UNIREMINGTON®
CORPORACIÓN UNIVERSITARIA REMINGTON
RES. 2661 MEN JUNIO 21 DE 1996

BASE DE DATOS II

INGENIERIA DE SISTEMAS

FACULTAD DE CIENCIAS BÁSICAS E INGENIERÍA

Vicerrectoría de Educación a Distancia y virtual

2016

AUTOR

Luisa María Jiménez Ramos
Ingeniera de Sistemas – Especialista en Gerencia Informática.
luisa.jimenez@uniremington.edu.co

Nota: el autor certificó (de manera verbal o escrita) No haber incurrido en fraude científico, plagio o vicios de autoría; en caso contrario eximió de toda responsabilidad a la Corporación Universitaria Remington, y se declaró como el único responsable.

RESPONSABLES

Jorge Mauricio Sepúlveda Castaño
Decano de la Facultad de Ciencias Básicas e Ingeniería
jsepulveda@uniremington.edu.co

Eduardo Alfredo Castillo Builes
Vicerrector modalidad distancia y virtual
ecastillo@uniremington.edu.co

Francisco Javier Álvarez Gómez
Coordinador CUR-Virtual
falvarez@uniremington.edu.co

GRUPO DE APOYO

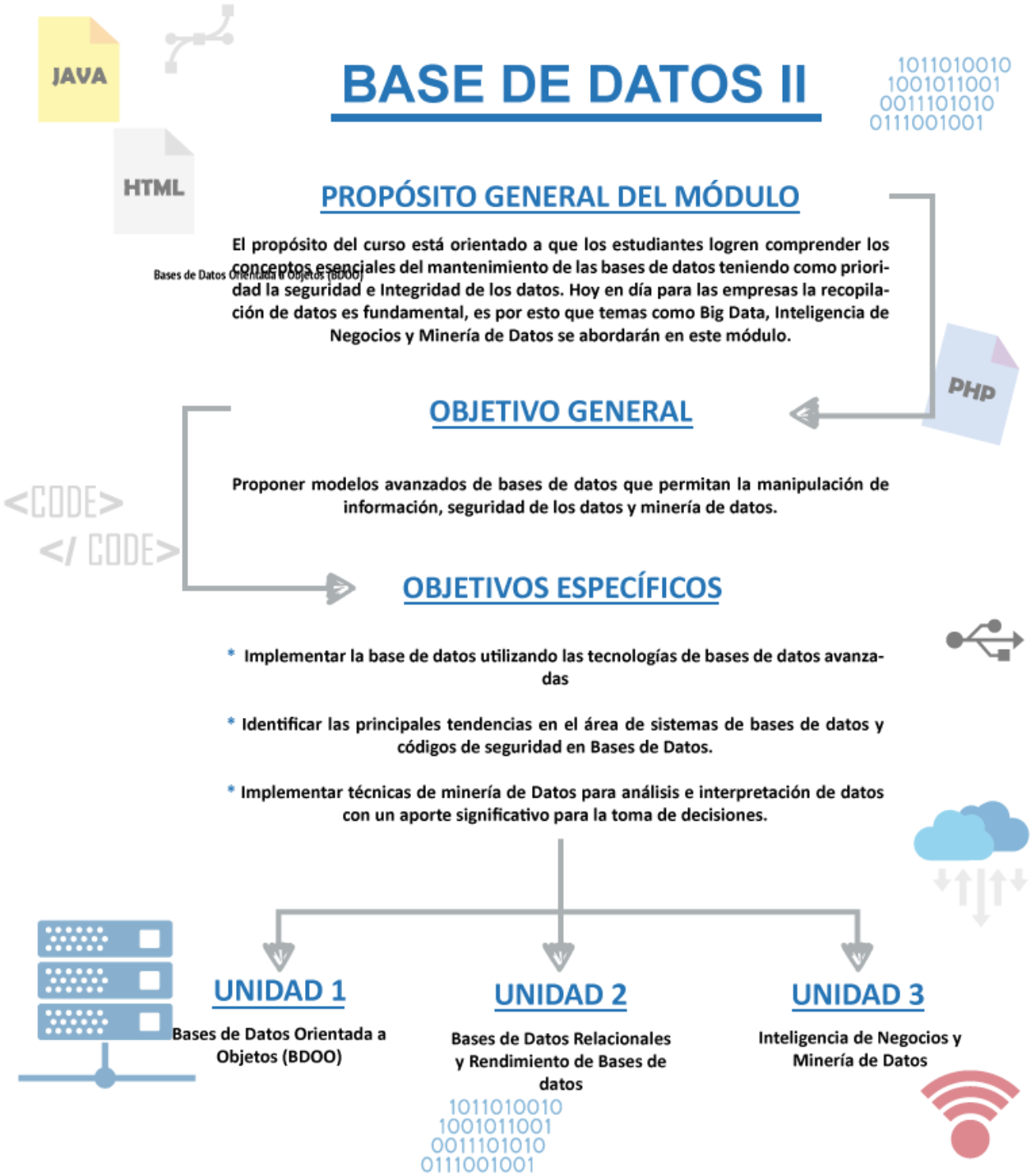
Personal de la Unidad CUR-Virtual
EDICIÓN Y MONTAJE
Primera versión. Febrero de 2011.
Segunda versión. Marzo de 2012
Tercera versión. noviembre de 2015
Cuarta Versión.2016

.

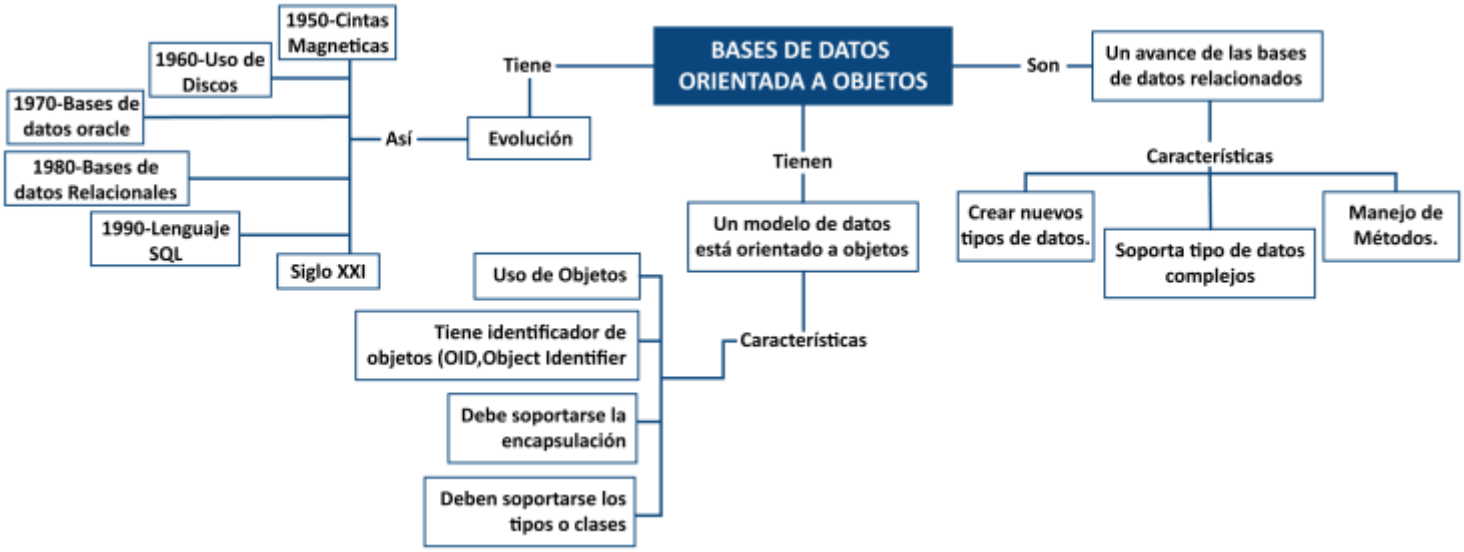
TABLA DE CONTENIDO

Pág.

1 MAPA DE LA ASIGNATURA



2 UNIDAD 1 BASES DE DATOS ORIENTADA A OBJETOS BDOO



Fuente: Propia.

CONCEPTOS BÁSICOS:

1. **Bases de datos Orientada a Objetos**: Base de datos que tiene Un modelo de datos está orientado a objetos y almacenan y recuperan objetos en los que se almacena estado y comportamiento.

2. **Paradigma Orientado a Objetos**: Es un método de implementación en el que los programas se organizan como colecciones cooperativas de objetos, cada uno de los cuales representa una instancia de alguna clase, y cuyas clases son miembros de una jerarquía de clases unidas mediante relaciones de herencia.

3. **Clase:** Plantilla implementada en software que describe un conjunto de objetos con atributos y comportamiento similares.

2.1 TEMA 1 EVOLUCIÓN DE LAS BASES DE DATOS

Como actividad introductoria se sugiere documentarse con los siguientes videos y enlaces:

Enlace: http://basesdedatosfecajoja.blogspot.com.co/2011/03/evolucion-historica.html

BASE DE DATOS , HISTORIA Y EVOLUCION: Enlace

Historia de las Bases de Datos: Enlace

2.1.1.1 EVOLUCIÓN DE LAS BASES DE DATOS

La evolución de las bases de datos se ha visto permeada por los avances de tecnológicos de cada época, es por eso que desde estos cambios se puede resaltar las siguientes décadas con los siguientes aportes.

DECADA	APORTE
Década de 1950	En este lapso de tiempo se da origen a las cintas magnéticas, las cuales sirvieron para suplir las necesidades de información de las nuevas industrias. Por medio de este mecanismo se empezó a automatizar la información de las nóminas, como por ejemplo el aumento de salario. Consistía en leer una cinta o más y pasar los datos a otra, y también se podían pasar desde las tarjetas perforadas. Simulando un sistema de Backup, que consiste en hacer una copia de seguridad o copia de respaldo, para guardar en un medio extraíble la información importante. La nueva cinta a la que se transfiere la información pasa a ser una cinta maestra. Estas cintas solo se podían leer secuencial y ordenadamente.
Década de 1960	El uso de los discos en ese momento fue un adelanto muy efectivo, ya que por medio de este soporte se podía consultar la información directamente, esto ayudo a ahorrar tiempo. No era necesario saber exactamente donde estaban los datos en los discos, ya que en milisegundos era recuperable la información. A diferencia de las cintas magnéticas, ya no era necesaria la secuencialidad, y este tipo de soporte empieza a ser ambiguo.
	Los discos dieron inicio a las Bases de Datos, de red y jerárquicas, pues los programadores con su habilidad de manipulación de estructuras junto con las ventajas de los discos era posible guardar estructuras de datos como listas y árboles.
Década de 1970	Edgar Frank Codd (23 de agosto de 1923 – 18 de abril de 2003), en un artículo "Un modelo relacional de datos para grandes bancos de datos compartidos" ("A Relational Model of Data for Large Shared Data Banks") en 1970, definió el modelo relacional y publicó una serie de reglas para la evaluación de administradores de sistemas de datos relacionales y así nacieron las bases de datos relacionales.
	A partir de los aportes de Codd el multimillonario Larry Ellison desarrollo la base de datos Oracle, el cual es un sistema de administración de base de datos, que se destaca por sus transacciones, estabilidad, escalabilidad y multiplataforma.

	Inicialmente no se usó el modelo relacional debido a que tenía inconvenientes por el rendimiento, ya que no podían ser competitivas con las bases de datos jerárquicas y de red. Ésta tendencia cambio por un proyecto de IBM el cual desarrolló técnicas para la construcción de un sistema de bases de datos relacionales eficientes, llamado System R.
Década de 1980	Las bases de datos relacionales con su sistema de tablas, filas y columnas, pudieron competir con las bases de datos jerárquicas y de red, ya que su nivel de programación era bajo y su uso muy sencillo.
	En esta década el modelo relacional ha conseguido posicionarse del mercado de las bases de datos. Y también en este tiempo se iniciaron grandes investigaciones paralelas y distribuidas, como las bases de datos orientadas a objetos.
Principios década de los 90	Para la toma de decisiones se crea el lenguaje SQL, que es un lenguaje programado para consultas. El programa de alto nivel SQL es un lenguaje de consulta estructurado que analiza grandes cantidades de información el cual permite especificar diversos tipos de operaciones frente a la misma información, a diferencia de las bases de datos de los 80 que eran diseñadas para las aplicaciones de procesamiento de transacciones. Los grandes distribuidores de bases de datos incursionaron con la venta de bases de datos orientada a objetos.
Finales de la década de los 90	El boom de esta década fue la aparición de la WWW "Word Wide Web" ya que por éste medio se facilitaba la consulta de las bases de datos. Actualmente tienen una amplia capacidad de almacenamiento de información, también una de las ventajas es el servicio de siete días a la semana las veinticuatro horas del día, sin interrupciones a menos que haya planificaciones de mantenimiento de las plataformas o el software.
Siglo XXI	En la actualidad existe gran cantidad de alternativas en línea que permiten hacer búsquedas orientadas a necesidades específicas de los usuarios, una de las tendencias más amplias son las bases de datos que cumplan con el protocolo Open Archives Initiative – Protocol for Metadata Harvesting (OAI-PMH) los cuales permiten el

almacenamiento de gran cantidad de artículos que permiten una mayor visibilidad y acceso en el ámbito científico y general.

Información tomada de: http://basesdedatosfecajoja.blogspot.com.co/2011/03/evolucion-historica.html

2.1.1.2 BASES DE DATOS ACTIVAS

En muchas aplicaciones, la base de datos debe evolucionar independientemente de la intervención del usuario como respuesta a un suceso o una determinada situación. En los sistemas de gestión de bases de datos tradicionales (pasivas), la evolución de la base de datos se programa en el código de las aplicaciones, mientras que en los sistemas de gestión de bases de datos activas esta evolución es autónoma y se define en el esquema de la base de datos. El poder especificar reglas con una serie de acciones que se ejecutan automáticamente cuando se producen ciertos eventos, es una de las mejoras de los sistemas de gestión de bases de datos que se consideran de gran importancia desde hace algún tiempo.

Mediante estas reglas se puede hacer respetar reglas de integridad, generar datos derivados, controlar la seguridad o implementar reglas de negocio. De hecho, la mayoría de los sistemas relacionales comerciales disponen de disparadores (triggers). Se ha hecho mucha investigación sobre lo que debería ser un modelo general de bases de datos activas desde que empezaron a aparecer los primeros disparadores.

El modelo que se viene utilizando para especificar bases de datos activas es el modelo evento–condición–acción. Mediante los sistemas de bases de datos activas se consigue un nuevo nivel de independencia de datos: la independencia de conocimiento. El conocimiento que provoca una reacción se elimina de los programas de aplicación y se codifica en forma de reglas activas. De este modo, al encontrarse las reglas definidas como parte del esquema de la base de datos, se comparten por todos los usuarios, en lugar de estar replicadas en todos los programas de aplicación. Cualquier cambio sobre el comportamiento reactivo se puede llevar a cabo cambiando solamente las reglas activas, sin necesidad de modificar las aplicaciones. Además, mediante los sistemas de bases de datos activas se hace posible el integrar distintos subsistemas (control de accesos, gestión de vistas, etc.) y se extiende el ámbito de aplicación de la tecnología de bases de datos a otro tipo de aplicaciones. Uno de los problemas que ha limitado el uso extensivo de reglas activas, a pesar de su potencial para simplificar el desarrollo de bases de datos y de aplicaciones, es el hecho de que no hay técnicas fáciles de usar para diseñar, escribir y verificar reglas. Por ejemplo, es bastante difícil verificar que un conjunto de reglas es consistente, es decir, que no se contradice. También es difícil garantizar la terminación de un conjunto de reglas bajo cualquier circunstancia. Para que las reglas activas alcancen todo su potencial, es necesario desarrollar herramientas para diseñar, depurar y monitorizar reglas activas que puedan ayudar a los usuarios en el diseño y depuración de sus reglas. (Marqués, 2002)

Una base de datos activa, son aquellas bases de datos capaz de detectar situaciones de interés y de actuar en consecuencia. (Mota, 2005). El mecanismo que se utiliza se parece a las reglas de producción utilizadas en el área de inteligencia artificial.

Tal y como se ha mencionado anteriormente Un sistema de bases de datos activas es un sistema de gestión de bases de datos (SGBD) que contiene un subsistema que permite la definición y la gestión de reglas de producción (reglas activas). Las reglas siguen el modelo evento–condición–acción (modelo ECA): cada regla reacciona ante un determinado evento, evalúa una condición y, si ´esta es cierta, ejecuta una acción. La ejecución de las reglas tiene lugar bajo el control de un subsistema autónomo, denominado motor de reglas, que se encarga de detectar los eventos que van sucediendo y de planificar las reglas para que se ejecuten.

PASOS PARA LA CONSTRUCCIÓN DE UN MODELO E – C- A (EVENTO- CONDICIÓN - ACCIÓN)

En el modelo ECA una regla tiene tres componentes:

1. El evento (o eventos) que dispara la regla. Estos eventos pueden ser operaciones de consulta o actualización que se aplican explícitamente sobre la base de datos. También pueden ser eventos temporales (por ejemplo, que sea una determinada hora del día) u otro tipo de eventos externos (definidos por el usuario).

2. La condición que determina si la acción de la regla se debe ejecutar. Una vez ocurre el evento disparador, se puede evaluar una condición (es opcional). Si no se especifica condición, la acción se ejecutará cuando suceda el evento. Si se especifica condición, la acción se ejecutará sólo si la condición se evalúa a verdadero.

3. La acción a realizar puede ser una transacción sobre la base de datos o un programa externo que se ejecutará automáticamente.

Casi todos los sistemas relacionales incorporan reglas activas simples denominadas disparadores (triggers), que están basados en el modelo ECA: - Los eventos son sentencias SQL de manejo de datos (INSERT, DELETE, UPDATE). - La condición (que es opcional) es un predicado booleano expresado en SQL. - La acción es una secuencia de sentencias SQL, que pueden estar inmersas en un lenguaje de programación integrado en el producto que se esté utilizando (por ejemplo, PL/SQL en Oracle).

El modelo ECA se comporta de un modo simple e intuitivo: cuando ocurre el evento, si la condición es verdadera, entonces se ejecuta la acción. Se dice que el disparador es activado por el evento, es considerado durante la verificación de su condición y es ejecutado si la condición es cierta. Sin embargo, hay diferencias importantes en el modo en que cada sistema define la activación, consideración y ejecución de disparadores.

Los disparadores relacionales tienen dos niveles de granularidad: a nivel de fila y a nivel de sentencia. En el primer caso, la activación tiene lugar para cada tupla involucrada en la operación y se dice que el sistema tiene un comportamiento orientado a tuplas. En el segundo caso, la activación tiene lugar sólo una vez para cada sentencia

SQL, refiriéndose a todas las tuplas invocadas por la sentencia, con un comportamiento orientado a conjuntos. Además, los disparadores tienen funcionalidad inmediata o diferida. La evaluación de los disparadores inmediatos normalmente sucede inmediatamente después del evento que lo activa (opción después), aunque también puede precederlo (opción antes) o ser evaluados en lugar de la ejecución del evento (opción en lugar de). La evaluación diferida de los disparadores tiene lugar al finalizar la transacción en donde se han activado (tras la sentencia COMMIT).

Un disparador puede activar otro disparador. Esto ocurre cuando la acción de un disparador es también el evento de otro disparador. En este caso, se dice que los disparadores se activan en cascada. (Universidad de Zulia, 2010).

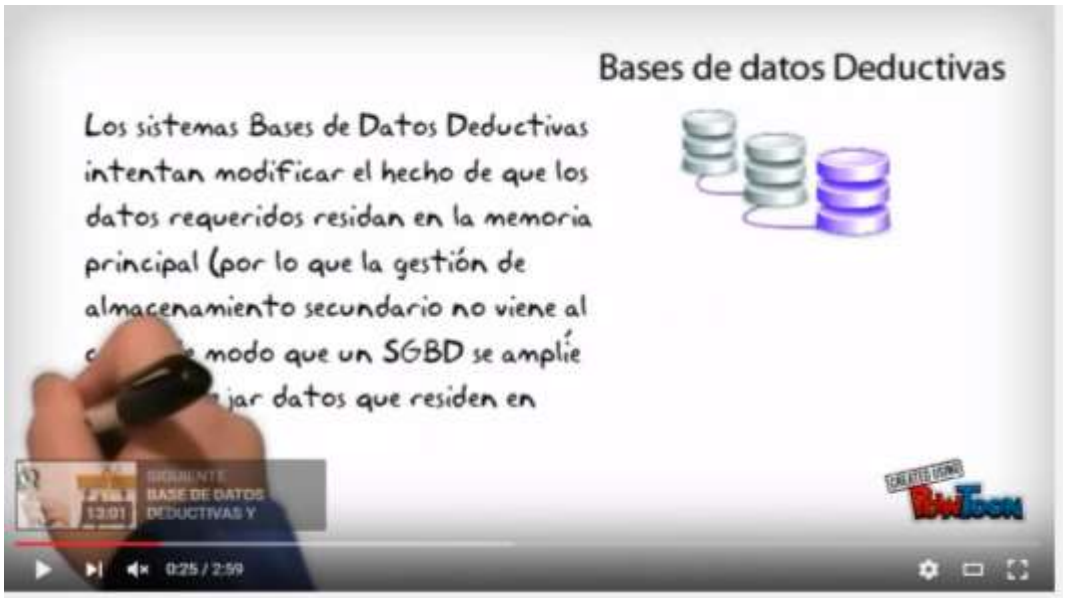
2.1.2 TALLER DE ENTRENAMIENTO

EVOLUCION DE BASES DE DATOS Y BASES DE DATOS ACTIVAS.

El siguiente taller de entrenamiento se propone para validar la aprehensión de los conceptos. Se debe leer detenidamente las preguntas propuestas y dar respuesta a ellas.

1. Enuncie los pasos para la construcción de un modelo E-C-A
2. Enuncie los aportes más importantes en la evolución de las bases de datos.

Para afianzar sus conocimientos se recomienda revisar los siguientes links:



Base de datos activa y difusa : Enlace

Enlace: https://equipo01.wordpress.com/2010/02/20/modelos-relacional/

http://www.academia.edu/3744541/Aplicaci%C3%B3n_de_Bases_de_Datos_Activas_en_un_sistema_bancario

2.2 TEMA 2 BASE DE DATOS ORIENTADA A OBJETOS Y FUNCIONALIDADES

2.2.1.1 BASES DE DATOS ORIENTADA A OBJETOS

Las bases de datos orientadas a objetos (BDOO) son aquellas cuyo modelo de datos está orientado a objetos y almacenan y recuperan objetos en los que se almacena estado y comportamiento. Su origen se debe a que en los modelos clásicos de datos existen problemas para representar cierta información, puesto que aunque permiten representar gran cantidad de datos, las operaciones que se pueden realizar con ellos son bastante simples.

Las clases utilizadas en un determinado lenguaje de programación orientado a objetos son las mismas clases que serán utilizadas en una BDOO; de tal manera, que no es necesaria una transformación del modelo de objetos para ser utilizado por un SGBDOO. De forma contraria, el modelo relacional requiere abstraerse lo suficiente como para adaptar los objetos del mundo real a tablas.

Las bases de datos orientadas a objetos surgen para evitar los problemas que surgen al tratar de representar cierta información, aprovechar las ventajas del paradigma orientado a objetos en el campo de las bases de datos y para evitar transformaciones entre modelos de datos (usar el mismo modelo de objetos). (Aberca, A., Galvez, J., 2008).

Para tener en cuenta:

Los conceptos relacionados con las bases de datos orientados son:

1. Base de datos orientada a objetos (BDOO): una colección persistente y compatible de objetos definida por un modelo de datos orientado a objetos.

2. Modelo de datos orientado a objetos: Un modelo de datos que captura la semántica de los objetos soportados en la programación orientada a objetos.

3. Sistema Gestor de Bases de Datos Orientadas a Objetos (SGBDOO): El gestor de una base de datos orientada a objetos.

En el siguiente link puede ampliar información sobre el origen de las bases de datos orientada a objetos:

Base de datos orientada a objeto: Enlace

2.2.1.2 CARACTERÍSTICAS Y DIFERENCIAS CON RESPECTO A LAS BASES DE DATOS RELACIONALES (BDR)

BASES DE DATOS ORIENTADA A OBJETOS	BASES DE DATOS RELACIONALES
Uso de objetos: cada entidad del mundo real se modela como un objeto.	**No usan objetos.** Los datos se almacenan representados en tablas.
La forma de identificar objetos es mediante un identificador de objetos (OID, Object Identifier), único para cada objeto. Generalmente este identificador no es accesible ni modificable para el usuario (modo de aumentar la integridad de entidades y la integridad referencial). Los OID son independientes del contenido. Es decir, si un objeto cambia los valores de atributos, sigue siendo el mismo objeto con el mismo OID. Si dos objetos tienen el mismo estado pero diferentes OID, son equivalentes pero tienen identidades diferentes.	N/A

Tienen el mismo comportamiento de las propiedades de la programación orientada o objetos, es decir: 1. **Encapsulamiento.** 2. **Clases** 3. **Herencia** 4. **Polimorfismo**	N/A

Fuente: Propia

Las características de una BDOO están definidas por el manifiesto Malcolm Atkinson, este se hizo en 1989 el cual propuso trece características de obligatoriedad para un SGBDOO y cuatro opcionales. Las trece características obligatorias estaban basadas en dos criterios: debía tratarse de un sistema orientado a objetos y un SGBD.

Características obligatorias de orientación a objetos:

1. Deben soportarse objetos complejos.

2. Deben soportarse mecanismos de identidad de los objetos.

3. Debe soportarse la encapsulación.

4. Deben soportarse los tipos o clases

5. Los tipos o clases deben ser capaces de heredar de sus ancestros.

6. Debe soportarse el enlace dinámico.

7. El DML debe ser computacionalmente complejo.

8. El conjunto de todos los tipos de datos debe ser ampliable.

Características obligatorias de SGBD:

9. Debe proporcionarse persistencia a los datos.

10. El SGBD debe ser capaz de gestionar bases de datos de muy gran tamaño.

11. El SGBD debe soportar a usuarios concurrentes.

12. El SGBD debe ser capaz de recuperarse de fallos hardware y software.

13. El SGBD debe proporcionar una forma simple de consultar los datos.

Características opcionales:

1. Herencia múltiple.

2. Comprobación de tipos e inferencia de tipos.

3. Sistema de base de datos distribuido.

4. Soporte de versiones.

2.2.1.3 VENTAJAS Y DESVENTAJAS DE LAS BDOO

VENTAJAS	DESVENTAJAS
Mayor capacidad de modelado. El modelado de datos orientado a objetos permite modelar el 'mundo real' de una manera mucho más fiel. Esto se debe a: o un objeto permite encapsular tanto un estado como un comportamiento o un objeto puede almacenar todas las relaciones que tenga con otros objetos o los objetos pueden agruparse para formar objetos complejos (herencia).	Carencia de un modelo de datos universal. No hay ningún modelo de datos que esté universalmente aceptado para los SGBDOO y la mayoría de los modelos carecen una base teórica.
Ampliabilidad. Esto se debe a: o Se pueden construir nuevos tipos de datos a partir de los ya existentes. O Agrupación de propiedades comunes de diversas clases e incluirlas en una superclase, lo que reduce la redundancia. O Reusabilidad de clases, lo que repercute en una mayor	Carencia de experiencia. Todavía no se dispone del nivel de experiencia del que se dispone para los sistemas tradicionales.

facilidad de mantenimiento y un menor tiempo de desarrollo.	
Lenguaje de consulta más expresivo. El acceso navegacional desde un objeto al siguiente es la forma más común de acceso a datos en un SGBDOO. Mientras que SQL utiliza el acceso asociativo. El acceso navegacional es más adecuado para gestionar operaciones como los despieces, consultas recursivas, etc.	Existe una carencia de estándares general para los SGBDOO.
Adecuación a las aplicaciones avanzadas de base de datos. Hay muchas áreas en las que los SGBD tradicionales no han tenido excesivo éxito como el CAD, CASE, OIS, sistemas multimedia, etc. en los que las capacidades de modelado de los SGBDOO han hecho que esos sistemas sí resulten efectivos para este tipo de aplicaciones.	Competencia. Con respecto a los SGBDR y los SGBDOR. Estos productos tienen una experiencia de uso considerable. SQL es un estándar aprobado y ODBC es un estándar de facto. Además, el modelo relacional tiene una sólida base teórica y los productos relacionales disponen de muchas herramientas de soporte que sirven tanto para desarrolladores como para usuarios finales.
Mayores prestaciones. Los SGBDOO proporcionan mejoras significativas de rendimiento con respecto a los SGBD relacionales. Aunque hay autores que han argumentado que los mejor que los SGBDOO en las aplicaciones tradicionales de bases de datos como el procesamiento de transacciones en línea (OLTP). Bancos de prueba usados están dirigidos a aplicaciones de ingeniería donde los SGBDOO son más adecuados. También está demostrado que los SGBDR tienen un rendimiento	La optimización de consultas compromete la encapsulación. La optimización de consultas requiere una compresión de la implementación de los objetos, para poder acceder a la base de datos de manera eficiente. Sin embargo, esto compromete el concepto de encapsulación.

