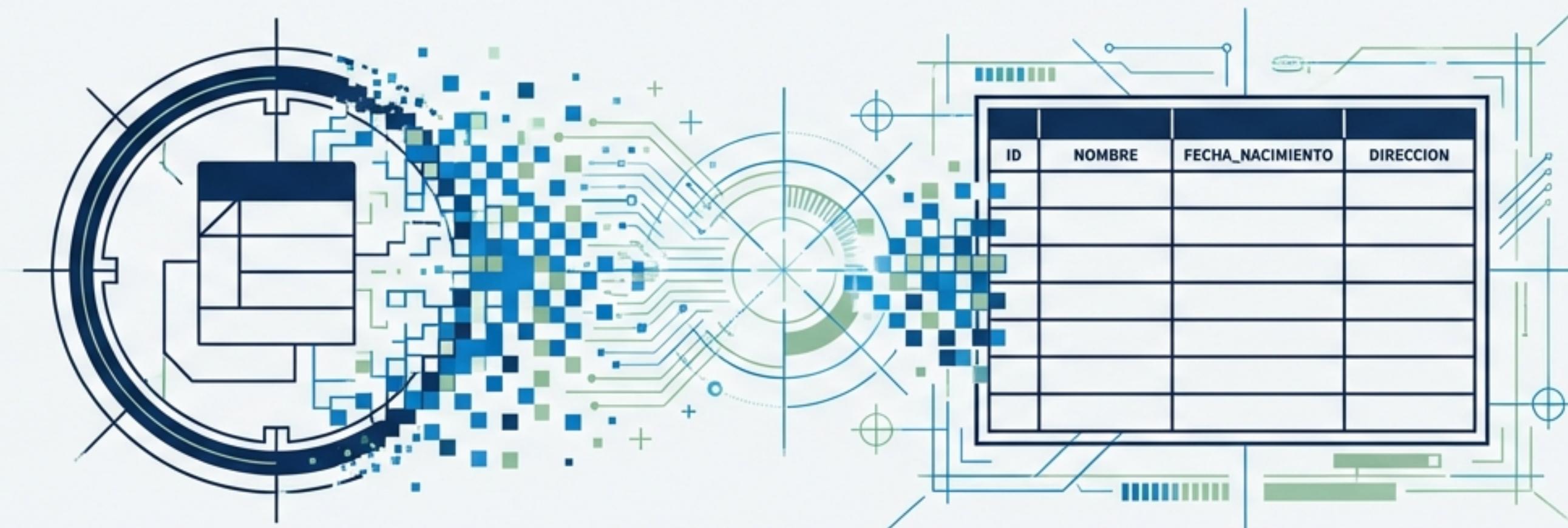


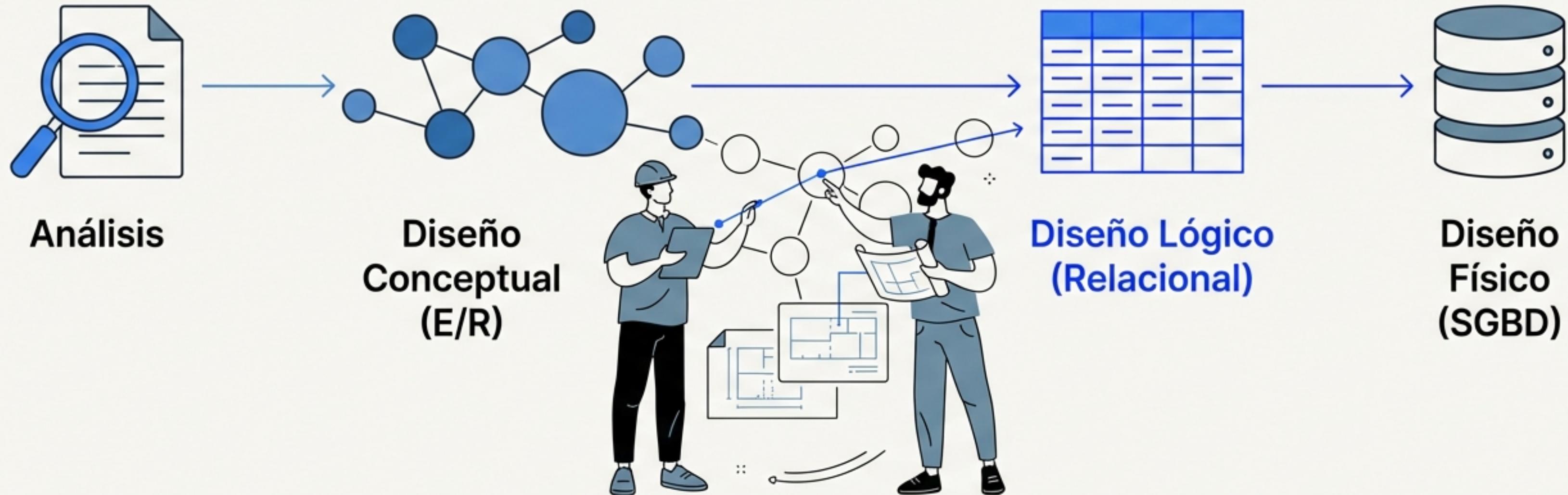
Diseño de Bases de Datos: Del Concepto a la Realidad

Guía de transformación: Del Diagrama E/R al Modelo Relacional y Normalización (Tema 7)



El diseño conceptual (E/R) es solo el mapa. El diseño lógico (Relacional) es el terreno. Este documento es el plano de construcción para transformar diagramas en estructuras de datos eficientes, sin redundancias y listas para su implementación en un SGBD.

