

9. Si determinamos que un dispositivo de almacenamiento lo es de almacenamiento secundario, estamos estableciendo una clasificación del dispositivo en función de:

- a. La volatilidad del dispositivo.
- b. El Acceso a la información.
- c. La pertenencia o no a la memoria interna del sistema informático.
- d. La retención de datos sin suministro energético.

10. En cuanto a las particiones del disco duro, señale la respuesta falsa

- a. En un disco duro se deben hacer, como mínimo, 4 particiones básicas.
- b. No puede haber más de una partición extendida.
- c. En una partición extendida puede haber una o varias particiones lógicas.
- d. Las particiones extendidas cuentan como si fueran particiones básicas.

UD2

Sistemas gestores de datos en sistemas ERP-CRM

- 2.1. Gestores de datos, tipos y características
 - 2.1.1. Definición de diferentes tipos de gestores de datos
 - 2.1.2. Definición de las características de los sistemas gestores de datos
- 2.2. Arquitectura y componentes de un sistema gestor de datos
 - 2.2.1. Definición de diferentes sistemas de bases de datos
 - 2.2.2. Definición de arquitectura de datos: establecer entidades, relaciones y propiedades
- 2.3. Procesos de instalación de un gestor de datos para albergar sistemas ERP y CRM
 - 2.3.1. Instalación de un sistema gestor de datos
 - 2.3.2. Creación de usuarios de acceso al sistema gestor de datos
 - 2.3.3. Relación de pruebas que verifiquen las diferentes funcionalidades
 - 2.3.4. Relación de pruebas que verifiquen las diferentes funcionalidades
 - 2.3.5. Documentación de los procesos realizados
- 2.4. Parámetros de configuración del gestor de datos
 - 2.4.1. Definición de los parámetros de configuración de un sistema gestor de datos
 - 2.4.2. Diferentes tipos de parámetros

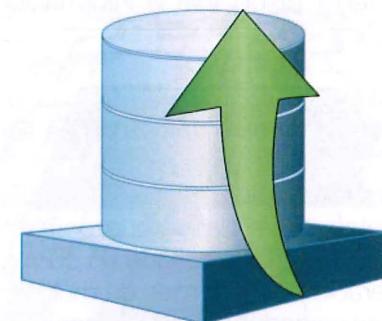
2.1. Gestores de datos, tipos y características

Por si mismos los sistemas ERP y/o CRM no almacenan la información que se genera, transmite, utilizan o captura los usuarios, necesitan ubicar la información en un repositorio. Ese repositorio es lo que llamábamos en la arquitectura del ERP, base de datos.

En estos sistemas es precisamente la base de datos la que les otorga su mayor funcionalidad, gracias a la unicidad de la información que les confiere.



Varios módulos, cada uno correspondiente a cada una de las áreas de gestión de la empresa, y una base de datos, que confiere unicidad a la información, conforman un ERP.



Base de datos, repositorio de información

La elección de esa parte del sistema es un hito importante en el diseño del ERP y/o el CRM- ¿Por qué? Por una razón muy sencilla, sin por eso dejar de ser importante. Es un hecho cierto, y no es la primera vez que lo ponemos de manifiesto, que el principal activo de una organización es la información que maneja. Para que ésta sea útil a la empresa debe de estar disponible mediante accesos fáciles y además ha de guardarse de forma segura, para que se mantenga su integridad y sea confiable.

Dado que la parte del sistema que va a gestionar los datos es el lugar donde se va a ubicar la información la elección del sistema gestor de los datos en un sistema ERP/CRM es una decisión importante.

Tendremos dos opciones a la hora de elegir la gestión que queremos hacer de los datos seleccionados, existen muchos tipos de ERP, y así mismo, también existen distintas opciones de gestores de datos.

Podemos optar por:

- O bien, que el sistema ERP trabaje con gestores de datos propios, que únicamente funcionan con esa aplicación.
- O bien, que el sistema ERP trabaje con gestores de datos de fabricantes externos (como pueden ser Oracle o Microsoft)

Como en el caso de todas las tomas decisiones, hay que barajar las desventajas y ventajas que las distintas opciones presentan antes de decantarse por una de ellas. En el caso de los ERP que usan sus propias bases de datos (BBDD), podemos encontrarnos frente a algunos inconvenientes:

- En ocasiones, es posible ni la base de datos, ni el propio ERP, puedan obtener informes de la información que captura el sistema
- No se pueden hacer informes a medida, por lo que se pierde la funcionalidad del ERP de tener disponible la información oportuna cuando el usuario la requiera
- Desde esta óptica, implementar un ERP del cual no podamos obtener la información con la que se le alimenta no parece ser la mejor opción.

Este inconveniente se solventa implementando un ERP que sea capaz de trabajar con bases de datos comerciales, que permitirán que la información con la que se alimenta el ERP sea consultada en cualquier instante de tiempo y que sea posible la generación de informes de múltiples formatos, elegidos y configurados según las necesidades de los usuarios.

Ventajas del uso BBDD comerciales en ERPs:

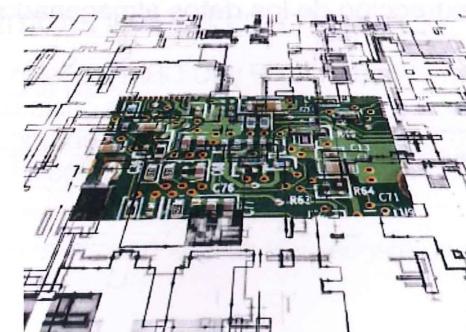
- La información puede consultarse en cualquier momento, permite generación de informes
- No vincularemos la aplicación a la base de datos, por lo que podremos cambiar de proveedor sin tener que hacer una migración de los datos
- Habrá disponible personal técnico especializado sin tener que incrementar en demasía el coste del proyecto

Puesta de manifiesto la necesidad que suponen, vamos a definir pues, que es un gestor de datos.



Llamaremos **gestor de datos** al conjunto de programas que podrán tener acceso y gestionar (almacenando, modificando y extrayendo) la información contenida en una BBDD. Además nos va a proporcionar las herramientas por medio de las cuales podremos añadir, modificar, eliminar y analizar los datos en ella contenidos.

Al conjunto formado por estos programas y la BBDD forman un **Sistema Gestor de Bases de Datos (SGBD)** [También reciben el nombre de DBMS – Data Base Management System]



Sistemas Gestores de Datos

Los gestores de datos surgen de las grandes limitaciones que los sistemas de gestión de archivos que soportaban los propios sistemas operativos tienen para tratar altos volúmenes de información. Estos sistemas los formaban un conjunto de programas que además de trabajar con los datos, los contenían. Los datos se ubicaban, codificados, dentro de los propios programas de aplicación, por lo que la estructura de los datos de los archivos no puede ser modificada si no se modifican también los programas con los que trabajan.

Cuando el volumen de información es alto, como ocurre en el caso de los ERP y/o CRM, modificar todos los programas que manejan los datos será ineficaz, además de costoso.

Para solventar esta traba surgió la idea de separar de los archivos que los manipulan los datos contenidos en los archivos. Este concepto, en principio simple, nos va a permitir poder modificar la estructura de los datos sin tener que modificar los programas por los que estos son usados.



La razón de usar SGBD (Sistemas Gestores de Bases de Datos), en vez de sistemas de gestión de archivos, es poder estructurar y organizar los datos para que se pueda acceder a ellos independientemente de los programas con los que se gestionan.

El OBJETIVO de un SGBD es proporcionar seguridad y eficiencia en el almacenamiento o extracción de los datos almacenados en las BBDD

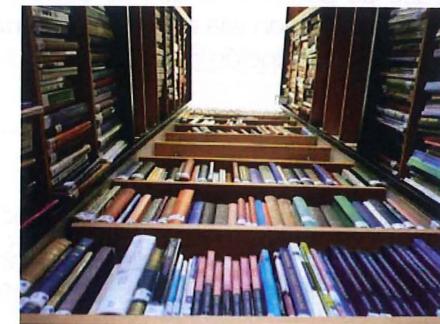
No es nada nuevo afirmar que los ERP y /o CRM manejan grandísimas cantidades de información. En el diseño de los sistemas gestores de datos, teniendo esto en cuenta, se tiene en cuenta, tanto la definición de las estructuras que van a contener los datos, como los procedimientos a través de los cuales se van a gestionar.

2.1.1. Definición de diferentes tipos de gestores de datos

El elemento principal de cualquier gestor de datos, sea del tipo que sea (a continuación veremos que son varios los tipos de gestores de datos disponibles) es la **base de datos (BBDD)**.



Una **BBDD** no es sino un gran contenedor donde se almacena información, perteneciente a un mismo contexto, organizados y relacionados entre sí.



Las bibliotecas también son bases de datos

¿Qué **CARÁCTERISTICAS** tiene una BBDD?:

- Sólo se define una vez
- Permite el acceso simultáneo a la información que contiene de varios usuarios
- Los datos están relacionados entre sí y asegura un mínimo de duplicidad entre ellos
- Se almacenan en ella los metadatos de la información, es decir su descripción.

BBDD

Definición única	Acceso simultaneo a la información	Evita duplicidades	Almacenan metadatos de archivos
------------------	------------------------------------	--------------------	---------------------------------

Características de las BBDD



El componente de la base de datos en el que se deposita la descripción de todos los datos que se almacenan en ella es el **diccionario de datos**. Es una guía en la que se detalla una descripción de la BBDD y de los objetos que la conforman.

En relación a la BBDD, el SGBD (Sistema Gestor de Bases de Datos) tendrá que proporcionar los siguientes servicios a los usuarios: definirla, crearla y mantenerla:

- **Definir y crear la base de datos.** Tendrá que permitirnos especificar la estructura que va tener nuestra base de datos, y que restricciones y relaciones les van a afectar.
 - **Manipulación de los datos.** Tendrá que permitirnos realizar consultas insertar datos y actualizarlos.
 - **Control de acceso a los datos.** Tendrá que proporcionar los mecanismos de seguridad apropiados para asegurar las restricciones en el control de acceso a los datos que contiene la BBDD.



Recuerda

El objetivo de los gestores de datos en un ERP y/o CRM es conseguir que las aplicaciones que van a formar parte del sistema puedan acceder a los datos sin tener la necesidad de saber cómo están estructurados.

Si conseguimos este objetivo, estaremos cumpliendo además, como veremos a continuación, una característica de los SGBD, la abstracción de la información. La BBDD consigue esta abstracción ocultando a los usuarios información, que no es necesaria para ellos, respecto al almacenamiento físico de los datos. El instrumento que va a hacer posible esta percepción de los datos es el **MODELO DE DATOS**.



Definición

Los datos, la relación entre ellos y las restricciones que respecto a ellos deben cumplirse describen la estructura de una base de datos. El conjunto de conceptos que nos ayudarán a describirla y el conjunto de operaciones de nivel básico que nos va a permitir actualizar y consultar los datos, es lo que denominaremos **modelo de datos**.

Modelo de datos

Los diferentes tipos de gestores de datos que vamos a encontrarnos estarán diferenciados por los diferentes modelos de datos que ofrecen. Así podremos encontrarnos con sistemas gestores de datos que utilicen modelos conceptuales, o modelos físicos o modelos lógicos.



Modelos conceptuales

Este modelo de datos hace una representación de los datos independiente-mente al SGBD. Establece unos esquemas internos, que aunque tratan de adaptarse a la realidad, a través de conceptos como entidades, atributos y relaciones, no siempre consiguen el objetivo deseado, si no es después de pasar por varios esquemas previos a la definición de la estructura final.

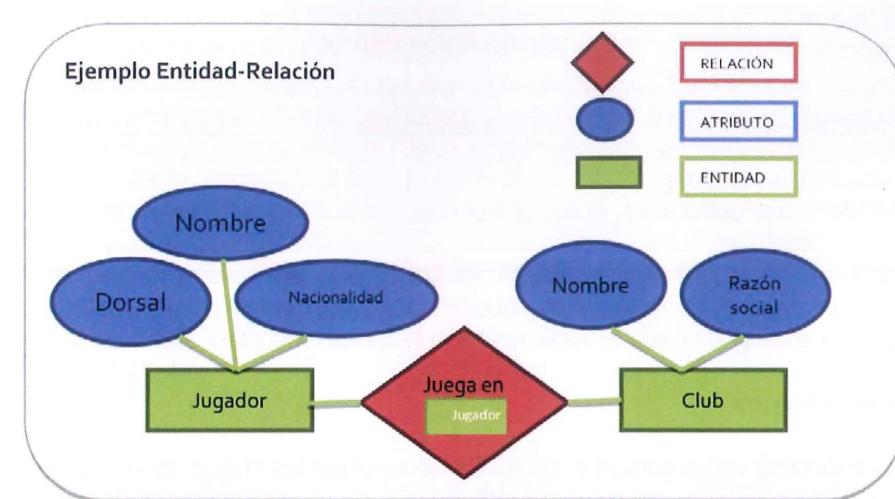
Este tipo de gestor de datos utiliza los conceptos de entidades, atributos y relaciones:

Entidad. Va a representar un objeto o concepto del mundo real

Atributo. Va a representar una propiedad de la entidad

Relación. Va a describir la interacción que existe entre las entidades

El siguiente ejemplo muestra el diseño conceptual, que no es otra cosa que un modelo basado en la entidad/interrelación:



Modelos lógicos

Este modelo de datos lógico, también llamado modelo orientado a registros, utiliza para representar los datos el concepto de estructuras de registros. Es-tas estructuras pueden estar soportadas por modelos lógicos diferentes, de ahí su nombre, que dan a su vez lugar a los **modelos relacional, de red y jerárquico**, que a continuación definiremos.

Muy próximo al modelo conceptual que acabamos de ver, es de reciente apa-ripción un nuevo modelo lógico, el **modelo orientado a objetos**, surgido de las necesidades derivadas de la aparición y auge de la programación orientada a objetos. En ambos, tanto en la programación como en el modelo de datos, es posible la cohesión de procedimientos y datos, de manera que en las estruc-turas que poseen datos se definan también los procedimientos que pueden llevarse a cabo con esos datos. En realidad es una evolución del modelo re-lacional, ya que trata de solucionar las limitaciones que este modelo presenta.



Las bases de datos orientadas a objetos son las llamadas BBDD de tercera generación.

Modelo de datos lógicos		
Modelo relacional	Modelo jerárquico	Modelo de red
Modelo orientado a objetos		

Modelos físicos

Estos modelos van a describir como se almacenan los datos dentro del equipo: la estructura que van a tener los ficheros, los métodos de acceso que se van a utilizar para llegar hasta ellos el formato de los registros, etc.

Importante

A la descripción que se hace de una BBDD a través de un modelo de datos se le llama **esquema de la base de datos**. Especificado durante la fase de diseño de la BBDD no se modifica una vez implementada. Lo que si se modifica son los datos que contiene. Los datos a lo largo de la vida de la base de datos se actualizan, se insertan nuevos, otros se eliminan, etc. A los datos que la BBDD, en un determinado momento del tiempo, contiene se le llama **ocurrencia de la base de datos u estado de la base de datos**.

- Estructura de la BBDD:
 - Descripción de la BBDD mediante un modelo de datos.
 - Una vez diseñada, no se modifica.
- Ocurrencia o Estado de la BBDD:
 - Conjunto de datos, que en un momento determinado, contiene la BBDD.
 - Se modifica frecuentemente

Establecer y tener clara la diferencia entre estos dos conceptos es importante. Debemos de saber que cuando definimos una BBDD tan solo especificamos su esquema al sistema gestor de bases de datos (SGBS). En el momento de definir su estructura la ocurrencia de la base de datos será "vacía", en ese momento no albergará ningún dato. Cuando por primera vez introducimos datos, pasará al estado "inicial", y el estado de la base de datos irá variando cada vez que se lleve a cabo una actualización de los datos que contiene.

El encargado de asegurar que todos los estados de la BBDD sean válidos y adecuados para la estructura especificada y que, además, cumplan con las restricciones que esta estructura determine será el sistema gestor de bases de datos (SGBD).

Esquema BBDD	Estado vacío	Estado inicial
Se crea en la fase de definición	Cuando se crea, la BBDD no contiene datos	Al introducir los primeros datos

2.1.2. Definición de las características de los sistemas gestores de datos

Importante

El objetivo que persiguen los sistemas gestores de datos en un sistema ERP y/o CRM es el de lograr que se puedan manejar de una forma clara, ordenada y sencilla, una relación de datos que con posterioridad van a convertirse en información de alguna forma relevante para la empresa.

