

Profesora: Olga Cuervo Miguélez

# TEMA 4

**DISEÑO LÓGICO. MODELO RELACIONAL**

Módulo: **Bases de Datos**

Ciclo: **DAM**

## 1. Objetivos

En el tema anterior aprendimos que los diagramas E-R se usan para describir gráficamente entidades y sus relaciones. En este tema se aprenderá cómo se puede usar el diseño E-R para diseñar una base de datos relacional, la forma en que los componentes básicos de una BD relacional se ajustan en tablas.



## 2. Introducción

Ya hemos visto que técnica (Diagramas Entidad/Relación) hay que utilizar para realizar el Modelo conceptual de datos. Ese Modelo conceptual se realiza en la fase de **Análisis** (dónde la pregunta sería “**Qué**” tenemos que hacer). En la siguiente fase, el **Diseño**, tendremos que realizar el modelo lógico de datos. En esta fase nos centraremos en “**Cómo**” tenemos que hacerlo.

FASES	MODELOS		
	Modelo de datos	Modelo de procesos	Modelo de eventos
Análisis	Modelo <u>conceptual</u> de datos	Análisis de procesos	Análisis de eventos
Diseño	Modelo <u>lógico</u> de datos	Diseño de procesos	Diseño de <u>interfaces de usuario</u>
Implementación	Modelo <u>físico</u> de datos	Codificación de procesos	Codificación de las interfaces de usuario

## 3. Modelo Lógico de Datos. Objetivos y Modelos

- **Base de Datos (BBDD):** Colección de datos interrelacionados almacenados sin redundancias perjudiciales o innecesarias.
- **Sistema de Gestión de Bases de Datos (SGBD):** Software para describir, recuperar y manipular los datos almacenados en una BBDD.

El **modelo lógico de datos** está orientado a describir los datos teniendo en cuenta “**Cómo**” se tienen que implementar los datos en un SGBD. Por tanto el modelo lógico depende del tipo de SGBD elegido.

---

Los **SGBD** se apoyan en modelos teóricos. Los principales modelos teóricos en los que se basan los SGBD son:

- Modelo Jerárquico (En desuso)
- Modelo En red (En desuso)
- **Modelo Relacional**
- Modelo Orientado a Objetos

El **objetivo** principal del modelo relacional **es proteger al usuario de la obligación de conocer las estructuras de datos físicas con las que se representa la información de una base de datos**. Desvincular estas estructuras de datos, que son complejas por naturaleza, del diseño lógico (Modelo Relacional), permite que la BD se pueda implementar en cualquier gestor de base de datos relacional (Oracle, MySQL, DB2...).

El modelo que siguen la gran mayoría de los SGBD comerciales es el Modelo Relacional, por tanto, este será el modelo que vamos a estudiar.

Los SGBD que se apoyan en el modelo relacional se denominan Sistemas de Gestión de Bases de Datos Relacionales (SGDBR) y las Bases de Datos que se crean con estos sistemas se denominarán Bases de Datos Relacionales.

Algunos de los SGBDR que existen en el mercado son: Access, SQL Server, Oracle ó MySQL Server.

En resumidas cuentas, como el modelo lógico de datos depende del SGBD elegido, y los SGBDR son los más utilizados, nosotros estudiaremos el **Modelo Lógico Relacional de Datos**.

## 4. Modelo lógico de datos. Modelo Relacional

El **Modelo Relacional**: Modelo de datos usado en la gestión de bases de datos basado en la *lógica de predicados*<sup>1</sup> y la *teoría de conjuntos*<sup>2</sup>. Este modelo considera la base de datos como una colección de relaciones.

---

<sup>1</sup> Basado en relaciones entre objetos, como también cualidades y atributos de tales objetos. Los objetos pueden ser personas, objetos físicos o conceptos. Relaciones o atributos se denominan predicado.

<sup>2</sup> Es la rama de las matemáticas que estudia los conjuntos, que son colecciones de objetos.

---

Componentes del Modelo Relacional:

- Una **estructura lógica de datos** representada por relaciones.
- Un **conjunto de reglas de integridad** para hacer cumplir que los datos sean y sigan siendo consistentes a lo largo del tiempo.
- Un **conjunto de operaciones** que define la forma en que los datos se manipulan.

**Base de Datos Relacional:** Conjunto de tablas con registros y atributos, que tienen un campo común, el cual es igual en ambas tablas y recibe el nombre llave principal, ID o clave primaria.

Este modelo supone una serie de ventajas:

- **Independencia física:** el modo en que se almacenan los datos no influye en la definición de los mismos, que se produce en esta fase.
- **Independencia lógica:** Las operaciones sobre los datos no afectan a cómo los datos son tratados.
- **Flexibilidad:** es flexible porque puede ofrecer diversos esquemas externos.
- **Uniformidad:** en cuanto a la representación de los datos es una técnica uniforme, facilitando de este modo el empleo de los mismos.
- **Sencillez:** Es fácil de comprender y utilizar.

## 4.1. Conceptos del Modelo Relacional

### 4.1.1. Tablas

Una **tabla** es percibida como una estructura en dos dimensiones compuesta de tuplas o registros y de columnas o campos. Una tabla también se conoce como una **relación** porque el creador del modelo relacional, Codd, utilizó el término **relación** como sinónimo de tabla. Podemos considerar una **tabla** como una representación persistente de una relación lógica, es decir, una relación cuyo contenido se puede guardar de manera permanente para un uso futuro.

Estructura de una tabla o relación:

- **Campos o atributos** (columnas).
- **Tuplas o registros** (filas).
- **Dominio** será el conjunto de valores que puede tomar un atributo.
- **Grado** el número de atributos de una tabla.
- **Cardinalidad**, es el número de tuplas de la misma.

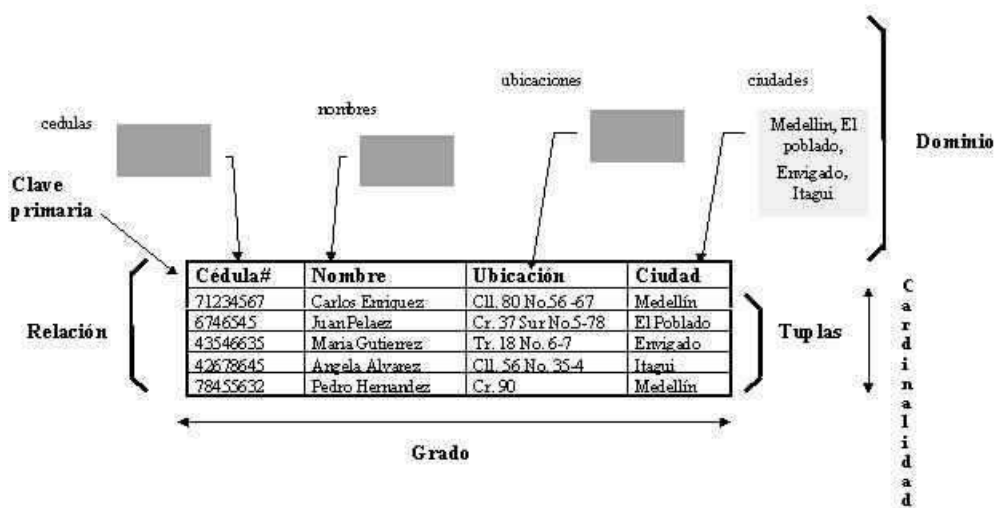
**NOTA:** No confundir la cardinalidad y el grado del modelo relacional con la cardinalidad y el grado en los diagramas E/R. Simplemente tienen el mismo nombre pero en el Diagrama E/R significan cosas distintas.

