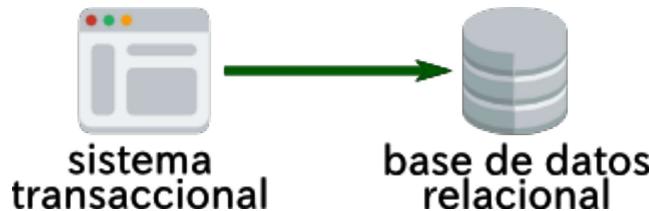


Capítulo 1: Business Intelligence

- Datos como producto secundario
 - Definiendo al BI
 - Aplicando BI
 - Ambito de aplicación
 - Orígenes del BI
 - Proceso BI
 - Beneficios
-

Datos como producto secundario

En toda organización se generan datos constantemente, para la ejecución de sus operaciones y transacciones. Es muy común, que estos datos se administren a través de sistemas transaccionales y se almacenen en bases de datos relacionales, aunque esto no es excluyente.



Al pasar de los años, la acumulación de estos datos NO produce utilidad alguna, o lo hace de manera poco relevante a través de esporádicas consultas históricas. HEFESTO tiene como propósito poner en evidencia que esa acumulación deje de ser eso, datos acumulados, para comenzar a jugar un papel mucho más importante, y pueda constituirse en un increíble valor agregado. De allí, nos preguntamos:

- ¿Qué pasaría si de alguna manera procesamos todos estos datos y los utilizamos como fuente de información para la toma de decisiones?
- ¿Qué pasaría si además contamos con herramientas de software especializadas en la presentación de los datos para el estudio analítico?

El Business Intelligence (BI - Inteligencia de Negocios) es quien brindará la solución a nuestros interrogantes, en pos de mejorar el proceso de toma de decisiones.

Definiendo al BI

Se puede describir el Business Intelligence (BI - Inteligencia de Negocios), como la actividad de:

- almacenar y procesar grandes cantidades de datos,
- para que mediante la utilización de herramientas de software especializadas,
- sea sencillo el análisis y exploración de dichos datos,
- con el principal objetivo de obtener conocimiento (knowledge) orientado a tomar decisiones en tiempo real.



Este conocimiento debe ser:

- oportuno,
- relevante,
- útil y
- adaptado al contexto de la organización.

Existe una frase muy popular acerca de BI, que dice:

Inteligencia de Negocios es el proceso de convertir datos en conocimiento; el conocimiento fundamenta decisiones y éstas orienten acciones pertinentes y eficaces según los propósitos asumidos por la organización.

Aplicando BI

Cuando aplicamos BI a una organización, la fuente de datos (data source) principal es la que conforman los datos que la organización ha generado; con esta base analizaremos su comportamiento a lo largo del tiempo, desde diferentes escenarios y puntos de vista.

Una de las premisas del BI es que los objetivos de la organización se traduzcan en Indicadores de Estudio, por ejemplo si se trata de una entidad comercial: cantidad vendida, importe pagado, etc. Estos indicadores serán analizados aplicándoles diversos criterios, por ejemplo: año de venta, proveedor, cliente, rubro, etc; el conjunto de estos criterios conformarán las Perspectivas de Análisis.

Podemos decir entonces, que el BI nos permitirá analizar los Indicadores desde diferentes Perspectivas, y mediante ello responder preguntas sobre:

- lo que está sucediendo en la organización,
- lo que ha sucedido,
- lo que puede llegar a suceder y
- por qué.



Precisamente, la inteligencia de negocios permite que el proceso de toma de decisiones esté fundamentado sobre un amplio conocimiento de los procesos internos de la organización y del entorno, minimizando de esta manera el riesgo y la incertidumbre.

Ambito de aplicación

Las soluciones BI, NO necesariamente son aplicables a grandes y/o medianas empresas, como sí lo fue en sus inicios.

En la actualidad, puede aplicarse BI a cualquier organización, sin importar su tamaño y complejidad. Esto se debe principalmente a dos factores:

- el desarrollo de suites BI con licencia Free Software / Open Source, y
- la proliferación en internet de documentación, comunidades, foros, wikis, tutoriales y cursos.



En sus orígenes, el BI se encontraba orientado y acotado, fundamentalmente, a resolver problemáticas relacionadas al aumento de la rentabilidad, la disminución de costos y la obtención de ventajas competitivas de las organizaciones empresariales. Se trataba de una herramienta aplicada casi con exclusividad al campo económico, y su implementación y despliegue eran muy costosas.

Con el tiempo, sus finalidades y aplicación fueron ampliando y diversificándose, de modo tal que se constituyó en una herramienta para distintos campos sociales, y no circunscrita a lo económico.

Esta evolución permitió que el BI pueda ser aplicado a una biblioteca popular, a un centro vecinal o cualquier institución u organización que necesite tomar decisiones pertinentes en relación a los datos que produce.

Veamos entonces dos ejemplos de aplicación de BI:

1) Empresa de venta de productos: en este caso la aplicación de BI podrá resolver las siguientes preguntas.

- ¿Quiénes son l@s mejores client@s?
- ¿Cómo minimizar costos y maximizar las utilidades?
- ¿Cuál será el pronóstico de ventas del próximo mes?
- ¿Cuáles son los productos más vendidos por estación?
- ¿Han mejorado las ventas respecto del mismo período de un año anterior?

2) Biblioteca vecinal: en este caso la aplicación de BI podrá resolver las siguientes preguntas.

- ¿Cuál es la temática más consultada?
- ¿Qué días hay mayor concurrencia, y por qué?
- ¿Qué libros deben ser adquiridos?
- ¿Cuál es el rango etario que más lee cómics?
- ¿Qué perfil tienen las personas que leen determinada temática?

Orígenes del BI

El Business Intelligence tiene sus raíces en:

- los Executive Information Systems (EIS – Sistemas de Información Ejecutiva) y en
- los Decision Support Systems (DSS – Sistemas para la Toma de Decisiones).

Executive Information Systems

El EIS proporciona medios sencillos para consultar, analizar y acceder a la información del estado del negocio.

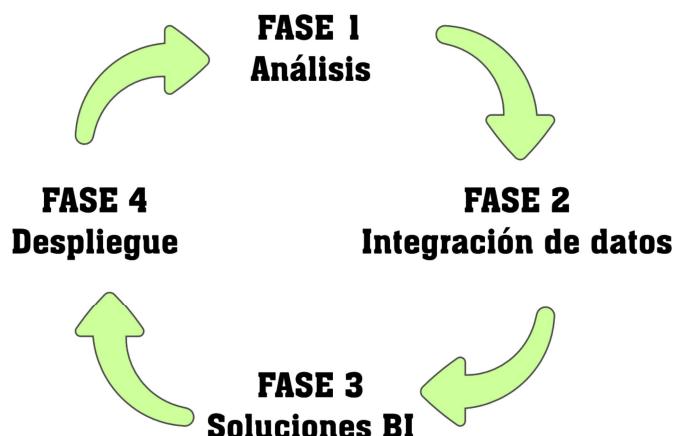
Decision Support Systems

Los DSS son una clase especial de sistemas de información cuyo objetivo es analizar datos de diferentes procedencias y brindar soporte para la toma de decisiones.

Proceso BI

El proceso mediante el cual una organización obtiene rédito de sus datos (los utiliza de forma inteligente) se denomina Proceso BI.

A continuación se enumeran sus fases:



- **FASE 1: Análisis.** Esta es la fase inicial en la que se recolectarán los requerimientos de información de l@s usuari@s. El foco estará puesto en conocer las necesidades de información de la organización, a partir de las cuales se redactarán las preguntas, cuyas respuestas ayudarán a tomar decisiones en pos de alcanzar los objetivos.
- **FASE 2: Integración de datos.** Esta fase tiene como fin extraer e integrar datos de diferentes fuentes (data sources). Las fuentes pueden ser de origen interno y/o externo según la necesidad. Las mismas producirán el flujo de datos que será el fundamento con el cual se encontrarán las respuestas a las preguntas planteadas en el paso anterior. Para la integración de los datos de las diversas fuentes, se les aplicará transformaciones a fin de compatibilizarlos con los requerimientos del análisis, y posteriormente se cargarán en la base de datos final, cuyo propósito es servir a BI.
- **FASE 3: Soluciones BI.** En esta fase se utilizarán herramientas, técnicas y componentes que permitirán la explotación de los datos. El objetivo será producir soluciones BI para que l@s usuari@s obtengan respuestas a las preguntas ya planteadas. Algunos de los componentes más utilizados son: reportes, indicadores, análisis interactivos, dashboards, gráficos estadísticos, etc.
- **FASE 4: Despliegue.** En esta fase se les entregará a l@s usuari@s los componentes BI que le correspondan y se les capacitará en su utilización, como así también en la creación de sus propias soluciones BI. L@s usuari@s obtendrán respuestas a sus preguntas e identificarán nuevas preguntas, cuyas respuestas requerirán que se inicie una nuevo ciclo del Proceso BI.

Beneficios

Entre los beneficios más importantes que BI proporciona a las organizaciones, se destacan los siguientes:

- Reduce el tiempo mínimo que se requiere para recoger todos los datos relevantes de un tema en particular, ya que los mismos se encontrarán integrados en una fuente única de fácil acceso.
- Automatiza la actualización de datos en la fuente integrada, a través de procesos predefinidos.
- Proporciona herramientas de análisis que servirán para establecer comparaciones y explorar los datos, con lo cual se mejorará notablemente la toma de decisiones.
- Completa el círculo que hace pasar de la decisión a la acción.
- Muchos análisis y reportes serán dinámicos y/o definidos por el usuari@ en el momento. Esto independizará a l@s usuari@s de los tradicionales informes pre-programados.
- Permite dar respuesta a preguntas de forma inmediata. También ayuda a la formulación de nuevas preguntas cuyas respuestas son clave para mejorar el desempeño de la organización.
- Permite acceder, analizar y monitorear directamente los Indicadores críticos de la organización.
- Identifica cuáles son los factores que inciden en el buen o mal funcionamiento de la organización.
- Se pueden detectar situaciones fuera de lo normal o potencialmente fuera de curso.
- Permite predecir el comportamiento futuro con un alto porcentaje de certeza, basado en el entendimiento del pasado.
- L@s usuari@s podrán consultar y analizar los datos de manera sencilla e intuitiva.

Capítulo 2: Data Warehousing & Data Warehouse

- Introducción
 - Data Warehousing
 - DWH vs DW
 - Data Warehouse:
 - ▶ Orientada al negocio
 - ▶ Integrada
 - ▶ Variante en el tiempo
 - ▶ No volátil
 - Cualidades del DW
 - Riesgos de aplicación
 - Redundancia
-

Introducción

La aplicación de una solución BI requiere integrar datos internos de la organización con datos externos, procesarlos, y luego almacenarlos en una base de datos que posibilite la accesibilidad y el posterior análisis.

Esto implica la necesidad de contar con un proceso que contemple y defina todas estas tareas. A este proceso se lo denomina Data Warehousing.

Data Warehousing

El Data Warehousing (DWH), es el proceso que reúne y ordena las tareas inherentes a:

- la extracción, transformación, consolidación, integración y
- centralización de los datos internos (datos que una organización genera en su actividad diaria, por ejemplo compras, ventas, producción, etc-) y los datos externos relacionados;
- permitiendo de esta manera el acceso, análisis y exploración,
- con el objetivo de dar soporte al proceso de toma de decisiones estratégico y táctico.

Para almacenar los datos necesarios para el análisis, se empleará un *Data Warehouse (DW)*.

Un Data Warehouse es una base de datos que posee una estructura multidimensional.

DWH vs DW

DWH (Data Warehousing) es un proceso que emplea un DW.

DW (Data Warehouse) es una base de datos con estructura multidimensional.

Pese a que las letras de su abreviatura son muy parecidas, y ello puede llevar a confusiones, es fundamental distinguir lo que es un proceso de lo que es una base de datos.



Data Warehousing (DWH)
es un proceso



Data Warehouse (DW)
es una base de datos

Data Warehouse

Una de las definiciones más famosas sobre DW, es la de William Harvey Inmon, quien es reconocido mundialmente como el padre del DW:

- Un Data Warehouse es una colección de datos orientada al negocio, integrada, variante en el tiempo y no volátil para el soporte del proceso de toma de decisiones de la gerencia.



Nota

Los términos almacén de datos y depósito de datos, son análogos a DW, y se utilizarán de aquí en adelante para referirse al mismo.

Orientada al negocio



Esto significa que:

- al DW solo ingresarán datos relevantes para el análisis y toma de decisiones. Por ejemplo, los datos referidos a los clientes, como su dirección de correo electrónico, fax, teléfono, código de identidad, código postal, etc., que son tan importantes de almacenar en cualquier sistema transaccional, NO son tenidos en cuenta para el DW por carecer de valor analítico, pero sí lo son aquellos que indican el tipo de cliente, su clasificación, ubicación geográfica, edad, etc.
- se manejarán entidades de alto nivel, es decir si una empresa maneja stock, listas de precios, cuentas corrientes, ventas, compras, etc., con lo que trabajaremos en el DW serán con entidades del tipo: clientes, productos, rubros, zonas, etc.
- la estructura será multidimensional, es decir que almacenará sus datos en tablas de Hechos y tablas de Dimensión.

Integrada



La integración implica que todos los datos provenientes de orígenes heterogéneos deben ser analizados a fin de asegurar su calidad y limpieza para luego ser consolidados en el DW. El proceso que permite esta consolidación, se denomina Integración de Datos, y cuenta con diversas técnicas y subprocesos para llevar a cabo sus tareas. Una de estas técnicas es el proceso ETL: Extracción, Transformación y Carga de datos (Extraction, Transformation and Load).

A continuación se describirán los orígenes de datos más comunes:

- Producidas por tipos de usuari@s:
 - ▶ Operacional: produce datos diariamente, en gran cantidad, muchos de los cuales son poco relevantes para el análisis por sí mismos, y la granularidad de estos datos es muy fina. Por ejemplo cuando se registran ventas de productos.
 - ▶ Medio: utiliza los datos operacionales para producir otros datos nuevos que tienen implicancia a corto/medio plazo. Por ejemplo, un control de stock, requerirá una o varias compras a fin de realizar el abastecimiento.
 - ▶ Gerencial: utiliza datos altamente procesados. En general, este perfil de usuari@ será el destinatario del DW, siempre produce una retroalimentación que permite generar nueva información para el análisis.
- Producidas por áreas o departamentos de la organización:
 - ▶ Las organizaciones se subdividen en áreas con responsabilidades bien definidas y diferentes. Cada una de estas áreas produce sus propios datos; éstos serán compartidos con otras áreas.
 - ▶ Las subdivisiones suelen ser geográficas, este tipo de división produce nuevos datos que deben ser incorporados. Por ejemplo: localización, cotizaciones, tipos de cambio, etc.
- Producidas por diferentes data sources:
 - ▶ Fuentes internas: datos que genera la empresa en sus actividades diarias.
 - ▶ Fuentes externas: datos que complementan y suplementan los datos internos. Por ejemplo: datos climáticos, análisis de tendencia, estadísticas, censos, etc.

Variante en el tiempo



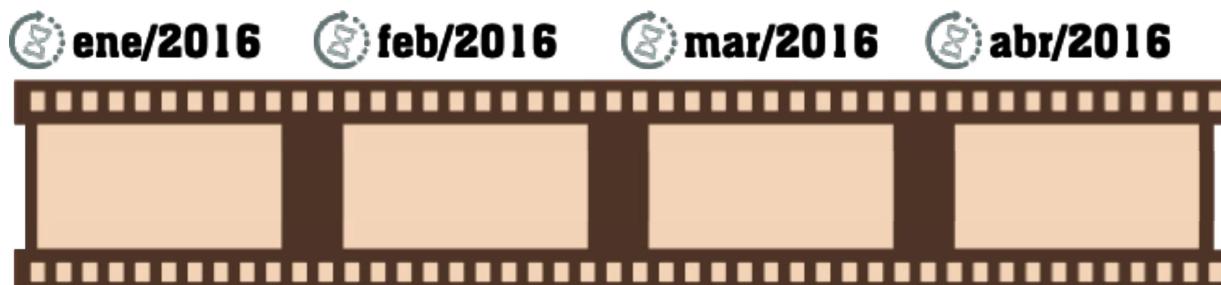
En el DW los datos actuales son almacenados junto a los datos históricos, y cada dato es marcado con su correspondiente sello de tiempo (timestamps).

