

BASE DE DATOS 2º DAM


Interpretación de diagramas entidad relación.

1. Pasos en la fase de Análisis y Diseño.

Fase de Análisis	Fase de Diseño
Análisis de entidades: Se trata de localizar y definir las entidades y sus atributos.	Diseño de tablas.
Análisis de relaciones: Se definirán las relaciones existentes entre entidades.	Normalización.
Obtención del Esquema Conceptual a través del modelo E-R.	Aplicación de 🔄 retrodiseño , si fuese necesario.
Fusión de vistas: Se reúnen en un único esquema todos los esquemas existentes en función de las diferentes vistas de cada perfil de usuario.	Diseño de transacciones: localización del conjunto de operaciones o transacciones que operarán sobre el esquema conceptual.
Aplicación del enfoque de datos relacional.	Diseño de sendas de acceso: se formalizan los métodos de acceso dentro de la estructura de datos.

2. ¿Qué es el Modelo E/R?

Es una herramienta de referencia para la representación conceptual de problemas del mundo real. Su objetivo principal, facilitar el diseño de bases de datos permitiendo la especificación de un esquema que representa la estructura lógica completa de una base de datos. El modelo de datos E-R representa el significado de los datos, es un modelo semántico. De ahí que no esté orientado a ningún sistema físico concreto y tampoco tiene un ámbito informático puro de aplicación.

 **Autoevaluación**

La información con la que el modelo Entidad-Relación trabaja ha de ser lo más detallada y fiel posible a la realidad del problema a representar.

Verdadero. ☒ Falso. ☐

Correcto. Debe ser así, si no estaríamos representando parcialmente la realidad a modelar y la funcionalidad de nuestra base de datos se vería comprometida.

3. Entidades. (Se nombran habitualmente con sustantivos en singular).

Entidad: Objeto real o abstracto, con características diferenciadoras capaces de hacerse distinguir de otros objetos.

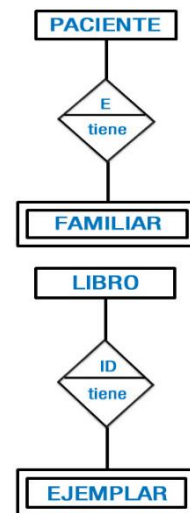
Conjunto de Entidades: Un grupo de entidades que poseen las mismas características o propiedades.

3.1 Tipos fuertes o débiles.

- Entidades fuertes o Regulares: son aquellas que tienen existencia por sí misma, es decir, su existencia no depende de otras entidades. Se representan con un rectángulo cerrado.
- Entidades Débiles: son aquellas cuya existencia dependen de la existencia de otras instancias de entidad. Es un tipo de entidad cuyas propiedades o atributos no la identifican completamente, sino que sólo la identifican de forma parcial.



- a. Dependencia en existencia: Entre entidades, si desaparece una instancia de entidad fuerte desaparecerán las instancias de entidades débiles que dependan de la primera. La representación de este tipo de dependencias incluirá una E en el interior de la relación débil.
- b. Dependencia en identificación: Debe darse una dependencia en existencia y, además, una ocurrencia de la entidad débil no pueda identificarse por sí misma, debiendo hacerse mediante la clave de la entidad fuerte asociada. La representación de este tipo de dependencia incluirá una ID en el interior de la relación débil.



Autoevaluación

Identifica cuál de las siguientes entidades no podría ser considerada como entidad débil:

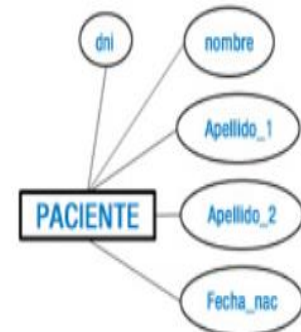
- ☒ PROVEEDOR (perteneciente a una base de datos de gestión de stocks).
- ☐ PAGO (perteneciente a una base de datos bancaria).
- ☐ FAMILIAR (perteneciente a una base de datos hospitalaria).

Efectivamente, esta entidad puede existir por sí misma sin depender de otras ocurrencias de entidad. Además, posee propiedades o atributos propios que la identifican frente a otras ocurrencias de la misma entidad.

4. Atributos.

Atributos: Son cada una de las propiedades o características que tiene un tipo de entidad o un tipo de relación se denomina atributo; los atributos toman valores de uno o varios dominios.

Aunque los dominios suelen ser amplios (números enteros, reales, cadenas de caracteres, etc.), a la hora de llevar a cabo el desarrollo de una base de datos, es mejor establecer unos límites adecuados para que el sistema gestor de la base de datos lleve a cabo las verificaciones oportunas en los datos que se almacenen, garantizando así la integridad de éstos.



4.1 Tipos de atributos.

a. Atributos obligatorios y opcionales:

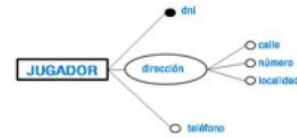
- a. Atributo obligatorio: Es aquél que ha de estar siempre definido para una entidad o relación. Una clave o llave es un atributo obligatorio.
- b. Atributo opcional: Es aquél que podría ser definido o no para la entidad, puede haber ocurrencias de entidad para las que ese atributo no esté definido o no tenga valor.

b. Atributos simples/atómicos o compuesto:

- a. Atributo simple o atómico: Es un atributo que no puede dividirse en otras partes o atributos. No es posible extraer partes más pequeñas que puedan tener significado.

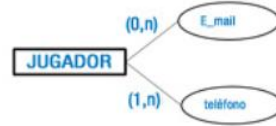


- b. Atributo compuesto: Son atributos que pueden ser divididos en subpartes, éstas constituirán otros atributos con significado propio.



c. Atributos monoevaluados o multievaluados:

- a. Atributo monoevaluado: Es aquél que tiene un único valor para cada ocurrencia de la entidad.
- b. Atributo multievaluado: Es aquél que puede tomar diferentes valores para cada ocurrencia de entidad.



- ✓ La **cardinalidad de un atributo** indica el número mínimo y el número máximo de valores que puede tomar para cada ejemplar de la entidad o relación a la que pertenece.
- ✓ La **cardinalidad mínima** indica la cantidad de valores del atributo que debe existir para que la entidad sea válida. (0-1).
- ✓ La **cardinalidad máxima** indica la cantidad máxima de valores del atributo que puede tener la entidad. (1-N).

- d. Atributos derivados o almacenados: El valor de este tipo de atributos puede ser obtenido del valor o valores de otros atributos relacionados, el ejemplo: de la fecha de nacimiento se saca la edad, por lo tanto, sería calculable dicha edad.

Autoevaluación

Rellena los huecos en blanco con los conceptos adecuados.

Si en nuestra base de datos tenemos una entidad **USUARIO**, los atributos **password** y **login** deberán ser atributos obligatorios ya que son imprescindibles para iniciar o jugar partidas. En cambio, un posible atributo **ranking** que indique en qué posición se encuentra el usuario entre todos los jugadores, podría considerarse un atributo derivado si tenemos en cuenta la puntuación obtenida por cada usuario.

Reiniciar
Mostrar las respuestas

Tu puntuación es 2/2.

Los atributos **password** y **login** deben ser **obligatorios** para que los usuarios puedan iniciar sesión y jugar partidas. Por otra parte, el atributo **ranking** puede considerarse **derivado**, ya que su valor puede obtenerse de las puntuaciones almacenadas para cada usuario.

4.2 Claves.

En el apartado anterior hablábamos de un tipo de atributo especial obligatorio, las **llaves** o **claves**. Ahora lo abordaremos con mayor detalle este concepto.

Los valores de los atributos de una entidad deben ser tales que permitan identificar unívocamente a la entidad. En otras palabras, no se permite que ningún par de entidades tengan exactamente los mismos valores de sus atributos.

Superclave (Superllave): Es cualquier conjunto de atributos que permite identificar de forma única a una ocurrencia de entidad. Una Superclave puede tener atributos no obligatorios, es decir, que no identificarían por si solos una ocurrencia de entidad.

