



## UD 3. MODELO LÓGICO

### Normalización

GUÍA DE  
ESTUDIO

**ceedcv**  
CENTRE ESPECÍFIC  
D'EDUCACIÓ A DISTÀNCIA DE  
LA COMUNITAT VALENCIANA

## Bases de Datos (BD)

CFGS Desarrollo de Aplicaciones Web (DAW)

Abelardo Martínez

[a.martinezserrano@edu.gva.es](mailto:a.martinezserrano@edu.gva.es)

Pau Miñana

[p.minanacliment@edu.gva.es](mailto:p.minanacliment@edu.gva.es)

Basado y modificado de Sergio Badal [www.sergiobadal.com](http://www.sergiobadal.com)

**a distancia**



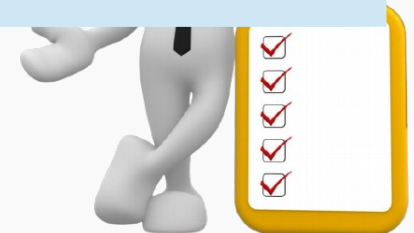
# UD 3. MODELO LÓGICO

- ¿Qué vimos la semana pasada?

- UD 3.1 MODELO LÓGICO RELACIONAL
  - Modelado lógico
  - Elementos
  - Restricciones
  - Transformaciones
- Boletín A: 9 ejercicios de complejidad básica (resueltos)
- Actividades propuestas (no evaluables)
  - Boletín B: 6 ejercicios de complejidad media

- ¿Qué veremos esta semana?

- Resolución de dudas de Boletines A y B
- UD 3.2 M.RELAC. NORMALIZACIÓN
  - Dependencias
  - Formas normales
- Actividades propuestas (no evaluables)
  - Boletín C: 3 ejercicios de complejidad alta



# UD 3. MODELO LÓGICO



## UD 3.2 NORMALIZACIÓN

### 3.2.1 ¿QUÉ ES LA NORMALIZACIÓN?

### 3.2.2 DEPENDENCIAS

### 3.2.3 PRIMERA FORMA NORMAL


### 3.2.4 SEGUNDA FORMA NORMAL

### 3.2.5 TERCERA FORMA NORMAL

### 3.2.6 MÁS FORMAS NORMALES

### 3.2.7 EJEMPLO COMPLETO

## 3.2.1 ¿QUÉ ES LA NORMALIZACIÓN?

 La **normalización** es un proceso de refinamiento para comprobar la calidad de nuestro modelo verificando que las relaciones o tablas del modelo Relacional obtenido no tienen redundancias ni inconsistencias.

- El proceso de normalización consiste en la aplicación de una serie de reglas que nos sirven para verificar, y en algunas ocasiones modificar, nuestro modelo relacional.
- Para ello, vamos a aplicar las siguientes fases de la normalización (existen más fases pero se salen de los objetivos de este curso):
  - Primer forma normal - **1FN**
  - Segunda forma normal - **2FN**
  - Tercera forma normal - **3FN**

## 3.2.1 ¿QUÉ ES LA NORMALIZACIÓN?

- El proceso de normalización se realizará de la siguiente forma, primero comprobaremos que todas nuestras relaciones o tablas están en 1FN y si no es así realizaremos los cambios necesarios para que así sea. Una vez lo tenemos todo en 1FN aplicaremos las reglas para ver si están en 2FN y así sucesivamente.
- Si decimos que nuestro modelo está en 3FN ya sabemos que cumple 1FN, 2FN y 3FN.
- Antes de comenzar a ver las reglas de la normalización debemos conocer el concepto de dependencia funcional y sus tipos.



# UD 3. MODELO LÓGICO



## UD 3.2 NORMALIZACIÓN

3.2.1 ¿QUÉ ES LA NORMALIZACIÓN?

### 3.2.2 DEPENDENCIAS

3.2.3 PRIMERA FORMA NORMAL

3.2.4 SEGUNDA FORMA NORMAL

3.2.5 TERCERA FORMA NORMAL

3.2.6 MÁS FORMAS NORMALES

3.2.7 EJEMPLO COMPLETO

# 3.2.2 DEPENDENCIAS

## Dependencia funcional



Un atributo B depende funcionalmente de otro atributo A si a **cada valor de A** le corresponde un **único valor de B**. Se dice que A implica B o lo que es lo mismo  $A \rightarrow B$ . En este caso A recibe el nombre de **implicante**.

Por ejemplo en la siguiente tabla *Personas* podemos tener:


**PERSONAS** (DNI, Nombre, Fecha\_Nacimiento);

Podemos decir que  $DNI \rightarrow Nombre$ , ya que un DNI se corresponde con un único Nombre, es decir, a través del DNI podemos localizar el nombre de la persona a la que pertenece.

Luego cuando tenemos un atributo A que implica B ( $A \rightarrow B$ ) es lo mismo que decir que B depende funcionalmente de A.

