

Unidad I

Introducción a las Bases de Datos

- 1.1 Conceptos Básicos.
- 1.2 Objetivo de las Bases de Datos.
- 1.3 Áreas de Aplicación de los Sistemas de Bases de datos.
- 1.4 Modelos de datos.
- 1.5 Clasificación de Bases de Datos.
- 1.6 Arquitectura de un DBMS.
- 1.7 Arquitectura de un Sistema de Bases de Datos.

1.1 Conceptos Básicos

Sistema de Información

Las **Bases de Datos** son parte de un **Sistema de Información**. Los Sistemas de Información son importantes porque las organizaciones requieren para su funcionamiento la transmisión de información entre los diferentes elementos que la componen:



Entre Personas



Entre Departamentos

1.1 Conceptos Básicos

En organizaciones complejas ...

- ◆ La comunicación interpersonal es muy riesgosa.
- ◆ Como en el juego del teléfono descompuesto, hay un gran riesgo de desintegración de la información: por olvido, mala interpretación, etc.



La transmisión de información interpersonal forma parte de un **sistema de información** informal (en los sistemas informales no hay reglas o no se respetan).

1.1 Conceptos Básicos

¿Qué es un **Sistema**?

La Real Academia de la Lengua Española, establece:

Conjunto de cosas que relacionadas entre sí, ordenadamente contribuyen a determinado objetivo.

Hay Sistemas Naturales ...



1.1 Conceptos Básicos

También hay sistemas **artificiales** (creados por las personas), las organizaciones de cualquier tipo son ejemplos de ellas.

Sistema: un conjunto de cosas que relacionadas entre sí, ordenadamente contribuyen a determinado objetivo.

Cosa: en el sentido más amplio, son personas, muebles, edificios, máquinas, métodos, etc.

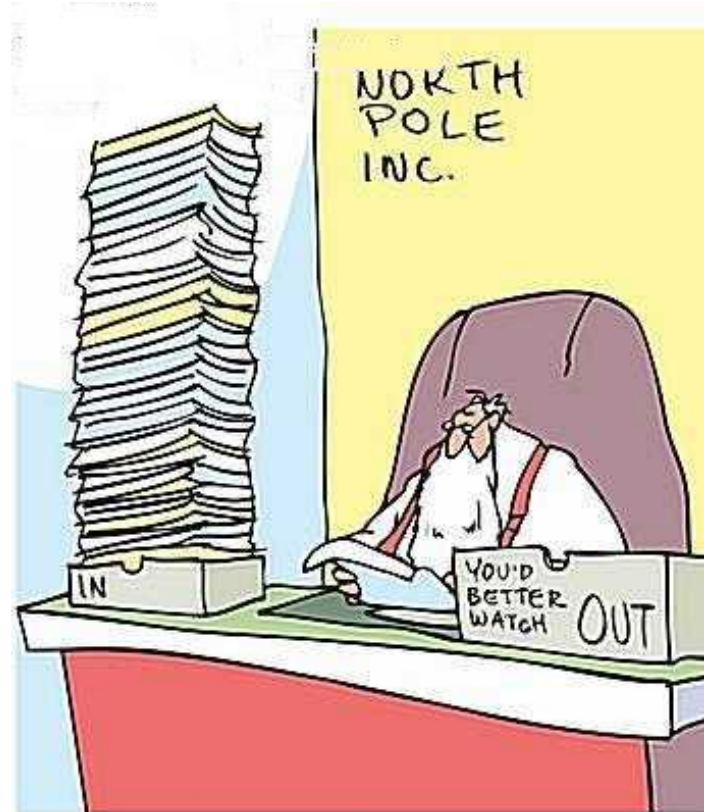
Objetivo: cumplir con la **misión** de la organización.



1.1 Conceptos Básicos

¿Qué es entonces un Sistema de Información?

- ◆ Colección de personas, procedimientos y equipo contruidos y operados para recolectar, almacenar, procesar y proveer información.
- ◆ Los **Sistemas de Información** no necesariamente tienen que estar automatizados.



1.1 Conceptos Básicos

- ◆ Sin embargo, desde hace mucho tiempo, la disminución de los costos de tecnología computacional ha permitido aprovechar las ventajas de la automatización.
- ◆ Los **Sistemas de Información** solo son una parte del Sistema Completo (la empresa o institución misma).



1.1 Conceptos Básicos

Base de Datos

Dato:

Aquel o aquellos valores almacenados en un campo, un registro o un archivo, sin importar si se trata de valores tradicionales como números o cadenas, archivos de texto o binarios como por ejemplo imágenes, audio, video, etc.

1.1 Conceptos Básicos

Ejemplos:

- ◆ Salario Diario de un empleado.
- ◆ Lista de materias que cursa un alumno durante el semestre actual.
- ◆ El Programa de Estudios de un Curso.
- ◆ Todos los datos de un profesor del ITD.
- ◆ Fotografía de un alumno.
- ◆ Video del interior de un edificio en construcción.
- ◆ Las Notas del Himno Nacional Mexicano.

1.1 Conceptos Básicos

¿Qué es una Base de Datos?

Colección de datos pertenecientes a una organización.

- Todos los datos dentro de la organización tienen relación entre sí aunque no siempre sea evidente.

1.1 Conceptos Básicos

Sistemas de
Información
Automatizados



1.2 Objetivos de las Bases de Datos

Los Sistemas de Bases de Datos fueron diseñados para resolver los problemas que se plantean enseguida.

1.2 Objetivos de las Bases de Datos

A) Redundancia e inconsistencia.

Uno de los Tecnólogos del TecNM inicia el desarrollo de un sistema de información automatizado.

Se solicita al ingeniero líder de proyecto **un módulo para guardar los datos de los maestros y alumnos.**

Observemos cuidadosamente, La organización de los datos que realizó el ingeniero encargado:

MAESTROS			
RFC	CURP	Nombre	Sueldo
PP11	P1	Luis Pasteur	15,000.00
AA00	A0	Abraham Lincoln	12,000.00
AA11	A1	Aristóteles	12,500.00
TT44	T4	Piotr Ilich Tchaikovsky	25,000.00
GG22	G2	Johannes Gutenberg	13,000.00
BB99	B9	Sandro Botticelli	20,000.00

ALUMNOS			
NumControl	CURP	Nombre	Semestre
17040151	P5	Conejo Pérez	2
17040587	J4	Benito Juárez	2
17040014	R7	José Revueltas	3
18040121	O1	Lorena Ochoa	2
18040150	G2	Johannes Gutenberg	1

1.2 Objetivos de las Bases de Datos

En las tablas (una tabla es similar a un archivo) de la diapositiva anterior se observa que hay una persona que es profesor y también es alumno en la misma escuela (estudia un doctorado mientras enseña en licenciatura) lo que causa que se conserve en la Base de Datos información redundante.

La redundancia no es un problema, pero **la inconsistencia que se puede presentar como consecuencia de la redundancia si lo es**, por ejemplo, si hubiera necesidad de cambiar la CURP o el nombre de Gutenberg y solo se cambia en uno de los dos lugares.

1.2 Objetivos de las Bases de Datos

Los criterios de diseño de BD nos muestran la mejor forma de organizar los datos para evitar redundancia.

Si por alguna razón se decide diseñar con redundancia, se debe instruir al DBMS

(Sistema para el manejo de Bases de Datos, del que hablaremos posteriormente) para que se asegure de que no se presenten inconsistencias estableciendo el criterio de que cambiando el valor de un dato, se cambie en los diferentes lugares donde se encuentra.

1.2 Objetivos de las Bases de Datos

B) Dificultad en el acceso a los datos.

El ingeniero encargado de la BD cuenta con los datos organizados de cierta forma para llevar el registro de las materias impartidas en el Tecnológico, así como los horarios de clases de los estudiantes.

En la diapositiva siguiente se muestra un ejemplo de una la organización de esos datos.

1.2 Objetivos de las Bases de Datos

ALUMNOS				
NumControl		CURP	Nombre	Semestre
17040151		P5	Conejo Pérez	2
17040587		J4	Benito Juárez	2
17040014		R7	José Revueltas	3
18040121		O1	Lorena Ochoa	2
18040150		G2	Johannes Gutenberg	1
MATERIAS				
Clave	Nombre	HsTeo	HsPra	Creditos
1810	Estructuras de Datos	3	2	5
1820	Fundamentos Bases de Datos	3	2	5
1500	Quimica I	4	0	4
1850	Taller de Bases de Datos	4	0	4
1600	Matemáticas II	5	0	5

HORARIOS_ALUMNOS				
Alumno	Materia	Dia	Hora	Salón
17040151	1810	Lunes	15:00	SC4
17040151	1810	Martes	15:00	SC4
17040151	1810	Miercoles	15:00	SC4
17040151	1810	Jueves	15:00	SC4
17040151	1810	Viernes	15:00	SC4
17040014	1500	Lunes	10:00	LC23
17040014	1500	Martes	10:00	LC23
17040014	1500	Miercoles	10:00	SC8
17040014	1500	Jueves	10:00	SC8
18040150	1600	Martes	10:00	T1
18040150	1600	Jueves	10:00	T1

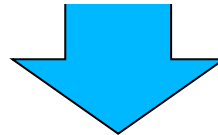
1.2 Objetivos de las Bases de Datos

- ✓ El coordinador de carrera necesita saber los nombres de los alumnos que tienen clase cierto día de la semana en cierto salón, para avisarles que se les va a asignar otra aula temporalmente por razones de mantenimiento, sin embargo, el sistema no cuenta una opción para obtener un reporte con esa información.
- ✓ Como es de esperar, habrá muchos datos en las tablas (los que se muestran son una pequeña parte) ya que el número de alumnos es importante.
- ✓ Si un programador tuviera que hacer el programa, se requerirá algo de tiempo con el que tal vez no se cuenta.
- ✓ Los sistemas de bases de datos permiten procesar información rápidamente mediante un lenguaje llamado **SQL**.

