

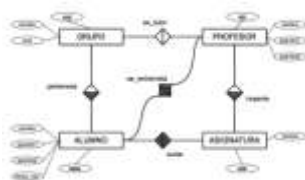
Profesora: Olga Cuervo Miguélez

TEMA 3

DISEÑO CONCEPTUAL. MODELO ENTIDAD-RELACIÓN

Módulo: **Bases de Datos**

Ciclo: **DAM**



En este tema se aprenderán las principales características de los componentes del modelo entidad-relación, cómo se definen, refinan e incorporan las relaciones entre entidades.

Introducción

Las **bases de datos** son un gran pilar de la programación actual, ya que nos permiten almacenar y usar de forma rápida y eficiente cantidades ingentes de datos con cierta facilidad. En la actualidad se usan de forma mayoritaria las bases de datos relacionales, dominadas por distintos gestores de bases de datos a través de un lenguaje para la definición, manipulación y consulta de los datos, como puede ser SQL.

En la etapa de modelado o **diseño conceptual** se busca la manera de representar la estructura de una base de datos, en su mayoría relacional, buscando la representación de los esquemas.

Un modelo muy usado es el que plantearon los informáticos Peter Pin-Shan Chen y Edgar Frank Codd, y que llamaron **modelo entidad-relación**. Este método compuesto por diagramas se usa, entre otros fines, para diseñar los esquemas que permiten implementar bases de datos relacionales haciendo uso de aplicaciones informáticas denominadas **sistemas gestores de bases de datos**, representándolo a través de diagramas.

Estos **diagramas** ayudan a entender los datos y cómo se relacionan entre ellos, y deben ser complementados por un informe que liste los atributos y las relaciones entre cada elemento. Los **elementos principales** con los que cuenta este modelo son las entidades, los atributos simples, las relaciones, la cardinalidad entre las relaciones, los atributos que comportan la clave principal o primaria, los atributos que comportan la clave externa o foránea y otras estructuras más avanzadas que representan jerarquía, especialización y otros conceptos más específicos.

En este tema se procede al estudio de cada uno de dichos elementos usando ejemplos sencillos para alcanzar su entendimiento con el fin de llegar a obtener verdaderas estructuras complejas de representación del modelo conceptual. Al final de cada apartado se incluirá una serie de casos de uso que aglutina todos los elementos estudiados hasta el momento.

En el desarrollo de los contenidos se hace necesario adelantar algunos conceptos para analizar posibles mejoras y optimizaciones, pero que más tarde se detallan y se ejercitan para aprender estos, integrándolos con conceptos anteriores.

3. El diseño conceptual en bases de datos

El **diseño de bases de datos** supone aquel proceso de generación de un diseño que permita dar soporte a la finalidad y a sus objetivos, con el fin de su implementación y explotación con el uso de sistemas gestores de bases de datos requeridos.

La intención de un modelado de datos es ofrecer unas técnicas para comprender el significado de los datos y facilitar los requisitos de información ubicados en un contexto, entendiendo como tal al universo limitado por el sistema de información que pretende gestionar toda esa información, por ejemplo, la gestión de una biblioteca, la gestión de las ventas de una empresa, la de sus compras, la gestión de un campeonato de fútbol, la gestión del proceso de trazabilidad en entornos de producción, la gestión de la logística de distribución de una organización, etc. La construcción de un modelo conceptual en el diseño de bases de datos pretende dar forma sobre aquello de lo que se quiere guardar información, enfocándose en cuatro niveles principales:

- 1. Detección de cada uno de los tipos de entidad:** detectar cada elemento del que se quiere almacenar información en el contexto dado.
- 2. Detección de sus atributos:** se deben reconocer las propiedades o características de estas entidades. Los atributos comportan la forma abstracta o concreta de las entidades.
- 3. Detección de las relaciones:** detectar la naturaleza de las relaciones que existen entre las entidades del contexto.
- 4. Detección de las restricciones:** conocer las posibles restricciones de cada una de las propiedades de cada tipo de entidad, tanto para los atributos como para las relaciones.

El modelo de datos es dependiente de la visión que el diseñador tiene de los requisitos de información.

La educación de requisitos, también llamada **elicitación de requisitos** (del inglés *requirements eliciting*), es el conjunto de técnicas iniciales en la ingeniería de requisitos y busca conocer el dominio del problema, atendiendo al sistema actual, para identificar las necesidades de los que quieren explotar el sistema de información que automatizar. La comunicación con el cliente, parte fundamental que plantea el problema que resolver, es de vital importancia en estos procesos. El problema fundamental radica en la propia carencia del cliente en técnicas para informar de sus necesidades e incluso el desconocimiento real de estas.

Este complejo proceso conlleva las siguientes fases:

- 1. Análisis del dominio del problema**, en el que se estudia la situación actual. En estas primeras acciones se desarrolla un glosario de términos y se obtiene el modelo de negocio actual.
- 2.** Cuando esta fase está lo suficientemente refinada, se **detectan las diferentes necesidades que cubrir**. Se obtienen documentos relativos a la información de las necesidades y el modelo del negocio que se quiere implantar. Las necesidades deben ser clasificadas, categorizadas y evaluadas en cuanto a prioridad, obteniendo el informe sobre prioridades.

El **modelado del negocio actual** describe cada uno de los procesos de negocio de la situación actual del negocio. Del mismo modo lo hace el modelado del negocio que se va a implantar. Se pueden usar diagramas de actividades y diagramas de casos de uso para representar de un modo global cada uno de esos procesos.

El **análisis de las diferentes entrevistas** llevadas a cabo con todo el personal competente de la organización a la que se le quiere mecanizar el proceso de negocio concluye con diferentes actas que especifican con claridad la información recopilada en cada una de las reuniones. Generalmente son los ingenieros de software, junto al equipo de analistas, los que realizan este tipo de trabajo.

Del proceso de **edución de requisitos**, que hace referencia a la parte de la información y su modelado, se obtiene una lista de requisitos de información que cumplir. De este se va a partir para comenzar el proceso de diseño de las bases de datos. El conjunto de estos requisitos será la entrada del primer proceso del diseño de la base de datos modelo conceptual de datos.

El análisis de requisitos de información se relaciona con los aspectos referentes a los requisitos de un sistema en cuanto al subsistema de información. El conjunto de requisitos de un sistema completo se aborda en la ingeniería de requisitos, con el fin de determinar las necesidades de aquellos que quieren explotarlo, tanto los aspectos hardware como los aspectos software. Por extensión, los requisitos de información hacen referencia a las necesidades relacionadas con la información. Para ello, se trabaja bajo un modelo de datos que define las estructuras que son capaces de contenerlos y las condiciones para que represente el contexto real en el que se ubica. Los **modelos de datos** más conocidos son:

1. Los **conceptuales**, hace referencia a la estructura y las condiciones de cumplimiento de los datos.
2. Los **lógicos**, se fundamentan en las operaciones más que en la descripción conceptual.
3. Los **físicos**, son la base a toda la construcción real haciendo uso de software que permite construir las bases de datos.

El modelo que mejor se adapta a la naturaleza de las bases de datos actuales que no son gigantes es el modelo relacional. Este está compuesto de elementos base denominados **relaciones**; tuplas que almacenan datos en una estructura bidimensional.

El modelo conceptual que mejor permite el diseño de las bases de datos de tipo relacionales es el **modelo entidad-relación**. Con este modelo se pretende obtener una representación simple sobre aquello de lo que se quiere conocer, sus características y las relaciones entre ellos. Estos elementos que dan base al diseño se denominan **entidades** y sus propiedades, **atributos** y **relaciones**.

Las **bases de datos orientadas a objetos** usan estructuras que no son tablas para almacenar la información. En la actualidad existen mecanismos para trabajar con bases de datos relacionales, bases de datos objetos y las orientadas a ambos usos, las bases de datos objeto-relacionales.

3.1. Entidades

Las **entidades** representan personas y objetos que se diferencian entre sí, y de las que se desea almacenar información. Puede ser una entidad concreta como un ordenador un libro, un profesor, un estudiante, un aula, un producto, etc., o una entidad abstracta con existencia conceptual, como puede ser una asignatura, una carrera, un departamento, una profesión, etcétera.

Teniendo en cuenta la abstracción de clasificación se distingue entre:

- ❖ *Tipo de entidades*, que es la estructura genérica que describe un conjunto de entidades.
- ❖ *Entidades*, que son los ejemplares de ese tipo de entidad.

Es decir, el tipo de entidad es el resultado de la clasificación de un conjunto de entidades.

En realidad, en los diagramas entidad-relación se representa la colección tipos de entidad que comparten los mismos atributos o propiedades. Por ejemplo, el tipo de entidad *profesor* aglutina a todas las instancias de dicho tipo de entidad que comparten los mismos atributos o características: *Antonio Marín Marea*, *Raquel Sánchez Mora*, *Iván Palacios Marín*, etc. Cada uno de estos profesores sería una entidad del tipo de entidad profesor.

Comúnmente, se sintetiza esta dualidad reduciendo a *entidad* el tipo de entidad al que puede pertenecer un objeto o cosa, eso sí, siempre que compartan los mismos atributos.

Uno de los problemas en el diseño conceptual es la decisión si un determinado objeto o concepto se modela como un tipo de entidad o no. Puede ser un atributo, o formar un tipo en su clasificación, etc. Se proponen tres **reglas** generales que debe cumplir una entidad:

1. Una entidad tiene que tener existencia propia.
2. Cada ejemplar de un tipo de entidad, lo que denominamos **entidad**, debe poder distinguirse de las demás.
3. Todos los ejemplares de un tipo de entidad deben tener las mismas propiedades.

Ejemplos de entidades puede ser: profesor, estudiante, asignatura, producto, cliente, empleado, vendedor, comercial, departamento, transportista, vehículo, vuelo, vivienda, aula, hotel, sucursal, banco, etcétera.

