



# Capítulo 1: Introducción

Fundamentos de Bases de datos, 5ª Edición.

©Silberschatz, Korth y Sudarshan

Consulte [www.db-book.com](http://www.db-book.com) sobre condiciones de uso





# Capítulo 1: Introducción

- Objetivo principal de los sistemas de bases de datos
- Visión de los datos
- Lenguajes de bases de datos
- Bases de datos relacionales
- Diseño de bases de datos
- Bases de datos basadas en objetos y semiestructuradas
- Almacenamiento de datos y consultas
- Gestión de transacciones
- Arquitectura de las bases de datos
- Usuarios y administradores de bases de datos
- Estructura general de un sistema
- Historia de los sistemas de bases de datos





# Sistema de gestión de bases de datos (SGBD)

- SGBD contiene información sobre una empresa en particular
  - Colección de datos relacionados
  - Conjunto de programas de acceso a los datos
  - Entorno que resulta *conveniente y eficiente* en su uso
- Objetivo: gestionar gran cantidad de información
  - Definición de estructuras para almacenar la información.
  - Provisión de mecanismos para su manipulación
  - Garantizar la fiabilidad de la información almacenada
  - Soportar caídas del sistema
  - Controlar intentos de accesos no autorizados
  - Administrar el uso del SGBD compartido entre usuarios evitando resultados anómalos





# Sistema de gestión de bases de datos (SGBD)

- Aplicaciones de los sistemas de bases de datos:
  - Banca: cualquier tipo de transacción
  - Líneas aéreas: reservas, información de planificación, etc.
  - Universidades: registros, cursos
  - Ventas: información de clientes, productos y compras
  - Minoristas en línea: seguimiento de órdenes, recomendaciones personalizadas
  - Producción: producción, inventarios, pedidos, cadena de producción
  - Recursos humanos: información sobre los empleados, salarios, impuestos y beneficios
- Las bases de datos tocan todos los aspectos de nuestras vidas





# Objetivo principal de los sistemas de bases de datos

- Inicialmente, las aplicaciones de los sistemas de bases de datos se construyeron encima de los sistemas de archivos
- Inconvenientes de utilizar sistemas de archivos para almacenar datos:
  - Redundancia e inconsistencia de datos
    - ▶ Diversos formatos de archivos, duplicación de la información en archivos diferentes
  - Dificultad en el acceso a los datos
    - ▶ Necesidad de escribir un programa nuevo por cada tarea nueva
  - Aislamiento de datos — diversos archivos con diferentes formatos
  - Problemas de integridad
    - ▶ Restricciones de consistencia (p. e. el saldo de una cuenta bancaria  $> 0$ ) pasan a formar parte del código del programa
    - ▶ Dificultad para añadir restricciones nuevas o modificar las existentes





# Objetivo principal de los sistemas de bases de datos (cont.)

- Inconvenientes de utilizar sistemas de archivos (cont.)
  - Problemas de atomicidad
    - ▶ Los fallos pueden dejar a las bases de datos en un estado de inconsistencia si se han llevado a cabo actualizaciones parciales
    - ▶ Por ejemplo, la transferencia de fondos de una cuenta bancaria a otra debe ser completa o no llevarse a cabo
  - Anomalías en el acceso concurrente
    - ▶ Se necesita el acceso concurrente para obtener un buen rendimiento
    - ▶ El acceso concurrente sin control puede conducir a tener datos inconsistentes
      - P. e. dos personas leyendo y actualizando un saldo al mismo tiempo
  - Problemas de seguridad
    - ▶ Dificultad de proporcionar acceso a parte de los datos pero no a todos
- Los sistemas de bases de datos ofrecen soluciones para todos los problemas anteriores





# Niveles de abstracción

**OBJETIVO:** ofrecer a los usuarios una visión abstracta de los datos

- **Nivel físico:** describe cómo se almacenan realmente los datos (p. e., nombre del cliente).
- **Nivel lógico:** describe qué datos se almacenan en la base de datos y las relaciones existentes entre ellos.

