



**Dpto. Ciencias e Ingeniería de la Computación
Universidad Nacional del Sur**

ELEMENTOS DE BASES DE DATOS

Segundo Cuatrimestre 2015

Clase 1:

**Presentación de la Materia –
Sistemas de Gestión de Bases de
Datos**

Mg. María Mercedes Vitturini
[mvitturi@uns.edu.ar]



Sobre la cátedra

- **Profesor:**

- María Mercedes Vitturini [mvitturi@cs.uns.edu.ar]
- Horario de Consulta: Lunes 15:00 hs. Oficina 210.

- **Asistente:**

- Walter Grandinetti [wmg@cs.uns.edu.ar]

- **Auxiliares:**

- Marian Fernández Benassati
- María Gabriela García Franz



Horarios de Clase



Teoría

- *Martes (aula 10) y Viernes (aula 11): de* 14:00 a 16:00.
Edificio Nuevo Palihue.

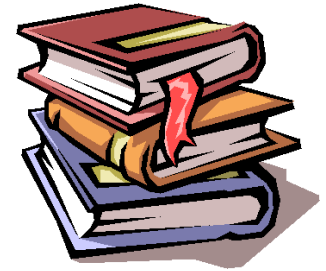
Práctica

- **Martes (aula 10) y viernes (aula 11)** de 16:00 a 18:00. Edificio Nuevo Palihue.
- Laboratorio: a confirmar

Material de la cátedra:

- Página de la materia <http://cs.uns.edu.ar/materias/ebd/>
- *En la fotocopiadora del CeCom, sin compromiso de dejar las copias al día.*

Bibliografía



Bases de Datos

- *Database System Concepts*. Abraham Silberschatz, Henry F. Korth & S. Sudarshan (Sixth Edition) 2010.
- *Database Systems – The Complete Book*. Hector García Molina, Jeffrey Ullman, Jennifer Widom. 2nd Ed. 2009
- *Databases and Transaction Processing*. P. Lewis. Addison-Wesley. 2003.
- *Principles of Database and Knowledge based Systems*. Jeffrey Ullman. Academic Press. 1990.

Ingeniería de Software

- *Software Engineering* – Carlo Ghezzi. 2001. Capítulos 1, 2, 3,7.
- *Ingeniería de Software* – Ian Sommerville.
- *Ingeniería de Software – Un enfoque Práctico*. Roger Pressman. Quinta Edición 2002.

Condiciones de Cursado

⇒ **Dos parciales**

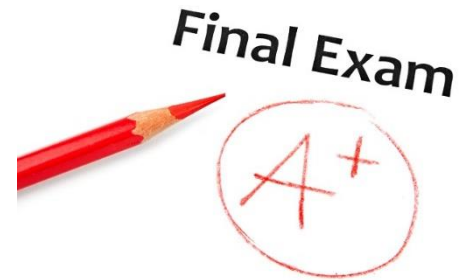
– **Martes 6/10 y Viernes 13/11**

⇒ **Un recuperatorio**

– **Martes 24/11**

⇒ **Entregas de proyecto.** A definir en la práctica.

- Los parciales se califican con A y B (aprobado), C y D (desaprobados)
- La nota final conceptual de los proyectos aprobados es A ó B.
- Los alumnos que cumplan con las condiciones de cursado y rindan en las fechas de diciembre acceden a un examen de promoción.



Elementos de Bases de Datos

ELEMENTOS DE BASES DE DATOS es una asignatura de la disciplina *Desarrollo de Sistemas*. El contenido de este curso abarca conceptos clásicos del proceso de desarrollo de productos de Ingeniería de Software, con especial hincapié en el uso de tecnologías de bases de datos.

- En este curso se estudian:
 - Conceptos de Ingeniería de Software,
 - Fundamentos de Bases de Datos Relacionales y Gestión de Transacciones
 - Gestión en Bases de datos para entornos centralizados concurrentes y distribuidos.
- También se presentan los avances en Bases de Datos Orientadas a Objetos y Datos no Estructurados.

Elementos de Bases de Datos

Competencias esperadas

- ✓ *discernir sobre la oportunidad de incluir el uso de servicios de bases de datos* en la resolución de un problema,
- ✓ seleccionar *criterios para evaluar y comparar entre los distintos sistemas de bases de datos* de acuerdo a las necesidades de un problema,
- ✓ *comprender, analizar y diseñar modelos de datos*,
- ✓ *configurar y administrar motores de Bases de Datos*, de acuerdo a las exigencias de cada aplicación.

Preguntas?



