

Diseño y Administración de Bases de Datos

Unidad III

Análisis

Habíamos mencionado en la unidad anterior que el objetivo del análisis es crear el **Modelo de Datos** que es la visión del mundo real orientado hacia un determinado objetivo.

En esta etapa entonces, se documenta la visión del analista sobre el circuito de datos reales bajo análisis, destacando cuales son los datos relevantes y la estructura lógica de los conjuntos de datos.

Diccionario de Datos

Un **diccionario de datos (DDD)** es un conjunto de metadatos que describe las características lógicas y puntuales de los datos que se van a utilizar en el sistema que se modela, incluyendo nombre, descripción, alias, contenido y organización de todos los elementos que componen el flujo de datos.

Contiene la lista de **todos** los elementos que forman parte del modelo de datos de todo el sistema con su estructura formal y detalles. Los elementos más importantes son flujos de datos, almacenes de datos y procesos.

El objetivo primario es que los analistas puedan conocer cuántos caracteres abarca un determinado dato o qué otros nombres recibe un elemento en distintas partes del sistema, o dónde se utiliza.

Construir un DDD es una de las labores más tediosas y largas del analista de sistemas. Pero también es una de las más importantes: sin un diccionario formal que defina el significado de los términos, no se puede esperar precisión.

Objetivos

- Para manejar los detalles en sistemas muy grandes donde existen enormes cantidades de datos, y aún en los sistemas mas chicos cuando la colección de datos es muy confusa.

Una particularidad de los sistemas que sufren cambios continuos, es que se vuelve de muy difícil manejo en todos los detalles. Por eso se registra la información, ya sea sobre hoja de papel o usando procesadores de texto. En factorías de software se suelen utilizar sistemas de diccionarios de datos diseñados específicamente para el análisis y diseño de software.

- Para asignar un solo significado a cada uno de los elementos y actividades del sistema. Los diccionarios de datos proporcionan asistencia para asegurar significados comunes para los elementos y actividades del sistema y registrando detalles adicionales relacionados con el flujo de datos en el sistema, de tal manera que todo pueda localizarse con rapidez.
- Para documentar las características del sistema, incluyendo partes o componentes así como también sus características que los distinguen. También documenta bajo que circunstancias se lleva a cabo cada proceso y con que frecuencia ocurre, produciendo una comprensión mas completa. Una vez que las características están articuladas y registradas, todos los participantes en el proyecto tendrán una fuente común de información con respecto al sistema.
- Para facilitar el análisis de los detalles con la finalidad de evaluar las características y determinar donde efectuar cambios en el sistema. Determina si son necesarias nuevas características o si están en orden los cambios de cualquier tipo.

Por ejemplo:

- Naturaleza de las transacciones: las actividades de la empresa que se llevan a cabo mientras se emplea el sistema.
 - Preguntas: solicitudes para la recuperación o procesamiento de información para generar una respuesta específica.
 - Archivos y bases de datos: detalles de las transacciones y registros maestros que son de interés para la organización.
 - Capacidad del sistema: Habilidad del sistema para aceptar, procesar y almacenar transacciones y datos
- Ayuda a la localización de errores y omisiones en el sistema.

Contenido de un registro del DDD

El diccionario tiene dos tipos de descripciones para el flujo de datos del sistema: los elementos datos y las estructuras de datos.

Elemento dato:

Son los bloques básicos para todos los demás datos del sistema. Se agrupan para formar una estructura de datos. El registro de un elemento dato contiene:

Un nombre: debe ser único, para distinguir un dato de otro.

Descripción: indica lo que representa en el sistema.

Alias: lista de nombres alternativos para el dato, dependiendo de quien usa este dato.

Longitud: cantidad de espacio necesario para cada dato.

Tipo: indica el tipo de dato más apropiado. Si los valores de los datos están restringidos a un intervalo específico, también debe estar documentado como una entrada del diccionario.

Notación: Un código que determina como se compone un dato válido. Esta entrada tiene una notación de tipo algebraica con un conjunto de símbolos que veremos a continuación.

Estructura de datos:

Es un grupo de elementos datos, que definen un objeto dentro del flujo de datos.

