

Índice

- ☐ Introducción
- ☐ Bases de Datos
 - ☐ Definiciones
 - ☐ Propiedades
 - ☐ Ventajas
 - ☐ Sistema Gestor de Bases de Datos
 - ☐ Usuarios
- ☐ Modelos de datos
 - ☐ Definición
 - ☐ Categorización
 - ☐ Esquemas, instancias o estado de la base de datos
- ☐ **Conceptos y arquitectura del sistema de bases de datos**
 - ☐ Arquitectura de tres niveles
 - ☐ Lenguajes
 - ☐ Interfaces
- ☐ Estructura general del sistema de bases de datos
 - ☐ Módulos componentes
 - ☐ Utilidades

Bibliografía

- R. ELMASRI Y S. V. NAVATHE, *Fundamentos de Sistemas de Bases de Datos* (5ªed.), Addison Wesley, 2007 (Capítulos 1 y 2)
- T. CONNOLLY Y C. BEGG, *Sistemas de Bases de Datos* (4.ed.) Addison-Wesley, 2005 (Capítulo 1)

Arquitectura de tres niveles

*En 1975, el comité ANSI-SPARC (American National Standard Institute – Standards Planning and Requirements Committee) propuso una **arquitectura de tres niveles para los SGBD** cuyo objeto objetivo principal era separar la BD física de los programas de aplicación.*

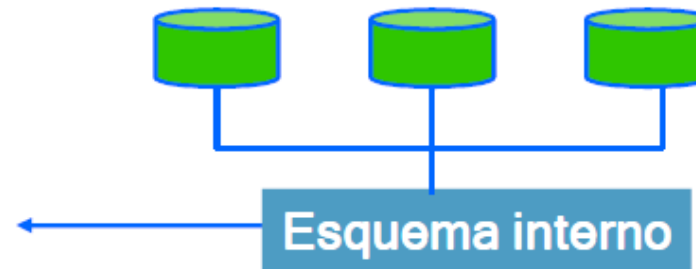
En concreto, esta arquitectura de tres niveles ayuda a conseguir las siguientes características:

- (1) Aislamiento de los programas y los datos,
- (2) Soporte de varias vistas de usuarios,
- (3) Uso de un catálogo y de un diccionario de datos,
- (4) Independencia de los datos.

Arquitectura de tres niveles: Descripción

Nivel Interno o Físico

Describe la estructura física de almacenamiento.



Nivel Conceptual

Describe entidades, tipos de datos, relaciones, operaciones de los usuarios y restricciones.

Esquema Conceptual

Nivel Externo o de Vista

Describe la parte de la BD que un grupo de usuarios está interesado y le oculta el resto de la BD.

Esquema Externo 1
(Vista Usuario 1)

Esquema Externo 2
(Vista Usuario 1)



Arquitectura de tres niveles: Descripción

Nivel Interno o Físico

- Es el más cercano al almacenamiento físico, es decir, tal y como están almacenados en el ordenador.
- Describe la estructura física de la BD mediante un **Esquema Interno**.
- Este esquema se especifica con un **Modelo Físico** y describe los detalles de cómo se almacenan físicamente los datos:
 - los archivos que contienen la información, su organización,
 - los métodos de acceso a los registros,
 - los tipos de registros y su longitud,
 - los campos que los componen.

Nivel Conceptual

- Describe la estructura de toda la BD para un grupo de usuarios mediante un **Esquema Conceptual**.
- Este esquema describe las entidades, atributos, relaciones, operaciones de los usuarios y restricciones, ocultando los detalles de las estructuras físicas de almacenamiento.
- Representa la información contenida en la BD.

Nivel Externo o de Vista

- Es el más cercano a los usuarios, es decir, es donde se describen varios **Esquemas externos o Vistas** de usuarios.
- Cada esquema describe la parte de la BD que interesa a un grupo de usuarios en este nivel se representa la visión individual de un usuario o de un grupo de usuarios.

Arquitectura de tres niveles: Descripción

Esquema Interno

Estrategia de almacenamiento

Caminos de acceso

Compresión de datos, correspondencia entre Esquema Interno y Conceptual, optimización,...

Esquema Conceptual

Visión global de los datos: descripción, interrelación, restricciones de integridad y confidencialidad,...

Esquema Externo

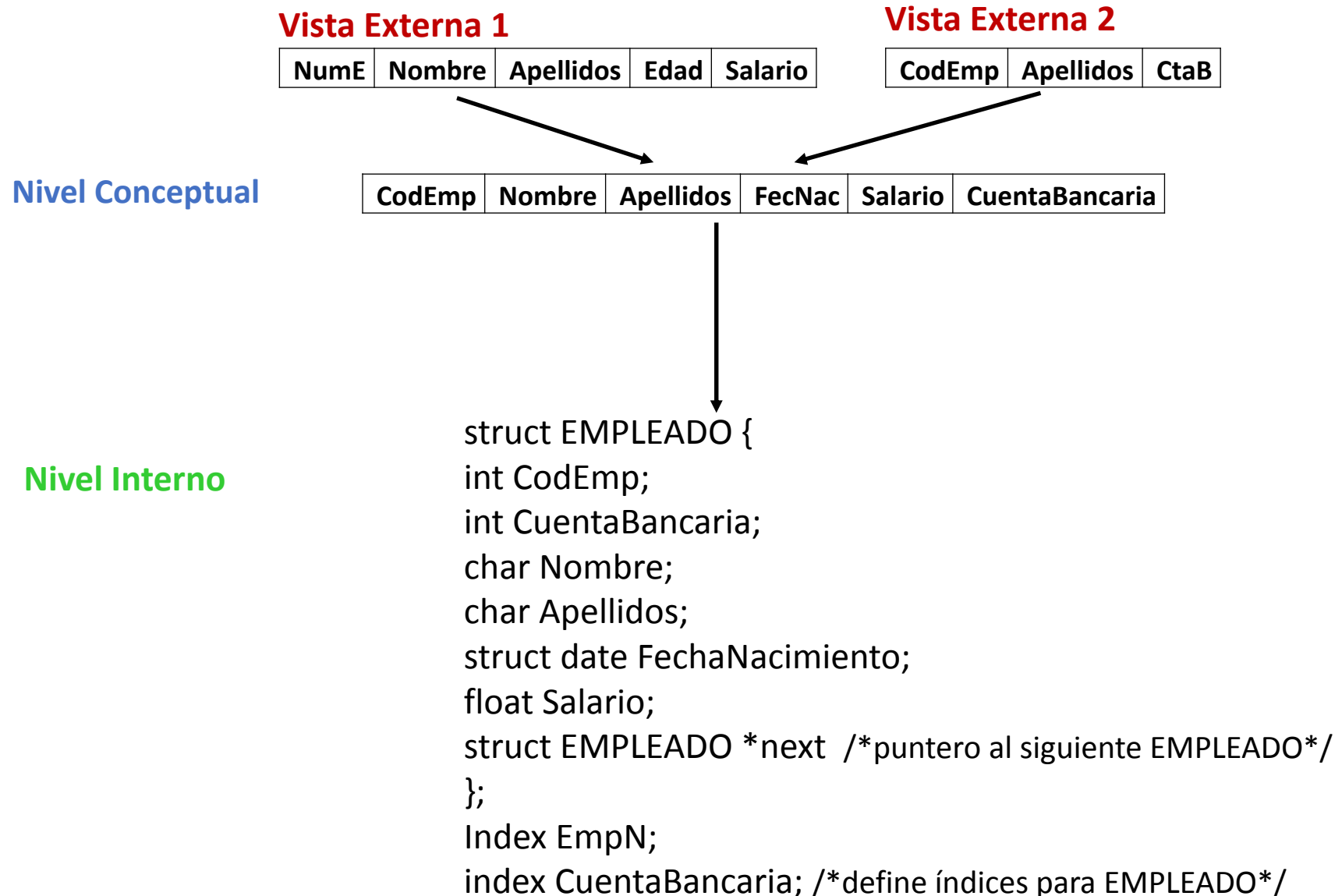
Visión de la BD de un usuario particular: sólo datos que le “interesen” , restricciones de uso,...

