

2017

FUNDAMENTOS DE BASES DE DATOS

Guía Práctica Conceptos básicos de Bases de
Datos

ALEXANDRA RUIZ GAONA
Universidad del Quindío
Programa de Ingeniería de Sistemas y
Computación



Tabla de Contenido

1. Introducción	4
2. Logros Esperados de la Actividad Académica	4
3. Conceptos Generales	4
4. Características del enfoque de bases de datos.....	6
4.1 Naturaleza auto descriptiva de los sistemas de bases de datos	6
4.2 Separación entre los programas y los datos, y abstracción de los datos	7
4.3 Manejo de múltiples vistas de los datos.....	7
4.4 Comportamiento de datos y procesamiento de transacciones multiusuario	7
5. Actores de un sistema de bases de datos	8
5.1 Administrador de la base de datos (DBA).....	8
5.2 Diseñador de la base de datos	8
5.3 Futuros usuarios o usuarios finales	9
5.4 Trabajadores detrás del telón (tras bambalinas).....	9
6. Características deseables de un SGBD	10
6.1 Control de redundancia	10
6.2 Restricción de los accesos no autorizados.....	10
6.3 Inferencias en la base de datos mediante reglas de deducción.....	10
6.4 Suministro de múltiples interfaces con usuarios.....	11
6.5 Representación de vínculos complejos entre los datos	11
6.6 Cumplimiento de las restricciones de integridad	11
6.7 Respaldo y recuperación.....	11
7. Modelos de Bases de Datos	12
7.1 Modelo Jerárquico	12
7.2 Modelo en Red.....	12
7.3 Modelo Relacional	13
7.4 Modelo Orientado a Objetos	13
8. Arquitectura ANSI/SPARQ.....	14
8.1 Independencia con respecto a los datos	16

8.2	Lenguajes Anfitriones	17
9.	Referencias.....	20
10.	Taller	21

Índice de Figuras

Figura 1. Usuarios Programadores	6
Figura 2. Ejemplo Modelo Jerárquico	12
Figura 3. Modelo en Red	13
Figura 4. Modelo Relacional	13
Figura 5. Modelo Orientado a Objetos	14
Figura 6. Arquitectura ANSI-SPARC para base de datos.....	14
Figura 7. Ilustración de los tres niveles de la arquitectura ANSI.....	15
Figura 8. Correspondencia entre los niveles de la arquitectura	16
Figura 9. Operación interna de una base de datos	18

1. Introducción

Hoy en día, las empresas y en sí la sociedad ha pasado de ser una sociedad capitalista a una sociedad del conocimiento, donde la información es el recurso más valioso dentro de la organización. Esto supone que la información debe gestionarse, es decir, la información se recoge, depura, almacena, analiza, transforma y transporta. Para lograr este objetivo, uno de los mecanismos que hace esto posible son las bases de datos. Las Bases de Datos tienen una gran relevancia a nivel empresarial, y se consideran una de las mayores aportaciones que ha dado la informática a las empresas. En la actualidad, cualquier organización que se precie, debe contar con bases de datos. Si una base de datos se gestiona adecuadamente, la organización obtendrá diferentes ventajas: aumentará su eficacia, mejorará la seguridad de los datos almacenados, se genera una mayor productividad, entre otras. De esta manera, las empresas consiguen que la información y el conocimiento sean los mayores activos de la compañía.

De acuerdo a este panorama, el aprendizaje de las bases de datos es imprescindible en la formación de los ingenieros en las áreas de la informática. La *Guía Conceptos Básicos de Bases de Datos* pretende ser el primer peldaño en el aprendizaje relacionado a las bases de datos. En esta guía el estudiante podrá aprender sobre los conceptos básicos de BD, las personas implicadas en su desarrollo, sistemas gestores de bases de datos y arquitecturas.

2. Logros Esperados de la Actividad Académica

Al termina la guía, el estudiante debe estar en capacidad de:

- Entender y conocer el concepto de una base de datos y sus principales características básicas.
- Entender y conocer los niveles existentes en una base de datos.

3. Conceptos Generales

Una **base de datos** es un conjunto de datos relacionados entre sí. Por **datos** entendemos hechos conocidos que pueden registrarse y que tienen un significado implícito.

Ejemplo. Nombres – Números Telefónicos – Direcciones

- Una base de datos representa algún aspecto del mundo real, en ocasiones llamado minimundo o universo de discurso. Las modificaciones del minimundo se reflejan en la base de datos.

- Una base de datos es un conjunto de datos lógicamente coherente, con cierto significado inherente. Una colección aleatoria de datos no puede considerarse propiamente una base de datos.
- Toda base de datos se diseña, construye y puebla con datos para un propósito específico. Está dirigida a un grupo de usuarios y tiene ciertas aplicaciones preconcebidas que interesan a dichos usuarios.

En otras palabras, una base de datos tiene una **fuente de la cual se derivan los datos**, cierto grado de **interacción con los acontecimientos del mundo real y un público que esta activamente interesado en el contenido de la base de datos**.

Las bases de datos pueden ser de cualquier tamaño y tener diversos grados de complejidad.

Por ejemplo, la lista de nombres y direcciones antes mencionada puede contener apenas unas cuantas centenas de registros, cada uno de ellos con una estructura muy simple. Por otro lado, el catálogo de una biblioteca grande puede contener medio millón de tarjetas clasificadas por categorías distintas -apellido del primer autor, tema, título, etc. Y ordenadas alfabéticamente en cada categoría

La generación y el mantenimiento de las bases de datos pueden ser **manuales o mecánicos**.

El catalogo en tarjetas de una biblioteca es un ejemplo de base de datos que se puede crear y mantener manualmente. Las bases de datos computarizadas se pueden crear y mantener con un grupo de programas de aplicación escritos específicamente para esa tarea o bien mediante un **sistema de gestión de bases de datos**.

