

# Base de Datos

Conceptos fundamentales






## **Datos-Información-Conocimiento**

- Los datos son considerados como hechos aislados
- El procesamiento de los datos da lugar a la información
- El procesamiento de la información da lugar al conocimiento

**La información es uno de los recursos más importantes con que cuenta la empresa debido a:**

- La influencia que tiene en la planificación
  - La toma de decisiones en las organizaciones
- 

# Evolución

Almacenamiento de los datos :

- Al principio los datos estaban **integrados en los programas**
- La aparición de los **ficheros** da lugar a que los datos aparezcan como una colección homogénea. Fichero como conjunto de datos estructurados.
  - Estructura lógica → como se muestran al usuario
  - Estructura física → almacenamiento en soporte físico
  - ORIENTADOS A LOS PROCESOS
- Las primeras **bases de datos** aparecen en la década de los sesenta
  - ORIENTADOS A LOS DATOS


Las **ventajas** que aportan las bases de datos sobre los sistemas de ficheros son:

- *Control sobre la redundancia de datos.* En los sistemas de bases de datos todos estos ficheros están integrados, por lo que NO se almacenan varias copias de los mismos datos.
- *Consistencia de datos.* Eliminando o controlando las redundancias de datos se reduce en gran medida el riesgo de que haya inconsistencias. Si un dato está almacenado una sola vez, cualquier actualización se debe realizar sólo una vez, y está disponible para todos los usuarios inmediatamente.
- *Compartición de datos.* En los sistemas de bases de datos, la base de datos pertenece a la empresa y puede ser compartida por todos los usuarios que estén autorizados. Además, las nuevas aplicaciones que se vayan creando pueden utilizar los datos de la base de datos existente.
- *Se simplifica el esfuerzo de programación y mantenimiento de los programas.*  
*Mantenimiento de estándares.* Gracias a la integración es más fácil respetar los estándares necesarios, tanto los establecidos a nivel de la empresa como los nacionales e internacionales.



Una **base de datos** es:

"Una colección de **datos integrados**, con **redundancia controlada**, y con una estructura que refleja las **interrelaciones y restricciones** semánticas existentes en el mundo real; los datos, que han de ser **compartidos** por diferentes usuarios y aplicaciones, deben mantenerse **independientes** de éstas y su definición y descripción, únicas para cada tipo de datos, han de estar almacenadas con los mismos. Los procesos de actualización y recuperación, comunes y bien determinados habrán de ser capaces de **conservar la integridad, seguridad y confidencialidad** del conjunto de los datos"



# Niveles de abstracción

El **nivel externo** está constituido por las vistas que tiene cada usuario de la base de datos

El **nivel conceptual** es el nivel en el que se hace el diseño conceptual de la base de datos. Consiste en una simple y lógica descripción de todos los elementos de los datos y sus interrelaciones

El **nivel interno** es la vista física de la base de datos: los dispositivos de disco, las direcciones físicas, los índices, los punteros y demás elementos

# Modelos de Bases de datos

Jerárquico.

- En red.
- Relacional.
- Orientadas a objetos
- Orientadas a internet

# Modelo jerárquico

Se le llama también *modelo en árbol*, ya que utiliza para su representación una estructura de tipo árbol invertido.

Una base de datos jerárquica es un conjunto de registros lógicamente organizados con una estructura de árbol invertido. Dentro de la jerarquía el nivel superior se percibe como el "padre" de los registros situados debajo de él, de forma que:

- Cada padre puede tener muchos hijos
- Cada hijo sólo tiene un padre

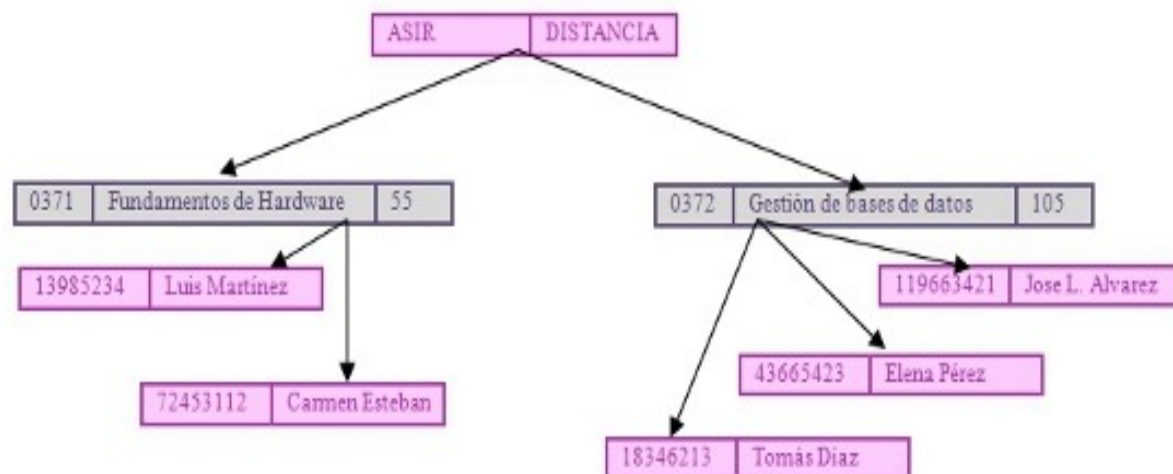


MODULOS

CODIGO	NOMBRE	HORAS
--------	--------	-------

PROFESORES

NRP	NOMBRE
-----	--------



## Ordenador HP 260



**Base de datos IMAGE**

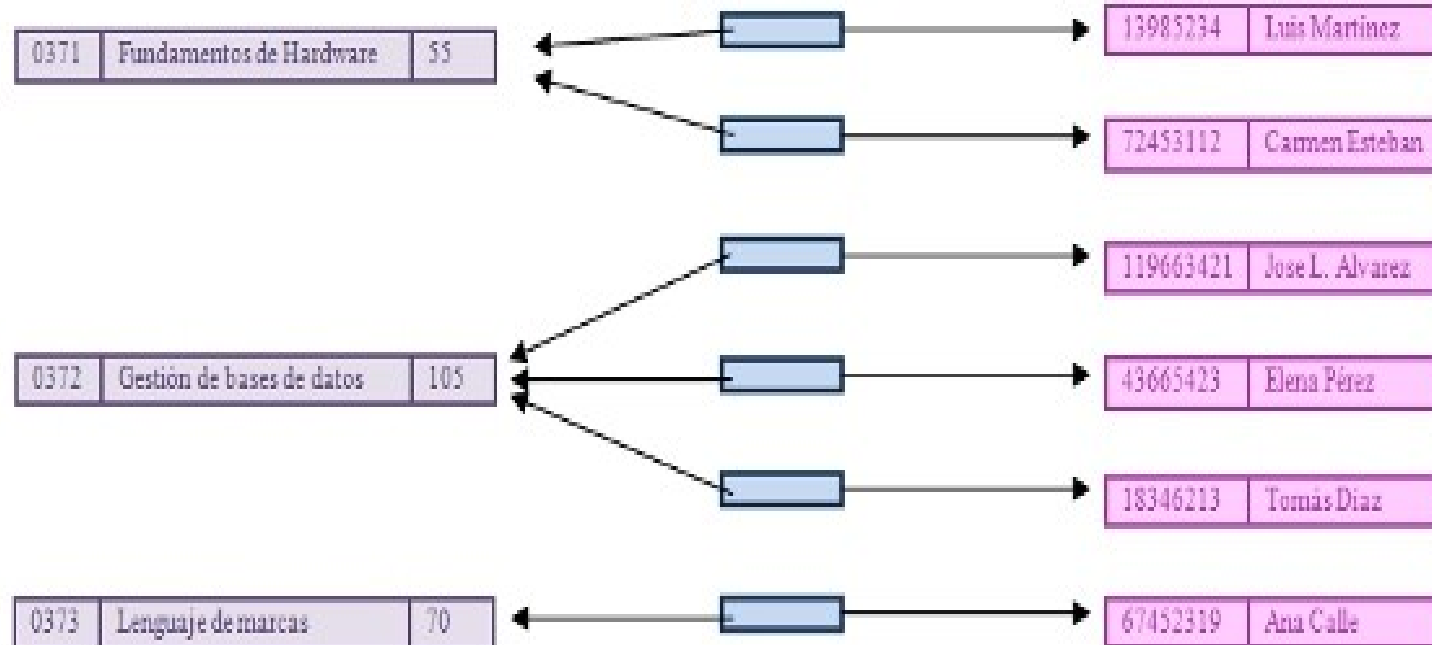
[http://www.hpmuseum.net/display\\_item.php?hw=250](http://www.hpmuseum.net/display_item.php?hw=250)

# Modelo en red

Es un modelo de datos propio de los sistemas comerciales de los años 70, que aún está vigente si no se piden demasiadas modificaciones al sistema.

Fue creado para representar relaciones más complejas y eficientes que las del modelo jerárquico y así imponer un estándar de bases de datos que ayudara a los diseñadores y a los programadores.

### Registro conector



# Modelo relacional

Fue desarrollado por Codd para **IBM** en los años 70, pero inicialmente los ordenadores carecían de prestaciones para poder ejecutarlo.

Actualmente es el modelo más utilizado para modelar problemas reales y administrar datos dinámicamente.

