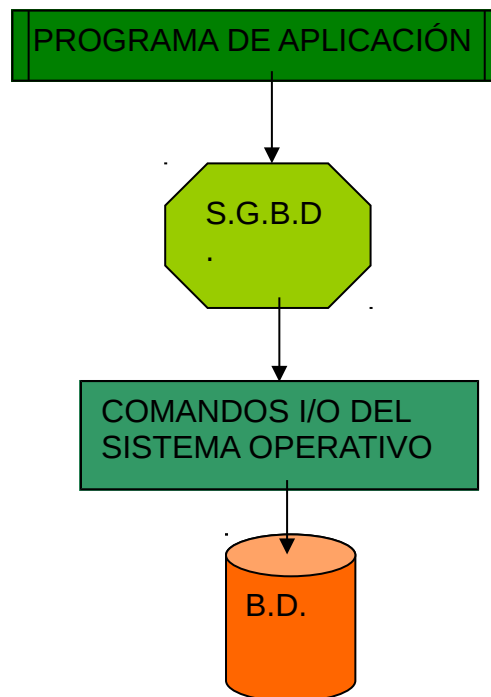


U.T. I.- Características de los Sistemas Gestores de Bases de Datos.

1.- Sistemas Gestores de Bases de Datos.

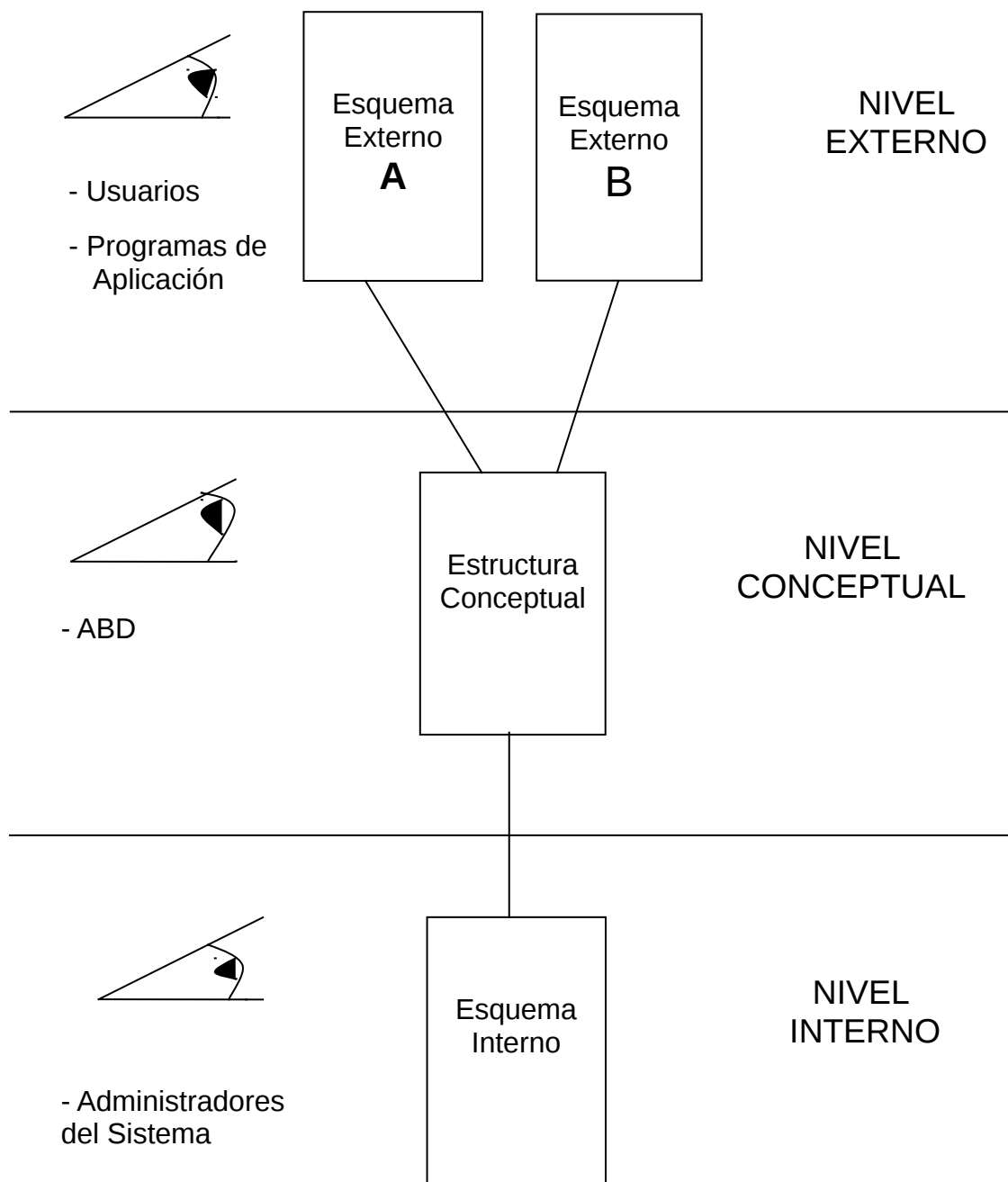
- Un SGBD es el software que permite gestionar BBDD, ocultando la parte física de la misma y permitiendo manejarla desde un nivel conceptual.
- Este software permite separar las aplicaciones de los datos, de modo que los programas deben "negociar" con el SGBD el acceso a los datos.