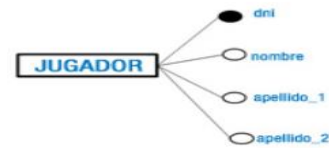
- Por lo tanto, **no pueden haber 2 superclaves, toda Superclave tiene que identificar de manera única toda la fila.**

La representación en el modelo E-R de las claves primarias puede realizarse de dos formas:

- ✓ Si se utilizan elipses para representar los atributos, se subrayarán aquellos que formen la clave primaria.




- ✓ Si se utilizan círculos para representar los atributos, se utilizará un círculo negro en aquello que forme la clave primaria.



Clave Candidata: Si de una Superclave no es posible obtener ningún subconjunto que sea a su vez Superclave, decimos que dicha Superclave es clave candidata, es decir, posibles claves que pueden ser únicas.

Clave primaria (Primary Key): También llamada llave primaria o clave principal. De todas las claves candidatas, el diseñador de la base de datos ha de escoger una, que se denominará clave principal o clave primaria. La clave primaria es un atributo o conjunto de ellos, que toman valores únicos y distintos para cada ocurrencia de entidad, identificándola unívocamente. No puede contener valores nulos.

Claves alternativas: Son el resto de claves candidatas que no han sido escogidas como clave primaria.

 **Autoevaluación**

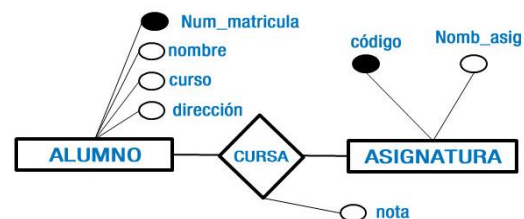
Sea la entidad TRABAJADOR, con los atributos nombre, apellido_1, apellido_2, dni, numero_afiliacion_ss, fecha_nacimiento y código_empresa. ¿Los atributos nombre, apellido_1 y apellido_2 podrían formar una clave candidata?

☐ Sí, y podrían ser elegidos para ser la clave primaria de TRABAJADOR.
☐ No, para esta entidad sólo el atributo dni será la clave primaria.
☒ No, si tenemos en cuenta que puede haber varios trabajadores con el mismo nombre y apellidos.

Efectivamente, los atributos dni y numero_afiliacion_ss, serían dos claves candidatas adecuadas. Si escogemos dni como clave primaria, numero_afiliacion_ss quedaría como clave alternativa.

4.3 Atributos de una relación.

Una relación puede también tener atributos que la describan. En el modelo E-R la representación de atributos asociados a relaciones es exactamente igual a la que utilizábamos para entidades. Podremos utilizar una elipse con el nombre del atributo en su interior, conectada con una línea a la relación, o bien, un círculo blanco conectado con una línea a la relación y junto a él, el nombre del atributo.



5. Relaciones.

Relación: Es una asociación entre diferentes entidades. En una relación no pueden aparecer dos veces relacionadas las mismas ocurrencias de entidad.

La representación gráfica en el modelo E-R corresponde a un rombo en cuyo interior se encuentra inscrito el nombre de la relación. Es interesante en algunas ocasiones que se indique el papel o rol que desempeña cada entidad en la relación. Dicho papel o rol son especialmente útiles en relaciones reflexivas.

- ✓ Grado de la relación.
- ✓ Cardinalidad de la relación.
- ✓ Cardinalidad de las entidades.

5.1 Grado de una relación.

Grado de una relación: Número de entidades que participan en una relación.

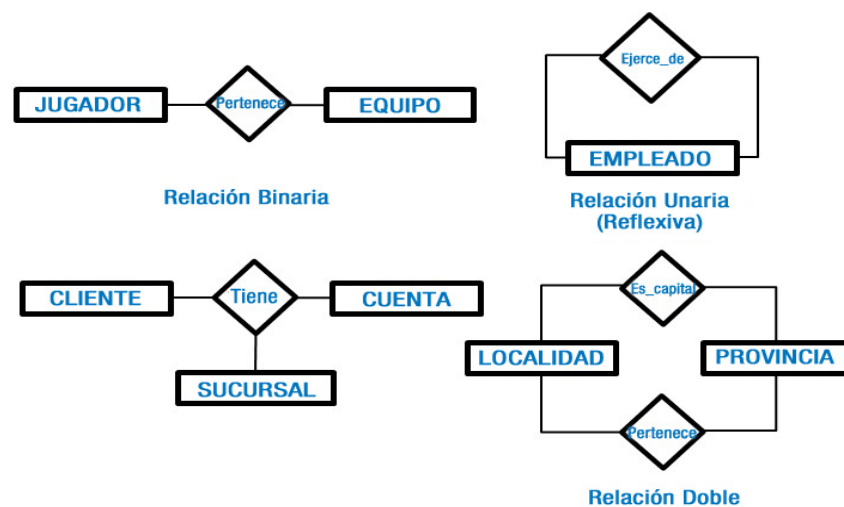
Relación Unaria o de grado 1: Es aquella relación en la que participa una única entidad, también llamadas reflexivas o recursivas.

Relación Binaria o de grado 2: Relación en la que participan dos entidades, tanto en una primera aproximación, como en los sucesivos refinamientos, el esquema conceptual de la base de datos buscará tener sólo este tipo de relaciones.

Relación Ternaria o de grado 3: Relación en la que participan tres entidades al mismo tiempo.

Relación N-aria o de grado n: Relación que involucra N entidades, no son usuales este tipo de relaciones y se deben simplificar hacia un grado menor.

Relación doble: Cuando dos entidades están relacionadas a través de dos relaciones, son muy complejas de manejar.



Autoevaluación

Rellena los huecos con los conceptos adecuados.

En la relación unaria que puedes ver en el gráfico anterior, un empleado puede ejercer el rol de **jefe** o el rol de **subordinado**.

[Reiniciar](#) [Mostrar las respuestas](#)

Tu puntuación es 2/2.

Al ser una relación reflexiva que relaciona una entidad consigo misma, un empleado puede ejercer el rol de **jefe** sobre uno o varios empleados y, a su vez, podría ejercer el rol de **subordinado** bajo las ordenes de un jefe.

5.2 Cardinalidad de relaciones.

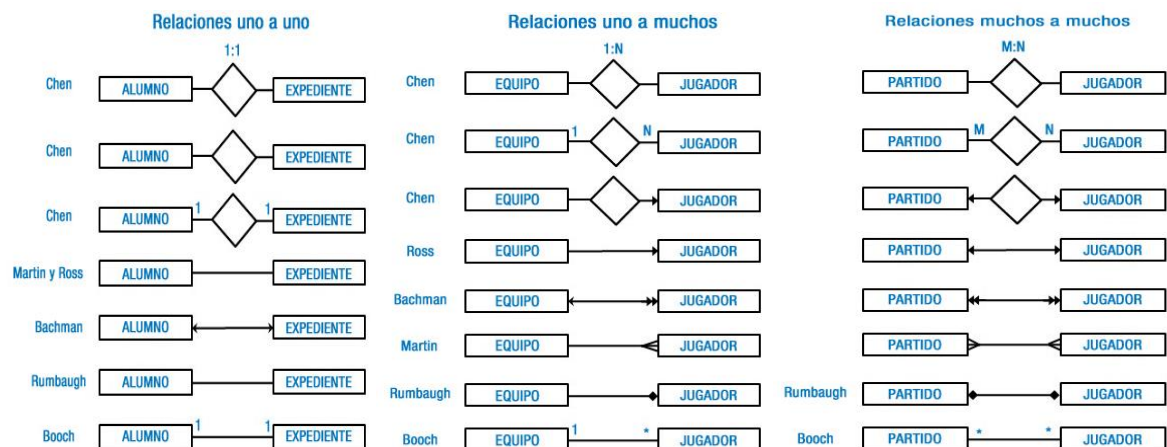
Cardinalidad de una relación: Es el número máximo de ocurrencias de cada entidad que pueden intervenir en una ocurrencia de relación, vendrá expresada siempre para relaciones entre dos entidades, pueden existir relaciones **uno a uno**, **uno a muchos**, **muchos a uno** y **muchos a muchos**.

Relaciones uno a uno (1:1): Sean las entidades A y B, una instancia u ocurrencia de la entidad A se relaciona únicamente con otra instancia de la entidad B y viceversa.

