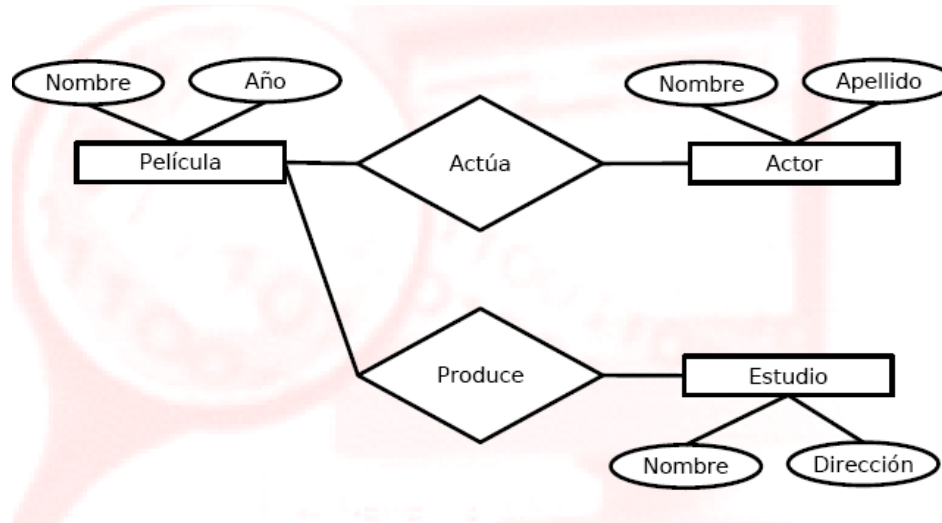




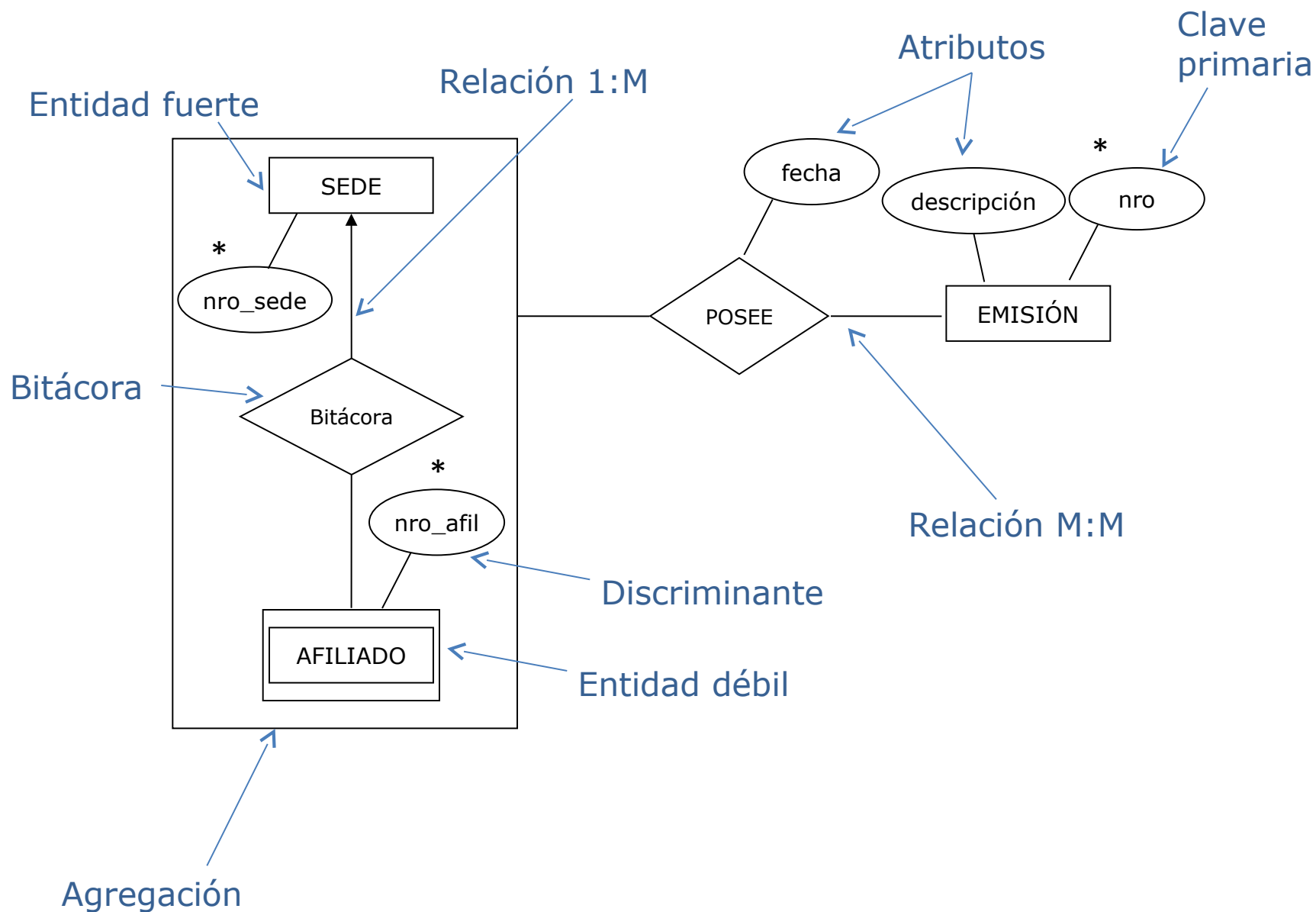
Universidad Nacional de la Patagonia San Juan Bosco
Facultad de Ingeniería

Cátedra: **Bases de datos I**

Modelo Entidad Relación



- Introducción - El modelo entidad-relación
 - Es el **modelo conceptual** más utilizado para el **diseño del esquema conceptual** de las BD.
 - Fue propuesto por Peter Chan a mediados de los 70.
 - Si se elige diseñar la BD en el **Modelo E-R**, luego se transforma al **Modelo Relacional** para implementarla.
 - Se basa en la percepción del mundo real y consiste en buscar las entidades básicas del sistema a modelar y luego la relación que existe entre ellas.
 - El modelo E-R describe los datos como entidades, relaciones (vínculos) y atributos y permite representar el **esquema conceptual** de una BD de forma gráfica mediante los **diagramas E-R**.
 - El diagrama E-R es un lenguaje gráfico tan rígido como uno de programación, **cada símbolo significa algo exacto**:
 - ✓ Rectángulos: Entidades
 - ✓ Rombos: relaciones entre conjuntos de entidades
 - ✓ Cardinalidad: sin flecha: muchos; flecha: uno
 - ✓ Elipses: atributos
 - ✓ Rectángulo doble: entidades débiles
 - ✓ Triángulo invertido: relación especial ES-UN. Para generalizaciones



- Entidades y conjunto entidades

- **Entidad:** Objeto básico que se representa en el modelo E-R. Es cualquier elemento del mundo real que se puede identificar claramente, sobre el cual queremos tener información en una BD. Una entidad puede ser un objeto con **existencia física** (una persona, una casa, un empleado, un coche) o un objeto con **existencia conceptual** (un accidente, un puesto de trabajo, un curso universitario).
- **Conjunto de entidades:** Totalidad de las entidades del mismo tipo que comparten las mismas propiedades o atributos. Es la clase o tipo al que pertenece cada entidad. En el modelo E-R trabajamos con conjuntos entidad y no con entidades. En los diagramas E-R se representan mediante un rectángulo y dentro del mismo se pone el nombre en singular y mayúsculas. Por ejemplo: *PELICULA, ACTOR, ESTUDIO*, etc.
- ✓ Debemos elegir nombres que comuniquen el significado de cada entidad.
- ✓ Se utilizan nombres en singular y no en plural.
- ✓ Un nombre de entidad sólo puede aparecer una vez en el esquema conceptual.
- Hay dos tipos de entidades:
- ✓ **Fuertes:** tienen existencia por sí mismas (P. e. *EDIFICIO*). Las entidades fuertes se representan con un rectángulo con trazo simple.



