

Administración de Bases de Datos

Licenciatura en Gestión de la Información

Miriam Myszne
mmyszne@gmail.com

Universidad Nacional de Lomas de Zamora – Facultad de Ingeniería



Asistencia y Evaluaciones

- ▶ Sábados de 9 a 13 hs. Asistencia 75%.
- ▶ Modalidad: teórico/práctica con uso de computadora para práctica de base de datos.
- ▶ Evaluaciones:
 - ▶ **1 parcial individual**. Se aprueban con nota 4 o superior.
 - ▶ **2 Trabajos prácticos grupales**: modelado de datos y Sql.
 - ▶ **1 Trabajo de investigación grupal** con exposición oral al curso (tema a definir).
- ▶ La materia es promocional con nota 7 o superior en el parcial, TPs y Trabajo de investigación/Monografía.
- ▶ **El parcial no aprobado se recupera en la última clase del cuatrimestre.**
- ▶ **El TP grupal no aprobado hay que rehacerlo y aprobarlo!**
- ▶ **Quienes no promocionen la materia, deben rendir examen final.**



Objetivos de la materia

- ▶ Conocer y/o llevar adelante una **Bases de Datos (BDD)** para que cualquier empresa u organización pueda **administrar correctamente su información**.
- ▶ Conocer las herramientas básicas necesarias para el diseño y desarrollo de BDD.
- ▶ Analizar y diseñar el **modelo de datos** que permita obtener la información requerida, transformarlo en un **modelo de datos relacional**.
- ▶ Conocer algunos elementos básicos de la estructura de un esquema de información SQL.
- ▶ Ver otras estrategias actuales de gestión de datos (No-Sql).

UNIDAD 1

Fundamentos de Bases de Datos

- ▶ 1.1 Datos e Información
- ▶ 1.2 Manejo de la Información
- ▶ 1.3 Características de la Información Útil
- ▶ 1.4 Sistemas de Información
- ▶ 1.5 Administración de Datos y de BDD
- ▶ 1.6 Persistencia de Datos
- ▶ 1.7 ¿Qué es un Sistema de Base de Datos?
- ▶ 1.8 Un poco de historia: Organización de Archivos
- ▶ 1.9 Inconvenientes en el uso de archivos
- ▶ 1.10 ¿Qué es una Base de Datos?
- ▶ 1.11 Ventajas en el Uso de Bases de Datos
- ▶ 1.12 Sistema Gestor de Base de Datos
- ▶ 1.13 Clasificación de los SGBD
- ▶ 1.14 Algunos SGBD actuales

UNIDAD 1

Fundamentos de Bases de Datos

- ▶ 1.15 Diseño de bases de datos
- ▶ 1.16 Arquitectura de las bases de datos

Introducción

- ▶ Las necesidades de información de nuestra sociedad actual exigen **información precisa, oportuna, coherente y adaptada a las necesidades específicas** de cada usuario y de cada circunstancia, para la investigación, la planificación y la **toma de decisiones**.
- ▶ Los tomadores de decisiones están comenzando a comprender que la información no es sólo un subproducto de la conducción, sino que a la vez alimenta a los negocios y puede ser el factor crítico para la determinación del éxito o fracaso de éstos.
- ▶ Las **bases de datos** se desarrollan como respuesta a estas necesidades de las organizaciones, buscando mejorar las prestaciones de los sistemas informáticos y aumentar su rendimiento en la **producción de información precisa**.



Un **sistema computacional** está formado por un conjunto de tecnologías capaz de efectuar cuatro operaciones:

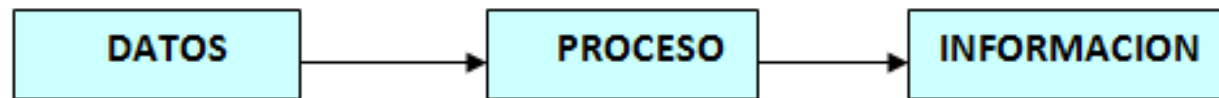
- ▶ **Aceptar entradas**: reciben **datos** del mundo exterior.
- ▶ **Procesar datos**: realizar operaciones aritméticas y lógicas para generar información.
- ▶ **Producir salidas**: dan información al mundo externo por diferentes medios: pantalla, impresión, mails, alertas, etc..
- ▶ **Almacenar información**: mueven y almacenan la información desde y hacia la **memoria de la computadora** y a los **dispositivos de almacenamiento**.

- ▶ Todo **sistema de cómputo** tiene componentes electrónicos y mecánicos.
- ▶ A estos componentes los conocemos como **Hardware** de la computadora.
- ▶ El hardware tiene gran capacidad de procesamiento de información, pero no puede efectuar ninguna operación si no se le dan instrucciones para hacerlo.
- ▶ A estas instrucciones las conocemos con el nombre de **Software**.

1.1 Datos e Información

3

Los términos “**datos**” e “**información**” no significan lo mismo.



En la figura anterior vemos tres etapas bien definidas en todo sistema:

- ▶ **Datos**: son los elementos básicos y fundamentales al inicio de una tarea a realizar, por sí solos, no poseen ningún valor agregado. **Los datos son la materia prima.**
- ▶ **Proceso**: es un conjunto **de acciones o tareas** que se realizan **en un orden específico** con un **principio** y un **fin** bien definidos.
- ▶ **Información**: es el resultado que se obtiene de **procesar los datos**, y fin primordial que se busca en todo sistema de información.

Como Recurso

- ▶ Para maximizar la utilidad de la información, un negocio la debe **manejar correctamente** tal como maneja los demás recursos.
- ▶ Los administradores necesitan comprender que hay **costos asociados** con la **producción, distribución, seguridad, almacenamiento y recuperación** de toda información.
- ▶ Aunque la **información** se encuentra a nuestro alrededor ésta **no es gratis**, y su uso es estratégico para posicionar la competitividad de un negocio.

Generada por Computadora

- ▶ La disponibilidad de computadoras/dispositivos (de todo tipo) ha creado una **explosión de información** a través de la sociedad en general y de los negocios en particular.
- ▶ Esto genera mayor cantidad de información de computadora a administrar. Es importante invertir en el costo de **organizarla y mantenerla!**
Ambas tareas son complejísimas!!!!
- ▶ La **información** generada debe ser **útil!**



1.3 Características de la Información Útil 1

La información debe ser:

- ▶ **Relevante:** la información debe relacionarse con el problema por resolver.

Por ejemplo, el total de años de estudio de una persona no es suficiente para saber si puede realizar un trabajo. La información relevante es que la persona ha estudiado una carrera acorde al puesto y que tiene años de experiencia laboral.

- ▶ **Completa:** la información parcial puede ser peor que la falta de información.

Por ejemplo, no puedo realizar una estadística de ventas comparativa anual, si faltan ventas de algún mes del año.

1.3 Características de la Información Útil 2

La información debe ser:

- ▶ **Precisa**: la información errónea o alterada puede conducir a decisiones desastrosas! **Validar y Controlar!**
- ▶ **Actual**: a menudo las decisiones se basan en la información más reciente disponible, lo que ayer fue realidad, hoy tal vez ya no lo sea. Por ejemplo, si las listas de precio de los productos que comercializo están en dólares y un cliente pide su valor en pesos, no puedo tomar la cotización del dólar de la semana pasada.
- ▶ **Económica**: en los negocios, el costo de obtener información es considerado un elemento de costo relacionado con cualquier decisión. Por ejemplo, es necesario investigar la demanda de un nuevo producto para reducir el riesgo del fracaso comercial, pero si la investigación de mercado es demasiado costosa, va a disminuir la ganancia por las ventas de ese producto.

1.4 Sistemas de Información

Un **Sistema de información** es un conjunto de componentes relacionados entre sí que recuperan, procesan, almacenan y distribuyen información para **apoyar la toma de decisiones** y ayudar al control en una organización.

Su objetivo primordial es obtener información certera y precisa en el menor tiempo posible.

Cada tipo de sistema de información de una empresa u organización tiene “**áreas funcionales**”.

En general las áreas funcionales incluyen:

- ▶ *Contabilidad*
- ▶ *Administración y Finanzas* (Cobranzas – Pagos - Inversiones)
- ▶ *Ventas* (de servicios y/o productos)
- ▶ *Marketing*
- ▶ *Recursos Humanos*
- ▶ Si fabrica productos para la venta:
Producción, Abastecimiento, Stock e Inventario

1.5 Administración de Datos y de BDD

1



El/La Administrador/a de Datos:

En una empresa u organización existe una persona identificable con la **responsabilidad central sobre los datos**. Es el **Administrador/a de Datos** (**DA**, Data Administrator).

El **DA** debe conocer la información y las necesidades de la empresa en un nivel gerencial superior.

El **DA** decide en primer término cuáles datos deben almacenarse en la base de datos y establecer políticas para mantenerlos y manejarlos.

