# ¿Qué es un fichero? (2.1)

**Fichero o archivo: conjunto** de **información** **relacionada**, tratada como un todo y **organizada** de forma estructurada. Es una **secuencia** de **dígitos** **binarios** que **organiza** **información** **relacionada** con un mismo aspecto.

|  |
| --- |
| Los ficheros están formados por **registros lógicos** que contienen **datos** **relativos** a un **mismo** **elemento** u objeto (datos de usuarios).  Los datos almacenados se pueden añadir, suprimir, actualizar o consultar individualmente en cualquier momento.  **Solo** se **puede** **llevar** a la **memoria** **principal** **partes** de los ficheros porque son **muy** **pesados**.  La **cantidad** de **información** que es **transferida** entre el **soporte** donde está almacenado el fichero, y la **memoria** **principal**, en una sola operación de lectura/grabación, **se** **llama** **registro físico**o **bloque**.  En una **operación** de lectura/grabación se **transfiere** un **bloque**, con **varios** **registros** lógicos del fichero, a estos registros dentro de un bloque se les llama **factor de blocaje**, y a la operación de agruparlos **bloqueo de registros.** |

**Tipos de ficheros (2.2)**

|  |  |  |  |
| --- | --- | --- | --- |
| Ficheros permanentes | | | |
| Contienen… | | Ejemplo | |
| Maestros | **Estado** **actual** de los datos que pueden **modificarse** desde la aplicación. Es la parte **central** de la aplicación, su **núcleo**. | | Usuarios de una plataforma educativa |
| Constantes | **Datos** **fijos** para la aplicación. **No** **suelen** ser **modificados** y se **accede** a ellos para **realización** de **consultas**. | | Códigos postales |
| Históricos | **Datos** que **fueron** **considerados** como **actuales** en un periodo o **situación** **anterior**. Se utilizan para la **reconstrucción** de **situaciones**. | | Usuarios que han sido dados de baja en la plataforma educativa |

|  |  |
| --- | --- |
| Ficheros temporales | |
| Almacenan... | |
| Intermedios | Resultados de una aplicación que serán utilizados por otra |
| Maniobras | Datos de una aplicación que no pueden ser mantenidos en memoria principal por falta de espacio. |
| Resultados | Datos que van a ser transferidos a un dispositivo de salida. |

# Los soportes de información (2.3)

|  |
| --- |
| Los soportes más utilizados para almacenar los ficheros son:   * Las cintas magnéticas * Los discos (magnéticos, ópticos, o magneto-ópticos). |

|  |  |  |
| --- | --- | --- |
| Tipos de soportes | | |
| Acceso Directo a los datos | Son los **más** **usados**  Se puede **acceder** a los datos de **forma** **directa**  Podemos **colocarnos** en la **posición** que nos **interesa** y **leer** a partir de ella. | Discos |
| Acceso Secuencial | Usados para **copias** de **seguridad**  **No** se puede **acceder** a los datos de **forma** **directa**.  Para leer un dato, hay que **leer** **todo** lo que hay **hasta** **llegar** a esa **posición**. | Cintas magnéticas |

# Métodos de acceso (2.4-2.8)

|  |
| --- |
| Objetivos fundamentales de estas modificaciones |
| * **Proporcionar** un **acceso** **rápido** a los **registros**. |
| Conseguir **economizar** el **almacenamiento**. |
| **Facilitar** la **actualización** de los **registros**. |
| **Permitir** que la **estructura** refleje la **organización** **real** de la información. |

|  |  |
| --- | --- |
|  | |
| **Ficheros secuenciales** | **Definición:** Los **registros** están **almacenados** de **forma** **contigua**, para acceder a un registro hay que **leer** **todos** los registros **hasta** **llegar** a él.  **Características:**  La **lectura** siempre se realiza hacia delante.  Son **ficheros** **monousuarios**, **no** **permiten** el **acceso** **simultáneo** de varios usuarios.  Tienen una **estructura** **rígida** de campos. **Todos** los **registros** deben **aparecer** en **orden** y la **posición** de los **campos** de cada **registro** **siempre** ha de ser **la** **misma**.  El modo de apertura del fichero, condiciona la lectura o escritura.  **Aprovechan** al **máximo** el **soporte** de **almacenamiento**, **al** **no** **dejar** **huecos** **vacíos**.  Se pueden **grabar** en **cualquier** **tipo** de **soporte**, secuenciales y direccionables.  Todos los **lenguajes** de programación pueden **trabajar** con este tipo de **ficheros**.  **No** se **pueden** **insertar** **registros** **entre** **los** que ya están **grabados**. |

|  |  |
| --- | --- |
| **Ficheros de acceso directo** | **Definición:** Se puede **acceder** a un **registro** indicando la **posición** **relativa** del mismo dentro del archivo o a través de una **clave** que forma parte del registro como un campo más. Estos archivos deben **almacenarse** en **dispositivos** de **memoria** masiva de **acceso** **directo**.  **Campo clave:** permite **identificar** y **localizar** un **registro** de manera **ágil** y **organizada**.  Cada uno de los **registros** se guarda en una **posición** **física**, que **dependerá** del **espacio** **disponible** en memoria masiva, de ahí que la distribución de los registros sea aleatoria dentro del soporte de almacenamiento. Para **acceder** a la **posición** **física** de un registro se **utiliza** una **dirección** o **índice.**  **Características:**  **Posicionamiento** **inmediato**.  Registros de **longitud** **fija**.  **Apertura** del fichero en **modo** **mixto**, para lectura y escritura.  Permiten **múltiples** **usuarios** utilizándolos.  Los registros se **borran** colocando un **cero** en la **posición** que **ocupan**.  Permiten la utilización de **algoritmos** de **compactación** de **huecos**.  Los **archivos** se crean con un **tamaño** **definido**, es decir, con un **máximo** de **registros** establecido durante la creación.  Esta organización sólo es posible en **soportes** **direccionables**.  **Se** **usan** cuando el acceso a los datos de un registro se hace siempre empleando la **misma** **clave** y la **velocidad** de **acceso** a un registro es lo que más nos importa.  Permiten la **actualización** de los **registros** en el mismo fichero, **sin** **necesidad** de **copiar** el **fichero**.  Permiten realizar procesos de **actualización** en **tiempo** **real**. |

|  |  |
| --- | --- |
| **Ficheros indexados** | Utilizando **índices***,* permiten el acceso a un registro del fichero de forma directa.  Existe una **zona de registros** en la que se encuentran los datos del archivo y una **zona de índices,** que contiene una tabla con las claves de los registros y las posiciones donde se encuentran estos. La tabla de índices estará ordenada por el campo clave.  La **tabla de índices** se carga en **memoria principal** para **buscar** la fila correspondiente a la **clave del registro** a encontrar, obteniéndose así la **dirección** donde se **encuentra** el **registro**.  Una vez **localizada** la dirección, sólo hay que **acceder** a la **zona** de **registros** en el **soporte** de almacenamiento y **posicionarnos** en la **dirección** indicada. Puesto que la tabla debe prever la inclusión de todas las direcciones posibles del archivo.  Su principal **inconveniente** resulta **determinar** su **tamaño** y **mantenerla** **ordenada** por los valores de la clave.  **Características:**    El diseño del registro tiene que tener un campo, o combinación de campos, que permita identificar cada registro de forma única, es decir, que no pueda haber dos registros que tengan la misma información en él. A este campo se le llama campo clave y es el que va a servir de índice. Un mismo fichero puede tener más de un campo clave, pero al menos uno de ellos no admitirá valores duplicados y se le llama clave primaria. A las restantes se les llama claves alternativas.  **Permiten** **utilizar** el modo de **acceso** **secuencial** y el modo de **acceso** **directo** para leer la información guardada en sus registros.  Utilizando el modo de acceso directo:  Se hace **conociendo** el **contenido** del **campo** **clave** del **registro** que queremos localizar. Con esa información el **sistema** **operativo** puede **consultar** el **índice** y **conocer** la **posición** del **registro** dentro del **fichero**.  Utilizando el modo de acceso secuencial:  Los **registros** son **leídos** **ordenados** por el **contenido** del **campo** **clave**, **independientemente** **del** **orden** en que se **fueron** **grabando** (el orden lógico no es igual al orden físico), debido a que el **acceso** a los **datos** se hace **a** **través** del **índice**, que para **hacer** **más** **fácil** la **búsqueda** de los **registros**, permanece **siempre** **ordenado** por el **campo** **clave**.  **Solamente** se puede **grabar** en un **soporte** **direccionable.** |