**type** *cliente* = **record**

```
id_cliente : string;  
nombre_cliente : string;  
calle_cliente : string;  
ciudad_cliente : integer;
```

**end;**

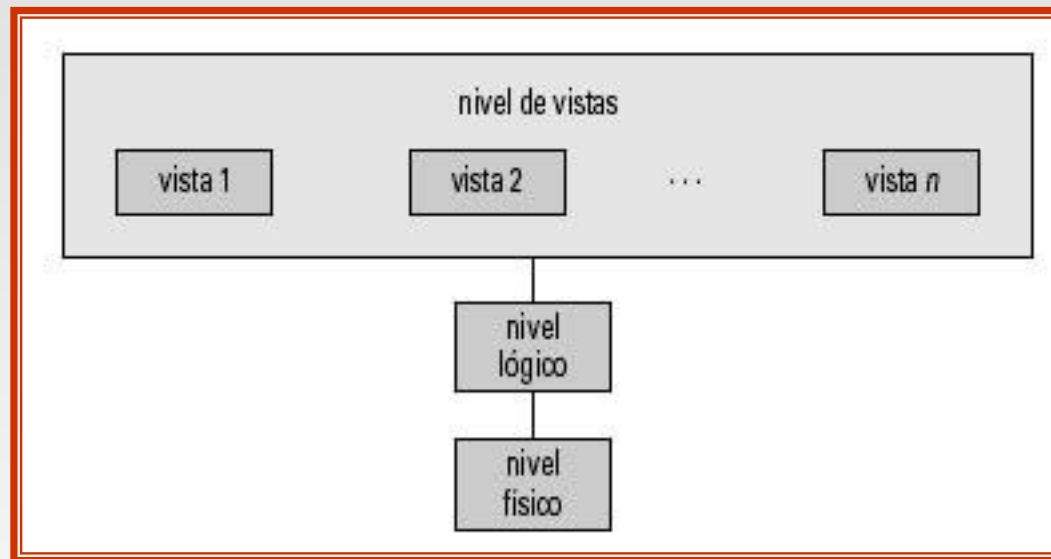
- **Nivel de vistas:** los programas de aplicación ocultan detalles de los tipos de datos. Las vistas también pueden ocultar información (p. e., salarios) por razones de seguridad.





# Visión de los datos

Una arquitectura para un sistema de bases de datos







# Instancias y esquemas

- Similar a los tipos y variables en los lenguajes de programación
- **Esquema** – la estructura lógica de la base de datos
  - Ejemplo: La base de datos se compone de información acerca de un grupo de clientes y cuentas, y de las relaciones entre ellos
  - Análogo a la información del tipo de una variable en un programa
  - **Esquema físico**: diseño de la base de datos a nivel físico
  - **Esquema lógico**: diseño de la base de datos a nivel lógico
- **Instancia** – el contenido real de la base de datos en un instante de tiempo determinado
  - Análogo al valor de una variable
- **Independencia física de los datos** – la capacidad de modificar el esquema físico sin cambiar el esquema lógico
  - Las aplicaciones dependen del esquema lógico
  - En general, las interfaces entre los diferentes niveles y componentes deben definirse adecuadamente de modo que los cambios en algunas partes no influyeran seriamente.





# Modelos de datos

- Colección de herramientas conceptuales para describir
  - los datos
  - las relaciones de datos
  - la semántica de los datos
  - consistencia entre los datos
- Modelo relacional
  - Colección de tablas para representar los datos y sus relaciones
  - Cada tabla tiene columnas, y las columnas un nombre único
  - Modelo basado en registros
  - Es el modelo de datos más ampliamente usado
- Modelo entidad-relación (principalmente para el diseño de bases de datos)
  - Se basa en una percepción del mundo real
  - Consiste en una colección de objetos básicos, entidades y relaciones