Mediante este sello de tiempo se podrá tener acceso a diferentes versiones de una misma situación.

Es decir que se podrán observar los datos como si se tratase de una película de rollo, en donde cada fotograma posee:

- una escena en particular (situación de la empresa a analizar) y
- su sello de tiempo correspondiente (fecha/hora en que sucedió).

De esta manera, utilizando el sello de tiempo se podrán avanzar y retroceder los fotogramas del rollo, manteniendo el foco de atención sobre la situación analizada.



Es importante tener en cuenta y planificar la granularidad con que se almacenarán los datos.

No volátil



La información solamente será útil para el análisis y la toma de decisiones siempre y cuando ésta sea estable.

La naturaleza de los sistemas operacionales hacen que los datos que éstos administran varíen de forma permanente. Esto NO ocurre en los DW, ya que una vez que los datos ingresan NO cambian.

En un ambiente operacional son habituales las acciones de actualización (insertar, eliminar y modificar) y consulta; en cambio, en el DW la manipulación de los datos es más simple, solo existen dos tipos de acciones:

- Insertar: esta acción la realizan de forma programada los procesos de Integración de Datos.
- Consultar: esta es la única acción que l@s usuari@s pueden realizar sobre los datos.



DATA SOURCES

Consultar

Insertar

Eliminar

Modificar



DATA WAREHOUSE

Consultar

Insertar

Cualidades del DW

Las siguientes, son las cualidades más significativas del DW:

- Maneja un gran volumen de datos, debido a que integra los datos recolectados durante años, proveniente de diversos orígenes y fuentes, en una sola base de datos centralizada.
- Almacena datos agregados, actuales e históricos.
- Estructura los datos de forma multidimensional.

Riesgos de aplicación

Los riesgos más comunes que se pueden presentar en la implementación de un proceso de Data Warehousing son:

- Requiere una gran inversión, que en muchos casos se subestima. El proceso de DWH implica un amplio abanico de tareas que NO son sencillas de llevar a cabo y que requieren costos específicos (horas de desarrollo, horas de testing, horas de implementación, adquisición de hardware/software, capacitación de l@s usuari@s, etc.).
- Existe resistencia al cambio por parte de l@s usuari@s. La mejor forma de abordar este inconveniente es mediante una correcta capacitación.
- Los beneficios de su implementación NO se apreciarán en el corto plazo, pero sí lo harán en el mediano y largo plazo. Este punto deriva del anterior, y básicamente se refiere a que NO tod@s l@s usuari@s confiarán en el DW en una primera instancia, pero sí lo harán una vez que comprueben su efectividad y ventajas. Además, su correcta utilización surge de la propia experiencia y la capacitación obtenida.
- La manipulación de datos, atentará contra la privacidad de los datos sensibles o confidenciales, por ejemplo: listados de client@s, direcciones, contratos, datos médicos, etc. En este punto es imprescindible definir un esquema de roles que asegure la privacidad de acceso; sin embargo l@s desarrollador@s que se encarguen de los procesos de Integración de Datos, muy probablemente, tendrán acceso a datos sensibles o confidenciales.
- La subvaloración de los recursos necesarios para las soluciones de Integración de datos. Los procesos de integración de datos son sumamente complejos y son en definitiva la base en la que descansan el resto de los procesos.
- La subvaloración del esfuerzo necesario para un diseño correcto y en donde se tengan en cuenta todos los requerimientos del negocio.
- El incremento continuo de los requerimientos de l@s usuari@s. Una vez que el DW comienza a ser utilizado, l@s usuari@s tendrán una nueva forma de ver la información y en consecuencia una explosión de nuevas ideas, que se traducen en requerimientos para cubrir nuevas necesidades.
- La subestimación de las capacidades que puede brindar la correcta utilización del DWH y de las herramientas de BI en general. Una única capacitación NO es suficiente, sobre todo si esta instancia inicial suele ser una capacitación técnica del uso de herramientas. Éstas NO serán utilizadas en todo su potencial si NO se brindan capacitaciones tendientes a que l@s usuari@s conozcan conceptos de BI que devuelven esta potencialidad.

Redundancia

Debido a que el DW recibe información histórica de diferentes orígenes y fuentes, suele deducirse de forma errónea, que existe una repetición masiva de datos entre el ambiente DW y el operacional. Este razonamiento es superficial y por lo tanto erróneo; en realidad existe una mínima redundancia de datos entre ambos ambientes.

Para entender claramente lo antes expuesto, se debe considerar lo siguiente:

- Del ambiente operacional solamente se seleccionan datos relevantes, los cuales se codifican, filtran y transforman antes de ingresar al DW.
- El horizonte de tiempo, nivel de granularidad y agregación, son muy diferentes entre los dos ambientes.
- Algunos estudios indican que la redundancia encontrada al cotejar los datos de ambos ambientes es mínima, y que generalmente resulta en un porcentaje menor del 1%.

Capítulo 3: Arquitectura Data Warehousing

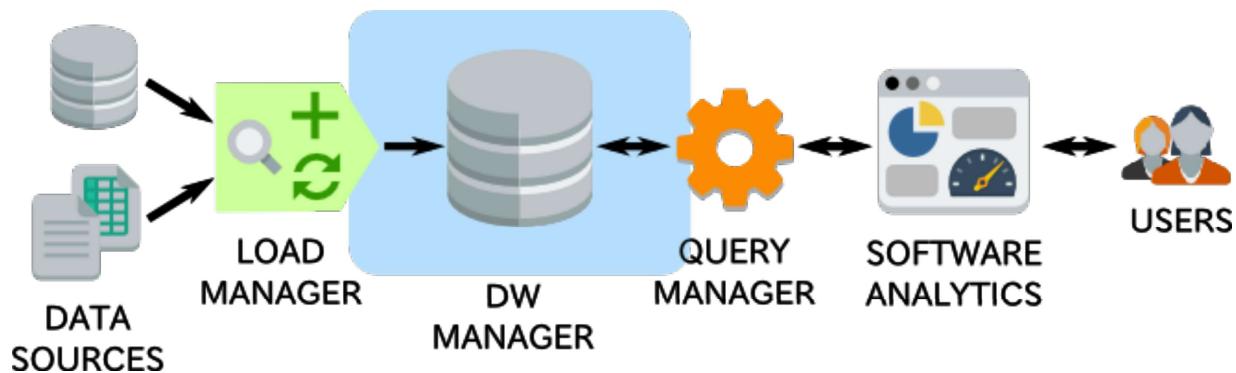
- Introducción
- 1) Data Sources
- 2) Load Manager
 - ▶ Extracción
 - ▶ Transformación
 - ▶ Codificación
 - ▶ Medida de Atributos
 - ▶ Fuentes múltiples
 - ▶ Data Cleaning
 - ▶ Carga
 - ▶ Proceso ETL
- 3) Data Warehouse Manager
 - ▶ Base de datos multidimensional
 - ▶ Tablas de Dimensiones
 - ▶ Tiempo
 - ▶ Tablas de Hechos
 - ▶ Agregadas y Preagregadas
 - ▶ Cubo Multidimensional: introducción
 - ▶ Representación matricial
 - ▶ Modelos del DW
 - ▶ Estrella
 - ▶ Copo de Nieve
 - ▶ Constelación
 - ▶ OLTP vs DW
 - ▶ Implementación
 - ▶ ROLAP
 - ▶ MOLAP
 - ▶ HOLAP
 - ▶ Cubo Multidimensional: profundización
 - ▶ Jerarquías
 - ▶ Metadatos
- 4) Query Manager
 - ▶ Drill-up
 - ▶ Drill-down
 - ▶ Drill-across
 - ▶ Roll-across
 - ▶ Pivot
 - ▶ Page

- ▶ Drill-through
 - 5) Software Analytics
 - ▶ Interacción
 - ▶ Características
 - ▶ Reporting
 - ▶ OLAP
 - ▶ Dashboards
 - ▶ Data Mining
 - ▶ Redes Neuronales
 - ▶ Sistemas Expertos
 - ▶ Programación Genética
 - ▶ Árboles de Decisión
 - ▶ Detección de Desviación
 - ▶ EIS
 - 6) Users
-

Introducción

Teniendo en cuenta las características que definen el Data Warehousing y que se han desarrollado en los apartados anteriores, en este capítulo se describirán los componentes de su arquitectura. .

En el siguiente gráfico se pueden visualizar los diferentes componentes de la estructura del DWH:



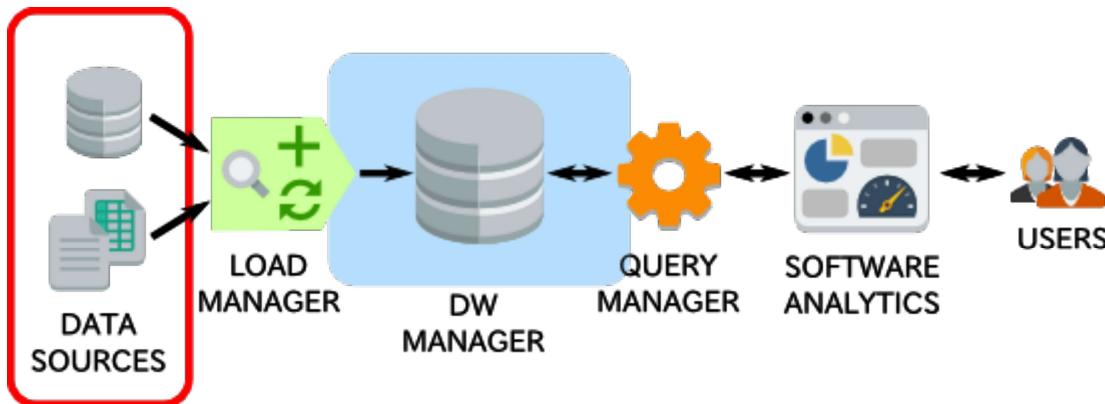
Como se puede apreciar, el ambiente del DWH está formado por diversos componentes que interactúan entre sí y cumplen una función específica dentro del sistema.

Su funcionamiento puede resumirse de la siguiente manera:

- Los datos son extraídos desde distintas fuentes, bases de datos, archivos, servicios web, etc. Estos datos, generalmente, residen en diferentes tipos de sistemas con arquitecturas y formatos variados.
- Los datos son integrados, transformados y limpiados, para luego ser cargados en el DW.
- Con base en el DW, se construirán Cubos Multidimensionales y/o Business Models.
- Los usuarios accederán a los Cubos Multidimensionales, Business Models (u otro tipo de estructura de datos) del DW, utilizando diversas herramientas de consulta, exploración, análisis, reportes, etc.

A continuación se enumerarán y explicarán los componentes de la arquitectura del Data Warehousing, utilizando como guía el gráfico antes expuesto.

1) Data Sources



Los Data Sources (orígenes de datos) representan los datos transaccionales que genera la empresa en su accionar diario, junto a otros datos internos y/o externos complementarios.

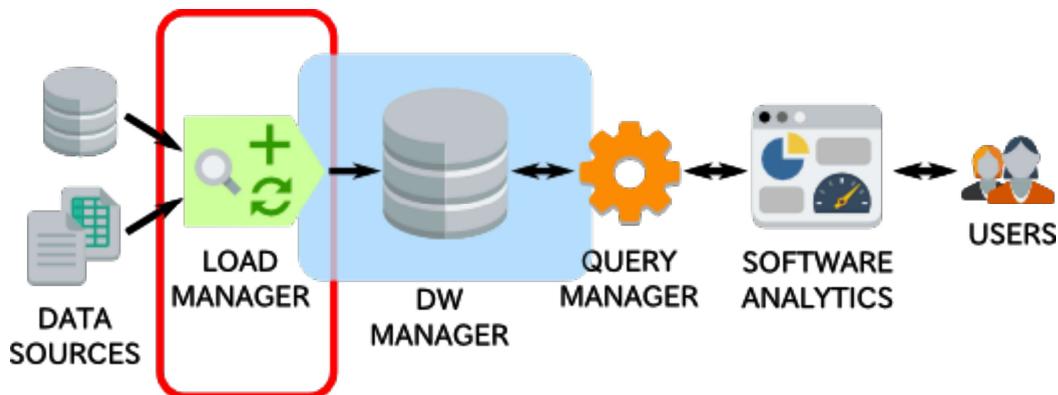
Los Data Sources poseen características muy disímiles entre sí, en formato, procedencia, función, etc. En la actualidad esto se ve potenciado gracias a los web services, redes sociales, y a la utilización cada vez más frecuente de bases de datos NoSQL.



Los Data Sources más habituales para extraer datos relevantes son:

- Archivos de textos
- Hojas de cálculos
- Informes semanales, mensuales, anuales, etc.
- Bases de datos transaccionales, SQL y *NoSQL*
- Información no estructurada (páginas web, mails)
- Redes sociales
- RSS
- Web Services

2) Load Manager



El componente Load Manager es el encargado de la ejecución y calendarización (scheduling) de los diferentes procesos de Integración de Datos a través de los cuales:

- se extraerán los datos desde los Data Sources,
- serán manipulados, integrados y transformados, para luego
- cargar los resultados obtenidos en el DW.

La Integración de Datos es una serie de técnicas y procesos que se encargan de llevar a cabo todas las tareas relacionadas con la extracción, manipulación, control, integración, depuración de datos, carga y actualización del DW, etc. Es decir, todas las tareas que se realizarán desde que se obtienen los datos de los diferentes Data Sources hasta que se cargan en el DW.

Si bien el proceso ETL (Extraction, Transformation, Load) es solo una de las muchas técnicas de la Integración de Datos, es la más importante, incluso en muchos casos constituyen el proceso de integración en si. En este orden, se puede ubicar el resto de las técnicas en las diferentes etapas del ETL:

- el proceso Extracción incluirá técnicas enfocadas por ejemplo a obtener desde diversas fuentes solamente los datos relevantes y mantenerlos en una *Staging Area* (almacenamiento intermedio);
- el proceso Transformación incluirá técnicas encargadas de compatibilizar formatos, filtrar y clasificar datos, relacionar diversas fuentes, etc;
- el proceso Carga incluirá técnicas propias de la carga de datos y actualización del DW.

A continuación, se detallarán cada una de estas etapas, se expondrá cuál es el proceso que llevan a cabo los ETL y se enumerarán cuáles son sus principales tareas.

Extracción

La selección de los Data Sources para proveer todos los datos que sean relevantes, tiene que hacerse teniendo en cuenta las necesidades de l@s usuari@s y requisitos definidos para la solución.

En la mayoría de los casos, los Data Sources a utilizar serán bases de datos relacionales, con lo cual la extracción puede llevarse a cabo mediante consultas SQL o procedimientos almacenados. Pero en el caso de Data Sources NO convencionales o NO estructurados, la obtención será más difícil.

Una vez seleccionados y extraídos, los datos, deben ser persistidos en una base de datos relacional Staging (almacenamiento intermedio), lo cual permitirá:

- Manipular los datos sin interrumpir ni sobrecargar los Data Sources y el DW.
- Crear una capa de abstracción entre la lectura y la carga.
- Almacenar y gestionar los metadatos que se generan en los procesos ETL.
- Facilitar la integración.

El Staging Area, generalmente, se constituye en una o más bases de datos relacionales donde la información puede ser persistida en tablas auxiliares, tablas temporales, etc. Una vez que los datos se encuentren en Staging Area, el proceso puede desconectarse de los Data Sources y continuar con la transformaciones necesarias para poblar el DW.

