



# 3. Interpretación de diagramas entidad/relación

▼ Class	Bases de datos
👤 Column	ⓧ Xerach Casanova
🕒 Lass Edited time	@Mar 29, 2021 9:23 PM

## 1. Análisis y diseño de bases de datos

## 2. ¿Qué es el modelo E/R?

## 3. Entidades

Tipos: Fuertes y débiles

## 4. Atributos

### 4.1. Tipos de atributos

Atributos obligatorios u opcionales

Atómicos o compuestos

Atributos monovaluados o multivaluados

### 4.2. Claves

### 4.3. Atributos de una relación

## 5. Relaciones

### 5.1. Grados de una relación

### 5.2. Cardinalidad de relaciones

### 5.3. Cardinalidad de entidades

## 6. Resumen de simbología del modelo E/R

## 7. El modelo E/R extendido

### 7.1. Restricciones en las relaciones

Restricción de exclusividad

Restricción de exclusión

Restricción de inclusividad

Restricción de inclusión

### 7.2. Generalización y especialización

### 7.3. Agregación

## 8. Elaboración de diagramas E/R

### 8.1. Identificación de entidades y relaciones

### 8.2. Identificación de atributos, claves y jerarquías

## 8.3. Metodologías

#### 8.4. Redundancia en diagramas E/R

##### 8.5. Propiedades deseables de un diagrama E/R

##### 9. Primeros pasos del diagrama E/R al modelo relacional

##### 9.1. Simplificación previa de diagramas

Transformación de atributos compuestos

Transformación de atributos multivaluados

Transformación a relaciones jerárquicas

Transformación de relaciones cíclicas

Transformación de relaciones ternarias

#### 10. Paso del diagrama E/R al modelo relacional

Relaciones 1/1 de E/R a esquema relacional

Relaciones reflexivas a esquema relacional

Jerarquías de E/R a esquema relacional

Relaciones de muchos a muchos a esquema relacional

Relaciones N-Arias a esquema relacional

#### 11. Normalización de modelos relacionales

##### 11.1. Tipos de dependencias

Dependencia funcional

Dependencia funcional completa

Dependencia transitiva

##### 11.2. Formas normales

1ª forma normal

2ª forma normal

3ª forma normal

Forma normal de Boyce Codd

Otras formas normales

#### Mapa Conceptual

El objetivo principal de la arquitectura de los SGBD es separar los programas de aplicación de bases de datos física, proponiendo **tres niveles de abstracción: nivel interno o físico, nivel lógico o conceptual y nivel externo o de visión del usuario.**

## **1. Análisis y diseño de bases de datos**

El nivel lógico o conceptual describe la estructura completa de la bbdd a través del esquema conceptual, así se representa la información de una manera independiente al SGBD.

En el desarrollo de BBDD se distinguen dos fases: **análisis y diseño.**

### **Fase de análisis**

- **Análisis de entidades:** localizar y definir las entidades y sus atributos.
- **Análisis de relaciones:** definir relaciones entre entidades
- **Obtención de esquema conceptual** a través del modelo E-R.
- **Fusión de vistas:** Un único esquema con todos los esquemas existentes en función de diferentes vistas de perfil de usuario.
- **Aplicación de enfoque de datos relacional.**

### **Fase de diseño**

- **Diseño de tablas**
- **Normalización**
- **Aplicación de rediseño** si fuese necesario (volver al análisis si fuese necesario).
- **Diseño de transacciones:** localización del conjunto de operaciones o transacciones que operarán sobre el esquema conceptual
- **Diseño de sendas de acceso:** se formalizan los métodos de acceso dentro de la estructura de datos.

Una correcta fase de análisis determina el éxito de la bbdd. Saltarse el esquema conceptual implica pérdida de info con respecto al problema real y debe reflejar los aspectos relevantes del mundo real que se va a modelar.

El modelo entidad/relación extendido (ERE) es el más aceptado y evoluciona del original modelo E/R.

## **2. ¿Qué es el modelo E/R?**

Es una herramienta que representa conceptualmente los problemas del mundo real y su objetivo es facilitar el diseño de la BBDD a través de un esquema que representa toda su estructura.

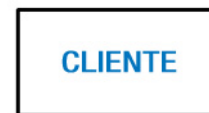
El esquema tiene descripciones textuales de la realidad que establecen los requerimientos del sistema, buscando ser lo más fiel posible al comportamiento del mundo real.

Al representar los datos semánticamente no está orientado a ningún sistema físico o informático. Se puede incluso utilizar para describir procesos de producción, estructuras de empresa, etc.

Este sistema utiliza símbolos y expresiones determinados, así como objetos básicos llamados entidades y relaciones entre esos objetos.

### 3. Entidades

En definitiva, una entidad es un objeto real o abstracto con características diferenciadoras capaces de hacerse distinguir de otros objetos, de la que además se desea guardar información.



Un conjunto de entidades es un grupo de entidades con las mismas características o propiedades. Por ejemplo: conjunto de clientes, conjunto de ríos existentes en una determinada zona, etc...

En el modelo E/R, una entidad se representa mediante un rectángulo con el nombre de la entidad dentro.

### Tipos: Fuertes y débiles

A su vez, las entidades se clasifican en:

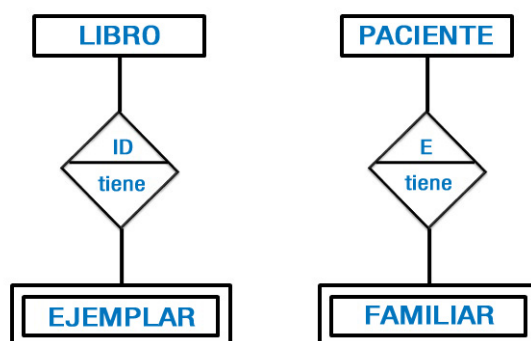
- **Fuertes o regulares:** su existencia no depende de otras entidades y se representan como se ha descrito anteriormente. Ejemplo: Doctor, Paciente.
- **Entidades débiles:** es un tipo de entidad cuyas propiedades o atributos no la identifican completamente, solo de forma parcial, por tanto debe participar en una relación con otra entidad que ayude a identificarla.



Estas entidades a su vez presentan dos tipos de dependencia.

- **Dependencia de existencia.** Si desaparece una instancia de entidad fuerte desaparecen las entidades débiles. Se representa con una E en el interior de una relación débil.
- **Dependencia de identificación.** También existe dependencia de la entidad fuerte, pero además no se puede identificar por sí misma y necesita de la ID de la entidad fuerte para poder identificarse. Se representa con "ID" en el interior de la relación débil. Por ejemplo. Existen una entidad fuerte llamada "Almacén" y otra débil llamada "Pedidos". Queremos tener una lista de pedidos numerados en cada almacén y puede existir un pedido de idéntica numeración en dos almacenes (por ejemplo 20340). La

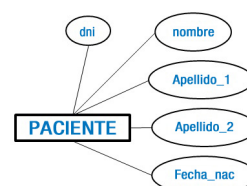
identificación única de un pedido será la combinación de la ID de Almacén con la ID del pedido (1,20340 y 2,20340).



## 4. Atributos

Los atributos son cada una de las propiedades o características que tiene un tipo de entidad o un tipo de relación.

En el modelo E/R se identifican mediante el nombre del atributo rodeado de una elipse y se conecta con la entidad con una línea recta. Cada atributo debe tener nombre único.



Los valores permitidos dentro de cada atributo son los dominios. Varios atributos pueden estar definidos dentro del mismo dominio. Por ejemplo "nombre, apellido\_1 y apellido\_2. Los dominios pueden ser amplios, pero se deben establecer límites en el SGBD para garantizar la integridad de datos.