**Ejemplo:** en el caso de la pizzería, las tablas del modelo relacional se corresponden con las entidades del modelo entidad-relación, en este caso una tabla podría ser *repartidor*.

Tabla REPARTIDOR		ATRIBUTOS ó CAMPOS			
		ID_REPARTIDOR	Fecha_nacimiento	Direccion	Telefono
TUPLA 1	→	R001	24/06/1079	Alcala 18	918767876
TUPLA 2	→	R002	04/05/1983	Princesa 3	916543782
		.....	.....	.....	.....
TUPLA N	→	R00n	12/09/1985	Filipinas 78	917653784

Ejemplo: Grado: 4 Cardinalidad: N

*Tabla en el modelo relacional*



### Características de una tabla relacional

- Una *tabla* es percibida como una estructura *bidimensional* compuesta por *filas* y *columnas*.
- Cada *fila* representa una *ocurrencia única* de entidad dentro del conjunto de entidades.
- Cada *columna* representa un *atributo* y cada columna tiene un *nombre distinto*.
- Cada *intersección de fila/columna* representa un *valor único de datos*.
- Todos los *valores de una columna* deben tener el *mismo formato*.
- Cada *columna* tiene un *intervalo específico de valores* conocido como dominio de atributos.
- El *orden de las columnas y filas* no tiene importancia en el SGBD.
- Cada tabla debe tener un *atributo o combinación de atributos que identifique de manera unívoca cada fila*.

### 4.1.2. Claves

En el modelo relacional, las claves son importantes porque:

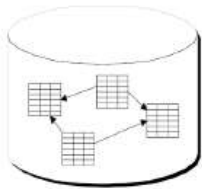
- Se usan para asegurar que cada fila de la tabla es identificable de manera única.
- Se usan para establecer relaciones entre tablas y asegurar la integridad entre los datos.

#### Claves de bases de datos relacional

Claves de bases de datos relacional	
<b>Superclave</b>	Un atributo o combinación de atributos de una tabla que de modo único identifica cada fila de la tabla.
<b>Clave Candidata</b>	Una <i>Superclave mínima</i> . Una superclave que no contiene atributos innecesarios.
<b>Clave Primaria (Primary Key)</b>	Una clave candidata seleccionada para identificar de modo único todos los valores de atributo en cualquier fila. No puede contener entradas nulas.
<b>Clave Alternativa (Alternative Key)</b>	Claves candidatas que no han sido seleccionadas como clave primaria.
<b>Clave Secundaria (Secondary Key)</b>	Un atributo o combinación de atributos de una tabla que se usa estrictamente para fines de recuperación de datos.
<b>Clave Foránea, Ajena o Extranjera (Foreign Key)</b>	Un atributo o combinación de atributos de una tabla cuyos valores deben corresponderse con la clave primaria en otra tabla o ser nulos.

Modelo Entidad Relación	Modelo Relacional
Ocurrencia	Fila ó Tupla
Atributo	Columna
Entidad	Tabla ó Relación
Grado	Número de columnas
Cardinalidad	Número de Filas

#### 4.1.3. Interrelaciones entre tablas

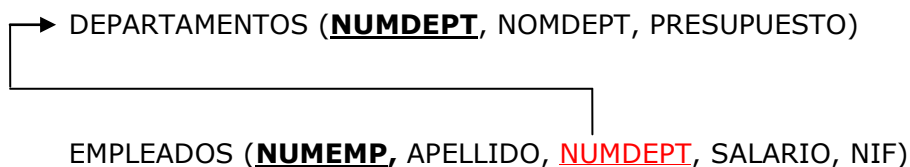


Una Base de Datos (relacional o de otro tipo) es una colección de datos (en nuestro caso relaciones o tablas) *interrelacionados*. Por tanto necesitamos asociar una relación (tabla) con otras tablas mediante las *claves ajenas*.

**Clave ajena (Foreign Key):** Se denomina clave ajena de una relación **R2** a uno o varios atributos cuyos valores han de coincidir con los valores de la clave primaria de una relación **R1**.

- R1 y R2 pueden ser la misma relación (o Tabla).
- La clave ajena y la clave primaria deben estar definidas sobre los mismos dominios.

Clave primaria			Clave primaria			Clave ajena	Clave alternativa	
NUMDEPT	NOMDEPT	PRESUPUESTO	NUMEMP	APELLIDO	NUMDEP	SALARIO	NIF	
D1	Marketing	1000	E1	López	D1	40	52.123.456-X	
D2	Desarrollo	1200	E2	Fernández	D2	42	40.123.456-Y	
D3	Investigación	500	E3	Martínez	D3	50	30.123.456-Z	
			E4	Sánchez	D2	34	20.123456-A	



Según este ejemplo. ¿Cuál sería el nombre del departamento al que pertenece el empleado Martínez?

### INVESTIGACION

#### 4.1.4. Reglas o Restricciones de Integridad

Las *reglas de integridad* de una base de datos relacional son muy importantes para un buen diseño de base de datos. Muchos de los SGBD hacen cumplir automáticamente las reglas de integridad. No obstante, es mucho más seguro verificar que el diseño cumple con dichas reglas.

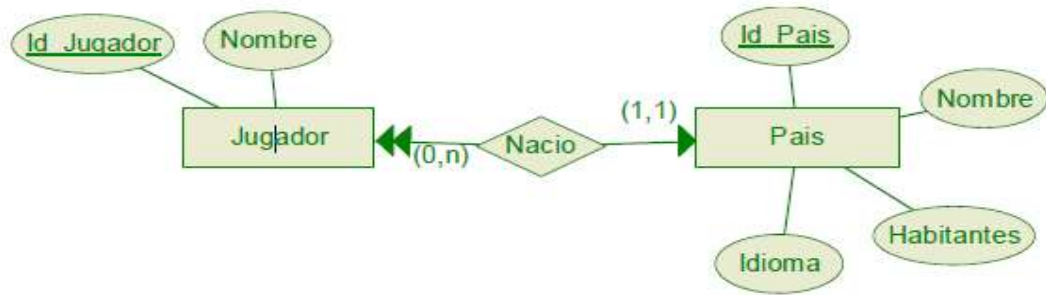
**NOTA:** El concepto NULO (NULL) no es ningún valor. No significa un CERO o un ESPACIO.

Restricciones Inherentes al Modelo ó Integridad de Entidad (impuestas por el propio modelo relacional)	
Todas las entradas de clave primaria son únicas y ninguna parte de una clave primaria puede ser nula.	
Cada fila tendrá una identidad única y los valores de clave foránea pueden referirse debidamente a valores de clave primaria.	
Una clave foránea puede tener una entrada nula, mientras que no forme parte de la clave primaria.	
NO se admiten atributos multivaluados.	
El orden de los campos (atributos) no es significativo.	
El orden de las tuplas (filas) no es significativo.	

Restricciones de Usuario (permiten recoger con la mayor fidelidad posible el mundo real)	
<b>Unicidad (UNIQUE)</b>	Obliga a que los valores de un conjunto de atributos (uno o más) no puedan repetirse en una relación. Esta restricción permite la definición de claves alternativas.
<b>Obligatoriedad (NOT NULL):</b>	No se admiten nulos, es decir, el valor de un atributo será siempre un valor que esté dentro del dominio al que pertenece el atributo.



