

Base de Datos I

Introducción

Conceptos

- Dato
- Información

Archivos

- Tipos
 - Texto
 - Binario
- Problemas en el uso de archivos
 - Redundancia
 - Falta de integridad
 - Dependencia de las aplicaciones que los utilizan

Medios de almacenamiento

- Físico

- Magnético
 - Disco Rígido
 - Cinta (Backups)
 - Otros: Diskette (Floppy Disk), ZIP Drive, Casette, etc.
- Óptico
 - CD, DVD, BR
- Memoria (No volátil)
 - Flash: Pen Drive, SD, SSD

- Virtual

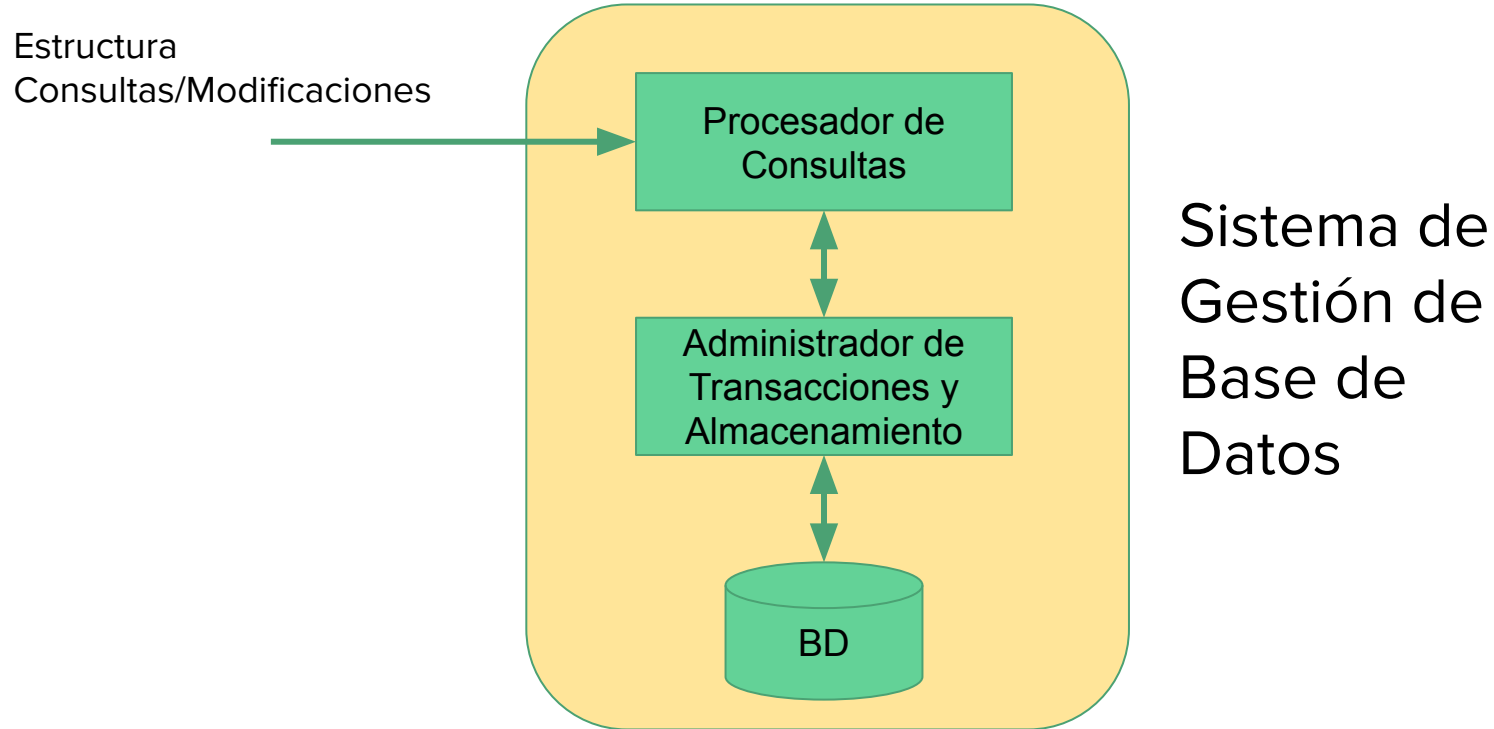
- Servidor de archivos: Samba, FTP, etc.
- Nube (Cloud): Drive, OneDrive, Dropbox, etc.

Bases de datos

“Conjunto de datos almacenados en forma independiente de las aplicaciones que las utilizan. Poseen redundancia mínima y permiten el control de integridad”

- Nacen en los años 60
- Son administradas por un Sistema de Gestión de Base de Datos (SGBD, o en inglés, DBMS)
- Otras características
 - Seguridad
 - Backups

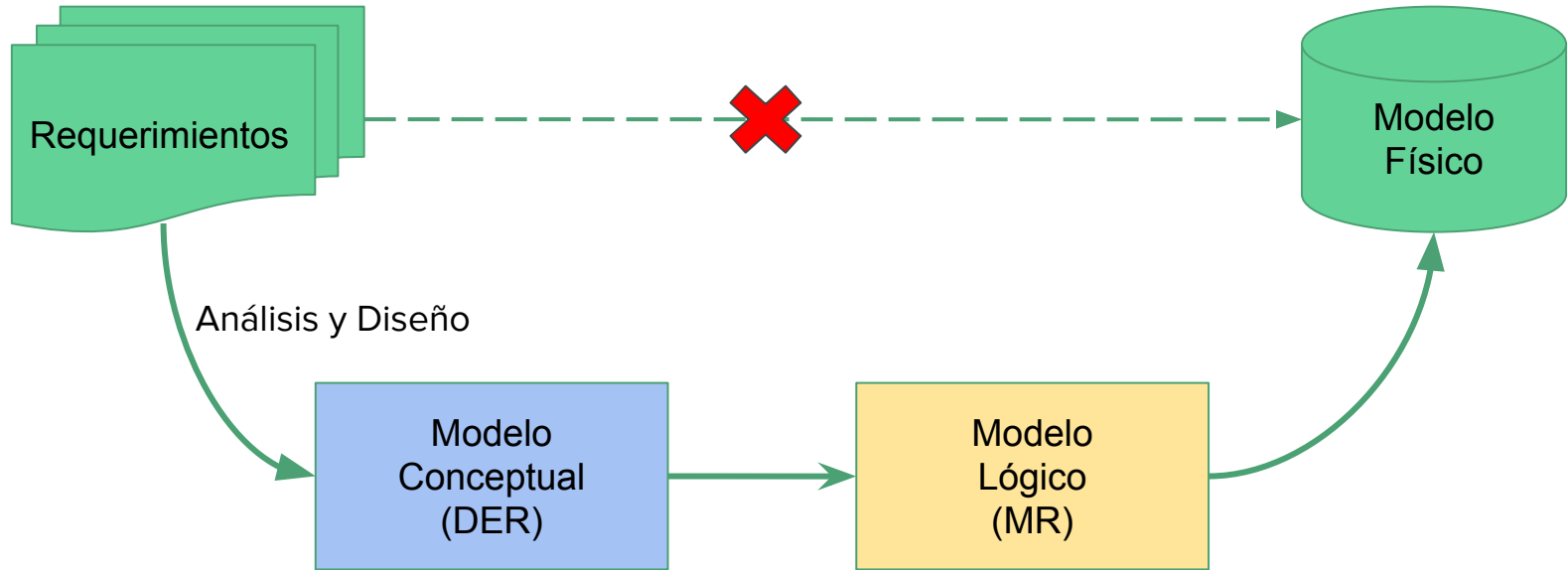
Diagrama de estructura



Tipos de bases de datos

- Variabilidad de los datos
 - Dinámica
 - Estática
- Estructura de almacenamiento
 - Relacional
 - Objetos
 - Documentos (No-SQL)
- Otras
 - Geográficas

Proceso de Diseño y Modelado



Base de Datos I

Diagrama Entidad-Relación (Básico)

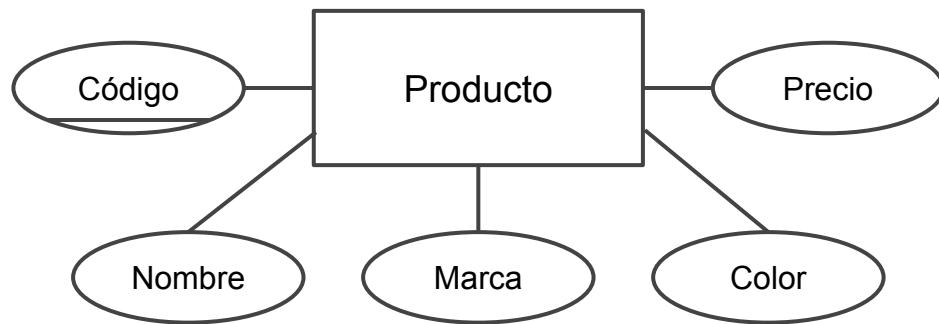
Diagrama Entidad-Relación (DER)

- Modelo gráfico conceptual
- Resultado de análisis de requerimientos
- Elementos
 - Entidad
 - Relación

Entidad

“Abstracción del dominio (cosa, objeto o concepto) que se distingue del resto”

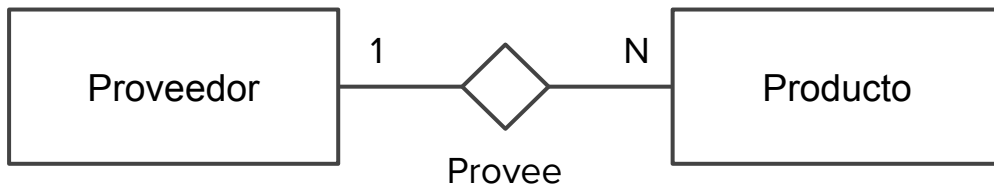
- Conjunto de entidades
- Sustantivo/Singular
- Atributos
 - Simple
 - Identificador



Relación

“Vínculo entre instancias de uno o más conjuntos de entidades”

- Verbo/Frase verbal
- Grado
 - Unaria (1)
 - Binaria (2)
 - Ternaria (3)
- Cardinalidad
 - Uno (1)
 - Muchos (N)
- Atributos de relación



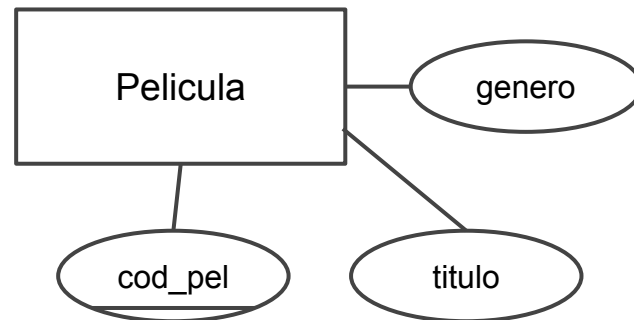
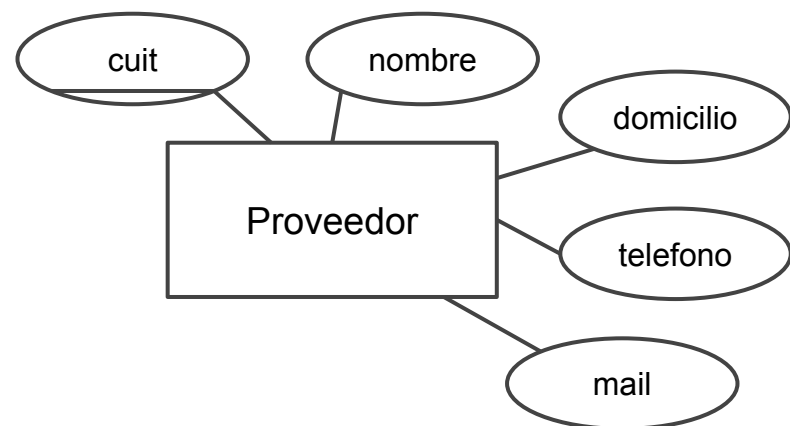
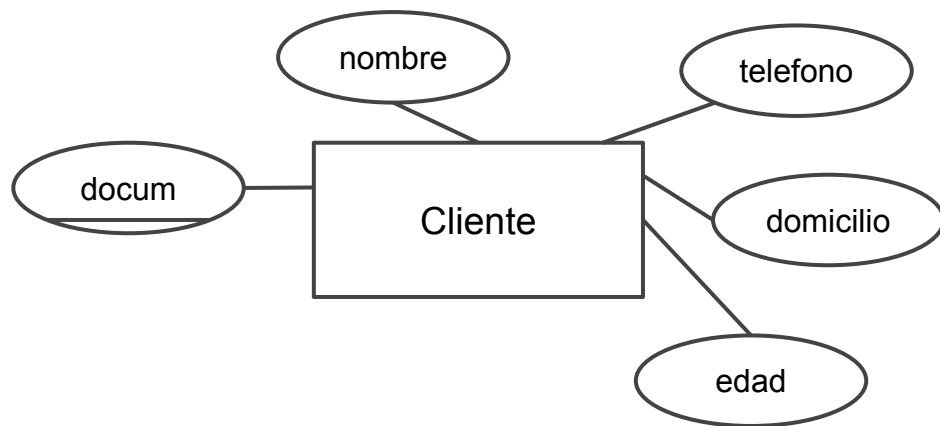
Ejercicio Práctico - Videoclub

“Un videoclub requiere informatizar el registro de sus clientes y alquileres de sus películas. Por cada cliente se conoce su documento, nombre, domicilio, edad y teléfono. Se necesitan conocer las películas que estos alquilan, de las cuales se registra su título y género. También se mantiene registro de los proveedores de películas, de quienes se conoce su CUIT, nombre, domicilio y teléfono. Una película es provista por un único proveedor. Es requerido también almacenar su e-mail para poder realizar pedidos en forma automática. Por otra parte, los clientes pueden tener a otros clientes como extensiones de su cuenta, pero un cliente solo puede ser extensión de un único titular. Se requiere también saber la fecha en que se realizó el alquiler de cada película”