EDIFICIO

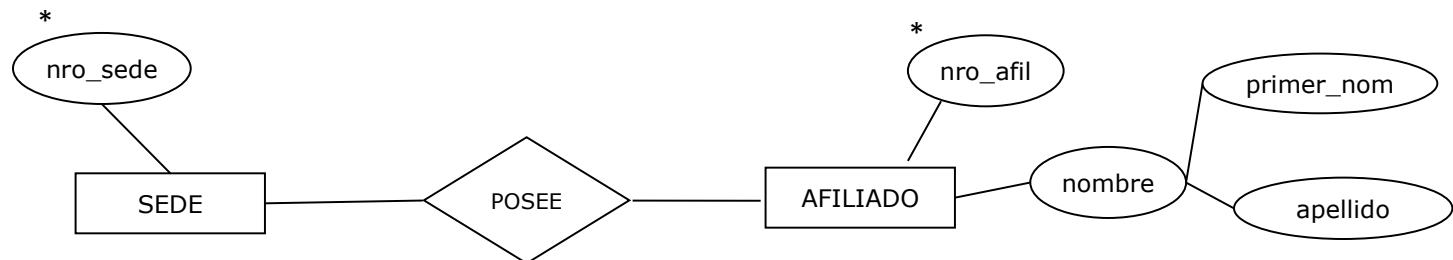
- ✓ **Débiles:** su existencia depende de otra entidad, en el sentido de que la entidad débil no puede existir en **ese contexto** sin la fuerte. P. e. en una BD de una inmobiliaria, el conjunto entidad *DEPARTAMENTO* depende de *EDIFICIO*. Se representan con un rectángulo con líneas de doble trazo. No tienen suficientes atributos para formar una clave primaria.



DEPARTAMENTO

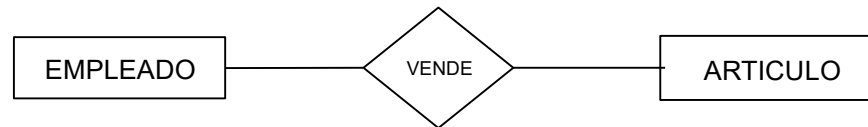
• Atributos

- Cada entidad tiene propiedades específicas, llamadas **atributos**, que permiten distinguir una entidad de otra. P. e. una entidad *PELÍCULA* puede describirse por su *nombre*, *año*, etc. Los atributos se representan por elipses que están conectadas a su entidad o relación mediante una línea recta.
- Un atributo en el modelo E-R se puede clasificar entre los siguientes tipos:
- ✓ **Atributos simples:** En los ejemplos considerados hasta ahora, los atributos han sido simples → no están divididos en subpartes.
- ✓ **Atributos compuestos:** se pueden dividir en subpartes (en otros atributos). P.e. *nombre* podría estar compuesto de *primer_nombre* y *apellido*. Se los usa en un esquema de diseño si el usuario desea referirse a un atributo entero en algunas ocasiones y a algún componente de atributo en otras. Al agrupar los atributos relacionados, hacen más claro al modelo.
- ✓ **Atributos nulos:** Se usa cuando una entidad no tiene valor para un atributo. P.e. si una vivienda no tiene piso o número de departamento, el valor del atributo piso para esa vivienda será nulo. Nulo puede también designar que el valor de un atributo es desconocido.
- ✓ **Atributos derivados:** El valor para este tipo de atributo se puede derivar de los valores de otros atributos o entidades. P. e. en el conjunto de entidades *ESTUDIO*, supongamos que tiene un atributo *cantidad_películas* que representa cuántas películas se hicieron en el estudio. Ese atributo se puede derivar contando el número de entidades *PELÍCULA* asociadas con el estudio.

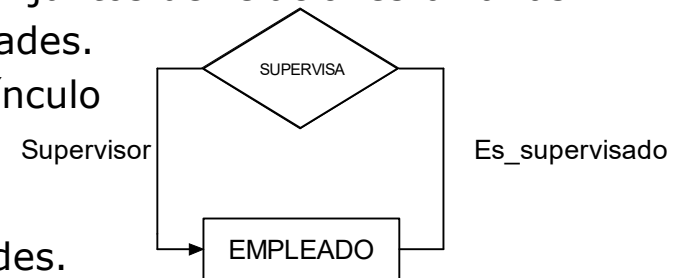


- Relaciones y conjunto de relaciones

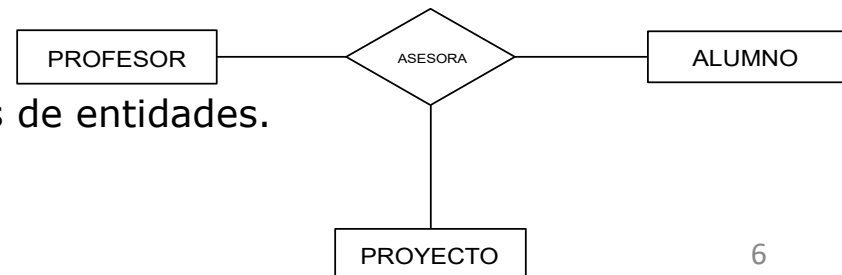
- Una **relación** se puede definir como una correspondencia, asociación o conexión entre dos o más conjuntos entidades. Un conjunto de relaciones son todas las relaciones del mismo tipo. En los diagramas E-R se representa gráficamente como un rombo y sus nombres son verbos. Por ejemplo: *VENDE*, *PERTENECE*, etc.



- La relación en este ejemplo es: *VENDE*, una instancia de esta relación es *Pérez vende computadoras*.
- Una relación puede tener **atributos descriptivos**. P. e. en la relación anterior, podría tener como atributo descriptivo *fecha_venta* y *cant_vta*. Ver la diferencia con el atributo cantidad de la entidad *ARTÍCULO*.
- **Grado de una relación**: Número de entidades que participan en la relación. Se aconseja restringir el modelo para incluir solo conjuntos de relaciones binarias.
- ✓ **Reflexiva**: Participa un único conjunto de entidades.
En estos casos requiere un nombre (rol) en el vínculo



- ✓ **Binaria**: Participan dos (2) conjuntos de entidades.

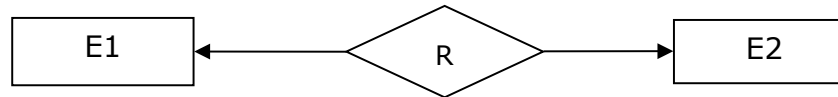


- ✓ **N-aria**: Participan más de dos conjuntos de entidades.