Arquitectura de tres niveles: Descripción

Los tres esquemas son **sólo**
Descripciones de los Datos

Los únicos datos que
realmente existen
están en el **nivel físico**

Arquitectura de tres niveles: Ejemplo



Arquitectura de tres niveles: Independencia de los datos

La arquitectura de tres esquemas ayuda a conseguir las características siguientes:

(1) aislamiento de los programas y los datos, (2) soporte de varias vistas de usuarios, (3) uso de un catálogo y de un diccionario de datos e **(4) independencia de los datos.**

A. Independencia lógica de los datos

- Es la capacidad de cambiar el **Esquema Conceptual** sin tener que cambiar los esquemas externos o los programas de aplicación.
- La independencia lógica de los datos es muy difícil de conseguir porque permite los cambios estructurales y restrictivos sin afectar a los programas de aplicación (un requisito más estricto).

Vista incentivos

cod_del	cod_ped	cod_cli
BH89	1234	9832A
BH89	8903	7329H
RT87	5432	5698Q
RT87	3425	9832A

- **Por ejemplo: añadir un campo, cambiar restricciones ...**

cod_cli	nombre	ciudad	país	contacto	correo
9832A	RERSA	Sabadell	España	Perez	rorsa@unisa.es
5642S	TURSA	Toulouse	Francia	Dubois	tursa@fransa.fr
7329H	ENERSA	Bristol	UK	Brown	enersa@tele.uk
5698Q	ITIRSA	Bolonia	Italia	Felici	itirsa@iris.it
32165R	PYRESA	Langreo	España	Pérez	pyresa@rete.es

Arquitectura de tres niveles: Independencia de los datos

B. Independencia física de los datos

- Es la capacidad de cambiar el **Esquema Interno** sin tener que cambiar el esquema conceptual. Por tanto, tampoco se cambian los esquemas externos.
- Por ejemplo: reorganización de archivos físicos (creación de estructuras de acceso adicionales) para mejorar el rendimiento de las consultas y/o actualizaciones.
- La Independencia física de los datos existe en la mayoría de las BBDD y de los entornos de archivos en los que al usuario se oculta la ubicación exacta de los datos en el disco, la fusión, la división de registros etc. Las aplicaciones obvian estos detalles.

Vista incentivos		
cod_del	cod_ped	cod_cli
BH89	1234	9832A
BH89	8903	7329H
RT87	5432	5698Q
RT87	3425	9832A

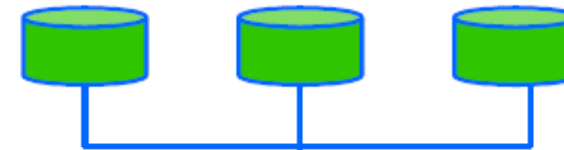
Establecemos un nuevo camino de acceso para agilizar la obtención de los registros

Cab_ped					
cod_ped	cod_cli	cod_del	portes	embalaje	fech_ped

Arquitectura de tres niveles: Relación con los Modelos de Datos

Nivel Interno o físico

Describe la estructura física de almacenamiento.
Modelo de bajo nivel (o físico).



Esquema interno

Nivel Conceptual

Describe entidades, tipos de datos, relaciones, operaciones de los usuarios y restricciones.
Modelo de alto nivel (o Conceptual) o modelo de bases de datos (o implementación).

Esquema Conceptual

Nivel Externo o de Vista

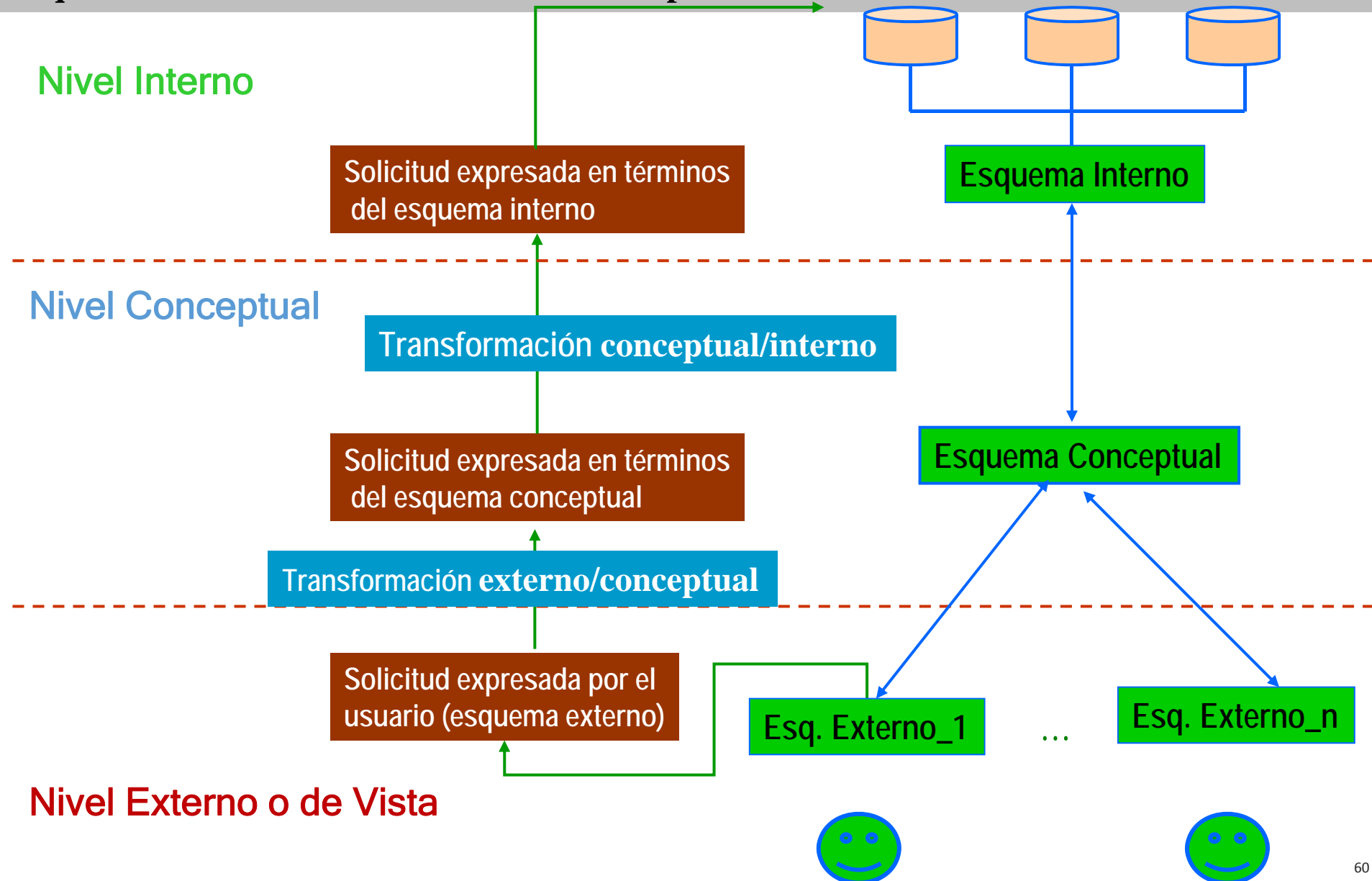
Describe la parte de la BD que un grupo de usuarios está interesado y le oculta el resto de la BD.
Modelo de alto nivel (o Conceptual) o modelo de bases de datos (o implementación).