# Modelos de datos

...continua

- Modelo de datos orientado a objetos
  - Parece una extensión del modelo Entidad-Relacion
  - Incluye los conceptos de encapsulación, métodos y la identidad de los objetos
- Modelo de datos semiestructurados (XML)
  - Permite la especificación de datos donde los elementos de datos individuales del mismo tipo pueden tener diferentes conjuntos de atributos
- Modelos anteriores:
  - Modelo de red
  - Modelo jerárquico





# Lenguaje de manipulación de datos (LMD)

- Lenguaje para acceder o manipular los datos organizados mediante el modelo de datos apropiado
  - El LMD también se conoce como lenguaje de consultas
- Existen dos clases de lenguajes
  - **Procedimentales** – el usuario especifica que datos se necesitan y cómo han de obtenerse dichos datos
  - **Declarativos (no procedimentales)** – el usuario especifica qué datos se necesitan sin especificar cómo se han de obtener
- SQL es el lenguaje de consultas utilizado más ampliamente
- SQL es un lenguaje **NO PROCEDIMENTAL**





# Lenguaje de definición de datos (LDD)

- Notación de especificación para definir el esquema de la base de datos

Ejemplo:      **create table** *cuenta* (  
                                *número\_cuenta*      **char**(10),  
                                *saldo*                      **integer**)

- El compilador LDD genera un conjunto especial de tablas denominado *diccionario de datos*
- El diccionario de datos contiene metadatos (i.e., datos acerca de los datos)
  - Esquema de base de datos
  - Lenguaje de *almacenamiento y definición* de datos
    - ▶ Especifica la estructura de almacenamiento y los métodos de acceso utilizados
  - Restricciones de integridad
    - ▶ Restricciones de dominio
    - ▶ Integridad referencial (restricción **references** en SQL)
    - ▶ Asertos
  - Autorización





# Modelo relacional

- Ejemplo de tabla de datos en el modelo relacional

atributos

<i>id_cliente</i>	<i>nombre_cliente</i>	<i>calle_cliente</i>	<i>ciudad_cliente</i>	<i>número_cuenta</i>
19.283.746	González	Arenal, 12	La Granja	C-101
19.283.746	González	Arenal, 12	La Granja	C-201
67.789.901	López	Mayor, 3	Peguerinos	C-102
18.273.609	Abril	Preciados, 123	Valsaín	C-305
32.112.312	Santos	Mayor, 100	Peguerinos	C-217
33.666.999	Rupérez	Ramblas, 175	León	C-222
01.928.374	Gómez	Carretas, 72	Cerceda	C-201





# Ejemplo de base de datos relacional

<i>id_cliente</i>	<i>nombre_cliente</i>	<i>calle_cliente</i>	<i>ciudad_cliente</i>
19.283.746	González	Arenal, 12	La Granja
67.789.901	López	Mayor, 3	Peguerinos
18.273.609	Abril	Preciados, 123	Valsaín
32.112.312	Santos	Mayor, 100	Peguerinos
33.666.999	Rupérez	Ramblas, 175	León
01.928.374	Gómez	Carretas, 72	Cerceda

(a) La tabla *cliente*

<i>número_cuenta</i>	<i>saldo</i>
C-101	500
C-215	700
C-102	400
C-305	350
C-201	900
C-217	750
C-222	700

(b) La tabla *cuenta*

<i>id_cliente</i>	<i>número_cuenta</i>
19.283.746	C-101
19.283.746	C-201
01.928.374	C-215
67.789.901	C-102
18.273.609	C-305
32.112.312	C-217
33.666.999	C-222
01.928.374	C-201

(c) La tabla *impositor*





# SQL

■ **SQL:** lenguaje no procedimental ampliamente utilizado

- Ejemplo: Encontrar el nombre de el cliente con id\_cliente 192-83-7465

```
select  cliente.nombre_cliente  
from    cliente  
where  cliente.id_cliente = '192-83-7465'
```