Notación

Los analistas usan símbolos especiales con la finalidad de no usar demasiada cantidad de texto para la descripción de las relaciones entre datos y mostrar con claridad las relaciones estructurales. En algunos casos se emplean términos diferentes para describir la misma entidad (alias) estos se representan con un signo igual (=) que vincula los datos.

1. Un signo de igual (=) significa “está compuesto de”.

2. Un signo de más (+) significa “y”.

3. Las llaves { } indican elementos repetidos, también llamados grupos repetidos o tablas. Puede haber uno o varios elementos repetidos dentro del grupo. El grupo repetido puede tener condiciones, tales como límites superior e inferior para la cantidad de repeticiones. se puede anteponer un valor numérico indicando que tiene cumplir una cantidad mínima de iteraciones y también se puede agregar un valor numérico al final indicando la máxima cantidad de iteraciones permitidas. por ejemplo 1{6 para al menos 1 dato y hasta 6 como máximo, 5{5 para exactamente 5 datos.

4. Los corchetes [] representan una situación disyuntiva. Puede estar presente un elemento u otro, pero no ambos. Los elementos listados entre corchetes son mutuamente excluyentes, y se separan mediante barras (|).

5. Los paréntesis () representan un elemento opcional. Los elementos opcionales pueden ser dejados en blanco en las pantallas de captura, y pueden contener espacios o ceros para los campos numéricos en las estructuras de archivo.

6. Arroba @ (o una definición subrayada) identifica un campo clave para una colección de datos.

7. Una frase entre asteriscos es un comentario (* *).

8. La palabra reservada **Alias** se utiliza para indicar que un dato es un sinónimo de otro previamente definido.

9. Los contenidos literales se especifican entre comillas dobles " ".

10. Un rango de datos secuencial se especifica indicando el límite inferior del rango, un guión medio - y luego el límite superior del rango.

Por ejemplo:

Nombre = Título + Primer-nombre + Apellido-paterno + Apellido-materno

Título = ["Sr." | "Sra." | "Dr." | "Ing."]

Primer-nombre = {carácter}30

Apellido-paterno = {carácter}30

Apellido-materno = (carácter)30

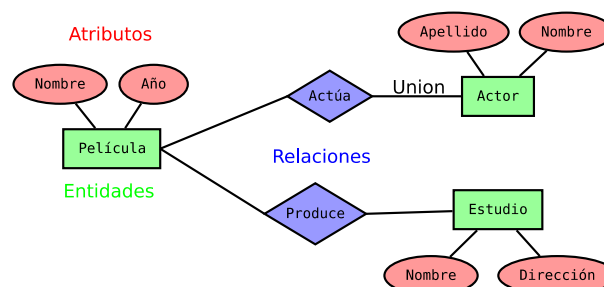
Carácter = [A-Z|a-z| '|']

DNI = 8{[0-9]}8

Modelo Entidad-Relación

Introducción a los Diagramas Entidad - Relación

El modelo entidad relación (E-R) es un modelo de datos que fue desarrollado por **Peter Chen** en 1976 para facilitar el diseño de las bases de datos, ya que permite la creación de un esquema que representa la estructura global lógica de la base de datos. Es un modelo semántico porque representa el significado de los datos. El modelo E-R emplea cuatro conceptos básicos: conjuntos de entidades, conjuntos de relaciones, uniones y atributos. Cada uno de estos cuatro conceptos se representa mediante un gráfico distinto: rectángulos para los conjuntos de entidades, rombos para los conjuntos de relaciones, óvalos para los atributos y líneas para las uniones.



Conjunto de Entidades

Una entidad es una cosa u objeto del mundo real que es diferente de los demás objetos o cosas. Una entidad posee un conjunto de propiedades y los valores de estas

propiedades identifican y distinguen a cada entidad de las otras. Hay dos tipos de entidades, las concretas y las abstractas. Las concretas son las que son tangibles y las abstractas son intangibles.

Un conjunto de entidades es un grupo de objetos que comparten las mismas propiedades, o atributos. Cada una de las entidades que constituyen un conjunto se conoce con el nombre de extensión de ese conjunto de entidades.

