

B4

Sistemas de Bases de Datos Relacionales

Tema 10. Estructura de un Sistema de Bases de Datos Relacional

10. Estructura de un SBD Relacional

1

Objetivos

- ❑ Conocer los **componentes** principales de un entorno de bases de datos
- ❑ Entender las características y propósito de la **arquitectura de tres niveles** de un sistema de bases de datos
- ❑ Aprender el significado de la **independencia de datos**
- ❑ Identificar las **funciones** más relevantes de un Sistema Gestor de Bases de Datos (SGBD) relacional
- ❑ Identificar los **componentes software** de un SGBD

10. Estructura de un SBD Relacional

2

Contenidos

- 10.1 Componentes del entorno de la base de datos
- 10.2 Arquitectura en tres niveles ANSI-SPARC
- 10.3 Independencia de datos
- 10.4 Funciones del Sistema Gestor de Bases de Datos (SGBD)
- Anexos:
 - ▣ Estructura general del SGBD
 - ▣ Ventajas y desventajas de los SGBD
 - ▣ Sistemas tradicionales de Procesamiento de Ficheros

10. Estructura de un SBD Relacional

3

Bibliografía

- [CB 2015] Connolly, T.M.; Begg C.E.: ***Database Systems: A Practical Approach to Design, Implementation, and Management***, 6th Edition. Pearson. Capítulos 1, 2 y 22.
- [EN 2016] Elmasri, R.; Navathe, S.B.: ***Fundamentals of Database Systems***, 7th Edition. Pearson. Capítulos 1, 2, 20, 21 y 22.

10.1. Componentes del entorno de la base de datos

4

□ **Hardware**

- ▣ Desde un simple PC ... a una red de ordenadores (*frontend*) con servidor central (*backend*)

□ **Software**

- ▣ SGBD + Aplicaciones + Sistema Operativo + Software de Red

□ **Datos**

- ▣ Datos + Metadatos

□ **Procedimientos**

- ▣ Instrucciones y reglas que gobiernan el diseño y uso de la base de datos y aplicaciones

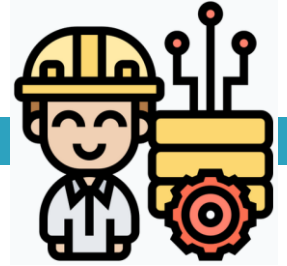
□ **Personas (actores)**

- ▣ Administrador de datos y de la base de datos, diseñadores, desarrolladores de software y usuarios finales... AMPLIEMOS...



10.1. Componentes del entorno de la base de datos: **actores**

5



Administrador de la base de datos (ABD)

- ▣ Persona (o grupo) responsable de administrar los recursos del SBD (a nivel técnico):
 - BD + SGBD + Software de aplicaciones o programas de acceso
- ▢ Las funciones del ABD incluyen (hay más):
 - ▣ Implementar la estructura de la BD y restricciones de los datos
 - ▣ Crear/Modificar estructuras de almacenamiento y métodos de acceso
 - ▣ Controlar la seguridad: conceder/denegar permisos de acceso y controlar dicho acceso a datos
 - ▣ Definir planes de copias de seguridad de los datos de la BD
 - ▣ Garantizar el funcionamiento correcto del sistema y proporcionar servicio técnico al usuario (respuesta lenta del sistema...)
 - ▣ Adquirir los recursos software y hardware necesarios

10.1. Componentes del entorno de la base de datos: actores



6

Diseñadores de bases de datos

- Antes de implementar la BD, interactúan con los futuros usuarios del sistema y...
 - ▣ Recogen y comprenden sus necesidades y objetivos
 - Requisitos**
 - ▣ Identifican **datos** que almacenar en la base de datos, las **relaciones** entre los datos, las **restricciones** que aplicar
 - Esquemas conceptuales E-R, UML, etc.
 - ▣ Eligen estructuras para representar y almacenar los datos
- Construyen...
 - ▣ **Vista** que satisface requisitos de **cada grupo de usuarios**
 - Una vista es un subconjunto de la información (de la BD)
 - ▣ **Diseño final** de BD que satisface necesidades de todos los usuarios (resultado de integrar las diferentes vistas)

10.1. Componentes del entorno de la base de datos: actores

7



Desarrolladores de Software

□ **Analistas** de Sistemas

- ▣ Determinan necesidades de procesamiento de los usuarios finales
- ▣ Especifican conjuntos de operaciones que satisfacen esas necesidades

□ **Desarrolladores** de aplicaciones

- ▣ Implementan dichas especificaciones, creando programas de aplicación mediante lenguajes de programación de 3ª o 4ª generación
- ▣ Responsables de probar, depurar, documentar y mantener los programas

10.1. Componentes del entorno de la base de datos: actores



8

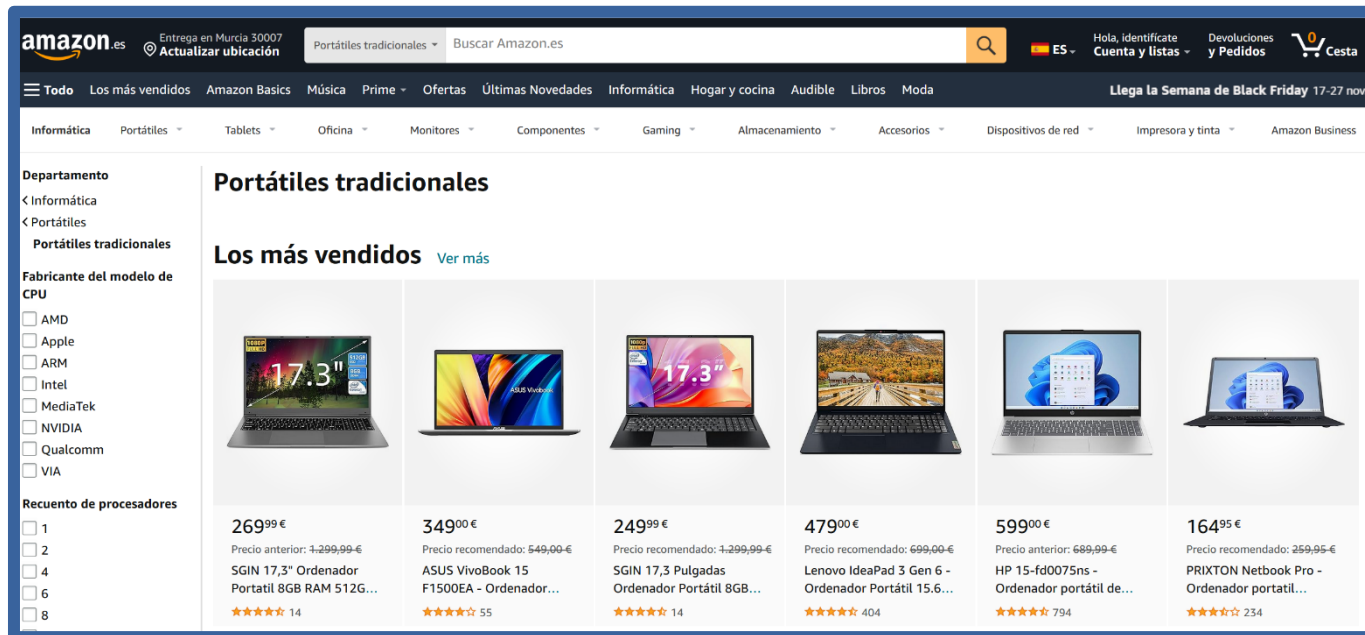
Usuarios finales

- Son los clientes: la BD se diseña, se crea, se implementa y mantiene para dar servicio a sus necesidades de información
- **Inexpertos**
 - ▣ Accesos muy frecuentes y repetitivos
 - ▣ No son conscientes de la existencia del SGBD
 - ▣ Acceden a la BD mediante programas de aplicación que facilitan sus operaciones al máximo
 - comandos simples o selección de opciones en un menú
 - Personal de caja de supermercados, o de un banco, etc.
- **Avanzados** o sofisticados
 - ▣ Familiarizados con la estructura de la base de datos y las funcionalidades ofrecidas por el SGBD
 - ▣ Pueden usar SQL para realizar sus operaciones
 - ▣ Acceso esporádico y distinto cada vez

10.2. Arquitectura de Tres Niveles ANSI-SPARC

9

- Uno de los principales objetivos de un SBD es proporcionar a los usuarios una **visión abstracta de los datos**: ocultar detalles de cómo se almacenan y manipulan los datos



10.2. Arquitectura de Tres Niveles ANSI-SPARC

10

- ❑ Pero, para conseguir una **recuperación eficiente de datos** es necesario **usar**
- 1) **estructuras de datos complejas** para representar/**almacenar** esos datos en la BD y
 - 2) **mecanismos complejos** para su manipulación

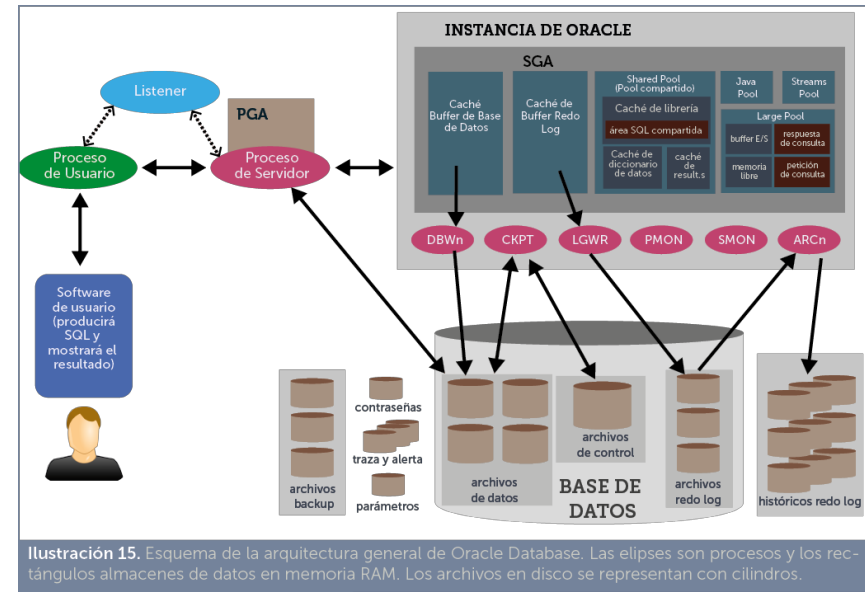


Ilustración 15. Esquema de la arquitectura general de Oracle Database. Las elipses son procesos y los rectángulos almacenes de datos en memoria RAM. Los archivos en disco se representan con cilindros.