### 4.1. Tipos de atributos

#### Atributos obligatorios u opcionales

- **Atributo obligatorio.** Siempre ha de estar definido en cada ocurrencia de una entidad o relación. Una clave o llave es un atributo obligatorio.
- **Atributo opcional.** Puede ser definido o no en cada ocurrencia de la entidad.

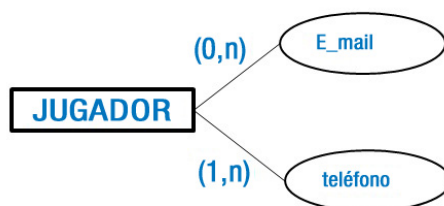
#### Atómicos o compuestos

- **Atributo atómico o simple.** Es un atributo que no puede dividirse en partes más pequeñas. Por ejemplo el atributo DNI.
- **Atributo compuesto.** es un atributo que puede ser dividido en subpartes que constituyen otros atributos con significado propio. Ejemplo: Dirección (puede estar compuesto por calle, número y localidad).

## Atributos monovaluados o multivaluados

- **Atributo monovaluado:** es aquel que tiene un único valor para cada ocurrencia. Por ejemplo: DNI.
- **Atributo Multivaluado:** puede tomar diferentes valores para cada ocurrencia de entidad. Por ejemplo, el e-mail de un empleado puede tomar varios valores para quien posea varias cuentas de correo. Hay que tener en cuenta:
  - **Cardinalidad de un atributo:** Indica el número mínimo y máximo de valores que puede tomar para cada ejemplar de la entidad o relación a la que pertenece.
  - **Cardinalidad mínima.** Indica la cantidad mínima (casi siempre 0 y 1).
  - **Cardinalidad máxima.** Indica la cantidad de máxima (entre 1 y n).

En el ejemplo, el atributo e-mail puede tener de cero a n valores y el teléfono puede tener de 1 a n valores



- **Atributos derivados, calculados o almacenados:** Son obtenidos del valor o valores de otros atributos. Por ejemplo, el atributo edad puede ser calculado a partir del atributo fecha de nacimiento. No conviene almacenar edad, ya que es un elemento que varía con el tiempo.

## 4.2. Claves

Las claves son un atributo especial obligatorio.

Los valores de los atributos de una entidad deben permitir identificar unívocamente a la entidad. Es decir, no puede haber dos ocurrencias exactamente iguales en una entidad.

- **Súperclave (superllave).** Es un conjunto de atributos que permite identificar de forma única a cada ocurrencia de la entidad. Puede tener atributos no obligatorios (no identificarían por si solos a la ocurrencia).
- **Clave candidata.** Puede ser un atributo único o una superclave siempre que no se pueda elegir un subconjunto que pueda ser superclave. una clave candidata debe ser única (no puede haber dos tuplas con los mismos valores para esos atributos) e irreducible (si se elimina algún atributo deja de ser única). Para elegir las claves candidatas debemos tener en cuenta lo siguiente:
  - Sus valores deben ser distintos de nulo.
  - La memoria que ocupen debe ser la menor posible.
  - Codificación sencilla.
  - El contenido de los valores no debe variar.
- **Clave primaria (Primary Key).** De todas las claves candidatas, el diseñador debe escoger una, que será la clave primaria. Es un atributo o conjunto de ellos que toman valores únicos y distintos para cada ocurrencia. No puede tener valores nulos. Entre varias claves candidatas se debe elegir la clave primaria atendiendo a los siguientes criterios:
  - La de menos longitud.
  - Simples antes que compuestas.
  - Numéricas sobre no numéricas.
  - Codificadas sobre no codificadas.
  - Ámbito local sobre ámbito más general.

Una vez elegida la clave primaria, las restantes claves candidatas son denominadas claves alternativas o secundarias.

La representación de modelo E/R de las claves primarias se realiza de dos formas:

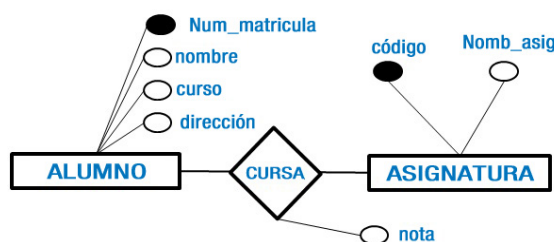
- En representación con elipses se subraya los atributos que formen la clave primaria

- En representación con círculo se utilizan círculos negros en los que formen la clave primaria



### 4.3. Atributos de una relación

Una relación puede tener atributos que la describan y se representa de la misma manera que cualquier otro atributo, pero con la línea recta uniéndolo con la propia relación.



## 5. Relaciones

La relación o interrelación es un elemento del modelo E/R que permite relacionar datos entre sí. Es una asociación entre diferentes entidades. No puede aparecer la misma ocurrencia relacionada dos veces en la misma relación.



La representación gráfica es un rombo con un verbo en su interior y líneas rectas que pueden o no acabar en punta de flecha uniendo las distintas relaciones.

### 5.1. Grados de una relación



Es el número de entidades que participan en una relación.

- **Relación Unaria o grado 1.** Participa una única entidad (reflexivas o recursivas).
- **Relación Binaria o grado 2.** Participan 2 entidades. En general se busca que el esquema conceptual de la bbdd tenga este tipo de relaciones.
- **Relación Ternaria o grado 3.** Participan 3 entidades al mismo tiempo.
- **Relación N-aria o grado n.** Involucra a un número n de entidades. Deben ser simplificadas y hacerlas de menor grado.
- **Relación dobe.** dos entidades se relacionan a través de dos relaciones. Complejidad de manejo.

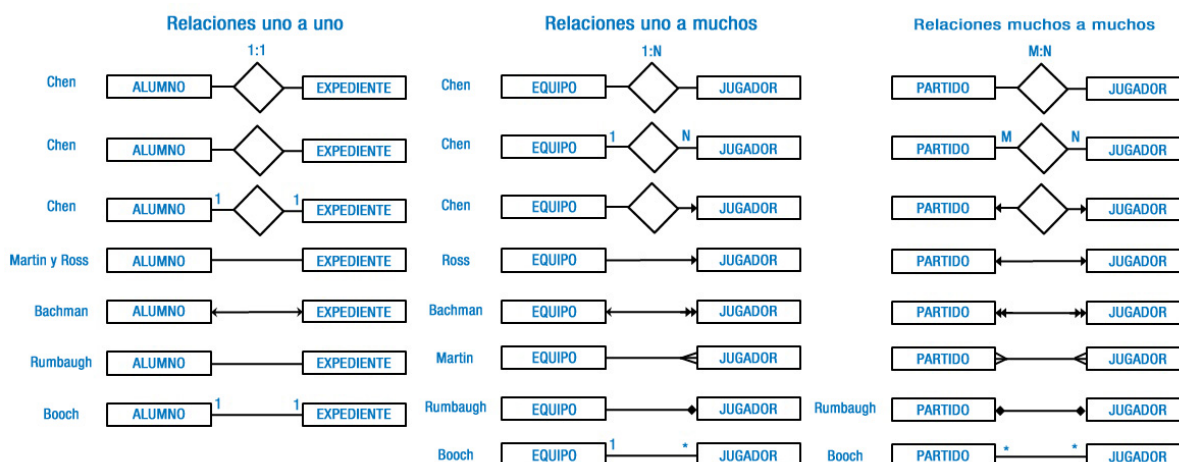
## 5.2. Cardinalidad de relaciones

Es el número máximo de ocurrencias de cada entidad que pueden intervenir en una ocurrencia de relación. Viene expresada para representaciones entre dos entidades y pueden ser:

Dadas las entidades A y B...