Tiene acceso ilimitado a los datos!



1.5 Administración de Datos y de BDD

2



El/La Administrador/a de BDD:

El **DA** es un gerente, no un técnico.

El técnico responsable de poner en práctica las decisiones del **DA** es el **Administrador/a de Base de Datos** (**DBA**, Data Base Administrador).

El **DBA** es un profesional en **procesamiento de datos**. Su tarea es crear la base de datos y poner en vigor los controles técnicos necesarios para apoyar las políticas dictadas por **DA**.

El **DBA** se encarga de garantizar el funcionamiento adecuado del sistema y de proporcionar otros servicios de índole técnica, por ejemplo: gestión del **storage**, **permisos a usuarios**, políticas para **respaldar y recuperar la información** (backup/restore), etc.

1.5 Administración de Datos y de BDD

3



1.6 Persistencia de Datos

- ▶ **Persistencia de datos**: Los datos se **guardan en algún lugar**, por ejemplo, en **archivos**, en **bases de datos**, etc., que están en algún **dispositivo de almacenamiento**, por ejemplo, **disco**, **cinta**, **tarjetas de memoria**, **nube**, etc..

Y según cómo se organicen lógicamente estos datos, **el acceso** a los mismos se facilitará o dificultará.

Lo primero es lo mejor!

- ▶ **Archivo**: Antes de la existencia de las BDD, todo se almacenaba en archivos. Un archivo esta compuesto por **registros** (filas) y cada registro por **campos** (columnas).
- ▶ **Base de datos**: Es un conjunto de datos pertenecientes a un mismo contexto, organizados lógicamente según un modelo estructural y almacenados sistemáticamente para su uso posterior.

1.7 ¿Qué es un Sistema de Base de Datos?1

Es un sistema computarizado cuyo propósito es mantener información almacenada y hacer **que esté disponible** cuando se solicite. Sus componentes principales son:

- ▶ **Datos:** los datos son la BDD propiamente dicha.
- ▶ **Hardware:** formado por dispositivos de almacenamiento y dispositivos periféricos (procesador, canales de comunicación, etc..).
- ▶ **Software:** formado por un conjunto de programas llamado **Sistema Manejador de Base de Datos**. (**DBMS**: DataBase Management System).
Este sistema resuelve todas las solicitudes formuladas por los usuarios/aplicaciones a la base de datos.

1.7 ¿Qué es un Sistema de Base de Datos?2

Usuarios: existen tres clases de usuarios relacionados con una BDD:

- ▶ El **desarrollador de aplicaciones**, quien crea programas que utilizan la BDD.



- ▶ El **usuario final**, quien accesa la BDD por medio de un lenguaje de consulta (**SQL**) o de programas de aplicación.



- ▶ El **administrador de la Base de Datos (DBA)**, quien se encarga del control general del Sistema de BDD.



1.8 Un poco de historia: Organización de Archivos

1

La **organización de archivos** se refiere a cómo se almacenan y acceden físicamente a los registro del archivo.

En la historia de la informática existen tres formas de organizar y acceder a la información de archivos:

- ▶ **Archivos secuenciales**
- ▶ **Archivos indexados**
- ▶ **Archivos relativos** o **directos** (hoy día casi en desuso para el desarrollador, lo usan los mismos DBMS)

1.8 Un poco de historia: **Archivos Secuenciales**

2

Un archivo secuencial está formado por:

- ▶ una **secuencia de registros**,
- ▶ con marcas de **principio (BOF)** y **fin de archivo (EOF)**.

Todos los registros tienen **sucesor** (salvo el último) y un **predecesor** (salvo el primero).

Tanto la lectura de un registro como la escritura, se hacen de a **uno por vez** y un registro detrás de otro.

Pueden estar ordenados por algún campo o no.

Los registros pueden ser de longitud fija o no.

El orden físico se corresponde con el orden lógico.

1.8 Un poco de historia:

Archivos Secuenciales

3

BOF		
Alvarez	555-8763	Munro
Castro	876-7645	Flores
....
Muñoz	...	Caballito
Perez	...	Lanús
....
Zapiola	...	Belgrano
EOF		

Ejemplo: un archivo secuencial con datos de contactos, **ordenado por el campo apellido ascendente**.

Cada contacto forma un registro.

Los datos del contacto Perez están después de los datos de Alvarez, Castro, Muñoz...

Para acceder a Perez se abre el archivo para lectura y se leen **desde el principio** los registros anteriores hasta llegar al de Perez.

1.8 Un poco de historia: **Archivos Secuenciales**

4

BOF		
....
Castro	876-7645	Flores
Zapiola	...	Belgrano
....
Perez	...	Lanús
Alvarez	555-8763	Munro

BOF		
Alvarez	555-8763	Munro
Castro	876-7645	Flores
....
Muñoz	...	Caballito
Perez	...	Lanús
....
Zapiola	...	Belgrano

EOF

¿Qué ocurre si el archivo secuencial estuviera **desordenado** y quiero buscar a Perez?

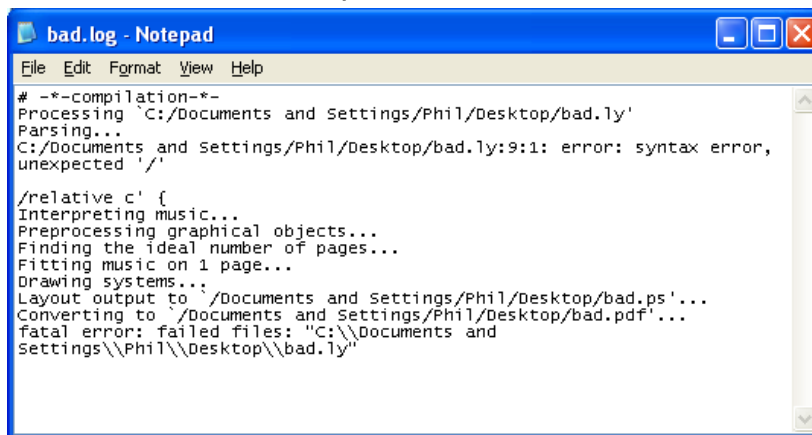
¿Qué ocurre si el archivo secuencial está **ordenado por apellido** y quiero mostrar los contactos de una zona específica?

1.8 Un poco de historia: Archivos Secuenciales

5

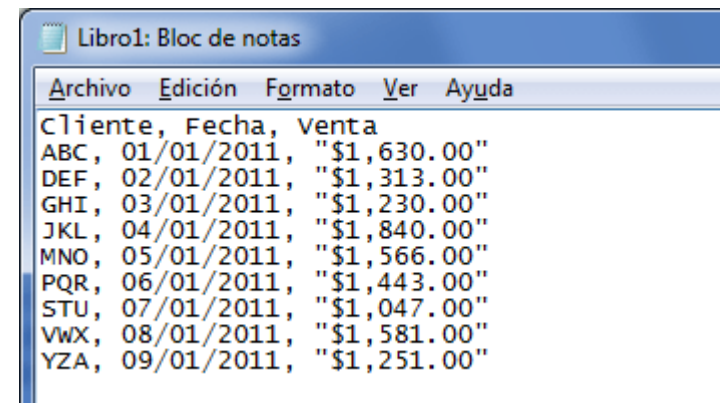
Otros ejemplos de archivos secuenciales:

- ▶ archivo de texto con información de log y **registros de longitud variable** (bitácora)
- ▶ archivos que generan los sistemas para subir a AFIP y/o ARBA con datos de IVA Ventas/Compras de una empresa, retenciones de IIBB de todo el país, con **registros de longitud fija** separados por: comas, tabs, pipe, etc., o sin separadores.
- ▶ archivo Word, archivo XML o JSON, etc..



```
# -*-compilation-*-
Processing 'C:/Documents and Settings/Phil/Desktop/bad.ly'
Parsing...
C:/Documents and Settings/Phil/Desktop/bad.ly:9:1: error: syntax error,
unexpected '/'

/relative c' {
Interpreting music...
Preprocessing graphical objects...
Finding the ideal number of pages...
Fitting music on 1 page...
Drawing systems...
Layout output to /Documents and Settings/Phil/Desktop/bad.ps'...
Converting to /Documents and Settings/Phil/Desktop/bad.pdf'...
fatal error: failed files: "C:\\Documents and
Settings\\Phil\\Desktop\\bad.ly"
```



Cliente, Fecha, Venta		
ABC,	01/01/2011,	"\$1,630.00"
DEF,	02/01/2011,	"\$1,313.00"
GHI,	03/01/2011,	"\$1,230.00"
JKL,	04/01/2011,	"\$1,840.00"
MNO,	05/01/2011,	"\$1,566.00"
PQR,	06/01/2011,	"\$1,443.00"
STU,	07/01/2011,	"\$1,047.00"
VWX,	08/01/2011,	"\$1,581.00"
YZA,	09/01/2011,	"\$1,251.00"

1.8 Un poco de historia:

Archivos Secuenciales Indexados

6

Para evitar el problema que se generaba con los archivos secuenciales, **se creó** una estructura que anexa al archivo de datos, **un índice** según un orden o criterio predefinido, con lo cual se agiliza el acceso a los datos.

