



Bases de Datos

U.T. 7

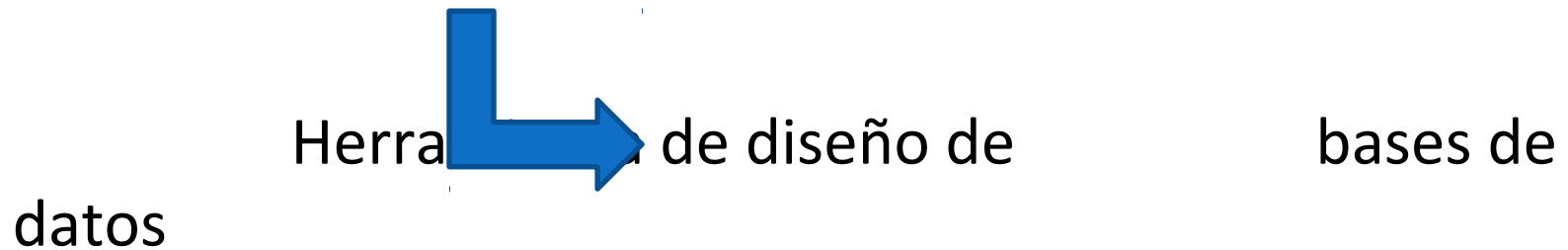
Diseño lógico relacional de Bases de Datos

Representación del problema

- ❑ Una base de datos representa la información contenida en algún dominio del mundo real.
- ❑ El diseño de base de datos consiste en extraer todos los datos relevantes de un problema.
- ❑ Se debe realizar un análisis en profundidad del dominio del problema para conocer:
 - ❑ datos esenciales para la base de datos
 - ❑ descartar los que no son de utilidad
- ❑ Una vez extraídos los datos: MODELIZACIÓN

Modelización

- ❑ Construir un esquema que exprese con total exactitud todos los datos que se quieren almacenar en la bd.



Especificación de Requisitos Software

- ❑ Los informáticos analizan un problema a través de diversas reuniones con los futuros usuarios del sistema.
- ❑ De estas reuniones, se extrae la **ERS**.
 - ❑ Estructura del documento estándar IEEE 830

Realizar actividad 2.1

ERS según estándar IEEE 830

1 Introducción

- 1.1 Propósito
- 1.2 Ámbito del Sistema
- 1.3 Definiciones, Acrónimos y Abreviaturas
- 1.4 Referencias
- 1.5 Visión general del documento

2 Descripción General

- 2.1 Perspectiva del Producto
- 2.2 Funciones del Producto
- 2.3 Características de los usuarios
- 2.4 Restricciones
- 2.5 Suposiciones y Dependencias
- 2.6 Requisitos Futuros

3 Requisitos Específicos

- 3.1 Interfaces Externas
- 3.2 Funciones
- 3.3 Requisitos de Rendimiento
- 3.4 Restricciones de Diseño
- 3.5 Atributos del Sistema
- 3.6 Otros Requisitos

4 Apéndices

5 Índice

El modelo de datos

- ❑ Modelar: representar el problema realizando múltiples abstracciones.
- ❑ Para modelar:
 - ❑ Se ha de contar con la experiencia de un futuro usuario de la base de datos que conozca pormenores del negocio
 - ❑ Seguir filosofía estándar para que el resto de la comunidad informática pueda entender y comprender el modelo realizado
 - ❑ Hay que tener en cuenta el SGBD a utilizar

Tres modelados – Tres pasos

□ *Modelo Conceptual*

- Capacidad expresiva para comunicarse con el usuario
 - Entidad / Relación => Notación Chen / Notación UML

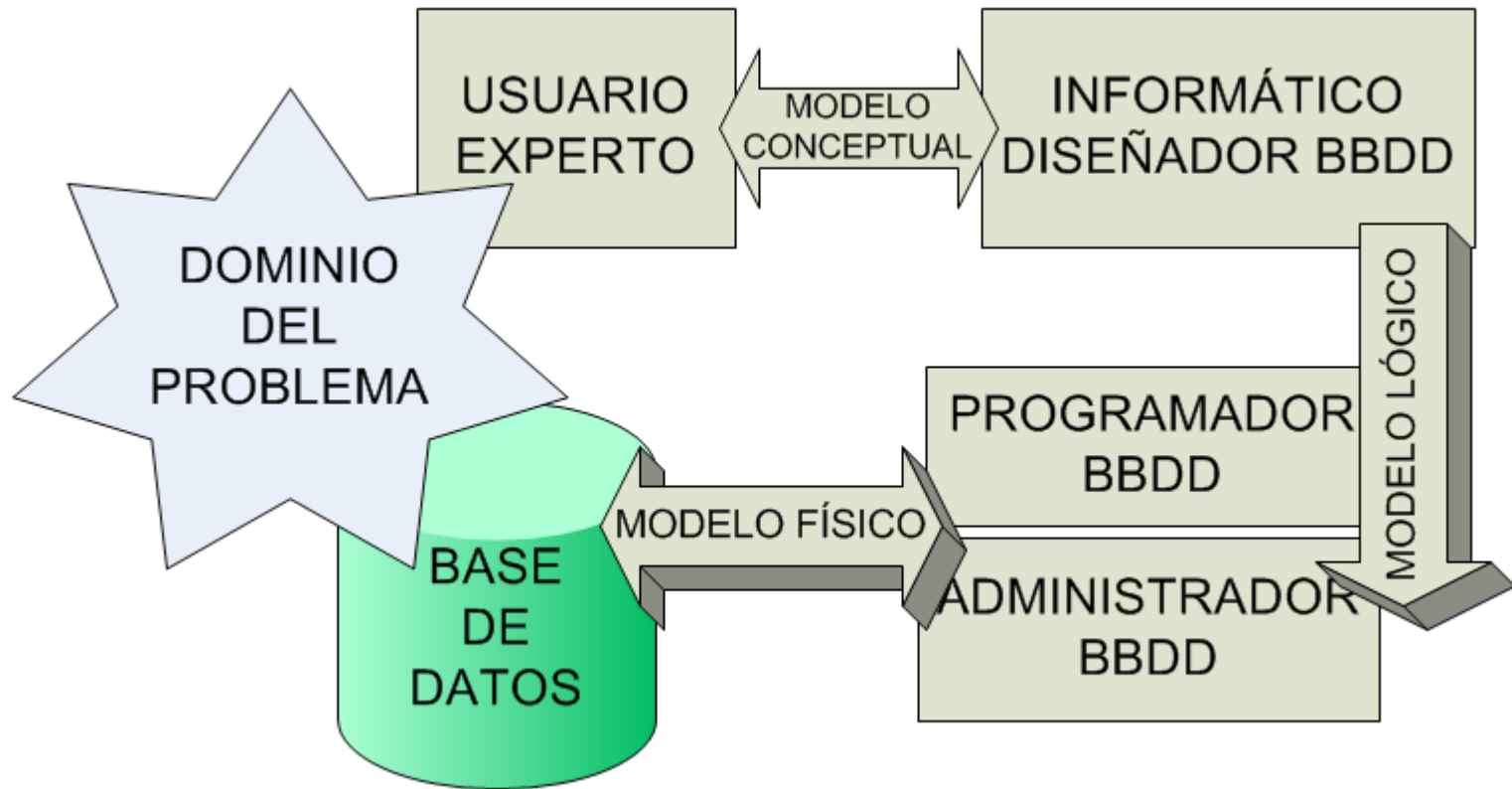
□ *Modelo Lógico*

- Más técnico que el conceptual.
- Traducción directa al modelo físico.
- Depende de la implementación de la base de datos
 - Modelo Relacional / Modelo Orientado a objetos

□ *El modelo físico*

- Resultado de aplicar el modelo lógico a un SGBD concreto.
- Utiliza SQL para expresarlo (sentencias DDL)