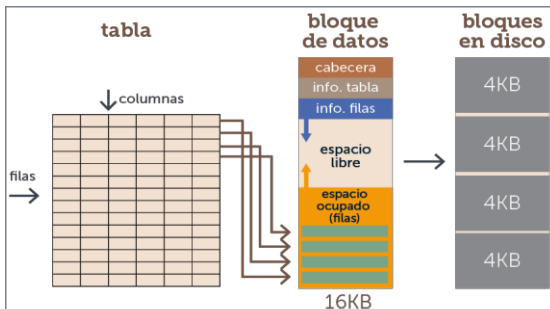


Ilustración 25. Funcionamiento del bloque de datos de Oracle, suponiendo un tamaño de bloque de 16 KB y un tamaño de bloque de Sistema Operativo de 4 KB.

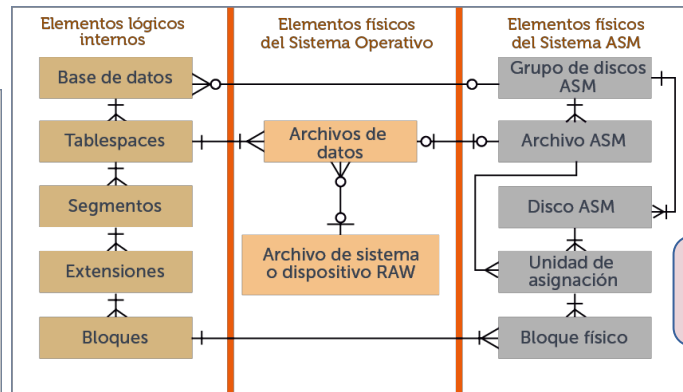


Ilustración 26. Relación entre las estructuras lógicas internas de Oracle, con los elementos físicos y los componentes físicos del sistema ASM de almacenamiento de Oracle.

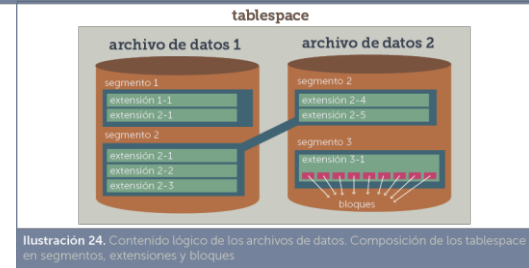


Ilustración 24. Contenido lógico de los archivos de datos. Composición de los tablespaces en segmentos, extensiones y bloques.

Argh! ¡solo soy un usuario!

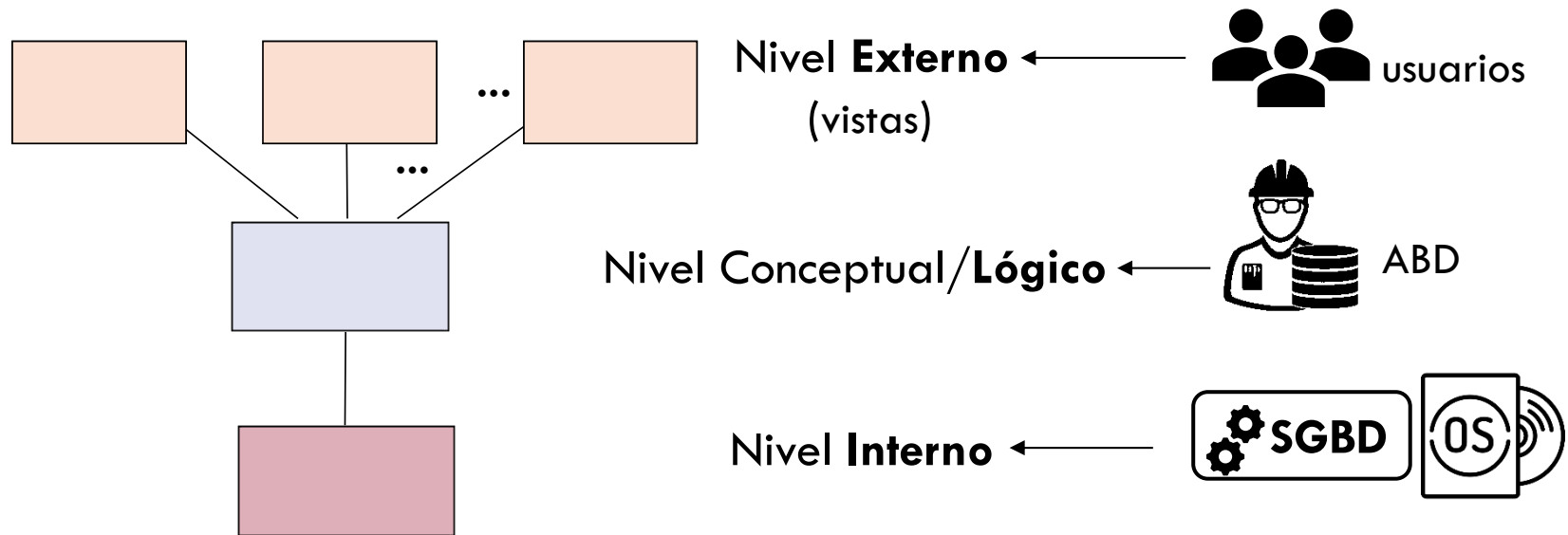
¡No debo/quiero ver esto!



10.2. Arquitectura de Tres Niveles ANSI-SPARC

11

- Esa complejidad se **oculta** mediante **niveles de abstracción**:



Arquitectura ANSI/X3/SPARC

ANSI: American National Standards Institute

X3: Database subcommittee

SPARC: Standards Planning and Requirements Committee

10.2. Arquitectura de Tres Niveles ANSI-SPARC

12

Nivel Externo

Nombre del Titular:		LISTADO DE MOVIMIENTOS			
Nº Cuenta:	ES99 9999 9999 9999 9999 9999	Fecha	Concepto	Importe	Saldo
Saldo actual :	999.999.999,99€	XX/XX/XXXX	AAAAAAA	9.999.999,99	999.999.999,99€

Nivel Conceptual / Lógico

```

CLIENTE (
  codigo NUMERO(4)      CLAVE PRINCIPAL,
  dni     CHARACTER(9)  NO NULO CLAVE,
  nombre  CHARACTER(30) NO NULO );

CUENTA (
  ccc      NUMERO(20)  CLAVE PRINCIPAL,
  titular  NUMERO(4)   NO NULO
              CLAVE AJENA A CLIENTE(codigo),
  apertura FECHA      NO NULO,
  tipo     CHARACTER(8) NO NULO,
  saldo    NUMERO(7,2) NO NULO );
  
```

```

OPERACION_CUENTA (
  numero    NUMERO(5),
  cuenta    NUMERO(20)
              CLAVE AJENA A CUENTA(ccc),
  concepto  CHARACTER(20),
  fecha     FECHA      NO NULO,
  importe   NUMERO(7,2) NO NULO,
  saldo_cuenta NUMERO(7,2) NO NULO,
  CLAVE PRINCIPAL(cuenta,numero) );
  
```

Nivel Interno

```

CLIENTE_ALMACENADO BYTES=48
PREF TIPO=BYTE(6),OFFSET=0
CLI# TIPO=BYTE(3),OFFSET=6,INDEX=ICL1
DNI  TIPO=BYTE(9),OFFSET=9,INDEX=ICL2
NOM  TIPO=BYTE(30),OFFSET=18
  
```

```

CUENTA_ALMACENADA BYTES=33
PREF TIPO=BYTE(6),OFFSET=0
CCC# TIPO=BYTE(11),OFFSET=6,INDEX=ICU1
CLI# TIPO=BYTE(3),OFFSET=17,INDEX=ICU2
FECHA TIPO=BYTE(8),OFFSET=20
SALDO TIPO=BYTE(5),OFFSET=28
  
```

```

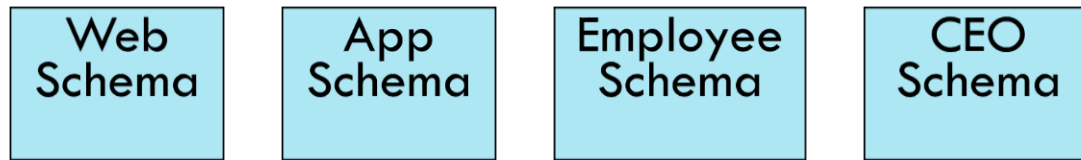
OPERACIONES_ALMACENADAS BYTES=59
PREF TIPO=BYTE(6),OFFSET=0
NUM# TIPO=BYTE(4),OFFSET=6
CCC# TIPO=BYTE(11),OFFSET=10,INDEX=IO1
CONCEPTO TIPO=BYTE(20),OFFSET=21
FECHA TIPO=BYTE(8),OFFSET=41
CANTIDAD TIPO=BYTE(5),OFFSET=49
SALDO TIPO=BYTE(5),OFFSET=54
  
```

10.2. Arquitectura de Tres Niveles ANSI-SPARC

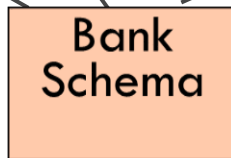
13

¿Por qué? ¿Para qué?

