



# DISEÑO CONCEPTUAL DE BASES DE DATOS



[www.sena.edu.co](http://www.sena.edu.co)





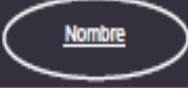
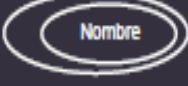

**CONSTRUCCIÓN** **Paso a Paso**  
**MER**  
**Modelo Entidad Relación**

# Concepto

El modelo Entidad-Relación es el modelo más utilizado para el diseño conceptual de bases de datos. Fue introducido por Peter Chen en 1976 y se basa en la existencia de objetos a los que se les da el nombre de entidades, y asociaciones entre ellos, llamadas relaciones.

Inicialmente (en la propuesta de Chen) sólo se incluían los conceptos de entidad, relación y atributos. Después se añadieron otras propuestas (atributos compuestos, generalizaciones,...) que forman el llamado **modelo entidad relación extendido** (se conoce con las siglas **ERE**)

# Símbolos del Modelo ER

ELEMENTO	REPRESENTACIÓN
Entidad o entidad fuerte	
Entidad debil	
Relación	
Atributo (mono valor)	
Atributo llave	
Atributo multi valorado	
Atributo compuesto	

Es el proceso que permite describir el tipo y las relaciones entre los datos. Que permite representar las entidades relevantes de una base de datos así como sus interrelaciones y propiedades. La representación de este se realiza a través de un diagrama con una simbología definida.

**Entidad:** es la representación de un objeto

**Atributo:** Característica o rasgo de una entidad

**Relación:** entre conjunto de entidades

**Conector**

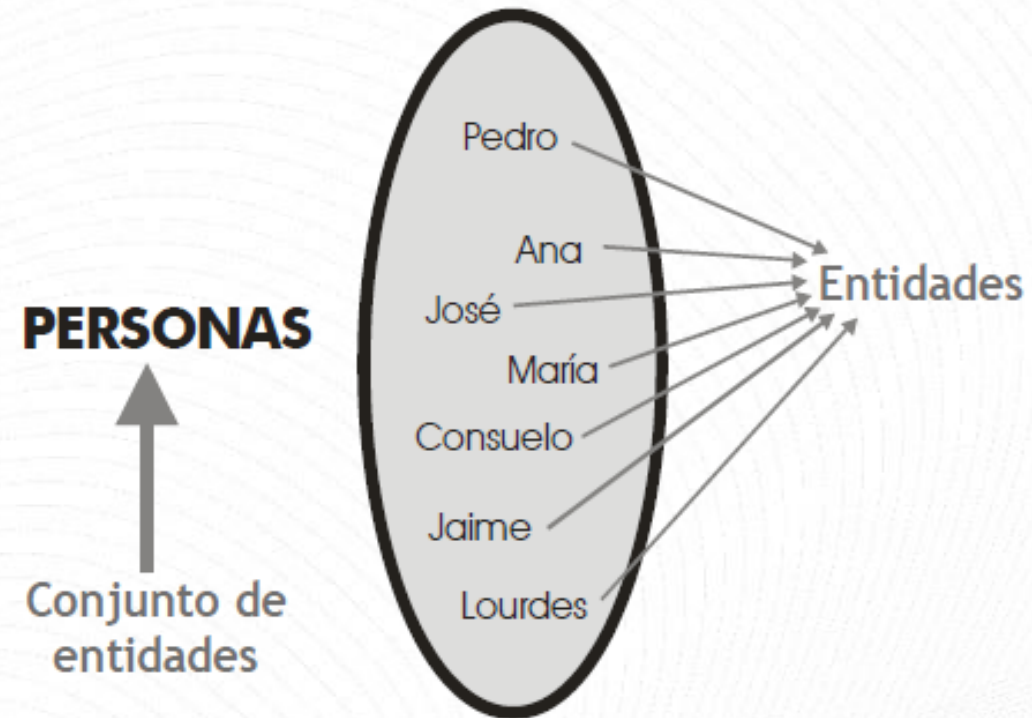
# Entidad



Se trata de cualquier objeto u elemento (real, (concreto) o abstracto) acerca del cual se pueda almacenar información en la base de datos. Ejemplos de entidades son Pedro, la factura, vehículo, etc.

Una entidad no es una propiedad concreta sino un objeto que puede poseer múltiples propiedades (atributos).

Las entidades que poseen las mismas propiedades forman conjuntos de entidades. Ej. Estudiante forma el grupo o conjunto Estudiantes





# Tipo de Entidades



**Fuertes, Dominantes.** Son las entidades normales que tienen existencia por sí mismas sin depender de otras. Tienen una clave primaria que las identifica de manera única

**PERSONAS**

**Débiles.** Su existencia depende de otras. Por ejemplo la entidad **tarea laboral** sólo podrá tener existencia si existe la entidad **trabajo**. Las entidades débiles se presentan de esta forma:

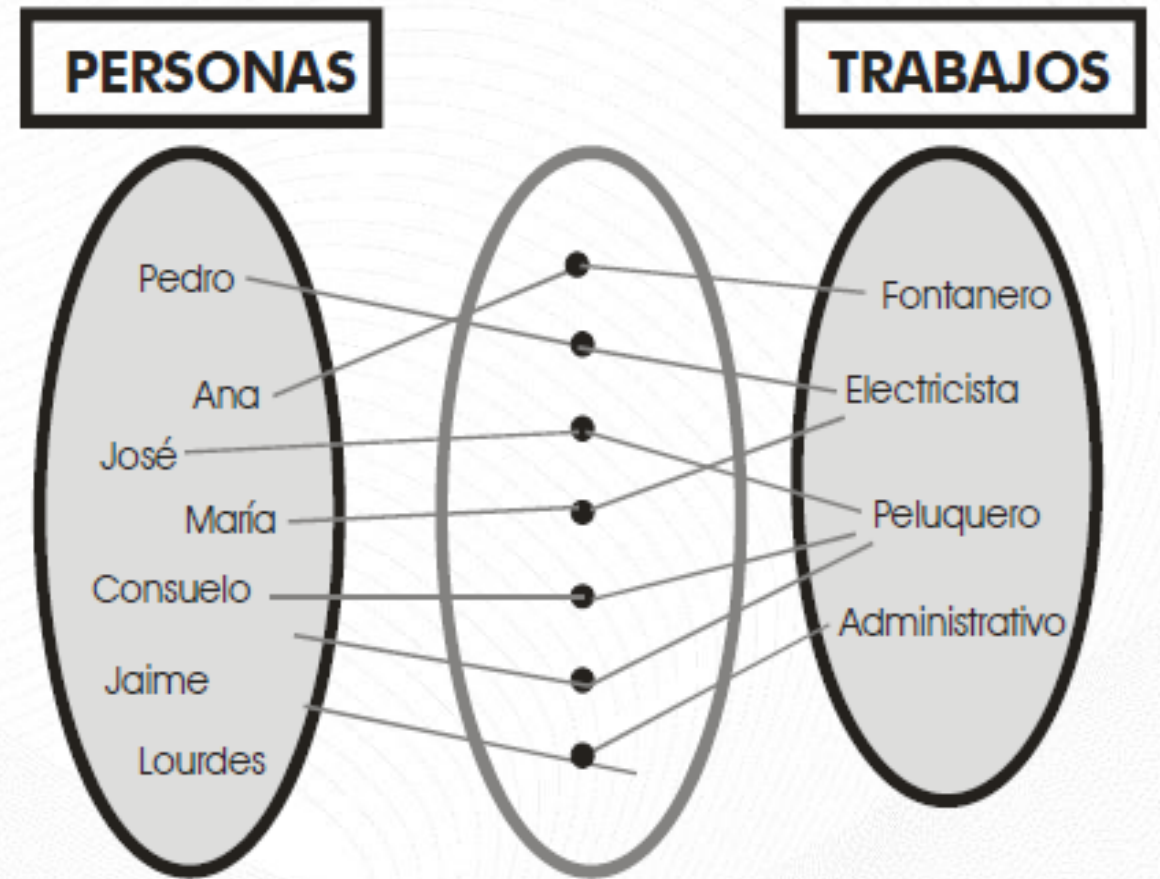
**TAREAS LABORALES**

NOTA: Cuando la Relación es entre una Entidad Fuerte con una Débil la relación siempre será (1:N)

# Relaciones



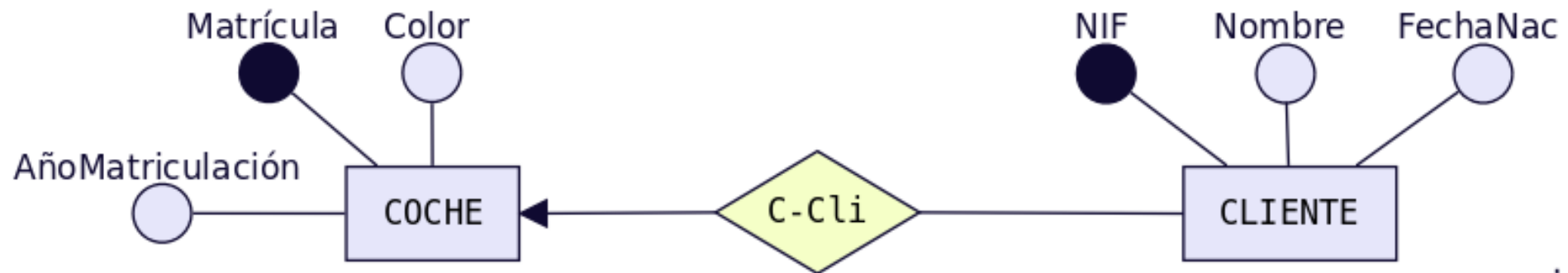
Representan asociaciones entre entidades. Es el elemento del modelo que permite relacionar en sí los datos del modelo. Por ejemplo, en el caso de que tengamos una entidad personas y otra entidad trabajos. Ambas se realizan ya que las personas trabajan y los trabajos son realizados por personas:



# Relaciones



**Una relación es la asociación que existe entre dos a más entidades.** Las relaciones se representan gráficamente mediante rombos y su nombre aparece en el interior.