- **Relaciones de uno a uno (1:1).** una ocurrencia de A se relaciona con una sola ocurrencia de B y viceversa. Ejemplo: un alumno solo puede tener un expediente y un expediente puede pertenecer a un solo alumno.
- **Relaciones uno a muchos (1:N).** una ocurrencia de A se relaciona con muchas ocurrencias de B y una ocurrencia de B solo se relaciona con una ocurrencia de A. Ejemplo: un docente puede tener varias ocurrencias de la entidad asignatura, pero la entidad asignatura solo tendrá una ocurrencia ligada con la entidad docente.
- **Relaciones muchos a uno (N:1).** una ocurrencia de A solo se relaciona con una ocurrencia de B, pero una ocurrencia de B puede relacionarse con muchas ocurrencias de A. Ejemplo: un jugador pertenece a un único equipo, pero a un equipo pueden pertenecer muchos jugadores.
- **Relaciones muchos a muchos (N:N).** Un ejemplar de A puede relacionarse con varias ocurrencias de B y viceversa. Ejemplo: un alumno se matricula en varias asignaturas y en una asignatura pueden estar matriculados varios alumnos.

La representación de las cardinalidad de las relaciones se puede hacer de varias maneras en los esquemas E/R:



### 5.3. Cardinalidad de entidades

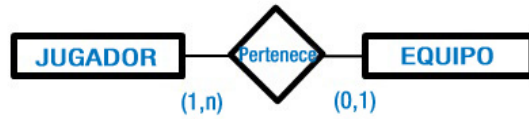
Es el número mínimo y máximo de correspondencias en las que puede tomar parte cada ocurrencia de la entidad.

Dadas las entidades A y B, la participación de A en una relación sería obligatoria (total) si la existencia de cada una de sus ocurrencias necesita como mínimo de una ocurrencia de B. En caso contrario la participación es opcional. La representación gráfica es de una etiqueta tipo (0,1), (1,1), (0,N) o (1,N) indicando los parámetros de cardinalidad mínima y cardinalidad máxima.

- **Cardinalidad mínima.** El número mínimo de asociaciones en las que aparecerá cada ocurrencia de la entidad. Es cero si es opcional o uno si es obligatoria (aunque pueda tener más de 1).
- **Cardinalidad máxima.** El número máximo de relaciones en las que aparecerá cada ocurrencia de entidad. Puede ser 1, otro valor concreto o N).

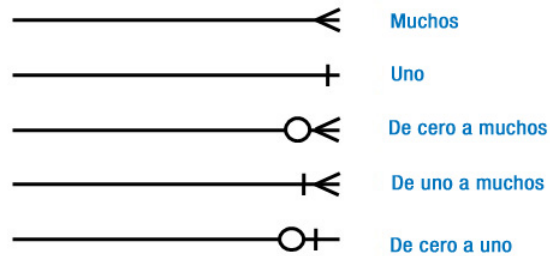
Ejemplo:

Un jugador, como mínimo puede pertenecer a ningún equipo y como máximo a 1 (0,1). En cambio, a un equipo pertenece como mínimo un jugador y como máximo varios (1,N). **Se representa colocando la etiqueta junto a la entidad con la que se relaciona, es decir, al lado opuesto de la relación.**



Otras maneras de representar la cardinalidad entre entidades:

Notación alternativa para representar  
cardinalidad de entidades



## 6. Resumen de simbología del modelo E/R

### Notaciones del modelo Entidad/Relación

	Entidad	
	Entidad Débil	
	Atributo simple o atómico	
	Atributo compuesto	
	Atributo multivaluado	 

### Notaciones del modelo Entidad/Relación

	Relación	
	Relación uno a uno	
	Relación uno a muchos	
	Relación muchos a muchos	
	Cardinalidad de entidad	

## Notaciones del modelo Entidad/Relación

	Atributo Derivado	
	Atributo opcional	
	Clave primaria	
	Clave alternativa	

## 7. El modelo E/R extendido

En el modelo E/R extendido se incorporan nuevas extensiones que permiten mejorar la capacidad para representar circunstancias especiales que el modelo E/R tradicional no puede representar.

### 7.1. Restricciones en las relaciones

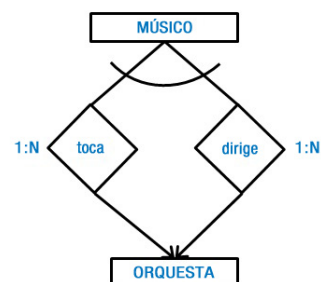
Son restricciones sobre la relación de los ejemplares.

#### Restricción de exclusividad

Existe si una entidad participa en dos o más relaciones, pero las ocurrencias de esa entidad solo pueden participar en una de las relaciones y no en varias a la vez.

Ejemplo: un músico puede dirigir una orquesta o tocar en ella, pero no puede hacer las dos cosas a la vez. Existen dos entidades (músico y orquesta), existen dos relaciones (dirigir y tocar) y existe relación de exclusividad (o dirige o toca).

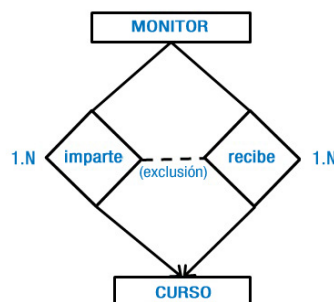
La representación es mediante un arco que englobe todas las relaciones.



## Restricción de exclusión

Esta restricción se produce cuando las ocurrencias de las entidades solo pueden asociarse utilizando una única relación. En el ejemplo del músico y la orquesta, podríamos modificar su funcionamiento diciendo que un músico puede dirigir orquestas y también puede tocar en ellas, pudiendo usar las dos relaciones a la vez, pero lo que no puede hacer es dirigir y tocar en la MISMA ORQUESTA.

Otro ejemplo sería un monitor que puede impartir diferentes cursos orientados a monitores y además también puede recibirlos. Pero lo que no puede hacer es recibir el mismo curso que imparte.



Se representa con una línea discontinua entre las dos relaciones.

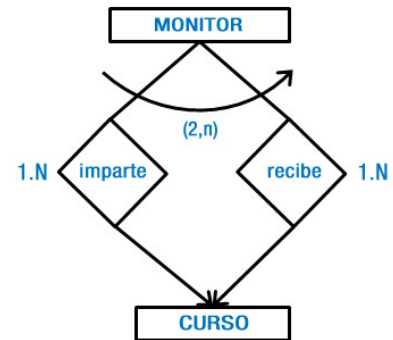
## Restricción de inclusividad

Dos ocurrencias de entidad que se asocian a través de una relación no pueden hacerlo hasta que se hayan asociado a través de otra relación.

Por ejemplo. Imaginemos que el requisito para que un monitor imparta un curso en concreto, primero debe haber recibido otros dos cursos. El curso que imparte no tiene por qué ser el mismo que los que previamente ha recibido.

Aplicando esta restricción estaremos indicando que cualquier ocurrencia de una entidad que

participa en una de las relaciones, debe participar obligatoriamente en la otra.



Se representa con un arco acabado en flecha que partirá desde una relación hasta la relación que ha de cumplirse y entre paréntesis pondremos el número mínimo y máximo de ocurrencias de dicha restricción.

## Restricción de inclusión

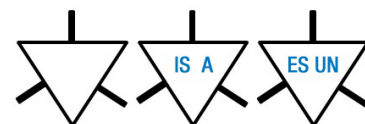
Es una restricción más fuerte que la de inclusividad, haciendo que no se pueda cumplir una relación de las ocurrencias de dos entidades hasta que no se haya cumplido otra relación entre esas mismas ocurrencias.