Fuente: https://basededatos2010.wikispaces.com/file/view/BD+O-O+ventajas+y+desventajas.pdf

2.2.1.4 BASES DE DATOS OBJETO – RELACIONALES.

Una Base de Datos Objeto Relacional (BDOR) es una base de datos que desde el modelo relacional evoluciona hacia una base de datos más extensa y compleja incorporando para obtener este fin, conceptos del modelo orientado

a objetos. Podemos decir que un Sistema de Gestión Objeto-Relacional (SGBDOR) contiene dos tecnologías; la tecnología relacional y la tecnología de objetos.

En una base de datos objeto-relacional se siguen almacenando tuplas, aunque la estructura de las tuplas no está restringida a contener escalares (tipos compuestos como vectores, conjuntos, etc.) sino que las relaciones pueden ser definidas en función de otras, que es lo que denominamos herencia directa.

Características:

1. Permite crear nuevos tipos de dato.
2. Soporta tipo de datos complejos como: Listas, pilas, colas, arreglos.
3. Se puede crear expresiones en algún lenguaje de programación.
4. Se puede crear operadores.
5. Soporta encadenamiento dinámico.
6. Manejo de Métodos.
7. Soporta Herencia.

Nota: Para complementar este tema se sugiere revisar los siguientes enlaces:

Enlace: https://basededatos2010.wikispaces.com/file/view/BD+O-O+ventajas+y+desventajas.pdf

Base de datos orientada a objeto: Enlace

2.2.2 TALLER DE ENTRENAMIENTO

BASES DE DATOS ORIENTADA A OBJETOS

El siguiente taller de entrenamiento se propone para validar la aprehensión de los conceptos. Se debe leer detenidamente las preguntas propuestas y dar respuesta a ellas.

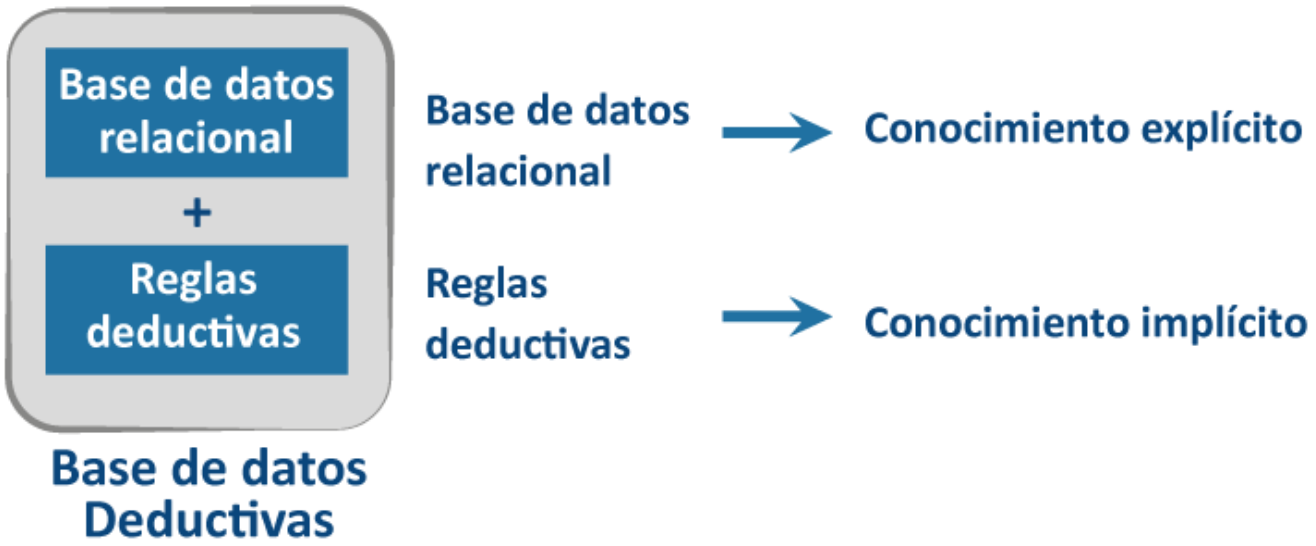
1. Enuncie las características de las BDOO

2. Enuncie diferencias entre BDOO y BDR

2.2.2.1 BASES DE DATOS DEDUCTIVA Y DIFUSAS

1. **Bases de datos Deductivas:** En el afán de ofrecer una respuesta a las necesidades planteadas por los usuarios y por las aplicaciones avanzadas, en donde se necesitan herramientas semánticamente más ricas que las provistas por las Bases de Datos Relacionales, aparecen recientes aplicaciones de los sistemas de bases de datos que consiste en ofrecer recursos para definir Reglas Deductivas que permitan deducir, inferir u obtener información nueva a partir de los datos almacenados.

La meta de estas aplicaciones es incorporar a las Bases de Datos Relacionales los beneficios de la lógica como instrumento para la formalización integrada de los aspectos estáticos y dinámicos del modelado de aplicaciones.



Fuente: Carolina Henao, 2011

Existen diversas clases de BDDs y para cada una de ellas existe una semántica bien definida. Las BDDs son muy usadas en las áreas de: inteligencia artificial, sistemas expertos, representación del conocimiento, tecnología de agentes, sistemas de información, integración de datos, por nombrar algunas.

Existe una importante relación entre BDDs y programación lógica. Una BDD es, en esencia, un programa lógico; mapeo de relaciones base hacia hechos, y reglas que son usadas para definir nuevas relaciones en términos de las relaciones base y el procesamiento de consultas.

Ventajas: Las principales ventajas al utilizar una BDD son las siguientes:

1. **Tener la capacidad de expresar consultas por medio de reglas lógicas.**
2. **Permitir consultas recursivas y algoritmos eficientes para su evaluación.**
3. **Contar con negaciones estratificadas.**
4. **Soportar objetos y conjuntos complejos.**
5. **Contar con métodos de optimización que garanticen la traducción de especificaciones dentro de planes eficientes de acceso.**

Como característica fundamental de una Base de Datos Deductiva es la posibilidad de inferir información a partir de los datos almacenados, es imperativo modelar la base de datos como un conjunto de fórmulas lógicas, las cuales permiten inferir otras fórmulas nuevas.

Inconvenientes: La explotación de las reglas de deducción en una BDD plantea algunos problemas:

1. **Encontrar criterios que permitan, para una ley dada; decidir su utilización como regla de deducción o como regla de coherencia.**
2. **Replantear correctamente, en un contexto deductivo, las convenciones habituales en una base de datos (representaciones de informaciones negativas, eficacia de las respuestas a las interrogaciones, cierre del dominio).**
3. **Desarrollar procedimientos eficaces de deducción. La posibilidad de caer en bucles infinitos es un problema muy importante. (Henao, 2011)**

Se sugiere tener en cuenta los siguientes link para trabajar detalladamente esta temática.

Enlace: http://www.academia.edu/5739624/BASES_DE_DATOS_DEDUCTIVAS_Y_BASES_DE_DATOS_DIFUSAS

BASE DE DATOS DEDUCTIVAS Y DIFUSAS: Enlace

2. Bases de datos difusas.

Una de las características del lenguaje natural, que hace difícil su utilización en sistemas computacionales es su imprecisión. Por ejemplo conceptos como pequeño o grande, tienen significados diferentes de acuerdo al contexto en el que se estén utilizando, e incluso dentro del mismo contexto, pueden significar cosas diferentes para diferentes individuos.

La teoría de los conjuntos difusos desarrollada por Zadeh, provee una poderosa herramienta para la representación y manejo de la imprecisión por lo que actualmente está siendo utilizada en varios campos para el diseño de sistemas basados en reglas difusas.

La teoría de conjuntos difusos, extiende la teoría clásica de conjuntos al permitir que el grado de pertenencia de un objeto a un conjunto sea representada como un número real entre 0 y 1 en vez del concepto clásico en el que solo se tiene la posibilidad de pertenecer a un conjunto o no pertenecer al mismo; en otras palabras, el grado de pertenencia a un conjunto en la teoría clásica tiene solo dos valores posibles: 0 y 1.

En el sentido más amplio, un sistema basado en reglas difusas es un sistema basado en reglas donde la lógica difusa es utilizada como una herramienta para representar diferentes formas de conocimiento acerca del problema a resolver, así como para modelar las interacciones y relaciones que existen entre sus variables. Debido a estas propiedades, los sistemas basados en reglas difusas han sido aplicados de forma exitosa en varios dominios en los que la información vaga o imprecisa emerge en diferentes formas. Actualmente, el modelo relacional no

permite el procesamiento de consultas del tipo "Encontrar a todos los gerentes cuyo sueldo no sea muy alto" dado que ni el cálculo ni el álgebra relacional, que establecen el resultado de cualquier consulta como una nueva relación, tienen la capacidad de permitir consultas de una manera difusa.

En los últimos años, algunos investigadores han lidiado con el problema de relajar el modelo relacional para permitirle admitir algunas imprecisiones; esto conduce a sistemas de bases de datos que encajan en el campo de la Inteligencia Artificial, ya que permiten el manejo de información con una terminología que es muy similar a la del lenguaje natural. Una solución que aparece recurrentemente en los trabajos de investigación actuales en esta área es la fusión de los sistemas manejadores de bases de datos relacionales con la lógica difusa, lo que da lugar a lo que se conoce como sistemas manejadores de bases de datos difusas o FRDBMS (por sus siglas en inglés, Fuzzy Relational Database Management System). (Henao, 2011)

Ventajas:

1. Almacenar Imprecisión: la información que tengamos de un atributo particular de un objeto, aunque esta información no sea el valor exacto. Suelen usar Etiquetas Lingüísticas con alguna definición asociada (por ejemplo, un conjunto difuso visto como una "Distribución de Posibilidad"), o sin ninguna definición asociada ("escalares" con una relación de similitud definida entre ellos).

2. Operar con esa información de forma coherente: (especialmente en las operaciones de consulta). Muchos autores estudian la consulta difusa en BD clásicas: (Tahani, 1977; Bosc et al., 1988, 1994, 1995; Wong, 1990; Galindo et. al., 1998a).

Inconvenientes:

1. Lenguaje de consulta incómodo, debido al gran número de parámetros que deben utilizarse.
2. Comparadores abstractos que hacen difícil la decisión de cuál debemos usar.
3. Falta de estandarización, derivado de la poca popularidad de este tipo de bases de datos. (Henao, 2011)

2.2.2.2 BASES DE DATOS MULTIMEDIA Y WEB

Este tipo de bases de datos tienen mucha relación con los SGBD objetos relacionales (SGBDOR) y los orientados a objetos (SGBDOO) ya que pueden almacenar tipos de datos multimedia. Podemos distinguir dos tipos de bases de datos multimedia fundamentales:

Bases de datos referenciales: son bancos de datos sobre material como películas, series de televisión o música. En la mayoría de los casos, la información que se almacena hace referencia a cuestiones descriptivas (autor, título, duración, productor, etc.) o a cuestiones técnicas (formato, duración, etc.).

Bases de datos descriptivas: se trata de sistemas de análisis de contenido que, más allá de los datos técnicos o generales que contiene la mayoría de bases de datos referenciales, aportan información específica sobre el contenido. Estos bancos de datos no resultan tan habituales y de hecho se encuentran en un estado de desarrollo embrionario, ya que el análisis de la imagen y del sonido no se halla tan automatizado como el del texto.

Existe, sin embargo, un número importante de bases de datos referenciales que actualmente se emplean tanto en entornos cerrados (por ejemplo, las bases de datos que gestionan las plataformas de televisión) como en redes abiertas del tipo Internet, que permiten una consulta en muchos casos gratuita y libre por parte de los usuarios. (Henao, 2011)

Ventajas E Inconvenientes de una Base de Datos Multimedia:

Ventajas:

1. La posibilidad de integrar en un único sistema una gran diversidad de formatos (imágenes, texto, video, sonido, etc).
2. Ofrecen mayor variedad a la hora de representar la información.
3. Un gran, y creciente, mercado potencial que augura que se siga investigando activamente en el futuro.

Inconvenientes

1. Necesita grandes espacios para almacenar toda la información que queremos. `
2. Este tipo de bases de datos necesitan grandes anchos de banda para obtener un rendimiento óptimo.
3. Complejidad en cuanto a programar operaciones, o incluso la interfaz, debido a la alta cantidad de formatos que hay que manejar, lo que puede repercutir en su rendimiento. (Henao, 2011)

Bases de datos Web:

La utilización de la World Wide Web (www) para presentar y acumular datos se ha desarrollado mucho más allá de la sencilla presentación de páginas, ya no se hacen los antiguos diseños web en los que los diseñadores creaban una página independiente para cada elemento de la colección que querían mostrar, esas páginas eran difíciles de mantener y de organizar.

Cabe hacer mención especial al gran número de aplicaciones a las que da soporte Internet, así como a la naturaleza de las mismas, ya que no son aplicaciones estáticas sino que están en constante renovación (esto hace que sea especialmente importante separar los datos con los que se trabaja de la aplicación que los gestiona). Todo ello influye en la forma de almacenar y organizar la información, debiendo de tener en cuenta todos estos factores a la hora de crear una BBDD para la web. Cualquier sitio web que presente información sobre un conjunto de elementos similares es candidato para la utilización de una base de datos web.

La solución general consiste en definir una base de datos, añadir un registro para cada elemento (directamente en la base de datos o dinámicamente por la web) y después consultar dicha base de datos para generar páginas web sobre la marcha. Una página de menú codificada en HyperText Markup Language (HTML) convierte en una consulta a una base de datos de varios registros. Esto supone una increíble ventaja sobre todo a la hora del mantenimiento ya que es más fácil tratar una base de datos que muchas páginas individuales. También un aumento de las capacidades del HTML ya que éste tiene muchas limitaciones. Las tecnologías web están reemplazando arquitecturas como la terminal o cliente-servidor, incluyendo servicios y servidores web y de base de datos entre ellos. (Henao, 2011)

Características De BBDD WEB: Los distintos Sistemas Gestores de Bases de Datos (SGBD) existentes incorporan en sus últimas versiones software de tipo middleware (capa de software que se sitúa sobre el SGBD) para añadir conectividad a la base de datos a través de Internet, por lo que realmente las bases de datos web no son más que SGBD utilizados y orientados con vistas a la web. Los middleware desarrollados en los distintos SGBD suelen emplear ODBC (traduce las consultas de datos de la aplicación en comandos que el SGBD entienda) para conectar

con la BD, junto con diversos conjuntos de herramientas para facilitar al usuario la implementación de la comunicación con la BD a través de Internet.

Las principales características que debe cumplir un SGBD utilizado en tecnología web son las siguientes:

1. Permitir acceso concurrente a los datos.
2. Ofrecer mecanismos de seguridad.
3. Soportar transacciones.
4. Permitir almacenar grandes volúmenes de datos, y almacenamiento de diferentes archivos.

2.2.2.3 ALMACENES DE DATOS Y BASES DE DATOS XML

XML es un metalenguaje (un lenguaje para describir otros lenguajes) que permite a los diseñadores crear sus propias etiquetas personalizadas para proporcionar funcionalidad no disponible en HTML.

Fue en 1998 cuando la W3C (World Wide Web Consortium) ratifico formalmente la primera versión de XML como un estándar de intercambio de datos. Existen dos modelos de datos principales a la hora de trabajar con XML.

Estos son el modelo centrado en los datos y el modelo centrado en los documentos. En un modelo centrado en los documentos, XML se utiliza como formato de almacenamiento e intercambio para datos que están estructurados.

En este caso, los datos podrían almacenarse en un SGBD relacional, objeto-relacional u orientado a objetos. Por tanto, para poder almacenar datos XML en SGBD tradicionales es necesario transformar las colecciones XML en esquemas compatibles con los SGBD tradicionales. Por ejemplo, XML ha sido completamente integrado en los sistemas Oracle9i, Oracle10g y Oracle11g a través de una extensión llamada Oracle XML DB. En el caso de las consultas SQL, en el estándar SQL: 2003 hay definidas una serie de extensiones a SQL que permiten la publicación de código XML. Estas extensiones son conocidas como SQL/XML.

En el caso de que XML se use para codificar datos semiestructurados, los SGBD tradicionales no podrán gestionarlos correctamente. Para realizar esto se necesitara un modelo centrado en los documentos. Para estos sistemas se usa una base de datos XML nativa (NXD, Native XML Database). Las NXD Definen un modelo de datos (lógico) para un documento XML (para el documento, no para los datos contenidos en el) y almacena y extrae documentos de acuerdo con dicho modelo. Ejemplos de estos modelos son el modelo de los datos de XPath, los XML Infoset, y los modelos explicitados por el DOM y los eventos en SAX 1.0.Se pueden distinguir dos tipos de NXD según su almacenamiento:

1. Basados en texto: almacenan el código XML como texto, por ejemplo como un archivo de un sistema de archivos o como un dato de tipo CLOB en un SGBD relacional.
2. Basados en modelo: almacenan el código XML en alguna representación interna en forma de árbol.

Middleware: es un software de computadora que conecta componentes de software o aplicaciones para que puedan intercambiar datos entre éstas. Es utilizado a menudo para soportar aplicaciones distribuidas. Esto incluye servidores web, servidores de aplicaciones, sistemas de gestión de contenido y herramientas similares.

Middleware es especialmente esencial para tecnologías como XML, SOAP, servicios web y arquitecturas orientada a servicios.

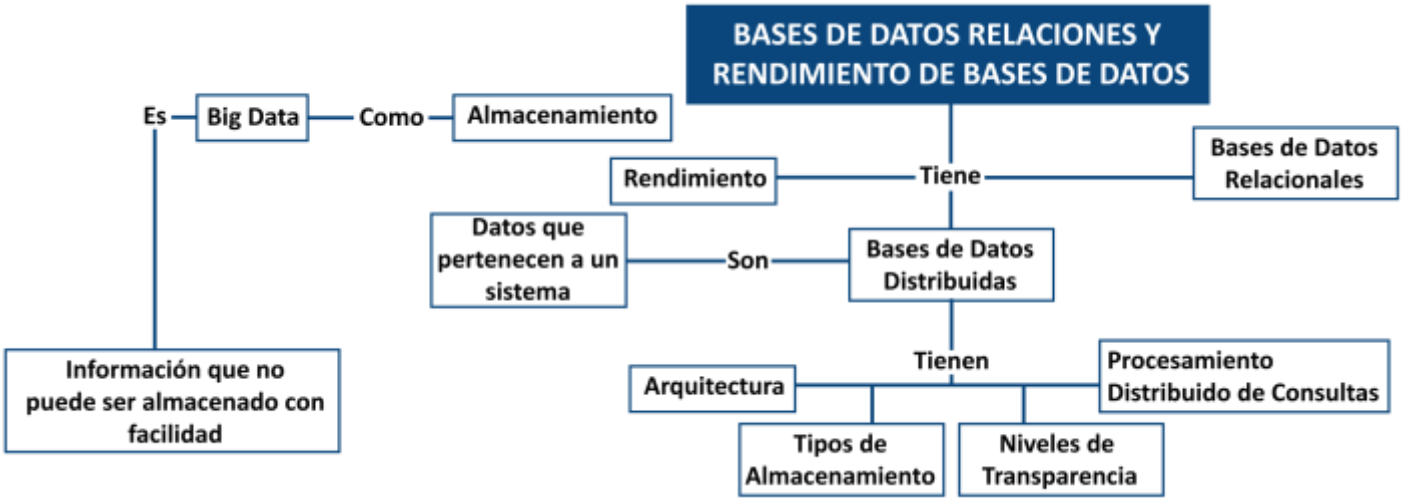
Middleware es una incorporación relativamente reciente en la computación. Obtuvo popularidad en los 80 como una solución al problema de cómo conectar nuevas aplicaciones con viejos sistemas. De todas maneras el término ha sido usado desde 1968. También facilitaba el procesamiento distribuido: conexión de múltiples aplicaciones para crear una aplicación más grande, generalmente sobre una red.

Referenciado de: http://www.alegsa.com.ar/Dic/middleware.php#sthash.vkeZJukL.dpuf

2.2.3 TALLER DE ENTRENAMIENTO UNIDAD 1.

Según el planteamiento de esta unidad, y desde su punto de vista como aspirante al título de Ingeniero de Sistemas, cree un ensayo en el cual plasme su punto de vista prospectivo sobre el futuro de la información y su manejo haciendo uso de las bases de datos que se mencionan anteriormente.

3 UNIDA 2 BASES DE DATOS RELACIONES Y RENDIMIENTO DE BASES DE DATOS



Fuente: Propia

Conceptos Básicos:

1. **Bases de datos Distribuidas**: Son un grupo de datos que pertenecen a un sistema pero a su vez está repartido entre ordenadores de una misma red.

2. **Bases de Datos Relacional**: tipo de base de datos (BD) que cumple con el modelo relacional (el modelo más utilizado actualmente para implementar las BD ya planificadas).

3. **Big Data:** información o grupo de datos que por su elevado volumen, diversidad y complejidad no pueden ser almacenados ni visualizados con herramientas tradicionales.

3.1 TEMA 1 BASES DE DATOS RELACIONALES

Una base de datos relacional es una colección de elementos de datos organizados en un conjunto de tablas formalmente descritas desde la que se puede acceder a los datos o volver a montarlos de muchas maneras diferentes sin tener que reorganizar las tablas de la base. La base de datos relacional fue inventada por E.F. Codd en IBM en 1970.

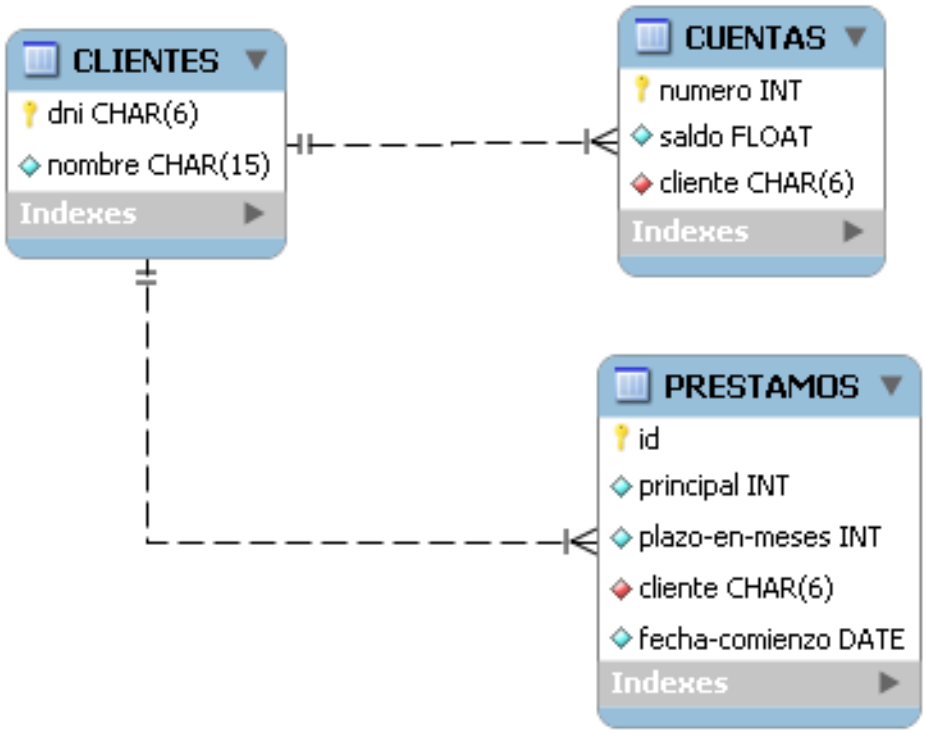
La interfaz estándar de programa de usuario y aplicación a una base de datos relacional es el lenguaje de consultas estructuradas (SQL). Los comandos de SQL se utilizan tanto para consultas interactivas para obtener base de datos relacional y para la recopilación de datos para los informes.

Además de ser relativamente fáciles de crear y acceder, una base de datos relacional tiene la importante ventaja de ser fácil de extender. Después de la creación original de una base de datos, una nueva categoría de datos se puede añadir sin necesidad de que todas las aplicaciones existentes sean modificadas.

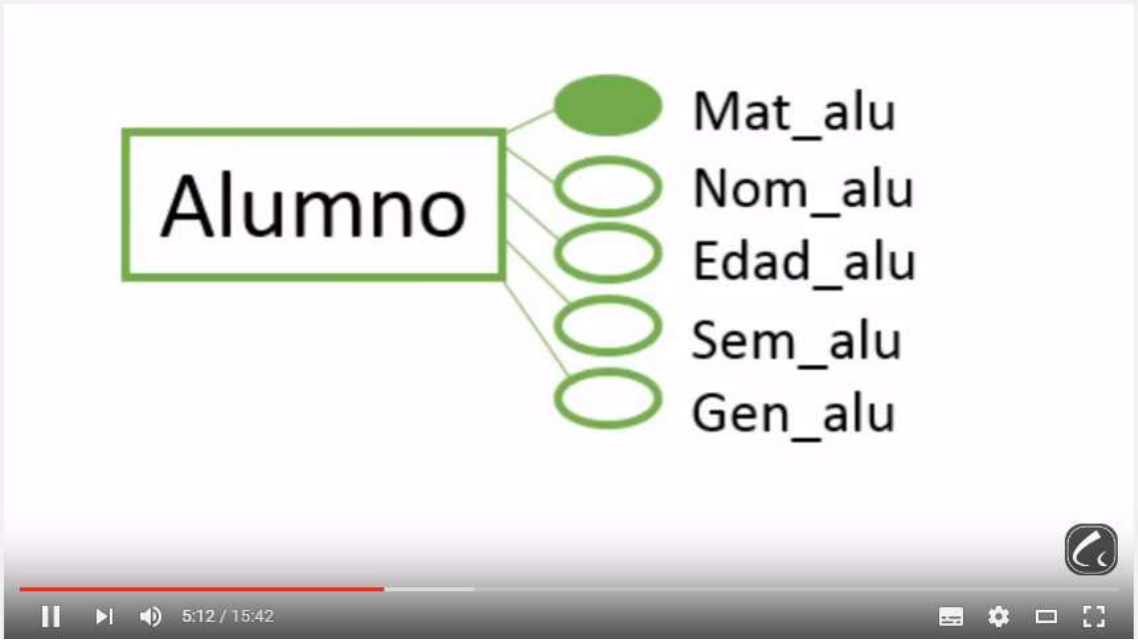
Una base de datos relacional es un conjunto de tablas que contienen datos provistos en categorías predefinidas. Cada tabla (que a veces se llaman 'relación') contiene una o más categorías de datos en columnas. Cada fila contiene una instancia única de datos para las categorías definidas por las columnas. Por ejemplo, una base de datos típica de ingreso de solicitudes de negocio incluiría una información de una tabla que describiera a un cliente con columnas para el nombre, dirección, número de teléfono, y así sucesivamente. Otra tabla identificaría el pedido: producto, cliente, fecha, precio de venta, y así sucesivamente. Un usuario de la base de datos podría obtener una vista de la base de datos que se ajuste a sus necesidades. Por ejemplo, un gerente de sucursal podría preferir una vista o informe sobre todos los clientes que han comprado productos después de una fecha determinada. Un gerente de servicios financieros en la misma empresa podría, desde las mismas tablas, obtener un informe sobre las cuentas que deben ser pagadas.

