



Capítulo 1: Introducción

Database System Concepts, 5th Ed.

©Silberschatz, Korth and Sudarshan

See www.db-book.com for conditions on re-use





Capítulo 1: Introducción

- Aplicaciones de los SBD
- Propósito de SBD
- Visión de los datos
- Lenguajes de BD
- BD relacionales
- Diseño de BD
- BD basados en objetos y semiestructuras
- Almacenamiento de datos y consultas
- Gestión de transacciones
- Arquitectura de BD
- Usuarios y Administradores de BD
- Historia de los SBD





SISTEMA GESTOR DE BASE DE DATOS (SGDB)

- SGDB contiene la información relevante para una empresa
 - Colección de datos interrelacionados
 - Conjunto de programas para acceder los datos
 - En un ambiente que sea práctico y eficiente
- 1.1 Aplicaciones de los SBD:
 - Banca: todo tipo de transacciones bancarias
 - Aerolíneas: reservaciones, información de horarios
 - Universidades: registros, cursos, calificaciones
 - Transacciones de tarjetas de crédito: clientes, productos, compras
 - Comercio en línea: seguimiento de pedidos y generación de listas de recomendaciones
 - Producción: inventarios, pedidos, cadena de proveedores
 - Recursos Humanos: info. Sobre empleados, salarios, impuestos





1.2 Propósito de los SBD

- En los años sesenta, una manera de guardar la información en la computadora era almacenarla en archivos del sistema operativo.
- Serie de inconvenientes en usar un sistema de procesamiento de archivos:
 - Redundancia e inconsistencia de los datos
 - ▶ Archivo con múltiples formatos, duplicación de la información en varios archivos
 - Dificultad en el acceso a los datos
 - ▶ Necesidad de escribir un nuevo programa de aplicación necesaria
 - Aislamiento de los datos — múltiples archivos y formatos
 - Problemas de integridad
 - ▶ Restricciones de consistencia (ejemplo: cuentas bancarias con un balance > 0)
 - ▶ Cuando se cambian las restricciones o añaden nuevas es difícil cambiar los todos programas





1.2 Propósito de los SBD (Cont.)

- Serie de inconvenientes en usar un sistema de procesamiento de archivos:(cont.)
 - Problemas de atomicidad
 - ▶ Fallas pueden dejar BD en un estado de inconsistencia con actualizaciones parciales
 - ▶ Por ejemplo:Transferencias de fondos de una cuenta a otra debe ser atómica –debe ocurrir en su totalidad o no ocurrir en lo absoluto
 - Anomalías en el acceso concurrente
 - ▶ Actualización de datos de varios usuarios simultáneamente
 - ▶ Actualización de datos simultaneos fuera de control puede crear muchas inconsistencias

Ejemplo: Dos personas leyendo su balance y actualizándolo al mismo tiempo
 - Problemas de seguridad
 - ▶ No todos los usuario de SBD deben poder acceder todos los datos
- SBD ofrecen soluciones para todo estos problemas.





1.3 Visión de los datos

1.3.1 Abstracción de los datos

- **Nivel físico:** describe *cómo* se almacenan realmente los datos .
- **Nivel lógico:** describe *qué* datos se almacenan en la BD y qué relaciones existen entre esos datos.

type *customer* = **record**

```
customer_id : string;  
customer_name : string;  
customer_street : string;  
customer_city : integer;
```

end;

- **Nivel de vistas:** aplicación de programas que esconden los detalles de todos los tipos de datos. El sistema puede proporcionar muchas vistas para la misma BD

