

# Administración de sistemas de bases de datos

## TEMA 43

---

ABACUS NT

Oposiciones 2021

## **Índice**

---

- 1. Introducción**
- 2. Sistemas Gestores de Bases de Datos (SGBD)**
  - 2.1. Concepto**
  - 2.2. Tareas del SGBD**
  - 2.3. Requisitos de un SGBD**
  - 2.4. Abstracción de la información**
- 3. Componentes de un SGBD**
  - 3.1. Lenguajes para la definición y manipulación de datos**
    - 3.1.1. Lenguajes para la definición de datos
    - 3.1.2. Lenguajes para la manipulación de datos
  - 3.2. Procesador de consultas.**
  - 3.3. Gestor de la base de datos.**
  - 3.4. Utilidades**
  - 3.5. Componentes físicos**
- 4. El administrador de la base de datos (DBA)**
  - 4.1. Usuarios de la base de datos**
  - 4.2. El administrador**
  - 4.3. Funciones del administrador de la base de datos**
  - 4.4. Relaciones con otros usuarios**
  - 4.5. Administración de datos y administración de bases de datos**
    - 4.5.1. Administrador de datos
    - 4.5.2. Administrador de la Base de Datos
  - 4.6. Herramientas del administrador de la base de datos**
- 5. Protección de datos: seguridad, integridad y confidencialidad**
- 6. Conclusión**
  - 6.1. Relación del tema con el sistema educativo actual**
- 7. Bibliografía**

## 1. Introducción

Las bases de datos surgen con la necesidad del almacenamiento de grandes volúmenes de datos. Los pioneros en Ciencias de la Computación de finales de los 50 y principios de los 60 desarrollaron muchas técnicas nuevas para manipular los ficheros. Los ficheros se pueden manipular mediante lenguajes de tercera generación convencionales empleando los servicios del sistema operativo.

La complejidad del trabajo de programación y la rigidez de las restricciones que imponía la tecnología de los ordenadores en aspectos como la inserción de datos o la gestión de casos especiales, hacía arduo el trabajo.

En una aplicación convencional con ficheros, éstos se diseñan de acuerdo con los programas. Esto es, una vez planteado el programa, se decide si debe haber ficheros, cuántos deben ser, qué organización tendrán, qué información contendrá cada uno, qué programas actuarán sobre ellos y cómo lo harán.

Esto tiene la ventaja, en principio, de que los programas son bastante eficientes, ya que la estructura de un fichero está pensada para el programa que lo va a usar; sin embargo, conlleva graves inconvenientes. En una aplicación convencional con ficheros aparecen, pues, los siguientes problemas:

- **Dificultad de mantenimiento.** Si hay ficheros con información duplicada, realizar las actualizaciones necesarias puede ser complejo y costoso. Incluso puede ser necesario utilizar archivos con diferentes organizaciones. Si la actualización no se realiza correctamente en todos los lugares se producirán inconsistencias en la base de datos.
- **Redundancia.** Este problema consiste en tener datos que no aportan información, ya que se pueden deducir de otros datos. El caso típico de redundancia es mantener el mismo dato almacenado en varios sitios.
- **Rigidez de búsqueda.** A cada fichero se le dará una determinada organización, de acuerdo con los tipos de acceso que se creían más frecuentes, pero puede ocurrir que se necesiten otros modos de acceso y sean difíciles de llevar a cabo sobre la organización ya existente.
- **Dependencia con los programas.** Las relaciones entre los datos almacenados en los ficheros no están presentes en éstos. Es el programa que los utiliza el que se encarga de ello. Esto implica que cualquier modificación de la estructura de un fichero obliga a modificar todos los programas que lo usen.
- **Confidencialidad y seguridad.** En general, es necesario impedir la consulta de determinados datos a determinados usuarios (confidencialidad) e impedir modificaciones efectuadas por usuarios no autorizados (seguridad). Si trabajamos convencionalmente con ficheros, el control deberá realizarlo el propio programa. En cualquier caso, no se podrá impedir que alguien construya un programa para modificar o ver el contenido de un fichero, si el sistema operativo le permite el acceso.