# 3.2.2 DEPENDENCIAS

## Dependencia funcional completa

 Dado un conjunto de atributos A formado por a\_1, a\_2, a\_3, ... (estos atributos formarán una clave primaria compuesta) existe una dependencia funcional completa cuando B depende de A pero no de un subconjunto de A.

Es decir que para obtener B son necesarios todos los elementos del conjunto A, o dicho de otra forma B depende de todos los atributos que forman la clave primaria A y no puede depender solamente de parte de ellos.

Por ejemplo cuando tenemos una tabla de *Compras* de la siguiente forma:

**COMPRAS** (Referencia\_Producto, Código\_Proveedor, Dirección\_Proveedor, Cantidad, Precio)


En este caso nuestra clave primaria está formada por dos campos, la *Referencia\_Producto* y el *Código\_Proveedor*. Pues bien, debemos comprobar si todos los demás atributos tienen una dependencia funcional de toda la clave primaria.

Podemos observar que la *Dirección\_Proveedor*, no dependerá de ambos, sino solamente del *Código\_Proveedor*, luego para ese atributo **no** existiría dependencia funcional completa. Sin embargo el resto de los campos, tanto la *Cantidad*, como el *Precio* acordado dependen del conjunto, es decir, dependen completamente de la clave



# 3.2.2 DEPENDENCIAS

## Dependencia transitiva

 Este tipo de dependencia implica a tres atributos. Cuando existe una dependencia  $A \rightarrow B$  y a su vez tenemos que  $B \rightarrow C$ , podemos decir que existe una dependencia funcional transitiva entre A y C.

Por ejemplo, si tenemos una tabla de *Productos* como la siguiente:

**PRODUCTOS** (Referencia\_Producto, Nombre, Precio, Stock, Fabricante, País);

Aquí podemos encontrar que el *Fabricante* depende de la *Referencia\_Producto*, es decir:

$\text{Referencia\_Producto} \rightarrow \text{Fabricante}$

Y por otro lado el *País* también puede depender del *Fabricante*, luego:

$\text{Fabricante} \rightarrow \text{País}$ .

Con estas dos dependencias podemos afirmar que el *País* **depende transitivamente** de la *Referencia\_Producto*.

# 3.2.2 DEPENDENCIAS

## Diagrama de dependencias

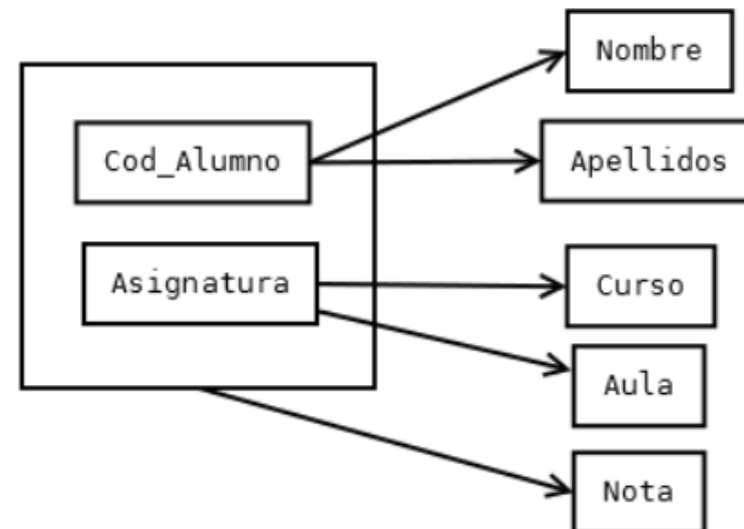
Los diagramas de dependencias muestran todos los atributos de una relación y sus dependencias. Habitualmente **se encierran en una caja el conjunto de atributos de la clave primaria y se unen mediante flechas** los atributos que muestren alguna dependencia entre ellos y también con el conjunto de claves.

Supongamos que tenemos una relación Alumno en la que representamos los datos de los alumnos y las notas de cada una de las asignaturas en que está matriculado. La CP es el código del alumno y la asignatura:

**Alumno** (cod\_alumno, nombre, apellidos, asignatura, nota, curso, aula)

Es obvio que no todos los atributos dependen completamente de la CP. Veamos las dependencias funcionales de cada uno de los atributos con respecto a los atributos de la clave:

- $\text{cod\_alumno} \rightarrow \text{nombre}$
- $\text{cod\_alumno} \rightarrow \text{apellidos}$
- $\text{asignatura} \rightarrow \text{curso}$
- $\text{asignatura} \rightarrow \text{aula}$
- $\{\text{cod\_alumno}, \text{asignatura}\} \rightarrow \text{nota}$



# UD 3. MODELO LÓGICO



## UD 3.2 NORMALIZACIÓN

3.2.1 ¿QUÉ ES LA NORMALIZACIÓN?