Al crear una base de datos relacional, se puede definir el dominio de posibles valores de una columna de datos y restricciones adicionales que pueden aplicarse a ese valor de dato. Por ejemplo, un dominio de posibles clientes podría permitir un máximo de diez posibles nombres de clientes pero estar compilado en una tabla que permita que sólo tres de estos nombres de clientes puedan ser especificados.

La definición de una base de datos relacional resulta en una tabla de metadatos o descripciones formales de las tablas, columnas, dominios y restricciones.



Fuente: http://carlianangonoaccess.blogspot.com.co/p/las-bases-de-datos-relacionales.html



Base de Datos #3| Ejercicio Diagrama Entidad Relación: Enlace

Enlace: http://carlianangonoaccess.blogspot.com.co/p/las-bases-de-datos-relacionales.html

3.2 TEMA 2 ALMACENAMIENTO DE BASES DE DATOS DISTRIBUIDAS

3.2.1.1 BASES DE DATOS DISTRIBUIDAS

Las bases de datos distribuidas son un grupo de datos que pertenecen a un sistema, pero a su vez está repartido entre ordenadores de una misma red, ya sea a nivel local o cada uno en una diferente localización geográfica, cada sitio en la red es autónomo en sus capacidades de procesamiento y es capaz de realizar operaciones locales y en cada uno de estos ordenadores debe estar ejecutándose una aplicación a nivel global que permita la consulta de todos los datos como si se tratase de uno solo.

CENTRALIZADO	DISTRIBUIDO
Control centralizado: un solo DBA	Control jerárquico: DBA global y DBA local

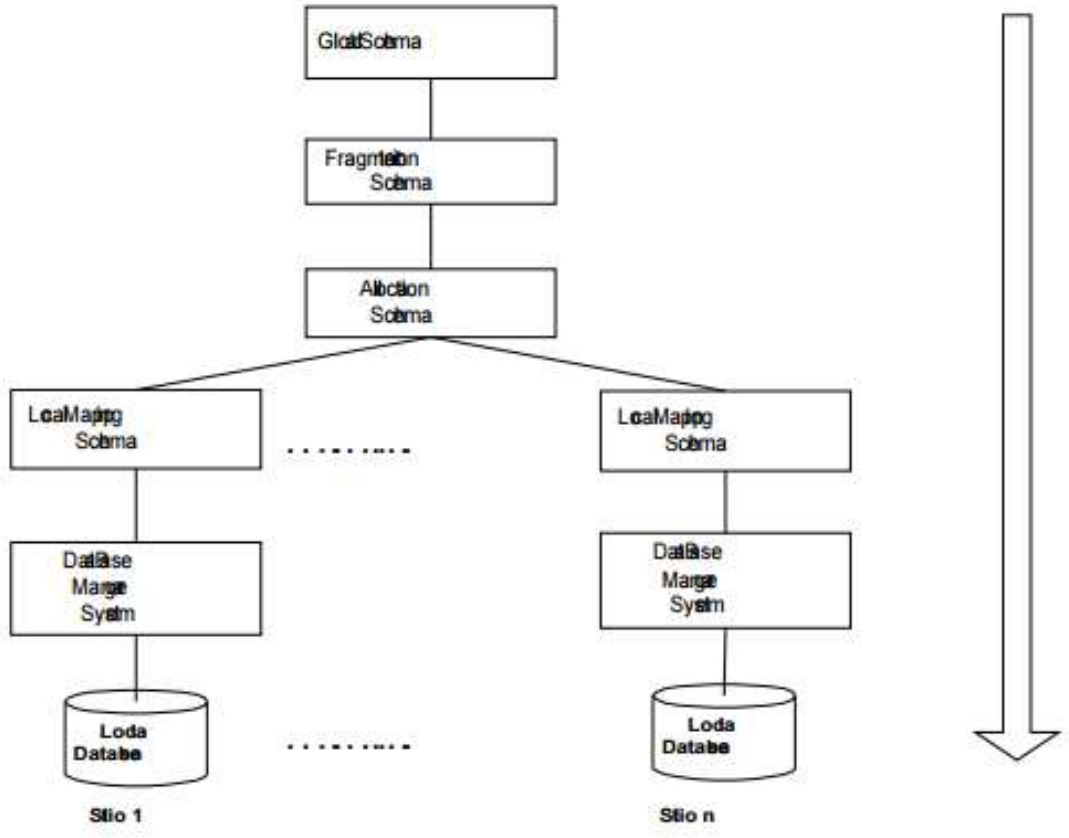
Independencia de Datos: Organización de los datos es transparente para el programador	Transparencia en la Distribución: Localización de los datos es un aspecto adicional de independencia de datos
Reducción de redundancia: Una sola copia de datos que se comparta	Replicación de Datos: Copias múltiples de datos que incrementa la localidad y la disponibilidad de datos
Estructuras físicas complejas para accesos eficientes	No hay estructuras intersitios. Uso de optimización global para reducir transferencia de datos.
Seguridad	Problema de seguridad intrínsecos

Fuente: Carolina Henao, 2011 – Modulo Bases de Datos 2 - Uniremington

Para tener una base de datos distribuida debe cumplirse las condiciones de una Red Computacional. Una red de comunicación provee las capacidades para que un proceso ejecutándose en un sitio de la red envíe y reciba mensajes de otro proceso ejecutándose en un sitio distinto. Parámetros a considerar incluyen: Retraso en la entrega de mensajes, Costo de transmisión de un mensaje y Confiabilidad de la red. Diferentes tipos de redes: POINT-TO-POINT, BROADCAST, LAN, WAN.

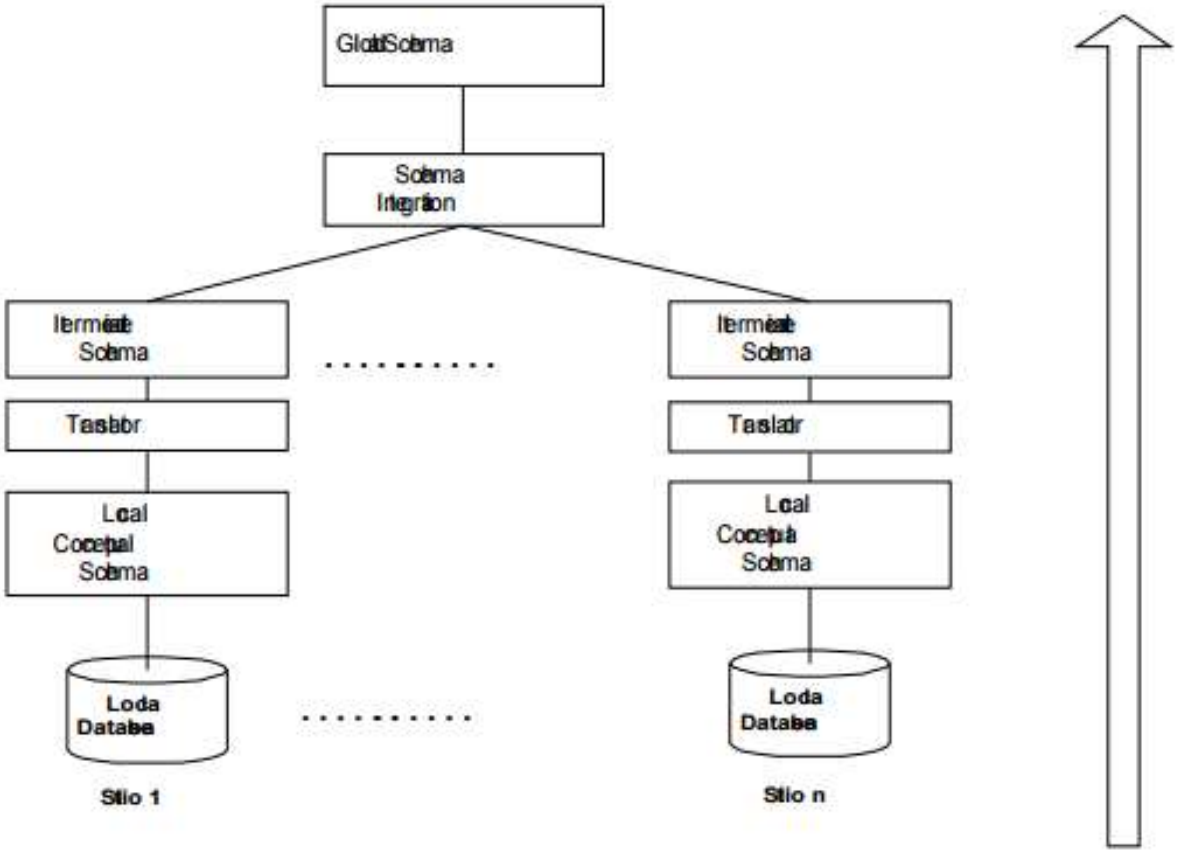
Arquitectura de las Bases de datos distribuidas:

Integración lógica por medio de diseño top-down (DistDB)



Fuente: Carolina Henao, 2011 – Modulo Bases de Datos 2 - Uniremington

Integración lógica por medio de bottom-up (Multidatabase)



Fuente: Carolina Henao, 2011 – Modulo Bases de Datos 2 - Uniremington

Global Schema: Define todos los datos que están incluidos en la bd distribuida tal como si la bd no fuera distribuida. Consiste de una definición de relaciones globales.

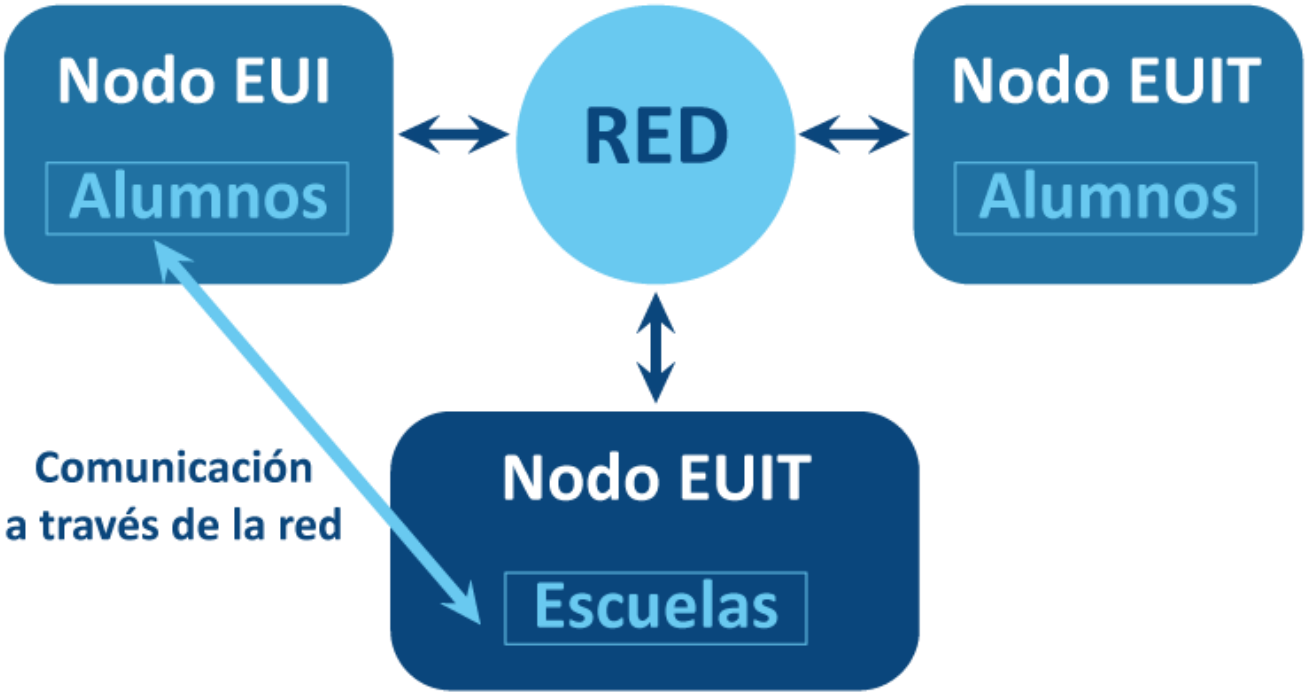
Fragmentation Schema: Traducción entre relaciones globales y fragmentos. (Una relación global puede consistir de varios fragmentos pero un fragmento está asociado con sólo una relación global).

Allocation Schema: Define el sitio (o sitios) en el cual un fragmento está localizado.

Local Maping Schema: Traduce los fragmentos locales a los objetos que son manejados por el SMBD local

Separación entre fragmentación y localización: Transparencia de Fragmentación, Transparencia de Localización, Control explícito de redundancia y Independencia de BD locales.

Ejemplo de Bases de datos Distribuidas:



Fuente: Carolina Henao, 2011 – Modulo Bases de Datos 2 - Uniremigton

Nodos de las Escuelas:

DNI	Escuela	Nombre	Nota ingreso	Beca

Nodos de las Escuelas:

Escuela	Situación	Número Alumnos

Nuevo alumno en la secretaría del centro: Transacción Local

Nuevo alumno en el rectorado: Transacción Global

Fuente: Carolina Henao, 2011 – Modulo Bases de Datos 2 - Uniremigton

TIPOS DE ALMACENAMIENTO:

Replica: El sistema conserva varias copias o réplicas idénticas de una tabla. Cada réplica se almacena en un nodo diferente.

Ventajas:
Disponibilidad: El sistema sigue funcionando aún en caso de caída de uno de los nodos.
Aumento del paralelismo: Varios nodos pueden realizar consultas en paralelo sobre la misma tabla. **Cuantas más réplicas existan de la tabla, mayor será la posibilidad de que el dato buscado se encuentre en el nodo desde el que se realiza la consulta,** minimizando con ello el tráfico de datos entre nodos.

Inconveniente:
Aumento de la sobrecarga en las actualizaciones: El sistema debe asegurar que todas las réplicas de la tabla sean consistentes. Cuando se realiza una actualización sobre una de las réplicas, los cambios deben propagarse a todas las réplicas de dicha tabla a lo largo del sistema distribuido.

Fragmentación: Existen tres tipos de fragmentación la horizontal, la vertical y la mixta.

Fragmentación Horizontal: Una tabla T se divide en subconjuntos, T1, T2,...Tn. Los fragmentos se definen a través de una operación de selección y su reconstrucción se realizará con una operación de unión de los fragmentos componentes. Cada fragmento se sitúa en un nodo. **Pueden existir fragmentos no disjuntos: combinación de fragmentación y replicación.**

3.2.2 EJERCICIO DE APRENDIZAJE

Tabla inicial de alumnos

DNI	ESCUELA	NOMBRE		NOTA INGRESO	BECA
87633483	EUI	Concha Queta		5.6	No
99855743	EUI	Josechu Letón		7.2	Si
33887293	EUIT	Oscar Romato		6.1	Si
05399075	EUI	Bill Gates		5.0	No
44343234	EUIT	Pepe Pótamo		8.0	No
44543324	EUI	Maite Clado		7.5	Si
66553234	EUIT	Ernesto Mate		6.6	No

Fuente: https://iessanvicente.com/colaboraciones/BBDDdistribuidas.pdf

Tabla de alumnos fragmentada

Fragmento de la EUIT: Escuela="EUI" (T)

DNI	ESCUELA	NOMBRE	NOTA INGRESO	BECA
87633483	EUI	Concha Queta	5.6	No
99855743	EUI	Josechu Letón	7.2	Si
05399075	EUI	Bill Gates	5.0	No
44543324	EUI	Maite Clado	7.5	Si

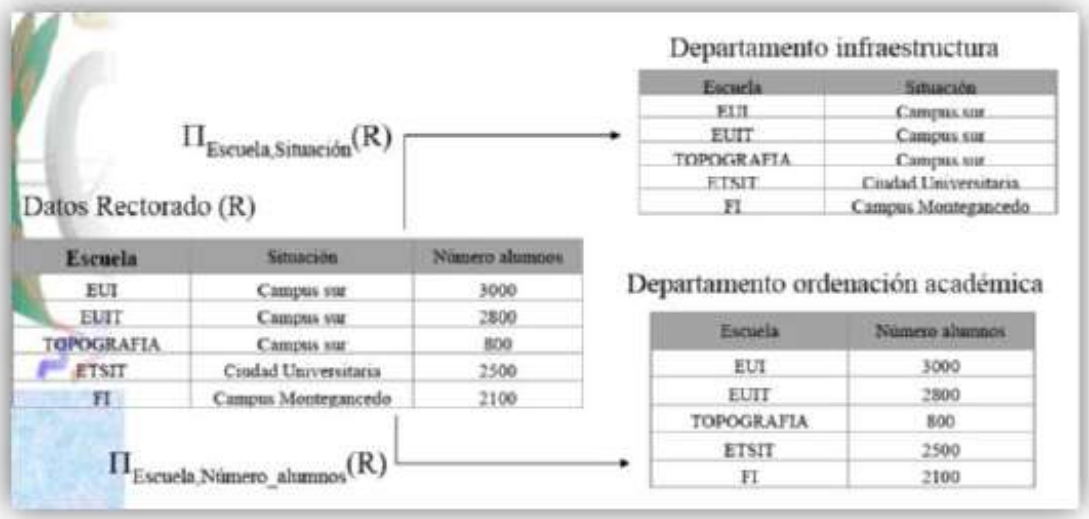
Fragmento de la EUIT: Escuela="EUI" (T)

DNI	ESCUELA	NOMBRE	NOTA INGRESO	BECA
33887293	EUI	Oscar Romato	6.1	Si
44343234	EUI	Pepe Pótamo	8.0	No
66553234	EUI	Ernesto Mate	6.6	No

Fuente: https://iessanvicente.com/colaboraciones/BBDDdistribuidas.pdf

Fragmentación Vertical: Una tabla T se divide en subconjuntos, T1, T2, ...Tn. Los fragmentos se definen a través de una operación de proyección. Cada fragmento debe incluir la clave primaria de la tabla. Su reconstrucción se realizará con una operación de join de los fragmentos componentes, pueden existir fragmentos no disjuntos: combinación de fragmentación y replicación.

Ejemplo:



Fuente: https://iessanvicente.com/colaboraciones/BBDDdistribuidas.pdf

Fragmentación Mixta: Como el mismo nombre indica es una combinación de las dos anteriores vistas he aquí un ejemplo a partir de una tabla fragmentada horizontalmente.

DNI	Escuela	Nombre	Beca
87633483	EUI	Concha Queta	No
99855743	EUI	Josechu Letón	Si
05399075	EUI	Bill Gates	No
44543324	EUI	Maite Clado	Si

$$\Pi_{DNI,Escuela,Escuela,Nombre,Beca}(E)$$

Fragmento de la EUI: $\sigma_{Escuela="EUI"}(T)$

DNI	Escuela	Nombre	Nota ingreso	Beca
87633483	EUI	Concha Queta	5.6	No
99855743	EUI	Josechu Letón	7.2	Si
05399075	EUI	Bill Gates	5.0	No
44543324	EUI	Maite Clado	7.5	Si

$$\Pi_{DNI,Escuela,Nombre,Nota\ ingreso}(E)$$

DNI	Escuela	Nombre	Nota ingreso
87633483	EUI	Concha Queta	5.6
99855743	EUI	Josechu Letón	7.2
05399075	EUI	Bill Gates	5.0
44543324	EUI	Maite Clado	7.5

Fuente: https://iessanvicente.com/colaboraciones/BBDDdistribuidas.pdf

Replica y Fragmentación: Las técnicas de réplica y fragmentación se pueden aplicar sucesivamente a la misma relación de partida. Un fragmento se puede replicar y a su vez esa réplica ser fragmentada, para luego replicar alguno de esos fragmentos.

Niveles de Transparencia en una Base de Datos Distribuida: El propósito de establecer una arquitectura de un sistema de bases de datos distribuidas es ofrecer un nivel de transparencia adecuado para el manejo de la información. La transparencia se define como la separación de la semántica de alto nivel de un sistema de los aspectos de bajo nivel relacionados a la implementación del mismo. Un nivel de transparencia adecuado permite ocultar los detalles de implementación a las capas de alto nivel de un sistema y a otros usuarios.
El sistema de bases de datos distribuido permite proporcionar independencia de los datos.

La independencia de datos se puede dar en dos aspectos: lógica y física.

Independencia lógica de datos. Se refiere a la inmunidad de las aplicaciones de usuario a los cambios en la estructura lógica de la base de datos. Esto permite que un cambio en la definición de un esquema no debe afectar a las aplicaciones de usuario. Por ejemplo, el agregar un nuevo atributo a una relación, la creación de una nueva relación, el reordenamiento lógico de algunos atributos.

Independencia física de datos. Se refiere al ocultamiento de los detalles sobre las estructuras de almacenamiento a las aplicaciones de usuario. La descripción física de datos puede cambiar sin afectar a las aplicaciones de usuario. Por ejemplo, los datos pueden ser movidos de un disco a otro, o la organización de los datos puede cambiar.

La transparencia al nivel de red se refiere a que los datos en un SBDD se accedan sobre una red de computadoras, sin embargo, las aplicaciones no deben notar su existencia. La transparencia al nivel de red conlleva a dos cosas:

Transparencia sobre la localización de datos. el comando que se usa es independiente de la ubicación de los datos en la red y del lugar en donde la operación se lleve a cabo. Por ejemplo, en Unix existen dos comandos para hacer una copia de archivo. Cp se utiliza para copias locales y rcp se utiliza para copias remotas. En este caso no existe transparencia sobre la localización.

Transparencia sobre el esquema de nombramiento. Lo anterior se logra proporcionando un nombre único a cada objeto en el sistema distribuido. Así, no se debe mezclar la información de la localización con en el nombre de un objeto.

La transparencia sobre replicación de datos se refiere a que si existen réplicas de objetos de la base de datos, su existencia debe ser controlada por el sistema no por el usuario. Se debe tener en cuenta que cuando el usuario se encarga de manejar las réplicas en un sistema, el trabajo de éste es mínimo por lo que se puede obtener una eficiencia mayor. Sin embargo, el usuario puede olvidarse de mantener la consistencia de las réplicas teniendo así datos diferentes. La transparencia a nivel de fragmentación de datos permite que cuando los objetos de la bases de datos están fragmentados, el sistema tiene que manejar la conversión de consultas de usuario definidas sobre relaciones globales a consultas definidas sobre fragmentos. Así también, será necesario mezclar las respuestas a consultas fragmentadas para obtener una sola respuesta a una consulta global. El acceso a una base de datos distribuida debe hacerse en forma transparente. En resumen, la transparencia tiene como punto central la independencia de datos.
La responsabilidad sobre el manejo de transparencia debe estar compartida tanto por el sistema operativo, el sistema de manejo de bases de datos y el lenguaje de acceso a la base de datos distribuida. Entre estos tres módulos se deben resolver los aspectos sobre el procesamiento distribuido de consultas y sobre el manejo de nombres de objetos distribuidos.

Procesamiento Distribuido de Consultas: El procesamiento de consultas es de suma importancia en bases de datos centralizadas. Sin embargo, en BDD éste adquiere una relevancia mayor. El objetivo es convertir transacciones de usuario en instrucciones para manipulación de datos. No obstante, el orden en que se realizan las transacciones afecta grandemente la velocidad de respuesta del sistema. Así, el procesamiento de consultas presenta un problema de optimización en el cual se determina el orden en el cual se hace la menor cantidad de operaciones. En BDD se tiene que considerar el procesamiento local de una consulta junto con el costo de transmisión de información al lugar en donde se solicitó la consulta.

Recuperación: En los entornos distribuidos de datos podemos encontrar lo siguientes:

Fallo de los nodos: Cuando un nodo falla, el sistema deberá continuar trabajando con los nodos que aún funcionan. Si el nodo a recuperar es una base de datos local, se deberán separar los datos entre los nodos restantes antes de volver a unir de nuevo el sistema

Copias múltiples de fragmentos de datos: El subsistema encargado del control de concurrencia es el responsable de mantener la consistencia en todas las copias que se realicen y el subsistema que realiza la recuperación es el responsable de hacer copias consistentes de los datos de los nodos que han fallado y que después se recuperarán.

Transacción distribuida correcta: Se pueden producir fallos durante la ejecución de una transacción correcta si se plantea el caso de que al acceder a alguno de los nodos que intervienen en la transacción, dicho nodo falla.

Fallo de las conexiones de comunicaciones: El sistema debe ser capaz de tratar los posibles fallos que se produzcan en las comunicaciones entre nodos. El caso más extremo es el que se produce cuando se divide la red. Esto puede producir la separación de dos o más particiones donde las particiones de cada nodo pueden comunicarse entre sí pero no con particiones de otros nodos.

Para implementar las soluciones a estos problemas, supondremos que los datos se encuentran almacenados en un único nodo sin repetición. De ésta manera sólo existirá un único catálogo y un único DM (Data Manager) encargados del control y acceso a las distintas partes de los datos. Para mantener la consistencia de los datos en el entorno distribuido contaremos con los siguientes elementos:

Catálogo: Programa o conjunto de programas encargados de controlar la ejecución concurrente de las transacciones.

CM (Cache Manager): Subsistema que se encarga de mover los datos entre las memorias volátiles y no volátiles, en respuesta a las peticiones de los niveles más altos del sistema de bases de datos. Sus operaciones son Fetch(x) y Flush(x).

RM (Recovery Manager): Subsistema que asegura que la base de datos contenga los efectos de la ejecución de transacciones correctas y ninguno de incorrectas. Sus operaciones son Start, Commit, Abort, Read, Write, que utilizan a su vez los servicios del CM.

DM (Data Manager): Unifica las llamadas a los servicios del CM y el RM.

TM (Transaction Manager): Subsistema encargado de determinar que nodo deberá realizar cada operación a lo largo de una transacción.

Las operaciones de transacción que soporta una base de datos son: Start, Commit y Abort. Para comenzar una nueva transacción se utiliza la operación Start. Si aparece una operación commit, el sistema de gestión da por terminada la transacción con normalidad y sus efectos permanecen en la base de datos. Si, por el contrario, aparece una operación abort, el sistema de gestión asume que la transacción no termina de forma normal y todas las modificaciones.

Ventajas y Desventajas:

Ventajas:

Los sistemas de bases de datos distribuidos tienen múltiples ventajas. En primer lugar los datos son localizados en lugar más cercano, por tanto, el acceso es más rápido, el procesamiento es rápido debido a que varios nodos intervienen en el procesamiento de una carga de trabajo, nuevos nodos se pueden agregar fácil y rápidamente. La comunicación entre nodos se mejora, los costos de operación se reducen, son amigables al usuario, la probabilidad de que una falla en un solo nodo afecte al sistema es baja y existe una autonomía e independencia entre los nodos.

Las razones por las que compañías y negocios migran hacia bases de datos distribuidas incluyen razones organizacionales y económicas, para obtener una interconexión confiable y flexible con las bases de datos existentes, y por un crecimiento futuro. El enfoque distribuido de las bases de datos se adapta más naturalmente a la estructura de las organizaciones. Además, la necesidad de desarrollar una aplicación global (que incluya a toda la organización), se resuelva fácilmente con bases de datos distribuidas. Si una organización crece por medio de la creación de unidades o departamentos nuevos, entonces, el enfoque de bases de datos distribuidas permite un crecimiento suave.

Los datos se pueden colocar físicamente en el lugar donde se accedan más frecuentemente, haciendo que los usuarios tengan control local de los datos con los que interactúan. Esto resulta en una autonomía local de datos permitiendo a los usuarios aplicar políticas locales respecto del tipo de accesos a sus datos.

Mediante la replicación de información, las bases de datos distribuidas pueden presentar cierto grado de tolerancia a fallos haciendo que el funcionamiento del sistema no dependa de un solo lugar como en el caso de las bases de datos centralizadas.

La independencia de datos se puede dar en dos aspectos: lógica y física.

Desventajas:

Las razones por las que compañías y negocios migran hacia bases de datos distribuidas incluyen razones organizacionales y económicas, para obtener una interconexión confiable y flexible con las bases de datos existentes, y por un crecimiento futuro. El enfoque distribuido de las bases de datos se adapta más naturalmente a la estructura de las organizaciones. Además, la necesidad de desarrollar una aplicación global (que incluya a toda la organización), se resuelva fácilmente con bases de datos distribuidas. Si una organización crece por medio de la creación de unidades o departamentos nuevos, entonces, el enfoque de bases de datos distribuidas permite un crecimiento suave.

