

Diseño de bases de datos relacionales

Sistema plano, sin jerarquías ni relaciones

CLIENTES

Simon Knutsön

Jean-Christophe Laurenbaum

Axelle Playoust-Brâure

EMPLEADOS

Thomas Matzinger

Olé Martínez Möen

David Peace

Tobias Lenaert

Jonathan Lughton

PROVEEDORES

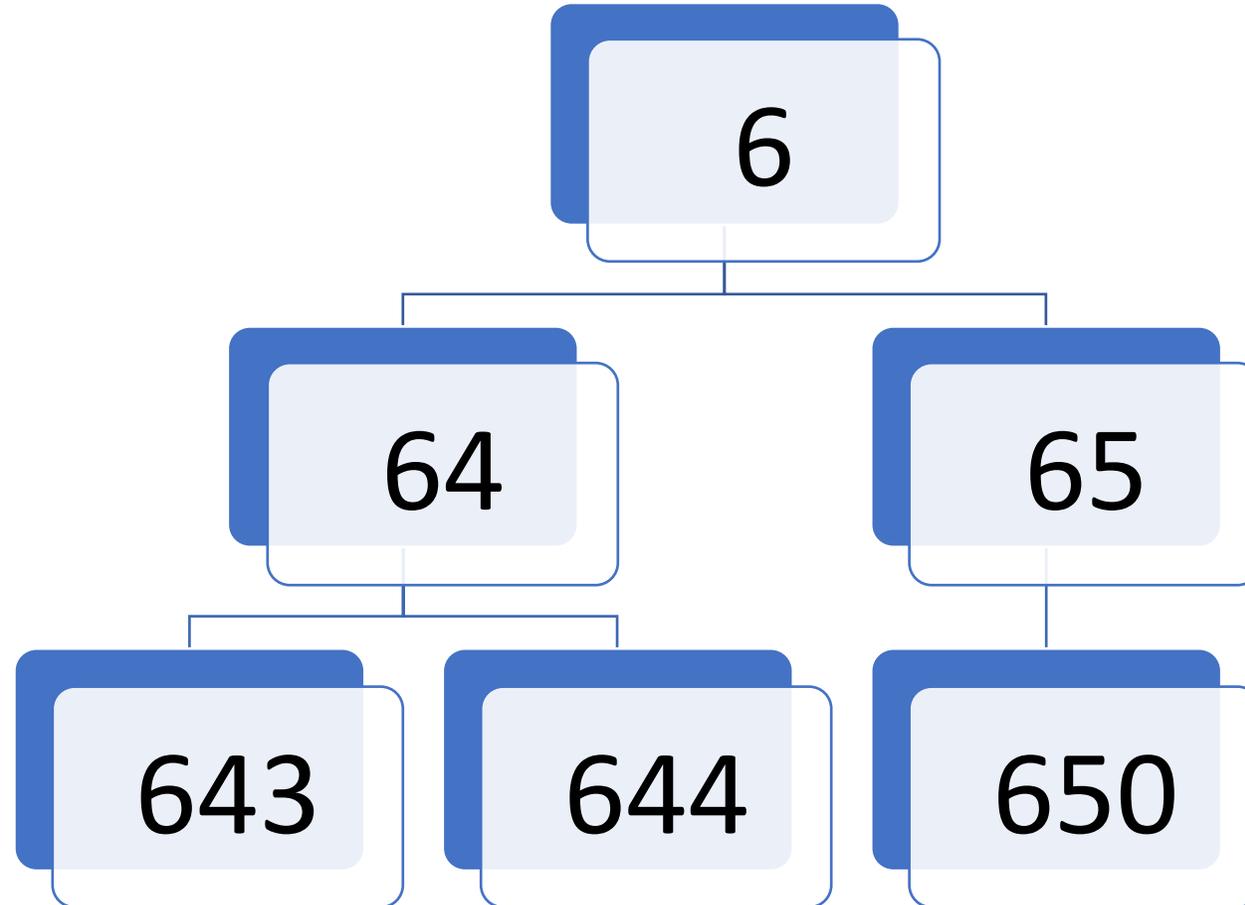
Jonathan Byrch

Axelle Playdust-Brâure

Sistema Jerárquico

- La característica principal de un sistema jerárquico es que existen relaciones “jerárquicas” (“padre-hijo” o “madre-hijo”) de manera que cada elemento o nodo puede tener varios hijos (o ninguno) pero un nodo sólo puede tener una madre (o ninguna, si es el elemento superior de toda la jerarquía)
- ¿Puede haber más de un elemento sin madre?
 - No, si sólo tenemos un árbol
 - Sí, si tenemos varios árboles desconectados

Sistema Jerárquico. Ejemplo: Plan contable



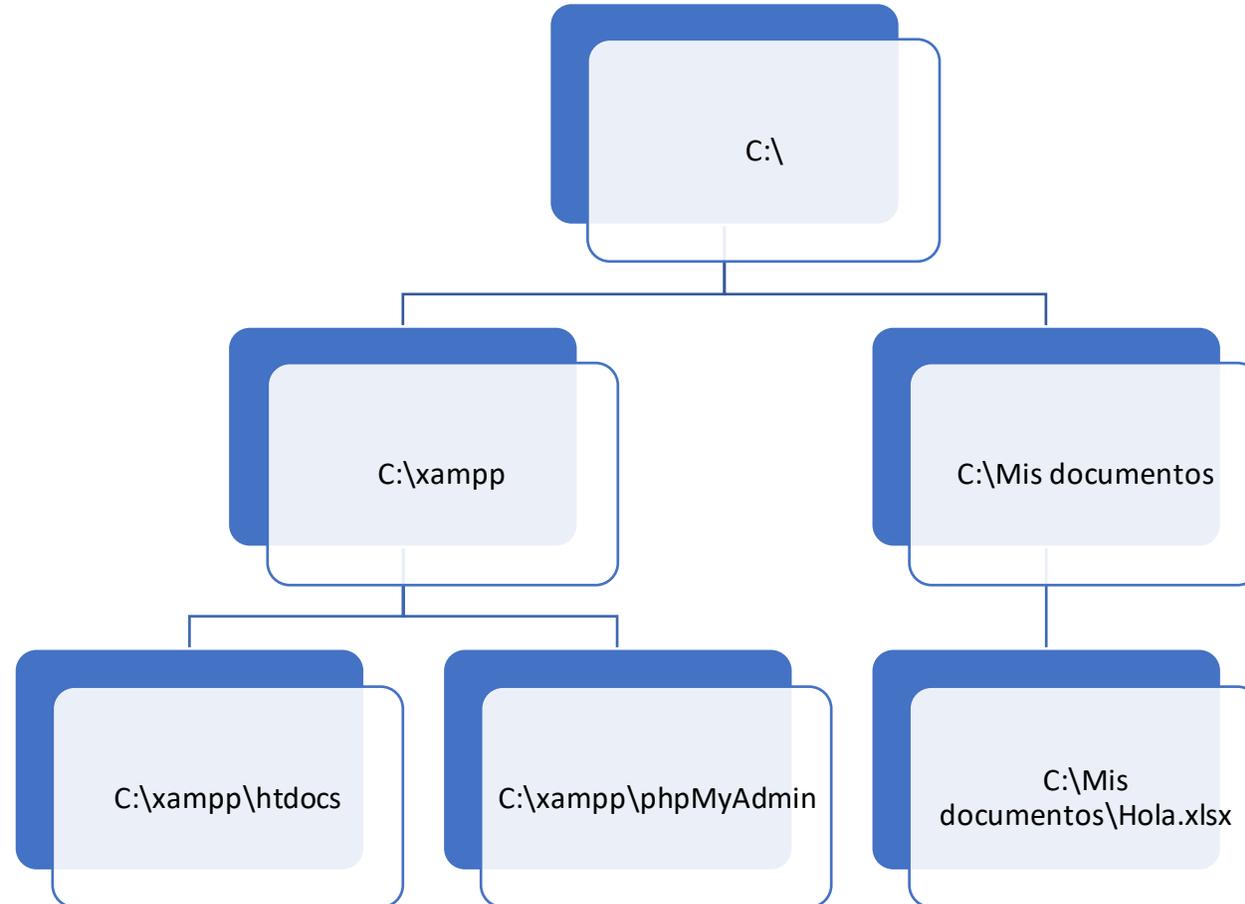
Sistema Jerárquico. Ejemplo: Plan contable

- Las relaciones 1-N se representan mediante punteros desde los hijos hacia el padre
- 6 → NULL
- 64 → 6
- 65 → 6
- 643 → 64
- 644 → 64
- 650 → 65

Sistema Jerárquico. Ejemplo: Empleados

- Las relaciones 1-N se representan mediante punteros desde los hijos hacia el padre (¿a quién reporta cada empleado?)
- Marko → NULL
- Suzy → Marko
- Alejandra → Marko
- Carlos → Alejandra
- Germán → Suzy
- Leo → Suzy

Sistema Jerárquico. Sistema de archivos de windows



Sistema relacional - verbos

- Un departamento tiene varios empleados
 - Un empleado pertenece a un solo departamento
- Un libro es prestado a un usuario
 - Un usuario puede tener en préstamo varios libros
- Un estudiante se matricula en varios cursos
 - En un curso se matriculan varios alumnos
- Un empleado posee un teléfono móvil de empresa
 - Un móvil de empresa solo puede pertenecer a un empleado

Bases de Datos Relacionales (SQL)

- MySQL
- Access
- SQL Server
- Oracle
- DB2
- ...
- ...

Diagrama E/R (Entidad-Relación) y Diseño Relacional

- Diagrama E/R (Entidad-Relación): [sí puede haber N-M]
 - Alto nivel, más alejado de la implementación en una BD real.
 - Describe las entidades del mundo real que se quiere modelar, y sus relaciones.
 - Se centra en los aspectos fundamentales que se quieren representar.
- Diseño Relacional de BD: [no puede haber N-M]
 - Nivel medio. Asume las limitaciones de un diseño de BD relacional, pero no establece la codificación para un tipo de BD particular.
- Diseño físico:
 - Bajo nivel, más cercano a la BD.
 - Diseño de tablas y restricciones en la BD, en lenguaje SQL específico para un manejador de BD, como por ejemplo, MySQL.

Proceso de diseño de BD. Estrategia

Propuesta de pasos para diseñar una BD:

1. Identificar las entidades (del mundo real)
2. Describir casos de uso y crear **datos de ejemplo en Excel**
3. Diagrama E/R <----- Modelo Conceptual
4. Diseño relacional <----- Modelo Lógico (BD-R)
5. Código SQL <----- Modelo "Físico" (MySQL)

El diagrama E/R ignora los aspectos informáticos. En él no se tiene en cuenta que después va a ser implementado en una BD relacional.

Definiciones

- Diagrama E/R (Entidad-Relación)
 - Centrado en las entidades del mundo real
- Diseño de BD Relacional
 - Centrado en cómo las representamos en una BD Relacional

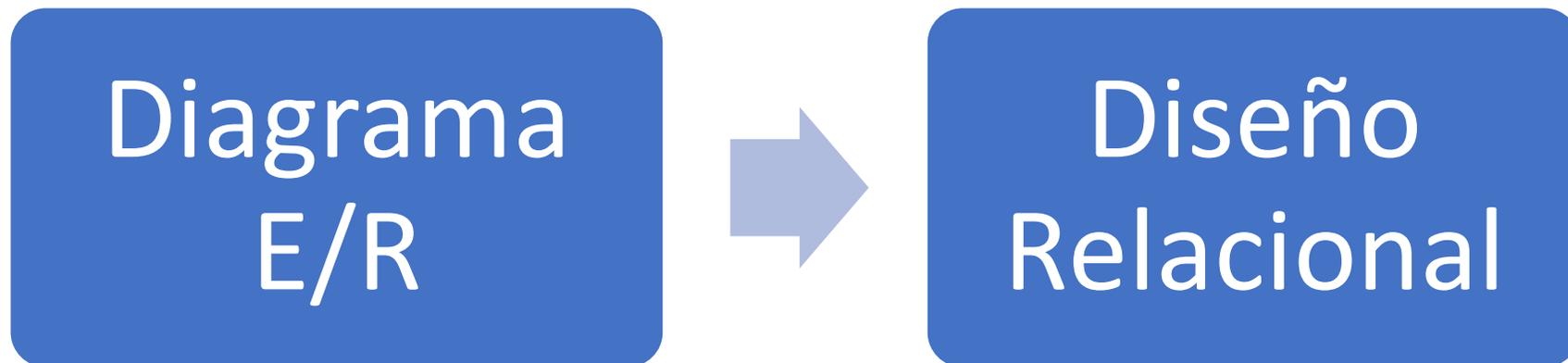


Diagrama E/R (Entidad-Relación)

- El **Diagrama Entidad-Relación (E/R)** es una representación gráfica que muestra las entidades del mundo real y las relaciones entre ellas. En un diagrama E/R existen:
- **Entidades:** Representan objetos o conceptos del mundo real (por ejemplo, *Cliente*, *Pedido*). Son agrupaciones de objetos homogéneos.
- **Relaciones:** Muestran cómo las entidades están conectadas (por ejemplo, un *Cliente* puede hacer *Pedidos*).
- **Cardinalidad:** Indica el tipo de relación entre entidades, como una a uno (1-1), uno a muchos (1-N), muchos a muchos (N-M), etc.