El orden físico del índice se corresponde con el orden lógico.

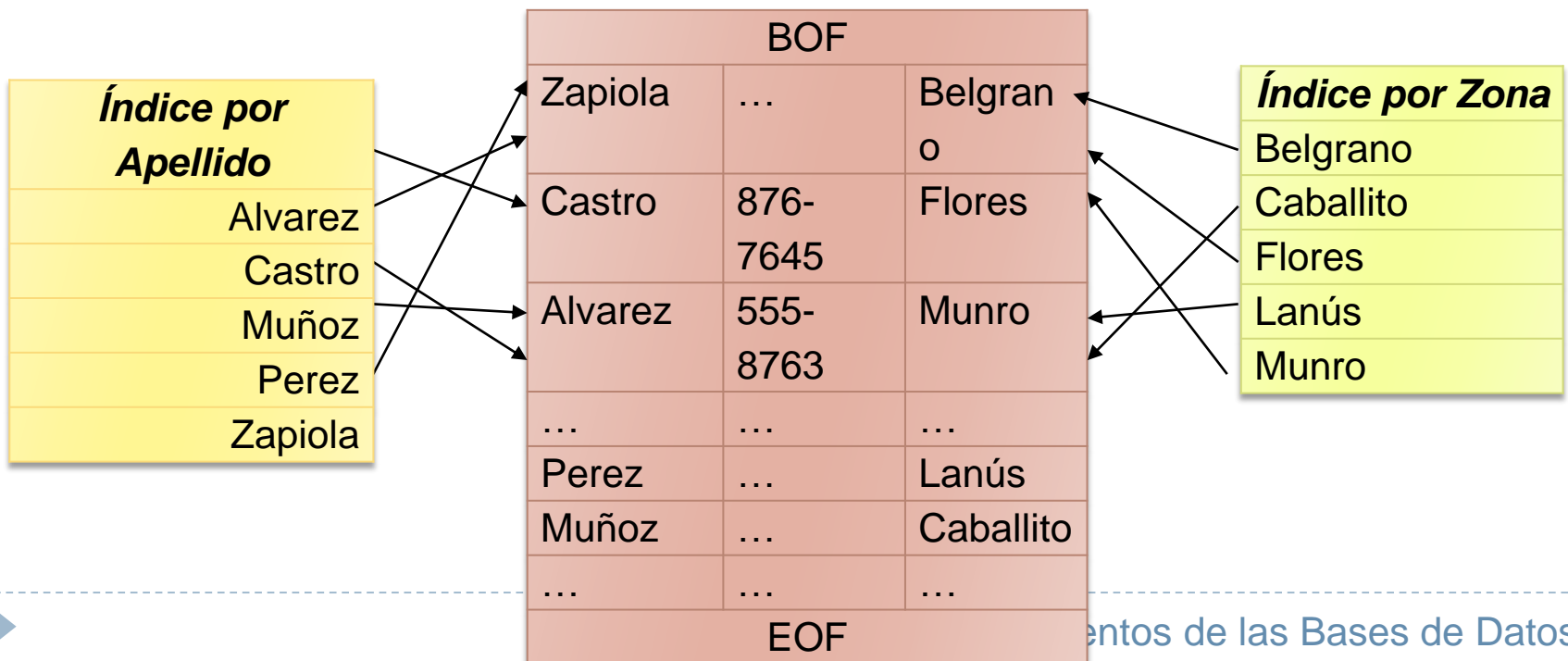
Un ejemplo de esto son los **índices de los libros**. Se puede leer un libro página por página (**en forma secuencial**) o si deseamos acceder a un tema específico del libro, **buscamos en el índice el nro. de página** donde se encuentra y accedemos al tema.

1.8 Un poco de historia:

Archivos Secuenciales Indexados

7

Volviendo al ejemplo del archivo de contactos, tendríamos los **datos sin importar el orden** y **dos índices**, uno por apellido y otro por zona. Se accede secuencialmente al índice para encontrar el registro.



1.8 Un poco de historia:

Archivos Relativos o Directos

8

En este tipo de organización el **orden físico de los registros no necesariamente se corresponde con el orden lógico.**

Existe una **relación** predecible **entre la clave para identificar un registro y la localización del registro dentro del archivo.**

Se puede acceder a cada uno de los registros sin necesidad de leer los que le preceden. **Se usa una fórmula** (algoritmo) formada por la clave del registro a buscar y la longitud de cada registro y se obtiene su ubicación física.

1.8 Un poco de historia: **Archivos Relativos o Directos**

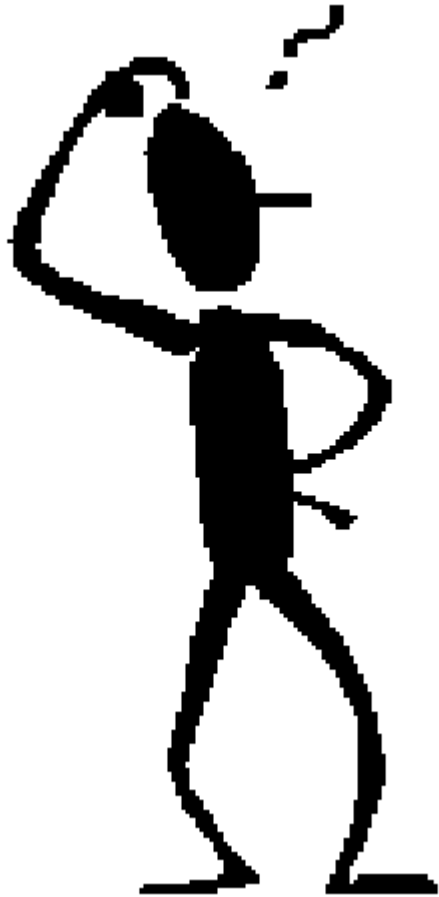
9

En el ejemplo, se busca el registro de la clave **Cristina**.
La clave ingresa como entrada al algoritmo y la salida es la ubicación física del registro de Cristina, el nro. **3**.



1.8 Un poco de historia: Organización de Archivos

10



Pregunta para los alumnos:

Bajo las mismas circunstancias de datos a almacenar, por ejemplo la agenda antes mencionada pero almacenada en tres archivos: uno **secuencial**, otro **secuencial indexado** y otro **relativo**.

¿Qué organización de archivos considera será la de **mejor performance** de acceso para recuperar datos?

1.9 Inconvenientes en el uso de archivos 1

Los inconvenientes en el uso de archivos para la persistencia de datos pueden ser:

- ▶ Redundancia de datos
- ▶ Problemas de consistencia e integridad de datos
- ▶ Dificultad para modificar su estructura lógica
- ▶ Aislamiento de datos

1.9 Inconvenientes en el uso de archivos

Redundancia de datos

2

Por lo general, para las aplicaciones que persisten sus datos en archivos, cada aplicación o programa está asociada a un archivo de datos.

Supongamos una empresa con un área de RRHH y otra de Ventas. El área de RRHH tiene un programa que mantiene datos de sus empleados en un archivo, y el registro de cada uno tiene:

Numero Legajo	Nombres	Apellido	DNI	Dirección	Teléfono	Email	Fecha Nacimiento	Fecha Ingreso
---------------	---------	----------	-----	-----------	----------	-------	------------------	---------------

1.9 Inconvenientes en el uso de archivos

Redundancia de datos

3

El área de Ventas tiene otro programa y un archivo propio con los datos de las ventas realizadas por cada empleado, y el registro de cada venta tiene:

Legajo Empleado	Nombre Empleado	Apellido Empleado	Número Venta	Importe Venta
--------------------	--------------------	----------------------	-----------------	------------------

Tanto el área de RRHH como la de Ventas, deben cargar y mantener sus archivos. El nro. de legajo, nombre y apellido de los empleados estará cargado **dos veces**.

Esto es redundancia de datos!

Numero Legajo	Nombres	Apellido	DNI	Dirección	Teléfono	Email	Fecha Nacimiento	Fecha Ingreso
------------------	---------	----------	-----	-----------	----------	-------	---------------------	------------------

1.9 Inconvenientes en el uso de archivos

Consistencia e integridad de datos 4

Supongamos que una empleada de la empresa se casó y **actualizó su apellido por el apellido de casada**.

Si el área de RRHH realiza la actualización del dato y el área de Ventas no lo hace, la información de la empleada quedará **inconsistente** y como consecuencia se pierde la **integridad de datos**, ya que para **el mismo legajo tendríamos dos personas diferentes**.

