

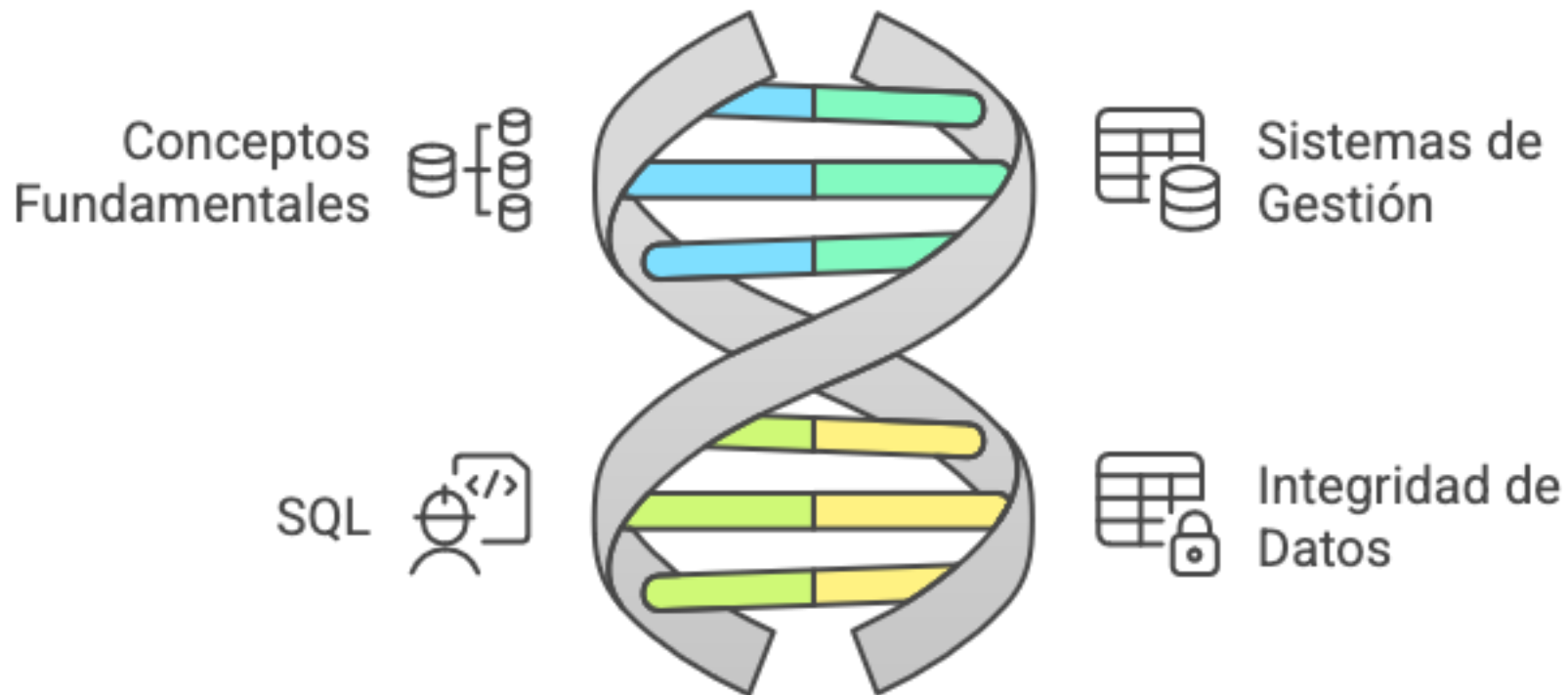
Base de Datos 1

Unidad 3

Modelo Relacional



Fundamentos y Aplicaciones del Modelo Relacional en Bases de Datos



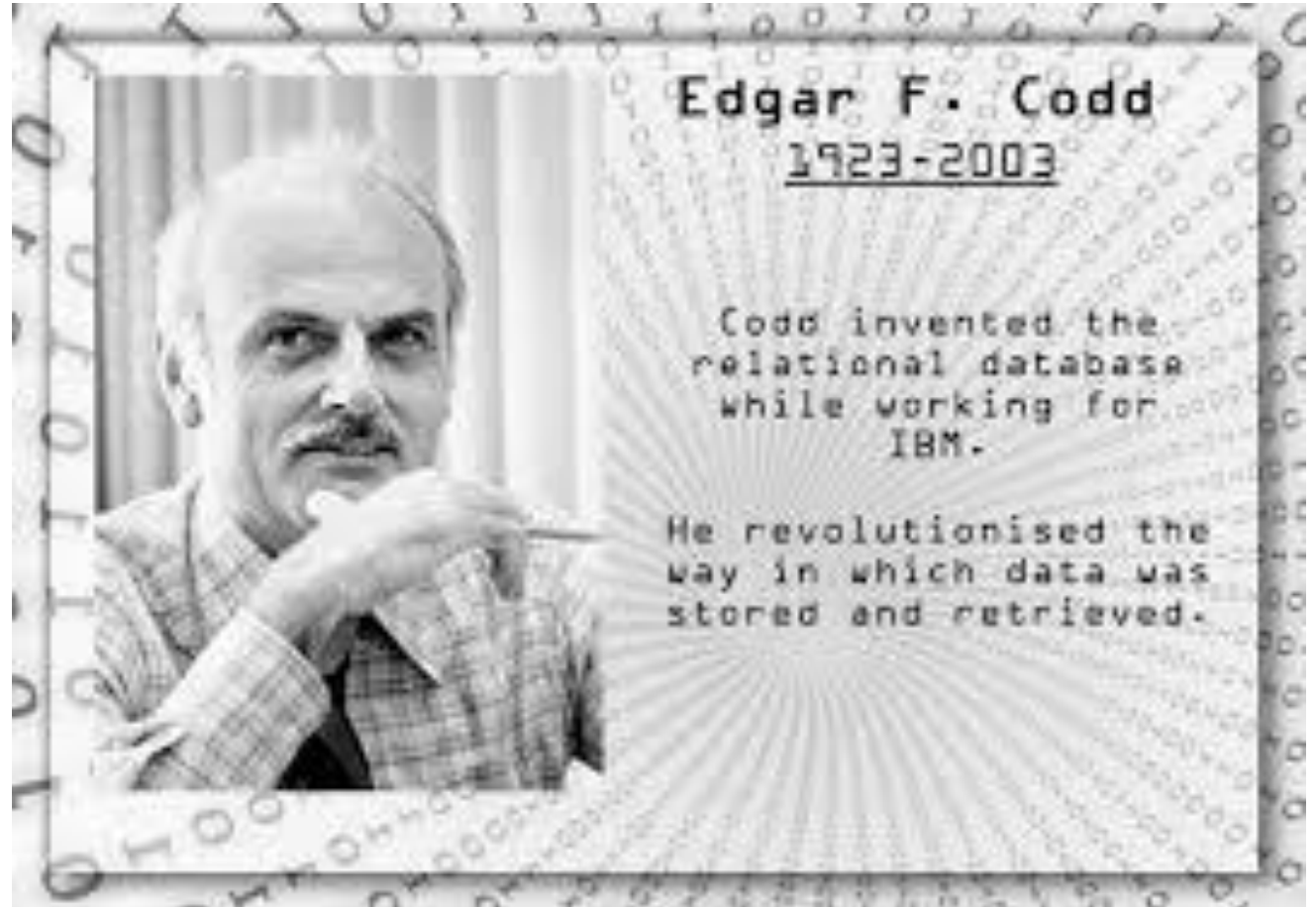


El modelo relacional

- Es un modelo de datos y, como tal, tiene en cuenta los tres aspectos siguientes de los datos:
 - La estructura, que debe permitir representar la información que nos interesa del mundo real.
 - La manipulación, a la que da apoyo mediante las operaciones de actualización y consulta de los datos.
 - La integridad, que es facilitada mediante el establecimiento de reglas de integridad; es decir, condiciones que los datos deben cumplir.
-

Principios

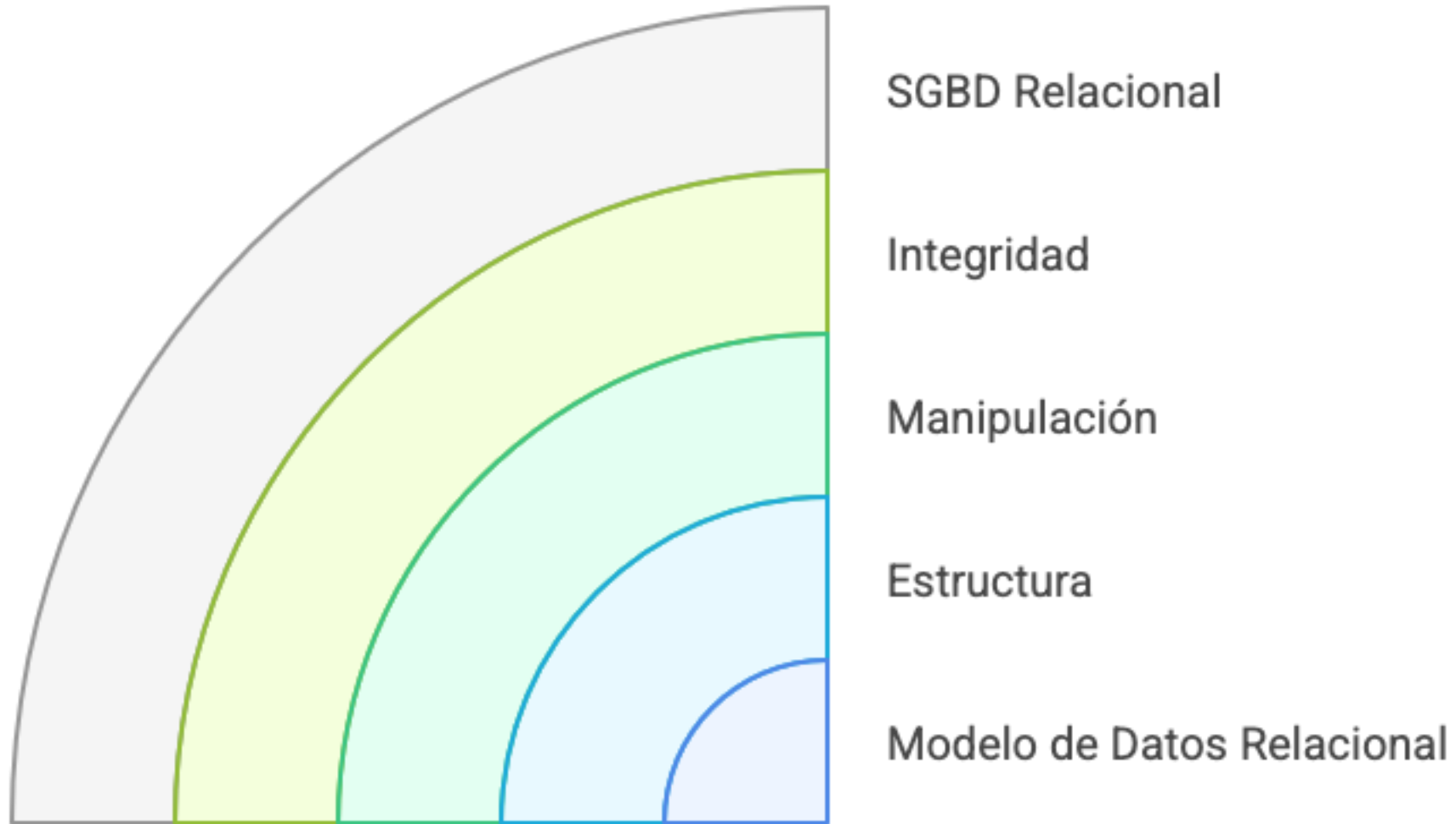
- Fueron establecidos por E.F. Codd en los años 1969 y 1970.
- “Edgar Frank "Ted" Codd fue un científico informático inglés (19 de agosto de 1923 - 18 de abril de 2003), conocido por crear el modelo relacional de bases de datos.”
 - Fuente: Wikipedia



Principios

- Hasta la década de los ochenta no se empezaron a comercializar los primeros SGBD relacionales con rendimientos aceptables.
- El principal objetivo es facilitar que la base de datos sea percibida o vista por el usuario como una estructura lógica que consiste en un conjunto de relaciones y no como una estructura física de implementación.
- Esta estructura lógica con la que se percibe la base de datos debe ser simple y uniforme.
- Toda la información se representa de una única manera: mediante valores explícitos que contienen las relaciones.
- Todos los valores de datos se consideran atómicos; es decir, no es posible descomponerlos.