## EJEMPLO



Las tablas (relaciones) que se corresponden con este diagrama entidad relación en el modelo relacional serán las siguientes:

Tabla (Relación): <b>PAISES</b>			
<u>IdPais</u>	Nombre	Habitantes	Idioma
ESP	ESPAÑA	9.000.000	ESPAÑOL
ING	INGLATERRA	12.000.000	INGLES
FRA	FRANCIA	11.345.655	FRANCÉS
BRA	BRASIL	23.098.476	PORTUGUÉS
HOL	HOLANDA	5.000.765	HOLANDÉS

Tabla (Relación): <b>JUGADORES</b>		
<u>IdJugador</u>	Nombre	<u>IdPais</u>
J001	Raul Lopez	ESP
J002	Robinho	BRA
J003	Zinedine Zidane	FRA
J004	Beckam	ING

En esta tabla, **IdPais**, actúa como clave ajena. Relacionando a la tabla con JUGADOR.

**Ejemplo1:** Sí interpretamos la relación de que un jugador pueda poseer sólo una nacionalidad, tendríamos:

**PAISES** (idPais, nombre, habitantes, idioma)

**JUGADORES** (idJugador, nombre, idPais)

**Ejemplo2:** Sí interpretamos la relación de que un jugador pueda poseer varias nacionalidades tendríamos un atributo multivaluado 'idPais'.

**PAISES** (idPais, nombre, habitantes, idioma)

**JUGADORES** (idJugador, nombre)

**POSEE\_NACIONALIDAD** (idPais, idJugador)

La **existencia de integridad referencial** tiene consecuencias sobre las **operaciones** de **modificación** de la clave y **borrado** de las tuplas, por lo que se debe de escoger entre una de las siguientes acciones que controlan la consistencia de los datos cuando se produce una de estas operaciones de borrado o modificación:

*Notación respecto al Borrado y Modificación en el cuadro siguiente:*

*B/M: Borrado/Modificación (Delete/Cascade).*

*R1: Relación o tabla a la que se referencia.*

*R2: Relación o tabla que contiene una clave ajena que referencia a R1.*

*BR, MR: Borrado Restringido, Modificación Restringida (No Action)*

*BC, MC: Borrado en Cascada, Modificación en Cascada (Cascade)*

*BN, MN: Borrado/Modificación a Nulo (Set Null)*

*BD, MD: Borrado/Modificación por defecto (Set Default)*

Tabla (Relación): PAISES			
IdPais	Nombre	Habitantes	Idioma
ESP	ESPAÑA	9.000.000	ESPAÑOL
ING	INGLATERRA	12.000.000	INGLES
FRA	FRANCIA	11.345.655	FRANCÉS
BRA	BRASIL	23.098.476	PORTUGUÉS
HOL	HOLANDA	5.000.765	HOLANDES

Tabla (Relación): JUGADORES		
IdJugador	Nombre	IdPais
J001	Raul Lopez	ESP
J002	Robinho	BRA
J003	Zinedine Zidane	FRA
J004	Beckam	ING

- **NO ACTION (operación restringida):** solo se permite borrar (BR) tuplas o modificar (MR) la clave primaria si no existen claves ajenas que la referencien.
- **CASCADE (transmisión en cascada):** el borrado de una tupla o la modificación de la clave primaria en R1 lleva consigo el BC/MC en cascada de las tuplas de R2. Por ejemplo, si borramos la fila de INGLATERRA en la tabla de PAISES, tendríamos que eliminar todas las filas de la tabla JUGADORES, que contengan como idPais ING, las tablas del ejemplo quedarían del siguiente modo:

Tabla (Relación): PAISES				Tabla (Relación): JUGADORES		
IdPais	Nombre	Habitantes	Idioma	IdJugador	Nombre	IdPais
ESP	ESPAÑA	9.000.000	ESPAÑOL	J001	Raul Lopez	ESP
<b>BORRADA</b>				J002	Robinho	BRA
FRA	FRANCIA	11.345.655	FRANCÉS	J003	Zinedine Zidane	FRA
BRA	BRASIL	23.098.476	PORTUGUÉS			
HOL	HOLANDA	5.000.765	HOLANDES			

- **SET NULL (puesta a nulos):** el borrado o modificación de tuplas de R1 pone a nulo los valores de las tuplas de R2. En el ejemplo, si borrásemos el país con idPais ING, las tablas quedarían del siguiente modo:

Tabla (Relación): PAISES				Tabla (Relación): JUGADORES		
IdPais	Nombre	Habitantes	Idioma	IdJugador	Nombre	IdPais
ESP	ESPAÑA	9.000.000	ESPAÑOL	J001	Raul Lopez	ESP
BORRADA				J002	Robinho	BRA
FRA	FRANCIA	11.345.655	FRANCÉS	J003	Zinedine Zidane	FRA
BRA	BRASIL	23.098.476	PORTUGUÉS	J004	Beckam	NULL
HOL	HOLANDA	5.000.765	HOLANDES			

- **SET DEFAULT (puesta a valor por defecto):** el borrado o modificación de tuplas de R1 pone a un valor por defecto definido los valores de la clave ajena R2. En las tablas del ejemplo, si consideramos que eliminamos el país INGLATERRA, la tabla quedaría del siguiente modo, considerando que tomamos como valor por defecto NGN, que significa ninguno:

Tabla (Relación): PAISES				Tabla (Relación): JUGADORES		
IdPais	Nombre	Habitantes	Idioma	IdJugador	Nombre	IdPais
ESP	ESPAÑA	9.000.000	ESPAÑOL	J001	Raúl Lopez	ESP
BORRADA				J002	Robinho	BRA
FRA	FRANCIA	11.345.655	FRANCÉS	J003	Zinedine	FRA
BRA	BRASIL	23.098.476	PORTUGUÉS	J004	Beckam	NGN
HOL	HOLANDA	5.000.765	HOLANDES			

## 5. Transformación del modelo E/R al modelo lógico relacional

La transformación del Modelo Entidad-Relación al Modelo Relacional produce cierta pérdida de semántica ya que las entidades y las interrelaciones se convierten en tablas, pero la pérdida de semántica no implica, sin embargo, un peligro para la integridad de la base de datos, ya que se definen las restricciones de integridad referencial que aseguran la conservación de la misma.

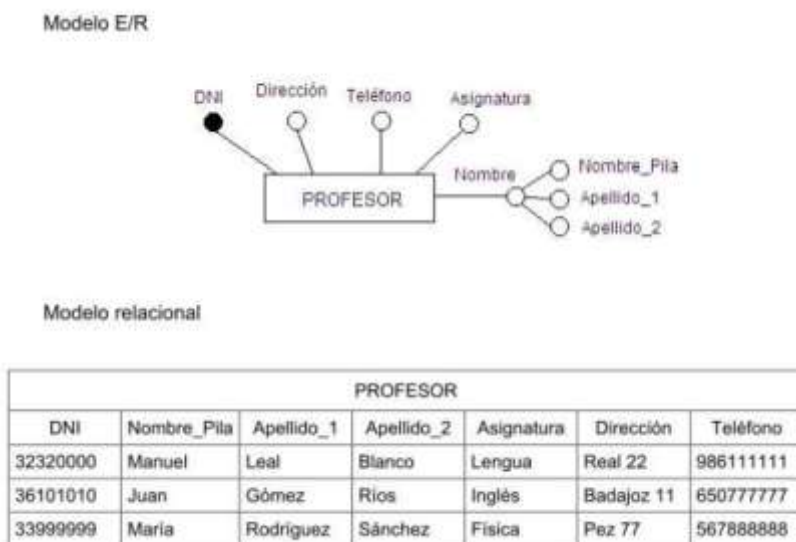
## 5.1. Tres reglas básicas en las transformaciones del modelo E-R

Existen **tres reglas básicas** que se deben cumplir en la transformación del modelo entidad relación al modelo relacional, reglas que se explicarán más detalladamente en los apartados que siguen a éste, son:

- Toda entidad se convierte en una relación (o tabla).
- Toda interrelación N:M se transforma en una relación (también llamada tabla).
- Toda interrelación 1:N se traduce en el fenómeno “propagación de claves”.