Transformación

Esta función es la responsable de aplicar todas las acciones necesarias sobre los datos, a fin de hacerlos consistentes, compatibles y congruentes con el DW.

Los casos más comunes en los que se debe realizar integración, son los siguientes:

- Codificación.
- Medida de Atributos.
- Fuentes múltiples.

Además de lo antes mencionado, esta función se encarga de realizar los procesos:

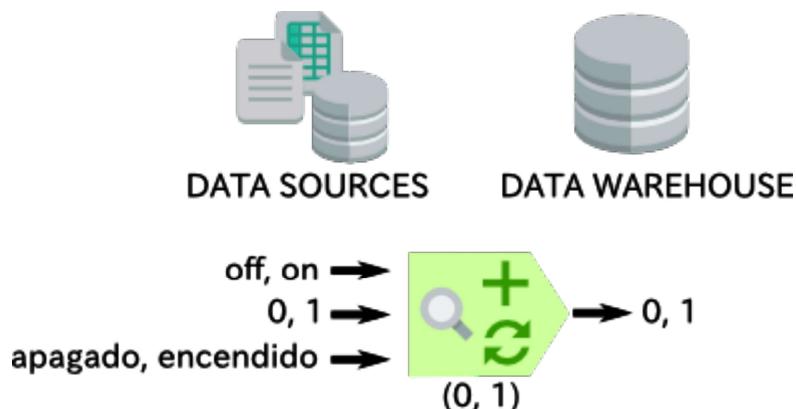
- Data Cleansing (Limpieza de Datos).
- Data Quality (Calidad de Datos).

Codificación

Una inconsistencia típica que se encuentra al integrar varios Data Sources, es la de contar con más de una forma de codificar un Atributo en común. Por ejemplo, al especificar el valor que tendrá el campo **status**, en un Data Source se pueden encontrar valores **0** y **1**, en otros **Apagado** y **Encendido**, en otros **off** y **on**, etc.

Se deberá seleccionar una de las codificaciones existentes o bien aplicar otra, pero todos los datos de ese Atributo deben respetar una única convención de codificación antes de ingresar al DW.

En la siguiente figura, se puede apreciar que de un conjunto de diferentes formas de codificar se escoge solo una. Cuando se obtiene un dato con una codificación que difiere de la seleccionada, es cuando se procede a su transformación.

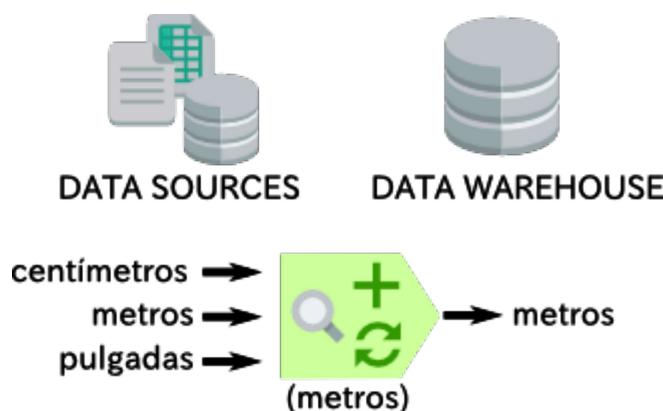


Medida de Atributos

Los tipos de unidades de medidas utilizados para representar los Atributos, pueden variar en cada Data Source. Por ejemplo, al registrar la longitud de un producto determinado, las unidades de medidas pueden ser expresadas en centímetros, metros, pulgadas, etc.

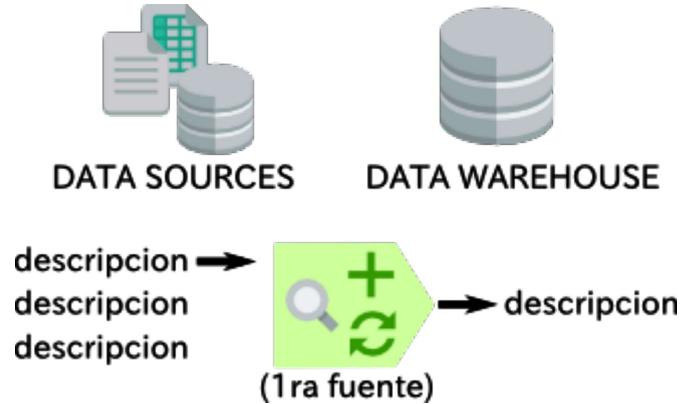
Este proceso implica redimensionar el valor asociado a la unidad de medida. Este redimensionamiento suele ejecutarse multiplicando el valor por un factor de escala. Por ejemplo si la unidad de medida seleccionada es metro y desde un Data Source vienen centímetros, los valores se deben multiplicar por el factor de escala **0,01** a fin de que se mantengan íntegros.

En la siguiente figura, se puede apreciar que de diferentes unidades de medida se escoge solo una. Cuando surge un dato con una unidad de medida que difiere de la seleccionada, es cuando se procede a su transformación.



Fuentes múltiples

Un mismo elemento puede derivarse desde varios Data Sources. En estos casos, se debe elegir aquella fuente que se considere más fiable y apropiada.



Data Cleansing

El objetivo principal del Data Cleansing (Limpieza de Datos) es entregar un flujo de datos lo más sanitizado posible. Por lo tanto podrá llevar a cabo acciones tales como eliminar datos erróneos o irrelevantes y subsanar aquellos que presenten inconsistencias.

Las acciones más típicas que se pueden llevar a cabo al encontrarse con Outliers (Datos Anómalos) son:

- Ignorarlos.
- Eliminar la columna (se eliminan el 100% de los datos).
- Filtrar la columna (se eliminan algunos de los datos).
- Filtrar la fila errónea, ya que a veces su origen, se debe a casos especiales.
- Reemplazar el valor.
- Discretizar los valores de las columnas. Por ejemplo, si un campo numérico presenta un valor de **1** a **2**, utilizamos el texto **Bajo**; de **3** a **7**, **Óptimo**; de **8** a **10**, **Alto**. Entonces, cuando suceda un Outlier se puede reemplazar por **Bajo** o **Alto**.

Las acciones que suelen efectuarse contra Missing Values (Datos Faltantes) son:

- Ignorarlos.
- Eliminar la columna (se eliminan el 100% de los datos).
- Filtrar la columna (se eliminan algunos de los datos).
- Filtrar la fila errónea, ya que a veces su origen, se debe a casos especiales.
- Reemplazar el valor.
- Esperar hasta que los datos faltantes estén disponibles.

Antes de elegir alguna acción, es muy importante que se identifique el por qué de la anomalía, para luego actuar en consecuencia, con el fin de evitar que se repitan, agregando de esta manera más valor a los datos de la organización. Puede suceder que en algunos casos, los valores faltantes sean inexistentes, por ejemplo, cuando l@s nuev@s asociad@s o client@s, no posean consumo medio del último año.

Carga

Esta función es responsable de ejecutar las tareas relacionadas con Carga Inicial (Initial Load) y Actualización periódica (Update).

- La Carga Inicial (Initial Load), se refiere a la primera carga de datos que recibe el DW. Generalmente, esta tarea consume un tiempo considerable, debido a que se insertan gran cantidad de registros pertenecientes a períodos largos de tiempo.
- La Actualización Periódica (Update), se refiere a la inserción de pequeños volúmenes de datos, y su frecuencia está dada en función de la granularidad (cuán resumidos se encuentran los datos) del DW y los requerimientos de l@s usuari@s. El objetivo de esta tarea es añadir al DW solo aquellos datos que se generaron a partir de la última actualización (delta de cambios).

Previo a una actualización, se deben identificar los cambios (delta) en las fuentes originales; esto se realiza, en la mayoría de los casos, mediante la fecha de la última actualización. Para efectuar esta operación, se pueden realizar las siguientes acciones:

- ▶ Cotejar las instancias de los Data Sources involucrados.
- ▶ Utilizar triggers (herramienta de los SGBD que consta de una porción de código que se dispara de forma automática ante un evento) para informar de los cambios sucedidos en los Data Sources.
- ▶ Recurrir a Marcas de Tiempo (Time Stamp), en los registros de los Data Sources.
- ▶ Comparar los datos existentes entre el Data Source y el DW.
- ▶ Hacer uso de técnicas mixtas.

Si este control consume demasiado tiempo y esfuerzo, o simplemente NO puede llevarse a cabo por algún motivo en particular, existe la posibilidad de cargar el DW desde cero: este proceso se denomina Carga Total (Full Load). Esta acción involucra el vaciado previo del DW.

El proceso de Carga también es responsable de mantener la estructura del DW e involucra conceptos como:

- Relaciones muchos a muchos.
- Claves Subrogadas.
- Dimensiones Lentamente Cambiantes.
- Dimensiones Degeneradas.

Proceso ETL

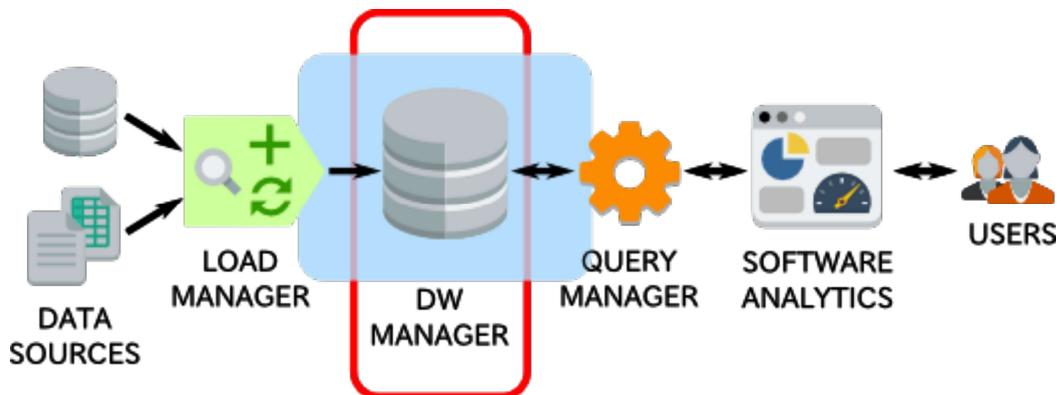
A continuación detallaremos la relación que existe entre los procesos ETL (Extracción, Transformación y Carga), los Data Sources, el Staging Area y el DW:



El proceso ETL trabaja de la siguiente manera:

- Se extraen los datos relevantes desde los Data Sources y se depositan en la Staging Area.
- Se integran y transforman los datos, para evitar inconsistencias.
- Se cargan los datos desde el Staging Area al DW.

3) Data Warehouse Manager



El DW Manager está compuesto por una serie de aplicaciones de software dedicadas a gestionar:

- el DW (SGDB),
- las conexiones a base de datos y otros Data Sources,
- las estructuras de datos (Cubos Multidimensionales, Business Models, etc.),
- información de autenticación y autorización (credenciales de acceso, users, roles, permisos, etc), y
- otros metadatos.

Base de datos multidimensional

Un Data Warehouse es una base de datos con estructura multidimensional, lo cual se traduce en una forma específica de almacenamiento en la cual se definen dos estructuras principales :

- tablas de Hechos y
- tablas de Dimensiones.

La utilización de tablas de Hechos y Dimensiones, facilita la creación de estructuras de datos (Cubos Multidimensionales, Business Models, etc.) y posibilita que las consultas al SGBD sean respondidas con mucha performance.

Tipos de Modelos lógicos

Existen tres formas de modelar las tablas de Hechos y Dimensiones:

-  Esquema en estrella (Star Scheme).
-  Esquema copo de nieve (Snowflake Scheme).
-  Esquema constelación (Starflake Scheme).

Estos modelos están concebidos con el objetivo de facilitar el acceso a consultas complejas y con gran cantidad de agregaciones, es por ello que se encuentran desnormalizadas o semi desnormalizadas, reduciendo de esta manera al mínimo la cantidad de JOINs que deben emplearse para acceder a los datos requeridos.

Tipos de Implementación

Los modelos lógicos pueden ser implementados de diferentes maneras:

-  Relacional – ROLAP.
-  Multidimensional – MOLAP.
-  Híbrido – HOLAP.

Tablas de Dimensiones

Las tablas de Dimensiones proveen el medio para analizar los datos dentro del contexto del negocio. Veamos algunos ejemplos:



Las tablas de Dimensiones:

- contienen datos cualitativos y
- representan los aspectos de interés,
- mediante los cuales l@s usuari@s podrán filtrar y manipular los Hechos almacenados en las tablas de Hechos.

Las tablas de Dimensión contienen los siguientes tipos de campos:

- Clave principal.
- Claves foráneas (solo para esquemas copo de nieve y constelación).
- Datos de referencia *primarios*: datos que identifican la Dimensión. Por ejemplo: nombre del cliente.
- Datos de referencia *secundarios*: datos que complementan la descripción de la Dimensión. Por ejemplo: e-mail del cliente, celular del cliente, etc. Estos datos no son significativos para tomar decisiones, pero son potencialmente valiosos para implementarla.

Usualmente la cantidad de tablas de Dimensiones, aplicadas a un tema de interés en particular, varían entre tres y quince.

Es recomendable que la clave principal de las tablas de Dimensiones sea independiente de las claves de los Data Sources, ya que, por ejemplo, si estos últimos son recodificados, el DW quedaría inconsistente. Estas claves independientes se denominan Claves Subrogadas.

Tiempo

En un DW, la creación y el mantenimiento de una tabla de Dimensión Tiempo es obligatoria, y la definición de granularidad y estructuración de la misma depende de la dinámica del negocio que se esté analizando. Toda la información dentro del DW posee su propio sello de tiempo que determina la ocurrencia de un Hecho específico, representando de esta manera diferentes versiones de una misma situación.

La Dimensión Tiempo NO es sola una secuencia cronológica representada de forma numérica, sino que mantiene niveles jerárquicos especiales que son representativos de las actividades de la organización. Esto se debe a que los usuarios podrán por ejemplo analizar las ventas realizadas teniendo en cuenta el día de la semana en que se produjeron, quincena, mes, trimestre, semestre, año, estación, etc.

La forma de diagramar la Dimensión Tiempo es muy sencilla, se debe tomar el campo que indique la fecha en que sucedieron los Hechos y luego analizar dicha fecha para crear los campos requeridos.

Veamos un ejemplo:

dw_tiemposlookup	
idFecha	INT(10)
anio	INT(11)
semestre	INT(11)
trimestre	INT(11)
bimestre	INT(11)
mesNúmero	INT(11)
mesLetra	VARCHAR(15)
semanaMes	INT(11)
diaSemanaNúmero	INT(11)
diaSemanaLetra	VARCHAR(15)
estacion	VARCHAR(10)
Indexes	

En el ejemplo anterior se ha definido más de una Jerarquía para la Dimensión tiempo. Esto es lógico ya que hay Jerarquías que NO son compatibles, por ejemplo **anio-semestre-bimestre** es una Jerarquía compatible, pero **anio-trimestre-bimestre** NO lo es ya que NO todos los **bimestres** están contenidos en un **trimestre**.

Existen muchas maneras de diseñar esta tabla, por lo cual debemos evaluar la temporalidad de los datos, la forma en que trabaja la organización, los resultados que se esperan obtener del DW relacionados con una unidad de tiempo y la flexibilidad que se desea obtener de dicha tabla.

Fecha & Hora



Si se requiere analizar los datos por fecha (año, mes, día, etc) y por hora (hora, minuto, segundo, etc), lo más recomendable es confeccionar dos tablas de Dimensión Tiempo:

- una conteniendo los datos referidos a la fecha y
- la otra, con los datos referidos a la hora.

Día Juliano

Si bien, el lenguaje SQL ofrece funciones del tipo **DATE**, en la tabla de Dimensión Tiempo, se modelan y presentan datos temporales que, en algunos casos, son complejos de calcular en una consulta.

