



BASE DE DATOS

Modelo conceptual a Modelo Lógico



Subsecretaría de
Empleo
Chaco Gobierno de todos



Ministerio de
Producción, Industria y Empleo
Chaco Gobierno de todos



CHACO
Gobierno de todos



Del Modelo Conceptual al Modelo Lógico

Luego de haber realizado el análisis de la situación real, modelando la misma mediante un Esquema Conceptual, a través del MER, nos abocaremos a la obtención del modelo lógico de nuestra Base de Datos.



En este apunte te contamos los pasos a seguir detalladamente a partir de diversos ejemplos.

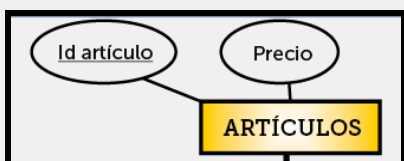


Entidades

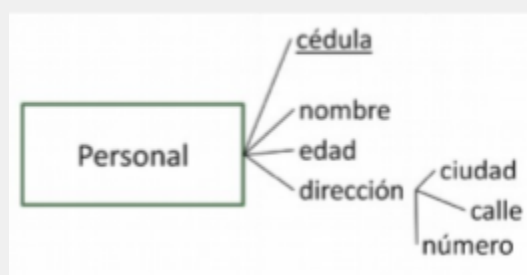
Todas las **entidades regulares** presentes en el MER se transforman en **tablas** en el modelo lógico, manteniendo el número y tipo de los atributos, así como las claves primarias.

- El **nombre** de la tabla será el nombre de la entidad.
- Cada **atributo** de la entidad será un **campo** de la tabla.
- Los atributos que definen la **clave primaria** serán, también, la **clave primaria** de la tabla.
- En el caso que hayan atributos compuestos se descompone el atributo en campos.

Ver ejemplo de Personal



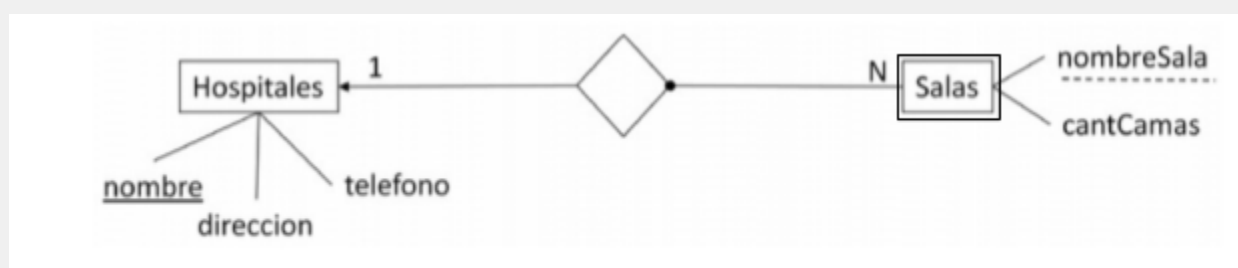
ARTÍCULOS
* ID_artículo Precio



PERSONAL
*Cédula Nombre Edad Ciudad Calle Número



Las **entidades débiles** también se convierten en tablas en el modelo relacional, manteniendo el **número y tipo de los atributos**, pero su clave primaria se forma por la composición de su clave primaria con la clave primaria de la entidad regular de la cual depende.



HOSPITAL
*NombreHospital
Dirección
Telefono

SALAS
*NombreHospital
*NombreSala
Dirección
Telefono



Relaciones

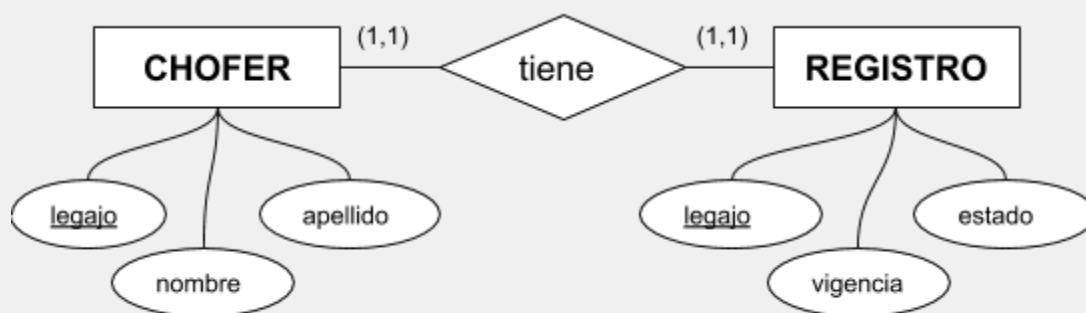
Para convertir las relaciones a tablas, hay que tener presente su cardinalidad: 1:1, 1-N, N:N

Dependiendo de cómo se definan las cardinalidades de las relaciones, éstas pueden generar una tabla o por el contrario traducirse en campos dentro de las tablas asociadas a las entidades que participan en dicha relación.