Además, en un sistema tradicional de ficheros, cada una de las aplicaciones deberá realizar la descripción de los registros, así como determinar la organización de ficheros, los tipos de acceso, etc. En un entorno de base de datos dichas especificaciones las realiza el SGBD, y es el administrador de la base de datos el encargado de realizar éstas y otras funciones sobre el sistema.

## 2. Sistemas Gestores de Bases de Datos (SGBD)

### 2.1. Concepto

Según la Real Academia de la Lengua, una BD es la **representación de una información de manera adecuada para su tratamiento por un ordenador**.

Además, es necesario distinguir entre la base de datos BD y el sistema que la gestiona (SGBD ó DBMS – Data Base Management System-)

Así, un SGBD es una colección de programas de aplicación que proporcionan al usuario de la base de datos los medios necesarios para realizar las siguientes tareas:

1. Definición de los datos a los distintos niveles de abstracción (físico, lógico y externo).
2. Manipulación de los datos en la base de datos. Es decir, la inserción, modificación y borrado y acceso o consulta a los mismos.
3. Mantenimiento de la integridad de la BD. Integridad en cuanto a los datos en sí, sus valores y las relaciones entre ellos.
4. Control de la privacidad y seguridad de los datos en la BD.

Se denomina **sistema de gestión de base de datos** al conjunto de software destinado a la creación, control y manipulación de la información sobre una base de datos.

### 2.2. Tareas del SGBD

Concretamente, un SGBD debe permitir la realización de las siguientes tareas:

- **Definición del esquema de la base de datos.** Una vez diseñado el esquema de la base de datos, hemos de describirlo mediante un conjunto de instrucciones. Esto se realiza mediante un lenguaje específico, denominado **lenguaje de definición de datos (DDL)**.
- **Acceso a los datos desde un lenguaje de alto nivel.** Esto se realiza también mediante un lenguaje específico, denominado **lenguaje de manipulación de datos (DML)**. El DML puede ser utilizado de dos formas diferentes. Por una parte, se incluyen sentencias DML en programas escritos en lenguaje de alto nivel. Esto hace necesario incluir en el SGBD un precompilador que traduzca las instrucciones DML en instrucciones reconocibles por el compilador del lenguaje de alto nivel (lenguaje anfitrión). La otra forma de utilización del DML es mediante programas que contengan exclusivamente sentencias propias de este lenguaje.

- **Acceso a la información en modo conversacional.** El SGBD debe incorporar una interfaz de usuario a través de la cual introducir sentencias directamente desde un terminal, para obtener información interactiva.
- **Gestión de ficheros.** Función realizada por un módulo gestor de ficheros que se encarga de la comunicación con el sistema operativo. Además, realiza otras funciones, tales como control de usuarios, recuperar la información tras fallos del sistema, organización física de la base de datos, control de seguridad y privacidad, y gestión de accesos concurrentes.

### 2.3. Requisitos de un SGBD

Los requisitos que debe cumplir un buen sistema de base de datos son:

- **Acceso múltiple.** Varios usuarios pueden acceder simultáneamente a la base de datos, sin que se produzcan conflictos ni vistas incoherentes.
- **Utilización múltiple.** Cada usuario podrá tener una imagen o visión particular de la estructura de la base de datos.
- **Flexibilidad.** Se podrán utilizar distintos métodos de acceso.
- **Confidencialidad y seguridad.** Se controlará el acceso a los datos, impidiéndoselo a los usuarios no autorizados.
- **Protección contra fallos.** Deben existir mecanismos concretos de recuperación en caso de fallo del ordenador.
- **Independencia física.** Se puede cambiar el soporte físico de la base de datos sin que repercuta en la base de datos ni en los programas que la usan.
- **Independencia lógica.** Se pueden modificar los datos contenidos en la base, las relaciones existentes entre ellos o incluir nuevos datos, sin que afecte a los programas que la usan.
- **Redundancia controlada.** Los datos se almacenan una sola vez, excepto en casos excepcionales.
- **Interfaz de alto nivel.** Existe una forma sencilla y cómoda de utilizar la base de datos desde un lenguaje de programación de alto nivel.
- **Interrogación directa (query).** Existe una utilidad que permite el acceso a los datos de forma conversacional.