El Contexto del Diseño



El Punto de Partida

Diagrama Entidad-Relación (E/R) ya definido.

El Objetivo

Construir el Modelo Relacional, cercano al SGBD.

El Método

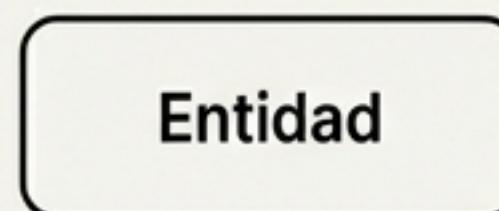
Aplicar reglas rigurosas de transformación a entidades y relaciones.

La Meta Final

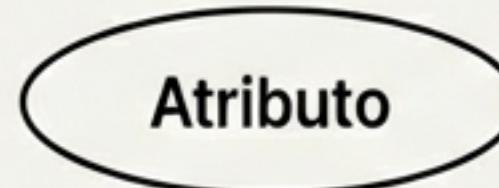
Un modelo normalizado, eficiente y sin redundancias.

La Piedra Rosetta: Equivalencias

MODELO E/R (Conceptual)



Entidad



Atributo

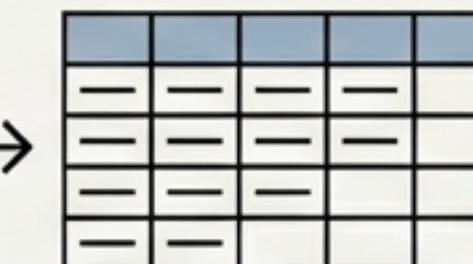


Identificador



Relación

Entidad se convierte en Tabla



Tabla

Atributo se convierte en Columna



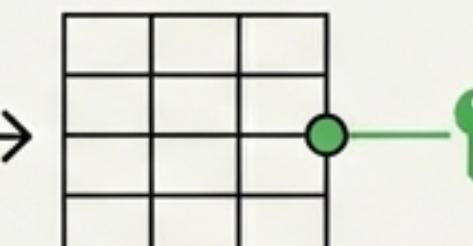
Columna

Identificador Principal se convierte en Clave Primaria (PK)



Clave Primaria (PK)

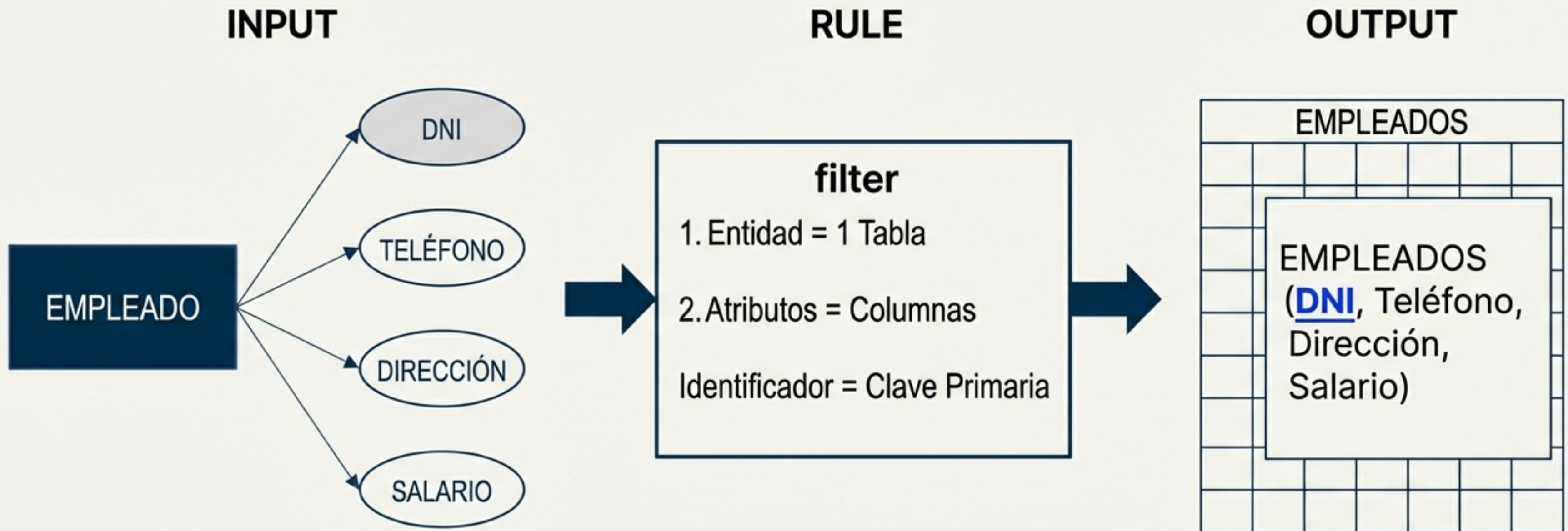
Relación se convierte en Clave Foránea (o Tabla)



Clave Foránea (FK)