3.2.2 DEPENDENCIAS

**3.2.3 PRIMERA FORMA NORMAL**

3.2.4 SEGUNDA FORMA NORMAL

3.2.5 TERCERA FORMA NORMAL

3.2.6 MÁS FORMAS NORMALES

3.2.7 EJEMPLO COMPLETO

## 3.2.3 PRIMERA FORMA NORMAL

### 1FN

Una tabla está en 1FN si y solo si los valores que componen cada atributo de una tupla son **atómicos** (ni compuestos, ni multivaluados) **y no derivados**.

Es decir, cada atributo de la relación toma un único valor del dominio correspondiente y no hay valores definidos en función de otros atributos.

Los atributos **derivados** se sustituyen por los atributos de los que dependen si ES necesario.

- Por ejemplo, dado que el atributo *edad* suele ser derivado de la fecha de nacimiento, eliminamos dicho atributo siempre que la fecha de nacimiento esté almacenada, ya que podemos recuperar su valor.

Los atributos **compuestos** se dividen en tantos campos como tenga la composición.

- Por ejemplo, el típico atributo *nombre completo* se sustituye por nombre y apellidos.

Los atributos **multivaluados** se sustituyen por una relación en ER y se convierten a tablas como proceda.

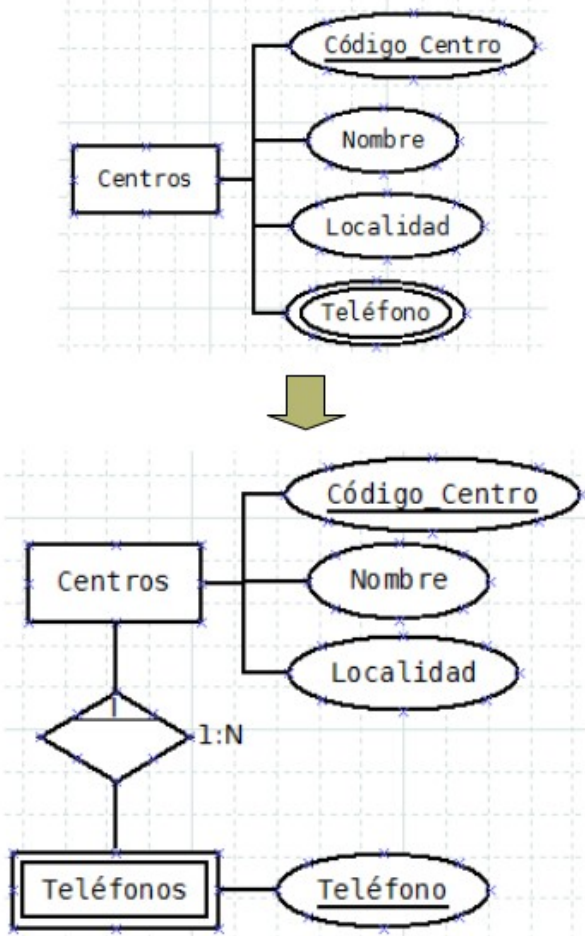
- Por ejemplo, el típico atributo *teléfonos* se sustituye por la relación que veremos a continuación.



## 3.2.3 PRIMERA FORMA NORMAL

# 1FN

| Código_Centro | Nombre              | Localidad            | Teléfono              |
|---------------|---------------------|----------------------|-----------------------|
| 45005467      | IES Ribera del Tajo | Talavera de la Reina | 925722233 - 925722804 |
| 45005239      | IES Azarquiel       | Toledo               | 925267843 – 925637843 |
| 36003748      | IES El Plantío      | Huelva               | 973847284             |



Nueva tabla **Centros**.

| Código_Centro | Nombre              | Localidad            |
|---------------|---------------------|----------------------|
| 45005467      | IES Ribera del Tajo | Talavera de la Reina |
| 45005239      | IES Azarquiel       | Toledo               |
| 36003748      | IES El Plantío      | Huelva               |

Y una nueva tabla que llamaremos **Teléfonos**.

| Código_Centro | Teléfono  |
|---------------|-----------|
| 45005467      | 925722233 |
| 45005467      | 925722804 |
| 45005239      | 925267843 |
| 45005239      | 925637843 |
| 36003748      | 973847284 |



# UD 3. MODELO LÓGICO



## UD 3.2 NORMALIZACIÓN

3.2.1 ¿QUÉ ES LA NORMALIZACIÓN?

3.2.2 DEPENDENCIAS

3.2.3 PRIMERA FORMA NORMAL

**3.2.4 SEGUNDA FORMA NORMAL**

3.2.5 TERCERA FORMA NORMAL

3.2.6 MÁS FORMAS NORMALES

3.2.7 EJEMPLO COMPLETO

## 3.2.4 SEGUNDA FORMA NORMAL

### 2FN

Una tabla está 2FN si y solo si está en 1FN y **cada atributo no clave primaria, tiene una dependencia funcional completa** con la clave primaria.

