

UNIDAD II: ALMACENES DE DATOS Y PROCESOS DE CONSTRUCCIÓN;

1 REPASO DEL CONCEPTO: BUSINESS INTELLIGENCE(BI)

La inteligencia de negocios o business intelligence (BI) es el conjunto de procesos, metodologías, estrategias, aplicaciones y tecnologías que facilitan la obtención rápida y sencilla de todos los datos generados por una empresa para su análisis e interpretación, de manera que puedan ser aprovechados para la toma de decisiones y se conviertan en conocimiento para los responsables del negocio.

BI es la combinación de conceptos, infraestructura técnica y procedimientos que abarca:

- La extracción, transformación y carga de datos (llamados “procesos ETL”) desde las diferentes fuentes de origen como bases de datos transaccionales, planillas de cálculo, etc., en un repositorio concebido para el análisis y visualización de los mismos (llamado Data Warehouse).
- El almacenamiento y administración de los datos en este repositorio (Data Warehouse)
- La definición de indicadores clave a ser medidos (llamados “kpi”).
- El análisis de datos y la medición de indicadores (evaluación de procesos, evaluación comparativa del rendimiento, el análisis descriptivo, etc.) mediante herramientas de; análisis. El análisis comprende el multidimensional (OLAP).
- La presentación de resultados mediante herramientas de visualización (llamados “Tableros de Control”, “Dashboards”, etc.)

Ésta fascinante rama de la tecnología surgió de la convicción de que las personas con información inexacta o incompleta tenderán a tomar decisiones peores que si tuvieran mejor información. Los creadores de modelos financieros reconocerán esto como un problema de "garbage in, garbage out" (“entra basura → sale basura”).; Los 3 componentes básicos de BI (ETL – ALMACEN – VISUALIZACIÓN)

Figura 1. Arquitectura y Componentes Básico DW; Nota: Las fuentes de datos externas (Base de datos Operacionales, Archivos TXT, CSV, Excel, log de Registros Web, etc.) no corresponden a un componente básico. Son las fuentes de; datos, ENTRADA o INPUT de un proyecto BI.

CONCEPTOS RELACIONADOS

Data Warehouse

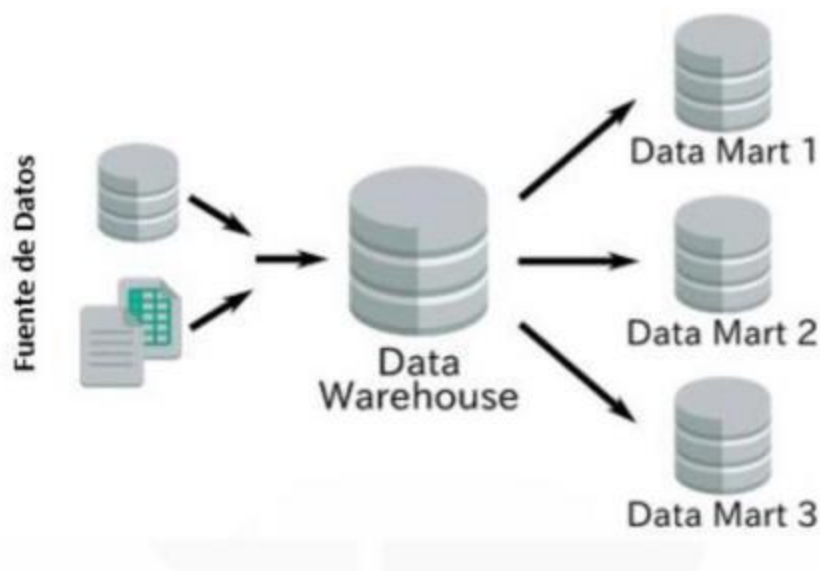
Un Data Warehouse (DW) es un gran repositorio lógico de datos que permite el acceso y manipulación flexible de grandes volúmenes de información provenientes tanto de transacciones detalladas como datos agregados de fuentes de distintas naturalezas (Archivos planos, csv, planillas de cálculo, etc.).

Datamart

Los Data Mart son subconjuntos de los DW, es decir que contienen subconjuntos de datos de toda la organización que son valiosos para diferentes grupos específicos de personas.

Por ejemplo, dentro del DW, el Data Mart de marketing contiene sólo datos relacionados con artículos, clientes y ventas.

La siguiente figura muestra una representación gráfica de un Data Mart:



ETL:

ETL, Extract – Transform – Load (Extracción - Transformación - Carga). Son procesos armados necesarios para la depuración y adecuación de los datos requeridos en el DW.

Estos procesos leen los datos de toda fuente que tenga información requerida en el DW, procesan las transformaciones necesarias y cargan en las tablas intermedias o finales del DW.

Para el armado de los procesos existen en el mercado de BI, herramientas propietarias y del tipo Open Source.

EXPLOTACION DE DATOS - VISUALIZACIÓN

El proceso de explotación de datos o visualización corresponde a las diferentes herramientas que se usan en BI para la presentación de la información (Dashboards), el análisis multidimensional o reportes ad-hoc dinámicos (reportes que con parámetros pueden seleccionar y filtrar datos).

2 DATA WAREHOUSE – ALMACEN DE DATOS;

Introducción

El término "Almacén de datos" o Data Warehouse (DW) fue acuñado por primera vez por Bill Inmon en 1990. Según Inmon, un almacén de datos es una recopilación de datos orientada por temas, integrada, con variación temporal y no volátil. Estos datos ayudan a los analistas a tomar decisiones informadas en una organización.

Una base de datos operativa sufre cambios frecuentes a diario debido a las transacciones que tienen lugar. Supongamos que un ejecutivo de negocios desea analizar comentarios anteriores sobre cualquier dato como un producto, un proveedor o cualquier dato del consumidor, entonces el ejecutivo no tendrá datos disponibles para analizar porque los datos anteriores se han actualizado debido a transacciones.

Un DW nos proporciona datos generalizados y consolidados en una vista multidimensional.

Junto con una vista generalizada y consolidada de datos, un almacén de datos también nos proporciona herramientas de procesamiento analítico en línea (OLAP). Estas herramientas nos ayudan en el análisis interactivo y efectivo de datos en un espacio multidimensional.