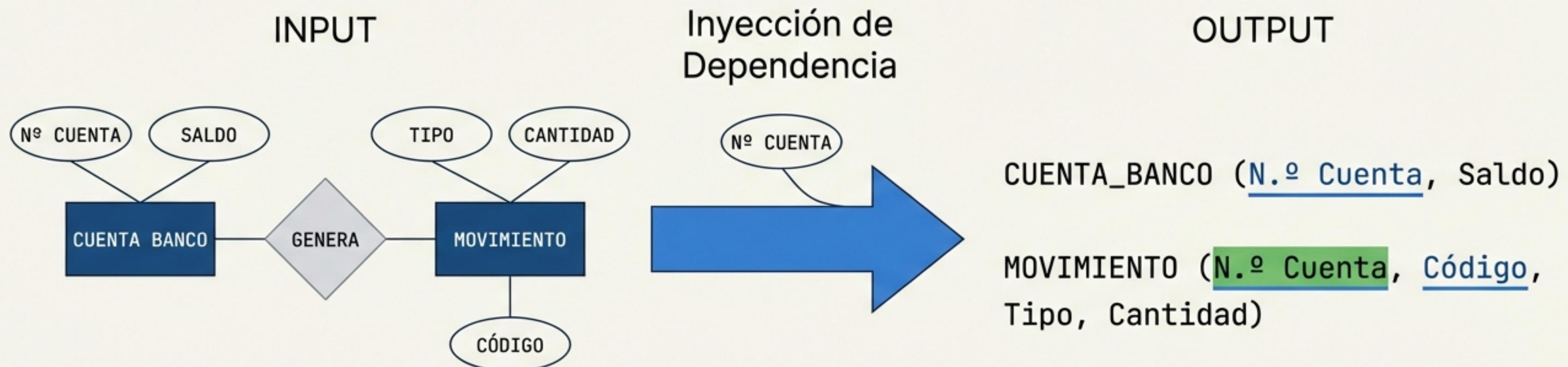
Insight Clave: Cada ocurrencia de una entidad se convierte en una fila (tupla) dentro de la tabla.

Transformación de Entidades Fuertes



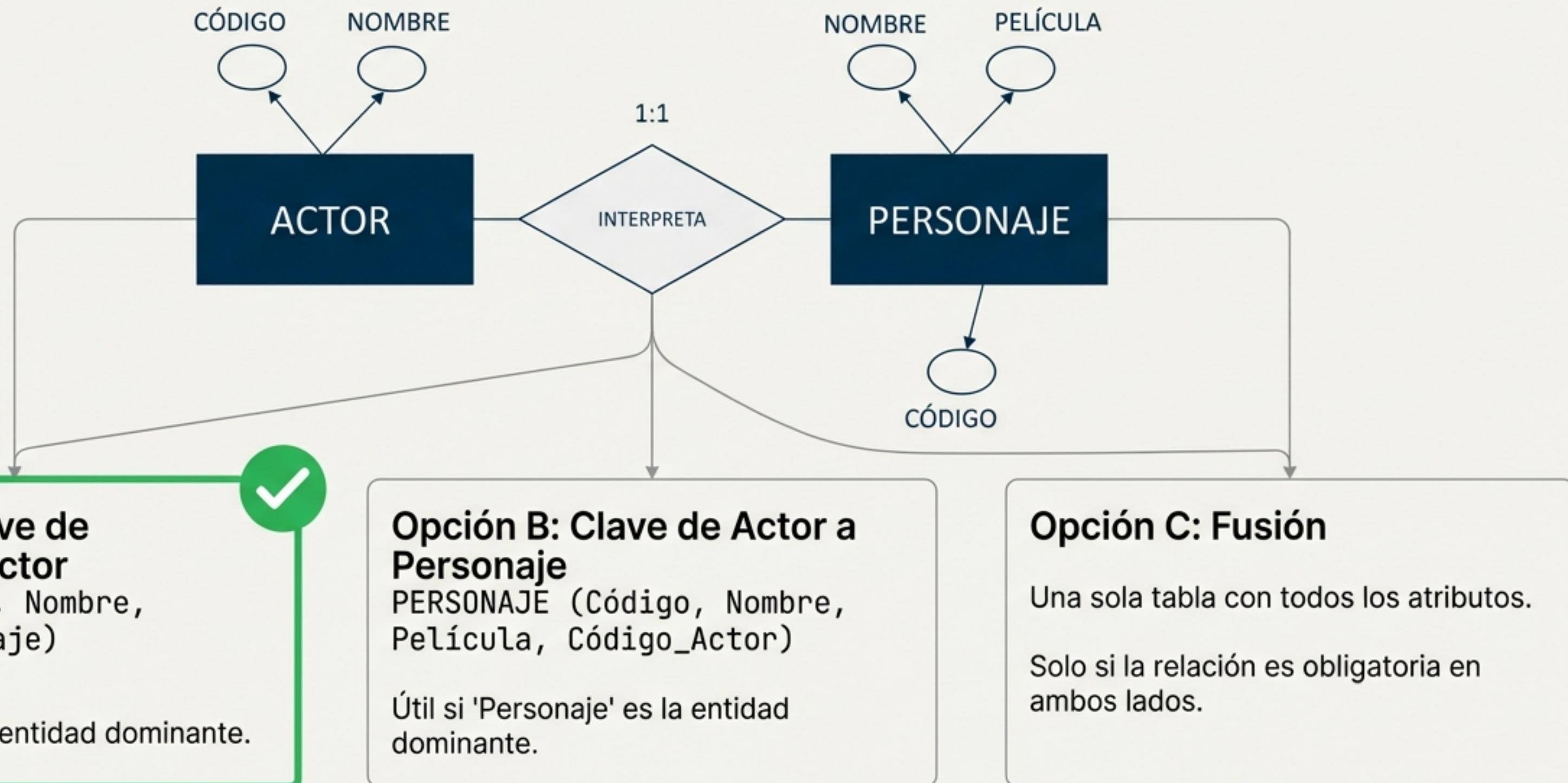
La entidad fuerte es el bloque básico. No depende de nadie. Se traduce directamente a una tabla independiente.