Los datos se pueden colocar físicamente en el lugar donde se accedan más frecuentemente, haciendo que los usuarios tengan control local de los datos con los que interactúan. Esto resulta en una autonomía local de datos permitiendo a los usuarios aplicar políticas locales respecto del tipo de accesos a sus datos.

Mediante la replicación de información, las bases de datos distribuidas pueden presentar cierto grado de tolerancia a fallos haciendo que el funcionamiento del sistema no dependa de un solo lugar como en el caso de las bases de datos centralizadas.

La independencia de datos se puede dar en dos aspectos: lógica y física.

3.2.3 TALLER DE ENTRENAMIENTO

BASES DE BASES DE DATOS DISTRIBUIDAS

El siguiente taller de entrenamiento se propone para validar la aprehensión de los conceptos. Se debe leer detenidamente las preguntas propuestas y dar respuesta a ellas.

1. ¿Qué son Bases de Datos Distribuidas?
2. Explique los tipos de almacenamiento.
3. ¿Cuáles son los niveles de transparencia en una Base de Datos Distribuida?
4. Explique los aspectos lógicos y físicos de independencia de datos.
5. ¿En qué consiste el Procesamiento Distribuido de Consultas?

3.2.3.1 BASES DE DATOS FEDERADAS

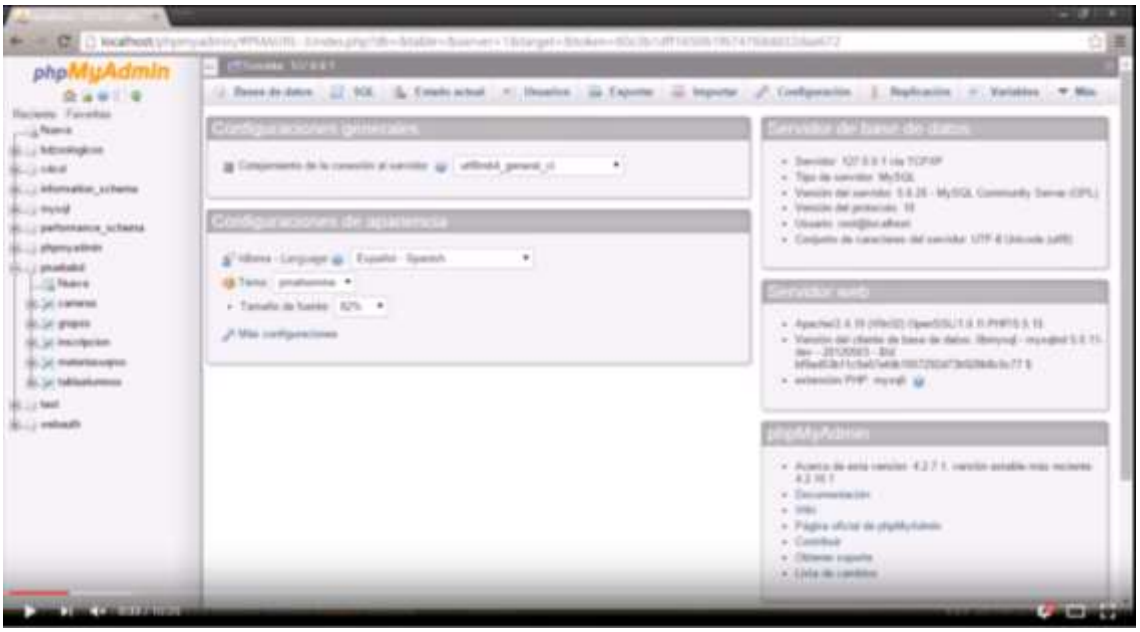
La definición de **Base de Datos Federada** ha sido realizada por varios autores. Así, Sheth A.P. and Larson, J.*A.* definieron el concepto base de datos federada como "una colección de sistemas de bases de datos independientes, cooperativos, posiblemente heterogéneos, que son autónomos y que permiten compartir todos o algunos de sus datos". Además, Larson amplió esta definición diciendo que "en un sistema federado los usuarios tienen acceso a los datos, de los distintos sistemas, a través de una interfaz común; sin embargo, no existe un esquema global que describa a todos los datos de las distintas bases de datos, sino que en su lugar hay varios esquemas unificados, cada uno describiendo porciones de bases de datos y archivos para el uso de cierta clase de usuarios".

La primera definición pone de manifiesto todas las propiedades que definen a una Base de Datos Federada: heterogeneidad, autonomía y distribución. Una Base de Datos Federada se dice que es **heterogénea** debido a que los sistemas de bases de datos que lo forman pueden tener cualquier arquitectura. En cuanto a la **autonomía**, esta propiedad se cumple ya que cada sistema de bases de datos funciona por sí mismo y de forma local. Por último, el concepto de **distribución** hace referencia a que cada sistema de bases de datos puede estar localizado en cualquier punto.

En cuanto a la segunda definición, se explica que un sistema federado está compuesto por los datos de las diferentes bases de datos que forman el esquema, pero que dichos datos no están presentes en ningún esquema global. Además, no necesariamente todos los datos de una base de datos son compartidos a los usuarios, sino que se tiene la posibilidad de compartir sólo una porción de los datos.

Para dar claridad en el tema se invita revisar los siguientes Link:

Base de Datos Federada Con MySQL: Enlace



Creación de tablas federadas con MySQL: Enlace

Enlace: https://modelosbd2012t1.wordpress.com/2012/03/15/bases-de-datos-federadas/

Características de los SGBDF:

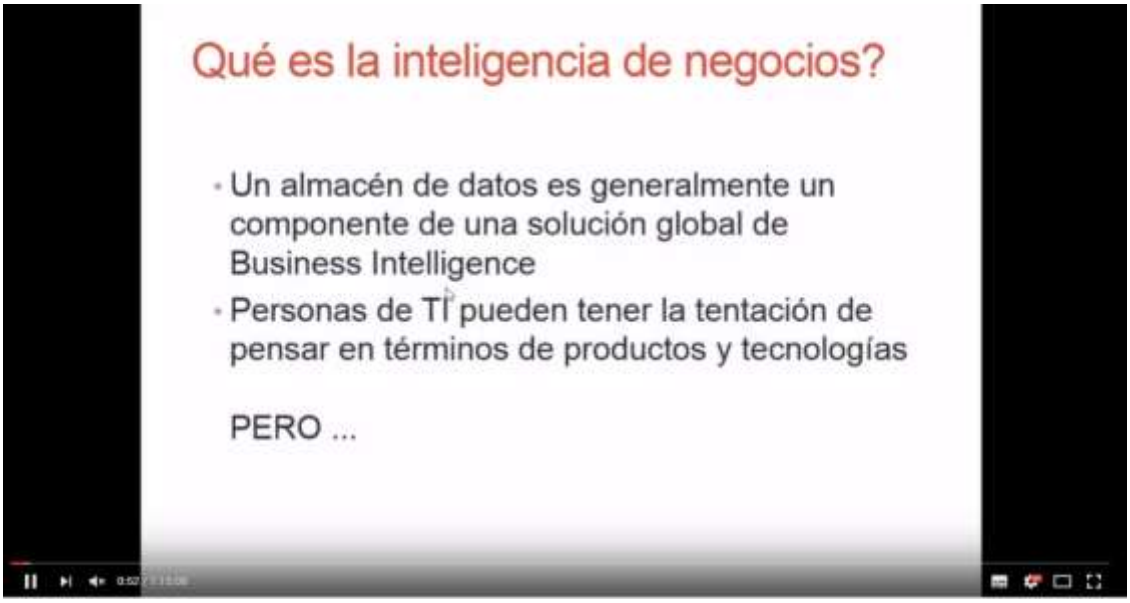
Un Sistema Gestor de Bases de Datos Federadas (SGBDF) es el responsable de proveer una vista de datos transparente al usuario. Esto significa que el usuario percibe a la base de datos federada como una única base

de datos local, y no múltiples bases de datos que contienen diferentes datos, que es como en realidad está configurado.

Un SGBDF no contiene ningún dato, sino que accede a los datos almacenados en las diferentes bases de datos que forman el esquema. Para acceder a dichos datos es necesario establecer las denominadas **federaciones**. Una federación es una vista que se establece en una base de datos en particular para exteriorizar los datos que se desean mostrar.

Para dar claridad en el tema se invita revisar los siguientes Link:

EQUIPO 5 BASES DE DATOS FEDERADAS: Enlace



Introducción a Data Warehouse: Enlace

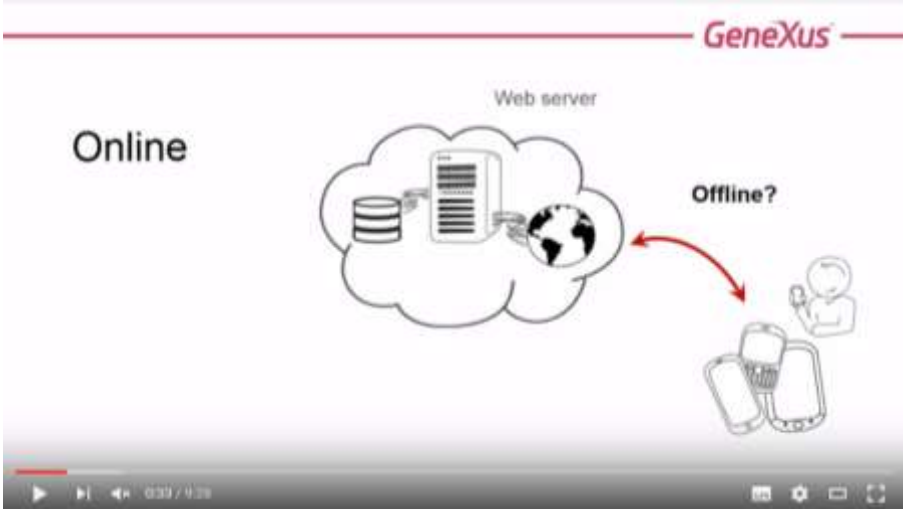
3.2.3.2 BASES DE DATOS MÓVILES

En los últimos años se han producido grandes avances en las tecnologías de comunicación inalámbricas. Estos avances, junto al uso cada vez más extendido de los dispositivos móviles, han causado la aparición de una nueva disciplina: la computación móvil. Gracias a la computación móvil, los usuarios pueden acceder a una base de datos remota en cualquier momento y en cualquier lugar. Los empleados de una empresa pueden trabajar desde su casa, desde las instalaciones del cliente o mientras están de viaje, de la misma forma que si estuvieran en la oficina. La computación móvil introduce el concepto de base de datos móvil. Una base de datos móvil es una base de datos portable, físicamente independiente del servidor corporativo de base de datos y capaz de comunicarse con ese servidor desde sitios remotos para compartir datos corporativos. Utilizando bases de datos móviles, los trabajadores pueden acceder a los datos corporativos desde cualquier dispositivo que disponga de conexión a Internet.

Arquitectura: La arquitectura general de una plataforma móvil es un modelo distribuido formado por computadores fijos, estaciones base y unidades móviles. Los computadores fijos son computadores de uso general que no disponen de medios para comunicarse con las unidades móviles. Las estaciones base disponen de enlaces inalámbricos para conectar con las unidades móviles; son máquinas que actúan de intermediarios entre las unidades móviles y los computadores fijos. Los computadores fijos y las estaciones base están interconectados por medio de una red fija (cableada) de alta velocidad. Las unidades móviles se conectan a las estaciones base mediante enlaces inalámbricos; los enlaces más comunes son el estándar 802.11 (Wi-Fi), el servicio GPRS y la tecnología Bluetooth.

Para dar más claridad al tema revisar los siguientes link:

base de datos móviles: Enlace



Arquitectura de las aplicaciones móviles online: Enlace

Las unidades móviles se pueden mover libremente por un espacio conocido como **dominio de movilidad geográfica**, cuyo alcance está determinado por la cobertura de los enlaces inalámbricos. Este dominio se divide en dominios más pequeños llamados **celdas**. Cada celda es controlada por una estación base. El movimiento de las unidades móviles dentro del dominio de movilidad geográfica no debe estar restringido, es decir, se debe garantizar el acceso a la información, aunque las unidades móviles se muevan entre las celdas.

Sistemas Gestores de Bases de Datos móviles: Muchos fabricantes ofrecen SGBD móviles capaces de comunicarse con los principales SGBD relacionales. Estos SGBD móviles están adaptados a los recursos limitados de las unidades móviles y proporcionan una serie de funcionalidades adicionales:

1. Comunicación con el servidor centralizado de base de datos mediante técnicas de comunicación inalámbrica.
2. Replicación de datos en el servidor centralizado de base de datos y en el dispositivo móvil.
3. Sincronización de datos entre el servidor centralizado de base de datos y el dispositivo móvil.
4. Gestión de datos en el dispositivo móvil.
5. Análisis de los datos almacenados en el dispositivo móvil.

Ejemplos de bases de datos móviles: iAnywhere Solutions, empresa filial de Sybase, lidera el ranking del mercado de bases de datos móviles gracias a SQL Anywhere. Este paquete proporciona bases de datos que pueden utilizarse tanto a nivel de servidor (soporta máquinas de hasta 64bits) como a nivel de dispositivo móvil. SQL Anywhere se compone de las siguientes tecnologías:

1. SQL Anywhere Server: sistema gestor de bases de datos relacionales para los sistemas de bases de datos móviles.
2. Ultralite: sistema gestor de bases de datos que puede embeberse en dispositivos móviles.
3. Mobilink: tecnología de sincronización para el intercambio de datos entre bases de datos relacionales y bases de datos no relacionales.
4. QAnywhere: facilita el desarrollo de aplicaciones móviles robustas y seguras.

5. SQL Remote: permite a los usuarios de dispositivos móviles sincronizar sus datos con otras bases de datos SQL Anywhere.

DB2 Everyplace de IBM es una base de datos relacional y un servidor de sincronización que permite extender las aplicaciones y los datos empresariales a dispositivos móviles. Gracias a un consumo de recursos reducido, esta base de datos puede integrarse en dispositivos como PDAs y teléfonos móviles.

Microsoft también ofrece una base de datos para dispositivos móviles. Se trata de Microsoft SQL Server Compact 3.5, un motor de bases de datos que permite desarrollar aplicaciones en cualquier plataforma Windows incluyendo Tablet PCs, Pocket PCs, Smart Phones y equipos de escritorio.

Oracle Database Lite 10g es la solución de Oracle para desarrollar aplicaciones en entornos móviles. Proporciona un cliente que permite la realización de consultas SQL para acceder a los datos locales del dispositivo y un servidor para gestionar los datos de forma centralizada.

Otros productos menos utilizados son **Borland's JDataStore**, una base de datos Java para dispositivos móviles y aplicaciones Web, o **MobiSnap**, un proyecto de investigación cuyo objetivo es soportar el desarrollo de aplicaciones con bases de datos relacionales en entornos móviles.

SIMILITUDES BASES DE DATOS FEDERADAS VS BASES DE DATOS MÓVILES	DIFERENCIAS BASES DE DATOS FEDERADAS VS BASES DE DATOS MÓVILES
Las dos bases de datos son modelos distribuidos de bases de datos a los cuáles se accede de forma remota, ya sea desde un equipo fijo o un equipo móvil.	Las bases de datos móviles son recomendadas en los casos en los que los usuarios deben estar moviéndose de un lugar para otro para realizar las funciones y cuando la información que deben tratar se puede mostrar y tratar en un dispositivo móvil.
Los dos sistemas tienen mecanismos de privilegios de usuarios. Dependiendo de los privilegios que tenga el usuario que accede al sistema, dicho usuario podrá acceder a una parte del esquema u a otra.	Los sistemas de bases de datos federados permiten dar acceso a una gran cantidad de datos que los demás sistemas no podrían permitir ni soportar.
En estos sistemas el usuario no es consciente de la disposición geográfica en la que se encuentra el servidor o servidores de bases de datos. Esta característica también da al usuario un gran nivel de movilidad y de acceso a los datos desde cualquier punto.	En el caso de las bases de datos móviles, el sistema se puede reducir sólo al dispositivo móvil que lo ejecuta, ya que éste es el único que contiene la base de datos y accede a ellos. Este tipo de sistemas son sistemas gestores de bases de datos embebidos en el mismo dispositivo.

Son sistemas complejos que necesitan una gran infraestructura que dé soporte a este tipo de bases de datos	Las bases de datos federadas son un conjunto de esquemas unificados, a diferencia de las bases de datos móviles, que sólo disponen de un esquema global.
Necesitan tener definidos mecanismos de concurrencia de los datos para que no se vea dañada la integridad de los mismos en el caso de que varios usuarios accedan a los mismos datos a la vez.	Los sistemas de bases de datos móviles permiten a los usuarios trabajar de forma desconectada con los datos, y una vez que éstos han sido modificados, los usuarios sincronizan dichos datos con el sistema. Esto evita que los dispositivos móviles tengan que estar siempre conectados al sistema para interactuar y acceder a la base de datos.
Al tener la parte de almacenamiento de datos, y el acceso a los mismos, distribuida, se libera de una gran carga computacional a los equipos implicados en el sistema.	Los sistemas de bases de datos móviles están formados por un sólo tipo gestor de bases de datos, y todos los equipos conectados al sistema atacan al mismo gestor de base de datos. A diferencia de éstos, las bases de datos federadas permiten conectar diferentes sistemas gestores de bases de datos, que conforman una sola base de datos.
Para acceder a dichos sistemas se necesita una interfaz, adaptada al dispositivo desde el cual se va a acceder, que dé soporte y acceso a las funcionalidades disponibles al usuario por parte del sistema.	A los sistemas de bases de datos móviles se accede por medio de dispositivos móviles. Éstos, a su vez, acceden por medio de estaciones base y están comunicados directamente con la base de datos, aunque también se puede acceder desde equipos fijos. Las bases de datos federadas sólo son accesibles desde equipos fijos conectados a la infraestructura del sistema de bases de datos.

Fuente: Propia

3.2.4 EJERCICIO DE ENTRNAMIENTO

SGBDF

El siguiente taller de entrenamiento se propone para validar la aprehensión de los conceptos. Después de realizar la lectura del módulo, revisar los videos y los enlaces recomendados, se debe leer detenidamente las preguntas propuestas y dar respuesta a ellas.

1. Defina en sus propias palabras que es una Base de Datos Federada.
2. ¿Cuáles son las principales características de los SGBDF?
3. ¿En qué consiste el SGBDF fuertemente acoplados?
4. ¿En qué consisten los dos tipos de arquitecturas para el manejo de bases de datos federadas?
5. La implementación de este tipo de base de datos concierne una serie de problemas que se cite algunos de ellos.
6. ¿Qué son las Bases de Datos Móviles?
7. ¿Cuál es la arquitectura general de una plataforma móvil?
8. Hay dos modos de funcionamiento para trabajar con los datos. Explíquelos.
9. ¿Cuáles son las tres categorías de en qué se clasifican los datos?
10. Cite tres ejemplos de Bases de Datos Móviles.

3.3 TEMA 3 RENDIMIENTO DE BASES DE DATOS

3.3.1.1 BASES DE DATOS PARALELAS

De forma general el concepto de paralelismo en las bases de datos lo podríamos definir como la partición de la base de datos (normalmente a nivel de relaciones) para poder procesar de forma paralela en distintos discos y con distintos procesadores una sola operación sobre la base de datos. **Hace unos años este tipo de bases de datos estaban casi descartadas** pero actualmente casi todas las marcas de bases de datos venden este producto con éxito. Esto se ha debido a:

1. Los requisitos transaccionales que tienen las empresas han aumentado al mismo tiempo que ha crecido el empleo de computadoras. Además los sitios web tienen millones de visitantes para los que se requieren bases de datos enormes.
2. Las empresas utilizan cada vez mayores volúmenes de datos para planificar sus actividades. Las consultas usadas para estos fines son de ayuda a la toma de decisiones y pueden necesitar hasta varios terabytes de datos que no se pueden manejar con un único procesador en el tiempo necesario.
3. La naturaleza orientada a conjuntos de las consultas se presta a la paralelización.
4. Las máquinas paralelas con varios procesadores son relativamente baratas.

El paralelismo se usa para mejorar la velocidad en la ejecución de consultas. Además, el paralelismo se usa para proporcionar dimensionabilidad ya que la creciente carga de trabajo se trata sin incrementar el tiempo de respuesta, pero incrementando el grado de paralelismo. Existen cuatro arquitecturas de sistemas paralelos:

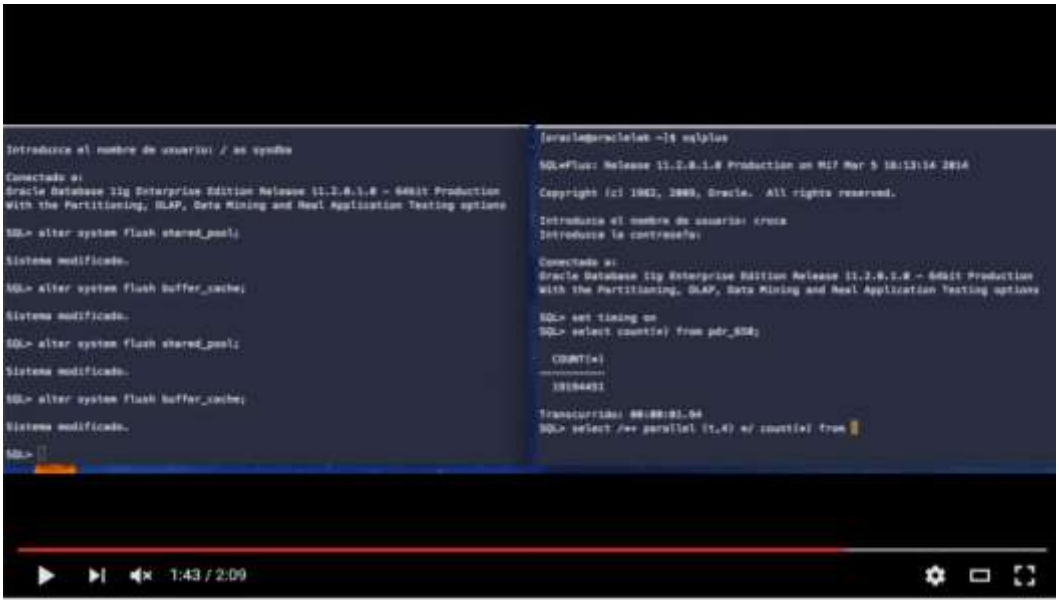
1. De memoria compartida: Todos los procesadores comparten una memoria común.
2. De discos compartidos: Todos los procesadores comparten un conjunto de discos común.
3. Sin compartimiento: Los procesadores no comparten ni memoria ni disco.
4. Jerárquica: Este modelo es un híbrido de las arquitecturas anteriores.

Paralelismo de E/S: De forma general podemos hablar de paralelismo de E/S cuando hablamos de divisiones en las relaciones entre varios discos para reducir el tiempo necesario de su recuperación. **Normalmente la división más común en un entorno de bases de datos paralelas es la división horizontal.** En este tipo de división las tuplas de cada relación se dividen entre varios discos de modo que cada tupla resida en un disco distinto.

Suponiendo que tenemos n discos (D0, D1,…, Dn-1) entre los que se van a dividir los datos, existen varias estrategias de división:

Para dar más claridad del tema se sugiere revisar los siguientes links:

Base Datos Distribuidas Y Paralelas: Enlace



Ejecución de consultas en paralelo en bases de datos Oracle: Enlace

Enlace: https://modelosbd2012t1.wordpress.com/2012/03/24/base-de-datos-paralelas/

3.3.1.2 BASES DE DATOS GRID

Grid es una tecnología que surgió como una nueva forma de computación distribuida. Ian Foster y Carl Kesselman son considerados los padres de esta tecnología, introducida por ellos en los años 90. Esta tecnología se basa en la utilización de recursos externos además de los locales, logrando con ello una mayor disponibilidad de recursos para la realización de una tarea.

La tecnología estándar, o una de las más utilizadas en su comienzo al menos, es el Globus Toolkit. El objetivo es permitir el uso de recursos libres de otras computadoras localizadas en otro lugar geográfico y que no estén utilizando toda su potencia. De este modo alguien que no disponga de la suficiente potencia o recursos en su lugar de trabajo, no se verá imposibilitado para realizar la tarea deseada ya que podrá hacer uso de recursos ajenos.

Como consecuencia del uso de una red Grid, un usuario puede hacer uso de recursos libres situados en los computadores que se encuentren dentro de esta red Grid, sin importar la localización del mismo. De este modo, el usuario dispone de un computador ficticio con la potencia, disco duro o memoria RAM necesitada.

Por otro lado, podemos decir que con Grid, no ponemos atención en los datos que se transmiten en sí, como es el caso de los sistemas Cliente-Servidor sino que el punto de interés y estudio son los recursos computacionales y el uso que se hace de ellos.

Otro avance de Grid es que genera un incremento de las posibilidades del uso de internet ya que proporciona un incremento de su usabilidad. De este modo se obtiene una mayor velocidad de procesamiento así como la facilidad de tener bases de datos de mayor tamaño.

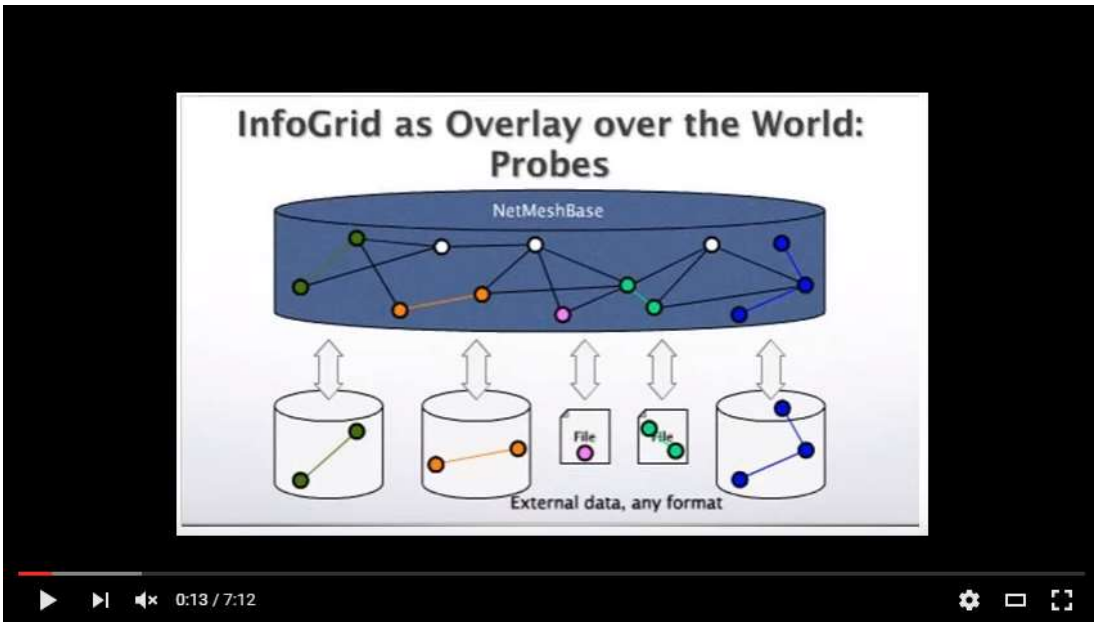
Una definición de computación Grid encontrada en wikipedia es la siguiente:

GRID COMPUTING: Es una tecnología innovadora que permite utilizar de forma coordinada todo tipo de recursos (entre ellos cómputo, almacenamiento y aplicaciones específicas) que no están sujetos a un control centralizado. En este sentido es una nueva forma de computación distribuida, en la cual los recursos pueden ser heterogéneos (diferentes arquitecturas, supercomputadores, clusters...) y se encuentran conectados mediante redes de área extensa (por ejemplo, Internet).