1.9 Inconvenientes en el uso de archivos

Modificar la estructura lógica

5

Supongamos que el área de Ventas quisiera registrar también la **fecha de la venta**, pues surgió la necesidad de relevar la cantidad de ventas que realiza cada empleado por mes.

Esto implica **modificar la estructura del registro** del archivo de ventas por:

Legajo Empleado	Nombre Empleado	Apellido Empleado	Número Venta	Importe Venta	Fecha Venta
--------------------	--------------------	----------------------	-----------------	------------------	------------------------

Habría que modificar todo el código del programa para que aceptara el nuevo diseño de registro.

1.9 Inconvenientes en el uso de archivos

Aislamiento de Datos

6

Los datos pueden estar en **varios archivos con distintos formatos**, que complican la creación de programas nuevos.



1.10 ¿Qué es una Base de Datos? 1

Una **BDD** es el lugar donde se guardan los “**datos en reposo**” y al que acceden las diferentes aplicaciones (sistemas/ programas) de una empresa/organización, a través de diferentes tipos de dispositivos.



1.10 ¿Qué es una Base de Datos? 2

Volviendo al ejemplo anterior, de las áreas de Ventas y RRHH, toda la información estaría guardada en una base de datos que contendría **dos tablas**:

<i>Empleados</i>
NumeroLegajo
Nombres
Apellido
DNI
Dirección
Teléfono
Email
FechaNacimiento
FechaIngreso

<i>Ventas</i>
LegajoEmpleado
NúmeroVenta
ImporteVenta
FechaVenta

Ya no tenemos el nombre y el apellido del empleado, duplicado. Sólo está en tabla Empleados.

1.11 Ventajas en el Uso de Bases de Datos 1

Describimos las principales ventajas del uso de BDD:

- ▶ **Normalizar** los datos de la BDD
- ▶ **Evitar** la **redundancia** de datos
- ▶ **Evitar** la **inconsistencia** de datos
- ▶ **Garantizar** la **integridad** de los datos
- ▶ **Garantizar** la **seguridad** de los datos
- ▶ **Compartir** los datos
- ▶ Facilidad de modificar los datos
- ▶ Globalización de la información

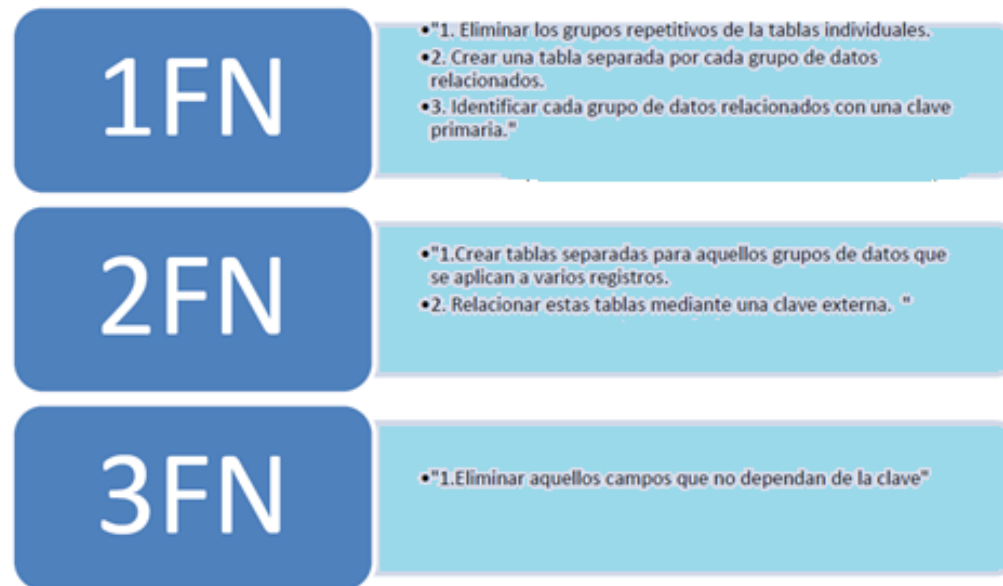
1.11 Ventajas en el Uso de Bases de Datos

Normalizar los datos

2

La normalización es una metodología que permite **minimizar la redundancia de datos**, y agiliza y garantiza la actualización de los mismos.

Lo veremos en detalle más adelante!



1.11 Ventajas en el Uso de Bases de Datos

Evitar la redundancia de datos

3

Se trata de **evitar al máximo la duplicidad de la información** guardada en las tablas.

Si quisiéramos saber el apellido de quién hizo la venta nro. N, estableceríamos **relaciones** (puentes) entre las tablas para obtener información relacionada.



La redundancia “cero” en una
no existe!

1.11 Ventajas en el Uso de Bases de Datos

Evitar la inconsistencia de datos

4

Las BDD usan **transacciones** para garantizar la inexistencia de inconsistencias de datos.

Una **transacción es una unidad básica de trabajo**, compuesta por un conjunto de tareas a realizar en forma **indivisible o atómica**.

El ejemplo habitual es la **transferencia bancaria**, traspaso de dinero entre cuenta origen y cuenta destino. Son dos operaciones distintas, en una se decrementa el saldo de la cuenta origen y en otra se incrementa el saldo de la cuenta destino.

1.11 Ventajas en el Uso de Bases de Datos

Evitar la inconsistencia de datos

5



¿Qué ocurriría si luego del decremento del saldo en la cuenta origen se cae el sistema?

Para garantizar la **consistencia de datos**, ambas operaciones deben ser atómicas, el sistema debe garantizar que el resultado final sea: se realizaron bien las dos operaciones o no se realizó ninguna de ellas.

1.11 Ventajas en el Uso de Bases de Datos

Integridad y Seguridad de los datos

6

Como se evitan la redundancia e inconsistencia de datos y se usan transacciones, la **integridad** está garantizada. Integridad significa que la información obtenida de la BDD **sea correcta en todo momento**.

El acceso a la BDD para los usuarios y las aplicaciones, se hacen por medio de permisos. Si los permisos están bien definidos, **nadie ingresará donde no debe!**

Los permisos los define el **DA** y los administra el **DBA**. Se pueden establecer permisos para cada tipo de acceso: consulta, modificación, eliminación, etc., a cada elemento de la base de datos.

1.11 Ventajas en el Uso de Bases de Datos

Compartir los datos/Modificar los datos⁷

Dado que los datos de una BDD se almacenan en un mismo lugar lógico, se pueden **compartir** sin problema entre usuarios y aplicaciones que los acceden. Se aclara que la implementación física de una BDD puede darse en 1 a N archivos físicos, pero esto es transparente a usuarios y aplicaciones que usan las BDD.

Como se evita la redundancia de datos y se guardan en un único lugar, **es más fácil realizar modificación de datos**. Se pueden hacer cambios en la estructura de la BDD sin hacer cambios en las aplicaciones o programas que la usan.

1.11 Ventajas en el Uso de Bases de Datos

Globalización de la información 8

Permite a los diferentes usuarios considerar la información como un **recurso corporativo** que carece de dueños específicos.



1.12 Sistema Gestor de Base de Datos

1

Un SGBD o DBMS (Database Management System) es una colección de datos más un conjunto de programas que permiten a los usuarios accederlos y modificarlos.

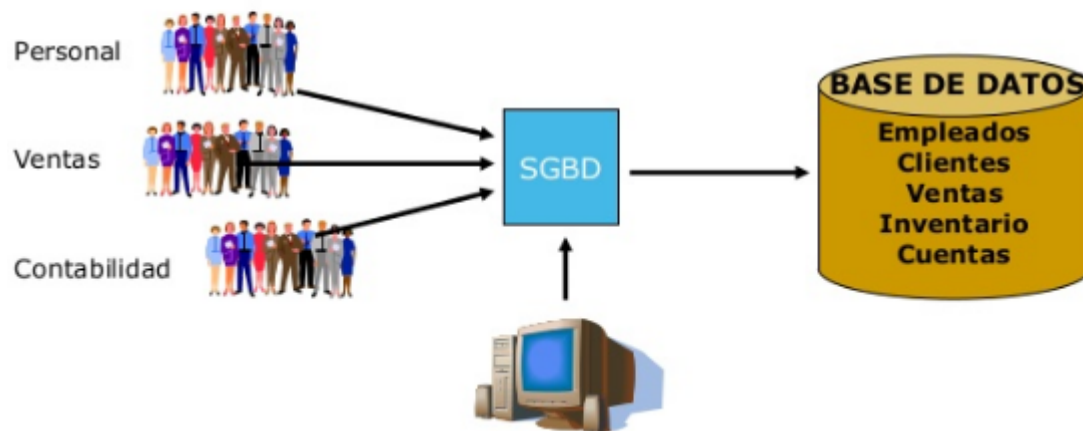
Un SGBD proporciona a los usuarios una visión abstracta de los datos.

