

# UNIDAD 3

## MODELO LÓGICO

**BASES DE DATOS 22/23**  
CFGS DAW

### BOLETÍN 3.C. MODELO (LÓGICO) RELACIONAL. NORMALIZACIÓN CON SOLUCIONES.

**Revisado por:**

Sergio Badal, Abelardo Martínez y Pau Miñana

**Autores:**

Raquel Torres

Paco Aldarias

Fecha: 13/11/22

Licencia Creative Commons



**Reconocimiento - NoComercial - CompartirIgual (by-nc-sa):** No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

## ÍNDICE DE CONTENIDO

<b>1. EJERCICIO 1 LOCALIDAD.....</b>	<b>3</b>
<b>2. EJERCICIO 2. PROVEEDOR.....</b>	<b>5</b>
<b>3. EJERCICIO 3. VINO.....</b>	<b>8</b>

## 1. EJERCICIO 1 LOCALIDAD

Se desea almacenar información de Provincias, Comarcas y Localidades de Valencia. Partimos del siguiente modelo lógico (relacional) y se pretende normalizar hasta 3FN.

Solo interesa guardar el nombre y el código tanto de la provincia, comarca y localidad.

Restricciones:

- Una localidad no puede estar en más de una comarca.
- Una comarca no puede estar en más de una provincia.

### Modelo lógico:

**Provincia**(cod\_prov, nom\_prov)

CP: {cod\_prov}

**Comarca**(cod\_com, nom\_com, cod\_prov)

CP: {cod\_com}

VNN: {cod\_prov}

CAj: {cod\_prov} → Provincia

**Localidad**(cod\_loc, nom\_loc, cod\_com)

CP: {cod\_loc}

VNN: {cod\_com}

CAj: {cod\_com} → Comarca

**Solución:****1FN**

Como ninguna de las relaciones anteriores tiene atributos multivaluados o compuestos, todas ellas están en 1FN.

**2FN**

Hay que eliminar las DF parciales. Como ninguna de las relaciones anteriores tiene claves compuestas, todas ellas están en 2FN.

**3FN**

Hay que eliminar las DF transitivas. Para ello realizamos un diagrama de DF para cada relación y estudiamos tabla por tabla:

**TABLA Provincia**

Provincia(cod\_prov, nom\_prov)

cod\_prov → nombre

**No hay DF transitivas, la tabla es 3FN.**

**TABLA Comarca**

Comarca(cod\_com, nom\_com, cod\_prov)

cod\_com → cod\_prov, nombre

**No hay DF transitivas, la tabla es 3FN.**

**TABLA Localidad**

Localidad(cod\_loc, nom\_loc, cod\_com)

cod\_loc → cod\_com, nombre

**No hay DF transitivas, la tabla es 3FN.**

Rellenando los datos se puede ver cómo quedaría este modelo en 3FN:

PROVINCIA		
Cod_pro	Nombre	
Castellon	12	
Valencia	46	
Alicante	3	

COMARCA		
cod_comarca	nom_comarca	cod_pro
24	Vall d'Albaida	46
27	L,Alcoia	3
23	La costera	46
8	Alt Palancia	12

LOCALIDAD		
cod_localidad	nom_localidad	cod_comarca
46839	Guadassequies	24
46830	Beniganim	24
3820	Cocentaina	27
12400	Segorve	8

## 2. EJERCICIO 2. PROVEEDOR

Normaliza a 3FN las siguientes relaciones:

**Artículo**(cod\_art, descrip, color, peso)

CP: {cod\_art}

**Proveedor**(cod\_prov, cod\_art, nombre, provincia, ciudad, cantidad)

CP: {cod\_prov, cod\_art}

CA: {cod\_art} → Artículo

Considera las siguientes dependencias funcionales (DF):

- cod\_prov → nombre, provincia, ciudad
- cod\_art → descripción, color, peso
- ciudad → provincia
- cod\_prov, cod\_art → cantidad

