



Dependencias Funcionales y Normalización

Diseño Relacional

- El diseño de una BD Relacional puede abordarse de **dos formas** distintas:
 - **Obteniendo el esquema relacional directamente:**
Objetos y reglas captadas del análisis del mundo real, representadas por un conjunto de esquemas de relación, sus atributos y restricciones de integridad.
 - **Realizando el diseño del esquema "conceptual" de la BD (modelo E/R(E)) y transformándolo a esquema relacional.**

Ejemplo de diseño inadecuado

Pedidos

<u>Artículo</u>	<u>cliente</u>	cantidad	precio	ciudad	distancia
A1	C1	12	100	Madrid	400
A1	C2	30	100	Valencia	200
A1	C3	15	100	Alicante	80
A2	C1	35	250	Madrid	400
A2	C2	20	250	Valencia	200
A2	C4	10	250	Madrid	400
A3	C3	25	175	Alicante	80

Ejemplo de diseño inadecuado

- **Duplicación de información:** ciudad, distancia (ciudad); precio (artículo).
- **Anomalías de modificación:** !podemos tener el mismo artículo con dos precios! (igual argumento para ciudad y distancia). → **inconsistencias**
- **Anomalías de inserción:** ¿Podemos registrar nuevo artículo?, ¿Nuevo cliente?, ¿Nueva ciudad, distancia?
- **Anomalías de borrado:** Si eliminamos tupla de pedido de artículo A3 o cliente C4 → **pérdida de información.**

