

# ¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.  
 Ese apoyo que siempre quiso tener...

## 1. Desarrollo de componentes y proyectos a medida



## 2. Auditoría de código y recomendaciones de mejora

## 3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



## 4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,  
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)  
 Gestor de contenidos (Alfresco)  
 Aplicaciones híbridas

Tareas programadas (Quartz)  
 Gestor documental (Alfresco)  
 Inversión de control (Spring)

Control de autenticación y  
 acceso (Spring Security)  
 UDDI  
 Web Services  
 Rest Services  
 Social SSO  
 SSO (Cas)

JPA-Hibernate, MyBatis  
 Motor de búsqueda empresarial (Solr)  
 ETL (Talend)

Dirección de Proyectos Informáticos.  
 Metodologías ágiles  
 Patrones de diseño  
 TDD

BPM (jBPM o Bonita)  
 Generación de informes (JasperReport)  
 ESB (Open ESB)



CoNcept

Lanzado

**TNTConcept versión 0.6** ( 12/07/2007)

Desde [Autentia](#) ponemos a vuestra disposición el software que hemos construido (100% gratuito y sin restricciones funcionales) para nuestra gestión interna, llamado TNTConcept (auTeNTia).

Construida con las últimas tecnologías de desarrollo Java/J2EE (Spring, JSF, Acegi, Hibernate, Maven, Subversion, etc.) y disponible en licencia GPL, seguro que a muchos profesionales independientes y PYMES os ayudará a organizar mejor vuestra operativa.

**Las cosas grandes empiezan siendo algo pequeño** ..... Saber más en: <http://tntconcept.sourceforge.net/>

 <p><b>Autor del tutorial:</b> <i>Cristhian Kirs Herrera Basurto</i></p> <p>§ <b>Lugar de residencia:</b> Quito - Ecuador</p> <p><i>Cuento con experiencia en el área de desarrollo de software y en la docencia académica. Dentro de la construcción de software he manejado las etapas de: análisis, diseño, personalización e implementación de aplicaciones bajo ambientes Cliente / Servidor e Internet.</i></p> <p><a href="mailto:Cristhian.Herrera@gmail.com">Cristhian.Herrera@gmail.com</a> / <a href="mailto:cherrera@kruger.com.ec">cherrera@kruger.com.ec</a></p>	<p><b>NUEVO CATÁLOGO DE SERVICIOS DE AUTENTIA (PDF 6,2MB)</b></p> <p><a href="http://www.adictosaltrabajo.com">www.adictosaltrabajo.com</a> es el Web de difusión de conocimiento de <a href="http://www.autentia.com">www.autentia.com</a></p>  <p><b>autentia</b> real business solutions</p> <p><a href="#">Catálogo de cursos</a></p>
---	--

Descargar este documento en formato PDF [datawarehouse2.pdf](#)

[Firma en nuestro libro de Visitas](#) <-----> [Asociarme al grupo AdictosAlTrabajo en eConozco](#)

## Trabaje desde casa - Ingresos extra desde casa.

Tiempo completo o parcial. [www.trabajaportucuenta.com](http://www.trabajaportucuenta.com)

Anuncios Google

Anunciarse en este sitio

**Fecha de creación del tutorial: 2007-10-30**

Apuntes Datawarehouse

[Viene de la 1ª parte](#)

[Diferencia entre OLTP y Datawarehouse 12](#)

[Ciclo de Desarrollo 14](#)

[Consideraciones previas al desarrollo de un Datawarehouse 15](#)

[Alcance de un Data Warehouse 16](#)

[Redundancia de Datos 16](#)

[Data Warehouses "Virtual" o "Point to Point" 17](#)

[Data Warehouses Distribuidos 17](#)

[Modelo de Planificación para un Datawarehouse 18](#)

[Requerimientos del Negocio 18](#)

[Misión 19](#)

[Decisiones 19](#)

[Decisiones Operativas 19](#)

[Decisiones Tácticas 19](#)

[Decisiones Estratégicas 20](#)[Usuarios 20](#)[Administradores 21](#)[Autores 21](#)[Activos 21](#)[Casuales 21](#)[Decisiones clasificadas por clase de usuario, tipo y función de negocio 21](#)[Requerimientos de Información 22](#)

## **Diferencia entre OLTP y Datawarehouse**

Los sistemas tradicionales de transacciones y las aplicaciones de Data Warehousing son polos opuestos en cuanto a sus requerimientos de diseño y sus características de operación. Es de suma importancia comprender perfectamente estas diferencias para evitar caer en el diseño de un Data Warehouse como si fuera una aplicación de transacciones en línea (OLTP).

Las aplicaciones de OLTP están organizadas para ejecutar las transacciones para los cuales fueron hechos, como por ejemplo: mover dinero entre cuentas, un cargo o abono, una devolución de inventario, etc. Por otro lado, un Data Warehouse está organizado en base a conceptos, como por ejemplo: clientes, facturas, productos, etc.

Otra diferencia radica en el número de usuarios. Normalmente, el número de usuarios de un Data Warehouse es menor al de un OLTP. Es común encontrar que los sistemas transaccionales son accesados por cientos de usuarios simultáneamente, mientras que los Data Warehouse sólo por decenas. Los sistemas de OLTP realizan cientos de transacciones por segundo mientras que una sola consulta de un Data Warehouse puede tomar minutos. Otro factor es que frecuentemente los sistemas transaccionales son menores en tamaño a los Data Warehouses, esto es debido a que un Data Warehouse puede estar formado por información de varios OLTP's.

Existen también diferencia en el diseño, mientras que el de un OLPT es extremadamente normalizado, el de un Data Warehouse tiende a ser desnormalizado. El OLTP normalmente está formado por un número mayor de tablas, cada una con pocas columnas, mientras que en un Data Warehouse el número de tablas es menor, pero cada una de éstas tiende a ser mayor en número de columnas.

Los OLTP son continuamente actualizados por los sistemas operacionales del día con día, mientras que los Data Warehouse son actualizados en batch de manera periódica.

Las estructuras de los OLTP son muy estables, rara vez cambian, mientras las de los Data Warehouses sufren cambios constantes derivados de su evolución. Esto se debe a que los tipos de consultas a los cuales están sujetos son muy variados y es imposible preverlos todos de antemano

### **OLPT**

- Orientada a transacciones
- Detallada
- Actualizada en línea
- Usuarios de nivel operativo
- Corre en base a repeticiones
- Muy sensitivo al desempeño
- Accesa unidades a la vez
- Orientado a operación
- Estructura estática

### **Data Warehouse**

- Orientada a Conceptos
- Sumarizada
- Representa valores a un tiempo (snapshot)
- Usuarios de nivel gerencial
- Corre heurísticamente
- Poco sensitivo al desempeño
- Accesa conjuntos de unidades a la vez
- Orientado a análisis
- Estructura flexible

- Sin redundancia
- Alta probabilidad de acceso
- Administrada como un todo
- Información bruta (Datos)
- Actualizada en línea
- Muchas tablas con pocas columnas

- Con mucha redundancia
- Modesta probabilidad de acceso
- Administrada por partes
- Información procesada (Información)
- Actualizada en Batch
- Pocas tablas con muchas columnas

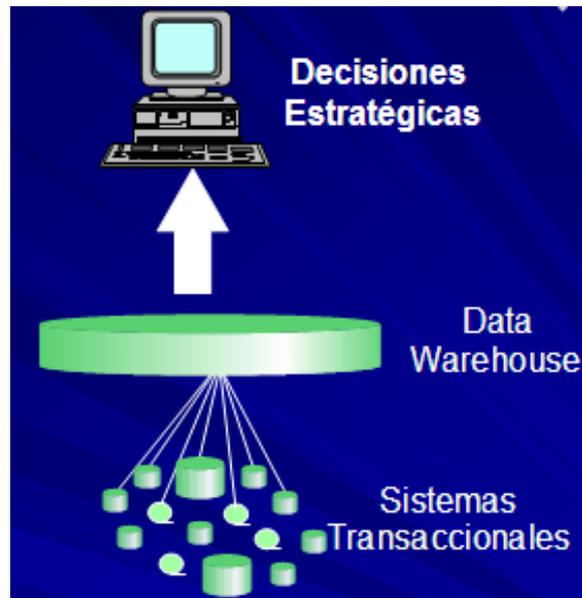
La siguiente es una comparación entre un Datawarehouse y los Sistemas Transaccionales tradicionales.