La **representación** en el modelo entidad-relación para un tipo de entidad es un **rectángulo** en cuyo interior se incluye el nombre del tipo de entidad. El nombre que se le da a una entidad suele ser un **SUSTANTIVO**.



A continuación, se representan los tipos de entidad: *empleado*, *móviles*, *fabricante*, *cliente*, *proveedor*, *tienda*, *oficina*, *alquiler*, *marca* y *libro*.



Figura 1. Los tipos de entidad indican los elementos de los que se quiere almacenar información. Se representa con un rectángulo

En relación al tipo de entidad *MÓVILES*, en general, el nombre de la entidad se escribe en singular, ya que identifica a un tipo de entidad: la entidad *Francisco Marín* es de tipo de entidad *Empleado*, siendo extraño decir la expresión «del tipo de entidad Empleados». En el ejemplo anterior, se ha escrito *MÓVILES* porque el diseñador hace referencia, que debería evitarse, a la tabla *Móviles*, que almacenará todos los registros del tipo Móvil.

3.2. Los atributos en los tipos de entidad

Los **atributos** definen o identifican las características de una entidad. Los atributos describen propiedades que posee cada miembro de un conjunto de entidades o tipo de entidad. Estos atributos pueden ser de diferentes tipos. Ejemplos de atributos para el tipo de entidad *Profesor* puede ser: DNI, nombre, primer apellido, segundo apellido, número de la Seguridad Social, fecha de nacimiento, domicilio habitual, etc. Todas las entidades comparten los mismos atributos. Las instancias de dichos atributos caracterizan de forma unívoca a dicha entidad, al menos, por uno de sus atributos.

Hay que tener en cuenta para la detección de los diferentes atributos de una entidad que, en ocasiones, un aparente atributo en realidad se trata de un tipo de entidad. No es difícil detectar este tipo de situaciones, pero debe prestarse atención a esta posibilidad. En general, se dice que una **propiedad** será un tipo de entidad si ella misma tiene sus propias propiedades. Se debe evitar usar tipos de entidad con tildes o con la eñe, ya que, en fases posteriores no se puede usar este tipo de caracteres en los identificadores de los objetos de la base de datos, por ejemplo, el tipo de entidad *MÓVILES* de la Figura 2.



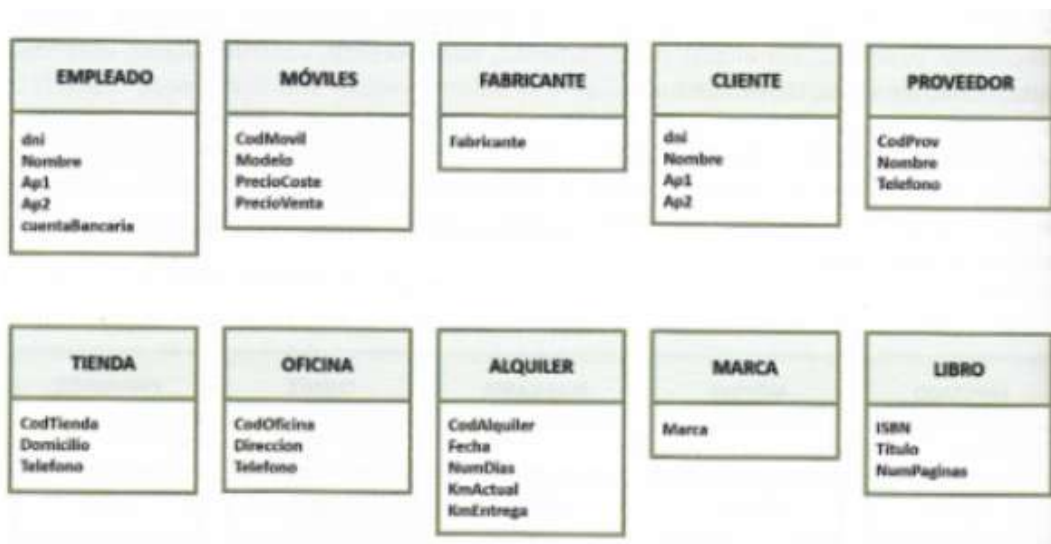
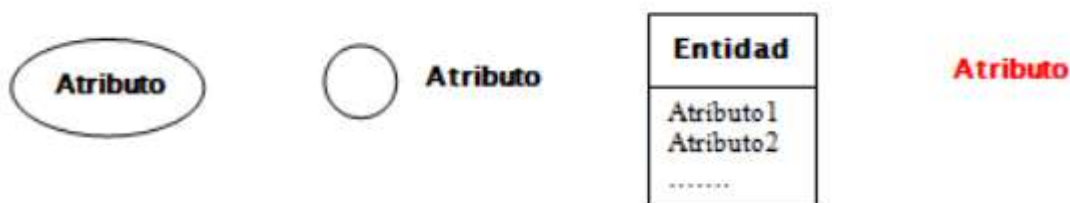


Figura 2. Los atributos caracterizan a cada tipo de entidad y toman valores en cada instancia de un tipo de entidad, es decir, en cada entidad.

4. Las propiedades

Las **propiedades** de los tipos de entidad hacen referencia a sus características, es decir, sus atributos, pero también son las consecuencias inherentes de estar relacionados con otros tipos de entidades. Los atributos que determinan cómo es un tipo de entidad y que son compartidos por todas las instancias de ese tipo de entidad se pueden clasificar atendiendo a sus especialidades. En una primera clasificación, estos son los **tipos de atributos**:

- ❖ Atributo principal o atributo secundario.
- ❖ Atributo simple o atributo compuesto.
- ❖ Atributo obligatorio u opcional.
- ❖ Atributo monovaluado o multivaluado.
- ❖ Atributo derivado.



4.1.1. Dominio de un atributo

El **dominio de un atributo** es el conjunto de valores permitidos para un atributo particular. Para cada atributo, existe un dominio de este y hace referencia al tipo de datos con el que se almacenará o a restricciones en los valores que el atributo puede tomar. El dominio se puede indicar:

- ❖ Implícitamente, a través de reglas.
- ❖ Explícitamente, especificando cada uno de los valores que dicho atributo puede tomar.

Puesto que un conjunto de entidades o tipo de entidad puede tener diferentes atributos, cada entidad se puede describir como un conjunto de pares de la forma (atributo, valor).

Un ejemplo para el tipo de entidad profesor, su atributo *DNI* puede tomar 9 cifras numéricas. El atributo día de la semana en el que nació puede tomar uno de los valores del conjunto explícito *Lunes, Martes, Miércoles, Jueves, Viernes, Sábado y Domingo*. Ejemplos de pares serían (DNI, 44229539), (nombre, Antonio), (primer apellido, Postigo), (ciudad de nacimiento, Albany), etcétera.

ATRIBUTO	DOMINIO
Edad	Números positivos
Dni	Cadenas de 8 números y una letra final
Nombre	Cadena de letras mayúsculas y minúsculas
Sexo	Hombre, Mujer
Color_pelo	Rubio, moreno, castaño, pelirrojo

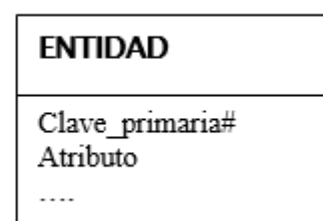
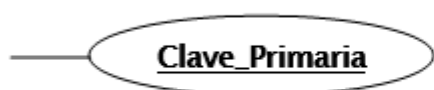
4.1.2. Atributo principal y atributo secundario

De la colección de atributos que caracteriza a un tipo de entidad, uno o más de ellos, identifica unívocamente cada una de sus posibles ocurrencias. A este atributo se le denomina **atributo principal**, *atributo clave*, *identificador principal* o *identificador candidato*. Por ejemplo, el siguiente tipo de entidad llamado *Profesor* tiene los siguientes atributos: dni, nombre, PrimerApellido, SegundoApellido, dirección y numeroCuenta. De todos ellos, el atributo principal sería dni, ya que identifica cada profesor de tipo *Profesor*.

El **atributo principal** posee las siguientes características:

- Es único para cada ocurrencia de un tipo de entidad.
- Toda ocurrencia debe tener un valor.
- Este valor es el que se usará para relacionar los diferentes tipos de entidad.
- Todos los tipos de entidad deben tener un atributo principal.
- El atributo principal puede estar compuesto por varios atributos simples.

Su representación varía según el modelo. Haciendo uso del modelo E/R se usa una circunferencia rellena en negro (● dni). En el modelo entidad-relación de este tema se usará la **letra P** delante del nombre del atributo y un punto . para separarlo (P.dni).



En la Figura 3 se puede observar la definición de cada uno de los atributos principales de cada tipo de entidad representados en la Figura 2.



Figura 3. Todo tipo de entidad tendrá uno o más atributos que identifiquen unívocamente a cada una de sus instancias o entidades. Este se llama atributo principal.

Generalmente el atributo principal está compuesto por un solo atributo, pero puede ocurrir que haya más de un atributo simple. En este caso, se coloca el símbolo en cada uno de los atributos. A continuación, se representa un tipo de entidad con un atributo principal compuesto por varios atributos simples (Figura 4).

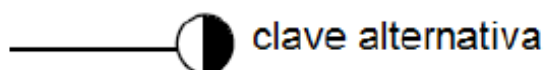


Figura 4. El atributo principal de un tipo de entidad puede estar compuesto por más de un campo.

El atributo principal de la cuenta bancaria está compuesto por los atributos simples código del banco (CodBanco), código de la sucursal (CodSucursal), código de control (CodControl) y el número de cuenta de diez dígitos dentro de dicha sucursal (NumCuenta).