1.2 Objetivos de las Bases de Datos

- ✓ Las expresiones **SQL** son mucho más simples que el código de un lenguaje de programación.

```
{
vector<string> vec;
Entrada.open("C:\\ITD\\HorariosAlumnos.dat", ios::in | ios::binary);
Entrada.seekg(0, ios::beg);
Entrada.read(reinterpret_cast<char *>(&RegistroTemp), sizeof(HorarioAlumno) );
while (not Entrada.eof())
{
    int pos = std::find(vec.begin(), vec.end(), RegistroTemp.NumControl);
    if (pos == vec.end()) && RegistroTemp.dia=='jueves')
        vec.push_back(RegistroTemp.NumControl);
    Entrada.read(reinterpret_cast<char *>(&RegistroTemp), sizeof(HorarioAlumno) );
};
Entrada.close();
cout << ' Total de Alumnos con clases los jueves: ' << vec.size() << endl ;
system("pause");
};
```



select distinct Alumno
from HorariosAlumnos where Dia='jueves'

1.2 Objetivos de las Bases de Datos

C) Problemas de Integridad.

MATERIAS				
Clave	Nombre	HsTeo	HsPra	Creditos
1810	Estructuras de Datos	3	2	5
1820	Fundamentos Bases de Datos	3	2	5
1500	Quimica I	4	0	4
1850	Taller de Bases de Datos	4	0	4
1600	Matemáticas II	5	0	5

Observemos los datos correspondientes a las Materias.

- ✓ La columna **Creditos** siempre es la suma de los valores de las columnas **HsTeo** y **HsPra** y puede producirse una inconsistencia si el usuario, mediante una interfaz, modifica una de las dos primeras columnas, pero no el atributo **Creditos**.
- ✓ En los sistemas de bases de datos se puede definir una **Columna Calculada** para que el valor del atributo **Creditos** se calcule de manera **automática** sin que el usuario tenga necesidad de escribir el valor.

1.2 Objetivos de las Bases de Datos

C) Problemas de Integridad. Integridad Referencial.

ALUMNOS			
NumControl	CURP	Nombre	Semestre
17040151	P5	Conejo Pérez	2
17040587	J4	Benito Juárez	2
17040014	R7	José Revueltas	3
18040121	O1	Lorena Ochoa	2
18040150	G2	Johannes Gutenberg	1

MATERIAS				
Clave	Nombre	HsTeo	HsPra	Creditos
1810	Estructuras de Datos	3	2	5
1820	Fundamentos Bases de Datos	3	2	5
1500	Quimica I	4	0	4
1850	Taller de Bases de Datos	4	0	4
1600	Matemáticas II	5	0	5

HORARIOS_ALUMNOS				
Alumno	Materia	Día	Hora	Salón
17040151	1810	Lunes	15:00	SC4
17040151	1810	Martes	15:00	SC4
17040151	1810	Miercoles	15:00	SC4
17040151	1810	Jueves	15:00	SC4
17040151	1810	Viernes	15:00	SC4
17040014	1500	Lunes	10:00	LC23
17040014	1500	Martes	10:00	LC23
17040014	1500	Miercoles	10:00	SC8
17040014	1500	Jueves	10:00	SC8
18040150	1600	Martes	10:00	T1
18040150	1600	Jueves	10:00	T1

Se presentaría una falla de integridad referencial, en caso de que **se permitiera añadir** un numero de control o la clave de una materia **inexistentes** en las tablas Alumnos o Materias.

1.2 Objetivos de las Bases de Datos

C) Problemas de Integridad. Integridad Referencial.

ALUMNOS			
NumControl	CURP	Nombre	Semestre
17040151	P5	Conejo Pérez	2
17040587	J4	Benito Juárez	2
17040014	R7	José Revueltas	3
18040121	O1	Lorena Ochoa	2
18040150	G2	Johannes Gutenberg	1

MATERIAS				
Clave	Nombre	HsTeo	HsPra	Creditos
1810	Estructuras de Datos	3	2	5
1820	Fundamentos Bases de Datos	3	2	5
1500	Quimica I	4	0	4
1850	Taller de Bases de Datos	4	0	4
1600	Matemáticas II	5	0	5

HORARIOS_ALUMNOS				
Alumno	Materia	Día	Hora	Salón
17040151	1810	Lunes	15:00	SC4
17040151	1810	Martes	15:00	SC4
17040151	1810	Miercoles	15:00	SC4
17040151	1810	Jueves	15:00	SC4
17040151	1810	Viernes	15:00	SC4
17040014	1500	Lunes	10:00	LC23
17040014	1500	Martes	10:00	LC23
17040014	1500	Miercoles	10:00	SC8
17040014	1500	Jueves	10:00	SC8
18040150	1600	Martes	10:00	T1
18040150	1600	Jueves	10:00	T1

El caso inverso al planteado en la diapositiva anterior produce también una falla de integridad referencial: **No se debe permitir eliminar** una materia o un alumno de las tablas Alumnos o Materias si se van a conservar aun los datos correspondientes de la tabla **Horarios_Alumnos**.

1.2 Objetivos de las Bases de Datos

D) Problemas de Atomicidad.

Cada vez que se realiza la titulación de un estudiante, a los profesores se les paga \$200.00 por cada titulación mediante Tesis profesional. Cuando la titulación se realiza por otra opción, se les paga \$100.00.

Estado de consistencia de los datos involucrados:

ALUMNOS_TITULADOS			
NumControl	Opcion	Fecha	Maestro Asesor
17040151	Tesis	10/02/2017	A1
17040587	Examen Global	20/11/2017	P1
17040014	Promedio	14/09/2017	A1
18040121	Examen Global	09/12/2017	P1
18040150	Examen Global	03/11/2017	T4

MAESTROS				
RFC	CURP	Nombre	Sueldo	Pagos Titulacion
PP11	P1	Luis Pasteur	15,000.00	200.00
AA00	A0	Abraham Lincoln	12,000.00	-
AA11	A1	Aristóteles	12,500.00	300.00
TT44	T4	Piotr Ilich Tchaikovsky	25,000.00	100.00
GG22	G2	Johannes Gutenberg	13,000.00	-
BB99	B9	Sandro Botticelli	20,000.00	-

1.2 Objetivos de las Bases de Datos

D) Problemas de Atomicidad.

- Átomo = indivisible.
- Cada vez que se registra la titulación de un alumno se ejecutan 2 procesos:
 1. Se añade un renglón con los datos del alumno, del profesor, la opción de titulación y la fecha a la tabla **ALUMNOS_TITULADOS**)
 2. Se acumula a cada profesor el importe del pago en la tabla **MAESTROS**.

1.2 Objetivos de las Bases de Datos

D) Problemas de Atomicidad.

- Si hubiera una caída del sistema (falla de energía, por ejemplo) justo después de la primera operación, pero antes de efectuarse la segunda, el total a pagar al profesor quedaría incorrecto.
- El DBMS (Sistema para el manejo de Bases de Datos) puede **restaurar los datos** al estado **correcto** anterior a la falla.
 - **Si no se realiza una de las instrucciones previstas, no debe realizarse ninguna, por eso debe deshacerse lo que se haya hecho en forma parcial.**

1.2 Objetivos de las Bases de Datos

E) Anomalías en el acceso concurrente

Concurrencia:

Simultaneidad de dos o más eventos (dos o más procesos afectan un mismo dato).

- Si se ejecuta un programa, que no considera la posibilidad de acceso concurrente, en una plataforma de red, puede haber anomalías.

1.2 Objetivos de las Bases de Datos

E) Anomalías en el acceso concurrente

- Supongamos que la aplicación que se escribió para registrar los pagos a profesores por exámenes de titulación estaba instalada en una computadora personal.
- Con el tiempo, el volumen de pagos aumenta, se instala una red y la aplicación empieza a usarse bajo un ambiente multiusuario.

1.2 Objetivos de las Bases de Datos

E) Anomalías en el acceso concurrente

Algoritmo de la aplicación para pagos de titulación:

- Se presenta un formulario inicial:
 - El usuario indica los datos Alumno y Maestro
 - El programa lee de la tabla **MAESTROS** el total de pagos a la fecha, correspondientes al profesor que se indicó.
- Luego se presenta otro formulario:
 - El usuario indica los datos Fecha y Opción de titulación.
 - Se guarda el nuevo renglón en la tabla **ALUMNOS_TITULADOS**.
 - Se suma al total el importe del pago según la opción y se guarda el nuevo valor para el maestro en la tabla **MAESTROS**.

1.2 Objetivos de las Bases de Datos

E) Anomalías en el acceso concurrente ... continuación

...

- ✓ Un programa como el anterior puede haber sido usado durante años bajo un ambiente monousuario o incluso bajo un ambiente de red sin presentar problemas.
- ✓ Sin embargo, en ambiente multiusuario, bajo ciertas condiciones, puede presentarse una situación como la que se plantea enseguida que pone en riesgo la integridad de los datos.

1.2 Objetivos de las Bases de Datos

E) Anomalías en el acceso concurrente ... continuación ...

- ✓ Se realizan muchas titulaciones de alumnos en cierto periodo y se asigna a dos personas a capturar la información.
- ✓ Por coincidencia, a cada persona le correspondió registrar exámenes de titulación asesorados por un mismo profesor y casi simultáneamente empiezan la captura.

1.2 Objetivos de las Bases de Datos

Usuario que captura los datos de titulaciones

- El usuario indica los datos
Alumno y Maestro
- El programa lee el total de pagos
al maestro correspondiente
(\$1,000)
- El usuario indica los datos
Fecha y Opción (Promedio)
- Se guarda el nuevo total
(\$1,100)

1.2 Objetivos de las Bases de Datos

Capturista 1

10:30 am.

- El usuario indica los datos Alumno y Maestro
- El programa lee el total de pagos al maestro correspondiente **(\$1,000)**
- Se levanta a prepararse un café.

10:35 am.

- Regresa del café
- El usuario indica los datos Fecha y Opción (Promedio)
- Se guarda el nuevo total **(\$1,100)**

Capturista 2

10:34 am.