#### Sistema Transaccional



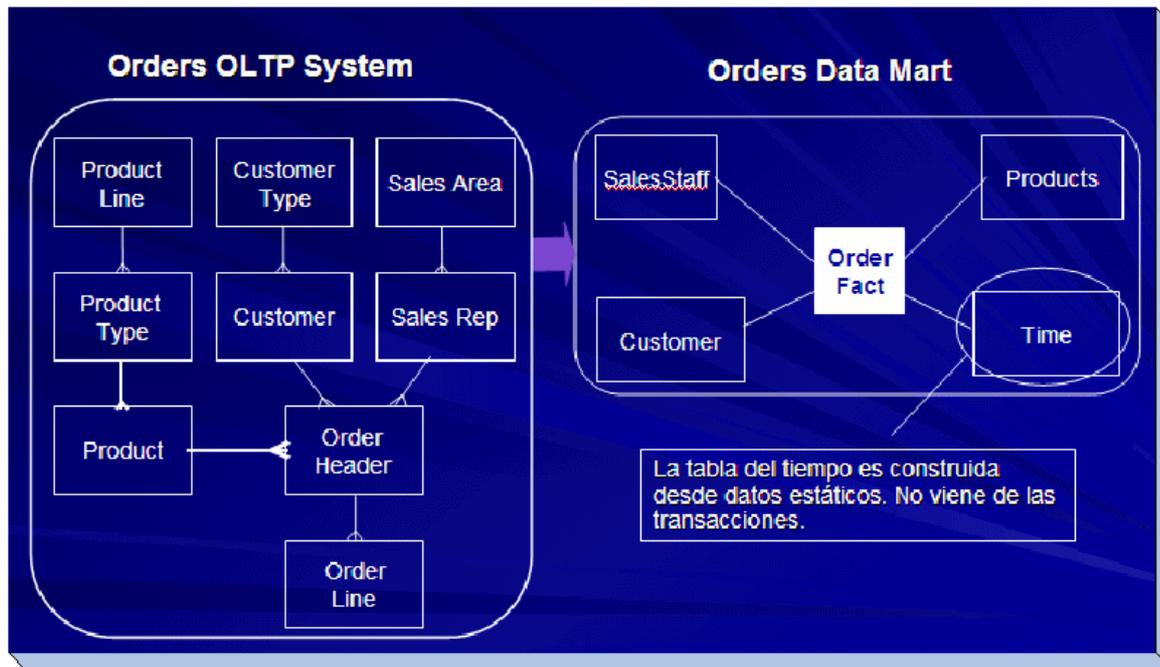
- No datos sumariados
- No drill down
- No datos históricos
- Aplicaciones no integradas

#### Datawarehouse



- Muchos datos sumariados
- Estructurado para análisis con drill down
- Datos históricos para análisis de tendencias
- Información integrada para análisis corporativos

A continuación un modelo expresado en su forma transaccional y también modelado como Data Mart.

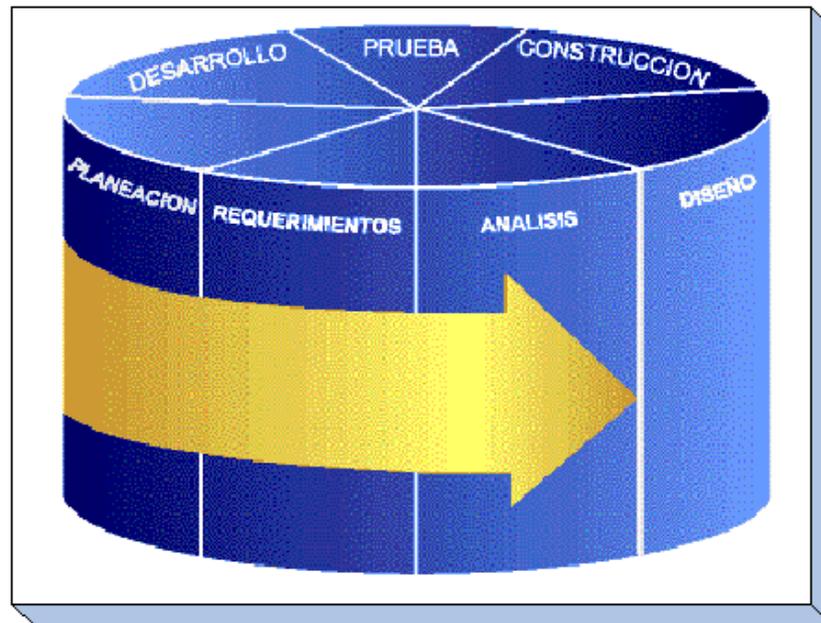


**Ciclo de Desarrollo**

El Data Warehouse sigue el mismo ciclo de perfeccionamiento que todos los desarrollos de software.



Las fases del ciclo son las mismas, lo mismo que su secuencia, sólo existen variantes únicas que se relacionan específicamente con el Data Warehouse para tareas dentro de estas fases. La siguiente figura muestra el ciclo clásico de desarrollo de software:



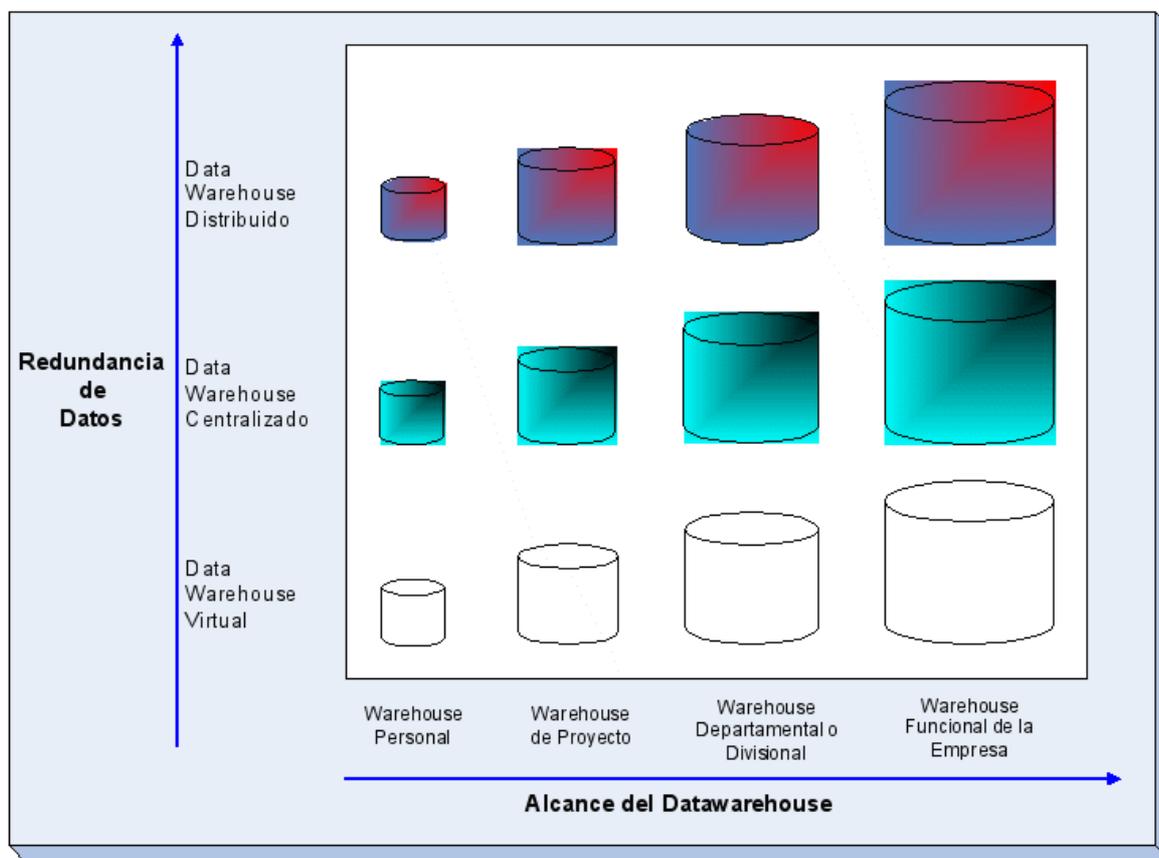
- Planeación:** La planeación es una fase importante de la implementación del Data Warehouse. Las decisiones tomadas durante la fase de planeación tienen un impacto significativo en el ámbito de implementación y en la magnitud del esfuerzo. Las decisiones clave de planeación incluyen la selección de un enfoque de arriba hacia abajo (de lo general a lo particular), de abajo hacia arriba (en sentido opuesto) o combinado; la selección de la arquitectura apropiada de Data Warehouse; la selección adecuada del ámbito de información, fuentes de datos y tamaño del metamodelo; y la estimación de planes de programa y proyecto y justificaciones de presupuesto.
- Requerimientos:** Durante la fase de requerimientos se debe considerar una diversidad de ellos. Los requerimientos son conducidos por el negocio y por la tecnología. La cuidadosa selección y especificación de requerimientos en esta etapa proporciona un proyecto cimentado que arroja resultados con rapidez.
- Análisis:** La fase de análisis es importante ya que determina la forma en que se cubrirán los requerimientos. Esta fase se enfoca principalmente en la conversión de especificaciones de requerimientos a especificaciones de metamodelo para el Data Warehouse. Después, estas especificaciones se usan para generar extractores del Data Warehouse y software de transformación, integración, resumen y adición.
- Construcción:** La fase de construcción resalta los diversos intercambios "construir en comparación con comprar". Mediante la selección adecuada de componentes suministrados por fabricantes, es posible construir una primera implementación del Data Warehouse rápida y eficaz.
- Despliegue:** La fase de despliegue en el ciclo de desarrollo del Data Warehouse tiene un componente único denominado comercialización de información. Esto reconoce que la mercancía que suministra el Data Warehouse a sus usuarios finales (clientes) es la propia información. Como un producto de mercancía, la información también debe comercializarse como los bienes de consumo. La comercialización comprende la capacidad de hacer énfasis en la disponibilidad, los beneficios y el empaque para hacerla atractiva al usuario final.

