

# Tema 2.1.

## Diseño lógico de Bases de Datos.

### Modelo

### Entidad - Relación

1. Objetivo de la unidad	3
2. Introducción	3
3. Metodología de diseño de bases de datos	4
4. Modelos de datos	4
5. El modelo entidad-relación	5
5.1 Elementos del modelo Entidad-Relación extendido	6
5.2 Entidad	6
5.3 Relación (interrelación)	8
5.4 Atributo	9
5.5 Identificador o Clave	10
5.6 Jerarquía de generalización o especificación	10
5.6.1 Tipos de relaciones jerárquicas (generalización/especialización)	12
6. Metodología de diseño conceptual	13
6.1 Identificar las entidades	14
6.2 Identificar las relaciones	14
6.3 Identificar los atributos y asociarlos a entidades y relaciones	15
6.4 Determinar los dominios de los atributos	16
6.5 Determinar los identificadores	16
6.6 Determinar las jerarquías de generalización	16
6.7 Dibujar el diagrama entidad-relación	17
6.8 Revisar el esquema conceptual local con el usuario	17

## 1. OBJETIVO DE LA UNIDAD

- **Diseñar modelos lógicos normalizados interpretando diagramas entidad/relación**
- Comprender la importancia del diseño de datos y de usar relaciones normalizadas.
- Describir la sintaxis de un lenguaje gráfico de representación de diseño conceptual de datos.
- Obtener diseños conceptuales y lógicos normalizados para representar datos y relaciones en un sistema de datos relacional.
- En un supuesto práctico planteado sobre la representación de datos y relaciones:
  - Representar gráficamente el diseño conceptual de datos y relaciones.
- Transformar el modelo E(R en modelo relacional.
- Conocer el proceso de normalización de tablas.
- Saber transformar las tablas a 3ª FN.

## 2. INTRODUCCIÓN

**El diseño de bases de datos es el proceso por el que se determina la organización de una base de datos, incluidos su estructura, contenido y las aplicaciones que se han de desarrollar.**

Durante mucho tiempo, el diseño de bases de datos fue considerado una tarea para expertos: más un arte que una ciencia. Sin embargo, se ha progresado mucho en el diseño de bases de datos y éste se considera ahora una disciplina estable, con métodos y técnicas propios.

A finales de la década de 1960, cuando las bases de datos entraron por primera vez en el mercado del software, los diseñadores de bases de datos actuaban como artesanos, con herramientas muy primitivas: diagramas de bloques y estructuras de registros eran los formatos comunes para las especificaciones, y el diseño de bases de datos se confundía frecuentemente con la implantación de las bases de datos. Esta situación ahora ha cambiado: los métodos y modelos de diseño de bases de datos han evolucionado paralelamente con el progreso de la tecnología en los sistemas de bases de datos. Se ha entrado en la era de los sistemas relacionales de bases de datos, que ofrecen poderosos lenguajes de consulta, herramientas para el desarrollo de aplicaciones e interfaces amables con los usuarios. La tecnología de bases de datos cuenta ya con un marco teórico, que incluye la teoría relacional de datos, procesamiento y optimización de consultas, control de concurrencia, gestión de transacciones y recuperación, etc.

Según ha avanzado la tecnología de bases de datos, así se han desarrollado las metodologías y técnicas de diseño. Se ha alcanzado un consenso, por ejemplo, sobre la descomposición del proceso de diseño en fases, sobre los principales objetivos de cada fase y sobre las técnicas para conseguir estos objetivos.

Desafortunadamente, las metodologías de diseño de bases de datos no son muy populares; la mayoría de las organizaciones y de los diseñadores individuales confía muy poco en las metodologías para llevar a cabo el diseño y esto se considera, con frecuencia, una de las

principales causas de fracaso en el desarrollo de los sistemas de información. Debido a la falta de enfoques estructurados para el diseño de bases de datos, a menudo se subestiman el tiempo o los recursos necesarios para un proyecto de bases de datos, las bases de datos son inadecuadas o ineficientes en relación a las demandas de la aplicación, la documentación es limitada y el mantenimiento es difícil.

Muchos de estos problemas se deben a la falta de una claridad que permita entender la naturaleza exacta de los datos, a un nivel conceptual y abstracto. En muchos casos, los datos se describen desde el comienzo del proyecto en términos de las estructuras finales de almacenamiento; no se da peso a un entendimiento de las propiedades estructurales de los datos que sea independiente de los detalles de la realización.

### 3. METODOLOGÍA DE DISEÑO DE BASES DE DATOS

El diseño de una base de datos es un proceso complejo que abarca decisiones a muy distintos niveles. La complejidad se controla mejor si se descompone el problema en subproblemas y se resuelve cada uno de estos subproblemas independientemente, utilizando técnicas específicas. Así, el diseño de una base de datos se descompone en diseño conceptual, diseño lógico y diseño físico.