Esquema Externo 1
(Vista Usuario 1)

Esquema Externo 2
(Vista Usuario 1)



Arquitectura de tres niveles: Relación con el proceso de **Transformación**



Arquitectura de tres niveles: Relación con el proceso de **Transformación**

- El SGBD debe de transformar cualquier petición de usuario (**Esquema Externo**) a una petición expresada en términos de **Esquema Conceptual**, para finalmente ser una petición expresada en el **Esquema Interno** que se procesará sobre la BD almacenada.
- El proceso de transformar peticiones y resultados de un nivel a otro se denomina ***correspondencia o TRANSFORMACIÓN***.
- Para ello el SGBD realiza los siguientes pasos:
 - El usuario solicita unos datos y crea una consulta.
 - El SGBD verifica y acepta el esquema externo para ese usuario.
 - Transforma la solicitud al esquema conceptual.
 - Verifica y acepta el esquema conceptual.
 - Transforma la solicitud al esquema físico o interno.
 - Selecciona la o las tablas implicadas en la consulta y ejecuta la consulta.
 - Transforma del esquema interno al conceptual, y del conceptual al externo.
 - Finalmente, el usuario ve los datos solicitados.

Arquitectura de tres niveles: Relación con el proceso de **Transformación**

- Los **SGBD comerciales** **NO** distinguen del todo los 3 niveles:
 - Algunos incluyen detalles físicos en el Esquema Conceptual.
 - Usan el mismo modelo de datos para especificar
 - ✓ Esquemas Externos (vistas) y
 - ✓ Esquema Conceptual
- Los 3 niveles son **descripciones** de datos
 - Los **datos reales** sólo están **en** el **nivel físico**

Lenguajes

- Todos los **SGBD** ofrecen lenguajes e interfaces apropiadas para cada tipo de usuario:
 - Lenguaje de Definición de Vistas (LDV o VDL)
 - Lenguaje de Definición de Datos (LDD o DDL)
 - Lenguaje de Definición de Almacenamiento (LDA o SDL).
- Una vez definidos los esquemas y pobladas las BBDD, los usuarios deben disponer de un lenguaje para la consulta, inserción y borrado de datos. El SGBD proporciona este lenguaje que se denota como **Lenguaje de Manipulación de Datos (LMD o DML)**.

Lenguajes: Clasificación

▪ Lenguaje de Definición de Vistas (LDV o VDL)

- Para conseguir Arquitectura de Tres Esquemas “verdadera”
- Usado para **especificar**
 - ✓ **Esquemas Externos** (o Vistas de usuario)
 - ✓ **Correspondencias** entre Vistas y el Esquema Conceptual
- No obstante, la mayoría de SGBD utilizan LDD para definir vistas
- Ejemplo:

```
CREATE VIEW PrestamoLibro AS
SELECT R.fecha, L.titulo, P.nombre
FROM PRESTAMO R, LIBRO L, PERSONA P
WHERE R.idLib = L.isbn AND R.idPer = P.id;
```

Lenguajes: Clasificación

■ Lenguaje de Definición de Datos (LDD o DDL)

- Usado por ABD y diseñadores para **especificar...**
 - ✓ **Esquema Conceptual**
 - ✓ **Esquema Interno**
 - si SGBD sin separación estricta de niveles conceptual e interno
 - ✓ **Correspondencias** entre los anteriores
- **Compilador de LDD:** procesa sentencias escritas en LDD, para **identificar descripciones** de elementos del esquema y **almacenarlas** en el **catálogo** del SGBD.
- Ejemplo:

```
CREATE TABLE Libro (  
isbn      CHAR(10)      PRIMARY KEY,  
titulo    VARCHAR(60) NOT NULL UNIQUE,  
apAutor   VARCHAR(15) NOT NULL,  
nomEdit   VARCHAR(15) NOT NULL,  
año       NUMERIC(4)  
... );
```


Lenguajes: Clasificación

▪ Lenguaje de Definición de Almacenamiento (LDA o SDL)

- En SGBD que distinguen entre niveles conceptual e interno, se usa...
 - ✓ LDA para especificar **Esquema Interno**
detalles de implementación de estructuras de almacenamiento y de acceso
 - ✓ LDD para especificar el **Esquema Conceptual**
entidades + relaciones + restricciones
 - ✓ LDD o LDA para definir Correspondencias entre los anteriores

• Lenguaje de Manipulación de Datos (LMD o DML)

- Consultar, Insertar, Eliminar y Modificar datos
- Dos tipos de LMD: **procedimental y declarativo**.

❖ LMD procedimental o de bajo nivel

- **Qué** datos obtener/manipular y **cómo** obtenerlos/manipularlos
- Siempre debe estar **empotrado** en un LP
- Es un LMD **orientado a registros**: Obtiene **un solo registro** y lo procesa por separado. Necesita usar elementos del LP (ej. bucles) para obtener registro a registro y procesarlos individualmente.

Lenguajes: Clasificación

- **Lenguaje de Manipulación de Datos (LMD o DML)**

- Obtención, Inserción, Eliminación y Modificación de datos
- Dos tipos de DML: **procedimental y declarativo**.

