Unidad I

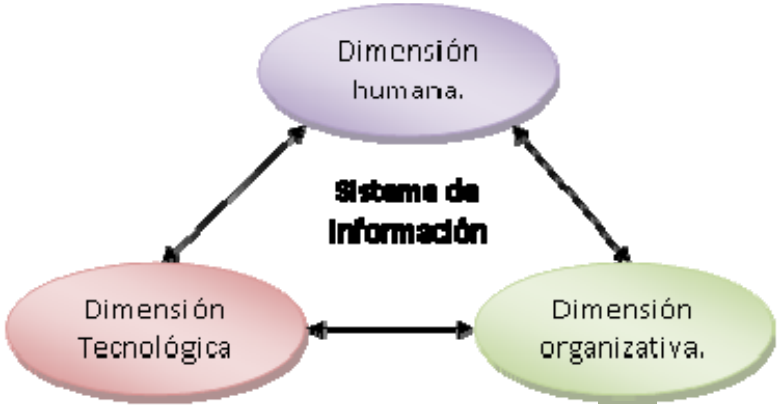
Introducción

M. en C. Euler Hernández Contreras
M. en C. Laura Méndez Segundo

Definición de un DSS (*Decision Support System*)

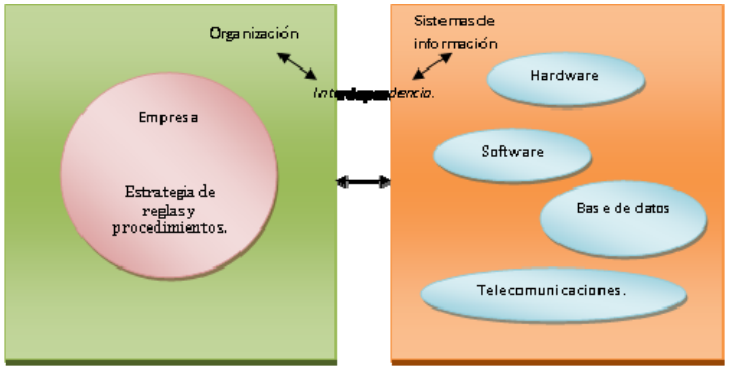
- Turban (1995) lo define como *“un sistema de información interactivo, flexible y adaptable basado en computadora, especialmente desarrollado para dar soluciones a problemas no estructurados que involucre el proceso de toma de decisiones, éste utiliza datos y proporciona una interfaz que permite al tomador de decisiones considerar su propia perspectiva”*.

Dimensiones y concepto de un Sistema de Información



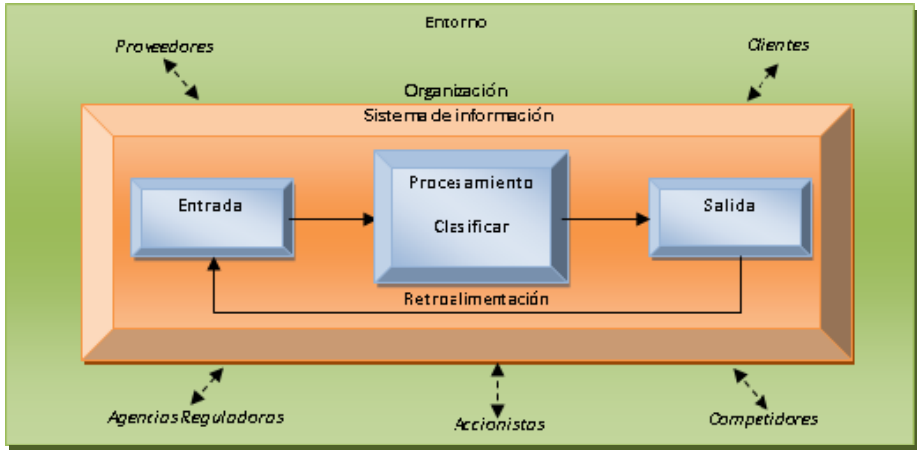
Sistema de información: *"Es un conjunto de componentes interrelacionados que recolectan (o recuperan), procesan, almacenan y distribuyen información para apoyar a los procesos de toma de decisiones y de control en una organización"*.

Interrelación entre los componentes de un Sistema de información



Actividades en un sistema de información:
a) Entrada: Captura o recoleta los datos en crudo desde el interior de la organización o a través de su entorno exterior.
b) Procesamiento: Convierte esa entrada en bruto en un formato significativo.
c) Salida: Transfiere la información procesada a las personas que harán uso de ella, o a las actividades para las que se utilizará.
Los sistemas de información también requieren:
d) Retroalimentación: La salida que se devuelve a los miembros apropiados de la organización para ayudarles a evaluar o corregir la etapa de entrada.

Funciones de un Sistema de información

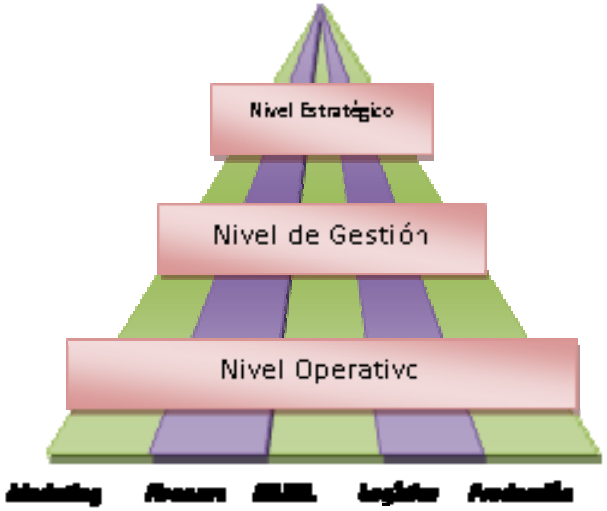


Clasificación de los Sistemas de información

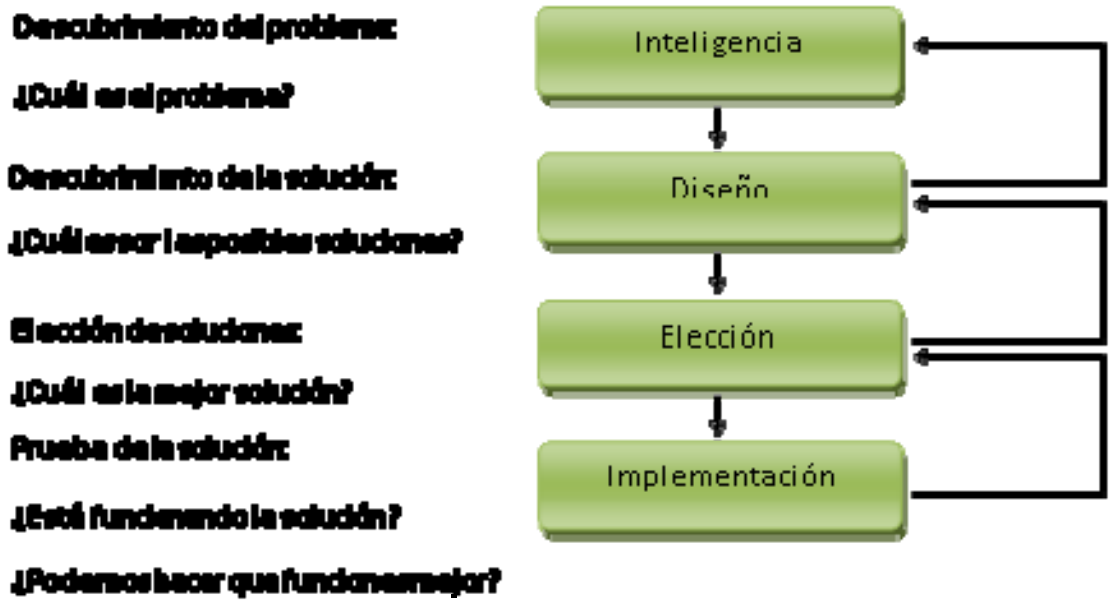
Soporte a las actividades operativas, que da lugar a sistemas de información para actividades más estructuradas (aplicaciones de contabilidad, nómina y pedidos y, en general, lo que se denomina “gestión empresarial”) o sistemas que permitan el manejo de información menos estructuradas: aplicaciones ofimáticas, programas técnicos para funciones de ingeniería, etc.

Soporte a las decisiones y el control de gestión, que puede proporcionarse desde las propias aplicaciones de gestión empresarial (mediante salidas de información existentes) o a través de aplicaciones específicas.

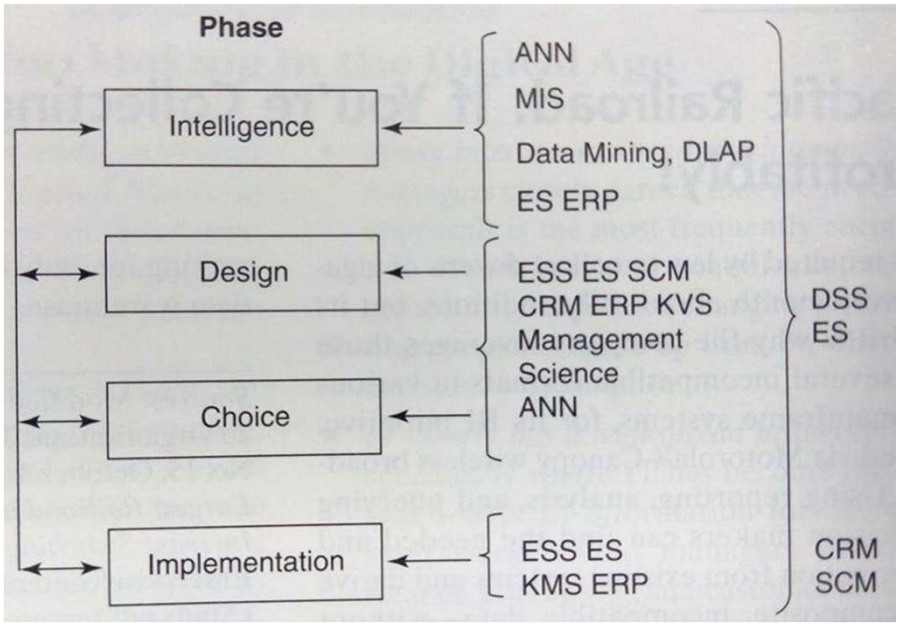
Así mismo también resulta habitual clasificar los sistemas en función del *tipo de función a la que se dirige*: financiera, recursos humanos, marketing, etc.



Etapas en la toma de decisiones



How decisions are support



ES: Expert System.
ANN: Artificial Neural Networks.
KMS: Knowledge Management System.
OLAP: Online Analytical Processing.
ERP: Enterprise Resource Planning.
SCM: Supply Chain Management.
CRM: Customer Relationship Management Systems.
ERM: Enterprise Resource Management
GSS: Group Support System.
EIS: Enterprise Information System
MIS: Management Information System.

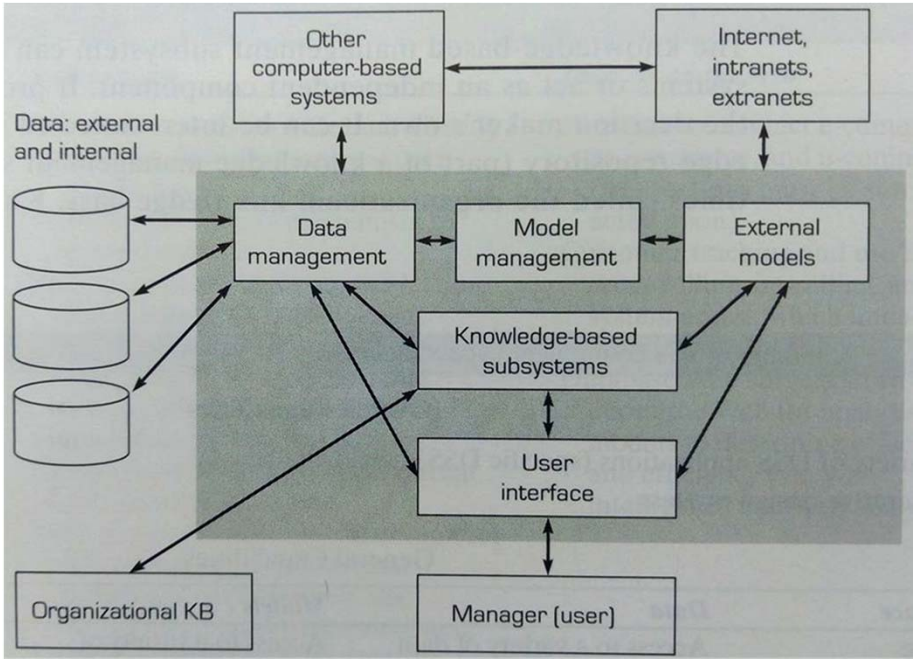
Components of decision support system

The data management subsystem, includes a database that contains relevant data for the situation.

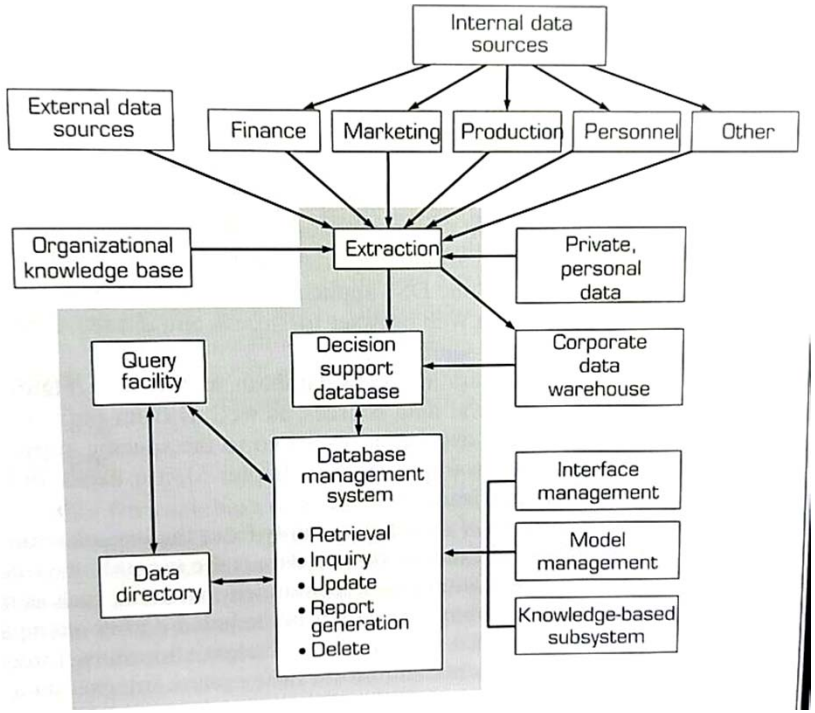
The model management subsystem,is a software package that includes financial, statistical, management science, or other quantitative models that provide the system's analytical capabilities and appropriate software management.