- Los SGBDs han crecido de manera exponencial estos últimos años debido al éxito de Internet, que ha provocado el acceso a miles de BBDD por parte de millones de usuarios cada día.

2.- Arquitectura del SGBD. Modelo ANSI/SPARC

- El grupo de trabajo ANSI/SPARC diseñó un modelo en el que se indicaba cómo debía funcionar un SGBD para asegurar la separación entre datos y aplicaciones y así especificó tres niveles:



- o **Nivel interno.** Describe la estructura física de la BD mediante un "esquema interno", es decir, cómo está almacenada la información en el dispositivo físico.
 - Hace referencia a cómo se organizarán los datos en ficheros, en qué dispositivos se almacenarán, si va a ser una BD distribuida, S.O. utilizado, configuración de servidores, política de copias de seguridad, etc.

- La gestión del nivel interno es responsabilidad del Administrador del Sistema.
- o **Nivel conceptual.** Describe la estructura lógica de la BD mediante un "esquema conceptual". En él se incluyen todos los objetos, y asociaciones entre ellos, que componen la BD.
 - En este nivel no se tiene en cuenta cómo están almacenados realmente estos objetos.
 - La implementación y mantenimiento del esquema conceptual es labor del ABD. Para ello suele partirse de un esquema E/R.
- o **Nivel externo.** Este nivel se describen varios esquemas externos o vistas de usuario/programa.
 - Un esquema externo comprende un subconjunto del conjunto total de objetos y asociaciones existentes en la BD, únicamente los que interesan a los usuarios y/o programas autorizados a utilizarlo.
 - Los usuarios y programas manejan los diferentes esquemas externos de este nivel puestos a su disposición por el Administrador de la BD (ABD).
- Mediante esta arquitectura se asegura la independencia de los datos desde dos puntos de vista:
 - o **Independencia lógica:** Permite modificar el esquema conceptual sin tener que modificar los diferentes esquemas externos ni, por ende, los programas de aplicación que los utilizan.
 - Por ejemplo ampliar la BD.
 - o **Independencia física:** Permite modificar el esquema interno, sin tener que modificar el esquema conceptual.
 - Por ejemplo reorganizar algunos ficheros físicos con el fin de mejorar el rendimiento de determinadas operaciones de consulta cuyo tiempo de respuesta se considere crítico.

3.- Funciones del SGBD.

