



CONTENIDO

**Módulo IV. Modelo Relacional..... 2**

4.1    Modelo Relacional ..... 3

    4.1.1    Introducción al Modelo Relacional ..... 3

    4.1.2    Elementos del Modelo ..... 4

    4.1.3    Reglas de Integridad..... 6

4.2    Obtención del Modelo Lógico a partir del Modelo Conceptual ..... 10

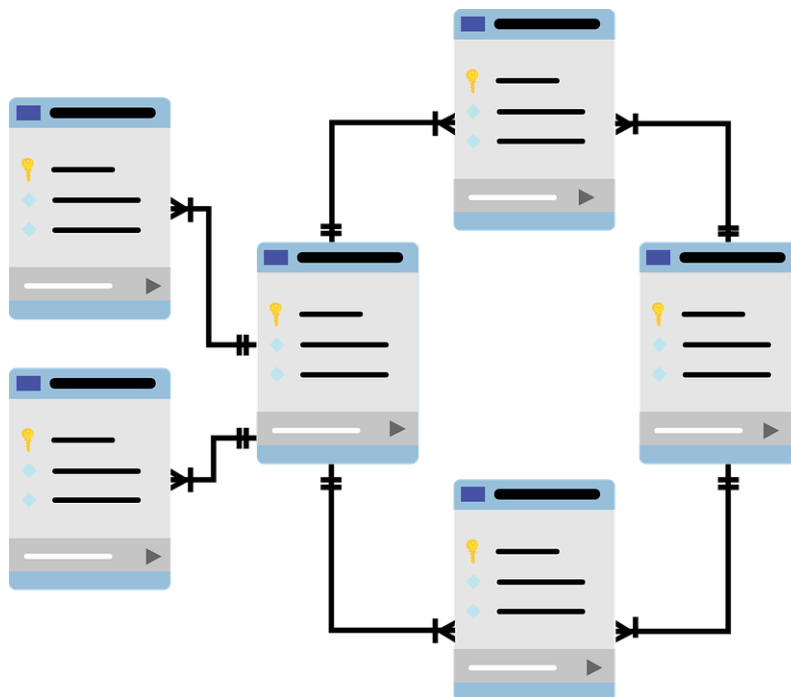
    4.2.1    Transformación del Esquema Conceptual al Lógico ..... 11

    4.2.2    Reglas de Transformación..... 11





## Módulo IV. Modelo Relacional



Modelo Relacional de Base de Datos  
Pixabay License

El modelo relacional es uno de los modelos más utilizados en los sistemas de gestión de base de datos en la actualidad y establece un conjunto de relaciones con sus correspondientes restricciones de integridad. Cada una de las relaciones dispone de un conjunto de datos definidos y debe establecer las llaves primarias y secundarias (llamadas foráneas) que permiten el movimiento entre las tablas o relaciones que componen dicho modelo.

En esta unidad se presentan algunos conceptos fundamentales del modelo relacional, así como el proceso de conversión que va desde el modelo entidad relación al modelo relacional resultante.



## **4.1 Modelo Relacional**

De acuerdo con la empresa Oracle, una base de datos relacional es un tipo de base de datos que almacena y proporciona acceso a puntos de datos relacionados entre sí. Las bases de datos relacionales se basan en el modelo relacional, una forma intuitiva y directa de representar datos en tablas.

En una base de datos relacional, cada fila de la tabla es un registro con un ID único llamado clave. Las columnas de la tabla contienen atributos de los datos, y cada registro generalmente tiene un valor para cada atributo, lo que facilita el establecimiento de las relaciones entre los puntos de datos.

### **4.1.1 Introducción al Modelo Relacional**

El modelo relacional lo introdujo el Dr. Edward Codd, en el año 1970 y es el modelo mayormente empleado en los Sistemas de Gestión de Base de Datos modernos.

El modelo de Codd persigue los siguientes objetivos:

- a. Independencia física de los datos, el modo de almacenamiento de los datos no debe influir en su manipulación lógica.
- b. Independencia lógica de los datos, los cambios que se realicen en los objetos de la base de datos no deben repercutir en los programas y usuarios que acceden a ella.
- c. Flexibilidad, para presentar a los usuarios los datos de la forma más adecuada.
- d. Uniformidad, en la presentación de la lógica de los datos, que son tablas, lo que facilita la manipulación de la base de datos por parte de los usuarios.



- e. Sencillez, este modelo es fácil de comprender y utilizar por el usuario.

Para conseguir estos objetivos Codd introduce el concepto de relación (tablas) como estructura básica del modelo. Todos los datos de una base de datos se representan en forma de relaciones cuyo contenido varía en el tiempo.

#### **4.1.2 Elementos del Modelo**

##### **Relación**

La RELACION es el único elemento utilizado para representar tanto entidades como asociaciones entre ellas y se representa como una tabla.