- El usuario indica los datos Alumno y Maestro
- El programa lee el total de pagos al maestro correspondiente **(\$1,000)**

10:36 am.

- El usuario indica los datos Fecha y Opción (Tesis)
- Se guarda el nuevo total **(1,200.00)**

1.2 Objetivos de las Bases de Datos

- ¿Cuál es el total que se conservará?
- ¿Es correcto?
- ¿Cual debe ser el total correcto?
- Esta situación puede presentarse aunque el intervalo entre la lectura del total anterior y la escritura del nuevo total sea una **fracción de segundo**.

1.2 Objetivos de las Bases de Datos

F) Problemas de Seguridad

Se presentan diversos escenarios respecto a los riesgos de seguridad, algunos de ellos son:

1.- Hipotéticamente un usuario pide a un programador hacer un cambio a un dato sin usar la interfaz del sistema, conocido comúnmente como **"acceso inseguro por debajo"** porque quiere evitar que una liga de **columna calculada** o **integridad referencial** se cumplan.

- Si se permitiera eso, se perdería la integridad de los datos.
- El software de base de datos **siempre interviene** entre cualquier usuario y los datos y obliga a la conservación de la integridad.

1.2 Objetivos de las Bases de Datos

F) Problemas de Seguridad

2.- Si algún intruso desea ver los datos sin que el software de base de datos esté presente, se encontrará con que no es posible acceder a ellos fácilmente ya que se encuentran fuertemente **encriptados**.

3.- No todos los usuarios de una BD, los programadores, por ejemplo, deben tener **acceso a la información completa**, ni deben tener la posibilidad de hacer cambios a las restricciones de la BD por seguridad. El software de BD permite otorgar el acceso a diferentes niveles a los diferentes usuarios.

1.3 Areas de aplicación de los Sistemas de Bases de Datos.

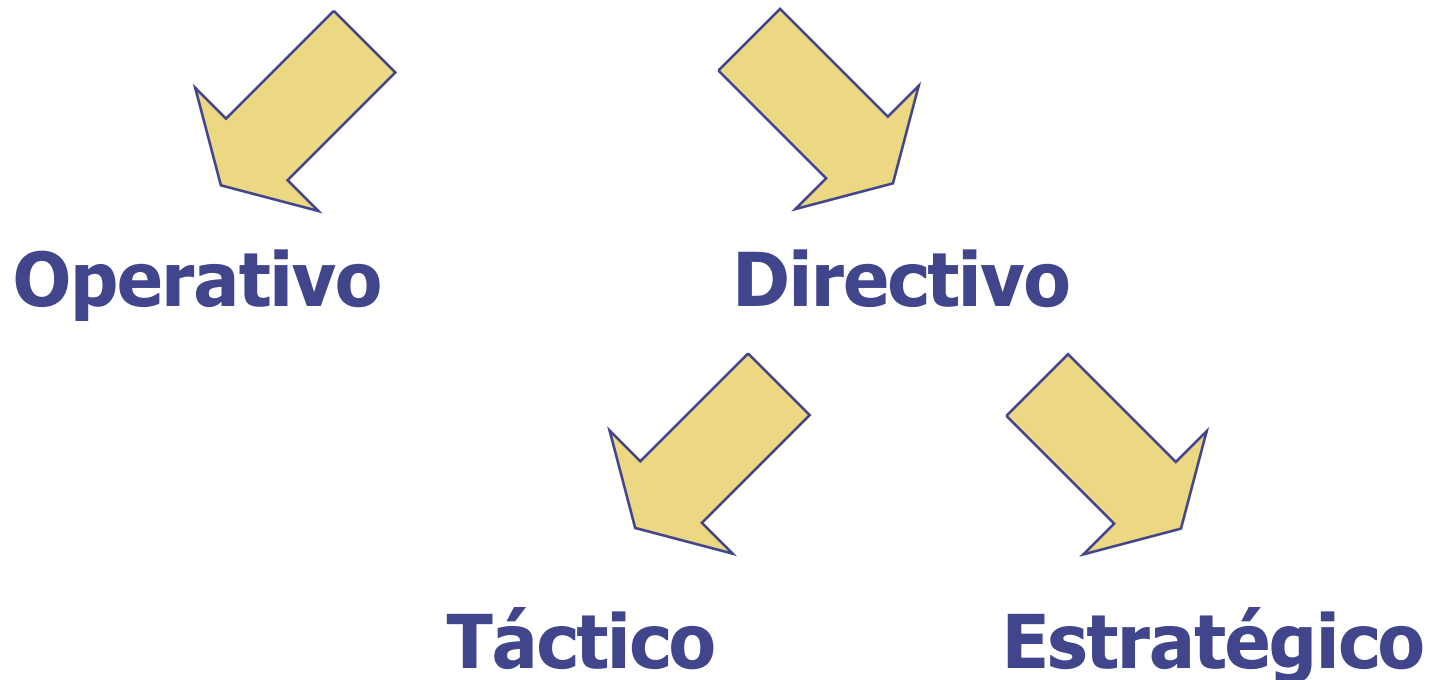
La administración de una organización, se lleva a cabo en todas las áreas de la empresa o institución.

Administrar

- Dirigir una Institución.
- Ordenar, disponer, organizar la riqueza.
- Graduar o dosificar el uso de algo, para obtener mayor rendimiento de ello o para que produzca mejor efecto.

1.3 Áreas de aplicación de los Sistemas de Bases de Datos.

Niveles de Administración de las Organizaciones



1.3 Áreas de aplicación de los Sistemas de Bases de Datos.

Ejemplos de Tareas Operativas de Administración

◆ Puntos de Venta



◆ Elaboración de Nóminas



◆ Pólizas Contables



1.3 Áreas de aplicación de los Sistemas de Bases de Datos.

Ejemplos de Labores Tácticas (Control) de Administración

Análisis Rendimiento
Vendedores (Enero)

EMPLEADO	VENTAS	SUELDO
Donald	7,000.00	5,000.00
Tío Rico	100,000.00	40,000.00
Daisy	50,000.00	5,000.00

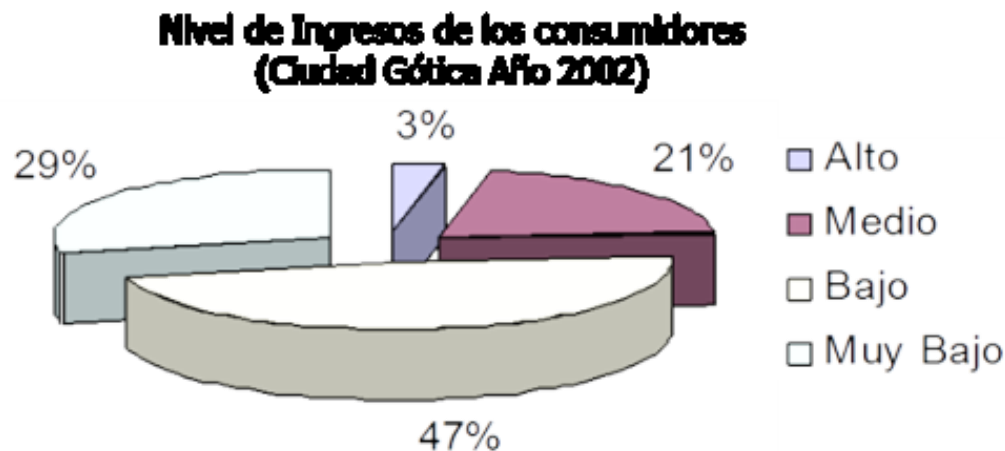
Supervisión taller
de servicio.



1.3 Áreas de aplicación de los Sistemas de Bases de Datos.

Ejemplo de Labores Estratégicas (Planeación) de Administración

Análisis de Mercado



1.3 Áreas de aplicación de los Sistemas de Bases de Datos.

Base de Datos de una Institución Educativa



- ❖ Datos de Alumnos, Carreras, Materias, Programas de Estudio
- ❖ Datos de Maestros, Secretarias, Intendentes, Categorías, Políticas de Promoción, Nóminas
- ❖ Datos de las Aulas, Laboratorios, Equipo y Accesorios, Bibliotecas, etc. (Inventarios en general).
- ❖ Datos de Proveedores
- ❖ Datos Contables y Fiscales
- ❖ Datos de Empresas Objetivo (bolsa de trabajo)
- ❖ Datos de Otras Instituciones Educativas que proveen estudiantes

Todos los Datos están relacionados entre sí .. ¿en qué forma pueden estarlo un intendente y un proveedor?

1.3 Áreas de aplicación de los Sistemas de Bases de Datos.

Base de Datos de un Aserradero (productor de madera dimensionada)



- ❖ Datos de los Proveedores de trocería
- ❖ Datos de los Clientes, Crédito y Cobranzas
- ❖ Inventarios de madera
- ❖ Datos de los Equipos de Transporte (Fletes)
- ❖ Activos Fijos
- ❖ Datos Contables y Fiscales
- ❖ Datos de los Empleados, Nominas y administración de personal

1.3 Áreas de aplicación de los Sistemas de Bases de Datos.

Base de Datos de una Concesionaria de automóviles



- ❖ Datos de los Proveedores de Autos
 - ❖ Fabricante de la Marca, otros concesionarios, particulares que dejan a cuenta sus vehículos.
- ❖ Datos de los Proveedores de Refacciones
- ❖ Datos del Inventario de Automóviles
- ❖ Datos del Inventario de Refacciones
- ❖ Datos de Vehículos en Servicio
- ❖ Datos Activos Fijos
- ❖ Datos Contables y Fiscales
- ❖ Datos de los Empleados, Nominas y administración de personal
- ❖ Datos de los Equipos de Reparto y de Servicio.

1.3 Áreas de aplicación de los Sistemas de Bases de Datos.

Los anteriores son solo unos ejemplos de las áreas de aplicación de las Bases de Datos, sin embargo, en la actualidad, todas las actividades humanas, pueden aprovechar las ventajas de la automatización de las bases de datos.

