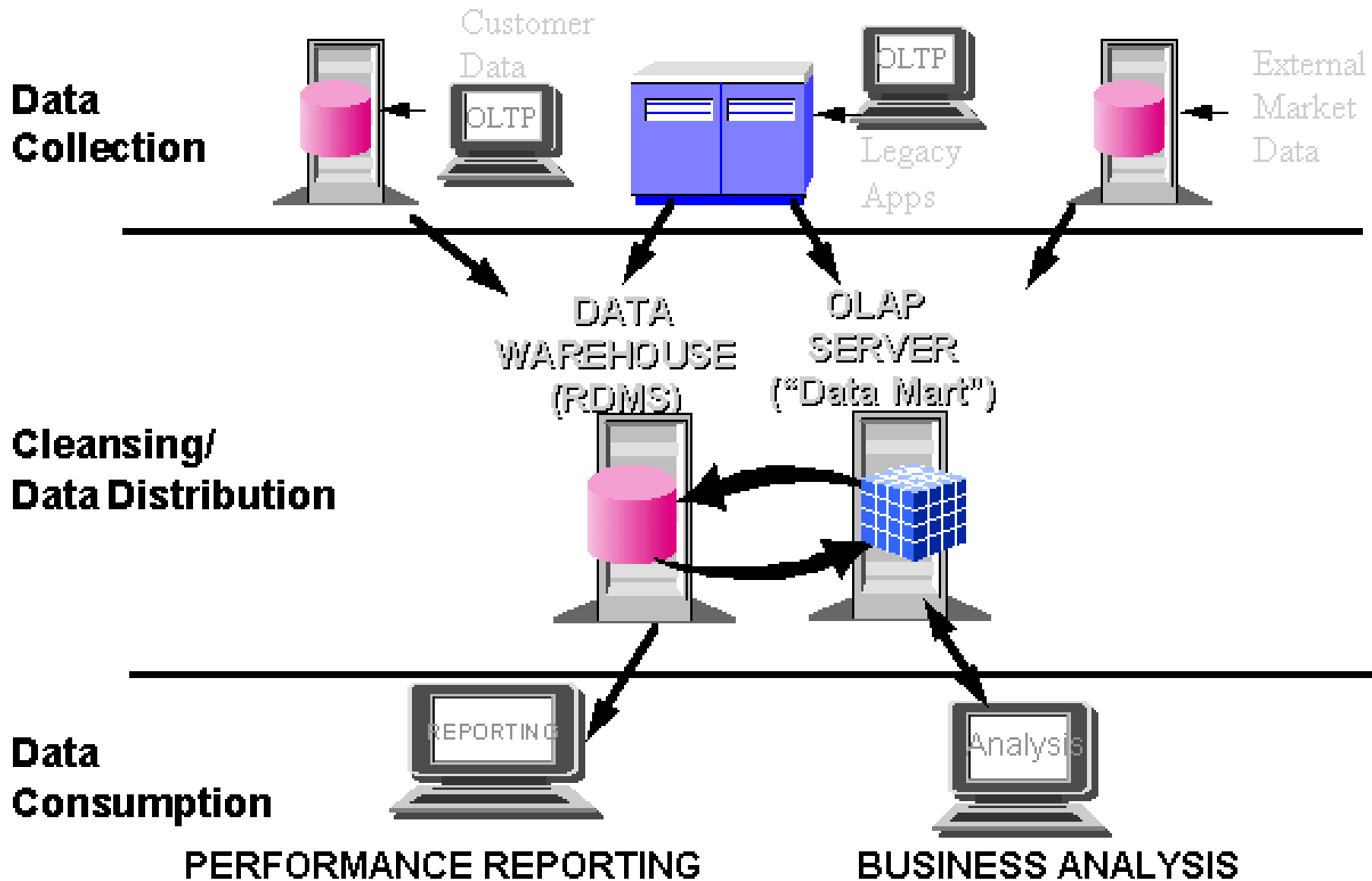
Comparativa OLTP - OLAP

Característica	OLTP	OLAP
Tamaño BD	GigaBytes	Giga a TeraBytes
Origen Datos	Interno	Interno y Externo
Actualización	On-Line	Batch
Estado/Periodos	Actual	Histórico
Consultas	Predecibles	Ad Hoc
Actividad	Operacional	Analítica

OLTP - OLAP

- Todas estas divergencias hacen que sea poco posible la convivencia en una única BD de los entornos OLAP y OLTP:
 - Pérdida de rendimiento del entorno OLTP
 - Falta de integración entre distintas aplicaciones OLTP
 - Tecnologías de BDs sin capacidad para soportar aplicaciones OLAP y OLTP a la vez.
 - Incorporación de datos externos difícilmente aplicable a la BDs OLTP
 - Organización de los datos no adecuada para análisis sobre bases OLTP

OLTP/OLAP Enterprise I.T. Architecture



DW – DM - Minería

- ❑ **Data warehouse:** Repositorio completo de datos de la empresa, donde se almacenan datos **estratégicos**, tácticos y operativos, con el objeto de obtener información estratégica y táctica
- ❑ **Data Mart:** Repositorio parcial de datos de la empresa, donde se almacenan datos **tácticos** y operativos, al objeto de obtener información táctica
- ❑ **Data Mining:** Técnicas de análisis de datos encaminadas a obtener información oculta en un Datawarehouse y en la BD OLTP

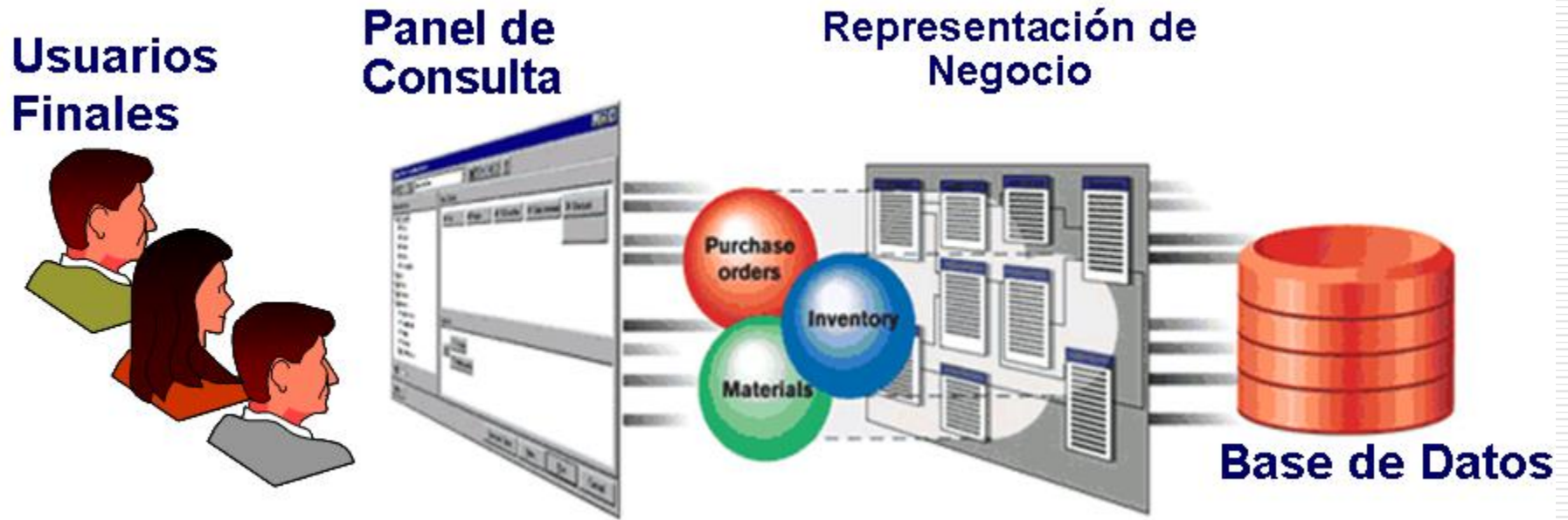
Un Data Warehouse es ...

- ❑ ... un modelo de datos da soporte a decisiones representando la información que una compañía necesita para tomar **DECISIONES ESTRATEGICAS**.
- ❑ ... basado gralmente en la estructura de un sistema de gestión de base de datos relacional el cual puede ser usado para **INTER-RELACIONAR** los datos contenidos en él.
- ❑ ... con el propósito de proporcionar a los usuarios finales un acceso **SENCILLO** a la información.

Ejemplo

- Ejemplo
 - Organización: Cadena de supermercados
 - Actividad objeto de análisis: ventas de productos
 - Objetivo: aumentar ventas con publicidad adecuada
- Problema 1: No necesitamos TODOS los datos de la BD, en OLTP están en formato no adecuado, y falta historia.
- Problema 2: Fuentes de datos diversas (BDs diferentes, archivos de texto, Planillas de Calculo, archivos XML...)
- Problema 3: Fuentes de datos externas
- Problema 4: Demasiados datos
- Problema 5: Análisis ON - LINE

Visión del Usuario



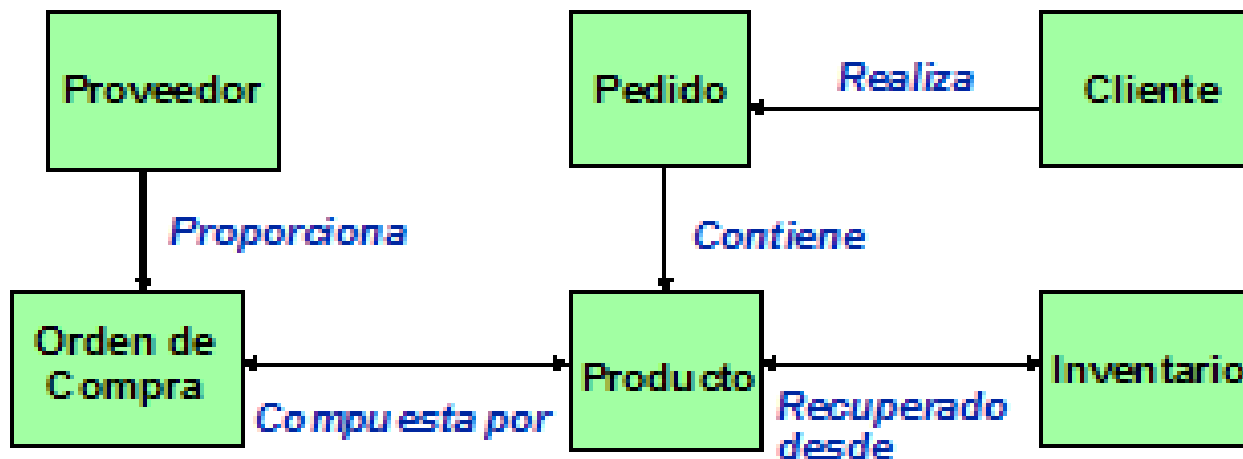
- ☐ Solución integrada de: Consultas, informes y análisis.
- ☐ Capa semántica que da una representación de los datos desde el punto de vista de negocio.
- ☐ Los usuarios utilizan términos de negocio, no términos informáticos.

Características Data Warehouse

- ❑ Orientado a un Tema
 - Colección de información relacionada organizada alrededor de un tema central
- ❑ Integrado
 - Datos de múltiples orígenes; importante garantizar la consistencia de datos
- ❑ Variable en el tiempo
 - 'Fotos' en el tiempo
 - Basado en fechas/periodos
- ❑ No-volátil
 - Sólo lectura para usuarios finales
- ❑ Menos frecuencia de cambios/actualizaciones
 - Usado para el Soporte a Decisiones y Análisis de Negocio

Orientado a Tema

Los usuarios piensan en términos de 'cosas' y sus relaciones', no en términos de procesos, funciones o aplicaciones.



Integrado

□ Contiene

- Convenciones de Nombres
- Descripciones
- Atributos físicos de los datos
- Valores de los datos

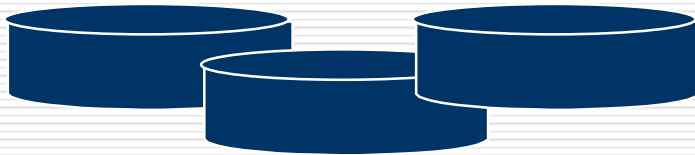
Consistentes



Variable en el tiempo

□ Entorno Operacional

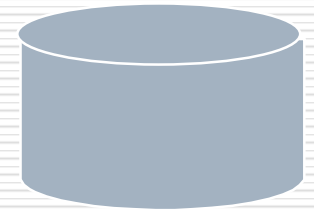
- *Datos con valores actuales*
- Horizonte de 30 - 90 días
- Exactitud en los accesos



Id de cliente
nombre
dirección
teléfono
ratio de crédito

□ Data Warehouse

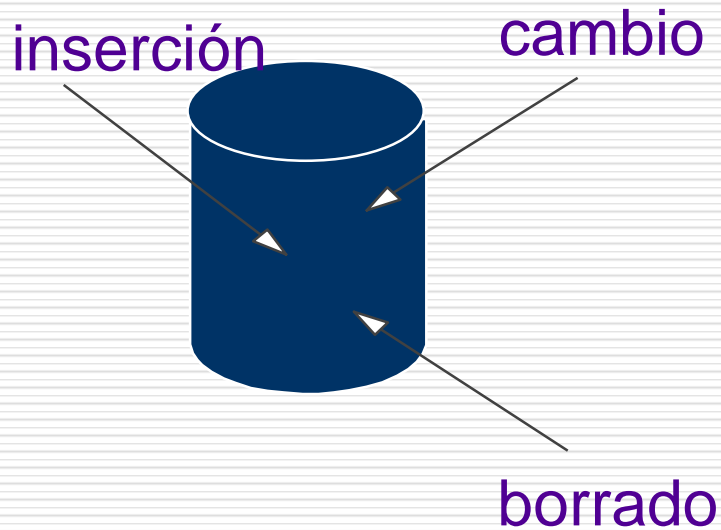
- *Datos en 'fotos'*
- Horizonte de 5, 10 o mas años
- Refleja la perspectiva desde un momento en el tiempo



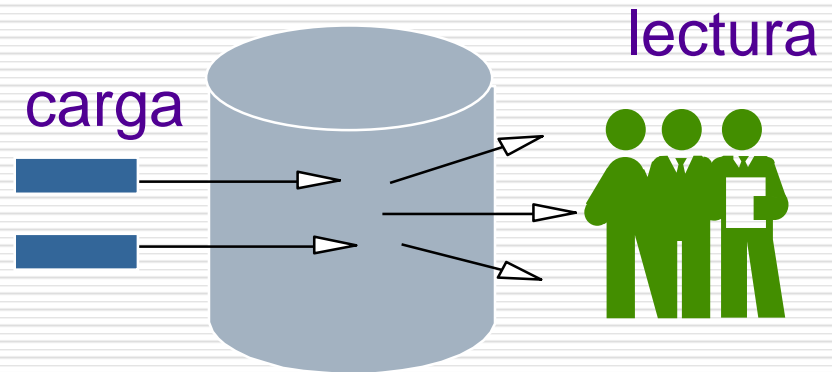
Id de cliente
fecha desde
fecha hasta
nombre
dirección
teléfono
ratio de crédito

Volátil

No-Volátil



Sistema OLTP
(dinámico)



Sistema SSD
(más estático / periodico)

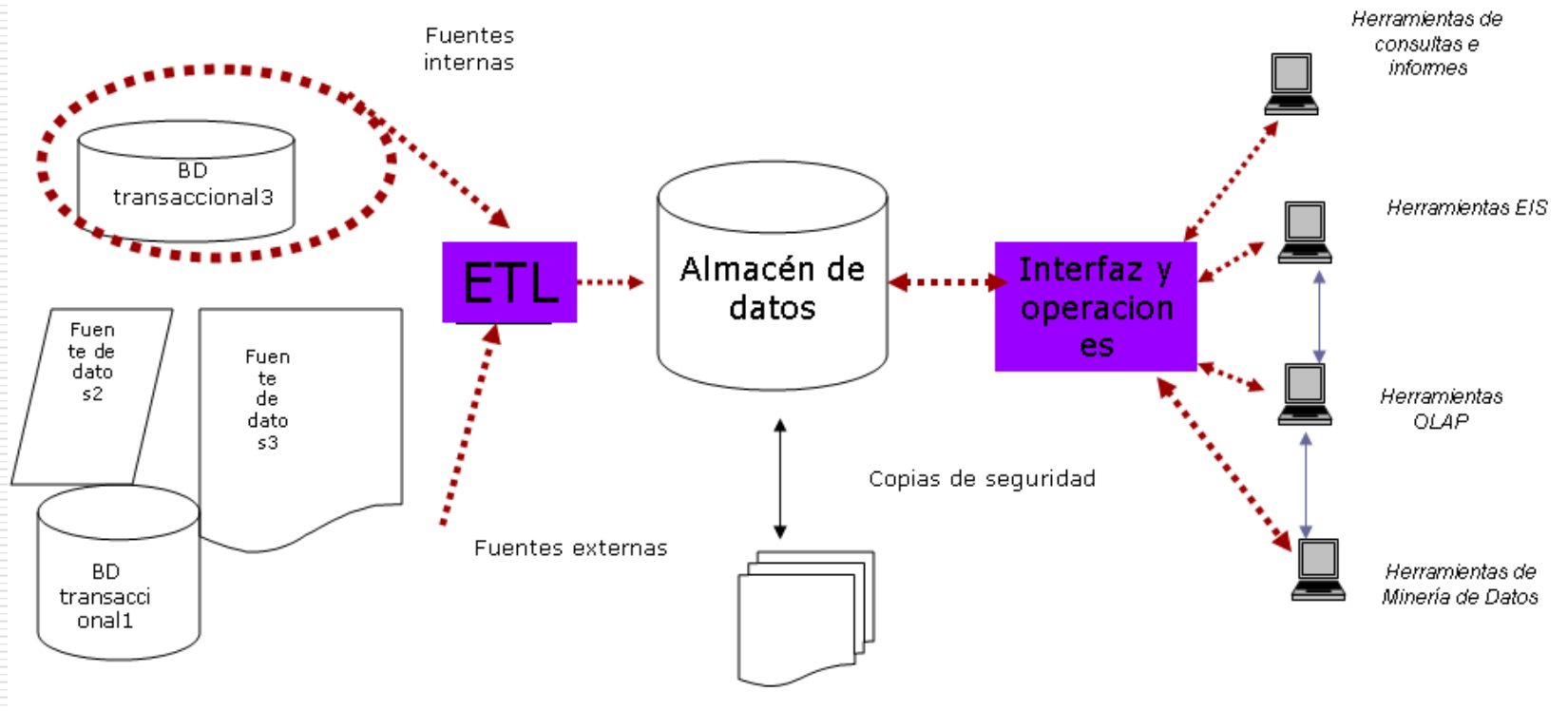
Nomenclatura Asociada a un DWH

□ Nomenclatura

- DWH: Data Warehouse, Bodega de Datos, Almacen de Datos
- DM: DataMart
- OLTP: Procesamiento transaccional en línea
- OLAP: Procesamiento analítico en línea
- ROLAP: Procesamiento analítico relacional en línea
- MOLAP: Procesamiento analítico multidimensional en línea
- ODS: Almacenamiento operacional de datos
- DSS: Sistema de soporte de decisión
- ETL: Extracción, Transformación y carga
- ETQL: Extracción, Transformación, calidad y carga
- EII: Integración de información empresarial
- EAI: Integración de aplicaciones empresarial
- EIS: Sistema de información ejecutiva
- ERP: Planificación de recursos empresariales
- CRM: Administración de Relaciones con Clientes

Arquitectura de un DWH

- La arquitectura de un Almacén de datos AD viene determinada por su situación central como fuente de información para las herramientas de análisis.



Data Mart vs. Datawarehouse

- Los DM son subconjuntos de datos de un DWH para áreas específicas
- Entre las características de un Data Mart destacan:
 - · Usuarios limitados.
 - · Área específica.
 - · Tiene un propósito específico.
 - · Tiene una función de apoyo

Data Mart

- ❑ Según define **Meta Group**, *"un Data Mart es una aplicación de Data Warehouse, construida rápidamente para soportar una línea de negocio simple"*.
- ❑ Los **Data Marts**, tienen las **mismas características** de integración, orientación temática y no volatilidad que el **Data Warehouse**.
- ❑ Representan una estrategia de **"divide y vencerás"** para ámbitos muy genéricos de un Data Warehouse.

Data Mart

- ❑ Esta estrategia es particularmente apropiada cuando el Data Warehouse central crece muy rápidamente y los departamentos requieren sólo una pequeña porción de los datos contenidos en él.
- ❑ La creación de estos Data Marts requiere algo más que una simple réplica de los datos: se necesitarán tanto la **segmentación** como algunos métodos adicionales de **consolidación**

Ejemplo Típico en organizaciones

- ❑ Es Habitual que se comience organizando Data Marts para dar soporte a decisiones de Sectores o Aéreas.
- ❑ Luego se presentan situaciones que ameritan organizar estos y crear un Data Warehouse centralizando algunas funciones.
- ❑ Veamos un caso de estudio para ...

Caso de Hipotético Ejemplo

Escenario de Estructura Descentralizada de un Data Mart

- ❑ El departamento de **Marketing**, emprende el primer proyecto de **Data Warehouse** como una solución departamental, creando el **primer Data Mart** de la empresa.
- ❑ Visto el éxito del proyecto, otros departamentos, como el de Riesgos, o el Financiero se lanzan a crear sus **Data Marts**. Marketing, comienza a usar otros datos que también usan los **Data Marts** de Riesgos y Financiero, y estos hacen lo propio.
- ❑ Esto parece ser una decisión normal, puesto que las necesidades de información de todos los **Data Marts** crecen conforme el tiempo avanza. Cuando esta situación evoluciona, el esquema general de integración entre los **Data Marts** pasa a ser, la del gráfico

Escenario de Estructura descentralizada de Data Marts

Almacenes



Inventarios



Envios



Compras



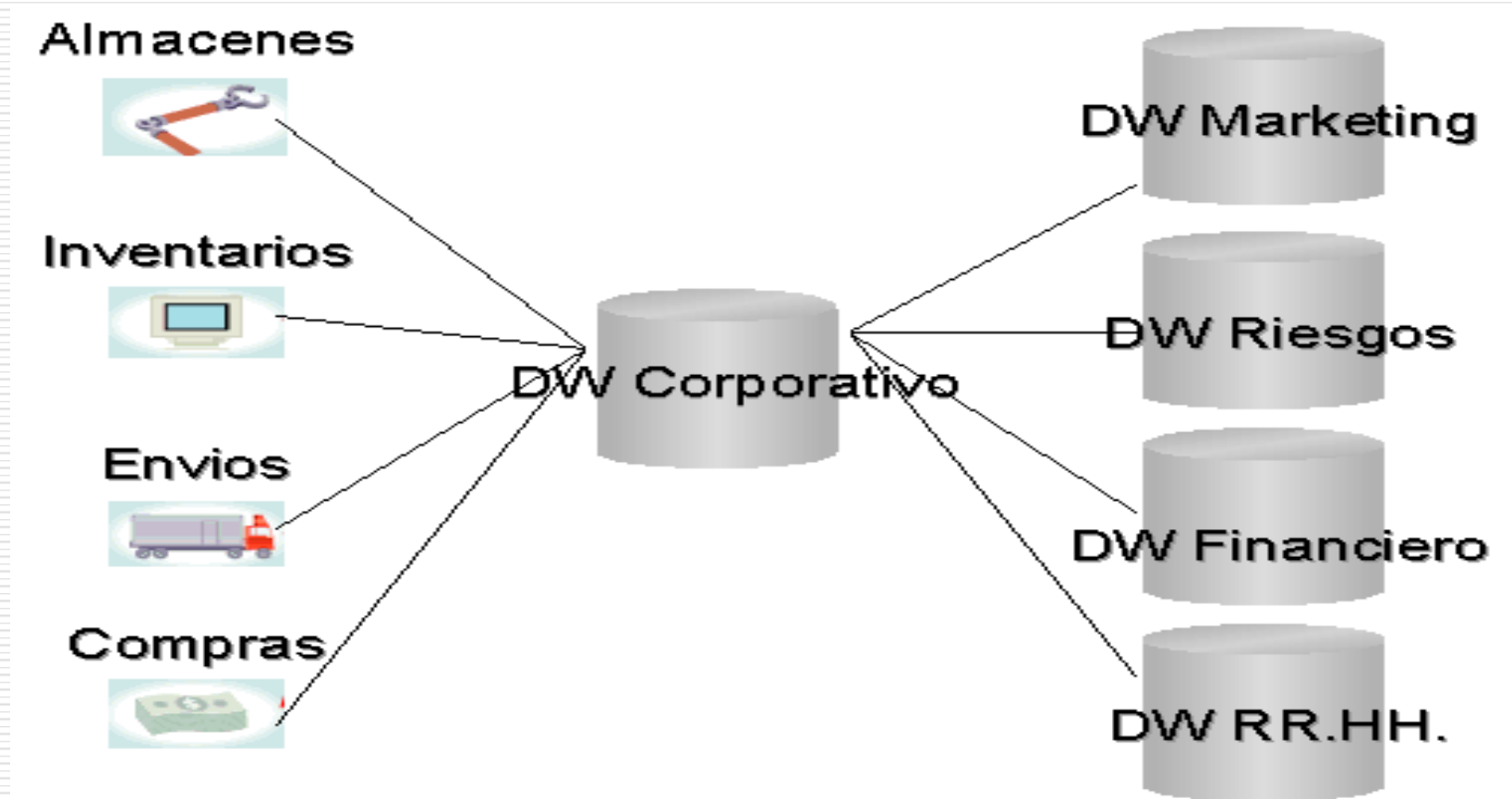
DW Marketing

DW Riesgos

DW Financiero

DW RR.HH.

Coordinación de la gestión de información de todos los Data Marts en un Data Warehouse centralizado



Coordinación de la gestión de información de todos los Data Marts en un Data Warehouse centralizado

- ❑ En esta situación los Data Marts obtendrían la información necesaria, ya previamente cargada y depurada en el Data Warehouse corporativo, simplificando el crecimiento de una base de conocimientos a nivel de toda la empresa.
- ❑ Esta simplificación provendría de la centralización de las labores de gestión de los Data Marts, en el Data Warehouse corporativo, generando economías de escala en la gestión de los Data Marts implicados.

FASES DE IMPLANTACIÓN DE UN DATA WAREHOUSE

El diseño del Data Warehouse

Cuatro características clave del Data Warehouse

1. Las evoluciones tecnológicas
2. La vinculación implícita con la estrategia de la empresa
3. Una lógica de mejora continua
4. Un nivel de madurez diferente según las empresas

1-Evoluciones tecnológicas

- ❑ El cliente/servidor y los sistemas abiertos son tecnologías implícitamente utilizadas en los sistemas DWH.
- ❑ En el ámbito de las metodologías y las técnicas de implementación es posible elegir uno o más métodos:
 - Merise (Método integrado de concepción y análisis de sistemas, Ingeniería de la información)
 - NIAM (método de análisis de información)
 - JAD/RAD (Metodologías ágiles), métodos orientados a objetos.

2-Vinculación con la Estrategia de la Empresa

- Un DWH está mucho más cerca de la estrategia de una empresa de lo que pueden estarlo generalmente las aplicaciones de carácter transaccional
- El objetivo del DWH se expresa en términos puramente de negocio como ***"mantener la fidelidad de los clientes"***

3-Mejora continua

- Un DWH una vez construido debe evolucionar en función de los usuarios o de los nuevos objetivos de la empresa y se sitúa, pues, en una lógica de mejora visible y frecuente.

4-Nivel de Madurez Diferente

- ❑ El nivel de madurez de cada empresa ante los sistemas de decisión puede diferir considerablemente.
- ❑ Para algunas que han abordado el tema, están en un proceso de mejora continua.
- ❑ Para otras se trata de un ámbito aún desconocido.