Es conveniente mantener en la tabla de Dimensión Tiempo un campo que se refiera al día Juliano. El día juliano se representa a través de un número secuencial e identifica únicamente cada día. Mantener este campo permitirá la posibilidad de realizar consultas que involucren condiciones de filtrado de fechas desde-hasta, mayor que, menor que, etc, de manera sencilla e intuitiva.

Por ejemplo, si a partir de determinada fecha se desean analizar los datos de los 81 días siguientes:

- el valor **desde** sería el día Juliano de la fecha en cuestión y
- el valor **hasta** sería igual a **desde** más **81**.

Tablas de Hechos

Las Tablas de Hechos contienen los Hechos que serán utilizados por l@s usuari@s del DW para analizar y responder preguntas de negocio.

Veamos un ejemplo:

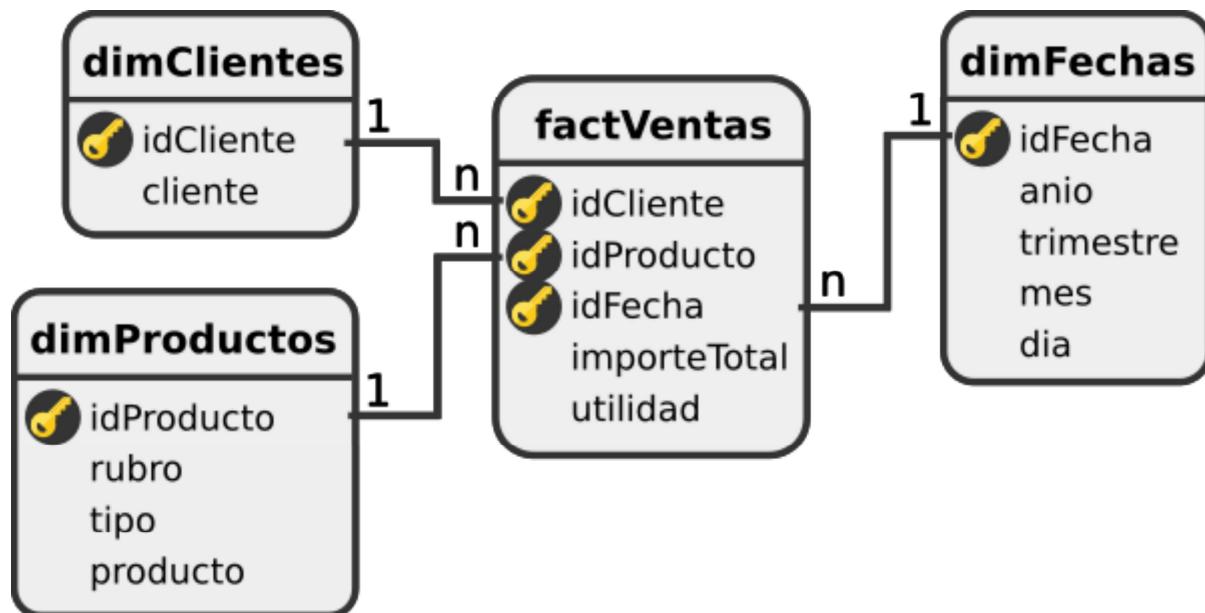


Los Hechos (o Dato agregado):

- son datos cuantitativos,
- que son filtrados, agrupados y explorados a través de condiciones definidas en las tablas de Dimensiones.

Los datos presentes en las tablas de Hechos constituyen el volumen del DW, y pueden estar compuestos por millones de registros dependiendo de su granularidad y la antigüedad de la organización.

El registro del Hecho posee una clave primaria que está compuesta por las claves primarias de las tablas de Dimensiones relacionadas a éste.



En la imagen anterior se puede apreciar un ejemplo de lo antes mencionado. La tabla de Hechos **factVentas** se ubica en el centro, e irradiando de ella se encuentran las tablas de Dimensiones **dimClientes**, **dimProductos** y **dimFechas**, que están conectadas mediante sus claves primarias. Es

por ello que la clave primaria de la tabla de Hechos es la combinación de las claves primarias de sus Dimensiones.

Los Hechos en este caso son `importeTotal` y `utilidad`.

Hechos vs Indicadores

En ocasiones se tiende a confundir los Hechos y los Indicadores. A continuación se expondrán las diferencias entre ellos:

- Los Hechos son aquellos datos que residen en una tabla de Hechos.
- Los Indicadores hacen uso de los Hechos para obtener un valor analizable y se definen mediante una serie de metadatos:
 - ▶ nombre representativo, descripción, I18N, etc;
 - ▶ tipo de agregación al momento de crear una estructura de datos (Cubo Multidimensional, Business Model). Las *agregaciones* más utilizadas son: SUM, MAX, MIN, COUNT, AVG, porcentajes, fórmulas, etc;
 - ▶ agregaciones alternativas;
 - ▶ tipo de datos (siempre numéricos).

Ejemplos de Hechos

Algunos ejemplos de Hechos y su constitución:

- `importeTotal = precioProducto * cantidadVendida`
- `rentabilidad = utilidad / patrimonioNeto`
- `cantidadVentas = cantidad`
- `promedioGeneral = AVG(notasFinales)`

A la izquierda de la igualdad se encuentran los Hechos y a la derecha los campos de los Data Sources. En el último ejemplo se realiza un cálculo de *agregación* para establecer el Hecho.

Hechos Básicos y Derivados

Existen dos tipos de Hechos, los Básicos y los Derivados, a continuación se detallará cada uno de ellos, teniendo en cuenta para su exemplificación la siguiente tabla de Hechos:



-
- Hechos Básicos: son aquellos que se encuentran representados por un campo de una tabla de Hechos. Los campos **precio** y **cantidad** de la tabla anterior son Hechos Básicos.
 - Hechos Derivados: son aquellos que se obtienen a partir de una expresión, en otras palabras, combinando uno o más Hechos con alguna operación matemática/lógica y que también residen en una tabla de Hechos. Tienen la ventaja de almacenarse ya calculados, por lo que serán accedidos a través de consultas SQL sencillas devolviendo resultados rápidamente. Por otro lado, requieren más espacio físico en el DW y más tiempo de proceso en los ETL. El campo **total** de la tabla anterior es un Hecho Derivado, y se conforma de la siguiente manera:
 - ▶ **total = precio * cantidad**

Agregadas y Preagregadas

Las tablas de Hechos Agregadas y Preagregadas se utilizan para almacenar un resumen de los datos, es decir, se guardan los datos en niveles de granularidad superior a los que inicialmente fueron obtenidos y/o gestionados.

Para generar tablas de Hechos Agregadas o Preagregadas, es necesario establecer un criterio con el cual realizar el resumen. Por ejemplo, esto ocurre cuando se desea obtener información de ventas sumarizadas por mes.

Cada vez que se requiere que los datos en una consulta se presenten en un nivel de granularidad superior al que se encuentran almacenados en el DW, se debe llevar a cabo un proceso de agregación.

El objetivo general de las tablas de Hechos Agregadas y Preagregadas es el mismo, pero cada una de ellas tiene una manera de operar diferente:

- Tablas de Hechos Agregadas: se generan luego de que se procesa la consulta correspondiente a la tabla de Hechos que se resumirá. En muchos casos, estos resúmenes son utilizados por las herramientas de software de análisis de forma automática a fin de mejorar la respuesta.
- Tablas de Hechos Preagregadas: se generan antes de que se procese la consulta correspondiente a la tabla de Hechos que se resumirá. De esta manera, la consulta se realiza contra una tabla que ya fue previamente agregada. Posee un nivel de granularidad menor al de la tabla de Hechos. Estos resúmenes deben generarse y almacenarse al momento de poblar/actualizar el DW, utilizando procesos ETL.

Más sobre las tablas de Hechos Preagregadas

Beneficios:

- Reduce la utilización de recursos de hardware en la que se incurre en el momento de calcular las agregaciones.
- Reduce el tiempo utilizado en la generación de consultas por parte de l@s usuari@s.
- Son muy útiles en los siguientes casos generales:
 - Cuando los datos a nivel detalle (menor nivel granular) son innecesarios y/o NO son requeridos.
 - Cuando una consulta a determinado nivel de granularidad es solicitada con mucha frecuencia.
 - Cuando el volumen de datos es muy grande y las consultas demoran en ser procesadas.

Desventajas:

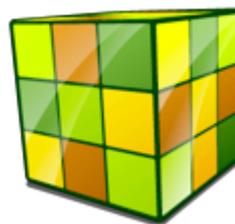
- Requieren de la creación y mantenimiento de nuevos procesos ETL.
- Requieren espacio de almacenamiento adicional en el DW.

- Resulta complejo reconocer qué agregaciones son necesarias mantener en tablas.
-

Cubo Multidimensional: introducción

El Cubo Multidimensional es la estructura de datos más utilizada y requerida, pero también es la más compleja de comprender.

Un Cubo Multidimensional, representa los datos planos (que se encuentran en filas y columnas), en una matriz de N Dimensiones.



Los componentes más importantes que se pueden incluir en un Cubo Multidimensional son:

- Indicadores,
- Atributos y
- Jerarquías.

Indicadores

- Los Indicadores son definiciones a partir de las cuales se obtendrán valores numéricos mediante los cuales se analizará la situación de la organización.
- Los Indicadores pueden calcularse aplicando funciones de agregación sobre los Hechos o mediante expresiones complejas.
- El valor que tomen los Indicadores, dependerá de los Atributos/Jerarquías que se empleen para su análisis.

Atributos

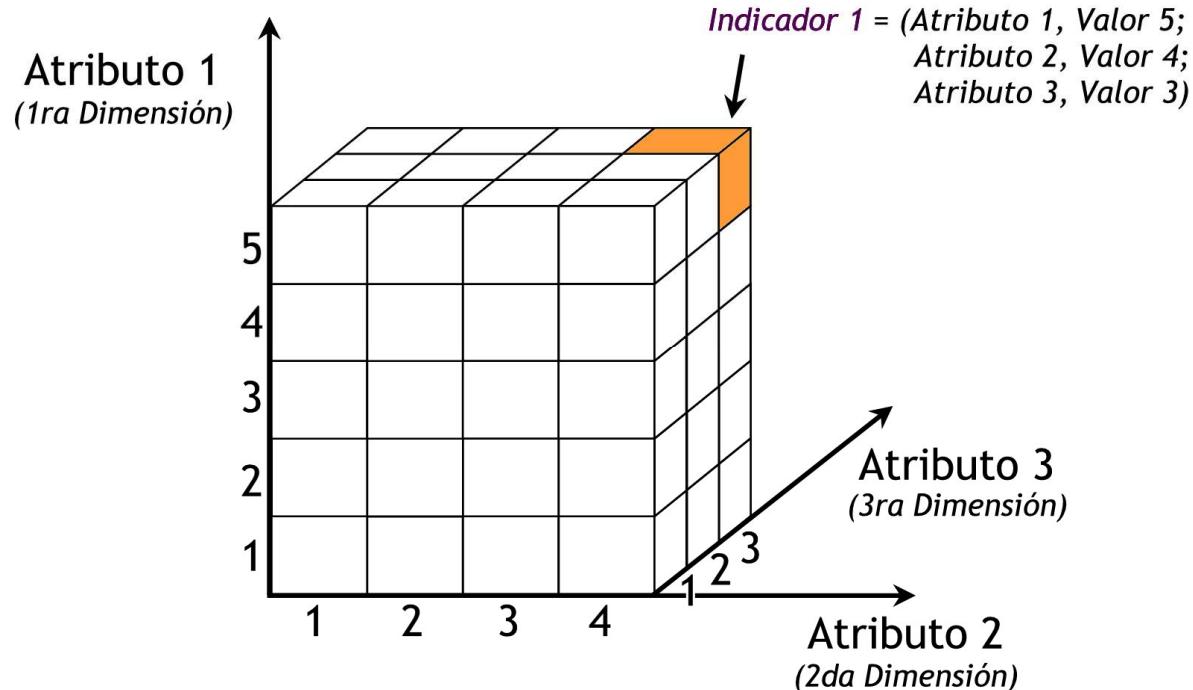
- Los Atributos son los criterios de análisis mediante los cuales analizaremos los Indicadores.
- El valor de los Atributos se obtiene de los campos de las tablas de Dimensión, aunque también pueden calcularse con expresiones complejas.

Jerarquías

- Una Jerarquía (en este contexto) es una relación lógica del tipo padre-hijo entre los Atributos.
- Al emplear Jerarquías podemos analizar los datos desde el nivel más general al más detallado y viceversa.
- Las Jerarquías manejan el nivel de agregación de los Hechos.

Representación matricial

En un Cubo Multidimensional, los Atributos existen a lo largo de varios ejes o Dimensiones, y de su intersección dependerá el valor del Indicador que se está evaluando.



El Cubo de la figura está compuesto de tres Dimensiones y un Indicador (**Indicador 1**). Cada una de estas Dimensiones posee solo un Atributo, es decir:

- 1ra Dimensión posee el **Atributo 1**,
- 2da Dimensión posee el **Atributo 2**, y
- 3ra Dimensión posee el **Atributo 3**.

En la imagen anterior se ha seleccionado una intersección para demostrar la correspondencia con los valores de los Atributos. En este caso, el **Indicador 1**, representa el cruce de:

- El valor 5 de **Atributo 1**,
- el valor 4 de **Atributo 2** y
- el valor 3 de **Atributo 3**.

El resultado del análisis está dado por los cruces matriciales de acuerdo a los valores de las Dimensiones seleccionadas.

Modelos del DW

A continuación se detallarán los tres tipos de estructuras de modelo del DW:

-  Esquema en Estrella (Star Scheme).
-  Esquema Copo de Nieve (Snowflake Scheme).
-  Esquema Constelación (Starflake Scheme).

