Curso de Doctorado
"Extracción de Conocimiento en Bases de Datos"

Introducción al Data Warehousing

Fernando Berzal Galiano

Diseño, Análisis y Aplicaciones de Sistemas Inteligentes

Departamento de Ciencias de la Computación e Inteligencia Artificial

Universidad de Granada

Introducción al *Data Warehousing*

Un "*data warehouse*" (DW de aquí en adelante) es un almacén de información normalmente proveniente de distintas bases de datos cuyo objetivo es ayudar en la toma de decisiones.

La idea de DW surge como solución al problema del acceso a un sistema heterogéneo distribuido por mediación (efectuando una consulta compleja que se descompone y envía a las distintas fuentes de información para después combinar los datos obtenidos resultantes de efectuar la consulta sobre las distintas fuentes de información). En el DW, la información almacenada se extrae previamente de las distintas fuentes de datos.

Obviamente, la obtención en información por demanda o mediación es más ineficiente que la realización de consultas sobre un DW, aunque puede ser útil cuando la información cambia rápidamente. Por su parte, el DW es más apropiado para realizar consultas sobre datos históricos y proporciona una visión global que facilita la toma de decisiones que ayuden a atender mejor a los clientes, reducir costes, mejorar ventas, detectar fraudes, incrementar la productividad...

OLAP vs. OLTP

Las aplicaciones informáticas de gestión suelen realizar tareas repetitivas muy bien estructuradas e implican transacciones cortas, actualizaciones generalmente [*OLTP: On-Line Transaction Processing*]. Sin embargo, los sistemas de ayuda a la decisión [*DSSs: Decision Support Systems*] requieren la realización de consultas complejas que involucran muchos datos e incluyen funciones de agregación. De hecho, las actualizaciones son operaciones poco frecuentes en este tipo de aplicaciones, denominado genéricamente "procesamiento analítico" [*OLAP: On-Line Analytical Processing*].

Los requerimientos característicos de las aplicaciones OLAP son, por tanto, muy diferentes a los de los sistemas OLTP.

Las transacciones OLTP se realizan sobre grandes bases de datos a las cuales se puede acceder eficientemente empleando índices (sobre las claves primarias usualmente) y es esencial garantizar su "acidez" (atomicidad, consistencia, aislamiento y durabilidad).

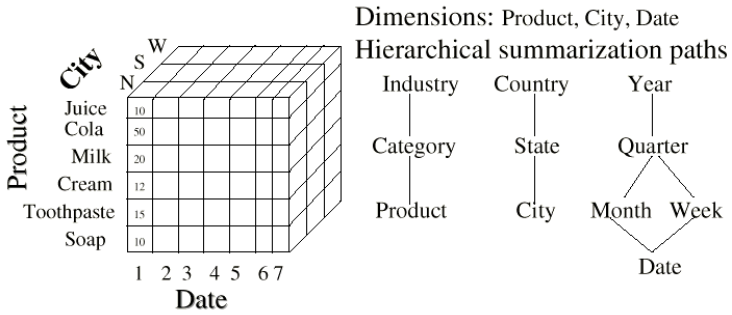
Por su parte, los DW, orientados al soporte de decisiones, almacenan datos resumidos de tipo histórico y han de responder en tiempo real a consultas complejas consultas que involucran reuniones y agregaciones. En este caso, la optimización de las consultas y el tiempo de respuesta son primordiales.

Aspecto	*OLTP* *Base de datos tradicional*	*OLAP* *Data Warehouse*
Usuarios	Operadores	Ejecutivos
Función	Operaciones diarias Procesamiento de transacciones	Soporte de decisiones Procesamiento analítico
Diseño	Orientado a las aplicaciones	Orientado al usuario
Datos	Actuales, atómicos (relacionales)	Históricos, resumidos (multidimensionales)
Uso	Rutinario	"ad hoc"
Acceso	Lectura/escritura Transacciones simples	Lectura Consultas complejas
Necesidades	"Acidez" de las transacciones Datos consistentes	Optimización de consultas Datos organizados

Dado que las consultas OLAP son muy ineficientes en las bases de datos operacionales, la información de un DW se suele almacenar por separado. Los DW se pueden implementar sobre bases de datos relacionales [*ROLAP: Relational OLAP*] o utilizar servidores que almacenan los datos directamente en una estructura multidimensional [*MOLAP: Multidimensional OLAP*], generalmente utilizando matrices.