Modelo de Datos Relacional



Comprendiendo la Estructura de Datos Relacional



Estructura de los datos

- El modelo relacional proporciona una estructura de los datos que consiste en un conjunto de relaciones con objeto de representar la información que nos interesa del mundo real.
- La estructura de los datos del modelo relacional se basa, pues, en el concepto de relación.

DER

Socio

Socio = @Cod_socio + Nombre + Apellido + Domicilio + Teléfono + Tipo

MR (Modelo Relacional)

SOCIO

<i>Cod_socio</i>	<i>Nombre</i>	<i>Apellido</i>	<i>Domicilio</i>	<i>Teléfono</i>	<i>Tipo</i>
00001	Susana	Hidalgo	Rios Ros 22	4138060	I
00002	Adolfo	Sánchez	San Ben 44	4131419	P
00003	Antonio	Martin	Ppe. Ver. 66	4139865	A
00004	Miguel	García	De María 60	7567676	I
⋮	⋮	⋮	⋮	⋮	⋮
03456	Elena	Martin	Goya 445	9192919	I

Clave Primaria

Modelo Relacional

Comprendiendo el Modelo Relacional

- VISION INFORMAL



Visión informal de una relación

- Se puede obtener una buena idea intuitiva de lo que es una relación si la visualizamos como una tabla o un fichero.
- En la figura se muestra la visualización tabular de una relación que contiene datos de empleados.
- Cada fila de la tabla contiene una colección de valores de datos relacionados entre sí; en nuestro ejemplo, son los datos correspondientes a un mismo empleado.
- La tabla tiene un nombre (EMPLEADOS) y también tiene un nombre cada una de sus columnas (DNI, nombre, apellido y sueldo).
- El nombre de la tabla (relación) y los de las columnas ayudan a entender el significado de los valores que contiene la tabla. Cada columna contiene valores de un cierto dominio; por ejemplo, la columna DNI contiene valores del dominio númerosDNI.

Conjunto de relaciones
Una base de datos relacional consta de un conjunto de relaciones, cada una de las cuales se puede visualizar de este modo tan sencillo.
La estructura de los datos del modelo relacional resulta fácil de entender para el usuario.

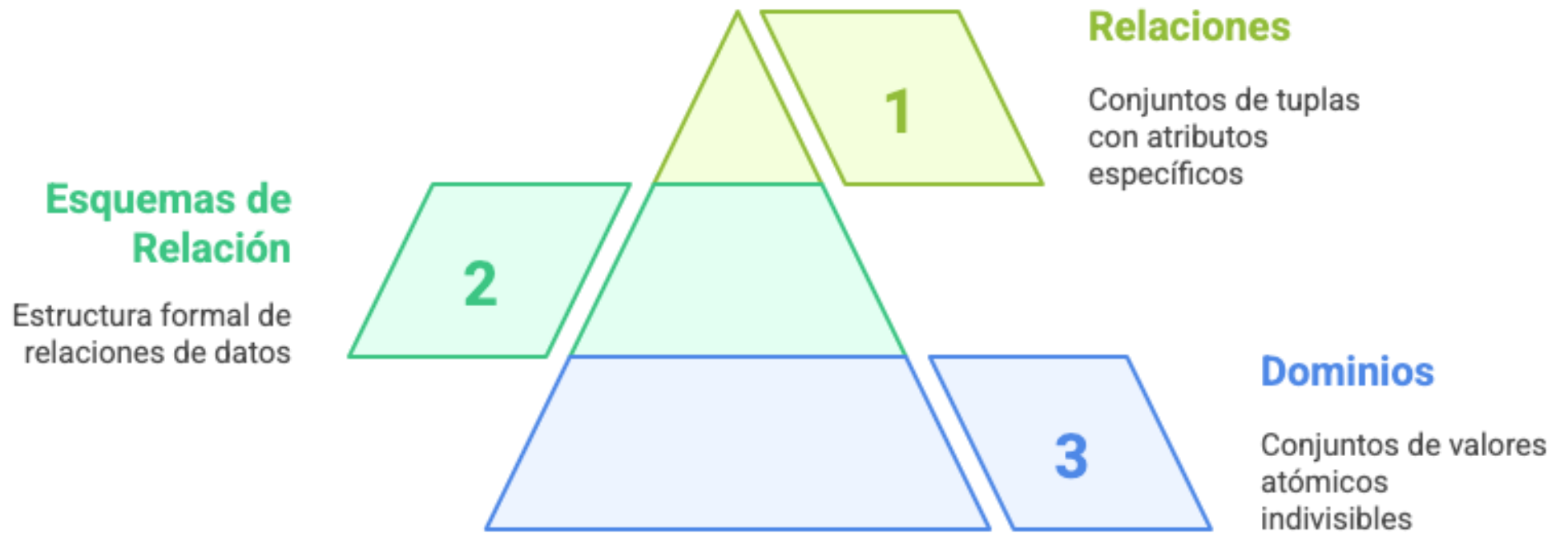
Relación EMPLEADOS

númerosDNI	nombres	apellidos	sueños
DNI	nombre	apellido	suelo
40.444.255	Juan	García	2.000
33.567.711	Marta	Roca	2.500
55.898.425	Carlos	Buendía	1.500

Modelo Relacional

- VISION FORMAL

Jerarquía de Estructura de Datos Relacionales



Visión formal de una relación - Dominio

- “Dominios. Un dominio describe un conjunto de posibles valores para cierto atributo. Como un dominio restringe los valores del atributo, puede ser considerado como una restricción.
- Matemáticamente, atribuir un dominio a un atributo significa cualquier valor de este atributo debe ser elemento del conjunto especificado”.

Visión formal de una relación - Dominio

- Dos tipos:
 - Dominios predefinidos, que corresponde a los tipos de datos que normalmente proporcionan los lenguajes de bases de datos, como por ejemplo los enteros, las cadenas de caracteres, los reales, etc.
 - Dominios definidos por el usuario, que pueden ser más específicos. Toda definición de un dominio debe constar, como mínimo, del nombre del dominio y de la descripción de los valores que forman parte de éste.

Dominio definido por el usuario

Por ejemplo, el usuario puede definir un dominio para las edades de los empleados que se denomine dom_edad y que contenga los valores enteros que están entre 16 y 65.

Visión formal de una relación - Esquema

- Una relación se compone del esquema (o intensidad de la relación) y de la extensión.

EMPLEADOS				Esquema
<i>DNI</i>	<i>nombre</i>	<i>apellido</i>	<i>suelo</i>	
40.444.255	Juan	García	2.000	Extensión
33.567.711	Marta	Roca	2.500	
55.898.425	Carlos	Buendía	1.500	

- El esquema de la relación consiste en un nombre de relación R y un conjunto de atributos {A1, A2, ..., An}.

Nombre y conjunto de atributos de la relación empleados

EMPLEADOS			
DNI	nombre	apellido	sueldo
40.444.255	Juan	García	2.000
33.567.711	Marta	Roca	2.500
55.898.425	Carlos	Buendía	1.500

Esquema

Extensión

Si tomamos como ejemplo la figura, el nombre de la relación es EMPLEADOS y el conjunto de atributos es:


{DNI, nombre, apellido, sueldo}



Tomaremos la convención de denotar el esquema de la relación de la forma siguiente:

$R(A_1, A_2, \dots, A_n)$

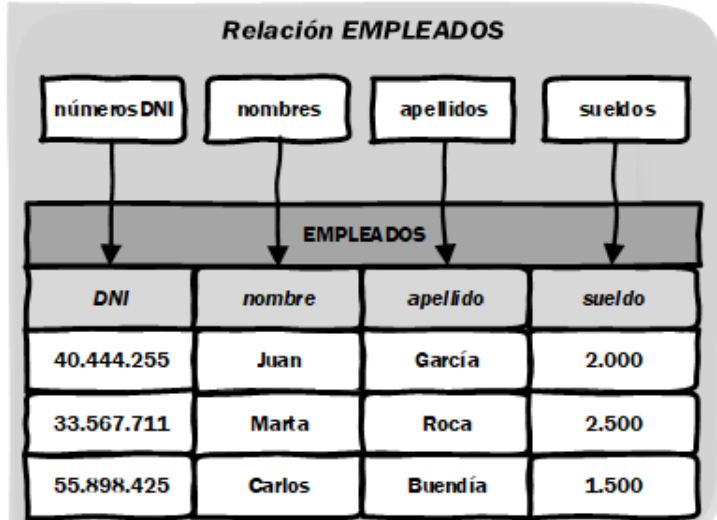
Donde R es el nombre la relación y A_1, A_2, \dots, A_n es una ordenación cualquiera de los atributos que pertenecen al conjunto $\{A_1, A_2, \dots, A_n\}$.



Denotación del esquema de la relación empleados

- El esquema de la relación de la figura se podría denotar, por ejemplo, como *EMPLEADOS(DNI, nombre, apellido, sueldo)*, o también, *EMPLEADOS(nombre, apellido, DNI, sueldo)*, porque cualquier ordenamiento de sus atributos se considera válida para denotar el esquema de una relación.
 - Un atributo A_i es el nombre del papel que ejerce un dominio D en un esquema de relación. D es el dominio de A_i y se denota como **dominio(A_i)**.
-

Dominio del atributo DNI



- Según la figura, el atributo DNI corresponde al papel que ejerce el dominio númerosDNI en el esquema de la relación EMPLEADOS y, entonces, $\text{dominio}(\text{DNI}) = \text{númerosDNI}$.
- Conviene observar que cada atributo es único en un esquema de relación, porque no tiene sentido que un mismo dominio ejerza dos veces el mismo papel en un mismo esquema. Por consiguiente, no puede ocurrir que en un esquema de relación haya dos atributos con el mismo nombre.
- En cambio, sí que se puede repetir un nombre de atributo en relaciones diferentes. Los dominios de los atributos, por el contrario, no deben ser necesariamente todos diferentes en una relación.

Ejemplo de atributos diferentes con el mismo dominio

- Si tomamos como ejemplo el esquema de relación **PERSONAS(DNI, nombre, apellido, telcasa, teltrabajo)**, los atributos telcasa y teltrabajo pueden tener el mismo dominio: *dominio(telcasa) = teléfono* y *dominio(teltrabajo) = teléfono*.
 - En este caso, el dominio teléfono ejerce dos papeles diferentes en el esquema de relación: el de indicar el teléfono particular de una persona y el de indicar el del trabajo.

Visión formal de una relación - Extensión

Número-cuenta	Nombre-sucursal	saldo
C-100	Centro	500
C-101	Flores	400
C-102	Morón	900
C-201	San Isidro	250
C-215	San Pedro	111
C-217	Mar del Plata	500
C-222	Azul	222
C-305	Tandil	500



Visión formal de una relación - Extensión

La extensión de la relación de esquema $R(A_1, A_2, \dots, A_n)$ es un conjunto de tuplas t_i ($i = 1, 2, \dots, m$), donde cada tupla t_i es, a su vez un conjunto de pares $t_i = \{ \langle A_1:vi_1 \rangle, \langle A_2:vi_2 \rangle \dots \langle A_n:vi_n \rangle \}$ y, para cada par $\langle A_j:v_{ij} \rangle$, se cumple que v_{ij} es un valor de $\text{dominio}(A_j)$, o bien un valor especial que denominaremos nulo.