Un sistema de gestión de bases de datos (SGBD; en inglés, database management system: DBMS) es un conjunto de programas que permite a los usuarios crear y mantener una base de datos. Por tanto, el SGBD es un sistema de software de propósito general que facilita el proceso de **definir, construir y manipular** bases de datos para diversas aplicaciones.

- **Para definir** una base de datos hay que especificar los tipos de datos, las estructuras y las restricciones de los datos que se almacenaran en ella.
- **Construir** una base de datos es el proceso de **guardar los datos** mismos en algún medio de almacenamiento controlado por el SGBD.
- **En la manipulación** de una base de datos intervienen funciones como **consultar** la base de datos para obtener datos específicos, **actualizar** la base de datos para reflejar cambios en el minimundo y **generar** informes a partir de los datos.

No hace falta un software de SGBD de propósito general para implementar una base de datos computarizada.

Al conjunto formado por la base de datos, el software, el hardware y el personal apropiado lo llamaremos **sistema de base de datos**. La Figura 1 se ilustra estas ideas.

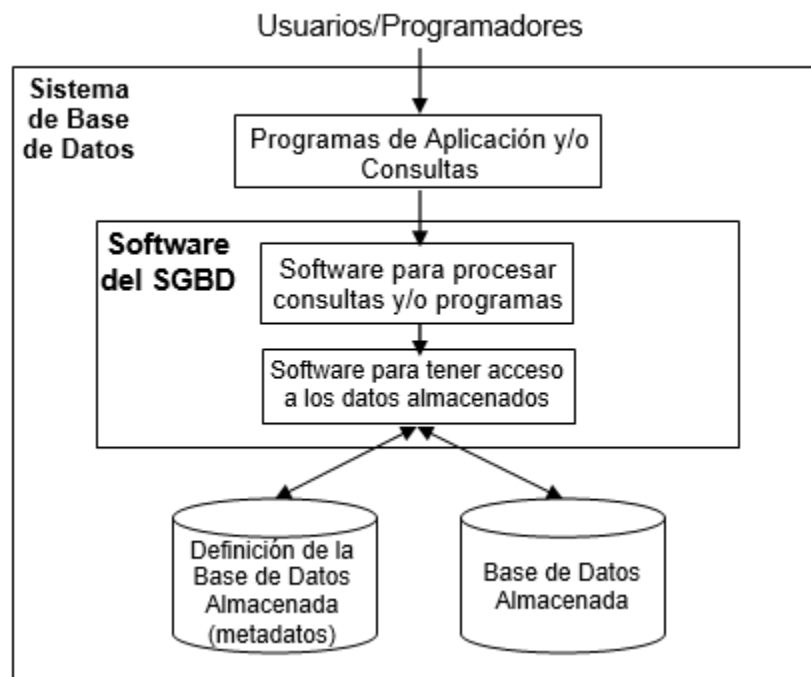


Figura 1. Usuarios Programadores

4. Características del enfoque de bases de datos

Hay varias características que distinguen el enfoque de bases de datos del enfoque tradicional de programación con archivos. (Comprendiendo las TICs, 2016)

4.1 Naturaleza auto descriptiva de los sistemas de bases de datos

Mientras que el software para el procesamiento de archivos SGBD puede tener acceso a bases de datos específicas, el software del SGBD puede acceder a diversas bases de datos al extraer del catálogo las definiciones correspondientes y darles un uso.

4.2 Separación entre los programas y los datos, y abstracción de los datos

En el procesamiento de archivos tradicional, la estructura de los archivos de datos viene integrada en los programas de acceso, así que cualquier modificación de la estructura de un archivo puede requerir la modificación de todos los programas que tienen acceso a dicho archivo. En cambio, los programas de acceso del SGBD se escriben de modo que sean independientes de cualesquiera archivos específicos, ya que las estructuras están almacenadas en los catálogos y no en los programas de acceso. Esto podría llamarse **independencia con respecto a los programas y datos**. La característica que hace esto posible se denomina **Abstracción de datos**.

Un SGBD ofrece a los usuarios una representación conceptual de los datos que no incluye muchos de los detalles de cómo se almacenan. En términos informales, un **modelo de datos** es un tipo de abstracción de los datos con que se obtiene esta representación conceptual. En el modelo de datos intervienen conceptos lógicos, como serían los objetos, sus propiedades y sus interrelaciones, que la mayoría de los usuarios pueden entender más fácilmente que los conceptos de almacenamiento en el computador. Por tanto, el **modelo de datos oculta los detalles del almacenamiento** que no interesan a la mayoría de los usuarios de la base de datos.

4.3 Manejo de múltiples vistas de los datos

Una base de datos suele tener muchos usuarios, cada uno de los cuales puede requerir una perspectiva o vista diferente de la mencionada base de datos. Una **vista** puede ser un **subconjunto de la base de datos** o contener datos virtuales que se deriven de los archivos de la base de datos, pero que no estén almacenados explícitamente. Es posible que algunos usuarios no necesiten saber si los datos a los que hacen referencia están almacenados o son derivados.

4.4 Comportamiento de datos y procesamiento de transacciones multiusuario

Todo SGBD multiusuario, como su nombre lo indica, debe permitir a varios usuarios tener **acceso simultáneo** a la base de datos. Esto es indispensable para que los datos de múltiples aplicaciones se **integren y mantengan** en una sola base de datos. El SGBD debe incluir software de control de concurrencia para asegurar que cuando varios usuarios intenten actualizar los mismos datos lo hagan de manera controlada para que el resultado de las actualizaciones sea correcto. En general, se dice que estas son aplicaciones de

procesamiento de transacciones, y una función fundamental del software del SGBD multiusuario es asegurar que las transacciones concurrentes se realicen de manera correcta sin interferencias.