FASES DE IMPLANTACIÓN

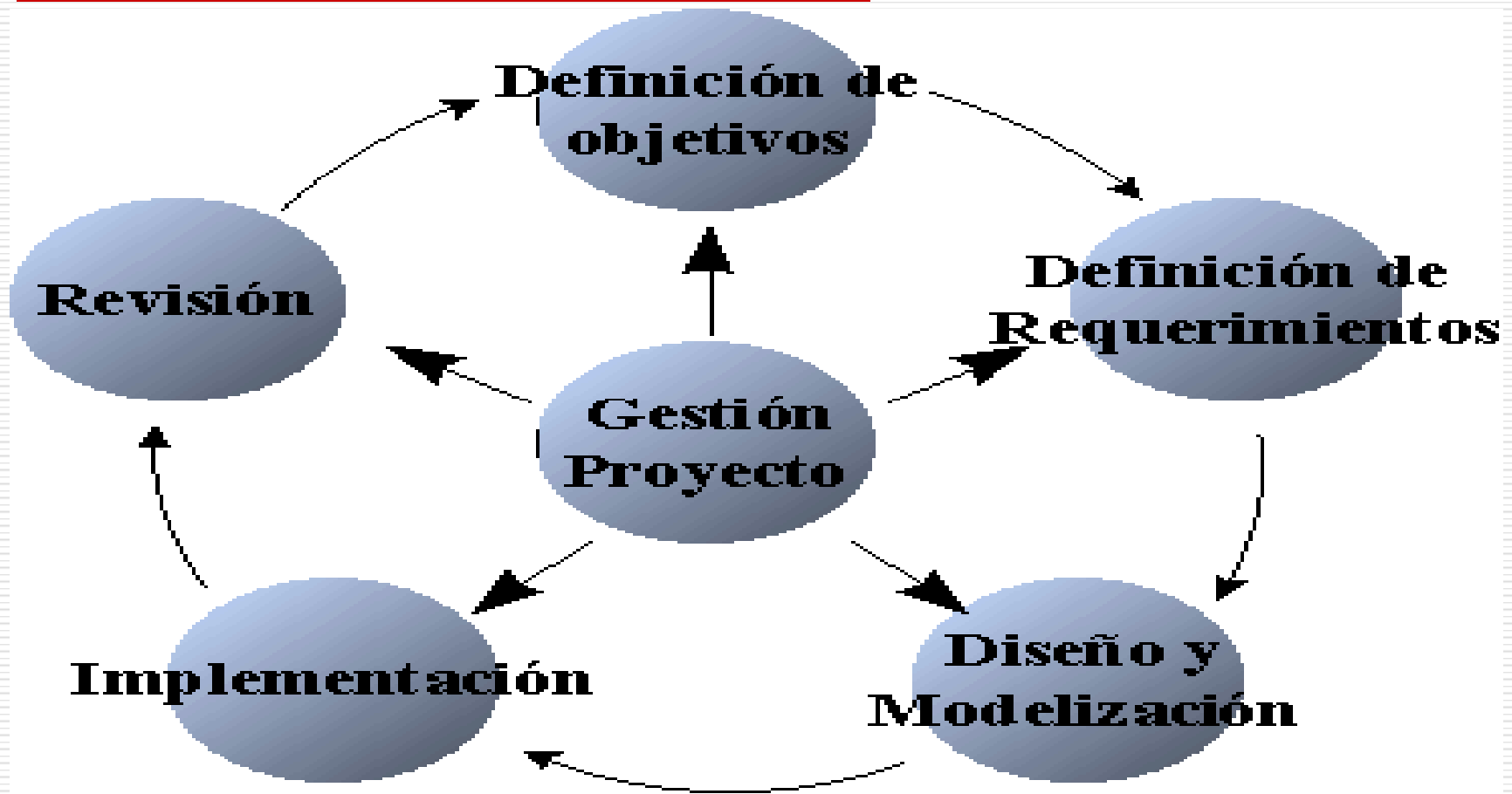
- Planteamos aquí la metodología propuesta por SAS Institute: la **"Rapid Warehousing Methodology"**. Dicha metodología es iterativa, y está basada en el desarrollo incremental del proyecto de **Data Warehouse** dividido en **cinco fases**.

"Rapid Warehousing Methodology"

Se basa en Rapid Application Development

- ❑ Método de implementación iterativa para el desarrollo de aplicaciones.
- ❑ Se basa en una inclusión temprana de los usuarios durante todo el ciclo de desarrollo.
- ❑ Diseñado por James Martin.

Fases propuestas



Definición de los Requerimientos de información

- Luego de definidos los objetivos, se deben evaluar las necesidad de información para lograrlos.

Para esto es preciso visualizar y comprender las ventajas que puede reportar un DWH para la organización.

Diseño y modelización

- ❑ Los requerimientos de información identificados durante la fase anterior proporcionarán las **bases** para realizar el **diseño y la modelización** del **DWH**.
- ❑ En esta fase se identificarán las **fuentes** de los datos (**sistema operacional, fuentes externas,..**) y las transformaciones necesarias para obtener el modelo lógico de datos del DWH. Este modelo estará formado por entidades y relaciones que permitirán resolver las necesidades de información.
- ❑ Se hace un modelo lógico (**DER**), luego con este se diseña el modelo físico de datos del DWH (**DDL**).
- ❑ La mayor parte estas definiciones de los datos del DWH estarán almacenadas en los **metadatos** y formarán parte del mismo.

Implementación

- Extracción de los datos del sistema operacional y transformación de los mismos.
- Carga de los datos validados en el DWH. Esta carga deberá ser planificada con una periodicidad que se adaptará a las necesidades de refresco detectadas durante las fases de diseño del nuevo sistema.
- Explotación del Data Warehouse mediante diversas técnicas dependiendo del tipo de aplicación que se de a los datos:

Implementación y Uso del DWH

- ❑ Consultas y Reportes de gestión.
- ❑ On-line analytical processing (OLAP)
- ❑ Executive Information System (EIS) ó Información de gestión
- ❑ Sistemas Soporte de Decisión (SSD)
- ❑ Visualización de la información
- ❑ Minería de Datos (Data Mining), etc

Revisión

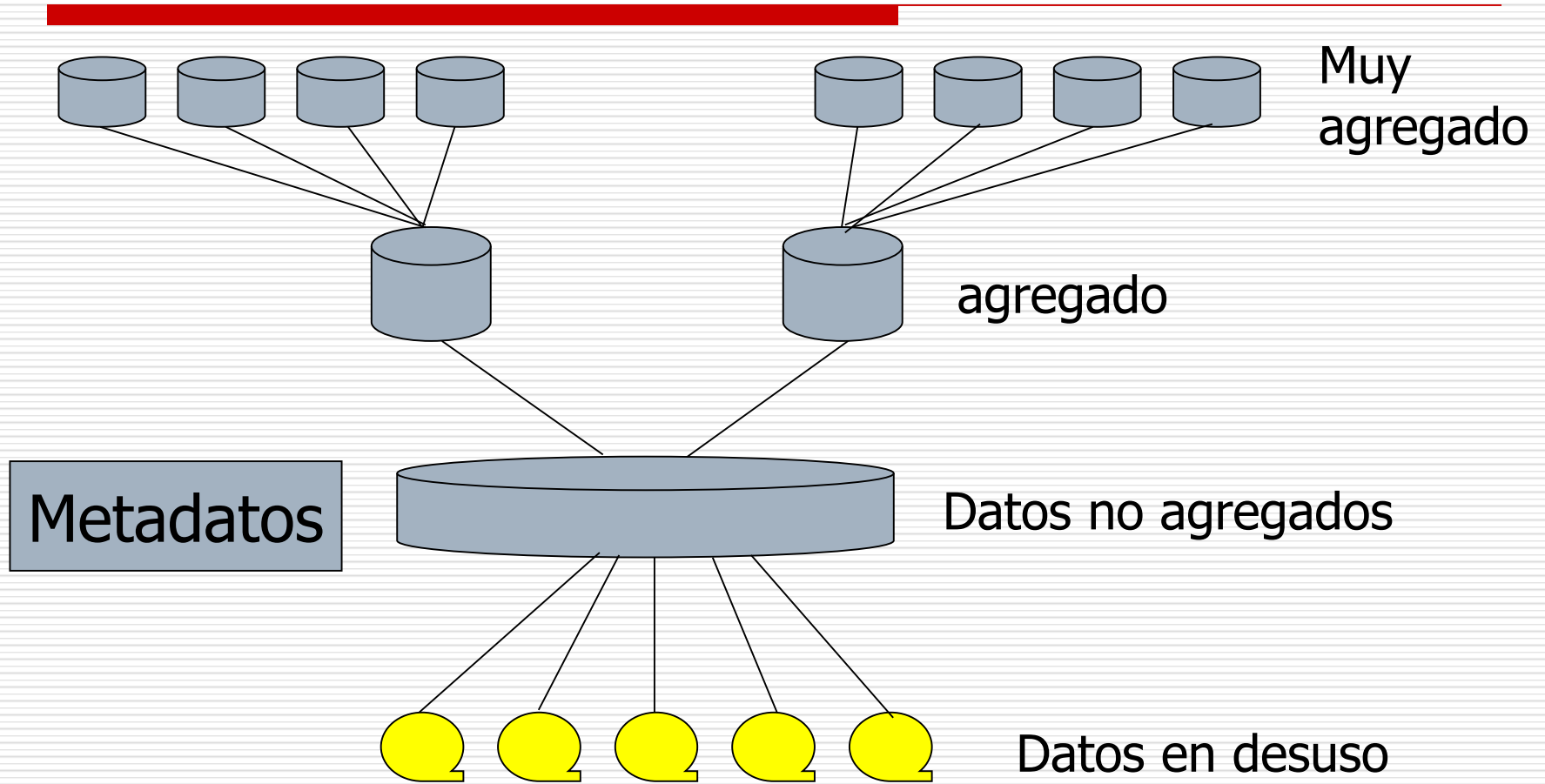
- ❑ La construcción del DWH no finaliza con la implantación del mismo, sino que es una tarea iterativa en la que se trata de incrementar su alcance aprendiendo de las experiencias anteriores.
- ❑ Después de implantarse, debería realizarse una revisión del DWH planteando preguntas que permitan, después un periodo de uso (seis o nueve meses posteriores a su puesta en marcha), definir cuáles serían los aspectos a mejorar o potenciar en función de la utilización que se haga del nuevo sistema

Capacitación: Diseño de la estructura de cursos de formación

- ❑ Con la información obtenida de reuniones con los distintos usuarios se diseñarán una serie de cursos a medida, que tendrán como objetivo el proporcionar la formación estadística necesaria para el mejor aprovechamiento de la funcionalidad incluida en la aplicación.
- ❑ Se realizarán prácticas sobre el desarrollo realizado, las cuales permitirán fijar los conceptos adquiridos y servirán como formación a los usuarios.

ESTRUCTURA DEL DATA WAREHOUSE

Estructura de datos en el DWH



Detalle de datos actuales

- En gran parte, el interés más importante radica en el detalle de los datos actuales, debido a que:
 - Refleja las ocurrencias más recientes, las cuales son de gran interés
 - Es voluminoso, ya que se almacena al más bajo nivel de granularidad.
 - Se almacena en los dispositivos mas veloces, discos de mas rápido acceso.

Detalle de Datos Historicos

Los datos antiguos son aquellos que se almacenan sobre alguna forma de almacenamiento masivo. No es frecuentemente accesada y se almacena a un nivel de detalle, consistente con los datos detallados actuales.

Es poco usual utilizar discos veloces como medio de almacenamiento debido a:

- Acceso no frecuente.
- Gran volumen de datos.

Datos ligeramente resumidos

Los datos ligeramente resumidos son aquellos que provienen de los datos actuales.

Este nivel del DWH casi siempre se almacena en disco veloces.

Los puntos en los que se basa el diseño son:

- Grano de tiempo. (Semana / Mes / Año).
- Distribución Geográfica (Punto / Localid / Región)
- Producto o Servicio de la organización.
- Que Atributos se agregaran (resumirán).

Metadata

- ❑ El componente final del data warehouse es la metadata.
- ❑ La metadata se sitúa en una dimensión diferente al de otros datos del DWH, debido a que su contenido no es tomado directamente desde el ambiente operacional, sino que contiene información descriptiva del proceso completo de construcción del DWH.

Metadata del DWH

Metadata: directorio de cada dato DWH

□ Función:

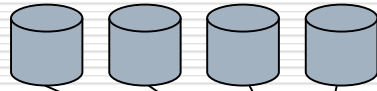
- ayudar al analista a localizar los contenidos del DWH
 - Guiar el mapeo de datos, en la medida que el dato es transformado
 - Guiar los algoritmos usados para agregación/sumarización
-

METADATA: Información contenida

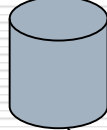
- ❑ identificación de la **fuentes de los datos**
 - ❑ descripción de la **transformación** sufrida al pasar el dato al DWH
 - ❑ información descriptiva del DWH (**tablas, atributos, relaciones, etc.**)
 - ❑ definición de los **términos usados, sinónimos.**
-

Ejemplo de un DWH (Contexto Año 2002)

**Ventas mensuales
por país
1995-2002**



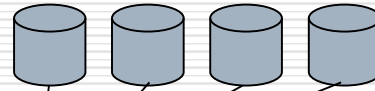
**Ventas semanales
por región
1995-2002**



**Ventas detalladas
2000-2002**



**Ventas mensuales
por línea de producto
1995-2002**



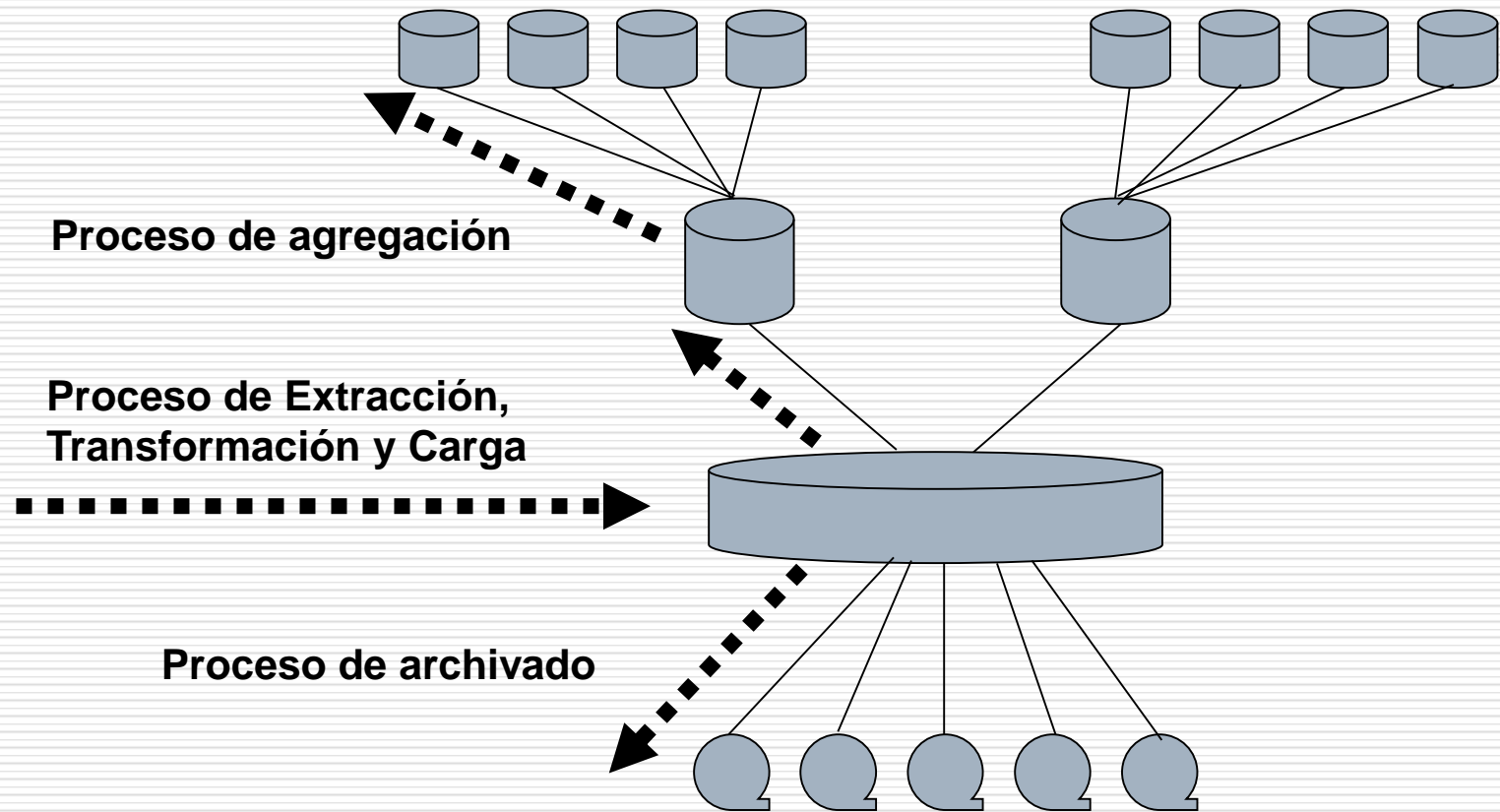
**Ventas semanales
por sub-product
1995-2002**



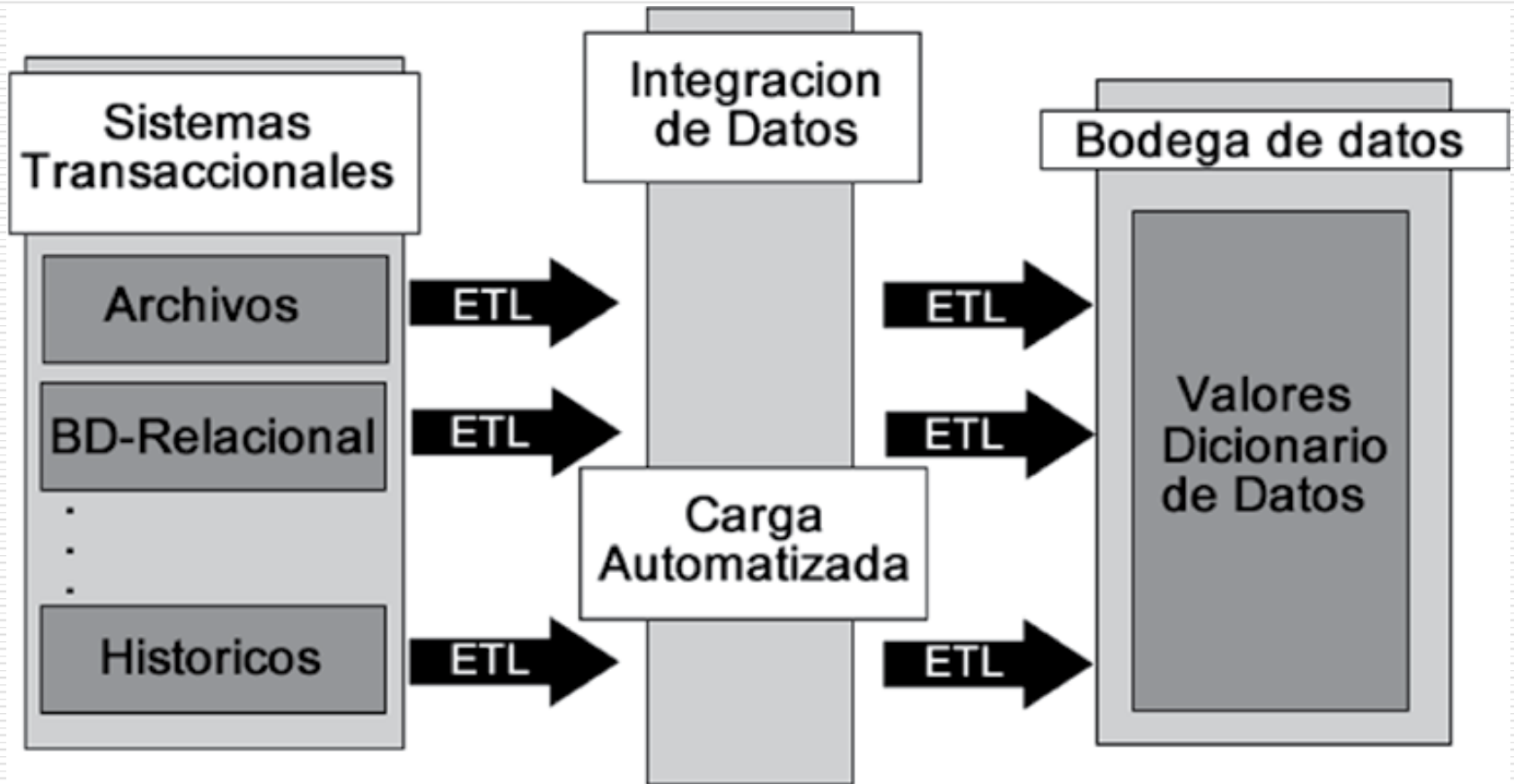
**Ventas detalladas
1995-1999**



Flujo de los Datos

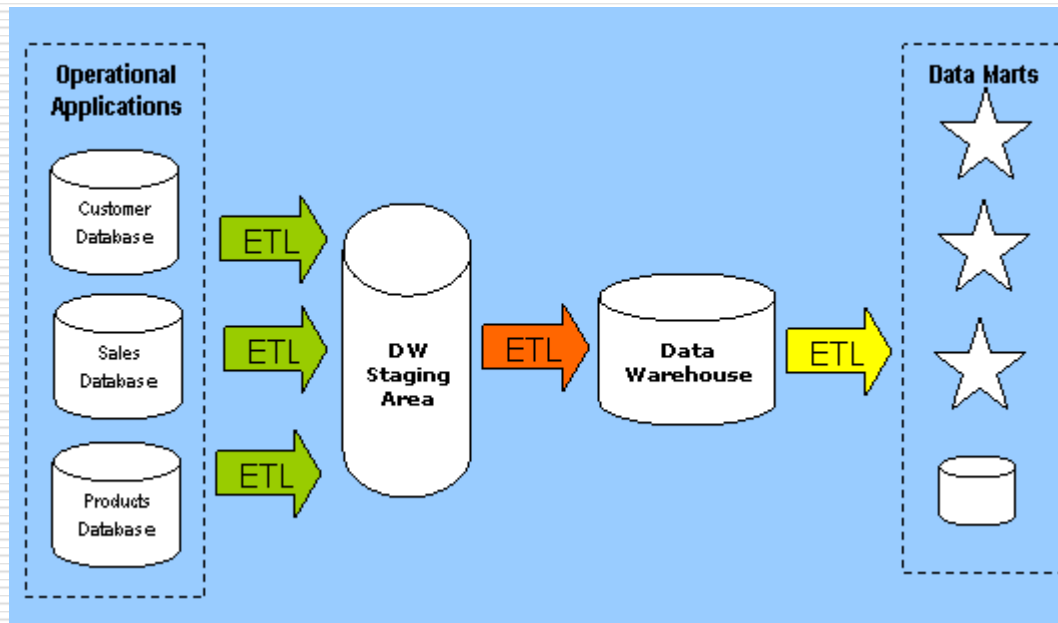


Carga de un DWH



Staging Área

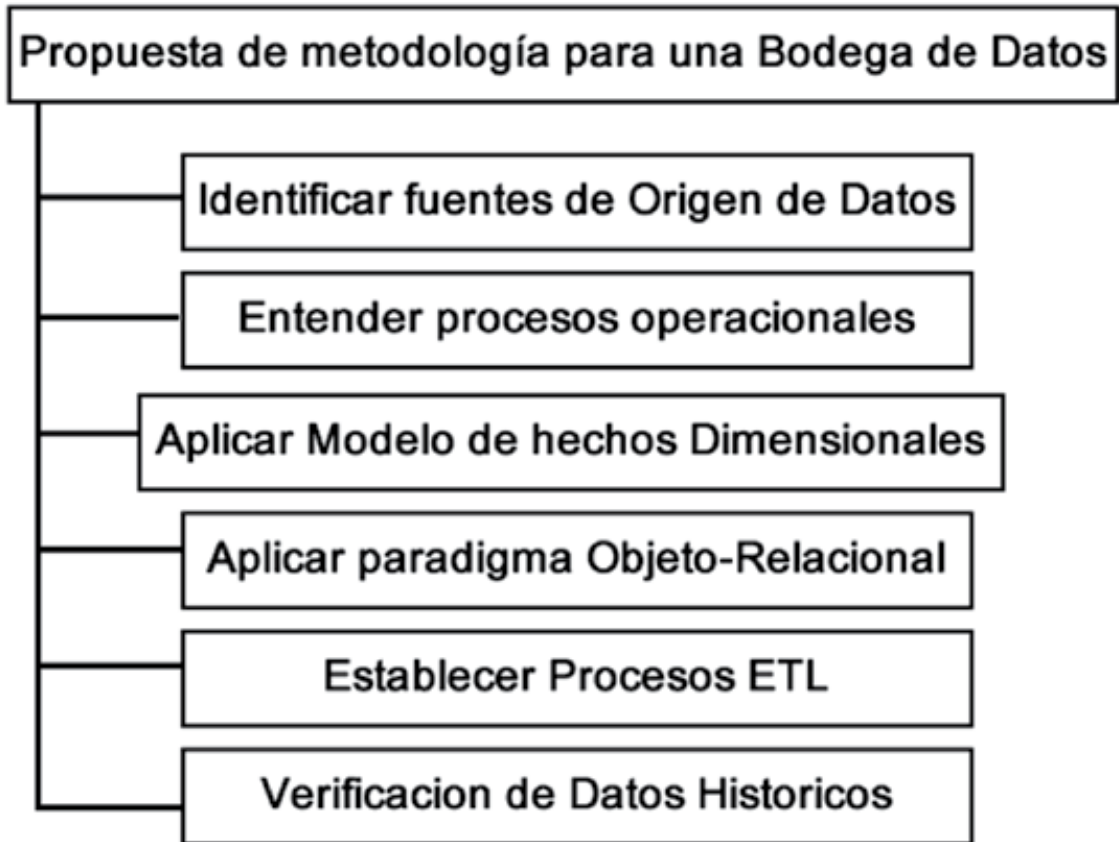
Es el área de trabajo del sistema que permanece entre las fuentes de datos y el data warehouse



Staging Área

- ❑ Facilitar la extracción de datos (los procesos ETL) desde las fuentes de origen de carácter múltiple realizando un pretratado.
- ❑ Realizar lo que se conoce como data cleansing (limpieza de datos).
- ❑ Mejorar la calidad de datos.
- ❑ Ser usado como cache de datos operacionales con el que posteriormente se realiza el proceso de Data Warehousing.
- ❑ Uso de la misma para acceder en detalle a información no contenida en el Data warehouse.

Proceso de transformación



Técnicas de Modelización Estructural

- En esta sección veremos técnicas que afectarán a diversos puntos
 - Consideraciones de Tiempo
 - Técnicas de Optimización

Consideraciones de Tiempo

ESTRUCTURAL	Tiempo		Staging Area	Data Warehouse	Data Marts	
					Relacional	Dimensional
		Actualidad de Datos				
		Agrupaciones basadas en tiempo				
		Retención de Histórico				

¿Cuál es el impacto del Tiempo en cada Almacén de Datos?

Todo el DW se ve afectado por cambios temporales, ya que por definición, es "Tiempo-dependiente"

Preguntas importantes:

¿Cuan actual deben ser los datos para satisfacer las necesidades de negocio?

¿Cuánta historia necesitamos en nuestro negocio?

¿Qué niveles de agregación son necesarios?

¿para qué ciclos de negocio?

Técnicas de Modelización Temporal

- ❑ Los **hechos son los indicadores de negocio** que dan sentido al análisis de las dimensiones. Las **tablas de hechos** incluyen los indicadores asociados a un proceso de negocio en concreto. Ejemplo de Hecho: **Venta**.
- ❑ Están asociados al tiempo

Técnicas de Modelización Temporal

- ❑ Unidades de tiempo
 - Calendarios de negocio
- ❑ Técnicas
 - Foto (Snapshot)
 - Trazado de Auditoría
- ❑ Metadatos temporales
 - Fechas Efectivas de Inicio y Fin
 - Fecha de cambio en Fuentes (evento) en OLTP.
 - Fecha de cambio en Destinos (carga) en OLAP.

Tablas temporales

- ❑ **Transaction Fact Tables:** representan eventos que suceden en un determinado espacio-tiempo. Se caracterizan por permitir analizar los datos con el máximo detalle. Reflejan las transacciones relacionadas con nuestros procesos de negocio (ventas, compras, producción, etc).
- ❑ **Factless Fact Tables:** Son tablas que no tienen medidas y representan la ocurrencia de un evento determinado. Por ejemplo, la asistencia a un curso puede ser una tabla de hechos sin métricas asociadas. Ej: Feriado por elecciones.
- ❑ **Periodic Snapshot Fact Tables:** Son tablas de hecho usadas para recoger información de forma periódica a intervalos de tiempo regulares sobre un hecho. Nos permiten tomar una foto de la situación en un momento determinado (por ejemplo al final del día, de una semana o de un mes). Un ejemplo puede ser la foto del stock de materiales al final de cada día. Arqueo de caja.
- ❑ **Accumulating Snapshot Fact Table:** representan el ciclo de vida completo de una actividad o proceso, que tiene un principio y final. Suelen representar valores acumulados. Flujo de Caja.
- ❑ **Consolidated Fact Tables:** tablas de hechos construidas como la acumulación, en un nivel de granularidad o detalle diferente, de las tablas de hechos de transacciones. Stock Analizado, con ingreso/egreso.