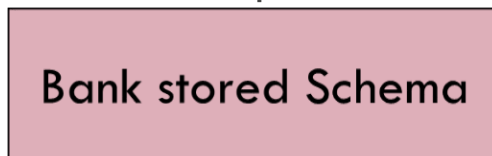
Nivel **Externo** (vistas)



Nivel **Conceptual / Lógico**



Nivel **Interno**



Cada **usuario** debe poder...
1) acceder a los datos con una **visión diferente y adaptada**
2) **cambiar** la forma en la que **ve los datos**, y que dichos cambios **no afecten a otros usuarios**

La **interacción del usuario** con la BD debe ser **independiente** de los **detalles de almacenamiento físico**

El **ABD** debe poder ...
1) **cambiar** las estructuras de **almacenamiento sin afectar a las vistas** de usuario
2) **cambiar** la **estructura conceptual/lógica** de la BD **sin afectar a todos los usuarios**

Los **cambios** en los aspectos **físicos de almacenamiento** **no deben afectar** a la estructura (esquema) **interna** de la BD

10.2. Arquitectura de Tres Niveles ANSI-SPARC

14

□ Nivel Externo o de Vistas

- ▣ Describe **qué parte** de los datos es relevante para cada (tipo de) usuario o aplicación
- ▣ Para ello usa múltiples **Esquemas Externos**
- ▣ Cada Esquema Externo (Vista) define la **porción de la BD** que interesa a cada (tipo de) usuario o aplicación
 - Para dicho usuario o aplicación, esa porción “es la base de datos”
 - Pueden existir múltiples vistas del mismo esquema conceptual
 - Varias vistas pueden solaparse entre sí, y mostrar los mismos datos de forma distinta

Web
Schema

App
Schema

Employee
Schema

CEO
Schema

10.2. Arquitectura de Tres Niveles ANSI-SPARC

15

□ Nivel Conceptual o Lógico

- ▣ Describe **qué** datos están almacenados en la BD y las relaciones entre ellos
- ▣ Para ello usa el **Esquema Conceptual** o **Lógico**, que define la estructura lógica de la BD **completa**, tal y como la ve el ABD
 - Entidades (tipos de entidad/relaciones/tablas ...), Tipos de datos,
 - Relaciones entre las entidades (tipos de relación/claves ajenas, ...),
 - Restricciones (integridad)
- ▣ Todo dato disponible en cada Esquema Externo debe estar contenido en, o ser derivado de, el nivel Conceptual/Lógico
- ▣ No contiene ningún detalle relativo al almacenamiento
- ▣ Actualmente se distinguen **2 esquemas** en este nivel:
 - Esquema **Conceptual**: organizativo, más cercano al usuario
 - Esquema **Lógico**: más cercano al SGBD

Bank
Schema

10.2. Arquitectura de Tres Niveles ANSI-SPARC

16

□ Nivel Interno

- ▣ Describe **cómo** los datos están almacenados en la BD
- ▣ Para ello usa el **Esquema Interno**, el cual define la implementación física de la BD **completa**, con el fin de optimizar el rendimiento y el uso del espacio
 - Asignación de espacio de almacenamiento para datos e índices
 - Descripción de los tipos de registros almacenados,
 - Colocación de los registros (ordenados, no ordenados...)
 - Estructuras de almacenamiento (*hashing*,...),
 - Estructuras de acceso (indexación,...)
 - etc.
- ▣ Muy cercano al *nivel físico*, pero **no** trata con registros físicos (bloques, páginas, ...) ni con unidades como cilindros o pistas.

Bank stored Schema

10.2. Arquitectura de Tres Niveles ANSI-SPARC

17

- Los 3 niveles de la A3N son descripciones de datos
- ▣ Los datos reales sólo están en el **nivel físico** (en el disco)

Los Esquemas Externos se crean usando un Modelo de Datos Conceptual (**MER**) o Lógico (**MR**)

El Esquema Conceptual/Lógico se crea usando un Modelo de Datos Conceptual (**MER**) o Lógico (**MR**)

El Esquema Interno se crea usando un **Modelo de Datos Físico**

El **nivel físico** es manejado por el SO dirigido por el SGBD

Arquitectura ANSI/X3/SPARC

Nivel Externo
(vistas)

EE1

EE2

...

EEn

Nivel Conceptual
/Lógico

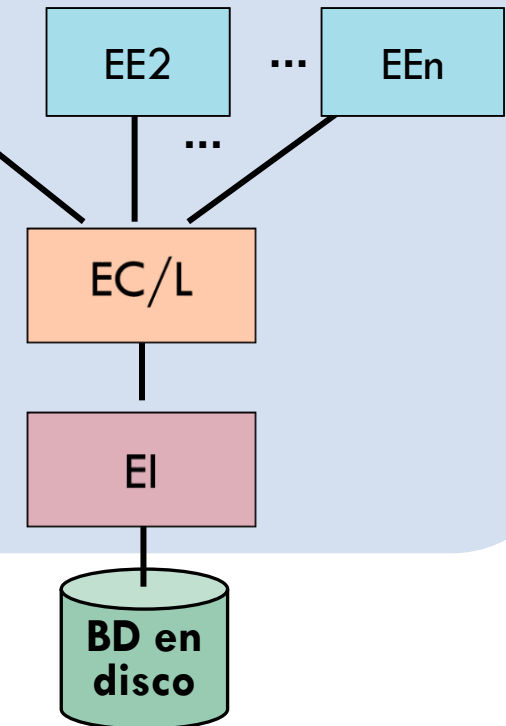
EC/L

Nivel Interno

EI

Nivel Físico

BD en
disco



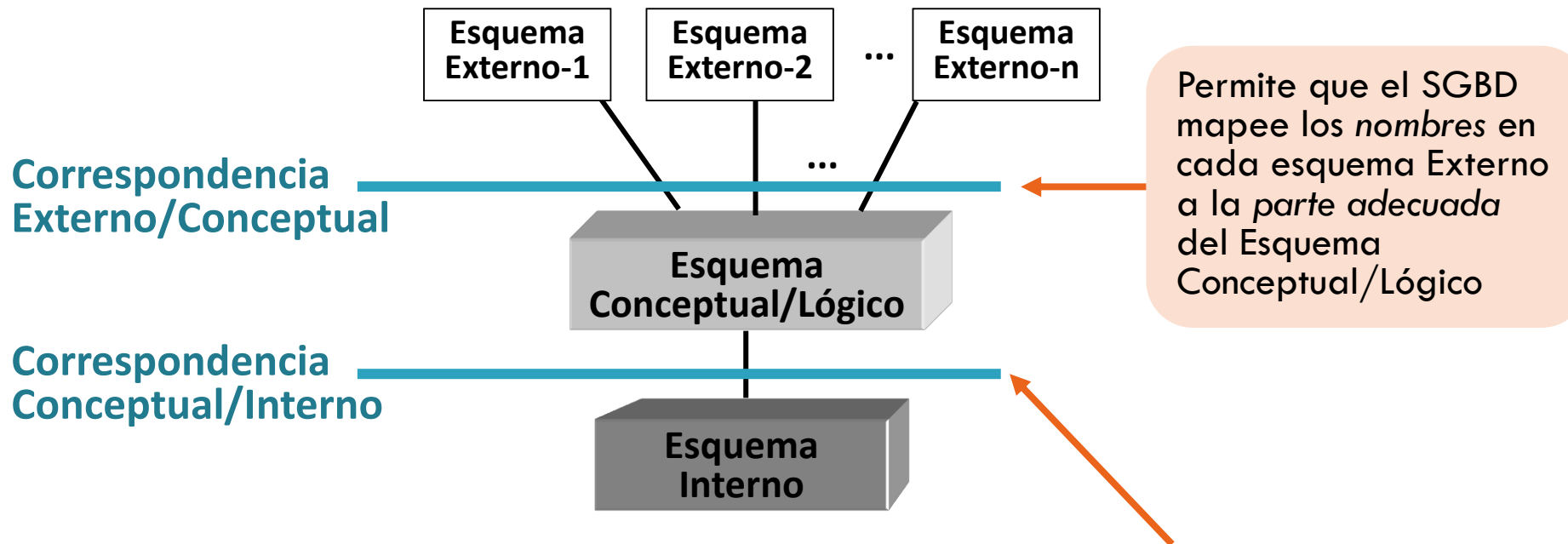
10.2. Arquitectura de Tres Niveles ANSI-SPARC

18

- El SGBD es el responsable de mantener la **correspondencia entre esquemas** a través de los niveles
 - ▣ Externo ↔ Conceptual/Lógico
 - ▣ Conceptual/Lógico ↔ Interno
- ¿Dónde están almacenadas esas correspondencias?
En el INFORMATION_SCHEMA del Catálogo del Sistema
 - ▣ Es decir, son parte de los **metadatos**
- Y debe comprobar la **consistencia** entre ellos:
 - ▣ Confirmar que cada *Esquema Externo* es derivable del *Esquema Conceptual/Lógico* (EC/L)
 - ▣ Usar la información en el *Esquema Conceptual/Lógico* para establecer una *correspondencia* entre cada *Esquema Externo* y el *Esquema Interno*

10.2. Arquitectura de Tres Niveles ANSI-SPARC

19



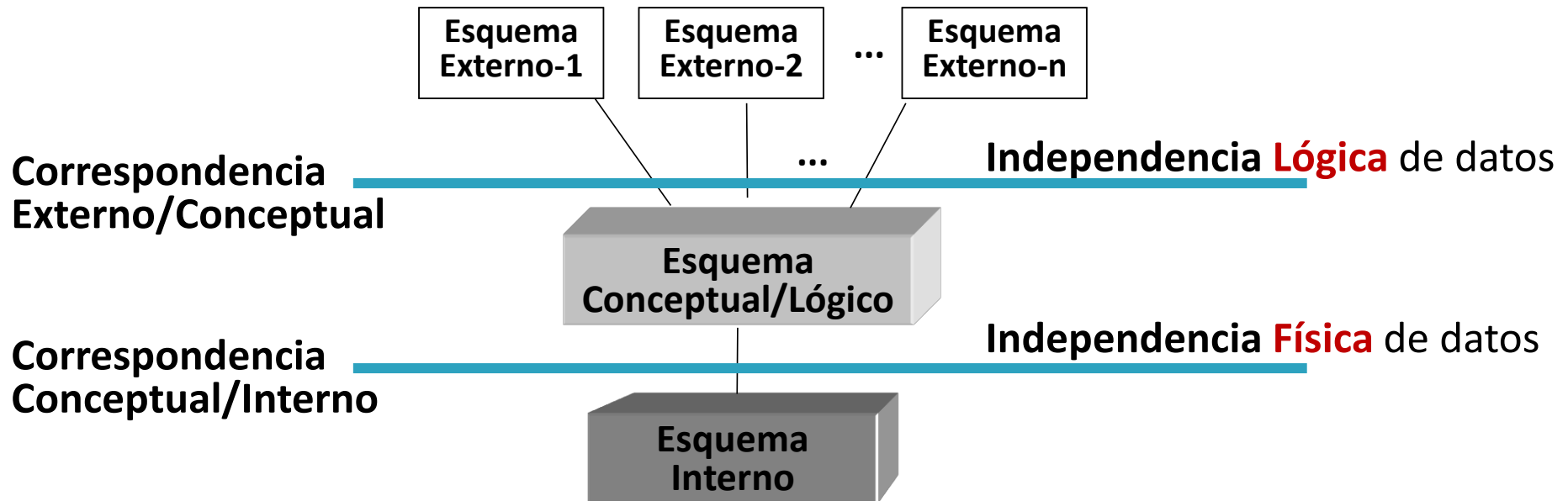
Permite que el SGBD...