Sistema de Manejo de Bases de Datos (SMBD) –

Database Management System (DBMS)



Conceptos Introductorios

- SMBD – DBMS – Sistema de Base de datos.
- Base de Datos
- Aplicación
- Algo de historia (*file system vs database* y *lectura EBD_01_2015_Molina,Ullman - Database Systems.pdf*)
- Servicios del SMBD: modelo de datos, tareas de administración de datos, seguridad, integridad, transacciones (transparencias 31 a 46).
 - Modelo de datos
 - Lenguaje de Bases de Datos: consulta o manipulación y definición
 - Instancia y Esquema
- Arquitecturas:
 - de la aplicación y
 - del DBMS

Cómo vamos a estudiar en EBD

- **Modelado de la Base de Datos Relacional de un problema**
 - Modelos de datos para bases de datos: Entidad/Relación y Relacional
- **Implementación del software Sistema de Bases de Datos**
 - Manejo físico de archivos
 - Acceso a los datos
 - Manejo de Transacciones

¿Cómo Seguimos?

Diseño del modelo de datos de un problema: vista conceptual y lógica.

Nivel
Conceptual

Modelo
Entidad
Relación

Nivel
Lógico

Modelo
Relacional

Base de Datos

BASE DE DATOS (BD o DB) – es *una colección organizada de elementos de datos interrelacionados de un negocio.*

- Representa a una parte de un dominio o universo.
- La base de datos puede estar **centralizada** en una computadora o **distribuida** sobre varias, | separadas geográficamente.



Base de datos

- (1) Una base de datos es una colección de datos o información *persistente*.
- (2) Una base de datos modela parte del mundo real, llamado la *empresa o negocio*.
- (3) Una base de datos generalmente es un *recurso compartido* de interés para un grupo de usuarios.
- (4) Una base de datos es, en general, administra un *volumen datos importante*
- (5) Existen una o más *aplicaciones* vinculadas a la base de datos.

Sistema de Gestión de Bases de Datos (DBMS)

Sistema de Manejo (o Gestión) de Bases de Datos (SMBD o DBMS): es una colección interrelacionada de *programas* preparados para administrar datos.

- Un SMBD es un sistema de computación responsable del almacenamiento eficiente y de la obtención de los elementos de una BD. Para hacer la administración de los datos más conveniente, generalmente se encapsula a las bases de datos dentro de un DBMS.
- **Otras denominaciones:** sistema de base de datos (database system)

Sistema de Base de Datos

Sistema de Base de Datos: recibe este nombre la combinación del DBMS + datos.

- Ejemplos en la Uns:
 - Sistema de Base de Datos Académico SIU-Guaraní
 - Sistema de Base de Datos de la Biblioteca.
 - ...

Algunas aplicaciones que usan SMBD

- El uso servidores de bases de datos está ampliamente difundido. Aplicaciones típicas:
 - *Bancarias* (clientes, cuentas, préstamos y transacciones).
 - *Aerolíneas* (reservas y planificaciones). Fueron las primeras en usar bases de datos distribuidas.
 - *Académicas* (alumnos, graduados y registros académicos).
 - *Transacciones para tarjetas de créditos.*
 - *Telecomunicaciones* (registro de llamadas, facturación, redes de comunicación).
 - *Comercio electrónico.*
 - Etc...



¿Antes de los SMBD?

- Antes de la existencia de SMBD's se usaban *sistemas de procesamiento de archivos*, los cuales tenían las siguientes desventajas y problemas:
 - Redundancia e inconsistencia de los datos.
 - Dificultad en el acceso de los datos.
 - Problemas de integridad.
 - Problemas de atomicidad.
 - Anomalías en accesos concurrentes.
 - Problemas de seguridad.

DBMS vs. File-System

Redundancia e inconsistencia de datos

- Aplicaciones desarrolladas en distintos lenguajes de programación graban información en **archivos con diferentes estructuras**.
- Los **datos se repiten**. Aumenta el costo en espacio y las posibilidades de inconsistencia.

Dificultad para acceder a los datos

- Un nuevo requerimiento no considerado requiere del **desarrollo de un nuevo programa**.
- La información está distribuida.

DBMS vs. File-System

Aislamiento de datos

- Datos distribuidos en archivos distintos, con formatos diferentes **oscurece los programas de aplicación.**

Problemas de integridad

- La confiabilidad de la información depende de que los datos satisfagan ciertas **restricciones de consistencias.**
- Ejemplos:
 - Sólo se puede tener registro de exámenes finales de alumnos o ex-alumnos.
 - Las calificaciones de los exámenes finales son números enteros del 0 al 10.