Al número de tuplas de una relación se le conoce como cardinalidad o grado de una tabla.

El esquema de una tabla R se denota  $(A_1, A_2, \dots, A_n)$  donde R es el nombre de la relación y  $(A_1, A_2, \dots, A_n)$  son los atributos.

Ejemplo: EMPLEADO (Cédula, Nombre, Sexo, Edad).

##### **Llaves**

###### *Llave Primaria*

- Una llave primaria (PK por sus siglas en inglés) es una columna o conjunto de columnas que identifican en forma única cada fila en una tabla o relación.
- Cada tabla debe tener una llave primaria, y la llave primaria debe ser única.
- Ninguna parte de la llave primaria debe ser nula; es decir su contenido no puede ser NULL.



↓

Llave Primaria			
No. De Empleado	Apellido	Nombre	No. Depto.
100	González	Melisa	10
310	Pérez	Luis	15
210	Sánchez	José	05
405	Espinosa	Lilibeth	15

### Llave Compuesta

Es una llave primaria que consiste en múltiples columnas.

Llave Primaria			
No de Sucursal	No. De Cuenta	Balance	Apertura
5	723525	12,000.00	21-oct-2014
10	723525	100.00	25-Sep-2005
3	253311	15,000.00	17-Dic-95
10	723528	2,000.00	20-Ago-2013

Cada fila en la tabla es identificada en forma única por los valores de No. Sucursal y No. Cuenta. Juntas son la llave primaria.

### Llave Foránea

Una llave foránea (FK, por sus siglas en inglés) es una columna, o combinación de columnas, en una tabla, que hacen referencia a una llave primaria en otra tabla.

Son usadas para unir tablas.



LLAVE FORÁNEA

↓

**Tabla Empleado**

No Empleado	Apellido	Nombre	No. Depto	No. Nómina
100	González	Melisa	10	9710
310	Pérez	Luis	15	8730
210	González	José	05	9730
405	Espinosa	Lilibeth	20	1157

Llave primaria ↓

**Tabla Departamento**

No. Depto	Nombre -Depto
10	Finanzas
15	Operaciones
05	Ventas
20	Manufactura

#### 4.1.3 Reglas de Integridad

Son reglas que deben ser cumplidas al momento de la implementación. Estas reglas de integridad están usualmente asociadas al sistema de gestión de base de datos y son de mucha importancia en el proceso de implementación, en particular para la comunicación entre las relaciones.

Las siguientes son algunos tipos de integridades:



- **Integridad de la ENTIDAD:** ningún atributo de una llave primaria puede tener valor nulo (NULL).
- **Integridad REFERENCIAL:** debido a que las llaves foráneas referencian llaves primarias, es esencial que el valor de una llave foránea coincida con el valor de alguna llave primaria.

A continuación, se asocian las reglas de integridad con el sistema de gestión Microsoft SQL Server. Pueden definirse en las instrucciones CREATE y ALTER TABLE:

- a. Datos requeridos
- b. Restricciones de dominio
- c. Integridad de entidades
- d. Integridad referencial

### **Restricción a los Datos (Datos Requeridos)**

NOT NULL

Permite que impongamos restricciones a los campos que obligatoriamente deben contener datos; por ejemplo, sucursal y nombre de una persona

Son definidos al momento de la creación de la tabla

NOMBRE\_SUCURSAL VARCHAR (12) NOT NULL

Nombre VARCHAR (10) NOT NULL

### **Restricciones de Dominio**

CHECK

Toda columna tiene un dominio, es decir un conjunto de valores legales; por ejemplo, el campo SEXO puede ser 'F' o 'M'.



Estas restricciones pueden ser definidas al momento de la creación de la tabla con la opción CHECK o haciendo un ALTER a una tabla creada.

sexo CHAR NOT NULL

CHECK (sexo in ('M','F'))

color\_bandera VARCHAR NOT NULL

CHECK (color\_bandera in ('rojo', 'azul'))

DEFAULT

Como su propio nombre indica, esta restricción introduce un valor por defecto en una columna cuando no se indica ningún valor para insertar. Con esta restricción aseguramos la integridad de dominio, ya que aseguramos valores válidos para nuevos registros que se inserten.

Precio money

set DEFAULT (0)

### **Restricción de Entidades**

Las llaves primarias que identifican a cada entidad deben ser definidas con un valor unívoco y no nulo en cada fila.

Por ejemplo, cada fila de la tabla empleado debe tener en el campo codigo\_empleado un único valor y nunca este valor podrá ser nulo.

PRIMARY KEY (codigo\_empleado)

En el caso de una llave compuesta:





PRIMARY KEY (no\_sucursal, no-cta)

La cláusula PRIMARY Key sólo puede especificarse una sola vez por tabla.

## UNIQUE

Esta restricción obliga a que todos los valores de una determinada columna no estén repetidos en otros registros.

