



Unidad 1: Introducción

*BBDD01, Sesión 1:
Introducción a las bases de
datos.*

*Rosalía Peña
Daniel Rodríguez García
J.A. Gutiérrez de Mesa
Josefa Gómez
Dept. Ciencias de la Computación
Universidad de Alcalá*



Planificación y evaluación

■ Inscripción

- Grupo de teoría
- Grupo prácticas

■ Las transparencias no son... “lo que hay que estudiar”

- Recomendable que tengas a mano un buen libro de BDR

■ Foros de trabajo, foto

- Rosalía Peña N235 rpr@uah.es
- Gutiérrez de Mesa jantonio.gutierrez@uah.es

■ Evaluación

- 3 PEC semanas 5,10,15
- 3 PECL (posible descomposición)
- No suspender mas de dos de las 6



Rendimiento BD 2015-16

		Nº Alumnos	Nº Alumnos 1ª matrícula*	Aprobados	Suspensos	No presentados	%aprobados/ presentados
GSI	ord	54	39 (72,2%)	33 (61,1%)	13 (24,1%)	8 (14,8%)	71,7
	extraord	20	12 (60,0%)	8 (40,0%)	1 (5,0%)	11 (55,0%)	88,9
	total	54	39 (72,2%)	41 (75,9%)	1 (1,9%)	11 (20,4%)	95,3
II M		37	32 (86,5%)	30 (81,1%)	4 (10,8%)	3 (8,1%)	88,2
		7	6 (85,7%)	0 (0,0%)	1 (14,3%)	6 (85,7%)	0
		37	32 (86,5%)	30 (81,1%)	1 (2,7%)	6 (16,2%)	88,2
II Tarde		11	9 (81,8%)	8 (72,7%)	2 (18,2%)	1 (9,1%)	80
		3	2 (66,7%)	0 (0,0%)	0 (0,0%)	3 (100,0%)	0 (0%)
		11	9 (81,8%)	8 (72,7%)	0 (0,0%)	3 (27,3%)	80
IC		43	30 (69,8%)	26 (60,5%)	9 (20,9%)	8 (18,6%)	74,3
		13	8 (61,5%)	1 (7,7%)	2 (15,4%)	10 (76,9%)	33,3
		43	30 (69,8%)	27 (62,8%)	2 (4,7%)	10 (23,3%)	81,8
4º Polit		5	5 (100%)	4 (80%)	0 (0%)	1 (18,6%)	100%
		1	1(100%)	1 (0%)	0 (0%)	1 (100,9%)	
		5	5 (100%)	4 (80%)	2 (4,7%)	10 (23,3%)	100%



INDICE

■ Introducción

- Problemas con gestión de datos en ficheros
- Instancias y esquemas

■ Clasificación de BD

- Según el modelo de datos
 - **MER**
 - De registros: jerárquico, red, **relacional**
 - No relacionales: OO, clave-valor, documentales,...
- Según la organización geográfica y distribución de tareas

■ Modelo relacional

- Niveles de abstracción
- Esquema de un SGBD
 - Usuarios de un BD
 - Procesadores de I/O: LDD, LMD
 - Gestor de almacén: Memoria intermedia, gestor archivos, transacciones, autorización



Introducción

- BD: Colección de datos interrelacionados con el objetivo de **integrar** y **compartir**
- SGBD (Sistema Gestor de Base de Datos): Conjunto de programas que acceden a los datos
- SGBD + BD contienen información sobre una parcela de la realidad
- SGBD proporciona un entorno conveniente y eficiente para gestión de datos
- Aplicaciones Bases de Datos:
 - Banca. Transacciones
 - Líneas Aéreas: reserva de billetes
 - Universidades: matrículas, calificaciones, nóminas, biblioteca
 - Ventas: productos, clientes, proveedores, etc.
 - Fabricación: inventario, producción, etc.
 - Recursos Humanos: salarios, registros empleados, etc.

→ Bases de Datos toca todos aspectos de nuestras vidas.



Problemas de la organización en ficheros

■ 1950-1980? ☒ Datos sobre ficheros

■ Características:

- Redundancia de programas: altas, bajas, modificaciones y consultas
- Redundancia e **Inconsistencias** de los datos: Múltiples formatos y duplicidad de información
- Dificultad para acceder a los datos: escribir un nuevo programa para una nueva tarea
- Aislamiento de Datos: múltiples formatos y ficheros
- Problemas de integridad (reglas contenidas en el programa) :
 - Restricciones de integridad (balance de una cuenta > 0)
 - Difícil añadir restricciones o cambiarlas
- Atomicidad de las modificaciones: fallos pueden producir inconsistencias
- Concurrencia de múltiples usuarios: necesario para incrementar rendimiento y controlar el acceso
- Problemas de seguridad: Derechos de acceso

■ Solución ☒ Utilización de Sistemas de Bases de Datos



Problemas de la organización en ficheros

- La gestión de la información (casi siempre) requiere 100 % eficacia

REDUNDANCIA



INCONSISTENCIA

Objetivo del diseño:






Un sitio para cada cosa y cada cosa en un sitio

- Organización es un sistema ... por tanto... continuos cambios

INDEPENDENCIA (datos/programas)





Base de datos, esquema e instancia

- Conceptos equivalentes a variable, tipo y valor en lenguajes de programación
 - BD  contenedor de información, al igual que una variable
 - Es deseable **la** (una) BD de **la** empresa, pero un economista que realiza la nómina de distintas empresas tendría una BD para cada una de ellas
 - **Esquema**  Estructura lógica de una base de datos (recoge restricciones y semántica en el mundo real)
 - Equivalente al tipo de una variable en un programa
 - Esquema físico  cómo almacenar los datos en el disco
 - Esquema lógico  qué datos son relevantes en el universo del discurso
 - **Instancia**  el contenido de una base de datos en un instante determinado (conjunto de datos concretos que almacena)
 - Análogo al valor de una variable
 - También llamada ocurrencia, ejemplar, estado (de la BD) o instantánea
 - Una copia de seguridad (back-up) contiene una instancia de una BD
 - El Universo Discurso (UD) evoluciona en el tiempo, y con él la BD, generando instancias distintas
 - Virtualmente el número de instancias puede ser infinito.





Esquema: Diccionario de datos

- Permite al SGBD saber qué reglas debe aplicar
- La validación de las instancias se lleva a cabo mediante el esquema, almacenado en el **diccionario de datos**
- Llamado también **Meta-Base de Datos**, o **Base de Datos del Sistema**, o **Catálogo del Sistema**
- Metadatos  **Intensión** de la Base de Datos
- Instancia  **Extensión** de la Base de Datos.