Entidades Débiles: La Dependencia



La conexión es implícita. Añadimos la Clave Primaria de la entidad fuerte (N.º Cuenta) dentro de la estructura de la débil.

Relaciones 1:1 (Uno a Uno)

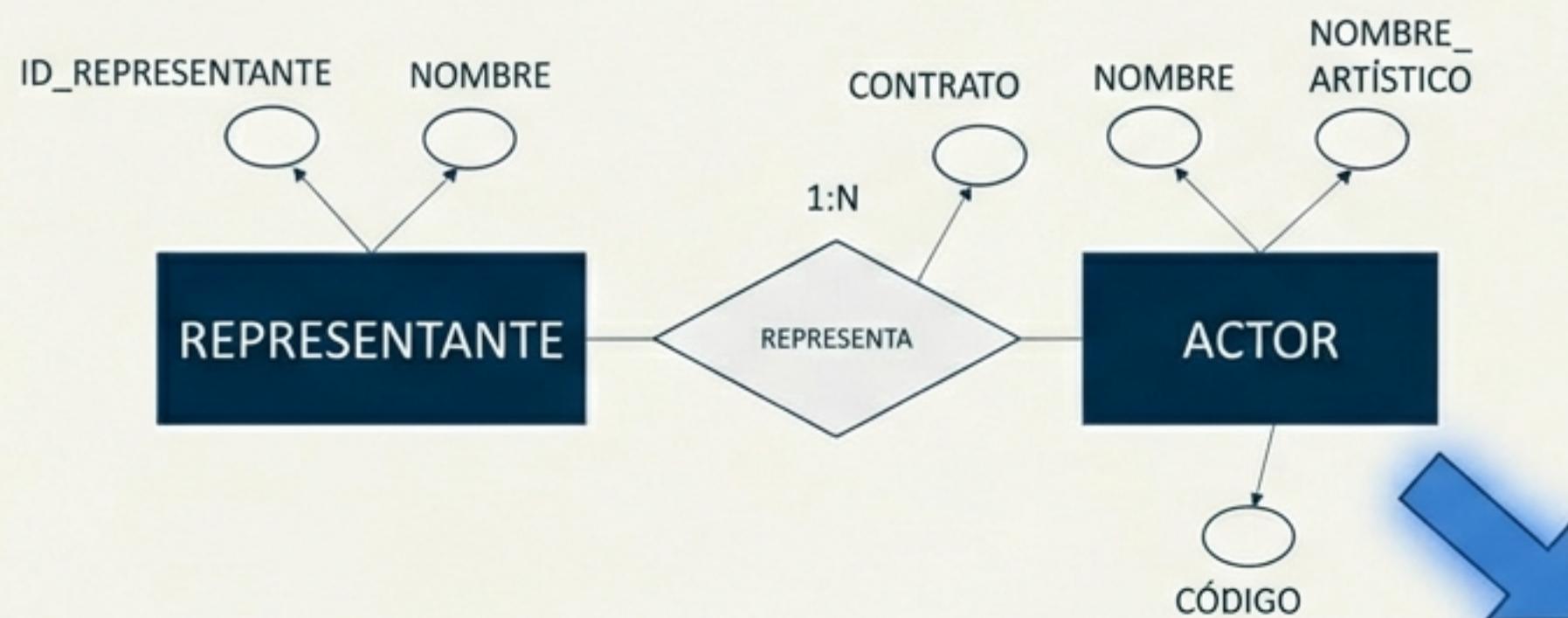


Regla General: No se genera nueva tabla. Se propaga la clave.

Relaciones 1:N (La Regla de Oro)

La Regla de Oro

La clave viaja del 1 al N.