Algunas entidades tienen **atributos secundarios o alternativos**. El atributo secundario tiene las mismas características que el atributo principal, identifica unívocamente a cada una de las entidades y es obligatorio, pero no es el que se usa para relacionar los tipos de entidad. En el tipo de entidad *Profesor*, un atributo secundario podría ser el número de la Seguridad Social (NSS). Es único y obligatorio, pero no es el que se usa para relacionarlo, por ejemplo, con el tipo de entidad *Asignatura*.

Su representación varia según el modelo. Haciendo uso del modelo entidad-relación se usa una circunferencia más pequeña rellena en negro, —● dni. En el modelo de este tema se usará la **letra U** delante del nombre del atributo y un punto . para separarlo, U.NSS.



En el ejemplo de la Figura 5, se ha decidido que el atributo principal de *Empleado* sea Código del empleado, *CodEmpl*, y el secundario su *dni*. Del tipo de entidad *Cliente* se ha decidido *dni* como atributo principal y su número de Seguridad Social, *NSS*, como atributo secundario. Del tipo de entidad *Proveedor* se ha elegido *CodProv* como atributo principal y el código de identificación fiscal, *CIF*, como atributo secundario. Evidentemente, esto dependerá de los requisitos de información detectados por el analista o, también, por cuestiones que mejoren el diseño.



Figura 5. Un atributo también identifica a cada entidad, pero no es el atributo que se usa para los tipos de entidad, más bien se usa para las consultas de datos.

Los atributos se clasifican en:

- **Simples:** no contienen otros atributos.
- **Compuestos:** incluyen otros atributos simples.
- **Monovalorados:** solo toman un valor para la entidad.
- **Multivalorados:** pueden tomar varios valores para una misma entidad.
- **Derivados:** sus valores pueden conseguirse con operaciones sobre valores de otros atributos.
- **Nulos:** son aquellos que en el algún momento no existen o no se conoce su valor.

4.1.3. Atributos simples y compuestos

La definición dada de atributos hasta este momento se refiere a los **atributos simples**. Estos son aquellos que no pueden subdividirse en más atributos. Sin embargo, aquellos que pueden dividirse en partes se denominan **atributos compuestos**. Por ejemplo, para el atributo fecha de nacimiento puede existir la subdivisión día, mes y año de nacimiento. El atributo cuenta bancaria, que consta de 20 dígitos numéricos, puede dividirse en banco, sucursal, dígitos de control y cuenta. El atributo domicilio puede dividirse en tipo de vía, nombre de la vía, número, planta, letra y resto del domicilio.

Los atributos compuestos son útiles para modelar situaciones en las que un usuario hace referencia al atributo compuesto como una unidad, pero otras se refieren a sus componentes. Si nunca se hace referencia a las partes de un atributo compuesto, no hay necesidad de subdividirlo. Para representarlo se usa una raya para cada parte del nombre compuesto. En la representación clásica los atributos, que son parte del atributo compuesto, tabulan con respecto al nombre compuesto con el fin de indicar que son partes de él.



*Figura 6. El atributo cuenta está compuesto por cuatro atributos denominados CodBanco, CodSucursal, CodControl, NumCuenta. Es por ello que se llama **atributo compuesto**.*

4.1.4. Atributos multivalorados

Generalmente los atributos tienen un solo valor para cada entidad de un conjunto de entidades o tipo de entidad: los atributos monovaluados. Cuando un atributo puede tomar más de un valor para una misma entidad se denominan **atributos multivalorados**. Ya veremos que este tipo de atributos no cumplen las formas normales de Chen y Codd, y por lo tanto es necesario su transformación. A nivel conceptual, se puede hacer uso de los atributos multivalorados para simplificar en estos casos el modelo conceptual que se intenta representar. Ejemplos de atributos multivalorados pueden ser el atributo teléfonos, el atributo domicilios, el atributo idiomas que se habla, etcétera.

Para representarlo se usa una flecha delante del atributo. En la representación usada en este tema se utiliza la **letra M** delante del nombre del atributo. En la Figura 7 se representan algunos ejemplos usando esta nomenclatura.

4.1.5. Atributos con valores nulos

Se dice que un atributo toma un **valor nulo** cuando una entidad no tiene un valor para un atributo específico. Un valor nulo puede representar la no existencia del valor correspondiente para dicho atributo o puede representar que aún no se conoce, en el momento de la instancia u ocurrencia de esa entidad, su valor. Por ejemplo, al instanciar una entidad del tipo de entidad *profesor*, se desconoce el número de teléfono de contacto o no tiene segundo apellido al tratarse de un profesor nacido en un país que solo hace uso de un apellido.



Figura 7. Los atributos multivalorados permiten representar multiplicidades que evitan complicar la representación conceptual.



Figura 8. Los atributos no nulos son muy importantes, determinan la obligatoriedad de la instancia de un valor.

Para representarlo se usa un círculo sin relleno delante del atributo —○. En la representación empleada en el tema se utiliza la **leta N** delante del nombre del atributo.

En la Figura 8 se representan algunos ejemplos.

4.1.6. Atributos derivados

Los **atributos derivados** son aquellos que se pueden obtener realizando alguna operación con uno o más atributos de un tipo de entidad. En realidad, no es necesario almacenarlos a menos que dichas operaciones sean complejas y el almacenamiento de dicho dato optimice temporalmente los accesos a estos evitando su cálculo continuo. En definitiva, su uso es cuestión de optimización espacial o temporal. Ejemplos de este tipo de atributos son la letra del DNI, el atributo edad si se dispone de la fecha de nacimiento, etc. Se representa a través del símbolo \square delante del nombre del atributo. En este tema se usa la **letra D**. Supóngase los siguientes requisitos de información:

- Se desea almacenar información de todos los campeonatos que una organización dirige. Se almacena su nombre (Descripción), la fecha y hora en el que comienza el campeonato y la fecha y hora de fin, el número de días que ha durado (duración), y el número de partidas que se ha jugado durante el campeonato.
- De cada campeonato se registrará la hora de comienzo, duración en minutos y el resultado de cada una de las partidas jugadas en él.

Si en campeonato existen dos atributos para registrar la fecha y hora de comienzo y la de fin, a través de operaciones se puede calcular el número de días transcurridos. Por tanto, este atributo es derivado. Para el atributo número de partidas, *NumPartidas*, es fácil calcular dicho valor si se cuenta cuántas partidas se han relacionado en la relación *CampPartida* para dicho campeonato (esto último se entenderá mejor cuando se estudien las interrelaciones). Por tanto, este atributo también es un atributo derivado. La Figura 9 representa el resultado en el diseño conceptual.



Figura 9. Los atributos derivados expresan valores que se conocen por otros medios, pero son útiles para acelerar los cálculos asociados.

Para definir las restricciones de los atributos se pueden utilizar diferentes nomenclaturas. En la Figura 10 se resumen estas pequeñas diferencias. La primera es la que se sigue en este tema.

ALUMNO	ALUMNO	ALUMNO
P•dni U•NSS Nombre Ap1 N•Ap2 MN•Telefono C•Localidad	P•dni (1,1) U•NSS (1,1) Nombre (1,1) Ap1 (1,1) N•Ap2 (0,1) MN•Telefono (0,n) C•Localidad (1,1)	—• dni —• NSS — Nombre — Ap1 —○ Ap2 →○ Telefono — Localidad

Figura 10. En la representación conceptual es muy normal indicar a cada atributo sus posibles restricciones. La nomenclatura que se usa en este tema es a través de uso de letras, la usada en la izquierda. Obviamente, como solo es una representación conceptual, se pueden usar otras nomenclaturas.

Actividad propuesta 3.1. Entidades y atributos. Caso de uso "Ventas de supermercado versión 1.0"

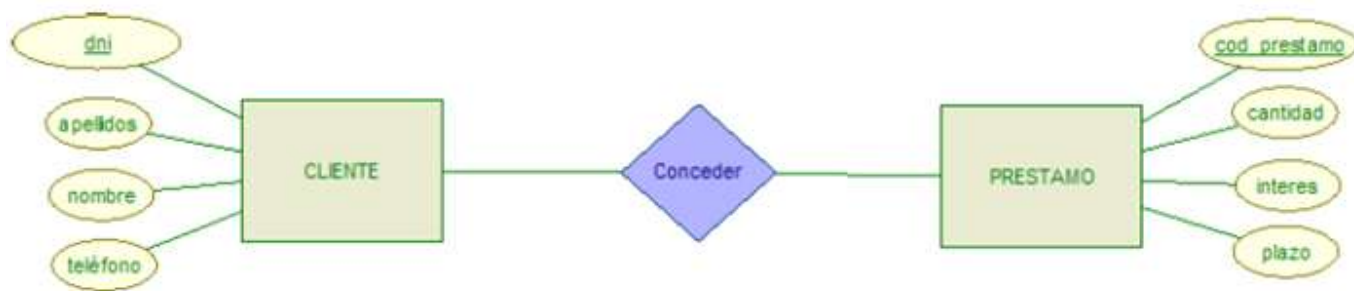
5. Las interrelaciones y su cardinalidad

Es un medio que permite definir una dependencia entre varias entidades exigiendo que compartan ciertos atributos de forma indispensable.

Las interrelaciones son algo fundamental y, sin ellas, una base de dato. Las **interrelaciones** permiten que varias entidades puedan cambiar información y a su vez obtengan resultados de dicha información. Se puede definir como la representación del modo en que las ocurrencias de un tipo de entidad, entidades, se relacionan con las ocurrencias de uno o más tipos de entidades.

