# Almacenes de datos

## *(Data Warehouses)*

Wladimiro Díaz Villanueva

Wladimiro.Diaz@uv.es

Universitat de València

# Almacenes de datos

1. Introducción.
2. Almacenes de datos: motivación, definición y características.
3. Modelado de datos en almacenes de datos.
4. Construcción de un almacén de datos.
5. Funcionalidad de un almacén de datos.
6. Procesamiento analítico en línea.
7. Problemas de implementación.

# 1. Introducción.

# 1. Introducción

- El cada vez mayor poder de procesamiento y sofisticación de las herramientas y técnicas analíticas ha dado como resultado la creación de los almacenes de datos.
- Proporcionan almacenamiento, funcionalidad y receptividad a las consultas que van más allá de las posibilidades de las bases de datos destinadas a transacciones.
- A este poder en progresivo aumento, se le ha unido una gran demanda para mejorar el rendimiento del acceso a datos que tienen las bases de datos.
  - Las bases de datos tradicionales equilibran el requisito de acceso a datos con la necesidad de asegurar la integridad de los mismos.

# 1. Introducción...

- Los ejecutivos de mandos intermedios y superiores necesitan que se les proporcione información precisa adecuada para su labor en la toma de decisiones.
  - Estos usuarios tan solo necesitan acceso de lectura a los datos.
  - Pero requieren un acceso muy rápido a un gran volumen de datos que pueden descargarse cómodamente en su computador personal.
- Los vendedores de software y el personal de mantenimiento de sistemas han comenzado a diseñar sistemas para realizar estas funciones.
- El mercado de almacenes de datos ha sufrido un rápido crecimiento desde mediados de los años noventa.

# 1. Introducción...

- Dado que se han creado almacenes de datos para satisfacer las necesidades particulares de las empresas, no existe una sola definición canónica del término almacén de datos.
  - Los artículos y libros especializados han ido variando su significado de formas diferentes.
  - Los vendedores han sacado partido de la popularidad del término para impulsar un mercado de diversos productos relacionados.
  - Los consultores han ofrecido una gran variedad de servicios, todos bajo el estandarte de almacenamiento de datos.

Los almacenes de datos difieren de las bases de datos tradicionales en su estructura, funcionamiento, rendimiento y propósito.

# 2. Almacenes de datos: motivación, definición y características.

# 2.1. Motivación

- La mayoría de decisiones de empresas, organizaciones e instituciones se basan en información de experiencias pasadas.
- Generalmente, la información que es necesario investigar sobre un cierto dominio de la organización se encuentra en:
  - Bases de datos, tanto internas como externas.
  - Otras fuentes muy diversas, no necesariamente bases de datos.
- Muchas de estas fuentes son las que se utilizan para el trabajo diario.

# 2.1. Motivación...

- Tradicionalmente el análisis para la toma de decisiones se realizaba sobre estas mismas bases de datos de trabajo o bases de datos transaccionales.
- Esto implica simultanear:
  - El trabajo transaccional diario de los sistemas de información originales (OLTP, *On-Line Transactional Processing*)
  - Con el análisis de los datos en tiempo real sobre la misma base de datos (OLAP, *On-Line Analytical Processing*).

# 2.1. Motivación...

Esto provoca problemas:

- Disturba el trabajo transaccional diario de los sistemas de información originales:
  - Se realizan consultas muy pesadas (*killer queries*).
  - En situaciones de carga alta, la perturbación es tal que el proceso analítico se debe realizar por la noche o en periodos festivos.
- La base de datos está diseñada para el trabajo transaccional y no para el análisis de los datos, por lo que el análisis es lento.

# 2.1. Motivación...

- Los costes de almacenamiento masivo y conectividad se han reducido en los últimos años.
- Una forma eficiente de operar consiste en copiar los datos necesarios para OLAP en un sistema unificado.

Este es el origen de los almacenes de datos (*data warehouses*) y toda la tecnología asociada (*data warehousing*).

- Facilitan el análisis de los datos en tiempo real (OLAP).
- No disturban el OLTP de las bases de datos originales.

Separar los datos a analizar con respecto a sus fuentes transaccionales requiere tener en cuenta cómo organizar los datos y cómo mantenerlos actualizados.

# 2.2. Definiciones

W.H. Inmon definió un almacén de datos como:

> *"un conjunto de datos orientado a temas, integrado, no volátil, variante en el tiempo, como soporte para la toma de decisiones"*

- Los almacenes de datos proporcionan acceso a datos para análisis complejos, revelación de conocimientos y toma de decisiones.
- Dan respuesta a las demandas de alto rendimiento de datos e información de una organización. Soportan varios tipos de aplicaciones, como OLAP, DSS y aplicaciones de minería de datos.

# 2.2. Definiciones...

- **OLAP** (***on-line analitical processing***): análisis de datos complejos del almacén de datos.
- Los **DSS** (***decission support systems***) proporcionan a las personas que han de tomar decisiones importantes dentro de una organización, datos de nivel superior para la toma de decisiones complejas.
- La **minería de datos** se emplea para el descubrimiento de conocimiento: es un proceso de búsqueda, a partir de los datos, de conocimientos nuevos y no anticipados.

# 2.2. Definiciones...

- Las bases de datos tradicionales soportan **OLTP**:
  - Operaciones de inserción, actualización y borrado que implican sólo algunas tuplas por relación.
  - Aunque también soporta requisitos de consultas de información, están optimizadas para procesar consultas que abarcan una pequeña parte de la base datos.

Por lo tanto, no pueden ser optimizadas para OLAP, DSS o minería de datos.

- Los almacenes de datos están diseñados precisamente para realizar eficientemente la extracción, procesamiento y presentación para el análisis y la toma de decisiones.

# 2.3. Características

Para examinar los almacenes de datos y distinguirlos de las bases de datos transaccionales es necesario contar con un modelo de datos que sea apropiado.

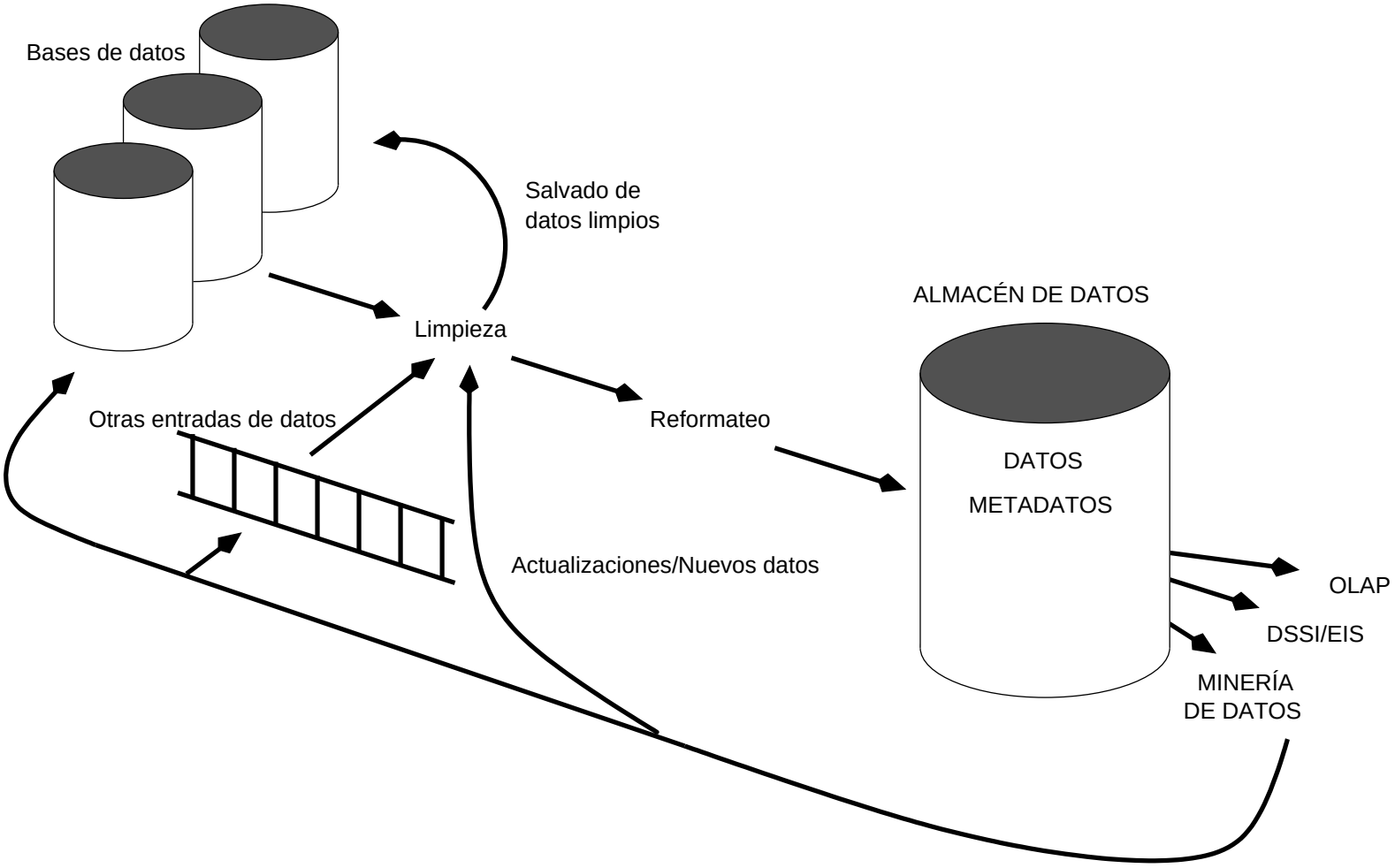
- El modelo de datos multidimensional es una buena opción para las tecnologías OLAP y de soporte a la toma de decisión.
- En un almacén de datos es con frecuencia un almacén de datos integrados provenientes de fuentes diversas, procesados para su almacenamiento en un modelo multidimensional.
- Los almacenes de datos suelen mantener series de tiempo y análisis de tendencia, que necesitan más datos históricos de los que contienen generalmente las bases de datos transaccionales.