## Consideraciones previas al desarrollo de un Datawarehouse

Hay muchas maneras para desarrollar data warehouses como tantas organizaciones existen. Sin embargo, hay un número de dimensiones diferentes que necesitan ser consideradas:

- Alcance de un data warehouse
- Redundancia de datos

La Figura muestra un esquema bidimensional para analizar las opciones básicas. La dimensión horizontal indica el alcance del depósito y la vertical muestra la cantidad de datos redundantes que deben almacenarse y mantenerse.



### Alcance de un Data Warehouse

El alcance de un data warehouse puede ser tan amplio como toda la información estratégica de la empresa desde su inicio, o puede ser tan limitado como un data warehouse personal para un solo gerente durante un año.

En la práctica, en la amplitud del alcance, el mayor valor del data warehouse es para la empresa y lo más caro y consumidor de tiempo es crear y mantenerlo. Como consecuencia de ello, la mayoría de las organizaciones comienzan con data warehouses funcionales, departamentales o divisionales y luego los expanden como usuarios que proveen retroalimentación.

## Redundancia de Datos

Hay tres niveles esenciales de redundancia de datos que las empresas deberían considerar en sus opciones de data warehouse:

- Data warehouses "virtual" o "Point to Point"
- Data warehouses "centrales"
- Data warehouses "distribuidos"

No se puede pensar en un único enfoque. Cada opción adapta un conjunto específico de requerimientos y una buena estrategia de almacenamiento de datos, lo constituye la inclusión de las tres opciones.

### *Data Warehouses "Virtual" o "Point to Point"*

Una estrategia de data warehouses virtual, significa que los usuarios finales pueden acceder a bases de datos operacionales directamente, usando cualquier herramienta que posibilite "la red de acceso de datos".

Este enfoque provee flexibilidad así como también la cantidad mínima de datos redundantes que deben cargarse y mantenerse. Además, se pueden colocar las cargas de consulta no planificadas más grandes, sobre sistemas operacionales.

Como se verá, el almacenamiento virtual es, frecuentemente, una estrategia inicial, en organizaciones donde hay una amplia (pero en su mayor parte indefinida) necesidad de conseguir la data operacional, desde una clase relativamente grande de usuarios finales y donde la frecuencia probable de pedidos es baja.

Los depósitos virtuales de datos proveen un punto de partida para que las organizaciones determinen qué usuarios finales están buscando realmente.

### *Data Warehouses "Centrales"*

El concepto de data warehouses centrales es el concepto inicial que se tiene del data warehouse. Es una única base de datos física, que contiene todos los datos para un área funcional específica, departamento, división o empresa.

Los data warehouses centrales se seleccionan por lo general donde hay una necesidad común de los datos informáticos y un número grande de usuarios finales ya conectados a una red o computadora central. Pueden contener datos para cualquier período específico de tiempo. Comúnmente, contienen datos de sistemas operacionales múltiples.

Los data warehouses centrales son reales. Los datos almacenados en el data warehouse son accesibles desde un lugar y deben cargarse y mantenerse sobre una base regular. Normalmente se construyen alrededor de RDBMS avanzados o, en alguna forma, de servidor de base de datos informático multidimensional.

### ***Data Warehouses Distribuidos***

Los data warehouses distribuidos son aquellos en los cuales ciertos componentes del depósito se distribuyen a través de un número de bases de datos físicas diferentes.

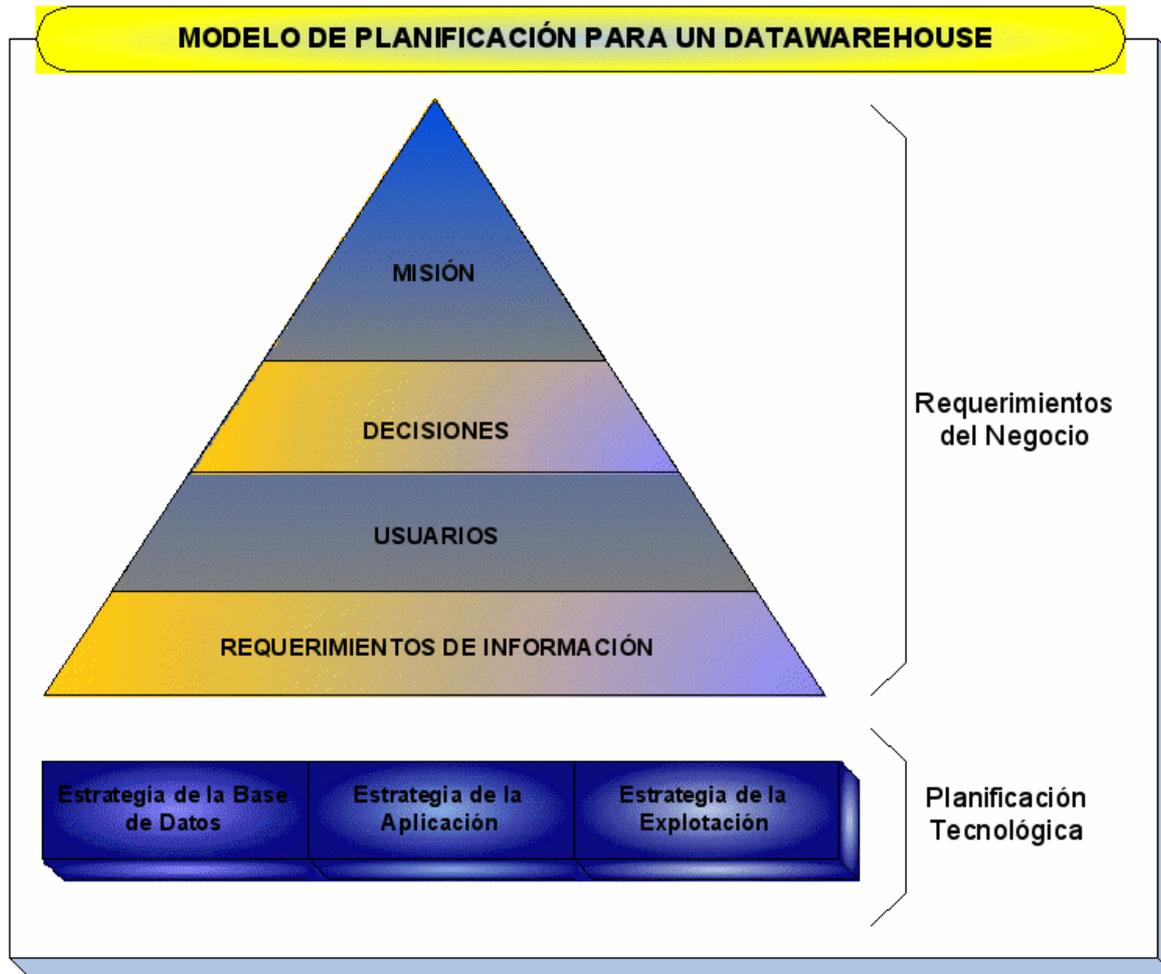
Cada vez más, las organizaciones grandes están tomando decisiones a niveles más inferiores de la organización y a la vez, llevando los datos que se necesitan para la toma de decisiones a la red de área local (Local Area Network - LAN) o computadora local que sirve al que toma decisiones.