### 2.4. Abstracción de la información

El SGBD debe proporcionar información a usuarios y desarrolladores a distintos niveles, representando cada uno de ellos una abstracción de datos:

- **Nivel de visión.** A este nivel, cada grupo de usuarios posee conocimiento únicamente de aquella parte de la base de datos que le afecta. El usuario sabe de la existencia de los datos y su significado, pero ignora los detalles sobre su formato, tipo, estructura, y, en general, cualquier aspecto físico.
- **Nivel conceptual.** La unión de todas las vistas da lugar a este nivel. En él se conoce la descripción de todos los datos y las relaciones existentes entre ellos.

- **Nivel físico.** En este nivel se describe cómo se encuentran los datos almacenados físicamente en memoria secundaria. Es el nivel más cercano al hardware y se encuentra íntimamente ligado a él.

### 3. Componentes de un SGBD

Un sistema gestor de bases de datos se divide en módulos que se encargan de las responsabilidades del sistema. Algunas de estas funciones las proporciona el sistema operativo.

Los componentes funcionales principales de un sistema gestor de base de datos son el procesador de consultas y el gestor de la base de datos. Además, serán necesario los soportes y estructuras para su almacenamiento físico.

La arquitectura completa está dividida en dos fases:

**Definición de datos.** La creación de la base de datos comienza con la elaboración del esquema conceptual, que se procesa utilizando una herramienta CASE que lo convierte en los metadatos. Utilizando esta información, se construyen los esquemas interno y externo.

**Manipulación de datos.** El usuario puede realizar operaciones sobre la base de datos. Esta petición es transformada por el transformador externo/conceptual que obtiene el esquema correspondiente ayudándose también de los metadatos. El resultado lo convierte otro transformador en el esquema interno usando también la información de los metadatos. Finalmente, del esquema interno se pasa a los datos usando el transformador que también accede a los metadatos y de ahí se accede a los datos. Para que los datos se devuelvan al usuario en formato adecuado para él se tiene que hacer el proceso contrario.

#### 3.1. Lenguajes para la definición y manipulación de datos

Según el autor **C.J Date**, en su libro *Introduction to Database Systems* (2003), “una base de datos es un conjunto de datos persistentes que es utilizado por los sistemas de aplicación de alguna empresa dada” (entendiéndose como empresa una organización).

Para trabajar con dicho conjunto de datos, la arquitectura ANSI/X3/SPARC diferencia dos tipos de lenguajes:

- **Lenguaje de definición de datos:**
- **Lenguaje de manipulación de datos**

A continuación, vamos a ver en qué consiste cada uno de ellos

##### 3.1.1. Lenguajes para la definición de datos

El lenguaje de descripción o definición de datos (DDL, Data Definition Language), propio de cada SGBD, permite al desarrollador de la base de datos especificar los elementos de datos

que integran su estructura, y las relaciones que existen entre ellos, las reglas de integridad semántica, las características de tipo físico y las vistas lógicas de los usuarios.

Generalmente, los DDL de los diferentes SGBD son lenguajes muy simples basados en una gramática muy sencilla.

La representación de los datos compilación es almacenada en denominado **Diccionario de Datos**.

el DDL cuenta con un **sublenguaje** encargado del control y seguridad de los datos, el cual se denomina **Lenguaje de Control de Datos (DCL)** y permite el control del acceso de a la información almacenada en el diccionario de datos (definición de privilegios y tipos de acceso), así como el control de seguridad de los datos.

### 3.1.2. Lenguajes para la manipulación de datos

Una vez descrita la estructura de la base de datos, los usuarios utilizarán un lenguaje de manipulación de datos (DML, Data Manipulation Language) para realizar las siguientes operaciones sobre dicha estructura:

- **Recuperar información:** Consultar la base de datos.
- **Actualizar la información:** La actualización supondrá tres tipos de operaciones
  - Inserción
  - Borrado
  - Modificación de los datos.

Al igual que el DDL, el DML está basado en un modelo de datos y, por tanto, los SGBD basados en distintos modelos de datos tienen diferente DML. Se trata también de un lenguaje basado en una gramática completa, sencilla y, generalmente, fácil de entender por usuarios no expertos.