Foto (Snapshot)

- ❑ Dos técnicas diferentes
 - Múltiples Tablas
 - Tabla Única
- ❑ Uso de *Fecha Efectiva (Snapshot)* en un ejemplo. Metadatos a nivel de registro

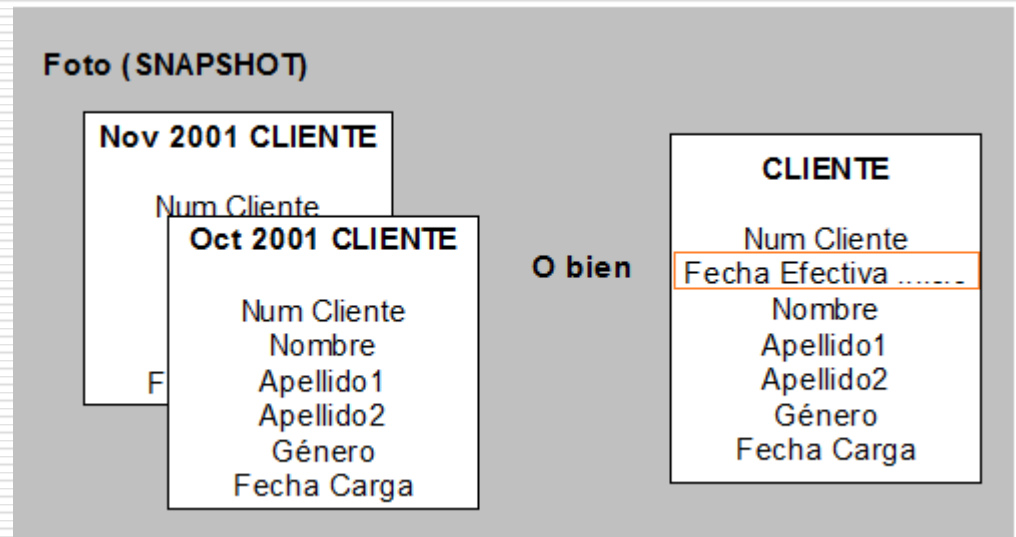


Foto (Snapshot) Múltiple

- ❑ Una tabla para cada período
- ❑ Se guardan TODOS los datos (cambien o no)
- ❑ Nombre de la tabla refleja el período
- ❑ Buen enfoque de (extracción/carga/modelado) para Data Marts. Cada mes, en el ejemplo, representa los datos tal y como estaban
- ❑ Mal enfoque para Staging, ya que hay mucha replicación de datos
- ❑ ¿Cómo obtendría las sig consultas?
 - Clientes nuevos de nov/01.
 - Clientes perdidos en nov/01.
 - Clientes que permanecen a nov/01.
 - Cli.agregados y eliminados nov/01.
 - Cantidades de agregados, eliminados y que permanecen nov/01.
 - Que objeto usaría para obtener estos dinámicamente con esta
 - Estrategia?

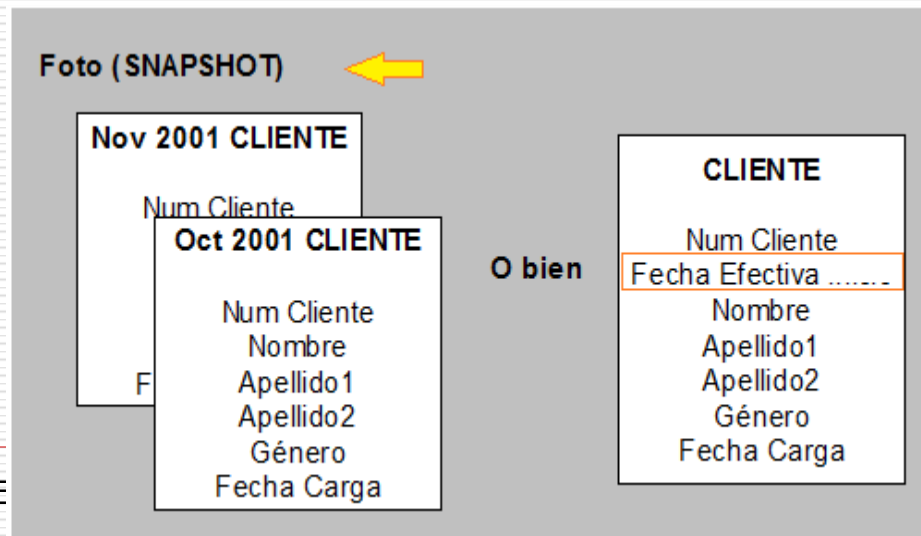


Foto (Snapshot) Única

- ❑ Una sola tabla
- ❑ Se **insertan** solo las filas nuevas (clientes para este caso)
- ❑ Buen enfoque para Data Marts y puede ser útil en el Warehouse.
- ❑ Buen enfoque para Staging, menos replica
- ❑ Time Stamps imprescindibles
- ❑ ¿Qué deberían hacer las ETL con esta estrategia?
- ❑ ¿Cómo se reflejan las bajas? ¿Deberíamos agregar ...?

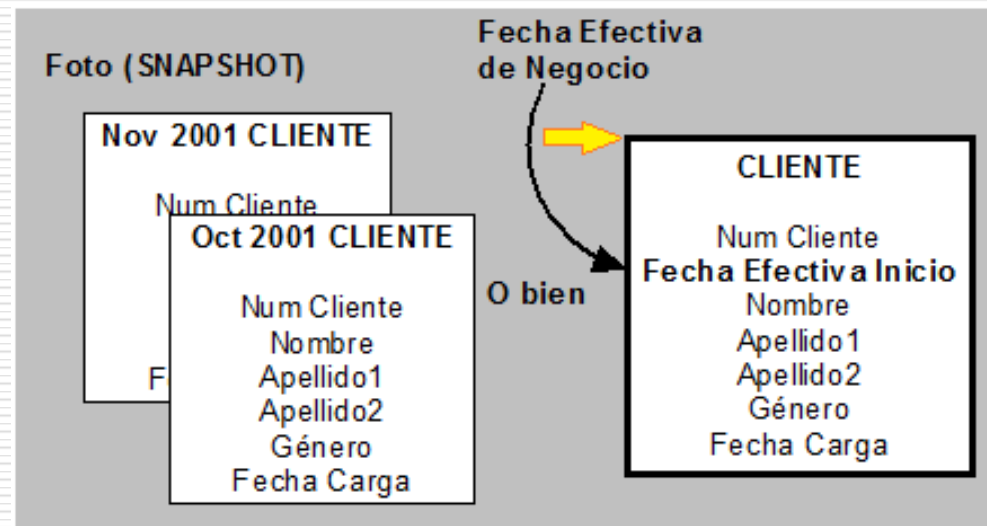


Foto (Snapshot) Única

- ❑ Fechas (Time Stamps) necesarias para identificar la validez de los datos:
 - Fecha efectiva de Inicio
 - Fecha efectiva de Fin (no está en el ejemplo)
 - Fecha de Carga

Num Cliente	Fecha Efectiva Inicio	Nombre	Género	Fecha Carga
2304	31/10/2001	Juan Reyes	Hombre	01/11/2001
5590	31/10/2001	Julia Astur	Mujer	01/11/2001
6720	31/10/2001	Carlos Márquez	Hombre	01/11/2001
7841	31/10/2001	Luis Tesquilo		01/11/2001
2304	30/11/2001	Juan Reyes	Hombre	01/12/2001
5590	30/11/2001	Julia Picado	Mujer	01/12/2001
6720	30/11/2001	Carlos Márquez	Hombre	01/12/2001
7841	30/11/2001	Luis Tesquilo		01/12/2001

Vemos la duplicidad de los datos

Trazado de Auditoría

- ❑ Guarda los cambios de los datos de interés
- ❑ Información:
 - Fecha del cambio
 - Razón del cambio
 - Cómo se ha detectado
 - ...
- ❑ Sólo se extraen/cargan valores modificados

CLIENTE	
<u>ID_cliente</u>	
nombre	
apellido1	
apellido2	
género	
fecha_aniversario	



AUDITORIA CLIENTE	
<u>ID_cliente</u>	
fecha_inicio_efectiva	
nombre	
apellido1	
apellido2	
género	
fecha_aniversario	
fecha_carga	

Metadato a nivel registro

Fecha de Negocio (no Metadato)

Metadato a nivel registro

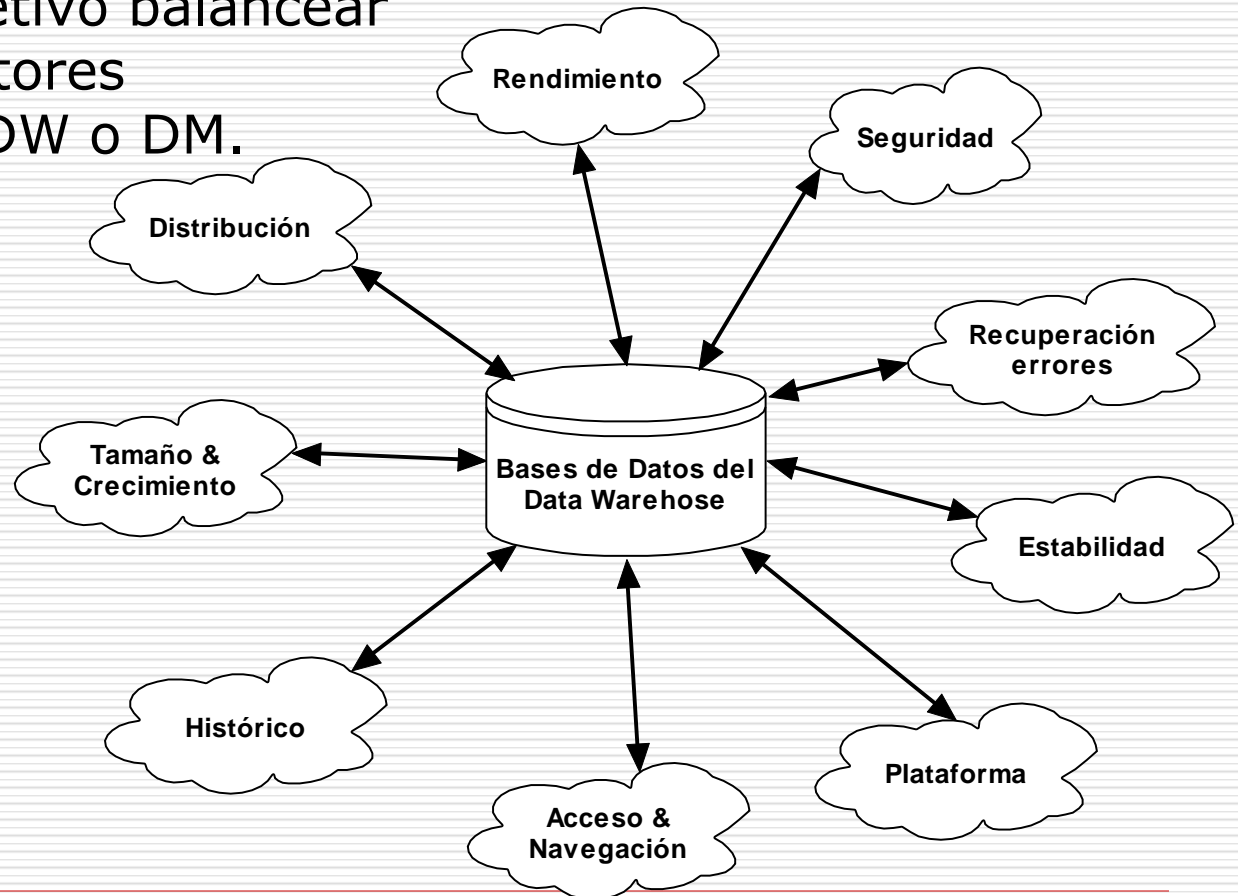
Trazado de Auditoría

Num Cliente	Fecha Efectiva Inicio	Nombre	Género	Fecha aniversario	Fecha Carga
2304	31/10/2001	Juan Reyes	Hombre	01/01/1964	01/11/2001
5590	31/10/2001	Julia Astur	Mujer	06/03/1948	01/11/2001
6720	31/10/2001	Carlos Márquez	Hombre	19/09/1960	01/11/2001
7841	31/10/2001	Luis Tesquilo		25/07/1952	01/11/2001
5590	30/11/2001	Julia Picado	Mujer	06/03/1948	01/12/2001

- Sólo cambios en la tabla
- Usado en Staging Area y Data Warehouse
- Posible en Data Marts, pero no es habitual ya que no es claro para un usuario final

Técnicas de Optimización

- Tiene por Objetivo balancear diferentes Factores dentro de un DW o DM.



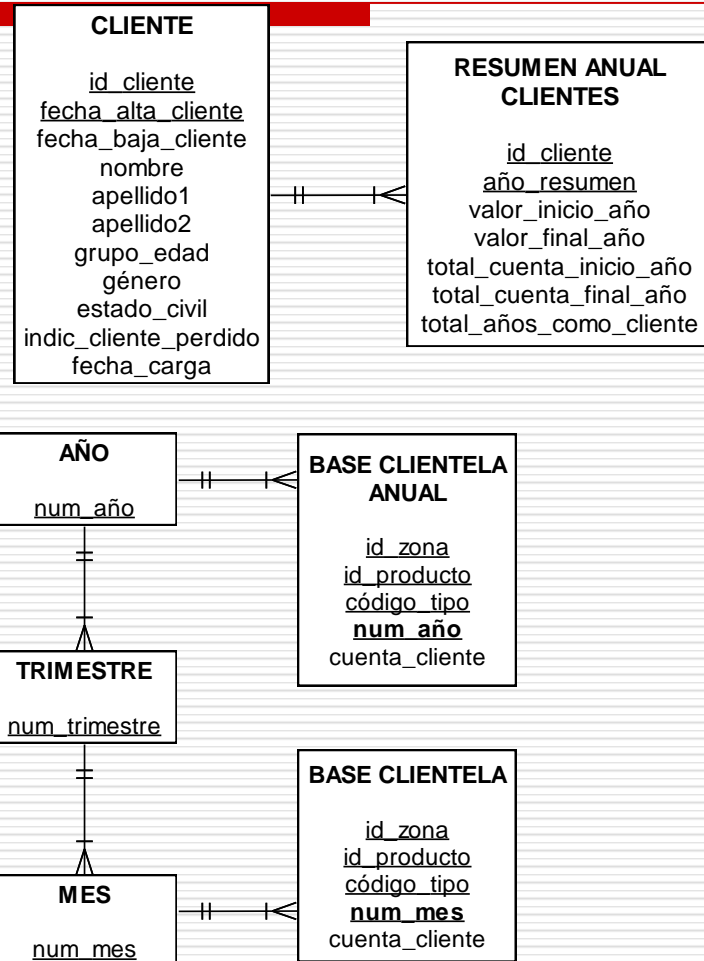
Técnicas de Optimización Estructural y Física

			Staging Area	Data Warehouse	Data Marts	
					Relacional	Dimensional
ESTRUCTURAL	Tiempo	Actualidad de Datos				
		Agrupaciones basadas en tiempo				
		Retención de Histórico				
	Posición	Seguridad				
		Distribución				
	Uso	Acceso				
		Navegación				
		Herramientas				
FÍSICO	Implementación	Rendimiento				
		Tamaño				
		Disponibilidad				
		Recuperación				
		DBMS				

¿Cómo debe optimizarse cada almacén de datos en la Implementación?

Técnicas de Optimización

- Sumarización
- Histórica
- Agrupada



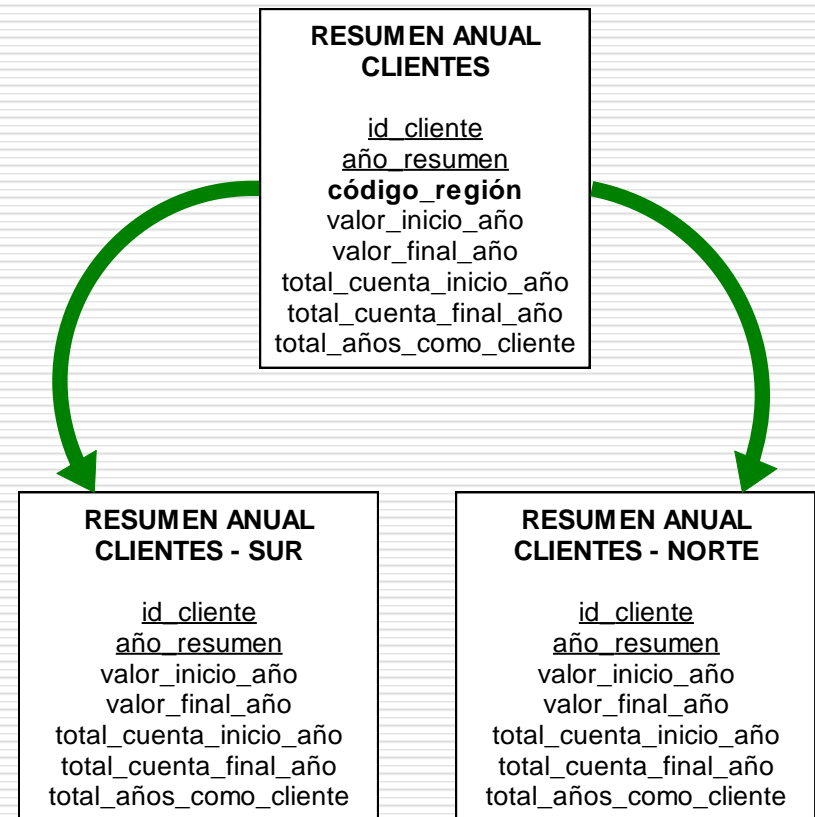
Valor y total
Corresponden a
Hechos sensibles
Para el negocio:
Ej: pos ctacte y/o
Total comprado.

Técnicas de Optimización

- Particionamiento Horizontal
 - Particiones por filas
 - Todos los campos repetidos en las nuevas tablas
 - Uso
 - Aislar datos sensibles
 - Reducción tamaño tablas

Partición Geográfica →

- ¿Para reconstruir la relación completa que operador debemos usar?

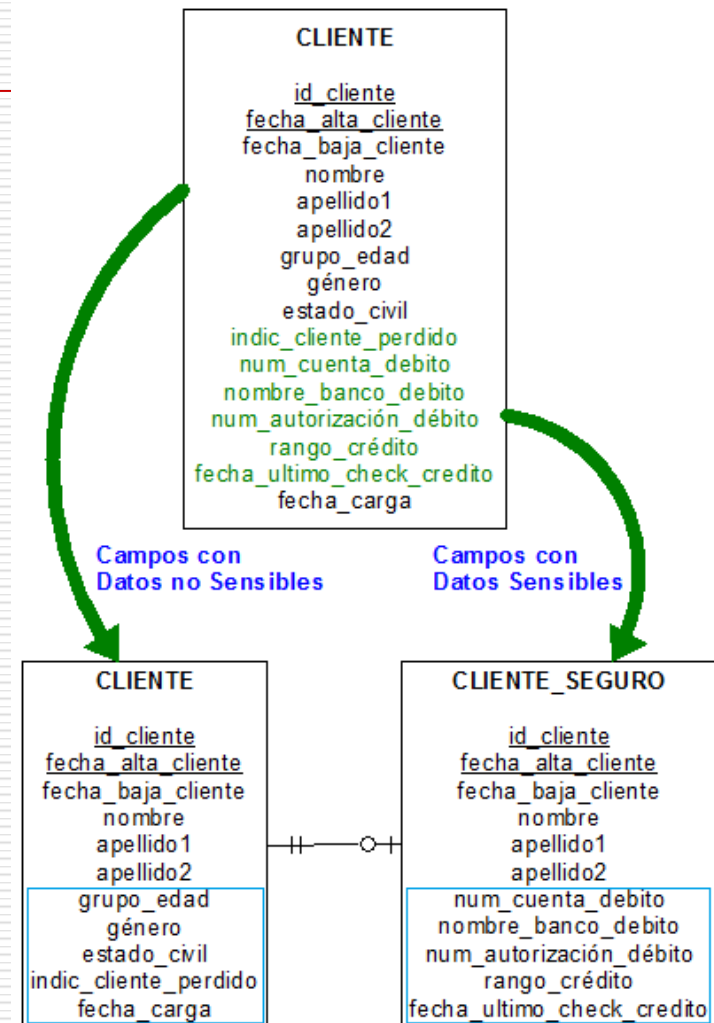


Técnicas de Optimización

- Particionamiento Vertical
 - División por columnas
 - Posibilidad de columnas redundantes
 - Uso
 - Seguridad
 - Distribución

¿Para reconstruir la relación completa que operador debemos usar?

- Puede ser que tengamos Horizontal y Vertical a la vez

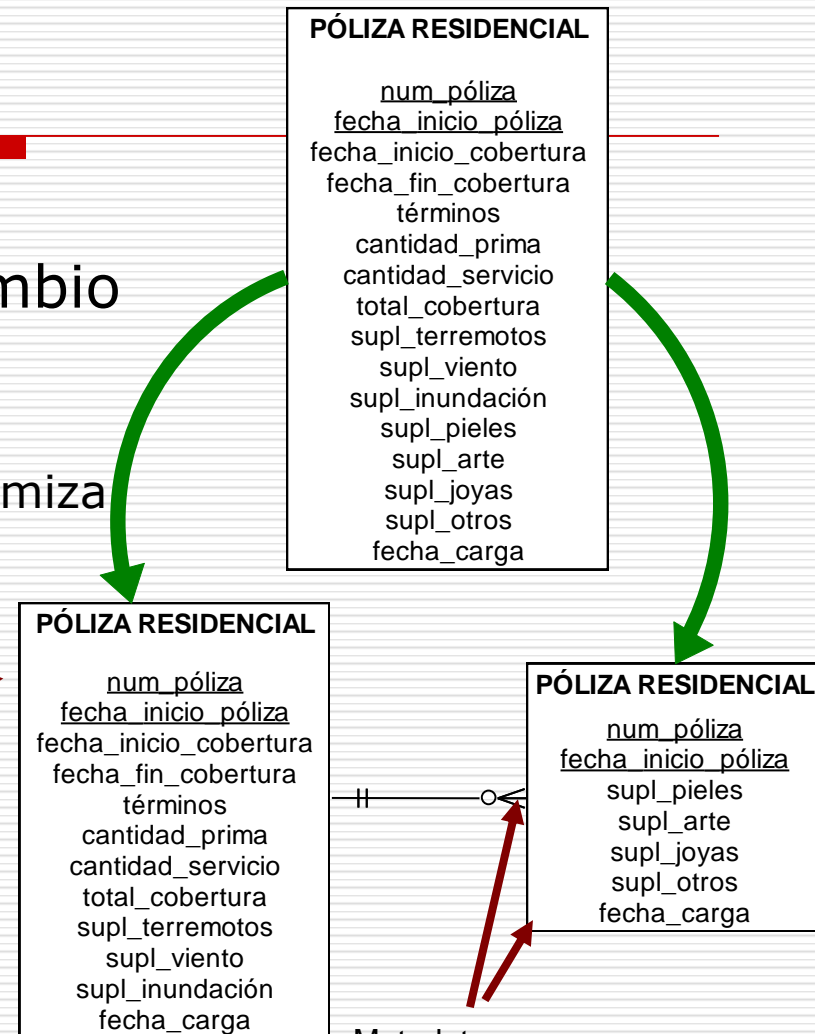


Técnicas de Optimización

- ❑ Particionamiento por Estabilidad
 - Basado en frecuencia de cambio
 - Uso en Staging Area
 - ❑ Velocidad de carga
 - ❑ Separar datos más volátiles minimiza cambios

¿Qué tipo de particionado se realiza con esta estrategia?

Claves Primarias
en ambas tablas

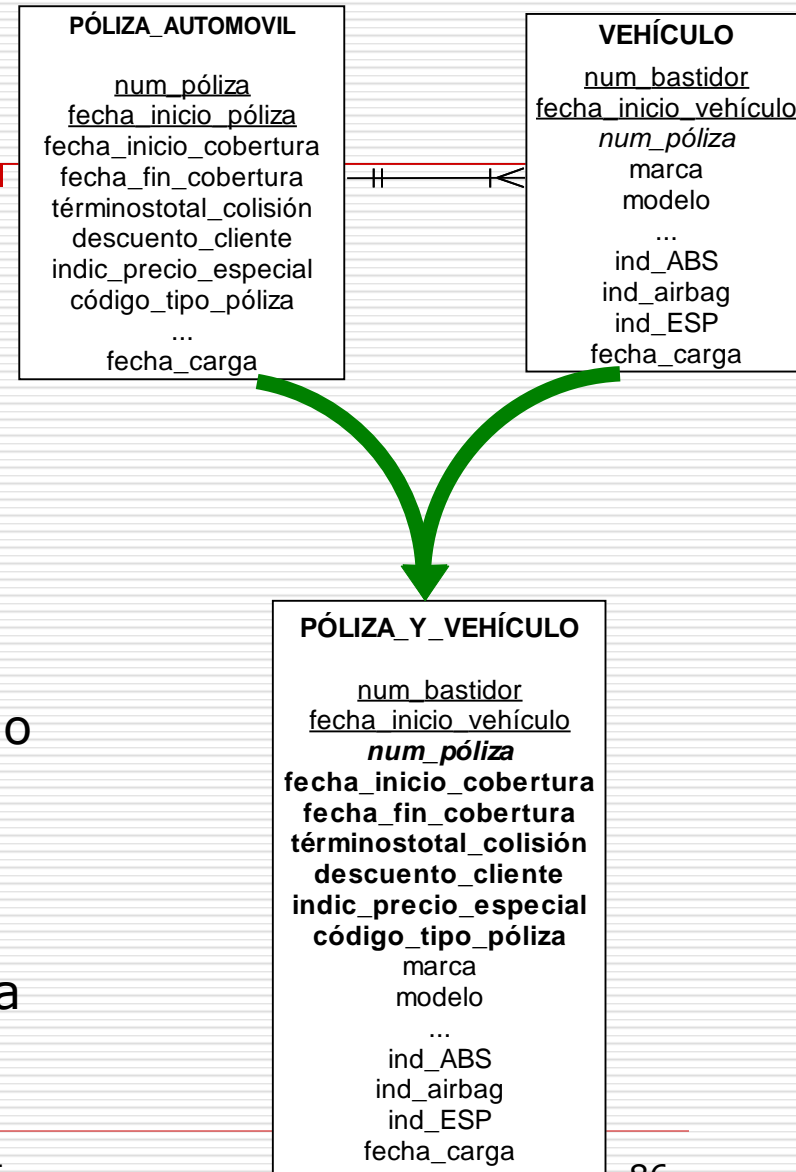


Metadatos a
Nivel Registro en
ambas tablas

Técnicas de Optimización

- Pre-Joins
 - Caso especial de Agregación
 - Data Warehouse y Data Marts
 - Existe redundancia de Información
 - Incrementeo uso espacio
 - Acceso mucho más rápido
 - En el DW
 - Mantendremos también las tablas separadas para cuando no necesitemos el Join

¿Qué objetos de las bases de datos relacionales permiten o facilitan esta técnica?



Técnicas de Optimización

- Cadenas de Datos
 - Caso especial de Agregación
 - Eficiente para Reporting
 - NUNCA en operacionales (OLTP) o Staging, pero muy útil en DW y DM

ARRAY CUENTA CLIENTE
Fecha_alta_Array [línea de negocio x3 [mes x12 [cuenta cliente cuenta tomador]]] fecha_carga



	CUENTAS CLIENTE POR MES Y LÍNEA DE NEGOCIO								
	1/00	2/00	3/00	4/00	...	12/00	1/01	2/01	...
Seguro Hogar									
Seguro Automóvil									
Seguro Vida									

MODELO MULTIDIMENSIONAL

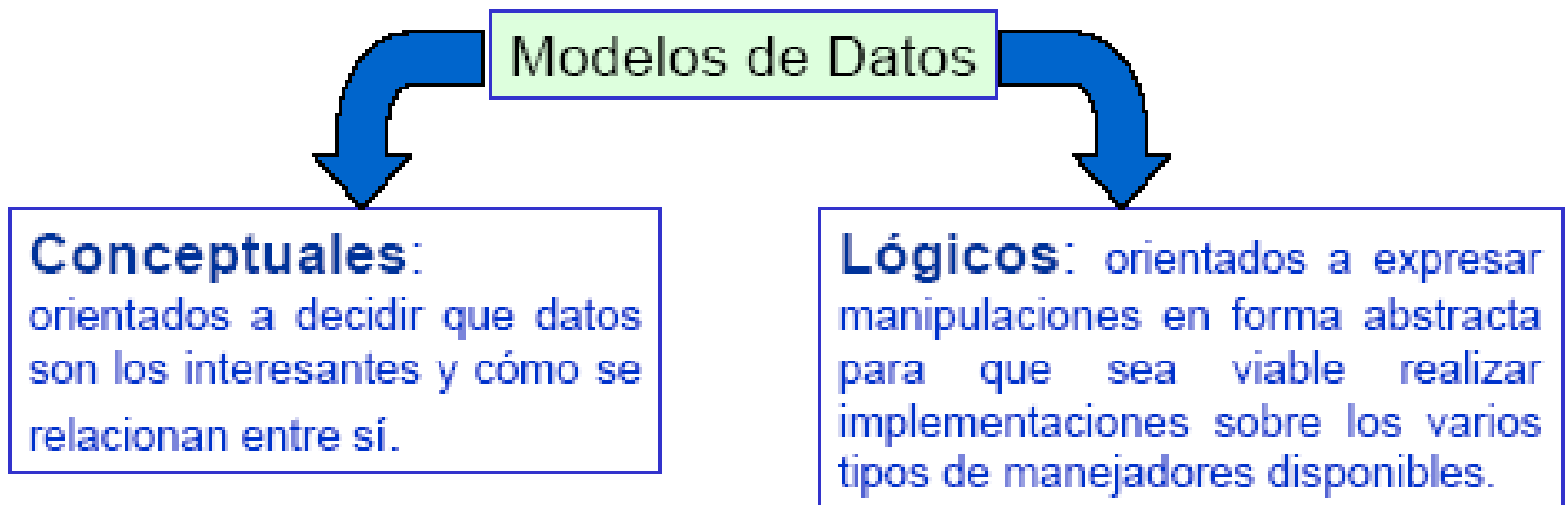
Modelo multidimensional

El modelo multidimensional describe la organización de la información en un DW.

Define los conceptos para agregar **hechos** a lo largo de muchos atributos, llamados **dimensiones**.

Diseño Conceptual MMD

¿ Cuáles son las herramientas que necesita el diseñador para poder razonar sobre los datos y presentárselos al usuario ?