Esquema (papel del..)

- Creación de base de datos: sólo se define su **esquema**, aún no tiene datos, por lo que se dice que tiene un **estado vacío** o es **instancia vacía**
- Carga de datos iniciales (tablas con datos “maestros”) 
Estado inicial o instancia inicial
- Los estados pueden ser infinitos debido a la evolución del sistema, pero siempre satisfaciendo las restricciones del UD
- El SGBD se encarga de que sólo se almacenen instancias válidas, ajustados a las reglas del esquema proporcionado
- Si las reglas son erróneas  Base de Datos contendrá estados o instancias imposibles en el mundo real (por fallos en el diseño). Ejemplo: (Sexo: Varón, nº embarazos=2).



Clasificación sistemas de Bases de Datos

- Por el nº de usuarios: monousuario, multiusuario
- Campo de aplicación:
 - Propósito general
 - Propósito específico: reserva de billetes de líneas aéreas (OLTP)
- Coste
- Por modelo de datos: relacional, objeto-relacional, jerárquico, red...
- N° de sitios en los que está dividido: centralizado, distribuido



Modelo de datos (definición y clasificación)

- Es una colección de herramientas conceptuales para describir:
 - Datos
 - Relaciones entre datos
 - La semántica de los datos
 - Restricciones de los datos
- Modelado conceptual: Entidad-Relación
- Modelos para informatizar:
 - Modelos basados en registros: (jerárquico, red ya superados, pero nivel físico), **relacional**
 - Modelo orientado a objetos
 - Modelos clave-valor
 - Modelos documentales.



Modelo entidad relación (MER)

- Modelo E-R (de UD del mundo real)
 - Entidades (objetos): clientes, cuentas
 - Atributos (características): nombre, apellidos, DNI
 - Relaciones entre entidades: cuenta pertenece a cliente

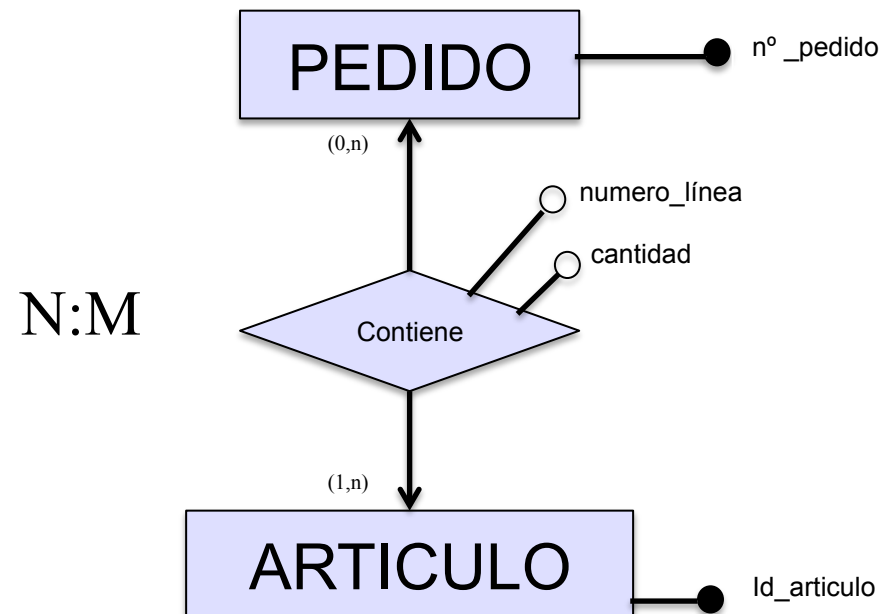
- Es un lenguaje gráfico
 - Sencillo, fácil de entender, usado incluso para hablar con nuestro cliente
 - Objetivo: tener una visión global compacta del UD
 - Estándar para diseño conceptual de bases de datos

- El diagrama E-R se convierte al modelo usado para procesamiento. En nuestro caso modelo relacional.



Modelo entidad relación (MER)

Ejemplo de esquema en el modelo entidad relación.

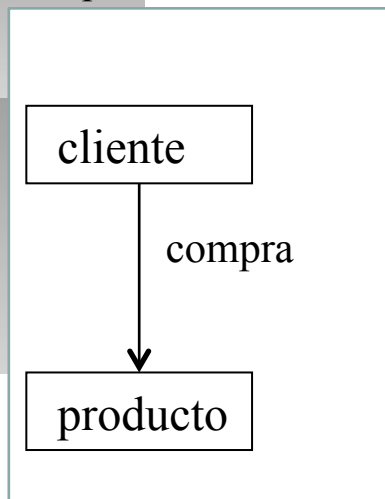




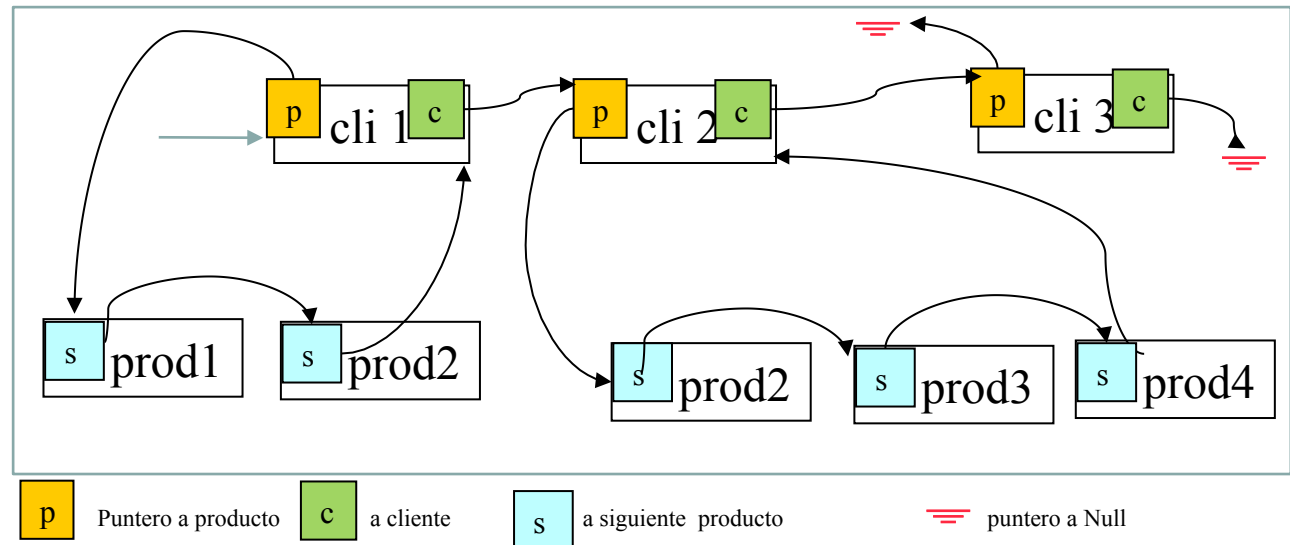
Modelo de datos jerárquico

- Modelo basado en registros para procesamiento
- Los registros se organizan como colecciones de árboles
- Establece relaciones de 1 a n