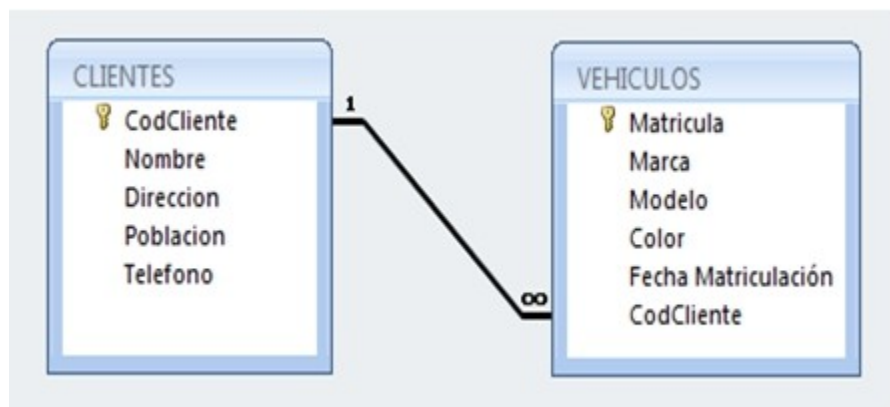
Su principal ventaja es que permite que el usuario y el diseñador operar en un entorno que se percibe como un conjunto de tablas y los detalles físicos complejos los maneja el sistema.

# *Las características del modelo Relacional*

- Este modelo representa los datos y las relaciones entre ellos como una colección de **tablas**,
- De manera simple, una relación representa una tabla que no es más que un conjunto de **filas**, cada fila es un conjunto de **campos** y cada campo representa un valor que describe el mundo real.
- Las tablas son independientes pero se relacionan mediante un vínculo común.
- Proporciona una redundancia y una inconsistencia mínima.  
La independencia de datos de las aplicaciones y del dispositivo de almacenamiento

CLIENTES				
CodCliente	Nombre	Dirección	Población	Teléfono
1	Francisco Álvarez	C/ La Mata 9.	Alcántara	925767788
2	Raquel Marcos	C/ La Amapola 7.	Toledo	925998811
3	Carlos Revuelta	C/ La Arboleda 12.	Madrid	919090771
4	José María Cabello	C/ La plaza 72	Madrid	914409071
5	Jorge Peña	C/ Fresnedo, 23	Guadalajara	949788896
6	Dolores Manzano	C/ Autonomía, 8	Madrid	916767560

VEHICULOS					
Matricula	Marca	Modelo	Color	Fecha Matriculación	CodCliente
4534 FNG	Ford	Focus	Negro	14/04/2007	2
1203 CLL	Cotroën	C4	Magenta	23/08/2005	5
3367 GHB	Suzuki	Vitara	Cobalto	16/05/2009	2
1004 JLG	Kia	Rio	Rojo	02/07/2009	6
6709 BFG	Peugeot	206	Gris plata	12/10/2006	3



# Modelo orientado a objetos

Los modelos de bases de datos intentan representar cada vez con más fidelidad los problemas del mundo real que cada vez son más complejos. Uno de los modelos que se han desarrollado recientemente (años 90) es el modelo de bases de datos orientada a objetos.

Se denomina así porque su estructura básica es un objeto, que recoge tanto datos como sus relaciones. Supone una forma diferente de definir y utilizar las entidades.

Un objeto se describe como un conjunto de hechos, pero incluye también información sobre la relación que tienen los hechos dentro del objeto y con otros objetos; así como todas las operaciones que puedan ser realizadas en él.



# Modelos de bases de datos de internet

El mercado de las bases de datos ha cambiado sustancialmente con el creciente desarrollo del uso de internet en las transacciones comerciales. Ahora los esfuerzos se dirigen a la creación y desarrollo de **bases de datos que se comuniquen fácilmente por internet.**

Se busca:

- Acceso flexible a internet.
- Que se conecten fácilmente con distintas estructuras de datos
- Que el diseño del modelo conceptual sea sencillo.
- Que disponga de herramientas de diseño, consulta, desarrollo de aplicaciones e interfaz gráfica potentes y que faciliten el trabajo.

# Big Data

Otro camino abierto es el denominado **Big Data** (del idioma inglés *grandes datos*) es en el sector de tecnologías de la información y la comunicación una referencia a los sistemas que manipulan grandes conjuntos de datos (o *data sets*).

Las dificultades más habituales en estos casos se centran en la captura, el almacenado, búsqueda, compartir, análisis, y visualización.