**El diseño lógico** parte de las especificaciones de requisitos de usuario y su resultado es **el esquema lógico de la base de datos**. Este esquema es una descripción de alto nivel de la estructura de la base de datos, independientemente del SGBD que se vaya a utilizar para manipularla. **El objetivo es describir el contenido de información de la base de datos y no las estructuras de almacenamiento que se necesitarán para manejar esta información.**

### 4. MODELOS DE DATOS

Un **modelo de dato** es un conjunto de herramientas que permiten describir los datos, sus relaciones, las restricciones de seguridad a aplicar y la terminología a utilizar.

Los modelos deben representar la realidad, por lo que deben poseer las siguientes cualidades:

- **Expresividad:** deben tener suficientes conceptos para expresar perfectamente la realidad.
- **Simplicidad:** deben ser simples para que los esquemas sean fáciles de entender.
- **Formalidad:** todos los conceptos deben tener una interpretación única, precisa y bien definida.

## 5. EL MODELO ENTIDAD-RELACIÓN

El modelo Entidad-Relación es el modelo conceptual más utilizado para el diseño conceptual de bases de datos. Fue introducido por Peter Chen en 1976. El modelo entidad-relación pretende 'visualizar' los objetos que pertenecen a la Base de Datos como entidades las cuales tienen unos atributos y se vinculan mediante relaciones.

Mediante una serie de procedimientos se puede pasar del modelo E-R a otros, como por ejemplo el modelo relacional.

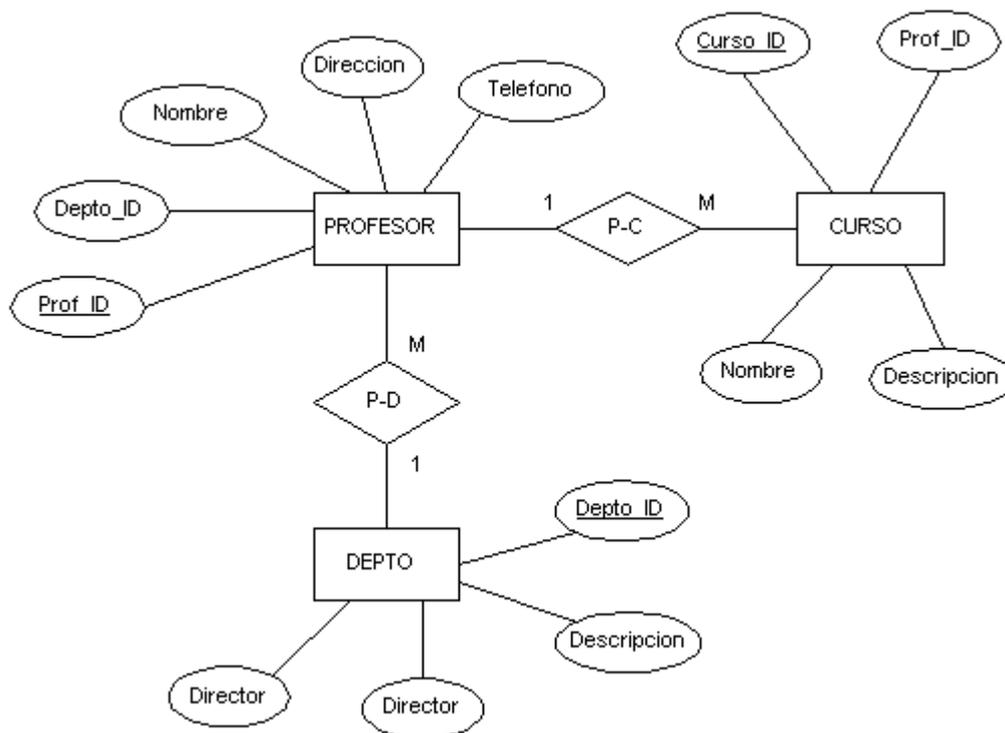
El **modelado Entidad-Relación** es una técnica para el modelado de datos utilizando diagramas entidad relación. No es la única técnica pero sí la más utilizada. Brevemente consiste en los siguientes pasos:

1. Se parte de una descripción textual del problema o sistema de información a automatizar (los requisitos).
2. Se hace una lista de los sustantivos y verbos que aparecen.
3. Los sustantivos son posibles entidades o atributos.
4. Los verbos son posibles relaciones.
5. Analizando las frases se determina la cardinalidad de las relaciones y otros detalles.
6. Se elabora el diagrama (o diagramas) entidad-relación.
7. Se completa el modelo con listas de atributos y una descripción de otras restricciones que no se pueden reflejar en el diagrama.

El modelado de datos no finaliza con el uso de esta técnica. Son necesarias otras técnicas para lograr un modelo directamente implementable en una base de datos. Brevemente:

- Transformación de relaciones múltiples en binarias.
- Normalización de una base de datos de relaciones (algunas relaciones pueden transformarse en atributos y viceversa).
- Conversión en tablas (en caso de utilizar una base de datos relacional).