|  |  |
| --- | --- |
| **Ficheros Secuenciales Indexados** | Al igual que en los **ficheros** **indexados** existe una **zona de índices** y otra **zona de registros de datos**, pero esta última se encuentra dividida en **segmentos** (bloques de registros) ordenados.  **Organización** muy **utilizada**, para **procesos** en los que **intervienen** **pocos** **registros** y en los que se **maneja** el **fichero** **completo**.  **Características:**  **Permite** el **acceso** **secuencial**, interesante cuando la **tasa** de **actividad** es **alta** y los **registros** se **leen** **ordenados** por el **campo** **clave**.  Permite el **acceso** **directo** a los registros, empleando para ello las **tablas** de **índices**. Primero **busca** la **clave** en el **área** de **índices** y luego va a **leer** al **área** de **datos** en la **dirección** que le **indica** la **tabla**.  Se pueden **actualizar** los **registros** en el **mismo** **fichero**, **sin** **necesidad** de **crear** un **fichero** **nuevo** de copia en el proceso de actualización.  **Ocupa** **más** **espacio** en el disco **que** **los** **ficheros** **secuenciales**, **debido** al uso del **área** de **índices**.  **Solo** se puede utilizar **soportes** **direccionables**.  **Inversión** económica **mayor**, por la **necesidad** de **programas** y **hardware** más **sofisticado**. |

|  |  |
| --- | --- |
| **Ficheros de Acceso Calculado o Hash** | **Permite** **accesos** más **rápidos**, ya que, **en** **lugar** de **consultar** una **tabla**, se **utiliza** una transformación o **función** **matemática** (función de hashing) conocida, que **a** **partir** de la **clave** **genera** la **dirección** de cada **registro** del **archivo**. Si la **clave** es **alfanumérica**, deberá **previamente** ser **transformada** en un **número**.  El **mayor** **problema** que presenta este tipo de ficheros es que **a** **partir** de **diferentes** **claves** se **obtenga** la **misma** **dirección** al aplicar la función matemática o transformación.  A este **problema** se le **denomina** **colisión**, y las claves que generan la misma dirección se conocen por **sinónimos**. Para **resolver** este **problema** se aplican **diferentes** **métodos**, como **tener** un **bloque** de **excedentes** o zona de **sinónimos**, o **crear** un **archivo** de **sinónimos**.  **Métodos de extracción:**  Módulo**:** La dirección será igual al resto de la división entera entre la clave y el número de registros.  Extracción**:** La dirección será igual a una parte de las cifras que se extraen de la clave. |

# Parámetros de utilización (2.9)

|  |  |  |
| --- | --- | --- |
|  | | |
| **Capacidad o volumen** | Espacio, en caracteres, que ocupa el fichero. La capacidad podrá calcularse multiplicando el número previsto de registros por la longitud media de cada registro | |
| **Actividad** | Permite conocer la cantidad de consultas y modificaciones que se realizan en el fichero | |
| **Tasa de consulta o modificación** | Porcentaje de registros consultados o modificados en cada tratamiento del fichero, respecto al número total de registros contenidos en él |
| **Frecuencia de consulta o modificación** | Número de veces que se accede al fichero para hacer una consulta o modificación en un periodo de tiempo fijo. |
| **Volatilidad** | Mide la cantidad de inserciones y borrados que se efectúan en un fichero | |
| **Tasa de renovación** | Tanto por ciento de registros renovados en cada tratamiento del fichero, respecto al número total de registros contenidos en él |
| **Frecuencia de renovación** | Número de veces que se accede al fichero para renovarlo en un periodo de tiempo fijo |
| **Crecimiento** | Variación de la capacidad del fichero y se mide con la tasa de crecimiento, que es el porcentaje de registros en que aumenta el fichero en cada tratamiento | |

# Conceptos (3.1)

**Base de datos: Es una colección de datos relacionados lógicamente entre sí, con una definición y descripción comunes y que están estructurados de una determinada manera.**

**Es un conjunto estructurado de datos que representa entidades y sus interrelaciones, almacenados con la mínima redundancia y posibilitando el acceso a ellos eficientemente por parte de varias aplicaciones y usuarios.**

|  |  |  |
| --- | --- | --- |
| Elementos de una Base de Datos | |  |
| **Entidades** | **Objeto** **real** o **abstracto** con **características** **diferenciadoras** de otros, del que se **almacena** **información** en la base de datos. | En una clínica veterinaria, podría ser: ejemplar, doctor, consulta |
| **Atributos** | **Datos** que se **almacenan** **de** la **entidad**. **Cualquier** **propiedad** o **característica** de una **entidad** puede ser atributo. | Raza, color, nombre, número de identificación |
| **Registros** | **Donde** se **almacena** la **información** de **cada** **entidad**. Es un **conjunto** de **atributos** que **contienen** los **datos** que **pertenecen** a una **misma** **repetición** de entidad. | 2123056, Sultán, Podenco, Gris, 23/03/2009 |
| **Campos** | **Donde** se **almacenan** los **atributos** de **cada** **registro**. | El valor Podenco |

|  |  |
| --- | --- |
| Ventajas | |
| **Acceso múltiple** | **Varios** **usuarios** o **aplicaciones** podrán **acceder** a la **base** de **datos**, **sin** que existan **problemas** en el acceso o los datos |
| **Utilización múltiple** | **Cada** **uno** de los **usuarios** o **aplicaciones** podrán disponer de una **visión** **particular** de la **estructura** de la **base** de **datos**, de tal manera que **cada** **uno** de ellos **accederá** **sólo** a la **parte** que **realmente** le **corresponde**. |
| **Flexibilidad** | La **forma** de **acceder** a la **información** puede ser **establecida** de **diferentes** **maneras**, ofreciendo **tiempos** de **respuesta** muy **reducidos**. |
| **Confidencialidad y seguridad** | El **control** del **acceso** a los **datos** podrá ser **establecido** para que **unos** **usuarios** o **aplicaciones** puedan **acceder** a **unos** **datos** y a **otros** **no**, **impidiendo** a los **usuarios** **no autorizados** la **utilización** de la **base** de **datos**. |
| **Protección contra fallos** | **En caso** de **errores** en la **información**, existen **mecanismos** bien **definidos** que **permiten** la **recuperación** de los **datos** de forma fiable |
| **Independencia física** | Un **cambio** de **soporte** **físico** de los datos (tipo de discos), **no** **afectaría** a la **base** de **datos** o a las aplicaciones que acceden a ellos |
| **Independencia lógica** | Los **cambios** **realizados** en la **base** de **datos** **no** **afectan** a las **aplicaciones** que la usan |
| **Redundancia** | Los **datos** se **almacenan**, por lo general, una **única** **vez**. Aunque si es necesario, **podríamos** **repetir** **información** de **manera** **controlada** |
| **Interfaz de alto nivel** | **Mediante** la **utilización** de **lenguajes** de **alto** **nivel** puede **utilizarse** la **base** de **datos** de **manera** **sencilla** |
| **Consulta directa** | **Existe** una **herramienta** para poder **acceder** a los **datos** **interactivamente** |

# Usos (3.2)

|  |  |
| --- | --- |
| Ventajas | |
| **El administrador** | **Persona** **encargada** de la **creación** o **implementación** **física** de la **base** **de** **datos**. Escoge los **tipos** de **ficheros**, los **índices** que, la **ubicación** de éstos, también, establecerá la **política** de **seguridad** y de acceso para **garantizar** el **menor** **número** de **problemas**. |
| **Los diseñadores** | **Personas** **encargadas** de **diseñar** cómo será la **base** de **datos**.  **Identificarán** los **datos**, las **relaciones** entre ellos, sus **restricciones**...  El **diseñador** de la base de datos debe **implicar** en el **proceso** a **todos** **los** **usuarios** de esta. |
| **Los programadores de aplicaciones** | Se **encargan** de **implementar** los **programas** de **aplicación** que **servirán** a los **usuarios** **finales**.  **Programas** que **realizan** **consultas** de **datos**, **inserción**, **actualización** o **eliminación** de los mismos.  Para **desarrollar** estos **programas** se utilizan [**lenguajes** de **tercera** o **cuarta** **generación**](https://moodlefp.educacion.navarra.es/pluginfile.php/83432/mod_resource/content/2/BD01_Contenidos_Imprimible/index.html#t149edf5d-e218-6122-0d61-973b85aaceb0). |
| **Los usuarios finales** | Son los **clientes** **finales** de la **base** de **datos**.  Al diseñar, implementar y mantener la base de datos se busca cumplir los requisitos establecidos por el cliente para la gestión de su información. |

