INDICE DE CONTENIDOS

1. FICHEROS

Los ficheros son estructuras de información que crean los sistemas operativos de los ordenadores para poder almacenar datos. Suelen tener un nombre (descriptor) y una extensión, que determina el formato de la información que contiene.

Un ordenador puedo almacenar muchos tipos de información como datos administrativos y contables, datos bancarios, música, vídeos, páginas webs, etc. Esta información se almacena en ficheros en dispositivos de almacenamiento como son discos duros, pen drives, et

1.1.Tipos de ficheros y formatos

El formato y el tipo determina la forma de interpretar la información que contiene. Todos los ficheros se almacenan como una ristra de bits (ceros y unos) con lo que es necesaria su interpretación para dar sentido a la información que almacena.

Prueba a abrir un fichero con extensión .doc con el bloc de notas (botón derecho → abrir con). El bloc de notas no reconoce el formato del fichero .doc por tanto no interpreta correctamente el contenido del fichero.

El sistema operativo trata un fichero atendiendo a dos criterios:

a) Según el contenido: de texto o datos binarios.
b) Según su tipo: imágenes, ejecutables, videos, etc.

1.1.1. Ficheros de texto

Se denominan ficheros planos o fichero ASCII (American Standard Code for Information Interchange). Es un estándar que asigna un valor numérico a cada carácter, con lo que se pueden representar los documentos de Texto Plano, es decir, los que son directamente legibles por seres humanos.

Prueba a buscar en Internet la tabla de códigos ascii de 8 bits y comprueba que se dan las siguientes características:

- Los 32 primeros caracteres, se llaman caracteres no imprimibles y se utilizaban tradicionalmente para el control de transmisiones.
- La distancia entre mayúsculas y minúsculas es exactamente 32 caracteres.
- Hay caracteres que son numéricos, y cuyo valor ascii es el resultado de sumarle 48. Por ejemplo, 6+48=54. 54 es el código ascii del carácter '6'.

Los ficheros de texto, aunque no necesitan un formato para ser interpretado, suelen tener extensiones para conocer qué tipo de texto se halla dentro del fichero, por ejemplo:

- **Ficheros de configuración**: Son ficheros cuyo contenido es texto sobre configuraciones del sistema operativo o de alguna aplicación (extensión .ini, .inf , .conf…).
- **Ficheros de código fuente**: Su contenido es texto con programas informático (.sql, .c, .java…).
- **Ficheros de páginas web**: Las páginas web son ficheros de texto con hipertexto que interpreta el navegador, .html, .php, .css, .xml…
- **Formatos enriquecidos**: Son textos que contienen códigos de control para ofrecer una visión del texto más elegante: .rtf, .ps , .tex

1.1.2. Ficheros Binarios

Los ficheros binarios son todos los que no son de texto, y requieren un formato para ser interpretado:

- **De imagen**: .jpg, .gif, .tiff, .bmp, .wmf, .png, .pcx; entre muchos otros
- **De vídeo**: .mpg, .mov, .avi, .qt
- **Comprimidos o empaquetados**: .zip, .Z, .gz, .tar, .lhz
- **Ejecutables o compilados**: .exe, .com, .cgi, .o, .a

2. BASES DE DATOS. CONCEPTOS

2.1.Introducción

Antes de entrar en una definición formal de lo que es una base de datos, vamos a ver con un ejemplo la forma tradicional de gestionar y almacenar unos datos, y las novedades que incorpora una base de datos.

Supongamos una pequeña empresa que se dedica a realizar ventas por correo. Ésta necesitará gestionar un mantenimiento de dientes, un control del almacén y un sistema de facturas. La forma tradicional de resolver este problema consiste en utilizar una serie de aplicaciones (programas) en los que cada aplicación dispone de su propio conjunto de ficheros, que contienen los datos necesarios, y que están organizados de acuerdo a la forma que tiene la aplicación de tratarlos. Estas organizaciones de los ficheros serían las ya conocidas: secuencial, directa o indexada.

Si se decide cambiar la estructura de alguno de estos ficheros, será necesario también cambiar la propia aplicación. A la inversa, si se tiene que cambiar la aplicación, casi con toda seguridad habrá que cambiar el número de ficheros, su organización, tipo de campos, etc.

Además, y dado que las aplicaciones son para la misma empresa, es seguro que existirá un amplio grado de redundancia (repetición) de los datos entre los distintos ficheros de las aplicaciones, por ejemplo, datos de clientes repetidos en varios ficheros.

A causa de este tipo de trabajo basada en ficheros, se denomina **Sistema orientado al proceso.**

Si se reemplazan todos estos ficheros por una colección datos apropiada, generalmente de gran tamaño, que sea accesible por todas las aplicaciones, que se consistes y que presente una redundancia mínima, entonces la colección de datos se transforma en una Base de Datos (**BD**). En el caso de la empresa de venta por correo la **BD** estará constituida por una única, colección de datos usada por las aplicaciones de clientes, almacén y facturas.

2.2.Definición

Una Base de Datos es una colección interrelacionada de datos, almacenados en un conjunto sin redundancias innecesarias, cuya finalidad es la de servir a una o más aplicaciones de la manera más eficiente.