Ejemplo de modelo E/R:



Este modelo diferenciamos tres **entidades**, Profesor, Curso y Departamento y varias **relaciones**. Un departamento tiene muchos profesores y un profesor puede dar muchos cursos. Para cada una de las entidades existe una propiedad que las identifica únicamente y que se corresponde con la clave primaria (en este caso clave subrogada). Las entidades, además, tienen otras propiedades que las describen (los atributos no clave).

## 5.1 Elementos del modelo Entidad-Relación extendido

Originalmente, el modelo entidad-relación sólo incluía los conceptos de entidad, relación y atributo. Más tarde, se añadieron otros conceptos en lo que se ha denominado modelo entidad-relación extendido.

## 5.2 Entidad

Cualquier **tipo de objeto o concepto sobre el que se recoge información relevante para el sistema**: cosa, persona, concepto abstracto o suceso. Por ejemplo: coches, casas, empleados, clientes, empresas, oficios, diseños de productos, conciertos, excursiones, etc. Las entidades se representan gráficamente mediante **rectángulos** y su nombre aparece en el interior. Un nombre de entidad sólo puede aparecer una vez en el esquema conceptual.

Hay dos tipos de entidades: **fuertes y débiles**. Una entidad débil es una entidad cuya existencia depende de la existencia de otra entidad. Una entidad fuerte es una entidad que no es débil.

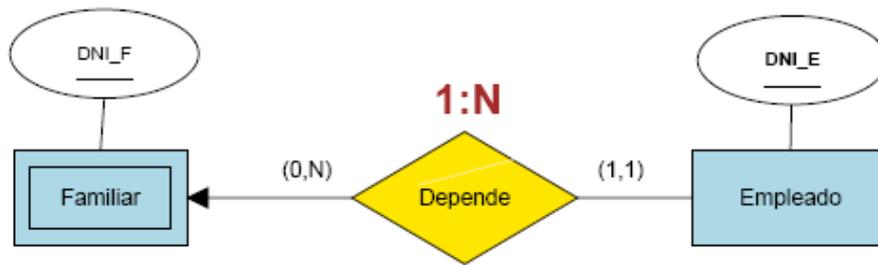
Las entidades fuertes se representan mediante un rectángulo y las débiles mediante un rectángulo doble, en el interior se escribe su nombre.



Las relaciones se clasifican en fuertes y en débiles, según estén asociando dos entidades fuertes, o una entidad débil con una entidad fuerte respectivamente

### Dependencia en Existencia

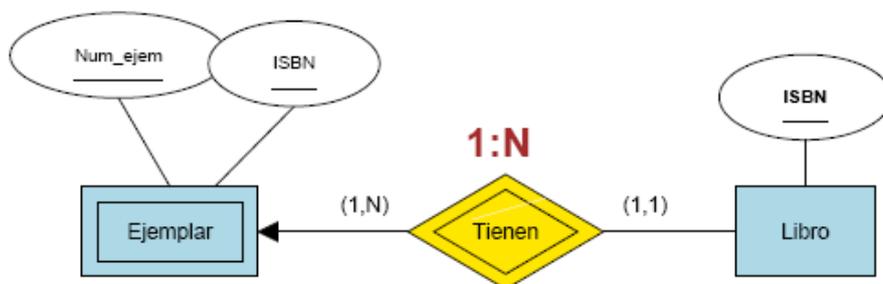
Cuando las ocurrencias de una entidad débil, no pueden existir si desaparece la ocurrencia de la entidad fuerte de la cual dependen



### Dependencia en Identificación

Cuando además de cumplirse la condición anterior, la entidad débil no se pueden identificar únicamente mediante los atributos propios de la misma y debe añadir la clave de la entidad fuerte de la cual dependen

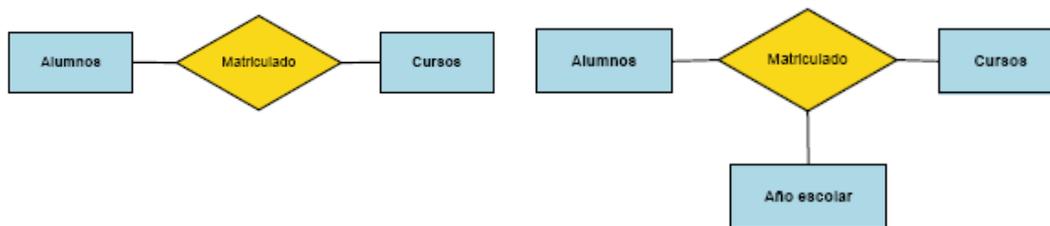
Se indica con un rombo doble:



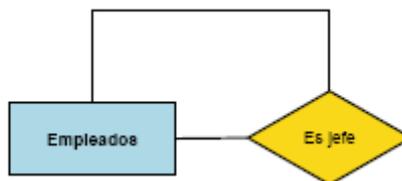
## 5.3 Relación (interrelación)

Es una **correspondencia o asociación entre dos o más entidades**. Cada relación tiene un nombre que describe su función. Las relaciones se representan gráficamente mediante rombos y su nombre aparece en el interior.

