

Gestión de Bases de Datos


Tema 1: Sistemas de almacenamiento de la información

Septiembre 2012

© Raúl Ruiz Padilla, Concepción Guisado Jurado, Jose
Domingo Muñoz, Septiembre de 2011

Algunos derechos reservados.
Este artículo se distribuye bajo la licencia
"Reconocimiento-CompartirIgual 3.0 España" de Creative
Commons, disponible en
<http://creativecommons.org/licenses/by-sa/3.0/es/deed.es>

Este documento (o uno muy similar)
esta disponible en (o enlazado desde)
<http://informatica.gonzalonazareno.org>



Índice

1. Bases de datos vs. Sistemas de ficheros.
2. Objetivos de las bases de datos.
3. Arquitectura en niveles de las bases de datos.
4. Componentes de las bases de datos.
5. Modelos de explotación de las bases de datos.



1. Bases de datos vs. Sistemas de Ficheros

Caso Real

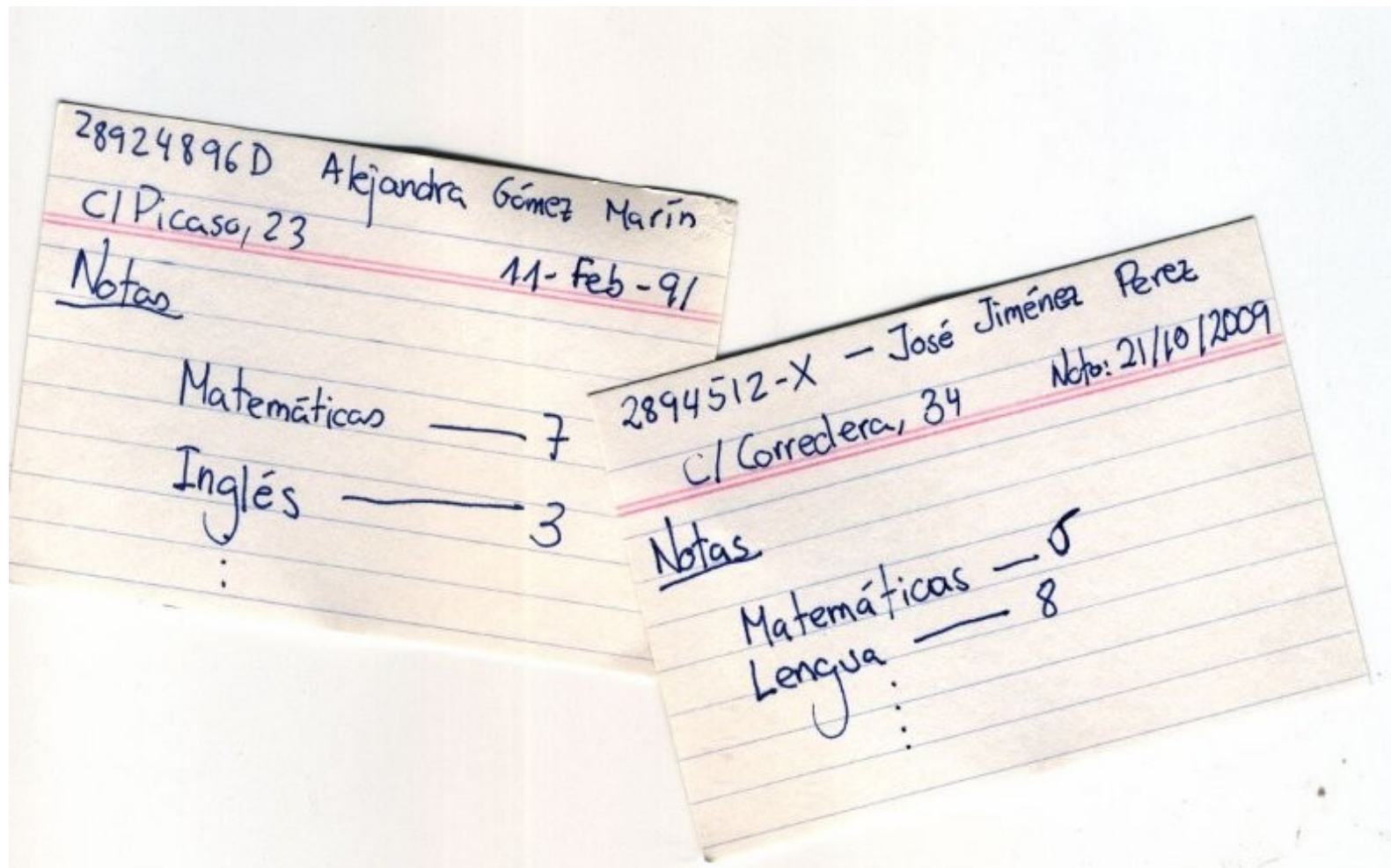
Vamos a estudiar un caso real, y vamos a ver como, posiblemente, se ha ido guardando la información que se nos plantea a lo largo de los últimos años.

