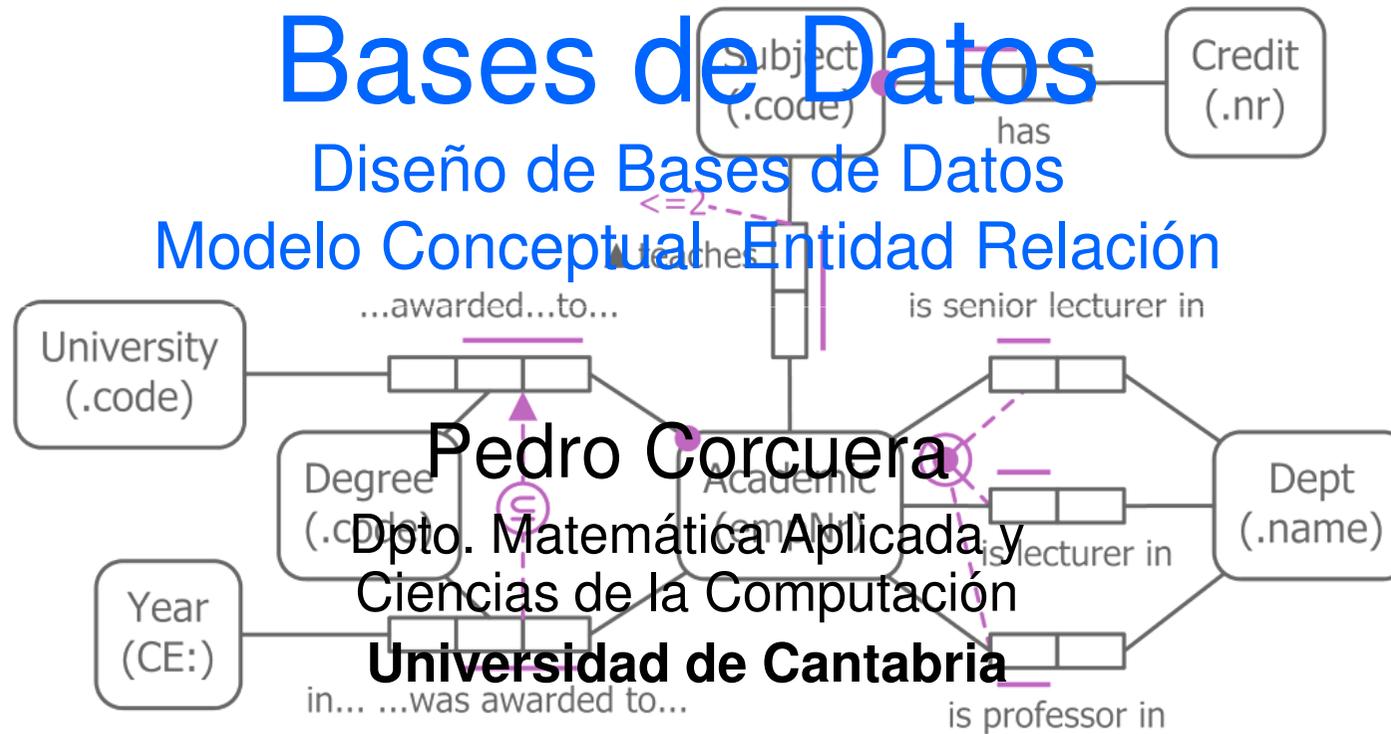


# Bases de Datos

Diseño de Bases de Datos

Modelo Conceptual Entidad Relación



**Pedro Corcuera**

Dpto. Matemática Aplicada y  
Ciencias de la Computación

**Universidad de Cantabria**

[corcuerp@unican.es](mailto:corcuerp@unican.es)

- Presentar los conceptos y técnicas del diseño de las bases de datos
- Presentar la técnica de modelado entidad-relación

- Modelos de datos
- Fases de diseño
- Modelo Entidad-Relación

- Colección de herramientas conceptuales que se emplean para especificar datos, las relaciones entre ellos, su semántica asociada y las restricciones de integridad
  - Los modelos de datos describen las relaciones entre los datos que forman una base de datos
  - No se refieren en ningún momento a los valores específicos que un elemento de datos debe tomar
  - Tratan a los datos como grupos genéricos, que pueden tomar cualquier conjunto de valores específicos

- 
- Un **modelo de datos** es una colección de conceptos para describir a los datos.
  - Un **esquema** es una descripción de una colección particular de datos usando un modelo de datos específico.
    - Un SGBD soporta un modelo de datos, que es usado para describir el esquema de la base de datos a utilizar.
  - Existen varios modelos de datos. En Bases de Datos se han usado tradicionalmente tres:
    - Jerárquico
    - Redes
    - Relacional
-