Los SGBD también son capaces de procesar peticiones de DML que se han formulado desde programas escritos en otros lenguajes de programación

Dentro de los **DML**, podemos destacar los siguientes ejemplos comerciales:

- **SQL (Structured Query Language):** Lo estudiaremos en el siguiente apartado.
- **QBE (Query By Example):** Creado a principios de los años 70. Hay varias **implementaciones** de este lenguaje, como el original de IBM (Sistema QBE), o QBE de Microsoft (en Access). Se fundamenta en el cálculo relacional orientado a dominios.
- **QUEL (Query Language):** Basado en el cálculo relacional orientado a tuplas.

### 3.2. Procesador de consultas.

Es el componente principal, transforma las consultas en un conjunto de instrucciones de bajo nivel que se dirigen al gestor de la base de datos:

- **Compilador de DML** (Data Manipulation Language). Traduce las instrucciones de DML a instrucciones de bajo nivel que entiende el motor de evaluación de consultas. Además, intenta transformar las peticiones del usuario en otras equivalentes, pero más eficientes, encontrando así una buena estrategia para ejecutar la consulta.
- **Precompilador de DML embebido**. Convierte las instrucciones de DML embebidas en un programa de aplicación en llamadas a procedimientos normales en el lenguaje anfitrión. Interactúa con el compilador de DML para generar el código apropiado.
- **Intérprete de DDL** (Data Definition Language). Interpreta las instrucciones de DDL y las registra en un conjunto de tablas que contienen metadatos (catálogo).
- **Motor de evaluación de consultas**. Ejecuta las instrucciones de bajo nivel generadas por el compilador de DML.

### 3.3. Gestor de la base de datos.

Proporciona la interfaz entre los datos de bajo nivel almacenados en la base de datos y los programas de aplicación. Acepta consultas y examina los esquemas externo y conceptual para determinar qué registros se requieren para satisfacer la petición.

Entonces realiza una llamada al gestor de ficheros para ejecutar la petición:

- **Gestor de autorización e integridad**. Comprueba que se satisfagan las restricciones de integridad y la autorización de los usuarios para acceder a los datos.
- **Gestor de transacciones**. Asegura que la base de datos quede en un estado consistente a pesar de los fallos del sistema, y que las ejecuciones de transacciones concurrentes ocurran sin conflictos.
- **Gestor de ficheros**. Maneja los ficheros en disco en donde se almacena la base de datos utilizando los métodos de acceso del sistema operativo que se encargan de leer o escribir los datos en el buffer del sistema. Este gestor establece y mantiene la lista de estructuras e índices definidos en el esquema interno.
- **Gestor de buffers**. Es responsable de traer los datos del disco de almacenamiento a memoria principal, y decidir qué datos tratar en la memoria caché.
- **Gestor del diccionario de datos**. Controla los accesos al diccionario de datos y se encarga de mantenerlo.
- **Gestor de recuperación**. Este módulo garantiza que la base de datos permanece en un estado consistente en caso de que se produzca algún fallo.
- **Planificador**. Este módulo es el responsable de asegurar que las operaciones que se realizan concurrentemente sobre la base de datos tienen lugar sin conflictos.
- **Optimizador de consultas**. Este módulo determina la estrategia óptima para la ejecución de las consultas.

### 3.4. Utilidades

Son aplicaciones que facilitan el trabajo a los usuarios y programadores. Tienen la característica común de tener un interfaz fácil de entender. Se basan en menús que guían al usuario para conseguir el objetivo final.

La mayoría de los SGBD incluyen:

- **Generador de menús:** Diseña la interfaz de usuario de una aplicación.
- **Generador de Informes:** Presenta datos en pantalla o impresora con un formato predefinido o fácil de definir sin conocer lenguajes de BD o programación.
- **Generador de Formularios:** Genera pantalla de diálogos que permiten la introducción sencilla de información; bien por teclado, bien por botones.