Para simplificar, tomaremos la convención de referirnos a una tupla $t_i = \{ \langle A_1:vi_1 \rangle, \langle A_2:vi_2 \rangle, \dots, \langle A_n:vi_n \rangle \}$ que pertenece a la extensión del esquema denotado como $R(A_1, A_2, \dots, A_n)$, de la forma siguiente: $t_i = \langle vi_1, vi_2, \dots, vi_n \rangle$

Visión formal de una relación - Extensión

Extensión de la relación de esquema EMPLEADOS(*DNI, nombre, apellido, sueldo*)

<40.444.255, Juan, García, 2.000>

<33.567.711, Marta, Roca, 2.500>

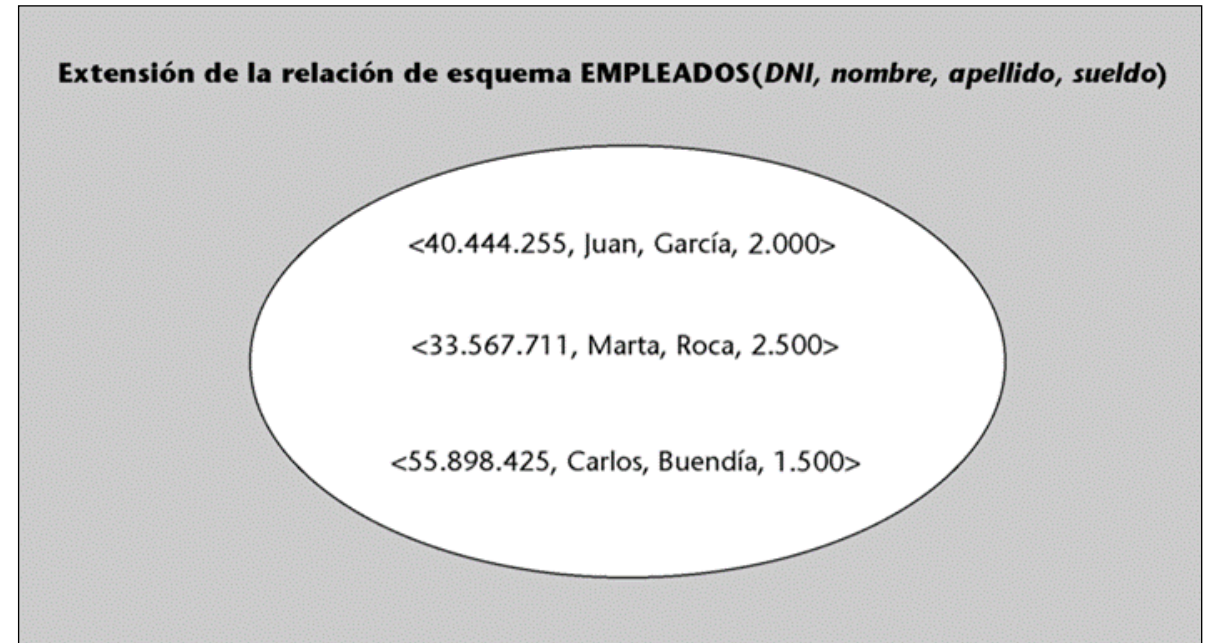
<55.898.425, Carlos, Buendía, 1.500>

Ejemplo de valor nulo

- Podríamos tener un atributo telcasa en la relación EMPLEADOS y se podría dar el caso de que un empleado no tuviese teléfono en su casa, o bien que lo tuviese, pero no se conociese su número. En las dos situaciones, el valor del atributo telcasa para la tupla correspondiente al empleado sería el valor nulo

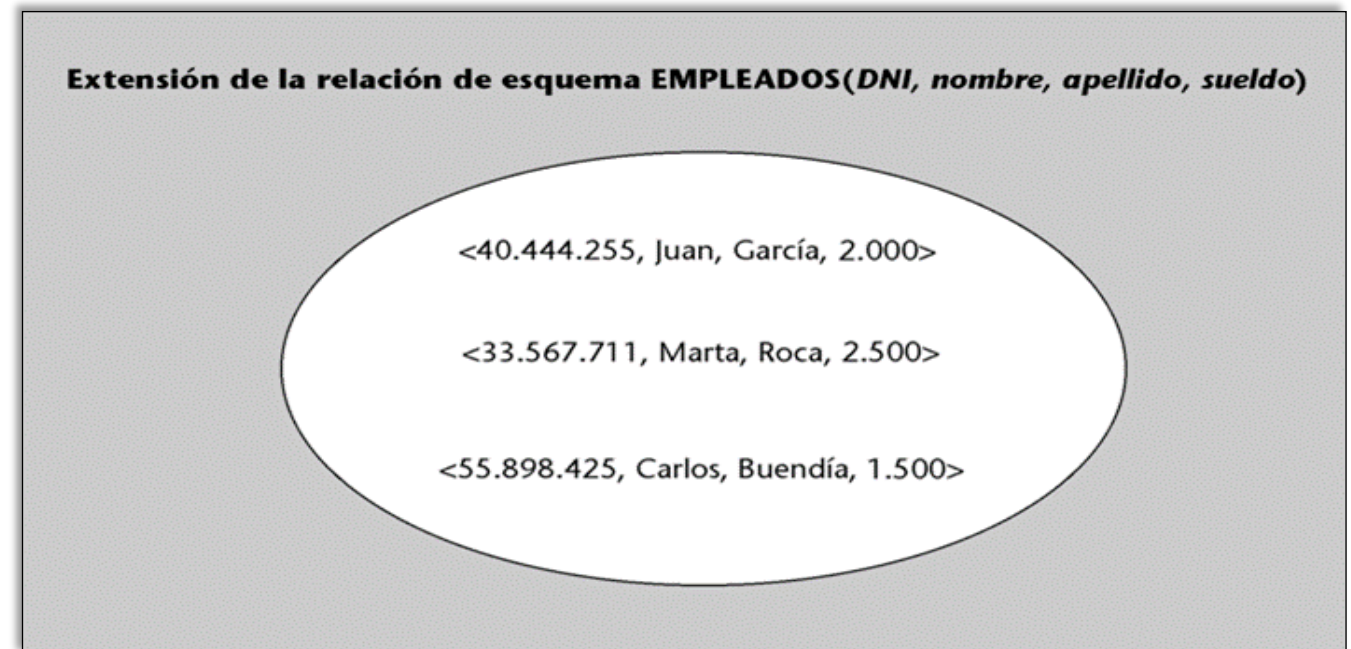
Grado de una relación

- El grado de una relación es el número de atributos que pertenecen a su esquema
- El grado de la relación de esquema **EMPLEADOS(DNI, nombre, apellido, sueldo)**, es 4.



Cardinalidad de una relación

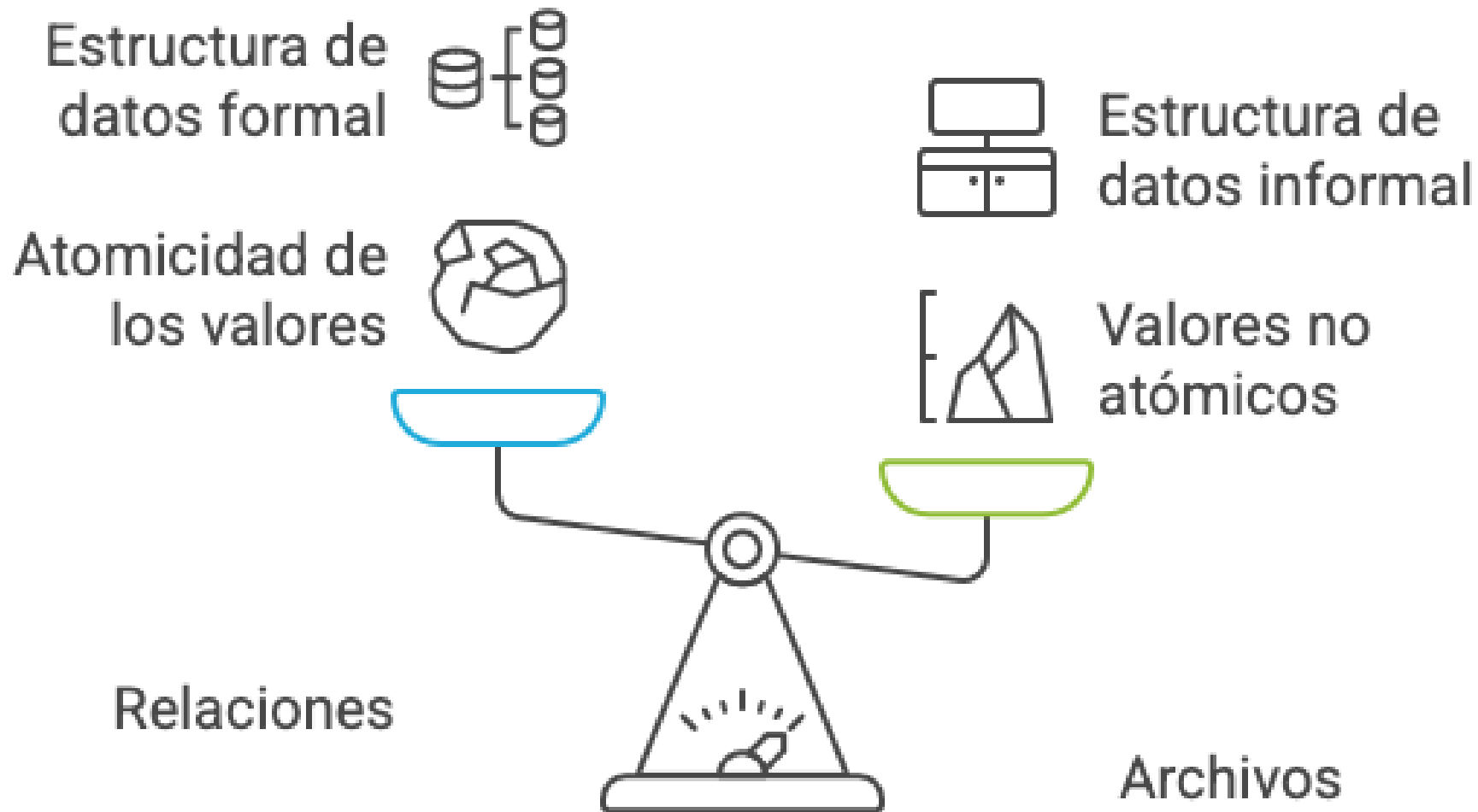
- La cardinalidad de una relación es el número de tuplas que pertenecen a su extensión.
- Observando la figura se deduce que la cardinalidad de la relación EMPLEADOS es 3.



Diferencia entre archivos y relaciones

- Atomicidad de los valores de los atributos
- No-repetición de las tuplas





Comparando la estructura de datos formal e informal

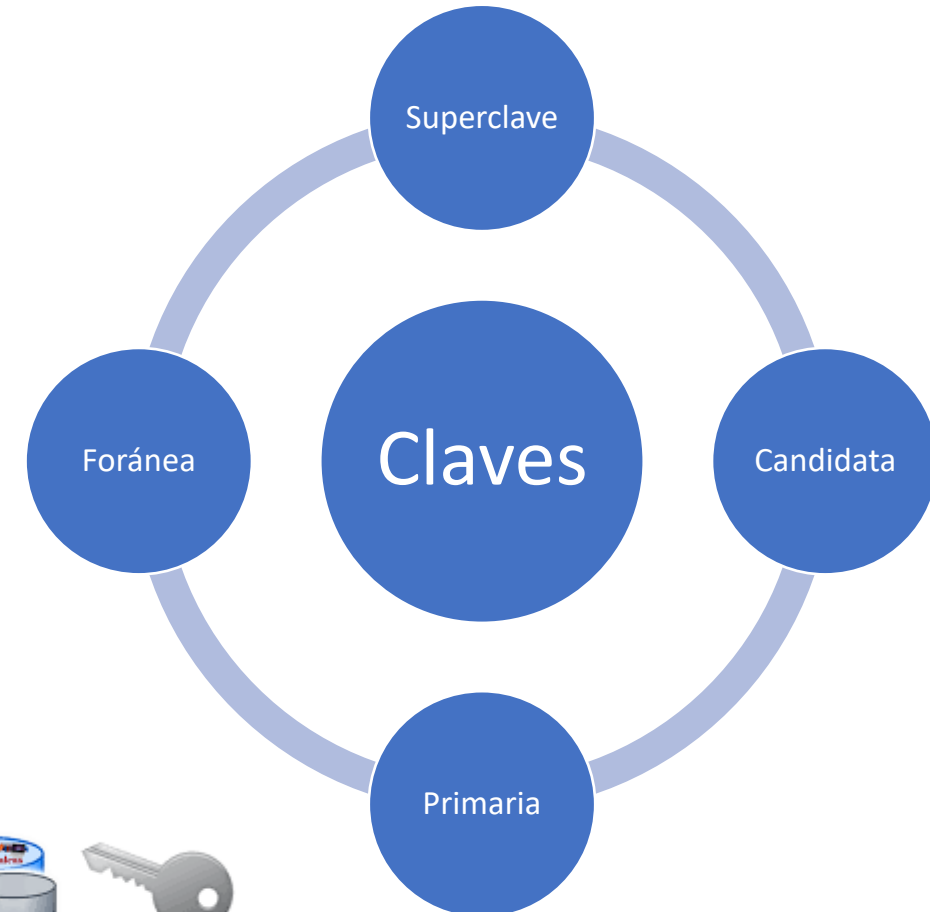
Diferencia entre archivos y relaciones

A thick yellow horizontal bar spans the width of the slide, with a vertical yellow bar extending downwards from its right end.

- No-ordenación de las tuplas
 - No-ordenación de los atributos
- 
- A thin grey horizontal bar spans the width of the slide at the bottom.