Si tenemos varias restricciones UNIQUE en una misma tabla, todas deben ser cumplidas a la vez para cada registro.

## Integridad Referencial

Las llaves foráneas enlazan cada fila de la tabla hijo (es decir la tabla que la contiene) con la fila de la tabla padre donde está la misma como llave primaria.

La integridad referencial quiere decir que si la llave foránea (externa) contiene un valor, dicho valor debe hacer referencia a una fila existente y válida dentro de la tabla padre.

El estándar ISO de SQL permite la definición de llaves primarias con la cláusula FOREIGN KEY en las instrucciones CREATE y ALTER TABLE.

FOREIGN KEY (Codigo\_DEPTO) references Tabla\_EMPLEADO

SQL rechaza cualquier intento de efectuar una operación INSERT o UPDATE a una tabla hijo cuando dicho valor a insertar no existe en la tabla padre.

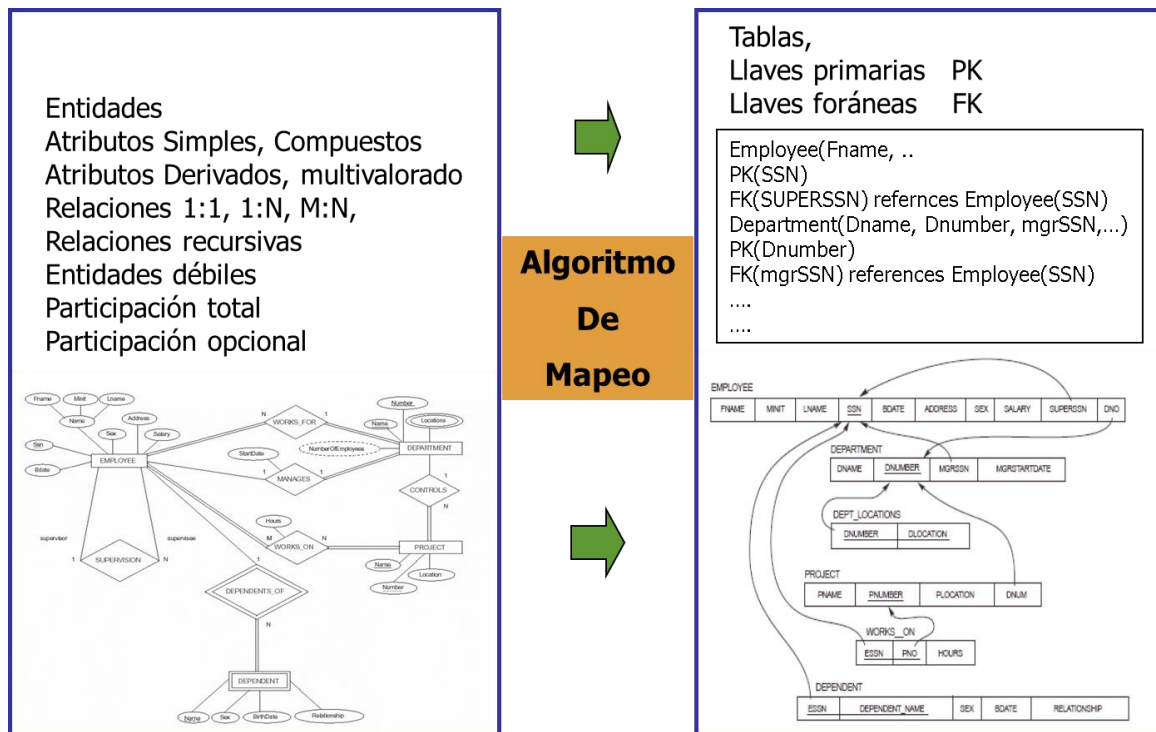
En el caso del borrado o actualización de un padre que tenga varios hijos, el SQL soporta su manejo de cuatro posibles formas:

- **CASCADE:** Borra o actualiza la fila de la tabla padre y borra o actualiza automáticamente todas las filas correspondientes en la tabla hijo (lo hace en cascada)



- SET NULL: Borra o actualiza la fila del padre y se asigna el valor NULL a las filas correspondientes en la tabla hijo
- SET DEFAULT: Borra o actualiza la fila en la tabla padre y se asigna a cada campo de la tabla hijo (llave foránea) el valor especificado como default.
- NO ACTION: Se rechaza la operación de borrado o actualización de la tabla padre. Esta es la opción predeterminada si no se especifica una regla ON DELETE