Este análisis da como resultado la generalización de datos y la minería de datos.

Las funciones de minería de datos, como asociación, agrupación, clasificación, predicción, se pueden integrar con las operaciones OLAP para mejorar la minería interactiva del conocimiento en múltiples niveles de abstracción. Es por eso que el DW se ha convertido en una plataforma importante para el análisis de datos y el procesamiento analítico en línea.

Comprensión de un DW

- ➤ Un DW es un sistema de almacenamiento de datos.
- ➤ Un DW es una base de datos, que se mantiene separada de la base de datos operativa de la organización. En algunas organizaciones son bases de datos que comparten el mismo servidor físico o virtual. Lo óptimo es que se encuentren en servidores físicos (o virtuales) separados.
- ➤ No se realizan actualizaciones frecuentes en un DW.
- ➤ Posee datos históricos consolidados, que ayudan a la organización a analizar su negocio.
- ➤ Un DW ayuda a los ejecutivos a organizar, comprender y usar sus datos para tomar decisiones estratégicas.
- ➤ Los DW ayudan en la integración de la diversidad de los sistemas de aplicaciones.
- ➤ Un DW ayuda en el análisis consolidado de datos históricos.

Por qué un DW está separado de las bases de datos operativas

Los DW se mantienen separados de las bases de datos operativas debido a los siguientes motivos:

- Se crea una base de datos operativa para tareas y cargas de trabajo bien conocidas, como la búsqueda de registros particulares, la indexación, etc. Por el contrario, las consultas del DW a menudo son complejas y presentan una forma general de datos.
- Las bases de datos operativas admiten el procesamiento concurrente de múltiples transacciones. Se requieren mecanismos de control y recuperación de concurrencia para las bases de datos operativas para garantizar la solidez y la consistencia de la base de; datos.
- Una consulta de base de datos operativa permite leer y modificar operaciones, mientras que una consulta OLAP solo necesita acceso de solo lectura de los datos almacenados.
- Una base de datos operativa mantiene los datos actuales. Por otro lado, un DW mantiene datos históricos.

Características del almacén de datos

Las características clave de un almacén de datos se analizan a continuación:

- Orientado a temas: un DW está orientado a temas porque proporciona información sobre un tema en lugar de las operaciones en curso de la organización. Estos temas pueden ser productos, clientes, proveedores, ventas, ingresos, etc. Un DW no se centra en las operaciones en curso, sino que se centra en el modelado y análisis de datos para la toma de decisiones.
- Integrado: un DW se construye integrando datos de fuentes heterogéneas, como bases de datos relacionales, archivos planos, etc. Esta integración mejora el análisis efectivo de los datos.

- Variante de tiempo: los datos recopilados en un almacén de datos se identifican con un período de tiempo particular. Los datos en un almacén de datos proporcionan información desde el punto de vista histórico.
- No volátil: no volátil significa que los datos anteriores no se borran cuando se agregan nuevos datos. Un almacén de datos se mantiene separado de la base de datos operativa; y, por lo tanto, los cambios frecuentes en la base de datos operativa no se reflejan en el almacén de datos.

Nota: Un DW no requiere controles de procesamiento de transacciones, recuperación y; concurrencia, ya que está físicamente almacenado y separado de la base de datos; operativa. Aunque algunas organizaciones por ahorrar costos quieren implementar la base de datos del DW en el mismo servidor de la Base de Datos Operacional esto no es; RECOMENDABLE.

Aplicaciones de almacenamiento de datos

Como se describió, un DW ayuda a los ejecutivos de negocios a organizar, analizar y usar sus datos para la toma de decisiones. Un DW sirve como parte única de un sistema de retroalimentación de "circuito cerrado" de planificación - ejecución - evaluación para la gestión empresarial. Los DW se usan ampliamente en los siguientes campos:

- Servicios financieros
- Servicios bancarios.
- Bienes de consumo
- Sectores minoristas
- Fabricación controlada.
- Sector Educación.

Nota: actualmente las Pymes también se encuentra implementando soluciones de Inteligencia de Negocio, por tal motivo es requerido la implementación de DW, en menor escala y dimensión comparado con una gran empresa. También cabe aclarar que también se ven beneficios en implementar soluciones BI, en niveles de mandos medio y gerenciales, no solo en el tradicional nivel Directivo o Ejecutivo.

Tipos de DW

El procesamiento de la información, el procesamiento analítico y la minería de datos son los tres tipos de aplicaciones de un DW que se analizan a continuación:

- de información: un almacén de datos permite procesar los datos almacenados en él. Los datos pueden procesarse mediante consultas, análisis estadísticos básicos, informes mediante tablas cruzadas, tablas, cuadros o gráficos.
- Procesamiento analítico: un almacén de datos admite el procesamiento analítico de la información almacenada en él. Los datos pueden analizarse mediante operaciones OLAP básicas, que incluyen explorar modelos multidimensionales, explorar hacia abajo (Drill Down), reducir exploración hacia arriba (Drill up) y pivotar (Cambiar entre filas y columnas).
- Minería de datos (Data Mining): la minería de datos admite el descubrimiento de conocimiento mediante la búsqueda de patrones y asociaciones ocultos, la construcción de modelos analíticos, la clasificación y la predicción. Estos resultados de minería se pueden presentar utilizando herramientas de visualización.

Arquitectura de un DW

La arquitectura general de un DW es la que se muestra en la Figura 2. Este diagrama muestra como primer componente dentro de la arquitectura de un DW a las fuentes de datos desde las cuales se extrae la información necesaria para poblar el DW. Conectada a cada una de las fuentes se encuentran los siguientes componentes básicos de la arquitectura, los wrappers o extractores, los cuales extraen y transforman la información de las fuentes.

Posteriormente través de un integrador dicha información se carga al DW, la cual constituye el siguiente componente básico de la arquitectura. Este proceso de cargado de la información ejecuta las tareas siguientes:

- Transforma los datos de acuerdo al modelo de datos del Warehouse.
- Limpia dichos datos para corregir y depurar errores que pueden contener las fuentes.
- Integra todos los datos para formar la base de datos en la cual se encontrará la; información.