- Encuentre el *registro* o *combinación de registros* en el *almacenamiento físico* que *constituyen un registro lógico* en el Esquema Conceptual/Lógico, junto con las *restricciones* que deben cumplirse en las operaciones sobre dicho registro lógico
- Resuelva cualquier *diferencia* en los nombres de los elementos de datos, de los atributos, en el orden entre atributos, los tipos de datos, etc.

10.3. Independencia de Datos

20

- « Capacidad de modificar el esquema de un nivel sin tener que cambiar el esquema del nivel inmediato superior »



10.3. Independencia de Datos

21

□ Independencia **lógica** de datos

▣ Capacidad de **modificar el Esquema Conceptual/Lógico...**

- Ampliar y/o reducir la BD: añadir/eliminar entidades/tablas y/o atributos
- Modificar restricciones

▣ **... sin alterar**

Esquemas Externos ni

Programas de aplicación

- Obviamente, los usuarios para los cuales han sido realizados los cambios sí se verán afectados, pero el resto no, y esto es lo importante

10.3. Independencia **Lógica** de datos

22

**Nivel
Externo**

Nombre del Titular:	AAAAAAAAAAAAAAAAAAAA	LISTADO DE MOVIMIENTOS			
Nº Cuenta:	ES99 9999 9999 9999 9999 9999	Fecha	Concepto	Importe	Saldo
Saldo actual :	999.999.999,99€	XX/XX/XXXX	AAAAAAA	9.999.999,99	999.999.999,99€

**Nivel
Conceptual
/ Lógico**

```

CLIENTE (
  codigo NUMERO(4)      CLAVE PRINCIPAL,
  dni    CHARACTER(9)   NO NULO CLAVE,
  nombre CHARACTER(30)  NO NULO );
  
```

```

CUENTA (
  ccc      NUMERO(20)   CLAVE PRINCIPAL,
  titular  NUMERO(4)    NO NULO
              CLAVE AJENA A CLIENTE(codigo),
  apertura FECHA        NO NULO,
  tipo     CHARACTER(8) NO NULO,
  saldo    NUMERO(7,2)  NO NULO );
  
```

```

OPERACION_CUENTA (
  numero    NUMERO(5),
  cuenta    NUMERO(20)
              CLAVE AJENA A CUENTA(ccc),
  concepto  CHARACTER(20),
  fecha     FECHA        NO NULO,
  importe   NUMERO(7,2)   NO NULO,
  saldo_cuenta NUMERO(7,2) NO NULO,
              CLAVE PRINCIPAL(cuenta,numero) );
  
```

- Esquema Externo y Esquema Lógico de partida, antes de realizar modificaciones en el Esquema Lógico

10.3. Independencia Lógica de datos

23

Nivel Externo

Nombre del Titular:	AAAAAAAAAAAAAAAAAAAA	LISTADO DE MOVIMIENTOS			
Nº Cuenta:	ES99 9999 9999 9999 9999 9999	Fecha	Concepto	Importe	Saldo
Saldo actual :	999.999.999,99€	XX/XX/XXXX	AAAAAAA	9.999.999,99	999.999.999,99€

Nivel Conceptual / Lógico

```

CLIENTE (
  codigo NUMERO(4)          CLAVE PRINCIPAL,
  dni      CHARACTER(9)      NO NULO CLAVE,
  nombre  CHARACTER(30)      NO NULO,
  apellidos CHARACTER(30)    NO NULO,
  telefono NUMERO(9)         NO NULO );
  
```

```

CUENTA (
  ccc      NUMERO(20)        CLAVE PRINCIPAL,
  titular  NUMERO(4)         NO NULO
  CLAVE AJENA A CLIENTE(codigo),
  apertura FECHA             NO NULO,
  tipo     CHARACTER(8)      NO NULO,
  saldo    NUMERO(7,2)       NO NULO,
  COMPROBAR (tipo IN ('corriente','ahorro',
                      'nómina')) );
  
```

```

OPERACION_CUENTA (
  numero    NUMERO(5),
  cuenta    NUMERO(20)
  CLAVE AJENA A CUENTA(ccc),
  concepto  CHARACTER(40),
  fecha     FECHA             NO NULO,
  importe   NUMERO(7,2)       NO NULO,
  saldo_cuenta NUMERO(7,2)    NO NULO,
  CLAVE PRINCIPAL(cuenta,numero) );
  
```

```

PRESTAMO (
  num_pre   NUMERO(20)        CLAVE PRINCIPAL,
  titular   NUMERO(4)         NO NULO
  CLAVE AJENA A CLIENTE(codigo),
  concesion FECHA             NO NULO,
  tipo_interes NUMERO(4,2)    NO NULO,
  cantidad  NUMERO(7,2)       NO NULO );
  
```

□ **El Esquema Externo permanece inalterado**

▣ Sólo hay que modificar la **correspondencia** entre esquemas:

■ “Nombre del Titular” ahora necesita concatenar los valores de las columnas “nombre” y “apellidos” de CLIENTE

10.3. Independencia de Datos

24

□ Independencia **física** de datos

▣ Capacidad de **modificar el Esquema Interno...**

■ Reestructurar alguna estructura de almacenamiento

- Cambiar la organización de un fichero (por ejemplo, de no ordenado (*heap*, montón) a ordenado)
- Mover a otro dispositivo de almacenamiento
- Modificar un índice
- Cambiar la función o algoritmo *hashing*

■ Crear nuevas estructuras de acceso

- Crear un nuevo índice

▣ ... **sin alterar**

**El Esquema Conceptual / Lógico ni
Programas de aplicación**

- El único efecto que pueden notar los usuarios será un cambio (mejora) en el rendimiento

10.3. Independencia **Física** de datos

25

Nivel Conceptual / Lógico

```
CLIENTE (  
  codigo NUMERO(4)    CLAVE PRINCIPAL,  
  dni    CHARACTER(9) NO NULO CLAVE,  
  nombre CHARACTER(30) NO NULO );
```

```
CUENTA (  
  ccc      NUMERO(20)  CLAVE PRINCIPAL,  
  titular  NUMERO(4)   NO NULO  
           CLAVE AJENA A CLIENTE(codigo),  
  apertura FECHA       NO NULO,  
  tipo     CHARACTER(8) NO NULO,  
  saldo    NUMERO(7,2) NO NULO );
```

```
OPERACION_CUENTA (  
  numero    NUMERO(5),  
  cuenta    NUMERO(20)  
           CLAVE AJENA A CUENTA(ccc),  
  concepto  CHARACTER(20),  
  fecha     FECHA      NO NULO,  
  importe   NUMERO(7,2) NO NULO,  
  saldo_cuenta NUMERO(7,2) NO NULO,  
           CLAVE PRINCIPAL(cuenta,numero) );
```

Nivel Interno

```
CLIENTE_ALMACENADO BYTES=48  
  PREF TIPO=BYTE(6),OFFSET=0  
  CLI# TIPO=BYTE(3),OFFSET=6,INDEX=ICL1  
  DNI  TIPO=BYTE(9),OFFSET=9,INDEX=ICL2  
  NOM  TIPO=BYTE(30),OFFSET=18,INDEX=ICL3
```

```
CUENTA_ALMACENADA BYTES=33  
  PREF TIPO=BYTE(6),OFFSET=0  
  CCC# TIPO=BYTE(11),OFFSET=6,INDEX=ICU1  
  CLI# TIPO=BYTE(3),OFFSET=17,INDEX=ICU2  
  FECHA TIPO=BYTE(8),OFFSET=20  
  SALDO TIPO=BYTE(5),OFFSET=28
```

```
OPERACIONES_ALMACENADAS BYTES=59  
  PREF TIPO=BYTE(6),OFFSET=0  
  NUM# TIPO=BYTE(4),OFFSET=6  
  CCC# TIPO=BYTE(11),OFFSET=10,INDEX=IO1  
  CONCEPTO TIPO=BYTE(20),OFFSET=21  
  FECHA TIPO=BYTE(8),OFFSET=41  
  CANTIDAD TIPO=BYTE(5),OFFSET=49  
  SALDO TIPO=BYTE(5),OFFSET=54
```

El fichero CLIENTE_ALMACENADO, que estaba ordenado por CLI#, ahora estará **desordenado** (fichero heap o montón)

En CLIENTE_ALMACENADO se crea un **nuevo índice** sobre NOM

OPERACIONES_ALMACENADAS se **mueve a otra unidad de disco**

□ El Esquema Conceptual/Lógico permanece inalterado

▣ Sólo hay que modificar la **correspondencia** entre esquemas

10.3. Independencia de Datos

26

□ Resumiendo...

¿**Cómo** conseguir la **independencia de datos**?