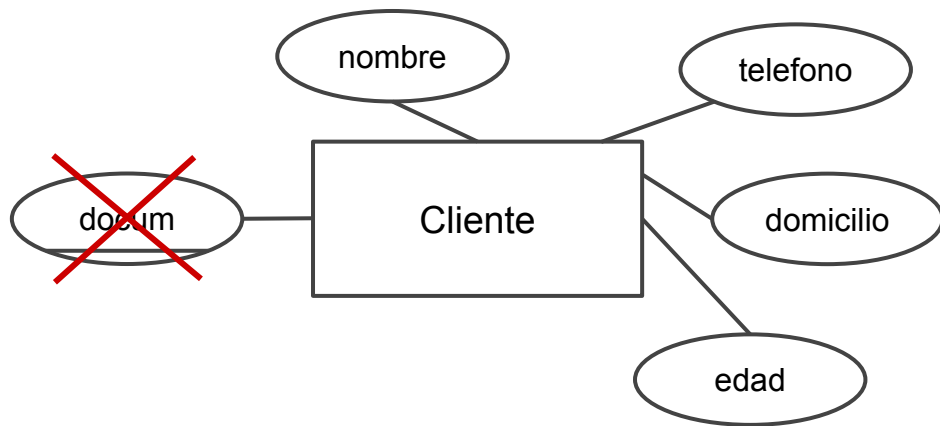
Videoclub - Entidades y Atributos

“Un videoclub requiere informatizar el registro de sus clientes y alquileres de sus películas. Por cada **cliente** se conoce su **documento, nombre, domicilio, edad y teléfono**. Se necesitan conocer las **películas** que estos alquilan, de las cuales se registra su **título y género**. También se mantiene registro de los **proveedores** de películas, de quienes se conoce su **CUIT, nombre, domicilio y teléfono**. Una película es provista por un único proveedor. Es requerido también almacenar su **e-mail** para poder realizar pedidos en forma automática. Por otra parte, los clientes pueden tener a otros clientes como extensiones de su cuenta, pero un cliente solo puede ser extensión de un único titular. Se requiere también saber la fecha en que se realizó el alquiler de cada película”

Videoclub - Entidades y Atributos

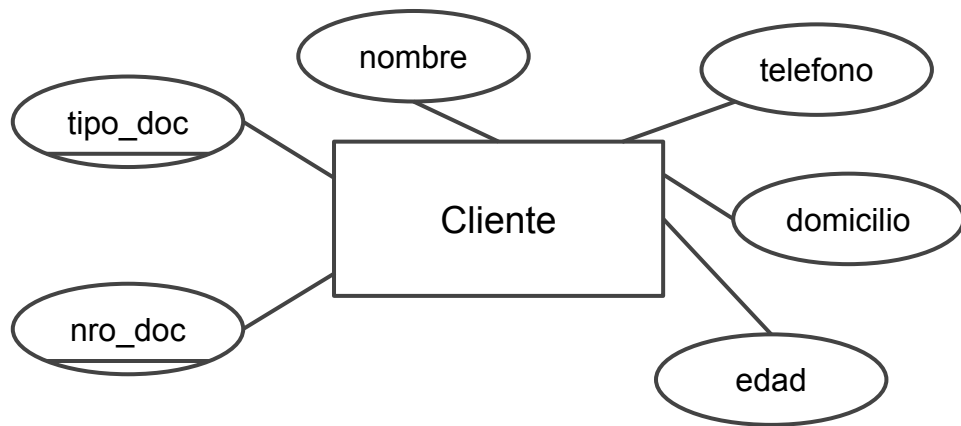


Videoclub - Documento duplicado



documento
123456
567890
123456

Videoclub - Identificador Compuesto

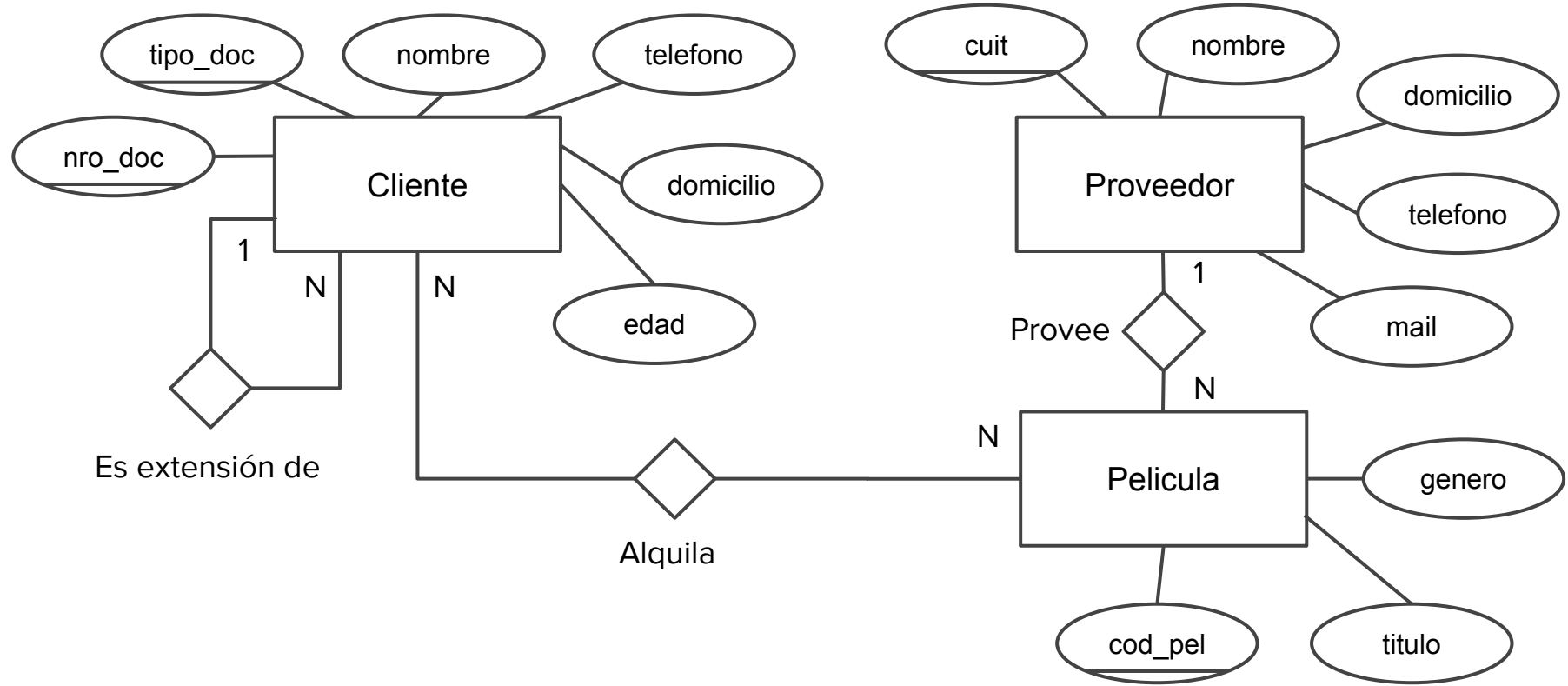


tipo_doc	nro_doc
DNI	123456
LC	567890
LC	123456

Videoclub - Relaciones

“Un videoclub requiere informatizar el registro de sus clientes y alquileres de sus películas. Por cada cliente se conoce su documento, nombre, domicilio, edad y teléfono. Se necesitan conocer las películas que estos **alquilan**, de las cuales se registra su título y género. También se mantiene registro de los proveedores de películas, de quienes se conoce su CUIT, nombre, domicilio y teléfono. **Una película es provista por un único proveedor**. Es requerido también almacenar su e-mail para poder realizar pedidos en forma automática. Por otra parte, los clientes pueden tener a **otros clientes como extensiones** de su cuenta, pero **un cliente solo puede ser extensión de un único titular**. Se requiere también saber la fecha en que se realizó el alquiler de cada película”

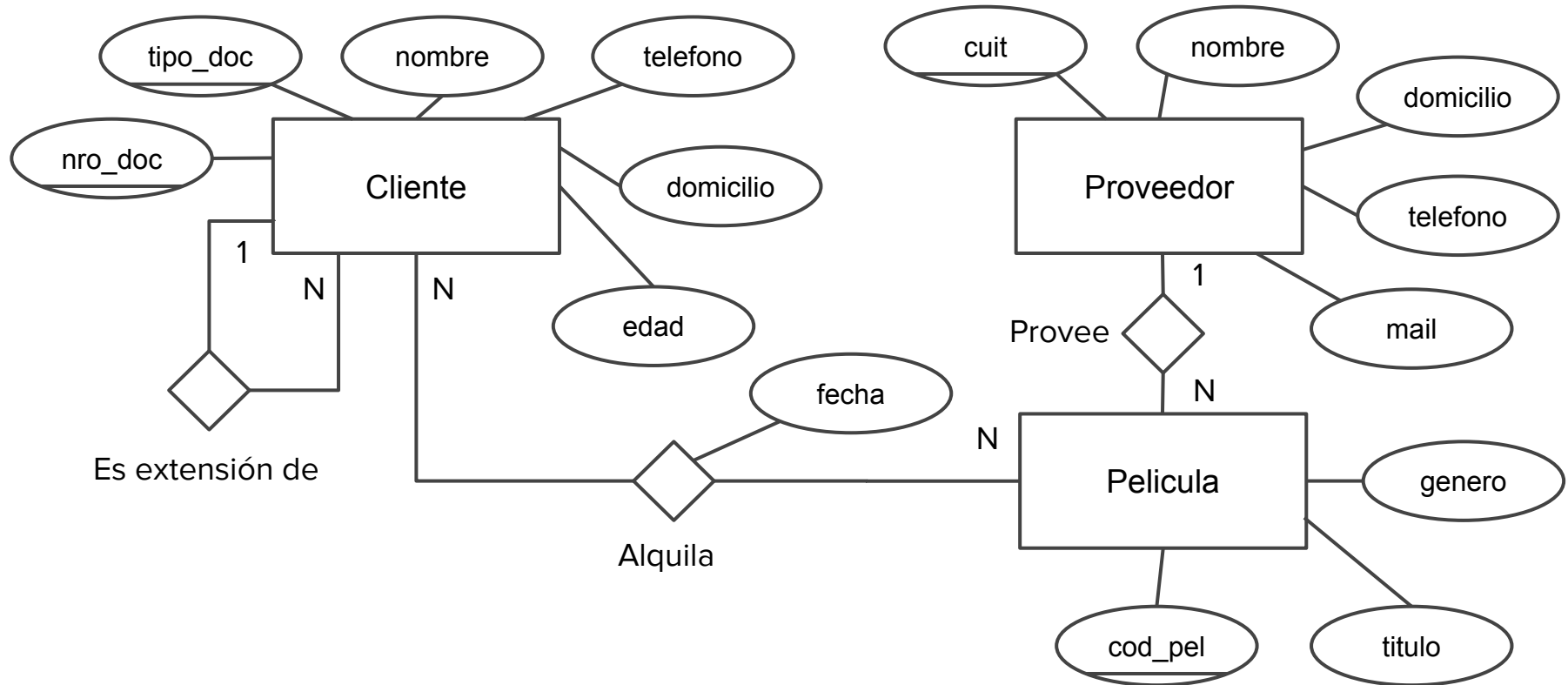
Videoclub - Relaciones



Videoclub - Atributos de Relación

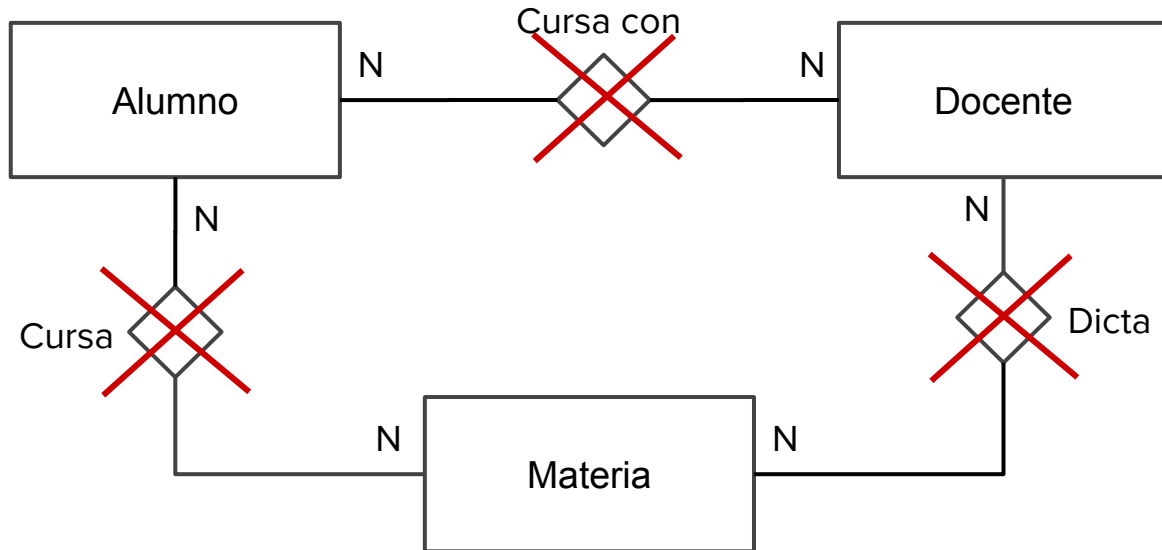
“Un videoclub requiere informatizar el registro de sus clientes y alquileres de sus películas. Por cada cliente se conoce su documento, nombre, domicilio, edad y teléfono. Se necesitan conocer las películas que estos alquilan, de las cuales se registra su título y género. También se mantiene registro de los proveedores de películas, de quienes se conoce su CUIT, nombre, domicilio y teléfono. Una película es provista por un único proveedor. Es requerido también almacenar su e-mail para poder realizar pedidos en forma automática. Por otra parte, los clientes pueden tener a otros clientes como extensiones de su cuenta, pero un cliente solo puede ser extensión de un único titular. Se requiere también saber la fecha en que se realizó el **alquiler** de cada película”