En el caso anterior, un monitor no podrá impartir un curso en concreto si previamente no lo ha recibido. De modo que toda la ocurrencia de la entidad monitor que esté asociada a una ocurrencia de la entidad curso a través de la relación imparte, ha de estar unida a la misma ocurrencia de la entidad curso a través de la relación recibe.

Se representa con una flecha discontinua desde una relación que depende de que la otra se haya cumplido, hasta la relación que debe cumplirse.

## 7.2. Generalización y especialización

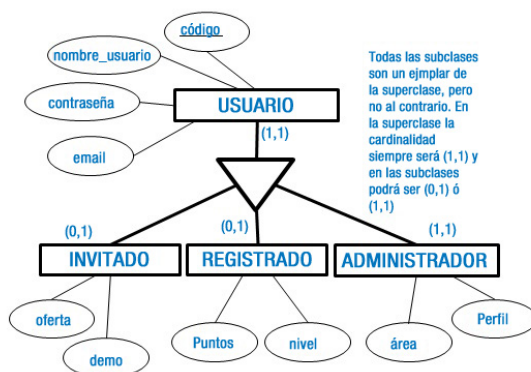
Son nuevos tipos de relaciones que permiten modelar de una manera más fiel, reciben el nombre de jerarquías y se basan en conceptos de generalización especialización y herencia.



Son entidades que poseen características comunes, permitiendo crear una entidad de nivel más alto que englobe dichas características, que a su vez se divida en entidades que tienen características diferenciadoras. Es un proceso de refinamiento ascendente/descendente y expresan entidades de nivel superior (superclase o supertipo) que engloban a entidades de nivel inferior (subclase o subtipo).

Esto hace que podamos realizar una especialización de una superclase en subclases y a su vez una generalización de subclases en superclases.

La jerarquía se caracteriza por la herencia, los atributos de superclase los hereda la subclase y si una superclase interviene en una relación también lo harán las subclases.



Se representan mediante un triángulo invertido con la notación IS A (es un) dentro de él.

Una generalización/especialización puede tener las siguientes restricciones:

- **Totalidad:** si todo ejemplar de la superclase pertenece a alguna de las subclases.
- **Parcialidad:** si no todos los ejemplares de la superclases pertenecen a alguna de las subclases.
- **Solapamiento:** si un mismo ejemplar de la superclase puede pertenecer a más de una subclase.
- **Exclusividad:** si un mismo ejemplar de la superclase pertenece a una sola subclase.

## 7.3. Agregación

En el modelo de entidad relación tradicional, no se pueden representar relaciones entre relaciones. En este caso la solución pasa por convertir las entidades y su relación en una sola entidad de nivel más alto, siendo utilizada para expresar otra relación con otra entidad (o grupos de entidades+relación).

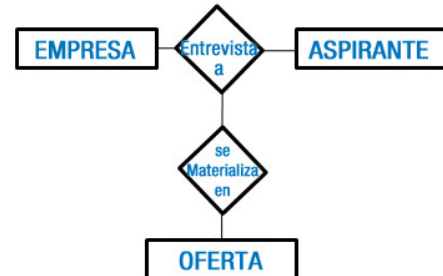
Por ejemplo. Una empresa selección de personal realiza entrevistas a diferentes aspirantes. Estas entrevistas pueden derivar o no en oferta de



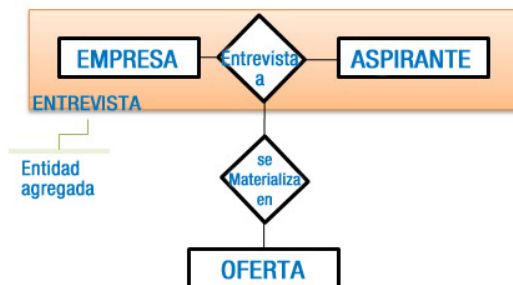
empleo.



**Solución 1:** Errónea, ya que estaríamos representando que, por cada entrevista realizada por una empresa a un aspirante, se genera una oferta de empleo



**Solución 2:** Errónea, porque en el modelo E/R no pueden establecerse relaciones entre varias relaciones



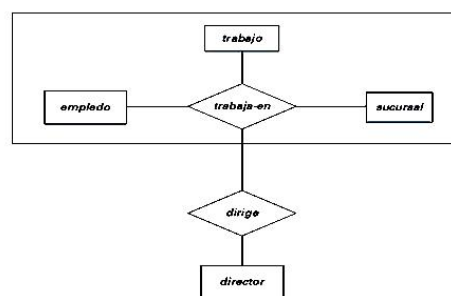
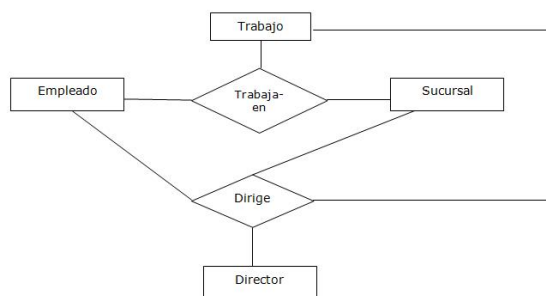
**Solución 3:** En el modelo E/R Extendido, puede crearse una entidad agregada llamada ENTREVISTA, compuesta por la relación "Entrevista a" que existe entre EMPRESA y ASPIRANTE. Entre esta nueva entidad y OFERTA si puede establecerse una relación "se materializa en"

Otro ejemplo.

Supongamos una entidad Director, la cual dirige a la entidad Empleado, la entidad Trabajo y la entidad Sucursal.

El conjunto de Empleado Trabajo y Sucursal están unidas por la relación trabajo. Este conjunto de entidades se puede relacionar con director formando una relación cuaternaria entre empleado, sucursal, trabajo y director.

Sin embargo, utilizando la agregación se crea una relación binaria dirige entre trabaja-en y director para representar quien dirige las tareas



## 8. Elaboración de diagramas E/R

Saltase el proceso de fase de diseño de la base de datos y el diagrama E/R supone la pérdida de información de la base de datos. La práctica de estos diagramas es fundamental, a veces hay que retocarlos e incluso rehacerlos.

### 8.1. Identificación de entidades y relaciones

Las etapas para la creación de diagramas E/R son:

1. **Identificación de entidades:** es un proceso intuitivo, se analiza la especificación de requerimientos en busca de nombres o sustantivos, si estos nombres son objetos importantes del problema seguramente serán entidades. También se ha de identificar que nombres referidos a características, cualidades o propiedades no serán entidades.

Otra forma de identificarlos es localizar los objetos o elementos que existen por sí mismos, acerca de los cuales se quiere guardar información, esa información serán atributos. En ocasiones también se pueden identificar elementos a través de los atributos que lo componen.

Para identificar entidades se deben cumplir 3 reglas:

- Existencia propia.
- Cada ejemplar de una entidad debe poder ser diferenciado del resto.
- Todos los ejemplares de una entidad deben tener las mismas propiedades.

El número de entidades obtenidas debe ser manejable y **se irán identificando con nombres, en mayúscula, singular y representativos de su función.**

2. **Identificación de relaciones.** Utilizaremos verbos o expresiones verbales que conecten las entidades previamente identificadas entre ellas. Se debe aplicar la lógica para no repetir las relaciones. La gran mayoría de las veces las relaciones se establecen entre dos entidades, pero pueden existir relaciones recursivas o unarias y relaciones entre más entidades.