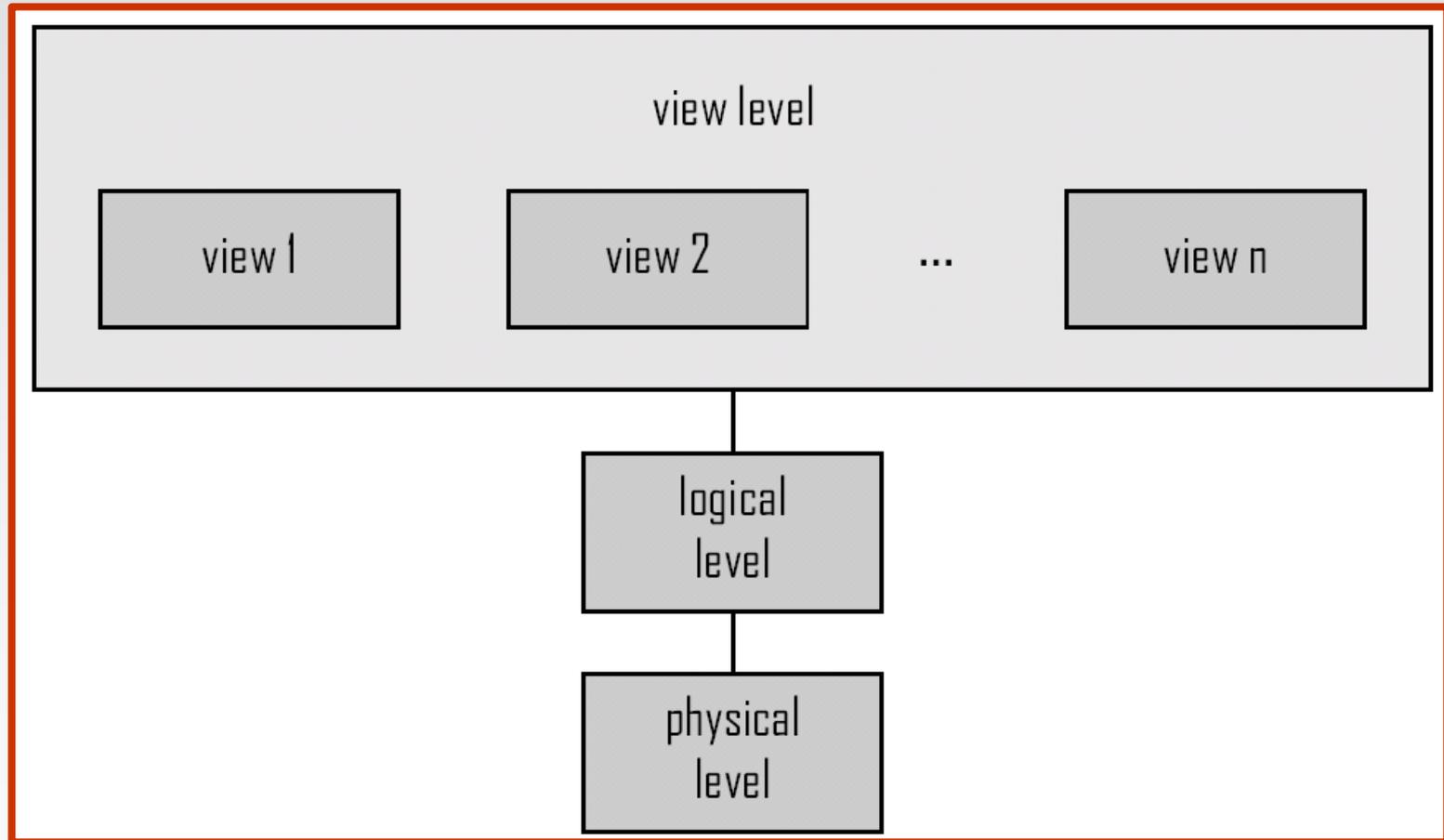
Las vistas pueden esconder información (tal como el salario del empleado) por cuestiones de seguridad.





1.3 Visión de los datos

Los 3 niveles de abstracción de datos





Ejemplares y Esquemas

- Similar a las variables y tipos en un lenguaje de programación
- **Ejemplar** – colección de información almacenada en la BD en un momento dado
 - Analogía los valores de una variable en un instante dado

Esquema – declaraciones de las variables

El diseño general de la BD y los esquemas rara vez se modifican

- Ejemplo: La BD consiste en información acerca de una serie de clientes y cuentas y la relación que hay entre ellos.
 - **Esquema físico:** describe el diseño de la BD a un nivel físico.
 - **Esquema lógico:** describe el diseño de la BD a un nivel lógico
 - **Subesquemas:** varios esquemas en el nivel de vistas
- **Independencia física respecto a los datos** – la habilidad de modificar el esquema físico sin cambiar el esquema lógico. En otras palabras, si no dependen del esquema físico y por tanto no hace falta volver a escribirlos si se modifica el esquema físico.
 - Las aplicaciones dependen del esquema lógico
 - En general, las interfaces entre varios niveles y componentes deben definirse bien de tal manera que los cambios en algunas partes no influyan seriamente otras.