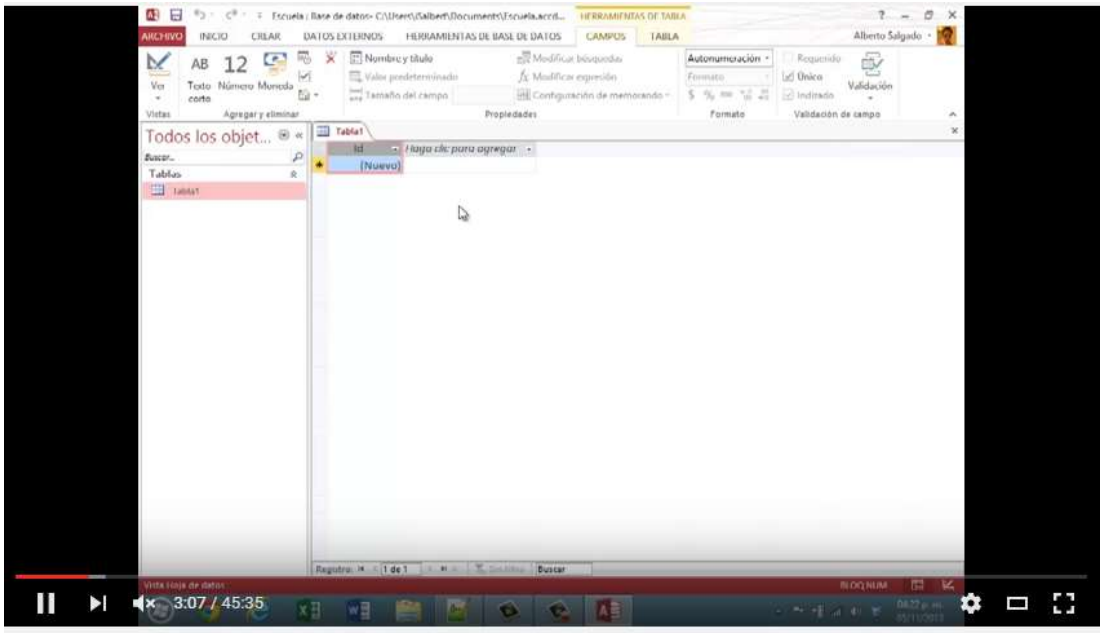
Desarrollado en ámbitos científicos a principios de los años 90 su entrada al mercado comercial siguiendo la idea de la llamada Utility Computing supone una revolución que dará mucho que hablar. Las características de esta arquitectura serían:

1. Capacidad de balanceo de sistemas: no habría necesidad de calcular la capacidad de los sistemas en función de los picos de trabajo, ya que la capacidad se puede reasignar desde la granja de recursos a donde se necesite;
2. Alta disponibilidad. Con la nueva funcionalidad, si un servidor falla, se reasignan los servicios en los servidores restantes;
3. Reducción de costos: Con esta arquitectura los servicios son gestionados por "granjas de recursos". Ya no es necesario disponer de "grandes servidores" y podremos hacer uso de componentes de bajo costo.

Para dar claridad al tema, se propone revisar los siguientes link:



BASES DE DATOS EN GRID: Enlace



Base de datos y load Grid:Enlace

Enlace: https://modelosbd2012t1.wordpress.com/2012/03/24/base-de-datos-grid-2/

http://www.scribd.com/doc/26369362/Base-de-Datos-Grid-y-Paralelas2#scribd

Seguridad en Bases de Datos Grid

Para utilizar Base de Datos en el GRID, estas primeramente deben de cumplir una serie de condiciones previas, como son las normas de seguridad en un GRID.
Algunos aspectos claves de la seguridad en el GRID son:

1. **Autentificación:** Verificación de la validez de la identidad de un usuario, recurso, servicio,..
2. **Autorización:** Cada recurso o usuario solo debe usar los servicios para los que está permitido (control de acceso).
3. **Integridad:** Asegura que los datos no han sido alterados fraudulentamente.
4. **Confidencialidad:** Información sensible como puede ser información de carácter. personal, orientación sexual, datos médicos o bancarios, no puede ser observada por terceros.
5. **Gestión de claves:** Hace referencia a la gestión de seguridad, proceso de distribución, generación y almacenamiento de claves.
6. **Encriptación:**
7. **Simétrica:** El proceso de encriptación se realiza usando la misma clave privada.
8. **Inconvenientes**: El emisor y el receptor deben intercambiar la clave.
9. **Asimétrica**: Se utilizan dos claves diferentes para encriptar y desencriptar datos. Criptografía de clave pública.
10. **Lentitud** considerable en mensajes grandes.
11. **Aparición de patrones** que puede simplificar su criptoanálisis.

12. **Secure Socket Layer/ Transport Layer Security (SSL/ TLS):** Protocolo de comunicación segura.
13. **Autentificación Mutua:** Dos entidades que quieren comunicarse usan su clave pública almacenada en un certificado digital para autentificarse.

Estos servicios fundamentales se garantizan mediante GRID Security Infrastructure (GSI) y Public Key Infrastructure (PKI).

Integración de Bases de datos en un sistema Grid: En los sistemas de bases de datos Grid los servicios ofrecidos deben estar, en la medida de lo posible estandarizados, y decimos en la medida de lo posible ya que resulta imposible que todos los servicios se estandaricen debido a la existencia de distintos tipos de bases de datos que pueden existir en el sistema y que pueden estar usando lenguajes diferentes que no pueden ser integrados en un único lenguaje debido a su naturaleza. De esta forma podemos aumentar la portabilidad de dichos sistemas. Otra de las ventajas de estandarizar siempre es reducir el esfuerzo de construcción del sistema.

Con lo mencionado anteriormente se hace imprescindible la presencia de los metadatos en este tipo de sistemas. Con los metadatos lo que conseguimos es que cada sistema que se conecta al Grid pueda comunicar al resto los servicios que ofrece. Del mismo modo podremos saber las operaciones que soporta cada uno. El sistema gestor de bases de datos (SGBD) va a ser el encargado de saber qué servicios ofrece cada una de las bases de datos, que operaciones se pueden realizar sobre ellas y de gestionar los permisos de acceso a cada una.

Los servicios que cada sistema debe tener disponibles dentro del Grid son los siguientes:

SERVICIOS TECNOLOGIA GRID	CARACTERISTICA
Metadatos	Nos dan la información sobre los servicios que ofrece el sistema. Además, cuando los usuarios del sistema soliciten un servicio no saben en qué sistema está y mediante los metadatos él se pueden construir dinámicamente las interfaces para acceder a los distintos sistemas de bases de datos que forman parte del Grid.
Manejo de consultas	Como hemos comentado más arriba los lenguajes pueden ser diferentes. Por eso en los metadatos se proporciona la información necesaria sobre el lenguaje de consulta que soporta cada base de datos. También es importante que los resultados de una consulta se puedan enviar a distintos destinos y que sean comprensibles por éstos para poder construir sistemas más amplios y complejos.
Transacciones	Estas operaciones son en las que interviene un único sistema de base de datos y a su vez que cada sistema individual tome parte en las transacciones distribuidas. La gran variedad de tipos de transacciones que maneja el sistema gestor de base de datos de un sistema Grid, debido sobre todo a la heterogeneidad de los sistemas individuales que lo componen, hace que el servicio deba poner claramente en conocimiento del resto cual es el tipo de transacciones que soporta el sistema individual de base de datos.
Carga del sistema o carga de datos	Cuando tenemos grandes cantidades de taos este tipo de servicio debe ser capaz de acceder a los protocolos de comunicación del sistema Grid para llevar a cabo la transferencia de esos datos.
Notificación	Sirve para notificar los cambios que se producen a los clientes que deseen recibir esa información. Los clientes deben poder expresar si están interesados en recibir las notificaciones cuando se inserten o se borren datos o cuando se realicen actualizaciones o en caso de varias acciones como insertar y actualizar. La forma más sencilla de que este servicio se ponga en funcionamiento es que el sistema gestor de base de datos subyacente proporcione la ayuda necesaria, por ejemplo mediante disparadores.

Planificación	Se debe permitir por ejemplo que cuando un superordenador conecte con un DBS, la información recuperada del DBS se pueda procesar por el superordenador.
A tener en cuenta	El ancho de banda en la red que los conecta necesita ser reservada. Como el acceso exclusivo a un DBS no es práctico, se requieren mecanismos con suficientes recursos (discos, CPUs, memoria, red).

Para dar claridad a los temas mencionados se sugiere revisar los siguientes Link:

Enlace: https://prezi.com/bmj2et5peblq/bases-de-datos-grid-y-paralelas/

https://ingenierosinformatica9.wordpress.com/

https://es.wikipedia.org/wiki/Computaci%C3%B3n_grid

3.3.2 TALLER DE ENTRENAMIENTO

BASES DE DATOS PARALELAS Y GRID

El siguiente taller de entrenamiento se propone para validar la aprehensión de los conceptos. Después de realizar la lectura del módulo, revisar los videos y los enlaces recomendados, se debe leer detenidamente las preguntas propuestas y dar respuesta a ellas.

1. Defina el concepto de Paralelismo en las Bases de Datos.
2. ¿Cuáles con las técnicas de división? Realice una comparativa entre ellas.
3. ¿Cuáles son las dos maneras de ejecutar en paralelo una sola consulta?
4. Dependiendo del criterio en la división de la relación se pueden distinguir dos tipos de ordenación. Explique cada uno de ellos.
5. Existe un problema por el cual no todas los tipos de reuniones pueden ser divididas por lo que existen distintas formas de proceder. ¿Cuáles son?. Explíquelas.
6. ¿Qué son los Optimizadores de consultas?
7. Un gran sistema paralelo de bases de datos debe abordar aspectos de disponibilidad. ¿Cuáles son?
8. Defina: Bases de Datos GRID.
9. Mencione algunos aspectos claves de la seguridad en el GRID.
10. ¿Cuáles son los servicios que cada sistema debe tener disponibles dentro del Grid?
11. Mencione algunas Ventajas e Inconvenientes de las BBDD en un Sistema GRID.

12. ¿Por qué son necesarios los metadatos?
13. ¿Cuáles son los servicios que describen los metadatos?

3.4 TEMA 4 BIG DATA

Antes de dar inicio al tema de Big Data se sugiere revisar los siguientes links de manera introductoria.

Enlace: http://www.ie.edu/fundacion_ie/Comun/Publicaciones/Publicaciones/Big%20Data%20ESP%207.pdf

https://www.centrodeinnovacionbbva.com/sites/default/files/bigdata_spanish.pdf

https://my.laureate.net/faculty/webinars/Documents/2013Agosto_Big%20data%20y%20la%20inteligencia%20de%20negocios.pdf

https://www.ibm.com/developerworks/ssa/local/im/que-es-big-data/

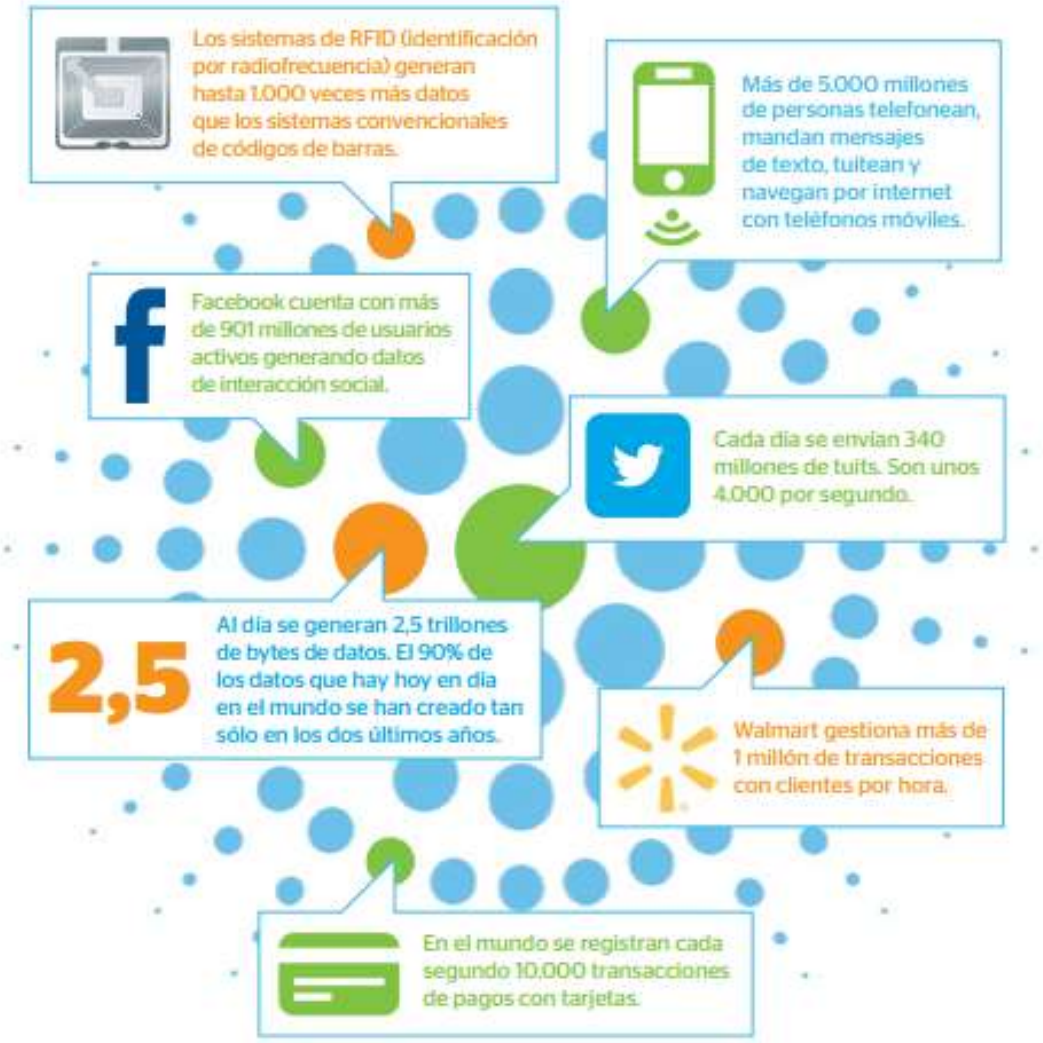
Big Data Analytics: Answers from Big Data: Enlace

Correos electrónicos, mensajes de textos, datos en formularios de Internet, vídeos, compras y facturación online, encuestas, máquinas que hacen mediciones climatológicas o de ingeniería, blogs, Twitter, Facebook, Tumblr... Hay demasiada información digital disponible en el mundo de hoy. Una tormenta constante que crece y se acumula. De hecho, cada día se generan 2,5 trillones de bytes de datos, según IBM.

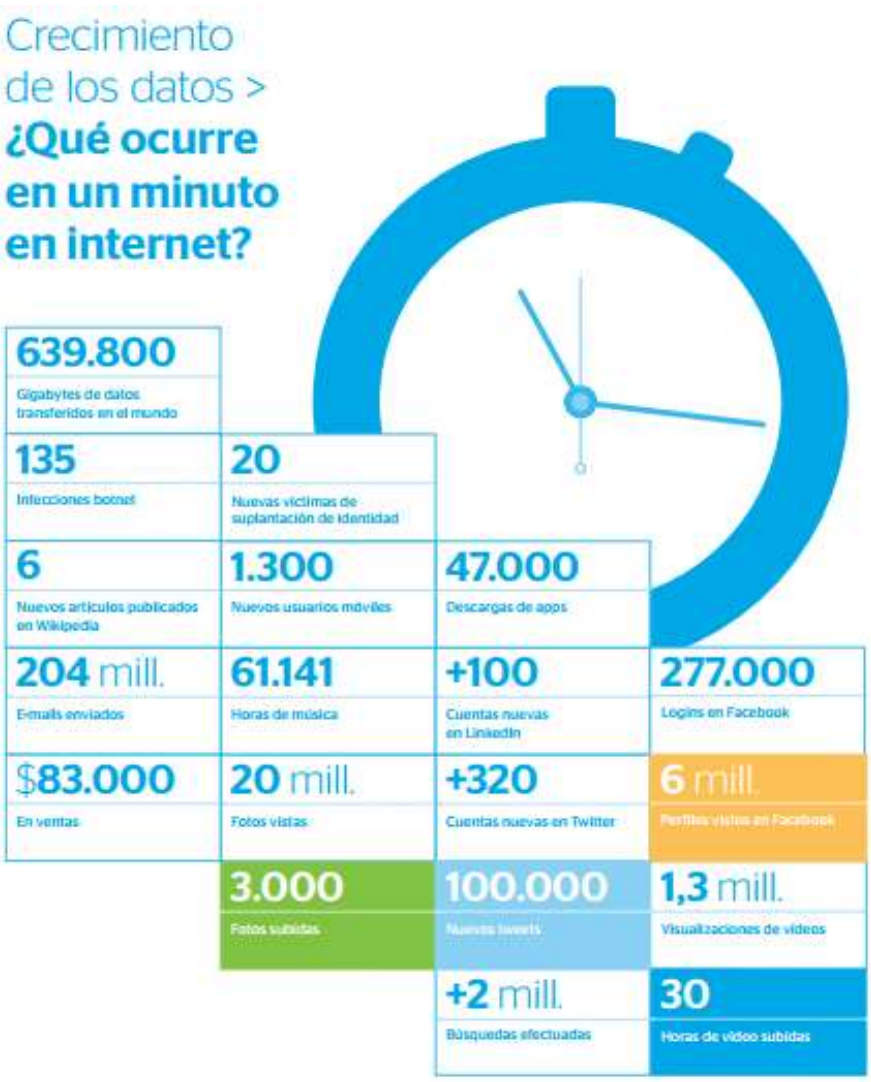
Big Data es una definición utilizada en tecnología para referirse a la información o grupo de datos que por su elevado volumen, diversidad y complejidad no pueden ser almacenados ni visualizados con herramientas tradicionales. Las dimensiones de estos datos obligan a las empresas a buscar soluciones tecnológicas para gestionarlos, pues un buen manejo del Big Data puede representar nuevas métodos para la toma de decisiones y oportunidades de negocio. El reto consiste en saber distinguir lo válido de lo superfluo y sacar provecho de ello

Otro concepto puede ser que Big Data es el término que se emplea hoy en día para describir el conjunto de procesos, tecnologías y modelos de negocio que están basados en datos y en capturar el valor que los propios datos encierran. Esto se puede lograr tanto a través de una mejora en la eficiencia gracias al análisis de los datos (una visión más tradicional), como mediante la aparición de nuevos modelos de negocio que supongan un motor de crecimiento. Se habla mucho del aspecto tecnológico, pero hay que tener presente que es crítico encontrar la forma de dar valor a los datos para crear nuevos modelos de negocio o de ayudar a los existentes. (BBVA New Technologies, 2015)

Ejemplos del mundo real de Big Data



Fuente: BBVA New Technologies, 2015

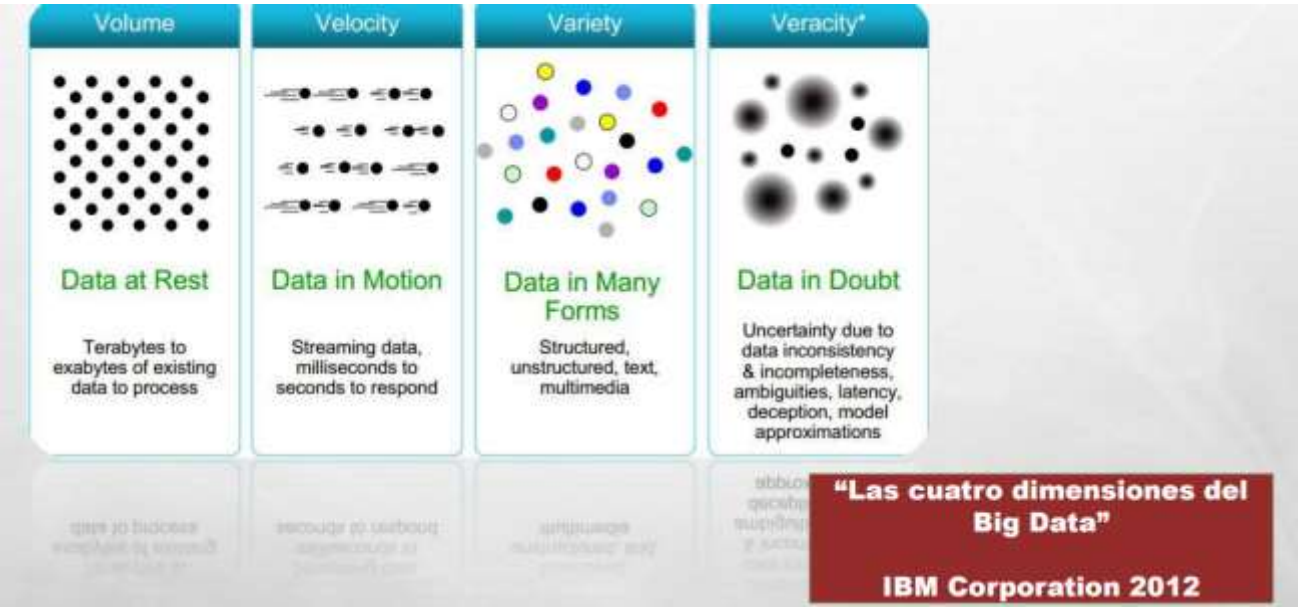


Fuente: BBVA New Technologies, 2015

CONCEPTOS CLAVES ASOCIADOS AL BIG DATA:

1. Volumen: El tamaño de la información.
2. Velocidad: Incluye tanto la media de velocidad en la que llegan los datos y también el tiempo en el que se debe actuar.
3. Variedad: Se refiere a la heterogeneidad de los datos, su representación y su semántica. Puede ser estructurada o no estructurada.
4. Privacidad: Los usuarios deben sentir confianza para suministrar la información. Las empresas deben tener procesos estrictos para su utilización. La protección de datos debe ser una prioridad.
5. Veracidad: Tiene que ver con la precisión y la confianza de los datos que se manejan.
6. Complejidad: Tiene que ver con transformar datos operativos en grandes plataformas de Big Data y la dificultad que implica gestionarlos en cualquier momento y desde cualquier lugar. La información

puede ser estructurada (base de datos, transacciones, claves, columnas, registros) o no estructurada (correos electrónicos, informes, hojas de cálculo)

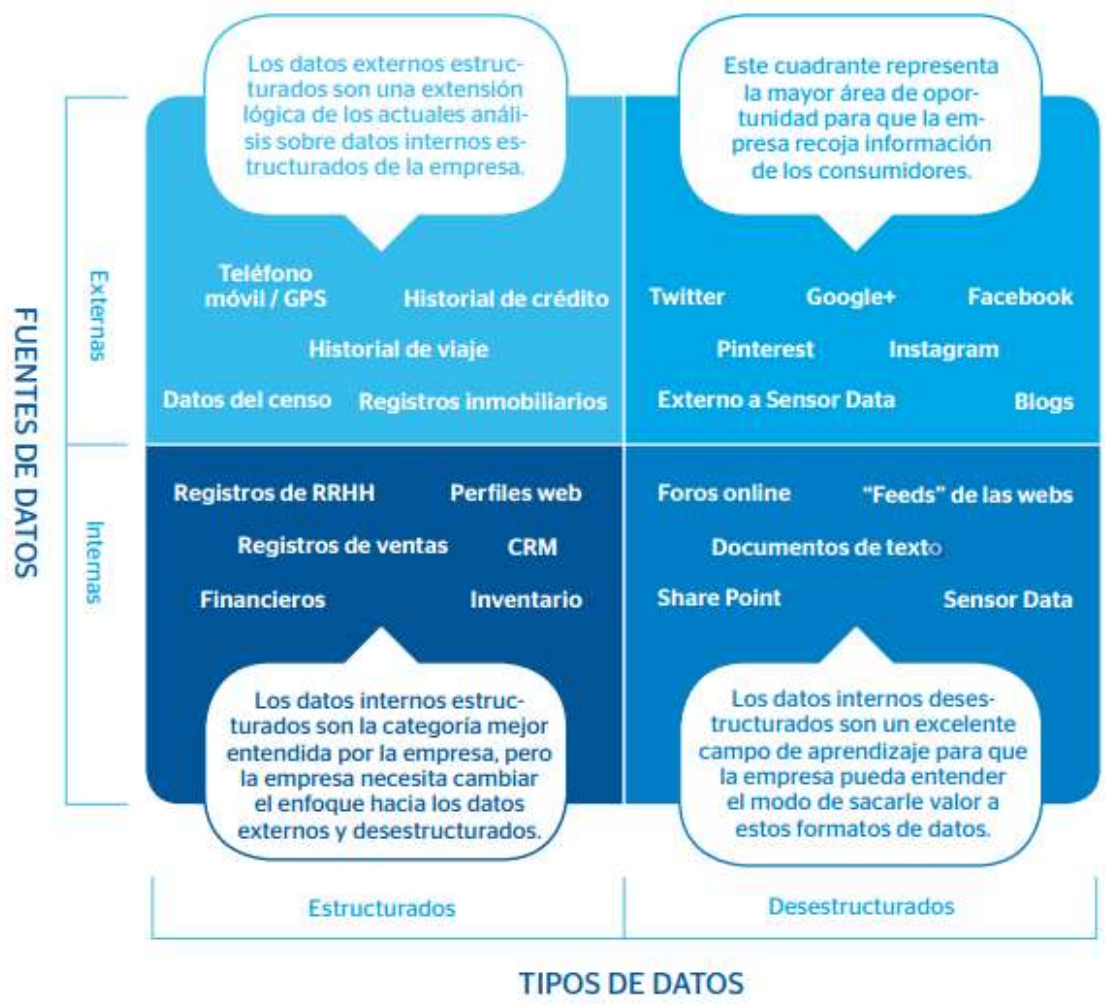


Fuente: IBM Corporation 2012

¿DE DÓNDE VIENEN ESOS DATOS?

La información relativa al Big Data tiene varias fuentes:

1. Generados por las personas a través mensajes de texto, vídeos, notas de voz.
2. Transacción de Big Data: registros de facturación y de llamadas telefónicas.
3. Redes sociales y web: correos electrónicos, redes sociales, blogs y contenidos de las páginas web.
4. Machine to machine (M2M): Son las tecnologías que comparten datos con dispositivos: medidores de temperatura, de altura, presión, química, etc que transforman información en valores.
5. Biométrica: los datos biométricos usados en el mundo de la seguridad e inteligencia.



Fuente: Booz & Company | Benefitting from Big Data, 2012

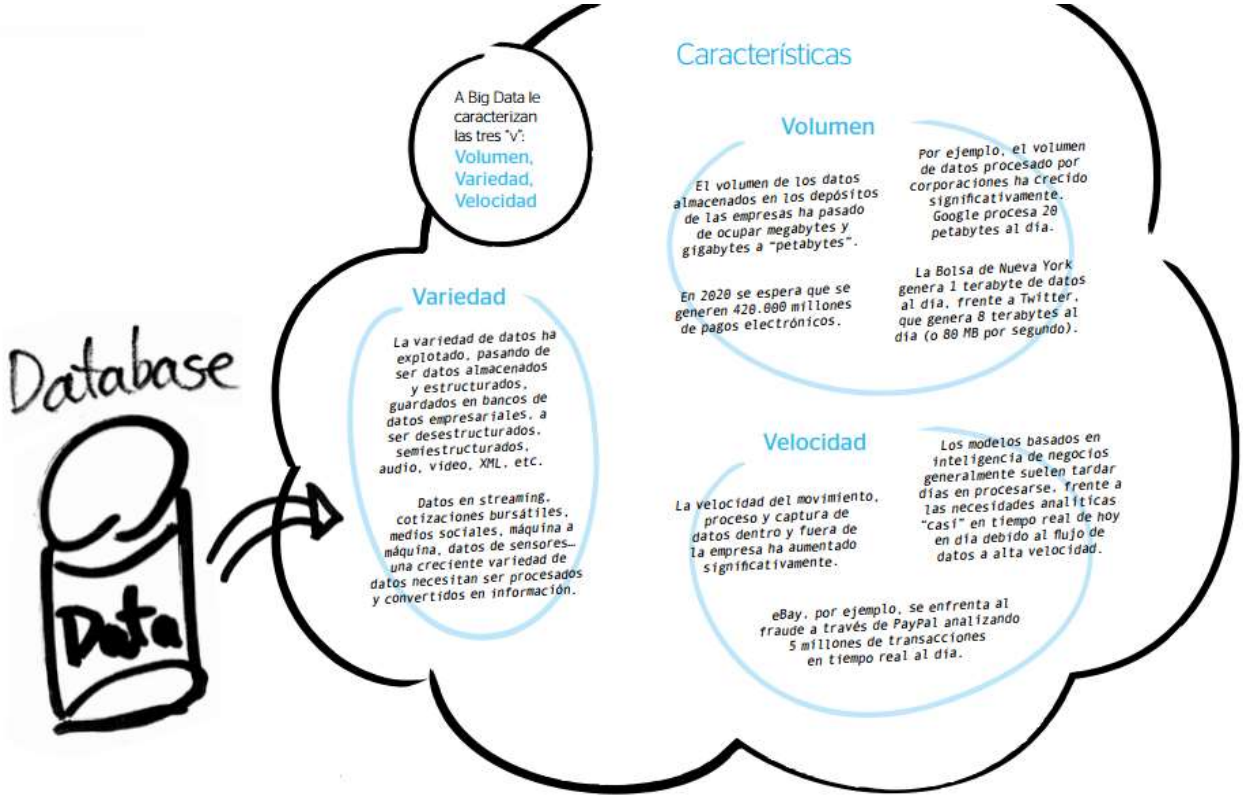
OBJETIVOS, VENTAJAS Y POSIBILIDADES

Con tanta información disponible en Internet, las empresas tienen que ser capaces de determinar cuándo habla su cliente y poder recoger, procesar, asimilar y gestionar esa información. Debe preguntarse ¿Qué nos están diciendo los datos? ¿Cómo los puedo usar para obtener beneficio?.