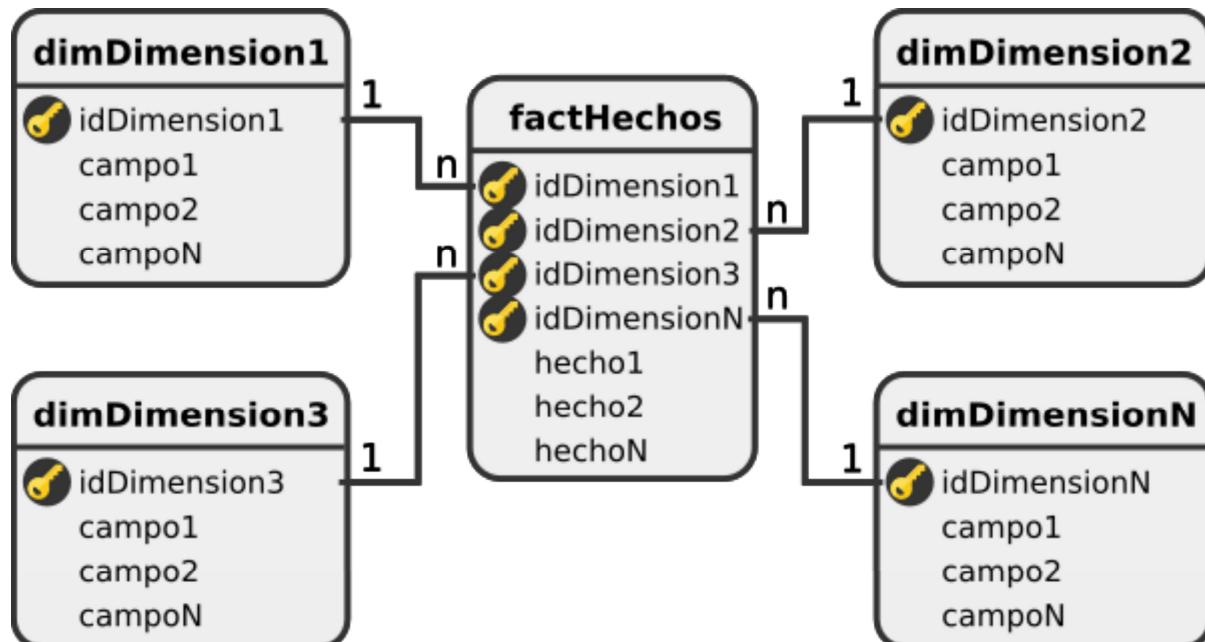
Estrella



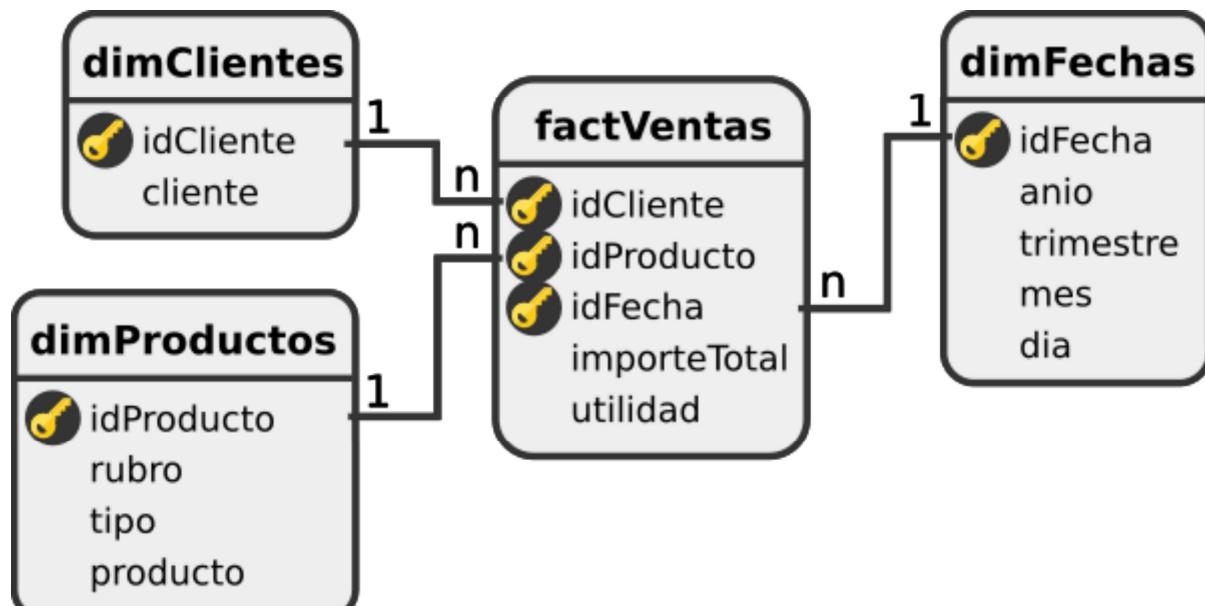
El Esquema en Estrella (Star Scheme) está formado por:

- una tabla de Hechos y
- una o más tablas de Dimensiones relacionadas a través de sus respectivas claves.

La siguiente figura representa un Esquema en Estrella estándar:



El modelo utilizado cuando se abordó el tema de las tablas de Hechos, es un Esquema en Estrella, por lo cual se lo volverá a mencionar para explicar sus cualidades:

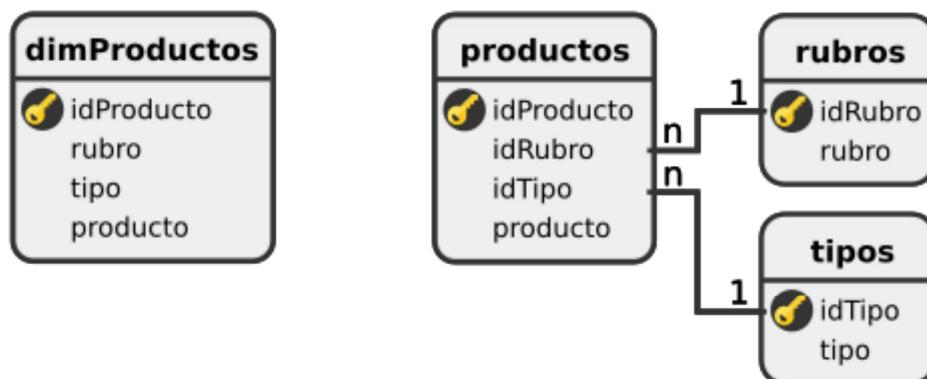


Las tablas de Dimensiones de este modelo, se encuentran desnormalizadas, es decir que no se presentan en tercera forma normal (3ra FN).

En la siguiente imagen se presentan dos modelos:

- **Normalizado:** modelo de la derecha, conformado por tablas relacionales estándar con sus respectivas uniones.
- **Desnormalizado:** modelo de la izquierda, conformado tras desnormalizar el modelo de la derecha.

Desnormalizado Normalizado



Al realizar la desnormalización sobre el modelo de la derecha:

- la tabla de Dimensión **dimProductos** elimina las uniones y las claves y
- mantiene solo los datos descriptivos: **rubro**, **tipo** y **producto**.

Cuando se normaliza se pretende eliminar la redundancia, es decir, la repetición de datos y las dependencias funcionales entre los datos, los modelos multidimensionales requieren precisamente lo contrario.

Entonces, la principal ventaja de la desnormalización es:

- evitar uniones (JOIN) entre las tablas cuando se realizan consultas, procurando así un mejor tiempo de respuesta y una mayor sencillez con respecto a su utilización.

Y la desventaja de la normalización es:

- redundancia y
- consumo adicional de espacio de almacenamiento.

Características del Esquema en Estrella

A continuación se destacarán las principales características del Esquema en Estrella:

- Es el más simple de interpretar .
- Posee los mejores tiempos de respuesta.
- Es soportado por todos los visores OLAP.
- Su diseño es sencillo de mantener y actualizar.

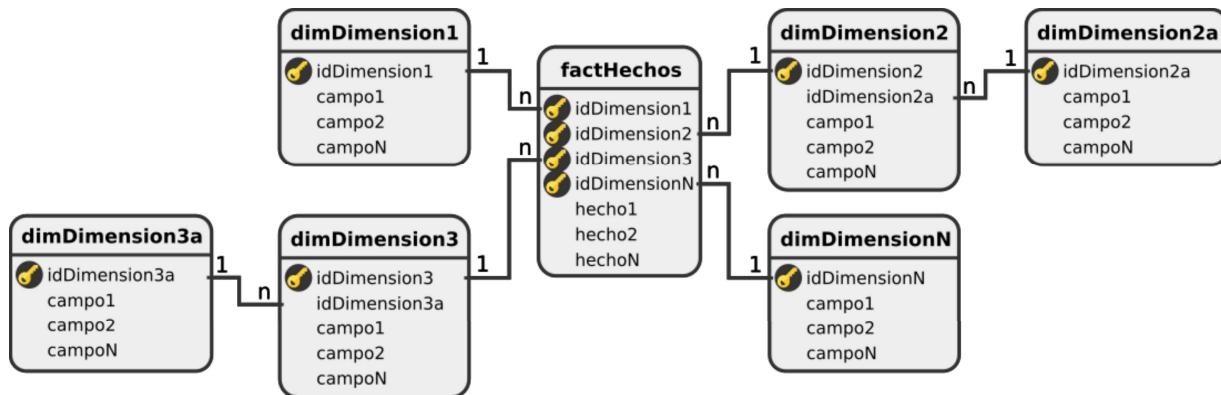
- Existe paralelismo entre su diseño y la forma en que l@s usuari@s visualizan y manipulan los datos.
- Es el modelo elegido para prototipado rápido.

Copo de Nieve



El Esquema Copo de Nieve (Snowflake Scheme) es:

- una extensión del modelo en Estrella, y
- se caracteriza por poseer tablas de Dimensiones organizadas en Jerarquías de Dimensiones.



Como se puede apreciar en la figura anterior:

- existe una tabla de Hechos (**factHechos**) que está relacionada con una o más tablas de Dimensiones, y
- algunas tablas de Dimensiones están relacionadas con otras tablas de Dimensiones.

Este modelo se parece más al modelo transaccional ya que algunas tablas de Dimensiones están normalizadas.

Características del Esquema Copo de Nieve

A continuación se destacarán las principales características del Esquema Copo de Nieve:

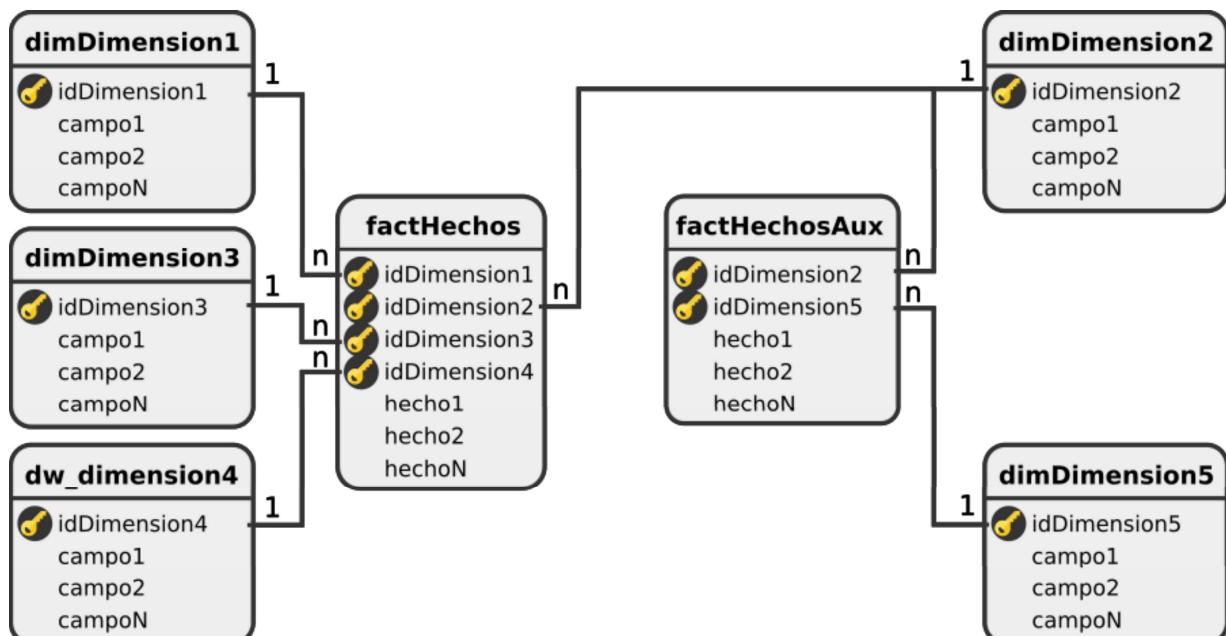
- Posibilita la segregación de los datos de las tablas de Dimensiones.
- Puede implementarse después de que se haya desarrollado un Esquema en Estrella.
- Posee mayor complejidad en su estructura.
- Utiliza menos espacio de almacenamiento.
- Es más eficiente en el caso de tablas de Dimensiones con gran cantidad de registros.
- Su semántica se ajusta a las representaciones de las diferentes Jerarquías de Dimensiones.
- Se deben planificar correctamente las uniones e el indexado, a fin de NO generar sobrecarga en la resolución de consultas.

Constelación



El Esquema en Constelación (Starflake Scheme):

- está compuesto por una serie de Esquemas en Estrella,
- posee una tabla de Hechos Principal (por ejemplo: **factHechos**) y
- una o más tablas de Hechos Auxiliares (por ejemplo: **factHechosAux**), las cuales pueden ser agregaciones de la Principal. Dichas tablas están relacionadas con sus respectivas tablas de Dimensiones.



NO es necesario que las diferentes tablas de Hechos compartan las mismas tablas de Dimensiones. Las tablas de Hechos Auxiliares pueden vincularse con solo algunas de las tablas de Dimensiones asignadas a la tabla de Hechos Principal, y también pueden hacerlo con nuevas tablas de Dimensiones.

Características del Esquema en Constelación

Las características y diseño del Esquema en Constelación son muy similares a las del Esquema en Estrella, con las siguientes diferencias:

- Permite tener más de una tabla de Hechos, por lo cual se tendrá mayor capacidad analítica.
- Contribuye a la reutilización de las tablas de Dimensiones, ya que una misma tabla de Dimensión puede utilizarse para varias tablas de Hechos.

OLTP vs DW

Los OLTP (sistemas transaccionales) se diseñan para dar soporte al procesamiento diario de datos de las organizaciones, es decir, gran cantidad de transacciones en cortos períodos de tiempo. Se enfatiza la maximización de la capacidad transaccional y la concurrencia. Las estructuras de las bases de datos se encuentran altamente normalizadas, de esta forma, se aumenta la eficiencia de las escrituras de datos. En cuanto a las lecturas o consultas, son mínimas y acceden a un pequeño número de registros. Estos sistemas están fuertemente condicionados por los procesos operacionales que deben soportar, administrar datos y muy poca información. A diferencia de los OLTP, los Data Warehouses están diseñados para llevar a cabo procesos de consulta y análisis complejos, que serán la base para la toma de decisiones estratégica y táctica de alto nivel.

A continuación una tabla comparativa entre los dos ambientes, que resume sus principales diferencias:

	OLTP	Data Warehouse
Objetivo	Soportar actividades transaccionales diarias	Consultar y analizar información táctica y estratégica
Tipo de datos	Operacionales	Para la toma de decisiones
Modelo de datos	Normalizado	Desnormalizado
Consulta	SQL	SQL más extensiones
Datos consultados	Actuales	Actuales e históricos
Horizonte de tiempo	60 - 90 días	5 - 10 años
Tipos de consultas	Repetitivas, predefinidas	NO previsibles, dinámicas
Nivel de almacenamiento	Nivel de detalle	Nivel de detalle y diferentes niveles de agregación
Acciones disponibles	Alta, baja, modificación y consulta	Carga y consulta
Número de transacciones	Alto	Medio - Bajo
Tamaño	Pequeño - Mediano	Grande
Tiempo de respuesta	Pequeño (segundos - minutos)	Variable (minutos - horas)
Orientación	Orientado a las aplicaciones	Orientado al negocio
Estructura	Estable	Varía de acuerdo a su propia evolución y utilización

Implementación

Veremos a continuación los tres tipos de implementación del modelo multidimensional:

-  Relacional – ROLAP.
-  Multidimensional – MOLAP.
-  Híbrido – HOLAP.

ROLAP

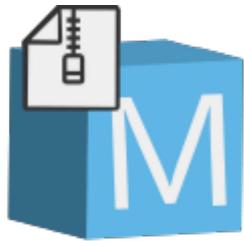


En los sistemas ROLAP (Relational On Line Analytic Processing), los Cubos Multidimensionales se generan en el momento en que se realizan las consultas.

Este proceso se puede resumir a través de los siguientes pasos:

1. Se describe la metadata del Cubo: Indicadores, Atributos, Jerarquías, etc.
2. Se almacena la metadata.
3. El Motor Multidimensional del visor OLAP que se esté utilizando, carga la metadata con la cual realizará un mapeo entre los datos del DW y los Atributos, Indicadores, etc.
4. Cada vez que se actualiza el DW, se debe borrar la caché del Motor Multidimensional a fin de visualizar los nuevos datos. Esto se debe a que los motores ROLAP hacen uso exhaustivo del caché, lo cual permite que el motor evite consultar dos veces el mismo dato, ya que una vez consultado será almacenado en caché.

MOLAP

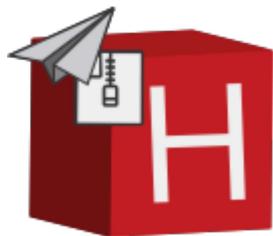


MOLAP (Multidimensional On Line Analytic Processing) precalcula los Cubos Multidimensionales y los almacena físicamente.