Esquema



Instancia



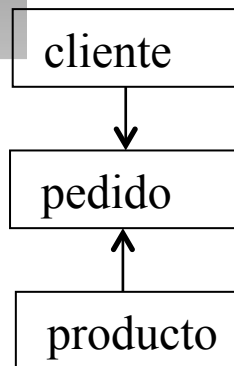
- Asimetría: ¿Qué compró cli2?/¿Quién compró prod2?
- Problemas: ¿dónde almaceno un producto aun no comprado? ¹⁵



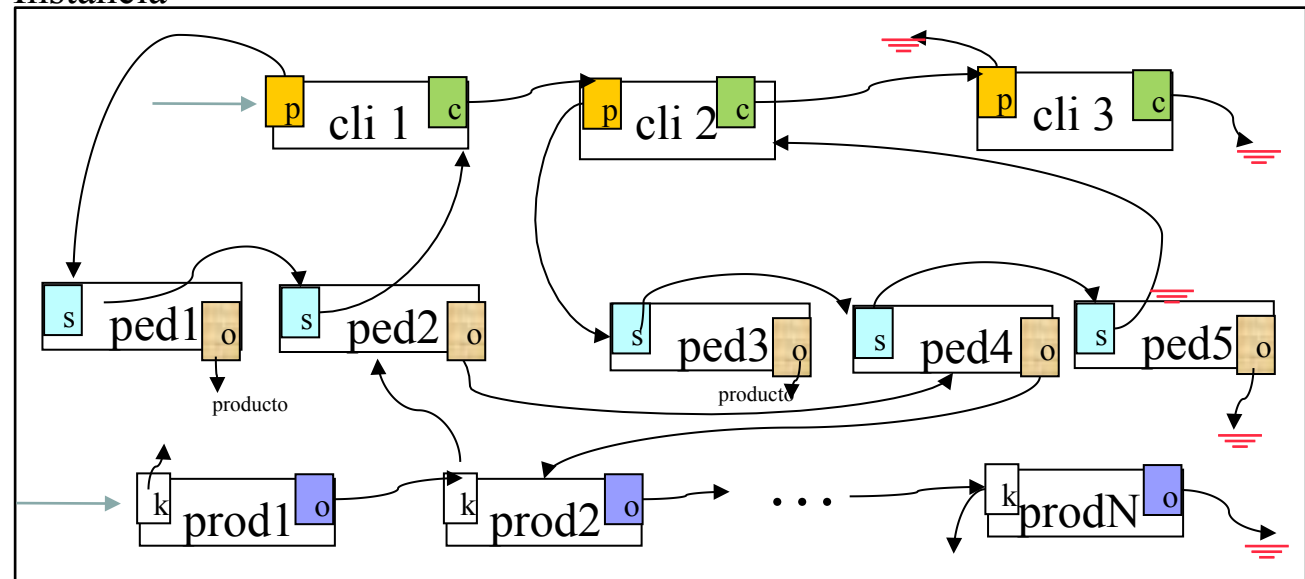
Modelo de datos red

- Se establecen grafos dirigidos, o más bien diversos árboles
- Establece relaciones de n a m, mediante un nexo (pedido en el ej)

Esquema



Instancia



Puntero a sig.cliente a sig.producto puntero a Null 1er pedido de cliente sig. pedido de cliente a "sig. pedido de producto" a 1er pedido de producto

- Simétrico
- Cada pedido 1 y sólo 1 "padre" de cada tipo
- El programador usa los punteros físicos



Modelo relacional

- Basado en registros
- Se utilizan tablas para los datos y relaciones (a nivel conceptual)
- Ejemplo de instancia en el modelo relacional

Atributo

Tabla

Ocurrencia

<i>id-cliente</i>	<i>nombre-cliente</i>	<i>calle-cliente</i>	<i>ciudad-cliente</i>
19.283.746	González	Arenal	La Granja
01.928.374	Gómez	Carretas	Cerceda
67.789.901	López	Mayor	Peguerinos
18.273.609	Abril	Preciados	Valsaín
32.112.312	Santos	Mayor	Peguerinos
33.666.999	Rupérez	Ramblas	León
01.928.374	Gómez	Carretas	Cerceda



Ejemplo de Base de datos relacional

- La relación entre tablas se establece por el valor contenido

<i>id-cliente</i>	<i>nombre-cliente</i>	<i>calle-cliente</i>	<i>ciudad-cliente</i>
19.283.746	González	Arenal	La Granja
01.928.374	Gómez	Carretas	Cerceda
67.789.901	López	Mayor	Peguerinos
18.273.609	Abril	Preciados	Valsain
32.112.312	Santos	Mayor	Peguerinos
33.666.999	Rupérez	Ramblas	León
01.928.374	Gómez	Carretas	Cerceda

(a) La tabla *cliente*

- Si hay punteros físicamente, son transparentes al programador

<i>número-cuenta</i>	<i>saldo</i>
C-101	500
C-215	700
C-102	400
C-305	350
C-201	900
C-217	750
C-222	700

(b) La tabla *cuenta*

<i>id-cliente</i>	<i>número-cuenta</i>
19.283.746	C-101
19.283.746	C-201
01.928.374	C-215
67.789.901	C-102
18.273.609	C-305
32.112.312	C-217
33.666.999	C-222
01.928.374	C-201

(b) La tabla *impositor*

¿Saldo del Cliente “Santos”?=750






Modelos de registros

- *Modelos jerárquico y red fueron desbancados por el modelo relacional a nivel lógico.*
- *A nivel físico una BD relacional puede tener subyacente uno de los anteriores*



Modelo de datos Orientado a Objetos

- Colección de Objetos  valores almacenados en variables + métodos que operan sobre ellos
- Objetos que tienen iguales valores y métodos  Se agrupan en clases
- Se accede a los valores por medio de métodos de la clase
- Cada objeto tiene identidad única independiente de los datos  dos objetos pueden tener valores iguales.