Los data warehouses distribuidos comúnmente involucran la mayoría de los datos redundantes y como consecuencia de ello, se tienen procesos de actualización y carga más complejos.

### ***Modelo de Planificación para un Datawarehouse***

El proceso de planificación de un DW/DM debe considerar:

- La diversidad de roles y responsabilidades de los usuarios
- El rápido crecimiento de la población de usuarios, especialmente usuarios no técnicos
- El mejoramiento constante de las habilidades técnicas de los usuarios
- La evolución de procesos de toma de decisión que son basados en comunicaciones y colaboración extensiva



## Requerimientos del Negocio

El diseño de un DW/DM debe ser consistente con la dirección estratégica de la organización; por lo tanto el proceso de planificación debería empezar en el nivel de alta gerencia con una definición completa de los requerimientos del negocio. Contempla 3 capas:

- Revisión de la misión corporativa, sus metas y objetivos
- Identificando puntos de decisión y preguntas necesarias
- Identificando a los usuarios

## Misión

- La especificación de la misión corporativa debe definir las metas de la empresa en términos de negocios y haciéndolo así, también define un claro propósito para invertir en datawarehouse.
- La especificación de la misión del negocio debe establecer metas específicas para ayudar a alcanzar el crecimiento del negocio. Ejemplos de metas
- Llegar a ser un productor de bajo costo
- Expandir geográficamente el mercado

- Expandir el tamaño del mercado
- Entrar en nuevos mercados
- Construir mercados compartidos

Por ejemplo, la meta para expandir geográficamente debe traducirse hacia un objetivo específico de incremento del beneficio desde las operaciones internacionales en un 25 %. Mientras más claro y resumido sean establecidos la misión, las metas y los objetivos del negocio, será mas fácil alinear el diseño del datawarehouse y de las aplicaciones con la misión de la organización.

## **Decisiones**

- Un enfoque útil para las necesidades de los usuarios es empezar definiendo las decisiones que ellos deben realizar
- Un datawarehouse debe ser diseñado para soportar tres tipos de decisiones
- Operacional
- Táctica
- Estratégica

### ***Decisiones Operativas***

Son realizadas cada día, generalmente por los ejecutivos de primera línea. Estas decisiones tienden a confiar en análisis repetitivo de nuevos datos y generalmente requieren sistemas de reportes y análisis de datos que son altamente orientados para un aspecto específico del negocio. Los tomadores de decisiones operacionales desean información como contenido, ya procesado, para la ayuda en la toma de decisión apropiada, mas que una herramienta analítica que puede ser usada para generar contenido.

Un ejemplo de decisión operativa es la facultad que tiene el encargado o gerente de un almacén de hacer nuevos pedidos de productos cuyo inventario está a punto de agotarse ó a la inversa devolver aquellos productos que no tienen el nivel de ventas esperado en lugar de esperar a que estos lleguen a su fecha de caducidad.

### ***Decisiones Tácticas***

Requieren mayor flexibilidad que las decisiones operacionales en términos de acceso a datos y las capacidades analíticas provistas a los usuarios.

Aunque aún muy centradas en entregar eficientemente contenido a los usuarios, aplicaciones de soporte a la decisión táctica también proveen a los usuarios con capacidades analíticas asociadas con el contenido.

Por ejemplo, una aplicación de soporte a la decisión táctica debe proveer al usuario con

un reporte que le permita desglosar en las dimensiones claves del reporte o añadir elementos a la dimensión.

Una decisión táctica típica puede recaer en el hecho de poner productos en oferta cuando su fecha de expiración se acerca ó cuando son productos nuevos y se requiere de llamar la atención de los posibles clientes.

### *Decisiones Estratégicas*

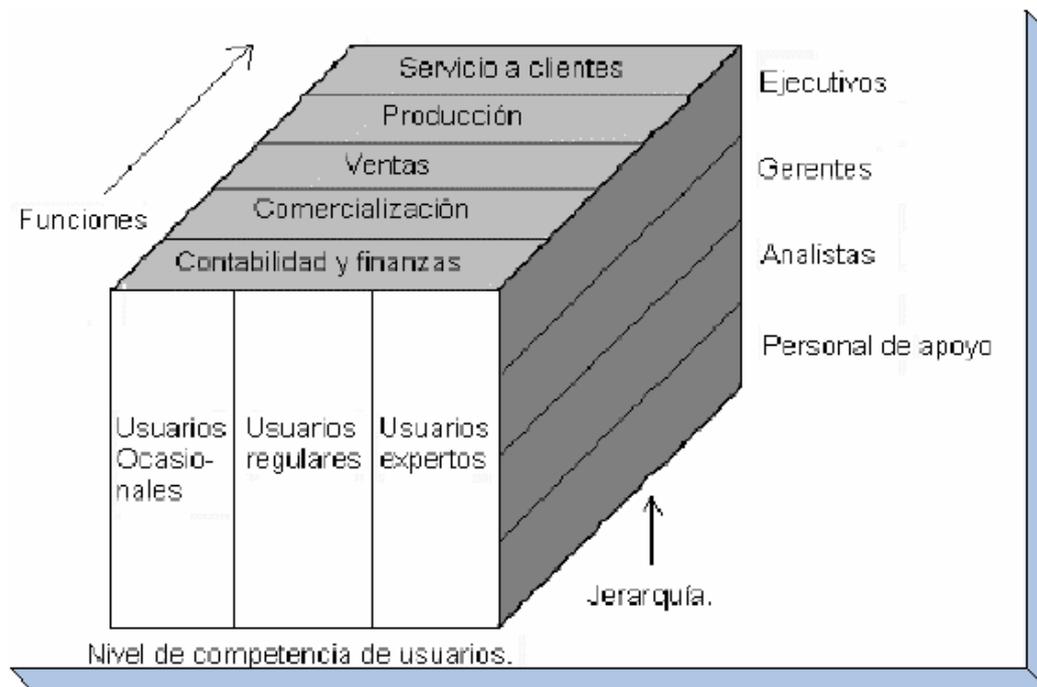
Requieren decisiones ad hoc de los datos contenidos en el warehouse, así como también recursos de datos que no son manejados como parte del datawarehouse. Soporte de decisiones estratégicas se centran en proveer a los usuarios con herramientas potentes que pueden ser usadas para crear contenido

Estas decisiones están orientadas a la gerencia alta de un negocio y tienen que ver con la orientación o destino que se requiere dar al negocio, algunos ejemplos de este tipo de decisión incluyen la apertura de nuevas sucursales de un negocio, la aplicación de estudios de mercado para el lanzamiento de nuevos productos, etc.

### **Usuarios**

El siguiente paso en la planificación del datawarehouse es identificar los varios grupos de usuarios responsables de las decisiones operacionales, tácticas y estratégicas.