The user interface subsystem,the user communicates with and commands the DSS through the user interface subsystem.

The knowledge-based management subsystem, It provides intelligence to augment the decision maker's own.



The structure of the Data Management Subsystem



The data management subsystem is composed of the following elements:

a) DSS database

A database is a collection of interrelated data, organized to meet the needs and structure of an organization that can be used by more than one person for more than one application.

b) DBMS

A database is created, accessed, and updated by a DBMS. Most DSS are built with a standard commercial relational DBMS.

c) Data directory

The data directory is a catalog of all the data in database. It contains data definitions, and its main function is to answer question about the availability of data items, their source, and their exact meaning.

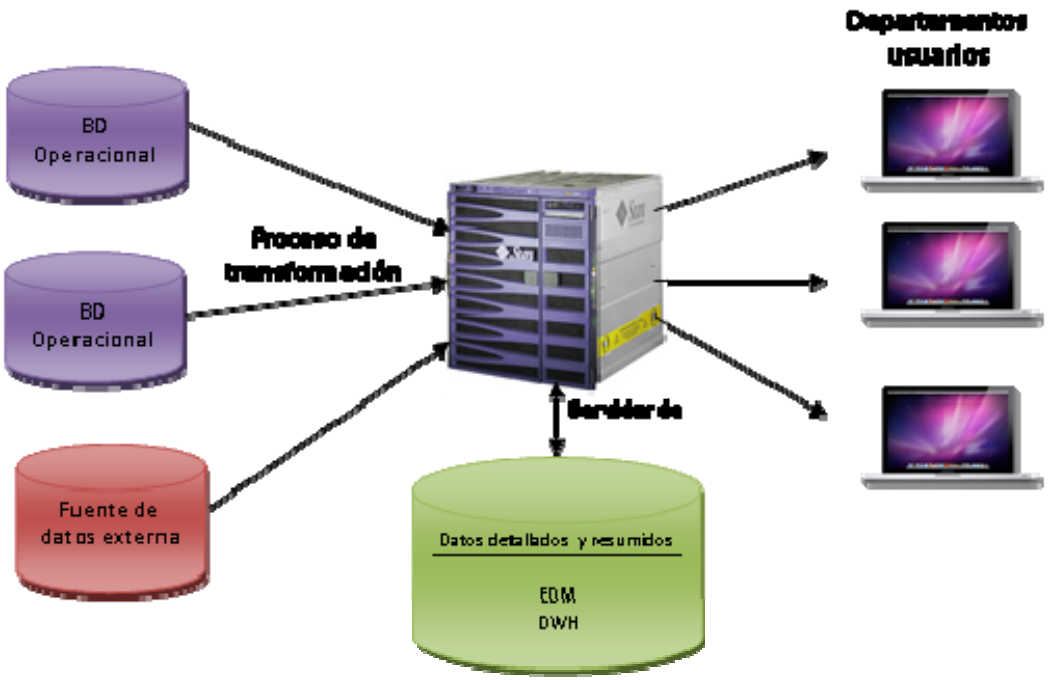
d) Query facility

In building and using DSS, it is often necessary to access, manipulate, and query data. The query facility performs there tasks.

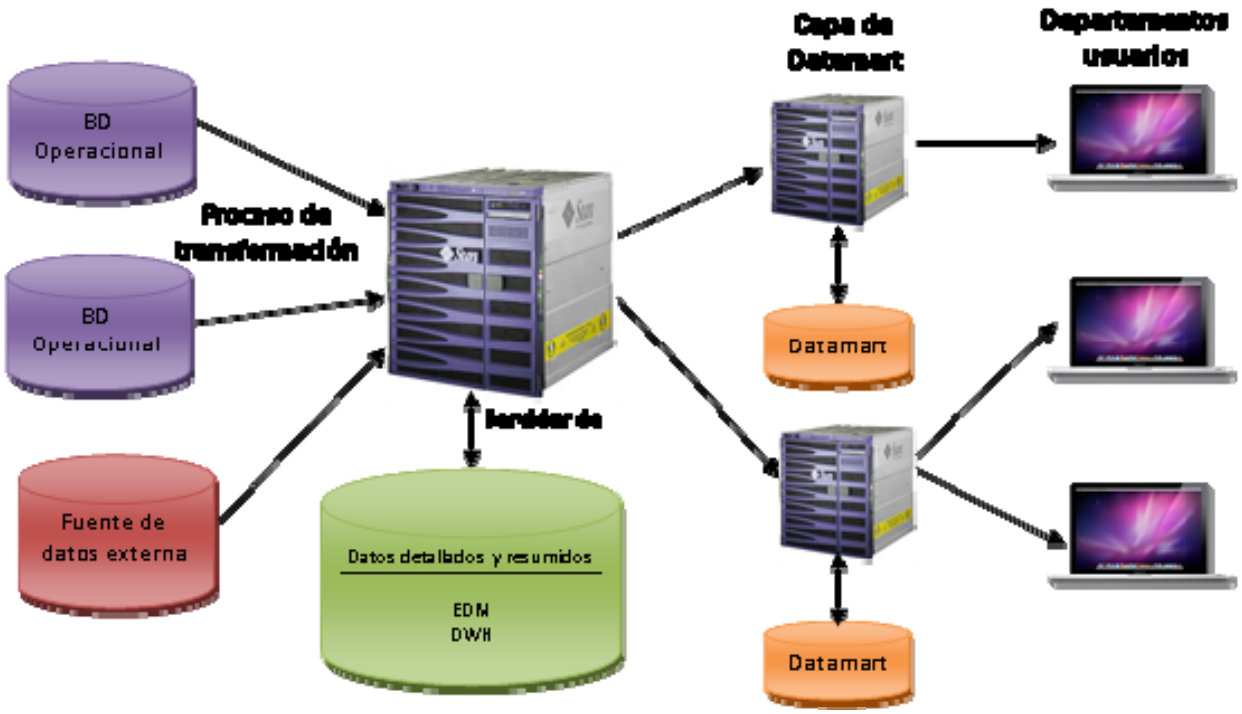
Definition of datawarehouse

A *Data Warehouse*(DW) is a combination of two primary components. The first is an integrated decision support database. The second is the related software programs used to collect, cleanse, transform, and store data from a variety of operational and external sources. Both of these parts combine to support historical, analytical, and business intelligence (BI) requirements.

Arquitectura para un datawarehouse (2 niveles)



Arquitectura para un datawarehouse (3 niveles)



Characteristics enable data warehouse

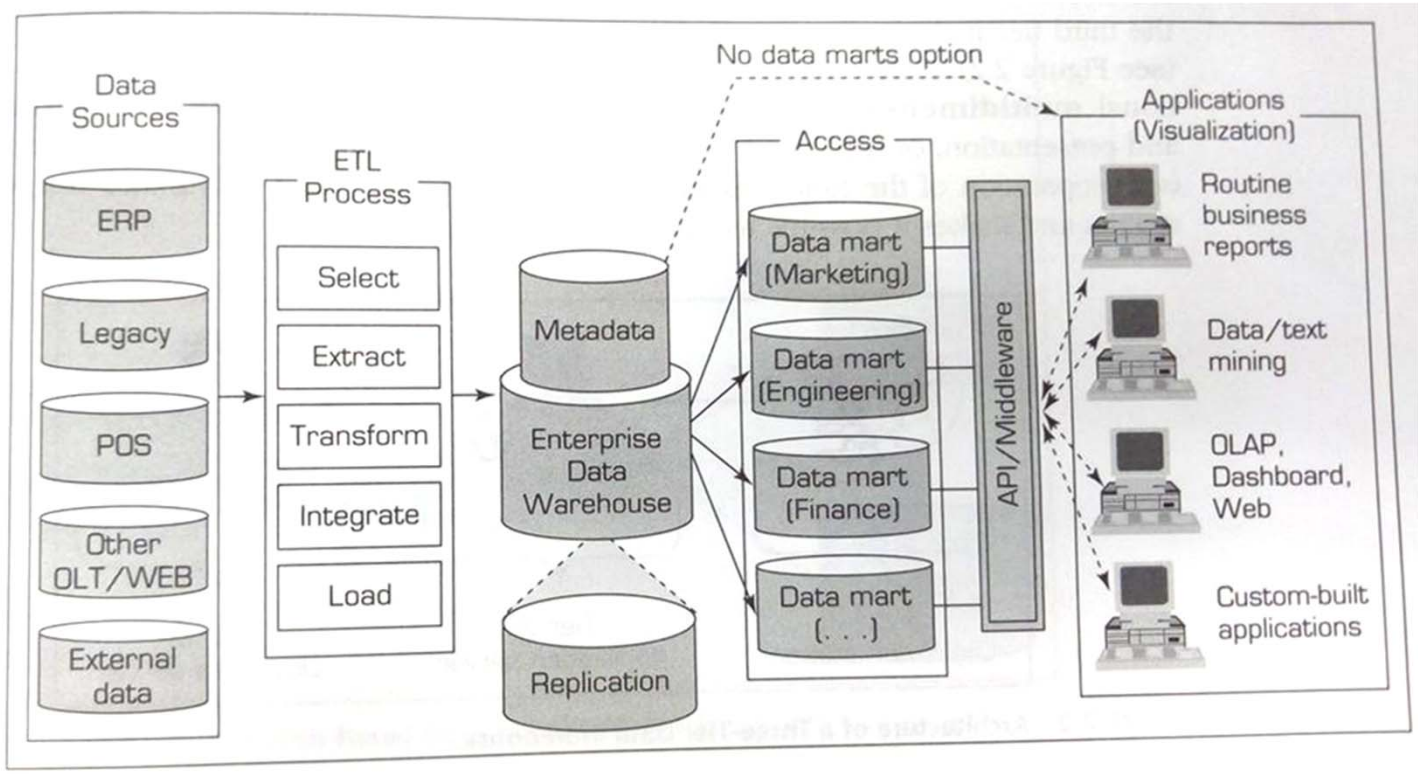
a)Web based.
Data warehouse are typically designed to provide an efficient computing environment for Web-based applications.
b) Relational/multidimensional.
A data warehouse uses either a relational structure or a multidimensional structure.
c)Client/server.
A data warehouse uses the client/server architecture to provide easy access for end users.
c) Real Time.
Newer data warehouse provide real-time, or active, data access and analysis capabilities.
d) Metadata.
A data warehouse contains metadata (data about data) about how the data are organized and how to effectively use them.

Unidad II

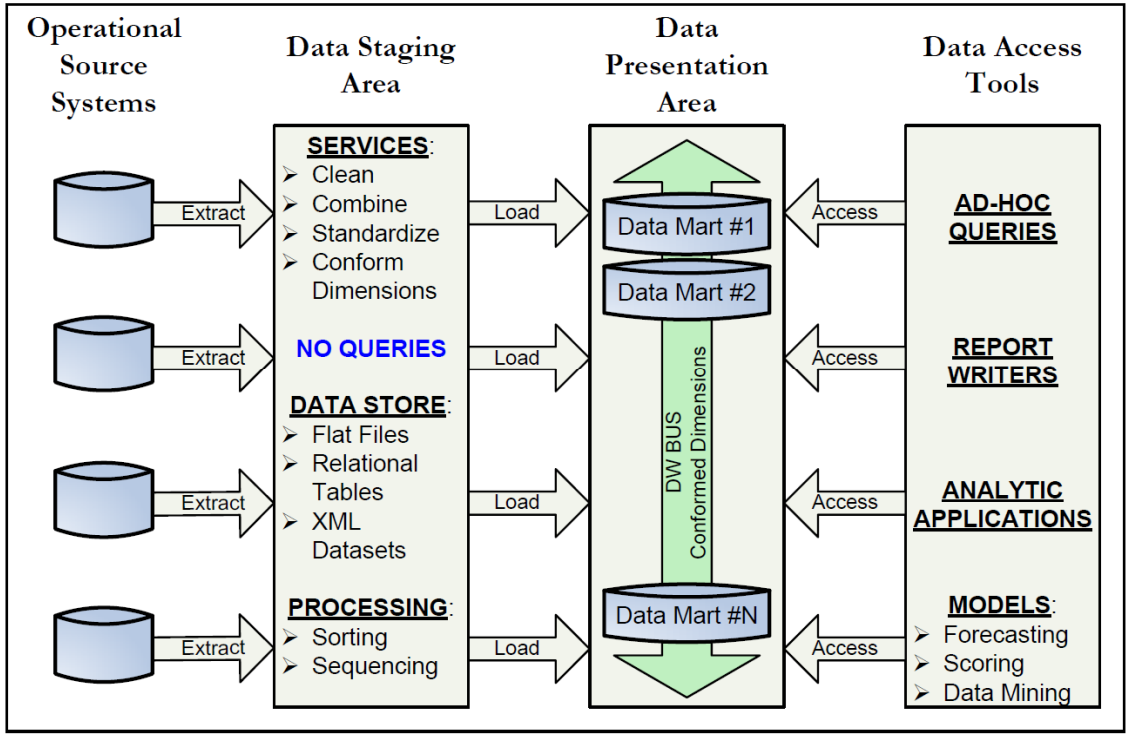
Arquitectura e Implementación de un Almacén de Datos