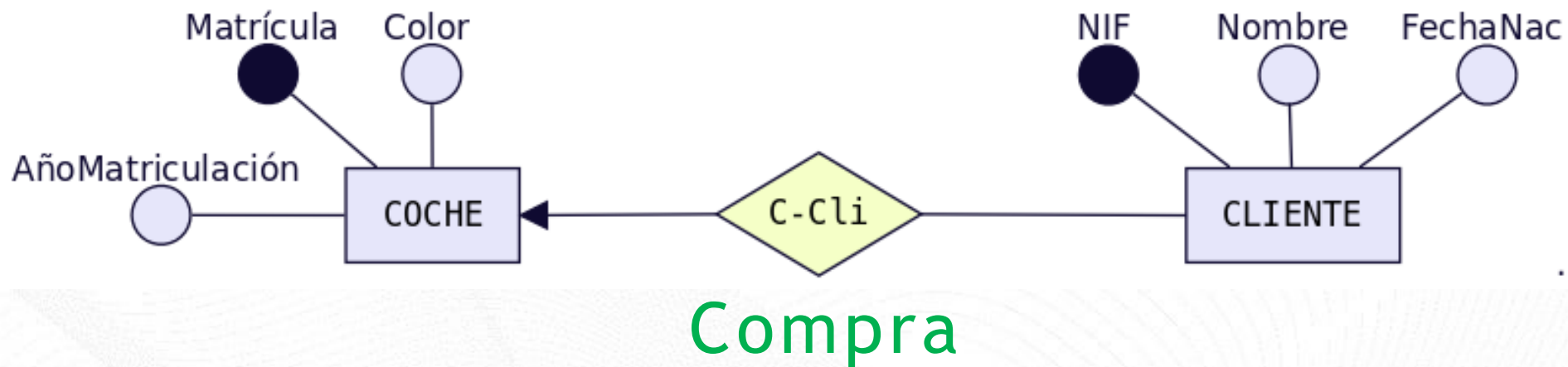
# TIPOS DE RELACION

# Tipos o Grado de Relaciones



El grado hace referencia al número de entidades relacionadas. Los tipos de grado pueden ser:

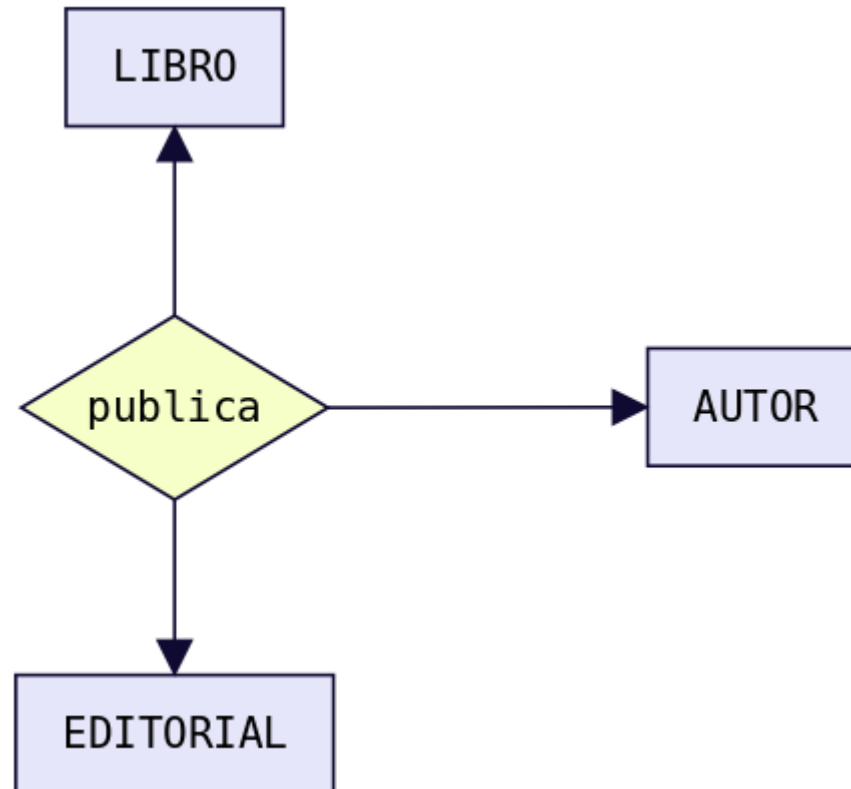
## Binaria o Grado 2



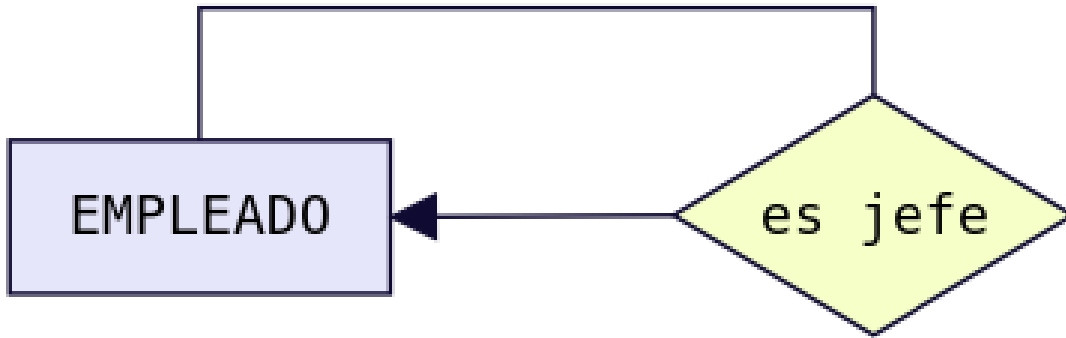
# Ternaria - mayor a 2



Relaciones de grado superior a 2 (ternarias, cuaternarias, etc.): aquellas que asocian más de dos entidades en la relación.



# Reflexiva - Grado 1



Relaciones de grado 1 (reflexivas): en la que se relaciona una entidad consigo misma. Ej. en una entidad "Empleado", podrías tener una relación reflexiva llamada "supervisa", donde un empleado supervisa a otro empleado. .

Aunque el modelo E-R permite relaciones de cualquier grado, la mayoría de las aplicaciones del modelo sólo consideran relaciones del grado 2