La tendencia a manipular ingentes cantidades de datos se debe a la necesidad en muchos casos de incluir los datos relacionados del análisis en un gran conjunto de datos relacionado, tal es el ejemplo de los análisis de negocio, los datos de enfermedades infecciosas, o el combate con el crimen organizado.

# Componentes:

Los **Sistemas Gestores de Bases de Datos** (*SGBD*) son paquetes de software complejos que deben proporcionar una serie de servicios que permiten almacenar y explotar los datos de forma eficiente.

# Lenguajes de los SGBD

- **LDD o DDL (Lenguaje de definición de datos):** se utiliza para definir el esquema conceptual y el esquema interno de la base de datos: los objetos de la base de datos, las estructuras de almacenamiento y las vistas de los distintos usuarios. Lo emplean los diseñadores de la base de datos y los administradores.
- **LMD o DML (Lenguaje de manipulación de datos):** se utiliza para consultar y actualizar los datos de la base de datos. Lo emplean los usuarios para consultar, insertar, modificar o borrar datos en una base de datos. A menudo estas sentencias están embebidas en un lenguaje de alto nivel llamado lenguaje anfitrión
- **Lenguaje de control de datos (LCD o DCL)** con el cual se controlarán los temas referentes a seguridad y administración de la base de datos

La mayoría de los SGBD incorporan **lenguajes de cuarta generación** que permiten al usuario desarrollar aplicaciones de forma fácil y rápida. Se denomina también *herramientas de desarrollo*.

# El diccionario de datos

Es una guía donde se describe la base de datos con todos los objetos que la forman. Se dice también que contiene "metadatos" porque es información sobre los datos.

Contiene las características lógicas como: nombre, descripción, alias, contenido y organización, donde se almacenan los datos del sistema. Identifica los procesos donde se emplean los datos y los sitios donde se necesita acceder a la información

El diccionario proporciona información acerca de:

- La estructura lógica y física de la base de datos.
- La definición de cada uno de los objetos: tablas, vistas, índices, funciones, procedimientos, etc.
- Los valores que toman las columnas de las tablas por defecto
- Información que permite garantizar la integridad de los datos almacenados
- Los privilegios y control de acceso de los usuarios
- Normas que garanticen la seguridad de los datos.
- Estadísticas y auditorías de los accesos a los objetos, etc.

# Seguridad e integridad de los datos

Son una serie de mecanismos que proporciona el SGBD para garantizar un acceso correcto, seguro y eficiente a los datos. Se hace mediante un componente software que se encarga de:

- Garantizar que el acceso a los datos se permita solo a los usuarios autorizados.
- Disponer de herramientas para planificar y realizar copias de seguridad y restauración.
- Realizar los procedimientos necesarios para recuperar los datos tras un fallo o pérdida temporal.
- Ofrecer mecanismos para implantar restricciones de integridad que los datos deberán cumplir.
- Controlar el acceso concurrente de varios usuarios a los datos sin que se pierda la consistencia

# Usuarios del SGBD

- **Programadores:** Son los responsables de la creación y ajuste de las aplicaciones que ataquen a los datos. Emplean DDL, DML y cualquier lenguaje anfitrión.
- **Usuarios expertos:** Emplean las utilidades de la base de datos y el DML para acceder a los datos y realizar sus propios procesos sobre los objetos para los que se les ha concedido permiso.
- **Usuarios ocasionales:** que utilizan programas de aplicación para acceder a las bases de datos, pero que solo pueden utilizar aquellos objetos para los que se les ha dado permiso de acceso.
- **Diseñadores-Administradores:** Los diseñadores planifican y desarrollan las bases de datos. Definen el esquema lógico y físico de la base de datos, optimizando el almacenamiento y generando la documentación de análisis necesaria para los programadores. Cuando las bases de datos están creadas los diseñadores tienen la función de administradores. Los administradores de la base de datos gestionan la seguridad (usuarios y permisos), y la integridad de los datos asegurando que las transacciones sean correctas y no se pierdan datos. También se ocupan de crear las copias de seguridad. Tienen el máximo nivel de acceso. Utilizan fundamentalmente DDL

# Hardware

Es el conjunto de dispositivos físicos sobre los que reside la base de datos. Consiste en una o más computadoras, unidades de disco, monitores, impresoras, unidades de almacenamiento, cableado y otros equipos Auxiliares.

Las unidades de disco constituyen el principal mecanismo de almacenamiento para las bases de datos y son esenciales, ya que permiten el acceso directo, sin el cual sería imposible el procesamiento de las bases de datos.

Los sistemas de base de datos han evolucionado en la medida en que lo ha hecho la tecnología del hardware.

Requieren memoria principal y espacio de almacenamiento en disco de grandes dimensiones dentro de ordenadores dedicados.