Los distintos conjuntos de entidades que existen en una base de datos no son necesariamente disjuntos. Cada entidad posee un conjunto de atributos que describen las características que posee cada miembro del conjunto de entidades. La designación de un atributo para un conjunto de entidades expresa que la base de datos almacena información parecida relativa a cada entidad del conjunto de entidades, pero cada entidad puede tener su propio valor para cada atributo. En la vida real las entidades podrían tener muchos más atributos de los que se expresen en el diagrama E-R, pero se omiten para no complicar el diagrama, solo se colocaran los atributos que son susceptibles de ser almacenados, es decir solo se colocaran los atributos que necesitemos almacenar para cada entidad.

Cada entidad tiene un valor para cada uno de sus atributos. Las bases de datos incluyen una serie de conjuntos de entidades, cada una de las cuales contiene cierto número de entidades del mismo tipo.

Conjunto de Relaciones

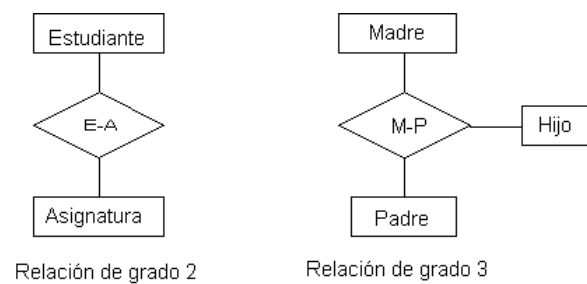
Una relación es una asociación entre dos o varias entidades, es decir es el vínculo que existe entre dos o más entidades.

La asociación entre un conjunto de entidades se conoce como participación. La función que desempeña una entidad en una relación se denomina rol de esa entidad.

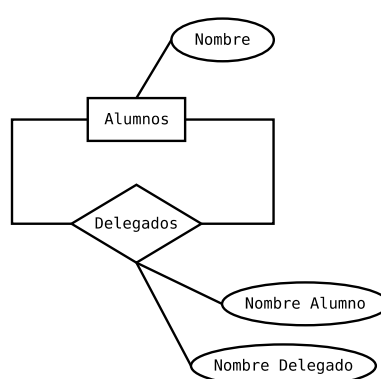
Una relación puede tener atributos llamados atributos descriptivos.

Grado: Es el número de conjuntos de entidades que participan en un conjunto de relaciones.

Los conjuntos de relaciones donde solo participan dos entidades se conocen con el nombre de binarias. La mayoría de los conjuntos de relaciones de los sistemas de bases de datos son binarios. A veces, no obstante, los conjuntos de relaciones implican a más de dos conjuntos de entidades. Los conjuntos de relaciones binarios tienen grado 2, los conjuntos de relaciones ternarios tienen grado 3, etc.



Las relaciones de grado uno son aquellas que relacionan dos entidades de un mismo conjunto de entidades. Por ejemplo la relación *delegado* de un grupo de alumnos es una relación que conecta a una entidad del conjunto de *alumnos* (llamado *delegado*) con un subconjunto del mismo grupo.



Atributos

Los atributos en un diagrama E-R tienen un conjunto de valores permitidos que son conocidos con el nombre de dominio o conjunto de valores para el atributo. Cada atributo de un conjunto de entidades es una función que asigna el conjunto de entidades a un dominio. Dado que el conjunto de entidades puede tener varios atributos, cada entidad se puede describir mediante un conjunto de pares (atributo, valor), un par por cada atributo del conjunto de entidades.

Los valores de los atributos que describen cada entidad constituyen una parte significativa de los datos almacenados en la base de datos.

Hay diferentes tipos de atributos:

Atributos simples y compuestos. Los atributos simples son aquellos que no están divididos en subpartes. Los atributos compuestos si se dividen en subpartes (es decir, en otros atributos), por ejemplo un atributo llamado nombre, puede estar formado por *nombre*, *apellido_paterno*, *apellido_materno*. O el atributo *Dirección* se puede descomponer en *calle*, *numero*, *ciudad*, *código_postal*. Los atributos compuestos ayudan a agrupar atributos relacionados, lo que hace que los modelos sean más claros.