El nombre de la relación debe ser preferiblemente un verbo en minúsculas y representativo del significado o acción de la relación. Si el identificador está compuesto por varias palabras se utilizará guión bajo para unirlos.

El siguiente paso será representar la cardinalidad mínima y máxima de las entidades participantes en cada relación y el tipo de correspondencia (1 a 1, 1 a muchos, o muchos a muchos).

Si hemos encontrado alguna relación recursiva, reflexiva o unaria, representaremos en el esquema los roles desempeñados por la entidad en dicha relación.

## 8.2. Identificación de atributos, claves y jerarquías

1. **Identificación de atributos.** Sobre el documento de especificación de requerimientos buscaremos nombres relativos a características, propiedades, identificadores o cualidades de las entidades o relaciones. Puede que no todos los atributos queden reflejados en el documento de especificación de requerimientos.

Hay que tener en cuenta si los atributos localizados son simples, compuestos, derivados o calculados y si algún atributo o conjunto de ellos se repite en varias entidades. Si esto último sucede podemos plantear establecer una jerarquía de especialización.

Se les debe asignar un nombre, preferiblemente en minúsculas, representativo de su contenido o función. Además se debe recopilar la siguiente información:

- Nombre y descripción.
  - Atributos simples que lo componen (si es compuesto).
  - Método de cálculo si es atributo derivado.
2. **Identificación de claves:** se establecen una o varias claves candidatas, se escoge una de ellas como clave primaria, formada por uno o varios atributos que identifican de manera unívoca a cada ocurrencia. Este proceso permitirá identificar si la entidad es fuerte (al menos una clave candidata) o débil (ninguna).
  3. **Determinación de jerarquías.** Además de identificar entidades con características comunes generalizadas en una entidad de nivel superior, también se ha de expresar en el esquema las particularidades de diferentes ejemplares de un tipo de entidad, creando subclases de superclases analizando con detenimiento el documento de especificación de requerimientos.

## 8.3. Metodologías

Son estrategias disponibles para la elaboración de esquemas conceptuales:

- **Metodología descendente (top-down).** A partir de un esquema general se descompone en niveles, cada uno con mayor número de detalles. Se parte de objetos muy abstractos y se van refinando hasta llegar al esquema final
- **Metodología ascendente (bottom-up).** Se parte desde el nivel más bajo, los atributos, se agrupan en entidades y después se crean las relaciones y jerarquías.
- **Metodología dentro-fuera (inside-out).** Se comienza el esquema en una parte del papel y se van completando con entidades y relaciones hasta ocupar todo el documento.
- **Metodología mixta.** Para problemas complejos, se dividen los requerimientos en subconjuntos y se van analizando independientemente. Se crea un esquema estructura en el que se interconectan los subconjuntos creados. Se utiliza la técnica descendente para dividir los requerimientos y en cada subconjunto se usará la ascendente.

Se utiliza la metodología que más útil resulte aplicar e incluso, alguna nueva a partir de las ya existentes.

## 8.4. Redundancia en diagramas E/R

La redundancia de datos es el almacenamiento de los mismos datos varias veces en diferentes lugares. Generan problemas como:

- Aumento de la carga de trabajo (grabación o actualización de datos varias veces).
- Gasto extra de espacio de almacenamiento.
- Inconsistencia. Los datos que están repetidos pueden contener distintos valores porque se han actualizado en unos sitios y en otros no.

Nuestros diagramas deben controlar la redundancia y para ello debemos analizar el esquema y valorar que puede estar incorporando redundancia. Para ello debemos buscar:

- **Atributos redundantes,** cuyo contenido se calcula en función de otros. Un atributo derivado puede ser origen de redundancia.

- **Ciclos.** Varias entidades unidas circularmente a través de varias relaciones. En caso de existir debemos tener en cuenta:
  - Que el significado de las relaciones que componen el ciclo sea el mismo.
  - Que si eliminamos la relación redundante, el significado del resto de relaciones sea el mismo.
  - Que si la relación eliminada tenía atributos asociados estos puedan ser asignados a alguna entidad participante del esquema sin perder significado.

No siempre en la existencia de un ciclo estamos ante una redundancia.

No toda redundancia es perjudicial y a veces es conveniente de una manera controlada. Por ejemplo ante la existencia de atributos derivados de cálculo complejo que ralenticen el funcionamiento de la BBDD.

## 8.5. Propiedades deseables de un diagrama E/R

Debemos intentar materializar la gran mayoría de propiedades para que nuestros diagramas o esquemas tengan mejor calidad. Estas son:

- **Compleitud.** Que todos los requerimientos estén representados en dicho diagrama y que cada representación tenga su equivalente en requerimientos.
- **Corrección:** Será correcto si emplean de manera adecuada todos los elementos del modelo E/R.
  - **Corrección sintáctica:** Que no se produzcan representaciones erróneas en el diagrama.
  - **Corrección semántica:** Que signifique lo que realmente está estipulado en requerimientos. Estos errores pueden ser: usar atributos en lugar de una entidad, una entidad en lugar de una relación, mismo identificador para dos entidades o dos relaciones, indicar mal u omitir cardinalidades...
- **Minimalidad:** es mínimo si se puede verificar que al eliminar algún concepto del diagrama se pierde información. Si es redundante no será mínimo.

- **Sencillez:** se representan los requerimientos de manera fácil de comprender.
- **Legibilidad:** Se puede interpretar fácilmente (aspectos estéticos del diagrama).
- **Escalabilidad:** Si es capaz de incorporar cambios derivados de nuevos requerimientos.

## 9. Primeros pasos del diagrama E/R al modelo relacional

El diagrama E/R permite una gran independencia en cuestiones relativas a la implementación física de la BBDD, la elección del SGBD, aplicaciones, lenguajes de programación o hardware no afectan hasta este momento.

Una vez hemos revisado, modificado y verificado que cumplimos los requerimientos del problema a modelar es hora de pasar al diseño lógico de la bbdd.

Consiste en construir un esquema lógico de la información relativo al problema, basado en un modelo de base de datos concreto, utilizando elementos y características del modelo de datos de la SGBD. Este modelo puede ser: modelo en red, jerárquico y los que se usan hoy en día: relacional y orientado a objetos.

Esta transformación parte del esquema obtenido en la fase de diseño conceptual y requiere aplicar algunas reglas que garanticen equivalencia entre esquema conceptual y lógico.

El siguiente pasos es la normalización, que consiste en diseñar de forma correcta la estructura lógica de los datos, usando un conjunto de técnicas que permiten validar esquemas lógicos basados en modelo relacional.



### 9.1. Simplificación previa de diagramas

Es un conjunto de procedimientos para aplicar a los diagramas E/R, obteniendo una transformación al modelo lógico basado en el modelo relacional, correcto y casi automático:

- Transformación de relaciones n-arias en binarias.
- Eliminación de relaciones cíclicas.

- Reducción a relaciones jerárquicas (uno a muchos).
- Conversión de entidades débiles en fuertes.

## Transformación de atributos compuestos

Los atributos compuestos han de ser descompuestos en atributos simples, ya que el modelo relacional no los permite.