De igual manera, los metas datos deben ser refrescados dentro de este proceso. Dicho proceso es crítico para asegurar la calidad de la información y soportar una adecuada toma de decisiones con datos correctos y previamente verificados. Una vez que los datos han sido cargados se encuentran disponibles para un sistema que soporte decisiones.

Sin embargo, las aplicaciones no acceden directamente el Warehouse debido a que es demasiado grande, además de poseer un esquema genérico no óptimo para el usuario final. Por consiguiente, vistas especializadas más pequeñas del DW son cargadas en los data marts, éstos son repositorios más pequeños con vistas materializadas para facilitar la consulta de los datos. Esta carga se realiza a través de un segundo proceso más simple debido a que los datos ya se encuentran ordenados y verificados dentro del DW.

Únicamente se seleccionan las vistas requeridas y a través de una serie de transformaciones necesarias quedan establecidas para facilitar y acelerar el proceso de consulta del usuario. Finalmente, los data marts son accedidos a través de las herramientas para el usuario final (OLAP o ambientes de consultas analíticas, generalmente), las cuales permiten analizar la; información disponible en el Warehouse para la generación de consultas especializadas, reportes, nuevas clasificaciones y tendencias que sirvan de apoyo a la toma de decisiones.

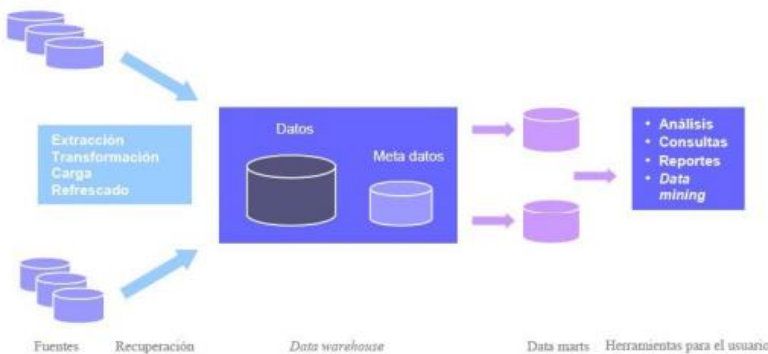


Figura 2. Arquitectura General DW

Diferencia entre OLTP Y OLAP

Data Warehouse (OLAP)	Operational Database (OLTP)
Implica el procesamiento histórico de la información.	Implica el procesamiento del día a día.
Los sistemas OLAP son utilizados por trabajadores del conocimiento como ejecutivos, gerentes y analistas.	Los sistemas OLTP son utilizados por empleados, DBA o profesionales de bases de datos.
Se utiliza para analizar el negocio.	Se utiliza para ejecutar el negocio.
Se centra en la información de salida.	Se centra en la información de entrada.
Se basa en Esquema Estrella (Star), Esquema Copo de Nieve (Snowflake) y Es basado en el modelo Entidad-Relación.	Es basado en el modelo Entidad-Relación.
Es orientado al análisis.	Está enfocado en aplicaciones.
Contiene datos históricos.	Se basa en datos corrientes (las aplicaciones usan dato corriente)
Proporciona datos resumidos y consolidados.	Proporciona datos primitivos y altamente detallados
Proporciona una vista resumida y multidimensional de los datos.	Proporciona una vista relacional detallada y plana de los datos.
Se enfoca en un acceso rápido y flexible de la información	Se enfoca en el control de concurrencia y consistencia de la base de datos.

3 DATA WAREHOUSE – CONCEPTOS

¿Qué es Data Warehousing?

El Data Warehousing es el proceso de construcción y uso de un DW. Un DW se construye integrando datos de múltiples fuentes heterogéneas que admiten informes analíticos, consultas estructuradas y / o ad hoc y toma de decisiones. El almacenamiento de datos implica la limpieza de datos, la integración de datos y la consolidación de datos.

Uso de la información del DW

Existen tecnologías de soporte de decisiones que ayudan a utilizar los datos disponibles en un DW. Estas tecnologías ayudan a los ejecutivos a utilizar el almacén de manera rápida y efectiva. Pueden recopilar datos, analizarlos y tomar decisiones basadas en la información presente en el almacén.

La información recopilada en un almacén se puede utilizar en cualquiera de los siguientes dominios:

- Ajuste de estrategias de producción: las estrategias de productos pueden ajustarse bien al reposicionar los productos y administrar las carteras de productos comparando las ventas trimestrales o anuales.
- Análisis del cliente: el análisis del cliente se realiza analizando las preferencias de compra del cliente, el tiempo de compra, los ciclos presupuestarios, etc.
- Análisis de operaciones: el almacenamiento de datos también ayuda en la gestión de las relaciones con los clientes y en la realización de correcciones ambientales. La información también nos permite analizar las operaciones comerciales.

Integrando bases de datos heterogéneas

Para integrar bases de datos heterogéneas, tenemos dos enfoques:

- A. Enfoque basado en consultas
- B. Enfoque basado en actualizaciones;

A - Enfoque basado en consultas

Este es el enfoque tradicional para integrar bases de datos heterogéneas. Este enfoque se usó para construir contenedores e integradores sobre múltiples bases de datos heterogéneas. Estos integradores también se conocen como mediadores.

Proceso de enfoque dirigido por consultas

1. Cuando se emite una consulta al lado del cliente, un diccionario de metadatos traduce la consulta a una forma apropiada para sitios heterogéneos individuales involucrados.
2. Ahora estas consultas se asignan y se envían al procesador de consultas local.
3. Los resultados de sitios heterogéneos se integran en un conjunto de respuestas global.

Desventajas

- El enfoque basado en consultas necesita procesos complejos de integración y filtrado.
- Este enfoque es muy ineficiente.
- Es muy costoso para consultas frecuentes.
- Este enfoque también es muy costoso para consultas que requieren agregaciones.

B - Enfoque basado en actualizaciones

Esta es una alternativa al enfoque tradicional. Los sistemas actuales de almacenamiento de datos siguen un enfoque basado en actualizaciones en lugar del enfoque tradicional discutido anteriormente. En el enfoque basado en actualizaciones, la información de múltiples fuentes heterogéneas se integra por adelantado y se almacena en un almacén.

Esta información está disponible para consultas y análisis directos.