1.3.3 Modelos de Datos

- Una colección de herramientas conceptuales para describir:
 - Datos
 - Sus relaciones
 - Su semántica
 - Restricciones de consistencia
- Modelo Relacional usa una colección de tablas para representar tanto los datos como sus relaciones. Cada tabla tiene varias columnas. (se usa mucho en el diseño de BD) . Cap 2-7 Ejemplo 1.10
- Modelo entidad relación se base en una percepción del mundo real que consiste en una colección de obj. básicos, denominados entidades, y de las relaciones entre ellos. Cap 6
- Modelo de datos orientado a objetos es una extensión del mod. E-R con los conceptos de encapsulación, mét. (func.) y la entidad de los objetos. Cap 9.
- 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.
- Otros antiguos modelos que precedieron al m. Relacional
 - Modelo de datos de red
 - Modelo de datos jerárquico

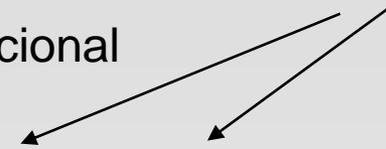




Modelo Relacional

- Ejemplo tabular de los datos en un modelo relacional

Attributos



<i>customer_id</i>	<i>customer_name</i>	<i>customer_street</i>	<i>customer_city</i>	<i>account_number</i>
192-83-7465	Johnson	12 Alma St.	Palo Alto	A-101
192-83-7465	Johnson	12 Alma St.	Palo Alto	A-201
677-89-9011	Hayes	3 Main St.	Harrison	A-102
182-73-6091	Turner	123 Putnam St.	Stamford	A-305
321-12-3123	Jones	100 Main St.	Harrison	A-217
336-66-9999	Lindsay	175 Park Ave.	Pittsfield	A-222
019-28-3746	Smith	72 North St.	Rye	A-201





Un ejemplo de BD relacional

<i>customer_id</i>	<i>customer_name</i>	<i>customer_street</i>	<i>customer_city</i>
192-83-7465	Johnson	12 Alma St.	Palo Alto
677-89-9011	Hayes	3 Main St.	Harrison
182-73-6091	Turner	123 Putnam Ave.	Stamford
321-12-3123	Jones	100 Main St.	Harrison
336-66-9999	Lindsay	175 Park Ave.	Pittsfield
019-28-3746	Smith	72 North St.	Rye

(a) The *customer* table

<i>account_number</i>	<i>balance</i>
A-101	500
A-215	700
A-102	400
A-305	350
A-201	900
A-217	750
A-222	700

(b) The *account* table

<i>customer_id</i>	<i>account_number</i>
192-83-7465	A-101
192-83-7465	A-201
019-28-3746	A-215
677-89-9011	A-102
182-73-6091	A-305
321-12-3123	A-217
336-66-9999	A-222
019-28-3746	A-201

(c) The *depositor* table





1.4.1 Lenguaje de Manipulación de Datos (LMD)

- Lenguaje para acceder y manipular los datos organizados mediante un modelo de datos correspondiente.
- Dos clases de lenguajes
 - **LMDs Procedimentales** – el usuario especifica qué datos son requeridos y cómo obtener esos datos.
 - **LMDs Declarativos (LMDs no procedimentales)** – el usuario especifica qué datos se necesitan sin que haga falta que especifique cómo obtener esos datos. (más fáciles de aprender)

Consulta es una instrucción para recuperación de info, el LMD implicada se llama lenguaje de consultas

- SQL es el lenguaje de consulta más usado
- SQL STRUCTURE QUERY LANGUAGE





Lenguaje de Definición de Datos (LDD)

- Los esquemas de las BD se especifican mediante un conjunto de definiciones expresadas en el lenguaje LDD
- La estructura de almacenamiento y los métodos de acceso usados por el SBD se especifican en un tipo especial de LDD denominado leng de almacenamiento y def de datos.