Relaciones 1:1

En este tipo de relaciones las entidades podrían utilizar la misma clave o cada cual la suya.

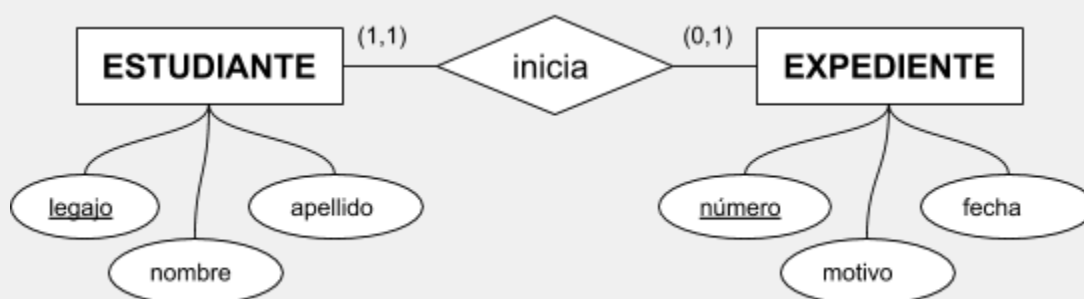
- Si las dos entidades tienen el mismo identificador, entonces se transforman en una única tabla por la agregación de los atributos de las dos entidades y la clave es la clave de las entidades (es la misma en ambas).



CHOFER
*legajo nombre apellido vigencia estado



- Si las dos entidades tienen **distinto identificador**, entonces cada entidad se transforma en una tabla con clave principal y cada tabla tendrá como **clave foránea** el identificador de la otra tabla con la cual está relacionada.



ESTUDIANTE
PK legajo
nombre
apellido

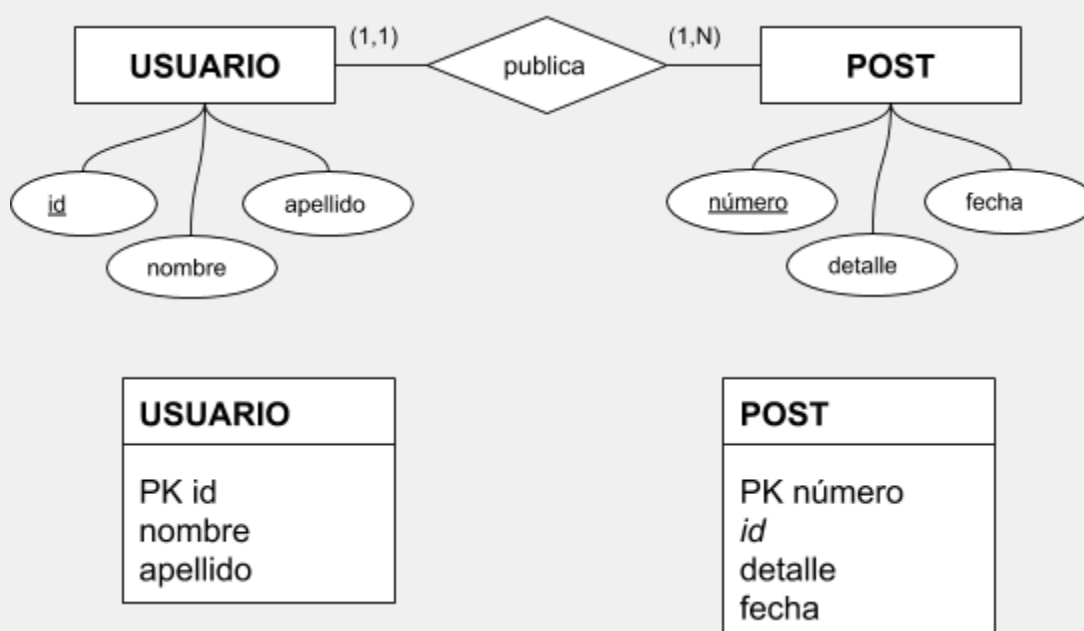
PK Primary key
FK Foreign key

EXPEDIENTE
PK número
FK legajo
motivo
fecha



Relaciones 1:N

En el tipo de relación binaria 1:N, **no es necesario generar otra tabla**, lo que se hace es incluir en la entidad del lado de N el identificador de la otra tabla y los posibles campos de la relación si los hubiere. Veamos cómo funciona:

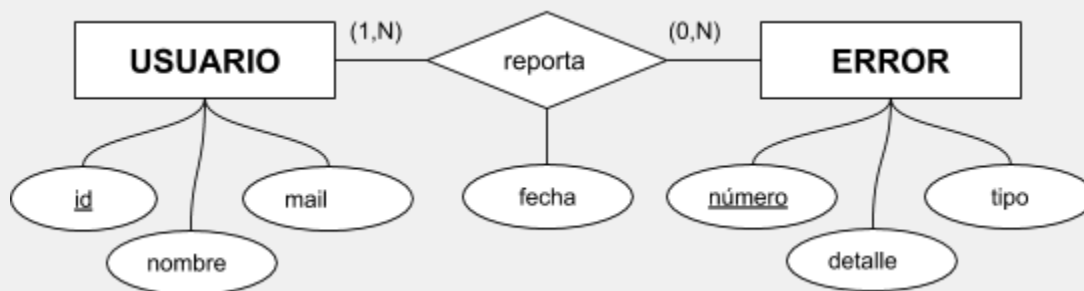




Relaciones N:N

En este caso crearemos una **nueva tabla** para representar la **relación**, agregando la clave primaria de cada una de las tablas, además de los campos que correspondan a atributos de la relación.