## 4.2 Obtención del Modelo Lógico a partir del Modelo Conceptual

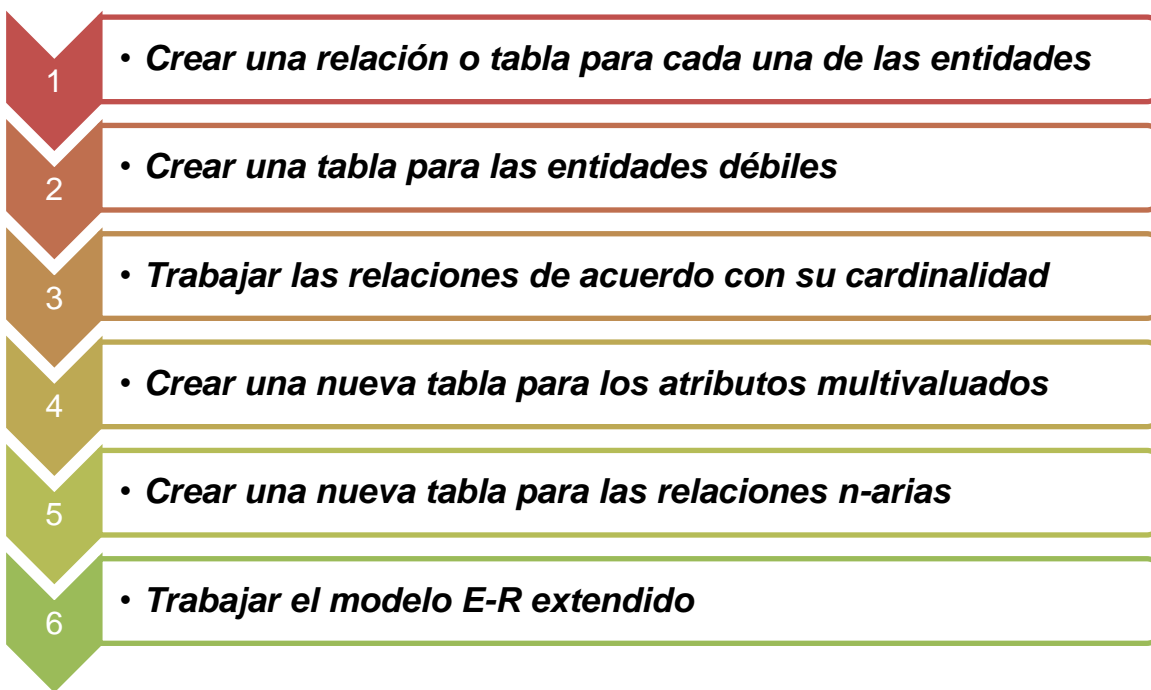


Para obtener el modelo lógico o de tablas usando el modelo conceptual (Entidad Relación), se necesita seguir un algoritmo de mapeo que indica el orden de conversión y creación de las relaciones involucradas.



#### 4.2.1 Transformación del Esquema Conceptual al Lógico

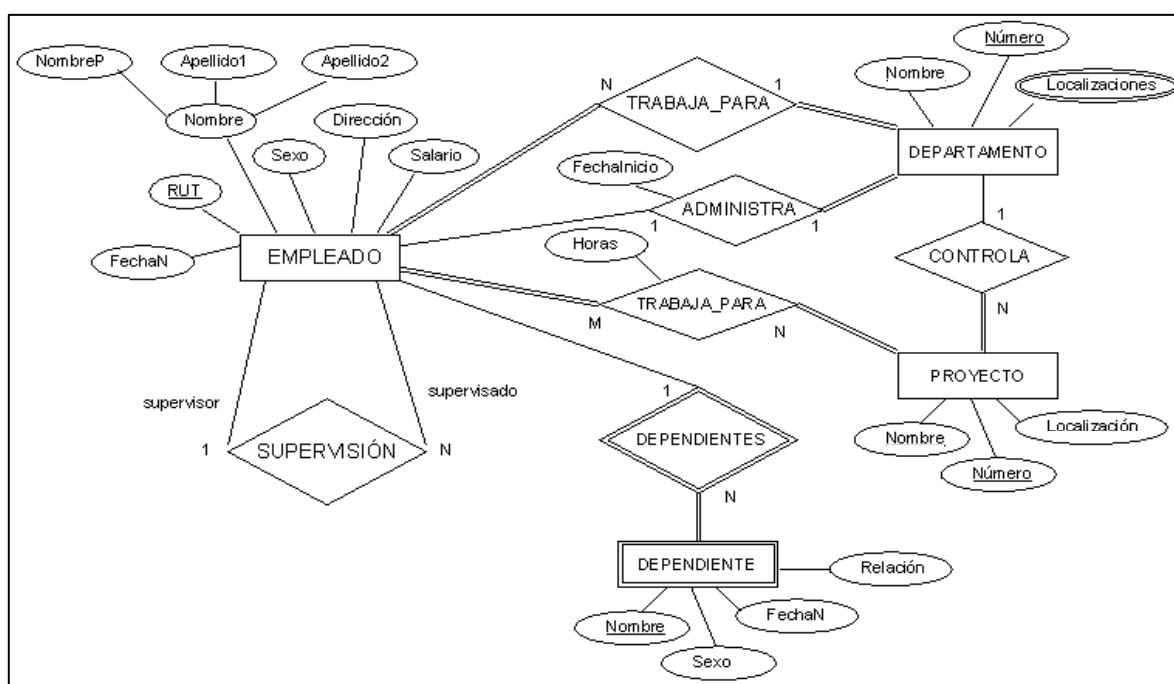
El proceso de transformación del modelo conceptual (modelo entidad relación) al modelo lógico (modelo relacional) sigue un conjunto de pasos que se ilustran en el siguiente diagrama:



#### 4.2.2 Reglas de Transformación

En la presente sección se explica y ejemplifica el Algoritmo de Transformación de ER en Relacional que son las reglas de transformación. Estas reglas de transformación son una propuesta de Elmasri y Navathe en la obra Fundamentos de Sistemas de Información.

Se utilizará un modelo conceptual que corresponde a la operación de una empresa con sus respectivos componentes.



### Paso 1

- Por cada tipo normal (fuerte) de entidades E del esquema ER, se crea una relación R que contenga todos los atributos simples de E.
- Se incluyen sólo los atributos simples componentes de un atributo compuesto.
- Se elige uno de los atributos clave de E como llave primaria de R.
- Si la llave elegida es compuesta, el conjunto de atributos simples que la forman constituirá la clave primaria de R.

***Cree una relación o tabla para cada una de las entidades y agregue sus atributos.***



Se crean las siguientes relaciones (observe que se dejan los paréntesis de cierre abiertos para trabajar más adelante si se hace necesario):

- EMPLEADO (RUT, NombreP, Apellido1, Apellido2, FechaN, Sexo, Dirección, Salario
- DEPARTAMENTO (NúmeroDepto, Nombre
- PROYECTO (NúmeroP, Nombre, Localización

## Paso 2

- Por cada tipo de entidad débil W en el esquema ER con tipo de entidad propietario E, se crea una relación R, y se incluyen todos los atributos simples (o componentes simples de los atributos compuestos) de W como atributos de R.
- Además, se incluyen como atributos de clave externa o foránea de R el atributo o atributos de clave primaria de la relación o relaciones que corresponden al tipo o tipos de entidades propietarias.
- La llave primaria de R es la combinación de las claves primarias de los propietarios y la clave parcial del tipo de entidad débil de W, si la hubiese.
- ***Cree una tabla para las entidades débiles y agregue sus atributos junto a la llave de la relación o tabla a la que está asociada como entidad fuerte. La llave es la unión de la llave primaria en la entidad fuerte con la llave de la entidad débil.***

Continuando con el ejemplo:

- EMPLEADO (RUT, NombreP, Apellido1, Apellido2, FechaN, Sexo, Dirección, Salario
- DEPARTAMENTO (NúmeroDepto, Nombre
- PROYECTO (NúmeroP, Nombre, Localización



- DEPENDIENTE (Nombre, RUT, Sexo, FechaN, Relación

### Paso 3

- Por cada tipo de relación binaria 1:1 R del esquema ER, se identifican las relaciones S y T que corresponden a los tipos de entidades que participan en R.
- Se elige una de las relaciones, por ejemplo, S, y se incluye como llave foránea en S la llave primaria de T.
- Se incluyen todos los atributos o componentes simples de los atributos compuestos del tipo de relación 1:1 como atributos de S.
- ***Agregue la llave primaria de una de las entidades a la otra cuando la relación sea de 1:1. Puede usar cualquiera de las 2 entidades.***

Continuando con el ejemplo:

- EMPLEADO (RUT, NombreP, Apellido1, Apellido2, FechaN, Sexo, Dirección, Salario
- DEPARTAMENTO (NúmeroDepto, Nombre, RUT\_Jefe, FechaInicio
- PROYECTO (NúmeroP, Nombre, Localización
- DEPENDIENTE (Nombre, RUT, Sexo, FechaN, Relación

### Paso 4

- Para cada tipo de relación binaria 1:N R, se identifica la relación S que representa el tipo de entidad participante del lado N del tipo de relación.
- Se incluye como llave foránea externa en S la clave primaria de la relación T que representa el otro tipo de entidad que participa en R.
- ***En otras palabras, la llave primaria del lado 1 pasa al lado muchos.***



Siguiendo el ejemplo:

- EMPLEADO (RUT, NombreP, Apellido1, Apellido2, FechaN, Sexo, Dirección, Salario, NúmeroDepto, RUT\_Sup
- DEPARTAMENTO (NúmeroDepto, Nombre, RUT\_Jefe, FechaInicio
- PROYECTO (NúmeroP, Nombre, Localización, NúmeroDepto
- DEPENDIENTE (Nombre, RUT, Sexo, FechaN, Relación