DBMS vs. File-System

Problemas de Atomicidad

- Los sistemas están sujetos a fallos. Para algunas aplicaciones es crucial que si se produce un fallo, al recuperar el sistema, los datos vuelvan al estado consistente inmediatamente anterior al fallo.
 - Ejemplo: un fallo en la mitad de una transferencia de fondos de una cuenta a otra.
- En el ejemplo anterior se dice que la operación de transferencia debe ser atómica.

DBMS vs. File-System

Anomalías de acceso concurrente

- Para mejorar la performance y tiempo de respuesta, algunos sistemas permiten que **múltiples usuarios actualicen el contenido** de datos en forma simultánea.
- La interacción **de actualización concurrente** puede resultar en datos inconsistentes.
 - Ejemplo: dos terminales operando sobre la misma cuenta.

DBMS vs. File-System

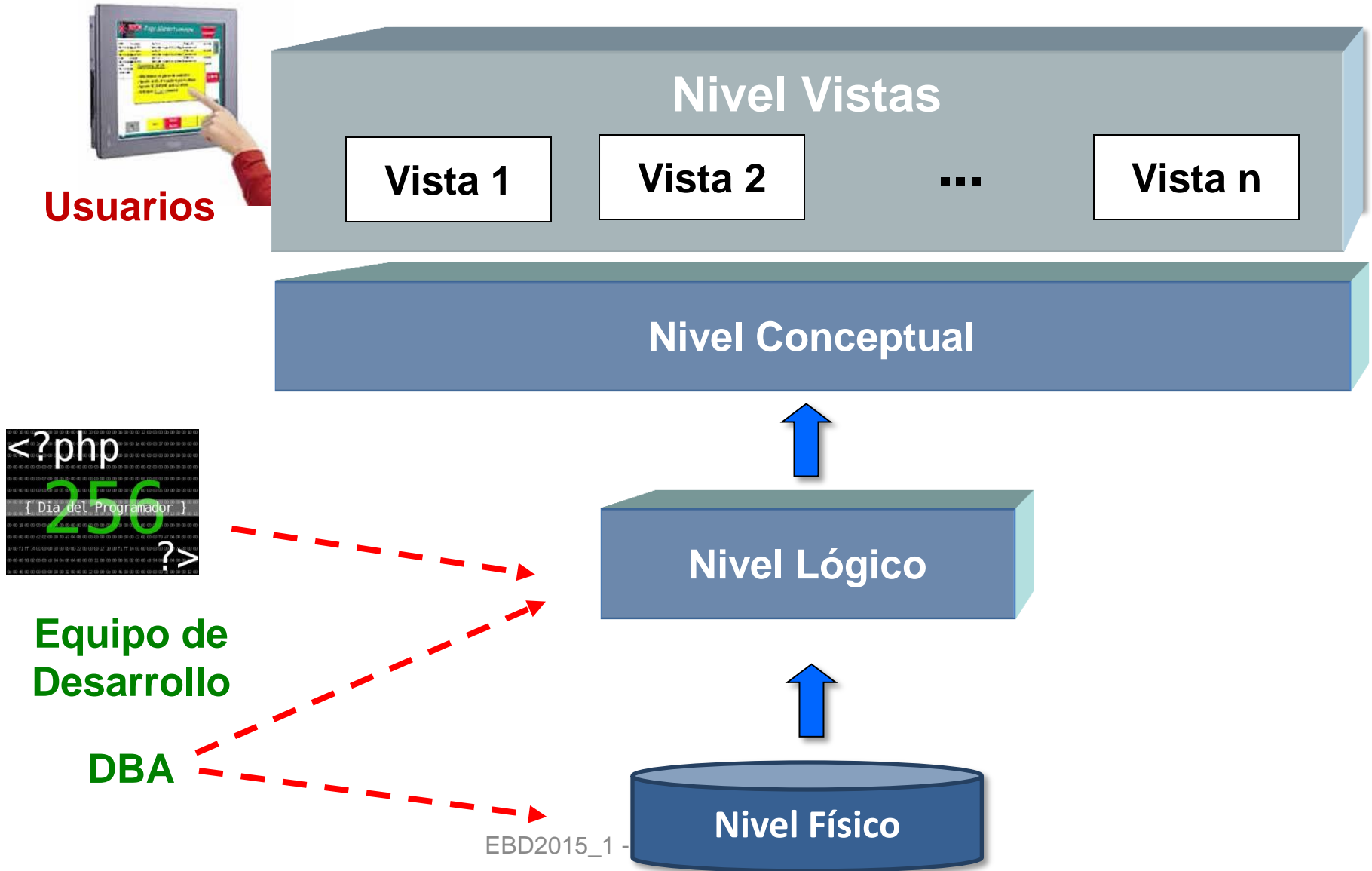
Seguridad

- No todos los usuarios deben tener **acceso a todos los datos**.
- Ejemplo:
 - Se espera que un sistema permita acceso a diferente información almacenada a los empleados que trabajan en la sección sueldos que a los vendedores.