La **clave primaria** de la nueva tabla será la **combinación de las claves primarias** de las tablas que participan en la relación.

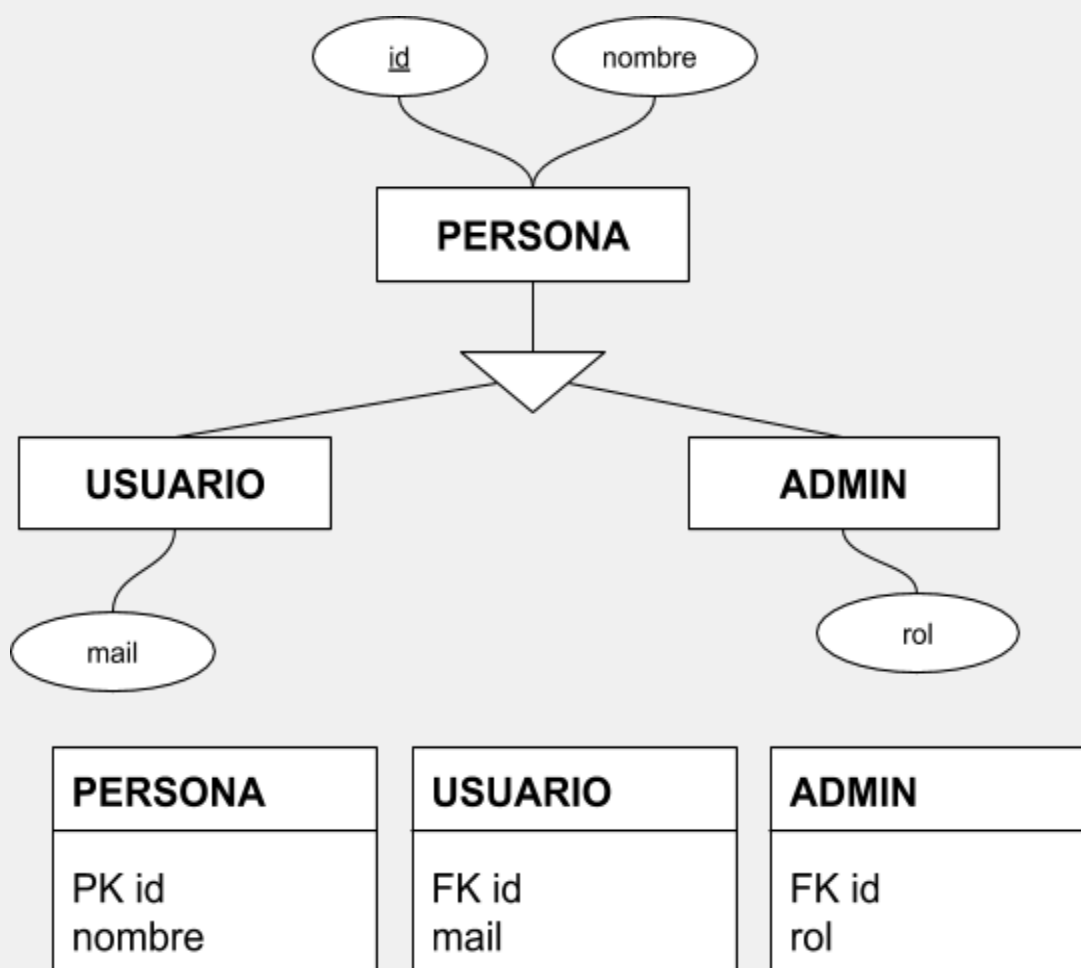


USUARIO	REPORTE	ERROR
PK id nombre mail	PK id PK número fecha	PK número detalle tipo



Relaciones de Herencia

En el caso de representaciones de Herencia normalmente se generan las siguientes tablas: una para la super entidad (nivel superior), con los atributos comunes, y otra tabla para cada una de las entidades subordinadas, con la misma clave que la entidad superior y los atributos propios de esa sub-entidad. Por ejemplo



Con esta solución un Usuario nunca puede ser Admin, ni viceversa....
qué opción hay para poder representar que es posible que un Usuario sea Admin?



Para solucionarlo, tenemos la opción de crear una tabla con la super entidad, sus atributos y claves e incorporar los atributos de las entidades subordinadas. Agregando además el campo *tipo* que permitirá distinguir si la persona es USUARIO o ADMIN.