Ventajas

Este enfoque tiene las siguientes ventajas:

- Este enfoque proporciona un alto rendimiento.
- Los datos se copian, procesan, integran, anotan, resumen y reestructuran en el almacén de datos semántico por adelantado.
- El procesamiento de consultas no requiere una interfaz para procesar datos en fuentes locales.

Funciones de las herramientas y utilidades de un DW

Las siguientes son las funciones de las herramientas y utilidades del almacén de datos:

- Extracción de datos: implica la recopilación de datos de múltiples fuentes heterogéneas.
- Limpieza de datos: implica encontrar y corregir los errores en los datos.
- Transformación de datos: implica convertir los datos del formato heredado al formato del DW.
- Carga de datos: implica ordenar, resumir, consolidar, verificar la integridad y crear índices y particiones.

- Actualización: implica la actualización desde los orígenes de datos al DW.

Nota: La limpieza de datos y la transformación de datos son pasos importantes para mejorar la calidad de los datos y los resultados de la minería de datos.

Estas funciones las ejecutan las Herramientas denominadas ETL (Extract -Transform- Load).

4 DATA WAREHOUSE – TERMINOLOGIA

Metadatos (Metadata)

Los metadatos se definen simplemente como datos sobre datos. Los datos que se utilizan para representar otros datos se conocen como metadatos. Por ejemplo, el índice de un libro sirve como metadatos para los contenidos del libro. En otras palabras, podemos decir que los metadatos son los datos resumidos que nos llevan a los datos detallados.

En términos de almacenamiento de datos, podemos definir metadatos de la siguiente manera:

- Los metadatos son una hoja de ruta para el almacén de datos.
- Los metadatos en el almacén de datos definen los objetos del almacén.
- Los metadatos actúan como un directorio. Este directorio ayuda al sistema de soporte de decisiones a localizar el contenido de un almacén de datos.

Repositorio de Metadatos (Metadata Repository)

El repositorio de metadatos es una parte integral de un sistema de almacenamiento de datos. Contiene los siguientes metadatos:

- Metadatos empresariales: contiene la información de propiedad de los datos, la definición empresarial y las políticas cambiantes.
- Metadatos operativos: incluye la moneda de datos y el linaje de datos. La moneda de los datos se refiere a los datos que están activos, archivados o purgados. El linaje de datos significa el historial de datos migrados y la transformación aplicada en ellos.
- Datos para mapear del entorno operativo al almacén de datos: los metadatos incluyen bases de datos de origen y sus contenidos, extracción de datos, partición de datos, limpieza, reglas de transformación, actualización de datos y reglas de purga.
- Los algoritmos para el resumen: incluye algoritmos de dimensión, datos sobre granularidad, agregación, resumen, etc.

Cubo de datos (Date Cube)

Un cubo de datos es una estructura lógica que nos ayuda a representar datos(hechos) en múltiples dimensiones. Se define por dimensiones(dimensions) y hechos(facts).

Por ejemplo, los hechos(facts) pueden ser las ventas de una empresa y las dimensiones serían la sucursal, el vendedor, la localidad, el tipo de producto vendido, el tiempo, etc. De esta manera, se pueden analizar las ventas sumando, agrupando o filtrando por cada una de esas dimensiones. Si tuviésemos dos dimensiones, podríamos realizar este análisis mediante un cuadro de doble entrada en una planilla de cálculo; al sumarse una tercera dimensión, la representación sería un cubo; al tener n dimensiones, es imposible visualizar los datos gráficamente, por lo que se necesitan herramientas que permitan hacer análisis multidimensionales.

Estructura de un Datawarehouse, se compone de:

La estructura de un DW contiene básicamente dos tipos de tablas donde se almacena toda la información: tablas fact y dim.

- “tablas fact” o tablas de hechos (ventas, facturación, etc.)
- “tablas dim” o tablas de dimensiones (Sucursal, Producto, Vendedor, Tiempo).

¿Qué son los fact y las tablas fact?

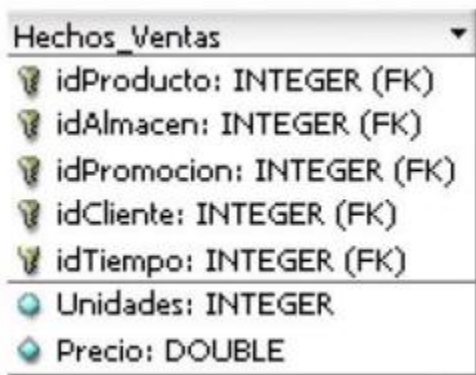
Las tablas fact contienen los datos correspondientes a un proceso de negocio en particular. Así, en un DW de una empresa es habitual que existan las siguientes tablas: Fact_ventas, Fact_pagos, Fact_facturaciones, Fact_cobros, etc.

En una tabla fact, cada fila representa un HECHO, es decir, un evento único asociado con un proceso, y esa fila contiene los campos cualitativos(atributos) y cuantitativos(medidas) asociados con ese evento.

Por ejemplo, la tabla Fact_ventas probablemente contenga los campos de información:

- Atributos: cliente, fecha, tipo de venta, canal de venta, sucursal, localidad, etc.
- Medidas: monto de la venta, descuento aplicado, valor del IVA, etc.

Las medidas (campos numéricos) contenidas en una tabla fact suelen ser fácilmente manipulables para hacer cuentas aritméticas incluso entre miles o millones de registros. En el ejemplo anterior, el usuario podría calcular de manera muy simple el total de ventas, el monto promedio, etc., y a su vez obtener estas métricas(mediciones) por localidad, por cliente, por mes, etc., es decir, por cada uno de los atributos o dimensiones.



Hechos_Ventas	
idProducto:	INTEGER (FK)
idAlmacen:	INTEGER (FK)
idPromocion:	INTEGER (FK)
idCliente:	INTEGER (FK)
idTiempo:	INTEGER (FK)
Unidades:	INTEGER
Precio:	DOUBLE

Granularidad de una tabla fact

Al diseñar una tabla fact, los desarrolladores deben prestar mucha atención a la granularidad de la tabla, que es el nivel de detalle contenido dentro de la tabla.

