

BASES DE DATOS

The background of the slide features several overlapping, wavy lines in orange, light blue, and lime green, creating a modern, abstract design.

CLASE 3

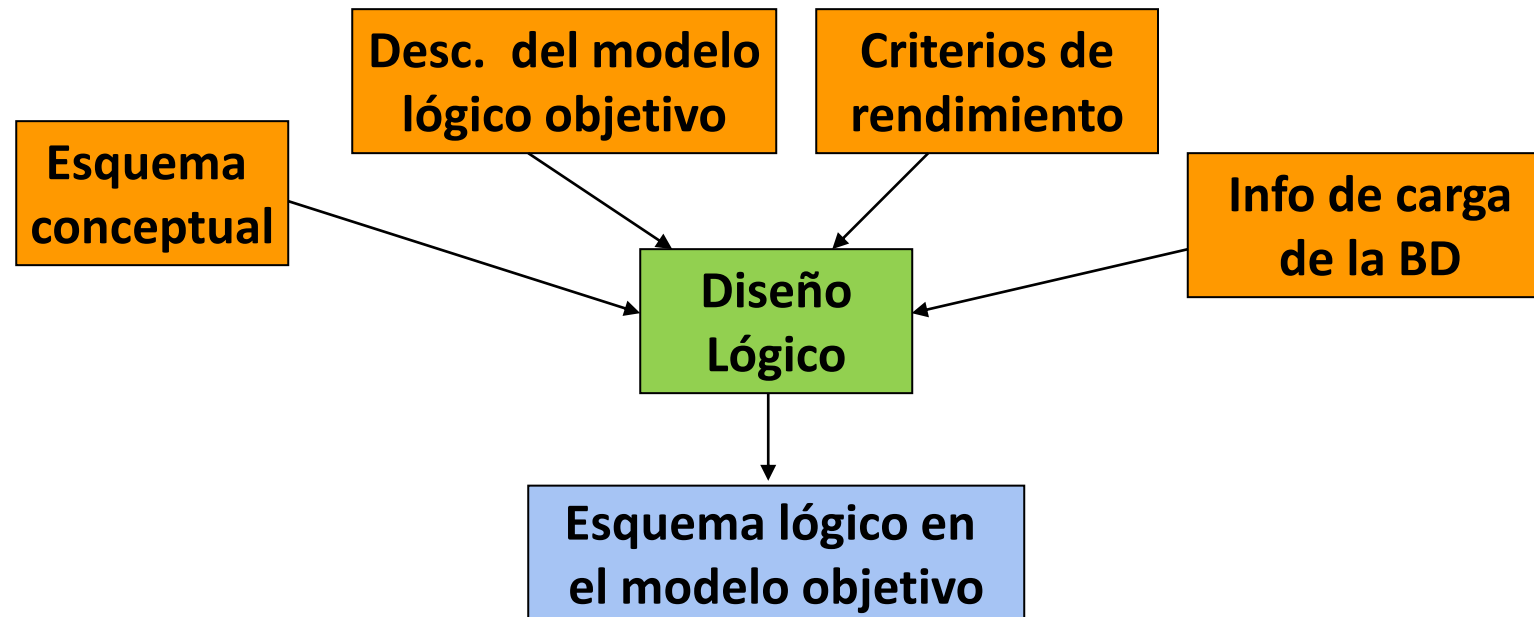
Diseño Lógico

- **Diseño lógico**
 - **Esquema Conceptual**: captación y representación clara de las **necesidades** definidas en la especificación de requerimientos
 - **Diseño lógico**: conversión del esquema conceptual en un esquema lógico

Diseño Lógico

- **Diseño lógico**

- Enfoque global del diseño lógico (independientemente del modelo utilizado)



Diseño Lógico

- **Simplificación del esquema conceptual**

→ modelo relacional ←

- Datos derivados y ciclos de relaciones
- Eliminación de atributos polivalentes
- Eliminación de atributos compuestos
- Eliminación de jerarquías de generalización
- Otras decisiones (no obligatorias)
 - Partición de Entidades / Interrelaciones
 - Fusión de Entidades / Interrelaciones

Diseño Lógico

- **Datos derivados**

- **Ventajas**

- No se necesita calcular el valor → reduce el número de accesos a la BD

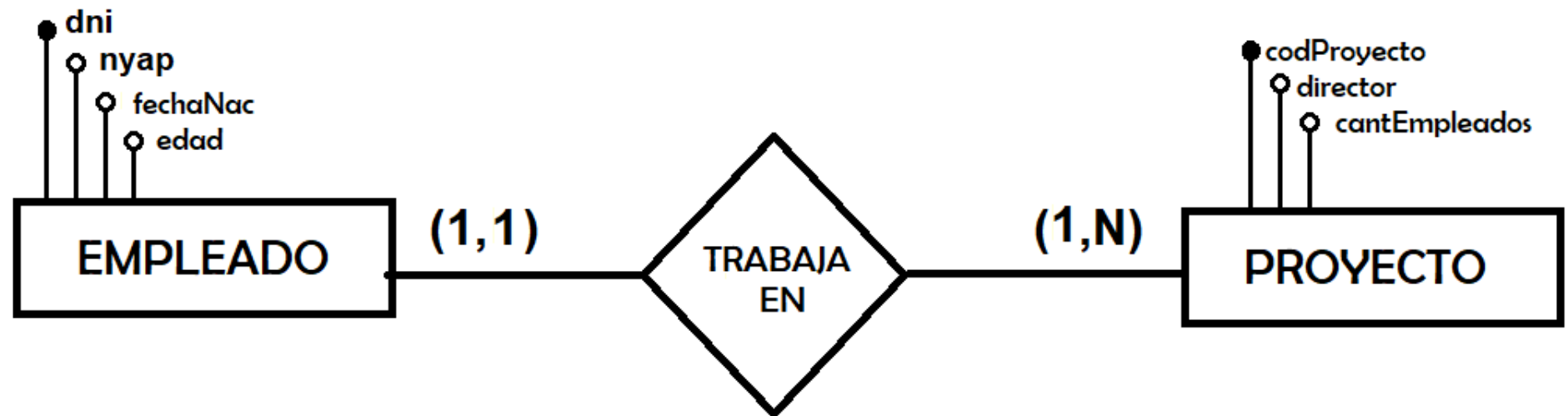
- **Desventajas**

- Procesamiento adicional para mantener los datos derivados
 - Requiere más espacio de almacenamiento

Diseño Lógico

- **Datos derivados**

- Ej:



Diseño Lógico

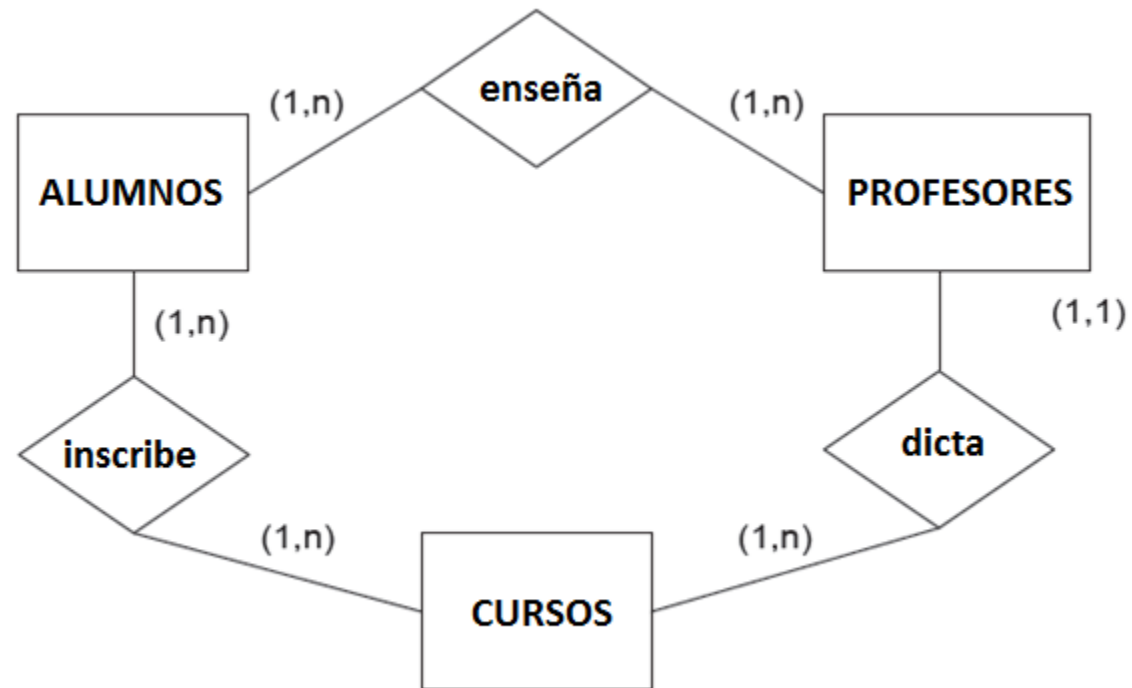
- **Ciclos de relaciones**

- En algunos casos, un ciclo puede generar **repetición** de información
 - Identificar relaciones responsables de la redundancia
 - Eliminar redundancia: criterio del diseñador
 - Conservarlos → afecta minimalidad
 - Eliminarlos → afecta performance

Diseño Lógico

- Ciclos de relaciones

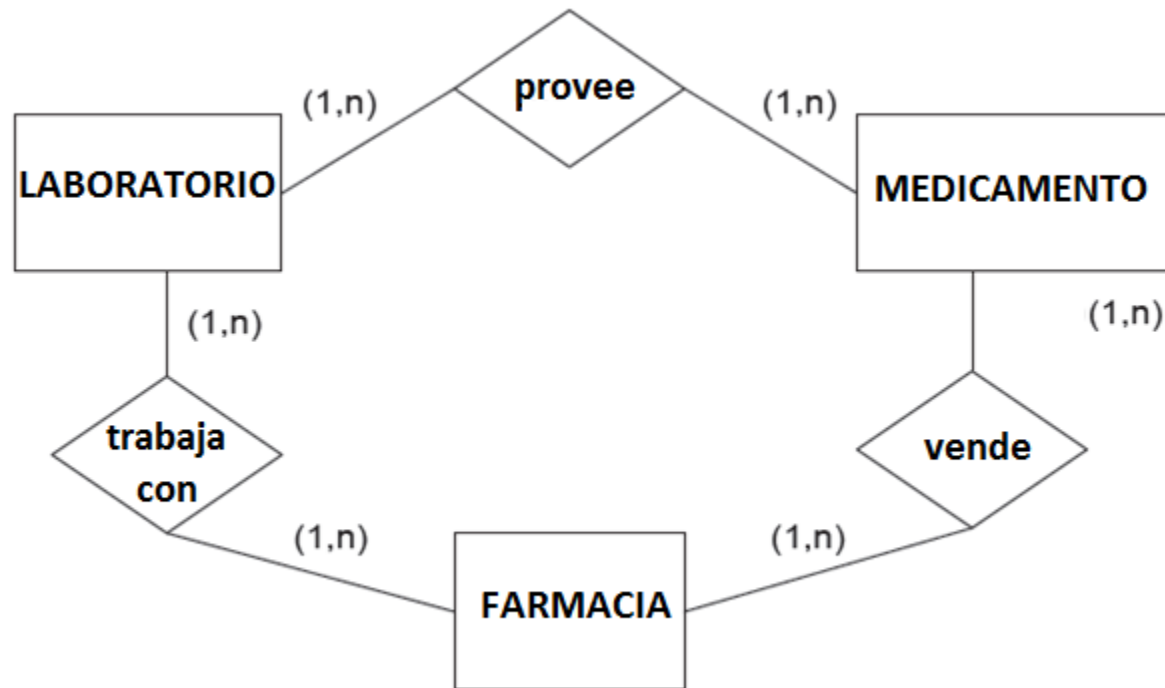
- Ej. 1:



Diseño Lógico

- Ciclos de relaciones

- Ej. 2:



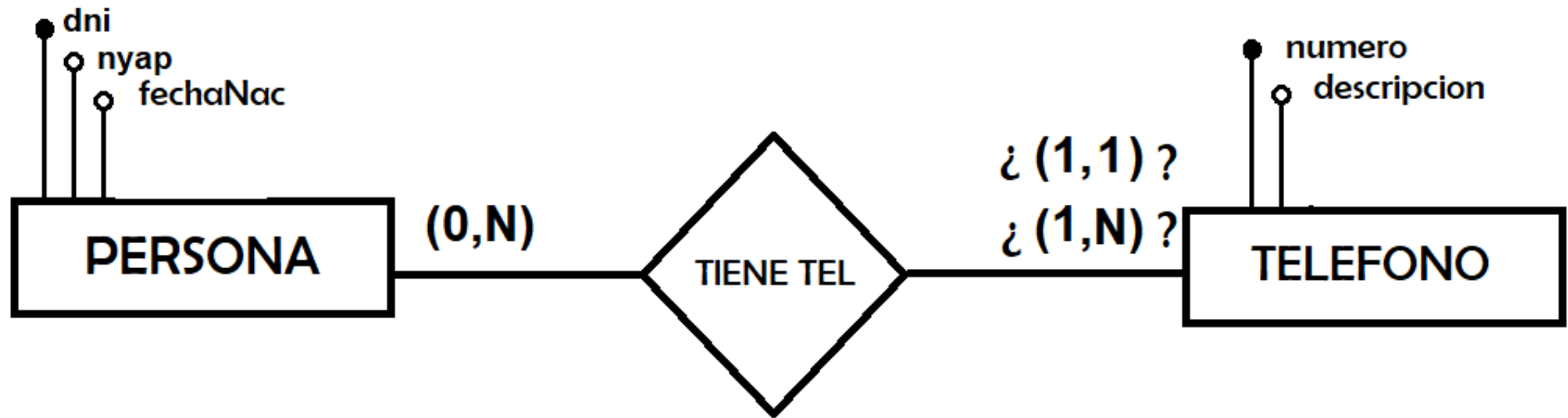
Diseño Lógico

- **Eliminación de atributos polivalentes**
 - Los SGBD no permiten atributos con múltiples valores de una dimensión determinada **dinámicamente**
 - Resolución
 - Quitar el atributo de la entidad
 - Generar una nueva entidad con ese atributo y establecer una nueva relación con la entidad a la cuál pertenecía