Para cumplir con este objetivo todo sistema gestor de datos ha de reunir una serie de **CARACTERÍSTICAS**, que serán:

- Abstracción de la información: Es el hecho de que los usuarios no tengan la necesidad de conocer los detalles relativos a la ubicación física de los datos
- Independencia: Es la capacidad que tiene que tener el gestor de datos para modificar su esquema sin tener que modificar las aplicaciones que utilizan los datos.
- Redundancia mínima: Capacidad de evitar duplicidad de datos
- Consistencia: Es la capacidad que deberá tener el gestor de datos para actualizar la información coherentemente es decir, los datos duplicados deben de actualizarse al mismo tiempo.
- Seguridad: Es la capacidad que tendrá el sistema gestor de datos de proteger la información que contiene frente a accesos no autorizados
- Integridad: Los datos almacenados deben de ser en todo momento válidos. El sistema gestor de datos debe de ser capaz de garantizar esta validez
- Respaldo y recuperación: El sistema debe de ser lo suficientemente eficiente a la hora de realizar una copia de seguridad de los datos que almacena
- Control de la concurrencia: El sistema ha de ser capaz de gestionar los accesos simultáneos a la BBDD, para evitar que se produzcan inconsistencias en la información.

2.2. Arquitectura y componentes de un sistema gestor de datos

Un sistema gestor de datos va a estar definido por su arquitectura y por sus componentes. Veamos primero los componentes.

Importante

Para proporcionar las funciones que hemos definido que debe proveer, el SGBD hará uso de algunas herramientas, son sus componentes.

Debido a la gran cantidad de SGBD con la que podemos encontrarnos, estandarizar sus componentes sería una misión prácticamente imposible. Lo que si es cierto es que hay algunos componentes que son comunes a todos ellos y requeridos para su correcto funcionamiento. Estos componentes son:

- Lenguajes de programación.
- Diccionario de datos.
- Mecanismos de seguridad e integridad de los datos.
- Administrador de la BBDD.

Lenguajes de programación de los SGBD

Los lenguajes que va a utilizar el sistema gestor de datos serán las herramientas que van a permitir a los usuarios realizar determinadas acciones sobre el sistema. Los SGBD van a contar con lenguajes e interfaces diferentes, en función del tipo de usuario que vaya a llevar a cabo según qué acción.

Seguro que después de ver los ejemplos que se muestran a continuación el concepto te queda mucho más claro:

- Los lenguajes les permitirán especificar cuál va a ser la estructura de la BBDD, cuáles serán los datos que la componen y su relación entre ellos.
- Definir las reglas de integridad de la información
- Definir las vistas externas de los usuarios

Principalmente, se pueden distinguir dos tipos de lenguajes, uno para la manipulación de los datos (DML) y otro para su definición (DDL)

Lenguajes SGBD	
Lenguajes de definición de datos (DDL)	Lenguajes de manipulación de datos (MDL)
Usados por los administradores	Usados por los usuarios

- Lenguajes de definición de datos (DDL)

Este es el lenguaje que se van a utilizar los administradores y diseñadores de las bases de datos para especificar qué esquema va a tener la BBDD, cuáles van a ser las estructuras de almacenamiento y para definir tanto el esquema conceptual como el esquema interno de la base de datos

- Lenguajes de manipulación de datos (DML)

Este es el lenguaje que utilizarán los usuarios de los SGBD para llevar a cabo sobre los datos contenidos en la BBDD acciones de lectura y actualizaciones. Así mediante este lenguaje podrán introducir datos, modificarlos, incluso podrán eliminarlos y realizar todo tipo de consultas. Dentro de este tipo de lenguaje se diferencia dos nuevos tipos:

- Lenguajes procedurales

En estos lenguajes lo normal es que el usuario sea un programador y llame a los procedimientos necesarios especificando las operaciones de acceso. Las BBDD en red y jerárquicas hacen uso de estos lenguajes de manipulación de datos.

- Lenguajes no procedurales o declarativos

Las bases de datos relacionales (a continuación veremos los distintos tipos de sistemas de bases de datos que podemos encontrarnos) utilizan este tipo de lenguajes. Mediante sencillos comandos, nos van a permitir especificar qué datos queremos obtener cuando realizamos una consulta o qué datos queremos modificar. Estas instrucciones llegarán al SGBD generalmente desde un terminal, aunque también es posible que lo hagan embebidas dentro de un lenguaje de programación de más alto nivel.



El lenguaje DML no procedural más utilizado por las BBDD relacionales es el SQL (Structured Quero Language).

Tanto los lenguajes DDL como los DML, pueden clasificarse en:

- Lenguajes autosuficientes ☒ Incorporan estructuras de control (if, Then , Else, bucles)
- Lenguajes huésped ☒ no incorporan estructuras de control

Los sistemas gestores de datos comerciales, en su gran mayoría, incluyen los llamados **lenguajes de cuarta generación (4GL)**, también **llamadas herramientas de desarrollo**. Este lenguaje va a permitir al usuario hacer desarrollo de aplicaciones fácil y rápidamente.

Herramientas de desarrollo del SGBD Oracle:

- SQL Forms: generación de formularios de pantalla y para interactuar con los datos.
- SQL Report: generación de informes de los datos contenidos en la BBDD
- PL/SQL Lenguaje: para crear procedimientos que interactúen con los datos que contiene la BBDD

Diccionarios de datos

Es el lugar del sistema que contendrá la información relativa a todos los datos que conforman la BBDD, contiene los metadatos. Contendrá por lo tanto información relativa a los sitios lógicos donde se ubica la información, así como el nombre de los mismos, su descripción y su contenido



Definición

El diccionario de datos es la guía del SGBD que describe la estructura de la BBDD y los elementos que la forman.

¿Sobre qué va a proporcionar información el diccionario de datos del SGBD? En el siguiente esquema vas a encontrar la respuesta a esta pregunta.

El diccionario de datos nos va a proporcionar información sobre:

- La estructura tanto física como lógica de la BBDD
- La definición de todos los objetos o elementos de la BBDD: vistas, índices, tablas, etc.
- El espacio que le ha sido asignado a los objetos, así como el espacio utilizado por ellos.
- Los valores por defecto de las columnas de cada una de las tablas.
- Información relativa a las restricciones que preservan la integridad.
- Los roles y privilegios que se otorgan a los usuarios

Seguridad e integridad

Otro de los elementos fundamentales de los SGBD son los mecanismos a través de los cuales éste debe tener bajo control todas las peticiones que realicen los usuarios y rechazar aquellas que supongan un quebrantamiento de las restricciones de integridad de la información y seguridad que estaban contendidas en el diccionario de datos y que habían sido definidas por el Administrador de la BBDD.



Sabías que

Al Administrador de la base de datos en el entorno de las mismas se le denomina DBA.

Este componente del SGBD ha de asegurar, al menos, lo siguiente:

- Tiene que garantizar que los datos estén perfectamente protegidos ante accesos no autorizados, independientemente de que éstos sean malintencionados o estén autorizados.

- Tiene que garantizar la recuperación de los datos contenidos en su BBDD en caso de ocurrencia de cualquier suceso que pueda dañarla
- Tiene que ser capaz de ofrecer mecanismos que conserven la consistencia de los datos, cuando dos usuarios realicen una actualización simultánea de la BBDD.



Seguridad e Integridad de los datos

Administrador de la BBDD

Vamos a ver el último componente requerido por el SGBD, nos referimos al Administrador.



Definición

El **DBA** va a ser el usuario encargado de llevar el control del sistema a un nivel técnico. Su objetivo principal será garantizar que la BBDD cumple con los objetivos que con su uso se había propuesto la organización. Posee el nivel más alto de privilegios de todos los usuarios que hacen uso del sistema.

Entre sus principales FUNCIONES están:

- Definir el esquema de la BBDD
- Realizar la definición del esquema conceptual
- Establecer las restricciones de integridad y seguridad
- Supervisar el rendimiento de la BBDD
- Establecer el enlace con los usuarios.

Usuarios en SGBD Oracle

- Usuarios DBA : administran la BBDD y tienen el nivel más alto de privilegios
- Usuarios RESOURCE: pueden crear sus propios objetos y, además, si se les ha concedido permiso para ellos, pueden acceder a ciertos objetos.
- Usuarios CONNECT: sólo pueden usar determinados objetos , aquellos para los que se les ha concedido permiso

En cuanto a la arquitectura del sistema gestor de bases de datos vamos a empezar por determinar en que está basada dicha arquitectura. Son tres las características que inherentemente a ellos han de tener los SGBD:

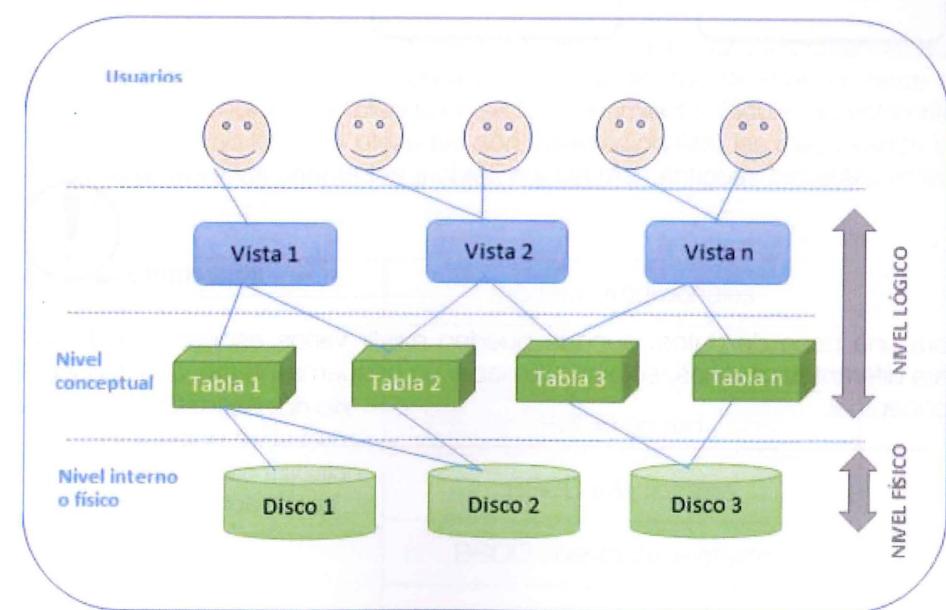
- Tiene que existir una separación entre los datos y los programas de aplicación que hacen uso de esos datos
- Los usuarios han de poder manejar diferentes vistas
- El SGBD ha de usar un catálogo que almacene el esquema definido para la base de datos.

Teniendo en cuenta estos tres principios, en 1975, el comité ASI-PARC, propuso en base a ellos una arquitectura para los SGBD de tres niveles, para poder separar la BBDD física de los programas de aplicación. Así, en las ar-

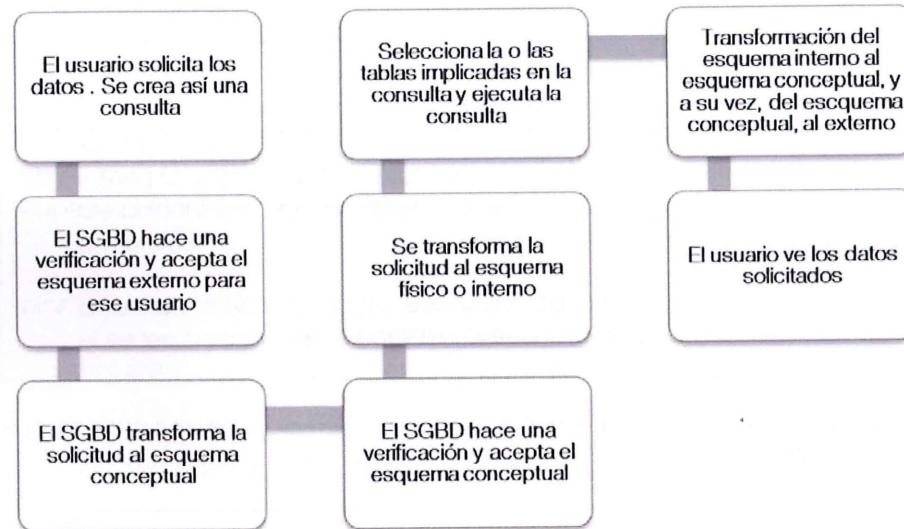
quitecturas de las BBDD nos vamos a encontrar con tres niveles: uno interno o físico, uno externo o de visión y otro conceptual.

Nivel físico o interno	Describe, a través de su esquema interno, la estructura física de la BBDD Describe que archivos contienen la información, cómo se organizan, la manera de acceder a los registros, su longitud...
Nivel de visión o externo	Es el más cercano a los usuarios y donde se describen las diferentes vistas de usuario
NIVEL CONCEPTUAL	Describe la estructura de toda la BBDD para un determinado grupo de usuarios mediante un esquema conceptual. Describe entidades, atributos, relaciones, operaciones de usuario, restricciones

En la siguiente figura podemos ver la representación de esta arquitectura que propone ANSI:



Aunque en realidad los datos realmente sólo se encuentran en el nivel físico, almacenados en alguna unidad de memoria, esta arquitectura trata de hacer una descripción a los tres niveles mencionados. Los usuarios harán sus propias peticiones al sistema, a través del esquema externo, y el sistema gestor de bases de datos transformará dicha solicitud a una petición realizada en términos del esquema conceptual, para finalmente convertirse en una solicitud expresada en el esquema interno que llevará a ejecutar un proceso en la BBDD. Resumido, el flujo que siguen las peticiones sería pues:



Para una base de datos, aunque pueden existir varios esquemas externos para diferentes usuarios, sólo puede haber un esquema interno y un esquema conceptual.

Decíamos que el origen de esta estructura a tres niveles tenía que ver con independizar los datos de las aplicaciones que hacen uso de ellos, así pues, con esta arquitectura, aparece explícito el concepto de **INDEPENDENCIA DE LOS DATOS**, encontrándonos con dos tipos:

- **Independencia física**, que va a permitirnos hacer una modificación en el esquema interno sin por ello tener que modificar ni el esquema externo, ni siquiera el esquema conceptual.
- **Independencia lógica**, que va a permitirnos hacer una modificación en el esquema conceptual sin tener que hacer una modificación en el esquema externo ni en ninguno de los programas de la aplicación.

2.2.1. Definición de diferentes sistemas de bases de datos

De la misma manera que cuando tenemos que hacer un regalo, y comprar un reloj, nos encontramos que en la tienda hay disponibles más de una marca, que además cada marca tiene varios modelos, y que incluso podemos elegir nuestro regalo en función de la tecnología de funcionamiento que queremos que tenga nuestro reloj, por ejemplo, si queremos un reloj con pilas, o lo que buscamos es un reloj con cuerda, cuando tenemos que seleccionar un base de datos para nuestro sistema gestor de datos también tenemos múltiples opciones, cada una asociada a un tipo distinto de necesidades y requerimientos.

Aunque las bases de datos podrían clasificarse según múltiples criterios, aquí vamos a centrar nuestra atención a la clasificación que de ellas se hace en función de la tecnología empleada en su funcionamiento. Siguiendo este criterio veremos como las más utilizadas son las relacionales, las más avanzadas las declarativas y las orientadas a objetos y las más antiguas las jerárquicas o en red.

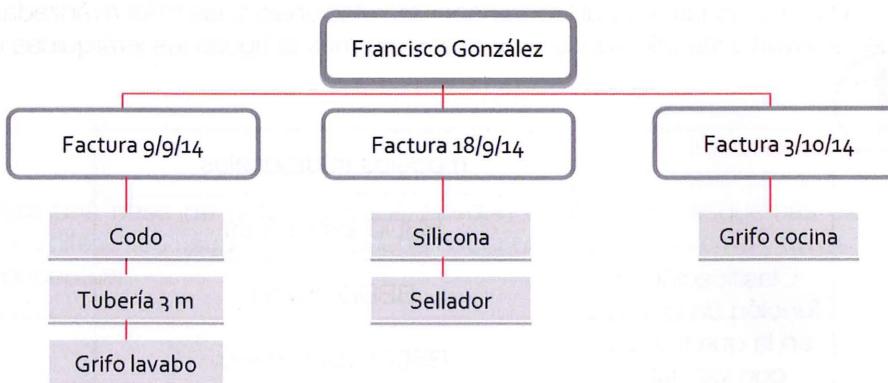
Clasificación en función de la forma en la que trabajan con los datos	modelos tradicionales
	BBDD jerárquicas
	BBDD en red
	BBDD relacionales
	BBDD orientadas a objetos
	BBDD declarativas

Modelo jerárquico

Este tipo de bases de datos es la utilizada por los primeros gestores de datos que aparecieron. El objetivo de estas bases de datos es establecer unas fichas ordenadas jerárquicamente, de tal manera que cada una de estas fichas, a su vez, pueda contener una lista de más fichas y así sucesivamente. Si implementamos esta definición de forma gráfica, veremos como los datos quedan organizados en una estructura de árbol, de ahí que estas bases de datos también sean conocidas como modelo de árbol.

