

## TEMA 5.- TRANSFORMACIÓN DE EE/R A RELACIONAL

### INDICE

1.- INTRODUCCIÓN.....	1
2.- REGLAS PREPARATORIAS.....	1
3.- TRANSFORMACIÓN DE ENTIDADES.....	3
4.- TRANSFORMACIÓN DE ATRIBUTOS DE ENTIDADES.....	3
5.- TRANSFORMACIÓN DE INTERRELACIONES.....	3
6.- TRANSFORMACIÓN DE ATRIBUTOS DE INTERRELACIONES.....	7
7.- TRANSFORMACIÓN DE DEPENDENCIA EN IDENTIDAD Y EXISTENCIA.....	7
8.- TRANSFORMACIÓN DE RELACIONES JERÁRQUICAS.....	8

### 1.- INTRODUCCIÓN

Generalmente, se acepta que los modelos conceptuales (ee/r) ofrecen una representación de las complejidades de un problema de aplicación más precisa que la de los modelos lógicos (relacional). Sin embargo, en la actualidad los modelos que pueden implementarse en un sistema son los lógicos, por lo que será necesario realizar una transformación del modelo conceptual al modelo lógico.

En esta transformación, siempre se pierde semántica por lo que ésta se tendrá que reflejar al implementar dicho modelo en el sistema con las restricciones y aserciones necesarias.

Restricciones: condiciones que deben cumplir los atributos de una relación

Aserciones: las condiciones se establecen sobre los atributos de varias relaciones.

### 2.- REGLAS PREPARATORIAS

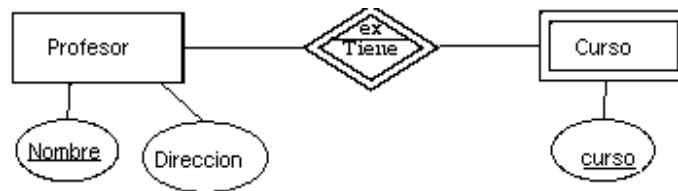
Como paso previo a la transformación del modelo ee/r al modelo relacional conviene preparar el modelo conceptual mediante la aplicación de unas reglas preparatorias.

a) Eliminación de los atributos multievaluados. Esta regla se basa en la primera forma normal



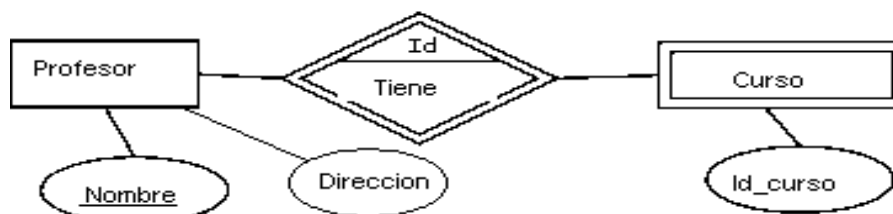
Todo atributo multievaluado se transformará en un tipo de entidad débil por existencia, el cual mantendrá una relación:

- 1:N, si el atributo es una clave alternativa en la entidad que estaba presente
- M:N, en caso contrario.



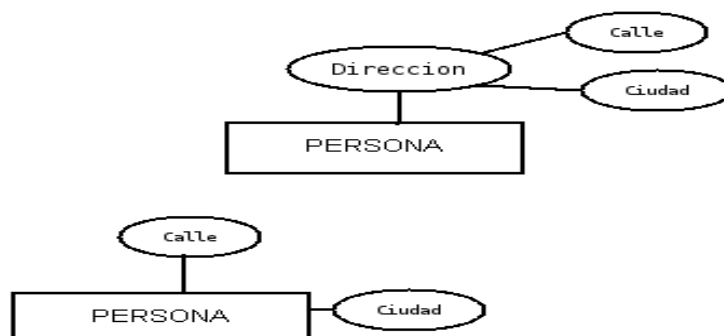
Si esta entidad débil no pudiese identificarse sin ambigüedad con el atributo que tiene, entonces debe proceder de algunas de las siguientes formas:

- El tipo de entidad débil se considerará que es débil por identificación.
- Se añadirá un nuevo atributo (externo o no al dominio del problema) que permita identificar sin ambigüedad a las ocurrencias de este tipo de entidad débil.



#### b) Eliminación de los atributos compuestos.

Todos los atributos compuestos deben ser descompuestos en atributos simples definidos en el mismo dominio.