2.3.CONCEPTOS

- ✓ **Dato**: El dato es un trozo de información concreta sobre algún concepto o suceso. Por ejemplo, si de una persona hablamos: día, mes y año de nacimiento, nombre, dni, peso, altura, nivel de estudios…

- ✓ **Tipo de Dato**: El tipo de dato indica la naturaleza del campo. Así, se puede tener datos numéricos, que son aquellos con los que se pueden realizar cálculos aritméticos (sumas, restas, multiplicaciones...) y los datos alfanuméricos, que son los que contienen caracteres alfabéticos y dígitos numéricos

- ✓ **Campo o columna**: Un campo es un identificador para toda una familia de datos. Cada campo pertenece a un tipo de datos. Por ejemplo, el campo "Dni" representa los DNIs de las personas que hay en la tabla. Este campo será de tipo alfanumérico.

- ✓ **Registro**: Es una recolección de datos referentes a un mismo concepto o suceso. Por ejemplo, los datos de una persona pueden ser su NIF, año de nacimiento, su nombre, su dirección, etc. A los registros también se les llama tupias o filas.

- ✓ **Campo Clave**: Es un campo especial que identifica de forma única a cada registro. Así, el NIF es único para cada persona, por tanto es campo clave.

- ✓ **Tabla**: Es un conjunto de registros bajo un mismo nombre que representa el conjunto de todos ellos. Por ejemplo, todos los clientes de una base de datos se almacenan en una tabla cuyo nombre es Clientes.

- ✓ **Consulta o *query***: Es una instrucción para hacer peticiones a una base de datos. Puede ser una búsqueda simple de un registro especifico o ima solicitud para seleccionar todos los registros que satisfagan un conjunto de criterios. búsqueda de información, que devuelven los campos y registros solicitados. Ejemplo: listado de todos los nombres y dnis completos de la tabla Clientes o listado nombres y dnis de Clientes que vivan en Talavera.

- ✓ **índice**: Es una estructura que almacena los campos clave de una tabla, organizándolos para hacer más fácil encontrar y ordenar los registros de esa tabla (similar al índice de un libro).

- ✓ **Vista**: Es una transformación que se hace a una o más tablas para obtener una nueva tabla. Esta nueva tabla es una tabla virtual, es decir, no está almacenada en los dispositivos de almacenamiento del ordenador, aunque sí se almacena su definición.

- ✓ **Informe**: Es un listado ordenado de los campos y registros seleccionados en un formato fácil de leer. Generalmente se usan como peticiones expresas de un tipo de información por parte de un usuario. Por ejemplo, un informe de las facturas impagadas del mes de enero ordenado por nombre de cliente.

- ✓ **Guiones**: o scripts. Son un conjunto de instrucciones, que ejecutadas de forma ordenada, realizan operaciones avanzadas de mantenimiento de los datos almacenados en la base de datos.

- ✓ **Procedimientos**: Son un tipo especial de script que está almacenado en la base de datos y que forma parte de su esquema

2.4.Estructura de una Base de Datos

Una base de datos almacena los datos a través de un esquema. El ***esquema es la definición de la estructura*** donde se almacenan los datos, contiene todo lo necesario para organizar la información mediante tablas, registros (filas) y campos (columnas). También contiene otros objetos necesarios para el tratamiento de los datos (procedimientos, vistas, índices, etc.). Al esquema también se le suele llamar metainformación, es decir, información sobre la información o metadatos.

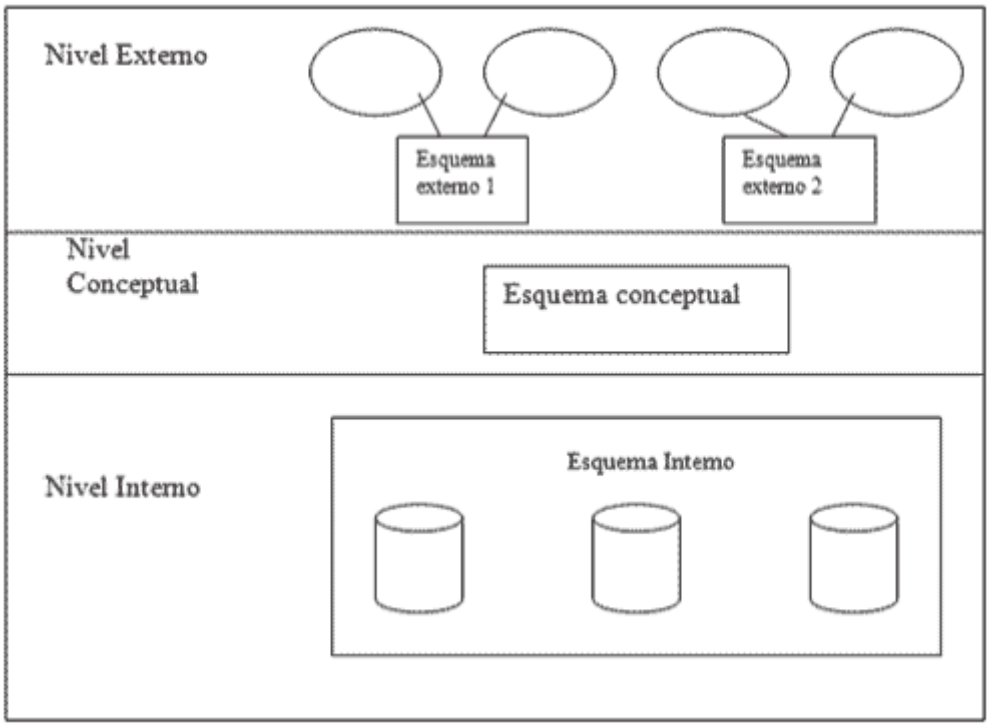
```
mysql> select table_schema, table_name, table_rows
    -> from information_schema.tables
    -> where table_schema='jardineria';
+--------------+---------------+------------+
| table_schema | table_name    | table_rows |
+--------------+---------------+------------+
| jardineria   | Clientes      |         36 |
| jardineria   | DetallePedidos|        295 |
| jardineria   | Empleados     |         32 |
| jardineria   | GamasProductos|          0 |
| jardineria   | Oficinas      |         10 |
| jardineria   | Pagos         |         26 |
| jardineria   | Pedidos       |        115 |
| jardineria   | Productos     |        276 |
+--------------+---------------+------------+
9 rows in set (0,01 sec)
```