Modelo Clave-Valor

- Problema a resolver: Escalabilidad (capacidad de crecer)
→ potenciadas por la web
- Almacenamiento por clave/valor; orientado a documento
- Asegurar la integridad de datos es responsabilidad de la aplicación
- No hay un estándar, cada SG su propio lenguaje y restricciones
- Usados por grandes empresas: Facebook, Amazon, Google
- Referencias
 - <http://rafinguer.blogspot.com.es/2009/12/estan-condenadas-las-bases-de-datos.html>
 - <http://rafinguer.blogspot.com.es/2009/12/la-nueva-generacion.html>
 - <http://rafinguer.blogspot.com.es/2009/12/contentientes-en-servicios-de-nube.html>
 - <https://www.acens.com/wp-content/images/2014/02/bbdd-nosql-wp-acens.pdf>



Modelo de BD documentales

- Almacenamiento no estructurado
- Claves + pesos
- Tesauro = estructuración de claves
- Búsquedas por “semejanza” = producto escalar de vectores peso
- Ejemplo de uso: Buscadores de internet, acceso al documento en bibliotecas
- Bibliografía: Introduction to Modern information retrieval. Salton, McGill McGraw Hill 1983 ISBN 0-07-054484-0

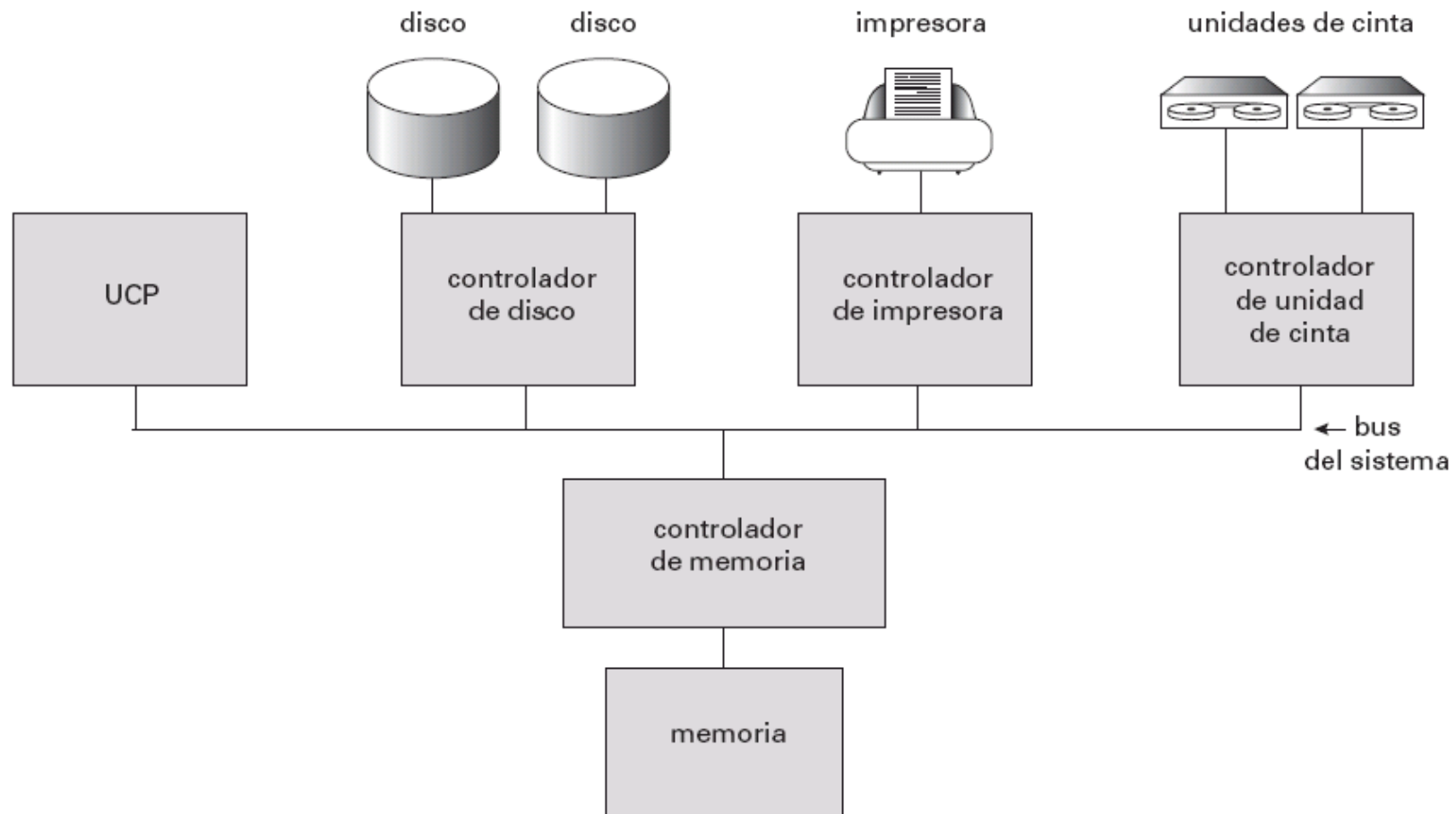


Clasificación BD por organización geográfica (Centralizado)

- El sistema de la Base de Datos se ejecuta en un único sistema informático, sin interactuar con ningún otro sistema
- Estos sistemas abarcan a los equipos monopuesto, y a los sistemas multipuesto, donde la base de datos está centralizada en un equipo
- No suelen ofrecer soluciones avanzadas en la gestión de la base de datos.




(Centralizado)



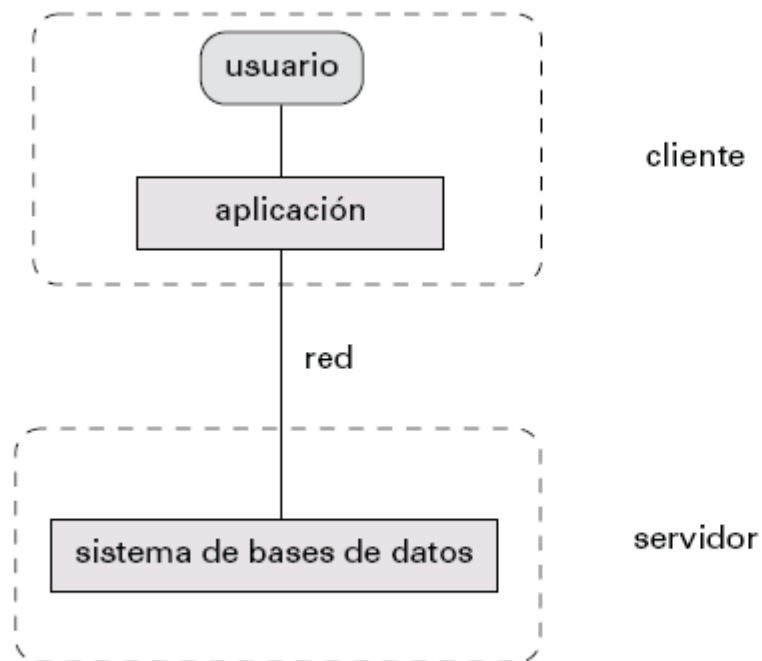


Clasificación por organización geográfica (Cliente-Servidor)