- ❖ **LMD declarativo** o de alto nivel

- **Qué** datos obtener para manipular sin necesidad de indicar también **cómo** obtenerlos.
- Usado para realizar operaciones complejas de BD...
 - Interactivamente -- **Lenguaje de Consulta**
 - Incorporado en un LP anfitrión (host) -- **LMD Empotrado**
- Es un LMD **orientado a conjuntos**:
 - Obtiene/Actualiza **muchos** registros con una única sentencia
 - Ejemplos:

```
SELECT titulo, nomEdit FROM LIBRO WHERE año =2011;  
INSERT INTO AUTOR VALUES (123,"Dolores", "Redondo");  
DELETE FROM PRESTAMO WHERE idLib = 988;  
UPDATE PERSONA SET telefono = 661661661 WHERE idpersona = 1;
```

SGBD comerciales actuales ofrecen **único lenguaje integrado,
mezcla de LDD, LDA, LDV y LMD (como SQL).**

Lenguajes: Clasificación

Nivel Interno o físico

Lenguaje de Definición de Almacenamiento (LDA)
(Lenguaje de Definición de Datos (LDD))

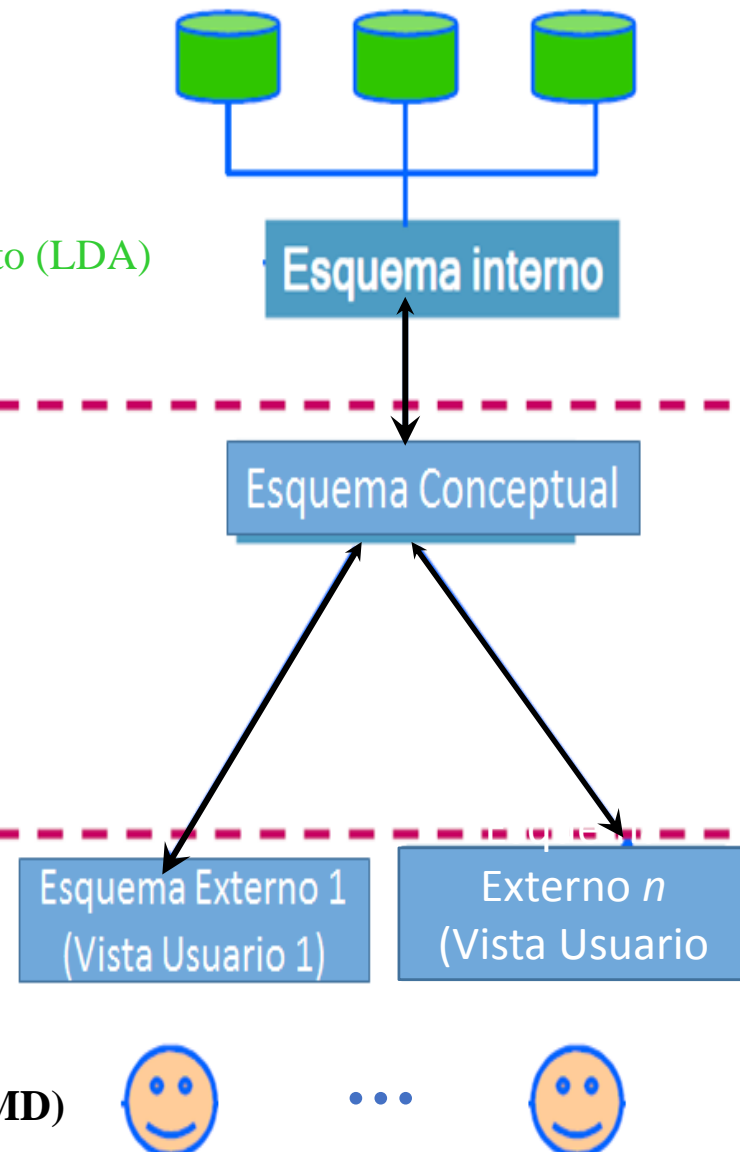
Nivel conceptual

Lenguaje de Definición de Datos (LDD)

Nivel Externo o de Vista

Lenguaje de Definición de Vistas (LDV)

Lenguaje de Manipulación de Datos (LMD)



Lenguajes: Acceso a BBDD desde programas de aplicación

- Programas escritos en un lenguaje (C++, Java, ...) anfitrión
- Dos modos de ejecutar instrucciones LMD desde el lenguaje anfitrión:
 - Usar **Interfaz de Programa de Aplicación** (conjunto de procedimientos)
 - ✓ **ODBC**, Open Data Base Connectivity (definido por Microsoft para C),
 - ✓ **JDBC**, Java Data Base Connectivity
 - **Extender la sintaxis del lenguaje** anfitrión para incorporar llamadas LMD dentro de los programas
 - ✓ Instrucciones LMD precedidas de un carácter especial.
 - ✓ El **preprocesador de LMD** convertirá estas instrucciones en llamadas a procedimientos normales del lenguaje anfitrión.

Interfaces: Tipos

- Basadas en **menús** (principalmente para clientes Web)
- Basadas en **formularios**
- **Gráficas de usuario**
- **De Lenguaje Natural**
- Para los **usuarios paramétricos**
- Para el **ABD**:
 - Crear/eliminar cuentas de usuario y establecer autorizaciones
 - Establecer parámetros de ajuste del rendimiento del sistema
 - Modificación de esquemas y correspondencias
 - Acceso al catálogo
 - Acceso al diccionario de datos
 - Reorganizar estructura de almacenamiento de la BD
 - Realizar/restaurar copias de seguridad
 - ...

Índice

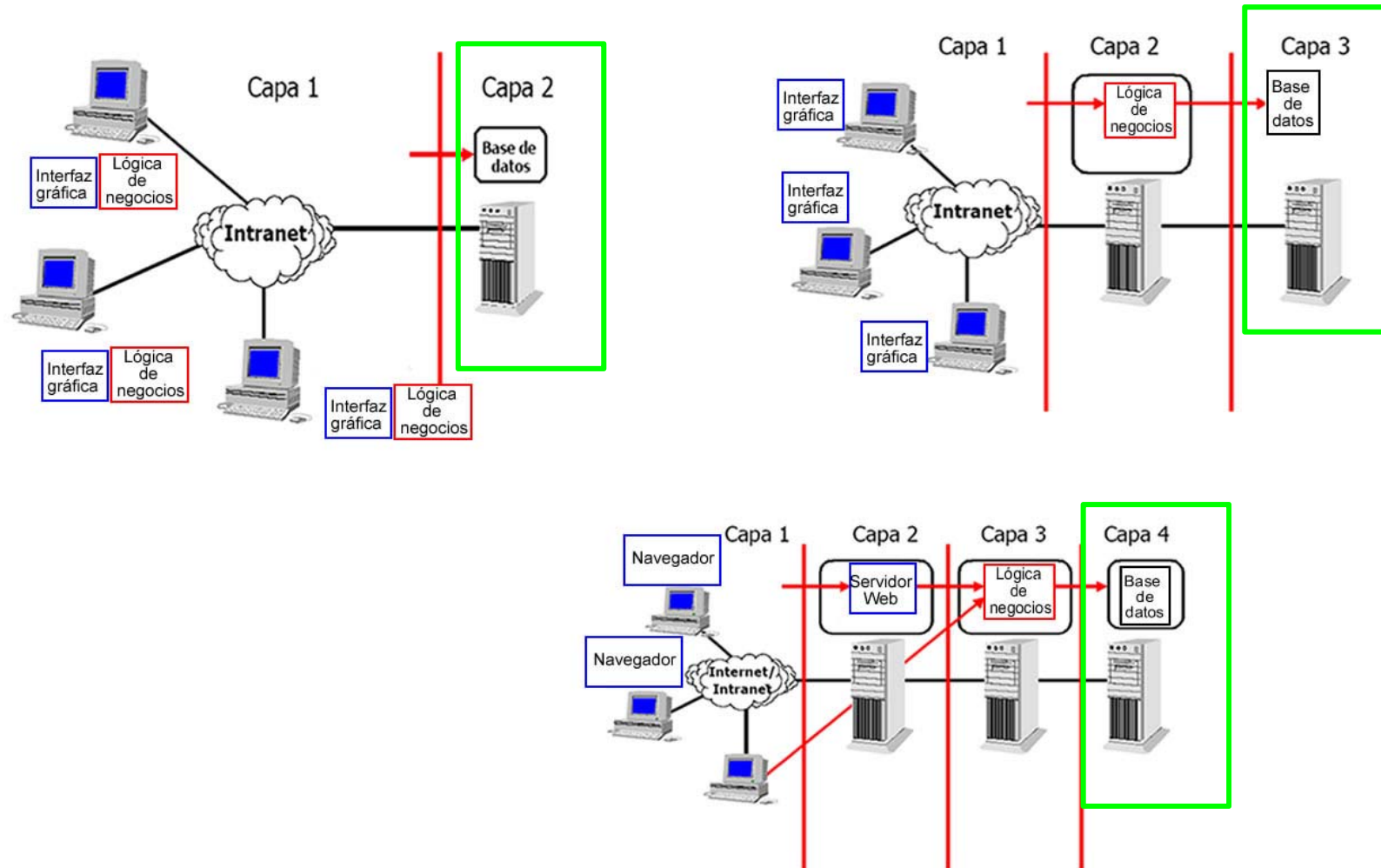
- ☐ Introducción
- ☐ Bases de Datos
 - ☐ Definiciones
 - ☐ Propiedades
 - ☐ Ventajas
 - ☐ Sistema Gestor de Bases de Datos
 - ☐ Usuarios
- ☐ Modelos de datos
 - ☐ Definición
 - ☐ Categorización
 - ☐ Esquema y ejemplar de la base de datos
- ☐ Conceptos y arquitectura del sistema de bases de datos
 - ☐ Arquitectura de tres niveles
 - ☐ Lenguajes
 - ☐ Interfaces
- ☐ **Estructura general del sistema de bases de datos**
 - ☐ Módulos componentes
 - ☐ Utilidades