Mysql almacena el esquema de la base de datos en tablas, de tal manera que el propio esquema de la base de datos se puede tratar como si fueran datos comunes de la base de datos.

2.4.1. Esquemas

Uno de los objetivos principales de un sistema de **BD** es, evitar a los usuarios los detalles relativos a la forma como los datos se almacenan y mantienen (abstracción).

Existen en el mercado varios paquetes de **SGBD** con diferentes arquitecturas una de ellas, la más estandarizada, es la que cumple los requerimientos de la normativa ANSI/X3/SPARC (Comité de Planificación y Requerimientos del Instituto Nacional Americano de estándares que a su vez depende del ISO), surgida en el año 1977, que en su división X3 establece que la arquitectura de una **BD** debe poseer tres niveles de abstracción.



2.4.1.1. Esquema interno

Este es el nivel más bajo de abstracción en el que se describe cómo se almacenan realmente los datos: tamaño de los bloques, posición relativa de los registros almacenados (ISAM, VSAM), métodos de direccionamiento, desbordamientos, índices, cambios en el almacenamiento, técnicas de compresión.

2.4.1.2. Esquema Conceptual

Es el siguiente nivel de abstracción. En él se describen cuáles son los datos reales que están almacenados en la base de datos y qué relaciones existen entre los datos: nombre, tamaño, tipo, relaciones, limitadores de integridad, etc. Este nivel y el anterior son utilizados sólo por el **DBA**.

2.4.1.3. Esquema Externo

Este es el nivel de abstracción más alto, en el cual se describe solamente una parte de la **BD**. Muchos usuarios no tienen por qué ocuparse de toda la información almacenada, pues necesitan solamente una parte. Para dar

adecuada respuesta a esta situación, se define para cada usuario que lo necesite una vista externa (o subesquema) de la **BD**. Por ejemplo, en un sistema bancario, las personas que tienen que preparar las nóminas sólo podrán ver la parte de la **BD** que contiene la información de los empleados, mientras que los cajeros sólo podrán tener acceso a la información sobre las cuentas.

2.5.Usos de las Bases de Datos

- ✓ **Bases de datos Administrativas**: Cualquier empresa necesita registrar y relacionar sus clientes, pedidos, facturas, productos, etc.
- ✓ **Bases de datos Contables**: También es necesario gestionar los pagos, balances de pérdidas y ganancias, patrimonio, declaraciones de hacienda...
- ✓ **Bases de datos para motores de búsquedas**: Por ejemplo, Google tienen una base de datos gigantesca donde almacenan información sobre todos los documentos de Internet.
- ✓ **Científicas**: Recolección de datos climáticos y medioambientales, químicos, genómicos, geológicos...
- ✓ **Configuraciones**: Almacenan datos de configuración de un sistema informático.
- ✓ **Censos**: Guardan información demográfica de pueblos, ciudades y países.
- ✓ **Virus**: Los antivirus guardan información sobre todos los potenciales software malicioso.
- ✓ **Otros muchos usos**: Militares, videojuegos, deportes, etc.

3. SISTEMAS GESTORES DE BASES DE DATOS

Un Sistema Gestor de Base de Datos, en adelante SGBD, como el conjunto de herramientas que facilitan la consulta, uso y actualización de una base de datos. Un ejemplo de software Gestor de Base de Datos que incorpora herramientas software que son capaces de estructurar en múltiples discos duros los ficheros de una base de datos, permitiendo el acceso a sus datos tanto a partir de herramientas gráficas como a partir de potentes lenguajes de programación como PLSQL, PHP, etc.

3.1.Funciones de un SGBD

1. Permiten a los usuarios almacenar datos, acceder a ellos y actualizarlos de forma sencilla y con un gran rendimiento, ocultando la complejidad y las características físicas de los dispositivos de almacenamiento.

2. Garantizan la integridad de los datos, respetando las reglas y restricciones que dicte el Programador de la base de datos. Es decir, no permiten operaciones que dejen cierto conjunto de datos incompletos o incorrectos.