Por ejemplo, si una empresa de venta de suministros de fontanería tiene modelada su base de clientes con este modelo de árbol, cada ficha de clientes a su vez, contendrá un listado de las facturas que se le han hecho, y a su vez, las facturas tendrán un listado de ítems que contendrán los suministros facturados.

 Sabías que
El sistema gestor de bases de datos de modelo jerárquico más conocido es el sistema IMD de IBM.



Ejemplo de registro en modelo jerárquico BBDD

Como puedes ver si observas el ejemplo, la información está organizada de tal manera que las entidades de la base de datos siempre mantienen una relación de tipo padre-hijo, está organizada en orden jerárquico.

En este modelo de bases de datos hay tres conceptos fundamentales de los cuales debemos comprender su significado para poder entender el modelo. Estos conceptos son:

- Tipo de registro
- Instancia u ocurrencia de registro
- Relación padre/hijo



Definición

El **tipo de registro** va a definir la estructura que tiene que tener cada registro, o lo que es lo mismo, deberá definir los campos que va a tener cada uno de sus segmentos y cual va ser la estructura jerárquica que entre ellos van a mantener.

La **instancia u ocurrencia de registro** es el valor de un tipo de registro.

La **relación padre/hijo**. Esta relación determina que el camino de acceso a los datos sea único.

Si no estás del todo familiarizado con estos conceptos puede ser difícil comprenderlos completamente, pero si lees con detalle el siguiente ejemplo seguro que te queda completamente clara la diferencia entre tipo e instancia de registro:

Ejemplo diferencia entre tipo e instancia de registro

- Un tipo de registro puede ser, por ejemplo, un tipo de fruta: Frutas de grano, Cítricos, Exóticas
- La instancia u ocurrencia del registro es una fruta concreta perteneciente a cada uno de esos grupos: naranja, manzana y mango

El ejemplo de la base de datos de la empresa de suministros mostraba un registro de la BBDD. El segmento que se encuentra en la cabeza de un registro se le llama segmento padre, en el caso del ejemplo, sería el cliente. Se llaman segmentos hijo a los que dependen de él. En el caso del ejemplo serían las diferentes facturas.

¿Cómo funciona el acceso a los datos en este tipo de BBDD? Definir la esquema de una base de datos jerárquica supondrá tener en cuenta los siguientes conceptos:

- Para desplazarnos en un registro, lo primero que hay que hacer es posicionarse en la raíz de una instancia, y posteriormente, ir desplazándose por los diferentes hijos según nuestras necesidades de consulta o modificación de datos.
- En una BBDD jerárquica no se puede acceder directamente a las instancias de un segmento hijo si con anterioridad no se han seleccionado las instancias de los segmentos padre de los cuales depende.

Si seguimos tomando como referencia el ejemplo de la empresa de suministros, no podríamos seleccionar una factura, si previamente no hemos solicitado un registro.



Estructura jerárquica, también llamada estructura en árbol

Resumimos a continuación las principales características y restricciones de un modelo jerárquico.

Características de las BBDD jerárquicas

El árbol se organiza en un conjunto de niveles

El nodo raíz, que es el más alto de toda la jerarquía, será el nivel 0

Las líneas que unen los diferentes nodos se llaman Caminos

Un hijo sólo puede tener un parente, y un parente puede tener varios hijos. Es decir, a un nodo de nivel inferior únicamente le puede corresponder un nodo de nivel superior

Todos los nodos tienen que tener un parente, sacando de esta norma al no raíz

HOJAS son las ramas sin hijos

ALTURA es el nº de niveles de la estructura

MOMENTO es el nº de nodos

PESO es el nº de hojas del árbol

FAMILIA, es la suma total de un nodo parente y sus hijos.

El árbol es una estructura recursiva, es decir, un nodo terminal y sus hijos forman un subárbol

Restricciones de las BBDD jerárquicas

Cada árbol tiene un segmento raíz único

Hay una única relación entre dos segmentos de un árbol

Esta estructura no permite que existan relaciones reflexivas de un segmento consigo misma

Un hijo no puede tener más de un parente

A cualquier información almacenada hay que acceder a través de la raíz del árbol

El camino que el árbol debe recorrer es siempre el camino jerárquico preestablecido

Una vez esté creada, la estructura del árbol no puede ser modificada.

Modelo en red

Este modelo nos va proporcionar más potencialidad que el modelo jerárquico, pero aún se va a quedar algún paso por detrás del modelo relacional. Al igual que en el modelo anterior, la información en el modelo en red se organiza en registros, que almacenarán los datos y que llamaremos nodos. Estos nodos estarán relacionados unos con otros a través de enlaces.



Importante

La principal diferencia entre un modelo jerárquico y un modelo en red es que en los modelos en red puede haber más de un padre.

Nodos + Enlaces = Modelo en red

En este modelo son 3 los conceptos fundamentales que se tienen que tener en cuenta al hacer la definición de un esquema asociado a una base de datos en red: registro, elementos o campos de datos y conjunto:

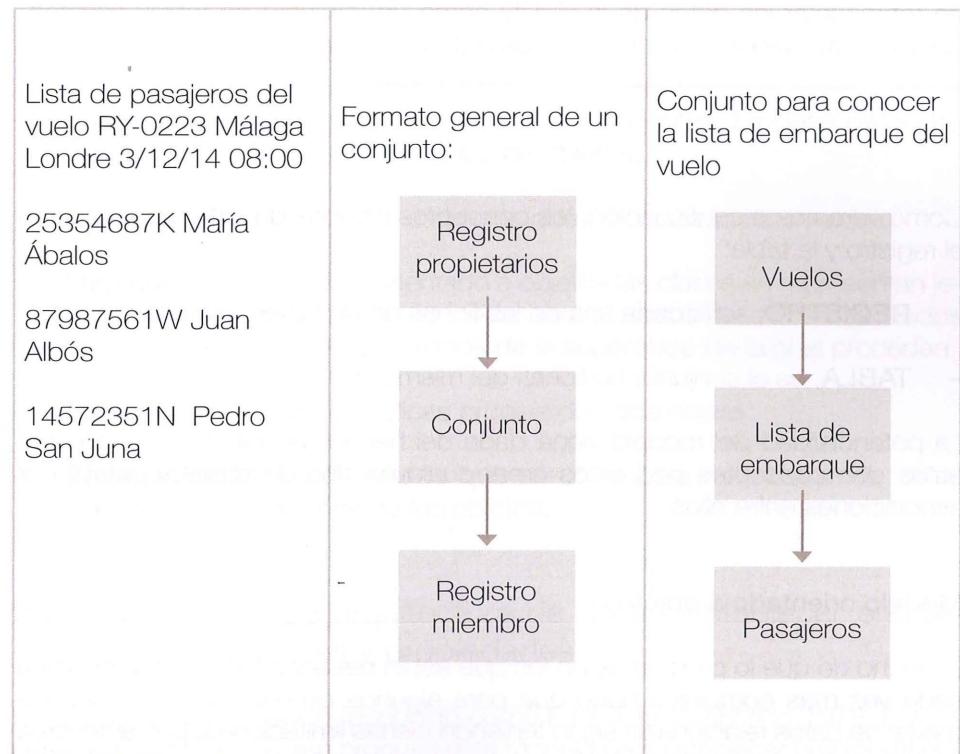
- **REGISTRO**, sería cada una de las fichas que se almacena en un fichero al uso. Un **tipo de registro** representa un nodo o dicho de otra forma, un conjunto de datos.
- **ELEMENTOS o CAMPOS DE DATOS**, son cada uno de los diferentes apartados que componen un registro.
- **CONJUNTO**, es el concepto a través del cual se relacionan distintos registros entre sí. Una ocurrencia de conjunto va a estar formada por un registro propietario más varios registros miembros. Una ocurrencia de conjuntos no es posible que pertenezca a varias ocurrencias del mismo conjunto.
- **DIAGRAMA**, es el esquema que va a representar el diseño de una BBDD en red.

Importante



La principal restricción del modelo en red para las BBDD es que una ocurrencia de registro miembro únicamente puede pertenecer a una única instancia de un determinado conjunto, lo que no significa, que pueda ser capaz de participar en varios tipos de varios conjuntos.

Mira este ejemplo sobre una ocurrencia de conjunto:



Por último para terminar con la definición de las BBDD en red tenemos que poner de manifiesto que el tipo que tenga un registro propietario no puede ser el mismo tipo que el registro miembro. Dicho de otra forma, tenemos que tener en cuenta, que un registro, dentro de un conjunto, no puede participar al mismo tiempo como propietario y como miembro.

Modelo relacional

En este punto solamente vamos a dar unas pinceladas de este modelo, ya que a continuación vamos a dedicarle un apartado completo, puesto que, en la actualidad es el modelo más potente y más extendido para la representación de datos.



El modelo relacional, también conocido como modelo entidad-relación, trata de representar los datos de la BBDD como un conjunto de tablas bidimensionales, que van a contener tanto los datos como las relaciones entre ellas.

Como veremos a continuación, los conceptos básicos de este modelo serán el registro y la tabla:

- **REGISTRO**, sería cada una de las fichas de un fichero clásico
- **TABLA**, es el conjunto de fichas del mismo tipo

La potencialidad del modelo viene dada del hecho de que con únicamente estos dos conceptos podremos crear cualquier tipo de datos y establecer asociaciones entre ellos.

Modelo orientado a objetos

El hecho de que los programas en los que están basadas las aplicaciones sea cada vez más complejos hace que para algunos de estas aplicaciones las bases de datos relacionales sigan teniendo ciertas limitaciones. Los anteriores modelos, principalmente en algunos campos de aplicación no permiten que sean aplicables, hablamos de áreas tales como la multimedia, el diseño asistido por ordenador, etc...

Este modelo está basado en la POO (programación orientada a objetos).



A los sistemas gestores de datos que utilizan BBDD orientadas a objetos se les denomina SGBO (Sistemas de gestión de bases de objetos) y gestionan diferentes objetos en los que están encapsulados los datos y además gestionan las operaciones que interactúan con ellos.

En los SGBO, el modelo de datos que presentan es único, las aplicaciones pueden acceder directamente al modelo y no existe diferencia entre el modelo conceptual y el modelo lógico.

Serán conceptos importantes en este modelo:

- **Clase**. Es el grupo formado por objetos semejantes. La clase define las características generales del grupo de objetos.
- **Mensaje**. Cada uno de los estímulos que son enviados a un objeto
- **Herencia**. En un modelo orientado a objetos las clases se representan jerárquicamente, hay superclases y subclases. Las subclases heredan todos los atributos, mensajes y métodos de la superclase de la cual proceden.
- **Estado**. Son las características propias de cada objeto
- **Encapsulación**. Es el método que se utiliza para no mostrar de forma evidente ciertos detalles de los objetos.

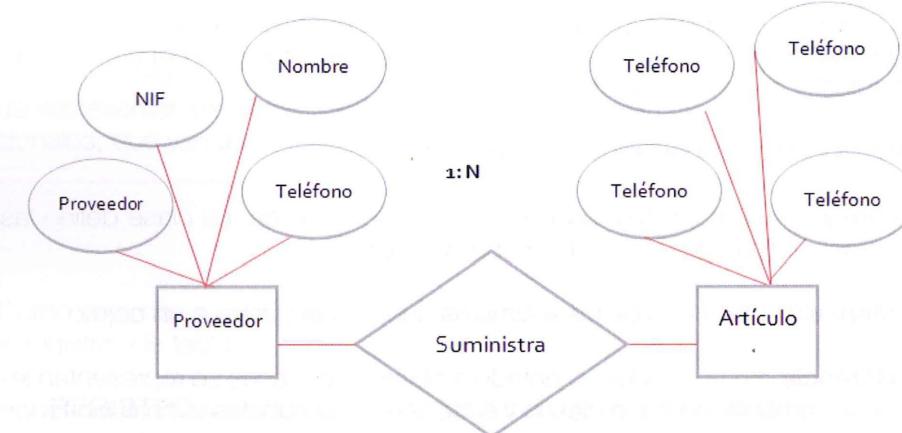
2.2.2. Definición de arquitectura de datos: establecer entidades, relaciones y propiedades

El modelo entidad-relación (conocido como E-R) data ya del año 1976. Entonces, fue Peter Chen, quien propuso este modelo para establecer una representación a nivel conceptual que pudiera reflejar lo que ocurría en el mundo real.

Este modelo en la actualidad es el más extendido y el más potente para llevar a cabo la representación de los datos. La extensión de su uso se deriva del hecho que el ANSI determinará que este iba a ser el modelo que se iba a estandarizar para todos los sistemas de diccionarios de recursos de información.

Importante

El modelo de bases de datos relacional, o modelo entidad-relación, o modelo E-R, utilizará tablas para representar los datos y las relaciones existentes entre ellos, así como grafos y símbolos gráficos.



Algunos conceptos son básicos para poder determinar la estructura de los datos en un modelo de datos relacional, para poder establecer de forma correcta las entidades, las relaciones y las propiedades. Estos conceptos básicos son:

- Entidad
 - Son los objetos del mundo real que van a tener interés para la organización
 - Su representación es un rectángulo con el nombre en el interior
- Conjunto de Entidades
 - Es un grupo de entidades del mismo tipo.
 - Puede haber entidades que pertenezcan al mismo tiempo a varios conjuntos de entidades.

- Entidad Fuerte
 - Entidad que no depende de otra para existir
- Entidad Débil
 - Entidad que depende de otra para poder existir
- Atributos o Campos
 - También se llaman campos. son las unidades de información que describen propiedades de las entidades. Los atributos toman valores.
 - Se representan con una elipse con el nombre en el interior.
- Dominio
 - Es el conjunto de valores que puede tener cada atributo
- Identificador o Superclave
 - También llamado identificador, es el conjunto de atributos que identifica de forma única a cada entidad.
- Clave Candidata
 - Se llama así a cada una de las superclaves formadas por el mínimos nº de campos posibles
- Clave Primaria o Clave Principal (Primary Key)
 - Es la clave candidata que selecciona el administrador para identificar cada una de las entidades.
- Clave Ajena o Clave Foránea (Foreign Key)
 - De una entidad, es el atributo o conjunto de ellos, que forman la clave primaria de otra entidad
- Relación
 - Es la asociación que se establece entre varias entidades

- Conjunto de Relaciones
 - Se llama así al grupo de relaciones del mismo tipo
 - Una relación también puede tener atributos descriptivos

Probablemente el contenido de la siguiente tabla te ayude a clarificar los conceptos que acabamos de exponer, ya que contiene algunos ejemplos muy explícitos, que relacionan los diferentes conceptos con hechos habituales del mundo real:

CONCEPTO	EJEMPLO
Entidad	CLIENTE en una empresa; ALUMNO, en un colegio
Conjunto de Entidades	ALUMNOS en un colegio
Entidad Fuerte /Débil	ALUMNO es una entidad fuerte, y NOTAS una entidad débil, puesto que necesita de la entidad ALUMNO para existir
Atributos o Campos	La entidad ALUMNO tiene como atributos el nombre, la dirección, el teléfono. Los atributos toman valores, el valor del atributo teléfono de la entidad alumno podría ser 656666555
Dominio	Por ejemplo el dominio del atributo "teléfono" será una cadena numérica de 9 dígitos.
Superclave o Identificador	En la entidad ALUMNO el Identificador podría ser el DNI, no habrá 2 alumnos con el mismo DNI
Clave Candidata	Para el caso del alumno podrían ser el DNI y el nº de la Seguridad Social
Clave Principal	Siguiendo con el ejemplo, el administrador podría elegir como clave principal el DNI

Ejemplos conceptos clave en BBDD relacionales

Para poder definir la estructura de las bases de datos, además de los conceptos claves y básicos que acabamos de ver, hay otros términos relativos a estas estructuras entidad-relación que también debemos conocer para poder definir de manera adecuada el esquema de una BBDD relacional. Nos estamos refiriendo a las RELACIONES y los CONJUNTOS de RELACIONES.



Llamaremos **relación** a la asociación que se establece entre diferentes entidades.

¿Qué deberíamos saber respecto a las RELACIONES?