Bibliografía

- R. ELMASRI Y S. V. NAVATHE, *Fundamentos de Sistemas de Bases de Datos* (5ªed.), Addison Wesley, 2007 (Capítulos 1 y 2)
- T. CONNOLLY Y C. BEGG, *Sistemas de Bases de Datos* (4.ed.) Addison-Wesley, 2005 (Capítulo 1)

Módulos componentes

Arquitectura de los paquetes software de SGBD: **Arquitectura cliente-servidor**



Módulos componentes

**Usuarios del entorno
de BD y sus interfaces**

**Almacenamiento de
datos y procesamiento
de Transacciones**

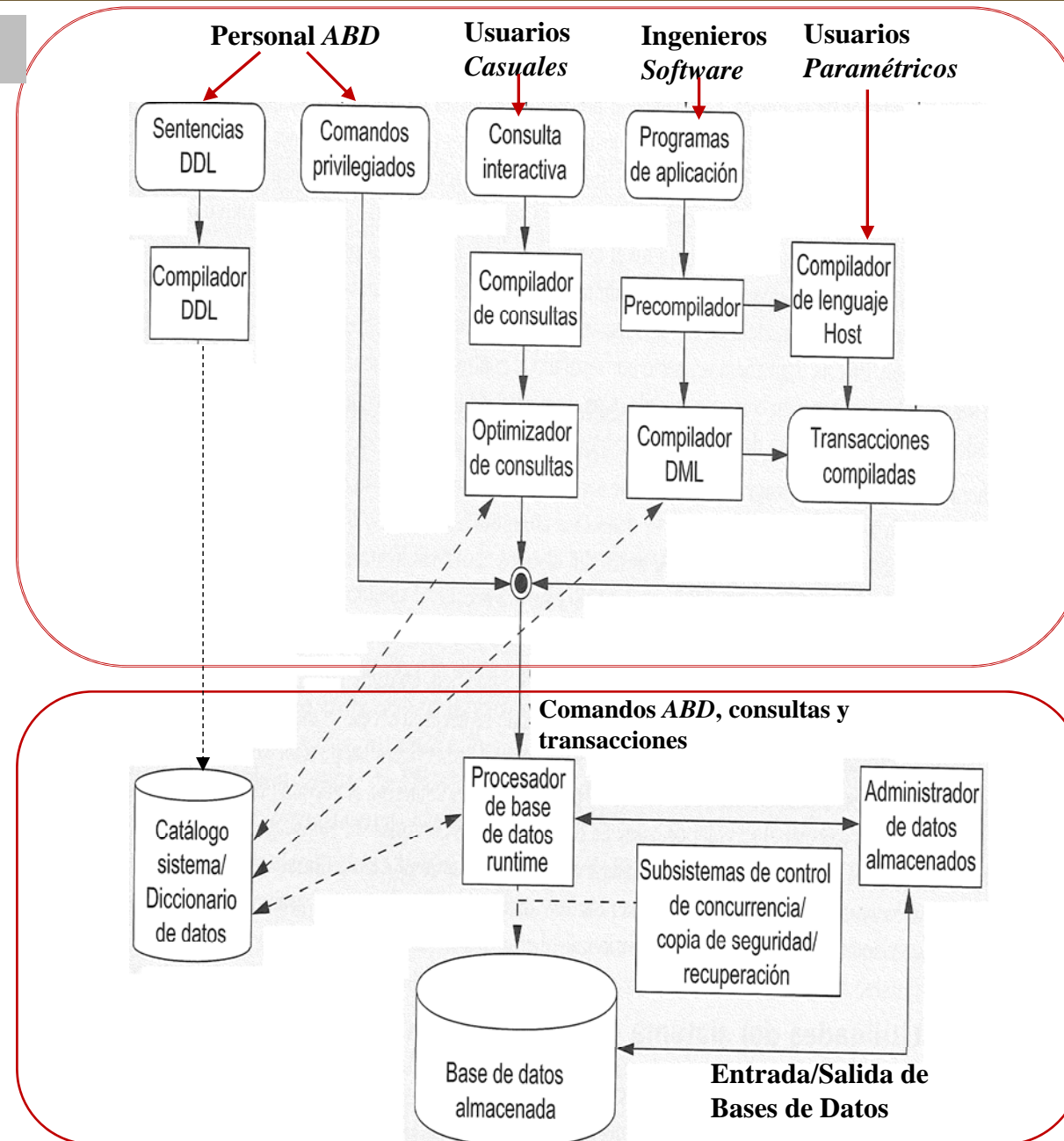
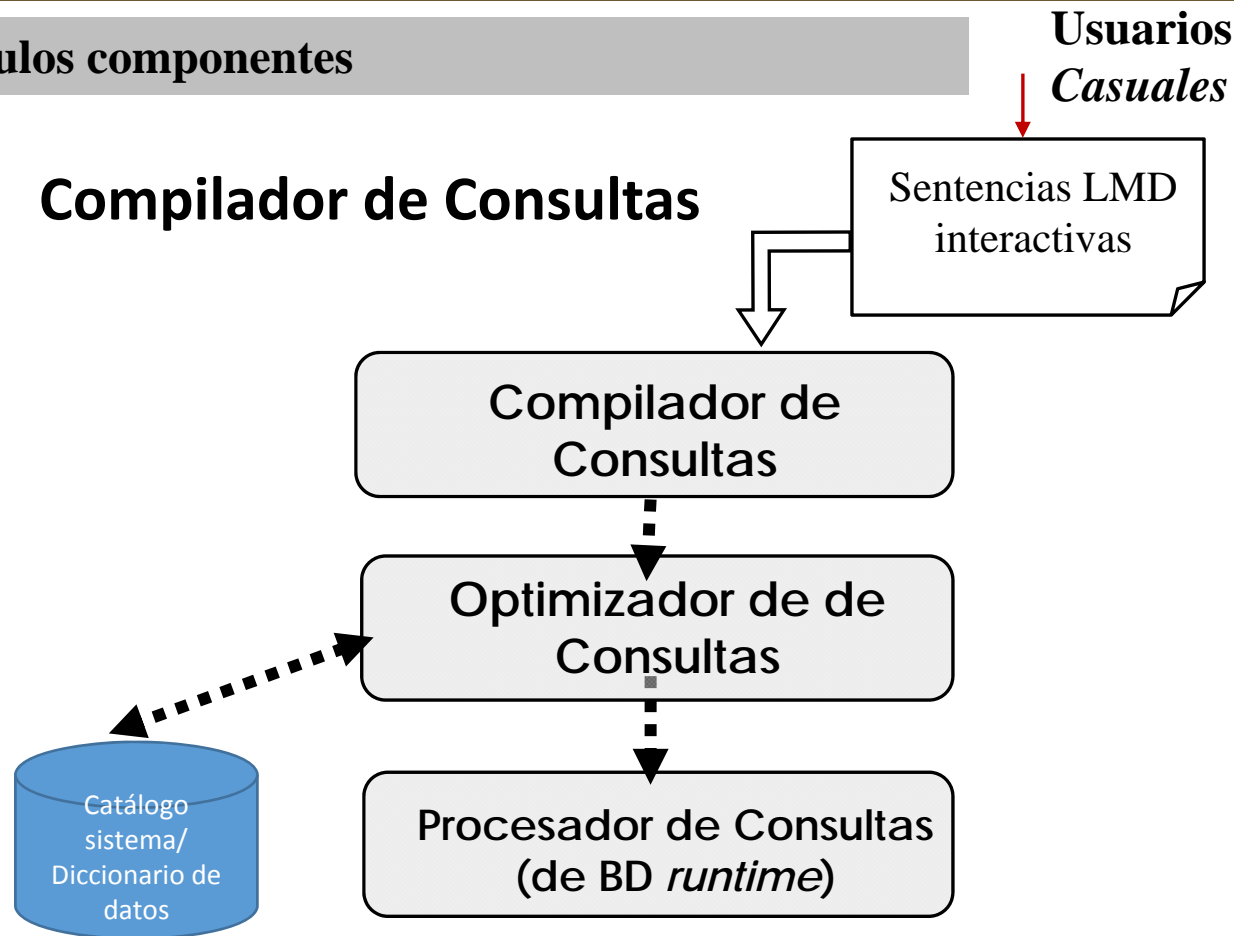