Visión de los datos

- *Un DBMS es una colección interrelacionada de **datos** y un conjunto de **programas que permite acceder y actualizar los datos**.*
- Uno de los propósitos del SMBD está ofrecer a los usuarios una **visión abstracta de los datos** (ocultando cómo se almacenan y mantienen).
- Se definen diferentes niveles de abstracción:
 - **Nivel Físico** (interno, de menor nivel de abstracción)
 - **Nivel Lógico** (o de desarrollo, nivel de abstracción intermedia)
 - **Nivel Conceptual y de Vistas**

Niveles de abstracción de datos



Niveles de Abstracción

Nivel Conceptual y de Vista

- Más abstracto.
- Describe “partes” de la base de datos.
- Un sistema puede proveer varias vistas de la misma base de datos.

Nivel Lógico

- Abstracción media
- Describe *qué datos se almacenan y cómo se relacionan en término de un conjunto reducido de estructuras simples.*
- Oculta la representación física de los datos.

Nivel Físico

- Es el nivel de menor abstracción.
- Define cómo los datos son efectivamente almacenados.

Definiciones: Instancias y Esquemas

- El contenido de la base de datos cambia en el tiempo en la medida que se agrega, borra o modifica información.

Instancia – corresponde con el conjunto de datos almacenados en un instante particular.

- *Ejemplo*: datos sobre libros sobre los libros que hay en la biblioteca de la UNS.

Esquema – representa el diseño de la base de datos. Los esquemas raramente se modifican.

- *Ejemplo*: de los libros se guardan datos sobre ISBN, título, autores, área de aplicación.

Esquemas – Niveles de Abstracción

Esquemas

- **Esquema Físico**: describe el diseño de la base de datos a nivel físico
- **Esquema Lógico**: describe el diseño de la base de datos a nivel lógico.
- Varios esquemas (denominados **sub-esquemas**) a nivel de vista.
- Las aplicaciones se construyen sobre el esquema lógico.

¿Por qué usar un SMBD?

- Un SMBD es un paquete de software que maneja las interacciones de una aplicación y la base de datos, proporcionando un entorno *práctico* y *eficiente*:
 - Ahorra desarrollos al programador.
 - Provee independencia de los datos.
 - Provee varias herramientas.

Independencia de datos

Independencia de datos – es la facilidad de modificar la definición del esquema en un nivel sin afectar su respectiva definición en el nivel superior inmediato.

- **Independencia de datos física:** es la habilidad de modificar el esquema físico (buscando mejorar la performance) sin que los programas de aplicación deban ser reescritos (+ fácil de lograr).
- **Independencia de datos lógica:** es la habilidad de modificar el esquema lógico (cuando se altera la estructura lógica de la BD) sin que los programas de aplicación deban ser reescritos (+ difícil de garantizar).

Servicios del SMBD

1. Soporte de al menos un *modelo de datos*.
2. Disponibilidad de *Lenguajes de alto nivel* para administrar o manipular la base de datos
 - *Lenguaje de Manipulación de Datos* (LMD/DML).
 - *Lenguaje de Definición de Datos* (LDD/DDDL).
3. *Eficiencia* al consultar los datos almacenados.
4. Provea *manejo de transacciones* (commit y rollback).
 - Mantener *integridad* y *consistencia* de datos.
 - Provea *control de concurrencia* y capacidades para *compartir datos*.
 - Permita *recuperaciones de fallos*.
5. Brinde *seguridad* y facilidades en la *administración de datos*.

1. Modelos de Datos

Detrás de la estructura de una base de datos está el *modelo de datos*, es decir, una colección de herramientas para:

- Describir los datos.
- Describir relaciones entre los datos.
- Describir la semántica de los datos.
- Describir restricciones de integridad.

Modelos de Datos – Modelo Relacional

Modelo Relacional:

- *Usa una colección de tablas* para representar datos y relaciones.
- Cada tabla está formada por columnas distinguidas por su nombre.
- *Es un modelo orientado a registro.*
- Es el modelo de datos más usado a nivel comercial.

| Nombre | Documento | Dirección | Ciudad | Nro.Cuenta |
|-----------|-----------|-----------------|---------------------|------------|
| Aguirre | 12345678 | San Martin 32 | Bahía Blanca | A-1111 |
| Racciatti | 22222222 | Belgrano 15 | Tres Arroyos | B-2222 |
| Sosa | 32324545 | Rivadavia 122 | Pigüe | C-3333 |
| Montero | 12127777 | Rosas 102 | Carmen de Patagones | D-4444 |
| Aguirre | 12345678 | San Martin 32 | Bahía Blanca | A-2244 |
| Maciel | 30012367 | 9 de Julio 1816 | Punta Alta | E-5555 |
| Echagüe | 54120121 | 25 de Mayo 1810 | Coronel Pringles | F-6666 |
| Racciatti | 22222222 | Belgrano 15 | Tres Arroyos | A-2244 |