Este proceso se puede resumir a través de los siguientes pasos:

1. Se seleccionan los Indicadores, Atributos, Jerarquías, etc., que compondrán el Cubo Multidimensional.
2. Se precisan los datos del Cubo, es decir, todas las combinaciones posibles entre los niveles de las Jerarquías de cada Dimensión.
3. Se ejecutan las consultas sobre los datos precalculados del Cubo.
4. Cada vez que se actualiza el DW, se debe precalcular y guardar el Cubo, para que contenga los nuevos datos.

HOLAP



HOLAP (Hybrid On Line Analytic Processing) constituye un sistema híbrido entre MOLAP y ROLAP, que combina estas dos implementaciones.

Los datos agregados y precalculados se almacenan en estructuras multidimensionales y los de menor nivel de detalle en estructuras relacionalles. Es decir:

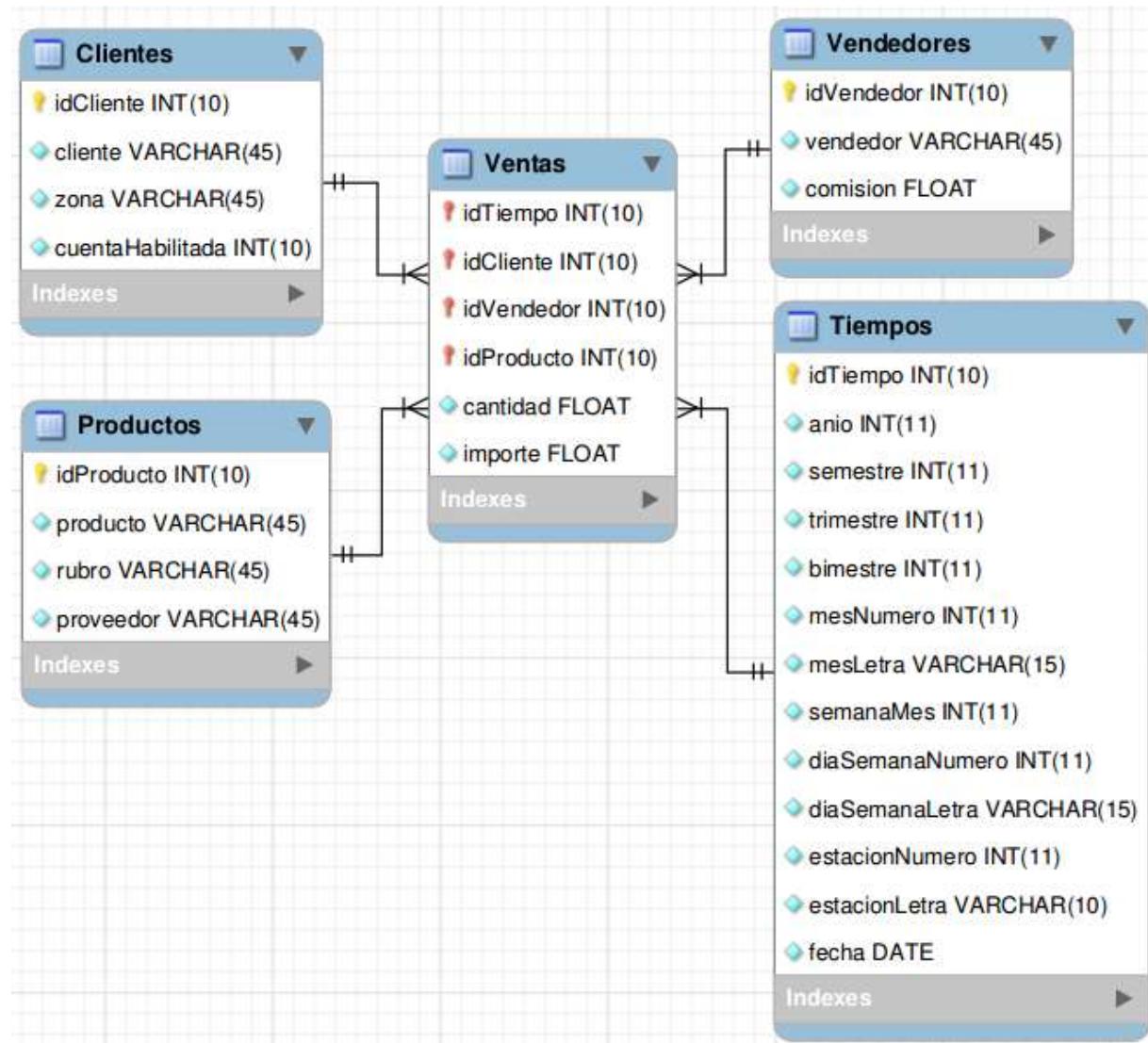
- se utilizará ROLAP para navegar y explorar los datos a bajos niveles de granularidad y
- se utilizará MOLAP para la explotación de datos precalculados, por lo general sumatorias o funciones de alto nivel de agregación, suelen ser los más utilizados en los dashboards.

Cubo Multidimensional: profundización

Con una idea clara acerca de las formas de modelar e implementar un DW, se detallará paso a paso el proceso de construcción de Cubos Multidimensionales.

Se utilizará para el caso una representación genérica, para que luego sea sencillo trasladar los conocimientos aquí adquiridos al momento de trabajar con un software de creación de Cubos Multidimensionales.

El siguiente Esquema en Estrella, constituye la base sobre la cual se exemplificará el desarrollo del Cubo:

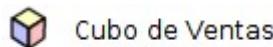


El Esquema en Estrella está compuesto por:

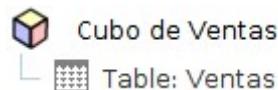
- una tabla de Hechos: **Ventas**.
- cuatro tablas de Dimensiones: **Clientes**, **Productos**, **Vendedores** y **Tiempos**.

Pasos básicos

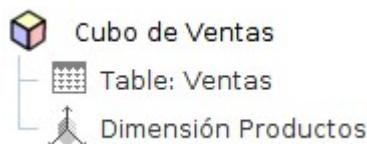
1) Se creará un Cubo Multidimensional llamado **Cubo de Ventas**:



2) Se indicará que la tabla de Hechos a utilizar es **Ventas**:



3) Se creará una Dimensión para analizar los productos. Su nombre será **Dimensión Productos**:



3.1) Se añadirá a **Dimensión Productos** una Jerarquía. Su nombre será **Jerarquía Productos**:



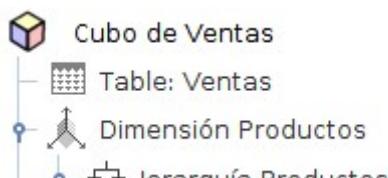
3.2) Se indicará que **Jerarquía Productos** estará basada en los campos de la tabla de **Dimensión Productos**:



3.3) Se añadirá a la **Jerarquía Productos** el Atributo **Producto**, que estará basado en el campo **producto**:



4) De forma similar que se hizo en el punto anterior, se creará la **Dimensión Clientes**:





5) Se creará un Indicador, llamado **Cantidad** que se calculará de la siguiente manera:

- SUM(cantidad)**



En este momento, se ha construido un Cubo Multidimensional de dos Dimensiones y un Indicador.

Si se analiza el Indicador **Cantidad** utilizando los Atributos **Producto** y **Cliente**, la representación matricial será la siguiente:

Producto 1	40	25	60
Producto 2	21	55	45
Producto 3	13	32	43
	Cliente 1	Cliente 2	Cliente 3

La intersección de las Dimensiones representa la cantidad vendida de cada producto a cada cliente.

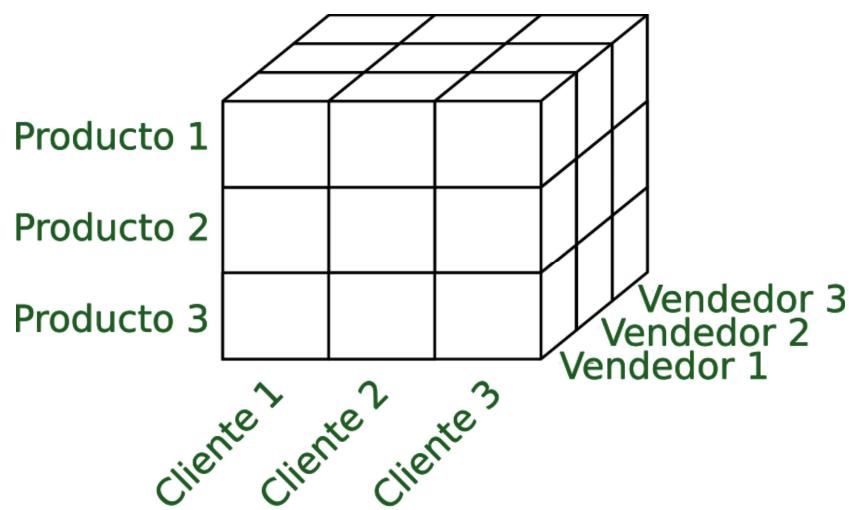
Por ejemplo:

- La cantidad vendida del **Producto 1** al **Cliente 1** es de **40** unidades.
- La cantidad vendida del **Producto 1** al **Cliente 2** es de **25** unidades.
- La cantidad vendida del **Producto 1** al **Cliente 3** es de **60** unidades.

6) De forma similar que se hizo en el punto anterior, se creará la **Dimensión Vendedores**:

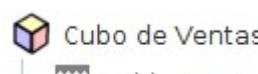


En este momento, se ha construido un Cubo Multidimensional de tres Dimensiones y un Indicador.



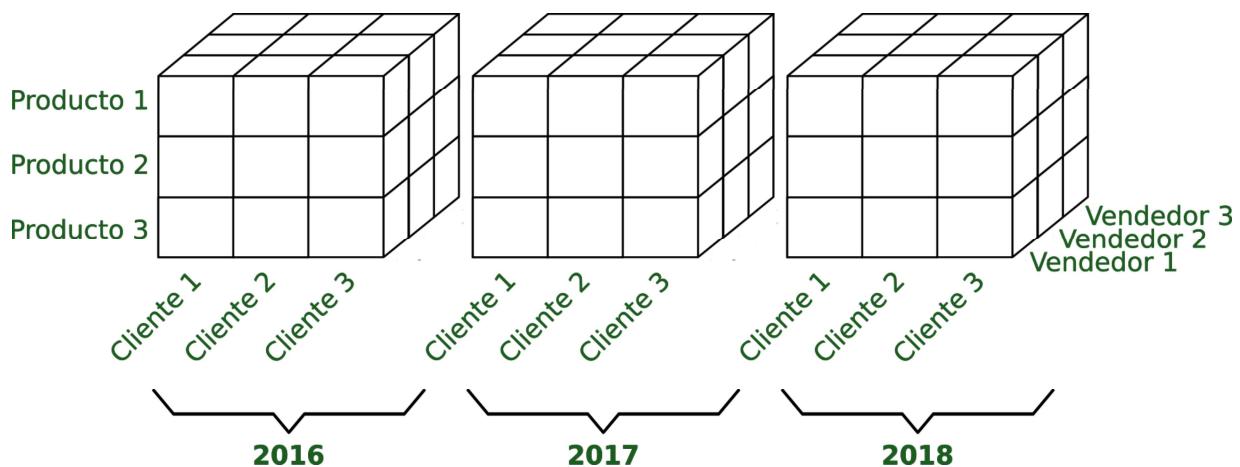
En este caso los valores del Indicador **Cantidad** están definidos por la intersección de tres Dimensiones. Se pueden medir las cantidades vendidas por **Producto**, **Cliente** y **Vendedor**.

7) De forma similar que se hizo en el punto anterior, se creará la **Dimensión Años**:





En este momento, se ha construido un Cubo Multidimensional de cuatro Dimensiones y un Indicador.



En esta versión del Cubo, la lectura del Indicador **Cantidad**, está condicionada por las cantidades vendidas de cada **Producto**, a cada **Cliente**, de cada **Vendedor**, en cada **Año**.

La última imagen expresa de forma muy clara los conceptos expuestos anteriormente sobre la Dimensión Tiempo, en donde se establecía que pueden existir diferentes versiones de la situación del negocio.

Cabe aclarar que pueden crearse tantos Cubos como sean necesarios sin que su coexistencia implique inconveniente alguno.

Jerarquías

En el Cubo Multidimensional que se construyó anteriormente creamos Dimensiones y dentro de ellas añadimos Jerarquías; y en cada Jerarquía definimos un solo Atributo.

Tenemos la posibilidad de añadir más Atributos dentro de las Jerarquías, en donde los Atributos consecutivos posean una relación de padre-hijo (1-n) entre el Atributo superior y el inferior.

Veamos la siguiente Jerarquía:



Dentro de la **Jerarquía Años** definimos tres Atributos **Año**, **Mes** y **Día**.

El Atributo más general se posiciona en la cabecera y los más particulares hacia abajo:

- Un *año* posee uno o más *meses del año*; y un *mes del año* pertenece solo a un *año*. En otras palabras, el *año 2017* posee muchos meses; y un *mes del 2017* solo pertenece al *año 2017*
- Un *mes del año* posee uno o más *días del año*; y un *día del año* pertenece solo a un *mes de año*. En otras palabras, *Enero de 2017* posee muchos días; y un *día de Enero de 2017* solo pertenece a *Enero de 2017*.



También tenemos la posibilidad de añadir más Jerarquías dentro de una Dimensión. Por ejemplo:



El Atributo Estación presenta las estaciones del año: **Otoño, Invierno, Primavera, Verano**.

Relación jerárquica

Una relación representa la forma en que dos Atributos interactúan dentro de una Jerarquía. Existen básicamente dos tipos de relaciones:

- Explícitas: son las más comunes y se pueden modelar a partir de Atributos directos y están en línea continua de una Jerarquía, por ejemplo, un país posee una o más provincias y una provincia pertenece solo a un país.
- Implícitas: su relación NO es de vista directa, por ejemplo, una provincia tiene uno o más ríos, pero un río pertenece a una o más provincias. Se trata de una *relación muchos a muchos*.

Metadatos



Los metadatos son datos sobre los datos, sirven para describir otros datos, que en este caso, existen en la arquitectura del Data Warehousing.

Brindan principalmente información de localización, estructura y significado de los datos, es decir, mapean a los mismos.

El concepto de metadatos es análogo al uso de índices para localizar objetos en lugar de datos.

Los metadatos NO son exclusivos del DWH, forman parte de las bases de datos transaccionales y de la mayoría de las estructuras de almacenamiento. En el caso del Data Warehousing los metadatos son gestionados y mantenidos por l@s usuari@s, pueden ser modificados, exportados, importados; existe una gran interacción entre usuari@s y metadatos, sea esta interacción manual o automática.

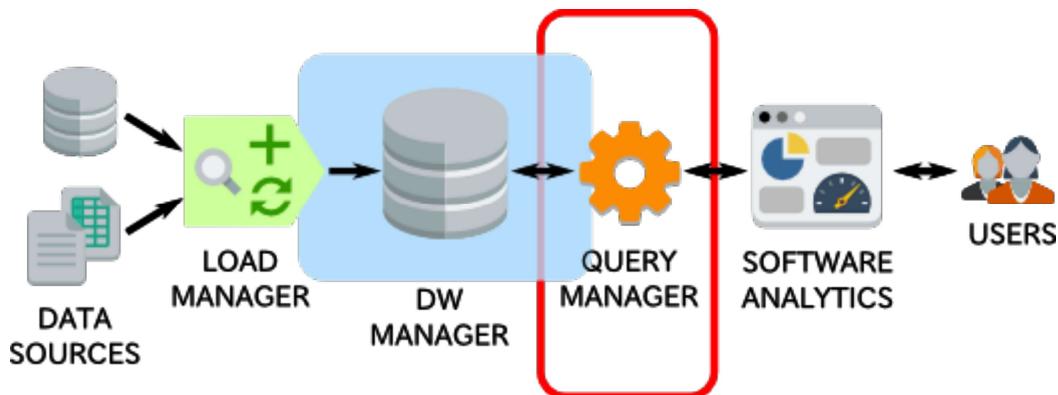
Algunos ejemplos de metadatos en el ambiente DWH son:

- metadatos utilizados por los procesos ETL,
- metadatos de conexión con bases de datos,
- metadatos de estructuras de datos (Cubos Multidimensionales, Business Models, etc.),
- metadatos utilizados por los software analíticos.