- Ejemplo: Encontrar el saldo de todas las cuentas del cliente cuyo id\_cliente sea 192-83-7465

```
select  cuenta.saldo  
from    impositor, cuenta  
where  impositor.id_cliente = '192-83-7465' and  
        impositor.número-cuenta = cuenta.número_cuenta
```

■ Los programas de aplicación generalmente acceden a la base de datos a través de

- Extensiones de lenguajes que premitan SQL empotrado
- Interfaces de programas de aplicación (p. e. ODBC/JDBC) que permiten el envío de consultas SQL a una base de datos







# Diseño de la base de datos

Proceso de diseño de la estructura general de una base de datos:

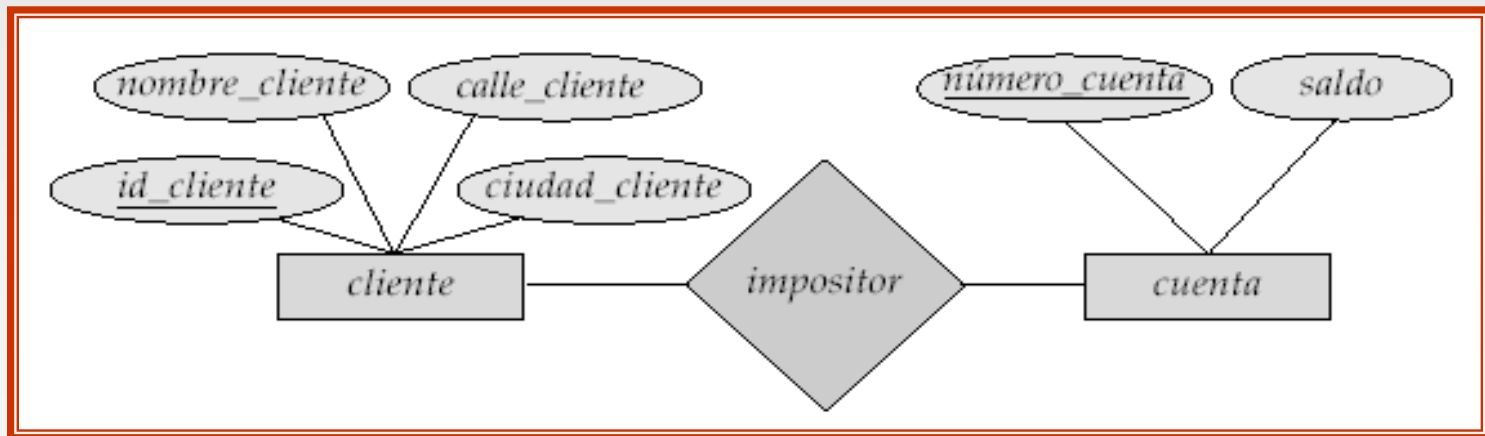
- Diseño lógico – Decidir el esquema de la base de datos. El diseño de la base de datos requiere encontrar una “buena” colección de esquemas de relación.
  - Decisión de negocio – ¿Qué atributos se deberían registrar en la base de datos?
  - Decisión informática – ¿Qué relación de esquemas se deberían utilizar y cómo se deberían distribuir los atributos entre los distintos esquemas de relación?
- Diseño físico – Decidir sobre las características físicas de la base de datos





# Modelo entidad-relación

- Modela una empresa como una colección de *entidades* y *relaciones*
  - Entidad: una “cosa” o un “objeto” en la empresa distinguible de otros objetos
    - ▶ Se describe mediante un conjunto de *atributos*
  - Relación: una asociación entre varias entidades
- Se representa gráficamente mediante un *diagrama entidad-relación*:



- *Concepto de Normalización*





# Modelos de datos relacional orientado a objetos

- Extiende el modelo de datos relacional incluyendo orientación a objetos y construcciones que manejan otros tipos de datos adicionales.
- Permite atributos de tuplas con tipos complejos, incluyendo valores no atómicos como son las relaciones anidadas.
- Preserva los fundamentos relacionales, en particular el acceso declarativo a los datos, aunque extiende la capacidad de modelado.
- Proporciona compatibilidad hacia arriba con lenguajes relacionales preexistentes.