# 2.3. Características...

- Los almacenes de datos son no volátiles. Esto significa que la información contenida en el almacén de datos cambia con menos frecuencia y puede considerarse como tiempo no real con actualización periódica.
- La información del almacén de datos es menos precisa (de grano grueso) y se actualiza de acuerdo a una política de actualización, elegida con cuidado, y que es generalmente incremental.
- Las actualizaciones del almacén de datos las realiza el componente de adquisición del almacén, que proporciona todo el procesamiento previo necesario.

# 2.3. Características...

Perspectiva general de la estructura conceptual de un almacén de datos:



# 2.3. Características...

Características distintivas de un almacén de datos:

- Visión conceptual multidimensional.
- Dimensionalidad genérica.
- Dimensiones ilimitadas y niveles de agregación.
- Operaciones de dimensiones cruzadas sin restricciones.
- Tratamiento de matriz *sparse* y dinámica.
- Arquitectura cliente-servidor.
- Soporte multiusuario.
- Accesibilidad.
- Transparencia.

# 2.3. Características...

- Manipulación de datos intuitiva.
- Buen rendimiento al crear informes consistentes.
- Creación de informes flexibles.

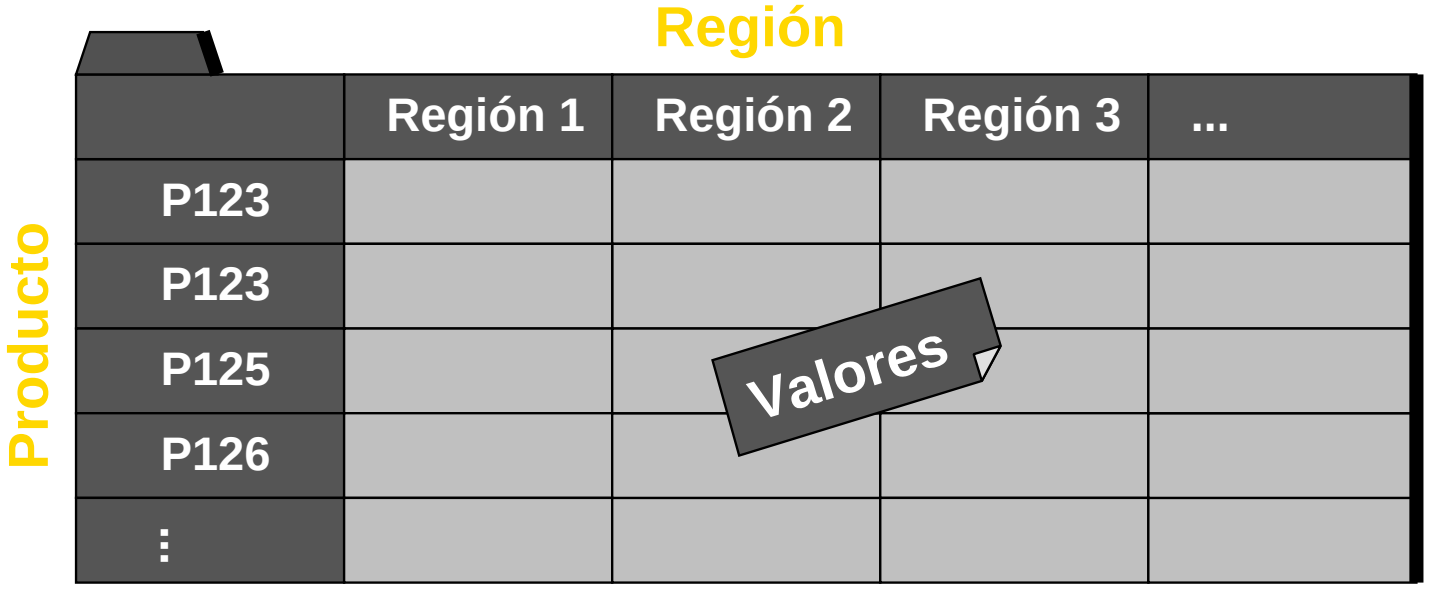
# 2.3. Características...

- Los almacenes de datos tienen un orden de magnitud (a veces dos) superior al de las bases de datos fuente.
- Este inmenso volumen de datos (probablemente de *terabytes*) ha sido tratado mediante:
  - Los **almacenes de datos en grandes empresas** son proyectos de gran tamaño que requieren una enorme inversión de tiempo y recursos.
  - Los **almacenes de datos virtuales** proporcionan vistas de bases de datos operacionales que se materializan para un acceso eficiente.
  - Los ***data marts*** tienen generalmente como objetivo un subconjunto de la organización.

# 3. Modelado de datos en almacenes de datos.

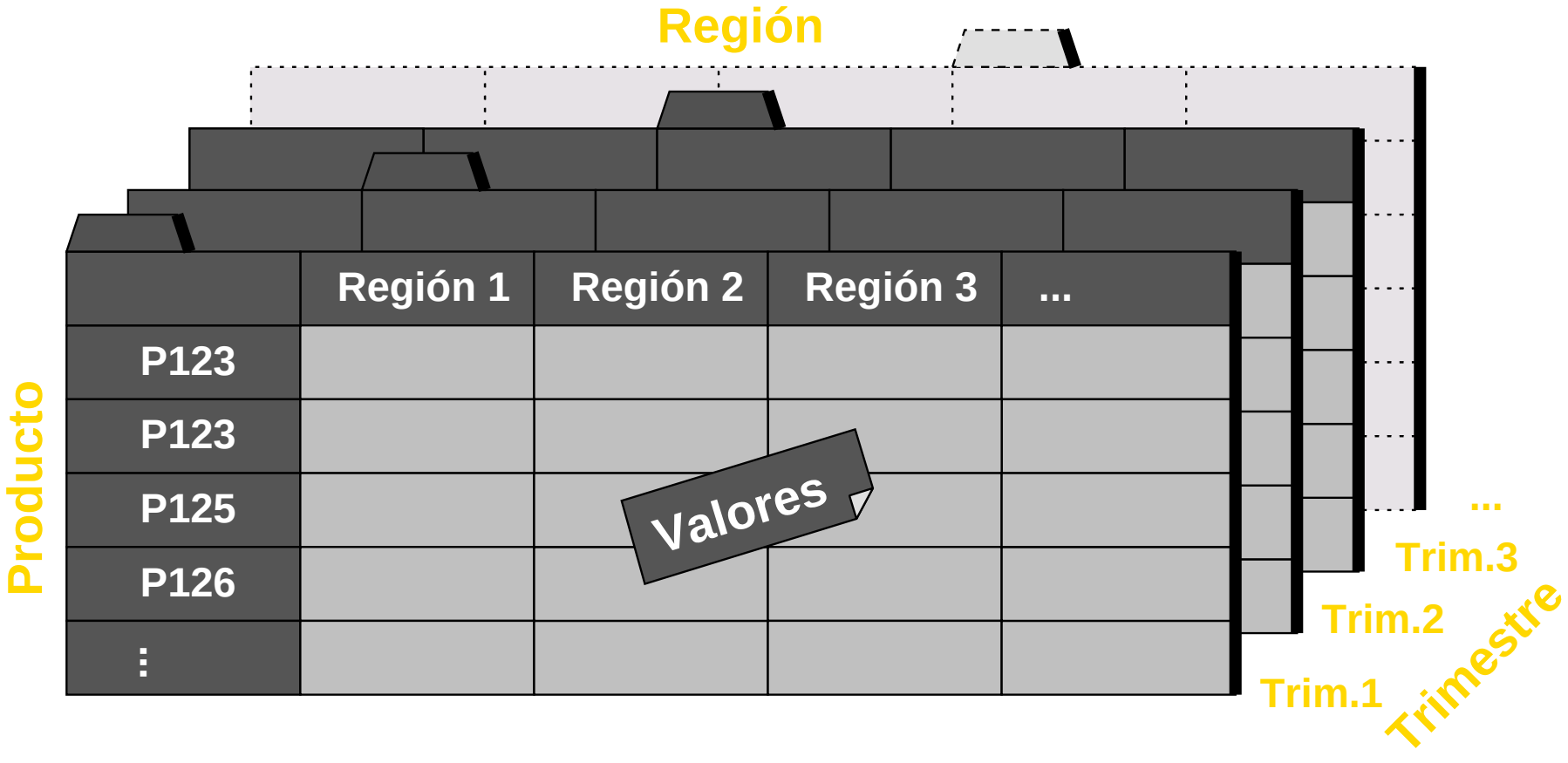
# 3. Modelado de datos

- Una hoja de cálculo estándar constituye una matriz bidimensional.