Example: **create table** *account* (
 account-number **char**(10),
 balance **integer**)

- Restricciones de consistencia
 - ▶ Restricciones de dominio; valores posibles (enteros, fecha), forma más elemental de restricción de integridad.

Ejemplo: Restricciones de consistencia (saldo en una cuenta)

 - ▶ Integridad referencial , el valor de una relación en un conj A también aparece para un conj B en otra relación
 - ▶ Aserto; una condición que la BD debe satisfacer siempre ej, préstamo
- Autorización, lectura, aut. inserción, de actualización, de eliminación





1.5.2 SQL

- **SQL:** no es procedimental. Entrada tabla(s) salida tabla
 - Ejemplo: Halla el nombre del cliente con el customer-id 192-83-7465

```
select    customer.customer_name
from      customer
where     customer.customer_id = '192-83-7465'
```
 - Ejemplo: Busca todos los números de cuenta y saldos del cliente con customer-id 192-83-7465

```
select    account.balance
from      depositor, account
where     depositor.customer_id = '192-83-7465' and
          depositor.account_number = account.account_number
```
- 1.5.4 Acceso a las BD desde prog de aplicación (los cuales se usan para interactuar con BD)
- Con SQL no se puede acessar algunos cálculos así que se necesita un lenguaje anfitrión como C++, java.
- Ejem; generación de nóminas, recargos, bonos
 - Extendiendo la sintaxis del lenguaje anfitrión que incorpore las llamadas LMD (to allow embedded SQL)
 - Aplicación de programas de interfase (e.g., ODBC/JDBC) (which allow SQL queries to be sent to a database)





Diseño de BD

El proceso de diseño de la estructura general de las BD:

- Diseño Lógico – Decidir en el esquema de la BD. El diseño de la BD requiere de encontrar una buena colección de esquemas .
 - Decisión de negocios- que atributos se debería guardar en la BD?
 - Computación – que relaciones de esquemas deberíamos tener en una base de datos y como los atributos se distribuyen entre los diversos esquemas?

- Diseño físico – Decidir en las características físicas de la BD como estructuras de almacenamiento interno, organización de los archivos.





El modelo de entidad relacion

- Modelos en una empresa es una colección de entidades y sus relaciones
 - Entidad una cosa o objeto en la empresa que es distinguible de otros objetos
 - ▶ Descritos por una serie de atributos
 - Relación: una relación entre varias entidades
- Representación grafica de una entidad-relación