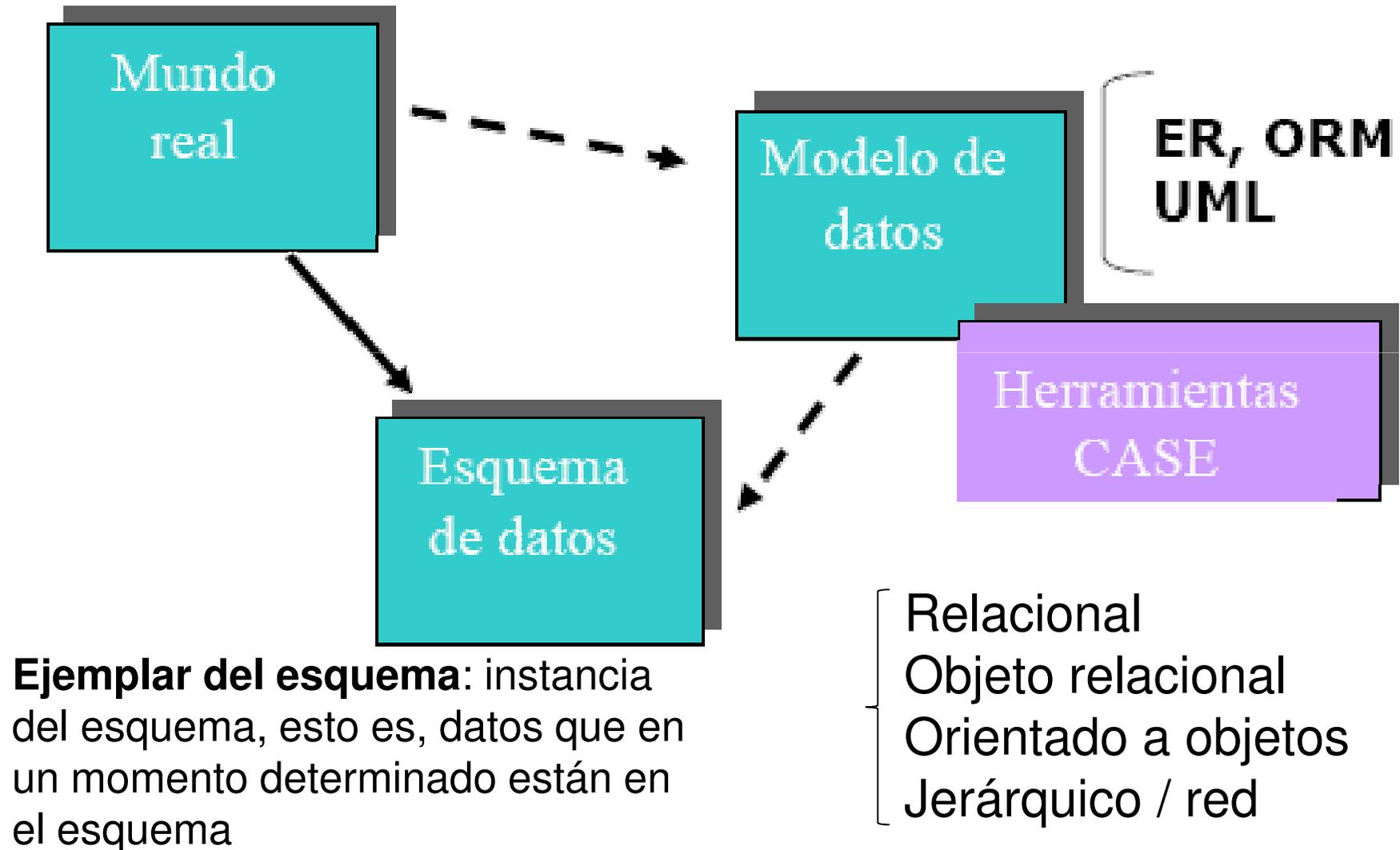
- Otros modelos incluyen:
  - Modelo Entidad-Relacion (ER)
  - Modelo Entidad Relacion Extendido (EER)
  - Modelo Orientado a Objetos
    - Lenguaje de Modelacion Unificado (UML)
  - Otros modelos semánticos (ORM)



- Fase inicial: análisis de requisitos. Descripción de la información a gestionar y sus procesos. Entrevistas con usuarios y expertos.
  - Análisis de requisitos. Especificación funcional
- Diseño conceptual: traducción del análisis de requisitos al esquema conceptual. Representación generalmente gráfica de las entidades y sus relaciones.
  - Modelo ER, modelo UML, ORM
  - DFD, diagrama de casos, diagramas de colaboración, de secuencia, etc.

- Implantación en el gestor:
  - Diseño lógico: traducción del modelo conceptual al LDD del gestor correspondiente. Modelo relacional, OO, OR
  - Diseño físico: determina la organización de archivos y las estructuras de almacenamiento interno.

# Modelo, Esquema y Ejemplar



- Características:
  - Independientes del SGBD
  - Mayor nivel de abstracción
  - Mayor capacidad semántica
  - Más enfocados al diseño de alto nivel
  - Interfaz usuario/informático

- Desarrollado por Peter Chen ( M.I.T.) en los 70's
- Es un Modelo Conceptual de alto nivel
- Se usa comúnmente para modelar aplicaciones de Bases de datos y en investigación de Bases de Datos
- Representa gráficamente y de manera lógica toda la información y como los datos se relacionan entre sí.
- Es independiente del DBMS en el cual se vaya a implementar