CLAVE CANDIDATA, CLAVE PRIMARIA, ALTERNATIVA Y CLAVE FORANEA

- Toda la información que contiene una base de datos se debe poder identificar de alguna forma.
- Para identificar los datos que la base de datos contiene, se pueden utilizar las claves *candidatas* de las relaciones.
- A continuación, definiremos qué se entiende por clave *candidata*, clave *primaria* y clave *alternativa* de una relación. Para hacerlo, será necesario definir el concepto de *superclave*





Superclave

- “Una superclave es un conjunto de uno o más atributos que, tomados colectivamente, permiten identificar de forma única una entidad en el conjunto de entidades. Por ejemplo, el atributo id-cliente del conjunto de entidades cliente es suficiente para distinguir una entidad cliente de las otras.”
 - Una superclave de una relación de esquema $R(A_1, A_2, \dots, A_n)$ es un subconjunto de los atributos del esquema tal que no puede haber dos tuplas en la extensión de la relación que tengan la misma combinación de valores para los atributos del subconjunto
-



Superclave

- Una superclave, por lo tanto, nos permite identificar todas las tuplas que contiene la relación.
 - Toda relación tiene, por lo menos, una superclave, que es la formada por todos los atributos de su esquema
 - En la relación de esquema EMPLEADOS(DNI, CUIL, nombre, apellido, teléfono), algunas de las superclaves de la relación serían los siguientes subconjuntos de atributos: {DNI, CUIL, nombre, apellido, teléfono}, {DNI, apellido}, {DNI} y {CUIL}.
-



Superclave

- Empleado {TipoDoc, NroDoc, Cuit, Nombre, Edad}
 - Superclaves Posibles
 - TipoDoc, NroDoc, CUIT, Nombre, Edad
 - TipoDoc, NroDoc, CUIT, Nombre
 - TipoDoc, NroDoc, Edad
 - TipoDoc, NroDoc, CUIT
 - CUIT
 - CUIT, EDAD
 - ...
-



Clave candidata

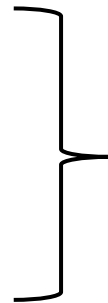
- Una clave candidata de una relación es una superclave C de la relación que cumple que ningún subconjunto propio de C es superclave
 - Es decir, C cumple que la eliminación de cualquiera de sus atributos da un conjunto de atributos que no es superclave de la relación.
 - Intuitivamente, una clave candidata permite identificar cualquier tupla de una relación, de manera que no sobre ningún atributo para hacer la identificación
 - Toda relación tiene por lo menos una superclave, por lo tanto, toda relación tiene como mínimo una clave candidata
 - En la relación de esquema EMPLEADOS(DNI, CUIL, nombre, apellido, teléfono), sólo hay dos claves candidatas: {DNI} y {CUIL}.
-

Clave Candidata

- Empleado {TipoDoc, NroDoc, Cuit, Nombre, Edad}
- Claves Candidatas Posibles

- TipoDoc, NroDoc

- CUIT



No tiene Subconjuntos que son SuperClaves

Clave Primaria

- “Clave primaria o clave principal a un campo o a una combinación de campos que identifica de forma única a cada fila de una tabla. Una clave primaria comprende de esta manera una columna o conjunto de columnas. No puede haber dos filas en una tabla que tengan la misma clave primaria.”



Clave Primaria

- Habitualmente, una de las claves candidatas de una relación se designa clave primaria de la relación. La clave primaria es la clave candidata cuyos valores se utilizarán para identificar las tuplas de la relación
 - El diseñador de la base de datos es quien elige la clave primaria de entre las claves candidatas
 - Si una relación sólo tiene una clave candidata, entonces esta clave candidata debe ser también su clave primaria.
 - Ya que todas las relaciones tienen como mínimo una clave candidata, podemos garantizar que, para toda relación, será posible designar una clave primaria
-

Clave Primaria


- Empleado {TipoDoc, NroDoc, Cuit, Nombre, Edad}
- Claves Primaria Elegida
 - TipoDoc, NroDoc

Clave Alternativa

- Las claves candidatas no elegidas como primaria se denominan claves alternativas
 - Utilizaremos la convención de subrayar los atributos que forman parte de la clave primaria en el esquema de la relación. Así pues, $R(A_1, A_2, \dots, \underline{A_i}, \dots, A_n)$ indica que los atributos A_1, A_2, \dots, A_i forman la clave primaria de R .
-

Clave Alternativa

- Empleado {TipoDoc, NroDoc, Cuit, Nombre, Edad}
- Claves Alternativa
 - CUIT



ELECCIÓN DE LA CLAVE PRIMARIA DE EMPLEADOS

- En la relación de esquema EMPLEADOS(DNI, CUIL, nombre, apellido, teléfono), donde hay dos claves candidatas, {DNI} y {CUIL}, se puede elegir como clave primaria {DNI}. Lo indicaremos subrayando el atributo DNI en el esquema de la relación EMPLEADOS(DNI, CUIL, nombre, apellido, teléfono). En este caso, la clave {CUIL} será una clave alternativa de EMPLEADOS.



CLAVE PRIMARIA DE LA RELACIÓN OFICINAS

- En la relación de esquema OFICINAS(edificio, número, superficie), la clave primaria está formada por los atributos edificio y número.
- En este caso, podrá ocurrir que dos despachos diferentes estén en el mismo edificio, o bien que tengan el mismo número, pero nunca pasará que tengan la misma combinación de valores para edificio y número.

Garantizar la integridad de la clave foránea





CLAVES FORÁNEAS DE LAS RELACIONES

- “Una clave foránea o clave ajena (o Foreign Key FK) es una limitación referencial entre dos tablas. La clave foránea identifica una columna o grupo de columnas en una tabla (tabla hija o referendo) que se refiere a una columna o grupo de columnas en otra tabla (tabla maestra o referenciada). Las columnas en la tabla referendo deben ser la clave primaria u otra clave candidata en la tabla referenciada.”
-

CLAVES FORANEAS

Es un atributo o combinación de atributos en una Relación $r(R)$ que es clave primaria de una relación $s(S)$.

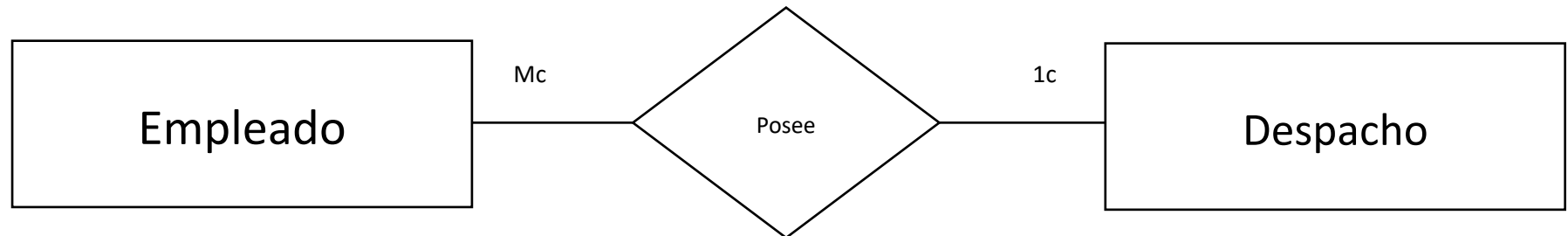
EMPLEADO	NOMBREP	INIC	APELLIDO	<u>NSS</u>	FECHAN	DIRECCIÓN	SEXO	SALARIO	NSSSUPER	ND
	José	B	Silva	123456789	09-ENE-55	Fresnos 731, Higuera, MX	M	30000	333445555	5

TRABAJA_EN	<u>NSSE</u>	<u>NÚMP</u>	HORAS
	123456789	1	32.5
	123456789	2	7.5

PROYECTO	NOMBREP	<u>NÚMEROP</u>	LUGARP	NÚMD
	ProductoX	1	Belén	5
	ProductoY	2	Sacramento	5

CLAVES FORÁNEAS DE LAS RELACIONES

- En otro ejemplo, una base de datos que tiene dos relaciones: una denominada EMPLEADOS, que almacenaría datos de los empleados de una empresa, y otra con el nombre DESPACHOS, que almacenaría los datos de los despachos que tiene la empresa entre los distintos hechos que se dan en el mundo real pueden existir lazos o vínculos los empleados que trabajan para una empresa pueden estar vinculados con los despachos de la empresa, porque a cada empleado se le asigna un despacho concreto para trabajar





CLAVES FORÁNEAS DE LAS RELACIONES

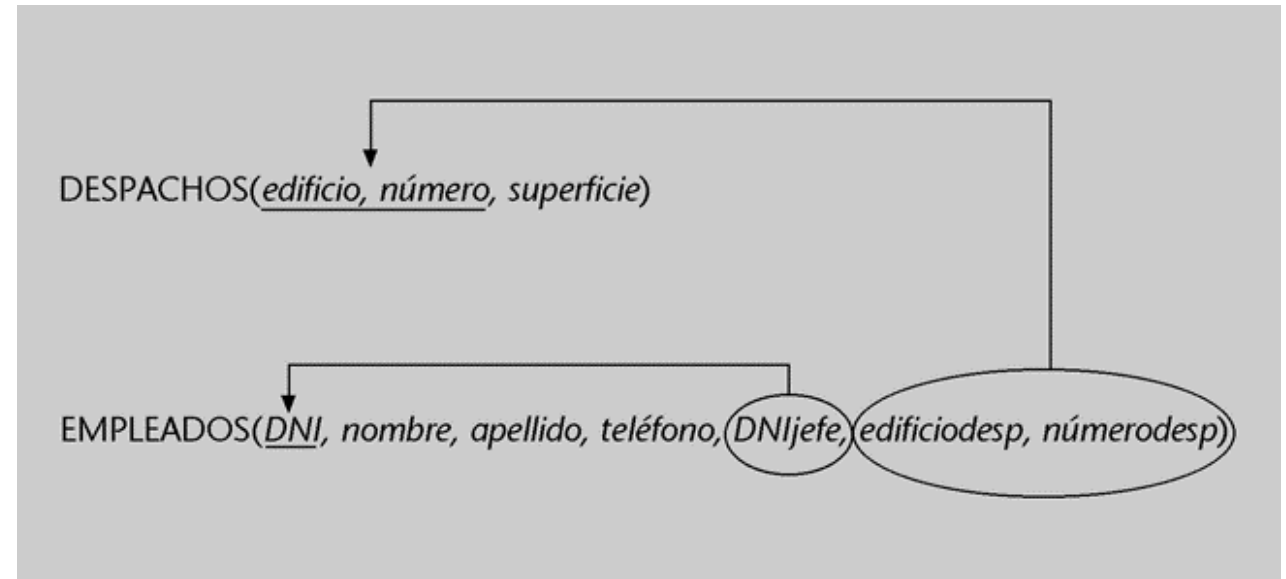
- En el modelo relacional tenemos la posibilidad de expresar conexiones entre las distintas tuplas de las relaciones.
 - Por ejemplo, en la base de datos anterior, que tiene las relaciones EMPLEADOS y DESPACHOS, puede ser necesario conectar tuplas de EMPLEADOS con tuplas de DESPACHOS para indicar qué despacho tiene asignado cada empleado.
 - En ocasiones, puede ser necesario reflejar lazos entre tuplas que pertenecen a una misma relación. Por ejemplo, en la misma base de datos anterior puede ser necesario conectar determinadas tuplas de EMPLEADOS con otras tuplas de EMPLEADOS para indicar, para cada empleado, quién actúa como su jefe.
-

CLAVES FORÁNEAS DE LAS RELACIONES

- El mecanismo que proporcionan las bases de datos relacionales para conectar tuplas son las claves foráneas de las relaciones. Las claves foráneas permiten establecer conexiones entre las tuplas de las relaciones. Para hacer la conexión, una clave foránea tiene el conjunto de atributos de una relación que referencian la clave primaria de otra relación (o incluso de la misma relación).

CLAVES FORÁNEAS DE LAS RELACIONES


- En la figura siguiente, la relación *EMPLEADOS*(*DNI*, *nombre*, *apellido*, *teléfono*, *DNIjefe*, *edificiodesp*, *númerodesp*), tiene una clave foránea formada por los atributos *edificiodesp* y *númerodesp* que se refiere a la clave primaria de la relación *DESPACHOS*(*edificio*, *número*, *superficie*). Esta clave foránea indica, para cada empleado, el despacho donde trabaja. Además, el atributo *DNIjefe* es otra clave foránea que referencia la clave primaria de la misma relación *EMPLEADOS*, e indica, para cada empleado, quien es su jefe.