3. Integran, junto con el sistema operativo, un sistema de seguridad que garantiza el acceso a la información exclusivamente a aquellos usuarios que dispongan de autorización.

4. Proporcionan un diccionario de metadatos, que contiene el esquema de la base de datos, es decir, cómo están estructurados los datos en tablas, registros y campos, las relaciones entre los datos, usuarios, permisos, etc. Este diccionario de datos debe ser también accesible de la misma forma sencilla que es posible acceder al resto de datos.

5. Permiten el uso de transacciones, garantizan que todas las operaciones de la transacción se realicen correctamente, y en caso de alguna incidencia, deshacen los cambios sin ningún tipo de complicación adicional.

6. Ofrecen, mediante completas herramientas, estadísticas sobre el uso del gestor, registrando operaciones efectuadas, consultas solicitas, operaciones fallidas y cualquier tipo de incidencia. Es posible de este modo, monitorizar el uso de la base de datos, y permiten analizar hipotéticos malfuncionamientos.

7. Permiten la concurrencia, es decir, varios usuarios trabajando sobre un mismo conjunto de datos. Además, proporcionan mecanismos que permiten arbitrar operaciones conflictivas en el acceso o modificación de un dato al mismo tiempo por parte de varios usuarios.

8. Independizan los datos de la aplicación o usuario que esté utilizándolos, haciendo más fácil su migración a otras plataformas.

9. Ofrecen conectividad con el exterior. De esta manera, se puede replicar y distribuir bases de datos. Además, todos los SGBD incorporan herramientas estándar de conectividad. El protocolo ODBC está muy extendido como forma de comunicación entre bases de datos y aplicaciones externas.

10. Incorporan herramientas para la salvaguarda y restauración de la información en caso de desastre. Algunos gestores, tienen sofisticados mecanismos para poder establecer el estado de una base de datos en cualquier punto anterior en el tiempo. Además, deben ofrecer sencillas herramientas para la importación y exportación automática de la información.

3.2.El Lenguage SQL

La principal herramienta de un gestor de base de datos es la interfaz de programación con el usuario. Este interfaz es un lenguaje muy sencillo denominado comúnmente SQL, Structured Query Language. Se divide en 4 sublenguajes que le permiten cumplir con las funcionalidades requeridas por CODD:

- ✓ **Lenguaje DML**: o lenguaje de manipulación de datos (Data Manipulation Language). Este lenguaje permite con 4 sentencias sencillas seleccionar determinados datos (SELECT), insertar datos (INSERT), modificarlos (UPDATE) o incluso borrarlos (DELETE).

- ✓ **Lenguaje DD**L: o lenguaje de definición de datos (Data Definition Language). Este lenguaje permite crear toda la estructura de una base de datos (desde tablas hasta usuarios). Sus cláusulas son del tipo DROP (Eliminar objetos) y CREATE (Crear objetos).

- ✓ **Lenguaje DCL**: o lenguaje de control de datos (Data Control Language). Incluye comandos (GRANT y REVOKE) que permiten al administrador gestionar el acceso a los datos contenidos en la base de datos.

- ✓ **Lenguaje TCL**: o lenguaje de control de transacciones. El propósito de este lenguaje es permitir ejecutar varios comandos de forma simultánea como si fuera un comando atómico o indivisible.

Si es posible ejecutar todos los comandos, se aplicala transacción (COMMIT), y si, en algún paso de la ejecución, sucede algo inesperado, se pueden deshacer todos los pasos dados (ROLLBACK).

4. MODELOS DE DATOS

Modelar consiste en definir un mundo abstracto y teórico tal que las conclusiones que se puedan sacar de él coinciden con las manifestaciones aparentes del mundo real.

Llamamos modelo al instrumento que se aplica a una parcela del mundo real (universo del discurso UD) para obtener una estructura de datos a la que denominamos esquema. El modelo será un conjunto de conceptos, reglas y convenciones que nos permiten describir los datos del UD.

A los valores que toman los distintos objetos de un esquema en un momento determinado los denominaremos instancia u ocurrencia del esquema.

Los modelos son la base para los lenguajes de datos. Por ejemplo, el, lenguaje SQL es el resultado de aplicar una determinada sintaxis al modelo relacional.

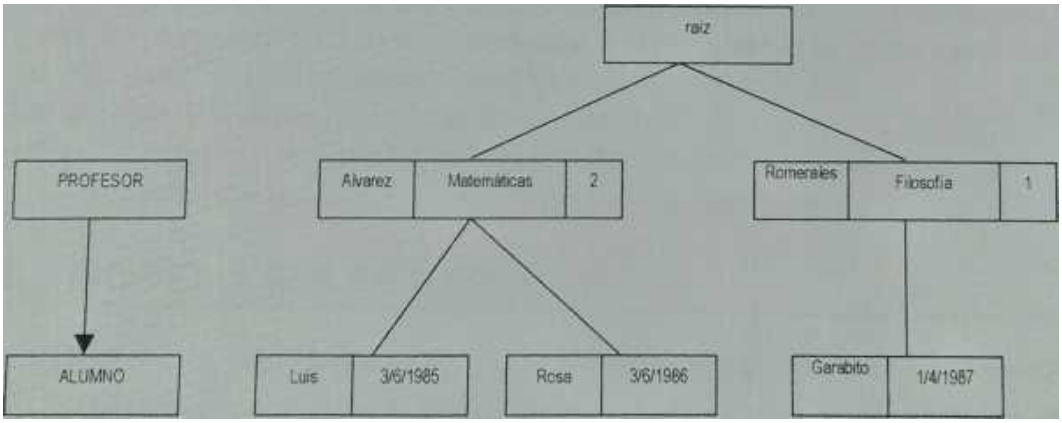
Los modelos han contado hasta hace poco con mayor grado de aceptación son tres:

- ✓ Modelo Jerárquico.
- ✓ Modelo en Red.
- ✓ Modelo Relacional.