Ejemplo de diseño inadecuado

- **Redundancia de información:**

- datos repetidos
- desperdicio de espacio,
- actualización complicada

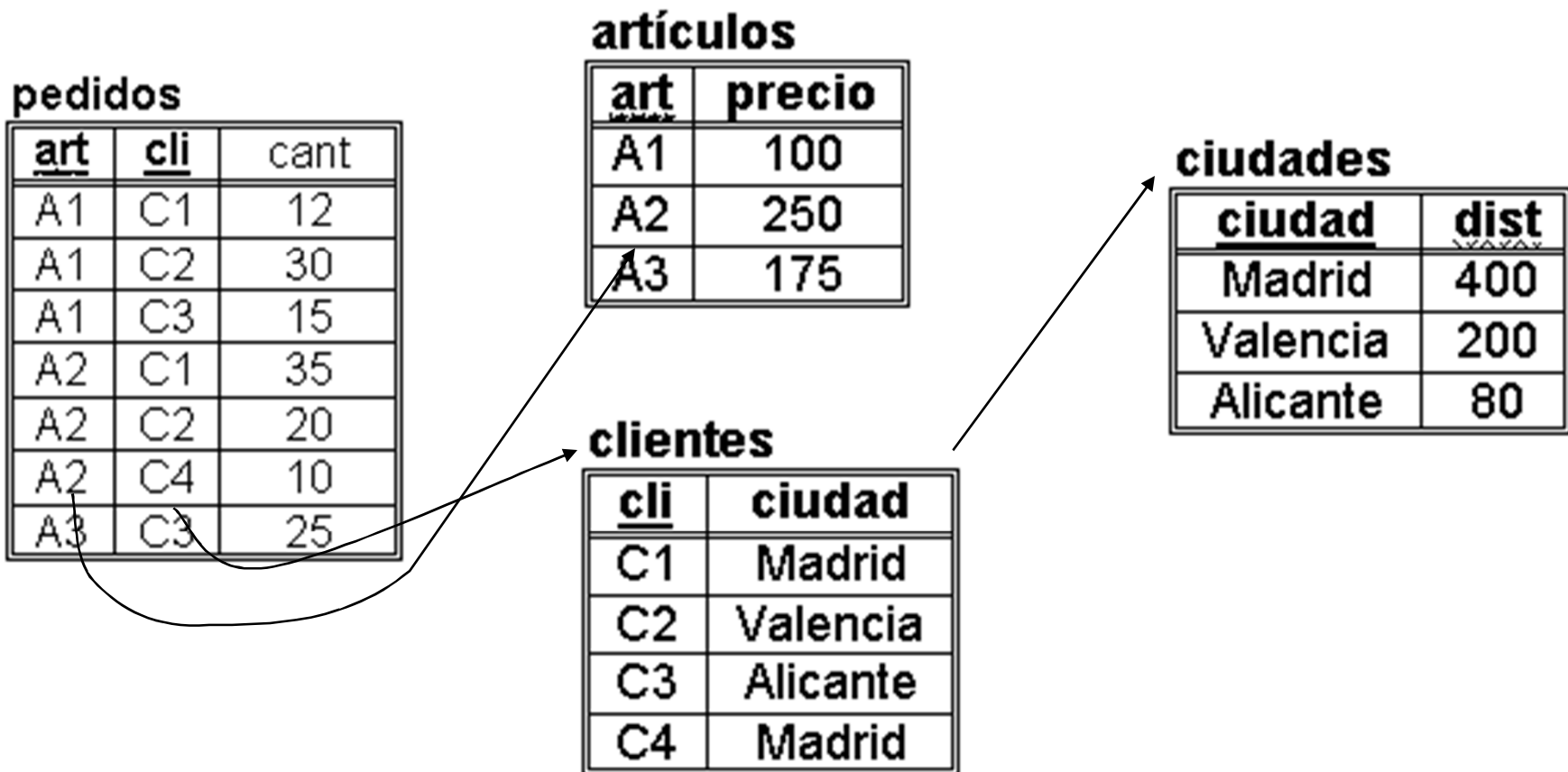
Ejemplo de diseño inadecuado

- **Mal diseño:**

- anomalías, redundancia e inconsistencias de la información
- imposibilidad para representar cierta información
- pérdida de información
- Otros, cuales?

Principio básico de diseño:

- *"hechos distintos se deben almacenar en objetos distintos"*



Teoría de Normalización



Técnica formal de análisis y organización de datos; trata de evitar la redundancia y anomalías de actualización.

- Introduce formalización en el diseño lógico de BDR.
- Además permite mecanizar parte del proceso al disponer de instrumentos algorítmicos de ayuda al diseño.
- Proceso de normalización: disminuye las anomalías de actualización, pero penaliza las consultas (combinación consume muchos recursos).

Formas normales

- un esquema de relación está en una determinada forma normal si **satisface un determinado conjunto específico de restricciones** definidas sobre los atributos del esquema (dependencias).
 - 1ª 2ª, 3ª FN (Codd, 1970)
 - Concepto de relación normalizada.
 - FNBC (Boyce/Codd, 1974)
 - Basadas en análisis de dependencias funcionales.
 - 4ª FN. Fagin, 1977
 - Basada en análisis de dependencias multivaluadas.
 - 5ª FN. Fagin, 1979
 - Basada en análisis de dependencias de proyección / combinación.

Formas normales



Dependencias

- **restricciones de integridad** impuestas por el **usuario**.
- propiedades inherentes al contenido **semántico** de los datos.
- se han de cumplir **para cualquier extensión** del esquema de **una** relación.
- No se pueden demostrar, pero sí **afirmar** por observación del minimundo a representar.
- Del análisis de la extensión de un esquema relacional sólo podremos deducir que no existe una determinada dependencia.
- Si una dependencia es cierta para **un determinado esquema de relación**, una extensión no será válida si no la cumple.

Dependencia funcional

- Sean A y B atributos de una misma tabla o relación R . Se dice que B es funcionalmente dependiente de A y se denota $A \rightarrow B$ si todo posible valor de A tiene asociado un único valor de B ,
- *o lo que es lo mismo*, en todas las tuplas de $r(R)$ en las que el atributo A toma el mismo valor “ v_1 ”, el atributo B toma también un mismo valor “ v_2 ”.

Dependencia funcional **definición**

- Sean X e Y subconjuntos de A (A , conjunto de atributos de R).

Decimos que X **determina o implica a Y en R**

(= Y **depende funcionalmente de X**)

si para dos tuplas cualesquiera de $r(R)$ que tengan iguales valores en X , también son iguales los valores en Y .

- Se representa por $X \rightarrow Y$

- Def: Sean $R(A:D)$, $X \subseteq A$, $Y \subseteq A$. $X \rightarrow Y$ si

$$\forall r(R), \forall t_i, t_j \in r(R), [\pi_X(t_i) = \pi_X(t_j) \Rightarrow \pi_Y(t_i) = \pi_Y(t_j)]$$

$$(\forall t_1, t_2 \in r (t_1[X] = t_2[X] \Rightarrow t_1[Y] = t_2[Y]))$$

Diagrama de dependencias funcionales

- Ejemplo: **R (A, DF).**

R: pedidos

A: {artículo, cliente, cantidad, precio, ciudad, distancia}.