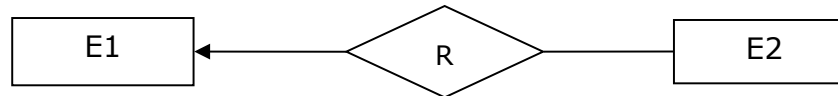
- Restricciones de asignación (Cardinalidad)

- **Cardinalidad:** Expresa el número de entidades con las que puede asociarse otra entidad mediante un conjunto de relaciones.

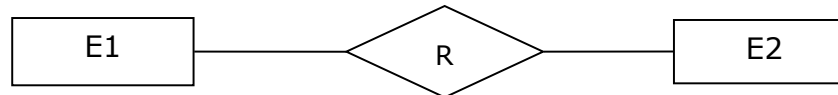
- ✓ **Uno a uno (1:1):** Una entidad del conjunto de entidades E1 se relaciona con una única entidad del conjunto de entidades E2 y viceversa.



- ✓ **Uno a muchos (1:M):** Una entidad del conjunto de entidades E1 se relaciona con muchas entidades del conjunto de entidades E2 y una entidad del conjunto de entidades E2 solo puede estar relacionada con una entidad del conjunto de entidades E1. También se puede dar Muchos a Uno (M:1).



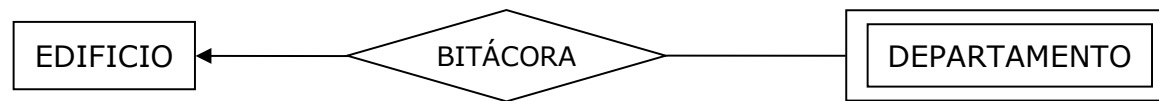
- ✓ **Muchos a muchos (M:M):** Una entidad del conjunto de entidades E1 está relacionada con muchas entidades del conjunto de entidades E2 y viceversa.



- Existen diversas maneras de representar la cardinalidad, Korth distingue en el diagrama la cardinalidad a partir de líneas dirigidas o no, la línea dirigida indica que la relación es a uno y la no dirigida es a muchos.

- Relaciones débiles

- Sean las entidades *EDIFICIO* y *DEPARTAMENTO*, en el contexto de una inmobiliaria *DEPARTAMENTO* es débil. Se presenta entre ambas la **relación bitácora, de 1:M, sin atributos** y especifica que un edificio determinado puede tener varios departamentos. Si se suprime una entidad edificio (deja de pertenecer a la inmobiliaria), entonces todas sus entidades departamento asociadas también deben suprimirse. ¿Y las entidades departamento?



Las relaciones débiles pueden ser:

- ✓ **Por Existencia (E):** una instancia de la entidad débil no puede existir si desaparece la instancia de la entidad fuerte de la que depende, pero si puede ser identificada por sí misma. P.e. La relación entre *Cliente* y *Pedido*, *Pedido* (débil) puede identificarse por sí misma (*código_pedido*).
- ✓ **Por Identificación (ID):** es una relación débil por existencia y una instancia de la entidad débil no puede ser identificada por sí misma. Es necesario añadir información de la instancia de la entidad regular con la que está relacionada. P.e. La relación entre *Cuenta_bancaria* y *Operación*, el *número_operación* no basta para identificarla, necesitamos además el *número_cuenta*.
- ⇒ Las dependencias por existencia constituyen otra clase importante de restricciones. Específicamente, si la existencia de entidad x depende de la existencia de la entidad de y → si se suprime y también se suprime x. La entidad y se dice que es una entidad dominante y x una entidad subordinada.

- Claves

- ✓ **Superclave:** Conjunto de atributos **no nulo**, que identifica en forma única una entidad dentro de un conjunto de entidades. Una superclave puede contener subconjuntos que son superclave.
- ✓ **Clave candidata:** Cada una de las **claves mínimas** que existe en un conjunto entidad. Una clave es mínima cuando al eliminarse uno de sus atributos deja de ser clave.
- ✓ **Clave primaria:** La clave candidata escogida por el diseñador. Atributo o conjunto de atributos que permiten identificar en forma única una tupla en la tabla y ningún subconjunto de ella posee esta propiedad.
- ✓ **Clave foránea:** Atributo de un conjunto entidad que es clave primaria en otra entidad con la cual se relaciona.
- ✓ **Discriminante:** Conjunto de atributos que diferencia una ocurrencia de una entidad débil de otra, para una misma clave de la entidad fuerte de la que depende.

- Ejemplos

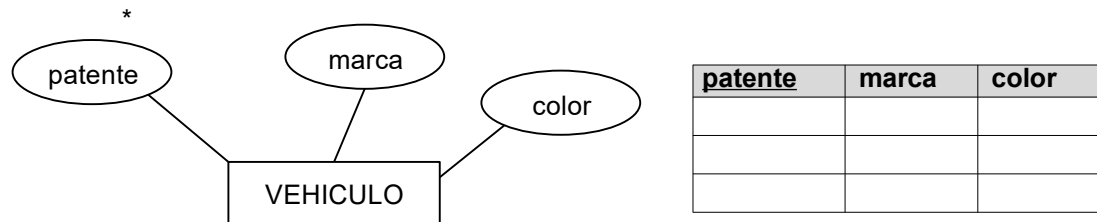
- ✓ Sea la entidad *VEHÍCULO*, cuáles de las siguientes son superclaves? *patente*, *marca*, *{patente, marca}*
- ✓ Varios conjuntos de atributos distintos pueden servir como claves candidatas. Supóngase que una combinación de *fecha_compra*, *marca* y *nombre_dueño* es suficiente para distinguir entre los miembros del conjunto de entidades *VEHÍCULO*. Entonces *{patente}* y *{fecha_compra, marca, nombre_dueño}* son claves candidatas. Aunque los atributos *patente* y *marca* juntos pueden distinguir entidades *VEHÍCULO*, sus combinaciones no forman una clave candidata, ya que el atributo *patente* por sí solo es una clave candidata.

- Problemas con las restricciones durante las operaciones
 - Inserción de una nueva tupla en una tabla
 - ✓ Todos los atributos de la clave primaria tienen valor no nulo.
 - ✓ El conjunto de todos los atributos que forman la clave primaria toma un valor único e inédito hasta el momento en la tabla.
 - ✓ Todos los atributos que son claves ajenas de otras tablas toman valores ya presentes en dichas tablas o bien nulos.
 - ✓ Todos los valores de los atributos satisfacen todas las restricciones adicionales que pudieran concernirles.
 - Modificación del valor de algún atributo de una o varias tuplas de una tabla
 - ✓ Si el atributo a modificar forma parte de la clave primaria, su valor no puede modificarse a nulo
 - ✓ Si el atributo a modificar forma parte de la clave primaria, su valor no puede modificarse a otro tal que la nueva clave primaria ya no sea única en la tabla.
 - ✓ Si el atributo a modificar es parte de una clave ajena en otra tabla, entonces hay que modificar automáticamente el viejo valor que tomaba en dicha tabla por el nuevo valor.
 - ✓ Sólo puede modificarse el valor de un atributo si el nuevo valor satisface todas las restricciones adicionales que puedan afectarle.