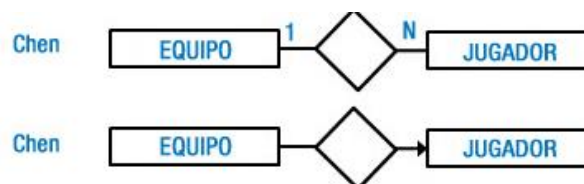
Relaciones uno a muchos (1:N): Sean las entidades A y B, una ocurrencia de la entidad A se relaciona con muchas ocurrencias de la entidad B y una ocurrencia de la entidad B sólo estará relacionada con una única ocurrencia de la entidad A.

Relaciones muchos a uno (N:1): Sean las entidades A y B, una ocurrencia de la entidad A está asociada con una única ocurrencia de la entidad B y un ejemplar de la entidad B está relacionado con muchas ocurrencias de la entidad A.

Relaciones muchos a muchos (M:N): Sean las entidades A y B, un ejemplar de la entidad A está relacionado con muchas ocurrencias de la entidad B y viceversa.



Nota: La flecha entre las relaciones uno a muchos (1:N) siempre va dirigida hacia la parte N o muchos. Mientras que en las muchos a muchos (M:N) llevan doble flecha.



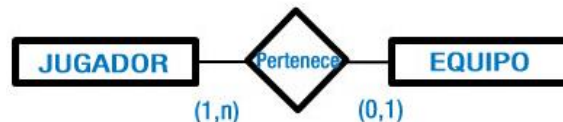
5.3 Cardinalidad de entidades.

Sean las entidades A y B, la participación de la entidad A en una relación es **obligatoria (total)** si la existencia de cada una de sus ocurrencias necesita como mínimo de una ocurrencia de la entidad B, en caso contrario, la participación es **opcional (parcial)**.

La cardinalidad de una entidad se representa con el número mínimo y máximo de correspondencias en las que puede tomar parte cada ejemplar de dicha entidad, reflejada entre paréntesis. (0,1), (1,1), (0,N), (1,N).

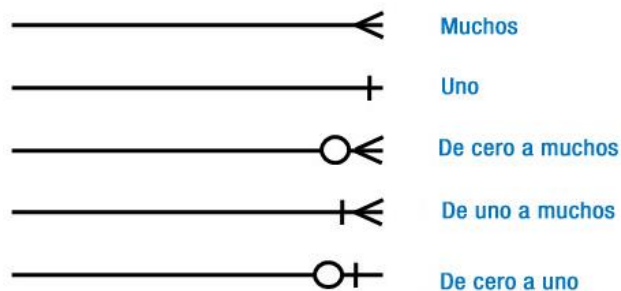
Cardinalidad mínima: Indica el número mínimo de asociaciones en las que aparecerá cada ocurrencia de la entidad, el valor cero se pondrá cuando la participación de la entidad sea opcional.

Cardinalidad máxima: Indica el número máximo de relaciones en las que puede aparecer cada ocurrencia de la entidad, puede tener un valor concreto mayor que uno (5) o puede tener muchos (N).



Nota: cuando se representa la cardinalidad de una entidad, el paréntesis y sus valores han de colocarse junto a la entidad con la que se relaciona. Es decir, en el lado opuesto.

Notación alternativa para representar cardinalidad de entidades



Autoevaluación

Supongamos que seguimos diseñando una base de datos para un sitio de juegos online. En un punto del proceso de diseño se ha de modelar el siguiente requisito: cada usuario registrado podrá crear las partidas que desee (a las que otros usuarios pueden unirse), pero una partida solo podrá estar creada por un único usuario. Un usuario podrá o no crear partidas. ¿Cuáles serían las etiquetas del tipo (cardinalidad mínima, cardinalidad máxima) que deberían ponerse junto a las entidades USUARIO y PARTIDA respectivamente, si éstas están asociadas por la relación CREAR (partida)?

- ☐ (1,N) y (0,N)
- ☐ (1,1) y (1,N)
- ☒ (1,1) y (0,N)

Efectivamente, con estas cardinalidades estarías indicando que un usuario puede crear varias partidas, o ninguna. Por otra parte, una partida deberá estar creada exclusivamente por un único usuario.

6. Simbología del modelo E-R.

Resumen básico de la simbología del modelo Entidad/Relación.

Resumen básico de la simbología del modelo Entidad/Relación

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	

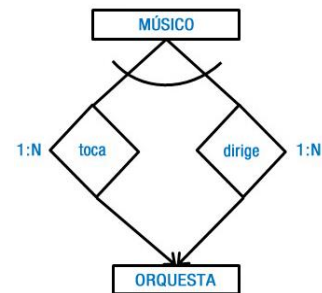
Notaciones del modelo Entidad/Relación		
	Atributo Derivado	
	Atributo opcional	
	Clave primaria	
	Clave alternativa	

7. El modelo E-R Extendido.

Hemos visto que a través del modelo Entidad/Relación se pueden modelar la gran mayoría de los requisitos que una base de datos debe cumplir. Pero existen algunos que ofrecen especial dificultad a la hora de representarlos a través de la simbología tradicional del modelo E-R. Para solucionar este problema, en el modelo Entidad/Relación Extendido se ha incorporado nuevas extensiones que permiten mejorar la capacidad para representar circunstancias especiales.

7.1 Restricciones en las relaciones.

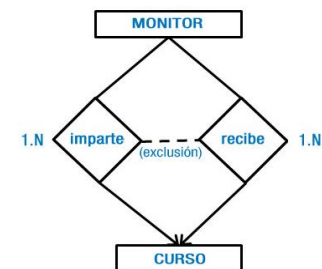
Restricción de exclusividad: Cuando existe una **entidad** que participa en dos o más relaciones y cada ocurrencia de dicha entidad sólo puede pertenecer a una de las relaciones únicamente, decimos que existe una restricción de exclusividad. Si la ocurrencia de entidad pertenece a una de las relaciones, no podrá formar parte de la otra, **o se produce una relación o se produce otra, pero nunca ambas a la vez.**



Supongamos que un músico puede dirigir una orquesta o tocar en ella, pero no puede hacer las dos cosas simultáneamente. Existirán, por lo tanto, dos relaciones **dirigen** y **toca**, entre las entidades **MÚSICO** y **ORQUESTA**, estableciéndose una relación de exclusividad entre ellas.

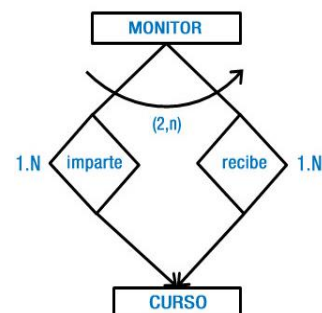
La representación gráfica en el modelo E-R Extendido de una restricción de exclusividad se realiza mediante un **arco** que engloba a todas aquellas relaciones que son exclusivas.

Restricción de exclusión: Se produce cuando las ocurrencias de las entidades sólo pueden asociarse utilizando una única relación. Se representa mediante una **línea discontinua entre las dos relaciones.**



Supongamos que un monitor puede impartir diferentes cursos de perfeccionamiento para monitores, y que éste puede a su vez recibirlos. Pero si un monitor imparte un determinado curso, no podrá estar recibéndolo simultáneamente y viceversa.

Restricción de inclusividad: Se aplica cuando es necesario modelar situaciones en las que para dos ocurrencias de entidad se asocien a través de una relación, tengan que haberlo estado antes a través de otra relación.

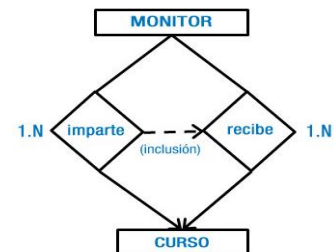


Supongamos que para que un monitor pueda impartir cursos de cocina sea necesario que reciba previamente dos cursos: nutrición y primeros auxilios. Como puedes ver, es posible que los cursos que el monitor deba recibir no tengan que ser los mismos que luego pueda impartir. Aplicando una restricción de inclusividad entre las relaciones **imparte** y **recibe**, estaremos

indicando que cualquier ocurrencia de la entidad **MONITOR** que participa en una de las relaciones (**imparte**) tiene que participar obligatoriamente en la otra (**recibe**).