Gráficamente la representaremos mediante un rombo

Las relaciones, de forma general, no tienen atributos

Si en la definición nos encontráramos una relación con atributos, tendríamos que pensar en que quizás debajo de ella hay una entidad que aún no ha sido definida (**entidad asociada**)

La entidad asociada será el origen de una nueva tabla que va a albergar los atributos de la misma.

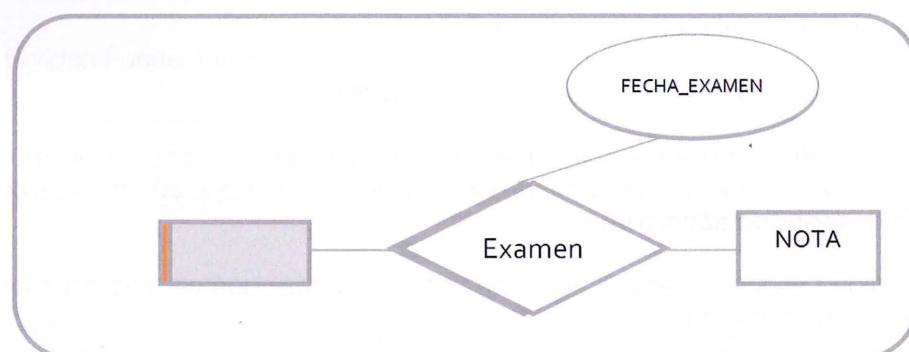


Llamaremos **conjunto de relaciones**, al grupo de relaciones que son del mismo tipo.

Lo normal es que los conjuntos de relaciones estén formados por dos entidades, aunque simplifica la estructura en gran medida el hecho de que los conjuntos de relaciones sean binarios. En la mayoría de los casos es posible transformar un conjunto de relaciones de más de dos entidades en varios conjuntos de relaciones binarias.

- Conjunto de relaciones binarias: ALUMNOS y NOTAS, todas las asociaciones que existan entre los alumnos y las notas van a formar un conjunto de relaciones.
- Conjunto de relaciones no binarias: CLIENTE, SUCURSAL y CUENTAS, todas las relaciones que existan entre el cliente, sus cuentas y la sucursal donde las tiene formarán un conjunto de relaciones.

Las relaciones no existirían sin las entidades, llamaremos **PAPEL**, a la función que cumple una entidad en una relación. Así mismo debemos hacer notar que las relaciones pueden tener atributos descriptivos, por ejemplo en el conjunto de relaciones ALUMNO_NOTAS, la FECHA_EXAMEN, nos va a especificar la última fecha en la que el alumno obtuvo un resultado en una prueba académica.



Ejemplo atributos descriptivos en una relación

Para poder determinar la estructura de una base de datos, dibujar la estructura gráficamente ayudará en gran medida. Los diagramas que los modelos relacionales utilizan para estructurar sus datos tienen ya unos símbolos determinados. Dichos símbolos están especificados en el siguiente esquema:

Símbolos estructura de datos en modelo E-R

La entidades se representan con rectángulos

Los atributos se representan con elipses

Las acciones se representan con rombos

Las líneas simples unirán los atributos con las entidades y los atributos con las relaciones

En una relación, las líneas pueden ser flechas. La orientación de la flecha va a señalar la cardinalidad de la relación.

Todos los componentes se van a etiquetar con el nombre que lo representan.

Hemos avanzado ya que la cardinalidad de la relación estaba determinada por la orientación de la flecha que unen atributos y entidades.

Para avanzar en el establecimiento de relaciones veamos a que llamamos **GRADO** de una relación.



Llamaremos **grado** de una relación al número de conjuntos de entidades que van a formar parte del conjunto de relaciones, o, dicho de otra forma, al número de entidades que van a formar parte de una relación.

Ya adelantamos antes que lo ideal es que el conjunto de entidades esté formado por dos y éstas sean binarias, pero también puede haber relaciones ternarias y reflexivas:

GRADO de las relaciones		
Relaciones Binarias (de grado 2)	Relaciones ternarias (de grado 3)	Relaciones reflexivas (de grado 1 o anillo)
Aquellas en las que participan 2 entidades	Aquellas en las que participan 3 entidades	Aquellas en las que participa únicamente una entidad

Para más información, consulta Ejemplos de relaciones binarias, ternarias y reflexivas en el anexo al final del libro.

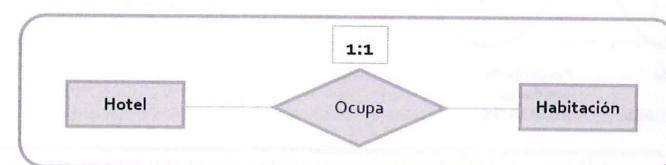
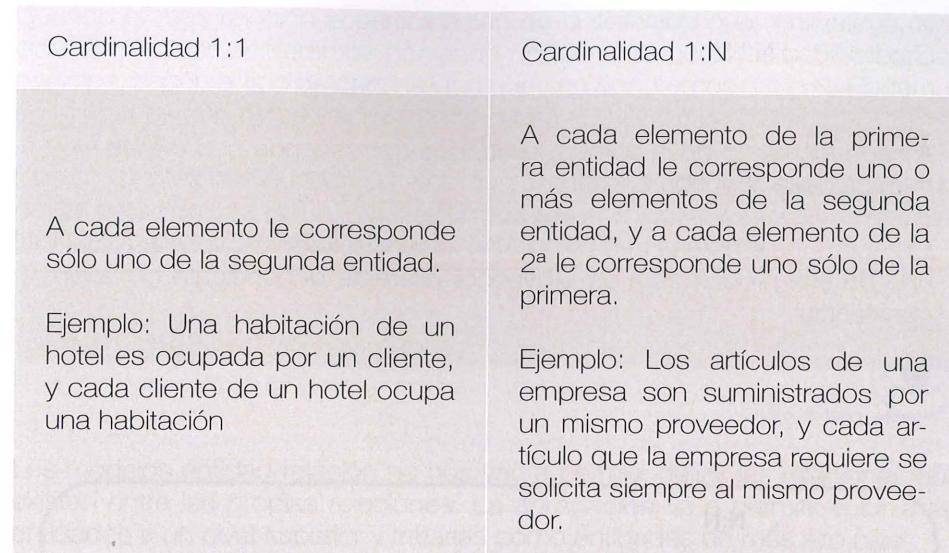
Como ocurre en la mayor parte de los modelos, también existen ciertas restricciones a la hora de aplicarlos. En el caso del modelo entidad-relación esta restricción a la que deben ajustarse los datos va a venir dado por **las cardinalidades de asignación**.



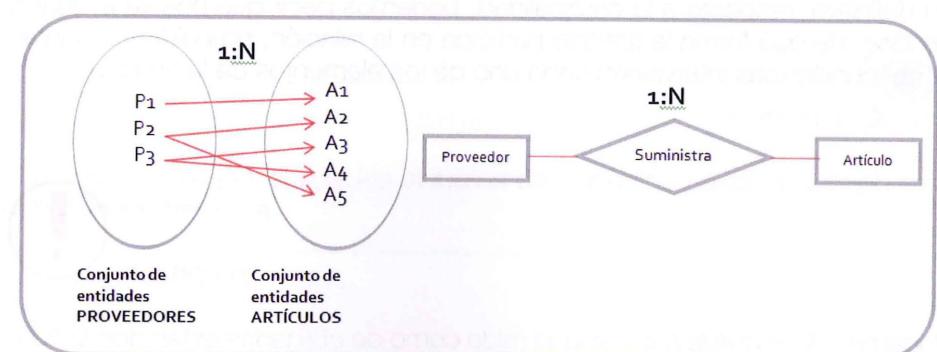
Las cardinalidad de asignación, únicamente aplicable para los conjuntos binarios de aplicaciones, va a definir el número de entidades a las que una entidad puede asociarse mediante un conjunto relación.

Las cardinalidades de asignación podrán ser

- Uno a uno (1:1)
 - Uno a muchos (1:N)
 - Muchos a muchos (N:N)



Ejemplo relación cardinalidad uno a uno, 1:1

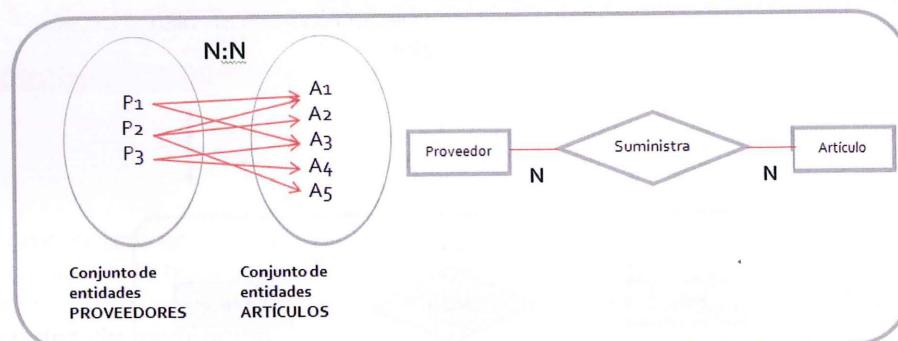


Ejemplo relación cardinalidad uno a muchos , 1:N

Cardinalidad N:N

A cada elemento de la primera entidad le corresponde uno o más elementos de la segunda entidad

Ejemplo: Una empresa trabaja con varios suministradores y compra cada uno de sus productos a un proveedor diferente dependiendo de la circunstancia



Ejemplo relación cardinalidad muchos a muchos , N:N

En definitiva, respecto a la cardinalidad, podemos decir que nos va ayudar a conocer de qué forma la entidad participa en la relación, nos dirá en cuantas correspondencias intervienen cada uno de los elementos de la entidad.

Importante

Podemos decir que la cardinalidad mide como de obligadas están dos entidades a corresponderse entre sí.

Cuando en una relación la participación de la entidad no es obligatoria, sino opcional, lo representaremos poniendo un "0" entre paréntesis. Si la entidad participa, según el tipo de cardinalidad que sea pondremos o 1 o N. Podemos encontrarnos con estos valores: (M:N), (1:N), (0,N), (0,1) o (1;1).

Dos conceptos nos quedan por analizar para terminar de definir la arquitectura de los datos en una BBDD. Nos referimos a la AGREGACIÓN y a la GENERALIZACIÓN y las JERARQUÍAS de generalización.

Definición

Los modelos entidad-relación no nos van a permitir definir las relaciones que existen entre las propias relaciones. La **agregación** va a permitir subir a las entidades a un nivel superior y tratarlas como entidades de más alto nivel

En cuanto a las generalizaciones no es sino un mecanismo de conceptualización que nos va a permitir que una entidad, que llamaremos supertipo, la especialicemos en subtipos.

¿Cuándo será necesario que utilicemos la generalización? Por ejemplo en el caso de que un conjunto de entidades tenga, por un lado, un conjunto de atributos que serán comunes al conjunto, y por el otro, ciertos atributos específicos. Los atributos particulares van a describir el subtipo y los comunes van a describir el supertipo.

Definición

El procedimiento por el cual los atributos del supertipo pasan al subtipo será denominado **herencia**.

El siguiente ejemplo te va a ayudar a entender los conceptos que acabamos de definir.

En un hospital se identifican las siguientes entidades:

- Empleado (Nombre, Nº Empleado, Puesto, Salario)
 - Médico, que además de incluir los atributos del empleado incluirán más específicos: Turno, Especialidad
 - Enfermera, que además de incluir los atributos del empleado incluirán más específicos: Años de experiencia, Quirófano
 - Administrativos, que además de incluir los atributos del empleado incluirán más específicos: Departamento_Soporte

La generalización puede ser de tres tipos, podremos tener una generalización total, o que sea parcial, o bien exclusiva:

Generalización		
Total	Parcial	Exclusiva
Si no hay ocurrencias en el supertipo que no pertenezcan a ninguno de sus subtipos	Si existe alguna entidad que no pertenece a ningún subtipo	Si cada entidad sólo puede pertenecer a un subtipo, y no a más de uno

2.3. Procesos de instalación de un gestor de datos para albergar sistemas ERP y CRM

A continuación vamos a especificar como proceder a la hora de llevar a cabo la instalación del Sistema Gestor de Bases de Datos.

El primer paso de la instalación, y antes del inicio del proceso propiamente dicho, es tener claro que necesidades hemos de cubrir con el sistema gestor. Analizar pues los requerimientos de nuestro ERP y/o CRM será el paso fundamental del proceso.

Podremos diseñar un SGBD potente y con unas características perfectas, pero si no cubre las necesidades que requiere nuestro ERP y/o CRM no servirá de nada.



Importante

Lo fundamental, a la hora de instalar el SGBD es plantearnos las necesidades, a fin de encontrar la solución óptima que resuelva de forma eficaz y eficiente nuestras necesidades.

Los pasos que seguiremos en la instalación del SGBD y que iremos desarrollando a continuación van a ser:

- Selección de requisitos
- Comprobar requerimientos sistema
- Obtención de documentación instalación
- Iniciar proceso de instalación sobre el equipo

Pues vamos a ello... lo primero será pensar sobre qué es lo que en realidad vamos a necesitar del SGBD. Para encontrar el SGBD que se adapta a nuestras necesidades tendremos que tener en cuenta: el tamaño de la base de datos que va a requerir nuestro ERP, la conectividad que vamos a necesitar, el número de usuarios que van a hacer uso del ERP y/o CRM, cuántas conexiones simultáneas se van a poder establecer, si estamos dispuestos o no a variar el hardware que ya posee la empresa e incluso la misma política de empresa a cuanto a paquetes de software se refiere.

Aspectos a valorar en la elección del SGBD		
Tamaño de la BBDD	Conectividad	Nº de usuarios
Nº de conexiones simultáneas	Aprovechamiento del software	Política de empresa

Tamaño de la BBDD

Esta es una cuestión muy a tener en cuenta, ya que, por ejemplo, utilizar una base de datos muy grande va a requerir que, si no tenemos espacio suficiente en las unidades de almacenamiento, tengamos que separar físicamente los datos, con los problemas que esa situación puede conllevar. Por otra parte un BBDD grande va a requerir un software potente, y esto también habrá que tenerlo en cuenta.

Conectividad

Tendremos que valorar y tener en cuenta, desde donde vamos a querer o tenemos que poder, conectarnos a nuestro ERP, ¿será suficiente con conectarlos a través de un solo equipo de acceso? ¿O nos conectaremos a través de una intranet únicamente? O por el contrario, ¿debe estar conectada a internet para poder acceder a nuestro ERP en cualquier sitio o en cualquier lugar?

Nº de conexiones simultaneas

Con casi toda probabilidad ha de ser el requisito al que más atención le tendremos que prestar. Si nuestro ERP va a soportar un número alto de conexiones simultaneas necesitaremos un sistema gestor de bases de datos que posea potencialidad en el trabajo concurrente. No todos, de hecho muy pocos, lo permiten.

Aprovechamiento de recursos

Con la implantación del ERP en la empresa, ¿se está contemplando renovar también el hardware informático si el sistema así lo requiriera?, o por el contrario, ¿la empresa pretende utilizar el hardware que ya posee? Este hecho puede limitar en gran medida la elección de uno u otro SGBD

Política de empresa

En cuanto a software se refiere, la organización puede seguir una política de pretender, siempre que sea posible, software libre o, por el contrario, puede tener determinados acuerdos con determinadas empresas. Ambos casos condicionarán la elección del sistema gestor de bases de datos.

Teniendo en cuenta los aspectos anteriores, ya hemos seleccionado el SGBD que va a formar parte nuestro ERP y/o CRM. El siguiente paso será comprobar

que los requerimientos que el sistema gestor de bases de datos que hemos elegido exige pueden ser soportados por nuestra infraestructura. Así tendremos que comprobar que el sistema operativo que utiliza nuestro ERP y/o CRM es soportado por el SGBD, que además es compatible con otras aplicaciones ya instaladas en nuestros equipos, y que además tanto la memoria RAM, como los procesadores y los discos duros de nuestros equipos tienen la capacidad que el sistema gestor va a requerir. Así mismo los requisitos de red y las incompatibilidades que puedan tener han de ser ítems a comprobar.

Comprobar requerimientos infraestructura existente			
Sistemas Operativos	Aplicaciones preinstaladas	Memoria RAM	Incompatibilidades
Procesador	Disco Duro	Requisitos de red	

Sistemas Operativos

No todos los SGBD son capaces de trabajar con todos los sistemas operativos, de hecho, algunos son multiplataforma y otros no lo son y sólo son compatibles con alguna de las plataformas existentes.

Aplicaciones Instaladas

La instalación de un ERP se suele hacer en empresas que ya están en funcionamiento, que ya tienen implementados otros sistemas de información en sus sistemas, y por lo tanto, es algo habitual, que el sistema gestor de bases de datos tenga convivir con software ya instalado previamente a la instalación del sistema gestor.