En el ejemplo anterior, al diseñar la tabla se debería decidir si el nivel de detalle de la tabla será (por ejemplo):

- una venta
- un ítem dentro de una venta

En el caso de llegar a un nivel de detalle de compra de un ítem individual, cada transacción de venta generaría múltiples registros en la tabla de hechos, correspondientes a cada ítem de venta.

La definición de la granularidad de una tabla fact es una decisión fundamental tomada durante el proceso de diseño que puede tener un impacto significativo en el núcleo de BI en el futuro.

¿Qué son las dimensiones y las tablas de dimensiones?

Las dimensiones describen los diferentes atributos que tiene un hecho de la tabla fact. En el ejemplo anterior, las dimensiones serían Tiempo, Cliente, Localidad, Tipo de venta, Sucursal, Canal de venta, etc.

Una tabla de dimensión contiene en detalle todos los valores posibles de un atributo, y a su vez contiene todas las características de cada uno de esos valores. Por ejemplo, la tabla de dimensión “dim_canal_de_venta” contendría todos los diferentes canales de venta para una transacción, con los campos:

- Código o “id” del canal de venta
- Nombre del canal de venta
- Descripción extensa del canal de venta



Las tablas fact (Hecho) y dim (Dimensión) están relacionadas entre sí a través de los códigos o “id” de cada dimensión. En la imagen anterior, la relación entre la fact_ventas y la dim_zona sería a través del campo “idzona”.

La forma de relacionar tablas fact con tablas dim es a través de claves foráneas, en el ejemplo la tabla fact_ventas (Hecho_Ventas) contiene una referencia de clave externa (foreign key) a la dim_zona, donde idzona es una clave principal (primary key) en esa tabla.

Atributos y Medidas

Los atributos o dimensiones son los diferentes campos cualitativos que permiten categorizar un hecho. Por ejemplo, si mi tabla de hechos es de facturación, cada fila será una factura y en cada factura tendremos seguramente los siguientes atributos: fecha, cliente, sucursal, localidad, etc.

Por otro lado, las medidas son los valores numéricos; en nuestro ejemplo, las medidas podrían ser el monto facturado, la cantidad artículos de la factura, etc.

Así, los atributos o dimensiones agregan contexto a las medidas y permiten realizar análisis multidimensionales: por localidad, lo acumulado de un mes, la facturación por cliente, etc.

5 DATA WAREHOUSE – DISEÑO

Estructura y Esquemas (Modelado) de un Data Warehouse

El Data Warehouse se compone básicamente de tablas Fact y tablas Dim. En algunas ocasiones por cuestiones de performance de carga del DW se crean tablas Staging (tablas intermedias), dichas tablas en pueden ser útiles para carga masiva de información con el mismo formato que la fuente que luego será la entrada del proceso de Transformación de un ETL.

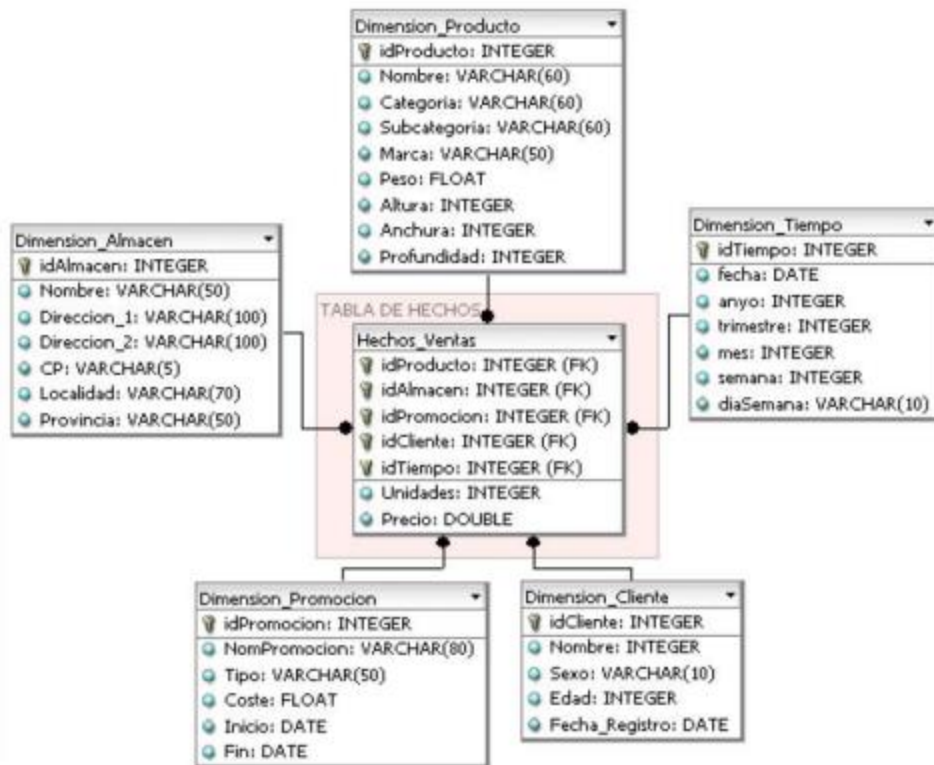
En el proceso de diseño existen 2 modelados (opciones de diseño):

- Modelo en Estrella (schema Star)
- Modelo Copo de Nieve (schema Snowflake).

Modelo Estrella (Modelo Star)

El modelo estrella es el esquema más simple y más ampliamente utilizado de un DW. Incluye una tabla fact y n tablas dim que se relacionan a la tabla fact mediante claves externa (es decir, códigos). El esquema estrella es fundamentalmente útil para el manejo de consultas básicas.

Su nombre se debe a que su modelo de relaciones se asemeja a la forma de una estrella, con una tabla fact en su centro y las tablas dim en su periferia.



Características del modelo estrella:

- Cada dimensión se representa con la única tabla de una dimensión.
- La tabla de dimensiones debe contener el conjunto de atributos de esa dimensión.
- La tabla de dimensiones se une a la tabla de hechos mediante una clave externa.
- Las tablas de dimensiones no están unidas entre sí.
- Las tablas de dimensiones no están normalizadas.
- El esquema es ampliamente soportado por todas las herramientas de BI.