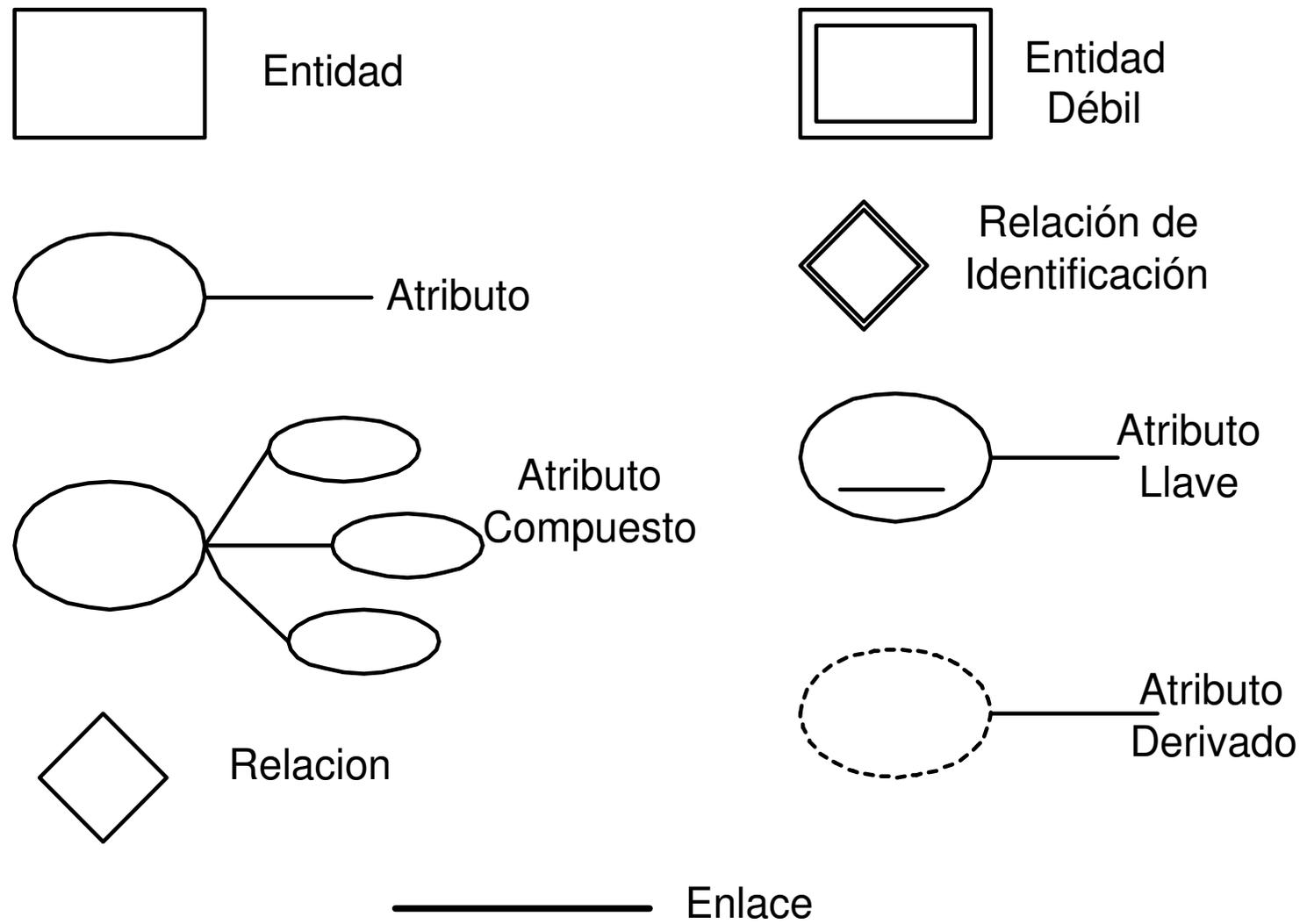
- Diseño conceptual. (ER y EER se usan en este nivel de abstracción)
  - ¿Cuáles son las entidades y relaciones en la empresa?
  - ¿Qué información acerca de esas entidades y relaciones deben almacenarse en la BD?
  - ¿Cuáles son las restricciones de integridad (o reglas de negocio) que se deben mantener?
  - Un esquema de la base de datos en estos modelos se puede representar gráficamente (Diagramas ER)
  - Los diagramas ER se pueden traducir a esquemas relacionales

- Entidad (entity)
  - Objeto que existe y se distingue de los demás
  - Se representa con una serie de atributos
  - Pueden ser concretos. pe.: un libro, una persona,..
  - O abstractas. pe.: préstamo, pedido,...
- Atributo (attribute)
  - Propiedades que caracterizan a las entidades.
  - Clave primaria: atributos que identifican a la entidad. pe.: ISBN (PK), título, idioma,... para entidad libro

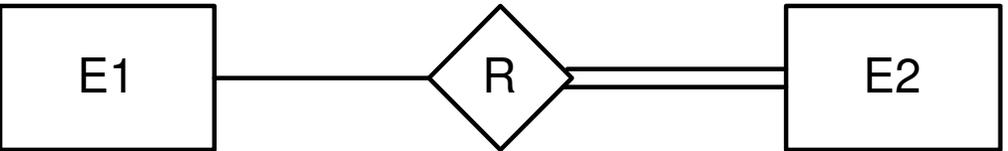
- Dominio (domain)
  - Conjunto de valores permitidos para un atributo
  - P. ej: indicando el tipo de datos (por intención)
  - P. ej: sexo-> M o F (por extensión)
- Entity Extension
  - Conjunto de elementos del mismo tipo. Los conjuntos no necesariamente deben ser disjuntos
- Relación (Relationship)
  - Conexión semántica entre dos o más entidades
- Relationship Instance
  - Conjunto de relaciones del mismo tipo

- Existen dos categorías de tipos de entidades:
  - Regulares o fuertes, que son aquellas cuyos ejemplares tienen existencia por sí mismos
    - Caso préstamos de la biblioteca: LIBRO y AUTOR
  - Débiles, en las cuales la existencia de un ejemplar depende de que exista un cierto ejemplar de otro tipo de entidad
    - Caso del EJEMPLAR que depende de LIBRO

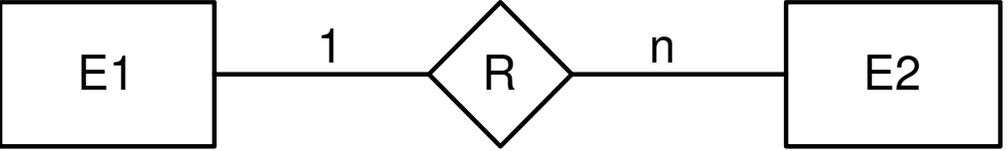
# Elementos gráficos del modelo Entidad - Relación