Se representará mediante un **arco acabado en flecha**, que partirá desde la relación que ha de cumplirse primero hacia la otra relación. En dicho arco se indicará la cardinalidad mínima y máxima de dicha restricción de inclusividad.

Restricción de inclusión: En algunas ocasiones aplicar una restricción de inclusividad no representa totalmente la realidad de modelar, entonces se hace necesario aplicar una restricción de inclusión que es aún más fuerte. Se representa con **una flecha discontinua entre las dos relaciones**.



Inclusivo: Que incluye o tiene virtud y capacidad para incluir.

Inclusión: Acción y efecto de incluir.

Exclusivo: Único, solo, excluyendo a cualquier otro.

Exclusión: Acción y efecto de excluir.



Autoevaluación

Supongamos que hemos de modelar mediante el modelo Entidad/Relación Extendido el siguiente requerimiento de una base de datos: Para que un hombre se divorcie de una mujer, primero ha de haber estado casado con ella.

Las entidades participantes son MUJER y HOMBRE, que estarán asociadas a través de dos relaciones: se casa, se divorcia. No tendremos en cuenta la cardinalidad de ambas relaciones.

¿Qué tipo de restricción sobre las relaciones hemos de establecer en nuestro esquema para representar correctamente este requisito?

- ☐ Restricción de exclusividad.
- ☐ Restricción de inclusividad.
- ☒ Restricción de inclusión.

Efectivamente, este tipo de restricción establece la obligatoriedad de haber un casamiento para que pueda haber un divorcio y, además, las entidades que se relacionan a través de la relación se casa, deben ser las mismas que las participantes en se divorcia.

7.2 Generalización y especialización.

La segunda extensión incorporada en el modelo E-R Extendido se centra en nuevos tipos de relaciones que van a permitir modelar la realidad de una manera más fiel. Estos nuevos tipos de relación reciben el nombre de **jerarquías** y se basan en los conceptos de generalización, especialización y herencia.

Cuando estamos diseñando una base de datos puede que nos encontremos con conjuntos de entidades que posean características comunes, lo que permitiría crear un tipo de entidad de nivel más alto que englobase dichas características. Y a su vez, puede que necesitemos dividir un conjunto de entidades en diferentes subgrupos de entidades por tener éstas, **características diferenciadoras**.

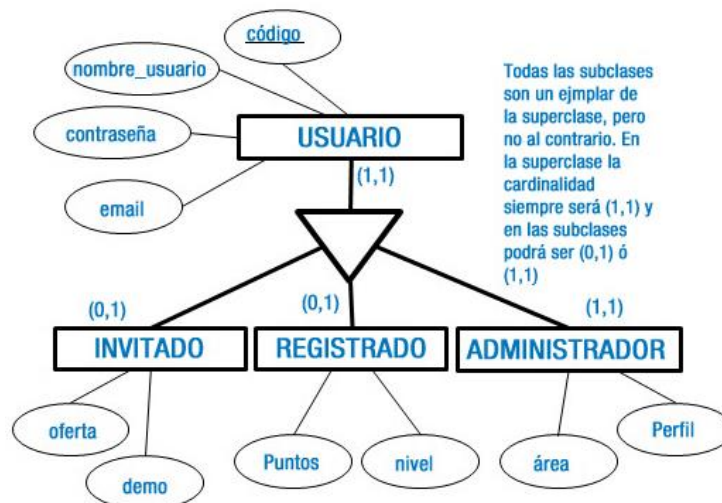
Éstas tienen un procesamiento ascendente/descendente, que permite expresar mediante la generalización la existencia de tipos de entidades en niveles superiores que engloban a niveles inferiores y viceversa.

A los conjuntos de entidades de nivel superior también se les denomina **superclase o supertipo**. A los conjuntos de entidades de nivel inferior se les denomina **subclase o subtipo**. Existirá por tanto la posibilidad de realizar una especialización de una **superclase en subclase**, y análogamente, establecer una generalización de las **subclases en superclases**. Las superclases tendrán características comunes mientras que las subclases tendrán características que las diferenciarán entre ellas.

Podemos identificar una generalización cuando encontremos una serie de atributos comunes a un conjunto de entidades, y otros atributos que sean específicos. Los atributos comunes conforman la superclase o supertipo y los atributos específicos la subclase o subtipo.

Las jerarquías se caracterizan por un concepto que hemos de tener en cuenta, **la herencia**. A través de la herencia los atributos de una superclase de entidad son heredados por las subclases, Si una superclase interviene en una relación, la subclase también lo hará.

Herencia: Existen varias notaciones, pero hemos de convenir que la relación que se establece entre una superclase de entidad y todos sus subtipos se expresa a través de las palabras **ES UN**, o en notación inglesa **IS A**, que correspondería con **ES UN TIPO DE**. Se representa mediante un triángulo invertido, sobre él quedará la entidad superclase y conectadas a él a través de líneas rectas, las subclases.



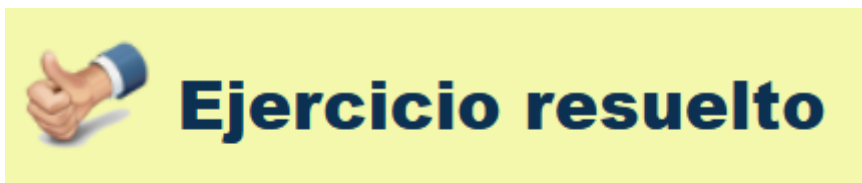
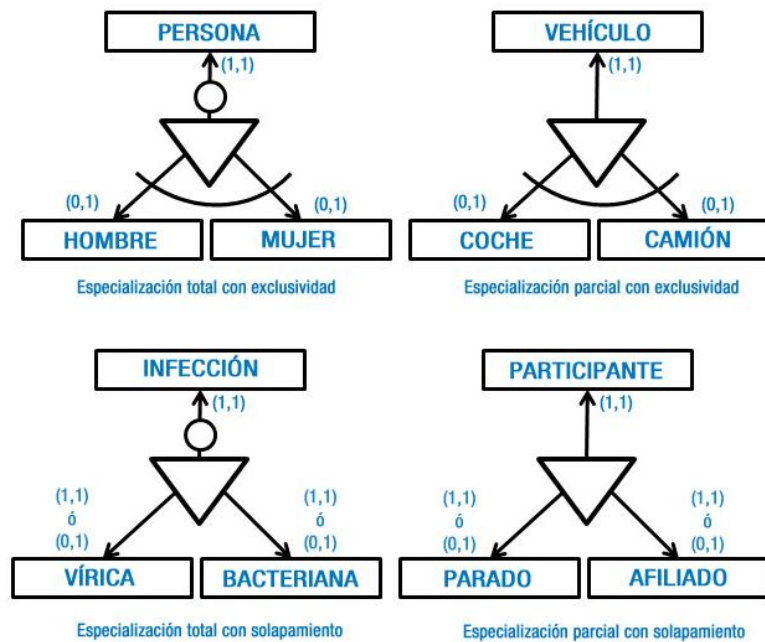
Una generalización/especialización podrá tener las siguientes restricciones semánticas:

Totalidad: una generalización/especialización será total si todo el ejemplar de la superclase pertenece a alguna de las subclases.

Parcialidad: una generalización/especialización será parcial si no todos los ejemplares de la superclase pertenecen a alguna de las subclases.

Solapamiento: una generalización/especialización presentará solapamiento si un mismo ejemplar de la superclase puede pertenecer a más de una subclase.

Exclusividad: una generalización/especialización presentará exclusividad si un mismo ejemplar de la superclase pertenece sólo a una subclase.



Supongamos la existencia de dos entidades TURISMO y CAMION. Los atributos de la entidad TURISMO son: Num_bastidor, Fecha_fab, precio y Num_puertas. Los atributos de la entidad CAMION son: Num_bastidor, Fecha_fab, precio, Num_ejes y Tonelaje.