Figura 2.3 del libro **ELMASRI Y NAVATHE**, *Fundamentos de Sistemas de Bases de Datos*, Addison-Wesley, 2007, 5ª edición,

Módulos componentes

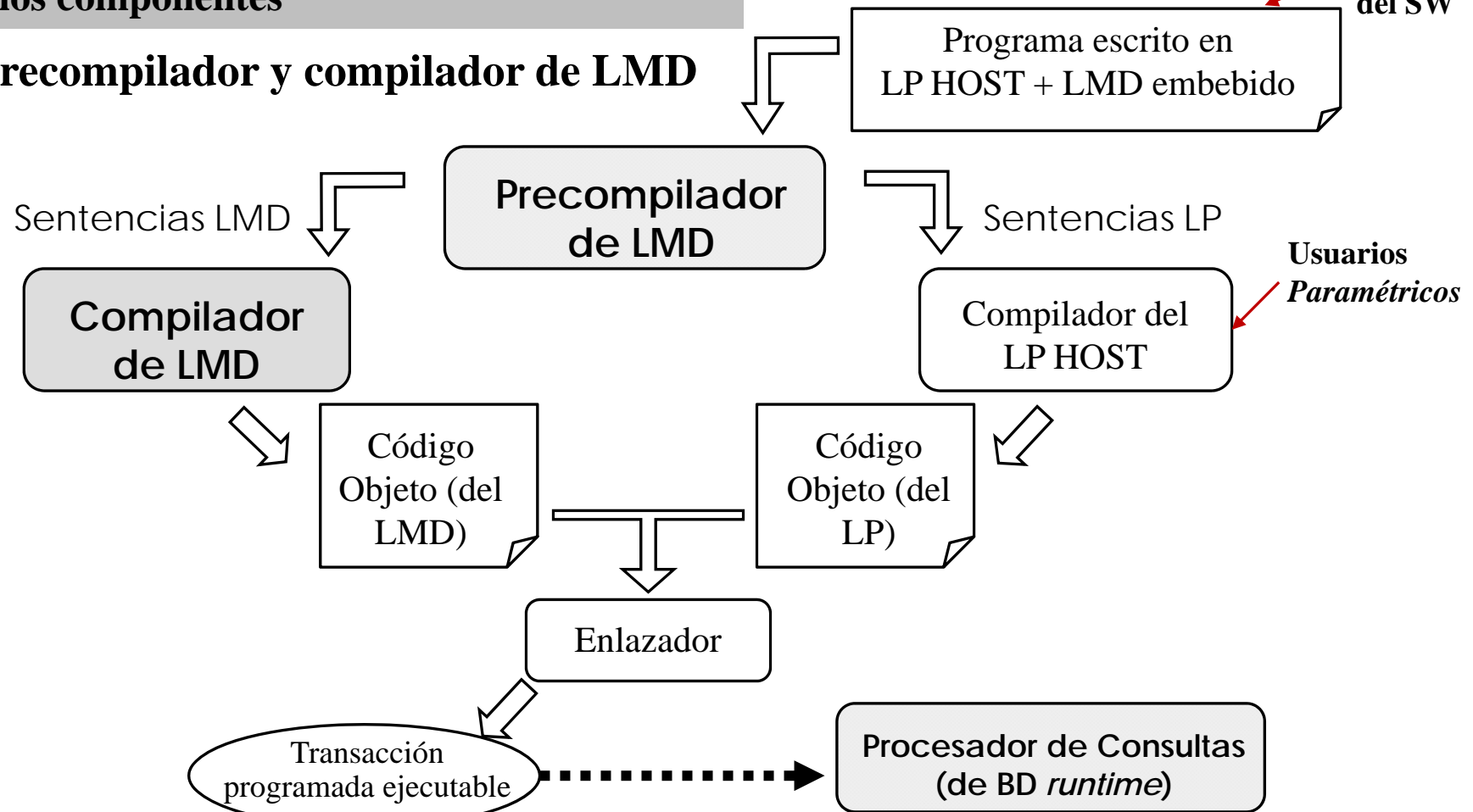
■ **Compilador de Consultas**



- El **compilador de consultas** trata las consultas de alto nivel (escritas en LMD) que se introducen de forma interactiva, analiza su sintaxis y lo compila en un formato interno.
- El genera **optimizador de consultas** se ocupa de la reconfiguración, reordenación de operaciones, eliminación de redundancias y uso de algoritmos e índices correctos durante la ejecución. Consulta el catálogo para información física de los datos almacenados y genera código ejecutable que lleva a cabo las operaciones necesarias para la consulta y realiza la llamada al **procesador de consultas** (procesador de BD runtime).