La 2FN se aplica a las relaciones que tienen claves primarias compuestas por dos o más atributos.

Si una relación está en 1FN y su clave primaria es simple, entonces también está en 2FN.

Para pasar una relación que está en 1FN a 2FN hay que **eliminar las dependencias parciales de la clave primaria**.

Para ello, se eliminan los atributos que son parcialmente dependientes y se ponen en una nueva relación y como clave primaria la parte de la que dependen de la clave primaria.

De esta manera tendremos dos tablas, una con los atributos que dependen de manera completa de la clave y otra tabla con los atributos que solo dependen de una parte de la clave.

## 3.2.4 SEGUNDA FORMA NORMAL

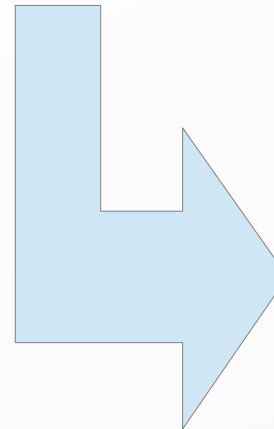
### 2FN

Una tabla está 2FN si y solo si está en 1FN y **cada atributo no clave primaria, tiene una dependencia funcional completa** con la clave primaria.

La 2FN se aplica a las relaciones que tienen claves primarias compuestas por dos o más atributos.

Si una relación está en 1FN y su clave primaria es simple, entonces también está en 2FN.

| <u>Empleado</u> | <u>Especialidad</u> | Empresa   |
|-----------------|---------------------|-----------|
| Juan Velasco    | Bases de Datos      | IBM       |
| Juan Velasco    | Sistemas Linux      | IBM       |
| Ana Comarce     | Bases de Datos      | Oracle    |
| Javier Gil      | Sistemas Windows    | Microsoft |
| Javier Gil      | Desarrollo .Net     | Microsoft |



| <u>Empleado</u> | Empresa   |
|-----------------|-----------|
| Juan Velasco    | IBM       |
| Ana Comarce     | Oracle    |
| Javier Gil      | Microsoft |

| <u>Empleado</u> | <u>Especialidad</u> |
|-----------------|---------------------|
| Juan Velasco    | Bases de Datos      |
| Juan Velasco    | Sistemas Linux      |
| Ana Comarce     | Bases de Datos      |
| Javier Gil      | Sistemas Windows    |
| Javier Gil      | Desarrollo .Net     |

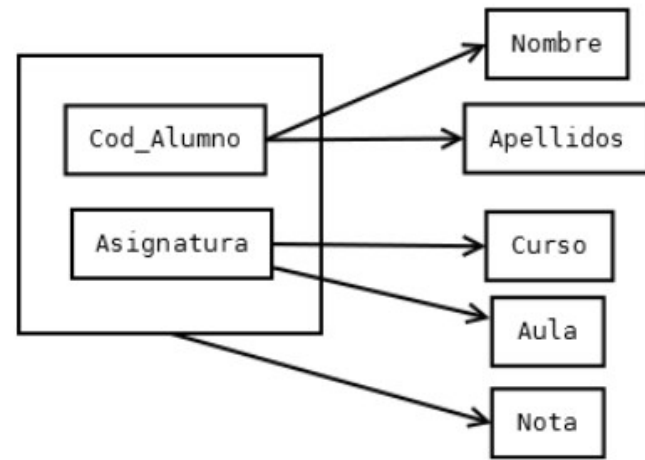
Podemos ver que como clave principal se ha elegido el conjunto Empleado + Especialidad.

Sin embargo podemos observar que el atributo Empresa no depende de toda la clave sino solo de parte de ella, en este caso de Empleado.



# 3.2.4 SEGUNDA FORMA NORMAL

## 2FN



**Alumno** (cod\_alumno, nombre, apellidos, asignatura, nota, curso, aula)

| <u>cod alumno</u> | nombre    | apellidos | <u>asignatura</u> | nota | curso | aula |
|-------------------|-----------|-----------|-------------------|------|-------|------|
| 1111              | Pepe      | García    | Lengua I          | 5    | 1     | 15   |
| 1111              | Pepe      | García    | Inglés II         | 5    | 2     | 16   |
| 2222              | María     | Suárez    | Inglés II         | 7    | 2     | 16   |
| 2222              | María     | Suárez    | Ciencias II       | 7    | 2     | 14   |
| 3333              | Juan      | Gil       | Plástica I        | 6    | 1     | 18   |
| 3333              | Juan      | Gil       | Matemáticas I     | 6    | 1     | 12   |
| 4444              | Francisco | Montoya   | Lengua II         | 4    | 2     | 11   |
| 4444              | Francisco | Montoya   | Matemáticas I     | 6    | 1     | 12   |
| 4444              | Francisco | Montoya   | Ciencias I        | 8    | 1     | 14   |