- Un Sistema Gestor de Bases de Datos debe permitir realizar las siguientes funciones:

- o **Función de descripción o definición.** Permite al diseñador de la BD crear las estructuras apropiadas para integrar adecuadamente los datos.
 - Esta función se realiza mediante el Lenguaje de Definición de Datos (LDD). Mediante este lenguaje:
 - o Se definen las estructuras de datos (metadatos).
 - o Se definen las relaciones entre ellas.
 - o Se definen las reglas que han de cumplir.
- o **Función de manipulación.** Permite consultar y actualizar los datos de la BD.
 - Esta función se realiza mediante el Lenguaje de Manipulación de Datos (LMD). mediante este lenguaje se puede:
 - o Añadir datos.
 - o Eliminar datos.
 - o Modificar datos.
 - o Consultar datos.
- o **Función de control.** Mediante esta función los ABDs poseen mecanismos para proteger las vistas de datos permitidas a cada usuario o programa.
 - Proporciona además elementos para la creación de usuarios y permisos.
 - Esta función se realiza mediante el Lenguaje de Control de Datos (LCD).
- Para garantizar que un SGBD cumple las funciones descritas anteriormente y además mantiene la independencia entre los tres niveles descritos en el apartado anterior, Edgar F. Codd (teórico de la BBDD relacionales en los años 70) especificó 12 reglas que deberían cumplir.
- En la actualidad algunas reglas han quedado un tanto obsoletas. En resumen, las funciones que se esperan de un buen SGBD son:
 - o Que estén dotados de un lenguaje que permita crear los elementos de la BD y gestionar su diccionario de datos.

- Normalmente este lenguaje es el SQL (Structured Query Language), aunque cada fabricante de SGBDs impone variantes al SQL estándar.
- o Que incorporen herramientas gráficas que faciliten tareas habituales tanto de gestión como de administración de la BD.
 - Los paquetes actuales suelen incorporar un IDE (Entorno de Desarrollo Integrado) para facilitar el trabajo tanto del administrador como de los usuarios.
- o Deben poder establecerse restricciones y reglas de integridad avanzadas e incluirlas como parte de la BD. Como mínimo, deberá incluir las siguientes:
 - Restricciones **PRIMARY KEY** (establecimiento de claves primarias en las tablas), **NOT NULL** (inexistencia de nulos), **UNIQUE** (unicidad), **CHECK** (cumplimiento de condiciones simples).
 - Reglas de integridad referencial entre tablas: **FOREIGN KEY**.
 - También deben poder establecerse restricciones más complejas que las que pueden incluirse en un **CHECK** por medio de disparadores (**TRIGGERS**).
- o Deben permitir la gestión de copias de seguridad.
 - Esta es una de las funciones críticas del SGBD, porque permitirá la recuperación de la información en caso de problemas.
- o Deben permitir restablecer la integridad de la BD en caso de desastre (fallos de hardware o de alimentación).
 - Si ocurre un desastre durante la ejecución de una instrucción de modificación de datos, la BD debe volver al estado anterior al comienzo de dicha ejecución.
- o Deben incorporar aplicaciones de importación/exportación de datos, con el objeto de poder utilizar los datos en otros SGBDs.
 - Algunos SGBDs con código propietario no permiten esta reutilización de los datos en sistemas de la competencia.
- o Deben incorporar ficheros LOG.
 - Los ficheros LOG permiten examinar las incidencias y monitorizar el funcionamiento de la BD.

- o Deben incluir herramientas para programar aplicaciones y automatizar tareas.
- o Deben permitir distribuir la BD entre varias máquinas y así mejorar la disponibilidad de los datos.
- o Deben incluir optimización de consultas con el fin de reducir el tiempo de respuesta en operaciones con los datos.
- o Deben incluir gestión de transacciones ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad).
 - Norma obligatoria que deben cumplir las BBDD para que una transacción pueda ser considerada como tal.