CLAVES FORÁNEAS DE LAS RELACIONES

- Las claves foráneas tienen por objetivo establecer una conexión con la clave primaria que referencian. Por lo tanto, los valores de una clave foránea deben estar presentes en la clave primaria correspondiente, o bien deben ser valores nulos. En caso contrario, la clave foránea representaría una referencia o conexión incorrecta
-



Claves foráneas – Definición formal

- Una clave foránea de una relación R es un subconjunto de atributos del esquema de la relación, que denominamos CF y que cumple las siguientes condiciones:
 - Existe una relación S (S no debe ser necesariamente diferente de R) que tiene por clave primaria CP. Se cumple que, para toda tupla t de la extensión de R, los valores para CF de t son valores nulos o bien valores que coinciden con los valores para CP de alguna tupla s de S.
 - Y entonces, se dice que la clave foránea CF referencia la clave primaria CP de la relación S, y también que la clave foránea CF referencia la relación S
-

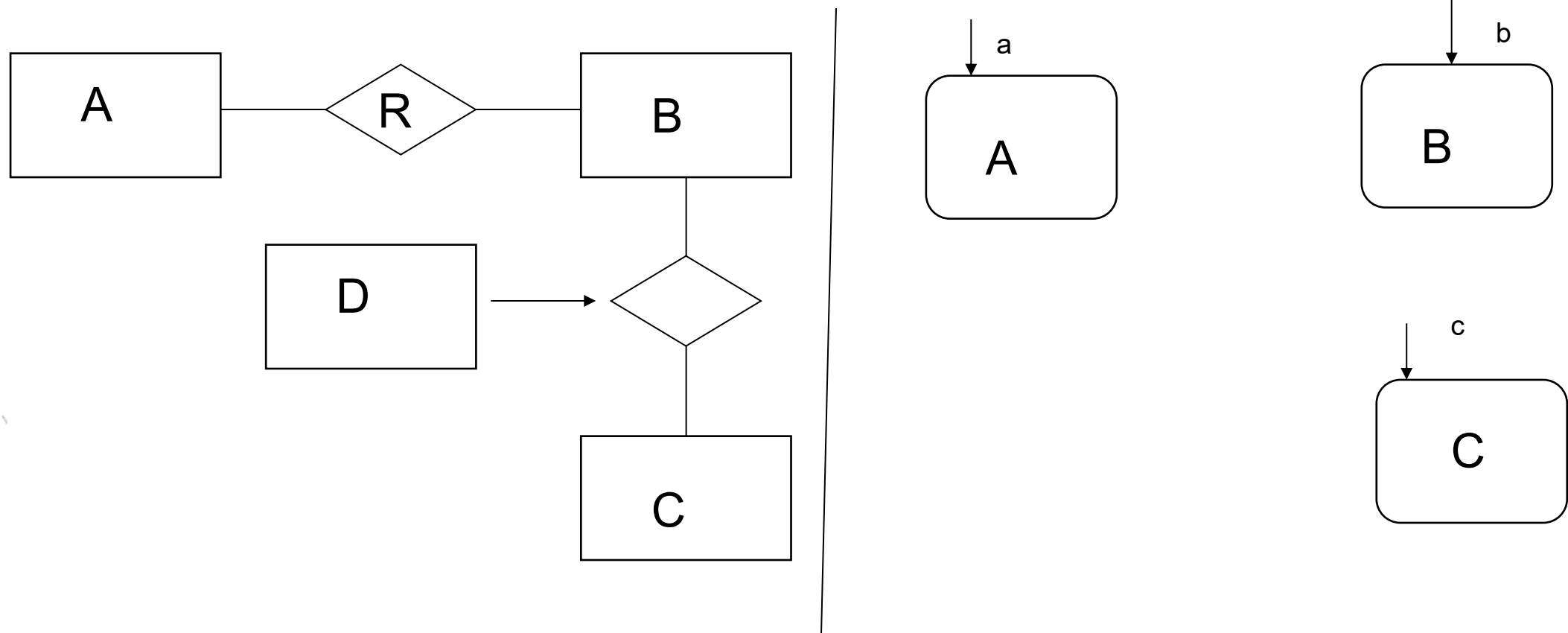


Creación de las relaciones de una base de datos

- Hemos visto que una base de datos relacional consta de varias relaciones. Cada relación tiene varios atributos que toman valores de unos ciertos dominios; también tiene una clave primaria y puede tener una o más claves foráneas. Los lenguajes de los SGBD relacionales deben proporcionar la forma de definir todos estos elementos para crear una base de datos.
 - Más adelante se verá con detalle la sintaxis y el significado de las sentencias de definición de la base de datos para el caso concreto del lenguaje SQL.
-

Conversiones del Modelo DER al modelo relacional.

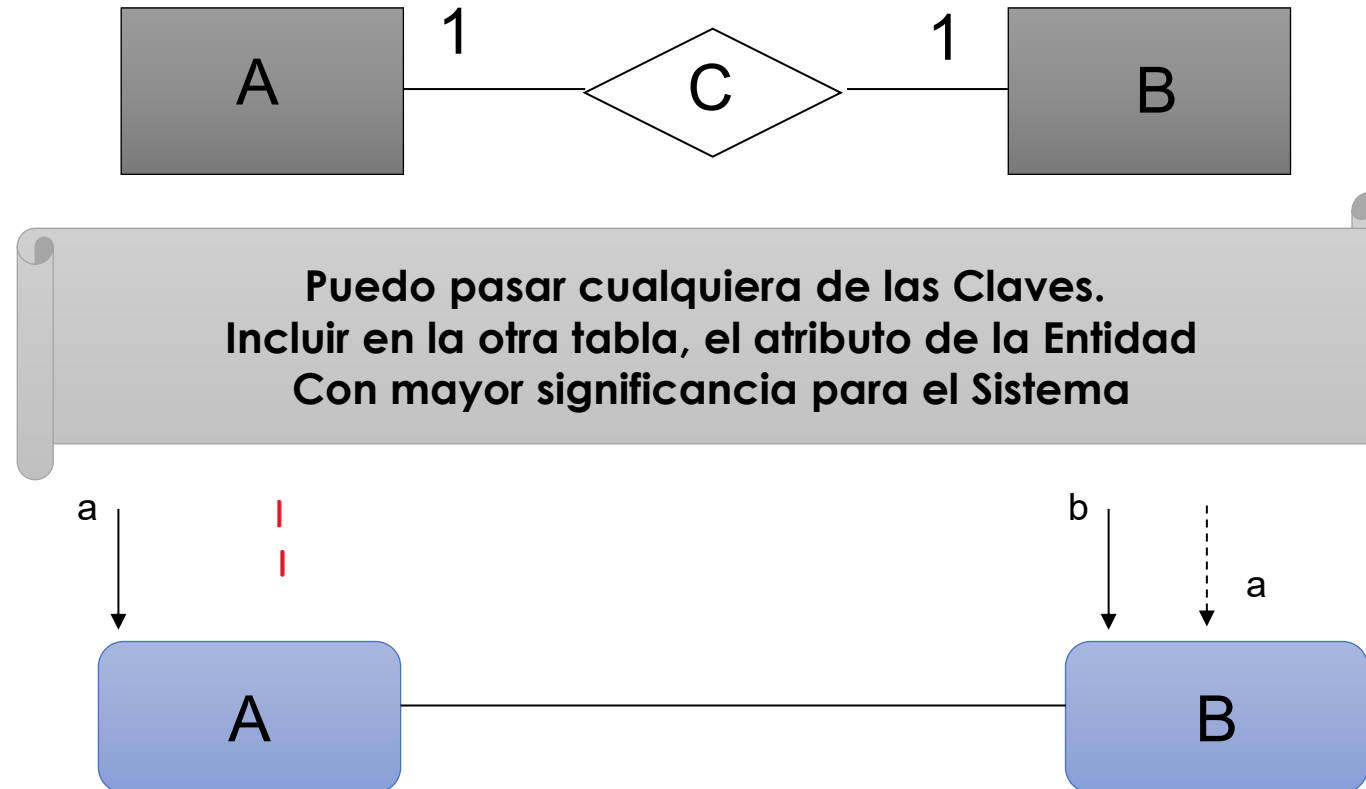
- 1) Cada Objeto del DER o entidad es una Relación
- 2) Defino cual es la clave Primaria de cada Objeto.



Conversiones del Modelo DER al modelo relacional.

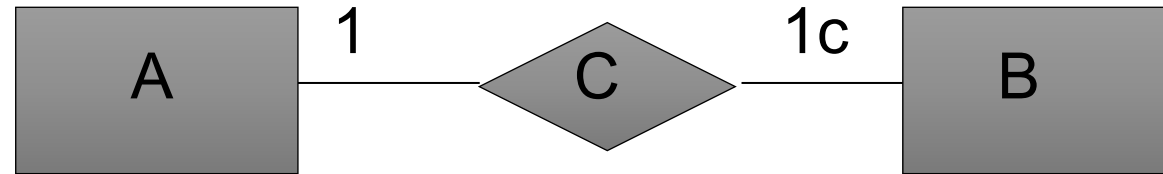
3) Implemento las relaciones según cardinalidad (1:1;1:M;N:M)

Relaciones 1:1



Conversiones del Modelo DER al modelo relacional.

Relaciones 1:1

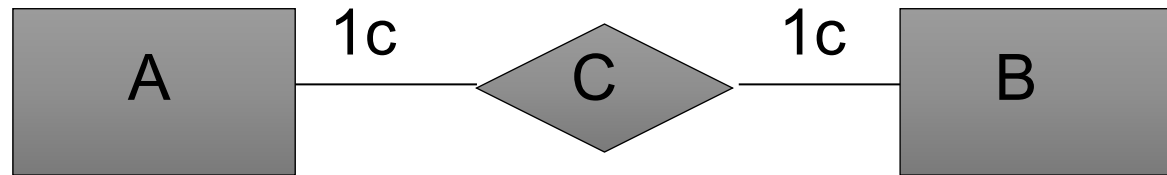


En la Tabla donde existe condicionalidad debe
Incluirse la clave primaria de la Otra Tabla.



Conversiones del Modelo DER al modelo relacional.

Relaciones 1:1

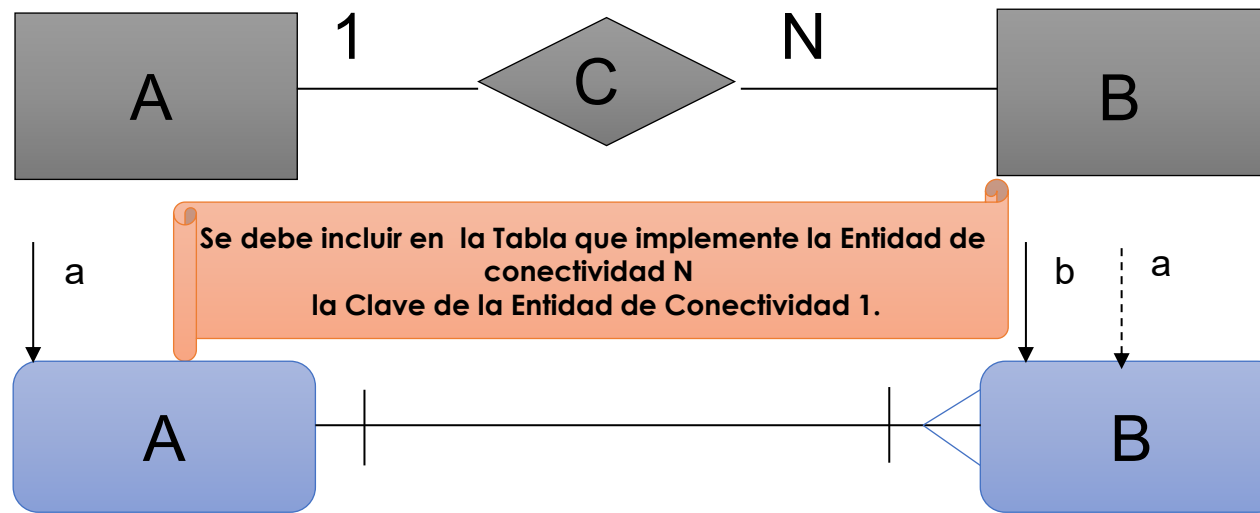


Se pasa la clave de Ambas Tablas, teniendo en Cuenta que debe aceptar valores nulos.