Etapas del diseño conceptual

- ❑ Las principales etapas son:
 - Definir un esqueleto de esquema:
 - ❑ Primer grupo de dimensiones/medidas.
- ❑ Establecer correspondencia entre requerimientos y datos fuentes.
- ❑ Completar jerarquías en las dimensiones.
- ❑ Especificar segundo grupo de medidas (calculadas).

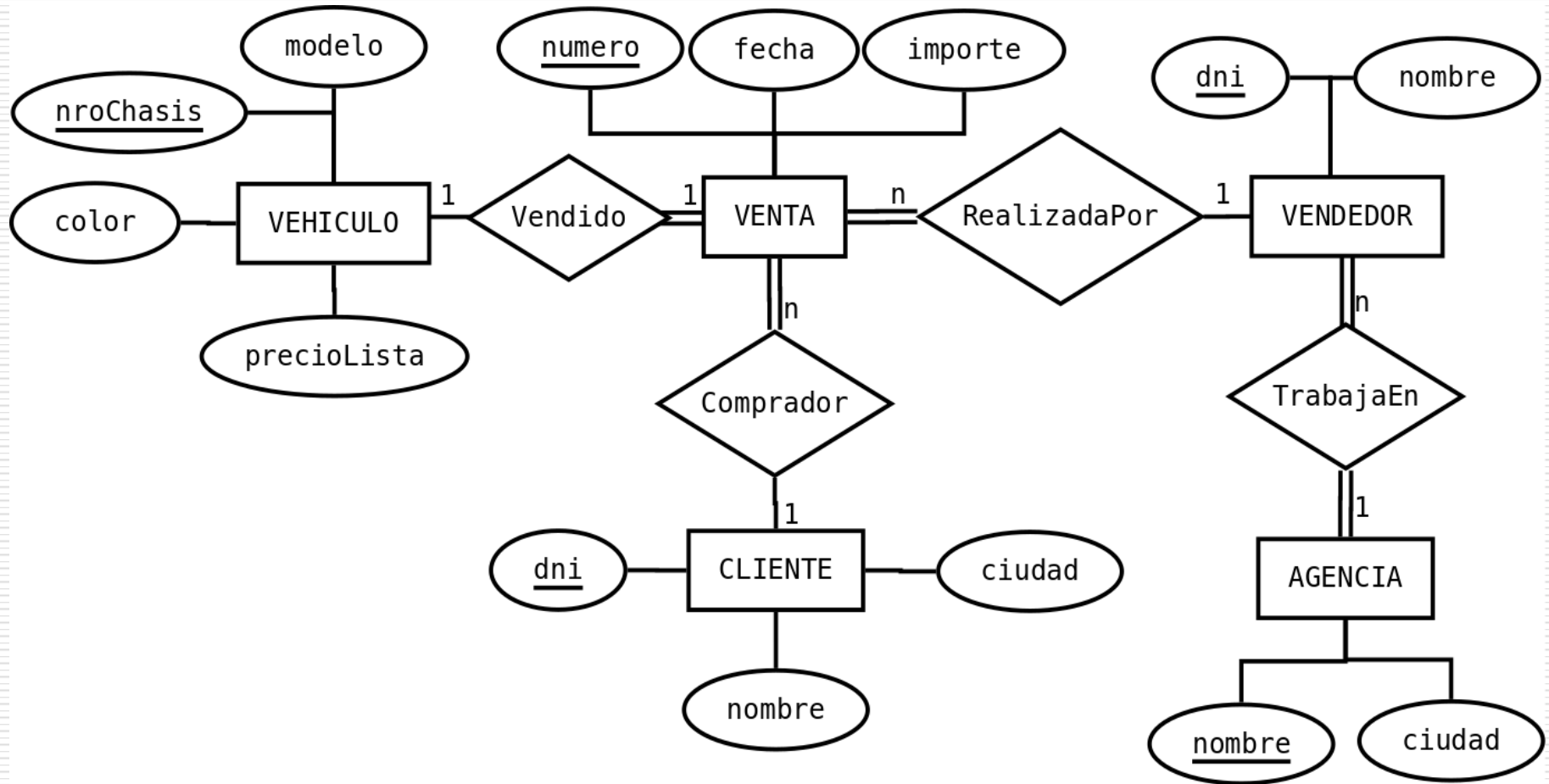
Caso de Estudio de MMD

El modelo multidimensional propuesto por el ***analista de negocios*** incluye:

- Modelo
- Color
- Vendedor
- Además de la dimensión temporal y geográfica (ciudad del cliente y de la agencia).

Modelo multidimensional

El DER del Minimundo del caso de estudio es:



Resultado Buscado: Análisis de Ventas por atributos significativos.

Tabla:

MODELO	COLOR	VOLUMEN-Ventas
MINI VAN	BLUE	6
MINI VAN	RED	5
MINI VAN	WHITE	4
SPORTS COUPE	BLUE	3
SPORTS COUPE	RED	5
SPORTS COUPE	WHITE	5
SEDAN	BLUE	4
SEDAN	RED	3
SEDAN	WHITE	2

Cuadro:

M O D E L O					COLOR
	Mini Van	6	5	4	
	Coupe	3	5	5	
	Sedan	4	3	2	
		Blue	Red	White	

Ejemplo: Ventas por modelo y color

Cuenta de Importe		Etiquetas de columna ▼			
Etiquetas de fila ▼		Blue	Red	White	Total general
Mnivan		6	5	4	15
Sedan		4	3	2	9
Sports Coupe		3	5	5	13
Total general		13	13	11	37

Siguiendo el ejemplo

Representación Tabular



A screenshot of a Microsoft Excel spreadsheet. The title bar reads 'Microsoft Excel - ejemplo.xls'. The menu bar includes 'Archivo', 'Edición', 'Vista', 'Formato', 'Herramientas', 'Datos', 'Ventana', and 'Ayuda'. The toolbar contains various icons for file operations and formatting. The spreadsheet has three columns labeled 'MODELO', 'COLOR', and 'VOLUMEN-Ventas'. The data is as follows:

	A	B	C
1	MODELO	COLOR	VOLUMEN-Ventas
2	MINI VAN	BLUE	6
3	MINI VAN	RED	5
4	MINI VAN	WHITE	4
5	SPORTS COUPE	BLUE	3
6	SPORTS COUPE	RED	5
7	SPORTS COUPE	WHITE	5
8	SEDAN	BLUE	4
9	SEDAN	RED	3
10	SEDAN	WHITE	2

MODELO	COLOR	VOLUMEN-Ventas
MINI VAN	BLUE	6
MINI VAN	RED	5
MINI VAN	WHITE	4
SPORTS COUPE	BLUE	3
SPORTS COUPE	RED	5
SPORTS COUPE	WHITE	5
SEDAN	BLUE	4
SEDAN	RED	3
SEDAN	WHITE	2

Ventas de autos en función de
Modelo y Color.

Siguiendo el ejemplo (2)

Representación Matricial



The screenshot shows a PowerPlay pivot table window titled 'PowerPlay: Sales, a slice of AUTOSCH (Excel97)'. The pivot table is set to 'MODEL' and 'COLOR'. The data is as follows:

	BLUE	RED	WHITE	
MINI VAN	6	5	4	15
SPORTS COUPE	3	5	5	13
SEDAN	4	3	2	9
MODELO	13	13	11	37

M
O
D
E
L
O

Mini Van

Coupe

Sedan

6	5	4
3	5	5
4	3	2

Blue

Red

White

COLOR

Ventas de autos en función de Modelo y Color.

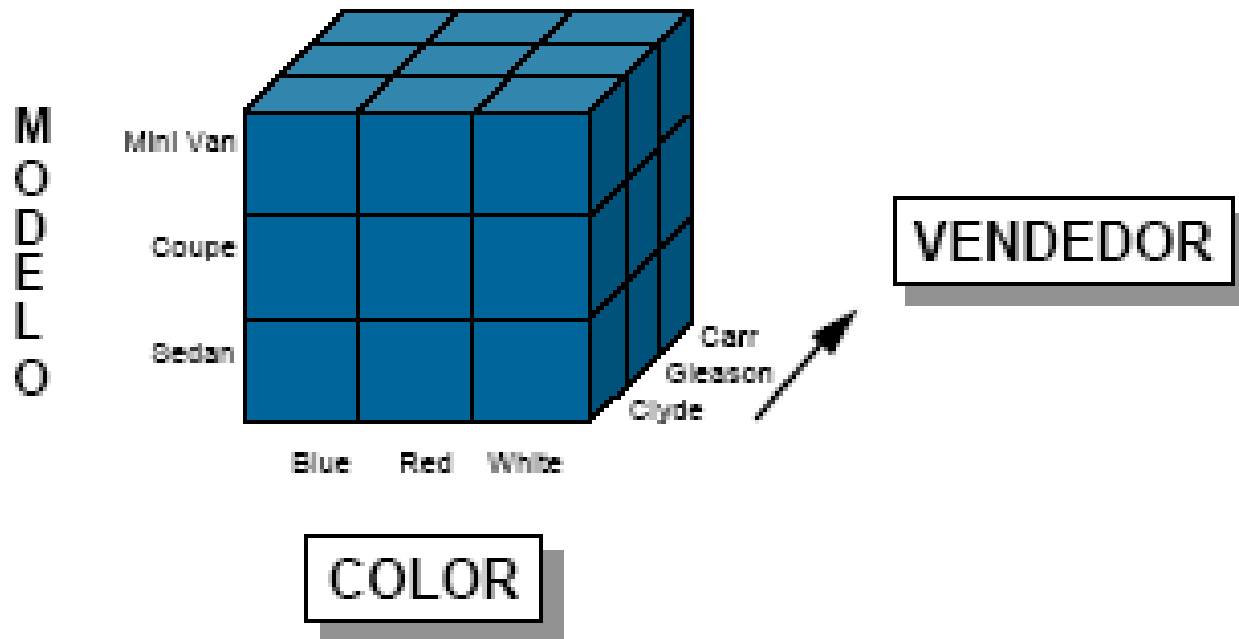
Siguiendo el ejemplo (2)

- ❑ **Se representan los datos como una matriz.**
 - En los ejes están los criterios de análisis.
 - En las intersecciones (celdas) están los valores a analizar.
 - A esta estructura se le llama **Cubo** o **Hipercubo**.

M O D E L O	Mini Van	6	5	4
	Coupe	3	5	5
	Sedan	4	3	2
		Blue	Red	White
		COLOR		

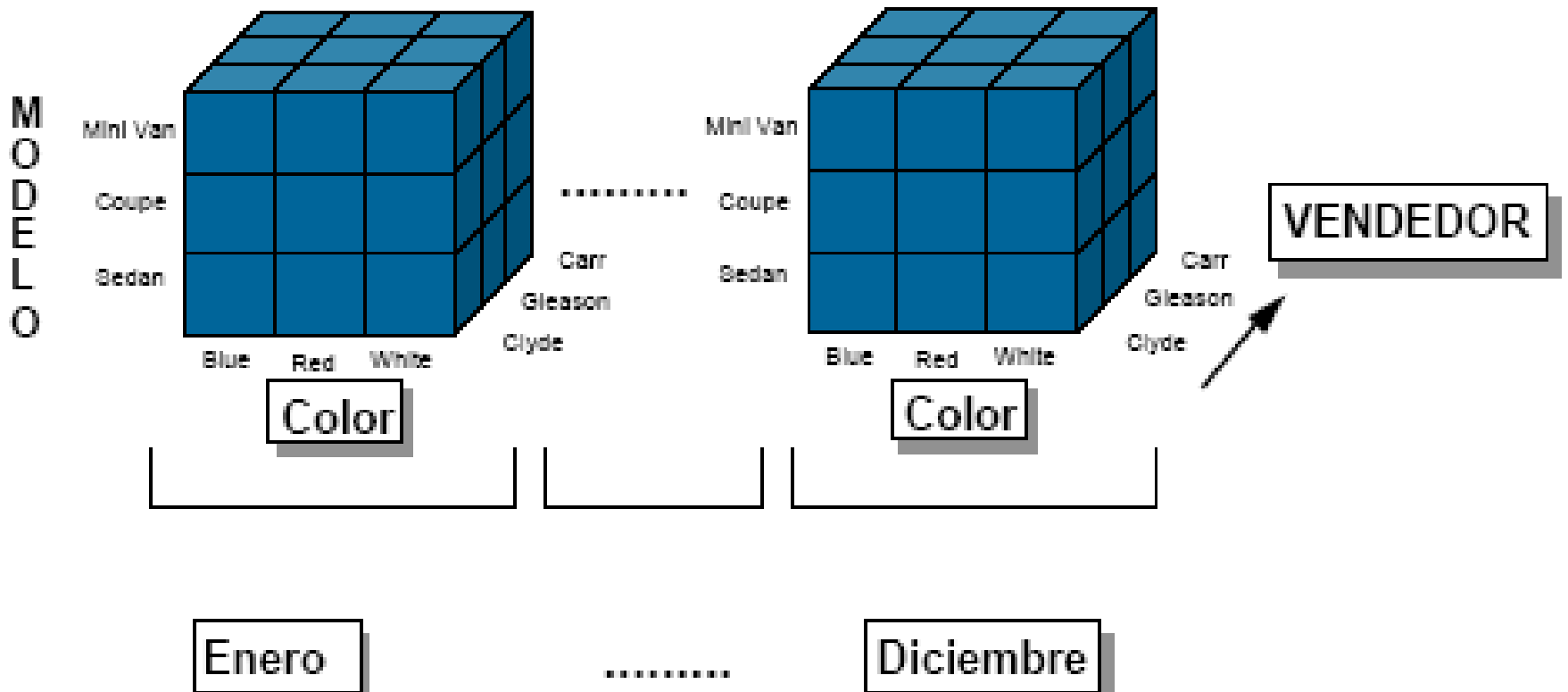
Siguiendo el ejemplo (3)

- Agregando una 3a. dimensión:



Siguiendo el ejemplo (4)

□ Agregando una 4a. dimensión:



Estructuras básicas

- **Los** Cubos o Hipercubos **constan de:**
 - Dimensiones:
 - Criterios de análisis de los datos.
 - Macro-objetos del problema.
 - ***Variables independientes.***
 - Ejes en el hipercubo.
 - Medidas:
 - Valores o indicadores a analizar.
 - Datos asociados a relaciones entre los objetos del problema.
 - ***Variables dependientes.***
 - Variables en la intersección de las dimensiones.

Siguiendo el ejemplo (5)

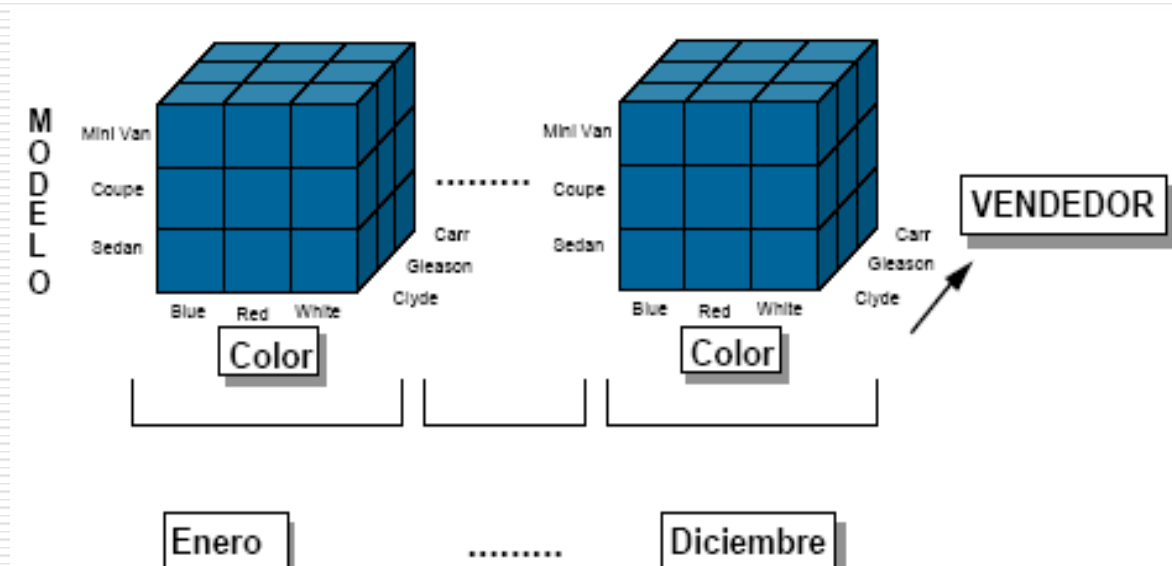
□ En el ejemplo anterior:

■ Dimensiones:

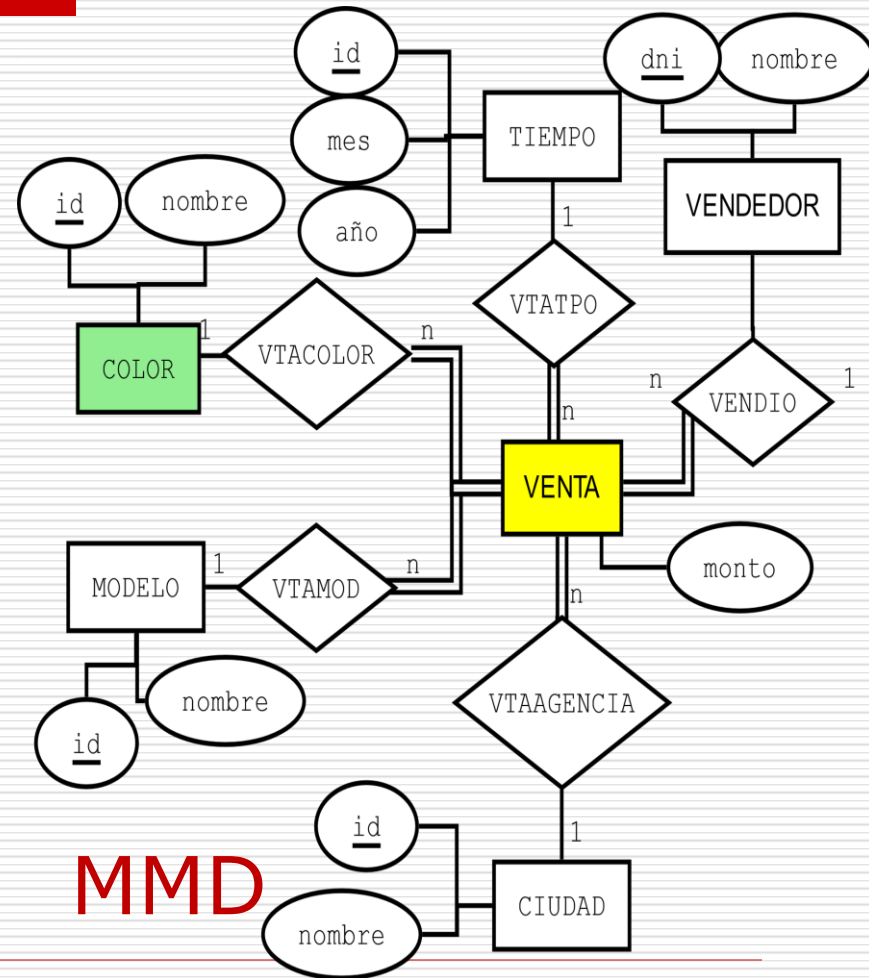
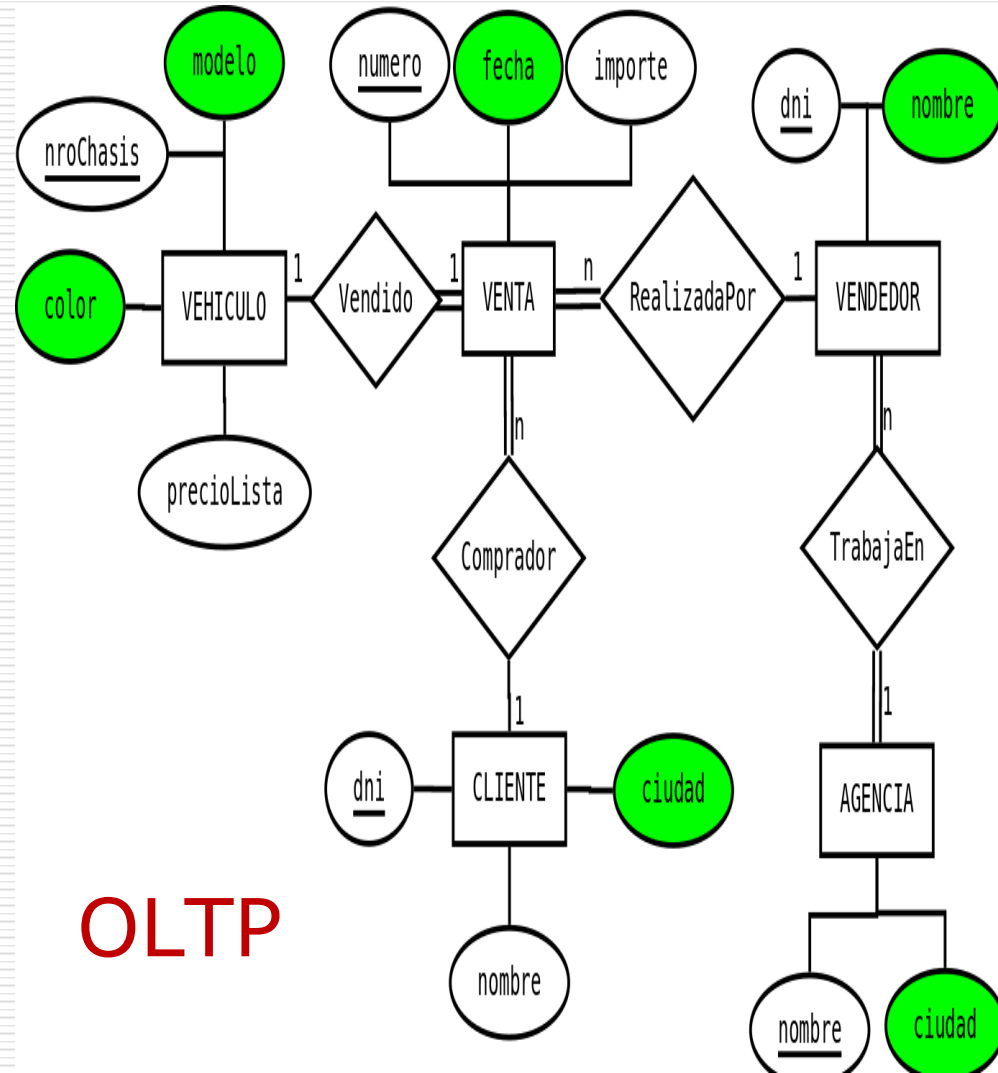
- Modelo
- Color
- Vendedor
- Fecha

■ Medida:

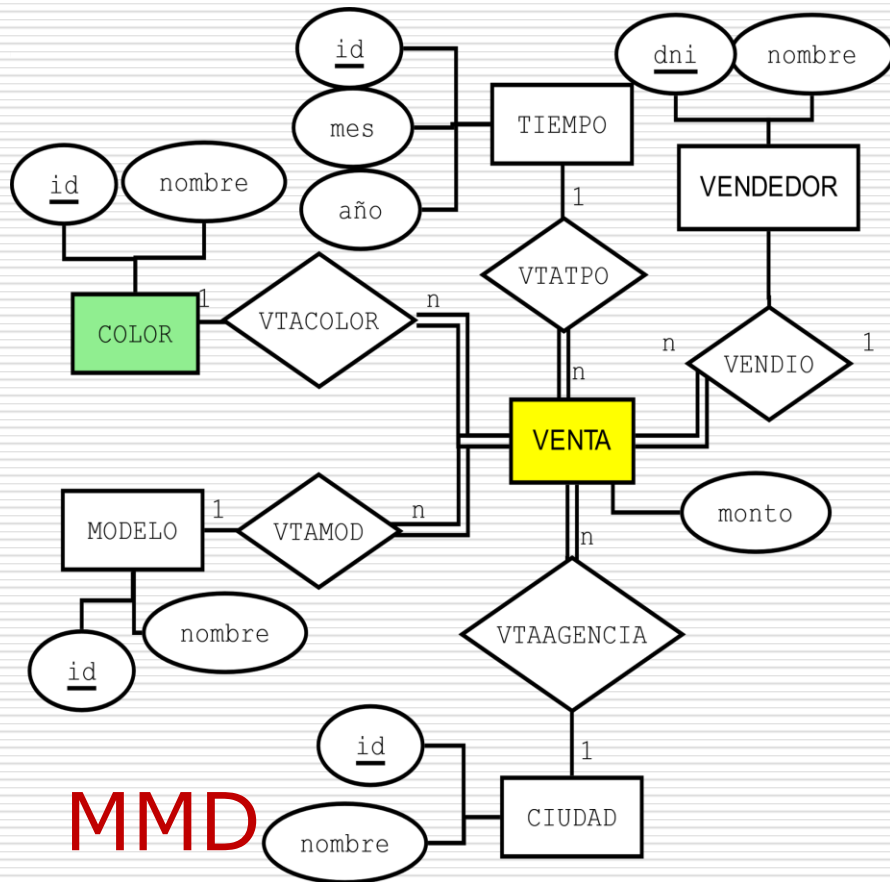
- Cantidad Vendida



Modelo multidimensional



Modelo multidimensional

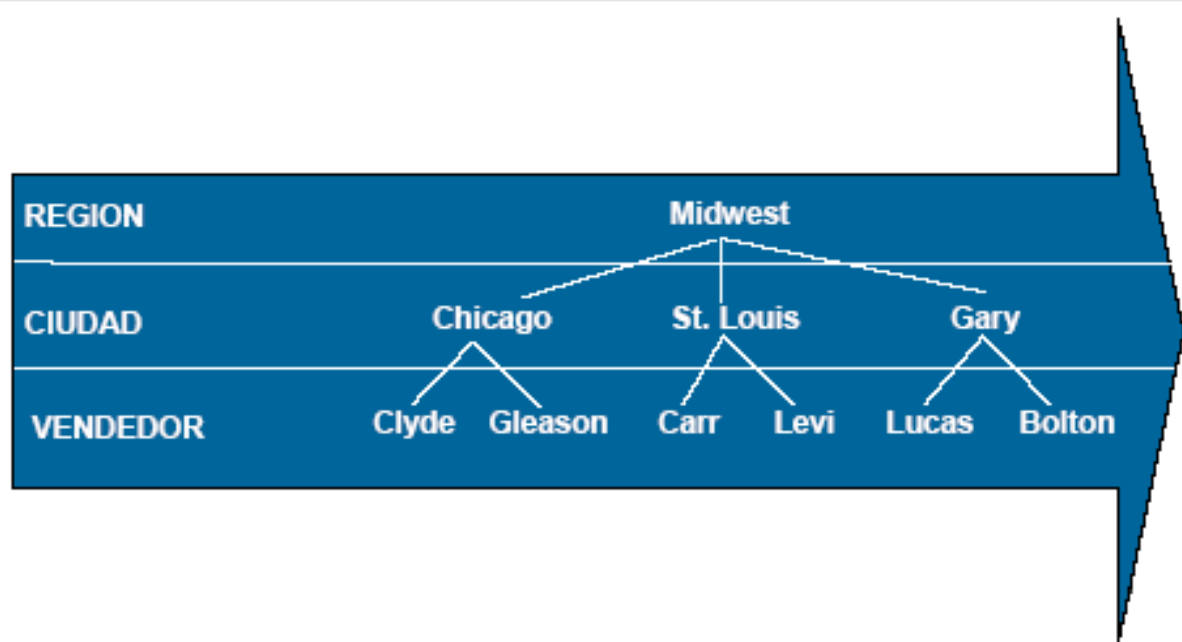


```
CREATE TABLE color (  
  id int PRIMARY KEY,  
  nombre varchar(25)  
  UNIQUE);
```

```
CREATE TABLE venta (  
  idColor int,  
  idModelo int,  
  idCiudadAge int,  
  idTiempo int,  
  dniVendedor int,  
  monto numeric(16,2));
```

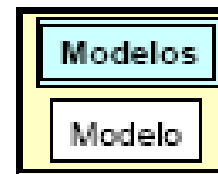
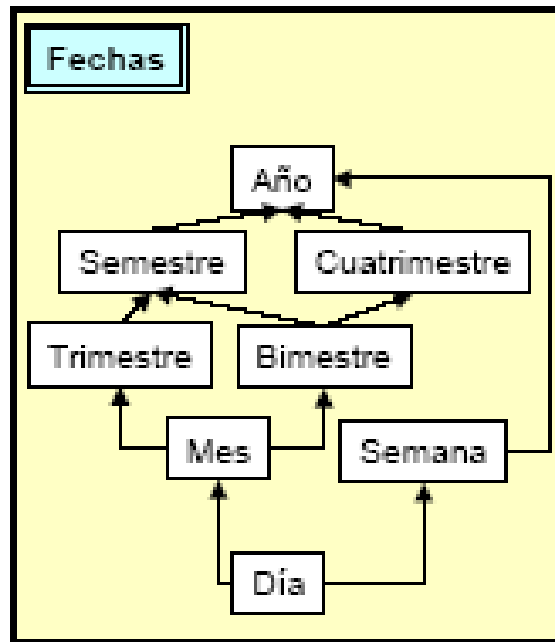
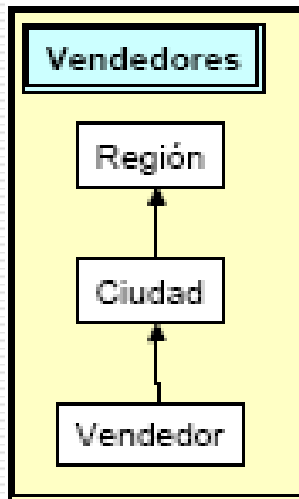

Jerarquías - Siguiendo el ejemplo (6)

- ❑ Jerarquías:
 - Los valores se organizan en jerarquías (categorías).
 - Por ejemplo: Dimensión: Vendedores

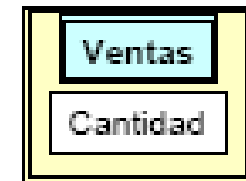


Siguiendo el ejemplo (6)

■ Dimensiones



■ Medidas




Medidas

□ Propiedades:

- Se ubican en la intersección de algunos valores de las dimensiones. Dado un valor para cada dimensión se puede determinar un valor para la medida.