▣ Gracias a la **Arquitectura de Tres Niveles (A3N)**

▣ Y a los **metadatos** (INFORMATION_SCHEMA del Catálogo)

■ Incluyen la información sobre las **correspondencias** entre esquemas

□ Así, **la modificación del esquema de un nivel provoca cambios en las correspondencias** entre niveles, *y el esquema del nivel superior no varía*



□ Inconvenientes

▣ Las correspondencias entre niveles deben mantenerse permanentemente actualizadas

▣ Gasto extra en compilación/ejecución de programas

▣ Y por tanto, menor eficiencia del SGBD

10.3. Independencia de Datos

27



ATTENTION

La independencia de datos **incluye** la **independencia entre programas y datos**:

- ❑ La **descripción** de la **estructura y restricciones** de los **datos** se almacena **en el Catálogo del sistema** (en el **INFORMATION_SCHEMA**), **separada de los programas**
- ❑ Así, la mayoría de los cambios en la estructura de los datos **NO implica modificar los programas**
- ❑ Esto no pasa en el *procesamiento de ficheros tradicional*:
 - ▣ Los datos se definen como parte del código de la aplicación
 - ▣ Así, un cambio en la estructura de los datos provoca modificar y recompilar todos los programas que acceden a dichos datos
- ❑ Además, **los programas no pueden “saltarse” las restricciones de integridad**



10.4. Funciones del SGBD

28

1. Almacenamiento, recuperación y actualización de datos
2. Servicios para permitir la independencia de datos ☒
3. Metadatos accesibles para el usuario ☒
4. Servicios de integridad ☒
5. Soporte y procesamiento de transacciones
6. Servicios de control de la concurrencia
7. Servicios de recuperación de fallos
8. Servicios de autorización (seguridad)
9. Soporte para la comunicación de datos
10. Utilidades adicionales

☒ Ya lo hemos estudiado

1. Almacenamiento, Recuperación y Actualización de datos

29

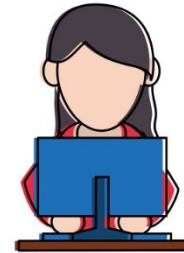
- Proporcionar al usuario la capacidad de **almacenar, recuperar y actualizar** datos en la base de datos y de forma **eficiente**
 - ▣ Ocultar detalles internos de implementación física (**A3N**)
 - ▣ **Procesamiento y Optimización**
 - Más info en el Tema 11

Productos		
ID de produ	Nombre del pro	Proveedor
1	Tapias	Exotic Liquids
2	Chang	Exotic Liquids

Clientes	
Nombre de la empresa	Nombre de con
Alfreds Futterkiste	María Anders
Ana Trujillo Emparedados y helados	Ana Trujillo

Pedidos		
Código de	Cliente	Empleado
10248	Wilman Kala	Buchanan, Ste
10249	Tradição Hipermercados	Suyama, Mich
10250	Hanari Carnes	Peacock, Marg

Registro: 1 de 830 Sin filtro Buscar



Datos Auditoria

Cliente: Auditoria | Enlaces | Estructura Contable | Observaciones | Vinculación

Razon Social: GOTHAM | Cuenta: 101250 | N.I.F.: A/48577066 | Período Contable: Fecha Inicio: 01/01/13 | Fecha Cierre: 31/12/13

Actividad: | Sede: | Epigrafe: 125.4 | C.N.A.E.: 2922

Domicilio: Polígono industrial Granada G-25 | Tipo de sede: Domicilio central y filial | Localidad: Bujella | CP: 48502 | Provincia: MIZCAYA | País: ESPAÑA | Tel: 946366525 | Fax: 946366525 | E-Mail: gotham@gotham.es

Contacto:

Nombre	Teléfono	Fax	Cargo	Móvil	Email
Anacleto			Director Financiero		
Evaristo			Gerente		

Introducir la información de los certificados

Empresa: ☒ DESEQUIP ☐ DESEQUIP (Stigo) ☐ CORRAL ☐ Otros

Transportada en: ☒ Contenedor ☐ Bulto ☐ Otros

Fecha de Medición: 28/09/2004

Datos del contenido: TOLU 452206-4

Confeccionado por: Odalys Ramos Vitor

Firmado por: Orlando Domínguez Ley

Idioma del Certificado: ☒ Español ☐ Inglés

Contenido: CHATARRA DE ACERO INOXIDABLE

Código	Fecha	Empresa	Transp. en	Contenedor	Contenido
SMHC/555/04	28/09/04	DESEQUIP	Contenedor	TOLU 452206-4	CHATARRA DE ACERO INOX

Botones: Añadir, Leer Reporte, Eliminar, Eliminar todas, Imprimir, Imprimir todas, Cerrar

PLACAS AUTOCROMAS

Nº Inventario Museo: | Nº grabado vidrio: | Nº Etiqueta papel: | Nº Fotos: |

Identificación en publicación: | Nº página: |

Estado de conservación: Química | Física

Defecto Vidrio: |

Ref. Bibliográficas: |

Foto: |

Botones: Agregar, Modificar, Borrar, Buscar, Salir

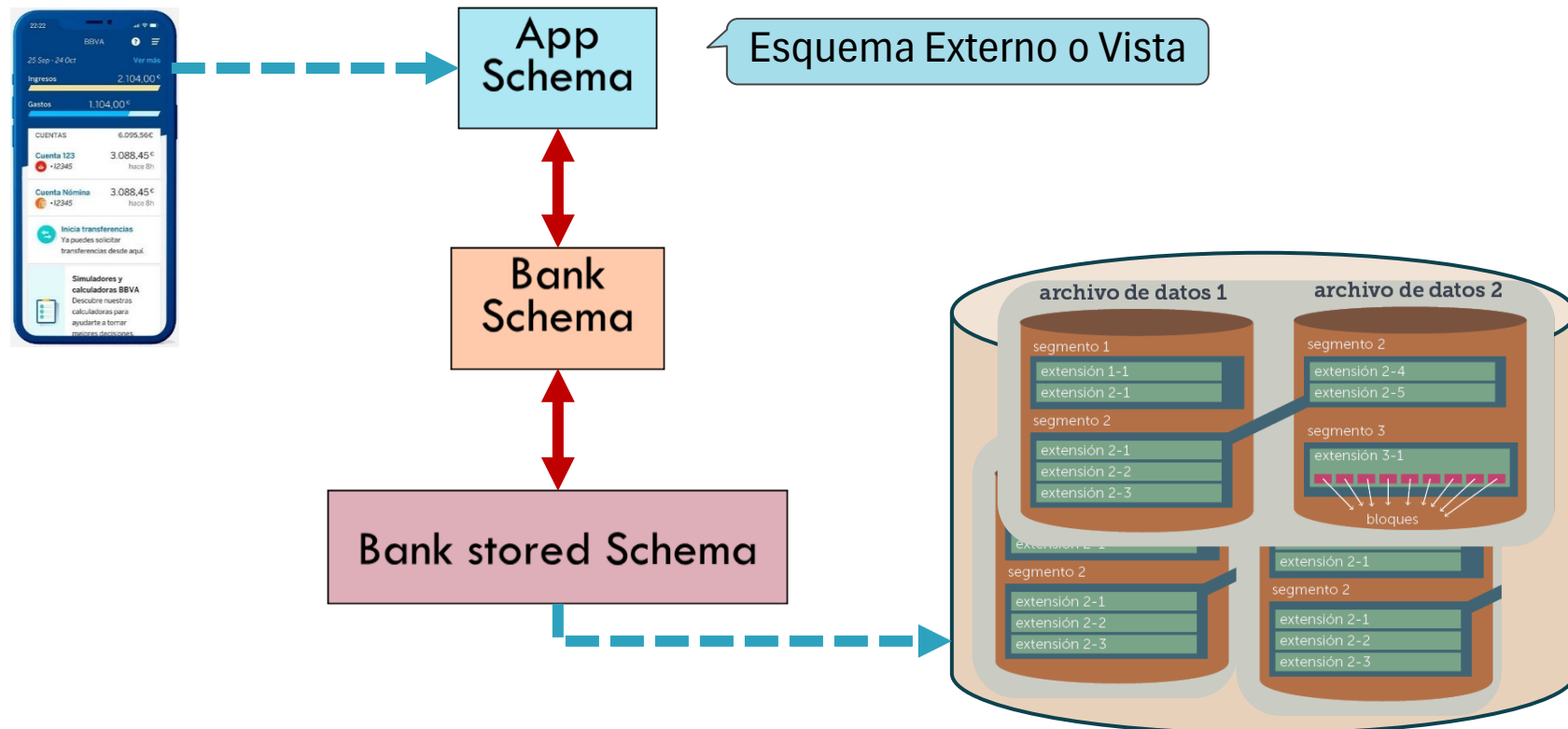
2. Servicio para permitir la independencia de datos



30

- Facilidades para soportar la independencia de los programas de la estructura física de la base de datos

▣ A3N



3. Metadatos accesibles para el usuario



31

□ INFORMATION_SCHEMA

- Metadatos almacenados en la base de datos y accesibles para el usuario y para el SGBD

❖ Más información en el Tema 1. Introducción a los SBD

- Componente **fundamental** del sistema:
imprescindible para hacer posible la Arquitectura de Tres Niveles (A3N) y para el funcionamiento de los subsistemas software que integran el SGBD
(integridad, seguridad, etc.)