De la misma forma que hay una gran cantidad de maneras para organizar un data warehouse, es importante notar que también hay una gama cada vez más amplia de usuarios finales.



Comúnmente hay 4 clases de usuarios finales del datawarehouse:

- Administradores (Power Users)
- Autores (Analistas Financieros y de Negocios)
- Usuarios activos (Usuarios de Soporte, de oficina, personal administrativo)
- Usuarios casuales (Ejecutivos y Gerentes)

### *Administradores*

Son los miembros del personal de soporte en Tecnología de Información quienes ejecutan tanto tareas relacionadas con el negocio así como técnicas tales como asegurar la calidad del contenido del datawarehouse y de las aplicaciones que lo accedan, manteniendo el meta dato para informar a los otros usuarios acerca de los cambios en el datawarehouse ó en las aplicaciones.

### *Autores*

- Son quienes desarrollan las aplicaciones del negocio.
- Los autores típicamente buscan las herramientas de análisis de datos más potentes y de riqueza funcional ad hoc disponibles y a menudo tienen las habilidades técnicas para usar estas herramientas óptimamente

### *Activos*

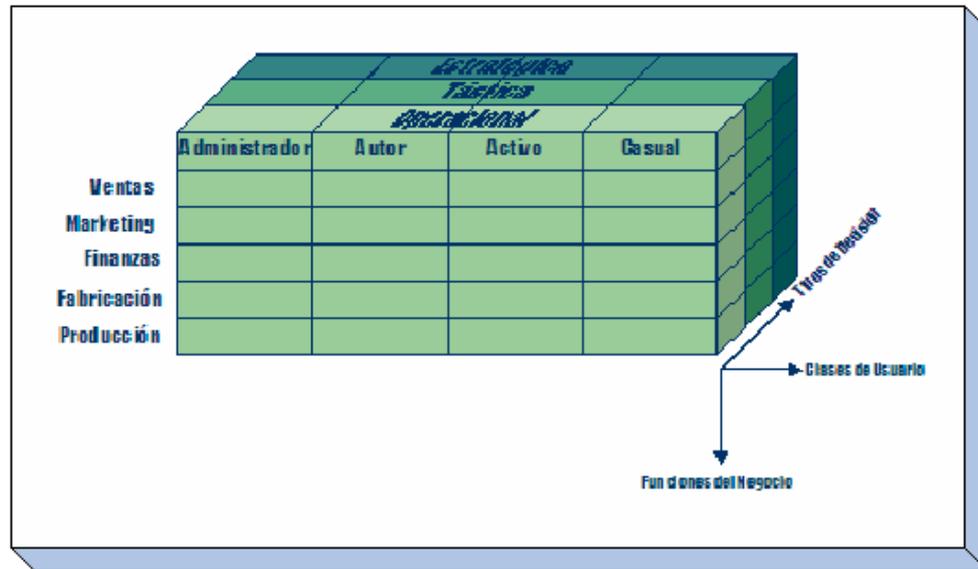
- Son menos hábiles técnicamente que los autores, generalmente gastan mucho tiempo buscando información desde el datawarehouse y analizando las respuestas
- Estos usuarios requieren rápidas y exactas respuestas a los aspectos del negocio. Este grupo tiende a ser la clase de usuarios más impaciente

### *Casuales*

- El grupo menos técnico, son también los usuarios menos frecuentes del datawarehouse, pero el más numeroso.
- Este grupo generalmente incluye ejecutivos de nivel senior y ejecutivos de primera línea, pero a menudo se extiende a todos los niveles de la estructura administrativa de una organización, desde ejecutivos hasta el campo de representantes de ventas
- Debido a que los usuarios casuales, por definición, ingresan infrecuentemente al sistema de datawarehouse, simplicidad es un requerimiento principal de diseño para este grupo. Ellos necesitan comandos y controles que son tanto fáciles de usar y fáciles de recordar

## Decisiones clasificadas por clase de usuario, tipo y función de negocio

El siguiente gráfico muestra la importancia de los usuarios y los tipos de decisiones que ellos deben tomar



## Requerimientos de Información

Hacen referencia a los requerimientos de Tecnología de Información que son necesarios en los emprendimientos de Datawarehouse.

## Planificación Tecnológica

## Estrategia de la Base de Datos

Se trata de la creación de la base de datos. Entre otras cosas incluye

- Contenido: Qué datos e información se requieren para solucionar las preguntas y necesidades de los usuarios
- Fuentes: Cuáles son las fuentes de la información y donde se encuentran las fuentes.
- Extracción: Cómo se extraen los datos y con qué periodicidad se cargan en el datawarehouse.
- Preparación: Qué se requiere para depurar y validar los datos fuentes
- Diseño: Cuál es el diseño apropiado para la base de datos
- Afinamiento: Qué aspectos de afinamiento y rendimiento se van a considerar
- Plataforma: Como será la plataforma en la que residirá el datawarehouse, como se

compone la red, cuales son los componentes de hardware y software.

- Administración: Qué se requiere para administrar el datawarehouse en términos de seguridad, procesos de actualización, gestión de metadatos, aseguramiento de la calidad, etc.

## **Estrategia de la Aplicación**

La estrategia de aplicación trata con la tecnología en dos puntos: la capa de lógica analítica y la capa de presentación. Identificando acceso a los datos y análisis de requerimientos define el conjunto de requerimientos básicos del usuario. Algunas preguntas de los usuarios pueden ser respondidas simplemente recuperando los datos desde el warehouse, pero muchas mas preguntas requieren algún tipo de rutinas analíticas a ser ejecutadas sobre los datos. Estas rutinas analíticas pueden ser clasificadas desde algo tal simple como cálculo del porcentaje de cambio del volumen de ventas hasta la creación de un modelo matemático complejo.

Se identifican las funciones de análisis de datos que se necesitan para satisfacer las necesidades de los usuarios.

- Acceso; Identificar que usuarios van a tener acceso a la información y también que nivel de información podrá ver cada uno de ellos.
- Análisis: Qué funciones de análisis de información serán necesarias para satisfacer los requerimientos.
- Modelamiento; Requerimientos para análisis estadísticos de datos, minería de datos, u otro soporte de modelamiento matemático
- Aplicaciones; Necesidades para aplicaciones específicas del negocio
- Procesos: Cómo ayuda el datawarehouse a los procesos de negocio, Qué mejoras en los procesos de negocio se logran con el datawarehouse.
- Soporte: Cómo los usuarios recibirán soporte y capacitación en el datawarehouse.