Definición: Se llama **coordenada** a una tupla formada por un valor de cada dimensión.

Medidas - Siguiendo el ejemplo (6)



M	Mini Van	6	5	4
O				
D				
E	Coupe	3		5
L				
O	Sedan	4	3	2
		Blue	Red	White

COLOR



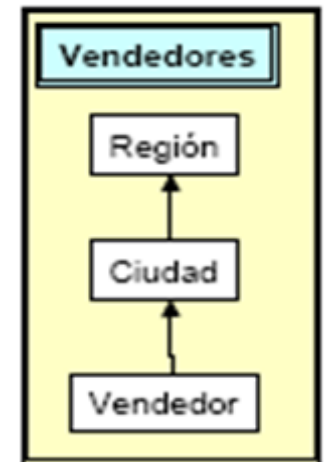
VENTAS("Mini Van", "Blue")=6

¿Cuales serían las coordenadas de este hipercubo?



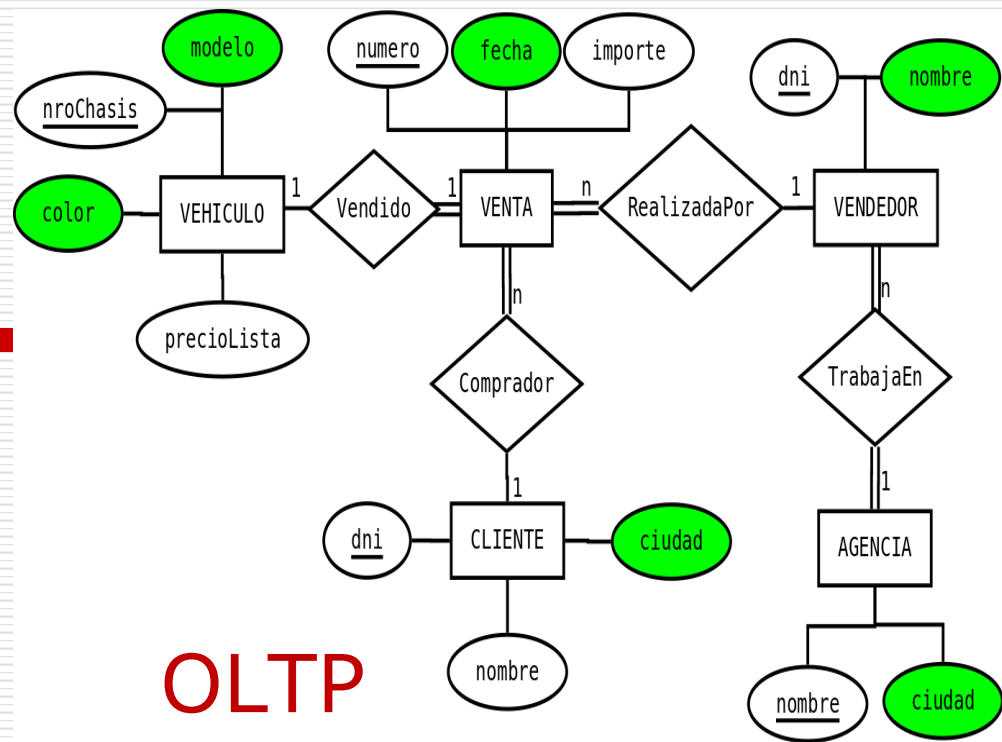
CUBOS

- La realidad se modela como un conjunto de cubos.
 - Cada cubo, esta formado por:
 - Un conjunto de *Dimensiones* organizadas en jerarquías.
 - Un conjunto de *Medidas* asociadas a cada Coordenada.
 - Es posible moverse en las jerarquías de las dimensiones y observar de esa forma, diferentes visiones de las medidas.



Armado CUBO

De la información del
Modelo Operativo
Saldría por agregación:



SELECT c.nombre color, m.nombre modelo,
SUM(cantidad) can, SUM(monto) monto

FROM venta v

JOIN color c **ON** c.id = idColor

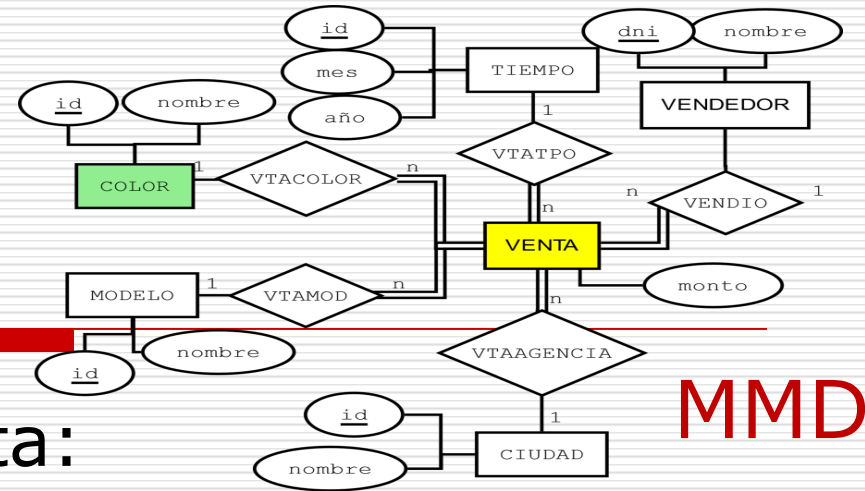
JOIN modelo m **ON** m.id = idModelo

GROUP BY c.nombre, m.nombre;

// Suma todas las ventas por color y modelo

Armado CUBO

En el modelo MMD es directa:



```
SELECT c.nombre color,          m.nombre modelo,  
          SUM(cantidad) can,    SUM(monto) monto  
FROM venta v  
JOIN color c      ON c.id = idColor  
JOIN modelo m    ON m.id = idModelo  
GROUP BY c.nombre, m.nombre
```

// Suma los distintos idTiempo e idCiudadAge
// las distintas agregaciones de MMD.

Modelo Multidimensional Conceptual

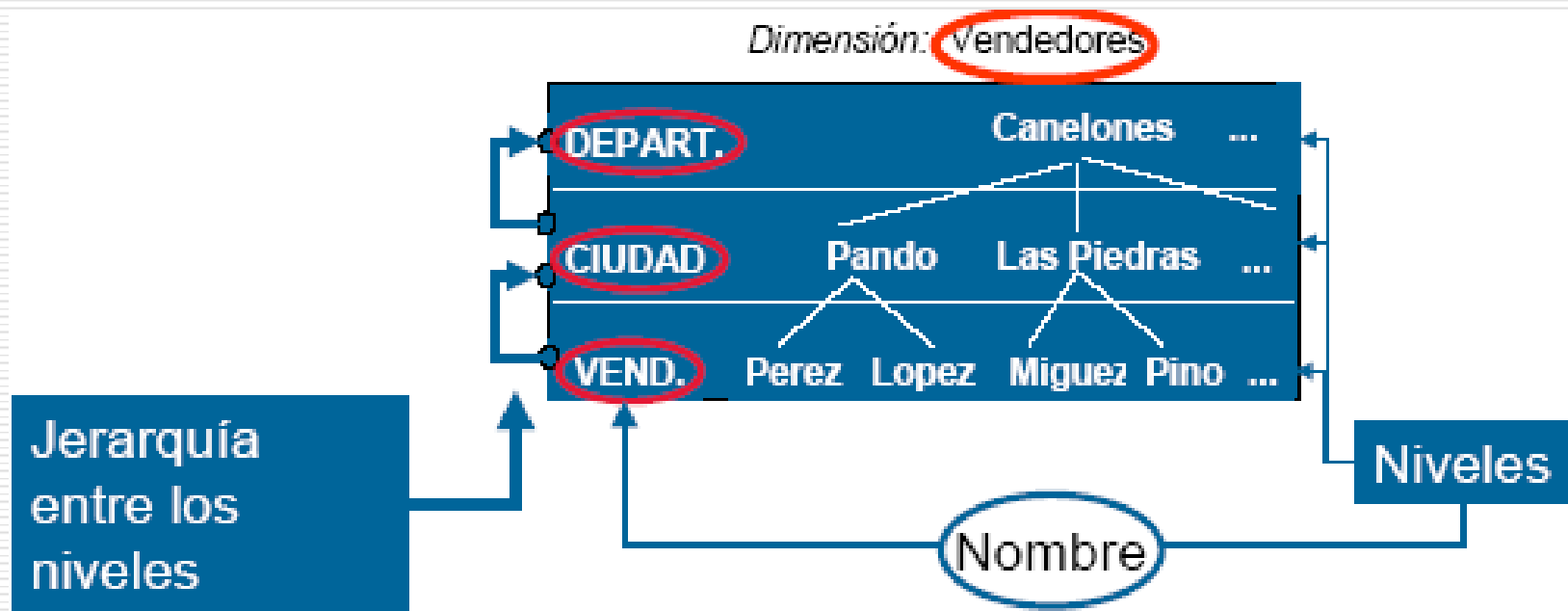
Estrategia basada en Medidas y Dimensiones

Estructuras básicas.

- Niveles.
 - Dimensiones.
 - Con Jerarquías, formadas por Niveles.
 - Incluye Medidas
- (Dimensionalidad Genérica).**
- Relaciones dimensionales.
 - Cubos.
 - Cruzamientos específicos.

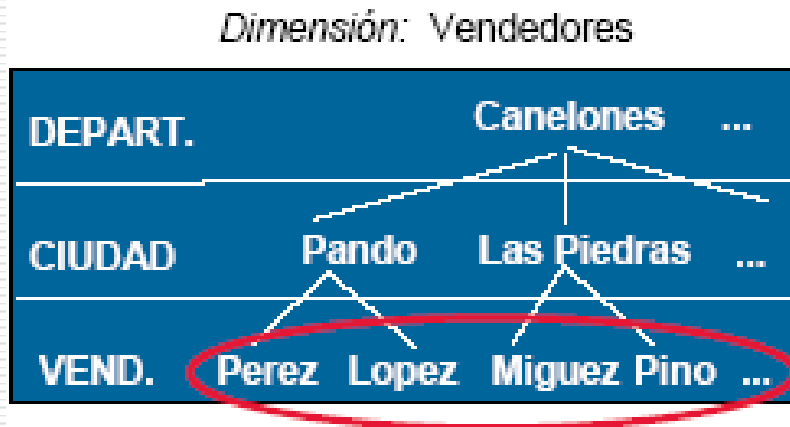
MODELO CMDM

Dimensiones:



MODELO CMDM

Niveles:



Los datos pueden ser no atómicos.



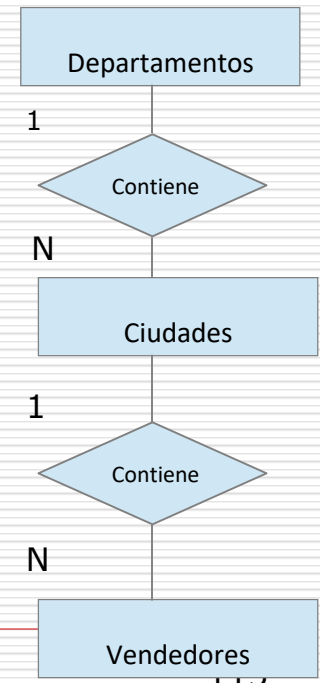
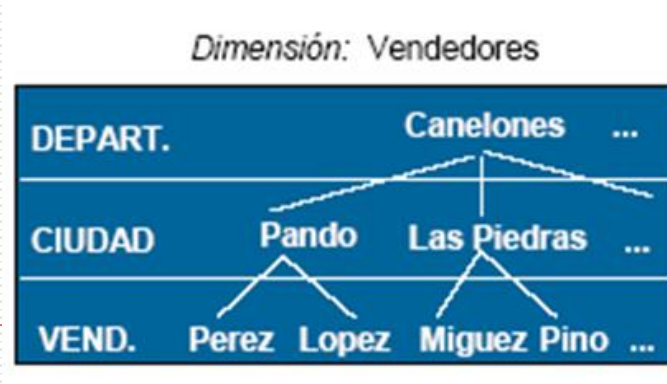
Nivel: Vendedor

VEND.	Id vend: 5376 Apellido: 'Perez' Nombre: 'Juan' Edad: 24	Id vend: 376 Apellido: 'Pino' Nombre: 'Jose' Edad: 55	...

MODELO CMDM

Jerarquías:

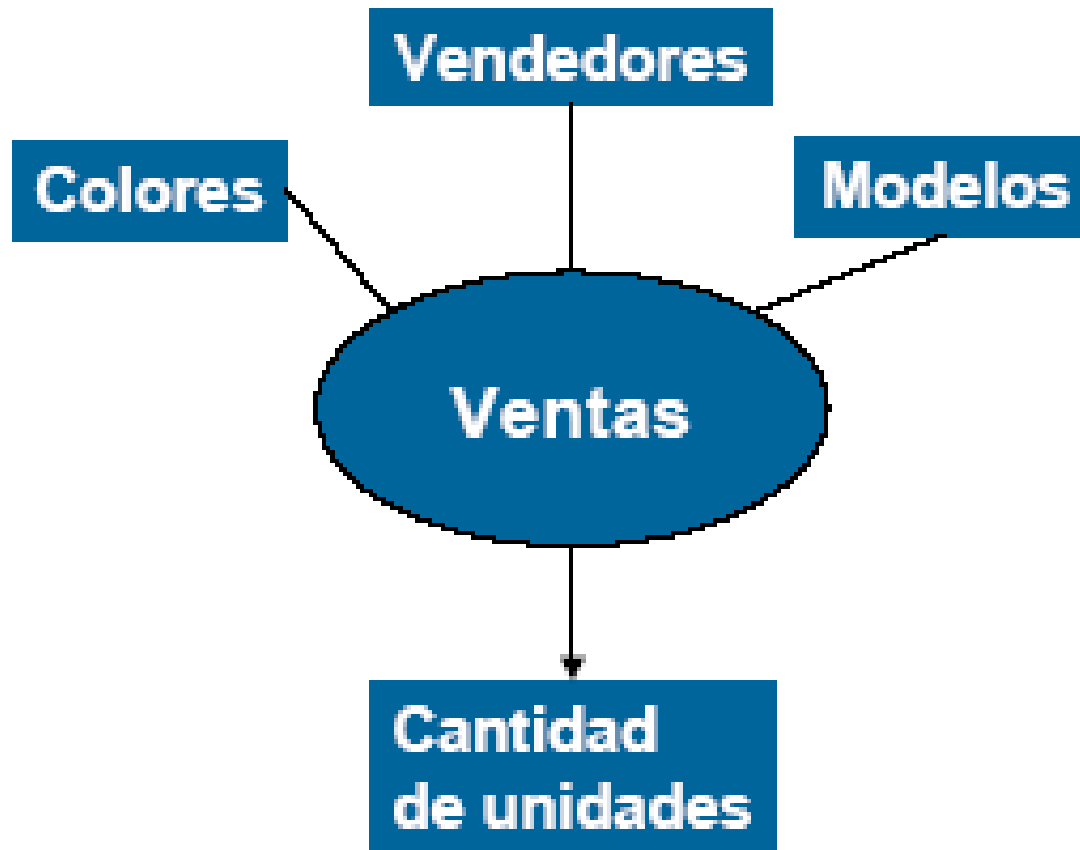
- Los niveles se organizan en **jerarquías**.
- Cada jerarquía está compuesta por uno o varios niveles.
- En cada jerarquía:
Se tiene una relación $<1-n>$ entre objetos de nivel superior e inferior



MODELO CMDM

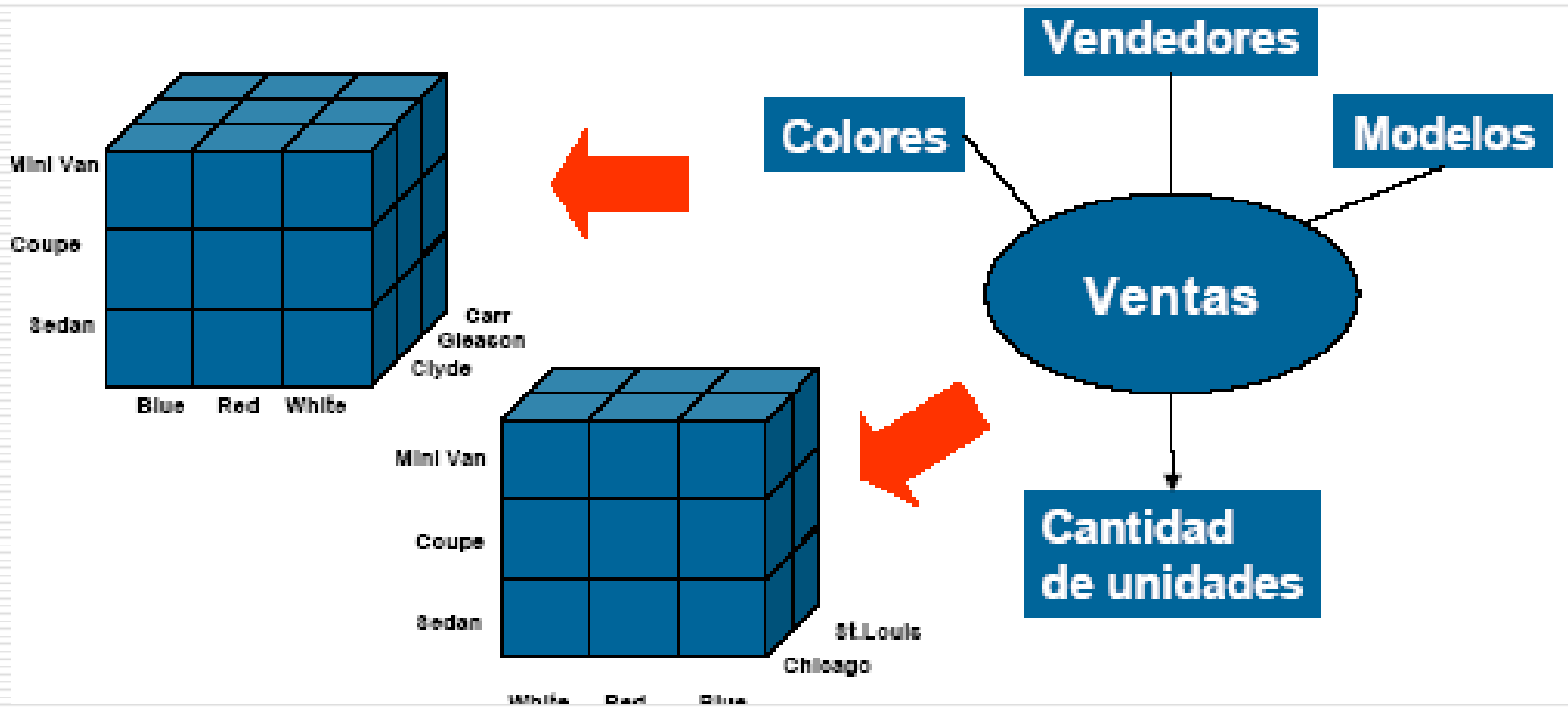
- ☐ Relaciones Dimensionales:
 - Representan cruzamientos entre Dimensiones.
 - Las Medidas participan como Dimensiones.
 - Vista como una relación:
 - ☐ Se tiene un elemento en el conjunto relación si y solo si hay un cruzamiento.
 - ☐ Esto obliga a que las Dimensiones participantes realmente sean cruzables.

MODELO CMDM

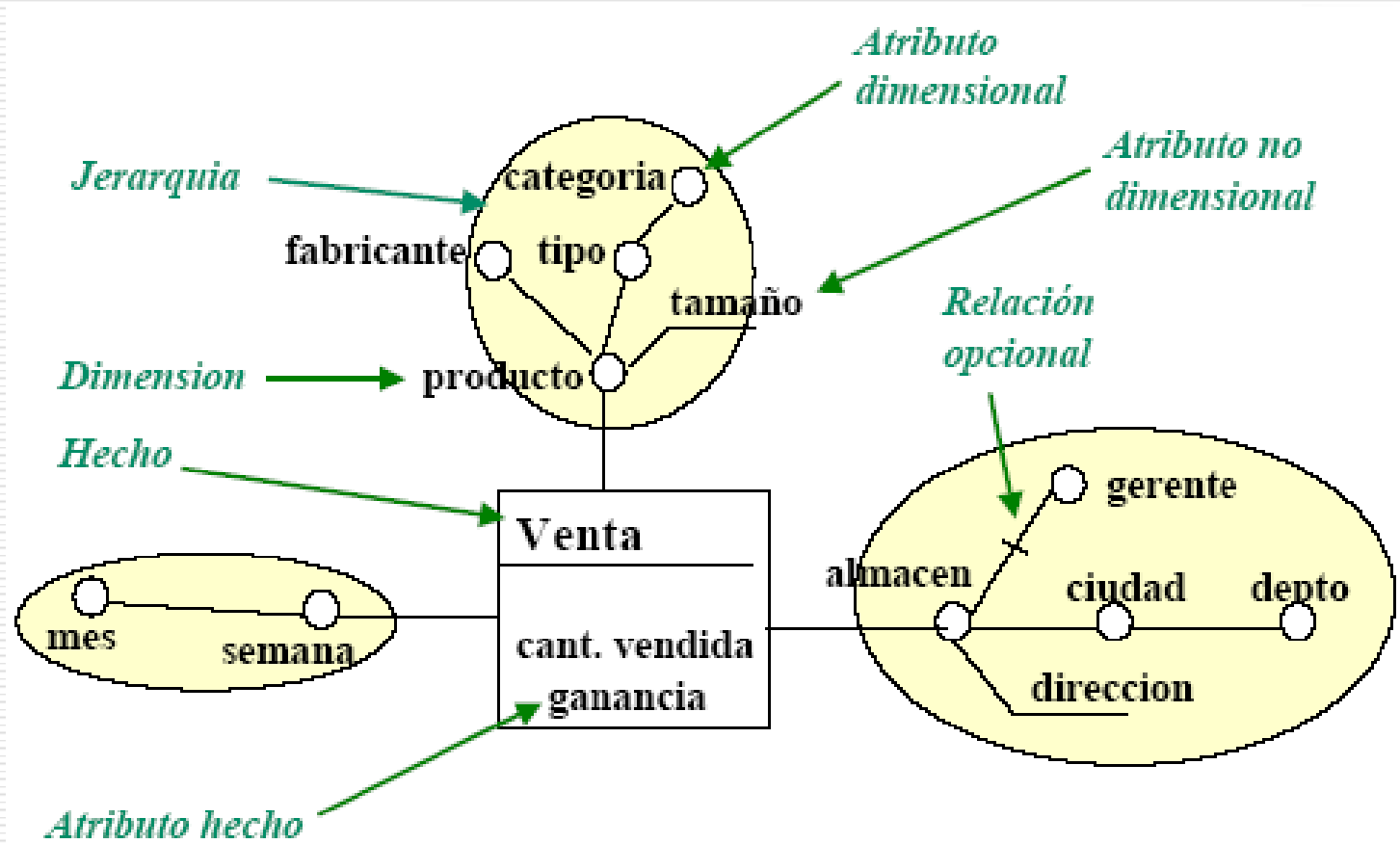


MODELO CMDM

■ Cubos: Ejemplo.

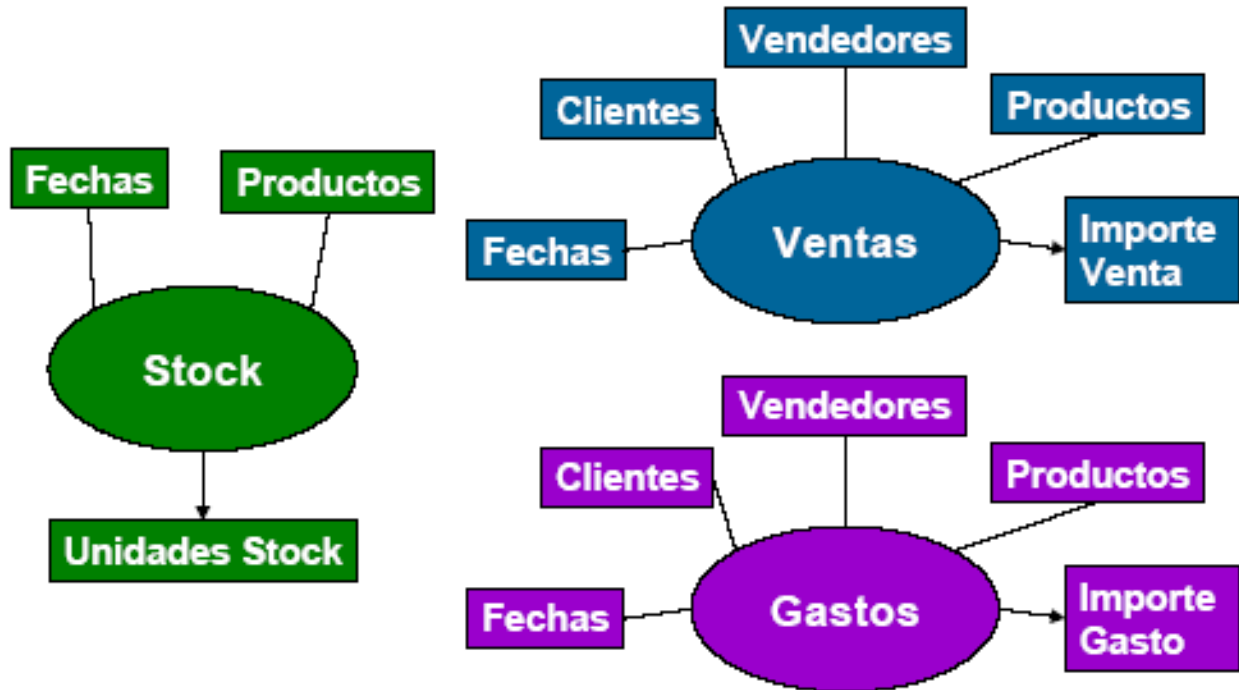


Orientación a datos



Relaciones Dimensionales

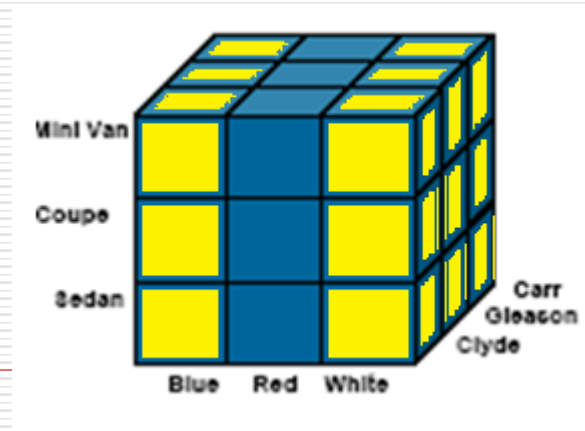
Ejemplos de distintas Medidas:



Operaciones – Slice (Rebanada)

- ❑ Slice (Rebanad):
Permite restringir los valores asociados a una dimensión del cubo, es decir toma un subconjunto de valores para la dimensión que se hace slicing.

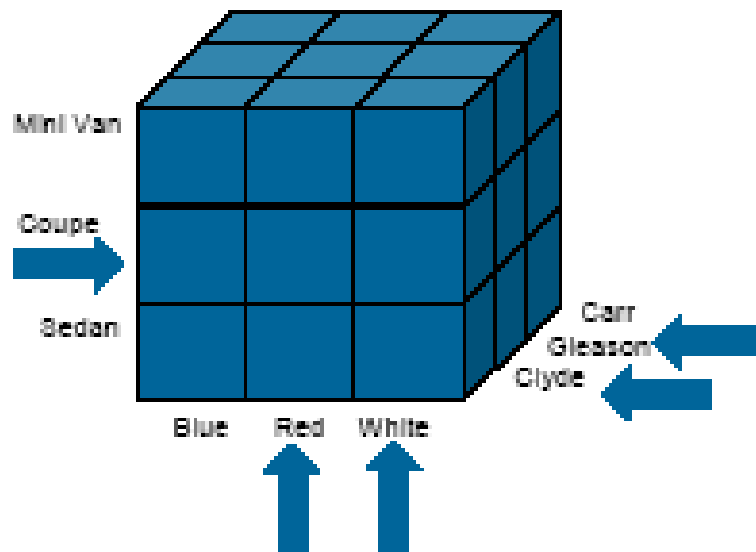
Ej reducir con Slice a solo colores Blue o White.