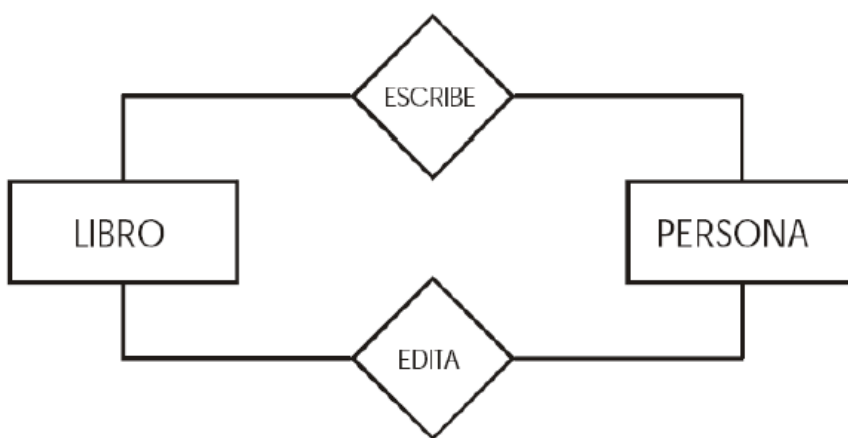
En las interrelaciones participan como mínimo dos entidades. En caso se denomina **interrelación binaria**. También existen las interrelaciones entre tres entidades diferentes. Ya estudiaremos las bases, transformaciones y resultados de este tipo de interrelaciones que pueden causar confusión.

Se pueden dar distintos tipos de interrelaciones según cómo participen en ellas las entidades que intervienen. Para ello, se usa la **cardinalidad**, elemento que permite indicar mediante un intervalo en cada extremo de la interrelación cuántas instancias de un tipo de entidad pueden intervenir en dicha relación, tanto en un sentido como en el otro.



CLIENTE				Conceder		PRESTAMO			
Dni	Apellidos	Nombre	Teléfono	Dni_cli	Cod_pre	Cod_prestamo	cantidad	Interés	plazo
34211	Blanco	Ana	67584	34211	P1	P1	15000	9	24
11321	González	Juan	75757	98117	P2	P2	30000	11	12
98117	Rivera	María	11182	34211	P3	P3	15000	9	24

Recaltar que entre dos tipos de entidad pueden existir tantas interrelaciones como sea necesario.



5.1. Tipos de interrelaciones en cuanto a su grado

Atendiendo al número de tipos de entidad que relaciona una interrelación, se pueden obtener varios tipos de interrelaciones. Si una relación relaciona un tipo de entidad consigo misma, se denomina **relación reflexiva, recursiva o en anillo**. En este tipo de interrelaciones solo participa un tipo de entidad, relacionándose consigo misma. Pueden existir tantas relaciones reflexivas para el mismo tipo de entidad como se considere oportuno.

Si en una interrelación participan dos tipos de entidad, esta se denomina **interrelación binaria**. En las interrelaciones binarias una ocurrencia de un tipo de entidad se puede relacionar con ninguna ocurrencia de la otra, con una sola ocurrencia o con muchas ocurrencias. Este análisis se debe hacer tanto en un sentido como en el otro, es decir, desde el tipo de entidad E1 hacia el tipo de entidad E2, como desde E2 a E1. Entre dos tipos de entidad pueden existir tantas interrelaciones binarias como se consideren oportunas. Este tipo de interrelación es la más común.

Si en una interrelación participan tres tipos de entidad, esta se denomina **interrelación ternaria**. En las interrelaciones ternarias es necesario estudiar cómo se relacionan cada par de entidades con la tercera entidad. Este tipo de interrelaciones supone un nivel de complejidad mayor en el diseño conceptual. A veces, hay que analizar si este tipo de interrelación puede transformarse en dos interrelaciones binarias sin perder la semántica del análisis. Otras veces, atendiendo al análisis de requisitos y teniendo en cuenta otras consideraciones en diseños posteriores, es necesario que exista como interrelación ternaria. Más adelante se tratará este tipo de relaciones.

Decir que, aunque también se puede hacer uso de las relaciones cuaternarias, estas son muy raras y complejas y que, por tanto, dificultan una solución óptima. Deben evitarse, aunque, como ya se verá, dependiendo de otras consideraciones relacionadas con etapas posteriores al diseño conceptual, su uso puede ser interesante.

5.2. Tipos de interrelaciones atendiendo a la cardinalidad

En el diseño conceptual es muy importante decidir el tipo de relación que existe entre las entidades de dos o más tipos de entidades. Tomar una decisión errónea en el tipo de interrelación afecta a tener resultados totalmente contrarios al análisis de requisitos del que se parte. Por tanto, es vital pararse a analizar las correspondencias entre entidades y su cardinalidad. Veamos a continuación qué tipos de interrelaciones disponemos.

La **cardinalidad** en una relación es el número de veces que una misma entidad de un tipo de entidad puede asociarse a otra entidad cuando dos tipos de entidad están relacionados. En realidad, es necesario analizar ambos sentidos de la relación. Atendiendo a este número de ocurrencias relacionadas, tenemos tres tipos de interrelaciones en cuanto a su cardinalidad se refiere: el tipo de correspondencia 1 a 1, 1 a muchos y muchos a muchos.

- ❖ La **interrelación tipo 1:1**, indica que una ocurrencia de un tipo de entidad solo puede relacionarse con una sola ocurrencia del otro tipo de entidad. Analizando el sentido contrario, esta misma ocurrencia del segundo tipo de entidad solo se relaciona con dicha ocurrencia del primero.

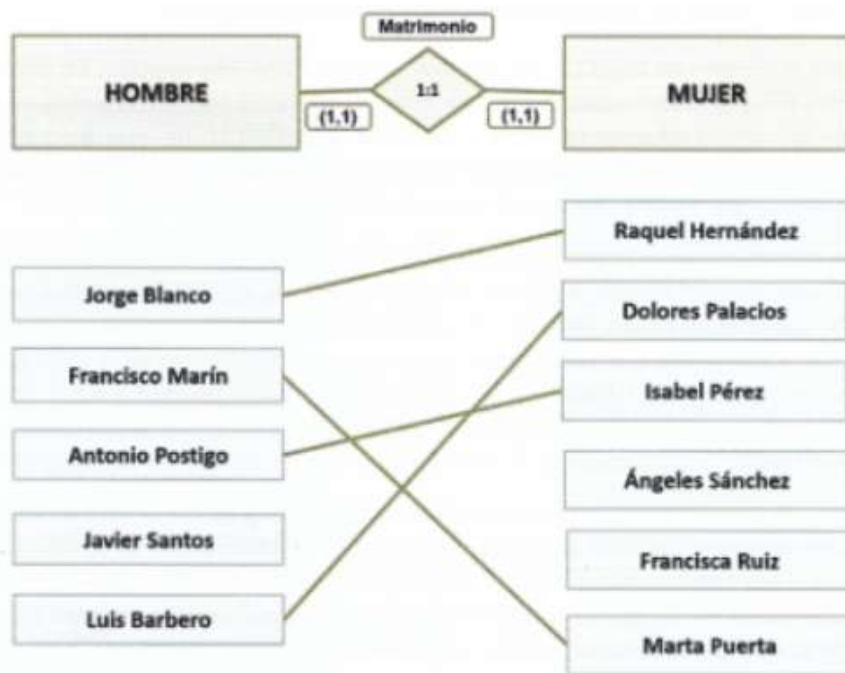


Figura 11. En la interrelación 1:1 una instancia solo se relaciona con otra una sola vez durante el ciclo de vida de la base de datos.

Sin embargo, cuando una de ellas se puede relacionar con más de una ocurrencia otra, pero al contrario no, es decir, esta ocurrencia de la segunda solo tiene dicha correspondencia con la primera, el tipo de interrelación es de **1:N** o de uno a muchos. En este tipo de interrelación es importante analizar el sentido 1:N indicando con claridad qué entidad participante posee cardinalidad 1 y cuál posee cardinalidad N. También se puede indicar como N:1, pero no es habitual, por lo tanto, identifiquemos siempre el sentido de la interrelación e indiquemos que el tipo de correspondencia es 1:N.

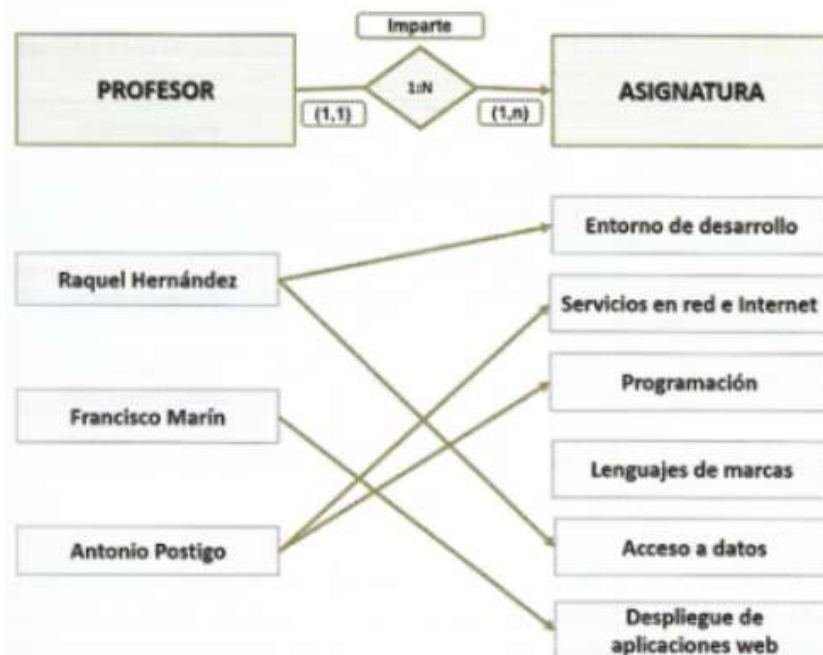


Figura 12. En la interrelación 1:N una instancia se puede relacionar con varias de otro tipo de entidad, pero estas solo lo harán una sola vez durante el ciclo de vida de la base de datos.