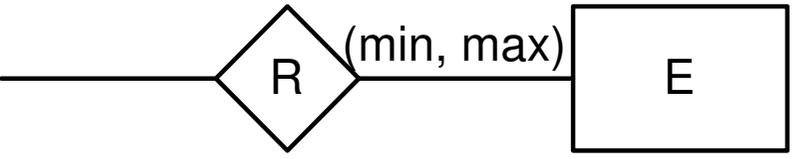
# Modelo ER



Participacion Total de E2 en R



Relación de Cardinalidad 1:n para E1:E2 en R



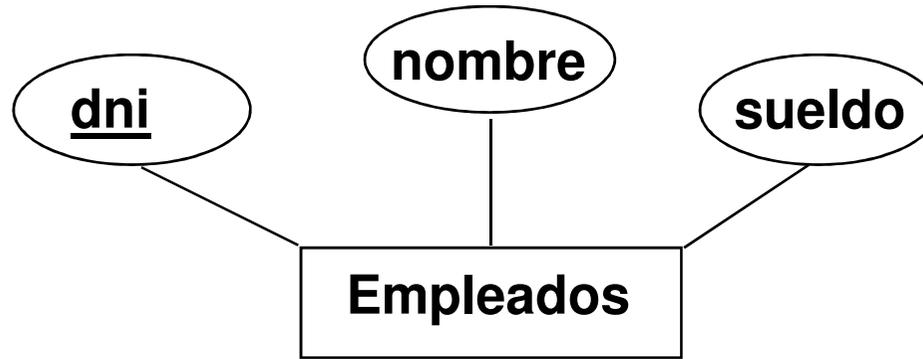
Restriccion Estructural (min, max) en la participacion de E en R

- Tres elementos básicos: entidades, atributos y relaciones.
  - ENTIDAD: Es todo objeto de datos que es diferenciable de otros objetos, ya sean abstractos o concretos.
  - ATRIBUTO: Permite describir a una entidad. Los atributos describen propiedades o características de una entidad
  - RELACIÓN : Describe la conexión o asociación existente entre **dos** o **más** entidades.

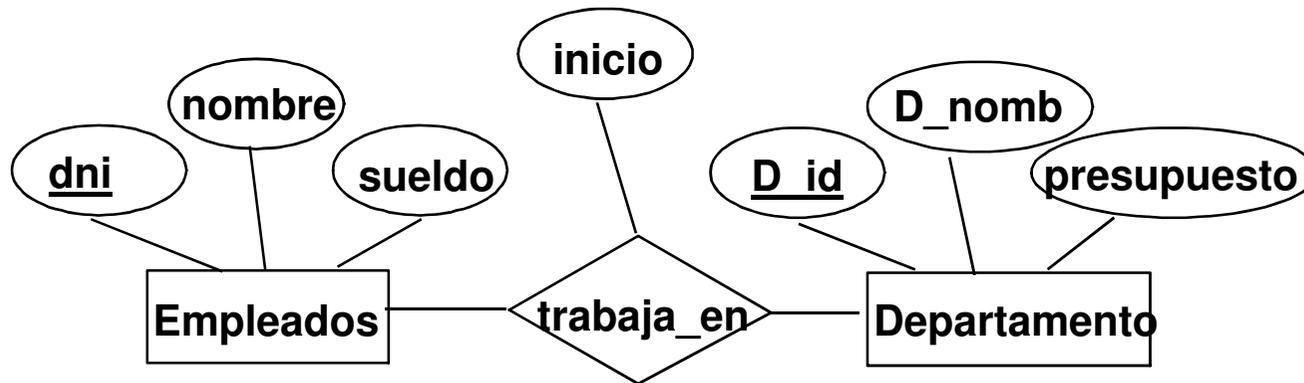
- 
- Cada atributo de una entidad tiene un nombre
  - El conjunto de valores permitidos para cada atributo se denomina **dominio** del atributo
  - Se requiere (normalmente) que los valores de los atributos sean **atómicos**; esto es, indivisibles
    - Los valores de los atributos multivalorados son no atómicos
    - Los valores de los atributos compuestos son no atómicos
  - El valor especial *null* es aceptable pero crea complicaciones en algunas operaciones
-

# Modelo ER

- ENTIDAD
- ATRIBUTO



- RELACIÓN



- Cardinalidad:  $n^{\circ}$  máximo de unidades de un conjunto que se conecta o relaciona con una entidad de otro y viceversa
- Cardinalidad 1:1
  - Una instancia de la entidad **A** está asociada con 0 o 1 instancia de la entidad **B**
  - Una instancia de la entidad **B** está asociada con 0 o 1 instancia de la entidad **A**

- Cardinalidad 1:N
  - Una instancia de la entidad *A* está asociada con 0 o más instancias de la entidad *B*
  - Una instancia de la entidad *B* está asociada con 0 o 1 instancia de la entidad *A*
- Cardinalidad M:N
  - Una instancia de la entidad *A* está asociada con 0 o más instancias de la entidad *B*
  - Una instancia de la entidad *B* está asociada con 0 o más instancias de la entidad *A*

- Participación total
  - Se representa con doble línea del lado de la entidad en la que todas sus instancias deben estar asociadas con alguna instancia de la otra entidad.