Es decir, el sistema esconde los detalles técnicos de cómo se almacenan físicamente y mantienen los datos. A su vez posee una capa de seguridad para restringir el acceso de datos a los usuarios según el uso que deban tener en la BDD.

1.12 Sistema Gestor de Base de Datos 2

Los SGBD surgen a mediados de los años 60's como un intento de mejorar los sistemas de procesamiento de archivos utilizados hasta entonces, de forma que una base de datos sea **unificada** y **compartida**.

Sistemas Gestores de Bases de Datos



1.12 Sistema Gestor de Base de Datos

Niveles de Abstracción

1

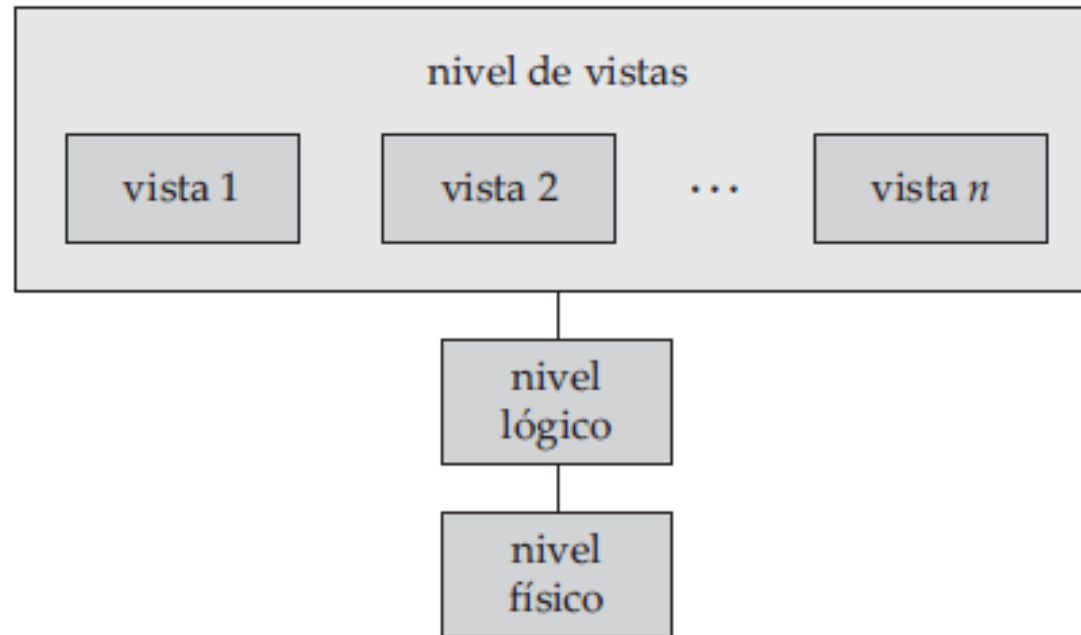
Para que el sistema sea útil debe **recuperar los datos eficientemente**:

- ▶ La necesidad de eficiencia llevó a los diseñadores a usar estructuras de datos complejas para la representación de los datos en la BDD.
- ▶ Muchos usuarios de SGBD no tienen formación en informática, los desarrolladores ocultan esa complejidad a los usuarios mediante varios **niveles de abstracción** para simplificar la interacción de los usuarios con el sistema.

1.12 Sistema Gestor de Base de Datos

Niveles de Abstracción

2



Nivel físico: Usado por el motor de la BDD.

Nivel lógico: Usado por el DBA/Desarrollador.

Nivel de vistas: Usado por usuarios/aplicaciones.

1.12 Sistema Gestor de Base de Datos

Niveles de Abstracción

3

Nivel físico: Es el de más baja abstracción y describe **cómo se almacenan realmente los datos**, en detalle las estructuras de datos complejas de bajo nivel.

Nivel lógico: Describe qué datos se almacenan en la BDD en **estructuras simples** y qué relaciones existen entre esos datos (**Entidades/Relaciones**).

Aunque la implementación de esas estructuras simples en el nivel lógico puede involucrar estructuras complejas del nivel físico, los usuarios del nivel lógico no necesitan preocuparse de esta complejidad. Los DBA/Desarrollador usan el nivel lógico.

1.12 Sistema Gestor de Base de Datos

Niveles de Abstracción

4

Nivel de vistas: El nivel más elevado de abstracción sólo describe parte de la base de datos.

Aunque el nivel lógico usa estructuras más simples, queda algo de complejidad debido a la variedad de información almacenada en grandes BDD. Muchos usuarios/aplicaciones no necesitan toda esta información; en su lugar sólo necesitan tener acceso a una parte de la BDD.

El nivel de abstracción de vistas existe para simplificar su interacción con el sistema. El sistema puede proporcionar muchas vistas para la misma BDD.

EJEMPLARES:

La BDD va cambiando a lo largo del tiempo conforme la información se inserta, modifica y elimina.

La colección de información almacenada en la BDD en un momento dado se denomina **ejemplar** de la BDD. Es como una **foto del estado de los datos** (sus valores) en un momento específico!



ESQUEMAS:

El diseño general de la BDD se denomina **esquema**.

El concepto de **esquema y ejemplar** de la BDD se puede comprender por analogía con los programas escritos en un lenguaje de programación. El esquema de la BDD se corresponde con las declaraciones de las variables de los programas.

Cada variable tiene un valor concreto en un instante dado. Los valores de las variables de un programa en un instante dado se corresponden con un ***ejemplar del esquema de la base de datos***.

ESQUEMAS DEL SGBD:

Los SGBD tienen varios esquemas divididos según los niveles de abstracción:

- ▶ El **esquema físico** describe el diseño de la BDD en el **nivel físico**;
- ▶ El **esquema lógico** describe el diseño de la BDD en el **nivel lógico**.
- ▶ Las BDD también pueden tener **varios esquemas en el nivel de vistas**, a veces denominados **subesquemas**, que describen diferentes vistas de la BDD.

El **esquema lógico** es el más importante por su efecto sobre los programas de aplicación; los programadores crean aplicaciones usando el esquema lógico.

El **esquema físico** (que es la implementación del esquema lógico) está oculto bajo el esquema lógico, y puede modificarse fácilmente sin afectar a los programas de aplicación. Los programas “**dialogan**” con el esquema lógico, por eso hay **independencia física respecto de los datos pues los programas no dependen del esquema físico!** Si cambia el esquema físico no hace falta reescribir los programas.

Los esquemas se describen con el lenguaje **DDL**.

1.13 Clasificación de los SGBD

1

Los SGBD clasifican a las BDD según el modelo de datos utilizado.

Un **modelo** es una abstracción del mundo real y trata de representar cómo se agrupan los datos y la relación que existe entre ellos en el mundo real.

Algunos modelos con frecuencia utilizados son:

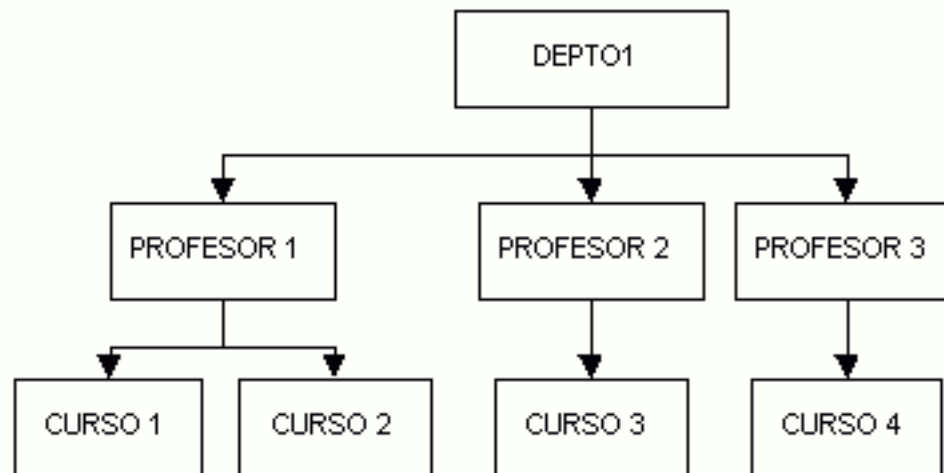
Jerárquico	Multidimensional
De red	Orientado a objetos
Relacional	Documental

1.13 Clasificación de los SGBD

Modelo Jerárquico

2

Modelo antiguo que almacena la información en una **estructura de árbol jerárquico** en donde un **nodo padre** de información puede tener varios **nodos hijo**, y así sucesivamente.