Base de datos Relacional

| Nro.Cuenta | Saldo |
|------------|-------|
| A-1111 | 500 |
| B-2222 | 700 |
| C-3333 | 4200 |
| D-4444 | 350 |
| A-2244 | 900 |
| E-5555 | 750 |
| F-6666 | 1000 |

Cientes

Cuentas

Modelos de Datos – Modelo Entidad/Relación

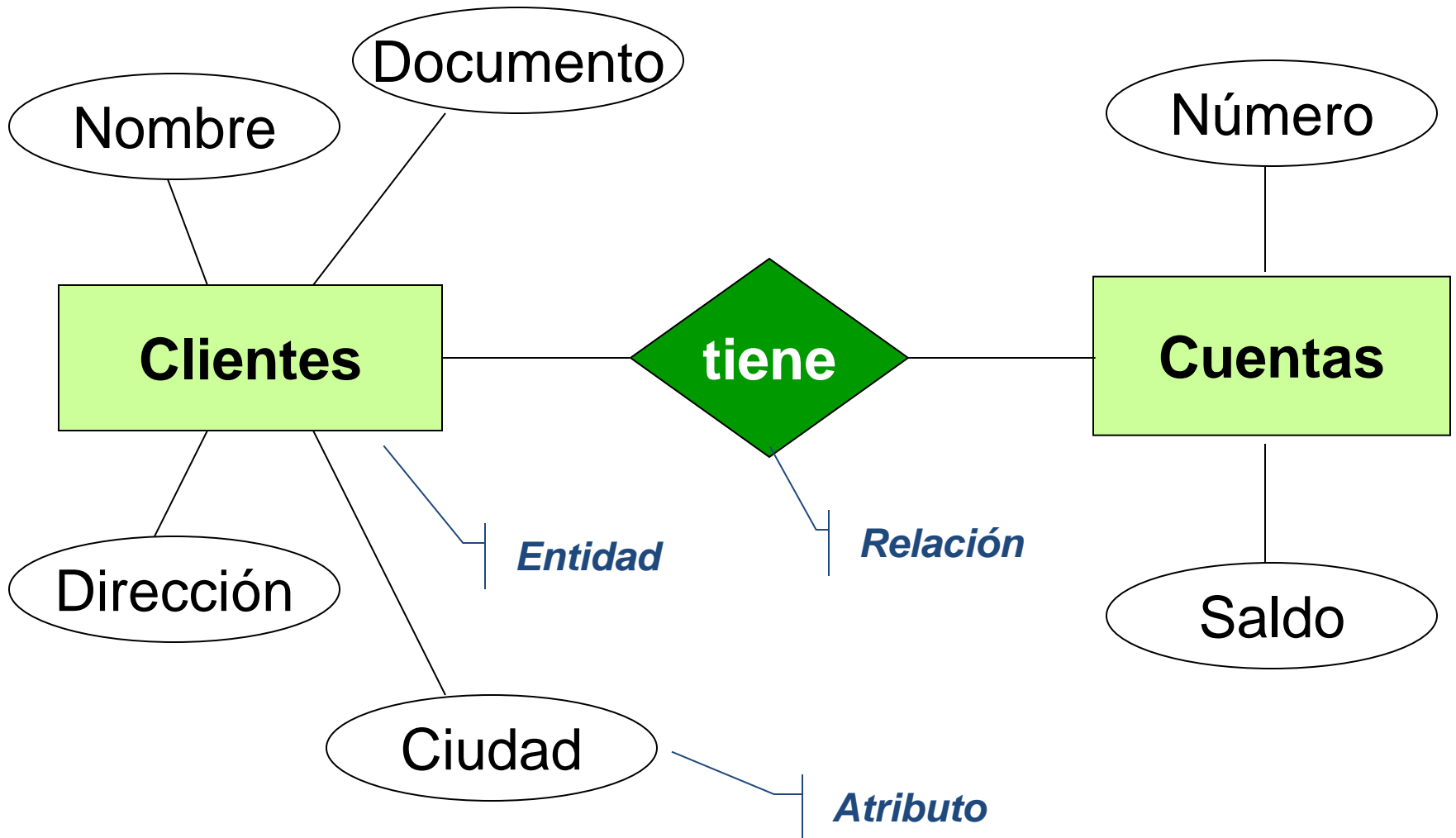
Modelo Entidad-Relación (ER):

- Está basado en una percepción del mundo real que consiste de objetos básico o *entidades* y sus *relaciones*.
- Una entidad es un objeto o cosa distinguible de otras.
- Cada entidad tiene asociado un conjunto de *atributos* que la describe
- El Modelo ER se usa para el diseño de bases de datos a nivel conceptual.