**Alumnos**(cod\_alumno, nombre, apellidos)

**Asignaturas**(asignatura, curso, aula)

**Notas**(cod\_alumno, asignatura, nota)

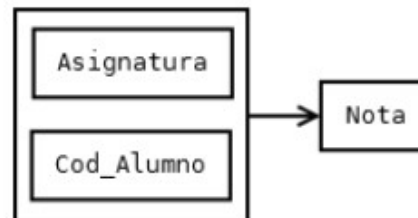
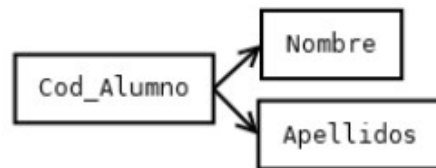
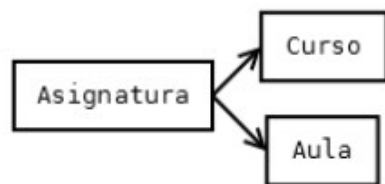
CAj: {cod\_alumno} → Alumnos

CAj: {asignatura} → Asignaturas

| <u>cod alumno</u> | <u>asignatura</u> | nota |
|-------------------|-------------------|------|
| 1111              | Lengua I          | 5    |
| 1111              | Inglés II         | 5    |
| 2222              | Ciencias II       | 7    |
| 2222              | Inglés II         | 7    |
| 3333              | Plástica I        | 6    |
| 3333              | Matemáticas I     | 6    |
| 4444              | Lengua II         | 4    |
| 4444              | Matemáticas I     | 6    |
| 4444              | Ciencias I        | 8    |

| <u>asignatura</u> | curso | aula |
|-------------------|-------|------|
| Lengua I          | 1     | 15   |
| Inglés II         | 2     | 16   |
| Ciencias II       | 2     | 14   |
| Plástica I        | 1     | 18   |
| Matemáticas I     | 1     | 12   |
| Lengua II         | 2     | 11   |

| <u>cod alumno</u> | nombre    | apellidos |
|-------------------|-----------|-----------|
| 1111              | Pepe      | García    |
| 2222              | María     | Suárez    |
| 3333              | Juan      | Gil       |
| 4444              | Francisco | Montoya   |



# UD 3. MODELO LÓGICO



## UD 3.2 NORMALIZACIÓN

3.2.1 ¿QUÉ ES LA NORMALIZACIÓN?

3.2.2 DEPENDENCIAS

3.2.3 PRIMERA FORMA NORMAL

3.2.4 SEGUNDA FORMA NORMAL

**3.2.5 TERCERA FORMA NORMAL**

3.2.6 MÁS FORMAS NORMALES

3.2.7 EJEMPLO COMPLETO



## 3.2.5 TERCERA FORMA NORMAL

### 3FN

Una tabla está en 3FN si está en 2FN y los atributos no-clave no dependen de otros atributos no-clave, es **decir no existen dependencias transitivas de la clave primaria**.

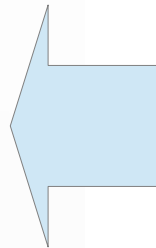
«**Todo atributo no clave debe proporcionar información sobre la clave, sobre toda la clave y nada más que la clave... con la ayuda de Codd**».

Para eliminar las dependencias transitivas se suele crear una nueva tabla con el campo del que dependen como clave principal.

# 3.2.5 TERCERA FORMA NORMAL

## 3FN

| <u>Torneo</u> | <u>Año</u> | Ganador        |
|---------------|------------|----------------|
| Alcores       | 2008       | Javier Gil     |
| Estoril       | 2008       | Alberto Sanz   |
| Getafe        | 2008       | Jacinto Martín |
| Santa María   | 2008       | Roberto Loaisa |
| Alcores       | 2009       | Jacinto Martín |
| Estoril       | 2009       | Ignacio Gómez  |
| Lisboa        | 2009       | Roberto Loaisa |
| Getafe        | 2009       | Roberto Loaisa |
| Santa María   | 2009       | Javier Gil     |



| <u>Torneo</u> | <u>Año</u> | Ganador        | Fecha_Nacimiento_Ganador |
|---------------|------------|----------------|--------------------------|
| Alcores       | 2008       | Javier Gil     | 20 de Abril de 1990      |
| Estoril       | 2008       | Alberto Sanz   | 15 de Julio de 1991      |
| Getafe        | 2008       | Jacinto Martín | 10 de Enero de 1989      |
| Santa María   | 2008       | Roberto Loaisa | 25 de Febrero de 1989    |
| Alcores       | 2009       | Jacinto Martín | 10 de Enero de 1989      |
| Estoril       | 2009       | Ignacio Gómez  | 20 de Septiembre de 1990 |
| Lisboa        | 2009       | Roberto Loaisa | 25 de Febrero de 1989    |
| Getafe        | 2009       | Roberto Loaisa | 25 de Febrero de 1989    |
| Santa María   | 2009       | Javier Gil     | 20 de Abril de 1990      |