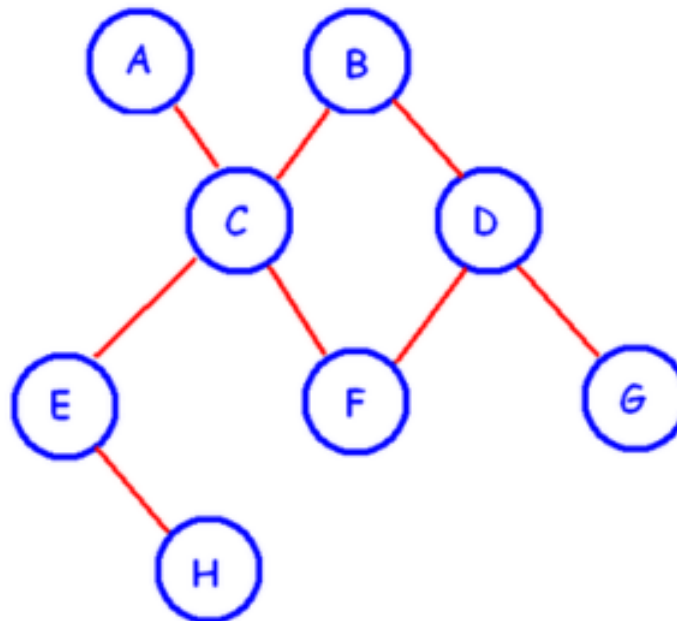


1.13 Clasificación de los SGBD

Modelo de Red

3

Modelo antiguo ligeramente distinto del jerárquico. Su diferencia fundamental es que se permite **que un mismo nodo tenga varios padres** (posibilidad no permitida en el modelo jerárquico).



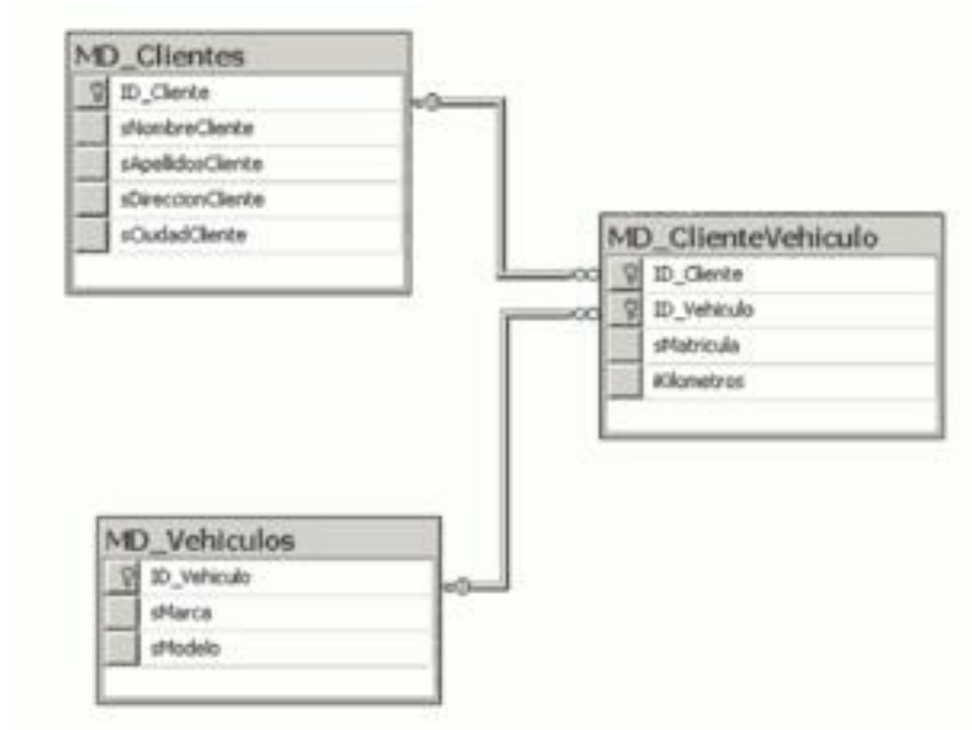
1.13 Clasificación de los SGBD

Modelo Relacional

4

Modelo muy usado. La BDD está formada por un conjunto de **tablas relacionadas**:

- ▶ Cada tabla tiene varias columnas
- ▶ Cada columna tiene un nombre único
- ▶ Cada tabla contiene registros



Este es el modelo que trataremos en la materia.

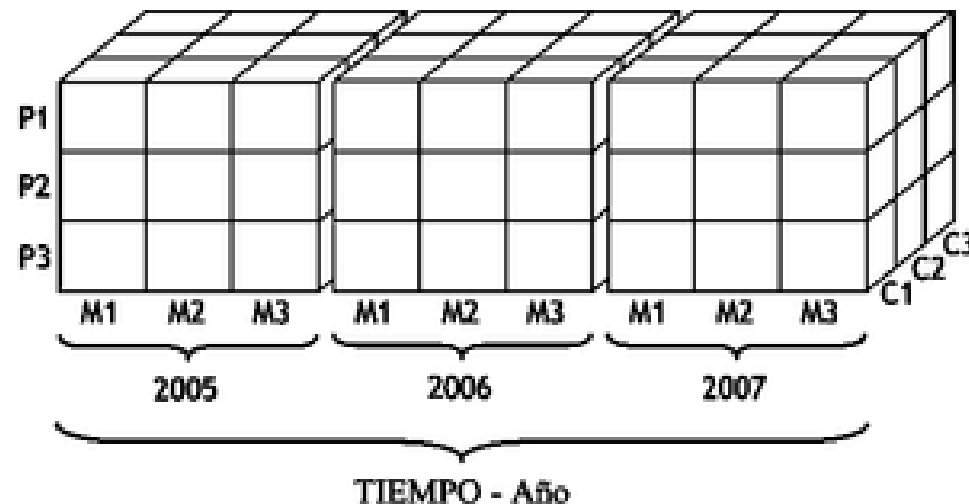
1.13 Clasificación de los SGBD

Modelo Multidimensional

5

La BDD tiene **N dimensiones** y ofrece el aspecto de una hoja de cálculo. Es una extensión del modelo relacional que tiene dos dimensiones.

Tiene usos muy específicos cuyo objetivo es agilizar la consulta de grandes cantidades de datos. Se usan para el proceso analítico de transacciones en línea (OLAP).

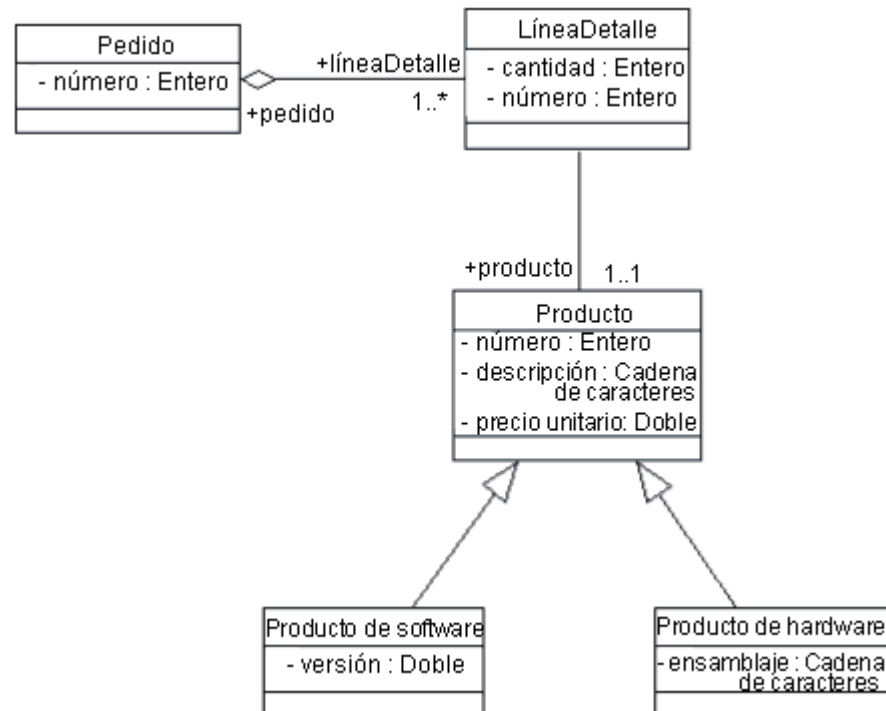


1.13 Clasificación de los SGBD

Modelo Orientado a Objetos

6

La BDD está formada por **objetos completos** (estado y comportamiento). Incorpora los conceptos del **paradigma de objetos**: encapsulamiento, métodos, herencia, polimorfismo.



1.13 Clasificación de los SGBD

Modelo Documental

7

Una BDD documental está constituida por un conjunto de programas que almacenan, recuperan y gestionan **datos de documentos** o **datos semi-estructurados**, pero **sin un formato fijo** (formato flexible). Este tipo de BDD son denominadas “Bases de datos NoSQL”.