- Restricción de estructura = cardinalidad +  
restricción de participación
- **Cardinalidad:** Como se relaciona 1 instancia de una entidad con respecto a las instancias de la otra entidad, Ej., 1:1, 1:N, M:N
- **Participación:** Determina la dependencia de una instancia de una entidad con respecto a las instancias de la otra entidad.
  - Total: Ej. Un empleado debe trabajar en un departamento (doble línea en el diagrama ER) 
  - Parcial: Ej. Un empleado puede ser administrador (línea sencilla en el diagrama ER) 

- Una instancia de una entidad debe distinguirse de otras instancias. Se debe especificar una llave (KEY)
  - **Llave (KEY):**  
conjunto de atributos de una entidad que identifica de manera única una instancia de dicha entidad.
  - **Super llave (Superkey):**  
conjunto de uno o más atributos de una entidad que identifica de manera única una instancia de dicha entidad
  - **Llave candidata (Candidate Key):**  
Super llave para la que ningún subconjunto propio es una superllave (contiene los atributos mínimos para identificar la tupla)

- **Primary Key:**

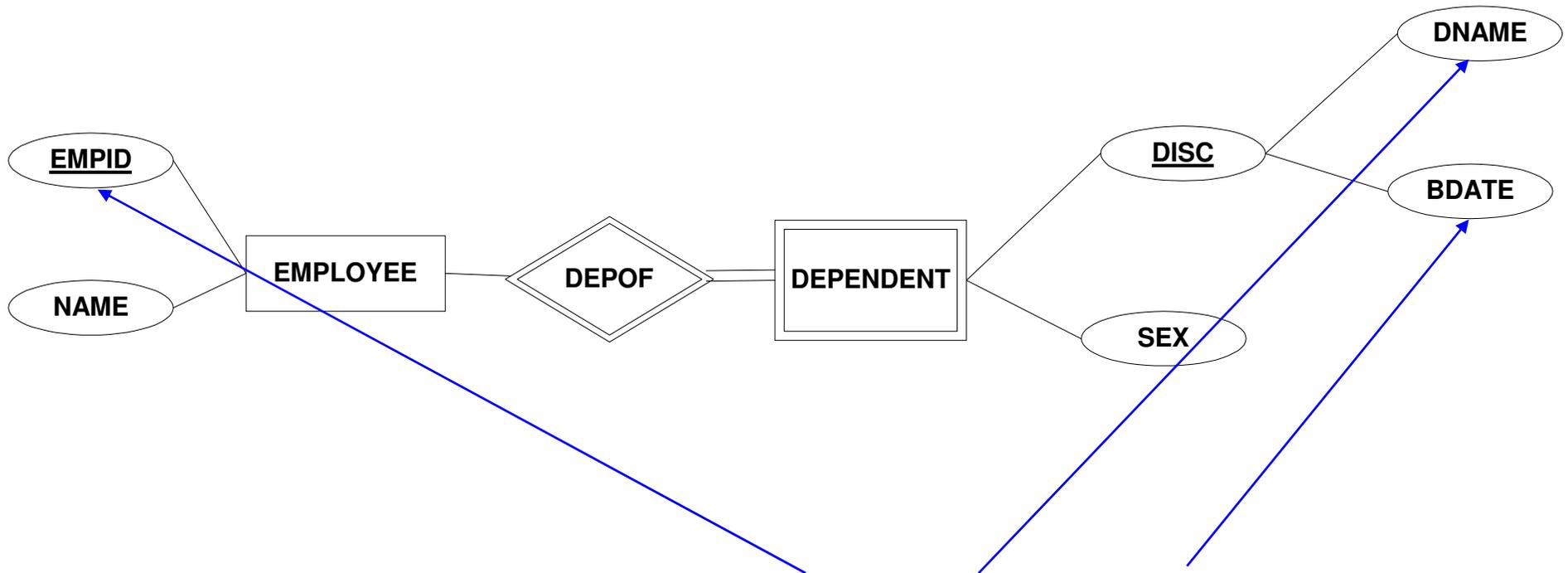
llave seleccionada por el DBA como el principal medio para identificar de manera única las instancias de una entidad

- En el modelo ER se subraya el atributo que es la llave primaria para esa entidad

- **Strong Entity**: entidad que tiene una llave Primaria
- **Weak Entity**: entidad que no tiene suficientes atributos para formar una llave primaria
- La llave primaria (primary key) de una entidad débil se forma con la llave primaria de la entidad fuerte (string) de la cual depende, concatenada con un discriminador (discriminator). El discriminador es el conjunto de atributos que permite distinguir las instancias de la entidad débil.