No obstante, la complejidad creciente de las aplicaciones informáticas actuales está haciendo que el modelo de **BD** orientadas a objetos gane cada vez más adeptos y, por el contrario, los modelos jerárquico y en red se están dejando de utilizar.

4.1.Modelo Jerárquico

Un **SGBD** de tipo jerárquico utiliza árboles, para la representación lógica de los datos. La figura siguiente muestra un diagrama de estructura en árbol con dos tipos de registros profesor y alumno y una posible instancia a la **BD**:



Un **SGBD** jerárquico posee las siguientes características:

- Los registros, llamados segmentos, están dispuestos en forma de árbol y no pueden existir ciclos.

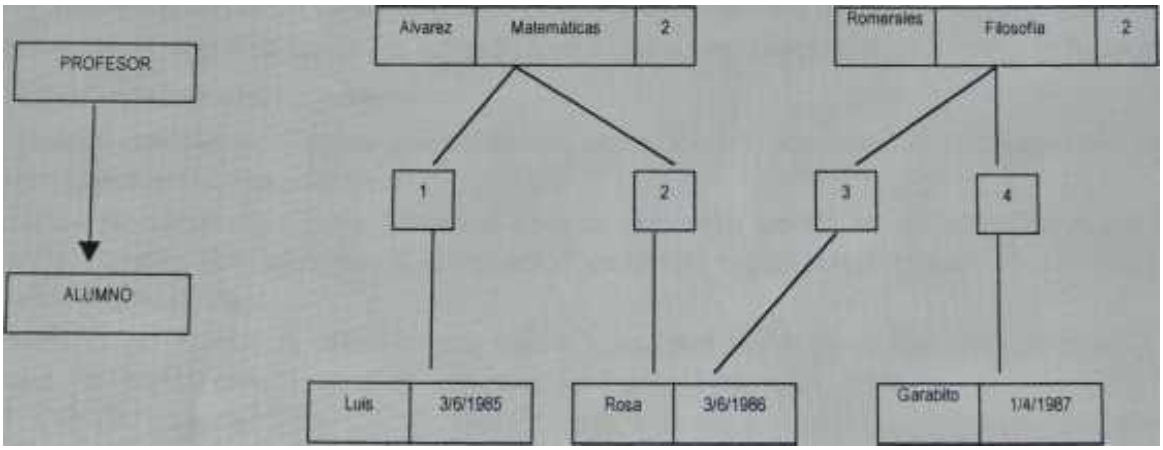
- Los registros sólo pueden estar enlazados mediante relaciones uno a uno o uno a muchos
- Cuando se elimina un registro padre se deben borrar todos sus registros hijos.

El SGBB jerárquico más conocido es IMS que utiliza el lenguaje de consulta DL/I.

4.2.Modelo en Red

Los **SGBD** en red se basan en la utilización de la estructura no lineal red (o Plex), en la que cada registro hijo puede tener más de un nodo padre.

El modelo red más extendido es el modelo CODASYL que presenta la restricción de que no permite directamente la relación muchos a muchos. Si se presenta, se utiliza un registro intermediario, llamado conector o liga-D, que contiene los campos clave principales de los registros que se desean relacionar, como se observa en la siguiente figura:



Un **SGBD** Codasyl tiene las siguientes características:

- La estructura principal del esquema conceptual es el conjunto DBTG que consiste en dos tipos de registros, con una relación entre ellos. Existe un solo propietario y uno o más miembros.
- Los dos tipos de registros deben de ser diferentes.
- Un registro miembro puede asociarse con más de un propietario.

Algunos **SGBD** Codasyl existentes en el mercado son: DMS1100, IDMS e IDS.

4.3.Modelo Relacional

Este modelo es posterior a los dos modelos anteriores, y fue desarrollado por Codd en 1970.

Un **SGBD** relacional utiliza tablas bidimensionales (relaciones) para la representación lógica de los datos y las relaciones entre ellos.

Se llamará registro o tupla a cada fila de la tabla y campo o atributo a cada columna de la tabla. Una clave será un atributo o conjunto de atributos que identifique de forma única a una tupla.

Las tablas deben cumplir una serie de requisitos:

- La tabla sólo puede tener un tipo de registro.
- No existen registros duplicados.
- Los registros dentro de una relación no tienen una secuencia determinada.

- Se pueden crear nuevas tablas relacionando campos procedentes de dos o más tablas ya existentes.

Algunos de los **SGBD** relacionales existentes son: Oracle, MySQL, MariaDB o Ms SQL Server.