Videoclub - Atributos de Relación



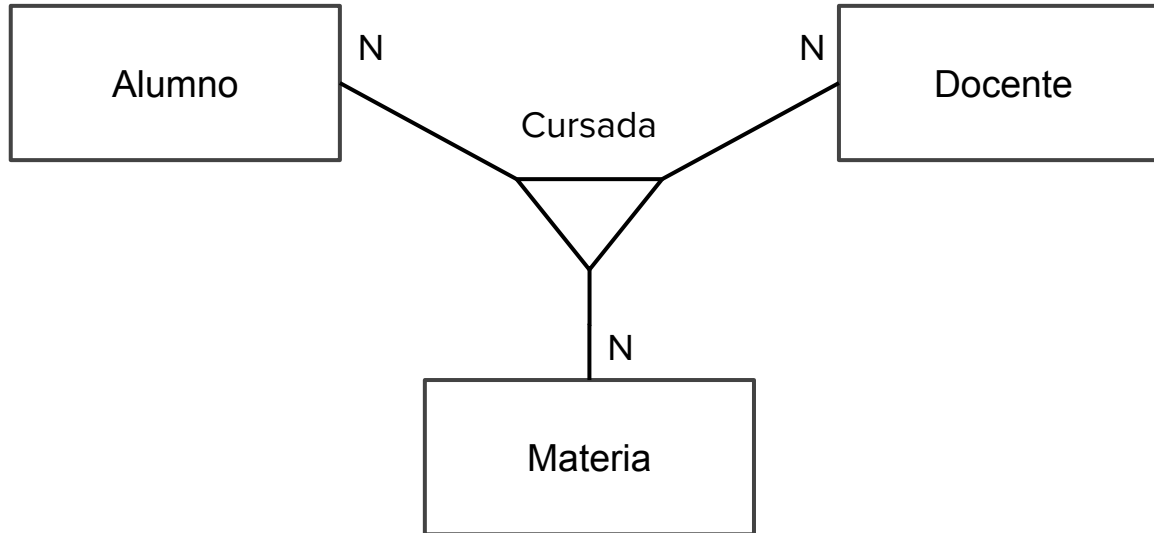
Ejercicio Práctico - Universidad

“Una universidad desea poder registrar y consultar, por cada alumno, qué materias cursa y con qué docentes en cada una de ellas”



Ejercicio Práctico - Universidad

“Una universidad desea poder registrar y consultar, por cada alumno, qué materias cursa y con qué docentes en cada una de ellas”



Ejercicio a Resolver

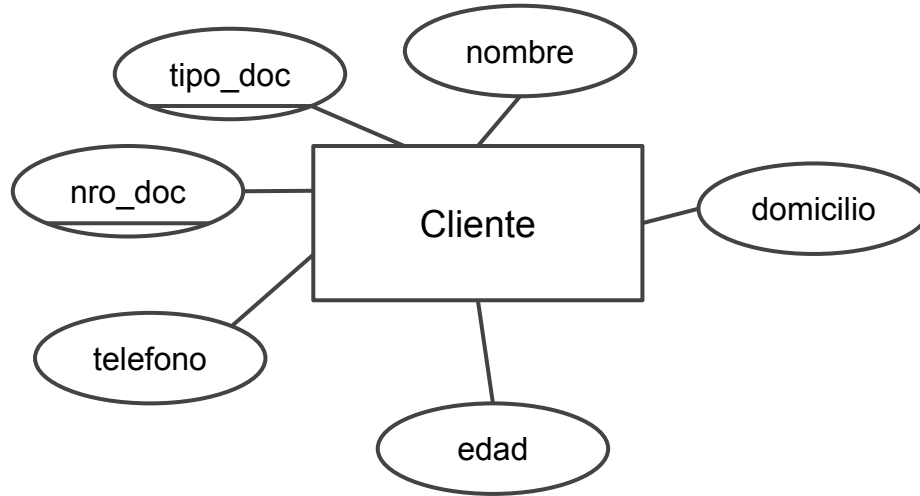
“Se necesita una base de datos de los diferentes modelos de autos del mercado. Por cada modelo se desea saber la marca, nombre y precio actual. También se requiere conocer el motor que posee, el cual tiene su nombre propio, cilindrada, potencia y marca (que no tiene por que ser igual a la del auto). Las marcas se encuentran clasificadas según su nacionalidad. Cada modelo se compone además de diferentes accesorios y, de acuerdo al modelo, el accesorio puede estar incluido o ser opcional en la compra (ej: alarma). Cada accesorio tiene su propio código”

Base de Datos I

Diagrama Entidad-Relación (Extendido)