Modelo conceptual: "visión multidimensional de los datos"

Un modelo de datos multidimensional contiene un conjunto de medidas numéricas objeto de análisis. Dichas medidas dependen de una serie de dimensiones. Cada medida particular es un punto en un espacio multidimensional, en el que los valores de cada dimensión se suelen jerarquizar.



Modelo multidimensional de los datos
[Chaudhuri & Dayal, 1997]

Los datos en un DW se modelan en *data cubes* (cubos de datos sería su traducción literal), estructuras multidimensionales (hipercubos, en concreto) cuyas operaciones más comunes se enumeran a continuación:

☞ *roll up* (incremento en el nivel de agregación)
☞ *drill down* (incremento en el nivel de detalle, opuesto a roll up)
☞ *slice & dice* (reducción de la dimensionalidad de los datos mediante selección y proyección)
☞ *pivotaje* (reorientación de la visión multidimensional de los datos)

Diseño

El diseño de un DW para una organización completa es un proceso bastante complejo y se puede dividir en "*data marts*" departamentales orientados a la resolución de problemas más concretos.

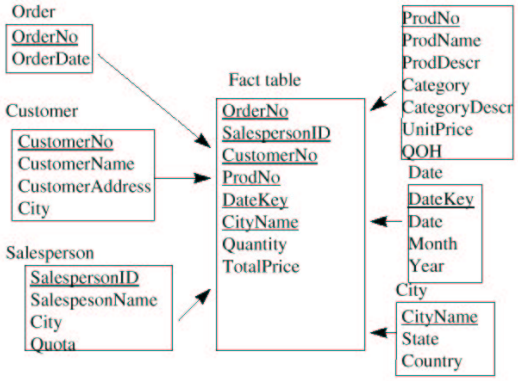
El modelo de datos multidimensional se implementa directamente con servidores MOLAP. Cuando se usan servidores relacionales [ROLAP], dicho modelo ha de transformarse en relaciones y consultas SQL:

- **ESQUEMA EN ESTRELLA**: La base de datos relacional consiste en una tabla simple de hechos y una tabla para cada dimensión. Cada tupla de la tabla de hechos incluye las medidas consideradas y una referencia a cada dimensión.

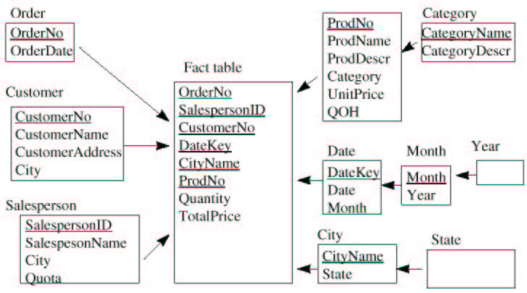
- **ESQUEMA EN BOLA DE NIEVE**: Refinamiento del esquema en estrella que soporta jerarquías manteniendo normalizadas las tablas.

Los esquemas anteriores pueden generalizarse con la inclusión de distintas tablas de hechos que compartan dimensiones (son las denominadas constelaciones de hechos [*fact constellations*]).

Además de las tablas de hechos y dimensiones, los DW pueden almacenar físicamente resúmenes con los datos agregados (en tablas adicionales a modo de constelaciones o en la propia tabla de hechos).