4.- Tareas del Administrador de Bases de Datos.

- Entendemos por Administrador de una Base de Datos (ABD) a la persona o conjunto de personas encargadas de la gestión global de la base de datos.
- Los ABDs tienen a su cargo una serie de tareas de vital importancia, algunas de ellas especialmente críticas.
- Las más comúnmente aceptadas como parte de la profesión de un ABD son las siguientes:
 - o Configurar e instalar el hardware necesario.
 - Esto incluye la gestión de la memoria interna, discos duros, así como gestión del S.O. para que la BD funcione correcta y rápidamente.
 - o Instalación y mantenimiento del SGBD.
 - Seleccionando la más adecuada forma de instalación y configurando lo necesario para su óptimo rendimiento acorde con las necesidades.
 - También deberá llevar a cabo las actualizaciones de software necesarias.
 - o Creación de las estructuras de almacenamiento.
 - Consiste en crear y configurar las estructuras físicas y los elementos lógicos que permitan un rendimiento optimizado de la BD.
 - o Creación y configuración de la BD.

- Consiste en crear la estructura lógica global de la BD (tablas, usuarios, permisos, vistas, procedimientos, funciones, disparadores, ...).
- El ABD deberá establecer políticas de seguridad en lo que se refiere a usuarios y permisos.
- Esta es la fase más crítica en la administración de una BD.
- o Monitorizar y optimizar el rendimiento de la BD.
 - Un ABD debe detectar los "cuellos de botella" y actuar en consecuencia.
 - Esto incluye optimizar las instrucciones SQL, por lo que deberá asistir a los desarrolladores de software para que utilicen las instrucciones más eficientes sobre la BD.
- o Realizar tareas de copia de seguridad y recuperación.
 - Deberá llevar a cabo acciones para que los datos puedan ser recuperados en caso de catástrofe.

5.- Opciones de funcionamiento de un SGBD.

5.1.- SGBD monocapa.

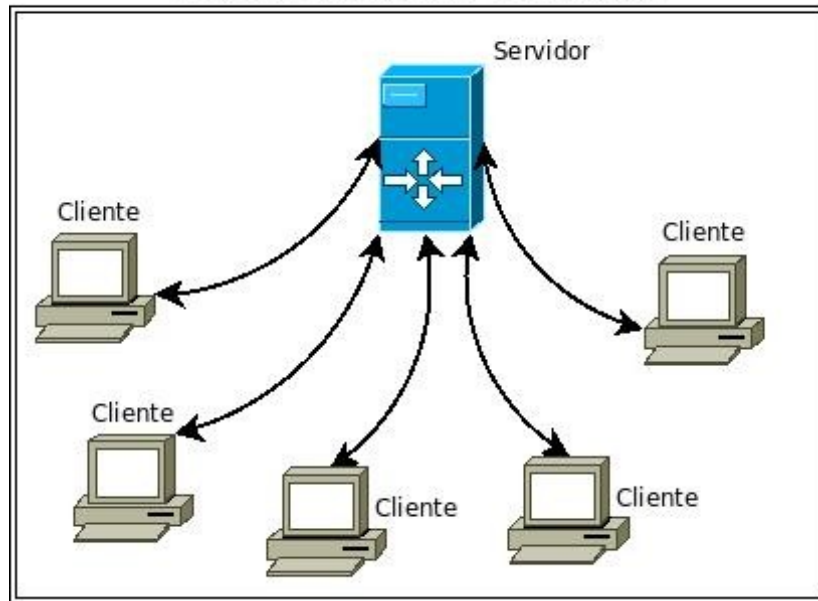
- Se trata de sistemas gestores y datos instalados en una máquina a la que se conectan los usuarios y administradores.
- Es un modelo que se utiliza con BBDD pequeñas y pocas conexiones.

5.2.- SGBD de dos capas.