5. Actores de un sistema de bases de datos

A continuación se describen los actores que participan en un sistemas de bases de datos. (Sanchez, 2015)

5.1 Administrador de la base de datos (DBA)

Las funciones de administración de la base de datos se entienden con el control general del sistema a nivel técnico. Un administrador de la base de datos (DBA) es la persona encargada de dar el soporte técnico a la base de datos con el fin de que esta permanezca consistente, segura y disponible a los grupos de usuarios autorizados. El administrador de la base de datos debe cumplir con las siguientes funciones: (Poveda, 2016)

- Definir las estructuras de almacenamiento y los métodos de acceso a la información.
- Definir o participar en la creación del esquema conceptual (DDL)
- Definir el esquema interno (DDL)
- Definir la comunicación con los usuarios (garantizar la disponibilidad de los datos)
- Definir verificaciones de seguridad e integridad.
- Definir procedimientos de respaldo y recuperación.
- Evaluar el desempeño y responder a cambios en los requerimientos.

5.2 Diseñador de la base de datos

Los diseñadores de bases de datos se encargan de identificar los datos que se almacenarán en la base de datos y de elegir las estructuras apropiadas para representar y almacenar dichos datos (Navathe & Elmasri, 2000). Por lo general, estas tareas se realizan antes de implementar la base de datos. Los diseñadores son los responsables de comunicarse con todos los usuarios futuros de la base de datos, así podrán comprender sus necesidades, y presentar un diseño que cumpla los requerimientos de los usuarios. Usualmente, los diseñadores hacen parte del equipo DBA y pueden asumir otras tareas una vez se haya culminado el diseño de la base de datos.

Casi siempre, los diseñadores interactúan con cada uno de los grupos de usuarios potenciales y desarrollan una vista de la base de datos que satisfaga los requerimientos de datos y de procesamiento para ese grupo. Después, se analizan las vistas y se integran con las de otros grupos de usuarios. El diseño final debe ser capaz de satisfacer las necesidades de todos estos grupos (Navathe & Elmasri, 2000)

5.3 Futuros usuarios o usuarios finales

Son las personas que acceden la base de datos para consultar, actualizar los datos o generar reportes; la existencia de la base de datos es esencialmente para que estos la usen. Existen diferentes tipos de usuarios finales (Navathe & Elmasri, 2000) **1) Los esporádicos** tienen acceso ocasionalmente a la base de datos, y puede suceder que cada vez que la acceden requieran información diferente. Para especificar sus solicitudes utilizan un lenguaje avanzado de consulta de base de datos, y suelen ser gerentes u otras personas que examinan de modo superficial y ocasional la base de datos. **2) Los simples** constituyen el porcentaje más alto de la totalidad de los usuarios finales. Su principal función se centra en las consultas y actualizaciones constantes de la base de datos, utilizando las llamadas **transacciones programadas**. **3) En los usuarios avanzados** podemos encontrar a los ingenieros, científicos, analistas de negocios y otros, quienes conocen completamente los recursos del SGBD para satisfacer sus complejos requerimientos. **4) Los usuarios autónomos** emplean bases de datos personalizadas gracias a los paquetes de programas comerciales que cuentan con interfaces de fácil uso, basadas en menús o en gráficos.

5.4 Trabajadores detrás del telón (tras bambalinas)

Los trabajadores detrás del telón, están conformado por aquellas personas que se encargan del software y el entorno del sistemas SGBD. Entre estos podemos encontrar a los **diseñadores e implementadores** del SGBD, que se encargan de diseñar e implementar los módulos e interfaces del SGBD en forma de paquetes de software; **los creadores de herramientas**, los cuales desarrollan e implementan paquetes para diseñar bases de datos, vigilar el rendimiento, proporcionar interfaces de lenguaje natural o de gráficos, elaborar prototipos, realizar simulaciones y generar datos de prueba. Finalmente están los **operadores y personal de mantenimiento**, que tienen a su cargo el funcionamiento y mantenimiento real del entorno de hardware y software del sistema de base de datos (Navathe & Elmasri, 2000).

6. Características deseables de un SGBD

El SGBD debe permitir que el diseñador defina datos redundantes, pero entonces tendría que ser el mismo SGBD el que hiciese automáticamente la actualización de los datos en todos los lugares donde estuviesen repetidos. (Camps Paré, 2016)

Las características deseables de un SGBD son:

6.1 Control de redundancia

En primer lugar, es necesario realizar una misma actualización lógica -como introducir los datos de un nuevo estudiante- varias veces: una vez en cada archivo en el que se registren datos de los estudiantes. Esto implica una duplicación del trabajo. En segundo lugar, se desperdicia espacio de almacenamiento al guardar los mismos datos en varios lugares, y este problema puede ser grave si las bases de datos son grandes. En tercer lugar es posible que los archivos que representan los mismos datos se tornen inconsistentes, quizá porque una actualización se haya aplicado a ciertos archivos pero no a otros. Incluso si la actualización -digamos la adición de un nuevo estudiante- se aplica a todos los archivos apropiados, persiste la posibilidad de que los datos relacionados con el estudiante sean inconsistentes porque cada grupo de usuarios aplica las actualizaciones de manera independiente.

6.2 Restricción de los accesos no autorizados

Cuando muchos usuarios comparten una misma base de datos, es probable que no todos tengan la autorización para tener acceso a toda la información que contiene. El SGBD debe contar con un subsistema de seguridad y autorización que permita al DBA crear cuentas y especificar restricciones para ellas. El SGBD deberá entonces obligar automáticamente al cumplimiento de dichas restricciones. Cabe señalar que el mismo tipo de controles se puede aplicar al software del SGBD.