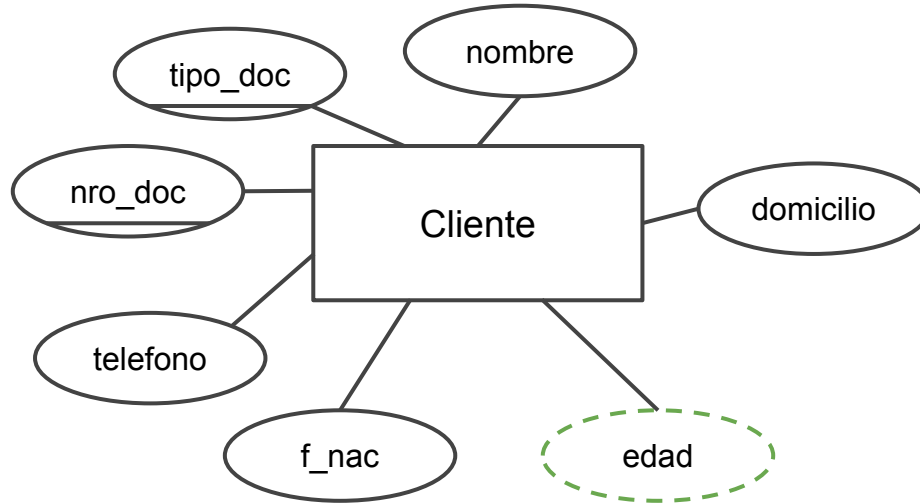
Atributo Calculado

- Problema en el uso de edad como atributo normal



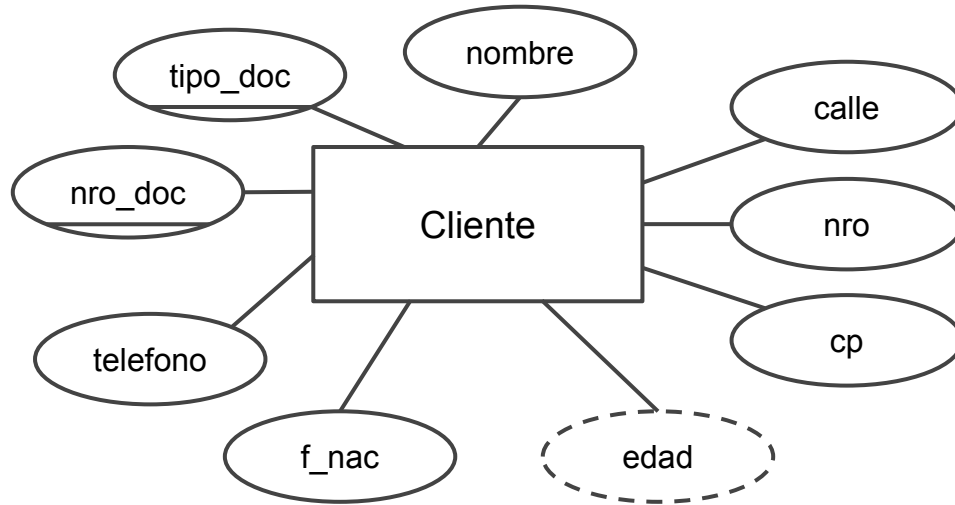
Atributo Calculado

- La edad se calcula en base a la fecha de nacimiento



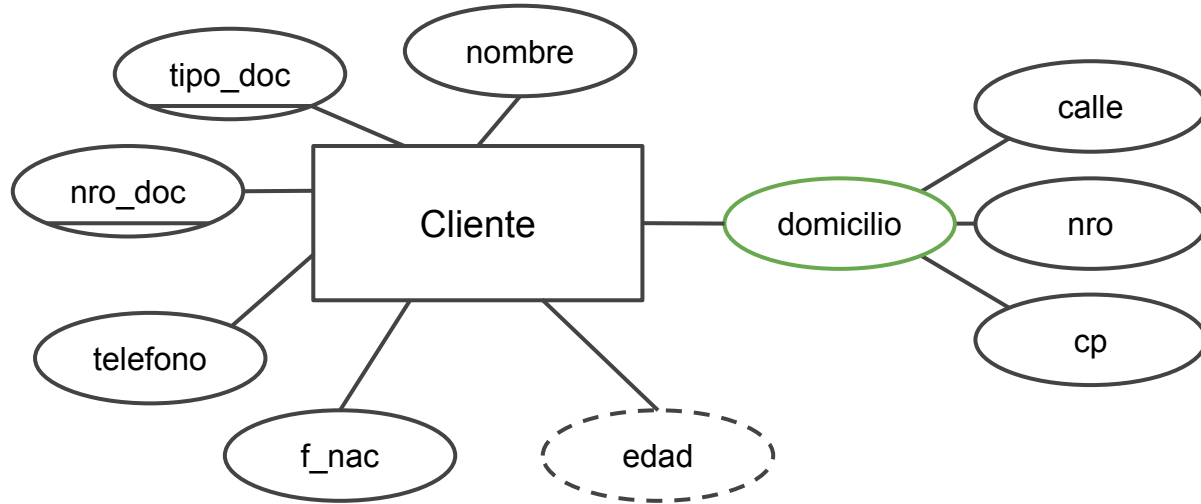
Atributo Agrupador

- Se desea desagregar la descripción del domicilio



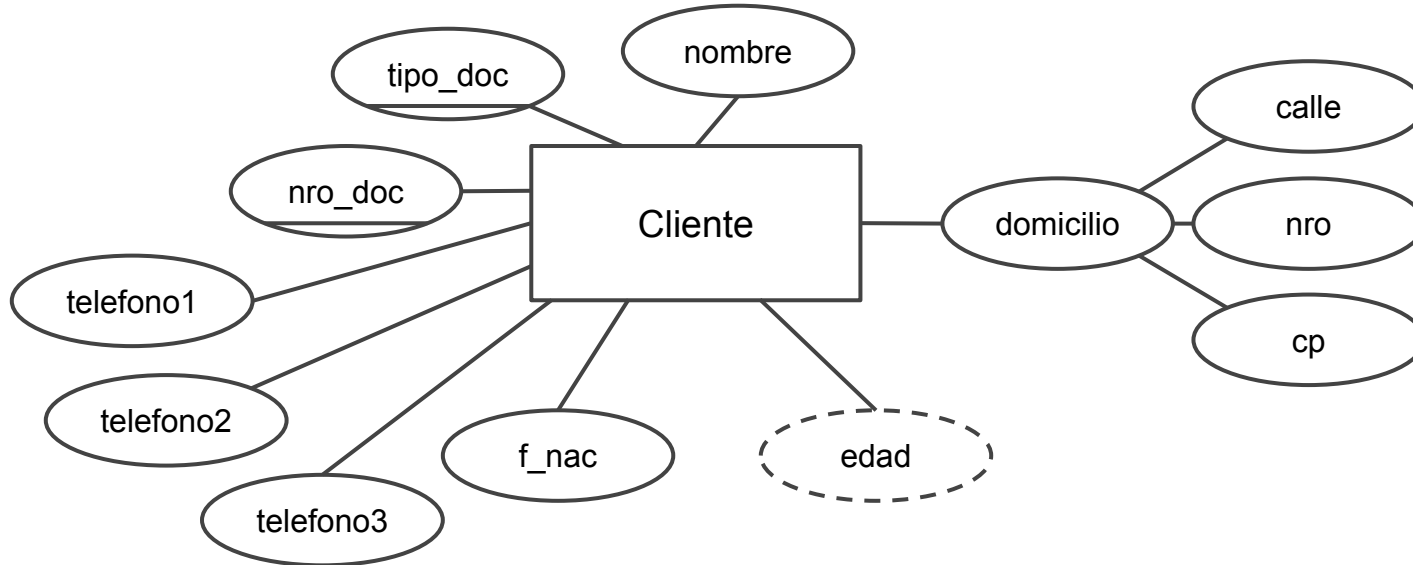
Atributo Agrupador

- El domicilio agrupa a sus atributos correspondientes



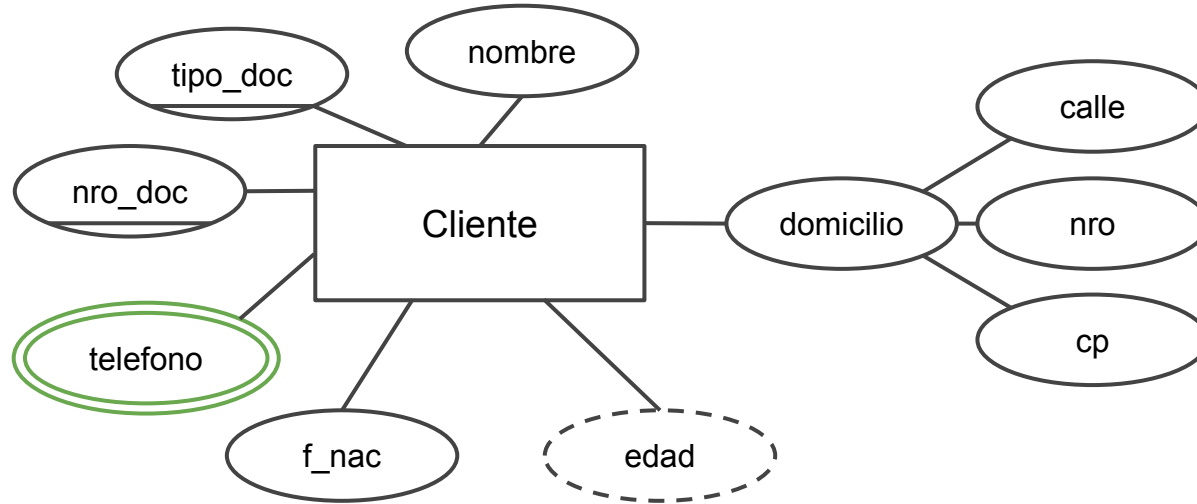
Atributo Multivaluado

- Se requiere almacenar más de un teléfono del cliente



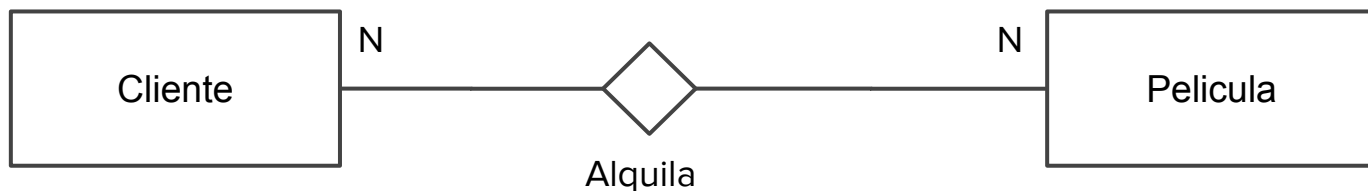
Atributo Multivaluado

- Ahora el atributo permite el almacenamiento de múltiples valores



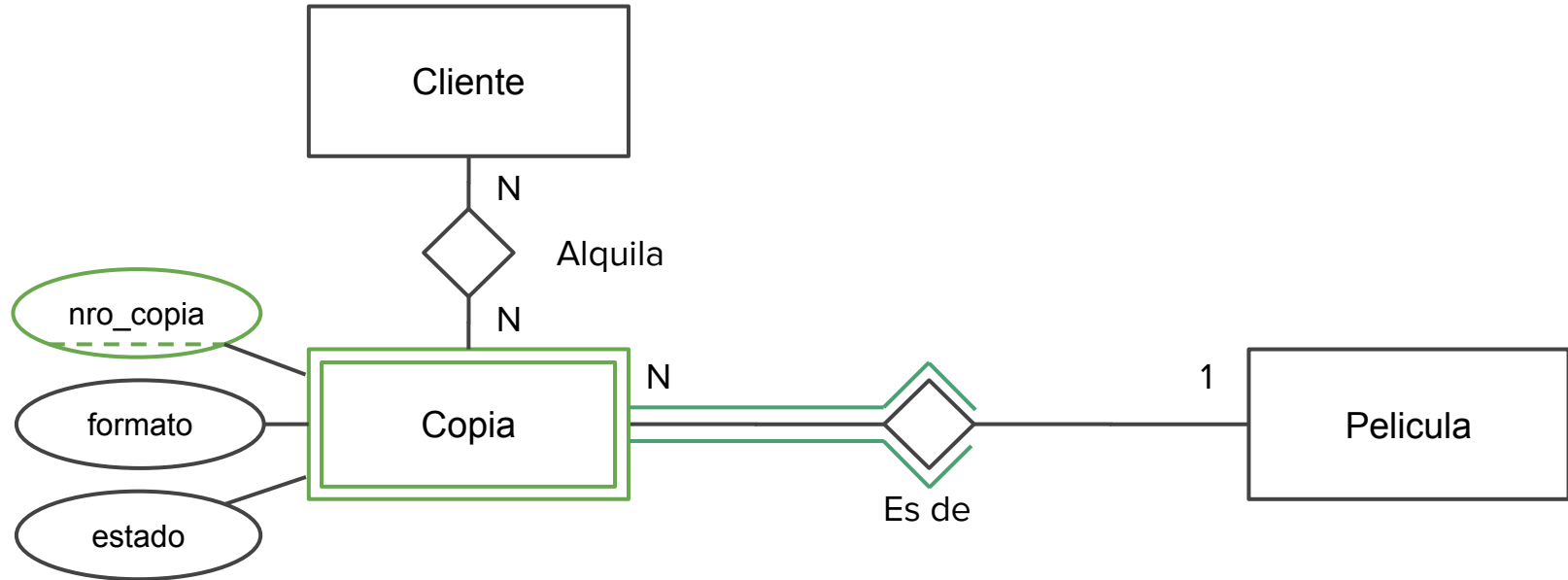
Entidad Débil

- Se necesita poder saber qué copia de nuestras películas se llevó cada cliente. Las mismas se identifican con el código de película y un número secuencial (ej: si la película tiene código 5555, las copias serían 5555-1, 5555-2, 5555-3, etc.)



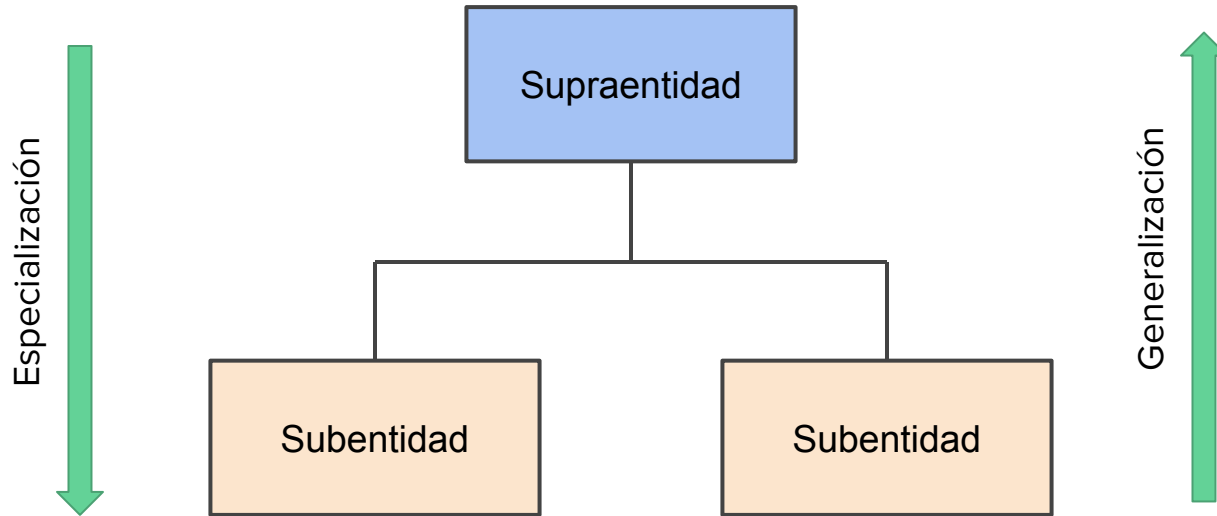
Entidad Débil

- Una entidad débil requiere de **otra entidad** para poder **identificarse**
- Posee además un **discriminante** y puede tener otros atributos



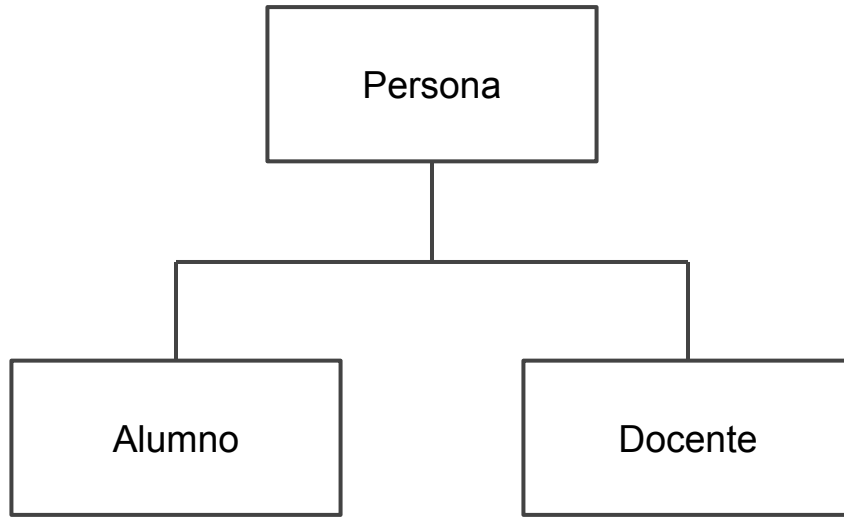
Jerarquía

- Supraentidad: posee **identificador** y atributos **comunes**
- Subentidad: tiene atributos **específicos** y hereda los de la supraentidad

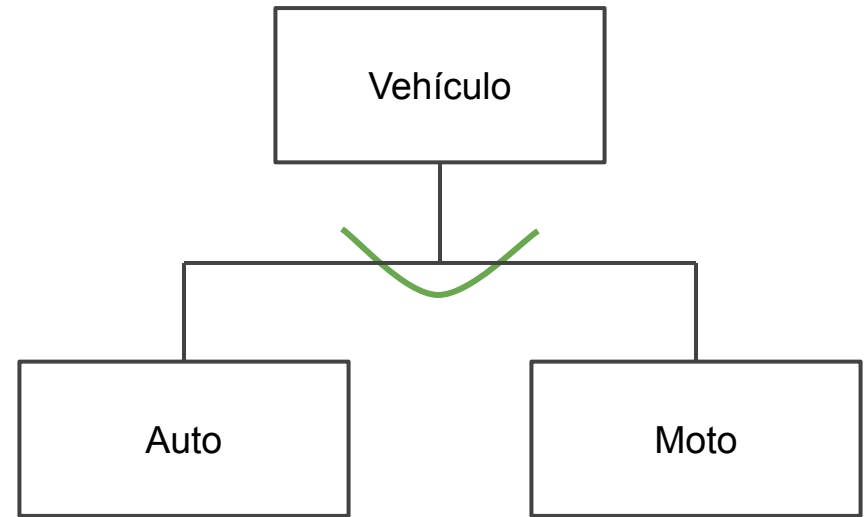


Jerarquía - Solapamiento

- Existe solapamiento cuando una instancia de una supraentidad puede ser a su vez una instancia de dos o más subentidades



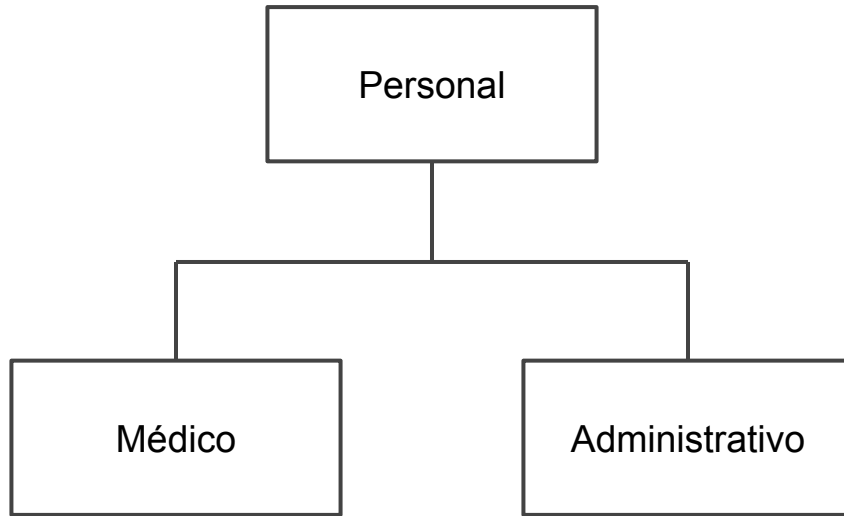
Con Solapamiento



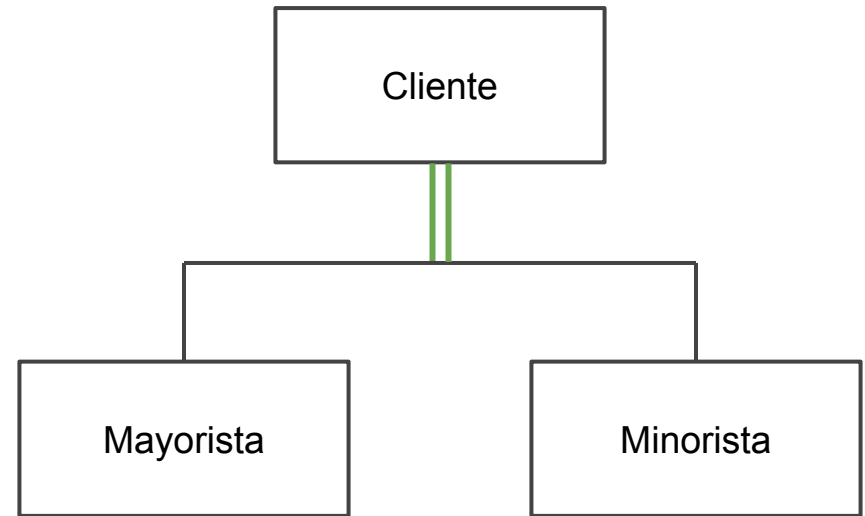
Sin Solapamiento

Jerarquía - Partición

- Parcial: una instancia de una supraentidad puede no ser una subentidad
- Total: toda instancia de una supraentidad es al menos una subentidad



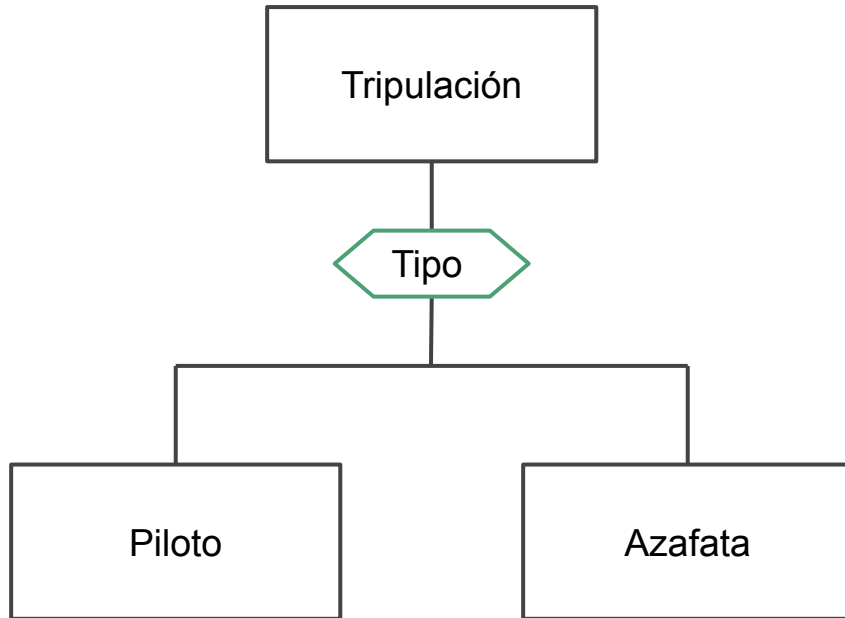
Partición Parcial



Partición Total

Jerarquía - Tipo

- Sólo existe en el caso más restrictivo: partición total, sin solapamiento



Ejercicio a Resolver

Modificar el ejercicio del videoclub, intentando aplicar el concepto de jerarquía.

Pregunta: Qué proceso de jerarquización estás aplicando?