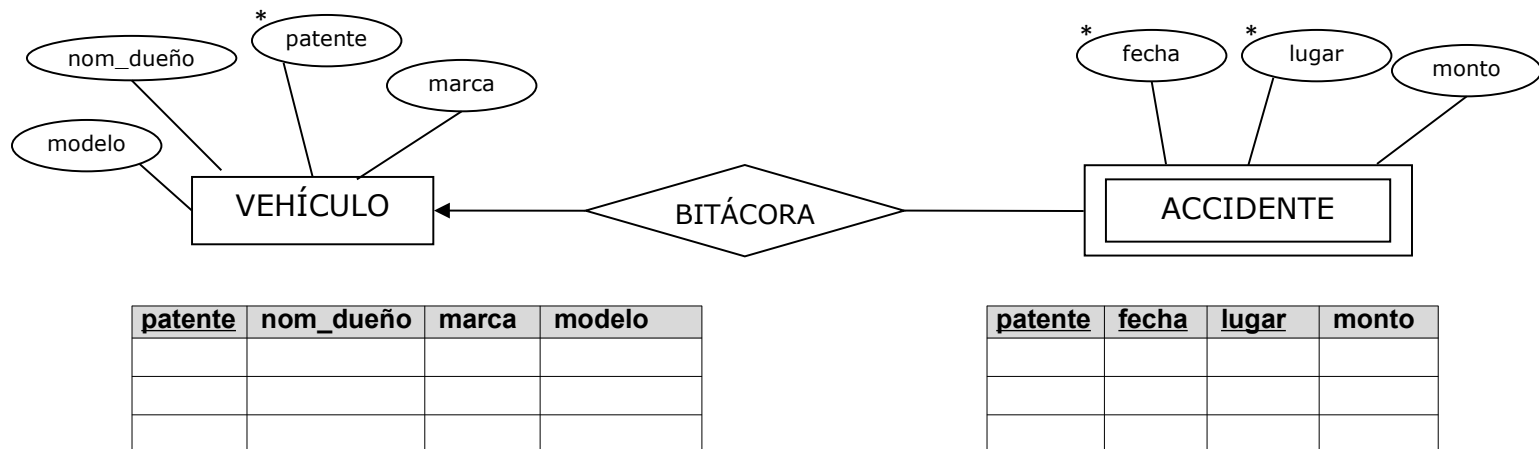
- Reducción de diagrama E-R a tablas

- Tanto el modelo E-R, como el modelo de BD relacional son representaciones abstractas y lógicas del desarrollo del mundo real. Debido a que los dos modelos emplean principios de diseño similares, se puede convertir un diseño E-R en un diseño relacional, siguiendo una serie de normas que podemos resumir de la siguiente forma:

- ✓ **Entidades:** Se genera una tabla con los atributos de una entidad. La clave primaria de la tabla es la misma que la de la entidad del modelo E-R.

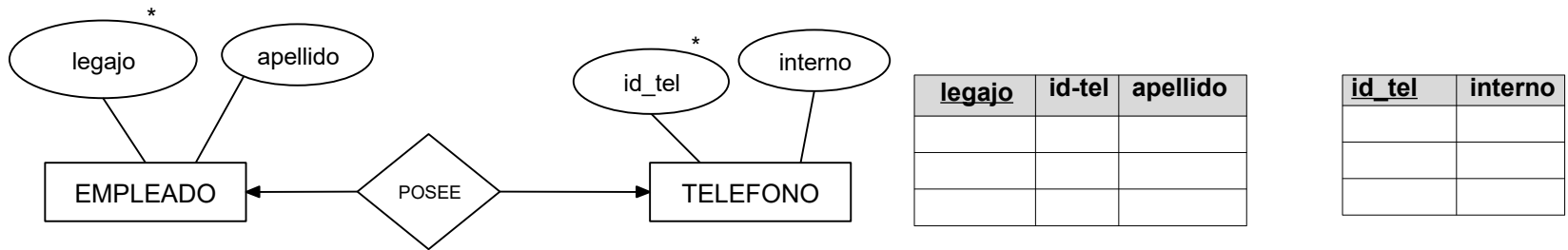


- ✓ **Entidades Débiles:** se genera una tabla con los atributos de la entidad débil, más la clave primaria de la entidad fuerte. La clave primaria de la tabla generada por la entidad débil estará formada por los atributos discriminantes de la entidad débil más los atributos clave de la entidad fuerte.

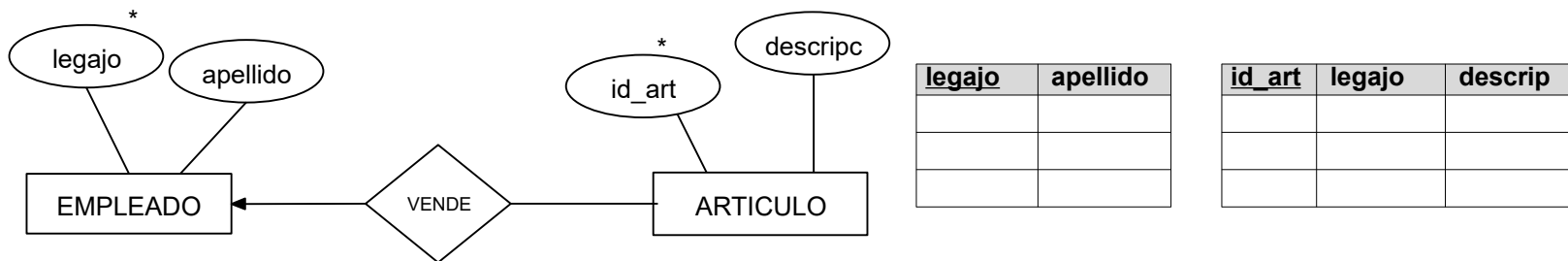


✓ RELACIONES:

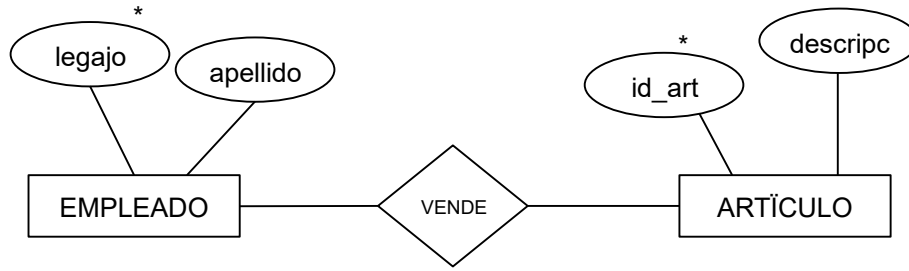
- I. Relación **1:1**, son necesarias dos tablas: En una de las tablas se añade una columna que contendrá la clave primaria de la otra entidad (clave foránea). La clave primaria de cada tabla del modelo relacional serán las mismas que las de las entidades asociadas del modelo E-R.



- I. Relación **1:M**, se necesita una tabla para cada entidad: A la tabla que representa la entidad M se le añade un atributo que contenga la clave primaria de la entidad con la que se relaciona (clave foránea).



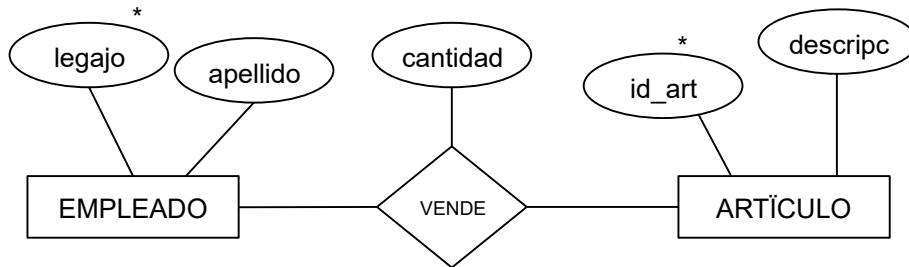
III. Relación **M:M**, se generan tres tablas: Una para cada entidad y otra que contiene la claves primarias de las entidades que participan en la relación. Si tiene atributos, se los colocan en la tabla correspondiente a la relación.