6.3 Inferencias en la base de datos mediante reglas de deducción

Otra aplicación reciente de los sistemas de base de datos consiste en ofrecer recursos para definir reglas de deducción que permitan *deducir* o *inferir* información nueva a partir

de los datos almacenados. A estos sistemas se les conoce como bases de datos **deductivas**.

6.4 Suministro de múltiples interfaces con usuarios

En vista de que muchos tipos de usuarios con diversos niveles de conocimientos técnicos utilizan las bases de datos, el SGBD debe ofrecer diferentes interfaces. Entre estas podemos mencionar **los lenguajes de consulta** para usuarios esporádicos, **las interfaces de lenguaje de programación** para programadores de aplicaciones, las formas y códigos de órdenes para los **usuarios paramétricos** y las interfaces controladas por menús y en lenguaje natural para los usuarios **autónomos**.

6.5 Representación de vínculos complejos entre los datos

Una base de datos puede contener **numerosos conjuntos de datos** que estén relacionados entre sí de muchas maneras. Es preciso que el SGBD pueda representar diversos vínculos complejos de los datos y también obtener y actualizar con **rapidez y eficiencia** datos que estén mutuamente relacionados.

6.6 Cumplimiento de las restricciones de integridad

La mayor parte de las aplicaciones de base de datos tienen ciertas restricciones de integridad que deben cumplir los datos. El SGBD debe **ofrecer recursos** para definir tales restricciones y hacer que se cumplan. La forma más simple de restringir la integridad consiste en especificar **un tipo de datos** para cada elemento de información. Es responsabilidad de los diseñadores de la base de datos identificar las restricciones de integridad durante el diseño. Algunas restricciones se pueden especificar en el SGBD, el cual hará automáticamente que se cumplan; otras pueden requerir verificación mediante programas de actualización o en el momento en que se introducen los datos.

6.7 Respaldo y recuperación

Todo SGBD debe contar con recursos para recuperarse de fallos de hardware o de software. Para ella está el subsistema de respaldo y recuperación del SGBD. Por ejemplo, si el sistema falla mientras se está ejecutando un complejo programa de actualización, el

subsistema de recuperación se encargara de asegurarse de que la base de datos se restaure al estado en el que estaba antes de que comenzara la ejecución del programa. Como alternativa, el subsistema de recuperación puede asegurarse de que el programa reanude su ejecución en el punto en que fue interrumpido, de modo que su efecto completo se registre en la base de datos.

7. Modelos de Bases de Datos

Los modelos de datos que han aparecido en la evolución de las bases de datos son fundamentalmente métodos conceptuales para estructurar los datos (Navathe & Elmasri, 2000). Los principales son:

7.1 Modelo Jerárquico

Fue uno de los primeros sistemas de bases de datos comerciales, cuya estructura estaba compuesta de un conjunto ordenado de árboles. En ella, un registro hijo tiene uno y sólo un padre. Su principal exponente fue IMS (System, 2016) de IBM (Figura 2).

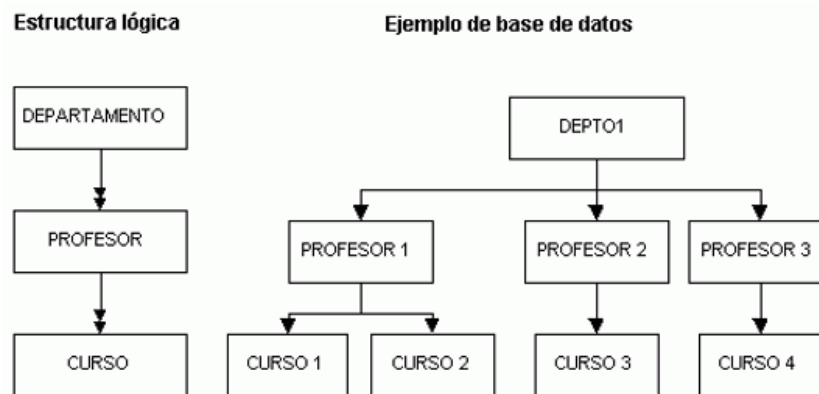


Figura 2. Ejemplo Modelo Jerárquico

7.2 Modelo en Red

Sistema cuya estructura se considera como una forma extendida de la jerárquica, con la diferencia de que un registro hijo puede tener cualquier número de padres. Un sistema de red fue IDMS (Integrated Database Management System) de Computer Associates (Figura 3).



Figura 3. Modelo en Red

7.3 Modelo Relacional

Es básicamente una forma de ver y representar los datos mediante tablas denominadas *relaciones* y una forma de manipular esa representación de dos maneras distintas pero equivalentes, denominadas *álgebra relacional* y *cálculo relacional* (Figura 4).

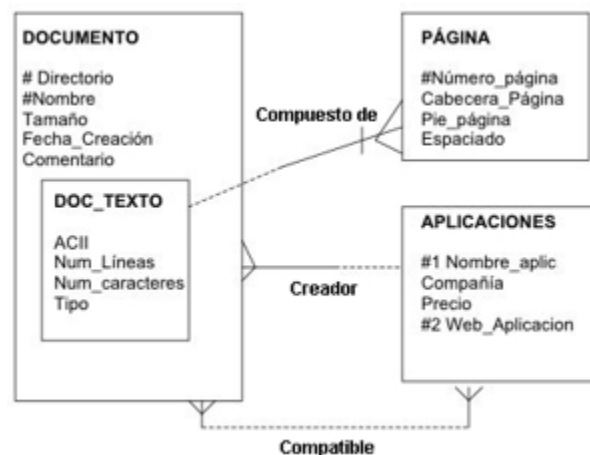


Figura 4. Modelo Relacional

7.4 Modelo Orientado a Objetos

Define una base de datos en términos de objetos, sus propiedades y sus operaciones. Los objetos con la misma estructura y comportamiento pertenecen a una clase, y las clases se organizan en jerarquías o grafos acíclicos. Las operaciones de cada clase se especifican en términos de procedimientos predefinidos llamados métodos. Ya hay ahora en el mercado varios sistemas basados en el paradigma orientado a objetos Figura 5. (Sanchez, 2015)