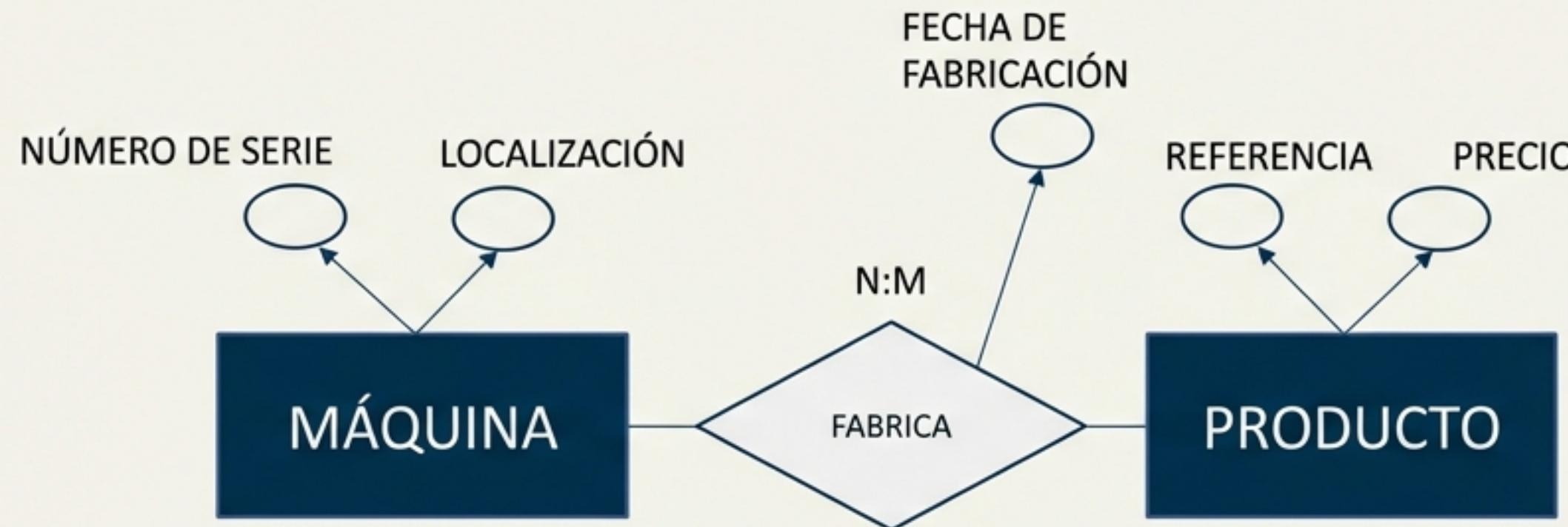


REPRESENTANTE ([Id_Representante](#), Nombre)

ACTOR ([Código](#), Nombre_artístico, Nombre)

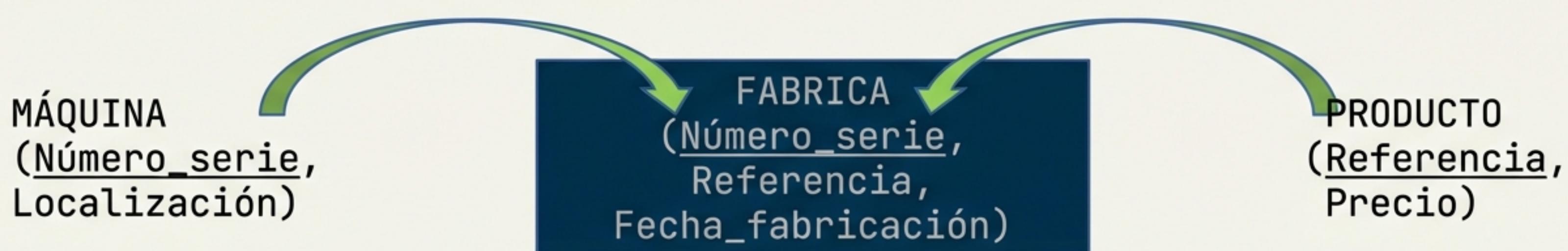
ACTOR ([Código](#), Nombre, [Id_Representante](#), Contrato)

Relaciones N:M (Muchos a Muchos)

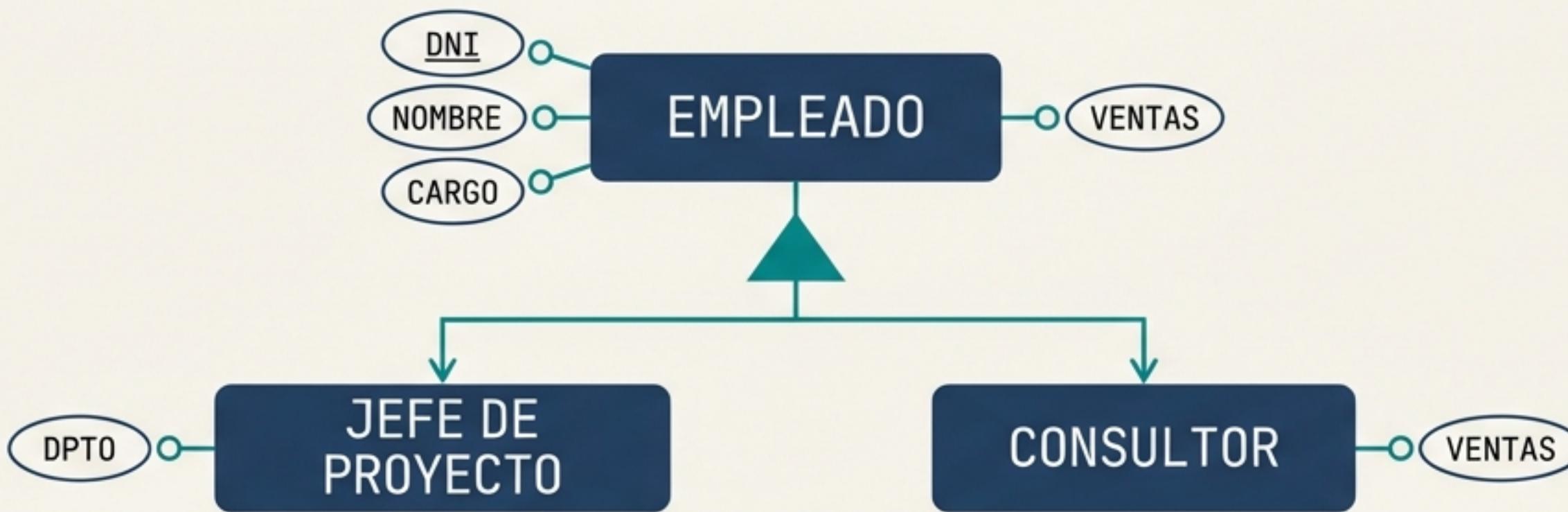


Problema: Un producto tiene muchas máquinas. Una máquina tiene muchos productos. No cabe en una celda.