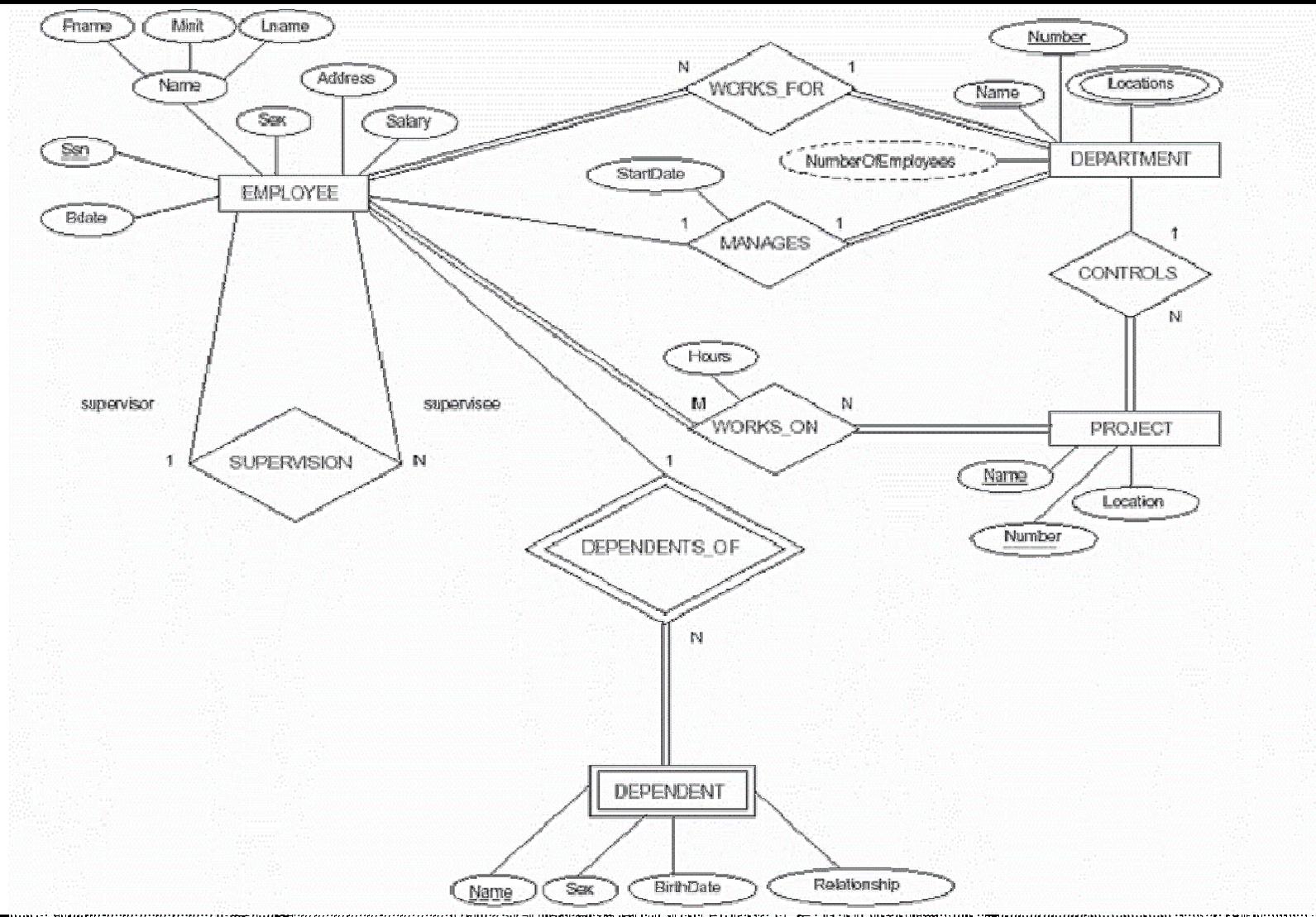
## Weak Entity Example

- Entidades débiles: Una entidad débil solo puede ser identificada de manera única al considerar la llave primaria de otra entidad (dueña)



Dependent's primary key: {empid, dname, birthdate}

# Análisis diagrama ER



- **Empleado**
  - Trabaja exactamente en un dept.
  - Puede o no administrar un solo departamento.
  - Trabaja en al menos 1 proyecto y máximo en n proyectos
  - Puede o no tener dependientes pero máximo n dependientes
  - Puede o no ser supervisor y si supervisa, supervisa a máximo n empleados
  - Puede o no ser supervisado por un solo supervisor.
- **Departamento**
  - Tiene al menos 1 empleado (max N)
  - Tiene exactamente un administrador
  - Puede o no controlar proyectos (max N)
- **Proyecto**
  - Controlado por exactamente 1 departamento
  - Tiene al menos 1 empleado asignado (max N)
- **Dependiente**
  - Depende de exactamente un empleado