Interacción entre modelos

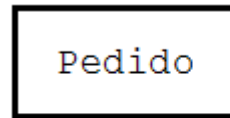


El modelo E/R. Diagramas

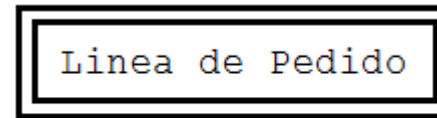
☐ Entidad

- ☐ Fuerte

- ☐ Débil



Entidad Fuerte



Entidad Débil

☐ Ocurrencia de una entidad

- ☐ Instancia particular de una entidad

- ☐ Entidad: Coche, Instancia: Seat Ibiza TDI 1442-FHD

☐ InterRelación

- ☐ asociación entre dos o más entidades

- ☐ tiene un nombre que describe su función

- ☐ clasificadas según su **grado**

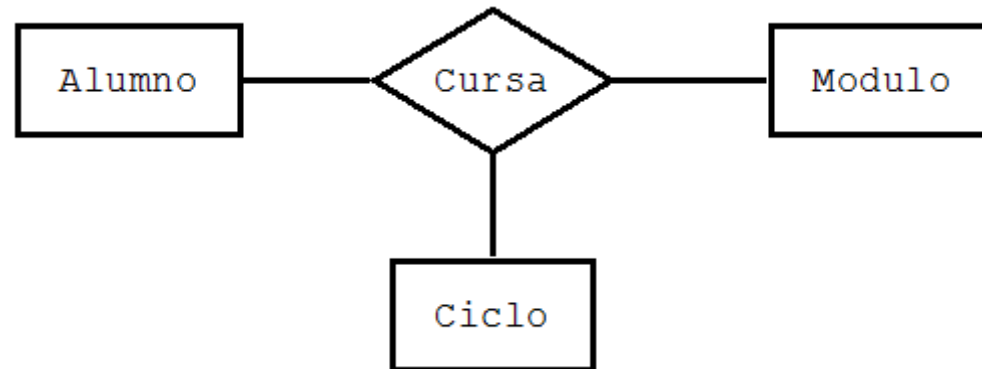
Diagramas E/R

Grado Relaciones

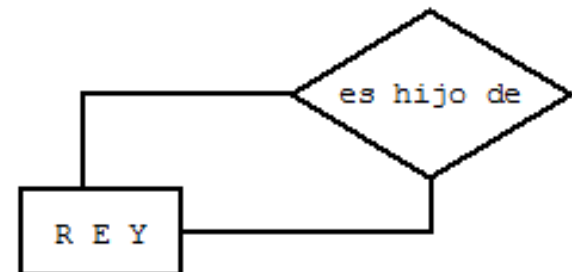
Binarias



Ternarias



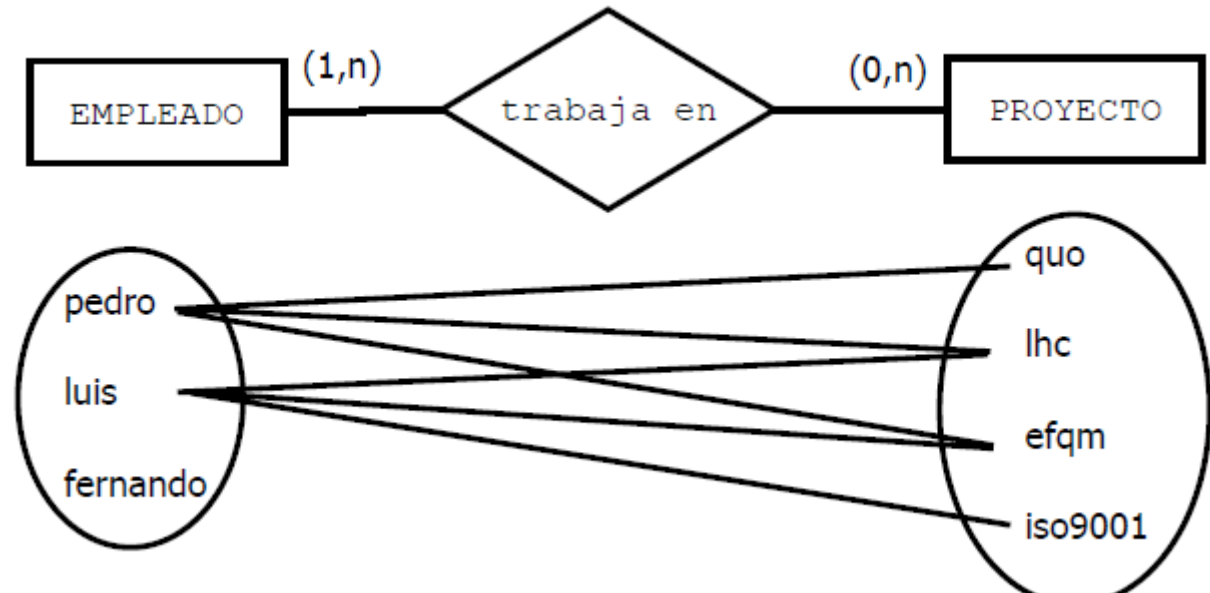
Unarias o reflexivas



Diagramas E/R

Participación

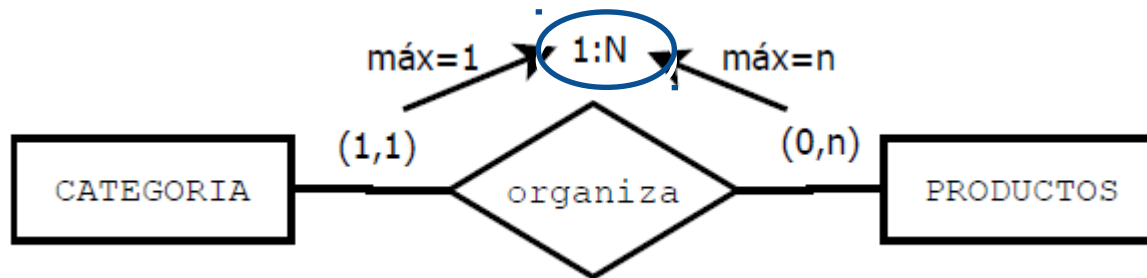
- mínimo y máximo número de veces que una ocurrencia de entidad puede aparecer en la relación asociada a otra ocurrencia de entidad
 - $(0,1)$, $(1,1)$, $(0,n)$, $(1,n)$



Realizar actividades 2.2 a 2.4

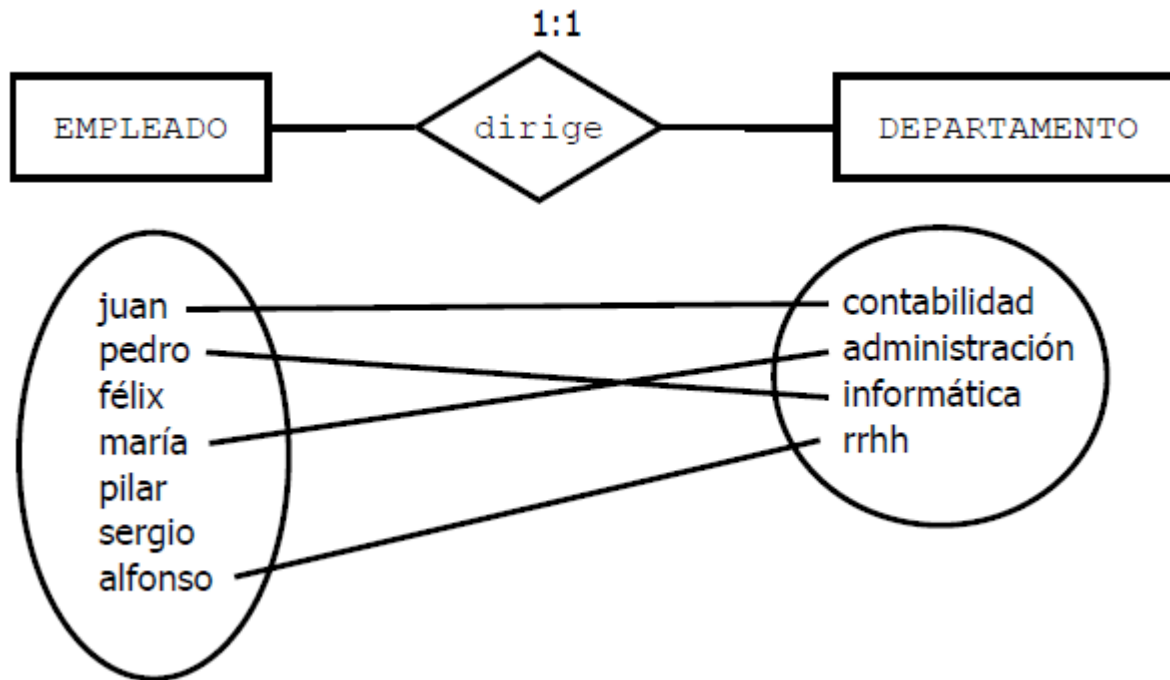
Cardinalidad de una relación

- Se toman el número máximo de participaciones de cada una de las entidades en la relación.