# Funciones

Un SGBD realiza funciones que garantizan la integridad y la consistencia de los datos en una base de datos. La mayoría de estas funciones son transparentes para los usuarios finales y que las realiza el propio SGBD.




### ***1. Administración del diccionario de datos:***

Las definiciones de los datos y sus relaciones guardan en el diccionario de datos. El SGBD utiliza este diccionario para buscar las estructuras y las relaciones de los datos que cada programa solicita. Cualquier cambio que se realice en la estructura de una base de datos queda automáticamente registrado aquí, sin que el usuario tenga que modificar los programas que accedan a la estructura modificada.

### ***2. Administración del almacenamiento de datos:***

Esta función permite al SGBD crear las estructuras necesarias para el almacenamiento de datos, liberando al usuario de tener que definir y programar las características físicas de los datos. Permite almacenar no sólo los datos, sino formularios de entrada, definiciones de filtros relacionados, de informes, reglas de validación, procedimientos, estructuras para datos con formatos de vídeo, imagen, hoja de cálculo, gráficos, etc.






### ***3. Transformación y presentación de datos:***

El SGBD transforma los datos que se introducen en las estructuras necesarias para guardarlos. Transforma las solicitudes lógicas en comandos que localizan y recuperan físicamente los datos, liberando así al usuario de esa tarea.

### ***4. Administración de la seguridad:***

Crea un sistema de seguridad que establece unas reglas que determinan que usuarios pueden acceder a la base de datos, a que datos pueden tener acceso y qué operaciones pueden realizar.





### ***5. Control de acceso de usuarios múltiples***


Es el control de concurrencia. Permite acceder a la base de datos a múltiples usuarios creando unos algoritmos complejos para no comprometer la integridad de la base de datos.

### ***6. Administración de tareas de respaldo y recuperación:***

Proporciona utilidades que permiten realizar procedimientos de respaldo y recuperación rutinarios y especiales, como un fallo en el suministro eléctrico, un sector defectuoso en el disco, etc.

### ***7. Administración de la integridad de los datos***

Controla que se cumplan las reglas de integridad de los datos y de las relaciones, con lo que se reduce al mínimo la redundancia y se aumenta la consistencia.






## ***8. Lenguajes de acceso a la base de datos e interfaces de programación de aplicaciones:***

Permite acceder a los datos utilizando un lenguaje de consulta.

Lenguaje de consulta: Se trata de un lenguaje que no tiene procedimientos.


Debemos sustituir la palabra término, por el término en cuestión y las siglas de la También permite a los programadores acceder a los datos mediante lenguajes de procedimientos y proporciona utilidades administrativas para crear, ejecutar y mantener la base de datos.





## ***9. Interfaces de comunicación de bases de datos:***

Permiten que la base de datos acepte solicitudes de usuarios conectados en una red de ordenadores y a través de internet. Esta comunicación con el SGBD puede establecerse de varias formas:

- Realizar peticiones mediante formularios desde el explorador de internet.
  - Publicar informes en internet que pueda explorar cualquier usuario.
  - Conectarse a otros sistemas mediante aplicaciones como correo electrónico.
- 



# Implementación de un SGBD



# Arquitectura de una Base de Datos

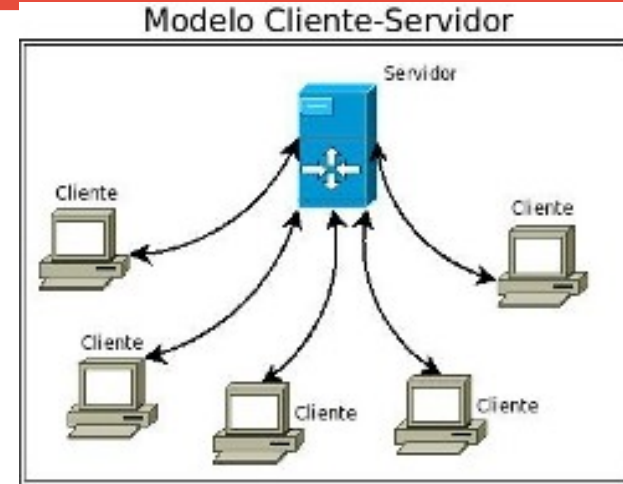
La arquitectura de un sistema de base de datos está influenciada en gran medida por el sistema informático subyacente en el que se ejecuta el sistema de base de datos. En la arquitectura de un sistema de base de datos se reflejan aspectos como la conexión de red, el paralelismo y la distribución.