El valor de los datos se desvela cuando se pueden relacionar con otros y se puede generar información nueva, poderosa y que proporcione nuevas oportunidades.

El Big Data, calificado por los expertos como uno de los motores de la empresa digital, permite crear servicios basados en el manejo de datos, la reducción de costes y de tiempo empleado, el incremento de la productividad, un mejor posicionamiento con respecto a la competencia y valor diferencial. No sólo se trata de diseñar nuevas y enormes bases de datos, implica sacar el mejor rendimiento a la información que se tiene.

La información que proporciona el Big Data ayuda a las empresas a conocer el patrón de comportamiento de los clientes y del mercado, por lo que se recomienda que los gerentes diseñen planes de aplicación de Big Data. El aprovechamiento de Big Data puede servir para apoyar las campañas y estrategias de marketing, facilitar los procedimientos de control de calidad, ayudar en la auditoría, mejorar el servicio al cliente y el cumplimiento de normativas, gestionar mejor el riesgo, etc.



Fuente: BBVA New Technologies, 2015

Plataforma Big Data: Para el correcto análisis y gestión de Big Data existen plataformas, productos y sistemas.

Big Data Analytics: es el proceso de analizar grandes cantidades de datos en tiempo real para descubrir patrones ocultos, correlaciónales, desconocidos a información útil y que así las empresas tomen mejores decisiones para incrementar la rentabilidad de su negocio... Los programas más utilizados son Hadoop (implementado por Google), los llamados NoSQl y los almacenes de datos MPP (Procesos Masivamente Paralelos).

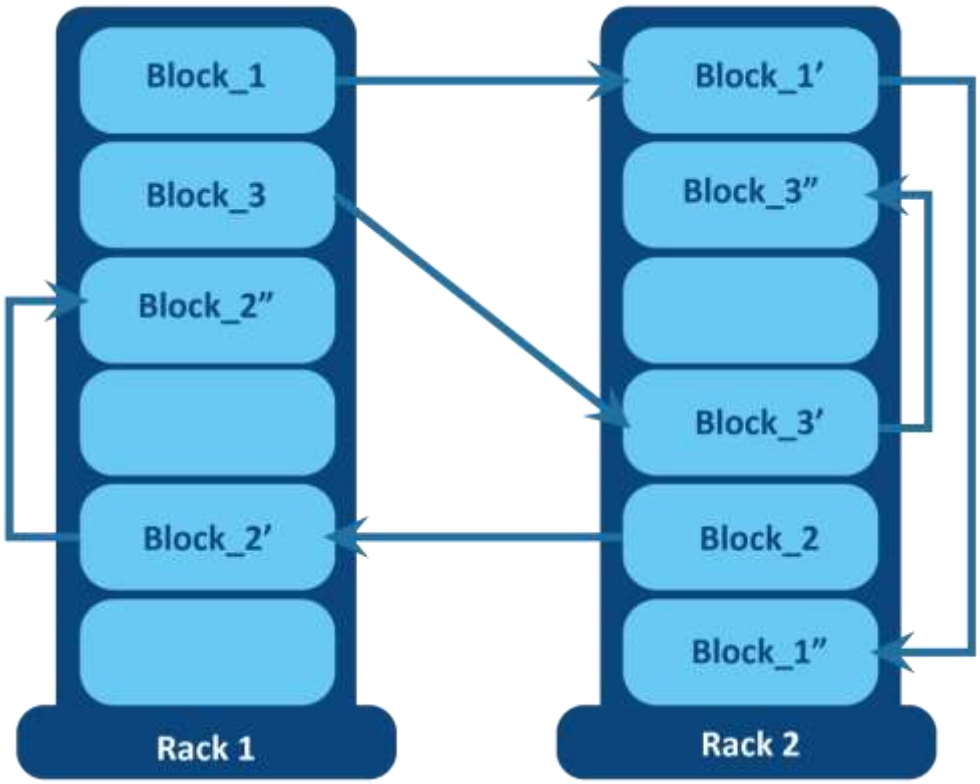
Hadoop es un software de código abierto que permite el almacenamiento y distribución de ficheros con terabytes y pentabytes de enormes dimensiones en los que la información no necesita estar estructurada. Es escalable, permite seleccionar los datos susceptibles a los análisis, es tolerante a fallos e impide pérdidas de información, tiene menor coste por terabyte y brinda la posibilidad de análisis paralelos complejos.

Hadoop está inspirado en el proyecto de Google File System(GFS) y en el paradigma de programación MapReduce, el cual consiste en dividir en dos tareas (mapper – reducer) para manipular los datos distribuidos a nodos de un

clúster logrando un alto paralelismo en el procesamiento. Hadoop está compuesto de tres piezas: Hadoop Distributed File System (HDFS), Hadoop MapReduce y Hadoop Common.

Hadoop Distributed File System(HDFS): Los datos en el clúster de Hadoop son divididos en pequeñas piezas llamadas bloques y distribuidas a través del clúster; de esta manera, las funciones map y reduce pueden ser ejecutadas en pequeños subconjuntos y esto provee de la escalabilidad necesaria para el procesamiento de grandes volúmenes.

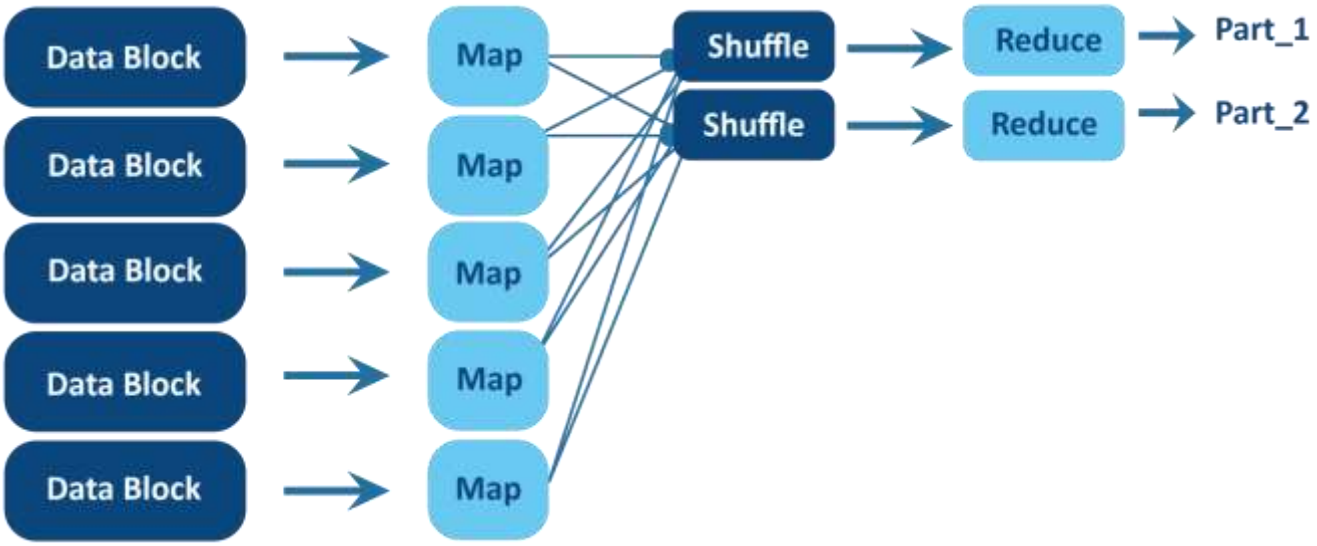
La siguiente figura ejemplifica como los bloques de datos son escritos hacia HDFS. Observe que cada bloque es almacenado tres veces y al menos un bloque se almacena en un diferente rack para lograr redundancia.



Fuente: https://www.ibm.com/developerworks/ssa/local/im/que-es-big-data/

HADOOP MAPREDUCE

MapReduce es el núcleo de Hadoop. El término MapReduce en realidad se refiere a dos procesos separados que Hadoop ejecuta. El primer proceso map, el cual toma un conjunto de datos y lo convierte en otro conjunto, donde los elementos individuales son separados en tuplas (pares de llave/valor). El proceso reduce obtiene la salida de map como datos de entrada y combina las tuplas en un conjunto más pequeño de las mismas. Una fase intermedia es la denominada Shuffle la cual obtiene las tuplas del proceso map y determina que nodo procesará estos datos dirigiendo la salida a una tarea reduce en específico. La siguiente figura ejemplifica un flujo de datos en un proceso sencillo de MapReduce.



Fuente: https://www.ibm.com/developerworks/ssa/local/im/que-es-big-data/

Otro muy usado es Map Reduce: es un modelo de programación utilizado para dar soporte a la computación de grupos de datos de ordenadores. Es utilizado por Google.

CASOS/ EJEMPLOS DE APLICACIÓN DE BIG DATA

El Big Data puede aplicarse en campos tan diversos entre sí como la investigación médica, la seguridad, la administración pública, logística y relación con el cliente. Los expertos consideran que puede revolucionar la ciencia, la investigación, la educación, el planeamiento urbano, el transporte inteligente, el ahorro de energía, la conservación del medio ambiente y los sistemas de riesgo de análisis financiero. Exponemos aquí algunos ejemplos:

1. La banca puede combinar los patrones de compra y pago de sus clientes con los datos de su nómina y las posibilidades de crédito. Hacer esto a gran escala le puede generar oportunidades de negocio para ofrecer nuevos productos financieros.
2. Empresas de telefonía observan el comportamiento de sus clientes, cómo es el contrato, cuánto pagan, cuál teléfono utilizan, cada cuánto cambian el aparato. Así pueden extraer cuál es el plan más exitoso según rangos de edad, cuál teléfono es el más popular y qué le pueden ofrecer.
3. Los grandes almacenes por departamento utilizan los datos de sus clientes para ver cuánto se ha vendido durante las rebajas, cuál producto se ha vendido más en todo el país, cuál producto debe retirarse del mercado, cuáles son las quejas de sus clientes.
4. Una empresa de agricultura puede cruzar los datos metereológicos con el funcionamiento de sus sistemas de riego y así toma la decisión de cuáles días deberá regar y con cuánta agua.
5. Un hospital puede determinar las horas críticas, las patologías con mayor reincidencia, los materiales que se utilizan más, cómo se puede rentabilizar la utilización de los quirófanos, la eficiencia energética, la base de datos de los pacientes y su historia médica electrónica.
6. Las empresas eléctricas puede revisar la información que le proporcionan los medidores para verificar el consumo y la demanda, verificar las tarifas según las zonas y puede ofrecer distintos planes.

Big Data y el campo de la Investigación: Los científicos e investigadores han analizado datos desde ya hace mucho tiempo, lo que ahora representa el gran reto es la escala en la que estos son generados.
Esta explosión de "grandes datos" está transformando la manera en que se conduce una investigación adquiriendo habilidades en el uso de Big Data para resolver problemas complejos relacionados con el descubrimiento científico, investigación ambiental y biomédica, educación, salud, seguridad nacional, entre otros.
De entre los proyectos que se pueden mencionar donde se ha llevado a cabo el uso de una solución de Big Data se encuentran:

1. El Language, Interaction and Computation Laboratory (CLIC) en conjunto con la Universidad de Trento en Italia, son un grupo de investigadores cuyo interés es el estudio de la comunicación verbal y no verbal tanto con métodos computacionales como cognitivos.
2. Lineberger Comprehensive Cancer Center - Bioinformatics Group utiliza Hadoop y HBase para analizar datos producidos por los investigadores de *The Cancer Genome Atlas(TCGA)* para soportar las investigaciones relacionadas con el cáncer.
3. El PSG College of Technology, India, analiza múltiples secuencias de proteínas para determinar los enlaces evolutivos y predecir estructuras moleculares. La naturaleza del algoritmo y el paralelismo computacional de Hadoop mejora la velocidad y exactitud de estas secuencias.
4. La *Universidad Distrital Francisco Jose de Caldas* utiliza Hadoop para apoyar su proyecto de investigación relacionado con el sistema de inteligencia territorial de la ciudad de Bogotá.
5. La *Universidad de Maryland* es una de las seis universidades que colaboran en la iniciativa académica de cómputo en la nube de IBM/Google. Sus investigaciones incluyen proyectos en la lingüistica computacional (machine translation), modelado del lenguaje, bioinformática, análisis de correo electrónico y procesamiento de imágenes.

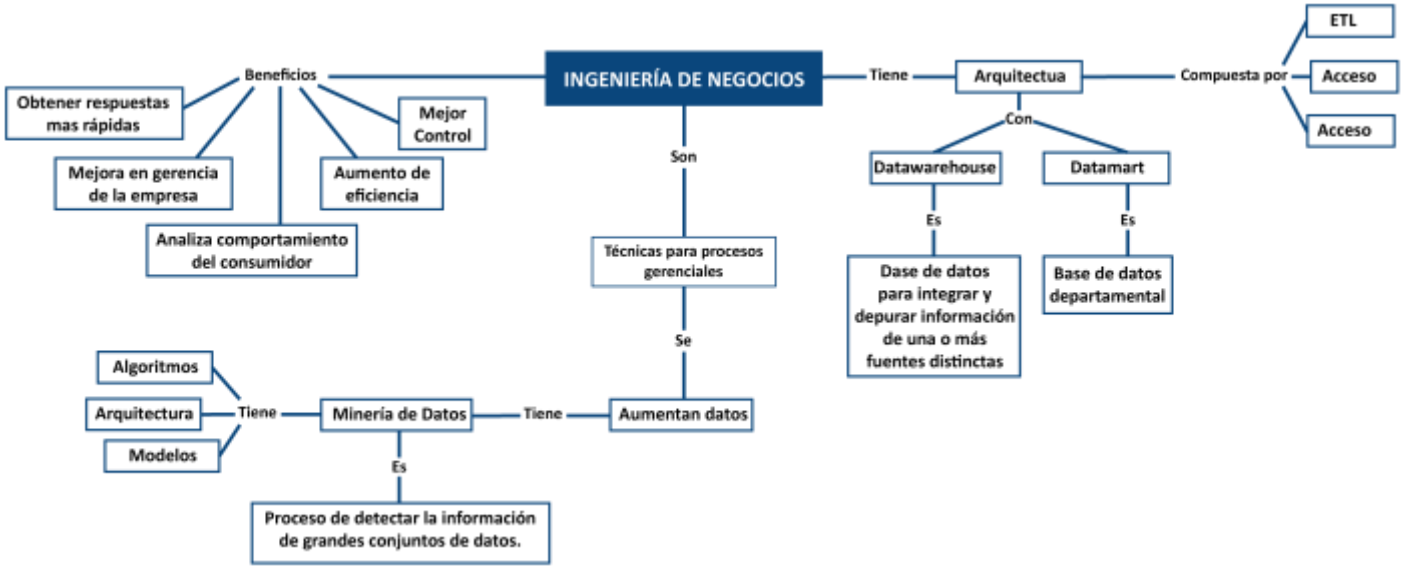
3.4.1 TALLER DE ENTRENAMIENTO

BIG DATA

El siguiente taller de entrenamiento se propone para validar la aprehensión de los conceptos. Se debe leer detenidamente las preguntas propuestas y dar respuesta a ellas.

1. Defina con sus propias palabras Big Data
2. Enuncie las características del Big Data
3. Enuncie la Importancia del Big Data

2. UNIDAD 3 INTELIGENCIA DE NEGOCIOS Y MINERIA DE DATOS



Conceptos Básicos:

Inteligencia de Negocios: conjunto de técnicas, procesos y arquitectura que transforman los datos recopilados por una compañía en información importante y relevante para los procesos gerenciales.

Datawarehouse: base de datos corporativa que se caracteriza por integrar y depurar información de una o más fuentes distintas, para luego procesarla permitiendo su análisis desde infinidad de perspectivas y con grandes velocidades de respuesta.

Minería de Datos: es el proceso de detectar la información de grandes conjuntos de datos. Utiliza el análisis matemático para deducir los patrones y tendencias que existen en los datos.

2.1 TEMA 1 INTELIGENCIA DE NEGOCIOS

Es un término acuñado por la consultora Gartner Group a finales de la década de los 80 y describe, básicamente, la capacidad de los integrantes de una empresa para acceder a la información residente en una base de datos y explorarla, de manera que el usuario pueda analizar esa información y desarrollar con ella teorías y conocimientos que serán básicos para la toma de determinadas decisiones críticas para el negocio.

Una interesante definición para inteligencia de negocios o BI, por sus siglas en inglés, según el Data Warehouse Institute, lo define como la combinación de tecnología, herramientas y procesos que me permiten transformar mis datos almacenados en información, esta información en conocimiento y este conocimiento dirigido a un plan o una estrategia comercial. La inteligencia de negocios debe ser parte de la estrategia empresarial, esta le permite optimizar la utilización de recursos, monitorear el cumplimiento de los objetivos de la empresa y la capacidad de tomar buenas decisiones para así obtener mejores resultados.

¿Por qué Inteligencia de Negocios?
¿Cuáles son algunos de los padecimientos que enfrentan las empresas hoy día?

1. Tenemos datos, pero carecemos de información – Es importante almacenar los datos de clientes, empleados, departamentos, compras, ventas, entre otros en aplicaciones, sistemas financieros o fuentes de datos. Si queremos que nuestra empresa tenga mayor ventaja sobre la competencia esta gestión no es suficiente. Necesitamos profundizar el nivel de conocimiento de nuestros clientes, empleados, operaciones para así, tener la capacidad de encontrar patrones de comportamiento, monitorear, rastrear, entender, administrar y contestar aquellas interrogantes que me permitan maximizar el rendimiento de nuestra empresa.
2. Fragmentación – Poseen aplicaciones independientes a través de todos los departamentos, pero se carece de una visión global de la empresa. Tal vez por la incapacidad de las herramientas de BI de integrar fuentes de datos heterogéneas. Esto limita a la empresa a tomar decisiones importantes sin tener todos los elementos imprescindibles a la mano. Esta fragmentación conduce a lo que se llama diferentes versiones de la verdad. Los gerenciales solicitan informes a los distintos departamentos obteniendo diferentes resultados del mismo informe. La tarea ya no es solo crear el informe sino justificar de donde y qué condiciones se utilizaron para la creación de este informe. Si el gerencial decide agregar una nueva variable a esta ecuación, recrear este informe puede conllevar un esfuerzo de semanas.
3. Manipulación manual – La necesidad de generar análisis de negocios e informes nos ha llevado a utilizar herramientas de BI y/o de reportes que no son las más confiables. Esta práctica conlleva la exportación de datos a distintas herramientas que resultan en un proceso lento, costoso, duplicación de trabajo, poca confiabilidad en los informes, propenso a errores y sujetos a la interpretación individual.
4. Poca agilidad – Debido a la carencia de información, la fragmentación y la manipulación manual me mantiene en un nivel de rendimiento bajo. Como dice el dicho: “Justo cuando me aprendí las respuestas me cambiaron las preguntas.”. Necesitamos de una herramienta lo suficientemente ágil que se ajuste a las necesidades del negocio.

BENEFICIOS DE BI

1. Ayuda a incrementar la eficiencia: Hazlo diferente a muchas compañías, que desperdician gran parte de su tiempo buscando información de departamento en departamento tratando de entender su negocio, si cuentan con suerte encontraran datos, deberán convertirlos, mezclarlos y realizar sus propios reportes, con BI toda la información se puede centralizar y visualizar en una misma plataforma y convertir en información útil y organizada, ahorrando tiempo y haciendo la toma de decisiones más eficiente.

2. Obtén respuestas más rápido para las preguntas que surgen del negocio: Un gerente debe tomar decisiones acertadas muchas veces bajo la presión del tiempo, este recurso tan preciado no se puede desperdiciar leyendo grandes cantidades de papel, informes de cada área. Con las opciones que ofrece la inteligencia de negocios, se puede obtener respuestas rápidas a grandes preguntas en minutos. Por ejemplo un solo informe de BI puede contener las cifras de ventas, de desempeño de marketing, de costos, de inventarios, de canales de distribución, etc.

3. Da pasos certeros en tu negocio con información precisa: Gerenciar un negocio es algo serio y no puede ser manejado con presentimientos o intuición, dado que esta practica no siempre funciona y puede generar grandes daños para la empresa. Con la información apropiada y estructurada se pueden tomar decisiones basadas en conocimiento que la misma empresa genera. BI puede proveer información histórica más acertada, actualizaciones en tiempo real, resumen de los datos entre sucursales, predicción y tendencias basadas en información y análisis situacional.

4. Analiza el comportamiento del consumidor: BI permite analizar hábitos de compra del consumidor y convertir esta información en rentabilidad para la empresa, también permite hacer más eficientes las campañas de fidelización. También puedes construir modelos predictivos que faciliten la venta cruzada, promociones, ventas de productos de lujo y otras estrategias dirigidas al cliente correcto gracias a la información adecuada.

5. Permite tener mejor control sobre las áreas funcionales de la empresa: Desde producción, inventario, marketing, compras, hasta servicio post-venta son susceptibles de estar incluidas en un sistema de BI, dado que en todas las áreas funcionales se utilizan y se necesitan datos, ya sea de los clientes, de los costos de materias primas, de investigación y desarrollo, en fin, el espectro de información es grande y al tenerla almacenada en un solo lugar con la posibilidad de cruzarla y analizarla en cuestión de minutos es un gran beneficio en costos y en el tiempo, disminuye los errores en la toma de decisiones.

ARQUITECTURA DE INTELIGENCIA DE NEGOCIOS

Es importante visualizar de alguna forma que comprende una arquitectura de inteligencia de negocios. En la siguiente figura nos representa esta arquitectura. Analicemos este diagrama de izquierda a derecha. Los primeros dibujos representan las distintas fuentes de datos (Cubos essbase, bases de datos Oracle, Sql Server, mainframe, archivos planos, archivos xml, hojas de Excel, etc.) que pudieran utilizarse para extraer los datos de múltiples fuentes simultáneamente.

El segundo dibujo representa el proceso de extracción, transformación y carga (ETL). Este proceso es en el que se definen de las fuentes heterogéneas que campos se van a utilizar, si necesitan algún tipo de modificación y/o transformación y donde quiero ubicar estos datos, este proceso se le conoce como "mapping".

El tercer dibujo representa el repositorio de datos. En este repositorio se encuentran los datos transformados representados visualmente en modelos multidimensionales, dimensiones y tablas de datos. Existe un proceso entre el repositorio de datos y la interfase de acceso al usuario, este es el motor de BI que me permite habilitar componentes, administrar consultas, monitorea procesos, cálculos, métricas. La interfase de acceso a usuarios permite interaccionar con los datos, representar de forma gráfica con aquellos resultados de las consultas y los indicadores de gestión que fueron construidos.

Fuente: http://www.oracle.com/ocom/groups/public/@otn/documents/webcontent/317529_esa.pdf

Para da claridad a la temática se sugiere revisar los siguientes link:

Enlace: http://sci2s.ugr.es/sites/default/files/files/Teaching/GraduatesCourses/InteligenciaDeNegocio/Tema01-Introduccion%20a%20la%20Inteligencia%20de%20negocio%202015-16.pdf

http://52.0.140.184/typo43/fileadmin/Revista_111/uno.pdf

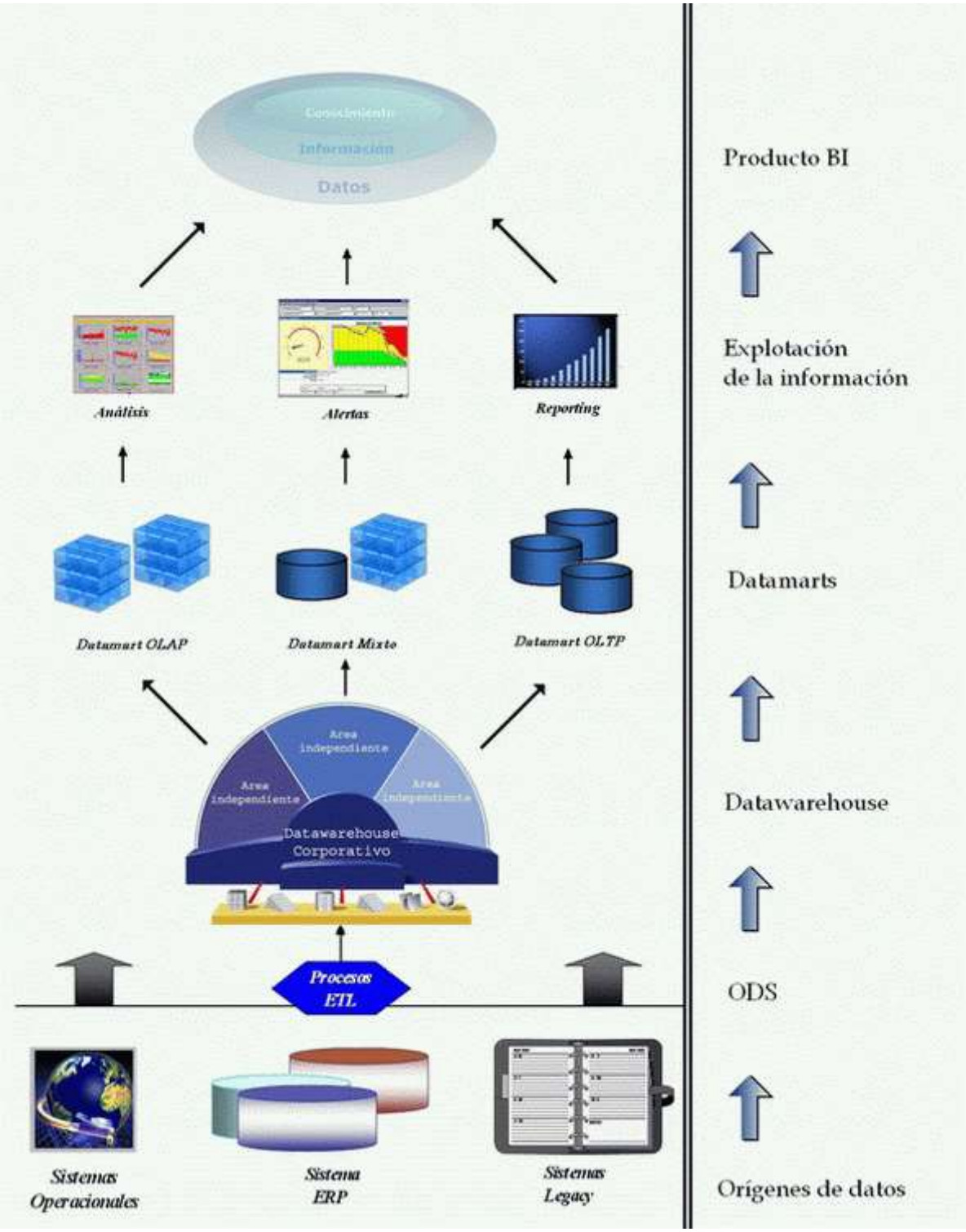
2.1 TEMA 2 DATAWAREHOUSE

Como actividad introductoria se sugiere revisar os siguientes links:

Enlace:
https://www2.deloitte.com/content/dam/Deloitte/uy/Documents/strategy/Evento%20Deloitte%20Inteligencia%20de%20Negocios.pdf

Una solución de Business Intelligence parte de los sistemas de origen de una organización (bases de datos, ERPs, ficheros de texto...), sobre los que suele ser necesario aplicar una transformación estructural para optimizar su proceso analítico. Para ello se realiza una fase de extracción, transformación y carga (ETL) de datos. Esta etapa suele apoyarse en un almacén intermedio, llamado ODS, que actúa como pasarela entre los sistemas fuente y los sistemas destino (generalmente un datawarehouse), y cuyo principal objetivo consiste en evitar la saturación de los servidores funcionales de la organización.