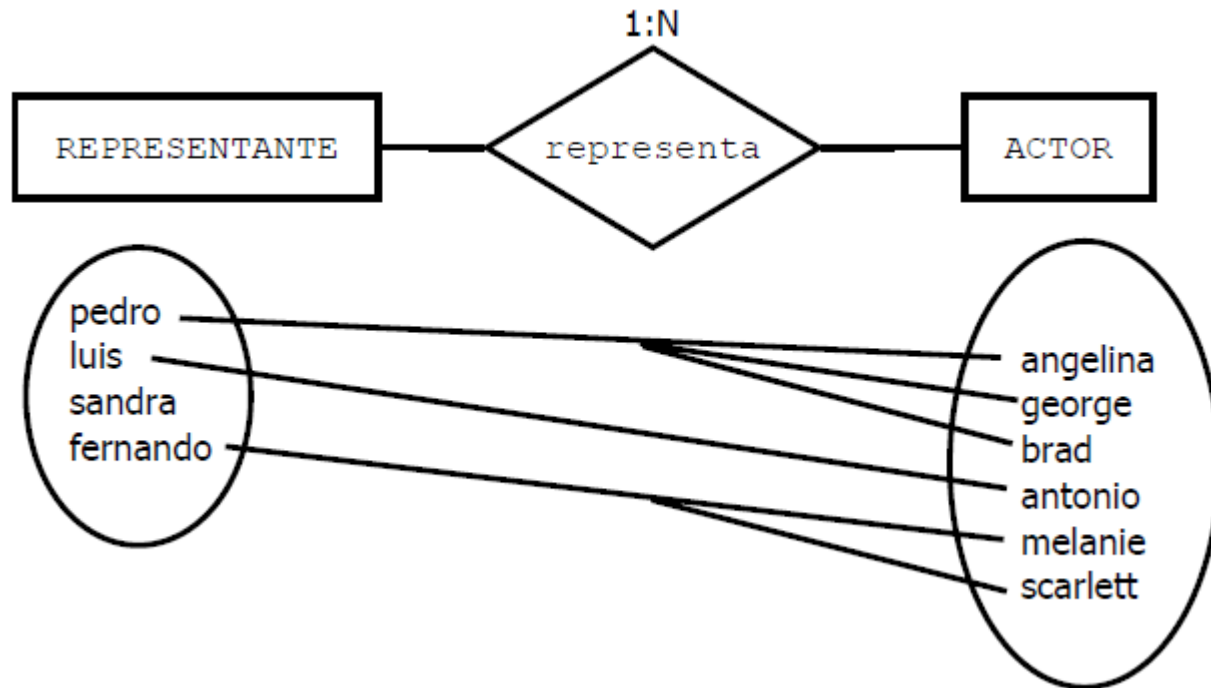
Tipos de cardinalidad

Cardinalidad 1:1



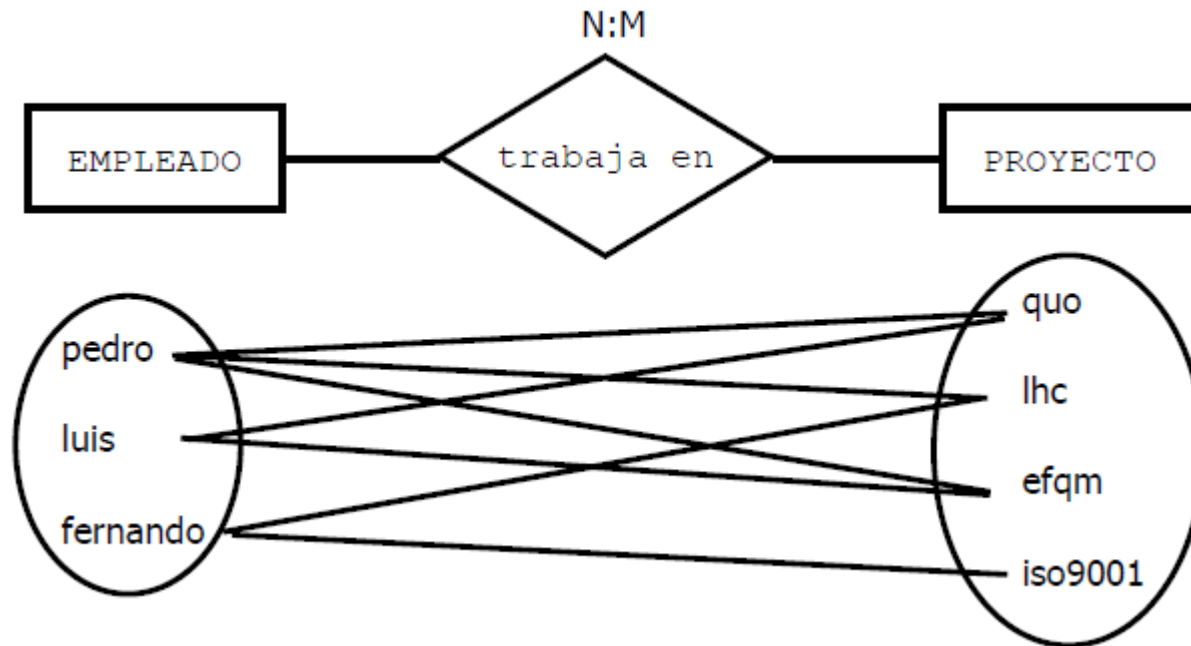
Tipos de cardinalidad

Cardinalidad 1:N



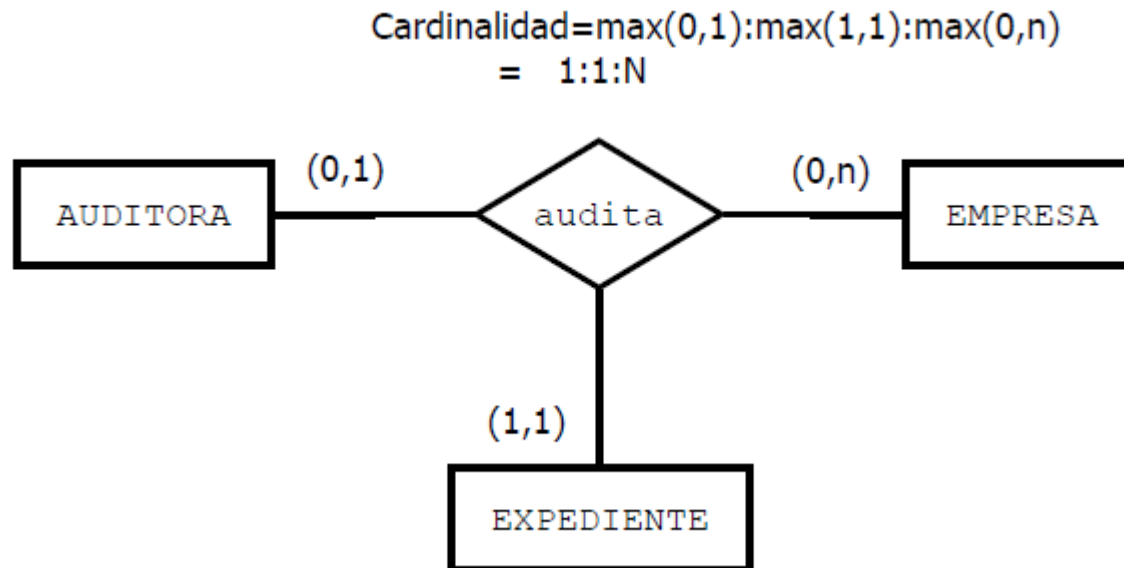
Tipos de cardinalidad

Cardinalidad N:M



Tipos de cardinalidad

- En las relaciones no binarias
 - se toma una de las tres entidades y se combinan las otras dos

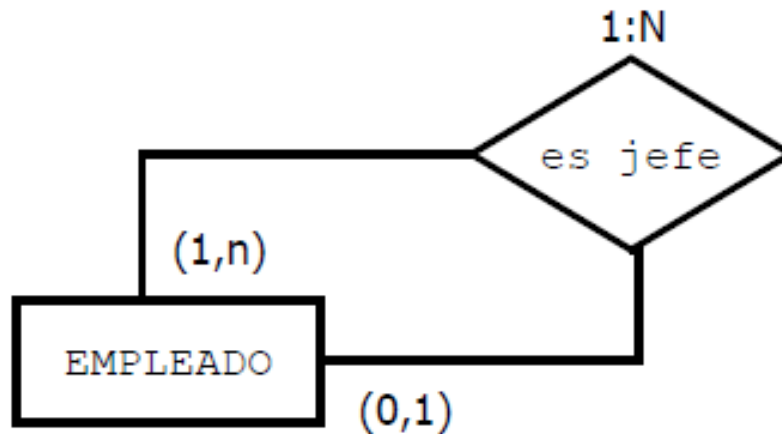


Realizar actividades 2.5 - 2.8

Tipos de cardinalidad

❑ Relaciones reflexivas

- ❑ extraer las participaciones según los dos roles existentes

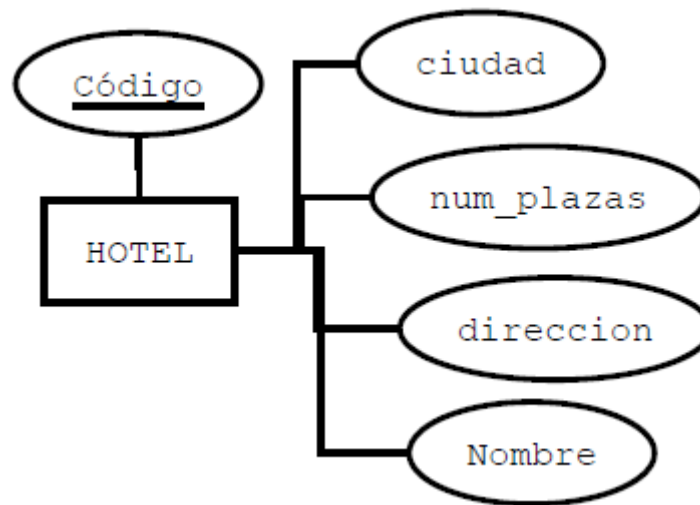


Realizar actividad 2.9

Diagramas E/R

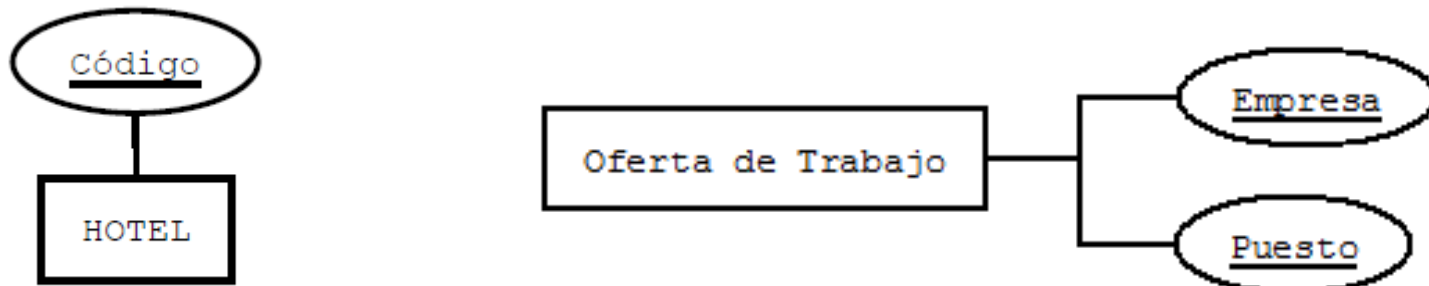
■ Atributos

- características o propiedades que la definen como entidad



Diagramas E/R

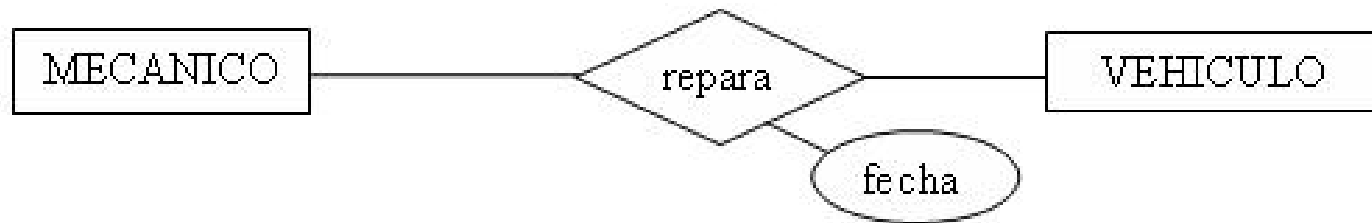
- ❑ Atributo clave
 - ❑ campo que identifica de forma unívoca cada ocurrencia de entidad
 - ❑ no puede repetir ninguna ocurrencia de entidad
 - ❑ puede ser compuesto



Diagramas E/R

❑ Atributo de relación

- ❑ Es propio de una relación y no puede ser cedido a las entidades que intervienen en la relación



Diagramas E/R

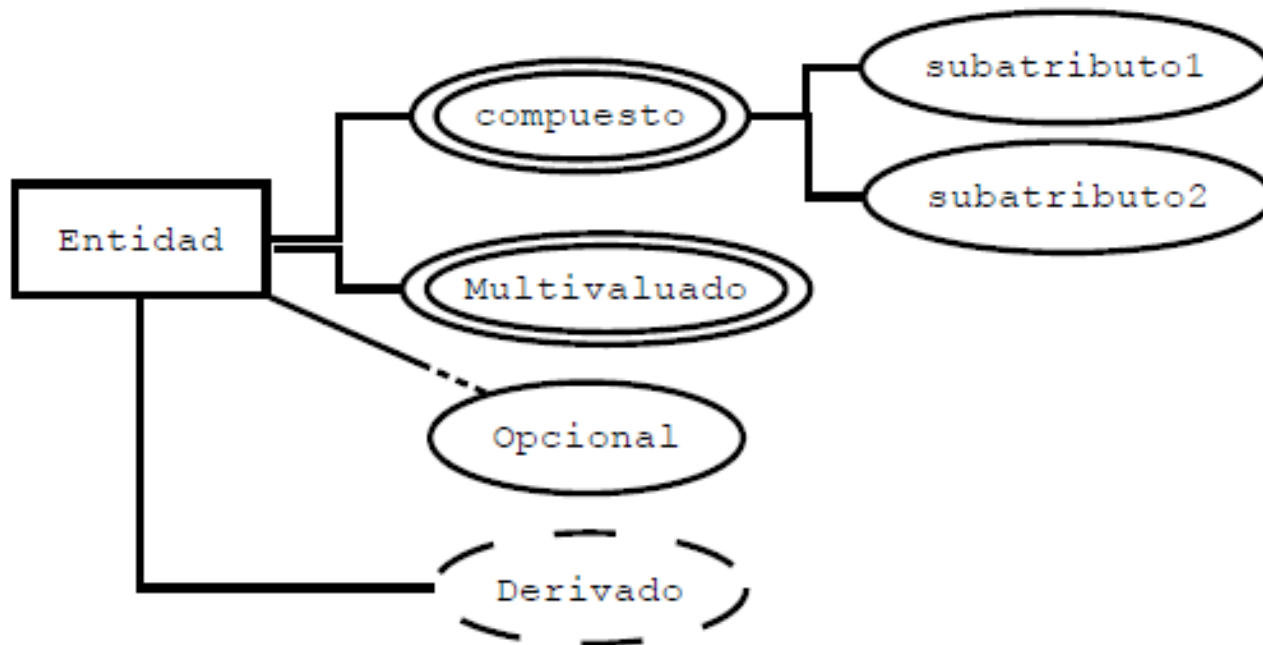
□ Dominios:

□ representan la naturaleza del dato

Atributo	Dominio
DNI	Cadena de Caracteres de longitud 10
Nombre	Cadena de Caracteres de longitud 50
Fecha_Nacimiento	Fecha
Dirección	Cadena de Caracteres de longitud 100
Sueldo	Números reales
Número de hijos	Números enteros
Departamento	Departamentos