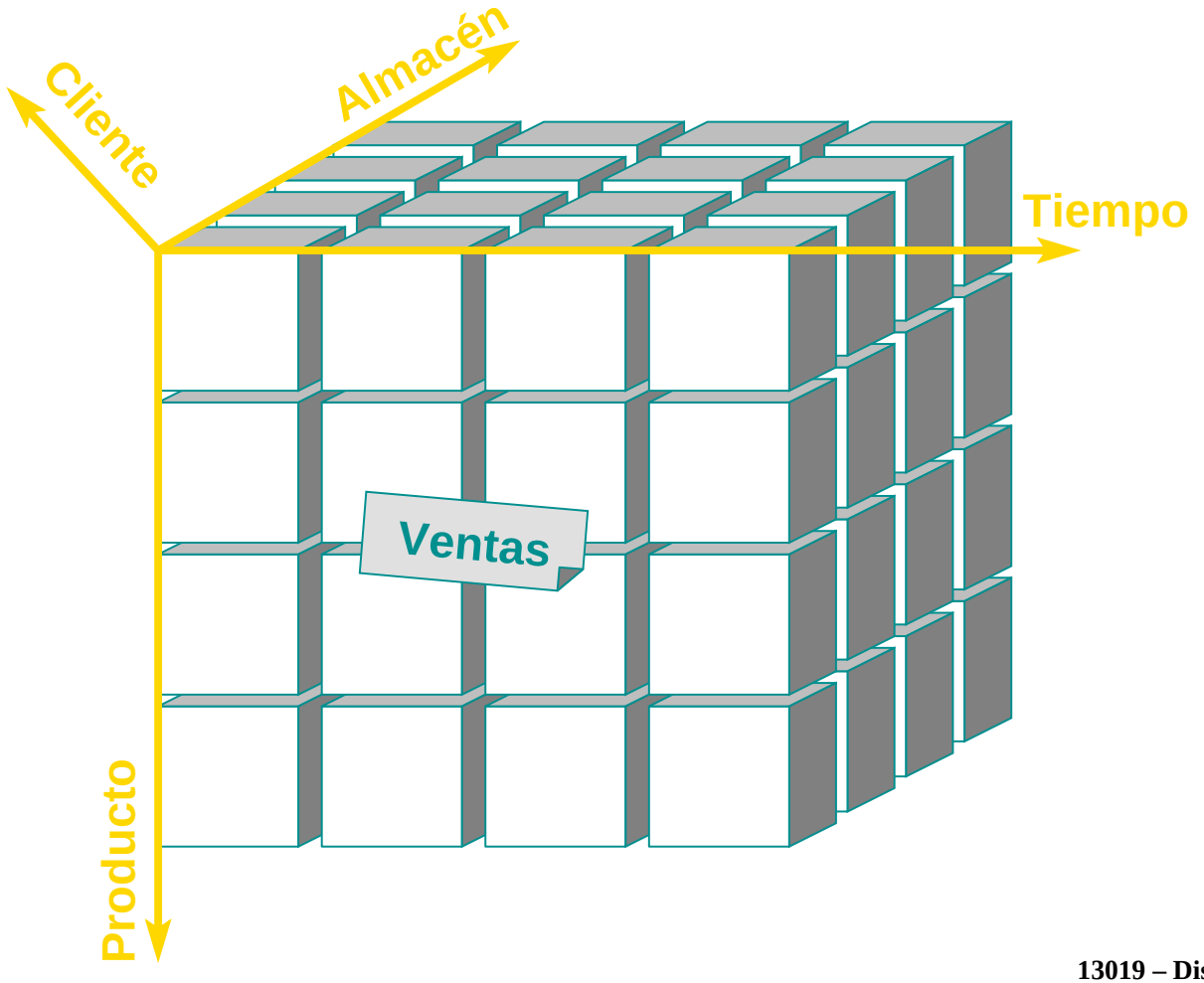
# 3. Modelado de datos...

- Si añadimos una dimensión temporal tendríamos una matriz tridimensional.



# 3. Modelado de datos...

- Las herramientas de explotación OLAP de los almacenes de datos han adoptado un modelo multidimensional de datos.



# 3. Modelado de datos...

- Los modelos multidimensionales se prestan fácilmente a representaciones jerárquicas en lo que se conoce como exploración ascendente (*roll-up*) y exploración descendente (*drill- down*).
  - La exploración ascendente desplaza la jerarquía hacia arriba, agrupándola en unidades mayores a través de una dimensión. Por ejemplo: resumiendo los datos semanales en trimestrales o en anuales.
  - La exploración descendente ofrece la función contraria (de grano más fino). Por ejemplo: disgregando las ventas nacionales en ventas por regiones y después éstas en ventas por subregiones.

# 3. Modelado de datos...

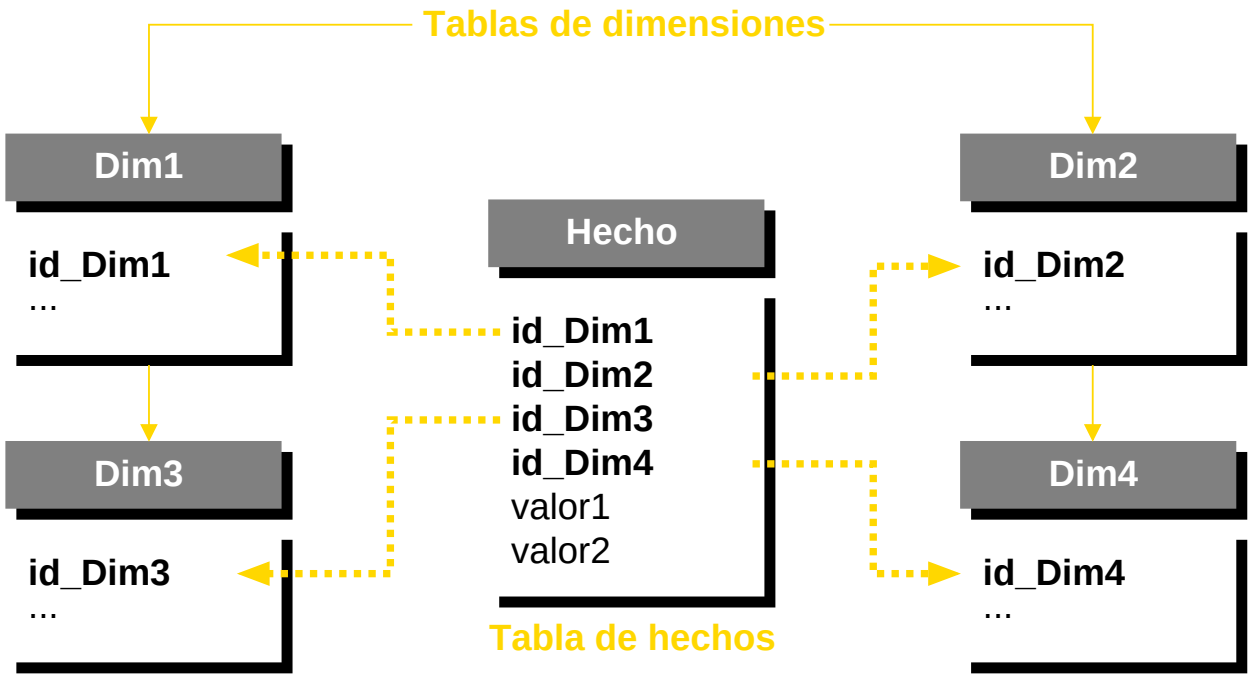
- El diseño multidimensional es un método de diseño de bases de datos basado en el modelo relacional.
- Está compuesto por dos tipos de tablas:
  - Varias tablas de dimensiones, cada una formada por tuplas de atributos de la dimensión.
  - Una tabla de hechos, compuesta por tuplas, una por cada hecho registrado. Este hecho contiene alguna variable o variables medidas u observadas y las identifica con punteros a las tablas de dimensiones.
- Esquema relacional compuesto por 1 tabla de hechos y $M$ tablas de dimensiones: relación $1 : M$.
- La tabla de hechos contiene los datos, las dimensiones identifican cada tupla en esos datos.