A diferencia de las BDD relacionales, están diseñadas sobre una noción abstracta de "**Documento**".



1.13 Clasificación de los SGBD

Modelo Documental

8

JavaScript
Object Notation

JSON

```
{  
  "siblings": [  
    {"firstName": "Anna", "lastName": "Clayton"},  
    {"firstName": "Alex", "lastName": "Clayton"}  
  ]  
}
```

eXtensible Markup
Language

XML

```
<siblings>  
  <sibling>  
    <firstName>Anna</firstName>  
    <lastName>Clayton</lastName>  
  </sibling>  
  <sibling>  
    <firstName>Alex</firstName>  
    <lastName>Clayton</lastName>  
  </sibling>  
</siblings>
```


1.13 Clasificación de los SGBD

Más sobre el Modelo Relacional

9

El **SGBD Relacional** tiene un conjunto de programas encargado de manejar la creación y todos los accesos a la BDD. Normalmente se lo llama **motor de BDD**.

Todo motor de BDD posee un lenguaje llamado **SQL**: **Structured Query Language**.

SQL es un estándar mundial supervisado por la organización **ANSI**: **American National Standard Institute**.

Todos los motores de BDD relacional se basan en este estándar, y cada empresa proveedora le agrega funcionalidad propia (Oracle, IBM, Microsoft, etc.).

1.13 Clasificación de los SGBD

Más sobre el Modelo Relacional

9

SQL se compone de dos lenguajes:

- ▶ Lenguaje de definición de datos para indicar el “**esquema**” de la BDD:
DDL (Data Definition Language), se usa para **describir** todas las **estructuras**: tablas, vistas, programas, funciones, etc.
- ▶ Lenguaje de manipulación de datos para “**consultar y modificar**” la BDD: **DML** (Data Manipulation Language): se usa para las **tareas con los datos**: insertar, recuperar, modificar, eliminar, etc.

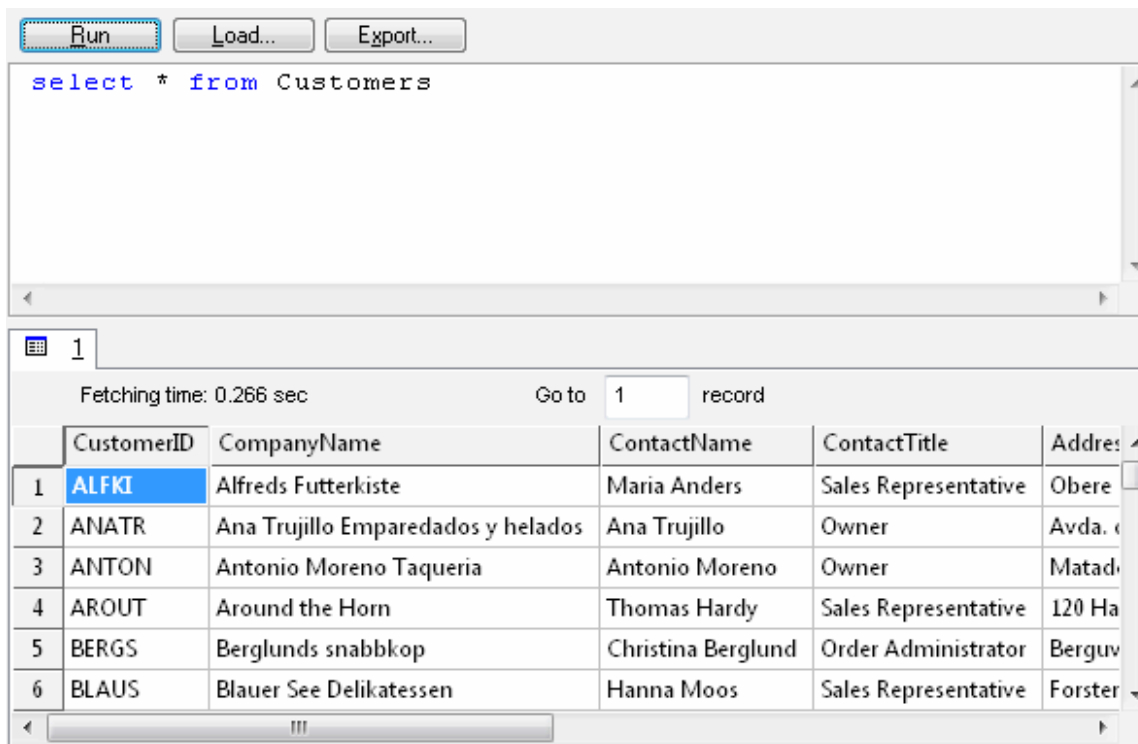
1.13 Clasificación de los SGBD

Más sobre el Modelo Relacional

10

Las instrucciones de los lenguajes DDL y DML, se pueden escribir en la **consola del SGBD** por usuarios que tengan acceso a ella o desde las aplicaciones que

realizan los desarrolladores en algún lenguaje de alto nivel, cuyas aplicaciones se conectan a la BDD.



The screenshot shows a database management interface. At the top, there are buttons for 'Run', 'Load...', and 'Export...'. Below these is a text area containing the SQL query: `select * from Customers`. Below the query area, there is a tab labeled '1'. Below the tab, it says 'Fetching time: 0.266 sec' and 'Go to 1 record'. Below this is a table with 6 rows and 5 columns: CustomerID, CompanyName, ContactName, ContactTitle, and Address. The first row is highlighted in blue.

	CustomerID	CompanyName	ContactName	ContactTitle	Address
1	ALFKI	Alfreds Futterkiste	Maria Anders	Sales Representative	Obere...
2	ANATR	Ana Trujillo Emparedados y helados	Ana Trujillo	Owner	Avda. c...
3	ANTON	Antonio Moreno Taqueria	Antonio Moreno	Owner	Matad...
4	AROUT	Around the Horn	Thomas Hardy	Sales Representative	120 Ha...
5	BERGS	Berglunds snabbkop	Christina Berglund	Order Administrator	Berguv...
6	BLAUS	Blauer See Delikatessen	Hanna Moos	Sales Representative	Forster...

1.14 Algunos SGBD actuales

1

► Base de datos relacionales comerciales:

Oracle, IBM DB2, Microsoft Sql Server, Teradata, SAP Sybase, Informix.

ORACLE®



Microsoft®
SQL Server®

SYBASE®
An SAP Company

Informix®

TERADATA.

Hoy en día el mercado sigue siendo dominado por DB2, SQL Server, Oracle e IBM. En Mainframe o Unix/Linux, Oracle y DB2 suelen ser los líderes, mientras que en sistemas operativos Windows SQL Server suele ser la mejor opción.

1.14 Algunos SGBD actuales

2

- ▶ **Bases de datos relacionales de libre distribución:** MySQL, María DB y PostgreSQL.



- ▶ **Bases de Datos NoSQL:** MongoDB, Redis, Cassandra, HBase, Microsoft Azure DocumentDB.



- ▶ **Bases de Datos embebidas para pequeños dispositivos:** SQLite (para aplicaciones Android).



- ▶ <https://blog.pandorafms.org/es/tipos-de-bases-de-datos-y-las-mejores-bases-de-datos-del-2016/>

1.15 Diseño de bases de datos

1

Las BDD se diseñan para gestionar grandes cantidades de información. Implica principalmente el diseño del **esquema** de la BDD.

El diseño de un entorno completo de aplicaciones para que la BDD satisfaga las necesidades de la empresa, **exige prestar atención** a un conjunto de aspectos más amplio.

La fases de diseño son:

1. Especificación de **requisitos de los usuarios**
2. Creación del **modelo conceptual** (modelo entidad-relación)
3. Validación del modelo conceptual contra los **requisitos funcionales**
4. Creación del **modelo lógico/físico** de la BDD en un SGBD

1. Especificación de requisitos de los usuarios:

Para identificar los requisitos de datos de los usuarios se realizan **relevamientos/entrevistas a usuarios!**

El resultado de esta fase es la **especificación de los requisitos de los usuarios.**

2. Creación del modelo conceptual: Luego el diseñador elige un **modelo de datos** y traduce los requisitos en un **modelo conceptual.**

Este **diseño conceptual ofrece una visión general detallada de la Empresa.** En este punto la atención se centra en describir los datos y sus relaciones, más que en especificar los detalles del almacenamiento físico.

En términos del modelo relacional, el proceso de diseño conceptual implica identificar *qué atributos* se desea capturar en la BDD y *cómo agruparlos en tablas*. El “**qué**” es una decisión conceptual (el **DA** define). La parte del “**cómo**” es esencialmente un problema informático (el **DBA** define).