M. en C. Euler Hernández Contreras
M. en C. Laura Méndez Segundo

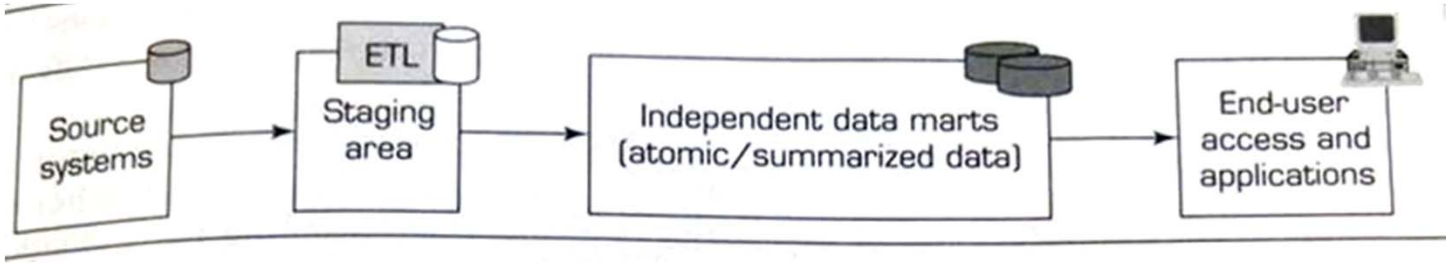
Arquitectura de un almacén de datos



Kimball's Data Warehouse Architecture

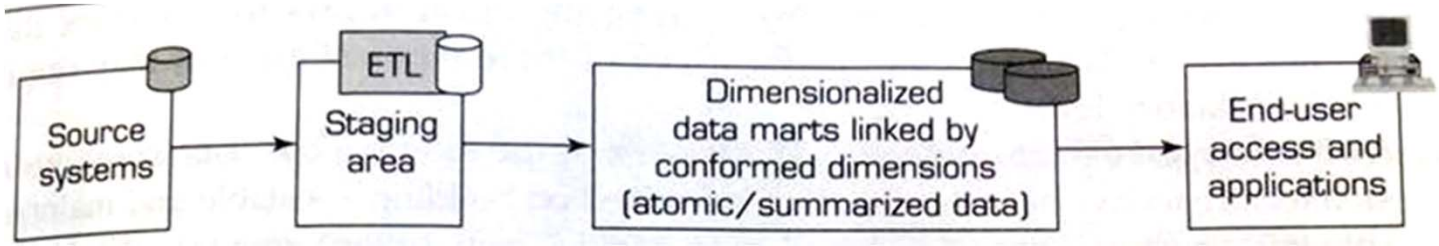


Alternative Data Warehousing Architectures 1/5



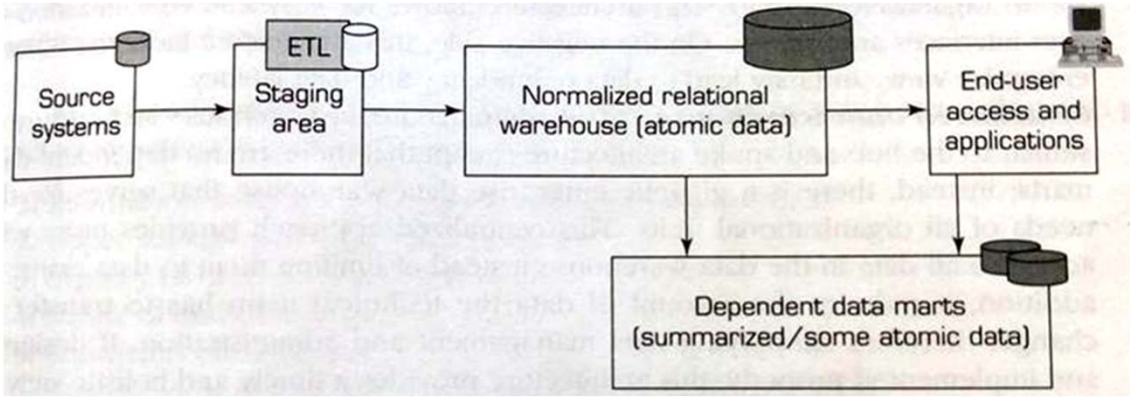
Independent data marts. The data marts are developed to operate independently of each other server for the needs of individual organizational units. Because of the independence, they may have inconsistent data definitions and different dimensions and measures, making it difficult to analyze data across the data marts.

Alternative Data Warehousing Architectures 2/5



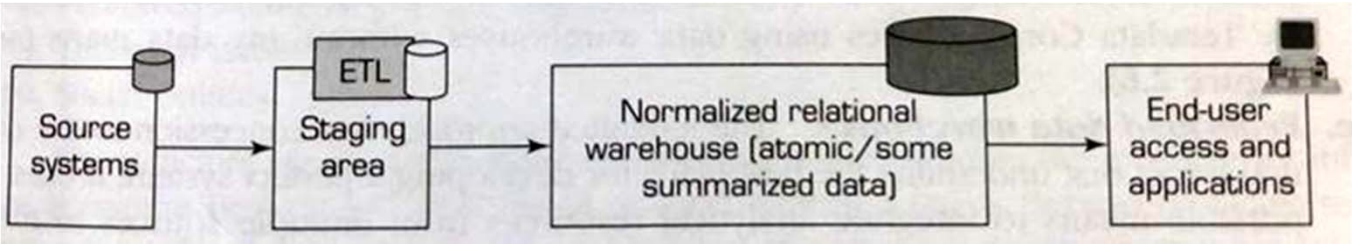
Data mart bus architecture. This architecture is a viable alternative to the independent data marts where the individual marts are linked to each other via some kind of middleware. Because the data are linked among the individual marts, there is a better chance of maintaining data consistency across the enterprise.

Alternative Data Warehousing Architectures 3/5



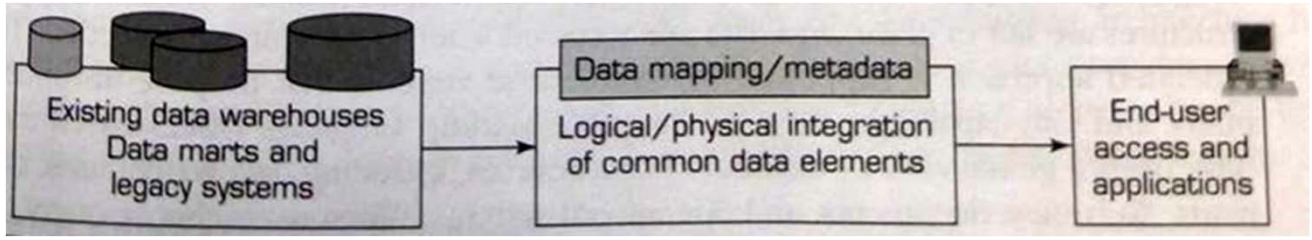
Hub-and-spoke architecture. This perhaps the most famous data warehousing architecture today. Here the attention is focused on building a scalable and maintainable infrastructure. This architecture allows for easy and customization of user interfaces and reports.

Alternative Data Warehousing Architectures 4/5



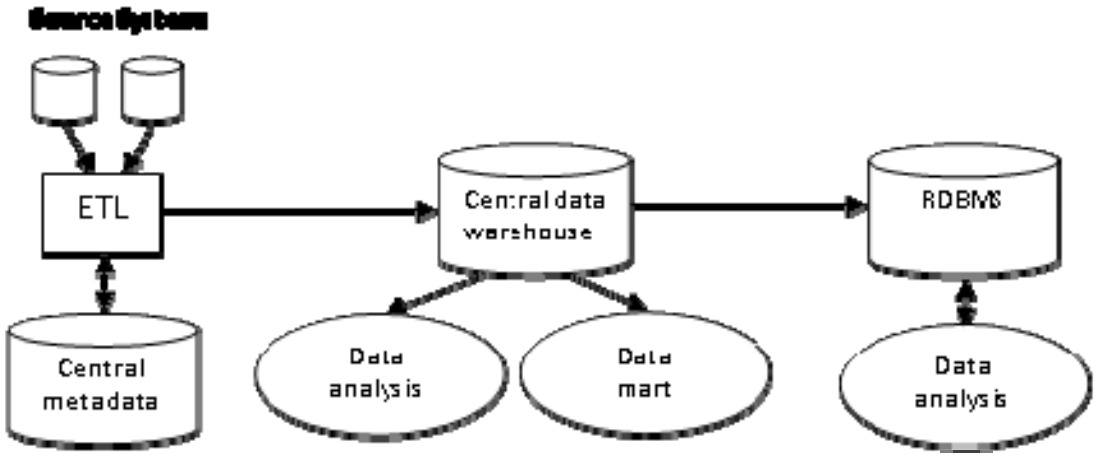
Centralized data warehouse. The centralized data warehouse architecture is similar to the hub-and-spoke architecture except that there are no dependent data marts; instead is a gigantic enterprise data warehouse that serves for the needs of all organizational units. This centralized approach provides users with access to all data in the data warehouse instead of limiting them to data marts. In addition, it reduces the amount of data the technical team has to transfer or change, therefore simplifying data management and administration.

Alternative Data Warehousing Architectures 5/5

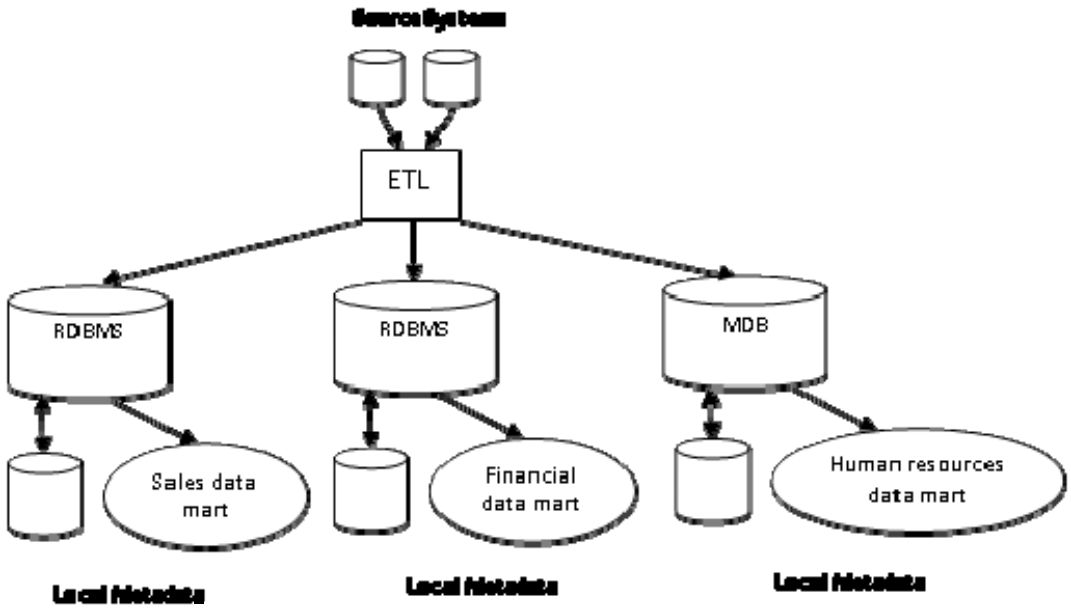


Federated data warehouse. The federated approach is a concession to the natural forces that undermine the best plans for developing a perfect system. It uses all possible means to integrate analytical resources from multiple sources to meet changing needs or business conditions. Essentially, the federated approach involves integrating disparate systems. In a federated architecture, existing decision support structures are left in place, and data are accessed from those sources as needed. The federated approach is supported by middleware vendors that propose distributed query and join capabilities.

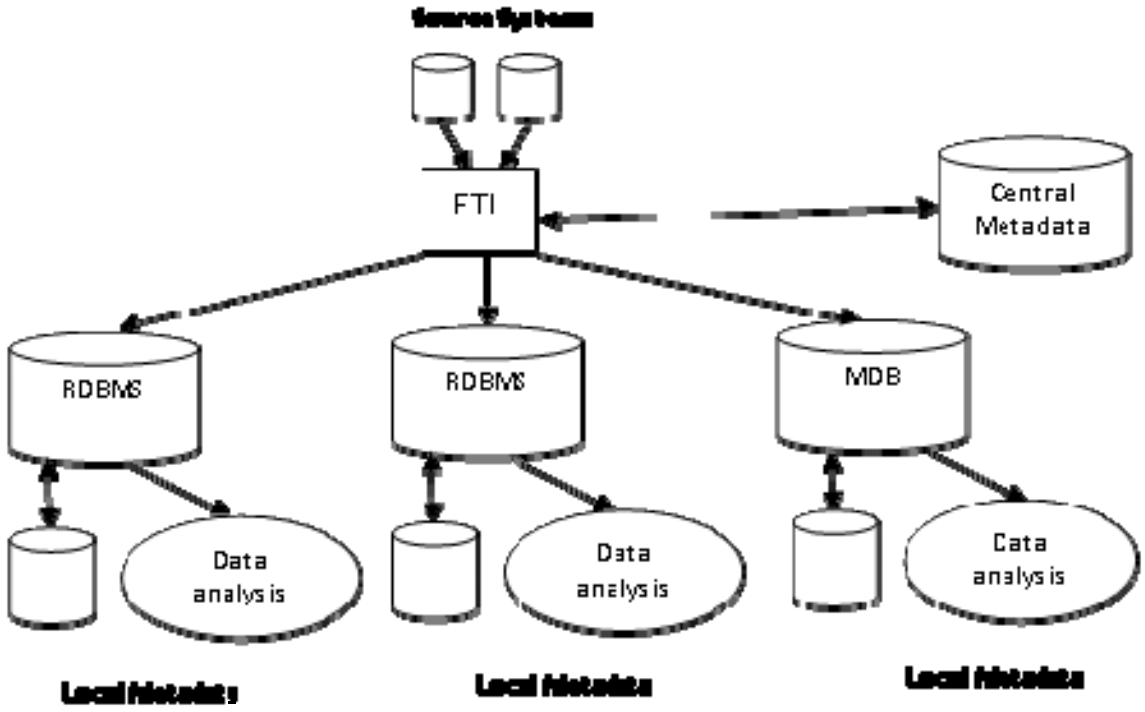
Enterprise Data Warehousing Architecture