4. Servicios de integridad



32

- ❑ Mecanismos para asegurar que tanto los datos en la base de datos, como los cambios sobre los datos, siguen ciertas reglas
 - ▣ Datos correctos y consistentes: datos de calidad
- ❑ **Reglas de integridad**
 - ▣ Integridad de entidad (PK) y referencial (FK)
 - ▣ Tipos de datos y restricciones sobre valores
 - ▣ Admisión de nulos y valores por defecto
 - ▣ ...
- ❖ Más información en los Temas 6 (DL) , 7 (LDD) y 9 (RI)



5. Soporte de Transacciones

33

□ Una **Transacción** es una secuencia de acciones que lee y/o actualiza el contenido de la BD, y que debe ser ejecutada como una unidad

□ **Soporte de Transacciones:** Mecanismo que asegura que **todos** los cambios hechos por una transacción se realizan correctamente y quedan almacenados en la BD, o bien que **ninguno** de esos cambios se ha realizado



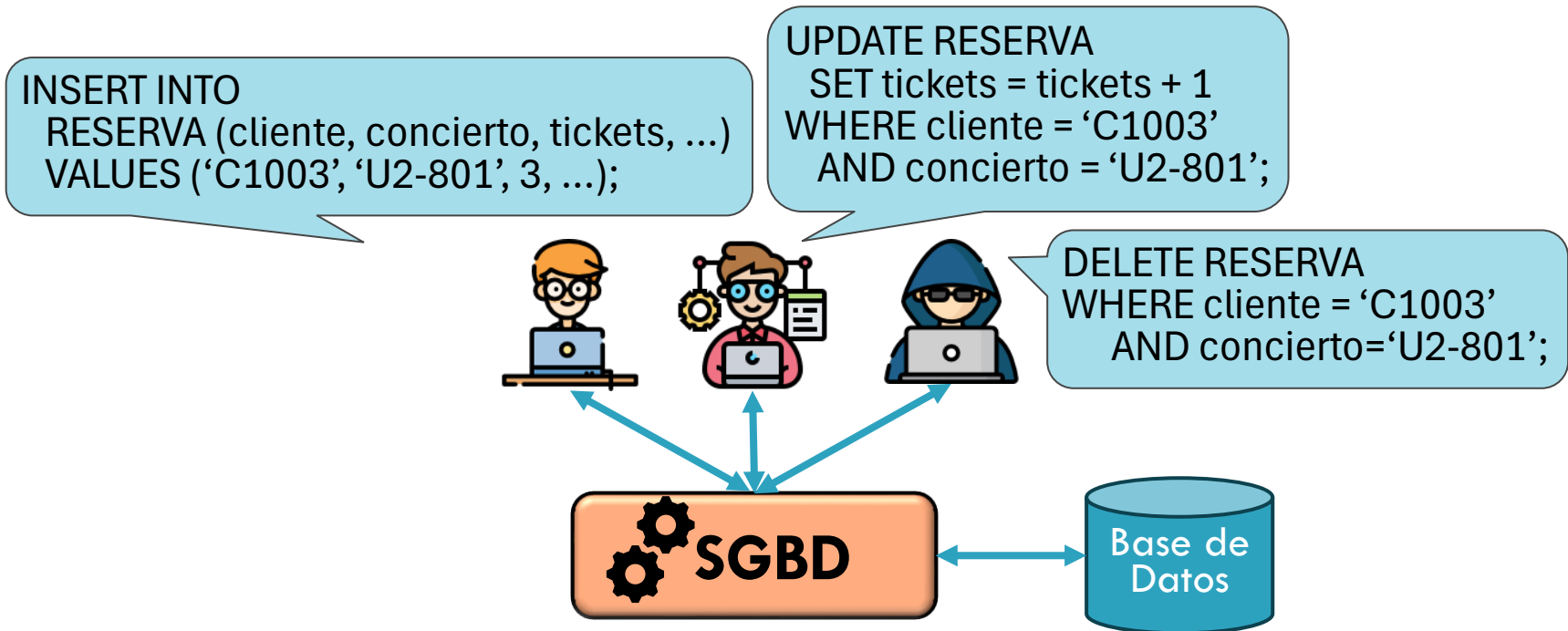
❖ Más información en el Tema 11

□ Esto se garantiza incluso en presencia de fallos del sistema, y teniendo en cuenta la **conurrencia** (los múltiples usuarios/aplicaciones que acceden a la BD al mismo tiempo)

6. Servicio de Control de la Concurrency

34

- ❑ Mecanismo que garantiza que la BD es actualizada correctamente cuando **múltiples usuarios/aplicaciones modifican los datos** simultáneamente

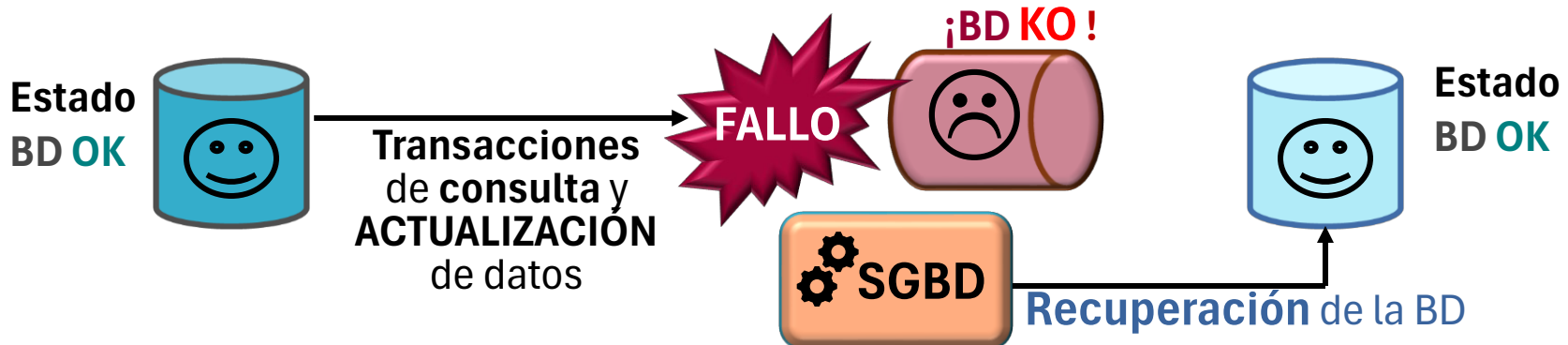


❖ Más información en el Tema 11

7. Servicio de Recuperación de Fallos

35

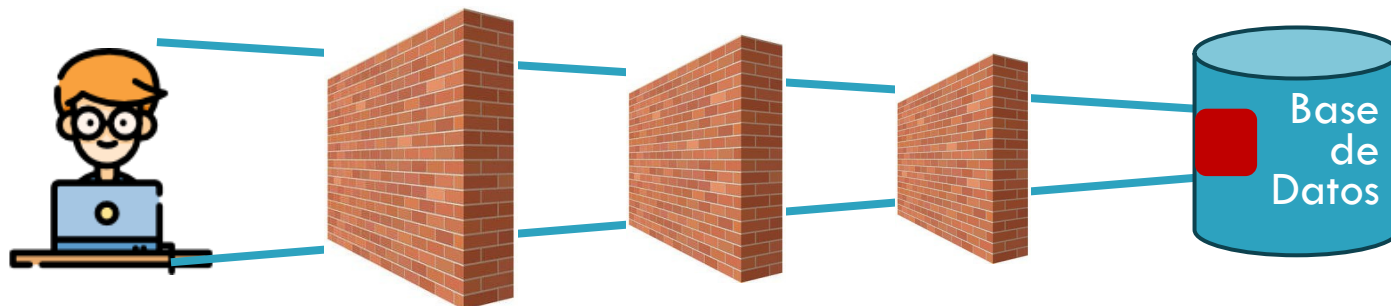
- ❑ Mecanismo para recuperar la BD tras la ocurrencia de fallos en el sistema que quizá han dejado los datos inconsistentes
 - ▣ Fallos del sistema (*system crash*)
 - ▣ Fallos en los medios de almacenamiento
 - ▣ Errores *hardware* o *software* que han parado el SGBD
 - ▣ Fallos provocados por el usuario (cancelación de ejecución), etc.
- ❑ La recuperación consigue dejar la BD en un estado en el que los **datos** son **consistentes**
 - ❖ Más información en el Tema 11



8. Servicio de Autorización (seguridad)

36

- ❑ Garantiza que **sólo los usuarios autorizados pueden acceder a la base de datos**
 - ▣ No todos los usuarios deben poder acceder a toda la BD
 - Información médica, financiera, diseños con propiedad intelectual, etc.
- ❑ Objetivo: **control de acceso selectivo**
 1. Sólo usuarios autorizados
 2. Sólo a ciertas partes de la base de datos
 3. Sólo para realizar ciertas operaciones con los datos



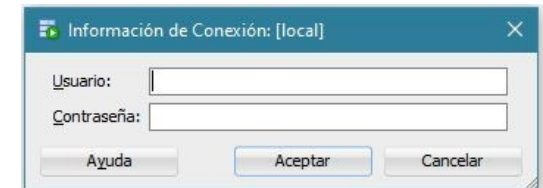
8. Servicio de Autorización (seguridad)

37

- El SGBD permite al ABD la creación de...
 - ▣ Cuentas de usuario protegidas con contraseña → ok objetivo 1
- ▣ Restricciones de seguridad para cada cuenta → ok objetivos 2 y 3

```

1 CREATE USER bdi100 IDENTIFIED BY cambiame
2   DEFAULT TABLESPACE "USERS"
3   TEMPORARY TABLESPACE "TEMP"
4   PASSWORD EXPIRE;
```



```

1 GRANT CONNECT, RESOURCE TO bdi100;
2 GRANT SELECT ON VIDEOTECA.PELICULA TO bdi100;
3 GRANT CREATE VIEW TO bdi100;
```

- El SGBD obliga al cumplimiento de las reglas/restricciones de seguridad y autorización

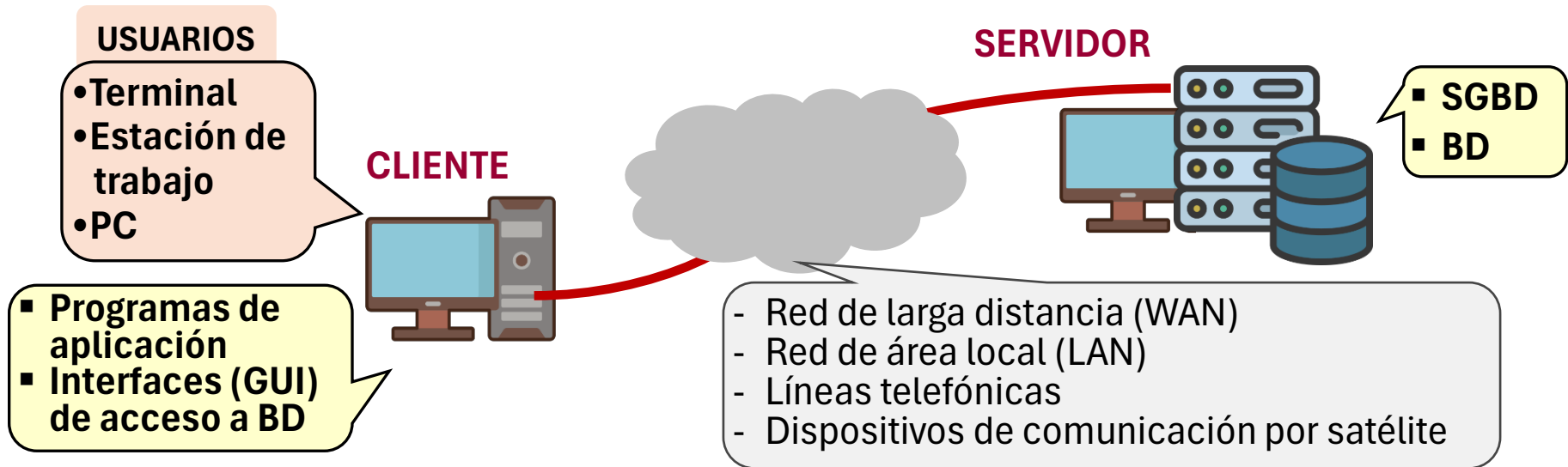
```

ORA-01031: privilegios insuficientes
01031. 00000 - "insufficient privileges"
*Cause:      An attempt was made to perform a database operation without
              the necessary privileges.
*Action:     Ask your database administrator or designated security
              administrator to grant you the necessary privileges
```