# 3. Modelado de datos...

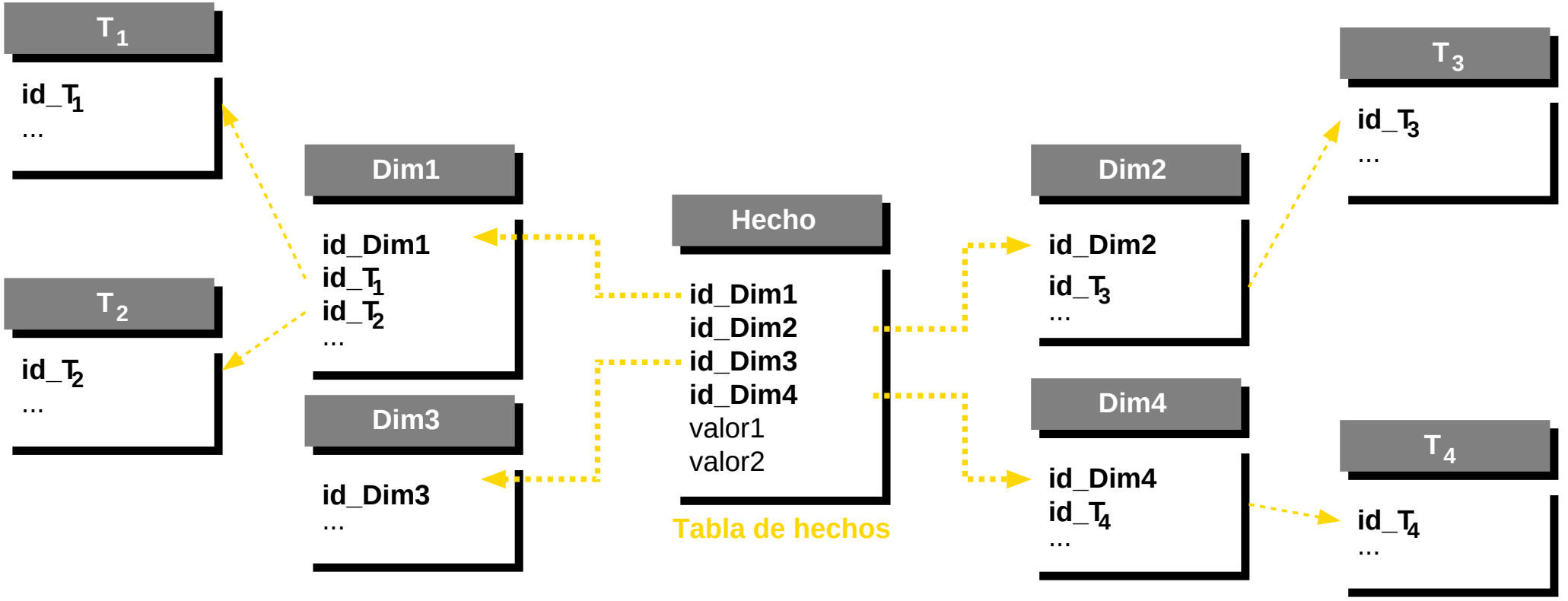
Tres son los esquemas multidimensionales comunes:

- **Esquema en estrella:** formado por una tabla de hechos con una única tabla para cada dimensión.
- **Esquema en copos:** es una variante del esquema de estrella en el que las tablas dimensionales de este último se organizan jerárquicamente mediante su normalización.
- **Constelación de hechos:** es un conjunto de tablas de hechos que comparten algunas tablas de dimensiones.