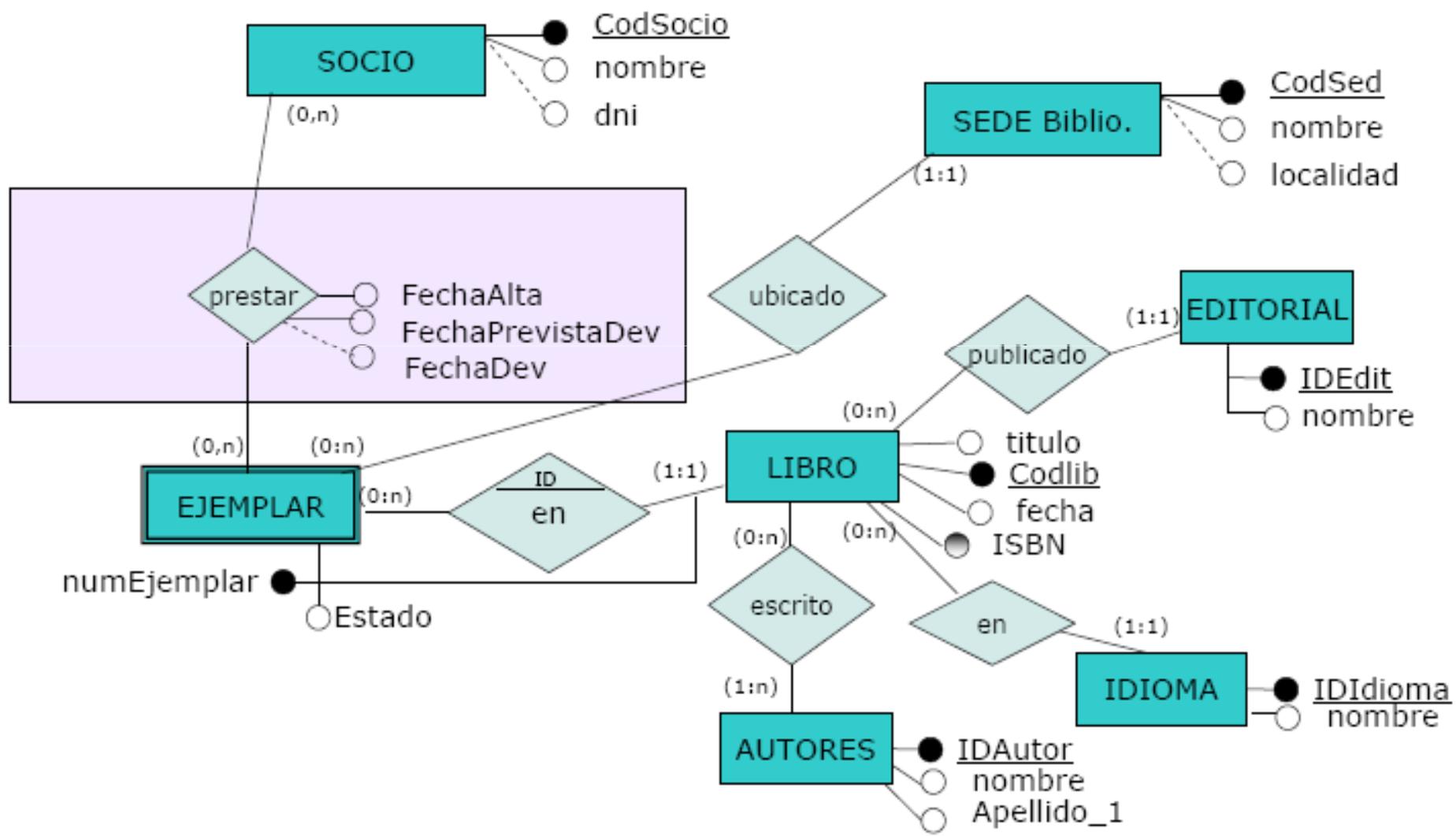
## Ejemplo: Gestión de préstamos

---

- Una biblioteca está interesada en automatizar la gestión de **préstamos** cuyo funcionamiento es:
    - Registrar el **socio** que se lleva el ejemplar del **libro**, fechas de entrega, devolución prevista y de devolución
    - La biblioteca está organizada en diversas **sedes** y el socio puede coger libros de cualquiera de ellas
    - Del socio se tienen los datos personales básicos
    - Y de los libros, todos los campos descriptivos que los caracterizan (título, **idioma**, **autores**, **editorial**, fecha,...).
    - Además de cada ejemplar se querrá conocer el estado en el que se encuentra (prestable, en reparación, fuera de circulación)
-