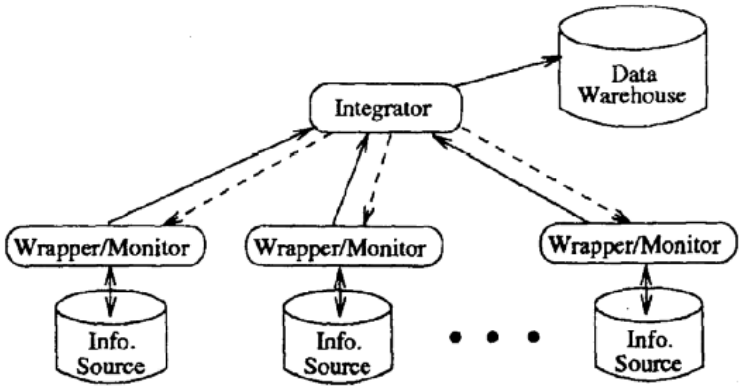


Esquema en estrella



Esquema en bola de nieve

Arquitectura general de un DW



Arquitectura general de un DW [Jennifer Widom, 1996]

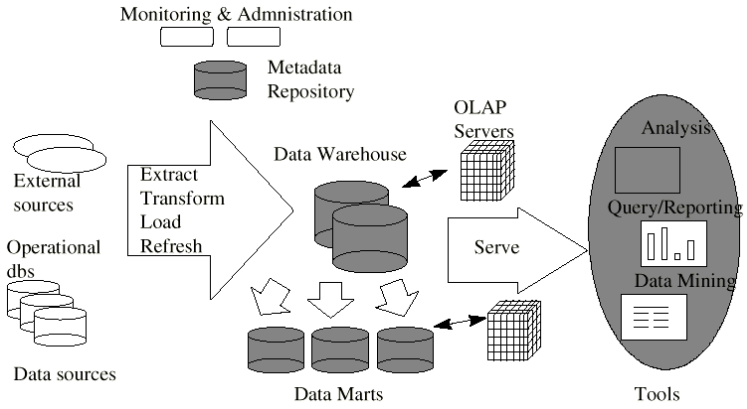
Las **fuentes de información** [*information sources*] pueden ser heterogéneas (incluir datos en distintos formatos: bases de datos relacionales, bases de conocimiento, documentos, páginas HTML...).

El **encapsulador** [*wrapper*], que traslada la información hacia el DW (normalmente off-line) y ha de convertir los datos al formato adecuado para el DW, incluye un **monitor** en contacto con la fuente de datos que detecta cuándo se producen modificaciones. La detección de cambios puede efectuarse en la propia fuente de información (la notificación se propaga por medio de disparadores) o realizarla el propio monitor consultando los archivos "log", efectuando consultas periódicas (que no deberían interferir en la utilización habitual del sistema OLTP) o realizando volcados de los datos. Obviamente, cada tipo de fuente de datos requerirá un encapsulador/monitor diferente.

El **integrador** [*integrator*] filtra, limpia, resume y unifica la información recibida desde las distintas fuentes. El DW puede verse como un conjunto de "vistas materializadas", por lo que la tarea del integrador se reduce a mantener dichas vistas, sin olvidar que el DW almacena información histórica que habitualmente no se mantiene en las bases de datos subyacentes.

Por otro lado, hay que tener en cuenta que la actualización de las vistas no suele realizarse a la vez que las transacciones sobre la base de datos. El mantenimiento del DW requiere el análisis de las actualizaciones y la comprobación de qué vistas se ven afectadas por ellas (filtrado de actualizaciones). También suelen incluirse en el DW vistas auxiliares con el objetivo de reducir el número de consultas a las fuentes de información (operaciones que son muy complejas) en el mantenimiento automático del DW y optimizar la materialización de las distintas vistas del DW.

Arquitectura detallada de un DW



Arquitectura detallada de un DW [Chaudhuri & Dayal, 1997]

La arquitectura incluye herramientas para extraer, limpiar, transformar e integrar datos provenientes de distintas fuentes. Además, se ha de cargar la información en el DW y ser puesta al día periódicamente. Los datos existentes en el DW y en los DMs son gestionados por servidores OLAP que presentan vistas multidimensionales de los mismos para la generación de informes, la realización de consultas mediante herramientas análisis exploratorio y el uso de herramientas de minería de datos [*Data Mining*]. Además, es necesario un almacén de metadatos.

El diseño y puesta en marcha de un DW requiere: definir la arquitectura, seleccionar los recursos hardware y software necesarios, integrar las fuentes de información, diseñar el esquema del DW y sus vistas asociadas, definir la organización física de los datos (situación, distribución y métodos de acceso), diseñar e implementar las herramientas de extracción de datos, limpieza, transformación y actualización, diseñar e implementar los interfaces de usuario...