La información resultante, ya unificada, depurada y consolidada, se almacena en un datawarehouse corporativo, que puede servir como base para la construcción de distintos datamarts departamentales. Estos datamarts se caracterizan por poseer la estructura óptima para el análisis de los datos de esa área de la empresa, ya sea mediante bases de datos transaccionales (OLTP) o mediante bases de datos analíticas (OLAP).

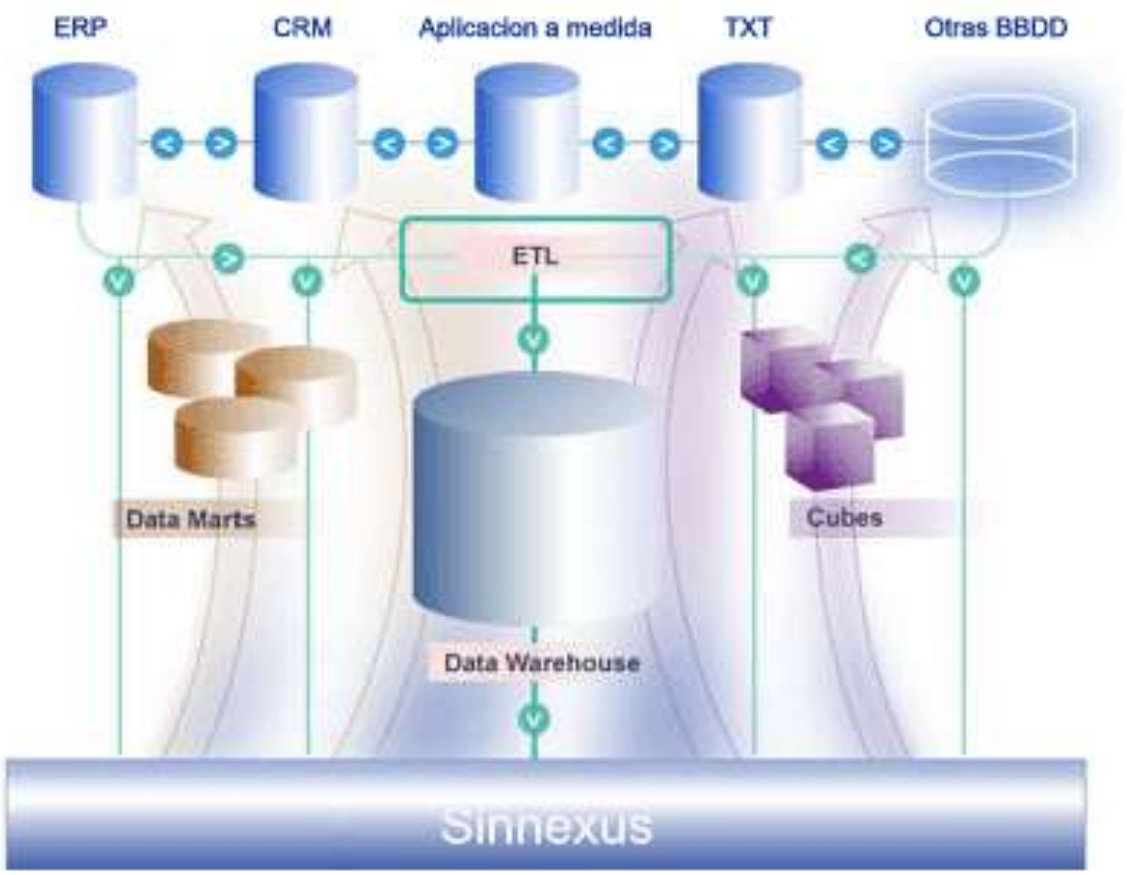


Fuente: http://www.sinnexus.com/business_intelligence/arquitectura.aspx

Los datos albergados en el datawarehouse o en cada datamart se explotan utilizando herramientas comerciales de análisis, reporting, alertas... etc. En estas herramientas se basa también la construcción de productos BI más completos, como los sistemas de soporte a la decisión (DSS), los sistemas de información ejecutiva (EIS) y los cuadros de mando (CMI) o Balanced Scorecard (BSC).

Un Datawarehouse es una base de datos corporativa que se caracteriza por integrar y depurar información de una o más fuentes distintas, para luego procesarla permitiendo su análisis desde infinidad de pespectivas y con grandes velocidades de respuesta. La creación de un datawarehouse representa en la mayoría de las ocasiones el primer paso, desde el punto de vista técnico, para implantar una solución completa y fiable de Business Intelligence.

La ventaja principal de este tipo de bases de datos radica en las estructuras en las que se almacena la información (modelos de tablas en estrella, en copo de nieve, cubos relacionales... etc). Este tipo de persistencia de la información es homogénea y fiable, y permite la consulta y el tratamiento jerarquizado de la misma (siempre en un entorno diferente a los sistemas operacionales).



Fuente: http://www.sinnexus.com/business_intelligence/datawarehouse.aspx

El término Datawarehouse fue acuñado por primera vez por Bill Inmon, y se traduce literalmente como almacén de datos. No obstante, y como cabe suponer, es mucho más que eso. Según definió el propio Bill Inmon, un datawarehouse se caracteriza por ser:

1. Integrado: los datos almacenados en el datawarehouse deben integrarse en una estructura consistente, por lo que las inconsistencias existentes entre los diversos sistemas operacionales deben ser eliminadas. La información suele estructurarse también en distintos niveles de detalle para adecuarse a las distintas necesidades de los usuarios.
2. Temático: sólo los datos necesarios para el proceso de generación del conocimiento del negocio se integran desde el entorno operacional. Los datos se organizan por temas para facilitar su acceso y entendimiento por parte de los usuarios finales. Por ejemplo, todos los datos sobre clientes pueden ser consolidados en una única tabla del datawarehouse. De esta forma, las peticiones de información sobre clientes serán más fáciles de responder dado que toda la información reside en el mismo lugar.
3. Histórico: el tiempo es parte implícita de la información contenida en un datawarehouse. En los sistemas operacionales, los datos siempre reflejan el estado de la actividad del negocio en el momento presente. Por el contrario, la información almacenada en el datawarehouse sirve, entre otras cosas, para realizar análisis de tendencias. Por lo tanto, el datawarehouse se carga con los distintos valores que toma una variable en el tiempo para permitir comparaciones.
4. No volátil: el almacén de información de un datawarehouse existe para ser leído, pero no modificado. La información es por tanto permanente, significando la actualización del datawarehouse la incorporación de los últimos valores que tomaron las distintas variables contenidas en él sin ningún tipo de acción sobre lo que ya existía.

Otra característica del datawarehouse es que contiene metadatos, es decir, datos sobre los datos. Los metadatos permiten saber la procedencia de la información, su periodicidad de refresco, su fiabilidad, forma de cálculo... etc.

Los metadatos serán los que permiten simplificar y automatizar la obtención de la información desde los sistemas operacionales a los sistemas informacionales. Los objetivos que deben cumplir los metadatos, según el colectivo al que va dirigido, son:

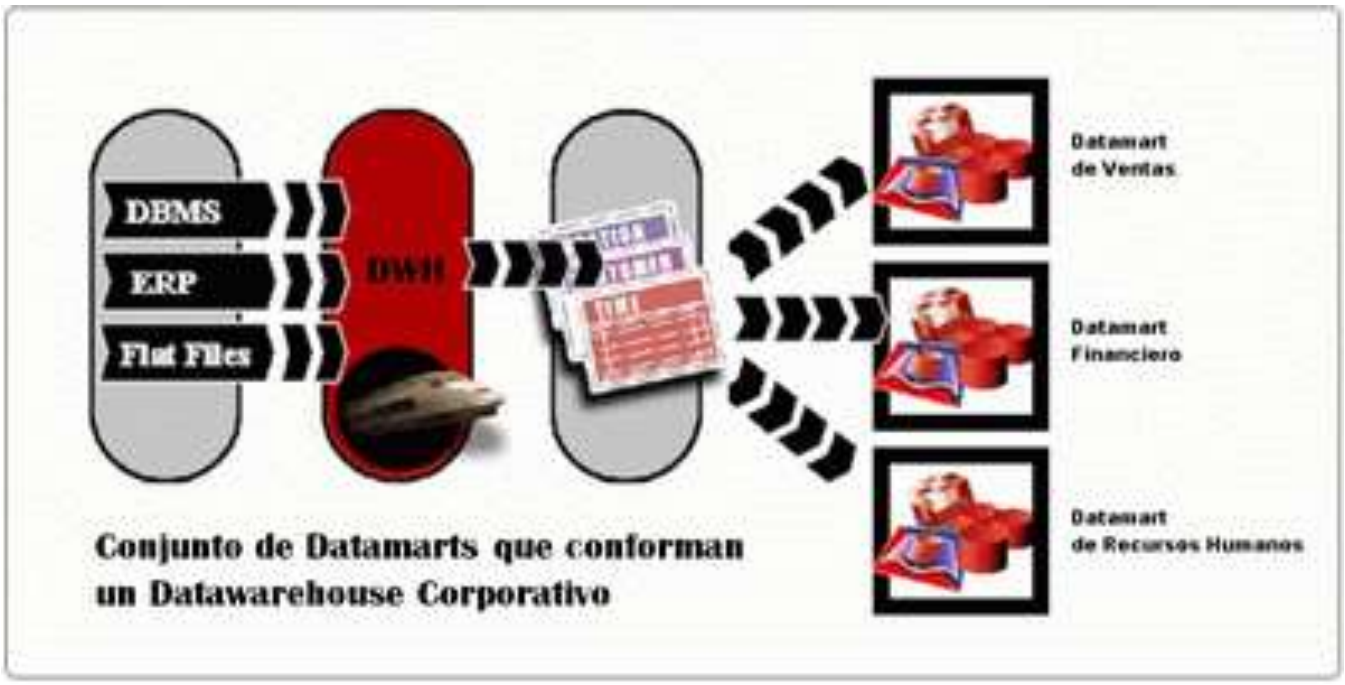
1. Dar soporte al usuario final, ayudándole a acceder al datawarehouse con su propio lenguaje de negocio, indicando qué información hay y qué significado tiene. Ayudar a construir consultas, informes y análisis, mediante herramientas de Business Intelligence como DSS, EIS o CMI.
2. Dar soporte a los responsables técnicos del datawarehouse en aspectos de auditoría, gestión de la información histórica, administración del datawarehouse, elaboración de programas de extracción de la información, especificación de las interfaces para la realimentación a los sistemas operacionales de los resultados obtenidos... etc.

3. Por último, destacar que para comprender íntegramente el concepto de datawarehouse, es importante entender cuál es el proceso de construcción del mismo, denominado ETL (Extracción, Transformación y Carga), a partir de los sistemas operaciones de una compañía:

Extracción: obtención de información de las distintas fuentes tanto internas como externas.

Transformación: filtrado, limpieza, depuración, homogeneización y agrupación de la información.

Carga: organización y actualización de los datos y los metadatos en la base de datos.



Fuente: http://www.sinnexus.com/business_intelligence/datawarehouse.aspx

Una de las claves del éxito en la construcción de un datawarehouse es el desarrollo de forma gradual, seleccionando a un departamento usuario como piloto y expandiendo progresivamente el almacén de datos a los demás usuarios. Por ello es importante elegir este usuario inicial o piloto, siendo importante que sea un departamento con pocos usuarios, en el que la necesidad de este tipo de sistemas es muy alta y se puede obtener y medir resultados a corto plazo.

PRINCIPALES APORTACIONES DE UN DATAWAREHOUSE

1. Proporciona una herramienta para la toma de decisiones en cualquier área funcional, basándose en información integrada y global del negocio.

2. Facilita la aplicación de técnicas estadísticas de análisis y modelización para encontrar relaciones ocultas entre los datos del almacén; obteniendo un valor añadido para el negocio de dicha información.
3. Proporciona la capacidad de aprender de los datos del pasado y de predecir situaciones futuras en diversos escenarios.
4. Simplifica dentro de la empresa la implantación de sistemas de gestión integral de la relación con el cliente.
5. Supone una optimización tecnológica y económica en entornos de Centro de Información, estadística o de generación de informes con retornos de la inversión espectaculares.

2.1 TEMA 3 CUBOS Y DATAMARTS

Un **Datamart** es una base de datos departamental, especializada en el almacenamiento de los datos de un área de negocio específica. Se caracteriza por disponer la estructura óptima de datos para analizar la información al detalle desde todas las perspectivas que afecten a los procesos de dicho departamento. Un datamart puede ser alimentado desde los datos de un datawarehouse, o integrar por sí mismo un compendio de distintas fuentes de información.

Por tanto, para crear el datamart de un área funcional de la empresa es preciso encontrar la estructura óptima para el análisis de su información, estructura que puede estar montada sobre una base de datos OLTP, como el propio datawarehouse, o sobre una base de datos OLAP. La designación de una u otra dependerá de los datos, los requisitos y las características específicas de cada departamento. De esta forma se pueden plantear dos tipos de datamarts:

Datamart OLAP

Se basan en los populares cubos OLAP, que se construyen agregando, según los requisitos de cada área o departamento, las dimensiones y los indicadores necesarios de cada cubo relacional. El modo de creación, explotación y mantenimiento de los cubos OLAP es muy heterogéneo, en función de la herramienta final que se utilice.

Datamart OLTP

Pueden basarse en un simple extracto del datawarehouse, no obstante, lo común es introducir mejoras en su rendimiento (las agregaciones y los filtrados suelen ser las operaciones más usuales) aprovechando las características particulares de cada área de la empresa. Las estructuras más comunes en este sentido son las tablas report, que vienen a ser fact-tables reducidas (que agregan las dimensiones oportunas), y las vistas materializadas, que se construyen con la misma estructura que las anteriores, pero con el objetivo de explotar la reescritura de queries (aunque sólo es posibles en algunos SGBD avanzados, como Oracle).

Los datamarts que están dotados con estas estructuras óptimas de análisis presentan las siguientes ventajas:

1. Poco volumen de datos
2. Mayor rapidez de consulta
3. Consultas SQL y/o MDX sencillas
4. Validación directa de la información
5. Facilidad para la historización de los datos

Fuente: http://www.sinnexus.com/business_intelligence/datamart.aspx

3.4.2 TALLER DE ENTRENAMIENTO

DATAWAREHOUSE, DATAMART Y CUBO

El siguiente taller de entrenamiento se propone para validar la aprehensión de los conceptos. Se debe leer detenidamente las preguntas propuestas y dar respuesta a ellas.

1. Defina Datawarehouse
2. Defina Datamart
3. Establezca diferencia entre Datawarehouse y Datamart

2.1 TEMA 4 MINERÍA DE DATOS

La minería de datos es el proceso de detectar la información de grandes conjuntos de datos. Utiliza el análisis matemático para deducir los patrones y tendencias que existen en los datos. Normalmente, estos patrones no se pueden detectar mediante la exploración tradicional de los datos porque las relaciones son demasiado complejas o porque hay demasiado datos.

Estos patrones y tendencias se pueden recopilar y definir como un modelo de minería de datos. Los modelos de minería de datos se pueden aplicar en escenarios como los siguientes:

1. Previsión: calcular las ventas y predecir las cargas de servidor o el tiempo de inactividad del servidor.
2. Riesgo y probabilidad: elegir los mejores clientes para la distribución de correo directo, determinar el punto de equilibrio probable para los escenarios de riesgo, y asignar probabilidades a diagnósticos u otros resultados.
3. Recomendaciones: determinar los productos que se pueden vender juntos y generar recomendaciones.
4. Buscar secuencias: analizar los artículos que los clientes han introducido en el carrito de compra y predecir los posibles eventos.
5. Agrupación: separar los clientes o los eventos en clústeres de elementos relacionados, y analizar y predecir afinidades.

La generación de un modelo de minería de datos forma parte de un proceso mayor que incluye desde la formulación de preguntas acerca de los datos y la creación de un modelo para responderlas, hasta la implementación del modelo en un entorno de trabajo. Este proceso se puede definir mediante los seis pasos básicos siguientes:

1. Definir el Problema: El primer paso del proceso de minería de datos, tal como se resalta en el siguiente diagrama, consiste en definir claramente el problema y considerar formas de usar los datos para proporcionar una respuesta para el mismo.

Fuente: https://msdn.microsoft.com/es-es/library/ms174949(v=sql.120).aspx#DefiningTheProblem

Este paso incluye analizar los requisitos empresariales, definir el ámbito del problema, definir las métricas por las que se evaluará el modelo y definir los objetivos concretos del proyecto de minería de datos. Estas tareas se traducen en preguntas como las siguientes:
¿Qué está buscando? ¿Qué tipos de relaciones intenta buscar?
¿Refleja el problema que está intentando resolver las directivas o procesos de la empresa?
¿Desea realizar predicciones a partir del modelo de minería de datos o solamente buscar asociaciones y patrones interesantes?
¿Qué resultado o atributo desea predecir?

¿Qué tipo de datos tiene y qué tipo de información hay en cada columna? En caso de que haya varias tablas, ¿cómo se relacionan?¿Necesita limpiar, agregar o procesar los datos antes de poder usarlos?
¿Cómo se distribuyen los datos?¿Los datos son estacionales?¿Los datos representan con precisión los procesos de la empresa?

Para responder a estas preguntas, puede que deba dirigir un estudio de disponibilidad de datos para investigar las necesidades de los usuarios de la empresa con respecto a los datos disponibles. Si los datos no abarcan las necesidades de los usuarios, podría tener que volver a definir el proyecto.
También debe considerar las maneras en las que los resultados del modelo se pueden incorporar en los indicadores de rendimiento clave (KPI) que se utilizan para medir el progreso comercial.

2. Preparar los datos: El segundo paso del proceso de minería de datos, como se indica en el siguiente diagrama, consiste en consolidar y limpiar los datos identificados en el paso Definir el problema.

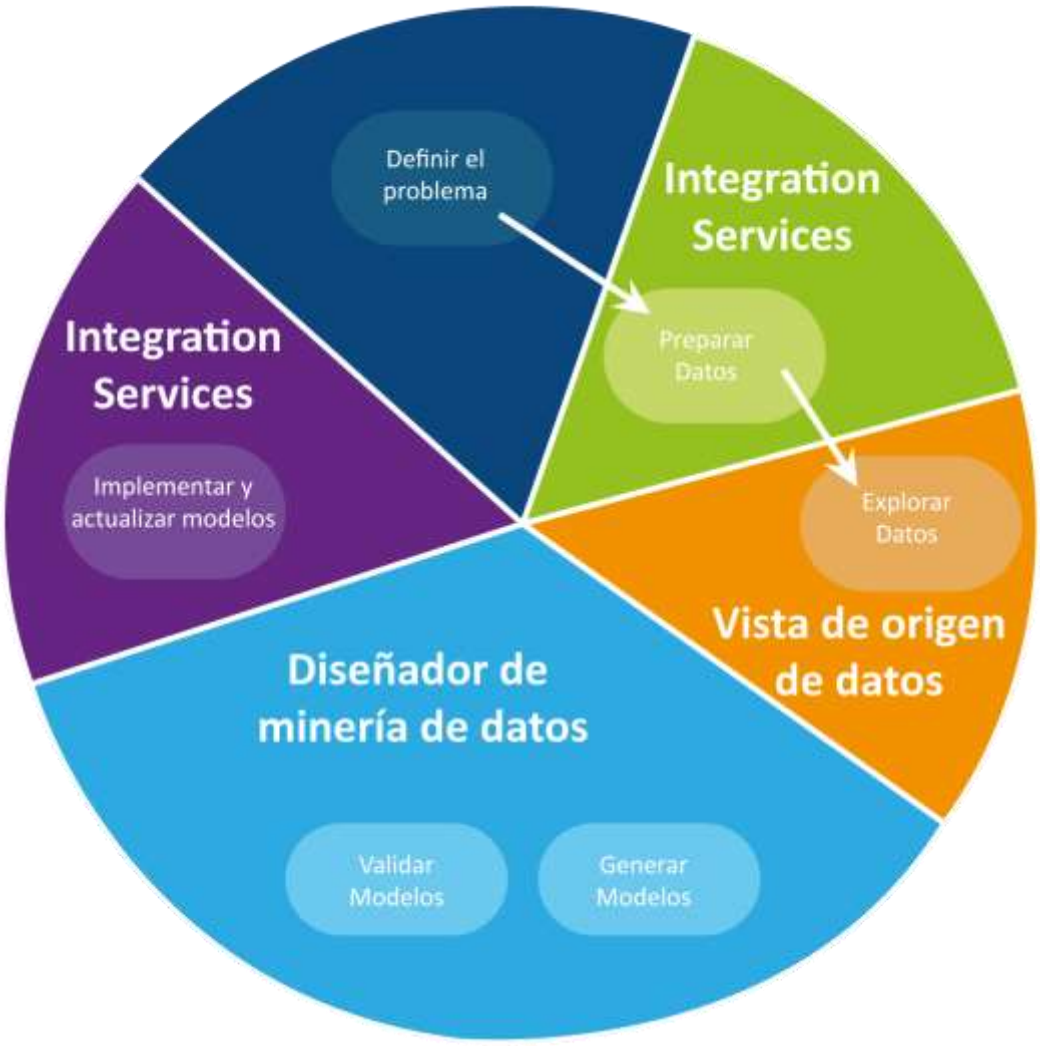
Fuente: https://msdn.microsoft.com/es-es/library/ms174949(v=sql.120).aspx#PreparingData

Los datos pueden estar dispersos en la empresa y almacenados en formatos distintos; también pueden contener incoherencias como entradas que faltan o incorrectas. Por ejemplo, los datos pueden mostrar que un cliente adquirió un producto incluso antes que se ofreciera en el mercado o que el cliente compra regularmente en una tienda situada a 2.000 kilómetros de su casa.
La limpieza de datos no solamente implica quitar los datos no válidos o interpolar valores que faltan, sino también buscar las correlaciones ocultas en los datos, identificar los orígenes de datos que son más precisos y determinar qué columnas son las más adecuadas para el análisis. Por ejemplo, ¿debería utilizar la fecha de envío o la fecha de pedido?¿Qué influye más en las ventas: la cantidad, el precio total o un precio con descuento? Los datos incompletos, los datos incorrectos y las entradas que parecen independientes, pero que de hecho están estrechamente correlacionadas, pueden influir en los resultados del modelo de maneras que no espera.

Por consiguiente, antes de empezar a generar los modelos de minería de datos, debería identificar estos problemas y determinar cómo los corregirá. En la minería de datos, por lo general se trabaja con un conjunto de datos de gran tamaño y no se puede examinar la calidad de los datos de cada transacción; por tanto, es posible que necesite usar herramientas de generación de perfiles de datos, y de limpieza y filtrado automático de datos, como las que se proporcionan en Integration Services, Microsoft SQL Server 2012 Master Data Services o SQL Server Data Quality Services para explorar los datos y buscar incoherencias.

Es importante tener en cuenta que los datos que se usan para la minería de datos no necesitan almacenarse en un cubo de procesamiento analítico en línea (OLAP), ni siquiera en una base de datos relacional, aunque puede usar ambos como orígenes de datos.Puede realizar minería de datos mediante cualquier origen de datos definido como origen de datos de Analysis Services.Por ejemplo, archivos de texto, libros de Excel o datos de otros proveedores externos.

3. Explorar los datos: El tercer paso del proceso de minería de datos, como se resalta en el siguiente diagrama, consiste en explorar los datos preparados.



Debe conocer los datos para tomar las decisiones adecuadas al crear los modelos de minería de datos. Entre las técnicas de exploración se incluyen calcular los valores mínimos y máximos, calcular la media y las desviaciones estándar, y examinar la distribución de los datos. Por ejemplo, al revisar el máximo, el mínimo y los valores de la media se podrían determinar que los datos no son representativos de los clientes o procesos de negocio, y que por consiguiente debe obtener más datos equilibrados o revisar las suposiciones que son la base de sus expectativas. Las desviaciones estándar y otros valores de distribución pueden proporcionar información útil sobre la estabilidad y exactitud de los resultados. Una desviación estándar grande puede indicar que agregar más datos podría ayudarle a mejorar el modelo. Los datos que se desvían mucho de una distribución estándar se podrían sesgar o podrían representar una imagen precisa de un problema de la vida real, pero dificultan el ajustar un modelo a los datos.

Al explorar los datos para conocer el problema empresarial, puede decidir si el conjunto de datos contiene datos defectuosos y, a continuación, puede inventar una estrategia para corregir los problemas u obtener una descripción más profunda de los comportamientos que son típicos de su negocio.

Puede usar herramientas como Master Data Services para sondear los orígenes de datos disponibles y determinar su disponibilidad para la minería de datos.Puede usar herramientas como SQL Server Data Quality Services, o el generador de perfiles de datos de Integration Services, para analizar la distribución de los datos y solucionar problemas, como la existencia de datos incorrectos o la falta de datos.
Cuando tenga definido los orígenes, combínelos en una vista del origen de datos con el Diseñador de vistas del origen de datos de SQL Server Data Tools. Este diseñador también contiene algunas herramientas que podrá usar para explorar los datos y comprobar que funcionarán a la hora de crear un modelo.

Tenga en cuenta que cuando se crea un modelo, Analysis Services crea automáticamente resúmenes estadísticos de los datos contenidos en él, que puede consultar para su uso en informes o análisis.

4. Generar Modelos: El cuarto paso del proceso de minería de datos, como se resalta en el siguiente diagrama, consiste en generar el modelo o modelos de minería de datos. Usará los conocimientos adquiridos en el paso Explorar los datos para definir y crear los modelos.