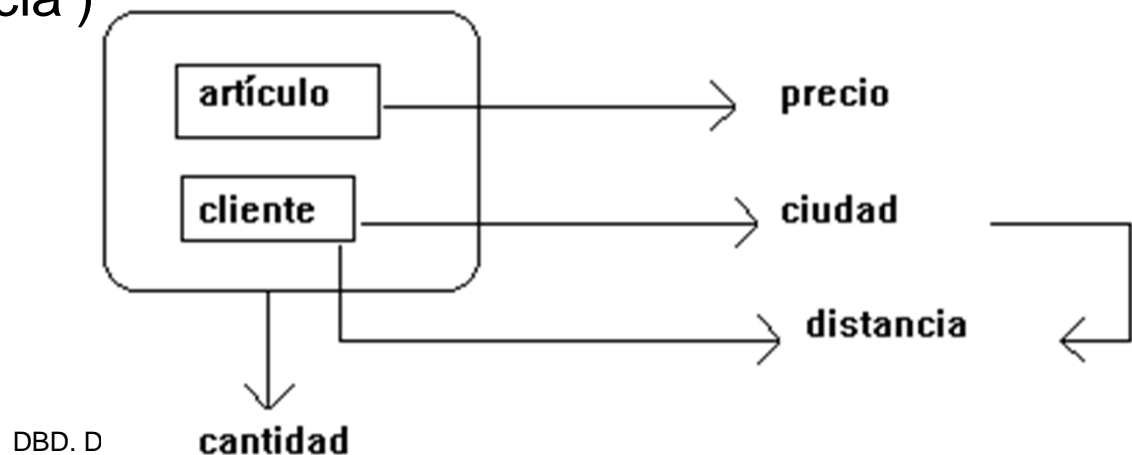
DF: ({artículo,cliente} → {cantidad,precio,ciudad, distancia},

artículo → precio,

cliente → {ciudad, distancia},

ciudad → distancia)