9. Soporte para Comunicación de Datos

38

- Un SGBD debe poder integrarse con *software* de comunicaciones (*Data Communication Programs*)
- Garantiza que **los usuarios accedan** sin problemas a una **BD centralizada desde ubicaciones remotas**
 - El SGBD debe poder ejecutarse en una red (ej. de área local, LAN) de forma que la BD centralizada pueda compartirse por los usuarios cuyos equipos están conectados a dicha red



10. Utilidades adicionales

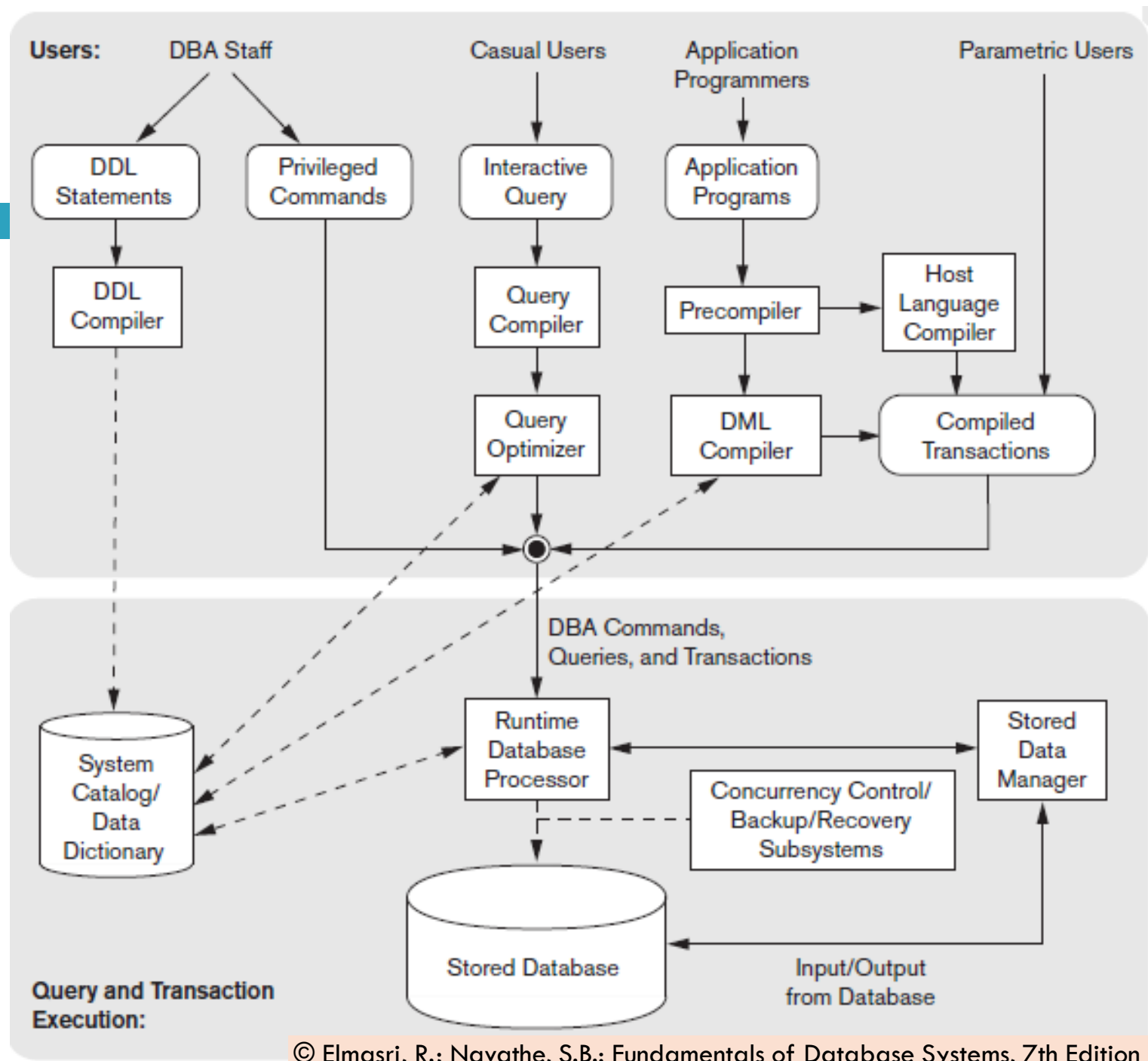
39

- El SGBD proporciona un conjunto de utilidades que ayudan al ABD a administrar la BD de manera efectiva
 - ▣ Herramientas de **importación** y **exportación** de datos
 - Carga de datos en la base de datos a partir de ficheros sin estructura
 - Descargar datos en ficheros de diversos tipos
 - **Intercambio de información** entre diferentes BD (*Oracle* \longleftrightarrow *MS Access*)
 - ▣ Utilidades de **monitorización** o supervisión, para controlar el uso, operación y rendimiento del sistema
 - ▣ Programas de **análisis estadístico** para examinar las estadísticas de rendimiento y uso
 - ▣ Facilidades de **reorganización** de ficheros o de índices
 - ▣ Herramientas de **backup**
 - ▣ Utilidades de ordenamiento, compactación y/o compresión de ficheros, etc.

Anexos

1. Estructura general del SGBD
2. Ventajas y desventajas de los SGBD
3. Sistemas tradicionales de Procesamiento de Ficheros

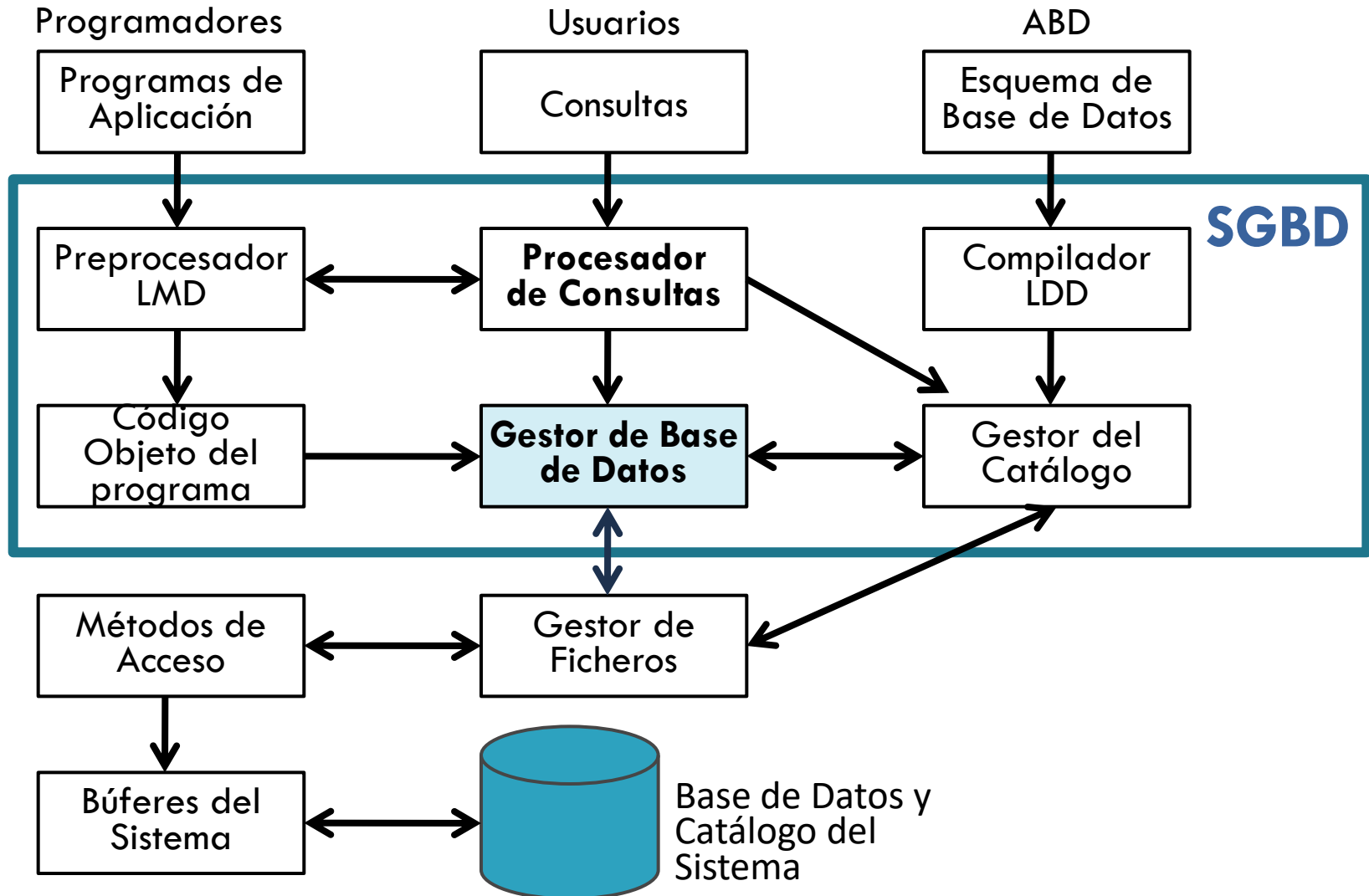
Anexo 1. Estructura general del SGBD. Módulos software componentes