Back End: Herramientas y utilidades

EXTRACCIÓN DE DATOS

La extracción de datos desde las fuentes de información externas al DW se suele realizar utilizando estándares (vg: ODBC, Oracle Open Connect...).

LIMPIEZA DE DATOS

Es necesaria porque los datos, provenientes de distintas fuentes, pueden incluir errores y anomalías tales como inconsistencias, valores perdidos, o violaciones de integridad. Se pueden emplear herramientas de migración de datos [*data migration*], herramientas de limpieza profunda [*data scrubbing*] y/o herramientas de inspección de datos [*data auditing*].

CARGA Y ACTUALIZACIÓN (REFRESCO) DE LOS DATOS

Tras extraer, limpiar y transformar los datos originales, éstos han de introducirse en el DW. En esta etapa se deben realizar los cálculos destinados a construir o actualizar incrementalmente las vistas del DW: ordenación, agregación, resumen... Este tipo de operaciones se suele realizar periódicamente cuando la actividad en el sistema es baja (vg: por las noches). El refresco suele hacer uso de técnicas de replicación (transporte de datos [*data shipping*] o transporte de transacciones [*transaction shipping*]).

Front End: Herramientas de usuario

El modelo multidimensional está inspirado en las tradicionales hojas de cálculo (que siguen siendo la aplicación de usuario más importante en OLAP). Las hojas de cálculo multidimensionales deben soportar *pivotaje*, *roll-up*, *drill-down*, *slice* y *dice* (las operaciones típicas de un "*data cube*"). Además de las hojas de cálculo, también se emplean entornos de consulta y herramientas de *Data Mining* para facilitar el análisis de los datos de un DW.

Gestión del DW

El DW refleja el modelo de una empresa y es esencial en su arquitectura el manejo de metadatos (una extensión del concepto de catálogo):

- **Metadatos administrativos** [*administrative metadata*]: Información necesaria para la puesta a punto y uso del DW.
- **Metadatos "del negocio"** [*business metadata*]: Términos y definiciones del dominio específico, propiedad de los datos...
- **Metadatos operacionales** [*operational metadata*]: Información recogida durante el funcionamiento del DW.

Existen múltiples herramientas que utilizan esta metainformación para planificar, diseñar o analizar un DW, así como gestionar y monitorizar su operación.

Servidores OLAP

Los DW han de mantener estructuras redundantes como índices y vistas materializadas para que su rendimiento sea óptimo. Además, dado el volumen de datos que han de manejar, es esencial el uso de paralelismo para reducir el tiempo de respuesta de las consultas.

ÍNDICES

Pueden ser índices tradicionales o índices de reunión (a través de claves externas), especialmente atractivos en servidores ROLAP (esquemas en estrella o en bola de nieve). Los índices en un DW utilizan RIDs o, preferiblemente, mapas de bits [*bitmaps*] con el objetivo de optimizar las consultas que involucren operaciones de intersección, unión, reunión o agregación.

VISTAS MATERIALIZADAS

Las consultas realizadas a un DW suelen requerir agregaciones, por lo que materializar los datos resumidos mejora el rendimiento del DW. Para ello hay que identificar las vistas que deben ser materializadas teniendo en cuenta la posibilidad de obtener una vista a partir de otra (generadores), utilizarlas para la resolución de consultas y actualizarlas en las operaciones de carga y refresco del DW.

PROCESAMIENTO DE CONSULTAS

Los servidores relacionales tradicionales no estaban preparados para el procesamiento OLAP (vg: uso inteligente de índices y vistas materializadas), por lo que han aparecido nuevas arquitecturas:

- **Servidores SQL especializados** para el procesamiento de consultas sobre esquemas en estrella o bola de nieve.
- **Servidores ROLAP**, que extienden la funcionalidad de los servidores relacionales clásicos (soporte de consultas OLAP, gestión de vistas materializadas...).
- **Servidores MOLAP**, que trabajan directamente con datos almacenados en estructuras multidimensionales [*data cubes*].