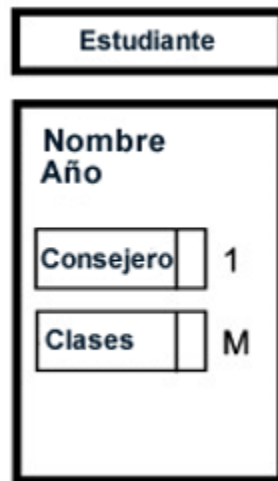


Figura 5. Modelo Orientado a Objetos

8. Arquitectura ANSI/SPARQ

La arquitectura de una base de datos, de acuerdo con la normatividad ANSI/SPARQ, básicamente está constituida por tres niveles los cuales forman las diferentes visiones o enfoques que puede tener una base de datos por los diversos tipos de usuarios con los que ella puede interactuar: programadores, administradores o usuarios finales. El objetivo de la arquitectura de tres esquemas consiste en formar **una separación entre las aplicaciones del usuario y la base de datos física** (Sanchez, 2015). Se hace referencia a tres niveles: **el externo, el interno y el conceptual** (ver Figura 6).

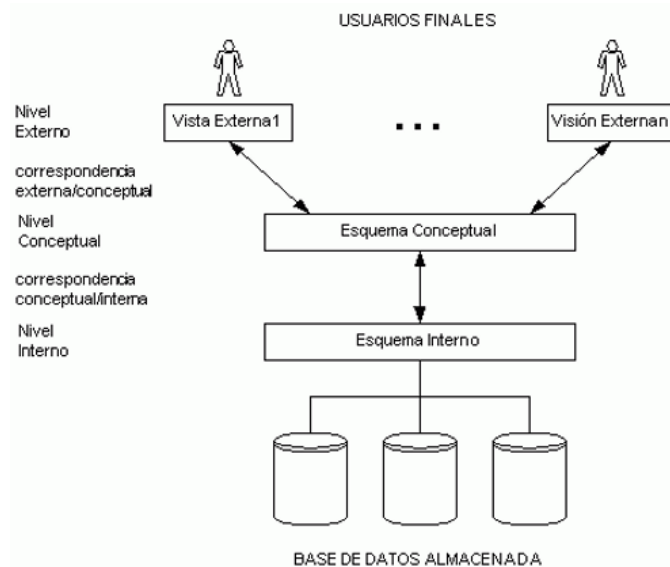


Figura 6. Arquitectura ANSI-SPARC para base de datos

- El **nivel interno** tiene un **esquema interno**, que describe la **estructura física de almacenamiento** de la base de datos. El esquema interno emplea un modelo físico de los datos y describe todos los detalles para su almacenamiento, así como los caminos de acceso para la base de datos. Aquí se define el diseño físico de la base de datos: La definición de las estructuras de almacenamiento, los parámetros y los espacios de almacenamiento y las ubicaciones físicas en los dispositivos de memoria secundaria.
- El **nivel conceptual** tiene un **esquema conceptual**, que describe la estructura de toda la base de datos para una comunidad de usuarios. El esquema conceptual oculta los detalles de las estructuras físicas de almacenamiento y se concentra en describir **entidades, tipos de datos, vínculos, operaciones de los usuarios y restricciones**. Precisamente en este nivel se manejan los conceptos de entidades y relaciones.
- El **nivel externo o de vistas** incluye varios **esquemas externos o vistas de usuario**. Cada esquema externo describe la parte de la base de datos que interesa a un grupo de usuarios determinado, y oculta a ese grupo el resto de la base de datos. Existen **muchas** vistas externas distintas y **una sola** vista conceptual.

En la Figura 7 se ilustran los tres niveles de la arquitectura ANSI.

<u>Externo (PL/I)</u>		<u>Externo (COBOL)</u>	
DCL	1 EMPP; 2 EMP# CHAR (6) 2 SAL FIXED BIN (31);	01 EMPC. 02 NUMEMP PIC X (6) 02 NUMEMP PIC X (4)	
<u>Conceptual</u>			
EMPLEADO	NUMERO_EMPLEADO NUMERO_DEPARTAMENTO SALARIO	CARÁCTER (6) CARÁCTER (4) NUMERICO (5)	
<u>Interno</u>			
EMP_ALMAC	LONGITUD = 18		
PREFIJO	TIPO=BYTE (6), DESPLAZAMIENTO =0		
EMP#	TIPO=BYTE (6), DESPLAZAMIENTO =6, INDICE =TEMP		
DEPTO#	TIPO=BYTE (4), DESPLAZAMIENTO =12		
PAGA	TIPO=PALABRA, DESPLAZAMIENTO=16		