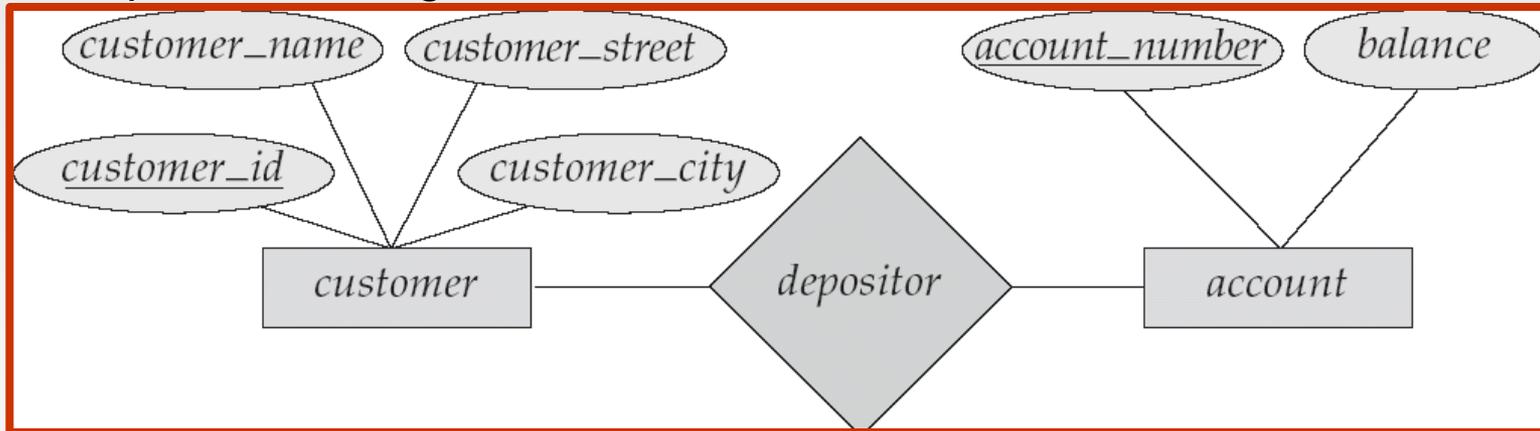




Figure 1.4

<i>customer_id</i>	<i>account_number</i>	<i>balance</i>
192-83-7465	A-101	500
192-83-7465	A-201	900
019-28-3746	A-215	700
677-89-9011	A-102	400
182-73-6091	A-305	350
321-12-3123	A-217	750
336-66-9999	A-222	700
019-28-3746	A-201	900





Gestor de Almacenamiento

- **Gestor de almacenamiento** es un modulo de programa que proporciona un interfaz entre los datos almacenados a bajo nivel en la BD y la aplicación de programas y consultas
- El gestor de almacenamiento es responsable de los siguiente:
 - Interacción con el manejo de archivos
 - Eficiencia en el almacenamiento, recuperación y actualización de los datos
 - Gestor de autorización e integridad, comprueban que satisfagan las restricciones de integridad y autorización de los usuarios para tener acceso a la BD
 - Gestor de transacciones garantiza que la BD quede en un edo consistente
 - Gestor de Archivos; la asignación de espacio almacenamiento de disco y las estructuras de datos usadas para representar la inf almacenada
 - Gestor de memoria intermedia; es el responsable de traer los datos desde el disco de almacenamiento a la mem principal (memoria cache)
- El G. de almacenamiento implementa varias estructuras de datos como parte de la implementación física del sistema:
 - Archivo de datos, que almacenan la BD en si misma
 - Diccionario de datos que almacena metadatos de la estructura de la BD en particular su esquema
 - Índices puede proporcionar un acceso rápido a los elementos de datos