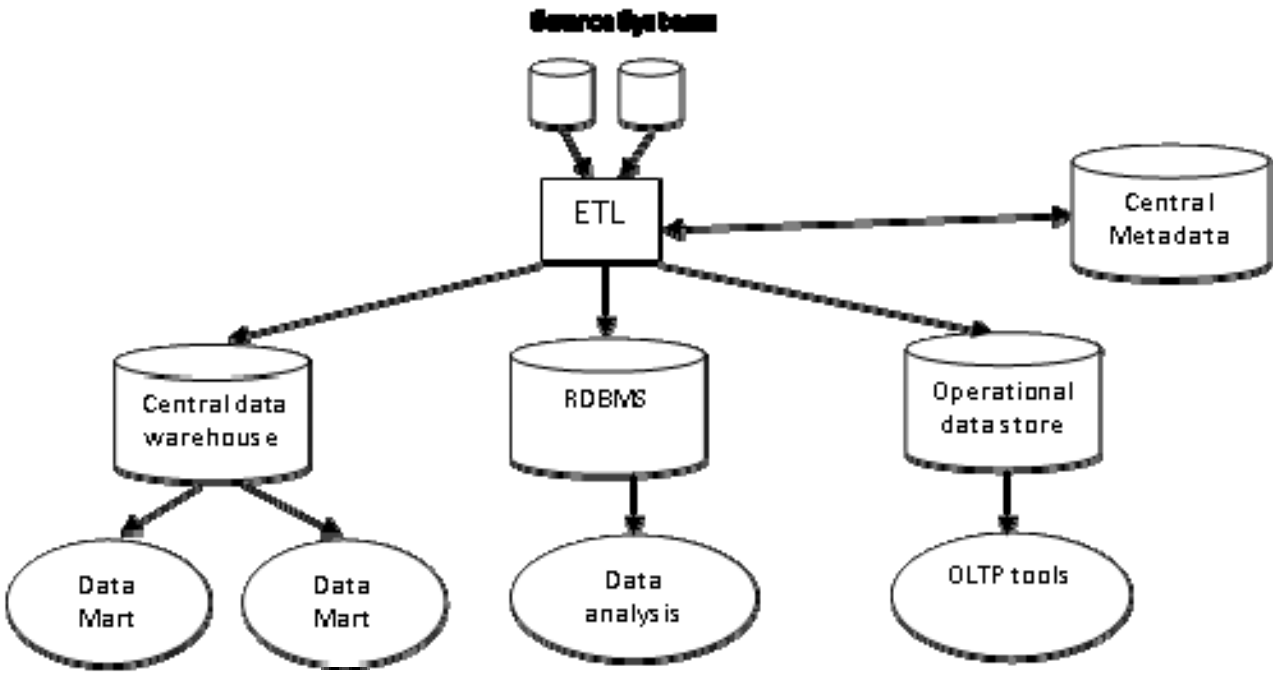
Data Mart Architecture



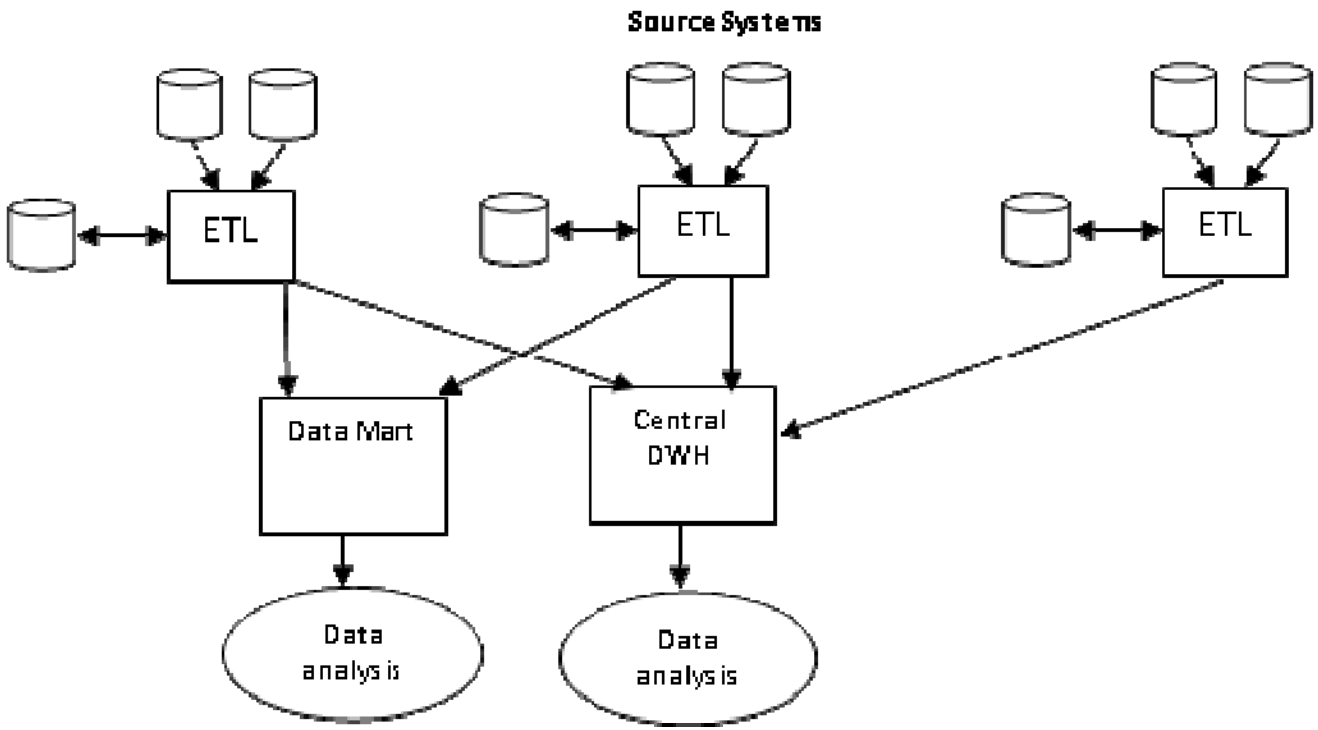
Hub-and-Spoke Data Mart Architecture



Enterprise Warehouse and Operational Data Store



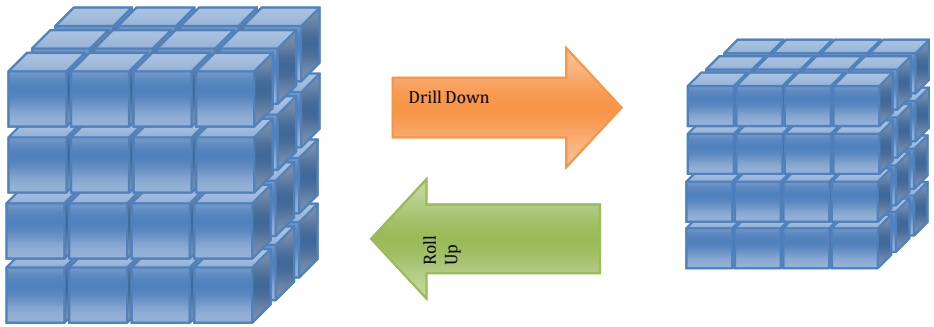
Distributed Data Warehouse Architecture



Funcionalidad Típica de un Almacén de Datos

Operación	Descripción
Roll up	Incremento en el nivel de agregación de los datos.
Drill down	Incremento en el nivel de detalle, opuesto a roll up
Slice	Reducción de la dimensionalidad de los datos mediante selección.
Dice	Reducción de la dimensionalidad de los datos mediante proyección.
Pivot o rotación	Reorientación de la visión multidimensional de los datos.

Funcionalidad Típica de un Almacén de Datos (*Drill Down – Roll Up*)



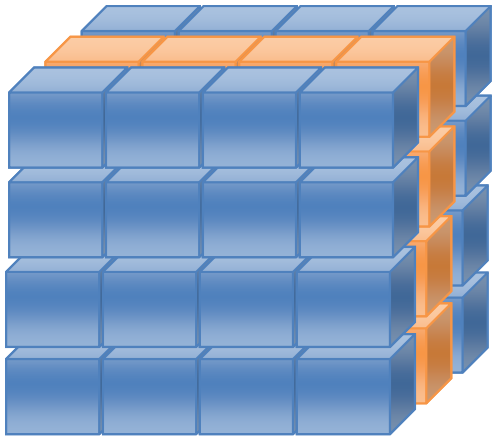
Drill Down – Roll Up: Es una técnica por la que el usuario puede navegar entre las jerarquías de una dimensión agrupando (*Roll-up*) o desagrupando (*Drill-down*) los datos.

El *drill down* y el *roll up* sirven para navegar el cubo sobre sus dimensiones, con el *roll up* se pasa desde el detalle a la generalización, y con el *drill down* se pasa desde un nivel general al detalle.

Funcionalidad Típica de un Almacén de Datos (*Pivot)*

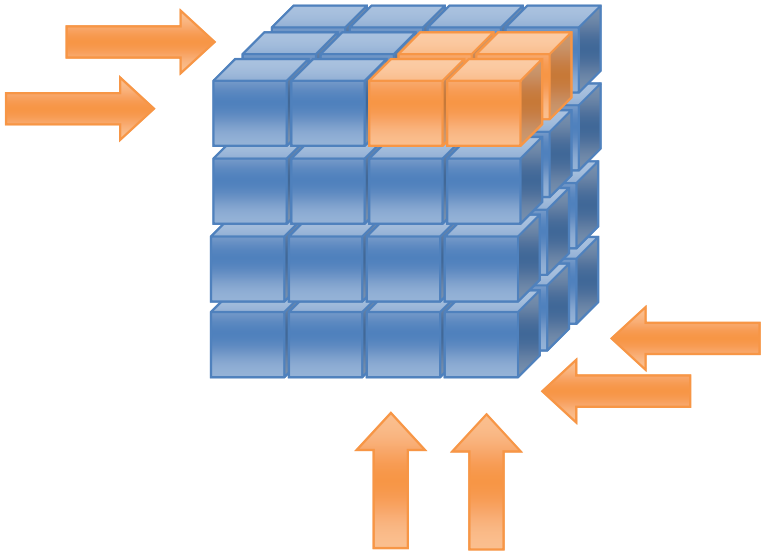
Rotación o *Pivot*: Selecciona el orden de visualización de las dimensiones, rota o gira el cubo según sus dimensiones

Funcionalidad Típica de un Almacén de Datos (*Slice)*



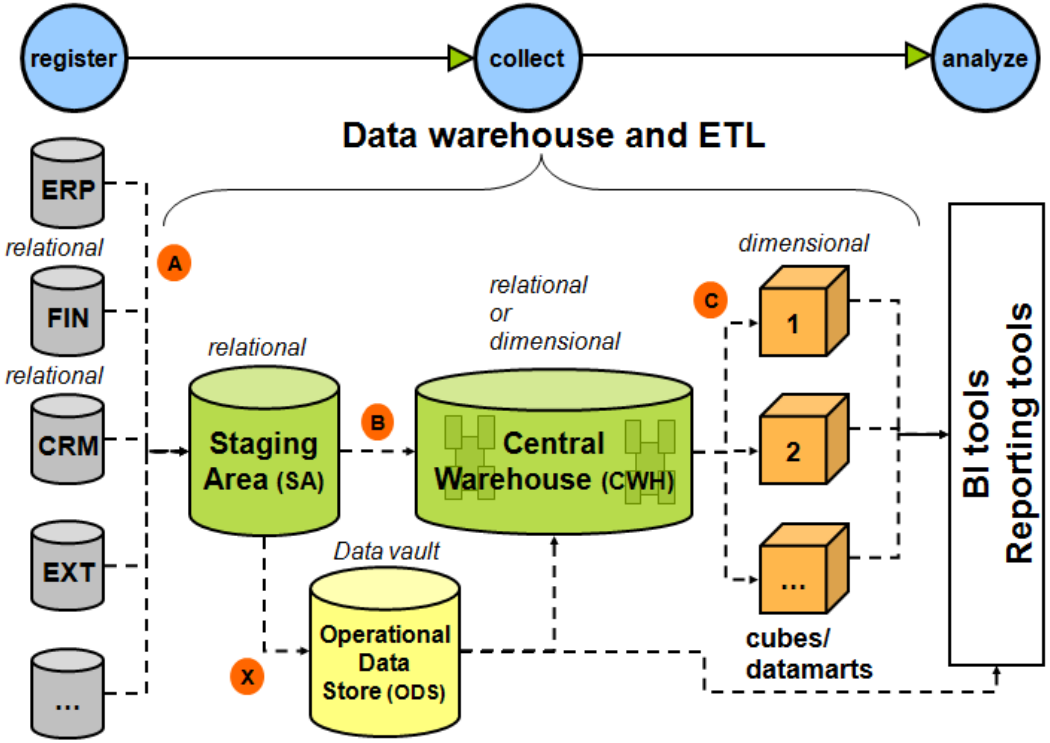
Slice: Al seleccionar un miembro en particular de una dimensión se forma una especie de rebanada (*slice*) del cubo original.

Funcionalidad Típica de un Almacén de Datos (*Dice)*



Dice: Al seleccionar varios miembros de varias dimensiones se forma sub-cubo, cubo más pequeño o dado (*dice*). Tanto *Slice* como *Dice* son formas particulares de Filtrado

Introducción a las Funciones ETL (Integración de Datos y ETL).



Categorías en las herramientas ETL

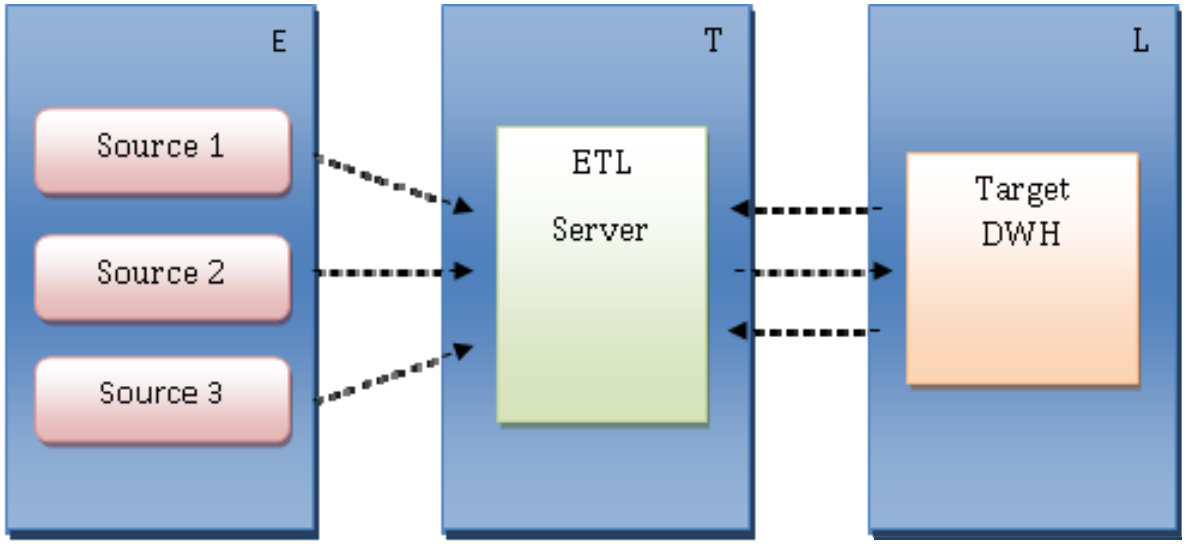
Generadores de código.
Crean programas de transformación 3GL/4GL personalizados basándose en las definiciones de los datos de origen y de destino. Los fabricantes están desarrollando componentes de gestión empleando técnicas tales como mecanismos de flujo de trabajo y sistemas automatizados de planificación.

Herramientas de replicación de datos.
Emplean disparadores de bases de datos o utilizan el registro de actividades de la base de datos para capturar los cambios efectuados en un origen de datos y aplicar dichos cambios a una copia de esos datos de origen que esté ubicada en un sistema diferente.

Estas herramientas pueden utilizarse para reconstruir una base de datos después de un fallo o para crear una base de datos para un mercado de datos, siempre que el número de orígenes de datos sea pequeño y el grado requerido de transformación de los datos sea relativamente bajo.

Motores de transformación dinámica.
Los motores de transformación dinámica dirigidos por reglas capturan datos de un sistema de origen a intervalos definidos por el usuario, transforman esos datos y luego envían y cargan los resultados en un entorno de destino.