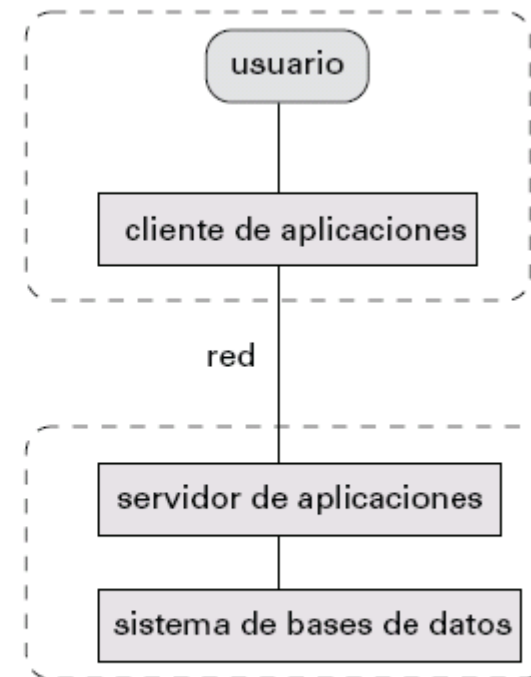
- Muy extendidos en la actualidad
- La base de datos se sitúa en un ordenador, el cuál realiza toda la gestión y almacenamiento de datos. Es el **servidor**
- Desde otros ordenadores (quizá menos potentes) mantienen y consultan los datos del servidor  proporcionando una interface amigable de acceso a datos al usuario, descargando de esta tarea al servidor. Son los **clientes**
- Sistema cliente-servidor típico: Servidores de páginas Web con conexión a base de datos.



(Cliente-Servidor)



a. arquitectura de dos capas



b. arquitectura de tres capas

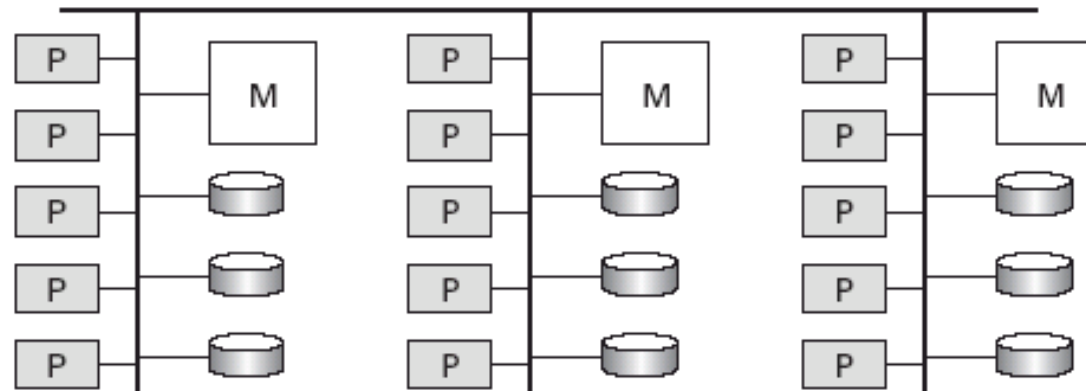


Clasificación por organización geográfica (Paralelos)

■ La BD distribuida en varios discos, gestionados por diferentes CPU en paralelo, para optimizar el rendimiento

■ Rendimiento:

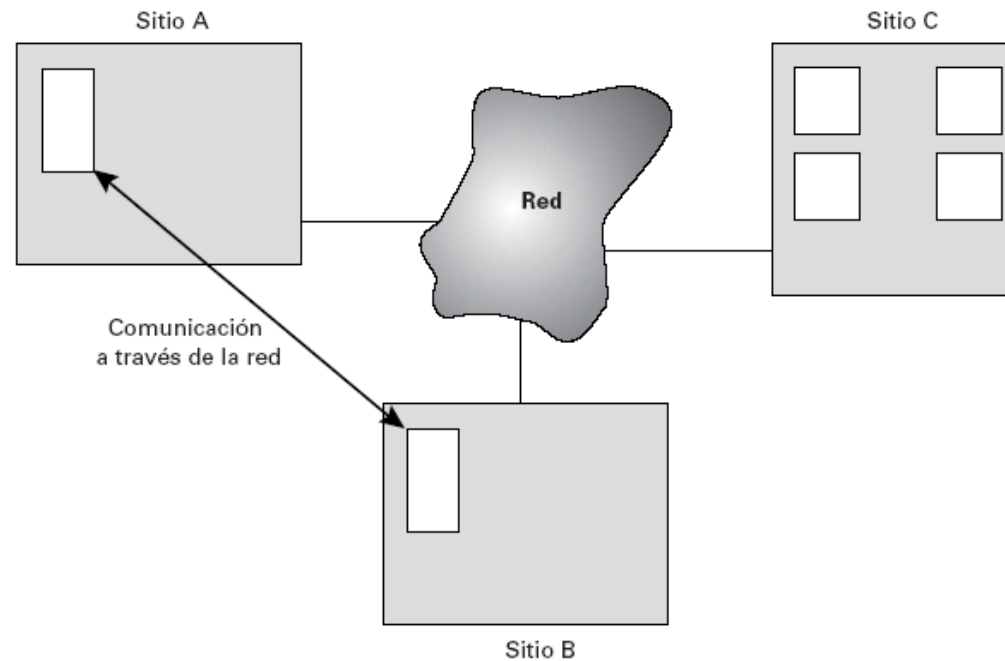
- Productividad : nº de tareas completadas en un intervalo de tiempo
- Tiempo de respuesta : tiempo en completar una tarea.





Clasificación por organización geográfica (Distribuidos)

- La información se almacena en varios ordenadores
- Los ordenadores están conectados entre sí por redes de comunicación.