4.4.Modelo orientado a Objetos

El modelo orientado a objetos se basa en encapsular código y datos en una única entidad, llamada objeto. El interfaz entre un objeto y el resto del sistema se define mediante un conjunto de mensajes.

Un objeto tiene asociado:

- Un conjunto de variables que contienen los datos del objeto.
- Un conjunto de mensajes a los que el objeto responde.
- Un método, que es el código que implementa la respuesta a un mensaje.

Los objetos similares en una **BD** se agrupan formando clases, organizadas en forma jerárquica.

Un **SGBD** orientado a objetos permite:

- Objetos complejos. Objetos que contienen otros objetos, por ejemplo, relaciones que pueden almacenarse en otras relaciones.
- Datos de comportamiento. Distintos objetos necesitan responder de diferente manera a la misma orden. Por ejemplo, la eliminación de ciertas tuplas puede llevar a la eliminación de otras tuplas asociadas.
- Metaconocimiento. A menudo son más importantes reglas generales sobre la relación, más que las tuplas específicas. Por ejemplo, la regla “Todas las cuentas con saldo mayor de 1000000 pagan el 5 por 100 de interés”. Esta regla no se puede representar fácilmente con un **SGBD** tradicional mientras que con uno orientado a objetos si se puede.

4.5.Modelo distribuido

La forma tradicional centralizada de tratar los datos mediante un solo ordenador está dejando paso que los datos se almacenen en varios ordenadores situados en diferentes lugares, bien en la misma localidad, bien en localidades distintas formado redes de área local (LAN) o de área global (WAN).

En las localidades, los ordenadores del sistema distribuido pueden variar en cuanto a su tamaño función: desde microordenadores hasta mainframes Estos ordenadores no comparten, la memoria principal y cada uno de ellos debe tener un **SGBD**

Un objetivo básico de un **SGBD** distribuido es que el usuario lo perciba como si de una **BD** centralizada se tratase, es decir, el usuario no tiene que saber dónde se encuentran almacenados los datos físicamente.

La principal ventaja de un **SGBD** distribuido es la capacidad para compartir y acceder a la información de manera eficiente, pero tiene las desventajas de la mayor complejidad y coste del software, y mayor posibilidad de errores.

Casi todos los **SGBD** actuales, de entre los que cuentan con amplia difusión, permiten la organización distribuida.

5. Legislación sobre protección de datos

La legislación sobre la protección de datos en España se basaba principalmente en el Reglamento General de Protección de Datos (RGPD) de la Unión Europea y la Ley Orgánica 3/2018 de Protección de Datos Personales y garantía de los derechos digitales (LOPDGDD). Estas leyes establecen las normas y regulaciones para el tratamiento de datos personales en España y se aplican tanto a las empresas como a las instituciones gubernamentales.

El marco normativo en España por tanto lo conforman los siguientes elementos:

- ✓ RGPD: El RGPD es un reglamento de la Unión Europea que establece los principios generales para el tratamiento de datos personales en toda la UE. España, como miembro de la UE, se adhiere a este reglamento, que entró en vigencia el 25 de mayo de 2018.

- ✓ LOPDGDD: La Ley Orgánica 3/2018 de Protección de Datos Personales y garantía de los derechos digitales es la ley española que adapta el RGPD a la legislación nacional y establece normas específicas adicionales. Esta ley regula cuestiones como la autoridad de control en España, los derechos de los ciudadanos en relación con sus datos personales, las sanciones por incumplimiento y otros aspectos relacionados con la protección de datos.

- ✓ Agencia Española de Protección de Datos (AEPD): La AEPD es la autoridad de control en España encargada de supervisar y garantizar el cumplimiento de las leyes de protección de datos. Tiene la facultad de imponer sanciones en caso de infracciones.

- ✓ Delegado de Protección de Datos (DPO): El RGPD exige que algunas organizaciones designen un Delegado de Protección de Datos, que es responsable de supervisar la conformidad con la legislación de protección de datos dentro de la organización.

5.1.LOPDGDD

A continuación, se detallan algunas de las especificaciones más importantes contenidas en la LOPDGDD:

- ✓ **Ámbito de Aplicación**: La ley establece su ámbito de aplicación, que incluye el tratamiento de datos personales en el sector público y privado, así como el ejercicio de los derechos digitales de los ciudadanos.

- ✓ **Definiciones**: La LOPDGDD proporciona definiciones claras de conceptos clave relacionados con la protección de datos y los derechos digitales.

- ✓ **Principios de Protección de Datos**: La ley establece los principios fundamentales que deben guiar el tratamiento de datos personales, como la licitud, lealtad y transparencia, la finalidad del tratamiento, la minimización de datos, la exactitud, la limitación del almacenamiento y la confidencialidad.

- ✓ **Derechos de las Personas**: La LOPDGDD detalla los derechos de las personas en relación con sus datos personales, incluyendo el derecho de acceso, rectificación, supresión, oposición, portabilidad y limitación del tratamiento.

- ✓ **Delegado de Protección de Datos (DPO**): La ley regula la figura del Delegado de Protección de Datos, estableciendo en qué circunstancias es necesario designarlo y cuáles son sus funciones.

- ✓ **Tratamiento de Datos en el Ámbito Laboral**: Se establecen regulaciones específicas para el tratamiento de datos personales en el contexto de las relaciones laborales, incluyendo la videovigilancia en el lugar de trabajo.