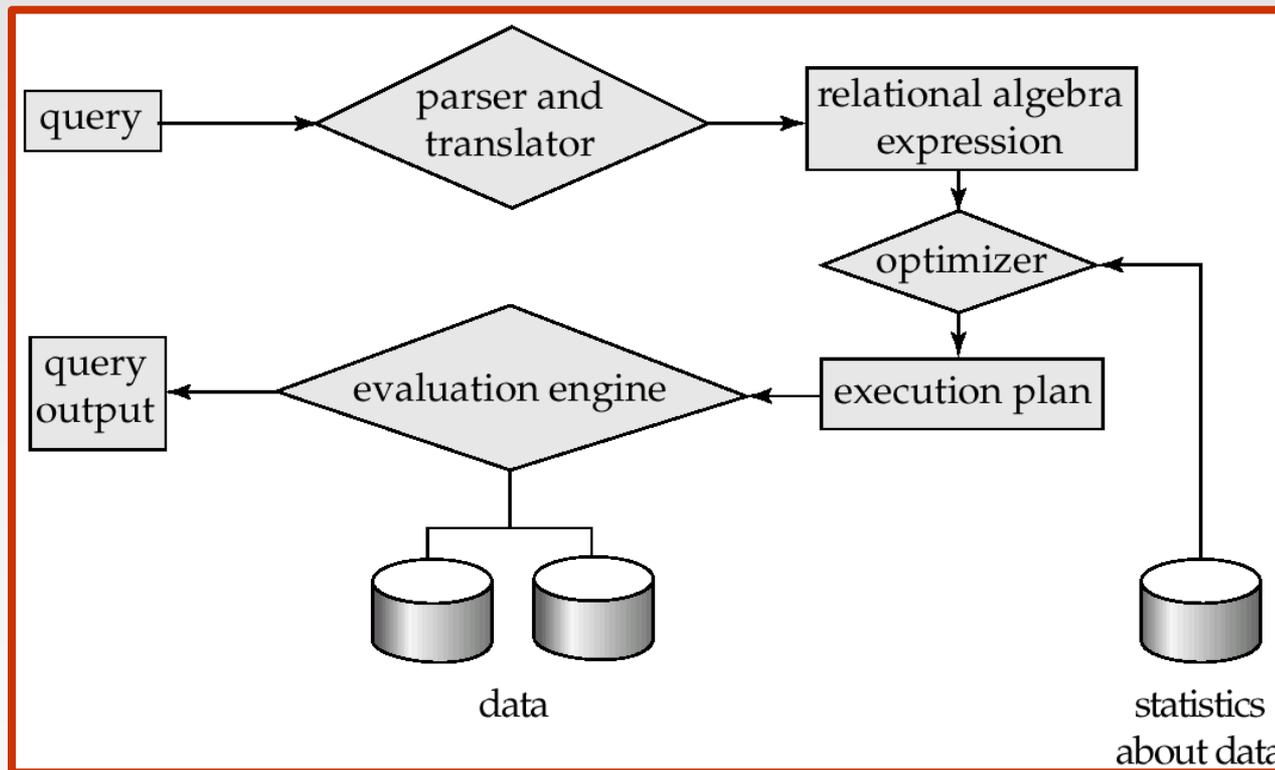




Procesamiento de Consulta

1. Interprete del LDD
2. Compilador del LMD
3. Motor de Evaluación

1. Parsing and translation
2. Optimization
3. Evaluation





Gestión de Transacciones

- Una transacción es un conjunto de operaciones que lleva a cabo una única función lógica en una aplicación de BD
- **Componente de gestión de transacciones** es responsable de garantizar las propiedades de atomicidad y durabilidad (persistencia aún en fallos)
- **Gestor de control de concurrencia** controla la interacción entre las transacciones concurrente para garantizar la consistencia de la BD.





Arquitectura de las BD

La arquitectura de los sistemas de BD se ve muy influenciada por el sistema informático subyacente sobre el que se ejecuta BD:

- Centralizado
- Cliente-servidor- usuarios remotos BD, donde se ejecutan BD
- Paralelo (multi-procesadores)
- Distribuidas (geográficamente)





Usuario de BD

Usuarios están diferenciados por la forma en que esperan interactuar con el sistema.

- **Usuarios normales** – son usuarios no sofisticados que interactúan con el sistema invocando alguno de los programas de aplicación que se han escrito previamente.
 - Ejemplo personas que accesan BD a través de la Web, cajeros
- **Programadores de aplicaciones** – interactúan con el sistema a través de desarrollo rápido de aplicaciones (DRA)
- **Usuarios sofisticados** – formulan consultas en un lenguaje SQL
- **Usuarios especializados** – escriben aplicaciones de BD especializadas que no encajan en el marco tradicional de procesamiento de datos.