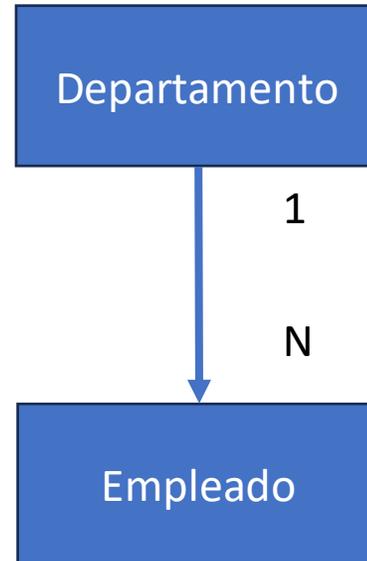
Diseño de BD Relacional

- El **Diseño Relacional** se basa en la teoría de conjuntos y las relaciones entre ellos. En este diseño encontramos:
- **Tablas**: Cada entidad se convierte en una tabla.
- **Filas**: Representan instancias de la entidad.
- **Columnas**: Representan atributos de la entidad.
- **Normalización**: Proceso de organizar los datos para reducir la redundancia (de todo lo que no son claves) y mejorar la integridad de los datos (evitar errores).
- **Claves Primarias (PK)**: Identifican de manera única cada fila en una tabla.
 - “No se puede repetir”
- **Claves Foráneas (FK)**: Enlazan tablas y representan relaciones entre ellas.
 - “El padre debe existir”
- **Otras restricciones**: Otros valores que también deben ser únicos, sin repetidos, pero sin ser clave primaria (claves candidatas o UNIQUE); valores que no pueden ser nulos (NOT NULL)...
- **Otras características**: Índices (para agilizar las búsquedas)

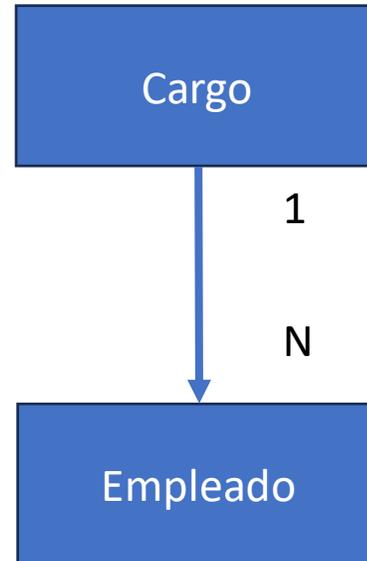
Diagramas E/R

- Los diagramas de entidad-relación son diagramas que muestran cómo ciertas entidades en una base de datos se relacionan con otras. Se utilizan para visualizar rápidamente la lógica de una base de datos, ayudar con el diseño de la base de datos o comunicar la estructura de la base de datos a los usuarios. Los conceptos principales son:
 - Entidades: Objetos que tenemos en nuestra base de datos (normalmente las tablas): Empleados, departamentos, clientes, proveedores, ventas, libros, países...
 - Atributos o columnas: Son las propiedades de las entidades: Nombre, apellidos, fecha de nacimiento, unidades vendidas, título, autor, capital...
 - Relaciones: Cómo se relacionan las diferentes entidades entre sí (mediante claves): Un país tiene una ciudad capital y una ciudad capital es la capital de un país. Un empleado puede realizar muchas ventas y una venta determinada la realiza un solo empleado. Un autor puede escribir muchos libros y un libro puede ser escrito por muchos autores.

Diagramas Entidad - Relación



Diagramas Entidad - Relación



Diagramas Entidad - Relación

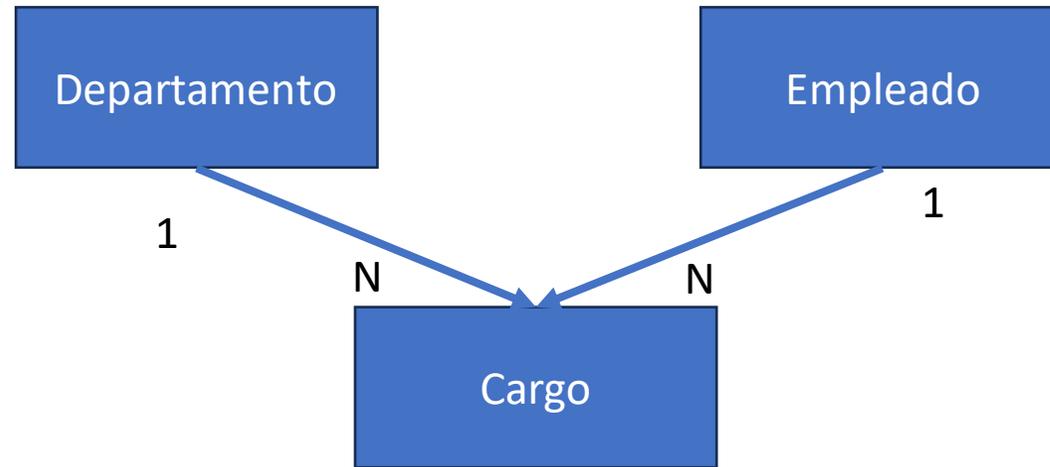
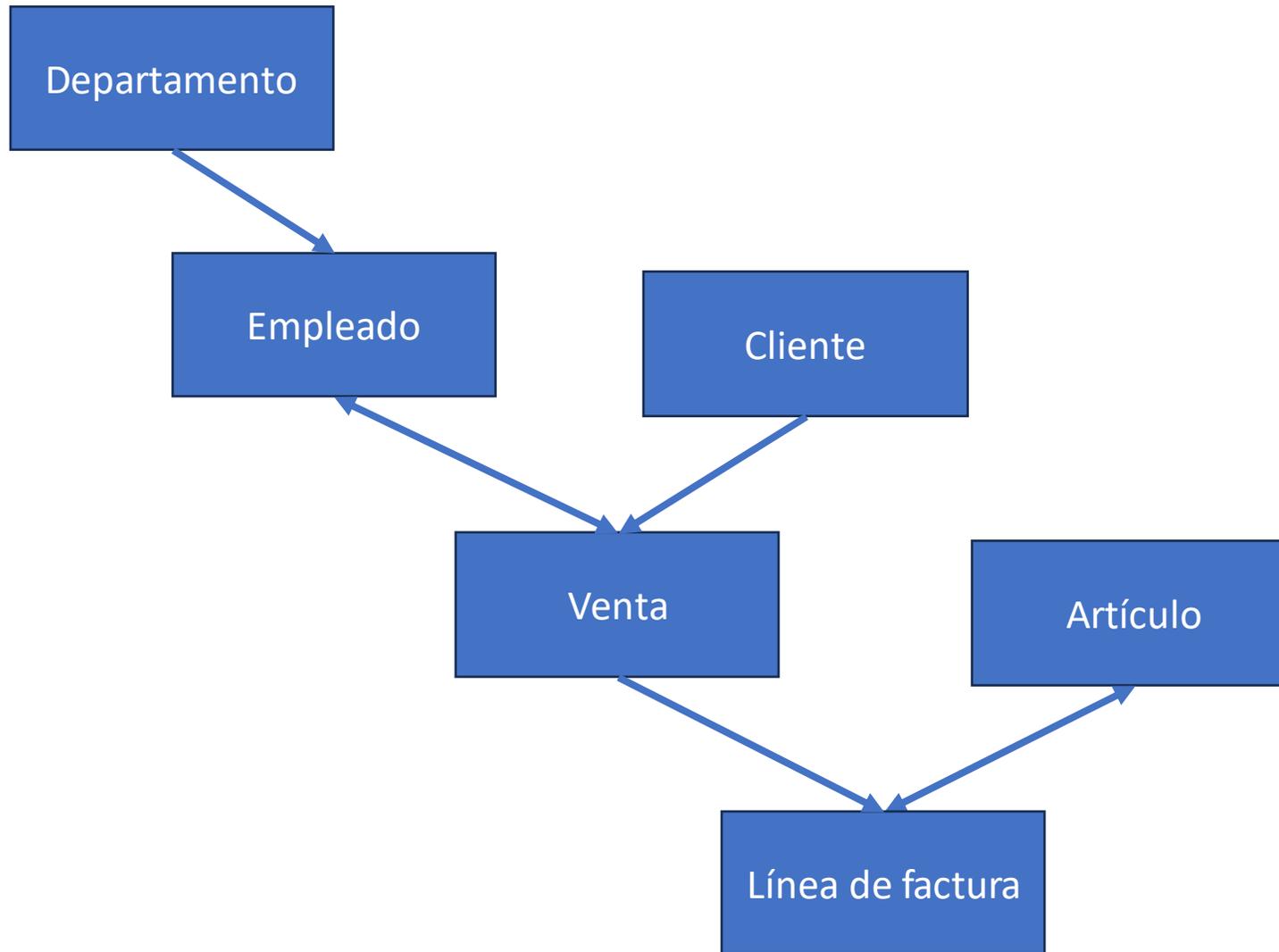
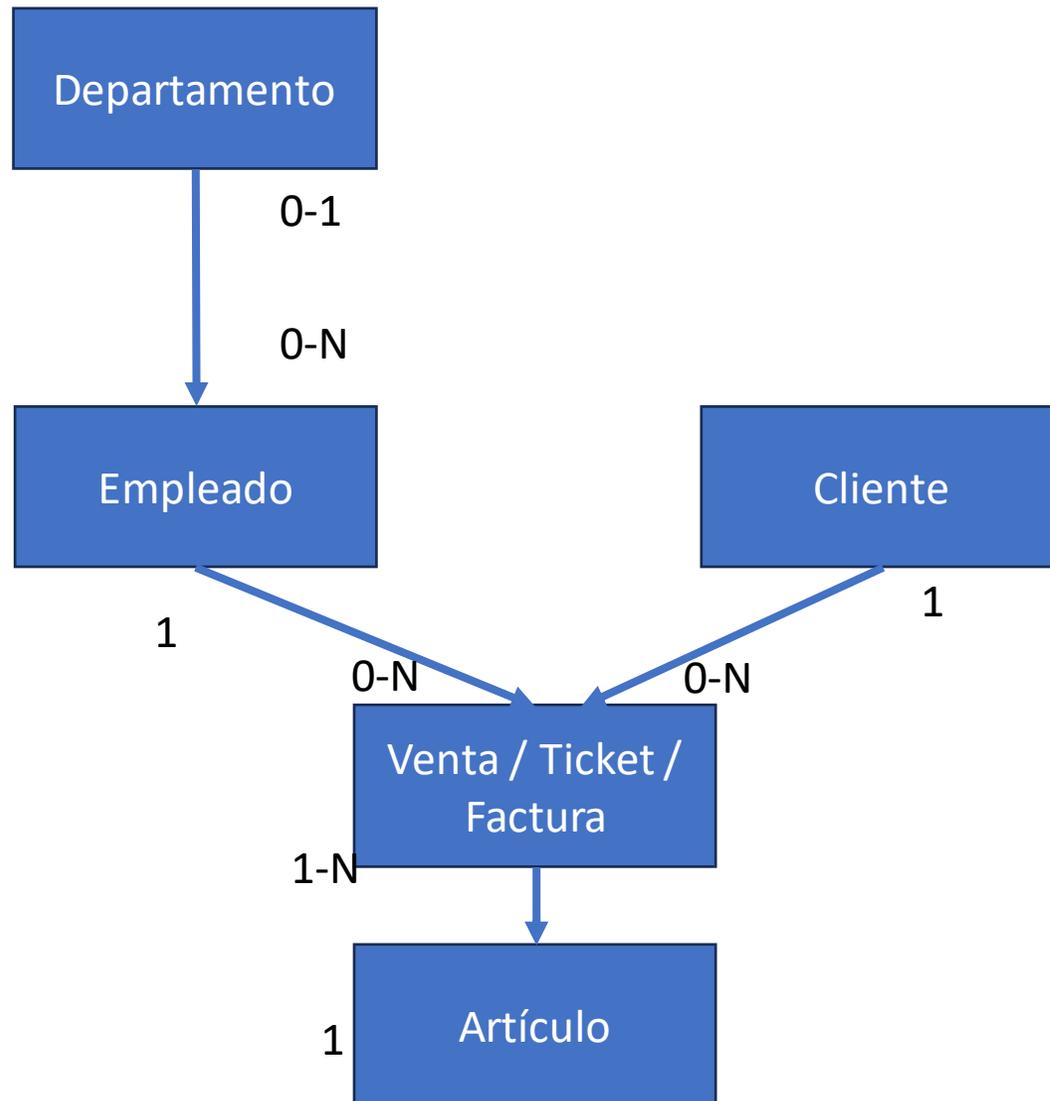