Queremos guardar los datos de los alumnos de nuestro instituto, así como las asignaturas de las que están matriculados y las notas que han obtenido en cada asignatura.



1. Bases de datos vs. Sistemas de Ficheros

En los años 60, usando papel y "boli"



1. Bases de datos vs. Sistemas de Ficheros

En los años 70 y 80, usando ficheros

Se utilizaban ficheros de texto donde se guardaba la información:

alumnos.txt

DNI	NOMBRE	DIRECCIÓN	FECHA NTO
2894512X	José Jiménez Perez	C/ Corredera,34	21-10-90
28924896D	Alejandra Gómez Marín	C/ Picasso, 23	11-02-91
...			

asignaturas.txt

DNI	NOMBRE	ASIGNATURA	NOTA
2894512X	José Jiménez Perez	Matemáticas	5
2894512X	José Jiménez Perez	Lengua	8
.....			
28924896D	Alejandra Gómez Marín	Matemáticas	7
28924896D	Alejandra Gómez Marín	Inglés	3

1. Bases de datos vs. Sistemas de Ficheros

En los años 90 y posteriores..., usando Bases de Datos

Se utiliza un SGBD donde se guarda la información en las siguientes tablas:

Alumnos (DNI, Nombre, Dirección, Fecha nacimiento)

2894512X	José Jiménez Perez	C/ Corredera,34	21-10-90
28924896D	Alejandra Gómez Marín	C/ Picasso, 23	11-02-91

...

Asignaturas (Código, Nombre)

001	Matemáticas
002	Lengua
003	Inglés

Notas(DNI, Código_asignatura, nota)

2894512X	001	5
2894512X	002	8
28924896D	001	7
28924896D	003	3



1. Bases de datos vs. Sistemas de Ficheros

Evolución

Sistemas de información
no informatizados
“Lápiz y papel”

Primeros sistemas de
información
informatizados
“Ficheros”

Sistemas de información
informatizados
Bases de Datos

Estudiamos a
continuación por
qué se ha producido
esta evolución:



1. Bases de datos vs. Sistemas de Ficheros

A la hora de obtener información: consultar

Queremos obtener la siguiente información: *Se quiere conocer el número de alumnos de más de veinticinco años y con nota media superior a siete que están matriculados actualmente en la asignatura Bases de datos I.*


- S.I. Sin informatizar: Obtener esta información puede requerir mucho tiempo y mucho trabajo, además hay que realizando cálculos (media,...) e ir mirando alumno por alumno.
- S.I. con ficheros: Podemos crear un programa que vaya obteniendo la información del fichero vaya realizando los cálculos y nos de los resultados.
- S.I. Con base de datos: Esta consulta es trivial usando un lenguaje de consulta de datos.



1. Bases de datos vs. Sistemas de Ficheros

Flexibilidad a los cambios

Si las necesidades del sistema de información cambian, ¿cómo se comporta cada uno de nuestros tres modelos?. Por ejemplo queremos guardar el nombre del profesor que imparte cada asignatura.