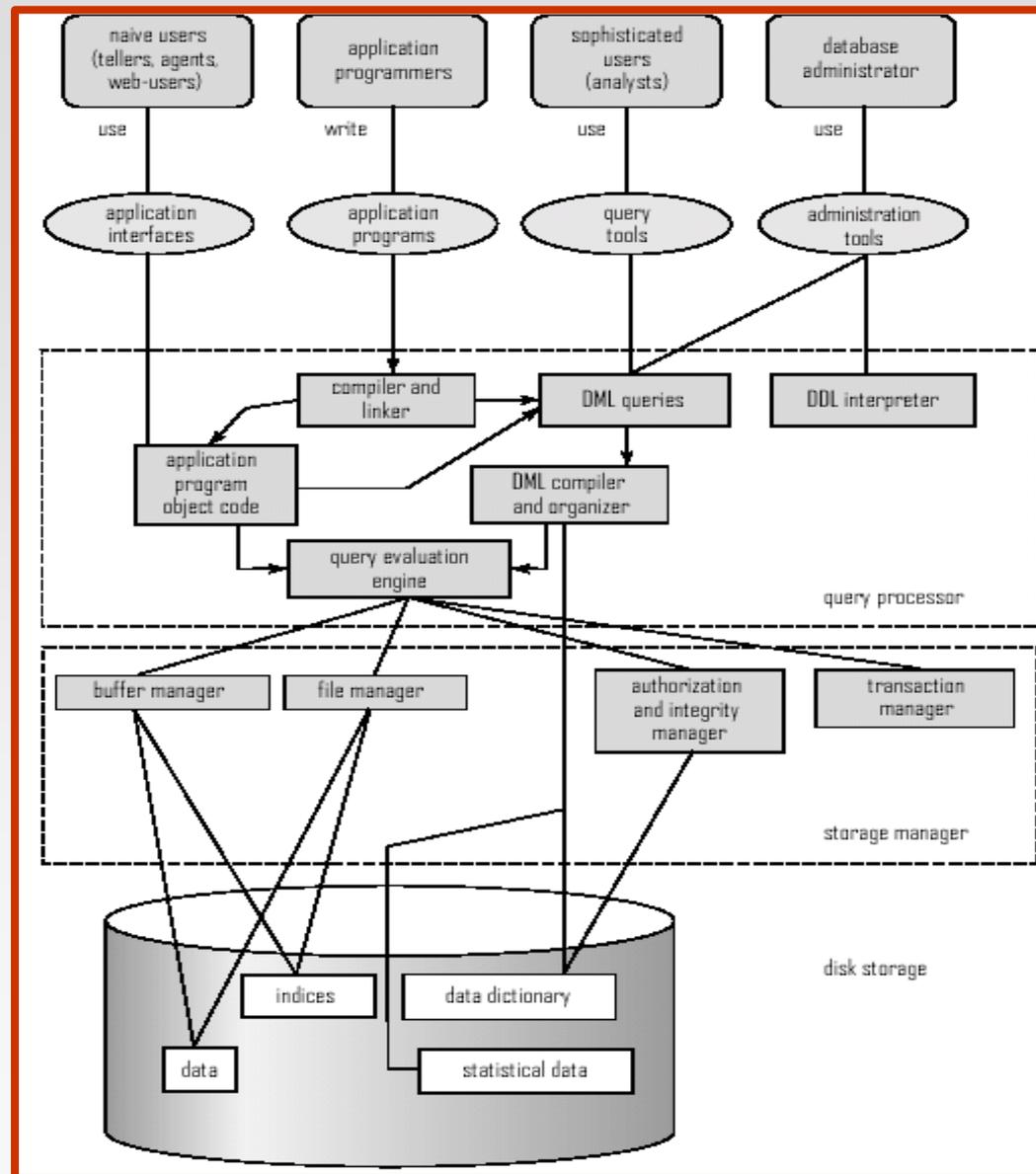
Administrador de BD

- Coordina todas las actividades de un SGBD, el administrador de BD tiene un buen conocimiento de la información de la empresa y sus necesidades.
- Las funciones de un ABD:
 - Definición del esquema
 - Definición de la estructura y de la organización física
 - Concesión de autorización para acceso de los datos
 - Mantenimiento rutinario





Arquitectura del sistema





Historia de los SBD

- 1950s y principios 1960s:
 - Cintas magnéticas para el procesamiento y almacenamiento de datos
 - ▶ Cintas podían solo leerse secuencialmente
 - Entrada de datos combinando las cintas y tarjetas perforadas
- finales 1960s y 1970s:
 - Discos duro que permite el acceso directo a los datos
 - Crearon BD de red y BD jerárquicas
 - Ted Codd define el modelo relacional +
 - ▶ Gana el ACM Turing Award
 - ▶ Investigación del IBM System R prototype
 - ▶ UC Berkeley comienza Ingres prototype





Historia (cont.)

- 1980s:
- La investigación de prototipo relacional evoluciona en sistemas comerciales como
- SQL que se vuelve estándar en la industria
- BD Paralelas y distribuida
- SBD orientado a objetos

- *1990s:*
- Toma de decisiones – consultas
- Procesamiento de transacciones – actualizaciones
- Crecimiento explosivo del Word Wide Web 24x7

- 2000s:
- XML y su lenguaje de consultas asociado
- Administración automática para minimizar el esfuerzo de administración





Tarea
5ta Edición
Ejercicios 1-4

Fin Capítulo 1

Database System Concepts, 5th Ed.

©Silberschatz, Korth and Sudarshan

See www.db-book.com for conditions on re-use





Figure 1.7