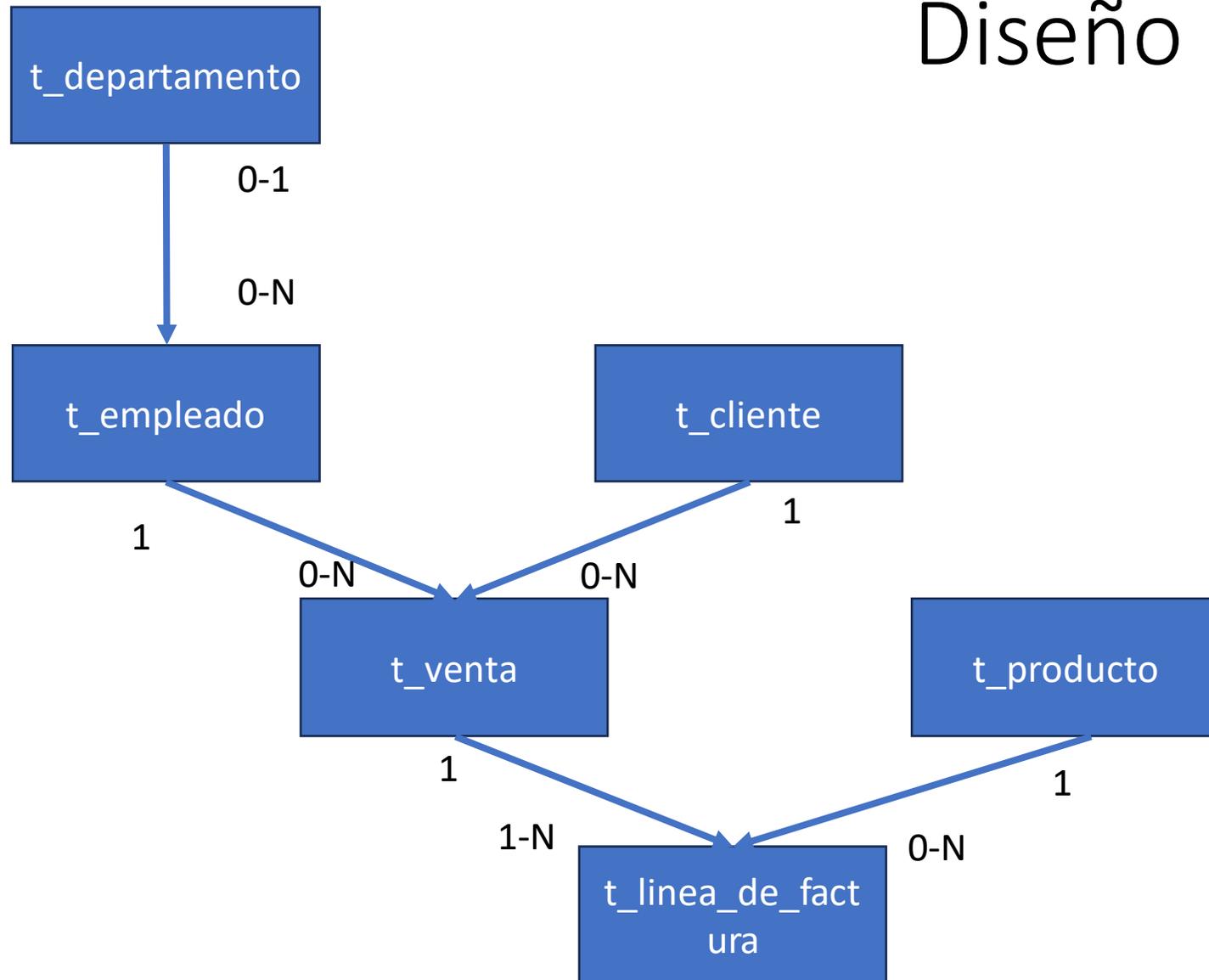
Diagrama Entidad - Relación



Diseño Relacional



Diseño Relacional



Diseño Relacional

Objetivo fundamental de un diseño de BD:

- Las tablas de las bases de datos relacionales deben crecer en número de filas (“hacia abajo”). Debemos tratar de evitar que crezcan en número de columnas (“hacia la derecha”).
- Un diseño de BD estable y robusto no requiere cambiar la estructura, no requiere añadir nuevas columnas (no requiere crecer “hacia la derecha”)
- Ejemplo: Ejemplos diseño 1-N (dos tablas) → Bandas de rock

Conceptos en un Diagrama E/R (mundo real)

- Entidad: agrupación homogénea de objetos del mundo real
 - Ejemplo: todos los clientes forman la entidad “Clientes”
 - Ejemplo: todos los empleados forman la entidad “Empleados”
- Ocurrencia, Instancia: cada uno de esos elementos homogéneos.

Conceptos en un Diseño Relacional (diseño de BBDD)

- Tabla, listado, “relación” (en el sentido de listado), conjunto de datos, ~~entidad~~
- Fila, registro, tupla, observación (“Data Point”), ~~instancia~~, ~~ocurrencia~~, ~~entidad~~
- Columna, campo, atributo, medida, variable

Diferencia entre Grado y Cardinalidad

- En una relación
- **Grado** es el número de entidades que participan en una relación (en una flecha) **en un diagrama E/R**. Por ejemplo, en la relación entre Departamentos y Empleados, **el grado es 2**.
- **Cardinalidad** establece cuántas ocurrencias de una entidad pueden estar relacionadas con una ocurrencia de la otra entidad. Por ejemplo, en una relación 1-N entre Empleados y Departamentos, la Cardinalidad de los Empleados de un Departamento es **"N"** (puede haber 0-N empleados de 1 departamento) y la Cardinalidad de un Departamento de un empleado es **"1"** (Un empleado puede pertenecer a 0-1 departamentos)

Cómo pasar de un Diagrama E/R a un Diseño Relacional

- Si de una entidad sólo existe un ejemplar
 - Normalmente, no necesitamos una tabla
- Cada relación N-M entre dos tablas se convierte en dos relaciones 1-N, empleando una tabla adicional.
 - Hay casos raros: ver ejemplo “Normalizar móviles” (la tabla adicional es la misma tabla de la relación)
- Si es posible y razonable, simplificar el diseño, de forma que una misma tabla almacene la información de varias entidades.
 - Móviles dentro de la tabla de empleados
 - Tabla persona_fisica almacenando empleados e inversores

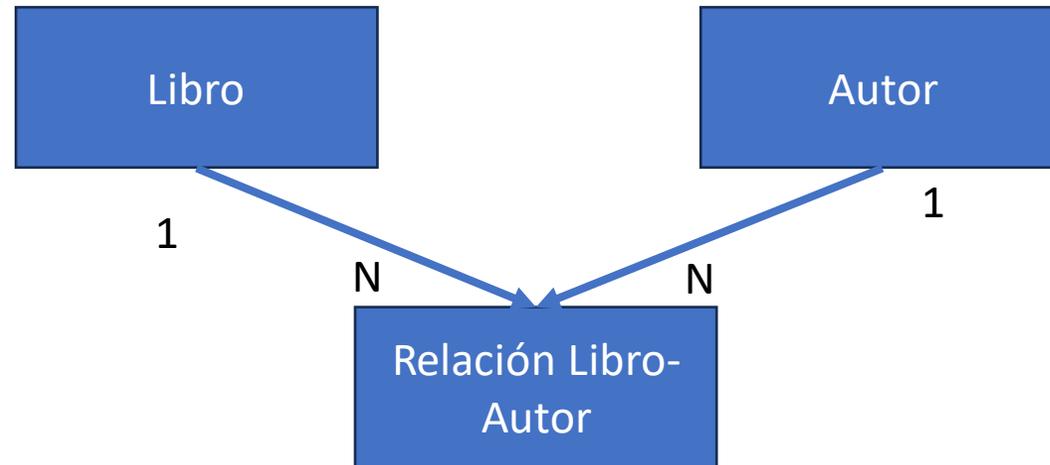
Cómo pasar de un Diagrama E/R a un Diseño Relacional

- En el mundo real (“Diagrama E/R”) hay relaciones:
 - 1 a 1
 - 1 a N
 - N a M
- En la implementación de BBDD Relacionales (“Diseño Relacional”), solo hay relaciones:
 - 1 a 1
 - 1 a N
 - ¿Por qué no hay N-M?

Relaciones muchos a muchos



Relaciones muchos a muchos



Diseño de BD relacionales - Resumen

Tres formas de representar las entidades de un sistema y sus relaciones (modelo de negocio)

- Texto (enunciado del problema)
- Datos de ejemplo no estructurados (Excel)
- Diagrama E/R (N-M)

Tres formas de representar el modelo de una BD relacional

- Diseño de BD Relacional (1-N)
- Datos de ejemplo estructurados (Excel)
- Código SQL de creación de tablas, PK, FK y otras restricciones

Diseño de BD relacionales - Resumen

- Identificar todas las entidades (agrupaciones homogéneas de objetos). Si solo hay uno, frecuentemente se puede ignorar.
- Crear datos de ejemplo en hojas Excel con el **mínimo número posible de columnas** y tantas filas como sea necesario.
- Asignar los atributos **específicos** de las entidades a las entidades (**datos que sean únicos**). Evitar información repetida. Si es necesario, crear nuevas tablas.
- Analizar si las relaciones son 1-1, 1-N ó N-M.
 - Transformar las relaciones N-M entre dos tablas en dos relaciones 1-N entre cada tabla y una tercera tabla auxiliar.
 - Las relaciones 1-1 pueden convertirse en una única tabla y viceversa.

Diseño de BD relacionales - Recomendaciones - casi requisitos

1. Independencia del orden

El significado de los datos debe mantenerse, aunque cambiemos el orden de las filas o de las columnas.

2. Todas las tablas tienen clave primaria -que no admite nulos- (por tanto, no hay filas repetidas)

Ejemplo de una posible excepción: un registro de eventos “LOG” en orden temporal sin PK

Diseño de BD relacionales - Otras recomendaciones

1. Atributos atómicos. Los campos (columnas) solo contienen un único valor

Ejemplo de una posible excepción: varios teléfonos en un único campo teléfono.

2. No hay columnas repetidas (varias columnas que representan la misma naturaleza de datos).

Ejemplo de una posible excepción: tres columnas “Email 1”, “Email 2”, “Email 3”

3. No hay redundancia en parte de las filas

El significado de los campos de una tabla depende de su clave primaria. Evitamos tramos repetidos.