# Ubicación de la información (3.3)

|  |  |
| --- | --- |
|  | |
| **Discos SATA** | **Interfaz** de **transferencia** de **datos** entre la **placa** **base** y algunos **dispositivos** de **almacenamiento:**  Disco duro, lectores y grabadores de CD/DVD/BD, Unidades de Estado Sólido...  SATA proporciona:   * **Mayores** **velocidades** * **Mejor** **aprovechamiento** cuando hay **varias** **unidades** * **Mayor** **longitud** del **cable** de **transmisión** de **datos** * **Capacidad** para **conectar** **unidades** al **instante**, es decir, insertar el dispositivo **sin** tener que **apagar** el **ordenador**   SATA 150 MB/s SATA II 300 MB/s SATA III 600 MB/s. |
| **Discos SCSI** | **Interfaces** para **discos** **duros** de **gran** **capacidad** de **almacenamiento** y **velocidad** de **rotación**.  Su **tiempo** **medio** de **acceso** puede llegar a **7 milisegundos** y su velocidad de **transmisión** **secuencial** de información es de:   * SCSI Estándar (Standard SCSI) - 5 MB/s en los discos * SCSI Rápido (Fast SCSI) - 10 MBps en los discos * SCSI Ancho-Rápido (Fast-Wide SCSI) - 20 MBps en los discos   Un controlador SCSI puede manejar hasta 7 discos duros SCSI. |
| **RAID** | Acrónimo de **Redundant** **Array** of **Independent** **Disks** o matriz de discos independientes,  Es un **contenedor** de **almacenamiento** **redundante**.  Se basa en el **montaje** en **conjunto** de **dos** **o más** **discos** duros, **formando** un **bloque** de trabajo, para **obtener** desde una **ampliación** de **capacidad** a **mejoras** en **velocidad** y **seguridad** de **almacenamiento**.  **Según** las **características** que queramos primar, se establecen **distintos** **sistemas** de RAID. |

|  |  |
| --- | --- |
| **Sistemas NAS** | Acrónimo de **Network** **Attached** **Storage** ó sistema de almacenamiento masivo en red.  Estos sistemas de almacenamiento **permiten** **compartir** la **capacidad** de **almacenamiento** **de** un **computador** (Servidor) con **ordenadores** **personales** o **servidores** clientes a **través** **de** una **red**, **haciendo** **uso** de un **sistema** **operativo** optimizado para **dar** **acceso** a los **datos** **a** **través** de **protocolos** de **comunicación** **específicos**.  Suelen ser **dispositivos** para **almacenamiento** **masivo** de datos con **capacidades** **muy** **altas**, de varios Terabytes, generalmente superiores a los discos duros externos y además se **diferencian** de estos al **conectar** por **red**. |
| **Sistemas SAN** | Acrónimo de **Storage** **Area** **Network** o red de área de almacenamiento. Se trata de una **red** **concebida** para **conectar** **servidores**, matrices (arrays) de discos y librerías de soporte.  La arquitectura de este tipo de sistemas **permite** que los **recursos** de **almacenamiento** estén **disponibles** para **varios** **servidores** **en** **una** **red** de **área** **local** o **amplia**.  Debido a que la **información** **almacenada** **no** **reside** directamente en **ninguno** de los **servidores** de la **red**, se **optimiza** el poder de **procesamiento** **para** **aplicaciones** **comerciales** y la **capacidad** de **almacenamiento** se puede **proporcionar** en el **servidor** donde **más** se **necesite**. |

# Modelos (4-4.3)

|  |  |
| --- | --- |
| Modelos | |
| **Jerárquico** | Se utiliza una **jerarquía** en la que la **relación** entre las entidades de este modelo siempre es del **tipo** **padre**/**hijo**. De tal manera que existen **nodos** que **contienen** **atributos** o **campos** y que se **relacionarán** con sus **nodos** **hijos.**  **Cada** **nodo** puede **tener** **más** de un **hijo**, pero **únicamente** tendrá un **padre**.  Los **datos** de este **modelo** se **almacenan** en **estructuras** **lógicas** **llamadas** **segmentos**. Los segmentos se **relacionan** entre sí **utilizando** **arcos**. La **forma** **visual** de este **modelo** es de **árbol** **invertido**, en la **parte** **superior** están los **padres** y en la **inferior** los **hijos**.  **Hoy** en día, debido a sus limitaciones, **está** en **desuso**. |
| **Red** | **Organiza** la **información** en **registros** (también llamados nodos) y enlaces. En los **registros** se **almacenan** los **datos**, mientras que los **enlaces** permiten **relacionar** estos **datos**. Las **bases** de **datos** en **red** son **parecidas** a las **jerárquicas** sólo que en ellas **puede** **haber** **más** de **un** **padre**.  Se pueden **representar** **perfectamente** **cualquier** tipo de **relación** entre los **datos**, pero hace **muy** **complicado** su **manejo**.  Al **no** tener que **duplicar** la **información** se **ahorra** **espacio** de almacenamiento.  El **sistema** de **gestión** de **información** **basado** en el modelo en red **más** **popular** es el **sistema** **IDMS**. |
| **Relacional** | **Hoy** en **día** las bases de datos **relacionales** son las más **utilizadas**.  En el **modelo** **relacional** la base de datos es un **conjunto** de **tablas**.  Esta percepción es **sólo** a **nivel** **lógico**, ya que a **nivel** **físico** puede estar **implementada** mediante **distintas** **estructuras** de almacenamiento.  El modelo relacional utiliza **tablas bidimensionales** (relaciones) para la **representación** **lógica** de los **datos** y las **relaciones** entre **ellos**. Cada **relación** (tabla) **posee** un **nombre** que es **único** y contiene un **conjunto** de **columnas**.  **Registro, entidad o tupla:** cada fila de la tabla y **campo o atributo** a cada **columna** de la **tabla**.  **Dominio: Conjuntos** de **valores** que puede tomar un **determinado** **atributo**.  **Clave:** atributo o conjunto de atributos que identifique de forma única a una tupla.  **Requisitos**  **Todos** los **registros** son del **mismo** **tipo**.  La **tabla** **sólo** puede tener **un** **tipo** de registro.  **No** **existen** **campos** o **atributos** **repetidos**.  **No** **existen** **registros** **duplicados**.  **No** **existe** **orden** en el **almacenamiento** de los **registros**.  **Cada** **registro** o tupla es **identificada** por una **clave** que puede estar **formada** por uno o **varios** **campos** o **atributos**. |
| **Orientado a Objetos** | **Base** de **datos** en términos de **objetos**, sus **propiedades** y sus **operaciones**.  Los objetos con la misma estructura y comportamiento pertenecen a una **clase**, y las clases se organizan en jerarquías.  Las **operaciones** de cada **clase** se **especifican** en **términos** de **procedimientos** **predefinidos** denominados **métodos**.  Algunos sistemas basados en el modelo relacional, han evolucionado incorporando conceptos orientados a objetos llamados **objeto-relacionales.**  El **objetivo** de este modelo es **cubrir** las **limitaciones** del **modelo** **relacional** ya quese **incorporan** **mejoras** como la **herencia** entre **tablas**, los **tipos** **definidos** por el **usuario**, **soporte** **multimedia…**  **Conceptos más importantes:**  Encapsulación: Propiedad que **permite** **ocultar** la **información** al **resto** de los **objetos**, **impidiendo** así **accesos** **incorrectos** o **conflictos**.  Herencia: Propiedad a través de la cual los **objetos** **heredan** **comportamiento** **dentro** de una **jerarquía** de **clases**.  Polimorfismo: Propiedad de una **operación** **mediante** la cual puede ser **aplicada** a **distintos** tipos de **objetos**.  Desde la **aparición** de la **programación** orientada a objetos (**POO** u OOP) se empezó a pensar en bases de datos adaptadas a estos lenguajes.  Este **modelo** es considerado como el **fundamento** de las **bases** de **datos** de **tercera** **generación**, siendo consideradas las bases de datos en red como la primera y las bases de datos relacionales como la segunda generación.  Aunque no han reemplazado a las bases de datos relacionales, si son el **tipo** de base de datos que más está **creciendo** en los últimos años. |