### Solución:

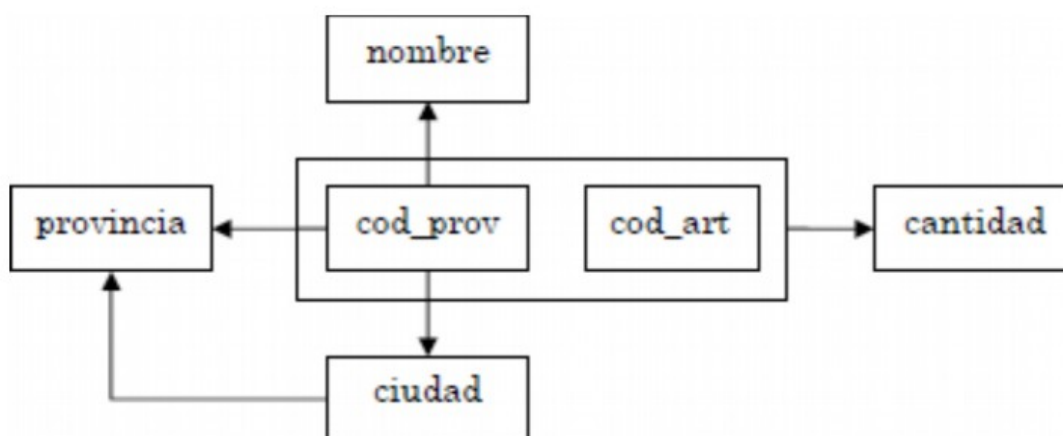
#### 1FN

Hay que eliminar atributos compuestos, multivaluados y derivados. Como ninguna de las dos relaciones anteriores tiene atributos de ese tipo, ya están en 1FN.

#### 2FN

Hay que eliminar las DF parciales. Como esto solo afecta a las relaciones cuya clave primaria sea compuesta, podemos afirmar que la relación Artículo ya se encuentra en 2FN.

Para ver si Proveedor está en 2FN primero tenemos que obtener el diagrama de dependencias:



Se puede observar que hay atributos que dependen parcialmente de la clave.

Para pasar esta relación a 2FN haremos lo siguiente:

- En Proveedor dejaremos solo la información que depende de `cod_prov`.
- Crearemos una nueva relación llamada Pedidos donde figure la cantidad de artículos pedidos a un determinado proveedor.

**Proveedor**(`cod_prov`, nombre, provincia, ciudad)

CP: {`cod_prov`}

**Pedido**(`cod_prov`, `cod_art`, cantidad)

CP: {`cod_prov`, `cod_art`}

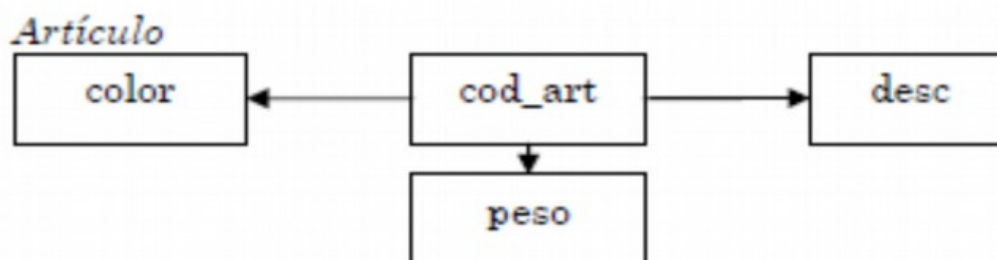
CAj: {`cod_art`} → Artículo

CAj: {`cod_prov`} → Proveedor

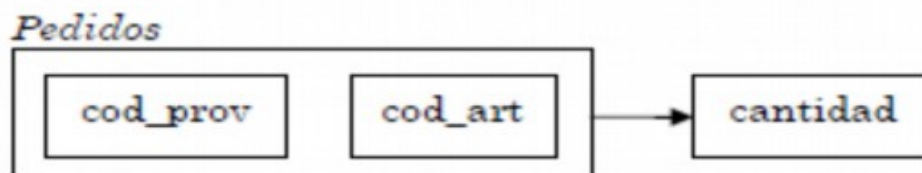
3FN

Hay que eliminar las DF transitivas. Para ello hacemos un diagrama para cada relación:

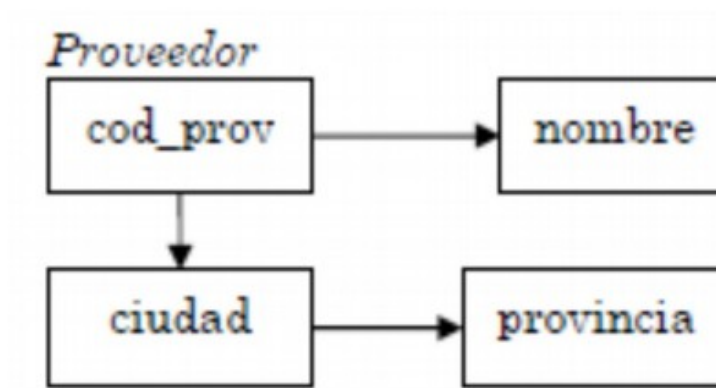
#### TABLA ARTÍCULO:



Como no hay ninguna dependencia transitiva, la relación **Artículo** se encuentra en 3FN.

**TABLA PEDIDO:**

Como no hay ninguna dependencia transitiva, la relación **Pedido** se encuentra en 3FN.

**TABLA PROVEEDOR:**

Como se puede observar, aquí tenemos una dependencia transitiva. Para eliminarla creamos una nueva tabla llamada **Ciudad** que asocie ciudades con provincia y eliminamos el atributo provincia de la tabla **Proveedor**.

**Ciudad**(ciudad, provincia)

CP: {ciudad}

**Proveedor**(cod\_prov, nombre, ciudad)

CP: {cod\_prov}

CAj: {ciudad} → Ciudad

Así pues el conjunto final de tablas 3FN es:

**Artículo**(cod\_art, descrip, color, peso)

CP: {cod\_art}

**Pedido**(cod\_prov, cod\_art, cantidad)

CP: {cod\_prov, cod\_art}

CAj: {cod\_art} → Artículo

CAj: {cod\_prov} → Proveedor

**Ciudad**(ciudad, provincia)

CP: {ciudad}

**Proveedor**(cod\_prov, nombre, ciudad)

CP: {cod\_prov}

CAj: {ciudad} → Ciudad

### 3. EJERCICIO 3. VINO

Considera el siguiente esquema lógico:

**Catador**(dni\_catador, nombre, experiencia, dni\_conyuge, edad, fecha\_nacimiento)

CP: {dni\_catador}

edad = Año\_Actual – fecha\_nacimiento

**Vino**(nom\_vino, añada, precio, bodega)

CP: {nom\_vino}

**Cata**(dni\_catador, nom\_vino, nota\_aromas, nota\_sabor, tipo\_vino)

CP: {dni\_catador, nom\_vino}

CAj: {dni\_catador} → Catador

CAj: {nom\_vino} → Vino

**Composición**(nom\_vino, tipo\_uva, porcentaje)

CP: {nom\_vino, tipo\_uva}

CAj: {nom\_vino} → Vino

El campo añada hace referencia al año que se vendimió la uva de ese vino, la bodega al almacén donde se elaboró, tipo\_vino indica si se trata de tinto, blanco o rosado, tipo\_uva puede ser garnacha, tempranillo, cariñena, etc. y porcentaje a la proporción en que esa uva está en ese vino (un vino puede estar formado por más de un tipo de uva). El resto de campos se sobreentienden.

Normaliza a 3FN las relaciones anteriores.



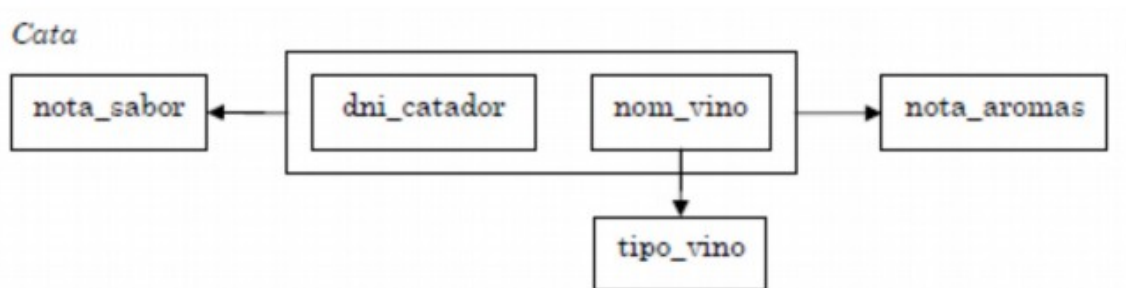
**Solución:****1FN**

Si quitamos el atributo edad y lo dejamos como una restricción (que el catador sea mayor de edad), ninguna relación tiene atributos multivaluados o compuestos, ni derivados y todas las relaciones están en 1FN.