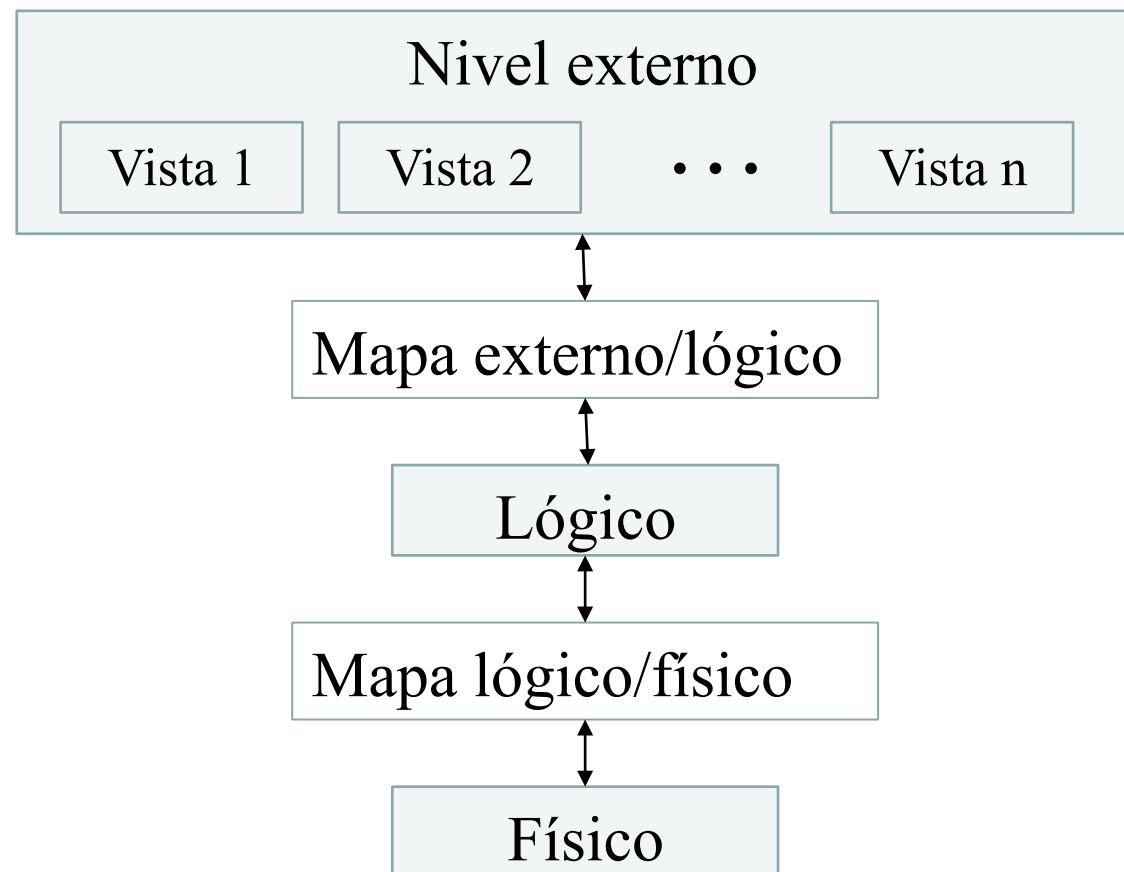
UA

Modelo relacional



Niveles de abstracción



Arquitectura de tres niveles: Abstracción de datos
(surge en BD relacionales)




independencia ↑



Niveles de abstracción

- Nivel físico  describe cómo se almacena un registro (ejemplo: cliente). Depende del SGBD y del Hw.
- Nivel lógico  describe qué datos almacena **la** base de datos y las relaciones entre ellos. Esquema global de BD

```
type cliente = record  
    nombre-cliente : string;  
    id-cliente : string;  
    calle-cliente : string;  
    ciudad-cliente : string;  
end;
```

- Nivel vistas o externo  oculta detalles de implementación
 - Programas de aplicación se escriben a este nivel
 - Cada usuario o grupos de usuarios tiene su propia vista
 - Vista solo ofrece los datos relevantes para sus usuarios (ej: matrícula no ve salarios, nóminas no ve estudiantes)



Niveles de abstracción

- Los niveles proporcionan independencia de datos y programas: No cambia el esquema sino el mapa entre dos niveles
- Ejemplos prácticos:
 1. Migrar una BD de Oracle a SQL Server.
 - No cambian: vistas, modelo lógico ni aplicaciones desarrolladas en ellas.
 - Cambia: modelo físico (de Oracle), regenerándolo para cumplir las reglas y peculiaridades del nuevo SGBD
 2. Una aplicación requiere un nuevo campo
 - La vista de esa aplicación contiene el nuevo campo
 - El modelo lógico y físico tienen un atributo mas
 - El resto de aplicaciones no se ven afectadas ya que sus esquemas externos no han cambiado.