## **Estrategia de la Explotación**

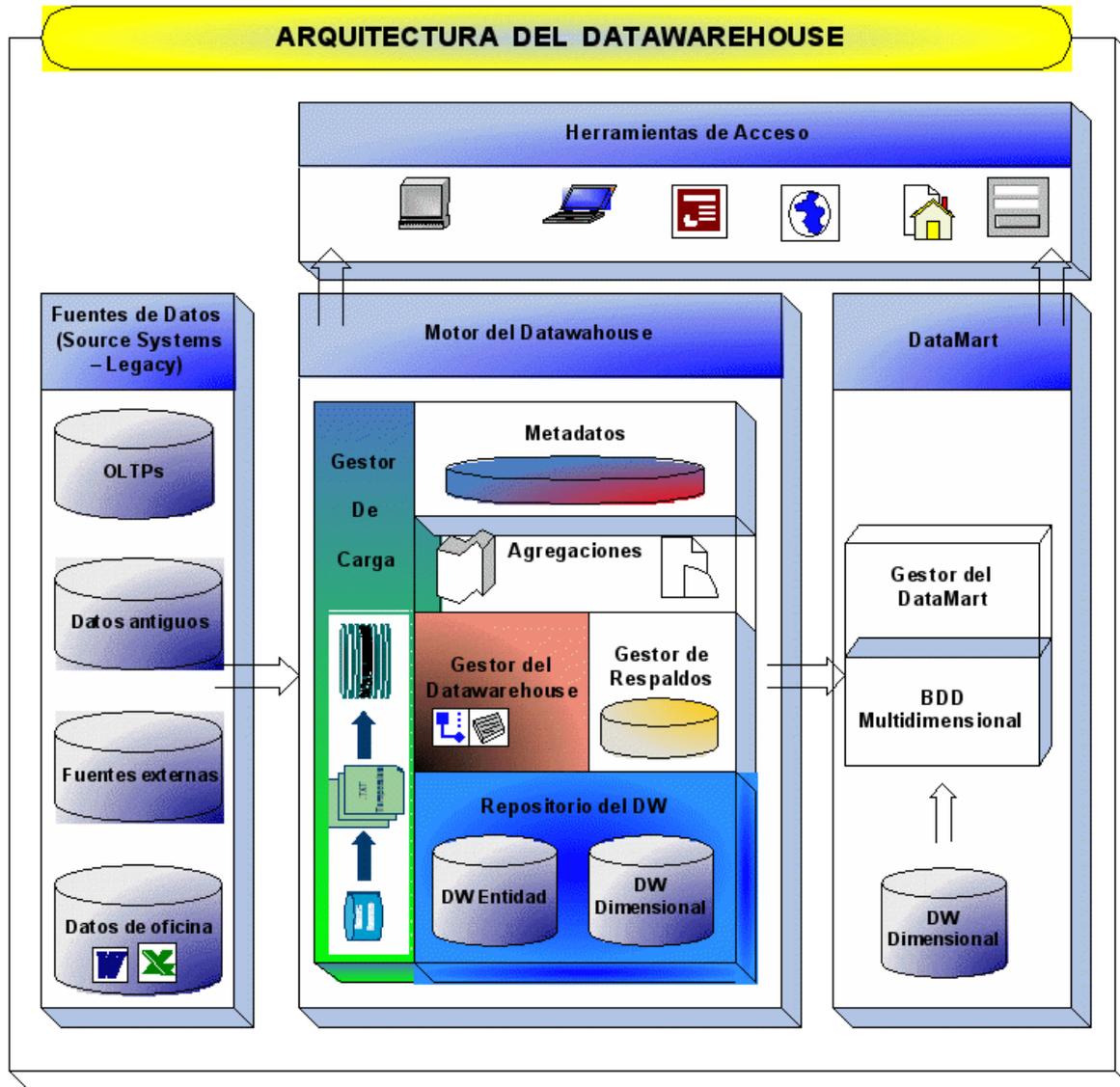
En la estrategia de explotación se consideran los siguientes aspectos

- Interfaz: Cuales usuarios usarán aplicaciones cliente servidor y cuales accederán a través de clientes web (browser)
- Colaboración; Como se promoverá la colaboración entre los usuarios.
- Agentes: Cómo se automatizarán los procesos de análisis y reportes.
- Motor de búsqueda; Cómo los recursos del datawarehouse serán registrados en motores de búsqueda.
- Seguridad: Cómo será garantizada la seguridad de la información y de la base de datos.

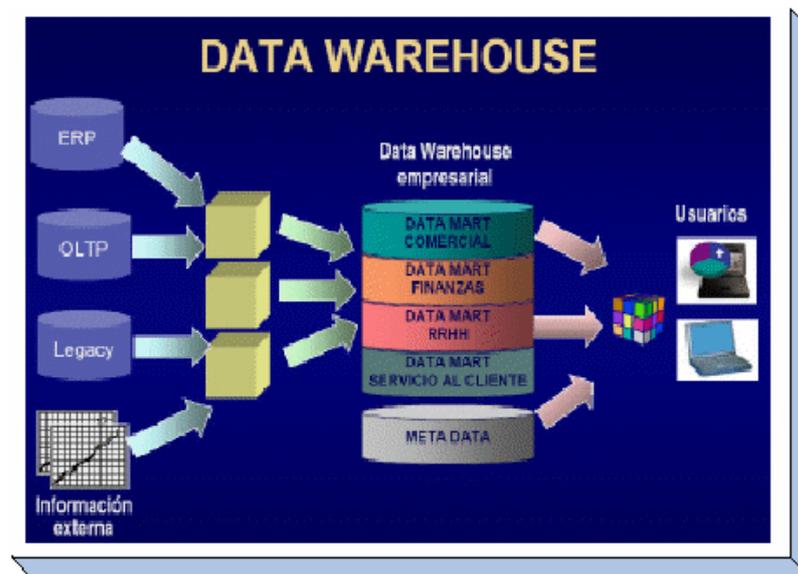
## **Arquitectura del Datawarehouse**

El siguiente gráfico muestra la arquitectura clásica de un Datawarehouse, compuesto por:

- Fuentes de Datos
- Motor del Datawarehouse
  - Gestor de Carga
  - Metadatos
  - Agregaciones
  - Gestor del Datawarehouse
  - Gestor de Respaldos
  - DW Repositorio
- DataMart
  - BDD Dimensional
  - Gestor del DataMart
- Herramientas de Acceso



En forma resumida la arquitectura puede verse expresada en la siguiente figura<sup>2</sup>:



**Fuentes de Datos**

Cualquier origen de información que pueda ser considerado para el datawarehouse, aquí se incluyen los siguientes elementos:

- Los sistemas OLTP's que son los sistemas de Legacy que actualmente operan en la empresa.
- Datos antiguos provenientes de migraciones.
- Fuentes externas como otros sistemas de la compañía, sistemas de otras empresas, sistemas de gobierno, internet, etc.
- Datos de oficina, archivos en formato de Word, Excel, archivos planos, PDFs, mails, etc.

## El motor del datawarehouse

Está integrado por los siguientes componentes

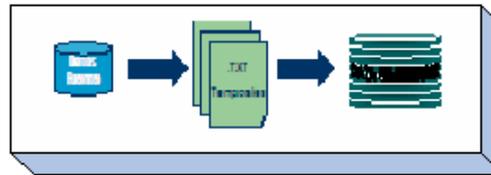
### Gestor de Carga

Quizá sea uno de los elementos más importantes para el datawarehouse, generalmente incluye las operaciones de