# XML: Lenguaje de marcas extensible

- Definido por el consorcio WWW (W3C)
- Originariamente como lenguaje de marcado de documentos, no como lenguaje de base de datos.
- La posibilidad de especificar nuevas etiquetas y crear estructuras de etiquetas anidadas convierten a XML en un mecanismo perfecto para el intercambio de **datos**, no solo de documentos.
- XML se ha convertido en la base de una nueva generación de formatos de intercambio de datos.
- Existen una gran variedad de herramientas de análisis, presentación y consulta de documentos/datos en XML





# Gestión de almacenamiento

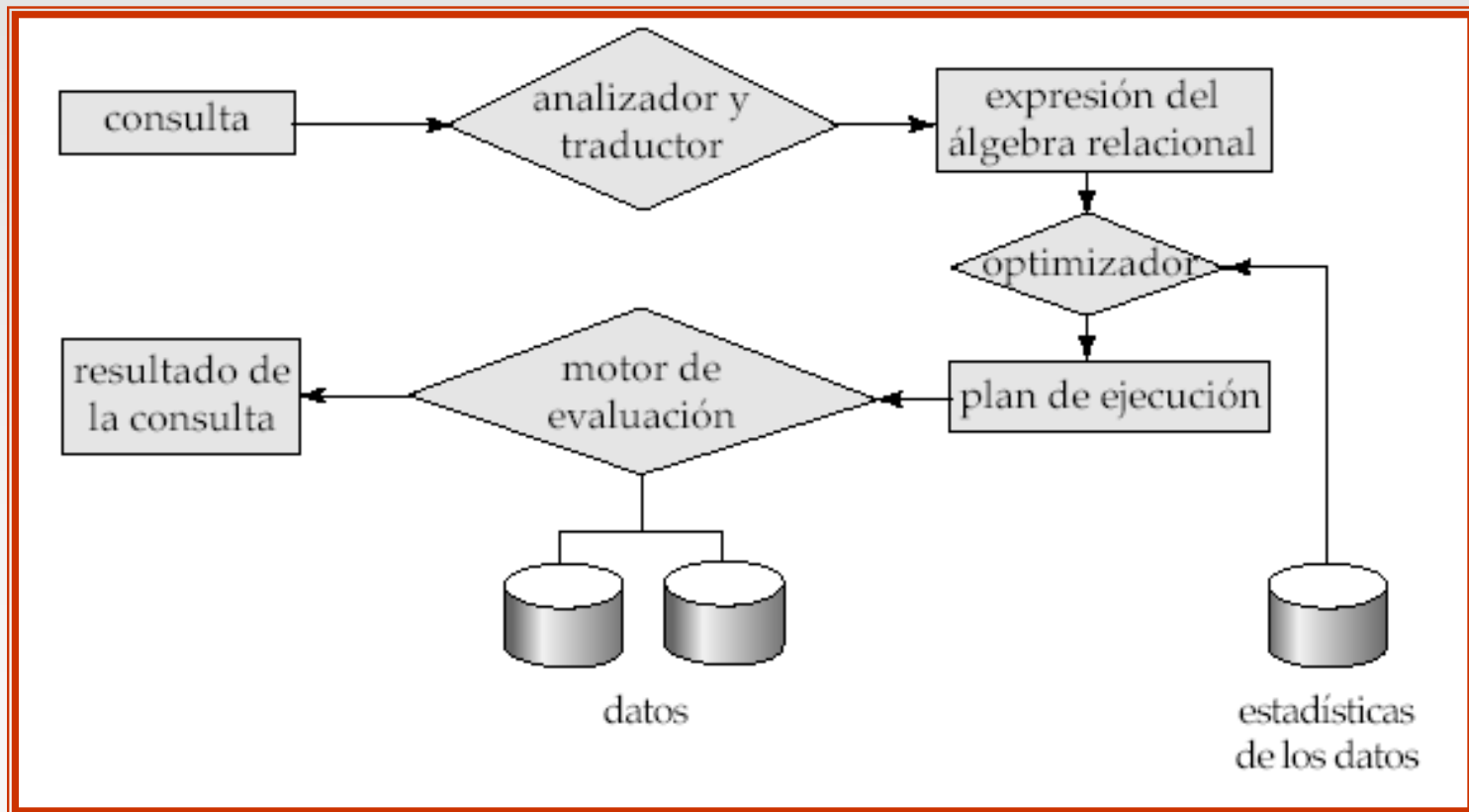
- El **gestor de almacenamiento** es un módulo de programa que proporciona la interfaz entre los datos de bajo nivel en la base de datos y los programas de aplicación y consultas emitidas al sistema.
- El gestor de almacenamiento es responsable de las siguientes tareas:
  - La interacción con el gestor de ficheros
  - El almacenamiento, recuperación y actualización eficiente de los datos
- Temas:
  - Almacenamiento
  - Organización de archivos
  - Indexación y asociación