Atributos monovalorados y multivalorados. Los atributos que poseen solo un valor se conocen con el nombre de atributos monovalorados, pero puede darse el caso que un

atributo tenga un conjunto de valores para una entidad concreta, un atributo multivalorado es aquel que puede contener varios valores, como por ejemplo al atributo *numero_telefono*. En ocasiones es necesario establecer límites inferior y superior al número de valores de un atributo multivalorado.

Atributos derivados. El valor de este tipo de atributos se puede obtener a partir del valor de otros atributos o entidades relacionadas. Por ejemplo supóngase el atributo *edad*, que se va a calcular a partir de otro atributo llamado *fecha_nacimiento*. Edad sería un atributo derivado y fecha de nacimiento un atributo básico o almacenado. El valor de los atributos derivados no se almacena, es decir no se inserta, se hacen cálculos para obtener su valor cada que sea necesario.

Uniones

Las uniones se representan como líneas que indican la asociación entre relaciones y entidades. Cada una de estas líneas indican que existe la posibilidad de que al menos una entidad posea una conexión con una o un subconjunto de un conjunto de relaciones.

La cantidad de entidades y de relaciones que se pueden conectar a una relación se denomina cardinalidad, pudiendo ser en forma simplificada, 1:1 (de uno a uno), 1:N (de uno a muchos) y N:N (de muchos a muchos).

La cardinalidad puede ser también de carácter opcional u obligatorio, con lo cual puede representarse una cardinalidad con dos factores adicionales: 0 cuando una relación es de carácter opcional, y 1 cuando es de carácter obligatorio.

Veamos un par de ejemplos:

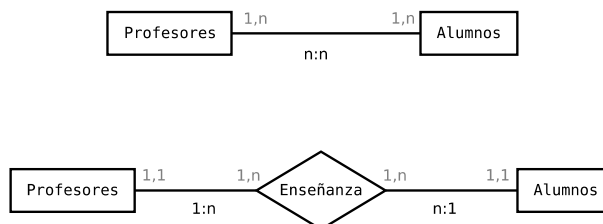
La relación de posesión que se establece entre un agente de policía (entidad) y un arma (entidad) posee dos uniones, la cardinalidad de la entidad agente a la relación posesión es de 1 a N, ya que un policía puede tener más de una posesión (más de un arma asignadas). Pero la relación entre posesión y arma es de 1 a 1 ya que un arma puede ser asignada una sola vez (a un y sólo un policía).



Pero si observamos más en detalle, podemos definir que un agente no tiene obligatoriedad en la posesión de un arma en el caso por ejemplo que realice tareas administrativas. En cambio si existe una posesión es porque hay un agente que la ejerce por lo tanto es obligatorio que exista el agente. En cuanto a la entidad arma no es obligatorio que esté en posesión, por ejemplo si se esta encuentra guardada en la armería, pero si existe posesión es obligatoria de sólo un arma.

Otro ejemplo. Pensemos en las entidades alumno y profesor, si los unimos directamente obtenemos una cardinalidad de N a N, ya que un alumno puede cursar

varias materias por lo tanto tiene relación con varios profesores y a la inversa un profesor en todos los cursos tiene relación con varios alumnos a la vez.



Si agregamos la relación de enseñanza entre ambos atributos la unión entre profesor y la relación de enseñanza será de cardinalidad 1 a N, ya que cada profesor puede tener varias relaciones de enseñanza y cardinalidades serán obligatorias ya que un profesor para existir al menos deberá tener una relación de enseñanza. Por otra parte la unión entre enseñanza y alumno será de cardinalidad N a 1 Ya que un alumno puede recibir varias enseñanzas, como mínimo una siendo de carácter obligatorio.

Atributos clave

Es un subconjunto del conjunto de atributos comunes en una colección de entidades, que permite identificar inequívocamente cada una de las entidades pertenecientes a dicha colección. Asimismo, permiten distinguir entre sí las relaciones de un conjunto de relaciones.

Dentro de los conjuntos de entidades existen los siguientes tipos de claves:

Superclave: Es un subconjunto de atributos que permite distinguir unívocamente cada una de las entidades dentro del conjunto de entidades. Si se añade un atributo al subconjunto, el resultado seguirá siendo una superclave.

Clave candidata: Se trata de superclave mínima, es decir, cualquier subconjunto de atributos de la misma no puede ser una superclave.