Si analizamos ambas entidades existen algunos atributos comunes y otros que no. Por tanto, podremos establecer una jerarquía. Para ello, reuniremos los atributos comunes y los asociaremos a una nueva entidad superclase denominada VEHICULO. Las subclases TURISMO y CAMION, con sus atributos específicos, quedarán asociadas a la superclase VEHICULO mediante una jerarquía parcial con solapamiento. En el siguiente gráfico puedes apreciar la transformación.

Turismo (num_bastidor, fecha_fabricación, precio, num_puertas).

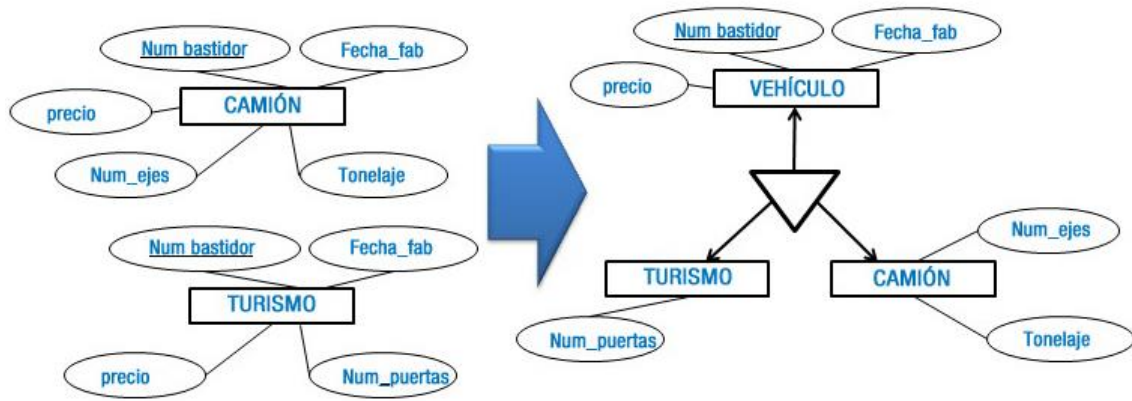
Camión (num_bastidor, fecha_fabricación, precio, num_ejes, tonelaje).

Superclase o supertipo

- Vehículo (num_bastidor, fecha_fabricación, precio).

Subclase o subtipo

- Turismo (num_puertas).
- Camión (num_ejes, tonelaje).

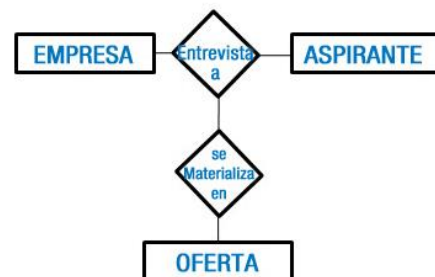


7.3 Agregación.

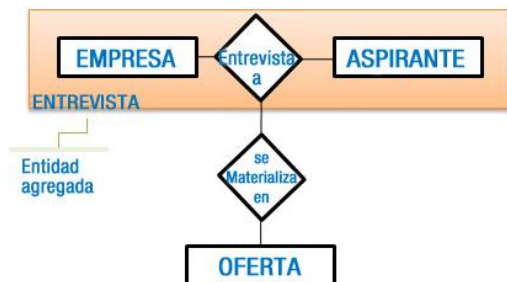
En modelo E-R no es posible representar relaciones entre relaciones. La agregación es una abstracción a través de la cual las relaciones se tratan como entidades de nivel más alto, siendo utilizada para expresar relaciones entre relaciones o entre entidades y relaciones.



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



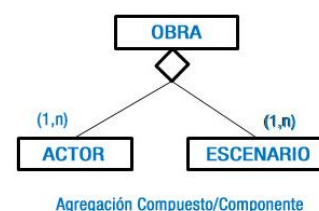
Solución 2: Errónea, porque en el modelo E/R no pueden establecerse relaciones entre varias relaciones



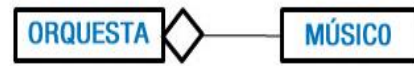
Solución 3: En el modelo E/R Extendido, puede crearse una entidad agregada llamada ENTREVISTA, compuesta por la relación "Entrevista a" que existe entre EMPRESA y ASPIRANTE. Entre esta nueva entidad y OFERTA si puede establecerse una relación "se materializa en"

Existen dos clases de agregaciones:

Compuesto/componente: esta abstracción permite representar que un todo o agregado se obtiene por la unión de diversas partes o componentes que pueden ser tipos de entidades distintas y que juegan diferentes roles en la agregación. Desempeñan distintos papeles en la agregación.



Miembro/colección: esta abstracción permite representar un todo o agregado como una colección de miembros, todos de un mismo tipo de entidad y todos jugando el mismo rol. Esta agregación puede incluir una restricción de orden de los miembros dentro de la colección, es decir, permite establecer un orden entre las partes.



Agregación Miembro/Colección



Autoevaluación

Si hemos de representar a través del modelo E/R Extendido los alumnos pertenecientes a una clase, podríamos utilizar una agregación del tipo Compuesto/Componente. ¿Verdadero o Falso?

Verdadero. ☐ Falso. ☒

Correcto. No sería correcto, al ser el alumnado un conjunto de elementos que representan el mismo rol en la relación, el tipo de agregación debería ser Miembro/Colección.

8. Elaboración de diagramas E/R

Los diagramas no siempre se crean del mismo modo y, en ocasiones, hay que retocarlos e incluso rehacerlos. A través de la resolución de diferentes problemas y la elaboración de múltiples diagramas, obtendrás la destreza necesaria para generar esquemas que garanticen una posterior y correcta conversión del modelo Entidad/Relación al modelo Relacional.

8.1 Identificación de entidades y relaciones.

Suponiendo que conocemos la simbología del modelo E-R y que entendemos su significado:

Identificación de entidades: es un proceso bastante intuitivo. Analizaremos la especificación de requerimientos en busca de nombres o sustantivos. Algunos autores indican que para poder considerarse como entidad se deben cumplir tres reglas:

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

Se le otorgarán nombres en **MAYÚSCULAS**.

Identificación de relaciones: analizaremos el documento de nuevo en busca de verbos o expresiones verbales que conecten entidades con otras. Suelen ser relaciones binarias (entidad con otra entidad) pero pueden darse el caso de relaciones unarias (entidad con su misma entidad).

Se le otorgarán nombres en **minúsculas**. Si la palabra es compuesta se recomienda usar guiones bajos para unirlos, ej: trabaja_para, es_supervisado.



Autoevaluación

Rellena los huecos con los conceptos adecuados.

Las entidades suelen localizarse en el documento de especificación de requerimientos a través de **sustantivos** y las relaciones a través de **verbos**. Pero hemos de tener cuidado, no siempre los **sustantivos** representarán entidades, pues podría tratarse de atributos.

[Reiniciar](#)

[Mostrar las respuestas](#)

Tu puntuación es 3/3.

En el documento de especificación de requerimientos buscaremos los sustantivos y los verbos para identificar entidades y relaciones respectivamente.

8.2 Identificación de atributos, claves y jerarquías.

Identificación de atributos: recorreremos el documento en búsqueda de nombres relativos a características, propiedades, identificadores o cualidades de entidades o relaciones. Se tendrá en cuenta si los atributos localizados son simples o compuestos, derivados o calculados y si algún atributo o conjunto se repite en varias entidades (Herencia).

Se recomienda recopilar la información de cada atributo:

- Nombre y descripción.
- Atributos simples que lo componen, si es atributo compuesto.
- Método de cálculo, si es atributo derivado o calculado.

Identificación de claves: de los atributos de una entidad se establecerán una o varias claves candidatas, escogiéndose una de ellas como clave o llave primaria de la entidad. Esta clave estará formada por uno o varios atributos que identificarán de manera unívoca cada ocurrencia de entidad.

Determinación de Jerarquías: si se identifica algún tipo de jerarquía, se deberá representar adecuadamente según el tipo de notación elegida, determinando si la jerarquía es total/parcial o exclusiva/con solapamiento.


8.3 Metodologías.

Metodología Descendente (top-down): se trata de partir de un esquema general e ir descomponiendo éste en niveles, cada uno de ellos con mayor número de detalles. Se parte de objetos muy abstractos, que se refinan paso a paso hasta llegar al esquema final.