- Es un modelo tipo cliente/servidor: La BD y el sistema gestor se alojan en un servidor al que se conectan los administradores y usuarios desde sus máquinas cliente.
 - o Un software de comunicaciones se encarga de permitir el acceso a través de la red.
- Existen dos posibilidades:
 - o Arquitectura cliente/servidor único.
 - Un solo servidor gestiona la BD.

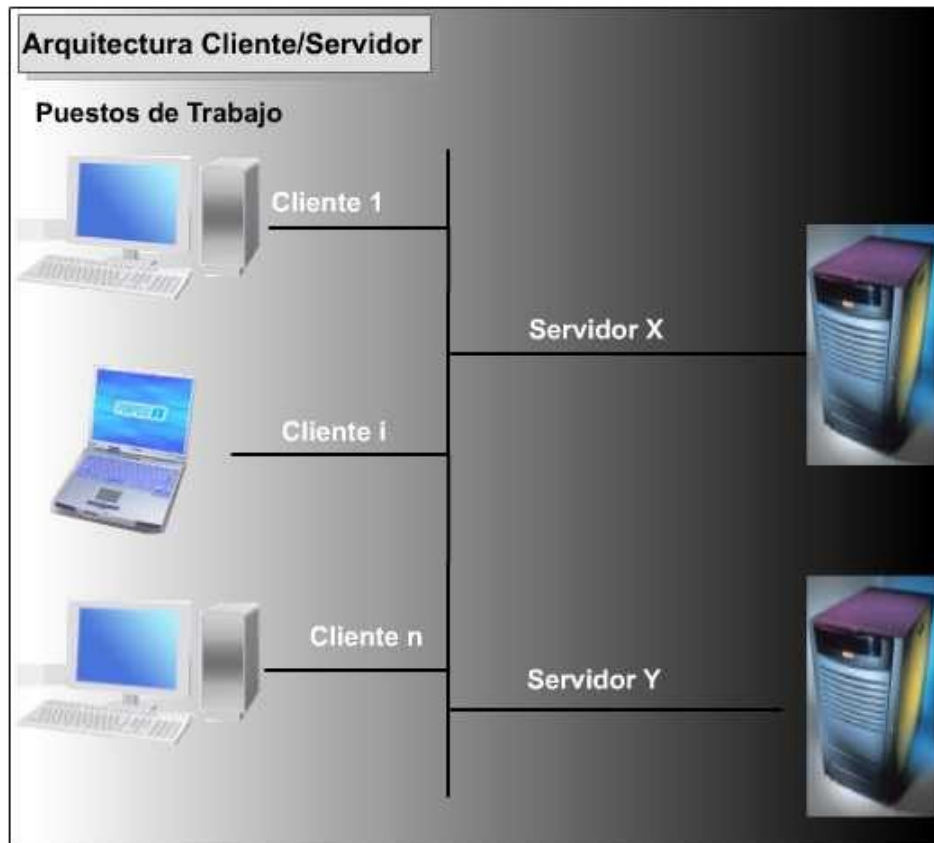
- Todos los clientes se conectan a él para realizar sus peticiones de datos.