1.4 Modelos de Datos

Los modelos de datos son conjuntos de conceptos usados para describir los datos y las relaciones entre ellos.

Son una herramienta esencial para el diseño de una base de datos.

Los modelos lógicos se clasifican en modelos basados en objetos y modelos basados en registros.

Los modelos lógicos se emplean para describir los datos al nivel de fácil comprensión para los diseñadores y programadores.

1.4 Modelos de Datos

Modelo Relacional.

El Modelo Relacional es un modelo basado en registros, como los de un archivo plano. Los datos y sus relaciones se representan mediante tablas (una tabla equivale a una relación matemática, por ello el nombre del modelo).

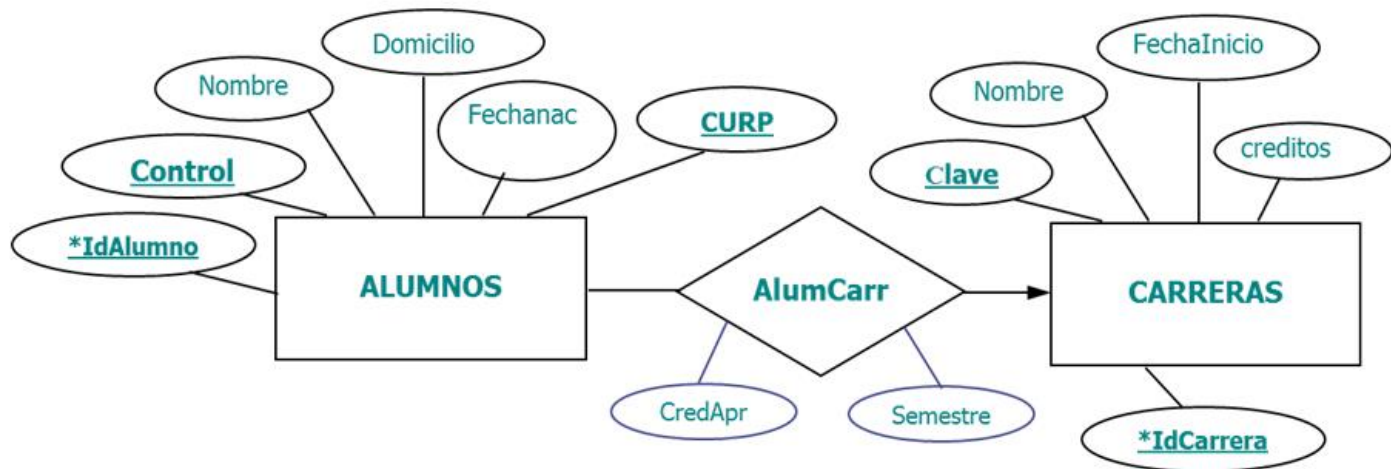
Control	Nombre-Alumno	Carrera	Materia	Grupo
98040151	Lopez Jimenez Carmen	Ing. En Sistemas	Estructuras de Datos I	3X
97040587	Alvarado Rios Jimena	Ing. Industrial	Inv. De Operaciones II	4Q
97040014	Molina Navar Luis	Ing. En Sistemas	Estructuras de Datos I	3X
96040121	Juarez Fernandez José	Ing. En Sistemas	Estructuras de Datos I	3X
98040151	Lopez Jimenez Carmen	Ing. En Sistemas	Matemáticas IV	4C
97040587	Alvarado Rios Jimena	Ing. Industrial	Estadística	6D
98040151	Lopez Jimenez Carmen	Ing. En Sistemas	Programación II	3W

Todas las tablas deben revisarse mediante una serie de definiciones llamada **Normalización**, para asegurarse de que cumplen con los criterios del modelo que aseguran la integridad de los datos y un buen grado de eficiencia durante la obtención de resultados.

1.4 Modelos de Datos

Modelo Entidad-Relación.

El modelo entidad relación es un modelo muy popular desde hace muchos años, describe los datos como una colección de objetos básicos.

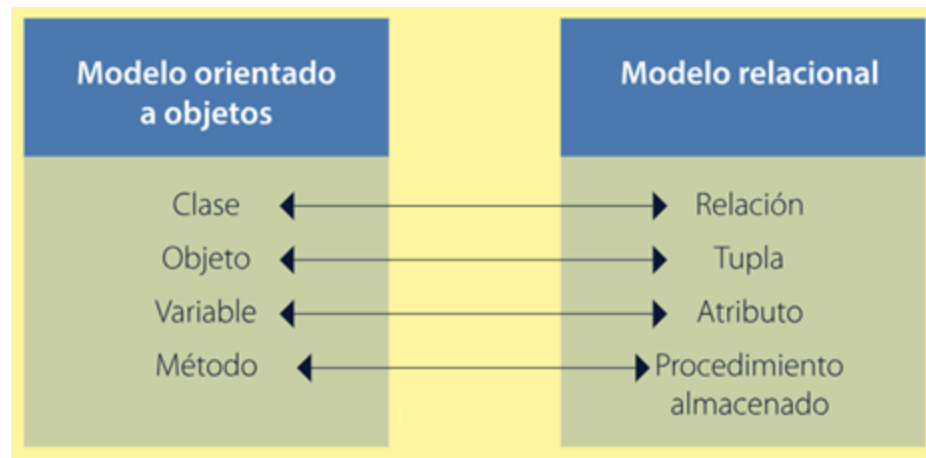


Una vez diseñada la BD, cada uno de estos objetos del diagrama se convierte a una tabla.

1.4 Modelos de Datos

Modelo orientado a objetos.

El modelo de datos orientado a objetos es una adaptación de la POO para los sistemas de bases de datos.



En los sistemas de BD que soportan estos modelos, los objetos se almacenan directamente, sin necesidad de transformar los datos en tablas.

1.5 Clasificación de Bases de Datos

Existen diversas clasificaciones de las Bases de Datos, una de ellas se refiere a la organización de los datos y tiene relación con el modelo (tema 1.4):

- Bases de Datos Relacionales (las Bases de Datos diseñadas bajo los modelos ER y Modelo Relacional, son bases de datos relacionales).
- Bases de Datos puramente orientadas a objetos.

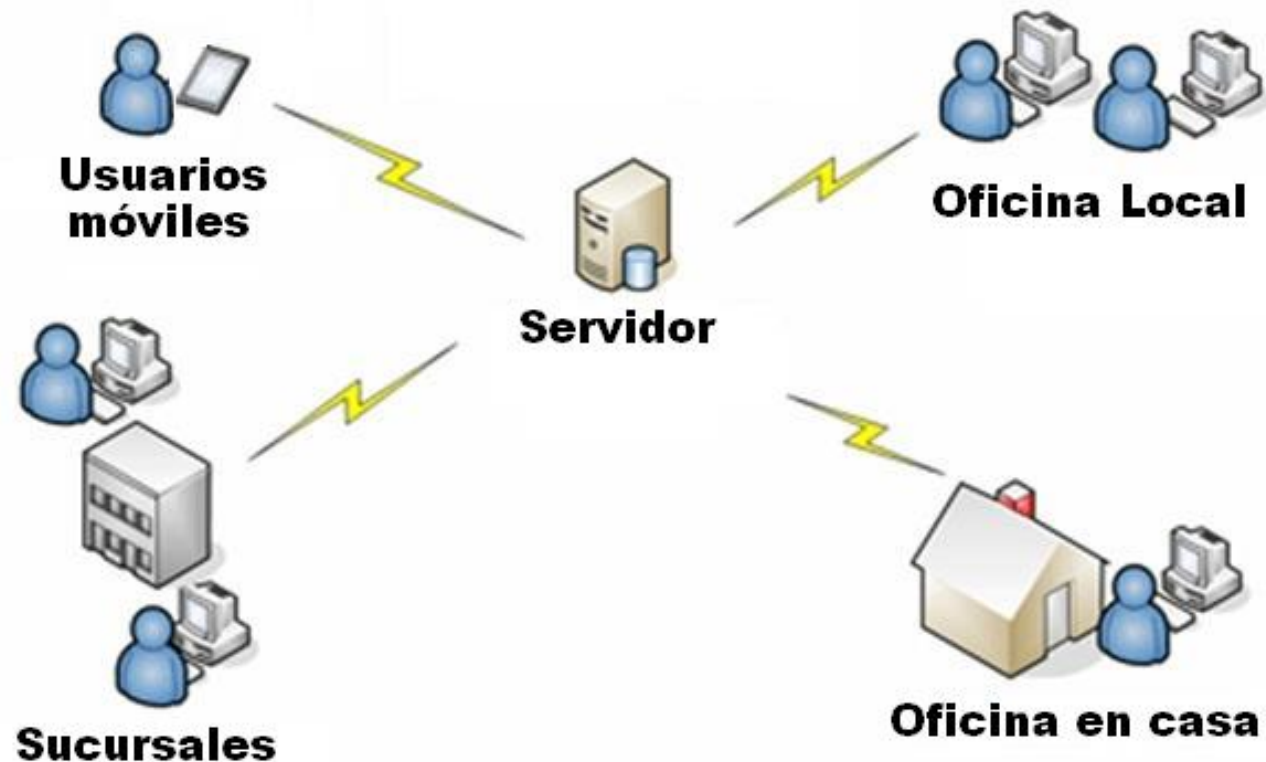
1.5 Clasificación de Bases de Datos

Hay otra clasificación respecto a la localización física de los datos:

- Bases de datos centralizadas.
- Bases de datos distribuidas.