Metodología Ascendente (bottom-up): inicialmente, se parte del nivel más bajo, los atributos. Se irán agrupando en entidades, para después ir creando las relaciones entre éstas y las posibles jerarquías hasta obtener un diagrama completo. Se parte de objetos atómicos que no pueden ser descompuestos y a continuación se obtienen abstracciones u objetos de mayor nivel de abstracción que forman el esquema.

Metodología Dentro-Fuera (inside-out): inicialmente se comienza a desarrollar el esquema en una parte del papel y a medida que se analiza la especificación de requerimientos, se va completando con entidades y relaciones hasta ocupar todo el documento.

Metodología Mixta: se emplea en problemas complejos. Se dividen los requerimientos en subconjuntos que serán analizados independientemente. Esta metodología utiliza las técnicas ascendente y descendente. Se aplicará la técnica descendente para dividir los requerimientos y en cada subconjunto de ellos, se aplicará la técnica ascendente.

 **Autoevaluación**

Rellena los huecos con los conceptos adecuados.
La metodología en la que se parte de un alto nivel de abstracción y que, tras un proceso de refinamiento sucesivo, se obtiene el esquema final se denomina:

☒ Metodología ☐ Ascendente ☒ Descendente

Tu puntuación es 2/2.

La **Metodología Descendente** permite iniciar el proceso a partir de un esquema general para conseguir un esquema detallado, a través de sucesivos procesos de refinamiento y descomposición.

8.4 Redundancia en diagramas E-R.

Redundancia: reproducción, reiteración, insistencia, reincidencia, reanudación. En base de datos hace referencia al almacenamiento de los mismos datos varias veces en diferentes lugares.

Puede provocar problemas como:

- Aumento de la carga de trabajo: al estar almacenado un dato en varios lugares, las operaciones de grabación o actualización de datos necesitan realizarse en varias ocasiones.
- Gasto extra de espacio de almacenamiento: ocupan un mayor espacio en disco, cuanto más grande sea la base de datos más patente será el problema.
- Inconsistencia: cuando repetimos datos y estos no tienen los mismos valores, es decir, se ha actualizado su valor en un lugar y en otro no, por lo que no se sabe que dato es el correcto.

Elementos que podrían presentar redundancia:

- Atributos redundantes cuyo contenido se calcula en función de otros.
- Varias entidades unidas circularmente o cíclica a través de varias relaciones, es lo que se conoce como **un ciclo**. A tener en cuenta con los ciclos:
 - o Que el significado de las relaciones que componen el ciclo sea el mismo.
 - o Que, si eliminamos la relación redundante, el significado del resto de relaciones es el mismo.
 - o Que, si la relación eliminada tenía atributos asociados, éstos puedan ser asignados a alguna entidad participante sin que se pierda su significado.

Pero hay que tener en cuenta que **NO** siempre que exista **un ciclo** estaremos antes una redundancia.

8.5 Propiedades deseables de un diagrama E-R.

Estas características o propiedades deseables se desglosan en:


Compleitud: será completo si es posible verificar que cada uno de los requerimientos está representado en dicho diagrama y viceversa, cada representación del diagrama tiene su equivalente en los requerimientos.

Corrección: será correcto si emplea de manera adecuada todos los elementos del modelo E-R.

- Corrección sintáctica: se reproducirá cuando no se produzcan representaciones erróneas en el diagrama.
- Corrección semántica: se producirá cuando las representaciones signifiquen exactamente lo que está estipulado e los requerimientos.

Posibles errores semánticos:

- La utilización de un atributo en lugar de una entidad.
 - El uso de una entidad en lugar de una relación.
 - Utilizar el mismo identificador para dos entidades o dos relaciones.
 - Indicar erróneamente alguna cardinalidad u omitirla.
- Minimalidad: será mínimo si se puede verificar que, al eliminar algún concepto presente en el diagrama, se pierde información, si es redundante no será mínimo.
- Sencillez: será sencillo si representa los requerimientos de manera fácil de comprender, sin artificios complejos.
- Legibilidad: será legible si puede interpretarse fácilmente, ésta propiedad tiene que ver mucho con aspectos estéticos del diagrama.
- Escalabilidad: será escalable si es capaz de incorporar posibles cambios derivados de nuevos requerimientos.



Autoevaluación

Si en un diagrama E/R asociamos un atributo a una entidad, pero este atributo debe asociarse realmente a una relación en la que interviene dicha entidad, estaríamos incumpliendo la propiedad de:

- ☐ Compleitud.
- ☒ Corrección semántica.
- ☐ Corrección sintáctica.

Efectivamente, la Corrección semántica se centra en analizar si cada representación del diagrama significa exactamente lo mismo que lo estipulado por el documento de especificación de requerimientos.

9. Paso del diagrama E-R al modelo relacional.

Tomando como referencia el esquema modificado/simplificado, se realizará el paso de éste al modelo de datos relacional. Esta transformación requerirá de la aplicación de determinadas reglas y condiciones que garanticen la equivalencia entre el esquema conceptual y el esquema lógico.

Como paso posterior, sobre la información del esquema lógico obtenido, será necesario llevar a cabo un proceso que permitirá diseñar de forma correcta la estructura lógica de los datos. Este proceso recibe el nombre de normalización, que se conforma como un conjunto de técnicas que permiten validar esquemas lógicos basados en el modelo relacional.

Resumiendo, un poco, simplificaremos nuestro diagrama E/R, lo transformaremos al modelo relacional, aplicaremos normalización y obtendremos lo que se conoce en el argot como el paso a tablas del esquema conceptual o, lo que es lo mismo, el esquema lógico. Desde ese momento, basándonos en este esquema, podremos llevarnos nuestra base de datos a cualquier SGBD basado en el modelo relacional e implementarla físicamente.

9.1 Simplificación previa de diagramas.

Existe un conjunto de procedimientos y normas que es necesario aplicar a nuestros diagramas E-R para que su transformación al modelo lógico basado en el modelo relacional, sea correcta y casi automática.

- Transformación de relaciones n-arias en binarias.
- Eliminación de relaciones cíclicas.
- Reducción a relaciones jerárquicas (uno a muchos).
- Conversión de entidades débiles en fuertes.

Veamos detalladamente cómo llevar a cabo las transformaciones:

Transformación de atributos compuestos: los atributos compuestos de una entidad han de ser descompuestos en los atributos simples por los que están formados.

Transformación de atributos multievaluados: este se ha de convertir en una entidad relacionada con la entidad de la que procede. Para esta nueva entidad se elegirá un nombre adecuado y tendrá un único atributo (el correspondiente al atributo múltiple). Ha de evaluarse las claves primarias.


Transformación a relaciones jerárquicas: se transforma las relaciones con cardinalidad muchos a muchos (M a N) en las relaciones con cardinalidad uno a muchos (1 a N). Si existiese un atributo asociado a la relación n-aria, quedaría asociado a la nueva entidad que se crea.

Transformación de relaciones cíclicas: se crea una nueva entidad cuya clave estará formada por dos atributos, que contendrán las claves de las ocurrencias relacionadas. Entre ambas entidades se establecen dos relaciones, cuya cardinalidad dependerá de la cardinalidad que tuviera la relación cíclica en un principio.

Transformación de relaciones ternarias: El tratamiento de las relaciones ternarias es similar al realizado para atributos asociados a relaciones, ya que una relación ternaria puede considerarse como una relación binaria a la que se le asocia una entidad. Por consiguiente, si en lugar de ser un conjunto de atributos los asociados a la relación es una entidad, se asociaría ésta mediante una nueva relación a la entidad resultante de eliminar la relación binaria.

Transformación de entidades débiles en fuerte: Para esta transformación sólo es necesario añadir a la entidad débil los atributos clave de la entidad que hace

posible la identificación de las ocurrencias. La clave de esta nueva entidad fuerte estará formada por los atributos clave de la que fuera entidad débil más los atributos adicionales.