Diseño Lógico

- Eliminación de atributos polivalentes

- Ej:



Diseño Lógico

- **Eliminación de atributos compuestos**
 - Alternativas de resolución
 - Considerar el atributo compuesto entero como **un solo atributo**
 - Considerar todos sus componentes como **atributos individuales**
 - Generar una **nueva entidad**, la que representa el atributo compuesto, conformada por cada uno de los atributos simples que contiene

Diseño Lógico

- **Eliminación de jerarquías de generalización**
 - Los modelos lógicos <relacionales> no permiten representar jerarquías de generalización
 - Se deben representar usando sólo entidades e interrelaciones
 - A tener en cuenta:
 - La **herencia** de atributos se debe indicar **explícitamente** → se debe captar la interrelación implícita **ES_UN**
 - Tres casos

Diseño Lógico

- **Eliminación de jerarquías de generalización**
 - **Caso 1: integrar la jerarquía de generalización**
 - La superentidad contendrá sus atributos y también los de todas las subentidades
 - Solución **muy simple**
 - Generan **valores nulos** de atributos → card (0,1)
 - Aplicable a **cualquier caso** de jerarquía de generalización

Diseño Lógico

- **Eliminación de jerarquías de generalización**
 - Caso 2: eliminar la superentidad pero retener las subentidades
 - Desventajas
 - Repetición de atributos y operaciones de la superentidad
 - El uso repetitivo de la misma interrelación crea una redundancia no deseable
 - Aplicable en caso de cobertura **total exclusiva**
 - Inadecuado para cobertura superpuesta o parcial

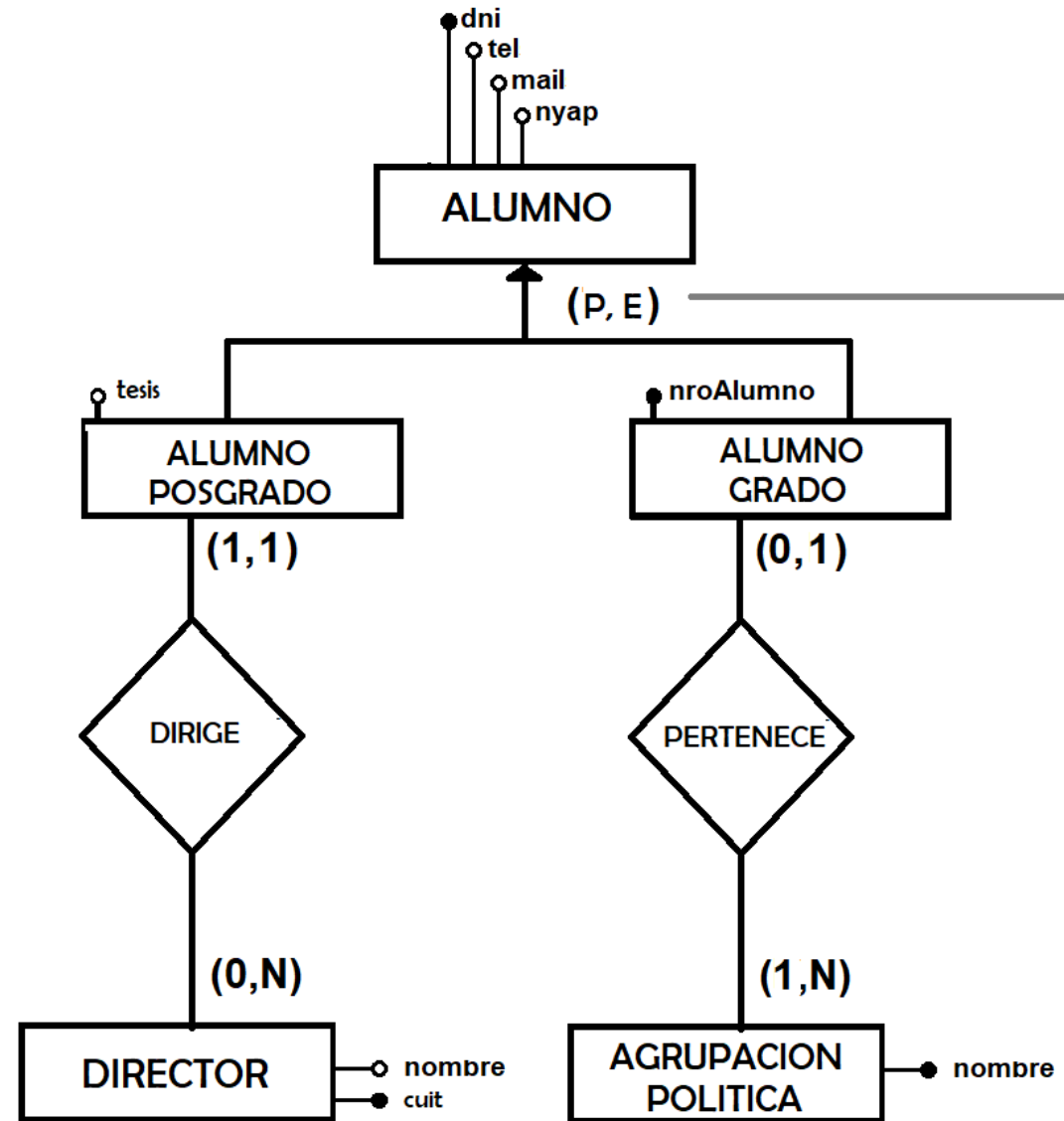
Diseño Lógico

- **Eliminación de jerarquías de generalización**
 - **Caso 3: retener todas las entidades**
 - Se establecen explícitamente las interrelaciones entre las superentidades y subentidades
 - Es el caso más **general** y **aplicable**
 - Proporciona **redundancia** inherente (a nivel conceptual) al representar la relación **ES_UN** de la jerarquía a través de una **interrelación explícita**