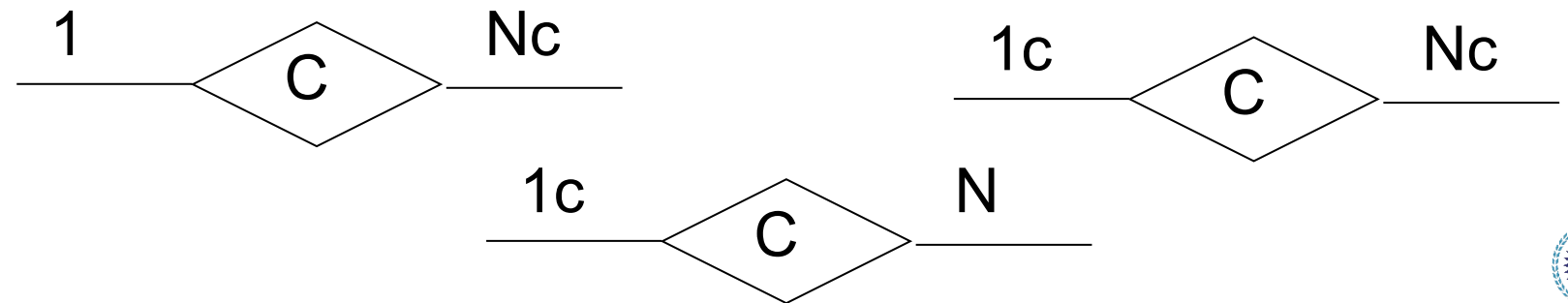


Conversiones del Modelo DER al modelo relacional.

Relaciones 1:M

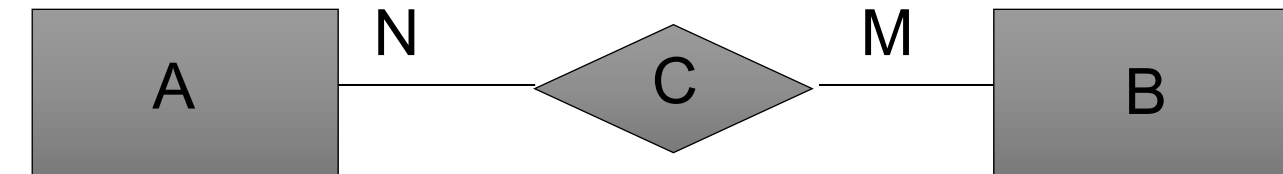


Se utiliza el mismo criterio para :

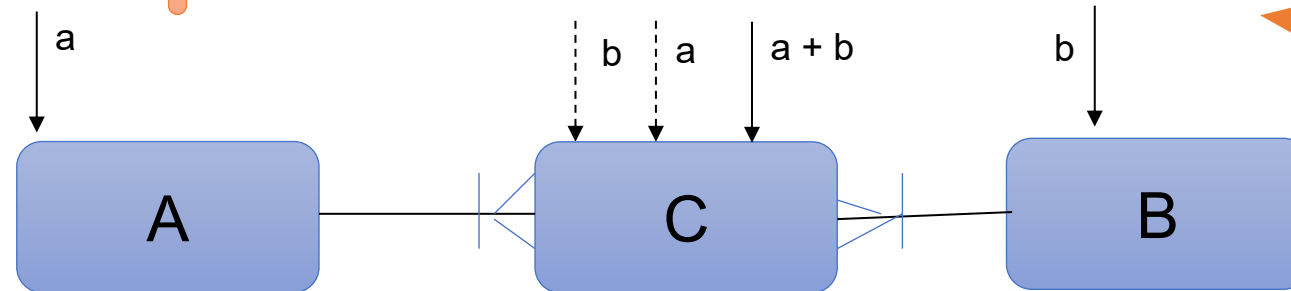


Conversiones del Modelo DER al modelo relacional.

Relaciones N:M



La relación genera una Tabla de Correlación.

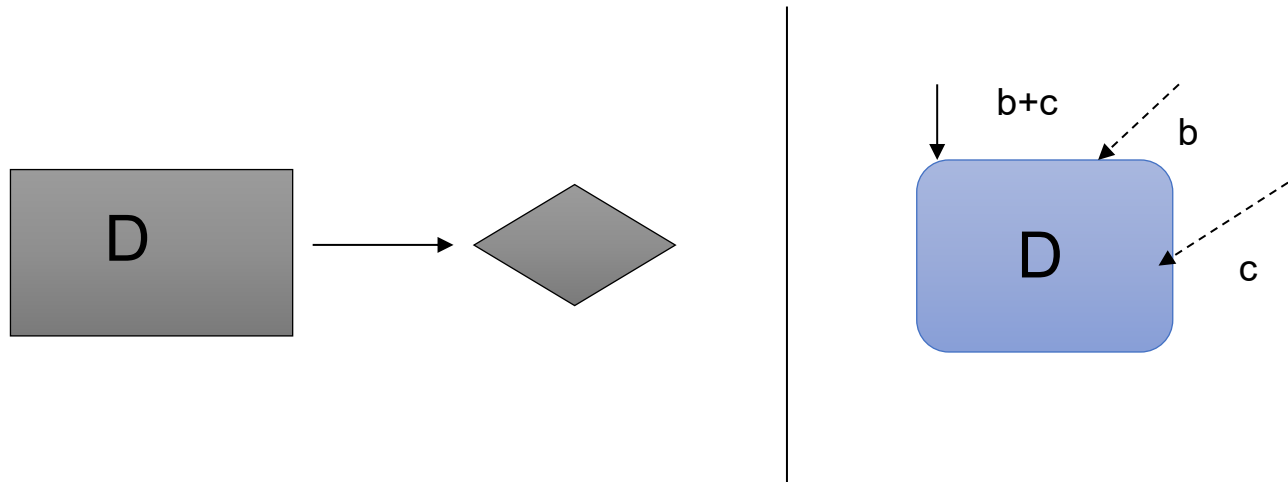


No existen Relaciones De N a M en un Esquema de Tablas

Depende la Cardinalidad y del Criterio de Diseño, $a+b$ pueden o no ser Claves primarias de C

Conversiones del Modelo DER al modelo relacional.

4) Implemento los Objetos Asociativos



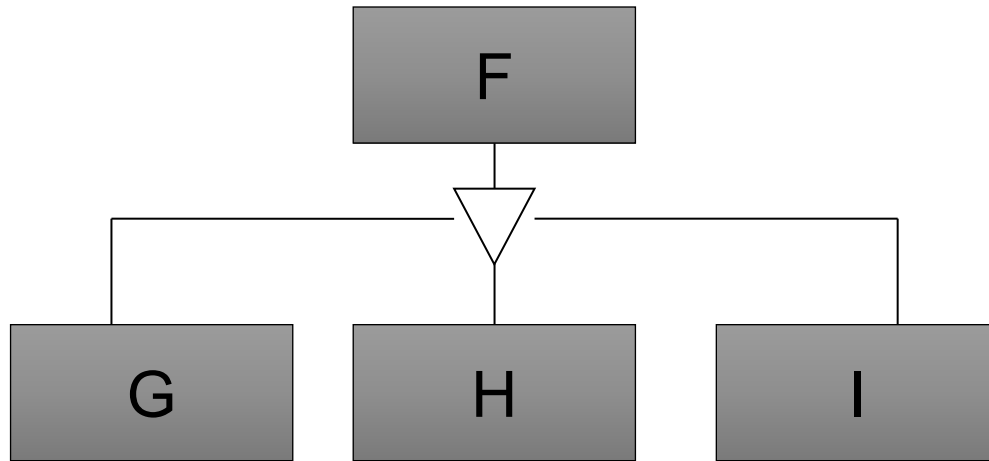
La Clave en principio puede ser la concatenación de las Claves de los Objetos que la conforman.

Pero es necesario analizar si puedo tener repetidos, y en ese caso definir una nueva clave

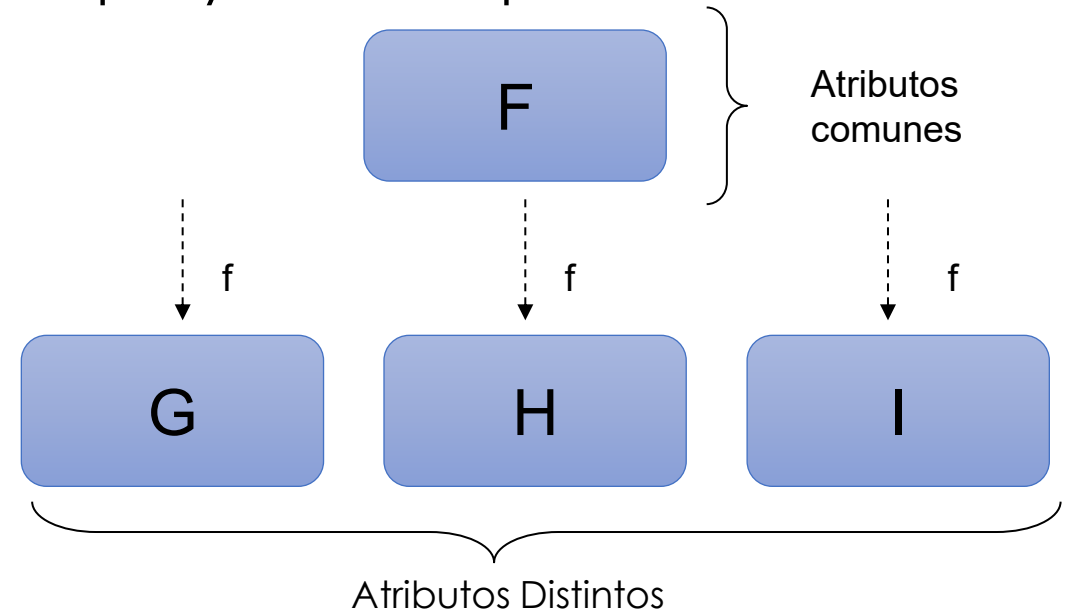
Conversiones del Modelo DER al modelo relacional.

5) Implemento los Supertipos y los Subtipos

a) Crear tablas separadas para el supertipo y sus subtipos



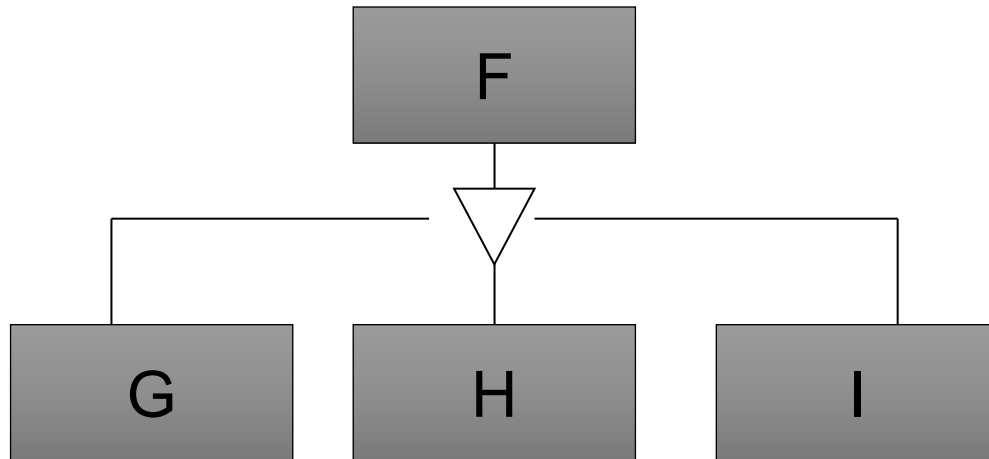
- ▶ Facilidad de adaptación
- ▶ Mejor Uso de espacio de Almacenamiento.
- ▶ Ideal con muchas ocurrencias de Supertipo y Subtipo.



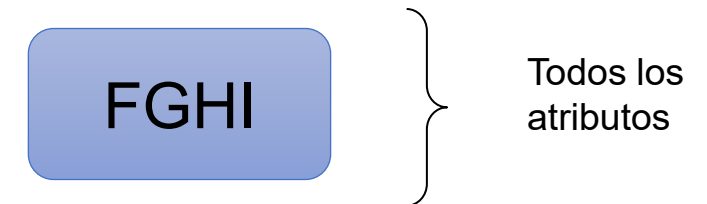
- ▶ Mayor Complejidad
- ▶ Se requiere mayor navegación entre las tablas para elegir una tupla del subtipo.

Conversiones del Modelo DER al modelo relacional.

b) Combinar el supertipo y todos sus subtipos en una sola tabla.