Operaciones – Dice

❑ Dice (Dado)

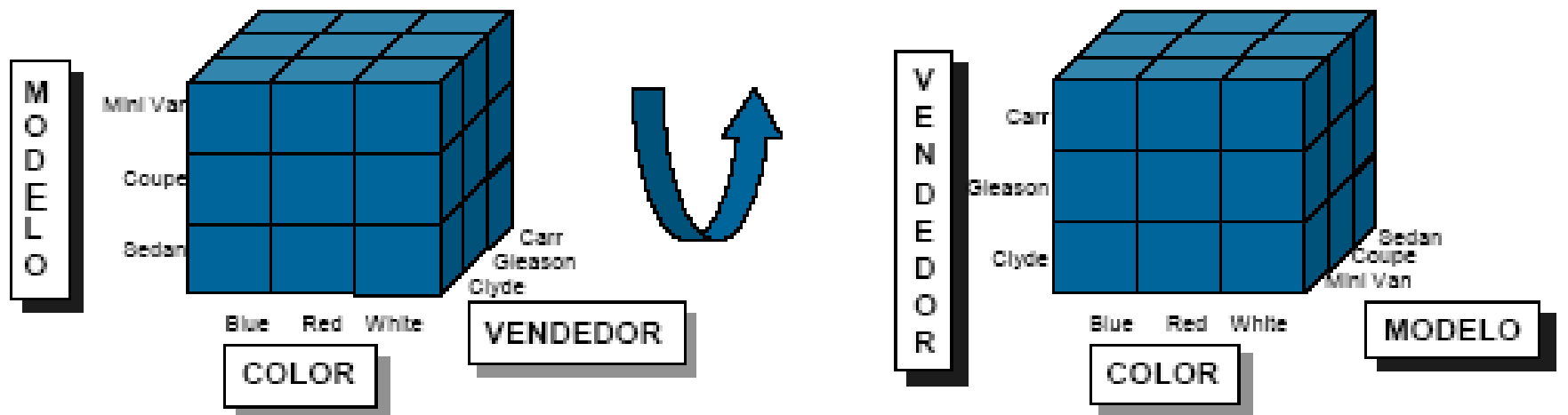
- ❑ Es un corte en mas de una dimensión, formando un subconjunto (subcubo).



Operaciones – Pivot

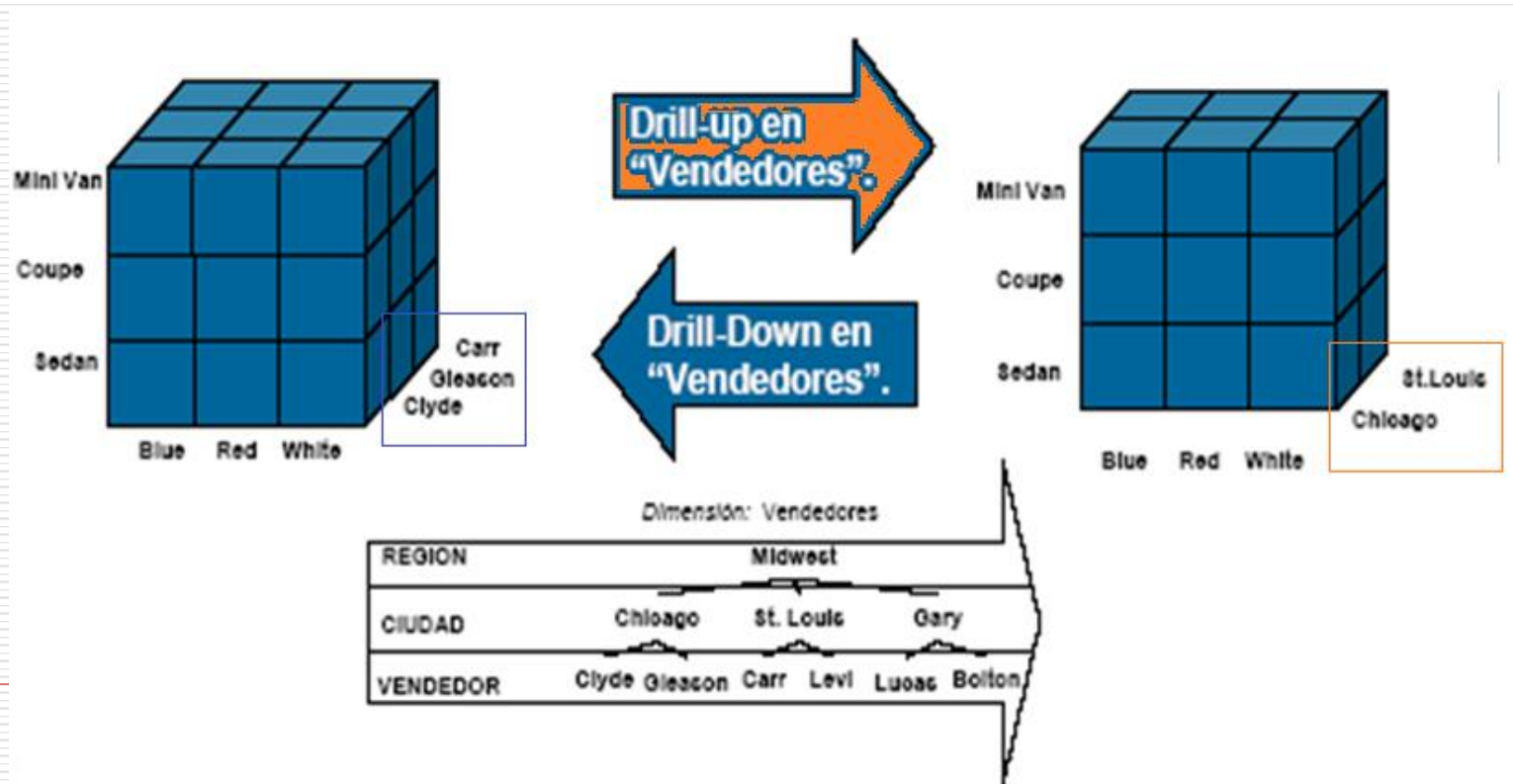
❑ Pivot (Rotación).

- ❑ Selecciona el orden de visualización de las dimensiones.



Operaciones: Drill-up, drill-down

- Movimientos en la Jerarquía de una Dimensión (Drill-up agrega, Drill-down desagrega)



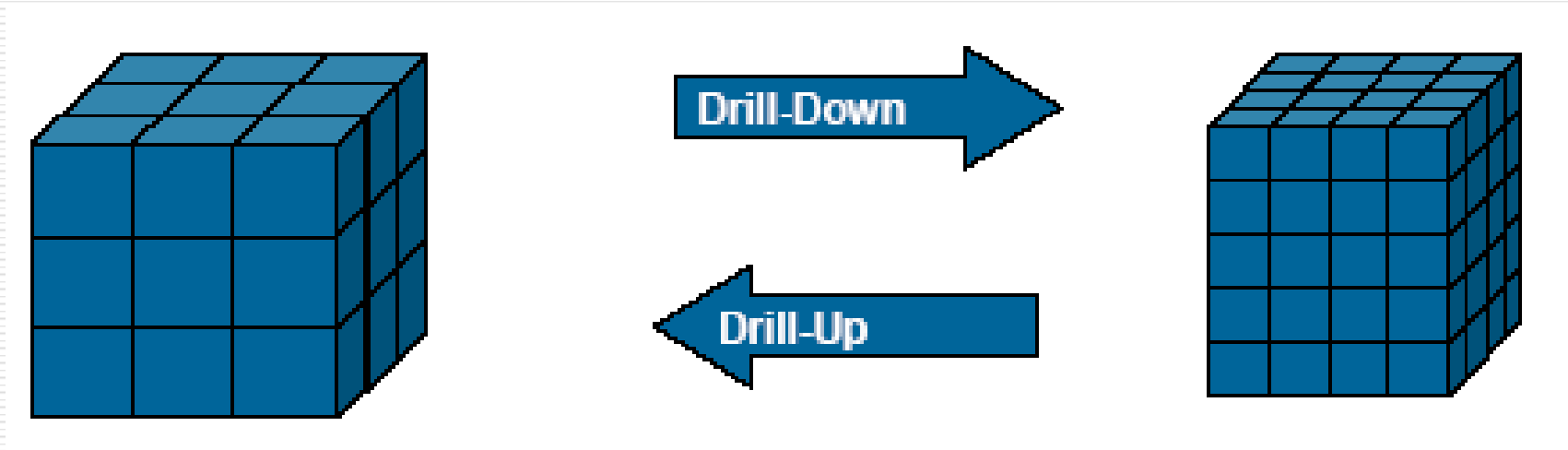
Operaciones: Drill-up, drill-down

Drill-up agrega medidas que van de un nivel N_i a un nivel más general N_j de una dimensión.

Drill-Down es la operación inversa. A partir de un nivel superior, me permite bajar de nivel.

Operaciones: Drill-up, drill-down

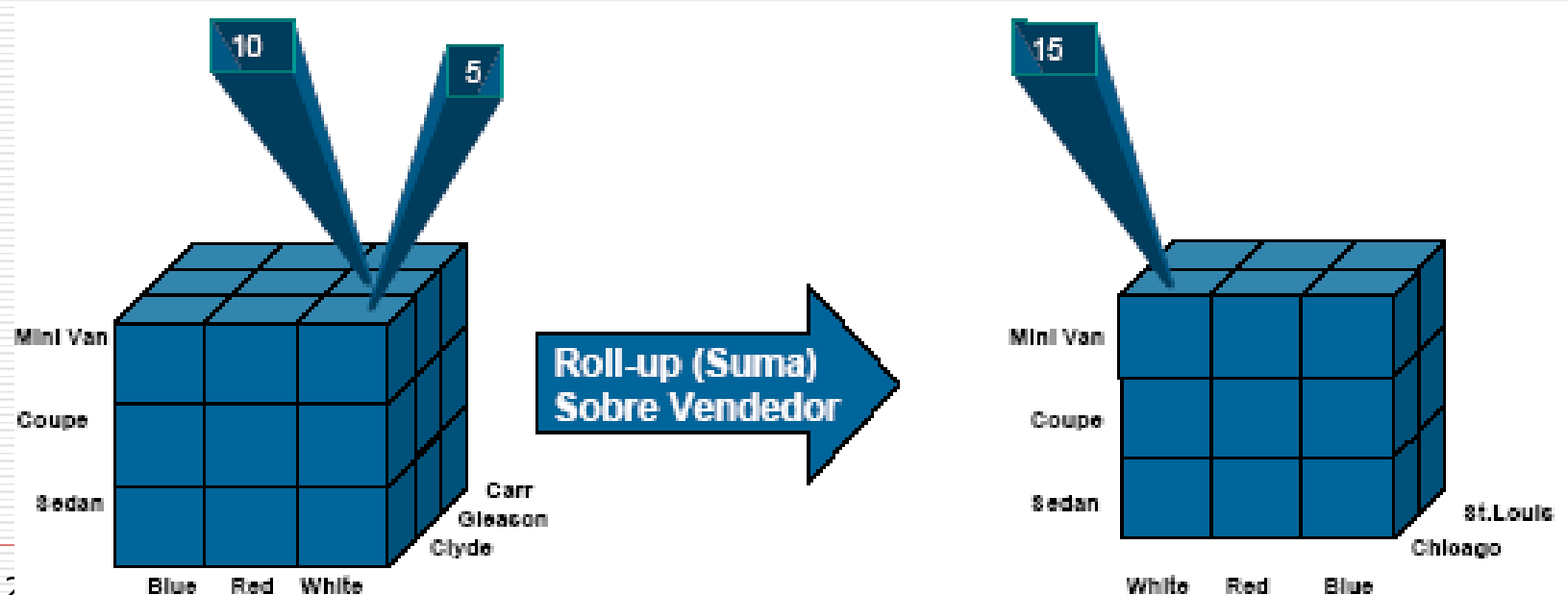
- ❑ Drill-Up o Drill-Down pueden verse como ajuste en las escalas de los ejes.
- ❑ Son agrupamientos y des-agrupamientos.



Operaciones: Roll-up

❑ Consolidación (Roll-Up).

- ❑ Calcula las medidas en función de agrupamientos.
- ❑ Realiza el re-cálculo de la medida de acuerdo a los ajustes de escala.



Herramientas

Existen distintos tipos de herramientas, comerciales y open source.

- ❑ Comerciales

- ❑ Microsoft → Análisis Services → Powe Bi

- ❑ Oracle → 10g, 11g, ...

- ❑ IBM → Cognos Powerplay transformer

- ❑ Microsoft Excel cuenta con funciones básicas.

- ❑ Open Source

- ❑ Pentaho

- ❑ Spago BI

Extensiones a GROUP BY con capacidades para OLAP

- ❑ SQL 1999 extiende el agrupamiento agregando los siguientes modificadores que otorgan capacidades de OLAP (Procesamiento Analítico OnLine):
 - ✓ **GROUPING SET**
 - ✓ **ROLLUP**
 - ✓ **CUBE**

- ❑ Las tres agregan información redundante comparado con GROUP BY básico.

Extensiones a GROUP BY Básico

SELECT modelo, color,
sum(importe) as monto

FROM ventacrudo

GROUP BY

modelo, color

GROUP BY básico		
modelo	color	monto
Sports Coupe	Red	46900
Mnivan	White	37800
Sports Coupe	White	52000
Mnivan	Red	46800
Sedan	White	19600
Sedan	Blue	39800
Sedan	Red	27000
Mnivan	Blue	62600
Sports Coupe	Blue	28100

Extensiones a GROUP BY

Básico

SELECT modelo, color,
sum(importe) as monto

FROM ventacrudo

GROUP BY

GROUPING SETS

((modelo),(color))

GROUP BY Grouping Sets		
modelo	color	monto
Mnivan		147200
Sports Coupe		127000
Sedan		86400
	White	109400
	Red	120700
	Blue	130500

Extensiones a GROUP BY Básico

```
SELECT modelo, color,  
        sum(importe)  
        as monto  
FROM ventacrudo  
GROUP BY  
ROLLUP  
((modelo),(color))
```

GROUP BY Rollup		
modelo	color	monto
		360600
Sports Coupe	Red	46900
Mnivan	White	37800
Sports Coupe	White	52000
Mnivan	Red	46800
Sedan	White	19600
Sedan	Blue	39800
Sedan	Red	27000
Mnivan	Blue	62600
Sports Coupe	Blue	28100
Mnivan		147200
Sports Coupe		127000
Sedan		86400

Extensiones a GROUP BY CUBE

```
SELECT modelo, color,  
        sum(importe) as monto  
FROM ventacrudo  
GROUP BY  
CUBE  
        ((modelo), (color))
```

GROUP BY CUBE

modelo	color	monto
		360600
Sports Coupe	Red	46900
Mnivan	White	37800
Sports Coupe	White	52000
Mnivan	Red	46800
Sedan	White	19600
Sedan	Blue	39800
Sedan	Red	27000
Mnivan	Blue	62600
Sports Coupe	Blue	28100
Mnivan		147200
Sports Coupe		127000
Sedan		86400
	White	109400
	Red	120700
	Blue	130500

Suma de Importe	Etiquetas de columna ▾			
Etiquetas de fila ▾	Blue	Red	White	Total general
Mnivan	62600	46800	37800	147200
Sedan	39800	27000	19600	86400
Sports Coupe	28100	46900	52000	127000
Total general	130500	120700	109400	360600

Diseño de un DW Relacional

☐ Modelo Dimensional

■ Tablas de hechos (**fact tables**)

- ☐ donde se guardan las medidas numéricas del negocio
- ☐ Intersección de todas las dimensiones
- ☐ granularidad
- ☐ clave compuesta (la combinación de las fk)

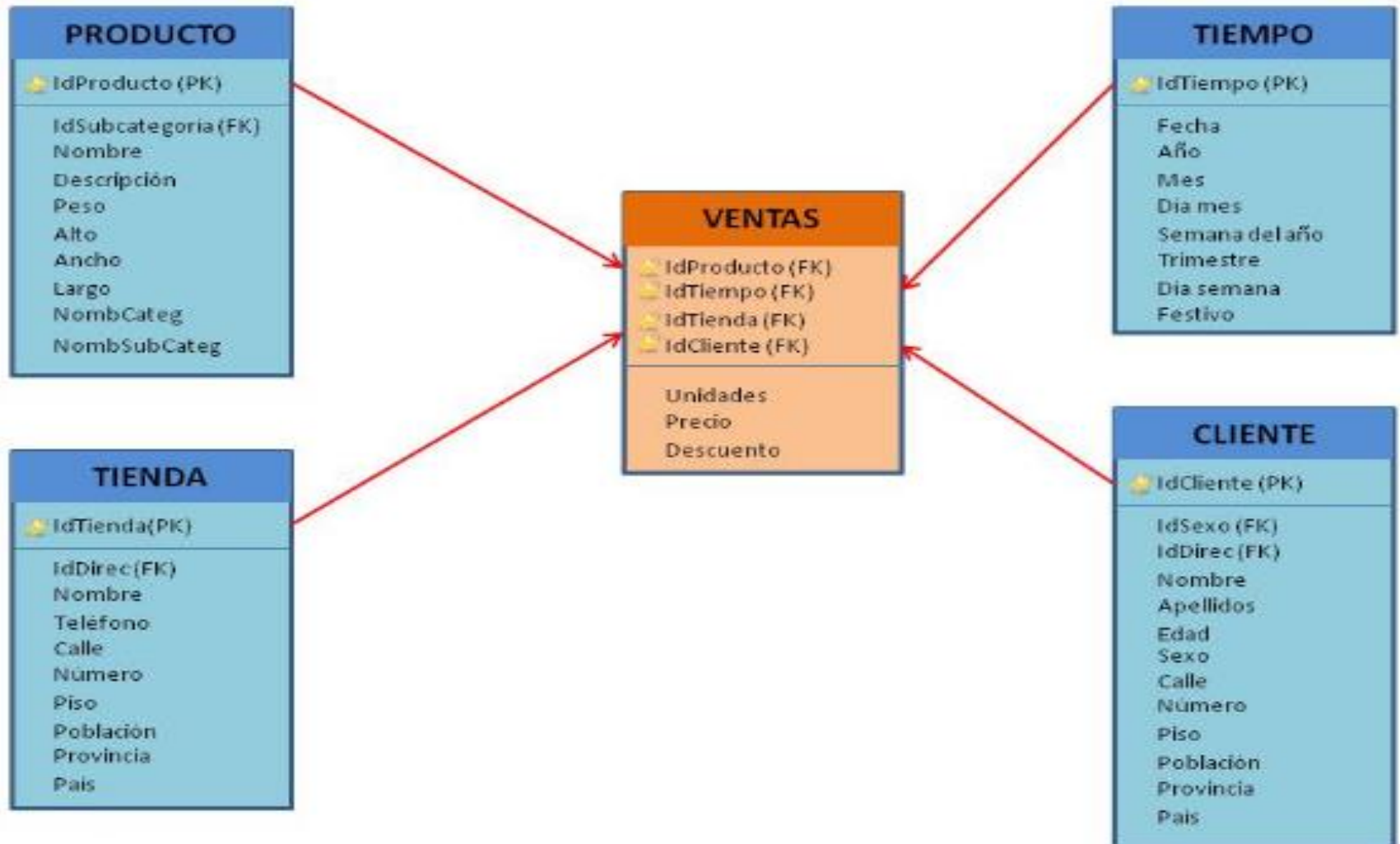
■ Tablas de dimensión (**dimension tables**)

- ☐ donde se guardan las descripciones textuales de las dimensiones del negocio
- ☐ Jerarquías: desnormalizadas o normalizadas

Tipos de esquemas en el MD-Rel – Estrella

- ❑ También llamado **dimensional**, contiene una tabla de hechos que es aquella que contiene **toda la información** y tiene varias **tablas de dimensiones** que contienen el **catálogo** de la información.
- ❑ Se asemeja mucho a un base de datos desnormalizada.

Tipos de esquemas – **Modelo Estrella**



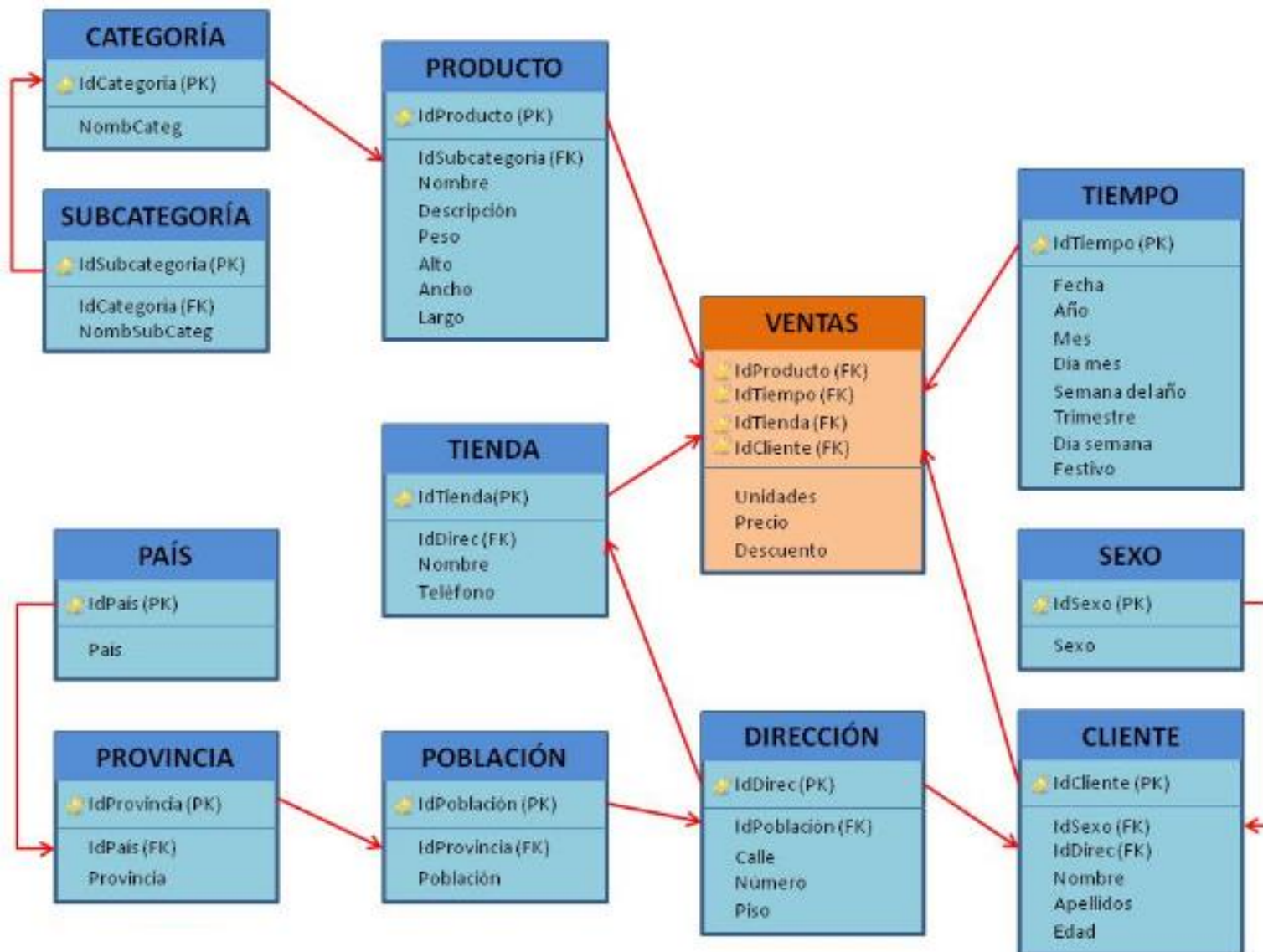
Tipos de esquemas – **Modelo Estrella**

- ❑ El esquema de estrella **sólo tiene un nivel**.
- ❑ El tener todos los hechos juntos y las dimensiones separadas permite que las “joins” **sean mínimos** ocupando **menos tiempo las consultas**.
- ❑ La tabla de hechos es generalmente más grande en atributos y tuplas que las de dimensiones.

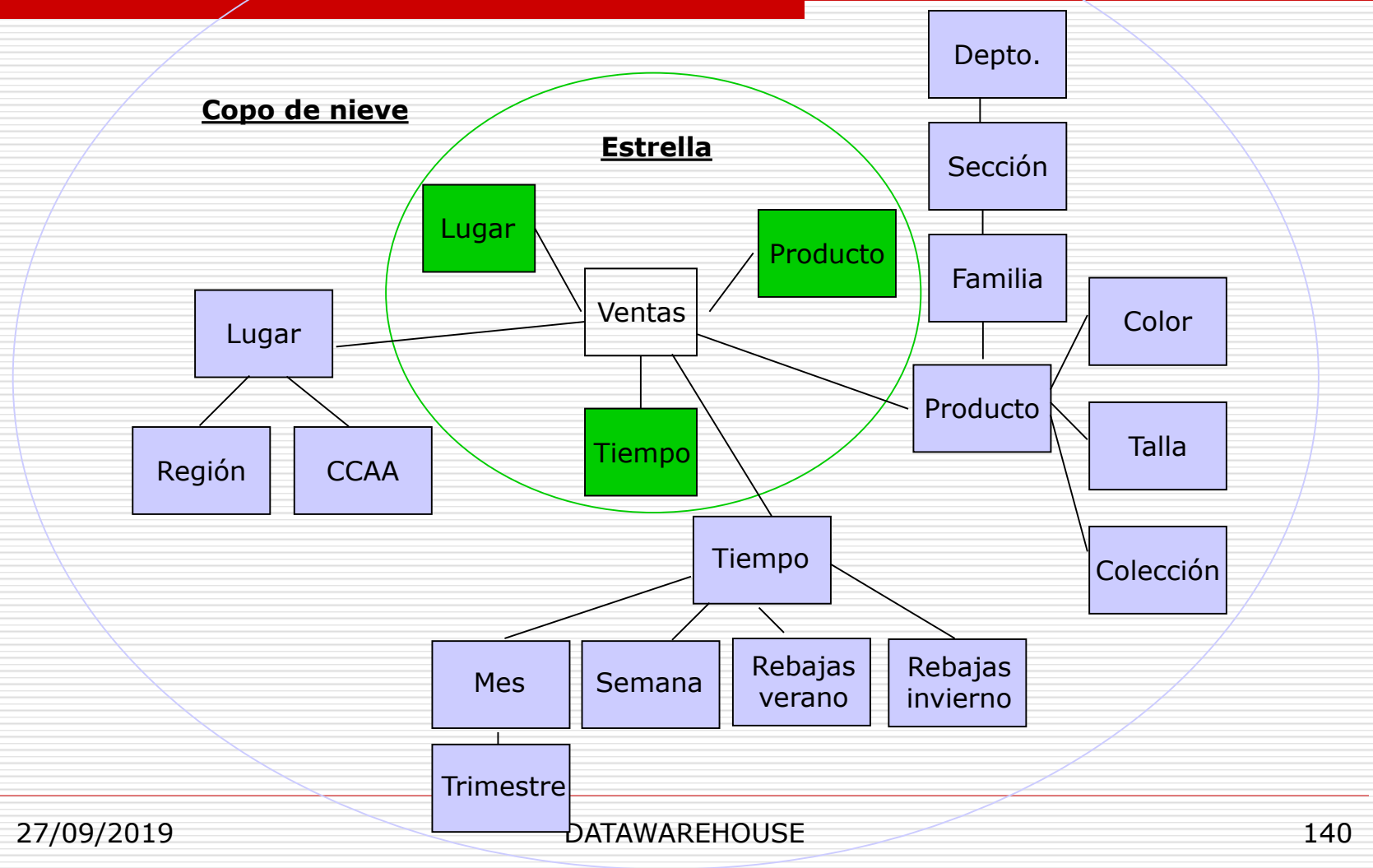
Tipos de esquemas – **Modelo Copo de nieve**

- ❑ Es una variante del esquema de estrella (**desnormalizado**).
Normaliza todas las tablas de dimensiones.
- ❑ Tiene algunas mejoras de espacio pero en ocasiones las consultas son más lentas.

Tipos de esquemas – Modelo Copo de nieve



Modelo Copo de nieve



Pasos para el diseño del modelo de datos

- Paso 1. Tomar un “proceso” de la organización para modelar.
- Paso 2. Decidir el gránulo (nivel de detalle) de representación del proceso.
- Paso 3. Identificar las dimensiones que caracterizan el proceso.
- Paso 4. Decidir la información a almacenar sobre el proceso.

Pasos para el diseño

□ Paso 1. Elegir un “*proceso*” de la organización para modelar.

■ *Proceso*: actividad de la organización soportada por un OLTP del cual se puede extraer información con el propósito de construir el almacén de datos.

- *Pedidos (de clientes)*
- *Compras (a suministradores)*
- *Facturación*
- *Envíos*
- *Ventas*
- *Inventario*
- ...