3. **Validación del modelo conceptual:** El modelo conceptual completamente desarrollado indica los *requisitos funcionales* de la empresa. En la especificación de requisitos funcionales **los usuarios describen los procesos y el tipo de operaciones** que se llevarán a cabo con los datos.

1.15 Diseño de bases de datos

4

En esta etapa del diseño conceptual el diseñador puede **revisar el esquema** para asegurarse de que **satisface los requisitos funcionales**.

4. **Creación del modelo lógico/físico:** El proceso de pasar de un modelo conceptual (modelo de datos abstracto) a la implementación de la BDD implica **relacionar el esquema conceptual de alto nivel** con el modelo de implementación de datos lógico/físico del sistema de bases de datos que se va a usar: Se elige el SGBD a usar (el motor) y se pasa el modelo conceptual creando la BDD.

1.16 Arquitectura de las bases de datos 1

La arquitectura de los sistemas de BDD se ve muy influida por el sistema informático subyacente sobre el que se ejecuta. Pueden diseñarse también para aprovechar las arquitecturas de **computadoras paralelas** (con varios procesadores).

La arquitectura puede ser:

- ▶ Centralizada
- ▶ Cliente-Servidor
- ▶ Distribuida

Centralizada:

El SGBD está **centralizado** en un solo lugar físico (una sola máquina y cpu), los usuarios trabajan en terminales que muestran sólo resultados.



1.16 Arquitectura de las bases de datos 3

Cliente - Servidor:

El SGBD del tipo **cliente-servidor**, en los que una máquina servidora ejecuta el trabajo que una máquina cliente le solicita.

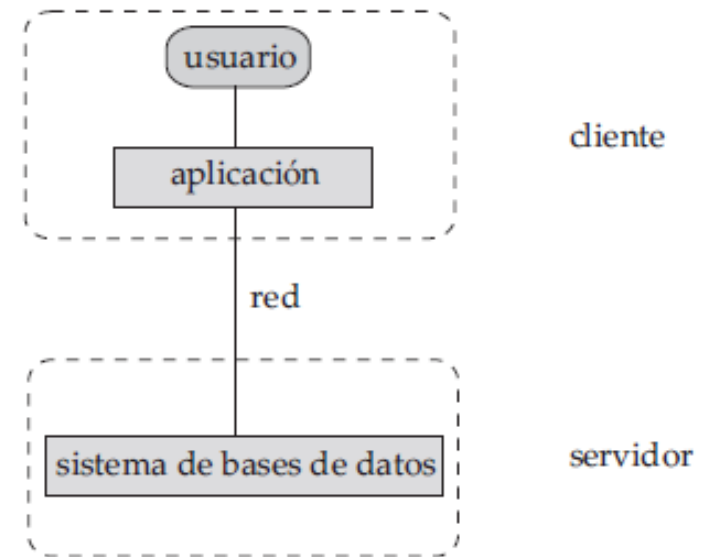
Hoy en día la mayor parte de los usuarios no está presente en el lugar físico donde está el SGBD, sino que se conectan a través de una red. **Lado cliente** es donde interactúa un usuario con una aplicación y **lado servidor** donde está el SGBD que tiene las BDD.

1.16 Arquitectura de las bases de datos 4

Cliente – Servidor 2 capas:

Las aplicaciones de BDD suelen dividirse en dos o tres partes o capas.

En una **arquitectura de dos capas**, la aplicación está en la máquina cliente y tiene la **interfaz de usuario** y la **lógica de negocio**, que llama a la funcionalidad del SGBD en la máquina servidor.



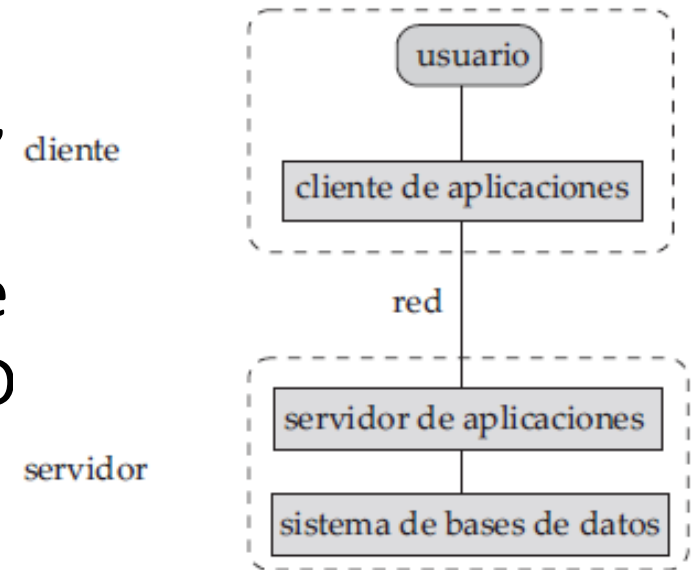
(a) Arquitectura de dos capas

1.16 Arquitectura de las bases de datos 5

Cliente – Servidor 3 capas:

En una **arquitectura de tres capas**, en la máquina cliente está la interfaz de usuario y no contiene ninguna llamada directa a la BDD. El extremo cliente se comunica con un **servidor de aplicaciones**.

El servidor de aplicaciones se comunica con el SGBD para tener acceso a los datos. La lógica de negocio de la aplicación, está en el servidor de aplicaciones, en lugar de estar distribuida entre múltiples clientes.



(b) Arquitectura de tres capas

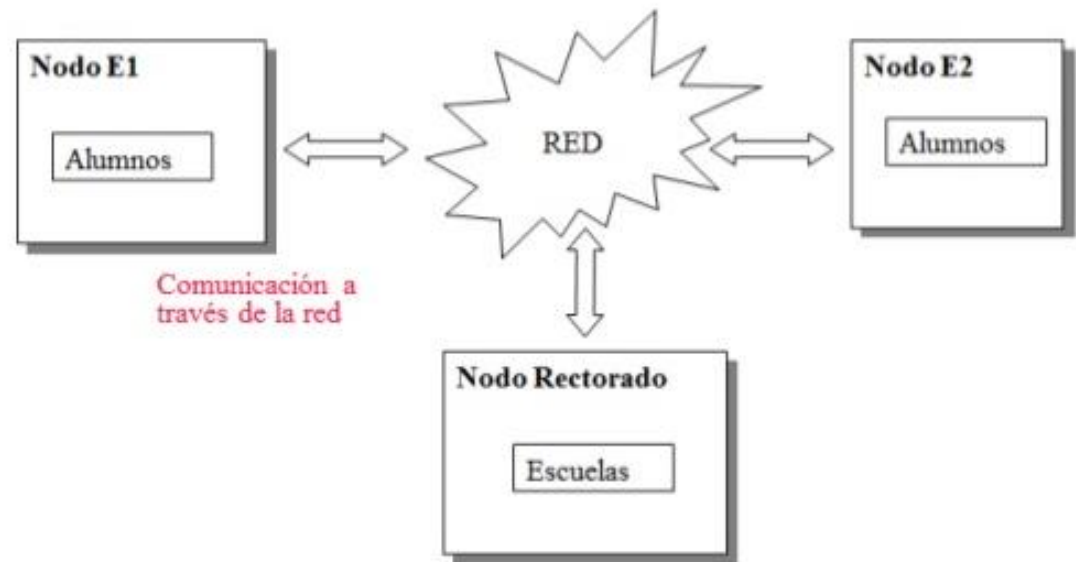
1.16 Arquitectura de las bases de datos 6

Distribuidas:

Las BDD **distribuidas** se extienden por varias máquinas geográficamente separadas. Cada una de las computadoras donde está la BDD se llama **nodo**.

Existe **replicación de datos** y mecanismos de **sincronización** hacia todas las réplicas.

Ejemplo de BD distribuida



Autoevaluación

1. ¿De qué forma podemos guardar los datos?
2. ¿Cuáles son los tipos de organización de archivos que conoce?
3. ¿Cuál es la diferencia que existe entre un registro y un campo?
4. ¿Qué es una BDD? ¿Qué es un DA? ¿Qué es un DBA?
5. ¿Cuáles son las ventajas de utilizar una base de datos?
6. ¿Qué es un sistema de información?
7. ¿Qué es un SGBD?
8. ¿Qué es Hardware? ¿Qué es Software?
9. ¿Qué significan los términos Datos e Información?
10. ¿Cuáles son las funciones del administrador de base de datos?

Autoevaluación

11. ¿Qué es un ejemplar de un esquema de la BDD?
12. Una empresa realiza una copia de seguridad de su BDD una vez por día. ¿Cuántos ejemplares de la BDD hay en la copia de seguridad?
13. ¿Cuántos niveles de abstracción tiene una BDD? Describa brevemente que usuarios/aplicaciones/niveles interactúan.
14. ¿Cuáles son las fases del diseño de una BDD?
15. Mencione las arquitecturas de las BDD según su modelo de ejecución.