Memoria RAM

Con toda seguridad, el principal requisito que ha de cumplir el sistema. Cuanta más capacidad de RAM disponible con más eficacia funcionará el SGBD

 Sabías que

Oracle aconseja tener disponible al menos, 1 GByte de RAM disponible antes de instalar su versión 11g

Disco Duro

Al igual que ocurría con la RAM, se requiere un "mínimo de espacio libre disponible en el disco duro, en el momento de la instalación del SGBD

Requisitos de Red

Puede ser un requisito que imponga el SGBD que, por ejemplo, el equipo donde va a instalarse tenga una IP fija, si nuestro equipo no es capaz de poder cumplir con este requisito no podremos instalarlo.

Requisitos de Red

Puede ser que el SGBD que hemos elegido exprese explícitamente que es incompatible con algún otro producto. Tendremos que comprobar que en nuestros sistemas no tenemos ninguno de esos productos instalados.

2.3.1. Instalación de un sistema gestor de datos

Para mostrar los pasos de la instalación de un sistema gestor de datos, hemos seleccionado el MySQL, ¿Por qué? Porque es la base de datos de código abierto que tiene mayor aceptación en la actualidad. De hecho, es el servidor de bases de datos relacionales más conocido.

Además es un programa Open Source, cualquier usuario puede descargarse desde Internet la aplicación y usarlo sin tener que pagar por ello. Es además seguro, rápido fácil de usar. Así que...vamos a ver cómo podemos instalar un SGBD en nuestro equipo.

- Los pasos a seguir, y que vamos a desarrollar a continuación serán:
- Comprobar hardware mínimo necesario
- Decidir la plataforma sobre la que lo vamos a instalar
- Decidir el formato de distribución, binaria o código fuente
- Instalar MySQL, o de forma manual o con el asistente
- Disponer de los manuales, o bien en línea o descargarlos
- Obtener los archivos de programa de la web
- Comprobar su funcionamiento
- Arrancar MySQL como un servicio
- Configurar los parámetros del SGBD

Antes de continuar con el proceso, vamos a hacer un apunte, y establecer la diferencia que hay entre elegir un formato u otro de distribución. Podremos instalar el SGBD utilizando una **distribución binaria**, o bien, **utilizar el código fuente**.



La **distribución binaria** va a ser la forma más sencilla y rápida de proceder a la instalación. Simplemente utilizaremos un instalador que ya está preparado para realizar esta función.

En la **distribución mediante código fuente** vamos a tener que realizar una compilación del código fuente del sistema gestor para que éste pueda funcionar.

Optar por hacer una distribución de código fuente, a pesar de complicar el proceso, puede darnos la ventaja de que nos va a permitir que el control que tengamos sobre los componentes a instalar sea mayor. Además podremos modificar los scripts y preparar ejecutables "a medida" de nuestro sistema.

MySQL es una aplicación que puede ser implementada sobre diversos sistemas operativos, aquí vamos a ver como se realizaría la instalación sobre Windows. Tenemos que hacer notar el hecho de que la estructura, o los pasos a seguir, para instalar un SGBD es muy similar, independientemente de la aplicación concreta de la que se trate. Todos los SGBD disponen de manuales de los que podemos hacer uso y en los que encontraremos los detalles concretos necesarios para realizar la instalación en cuestión.

INSTALACIÓN MySQL SOBRE WINDOWS

La opción disponible en la web nos ofrece dos opciones para instalarlo, podemos elegir entre la distribución binaria o la distribución de código fuente:

- La opción de distribución binaria disponible es, o bien, un archivo msi, que va a permitir iniciar de forma inmediata el servidor. Mediante esta opción la instalación se lleva a cabo prácticamente de forma automática. La otra opción disponible es la instalación desde un fichero comprimido zip. **Vamos a desarrollar, a modo de ejemplo, la instalación desde el archivo msi....**

Un archivo **msi**, Microsoft Installer, es un paquete de software que, por sí sólo, es capaz de instalar, mantener y eliminar programas de una plataforma Windows. Estos paquetes hacen que en un proceso de instalación la intervención del usuario sea mínima.

- Si seleccionamos la distribución por código fuente, ésta ha de ser compilada con VC++ 6.0

INSTALACIÓN CON EL PAQUETE msi

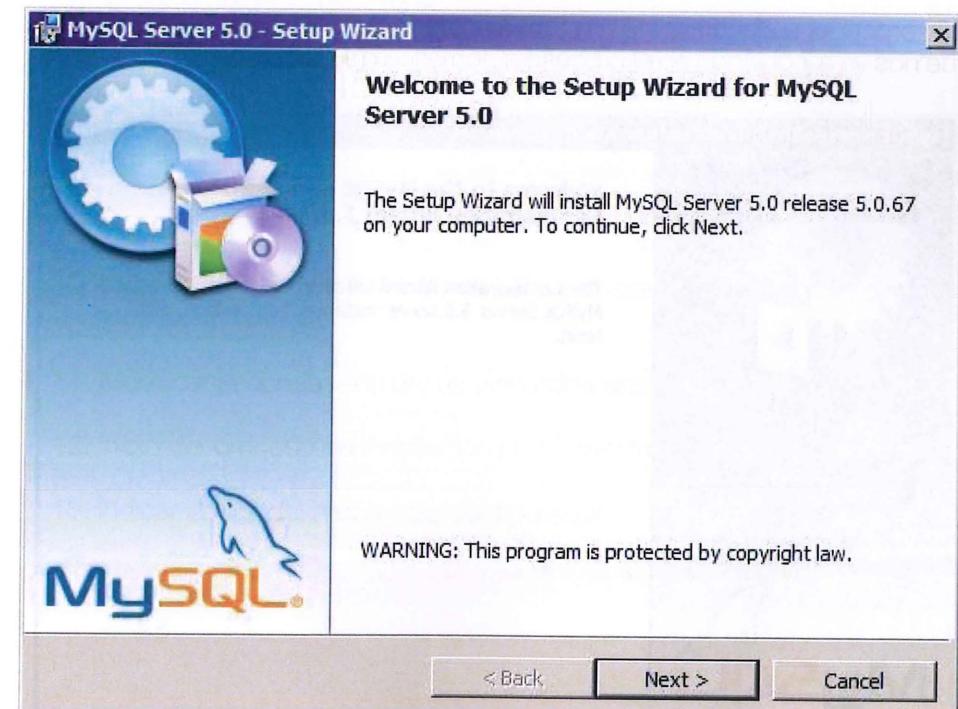
Elegiremos la opción de distribución binaria, si tenemos disponible un paquete msi que nos proporciona un proceso de instalación con asistente, ¿Por qué no hacer uso de él? Empecemos pues el proceso:

Instalación con el asistente	Instalar la instancia MySQL	Comprobar el funcionamiento de MySQL	Desinstalación con asistente
------------------------------	-----------------------------	--------------------------------------	------------------------------

- Al seleccionar la opción de distribución binaria con archivo msi, automáticamente nos aparecerá una ventana emergente, sobre la cual tendremos que aceptar la información inicial que se nos especifica y elegir el tipo de instalación que queremos realizar, la recomendada (que será la que Windows haga por defecto) o bien la personalizada. Como recomendación, la opción personalizada es más compleja.
- Una vez hayamos confirmado que queremos realizar la instalación, el paquete msi automáticamente modificará el registro de Windows, de tal manera, que en el **Menú de Inicio** aparecerá la nueva aplicación instalada y se creará una **carpeta dentro de la carpeta de Archivos de Programa de Windows** con el nombre del SGBD instalado

Ejemplo ruta carpeta instalación MySQL generada automáticamente

C:\Program Files\MySQL\MySQL 6.0



Pantalla Inicio Asistente Instalación automática del servidor MySQL

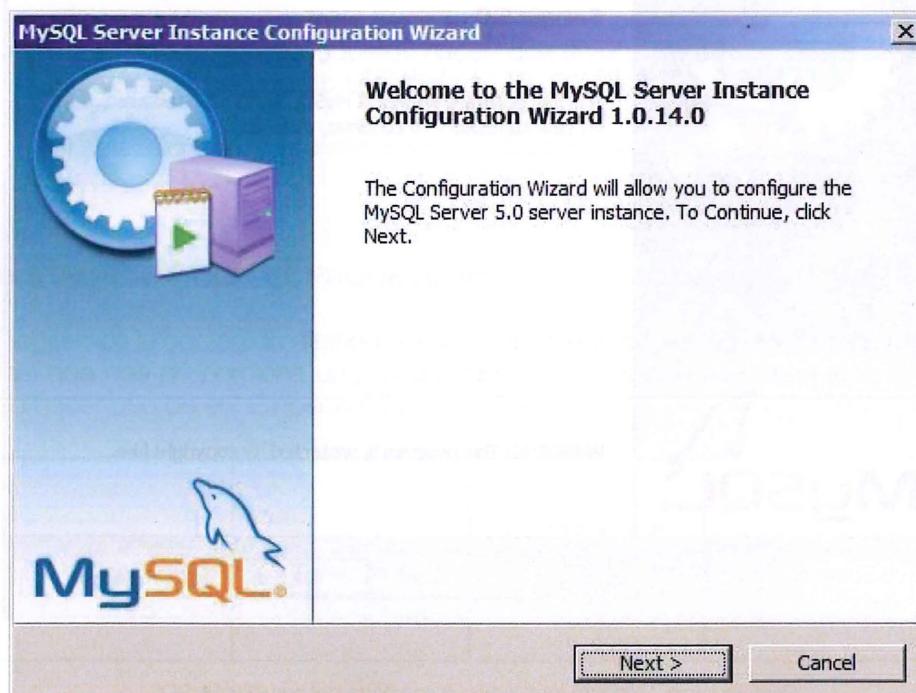
- El siguiente paso será realizar la instalación de la instancia de MySQL.



La **instancia** es el proceso de la BBDD que va a permitir que se lleve a cabo el acceso a la misma.

Podemos hacerlo de dos maneras, o bien seleccionar la opción en el último paso de la anterior opción, antes de confirmar la instalación, o como segunda opción tenemos la posibilidad de lanzarlo a través de un fichero del directorio. Si elegimos esta última opción, el programa a ejecutar (que estará situado en la **carpeta bin** del directorio antes mencionado que automáticamente ha generado Windows) será:

Tras proceder a ejecutar este programa nos aparecerá una pantalla emergente que nos va a ir guiando en el proceso automático de instalación de la instancia.



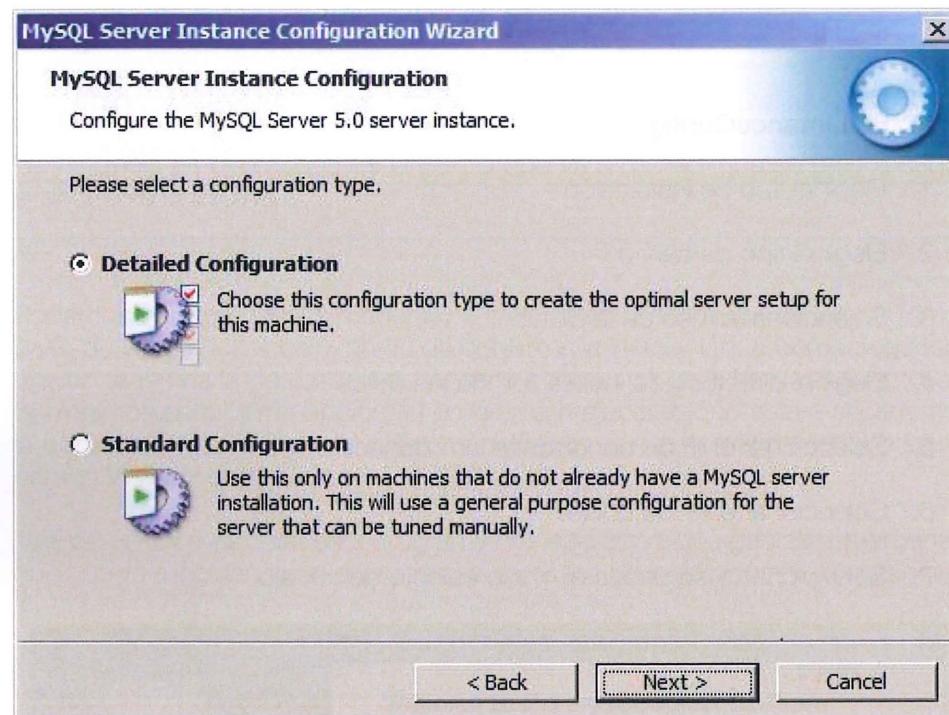
Pantalla Inicio del asistente para instalación automática de la instancia del servidor MySQL

El asistente nos va a ir dando diferentes opciones, tendremos que completar el siguiente proceso hasta finalizar la instalación

MySQLInstanceConfig

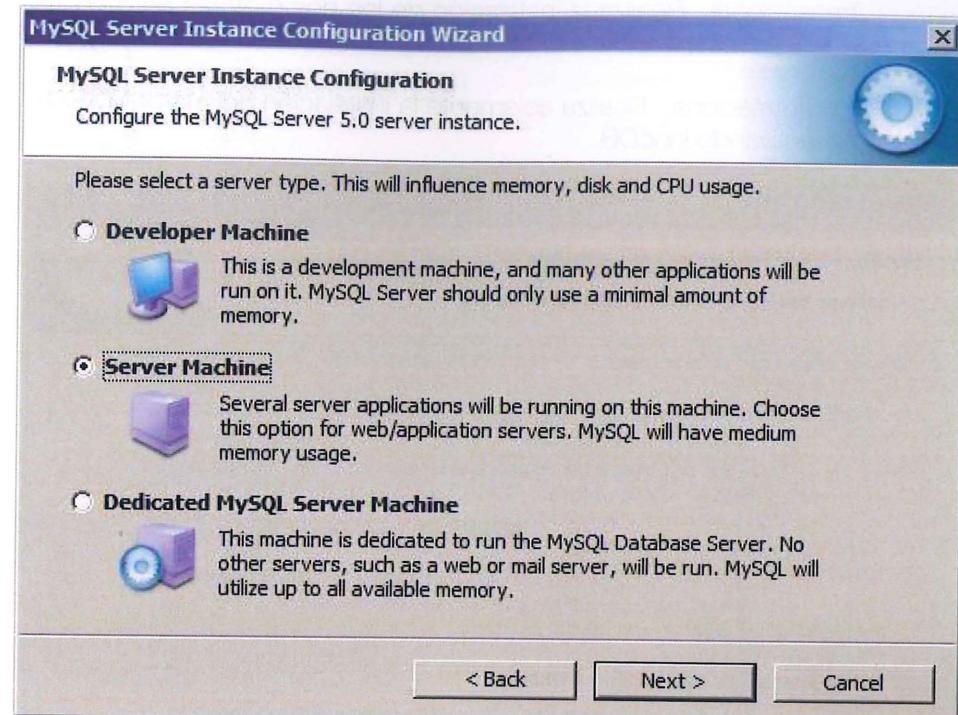
1. Elegir el tipo de instalación
2. Elegir el tipo de servidor
3. Seleccionar el uso de la BBDD
4. Elegir el directorio de instalación de los datos
5. Seleccionar el nº de conexiones simultaneas
6. Seleccionar el nº de puerto
7. Seleccionar si se desea el modo estricto de trabajo
8. Elegir el juego de caracteres con el que se codificarán los datos de la BBDD
9. Elección del modo de trabajo del servicio
10. Modificar el PATH de Windows para cambiar de ubicación el directorio bin
11. Modificar la contraseña del usuario administrador
12. Inicio del proceso de instalación propiamente dicho
13. Indicar la ruta del archivo de configuración

- Seleccionar el tipo de instalación. Podremos elegir entre la instalación estándar o la instalación detallada.