Anexo 1. Estructura general del SGBD

Módulos software componentes

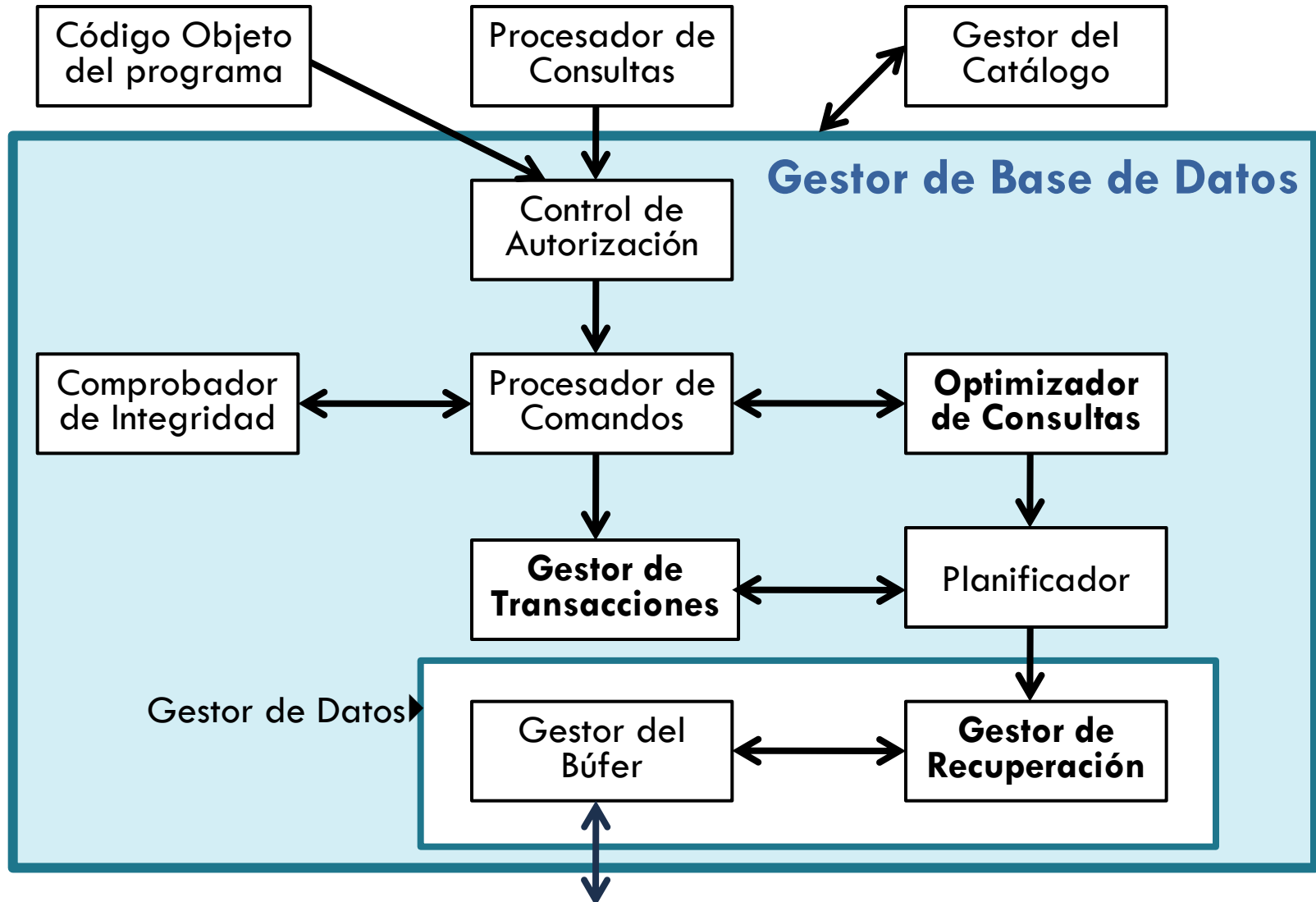
42



Anexo 1. Estructura general del SGBD

Módulos software componentes

43

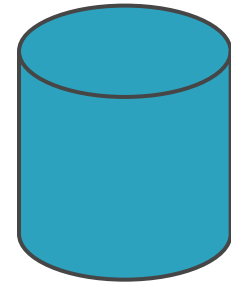


Anexo 1. Estructura general del SGBD

Estructuras de Datos

44

- ❑ **Ficheros de datos** en disco
- ❑ **Estructuras de acceso** (índices, etc.)
 - ▣ Agilizan el acceso a los datos
- ❑ **INFORMATION_SCHEMA**
 - ▣ **Metadatos** acerca de la estructura y organización de los datos, restricciones de integridad y de autorización, etc., etc.
 - ▣ Y estadísticas sobre los datos en la BD, usadas para optimización de consultas
 - ▣ Acceso muy frecuente por diferentes módulos del SGBD
 - Es necesario un buen diseño e implementación que aumente la eficiencia



Anexo 2. Ventajas de los SGBD

45

- ❑ Control de la redundancia de datos
- ❑ Consistencia de los datos
- ❑ Más información a partir de la misma cantidad de datos
- ❑ Compartición de los datos
- ❑ Integridad de datos mejorada
- ❑ Seguridad mejorada
- ❑ Refuerzo del uso de estándares
- ❑ Economía de escala
- ❑ Equilibrio entre requisitos conflictivos
- ❑ Mejora en tiempos de respuesta y accesibilidad de datos
- ❑ Incremento de la productividad
- ❑ Mantenimiento mejorado gracias a la independencia de datos
- ❑ Concurrencia mejorada
- ❑ Servicios de backup y recuperación mejorados

Anexo 2. Desventajas del uso de SGBD

46

- ❑ Complejidad
- ❑ Tamaño
- ❑ Coste del software de SGBD
- ❑ Costes hardware adicionales
- ❑ Coste de conversión
- ❑ Rendimiento
- ❑ Mayor impacto de los fallos

Anexo 3. Sistemas de Procesamiento de Ficheros

47

- ❑ Surgieron al tratar de informatizar el manejo de los archivadores manuales, con el objetivo de proporcionar un acceso más eficiente a los datos
- ❑ Se escogió un modelo descentralizado en el que **cada departamento** (o sección) de la organización (o empresa) **almacena y gestiona sus propios datos de modo independiente**
- ❑ Un sistema de procesamiento de ficheros es un **conjunto de programas** que prestan servicio a los usuarios finales
- ❑ **Cada programa define y maneja sus propios datos**
- ❑ La **estructura física de los ficheros de datos y sus registros** está **definida en los programas de aplicación**
 - ❑ Es parte del **código**

Anexo 3. Sistemas de Procesamiento de Ficheros

48

EJEMPLO.

Ficheros de la aplicación para el Departamento de Ventas de una inmobiliaria

INMUEBLE

icod	calle	area	poblac	tipo	hab	alquiler	propiet
IA14	Platería, 12	Centro	Murcia	Casa	6	600	P46
IL94	Mayor, 22	Ronda Sur	Murcia	Piso	4	350	P87
IG4	Salsipuedes, 2	Norte	Yecla	Piso	3	300	P40
IG36	Sol, 33		Archena	Piso	3	325	P93
IG21	Colón, 42		Mazarrón	Casa	5	550	P87
IG16	Zalamea	Centro	Murcia	Piso	4	400	P93

PROPIETARIO

pcod	nombre	apellido	direccion	telefono
P46	Carmen	Carpena	Gran Vía 53, Murcia	968230680
P87	Manuel	Méndez	Av. Libertad 15, Yecla	968790001
P40	Alberto	Alcaraz	Renacimiento 1 Murcia	968710430
P93	Belinda	Bono	Carril 10, Archena	968401074

INQUILINO

qcod	nombre	apellido	direccion	telefono	tipo	alquiler
Q76	Juan	Jiménez	Benavente 47, Villena	661282540	Piso	375
Q56	Ana	Alonso	San Rafael 45, Castellón	663551110	Piso	300
Q74	Elena	Espina	Navarra 76, Jumilla	618205560	Casa	700
Q62	Alfredo	Alcántara	Mayor 21, Almansa	621229580	Piso	550

Anexo 3. Sistemas de Procesamiento de Ficheros

EJEMPLO.

49

Ficheros de la aplicación para el **Departamento de Contratos de una inmobiliaria**

INMUEBLE

icod	calle	area	poblac	alquiler
IA14	Platería, 12	Centro	Murcia	600
IL94	Mayor, 22	Ronda Sur	Murcia	350
IG21	Colón, 42		Mazarrón	550

CONTRATO

ccod	icod	qcod	importe	pago	deposito	pagado?	inicio	fin	meses
10024	IA14	Q62	600	Visa	1200	S	1/6/03	31/5/04	12
10075	IL94	Q76	350	Efectivo	700	N	1/1/00	30/6/00	6
10012	IG21	Q74	550	Cheque	1100	S	1/7/99	30/6/00	12

INQUILINO

qcod	nombre	apellido	direccion	poblacion	telefono
Q76	Juan	Jiménez	Benavente, 47	Villena	661282540
Q74	Elena	Espina	Navarra, 76	Jumilla	618205560
Q62	Alfredo	Alcántara	Mayor, 21	Almansa	621229580

Anexo 3. Sistemas de Procesamiento de Ficheros: **inconvenientes**

50

- ❑ **Separación y aislamiento de los datos**
 - ▣ Es más complicado acceder a los datos almacenados en ficheros distintos al tener que sincronizar su procesamiento para asegurar que se obtienen los datos adecuados
- ❑ **Redundancia de datos**
 - ▣ Esto implica desperdicio de espacio y posible inconsistencia de datos
- ❑ **Programas dependientes de los datos**
 - ▣ Los cambios en la estructura de los datos implican identificar los programas afectados, modificarlos y depurarlos
- ❑ **Datos dependientes del lenguaje de programación**
 - ▣ Puede implicar la incompatibilidad entre ficheros generados por lenguajes distintos, lo que dificulta su procesamiento conjunto
- ❑ **Dependencia del programador de aplicaciones**
 - ▣ Cualquier consulta o informe ha de ser programado
 - ▣ Esto puede llevar al aumento del nº de programas para resolver todo tipo de consultas, que se deben validar, mantener, documentar...