Modelo Entidad-Relación y Diagrama Entidad-Relación

- La estructura de una base de datos se suele expresar gráficamente mediante un *diagrama entidad-relación*.
- Un **Diagrama Entidad-Relación (DER)** está compuesto de:
 - **Rectángulos**: representando conjuntos de entidades.
 - **Elipses**: representando atributos.
 - **Rombos**: representando relaciones entre conjuntos de entidades.
 - **Líneas**: vinculando conjuntos de entidades entre si o conjuntos de entidades con relaciones.

Diagrama Entidad-Relación



Modelos de Datos – Modelo Orientado a Objetos

Modelo de datos orientados a objetos:

- Está creciendo en importancia.
- Se puede pensar como *una extensión del Modelo E-R con conceptos de encapsulación, métodos e identidad de objetos.*
- El *modelo Orientado a Objetos relacional* combina los modelos de datos orientados a objetos y relacionales.

Modelos de Datos – Modelo Semiestructurado

Modelos de Datos Semi-Estructurados

- Permite especificar datos donde ítems individuales del mismo tipo que difieren en sus atributos.
- Los datos semiestructurados se definen usando alguna variante de XML (Extensible Markup Lenguaje).
- Las capacidades para especificar tags y crear estructuras anidadas hace posible manejar también **datos**.
- Existe una variedad de herramientas disponibles para chequear, mostrar y consultar datos XML.

2. Lenguajes de Bases de Datos

Los SMBD proveen dos tipos de lenguajes:

- **Lenguaje de Definición de Datos (LDD/DDDL):** es el lenguaje que permite especificar mediante un conjunto de sentencias *el esquema de una base de datos*.
- **Lenguaje de Manipulación de Datos (LMD/DML):** es el lenguaje que nos permite *modificar y consultar la información almacenada* en una base de datos, ie. manipular instancias.

Lenguaje de Manipulación de Datos (DML)

Permiten realizar acciones como:

- *Recuperar* información almacenada en la BD.
- *Agregar* nueva información a la BD.
- *Borrar* información de la BD.
- *Modificar* información de la BD.

Los lenguajes pueden ser:

- *Procedimentales/Operacionales*: especifican *qué* datos se necesitan y *cómo* obtenerlos.
- *Declarativos/No Procedimentales*: especifican *qué* datos se necesitan sin especificar como obtenerlos.

Lenguaje de Definición de Datos (DDL)

- Sirven para especificar *el esquema de base de datos y otras propiedades*:
 - Estructurar el almacenamiento y métodos de acceso.
 - Definir restricciones de consistencia sobre los datos almacenados:
 - Restricciones de dominio
 - Restricciones de integridad referencial
 - Autorizaciones

4. Manejo de Transacciones

Se denomina **transacción** a la colección de operaciones que realiza una única función lógica en una aplicación de base de datos.

- *Ejemplo:* supongamos que una transacción T transfiere \$50 pesos de una cuenta A a una cuenta B.
 - Esto es, debita \$ 50 de A.
 - Acredita \$ 50 en B.

Entre los servicios que puede ofrecer un DBMS está la capacidad de manejar transacciones.

Manejo de Transacciones

- La transacción T debe realizarse de manera completa (debe ejecutarse con éxito el débito en A y el crédito en B). El requerimiento “todo-o-nada” (**atomicidad**).
- Después de que se ejecute con éxito la transacción T debe preservarse la consistencia de la BD, esto es, la suma de A+B debe ser igual (**consistencia**).
- Después de la ejecución exitosa de una transferencia, los nuevos valores de las cuentas A y B deben persistir, a pesar de la posibilidad de fallas en el sistema (**durabilidad**).
- Cada transacción es una unidad de atomicidad y consistencia que no debe violar las restricciones de una base de datos (**aislamiento**).

Manejo de Transacciones

- Asegurar la *consistencia* de la base de datos es responsabilidad del **programador**.
- Asegurar las propiedades de *atomicidad* y *durabilidad* es responsabilidad del SMBD, particularmente del **Gestor de Transacciones**.
- Cuando varias transacciones actualizan en forma concurrente la base de datos, la consistencia no puede dejar de ser preservada, de igual modo que si se ejecutaran de a una. Es responsabilidad del **Gestor de Control de Concurrency** controlar la interacción de las transacciones concurrentes con el fin de preservar *consistencia* en la base de datos.