Ventajas del modelo estrella

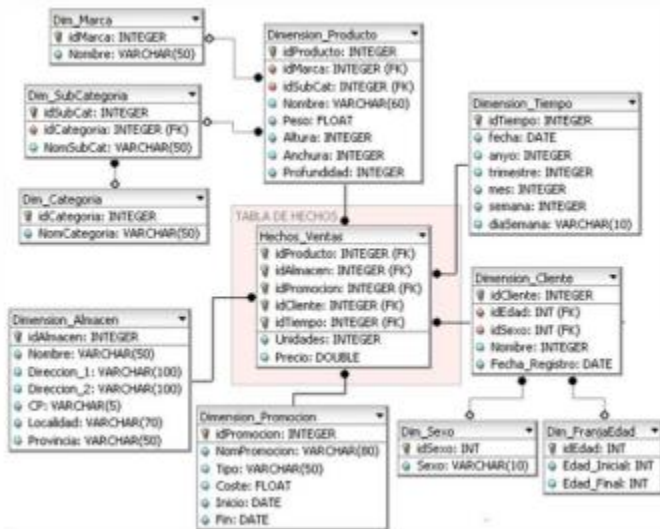
- Consultas más simples: la lógica de unión del modelo estrella es bastante sencilla en comparación con otra lógica de unión que se necesita para obtener datos de un esquema transaccional que está altamente normalizado.
- Lógica de reporting de negocios simplificada: en comparación con un esquema transaccional que está altamente normalizado, el esquema en estrella simplifica la lógica común de reporting de negocios, como los informes de fecha y de finalización del período.

Desventajas del modelo estrella

La integridad de los datos no se aplica correctamente ya que se encuentra en un estado de esquema que no está normalizado. Por ejemplo, si en la tabla fact tengo un campo país y otro campo provincia, podría existir un registro que contenga los valores Uruguay y Córdoba respectivamente (en forma errónea) y, sin embargo, no habría ninguna restricción para comprobar esa integridad.

Modelo Snowflake (copo de nieve)

El esquema de Snowflake es una variante del esquema en estrella. Aquí, la tabla de hechos centralizada sigue relacionada a múltiples tablas dim, pero, a su vez, ahora estas tablas dim pueden estar conectadas a otras tablas dim.



Las tablas dim más “externas” se unen a las tablas dim más cercanas a la tabla fact, mediante códigos o claves externas (foreign key).

En este esquema, se logra una mayor normalización. Por ejemplo, en el modelo estrella podríamos tener una dim_provincia que tenga como uno de sus atributos el campo “país”, por lo tanto, el valor “Argentina” aparecería muchas veces (una vez por cada provincia de la tabla); en cambio, en el modelo Snowflake, se podría tener una dim_provincia relacionada a una dim_pais, por lo tanto, en cada registro de la dim_provincia tendríamos un código de país (“id_pais”) que se relacionaría a la dim_pais y, así, cada país sólo aparecería una sola vez en el DW.

Este esquema se utiliza cuando las dimensiones tienen varios niveles de relación. El efecto de Snowflake afecta solo a las tablas de dimensiones y no afecta a las tablas fact.

Características del modelo Snowflake:

- Usa poco espacio en disco
- La tabla dim consta de una clave primaria (código de provincia, en el ejemplo) y otras claves externas (código de país, en el ejemplo) que definen la información en diferentes niveles de granularidad.

Ventajas del modelo Snowflake:

- Proporciona datos estructurados que reducen el problema de la integridad de los datos.
- Utiliza un menor espacio en disco porque los datos están altamente estructurados, aunque ese ahorro suele ser insignificante frente al tamaño del DW.

Desventajas del modelo Snowflake:

- Se reduce el rendimiento, ya que las consultas son más complejas.

Proceso de diseño de un Data Warehouse (Pasos)

El proceso de diseño de un Data Warehouse se compone de varios pasos que detallaremos a continuación.

Antes de iniciar el diseño de un DW considerar lo siguiente:

Reconocer que el trabajo será más duro de lo que se esperaba inicialmente.

Es muy habitual encontrarse con que más de un 30 % de la información contenida en los sistemas operaciones o es incorrecta o incompleta para incorporarla al DW. Esta mala calidad de los datos incide en la complejidad del trabajo.

Otro ejemplo, es el de los administradores de la base de datos, que usan números en lugar de nombres de ciudades para optimizar el cálculo en sistemas operacionales. En un entorno Business Intelligence, este tipo de 'trucos' no son muy útiles, pues los usuarios necesitan descripciones completas.

Otro tema que hace el trabajo más duro del esperado, es la aparición de nuevos productos o divisiones durante el proceso de implementación. Algo habitual dada la duración del mismo.

Pasos para el diseño e implementación de un DW:

- 1) Se debe relevar que áreas formarán parte del proyecto de BI. Esto se logra en reuniones con el equipo Directivo, son los que deben determinar qué áreas son de interés y el alcance. Además, se debe acordar cuáles son los aspectos de interés que el Directivo considera importante para medir y controlar.

Nota: En algunas Organizaciones ya se encuentra diseñado un Plan Estratégico por lo cual ya se dispone de los Objetivos e indicadores de desempeño clave (KPI) a medir.

- 2) Una vez cerrado con la alta dirección se deberá avanzar en reuniones con cada área que componga el proyecto BI.

Consejo: Siempre es preferible dar resultados rápidos para comprometer más a la alta dirección y mandos medios. Por tal motivo se aconseja de una vez terminado de relevar el área de mayor interés por la Dirección o Directivos avanzar rápidamente en los siguientes pasos hasta llegar a un producto que se pueda mostrar y luego trabajar en los ajustes. Luego avanzar en las demás áreas.

- 3) Conocer los datos en los sistemas origen.

Antes de comenzar a construir el DW es muy importante analizar los datos y sus interrelaciones entre todas las Bases de Datos disponibles.

Posteriormente, al migrar esa información al DW, será necesario mantener esas relaciones, por lo que es muy importante hacerlo bien para evitar inconsistencias en el modelo de datos que pueden provocar muchos quebraderos de cabeza.

- 4) Definir donde implementar el DW.