Autoevaluación

Sea la entidad ALUMNADO que participa en la relación COLABORA con otra entidad llamada GRUPO_TRABAJO. Un alumno o alumna puede colaborar en varios grupos de trabajo simultáneamente y, a su vez, en un grupo de trabajo pueden colaborar un número indeterminado de alumnos. Se necesita registrar los días en los que el alumnado colabora con cada grupo de trabajo, para ello se asocia a la relación COLABORA un atributo denominado fecha_colaboración. Este atributo registrará en qué fecha un determinado alumno/a ha colaborado en un determinado grupo de trabajo.

¿Si tuvieras que hacer la transformación de esta parte del esquema conceptual para eliminar la relación M a N COLABORA, dónde colocarías el atributo fecha_colaboración?

- ☐ En la entidad ALUMNADO, ya que en esta entidad es donde se almacenan todos los datos asociados al alumnado. Si consultamos el alumno o alumna, sabremos cuándo a colaborado en un grupo.
- ☒ En una nueva entidad que es combinación de ALUMNADO y GRUPO_TRABAJO, a la que podríamos llamar ALUMNADO_GRUPO.
- ☐ En la entidad GRUPO_TRABAJO.

Correcto, al transformar la relación M a N, se crean dos relaciones 1 a N entre ALUMNADO-ALUMNADO_GRUPO Y GRUPO_TRABAJO-ALUMNADO_GRUPO, siendo ALUMNADO_GRUPO una nueva entidad que tendrá por claves las claves primarias de ALUMNADO Y GRUPO_TRABAJO, recibiendo como atributo el atributo que estaba asociado a la relación COLABORA. Para cada par ALUMNADO/GRUPO_TRABAJO tendremos registrado cuándo se realizó la colaboración.

10. Paso del diagrama E-R al Modelo Relacional.

Si se ha llevado a cabo el proceso preparatorio de nuestro esquema conceptual o diagrama E/R, según se ha indicado en epígrafes anteriores, dispondremos de un Esquema Conceptual Modificado (ECM) en el que sólo existirán exclusivamente entidades fuertes con sus atributos y relaciones jerárquicas (1 a N). Pues bien, la aplicación del modelo de datos Relacional es automática, para ello se deben tener en cuenta las siguientes cuestiones:

- Toda entidad se transforma en una tabla.
- Todo atributo se transforma en columnas dentro de una tabla.
- El atributo clave de la entidad se convierte en clave primaria de la tabla y se representará subrayado en la tabla.
- Cada entidad débil generará una tabla que incluirá todos sus atributos, añadiéndose a ésta los atributos que son clave primaria de la entidad fuerte con la que esté relacionada. Estos atributos añadidos se constituyen como clave foránea que referencia a la entidad fuerte. Seguidamente, se escogerá una clave primaria para la tabla creada.
- Las relaciones uno a uno podrán generar una nueva tabla o propagar la clave en función de la cardinalidad de las entidades.
- Las relaciones uno a muchos podrán generar una nueva tabla o propagar la clave.
- Las relaciones reflexivas o cíclicas podrán generar una o varias tablas en función de la cardinalidad de la relación.
- Las jerarquías generarán la reunión, eliminación o creación de relaciones 1 a 1.
- Las relaciones muchos a muchos se transforman en una tabla que tendrá como clave primaria las claves primarias de las entidades que asocia.

No obstante, si en el proceso de generación del diagrama E/R o esquema conceptual hemos aplicado correctamente las reglas de simplificación de diagramas, nuestro Esquema Conceptual Modificado nos permitirá el paso a

tablas teniendo en cuenta sólo las transformaciones asociadas a entidades, relaciones 1 a N, 1 a 1 y Jerarquías.

El paso a tablas de dicho esquema sería el siguiente:

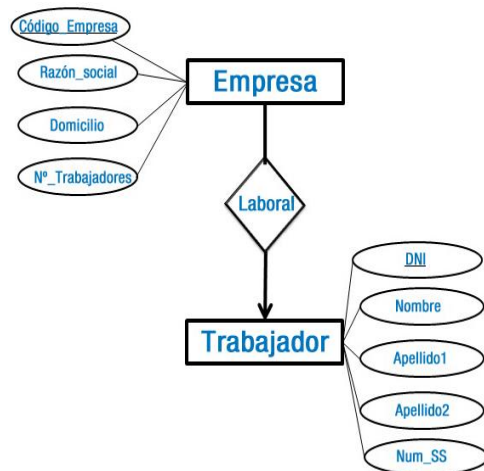
EMPRESA (Código_empresa, razón_social, domicilio, Nº_Trabajadores)

TRABAJADOR (DNI, Nombre, Apellido1, Apellido2, Num_SS)

Para materializar la relación de uno a muchos LABORAL, se incluye una clave foránea en la entidad TRABAJADOR, que referencia a la entidad EMPRESA, quedando:

EMPRESA (Código_empresa, razón_social, domicilio, Nº_Trabajadores)

TRABAJADOR (DNI, Nombre, Apellido1, Apellido2, Num_SS, Código_empresa)



11. Normalización de modelos relacionales.

Normalización: Proceso que consiste en imponer a las tablas del modelo Relacional una serie de restricciones a través de un conjunto de transformaciones consecutivas. Este proceso garantizará que las tablas contienen los atributos necesarios y suficientes para describir la realidad de la entidad que representan, permitiendo separar aquellos atributos que por su contenido podrían generar la creación de otra tabla.

Pero esta técnica no ha de utilizarse para el diseño de la base de datos, sino como un proceso de refinamiento que debe aplicarse después de lo que conocemos como “paso a tablas”, o lo que formalmente se denomina traducción del esquema conceptual al esquema lógico. Este proceso de refinamiento conseguirá:

- Suprimir dependencias erróneas entre atributos.
- Optimizar los procesos de inserción, modificación y borrado en la base de datos.

Es un proceso que se realiza en 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. Existen varias formas normales: **Primera, Segunda, Tercera, Boyce-Codd, Cuarta, Quinta y Dominio-Clave**. Como hemos indicado, el paso de una forma normal a otra es consecutivo, si no se satisface una determinada forma normal no puede pasarse al análisis de la siguiente. Según vamos avanzando en la normalización, los requisitos a cumplir serán cada vez más restrictivos, lo que hará que nuestro esquema relacional sea cada vez más robusto.

Como norma general, para garantizar que no existan problemas en la actualización de datos, es recomendable aplicar el proceso de normalización **hasta Tercera Forma Normal o incluso hasta Forma Normal de Boyce-Codd**. En los siguientes epígrafes se describen las características y requisitos de cada una de las formas normales.

11.1 Tipos de dependencias.

Dependencia Funcional: Dados los atributos A y B, se dice que B depende funcionalmente de A, sí, y solo sí, para cada valor de A sólo puede existir un valor de B. La dependencia funcional siempre se establece entre atributos de una misma tabla. El atributo A se denomina determinante, ya que A determina el valor de B. Para representar esta dependencia funcional utilizamos la siguiente notación: $A \rightarrow B$. Hay que indicar que A y B podrían ser un solo atributo o un conjunto de ellos.

Dependencia Funcional Completa: Dados los atributos A1, A2, ...Ak y B, se dice que B depende funcionalmente de forma completa de A1, A2, ...Ak, si y solo si B depende funcionalmente del conjunto de atributos A1, A2, ...Ak, pero no de ninguno de sus posibles subconjuntos.

Dependencia Transitiva: Dados tres atributos A, B y C, se dice que existe una dependencia transitiva entre A y C, si B depende funcionalmente de A y C depende funcionalmente de B. A, B y C podrían ser un solo atributo o un conjunto de ellos.



Ejercicio resuelto

Dadas las siguientes tablas:

EMPLEADO(DNI, Nombre, Dirección, Localidad, Cod_Localidad, Nombre_hijo, Edad_hijo)
LIBRO(Título_libro, Num_ejemplar, Autor, Editorial, Precio)

Resuelve las siguientes cuestiones:

- Indica qué atributos presentan una dependencia funcional de la clave primaria de la tabla **EMPLEADO**.
- Indica qué atributos presentan una dependencia funcional completa en la tabla **LIBRO**.
- Indica qué atributos presentan una dependencia transitiva en la tabla **EMPLEADO**.