Arquitectura CLÁSICA de una herramienta ETL



Unidad III

Modelo de Datos y OLAP

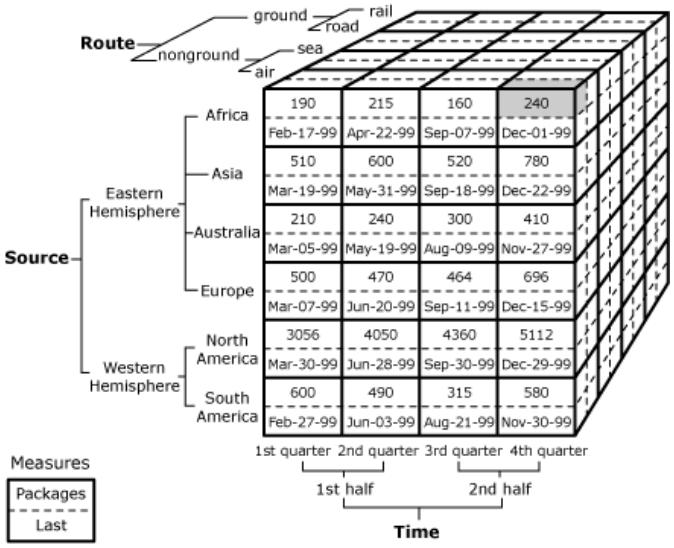
M. en C. Euler Hernández Contreras
M. en C. Laura Méndez Segundo

OLTP vs OLAP

Sistemas OLTP	Almacenes de Datos (OLAP)
Almacena datos actuales	Almacena datos históricos
Almacena datos detallados	Almacena datos resumidos en poca o gran medida
Los datos son dinámicos	Los datos son principalmente estáticos
Procesamiento repetitivo	Procesamiento ad hoc, no estructurado y heurístico
Alta tasa de transacciones	Tasa media o baja de transacciones
Patrón de uso predecible	Patrón de uso impredecible
Dirigido por transacciones	Dirigido por análisis
Orientado a la aplicación	Orientado a los temas
Soporta las decisiones cotidianas	Soporta las decisiones estratégicas
Sirve a un gran número de usuarios administrativos/operacionales	Sirve a un número relativamente bajo de usuarios de tipo gerencial

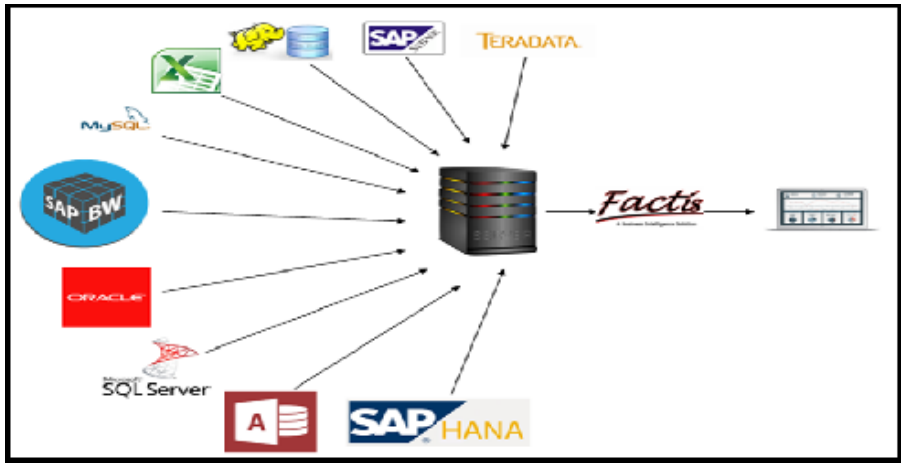
Reglas de Codd para las herramientas OLAP

1. Vista conceptual multidimensional
Las herramientas OLAP deben proporcionar a los usuarios un modelo multidimensional que se corresponda con las vistas que los usuarios tengan de la empresa, y que sea intuitivamente analítico y fácil de usar.



Reglas de Codd para las herramientas OLAP

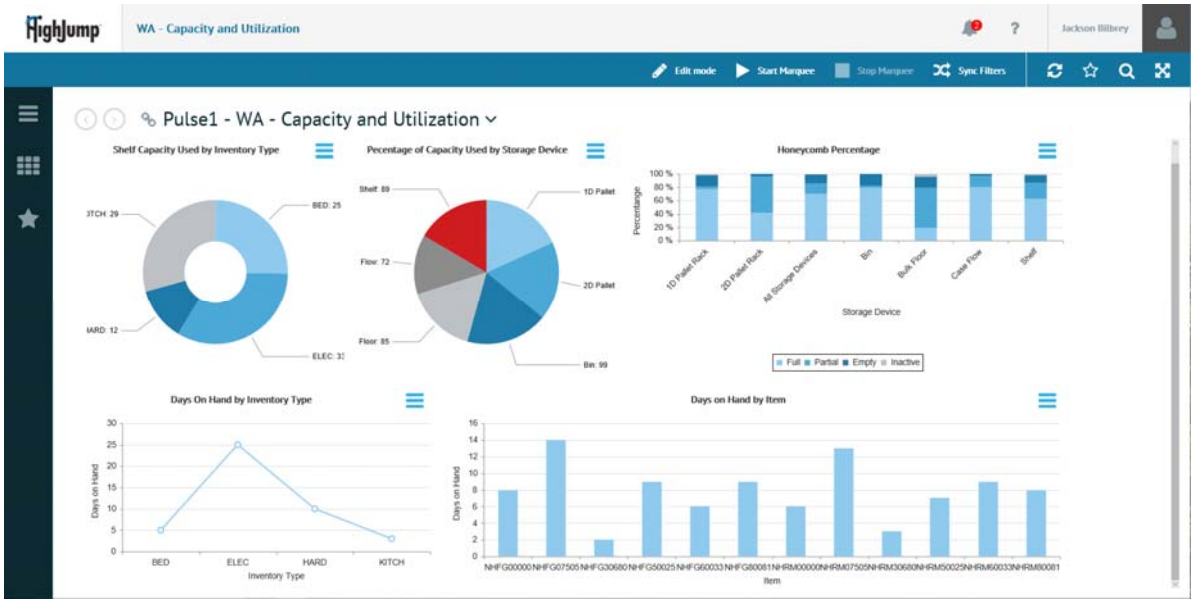
2. Transparencia
La tecnología OLAP, la arquitectura y la BD subyacente, y la posible heterogeneidad de las fuentes de datos de entrada pueden ser transparentes a los usuarios.



Reglas de Codd para las herramientas OLAP

3. Accesibilidad
La herramienta OLAP debe poder acceder a los datos requeridos para el análisis desde múltiples orígenes de datos empresariales heterogéneos.

Reglas de Codd para las herramientas OLAP

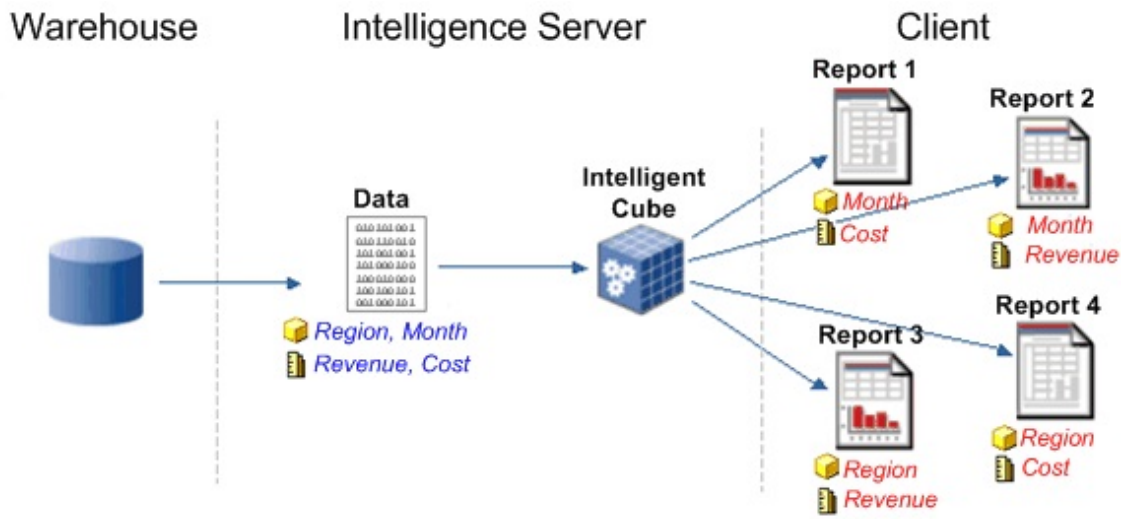


4. Prestaciones coherentes en la generación de informes

A medida de que se incrementan el número de dimensiones, los niveles de agregación y el tamaño de la base de datos, los usuarios no deben percibir ninguna degradación significativa en el rendimiento. No debe haber ninguna alteración en la forma de calcular los principales valores de rendimiento.

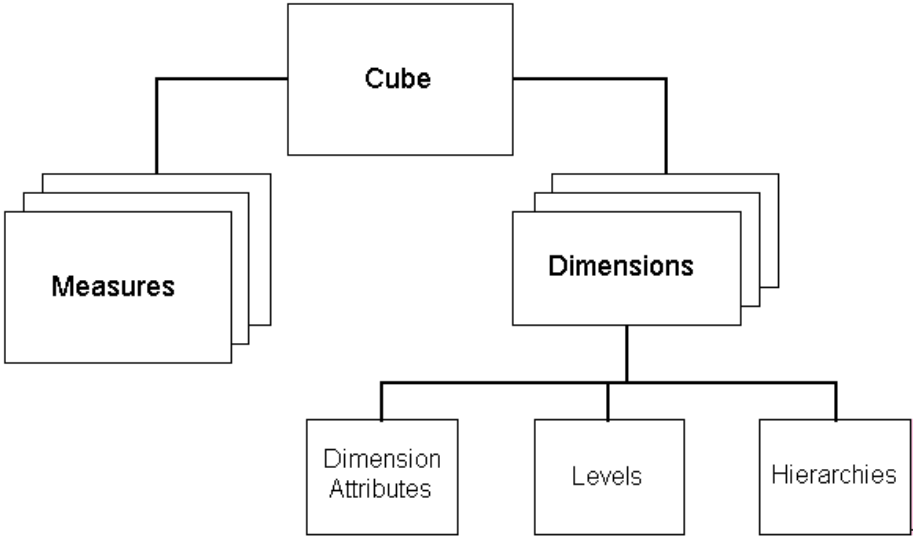
Reglas de Codd para las herramientas OLAP

5. Arquitectura cliente-servidor
El sistema OLAP debe ser capaz de operar de manera eficiente en un entorno cliente-servidor. La arquitectura debe proporcionar unas prestaciones, una flexibilidad, una adaptabilidad, una escalabilidad y una interoperabilidad óptimas.



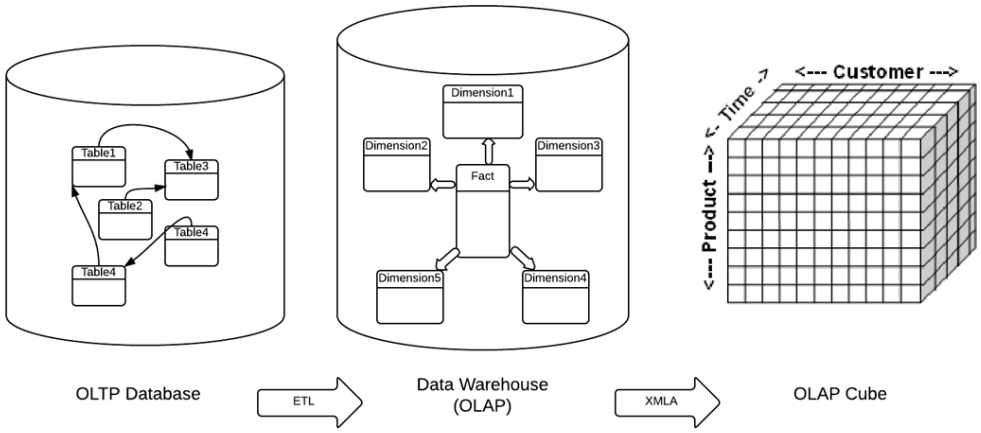
Reglas de Codd para las herramientas OLAP

6. Dimensionalidad genérica
Todas las dimensiones de los datos deben de ser equivalentes tanto en su estructura como en las capacidades de operación. En otras palabras, la estructura básica, las fórmulas y las capacidades de generación de informes no deben estar sesgadas hacia ninguna dimensión concreta.

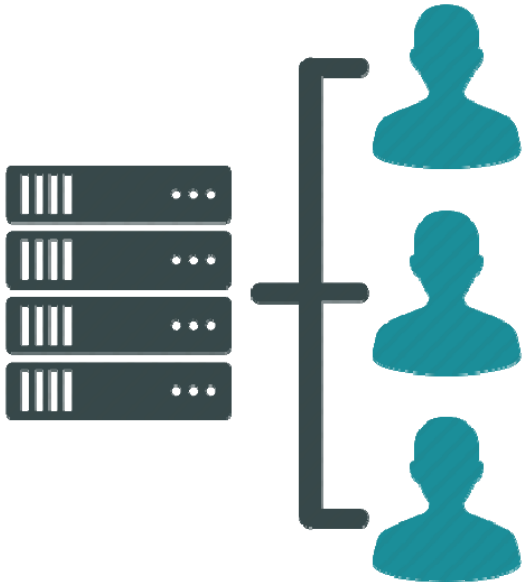


Reglas de Codd para las herramientas OLAP

7. Gestión dinámica de matrices dispersas
El sistema OLAP debe poder adaptar su esquema físico al modelo analítico específico que optimice la gestión de matrices dispersas, con el fin de conseguir y mantener el nivel de prestaciones requerido. Los valores nulos deben almacenarse de una manera eficiente y no afectar de forma adversa ni a la precisión ni a la velocidad del acceso a los datos.

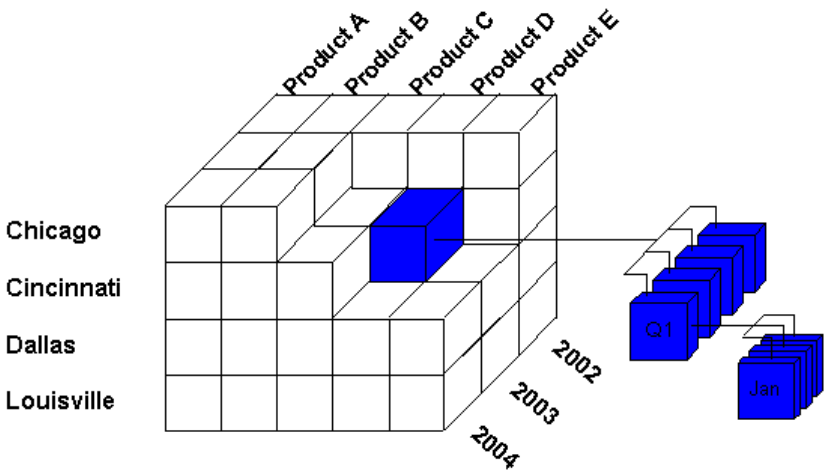


Reglas de Codd para las herramientas OLAP



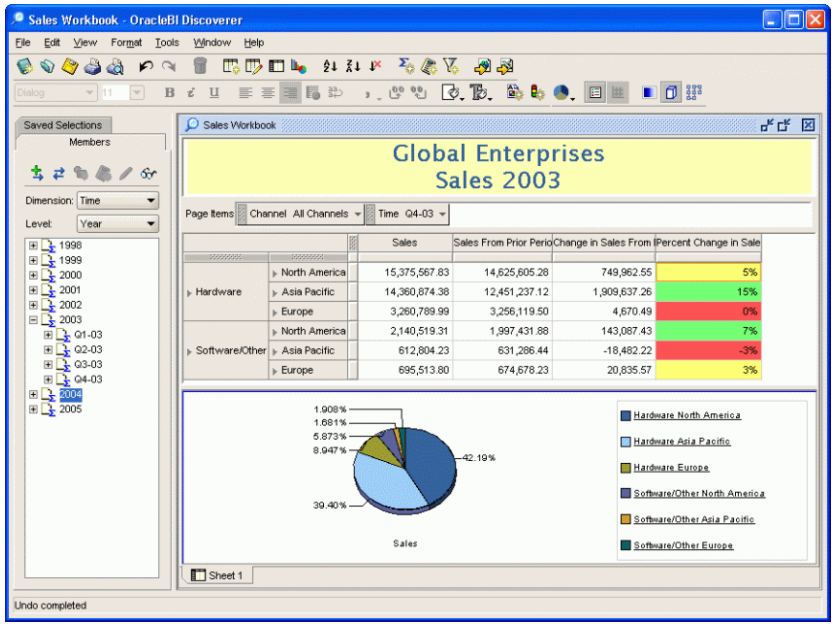
8. Soporte multiusuario
El sistema OLAP debe poder soportar grupos de usuarios trabajando concurrentemente sobre el mismo modelo de datos de la empresa, o sobre modelos distintos.