Fuente: https://msdn.microsoft.com/es-es/library/ms174949(v=sql.120).aspx#BuildingModels

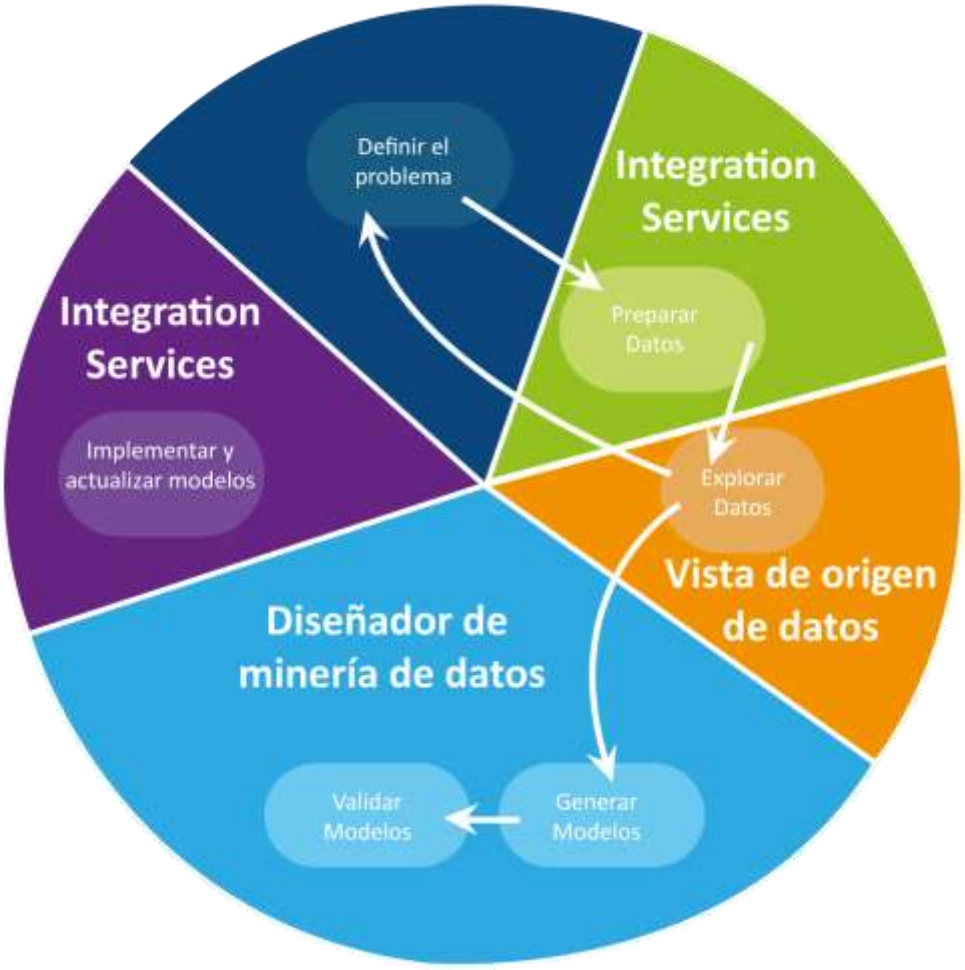
Deberá definir qué columnas de datos desea que se usen; para ello, creará una estructura de minería de datos. La estructura de minería de datos se vincula al origen de datos, pero en realidad no contiene ningún dato hasta que se procesa. Al procesar la estructura de minería de datos, Analysis Services genera agregados y otra información estadística que se puede usar para el análisis. Cualquier modelo de minería de datos que esté basado en la estructura puede utilizar esta información.

Antes de procesar la estructura y el modelo, un modelo de minería de datos simplemente es un contenedor que especifica las columnas que se usan para la entrada, el atributo que está prediciendo y parámetros que indican al algoritmo cómo procesar los datos. El procesamiento de un modelo a menudo se denomina entrenamiento. El entrenamiento hace referencia al proceso de aplicar un algoritmo matemático concreto a los datos de la estructura para extraer patrones. Los patrones que encuentre en el proceso de entrenamiento dependerán de la selección de los datos de entrenamiento, el algoritmo que elija y cómo se haya configurado el algoritmo. SQL Server 2014 contiene muchos algoritmos diferentes. Cada uno está preparado para un tipo diferente de tarea y crea un tipo distinto de modelo.

También puede utilizar los parámetros para ajustar cada algoritmo y puede aplicar filtros a los datos de entrenamiento para utilizar un subconjunto de los datos, creando resultados diferentes. Después de pasar los datos a través del modelo, el objeto de modelo de minería de datos contiene los resúmenes y modelos que se pueden consultar o utilizar para la predicción.
Puede definir un modelo nuevo mediante el Asistente para minería de datos de SQL Server Data Tools o con el lenguaje DMX (Extensiones de minería de datos).

Es importante recordar que siempre que los datos cambian, debe actualizar la estructura y el modelo de minería de datos. Al actualizar una estructura de minería de datos volviéndola a procesar, Analysis Services recupera los datos del origen, incluido cualquier dato nuevo si el origen se actualiza dinámicamente, y vuelve a rellenar la estructura de minería de datos. Si tiene modelos que están basados en la estructura, puede elegir actualizar estos, lo que significa que se vuelven a entrenar con los nuevos datos, o pueden dejar los modelos tal cual.

5. Explorar y Validad los modelos: El quinto paso del proceso de minería de datos, como se resalta en el siguiente diagrama, consiste en explorar los modelos de minería de datos que ha generado y comprobar su eficacia.



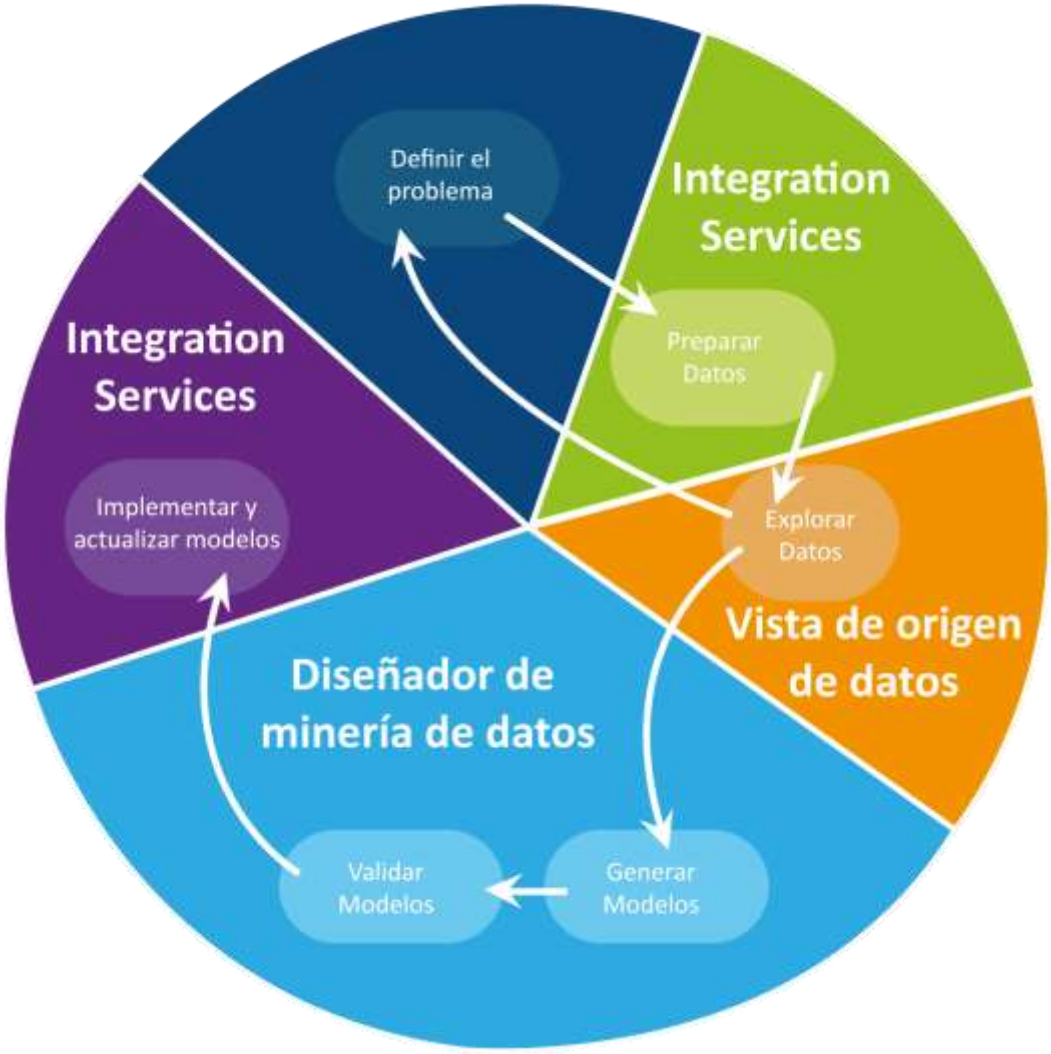
Fuente: https://msdn.microsoft.com/es-es/library/ms174949(v=sql.120).aspx#ValidatingModels

Antes de implementar un modelo en un entorno de producción, es aconsejable probar si funciona correctamente. Además, al generar un modelo, normalmente se crean varios con configuraciones diferentes y se prueban todos para ver cuál ofrece los resultados mejores para su problema y sus datos.
Analysis Services proporciona herramientas que ayudan a separar los datos en conjuntos de datos de entrenamiento y pruebas, para que pueda evaluar con precisión el rendimiento de todos los modelos en los mismos datos. El conjunto de datos de entrenamiento se utiliza para generar el modelo y el conjunto de datos de prueba para comprobar la precisión del modelo mediante la creación de consultas de predicción. En SQL Server 2014 Analysis Services (SSAS), estas particiones se pueden hacer automáticamente mientras se genera el modelo de minería de datos.

Puede explorar las tendencias y patrones que los algoritmos detectan mediante los visores del diseñador de minería de datos de SQL Server Data Tools. También puede comprobar si los modelos crean predicciones correctamente mediante herramientas del diseñador como el gráfico de mejora respecto al modelo predictivo y la matriz de clasificación. Para comprobar si el modelo es específico de sus datos o se puede utilizar para realizar inferencias en la población general, puede utilizar la técnica estadística denominada *validación cruzada* para crear automáticamente subconjuntos de los datos y probar el modelo con cada uno.

Si ninguno de los modelos que ha creado en el paso Generar modelos funciona correctamente, puede que deba volver a un paso anterior del proceso y volver a definir el problema o volver a investigar los datos del conjunto de datos original.

6. Implementar y Actualizar los modelos: El último paso del proceso de minería de datos, como se resalta en el siguiente diagrama, consiste en implementar los modelos que funcionan mejor en un entorno de producción.



Fuente: https://msdn.microsoft.com/es-es/library/ms174949(v=sql.120).aspx#DeployingandUpdatingModels

Una vez que los modelos de minería de datos se encuentran en el entorno de producción, puede llevar acabo diferentes tareas, dependiendo de sus necesidades. Las siguientes son algunas de las tareas que puede realizar:

1. Use los modelos para crear predicciones que luego podrá usar para tomar decisiones comerciales. SQL Server pone a su disposición el lenguaje DMX, que podrá usar para crear consultas de predicción, y el Generador de consultas de predicción, que le ayudará a generar las consultas.
2. Crear consultas de contenido para recuperar estadísticas, reglas o fórmulas del modelo.
3. Incrustar la funcionalidad de minería de datos directamente en una aplicación. Puede incluir Objetos de administración de análisis (AMO), que contiene un conjunto de objetos que la aplicación pueda utilizar para crear, cambiar, procesar y eliminar estructuras y modelos de minería de datos. También puede enviar mensajes XML for Analysis (XMLA) directamente a una instancia de Analysis Services.
4. Utilizar Integration Services para crear un paquete en el que se utilice un modelo de minería de datos para dividir de forma inteligente los datos entrantes en varias tablas. Por ejemplo, si una base de datos se actualiza continuamente con clientes potenciales, puede utilizar un modelo de minería de datos junto con Integration Services para dividir los datos entrantes en clientes que probablemente compren un producto y clientes que probablemente no compren un producto.
5. Crear un informe que permita a los usuarios realizar consultas directamente en un modelo de minería de datos existente.
6. Actualizar los modelos después de la revisión y análisis. Cualquier actualización requiere que vuelve a procesar los modelos.
7. Actualizar dinámicamente los modelos, cuando entren más datos en la organización, y realizar modificaciones constantes para mejorar la efectividad de la solución debería ser parte de la estrategia de implementación.

Soluciones de Minería de datos: Una solución de minería de datos es una solución de Analysis Services que contiene uno o más proyectos de minería de datos.

Una solución de minería de datos se puede basar en datos multidimensionales, es decir, en un cubo existente; o en datos puramente relacionales, como las tablas y las vistas de un almacenamiento de datos; o bien en archivos de texto, libros de Excel u otros orígenes de datos externos.

1. Puede crear objetos de minería de datos en una solución de base de datos multidimensional existente. Normalmente, se crearía una solución como esta si ya ha creado un cubo y desearía realizar la minería de datos utilizando el cubo como origen de datos. Al mover y crear copias de seguridad de modelos basados en un cubo, el cubo también debe moverse o copiarse.

2. Puede crear una solución de minería de datos que solo contenga objetos de minería de datos, incluidos los orígenes de datos y vistas del origen de datos que admiten, y que use un origen de datos relacional solamente.

Este es el método preferido para crear modelos de minería de datos, dado que el procesamiento y la consulta normalmente es más rápido en orígenes de datos relacionales. También puede mover y hacer copia de seguridad fácilmente de los modelos entre servidores copiando los comandos EXPORT e IMPORT.

Algoritmos de Minería de Datos

Algoritmo de minería de datos es un conjunto de cálculos y reglas heurísticas que permite crear un modelo de minería de datos a partir de los datos. Para crear un modelo, el algoritmo analiza primero los datos proporcionados, en busca de tipos específicos de patrones o tendencias. El algoritmo usa los resultados de este análisis para definir los parámetros óptimos para la creación del modelo de minería de datos. A continuación, estos parámetros se aplican en todo el conjunto de datos para extraer patrones procesables y estadísticas detalladas.

El modelo de minería de datos que crea un algoritmo a partir de los datos puede tomar diversas formas, incluyendo:

Un conjunto de clústeres que describe cómo se relacionan los casos de un conjunto de datos.
Un árbol de decisión que predice un resultado y que describe cómo afectan a este los distintos criterios.
Un modelo matemático que predice las ventas.
Un conjunto de reglas que describen cómo se agrupan los productos en una transacción, y las probabilidades de que dichos productos se adquieran juntos.

Microsoft SQL Server Analysis Services proporciona varios algoritmos que puede usar en las soluciones de minería de datos. Estos algoritmos son implementaciones de algunas de las metodologías más conocidas usadas en la minería de datos. Todos los algoritmos de minería de datos de Microsoft se pueden personalizar y son totalmente programables, bien mediante las API proporcionadas o bien mediante los componentes de minería de datos de SQL Server Integration Services.

También puede usar algoritmos de minería de datos desarrollados por terceros que cumplan la especificación OLE DB para minería de datos, o desarrollar algoritmos personalizados que se pueden registrar como servicios para usarlos a continuación en el marco de la minería de datos de SQL Server.

Elegir el algoritmo correcto

La elección del mejor algoritmo para una tarea analítica específica puede ser un desafío.Aunque puede usar diferentes algoritmos para realizar la misma tarea, cada uno de ellos genera un resultado diferente, y algunos pueden generar más de un tipo de resultado. Por ejemplo, puede usar el algoritmo Árboles de decisión de Microsoft no solo para la predicción, sino también como una forma de reducir el número de columnas de un conjunto de datos, ya que el árbol de decisión puede identificar las columnas que no afectan al modelo de minería de datos final.

Analysis Services incluye los siguientes tipos de algoritmos:

ALGORITMO	CARACTERISTICA
Algoritmos de clasificación	Que predicen una o más variables discretas, basándose en otros atributos del conjunto de datos.

Algoritmos de regresión	Que predicen una o más variables continuas, como las pérdidas o los beneficios, basándose en otros atributos del conjunto de datos.
Algoritmos de segmentación	Que dividen los datos en grupos, o clústeres, de elementos que tienen propiedades similares
Algoritmos de asociación	Que buscan correlaciones entre diferentes atributos de un conjunto de datos. La aplicación más común de esta clase de algoritmo es la creación de reglas de asociación, que pueden usarse en un análisis de la cesta de compra.
Algoritmos de análisis de secuencias	Que resumen secuencias o episodios frecuentes en los datos, como un flujo de rutas web.

Sin embargo, no hay ninguna razón por la que deba limitarse a un algoritmo en sus soluciones. Los analistas experimentados usarán a veces un algoritmo para determinar las entradas más eficaces (es decir, variables) y luego aplicarán un algoritmo diferente para predecir un resultado concreto basado en esos datos.

La minería de datos de SQL Server le permite generar varios modelos en una única estructura de minería de datos, por lo que en una solución de minería de datos puede usar un algoritmo de clústeres, un modelo de árboles de decisión y un modelo de Bayes naïve para obtener distintas vistas de los datos.

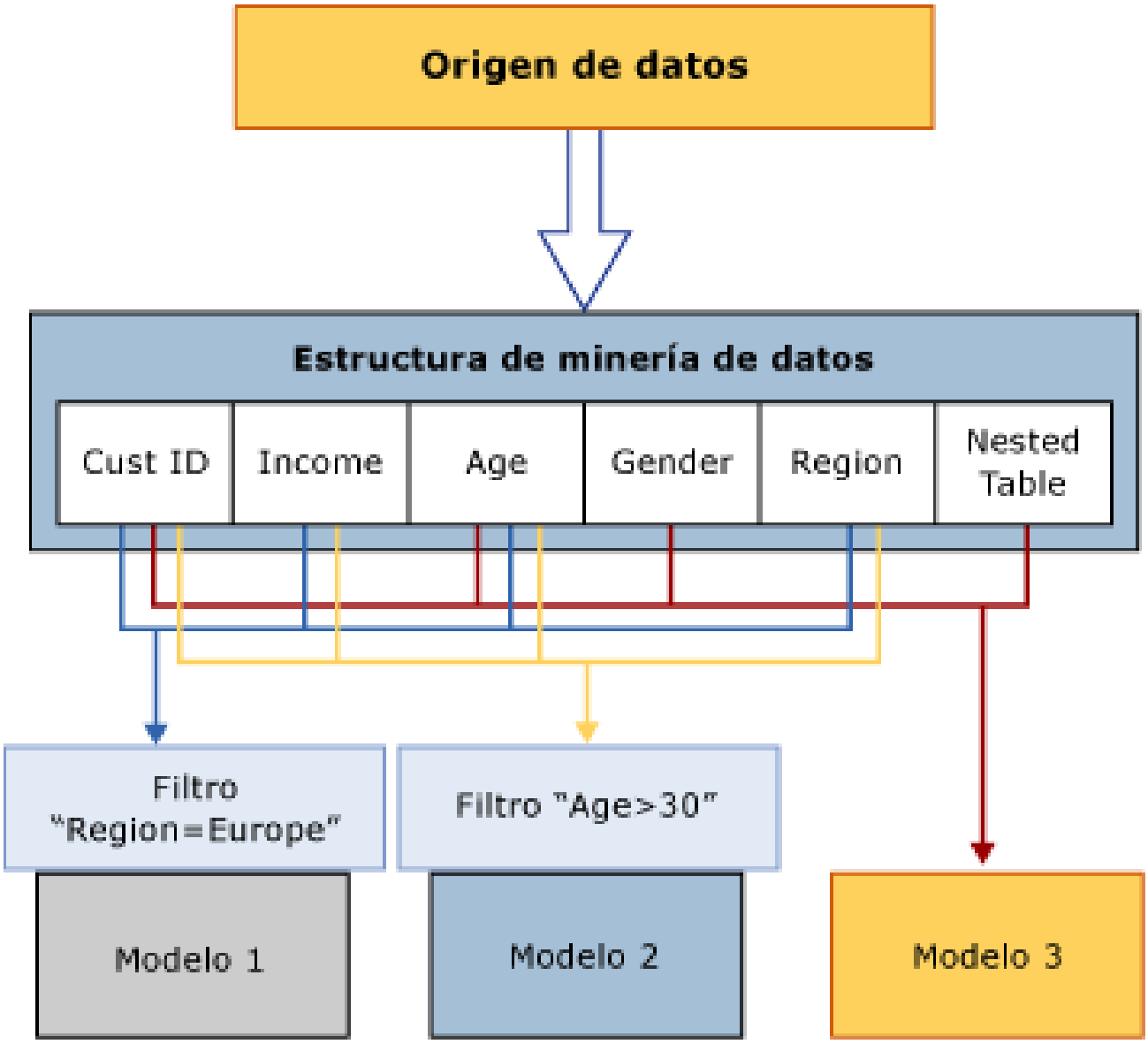
También puede usar varios algoritmos dentro de una única solución para realizar tareas independientes: por ejemplo, podría usar la regresión para obtener predicciones financieras, y un algoritmo de red neuronal para realizar un análisis de los factores que influyen en las ventas.

Para dar más claridad en este tema se sugiere revisar el siguiente link:

Enlace: https://msdn.microsoft.com/es-es/library/ms175595(v=sql.120).aspx

ESTRUCTURA DE MINERÍA DE DATOS

La estructura de minería de datos define los datos a partir de los cuales se generan los modelos de minería de datos: especifica la vista de datos de origen, el número y el tipo de columnas, y una partición opcional en conjuntos de entrenamiento y de pruebas. Una misma estructura de minería de datos puede admitir varios modelos de minería de datos que comparten el mismo dominio. En el diagrama siguiente, se muestra la relación de la estructura de minería de datos con el origen de datos y con los modelos de minería de datos que la componen.



Fuente: https://msdn.microsoft.com/es-es/library/ms174757(v=sql.120).aspx

La estructura de minería de datos del diagrama está basada en un origen de datos que contiene varias tablas o vistas, combinadas en el campo CustomerID. Una tabla contiene información sobre los clientes, como la región geográfica, la edad, los ingresos y el sexo, mientras que la tabla anidada relacionada contiene varias filas de información adicional sobre cada cliente, como los productos que ha adquirido. En el diagrama, se muestra que se pueden generar varios modelos de minería de datos a partir de una misma estructura de minería de datos, y que los modelos pueden usar columnas de la estructura diferentes.

Modelo 1: usa CustomerID, Income, Age, Region y filtra los datos de Region.
Modelo 2: usa CustomerID, Income, Age, Region y filtra los datos de Age.
Modelo 3: usa CustomerID, Age, Gender y la tabla anidada, sin filtros.

Dado que los modelos usan columnas diferentes para la entrada, y dado que dos de los modelos, además, restringen sus datos mediante la aplicación de un filtro, los modelos pueden tener resultados muy diferentes aunque estén basados en los mismos datos. Observe que la columna CustomerID es obligatoria en todos los modelos porque es la única columna disponible que se puede usar como clave de caso.

3.4.3 TALLER DE ENTRENAMIENTO

MINERIA DE DATOS

El siguiente taller de entrenamiento se propone para validar la aprehensión de los conceptos. Se debe leer detenidamente las preguntas propuestas y dar respuesta a ellas.

1. Defina Minería de Datos
2. Enuncie los algoritmos de Minería de Datos
3. Defina la Estructura de Minería de Datos
4. Enuncie los modelos de Minería de Datos

3. PISTAS DE APRENDIZAJE

Recuerde: Las bases de datos tradicionales, son llamadas también pasivas.

Tenga en Cuenta: Cualquier modificación sobre el comportamiento reactivo se puede llevar a cabo cambiando solamente las reglas activas, sin necesidad de variar las aplicaciones.

Tener Presente: Para definir Reglas Deductivas que permiten concluir, inferir u obtener información nueva a partir de los datos almacenados o sucesos condicionados.

Recuerde: Una base de datos multimedia es un conjunto de información combinada, la cual puede ser texto, arte gráfico, sonido, animación y vídeo.

Tenga en Cuenta: El Almacén de Datos permite recopilar la información de una forma confiable, segura y de calidad.

Tenga Presente: Las bases de datos distribuidas son un grupo de información que corresponde a un sistema que se encuentra repartido entre computadores de una misma red.

Recuerde: Una red de comunicación suministra las capacidades para que un proceso en ejecución en un sitio de la red envíe y reciba mensajes de otro que se encuentra ejecutándose en un sitio distinto.

Acuérdese: En un sistema federado los usuarios tienen acceso a la información, de los distintos sistemas, a través de una interfaz común.

Tenga en Cuenta: Las propiedades de una base de datos federada son: heterogeneidad, autonomía y distribución.

Recuerde: Gracias a esta computación, los usuarios logran acceder a una base de datos remota en cualquier momento y en cualquier lugar.

Tener Presente: Una base de datos paralela es una tecnología innovadora que maneja ordenadamente todo tipo de recursos, entre ellos equipos de cómputo, almacenamiento y aplicaciones definidas.

4. GLOSARIO

1. **Bases de datos Orientada a Objetos**: Base de datos que tiene Un modelo de datos está orientado a objetos y almacenan y recuperan objetos en los que se almacena estado y comportamiento.

2. **Bases de datos Distribuidas**: Son un grupo de datos que pertenecen a un sistema pero a su vez está repartido entre ordenadores de una misma red.

3. **Bases de Datos Relacional**: tipo de base de datos (BD) que cumple con el modelo relacional (el modelo más utilizado actualmente para implementar las BD ya planificadas).
4. **Bases de Datos Activas:** Un sistema de bases de datos activas es un sistema de gestión de bases de datos (SGBD) que contiene un subsistema que permite la definición y la gestión de reglas de producción (reglas activas).

5. **Big Data**: información o grupo de datos que por su elevado volumen, diversidad y complejidad no pueden ser almacenados ni visualizados con herramientas tradicionales.

6. **Clase:** Plantilla implementada en software que describe un conjunto de objetos con atributos y comportamiento similares.

7. **Datawarehouse** es una base de datos corporativa que se caracteriza por integrar y depurar información de una o más fuentes distintas, para luego procesarla permitiendo su análisis desde infinidad de pespectivas y con grandes velocidades de respuesta.

8. **Datamart** es una base de datos departamental, especializada en el almacenamiento de los datos de un área de negocio específica.
9. **Datos:** Conjunto de símbolos que representan una determinada información.
10. **Distribución:** Se denomina distribución al reparto de uno o varios elementos.

11. **Evento:** Un evento es una variante de las propiedades para los campos cuyos tipos sean delegados. Es decir, permiten controlar la forman en que se accede a los campos delegados y dan la posibilidad de asociar código a ejecutar cada vez que se añada o elimine un método de un campo delegado

12. Sistema Gestor de Bases de Datos Orientadas a Objetos (SGBDOO): El gestor de una base de datos orientada a objetos.

5. BIBLIOGRAFÍA

Sistemas de bases de datos orientadas a objetos: Conceptos y arquitecturas. Editorial: Addison-Wesley / Diaz de Santos. Autores: Elisa Bertino, Lorenzo Martino.

Sistemas de bases de datos: Un enfoque práctico para diseño, implementación y gestión. 4ª Edición. Editorial: Pearson Addison- Wesley. Autores: Thomas M. Connolly, Carolyn E. Begg.

Fundamentos de Bases de datos. 5ª Edición. Editorial: McGraw Hill. Autores: Silberschatz, Korth, Sudarshan
Bases de Datos Orientadas a Objeto y el estándar ODMG. Autores: Clara Martín Sastre y Enrique Medarde Caballero.

L.Mota Herranz y M. Celma Giménez - Métodos para la comprobación de la integridad de en bases de datos deductivas

González Alvarado, Carlos. Sistema de Bases de Datos. Editorial Tecnológica de Costa Rica, Primera Edición, 1996.

Elmasri, Ramez. Sistemas de Bases de Datos. Editorial Addison –Wesley Iberoamericana S-A. Segunda Edición, 1997.
I.C. Silvia Eloisa Carlín Salgado y M.Sc. Rosendo Moreno Rodríguez - Valorización de las bases de datos deductivas y de las bases de datos activas

http://gpd.sip.ucm.es/ Rafael Caballero Roldán. Introducción a las bases de datos deductivas
P. Julián Iranzo. Apuntes de Programación Declarativa, 2002

http://sistemas.itlp.edu.mx/revistadsyc Marco Antonio Castro Liera - Bases de Datos Relacionales Difusas
Grosky, William I. “Managing Multimedia Information in Database Systems”, University of Detroit, 1997

Connolly T., Begg C., “Sistemas de b ases de datos - Un enfoque practica para diseño, implementación y gestión”. Ed PearsonAddison-Wesley.
Rob P., Coronel C., “Sistemas de bases de datos - Diseño Implementacion y Administracion”. Ed. Thomson.
Atzeni P.,Stefano C., “Database Systems - Concepts, Languages and Architectures”. Ed. McGraw Hill.

Introducción a la Documática http://tramullas.com/documatica/indice.html, Jesús Tramullas y Kronos © 1997, 2000.

“Tecnología y Diseño de Bases de Datos”, PIATTINI VELTHUIS, MARIO G / MARCOS MARTINEZ, ESPERANZA / CALERO MUÑOZ, CORAL / VELA SÁNCHEZ, BELÉN

“Sistemas de Bases de Datos. Un enfoque práctico para diseño, implementación y gestión”, THOMAS M. CONNOLLY/ CAROLYN E. BEGG

Oracle9i Data Warehousing Guide Release 2 (9.2):

DATABASES AND THE GRID. Watson, P. University of Newcastle [2001] WHAT IS THE GRID? A THREE POINT CHECKLIST. Foster, I., University of Chicago [2002]

ORACLE DATABASE 10G: THE DATABASE FOR THE GRID. An Oracle White Paper.Oracle [2003]