### Paso 5

- Por cada tipo de relación binaria N:M R, se crea una nueva relación S para representar a R.
- Se incluyen como atributos de llave foránea en S las llaves primarias de las relaciones que representan los tipos de entidades participantes; su combinación será la llave primaria de S.
- Se incluyen los atributos o componentes simples de atributos compuestos de la relación como atributos de S.
- ***Cree una nueva tabla para la relación N:M cuya llave es la combinación de las llaves de las tablas que relaciona. Agregue los atributos de la relación.***

Siguiendo con el ejemplo de la empresa:

- EMPLEADO (RUT, NombreP, Apellido1, Apellido2, FechaN, Sexo, Dirección, Salario, NúmeroDepto, RUT\_Sup
- DEPARTAMENTO (NúmeroDepto, Nombre, RUT\_Jefe, FechaInicio
- PROYECTO (NúmeroP, Nombre, Localización, NúmeroDepto
- DEPENDIENTE (Nombre, RUT, Sexo, FechaN, Relación
- TRABAJA\_PARA (RUT, NúmeroP, Horas



### Paso 6

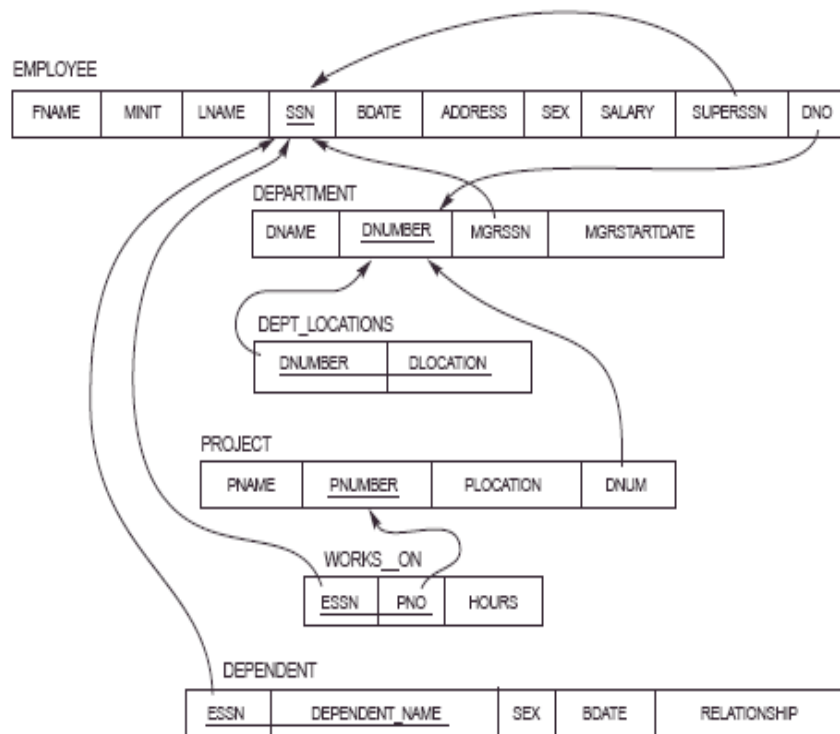
- Por cada atributo multivaluado A, se crea una nueva relación R.
- Esta relación R incluirá un atributo correspondiente a A, más el atributo de llave primaria K (como clave externa en R) de la relación que representa el tipo de entidades o tipo de relaciones que tiene A como atributo.
- La clave primaria de R es la combinación de A y K.
- Si el atributo multivaluado es compuesto, se incluyen sus componentes simples.
- ***Cree una nueva tabla para los atributos multivaluados. La llave será la llave primaria de la entidad donde se encuentra el atributo multivaluado más el atributo multivaluado.***

En nuestro ejemplo se tiene lo siguiente:

- EMPLEADO (RUT, NombreP, Apellido1, Apellido2, FechaN, Sexo, Dirección, Salario, NúmeroDepto, RUTSup)
- DEPARTAMENTO (NúmeroDepto, Nombre, RUT\_Jefe, FechaInicio)
- PROYECTO (NúmeroP, Nombre, Localización, NúmeroDepto)
- DEPENDIENTE (Nombre, RUT, Sexo, FechaN, Relación)
- TRABAJA\_PARA (RUT, NúmeroP, Horas)
- LOCALIZACIONES\_DEPTO (NúmeroDepto, Localizaciones)

De forma gráfica los primeros seis pasos nos permite tener un modelo relacional como el que se muestra en la siguiente imagen:





Ahora se presentan los pasos que continúan el algoritmo.

### Paso 7

- Por cada tipo de relaciones n-arias  $R$ ,  $n > 2$ , se crea una nueva relación  $S$  que represente a  $R$ .
- Se incluyen como atributos de llave foránea en  $S$  las claves primarias de las relaciones que representan los tipos de entidades participantes.
- También se incluyen los atributos simples y los componentes simples de los compuestos como atributos de  $S$ .
- La llave primaria de  $S$  casi siempre es una combinación de todas las llaves externas que participan.