Mapping

El término mapping, se refiere a relacionar un conjunto de objetos, tal como actualmente están almacenados en memoria o en disco, con otros objetos. Por ejemplo: una estructura de base de datos lógica, se proyecta sobre la base de datos física. El mapeo se estructura como metadatos.

4) Query Manager



El Query Manager es una pieza fundamental y compleja del proceso de DWH, pues es el encargado de realizar las operaciones necesarias para soportar los procesos de gestión y ejecución de:

- consultas relacionales: como JOIN y agregaciones (SUM, COUNT, AVG, etc), y
- consultas propias del análisis de datos: como DRILL-UP y DRILL-DOWN.

El funcionamiento del Query Manager es el siguiente:

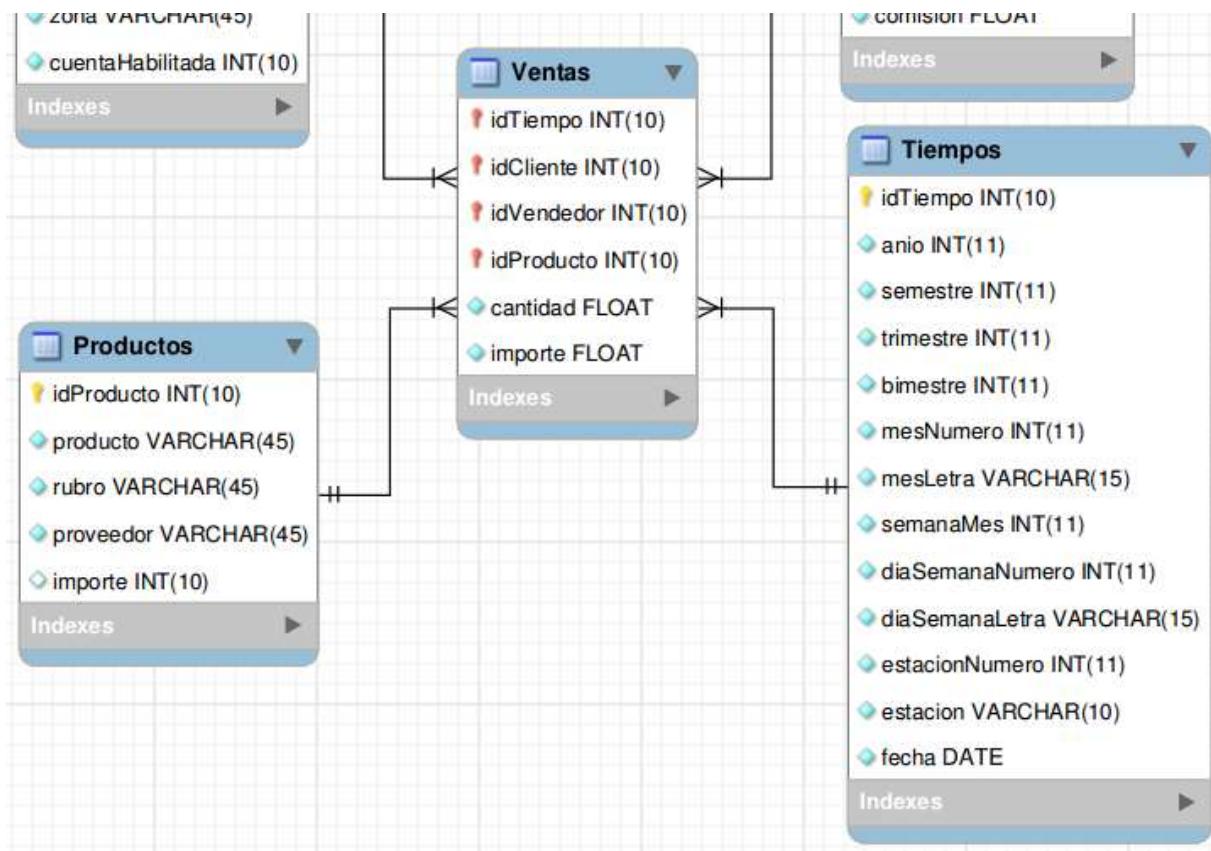
1. recibe consultas de l@s usuari@s, que en general están escritas en un lenguaje de alto nivel (por ejemplo MDX);
2. lee los metadatos que describen los mapeos (Cubo Multidimensional, Business Models, etc.) y reescribe las consultas para que sean ejecutadas en el sistema destino (por lo general SQL);
3. una vez que obtiene los datos y utilizando, nuevamente, las estructuras de metadatos, éstos son transformados a un formato final de alto nivel que será interpretado y renderizado por las herramientas de visualización.

Las principales operaciones que se pueden realizar sobre modelos multidimensionales son:

- Drill-down
- Drill-up
- Drill-across
- Roll-across
- Pivot
- Page
- Drill-through

A continuación, se explicará cada una de ellas y se ejemplificará su utilización, para lo cual se utilizará como guía el siguiente Esquema en Estrella:





El modelo consta de cuatro tablas de Dimensiones y una tabla de Hechos que se encuentra en el centro.

Se volverá a utilizar el Cubo Multidimensional creado anteriormente:



Para simplificar los ejemplos que se presentarán, se utilizará solo una pequeña muestra de datos.

Drill-down

Drill-down es una operación que permite apreciar los datos con un mayor nivel de detalle. Se aplica bajando por los niveles de una Jerarquía definida en un Cubo. Esto brinda la posibilidad de introducir un nuevo nivel o criterio de agregación en el análisis.

Drill-down implica ir de lo general a lo específico.



Para explicar y ejemplificar esta operación se utilizará la siguiente representación tabular:

Dimensión Productos Jerarquía Productos Producto	Dimensión Años Jerarquía Años Año	Indicador Cantidad
Producto 1	2017	40
Producto 2	2017	52

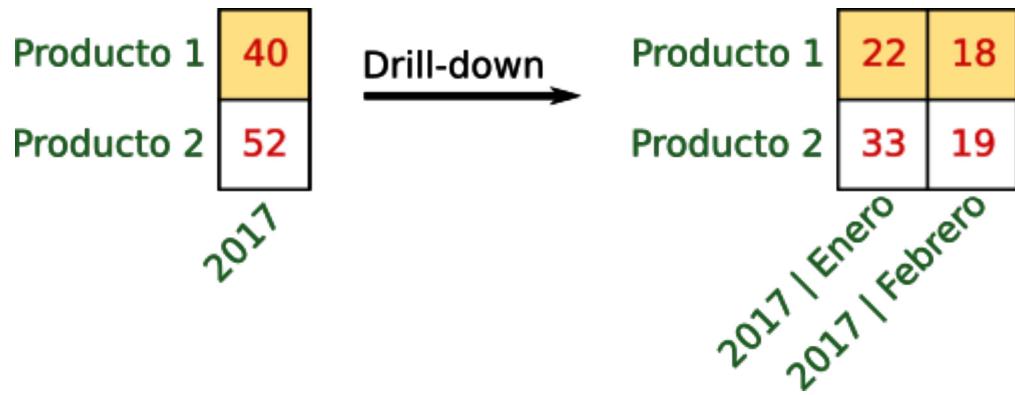
Como puede apreciarse, en la cabecera de la tabla se encuentran los Atributos y el Indicador (destacado con color de fondo diferente) definidos anteriormente en el Cubo Multidimensional; y en el cuerpo se encuentran los valores correspondientes a cada cruce. Se ha resaltado la primera fila, ya que luego se la analizará en detalle.

Se aplicará la operación drill-down sobre **Jerarquía Años**, añadiendo un nivel más de detalle. En este caso, al bajar por la **Jerarquía Años** se añade el Atributo **Mes**.

Dimensión Productos Jerarquía Productos Producto	Dimensión Años Jerarquía Años Año Mes	Indicador Cantidad
Producto 1	2017 Enero	22
Producto 1	2017 Febrero	18
Producto 2	2017 Enero	33
Producto 2	2017 Febrero	19

Como puede apreciarse en los ítems resaltados de la tabla, se agregó un nuevo nivel de detalle (**Año | Mes**) a la lista inicial, y el valor **40** que pertenecía a las ventas del **Producto1**, en el año **2017**, se dividió en dos filas. Esto se debe a que ahora se tendrá en cuenta el Atributo **Mes** para realizar la agregación del Indicador **Cantidad**.

La siguiente imagen muestra este mismo proceso, representado matricialmente:



En adelante se utilizará esta forma para explicar cada operación.

Drill-up

Drill-up es una operación que permite apreciar los datos con un menor nivel de detalle, subiendo por una Jerarquía definida en un Cubo. Esto brinda la posibilidad de quitar un nivel o criterio de agregación en el análisis.

Drill-up implica ir de lo específico a lo general.



Siguiendo con el ejemplo de Drill-down, se tomarán como referencia los resultados anteriores:

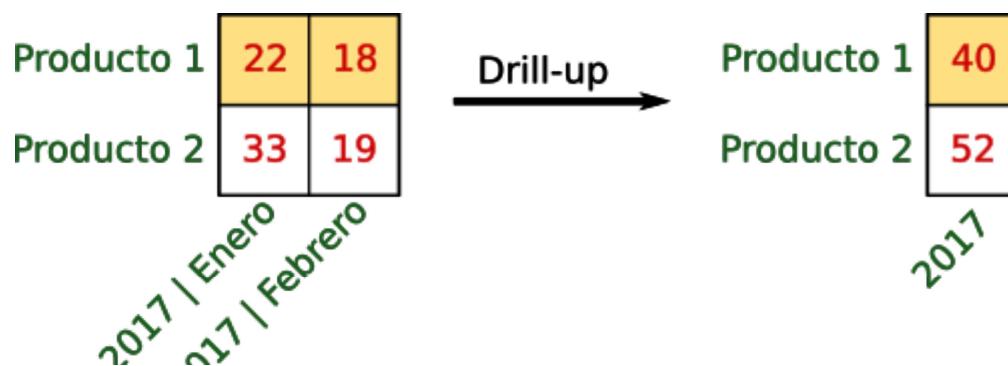
Dimensión Productos Jerarquía Productos		Dimensión Años Jerarquía Años	Indicador
Producto		Año Mes	Cantidad
Producto 1		2017 Enero	22
Producto 1		2017 Febrero	18
Producto 2		2017 Enero	33
Producto 2		2017 Febrero	19

Se aplicará drill-up sobre la **Jerarquía Años**, entonces:

Dimensión Productos Jerarquía Productos		Dimensión Años Jerarquía Años	Indicador
Producto		Año	Cantidad
Producto 1		2017	40
Producto 2		2017	52

Como se puede apreciar, en la fila resaltada se sumarizaron los valores **22** y **18** de la tabla inicial, debido a que al eliminar el Atributo **Mes**, las ventas se agregaron de acuerdo a **Producto** y **Año**.

La siguiente imagen muestra este mismo proceso, representado matricialmente:



✓ 2v

Drill-across

Drill-across es una operación que trabaja de forma similar a drill-down, con la diferencia de que drill-across NO se aplica sobre una Jerarquía, sino que su forma de ir de lo general a lo específico es agregar un Atributo a la consulta como nuevo criterio de análisis.

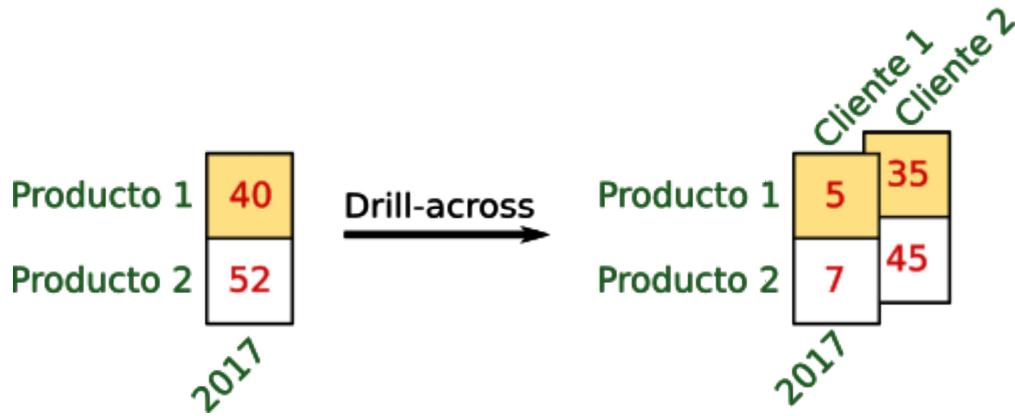
En el siguiente ejemplo, se partirá de los siguientes resultados:

Dimensión Productos Jerarquía Productos Producto	Dimensión Años Jerarquía Años Año	Indicador Cantidad
Producto 1	2017	40
Producto 2	2017	52

Ahora, se aplicará drill-across, al agregar el Atributo **Cliente**. Entonces:

Dimensión Productos Jerarquía Productos Producto	Dimensión Años Jerarquía Años Año	Dimensión Clientes Jerarquía Clientes Cliente	Indicador Cantidad
Producto 1	2017	Cliente 1	5
Producto 1	2017	Cliente 2	35
Producto 2	2017	Cliente 1	7
Producto 2	2017	Cliente 2	45

La siguiente imagen muestra este mismo proceso, representado matricialmente:



Roll-across

Roll-across es una operación que trabaja de forma similar a drill-up, con la diferencia de que roll-across NO se hace sobre una Jerarquía, sino que su forma de ir de lo específico a lo general es quitar un Atributo de la consulta, eliminando de esta manera un criterio de análisis.

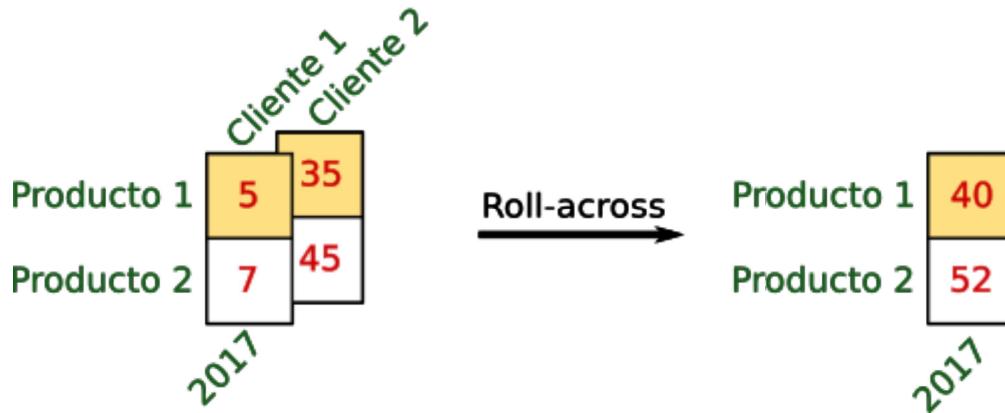
Para mostrar su funcionamiento, se tomará como base la representación tabular anterior:

Dimensión Productos Jerarquía Productos Producto	Dimensión Años Jerarquía Años Año	Dimensión Clientes Jerarquía Clientes Cliente	Indicador Cantidad
Producto 1	2017	Cliente 1	5
Producto 1	2017	Cliente 2	35
Producto 2	2017	Cliente 1	7
Producto 2	2017	Cliente 2	45

Se aplicará la operación roll-across, quitando de la consulta el Atributo **Cliente**, entonces:

Dimensión Productos Jerarquía Productos Producto	Dimensión Años Jerarquía Años Año	Indicador Cantidad
Producto 1	2017	40
Producto 2	2017	52

La siguiente imagen muestra este mismo proceso, representado matricialmente:



Pivot

Pivot es una operación que permite seleccionar el orden de visualización de los Atributos e Indicadores, con el objetivo de analizar la información desde diferentes puntos de vista.

