

# T2: Diseño



# TEMA 1: Diseño BBDD

## Diseño Conceptual

- Descripción de alto nivel de la información que debe almacenar la BBDD
- Independiente del SGBD
- Modelo Entidad - Relación (ER - entity-relationship)

## Diseño Lógico

- Descripción de la estructura de la BBDD según el modelo (tipo) del SGBD
- Modelo relacional
- Normalización

## Diseño Físico

- Descripción de la implementación de la BBDD en disco
- Depende totalmente del SGBD elegido
- SQL

## Modelo Entrada - Relación (ER)

**Modelo de datos:** colección de herramientas conceptuales para descripción de los elem. de la realidad q intervienen en un problema, dado (datos)

relaciones entre datos  
semántica de datos  
restrictiones de integridad q garantizan la consistencia de los datos

**Modelo E-R:** modelo de datos de Alto Nivel → base  
Real **representa** relación de entidades + relaciones entre ellas

**Entidades**  
puede ser concreta o abstracta.

Entidad → conjunto de Propiedades → identificar entidad (nombres)  
Conjunto de Entidades → entidades del mismo tipo

**Atributos** → (propiedades entidad)

conjunto entidades → = atributos

dominio → conjunto de valores attr. reunidos

tipos → simples y compuestos

multivariados y multivalorados

derivados → edad = fecha nacimiento

## Relaciones

asociación o conexión semántica entre ≠ entidades

- Pueden realizarse entre 2 o más ent. (recomendado: solo 2)
- $f(r) = "R1" \& "R2"$
- Grado → nº entidades q participan
  - grado 1 (unaria - reflexiva)
  - grado 2 (binaria)
  - grado 3 (ternaria)
  - ...

**Cardinalidad:** restricción q permite indicar el número de elementos de una entidad → puede estar restringido un elem. de otra entidad

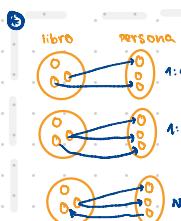
Card. Máx: indica nº máx relaciones posibles (1, N ó \*)  
Card. Mín: indica nº relaciones posibles (0, 1)

Relaciones grado 2 → Card. Max
 

- Uno a Uno (1:1) → solo se puede relacionar un elem. de otra entidad
- Uno a Varios (1:N) → elem. entidad: relacionar con varios
- Varios a Varios (N:M) → elem. entidad: relacionar con varios y viceversa

## Ejemplo:

- Los alumnos se **matrículan** de varias asignaturas (al menos una)
- Una asignatura puede tener varios alumnos matriculados
- Todo profesor **pertenece** a un departamento concreto
- Todo departamento **debe tener** un director, que es un profesor
- Un profesor puede **impartir** varios grupos de la misma asignatura (o de asignaturas diferentes)
- Un grupo de una asignatura ha de **estar impartido**, al menos, por un profesor
- Las clases de cada asignatura se **imparten** en días, horas y aulas determinadas
- Los atributos de cada entidad son "los habituales"



Entidades	Atributos	Asignatura	Alumno	Profesor	Departamento	Aula	Grupo
Atributos	ID Nombre Créditos Carácter Curso	ID Nombre Categoría Área	ID Nombre Dirección Email	ID Nombre Categoría Área	ID Nombre	ID Capacidad	ID Tipo
Relación	Entidades participantes	Cardinalidad	Atributos				
se matrícula en	Alumno - Grupo	N:M	Calificación				
enseña	Profesor - Grupo	N:M					
Impartida en	Asignatura - Grupo	1:N					
asignada a	Aula - Grupo	N:M	Día, hora				
pertenece a	Profesor - Departamento	N:1					
dirige	Profesor - Departamento	1:1					

## Modelado Conceptual de BBDD

## Claves

Claves → Conjunto de attr q permiten identificar una entidad

## Tipos de claves:

- SuperClave** → identificar una entidad dentro de un conjunto entid.
- Clave Candidata** → Superclave con nº min. attr.
- Clave Primaria (PK)** → clave candidata elegida diseñador
- Clave Alternativa** → cualquier clave candidata no elegida en PK
- Clave Extranja o ajena (FK)** → formada 1 o más attr de otra tabla cuyos valores coinciden con la Clave Primaria de otra tabla [Integridad Referencial]
- Clave Subrogada (surrogate key)** → identificador único q origina automáticamente el SGBD

## Ejemplo:

Persona (Nombre) no es una clave  
(DNI) es una clave candidata  
(DNI, Nombre) es una superclave  
(Nombre, Dirección) podría serlo (lo no?)