- Cree una nueva tabla para las relaciones n-arias con llave primaria compuesta (atributos llaves de cada relación que participa).

## Paso 8

Convertir cada especialización con m subclases  $\{S_1, S_2, \dots, S_m\}$  y (generalizadas en la) superclase C, donde los atributos de C son  $\{k, a_1, \dots, a_n\}$  y k es la llave primaria, en esquemas de relación, empleando una de las siguientes cuatro opciones. CP representa la llave o clave primaria.

### Opción 8A:

- Crear una relación L para C con atributos  $Atrs(L) = \{k, a_1, \dots, a_n\}$  y  $CP(L) = k$ . Crear una relación  $L_i$ , por cada subclase  $S_i$ ,  $1 \leq i \leq m$ , con los atributos  $Atrs(L_i) = \{k\} \cup \{\text{atributos de } S_i\}$  y  $CP(L_i) = k$
- Se crea una relación para la superclase y una relación para cada una de las subclases. Las subclases tendrán el atributo llave de la superclase.

### Opción 8A:

#### EMPLEADO

<u>NSS</u>	NompreP	Apellido	Dirección	FechaN	TipoTrabajo
------------	---------	----------	-----------	--------	-------------

#### SECRETARIO

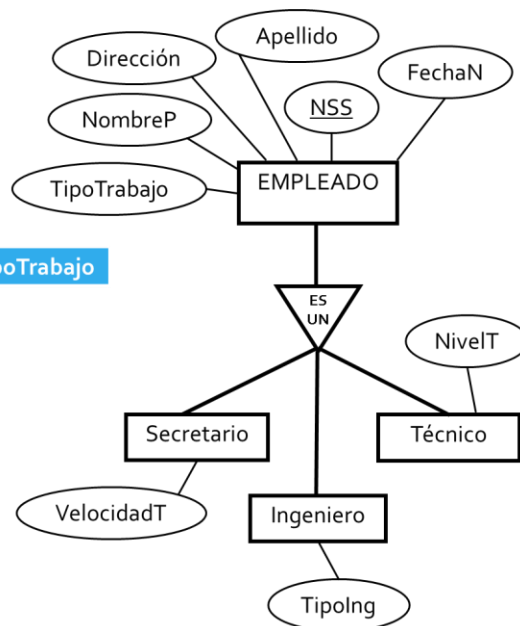
<u>NSS</u>	VelocidadT
------------	------------