Módulos componentes

■ Precompilador y compilador de LMD

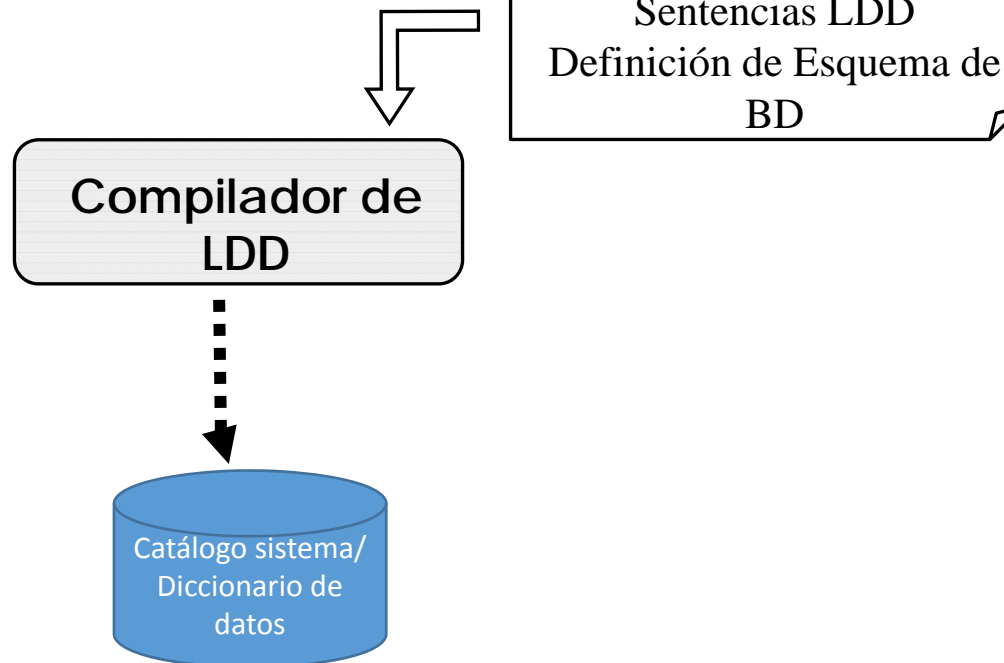


- ✓ El **precompilador de LMD** extrae las sentencias en LMD de un programa escrito en un lenguaje host y las envía al **compilador de LMD**.
- ✓ El **compilador de LMD** las convierte en código objeto (instrucciones de bajo nivel que entiende el *procesador de consultas*) para el acceso a la BD. El **compilador de LMD** optimiza las sentencias: transformarlas en otras equivalentes, pero más eficientes.
- ✓ El resto del programa se envía al *compilador del lenguaje host*.
- ✓ El código objeto de las sentencias LMD se enlaza (*link*) con el código objeto del resto del programa, formando una *transacción programada* cuyo código ejecutable incluye llamadas al *procesador de consultas* de la base de datos.

Módulos componentes

Personal *ABD*

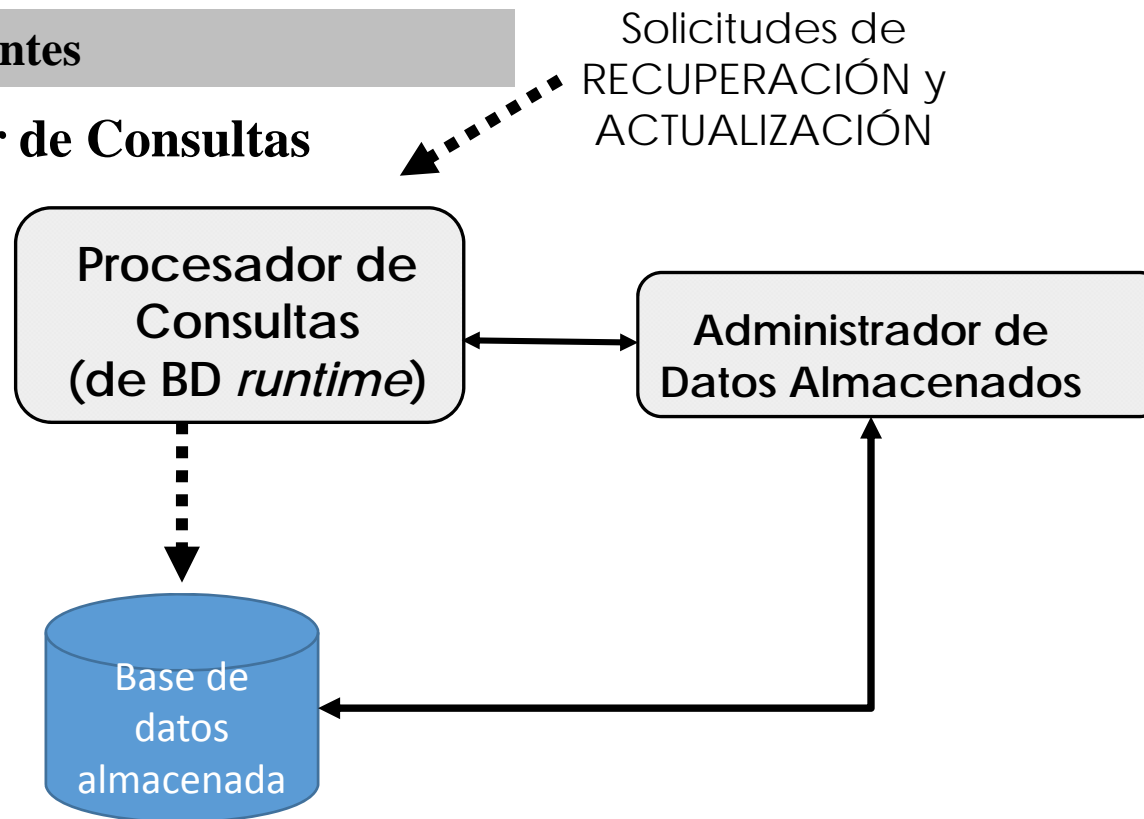
▪ **Compilador de LDD**



- ✓ El **compilador de LDD** procesa las definiciones de esquema escritas en LDD, y almacena las descripciones de los esquemas (metadatos) en el catálogo del SGBD. Otros módulos del SGBD necesitan conocer la información contenida en el catálogo.