Las entidades que están involucradas en una determinada relación se denominan entidades participantes. El número de participantes en una relación es lo que se denomina **grado** de la relación. Por lo tanto, una relación en la que participan dos entidades es una relación binaria; si son tres las entidades participantes, la relación es ternaria; etc.

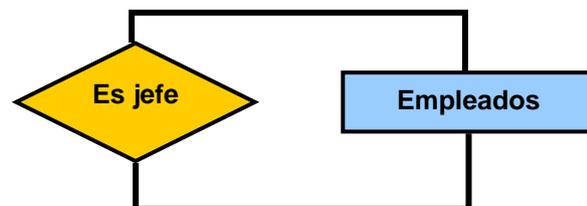


Una relación **recursiva** o reflexiva es una relación donde la misma entidad participa más de una vez en la relación.



### Grado de una relación

**Grado 1:** Sólo participa una entidad



**Grado 2 o binarias:** Participan 2 entidades



**Grado 3 o ternarias:** Participan 3 entidades



## Cardinalidad de las entidades

La **cardinalidad** con la que una entidad participa en una relación especifica el número mínimo y el número máximo de correspondencias en las que puede tomar parte cada ocurrencia de dicha entidad. La participación de una entidad en una relación es obligatoria (total) si la existencia de cada una de sus ocurrencias requiere la existencia de, al menos, una ocurrencia de la otra entidad participante. Si no, la participación es opcional (parcial).

Indica el número de relaciones en las que una entidad puede aparecer. Se anota en términos de:



**Cardinalidad mínima.** Indica el número mínimo de asociaciones en las que aparecerá cada ejemplar de la entidad (el valor que se anota es de cero o uno, aunque tenga una cardinalidad mínima de más de uno, se indica sólo un uno)

**Cardinalidad máxima.** Indica el número máximo de relaciones en las que puede aparecer cada ejemplar de la entidad. Puede ser uno, otro valor concreto mayor que uno (tres por ejemplo) o muchos (se representa con n)

En los esquemas entidad / relación la cardinalidad se puede indicar de muchas formas. Quizá la más completa (y la que se utiliza en este documento es ésta) consiste en anotar en los extremos la cardinalidad máxima y mínima de cada entidad en la relación.

## Tipo de correspondencia

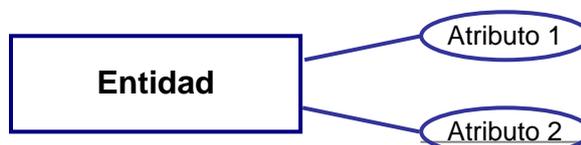
Indica el nº máximo de entidades de un tipo que se puede relacionar por cada entidad del otro tipo asociadas por la relación, lo obtenemos a partir de las cardinalidades.

Las más frecuentes son:

- **1:1** Por cada elemento de una entidad sólo aparece uno de la otra
- **1:N** Por cada elemento de una entidad aparece un nº indeterminado de elementos de la otra
- **N:M** Lo anterior ocurre en ambos sentidos

## 5.4 Atributo

Es una **característica de interés** o un hecho sobre una entidad o sobre una relación. Los atributos representan las **propiedades básicas de las entidades y de las relaciones**. Toda la información extensiva es portada por los atributos. Gráficamente, se representan mediante bolitas que cuelgan de las entidades o relaciones a las que pertenecen.



Cada atributo tiene un conjunto de valores asociados denominado dominio. El dominio define todos los valores posibles que puede tomar un atributo. Puede haber varios atributos definidos sobre un mismo dominio.

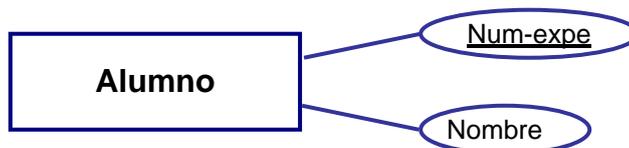
## 5.5 Identificador o Clave

Un identificador de una entidad es un **atributo o conjunto de atributos que determina de modo único cada ocurrencia de esa entidad**. Un identificador de una entidad debe cumplir dos condiciones:

1. No pueden existir dos ocurrencias de la entidad con el mismo valor del identificador.
2. Si se omite cualquier atributo del identificador, la condición anterior deja de cumplirse.

Toda entidad tiene al menos una clave y puede tener varios claves alternativas. Las relaciones no tienen identificadores.