Película (Título) no es una clave  
(Título, género, año) podría serlo  
(Título, director, año) es una clave (candidata?)

## Entidades Fuertes y Débiles

conjunto de entidades q tienen clave primaria → entidades fuertes (o regulares)

no tienen suficientes attr para formar clave primaria  
conjunto entidades → no tienen 3 o más participar en la relación ⇒ entidades débiles

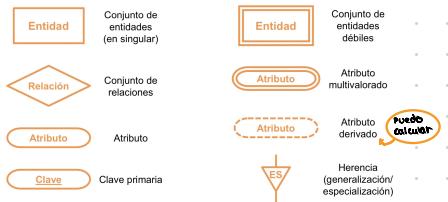
Datos de un préstamo, nº de edición de un libro, ...

# TEMA 1: Diseño BBDD

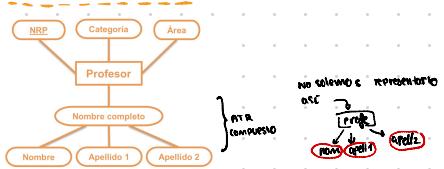
## Modelado Conceptual de BBDD

### Diagrama Entidad-Relación

#### Diagrama Entidad-Relación → representación gráfica



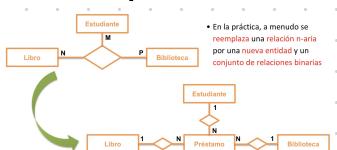
#### Representación de E-R



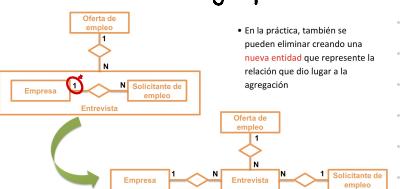
#### Representación de Relaciones:

- Relación uno a uno
  - Relación uno a muchos
  - Relación muchos a muchos
  - Cardinalidad mínima
    - Relación obligatoria: Una cuenta ha de tener un titular como mínimo
    - Relación opcional: Un cliente puede ser o no titular de una cuenta
- \* 1 cliente es titular de 0 o más cuentas (cuentas sin clientes vv)

#### → de n-arias (grado relación → 2, 3, 4, ...)



#### agregaciones (relaciones entre relaciones o entre relaciones y conjuntos de entidades)



#### Reflexivas, e involutivas (relaciones de un conjunto de entidades conigo mismo)



\* card máx

### Modelo Entidad-Relación extendido

- Modelo E-R → limitaciones semánticas: entidades fuertes y débiles, cardinalidad relaciones (tanto máx como min).
- Modelo E-R extendido
  - Herencia
  - Aggregación
  - Relación

Hijo hereda attr y relaciones hijo

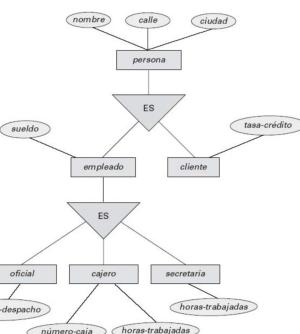


**Supertipo:** tipo de entidad q incluye 2 o más subtipos  
Subtipos ≠ de ocurrencias q deben ser representados

**Subtipo:** cada uno de los subgrupos de ocurrencias

**Especialización:** proceso de designación de subgrupos de dentro → conjunto de entidades  
Diseño **Top-down** o **descendente** **es un**

**Generalización:** proceso de síntesis en un conjunto de entidades de nivel más alto  
Diseño **Bottom-up** o **ascendente**



#### Pasos para la realización del diseño (esquema) conceptual

- Identificar las entidades (sustantivos?)
- Identificar las relaciones (verbos?)
- Identificar los atributos y asociarlos a entidades y relaciones
- Determinar los dominios de los atributos
- Determinar las claves
- Determinar las jerarquías de generalización (si las hay)
- Dibujar el diagrama Entidad – Relación
- Revisar el esquema conceptual con el usuario