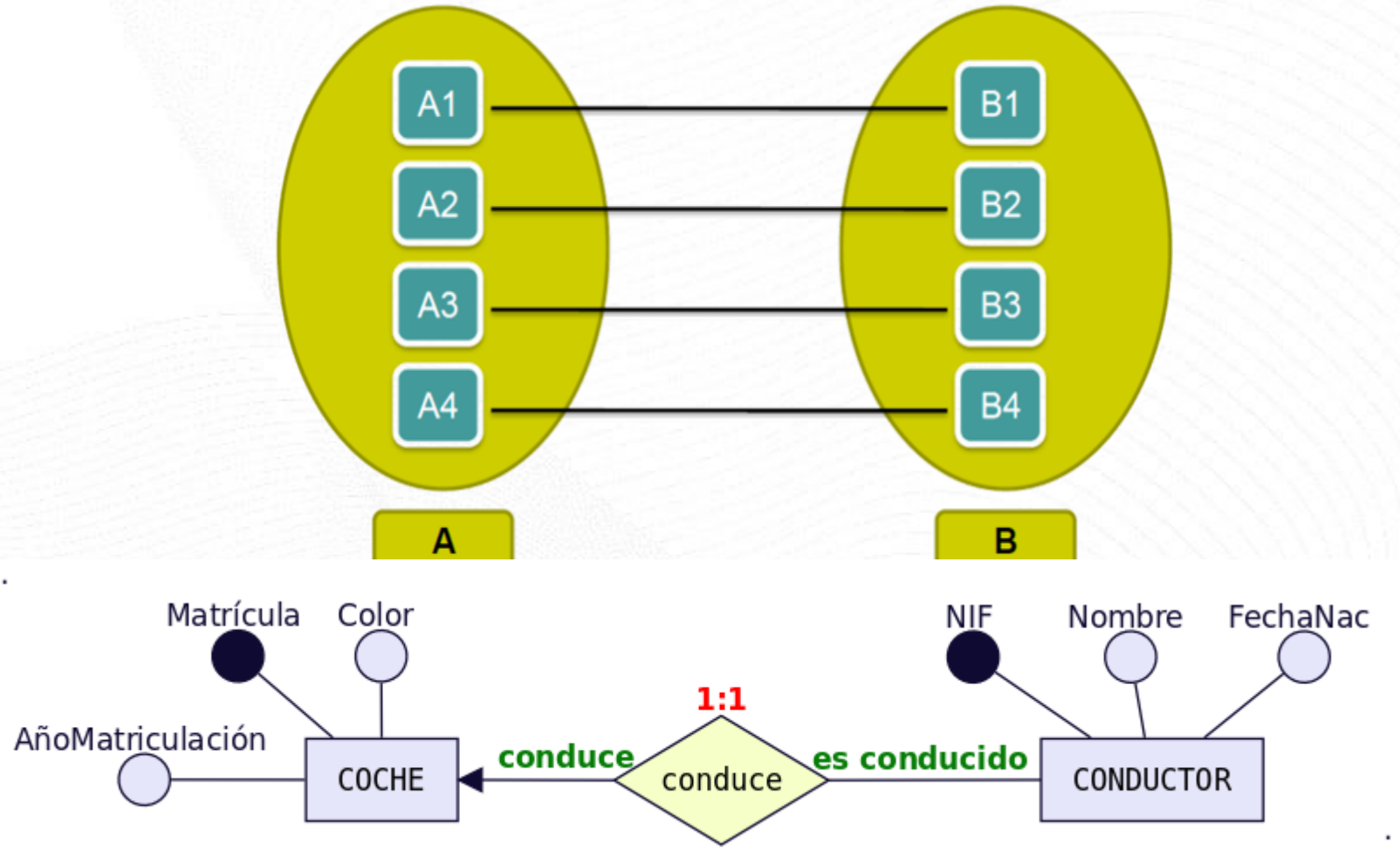
# CARDINALIDAD

---

Cuando la relación es binaria, cosa que ocurre en la mayoría de los casos, la cardinalidad es el número de ocurrencias de una entidad asociadas a una ocurrencia de la otra entidad. Existen 3 casos:

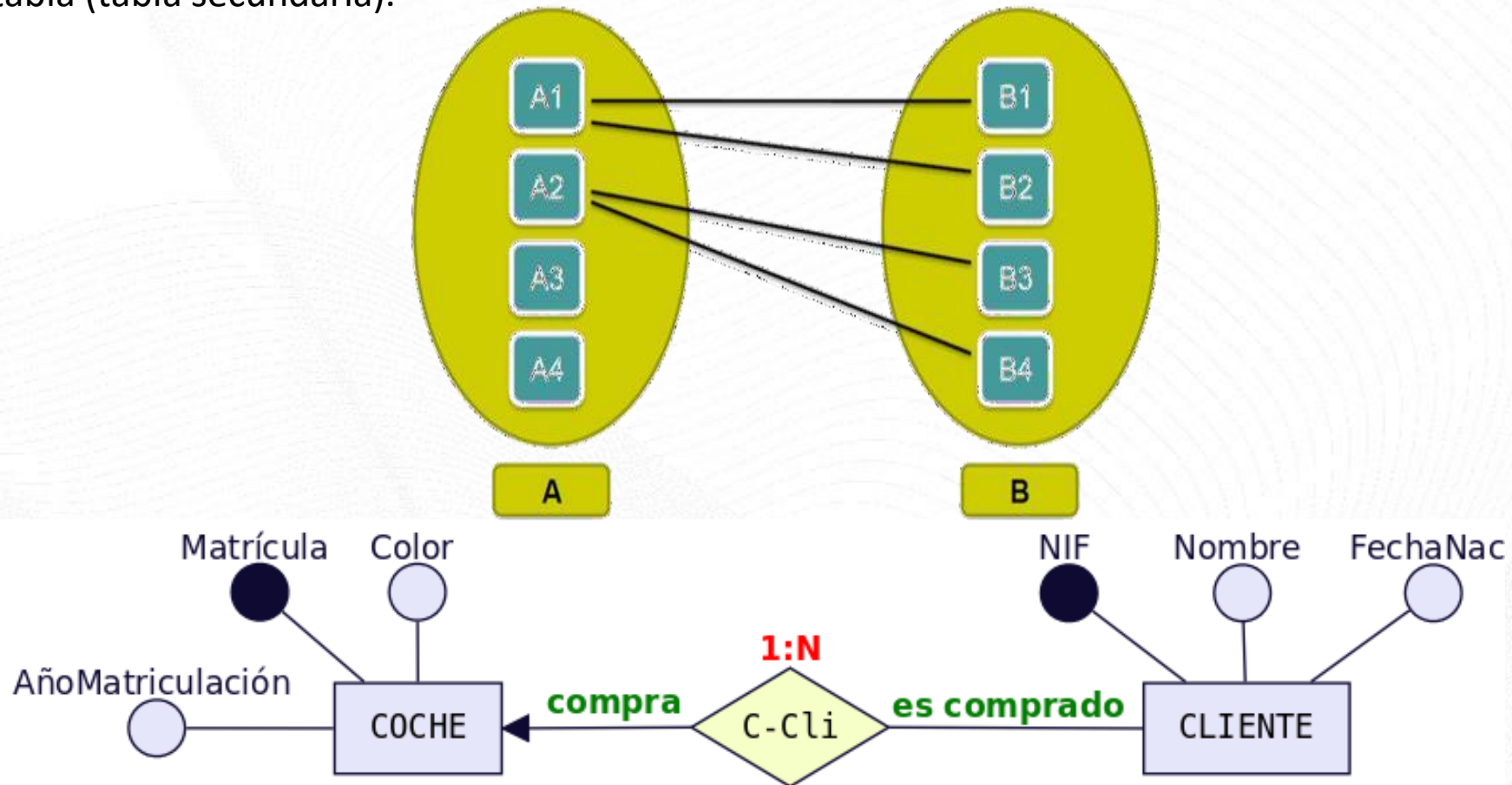
# Relación Uno a Uno (1:1)

Cuando un registro de una tabla sólo puede estar relacionado con un único registro de la otra tabla y viceversa.



# Relación uno a Muchos (1:N)

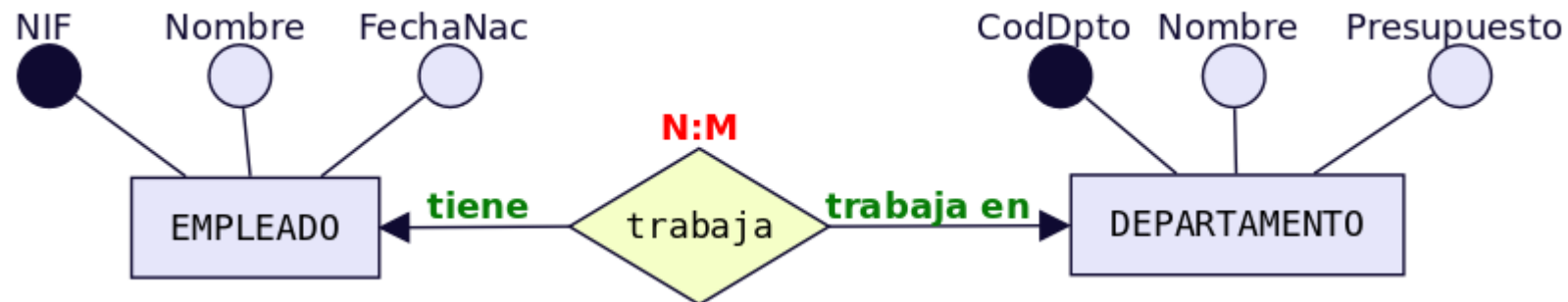
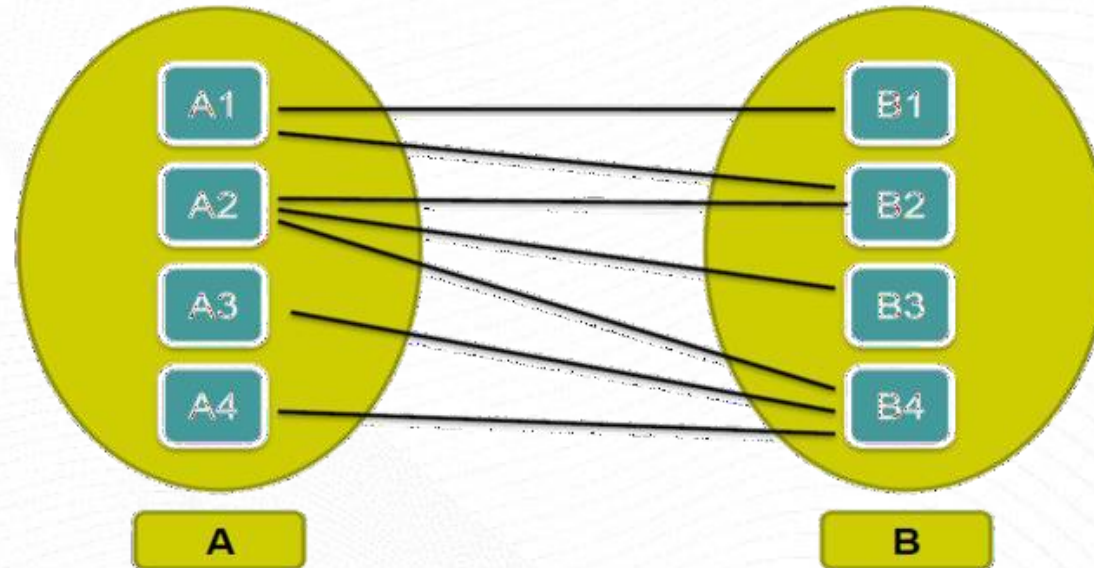
Cuando un registro de una tabla (tabla secundaria) sólo puede estar relacionado con un único registro de la otra tabla (tabla principal) y un registro de la otra tabla (tabla principal) puede tener más de un registro relacionado en la primera tabla (tabla secundaria).