Diagramas E/R

Tipos de atributos

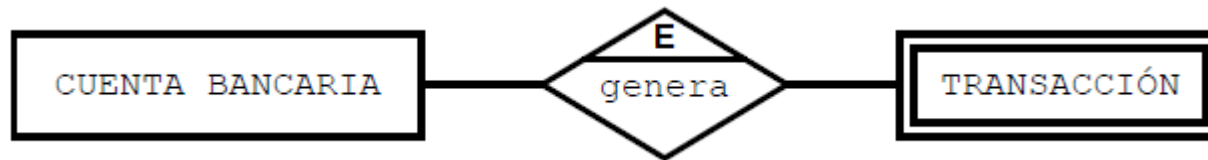


Realizar actividad 2.10

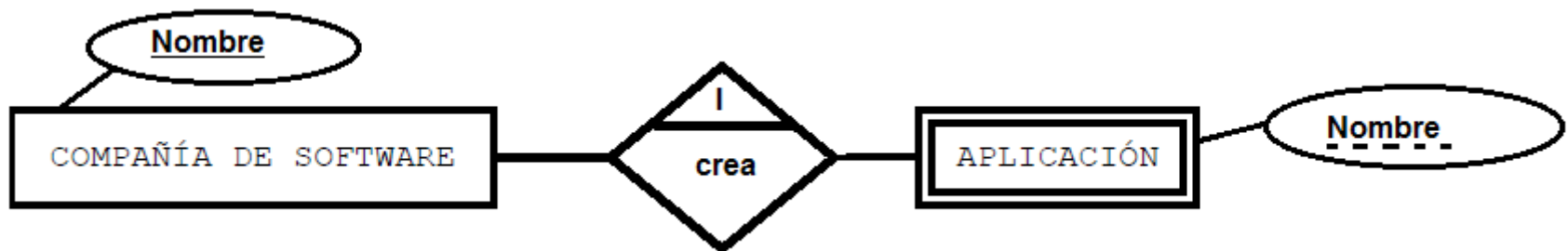
Diagramas E/R

Las entidades débiles

- Dependen de una entidad fuerte mediante una relación
- Esta relación también es débil
 - Dependencia de existencia

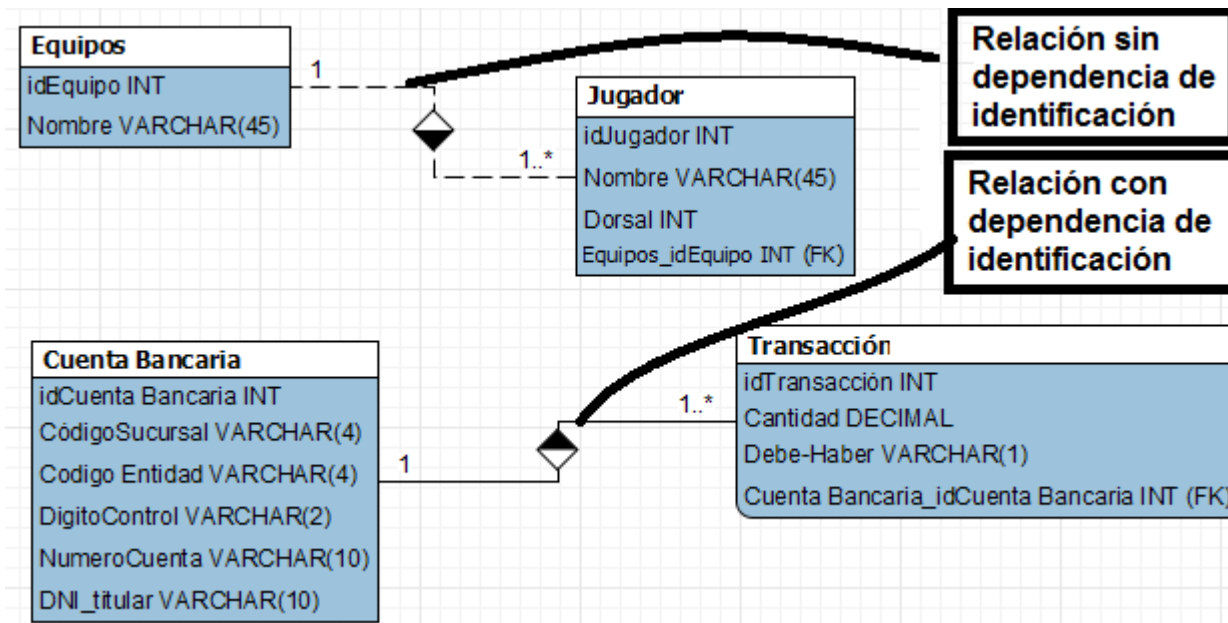


- Dependencia de identificación



Otras notaciones

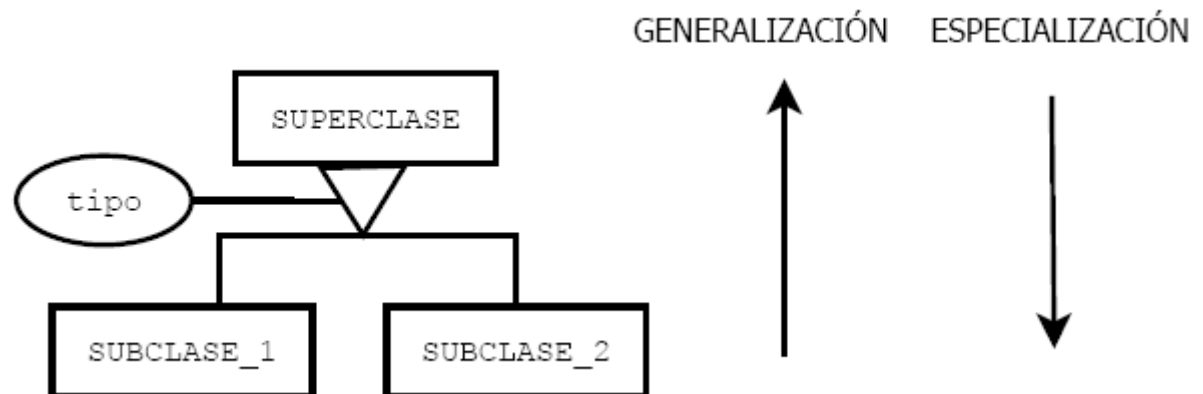
MySQL Workbench



Realizar actividades 2.11-2.12

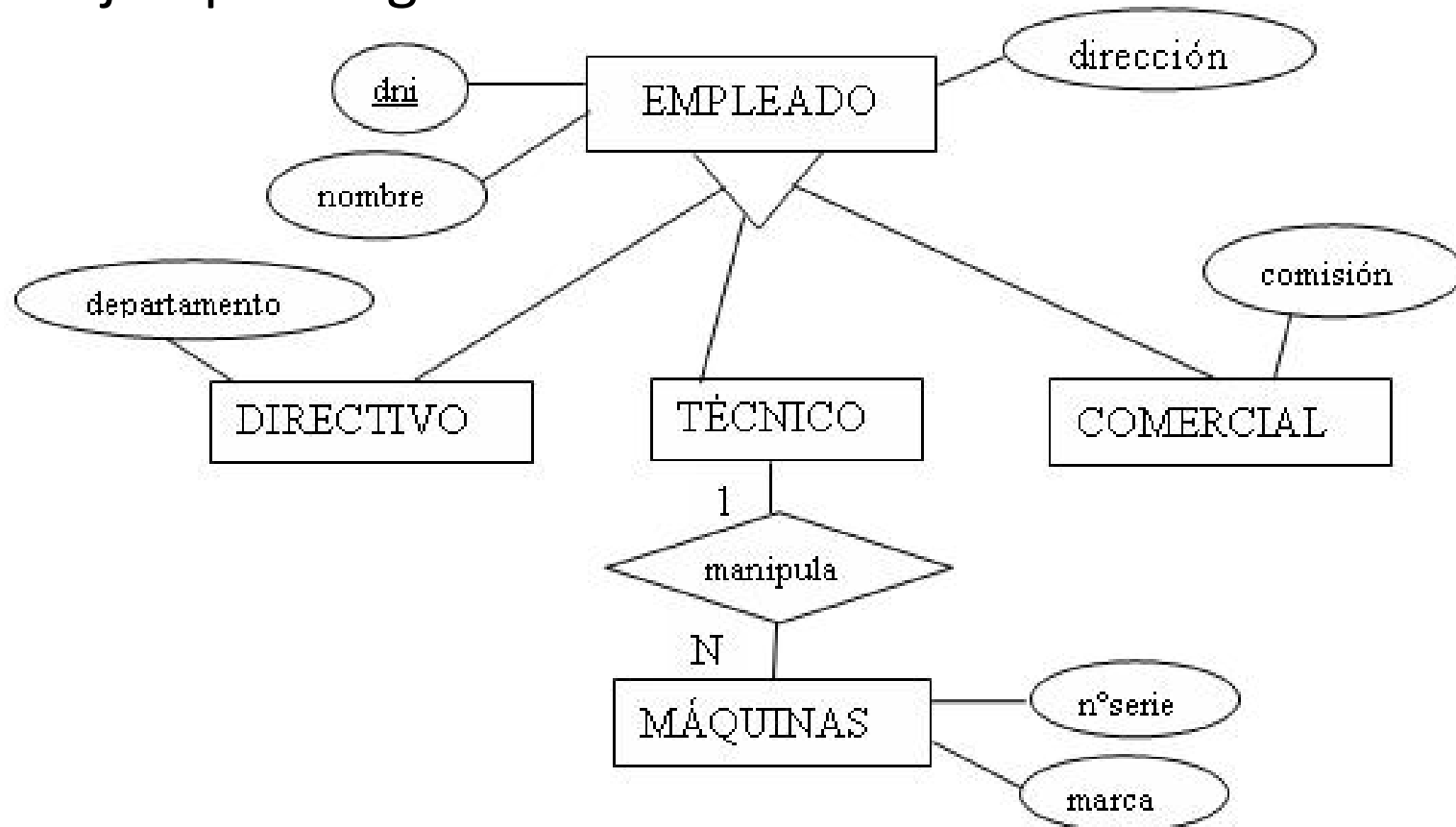
El modelo E/R ampliado

- ❑ Incluye relaciones jerárquicas
- ❑ Generalización y Especialización
 - ❑ Una entidad E es una generalización de un grupo de entidades E_1, E_2, \dots, E_n si cada ocurrencia de cada una de esas entidades es también una ocurrencia de E .
 - ❑ Se heredan los atributos de la superclase



El modelo E/R ampliado

□ Ejemplo de generalización



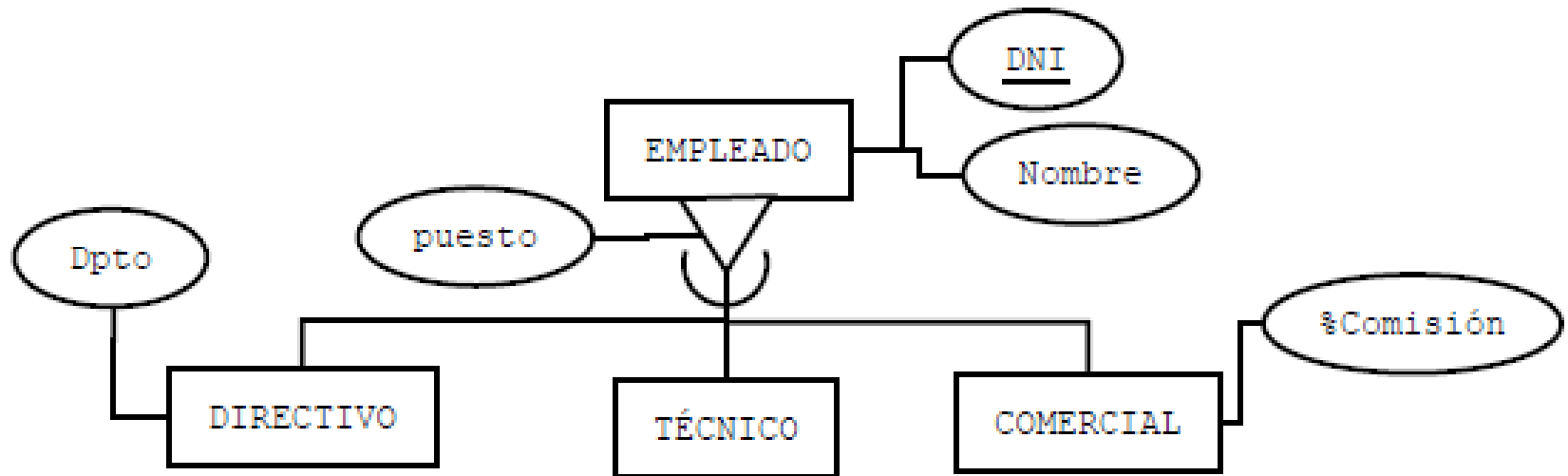
Tipos de especialización

- ☐ Exclusiva (sólo una especialización)
- ☐ Inclusiva (más de una especialización)
- ☐ Total (obligatoria)
- ☐ Parcial (opcional)
- ☐ Se pueden combinar