Podemos observar que el atributo *Ganador* depende del nombre del *Torneo* y del *Año* del mismo (ambas forman la clave primaria), sin embargo la fecha de nacimiento del ganador no depende de la clave principal, sino que depende del ganador. Aquí podemos ver una dependencia transitiva de la clave principal:

Torneo + Año → Ganador

Ganador → Fecha\_Nacimiento\_Ganador

Luego, *Fecha\_Nacimiento\_Ganador* depende transitivamente de la clave primaria.

| <u>Ganador</u> | Fecha_Nacimiento_Ganador |
|----------------|--------------------------|
| Javier Gil     | 20 de Abril de 1990      |
| Alberto Sanz   | 15 de Julio de 1991      |
| Jacinto Martín | 10 de Enero de 1989      |
| Roberto Loaisa | 25 de Febrero de 1989    |
| Ignacio Gómez  | 20 de Septiembre de 1990 |

# UD 3. MODELO LÓGICO



## UD 3.2 NORMALIZACIÓN

3.2.1 ¿QUÉ ES LA NORMALIZACIÓN?

3.2.2 DEPENDENCIAS

3.2.3 PRIMERA FORMA NORMAL

3.2.4 SEGUNDA FORMA NORMAL

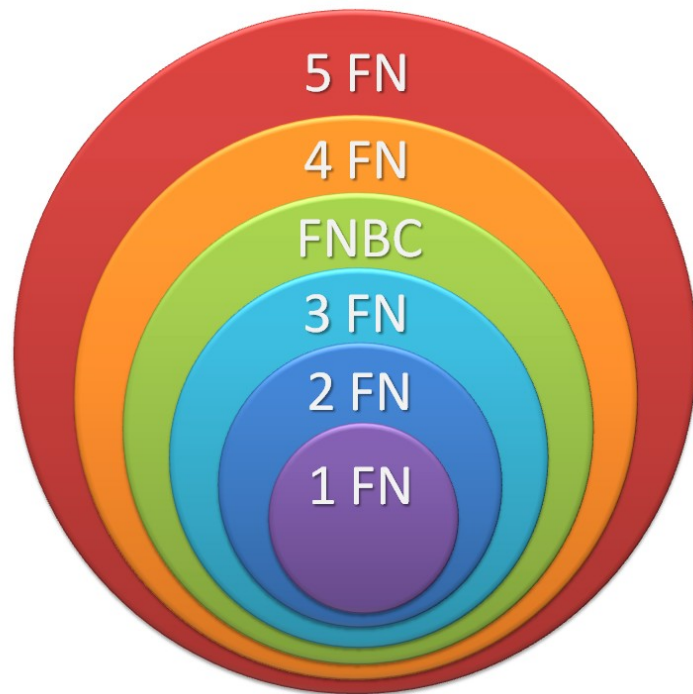
3.2.5 TERCERA FORMA NORMAL

**3.2.6 MÁS FORMAS NORMALES**

3.2.7 EJEMPLO COMPLETO



## 3.2.6 MÁS FORMAS NORMALES



Además de estas formas normales, existen otras más cuyo estudio es muchas veces más teórico que práctico.

Se pueden demostrar matemáticamente pero que prácticamente no se suelen aplicar al crear un diseño real, por ello no las vamos a tratar en este curso.



<https://picodotdev.github.io/blog-bitix/2018/02/las-6-plus-2-formas-normales-de-las-bases-de-datos-relacionales/>

# UD 3. MODELO LÓGICO



## UD 3.2 NORMALIZACIÓN

3.2.1 ¿QUÉ ES LA NORMALIZACIÓN?

3.2.2 DEPENDENCIAS

3.2.3 PRIMERA FORMA NORMAL

3.2.4 SEGUNDA FORMA NORMAL

3.2.5 TERCERA FORMA NORMAL

3.2.6 MÁS FORMAS NORMALES

**3.2.7 EJEMPLO COMPLETO**



# 3.2.7 EJEMPLO COMPLETO



## Paso a 3FN

**Cliente** (numcli, nif, nombre, dir, nom\_ciudad, nom\_prov, telf, {cuenta}<sup>n</sup>)

( {cuenta}<sup>n</sup> usaremos esta notación para expresar atributos compuestos {} y multivaluados <sup>n</sup>)

**Cuenta** (num\_cuenta, tipo\_c, saldo, {tarjeta}<sup>n</sup>)

El atributo *tarjeta* tambien es compuesto y multivaluado:  
tarjeta(num\_tarjeta, tipo\_t, comision, limite)

# 3.2.7 EJEMPLO COMPLETO

## 1FN?

**Cliente** (numcli, nif, nombre, dir, nom\_ciudad, nom\_prov, telf, {cuenta}<sup>n</sup>)

( {cuenta}<sup>n</sup> usaremos esta notación para expresar atributos compuestos {} y multivaluados <sup>n</sup>)