|  |  |
| --- | --- |
| Otros Modelos | |
| **Objeto-Relacional** | Son un **híbrido** entre las bases del **modelo** **relacional** y el **orientado** a **objetos**.  El mayor **inconveniente** son los **costes** de la **conversión** de las BD **relacionales** a BD **orientadas** a **objetos**.  Se busca **obtener** lo **mejor** del **modelo** **relacional**, **incorporando** las **mejoras** **ofrecidas** por la **orientación** a **objetos**.  Se siguen **almacenando** **tuplas**, aunque la **estructura** de las **tuplas** **no** está **restringida**, sino que las **relaciones** pueden ser **definidas** en **función** de **otras**, que es lo que denominamos **herencia** **directa**.  El **estándar** en el que se **basa** este **modelo** es [**SQL99**](https://moodlefp.educacion.navarra.es/pluginfile.php/83432/mod_resource/content/2/BD01_Contenidos_Imprimible/index.html#t846d4b5d-6cd0-3162-7d2e-f16ce87abcbb). Este estándar **ofrece** la **posibilidad** de **añadir** a las **bases** de **datos** **relacionales** **procedimientos** **almacenados** de **usuario**, **triggers**, **tipos** **definidos** por el **usuario**, **consultas** **recursivas**, **bases** de **datos** [OLAP](https://moodlefp.educacion.navarra.es/pluginfile.php/83432/mod_resource/content/2/BD01_Contenidos_Imprimible/index.html#t01f42944-9f69-4653-a163-0eaceb36a84b), **tipos** [**LOB**](https://moodlefp.educacion.navarra.es/pluginfile.php/83432/mod_resource/content/2/BD01_Contenidos_Imprimible/index.html#t432afe15-6410-5ee8-cbcb-04ea743eceb0), ...  Otra **característica** a **destacar** es la **capacidad** para **incorporar** **funciones** que tengan un **código** en algún **lenguaje** de **programación** como, por ejemplo: **SQL**, **Java**, **C**…  La gran **mayoría** de las **bases** de **datos** **relacionales** **clásicas** de **gran** **tamaño**, como **Oracle**, **SQL Server**, son **objeto-relacionales**. |
| **Bases de datos deductivas** | **Almacenan** la **información** y **permiten** **realizar** **deducciones** a **través** de [**inferencias**.](https://moodlefp.educacion.navarra.es/pluginfile.php/83432/mod_resource/content/2/BD01_Contenidos_Imprimible/index.html#tbcb6a61b-7cbc-6ef8-78b4-46e4cbc6b8eb) Es decir, se **derivan** **nuevas** **informaciones** a **partir** de **las** que se han **introducido** **explícitamente** en la **base** de **datos** por parte del **usuario**.  **También** son **llamadas** BD **lógicas**, al **basarse** en **lógica** **matemática**.  Surgieron para **contrarrestar** las **limitaciones** del **modelo** **relacional** para la **respuesta** a **consultas** [**recursivas**](https://moodlefp.educacion.navarra.es/pluginfile.php/83432/mod_resource/content/2/BD01_Contenidos_Imprimible/index.html#t939feba5-bcd8-2f9f-aed6-5c20e457af0e) y la **deducción** de **relaciones** **indirectas** entre los **datos** **almacenados**. |

|  |  |
| --- | --- |
| **Bases de datos multidimensionales** | **Ideadas** para **desarrollar** **aplicaciones** muy **concretas**.  **Almacena** sus **datos** con **varias** **dimensiones**, **en** **vez** de **un** **valor**, **encontramos** **varios** **dependiendo** de los **ejes** **definidos** o **una** **BD** de **estructura** **basada** en **dimensiones** **orientada** a **consultas** **complejas** y **alto** **rendimiento**.  La **información** se **representa** como **matrices multidimensionales**, **cuadros** de **múltiples** **entradas** o **funciones** de **varias** **variables** sobre **conjuntos** **finitos**.  **Cada** **una** de **estas** **matrices** se denomina **cubo**. Eso **facilita** el **manejo** de **grandes** **cantidades** de **datos**, dándole a esto una **amplia** **aplicación** **dentro** de **varias** **áreas** y **diferentes** **campos** del **conocimiento** **humano**. |
| **Bases de datos transaccionales** | **Caracterizadas** por su **velocidad** para **gestionar** el **intercambio** de **información**, **utilizadas** en **sistemas** **bancarios**, **análisis** de **calidad** y **datos** de **producción** industrial.  Son **bases** de **datos** muy **fiables**, ya que **las** **operaciones** de **inserción**, **actualización** o **borrado** se **realizan** **completamente** **o** se **descartan**.  Entre la más destacadas se encuentra **Oracle**. |
| **Modelo de bases de datos orientadas a documentos** | El **principal** **objeto** de **gestión** es el **documento** que **contiene** **datos** **semiestructurados** que podrán estar **almacenados** en **algún** **formato**, por ejemplo, XML.  [**MongoDB**](https://www.mongodb.com/es)y **[CouchDB](http://couchdb.apache.org/" \t "_blank" \o "Accede a la página oficial de CouchDB (se abre en una ventana nueva))** |

# Tipos de bases de datos (5)

|  |  |  |  |
| --- | --- | --- | --- |
| Según su contenido | | | |
| **Información actual** | **Contienen** **información** muy **concreta** y **actualizada**, normalmente, de **tipo** **numérico**: **estadísticas**, series **históricas**, **resultados** de **encuestas**, **convocatorias** de **becas…** | | |
| **Directorios** | **Recogen** datos sobre **personas** o **instituciones** **especializadas** en una **actividad** o **materia** **concreta**.  Hay directorios de **profesionales**, de **bibliotecas**, de **revistas** **científicas…** | | |
| **Documentales** | **Cada** **registro** se **corresponde** con un **documento**, sea éste **de** **cualquier** **tipo**:  Una **publicación** **impresa**, un **documento** **audiovisual**, gráfico.  **Dependiendo** de si **incluyen** **o no** el **contenido** **completo** de los **documentos** que **describen**, podremos tener: | |
| **Texto completo** | **Constituidas** por los propios **documentos** en **formato** **electrónico**, por un **volcado** **completo** de su texto. |
| **Archivos electrónicos de imágenes** | **Constituidos** por **referencias** que **permiten** un **enlace** **directo** con la **imagen** del **documento** original. |
| **Referenciales** | Sus **registros** **no** **contienen** el **texto** **original** sino **tan** **sólo** la **información** **fundamental** para **describir** y **permitir** la **localización** de **documentos** obtener **referencias** sobre **documentos** que **habrá** que **localizar** **posteriormente** en **otro** **servicio.** |

|  |  |
| --- | --- |
| Según su uso | |
| **Individual** | **Utilizada** básicamente por **una** **persona**.  El **sistema** **administrador** de la **base** de **datos** y los **datos** son **controlados** por el **mismo** **usuario**.  **Puede** estar **almacenada** en la **unidad** de **disco** **duro** del usuario **o** en el **servidor** de **archivos** de una **red** de **área** **local**.  Por ejemplo, un **gerente** de **ventas** podría contar con una **base** de **datos** para el **control** de sus **vendedores** y su **desempeño**. |
| **Compartida** | Son **bases** de **datos** con **múltiples** **usuarios** y que muy probablemente pertenezcan a la **misma** **organización**.  Se **encuentra** **almacenada** en una **computadora** **potente** y bajo el **cuidado** de un **profesional** en el **área**, el **administrador** de la **base** de **datos**.  Los **usuarios** tienen **acceso** a la **base** de **datos** **mediante** una **red** de **área** **local** o una **red** de **área** **extensa**. |
| **Acceso Publico** | Son **bases** de **datos** **accesibles** por **cualquier** **persona**.  Puede **no** ser **necesario** **pagar** un **canon** para hacer **uso** de los **datos** contenidos en ellas. |
| **Propietarias o Bancos de Datos** | **Bases** de **datos** de **gran** **tamaño**, **desarrolladas** por una **organización** y que **contienen** **temas** **especializados** o de **carácter** **particular**.  El **público** **general** puede tener **acceso** a estas **bases** a veces de forma **gratuita** y otras **mediante** el **pago** de una cuota.  Pueden **ofrecer** **información** que va desde **negocios**, economía, **inversión**, técnica y **científica…**.  Permiten **encontrar** en **minutos** lo que tardaría **horas** **ojeando** **revistas**. |