Sobre Transacciones

Datos interesantes

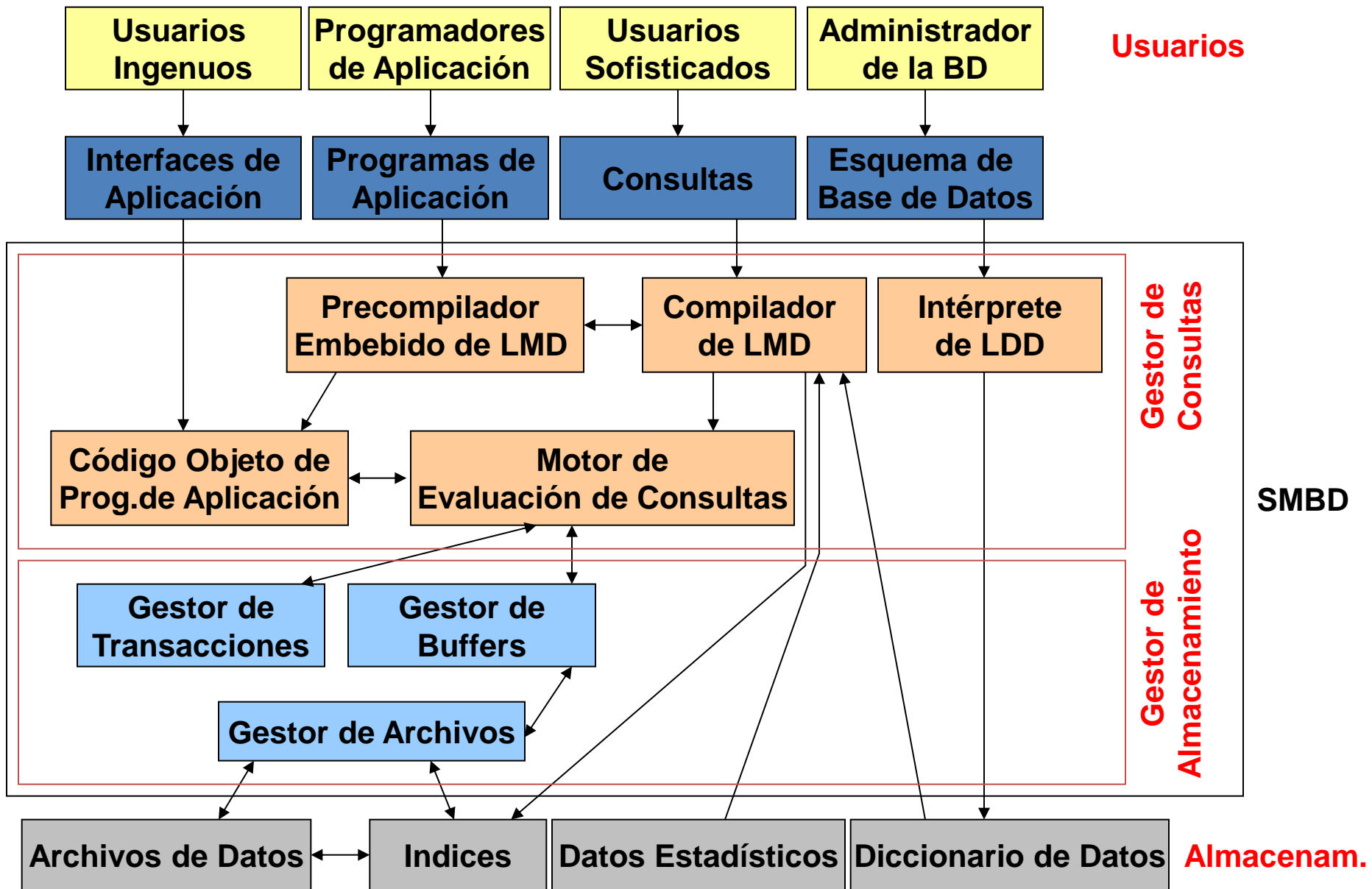
- Más de *20 millones de transacciones* comerciales usando tarjetas de crédito se procesan diariamente en más de *10 millones de negocios* para más de *20000 bancos*, involucrando **billones de dólares**.
- La seguridad y disponibilidad de estas bases de datos, además de los requisitos de *correctitud* y *performance* de las transacciones que las acceden, es crítica en el negocio de tarjetas de créditos

Data Mining y Análisis

- Con el término *data mining* se refiere a procesos semi-automáticos que analizan grandes volúmenes de datos buscando patrones: “knowledge discovery”.
- En general se requiere de la interacción de expertos del dominio para encontrar patrones útiles.