**Cuenta** (num\_cuenta, tipo\_c, saldo, {tarjeta}<sup>n</sup>)

El atributo *tarjeta* también es compuesto y multivaluado:

*tarjeta*(num\_tarjeta, tipo\_t, comision, limite)

### Paso a 1FN

Lo primero es eliminar los atributos multivaluados y los atributos compuestos. La relación **Cliente** sólo tiene un atributo multivaluado llamado cuenta. Para ello quitamos el atributo de la tabla principal, buscamos la clave primaria y añadimos las restricciones:

**Cliente**(numcli, nif, nombre, dir, nom\_ciudad, nom\_prov, telf)

CP: {numcli}

UNIQUE: {nif}

**Cuenta**(numcli, num\_cuenta, tipo\_c, saldo)

CP: {num\_cuenta}

CAj: {numcli} → Cliente

**Tarjeta**(num\_cuenta, num\_tarjeta, tipo\_t, comisión, límite)

CP: {num\_tarjeta}

CAj: {num\_cuenta} → Cuenta

# 3.2.7 EJEMPLO COMPLETO

## Paso a 2FN

La 2FN obliga a que todos los atributos que no forman parte de la clave, dependan **completamente** de esta. Como ninguna de las relaciones obtenidas en el paso anterior tiene una clave compuesta, todas ellas se encuentran ya en 2FN.

# 2FN?

## Paso a 1FN

Lo primero es eliminar los atributos multivaluados y los atributos compuestos. La relación **Ciente** sólo tiene un atributo multivaluado llamado cuenta. Para ello quitamos el atributo de la tabla principal, buscamos la clave primaria y añadimos las restricciones:

**Ciente**(numcli, nif, nombre, dir, nom\_ciudad, nom\_prov, telf)

CP: {numcli}

UNIQUE: {nif}

**Cuenta**(numcli, num\_cuenta, tipo\_c, saldo)

CP: {num\_cuenta}

CAj: {numcli} → Ciente

**Tarjeta**(num\_cuenta, num\_tarjeta, tipo\_t, comisión, límite)

CP: {num\_tarjeta}

CAj: {num\_cuenta} → Cuenta

# 3.2.7 EJEMPLO COMPLETO

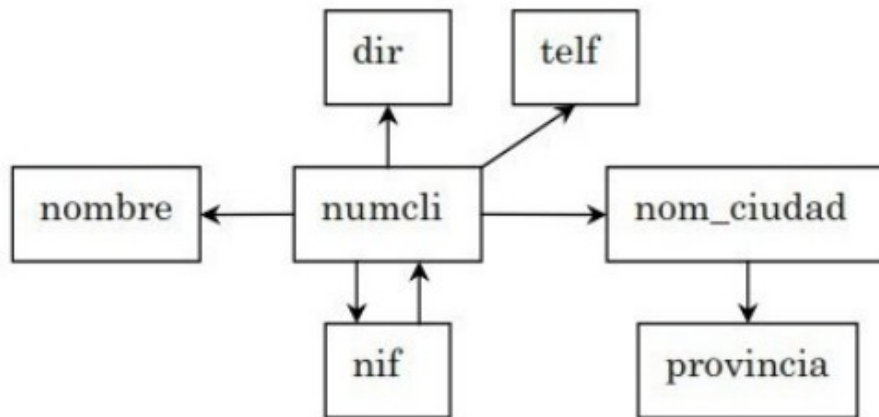
## Paso a 3FN

Ahora queda eliminar las dependencias transitivas y entre atributos. Veamos primero las dependencias de cada una de las tablas:

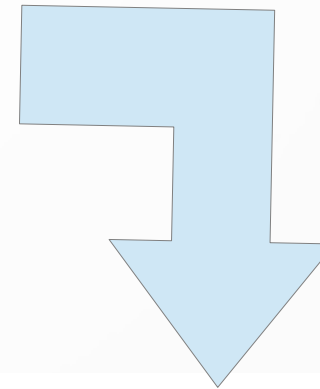
Cliente(*numcli*: d\_numcli, *nif*: d\_nif, *nombre*: d\_nombre, *dir*: dom\_dir, *ciudad*: d\_nom\_ciudad, *provincia*: d\_nom\_prov, *telf*: d\_telf)