## 5.2. Transformación de entidades

- Cada entidad del esquema E/R dará lugar a una nueva **relación ó tabla** cuya clave primaria es el identificador principal de la entidad.
- Cada atributo de una entidad se transforma en un atributo de la relación creada para la entidad teniendo en cuenta los distintos tipos de restricciones semánticas.
  - **Atributos univaluados:** dan lugar a un atributo de la relación.
  - **Atributos multivaluados:** dan lugar a una nueva relación (tabla) cuya clave primaria es la concatenación de la clave primaria de la entidad en la que se sitúa el atributo multivaluado más el nombre del atributo multivaluado.
  - **Atributos obligatorios:** atributos con la restricción de NOT NULL.
  - **Atributos opcionales:** atributos que pueden tomar ‘valores’ NULL.
  - **Identificador principal:** atributos que forma la clave primaria.
  - **Identificador alternativo:** atributos con la restricción de UNIQUE.



Nosotros lo representaremos así:

**PROFESORES** (DNI, nombrePila, apellido1, apellido2, asignatura, dirección, telefono)

### 5.3. Transformación de entidades débiles

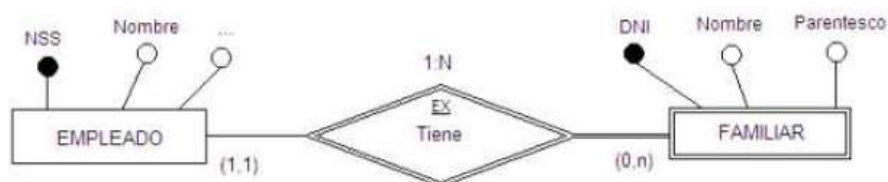
Cada atributo de una entidad débil se transforma en una columna de la relación, igual que en el caso anterior pero, además, hay que incluir como clave foránea el/los atributos que formen la clave primaria en la tabla principal. Además, en el caso de que la entidad sea débil en identificación, la clave principal de esta relación o tabla estará formada por los atributos de la clave foránea más los atributos que formen la clave parcial de la entidad débil.

#### ❖ DEPENDENCIA EN EXISTENCIA

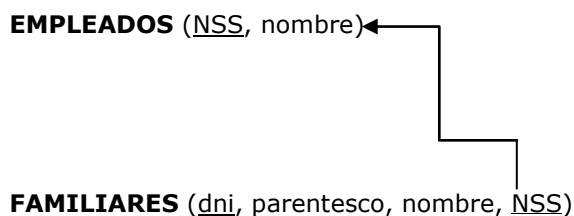
Una interrelación **1:N** de dependencia en existencia **se transforma propagando la clave primaria como clave ajena de la entidad fuerte a la entidad débil** con las siguientes características:

- No permitir nulos.
- Modificación en cascada (ON UPDATE CASCADE).
- Borrado en cascada (ON DELETE CASCADE). Es decir, no puede existir ninguna ocurrencia de la entidad hijo que no esté relacionada con una ocurrencia de la entidad padre.

Modelo E/R



Modelo Relacional:



#### ❖ DEPENDENCIA EN IDENTIFICACION

Suelen ser siempre **1:N** por lo que no generan tablas nuevas. La propagación de la clave se hace de forma que la clave primaria de la entidad débil estará formada por la **concatenación de las dos claves** (además de ser clave ajena).

Modelo E/R



Modelo Relacional:

**PRESTAMOS** (numPrestamo, importe)

**PAGOS** (numPrestamo, numPago, fecha, importePago)

## 5.4. Transformación de relaciones binarias

### 5.4.1. INTERRELACIÓN N:M

Siempre dan lugar a **NUEVA TABLA o RELACIÓN** cuya **clave primaria será la concatenación de las claves primarias de las entidades que interrelacionan**. De esta forma los atributos que forma la clave primaria en esta nueva relación son claves ajenas respecto a las relaciones en las que estos atributos son clave primaria.

Por ejemplo:

Modelo E/R:



Modelo relacional:

**PROFESOR** (Codigo\_P, Nombre\_P, Direccion...)

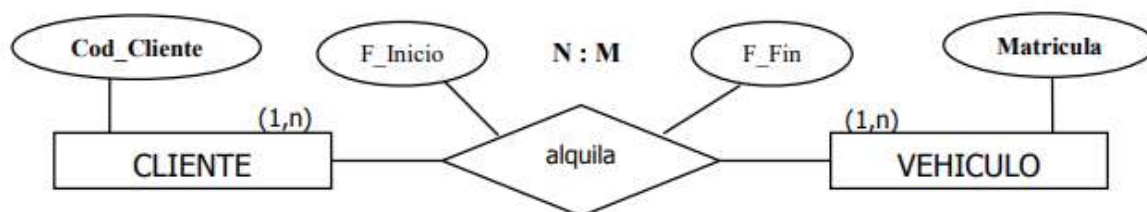
**CURSO** (Codigo\_C, Nombre\_C, NumHoras...)

**IMPARTE** (Codigo\_P, Codigo\_C)

### TRANSFORMACION DE LA DIMENSION TEMPORAL

En algunos casos en que la relación tenga **atributos de tipo fecha**, será necesario incluir al menos una fecha como parte del atributo identificador principal para recoger la dimensión temporal del modelo. En otros casos la fecha puede ser una entidad más o solo un atributo.

**Ejemplo:** Relación entre los clientes que alquilan los vehículos de una empresa de alquileres. Recogemos los alquileres realizados a nuestros clientes a lo largo del tiempo.



alquila ( Cod\_Cliente, Matricula, F\_Inicio, F\_Fin)

CLIENTE ( Cod\_Cliente, ..... )

VEHICULO ( Matricula, ..... )

## 5.4.2. INTERRELACIÓN 1:N

Existen dos posibilidades:

- 1. Propagación de clave:** Como norma general, se propagan los atributos que forman la clave principal del tipo de entidad de cardinalidad 1 a la que tiene cardinalidad N, desapareciendo el nombre de la interrelación. Si la interrelación tuviese atributos propios y no son muchos, estos se propagarían en el mismo sentido que la clave.