### 3.- TRANSFORMACIÓN DE ENTIDADES

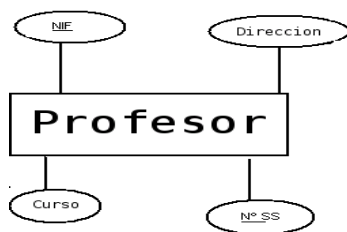
Cada tipo de entidad se convierte en una relación. La tabla se llamará igual que el tipo de entidad de donde proviene.

La transformación es directa y no hay pérdida de semántica

### 4.- TRANSFORMACIÓN DE ATRIBUTOS DE ENTIDADES.

Cada atributo de una entidad se transforma en una columna de la relación a la que ha dado lugar la entidad, teniendo en cuenta que:

- Los atributos clave pasan a ser la clave primaria de la relación
- Los atributos clave alternativa, no podrán ser nulos en el modelo relacional



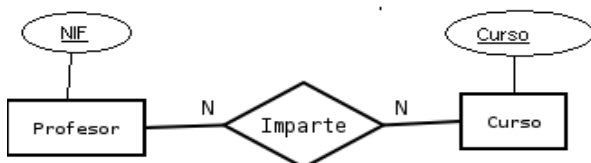
PROFESOR (NIF, N°SS<sup>No nulo</sup>, Dirección, Curso)

### 5.- TRANSFORMACIÓN DE INTERRELACIONES.

La manera de realizar la transformación en el esquema relacional variará dependiendo del tipo de la cardinalidad de la interrelación.

#### a) INTERRELACIÓN N:M.

Se transforma en una relación que tendrá como clave primaria, la concatenación de las claves de los tipos de entidad que asocia. Cada uno de los atributos que forman la clave primaria de esta relación son clave ajena respecto a cada una de las tablas donde este atributo es clave primaria.



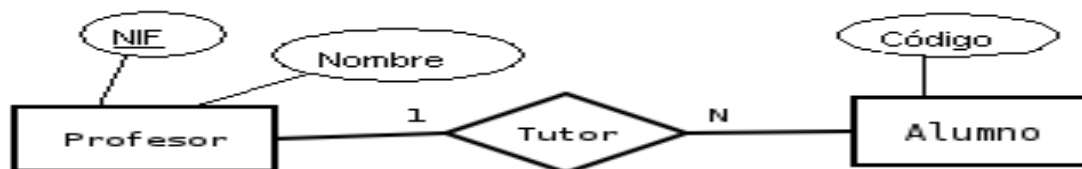
► PROFESOR (NIF, ....)  
 IMPARTE (NIF, Curso)  
 ► CURSO (Curso, ....)

Hay que estudiar que ocurre en los casos de borrado y modificación de la clave primaria referenciada (operación restringida, puesta a nulo, etc.)

Otra característica que debemos recoger son las cardinalidades máxima y mínimas de cada una de las entidades que participan en interrelación, lo que se hace mediante especificaciones de restricciones.

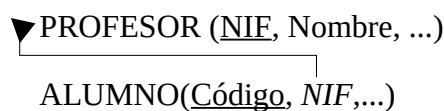
## b) INTERRELACIÓN 1:N

Existen dos soluciones para la transformación de una interrelación 1:N

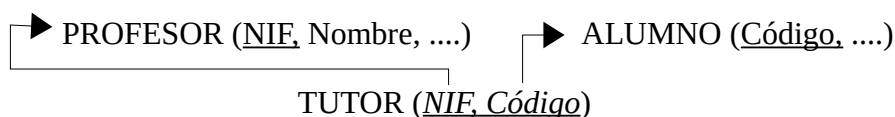


Primera. (la mas usada y la que se aconseja)

Propagar la clave del tipo entidad que tiene cardinalidad máxima a 1 a la que tiene N, desapareciendo el nombre de la relación. (



Segunda.- Transformarla en una relación. Los casos en los que es mejor tomar esta solución son:



1. Cuando el número de ocurrencias interrelacionadas de la entidad que propaga su clave es muy pequeño y cabe, por tanto, la posibilidad de que existan muchos valores nulos.
2. Cuando se prevé que dicha interrelación en un futura pueda convertirse en una de tipo N:M
3. Cuando la interrelación tiene varios atributos propios.

Hay que tener en cuenta que la propagación de la clave causa la aparición de clave ajena, con los mecanismos de borrado y actualización correspondiente.