## Transformación de atributos multivaluados

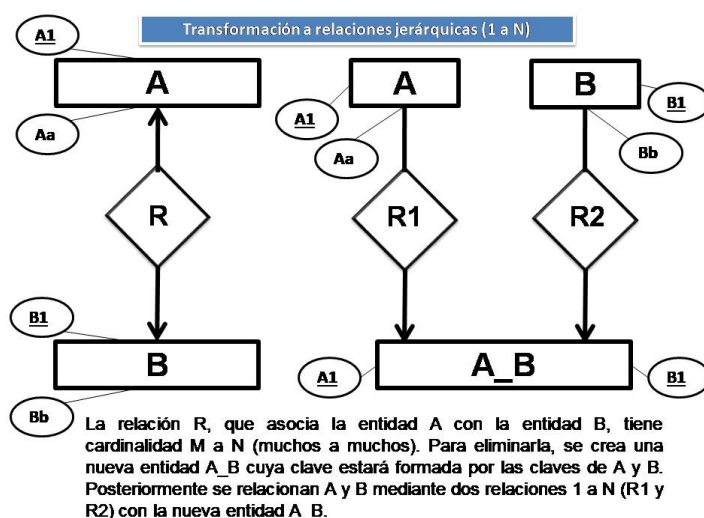
Los atributos multivaluados se convierten en entidad relacionada con la entidad que le procede. El modelo relacional no los permite.

Esta nueva entidad tendrá un único atributo, a no ser que no sea posible que dicho atributo funcione como clave primaria. En ese caso se trata de una entidad débil y debemos ajustar las claves primarias.

## Transformación a relaciones jerárquicas

Transformamos las relaciones de cardinalidad muchos a muchos (M/N) a relaciones con cardinalidad uno a muchos. La manera de hacerlo es creando una entidad intermedia con relación 1/N entre las entidades existentes y la nueva entidad. Los atributos pertenecientes a la relación N-Aria pasan a ser parte de la nueva entidad..

Por ejemplo, ante dos entidades A y B relacionadas con cardinalidad N/M, crearemos una tercera entidad llamada A\_B, cuya clave estará formada por las claves foráneas de A y B. Posteriormente se relacionan A y B con la nueva entidad A\_B mediante relaciones 1/N. Los atributos que formaban parte de la relación anterior.



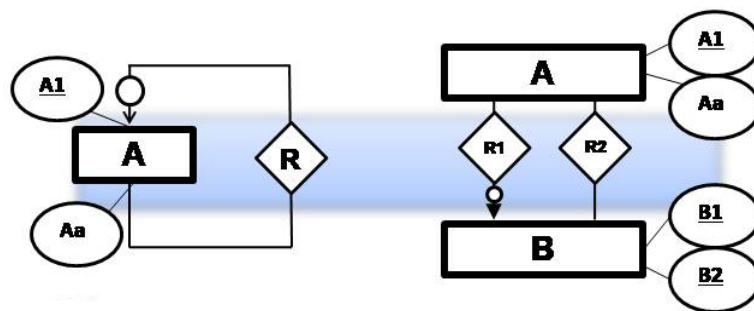
## Transformación de relaciones cíclicas

Normalmente, para eliminar transformaciones cíclicas, se crea una nueva entidad, cuya clave estará formada por dos atributos, que contienen las claves de las ocurrencias relacionadas y se crean dos relaciones, cuya cardinalidad depende de la cardinalidad que tenían al principio. Este tipo de conversiones se entiende mejor a la hora de pasarlas al modelo relacional, en el siguiente apartado.

### Reflexivas N/M:



### Reflexivas 0/N:



### Reflexivas 1/1:



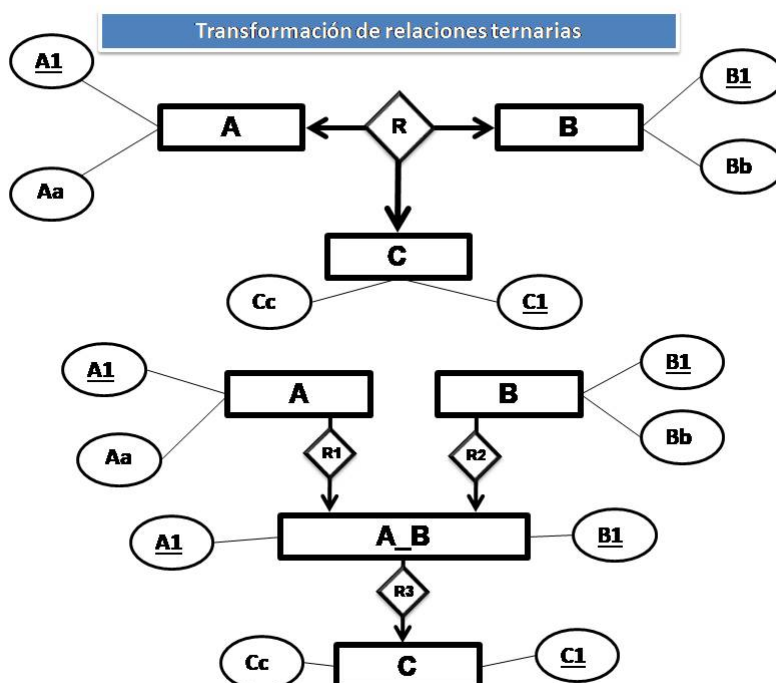
## Transformación de relaciones ternarias



Una relación ternaria puede considerarse como una relación binaria a la que se le asocia una entidad. La solución es convertir la relación en una nueva entidad que asocie dos de las entidades y a su vez, crear una nueva relación entre esta nueva entidad y la tercera.

Por ejemplo, entre las entidades A, B y C hay una relación. Esta relación la transformaremos en una entidad llamada A\_B, que contendrá las claves de A y B como primarias y las entidades A y B tendrán una relación 1/N con la entidad A\_B.

A su vez, esta nueva entidad A\_B tendrá una relación 1/N con la entidad C.



## 10. Paso del diagrama E/R al modelo relacional

Después de la simplificación habremos logrado disponer de un esquema conceptual modificado (ECM), en el cual solo existen entidades fuertes con sus atributos relaciones jerárquicas. Ahora debemos tener en cuenta las siguientes cuestiones:

- Toda entidad se transforma en tabla.
- Todo atributo se transforma en columna de tabla.

- El atributo clave de la entidad se convierte en clave primaria y se representa subrayado en tabla.
- Cada entidad débil genera una tabla que incluirá sus atributos, además de los atributos que son clave primaria de la entidad fuerte con la que esté relacionada. Estos atributos son clave foránea referenciando a la entidad fuerte y por último se escoge una clave primaria de la tabla creada.
- Las relaciones uno a uno pueden generar una nueva tabla o propagar la clave en función de la cardinalidad de las entidades.

## Relaciones 1/1 de E/R a esquema relacional

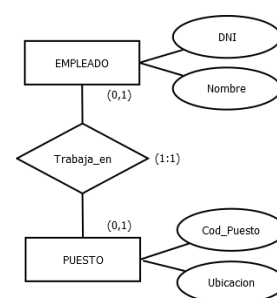
- Si ambas entidades poseen cardinalidades 0,1, la relación se convierte en tabla.

Ejemplo:

EMPLEADO (DNI, Nombre)

EMPLEADO\_PUESTO (DNI, Cod\_Puesto)

PUESTO (Cod\_Puesto, Ubicacion)

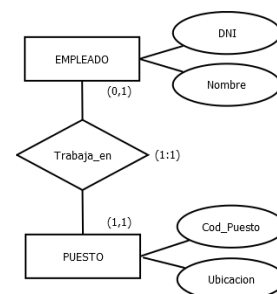


- Si una entidad posee cardinalidad 0,1 y la otra 1,1, se propaga la clave de la entidad 1,1 a la entidad 0,1

Ejemplo:

EMPLEADO (DNI, Nombre)

PUESTO (Cod\_puesto, Ubicacion, DNI)



- Si ambas entidades tienen cardinalidad 1,1 se propaga la clave de cualquiera de ellas a la otra tabla. Siguiendo el ejemplo anterior, podría realizarse la misma conversión, o la siguiente:

EMPLEADO (DNI, Nombre, Cod\_Puesto)

PUESTO (Cod\_puesto, Ubicacion).