Es necesario definir antes de crear las tablas Fact y Dim en qué sistema de Base de Datos se va a implementar el DW, se debe considerar si la organización o empresa ya dispone de una Base de Datos para tal fin. Si utilizan Base de Datos propietarias o no se oponen a una Base de Datos no licenciada

(MySQL, PostgreSQL). Estimar el volumen de información a gestionar, si conviene implementar el DW en un servidor dentro de la empresa u organización, o conviene una implementación CLOUD que permita ir creciendo en requerimientos de hardware a medida que crece los requerimientos del DW.

Nota: No hay que infravalorar los requerimientos de Hardware.

En un DW, los requerimientos de hardware son uno de los principales temas a valorar. A veces, se diseña una arquitectura que puede ser ampliamente suficiente para la entrada en producción, pero a menudo se olvida que estos sistemas crecen muy rápidamente, se necesitan sistemas de copia seguros y las necesidades de rendimiento, en términos de agilizar los cálculos son muy importantes. Por eso, nunca conviene infravalorar el número de CPU's, memoria RAM y disco disponibles.

5) Saber reconocer entidades equivalentes.

Uno de los principales problemas que surgen cuando se analizan sistemas heterogéneos, es la de identificar como una misma entidad, elementos que aparecen con nombres y descripciones diferentes, pero que se refieren a lo mismo.

Por ejemplo, dos departamentos diferentes (Comercial y Finanzas), pueden estar registrando en sus sistemas información sobre un mismo cliente, pero puede que este registrado con nombre diferentes (nº cliente, nº fiscal, nombre social, etc....);

6) Calidad de los Datos.

Es importante antes del diseño e implementación del DW en trabajar cuidadosamente en la calidad de los datos, se debe verificar si está garantizado que las fuentes tienen datos de confianza y si está garantizado la calidad. Si no esta garantizado desde las fuentes la calidad se debe trabajar en ello para determinar si es necesario una depuración previa o en los procesos de carga se implementará lógica de depuración y / o corrección de datos.

7) Usar metadatos como soporte a la calidad de los datos.

El uso de metadatos (datos sobre los datos), es crucial para el éxito de un DW. Es muy importante empezar a recoger y almacenar metadatos desde las fases iniciales del proyecto e incluir todas las fases del mismo.

También es muy útil integrar todos los metadatos en un lugar común. Esto será especialmente interesante cuando estemos trabajando con diferentes herramientas, cada una de las cuales, genera sus propios metadatos.

8) Diseñar e implementar las tablas Fact y Dim para un Data Mart.

Es importante que el diseño del DW crezca por parte, una vez que se arranca por un área en particular (Por ejemplo, Ventas o Marketing) es necesario diseñar e implementar las tablas Fact y Dim que dé soporte a los análisis requeridos (Datos obtenidos de los puntos 1 y 2). Si es necesario es importante volver a reunirse con los interesados del análisis antes de implementar el Data Mart como parte del DW.

La reducción de la complejidad de estos sistemas y el enfoque incremental utilizado en su creación, hacen que se pueda empezar a ver algunos frutos en un corto plazo de tiempo.

Esto tiene que ser aprovechado para que la organización valide lo realizado y apoye con sugerencias y compromiso los nuevos desarrollos que aún están pendientes.

9) Seleccionar las herramientas ETL adecuadas.

Las herramientas ETL, se encargan de la extracción de datos de los sistemas fuente, de su transformación y posterior carga en el DW o en algún sistema intermedio para posteriores transformaciones.

A la hora de seleccionar una herramienta ETL, será muy útil que tenga un manejo sencillo y represente de forma visual todas las transformaciones. También se debe evaluar si la Base de Datos donde se implementa el DW cuenta con herramientas ETL integradas (Por ejemplo, Microsoft SQL Server dispone de sus propias herramientas, y si se dispone de las licencias adecuadas estas brindan funcionalidades ya integradas en la Base de Datos, como por ejemplo el manejo de JOB o tareas programadas), otros ejemplos son Oracle, IBM, etc.

Así mismo, será muy útil que pueda ir generando metadatos, conforme se vaya realizando el proceso ETL. Por último si no se dispone de funcionalidades de tarea programada, integrado en la Base de Datos, se debe analizar la manera de crear JOB con tareas programadas para automatizar la tarea de carga del DW.

10) Tomar ventaja de las fuentes externas

La integración de fuentes externas a los sistemas operacionales, como puede ser la información de encuestas de satisfacción de los clientes o los estudios de mercado de terceros, o información sobre competidores, puede aportar un valor añadido muy importante al DW.

Esta información nos permitirá sacar conclusiones mucho más avanzadas sobre el negocio, que las meramente internas como ventas, costes, etc....

11) Utilizar nuevos métodos de distribución de la información.

Antiguamente, se necesitaba de la participación de analistas que prepararan los informes para cada necesidad de los usuarios. Sin embargo, ahora se pueden utilizar informes parametrizables, envíos vía e-mail, alertas, etc.... de modo que son los usuarios finales los que acceden directamente a la información que necesitan y pueden configurarse sus propias consultas. Las herramientas actuales permiten que usuarios con poca experiencia armen sus propios reportes (Ejemplo Tableau o Microsoft Power BI).

12) Considerar el Outsourcing para el desarrollo y mantenimiento del DW

Muchas compañías de mediano y gran tamaño utilizan el outsourcing como medio de garantizar el complejo, largo y costoso proceso de poner en funcionamiento un DW y evitan la dificultad de encontrar y retener profesional IT capacitados.

El outsourcing puede llegar a generar nuevas ideas y desarrollos en base a su conocimiento profundo del DW y de su arquitectura, además no tiene los problemas de falta de personal capacitado de muchas empresas.

Consideraciones finales:

Siempre hay que considerar que el diseño e implementación de un DW es incremental, y cuando se inicia el proceso se puede optar por trabajar primero por el área de mayor interés por los Directivos o Gerente Gral., o se puede optar por iniciar por el área más simple que no requiere análisis complejos para dar resultados rápidos.

5 ETL– (EXTRACT - TRANSFORM - LOAD)

Una vez terminado el diseño e implementado el DW es momento de poblar las tablas que la componen.