Diagrama de
Dependencias
Funcionales



Dependencia funcional plena o completa

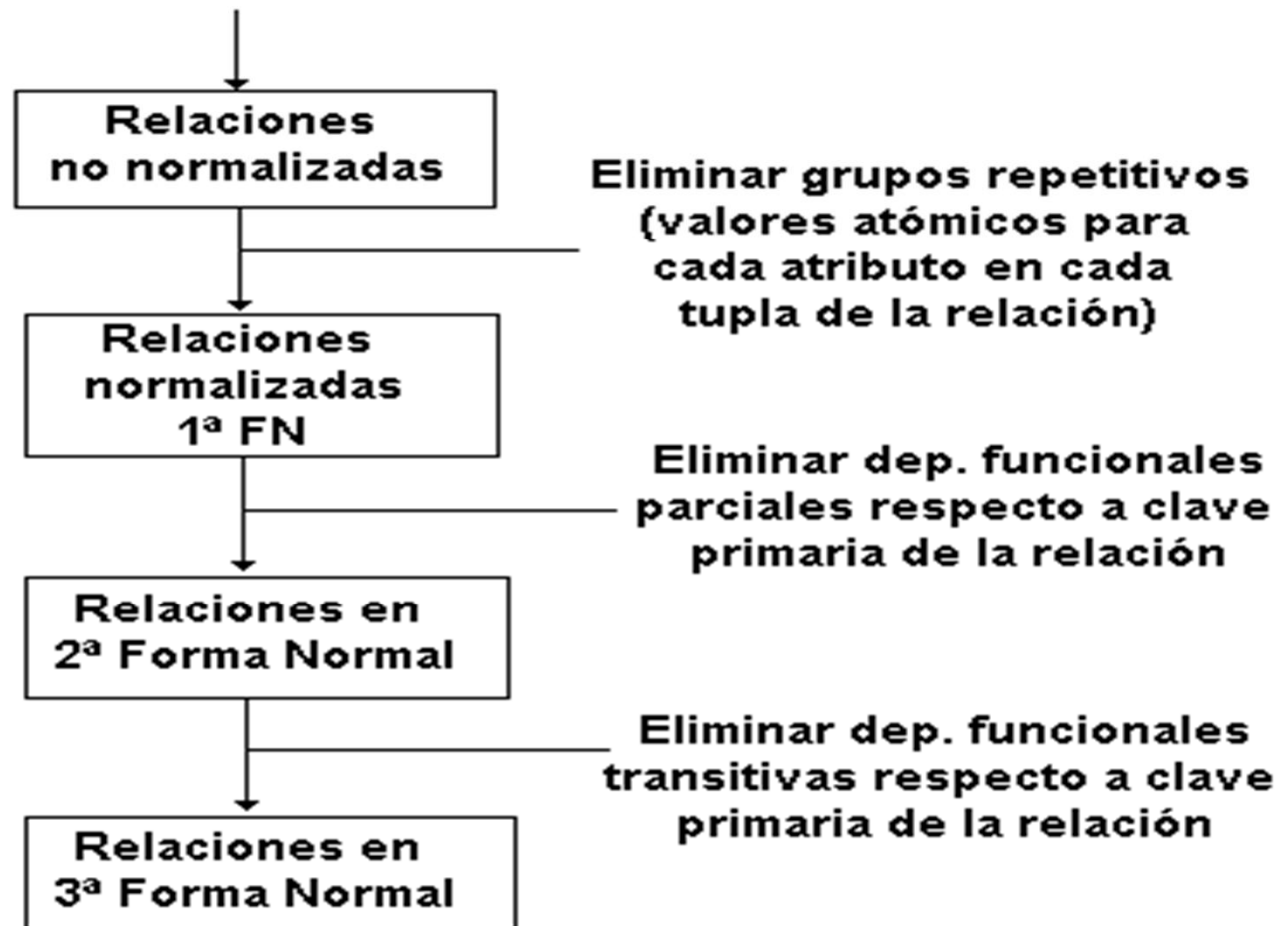
- Sea X (conjunto de atributos). Se dice que Y tiene dependencia funcional plena o completa de X ,
 - si depende funcionalmente de X
 - pero no depende de ningún subconjunto del mismo
- $X \Rightarrow Y$ sii $\neg \exists X' \subset X \mid X' \rightarrow Y$ (X determina completamente " \Rightarrow " a Y si y solo si no existe un subconjunto X' de X tal que ese subconjunto X' determine funcionalmente a Y)
- P.e. $(\text{artículo}, \text{cliente}) \Rightarrow \text{cantidad}$ es una DF completa, pero
- $(\text{artículo}, \text{cliente}) \rightarrow \text{precio}$ no es una DF completa puesto que $\text{artículo} \rightarrow \text{precio}$;

Dependencia funcional transitiva

- Si $X \rightarrow Y$, $Y \not\rightarrow X$, $Y \rightarrow Z$ entonces Z depende transitivamente de X ($X \twoheadrightarrow Z$).
- P.e.
 - Cliente \rightarrow ciudad,
 - ciudad $\not\rightarrow$ cliente (no determina funcionalmente), y
 - cliente \rightarrow distancia,
 - por tanto, ciudad \twoheadrightarrow distancia
 - (cliente determina “transitivamente” a distancia).

Normalización de un esquema de BD Relacional.

REQUERIMIENTOS
DE USUARIO



Para recordar el ejemplo

Pedidos

<u>Artículo</u>	<u>cliente</u>	cantidad	precio	ciudad	distancia
A1	C1	12	100	Madrid	400
A1	C2	30	100	Valencia	200
A1	C3	15	100	Alicante	80
A2	C1	35	250	Madrid	400
A2	C2	20	250	Valencia	200
A2	C4	10	250	Madrid	400
A3	C3	25	175	Alicante	80

Segunda Forma Normal (2ªFN).

- Un esquema de relación $R(A,DF)$ está en 2ªFN si y sólo si
 - está en 1ªFN (es decir, si la relación está normalizada)
 - y sus **atributos no primarios** dependen completamente de la clave primaria de R.
(atributos no primarios: que no formen parte de la clave primaria).
- Si una relación R no está en 2ª FN, se puede normalizar **descomponiendo** esa relación en:
 - Una relación con los atributos de clave primaria, más los atributos con dependencia completa de ella.
 - Una relación para cada "parte" de la clave primaria, más los atributos que dependan funcionalmente de esa parte.