Modelo Cliente-Servidor



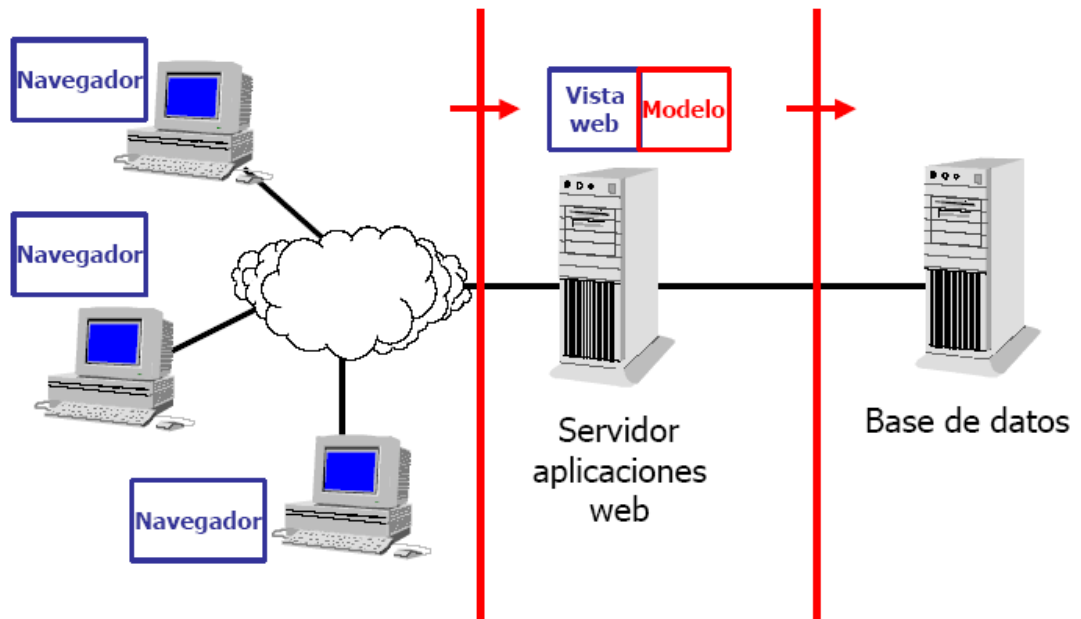
o Arquitectura cliente/multiservidor.

- La BD se distribuye entre varios servidores.
- El usuario no sabe a qué servidor conecta realmente, el software de control de comunicaciones se encargará de dirigirle al servidor adecuado.



5.3.- SGBD de tres o más capas.

- En este caso entre el cliente y el servidor hay al menos una capa intermedia (puede haber más de una).
- Esta capa o capas se encargan de recoger las peticiones de los clientes y de comunicarse con el servidor o servidores para recibir la respuesta y enviársela al cliente.
- Un caso típico es que la capa intermedia sea un servidor web, que recibe las peticiones a través de páginas web.
 - o De este modo, para conectarse a la BD el usuario sólo precisa de un navegador web.



- El modelo de tres o más capas es el que más se está potenciando en la actualidad por motivos de seguridad y portabilidad de la BD.

6.- Sistemas Gestores de Bases de Datos comerciales.

- En lo que respecta a la protección del software podemos distinguir entre SGBDs de código cerrado y de código abierto.
 - o Los defensores del código cerrado argumentan que es lógico protegerlo para evitar que empresas competidoras copien su tecnología.
 - o Los defensores del código abierto consideran que de este modo éste puede ser modificado por programadores de todo el mundo que pueden aportar mejoras.

6.1.- SGBDs de código cerrado.

- **Oracle.** Propiedad de Oracle Corporation.
 - o Es el SGBD relacional más antiguo y también el más influyente, dado que la mayoría de las mejoras al SQL original se desarrollaron sobre él.
 - o Sigue siendo uno de los SGBDs comerciales más utilizados.
 - o Presume de su gran estabilidad y escalabilidad, un control avanzado de transacciones y de su lenguaje PL/SQL.

- o Es un SGBD multiplataforma con certificación para instalarse en Linux, aunque en unas condiciones concretas.
- **DB2.** Propiedad de IBM.
 - o Desarrollada para Windows, UNIX y Linux, implementa XML de manera nativa.
 - o Dispone de amplias facilidades de migración de datos, especialmente desde Oracle, así como uso de transacciones avanzadas.
- **SQL Server.** Propiedad de Microsoft Corporation.
 - o Originalmente basado en el SGBD SysBase, es distribuido sólo para Windows.
 - o Dispone de una gran escalabilidad, estabilidad, uso de transacciones, entorno gráfico avanzado.
 - o Es muy utilizado por los programadores de la plataforma .NET, también propiedad de Microsoft.
- Los tres son los SGBDs más utilizados en la actualidad por su contrastada potencia.
 - o Ninguno de los tres cumple completamente los estándares y aportan sus propios lenguajes y formas de trabajo.
 - o Los tres disponen de versiones gratuitas limitadas para uso personal.