Clave primaria: Es una de las claves candidata elegida por el diseñador de la base de datos para identificar unívocamente las entidades en un conjunto de entidades. Por cada conjunto de entidades existe una única clave primaria.

Los valores de los atributos de una clave, no pueden ser todos iguales para dos o más instancias.

Ejemplo: en el caso de la entidad empleado que posee los atributos (*DNI, CUIL, Apellido y Nombre, Número de Legajo, fecha de alta, fecha de nacimiento, nivel jerárquico, nacionalidad*)

Podemos definir como superclaves a los subconjuntos (DNI), (CUIL), (Apellido y nombre + Fecha de Nacimiento), (Número de legajo) o la combinación de todos ellos.

Como claves candidatas tenemos a los mismos subconjuntos pero no a las combinaciones.

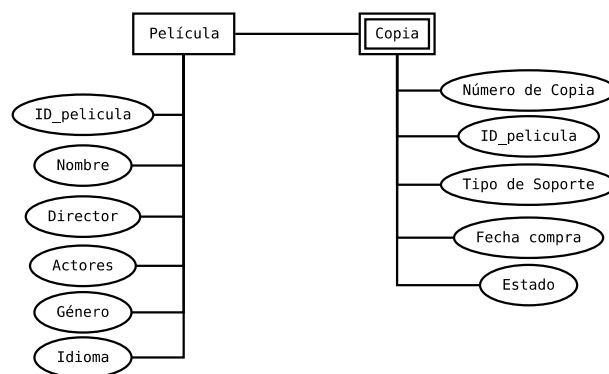
Como clave primaria elegimos el atributo Número de Legajo solo porque nos parece aceptable, pero podría ser cualquiera excepto quizás (Apellido y nombre + Fecha de Nacimiento) porque es poco práctica.

Entidades fuertes y débiles

Cuando una entidad participa en una relación puede adquirir un papel fuerte o débil. Una entidad débil es aquella que no puede existir sin participar en la relación, es decir, aquella que no puede ser unívocamente identificada solamente por sus atributos.

Una entidad fuerte (también conocida como entidad regular) es aquella que sí puede ser identificada unívocamente. En los casos en que se requiera, se puede dar que una entidad fuerte "preste" algunos de sus atributos a una entidad débil para que esta última se pueda identificar.

Las entidades débiles se representan mediante un doble rectángulo; es decir, un rectángulo con doble línea.



Se puede hablar de la existencia de 2 tipos de dependencias en las entidades débiles:

Dependencia por existencia. Las ocurrencias de la entidad débil pueden identificarse mediante un atributo identificador clave sin necesidad de identificar la entidad fuerte relacionada.

Dependencia por identidad. La entidad débil no puede ser identificada sin la entidad fuerte relacionada. Por ejemplo si tenemos una entidad *LIBRO* y otra relacionada *EDICIÓN*, para identificar una edición necesitamos conocer el identificador del libro.

Conversión de un DER en Tablas Relacionales

Cuando un diagrama de Entidad-Relación está completo y con todos sus atributos debidamente relevados, es posible realizar la conversión a tablas aplicando las siguientes reglas:

- Cada conjunto de entidades fuerte se representa con una tabla, cuyas columnas corresponden a los atributos de las entidades.
- Cada conjunto de entidades débil se representa con una tabla, con una columna por cada atributo de las entidades más una columna por cada atributo de la clave primaria de la entidad fuerte de la cual el conjunto de entidades débil depende.
- Cuando existe una relación “uno a varios” se va a generar una tabla que incluye los atributos de la entidad del extremo “varios”, es decir una columna por cada uno de los atributos de la entidad varios y una columna del atributo principal de la entidad del extremo “uno”. En otras palabras se toma el campo clave del extremo uno y se inserta en la tabla del extremo varios.
- Cuando existe una relación “varios a varios” (binaria) y toda relación donde el grado de participación sea de 2 o más de dos conjuntos de entidades (ternaria, cuaternaria) se representa con una tabla, la cual tiene una columna por cada atributo de las claves primarias de los conjuntos de entidades a los que participan en la relación, más una o más columnas por cada atributo que fueron necesarios para describir la relación.
- Si existieran campos compuestos en cualquiera de las entidades, conviene evaluar si se necesitara en la base de datos hacer búsquedas por los elementos individuales o atributos que componen el atributo compuesto, si se requiere hacer dichas búsquedas, entonces cada atributo que compone el atributo compuesto deberá ser un campo de la tabla, en caso de que no, la tabla solo contendrá una campo con el nombre del atributo compuesto y el valor de cada registro de este campo estará formado por los valores de los atributos que lo componen. Esto debido a que para hacer búsquedas en un atributo compuesto, es más fácil si se tiene una columna por cada campo que compone el campo compuesto.
- Si existe un atributo multivalorado en una tabla, este se convierte en una tabla que va a estar compuesta por una columna para el campo clave de esta nueva tabla, otro campo que será el campo clave de la tabla de donde proviene el atributo multivalorado (clave foránea) y finalmente un campo que será el que representa al atributo multivalorado, en la tabla habrá un registro por cada valor del atributo multivalorado, con diferente campo clave, y donde se va a repetir la clave foránea para conocer que

registros de esta nueva tabla corresponden a un registro de la tabla original.

- Los campos derivados se representan como una columna de la tabla.
- Si una relación contiene atributos, automáticamente se convierte en tabla, tomando los atributos de la relación como campos de la tabla y los campos claves de las tablas que participan en la relación como campos de esta nueva tabla.



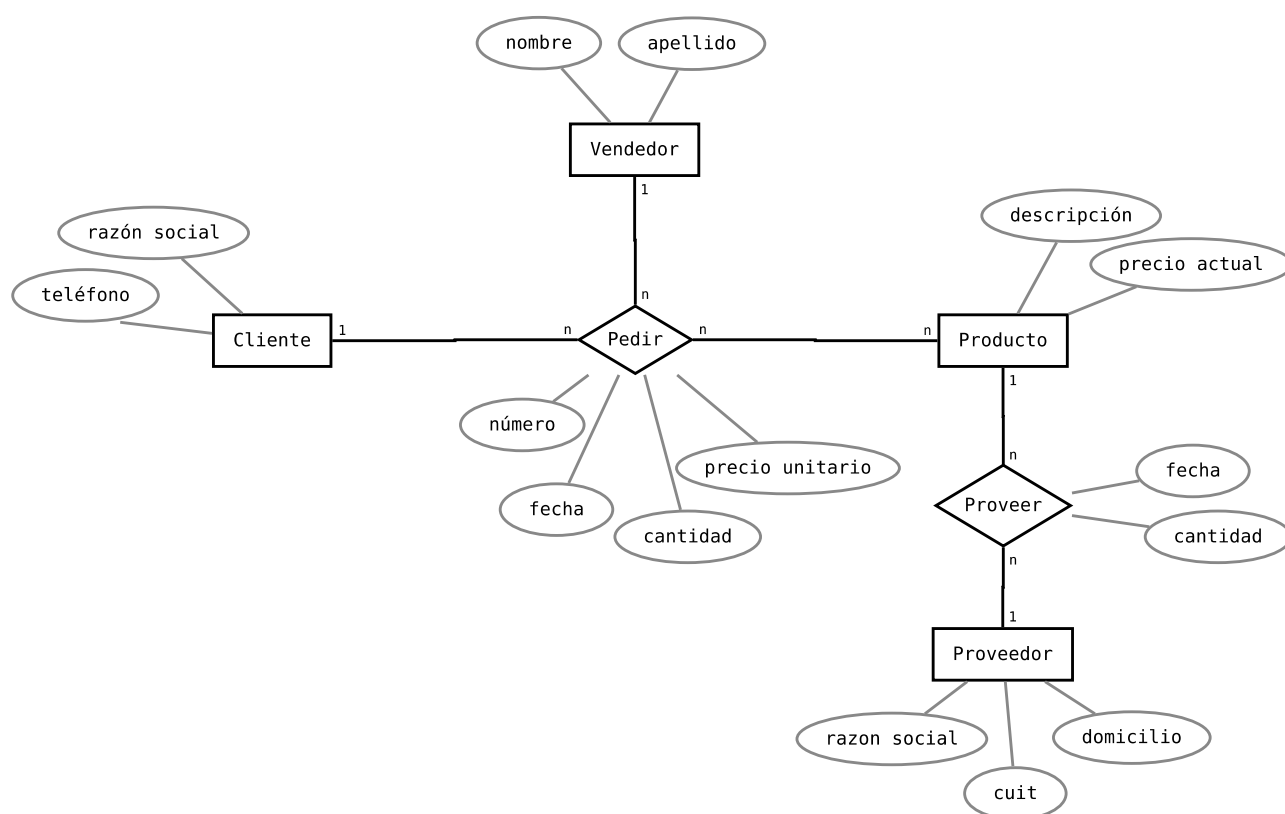
COCHE	
PK	idCoche
FK	idMotor
Color	
Longitud	
Peso	
Nº Puertas	