Pantalla selección Tipo de Instalación del asistente MySQL

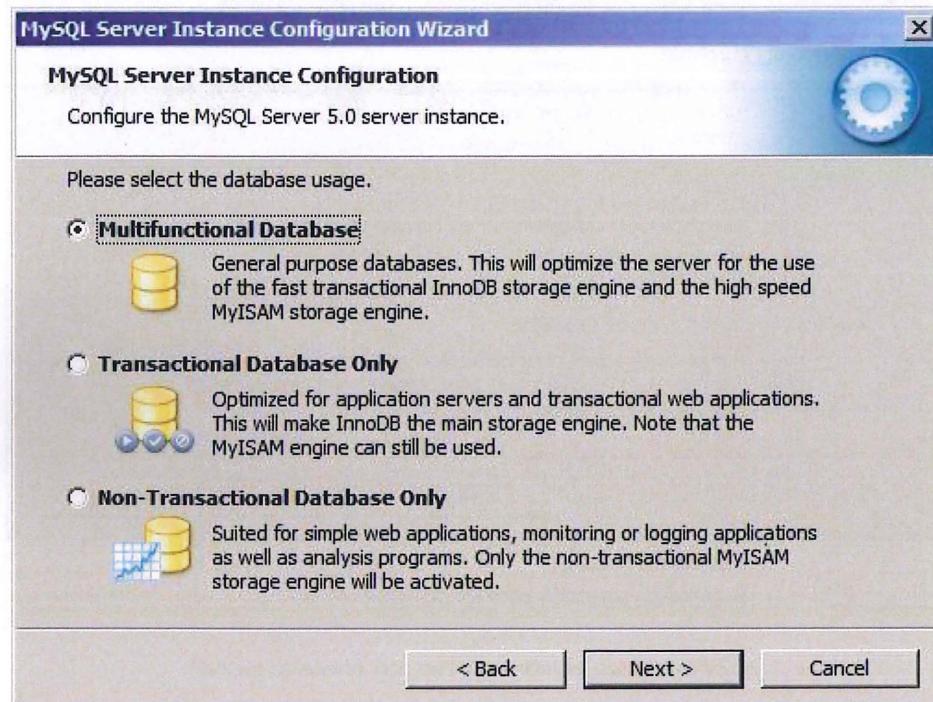
- Determinar cuál de los tipos de servidores disponibles queremos. Podremos elegir entre un servidor que consuma muy pocos recursos y que sea únicamente para uso individual, es decir, una máquina de desarrollo, o bien un servidor dedicado exclusivamente para MySQL o, como opción intermedia, seleccionar un servidor, pero que comparta espacio con, por ejemplo, los servidores de correo o los servidores web o ftp, de modo que consumamos una cantidad más moderada de los recursos disponibles.
 - Dedicated Machine: Servidor dedicado
 - Server Machine: Servidor compartido
 - Developer Machine: Máquina de desarrollo



Pantalla selección servidor del asistente MySQL

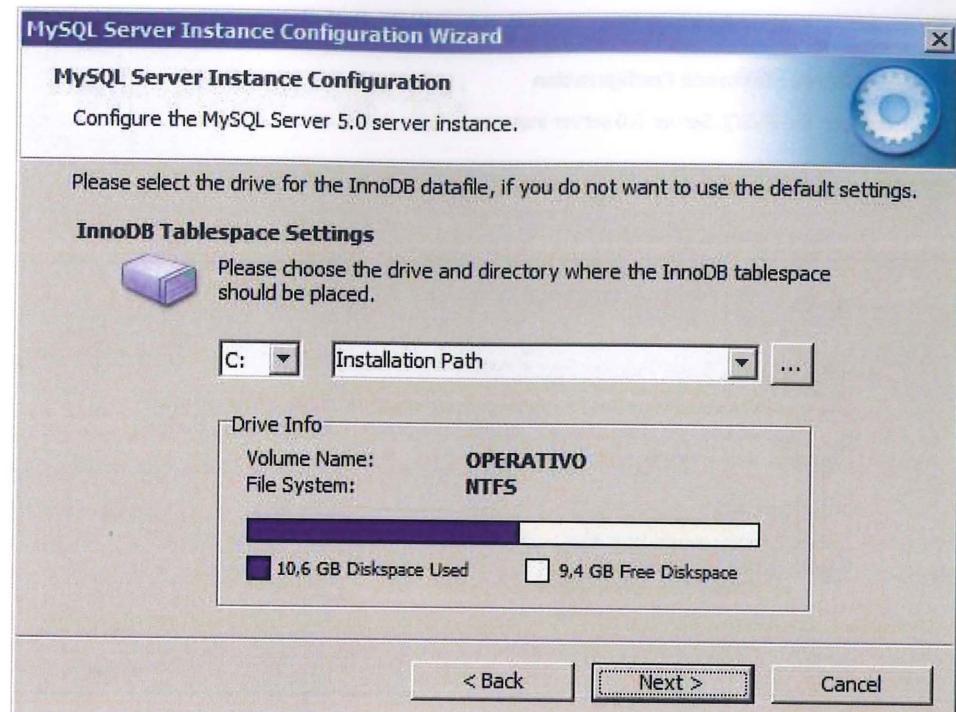
- El siguiente paso va a ser determinar cuál de los motores de búsqueda disponibles queremos usar en la BBDD. Los motores de búsqueda que posee MySQL son dos:
 - Motor de búsqueda InnoDB. Capacidad de manejar transacciones buena, pero relativamente lenta a la hora de añadir nuevos datos
 - Motor de búsqueda MyISAM. Más veloz que el motor InnoDB, pero como menor control sobre la integridad de los datos
- El asistente de configuración de la instancia de MySQL nos va a ofrecer tres opciones de configuración distintas:
- Multifuncional. Realiza la instalación de los dos motores y les confiere a ambos el mismo peso

- Transacional. Realiza la instalación de los dos motores pero prioriza el correcto manejo de las transacciones dándole prioridad a InnoDB
- No Transaciona. Realiza solamente la instalación del motor MyISAM, desactivando InnoDB



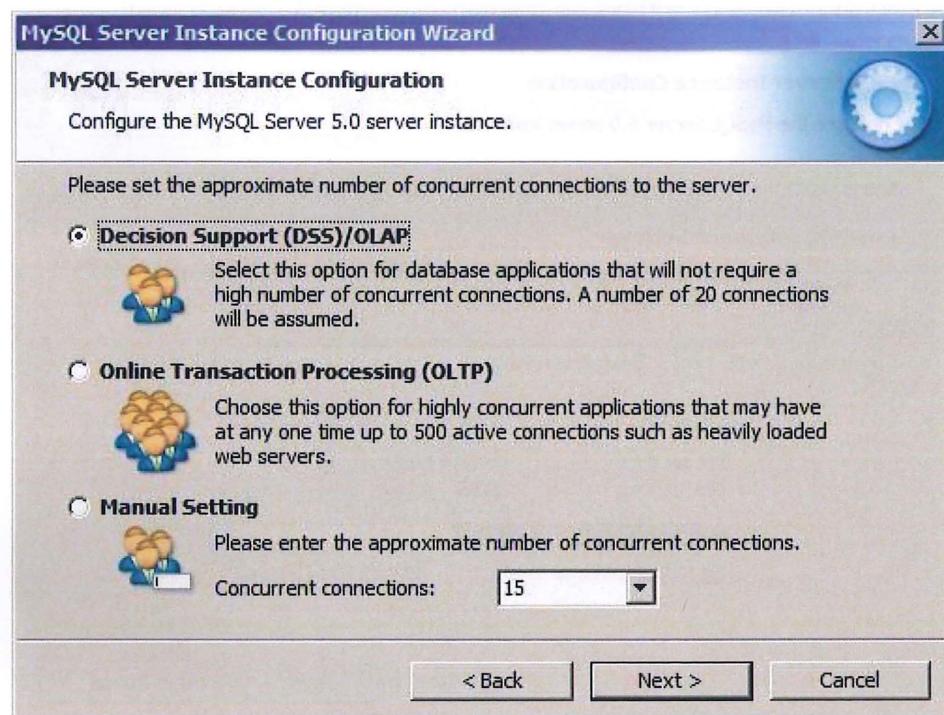
Pantalla selección motor de búsqueda del asistente MySQL

- El siguiente a deberemos seleccionar el directorio donde se van a ubicar físicamente los datos que va a contener la BBDD. En MySQL este punto del proceso se denomina ubicación de la tablespace. Si, y sólo si, estamos utilizando InnoDB podemos ubicar los datos en una ubicación distinta, incluso puede tratarse de otra unidad de disco, de donde se encuentran ubicados los archivos de programa de MySQL.

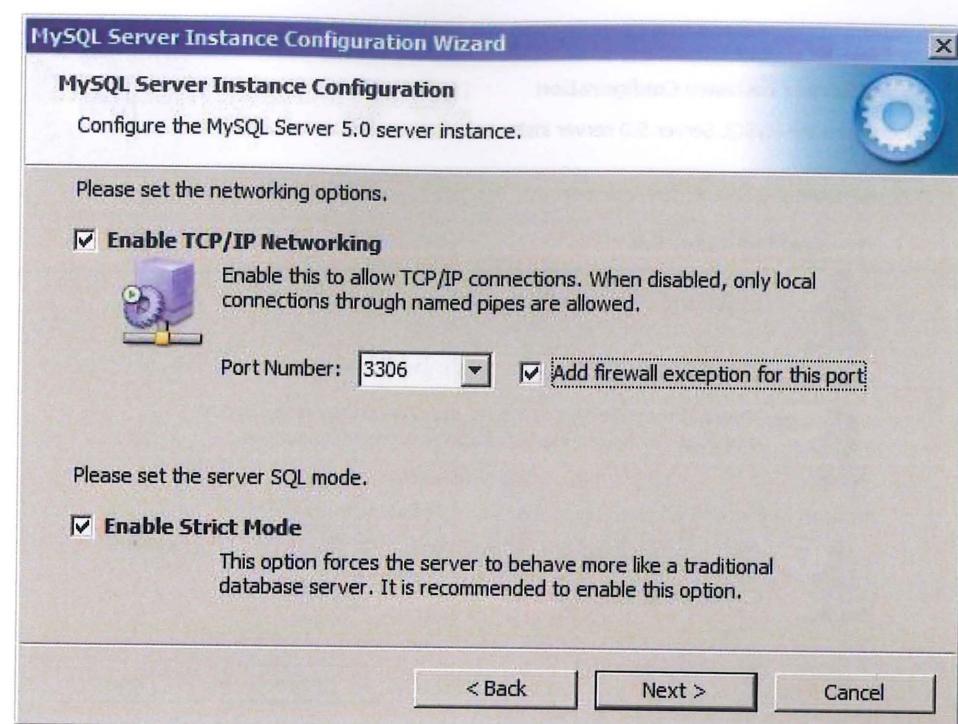


Pantalla selección directorio para datos del asistente MySQL

- Tras seleccionar la ubicación de los datos, es el momento de indicar el número de usuarios que de forma simultánea van a acceder al sistema, el asistente nos va a proporcionar a este respecto tres opciones:
 - 1^a opción. N° bajo de conexiones simultaneas
 - 2^a opción. Transacciones OLTP, transacciones en línea
 - 3^a opción. Elegir, introduciendo manualmente, el n° de conexiones simultaneas



Pantalla selección número de usuarios del asistente MySQL

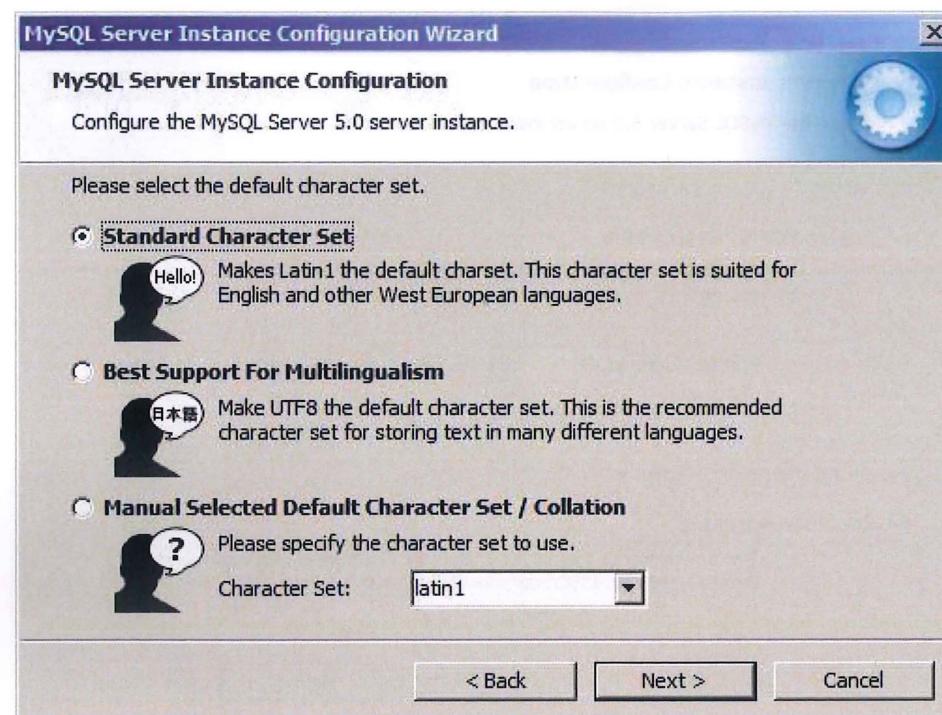


Pantalla selección de puerto del asistente MySQL

– Hemos llegado al punto del proceso en el cuál tendremos que determinar si deseamos o no el modo estricto de trabajo y sobre qué puerto vamos a trabajar. Valores normales de estos parámetros serían:

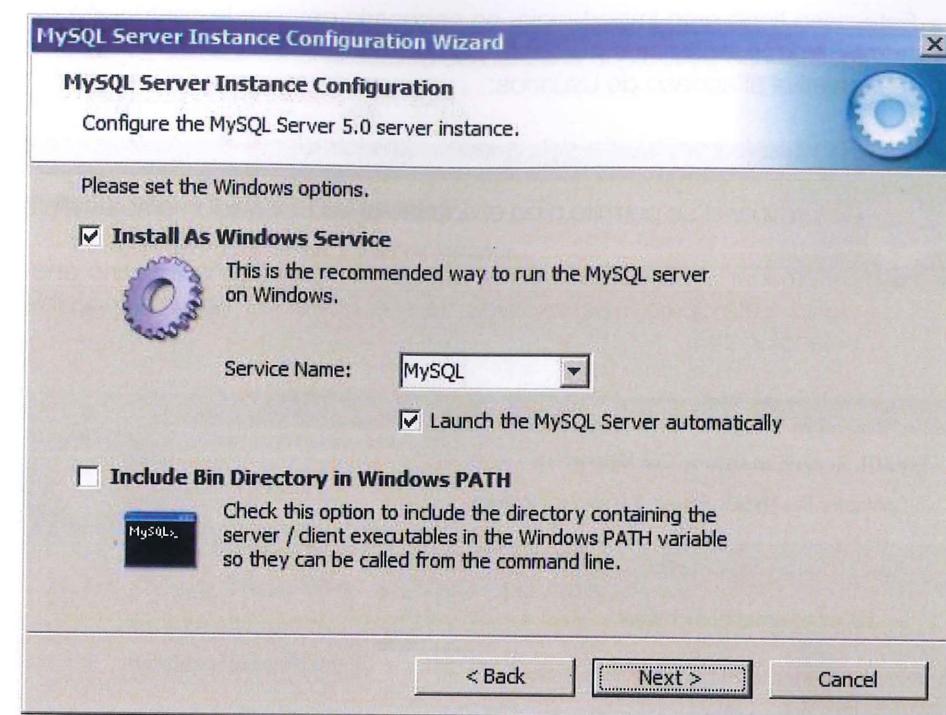
- Nº de puerto: 3306
- ¿Modo estricto de trabajo? SI

– El texto de la BBDD debe estar codificado, tras elegir el puerto y el modo de trabajo tendremos que seleccionar que juego de caracteres vamos a utilizar para codificar el texto que contendrá la BBDD.



Pantalla selección de juego de caracteres de codificación del asistente MySQL

- También, y es en este momento cuando deberemos hacerlo, tendremos que seleccionar el modo de trabajo. Lo normal es que instalamos el SGBD como cualquier otro servicio de Windows, pero se nos ofrecerá la posibilidad de realizar modificar el nombre del servicio. Sin avanzar de pantalla también podemos modificar la ubicación del directorio bin y colocarlo en el PATH de Windows.

