

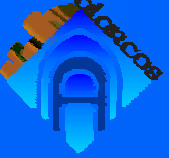
# Diseño y Creación de BD

Universidad de Castilla-La Mancha

4

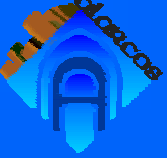
## Reglas para el Diseño

*Francisco Ruiz*  
*octubre-noviembre 2002*



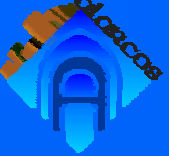
# Contenidos

- Reglas para el Diseño Inicial.
- Reglas básicas.
  - Clases de objetos.
  - Atributos de los objetos.
  - Asociaciones entre clases de objetos.
- Reglas complementarias.
  - Atributos que pueden tener varios valores a la vez.
  - Dependencia existencial.
  - Generalización / Especialización.
  - El Factor Tiempo.
- Reglas para el Diseño Detallado.



# Reglas para el Diseño Inicial

- Nos dicen cómo producir los documentos del Diseño Inicial: el diagrama y la lista de tablas.
- Básicas.
  - Referidas a los conceptos manejados en todas las BD: clases de objetos, atributos y asociaciones.
- Complementarias.
  - Nos dan una receta para situaciones especiales. Sólo incluimos las más importantes.



# R1- Clases de objetos

- Cada clase de objetos se representa como una tabla (*clasificación*).

– Ejemplos:

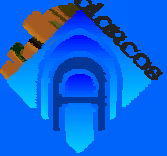
» **Persona.**

» Factura.

» Territorio.

» Vehículo.

» ...



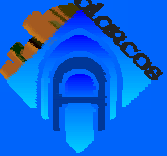
## R2- Atributos de los objetos

- Se representan como campos de la tabla correspondiente (*agregación*).
  - Ejemplo: una Persona tiene DNI, nombre y edad.

⇒

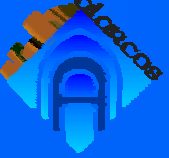
la tabla Persona tiene los campos DNI, nombre y edad.

Persona(dni, nombre, edad)



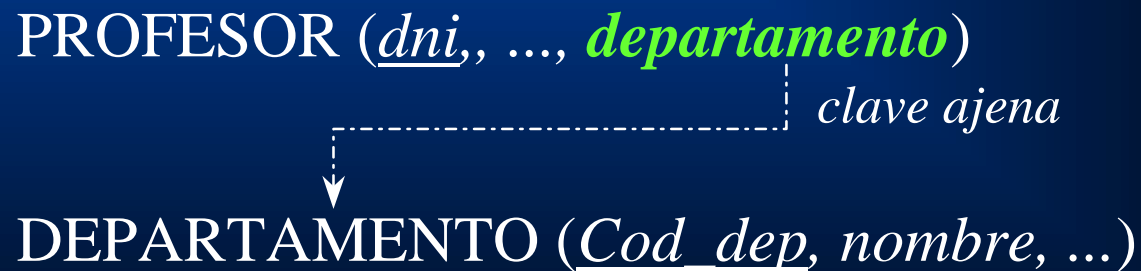
## R3 – Asociaciones entre clases de objetos

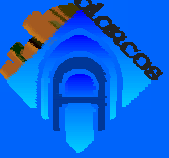
- Para cada asociación entre dos clases de objetos, se estudia su tipo:
  - 1 a muchos  $\Rightarrow$  Regla R3a.
  - Muchos a muchos  $\Rightarrow$  Regla R3b.
- No utilizar asociaciones entre más de 2 clases de objetos.
  - Son muy difíciles de manejar.
  - Siempre se pueden sustituir por varias asociaciones entre 2 clases de objetos.



## R3a – Asociaciones 1 con muchos

- Se representan añadiendo la clave primaria de la tabla del lado del 1 (tabla A) a la del lado del muchos (tabla B) y haciendo que sea clave ajena apuntando desde B a A.
  - Ejemplo: A cada Departamento pertenecen varios Profesores. Un Profesor sólo pertenece a un Departamento.





## R3b – Asociaciones muchos con muchos

- Se representan mediante dos asociaciones 1 a muchos entre las tablas originales y una nueva tabla intermedia que incluye ambas claves primarias.

– Ejemplo: Un Profesor puede impartir varias asignaturas. Una Asignatura la pueden impartir varios Profesores.

PROFESOR (dni, ...)

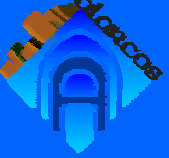
*Clave ajena*

ASIGNATURA (Cód\_asi, ...)

*Clave ajena*

**IMPARTE (asignatura, profesor)**

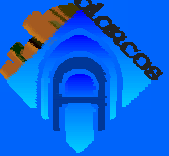




# R4 – Asociaciones muchos con muchos

- Algunas veces, las asociaciones pueden tener propiedades.  
=> Se añaden como campos en la tabla intermedia correspondiente.