MOTOR	
PK	idMotor
FK	idCoche
Potencia	
Rendimiento	
Consumo	

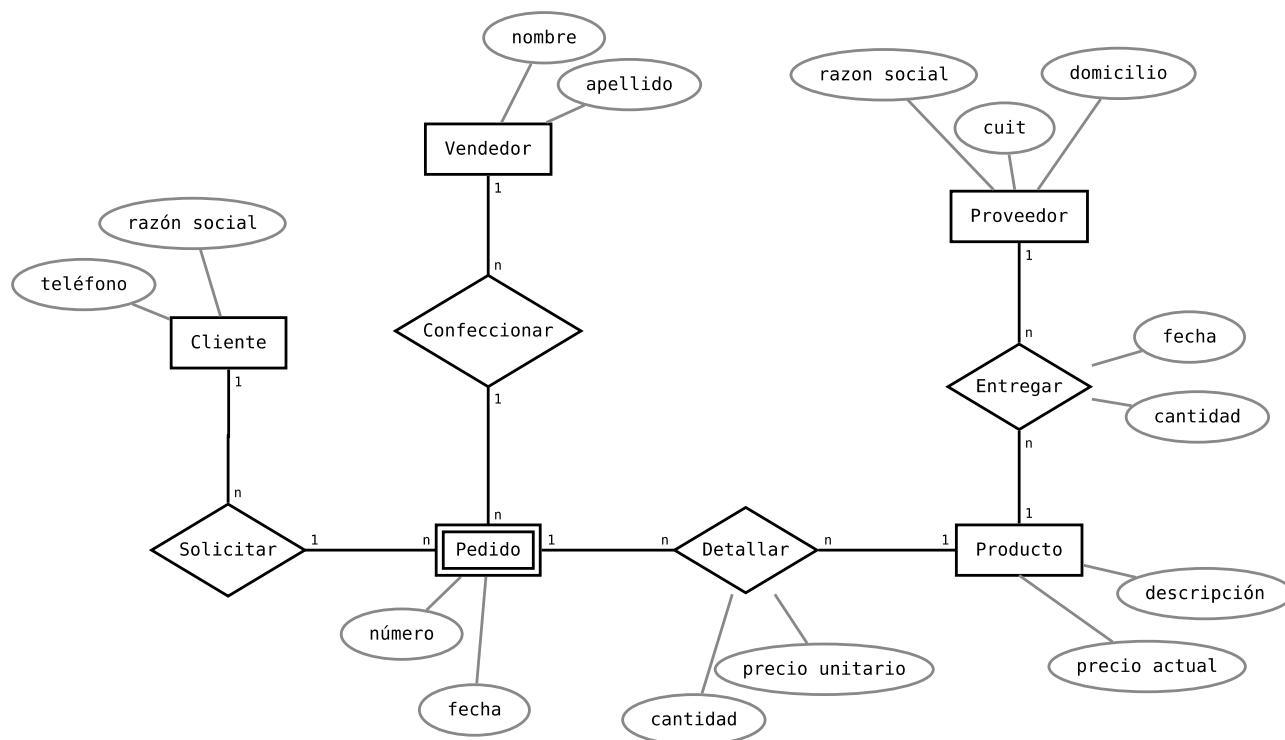
Características relevantes de las Relaciones

Veamos un ejemplo muy básico de un esquema de tablas del área comercial de una empresa típica. Para ejemplificar su implementación utilizaremos el administrador de bases de datos Access de Microsoft.