<u>legajo</u>	apellido

<u>legajo</u>	<u>id_art</u>

<u>id_art</u>	descrip



<u>legajo</u>	<u>id_art</u>	cantidad

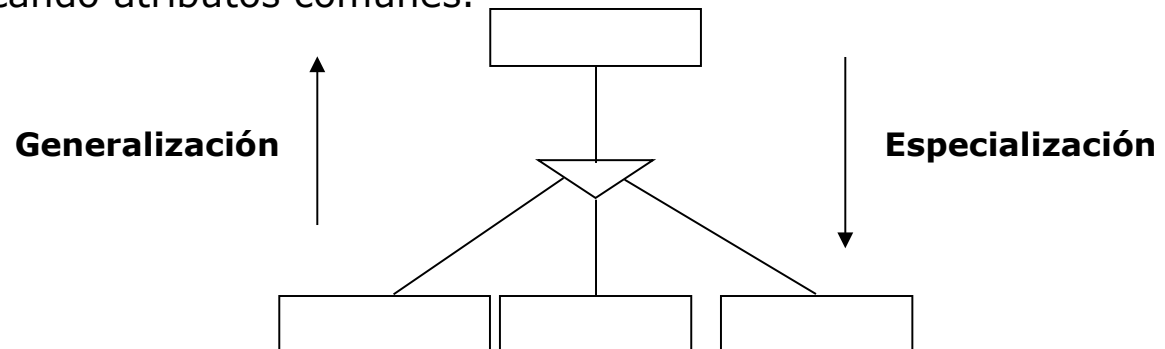
- **Modelo Relacional Extendido:**

- El modelo E-R extendido pretende aportar soluciones a requerimientos un tanto más complejos no contemplados en el modelo E-R propuesto por Codd. Así se incorporan al modelo E-R dos nuevos elementos: Generalización y Agregación.
- **Generalización:** Su uso está justificado en los siguientes casos:
 - Cuando las subclases tienen atributos particulares que no tiene la superclase.
 - Cuando existen tipos relación en los que participan solo algunas subclases.
- ✓ **Superclase:** Todo tipo entidad sobre el que se definen subclases, se representa con un rectángulo.
- ✓ **Subclases:** Es un subconjunto del tipo entidad que tiene atributos particulares, se representa con un rectángulo.
- ✓ **Relación:** El tipo relación entre una superclase y sus subclases, se dice que es un tipo ES_UN (IS_A). Este tipo relación se representa a diferencia del resto de relaciones con un triángulo.
- ✓ **Especialización:** es el proceso de definir un conjunto de subclases a partir de un conjunto entidad.
- ✓ **Generalización:** es el proceso de suprimir las diferencias entre varios conjuntos entidad, identificando atributos comunes.

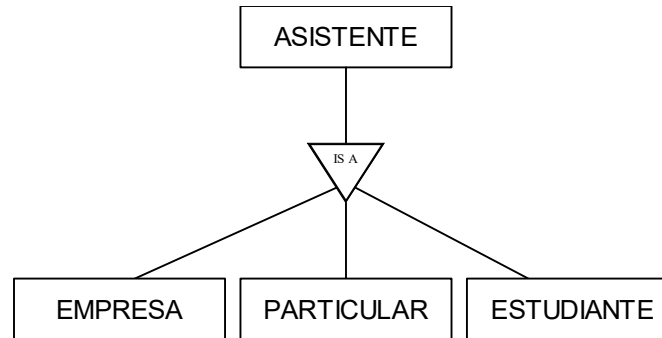
Superclase

Relación ES_UN

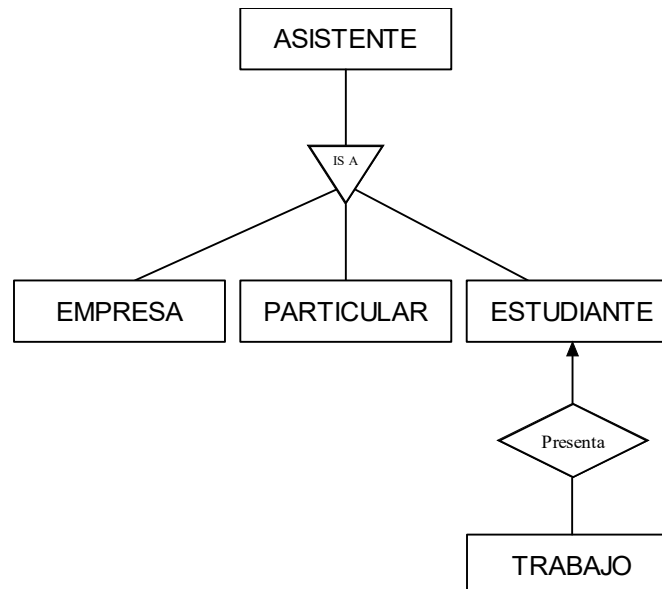
Subclases



- Suponga la siguiente situación: Se va a dictar un curso en una Universidad dirigido a estudiantes y graduados. En una BD se desea almacenar información referente a los inscriptos: D.N.I., Apellido y Nombre , además si es graduado y actualmente trabaja en una empresa interesa guardar el nombre de la empresa y el cargo que posee en la misma, si es graduado particular se debe guardar la dirección particular y si es estudiante, la carrera y el año que cursa.

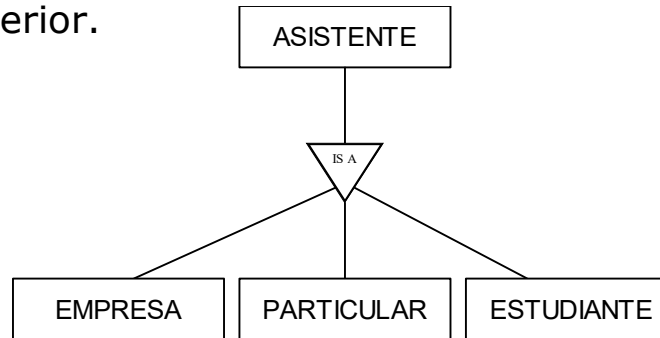


- ¿Qué sucede si los estudiantes deben presentar trabajos relacionados con el curso? Estos trabajos deben tener un título y una breve descripción.



- **Cómo bajar a tablas la generalización:**

- Se crea una tabla para el conjunto de entidades del nivel mas alto. Para el conjunto de entidades de nivel mas bajo se crea una tabla con una columna para cada uno de los atributos, mas una columna con la clave primaria del conjunto de entidades de nivel superior.