|  |  |
| --- | --- |
| Según la variabilidad de la información | |
| **Estáticas** | Son **BD** de **sólo** **lectura**.  Se **utilizan** para el **almacenamiento** de **datos** **históricos** que pueden ser **analizados** y **utilizados** para el **estudio** del **comportamiento** de un **conjunto** de **datos** a **través** del **tiempo**.  Permiten **realizar** **proyecciones** y **toma** de **decisiones**. |
| **Dinámicas** | Son **BD** donde la **información** **almacenada** se **modifica** con el **tiempo**, **permitiendo** **operaciones** como **actualización** y **adición** de **datos**, además de las **operaciones** **fundamentales** de **consulta,** |

|  |  |  |  |
| --- | --- | --- | --- |
| Según la localización de la información | | | |
| **Distribuidas** | Se **sitúan** en un **lugar** o **lugares** **diferentes** a **donde** se **encuentran** los **usuarios**.  Es la **unión** de las **bases** de **datos** **mediante** **redes**.  Los **usuarios** se **vinculan** a los **servidores** de **BD** **distantes** **mediante** una **amplia** **variedad** de **redes** de **comunicación**. | | |
| **Centralizadas** | **Ubicadas** en un **único** **lugar**, un **único** **computador**.  Pueden ser **bases** de **datos** **monousuario** que se **ejecutan** en **ordenadores** **personales** o **sistemas** de **BD** de **alto** **rendimiento** que se **ejecutan** en **grandes** **sistemas**.  **Facilita** las **labores** de **mantenimiento**, sin embargo, **hace** que la **información** contenida en **dicha** **base**, sea más **vulnerable** a **posibles** **fallos** y **limita** su **acceso**.  **Ofrece** **dentro** de la **arquitectura** **Cliente/Servidor** dos configuraciones: | |
| **Anfitrión** | La **máquina** **cliente** y la **máquina** **servidor** son la **misma**.  Los **usuarios** se **conectarán** **directamente** a la **máquina** **donde** se **encuentra** la **BD**. |
| **Cliente/Servidor** | La **BD** **reside** en una **máquina** **servidor** y los **usuarios** **acceden** a la **BD** **desde** su **máquina** **cliente** a **través** de una **red.** |

# 

# 

# 

# Sistemas gestores de base de datos (6)

Sistema Gestor de Base de Datos: **Conjunto** coordinado de **programas**, **procedimientos**, **lenguajes**, que **suministra**, tanto a los **usuarios** no informáticos, como a los **analistas**, **programadores**, o al **administrador**, los **medios** **necesarios** para **describir** y **manipular** los **datos** contenidos en la **base** de **datos**, **manteniendo** su [**integridad**](https://moodlefp.educacion.navarra.es/pluginfile.php/83432/mod_resource/content/2/BD01_Contenidos_Imprimible/index.html#t4b80553d-e04a-242e-5725-226d322db9d0), **confidencialidad** y **seguridad**.

El SGBD **permite** a los **usuarios** la **creación** y el **mantenimiento** de una **base** de **datos**, facilitando la **definición**, **construcción** y **manipulación** de la **información** contenida en ésta. **Definir** una **base** de **datos** **consistirá** en **especificar** los **tipos** de **datos**, las **estructuras** y las [**restricciones**](https://moodlefp.educacion.navarra.es/pluginfile.php/83432/mod_resource/content/2/BD01_Contenidos_Imprimible/index.html#tec9091cf-4ae4-33b3-1ce3-c8eabc18df2e) **que** han de **cumplir los** **datos** a la **hora** de **almacenarse** en esta.

Por otro lado, la **construcción** de la **base** será el **proceso** de **almacenamiento** de datos **concretos** en algún **medio** o **soporte** de almacenamiento que esté **supervisado** por el SGBD.

Finalmente, la **manipulación** de la **base** de **datos** incluirá la **posibilidad** de **realización** de **consultas** para **recuperar** **información** específica, la **actualización** de los datos y la **generación** de **informes** **a** **partir** de su **contenido**.

|  |
| --- |
| Ventajas |
| **Proporcionan** al **usuario** una **visión** **abstracta** de los **datos**, **ocultando** parte de la **complejidad** relacionada con **cómo** **se** **almacenan** y **mantienen** los datos.  Ofrecen **Independencia física:** la **información** que ve el **usuario**, y la **manipulación** de los **datos** almacenados en la Base de Datos, es **independiente** de **cómo** estén **almacenados** **físicamente**.  **Disminuyen** la **redundancia** y la **inconsistencia** de **datos**.  **Aseguran** la **integridad** de los datos.  **Facilitan** el **acceso** a los **datos**, aportando **rapidez** y evitando la **pérdida** de **datos.**  **Aumentan** la **seguridad** y **privacidad** de los datos.  Mejoran la **eficiencia**.  Permiten **compartir** **datos** y **accesos** **concurrentes**.  Facilitan el **intercambio** de **datos** entre **distintos** **sistemas**.  Incorporan **mecanismos** de **copias** de **seguridad** y **recuperación** para **restablecer** la **información** en caso de **fallos** en el **sistema**. |

# Funciones (6.1)

|  |  |
| --- | --- |
| Descripción o Definición | |
| **Permite** al **diseñador** de la base de datos **crear** las **estructuras** **apropiadas** para **integrar** **adecuadamente** los **datos,** permite **definir** las **tres** **estructuras** de la base de datos.  Esta **función** se **realiza** **mediante** el **lenguaje** de **descripción** de **datos** o **DDL**.  Mediante ese lenguaje: se **definen:**   * Las **estructuras** de **datos**, * Las **relaciones** entre los **datos** * Las **reglas** (restricciones) que han de cumplir los datos.   Se **especificarán** las **características** de los **datos** a **cada** **uno** de los **tres** **niveles** y el **SGBD** **se** **ocupará** de la **transformación** de las **estructuras** **externas** orientadas a los usuarios a las **estructuras** **conceptuales** y de la **relación** de ésta y la **estructura** **física**. | |
| **Interno** | **Indicar** el **espacio** de **disco** reservado para la base de datos, la **longitud** de los **campos**, su **modo** de **representación** (lenguaje para la definición de la estructura externa). |
| **Conceptual** | **Proporcionan** **herramientas** para la **definición** de las **entidades** y su **identificación**, **atributos** de las mismas, **interrelaciones** entre ellas, **restricciones** de **integridad**, etc.  Es decir, el **esquema** de la **base** de **datos** (lenguaje para la definición de estructura lógico global). |
| **Externo** | **Definir** las **vistas** de los **distintos** **usuarios** a **través** del **lenguaje** para la **definición** de **estructuras** **externas**. |

|  |
| --- |
| Manipulación |
| **Permite** a los **usuarios** de la base **buscar**, **añadir**, **suprimir** o **modificar** los **datos** de la misma, siempre de **acuerdo** con las **especificaciones** y las **normas** de **seguridad** **dictadas** por el **administrador**. Se llevará **a cabo** por medio de un **lenguaje** de **manipulación** de **datos** (DML) que **facilita** los **instrumentos** **necesarios** para la **realización** de estas **tareas**.  También se encarga de **definir** la [**vista externa**](https://moodlefp.educacion.navarra.es/pluginfile.php/83432/mod_resource/content/2/BD01_Contenidos_Imprimible/index.html#t2846abee-b7d6-3ad8-4cc3-125956a84c14) de **todos** los **usuarios** de la **base** de **datos** o **vistas** **parciales** que **cada** **usuario** tiene **de los datos** **definidos** con el **DDL**.  **Por manipulación de datos entenderemos:**  La **recuperación** de **información** **almacenada** en la **base** de **datos**, llamado, **consultas**.  La **inserción** de **información** **nueva**.  El **borrado** de **información**.  La **modificación** de **información** **almacenada**. |