- ✓ **Tratamiento de Datos en el Ámbito de la Salud:** La LOPDGDD regula el tratamiento de datos de salud y establece la creación de un registro de tratamientos de datos de salud.

- ✓ **Tratamiento de Datos en el Ámbito Educativo**: Se establecen regulaciones para el tratamiento de datos en instituciones educativas y el uso de tecnologías en la educación.

- ✓ **Datos Genéticos, Biométricos y de Orientación Sexual**: La ley establece normas específicas para el tratamiento de datos genéticos, biométricos y de orientación sexual.

- ✓ **Régimen Sancionador**: Se detallan las sanciones y multas que pueden imponerse en caso de incumplimiento de la ley, que pueden variar en gravedad según la naturaleza de la infracción.

- ✓ **Protección de los Derechos Digitales**: La LOPDGDD incluye disposiciones relacionadas con la protección de derechos digitales, como la neutralidad en la red y la garantía de acceso a Internet.

- ✓ **Transferencias Internacionales de Datos**: Se establecen normas para las transferencias internacionales de datos personales fuera de la Unión Europea.

- ✓ **Autoridades de Control**: La ley define las competencias y responsabilidades de la Agencia Española de Protección de Datos (AEPD) y de las autoridades de control autonómicas.

- ✓ **Información no Personal en Posesión de las Administraciones Públicas**: Regula el tratamiento de información no personal en posesión de las administraciones públicas.

- ✓ **Control Interno en Entidades del Sector Público**: Establece normas para el control interno en entidades del sector público en relación con la protección de datos.

- ✓ **Notificaciones Electrónicas**: Introduce el sistema de notificaciones electrónicas obligatorias para ciertas comunicaciones de la administración pública.

- ✓ **Protección de Datos en el Ámbito de la Justicia**: Se establecen regulaciones específicas para el tratamiento de datos personales en el ámbito judicial.

6. BIG DATA

Se puede dar una definición del Big Data desde dos puntos de vista:

Desde la perspectiva del **tamaño de los datos**, puede definirse como una gran cantidad de datos que no caben en una sola máquina, que se producen de una forma muy rápida y que, a veces, también es necesario interpretarlos y procesarlos en tiempo real.

Desde un punto de vista **puramente tecnológico**, se define como un conjunto de procesos y tecnologías que permiten recoger y almacenar cantidades enormes de datos de distintas procedencias y tipologías, siendo la base tanto de la digitalización masiva del mundo analógico, como del almacenamiento de los propios datos generados en el mundo digital.

Por lo tanto, podemos decir que son conjuntos de datos de gran tamaño, complejidad y velocidad de crecimiento, que hacen difícil su captura, gestión y procesamiento a través de herramientas convencionales, como pueden ser las bases de datos relacionales.

¿Y de cuánto volumen hablamos? La mayoría de expertos actualmente lo sitúa en torno a los 30-50 TB como mínimo y con un máximo que alcanza los varios petabytes.

(Busca la escala de unidades de almacenamiento y hazte una idea de la cantidad de información de la que estamos hablando)

La complejidad del Big Data proviene de la naturaleza no estructurada de los datos que generan las tecnologías modernas, como las redes sociales, los smartphones, los blogs, los sensores que incorporan los dispositivos actuales, los sistemas de identificación por radiofrecuencia, los GPS, etc. Esto, junto al gran volumen de datos, hace necesario el empleo de herramientas de Big Data tanto para la recolección de datos como para su posterior análisis.

6.1. Las 5 Vs del Big Data

- ✓ **Volumen**

Como ya hemos mencionado, *no hay una determinada cantidad de datos a partir de la cual se consideren datos masivos:* **muchos Terabytes**. Una mediana empresa puede utilizar un volumen de datos de cientos de Gigabytes pero esa información no se considera Big Data. De hecho, una empresa que guarde ese volumen de datos no utiliza la tecnología Big Data. Sí se usa, en cambio, por empresas muy grandes. Por ejemplo, de comercio electrónico o una entidad financiera.

- ✓ **Velocidad**

Los datos usados en Big Data se trabajan a mayor velocidad que los gestionados en bases de datos tradicionales. La inteligencia de datos se ocupa de datos que se generan en tiempo real, o incluso a velocidad superior de un dato por segundo. Por ejemplo, las transacciones que se realizan en la Bolsa de Nueva York en un día, donde las operaciones se ejecutan en menos de un nanosegundo.

Internet ha pasado a ser sin duda en el mayor motor de generación de contenidos y de mayor velocidad. En un minuto se envían más de 2.000 millones de e-mails, se realizan más de cuatro millones de búsquedas en Google o se suben una media de 300 horas de vídeo a YouTube.

- ✓ **Variedad**

Si en alguna ocasión has trabajado con una base de datos sabrás que, en su mayoría, los datos que contiene son texto y números, a menudo relacionados entre sí en una base de datos relacional. Big Data, sin embargo, usa formatos más variados, no se trabaja solo con textos y números si no fotografías, vídeos, audio, series de datos temporales, etc. Además, a menudo son datos no estructurados como puede ser el contenido que se genera en un blog o en Twitter.