Módulos componentes

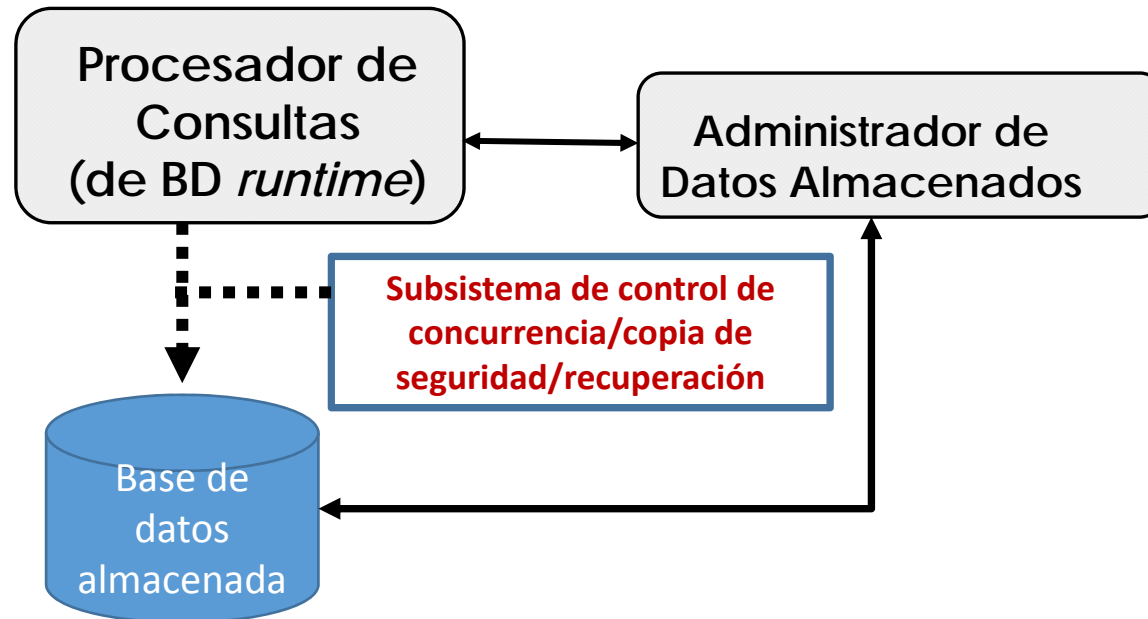
■ Procesador de Consultas



- ✓ El **procesador de consultas** (o procesador de BD) en tiempo de ejecución (*runtime*) se encarga de recibir solicitudes de recuperación o actualización (de bajo nivel, generadas por el compilador de LMD o el optimizador de consultas o de transacciones compiladas), y las ejecuta sobre la base de datos.
- ✓ El acceso a los datos (a disco) se realiza mediante el *Administrador de datos almacenados*.

Módulos componentes

■ Subsistemas de control de...



- **Concurrencia.** Asegurar consistencia y coherencia de datos frente actualizaciones simultáneas de datos por múltiples usuarios
- **Recuperación.** Detectar fallos o caídas del sistema y Restaurar la BD a un estado consistente.
- **Integridad y Seguridad**
 - ✓ ¿Son correctas las actualizaciones de los datos?
 - Detectar violación de Restricciones de Integridad y
 - Realizar acción de recuperación de la integridad de datos
 - ✓ ¿Se cumplen las restricciones de seguridad de acceso?
 - Comprobar las autorizaciones de acceso al SGBD y a los datos

▪ Mención especial al Catálogo del sistema y al Diccionario de datos

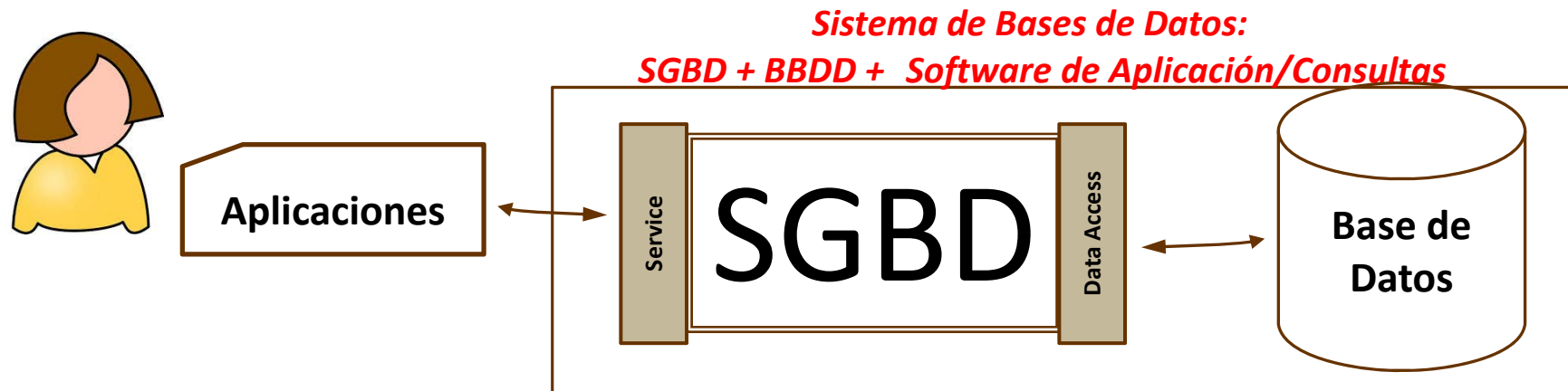
Catálogo del sistema

- El catálogo del sistema es una mini-base de datos que almacena los esquemas (conceptual, externo e interno, así como de las transformaciones entre ellos) de las bases de datos que gestiona el SGBD.
- Cada BD está descrita por los datos almacenados en el catálogo, llamados **metadatos** (datos sobre los datos), que describen su estructura, restricciones, autorizaciones, etc.
- Contiene la información necesaria para los componentes del SGBD relacionados con el procesamiento de consultas, de seguridad y de autorización.

Diccionario de datos

- El diccionario de datos gestiona los **metadatos** del SGBD (es decir, la información contenida en el catálogo) **junto con** otro tipo de información.
- Permite almacenar y controlar:
 - Descripciones de los *esquemas* del SBD
 - Información acerca del *diseño físico* de la BD: estructuras de almacenamiento (tipos de ficheros), estructuras de acceso a los datos, tamaño de los ficheros, registros, etc.
 - Descripción de los *usuarios*, sus *responsabilidades y derechos de acceso*, etc.
 - Descripciones de alto nivel de las transacciones y aplicaciones de la BD, y de las relaciones entre los usuarios y las transacciones.
 - Relación entre las transacciones y la información a la que hacen referencia (consultan o modifican).
 - Cifras estadísticas de uso: frecuencias de consultas, transacciones, nº de accesos a los datos.

Utilidades del Sistema de Bases de Datos



El SGBD es el componente software más importante en un *Sistema de Bases de Datos*, pero existen otras aplicaciones que ayudan al *ABD* a manejar el sistema:

- **Carga de datos**
Intercambio de información entre BD de diferente tipo (Oracle ↔ Access)
- **Respaldo**
- **Reorganización de ficheros** de la BD
Para mejorar rendimiento del sistema
- **Monitorización**
Vigilancia del funcionamiento del sistema y su rendimiento
Datos estadísticos que el **ABD** usa para tomar decisiones de ajuste y mejora del sistema
- **Control de Accesos de usuario**
- **Acceso al Diccionario de datos**
- **Otras utilidades**
ordenamiento, compresión de ficheros, etc...