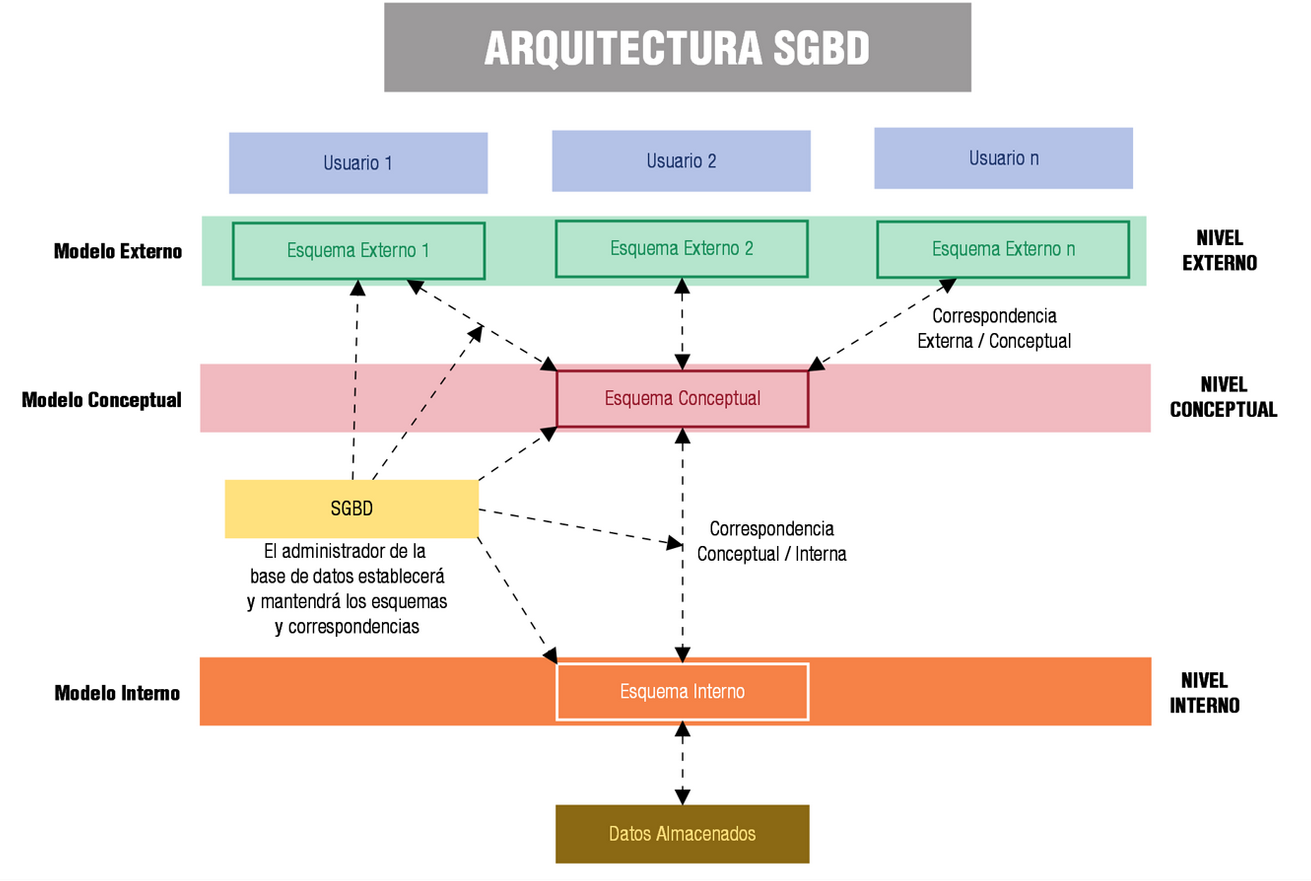
|  |
| --- |
| Control |
| Permite al **administrador** de la base de datos **establecer** **mecanismos** de **protección** de las **diferentes** **visiones** de los **datos** **asociadas** a **cada usuario**, **proporcionando** **elementos** de **creación** y **modificación** de dichos **usuarios**.  Adicionalmente, **incorpora** **sistemas** para la **creación** de **copias** de **seguridad**, **carga** **de** **ficheros**, **auditoría**, **protección** de **ataques**, **configuración** del **sistema**, etc.  El **lenguaje** que **implementa** esta **función** es el **lenguaje** de **control** de **datos** o **DCL** |

# Componentes (6.2)

|  |  |
| --- | --- |
|  | |
| **Lenguajes de la base de datos** | A través de los lenguajes se pueden **especificar** los **datos** que **componen** la **BD**, su **estructura**, **relaciones**, **reglas** de **integridad**, **control** de **acceso**, **características** **físicas** y **vistas** **externas** de los usuarios.  Los lenguajes del SGBD son:   * Lenguaje de Definición de los Datos (**DDL**) * Lenguaje de Manejo de Datos (**DML**) * Lenguaje de Control de Datos (**DCL**). |
| **El diccionario de datos** | **Descripción** de los **datos** **almacenados**.  Es el **lugar** donde se **deposita** la **información** sobre la **totalidad** de los **datos** que **forman** la **base** de **datos**.  Contiene las **características** **lógicas** de las **estructuras** que **almacenan** los **datos**, su **nombre**, **descripción**, **contenido** y **organización**.  El diccionario de datos aportará información sobre:   * **Estructura** **lógica** y **física** de la BD. * **Definición** de **tablas**, **vistas**, **índices**, **disparadores**, **procedimientos**, **funciones**. * **Cantidad** de **espacio** **asignado** y **utilizado** por los **elementos** de la BD. * **Descripción** de las **restricciones** de **integridad**. * **Información** sobre los **permisos** **asociados** a cada usuario. * **Auditoría** de **acceso** a los **datos**, utilización. |
| **El gestor de la base de datos** | Es la **parte** de **software** **encargada** de **garantizar** el **correcto**, **seguro**, **íntegro** y **eficiente** **acceso** y **almacenamiento** de los **datos**.  Es el **encargado** de:   * **Proporcionar** una **interfaz** **entre** los **datos** **almacenados** y los **programas** de **aplicación** que los manejan. * **Garantizar** la **privacidad**, **seguridad** e **integridad** de los **datos**, **controlando** los **accesos** **concurrentes** e **interactuando** con el **sistema** **operativo**   Es un **intermediario** entre el **usuario** y los **datos**. |
| **Usuarios de la base de datos** | En los SGBD existen **diferentes** **perfiles** de **usuario,** **cada** **uno**, con **unos** **permisos** sobre los **objetos** de la BD:  **El administrador de la base de datos** (DBA): **persona** o **conjunto** de **ellas** **encargadas** de la **función** de **administración** de la base de datos.  Tiene el **control** **centralizado** de la BD y es el **responsable** de su **buen** **funcionamiento**. Es el **encargado** de **autorizar** el **acceso** a la **base** de **datos**, de **coordinar** y **vigilar** su **utilización** y de **adquirir** los **recursos** **software** y **hardware** que sean **necesarios**.  **Los usuarios de la base de datos: usuarios** de la BD con **diferentes** **necesidades** sobre los datos, **diferentes** **accesos** y **privilegios**.  Podemos establecer la siguiente clasificación:   * Diseñadores. * Operadores y personal de mantenimiento. * Analistas y programadores de aplicaciones. * Usuarios finales: ocasionales, simples, avanzados y autónomos. |
| **Herramientas de la base de datos** | **Conjunto** de **aplicaciones** que **permiten** a los **administradores** la **gestión** de la **base** de **datos**, de los **usuarios** y **permisos**, **generadores** de **formularios**, **informes**, **interfaces** **gráficas**, **generadores** de **aplicaciones…** |

# Componentes (6.3)

|  |  |
| --- | --- |
|  | |
| **Para** una **base** de **datos**, **sólo** **existirá** un **único** **esquema** **interno,** un **único** **esquema** **conceptual** y podrían **existir** **varios** **esquemas** **externos** **definidos** para uno o varios usuarios.  La arquitectura propuesta proporciona **tres** **niveles** de **abstracción**: | |
| **Nivel interno o físico** | Se **describe** la **estructura** **física** de la **BD** a través de un **esquema** **interno** **encargado** de **detallar** el **sistema** de **almacenamiento** de la **BD** y sus **métodos** de **acceso**.  Es el **nivel** más **cercano** al **almacenamiento** **físico**.  A **través** del **esquema** **físico** se **indican**, los **archivos** que **contienen** la **información**, su **organización**, los **métodos** de **acceso** a los **registros**, los **tipos** de **registros**, la **longitud**, los **campos** que los **componen**, las **unidades** de **almacenamiento..**. |
| **Nivel lógico o conceptual** | Se **describe** la **estructura** **completa** de la **BD** a **través** de un **esquema** que **detalla** las **entidades**, **atributos**, **relaciones**, **operaciones** de los **usuarios** y **restricciones**.  Los **detalles** **relacionados** con las **estructuras** de **almacenamiento** se **ocultan**, **permitiendo** **realizar** una **abstracción** a más **alto nivel**. |
| **Nivel externo o de visión del usuario** | Se describen las **diferentes** **vistas** que los **usuarios** **percibirán** de **BD**.  **Cada** **tipo** de **usuario** o **grupo** de ellos **verá** **sólo** la **parte** de la **BD** que le **interesa**, **ocultando** el **resto**. |
| Gracias a esta arquitectura se consigue la **independencia de datos** a dos niveles: | |
| **Independencia lógica** | Podemos **modificar** el **esquema** **conceptual** **sin** **alterar** los **esquemas** **externos** **ni** los **programas** de **aplicación**. |
| **Independencia física** | Podemos **modificar** el **esquema** **interno** **sin** **necesidad** de **modificar** el **conceptual** o el **externo**.  Se puede **cambiar** el **sistema** de **almacenamiento**, **reorganizar** los **ficheros**, **añadir** **nuevos..**., **sin** que **esto** **afecte** al **resto** de **esquemas**. |