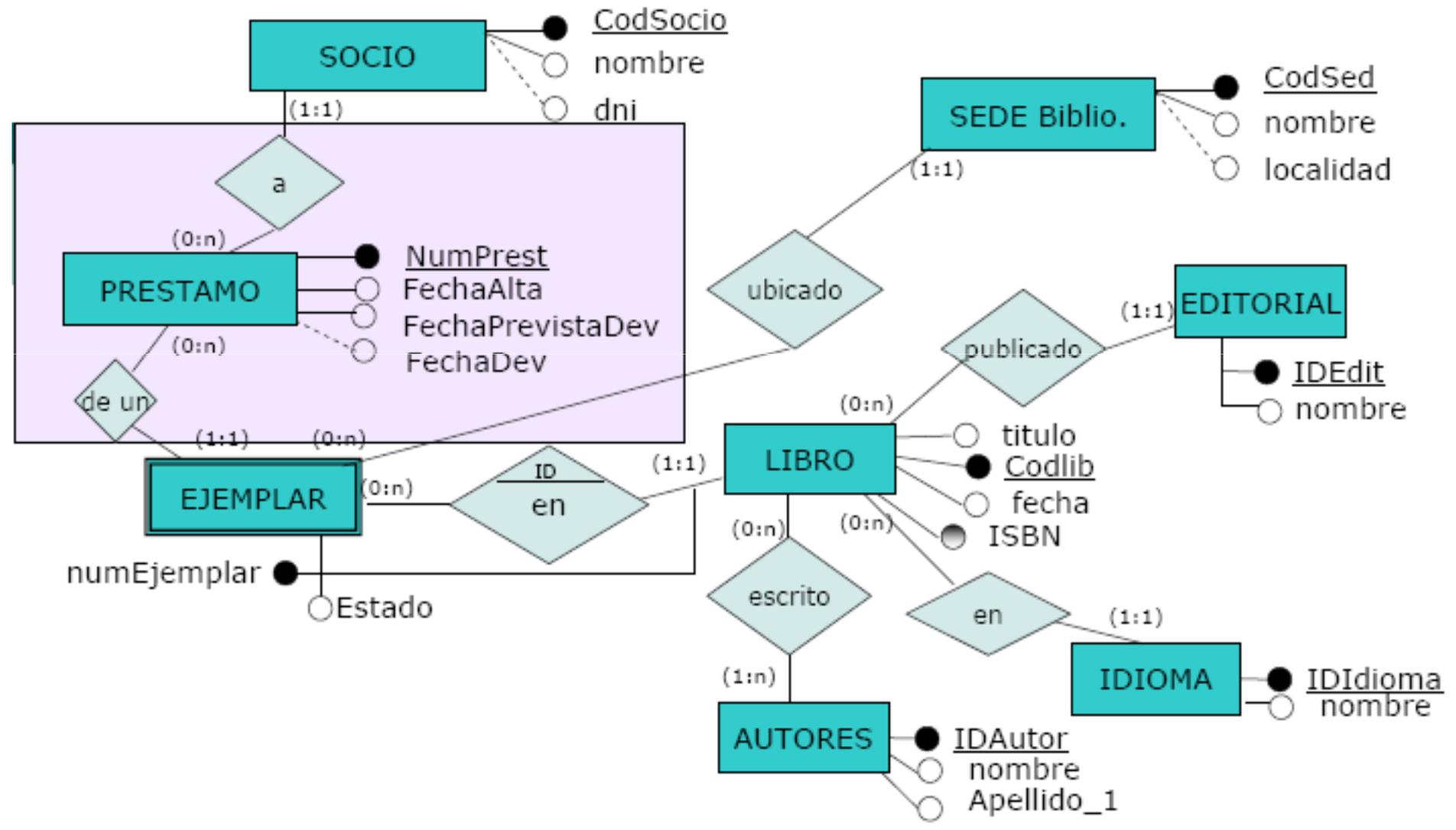
# SQL

## Ejemplo: Gestión de préstamos



# SQL

## Ejemplo: Gestión de préstamos

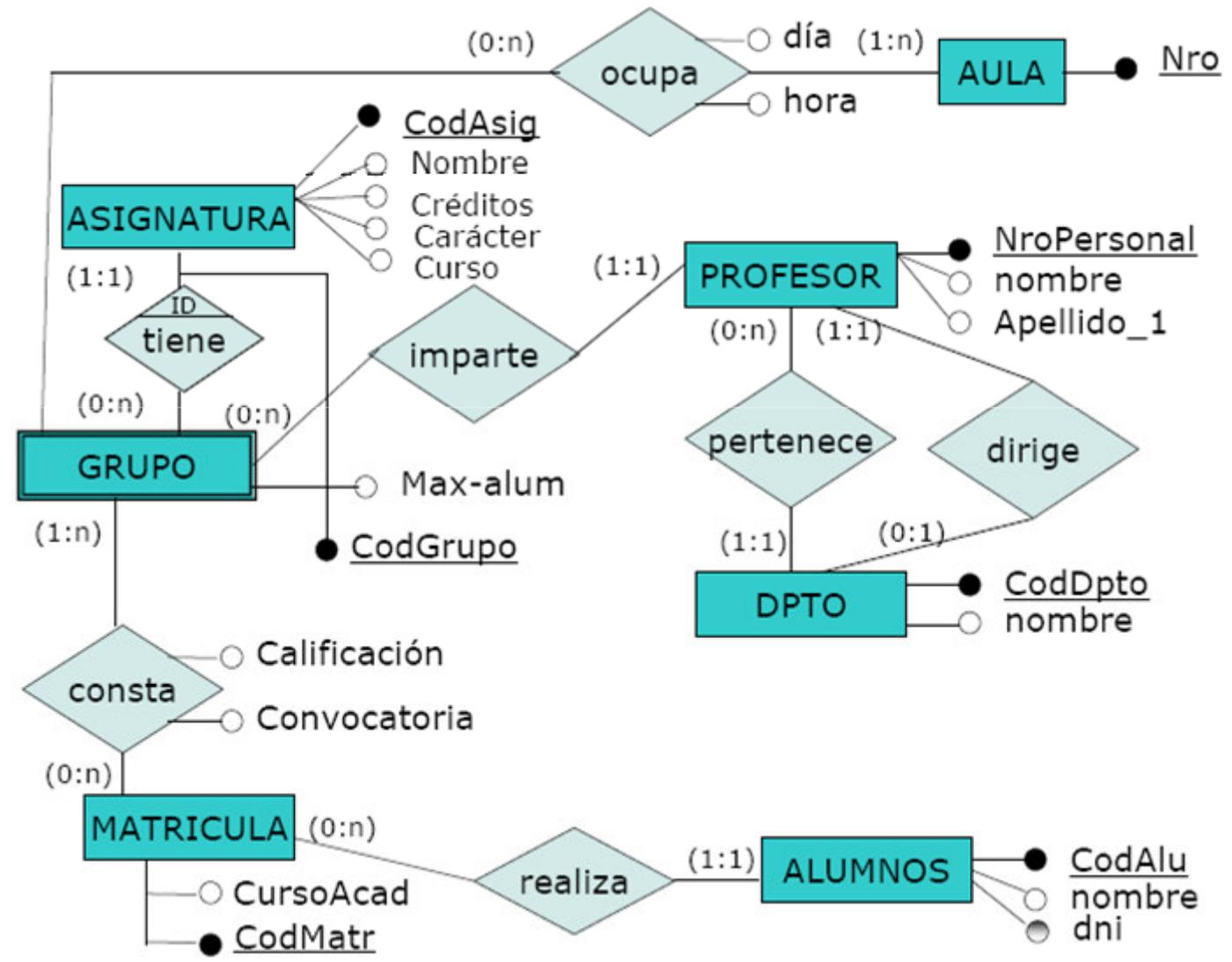


## Ejemplo: Gestión docente

---

- Cada **profesor** pertenece a un sólo **departamento** y debe pertenecer a uno
- El profesor puede impartir varios **grupos** de la misma o distinta **asignatura**, y un grupo debe ser enseñado por un profesor
- Los **alumnos** se matriculan de varias asignaturas (al menos una) cada curso académico pero han de hacerlo en un grupo. A su vez un grupo tendrá varios alumnos matriculados. Cada grupo tendrá asignado un **aula** para cada día y hora de la semana.
- La **matrícula** dará opción a dos convocatorias de examen con su respectiva calificación.
- Todo departamento debe tener un director, que es profesor
- Los atributos de cada entidad son los que cabría esperar

# Ejemplo: Gestión docente



## Ejemplo: Gestión de compras

---

- Una empresa está interesada en automatizar su proceso de compras cuyo flujo de funcionamiento es:
  - Requiere registrar la hoja del pedido que realiza a un determinado proveedor en una determinada fecha
  - En la hoja del pedido queda constancia del número de unidades que compra de cada artículo y el precio de compra, y en caso de que el proveedor o bien por volumen o por promoción, le realiza un descuento, también lo anota
  - Los productos que compran tienen distinto IVA
  - Generalmente el paga a sus proveedores al mes de recibir la mercancía y por transferencia, aunque lo puede hacer a plazos
  - Los atributos de cada entidad son los que cabría esperar

# Ejemplo: Gestión de compras