Diseño Lógico

- Eliminación de jerarquías de generalización

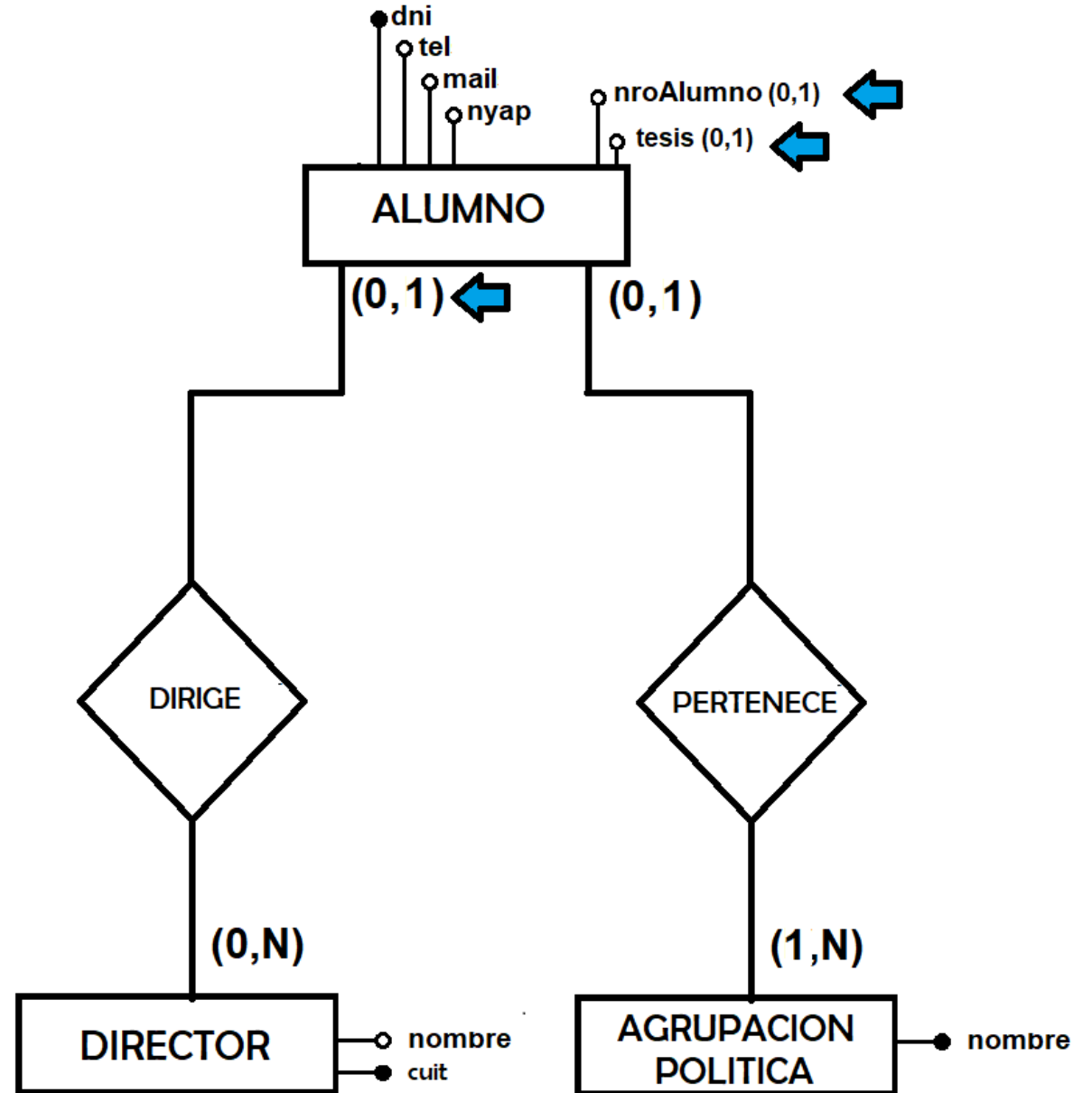
• Ej:



Diseño Lógico

- Eliminación de jerarquías de generalización

- Ej: Caso 1.



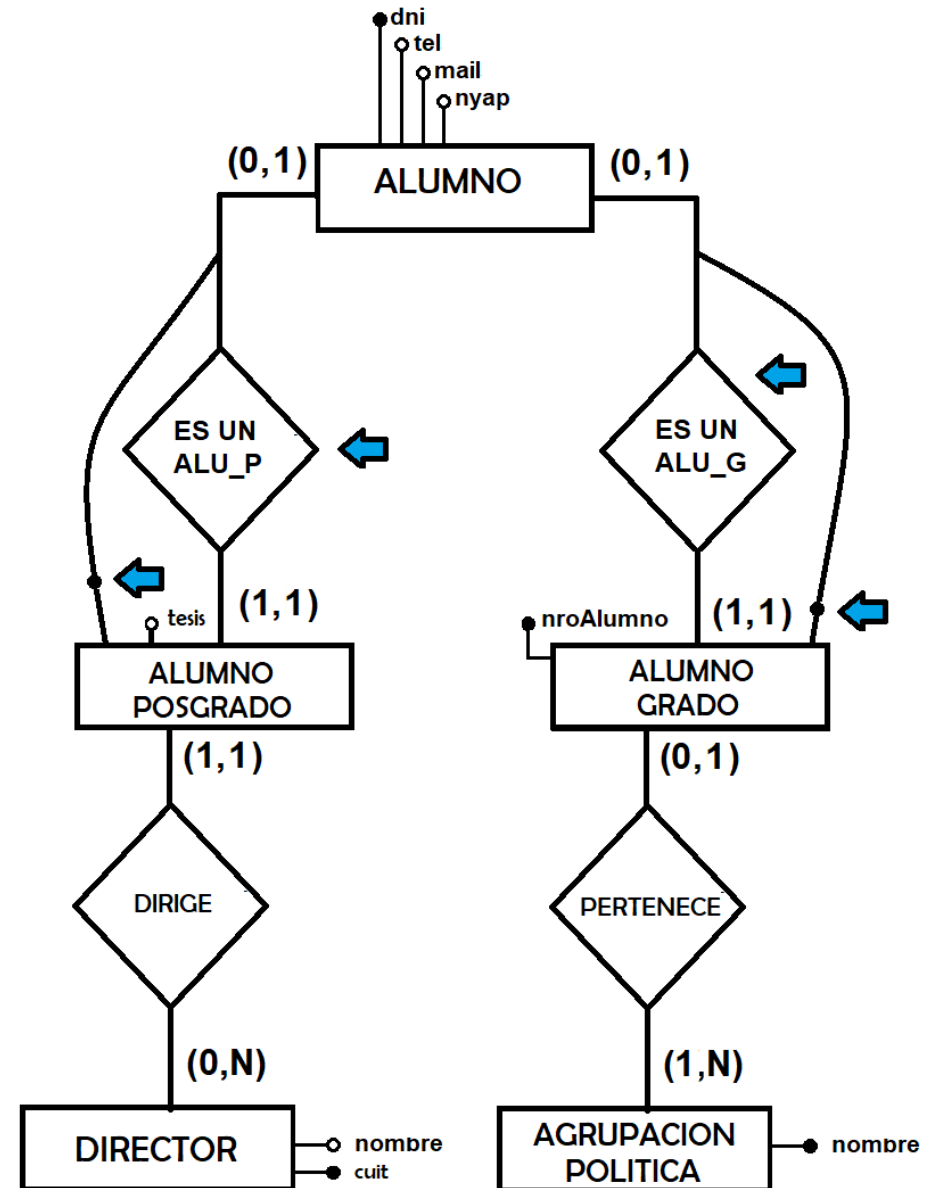
Diseño Lógico

- **Eliminación de jerarquías de generalización**
 - Ej: **Caso 2.**
 - En este ejemplo **NO ES POSIBLE** esta alternativa, ya que tiene cobertura PARCIAL
 - Si quitamos a la entidad ALUMNO, los alumnos ingresantes no tienen cabida en el esquema → **se pierden datos**

Diseño Lógico

- Eliminación de jerarquías de generalización

- Ej: Caso 3.