6.2.- SGBDs de código abierto.

- **MySQL.** Desarrollado por la empresa MySQL AB, que fue adquirida por Sun Microsystems que a su vez ha sido comprada por Oracle Corporation.
 - o Es considerado como el principal SGBD por la comunidad de programadores de código abierto, siendo en la actualidad el principal gestor de BBDD de la red.
 - o Mantiene su licencia GPL (General Public License, permite modificar el código y redistribuirlo, pero manteniéndolo abierto), aunque posee una segunda licencia cerrada para opciones de trabajo más avanzadas.

- o Es muy popular por su asociación con PHP, su buena estabilidad, gran escalabilidad y uso de transacciones y lenguaje procedimental.
- o Es multiplataforma, pudiendo instalarse sin problemas en la mayoría de los SSOO conocidos.
- **PostgreSQL.** Versión de código abierto basado en el producto Ingres, de la Universidad de Berkeley.
 - o Usa licencia de tipo MIT (del Instituto Tecnológico de Massachusetts), que permite su modificación y redistribución incluso como código cerrado.
 - o Está considerado el SGBD de código abierto más potente y fidedigno con los estándares.
 - o Posee uso de transacciones avanzadas, lenguaje procedimental, gran estabilidad y escalabilidad.
 - o A partir de su núcleo se han creado otros productos libres de software de BBDD.
- **SQLite.** Proyecto de dominio público creado por Richard Hipp.
 - o Está desarrollado en C y es compatible con ACID.
 - o Es un SGBD que se almacena en una biblioteca en C y ocupa muy poco espacio,
 - o A diferencia de los gestores basados en la arquitectura cliente/servidor, el motor de SQLite no es un proceso independiente con el que el programa principal se comunica si no que se enlaza con el programa pasando a formar parte del mismo.
 - o Debido a los dos apartados anteriores es muy popular entre los desarrolladores de aplicaciones para dispositivos móviles.

7.- Arquitectura de un SGBD.

- La arquitectura de un SGBD hace referencia al modelo interno de funcionamiento del sistema, es decir, a las estructuras internas/físicas que proporciona para el almacenamiento de datos y su relación con las estructuras lógicas/conceptuales.
 - o Cada SGBD dispone de diferentes arquitecturas.

7.1.- Estructuras lógicas de una BD.

- **Tablas.** Son estructuras formadas por filas y columnas en las que se almacenan los datos
 - o La mayoría de los SGBDs utilizan distintos tipos de tablas, pero en general cuando se habla de tablas se habla del elemento lógico encargado de almacenar los datos.
- **Restricciones.** Se definen al crear las tablas, aunque se almacenan aparte.
 - o Las restricciones se encuentran disponibles en el diccionario de datos y marcan las reglas que han de cumplir los datos para poder ser considerados válidos.
- **Índices.** Un índice es una lista ordenada de claves que permite acceder al contenido de una o más columnas de una forma más rápida.
- **Vistas.** Son consultas almacenadas que nos permiten mostrar de forma personalizada los datos contenidos en una o más tablas.
- **Procedimientos, funciones y disparadores.** Se trata de bloques de código del lenguaje procedimental del SGBD (SQL) que se utilizan para realizar acciones sobre las tablas.

7.2.- Estructuras físicas de una BD.

- Al final todos los elementos lógicos deben almacenarse en ficheros cuyo tamaño, dirección, etc. debe ser controlado por el ABD.
- Si en el caso de las estructuras lógicas existen diferencias entre los distintos SGBDs, en el caso de las estructuras físicas esa diferencia puede ser total, lo que obliga a conocer muy bien la parte interna del sistema que se esté utilizando.