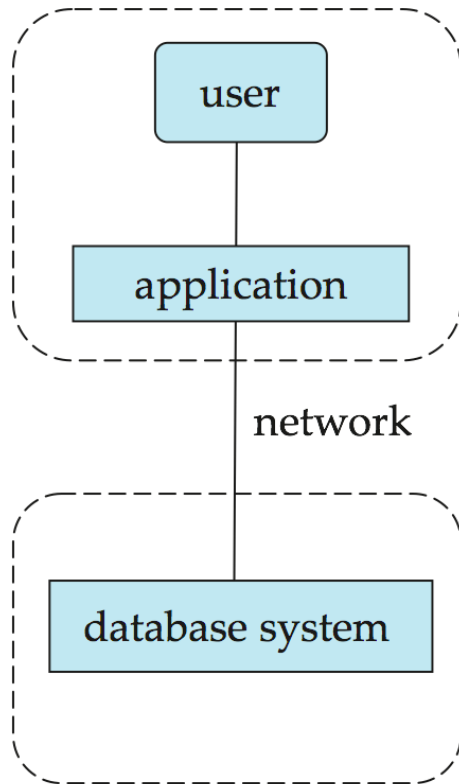
Arquitectura del SMBD

- Para proveer sus servicios un SMBD cuenta con varios componentes organizados en (figura transparencia 45)
 - Procesamientos de consultas
 - Administradores de almacenamiento.
- La **arquitectura de DBMS** está fuertemente influenciada por el **sistema físico** donde este corriendo: centralizado, cliente-servidor, arquitecturas paralelas, o distribuidas.
- A nivel de aplicaciones, generalmente los usuarios se conectan el SMBD a través de la red, diferenciándose la **máquina cliente-maquina servidor**.



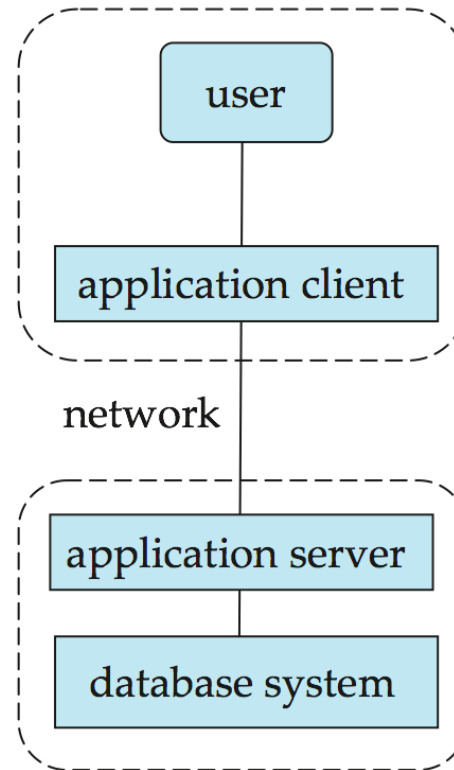
Arquitecturas

- Arquitecturas usadas en sistemas de bases de datos



(a) Two-tier architecture

client



(b) Three-tier architecture

server

Fuente: Database System Concepts – I. Silberschatz

Tipos de Usuarios de la BD

- Una meta primaria de un sistema de base de datos es proveer un entorno para obtener o almacenar información en la BD. Existen distintos tipos de usuarios:
- **Administrador de bases de datos.**
- **Programadores de Aplicación:** desarrolladores.
- **Usuarios Sofisticados:** interactúan con la base de datos usando lenguajes de consulta.
- **Usuarios Ingenuos:** quienes interactúan usando programas de aplicación permanentes.

Administrador de una Base de Datos

- El **administrador de base de datos (ABD/DBA)** es la persona encargada de configurar y administrar el SMBD. Sus principales funciones son:
 - Definir esquemas.
 - Definir estructuras de almacenamiento y métodos de acceso.
 - Modificar esquemas y organización física.
 - Conceder autorización para el acceso de los datos.
 - Especificar restricciones de integridad.
 - Asegurar la performance del sistema.

Temas de la clase de hoy

- Bases de Datos y Sistemas de Manejo de Bases de Datos (DBMS). Visión de los Datos
- Servicios de un DBMS.
 - Modelos de Datos
 - Lenguajes
 - Manejo de transacciones
- Arquitectura de un DBMS.
- **Bibliografía:**
 - *“DATABASE System Concepts”* – A. Silberschatz – Capítulo 1.
 - *“Principles of Database and Knowledge-Base Systems”* – J. Ullman. Capítulo 1.