Diseño Lógico

- **Otras decisiones (no obligatorias)**
 - **Partición de Entidades**
 - Horizontal (entidades)
 - Vertical (atributos)
 - **Partición de Interrelaciones**
 - En situaciones de relaciones de uno a muchos y de muchos a muchos
 - **Fusión de Entidades e Interrelaciones**

Diseño Físico

- El **Modelo Relacional** es un modelo basado en registros ampliamente difundido y utilizado
 - Propuesto por **Edgar F. Codd** en 1970
 - Simple, potente y formal
 - Representa a la BD como una **colección de archivos** → **tablas**

Diseño Físico

- **Modelo Relacional**
 - **Tabla (o relación, de ahí el nombre del modelo):** elemento básico del modelo
 - **Esquema de tabla (relación):** una agregación de atributos
 - **Dominio:** conjunto de valores que puede tomar un atributo
 - **Esquema de BD** → colección de definiciones de relaciones

Diseño Físico

- **Modelo Relacional**
 - **Tabla:** contiene los datos en filas y columnas
 - Filas de la tabla → **tuplas**
 - Columnas de la tabla → **atributos**
 - Una tabla **no puede tener tuplas duplicadas**

Diseño Físico

- **Conversión del esquema lógico al físico**

→ modelo relacional ←

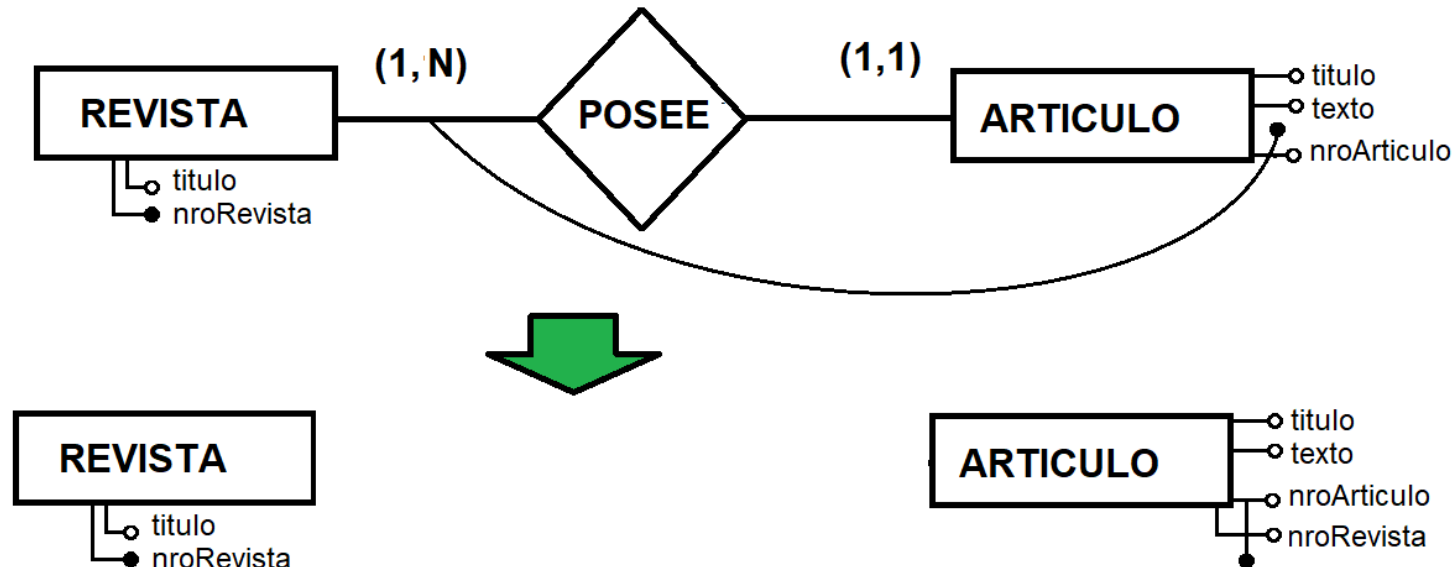
- Eliminación de identificadores externos
- Selección de claves
 - Candidatas
 - Primaria
 - Secundarias
- Conversión de entidades
- Conversión de relaciones

Diseño Físico

- **Eliminación de identificadores externos**
 - Cada una de las entidades que conforman el esquema lógico, ahora debe poseer sus **identificadores definidos en forma interna**
 - Incorporación de atributos externos
 - Si hay más de un identificador se debe elegir el más adecuado/representativo → **selección de claves**

Diseño Físico

- Eliminación de identificadores externos
 - Ej. de incorporación de atributo externo en una entidad
 - El dibujo es a modo de ejemplo, ya que el esquema físico **NO ES GRÁFICO**.



Diseño Físico

- **Claves**

- Clave

- Similar al concepto de **identificador** en el modelo ER
 - Un atributo o grupo de atributos de una tabla que **identifica de manera unívoca cada tupla** de la misma
 - El modelo relacional solo acepta **identificación interna**

Diseño Físico

- **Claves**

- Clave Candidata

- Una tabla puede tener una o más claves → cada una se denomina **clave candidata**

- Clave Primaria

- Es la **clave candidata elegida** para identificar las tuplas de la tabla → se subraya en el esquema de la relación

- Clave Foránea

- Es un atributo o grupo de atributos de una tabla, que es **clave primaria en otra tabla**

Diseño Físico

- **Claves**
 - Claves Primaria y Candidatas → importancia de su correcta elección
 - Índices primarios y secundarios
 - Performance final de la BD
 - Dominios autoincrementales

Diseño Físico

- **Integridad referencial**
 - La **integridad referencial (IR)** es una propiedad deseable, no obligatoria, de las BD relacionales
 - Plantea **restricciones entre tablas** y sirve para mantener la **consistencia** entre las tuplas de dichas tablas
 - Ejemplo:
 - **FACTURAS** = (nroFactura, fecha, monto, nroCliente)
 - **CLIENTES** = (nroCliente, nombre, dirección)

Diseño Físico

- **Integridad referencial**
 - Posibilidades de **IR** que permite un **SGBD**:
 - Restringir la operación
 - Realizar la operación “en cascada”
 - Establecer la clave foránea en nulo
 - No hacer nada → **se delega responsabilidad**

Diseño Físico

- **Conversión de entidades**
 - Cada una de las entidades definidas se convierte en una **tabla**
 - Excepciones
 - La clave primaria se indica **subrayada**
 - Es posible agregar campos para **índices autoincrementales**

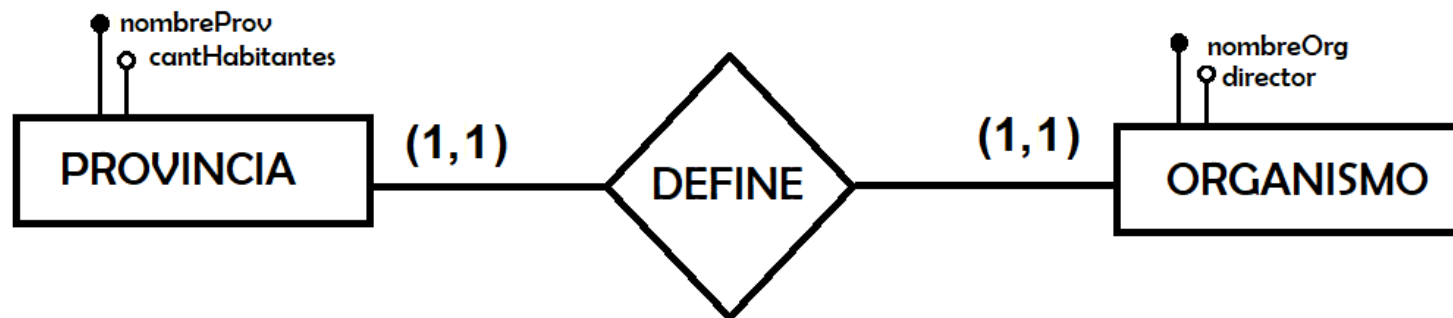
Diseño Físico

- **Conversión de relaciones**
 - **Uno a uno**
 - Participación total de ambos lados
 - Participación parcial de un lado
 - Participación parcial de ambos lados