Figura 7. Ilustración de los tres niveles de la arquitectura ANSI

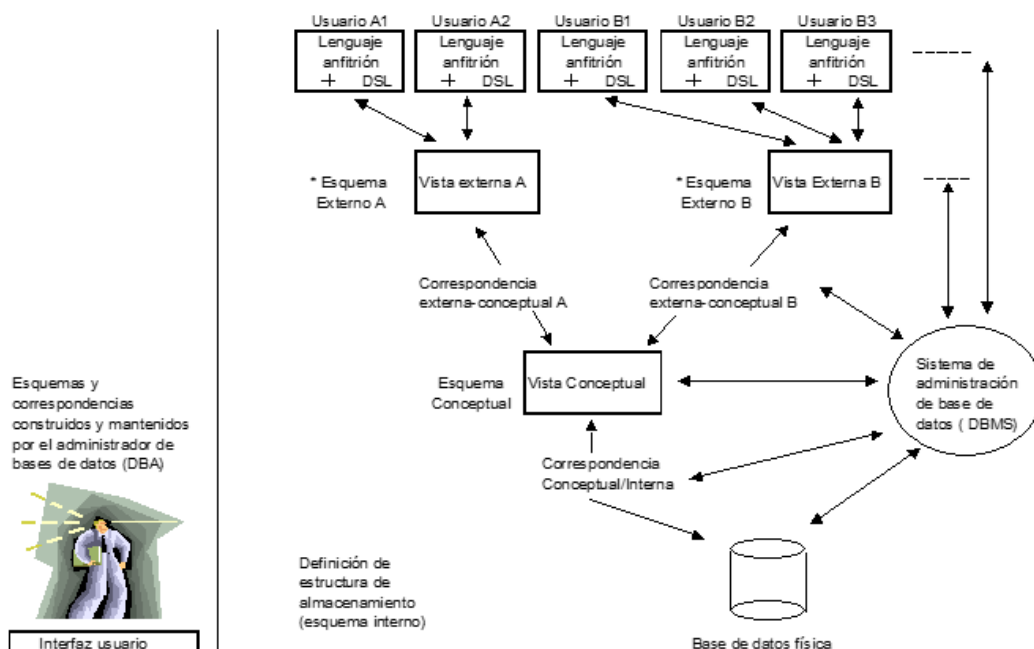


Figura 8. Correspondencia entre los niveles de la arquitectura

Los tres esquemas no son más que descripciones de los datos; los únicos datos que existen realmente están en el **nivel físico**. En un SGBD basado en la arquitectura de tres esquemas, cada grupo de usuarios hace referencia exclusivamente a su propio esquema externo; por tanto, el SGBD debe transformar una solicitud expresada en términos del esquema conceptual, y luego a una solicitud en el esquema interno que se procesará sobre la base de datos almacenada. Si la solicitud es una obtención de datos, será preciso modificar el formato de la información extraída de la base de datos almacenada para que coincida con la vista externa del usuario. El proceso de transformar solicitudes y resultados de un nivel a otro se denomina **correspondencia** o **transformación** (*mapping*).

8.1 Independencia con respecto a los datos

La arquitectura de tres esquemas puede servir para explicar el concepto de **independencia con respecto a los datos**, que podemos definir como la capacidad para modificar el esquema de un nivel del sistema de base de datos sin tener que modificar el esquema del nivel inmediato superior. Podemos definir dos tipos de independencia con respecto a los datos: (Navathe & Elmasri, 2000)

1. La **independencia lógica con respecto a los datos** es la capacidad de modificar el **esquema conceptual** sin tener que alterar los esquemas externos ni los programas de aplicación. Podemos modificar el esquema conceptual para ampliar la base de datos (añadiendo un nuevo tipo de registro o un elemento de información), o para

reducir la base de datos (eliminando un tipo de registro o un elemento de información). En el segundo caso, la modificación no deberá afectar los esquemas externos que solo se refieran a los datos restantes.

2. La **independencia física con respecto a los datos** es la capacidad de modificar el **esquema interno** sin tener que alterar el esquema conceptual (o los externos). Tal vez sea preciso modificar el esquema interno por la necesidad de reorganizar ciertos archivos físicos -por ejemplo, al crear estructuras de datos adicionales- a fin de mejorar el rendimiento de las operaciones de obtención o actualización. Si la base de datos aún contiene los mismos datos, no deberá ser necesario modificar el esquema conceptual.

8.2 Lenguajes Anfitriones

Son lenguajes a disposición de cada usuario, sea usuario final o programador de aplicaciones; por ejemplo **Cobol, C++, PL/1, PL/SQL**, generadores de formularios de diversos productos, lenguajes de cuarta generación o aplicaciones en línea. Los lenguajes anfitriones deben incluir un **sublenguaje de datos o DSL** (Data Sublanguage) para poder acceder a la base de datos.

Sublenguaje de Datos

Lenguaje incorporado en un lenguaje anfitrión. Sirve para **operar los objetos** de la base de datos. Consta de tres partes fundamentales que son:

- Un lenguaje de definición de datos o **DDL**: especificar los **esquemas conceptual e interno** de la base de datos y cualquier correspondencia entre ambos.
- Un lenguaje de manipulación de datos o **DML**: Las operaciones de manipulación más comunes son la **obtención, la inserción, la eliminación y la modificación** de datos.
- Un lenguaje de control o DCL

El sublenguaje de datos más estándar es el SQL.