Base de Datos I

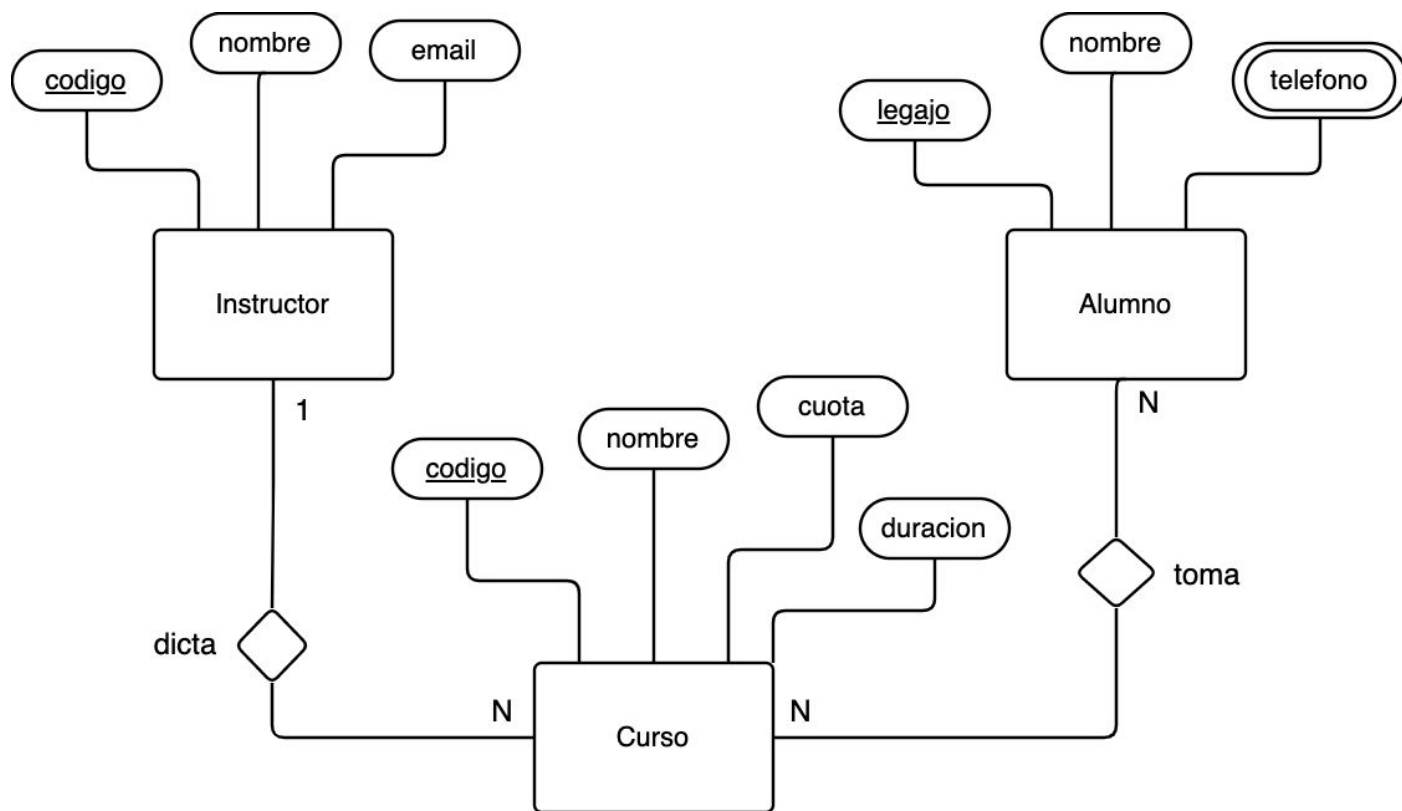
Práctica DER

Ejercicio 1

“Soy el gerente de una empresa de capacitación que brinda cursos relacionados con la informática. Dictamos varios cursos, cada uno de los cuales tiene un código, un nombre y valor de cuota. Introducción a UNIX y Programación C son dos de nuestros cursos más populares. También necesitaríamos conocer la duración de los cursos, lo cual varía de uno a cuatro días. Un instructor puede dictar varios cursos. Pablo Basile y Daniel Abadi son dos de nuestros mejores profesores. Registramos el nombre y dirección de correo de cada instructor. Cada curso es dictado por un único instructor. Nuestros instructores se encuentran capacitados para dictar diferentes cursos.

Los alumnos pueden tomar varios cursos al mismo tiempo, y muchos de ellos lo hacen. ¡Mariana Carpovich de AT&T se anota en cada curso que ofrecemos!. También registramos el nombre y números telefónicos de cada uno de nuestros alumnos (pueden poseer más de un teléfono)”

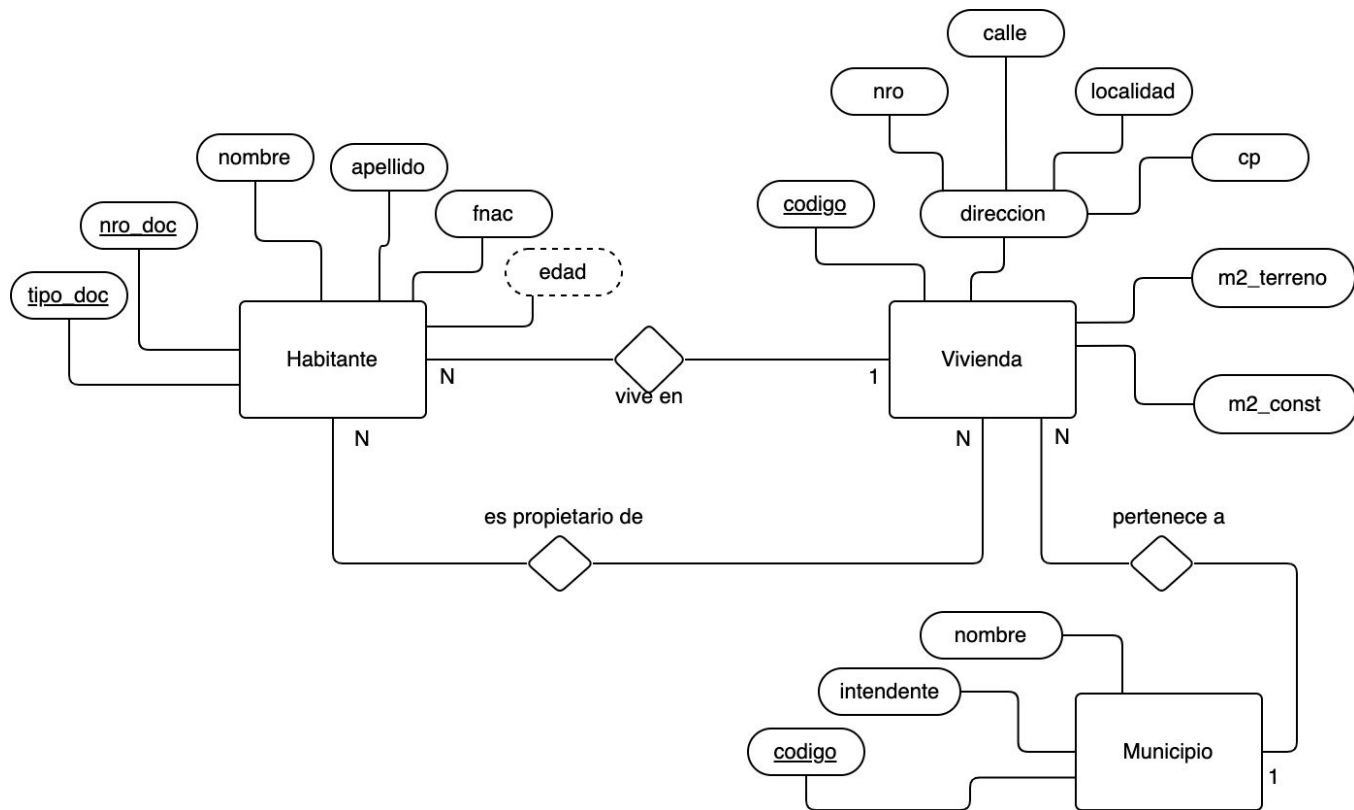
Ejercicio 1



Ejercicio 2

“La Secretaría de Vivienda de la Provincia de Córdoba desea almacenar información relacionada a los habitantes y las viviendas donde residen los mismos. De los habitantes, nos interesa almacenar su tipo y número de documento, su nombre, apellido, fecha de nacimiento y edad. Sabemos que cada persona debe vivir en una única vivienda, la cual no necesariamente es de su propiedad y además, una persona puede ser propietaria de varias viviendas. Cabe destacar que existen propiedades que tienen múltiples propietarios. De las viviendas nos interesa almacenar la dirección (calle, número, localidad y código postal), la cantidad de metros cuadrados del terreno y la cantidad de metros cuadrados que se encuentran edificados, además del municipio al que pertenece. Respecto a los municipios, deseamos conocer su nombre y el intendente actual”

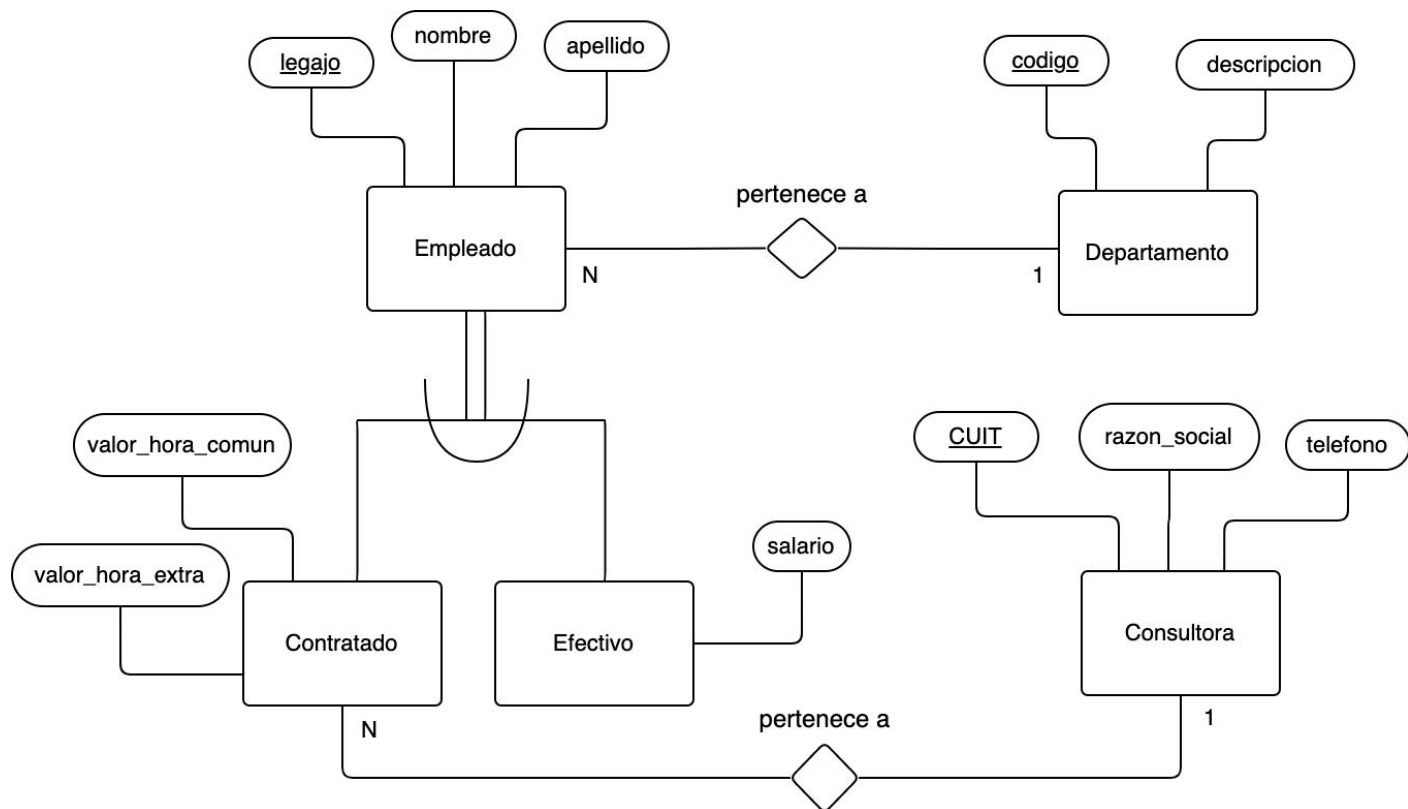
Ejercicio 2



Ejercicio 3

“Una empresa posee dos tipos de empleados: contratados y efectivos. Para todos ellos, se desea almacenar su nro. de legajo, nombre, apellido y departamento en el que trabaja. Cada departamento tiene un código y una descripción. Para los empleados efectivos, se debe almacenar su salario. Para aquellos empleados contratados, almacenar el precio que cobra por hora común y por hora extra, y la consultora a la que pertenecen. De cada consultora es necesario saber su nro. de CUIT, Razón Social y teléfono”

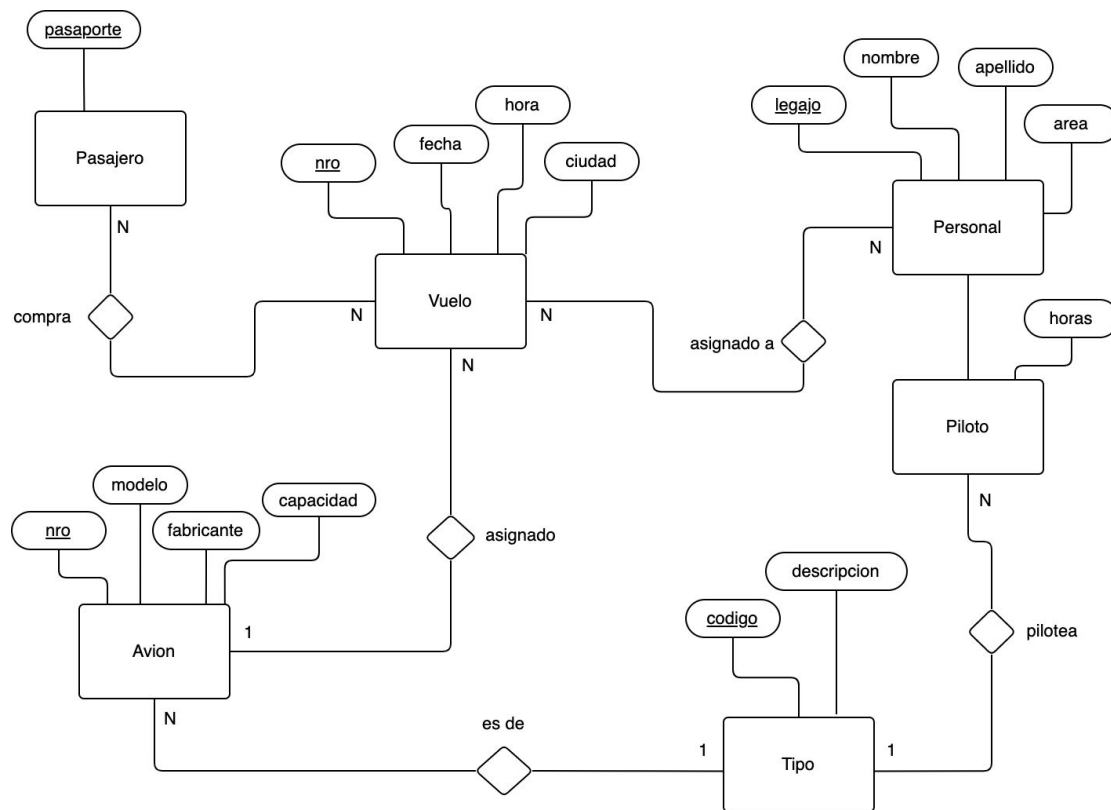
Ejercicio 3



Ejercicio 4

“Una aerolínea maneja información de pasajeros, vuelos y personal. Para los pasajeros se considera de interés el pasaporte y los vuelos comprados. Para los vuelos: el nro. de vuelo, fecha, hora, ciudad donde hace escala, personal asignado y avión. Para los aviones se considera modelo, fabricante, capacidad, hangar y tipo de avión. Por último, para el personal, se tiene en cuenta el nombre y apellido, área asignada, y en particular para los pilotos se conoce la cantidad de horas de vuelo y el tipo de avión que pilotea”