# Tipos (6.4)

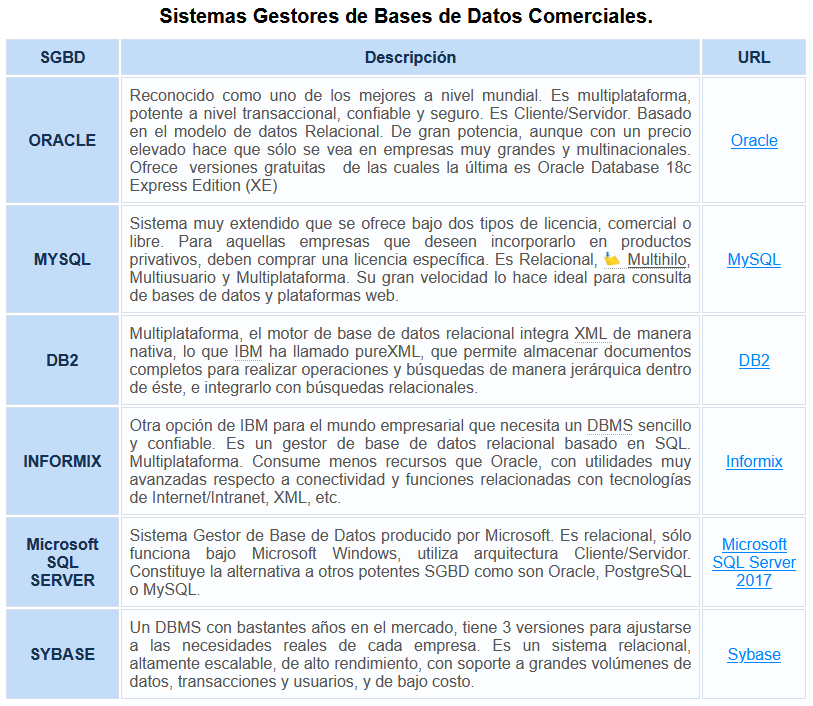
|  |
| --- |
| Según el modelo lógico en que se basan |
| **Actualmente**, el **modelo** **lógico** que **más** se **utiliza** es el relacional.  Los modelos en **red** y **jerárquico** han quedado **obsoletos**.  Otro de los modelos que **más** **extensión** está teniendo es el **modelo** **orientado** a **objetos**.  Por tanto, en esta primera clasificación tendremos:   * Modelo Jerárquico. * Modelo de Red. * Modelo Relacional. * Modelo Orientado a Objetos. |

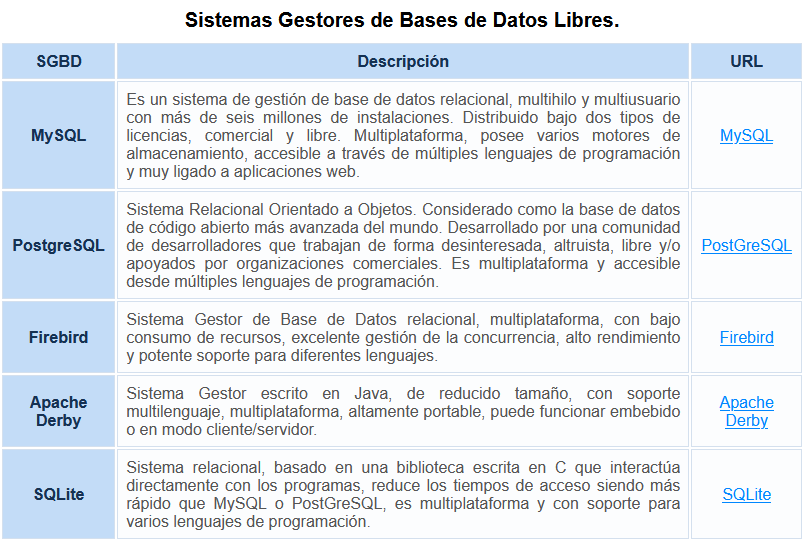
|  |  |
| --- | --- |
| Según el número de usuarios | |
| **Monousuario** | **Sólo** atienden a **un** **usuario** **a la vez**, y su **principal** **uso** se da en los **ordenadores** **personales**. |
| **Multiusuario** | Entre los que se encuentran la mayor parte de los SGBD, **atienden** a **varios** **usuarios** al **mismo** **tiempo**. |

|  |  |
| --- | --- |
| Según el número de sitios en los que está distribuida la base de datos | |
| **Centralizados** | Sus **datos** se **almacenan** en **un** **solo** ordenador.  Los SGBD centralizados **atienden** a **varios** **usuarios**, **pero** el **SGBD** y la **BD** **residen** **por** **completo** en una **sola** **máquina**. |
| **Distribuidos** | La **BD** **real** y el **propio** **software** del **SGBD** están **distribuidos** en **varios** **sitios** **conectados** por una **red**.  Los **sistemas** **homogéneos** **utilizan** el **mismo** **SGBD** en **múltiples** **sitios**.  Una tendencia reciente consiste en crear software para tener acceso a varias bases de datos autónomas preexistentes almacenadas en sistemas distribuidos heterogéneos.  Esto da lugar a los **SGBD** **federados** o **sistemas** **multibase** de datos en los que los **SGBD** participantes **tienen** **cierto** **grado** de **autonomía** **local**. |

|  |
| --- |
| Según el coste |
| La mayor parte de los paquetes cuestan entre 10.000 y 100.000 euros. Los sistemas monousuario más económicos para microcomputadores cuestan entre 0 y 3.000 euros. En el otro extremo, los paquetes más completos cuestan más de 100.000 euros. |

|  |  |
| --- | --- |
| Según el propósito o finalidad | |
| **General** | **Utilizados** para el **tratamiento** de **cualquier** tipo **BD** y **aplicación**. |
| **Específico** | Cuando el **rendimiento** es **fundamental**, se puede **diseñar** y **construir** un **software** de **propósito** **especial** para una **aplicación** **específica**, y **este** **sistema** **no** **sirve** para **otras** **aplicaciones**.  Muchos **sistemas** de **reservas** de **líneas** **aéreas** son de **propósito** **especial** y **pertenecen** a la **categoría** de **sistemas de procesamiento de transacciones en línea**, que deben **atender** un **gran** **número** de **transacciones** **concurrentes** **sin** **imponer** **excesivos** **retrasos**. |

****

****

# Bases de datos centralizadas (9)

**Sistema de base de datos centralizado: Estructura** en la que el **SGBD** está **implantado** en **una** **sola** **plataforma** u ordenador desde **donde** se **gestiona** **directamente**, de **modo** **centralizado**, la **totalidad** de los **recursos**. Es la **arquitectura** de los **centros** de **proceso** de **datos** **tradicionales**. Se basa en **tecnologías** **sencillas**, muy **experimentadas** y de **gran** **robustez**.