# Relación Muchos a Muchos (N:N)

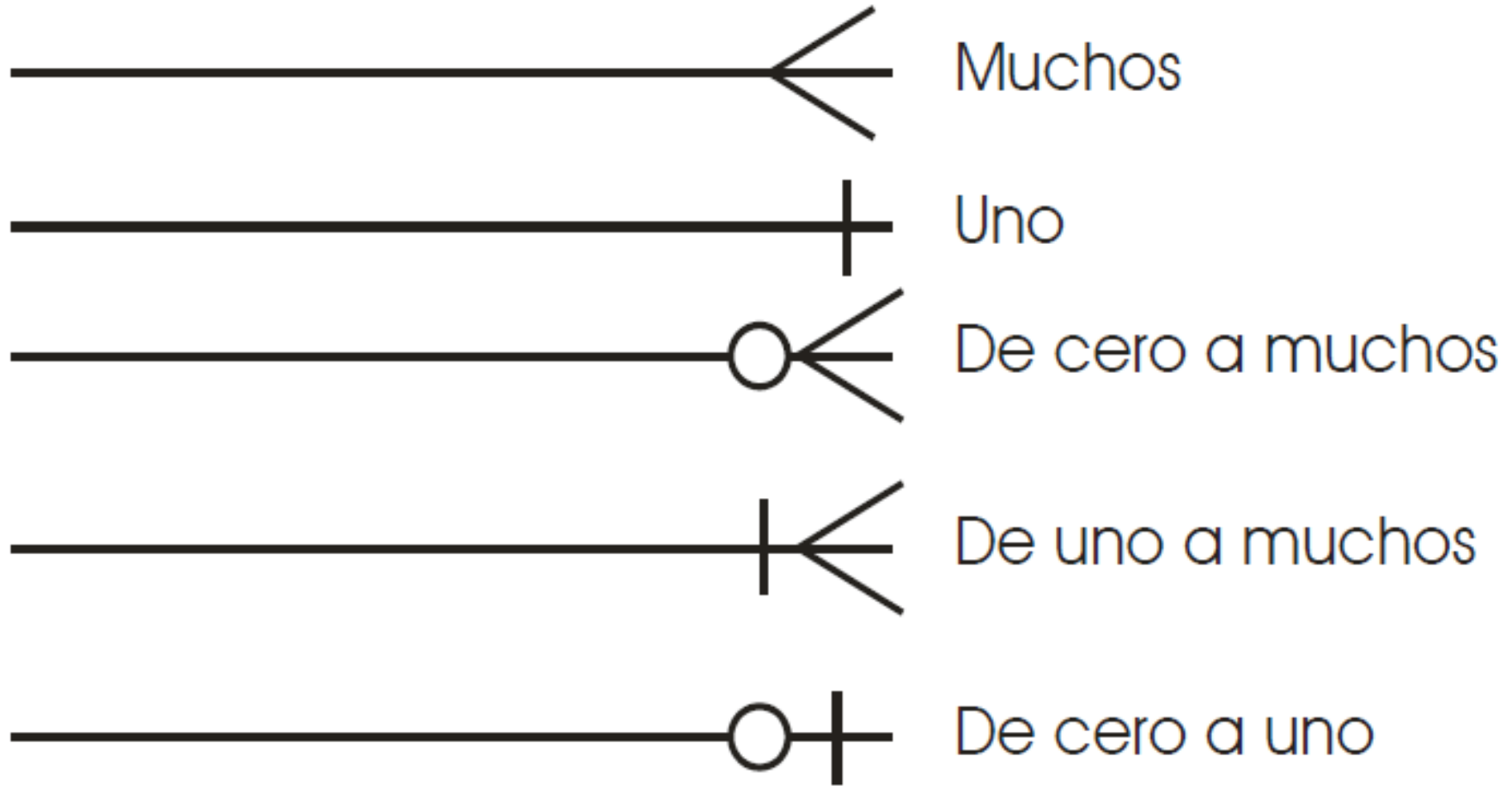


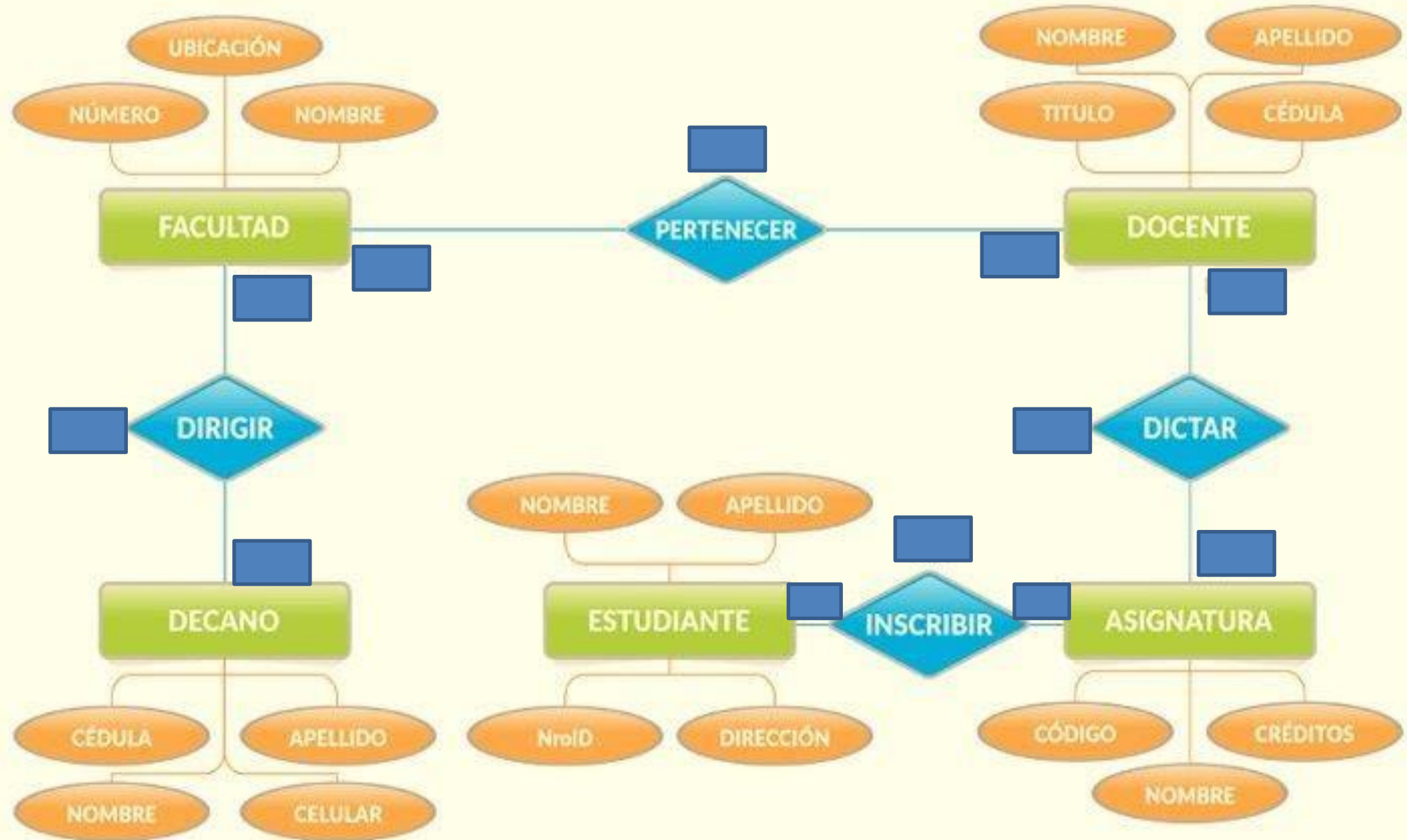
Cuando un registro de una tabla puede estar relacionado con más de un registro de la otra tabla y viceversa.