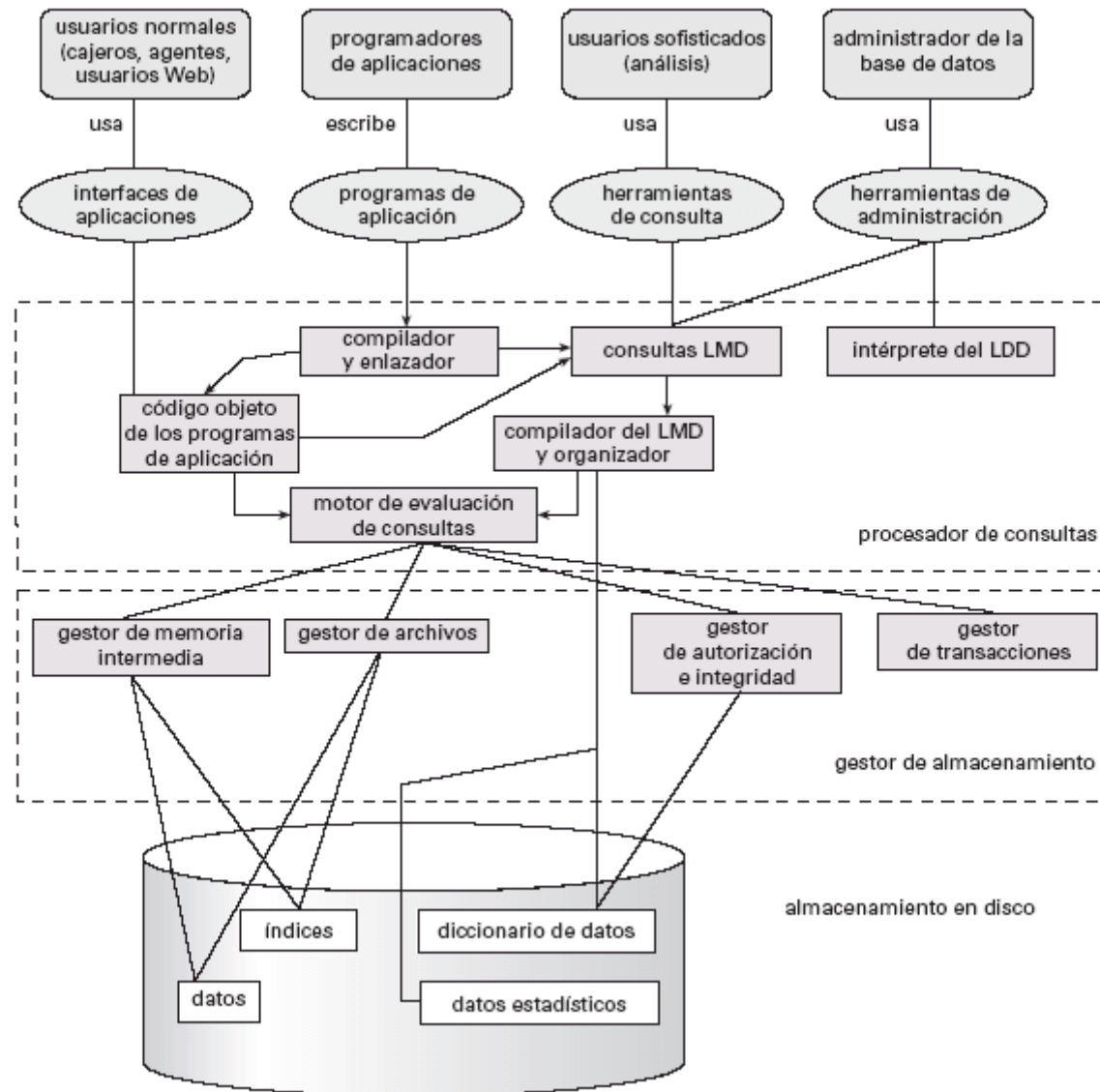
Niveles de abstracción

■ Independencia de datos:

- **Lógica:** es la capacidad de modificar el esquema lógico sin alterar los esquemas externos (solo el mapa)
- Los programas de aplicación dependen del esquema externo
→ no cambian
- **Física:** es la capacidad de modificar el esquema interno sin tener que alterar el esquema lógico
- La interfaz (o mapa) entre niveles esta definida para que cambios en una parte, no influyan en las superiores
- **Ojo:** La independencia es sólo de arriba abajo (si hay que incluir un campo en una vista, afecta al global y al físico. No afecta al resto de las vistas).



Esquema de un SGBD



Tipos de usuarios

Procesadores de I/O

Gestor de almacenamiento



Tipos de usuarios de la BD

Según la manera en que interactúen con la BD:

- Programadores de aplicaciones [W] a través de llamadas al LMD sobre otro lenguaje (anfitrión)
- Usuarios sofisticados [W] 4GL
- Usuarios especializados [W] realizan peticiones usando un lenguaje de consultas
- Usuarios normales [W] a través de aplicaciones permanentes:
 - Oficinistas, clientes que acceden a través de web o puestos de consulta. Ej: reservas aéreas, banca.
- **Administrador** de la Base de Datos (ABD) [W] control central sobre el sistema. Una o varias personas.
 - Definición del esquema
 - Definición de la estructura y métodos de acceso
 - Modificación del esquema y de la organización física
 - Concesión de la autorización para el acceso a los datos
 - Mantenimiento: Back-up, espacio en disco, supervisión, etc.



Lenguajes de un SGBD: LDD

■ Lenguaje de definición de datos. Se puede dividir en:

- LDA (Lenguaje de definición de Almacenamiento): utilizado sólo para crear el esquema
- LDV (Lenguaje de definición de vistas)

Ej: **create table** *cuenta*
(*número-cuenta* **char**(10),
saldo **integer**)

- Genera el esquema de la BD ☒ Metadatos (datos sobre los datos)
- El esquema se guarda en el catálogo de la Base de Datos.



Lenguajes de un SGBD: LMD

Lenguaje de manipulación de datos

- De Alto Nivel o no procedimental:
 - Típico lenguaje de consulta orientado a conjuntos
 - **Qué** obtener pero no **cómo** obtenerlo
 - Son declarativos
- De Bajo Nivel o procedimental:
 - Trabajan registro a registro
 - Están integrados en un lenguaje de programación de propósito general (**Lenguaje anfitrión**).
- Los LMD utilizados de forma independiente se llaman **lenguajes de consulta: 4GL**
- **SQL (Select Query Language)** es el lenguaje de consultas más utilizado. No procedimental. **Es un estándar.**



Lenguajes de un SGBD: LMD SQL

SQL, ejemplo: En la BD de la transparencia 17

- Encontrar el nombre del cliente cuyo identificador es 19.283.746

```
SELECT cliente.nombre_cliente
FROM cliente
WHERE cliente.id_cliente= '19.238.746'
```

id-cliente	nombre-cliente	calle-cliente	ciudad-cliente
19.283.746	González	Arenal	La Granja
01.928.374	Gómez	Carretas	Cerceda
67.789.901	López	Mayor	Peguerinos
18.273.609	Abril	Preciados	Valsain
32.112.312	Santos	Mayor	Peguerinos
33.666.999	Rupérez	Ramblas	León
01.928.374	Gómez	Carretas	Cerceda

(a) La tabla *cliente*

- Encontrar el saldo del cliente anterior

```
SELECT cuenta.saldo
FROM impositor, cuenta
WHERE impositor.id_cliente= '19 238 746'
AND impositor.numero_cuenta=cuenta.numero_cuenta
```

número-cuenta	saldo
C-101	500
C-215	700
C-102	400
C-305	350
C-201	900
C-217	750
C-222	700

(b) La tabla *cuenta*

id-cliente	número-cuenta
19.283.746	C-101
19.283.746	C-201
01.928.374	C-215
67.789.901	C-102
18.273.609	C-305
32.112.312	C-217
33.666.999	C-222
01.928.374	C-201

(b) La tabla *impositor*

Los programas de aplicación acceden a la base de datos por:

- Extensiones de un lenguaje de programación convencional (COBOL, PL1..) que permiten embeber SQL
- Interfaces de aplicación (ODBC/JDBC) que permiten enviar consultas SQL a la base de datos



Esquema de un SGBD: Procesadores de I/O

- Intérprete del LDD [W] interpreta las instrucciones LDD y las registra en el catálogo
- Compilador del LMD [W] traduce instrucciones del LMD a instrucciones de bajo nivel que entiende el motor de evaluación de consultas
- Precompilador del LMD [W] convierte las instrucciones del LMD en llamadas a procedimientos de un lenguaje de programación (anfitrión)
- Motor de evaluación de consultas [W] ejecuta las instrucciones de bajo nivel generadas por el compilador del LMD. Incluye **Optimizador de consultas** [W] obtiene la consulta más eficiente equivalente a la original para ser procesada posteriormente.