Solución: Nace una Tabla Intermedia.



Jerarquías: Generalización y Especialización



Estrategia 1: Todas las tablas

3 Tablas. Subclases tienen PK de la Superclase.

EMPLEADOS(DNI, Nombre, Cargo...),
JEFES(DNI, DPTO...),
CONSULTORES(DNI, VENTAS...)

Estrategia 2: Solo Subclases

2 Tablas. Eliminar Empleado. Repetir atributos.

JEFES(DNI, Nombre, Cargo, DPTO...),
CONSULTORES(DNI, Nombre, Cargo, VENTAS...)

Estrategia 3: Una Tabla (Tipo)

1 Tabla gigante. Campo discriminador.

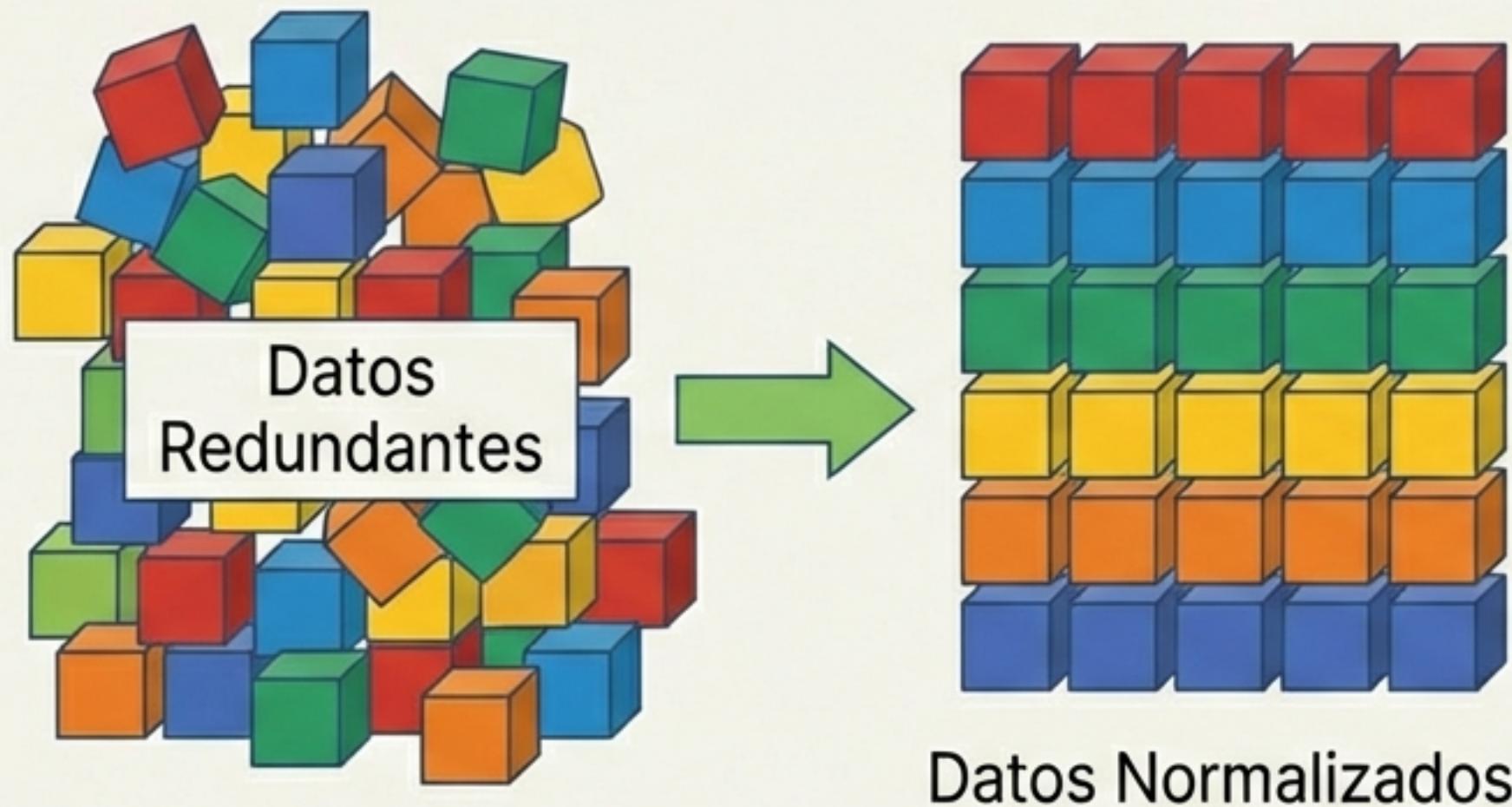
EMPLEADOS(DNI, Nombre, Cargo, DPTO, VENTAS,
Tipo_Empleado)