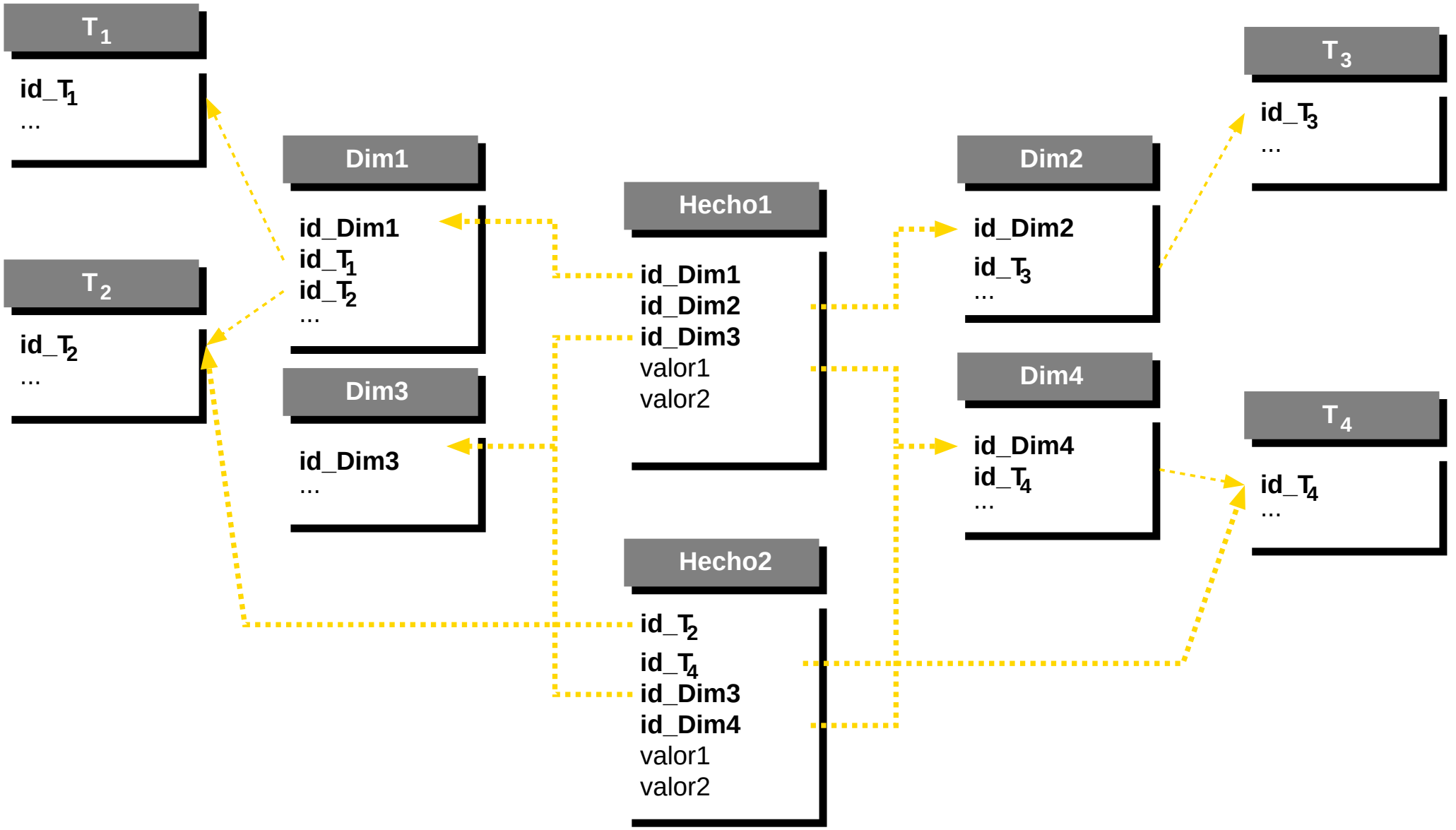
# Esquema en estrella



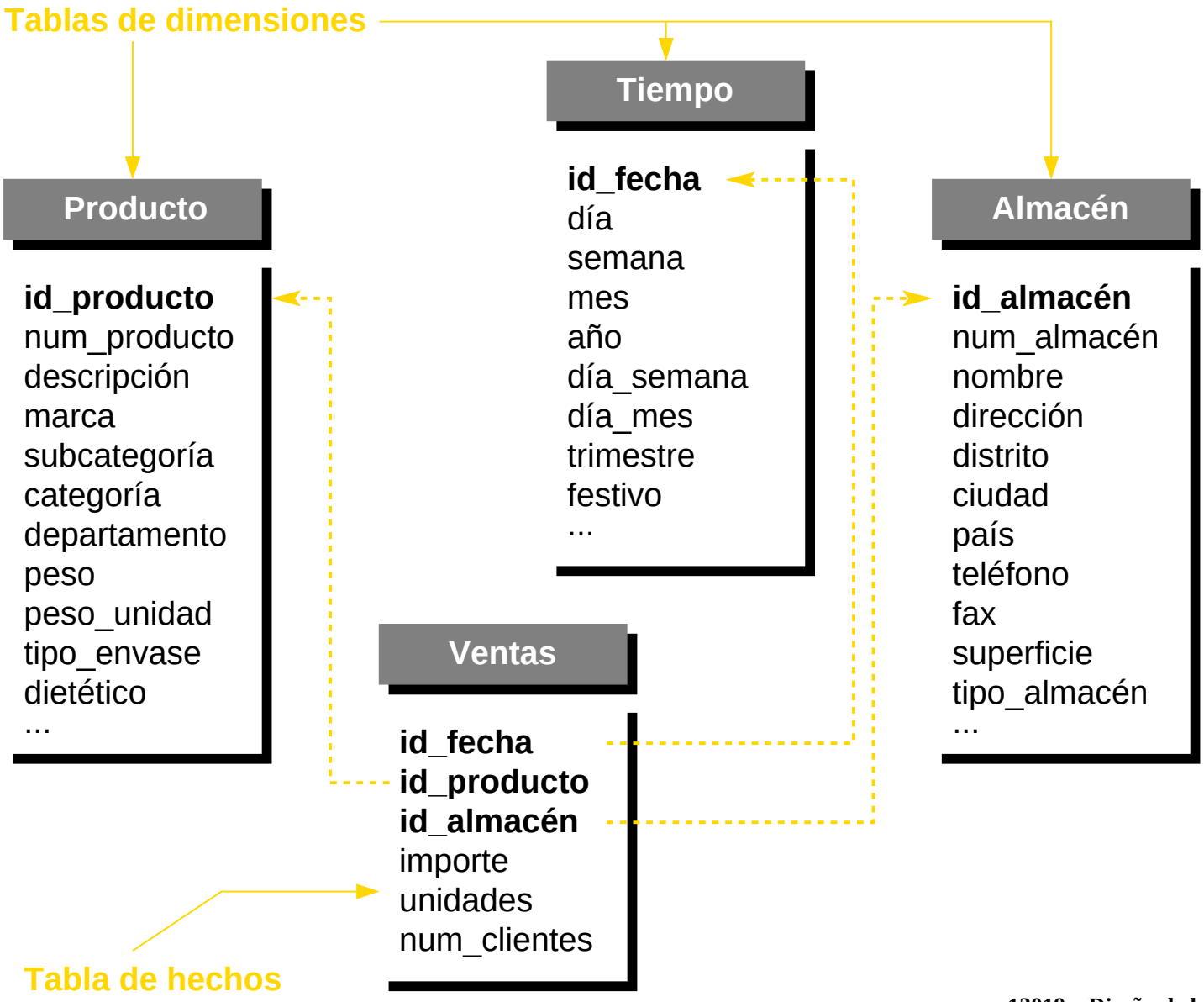
# Esquema en copos



# Constelación de hechos



# 3. Modelado de datos: un ejemplo



# 4. Construcción de un almacén de datos

# 4. Construcción de un almacén de datos

A la hora de construir un almacén de datos:

- Los diseñadores deben tener una amplia perspectiva del uso que se espera del almacén.
  - No existe un modo de anticipar todas las consultas o análisis posibles durante la fase de diseño.
  - Sin embargo, el diseño debería soportar específicamente las consultas *ad hoc*.
  - Ejemplo: una empresa de productos de consumo con un gran soporte de *marketing* necesita organizar el almacén de datos de forma diferente a como lo hace otra basada en la recaudación de fondos con fines no lucrativos.
- Es necesario seleccionar un esquema adecuado que refleje el uso previsto.

# 4.1. Preparación de los datos

- Muchas de las cuestiones que rodean a los sistemas de apoyo para la toma de decisiones, se refieren en primer lugar a las tareas de obtener y preparar los datos.
- Los datos deben ser *extraídos* de diversas fuentes, *limpiados*, *transformados* y *consolidados* en la base de datos de apoyo para la toma de decisiones. Posteriormente, debe ser *actualizados* periódicamente.
- Cada una de estas operaciones involucra sus propias consideraciones especiales.

# Extracción

- La extracción es el proceso de capturar datos de las bases de datos operacionales y otras fuentes.
- Hay muchas herramientas disponibles para ayudar en esta tarea, incluyendo herramientas proporcionadas por el sistema, programas de extracción personalizados y productos de extracción comerciales (de propósito general).
- El proceso de extracción tiende a ser intensivo en E/S y por lo tanto, puede interferir con las operaciones críticas.

# Limpieza

- Pocas fuentes de datos controlan adecuadamente la calidad de los datos.
- Los datos requieren frecuentemente de una **limpieza** antes de que puedan ser introducidos.
- Las operaciones de limpieza típicas incluyen:
  - El llenado de valores ausentes, la corrección de errores tipográficos y otros de captura de datos.
  - El establecimiento de abreviaturas y formatos estándares.
  - El reemplazo de sinónimos por identificadores estándares, etcétera.
- Los datos que son erróneos y que no pueden ser limpiados, serán reemplazados.

# Limpieza...

- La información obtenida durante el proceso de limpieza puede ser usada para identificar la causa de los errores en el origen y por tanto, mejorar la calidad de los datos.

# Transformación y consolidación

- Aun después haber sido limpiados, es probable que los datos todavía no estén en la forma en que se requiere.
- Por lo tanto, deberán ser **transformados** adecuadamente.
- En general, la forma requerida es un conjunto de archivos, uno por cada tabla identificada en el esquema físico.
- La transformación de los datos puede involucrar la división o la combinación de registros fuente
- En ocasiones, los errores de datos que no fueron corregidos durante la limpieza son encontrados durante el proceso de transformación.
- Como antes, cualquier dato incorrecto es rechazado.

# Transformación y consolidación...

- La transformación es particularmente importante cuando necesitan mezclarse varias fuentes de datos
- Este proceso se llama **consolidación**.
- En estos casos, cualquier vínculo implícito entre datos de distintas fuentes necesita volverse explícito (introduciendo valores de datos explícitos).
- Además, las fechas y horas asociadas con el significado que tienen los datos en los negocios, necesitan ser mantenidas y correlacionadas entre fuentes; un proceso llamado *"sincronización en el tiempo"*.
- Las operaciones de transformación pueden ser intensivas tanto en E/S como en CPU.

# Transformación y consolidación...

La sincronización en el tiempo puede ser un problema difícil.
Ejemplo:

- Queremos encontrar el promedio de las ganancias por cliente y por vendedor en cada trimestre.
- Los datos del cliente contra las ganancias son mantenidos por trimestre fiscal en una base de datos de contabilidad.
- Los datos del vendedor contra el cliente son mantenidos por trimestre de calendario en una base de datos de ventas.
- La consolidación de clientes es fácil, involucra simplemente la coincidencia de IDs de clientes.
- Sin embargo, la sincronización de tiempo es mucho más difícil: no podemos decir qué vendedores fueron responsables de cuáles clientes en ese momento.

# Carga

- Los fabricantes de DBMS han puesto considerable importancia en la eficiencia de las operaciones de carga.
- Las “operaciones de carga” incluyen:
  - El movimiento de los datos transformados y consolidados hacia la base de datos de apoyo para la toma de decisiones.
  - La verificación de su consistencia (es decir, verificación de integridad).
  - La construcción de cualquier índice necesario.

# Carga...

- *Movimiento de datos.*
  - Por lo general, los sistemas modernos proporcionan herramientas de carga en paralelo.
  - En ocasiones formatearán previamente los datos para darles el formato físico interno requerido por el DBMS de destino antes de la carga real.
  - Una técnica alternativa consiste en cargar los datos en tablas de trabajo que se asemejan al esquema de destino.
    - La verificación de la integridad necesaria puede ser realizada en esas tablas de trabajo.
    - Posteriormente, se puede usar los INSERTs de conjunto para mover los datos desde las tablas de trabajo hacia las tablas de destino.

# Carga...

- *Verificación de integridad.*
  - La mayor parte de la verificación de integridad de los datos puede ser realizada antes de la carga real, sin hacer referencia a los datos que ya están en la base de datos.
  - Sin embargo, ciertas restricciones no pueden verificarse sin examinar la base de datos existente.
  - Ejemplo: una restricción de unicidad tendrá que ser verificada, por lo general, durante la carga real.