- ✓ **Veracidad**

Cuando hablamos de veracidad nos referimos a la incertidumbre de los datos, es decir, al grado de fiabilidad de la información recibida. Big Data usa métodos que intenta detectar la *infoxicación* de los datos como las fake news y datos imprevisibles para la toma de decisiones.

- ✓ **Valo**r

El valor de los datos a gran escala está unido a la **ventaja que podamos obtener de los datos**. Usando esos datos a través de la analítica o del Big Data analytics, podríamos realizar las siguientes acciones:

- Optimizar procesos
- Conocer mejor a sus clientes
- Ofrecerles publicidad asociada a sus gustos.
- Mejorar la competitividad.

Hay una frase de Stefan Gross-Selbeck, presidente de Xing que define el valor de los datos:

“Los datos personales son el petróleo del siglo XXI”.

El Big Data sería el equivalente a la extracción de la materia prima mientras que la analítica (esto es, la aplicación de algoritmos sobre los datos) supondría la acción de refinar esos datos a fin de darles un valor añadido en el mercado, siendo muchas veces el paso previo para la creación de una inteligencia artificial.

6.2.Tipos de Big Data

Cuando hablamos de Big Data y los datos asociados a esta técnica, podemos clasificarlos en dos tipos: **según su procedencia y según su estructura.**

Las fuentes según su procedencia son:

- Páginas web y blogs, todos aquellos datos que los usuarios generan al navegar por la Red.
- Redes sociales.

- Transacciones.
- Datos generados por la interacción entre sensores inteligentes en máquinas, también llamada comunicación machine-to-machine.
- Datos generados por la tecnología de reconocimiento biométrico.
- Datos generados por personas y organizaciones públicas y privadas a través de emails, mensajes, grabaciones de llamadas, estadísticas, historiales, etc.

Las fuentes según de su estructura, los datos pueden ser:

- Estructurados, datos con formato, tamaño y longitud definidas.
- Semiestructurados, son datos con una estructura flexible, como los que se usan en XML y HTML o JSON.
- No estructurados, aquellos datos que no tienen un formato específico, como los textos o los contenidos multimedia.

6.3.Elementos de la tecnología Big Data

✓ **Sistema de almacenamiento**

Es la infraestructura, física y lógica que almacena de forma eficiente las grandes cantidades de datos que se usan como fuente para el Big Data. Debe proporcionar capacidad y velocidad de proceso por lo que no sirven las estructuras de los medios tradicionales de almacenamiento. Se ha optado por usar muchos servidores con poca capacidad individualmente pero sí de forma conjunta. El software tradicional tampoco era válido para los fines del Big Data. Por eso se crearon nuevos gestores de datos como puede ser Hadoop.

✓ **Sistema de procesamiento**

Por eso, además de ser capaces de almacenar los datos, debemos poder realizar cálculos y operaciones matemáticas partiendo de los mismos.

✓ **Sistema de comunicación**

Tanto para poder almacenar datos, como para poder acceder a ellos, resulta necesario una infraestructura de red.

6.4.Para qué sirve Big Data y por qué es tan importante

Analítica y estadística.

A partir de la visualización de datos y su análisis podemos comprender cuestiones que eran desconocidas, como:

- ✓ Conocer la situación de un determinado elemento (por ejemplo, las ventas de una compañía) y las razones por las que sucede.
- ✓ Predecir el futuro más cercano a partir de los datos, para prepararnos y anticipar decisiones.

Esto ha supuesto la aparición de nuevos profesionales capaces de interpretar los datos, como el Big Data Analyst o el Big Data Architect, así como para adoptar decisiones estratégicas con respecto a los mismos, como el Chief Data Officer.

- ✓ Los particulares también pueden beneficiarse de las aplicaciones de salud que recogen los datos de relojes inteligentes que miden nuestras constantes vitales y pueden realizar un diagnóstico precoz de enfermedades, por ejemplo.
- ✓ Para empresas, uno de los principales usos es la publicidad conductual o publicidad dirigida, que es una forma de predecir, mediante el comportamiento de los usuarios, qué servicio o producto podría interesarte comprar. De este modo, nos aparecen los anuncios de aquellos productos que verdaderamente nos interesan.

6.5. Herramientas de Big Data

- ✓ **Las bases de datos NoSQL**: permiten trabajar con datos no estructurados, además son fácilmente escalables, lo que facilita el trabajo con grandes volúmenes de datos. MongoDB o Apache Cassandra son dos ejemplos de este tipo de bases de datos.

- ✓ **Hadoop** es una herramienta de código abierto con la que se pueden gestionar grandes cantidades de datos, analizarlos y procesarlos.

- ✓ **Lenguajes de programación** que funcionan especialmente bien con Big Data**,** como **R o Python**.

- ✓ **La biblioteca de JavaScript D3.js** permite producir visualizaciones dinámicas e interactivas de datos en navegadores web mediante HTML, SVG y CSS.

- ✓ **Elasticsearch** permite procesar grandes cantidades de datos y ver su evolución en tiempo real. También proporciona gráficos para presentar la información.

- ✓ **Apache Storm** es una herramienta de código abierto que puede usar con cualquier tipo de lenguaje de programación y es capaz de procesar en tiempo real grandes cantidades datos, creando topologías para transformarlos y analizarlos.