

# **FBD - Relación de Ejercicios Tema 2: Arquitectura de un SGBD**

**Ismael Sallami Moreno**

**`ism350zsallami@correo.ugr.es`**

**Universidad de Granada**



# Índice general

<b>1. Arquitectura de un SGBD</b>	<b>5</b>
1.1. Relación de Ejercicios T2 . . . . .	5



# Capítulo 1

## Arquitectura de un SGBD

### 1.1. Relación de Ejercicios T2

1. Explicar la relación existente entre los niveles de una base de datos y el concepto de independencia.

La relación entre los niveles de una base de datos y el concepto de independencia radica en la estructura de tres niveles propuesta por el modelo ANSI/SPARC, que busca separar la forma en que los datos se almacenan, representan y acceden. Estos niveles son:

- **Nivel interno (físico):** Describe cómo se almacenan los datos físicamente en el sistema de almacenamiento.
- **Nivel conceptual:** Proporciona una vista abstracta de la base de datos, definiendo la estructura de los datos y sus relaciones sin preocuparse por los detalles de almacenamiento.
- **Nivel externo (vista de usuario):** Presenta a los usuarios una versión personalizada de los datos, mostrando solo la información relevante para ellos.

#### Independencia de los datos

Este concepto se divide en dos tipos:

- **Independencia lógica de los datos:** Se refiere a la capacidad de modificar el esquema conceptual sin afectar los esquemas externos. Por ejemplo, se pueden agregar nuevas entidades o atributos sin necesidad de cambiar la forma en que los usuarios ven los datos.
- **Independencia física de los datos:** Permite cambiar la forma en que los datos están almacenados (como cambiar de disco duro o modificar estructuras de almacenamiento) sin afectar la forma en que los datos son representados en los niveles superiores.

*Podemos concluir con que la arquitectura de tres niveles facilita la independencia de datos, permitiendo modificar estructuras internas o lógicas sin afectar la interacción del usuario con la base de datos.*

2. Explicar la diferencia entre esquema externo y aplicaciones de usuario.

La diferencia entre el esquema externo y las aplicaciones de usuario radica en su propósito y función dentro del sistema de bases de datos.

### Esquema Externo

El esquema externo representa la vista de los datos desde la perspectiva de los usuarios. Cada usuario o grupo de usuarios puede tener un esquema externo diferente, dependiendo de sus necesidades y permisos. Se define a partir del esquema conceptual y no afecta la estructura interna de la base de datos.

Por ejemplo, un *cajero* en un banco solo puede ver los saldos y transacciones de clientes, mientras que un gerente puede acceder a reportes financieros más detallados.

### Aplicaciones de Usuario

Las aplicaciones de usuario son los programas o interfaces que interactúan con la base de datos para realizar operaciones como consultas, inserciones, modificaciones o eliminaciones de datos. Estas aplicaciones pueden utilizar SQL, interfaces gráficas o APIs para comunicarse con la base de datos y presentar los datos según el esquema externo definido para el usuario.

### Diferencia Clave

- El esquema externo es una estructura lógica que define qué datos puede ver un usuario y cómo están organizados.
- Las aplicaciones de usuario son herramientas o programas que permiten interactuar con la base de datos utilizando el esquema externo para presentar la información.

*El esquema externo define la visión restringida de los datos para cada usuario, mientras que las aplicaciones de usuario son los medios mediante los cuales se accede y manipulan esos datos.*

3. Explica el motivo por el que, a tu juicio, no se han desarrollado DDLs a nivel interno.

No se han desarrollado DDLs a nivel interno porque la gestión del almacenamiento físico *es responsabilidad del SGBD*. Esto garantiza eficiencia, independencia de datos, seguridad y portabilidad, evitando que los usuarios interfieran con la optimización del sistema.

4. Explica el motivo por el que, a tu juicio, no se han desarrollado DMLs a nivel externo.

No se han desarrollado DDLs a nivel interno porque el SGBD gestiona automáticamente el almacenamiento físico para optimizar rendimiento, garantizar independencia de datos y evitar configuraciones ineficientes por parte del usuario.

5. Buscar tres ejemplos de lenguajes de cuarta generación. Indicar sus objetivos o funciones.

6. ¿Cuál es el enfoque actual del concepto de lenguaje anfitrión? Dar ejemplos de lenguajes anfitrión.

El lenguaje anfitrión es un lenguaje de programación que integra comandos SQL para interactuar con una base de datos. Actualmente, el enfoque se centra en la integración fluida con los SGBD y la seguridad en la ejecución de consultas.

Ejemplos: Java (JDBC), Python (SQLAlchemy), PHP (PDO), C# (Entity Framework).

7. ¿Qué elementos conciernen al nivel interno de una base de datos?

Incluye estructuras físicas de almacenamiento, organización de archivos, índices, métodos de acceso y estrategias de optimización para el almacenamiento y recuperación de datos.

8. ¿Qué cuestiones debe cubrir a tu juicio una buena herramienta de gestión privilegios de usuarios?

- a) Control de acceso (roles y permisos).
- b) Control y registro de actividad.
- c) Administración centralizada.
- d) Compatibilidad con autenticación externa.
- e) Facilidad de uso y escalabilidad.

9. Explicar las ventajas de la arquitectura cliente/servidor a tres niveles.

- a) Escalabilidad: Mejora el rendimiento al distribuir la carga.
- b) Seguridad: Separa la lógica de negocio y el acceso a datos.
- c) Mantenimiento fácil: Cambios en una capa no afectan las demás.
- d) Optimización de recursos: Reduce la dependencia del cliente en el procesamiento.





# Bibliografía

- [1] Ismael Sallami Moreno, **Estudiante del Doble Grado en Ingeniería Informática + ADE**, Universidad de Granada, 2025.