

Resumen de Diseño y Modelos de Bases de Datos

Síntesis de las Unidades Didácticas 1, 2 y 3

1 de diciembre de 2025

Resumen

Este documento resume los conceptos clave del diseño de Bases de Datos (BBDD), abarcando desde la introducción a sus características y arquitectura (UD1), hasta el detalle del diseño conceptual con el Modelo Entidad/Relación (UD2), y finalmente, el proceso de transformación al diseño lógico y el Modelo Relacional (UD3).

Índice

1 UD1. Introducción a Bases de Datos	2
1.1 Conceptos Fundamentales	2
1.2 Arquitectura y Ciclo de Vida	2
1.3 Explotación y Tendencias	2
2 UD2. Diseño Conceptual: Modelo Entidad/Relación (E/R)	2
2.1 Elementos del Modelo E/R	2
2.2 Cardinalidad y Relaciones	3
2.3 Generalización y Agregación	3
3 UD3. Diseño Lógico: Modelo Relacional	3
3.1 Elementos del Modelo Relacional	3
3.2 Reglas de Transformación E/R a Relacional	4
3.3 Normalización	4

1 UD1. Introducción a Bases de Datos

1.1 Conceptos Fundamentales

Una Base de Datos es un repositorio de datos organizado que permite almacenar y recuperar información de manera eficiente y controlada. Las características principales incluyen:

- **Persistencia:** Los datos se mantienen a lo largo del tiempo.
- **Independencia de datos:** La estructura lógica y física de los datos están separadas.
- **Control de redundancia:** Mecanismos para evitar la duplicidad de información.
- **Integridad de los datos:** Reglas para mantener la calidad y coherencia de los datos.

1.2 Arquitectura y Ciclo de Vida

La arquitectura típica de BBDD incluye el esquema interno (visión física), esquema conceptual (visión lógica global) y esquema externo (visión de usuario). Las arquitecturas comunes son:

- **Cliente/Servidor:** Permite la distribución de tareas.
- **Dos Niveles:** Cliente (lógica de presentación y negocio) y Servidor (BBDD).
- **Tres Niveles:** Cliente (presentación), Servidor de Aplicación (lógica de negocio) y Servidor de BBDD.

El **ciclo de vida** de una BBDD incluye las etapas de: requisitos, diseño conceptual, diseño lógico, diseño físico, implementación, explotación y mantenimiento.

1.3 Explotación y Tendencias

La explotación de datos incluye técnicas avanzadas como:

- **Data Mining:** Descubrimiento de patrones ocultos en grandes volúmenes de datos.
- **Big Data:** Tecnologías para manejar volumen, velocidad y variedad masivos de datos.
- **Business Intelligence (BI):** Uso de datos para la toma de decisiones estratégicas.

2 UD2. Diseño Conceptual: Modelo Entidad/Relación (E/R)

El Diseño Conceptual busca traducir los requisitos del mundo real a un esquema formal, independiente del SGBD a utilizar. El Modelo E/R es la herramienta principal.

2.1 Elementos del Modelo E/R

- **Entidades:** Objetos o conceptos del mundo real con existencia propia (se representan con rectángulos).
- **Atributos:** Propiedades que describen a una entidad (se representan con elipses).

- **Clave Primaria (PK):** Atributo o conjunto de atributos que identifica únicamente una instancia de la entidad (subrayado).
 - **Simples/Compuestos:** Atributos que no se pueden dividir vs. aquellos que sí.
 - **Monovaluados/Multivaluados:** Tienen un solo valor vs. múltiples valores.
 - **Derivados:** Se calculan a partir de otros (línea discontinua).
- **Interrelaciones:** Asociaciones entre dos o más entidades (se representan con rombos).

2.2 Cardinalidad y Relaciones

La **cardinalidad** define el número de instancias de una entidad que pueden asociarse con instancias de otra.

- **Uno a Uno (1:1):** Una instancia de A se relaciona con, a lo sumo, una de B.
- **Uno a Muchos (1:N):** Una instancia de A se relaciona con cero o varias de B.
- **Muchos a Muchos (N:M):** Una instancia de A se relaciona con cero o varias de B, y viceversa.

Una **Entidad Débil** depende para su identificación de la Clave Primaria de otra entidad (**Entidad Fuerte**).

2.3 Generalización y Agregación

- **Generalización/Especialización:** Representa la herencia de propiedades, donde un Supertipo engloba subtipos con características propias (P.e., PERSONA es supertipo de CLIENTE y EMPLEADO).
- **Agregación:** Trata una relación como si fuera una entidad de orden superior para establecer otra relación con ella.

3 UD3. Diseño Lógico: Modelo Relacional

El Diseño Lógico transforma el esquema conceptual (E/R) a la estructura lógica del SGBD elegido, siendo el **Modelo Relacional** el estándar.

3.1 Elementos del Modelo Relacional

La estructura básica es la **relación** o **tabla**, compuesta por:

- **Tuplas:** Filas o registros.
- **Atributos:** Columnas.
- **Clave Primaria (PK):** Atributo(s) que identifica una tupla.
- **Clave Ajena (FK):** Atributo(s) en una tabla que apunta a la PK de otra tabla, estableciendo la conexión entre relaciones.

3.2 Reglas de Transformación E/R a Relacional

La clave del Diseño Lógico es aplicar reglas de transformación sistemáticas:

- **Entidades:** Cada entidad se convierte en una tabla. Sus atributos simples pasan a ser columnas, incluyendo la PK.
- **Relaciones 1:1 o 1:N:** Se propagan las FK. La PK de la entidad del lado “1” se añade como FK a la entidad del lado “N” (o a cualquiera en 1:1, con restricción de unicidad).
- **Relaciones N:M:** Se requiere la creación de una **nueva tabla intermedia**. La PK de esta nueva tabla será la composición de las PK de las dos entidades relacionadas (que se convierten en FK en la tabla intermedia).
- **Entidades Débiles:** Se transforman en una tabla cuya PK es la composición de la PK de la entidad fuerte (como FK) y su propio discriminador (si existe).
- **Generalización (Herencia):** Existen varias estrategias, como:
 - **Opción 1 (Una tabla por Supertipo y Subtipo):** La tabla Supertipo contiene atributos comunes; cada tabla Subtipo contiene sus atributos específicos y la FK/PK del Supertipo.
 - **Opción 3 (Una sola tabla para todos):** Se crea una tabla grande con todos los atributos (comunes y específicos) más un campo discriminador, permitiendo valores nulos en los atributos específicos.

3.3 Normalización

La normalización es el proceso de organizar las columnas y tablas de una BBDD relacional para minimizar la redundancia y dependencia de datos. Esto se logra aplicando las **Formas Normales (FN)**, siendo las más comunes:

- **1FN:** Los atributos deben ser atómicos (no multivaluados ni compuestos).
- **2FN:** Debe estar en 1FN y todo atributo no clave debe depender de **toda** la PK compuesta.
- **3FN:** Debe estar en 2FN y los atributos no clave no deben depender de otros atributos no clave (dependencia transitiva).