Pasos para el diseño

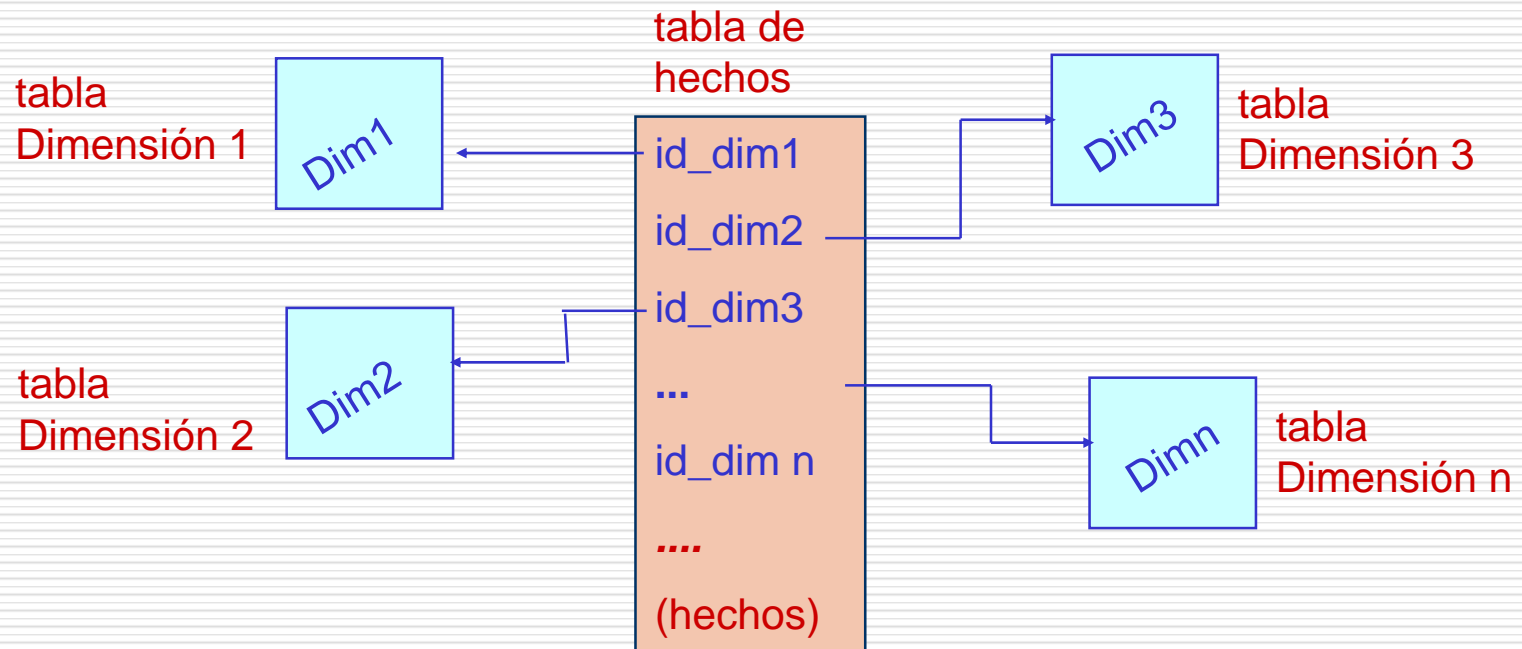
- **Ejemplo:** Cadena de supermercados.
 - Cadena de supermercados con 300 almacenes en la que se venden unos 30.000 productos distintos.

- **Actividad:** *Ventas.*
 - La actividad a modelar son las ventas de productos en los depositos de la cadena.

Pasos para el diseño

- Paso 2. Decidir el gránulo (nivel de detalle) de representación.
- **Gránulo**: es el nivel de detalle al que se desea almacenar información sobre la actividad a modelar.
 - El **gránulo** define el nivel atómico de datos en el almacén de datos.
 - El **gránulo** determina el significado de las tuplas de la **tabla de hechos**.
 - El **gránulo** determina las **dimensiones básicas** del esquema
 - *transacción en el OLTP*
 - *información diaria*
 - *información semanal*
 - *información mensual.*

Pasos para el diseño



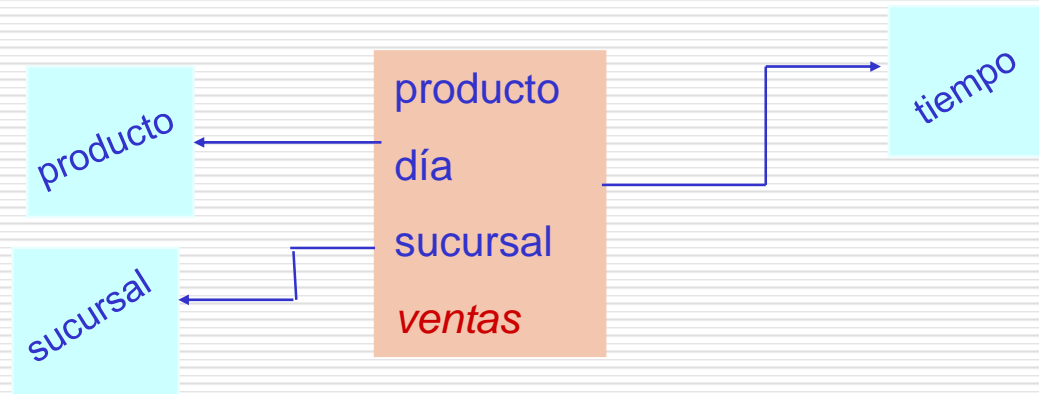
Pasos para el diseño

Ejemplo: Cadena de supermercados.

Gránulo: “se desea almacenar información sobre las ventas diarias de cada producto en cada sucursal de la cadena”.

Gránulo:

define el significado de las tuplas de la tabla de hechos.
determina las dimensiones básicas del esquema.



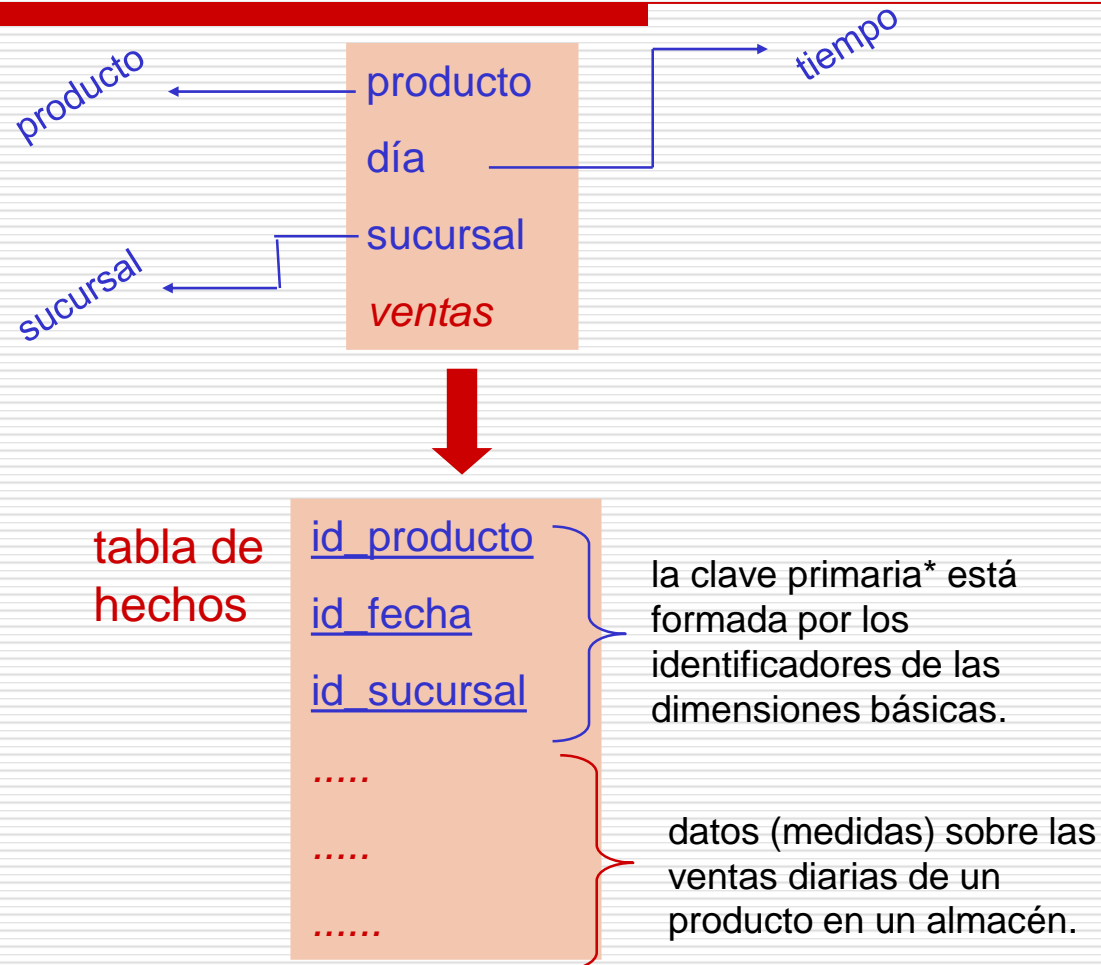
Pasos para el diseño

Gránulo inferior: no se almacena información a nivel de **línea de ticket** porque no se puede identificar siempre al cliente de la venta lo que permitiría hacer análisis del comportamiento (hábitos de compra) del cliente. El inferior puede ser entonces **día**.

Gránulo superior: no se almacena información a nivel **semanal** o **mensual** porque se perderían opciones de análisis interesantes: ventas en días previos a vacaciones, ventas en fin de semana, ventas en fin de mes,

En un almacén de datos se almacena información a un nivel de detalle (gránulo) fino porque se vaya a interrogar el almacén a ese nivel sino porque ello permite clasificar y estudiar (analizar) la información desde muchos puntos de vista.

Pasos para el diseño



Pasos para el diseño

Paso 3. Identificar las dimensiones relevantes del proceso.

- ✓ **Dimensiones**: dimensiones que caracterizan la actividad al nivel de detalle (gránulo) que se ha elegido.

Tiempo (dimensión temporal: ¿cuándo se produce la actividad?)

Producto (dimensión ¿cuál es el objeto de la actividad?)

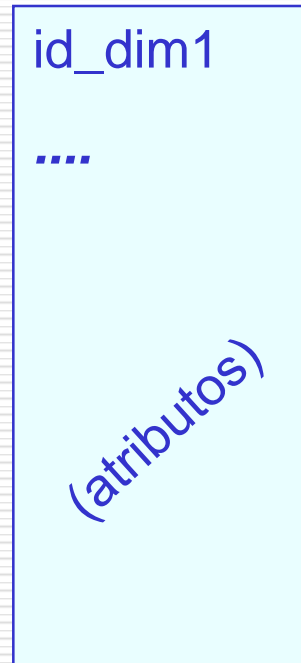
Sucursal (dimensión geográfica: ¿dónde se produce la actividad?)

Cliente/Proveedor (dimensión ¿quién es el destinatario o proveedor de la actividad?)

- ✓ De cada **dimensión** se debe decidir los atributos (propiedades) relevantes para el análisis de la actividad.
- ✓ Entre los atributos de una dimensión existen jerarquías naturales que deben ser identificadas (**día-mes-año**)

Diseño de un Almacén de Datos

tabla
Dimensión 1



Diseño de un Almacén de Datos

Ejemplo: Cadena de supermercados.



Nota: En las aplicaciones reales el número de dimensiones suele variar entre 3 y 15 dimensiones.

Diseño de un Almacén de Datos

Dimensión Tiempo:

- ✓ dimensión presente en todo AD porque el AD contiene información histórica sobre la organización.
- ✓ aunque el lenguaje SQL ofrece funciones de tipo DATE, una dimensión Tiempo permite representar otros atributos temporales no calculables en SQL.
- ✓ se puede calcular de antemano
- ✓ atributos frecuentes:
 - nro. de día, nro. de semana, nro. de año: valores absolutos del calendario juliano que permiten hacer ciertos cálculos aritméticos.
 - día de la semana (lunes, martes, miércoles,...): permite hacer análisis sobre días de la semana concretos (ej. ventas en sábado, ventas en lunes,...).

Diseño de un Almacén de Datos

Dimensión Tiempo:

✓ atributos frecuentes:

- día del mes (1..31): permite hacer comparaciones sobre el mismo día en meses distintos (ventas el 1º de mes). → [función extract](#)

- marca de fin de mes, marca de fin de semana : permite hacer comparaciones sobre el último día del mes o días de fin de semana en distintos meses. [UDF](#)
- trimestre del año (1..4): permite hacer análisis sobre un trimestre concreto en distintos años. [UDF](#)
- marca de día festivo: permite hacer análisis sobre los días contiguos a un día festivo.
 - [Tabla específica del AD de Días Festivos y UDF](#)
- estación (primavera, verano..) [UDF](#)
- evento especial: permite marcar días de eventos especiales (final de futbol, elecciones...) [Tabla específica de Días Festivos y UDF](#)

✓ jerarquía natural: → día - mes - trimestre –año – lustro - década

Diseño de un Almacén de Datos

Dimensión Tiempo: Para granos Fecha, cada fila tiene la información



Diseño de un Almacén de Datos

Dimensión Tiempo: Implementaciones

Estática → cargada por la ETL

ID (PK)
Fecha
Año
Mes
Dia
DiaSemana
Trimestre
DiadelAño

Dinámica → Obtenidas por UDF

Alt 1

ID (PK)
Fecha

Alt2 (sin tab Dimen)

id_dim1
id_dim2
Fecha
id_dimn

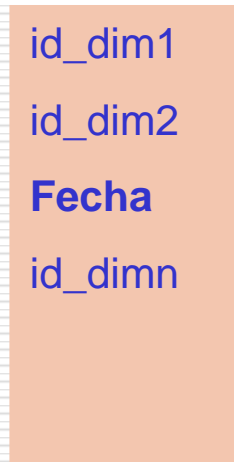
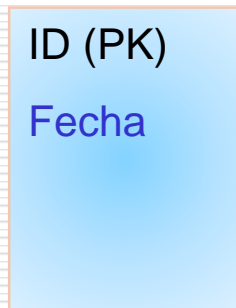
Diseño de un Almacén de Datos

Dimensión Tiempo: Implementación **SQLSERVER**

Dinámica → Obtenidas por UDF Funciones Usadas Dim Tiempo

Alt 1

Alt2 (sin tab Dimen)



→ SqlServer
datepart

<i>datepart</i>	Abbreviations	
year	yy, yyyy	
quarter	qq, q	
month	mm, m	
dayofyear	dy, y	
day	dd, d	
week	wk, ww	
weekday	dw	
hour	hh	
minute	mi, n	
second	ss, s	
millisecond	ms	
microsecond	mcs	
nanosecond	ns	
TZoffset	tz	156
ISO_WEEK	isowk, isoww	

Diseño de un Almacén de Datos

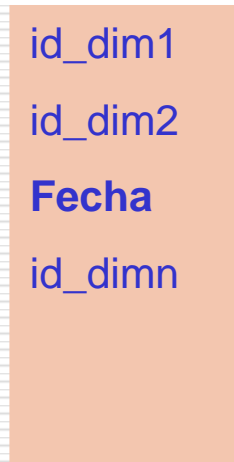
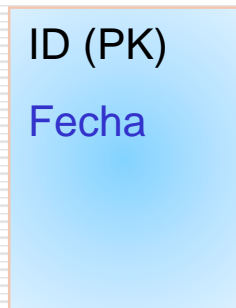
Dimensión Tiempo: Implementación PostgreSQL

Dinámica → Obtenidas por UDF

Funciones Usadas Dim Tiempo

Alt 1

Alt2 (sin tab Dimen)



→ PostgreSQL
extract

<i>extract</i>	detalle
year	Año yy, yyyy
quarter	Trimestre qq, q
month	Mes mm, m
Dayofyear doy	Día del año dy, y
Day	Día dd, d
week	Semana del año wk
weekday	Día semana dw
hour	hh
minute	mi, n
second	ss, s
millisecond	ms
microsecond	mcs
nanosecond	ns
TZoffset	tz 157
ISO_WEEK	isowk, isoww

Diseño de un Almacén de Datos

Dimensión Producto:

- ✓ la dimensión Producto se define a partir de la tabla de productos del sistema OLTP.
- ✓ las actualizaciones de la tabla de productos deben reflejarse en la dimensión Producto (¿cómo?).
- ✓ la dimensión Producto debe contener el mayor número posible de atributos descriptivos que permitan un análisis flexible. Un número frecuente podría ser de 50 atributos (según grado tabla OLTP) .
- ✓ atributos frecuentes: identificador (código estándar), descripción, tamaño del envase, marca, categoría, departamento, tipo de envase, producto dietético, peso, unidades de peso, unidades por envase, fórmula, ...
- ✓ jerarquías: producto-categoría-departamento

Diseño de un Almacén de Datos

Dimensión Establecimiento (*store*) - Espacio:

- ✓ la dimensión Sucursal representa la información geográfica básica.
- ✓ esta dimensión suele ser creada explícitamente recopilando información *externa* que sólo tiene sentido en el A.D y que no la tiene en un OLTP (número de habitantes de la ciudad del establecimiento, caracterización del tipo de población del distrito, ...)
- ✓ atributos frecuentes: identificador (código interno), nombre, dirección, barrio, distrito, región, ciudad, país, director, teléfono, fax, tipo de almacén, superficie, fecha de apertura, fecha de la última remodelación, superficie para congelados, superficie para productos frescos, georeferenciación, datos de la población del distrito, zona de ventas, ...
- ✓ jerarquías:
 - establecimiento – barrio - distrito - ciudad - región - país (jerarquía geográfica)
 - establecimiento - zona_ventas - región_ventas (jerarquía de ventas)

Diseño de un Almacén de Datos

Tiempo

id_fecha

día

semana

mes

año

día_semana

día_mes

trimestre

festivo

....

Establecimiento/Sucursal

id_establec

nro_establec

nombre

dirección

distrito

ciudad

país

tlfno

fax

superficie

tipo_almacén

...

Producto

id_producto

nro_producto

descripción

marca

subcategoría

categoría

departamento

peso

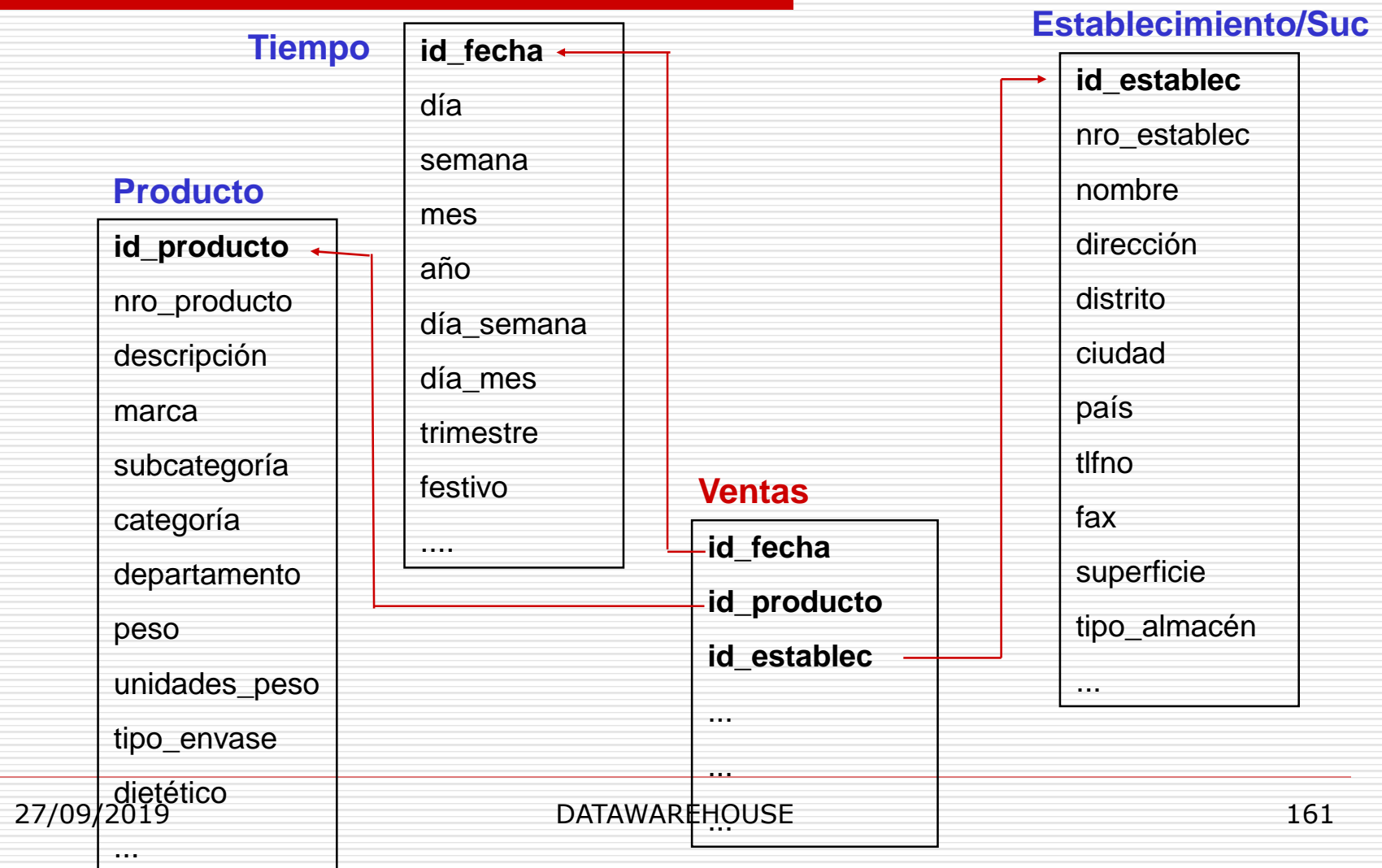
unidades_peso

tipo_envase

dietético

...

Diseño de un Almacén de Datos



Consejos sobre Dimensiones

¿Cómo detectamos las Dimensiones?

La principal dimensión se debe buscar en el objetivo de la organización.

- Para el caso de un supermercado → vender → Producto
- Para el caso de una Fabrica → producir → Producto / Servicio

Prácticamente todos los sistemas cuentan con las dimensiones:

- Tiempo
- Distribución geográfica.

Consejos sobre Dimensiones

Resto de dimensiones: Centrarse en los motores de los procesos:

- Productores
- Consumidores.

Otros Actores del modelo:

- Competencia
- y Buscar requerimientos de entes de regulación (Estado, Sindicatos, Entes Reguladores de la actividad del negocio).

Diseño de un Almacén de Datos

Paso 4. Decidir la información a almacenar sobre el proceso.

Hechos: información (sobre la actividad) que se desea almacenar en cada tupla de la tabla de hechos y que será el objeto del análisis.

Precio

Unidades

Importe

....

Nota: algunos datos que en el OLTP coincidirían con valores de atributos de dimensiones, en el almacén de datos pueden representar hechos. (Ejemplo: el precio de venta de un producto).

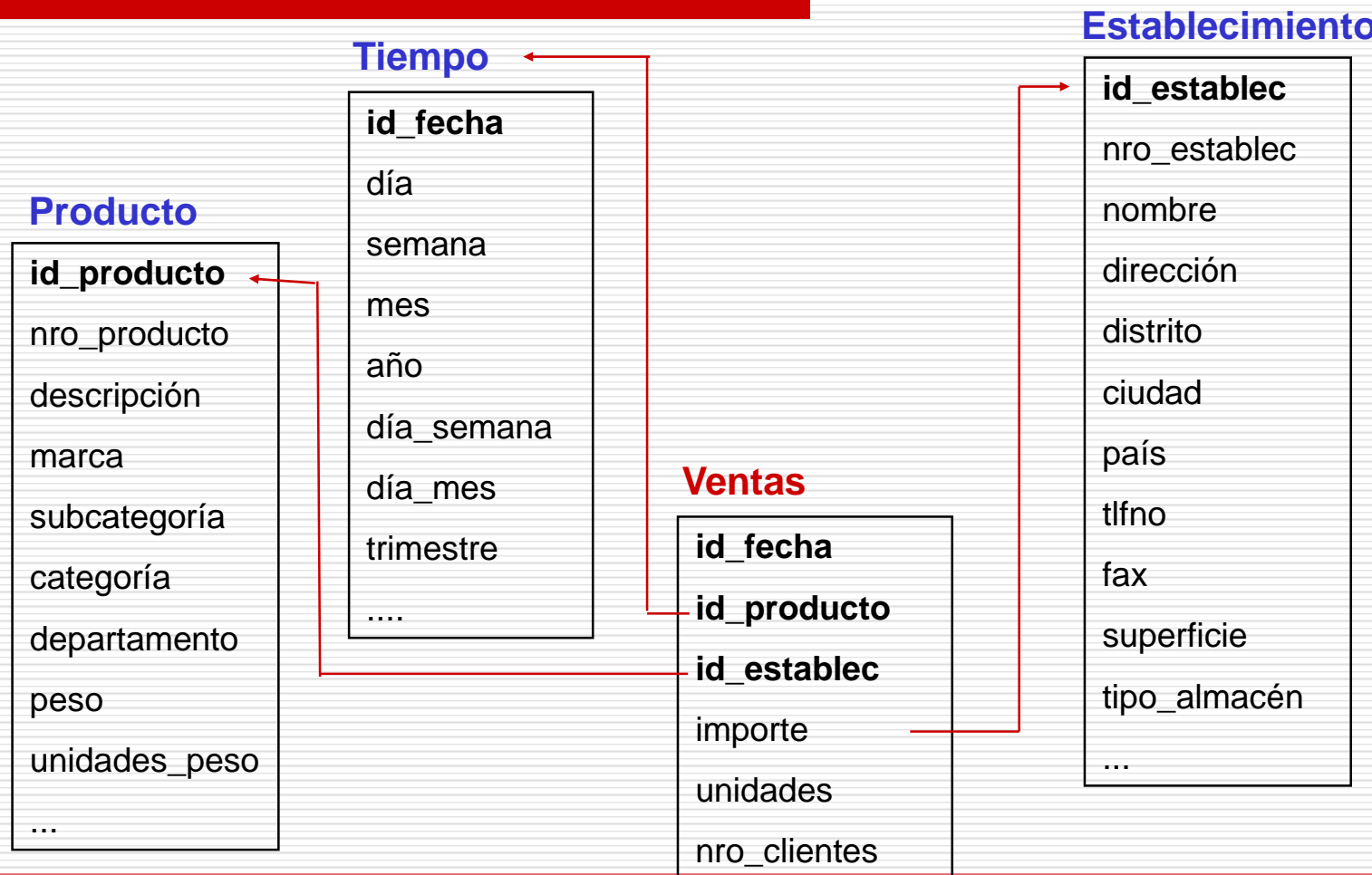
Diseño de un Almacén de Datos

Ejemplo: Cadena de supermercados.

Gránulo: “se desea almacenar información sobre las ventas diarias de cada producto en cada establecimiento de la cadena”.

- importe total de las ventas del producto en el **día**
- cantidad de unidades vendidas del producto en el **día**
- número total de clientes distintos que han comprado el producto en la **semana**.

Diseño de un Almacén de Datos



ETL

El sistema encargado del mantenimiento del almacén de datos es el **Sistema E.T.L (Extracción - Transformación -Carga)**

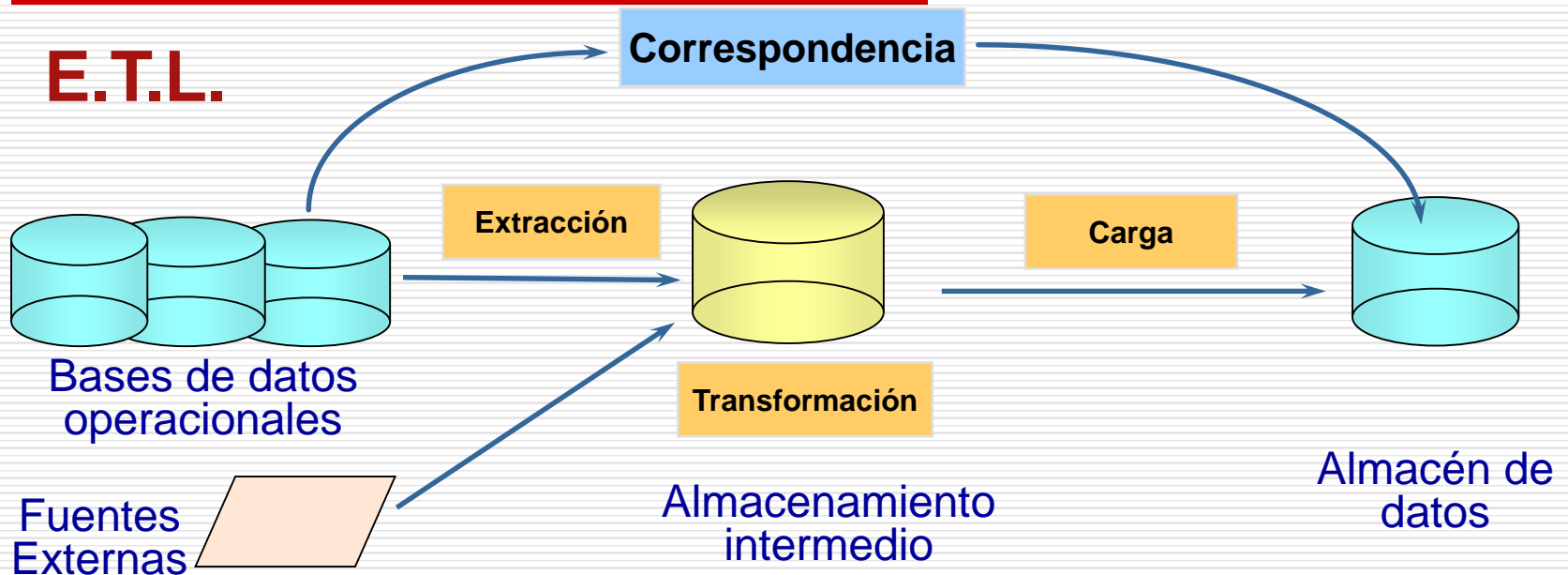
- La construcción del Sistema E.T.L es responsabilidad del equipo de desarrollo del almacén de datos.
- El Sistema E.T.L es construido específicamente para cada almacén de datos. Aproximadamente 50% del esfuerzo.
- En la construcción del E.T.L se pueden utilizar herramientas del mercado o programas diseñados específicamente.