Modelo E/R:



Modelo relacional:

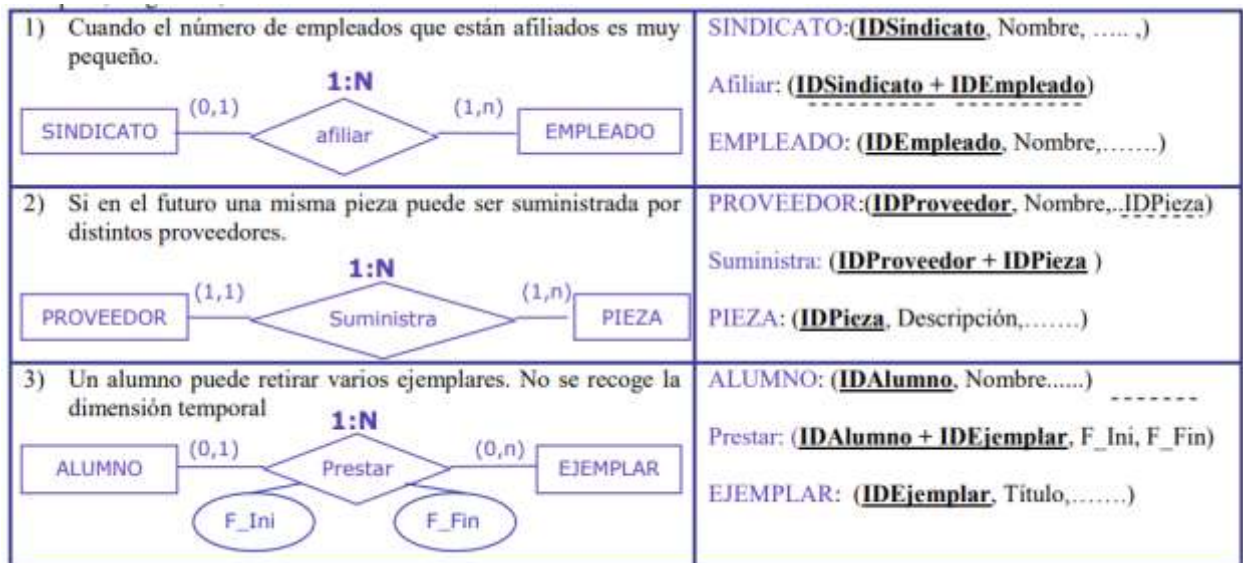
**PROFESOR** ( Codigo\_P, Nombre\_P, Dirección..., Codigo\_D )

**DEPARTAMENTO** ( Codigo\_D, Nombre\_D... )

**EXCEPCIONES:** En los siguientes casos interesa más crear una nueva tabla a partir de la relación como en el caso de correspondencias M:N

- Quando el número de ocurrencias de la entidad que propaga la clave es muy pequeño y cabe la posibilidad de que al propagar la clave quedan muchos **valores repetidos o nulos**.
- Quando se prevea que en el futuro se puede convertir en una **relación N:M**.

- c. Cuando la relación tenga atributos propios. En algunos casos se pueden migrar estos atributos junto con la clave pero, en general, se crea una nueva tabla.



- 2. Transformar la interrelación 1:N como si fuese una interrelación N:M.** Es decir, se crea una nueva relación (tabla) cuya clave primaria es, en este caso, sólo la clave primaria de la tabla a la que le corresponde la cardinalidad N. Por ejemplo, para el modelo E/R anterior, la transformación al modelo relacional daría lugar a las relaciones siguientes:

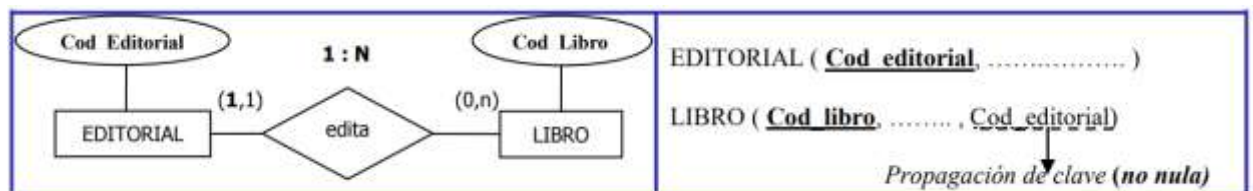
**PROFESOR** (Codigo\_P, Nombre\_P, Dirección...)

**DEPARTAMENTO** (Codigo\_D, Nombre\_D...)

**PERTENECE** (Codigo\_P, Codigo\_D)

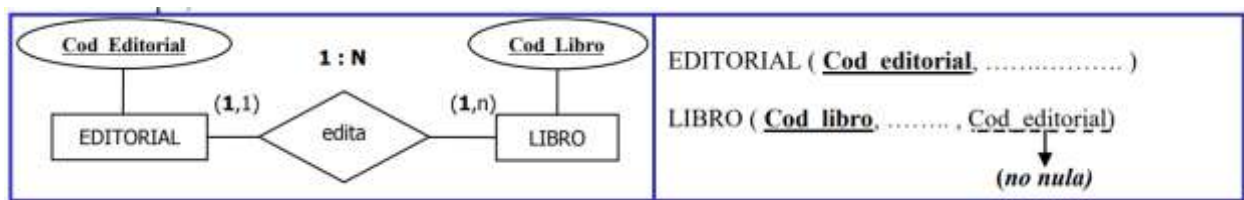
### CARDINALIDADES:

- a. Cuando la entidad que tiene cardinalidad máxima **1**, tiene también 1 de cardinalidad mínima, tendremos que tener en cuenta al propagar la clave que en la tabla que recibe la clave, como clave extranjera, no pueda tener valores nulos.



- b. Cuando la entidad que tiene cardinalidad máxima **n**, tiene de cardinalidad mínima 1, tendremos que controlar por software que, al dar de alta una fila de la otra tabla se introduzca al menos una fila en esta.





### 5.4.3. INTERRELACIONES 1:1

Una interrelación de tipo 1:1 es un caso particular de una N:M o también de una 1:N, por lo que no hay regla fija para la transformación: puede crearse una nueva relación o bien efectuar una propagación de clave. Puede optarse por la solución basada en:

- Recoger la mayor cantidad de semántica posible.
- Tener en cuenta las cardinalidades mínimas.
- Evitar los valores nulos.
- Motivos e eficiencia.

- a) Cuando las cardinalidades de **ambas entidades son (1,1)** se pueden adoptar distintas soluciones:
- a. No se necesitan 2 tablas, se puede crear **una única tabla** en la que se incluyan los atributos de las dos entidades, cuya clave principal será cualquiera de los atributos identificadores principales.
  - b. **Propagar la clave** de cualquiera de ellas a la otra tabla, teniendo en cuenta a cuál de ellas se le efectúan los accesos más frecuentes.

El siguiente ejemplo se ha resuelto mediante la creación de una nueva relación.



Modelo relacional:

**PROFESOR** (Codigo\_P, Nombre\_P, Direccion...)  
**CURSO** (Codigo\_C, Nombre\_C, Num\_alumnos...)  
**ES\_TUTOR\_DE** (Codigo\_P, Codigo\_C)

La clave de la nueva relación (ES\_TUTOR\_DE) puede ser cualquiera de sus dos atributos.

- b) **Sí una entidad de las que participan en la interrelación posee cardinalidad (0,1) mientras que la otra posee (1,1)**, se propaga la clave de la entidad con cardinalidad (1,1) a la que tiene (0,1). Habrà que controlar además que la clave propagada como clave ajena no pueda tomar valores nulos.