Ejercicio 4



Base de Datos I

Modelo Relacional (MR)

Relación

- Relación \equiv Tabla
- Tupla \equiv Fila \equiv Registro
- Atributo \equiv Columna \equiv Campo

Cliente

Nombre	Domicilio	Telefono
Juan	Av. Villegas 234	1144552445
María	Bolivar 2253	1156467254
Javier	Espora 24	1134227685

Modelo Relacional - Elementos

- Lista de tablas y campos
- Restricciones de integridad
 - Clave primaria (PK)
 - Clave foránea (FK)

Modelo Relacional - Lista de Tablas

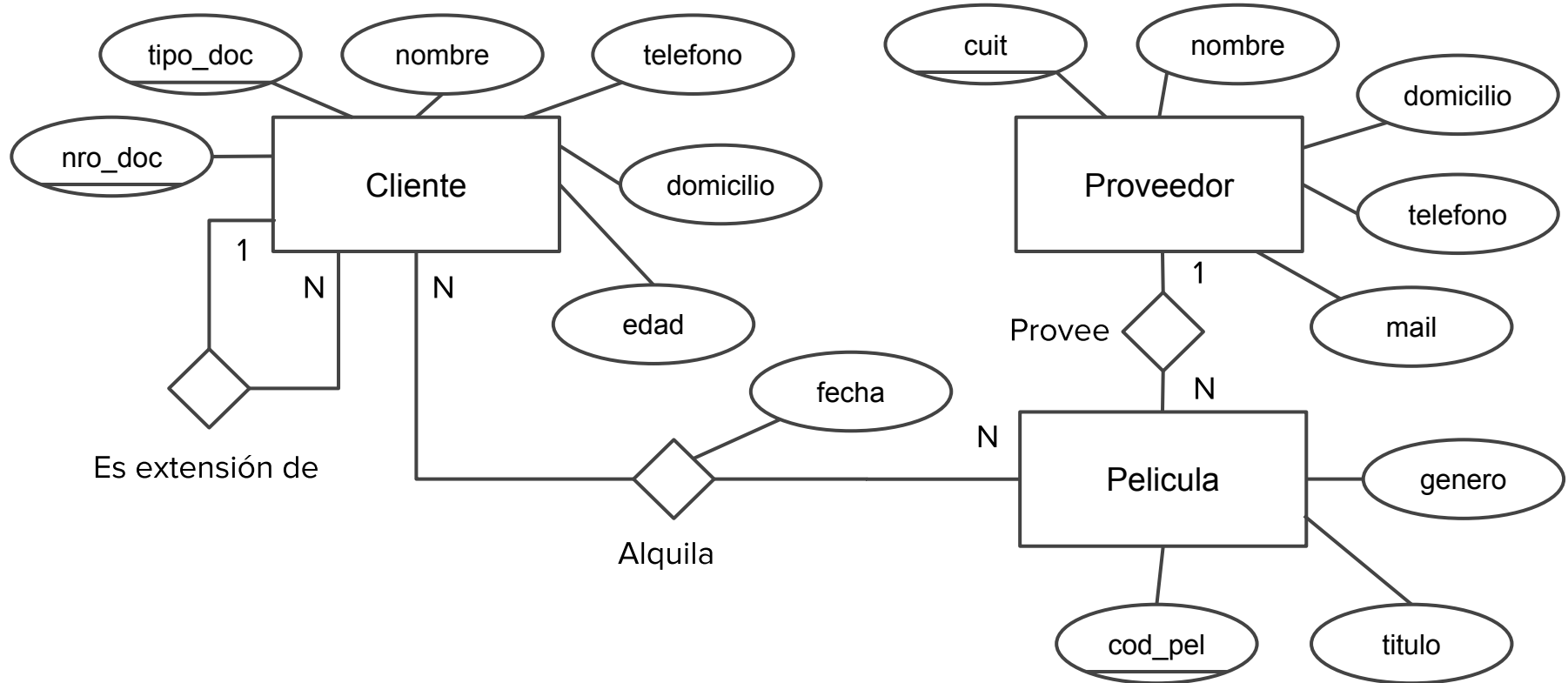
Tabla1 (Campo1, Campo2, ... , CampoN)

Tabla2 (Campo1, Campo2, ... , CampoN)

....

TablaM (Campo1, Campo2, ... , CampoN)

Ejemplo Práctico - Videoclub



Regla 1 - Entidades fuertes

- Toda entidad fuerte pasará a ser una nueva tabla
- Cada atributo se transforma en un nuevo campo
- Los campos correspondientes a los atributos identificadores formarán la **clave primaria (PK)**
 - Se subraya con línea continua a aquellos campos que la forman
 - Puede ser simple (un campo) o compuesta (dos o mas campos)
 - Garantiza que no existirán dos registros en una misma tabla con los mismos valores para todos los campos que la conforman

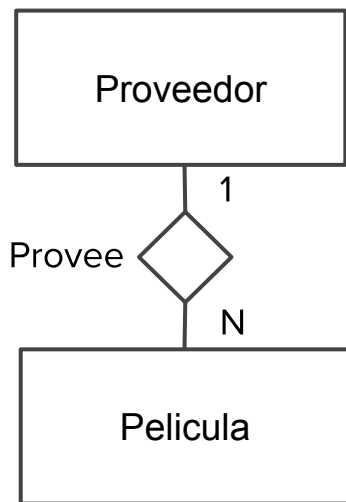
Regla 1 - Entidades fuertes

Cliente (tipo_doc, nro_doc, nombre, telefono, domicilio, edad

Proveedor (CUIT, nombre, domicilio, telefono, mail

Pelicula (cod_pel, titulo, genero

Regla 2 - Relaciones Unarias/Binarias



Proveedor

<u>CUIT</u>	nombre	domicilio	telefono	mail
1234	AVH	Bolivar 2253	1144552445	info@avh.com
5678	Fox	Espora 24	1156467254	info@fox.com

Pelicula

<u>cod_pel</u>	titulo	genero	CUIT_prov
1000	Volver al futuro	C. Ficción	1234
1001	La llamada	Terror	1234
1002	Duro de matar	Acción	5678

Regla 2 - Relaciones Unarias/Binarias

- Por cada relación 1:N se agregará, en la tabla correspondiente al lado de la N, tantos campos como aquellos que forman la clave primaria del lado del 1
- Todos estos campos formarán una **clave foránea (FK)**
 - Se subraya con línea punteada a aquellos campos que la forman
 - Puede ser simple (un campo) o compuesta (dos o mas campos)
 - Se debe definir una **lista de claves foráneas** donde se indica cada clave a que tabla referencia
 - Garantiza que los valores referenciados por la clave existen en la tabla destino

TablaOrigen.Campo → TablaDestino.Campo

Regla 2 - Relaciones Unarias/Binarias

Cliente (tipo_doc, nro_doc, nombre, telefono, domicilio, edad

Proveedor (CUIT, nombre, domicilio, telefono, mail

Pelicula (cod_pel, titulo, genero, CUIT_prov

Lista de claves foráneas

Pelicula.CUIT_prov → Proveedor.CUIT

Regla 2 - Relaciones Unarias/Binarias

- Si la relación es unaria, se aplica la regla de la misma manera que con las binarias, considerando a la misma tabla como la referenciada
- Si la cardinalidad de la relación fuera 1:1, se agregan los campos y la clave foránea en cualquiera de las dos tablas, **pero sólo en una de ellas**

Regla 2 - Relaciones Unarias/Binarias

Cliente (tipo_doc, nro_doc, nombre, telefono, domicilio, edad, t_doc_tit, n_doc_tit)

Proveedor (CUIT, nombre, domicilio, telefono, mail)

Pelicula (cod_pel, titulo, genero, CUIT_prov)

Lista de claves foráneas

Pelicula.CUIT_prov → Proveedor.CUIT

Cliente.t_doc_tit + nro_doc_tit → Cliente.tipo_doc + nro_doc

Regla 3 - Relaciones N:N

- Se creará una nueva tabla cuyo nombre deberá ser un sustantivo o concatenación de tablas (siempre que tenga sentido)
- Contendrá como campos a aquellos que forman las claves primarias de las tablas que vincula
- Se definirán **dos claves foráneas**, referenciando a las tablas que dieron origen a los campos correspondientes
- Todos estos campos formarán la clave primaria de la nueva tabla

Regla 3 - Relaciones N:N

Cliente (tipo_doc, nro_doc, nombre, telefono, domicilio, edad, t_doc_tit, n_doc_tit)

Proveedor (CUIT, nombre, domicilio, telefono, mail)

Pelicula (cod_pel, titulo, genero, CUIT_prov)

Alquiler (cod_pel, tipo_doc, nro_doc)

Lista de claves foráneas

Pelicula.CUIT_prov → Proveedor.CUIT

Cliente.t_doc_tit + nro_doc_tit → Cliente.tipo_doc + nro_doc

Alquiler.cod_pel → Pelicula.cod_pel

Alquiler.tipo_doc + nro_doc → Cliente.tipo_doc + nro_doc

Regla 4 - Atributos de relación

- Si llegan a existir **atributos en una relación que forma una nueva tabla**, se agregan como campos en la nueva tabla
 - Si el atributo además se encuentra marcado como identificador, el mismo formará **parte** de la clave primaria de la nueva tabla (no la reemplaza, sino que **la complementa**)
- Si los atributos son de relaciones que no forman tabla, se crean como campos nuevos en la tablas del lado de la N de la relación (o indistintamente si no existiera N)

Regla 4 - Atributos de relación

Cliente (tipo_doc, nro_doc, nombre, telefono, domicilio, edad, t_doc_tit, n_doc_tit)

Proveedor (CUIT, nombre, domicilio, telefono, mail)

Pelicula (cod_pel, titulo, genero, CUIT_prov)

Alquiler (cod_pel, tipo_doc, nro_doc, fecha)

Lista de claves foráneas

Pelicula.CUIT_prov → Proveedor.CUIT

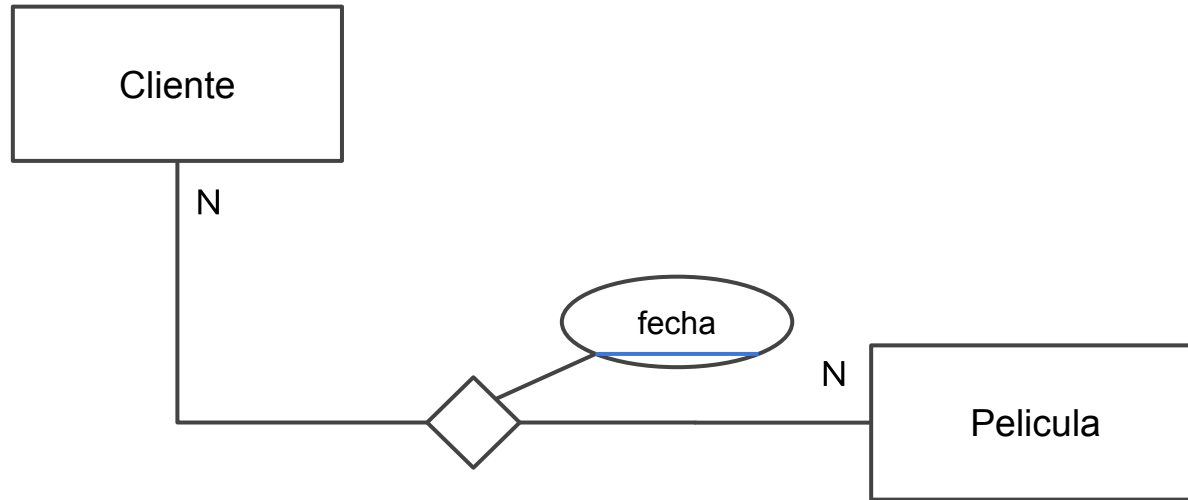
Cliente.t_doc_tit + nro_doc_tit → Cliente.tipo_doc + nro_doc

Alquiler.cod_pel → Pelicula.cod_pel

Alquiler.tipo_doc + nro_doc → Cliente.tipo_doc + nro_doc

Regla 4 - Atributos de relación

- Qué sucede si un cliente alquila la misma película más de una vez?
 - Modificamos el DER para que el atributo “fecha” sea un identificador de relación
 - El campo ahora forma parte de la clave primaria



Modelo Relacional completo

Cliente (tipo_doc, nro_doc, nombre, telefono, domicilio, edad, t_doc_tit, n_doc_tit)

Proveedor (CUIT, nombre, domicilio, telefono, mail)

Pelicula (cod_pel, titulo, genero, CUIT_prov)

Alquiler (cod_pel, tipo_doc, nro_doc, fecha)