### Control de Cardinalidades.

Si la interrelación se transforma en una relación el control de las reglas se hace mediante especificaciones de restricciones.

Si se utiliza el método de propagación de clave, las cardinalidades máximas también se realiza por medios de restricciones, pero las mínimas tienen un trato especial:

(1,1).- No admite valores nulos cuando se propaga la clave

(0,1).- Admite valores nulos cuando se propaga la clave.

(1,N), (0,N).- No se puede definir con la cláusula NOT NULL, por lo que habrá que especificar la correspondiente restricción.

#### c) INTERRELACIÓN 1:1

No hay regla fija para la transformación de este tipo de interrelación, pudiéndose crear una relación o propagar la clave correspondiente. En este último caso, la propagación de la clave puede efectuarse en ambas direcciones.

Criterios a seguir, para evitar valores nulos:

1. Cuando las entidades que se asocian poseen cardinalidades (0,1), la interrelación se debe transformar en una relación.
2. Si una de las entidades posee cardinalidad (0,1) y la otra (1,1), conviene propagar la clave de la entidad con cardinalidad (1,1) a la tabla resultante de la entidad con cardinalidad (0,1).

Con esta opción también se recoge la cardinalidad mínima 1, que en caso de propagar la clave en sentido contrario no se evitaría.

3. Si ambas entidades poseen cardinalidad (1,1), se puede propagar la clave de ellas a la tabla resultante de la otra, teniendo en cuenta en este caso los accesos mas frecuentes y prioritarios de las tablas.

## **6.- TRANSFORMACIÓN DE ATRIBUTOS DE INTERRELACIONES.**

Si la interrelación se transforma en una relación todos los atributos pasan a ser columnas de la relación.

Si la interrelación se transforma mediante propagación de clave, sus atributos migran junto con la clave a la relación correspondencia.

Advertencia: Siempre es mejor crear una nueva relación para representar la interrelación que tiene varios atributos.

## **7.- TRANSFORMACIÓN DE DEPENDENCIA EN IDENTIDAD Y EXISTENCIA.**

Las dependencias en identidad y existencias no son recogidas directamente en el modelo relacional. La transformación de la interrelación débil se realiza siguiendo los pasos habituales con la única condición de obligar a una modificación y borrado en cascada.

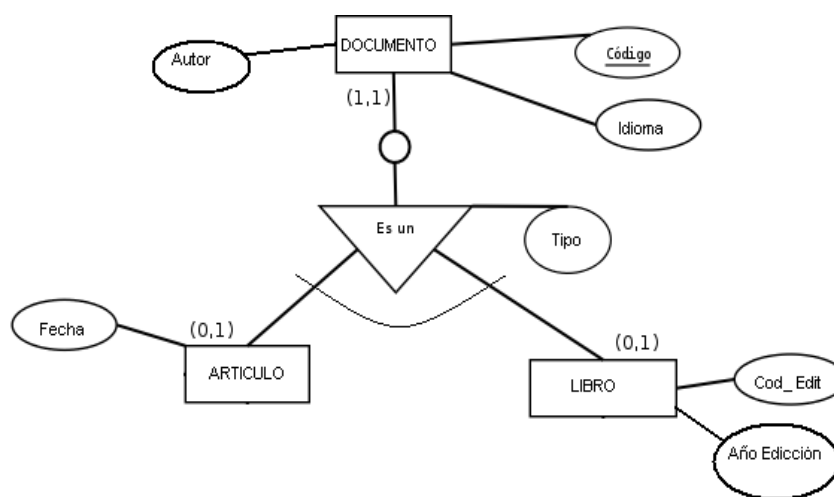
Es conveniente que la interrelación utilice el mecanismo de propagación de clave, creando una clave ajena con nulos no permitidos.

En el caso de dependencia en identidad la clave primaria de la relación de la entidad débil debe estar formada por la concatenación de las claves de las dos entidades participantes en la interrelación.

## 8.- TRANSFORMACIÓN DE RELACIONES JERÁRQUICAS.

Con respecto a los supertipos y subtipos, el modelo relacional no dispone de mecanismos fáciles de usar que permitan la representación de las relaciones jerárquicas. Hay varias soluciones de transformación, con la correspondiente pérdida de semántica dependiendo de la estrategia elegida.

Podemos elegir entre tres opciones a la hora de realizar la transformación, decidiendo en cada caso particular la más adecuada.