# Tipos de Relación







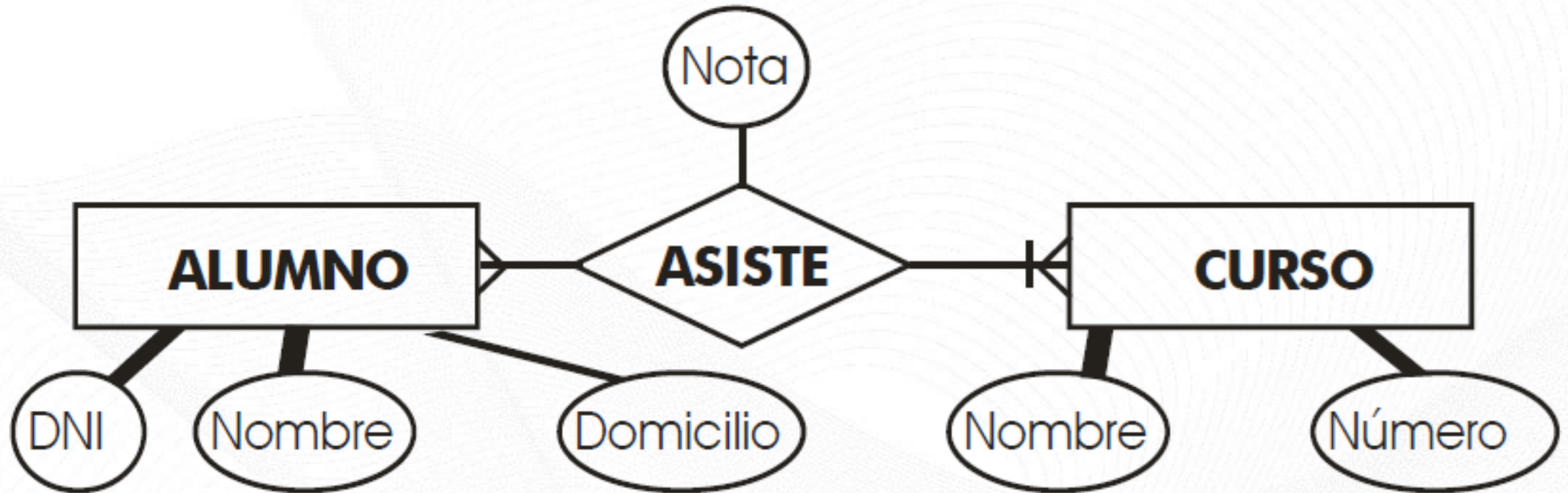
# ATRIBUTOS

---

# Concepto



Describen propiedades de las entidades y las relaciones. Los atributos describen propiedades que posee cada miembro de un conjunto de entidades. En este modelo se representan con un círculo, dentro del cual se coloca el nombre del atributo. Ejemplo:





# Dominios y Valores



Podemos definir el dominio de un atributo como la especificación de todos los valores que pueden estar contenidos en dicho atributo. Se trata pues de una restricción de cuáles son los valores aceptables para ese atributo concreto.

Algunos tipos de valores son:

**Numérico:** Puede ser entero, decimal, real, etc.

**Cadena de caracteres o texto:** Incluye cualquier secuencia de caracteres, como nombres o direcciones.

**Fecha/Hora:** Registra fechas y/u horas.

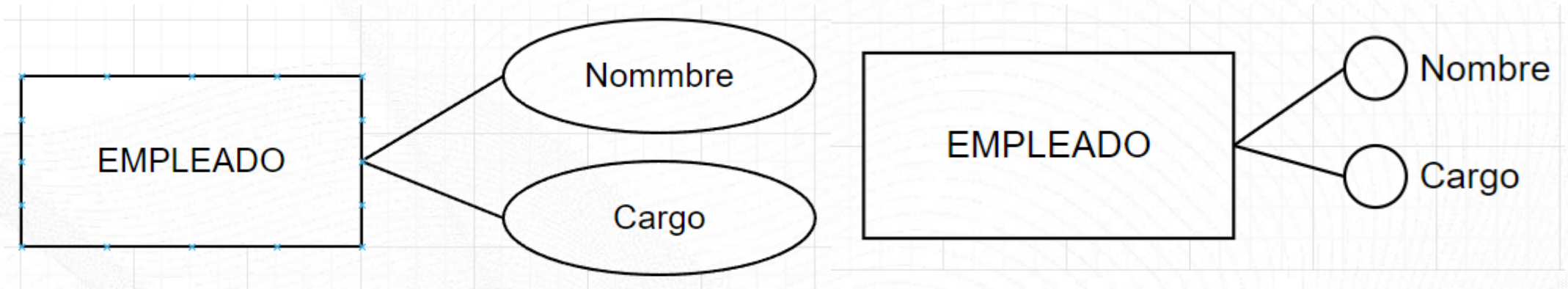
**Booleano:** Valores de verdad, generalmente verdadero o falso.

**Blob (Binary Large Object):** Almacena datos binarios como imágenes o contenido multimedia.

# Tipos de Atributos



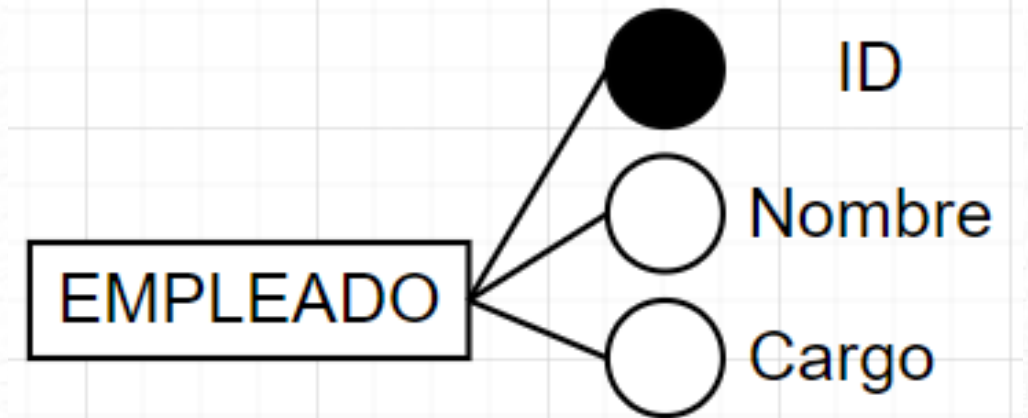
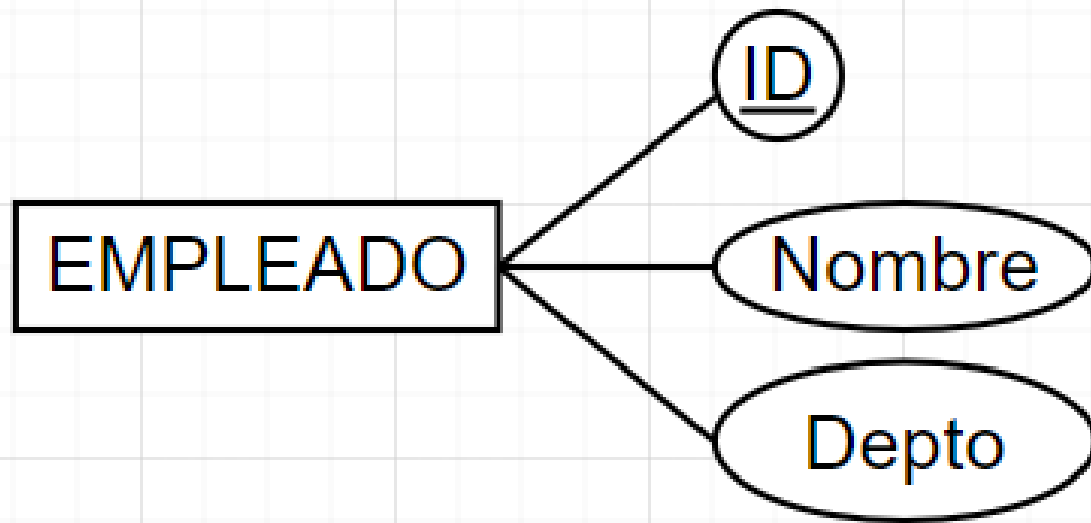
**Atributo simple o Descriptivo:** un atributo simple es un atributo que describe una característica única de una entidad. Por ejemplo, el nombre de una persona, la fecha de nacimiento, etc. Y se representa mediante una elipse