Diseño de BD relacionales - Prioridades

1. Versatilidad

Poder almacenar tantas instancias como sea necesario sin tener que cambiar la estructura de la BD (por ejemplo, sin tener que añadir columnas adicionales)

2. Optimizar el rendimiento, la seguridad y sencillez

Disponer de un modelo fácil de entender y ágil. Prevenir la pérdida de datos y accesos no autorizados. Facilitar el desarrollo y la actualización.

3. Evitar la redundancia (de todo lo que no son claves)

Para facilitar la modificación y borrado, evitar errores y minimizar el tamaño.

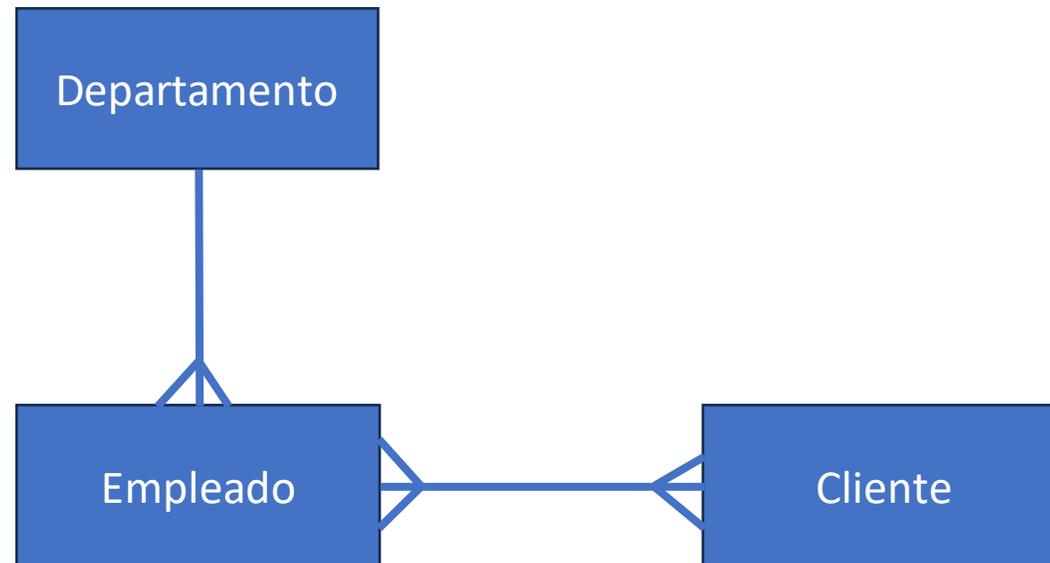
4. Simplificar el diseño

El modelo debe ser tan pequeño como sea posible.

Diseño de BD relacionales - Resumen

- Para pasar de un Diagrama E/R a un Diseño Relacional
 - Las relaciones N-M se convierten en 1-N
- Para pasar de un Diseño Relacional a un Diseño de BD (SQL)
 - Los hijos apuntan a los padres
 - Las PK no pueden ser NULL
 - Las FK sí pueden ser NULL
 - Restricciones UNIQUE, CHECK, NOT NULL, DEFAULT

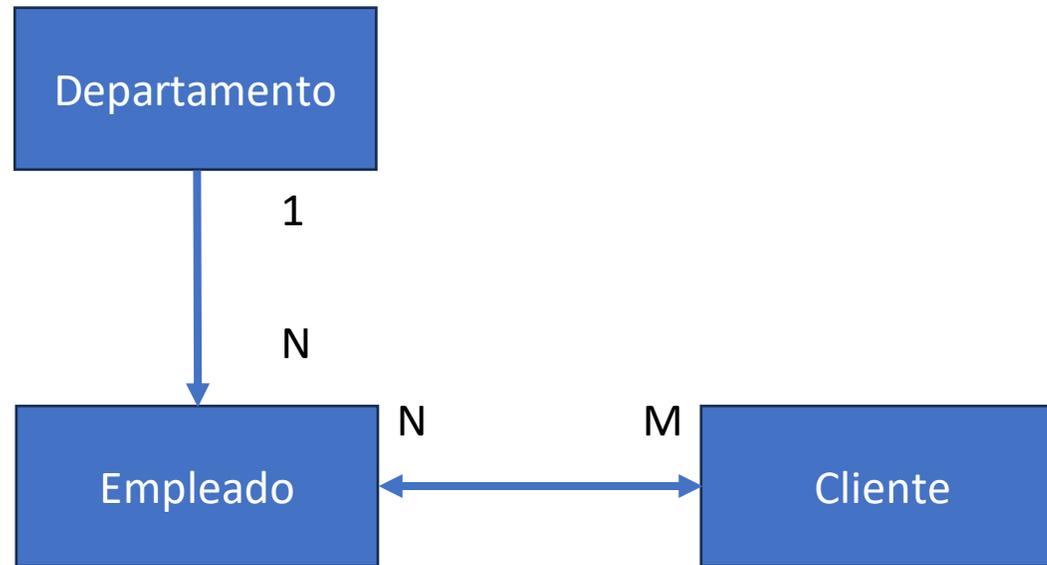
Diagrama Entidad Relación



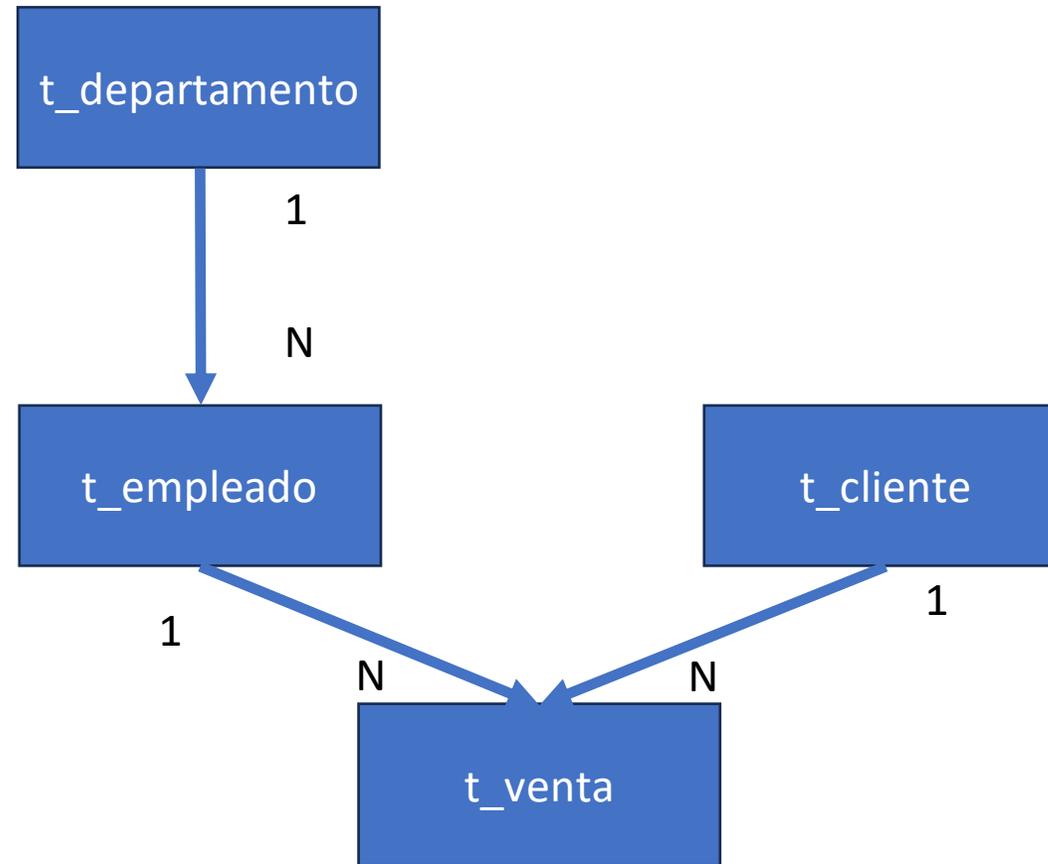
Interpretación de las relaciones

- 1-N
 - Un Departamento puede tener varios empleados
 - Un empleado sólo pertenece a un Departamento
- N-M
 - Un empleado vendedor puede tener varios clientes
 - Un cliente puede ser atendido por varios vendedores