### 3.5. Componentes físicos

Los componentes físicos de un sistema gestor de base de datos son los siguientes:

- **Ficheros de datos.** Almacenan la base de datos en sí.
- **Diccionario de datos.** Almacena metadatos acerca de la estructura de la base de datos.
- **Índices.** Proporcionan acceso rápido a elementos de datos que tienen valores particulares.

**Datos estadísticos.** Almacenan información estadística sobre los datos en la base de datos. El procesador de consultas usa esta información para ejecutar una consulta.

## 4. El administrador de la base de datos (DBA)

### 4.1. Usuarios de la base de datos

Se consideran tres clases de usuarios que manipulan un base de datos:

- **Programador de aplicaciones.** Encargado de escribir programas de aplicación que utilicen las bases de datos. Estos programas se escriben en algún lenguaje de programación de alto nivel, y están diseñados para apoyar a un usuario final.
- **Usuario final.** Accede a la base de datos desde un terminal empleando un lenguaje de consulta (DML) o a través de un programa de aplicación.
- **Administrador de la base de datos.** Es la persona (o grupo de personas) encargada del control general del sistema de base de datos.

### 4.2. El administrador

El administrador de la Base de Datos (ABD) (también conocido por las siglas en inglés DBA, Data Base Administrator), **es el responsable del diseño, control y administración de la Base de Datos.** Se trata en realidad de una función (por lo que sería más correcto decir *administración de la Base de Datos*) que puede ser desempeñada por una persona o por un grupo de personas dependiendo de la envergadura del proyecto.

La función de administración ha sido reconocida por los diferentes grupos de estandarización. Por ejemplo, **Codasyl** creó en su seno un grupo de trabajo especial para tratar los temas relacionados con esta función, al que llamó DBAWG (Data Base Administrator Working Group), que se ocupó, entre otros importantes temas, de especificar los lenguajes de descripción de los datos, de hacer un estudio sobre análisis y evaluación de los SGBD, etc.

También el grupo ANSI/X3/SPARC se refiere a esta función, proponiendo tres niveles, tres **tipos** distintos de administradores:

**Administrador de la empresa:** Se encarga del diseño conceptual y lógico de la Base de Datos.

**Administrador de la Base de Datos:** Es el responsable de las funciones de diseño físico, mantenimiento, seguridad, etc., y de la labor de ajuste o afinamiento (tuning) a efectos de eficiencia de la Base de Datos.

**Administrador de aplicaciones:** Tiene a su cargo la creación de las vistas o esquemas externos que necesitan los programadores de aplicaciones para escribir los programas de consulta y actualización de la Base de Datos.

El Administrador de la Base de Datos va a manejar todas las solicitudes de acceso a la Base de Datos formuladas por los usuarios. Una función general del Administrador es, por tanto, proteger a los usuarios de la Base de Datos contra detalles del hardware. Ofrece una vista que está por encima del nivel de hardware y apoya las operaciones del usuario que se expresen en términos de esa vista de nivel superior.

### 4.3. Funciones del administrador de la base de datos

Las siguientes funciones están expresadas sin diferenciar si deben separarse o no en grupos o personas distintas, ya que ello depende de la estructura del sistema de información (características de uso, distribución, etc.) y del marco organizativo (mayor o menor descentralización) de cada institución. Será responsabilidad del Administrador de la Base de Datos:

- **La estructura de la Base de Datos**, en el sentido de determinar qué información va a ser necesario almacenar en la misma, después de haber analizado los requisitos de los distintos usuarios (en algunos casos esta función debe considerarse como de Administración de datos, es decir, del contenido, y se separa del resto de las funciones de administración más relacionadas con aspectos informáticos).
- **La descripción conceptual y lógica de la Base de Datos**. Una vez especificados los requisitos de la información, es preciso realizar el diseño conceptual de la Base de Datos, para luego después adecuar la estructura conceptual a un SGBD específico y, por tanto, también a un modelo convencional de datos concreto.
- **La descripción física de la Base de Datos**, para encontrar una estructura interna que soporte el esquema lógico y los objetivos de diseño con la máxima eficiencia de los recursos de máquina. Se trata de una labor que se extiende a lo largo de la vida de la Base de Datos, ya que este ajuste nunca puede darse por finalizado, y en tanto exista la Base de Datos, el Administrador de la Base de Datos tendrá que variar

parámetros, reorganizar los datos, modificar estructuras de almacenamiento, realizar nuevas distribuciones de los ficheros en los soportes, etc.