EXTENSIONES DE SQL

Se han realizado distintas propuestas para facilitar el procesamiento de consultas OLAP: funciones de agregación extendidas (percentiles, moda...), herramientas avanzadas de producción de informes (vg: uso de ventanas temporales), GROUP BY múltiples (operadores *cube* y *rollup*), comparaciones de datos derivados de agregaciones...

Características deseables de los productos OLAP

Igual que los complejos sistemas pre-relacionales fueron reemplazados por sistemas relacionales que mantienen la integridad de los datos y permiten manipularlos de una forma cómoda y eficaz, en OLAP éstos deberían ser reemplazados por otros más adecuados a la hora de realizar un análisis multidimensional de los datos. Codd ha propuesto 12 reglas que recogen características deseables de un producto OLAP:

1. Vista conceptual multidimensional de los datos (facilita el análisis y diseño de modelos de decisión).

2. Transparencia respecto al usuario (la complejidad del sistema no debe ser percibida por el usuario, con el fin de no disminuir su productividad).

3. Accesibilidad (el análisis ha de poder realizarse sobre datos provenientes de fuentes de datos heterogéneas, ofreciendo una vista unificada, coherente y consistente de los mismos)

4. Rendimiento (no debe disminuir al aumentar el tamaño de la base de datos o el número de dimensiones, para que el usuario no tenga que circunventar las limitaciones del sistema artificialmente).

5. Arquitectura cliente/servidor (el sistema OLAP ha de funcionar en un entorno cliente/servidor).

6. Dimensiones simétricas (todas las operaciones han de permitirse sobre cualquiera de las dimensiones).

7. Manejo óptimo de matrices poco densas (el almacenamiento físico de los datos ha de ajustarse a la distribución de sus valores).

8. Soporte multi-usuario (las herramientas OLAP deben soportar el acceso concurrente de varios usuarios).

9. Operaciones sin restricciones entre dimensiones (para lo cual el producto OLAP debe incluir un lenguaje adecuado que permita la especificación de operaciones entre cualquier número de dimensiones)

10. Manipulación intuitiva de los datos (las operaciones típicas de un *data cube* deberían poder realizarse directamente sobre una hoja de cálculo [por ejemplo, con el ratón]: reorientación, drill-down, roll-up...).

11. Generación de informes flexible (tanto filas como columnas han de poder incluir cualquier número de dimensiones en cualquier orden, mostrando para cada dimensión cualquier subconjunto de datos en cualquier orden).

12. Número de dimensiones y niveles de agregación ilimitados (una herramienta OLAP debería permitir definir modelos multidimensionales con 20 dimensiones con un número ilimitado de niveles de agregación).

Ejemplos de productos OLAP comerciales

- ✓ Hyperion Essbase
- ✓ Oracle 8i for Data Warehousing

Líneas de investigación

Hay varios aspectos relacionados con los DWs en los que aún queda mucho por hacer:

- Limpieza de datos (búsqueda de inconsistencias en los datos y en los esquemas de las fuentes de datos)
- Diseño de DW (selección de índices, particionamiento de los datos, selección de vistas materializadas...)
- Gestión de DW (detección de cuellos de botella y asignación de recursos, técnicas de actualización incremental...)

Referencias

S. CHAUDHURI & U. DAYAL:
"An Overview of Data Warehousing and OLAP Technology"
ACM SIGMOD Record, 1997

E.F. CODD, S.B. CODD & C.T. SALLEY:
"Providing OLAP to User-Analysts"
E.F. Codd & Associates, 1993

HYPERION
"The Role of the OLAP Server in a Data Warehousing Solution"
Hyperion Solutions Corporation, 1998

ORACLE
"Oracle 8i for Data Warehousing: Features Overview"
Oracle Corporation, February 1999

ORACLE
"Oracle 8i for Data Warehousing:
Fast and Simple for More Data and More Users. An Oracle Technical White Paper"
Oracle Corporation, February 1999

J. WIDOM
"Research Problems in Data Warehousing"
CIKM'96, Baltimore MD USA, 1996