### OPCIÓN A

Englobar todos los atributos de la entidad supertipo y sus subtipos en una sola relación.

En general, adoptaremos esta solución cuando los subtipos se diferencien en muy pocos atributos y las interrelaciones que los asocian con el resto de las entidades del esquema sean las mismas para todos los subtipos.

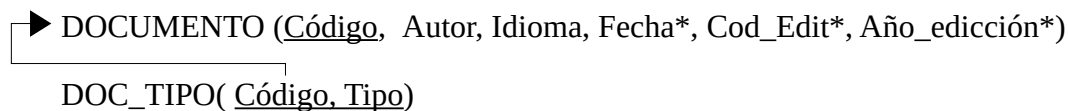
Se añade a la relación un atributo adicional, que es el atributo discriminante de la jerarquía. Este atributo podrá admitir valores nulos en el caso de que la jerarquía sea parcial y deberá declararse como no nulo si la jerarquía es total. Si existe solapamiento, el atributo discriminante constituirá un grupo repetitivo, por tanto, este atributo se debe separar en una relación aparte que asocie este atributo con la relación resultante del supertipo o formar parte de la clave.

Ejemplo1:

\* Puede tomar valor Nulo

DOCUMENTO (Código, Tipo, Autor, Idioma, Fecha\*, Cod\_Edit\*, Año\_edición\*)

Ejemplo 2:



Todo esto implica que a la hora de definir la creación de relaciones habrá que especificar las restricciones semánticas correspondientes. Por ejemplo:

CHECK ( (Tipo='Articulo' AND Año\_edicion IS NULL AND Cod\_edit IS NULL and Fecha IS NOT NULL)

OR

(Tipo='Libro' AND Año\_edicion IS NOT NULL AND Cod\_edit IS NOT NULL and Fecha IS NULL)

El acceso a una fila que refleje toda la información de una entidad es mucha más rápido

## OPCIÓN B

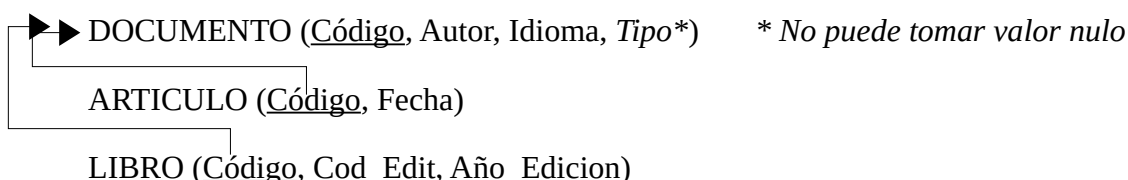
Crear una relación para el supertipo y tantas relaciones como subtipos haya con sus atributos correspondientes. El atributo discriminante se puede desestimar.

*Nota: Si la jerarquía es total y pongo el atributo discriminante con valor no nulo, reflejo la totalidad con mas fuerza.*

Esta es la solución mas adecuada cuando existen muchos atributos distintos entre los subtipos y se quiere mantener los atributos comunes a todos ellos en una relación.

Los subtipos se convierten en entidades débiles por identidad del supertipo, por lo que las opciones de modificación y de borrado siempre será en cascada.

Ejemplo:



Al igual que en el caso anterior habrá que crear las restricciones oportunas.



### OPCIÓN C

Desaparece el supertipo. Se considera relaciones distintas para cada subtipo, que contienen además de sus atributos específicos, los atributos comunes. El atributo discriminante se desestima.

Esta regla sólo se puede aplicar en jerarquías totales, puesto que implica la eliminación del supertipo y por tanto, del conjunto de ejemplares no especializados en los distintos supertipos.

Ejemplo:

LIBRO (Código, Idioma, Autor, Cod\_Edit, Año\_Edición)

ARTICULO (Código, Idioma, Autor, Fecha)

Esta solución es la mas adecuadas cuando, al igual que en el caso anterior, hay muchos atributos distintos entre los subtipos y los accesos realizados sobre los datos de los distintos subtipos siempre afectan a atributos comunes.

Con esta solución se aumenta la eficiencia entre determinadas consultas, por ejemplo, las que afectan a todos los atributos de un subtipos, pero las disminuye ante otras, las que conciernen a los atributos comunes de los distintos subtipos.

Solo es conveniente cuando hay pocos atributos comunes.

La exclusividad hay que indicarla a la hora de crear las tablas con las restricciones oportunas.