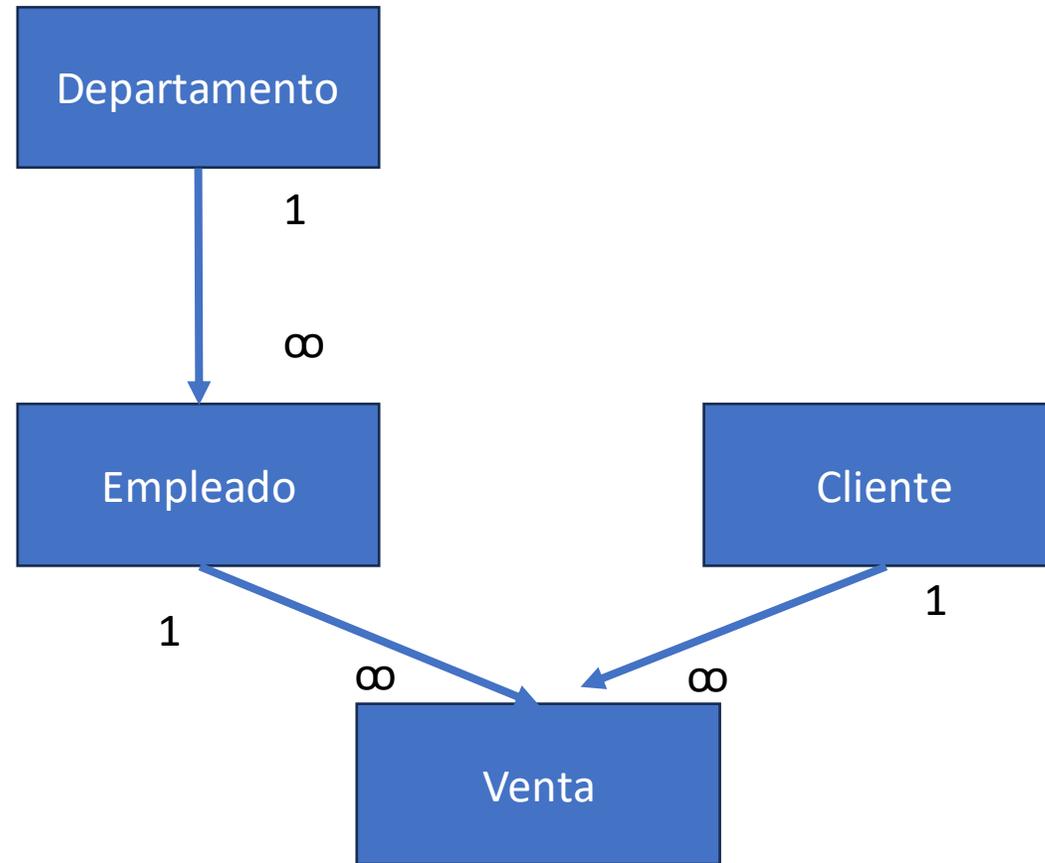
Diagrama Entidad Relación



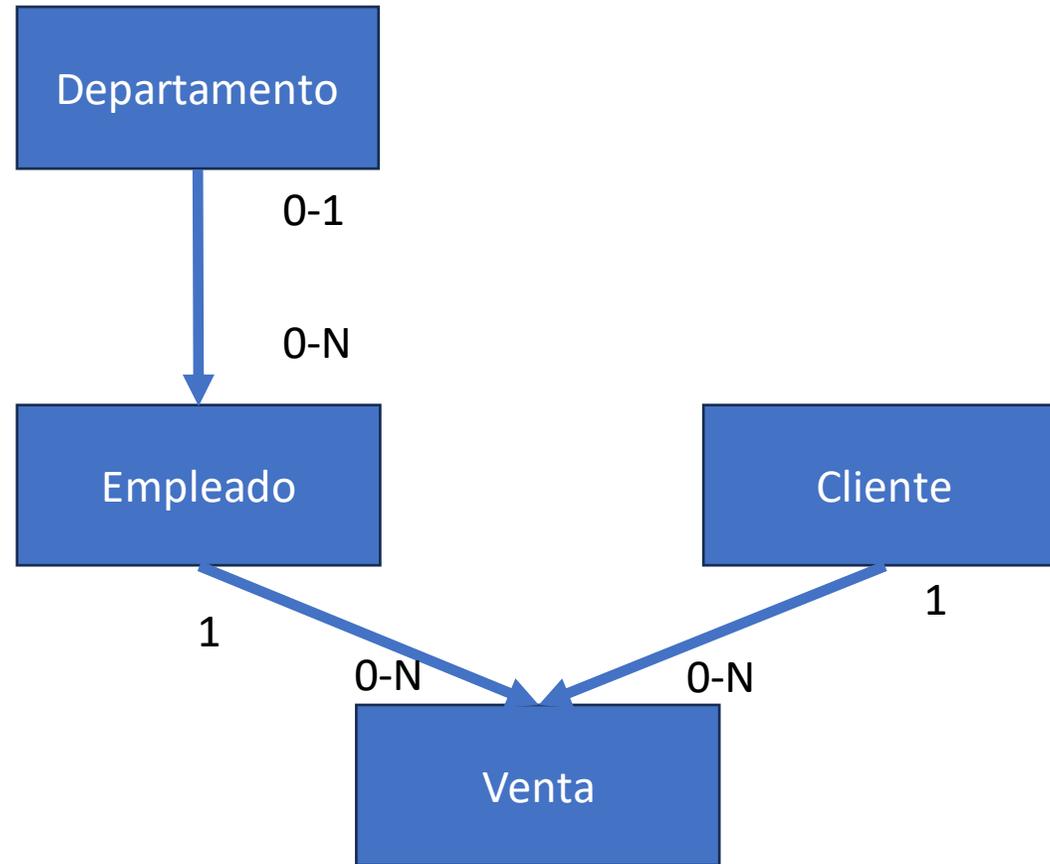
Diseño Relacional



Diseño Relacional



Diseño Relacional

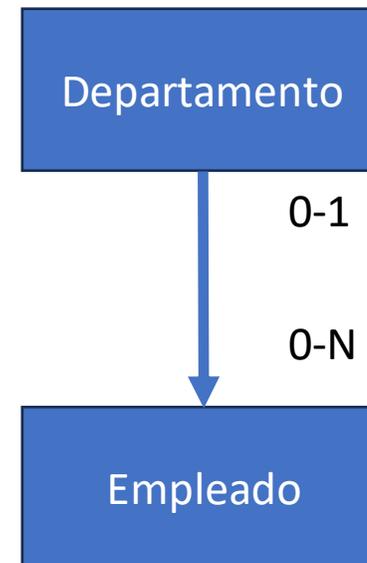


Interpretación de las relaciones

- 1-N
 - Un Departamento puede tener varios empleados
 - Un empleado sólo pertenece a un Departamento o ninguno
- 1-N
 - Un empleado vendedor puede tener varias ventas o ninguna
 - Una venta se asigna a exactamente un empleado
- 1-N
 - Un cliente puede no tener ventas, o tener varias
 - Una venta siempre se asigna a exactamente un cliente

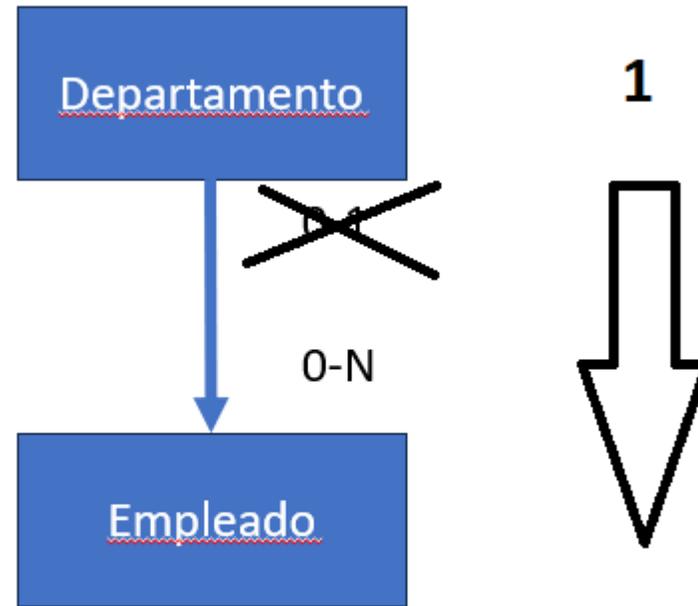
Cómo leer las relaciones

- Cuando leemos de arriba abajo, ignoramos lo que ponga arriba y decimos “un”.
- Cuando leemos de abajo a arriba ignoramos lo que ponga abajo y decimos “un”.



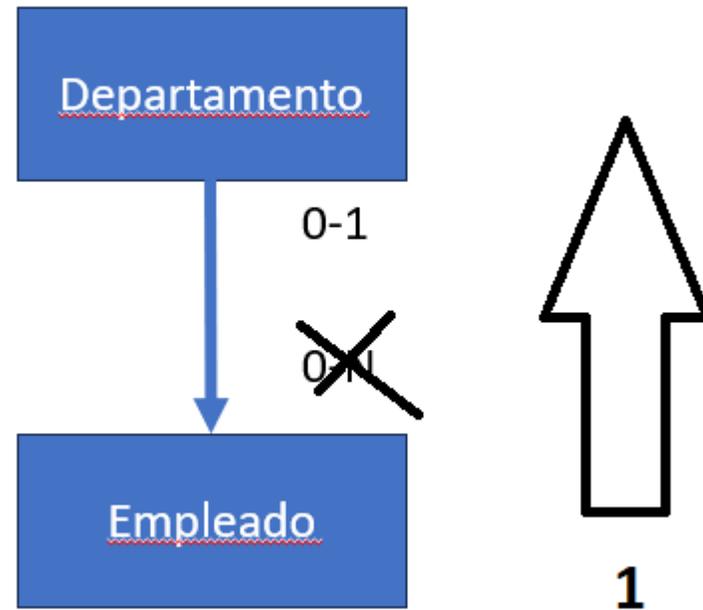
Cómo leer las relaciones

- Un departamento puede tener 0, 1, 2, 3... empleados (0-N)



Cómo leer las relaciones

- Un empleado puede pertenecer a uno o ningún departamento (0-1)



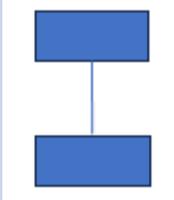
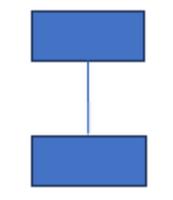
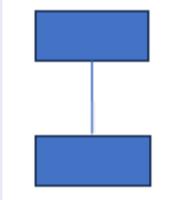
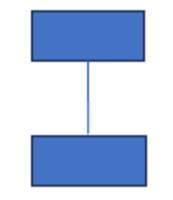
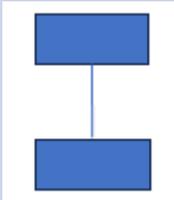
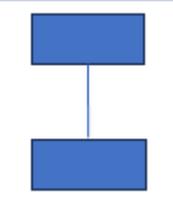
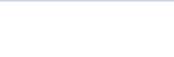
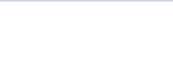
Cómo leer las relaciones

- Cuando una única relación la analizamos en los dos sentidos y nos sale 1-N y 1-N, es una N-M
- Cuando una única relación la analizamos en los dos sentidos y nos sale 1-N y 1-1, es una 1-N

Visión a alto nivel y detallada de las relaciones

“1” = Ninguno
o uno

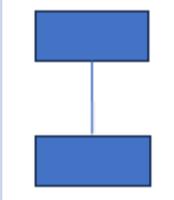
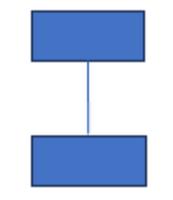
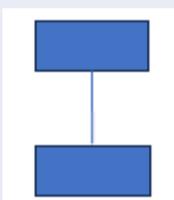
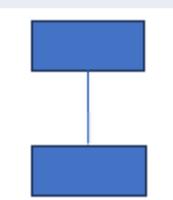
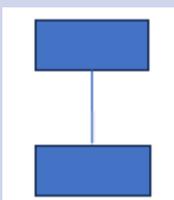
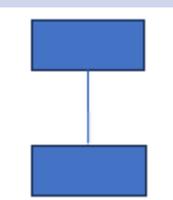
“N” = Ninguno,
uno o varios

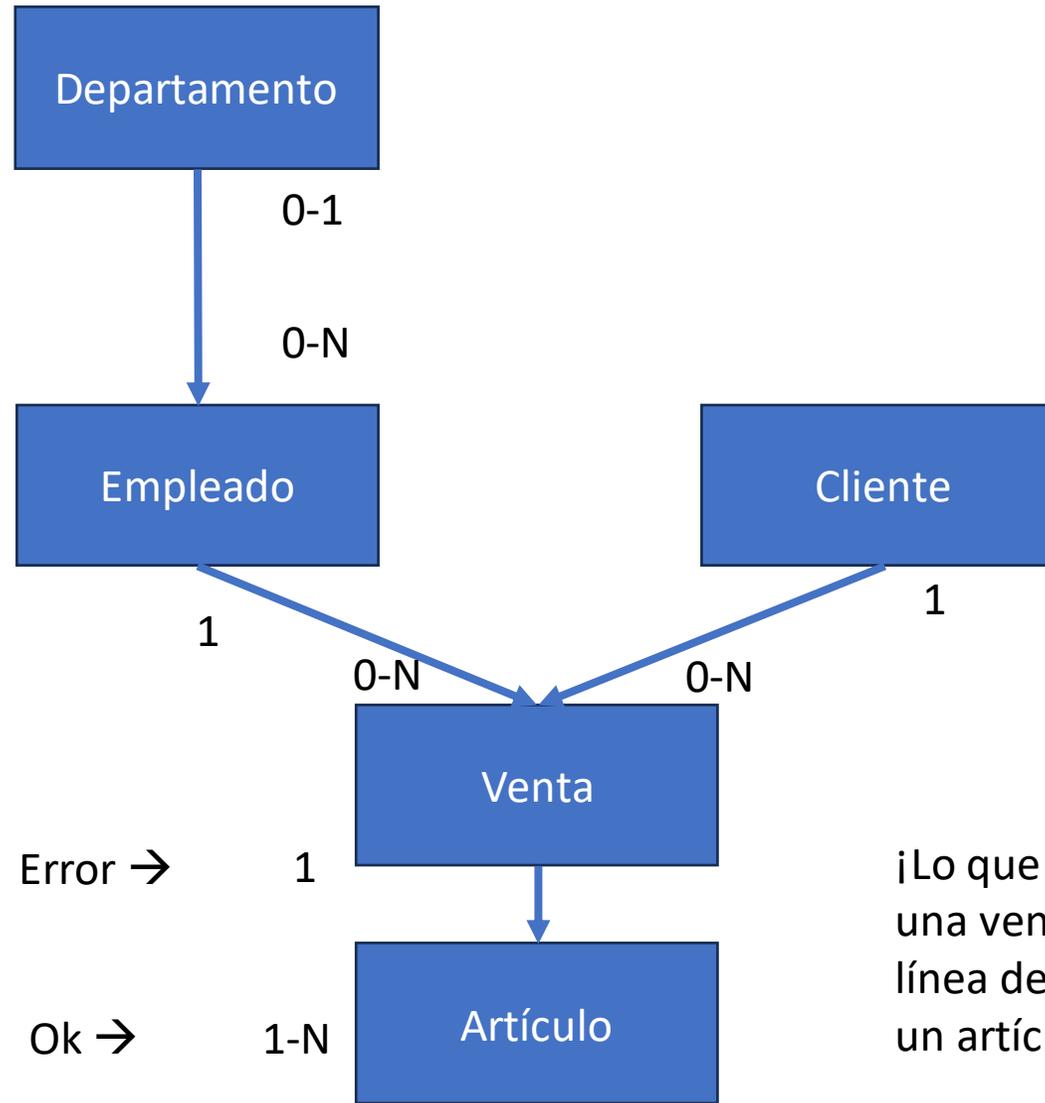
	Visión a alto nivel	Visión detallada
		
		
		
		

Visión a alto nivel y detallada de las relaciones

“Uno” =
Ninguno o uno

“Muchos” =
Ninguno, uno o
varios

Visión a alto nivel	Visión detallada
 <p data-bbox="1016 562 1098 719">uno a uno</p>	 <p data-bbox="1735 562 1791 719">0-1 0-1</p>
 <p data-bbox="1016 809 1174 966">uno a muchos</p>	 <p data-bbox="1735 809 1862 966">1 / 0-1 0-N / 1-N</p>
 <p data-bbox="1016 1042 1174 1199">muchos a muchos</p>	 <p data-bbox="1735 1042 1913 1199">0-N / 1-N 0-N / 1-N</p>



¿Una venta tiene varios artículos, pero un artículo solo está en una venta?