# Carga...

- *Construcción de índices.*
  - La presencia de índices puede hacer significativamente lento el proceso de carga.
    - La mayoría de los DBMS actualizan los índices conforme cada fila es insertada en la tabla subyacente.
  - En ocasiones es buena idea eliminar los índices antes de la carga y luego volverlos a crear. Sin embargo, este enfoque presenta problemas:
    - No vale la pena cuando el volumen de los nuevos datos es pequeño respecto a los ya existentes.
    - La creación de un índice grande puede dar lugar a errores de asignación irrecuperables.
  - La mayoría de los DBMS soportan la creación de índices en paralelo (agilizar los procesos de carga y de construcción de índices).

# Actualización

- La mayoría de las bases de datos de apoyo para la toma de decisiones requieren una **actualización** periódica de los datos.
- La actualización involucra por lo general una carga parcial.
- Algunas aplicaciones de apoyo para la toma de decisiones requieren la eliminación de la base de datos y una recarga completa.
- La actualización involucra todos los problemas que están asociadas con la carga, pero también es probable que deba realizarse mientras los usuarios están accediendo a la base de datos.

# Actualización...

La política de actualización surgirá probablemente como un compromiso que tiene en consideración las respuestas a las siguientes cuestiones:

- ¿Qué grado de actualidad deben tener los datos?
- ¿Puede un almacén de datos quedarse fuera de línea (*off-line*) y durante cuánto tiempo?
- ¿Qué interdependencias tienen los datos?
- ¿Cuál es la disponibilidad de almacenamiento?
- ¿Cuáles son los requisitos de distribución (por ejemplo, para replicación y partición)?
- ¿Cuál es el tiempo de carga (incluyendo la limpieza, formateo, copia, transmisión y costos adicionales, como la reconstrucción de índices)?

# 5. Funcionalidad de un almacén de datos.

# 5. Funcionalidad de los almacenes de datos

- Los almacenes de datos existen para facilitar las consultas complejas, que involucran a gran cantidad de datos y que son con frecuencia *ad hoc*.
- Por lo tanto, deben proporcionar un soporte de consulta mucho mayor y más eficaz que el exigido por las bases de datos transaccionales.
- El componente de acceso de los almacenes de datos soporta una funcionalidad de hoja de cálculo extendida, un procesamiento de consultas eficiente, consultas estructuradas, consultas *ad hoc* y minería de datos.
- La funcionalidad de hoja de cálculo extendida incluye un soporte para lo más novedoso en aplicaciones de hojas de cálculo.

# 5. Funcionalidad...

- También proporciona soporte para programas de aplicaciones OLAP:
  - Exploración ascendente (*roll up*): los datos se resumen con una generalización en aumento.
  - Exploración descendente (*drill down*): se muestran niveles de detalle cada vez mayores.
  - Pivotación (rotación): se realiza una tabulación cruzada.
  - Rodaja y cubo: ejecución de operaciones de proyección en las dimensiones.
  - Clasificación: los datos se ordenan por valor ordinal.
  - Atributos derivados (calculados): los atributos se calculan mediante operaciones con valores almacenados y derivados.

# 6. Procesamiento analítico en línea.

# 6. Procesamiento analítico en línea

- El término OLAP puede ser definido como:

  *"el proceso interactivo de crear, mantener, analizar y elaborar informes sobre datos"*

- Los datos en cuestión son percibidos y manejados como si estuvieran almacenados en una *"matriz multidimensional"*
- El procesamiento analítico requiere invariablemente algún tipo de agregación de datos (por lo general en muchas formas diferentes).
- Uno de los problemas fundamentales del procesamiento analítico es que la cantidad de agrupamientos posibles llega rápidamente a ser muy grande y los usuarios deben considerarlos todos o casi todos.

# 6. Procesamiento analítico en línea...

- Los lenguajes relacionales soportan la agregación requerida.
- Sin embargo, cada consulta individual en SQL produce como resultado una sola tabla (y todas las filas de esa tabla tienen la misma forma y el mismo tipo de interpretación).
- Por lo tanto, para obtener $n$ agrupamientos distintos se requieren $n$ consultas distintas y $n$ tablas de resultados distintas.

# 6. Procesamiento analítico en línea...

Ejemplo:

- Supongamos una tabla que mantiene información sobre los envíos de partes realizados por un conjunto de proveedores:

**VP**

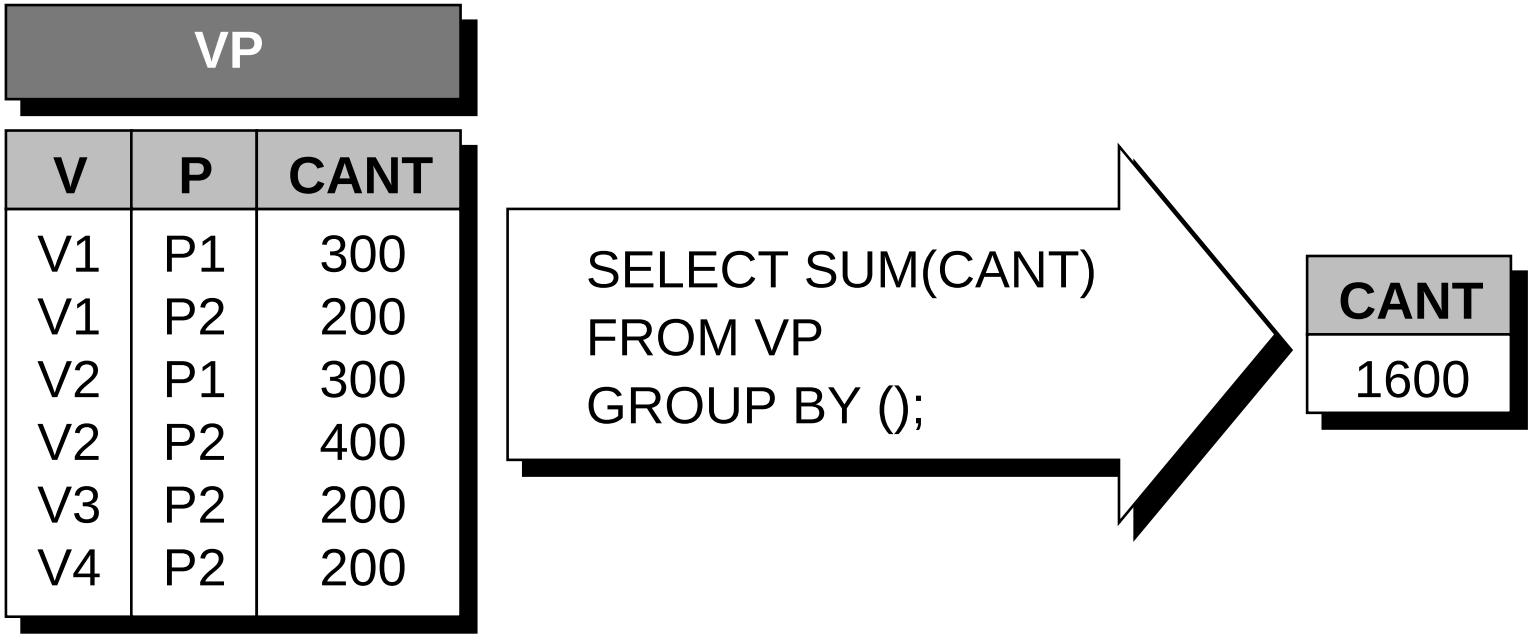
| V | P | CANT |
|---|---|---|
| V1 | P1 | 300 |
| V1 | P2 | 200 |
| V2 | P1 | 300 |
| V2 | P2 | 400 |
| V3 | P2 | 200 |
| V4 | P2 | 200 |

# 6. Procesamiento analítico en línea...

- Deseamos realizar las siguientes consultas sobre la tabla:
  1. Obtener la cantidad total de envíos.
  2. Obtener las cantidades totales de envíos por proveedor.
  3. Obtener las cantidades totales de envíos por partes.
  4. Obtener las cantidades totales de envíos por proveedor y parte.