- S.I. Sin informatizar: Tenemos que ir escribiendo el nombre de profesor en cada ficha.
 - S.I. Con ficheros: tendríamos que cambiar el fichero de notas.txt e ir escribiendo una columna más, mucho trabajo.
 - S.I. Con bases de datos: simplemente habría que añadir un atributo a la tabla asignaturas, con lo que sólo se escribiría una vez el nombre del profesor de cada asignatura.
- 


1. Bases de datos vs. Sistemas de Ficheros

Problema de la redundancia y la consistencia

La redundancia es la cantidad de datos repetidos en la información guardada. El objetivo es reducir todo lo posible la redundancia: con ello conseguimos dos cosas, que la información ocupe menos espacio y que sea lo más coherente posible.

La inconsistencia de los datos se produce cuando un dato redundante es diferente en dos o más sitios. Es el gran problema de la redundancia.

¿Cuál de los modelos presentados crees que tiene menos redundancia?




1. Bases de datos vs. Sistemas de Ficheros

Integridad de la información

La información que guardamos debe ser coherente y veraz. ¿Qué ocurriría en cada uno de los modelos presentados en los siguientes casos?

- Una persona se ha mudado y cambia su dirección.
- Nos hemos equivocado a introducir los datos de una persona y tenemos que cambiar el nombre.
- Cambiamos el nombre de una asignatura
- Desaparece una asignatura del plan de estudio

Las bases de datos aseguran automáticamente la integridad de los datos, sin que el usuario tenga que realizar ninguna operación.



1. Bases de datos vs. Sistemas de Ficheros

Concurrencia: ¿Pueden varios usuarios trabajar a la vez?

Concurrencia de usuarios: Por ejemplo tenemos tres administrativas que están trabajando con la información que tenemos guardada.

- S.I. Sin informatizar: Si por ejemplo nuestra fichas en papel están encuadernadas, es complicado que varias personas puedan trabajar al mismo tiempo con la información.
- S.I. Con ficheros: Si tenemos a las tres administrativas con programas que leen y modifican los ficheros de textos, puede ocurrir que en un determinado momento una de ellas este leyendo un dato incorrecto.
- S.I. Con base de datos: Existe el concepto de transacción, por el que se asegura que la información va a ser siempre consistente.

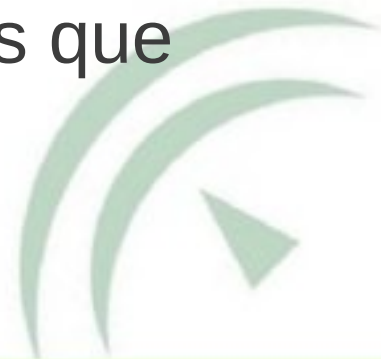
1. Bases de datos vs. Sistemas de Ficheros

Seguridad

Estamos trabajando con datos sensibles, que no todo el mundo puede tener acceso a ellos. El tema de la seguridad es muy importante en la actualidad.

Sólo determinadas personas deben poder acceder a algunas informaciones: datos personales, historial médico, historial policial, etc...

¿Cómo de seguro es cada una de los modelos que hemos estudiado?



1. Bases de Datos vs. Sistemas de Ficheros

Definición de Base de Datos y de SGBD

Una base de datos es un conjunto estructurado de datos que representa entidades y sus Interrelaciones.

Un sistema de gestión de bases de datos (SGBD) (en inglés database management system, abreviado DBMS) es una colección de datos relacionados entre si estructurados y organizados y un conjunto de programas que acceden y gestionan esos datos.



2. Objetivos de los SGBD

- Permitir consultas no predefinidas y complejas.
- Ofrecer flexibilidad e independencia de datos.
- Minimizar redundancia.
- Garantizar integridad de los datos y referencial.
- Permitir concurrencia de usuarios.
- Proporcionar seguridad de la información.



3. Arquitectura en niveles de las bases de datos.

El comité ANSI/SPARC define en 1975 una arquitectura para los sistemas gestores de bases de datos.