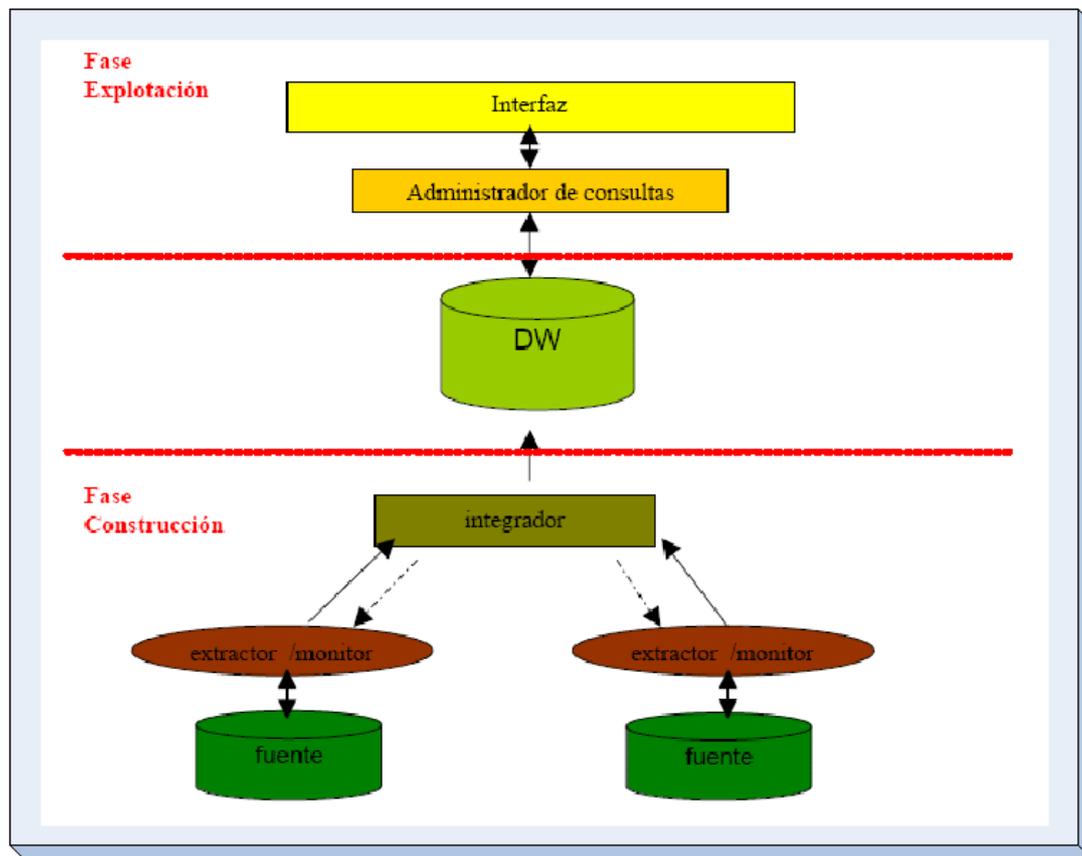
- **Extracción:** Es el proceso que accesa a los datos OLTP existentes, en cualquier forma que exista, desde cualquier DBMS en que exista. Típicamente, extracción y el siguiente paso, propagación, son administrados por el mismo producto. No todas las herramientas de extracción y propagación soportan todas las plataformas, de tal manera que una faceta importante de la selección de herramientas es si la herramienta soporta los sistemas operativos y las bases de datos que se esté usando para el datawarehouse.
- **Propagación:** Es el proceso de mover datos desde los sistemas fuente hacia el sistema objetivo que contendrá el data warehouse. El proceso de propagación toma lugar en tiempo real, o en un calendario predeterminado (batch), o sobre demanda, y puede efectuar un refresco total del warehouse o justo un cambio neto. Cuando se selecciona una herramienta de propagación, se aspira que ésta ofrezca la gestión de cambios netos como también refresco total y permitirá tanto actualizaciones en tiempo real y calendarizadas (batch).
- **Depuración (Limpieza):** El nivel lógico cubre problemas de valores de datos que son inconsistentes dentro de la información importada (ejemplo, clientes con estado casado, pero con una edad de 3 años). El nivel técnico evalúa problemas de información tales como campos no inicializados o valores inválidos en los datos importados (ejemplo, valor de la fecha Febrero 31).
- **Transformación:** Convierte datos desde su formato OLTP al apropiado formato del datawarehouse ejecutando funciones tales como desnormalización de datos, traduciendo códigos hacia texto significativo, convirtiendo una variedad de formatos de fechas hacia un formato estándar, convirtiendo texto tal como nombres de ciudades hacia texto estándar y renombrando campos desde nombres

técnicos no significativos hacia nombres significativos que un usuario final entenderá.

- **Carga:** Los datos fuentes normalmente son extraídos y almacenados en archivos temporales tipo texto, los mismos que deben ser cargados a la base de datos del datawarehouse. La figura resume el proceso de carga, los archivos temporales finalmente son colocados en la base de datawarehouse de destino.



El módulo de Gestor de Carga también es conocido como Integrador, y es muy importante tanto en la Fase de Construcción como en la Fase de Explotación de un DataWarehouse.



### Confiabilidad de los datos<sup>3</sup>

La data "sucia" es peligrosa. Las herramientas de limpieza especializadas y las formas de programar de los clientes proporcionan redes de seguridad.

No importa cómo esté diseñado un programa o cuán hábilmente se use. Si se alimenta

mala información, se obtendrá resultados incorrectos o falsos. Desafortunadamente, los datos que se usan satisfactoriamente en las aplicaciones de línea comercial operacionales pueden ser basura en lo que concierne a la aplicación data warehousing.

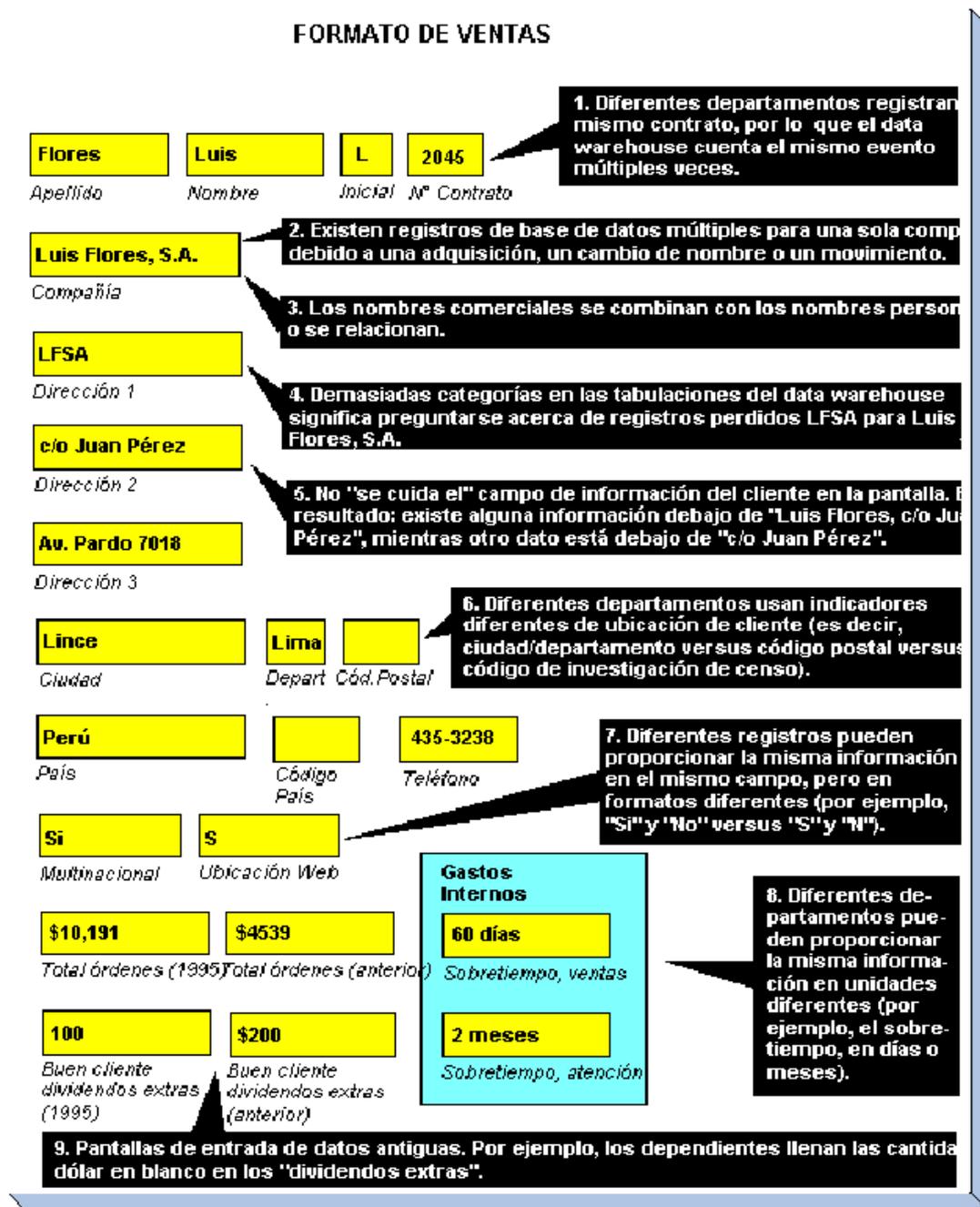
Los datos "sucios" pueden presentarse al ingresar información en una entrada de datos (por ejemplo, "Sistemas S. A." en lugar de "Sistemas S. A.") o de otras causas. Cualquiera que sea, la data sucia daña la credibilidad de la implementación del depósito completo. A continuación, en la Figura se muestra un ejemplo de formato de ventas en el que se pueden presentar errores.

Afortunadamente, las herramientas de limpieza de datos pueden ser de gran ayuda. En algunos casos, puede crearse un programa de limpieza efectivo. En el caso de bases de datos grandes, imprecisas e inconsistentes, el uso de las herramientas comerciales puede ser casi obligatorio.

Decidir qué herramienta usar es importante y no solamente para la integridad de los datos. Si se equivoca, se podría malgastar semanas en recursos de programación o cientos de miles de dólares en costos de herramientas.