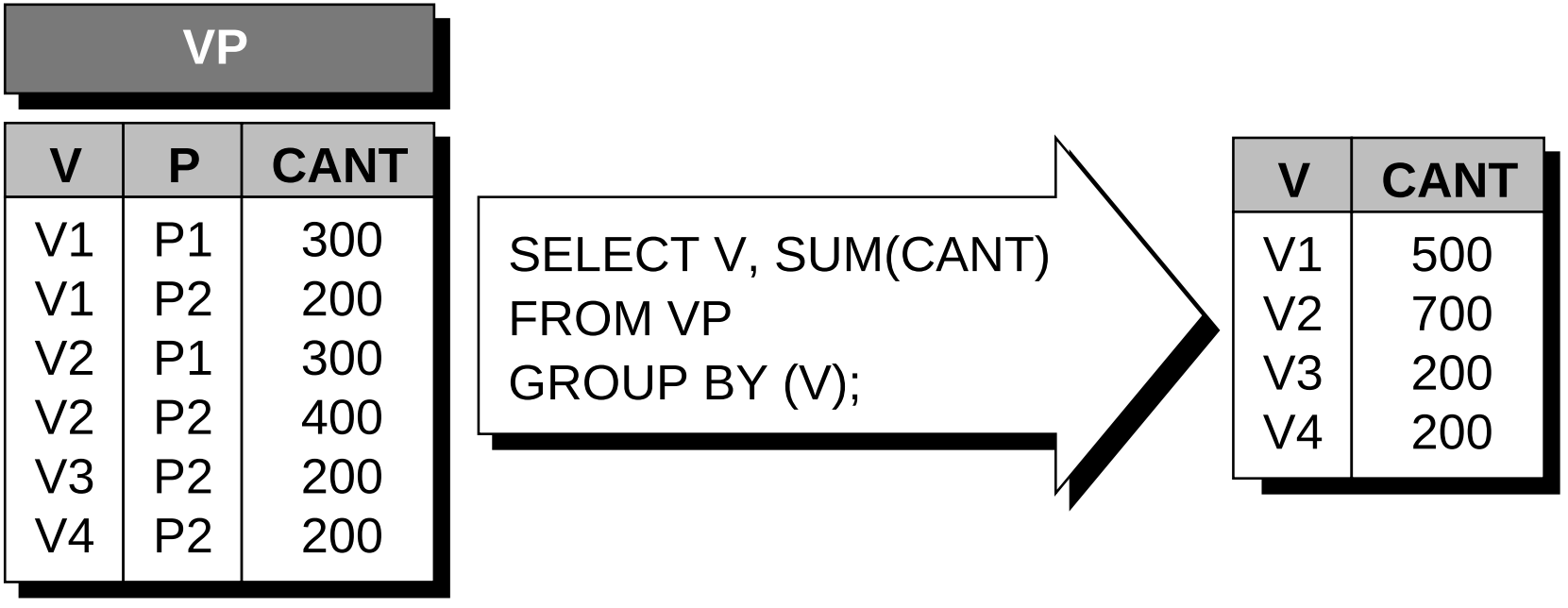
# 6. Procesamiento analítico en línea...

1. Obtener la cantidad total de envíos.



# 6. Procesamiento analítico en línea...

2. Obtener las cantidades totales de envíos por proveedor.



# 6. Procesamiento analítico en línea...

3. Obtener las cantidades totales de envíos por partes.

**VP**

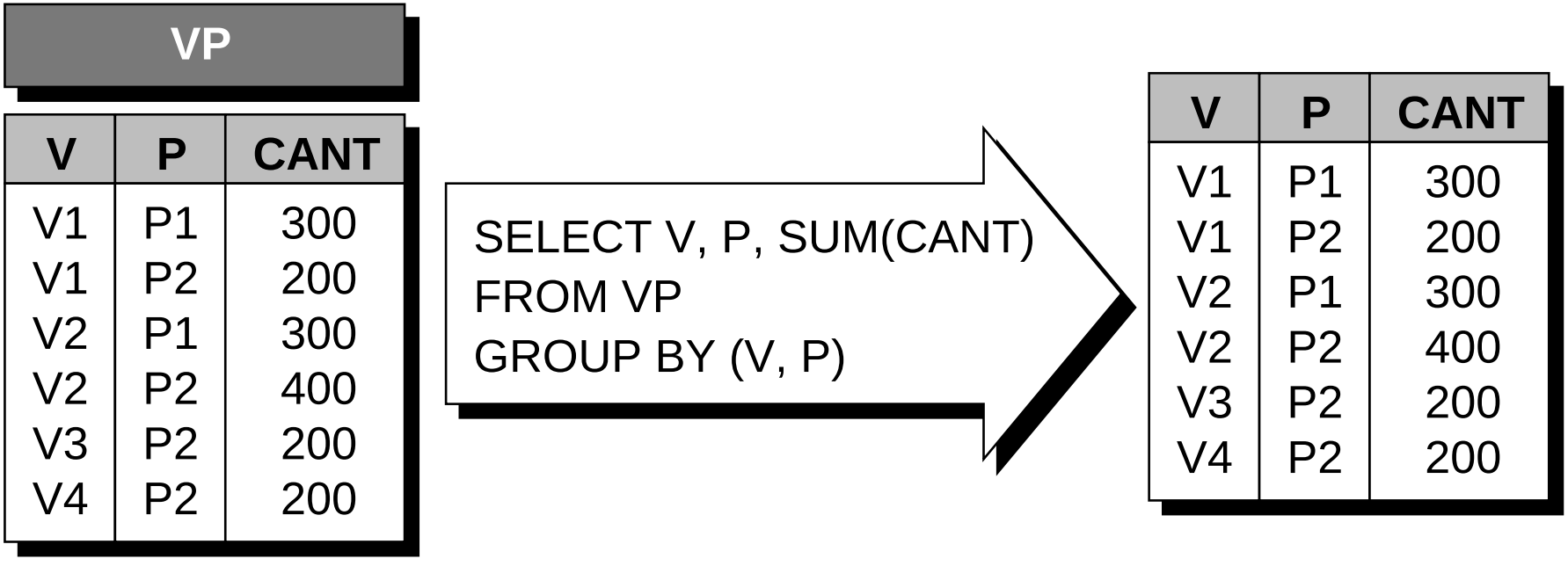
| V | P | CANT |
|---|---|---|
| V1 | P1 | 300 |
| V1 | P2 | 200 |
| V2 | P1 | 300 |
| V2 | P2 | 400 |
| V3 | P2 | 200 |
| V4 | P2 | 200 |

```
SELECT P, SUM(CANT)
FROM VP
GROUP BY (P)
```

| P | CANT |
|---|---|
| P1 | 600 |
| P2 | 1000 |

# 6. Procesamiento analítico en línea...

4. Obtener las cantidades totales de envíos por proveedor y parte.

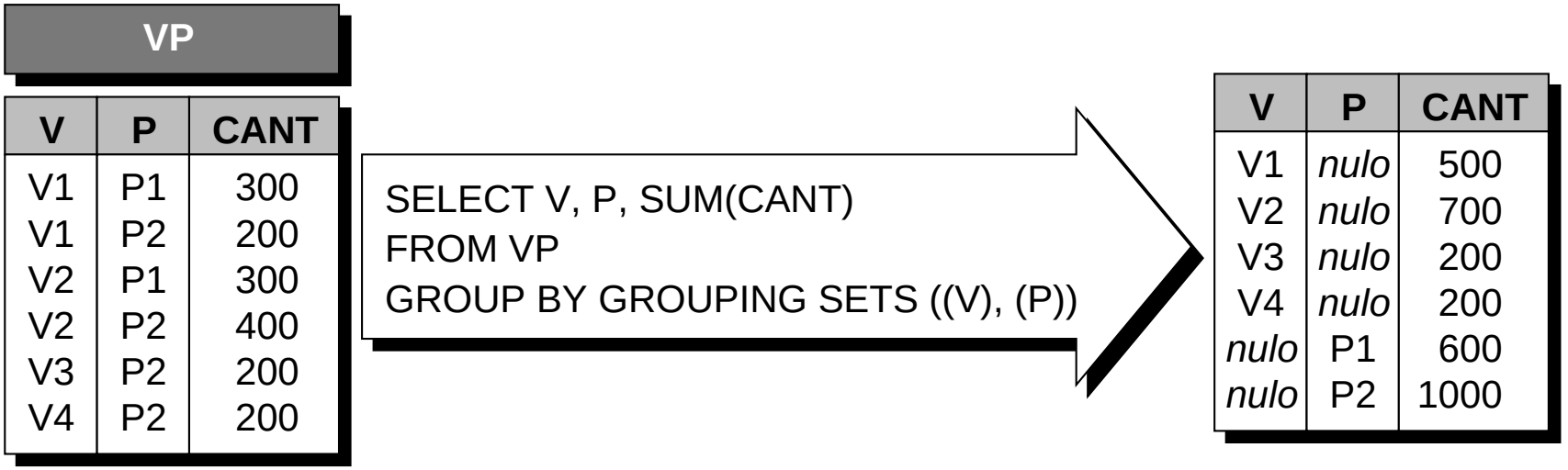


# 6. Procesamiento analítico en línea...

- Las desventajas de este enfoque son obvias:
  - La formulación de estas consultas (similares pero distintas) es tediosa para el usuario.
  - La ejecución de todas esas consultas (pasan y otra vez por los mismos datos) es probablemente bastante costosa en tiempo de ejecución.
- Vale la pena tratar de encontrar una forma de:
  - Solicitar varios niveles de agregación en una sola consulta.
  - Ofrecer a la implementación la oportunidad calcular todas esas agregaciones de manera más eficiente (es decir, en un solo paso).
- Estas consideraciones son la motivación que hay tras las opciones GROUPING SETS, ROLLUP y CUBE de la cláusula GROUP BY.