- **Las especificaciones y vistas (o subesquemas) para los programas.** Del estudio de los requisitos de los usuarios se obtendrá un conjunto de necesidades en cuanto a procesos a realizar sobre los datos. Con esta información será preciso definir las vistas externas y la normativa necesarias para los programas de aplicación. Aunque la función de programación de las aplicaciones no es en absoluto de la competencia del administrador, si lo es, sin embargo, dictar las normas y especificaciones que tienen que cumplir las aplicaciones en su acceso a la Base de Datos.
- **Los estándares,** por los que se va a regir la organización en cuanto a documentación de la Base de Datos, metodologías de diseño de la misma, etc.
- **La estrategia de transición** del sistema existente (informatizado o no) al nuevo sistema de información soportado en una Base de Datos. El Administrador de la Base de Datos deberá decidir sobre la posible puesta en marcha en paralelo del nuevo sistema con el antiguo, las fases de implantación del mismo, los controles necesarios, la construcción de prototipos, etc. Todas estas decisiones habrán de tomarse en función de los objetivos marcados y de forma que se cause el mínimo trastorno a los usuarios.
- **Los procedimientos de explotación y uso;** es decir, establecer la normativa necesaria para la utilización de la Base de Datos, el modo de solicitar el acceso a la misma, su actualización, etc.
- **Los aspectos relativos a la seguridad, integridad y confidencialidad,** incluidos los procedimientos de control y las auditorías.
- **El control y la interacción entre la red y la Base de Datos,** es el caso de Bases de Datos accedidas a través de redes informáticas o de Bases de Datos distribuidas.

En definitiva, el administrador interviene en todas las etapas del ciclo de vida de una Base de Datos, teniendo en todas ellas un papel fundamental, que a veces no se reconoce con toda su amplitud.

#### 4.4. Relaciones con otros usuarios

Para cumplir con las funciones anteriores el Administrador de la Base de Datos tendrá que interactuar con todo el personal de la organización.

Es responsabilidad del Administrador vincularse con los usuarios, garantizando que los datos que requieran estén disponibles escribiendo los esquemas externos necesarios.

Esta figura se puede resumir en las siguientes **interfaces** con:

- **Usuarios en general:** Habrá que especificar los requisitos que debe cumplir la Base de Datos, estudiar las necesidades y la mejor manera de satisfacerlas. Además, habrá que conseguir la conformidad respecto a los estándares y a la documentación.
- **Directivos:** Fijarán los objetivos de la Base de Datos para que ésta responda a los objetivos generales de la empresa y se integre en los planes estratégico, táctico y operativo de la misma, proporcionando la información necesaria para la elaboración de dichos planes y para su puesta en marcha. Además, el diseño y la explotación de la Base de Datos requerirán muchos recursos que deberá asignar la dirección de la empresa. Como vemos es muy importante involucrar a los directivos en el

desarrollo del Sistema de Información y su apoyo es una condición indispensable para conseguir el éxito del proyecto.

- **Analistas y programadores:** El Administrador de la Base de Datos les deberá proporcionar información acerca de la estructura de los datos, especificaciones para las aplicaciones que vayan a desarrollarse, etc. Por su parte, estos usuarios deberán informar al administrador sobre las modificaciones o datos nuevos que precisan a fin de que éste estudie las medidas a adoptar.
- **Operadores:** Con ellos, el Administrador de la Base de Datos deberá analizar los procedimientos de recuperación a aplicar en la explotación de la Base de Datos ante fallos del sistema, normas para copias de seguridad, reinicio, etc.
- **Suministradores:** el Administrador de la Base de Datos tendrá que estar en estrecho contacto con ellos para corregir problemas que puedan producirse en el sistema de Base de Datos, además de estar al corriente de nuevas herramientas, interfaces, programas, etc. que estén desarrollándose y que sean de interés en el entorno de la Base de Datos.
- **Personal de formación:** El Administrador de la Base de Datos deberá estar en estrecho contacto con este personal a fin de preparar los planes adecuados de formación sobre Bases de Datos a todos los niveles de la empresa, y completar los conocimientos de los técnicos y de los usuarios, así como difundir la filosofía de las Bases de Datos.