# Tipos de Atributos



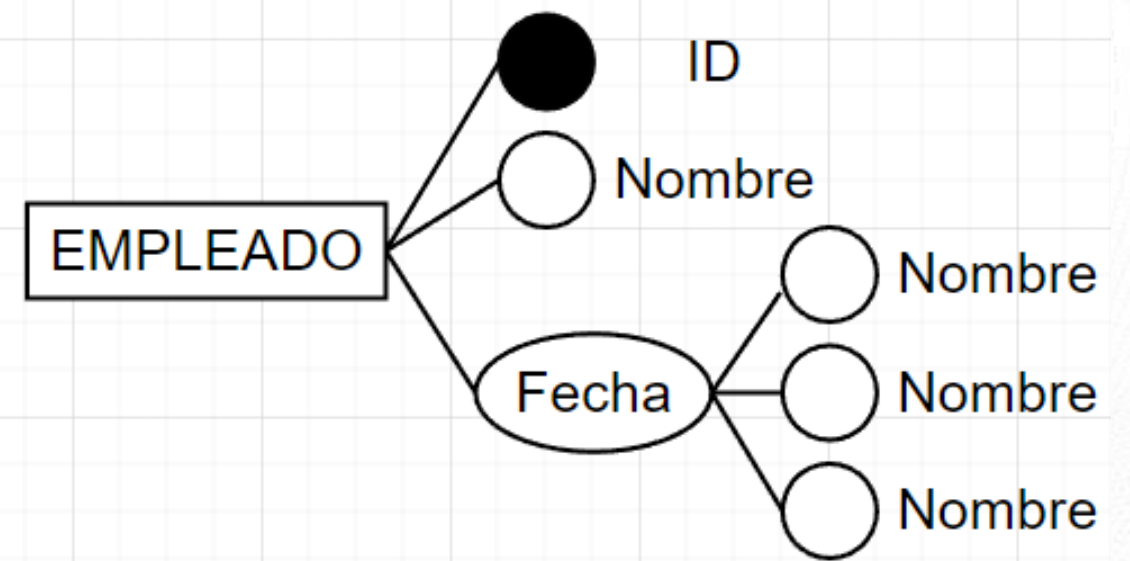
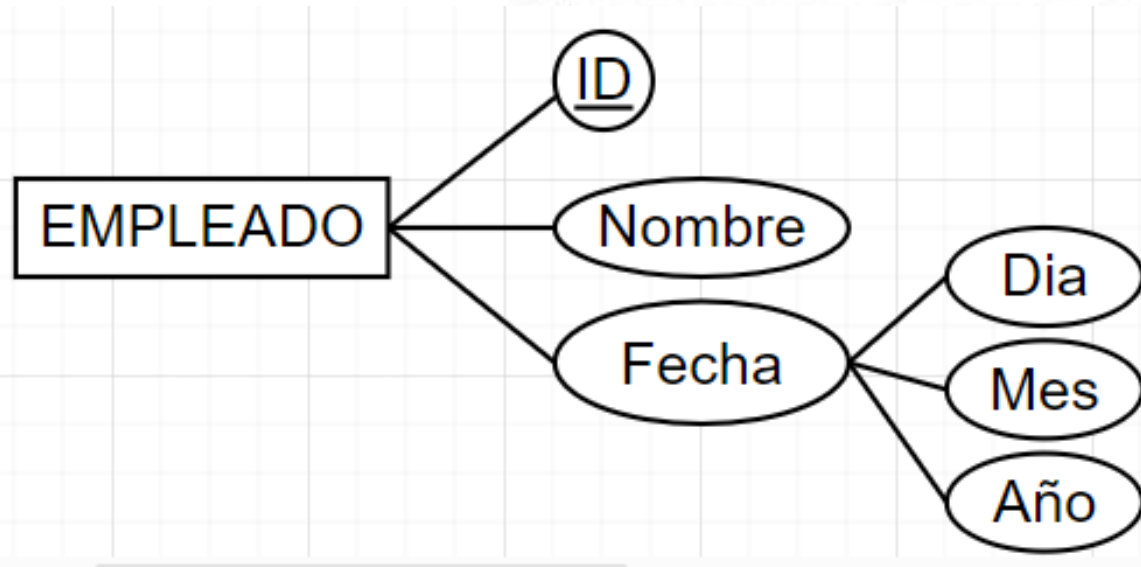
**Atributo clave:** Es un atributo que se utiliza para identificar únicamente a una entidad. Por ejemplo, el número de identificación de un estudiante, que lo identifica únicamente entre todos los estudiantes de la escuela.



# Tipos de Atributos



**Atributo Compuesto:** Es un atributo que se utiliza para identificar únicamente a una entidad. Por ejemplo, el número de identificación de un estudiante, que lo identifica únicamente entre todos los estudiantes de la escuela.

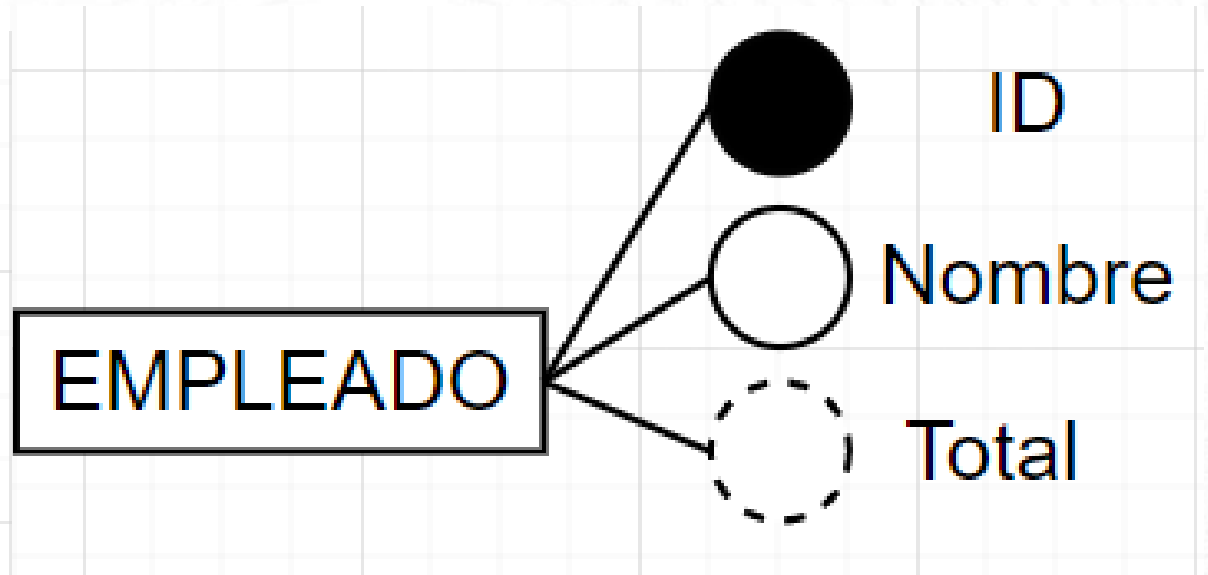
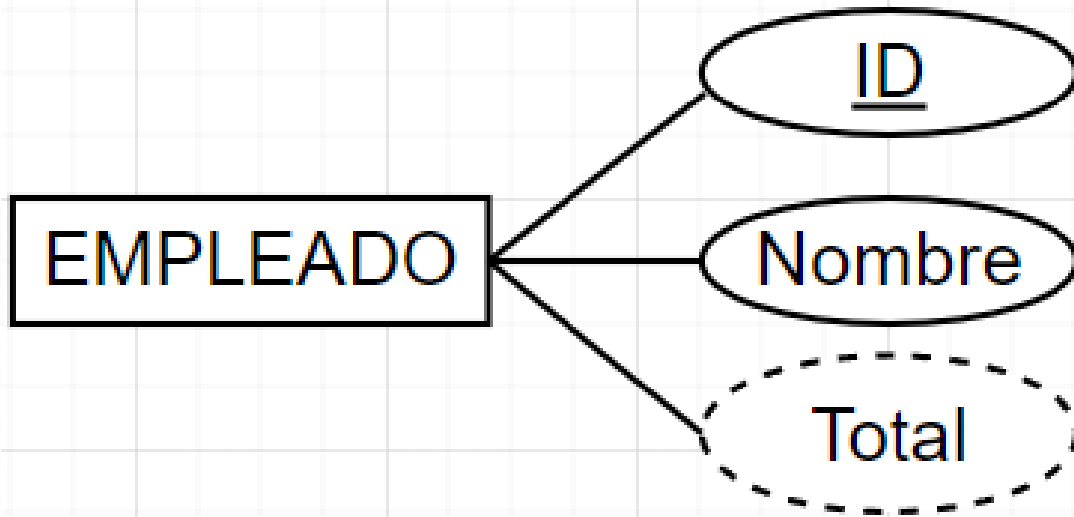




# Tipos de Atributos



**Derivado:** Es aquel que se genera a partir de otros atributos u otras entidades



# Generalización / Especialización

Se trata de un mecanismo muy útil de abstracción, que permite definir relaciones y asignar atributos de subclases entre objetos.

Notación y otros aspectos a tener en cuenta:

- La cardinalidad es siempre 1:1 en la superclase y 0:1 o 1:1 en las subclases.
- Las subclases heredan los atributos de la superclase.
- Se denota con un triángulo con la base paralela a la superclase (en el triángulo se puede incluir un atributo, llamado atributo selector).
- Se pueden añadir dos propiedades a la generalización/especialización:
  - Exclusión/Solapamiento: denotado por un arco.
  - Totalidad/Parcialidad: denotado por un círculo en el arco.