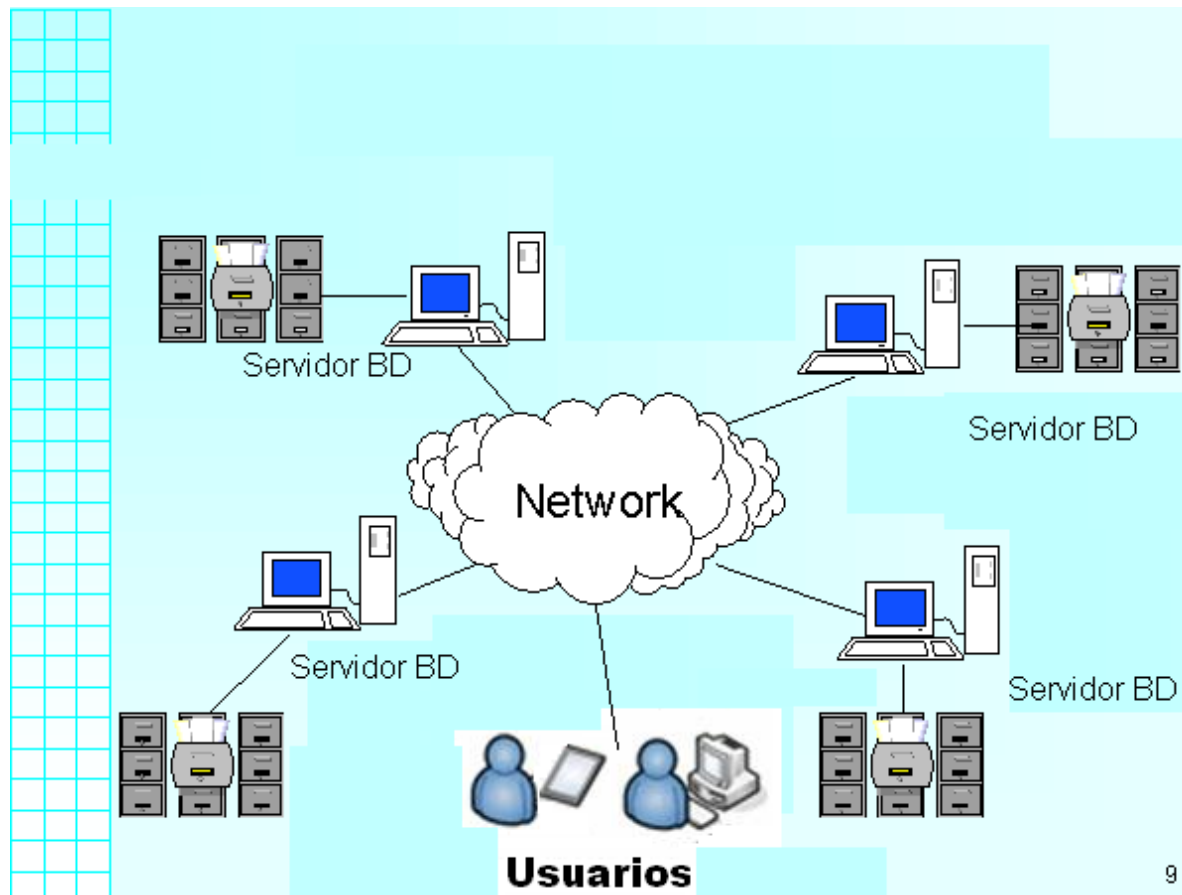
1.5 Clasificación de Bases de Datos

Bases de datos centralizadas



1.5 Clasificación de Bases de Datos

Bases de datos distribuidas



1.6 Arquitectura de un Sistema para el Manejo de Bases de Datos (DBMS)

- ◆ Consiste en los programas necesarios para facilitar la organización, almacenamiento y recuperación los datos.
- ◆ Además, contribuye a mantener la **seguridad** e **integridad** de la base de datos a pesar de las caídas del sistema o los intentos de acceso sin autorización.

1.6 Arquitectura de un Sistema para el Manejo de Bases de Datos (DBMS)

- ◆ Un **DBMS** recibe las solicitudes de datos provenientes de un comando o un programa de aplicación completo.
- ◆ Luego instruye al sistema operativo para transferir la información apropiada.
- ◆ Por lo tanto, es un intermediario.

1.1 Conceptos Básicos

Programas de Aplicación



DBMS



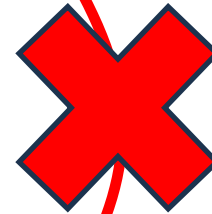
Sistema Operativo



Base de Datos



Cuando hacemos un programa en algún lenguaje y accedemos directamente a los datos, caemos en este caso.



1.6 Arquitectura de un Sistema para el Manejo de Bases de Datos (DBMS)

Analogía: DBMS vs EMPLEADO DE UNA REFACCIONARIA

Una Refaccionaria cuenta con miles de artículos, ¿qué actividades realiza el empleado?



- **Prepara el almacén** para recibir las refacciones (les asigna un lugar sin importar que haya o no existencia).
- **Colocar las refacciones** en su lugar asignado una vez que se reciben en el almacén.
- **Localiza la refacción** solicitada cuando un cliente hace una solicitud. Pasa al almacén, hace un búsqueda rápida y entrega.

1.6 Arquitectura de un Sistema para el Manejo de Bases de Datos (DBMS)

Si el **cliente** tuviera que localizar el artículo por si mismo en el almacén, tal vez **tardaría horas o días**.



DBMS

- Facilita la organización (asignación del lugar) para los datos.
- Almacena los datos en su lugar conforme llegan.
- Recupera los datos que solicitan los usuarios.

1.6 Arquitectura de un Sistema para el Manejo de Bases de Datos (DBMS)

- Un DBMS no solo debe cumplir con las tareas que hemos mencionado
- debe hacerlo eficientemente sin importar si se trata de grandes cantidades de datos.

Ejemplos de **DBMS's**:

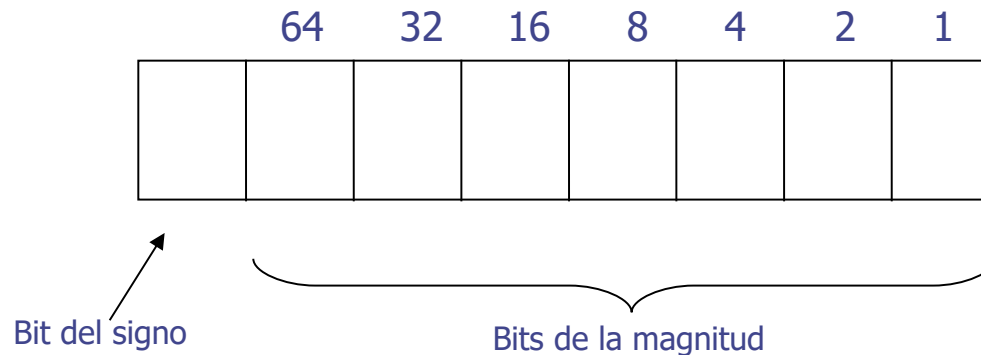


Niveles de Abstracción de las Bases de Datos

Esquemas e Instancias

Analogía: variables tipo **signed char**

Esquema:



Niveles de Abstracción de las Bases de Datos

Esquemas e Instancias

Instancias:
signed char x1,x2;

x1 = 22;

	64	32	16	8	4	2	1
	0	0	0	1	0	1	1
	0						0

x1 = -67;

	64	32	16	8	4	2	1
	1	1	0	0	0	0	1
	1						1

x2 = 8;

	64	32	16	8	4	2	1
	0	0	0	0	1	0	0
	0						0

Niveles de Abstracción de las Bases de Datos

Esquemas e Instancias

- ◆ Cada **variable** y el **dato** almacenado en un momento dado en ella es una **instancia**.
- ◆ En B.D., el término esquema se aplica al diagrama general de todos los tipos de datos y las relaciones entre ellos.
- ◆ **Sub-esquema** se refiere a la forma en la que diferentes usuarios ven la BD (aunque se trate de los mismos datos, se pueden mostrar de diferente forma de acuerdo a las capacidades del usuario).
 - Como hay una diversidad de usuarios, del mismo esquema se pueden derivar muchos **sub-esquemas** diferentes.

1.6 Arquitectura de un Sistema para el Manejo de Bases de Datos (DBMS)

Independencia de los datos

Las organizaciones son dinámicas y es indispensable hacer cambios en el esquema de la base de datos.

Independencia lógica

- Posibilidad de incorporar cambios en el esquema sin que se deban modificar los programas de aplicación.

Independencia física

- En ocasiones se requiere que la organización física de los datos se modifique, sin que se afecte al esquema ni a los programas de aplicación.

1.6 Arquitectura de un Sistema para el Manejo de Bases de Datos (DBMS)

Independencia de los datos

En general, **Independencia de Datos** es la habilidad de incorporar cambios en los niveles lógico o físico de la B.D. sin que se afecte a los niveles superiores.

Ejemplo:

- *Hay un sistema escrito en cierto lenguaje.*
- *Se guardan en un archivo binario los siguientes **datos de alumnos** de una escuela:*
 - *Nombre*
 - *Carrera que cursa*
 - *Promedio*

1.6 Arquitectura de un Sistema para el Manejo de Bases de Datos (DBMS)

Independencia de los datos

- ❑ *El sistema contiene un programa para obtener un **reporte** de los datos.*
- ❑ ***nombre** es una cadena de 45 caracteres, **carrera** es de 25. **promedio** es un dato float (6 bytes ocupados de cada registro del archivo).*
- ❑ *Ya instalado el sistema, cuenta con una buena cantidad de registros y se presentan dos situaciones independientes:*

Primer Caso:

1. *Se decide incorporar un dato nuevo para cada uno de los alumnos de la institución: **escuela de procedencia**.*

1.6 Arquitectura de un Sistema para el Manejo de Bases de Datos (DBMS)

Independencia de los datos

Caso 1

Mickey Mouse	Ingeniería Civil	95.1
Pedro Picapiedra	Arquitectura	70.0
Pato Donald	Terapia del Lenguaje	78.0
Blanca Nieves	Educación Especial	96.7
La Bella Durmiente	Costura	90.0
La Bella y la Bestia	Cirugía Plástica	85.2

```
struct datos  
{ char nombre[46];  
  char carrera[26];  
  float prom;  
};
```

Se añade el espacio para un nuevo dato



Mickey Mouse		Ingeniería Civil	95.1
Pedro Picapiedra		Arquitectura	70.0
Pato Donald		Terapia del Lenguaje	78.0
Blanca Nieves		Educación Especial	96.7
La Bella Durmiente		Costura	90.0
La Bella y la Bestia		Cirugía Plástica	85.2

PREGUNTAS

- ☐ ¿qué cambios acarrea la incorporación de la "escuela de procedencia" en el resto de los programas de aplicación, en el reporte y en la captura de los datos?
- ☐ ¿Por qué surge la necesidad de esos cambios?
- ☐ ¿qué programas especiales de mantenimiento requiere escribir el programador para que se conserven los datos previos?