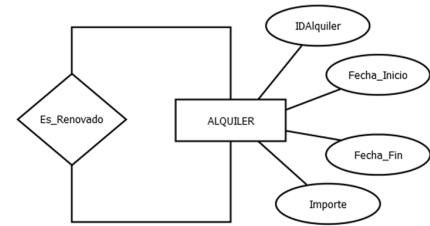
## Relaciones reflexivas a esquema relacional

En este tipo de relaciones se debe tener muy en cuenta la cardinalidad. Se pueden dar los siguientes casos.

- **Relación 1 a 1.** No genera tabla nueva. La clave de la entidad se repite pero con identificadores distintos. Un identificador será PK y el otro FK de ella misma.

Ejemplo. Queremos almacenar los distintos alquileres y si los alquileres han sido renovados, almacenar la id del alquiler anterior. Aunque en este ejemplo no se ha identificado la cardinalidad de la entidad, podría asignarse una cardinalidad 0,1, de manera que si dicho alquiler nunca ha sido renovado, no contendrá la ID de ningún alquiler anterior:

ALQUILER (IDAquiler, Fecha\_Inicio, Fecha\_Fin, Importe, IDAquiler\_Anterior).



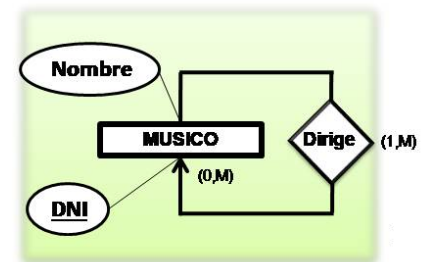
- **Relación 1 a M.** Se pueden dar dos casos.
  - Si la entidad "Muchos" es siempre obligatoria, actuaremos igual que en una relación 1 a 1.
  - Si la entidad "Muchos" no es obligatoria crearemos una nueva tabla cuya clave será la del lado "Muchos" y se propaga hasta esta nueva tabla la clave como FK.

Ejemplo:

MUSICO (DNI, Nombre)

DIRIGE (DNI, DNI\_Director)

DNI\_Director es el DNI del músico que juega el rol de director y es clave foránea de la tabla MUSICO

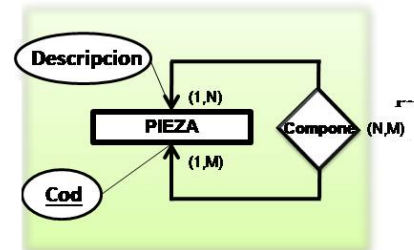


- **Relación N a M.** También se crea una nueva tabla, pero en este caso la PK será la combinación de las dos claves de la otra entidad.

Ejemplo:

PIEZA (Cod\_Pieza, Descripcion)  
 PIEZA\_COMPUESTA (Cod\_Pieza,  
Cod\_Pieza\_Comp)

Cod\_Pieza es la clave de la pieza en cuestión y Cod\_Pieza y Cod\_pieza\_Comp es la pieza o piezas que la componen. Los dos son PK.



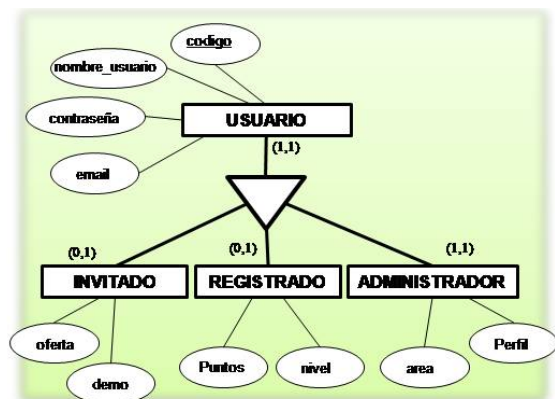
## Jerarquías de E/R a esquema relacional

Hay tres formas de tratar este tipo de relaciones:

- Crear una única entidad que aglutine todos los subtipos y sus atributos, esta unión permite simplicidad pero puede provocar valores nulos en atributos propios de cada subtipo.

Ejemplo:

USUARIO (Codigo,  
 Nombre\_Usuario, Contraseña,  
 Email, Oferta, Demo, Puntos,  
 Nivel, Area, Perfil).

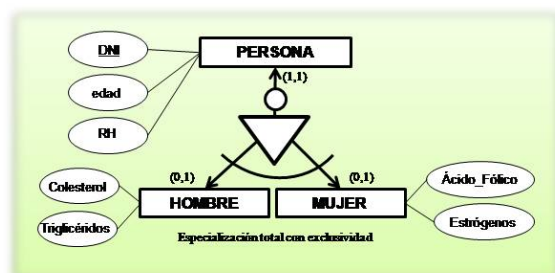


- Anulación del supertipo, de modo que sus atributos pasan a todos los subtipos y las relaciones del supertipo se han de reproducir en cada subtipo. La clave de supertipo pasa a subtipo.

Ejemplo:

HOMBRE (DNI, edad, RH,  
 Colesterol, Triglicéridos)

MUJER (DNI, edad, RH,  
 Ácido\_Fólico, Estrógenos)



- Añadir relaciones 1 a 1 entre el supertipo y los subtipos. Los atributos de supertipo se mantienen y cada uno de los subtipos tendrá clave foránea del supertipo.

Ejemplo:

PERSONA (DNI, edad, RH)

HOMBRE (DNI, Colesterol, Triglicéridos)

MUJER (DNI, Ácido\_Fólico, Estrógenos)

## Relaciones de muchos a muchos a esquema relacional

Se transforman en una nueva tabla que tendrá como clave primaria la unión de las claves primarias de las entidades que asocia.

Ejemplo:

PRODUCTO (Ref, Descripción)

TRABAJADOR (DNI, Nombre)

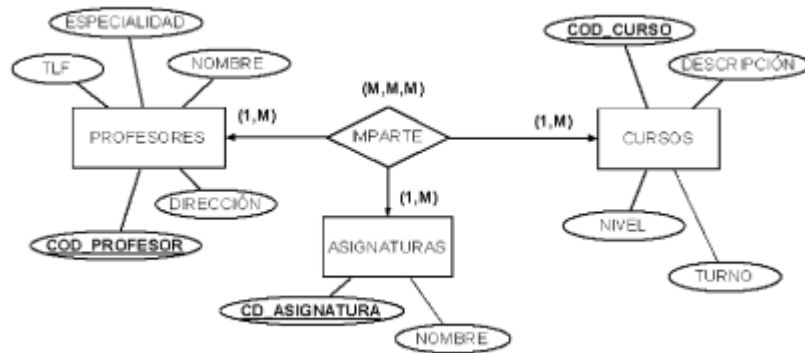
ELABORA (Ref; DNI, Fecha\_Elaboración)



## Relaciones N-Arias a esquema relacional

Cada entidad se convierte en tabla y también la relación que contendrá sus atributos propios más las claves de todas las entidades. La clave principal serán todas las claves de entidades. Se dan dos casos dependiendo de la cardinalidad.

- **N:M:N.** Si todas las entidades que participan tienen cardinalidad máxima de muchos, la clave resultante será la unión de todas las entidades que relaciona.