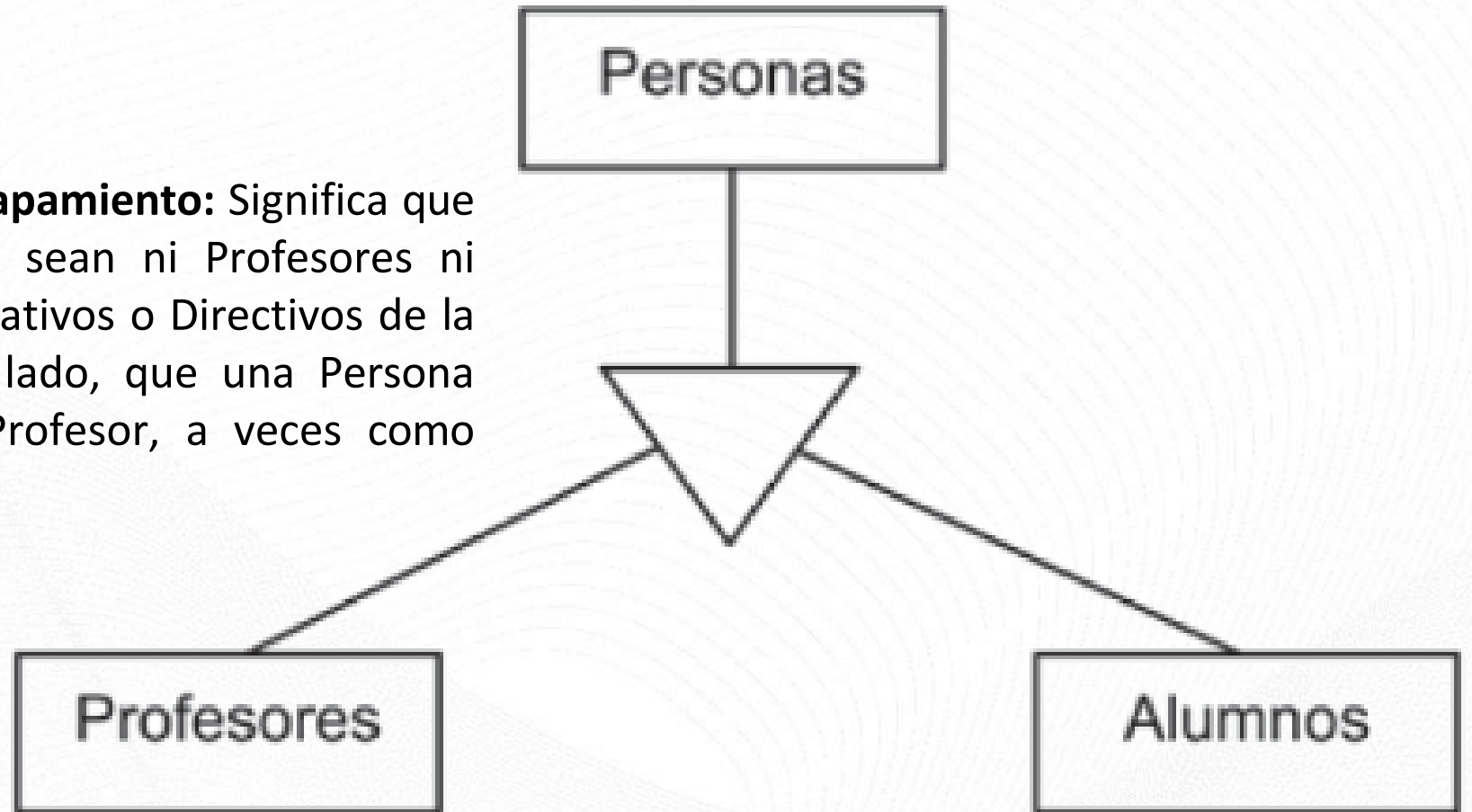
# Generalización / Especialización

- Generalización: proceso de abstracción que lleva a la definición de un nuevo tipo de entidad (superclase) a partir de elementos “similares” de otras (subclases).
- Especialización: proceso de reclasificación que permite obtener nuevas entidades (subclase) a partir de las “diferencias” entre los elementos de una clase superior (superclase).

# Generalización parcial y con solapamiento



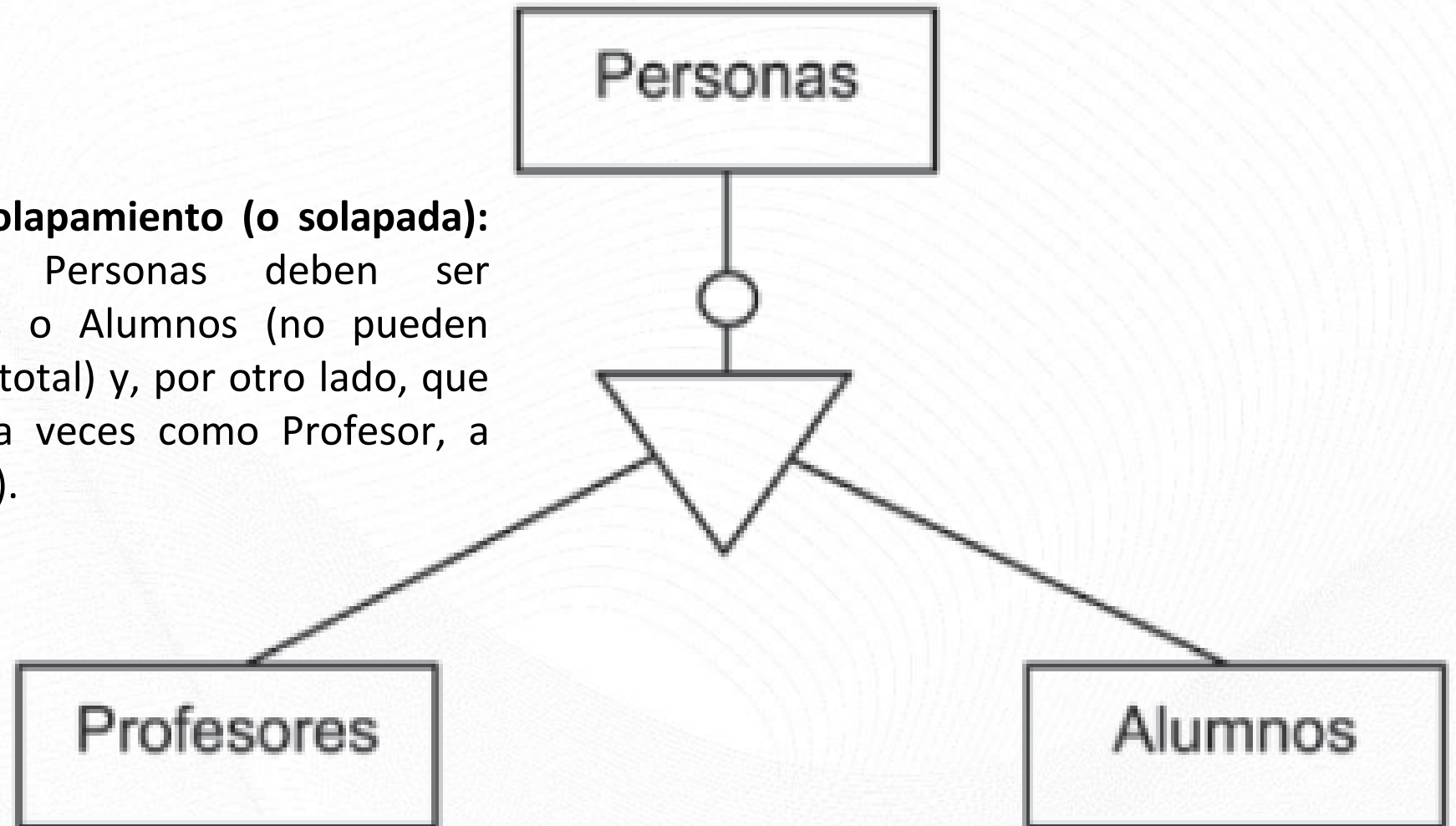
**Generalización parcial y con solapamiento:** Significa que puede haber Personas que no sean ni Profesores ni Alumnos, por ejemplo Administrativos o Directivos de la academia (parcial) y, por otro lado, que una Persona puede actuar, a veces como Profesor, a veces como Alumno (solapamiento)





# Generalización total y con solapamiento (o solapada)

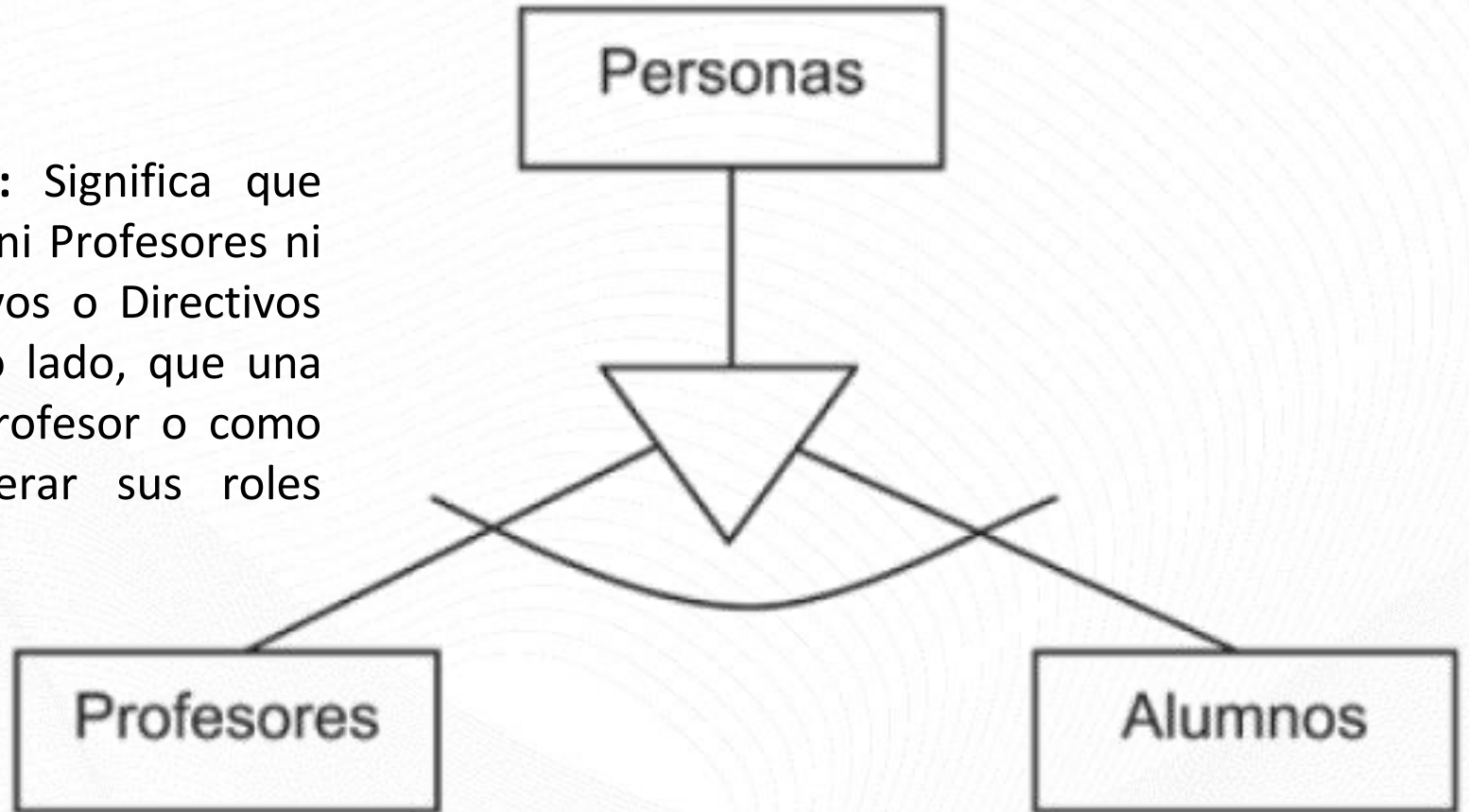
**Generalización total y con solapamiento (o solapada):** Significa que todas las Personas deben ser exclusivamente o Profesores o Alumnos (no pueden pertenecer a otros sectores) (total) y, por otro lado, que una Persona puede actuar, a veces como Profesor, a veces como Alumno (solapada).