Por ejemplo:

Modelo E/R



Modelo relacional

**PROFESOR** (Codigo\_P, Nombre\_P, Dirección...)  
**DEPARTAMENTO** (Codigo\_D, Nombre\_D..., Codigo\_P)

Se ha propagado la clave de la entidad PROFESOR ("Codigo\_P") porque es la entidad que tiene cardinalidad (1,1). Esta solución evita los valores nulos que se producirían de haber hecho la propagación en sentido contrario. Es decir, si el paso a modelo relacional se hubiera hecho propagando la clave de la entidad DEPARTAMENTO a la relación PROFESOR, el resultado hubiera sido:

**PROFESORES** (Codigo\_P, Nombre\_p, Direccion... Codigo\_D<sup>3</sup>)

**DEPARTAMENTOS** (Codigo\_P, Nombre\_D...)

Como hay profesores que no dirigen ningún departamento, lo que viene indicado por la cardinalidad (0,1), habrá tuplas en la relación PROFESOR en las que el campo "Codigo\_D" no tenga ningún valor, o lo que es igual, tenga valor nulo.

- c) **Sí las dos entidades** que participan en la interrelación con **cardinalidad (0,1)**, la relación se transforma en una nueva tabla para evitar los valores nulos. Por ejemplo:

<sup>3</sup> Clave Ajena NOT NULL

## Modelo E/R



## Modelo relacional

**HOMBRE** (DNI\_H, Nombre\_H, Dirección\_H...)  
**MUJER** (DNI\_M, Nombre\_M, Dirección\_M...)  
**CONTRAE\_MAT.** (DNI\_H, DNI\_M)

La clave de la nueva relación (CONTRAE\_MATRIMONIO) puede ser cualquiera de sus dos atributos.

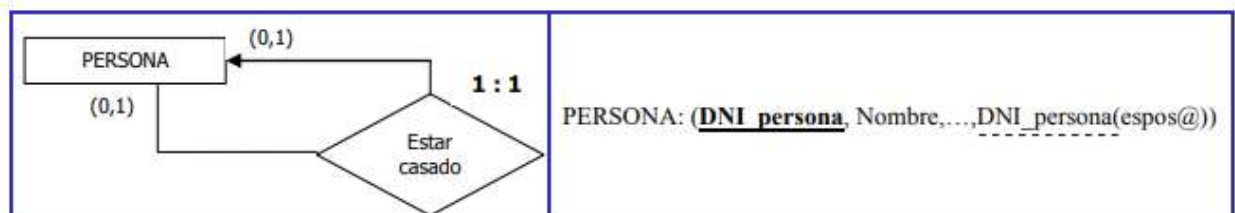
Al crear una nueva relación, se evitan los valores nulos que se producirían de haber propagado la clave de la relación MUJER a la relación HOMBRE o viceversa, ya que, según refleja la cardinalidad, no todos los hombres ni todas las mujeres están casados.

## 5.5. Transformación de las relaciones reflexivas

Se trata de relaciones en las que solo participa una entidad. Como **regla general** toda relación reflexiva se convierte en dos tablas: una para la entidad y otra para la relación. Se pueden presentar los siguientes casos:

### Relación 1:1

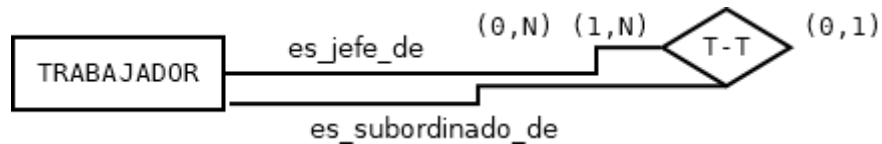
No se crea una tabla para la relación. La clave de la entidad se repite, con lo que la tabla resultante tendrá ese atributo dos veces, una como clave primaria y otra como clave ajena de ella misma.



### Relación 1:N

En este caso hay que tener en cuenta la cardinalidad del lado muchos (**N**):

- Si no es obligatoria se crea una nueva tabla cuya clave será la de la entidad del lado muchos (Cod\_tema\_S) y se propaga la clave a la nueva tabla como clave ajena.
- Si es siempre obligatoria no se crea una nueva tabla.



**Solucion A:** (cardinalidad mínima 0 o 1)

T-T (cod Trabajador, cod Trabajador jefe)

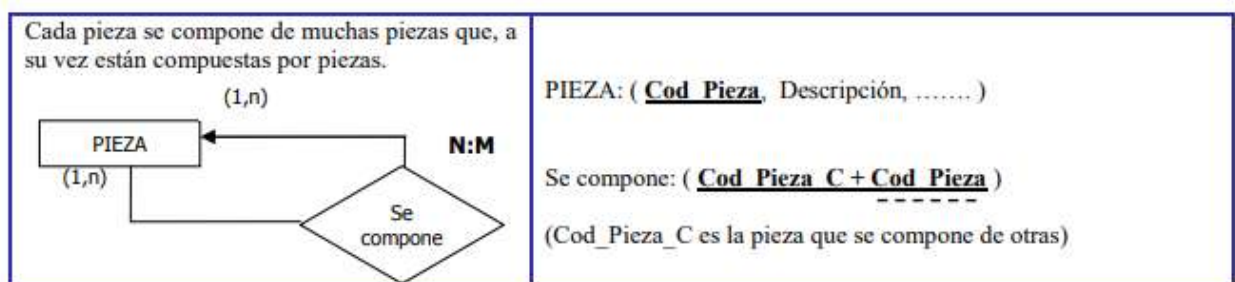
TRABAJADORES (cod Trabajador,.....)

**Solucion B:** (cardinalidad mínima 1)

TRABAJADORES (cod Trabajador,....., cod Trabajador jefe)

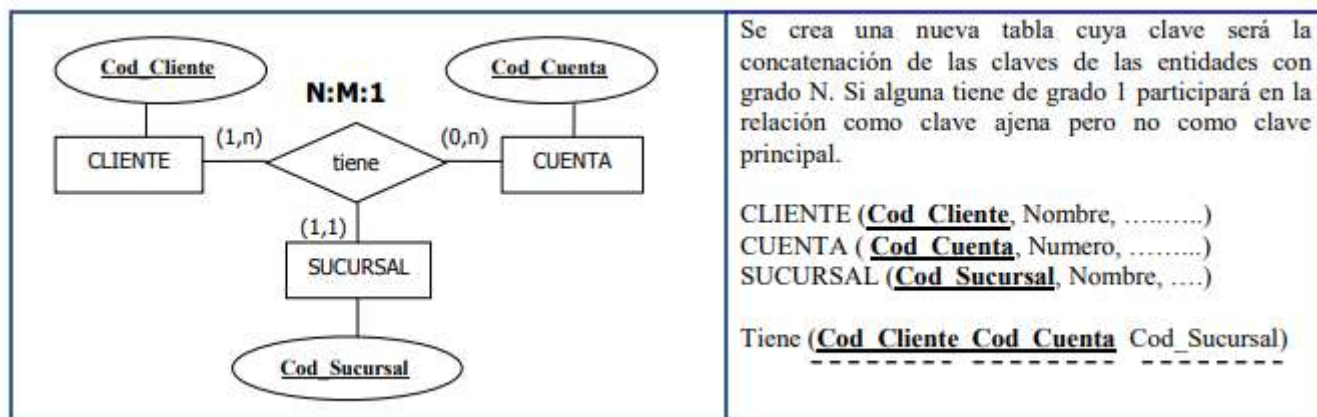
### Relación N:M

La tabla que resulta de la relación contendrá dos veces la clave primaria de la entidad del lado muchos, más los atributos de la relación, si los hay. La clave de esta nueva tabla será la combinación de las dos.