Diseño Físico

- **Conversión de relaciones**

- **Uno a uno** → Participación total de ambos lados (excepción)



Provincia (nombreProv, nombreOrg, cantHabitantes, director)

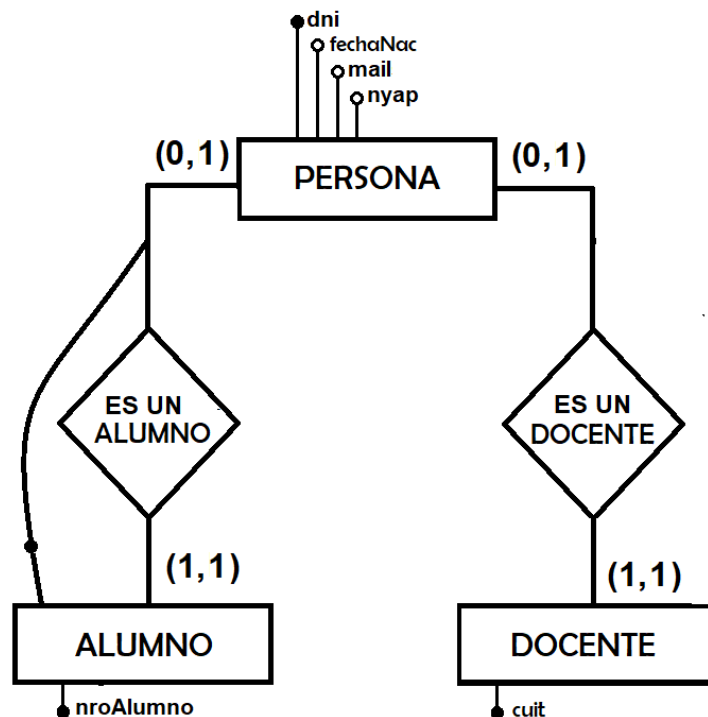
ó

Organismo (nombreOrg, nombreProv, cantHabitantes, director)

Diseño Físico

- **Conversión de relaciones**

- **Uno a uno** → Participación parcial de un lado



Persona (dni, fechaNac, mail, nyap)

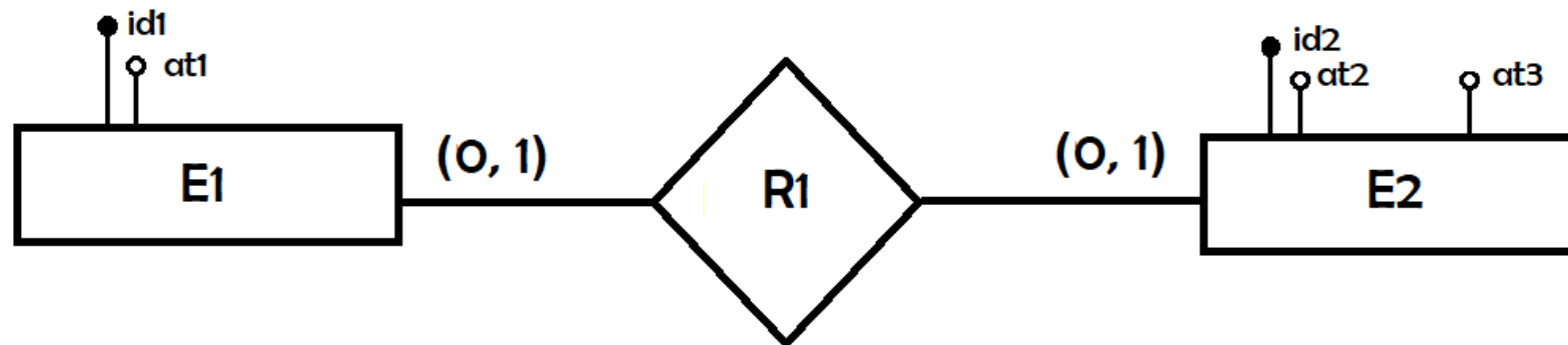
Alumno (nroAlumno, dni)

Docente (cuit, dni)

Diseño Físico

- **Conversión de relaciones**

- **Uno a uno** → Participación parcial de ambos lados (muy atípico)



E1 (id1, at1)

E2 (id2, at2, at3)

R1 (id1, id2) ó **R1** (id1, id2)

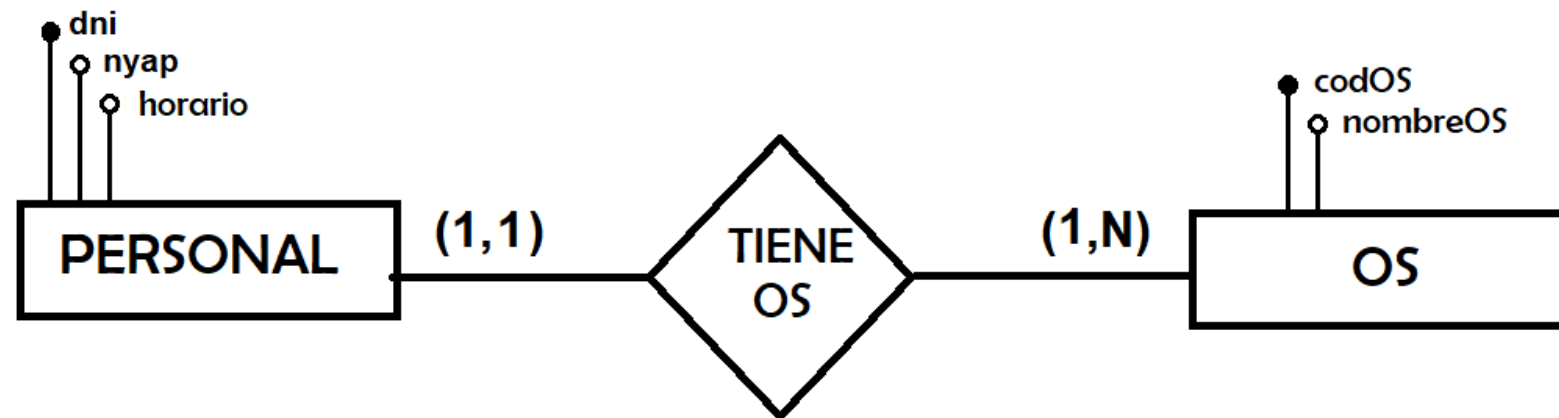
Diseño Físico

- **Conversión de relaciones**
 - **Uno a muchos**
 - Participación total de ambos lados
 - Participación parcial del lado de muchos
 - Participación parcial del lado de uno
 - Participación parcial de ambos lados