CP: {numcli}

Único: {nif}



# 3FN?



**Ciudad**(nom\_ciudad, nom\_prov)

CP: {nom\_ciudad}

**Cliente**(numcli, nif, nombre, dir, nom\_ciudad, telf)

CP: {numcli}

CAlt: {nif}

CAj: {nom\_ciudad} → Ciudad

UNIQUE: {nif}



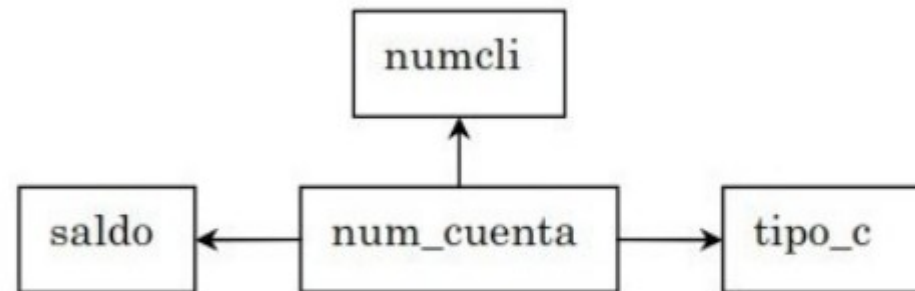
# 3.2.7 EJEMPLO COMPLETO

## 3FN?

Cuenta(*numcli*: d\_num\_cli, *num\_cuenta*: d\_num\_cuenta, *tipo\_c*: d\_tipo\_c, *saldo*: d\_saldo)

CP: {num\_cuenta}

CA: {numcli} → Cliente



Esta relación no tiene relaciones transitivas entre atributos, ya se encuentra en 3FN.



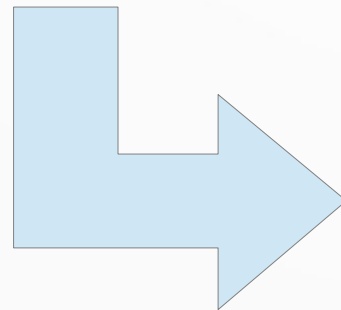
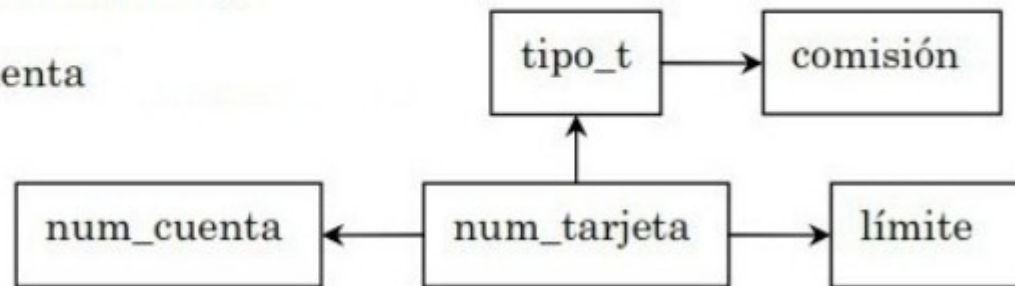
# 3.2.7 EJEMPLO COMPLETO

## 3FN?

Tarjeta(*num\_cuenta*: d\_num\_cuenta, *num\_tarjeta*: d\_num\_tarjeta, *tipo\_t*: d\_tipo\_t, *comisión*: d\_comisión, *límite*: d\_límite)

CP: {num\_tarjeta}

CA: {num\_cuenta} → Cuenta



**Comisión**(comisión, tipo\_t)

CP: {tipo\_t}

**Tarjeta**(num\_cuenta, num\_tarjeta, tipo\_t, límite)

CP: {num\_tarjeta}

CAj: {num\_cuenta} → Cuenta

CAj: {tipo\_t} → Comisión

# 3.2.7 EJEMPLO COMPLETO

3FN!

**Ciudad**(nom\_ciudad, nom\_prov)

CP: {nom\_ciudad}

**Cliente**(numcli, nif, nombre, dir, nom\_ciudad, telf)

CP: {numcli}

CAlt: {nif}

CAj: {nom\_ciudad} → Ciudad

UNIQUE: {nif}

**Cuenta**(numcli, num\_cuenta, tipo\_c, saldo)

CP: {num\_cuenta}

CAj: {numcli} → Cliente

**Comisión**(comisión, tipo\_t)

CP: {tipo\_t}

**Tarjeta**(num\_cuenta, num\_tarjeta, tipo\_t, límite)

CP: {num\_tarjeta}

CAj: {num\_cuenta} → Cuenta

CAj: {tipo\_t} → Comisión

# ACTIVIDADES PARA LA SEMANA QUE VIENE



Consulta los ejercicios sugeridos que encontrarás en el Aula Virtual adjuntos en un único PDF llamado:

- **BD.U3S2.Boletín actividades C**

Comprender estas actividades no evaluables es esencial para resolver la tarea evaluable que tenemos por delante.

**Si buscas las soluciones navegando por Internet te estarás engañando a ti mismo.**

Intenta solucionar el problema utilizando los recursos que te facilitamos y la documentación ampliada que encontrarás en el Aula Virtual, **acudiendo al foro de la unidad** si tienes cualquier duda que no sepas resolver.

La próxima semana encontrarás las soluciones propuestas junto a ese PDF.