# Procesamiento de consultas

1. Análisis y traducción
2. Optimización
3. Evaluación





# Procesamiento de consultas (cont.)

- Alternativas de evaluación de una consulta
  - Expresiones equivalentes
  - Algoritmos diferentes para cada operación
- La diferencia de coste entre una forma buena y una mala de evaluar una consulta puede ser enorme.
- Se necesita estimar el coste de las operaciones
  - Depende de forma crítica de la información estadística sobre las relaciones que debe mantener la base de datos.
  - Se necesita estimar las estadísticas para los resultados intermedios para estimar el coste de expresiones complejas.





# Gestión de transacciones

- Una **transacción** es una colección de operaciones que se llevan a cabo como una única función lógica en una aplicación de base de datos.
- El **componente de gestión de transacciones** asegura que la base de datos permanezca en un estado consistente (correcto) a pesar de los fallos del sistema (p.e., fallos de energía y caídas del sistema operativo) y de los fallos en las transacciones.
- El **gestor de control de concurrencia** controla la interacción entre las transacciones concurrentes para asegurar la consistencia de la base de datos.







# Arquitectura de la base de datos

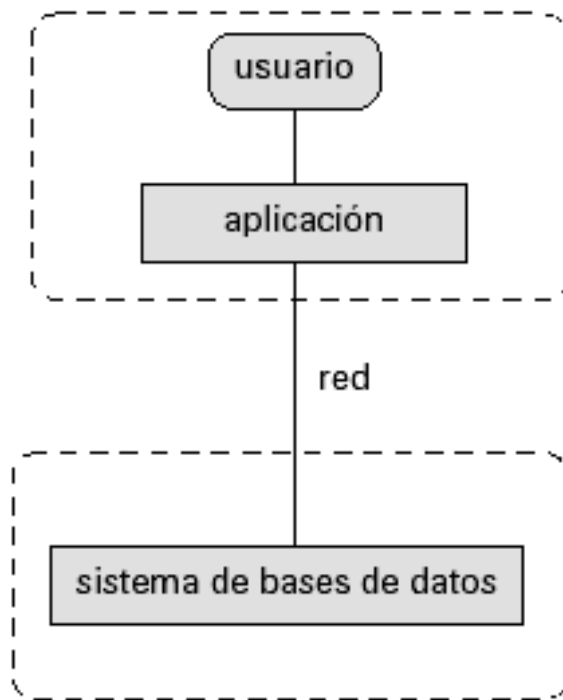
La arquitectura de una base de datos se ve muy influenciada por el sistema informático subyacente sobre el que se está ejecutando:

- Centralizado
- Cliente-servidor
- Paralelo (multi-procesador)
- Distribuido





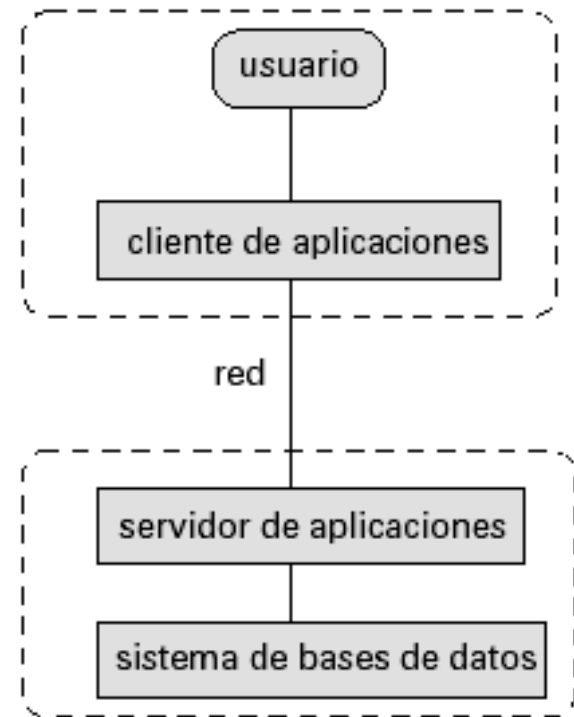
# Cliente Servidor - capas



a. arquitectura de dos capas

cliente

servidor



b. arquitectura de tres capas





# Usuarios de bases de datos

Los **usuarios** se diferencian por la forma en que esperan interactuar con el sistema

- **Programadores de aplicaciones** – interactúan con el sistema por medio de llamadas LMD
- **Usuarios sofisticados** – formulan sus peticiones en un lenguaje de consulta de bases de datos
- **Usuarios especializados** – escriben aplicaciones de bases de datos especializadas que no cuadran con el marco de procesamiento de datos tradicional
- **Usuarios normales** – invocan uno de los programas de aplicación permanentes que se han escrito con anterioridad
  - Ejemplos, las personas que acceden a bases de datos a través de la web, los cajeros automáticos, el personal de oficina