Estrategia 4: Una Tabla (Booleanos)

1 Tabla. Flags para cada rol.

EMPLEADOS(DNI, Nombre, Cargo, DPTO, VENTAS,
EsJefe, EsConsultor)

Normalización: El Control de Calidad

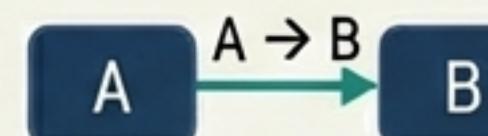


Objetivos

- Evitar Redundancia (Ahorro de espacio)
- Integridad de Datos (Evitar inconsistencias)

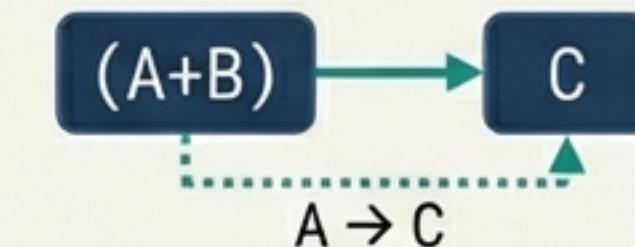
La Clave: Dependencias Funcionales

Completa



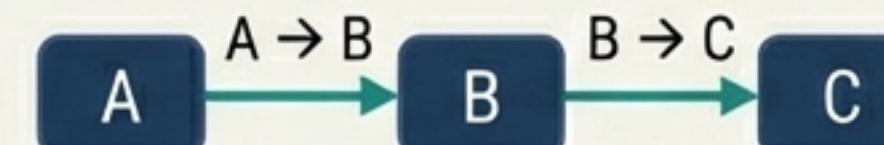
A determina B
únivamente.

Parcial



C depende solo de
una parte de la clave.

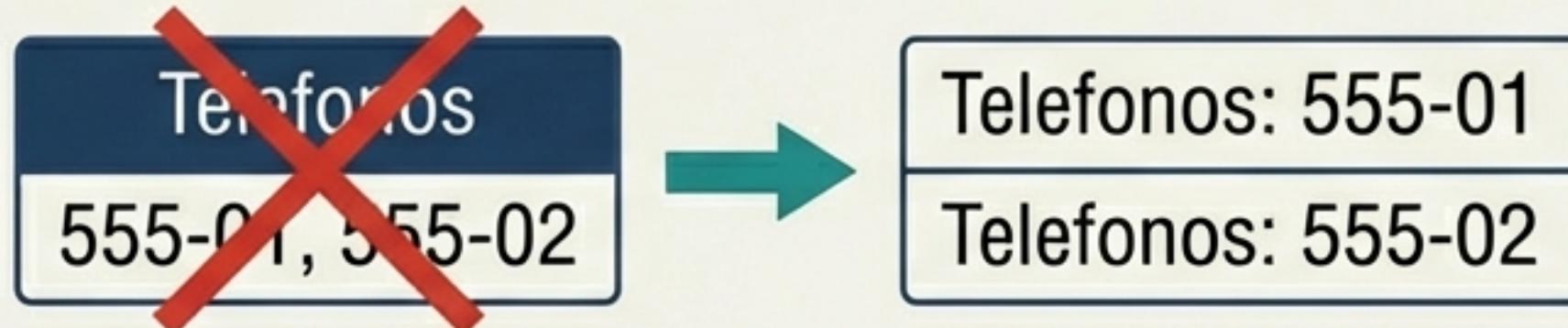
Transitiva



Dependencia indirecta.

El Estándar de Oro: FN1, FN2 y FN3

✓ 1^a Forma Normal (FN1) - Atomicidad



Rule

Cada celda debe contener un solo valor. No grupos repetidos.

✓ 2^a Forma Normal (FN2) - Dependencia Completa

Rule

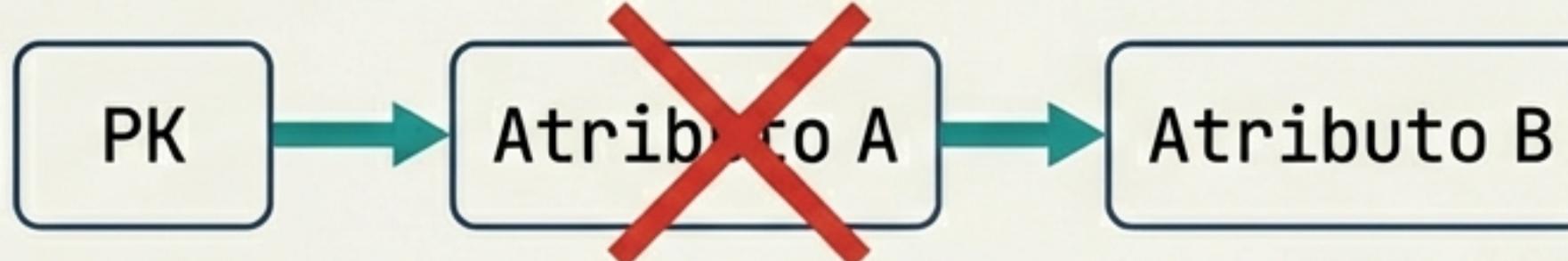
Eliminar dependencias parciales.
Todo atributo debe depender de *toda* la clave primaria.