Por ejemplo, en la Figura 12 se puede observar que Raquel Hernández puede impartir más de una asignatura, Entorno de desarrollo y Acceso a datos. En el sentido contrario de la interrelación, la asignatura Acceso a datos solo la imparte Raquel, nunca la impartirá otro profesor. Cuando ocurra esto, la interrelación será de multiplicidad 1:N. También se puede considerar que si Raquel ya no imparte Acceso a datos, la relación sería eliminada y se crearía una nueva con el profesor que competa. Esto último no provoca que la relación no siga siendo 1:N.

Por último, cuando cualquier ocurrencia de una entidad participante puede relacionarse con más de una entidad del segundo tipo de entidad, y en el sentido contrario ocurre lo mismo, es decir, dicha entidad del segundo tipo de entidad se puede relacionar con más de una entidad del primer tipo de entidad, el tipo de correspondencia que se obtiene se denomina **N:M** o muchos a muchos.

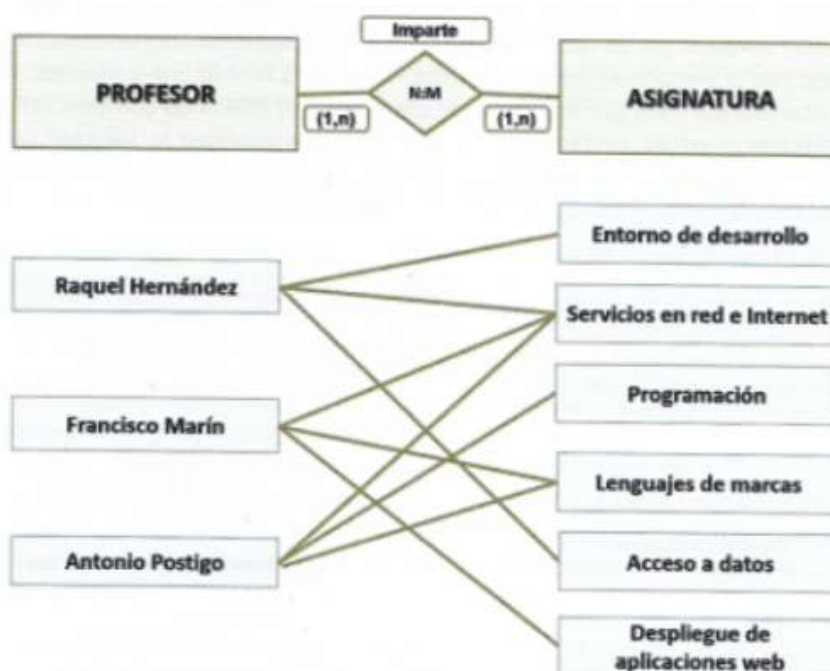
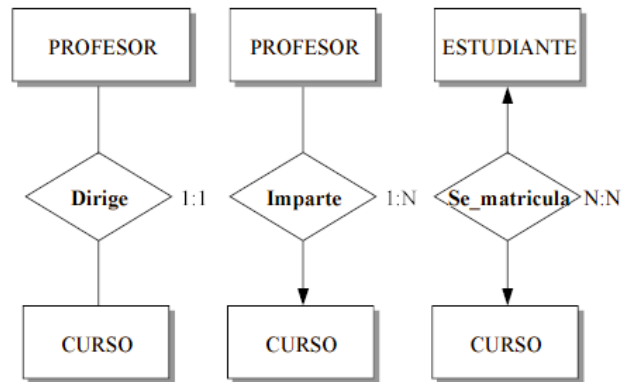


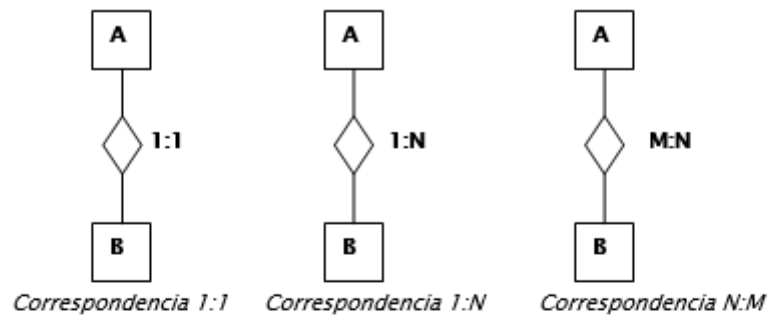
Figura 13. En la interrelación de tipo N:M las ocurrencias de un tipo de entidad se pueden relacionar con muchas de otro tipo y estas con muchas otras del primer tipo.

Por ejemplo, en la Figura 13 se puede observar que Servicios en red e internet la puede impartir más de un profesor y que un profesor, por ejemplo, Francisco Marín, puede impartir más de una asignatura. Siempre que al menos una ocurrencia tenga esta propiedad, la relación es de multiplicidad N:M.

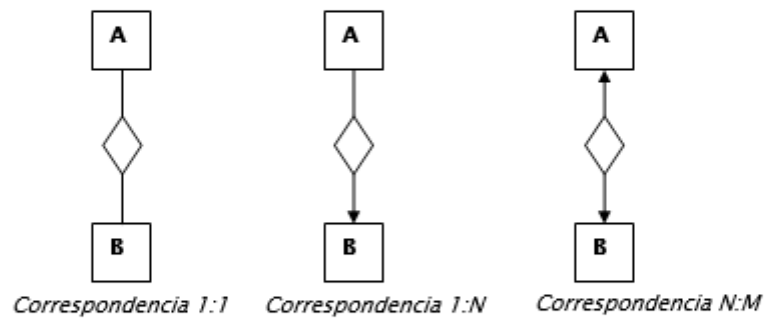


Existen diversas formas de representar la correspondencia, se algunas a continuación.

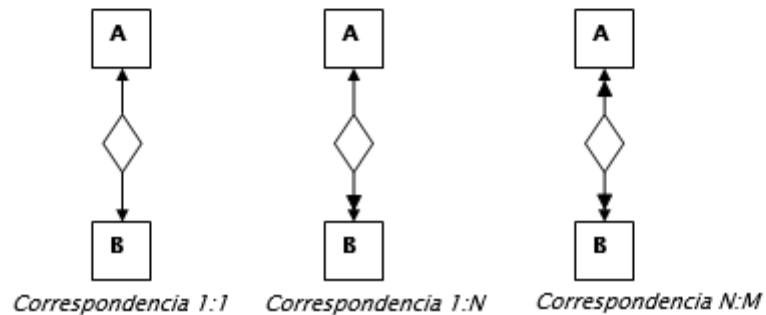
a) Indicándolo con el número al lado de la relación (notación Piattini)

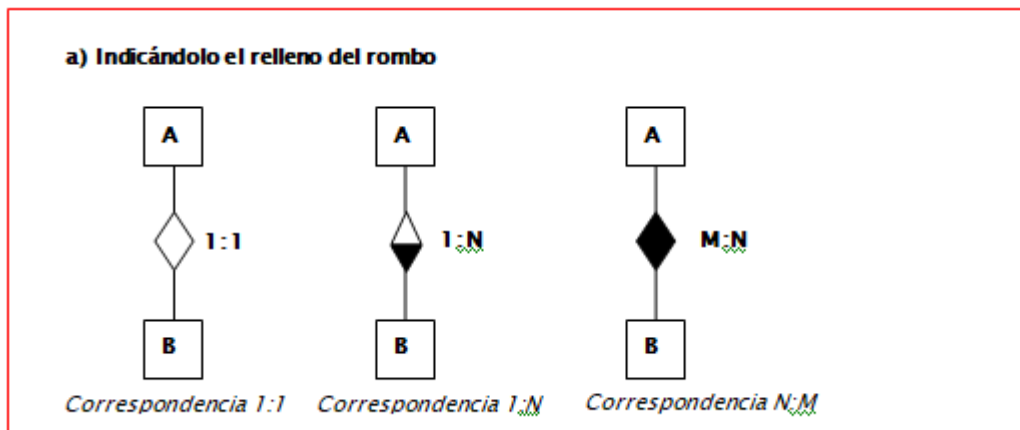


b) Reflejándolo con una flecha en el lado N (notación Chen)



c) Reflejándolo con una flecha simple en el lado 1 y con una flecha doble en el lado N





En este [enlace](#) puedes ver otros tipos de notaciones.

Actividad propuesta 3.2. Relaciones. Caso de uso "Ventas de supermercado versión 1.0"

5.3. Cardinalidad

Nº máximo y mínimo de ocurrencias de una entidad en una relación, se representa entre paréntesis, en cada uno de los lados de la relación (min, max).

- **Min**: el valor mínimo podrá ser cualquier valor igual o mayor que cero.
- **Max**: el valor máximo podrá ser cualquier valor mayor que cero (cero nunca) y el valor N para indicar un número indeterminado de ocurrencias.

De todas formas, aunque min y max podrían tener prácticamente cualquier valor numérico, los valores típicos que suelen aparecer son los siguientes estableciendo la "obligatoriedad" o no de la ocurrencia:

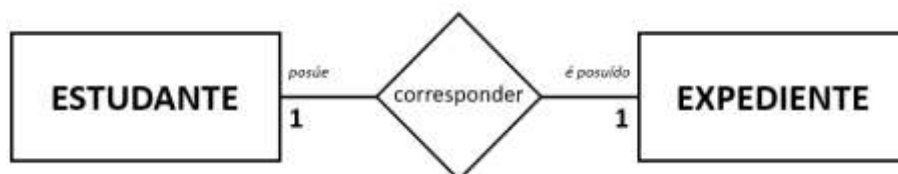
- **(1,1)**: Indica que a cada elemento de la entidad le corresponde otro en la otra entidad (obligatoriedad).
- **(0,1)**: Indica que a cada elemento de la entidad le puede corresponder uno o ningún elemento en la otra entidad (no obligatoriedad).
- **(1,N)**: Indica que a cada elemento de la entidad le puede corresponder uno o más elementos en la otra entidad (obligatoriedad).
- **(0,N)**: Indica que a cada elemento de la entidad le puede corresponder ninguno, uno o más elementos en la otra entidad (no obligatoriedad).