SEGUNDO CASO (cambios a nivel físico)

La compañía que desarrolló el compilador del lenguaje con el que se escribió el programa, informa que el tipo de dato **float** pasará de ocupar de 4 a 6 bytes en los archivos de la BD.

- ¿Qué cambios acarrea la modificación del espacio ocupado por la variable "promedio" en el esquema y los programas de aplicación?
- ¿Se puede hacer ese cambio sin que haya otros en los programas de aplicación?

En un sistema basado en el proceso de archivos como el de este ejemplo, ¿hay independencia de datos?

Se obtiene un buen grado de independencia de datos?
eliminando de los programas de aplicación las definiciones de datos y guardándolas por separado.

Niveles de Abstracción de las Bases de Datos.

- ◆ Las Bases de Datos no se diseñan solo para usuarios especialistas.
 - Técnicos, ingenieros, programadores, diseñadores o analistas de sistemas.
- ◆ Se diseñan para cualquier persona de la organización que requiera o proporcione datos.
- ◆ Por lo tanto, la complejidad natural de los datos debe esconderse para los usuarios.
 - Incluso aquellos con experiencia.

¿Qué es la **abstracción**?

Análisis de una cosa prescindiendo de los detalles de sus componentes

"El Caballito"

Sebastián



Nulo nivel de
abstracción

"El Caballito"

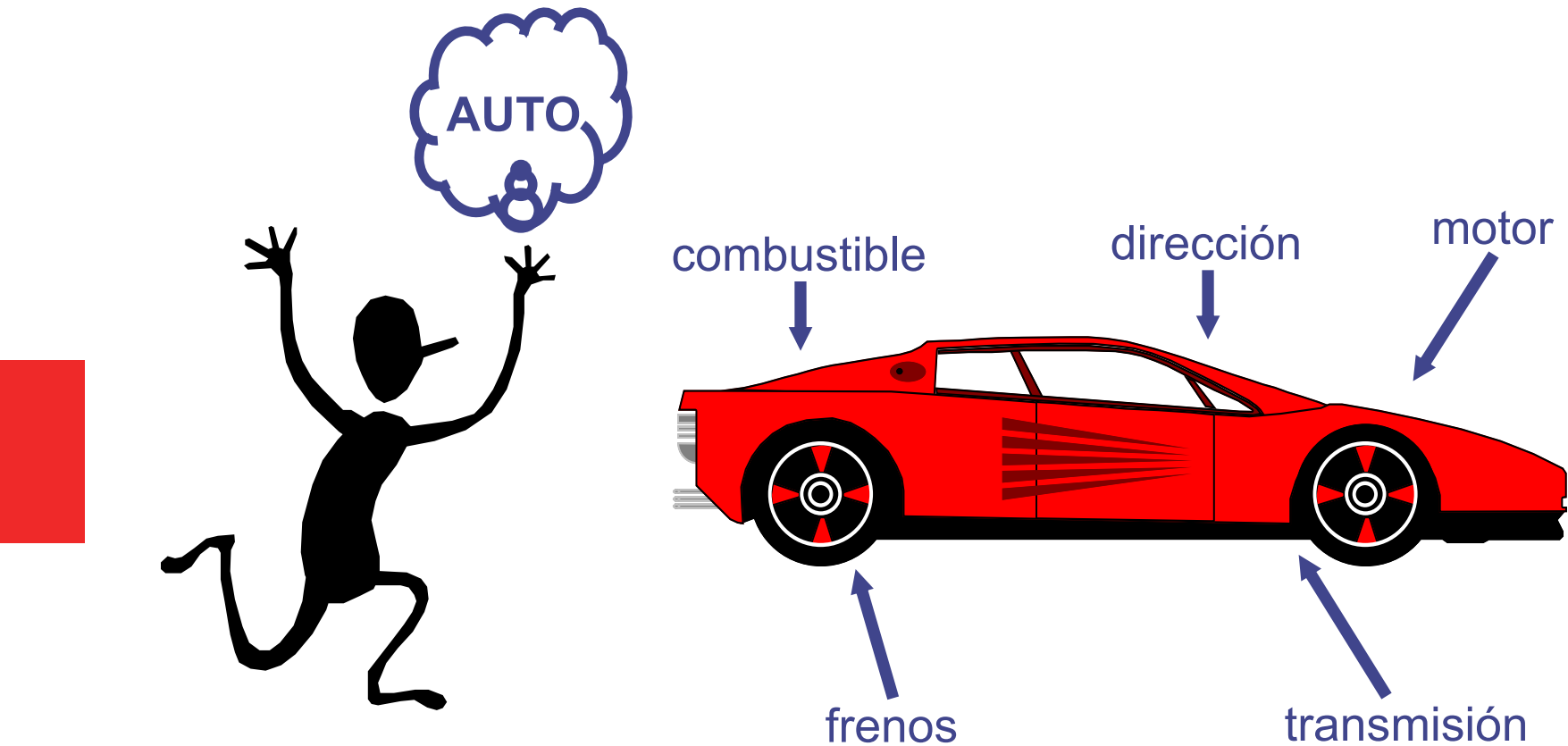


Manuel Tolsá



Alto nivel de
abstracción

Abstracción



Niveles de Abstracción de las Bases de Datos

◆ Nivel Físico.

- Los componentes más pequeños del auto.
- Solo los Ingenieros que lo diseñaron conocen su real funcionamiento.

◆ Nivel Lógico.

- Los conjuntos de componentes facilitan la **operación y comprensión**.
- Cualquier técnico mecánico puede resolver fallas reemplazando conjuntos completos (aunque desconozca su funcionamiento interno).

◆ Nivel de Vistas.

- Diferentes tipos de usuarios ven al auto de diferente manera.
- Un joven para impresionar a su novia.
- Un padre o madre para llevar a sus hijos a la escuela.
- Un taxista como su medio de trabajo.

Niveles de Abstracción de las Bases de Datos

Otra analogía: Clase Pila



Los programadores usan niveles de abstracción para esconder la complejidad y aumentar su productividad.

Niveles de Abstracción de las Bases de Datos

Clase Pila

◆ Nivel 1.

- Representación en la memoria de las estructuras de datos básicas:
 - ◆ Número de bits necesarios para un entero.
 - ◆ Método de direccionamiento de arreglos.
 - ◆ Manejo dinámico de la memoria.

◆ Nivel 2.

- Los componentes de la pila en función de las estructuras básicas del Nivel anterior:
 - ◆ Pila = arreglo+entero.
 - ◆ Pila = lista encadenada.

◆ Nivel 3.

- El programador solo indica “guardar en pila”, “retirar de pila”, etc.
- ◆ Cada nivel superior permite ignorar por completo los detalles de los niveles anteriores.



¿Que son los Niveles de Abstracción en los Sistemas de Bases de Datos?

Nivel Físico

- ◆ Es el nivel más bajo de abstracción.
- ◆ Es como se almacenan realmente los datos.
- ◆ En este nivel se describen a detalle las estructuras de datos de bajo nivel.
- ◆ Por ejemplo:
 - Archivos de acceso directo.
 - ◆ Lugar donde se guardan los datos.
 - Indexación para búsqueda eficiente.
 - ◆ Árboles B, Hashing.

Niveles de Abstracción de las Bases de Datos

Nivel Lógico o Conceptual

- ◆ Es el siguiente nivel hacia arriba desde el nivel físico.
- ◆ Describe **los datos** y las relaciones que entre ellos existen.
- ◆ Número pequeño de estructuras relativamente simples.
- ◆ No hay que preocuparse de la complejidad del nivel físico.
- ◆ Por ejemplo:
 - Para efectos de diseño y restricciones, un profesor o una materia, por ejemplo, se pueden tratar como objetos sin preocuparse de los detalles de su estructura interna.
 - Se indican en este nivel las relaciones existentes entre los diferentes objetos de la BD.
 - “Un profesor imparte una materia”.
 - Los **esquemas** describen la estructura al **nivel lógico**.

Niveles de Abstracción de las Bases de Datos

Nivel de Vistas

- ◆ Nivel más alto de abstracción.
 - ◆ Describe parte de los datos según el tipo de usuario, las diferentes descripciones son los **sub-esquemas** de la BD.
 - ◆ La simplificación del nivel lógico es insuficiente.
 - Una “aparente redundancia” puede ser deseable en este nivel para simplificación a los usuarios.
 - ◆ Permite niveles altos de seguridad, ya que los diferentes usuarios ***solo pueden ver*** o cambiar una parte de la BD.
- Ejemplo:
- Los programadores junior requieren consultar los datos de forma mas simple que los programadores expertos para minimizar el riesgo de errores.