- La conexión de red: de varias computadoras permite que algunas tareas se ejecuten en un sistema servidor y que otras se ejecuten en los sistemas clientes. Esta división de trabajo ha conducido al desarrollo de sistemas de base de datos cliente-servidor.
- El procesamiento paralelo: dentro de una computadora permite acelerar las actitudes del sistema de base de datos, proporcionando a las transacciones una respuesta más rápida, así como la capacidad de ejecutar más transacciones por segundo.
- La distribución de datos: A través de las distintas sedes o departamentos de una organización permite que estos datos residan donde han sido generados o donde son más necesarios, pero continuar siendo accesibles desde otros lugares o departamentos diferentes.



# Arquitectura cliente/servidor

El objetivo de un sistema de bases de datos es facilitar el desarrollo y ejecución de aplicaciones. Por tanto, desde un punto de vista amplio, un sistema de bases de datos posee una estructura compuesta de dos partes: un servidor y un conjunto de clientes.



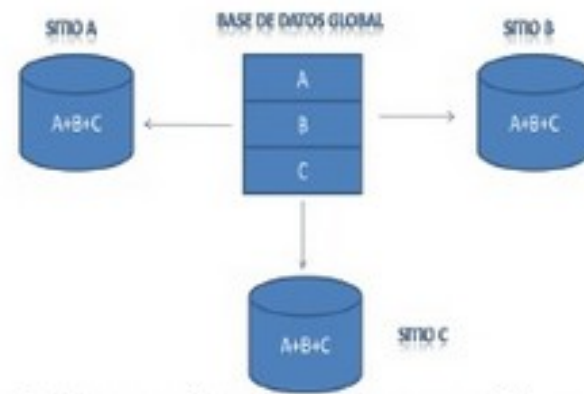
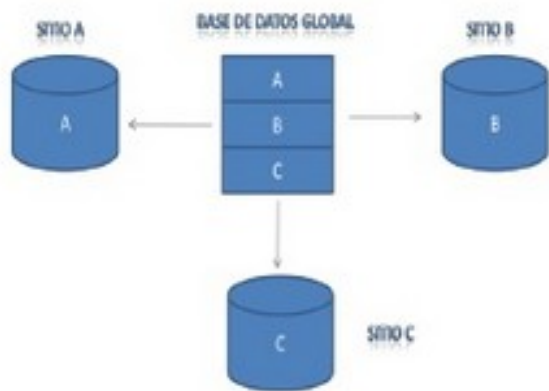
El **servidor** permite llevar a cabo las funciones propias del SGBD, se puede decir que *el* servidor es en sí, el SGBD.

Un **cliente** de una base de datos es cada consumidor de recursos de la base de datos: las aplicaciones del servidor, las aplicaciones de usuario y cualquier otro elemento de aplicación que acceda al servidor.

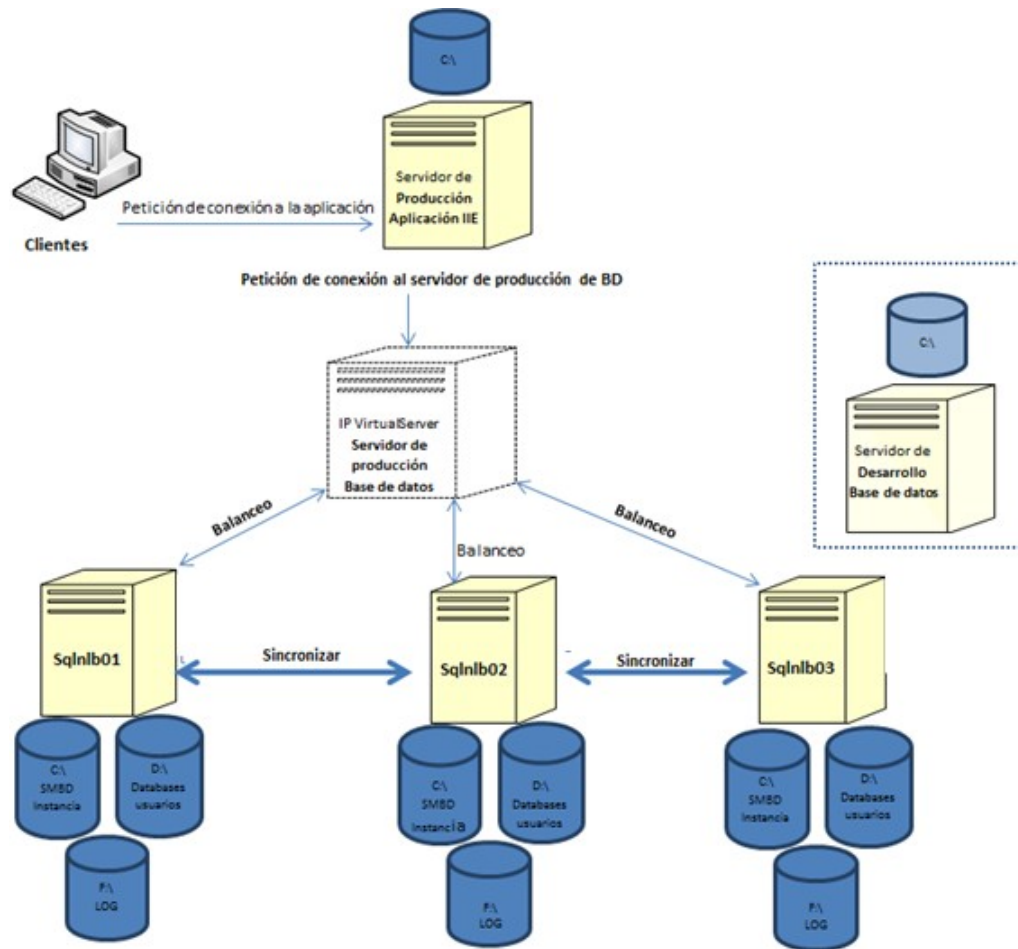
Los dos elementos de la base de datos, clientes y servidor, se pueden ejecutar en la misma máquina o en máquinas distintas, interconectadas a través de algún sistema de comunicación. Lo habitual es que sean máquinas distintas tal como se ve en el esquema.