Tipos de especialización

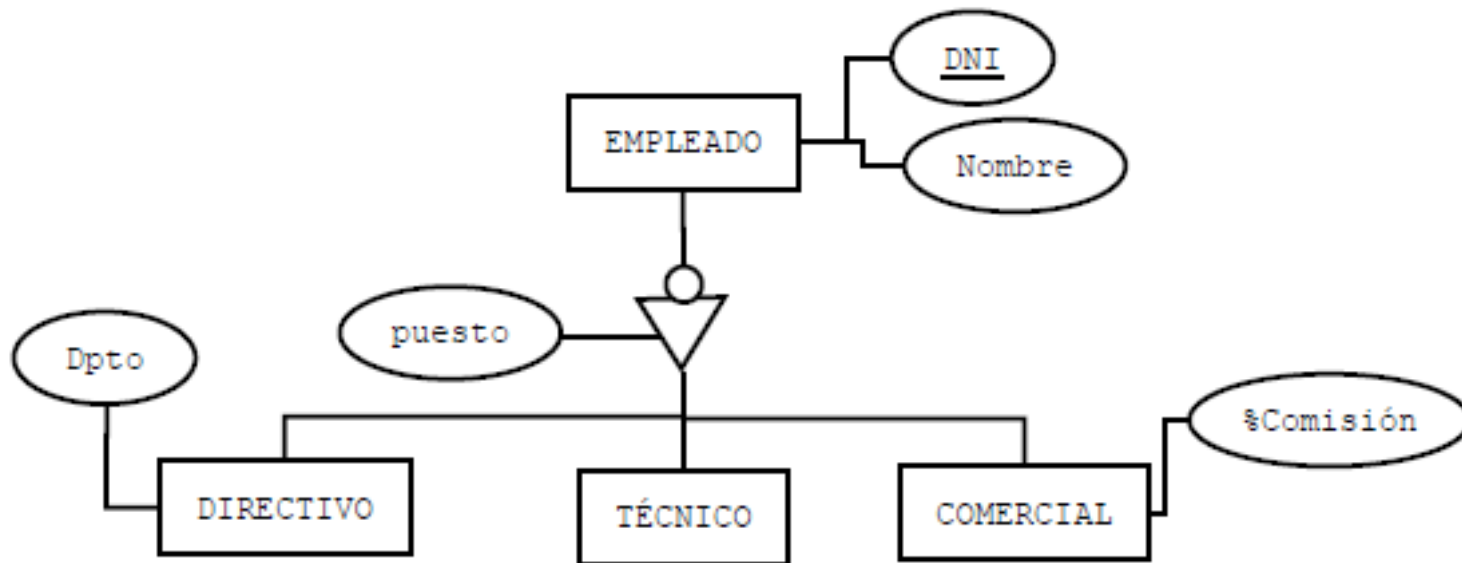
❑ Especialización Exclusiva-Parcial

- ❑ Un empleado sólo puede ser técnico, directivo ó comercial. Además puede no ser ningún perfil específico



Tipos de especialización

❑ Especialización Inclusiva-Total



Realizar actividad 2.13

Construcción de un diagrama E/R

Leer varias veces el problema hasta memorizarlo.

Obtener una lista inicial de candidatos a entidades, relaciones y atributos.

Refinar el diagrama.

Si hay dudas, ERS o usuario:

- ☐ Reiniciar análisis si es necesario

Construcción de un diagrama E/R

2. Obtener una lista inicial de candidatos a entidades, relaciones y atributos

- ☐ Identificar las entidades -> nombres comunes
- ☐ No obsesionarse en los primeros pasos por distinguir las entidades fuertes de las débiles.
- ☐ Extraer los atributos de cada entidad -> adjetivos
 - ☐ Distinguir los atributos clave y tipos de atributos
- ☐ Generalizaciones si se obtiene un atributo que es aplicable a más de una entidad.
- ☐ Identificar atributos de relaciones.

Construcción de un diagrama E/R

2. Obtener una lista inicial de candidatos a entidades, relaciones y atributos

- ☐ Es posible que los nombres comunes contengan muy poca información y no sea posible incluirlas como entidades. En este caso, se pueden seleccionar como atributos de otra entidad
- ☐ Extraer los dominios de los atributos
- ☐ Identificar las relaciones -> verbos
- ☐ Afinar cómo afecta la relación a las entidades implicadas: distinguir entidades débiles y fuertes

Construcción de un diagrama E/R

2. Obtener una lista inicial de candidatos a entidades, relaciones y atributos

- ☐ Averiguar las participaciones y cardinalidades
- ☐ Colocar todos los elementos en un mapa
- ☐ Replantear si es necesario su pertenencia a una categoría

El modelo relacional

❑ Modelo Físico

- ❑ Objetivo: Proteger al usuario de la obligación de conocer tecnología BBDD.
- ❑ Relación es el elemento fundamental
 - ❑ Usuarios ven la base de datos como una colección de relaciones
 - ❑ Modelo independiente de dónde se almacenan los datos.
 - ❑ Fuerte base matemática. Teoría de conjuntos

Relación: propiedades

- ❑ Cada relación tiene un nombre diferente.
- ❑ Cada atributo de la relación tiene un único valor en cada tupla.
- ❑ Cada atributo tiene un nombre diferente en cada tabla (aunque pueden coincidir en tablas diferentes).
- ❑ Cada tupla es única (no hay tuplas duplicadas).
- ❑ El orden de los atributos no importa.
- ❑ El orden de las tuplas no importa.

Las relaciones en el Modelo Relacional

Relación

Conjunto de atributos que pertenecen a un dominio

Relación: ImpuestoVehículos					Nombre	Esquema
Vehículo	Dueño	TeléfonoDueño	Matrícula	Cuota	Cabecera	
Seat Ibiza TDI 1.9	Francisco	925884721	9918-FTV	92,24	Cuerpo	Estado
Volkswagen Polo TDI 1.0	Pedro	918773621	4231-FHD	61,98		
Renault Laguna Coupé	María	929883762	7416-GSJ	145,32		
Fiat Punto 1.0	Ernesto	646553421	9287-BHF	45,77		

Restricciones

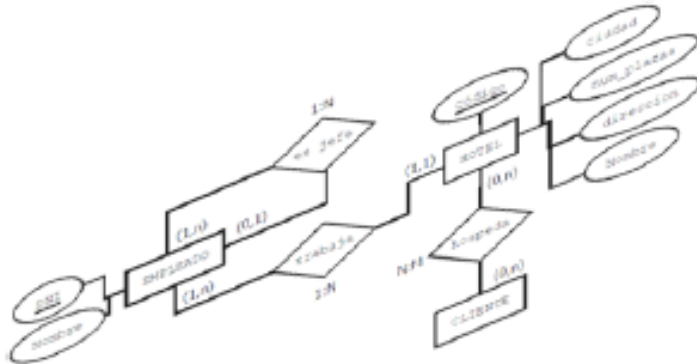
- ❑ Condiciones que deben cumplir los datos para su correcto almacenamiento:
 - ❑ Garantizan la consistencia de los datos
 - ❑ De clave
 - ❑ Valor Único (UNIQUE)
 - ❑ Integridad Referencial (REFERENCES)
 - ❑ De dominio
 - ❑ Verificación (CHECK)
 - ❑ Valor Nulo (NULO)
 - ❑ Disparadores
 - ❑ Genéricas o aserciones (ASSERT)

Claves

- ❑ Conjunto de atributos que identifican de forma única una ocurrencia de entidad
- ❑ Tipos
 - ❑ Superclave
 - ❑ Pueden ser no mínimas
 - ❑ Clave Candidata
 - ❑ Mínima superclave
 - ❑ Clave Primaria
 - ❑ Candidata elegida
 - ❑ Clave Foránea
 - ❑ Atributo que es clave en otra entidad

Transformación E/R -> Relacional

MODELO CONCEPTUAL (CHEN)