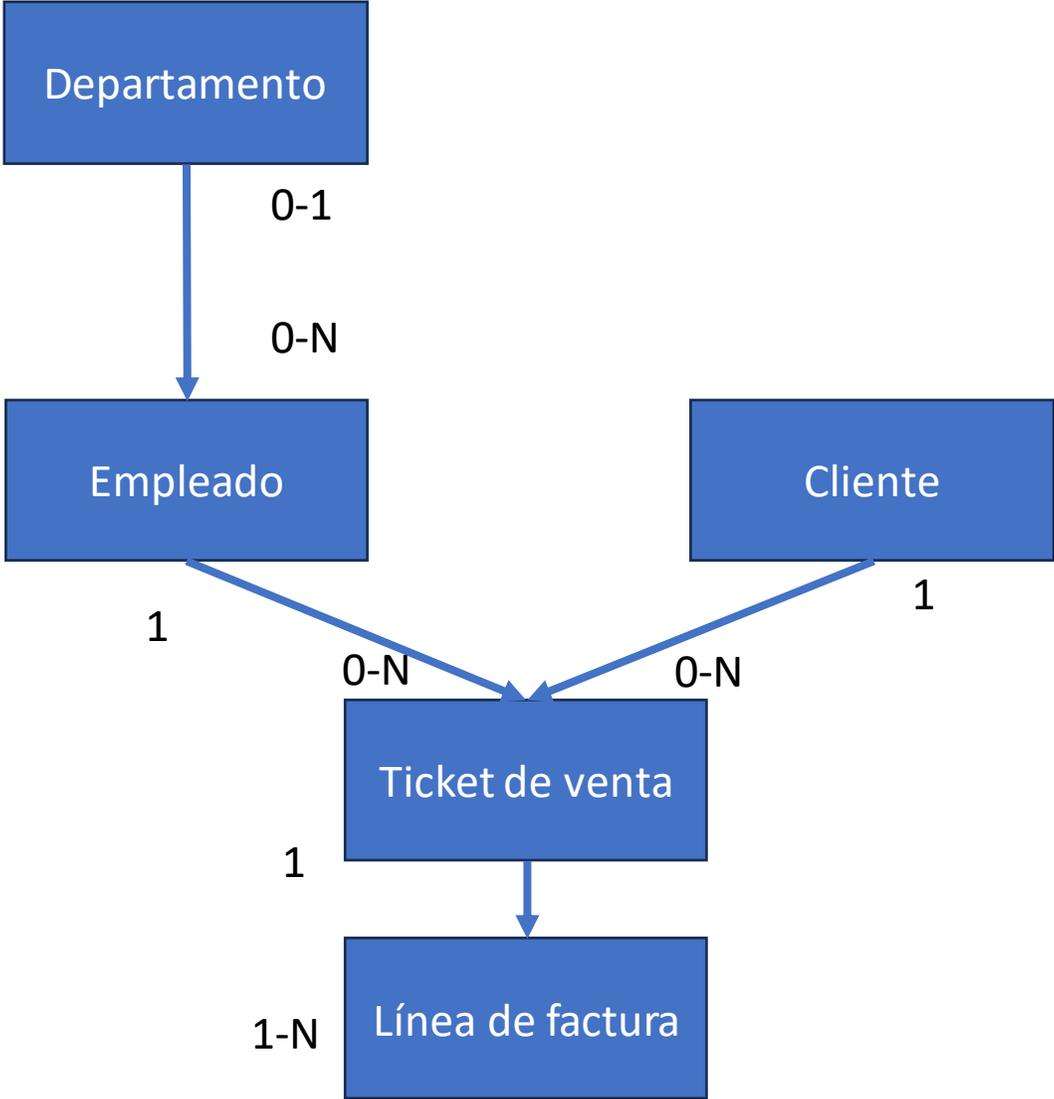
Error →

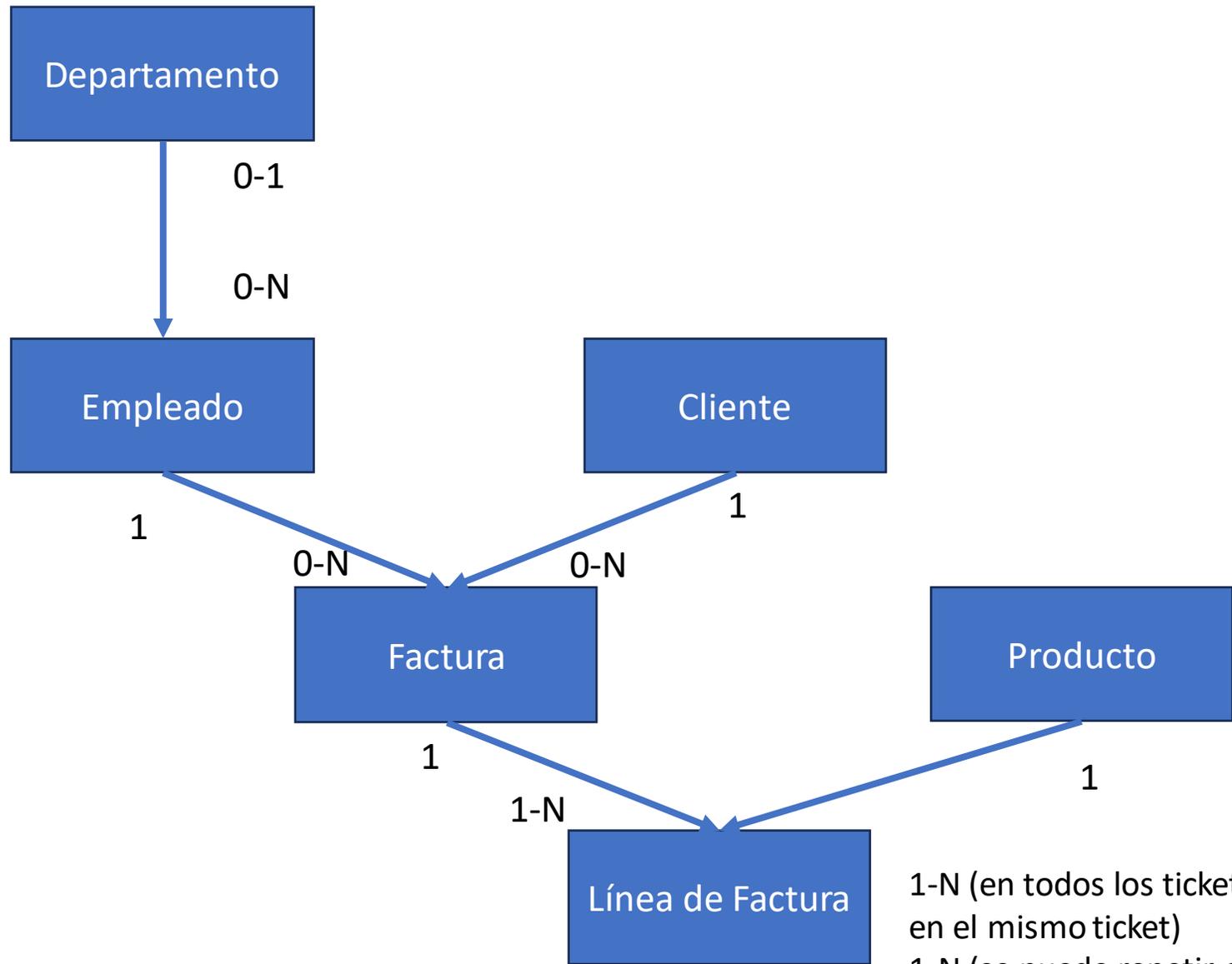
1

Ok →

1-N

¡Lo que solo está en una venta es una línea de factura, no un artículo!