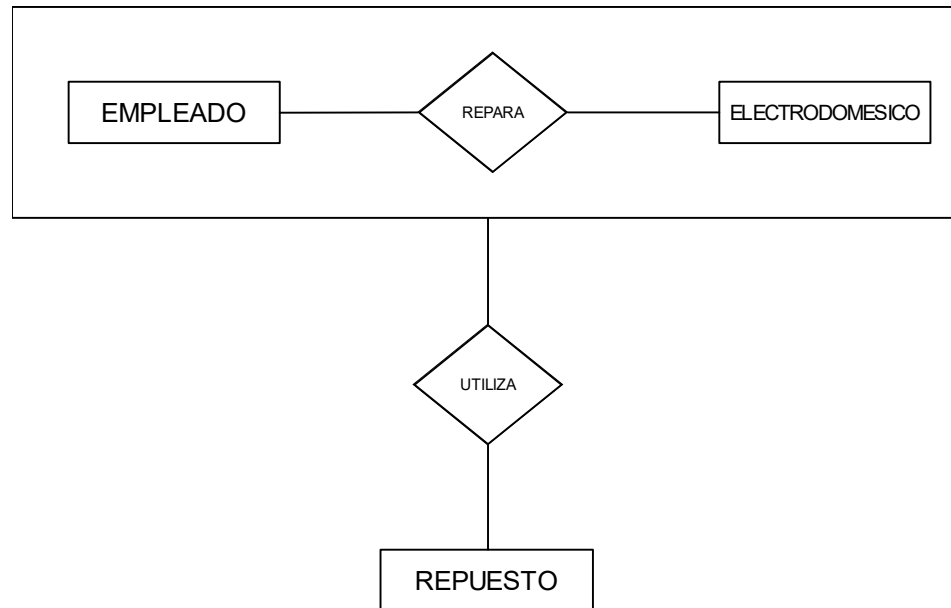
ASISTENTE			EMPRESA			PARTICULAR		ESTUDIANTE		
<u>dni</u>	apellido	nombre	<u>dni</u>	nom_empresa	cargo	<u>dni</u>	dirección	<u>dni</u>	carrera	año

- Otra posibilidad es bajar la información a tres tablas, cada una con la información del conjunto de entidades de más alto nivel más los propios:

EMPRESA					PARTICULAR			
<u>dni</u>	apellido	nombre	nom_empresa	cargo	<u>dni</u>	apellido	nombre	dirección

ESTUDIANTE				
<u>dni</u>	apellido	nombre	carrera	año

- **Agregación:** Nos permite relacionar una relación con otra entidad. Hasta el momento sólo se podían relacionar conjuntos de entidades. Este nuevo elemento permite relacionar relaciones con entidades, o relaciones entre sí, si siempre que el caso lo requiera. Sólo admite relaciones M:M o Bitácoras.
- Supongamos la siguiente situación problemática: En un comercio de reparación de electrodomésticos es importante saber cuál empleado reparó qué electrodoméstico, un mismo electrodoméstico puede ser reparado por varios empleados. Además se necesita saber qué repuestos se usaron en una reparación.



- Para bajar a tablas la agregación: cada conjunto entidad se baja a tabla como vimos. El conjunto relación **REPARA** también se explicó como se baja a tablas. El conjunto relación **UTILIZA** se baja a tablas con la clave de la agregación y la clave del conjunto entidad **REPUESTO**.

Modelo Relacional

Esquema_sucursal=(nombre_sucursal, activo, ciudad_sucursal)

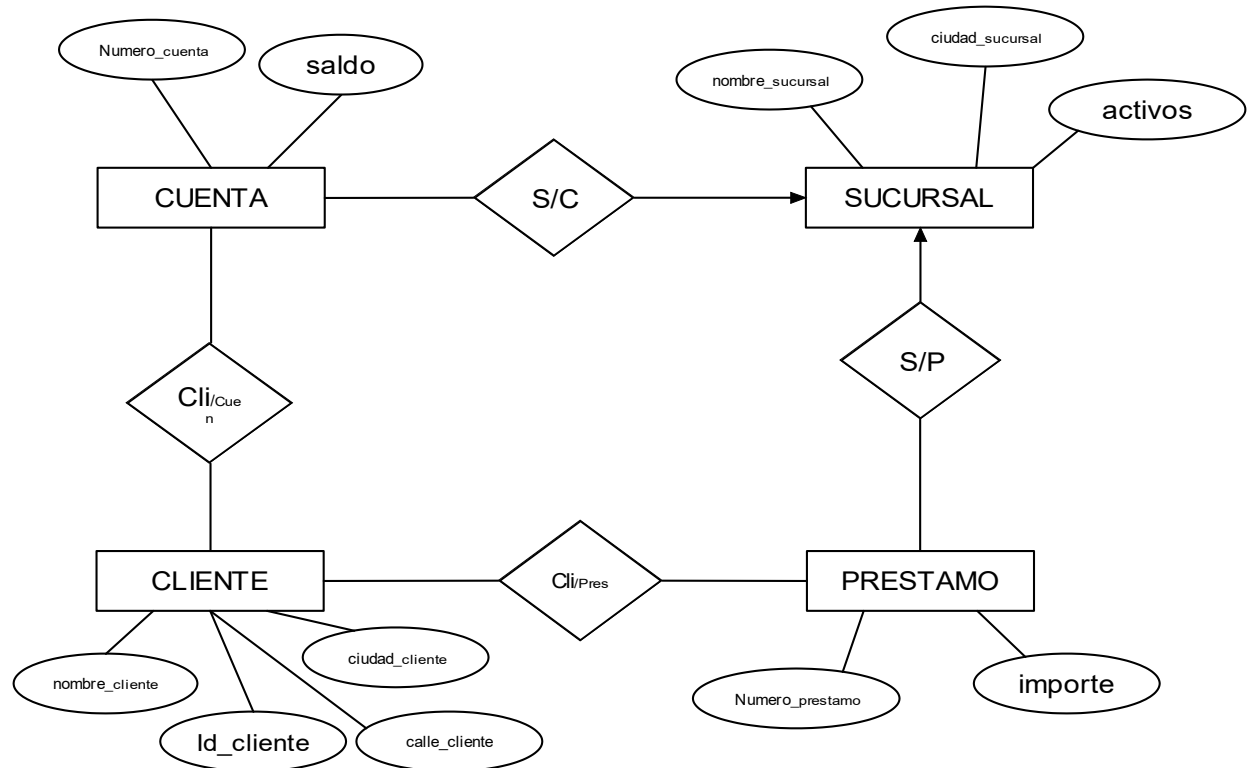
Esquema_cliente=(Id_cliente, nombre_cliente, calle, ciudad_cliente)

Esquema_cuenta=(nombre_sucursal, número_cuenta, saldo)

Esquema_préstamo=(nombre_sucursal, número_préstamo, importe)

Esquema_cli/cuen=(nombre_cliente, número_cuenta)

Esquema_Cli/Pres= (nombre_cliente, numero_prestamo)



Introducción

En los años siguientes a la introducción del modelo relacional se ha desarrollado una **teoría esencial para las BDR** que ayuda al **diseño** de BDR y al **procesamiento eficiente de solicitudes de información** a la BD por parte de los usuarios.

El modelo relacional se ha establecido como el principal modelo de datos para aplicaciones comerciales de procesamiento de datos y a su aplicación en otros entornos.

Estructura de las Bases de Datos Relacionales

Una BDR consiste en una colección de tablas a las cuales se le asigna un nombre único. Cada tabla tiene una estructura similar a la presentada anteriormente, donde se presentan en diagramas E-R mediante tablas. Cada fila de una tabla representa una relación entre un conjunto de valores. Puesto que una tabla es una colección de dichas relaciones, hay una estrecha correspondencia entre el concepto de tabla y el concepto **matemático de relación**, del cual toma su nombre el modelo de datos relacional.