## 5.6. Transformación de relaciones ternarias

Se crea una nueva tabla para la relación. Esta tabla tendrá como **claves ajenas** las claves primarias de las relaciones que relaciona, y su **clave primaria** estará formada por las claves ajenas de aquellas tablas en las que su cardinalidad es (x, N). Si alguna entidad tiene grado 1 participará en la relación como clave ajena pero no como clave principal.



## 5.7. Transformación de atributos multivaluados

Para los **atributos multivaluados** se creará una relación o tabla que incluya:

- La **clave primaria** (PK) de la relación correspondiente al tipo de entidad (o al tipo de interrelación) que contiene el atributo multivaluado.
- Un atributo (A) que corresponda al propio atributo multivaluado.

La **clave primaria de la nueva relación** estará formada por K y A.

Por ejemplo, supongamos en el siguiente esquema el atributo multivaluado "Ubicación", referente a la entidad DEPARTAMENTO:



Al transformarlo al modelo relacional, obtenemos las relaciones que se indican a continuación:

**PROFESOR** (Código\_P, Nombre\_P, Dirección...)  
**DEPARTAMENTO** (Código\_D, Nombre\_D..., Código\_P)  
**UBICACIÓN** (Código\_D, Ubicación)

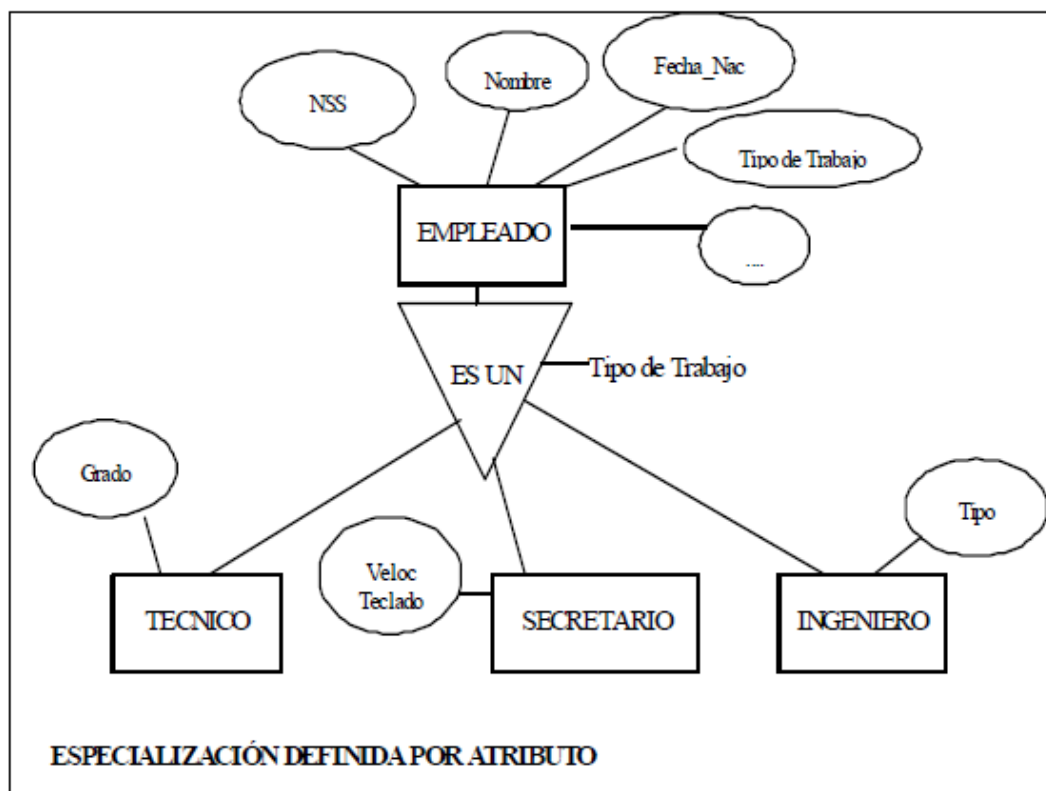
Se ha creado una nueva relación llamada UBICACIÓN, que tiene como atributos:

- La clave primaria de la relación DEPARTAMENTO, correspondiente a la entidad del mismo nombre, la cual contiene el atributo multivaluado.
- Un atributo llamado "Ubicación", correspondiente al propio atributo multivaluado.

Ambos atributos conforman la clave primaria.

## 5.8. Transformación de relaciones jerárquicas

Suponiendo el siguiente diagrama ER, para convertir cada superclase **C** con **m** subclases {s1,s2,...,Sn} a relaciones ...



... podríamos seguir una de las siguientes opciones:

**1) Crear una sola tabla** con todos los atributos de la entidad y de los subtipos, añadiendo como un atributo más el atributo discriminante. Esto se aplica cuando:

- a. Los subtipos se diferencian en muy pocos atributos.
- b. Las relaciones que los asocian al resto de las entidades sean las mismas para los subtipos.

**Si la jerarquía es:**

- **Total:** el atributo discriminante no admitirá nulos.
- **Parcial:** el atributo discriminante si admitirá nulos.

**Si entre los subtipos puede haber:**

- **Solapamiento:** Se forman grupos repetitivos, por tanto será necesario crear una nueva tabla que asocie el atributo discriminante con el supertipo.
- **Exclusividad:** No es necesaria una tabla nueva.

**Ejemplo:** Exclusiva

**EMPLEADOS** (NSS\_EMP, NOMBRE\_EMP, FNAC\_EMP, TIPOTRABAJO, VELO\_SEC, GRADO\_TEC, TIPO\_ING)

**Ejemplo:** Solapada

Como en la opción anterior, pero en lugar de añadir un solo campo "tipo", se añaden varios campos que indiquen si se cumple un perfil.

**EMPLEADOS** (NSS\_EMP, NOMBRE\_EMP, FNAC\_EMP, VELO\_SEC, GRADO\_TEC, TIPO\_ING esSecretaria, esTecnico, esIngeniero)

**2) Crear una tabla para cada tipo y subtipos** que haya. Esto se aplica cuando:

- a. Existen muchos atributos distintos entre los subtipos.
- b. Se quieren mantener los atributos comunes en una tabla.

**EMPLEADO** (NSS\_EMP, NOMBRE\_EMP, FNAC\_EMP, TIPOTRABAJO)

**SECRETARIO** (NSS\_EMP, VELO\_SEC)

**TECNICO** (NSS\_EMP, GRADO\_TEC)

**INGENIERO** (NSS\_EMP, TIPO\_ING)

**3) Crear una tabla por cada subtipo**, incluyendo los atributos comunes en cada una. Esto se aplica cuando:

- a. Existen muchos atributos distintos entre los subtipos.
- b. Los accesos a los datos de los subtipos siempre afectan a los atributos comunes.

---

**SECRETARIO (NSS\_EMP, NOMBRE\_SEC, FNAC\_SEC, VELO\_SEC)**

**TECNICO (NSS\_EMP, NOMBRE\_TEC, FNAC\_TEC, GRADO\_TEC)**

**INGENIERO (NSS\_EMP, NOMBRE\_ING, APE1, FNAC\_ING, TIPO\_ING)**

**Ventajas e inconvenientes:**

**Opción 1)** Es la más rápida por tener que acceder a una sola entidad.

**Opción 2)** La menos eficiente. La mejor desde un punto de vista semántico.

**Opción 3)** Más eficiente en consultas que afecten a todos los atributos de un subtipo.

Menos eficiente en consultas que afecten a los atributos comunes.