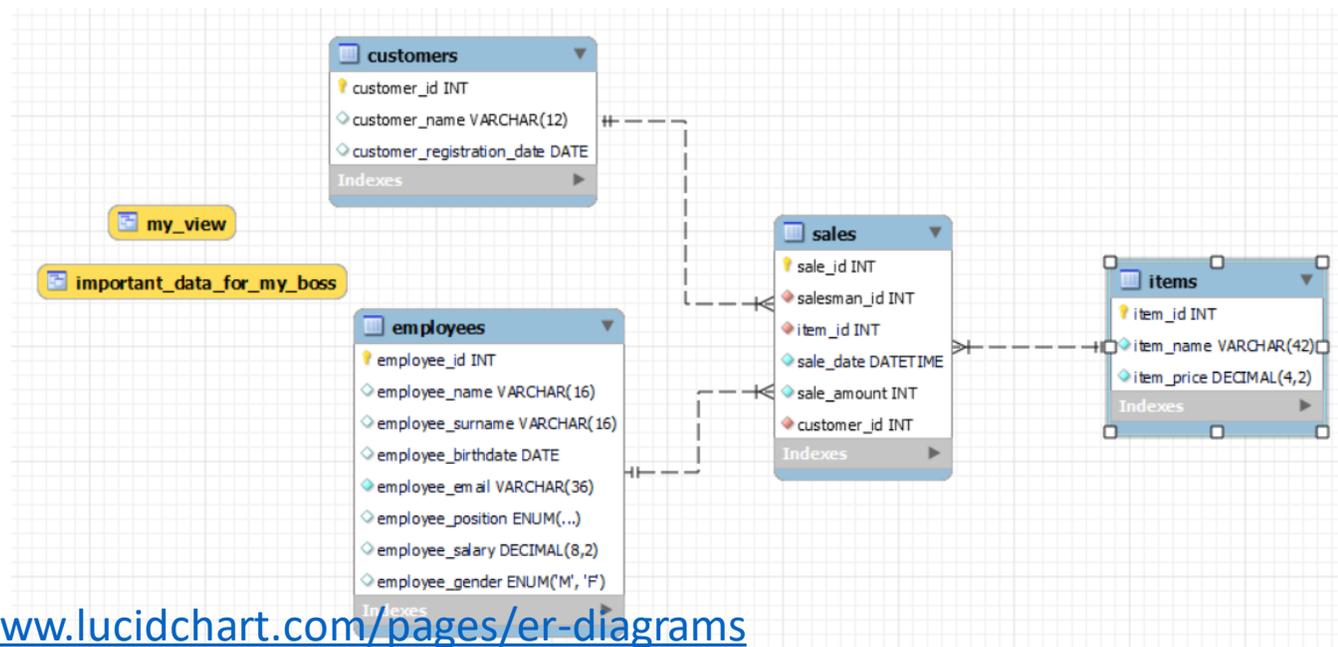




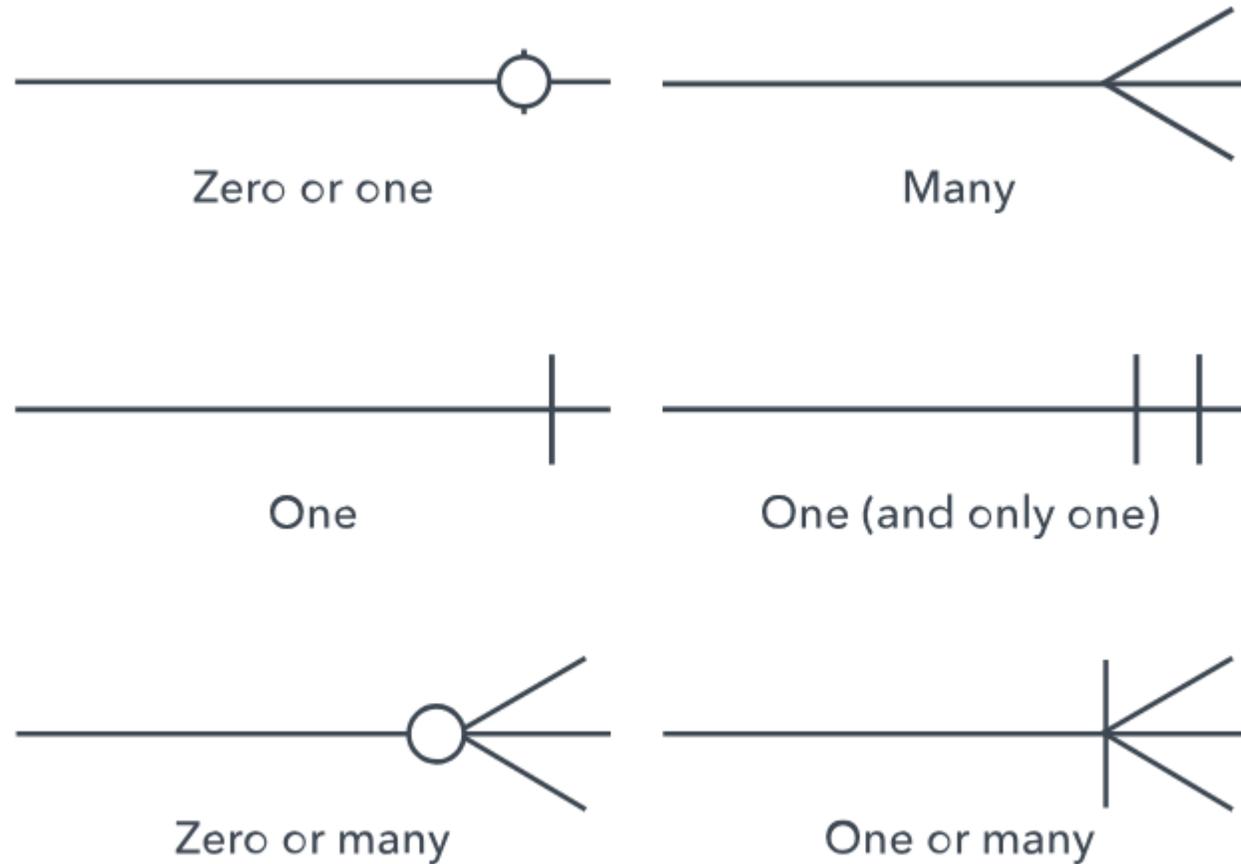
1-N (en todos los tickets. El producto no se repite en el mismo ticket)
 1-N (se puede repetir en el mismo ticket, es el caso de los supermercados)

Herramientas para crear Diagramas E-R y Diseños Relacionales

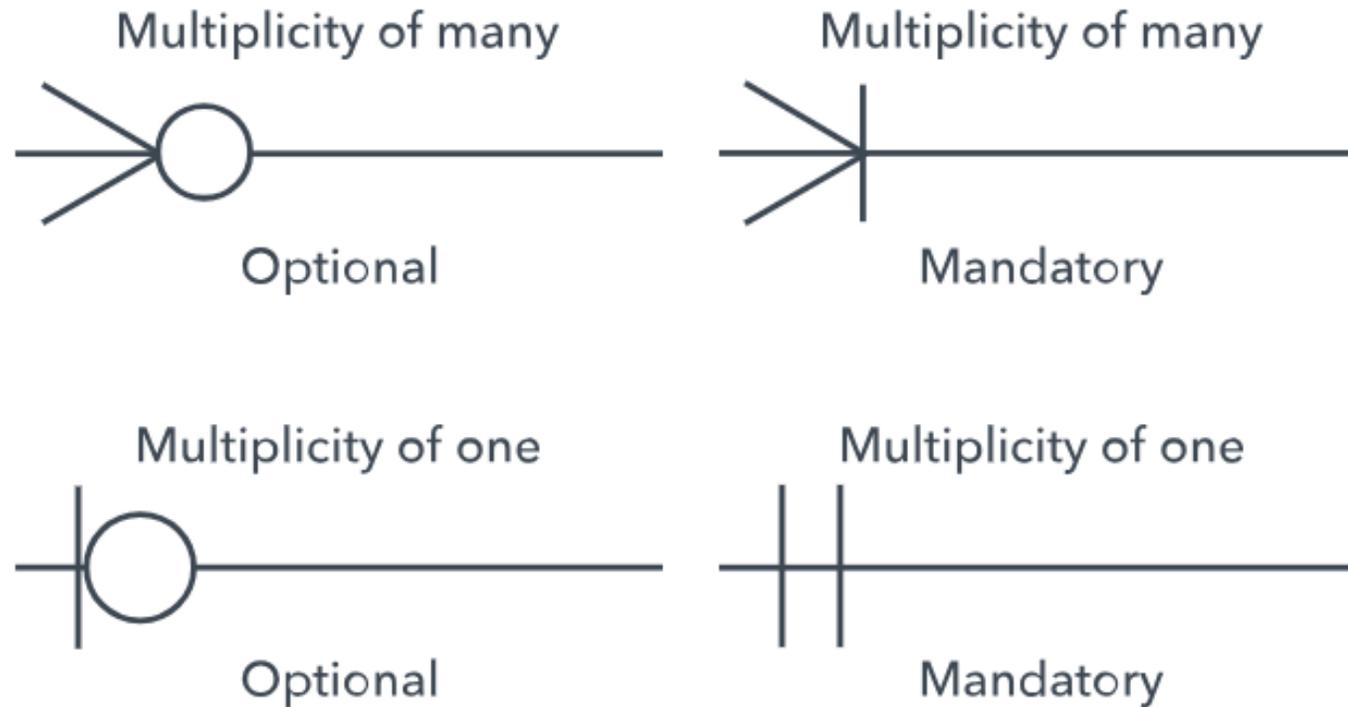
- Existen ciertos estándares para representar diagramas ER, sin embargo, cada DBMS (RDBMS) utiliza una notación ligeramente diferente. Veremos la notación utilizada por MySQL.



Diagramas E-R



Diagramas E-R



Diagramas E-R



one-to-many



one-to-one



mandatory relationship

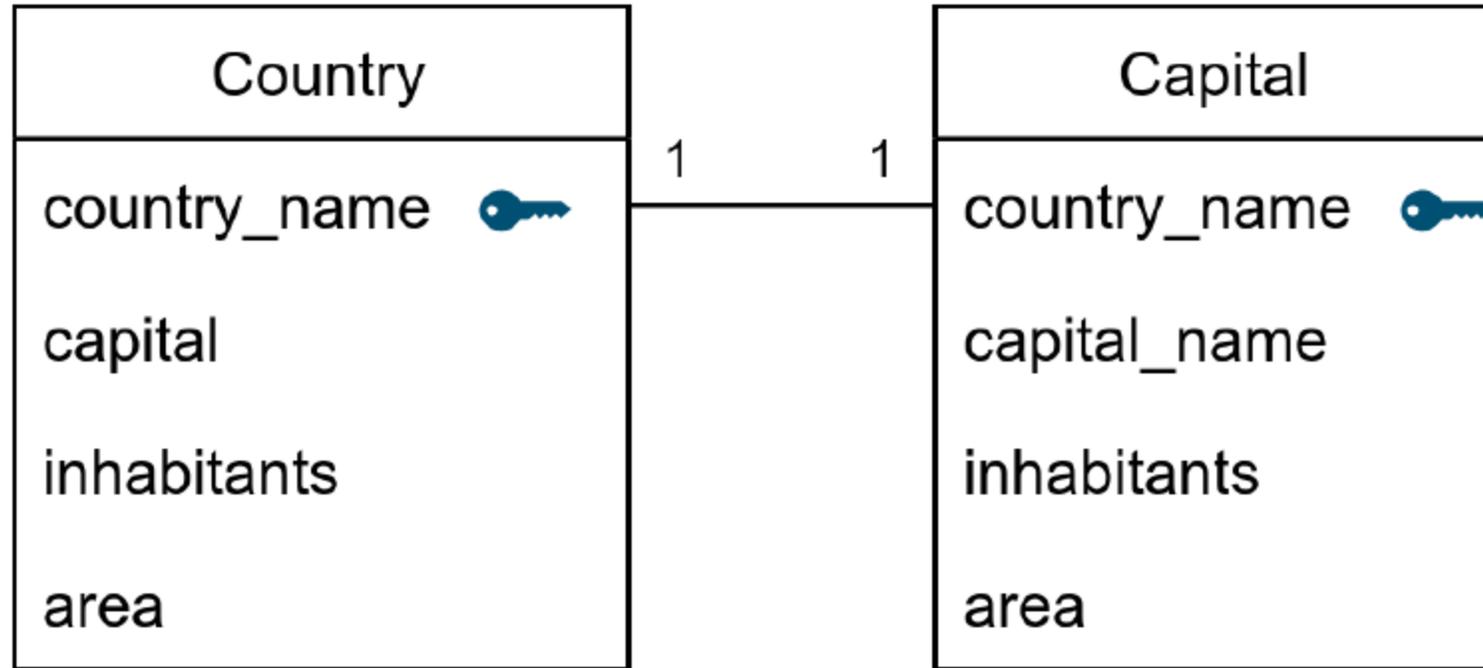


optional relationship

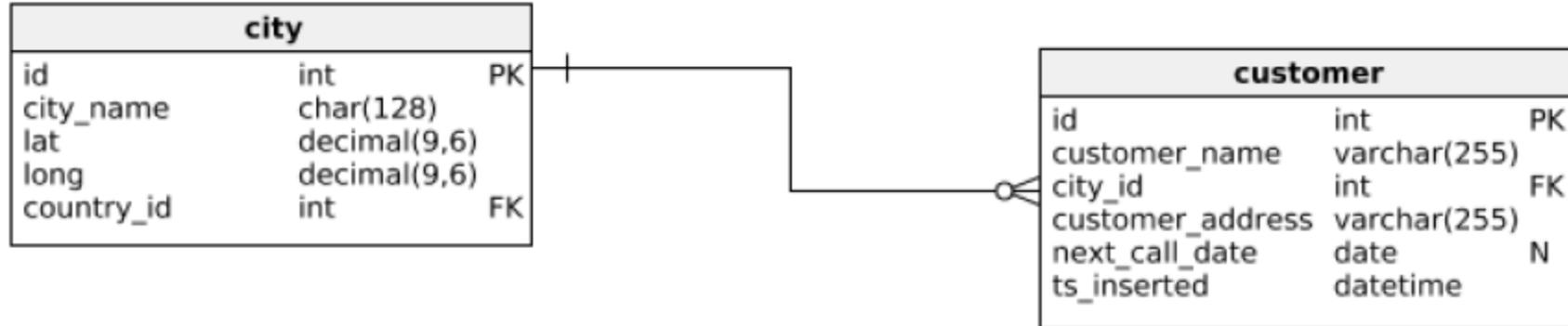


many-to-many

Relaciones uno a uno



Relaciones uno a muchos



Relaciones

- Uno a uno
 - Uno a uno o ninguno
- Uno a muchos
 - Uno a ninguno, uno o varios
- Muchos a muchos

Relaciones

- “Uno” = Ninguno o uno
- “Muchos” = Ninguno, uno o varios

Arquetipos de relaciones

- 1-1: Empleado y Móvil de empresa
 - Un empleado solo puede tener un móvil
 - Un móvil solo puede ser asignado a un empleado
- 1-N: Departamento y Empleado
 - Un departamento puede tener varios empleados
 - Un empleado sólo puede pertenecer a un departamento
- N-M: Cliente y Producto
 - Un cliente puede comprar varios productos
 - Un producto puede ser comprado por varios clientes

Relación 1-N

- “Uno a muchos” (o “uno a N”)
 - Uno de la tabla superior está relacionado con ninguno, uno o varios de la tabla inferior
 - Uno de la tabla inferior está relacionado con uno de la tabla superior