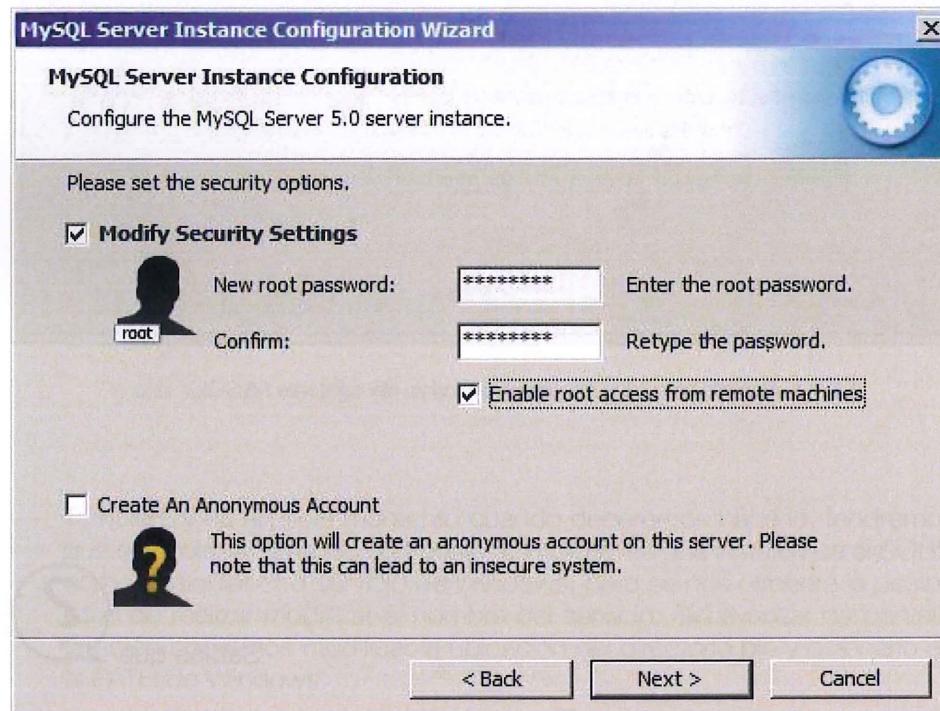


Pantalla selección nombre del servicio del asistente MySQL



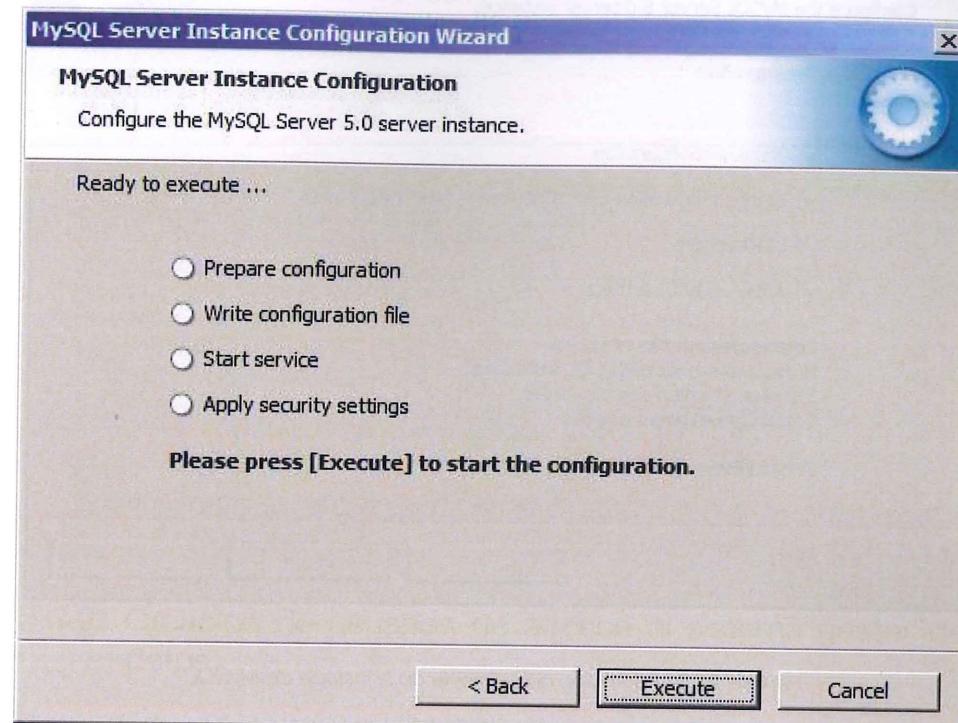
Lo que vamos a conseguir modificando la ubicación del directorio bin, y colocándolo en el PATH de Windows, es que vamos a poder ejecutar comandos propios de MySQL desde la línea de comandos sin tener que modificar la ruta.

- Este paso tiene gran importancia, es necesario cambiar la contraseña del administrador. Además es en este punto añadiremos alguna información más relativa al acceso de usuarios:
 - Cambiar la contraseña del usuario administrador
 - Determinar si se permite o no el acceso al administrador remotamente
 - Determinar si se activa o no la cuenta sin contraseña (usuario anónimo). Esta opción, por motivos de seguridad, no es recomendable seleccionarla.



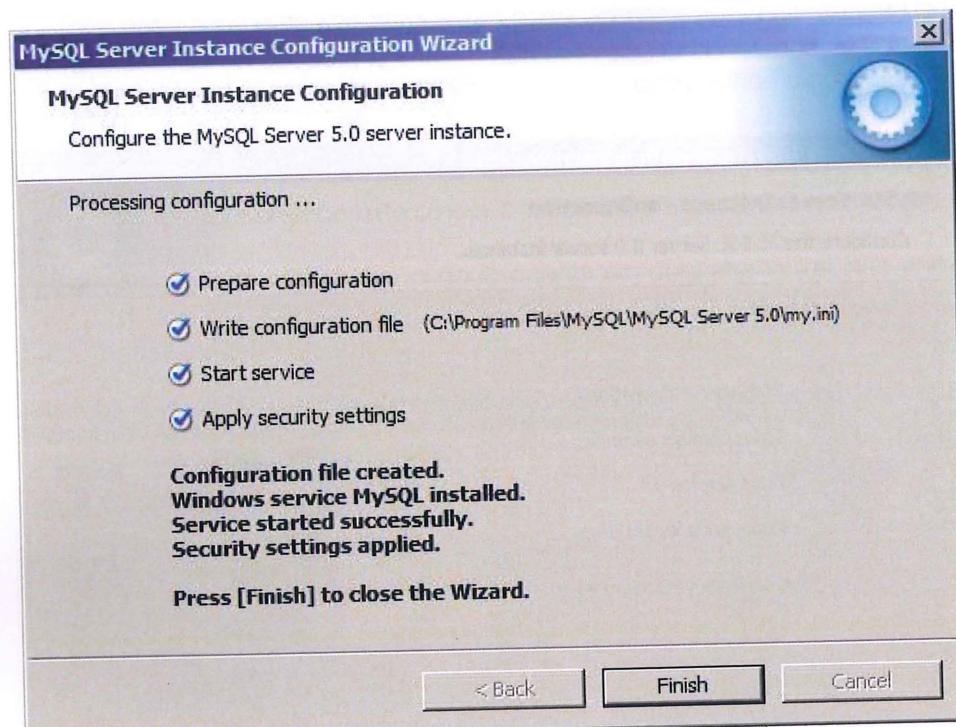
Pantalla cuentas usuario del asistente MySQL

- Una vez llegados a este punto y seleccionado todos los parámetros que nos va solicitando el asistente comenzaría el proceso de instalación MySQL propiamente dicho.



Pantalla inicio proceso de instalación del asistente de la instancia de MySQL

- El propio sistema nos indicará que la instalación se ha llevado a cabo de forma satisfactoria mostrándonos una nueva ventana emergente que nos lo va a confirmar. Esta ventana es la que muestra la figura adjunta.



Pantalla fin de instalación del asistente de la instancia de MySQL

Los parámetros con los que se ha configurado el SGBD quedarán recogidos en un fichero, llamado my.ini . La pantalla final del asistente nos va a facilitar la ubicación de este fichero, que será modificable y nos permitirá hacer cambios en el funcionamiento del SGBD, de forma independiente a lo que hayamos especificado durante el proceso de instalación.



Importante

Llegado el caso, y si surge la necesidad de desinstalar el SGBD los pasos a seguir serían:

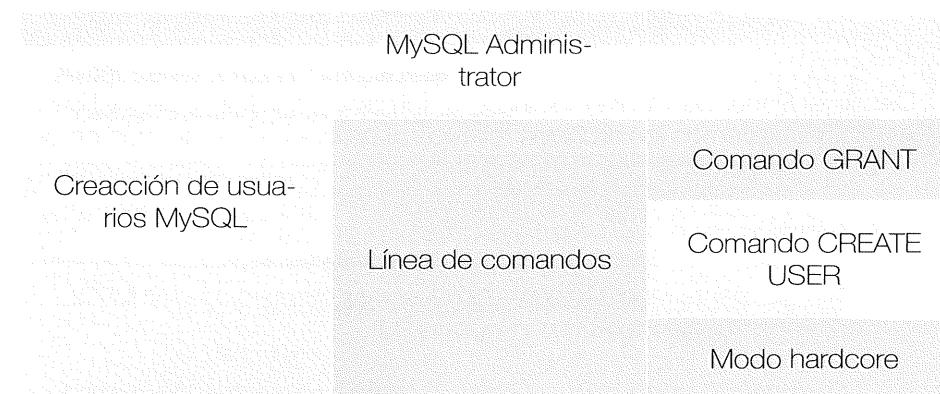
- Desde el Panel de Control localizar la aplicación MySQL y desinstalarlo. Así desinstalaremos los archivos relativos al programa que se crearon durante la instalación.
- Ir al directorio :
 - C:\Documents and Settings\All Users\Application Data\MySQL, si estamos trabajando sobre Windows 98
 - C:\ProgramData\MySQL, si estamos trabajando sobre Windows Server 2008
- Para poder ver la carpeta ProgramData tendrás que habilitar la opción "Mostrar carpetas", pues es una carpeta que, por defecto, aparece oculta.
- Comprobar que la carpeta de Archivos de Programa donde instalamos la aplicación, tras llevar a cabo los pasos anteriores, ha quedado vacía.

2.3.2. Creación de usuarios de acceso al sistema gestor de datos

Continuando con el ejemplo de MySQL vamos a ver cómo se lleva a cabo la gestión de usuarios (creación de los mismos y asignación de privilegios) en este SGBD.

Para la creación de usuarios tenemos dos opciones, por una parte tendremos la opción de crearlos directamente a través de la línea de comandos, y por otro podríamos hacer uso de la herramienta, gratuita, al igual que lo es la aplicación MySQL, MySQL Administrator. Esta herramienta es muy intuitiva y cuenta con asistente, así que puedes animarte a descargarla y probar a crear usuarios con ella. Aquí nos vamos a centrar en los comandos que permiten la gestión de los usuarios del sistema gestor de bases de datos, porque es más genérico y nos va a aportar más información y más conocimiento que el uso del asistente.

Utilizando sentencias lanzadas a través de la línea de comandos, tenemos 3 opciones para crear usuarios en un sistema MySQL, aunque en las versiones actuales del sistema gestor la que más comúnmente se utiliza es la sentencia Create User. Mira atentamente el siguiente esquema:



Comando CREATE USER

Para crear un usuario en MySQL utilizaremos el comando Create User. Antes de crear un usuario, debemos conectarnos a la base de datos como usuario administrador, para ello ejecutamos el siguiente comando:

```
$ mysql -u root
```

El parámetro `-u` del comando se utiliza para indicarle que usuario quiere conectarse al sistema. Si tuviera definida una contraseña este usuario tendría que haber añadido el parámetro `-p`.

El usuario podemos crearlo y al mismo tiempo definir una contraseña para él, añadiendo al comando `IDENTIFIED BY` y determinar si sólo podrá conectarse desde el servidor MySQL, añadiendo al comando "localhost".

La estructura del comando CREATE USER será pues:

```
Mysql> CREATE USER 'nombredeusuario'@'localhost' IDENTIFIED BY 'password';
```

Mira el siguiente ejemplo, resultado de ejecutar el comando para creación. En él se ha creado un usuario, de nombre 'fernando', que requerirá una contraseña para poder acceder, 'fer_pass' y sobre el que determinaremos que el usuario sólo se podrá conectar a través del servidor MySQL. Tras crear el usuario, ejecutaremos la conexión al sistema MySQL a través del usuario creado.

- conexión a la BBDD por el usuario Administrador


```
$ mysql -u root
```

 Welcome to the MySQL monitor. Commands end with ; or \g.
 Your MySQL connection id is 20
 Server version: 5.0.67 Source distribution
- Creación del usuario 'fernando'


```
mysql> CREATE USER 'fernando'@'localhost' IDENTIFIED BY 'fer_pass';
```

 Query OK, 0 rows affected (0,00 sec)
- Conexión a la BBDD por el usuario 'fernando'


```
$ mysql -u fernando -p
```

 Enter password:
 Welcome to the MySQL monitor. Commands end with ; or \g.
 Your MySQL connection id is 22
 Server version: 5.0.67 Source distribution

CEjemplo ejecución comando CREATE USER

El usuario así creado no tiene asociado ningún privilegio. Le asignaremos los mismos mediante el comando GRANT, y podremos quitárselos mediante el comando REVOKE.

La sintaxis del comando GRANT sería:

- Sintaxis general del comando


```
GRANT privilegios ON base/tabla TO usuario [IDENTIFIED by 'contraseña'] [WITH GRANT OPTION];
```

- Ejemplo asignación de todos los privilegios sobre una determinada BBDD
GRANT ALL ON BD.* TO USER@localhost IDENTIFIED BY "PASSWORD";
- Ejemplo asignación algunos privilegios sobre una determinada BBDD
GRANT select,insert,update,delete ON BD.* TO USER@localhost IDENTIFIED BY "PASSWORD";

La sintaxis del comando REVOKE sería:

- Sintaxis general del comando
REVOKE privilegios ON base/tabla FROM usuario
- Ejemplo eliminación de todos los privilegios de un usuario
REVOKE ALL ON BD.*from USER@localhost

2.3.3. Relación de pruebas que verifiquen las diferentes funcionalidades

A nivel general, todos los sistemas gestores de bases de datos pueden someterse a diferentes pruebas de funcionalidad que arrojarán datos con los que poder medir su funcionalidad. Estas pruebas son de muy diversa índoles. A continuación definimos alguna de las pruebas que se utilizan para verificar la funcionalidad de los SGBD



La **funcionalidad** de un sistema son las características que hacen que éste aporte a la organización utilidad y que resulte práctico.

Prueba unitaria	Busca asegurar que el código de la aplicación funciona de acuerdo con las especificaciones y que su módulo lógico es válido.
Prueba de integración	Intenta identificar problemas que puedan surgir cuando combinamos programas que de forma independiente funcionan de forma correcta. Verificará que el funcionamiento es correcto entre las entidades externas y las diferentes aplicaciones.
Pruebas de regresión	Determinan si algún cambio realizado en alguna parte de la aplicación en un momento cercano en el tiempo puede tener algún efecto negativo en otra parte de la aplicación
Pruebas de sistema	Permite que queden aseguradas la introducción y recuperación de los datos y la navegación en el sistema
Pruebas de desempeño	Hace una validación del cual es el tiempo de respuesta de los procesos
Pruebas de carga	Hace una verificación del tiempo de respuesta del sistema ante determinadas situaciones de carga
Prueba de Stress	Evaluá la respuesta del sistema ante situaciones delicadas, como puede ser poca disponibilidad del servidor, memoria baja, muchos usuarios conectados, etc.)
Pruebas de volumen	Verifica el funcionamiento del sistema cuando el volumen de datos es máximo y cuando también lo es el tamaño de la BBDD
Pruebas de integridad	Se utilizarán para que ante los diferentes métodos de acceso y al ejecutarse los diferentes procesos los datos no se corrompen.
Pruebas funcionales	Con ellas se pretende asegurar que los requisitos del sistema se cumplen, en cuanto a entrada de datos, navegación por el sistema, procesamiento de los datos y obtención de resultados.

2.3.4. Relación de pruebas que verifiquen las diferentes funcionalidades

Siguiendo con el ejemplo de la instalación vamos a concretar a que pruebas de funcionalidad podemos someter a nuestra sistema MySQL para confirmar que va a resultar práctico y eficaz como parte del ERP que queremos implementar.

La prueba más básica consistirá en ejecutar algunos comandos simples que, sencillamente, verificarán que se inicia correctamente al arranque del sistema y que la parada y arranque cuando el usuario lo desea también funciona a la perfección. Estos serían los comandos que nos permitirían comprobar las funcionalidades que acabamos de describir:

- **Inicio al arranque**

MySQL se deberían iniciar automáticamente cuando arranque Windows

- **Para detener el servicio**

`net stop MySQL`

- **Para reiniciar el servicio**

`net start MySQL`

- **Comprobar que el servicio está activo**

`Mysql -u root -p`

Como también ocurría en la gestión de usuarios, además de a través de comandos, también existe disponible un paquete de aplicaciones descargable, el MySQL Test.