Lista de claves foráneas

Pelicula.CUIT_prov → Proveedor.CUIT

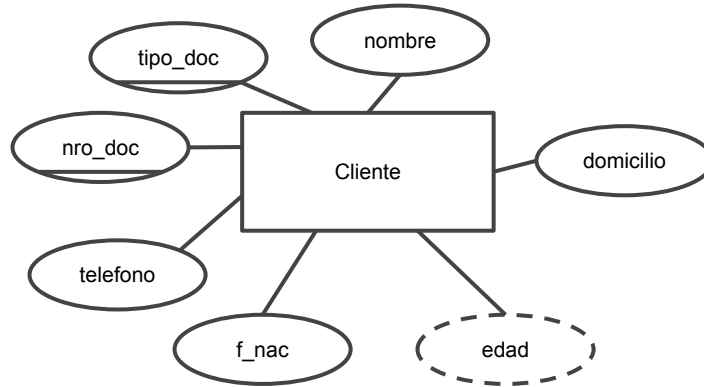
Cliente.t_doc_tit + nro_doc_tit → Cliente.tipo_doc + nro_doc

Alquiler.cod_pel → Pelicula.cod_pel

Alquiler.tipo_doc + nro_doc → Cliente.tipo_doc + nro_doc

Regla 5 - Atributos calculados

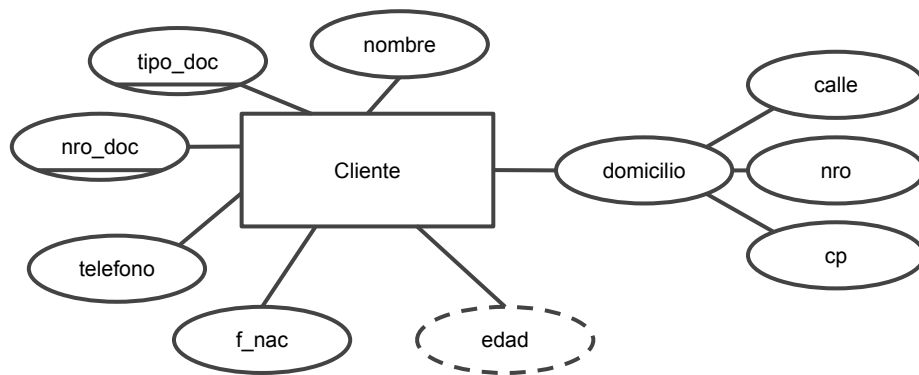
- No se trasladan al modelo relacional, ya que es un dato que no se persiste



Cliente (tipo_doc, nro_doc, nombre, telefono, domicilio, f_nac)

Regla 6 - Atributos agrupadores

- El agrupador no se traslada, pero los agrupados se prefijan con el nombre del agrupador

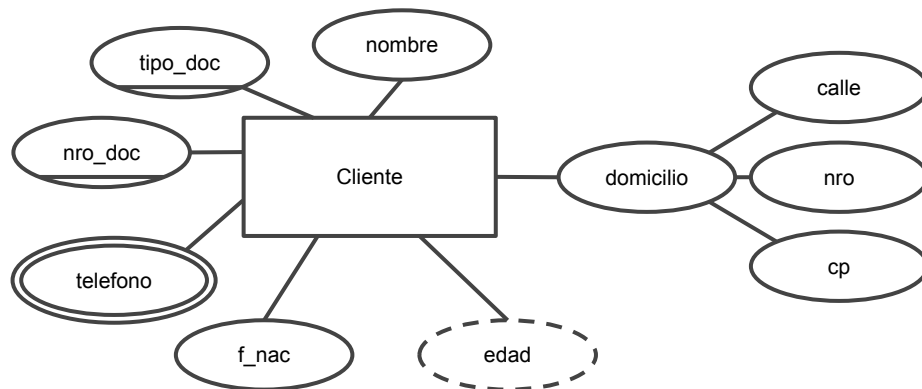


Cliente (tipo_doc, nro_doc, nombre, telefono, f_nac, domicilio_calle, domicilio_nro, domicilio_cp)

Regla 7 - Atributos multivaluados

- Se crea una nueva tabla que tendrá como campos a la clave primaria de la tabla que posee el atributo, formando la clave foránea correspondiente
- Se agrega un campo para almacenar el atributo multivaluado
- Todos los campos forman la clave primaria

Regla 7 - Atributos multivaluados



Cliente (tipo_doc, nro_doc, nombre, f_nac, domicilio_calle, domicilio_nro, domicilio_cp)

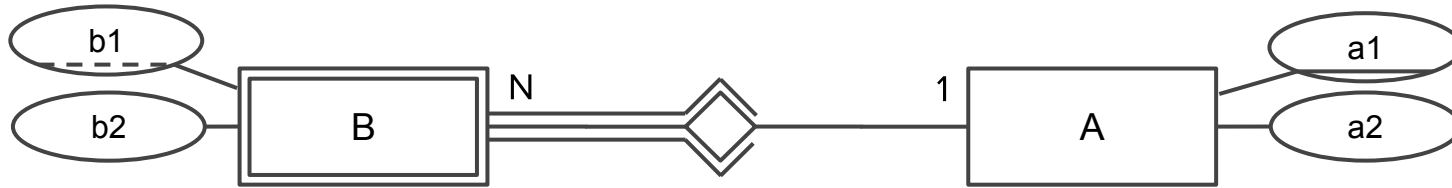
Telefono (tipo_doc, nro_doc, telefono)

Lista de claves foráneas

Telefono.tipo_doc + nro_doc → Cliente.tipo_doc + nro_doc

Regla 8 - Entidades Débiles

- Misma transformación que una fuerte, pero los campos de la clave foránea de la relación de dependencia formarán la clave primaria junto a los discriminantes



A (a1, a2)

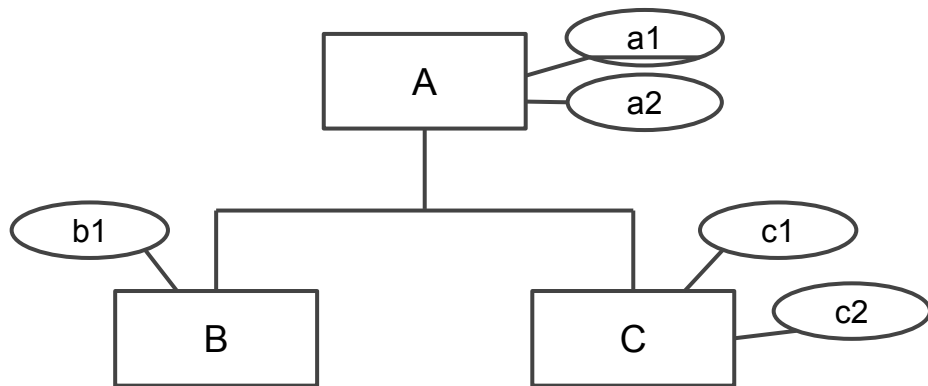
B (a1, b1, b2)

Lista de claves foráneas

B.a1 → A.a1

Regla 8 - Jerarquías

- Subentidades pasan como una fuerte, pero se adicionan los campos que forman la clave de la supraentidad, creando la clave foránea correspondiente
- La clave primaria estará formada únicamente por estos nuevos campos
- Se pierden las restricciones (solapamiento y partición)
- Atributo de tipo pasa como campo a la supraentidad



A (a1, a2)

B (a1, b1)

C (a1, c1, c2)

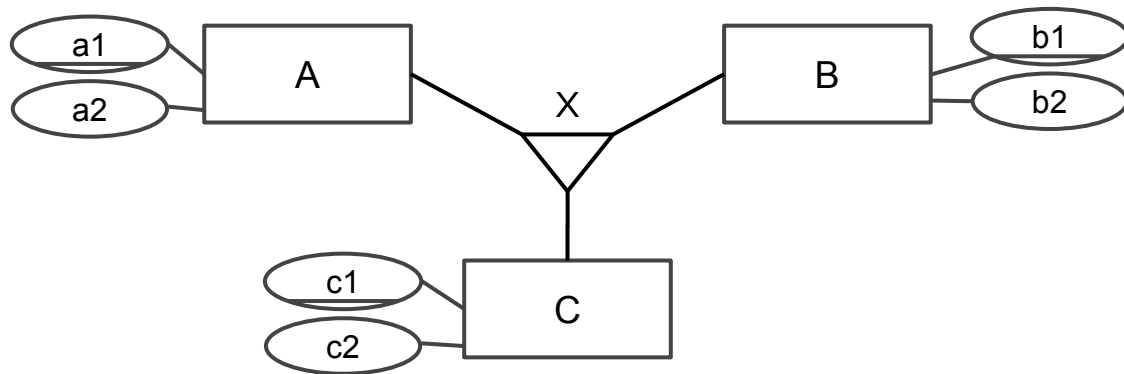
Lista de claves foráneas

B.a1 → A.a1

C.a1 → A.a1

Regla 9 - Ternarias

- Se crea una nueva tabla, teniendo como campos a las claves primarias de las tablas que relaciona, formando además 3 claves foráneas
- Si existen atributos de relación, se adicionan a la nueva tabla formando parte de la clave primaria si son identificadores



A (a1, a2)

B (b1, b2)

C (c1, c2)

X (a1, b1, c1)

Lista de claves foráneas

X.a1 → A.a1

X.b1 → B.b1

X.c1 → C.c1

Regla 9 - Ternarias

- La clave primaria se definirá según la cardinalidad de la relación con la siguiente regla: se incluyen **al menos** dos de las claves foráneas, tomando **siempre** las que tengan una **N**

N : N : N

N : N : 1

N : 1 : 1

1 : 1 : 1

Ejercicio a Resolver

Realizar el Modelo Relacional para el ejercicio del videoclub

Base de Datos I

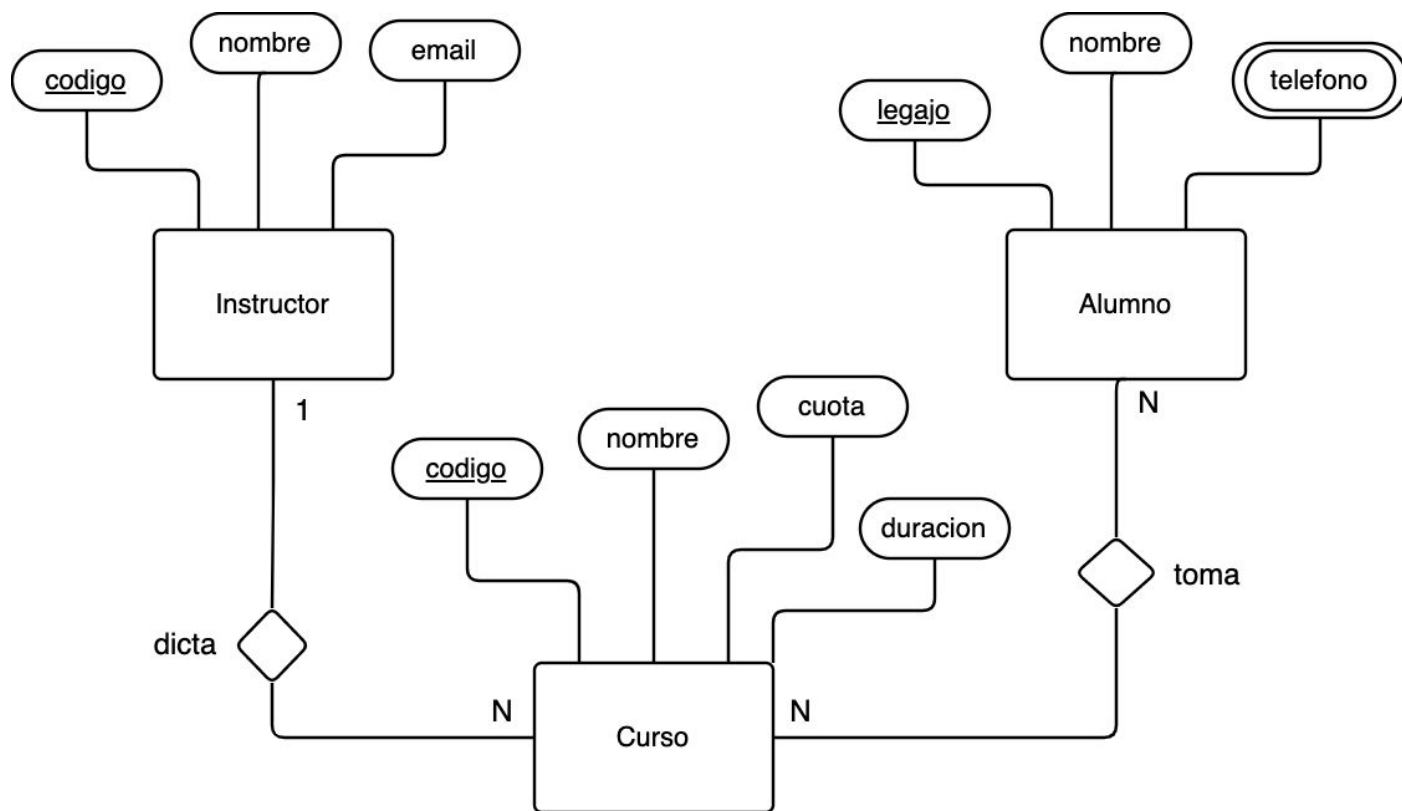
Práctica MR

Ejercicio 1

“Soy el gerente de una empresa de capacitación que brinda cursos relacionados con la informática. Dictamos varios cursos, cada uno de los cuales tiene un código, un nombre y valor de cuota. Introducción a UNIX y Programación C son dos de nuestros cursos más populares. También necesitaríamos conocer la duración de los cursos, lo cual varía de uno a cuatro días. Un instructor puede dictar varios cursos. Pablo Basile y Daniel Abadi son dos de nuestros mejores profesores. Registramos el nombre y dirección de correo de cada instructor. Cada curso es dictado por un único instructor. Nuestros instructores se encuentran capacitados para dictar diferentes cursos.