#### 4.5. Administración de datos y administración de bases de datos

Hemos visto que, en algunos organismos, sobre todo los que disponen de Sistemas de Información muy evolucionados, distinguen entre la función de administración de datos y administración de la Base de Datos. Las competencias asignadas a cada una de estas dos figuras son muy distintas:

##### 4.5.1. Administrador de datos

Es el responsable de establecer la arquitectura de datos del organismo en el ámbito conceptual, y suele desempeñar funciones más gerenciales. Este administrador es el encargado del contenido de la Base de Datos y de implantar la filosofía de gestión de recursos de información, (IRM, Information Resource Management) en la empresa. Cuando existe esta función, su responsable no suele encuadrarse en el departamento de informática, dependiendo muchas veces de la dirección general, a fin de poder cumplir mejor las funciones de coordinación que tiene a su cargo.

##### 4.5.2. Administrador de la Base de Datos

Ocupa un puesto más técnico y es el encargado del diseño lógico y físico de la Base de datos, el ajuste y la optimización de la misma.

#### 4.6. Herramientas del administrador de la base de datos

Las herramientas que utiliza el Administrador de la Base de Datos para cumplir sus funciones son:

- **Lenguajes de definición de datos (LDD),** mediante los cuales especificará todos los objetos, atributos e interrelaciones que se almacenarán en la Base de Datos, su

organización física en los soportes de ordenador, y las vistas de usuario. Mediante estos lenguajes también se definirán las restricciones de integridad y de confidencialidad.

- **Utilidades del SGBD** para copias de seguridad, menús para el arranque de la Base de Datos ante caídas, asignación de usuarios, etc. Otro tipo de herramienta muy importante es la de carga y descarga de datos de la Base de Datos a ficheros externos, y viceversa.
- **Estándares** a seguir en la documentación del sistema, en la manipulación de datos, etc.
- **Simuladores y monitores**, que permitirán al administrador estudiar distintas alternativas de instrumentación física con el fin de optimizar la Base de Datos. A este respecto cabe destacar las estadísticas que debe proporcionarle el sistema para que pueda ir realizando el afinamiento de la Base de Datos.
- **Herramientas de ayuda y diseño**, que se están desarrollando extraordinariamente en la actualidad, son las denominadas CASE (Computer Aided Software Engineering), que permiten automatizar en parte y dar soporte en el laborioso proceso de diseño de la Base de Datos, desde el análisis de los requisitos hasta la instrumentación, Mc Clure (1989). Algunas de estas herramientas proporcionan también ayudas para la normalización, el dimensionamiento de la Base de Datos, etc.
- **Diccionario de datos**, que es, según Perkinson (1984), *el arma más potente del arsenal del Administrador de la Base de Datos*, y que ha experimentado una gran evolución a lo largo de los últimos años, hasta culminar en el concepto actual de **diccionario de recursos de información**.

## 5. Protección de datos: seguridad, integridad y confidencialidad

El aspecto global de protección de datos está muy vinculado al propio concepto de Bases de Datos: *Un conjunto de datos integrados, adecuado a varios usuarios y a diferentes usos*. Es el uso concurrente de los datos el que, en muchos casos, conlleva problemas de seguridad, integridad y confidencialidad que el administrador debe paliar en la medida de lo posible.

La **protección de datos** deberá realizarse contra **fallos físicos** (de memoria secundaria, de memoria principal, etc.), **fallos lógicos** (de programación, del Sistema Operativo, del SGBD, etc.) y **fallos humanos**, sean éstos intencionados o no.

Estos fallos alteran indebidamente los datos, los corrompen, con lo que la Base de Datos ya no puede servir a los fines para los que fue creada. Es, por tanto, obligación del administrador conocer y remediar dichos fallos.

El SGBD facilita normalmente mecanismos para *prevenir* los fallos (subsistemas de control), para *detectarlos* una vez que se han producido (subsistemas de detección) y para *corregirlos* después de haber sido detectados (subsistemas de recuperación).