# Bases de datos distribuidas

Una base de datos distribuida (BDD) es un conjunto de múltiples bases de datos lógicamente relacionadas las cuales se encuentran distribuidas en diferentes espacios lógicos y geográficos (pej. un servidor corriendo 2 máquinas virtuales) e interconectados por una red de comunicaciones. Dichas BDD tienen la capacidad de realizar procesamiento autónomo, esto permite realizar operaciones locales o distribuidas. Un sistema de Bases de Datos Distribuida (SBDD) es un sistema en el cual múltiples sitios de bases de datos están ligados por un sistema de comunicaciones de tal forma que, un usuario en cualquier sitio puede acceder los datos en cualquier parte de la red exactamente como si estos fueran accedidos de forma local.



# Alta disponibilidad en Bases de datos





# Modelo relacional



## Estructura del modelo relacional

La relación es el elemento básico en el modelo relacional y se puede representar como una tabla:

### Nombre

Atributo 1	Atributo 2	.....	Atributo n	
XXXXXXXXX	XXXXXXXXX	XXXXXXXXX	XXXXXXXXX	<b>Tupla 1</b>
XXXXXXXXX	XXXXXXXXX	XXXXXXXXX	XXXXXXXXX	<b>Tupla 2</b>
XXXXXXXXX	XXXXXXXXX	XXXXXXXXX	XXXXXXXXX	.
XXXXXXXXX	XXXXXXXXX	XXXXXXXXX	XXXXXXXXX	.
XXXXXXXXX	XXXXXXXXX	XXXXXXXXX	XXXXXXXXX	<b>Tupla n</b>

## Dominio y atributo

Un **dominio**  $D$  es un conjunto finito de valores homogéneos y atómicos caracterizados por un nombre; decimos homogéneos porque son todos del mismo tipo y atómicos porque son indivisibles.

Todo dominio ha de tener un nombre por el cual nos podamos referir a él y un tipo de datos; así el tipo de datos del dominio "nacionalidades" es una tira de caracteres de longitud 10.

Un **atributo** es el papel que tiene un determinado dominio en una relación.

## Relación

Matemáticamente, una relación se puede definir como un subconjunto del producto cartesiano de una lista de dominios, donde cada elemento de la relación, tupla, es una serie de  $n$  valores ordenados.



## Restricciones

En el modelo relacional, existen restricciones, es decir, estructuras u ocurrencias no permitidas, siendo preciso distinguir entre restricciones inherentes y restricciones de usuario.

### Restricciones inherentes

Además de las derivadas de la definición matemática de "relación" como eran que:

- No hay dos tuplas iguales.
- El orden de las tuplas no es significativo.
- El orden de los atributos (columnas) no es significativo.
- Cada atributo sólo puede tomar un único valor del dominio, no admitiéndose por tanto los grupos repetitivos.

Tenemos que la **regla de integridad de entidad** establece que "Ningún atributo que forme parte de la clave primaria de una relación puede tomar un valor nulo"; esto es, un valor desconocido o inexistente. Esta restricción debería aplicarse también a las claves alternativas, pero el modelo no lo exige.

## Restricciones de usuario

Podemos considerar la restricción de usuario, dentro del contexto relacional, como un predicado definido sobre un conjunto de atributos, de tuplas o de dominios, que debe ser verificado por los correspondientes objetos para que éstos constituyan una ocurrencia válida del esquema.