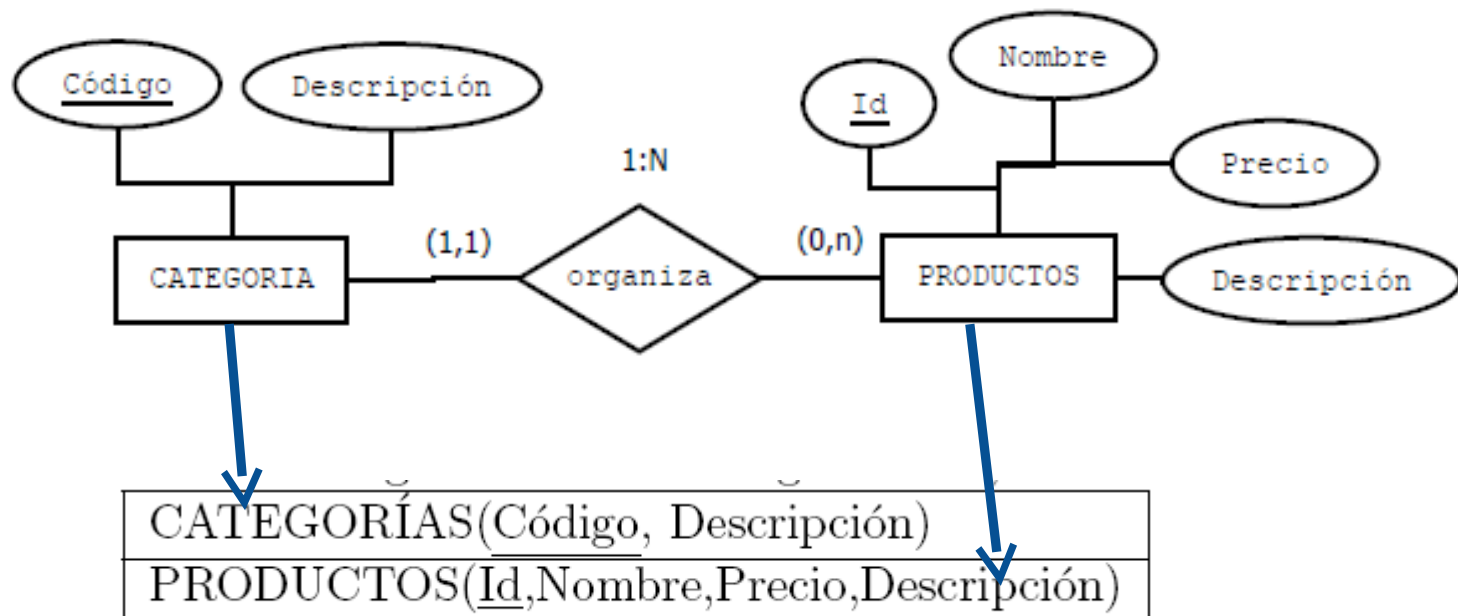


REGLAS DE TRANSFORMACIÓN

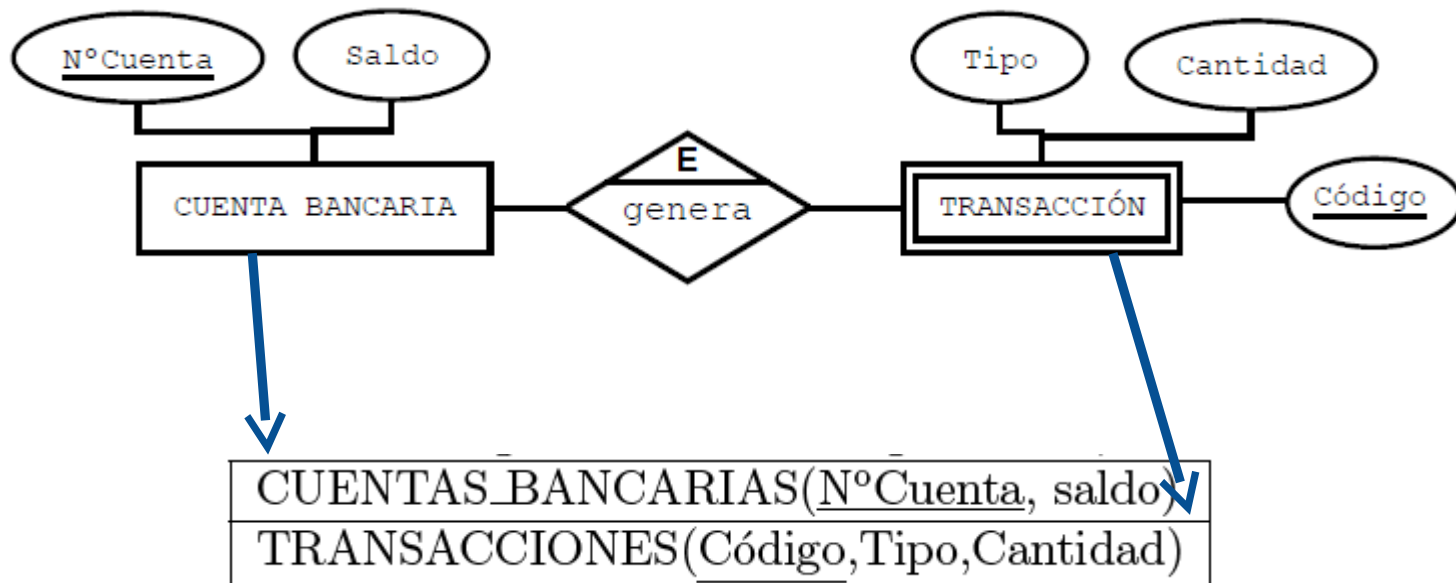
MODELO RELACIONAL (CODD)



Transformación: Entidades Fuertes

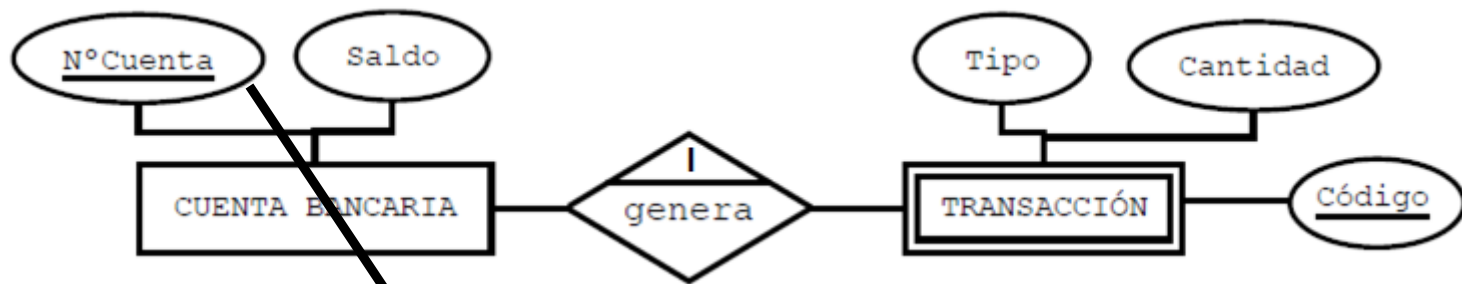


Transformación: Entidades Débiles



Transformación: Entidades Débiles

- ❑ Si es con dependencia de identificación



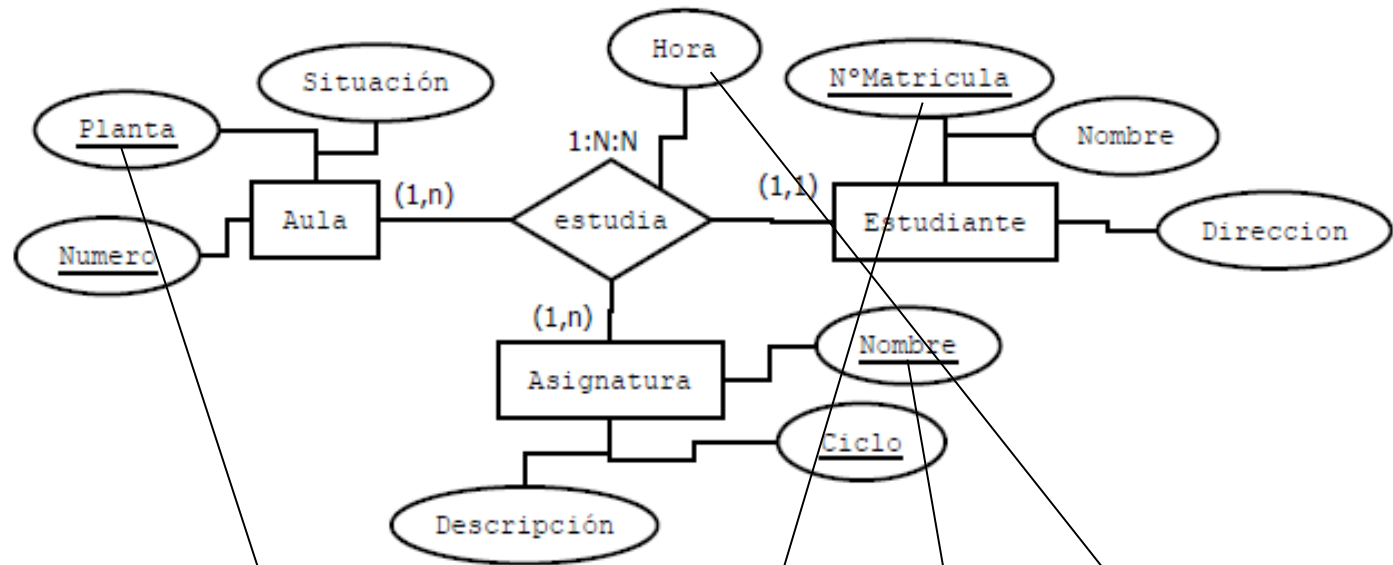
CUENTA BANCARIA (NºCuenta, Saldo)

TRANSACCION(NºCuenta, Codigo, Tipo, Cantidad)

Transformación: Relaciones

Regla general

- Una tabla por cada entidad y una tabla para la relación.
- La tabla de la relación importa los campos clave de las entidades relacionadas y los atributos de la relación



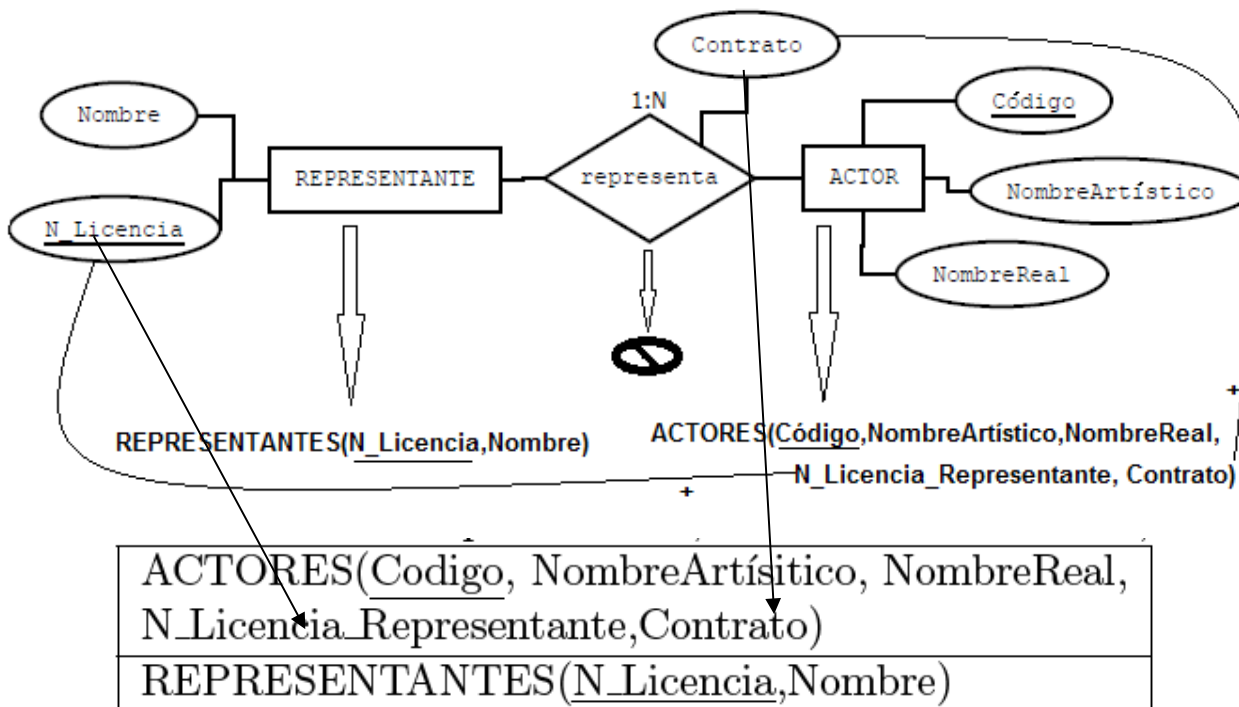
AULAS(<u>Numero</u> , Planta, Situación)
ESTUDIANTES(<u>N°Matricula</u> , Nombre, Dirección)
ASIGNATURAS(Nombre, <u>Ciclo</u> , Descripción)
ESTUDIOS(<u>Numero</u> , Planta, <u>N°Matricula</u> , Nombre, <u>Ciclo</u> , Hora)