#### TÉCNICO

<u>NSS</u>	NivelT
------------	--------

#### INGENIERO

<u>NSS</u>	Tipolng
------------	---------

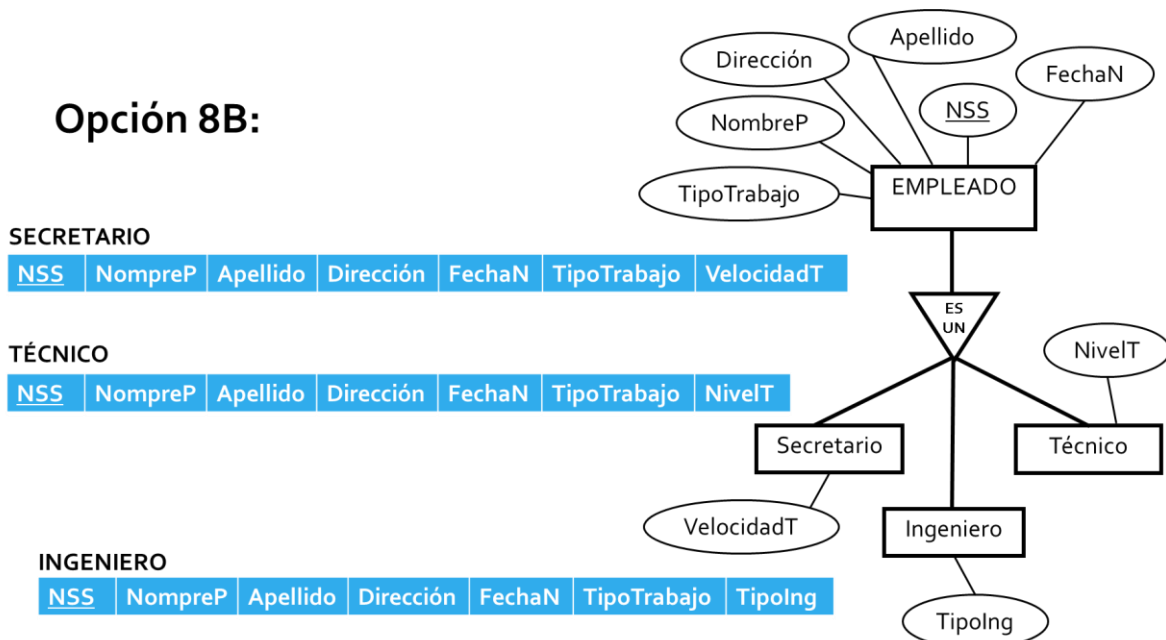




### Opción 8B:

- Crear una relación  $L_i$ , por cada subclase  $S_i$ ,  $1 \leq i \leq m$ , con los atributos  $Atrs(L_i) = \{\text{atributos de } S_i\} \cup \{k, a_1, \dots, a_n\}$  y  $CP(L_i) = k$
- **Se crea una relación para cada una de las subclases. Las relaciones tendrán los atributos de la superclase más los atributos de cada una de las subclases. La llave corresponde a la superclase.**

### Opción 8B:



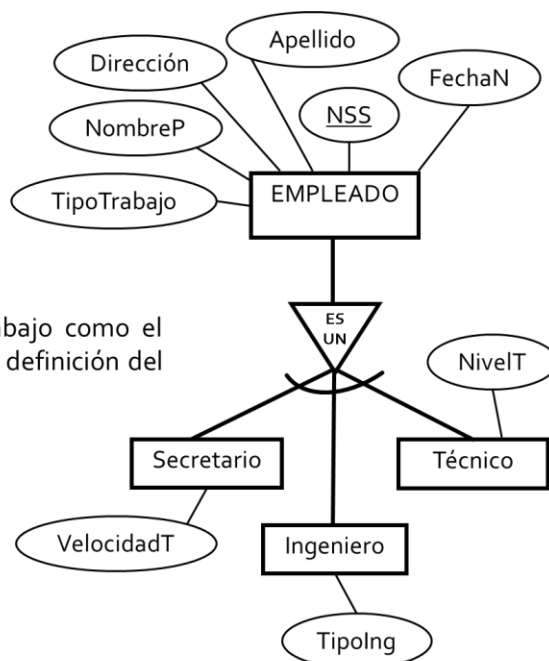
### Opción 8C:

- Crear una sola relación  $L$  con atributos  $Atrs(L) = \{k, a_1, \dots, a_n\} \cup \{\text{atributos de } S_i\} \cup \dots \cup \{\text{atributos de } S_m\} \cup \{t\}$  y  $CP(L) = k$ . Esta opción es para una especialización cuyas subclases son disjuntas, y  $t$  es un atributo discriminador que indica la subclase a la que pertenece cada una de las tuplas.



## Opción 8C:

Para esta opción, se ha usado el atributo TipoTrabajo como el discriminador o atributo tipo que se especifica en la definición del paso 8C. Note la relación Empleado resultante.



EMPLEADO

<u>NSS</u>	NombreP	Apellido	Dirección	FechaN	TipoTrabajo	VelocidadT	NivelT	TipoIng
------------	---------	----------	-----------	--------	-------------	------------	--------	---------

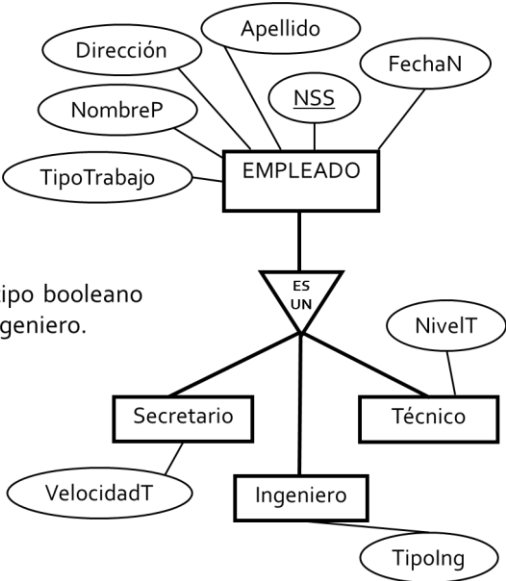
## Opción 8D:

- Crear una sola relación L con atributos  $Atrs(L) = \{k, a_1, \dots, a_n\} \cup \{\text{atributos de } S_i\} \cup \dots \cup \{\text{atributos de } S_m\} \cup \{t_1, t_2, \dots, t_m\}$  y  $CP(L) = k$ . Esta opción es para una especialización cuyas subclases son solapan, y cada t es un atributo booleano que indica si una tupla pertenece o no a una subclase  $S_i$ .



Opción 8D:

Para esta opción, se ha agregado una columna de tipo booleano para cada una de las subclases Secretario, Técnico e Ingeniero.



EMPLEADO											
<u>NSS</u>	NompreP	Apellido	Dirección	FechaN	TipoTrabajo	Val_S	VelocidadT	Val_T	NivelT	Val_I	Tipolng