Diseño Físico

- **Conversión de relaciones**

- **Uno a muchos** → Participación total de ambos lados



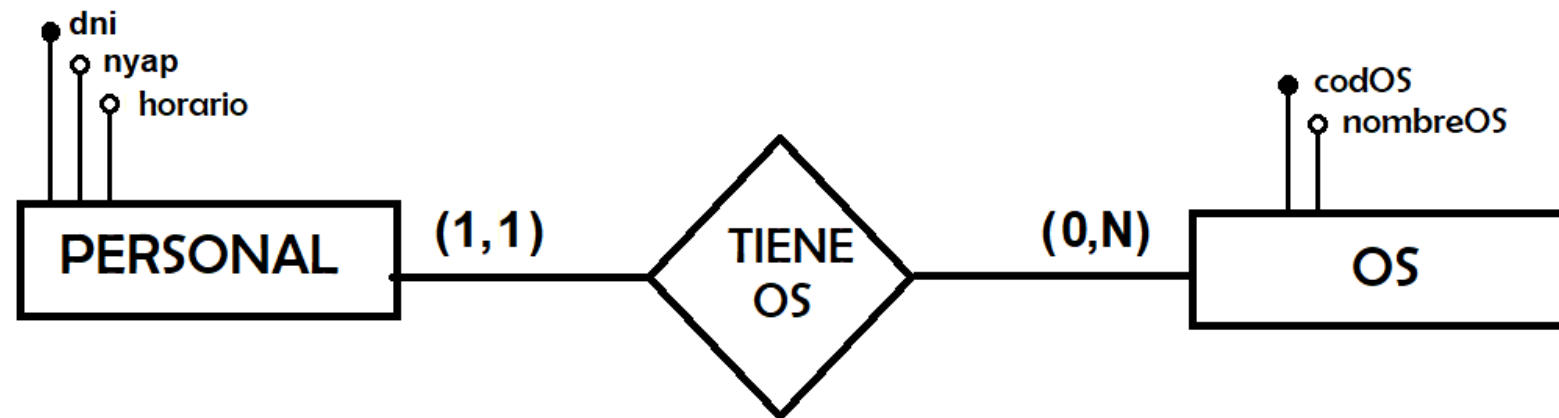
OS (codOS, nombreOS)

Personal (dni, nyap, horario, codOS)

Diseño Físico

- **Conversión de relaciones**

- **Uno a muchos** → Participación parcial del lado de muchos



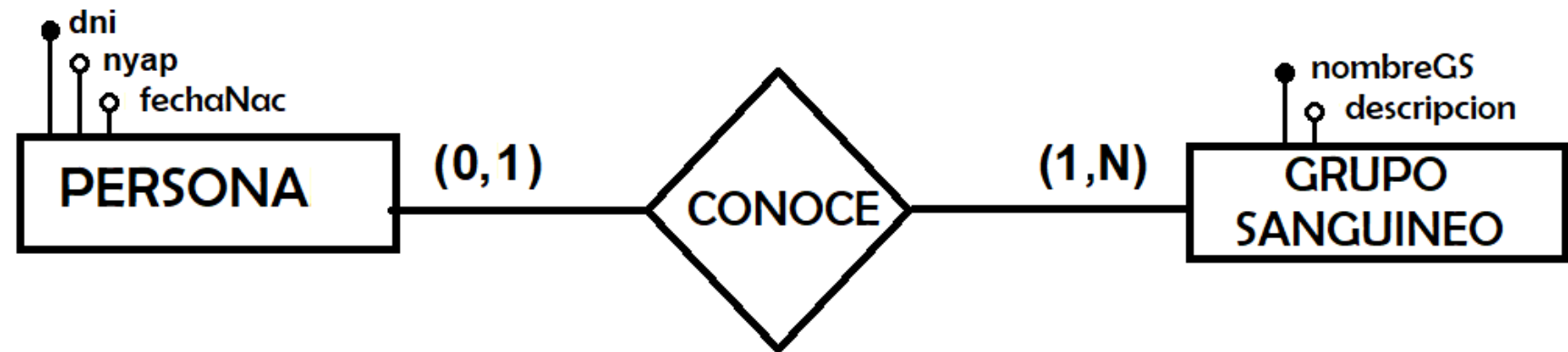
OS (codOS, nombreOS)

Personal (dni, nyap, horario, codOS)

Diseño Físico

- **Conversión de relaciones**

- **Uno a muchos** → Participación parcial del lado de uno



- Solución 1:

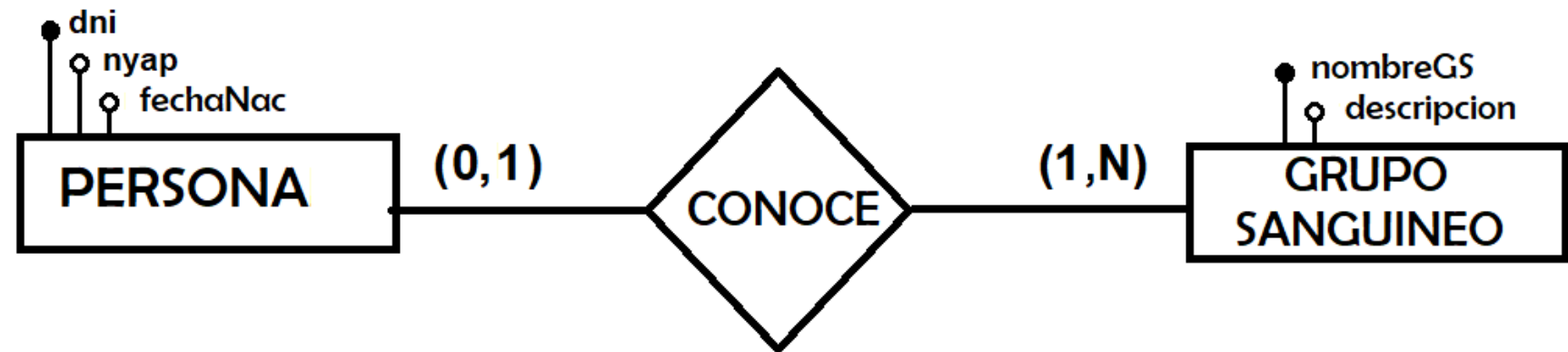
Persona (dni, nyap, fechaNac, nombreGS)

GrupoSanguineo (nombreGS, descripcion)

Diseño Físico

- **Conversión de relaciones**

- **Uno a muchos** → Participación parcial del lado de uno



- Solución 2:

Persona (dni, nyap, fechaNac)