**2FN**

Hay que eliminar las DF parciales. Como las relaciones Catador y Vino no tienen claves principales compuestas, ya están en 2FN.

Para las relaciones Cata y Composición obtenemos el diagrama de DF para ver si hay dependencias parciales:



Como el atributo tipo\_vino solo depende de una parte de la clave, lo quitamos de la tabla Cata y lo agregamos en una nueva tabla llamada Tipo\_vino.

**Cata**(dni\_catador, nom\_vino, nota\_aromas, nota\_sabor)

CP: {dni\_catador, nom\_vino}

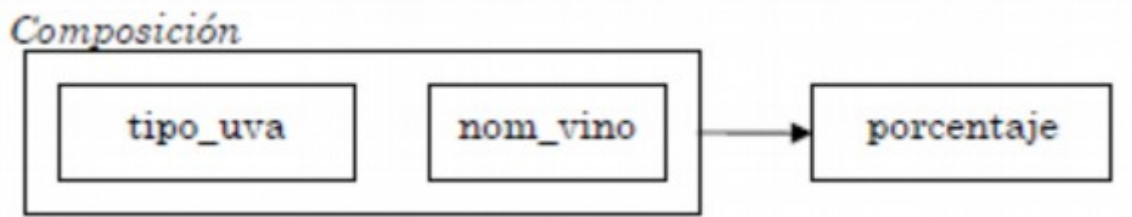
CAj: {dni\_catador} → Catador

CAj: {nom\_vino} → Vino

**Tipo\_vino**(nom\_vino, tipo\_vino)

CP: {nom\_vino}

CAj: {nom\_vino} → Vino



Como el único atributo de la tabla depende de manera completa de la CP, la relación ya está en 2FN.

### 3FN

Hay que eliminar las DF transitivas. Para ello es necesario calcular el diagrama de DF de cada relación:

#### **TABLA CATADOR:**

**Catador**(dni\_catador, nombre, experiencia, dni\_conyuge, fecha\_nacimiento))

Como todos los atributos dependen directamente de la CP, la relación ya está en 3FN.

#### **TABLA VINO:**

**Vino**(nom\_vino, añada, precio, bodega)

Todos los atributos dependen directamente de la CP, luego la relación ya está en 3FN.

#### **TABLA CATA:**

**Cata**(dni\_catador, nom\_vino, nota\_aromas, nota\_sabor)

Al depender todos los atributos de la CP la relación ya está en 3FN.

#### **TABLA TIPO VINO:**

**Tipo\_vino**(nom\_vino, tipo\_vino)

Como solo hay un atributo no-clave, no puede haber dependencias transitivas, luego la relación ya está en 3FN.

#### **TABLA COMPOSICIÓN:**

**Composición**(nom\_vino, tipo\_uva, porcentaje)

Como la relación solo tiene un atributo no-clave, no puede haber dependencias transitivas, luego la relación ya está en 3FN.

Así pues el conjunto final de relaciones en 3FN es el siguiente:

**Catador**(dni\_catador, nombre, experiencia, dni\_conyuge, fecha\_nacimiento))

CP: {dni\_catador}

**Vino**(nom\_vino, añada, precio, bodega)

CP: {nom\_vino}

**Cata**(dni\_catador, nom\_vino, nota\_aromas, nota\_sabor)

CP: {dni\_catador, nom\_vino}

CAj: {dni\_catador} → Catador

CAj: {nom\_vino} → Vino

**Tipo\_vino**(nom\_vino, tipo\_vino)

CP: {nom\_vino}

CAj: {nom\_vino} → Vino

**Composición**(nom\_vino, tipo\_uva, porcentaje)

CP: {nom\_vino, tipo\_uva}

CAj: {nom\_vino} → Vino