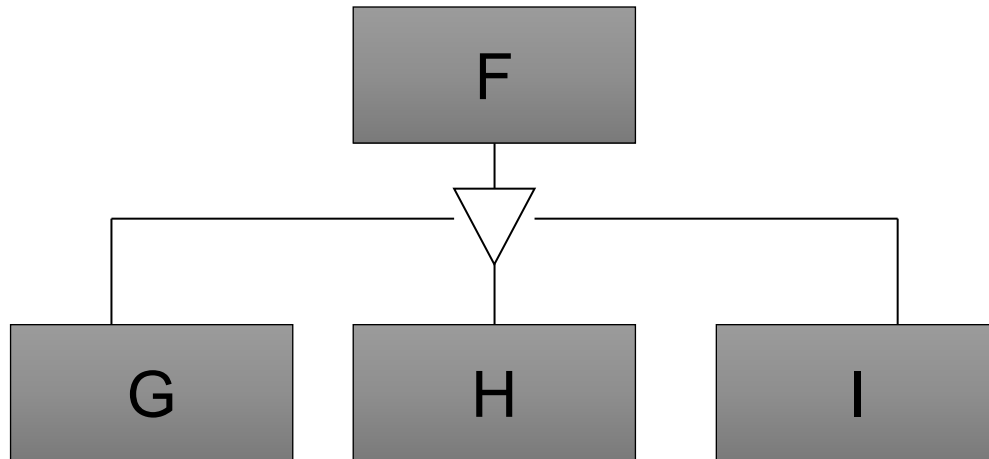
- La tabla resultante posee todos los elementos resultantes.
- Menor Acceso a disco.
- Menor Navegación entre tablas.



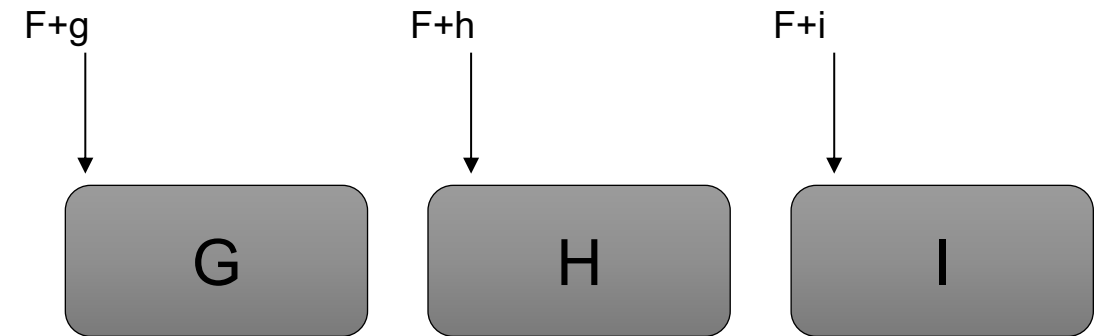
- Mayor ocupación de Disco.
- Gran Número de columnas con Valores Nulos.

Conversiones del Modelo DER al modelo relacional.

c) Representar todos los subtipos como tablas separadas y no Implementar el Supertipo.



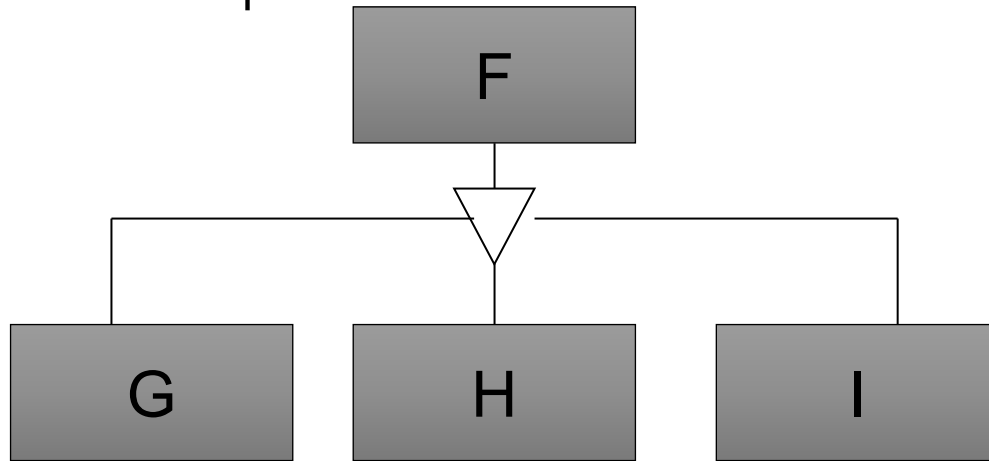
- Las tablas que implementan los subtipos tienen sus propios atributos y los del supertipo.
- Es viable cuando el supertipo tiene pocos atributos.



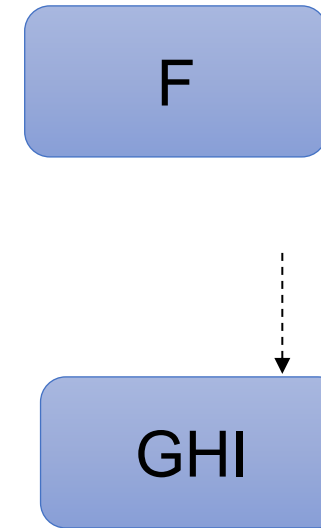
- Es complicado mantener una clave de Supertipo Común.

Conversiones del Modelo DER al modelo relacional.

d) Implementar el supertipo como una tabla y combinar todos los subtipos en una tabla

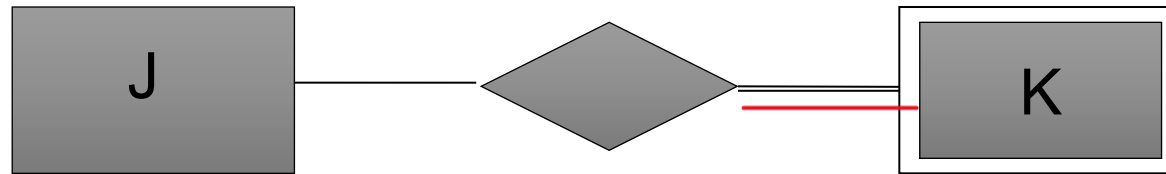


- La Tabla que contiene los Subtipos contiene todos los atributos de los mismos.
- Es viable cuando la mayoría de las consultas acceden al supertipo.

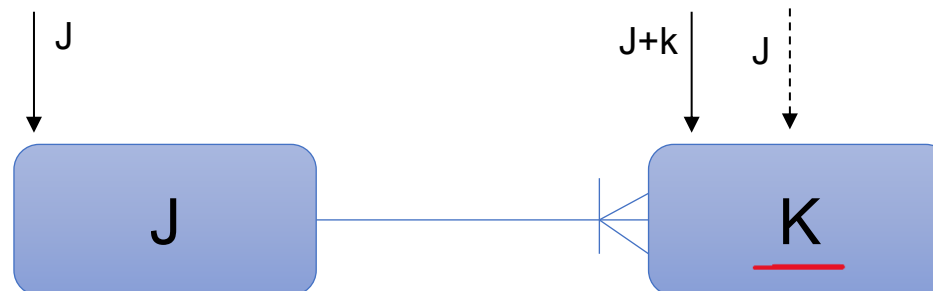


- Se generan atributos con valores nulos.
- La navegación para determinar los atributos de los subtipos es compleja.

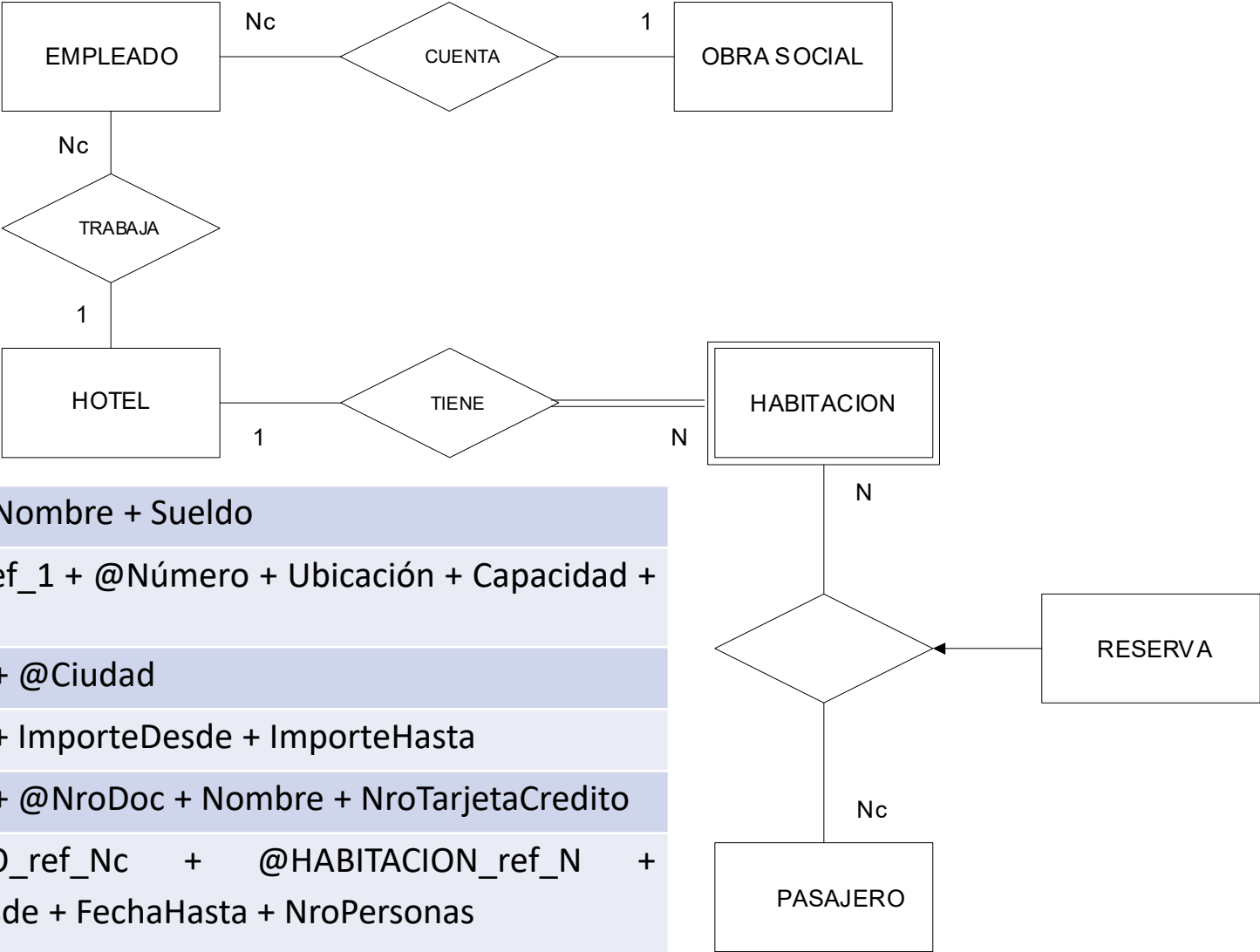
5) Implemento las Entidades débiles y las Entidades fuertes



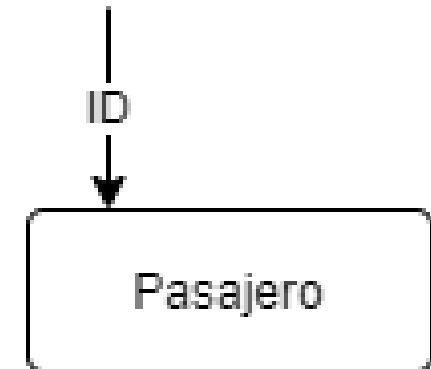
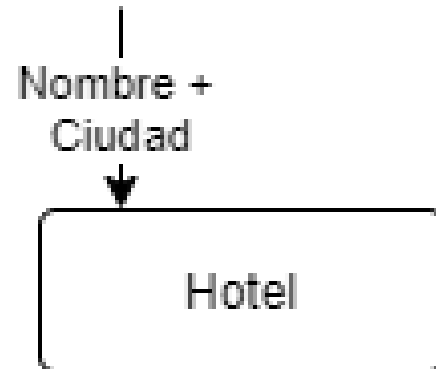
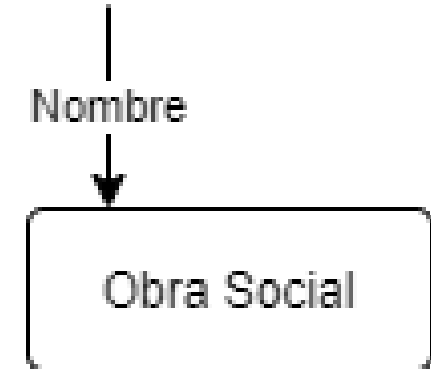
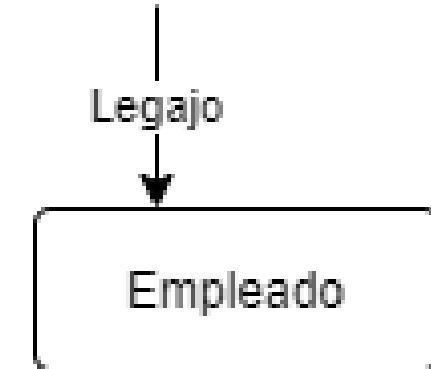
La Clave Será la
Suma de la clave
De K más
la Foránea