Reglas de Codd para las herramientas OLAP



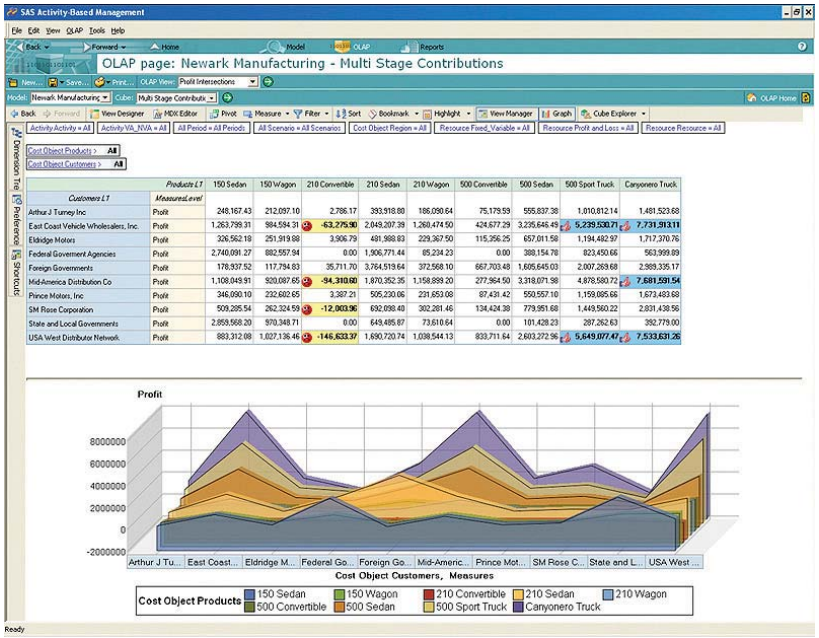
9. Operaciones interdimensionales no restringidas
El sistema OLAP debe poder reconocer las jerarquías de las dimensiones y realizar automáticamente los cálculos de totalización asociados dentro de una dimensión y entre diversas dimensiones.

Reglas de Codd para las herramientas OLAP



10. Manipulación de datos intuitiva
El pivotaje, la profundización y la consolidación, así como otras manipulaciones, deben realizarse mediante asociaciones de tipo “apuntar y hacer click” y “arrastrar y soltar” sobre las celdas del cubo.

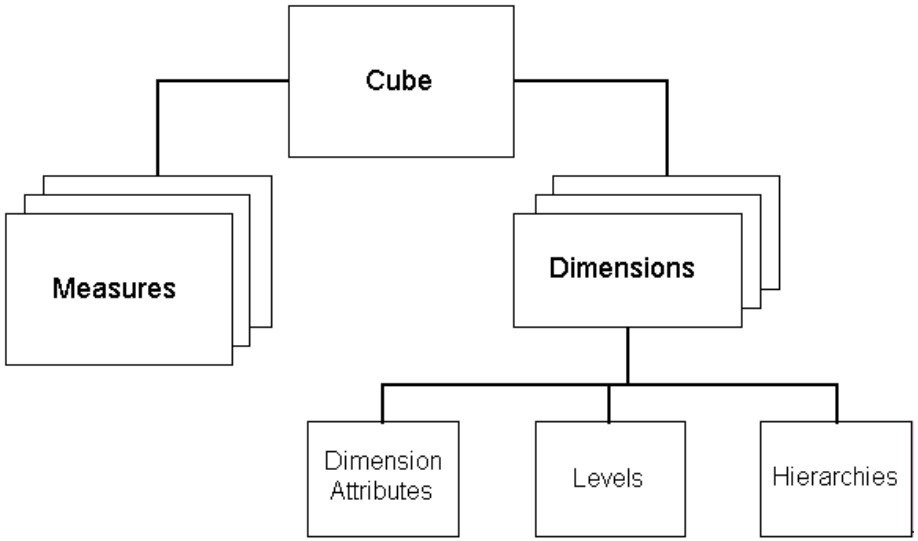
Reglas de Codd para las herramientas OLAP



11. Generación flexible de informes
Debe existir la capacidad de disponer las filas, las columnas y las celdas de modo tal que se facilite el análisis mediante una presentación visual intuitiva de los informes analíticos.

Reglas de Codd para las herramientas OLAP

12. Dimensiones y niveles de agregación ilimitados.
Dependiendo de los requisitos de la organización, un modelo analítico puede tener numerosas dimensiones, cada una de las cuales puede disponer de múltiples jerarquías. El sistema OLAP no debe poner ninguna restricción artificial sobre el número de dimensiones ni sobre los niveles de agregación.



Modos de almacenamiento de OLAP

OLAP puede trabajar con tres tipos de almacenamiento:

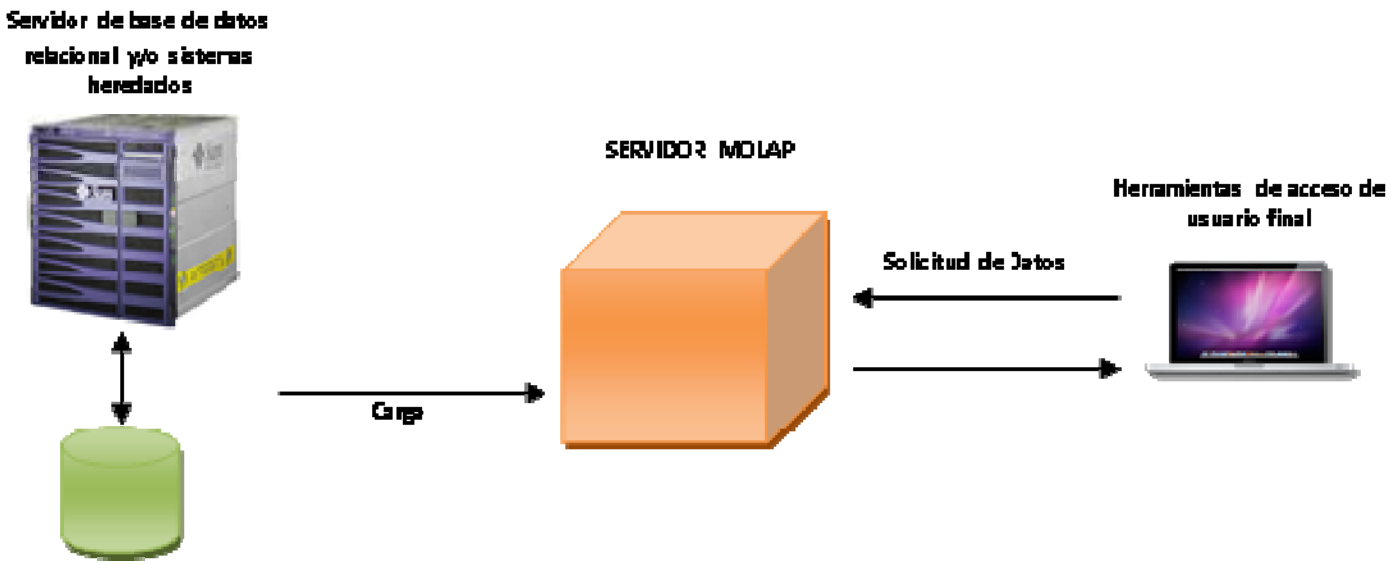
a) Almacenamiento MOLAP(*Multidimensional OLAP*).

b) Almacenamiento ROLAP (*Relational OLAP*).

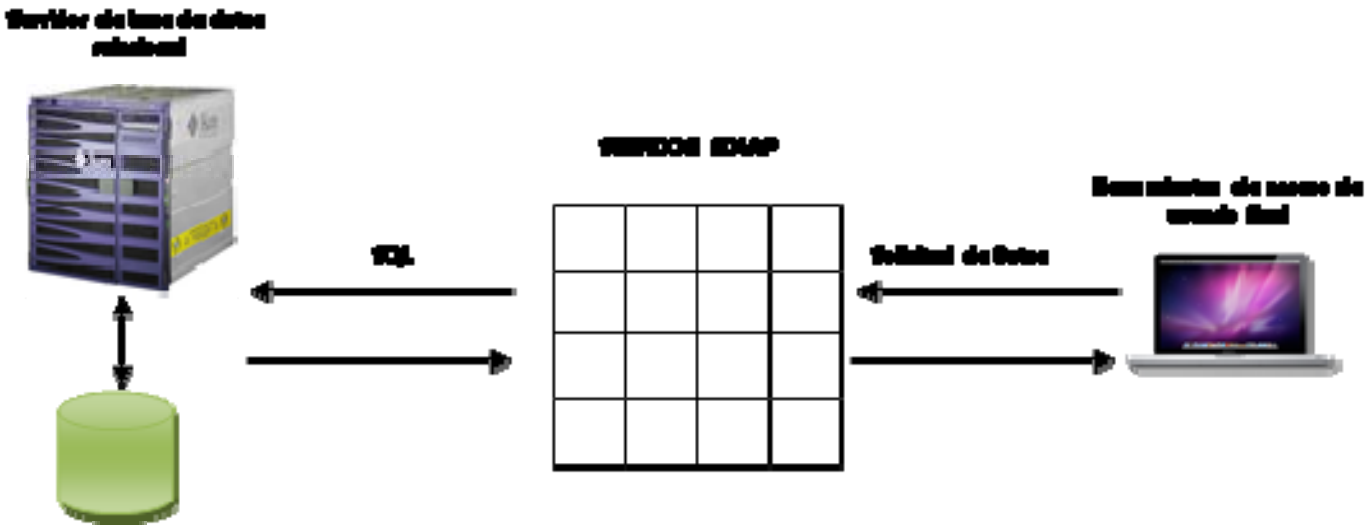
c) Almacenamiento HOLAP(*Hybrid OLAP*).

d) OLAP de escritorio DOLAP(*Desktop OLAP*)

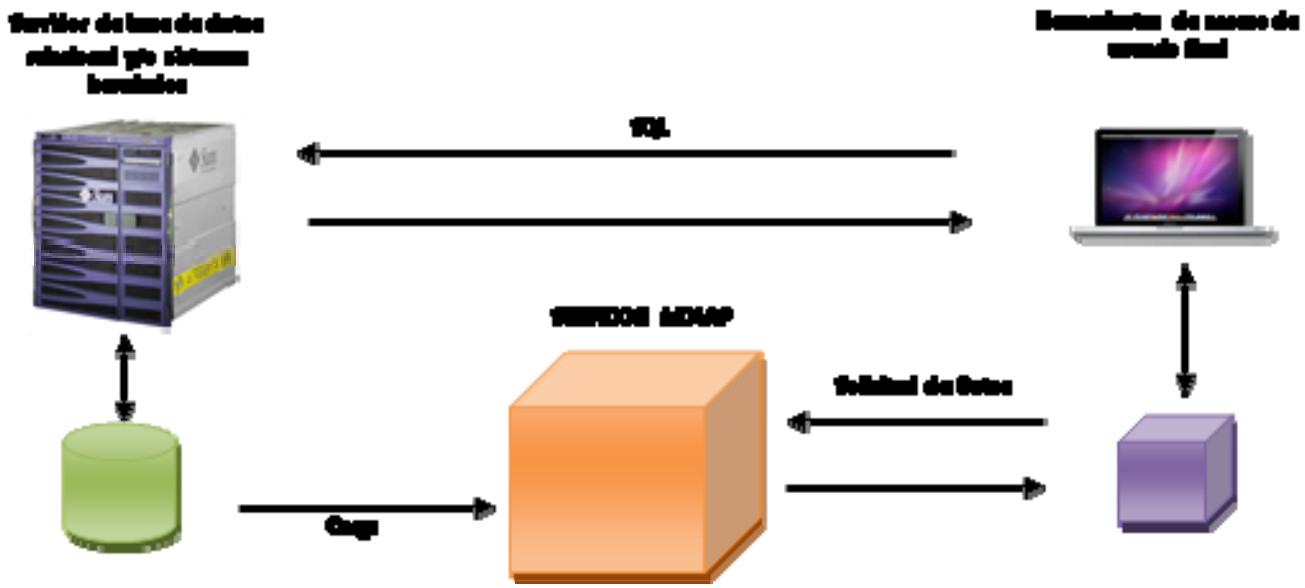
MOLAP (*Multidimensional OLAP*)