Transformación: Relaciones

❑ Excepciones a la regla general: 1:N

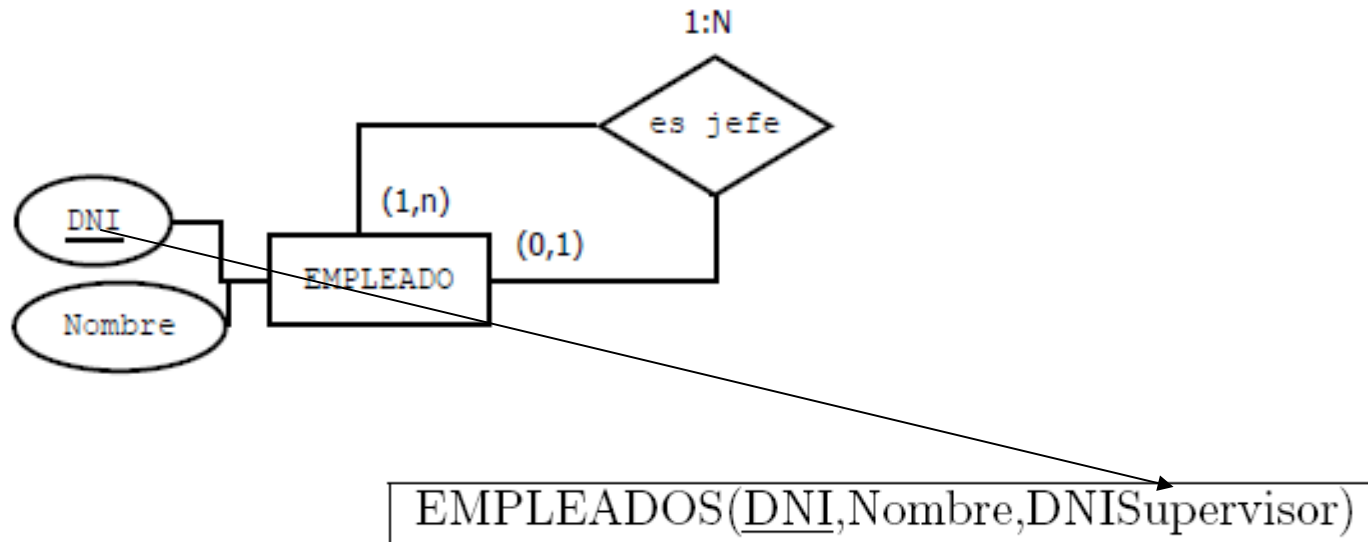
con participaciones (0/1,n) ----- (0/1,1)

❑ La relación no genera tabla. Se importa la clave de la entidad con participación máxima 1 a la entidad con participación máxima N.



Transformación: Relaciones

- ❑ Excepciones a la regla general: 1-N reflexivas
 - ❑ Se añade otra vez la clave cambiada de nombre



Realizar actividad 2.14

Transformación: Relaciones

- Relaciones 1:1: Varias opciones, dependiendo de las participaciones mínimas de las entidades implicadas.

- Los dos tipos de entidad participan de forma completa

(1,1) ----- (1,1)

- Si las dos entidades tienen el mismo identificador, se genera una única tabla formada por la agregación de los atributos de ambas entidades y el mismo identificador como campo clave.
- Si tienen diferente identificador, cada entidad se transforma en una tabla con su propio identificador y además, cada tabla recibe la propagación de clave de la otra entidad.

Transformación: Relaciones

- Relaciones 1:1: Varias opciones, dependiendo de las participaciones mínimas de las entidades implicadas.

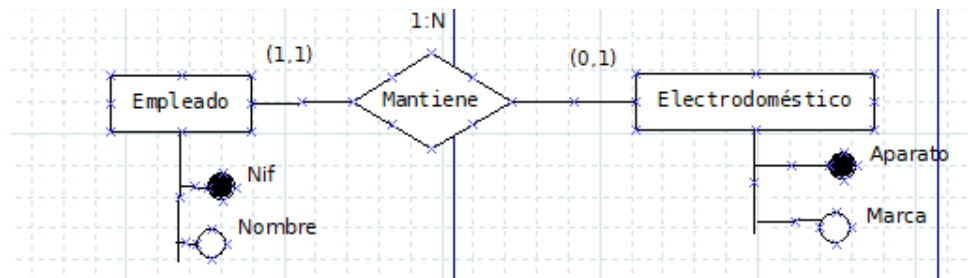
- Alguno de los tipos de entidad participa de forma parcial

(0,1) ----- (1,1)

- Cada entidad se transforma en una tabla con su propio identificador y además, el identificador del tipo de entidad que participa parcialmente pasa como atributo a la otra tabla.

Empleado(**nif**, nombre)

Electrodoméstico(**Aparato**, Marca, nif)



Transformación: Relaciones

- Relaciones 1:1: Varias opciones, dependiendo de las participaciones mínimas de las entidades implicadas.

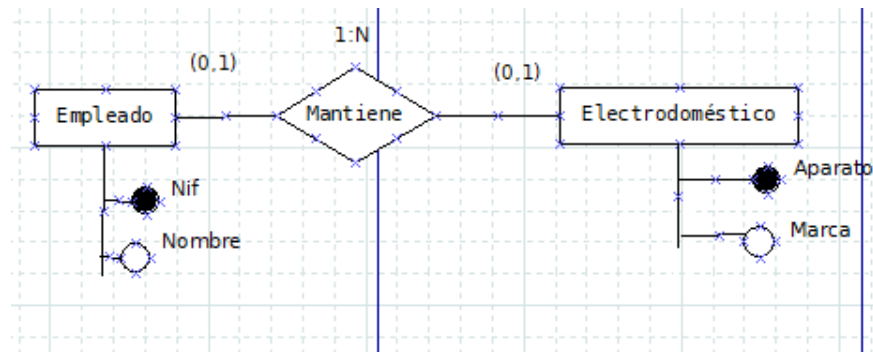
- Ambos tipos de entidad participan de forma parcial

(0,1) ----- (0,1)

- Cada entidad se transforma en una tabla con su propio identificador y además, los identificadores de ambos tipos de entidad pasan como atributos de la otra tabla, respectivamente.

Empleado(**nif**, nombre,aparato)

Electrodoméstico(**Aparato**, Marca, nif)

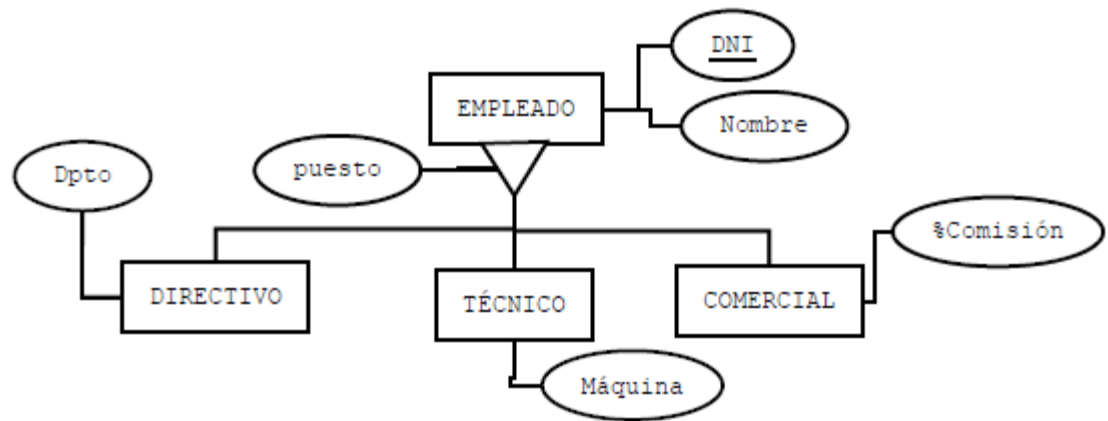


Participaciones 0,x

- ❑ Cuando la participación tiene un mínimo de 0, se debe restringir el atributo a opcional (NULL)
- ❑ Cuando la participación tiene un mínimo de 1, se debe restringir el atributo a obligatorio (NOT NULL)

Transformación de jerarquías

□ 4 opciones



EMPLEADOS(DNI, Nombre, Puesto, Dpto, Máquinas, Comisión, Tipo)

EMPLEADOS(DNI, Nombre, Puesto, Dpto, Máquinas, Comisión,
EsDirectivo, EsTécnico, EsComercial)

DIRECTIVOS(DNI, Nombre, Puesto, Dpto)

TÉCNICOS(DNI, Nombre, Puesto, Máquinas)

COMERCIALES(DNI, Nombre, Puesto, Comisión)

EMPLEADOS(DNI, Nombre, Puesto)

DIRECTIVOS(DNI, Dpto)

TECNICOS(DNI, Máquinas)

COMERCIALES(DNI, Comisión)

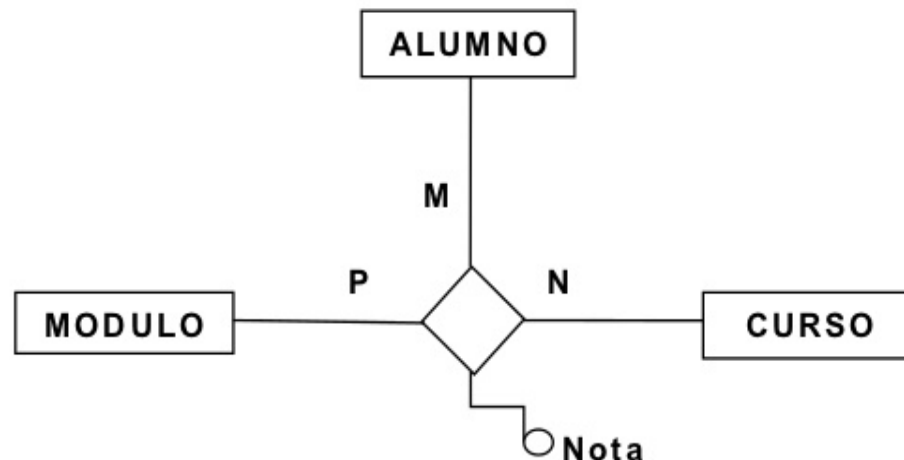
Transformación de relaciones ternarias

Relaciones en las que intervienen tres entidades.

La transformación de estas relaciones al modelo relacional siempre da lugar a una nueva relación que contendrá como atributos las claves primarias de las tres entidades interrelacionadas y todos los atributos que tenga la interrelación. La clave primaria de la nueva tabla depende de la conectividad de la interrelación.

Conectividad M:N:P

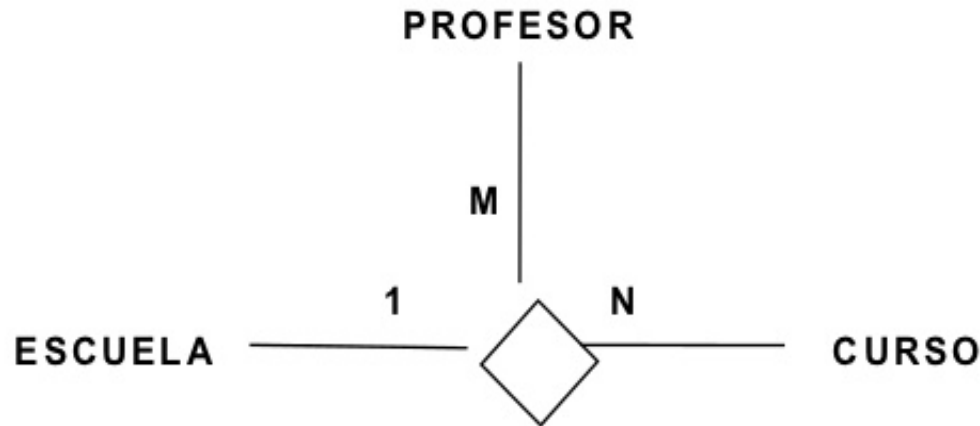
La tabla que se obtiene de la interrelación ternaria tiene como clave primaria todos los atributos que forman las claves primarias de las tres entidades interrelacionadas.