El Motor de una Base de Datos

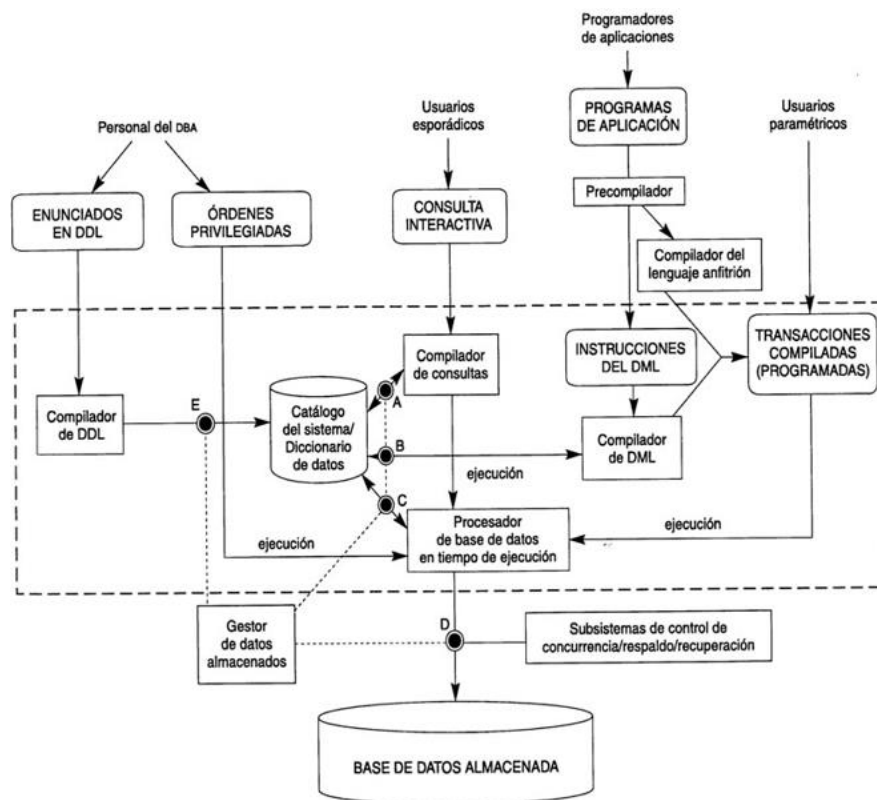


Figura 9. Operación interna de una base de datos

La Figura 9 los componentes de un SGBD representativo. La base de datos y el catálogo del SGBD casi siempre se almacenan en disco. El acceso al disco suele controlarlo primordialmente el sistema operativo (OS: operating system), que programa la entrada/salida del disco. Un módulo gestor de datos almacenados del SGBD, de más alto nivel, controla el acceso a la información del SGBD almacenada en el disco, ya sea que forme parte de la base de datos o del catálogo. Las líneas punteadas y los círculos rotulados A, B, C, D y E en la figura 2.3 ilustran accesos que están bajo el control de este gestor de datos almacenados. Este último puede emplear servicios básicos del S.O para transferir los datos de bajo nivel entre el disco y la memoria principal del computador, pero controla otros aspectos de la transferencia de datos, como el manejo de las áreas de almacenamiento intermedio (buffers) en la memoria principal. Una vez que los datos estén en dicho almacenamiento intermedio, otros módulos del SGBD podrán procesarlos.

El compilador de DDL procesa las definiciones de esquemas, especificadas en el DDL, y almacena las descripciones de los esquemas (metadatos) en el catálogo del SGBD. El catálogo contiene información como los nombres de los archivos y de los elementos de información, los detalles de almacenamiento de cada archivo, la información de

correspondencia entre los esquemas y las restricciones. Los módulos del SGBD que necesiten conocer esta información deberán consultar el catálogo.

El procesador de la base de datos en tiempo de ejecución se encarga de los accesos a ella durante la ejecución; recibe operaciones de obtención o de actualización y las ejecuta sobre la base de datos. El acceso al disco se tiene mediante el gestor de datos almacenados. El compilador de consultas maneja las consultas de alto nivel que se introducen interactivamente. Analiza la sintaxis y el contenido de las consultas y luego genera llamadas al procesador en tiempo de ejecución para atender la solicitud.

El precompilador extrae órdenes en DML de un programa de aplicación escrito en un lenguaje de programación anfitrión. Estas órdenes se envían al compilador de DML para convertirlas en código objeto para el acceso a la base de datos, y el resto del programa se envía al compilador del lenguaje anfitrión. El código objeto de las órdenes en DML y el del resto del programa se enlazan, formando una transacción programada cuyo código ejecutable incluye llamadas al procesador de la base de datos durante el tiempo de ejecución. Con la figura 2.3 no se intenta describir un SGBD específico; más bien, lo utilizamos para ilustrar los módulos representativos de un SGBD. El SGBD interactúa con el sistema operativo cuando se requiere acceso al disco (a la base de datos o al catálogo).

Si muchos usuarios comparten el mismo sistema de cómputo, él os programara las solicitudes de acceso a disco del SGBD junto con otros procesos. El SGBD también se comunica con los compiladores de los lenguajes de programación anfitriones, y puede ofrecer interfaces amables con el usuario como ayuda para cualquiera de los tipos de usuarios mostrados en la Figura 9 cuando especifican sus solicitudes.

9. Referencias

Camps Paré, R. (2016). *Introducción a las Bases de Datos*.

Comprendiendo las TICs. (2016). Obtenido de <http://comprendiendolastics.blogspot.com.co/2011/04/caracteristicas-del-enfoque-de-bases-de.html>

Navathe, S., & Elmasri, R. (2000). *Sistema de Bases de Datos*.

Poveda, J. (2016).

Sanchez, J. (2015). *Sistema de Gestión Base de Datos*. España.

Silberschatz, A., Korth, H. F., & Sudarshan, S. (2002). *FUNDAMENTOS DE BASES DE DATOS Cuarta edición*. Madrid: MCGRAW-HILL .