La aplicación funciona generando ciertos test prefijados, que deben de arrojar todos ellos resultado positivo para dar por buena la funcionalidad. También permite que los usuarios generen sus propios test. La aplicación de testeo está constituida por varios elementos: un script para ejecutar los test, los casos de utilidad de test actuales, los resultados que se esperan de ellos y un script para ejecutar los test.

Sistema de Test	Interprete de idioma	mysqltest
	Shell script para ejecutar los test	mysql-test-rum

Caso de uso del test y resultados esperados

Si, tal como ha sido nuestro caso, la instalación de MySQL se ha llevado a cabo en una distribución binaria, ejecutaremos los test lanzando el siguiente comando:

`scripts / mysql-test-rum`

2.3.5. Documentación de los procesos realizados

Documentar los procesos que se realizan en los sistemas de información es algo fundamental. ¿Qué ventajas puede aportar el hecho de tener documentados los procesos?

- El saber en qué punto de desarrollo están los sistemas y cuáles son las características del mismo facilitarán la relación con los posibles clientes de la organización.
- Todo el mundo sabe cómo funciona el sistema, por lo que nadie se convertirá en pieza imprescindible del sistema.
- No sólo los miembros de la organización, sino también los que se vayan incorporando a ella, podrán usar el sistema y todas sus herramientas de manera correcta, además de poder aportar ideas para sus posibles mejoras.

Para que esta documentación sea válida ha de tener algunas características: ha de ser pertinente, exhaustiva, económica y rápida.

Características de la documentación de sistemas	Pertinencia	Los documentos que se proporcionan han de dar respuesta a las necesidades concretas de quienes los van a utilizar
	Exhaustividad	Han de estar disponibles para el usuario todos aquellos documentos que requiera para desempeñar su trabajo
	Rapidez	Los documentos que describen procesos han de llegar a los usuarios nada más publicarse
	Economía	La generación de la documentación no debe tener un coste alto

Esto en cuanto a la documentación, y concretando en los documentos que la forman, estos han de cumplir una serie de funciones:

- Han de constituir un medio de comunicación
- Ha de ser un medio de formación
- Puede contribuir a la formación colectiva, al recoger diversas categorías de conocimiento.

Consideraremos que un documento es válido, es decir, que cumple con las funciones mencionadas si es original, utilizable, fiable y confidencial.

Documento válido			
Original	Fiable	Utilizable	Confidencial

2.4. Parámetros de configuración del gestor de datos



En un SGBD los parámetros son las variables que nos van a permitir modificar las características del sistema para, o bien, optimizar su rendimiento, o bien para adecuarlo a las especificaciones determinadas requeridas por un determinado uso o por un usuario.

Hay tres formas de optimizar el rendimiento del sistema: mejorando la estructura de las tablas, mejorando las consultas y mejorando la configuración del servidor MySQL. Podremos realizar modificaciones en la BBDD con objeto de conseguir estas mejoras, a través de los parámetros que para ello se definen en cada uno de los SGBD.

¿Cómo optimizamos el rendimiento del sistema?	Por medio de parámetros que actuarán sobre	La estructura de las tablas
El formato de las consultas	La configuración del servidor del SGBD	

En todos los casos, deberemos encontrar un equilibrio entre la optimización del sistema y el coste, evaluado a todos los niveles, que eso supone

Por ejemplo, ¿Por qué y cómo podemos mejorar, mediante la modificación de la estructura de las tablas, el rendimiento del sistema?

- Está comprobado que el hecho de que las tablas estén desordenadas provoca que las búsquedas secuenciales sean lentas, y por lo tanto, que el rendimiento del sistema empeore, pero tener una tabla perfectamente ordenada, también es costoso, ya que va a requerir un constante acceso de escritura a los registros de la tabla y porque hay que poner especial cuidado en no desordenar el resto cuando ordenamos alguna columna.
- Una opción para crear consultas más rápidas es la utilización de índices, ¿Por qué?
 - Porque algunas consultas se ejecutan más rápido si existen los índices
 - Porque es menos costoso insertar, modificar o eliminar un índice que hacerlo directamente en una tabla ordenada
- También podemos mejorar el rendimiento si fijamos la longitud de los registros en vez de permitir que estos sean variables. Así:
 - Disminuiremos la fragmentación
 - Aumentaremos la velocidad
 - Disminuiremos el consumo de memoria
- Otra opción sería desfragmentar las tablas, para de esta manera, ganar espacio y velocidad.

2.4.1. Definición de los parámetros de configuración de un sistema gestor de datos

Algunos de los parámetros configurables en un sistema gestor de datos van a ser: el tipo de servidor a utilizar, el uso que se va a hacer de la base de datos, el nº de conexiones simultáneas que se permitirán, el puerto a través del cual se van a gestionar las conexiones, el modo de trabajo, el juego de caracteres, el modo de trabajo del servicio y las contraseñas de usuarios.

– Parámetros elección tipo de servidor

Los diferentes servidores disponibles para el SGBD se elegirán en función de si queremos que esté optimizado para determinadas funciones, por ejemplo, si está optimizado para un determinado tipo de sistema operativo, o de si queremos que soporte un tipo determinado de tablas

– Parámetros de manejo de memoria

Estos parámetros son los que van a determinar que la gestión de la memoria se lleve a cabo de una forma u otra. Los hay globales, y también que controlan cada una de las conexiones de los clientes al servidor

– Parámetros de consultas a la Caché

Al activar esta funcionalidad del servidor agilizamos en gran medida la respuesta de las consultas que se realizan a la base de datos. Por defecto es un parámetro que vendrá deshabilitado

– Optimización parámetros cliente

Serán los parámetros que afecten al tamaño de los buffer de los datos que los clientes pueden utilizar al insertar, modificar u obtener los datos de respuesta de alguna consulta.

– Otros parámetros: nº de conexiones simultáneas que se permitirán, el puerto a través del cual se van a gestionar las conexiones, el modo de trabajo, el juego de caracteres, el modo de trabajo del servicio y las contraseñas de usuarios

2.4.2. Diferentes tipos de parámetros



Cuando finalizamos la instalación del sistema MySQL con el asistente, la última de las pantallas nos decía donde quedaba ubicado el archivo **my.ini**. Este archivo guardaba los parámetros con los que habíamos llevado a cabo la instalación del sistema gestor de datos. La modificación de este archivo nos va a permitir modificar los parámetros de la aplicación, independientemente de cuáles fueran los originales en la instalación.

Continuando con el ejemplo de MySQL, vamos a ver algunos de los parámetros concretos utilizados para optimizar seleccionar el tipo de servidor, gestionar la memoria y las consultas a la memoria caché

Parámetros para seleccionar el tipo de servidor MySQL

Parámetro	Descripción
mysqld-debug	Compilado con el máximo de funciones de depuración y control automático de asignación de memoria, así como con soporte para tablas InnoDB y BDB.
mysqld	Ejecutable optimizado con soporte para InnoDB
mysqld-nt	Ejecutable optimizado para Windows NT, 2000, y XP con soporte para "named pipes".
mysqld-max	Ejecutable optimizado con soporte para tablas InnoDB y BDB.
mysqld-max-nt	Similar a mysqld-max, pero compilado con soporte para "named pipes".

Parámetros para seleccionar el tipo de servidor MySQL

Parámetro	Descripción
key_buffer_size	Cantidad de memoria destinada a cachear bloques de los índices de tipo MyISAM.
table_cache	Número máximo de tablas abiertas.
open_files_limit	Número máximo de archivos abiertos.
thread_cache_size	Número de hilos siempre activos en el sistema.
connect_timeout	Tiempo de desconexión para conexiones entrantes.

Parámetros para configurar las consultas a la caché

Parámetro	Descripción
query_cache_size	Le indica al servidor la cantidad de memoria que se destinará en la caché las consultas. Este parámetro siempre será alineado a 1024
query_cache_limit	Establece un límite para que un resultado individual de una consulta sea puesto en la caché. Con este parámetro se evita meter a los "full scan table" sacarí a datos de la caché.
query_cache_type	Va a definir cómo funciona la memoria caché
query_cache_type=0	Si le asignamos este valor al parámetro, ni cogerá ni guardará datos en la caché
query_cache_type=1	Valor del parámetro que asegura un comportamiento más óptimo. Todas las consultas, excepto 'SELECT SQL_NO_CACHE', utilizan la caché
query_cache_type=2	Este valor del parámetro va a implicar que sólo las consultas que explícitamente lo solicitan mediante la expresión 'SELECT SQL_CACHE', ocuparán la caché

UD2

Lo más importante

- Los sistemas gestores de datos tienen como componente fundamental la base de datos (BBDD), que es el elemento que les otorga su máxima funcionalidad gracias a la capacidad que éstas poseen de almacenar información. Subobjetivo de un sistema gestor de datos es proporcionar seguridad y eficiencia en el almacenamiento o extracción de los datos almacenados en las BBDD.
- En relación a la BBDD, el SGBD (Sistema Gestor de Bases de Datos) tendrá que proporcionar los siguientes servicios a los usuarios: definirla, crearla y mantenerla y controlar el acceso a ella. La estructura de la BBDD es quien va a describirla mediante un modelo de datos (conceptual, lógico, físico). La ocurrencia de la BBDD hace referencia a los datos que contiene en un determinado momento del tiempo.
- Son características de los sistemas gestores de datos: la abstracción de la información, la redundancia mínima, la consistencia, la integridad, la seguridad, el respaldo y la recuperación y el control de la concurrencia.
- Los componentes del Sistema Gestor de Bases de Datos (SGBD) son: los lenguajes de programación, el diccionario de datos, los mecanismos de seguridad e integridad de los datos, el administrador de la base de datos. Son características inherentes a los SGBD: que exista una separación entre los datos y los programas de aplicación que hacen uso de esos datos; los usuarios deben de poder manejar diferentes vistas; El SGBD ha de usar un catálogo que almacene el esquema definido para la base de datos.

- En las arquitecturas de las BBDD nos vamos a encontrar con tres niveles: uno interno o físico, uno externo o de visión y otro conceptual.
- En función de la tecnología empleada en su funcionamiento nos vamos a encontrar con diferentes tipos de bases de datos: BBDD en red, BBDD jerárquicas, BBDD relacionales y BBDD orientadas a objetos
- El requisito previo a la instalación de cualquier SGBD es plantearnos las necesidades, a fin de encontrar la solución óptima que resuelva de forma eficaz y eficiente nuestras necesidades.
- Los aspectos a valorar ante la elección del SGBD a instalar son: Tamaño de la BBDD, la conectividad, el nº de usuarios que van a utilizarlo, el nº de conexiones simultáneas, la política de empresa y el aprovechamiento del software
- Previo a la instalación es necesario comprobar si la infraestructura existente cumple los requerimientos necesarios. Hay que valorar el sistema operativo, las aplicaciones preinstaladas, la memoria RAM, el procesador, el disco duro, los requisitos de red y evaluar posibles incompatibilidades.
- Los usuarios en los sistemas gestores pueden ser anónimos, no necesitarán contraseña, root, que serán los administradores, o usuarios convencionales sobre los que puedes definirte o no contraseñas. El administrador tendrá todos los privilegios y al resto de usuarios se les podrán asignar los que se estime oportuno en función de sus necesidades.
- La funcionalidad del sistema se va a verificar mediante diversas pruebas, que nos van a decir como de útil y práctico resulta ser el sistema para la organización.
- Documentar los procesos realizados a la hora de seleccionar e instalar un SGBD tiene importantes ventajas para la empresa. Para que pueda cumplir con sus funciones la documentación ha de tener ciertas características: ha de ser pertinente, exhaustiva, económica y rápida.
- Mediante los parámetros de configuración vamos a poder variar las características de la base de datos. Podremos modificar el número posible de conexiones simultáneas, el modo de trabajo del servicio, el juego de caracteres con el que se van a codificar los datos, la contraseña del usuario administrador...

UD2

Autoevaluación

1. Señala cuál de estas afirmaciones es falsa:
 - a. Los sistemas gestores de datos pueden trabajar con BBDD comerciales externas a ellos.
 - b. Las únicas opciones disponibles para un gestor de datos, en cuanto a sus BBDD se refiere, es que sean propias de la aplicación, o extendidas de fabricantes comerciales.
 - c. Los sistemas gestores de datos sólo puede trabajar con bases de datos propias de la aplicación.
 - d. En los sistemas gestores de datos, es la BBDD quién les otorga su mayor funcionalidad.
2. Es verdadero que en relación a las BBDD, los SGBD:
 - a. Nos van a permitir crearla y modificarla, pero no crearla.
 - b. Nos va a proporcionar los mecanismos de seguridad que controlarán el acceso a los datos.
 - c. Nos va a permitir definirla y crearla, pero no modificarla.
 - d. Nos va a permitir controlar el acceso a los datos, pero no definirla ni crearla ni modificarla.

3. El objetivo que persigue un sistema gestor de datos es
- Lograr que se puedan manejar de una forma fácil y ordenada los datos que contiene.
 - Almacenar los datos que la empresa necesita.
 - Manejar de forma sistemática y procedimentada los datos que contiene.
 - Conseguir que los usuarios sepan exactamente cuál es la ubicación física de los archivos.
4. Señala la respuesta falsa: "Los lenguajes que va a utilizar el sistema gestor de datos serán las herramientas que van a permitir a los usuarios..."
- Especificar cuál va a ser la estructura de la BBDD.
 - Definir las reglas de integridad de la información.
 - Manipular los datos contenidos en la BBDD.
 - Modificar la estructura de la BBDD.
5. El modelo en red para una BBDD trata de superar los inconvenientes que surgen en un modelo
- Relacional.
 - Jerárquico.
 - Orientado a Objetos.
 - Declarativo.
6. Indica qué concepto define la siguiente definición: "Es la clave candidata que selecciona el administrador para identificar cada una de las entidades."
- Clave Candidata.
 - Clave primaria.
 - Clave Ajena.
 - Superclave.
7. En un diagrama que representa la estructura de datos en un modelo E-R los atributos se representan por
- Rombos.
 - Rectángulos.
 - Elipses.
 - Círculos.
8. El tamaño de la BBDD, la conectividad, la política de empresa o el aprovechamiento del software son:
- Requerimientos que debe cumplir el sistema para instalar el SGBD.
 - Aspectos a valorar, respecto a la infraestructura existente, previos a la selección del SGBD a instalar.
 - Aspectos a valorar, respecto a la infraestructura existente, previos a la instalación del SGBD, pero posteriores a su elección.
 - Subprocesos del proceso de instalación del sGBD.

9. Señala la respuesta verdadera. El comando que hay que ejecutar si queremos eliminar todos los privilegios que un usuario tiene asignados es:

- a. REVOKE ALL ON BDlocalhost;
- b. GRANT ALL ON BD. TO USER@localhost IDENTIFIED BY "PAS-SWORD";
- c. REVOKE privilegios ON base/tabla FROM usuario;
- d. REVOKE ALL ON BD.from USER@localhost;

10. Respecto a los parámetros de configuración de las BBDD, señala cuál de las respuestas es falsa

- a. Una vez instalada, se pueden modificar las características de la base de datos.
- b. Los parámetros de una base de datos se pueden modificar modificando el archivo my.ini.
- c. Los parámetros de una base de datos se pueden modificar modificando el archivo ini.MySQL.
- d. Son parámetros modificables que variarán la configuración de la base de datos, el modo de trabajo del servicio o el número posible de conexiones simultaneas.

Glosario

- **Journaling:** Es un mecanismo que se utiliza en los sistemas de archivo para registrar la información que se necesitaría para el restablecimiento de los datos que se vieran afectados por un posible error en una transacción.
- **Tabla maestra de archivos, MTF:** Tabla que contiene valores numéricos. La asignación de clústers que forman cada partición está contenida en una celda de esta tabla
- **Multiprogramación:** Es la técnica por la cual dos, o incluso más procesos, pueden ser alojados en la memoria principal de un equipo y, además, pueden ser ejecutados de forma simultanea por el procesador
- **Máquina extendida:** Es otro de los términos que se utiliza para denominar a una máquina virtual. Es el SO que funciona sobre el hardware del equipo interpretando ordenes programadas que controla y administra de forma autónoma. Presenta su propia interfaz de usuario, que le permite, a través de ella, comunicarse con el hardware
- **Scheduler:** El shceduler, o planificador, es el componente de los SO multitarea y multiprocesos que permite que el tiempo del que dispone el procesador se reparta de forma eficiente entre todos los procesos que han de ejecutarse.
- **Granja de servidores:** También llamadas Server farm o Server Cluster están formadas por una serie de servidores conectados entre sí y, que normalmente, están ubicados en espacios especialmente habilitados para ellas.