Apartado a)

Los atributos Nombre, y Dirección dependen funcionalmente de DNI, ya que para un DNI específico sólo podrá haber un nombre y una dirección. Pero los atributos Nombre_hijo y Edad_hijo no presentan esa dependencia funcional de DNI, ya que para un DNI específico podríamos tener varios valores diferentes en esos atributos. (Consideraremos para este ejemplo que todos los empleados registrados en esta base de datos tienen nombres distintos). Expresemos estas dependencias funcionales mediante su notación:

$DNI \rightarrow Nombre$

$DNI \rightarrow Dirección$

Apartado b)

Los atributos Editorial y Precio dependen funcionalmente del conjunto de atributos que forman la clave primaria de la tabla, pero no dependen de Título_libro o de Num_ejemplar por separado, por lo que presentan una dependencia funcional completa de la clave. El atributo Autor depende funcionalmente sólo y exclusivamente de Título_libro, por lo que no presenta una dependencia funcional completa de los atributos que forman la clave.

Apartado c)

Los atributos Cod_Localidad y Localidad dependen funcionalmente de DNI, pero entre Cod_Localidad y Localidad existe otra dependencia funcional. Por tanto, se establece que Localidad depende funcionalmente de Cod_Localidad, y a su vez, Cod_Localidad depende funcionalmente de DNI. Con lo que podemos afirmar que existe una dependencia transitiva entre Localidad y DNI. Si lo representamos con la notación asociada a las dependencias funcionales, quedaría: $DNI \rightarrow Cod_Localidad \rightarrow Localidad$.

11.2 Formas normales.

Una vez conocidos los conceptos sobre los que se basa el proceso de normalización, se han de llevar a cabo una serie de etapas consecutivas en las que se aplicarán las propiedades de cada una de las formas normales definidas por Codd. A continuación, se exponen los requisitos a cumplir por las tablas de nuestra base de datos según la forma normal que apliquemos.

1ª Forma Normal: Una tabla está en Primera Forma Normal (1FN o FN1) sí, y sólo sí, todos los atributos de la misma contienen valores atómicos, es decir, no hay grupos repetitivos. Dicho de otra forma, estará en 1FN si los atributos no claves, dependen funcionalmente de la clave. ¿Cómo se normaliza a Primera Forma Normal?

Se crea, a partir de la tabla inicial, una nueva tabla cuyos atributos son los que presentan dependencia funcional de la clave primaria. La clave de esta tabla será la misma clave primaria de la tabla inicial. Esta tabla ya estará en 1FN.

Con los atributos restantes se crea otra tabla y se elige entre ellos uno que será la clave primaria de dicha tabla. Comprobaremos si esta segunda tabla está en 1FN. Si es así, la tabla inicial ya estará normalizada a 1FN y el proceso termina. Si no está en 1FN, tomaremos la segunda tabla como tabla inicial y repetiremos el proceso.

2ª Forma Normal: Una tabla está en Segunda Forma Normal (2FN o FN2) sí, y sólo sí, está en 1FN y, además, todos los atributos que no pertenecen a la clave dependen funcionalmente de forma completa de ella. Es obvio que una tabla que esté en 1FN y cuya clave esté compuesta por un único atributo, estará en 2FN. ¿Cómo se normaliza a Segunda Forma Normal?

Se crea, a partir de la tabla inicial, una nueva tabla con los atributos que dependen funcionalmente de forma completa de la clave. La clave de esta tabla será la misma clave primaria de la tabla inicial. Esta tabla ya estará en 2FN.

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. Se comprueba si esta tabla está en 2FN. Si es así, la tabla inicial ya está normalizada y el proceso termina. Si no está en 2FN, tomamos esta segunda tabla como tabla inicial y repetiremos el proceso.

3ª Forma Normal: Una tabla está en Tercera Forma Normal (3FN o FN3) sí, y sólo sí, está en 2FN y, además, cada atributo que no está en la clave primaria no depende transitivamente de la clave primaria. ¿Cómo se normaliza a Tercera Forma Normal?

Se crea, a partir de la tabla inicial, una nueva tabla con los atributos que no poseen dependencias transitivas de la clave primaria. Esta tabla ya estará en 3FN.

Con los atributos restantes, se crea otra tabla con los dos atributos no clave que intervienen en la dependencia transitiva, y se elige uno de ellos como clave primaria, si cumple los requisitos para ello. Se comprueba si esta tabla está en 3FN. Si es así, la tabla inicial ya está normalizada y el proceso termina. Si no está en 3FN, tomamos esta segunda tabla como tabla inicial y repetiremos el proceso.

Forma Normal de Boyce Codd: Una tabla está en Forma Normal de Boyce-Codd (FNBC o BCFN) sí, y sólo sí, está en 3FN y todo determinante es una 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 la que todos sus atributos forman parte de la clave primaria, estarán en FNBC. Por tanto, si encontramos un determinante que no es clave candidata, la tabla no estará en FNBC. Esta redundancia suele ocurrir por una mala elección de la clave. Para normalizar a FNBC tendremos que descomponer la tabla inicial en dos, siendo cuidadosos para evitar la pérdida de información en dicha descomposición.

Otras formas normales

Existen también la Cuarta Forma Normal (4FN o FN4), Quinta Forma Normal (5FN o FN5) y Forma Normal de Dominio-Clave (DKFN), aunque se ha recomendado normalizar hasta 3FN o FNBC. La 4FN se basa en el concepto de Dependencias Multivaluadas, la 5FN en las Dependencias de Join o de reunión y la DKFN en las restricciones impuestas sobre los dominios y las claves.

[Cuarta Forma Normal.](#)

[Quinta Forma Normal.](#)

[Forma Normal Dominio-Clave.](#)



Ejercicio resuelto

Sea la siguiente tabla:

COMPRAS (cod_compra, cod_prod, nomb_prod, fecha, cantidad, precio, fecha_rec, cod_prov, nomb_prov, tfno).

Se pide normalizarla hasta FNBC.

Comprobamos 1FN:

La tabla COMPRAS está en 1FN ya que todos sus atributos son atómicos y todos los atributos no clave dependen funcionalmente de la clave.

Comprobamos 2FN:

Nos preguntaremos ¿Todo atributo depende de todo el conjunto de atributos que forman la clave primaria, o sólo de parte? Como vemos, existen atributos que dependen sólo de una parte de la clave, por lo que esta tabla no está en 2FN.

Veamos las dependencias:

$\text{cod_prod} \rightarrow \text{nomb_prod}$, y cod_prod es parte de la clave primaria.

Al no estar en 2FN, hemos de descomponer la tabla COMPRAS en:

COMPRA1 (cod_compra, cod_prod, fecha, cantidad, precio, fecha_rec, cod_prov, nomb_prov, tfno).

PRODUCTO (cod_prod, nomb_prod).

Una vez hecha esta descomposición, ambas tablas están en 2FN. Todos los atributos no clave dependen de toda la clave primaria.

Comprobamos 3FN:

PRODUCTO está en 3FN, ya que por el número de atributos que tiene no puede tener dependencias transitivas.

¿COMPRA1 está en 3FN? Hemos de preguntarnos si existen dependencias transitivas entre atributos no clave.

Veamos las dependencias:

$\text{cod_prov} \rightarrow \text{nomb_prov}$

$\text{cod_prov} \rightarrow \text{tfno}$

(siendo cod_prov el código del proveedor y nomb_prov el nombre del proveedor)

COMPRA1 no está en 3FN porque existen dependencias transitivas entre atributos no clave, por tanto, hemos de descomponer:

COMPRA2 (cod_compra, cod_prod, fecha, cantidad, precio, fecha_rec,
cod_prov)
PROVEEDOR (cod_prov, nomb_prov, tfno)

Comprobamos FNBC:

PRODUCTO está en FNBC, ya que está en 3FN y todo determinante es clave candidata.

COMPRA2 está en FNBC, ya que está en 3FN y todo determinante es clave candidata.

PROVEEDOR está en FNBC, ya que está en 3FN y todo determinante es clave candidata.

La tabla inicial COMPRAS queda normalizada hasta FNBC del siguiente modo:

PRODUCTO (cod_prod, nomb_prod)

COMPRA2 (cod_compra, cod_prod, fecha, cantidad, precio, fecha_rec,
cod_prov)

PROVEEDOR (cod_prov, nomb_prov, tfno)