La clave puede estar formada por uno o varios atributos y se representa con ese/os atributos subrayados



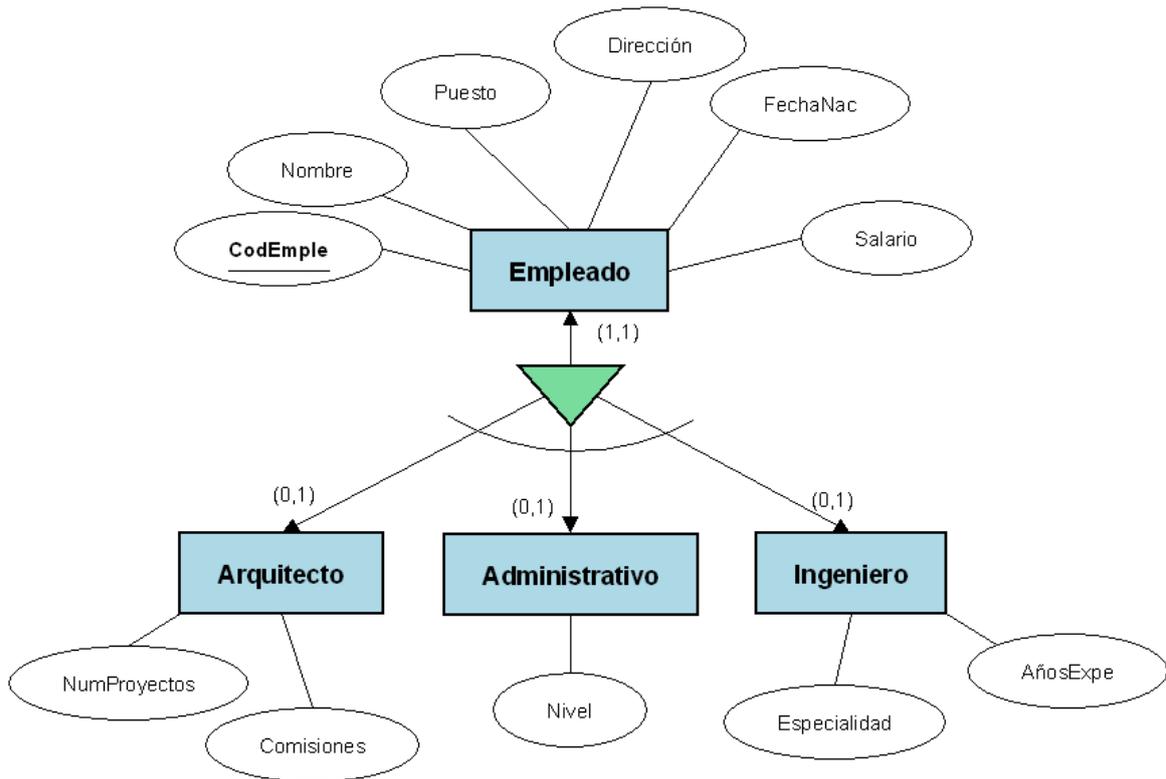
## 5.6 Jerarquía de generalización o especificación

Una entidad E es una generalización de un grupo de entidades E1, E2, ... En, si cada ocurrencia de cada una de esas entidades es también una ocurrencia de E. Todas las propiedades de la entidad genérica E son heredadas por las subentidades.

Se utilizan para unificar entidades agrupándolas en una entidad más general (generalización) o bien para dividir una entidad general en entidades más específicas (especificación).

Se habla de **generalización** si inicialmente partimos de una serie de entidades que al estudiarlas en detalle descubrimos que todas ellas pertenecen al mismo conjunto. En la generalización las entidades son totalmente heterogéneas, es decir, los atributos son diferentes. La entidad general se llama **superentidad** las otras se denominan **subentidades**. La superentidad normalmente tiene una clave principal distinta de las subentidades.

La **especialización** ocurre cuando partimos de una entidad que podemos dividir en subentidades para detallar atributos que varían en las mismas. Comparten clave con la superentidad y los atributos de la superclase se heredan en las subclases.

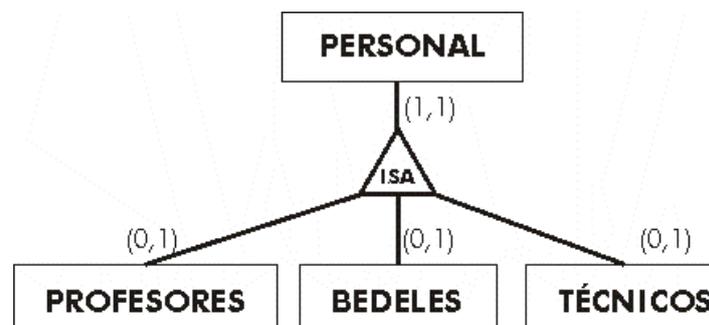


La entidad general empleado se ha dividido en tres subentidades. La cuestión de si es generalización o especialización no suele ser excesivamente importante, sí lo es la cardinalidad.