|  |
| --- |
| Características |
| Se **almacena** **completamente** en una **ubicación** **central**, **todos** los **componentes** del **sistema** **residen** en un **solo** **computador** o **sitio**.  **No** **posee** **múltiples** **elementos** de **procesamiento** **ni** **mecanismos** de **intercomunicación** **como** las **bases** de **datos** **distribuidas**.  Los **componentes** de las **bases de datos** **centralizadas** son:   * Los **datos** * El **software** de **gestión** de **bases** de **datos** * Los **dispositivos** de **almacenamiento** **secundario** asociados   Son **sistemas** en los que su **seguridad** puede verse **comprometida** más **fácilmente**. |

|  |  |
| --- | --- |
|  | |
| **Ventajas** | Se **evita** la **redundancia** **debido** a la **posibilidad** de **inconsistencias** y al **desperdicio** de **espacio**.  Se **evita** la **inconsistencia**. Ya que, si un **hecho** **específico** se **representa** por **una** **sola** **entrada**, la **no-concordancia** de **datos** **no** **puede** **ocurrir**.  La **seguridad** se **centraliza**.  Puede **conservarse** la **integridad**.  El **procesamiento** de los **datos** **ofrece** un **mejor** **rendimiento**.  **Mantenimiento** más **barato**.  **Mejor** **uso** de los **recursos** y **menores** **recursos** **humanos**. |
| **Inconvenientes** | **Un mainframe** en **comparación** de un **sistema** **distribuido** **no** tiene **mayor** **poder** de **cómputo**.  **Si** **falla**, se **pierde** **toda** **disponibilidad** de **procesamiento** y la **información** **confiada** al **sistema**.  En caso de **desastre** o **catástrofe**, la **recuperación** es **difícil** de **sincronizar**.  Las **cargas** de **trabajo** **no** se **pueden** **difundir** entre **varias** **computadoras**, ya que **los** **trabajos** **siempre** se **ejecutarán** en la **misma** **máquina**.  Los **departamentos** de **sistemas** **retienen** el **control** de **toda** la **organización**.  Los **sistemas** **centralizados** **requieren** un **mantenimiento** **central** de **datos**. |

# Bases de datos distribuidas (10)

|  |  |
| --- | --- |
|  | |
| **Base de datos distribuida** **(BDD)** | **Conjunto** de **múltiples** **bases** de **datos** **lógicamente** **relacionadas** las cuales se **encuentran** **distribuidas** entre **diferentes** **nodos** **interconectados** por una **red**. |
| **Sistema de bases de datos distribuida** **(SBDD)** | Sistema en el cual **múltiples** **sitios** de **BD** están **ligados** por un **sistema** de **comunicaciones**.  Un **usuario** en **cualquier** **sitio** puede **acceder** los **datos** en **cualquier** **parte** de la red **exactamente** como si los **datos** **estuvieran** **almacenados** en su **sitio** **propio**. |
| **Sistema gestor de bases de datos distribuida (SGBDD)** | Aquel que se **encarga** del **manejo** de la **BDD** y **proporciona** un **mecanismo** de **acceso** que hace que la **distribución** sea **transparente** a los **usuarios**.  La **aplicación** **trabajaría**, desde un **punto** de **vista** **lógico**, como si un solo **SGBD** **ejecutado** en una **sola** **máquina** **administrara** esos **datos**. |

Un SGBDD desarrollará su trabajo a través de un conjunto de sitios o nodos, que poseen un sistema de procesamiento de datos completo con una base de datos local, un sistema de gestor de bases de datos e interconectados entre sí.

Si estos nodos están dispersos geográficamente se interconectarán a través de una red de área amplia o WAN, pero si se encuentran en edificios relativamente cercanos, pueden estar interconectados por una red de área local o LAN.

Este tipo de sistemas es utilizado en: organizaciones con estructura descentralizada, industrias de manufactura con múltiples sedes (automoción), aplicaciones militares, líneas aéreas, cadenas hoteleras, servicios bancarios, etc.

|  |  |
| --- | --- |
|  | |
| **Ventajas** | El **acceso** y **procesamiento** de los **datos** es **más** **rápido** ya que **varios** **nodos** **comparten** **carga** de **trabajo**.  Desde **una** **ubicación** puede **accederse** a **información** **alojada** en **diferentes** **lugares**.  Los **costes** son **inferiores** a los de las **bases** **centralizadas**.  Existe **cierta** **tolerancia** a **fallos**. **Mediante** la [**replicación**](https://moodlefp.educacion.navarra.es/pluginfile.php/83432/mod_resource/content/2/BD01_Contenidos_Imprimible/index.html#tf6cf01bb-7067-cb6e-21cd-bbb73be4f6a3), **si un nodo** **deja** de **funcionar** el **sistema** **completo** sigue funcionando.  El enfoque distribuido de las bases de datos se adapta más naturalmente a la estructura de las organizaciones.  **Permiten** la **incorporación** de **nodos** de **forma** **flexible** y **fácil**.  Aunque los **nodos** están **interconectados**, tienen **independencia** **local**. |
| **Inconvenientes** | La **probabilidad** de **violaciones** de **seguridad** es **creciente** si no se toman las precauciones debidas.  Existe una **complejidad** **añadida** que es **necesaria** para **garantizar** la **coordinación** **apropiada** entre los nodos.  La **inversión** **inicial** es **menor**, pero el **mantenimiento** y **control** puede **resultar costoso**.  Dado que los **datos** **pueden** estar **replicados**, el **control** de **concurrencia** y los **mecanismos** de **recuperación** son **mucho** **más** **complejos** que en un sistema centralizado.  El **intercambio** de **mensajes** y el **cómputo** **adicional** **necesario** para **conseguir** la **coordinación** entre los **distintos** **nodos** **constituyen** una forma de **sobrecarga** que **no surge** en los **sistemas** **centralizados**.  Dada la **complejidad** del **procesamiento** entre **nodos** es **difícil** **asegurar** la **corrección** de los **algoritmos**, el **funcionamiento** **correcto** **durante** un **fallo** **o** la **recuperación**. |

# Fragmentación (10.1)

En los **sistemas** de **BD** **distribuidas** la **información** se encuentra **repartida** en **varios** **lugares**.

Para **extraer** los **datos** **consultados** puede **realizarse** mediante la **fragmentación** de **distintas** **tablas** pertenecientes a **distintas** **bases** de **datos** que se encuentran en **diferentes** **servidores**.

El **problema** de **fragmentación** se refiere al **particionamiento de la información** para **distribuir** **cada** **parte** a los **diferentes** **sitios** de la red.

Pero hay que tener en cuenta el **grado de fragmentación** que se aplicará, ya que éste es un factor determinante a la hora de la ejecución de consultas.

**Si** **no** **existe** **fragmentación**, se **tomarán** las **relaciones** o **tablas** como la unidad de fragmentación. Pero también puede fragmentarse a nivel de tupla (fila o registro) o a nivel de atributo (columna o campo) de una tabla.

No será adecuado un grado de fragmentación nulo, ni tampoco un grado de fragmentación demasiado alto.

El grado de fragmentación deberá estar equilibrado y dependerá de las particularidades de las aplicaciones que utilicen dicha base de datos.

Concretando, el objetivo de la fragmentación es encontrar un nivel de particionamiento adecuado en el rango que va desde tuplas o atributos hasta relaciones completas.

|  |  |
| --- | --- |
| Reglas fundamentales a cumplir | |
| **Completitud** | Si una relación R se descompone en fragmentos R1, R2, ..., Rn, cada elemento de datos que pueda encontrarse en R deberá poder encontrarse en uno o varios fragmentos Ri |
| **Reconstrucción** | Si una relación R se descompone en una serie de fragmentos R1, R2, ..., Rn, la reconstrucción de la relación a partir de sus fragmentos asegura que se preservan las restricciones definidas sobre los datos |
| **Disyunción** | Si una relación R se descompone verticalmente, sus atributos primarios clave normalmente se repiten en todos sus fragmentos |

|  |  |
| --- | --- |
| **Tipos de fragmentación** | |
| **Horizontal** | Se **realiza** sobre las **tuplas** de la **relación**, **dividiendo** la **relación** en **subrelaciones** que **contienen** un **subconjunto** de las **tuplas** que **alberga** la **primera**.  Existen **dos** **variantes**: la **primaria** y la **derivada**. |
| **Vertical** | Se basa en los **atributos** de la **relación** para **efectuar** la **división**.  Una relación R produce fragmentos R1, R2, ..., Rr, cada uno de los cuales contiene un subconjunto de los atributos de R así como la llave primaria de R.  El objetivo de la fragmentación vertical es particionar una relación en un conjunto de relaciones más pequeñas de manera que varias de las aplicaciones de usuario se ejecutarán sobre un fragmento.  En este contexto, una fragmentación óptima es aquella que produce un esquema de fragmentación que minimiza el tiempo de ejecución de las consultas de usuario.  La fragmentación vertical es más complicada que la horizontal, ya que existe un gran número de alternativas para realizarla. |
| **Híbrida o mixta** | Podemos combinar ambas, utilizando por ello la denominada fragmentación mixta.  Si tras una fragmentación vertical se lleva a cabo otra horizontal, se habla de la fragmentación mixta (HV).  Para el caso contrario, estaremos ante una fragmentación (VH). Para representar los dos tipos de fragmentación, se utilizan los árboles. |