# TEMA 1: Diseño BBDD

## Diseño Relacional de BBDD

### Modelo Relacional

- de base matemática (**álgebra relacional**) lenguaje de predicados teoría de conjuntos
- Relaciones → tablas/entities
- Tuplas + registros con campos (columnas)
- Consultas utilizando **Lenguajes de Alto Nivel** (SQL)

### Ventajas

minimiza Redundancia → mecanismos evitar duplicidad/registros  
Integridad Referencial

- Interior registro → algo q es coherente
- eliminar registro → eliminan registros dependientes

Favorece Normalización

- Datos almacenados en relaciones

- Relación → Tabla/conjunto filas
- Fila: conjunto campos → registros o Tuplas
- Cada campo → MTR

### Campos

DNI	Nombre	Apellido	Edad	Población	Sueldo
11223344	Luis	López	35	Madrid	1250
55667788	Angela	Pérez	56	Madrid	3500
99001122	Antonio	Camino	45	Sevilla	2150
33445566	Susana	Sánchez	24	Toledo	930
77889900	Alberto	Jiménez	62	Madrid	1560

### Registros o Tuplas

### Valores

- Esquema identificador nombre de la relación  
Nombre de los campos indicando cuál es su clave Primaria

EMPLEADO (DNI, NSS, nombre, apellidos, sueldo)

COCHE (matrícula, marca, modelo, color, precio)

CUENTA (banco, sucursal, dc, número, tipo, saldo)

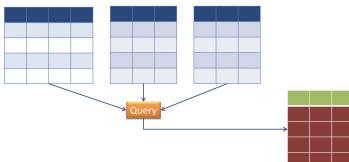
### Álgebra Relacional

- Lenguaje de consulta procedimental
- Hay **Formal** → basado Lógica conjuntos (matemática discreta)
- base SQL

### Operaciones sobre conjuntos

- Unión: tablas = esquema
- Diferencia: tablas = esquema
- Intersección: tablas = esquema
- Producto cartesiano: cualquier tabla
- Proyección: extraer columnas
- Selección: extraer una fila
- Unión: Unir dos tablas en una
- División: expresiones "para todo"

Info devuelta por consulta → son relaciones (tablas)



### Transformación Relación n-Arias

- Genera tabla para cada entidad
- Nueva Tabla adicional → relación
- Nueva Tabla
  - los claves primarias de todas las entidades que asocia la relación
  - los propios de la relación

### Entidad

ESTUDIANTE (DNI, ....)

ASIGNATURA (Código, ....)

EVALUACION (DNI, Código, nota, ....)

### Paso a tablas

CONVERSIÓN: diseño conceptual (Modelo ER) diseño lógico (Modelo Relacional)

- Objetivo: menor redundancia y máxima coherencia

### Transformación de Entidades



Si la entidad es débil  
+ añade clave de la entidad débil

clave foránea no admite valores nulos

### Transformación de Relaciones Binarias

#### Cardinalidad máx 1:1

- No se genera tabla nueva
- Hay q considerar card mínimas

card. min = [0:0] (1:1)

clave entidad clave foránea entidad  
si 3 attr de la relación → añaden entidad "destino"  
card. min ≠ [0:1], (1:0]

clave entidad card 1 clave foránea attr 0  
si 3 attr de la relación → añaden entidad card 0

#### Cardinalidad máx 1:N - N:1

- No se genera nueva tabla
- clave entidad con card 1 clave foránea ent. card N
- Si 3 attr de la relación → añaden entidad card N

#### Cardinalidad máx M:N

- Nueva tabla
- claves primarias q asocia relación
- los propios de la relación



ESTUDIANTE (DNI, ....)

ASIGNATURA (Código, ....)