Introduce redundancias.

Es la que pierde más semántica.

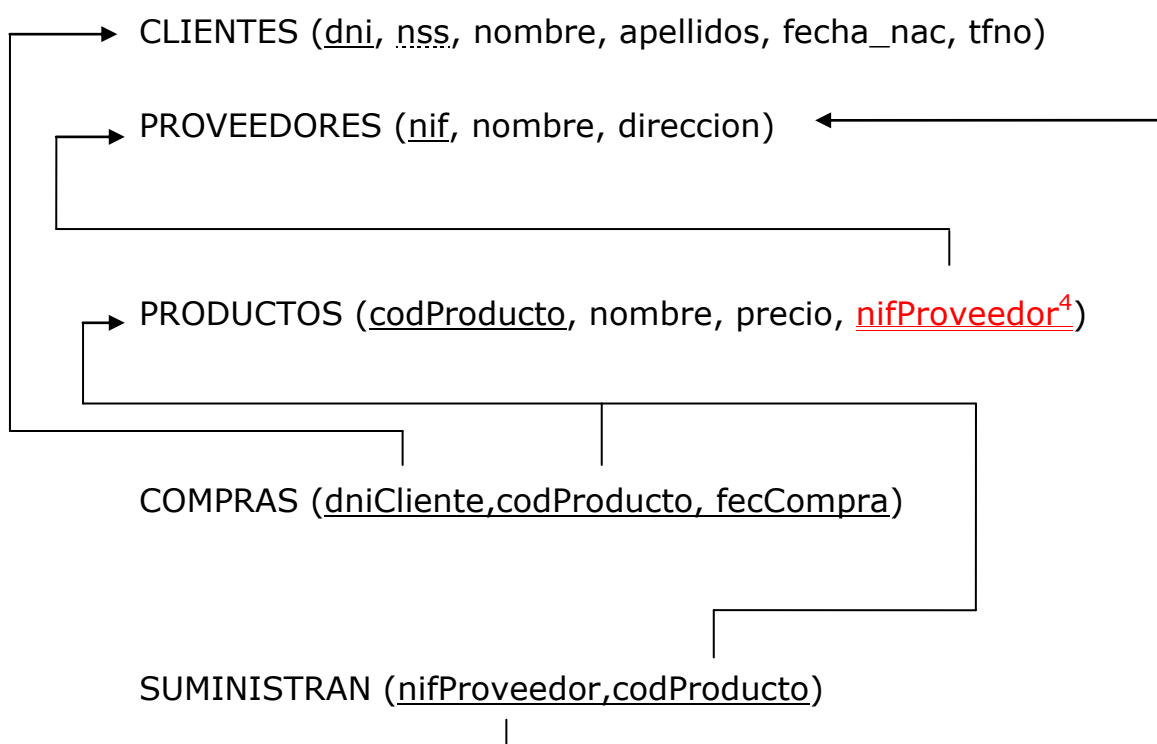
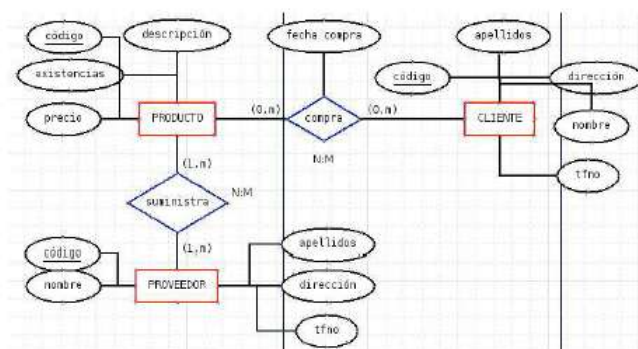
## 6. Notación

Un esquema relacional se representará mediante un grafo, conocido como grafo relacional; básicamente, se trata de un grafo dirigido cuyos nodos son las relaciones de la BD y los arcos representan las restricciones de clave ajena, y en el aparecerán además de las distintas relaciones con sus atributos y las restricciones de clave ajenas las restricciones de clave primaria, unicidad y obligatoriedad. Las convecciones empleadas para la representación del grafo son:

- a) El **nombre de las tablas** en mayúsculas y plural seguido de los atributos encerrados entre paréntesis.
- b) Las **claves primarias** subrayadas con trazo continuo.
- c) Las **claves alternativas** subrayadas con trazo discontinuo.
- d) Las **claves ajenas o foráneas** con doble subrayado y referencian a la relación en la que son clave primaria mediante una flecha. (ver en ejemplo).
- e) Los atributos que pueden tomar valores nulos aparecen con un asterisco (\*).
- f) Las opciones para la integridad referencial son:
  - ❖ BC: Borrado en Cascada.
  - ❖ BD: Borrado con puesta a valor por Defecto.
  - ❖ BN: Borrado con puesta a Nulos.
  - ❖ BR: Borrado Restringido.
  - ❖ MC: Modificación en Cascada.
  - ❖ MD: Modificación con puesta a valor por Defecto.
  - ❖ MN: Modificación con puesta a Nulos.
  - ❖ MR: Modificación Restringido.



## 7. Ejemplo



**NOTA:** En todos los casos se aplicará borrado restringido y modificación en cascada.

<sup>4</sup> solo en el caso de que las relación entre proveedores y productos fuese 1:N

---

# ÍNDICE

---

1.	OBJETIVOS .....	1
2.	INTRODUCCIÓN .....	1
3.	MODELO LÓGICO DE DATOS. OBJETIVOS Y MODELOS .....	1
4.	MODELO LÓGICO DE DATOS. MODELO RELACIONAL .....	2
4.1.	CONCEPTOS DEL MODELO RELACIONAL .....	3
4.1.1.	TABLAS .....	3
4.1.2.	CLAVES .....	5
4.1.3.	INTERRELACIONES ENTRE TABLAS .....	6
4.1.4.	REGLAS O RESTRICCIONES DE INTEGRIDAD .....	7
5.	TRANSFORMACIÓN DEL MODELO E/R AL MODELO LÓGICO RELACIONAL .....	10
5.1.	TRES REGLAS BÁSICAS EN LAS TRANSFORMACIONES DEL MODELO E-R .....	11
5.2.	TRANSFORMACIÓN DE ENTIDADES .....	11
5.3.	TRANSFORMACIÓN DE ENTIDADES DÉBILES .....	12
5.4.	TRANSFORMACIÓN DE RELACIONES BINARIAS .....	13
5.4.1.	INTERRELACIÓN N:M .....	13
5.4.2.	INTERRELACIÓN 1:N .....	14
5.4.3.	INTERRELACIONES 1:1 .....	16
5.5.	TRANSFORMACIÓN DE LAS RELACIONES REFLEXIVAS .....	18
5.6.	TRANSFORMACIÓN DE RELACIONES TERNARIAS .....	20
5.7.	TRANSFORMACIÓN DE ATRIBUTOS MULTIVALUADOS .....	20
5.8.	TRANSFORMACIÓN DE RELACIONES JERÁRQUICAS .....	21
6.	NOTACIÓN .....	23
7.	EJEMPLO .....	24