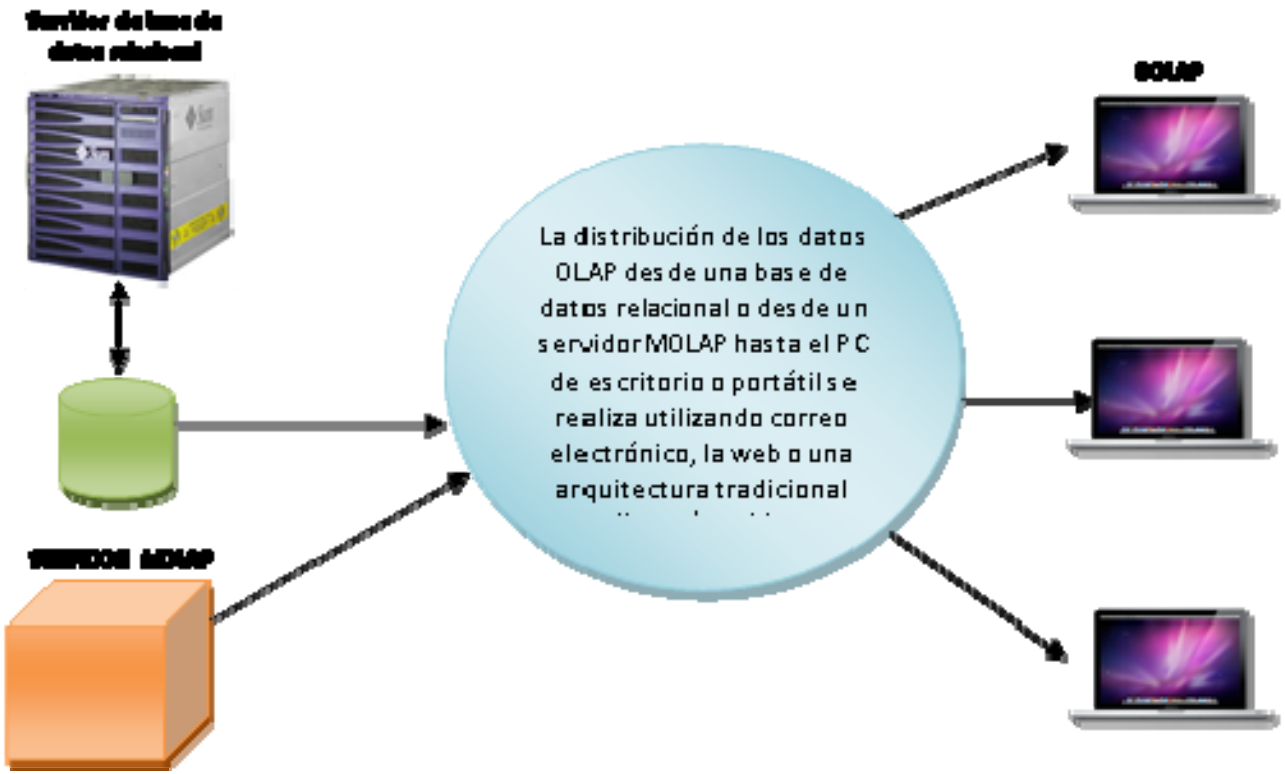
ROLAP (*Relational OLAP*)



HOLAP (*Hybrid OLAP*)



DOLAP (*Desktop OLAP*)

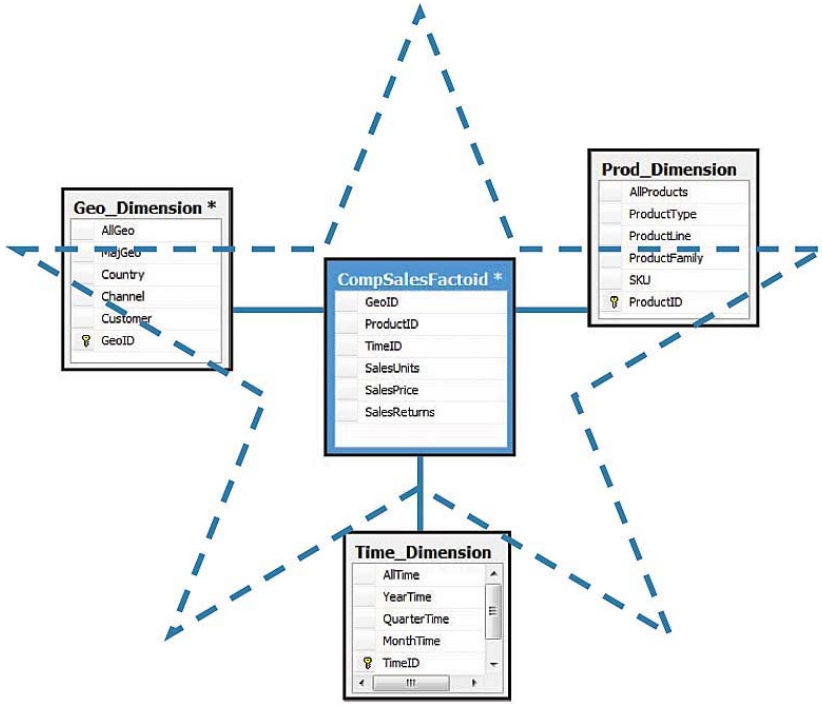


Modelos multidimensionales

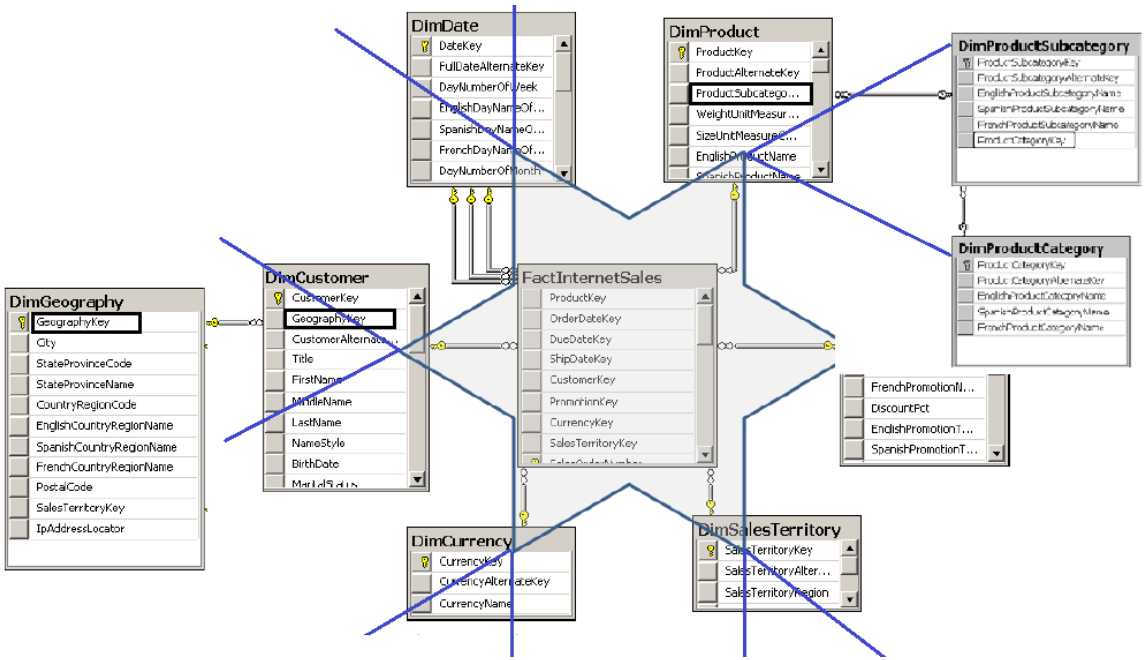
Tres son los esquemas multidimensionales comunes:

a) Esquema de Estrella.
Formado por una tabla de hechos con una única tabla para cada dimensión.
b) Esquema en copos de nieve.
Es una variante del esquema estrella en el que las tablas dimensiones de este último se organizan jerárquicamente mediante su normalización.
c) Constelación de hechos.
Es un conjunto de tablas de hechos que comparten algunas tablas de dimensiones.

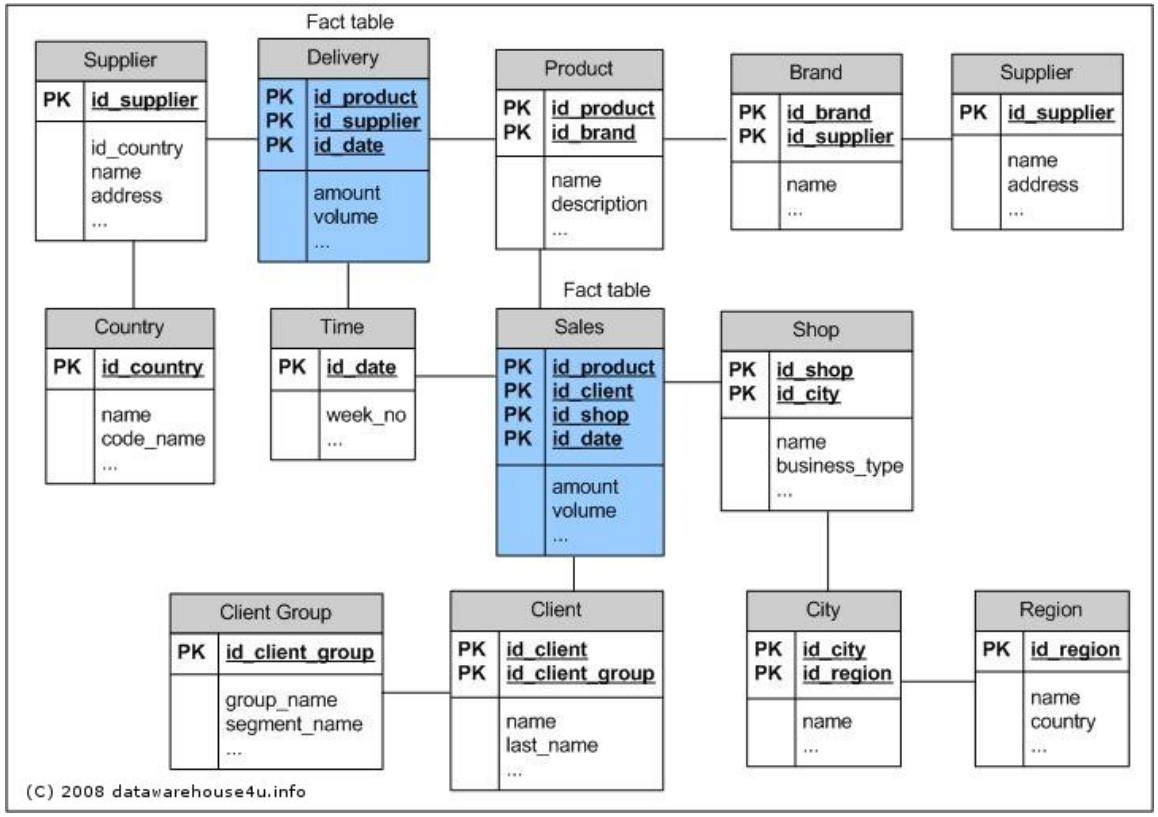
Modelos multidimensionales (Esquema Estrella –star schema)



Modelos multidimensionales (Copos de nieve – snowflake schema)



Modelos multidimensionales (Constelación de hechos – fact constellation schema)



Unidad IV

Descubrimiento de conocimiento de datos

M. en C. Euler Hernández Contreras
M. en C. Laura Méndez Segundo

Definición de Minería de Datos

La minería de datos es un proceso de descubrimiento de información valiosa de grandes cantidades de datos almacenados en bases de datos, datawarehouses o algún otro repositorio de información.

Definición de KDD (*Knowledge Discovery in Databases*)

Proceso no trivial de identificar patrones válidos, novedosos, potencialmente útiles y, en última instancia, comprensibles a partir de los datos.

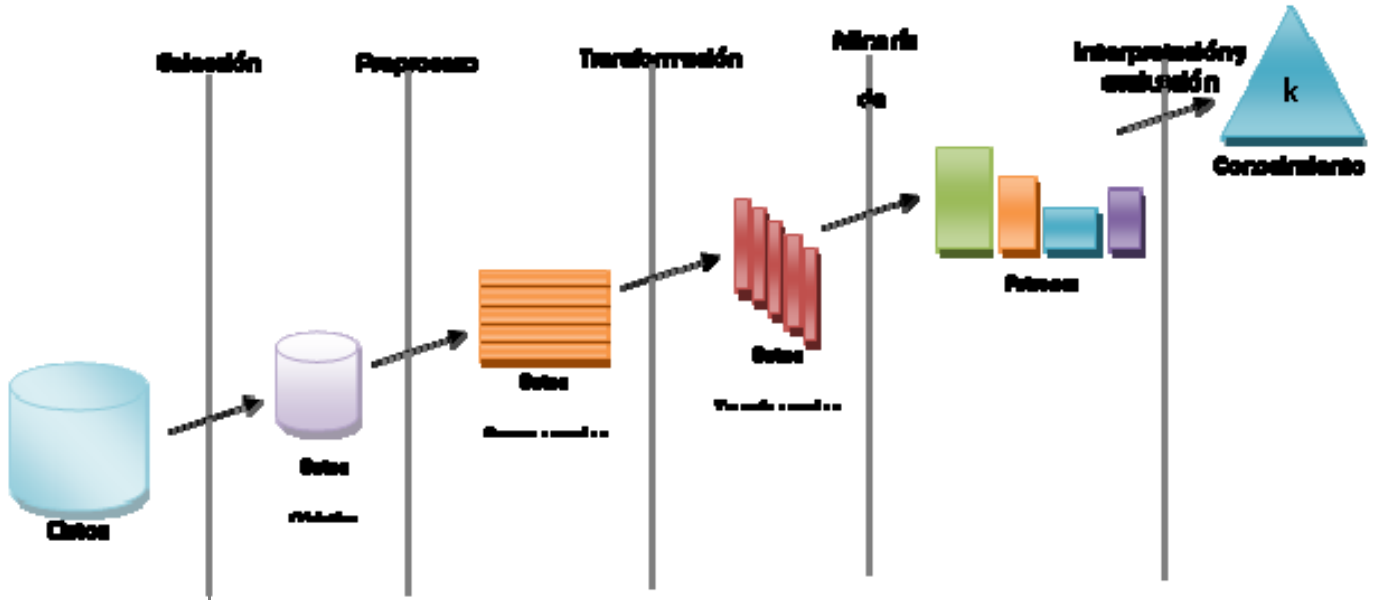
Metas de KDD (*Knowledge Discovery in Databases*)

1. Procesar automáticamente grandes cantidades de datos crudos.
2. Identificar patrones más significativos y relevantes.
3. Presentar los patrones identificados como conocimiento apropiado para satisfacer las metas del usuario.

Proceso de KDD (*Knowledge Discovery in Databases*)

1. *Filtración de Datos*. Eliminar ruido e inconsistencia en los datos.
2. *Integración de Datos*. Múltiples fuentes de datos pueden ser combinados.
3. *Selección de Datos*. Los datos relevantes para la tarea de análisis son recuperados desde la base de datos.
4. *Transformación de Datos*. Los datos son transformados o consolidados en formas apropiadas para la minería.
5. *Minería de Datos*. Es el proceso esencial donde métodos inteligentes son aplicados para extraer patrones de datos.
6. *Evaluación de Patrones*. Identificar patrones verdaderamente interesantes representando el conocimiento basado en algunas medidas de interés.
7. *Representación del Conocimiento*. Las técnicas de representación de conocimiento son usadas para presentar el conocimiento extraído al usuario.

Proceso de KDD (*Knowledge Discovery in Databases*)



Data Mining task

Prediction Methods
– Use some variables to predict unknown or future values of other variables.
Description Methods
– Find human-interpretable patterns that describe the data.

Classification [Predictive]
Clustering [Descriptive]
Association Rule Discovery [Descriptive]
Sequential Pattern Discovery [Descriptive]
Regression [Predictive]
Deviation Detection [Predictive]

Data Mining task (Classification:definition)

Given a collection of records (*training set*)
– Each record contains a set of *attributes*, one of the attributes is the *class*.
Find a *model* for class attribute as a function of the values of other attributes.
Goal: previously unseen records should be assigned a class as accurately as possible.
– A *test set* is used to determine the accuracy of the model. Usually, the given data set is divided into training and test sets, with training set used to build the model and test set used to validate it.