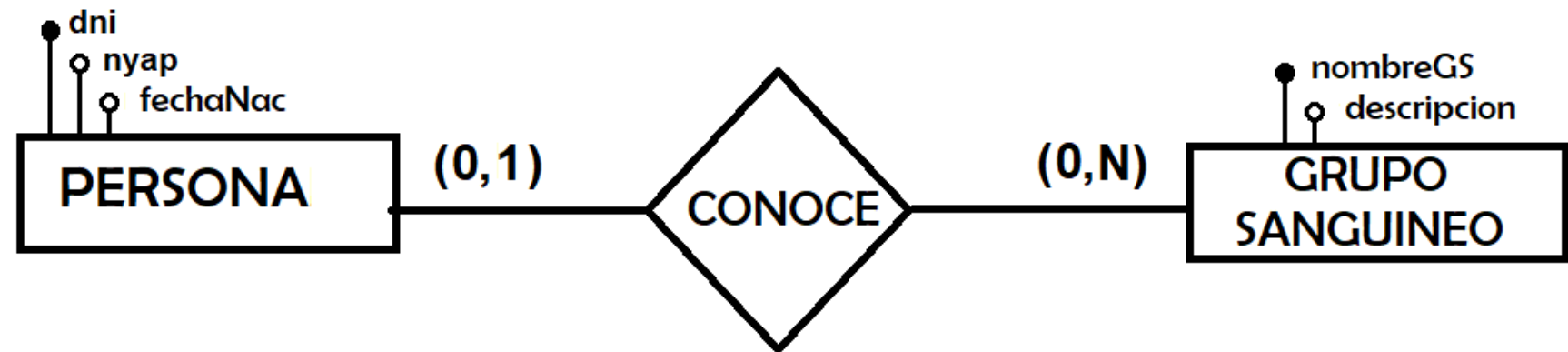
GrupoSanguineo (nombreGS, descripcion)

Conoce (dni, nombreGS)

Diseño Físico

- **Conversión de relaciones**

- **Uno a muchos** → Participación parcial de ambos lados



Idem anterior, es mejor la segunda alternativa.

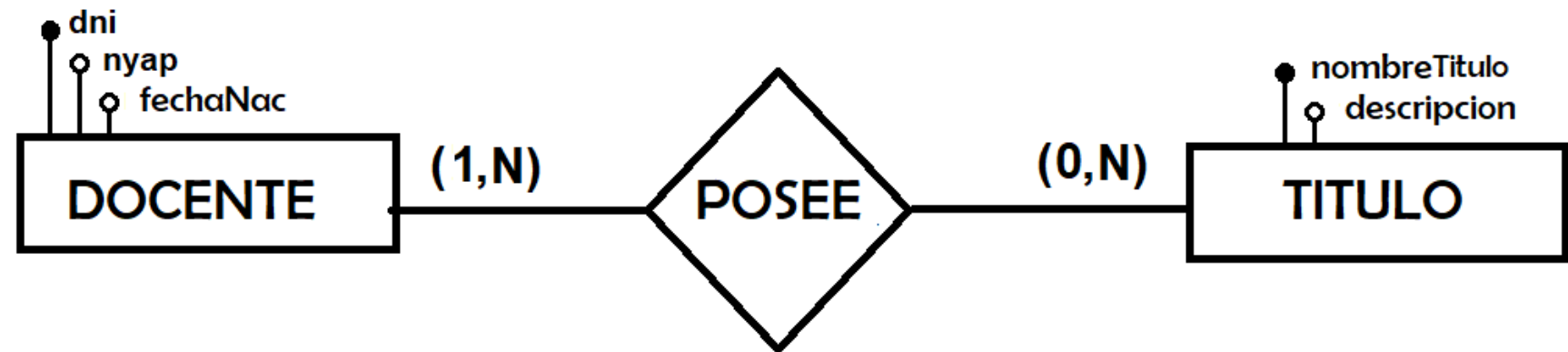
Diseño Físico

- **Conversión de relaciones**
 - Muchos a muchos
 - Recursivas
 - N-arias

Diseño Físico

- Conversión de relaciones

- Muchos a muchos



Docente (dni, nyap, fechaNac)

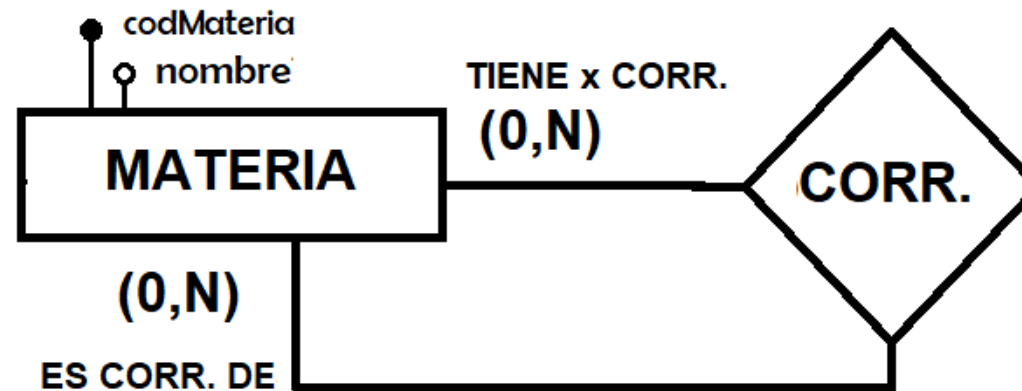
Titulo (nombreTitulo, descripcion)

Posee (dni, nombreTitulo)

Diseño Físico

- Conversión de relaciones

- Recursivas



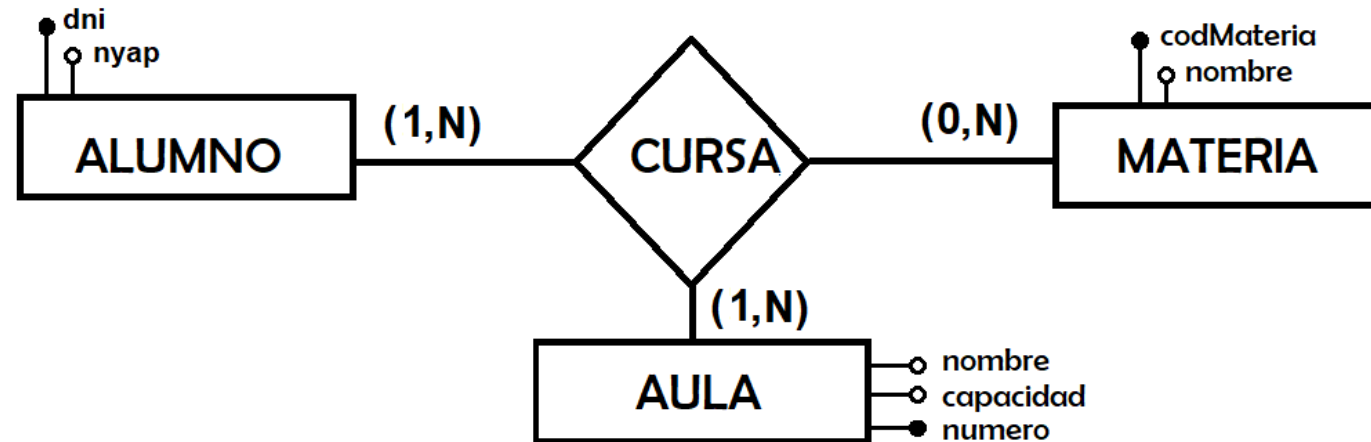
Materia (codMateria, nombre)

Correlativa (codMateriaOriginal, codMateriaCorrelativa)

Diseño Físico

- Conversión de relaciones

- N-arias



Alumno (dni, nyap)

Materia (codMateria, nombre)

Aula (numero, nombre, capacidad)

Cursa (dni, codMateria, numero)