Relación N-M

- **Las relaciones N-M necesitan tres tablas**
- Una única relación N-M entre dos tablas A y B se puede representar, añadiendo una tercera tabla C, como dos relaciones 1-N entre las tablas A y C; y B y C.
- Por ejemplo: Un cliente puede comprar varios productos, y un producto puede ser comprado por varios clientes. Esta relación N-M necesita de tres tablas: Cliente, Producto y Venta

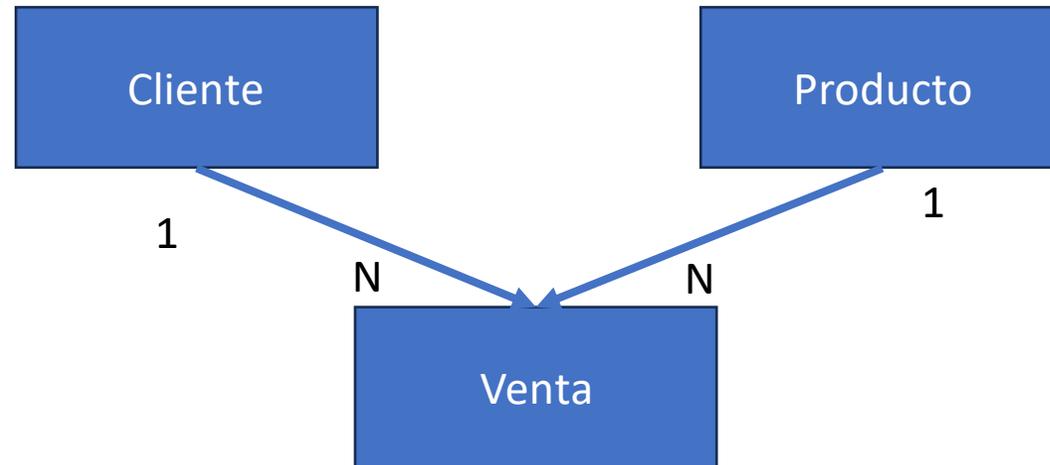
Pasar del diseño relacional al diagrama E/R

- Cuando una única relación la analizamos en los dos sentidos y nos sale 1-N y 1-N, es una N-M
- Cuando una única relación la analizamos en los dos sentidos y nos sale 1-N y 1-1, es una 1-N

Relaciones muchos a muchos (Diagrama “Entidad – Relación”)



Relaciones muchos a muchos (Diseño relacional)



Significado de “N”

Ambigüedad:

- N: 0, 1, 2, 3, ...
 - N: 1, 2, ...
 - N: 2, 3, ...
-
- “N” puede significar “0, 1, 2, 3, ...” o puede significar “1, 2, ...” o puede significar “2, 3, ...”

Significado de “1”

Ambigüedad:

- 1: 1
- 1: 0, 1

- “1” puede significar “uno” o puede significar “uno o ninguno”

Relaciones 1 a 1

Empleado

- Id
- Nombre
- Teléfono personal
- Email personal
- Email corporativo
- Móvil de empresa

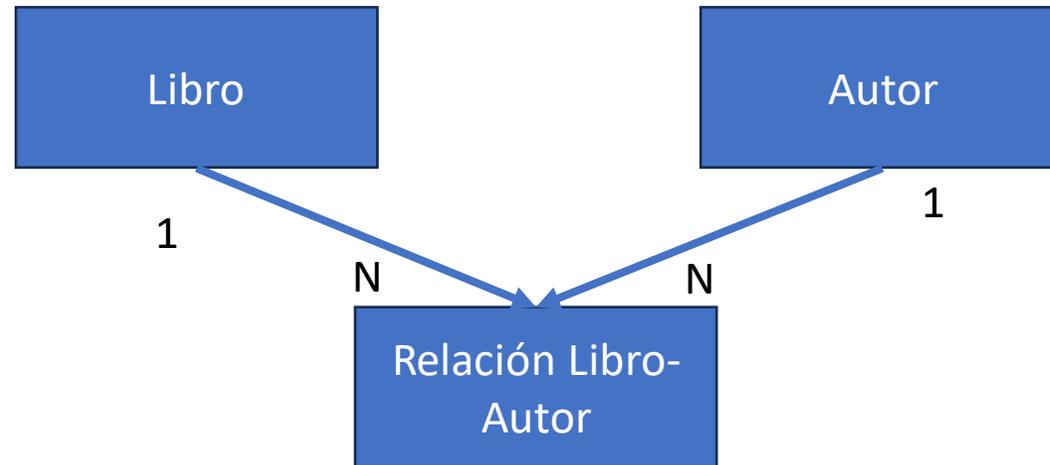
Relaciones 1 a 1



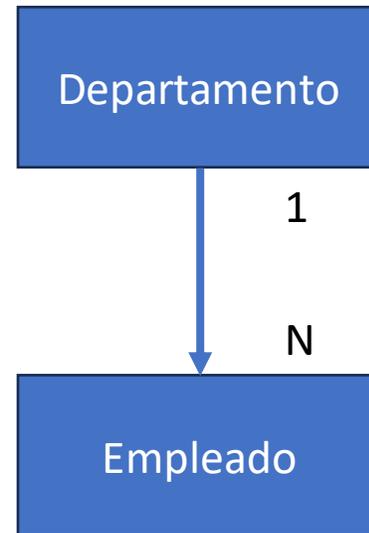
Relaciones muchos a muchos



Relaciones muchos a muchos



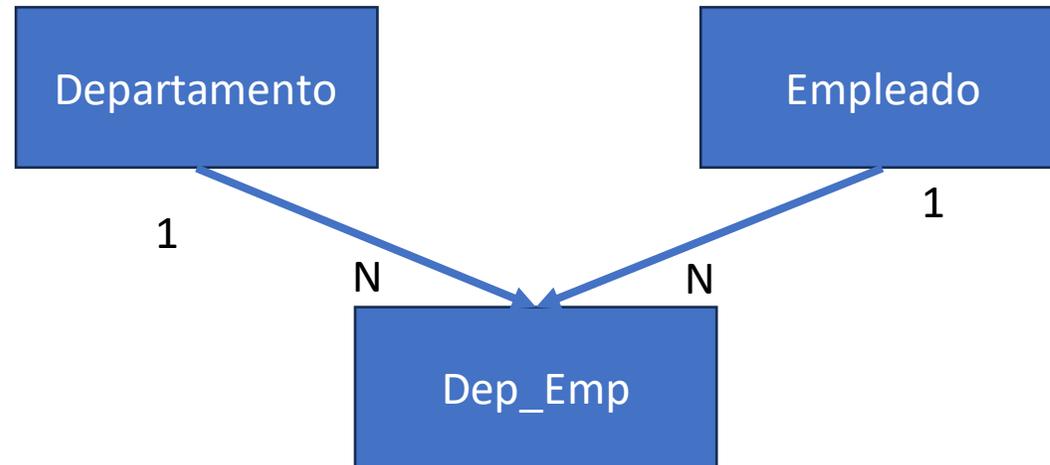
Relaciones muchos a muchos



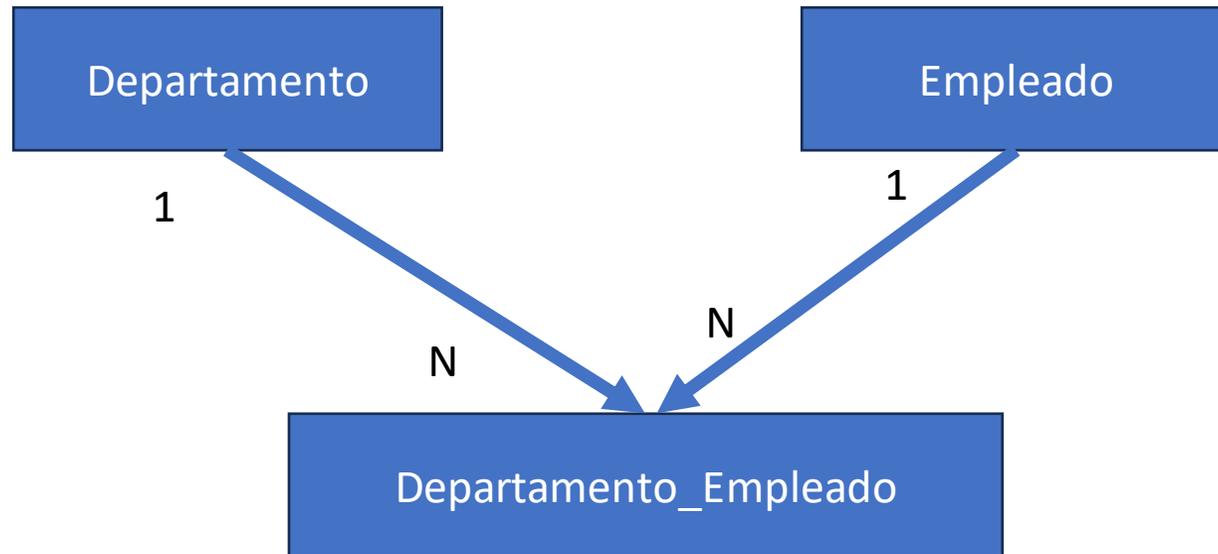
Relaciones muchos a muchos



Relaciones muchos a muchos



Implementación de una relación N - M



Implementación de una relación muchos a muchos

id	nombre
1	Título 1
2	Título 2
3	Título 3
4	Título 4
5	Título 5
6	Título 6
7	Título 7
8	Título 8

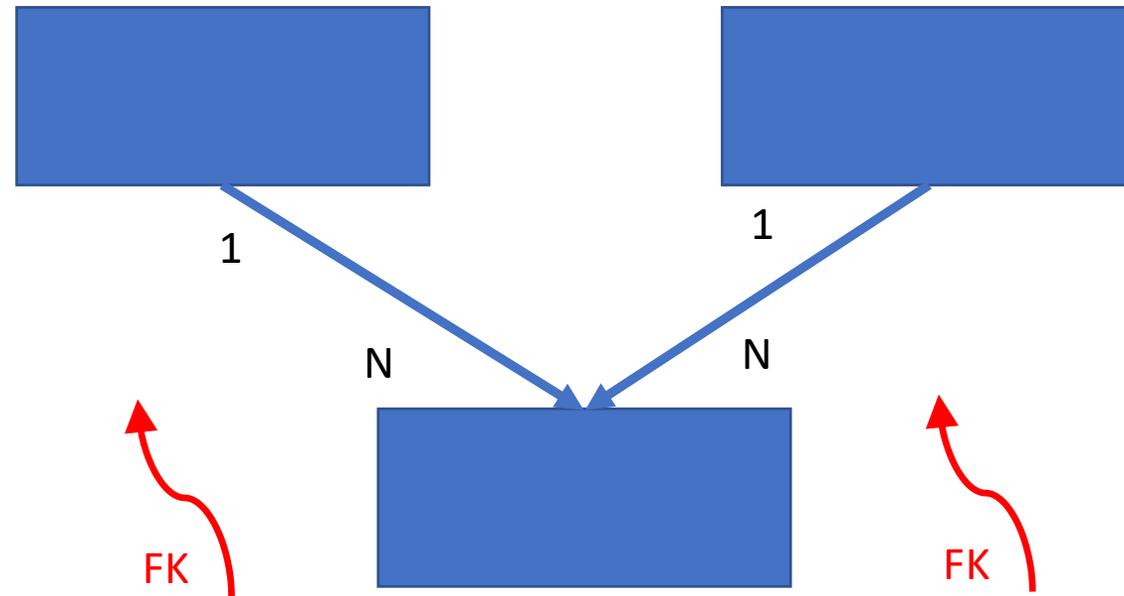
id	nombre
1	Marko
2	Patricia
3	Suzy
4	Miriam
5	Leo
6	Germán
7	Alejandra
8	Carlos
9	Mónica

Primary Key			
id_libro	id_autor	orden de autoría	
1	1	1	Un autor que ha escrito dos
2	1	1	
3	1	1	
4	5	1	Un libro con dos autores
4	2	2	
5	6	1	
6	7	1	
7	7	1	
8	6	1	Un libro con dos autores
8	8	2	
9	8	1	

Implementación de una relación muchos a muchos



Implementación de una relación muchos a muchos



Gracias

Manu Herrán

manuherran.com