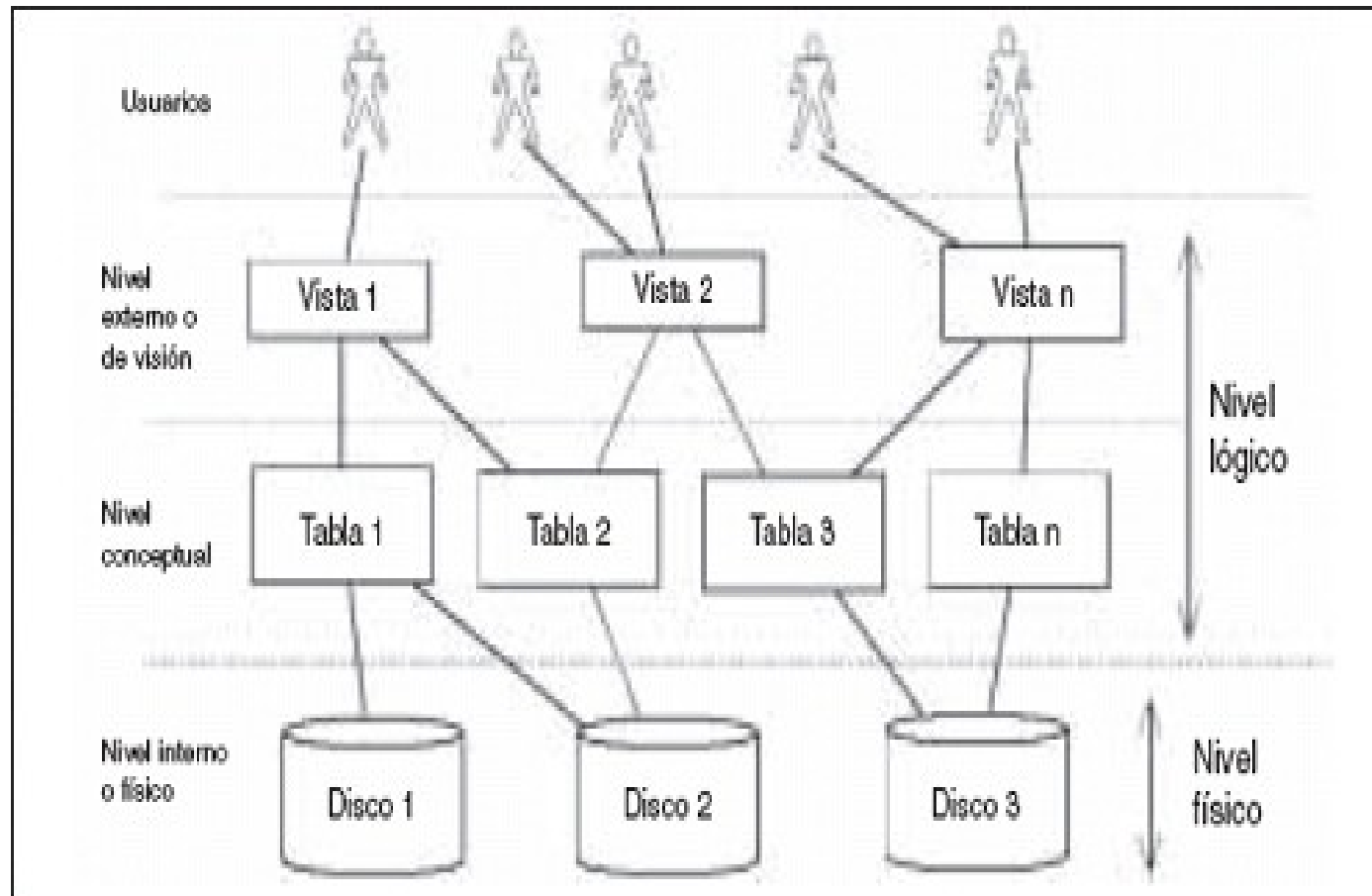
Consta de tres niveles:

- Nivel externo o de visión: Se compone de las distintas aplicaciones basadas en vistas de la base de datos. Es lo que ven los usuarios finales.
- Nivel conceptual: Se compone de las distintas tablas con sus atributos. Es el nivel que conocen los programadores.
- Nivel interno o físico: Define qué discos y archivos componen la base de datos y qué hay en cada uno de ellos. Sólo acceden a este nivel los administradores.