Esquema de un SGBD: Gestor de almacenamiento

Interface entre los datos y programas de alto nivel y consultas

Responsable del almacenamiento, recuperación y actualización de la base de datos

Componentes:

- Gestor autorización e integridad  satisface las ligaduras de integridad y la autorización de usuarios para acceder a datos
- Gestor de transacciones  asegura que la BD quede en estado consistente (correcto) a pesar de fallos en el sistema y transacciones concurrentes (ACID)
- Gestor de archivos  gestiona la reserva de espacio en disco y las estructuras de archivos empleadas para la representación de la información almacenada
- Gestor de memoria intermedia  trae los datos del disco a la memoria principal y decide qué datos tratar en la memoria cache.



Esquema de un SGBD: Herramientas de Administración

- Carga de datos de ficheros existentes
- Herramientas de conversión (importar/exportar)
- Copia de seguridad (Back-up)
- Reorganización de ficheros
- Control del rendimiento para la supervisión de la BD
- Compresión de datos
- Sistema de comunicaciones



Esquema de un SGBD: Almacenamiento

- Datos del UD
- Índices y mecanismos de acceso
- Datos estadísticos para optimización
- Catálogo o diccionario de datos.



El Diccionario de datos de un SGBD R

Almacena el esquema de la base de datos:

- Nombre de las relaciones
- Nombre de atributos
- Restricciones:
 - Dominio de datos
 - Claves candidatas y primarias
 - Claves extranjeras o foráneas
 - Valores NULL/NOT NULL
- Vistas
- Estructura de almacenamiento
- Descripción de métodos de acceso (índices/hash/punteros)
- Autorización: Usuarios/Permisos/Datos
- En sistemas avanzados también almacena:
 - Recuperación: políticas de copias de seguridad
 - Estadísticas para la optimización del SGBD
 - Funciones de usuario
 - Disparadores (triggers)



El Diccionario de datos de un SGBD R

Es una BD sobre la BD

Ejemplo: relación de catálogo que describe esquemas de relación.

REL_AND_ATTR_CATALOG

REL_NAME	ATTR NAME	ATTR_TYPE	MEMBER_OF_PK	MEMBER_OF_FK	FK_RELATION
EMPLEADO	NOMBRE	VSTR15	no	no	
EMPLEADO	INIC	CHAR	no	no	
EMPLEADO	APELLIDO	VSTR15	no	no	
EMPLEADO	NSS	STR9	sí	no	
EMPLEADO	FECHA_NCTO	STR9	no	no	
EMPLEADO	DIRECCIÓN	VSTR30	no	no	
EMPLEADO	SEXO	CHAR	no	no	
EMPLEADO	SALARIO	INTEGER	no	no	
EMPLEADO	NSS_SUPERV	STR9	no	sí	
EMPLEADO	ND	INTEGER	no	sí	EMPLEADO
DEPARTAMENTO	NOMBRED	VSTR10	no	no	DEPARTAMENTO
DEPARTAMENTO	NÚMEROD	INTEGER	sí	no	
DEPARTAMENTO	MGRSSN	STR9	no	sí	
DEPARTAMENTO	FECHA_INIC_JEFE	STR10	no	no	EMPLEADO
LOCALIZACIONES_DEPT	NÚMEROD	INTEGER	sí	sí	DEPARTAMENTO
LOCALIZACIONES_DEPT	LOCALIZACIÓN	VSTR15	sí	no	
PROYECTO	NOMBREP	VSTR10	no	no	
PROYECTO	NÚMEROP	INTEGER	sí	no	
PROYECTO	LOCALIZACIÓNP	VSTR15	no	no	
PROYECTO	NUMD	INTEGER	no	sí	
TRABAJA_EN	NSSE	STR9	sí	sí	DEPARTAMENTO
TRABAJA_EN	NP	INTEGER	sí	si	EMPLEADO
TRABAJA_EN	HORAS	REAL	no	no	PROYECTO
DEPENDIENTE	NSSE	STR9	sí	sí	
DEPENDIENTE	NOMBRE DEPENDIENTE	VSTR15	sí	no	EMPLEADO
DEPENDIENTE	SEXO	CHAR	no	no	
DEPENDIENTE	FECHA_NCTO	STR9	no	no	



The screenshot shows the pgAdmin III application window. The left pane displays a tree view of the database structure, with 'Funciones (19)' expanded under the 'public' schema. The right pane shows a list of functions, with 'pg_postmaster_starttime()' selected. Below the list, the 'Propiedades' tab is active, displaying the function's definition in SQL code.

Database Structure (Left Pane):

- Bases de Datos (1)
 - mydb
 - Castos (0)
 - Lenguajes (1)
 - plpgsql
 - Esquemas (1)
 - public
 - Agregados (0)
 - Conversiones (0)
 - Dominios (0)
 - Funciones (19)
 - Funciones Disparadoras (0)
 - Operadores (0)
 - Operador Clases (0)
 - Secuencias (0)
 - Tablas (1)
 - cities
 - Columnas (2)
 - Restricciones (0)
 - Índices (0)
 - Reglas (0)
 - Disparadores (0)
 - Tipos (0)
 - Vistas (1)

Function List (Right Pane):

Función	Comentario
database_size(name)	
pg_database_size(oid)	
pg_dir_ls(text, bool)	
pg_file_length(text)	
pg_file_read(text, int8,...)	
pg_file_rename(text, t...	
pg_file_rename(text, t...	
pg_file_stat(text)	
pg_file_unlink(text)	
pg_file_write(text, text...	
pg_logdir_ls()	
pg_postmaster_startti...	
pg_relation_size(oid)	
pg_reload_conf()	

Function Properties (Bottom Pane):

```
-- Function: pg_postmaster_starttime()

-- DROP FUNCTION pg_postmaster_starttime();

CREATE OR REPLACE FUNCTION pg_postmaster_starttime()
  RETURNS "timestamp" AS
  '$libdir/admin', 'pg_postmaster_starttime'
  LANGUAGE 'c' VOLATILE STRICT;
```

Obteniendo Función detalles... Hecho. 0.00 seg



Bibliografía

- Silverchatz
- Date
- Navate
- Concepción y diseño de bases de datos. Adoración de Miguel