# 6. Procesamiento analítico en línea...

- La opción GROUPING SETS permite al usuario especificar con exactitud qué agrupamientos específicos van a realizarse.
- Ejemplo: la siguiente instrucción SQL es una combinación de las consultas 2 y 3:




- Aquí la cláusula GROUP BY solicita al sistema que ejecute dos consultas, una en donde el agrupamiento es por V y otra en la que es por P.

# 6. Procesamiento analítico en línea...

- ROLLUP y CUBE son abreviaturas para ciertas combinaciones de GROUPING SETS.

**VP**

| V | P | CANT |
|---|---|---|
| V1 | P1 | 300 |
| V1 | P2 | 200 |
| V2 | P1 | 300 |
| V2 | P2 | 400 |
| V3 | P2 | 200 |
| V4 | P2 | 200 |

```
SELECT V, P, SUM(CANT)
FROM VP
GROUP BY ROLLUP (V,P);
```

| V | P | CANT |
|---|---|---|
| V1 | P1 | 300 |
| V1 | P2 | 200 |
| V2 | P1 | 300 |
| V2 | P2 | 400 |
| V3 | P2 | 200 |
| V4 | P2 | 200 |
| V1 | *nulo* | 500 |
| V2 | *nulo* | 700 |
| V3 | *nulo* | 200 |
| V4 | *nulo* | 200 |
| *nulo* | *nulo* | 1600 |

- Esta cláusula es lógicamente equivalente a:

```
GROUP BY GROUPING SETS ( (V,P), (V), () )
```

- Combinación de las consultas 4, 2 y 1.

# 6. Procesamiento analítico en línea...

- El término ROLLUP se deriva del hecho de que las cantidades han sido *"enrolladas"* en toda la dimensión del proveedor.
- En general, GROUP BY ROLLUP $(A, B, \ldots, Z)$ significa *"agrupar por todas las combinaciones siguientes"*:

  $(A, B, \ldots, Z)$
  $(A, B, ...)$
  $(A, B)$
  $(A)$
  $()$

- Observe que hay muchos *"enrollar en toda la dimensión A"* distintos, dependiendo de qué otras columnas son mencionadas en la lista de ROLLUP.
- Observe también que GROUP BY ROLLUP (A, B) y GROUP BY ROLLUP (B, A) tienen significados diferentes.

# 6. Procesamiento analítico en línea...

- Considere la siguiente consulta CUBE:

**VP**

| V | P | CANT |
|---|---|---|
| V1 | P1 | 300 |
| V1 | P2 | 200 |
| V2 | P1 | 300 |
| V2 | P2 | 400 |
| V3 | P2 | 200 |
| V4 | P2 | 200 |

```
SELECT V, P, SUM(CANT)
FROM VP
GROUP BY CUBE (V,P);
```

| V | P | CANT |
|---|---|---|
| V1 | P1 | 300 |
| V1 | P2 | 200 |
| V2 | P1 | 300 |
| V2 | P2 | 400 |
| V3 | P2 | 200 |
| V4 | P2 | 200 |
| V1 | *nulo* | 500 |
| V2 | *nulo* | 700 |
| V3 | *nulo* | 200 |
| V4 | *nulo* | 200 |
| *nulo* | P1 | 600 |
| *nulo* | P2 | 1000 |
| *nulo* | *nulo* | 1600 |

- La cláusula GROUP BY es lógicamente equivalente a:

  GROUP BY GROUPING SETS ( (V,P), (V), (P), () )

- La consulta es una formulación SQL combinada de las cuatro consultas originales 4, 3, 2 y 1.

# 6. Procesamiento analítico en línea...

- El término CUBE deriva del hecho de que en la terminología OLAP (multidimensional), los valores de datos pueden ser percibidos como si estuvieran almacenados en las celdas de una matriz multidimensional o hipercubo.
- En el caso que estamos viendo:
  - Los valores de datos son cantidades.
  - El *"cubo"* es de dos dimensiones: una dimensión de proveedores y una dimensión de partes (se trata un *"cubo"* bastante plano).
  - Las dimensiones son de tamaños desiguales (por lo que en realidad no es un *"cubo"*, sino un *"paralepípedo"*).
- GROUP BY CUBE $(A, B, \ldots, Z)$ significa *"agrupar por todos los subconjuntos posibles del conjunto* $(A, B, \ldots, Z)$*"*.

# 6.1. Tabulaciones cruzadas

- Con frecuencia, los productos OLAP muestran los resultados no como tablas al *"estilo SQL"*, sino como **tabulaciones cruzadas** (*"tabcruz"*).
- Ejemplo: Consulta 4 (*"total de envíos por proveedor y parte"*):

| | P1 | P2 |
|---|---|---|
| V1 | 300 | 200 |
| V2 | 300 | 400 |
| V3 | 0 | 200 |
| V4 | 0 | 200 |

- Las cantidades de la parte P1 para los proveedores V3 y V4 se muestran correctamente como cero.
- La tabla que produce SQL no contiene filas para (V3, P1) ni para (V4, P1): la construcción de la *tabcruz* a partir de la tabla no es trivial.

# 6.1. Tabulaciones cruzadas...

- Una *tabcruz* es una forma más compacta y legible de representar el resultado .
- Sin embargo, la cantidad de columnas en esta *“tabla”* depende de los datos reales: hay una columna para cada tipo de parte.
- Una tabcruz no es una relación, sino un informe.
- Esta *tabcruz* tienen dos dimensiones: en este caso proveedores y partes.
  - Las dimensiones son tratadas como si fueran variables *independientes*.
  - Las *”celdas”* de intersección contienen los valores de las variables *dependientes* correspondientes.

# 6.1. Tabulaciones cruzadas...

- Otro ejemplo de tabcruz que representa el resultado del ejemplo anterior de CUBE:

| | P1 | P2 | *total* |
|---|---|---|---|
| **V1** | 300 | 200 | 500 |
| **V2** | 300 | 400 | 700 |
| **V3** | 0 | 200 | 200 |
| **V4** | 0 | 200 | 200 |
| ***total*** | 600 | 1000 | 1600 |

  - La columna del extremo derecho contiene totales de fila.
  - La fila inferior contiene totales de columna.
  - La celda de la parte inferior derecha contiene el *gran total*, que es el total de fila de todos los totales de columna y el total de columna de todos los totales de fila.

# 7. Problemas de implementación.

# 7. Problemas de implementación

Cuestiones operacionales significativas en los almacenes de datos: la construcción, la administración y el control de calidad.

- **La gestión del proyecto** (diseño, construcción e implementación del almacén de datos) supone un reto.
  - La construcción de un almacén de datos empresarial en una gran organización es una tarea de gran importancia.
  - Potencialmente supone una labor de años desde su concepción hasta su implementación.
  - El desarrollo y utilización extendida de data marts pueden proporcionar una alternativa interesante, especialmente para aquellas organizaciones que tienen necesidades urgentes de soporte OLAP, DSS o de minería de datos.

# 7. Problemas de implementación...

- **La administración** de un almacén de datos es una labor intensa, proporcional al tamaño y complejidad del almacén.
  - Una organización que intente administrar un almacén de datos debe comprender de forma realista la naturaleza compleja de su administración.
  - Aunque esté diseñado para operaciones de lectura, su estructura no es más estática de lo que puedan serlo sus fuentes de información.
  - Se puede esperar que las bases de datos fuente evolucionen. También debe esperarse que el esquema del almacén de datos y el componente de adquisición sean actualizados para manejar dicha evolución.

# 7. Problemas de implementación...

- Un aspecto significativo del almacenamiento de datos es el **control de calidad de los datos**.
  - Aunque los datos atraviesen un proceso de limpieza durante su obtención, la calidad y la coherencia siguen siendo aspectos significativos para el administrador de la base de datos.
  - La combinación de datos procedentes de fuentes heterogéneas y dispares es uno de los retos principales si consideramos las diferencias de denominación, definiciones de dominios y números de identificación entre otros.

# 7. Problemas de implementación...

Más aún...

- Cada vez que una base de datos fuente cambia, el administrador del almacén de datos debe considerar las posibles interacciones con otros elementos del almacén.
- El almacén debe diseñarse para tener en cuenta la incorporación de fuentes de datos y su caducidad, sin necesidad de un rediseño importante.
- Las fuentes y sus datos evolucionarán y el almacén debe contemplar dichos cambios. El ajuste de los datos fuente disponibles al modelo de datos del almacén constituirá un reto continuo.
- Debido a la rápida y constante evolución de las tecnologías, tanto las necesidades como las posibilidades del almacén sufrirán una transformación considerable con el tiempo.