DER Hoteles



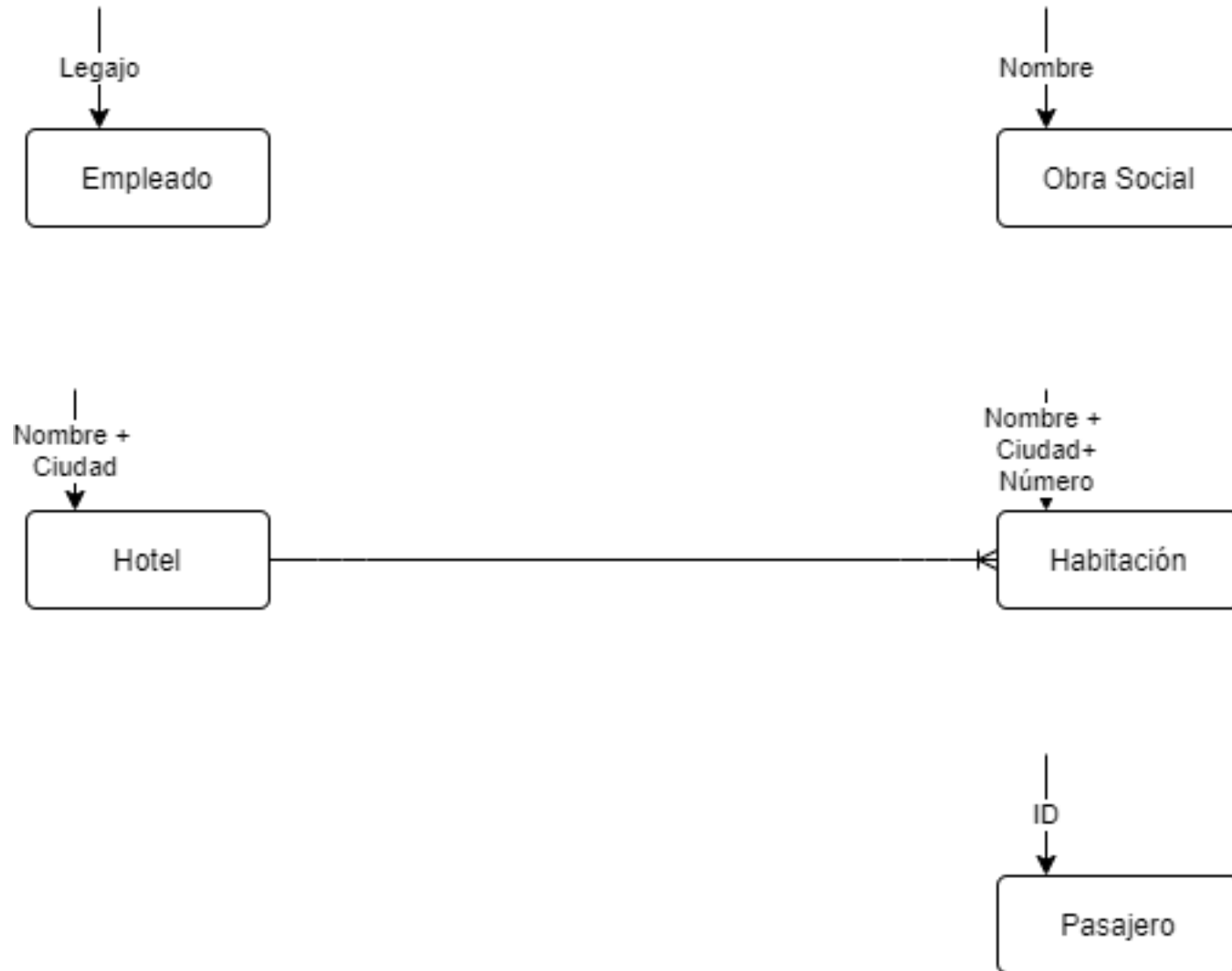
Definir una tabla por cada objeto



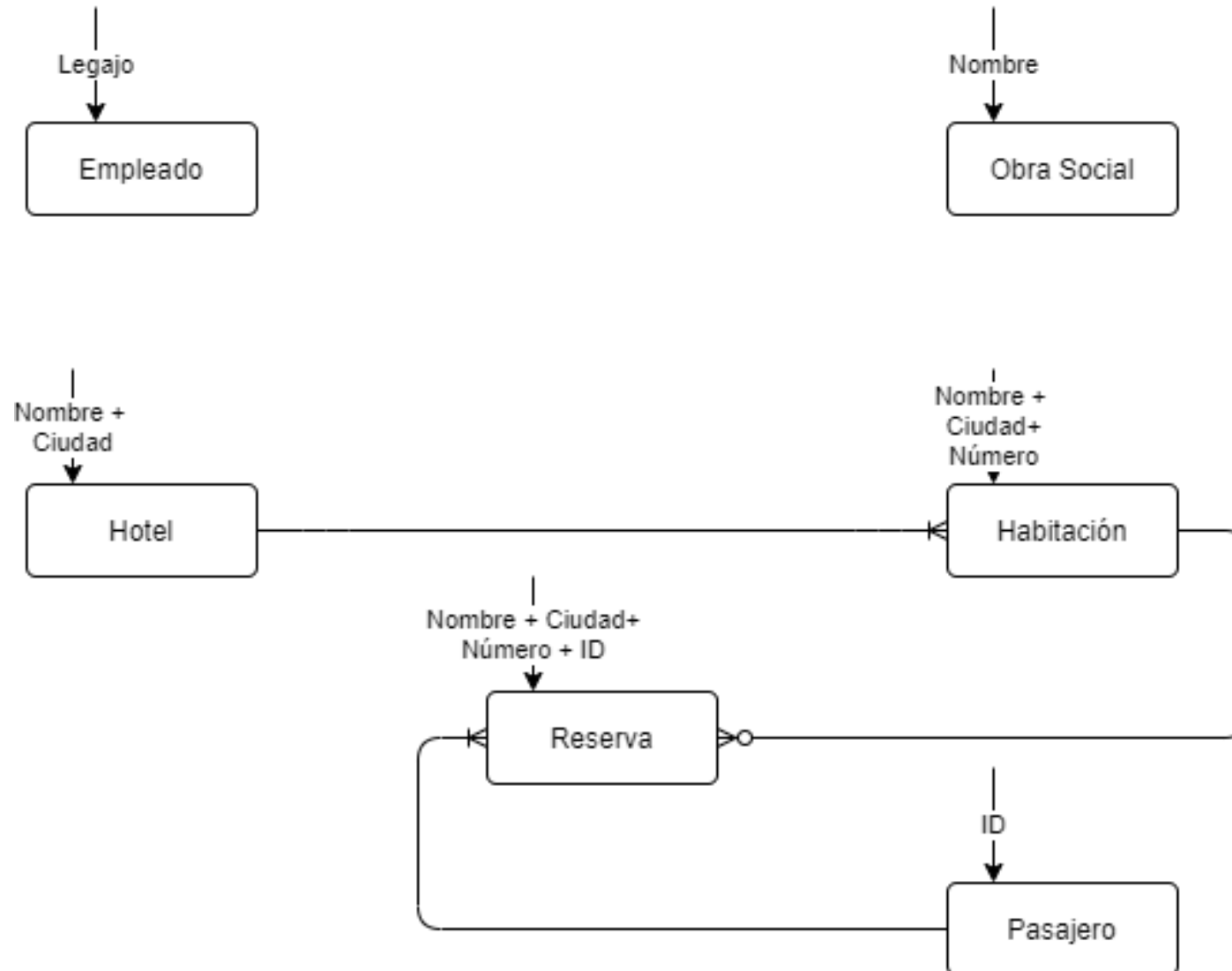
Implementar Super y Subtipos

- En este caso no hay

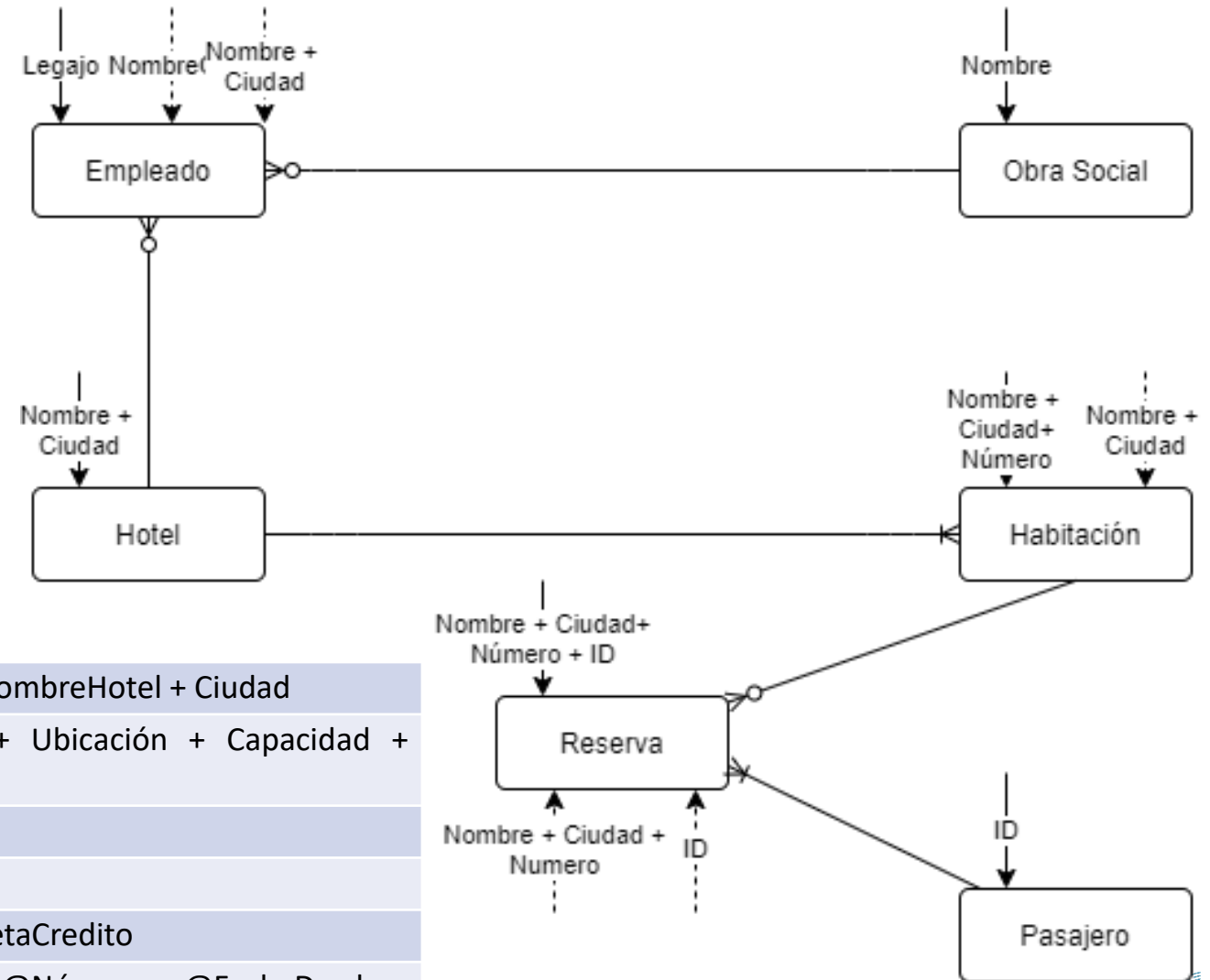
Implementar Objetos Débiles



Implementar Objetos Asociativos



Implementar relaciones



Empleado	=	@Legajo + Nombre + Sueldo + NombreOS + NombreHotel + Ciudad
Habitación	=	@NombreHotel + @Ciudad + @Número + Ubicación + Capacidad + Categoría
Hotel	=	@Nombre + @Ciudad
Obra_Social	=	@Nombre + ImporteDesde + ImporteHasta
Pasajero	=	@ID + TipoDoc + NroDoc + Nombre + NroTarjetaCredito
Reserva	=	@IDPasajero + @NombreHotel + @Ciudad + @Número + @FechaDesde + FechaHasta + NroPersonas