Se tomará como referencia, para explicar esta operación, la siguiente tabla:

Dimensión Productos Jerarquía Productos Producto	Dimensión Años Jerarquía Años Año	Dimensión Clientes Jerarquía Clientes Cliente	Indicador Cantidad
Producto 1	2017	Cliente 1	5
Producto 1	2017	Cliente 2	35
Producto 2	2017	Cliente 1	7
Producto 2	2017	Cliente 2	45

Como puede apreciarse, el orden de los Atributos es: **Producto, Año y Cliente**.

Ahora, se hará Pivot, reorientando la vista multidimensional:

Dimensión Productos Jerarquía Clientes Cliente	Dimensión Años Jerarquía Años Año	Dimensión Clientes Jerarquía Productos Prodcto	Indicador Cantidad
Cliente 1	2017	Producto 1	5
Cliente 1	2017	Producto 2	7
Cliente 2	2017	Producto 1	35
Cliente 2	2017	Producto 2	45

El nuevo orden de los Atributos es: **Cliente, Año y Producto**.

Pivot permite realizar las siguientes acciones:

- Mover un Atributo o Indicador desde el encabezado de Fila al encabezado de Columna.
- Mover un Atributo o Indicador desde el encabezado de Columna al encabezado de Fila.
- Cambiar el orden de los Atributos o Indicadores del encabezado de Columna.
- Cambiar el orden de los Atributos o Indicadores del encabezado de Fila.

Page

Page es una operación que presenta el Cubo dividido en secciones, a través de los valores de un Atributo, como si se tratase de páginas de un libro.

Page es muy útil cuando las consultas devuelven muchos registros y es necesario desplazarse por los datos para poder verlos en su totalidad.

Se tomará como referencia, para explicar esta operación, la siguiente tabla:

Dimensión Productos Jerarquía Productos Producto	Dimensión Años Jerarquía Años Año Mes	Indicador Cantidad
Producto 1	2017 Enero	22
Producto 1	2017 Febrero	18
Producto 2	2017 Enero	33
Producto 2	2017 Febrero	19

Se realizará Page sobre el Atributo **Producto**, entonces se obtendrán las siguientes páginas:

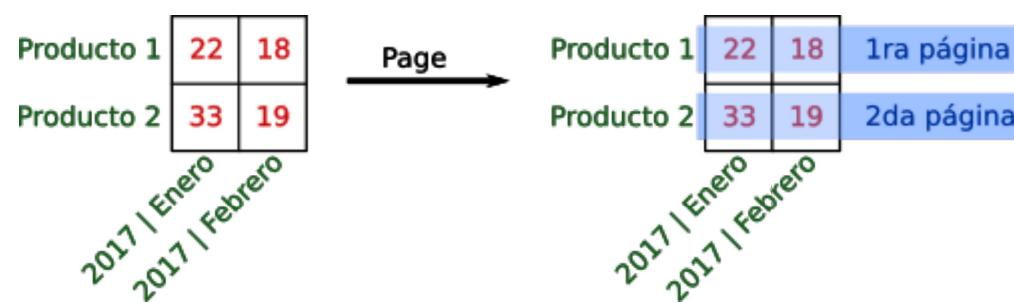
- 1ra página, correspondiente a **Producto 1**:

Dimensión Años Jerarquía Años Año Mes	Indicador Cantidad
2017 Enero	22
2017 Febrero	18

- 2da página, correspondiente a **Producto 2**:

Dimensión Años Jerarquía Años Año Mes	Indicador Cantidad
2017 Enero	33
2017 Febrero	19

Matricialmente se representa de la siguiente manera:



Drill-through

Drill-through es una operación que permite visualizar cuáles son los datos relacionados al valor de un Indicador.

Los datos se mostrarán en su máximo nivel de detalle.

Se tomará como referencia, para explicar esta operación, la siguiente tabla:

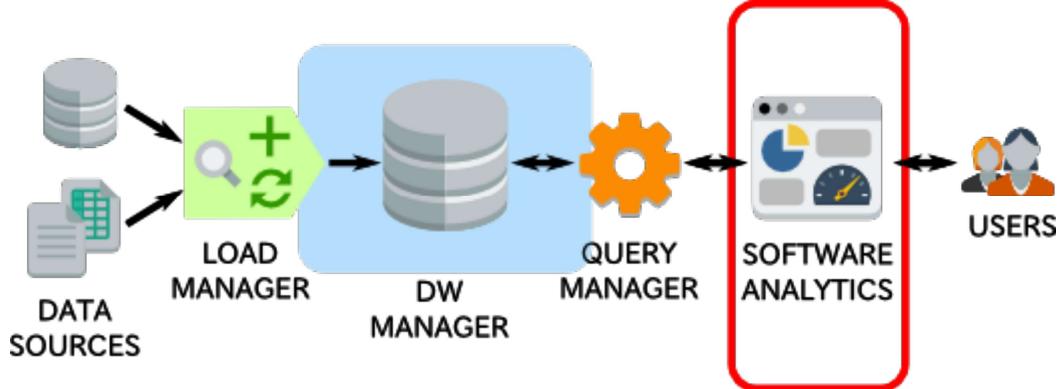
Dimensión Productos Jerarquía Productos Producto	Dimensión Años Jerarquía Años Año	Indicador Cantidad
Producto 1	2017	40
Producto 2	2017	52

Se aplicará la operación drill-through sobre el Indicador de la fila seleccionada, para obtener el detalle de este valor:

Dimensión Años Jerarquía Años Año Mes Día	Dimensión Clientes Jerarquía Clientes Cliente	Dimensión Productos Jerarquía Productos Producto	Dimensión Vendedores Jerarquía Vendedores Vendedor	Indicador Cantidad
2017 Enero 1	Cliente 1	Producto 1	Vendedor 1	3
2017 Enero 5	Cliente 1	Producto 1	Vendedor 1	2
2017 Enero 10	Cliente 2	Producto 1	Vendedor 1	10
2017 Enero 18	Cliente 2	Producto 1	Vendedor 1	6
2017 Enero 21	Cliente 2	Producto 1	Vendedor 1	1
2017 Febrero 4	Cliente 2	Producto 1	Vendedor 1	5
2017 Febrero 12	Cliente 2	Producto 1	Vendedor 1	4
2017 Febrero 16	Cliente 2	Producto 1	Vendedor 1	5
2017 Febrero 23	Cliente 2	Producto 1	Vendedor 1	3
2017 Febrero 24	Cliente 2	Producto 1	Vendedor 1	1

5) Software Analytics

Cuando hablamos de Software Analytics, nos referimos a todas aquellas herramientas de software mediante las cuales podremos explorar y explotar los datos almacenados en el DW.



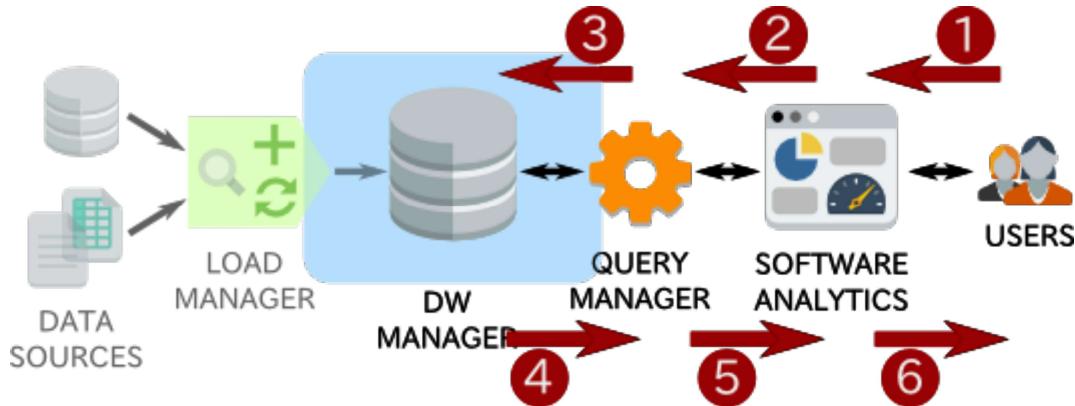
Estas herramientas constituyen el nexo entre el DW y l@s usuari@s, y son la parte visible y/o tangible del proceso de Data Warehousing.

Para la obtención de datos del DW, se utiliza principalmente:

- la metadata de las estructuras de datos que han sido creadas previamente (Cubos Multidimensionales, Business Models, etc.) y
- conexiones a bases de datos (JNDI, JDBC, ODBC)

Interacción

Cada vez que el usuari@ interactúa con el Software Analytic para explorar los datos del DW se llevan a cabo los siguientes pasos generales:



1. L@s Users seleccionan o establecen qué datos desean obtener del DW, mediante la GUI (interfaz gráfica) del Software Analytics.
2. El Software Analytics procesa el pedido de l@s Users, construye las consultas (utilizando la metadata) y las envía al Query Manager.
3. El Query Manager ejecuta las consultas sobre la estructura de datos con la que se esté trabajando (Cubo Multidimensional, Business Model, etc.).
4. El Query Manager obtiene los resultados de las consultas.
5. El Query Manager envía los datos al Software Analytics.
6. El Software Analytics presenta a l@s Users los datos requeridos.

Características

La mayoría de los Software Analytics comparten las siguientes características:

-  Accesibilidad a la información: el acceso a la información es transparente a l@s usuari@s finales. Esto se realiza a través de diferentes estructuras de datos permitiendo al usuari@ enfocarse exclusivamente en el análisis, sin preocuparse del origen y procedencia de los datos.
-  Apoyo en la toma de decisiones: permiten la exploración de los datos, a fin de seleccionar, filtrar y personalizar los mismos, para la obtención de información oportuna, relevante y útil, para apoyar el proceso de toma de decisiones.
-  Orientación a l@s usuari@s finales: disponen de GUIs (interfaz gráfica) amigables e intuitivas, que permiten a l@s usuari@s realizar análisis y consultas complejas, sin poseer conocimientos técnicos. Si bien los activos más valiosos son los datos en si, solo se podrán interpretar y analizar en la medida en que sean correctamente presentados.

Existen diferentes tipos de herramientas de consulta y análisis, y de acuerdo a la necesidad, tipos de usuari@s y requerimientos de información, se deberán seleccionar las más apropiadas al caso. Entre ellas se destacan las siguientes.

- Reporting
- OLAP
- Dashboards
- Data Mining
- EIS

Reporting

Las herramientas de Reporting ofrecen a l@s usuari@s, a través de pantallas gráficas intuitivas, la posibilidad de generar informes avanzados y detallados del tema de interés que se esté analizando. L@s usuari@s sólo deben seguir una serie de simples pasos, como por ejemplo seleccionar opciones de un menú, presionar tal o cual botón para especificar los elementos de datos, sus condiciones, criterios de agrupación y demás características que se consideren significativas. A continuación se presentan ejemplos de reportes:



DIA	CANTIDAD	TOTAL
2 (martes)	1.890	\$ 116.871.833
6 (sabado)	2.724	\$ 164.729.408
7 (domingo)	1.553	\$ 124.876.056
8 (lunes)	914	\$ 40.164.602
9 (martes)	1.058	\$ 91.513.364
13 (sabado)	770	\$ 20.669.667
18 (jueves)	1.200	\$ 74.341.779
19 (viernes)	477	\$ 14.837.290
21 (domingo)	1.623	\$ 197.417.892
23 (martes)	1.686	\$ 74.243.188
28 (domingo)	4.925	\$ 320.472.957

PRODUCTOS	RUBROS	TOTAL	INDICADOR
PRODUCTO 105	Rubro 7	\$ 47.427	🔴
PRODUCTO 107	Rubro 7	\$ 8.805.000	🟡
PRODUCTO 108	Rubro 3	\$ 9.099.984	🟢
PRODUCTO 110	Rubro 4	\$ 261.238	🟡
PRODUCTO 117	Rubro 9	\$ 650.180	🟡
PRODUCTO 118	Rubro 9	\$ 17.822.403	🟢
PRODUCTO 122	Rubro 2	\$ 3.541.584	🟡

PRODUCTO	RUBRO	VALOR	ESTADO
PRODUCTO 124	Rubro 9	\$ 3.690.168	
PRODUCTO 125	Rubro 1	\$ 22.415.368	
PRODUCTO 127	Rubro 8	\$ 458.834	
PRODUCTO 129	Rubro 4	\$ 23.192.623	
PRODUCTO 13	Rubro 3	\$ 24.745.322	
PRODUCTO 131	Rubro 10	\$ 472.428	

Actualmente las herramientas de Reporting cuentan con muchas prestaciones, las cuales permiten dar variadas formas y formatos a la presentación de la información. Entre las opciones más comunes se encuentran las siguientes:

- Parametrización de los datos devueltos.
- Selección de formatos de salida (planilla de cálculo, HTML, PDF, etc.).
- Inclusión de bar charts, pie charts, sparklines, etc.
- Utilización de plantillas de formatos de fondos.
- Inclusión de imágenes.
- Formatos tipográficos.
- Links a otros reportes.

OLAP

OLAP (On Line Analytic Processing) es el componente más poderoso del Data Warehousing, ya que contiene el motor de consultas multidimensionales especializado del DW.

Su principal objetivo es el de brindar respuestas rápidas a complejas preguntas, para interpretar la situación del negocio y tomar decisiones. Cabe destacar que lo que es realmente interesante en OLAP, NO es la ejecución de simples consultas tradicionales, sino la posibilidad de utilizar operadores tales como Drill-up, Drill-down, etc.

Se presentan aquí ejemplos de consultas OLAP:

Markets			Measures	
(All)	Territory	Country	Sales	Quantity
All Markets	APAC		1,281,706	12,878
	APAC	Australia	630,623	6,246
		New Zealand	535,584	5,396
		Singapore	115,499	1,236
	EMEA		5,008,224	49,578
	Japan		503,958	4,923
	Japan	Hong Kong	48,784	596
		Japan	188,168	1,842
		Philippines	94,016	961
		Singapore	172,990	1,524
NA			3,852,061	37,952
NA	Canada		224,079	2,293
	USA		3,627,983	35,659
Total			10,645,949	105,331

	2003	2004	2005	
Territory	Sales	Sales	Sales	
APAC	343.082	601.606	337.018	
EMEA	1.681.987	2.396.408	929.829	
Japan	292.558	168.479	42.921	
NA	1.359.757	1.821.247	671.057	

Además, a través de este tipo de herramientas, se puede analizar el negocio desde diferentes escenarios históricos, y proyectar el comportamiento y evolución desde una visión multidimensional, o sea, mediante la combinación de diferentes Perspectivas, temas de interés o Dimensiones. Esto

permite deducir tendencias, por medio del descubrimiento de relaciones entre las Perspectivas, que a simple vista serían difíciles de encontrar.

Las herramientas OLAP requieren que los datos estén organizados dentro del DW en forma multidimensional, por ello se utilizan las estructuras denominadas Cubos Multidimensionales.

Dashboards

Los Dashboards se pueden entender como una colección de componentes gráficos de análisis como: reportes, tablas, gráficos, consultas y análisis interactivos, etc; que hacen referencia a un tema en particular y que están relacionados entre sí. Por ejemplo:



Existen diversas maneras de diseñar un Dashboard, cada una de las cuales tiene sus objetivos particulares, pero a modo de síntesis se expondrán algunas características generales que poseen:

- Presentan la información altamente resumida.
- Se componen de consultas, reportes, análisis interactivos, charts (bar, pie, line, etc), semáforos, Indicadores causa-efecto, etc.
- Permiten evaluar la situación de la empresa con un solo golpe de vista.
- Poseen un formato de diseño visual muy llamativo.