Tipos de Lenguajes

Lenguaje de Definición de Datos (DDL)

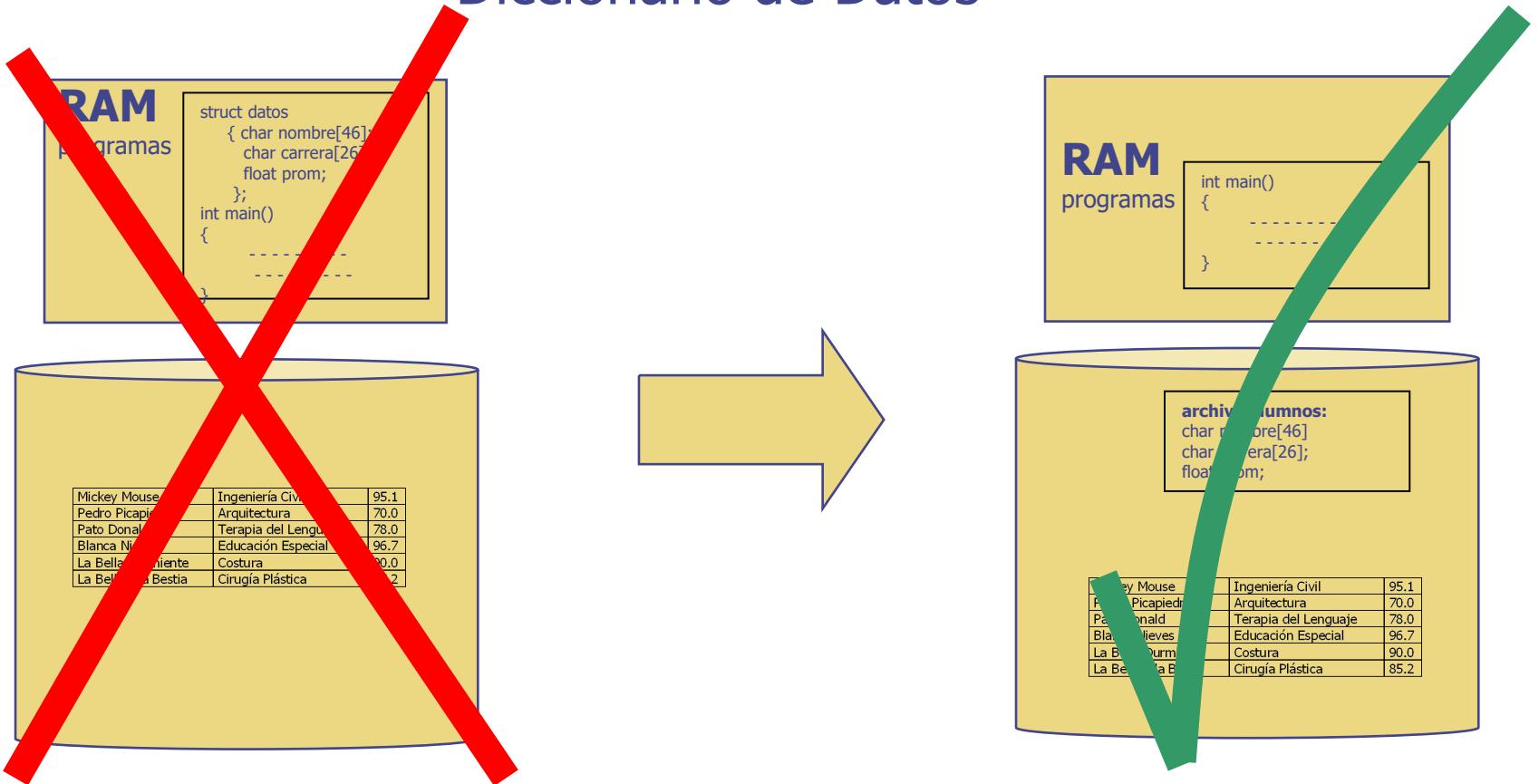
◆ DDL

- Conjunto de instrucciones y definiciones usadas para crear el diccionario de datos.
 - ◆ CREATE TABLE, ALTER TABLE, DROP TABLE, etc.

◆ Diccionario de datos (DD)

- Lugar donde se guarda el **esquema** de la base de datos.
- Puede ser uno o varios archivos.
- Contiene **metadatos**
 - ◆ Datos acerca de los datos.
- **Se consulta al leer o escribir los datos reales** en la B.D.

Tipos de Lenguajes Diccionario de Datos



La independencia de datos se obtiene con el DD

- Porque los esquemas, al no estar en los programas, solo tendrán que ser modificados una vez y **no por el programador**.
- Además los programas no tendrán que modificarse a menos que sea indispensable cuando se elimina un atributo o el cambio es de gran magnitud.

Lenguaje de Definición de Datos (DDL)

¿Qué metadatos hay en un diccionario de datos?

- Nombre de cada tabla (archivo) de la B.D.
- Detalles de cada atributo (campo):
 - Nombre, Descripción Breve, Descripción Amplia.
 - Tipo del dato correspondiente.
 - Dominio (valores válidos).
- Índices de las tablas.
 - Archivos ordenados asociados a las tablas para optimizar la velocidad de las consultas.
- Integridad Referencial.
- Disparadores (*triggers*).
- Procedimientos y Funciones almacenados.
- Columnas calculadas.
- Las diferentes Vistas de la B.D.
- Derechos de acceso de cada usuario.

Lenguaje de Manipulación de Datos (DML).

◆ DML

- Lenguaje que permite a los usuarios de una B.D. realizar los siguientes procesos:
- Recuperación de datos almacenados en la B.D.
 - ◆ **SELECT**
- Inserción de nuevos datos.
 - ◆ **INSERT.**
- Eliminación de datos.
 - ◆ **DELETE**
- Modificación de datos almacenados en la B.D.
 - ◆ **UPDATE.**

Lenguaje de Manipulación de Datos (DML).

- ◆ La manipulación de datos se clasifica de acuerdo a los niveles de abstracción.
 - Niveles **físico** y **conceptual**:
 - ◆ Se busca **acceso eficiente** a los datos para:
 - optimizar el uso del hardware.
 - que los procesos recurrentes no tomen mucho tiempo.
 - Nivel de **vistas**:
 - ◆ Se busca la **facilidad de uso** para:
 - Simplificar interacción entre los usuarios y la base de datos.
 - Obtener información inmediata sin necesidad de escribir complejos programas.
- ◆ Por lo tanto, hay **dos tipos** de ***DML's*** :

Lenguaje de Manipulación de Datos (DML).

Lenguajes Procedurales.

- ◆ Lenguaje tradicional (se indica **que** se quiere obtener y **como** obtenerlo), tomemos como ejemplo el siguiente archivo:

97040225	3608	75	EXT
97040225	3609	78	NOR
99040698	3609	85	NOR
99040698	3610	86	REG1
97040225	3611	80	NOR
97040225	3622	75	REG2
97040225	5503	70	EXT
98040150	5503	93	NOR
99040698	5503	81	EXT
98040150	5508	100	NOR
98040150	5566	92	NOR
98040150	5577	75	NOR
98040150	5578	80	REG1
99040698	6615	83	ESP

Lenguaje de Manipulación de Datos (DML).

El algoritmo siguiente calcula el promedio de las calificaciones de un alumno (98040150).

- Abrir archivo de calificaciones.
- Iniciar en cero las siguientes variables:
 - suma
 - contador
- Repetir las siguientes instrucciones para cada uno de los registros del archivo:
 - Si numero de control==98040150:
 - suma+=calificación
 - Contador++
- Al final hacer la siguiente operación:
- Promedio = suma / contador.

0	97040225	3608	75	EXT
1	97040225	3609	78	NOR
2	99040698	3609	85	NOR
3	99040698	3610	86	REG1
4	97040225	3611	80	NOR
5	97040225	3622	75	REG2
6	97040225	5503	70	EXT
7	98040150	5503	93	NOR
8	99040698	5503	81	EXT
9	98040150	5508	100	NOR
10	98040150	5566	92	NOR
11	98040150	5577	75	NOR
12	98040150	5578	80	REG1
13	99040698	6615	83	ESP

Lenguaje de Manipulación de Datos (DML).

No-procedurales.

- ☐ Solo se indica **que** se desea **pero no** se indica **como**. Es más **fácil de usar**, aunque puede generar código **menos eficiente**.
- ☐ Con el siguiente comando se obtiene el mismo resultado que en el proceso anterior:
 - "calcular el promedio de calificaciones del alumno 98040150"*
 - select avg(calific) from vCalificaciones where NumControl=98040150***

Lenguaje de Manipulación de Datos (DML).

◆ Consulta (**query**)

- solicitud para recuperar información de la base de datos.

◆ Query Lenguaje

- **QL**
- lenguaje utilizado para realizar consultas.
- **SQL** el más popular.

◆ Observación

- QL y DML **no son sinónimos.**
- QL es una parte (muy importante) de DML.
 - ◆ También es incorrecto pensar que DML es una parte de QL.

Tipos de Usuarios



**Clasificación
de acuerdo a
la forma en
que
interactúan
con la Base de
Datos**

- Programadores de aplicaciones
- Usuarios Sofisticados
- Usuarios Ingenuos
- Administradores de la Base de Datos

Tipos de Usuarios

□ Programadores de Aplicaciones.

- Crean programas de aplicación.
- Usan un lenguaje de programación tradicional o un lenguaje propio del DBMS.
 - Los lenguajes nativos de los DBMS's son similares a cualquier otro lenguaje de programación, contienen estructuras de control y de datos como cualquiera.
 - Ambos casos cuentan con una gran ventaja: instrucciones DML (lenguaje de manipulación de datos) que facilitan grandemente el proceso de los datos.

Tipos de Usuarios

❑ Usuarios Sofisticados.

- Obtienen información de la Base de Datos de manera interactiva.
 - Mediante un Lenguaje de Consultas (QL).

❑ Usuarios Ingenuos.

- La única manera de interactuar con la Base de Datos es mediante programas de Aplicación.

Tipos de Usuarios

Administrador de la Base de Datos (DBA)

- Persona (o personas, en grandes organizaciones) que ejerce el control de los datos, y los programas que acceden a ellos.
- Eventualmente puede detectar errores en el esquema y tomar las acciones necesarias para corregirlos.
- Tiene una perspectiva global de los datos de toda la organización.
- Equivale al Director General o Gerente .

Funciones del DBA

◆ Definición del Esquema, Estructuras de Almacenamiento y Métodos de Acceso

- Debe indicar los elementos principales del esquema.
 - ◆ Entidades y relaciones entre ellas.
 - ◆ Vistas, etc.
- Debe definir los detalles del esquema (muchos de los cuales se esconden al usuario).
 - ◆ Llaves.
 - ◆ Indices.
 - ◆ Detalles de los atributos (nombre, descripción, tipo, dominio, etc.).