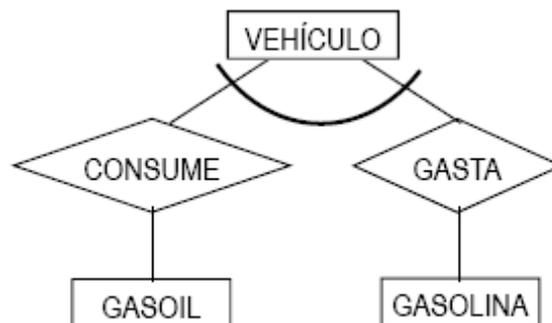
Las generalizaciones pueden ser:

- Solapadas o inclusivas
- Exclusivas

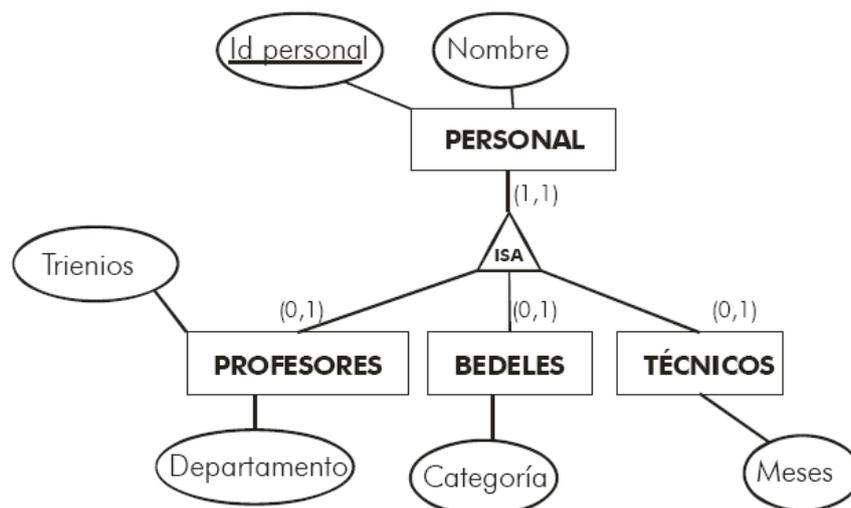
**Solapadas o inclusivas.** Cuando que un ejemplar de la superentidad puede relacionarse con más de una subentidad (el personal puede ser profesor y bedel). Ocurren cuando **no** hay dibujado un arco de exclusividad



**Exclusivas.** Indican que un ejemplar de la superentidad sólo puede relacionarse con uno de la subentidad (el vehículo consume gasoil o gasolina). Ocurren cuando hay dibujado un arco de exclusividad.



Con atributos el esquema sería:



En la especialización anterior los profesores, bedeles y técnicos heredan el atributo id personal y el nombre, el resto son atributos propios sólo de cada entidad (trienios pertenece sólo a los profesores, en este ejemplo)

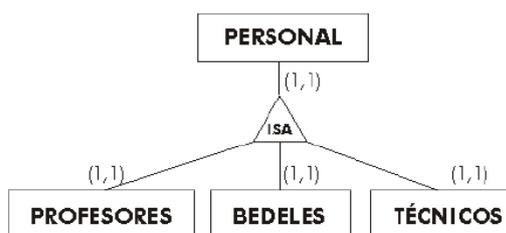
### 5.6.1 Tipos de relaciones jerárquicas (generalización/especialización)

En base a lo comentado anteriormente, podemos tener los siguientes tipos de relaciones:

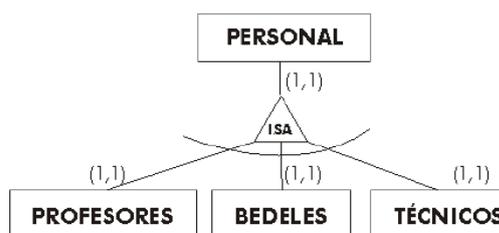
- Relaciones de **jerarquía solapada**. Indican que un ejemplar de la superentidad puede relacionarse con más de una subentidad (el personal puede ser profesor y bedel). Ocurren cuando no hay dibujado un arco de exclusividad.
- Relaciones de **jerarquía exclusiva**. Indican que un ejemplar de la superentidad sólo puede relacionarse con uno de la subentidad (el personal no puede ser profesor y bedel). Ocurren cuando hay dibujado un arco de exclusividad.

- Relaciones de **jerarquía parcial**. Indican que hay ejemplares de la superentidad que no se relacionan con ninguna subentidad (hay personal que no es ni profesor, no bedel ni técnico). Se indican con cardinalidad mínima de cero en la superentidad.
- Relaciones de **jerarquía total**. Indican que todos los ejemplares de la superentidad se relacionan con alguna subentidad (no hay personal que no sea ni profesor, ni bedel ni técnico). Se indican con cardinalidad mínima de uno en alguna superentidad.