Data Mining task (Clustering:definition)

Given a set of data points, each having a set of attributes, and a similarity measure among them, find clusters such that

– Data points in one cluster are more similar to one another.

– Data points in separate clusters are less similar to one another.

Similarity Measures:

– Euclidean Distance if attributes are continuous.

– Other Problem-specific Measures.

Data Mining task (Association Rule Discovery:definition)

Given a set of records each of which contain some number of items from a given collection;

– Produce dependency rules which will predict occurrence of an item based on occurrences of other items.

TID	*Items*
1	Bread, Coke, Milk
2	Beer, Bread
3	Beer, Coke, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Coke, Diaper, Milk

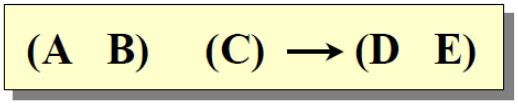
Rules Discovered:

{Milk} --> {Coke}

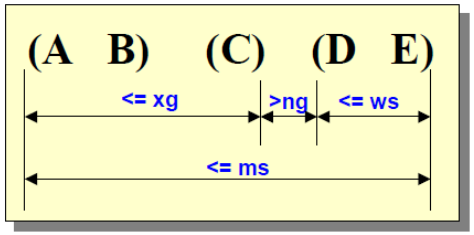
{Diaper, Milk} --> {Beer}

Data Mining task (Secuential Pattern Discovery:definition)

Given is a set of *objects*, with each object associated with its own *timeline of events*, find rules that predict strong sequential dependencies among different events.



Rules are formed by first discovering patterns. Event occurrences in the patterns are governed by timing constraints.



Data Mining task (Regresion:definition)

Predict a value of a given continuous valued variable based on the values of other variables, assuming a linear or nonlinear model of dependency.
Greatly studied in statistics, neural network fields.
Examples:
– Predicting sales amounts of new product based on advetising expenditure.
– Predicting wind velocities as a function of temperature, humidity, air pressure, etc.
– Time series prediction of stock market indices.

ID3 ALGORITHM

ID3 ALGORITHM

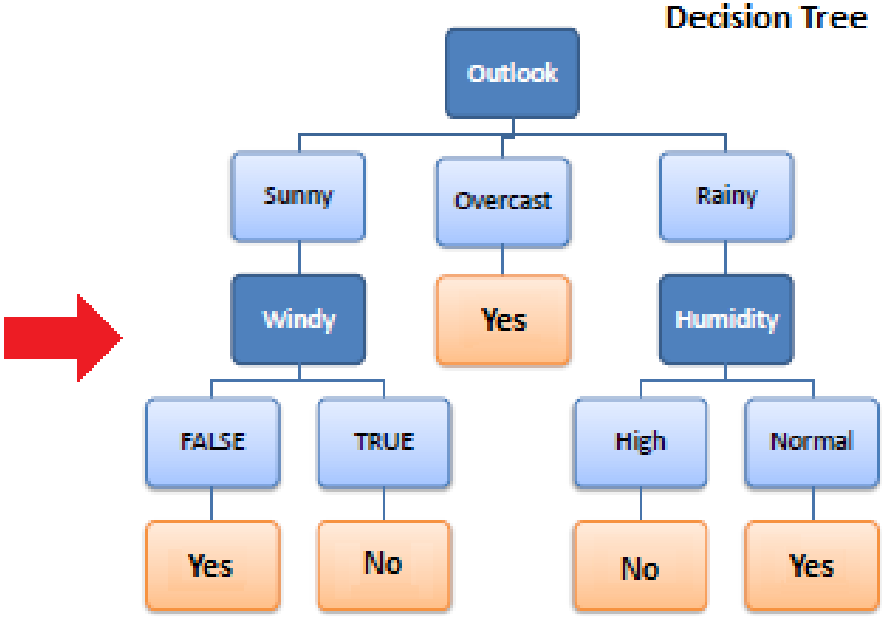
Ross Quinlan invented the Iterative Dichotomiser 3 (ID3) algorithm which is used to generate decision trees.

The core algorithm for building decision trees called **ID3** by J. R. Quinlan which employs a top-down, greedy search through the space of possible branches with no backtracking.

ID3 uses *Entropy* and *Information Gain* to construct a decision tree.

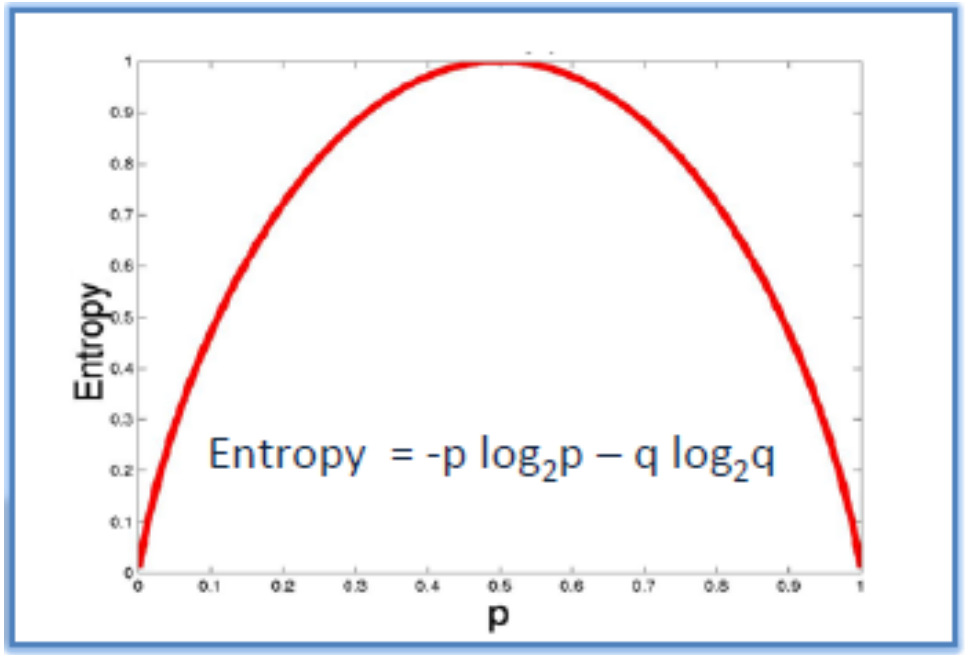
Predictors | Target

Outlook	Temp.	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No



ENTROPY

- **A decision tree is built top-down from a root node and involves partitioning the data into subsets that contain instances with similar values (homogenous). ID3 algorithm uses entropy to calculate the homogeneity of a sample. If the sample is completely homogeneous the entropy is zero and if the sample is an equally divided it has entropy of one.**



Entropy = $-0.5 \log_2 0.5 - 0.5 \log_2 0.5 = 1$

ENTROPY

- To build a decision tree, we need to calculate two types of entropy using frequency tables as follows:
- a) Entropy using the frequency table of one attribute:

$$E(S) = \sum_{i=1}^{c} - p_i \log_2 p_i$$

Play Golf	
Yes	No
9	5

Entropy(PlayGolf) = Entropy (5,9)

= Entropy (0.36, 0.64)

= - (0.36 $\log_2$ 0.36) - (0.64 $\log_2$ 0.64)

= 0.94

ENTROPY

- b) Entropy using the frequency table of two attributes:

$$E(T,X) = \sum_{c \in X} P(c)E(c)$$

		Play Golf		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5
				14

E(PlayGolf, Outlook) = **P**(Sunny)***E**(3,2) + **P**(Overcast)***E**(4,0) + **P**(Rainy)***E**(2,3)

= (5/14)*0.971 + (4/14)*0.0 + (5/14)*0.971

= 0.693

Information Gain

- **The information gain is based on the decrease in entropy after a dataset is split on an attribute. Constructing a decision tree is all about finding attribute that returns the highest information gain (i.e., the most homogeneous branches).**

Step 1

Calculate entropy of the target.

$$
\begin{aligned}
\textbf{Entropy(PlayGolf)} &= \text{Entropy}\ (5,9) \\
&= \text{Entropy}\ (0.36,\ 0.64) \\
&= -\ (0.36 \log_2 0.36) - (0.64 \log_2 0.64) \\
&= 0.94
\end{aligned}
$$

Step 2

The dataset is then split on the different attributes.

- The entropy for each branch is calculated. Then it is added proportionally, to get total entropy for the split. The resulting entropy is subtracted from the entropy before the split.

		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3
Gain = 0.247			

		Play Golf	
		Yes	No
Temp.	Hot	2	2
	Mild	4	2
	Cool	3	1
Gain = 0.029			

		Play Golf	
		Yes	No
Humidity	High	3	4
	Normal	6	1
Gain = 0.152			

		Play Golf	
		Yes	No
Windy	False	6	2
	True	3	3
Gain = 0.048			

- The result is the Information Gain, or decrease in entropy.

$$Gain(T, X) = Entropy(T) - Entropy(T, X)$$

G(PlayGolf, Outlook) = **E**(PlayGolf) – **E**(PlayGolf, Outlook)

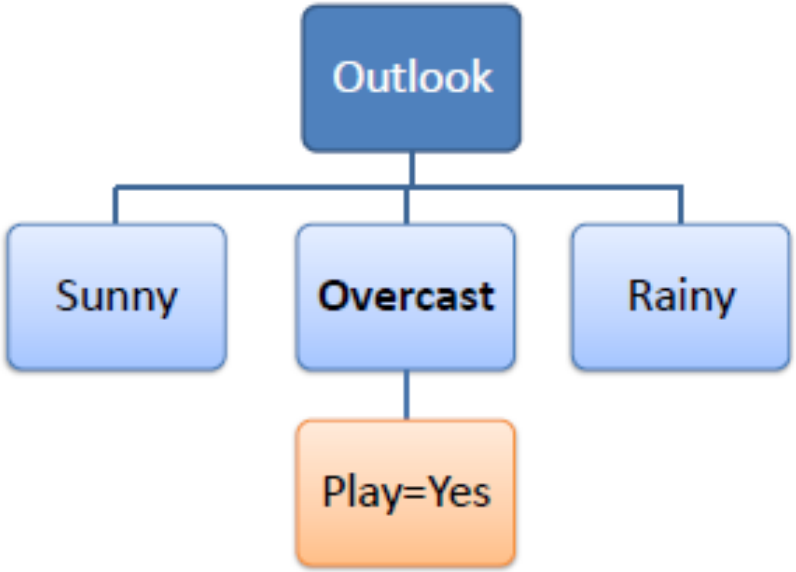
= 0.940 – 0.693 = 0.247

Step 3

Choose attribute with the largest information gain as the decision node.

		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3
Gain = 0.247			

Temp	Humidity	Windy	Play Golf
Hot	High	FALSE	Yes
Cool	Normal	TRUE	Yes
Mild	High	TRUE	Yes
Hot	Normal	FALSE	Yes
Hot	High	FALSE	Yes



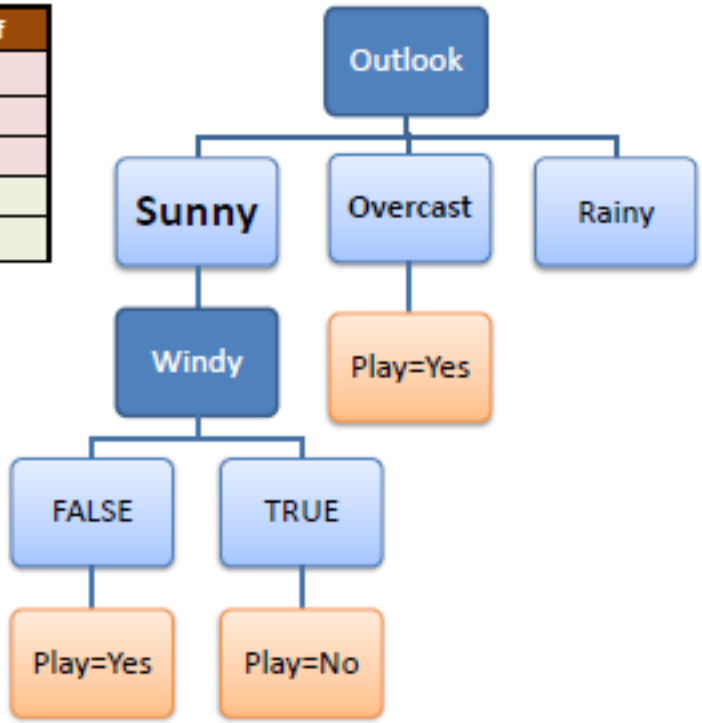
Step 4a:

A branch with entropy of 0 is a leaf node.

Step 4b

A branch with entropy more than 0 needs further splitting.

Temp.	Humidity	Windy	Play Golf
Mild	High	FALSE	Yes
Cool	Normal	FALSE	Yes
Mild	Normal	FALSE	Yes
Cool	Normal	TRUE	No
Mild	High	TRUE	No



Step 5

The ID3 algorithm is run recursively on the non-leaf branches, until all data is classified.

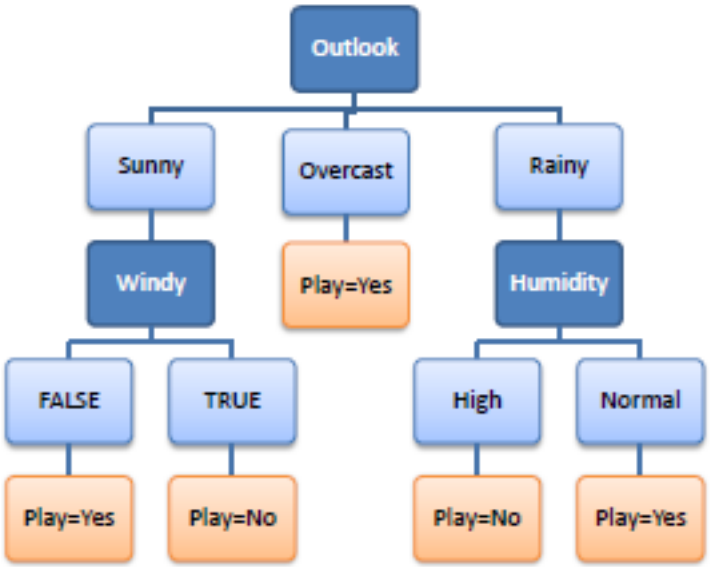
Decision Tree to Decision Rules

- A decision tree can easily be transformed to a set of rules by mapping from the root node to the leaf nodes one by one.

R_1: IF (Outlook=Sunny) AND (Windy=FALSE) THEN Play=Yes

R_2: IF (Outlook=Sunny) AND (Windy=TRUE) THEN Play=No

R_3: IF (Outlook=Overcast) THEN Play=Yes

R_4: IF (Outlook=Rainy) AND (Humidity=High) THEN Play=No

R_5: IF (Outlook=Rain) AND (Humidity=Normal) THEN Play=Yes