Dentro de las restricciones de usuario destaca la restricción de **integridad referencial** que dice que los valores de clave ajena deben coincidir con los de clave primaria asociada a ella o ser nulos.



# Ejemplos de SGBD



# ORACLE

**Oracle** es un sistema de gestión de base de datos objeto-relacional (o ORDBMS por el acrónimo en inglés de Object-Relational Data Base Management System), desarrollado por Oracle Corporation.

Se considera a Oracle como uno de los sistemas de bases de datos más completos, destacando:

- soporte de transacciones,
- estabilidad,
- escalabilidad y
- Soporte multiplataforma.

Su dominio en el mercado de servidores empresariales ha sido casi total hasta hace poco, recientemente sufre la competencia del Microsoft SQL Server de Microsoft y de la oferta de otros RDBMS con licencia libre como PostgreSQL, MySQL o Firebird. Las últimas versiones de Oracle han sido certificadas para poder trabajar bajo GNU/Linux.

<http://www.oracle.com/es/database/overview/index.html>

# MySQL

**MySQL** es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB —desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual.

Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C. Al contrario de proyectos como Apache, donde el software es desarrollado por una comunidad pública y los derechos de autor del código están en poder del autor individual, MySQL es patrocinado por una empresa privada, que posee el copyright de la mayor parte del código. Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado. Además de la venta de licencias privativas, la compañía ofrece soporte y servicios. Para sus operaciones contratan trabajadores alrededor del mundo que colaboran vía Internet. MySQL AB fue fundado por David Axmark, Allan Larsson y Michael Widenius.

<http://www.mysql.com/>

# SQL Server

**Microsoft SQL Server** es un sistema para la gestión de bases de datos producido por Microsoft basado en el modelo relacional. Sus lenguajes para consultas son T-SQL y ANSI SQL. **Microsoft SQL Server** constituye la alternativa de Microsoft a otros potentes sistemas gestores de bases de datos como son *Oracle*, *PostgreSQL* o *MySQL*.

Soporta transacciones.

Soporta procedimientos almacenados.

Incluye también un entorno gráfico de administración, que permite el uso de comandos DDL y DML gráficamente.

Permite trabajar en modo cliente-servidor, donde la información y datos se alojan en el servidor y los terminales o clientes de la red sólo acceden a la información.

Además permite administrar información de otros servidores de datos.

Este sistema incluye una versión reducida con el mismo motor de base de datos pero orientado a proyectos más pequeños, el SQL Express Edition, que se distribuye en forma *gratuita*.

<http://www.microsoft.com/es-es/server-cloud/products/sql-server/>

# PostgreSQL

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones no tiene nada que envidiarle a otras bases de datos comerciales.

PostgreSQL utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando.

<http://www.postgresql.org.es/>

# Comparativa

[https://es.wikipedia.org/wiki/Anexo:Comparaci%C3%B3n\\_de\\_sistemas\\_administradores\\_de\\_bases\\_de\\_datos\\_relacionales](https://es.wikipedia.org/wiki/Anexo:Comparaci%C3%B3n_de_sistemas_administradores_de_bases_de_datos_relacionales)



## *Otras:*

MariaSQL

Berkeley DB (librería usada por múltiples SGBD)

Firebird

DB2

Sybase

Teradata

Informix

Ingres

Amazon's SimpleDB

SQLite

...

MongoDB  
CouchDB



ORACLE®

PostgreSQL












TERADATA

# Magic Quadrant for Operational Database Management Systems

de Gartner 12 Octobre 2015



<https://www.gartner.com/doc/reprints?id=1-2PMFPEN&ct=151013>

Rank			DBMS	Database Model	Score		
Sep 2016	Aug 2016	Sep 2015			Sep 2016	Aug 2016	Sep 2015
1.	1.	1.	Oracle	Relational DBMS	1425.56	-2.16	-37.81
2.	2.	2.	MySQL 	Relational DBMS	1354.03	-3.01	+76.28
3.	3.	3.	Microsoft SQL Server	Relational DBMS	1211.55	+6.51	+113.72
4.	 5.	 5.	PostgreSQL	Relational DBMS	316.35	+1.10	+30.18
5.	 4.	 4.	MongoDB 	Document store	316.00	-2.49	+15.43
6.	6.	6.	DB2	Relational DBMS	181.19	-4.70	-27.95
7.	7.	 8.	Cassandra 	Wide column store	130.49	+0.26	+2.89
8.	8.	 7.	Microsoft Access	Relational DBMS	123.31	-0.74	-22.68
9.	9.	9.	SQLite	Relational DBMS	108.62	-1.24	+0.97
10.	10.	10.	Redis	Key-value store	107.79	+0.47	+7.14

<http://db-engines.com/en/ranking>