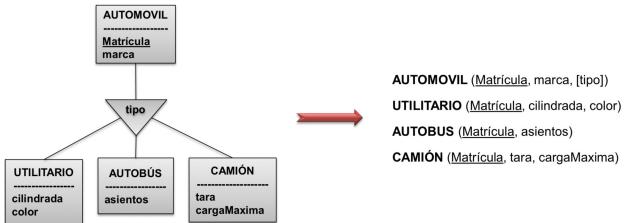
EVALUACION (DNI, Código, nota, ....)

genera una nueva tabla

# TEMA 1: Diseño BBDD

## Paso a tablas

### Transformación de jerarquías de Generalización/Especialización



- Tabla → Superclase
- Tabla → cada Subclase  $\Rightarrow$  clave primaria  $\rightarrow$  clave Superclase

1º Forma Normal: eliminar valores repetidos

invoice_id	date	client_id	client_name
001	2014-09-17	C01	Dracolienda
002	2014-09-17	C02	Tesoros de la Marca
003	2015-01-10	C01	Dracolienda

invoice_id	product_id	product_name	product_price	tax_rate	amount
001	ISH01	La Puerta de Istar	38.00€	4%	10
001	ISH02	Istar: Pantalla	19.50€	21%	5
002	ISH01	La Puerta de Istar	38.00€	4%	7
002	ABL01	Ablandada	14.00€	4%	3
003	ABL01	Ablandada	14.00€	4%	10
003	ABL02	Relatos de Ablandada	9.50€	4%	5

2º Forma Normal: 1FN + Todos los atr deben depender solo de la clave primaria

Base de Datos en 2FN

invoice_id	date	client_id	client_name
001	2014-09-17	C01	Dracolienda
002	2014-09-17	C02	Tesoros de la Marca
003	2015-01-10	C01	Dracolienda

product_id	product_name	product_price	tax_rate
ISH01	La Puerta de Istar	38.00€	4%
ISH02	Istar: Pantalla	19.50€	21%
ABL01	Ablandada	14.00€	4%
ABL02	Relatos de Ablandada	9.50€	4%

## Normalización

- Proceso de organizar atr y tablas (✓ redundancias)
- tablas Grandes  $\rightarrow$  tablas pequeñas
- se busca tb  $\rightarrow$  problemas actualización
- se busca tb  $\rightarrow$  Evita perder info.  
Protege Integridad datos

### Dependencia Funcional (DF)

conexión entre 1 o más atrs

DO SI YR CANCE 1  $\Rightarrow$  se borran los otros  
BNC nombre y apellido dos

Ejemplo: BD sin Normalizar

invoice_id	date	client_id	client_name	product_id	product_name	product_price	tax_rate	amount
001	2014-09-17	C01	Dracolienda	ISH01	La Puerta de Istar	38.00€	4%	10
				ISH02	Istar: Pantalla	19.50€	21%	5
002	2014-09-17	C02	Tesoros de la Marca	ISH01	La Puerta de Istar	38.00€	4%	7
				ABL01	Ablandada	14.00€	4%	3
003	2015-01-10	C01	Dracolienda	ABL01	Ablandada	14.00€	4%	10
				ABL02	Relatos de Ablandada	9.50€	4%	5

## Formas Normales (FN)

det. grado Vulnerabilidad de una Base Datos a inconsistencias lógicas

- ↳ BD está Junto normal  $\times$  (FN)  $\rightarrow$  todas sus tablas: Formas Normales
- ↳ Varios FN para indicar el grado

- 1FN (Primera Forma Normal)
- 2FN (Segunda Forma Normal)
- 3FN (Tercera Forma Normal)
- FNBC (Forma Normal de Boyce-Codd)
- 4FN (Cuarta Forma Normal)
- 5FN (Quinta Forma Normal)



(Ciudad, nombre, País, continente)

nombre  $\rightarrow$  País, País  $\rightarrow$  continente ; nombre  $\rightarrow$  continente XX

3º Forma Normal: 2FN + Dependencia Funcional transitoria

Base de Datos en 3FN

client_id	client_name
C01	Dracolienda
C02	Tesoros de la Marca

product_id	product_name	product_price	tax_rate
ISH01	La Puerta de Istar	38.00€	4%
ISH02	Istar: Pantalla	19.50€	21%
ABL01	Ablandada	14.00€	4%
ABL02	Relatos de Ablandada	9.50€	4%

invoice_id	date	client_id
001	2014-09-17	C01
002	2014-09-17	C02
003	2015-01-10	C01

invoice_id	product_id	amount
001	ISH01	10
001	ISH02	5
002	ISH01	7
002	ABL01	3
003	ABL01	10
003	ABL02	5