Descomposición a 2ª FN (informal)

- Ejemplo: PEDIDOS se descompone en:

PEDIDOS'({**artículo**,**cliente**, cantidad},
{{artículo,cliente} → cantidad})

ARTICULOS ({**artículo**, precio}, {artículo → precio})

CLIENTES({**cliente**,ciudad,distancia},

Pedidos'

<u>articulos</u>	<u>clientes</u>	cantidad
A1	C1	12
A1	C2	30
A1	C3	15
A2	C1	35
A2	C2	20
A2	C4	10
A3	C3	25

artículos

<u>artículo</u>	precio
A1	100
A2	250
A3	175

clientes

<u>cliente</u>	ciudad	Distancia
C1	Madrid	400
C2	Valencia	200
C3	Alicante	80
C4	Madrid	400

Tercera Forma Normal (3^aFN)

- Un esquema de relación $R(A,DF)$ está en FN3 si y sólo si
 - está en FN2 y
 - ninguno de sus **atributos no primarios** depende transitivamente de la clave primaria de R.
- Es decir no hay DF transitivas.
- **Descomposición informal:** Si una relación no está en 3^a FN, y tenemos los conjuntos de atributos X, Y, Z tales que $X \rightarrow Y$, $Y \rightarrow Z$ (por tanto $X \dashrightarrow Z$):
 - Una relación formada por la clave primaria (X) más los atributos que dependen directamente de ella.
 - Una relación para los atributos de Y y Z.

Descomposición a 3ª FN (informal)

- Ejemplo: CLIENTES la descomponemos en:

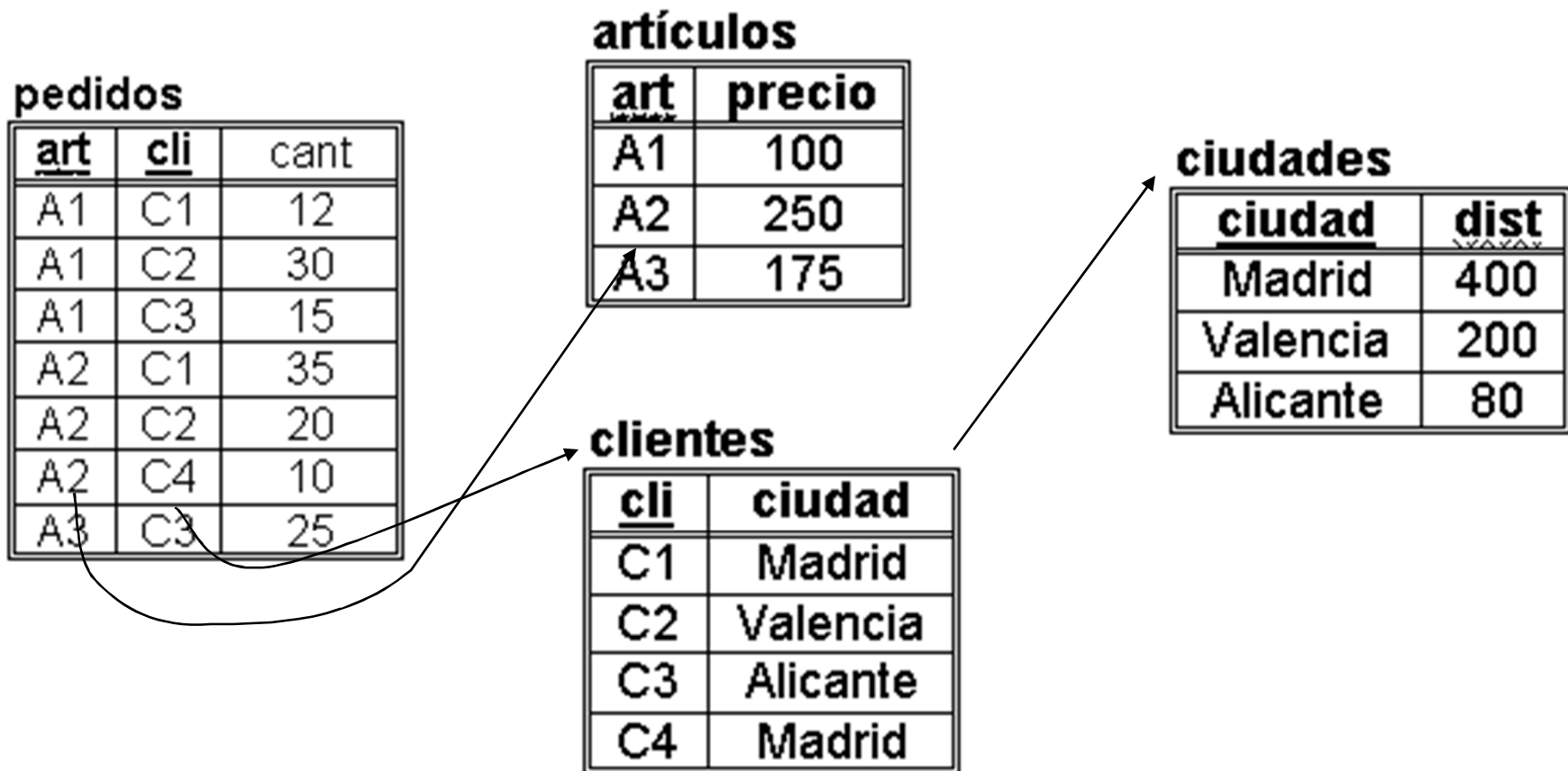
Ciudades

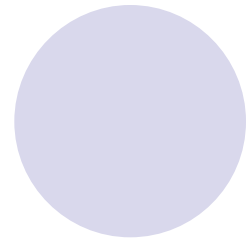
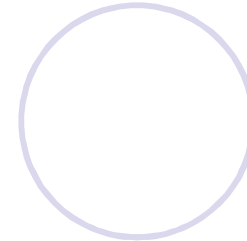
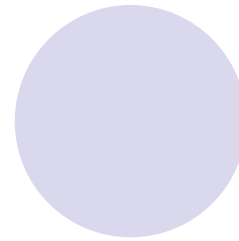
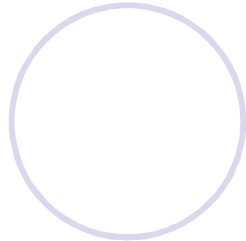
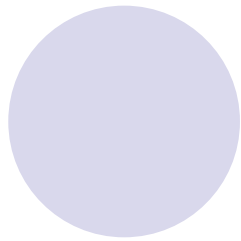
ciudad	distancia
Madrid	400
Valencia	200
Alicante	80

Cientes'

cliente	ciudad
C1	Madrid
C2	Valencia
C3	Alicante
C4	Madrid

Tabla en 3FN





- Tabla sin normalizar:

Nº alumno	Tutor	Despacho-Tut	Clase1	Clase2	Clase3
1022	García	412	101-07	143-01	159-02
4123	Díaz	216	201-01	211-02	214-01

Primera forma normal:

- eliminando el grupo repetido (Nº clase), según se muestra a continuación:

Nº alumno	Tutor	Despacho-Tut	Nº clase
1022	García	412	101-07
1022	García	412	143-01
1022	García	412	159-02
4123	Díaz	216	201-01
4123	Díaz	216	211-02
4123	Díaz	216	214-01

Segunda forma normal:

- eliminar los datos redundantes

Observe los diversos valores de

- N° clase para cada valor de N° alumno en la tabla anterior.
- N° clase no depende funcionalmente de N° alumno (la clave principal),
- de modo que la relación no cumple la segunda forma normal.
- Las dos tablas siguientes demuestran la segunda forma normal:

Alumnos:

N° alumno	Tutor	Despacho-Tut
1022	García	412
4123	Díaz	216

Registro:

N° alumno	N° clase
1022	101-07
1022	143-01
1022	159-02
4123	201-01
4123	211-02
4123	214-01

Tercera forma normal:

- eliminar los datos no dependientes de la **clave**, En el último ejemplo, Despacho-Tut (el número de despacho del tutor) es funcionalmente dependiente del atributo Tutor.
- La solución es pasar ese atributo de la tabla Alumnos a la tabla Personal, según se

muestra a continuación:

Alumnos:

Nº alumno	Tutor	Despacho-Tut
1022	García	412
4123	Díaz	216

Alumnos:

Nº alumno	Tutor
1022	García
4123	Díaz

Personal:

Nombre	Habitación	Dept
García	412	42
Díaz	216	42