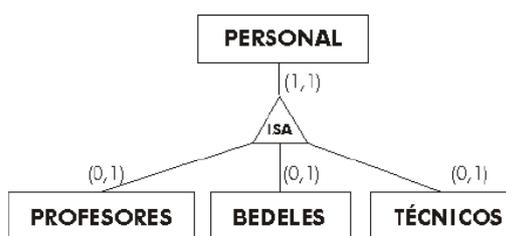
Los posibles ejemplos de relaciones ISA atendiendo a la cardinalidad son los siguientes:



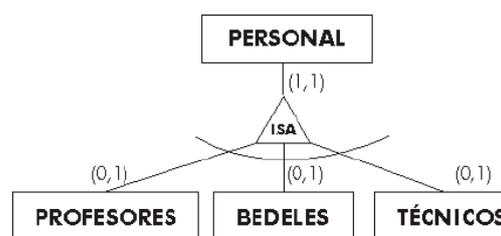
Relación ISA solapada total



Relación ISA exclusiva total



Relación ISA solapada parcial



Relación ISA exclusiva parcial

## 6. METODOLOGÍA DE DISEÑO CONCEPTUAL

El primer paso en el diseño de una base de datos es la producción del modelo lógico. Pueden construirse varios esquemas lógicos, cada uno para representar las distintas visiones que los usuarios tienen de la información. Cada una de estas visiones suelen corresponder a las diferentes áreas funcionales de la empresa como, por ejemplo, producción, ventas, recursos humanos, etc.

Las tareas a realizar en el diseño lógico son las siguientes:

1. Identificar las entidades.
2. Identificar las relaciones.
3. Identificar los atributos y asociarlos a entidades y relaciones.
4. Determinar los dominios de los atributos.
5. Determinar los identificadores.
6. Determinar las jerarquías de generalización (si las hay).
7. Dibujar el diagrama entidad-relación.
8. Revisar el esquema lógico con el usuario.

## 6.1 Identificar las entidades

En primer lugar hay que **definir los principales objetos que interesan al usuario**. Estos objetos serán las entidades. Una forma de identificar las entidades es examinar las especificaciones de requisitos de usuario. En estas especificaciones se buscan los nombres o los sintagmas nominales que se mencionan (por ejemplo: número de empleado, nombre de empleado, número de inmueble, dirección del inmueble, alquiler, número de habitaciones). También se buscan objetos importantes como personas, lugares o conceptos de interés, excluyendo aquellos nombres que sólo son propiedades de otros objetos.

Otra forma de identificar las entidades es buscar aquellos objetos que existen por sí mismos. Por ejemplo, empleado es una entidad porque los empleados existen, sepamos o no sus nombres, direcciones y teléfonos. Siempre que sea posible, el usuario debe colaborar en la identificación de las entidades.

A veces, es difícil identificar las entidades por la forma en que aparecen en las especificaciones de requisitos. Los usuarios, a veces, hablan utilizando ejemplos o analogías. En lugar de hablar de empleados en general, hablan de personas concretas, o bien, hablan de los puestos que ocupan esas personas.

No siempre es obvio saber si un objeto es una entidad, una relación o un atributo. Por ejemplo ¿cómo se podría clasificar matrimonio? Pues de cualquiera de las tres formas. El análisis es subjetivo, por lo que distintos diseñadores pueden hacer distintas interpretaciones, aunque todas igualmente válidas. Todo depende de la opinión y la experiencia de cada uno. Los diseñadores de bases de datos deben tener una visión selectiva y clasificar las cosas que observan dentro del contexto de la empresa u organización. A partir de unas especificaciones de usuario es posible que no se pueda deducir un conjunto único de entidades, pero después de varias iteraciones del proceso de análisis, se llegará a obtener un conjunto de entidades que sean adecuadas para el sistema que se ha de construir.

Conforme se van identificando las entidades, se les dan nombres que tengan un significado y que sean obvias para el usuario. Cuando sea posible, se debe anotar también el número aproximado de ocurrencias de cada entidad.

## 6.2 Identificar las relaciones

Una vez definidas las entidades, se deben definir las relaciones existentes entre ellas. Del mismo modo que para identificar las entidades se buscaban nombres en las especificaciones de requisitos, para identificar las relaciones se suelen buscar las **expresiones verbales** (por ejemplo: oficina tiene empleados, empleado gestiona inmueble, cliente visita inmueble). Si las especificaciones de requisitos reflejan estas relaciones es porque son importantes para la empresa y, por lo tanto, se deben reflejar en el esquema conceptual. Pero sólo interesan las relaciones que son **necesarias**.

La mayoría de las relaciones son binarias (entre dos entidades), pero no hay que olvidar que también puede haber relaciones en las que participen más de dos entidades, así como relaciones recursivas.

Es muy importante repasar las especificaciones para comprobar que todas las relaciones, explícitas o implícitas, se han encontrado.

Una vez identificadas todas las relaciones, hay que determinar **la cardinalidad mínima y máxima** con la que participa cada entidad en cada una de ellas. De este modo, el esquema representa de un modo más explícito la semántica de las relaciones. La cardinalidad es un tipo de restricción que se utiliza para comprobar y mantener la calidad de los datos. Estas restricciones son aserciones sobre las entidades que se pueden aplicar cuando se actualiza la base de datos para determinar si las actualizaciones violan o no las reglas establecidas sobre la semántica de los datos.