✓ 3^a Forma Normal (FN3) - Sin Transitividad

Rule

Los atributos no clave no pueden depender de otros atributos no clave.



Forma Normal de Boyce-Codd (FNBC)

Una versión más estricta de la 3^a Forma Normal.

Todo determinante debe ser una clave candidata.

✗ The Failure

ASIGNACIÓN

	Empleado	Departamento	Responsable
1	Antonio	RR.HH.	Ana
2	Maria	RR.HH.	Ana



Ana (Responsable) determina RR.HH. (Dpto), pero Ana NO es clave única (podría llevar otros).

✓ The Fix

PERSONAL

Empleado

Departamento

DEPARTAMENTO

Departamento

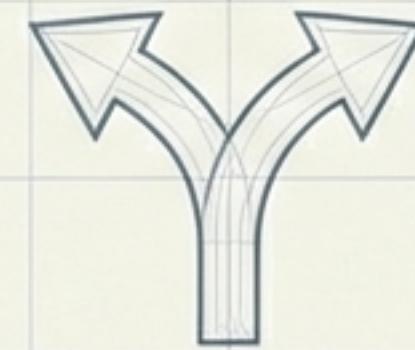
Responsable



Ahora los determinantes son claves en sus propias tablas.

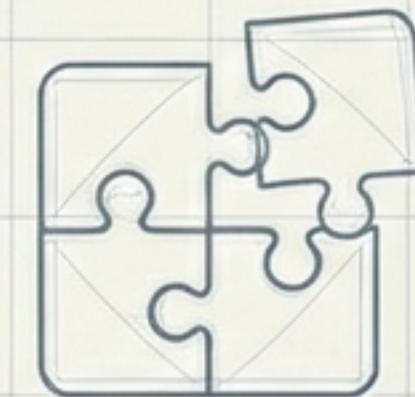
Teoría Avanzada

FN4, FN5 y Dominio-Clave (Aplicación Teórica)



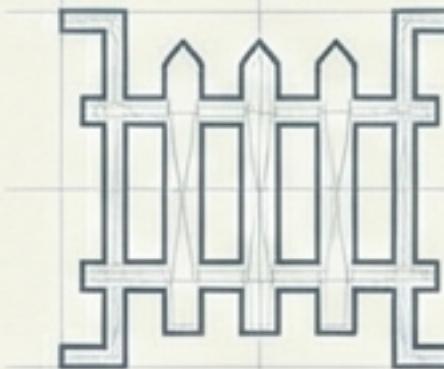
4^a Forma Normal (FN4)

Elimina dependencias multivaluadas independientes dentro de una misma clave.



5^a Forma Normal (FN5)

Dependencias de Join. Garantiza que dividir y volver a unir tablas no pierda información.



Dominio-Clave (DKNF)

La más estricta. Basada puramente en restricciones de dominio y claves.

Inter

Source Sans Pro

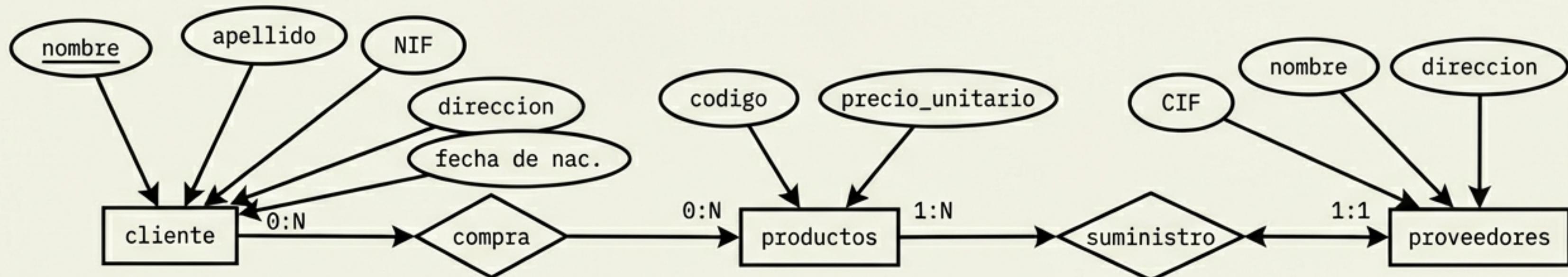
La Desnormalización: Rompiendo las Reglas



Ventajas	Riesgos
<ul style="list-style-type: none">✓ Consultas más rápidas.✓ Menos complejidad en SQL.	<ul style="list-style-type: none">⚠ Inconsistencia de datos.⚠ Mayor almacenamiento.⚠ Updates más lentos.

Uso estratégico: Tablas de resumen o históricos.

Caso Práctico Final: Tienda de Informática



Resultado: Esquema Relacional

1. CLIENTE (NIF, Nombre, Apellido, Dirección, Fecha_Nacimiento)
[PK: NIF]
2. COMPRA (NIF_Cliente, Código_Producto, Fecha)
[PK: Compuesta]
3. PRODUCTOS (Código, Nombre, Precio_Unitario, CIF_Proveedor)
[FK: CIF_Proveedor]
4. PROVEEDORES (CIF, Nombre, Dirección)
[PK: CIF]

Listo para implementación en SQL.