Ejemplo:

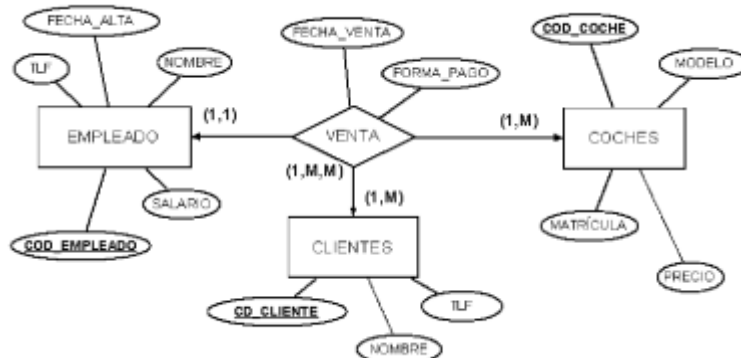
PROFESORES (CodProfesor, Dirección, Nombre, Teléfono Especialidad)

CURSOS (CodCurso, Descripción, Nivel, Turno)

ASIGNATURAS (CodAsignatura, Nombre)

IMPARTE (CodProfesor (FK), CodCurso (FK), CodAsignatura (FK))

- **1:N:M.** En caso de que una de las entidades participe con cardinalidad máxima 1, la clave de esta entidad será un atributo más de la nueva tabla, pero no será PK.



Ejemplo:

CLIENTES (CodCliente, Nombre Teléfono)

EMPLEADO (CodEmpleado, Nombre, Telefono, Salario, FechaAlta)

COCHES (CodCoche, Matrícula, Modelo, Precio)

Venta (CodCoche (FK), CodCLiente (FK), CodEmpleado (FK), FormaPago, FechaVenta)

No obstante, Las relaciones N-Arias no deberían existir en nuestro esquema conceptual modificado (ECM) si hemos realizado la tarea de simplificación correctamente.

# 11. Normalización de modelos relacionales

Es el proceso que consiste en imponer a las tablas del modelo relacional una serie de restricciones a través de un conjunto de transformaciones consecutivas. Esto garantiza que las tablas contengan los atributos necesarios y suficientes para describir la realidad de la entidad que representan y permite separar los atributos que por su contenido podrían generar la creación de otra tabla.

La técnica de normalización se utiliza como proceso de refinamiento que debe aplicarse después de lo que conocemos "paso a tablas" o traducción del esquema conceptual al esquema lógico. Este proceso consigue:

- Suprimir dependencias erróneas entre atributos.
- Optimizar procesos de inserción, modificación y borrado de la BBDD.

Esta normalización se aplica de varias etapas secuenciales. Cada etapa está asociada a una forma normal, que establece unos requisitos a cumplir por la tabla sobre la que se aplica y no se puede pasar a la siguiente hasta que no se satisface completamente.

Estas formas normales son: Primera, Segunda, Tercera Boyce-Codd, Cuarta, Quinta y Dominio-Clave. Cada forma normal es más restrictiva que la anterior.

Es recomendable aplicar la normalización hasta tercera forma normal o hasta Boyce-Codd.

## 11.1. Tipos de dependencias

Para aplicar las formas normales es necesario conocer este concepto.

### Dependencia funcional

Dados los atributos A y B, B depende funcionalmente de A sí, y solo sí, para cada valor de A sólo puede existir un valor de B. A es atributo determinante, ya que determina el valor de B. Su representación es la siguiente:  $A \rightarrow B$ . A y B podrían ser un solo atributo o conjunto de ellos. Ejemplo. Entre DNI y NOMBRE existe una dependencia funcional:  $DNI \rightarrow Nombre$ .

Se estudian las DF para encontrar claves candidatas y obtener el mínimo conjunto posible de atributos que sirvan de clave principal.

### Dependencia funcional completa

Dados los atributos  $A_1, A_2, \dots, A_k$  y  $B$ ,  $B$  depende funcionalmente de forma completa de  $A_1, A_2, \dots, A_k$ , si y solo si  $B$  depende funcionalmente del conjunto completo de atributos pero no de ninguno de sus posibles subconjuntos.

Ejemplo. Entre  $\text{DNI.Empresa} \rightarrow \text{Nombre}$  no existe DF completa, sino parcial, porque nombre no depende de empresa. En cambio si que existiría entre los atributos  $\text{DNI.Empresa} \rightarrow \text{Sueldo}$ .

Las dependencias totales se utilizan para tratar anomalías y solucionarlas y se tratan en la 2ª FN.

## Dependencia transitiva

$A$  y  $C$  tienen dependencia transitiva si  $B \rightarrow C$  y si  $A \rightarrow B$

Ejemplo. Tenemos los atributos  $\text{Num\_matrícula}$ ,  $\text{grupo\_asignado}$  y  $\text{aula\_grupo}$  con el condicionante de que un alumno solo tiene un grupo asignado y un grupo siempre corresponde a un único aula.

$\text{Num\_Mat} \rightarrow \text{Grupo\_Asig} \mid \text{Aula\_grupo}$

$\text{Grupo\_Asig} \rightarrow \text{Aula\_Grupo}$

El atributo  $\text{Aula\_Grupo}$  es transitivamente dependiente de  $\text{Num\_Mat}$ .

## 11.2. Formas normales

### 1ª forma normal

Una tabla está en 1FN o FN1 si, y solo si, todos los atributos de la misma contienen valores atómicos (si los atributos no clave, dependen funcionalmente de la clave).

1. Se crea a partir de la tabla inicial una nueva tabla cuyos atributos son los que presentan dependencia funcional de la clave primaria. Esta tabla estará ya en 1FN.
2. Con los atributos restantes se crea otra tabla y se elige la clave primaria. Comprobamos si esta segunda tabla está en 1FN y si no tomaremos la segunda tabla como tabla inicial y repetimos el proceso.

### 2ª forma normal

Una tabla está en segunda forma normal si, y solo si, está en 1FN y además todos los atributos que no pertenecen a la clave dependen funcionalmente de forma completa de ella.



1. Se crea a partir de la tabla inicial una nueva tabla con atributos que dependen funcionalmente de forma completa de la clave. La clave será la misma que la clave primaria de la tabla inicial.
2. Con los atributos restantes se crea otra tabla que tendrá por clave el subconjunto de atributos de la clave inicial de los que dependen de forma completa. Si esta tabla está en 2FN, la tabla inicial ya está normalizada y el proceso termina, si no, tomaremos esta segunda tabla como inicial y repetimos el proceso.

### 3ª forma normal

Una tabla está en tercera forma normal sí y solo sí está en 2FN y además cada atributo que no está en la clave primaria no depende transitivamente de la clave primaria.

1. Se crea a partir de la tabla inicial una nueva tabla con los atributos que no poseen dependencia transitiva de la clave primaria.
2. Con los atributos restantes se crea otra tabla con los dos atributos no clave que intervienen en la dependencia transitiva, se elige uno como clave primaria si cumple requisitos. Se comprueba si la tabla está en 3FN, si es así, la tabla inicial está normalizada y el proceso termina, si no, tomamos la segunda tabla como inicial y repetimos el proceso.

### Forma normal de Boyce Codd

Una tabla está en FNBC o BCFN si, y solo si, está en 3FN y todo determinante es clave candidata. Un determinante será todo atributo simple o compuesto del que depende funcionalmente de forma completa algún otro atributo de la tabla.

Aquellas tablas en las que sus atributos forman parte de la PK estarán en FNBC. Si encontramos un determinante que no es clave candidata, la tabla no estará en FNBC.

Tendremos que descomponer la tabla inicial en dos, siendo cuidadosos para evitar pérdida de información.

### Otras formas normales

La 4FN se basa en el concepto de dependencias multivaluadas, la 5FN en las dependencias de Join o reunión y la DKFN en restricciones impuestas sobre dominios y claves.

# Mapa Conceptual