Para los ejemplos que hemos ido viendo en el apartado de correspondencia, si indicamos la cardinalidad quedarían del siguiente modo:

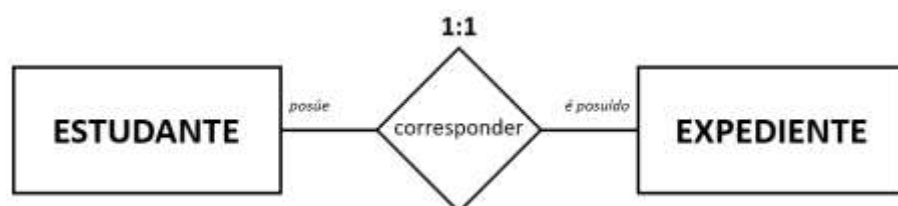
❖ **1 a 1**



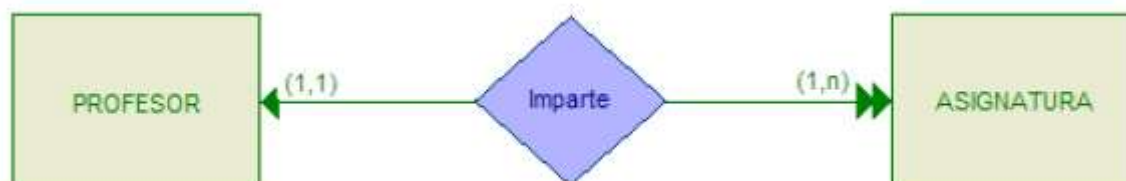
➤ **Notación Chen**



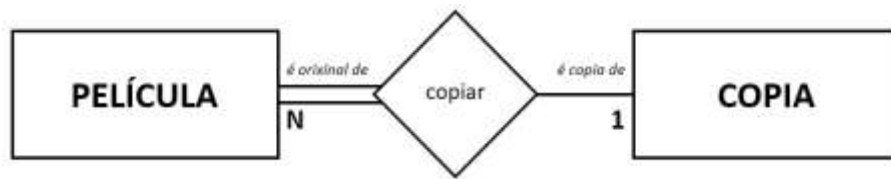
➤ **Notación Piattini**



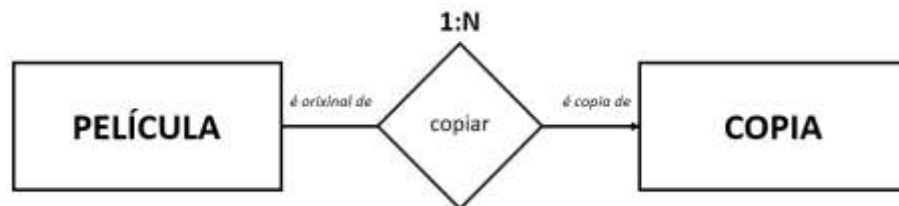
❖ **1 a N**



➤ **Notación Chen**



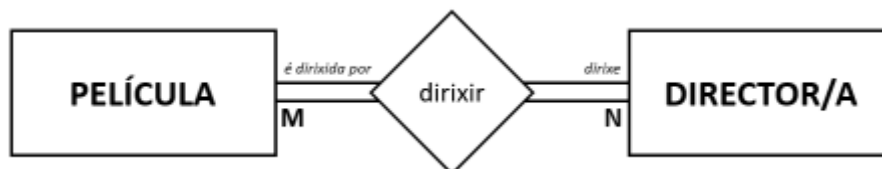
➤ **Notación Piattini**



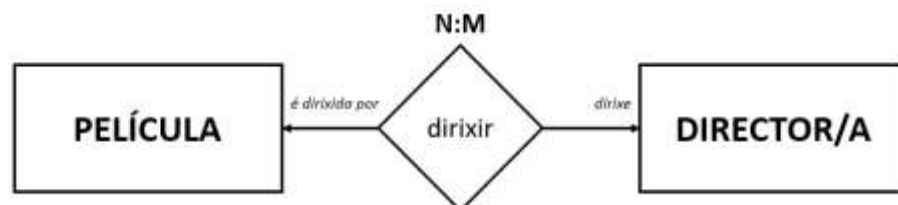
❖ **N a M**



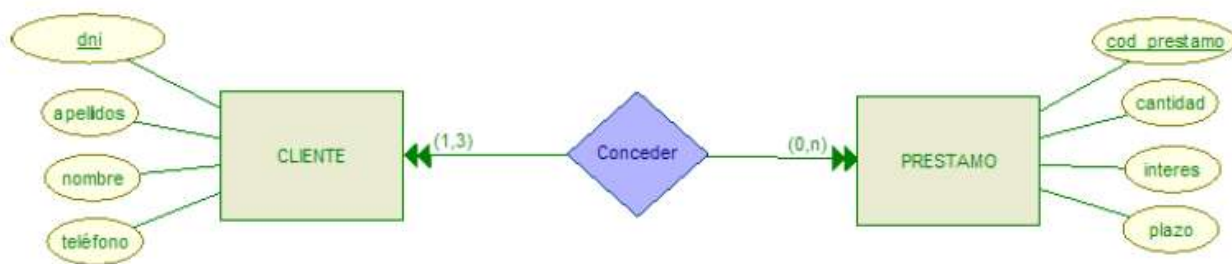
➤ **Notación Chen**



➤ **Notación Piattini**



En el siguiente ejemplo, vamos a suponer que el préstamo se puede conceder hasta un máximo de 3 personas (por ejemplo, un préstamo hipotecario muchas veces se concede a un matrimonio, es decir, 2 personas). Por tanto la correspondencia sería muchos a muchos (N:M) y las cardinalidades quedarían de la siguiente forma:



Fijarse que en este caso la cardinalidad del lado cliente es (1,3), esto representa la restricción semántica de que un préstamo se concede obligatoriamente como mínimo a 1 cliente, y que como máximo se puede conceder a 3 clientes.

Actividad propuesta 3.3. Caso de uso "Centro de formación versión 1.0"

Actividad propuesta 3.4. Caso de uso "Préstamos en una biblioteca versión 1.0"

5.4. Atributos en las interrelaciones

Como ya se dijo, un atributo es una propiedad de un tipo de entidad, pero también lo pueden ser de una interrelación. Una **propiedad en una interrelación** es aquella que se origina cuando ambas entidades se relacionan. No es una propiedad ni de la entidad origen ni de la entidad destino, sino de la relación entre ellas. Por ejemplo, la relación entre tipo de entidad Asignaturas y el tipo de entidad Alumno puede originar el atributo calificación, que es propia de la relación de un alumno con su asignatura. No es una propiedad del alumno, ni de la asignatura, sino que es necesario registrar cada calificación en cada asignatura por cada alumno. Esto se representa tal como se indica en la Figura 14.

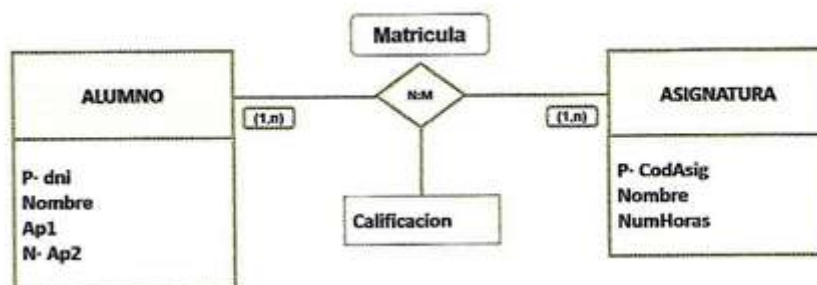


Figura 14. Es muy común que una interrelación tenga atributos en ella. Estos atributos se originan solo cuando las instancias se relacionan y no son propias de ninguno de los tipos de entidad que interrelaciona.

Ejemplo: En un comercio un cliente compra varios productos, y un producto puede ser comprado por varios clientes. Se quiere guardar además de los datos del cliente y del producto, la cantidad de cada producto que compró el cliente.



Atributos de una relación

En este caso el atributo cantidad pertenecería a la relación "compra" entre cliente y producto. No podría pertenecer a CLIENTE porque un cliente podría comprar varios productos, y de cada uno una cantidad distinta. No podría pertenecer a PRODUCTO ya que un mismo producto lo pueden comprar varios clientes, y pueden adquirir distintas cantidades.

Un ejemplo de ocurrencias de las entidades y de la relación podría ser:

CLIENTE			
Dni	Apellidos	Nombre	Teléfono
34211	Blanco	Ana	67584
11321	González	Juan	75757
98117	Rivera	María	11182

compra		
Dni_cli	Cod_prod	Cantidad
34211	P1	4
98117	P1	2
34211	P2	8

PRODUCTO		
Cod_producto	Nombre	stock
P1	Pipas	24
P2	Yogur	12
P3	Leche	24

5.5. Diferentes representaciones

Para definir las restricciones de las interrelaciones se pueden utilizar diferentes nomenclaturas. La notación Crow, también conocida como «pata de gallos», usa símbolos en los extremos de las relaciones para representar la cardinalidad mínima y máxima. En la Figura 15 se muestra la representación para las diferentes posibilidades (0,1), (1,1), (0,n) y (1,n).