Conforme se van identificando las relaciones, se les van asignando nombres que tengan significado para el usuario.

### 6.3 Identificar los atributos y asociarlos a entidades y relaciones

Al igual que con las entidades, se buscan nombres en las especificaciones de requisitos. Son atributos los nombres que identifican **propiedades, cualidades, identificadores o características de entidades o relaciones**.

Lo más sencillo es preguntarse, para cada entidad y cada relación, ¿qué información se quiere saber de ...? La respuesta a esta pregunta se debe encontrar en las especificaciones de requisitos. Pero, en ocasiones, será necesario preguntar a los usuarios para que aclaren los requisitos.

También se deben identificar los **atributos derivados o calculados**, que son aquellos cuyo valor se puede calcular a partir de los valores de otros atributos. Por ejemplo, el número de empleados de cada oficina, la edad de los empleados o el número de inmuebles que gestiona cada empleado. Algunos diseñadores no representan los atributos derivados en los esquemas conceptuales. Si se hace, se debe indicar claramente que el atributo es derivado y a partir de qué atributos se obtiene su valor.

Cuando se están identificando los atributos, se puede descubrir alguna entidad que no se ha identificado previamente, por lo que hay que volver al principio introduciendo esta entidad y viendo si se relaciona con otras entidades.

Hay que tener mucho cuidado cuando parece que un mismo atributo se debe asociar a varias entidades. Esto puede ser por una de las siguientes causas:

- Se han identificado varias entidades, como director, supervisor y administrativo, cuando, de hecho, pueden representarse como una sola entidad denominada empleado. En este caso, se puede escoger entre introducir una **jerarquía de generalización**, o dejar las entidades que representan cada uno de los puestos de empleado.

- Se ha identificado una relación entre entidades. En este caso, se debe asociar el atributo a una sola de las entidades y hay que asegurarse de que la relación ya se había identificado previamente. Si no es así, se debe para recoger la nueva relación.

Conforme se van identificando los atributos, se les asignan nombres que tengan significado para el usuario. De cada atributo se debe anotar la siguiente información:

- Nombre y descripción del atributo.
- Alias o sinónimos por los que se conoce al atributo.
- Tipo de dato y longitud.
- Valores por defecto del atributo (si se especifican).
- Si el atributo siempre va a tener un valor (si admite o no nulos).
- Si el atributo es compuesto y, en su caso, qué atributos simples lo forman.
- Si el atributo es derivado y, en su caso, cómo se calcula su valor.

## 6.4 Determinar los dominios de los atributos

El dominio de un atributo es el conjunto de valores que puede tomar el atributo.

Un esquema conceptual está completo si incluye los dominios de cada atributo: los valores permitidos para cada atributo, su tamaño y su formato. También se puede incluir información adicional sobre los dominios como, por ejemplo, las operaciones que se pueden realizar sobre cada atributo, qué atributos pueden compararse entre sí o qué atributos pueden combinarse con otros. Aunque sería muy interesante que el sistema final respetara todas estas indicaciones sobre los dominios, esto es todavía una línea abierta de investigación.

## 6.5 Determinar los identificadores

Cada entidad tiene al menos un identificador. En este paso, se trata de encontrar todos los identificadores de cada una de las entidades. Los identificadores pueden ser simples o compuestos. De cada entidad se escogerá uno de los identificadores como **clave primaria** en la fase del diseño lógico.

Cuando se determinan los identificadores es fácil darse cuenta de si una entidad es fuerte o débil. Si una entidad tiene al menos un identificador, es fuerte (también llamada padre, propietaria o dominante). Si una entidad no tiene atributos que le sirvan de identificador, es débil (otras denominaciones son hijo, dependiente o subordinada).

## 6.6 Determinar las jerarquías de generalización

En este paso hay que observar las entidades que se han identificado hasta el momento. Hay que ver si es necesario reflejar las diferencias entre distintas ocurrencias de una entidad, con lo que surgirán nuevas subentidades de esta entidad genérica; o bien, si hay entidades que

tienen características en común y que realmente son subentidades de una nueva entidad genérica.

En cada jerarquía hay que determinar si es total o parcial y exclusiva o solapada.

## **6.7 Dibujar el diagrama entidad-relación**

Una vez identificados todos los conceptos, se puede dibujar el diagrama entidad-relación correspondiente a una de las vistas de los usuarios. Se obtiene así un esquema conceptual local.

## **6.8 Revisar el esquema conceptual local con el usuario**

Antes de dar por finalizada la fase del diseño conceptual, se debe revisar el esquema conceptual local con el usuario. Este esquema está formado por el diagrama entidad-relación y toda la documentación que describe el esquema. Si se encuentra alguna anomalía, hay que corregirla haciendo los cambios oportunos, por lo que posiblemente haya que repetir alguno de los pasos anteriores. Este proceso debe repetirse hasta que se esté seguro de que el esquema conceptual es una fiel representación de la parte de la empresa que se está tratando de modelar.