# Generalización parcial y exclusiva



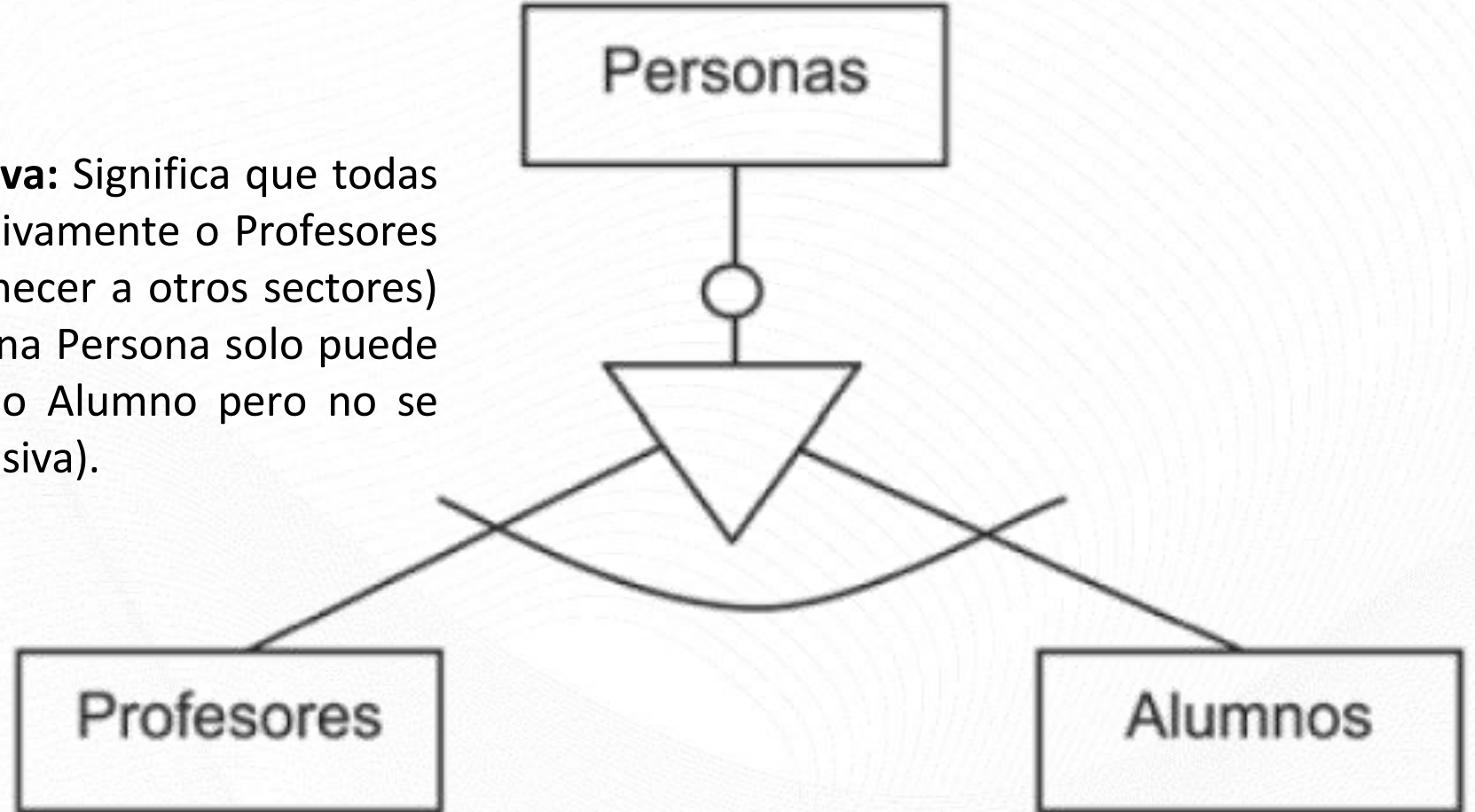
**Generalización parcial y exclusiva:** Significa que puede haber Personas que no sean ni Profesores ni Alumnos, por ejemplo Administrativos o Directivos de la academia (parcial) y, por otro lado, que una Persona solo puede actuar como Profesor o como Alumno pero no se pueden alterar sus roles (exclusiva).



# Generalización total y exclusiva



**Generalización total y exclusiva:** Significa que todas las Personas deben ser exclusivamente o Profesores o Alumnos (no pueden pertenecer a otros sectores) (total) y, por otro lado, que una Persona solo puede actuar como Profesor o como Alumno pero no se pueden alterar sus roles (exclusiva).





# Practica



Logotipo de la empresa		EMPRESA XYZ S.A. ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE No 001	Formato XYZ-GC-001
ID	Nombre	Descripción	Prioridad
1	Módulo de servicio al cliente para una empresa de servicios públicos.	<p>Se requiere un módulo para atender las peticiones, quejas y recursos (PQR's) que interponen los clientes en nuestras oficinas, telefónicamente o por correo electrónico.</p> <p>El sistema deberá permitir la consulta, registro, actualización e impresión de cada PQR. El formulario de PQR deberá contener:</p> <ol style="list-style-type: none"><li>1. Tipo: petición, queja, recurso, otros servicios.</li><li>2. Fecha y hora: deberá traer la fecha y hora del sistema al momento de grabar el servicio.</li><li>3. Cédula o identificación del cliente.</li><li>4. Descripción de la PQR.</li><li>5. Origen de la PQR: oficina, telefónico, correo electrónico, carta.</li><li>6. Teléfono del cliente.</li><li>7. Estado: solicitada, resuelta, anulada.</li></ol> <p>El sistema deberá proveer un programa para que las PQR's no solucionados sean asignados al área que corresponda. Ejemplo los PQR's por exceso de consumo deberán ser asignados al área técnica.</p> <p>El sistema deberá proveer una relación con todos los PQR's permitiendo filtrar por fecha, estado, cédula, nombre y origen. Esta relación puede ser vista por pantalla, enviada a impresora, o enviada a una hoja electrónica.</p>	Alta

Se requiere un módulo para atender las peticiones, quejas y recursos (PQR's) que interponen los clientes en nuestras oficinas, telefónicamente o por correo electrónico.

El sistema deberá permitir la consulta, registro, actualización e impresión de cada PQR.

El formulario de PQR deberá contener:

1. Tipo: petición, queja, recurso, otros servicios.
2. Fecha y hora: deberá traer la fecha y hora del sistema al momento de grabar el servicio.

3. Cédula o identificación del cliente.

4. Descripción de la PQR.

5. Origen de la PQR: oficina, telefónico, correo electrónico, carta.

6. Teléfono del cliente.

7. Estado: solicitada, resuelta, anulada. El sistema deberá proveer un programa para que las PQR's no solucionados sean asignados al área que corresponda. Ejemplo los PQR's por exceso de consumo deberán ser asignados al área técnica.



# Bibliografía



- Martínez López, F. J. y Gallegos Ruiz, A. (2017). Programación de bases de datos relacionales. Paracuellos de Jarama, Madrid, RA-MA Editorial. Recuperado de <https://elibro-net.bdigital.sena.edu.co/es>.



# GRACIAS

Línea de atención al ciudadano: 01 8000 910270  
Línea de atención al empresario: 01 8000 910682  
Instructor: Pablo Ortiz



[www.sena.edu.co](http://www.sena.edu.co)