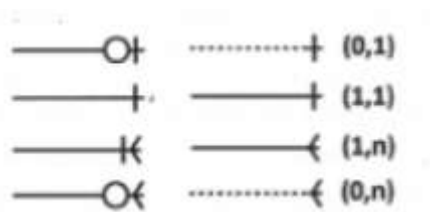


Figura 15. Existen diferentes maneras para representar la multiplicidad y las cardinalidades mínimas y máximas

Actividad propuesta 3.5. Caso de uso "Tiendas de móviles versión 1.0"

6. Las interrelaciones reflexivas/recursivas

Una **interrelación reflexiva**, también llamada **recursiva**, es aquella que relaciona entidades que pertenecen al mismo tipo de entidades. Es necesario que se relacione consigo misma. Un ejemplo de este tipo de interrelación podría ser la que existe entre las personas y aquellos que son padres o madres de sus hijos. Para este ejemplo, se entiende que todas las entidades personas están en el mismo tipo de entidad Persona, sea hijo, padre o madre. Evidentemente, todas las personas son hijos de un padre o de una madre, pero no todos son padres o madres. De cualquier modo, lo que está claro es que las entidades del tipo Persona se relacionan con otras del tipo Persona. He aquí la interrelación reflexiva que se da entre el tipo Persona y el tipo Persona.

En las interrelaciones reflexivas las entidades «destino» toman un rol para indicar qué papel cuenta cuando se relaciona con las entidades «origen».

Supóngase que se quiere modelar el siguiente requisito:

Se desea registrar las asignaturas que el centro de formación recomienda cursar con anterioridad para matricularse en una asignatura.

Obviamente, una asignatura puede tener como mínimo 0 asignaturas recomendables hasta N como máximo. De igual modo, una asignatura en cuestión puede ser recomendación de ninguna, cardinalidad mínima 0, hasta N asignaturas, cardinalidad máxima N. Por tanto, la relación que se llamará *Recomendacion* es de multiplicidad N:M y hace referencia a la asignatura recomendada. El rol que esta asignatura tiene con la que es recomendable se ha llamado *AsigRecom*. En la Figura 16 se representa el resultado de este requisito en el diagrama entidad-relación.

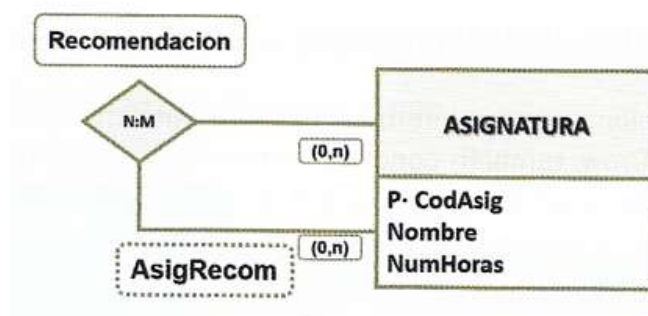


Figura 16. Las interrelaciones reflexivas relacionan instancias de un mismo tipo de entidad

Supóngase los siguientes requisitos de información para una empresa que requiere implantar una base de datos para organizar los proyectos en los que sus programadores intervienen, organizándose a través de equipos de desarrollo.

- **R1.** La empresa tiene diferentes sedes repartidas por el país identificadas por número y se quiere almacenar el nombre de la sede y su dirección.
- **R2.** En cada sede existen diferentes departamentos, de los que se quiere almacenar el nombre y la ubicación en el edificio.
- **R3.** Los programadores pertenecen a un único departamento y se necesita almacenar sus datos personales y teléfono, su cuenta bancaria y el tipo de programador (A1, A2, S1, S2, S3, M1, M2, D).
- **R4.** Los programadores experimentados son mentores de otros programadores hasta que dejan de serlo.
- **R5.** Los programadores se organizan en equipo de desarrollo y pueden pertenecer a tantos como se requiera.
- **R6.** Los equipos tienen un código y una descripción, y se necesita almacenar programador que es jefe de dicho equipo. Un programador puede ser jefe de más de un equipo.
- **R7.** A los equipos se les asignan diferentes proyectos, pero un proyecto solo es desarrollado por un único equipo de desarrollo.
- **R8.** Los proyectos tienen un código único, una descripción y una fecha de inicio de fin. Los proyectos pueden estar compuestos por proyectos más pequeños llamados «subproyectos», pero que tienen las mismas propiedades que los proyectos (descripción, fecha de inicio y fecha de finalización).

Atendiendo a R1, R2 y R3, existen los tipos de entidad Sede, Departamento y Programador. La Sede se relaciona con Departamento con multiplicidad 1:N (una sede tiene departamentos), y el Departamento lo hace con Programador también con multiplicidad 1:N (en un departamento trabajan n programadores). Con el requisito R4 se requiere de una relación reflexiva, ya que se pretende relacionar un programador con otros programadores y estos están representados en el mismo tipo de entidad. Esta parte se representa como recoge la Figura 17.

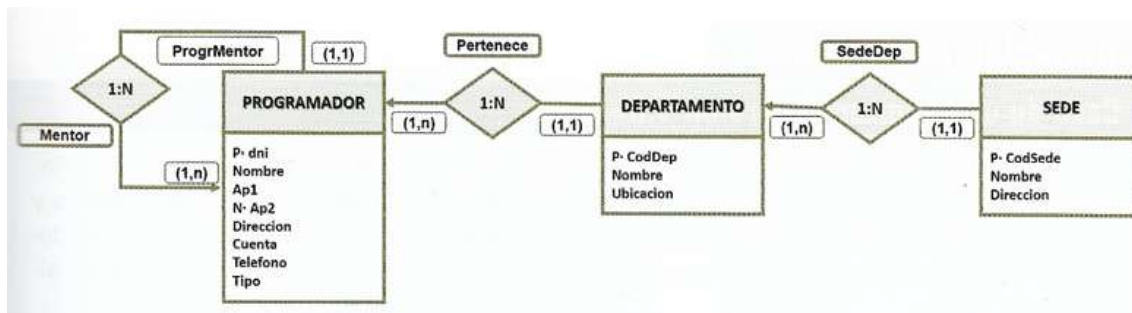


Figura 17. Primera parte del diagrama "Equipos de programadores Versión 1.0. En una sede hay n departamentos y en cada departamento trabajan un conjunto de programadores. Los programadores tienen otro programador mentor. "

Con R5, R6 y R7 se entiende que existen los tipos de entidad Equipo y Proyecto. Los atributos de Equipo son CodEquipo y Descripcion y de Proyectos, según R8, son CodProd, Descripcion, FechaIni y FechaFin.

Según R5, existe una interrelación entre Equipo y Programador de multiplicidad N:M para relacionar todos los programadores que intervienen en cada equipo, y pueden hacerlo en más de uno. Según R6, existe una interrelación entre Equipo y Programador de multiplicidad 1:N para relacionar el programador jefe de cada equipo y puede serlo de más de uno.

Según R8, se entiende que existen relaciones entre el tipo de entidad Proyecto consigo mismo, ya que cada proyecto se puede relacionar con otros proyectos. La multiplicidad es 1:N. La solución final puede verse en la Figura 18.

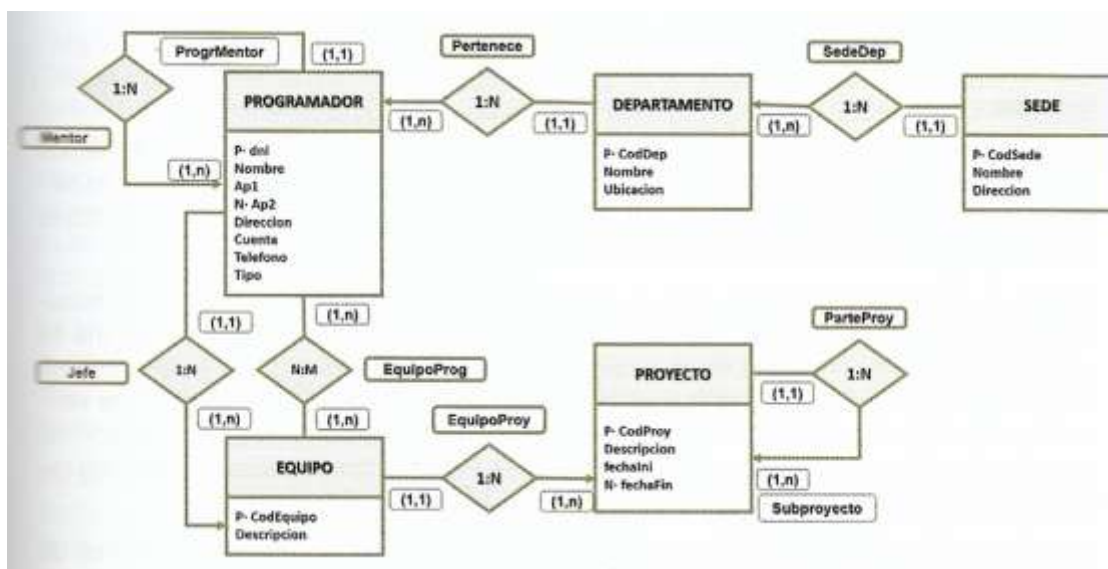


Figura 18. Representación definitiva del caso de uso "Equipos de programadores Versión 1.0"

7. Las entidades regulares/fuertes y las débiles

Los tipos de entidad se pueden clasificar atendiendo a su fuerza en su existencia con dependencia a otro tipo de entidad. Este tipo de entidad cuya existencia no depende de la existencia de otro tipo de entidad es conocido como **tipo de entidad fuerte o tipo de entidad regular**. A la entidad que depende de la existencia de otra entidad se la conoce como **entidad débil**.

El **grado de la interrelación** que las une es siempre de dos, es decir, siempre son interrelaciones binarias. Por tanto, estas interrelaciones son débiles y se dan entre un tipo de entidad fuerte y un tipo de entidad débil. En cuanto a la **multiplicidad** entre el tipo de entidad fuerte y el tipo débil, el tipo de correspondencia es siempre de uno a muchos.