El administrador deberá utilizar los instrumentos que le brinde el SGBD para proteger la Base de Datos, sin embargo, no todos los SGBD proporcionan mecanismos que aseguren el mismo tipo y nivel de protección.

El objetivo del concepto de seguridad (que suele conocerse, a veces por recuperación: recovery) es el de proteger la Base de Datos contra fallos lógicos o físicos que destruyan los datos en todo o en parte.

El objetivo del concepto de **Sistema de Integridad** es proteger a la Base de Datos contra operaciones que introduzcan inconsistencias en los datos, por eso hablamos de integridad en el sentido de *corrección, validez o precisión de los datos de la base*. El subsistema de integridad de un SGBD debe detectar y corregir, en la medida de lo posible, las operaciones incorrectas.

El objetivo del concepto de **confidencialidad** es *proteger la Base de Datos contra accesos no autorizados*; se suele llamar también “privacidad” (privacy). La protección deberá incluir no sólo a los datos y sus interrelaciones, sino también a los esquemas de la Base de Datos, programas que acceden a la misma, etc.

Una técnica de protección de confidencialidad que puede utilizarse en los SGBD es la **criptografía**, que permite transformar el contenido de la base, haciéndola ininteligible a cualquier usuario que acceda a la misma sin la correspondiente clave de descifrado.

## 6. Conclusión

Desde que en los años 70 del siglo pasado se popularizarasen las bases de datos como soluciones comunes a la manipulación de grandes volúmenes de datos por parte de aplicaciones diversas, se han desarrollado varias arquitecturas, siendo actualmente dos, las relacionalles (SQL) y las NoSQL (No relacionalles) las que son principalmente utilizadas hoy en día.

El administrador de la base de datos es el encargado de implementar el diseño de la base de datos en un DBMS concreto creando los índices necesarios, las vistas, los formularios de introducción de datos y todas las posibles mejoras que se deban adoptar. También debe mejorar el rendimiento ajustando ciertos parámetros, realizar un estudio de la carga del sistema, especialmente de las consultas y los procesos de actualización que involucren múltiples registros.

El administrador debe velar por el buen funcionamiento de la base de datos, asegurando la fiabilidad y confiabilidad de la misma, utilizando para ello todas las herramientas que provee el DBMS.

### 6.1. Relación del tema con el sistema educativo actual

Este tema es aplicado en el aula en los módulos profesionales siguientes, con las atribuciones docentes indicadas (PES/SAI):

**Formación profesional básica**

- Operaciones auxiliares para la configuración y la explotación(TPB en Informática de Oficina/ TPB en informática y Comunicaciones) (PES/SAI)
- Ofimática y archivo de documentos (TPB en Informática de Oficina) (PES/SAI)

### Grado Medio

- Aplicaciones ofimáticas (GM de SMR) (PES/SAI)

### Grado Superior

- Gestión de bases de datos (ASIR) (PES)
- Bases de Datos (DAW/DAM) (PES)

### Bachillerato:

- 4º ESO – Tecnología de la Información y la comunicación (PES)
- Bachillerato – Tecnologías de la Información y la Comunicación (PES)

## 7. Bibliografía

- C.J. Date: **Introducción a los sistemas de bases de datos** Pearson, 2001.
- Elmasri, R.A. y Navathe S.B: "**Fundamentos de Sistemas de Bases de Datos**". Addison-Wesley, 3<sup>a</sup> Edic, 2002.
- Olga Pons, J M Medina, M.A. Vila. **Introducción a los Sistemas de Bases de Datos** Edt Paraninfo (2005)
- Korth, H.F. y Silberschatz: "**Fundamentos de Bases de Datos**". McGraw -Hill, 4<sup>a</sup> Edic., 2002.
- Garcia-Molina, H.; Ullman, J.D.; Widom, J. **Database systems: the complete book** - Pearson Education Limited, 2013.
- Abraham Silberschatz, Henry F. Korth, y S. Sudarshan, **Fundamentos de bases de datos** Edt. Mc Graw-Hill (2014)