Utilizaremos varias relaciones diferentes para ilustrar los diversos conceptos en los que se basa el modelo de datos relacional. Estas relaciones representan parte de una empresa bancaria. Trataremos las estructuras relacionales apropiadas con detalle.

Estructuras básicas

Sea la siguiente relación (tabla) *CUENTA*

Con los atributos: *nombre_sucursal*,
número_cuenta, y *saldo*.

nombre_sucursal	número_cuenta	saldo
Norte	101	500
Sur	215	700
Oeste	102	400
Centro	305	350
Oeste	201	900
Norte	222	1800

Para cada atributo hay un conjunto de valores permitidos, llamado **dominio**.

Para el atributo *nombre_sucursal*, el dominio es el conjunto de todos los nombres de las sucursales. Sea D_1 ese conjunto y sea D_2 el conjunto de todos los *números_cuenta*, D_3 el conjunto de todos los *salDOS*. Cada una de las filas de *CUENTA* debe constar de 3 tuplas (v_1, v_2, v_3), donde v_1 es un nombre de sucursal (es decir, v_1 está en el dominio D_1), v_2 es un número de cuenta (es decir, v_2 está en el dominio D_2), v_3 es un saldo (es decir, v_3 está en el dominio D_3). En general, *CUENTA* contendrá únicamente un subconjunto del conjunto de todas las filas posibles. Por tanto, *CUENTA* es un subconjunto de $D_1 \times D_2 \times D_3$

En general, una tabla de n columnas debe ser un subconjunto de $D_1 \times D_2 \times \dots \times D_n$

Los matemáticos definen una **relación** como un **subconjunto de un producto cartesiano** de una lista de dominios. Esto corresponde casi exactamente con nuestra definición de tabla. La única diferencia es que nosotros asignamos nombres a atributos, mientras que los matemáticos se basan en "nombres" numéricos, usando el entero 1 para indicar el atributo cuyo dominio aparece primero en la lista de dominios; 2 para indicar el atributo cuyo dominio aparece segundo, y así sucesivamente. Puesto que las tablas son esencialmente relaciones, usaremos los términos matemáticos **relación** y **tupla** en lugar de los términos tabla y fila.

nombre_sucursal	número_cuenta	saldo
Norte	101	500
Sur	215	700
Oeste	102	400
Centro	305	350
Oeste	201	900
Norte	222	1800

¿Cuántas tuplas hay en la relación *CUENTA*?

Si la variable tupla **t** hace referencia a la primera tupla de la relación:

- ✓ Usamos la notación **t[nombre_sucursal]** para indicar el valor de t en el atributo *nombre_sucursal* → Así, t[nombre_sucursal]="Norte".
- ✓ De la misma forma t[número_cuenta]=101, es el valor de t en el atributo *número_cuenta*.

Alternativamente podemos escribir t[1] para indicar el valor de la tupla t en el atributo *nombre_sucursal*, t[2] para indicar el *número_cuenta*, y así sucesivamente.

Puesto que una relación es un conjunto de tuplas, usamos la notación matemática $t \in r$ para indicar que la tupla t está en la relación r.

Esquema de base de datos

Cuando hablamos de una BD debemos diferenciar entre el esquema de la BD (su diseño lógico) y una instancia de la BD ... ¿qué es una instancia de la BD?

El concepto de esquema de una relación corresponde a la noción de definición de tipo en los lenguajes de programación. Una variable de un tipo dado tiene un valor determinado en un instante de tiempo dado y se corresponde al concepto de una instancia de una relación.

Es conveniente dar un nombre al esquema de una relación. Adoptamos el convenio de usar nombres en mayúsculas para relaciones y nombres empezando con una letra mayúscula para los esquemas de relaciones. Siguiendo esta notación usamos *Esquema_cuenta* para indicar el esquema de relación para la relación *CUENTA*.

✓ $\text{Esquema_cuenta} = (\text{nombre_sucursal}, \text{número_cuenta}, \text{saldo})$

Indicamos el hecho de que *CUENTA* es una relación sobre el *Esquema_cuenta* de esta manera: *CUENTA* (*Esquema_cuenta*)

En general, el esquema de una relación es una lista de atributos y sus correspondientes dominios. No nos preocuparemos de dar una definición precisa del dominio de cada atributo hasta que tratemos las restricciones de integridad.

Dos relaciones más de esta BD, la relación *CLIENTE* y la relación *SUCURSAL*:

<u>id_cliente</u>	nombre_cliente	calle	ciudad_cliente
C-123	Lopez	San Martin	CRD
C-230	Sosa	Rivadavia	TRW
C-212	Herrera	Alem	MDY
C-235	Torres	Canada	CRD
C-189	Williams	Gales	TRW

nombre_sucursal	ciudad_sucursal	activos
Norte	CRD	1.800.000
Sur	TRW	420.000
Oeste	MDY	80.000
Centro	CRD	12.500.000
Roca	CRD	10.000.000

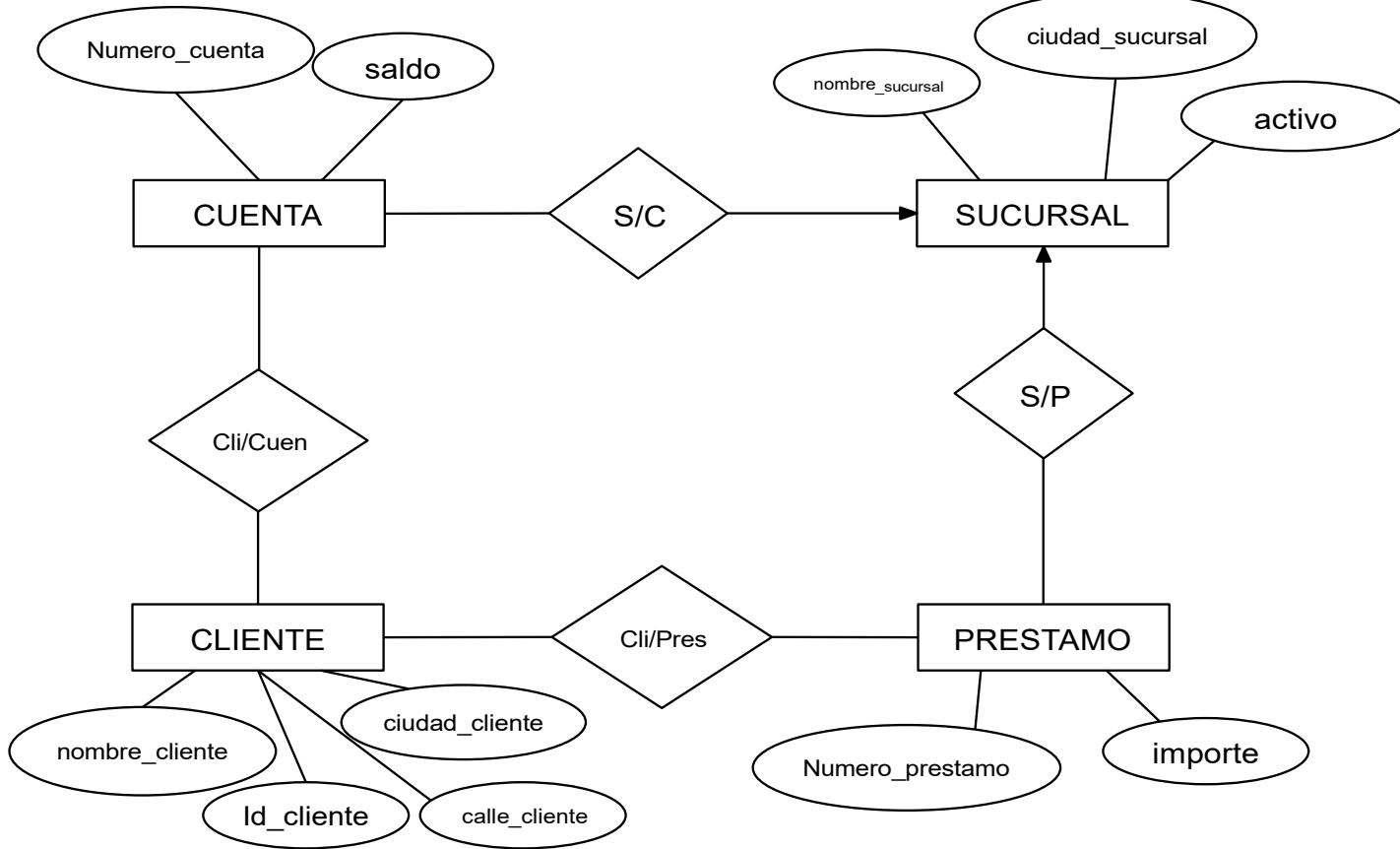
El esquema para estas relaciones es:

- ✓ Esquema_cliente=(id_cliente, nombre_cliente, calle, ciudad_cliente)
- ✓ Esquema_sucursal=(nombre_sucursal, ciudad_sucursal, activos)

Observe que el atributo *ciudad_sucursal* y *ciudad_cliente* que aparece en los esquemas de relaciones *SUCURSAL* y *CUENTA* → El uso de atributos comunes en esquemas de relaciones es una forma de relacionar tuplas de distintas relaciones.

Por ejemplo, supóngase que queremos información sobre todas las cuentas abiertas en sucursales ubicadas en CRD. Primero miramos la relación sucursal para encontrar todas las sucursales sitas en CRD. Después, para cada una de ellas, miraríamos en la relación cliente para encontrar información sobre las cuentas abiertas en esas sucursales.

Supongamos tener el siguiente diagrama E-R para la empresa bancaria:



Los esquemas resultantes son:

- ✓ Esquema_sucursal=(nombre_sucursal, activo, ciudad_sucursal)
- ✓ Esquema_cliente=(id_cliente, nombre_cliente, calle_cliente, ciudad_cliente)
- ✓ Esquema_cuenta=(nombre_sucursal, número_cuenta, saldo)
- ✓ Esquema_préstamo=(nombre_sucursal, número_préstamo, importe)
- ✓ Esquema_cli/cuen=(id_cliente, número_cuenta)
- ✓ Esquema_cli/Pres= (id_cliente, numero_prestamo)

Mostremos ahora las relaciones cli/cuen y cli/pres:

id_cliente	número_cuenta
C-123	101
C-230	215
C-212	102
C-235	305
C-123	222

id_cliente	número_préstamo
C-212	17
C-230	23
C-189	15
C-235	14
C-230	93

Puede parecer más fácil para un usuario pensar en términos de un esquema de relaciones con toda la información en lugar de varias relaciones asociadas. Supóngase que usamos sólo una relación para nuestro ejemplo:

✓ Esquema_info_cta=(nombre_sucursal, ciudad_sucursal, activos, nombre_cliente, calle, ciudad, numero_cuenta, saldo)

Observe qué ocurre si un cliente tiene varias cuentas → debemos repetir cierta información varias veces. Esta repetición no es necesaria y se evita mediante el uso de relaciones.

Si una sucursal recién creada aún no tiene clientes, no tiene ninguna cuenta y no podemos construir una tupla en esquema_info_cta, ya que no hay datos disponibles. Para representar tuplas incompletas debemos usar valores nulos. Así, los valores para calle, ciudad, numero_cuenta etc. deberían ser nulos.

Usando varias relaciones, podemos representar la información de las sucursales de un banco sin clientes, sin usar valores nulos. Simplemente usamos una tupla en esquema_sucursal para representar la información sobre la sucursal y no creamos tuplas en esquema_cliente ni en esquema_cuenta, hasta que la información esté disponible.

No siempre es posible eliminar los valores nulos. Supóngase que incluimos el atributo número_teléfono en el esquema_cliente. ¿Puede ser que un cliente no tenga un número de teléfono?

Claves

Las nociones de superclave, clave candidata y clave primaria, también se aplican al modelo relacional.

Por ejemplo, en esquema_sucursal:

- ✓ {nombre_sucursal} y {nombre_sucursal, ciudad_sucursal} son superclaves
- ✓ {nombre_sucursal, ciudad_sucursal} no es una clave candidata, ya que {nombre_sucursal} es un subconjunto de {nombre_sucursal, ciudad_sucursal} y {nombre_sucursal} es una superclave.
- ✓ El atributo {nombre_sucursal}, es una clave candidata y será clave primaria.
- ✓ El atributo ciudad_sucursal no es una superclave, puesto que dos sucursales de la misma ciudad pueden tener nombres diferentes (y diferentes activos).

Por ejemplo, en esquema_cliente:

- ✓ La clave primaria es Id_cliente.

- Ejercicios Diagrama E-R

- Vamos a desarrollar el siguiente caso:

- ✓ Se quiere diseñar una BD relacional para gestionar los datos de los socios de un club náutico.
 - ✓ De cada socio se guardan los datos personales: número de socio, nombre, apellido, dirección, y los datos del barco o barcos que posee: número de matrícula, nombre, número del amarre y valor de cuota que paga por el mismo.
 - ✓ Un barco tiene un único dueño.
 - ✓ Además, se quiere mantener información sobre las salidas realizadas por cada barco, como la fecha y hora de salida, y el destino.
 - a) Suponga que un barco sale a navegar siempre el mismo día de la semana a una hora determinada y con un destino único.
 - b) Suponga que se desea almacenar un historial con todas las salidas realizadas por el barco.

- Ejercicio 1

- Una mutualista de la ciudad posee un sanatorio y tiene un conjunto de médicos que trabajan en ella. La información relevante de cada médico es su nombre, matrícula y especialidad. Además, se mantiene información de los puntos de contacto de cada profesional y se representan de tres formas: dirección postal, teléfono y dirección de e-mail. Los mismos se clasifican en primarios y secundarios según la probabilidad de encontrar al profesional en dicho punto. Las direcciones postales se registran con su calle, número y localidad. De una localidad interesa su nombre y código postal. Los teléfonos se registran indicando, aparte del número, si son fijos o móviles. Del e-mail se registra simplemente su dirección.
- El sanatorio posee ambulancias y se registran los datos de cada una, como ser matrícula, modelo, datos del conductor y según la fecha, el médico asignado. Al comenzar cada año la mutualista planifica la asignación de los médicos a guardias e interesa registrar la fecha.

- **Ejercicio 2**
 - Obtener el esquema correspondiente (tablas).
 - Según el esquema responda los siguientes ítems y justifique en cada caso:
 - I. ¿Es posible almacenar todos los diagnósticos por fecha que un paciente se realizó en la clínica?.
 - II. ¿Es posible determinar con qué obra social se atendió en cada consulta?
 - III. ¿Se puede determinar el nombre del doctor que atendió a un paciente en cada consulta?