Las interrelaciones débiles se pueden clasificar en dos tipos atendiendo a varias restricciones especiales: dependencia en existencia y dependencia en identificación. A continuación, se indican las diferencias principales:

- ❖ **Dependencia en existencia:** las entidades del tipo débil no pueden existir si desaparece la entidad del tipo fuerte de la que depende. Para indicar que la interrelación es por existencia, en el borde del rombo de la interrelación se escribe el texto *EX*.
- ❖ **Dependencia en identificación:** además de la dependencia en existencia, las entidades del tipo de entidad débil requieren de atributos de la entidad fuerte para su propia de identificación. Este tipo de dependencia depende de las decisiones tomadas en el análisis y, por tanto, toda interrelación en identificación puede sustituirse por una dependencia en existencia, evitando así usar atributos de la entidad fuerte para codificar la entidad débil. En otras ocasiones, será de gran interés que esto sea así. Por ejemplo, si se vieran los dos tipos de entidad Hotel y Habitaciones, cada una de las habitaciones un hotel podría identificarse por un identificador propio y único e independiente del hotel en el que se encuentre, o por un identificador dependiente del identificador del hotel unido a otro identificador no único de la habitación de dicho hotel. Para indicar que la interrelación es por identificación, en el borde del rombo de la interrelación se escribe el texto *ID*. La entidad débil se representa con doble rectángulo.

Supóngase el siguiente requisito:

- ❖ Se desea almacenar las asignaturas de cada uno de los ciclos, codificándose atendiendo a su código. De los ciclos formativos se almacenará el nombre, el número de horas de dicho ciclo y si es de grado medio o de grado superior; de las asignaturas, su nombre y el número de horas en dicho ciclo.

La asignatura depende del ciclo formativo, *CicloFormat*, en cuanto a su existencia. Sin existe un ciclo determinado, no se pueden dar entidades del tipo de asignatura de dicho ciclo. Además, se requiere usar el código del tipo de entidad fuerte, en este ejemplo *CicloFormat*, para identificar cada una de las entidades del tipo de entidad Asignatura. Obviamente, es una relación débil por identificación. Se considera que puede existir un ciclo formativo sin asignatura alguna, por lo que la cardinalidad mínima y máxima ser (0,n). Si se hubiera considerado que todo ciclo, como mínimo, tiene una asignatura, a estas cardinalidades serían (1,n). En la Figura 19 se representa el resultado final.

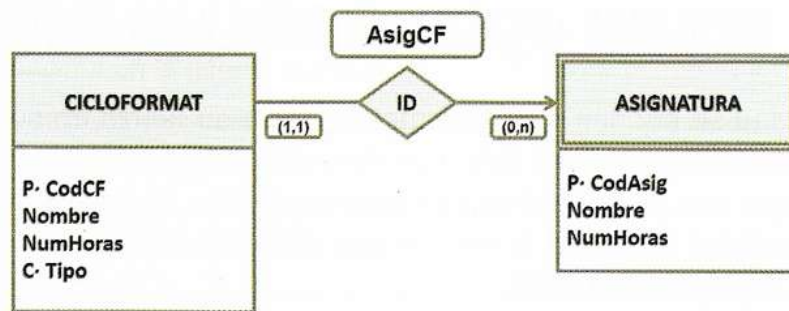


Figura 19. Representación de un tipo de entidad fuerte denominado CicloFormat, y otro débil, Asignatura, que depende por identificación del primero.

Supóngase que, para la Figura 18, se decide como atributo principal del tipo de entidad llamado Departamento la concatenación del atributo principal del tipo Sede y un código de departamento dentro de esa sede. Es decir, cada una de las instancias de las diferentes entidades se van a identificar con el código de la sede y el código del departamento dentro de la sede que corresponda. Para ello, se usa las entidades débiles por identificación. La solución se recoge en la Figura 20.

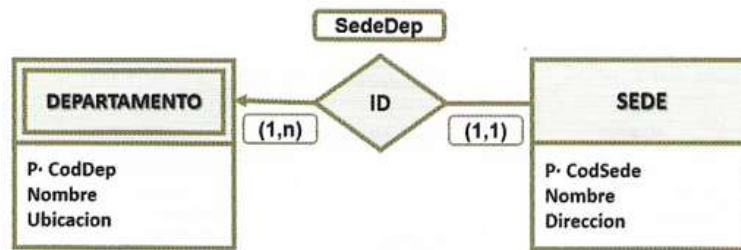


Figura 20. Representación de un tipo de entidad fuerte denominado CicloFormat, y otro débil, Asignatura, que depende por identificación del primero.

Una entidad que es débil puede ser fuerte de otra. Por ejemplo, para la Figura 18 se decide ahora que a los programadores se les va a identificar según el departamento al que pertenecen y una secuencia correlativa dentro de cada departamento. Esto se representaría como se ve en la Figura 21.

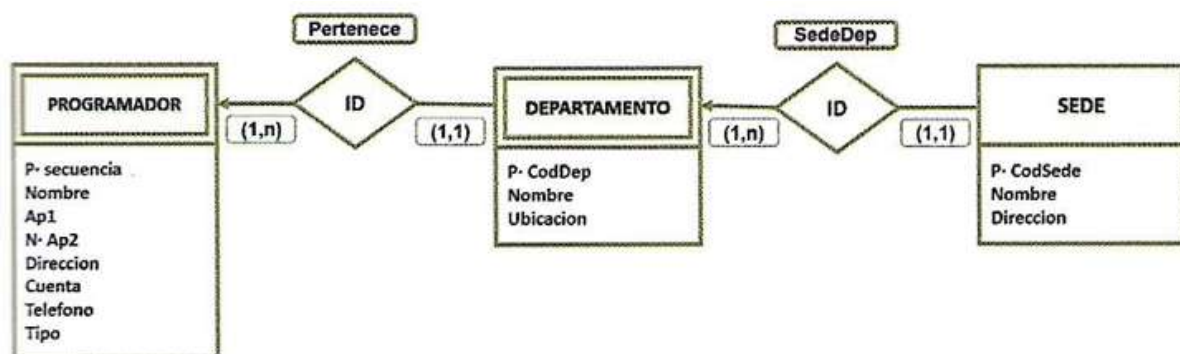


Figura 21. Un tipo de entidad débil puede ser fuerte para otro tipo de entidad.

8. Eliminar redundancias de interrelaciones

Una **interrelación** es **redundante** si puede ser eliminada sin pérdida de semántica porque existe la posibilidad de realizar la misma asociación a través de otras relaciones.

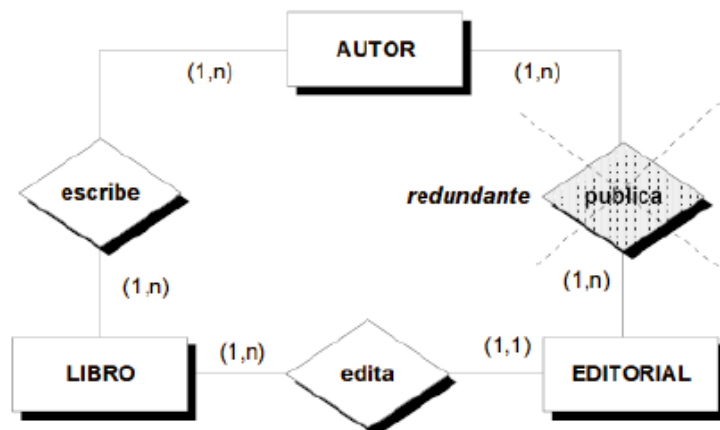


Figura 22. Ejemplo de interrelación redundante .

Las condiciones que se tienen que cumplir para que exista redundancia son:

- ❖ Debe haber un **ciclo** en el diagrama E/R como en el ejemplo.
- ❖ Las relaciones implicadas en el ciclo deben ser **semánticamente equivalentes** (es decir, tener un significado parecido). En nuestro ejemplo "publica" y "edita" son semánticamente equivalentes.

La cardinalidad mínima de las relaciones equivalentes NO pueden ser 0. En el ejemplo un autor escribe (1,n) libros, y un libro se edita en una única editorial (1,1), por tanto la relación "pública" se puede obtener a través de las relaciones "escribe" y "edita".

ÍNDICE

INTRODUCCIÓN.....	1
3. EL DISEÑO CONCEPTUAL EN BASES DE DATOS.....	2
3.1. ENTIDADES.....	4
3.2. LOS ATRIBUTOS EN LOS TIPOS DE ENTIDAD	5
4. LAS PROPIEDADES.....	6
4.1.1. DOMINIO DE UN ATRIBUTO.....	6
4.1.2. ATRIBUTO PRINCIPAL Y ATRIBUTO SECUNDARIO.....	7
4.1.3. ATRIBUTOS SIMPLES Y COMPUESTOS.....	10
4.1.4. ATRIBUTOS MULTIVALORADOS	10
4.1.5. ATRIBUTOS CON VALORES NULOS.....	11
4.1.6. ATRIBUTOS DERIVADOS.....	12
5. LAS INTERRELACIONES Y SU CARDINALIDAD	13
5.1. TIPOS DE INTERRELACIONES EN CUANTO A SU GRADO.....	14
5.2. TIPOS DE INTERRELACIONES ATENDIENDO A LA CARDINALIDAD	15
5.3. CARDINALIDAD.....	19
5.4. ATRIBUTOS EN LAS INTERRELACIONES	22
5.5. DIFERENTES REPRESENTACIONES.....	23
6. LAS INTERRELACIONES REFLEXIVAS/RECURSIVAS.....	24
7. LAS ENTIDADES REGULARES/FUERTES Y LAS DÉBILES.....	27
8. ELIMINAR REDUNDANCIAS DE INTERRELACIONES	29