La limpieza de una data "sucias" es un proceso multifacético y complejo. Los pasos a seguir son los siguientes:

1. Analizar sus datos corporativos para descubrir inexactitudes, anomalías y otros problemas.
2. Transformar los datos para asegurar que sean precisos y coherentes.
3. Asegurar la integridad referencial, que es la capacidad del data warehouse, para identificar correctamente al instante cada objeto del negocio, tales como un producto, un cliente o un empleado.
4. Validar los datos que usa la aplicación del data warehouse



## Meta Datos

Esta área del warehouse almacena todas las definiciones de los meta datos (datos acerca de los datos) usados por todos los procesos en el warehouse. Los meta datos son usados para una variedad de propósitos incluyendo:

- Los procesos de extracción, transformación y carga (meta datos es usado para mapear las fuentes de datos a una vista común de la información dentro del warehouse).
- Los procesos de gestión del warehouse (cada tabla es descrita incluyendo su estructura, índices, vistas; meta datos es usado también para automatizar la producción de tablas resumen).
- Como parte de los procesos de gestión de consulta (meta datos es usado para dirigir una consulta a la fuente de datos más apropiada)

## **Agregaciones**

Este componente del warehouse almacena todos los datos agregados, predefinidos y generados por el gestor del warehouse.

El propósito de información resumida es para mejorar el rendimiento de las consultas. Aunque hay costos operacionales incrementados asociados con la agregación inicial de los datos, esto debería ser compensado eliminando el requerimiento para ejecutar continuamente operaciones de agregación (tales como clasificación o agrupación) en las respuestas a las consultas de los usuarios. El dato agregado es actualizado continuamente en la medida que nuevos datos son cargados en al warehouse.

## **Gestor del Datawarehouse**

En algunos casos el gestor del warehouse también genera perfiles de consultas para determinar qué índices y agregaciones son apropiadas. Un perfil de consulta puede ser generado para cada usuario, grupo de usuario, o el data warehouse y está basada en la información que describe las características de las consultas tales como la frecuencia, tablas objetivo, y tamaño de los results set.

## **Gestor de Respaldos**

Es el componente que se encarga de respaldar constantemente la información del repositorio del datawarehouse.

## **Repositorio del Datawarehouse**

Es el repositorio en si o la base de datos física donde se almacena la información del datawarehouse.

Un DBMS para trabajar con un sistema de Datawarehouse debe cumplir con los siguientes requerimientos

### ***Rendimiento de carga***

- Datawarehouse requiere de carga incremental de nuevos datos en una base periódica dentro de ventanas de tiempo pequeñas
- El rendimiento de procesos de carga debería ser medido en cientos de millones de filas o gigabytes de datos por hora y no debería haber un límite máximo que restrinja al negocio

### ***Procesamiento de carga***

- Muchos pasos deben ser dados para cargar un dato nuevo o actualizado hacia el datawarehouse incluyendo conversión de datos, filtrado, reformateado, chequeos de integridad, almacenamiento físico, indexación y actualización de los meta datos
- Aunque cada paso en la práctica puede ser atómico, el proceso de carga debería parecer que se ejecuta como una unidad de trabajo única.

### ***Gestión de calidad de los datos***

- El datawarehouse debe asegurar consistencia local, consistencia global e integridad referencial a pesar de las fuentes "sucias" y tamaños masivos de bases de datos
- la preparación y carga son pasos necesarios, ellos no son suficientes. La habilidad para responder a las consultas de los usuarios finales es la medida del éxito para una aplicación de datawarehouse.
- Mientras más preguntas son respondidas, los analistas tienden a solicitar preguntas más complejas y creativas

### ***Rendimiento de consultas***

- Gestión basada en hechos y análisis ad hoc no deben ser retardadas o inhibidas por el rendimiento del RDBMS datawarehouse.
- Consultas complejas y grandes para operaciones claves del negocio deben ser completadas en un período de tiempo razonable.

### ***Escalabilidad de terabytes***

- El tamaño de data warehouse está creciendo a enormes tasas con tamaños en el rango de cientos de gigabytes hasta los terabytes y petabytes ( $10^{15}$ ).
- Los RDBMS no deben tener ninguna limitación arquitectural para el tamaño de la base de datos y deberían soportar gestión modular y paralela. En el evento de fallas, el RDBMS debería soportar disponibilidad continua, y proveer mecanismos para recuperación. El RDBMS debe soportar dispositivos de almacenamiento en masa tales como discos ópticos y dispositivos de gestión de almacenamiento jerárquico.
- Finalmente, rendimiento de consultas no debería ser dependiente del tamaño de la base de datos, sino más bien de la complejidad de la consulta.

**1 Referencia [11] de la Bibliografía.**

[2 Imagen perteneciente al sitio de Rueda Tecnológica. Referencia \[8\] de la Bibliografía](#)

[3 Referencia 7 de Bibliografía, Datawarehousing Fácil.](#)

[4 Información e imágenes tomadas del sitio de TODO BI.](#)

[5 Sección basada en su mayor parte de la referencia \[4\] de la bibliografía: Modelamiento Dimensional, Carmen Wolf](#)

---

[Continuación del tutorial \(3ª parte\)](#)

---

**Ing Cristhian Herrera 64**



This work is licensed under a [Creative Commons Attribution-NonCommercial-No Derivative](#)



[Works 2.5 License.](#)  
[Puedes opinar sobre este tutorial aquí](#)

## Recuerda

que el personal de [Autentia](#) te regala la mayoría del conocimiento aquí compartido ([Ver todos los tutoriales](#))

¿Nos vas a tener en cuenta cuando necesites consultoría o formación en tu empresa?

**¿Vas a ser tan generoso con nosotros como lo tratamos de ser con vosotros?**

[info@autentia.com](mailto:info@autentia.com)

Somos pocos, somos buenos, estamos motivados y nos gusta lo que hacemos .....

**Autentia = Soporte a Desarrollo & Formación**

[Autentia S.L.](#) Somos expertos en:  
**J2EE, Struts, JSF, C++, OOP, UML, UP, Patrones de diseño ..**  
y muchas otras cosas

---

## Nuevo servicio de notificaciones

Si deseas que te enviemos un correo electrónico cuando introduzcamos nuevos tutoriales, inserta tu dirección de correo en el siguiente formulario.

Subscribirse a Novedades	
e-mail	<input type="text"/>
	<input type="button" value="Enviar"/>

---

## Otros Tutoriales Recomendados ([También ver todos](#))

**Nombre Corto**

**Descripción**

[Administración Web de MySQL](#)

En este tutorial se mostrará como administrar MySQL de forma rápida y muy sencilla a través de páginas webs implementadas con tecnología PHP, para ello se utilizará la herramienta PHPmyAdmin

[Administracion Web de MySQL](#)

Os mostramos como instalar en vuestro PC phpMySQL, un potente gestor Web de MySQL utilizado en la mayoría de los Hostings.

[Modelado de MySQL con herramientas gratuitas](#)

Os mostramos otra alternativa de modelado gráfico de MySQL.

[Pool de conexiones a BBDD con struts](#)

Os mostramos como configurar un pool de conexiones a base de datos en vuestras aplicaciones construidas con struts

[Todo lo que querias saber sobre Datawarehouse \(IV\)](#)

En este documento Christian nos enseña exhaustivamente qué es y como funciona un datawarehouse.

[Modelado Gráfico de MySQL](#)

Os mostramos como instalar y utilizar DeZing e Importer de Datanamic para crear el modelo lógico y físico de bustra base de datos MySQL

[Todo lo que querias saber sobre Datawarehouse \(I\)](#)

En este documento Christian nos enseña exhaustivamente qué es y como funciona un datawarehouse.

[Todo lo que querias saber sobre Datawarehouse \(III\)](#)

En este documento Christian nos enseña exhaustivamente qué es y como funciona un datawarehouse.

Nota: Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento.

Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores.

En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo.

Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador [rcanales@adictosaltrabajo.com](mailto:rcanales@adictosaltrabajo.com) para su resolución.

[Patrocinados por enredados.com .... Hosting en Castellano con soporte Java/J2EE](#)



[www.AdictosAlTrabajo.com](http://www.AdictosAlTrabajo.com) Optimizado 800X600