Transformación de relaciones ternarias

Conectividad M:N:1

La tabla que se obtiene de la interrelación ternaria tiene como clave primaria todos los atributos que forman las claves primarias de las entidades de los lados de la interrelación etiquetados con M y N.



Profesor (idProfesor, ...)

Curso (idCurso, ...)

Escuela (idEscuela, ...)

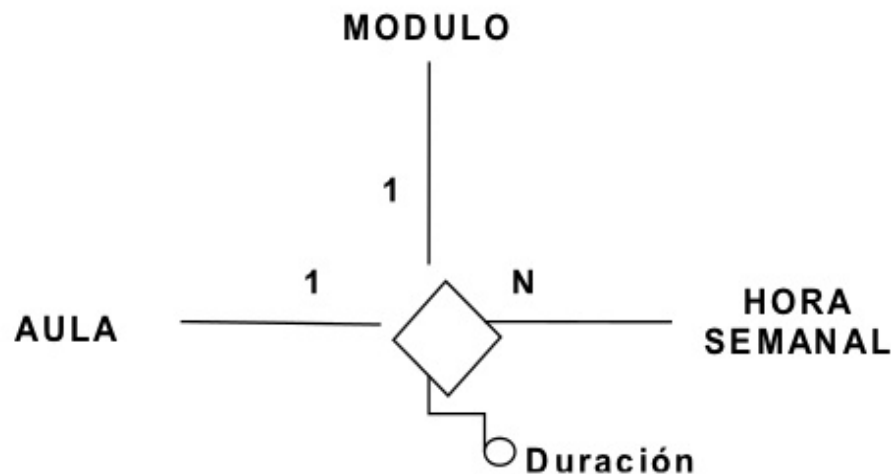
Destino (idProfesor, idCurso, idEscuela)

Transformación de relaciones

ternarias

Conectividad N:1:1

La tabla que se obtiene de la interrelación ternaria tiene como clave primaria todos los atributos que forman las claves primarias de la entidad del lado de la interrelación etiquetado con N y los atributos que forman la clave primaria de cualquiera de las dos entidades que están conectadas con 1.



Opción A:

HoraSemanal (idHora, ...)

Aula (idAula, ...)

Modulo (idModulo, ...)

Clase (idHora, idAula, idModulo,
duración)

Opción B:

HoraSemanal (idHora, ...)

Aula (idAula, ...)

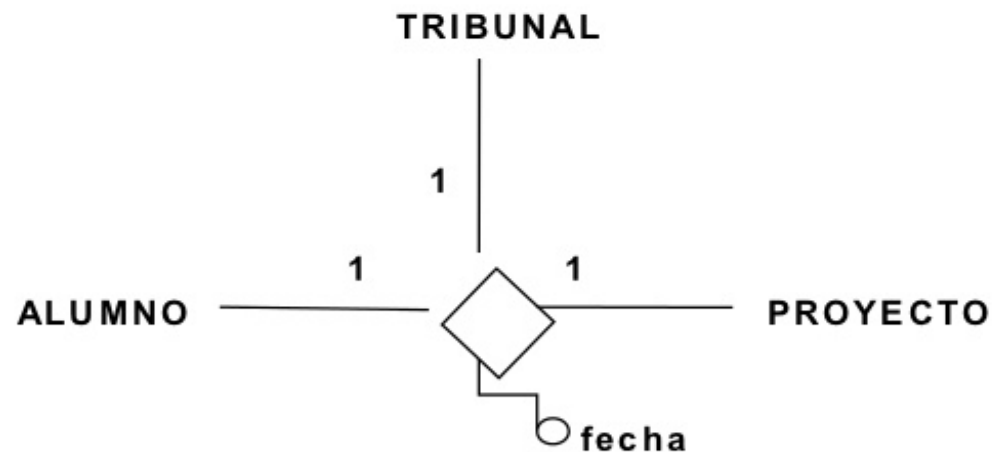
Modulo (idModulo, ...)

Clase (idHora, idAula, idModulo,
duración)

Transformación de relaciones ternarias

Conectividad 1:1:1

La tabla que se obtiene de la interrelación ternaria tiene como clave primaria todos los atributos que forman las claves primarias de dos entidades cualesquiera de las tres interrelacionadas.



Tribunal (idTribunal, ...)

Alumno (idAlumno, ...)

Proyecto (idProyecto, ...)

Opción A:

Defensa (idTribunal,
idAlumno, idProyecto,
fecha)

Opción B:

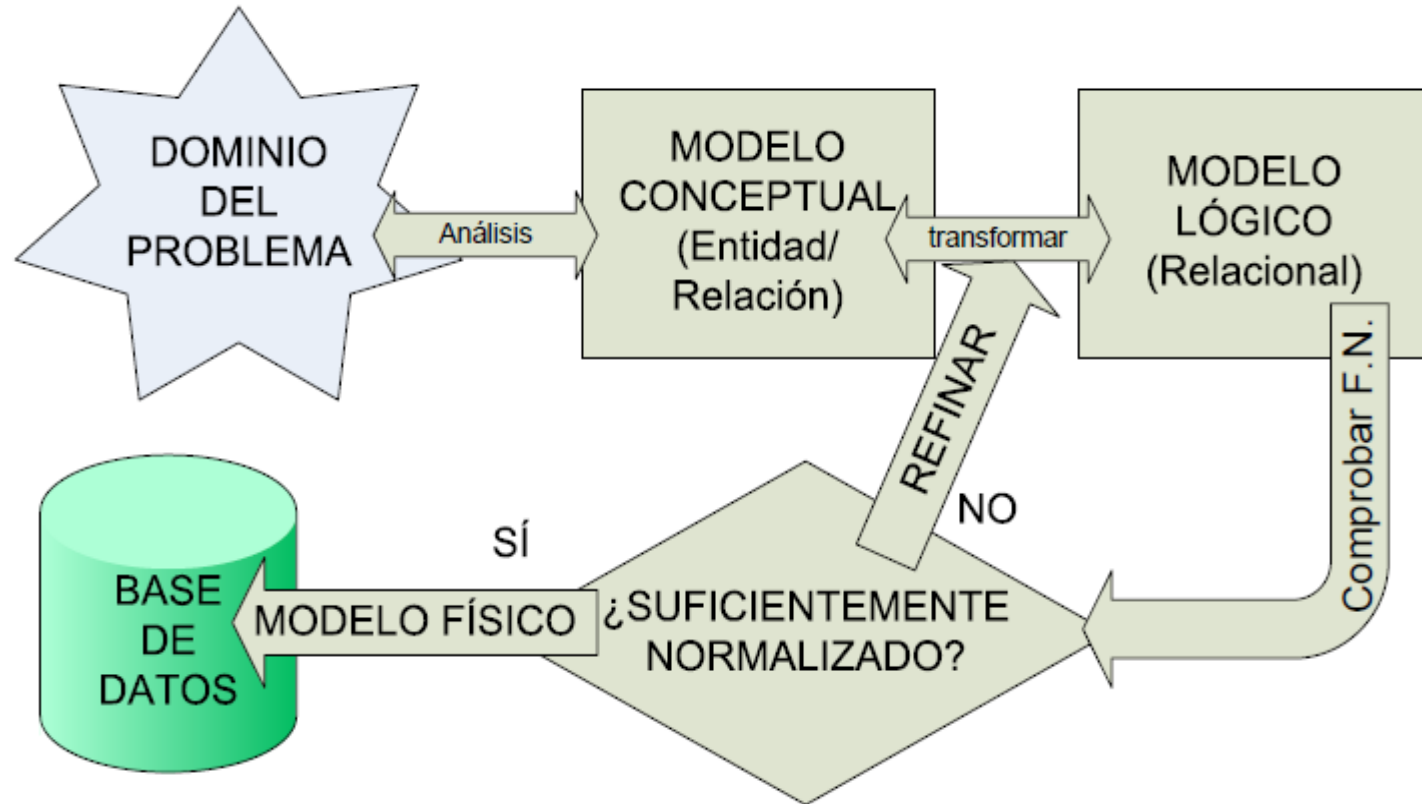
Defensa (idTribunal,
idAlumno, idProyecto,
fecha)

Opción C:

Defensa (idTribunal,
idAlumno,
idProyecto, fecha)

Normalización

- Medida de calidad de un diseño



Normalización

1. Introducción.
2. Formas normales.
 1. Primera forma normal.
 2. Segunda forma normal.
 3. Forma normal Boyce-Codd.
 4. Cuarta forma normal.
 5. Quinta forma normal.

Normalización : introducción

- La normalización es un proceso que consiste en la aplicación de unas reglas sobre las relaciones que hemos obtenido al pasar el modelo E/R al modelo relacional.
- Con estas reglas se intenta:
 - Evitar la redundancia de datos.
 - Proteger la integridad de los datos, sin dejar huérfano ningún atributo o tabla que tenga alguna dependencia.
 - Igualar la estructura del modelo relacional al sistema de base de datos: tablas con nombres únicos y columnas y atributos del mismo tipo.

Normalización: formas normales

- Las formas normales son una manera de dividir los atributos según su dependencia respecto a la clave principal. Se trata de una nomenclatura que permite estructurar los atributos de manera estándar para facilitar el trabajo con las bases de datos en proyectos complejos.

Normalización: Primera forma normal

- Una tabla se halla en la 1ª FN si todos sus atributos son atómicos, es decir, indivisibles.
- Ejemplo:

Persona (dni, nombre, dirección, **teléfonos**)



Persona (dni, nombre, dirección)

Teléfono (dni, numero_teléfono)

Normalización: Segunda forma normal

- Una tabla se encuentra en la 2ª FN si está en la 1ª FN y cada atributo que NO es clave es totalmente dependiente de la clave.
- Ejemplo:

Trabajo (dni, num_proy, horas, nomb_proy, nomb_empl)

“nomb_proy” depende únicamente de “num_proy” y “nomb_empl” de “dni”.



Trabajo (dni, num_proy, horas)

Empleado (dni, nomb_empl)

Proyecto (num_proy, nomb_proy)

Normalización: Forma normal de Boyce-Codd

- Una relación está en la FNBC si está en la 2ª FN y todos los determinantes son claves candidatas.
 - Un determinante es un atributo del cual depende por completo algún otro atributo.
- Otra forma de ver si una relación está en la FNBC es comprobando que, además de estar en la 2ª FN, no existen claves candidatas compuestas (con varios atributos) que tengan un elemento común.

Normalización: Forma normal de Boyce-Codd

- Ejemplo:

Vivienda (cod_postal, dirección, población)

Las claves candidatas de la tabla son “cod_postal, población” y “dirección, población”.



Código_dirección (cod_postal, dirección)

Código_Población (cod_postal, población)

Objetivos discutibles de la normalización

- ❑ Almacenar en la base de datos cada hecho sólo una vez
 - ❑ Evita redundancias y aprovecha el espacio 😊
 - ❑ En teoría es bueno, en la práctica puede ser un “performance killer” de la base de datos 😞
- ❑ Que los hechos distintos se almacenen en sitios distintos.
 - ❑ Evita ciertas anomalías a la hora de operar con los datos.
 - ❑ Malo para bases de datos multidimensionales (exceso de consultas para recuperar información sencilla) 😊