# Administrador de la base de datos

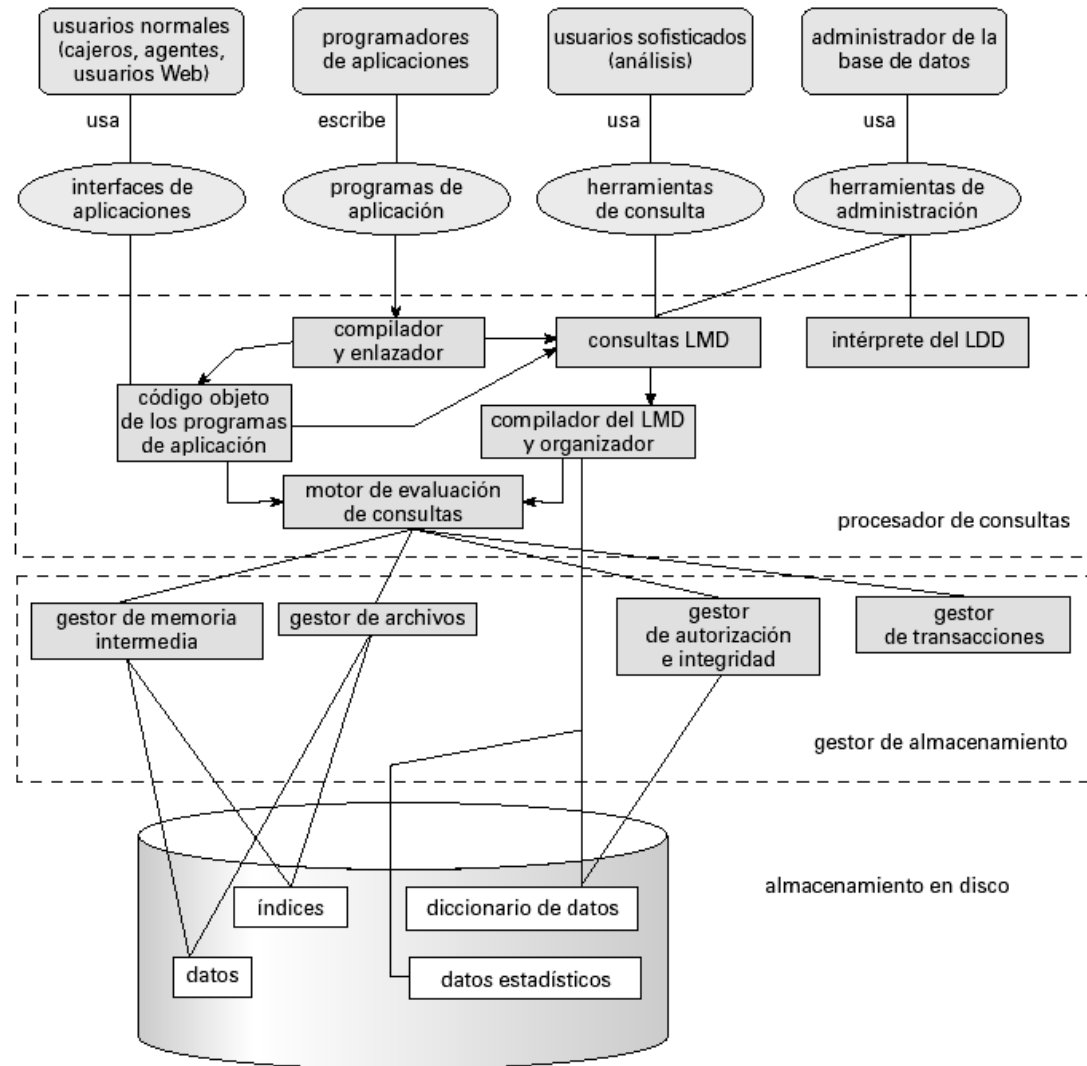
- Coordina todas las actividades del sistema de la base de datos; posee buenos conocimientos de los recursos y necesidades de información de la empresa.
- Entre las funciones del ABD se incluyen:
  - Definición del esquema
  - Estructura de almacenamiento y definición del método de acceso
  - Modificación del esquema y organización física
  - Concesión de autorización para el acceso a los datos
  - Especificación de las restricciones de consistencia
  - Actuar como enlace con los usuarios
  - Supervisión de rendimiento y respuesta a cambios de los requisitos





# Estructura general del sistema

## FUNDAMENTOS DE BASES DE DATOS





# Historia de los sistemas de bases de datos

- Años 50 y primeros 60:
  - Procesamiento de datos con cintas magneticas como almacenamiento
    - ▶ Las cintas sólo proporcionan acceso secuencial
  - Tarjetas perforadas como entrada
- Finales de los 60 y años 70:
  - Los discos duros permiten acceso directo a los datos
  - Modelos de datos jerárquicos y en red en amplio uso
  - Ted Codd define el modelo de datos relacional
    - ▶ Obtuvo el premio Turing de la ACM por su contribución
    - ▶ IBM Research comienza el prototipo System R
    - ▶ UC Berkeley comienza el prototipo Ingres
  - Alto rendimiento en el procesamiento de transacciones (para su época)





# Historia (cont.)

- Años 80:
  - Los prototipos relacionales de investigación evolucionan a sistemas comerciales
    - ▶ SQL se convierte en el estándar industrial
  - Sistemas de bases de datos paralelos y distribuidos
  - Sistemas de bases de datos orientados a objetos
- Años 90:
  - Grandes aplicaciones de ayuda a la toma de decisiones y minería de datos
  - Grandes almacenes de datos multi-terabyte
  - Emerge el comercio Web
- Años 2000:
  - Estándares XML XQuery
  - Administración automatizada de bases de datos





# Fin capítulo 1

**Fundamentos de Bases de datos, 5ª Edición.**

©Silberschatz, Korth y Sudarshan

Consulte [www.db-book.com](http://www.db-book.com) sobre condiciones de uso