3. Arquitectura en niveles de las bases de datos

Veamos esta arquitectura con una imagen:



3. Arquitectura en niveles de las bases de datos.

La ventaja de esta arquitectura en niveles es que proporciona independencia lógica y física de los datos respecto a las aplicaciones:

- Independencia lógica: Se pueden realizar cambios en el nivel conceptual (añadir tablas o atributos) sin que sea necesario reescribir todas las aplicaciones.
- Independencia física: Es posible modificar la ubicación de los ficheros que contienen los datos sin que se vean afectadas las aplicaciones.



4. Componentes de un SGBD.

Los SGBD se componen de:

- Lenguajes.
- El diccionario de datos.
- Mecanismos de seguridad e integridad.
- Factor humano.

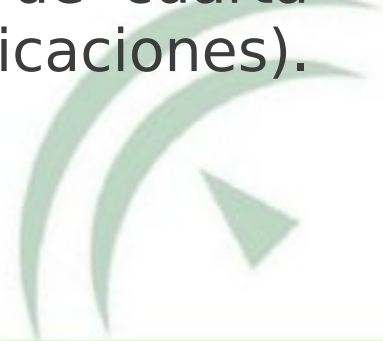
Veamos con detalle cada uno de ellos.



4. Componentes de un SGBD. Lenguajes.

Los lenguajes que tenga un SGBD deben permitir:

- Crear la estructura de la base de datos, incluyendo todos los objetos que puede incluir la misma (tablas, vistas, usuarios, procedimientos, funciones, triggers, etc.). Ej: DDL
- Consultar y manipular la información almacenada en la base de datos. Ej.: DML
- Asignar privilegios a usuarios, confirmar o abortar transacciones, etc. Ej.: DCL.
- En algunos casos, también incluyen un lenguaje de cuarta generación (4GL) para RAD (desarrollo rápido de aplicaciones). Ej: Asistentes de Access, Oracle Developer Suite



4. Componentes de un SGBD. El diccionario de datos.

El diccionario de datos contiene los metadatos (datos acerca de los datos) de la base de datos, esto es:

- La definición de todos los objetos existentes en la base de datos: tablas con sus columnas, vistas, procedimientos, triggers, índices, etc...
- La ubicación física de los objetos y el espacio asignado a los mismos.
- Los privilegios y roles asignados a los usuarios.
- Las restricciones de las tablas.
- Información de auditoría.
- Estadísticas de uso de la base de datos.
- Información del consumo de recursos actual.
- Y un larguísimo etcétera...



4. Componentes de un SGBD. Mecanismos de seguridad e integridad.

Un SGBD debe proporcionar utilidades que permitan:

- La realización de copias de seguridad de los datos y la restauración de las mismas.
- Garantizar la protección de los datos ante accesos no autorizados.
- Implantar restricciones de integridad de los datos para evitar daños accidentales de los datos.
- Recuperar la base de datos hasta un estado consistente en caso de error del sistema o cualquier otro imprevisto.
- Controlar el acceso concurrente de los usuarios para evitar errores de integridad.



4. Componentes de un SGBD. El factor humano.

Un SGBD siempre va a tener distintas categorías de usuarios:

- Usuarios finales: Podrán acceder a la información sobre la que le hayan sido concedidos privilegios.
- Programadores: Realizan aplicaciones sobre los objetos de la base de datos para facilitar su trabajo a los usuarios finales.
- Administradores o DBAs: Garantizan el correcto funcionamiento de la base de datos y gestionan todos sus recursos. Tienen el nivel más alto de privilegios y responsabilidades legales en caso de que los datos tengan algún tipo de protección. Su objetivo es que la base de datos esté siempre disponible y con un rendimiento óptimo.



5. Modelos de explotación de las bases de datos.

En nuestro entorno podemos encontrar los SGBD implantados de diferentes formas:

- Monopuesto: La base de datos se encuentra en una máquina y es explotada desde la misma máquina. Típico en SGBD de escritorio: Access, OpenBase.
 - Cliente/Servidor: El SGBD está en una máquina pero se accede a él desde muchas usando, por lo general, distintas aplicaciones.
 - Grid de servidores: La base de datos está en distintas máquinas que trabajan colaborativamente para dar servicio a los clientes.
 - BD distribuida: La información está en distintos servidores, pero no trabajan como una única máquina.
 - Capas: Cliente → Servidor web → (Servidor de aplicaciones) → Servidor de BD.
- 