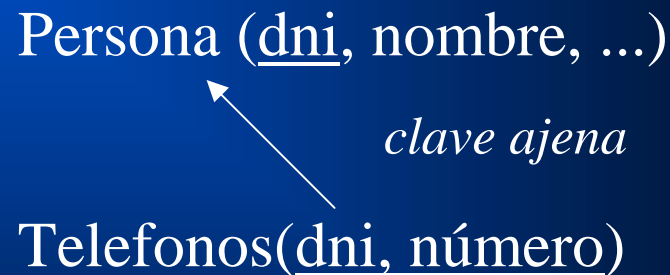




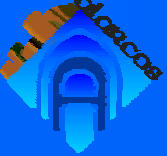
# RC1 - Atributos que pueden tener varios valores a la vez

- Se representan como una tabla que apunta a la tabla original.
  - Ejemplo: Una Persona puede tener varios teléfonos.

⇒



–Los únicos campos (y clave primaria) de la nueva tabla son la clave primaria de la tabla original y el atributo especial. Además, se debe crear una clave ajena apuntando a la tabla original.



## RC2 – Dependencia existencial.

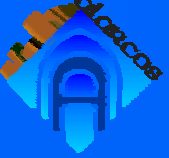
- Los objetos de la clase B sólo pueden existir si existe otro objeto de la clase A con el que están asociados (*tipo especial de asociación 1 a muchos*).

=> La clave primaria de la tabla B incluye a la clave primaria de la tabla A y se crea una clave ajena apuntando desde B a A.

LIBRO (*código*, título, ...)

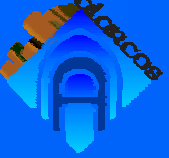


EJEMPLAR (*código*, *número-ejemplar*, *signatura* ...)



## RC3 – Generalización / Especialización

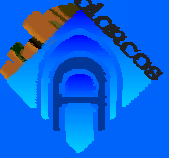
- Las clases de objetos B1, B2, ... son especializaciones de la clase A.
- Ejemplos:
  - Profesor, Alumno y PAS son especializaciones de Persona.
  - Vehículo es una generalización de automóvil, tractor, bicicleta, carro y patinete.
  - Libro, Apuntes, Revista, Página web son especializaciones de Documento.



## RC3 – Generalización / Especialización

### ■ Propiedades:

- Los objetos de una clase  $B_i$  también son a la vez objetos de la clase  $A$ .
  - » Luis es Profesor  $\Rightarrow$  Luis es Persona.
- Los objetos de una clase  $B_i$  tienen los atributos de  $A$  más los suyos propios.
  - » Si Persona tiene dni, nombre y edad  $\Rightarrow$  Luis tiene dni, nombre y edad.
  - » Luis es Profesor  $\Rightarrow$  Luis tiene departamento y centro.

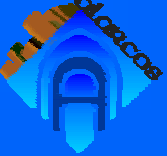


# RC3 – Generalización / Especialización

## ■ Representación:

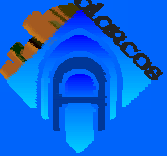
- Cada tabla de cada Bi tiene la misma clave primaria que A, que a la vez es clave ajena apuntando de Bi a A.





## RC4 – El Factor Tiempo

- En función de la manera de considerar la evolución a lo largo del tiempo existen dos opciones para diseñar las BD:
  - a) Sólo interesa conocer el estado actual de cada objeto.
    - » ¿está prestado este libro?, ¿a quién?
  - b) Interesa conocer el estado en cualquier momento actual o pasado (histórico).
    - » ¿qué prestamos se hicieron de este libro el año 1998?, ¿quién tenía el libro el 16-10-1997?, ¿cuánto tiempo lo tuvo prestado?



# RC4 – El Factor Tiempo

## ■ Representación:

– Instantes o momentos: mediante un campo de tipo fecha o similar.

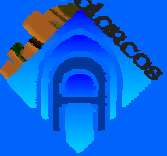
Persona(dni,nombre, **fecha-nacimiento**, ...)

– Intervalos de tiempo: dos campos de fecha o similares, uno para el inicio y otro para el final.

» Prestamo(...,**fecha-inicio,fecha-fin**, ....)

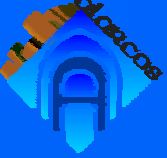
» Reunión(..., **hora-inicio, hora-fin**, ...)





## RC4 – El Factor Tiempo

- Cuidado con las claves primarias.
  - Préstamo(libro, usuario, fecha-ini, fecha-fin, ...)
    - ⇒ Sólo puede tener un único préstamo para cada pareja libro-usuario.
    - ¿Y si quiero que cada usuario puede sacar prestado el mismo libro más de una vez?. No puedo.
- En las tablas históricas es necesario incluir un campo temporal en la clave primaria:
  - Préstamo(libro, usuario, fecha-ini, fecha-fin, ...)
    - Ahora sí puedo hacerlo.



# Reglas para el Diseño Detallado

- Nos dicen cómo crear el diseño detallado de la BD en el ordenador.
- Se refieren a aspectos tales como ...
  - Elegir el tipo de dato de cada campo.
  - Decidir si un campo admite nulos.
  - Decidir si una relación exige integridad referencial.
  - Etc.
- Dependen del SGBD utilizado (p.e. MS Access).
- En realidad, se refieren a la manera de utilizar cada SGBD (programa) para crear BD's.
  - Para MS Access, utilizar el libro digital entregado.