Tipos de Usuarios

Funciones del DBA

- ◆ **Modificación del Esquema y Estructuras de Almacenamiento.**
 - Cuando el dinamismo de la organización lo amerite, realizará los cambios al esquema general o a sus detalles.
- ◆ **Concesión de autorización para el acceso a los datos.**
 - Debe otorgar o revocar derechos de acceso según el perfil de cada usuario o tipo de usuario.

Tipos de Usuarios

Funciones del DBA

◆ **Especificación de las Restricciones de Integridad.**

- Llaves únicas.
- Restricciones de dominio.
- Integridad Referencial.
- Transacciones (operaciones atómicas).
- Restricciones de acceso concurrente.

Con el objeto de evitar inconsistencia de datos.

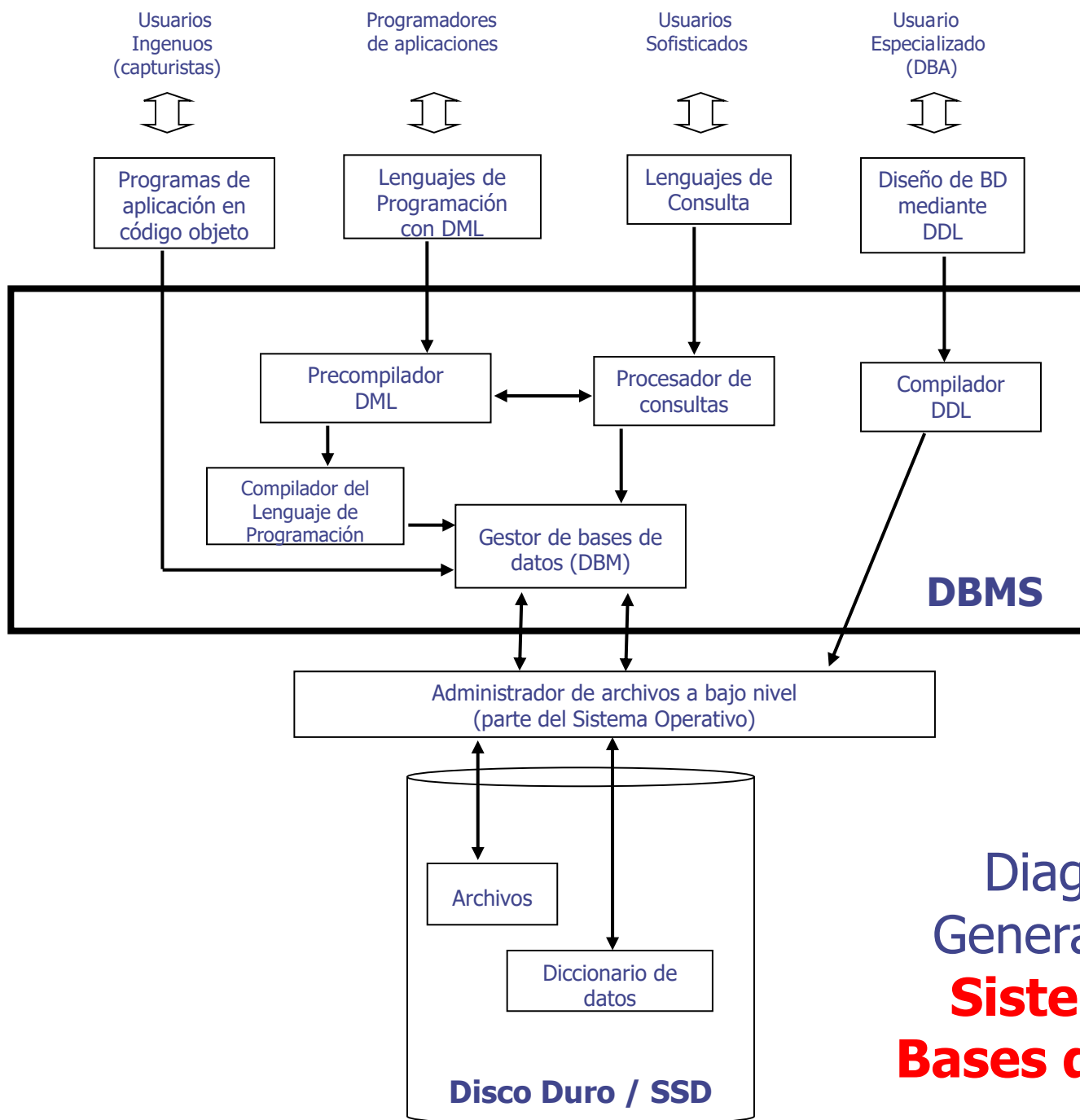


Diagrama
General de un
**Sistema de
Bases de Datos**

Todos los componentes involucrados en un Sistema de Bases de Datos son:

❑ Administrador de Archivos a bajo nivel.

- ❑ Manejo de los Archivos de datos a bajo nivel.
- ❑ Forma parte del Sistema Operativo.

❑ Gestor de Base de Datos (Database Manager DBM).

- ❑ Para ponerlo en una frase:
Es la **interfaz** entre el bajo nivel de los datos y los módulos de consultas y aplicaciones, todas las solicitudes pasan a través del DBM.

Gestor de Bases de Datos (DBM)

- ❖ DBM = DataBase Manager.
 - Parte del DBMS.
- ❖ Le llamamos **Gestor** para distinguirlo del **DBMS**.
 - Los datos se guardan en Discos.
 - ◆ Pero no son tan rápidos como la RAM.
 - **DBM** es el **programa** que, entre muchas otras cosas, organiza los datos para acceso rápido.
- ❖ **Responsabilidades**
 - **Interacción con el Gestor de Archivos del S.O.**
 - ◆ Para leer el esquema del diccionario de datos.
 - ◆ Para consultar o guardar datos.

Gestor de Bases de Datos (DBM). Responsabilidades

❖ **Conservación de la integridad**

- **Vigila** que no se violen las restricciones de Integridad.
 - En el DD deben existir metadatos de integridad

❖ **Implementación de Seguridad**

- **Aplica** los subesquemas adecuados de acuerdo al perfil del usuario
- Impide el acceso por la “puerta trasera”,
incluso a usuarios expertos.

Gestor de Bases de Datos (DBM).

❖ **Respaldo y recuperación**

- **Detecta** si hubo fallas de hardware/software y **Restaura** la BD al estado previo a la falla.

❖ **Control de concurrencia**

- **Evita** inconsistencias por accesos simultáneos.

✗ **Los programas basados en el proceso de archivos**

- × La conservación de la integridad se deja a los programas.
- × Respaldos y recuperación de datos se deja a los usuarios.

❑ **Procesador de Consultas.**

- ❑ Compilador del **QL** a instrucciones del **Gestor de Base de Datos**.
- ❑ Trata de generar el código más eficiente posible de acuerdo a la consulta de ciertos usuarios.

❑ **Precompilador de DML.**

- ❑ Convierte las instrucciones **DML** al código equivalente en el lenguaje básico.
 - ❑ Las instrucciones DML se pueden ejecutar en los programas de aplicación de los lenguajes modernos.
- ❑ Como uno de los componentes del DML es QL, el Precompilador de DML se apoya en el procesador de consultas.

❑ **Compilador DDL.**

- ❑ Convierte las instrucciones DDL en las tablas de metadatos del DD.

❑ **Archivos.** Donde físicamente se guardan:

- ❑ Los datos y las estructuras (tablas de índices) que proveen acceso rápido a ellos.
- ❑ El diccionario de datos (metadatos).

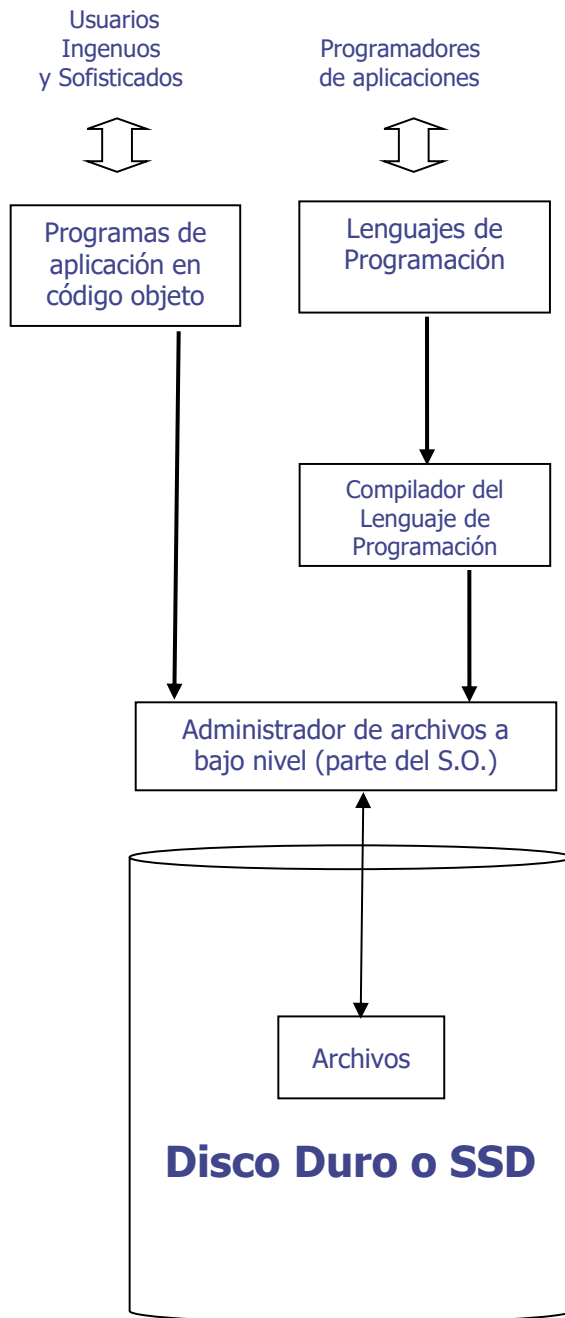


Diagrama de los sistemas basados en el **Proceso de Archivos**