El diagrama de Entidad Relación nos muestra básicamente:



El único detalle relevante es que en el caso de la relación **pedir** observamos una cardinalidad de n a n con la tabla de productos, esto se explica porque un pedido puede involucrar varios productos y viceversa, entonces esta relación también se podría representar como una entidad débil y para desarmar la relación n a n se agregaría una relación intermedia denominada detalle de pedido de la siguiente forma:



Cada una de estas entidades representaría una tabla, con el agregado de una tabla para el detalle del pedido que tiene atributos propios. Más adelante seremos más específicos con el razonamiento utilizado para representar entidades y relaciones en tablas y atributos en campos, pero por ahora es suficiente con establecer una estructura básica.

Una vez creadas tablas diferentes, una para cada entidad de la base de datos se necesita una forma de indicar a Microsoft Access cómo debe volver a combinar esa información. El primer paso de este proceso es definir las relaciones entre las tablas.

Una vez realizada esta operación, puede crear consultas, formularios e informes para mostrar información de varias tablas a la vez. Por ejemplo, el formulario de la imagen incluye información de cuatro tablas:

1. De la tabla Clientes
2. De la tabla Pedidos
3. De la tabla Productos
4. De la tabla Detalles de pedidos

Producto:	Precio Uni.	Cantidad
Gudgrandsalsost	24,00 \$	14
Scottish Longbread	12,50 \$	20
Pâte chinois	36,00 \$	10

Cómo funcionan las relaciones

Siguiendo en el ejemplo anterior, los campos de las cuatro tablas deben coordinarse de modo que muestren información acerca del mismo pedido. Esta coordinación se lleva a cabo mediante las relaciones entre las tablas. Una

relación hace coincidir los datos de los campos clave. En la mayoría de los casos, estos campos coincidentes son la clave principal de una tabla, que proporciona un identificador único para cada registro, y una clave externa de la otra tabla. Por ejemplo, los empleados pueden asociarse a los pedidos de los que son responsables mediante la creación de una relación entre los campos Id. de empleado.



Id. de empleado aparece en ambas tablas,

1. como clave principal ...
2. ... y como clave foránea.

Una relación uno a varios

La relación uno a varios es el tipo de relación más común. En este tipo de relación, un registro de la Tabla A puede tener muchos registros coincidentes en la Tabla B, pero un registro de la Tabla B sólo tiene un registro coincidente en la Tabla A.



1. Un proveedor ...
2. ... puede suministrar más de un producto ...
3. ... pero cada producto tiene un único proveedor.

Una relación varios a varios

En una relación varios a varios, un registro de la Tabla A puede tener muchos registros coincidentes en la Tabla B, y viceversa. Este tipo de relación sólo es posible si se define una tercera tabla (denominada tabla de unión) cuya clave principal consta de dos campos : las claves externas de las Tablas A y B. Una

relación de varios a varios no es sino dos relaciones de uno a varios con una tercera tabla. Por ejemplo, la tabla Pedidos y la tabla Productos tienen una relación de varios a varios que se define mediante la creación de dos relaciones de uno a varios con la tabla Detalles de pedidos.



1. Clave principal de la tabla *Pedidos*
2. Clave principal de la tabla *Productos*
3. Un pedido puede incluir muchos productos ...
4. ... y cada producto puede aparecer en muchos pedidos.

Una relación uno a uno

En una relación uno a uno, cada registro de la Tabla A sólo puede tener un registro coincidente en la Tabla B y viceversa. Este tipo de relación no es habitual, debido a que la mayoría de la información relacionada de esta forma estaría en una sola tabla. Puede utilizar la relación uno a uno para dividir una tabla con muchos campos, para aislar parte de una tabla por razones de seguridad o para almacenar información que sólo se aplica a un subconjunto de la tabla principal. Por ejemplo, puede crear una tabla que registre los empleados participantes en un partido de fútbol benéfico.



1. Cada jugador de fútbol de la tabla *Jugadores de fútbol* tiene un registro coincidente en la tabla *Empleados*.

2. Este conjunto de valores es un subconjunto del campo *Id. de empleado* de la tabla Empleados.