Como recomendación antes de comenzar armar los procesos ETL es necesario evaluar las fuentes, tamaño de las tablas y archivos, procesos a realizar (principalmente transformaciones). Una vez evaluado los puntos mencionados es necesario determinar si algunos de los procesos que realizará el ETL no es conveniente hacerlo en el ámbito de base de datos y trabajar con tablas intermedias (Tablas de temporales físicas o verdaderas temporales).

Esto es porque en algunas ocasiones realizar el mismo proceso de inicio (captura de datos) en un ETL es más costoso (lleva más tiempo y recursos) en término de proceso que hacerlo dentro del ámbito de la base de Datos de origen.

¿Cuándo es necesario trabajar con tablas INTERMEDIAS?

- Cuando las fuentes necesitan el cruce de varias tablas grandes (SQL con inner join, etc.).

Consultas SQL complejas para obtener una fuente de inicio (donde comenzará los procesos ETL). En ocasiones se requiere consultas donde se manejan operaciones de conjunto, etc. Además, obtienen una gran cantidad de registros de las consultas.

Operaciones de actualización masiva que en SQL son mucho más efectiva que hacerlo en un ETL. Sobre todo, procesos de actualización (UPDATE) y más aún con actualizaciones condicionadas.

Si por algunos de estos motivos es necesario trabajar con tablas intermedias lo que se procede es a la creación de estas tablas (con mismo formato) en las Bases fuentes (de origen) y en la Base donde se encuentra el DW. Luego de procesar en la Base fuente y cargar las tablas intermedias se procede a dar inicio (captura de datos) de los procesos ETL, donde su primera actividad será copiar de la tabla intermedia de la Base fuente hacia la tabla intermedia de la Base del DW. A partir de allí el ETL realizará sus procesos definidos.

Nota: se aclara, que todos los pasos intermedios para poblar las tablas intermedia en la Base fuente se puede obviar y procesarla el 100 % dentro de los pasos iniciales del ETL, solo que los tiempos de procesamiento serán mayores.

TABLAS INTERMEDIAS DE FACT Y DIM

Al crear un proceso ETL nunca se debe cargar directamente de una fuente hacia las tablas principales del DW, las tablas FACT y DIM. Se debe cargar tablas intermedias.

El motivo de cargar tablas intermedias y no directamente las tablas operativas son: ante un error de proceso, se puede desprocesar (borrar todos los datos de las intermedias) sin problema y no afectar la consistencia del DW.

Solo cuando todas las tablas intermedias fueron cargadas correctamente se debería proceder actualizar (insertar los nuevos registros) el DW.

Se suele agregar un prefijo a las tablas, por ejemplo, las siglas: int_fact_xxx o int_dim_xxx.

PASOS PARA CREAR UN ETL

Los pasos para crear procesos ETL son los siguientes:

- 1) Análisis de todas las fuentes intervinientes.
- 2) Análisis de conveniencia de realizar cálculos y procesos intermedios en la Base fuente. El resultado cargar en tablas intermedias.
- 3) Definición de las conexiones necesarias para los procesos ETL.
- 4) Creación y carga de tablas de auditoría. Estas tablas son necesarias para tener un seguimiento de los procesos ETL ejecutados, parámetros enviados (nombre y valor del parámetro), que persona(usuario) los ejecutó, resultado de las ejecuciones, errores encontrados. Esto es necesario para un control de si los procesos se ejecutaron normalmente o si se debe hacer un reproceso.
- 5) Diseño e implementación de los procesos ETL.
- 6) Pruebas y ajustes de los procesos ETL. Se debe probar y garantizar los procesos de contingencia que ante un error se desprocese todo lo cargado y que el DW queda en un estado consistente.
- 7) Automatización de la ejecución de los procesos ETL. Dependiendo de las bases donde se encuentra el DW este paso de automatización se realiza todo dentro de la misma base como un JOB o tarea programada. En otros casos se debe crear la tarea programada (por sistema operativo).
- 8) Puesta operativa de los procesos ETL.
- 9) Seguimiento, Control y ajustes necesarios de los ETL operativos.

QUE COMPRENDE UN PROCESO ETL

Un proceso ETL comprende lo siguiente:

Extract (Extracción)

- Extracción o captura de datos es el primer paso.
- Comprende la lectura de los datos provenientes de las fuentes de datos.
- Si los datos provienen de Base de datos se deberá crear conexiones a cada Base.
- Si los datos provienen de Archivos (Excel, CSV, Texto) se deberá generar un mecanismo de INPUT (lectura del archivo).

Transform (transformación)

- Los datos que provienen de las fuentes pasan por una serie de transformaciones antes de cargarse en las tablas definitivas del DW. Esto es porque son estructuras diferentes y con

objetivos diferentes. En el DW las tablas se encuentran resumidas y preparadas para toma de decisiones.

- Las transformaciones normalmente involucran: Cálculos, Limpieza de datos, Transformaciones de formato, Aplicaciones de Reglas de validación, transposición de filas y columnas, Filtrado de datos, Integraciones de datos, etc.

Load (Carga)

- El último paso de un ETL es la carga del DW.
- Es recomendable que todo el proceso cargue primero en tablas staging para las fact y dim con un prefijo para determinar que son intermedias, por ejemplo, pueden ser stg_fact_XXX o stg_dim_XXX. Las tablas intermedias de volcado(staging) deberán tener la misma estructura (igual cantidad de campos y el mismo tipo de dato para cada campo). De esta manera se garantiza de que terminado el proceso completo del ETL se procede a la carga completa del DW, si algún paso dentro del proceso general del ETL genera un error se desprocesa por completo la carga de cada una de las tablas intermedias sin perjudicar la consistencia del DW.

Secuencia de carga de un DW

- 1) Primero se cargan las tablas intermedias DIM. Se procesa las transformaciones necesarias.
- 2) Si todas las tablas intermedias DIM fueron exitosas, se cargan las tablas intermedias FACT y se procesa las transformaciones necesarias.
- 3) Si todas las intermedias DIM y FACT fueron exitosas se procede a la carga definitiva de las tablas DIM y FACT del DW.

Si algún paso genera un error se debe desprocesar todo lo cargado por el ETL y dejar en estado consistente el DW. Es necesario manejar log de errores para saber que sucedió, corregir y volver a procesar.