System, I. M. (2016). *IBM*. Obtenido de <https://www-01.ibm.com/software/data/ims/>

11.Taller

1. Emparejar la Columna A y columna B de acuerdo al verdadero significado de cada termino.

COLUMNA A	COLUMNA B
A) Independencia Lógica	Permite realizar obtención, inserción, eliminación y modificación de datos (____)
B) Nivel Interno	Es la capacidad de modificar el esquema interno sin tener que alterar el esquema conceptual o los externos (____)
C) Independencia Física	Característica indispensable para que los datos de varias aplicaciones se integren y mantengan en una sola base de datos. (____)
D) Modelo de Datos	subconjunto de la base de datos o datos virtuales que se deriven de los archivos de la base de datos(____)
E) Nivel Conceptual	Es la capacidad de modificar el esquema conceptual sin tener que alterar el esquema externo ni los programas de aplicación (____)
F) Multiusuario	se concentra en describir entidades, tipos de datos, vínculos, operaciones de los usuarios y restricciones (____)
G) Arquitectura ANSI/SPARK	Su objetivo consiste en formar una separación entre las aplicaciones del usuario y la base de datos física (____)
H) SGBD	Conjunto formado por la base de datos, el software, el hardware y el personal apropiado. (____)
I) Vista	Describe la estructura física de almacenamiento de la base de datos (____)
J) Sistema de Bases de Datos	Es un tipo de abstracción de los datos que oculta los detalles de almacenamiento (____)

2. Responda Falso (F) o verdadero (V) a las siguientes afirmaciones. Justifique las falsas.
 - a) Dentro de las características deseables de un SGBD está el control de redundancia. (____)
 - b) El diseñador de la base de datos es el encargado del control general a nivel técnico (____)

- c) Los usuarios esporádicos son gerentes u otras personas que examinan de modo superficial y ocasional la base de datos (____)
 - d) Dentro de las características deseables de un SGBD está el control de redundancia (____)
 - e) Las reglas de integridad buscan que los datos almacenados en la base no puedan representar hechos ilógicos (____)
 - f) La característica asociada a la naturaleza autodescriptiva de los sistemas de bases de datos se debe a la existencia del catálogo de la base de datos o a los metadatos. (____)
 - g) Una base de datos es un conjunto de datos relacionados entre sí. (____)
 - h) Las reglas de integridad buscan evitar que se registren sucesos erróneos que puedan ocurrir en el mundo real. (____)
 - i) el software del SGBD puede acceder a diversas bases de datos al extraer su catálogo. (____)
 - j) La mayor parte de las aplicaciones de base de datos tienen ciertas restricciones de integridad que deben cumplir los datos. (____)
- 3.Cuál de las siguientes afirmaciones no corresponden a las funciones del administrador de la base de datos (**DBA**)
- a) Definir el esquema interno (DDL).
 - b) Evaluar el desempeño y responder a cambios en los requerimientos.
 - c) Definir el tamaño de la unidad de almacenamiento de la información contenida en la base de datos.
 - d) Definir verificaciones de seguridad e integridad.
- 4.Cuál de las siguientes afirmaciones no corresponde a los tipos de usuarios finales.
- a) Los usuarios esporádicos.
 - b) Los usuarios externos.
 - c) Los usuarios simples.
 - d) Los usuarios autónomos.
5. El SGBD debe de contar con un subsistema de seguridad y autorización que permita al DBA crear cuentas y especificar restricciones para ellas. La característica que está relacionada a este hecho es:
- a) Cumplimiento de las restricciones de integridad.
 - b) Restricciones de los accesos no autorizados.
 - c) Representación de vínculos complejos entre los datos.
 - d) Suministro de múltiples interfaces con los usuarios.
6. Entre las funciones de un **DBA** están:
- a) Definir verificaciones de seguridad e integridad.

- b) Definir procedimientos de respaldo y recusación.
 - c) Definir o participar en la creación del esquema.
 - d) Diseñar e implementar los módulos e interfaces del **SGBD**.
- 7.Cuál de las siguientes afirmaciones no corresponden a las características del enfoque de bases de datos.
- a) Comportamiento de datos y procesamiento de transacciones multiusuario.
 - b) Naturaleza autodescriptiva de los sistemas de bases de datos.
 - c) Separación entre los programas y los datos.
 - d) Integridad de los datos en los sistemas de bases de datos.
8. La arquitectura de una base de datos, de acuerdo con la normatividad ANSI/SPARQ, básicamente está constituida por tres niveles los cuales forman las diferentes visiones o enfoques que puede tener una base de datos estos niveles son.
- a) Nivel externo, nivel conceptual, nivel interno.
 - b) Nivel externo, nivel conceptual, nivel lógico.
 - c) Nivel conceptual, nivel lógico, nivel físico.
 - d) Nivel lógico, nivel físico, nivel interno.
9. Completar las siguientes afirmaciones.
- a) Los diseñadores de bases de datos se encargan de _____ los datos que se almacenarán en la base de datos y de _____ apropiadas para representar y almacenar dichos datos.
 - b) El SGBD debe _____ para definir tales restricciones y hacer que se cumplan. La forma más simple de restringir la integridad consiste en especificar _____ para cada elemento de información.
 - c) Otra aplicación reciente de los sistemas de base de datos consiste en ofrecer recursos para definir _____ que permitan deducir o inferir información nueva a partir de los datos almacenados. A estos sistemas se les conoce como bases de datos _____.
 - d) El conjunto formado por la base de datos, el software, el hardware y el personal apropiado lo llamaremos _____.
 - e) Todo SGBD multiusuario, como su nombre lo indica, debe permitir a varios usuarios tener _____ a la base de datos.
 - f) Una vista puede ser un _____ o contener datos virtuales que se deriven de los archivos de la base de datos, pero que no estén almacenados explícitamente.
 - g) Un SGBD ofrece a los usuarios una representación conceptual de los datos que no incluye muchos de los detalles de cómo se almacenan. En términos

informales, un _____ es un tipo de abstracción de los datos con que se obtiene esta representación conceptual.

- h) Es preciso que el SGBD pueda representar diversos vínculos complejos de los datos y también obtener y actualizar con _____ datos que estén mutuamente relacionados.
- i) El nivel externo o de vistas incluye varios _____ o _____.
- j) El proceso de transformar solicitudes y resultados de un nivel a otro se denomina _____ o _____.

10. Explique las características deseables de un sistema gestor de bases de datos.