Funciones del Sistema E.T.L:

- Carga inicial. (**initial load**)
- Mantenimiento o *refresco* periódico: inmediato, diario, semanal, mensual,... (**refreshment**)

ETL

E.T.L.

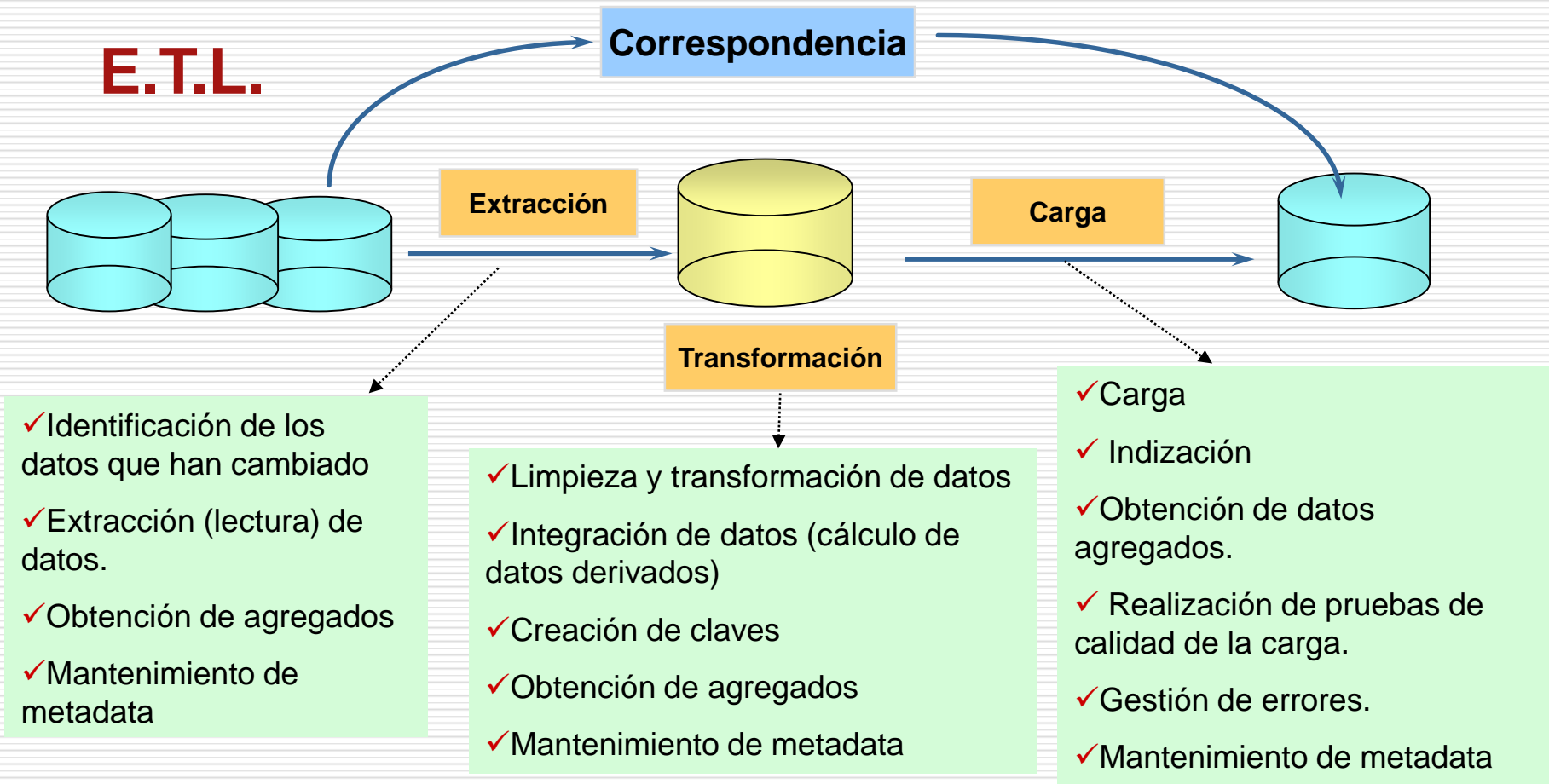


El Almacenamiento intermedio permite (Staging Area):

- Realizar transformaciones sin paralizar las bases de datos operacionales y el DW.
- Almacenar metadatos.
- Facilitar la integración de fuentes externas.

ETL

E.T.L.



ETL – Retroalimentación de calidad

La “calidad de los datos” es la clave del éxito de un almacén de datos.

Definir una estrategia de calidad:

- actuación sobre los sistemas operacionales: modificar las reglas de integridad, los disparadores (triggers) y las aplicaciones de los sistemas operacionales.
- documentación de las fuentes de datos.
- definición de un proceso de transformación.
- nombramiento de un responsable de calidad del sistema (*Data Quality Manager*).

ETL - EXTRACCION

Extracción: lectura de datos del sistema operacional.

- a) durante la carga inicial (**initial load**)
- b) mantenimiento del AD (**refreshment**)

Ejecución de la extracción:

- a) si los datos operacionales están mantenidos en **un SGBDR**, la **extracción** de datos se puede reducir a codificar consultas **en SQL** o rutinas programadas (stored procedure) en la herramienta ETL usada.
- b) si los datos operacionales están en un **sistema propietario** (no se conoce el formato de los datos) **o** en **fuentes externas** textuales, hipertextuales u hojas de cálculo, **la extracción puede ser muy difícil** y puede tener que realizarse a partir de informes o volcados de datos proporcionados por los propietarios que deberán ser procesados posteriormente.

ETL - EXTRACCION

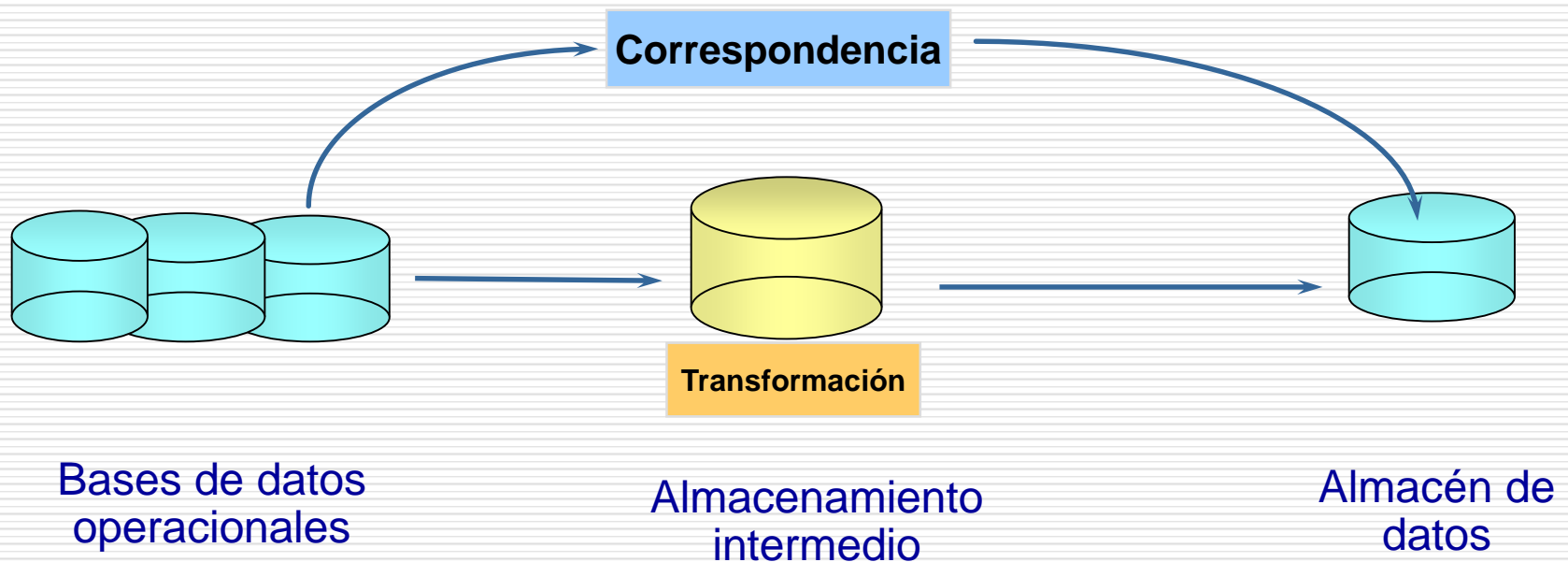
Extracción: en el mantenimiento/refresco del DW. Antes de realizar la extracción es preciso **Identificar los Cambios**.

Identificación de Cambios.

- Identificar los datos operacionales (relevantes) que han sufrido una modificación desde la fecha del último mantenimiento.
- Métodos
 - **Carga total:** cada vez se empieza de cero.
 - **Comparación de instancias** de la base de datos operacional.
 - Uso de **marcas de tiempo** (*time stamping*) en los registros del sistema operacional.
 - Uso de **trigger** en el sistema operacional.
 - Uso del **archivo de log** (gestión de transacciones) del sistema operacional.

ETL - TRANSFORMACION

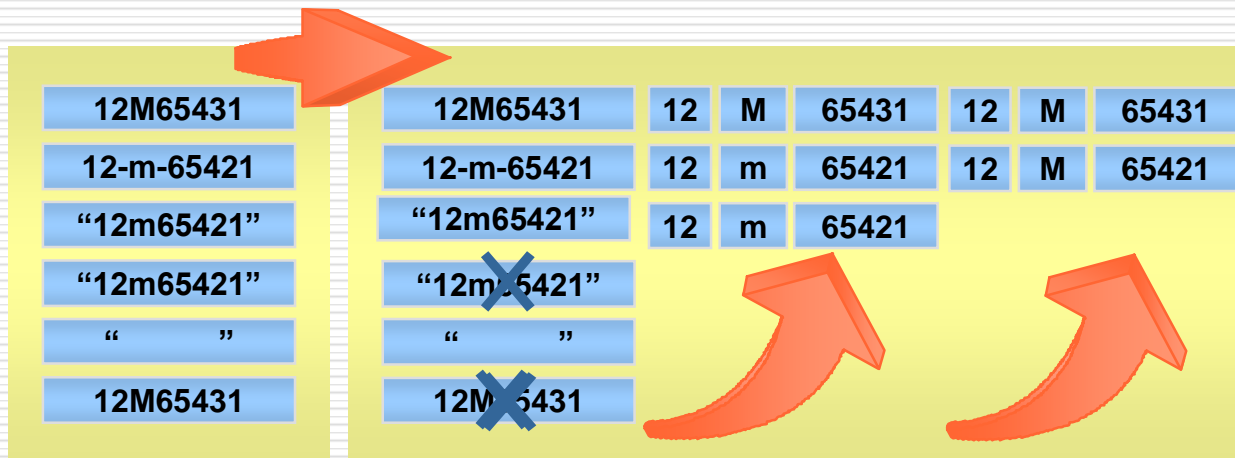
Transformación.



- Transformar los datos extraídos de las fuentes operacionales: limpieza, estandarización. (cleansing)
- Calcular los datos derivados: aplicar las leyes de derivación. (integration)

ETL - TRANSFORMACION

Transformación.



– Eliminar anomalías:

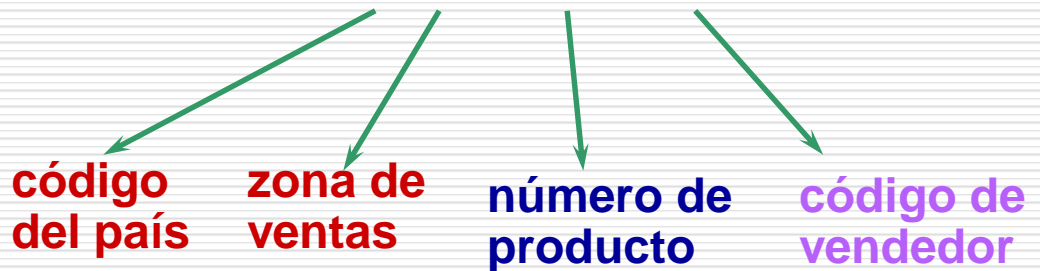
- Limpieza de datos: eliminar datos, corregir y completar datos, eliminar duplicados, ...
- Estandarización: codificación, formatos, unidades de medida, ...

ETL - TRANSFORMACION

Transformación.

- **Campos o atributos compuestos:** descomponer en valores atómicos

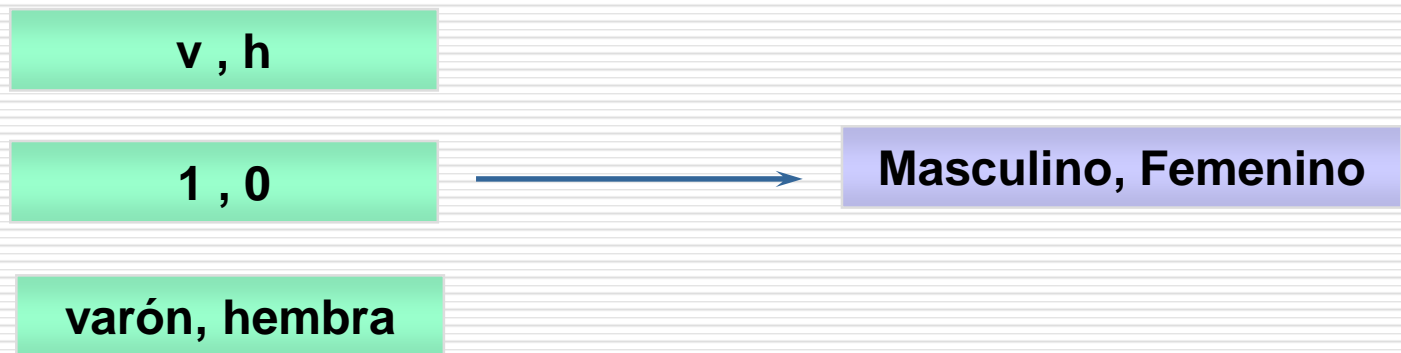
Código de producto = 12M65431345



ETL - TRANSFORMACION

Transformación.

- Unificar codificaciones: existencia de codificaciones múltiples.

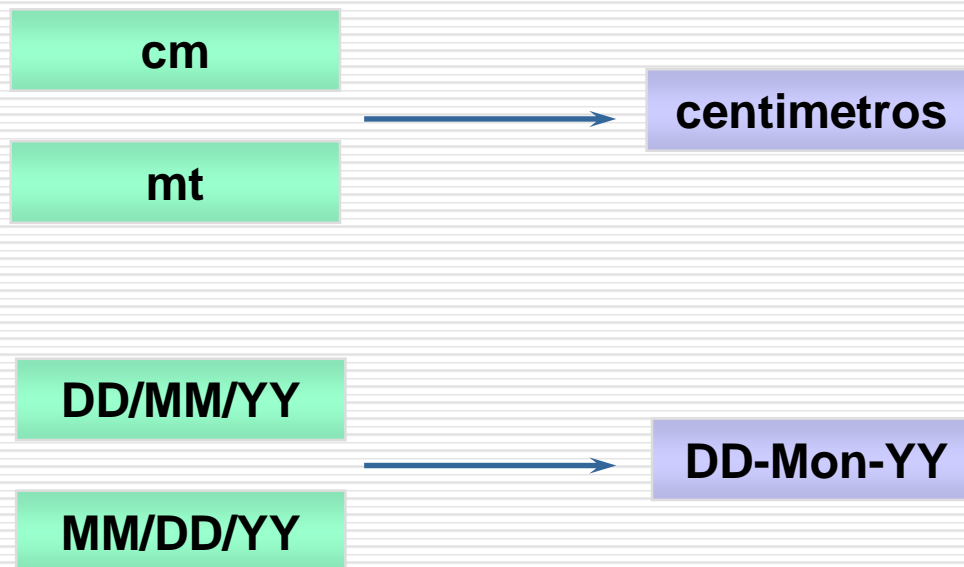


- Deben detectarse los valores erróneos y:
 - Ajustar los sistemas (caso de OLTP)
 - O Proponer soluciones (menor de los males).

ETL - TRANSFORMACION

Transformación.

- Unificar estándares: unidades de medida, unidades de tiempo, moneda,...



ETL - TRANSFORMACION

Transformación. Creación de claves y/o resumen.

#1	Venta	1/2/98	12:00:01 Ham Pizza	\$10.00
#2	Venta	1/2/98	12:00:02 Cheese Pizza	\$15.00
#3	Venta	1/2/98	12:00:02 Anchovy Pizza	\$12.00
#4	Devolución	1/2/98	12:00:03 Anchovy Pizza	- \$12.00
#5	Venta	1/2/98	12:00:04 Sausage Pizza	\$11.00



#dw1	Venta	1/2/98	12:00:01 Ham Pizza	\$10.00
#dw2	Venta	1/2/98	12:00:02 Cheese Pizza	\$15.00
#dw3	Venta	1/2/98	12:00:04 Sausage Pizza	\$11.00

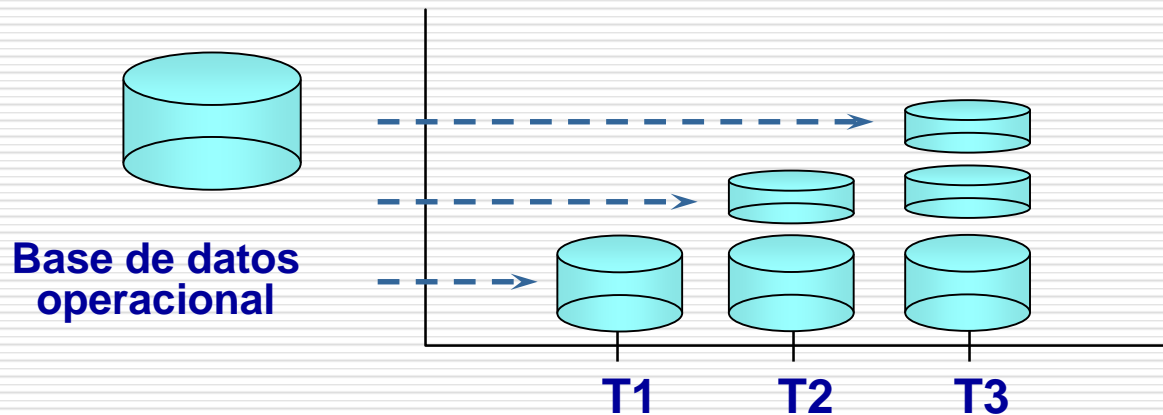
ETL - CARGA

Carga

- La fase de **Carga** consiste en mover los datos desde las fuentes operacionales o el almacenamiento intermedio hasta el almacén de datos del DW y cargar los datos en las tablas respectivas.
- La **carga** puede consumir mucho tiempo.
- En la **carga** inicial del DW se mueven grandes volúmenes de datos.
- En los mantenimientos periódicos del DW se mueven pequeños volúmenes de datos.
- La frecuencia del mantenimiento periódico está determinada por el gránulo del DW y los requisitos de los usuarios.

ETL - CARGA

Carga. Creación y mantenimiento de una BD.

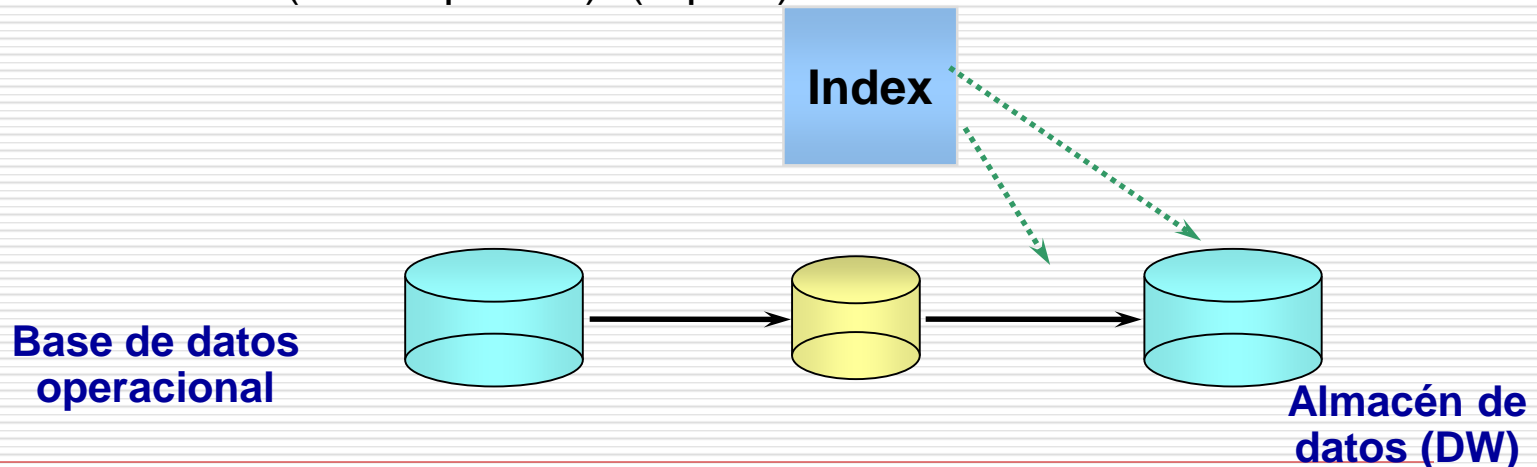


- En intervalos de tiempo fijos añadir cambios a la BD. Se deben determinar las “ventanas de carga” más convenientes para no saturar la base de datos operacional.
- Ocasionalmente archivar o eliminar datos obsoletos que ya no interesan para el análisis.

ETL - CARGA

Procesos anteriores y posteriores a la carga: indexación

- Antes de la carga:
 - Muchas veces es conveniente **deshabilitar índices**
 - **Deshabilitar** el uso del **DW**.
- Después de la carga:
 - **Habilitar índices** deshabilitados
 - **Creación del índice** (total o parcial). (rápido)



Practica Sugerida

La base de datos del sistema de registros de operaciones agropecuarias de una Institución Nacional contiene los siguientes datos que los productores aportan anualmente:

Establecimiento: nombre, localidad, departamento, provincia

Producción de granos:

- Cosecha fina: tipo de grano (trigo, centeno, cebada, ...), Toneladas, Mes de cosecha, Año
- Cosecha gruesa: tipo de grano (soja, maíz, maní, ...), Toneladas, Mes de cosecha, Año
- Stock al 31/12 en silos propios: tipo de grano, Toneladas.
- Ventas durante el año: tipo de grano, Toneladas, Destino (consumo interno o exportación).

Practica Sugerida

Producción de carne:

- Stock al 31/12: Tipo de ganado (ovino, bovino, caprino, ...), Raza, Número de cabezas
- Ventas mercado interno: Tipo de ganado, Raza, Kilogramos, Segmento (conserva, novillos, ternera, ...), Destino (carnicería, frigorífico), Fecha
- Exportación: Tipo de ganado, Raza, Kilogramos, Fecha, país

La institución desea tener un Data Mart del cual pueda dar respuesta a los siguientes requerimientos de información:

- 1 Toneladas de granos producidos por cosecha (Fina o Gruesa), por tipo de grano, por año y por departamento.
- 2 Toneladas de grano vendidas, por tipo de grano, por mes y por destino.

Practica Sugerida

3. Porcentaje de la producción de granos, por tipo de grano, por año y por departamento, no vendidas al 31/12 respecto al total cosechado en el año. (NOTA: tener en cuenta que los granos de una cosecha siempre se venden en su totalidad antes de la nueva cosecha)
4. Toneladas de carne exportada mensualmente, por tipo de ganado y por raza.
5. Toneladas de carne vendida para consumo interno, mensualmente, por tipo de ganado y por segmento.
6. Variación anual del stock de cabezas, por tipo de ganado, por raza, por provincia.
7. Stock promedio del último trienio de cabezas por tipo de ganado, por raza, por departamento.
8. Porcentaje anual de exportación de granos [Tn exportadas/Tn vendidas], por tipo de cosecha.
9. Porcentaje anual de exportación de carne [Tn exportadas/Tn vendidas], por tipo de ganado

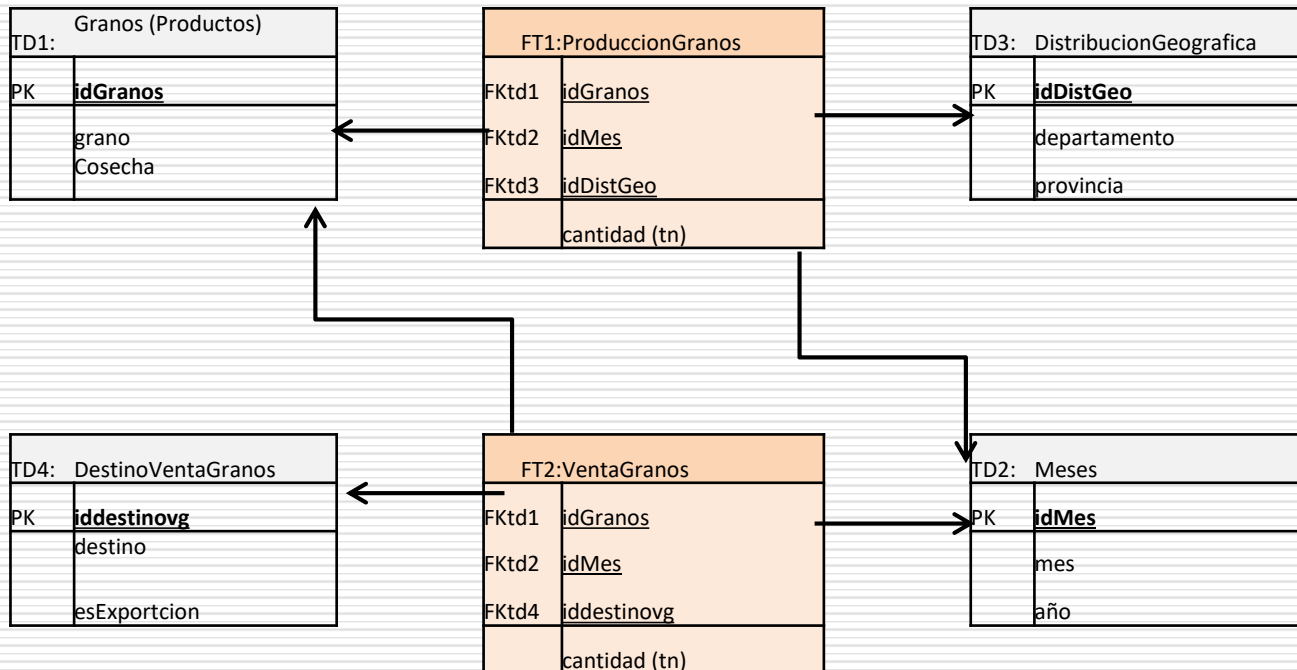
Practica Sugerida

Se pide:

- a) Desarrollar un modelo multidimensional de datos que se ajuste a los requerimientos (no inferir otros posibles requerimientos)
- b) Describir cómo generaría los requerimientos a partir de las Tablas de Hechos, detallando como se harían las agregaciones de los hechos respectivos (suma, promedio, etc) y el nivel de agregación de cada dimensión. Req 1.
- c) Genere una vista que todo el modelo multidimensional lo permita para venta de granos.

Practica Sugerida – Solución a)

Solución para Producción y Venta de Granos



Practica Sugerida – Solución b)

Solución requerimiento b, del requerimiento 1: Toneladas de granos producidos por cosecha (Fina o Gruesa), por tipo de grano, por año y por departamento.

```
SELECT      g.grano,
            g.cosecha,
            m.año,
            d.departamento,
            SUM(cantidad) as cantidadProducida
FROM ProduccionGranos pg
JOIN meses m                ON m.idmes = pg.idmes
JOIN granos g                ON g.idgrano = pg.idgrano
JOIN distribuciongeografica d ON d.iddistgeo = pg.distgeo
GROUP BY g.grano, g.cosecha, m.año, d.departamento
```

Practica Sugerida – Solución c)

Solucion requerimiento c) Genere una vista que todo el modelo multidimensional lo permita para venta de granos.

```
CREATE VIEW VVENTAGRANOS
SELECT      g.grano,                g.cosecha,
            m.año,                  m.mes,
            d.destino,              d.esexportacion,
            SUM(cantidad) as cantidadProducida
FROM ProduccionGranos pg
JOIN meses m                        ON m.idmes = pg.idmes
JOIN granos g                      ON g.idgrano = pg.idgrano
JOIN destinoventagranos d          ON d.iddestinovg = pg.destinovg
GROUP BY g.grano, g.cosecha, m.año, m.mes , d.destino,
        d.esExportacion
```

Fuentes

Business Intelligence Roadmap, L.T.Moss & S. Atre, Addison-Wesley IT series, Boston MA, 2006

The Data Warehouse Lifecycle Toolkit.
R. Kimball, John Wiley & Sons, 1998

Building the Data Warehouse.
W.H. Inmon, John Wiley & Sons, 1996