Los alumnos pueden tomar varios cursos al mismo tiempo, y muchos de ellos lo hacen. ¡Mariana Carpovich de AT&T se anota en cada curso que ofrecemos!. También registramos el nombre y números telefónicos de cada uno de nuestros alumnos (pueden poseer más de un teléfono)”

Ejercicio 1



Ejercicio 1

Instructor (codigo, nombre, email)

Curso (codigo, nombre, cuota, duracion, cod_instructor)

Alumno (legajo, nombre)

Alumno_Curso (leg_alumno, cod_curso)

Telefono (leg_alumno, telefono)

Lista de claves foráneas

Curso.cod_instructor → Instructor.codigo

Alumno_Curso.leg_alumno → Alumno.legajo

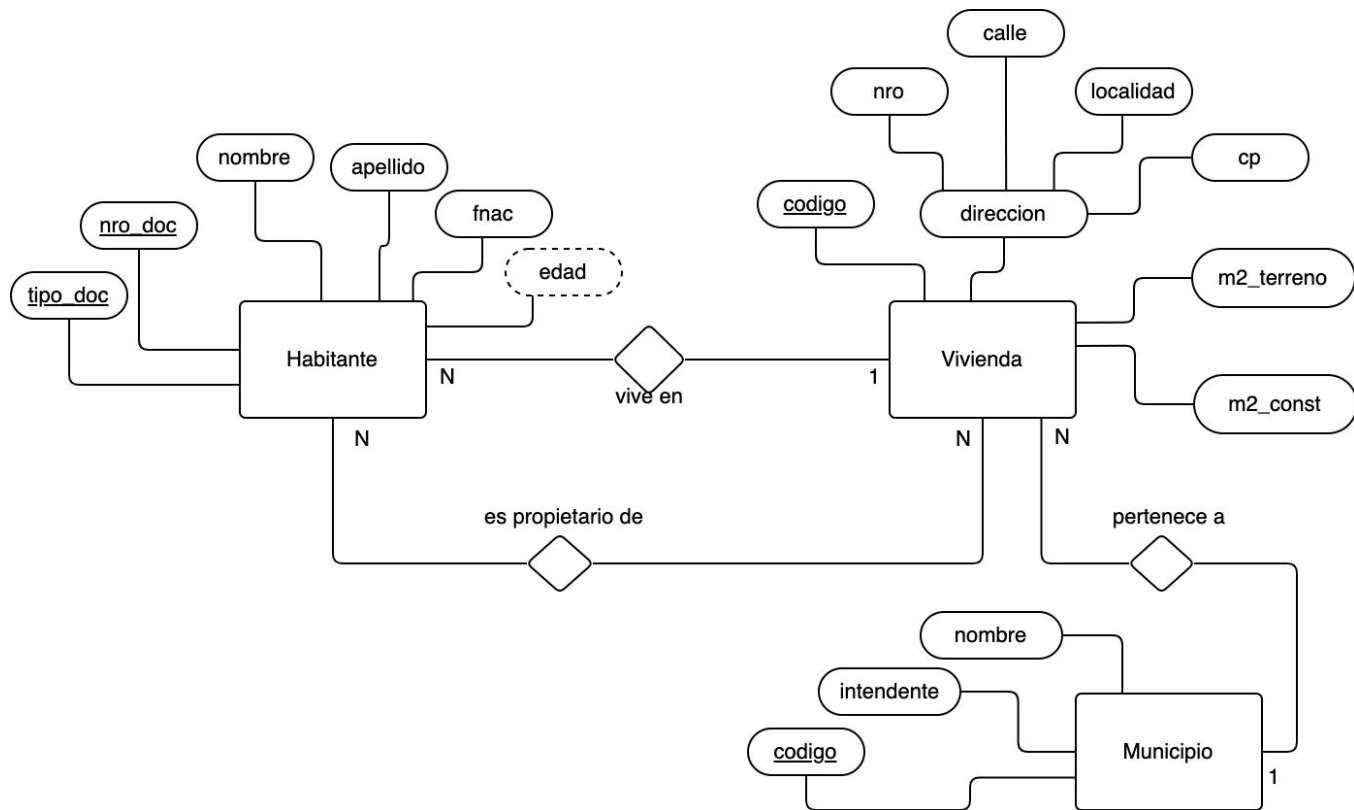
Alumno_Curso.cod_curso → Curso.codigo

Telefono.leg_alumno → Alumno.legajo

Ejercicio 2

“La Secretaría de Vivienda de la Provincia de Córdoba desea almacenar información relacionada a los habitantes y las viviendas donde residen los mismos. De los habitantes, nos interesa almacenar su tipo y número de documento, su nombre, apellido, fecha de nacimiento y edad. Sabemos que cada persona debe vivir en una única vivienda, la cual no necesariamente es de su propiedad y además, una persona puede ser propietaria de varias viviendas. Cabe destacar que existen propiedades que tienen múltiples propietarios. De las viviendas nos interesa almacenar la dirección (calle, número, localidad y código postal), la cantidad de metros cuadrados del terreno y la cantidad de metros cuadrados que se encuentran edificados, además del municipio al que pertenece. Respecto a los municipios, deseamos conocer su nombre y el intendente actual”

Ejercicio 2



Ejercicio 2

Habitante (tipo_doc, nro_doc, nombre, apellido, fnac, cod_viv)

Vivienda (codigo, dir_calle, dir_nro, dir_localidad, dir_cp, m2_terreno, m2_const, cod_mun)

Municipio (codigo, nombre, intendente)

Propiedad (tipo_doc_hab, nro_doc_hab, cod_viv)

Lista de claves foráneas

Habitante.cod_viv → Vivienda.codigo

Vivienda.cod_mun → Municipio.codigo

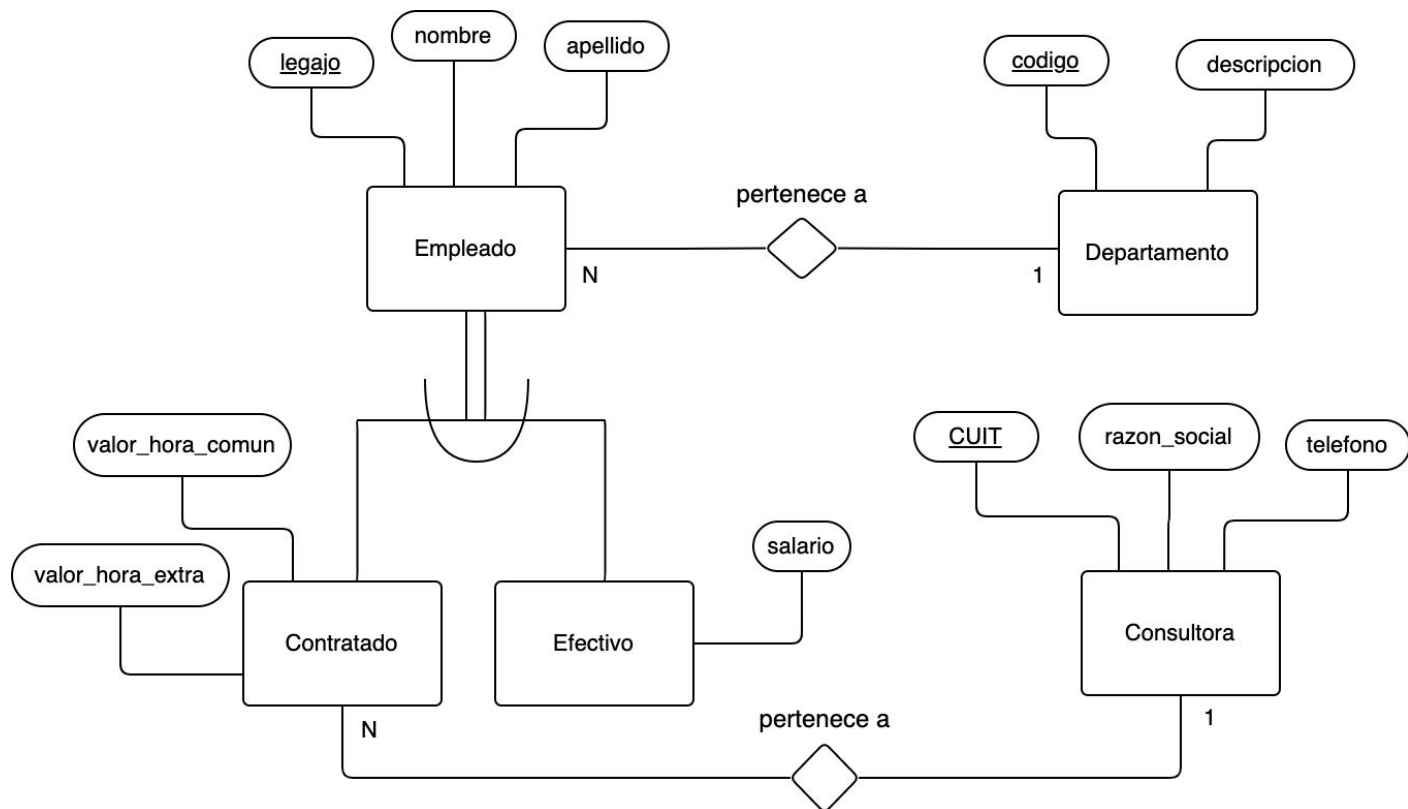
Propiedad.tipo_doc_hab + nro_doc_hab → Habitante.tipo_doc + nro_doc

Propiedad.cod_vid → Vivienda.codigo

Ejercicio 3

“Una empresa posee dos tipos de empleados: contratados y efectivos. Para todos ellos, se desea almacenar su nro. de legajo, nombre, apellido y departamento en el que trabaja. Cada departamento tiene un código y una descripción. Para los empleados efectivos, se debe almacenar su salario. Para aquellos empleados contratados, almacenar el precio que cobra por hora común y por hora extra, y la consultora a la que pertenecen. De cada consultora es necesario saber su nro. de CUIT, Razón Social y teléfono”

Ejercicio 3



Ejercicio 3

Empleado (legajo, nombre, apellido, cod_dpto)

Contratado (legajo, valor_hc, valor_he, CUIT_consultora)

Efectivo (legajo, salario)

Departamento (codigo, descripcion)

Consultora (CUIT, razon_social, telefono)

Lista de claves foráneas

Empleado.cod_dpto → Departamento.codigo

Contratado.legajo → Empleado.legajo

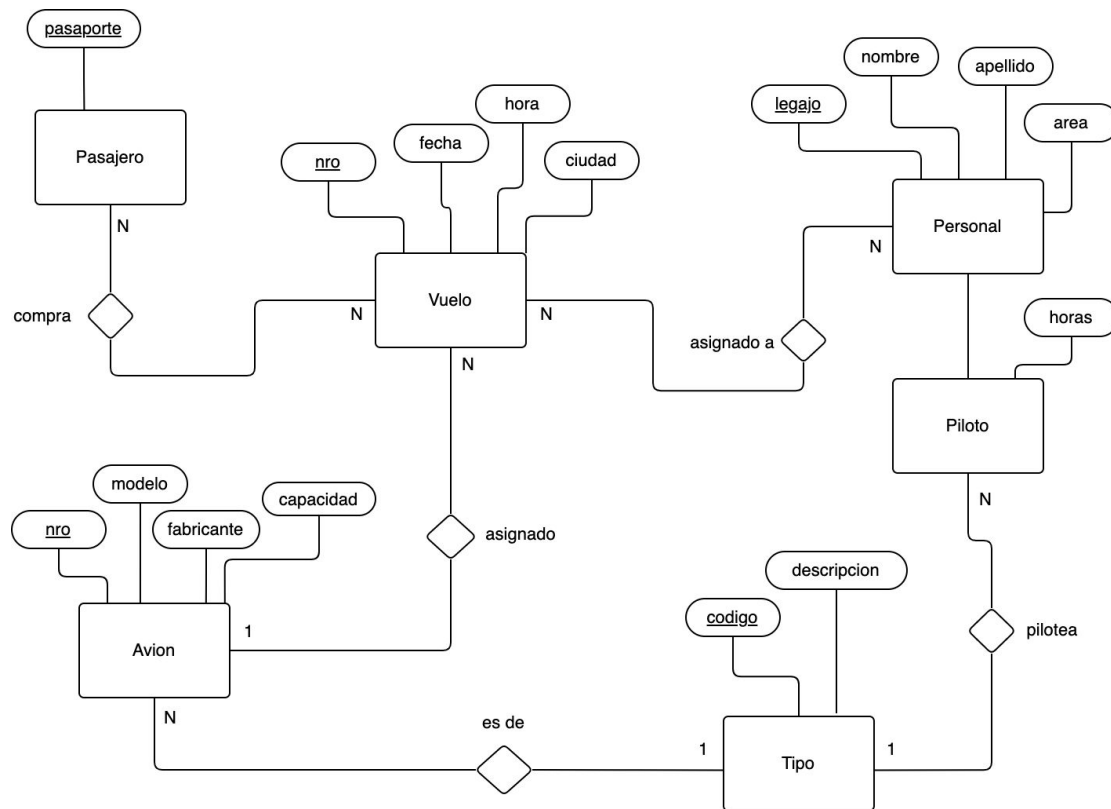
Contratado.CUIT_consultora → Consultora.CUIT

Efectivo.legajo → Empleado.legajo

Ejercicio 4

“Una aerolínea maneja información de pasajeros, vuelos y personal. Para los pasajeros se considera de interés el pasaporte y los vuelos comprados. Para los vuelos: el nro. de vuelo, fecha, hora, ciudad donde hace escala, personal asignado y avión. Para los aviones se considera modelo, fabricante, capacidad, hangar y tipo de avión. Por último, para el personal, se tiene en cuenta el nombre y apellido, área asignada, y en particular para los pilotos se conoce la cantidad de horas de vuelo y el tipo de avión que pilotea”

Ejercicio 4



Ejercicio 3

Pasajero (pasaporte)

Vuelo (nro, fecha, hora, ciudad, nro_avion)

Personal (legajo, nombre, apellido, area)

Piloto (legajo, horas, cod_tipo)

Tipo (codigo, descripcion)

Avion (nro, modelo, fabricante, capacidad, cod_tipo)

Vuelo_Personal (nro_vuelo, legajo_pers)

Pasajero_Vuelo (pasaporte, nro_vuelo)

Lista de claves foráneas

Vuelo.nro_avion → Avion.nro

Piloto.legajo → Personal.legajo

Piloto.cod_tipo → Tipo.codigo

Avion.cod_tipo → Tipo.codigo

Vuelo_Personal.nro_vuelo → Vuelo.nro

Vuelo_Personal.legajo_pers → Personal.legajo

Pasajero_Vuelo.pasaporte → Pasajero.pasaporte

Pasajero_Vuelo.nro_vuelo → Vuelo.nro

Base de Datos I

Ingeniería Inversa (MR → DER)

Ingeniería Inversa

- Definición
- Proceso
 - Múltiples resultados (unificación de criterios)
 - Pérdida
 - Ambigüedad

Ejercicio

Mecanico (id, sueldo, cod_esp)

Vendedor (id, comision)

Empleado (id, nombre, jefe, cod_loc)

Cliente (**cod_cli**, nombre, telefono)

Auto (**patente**, modelo)

Especialidad (**cod_esp**, descripcion)

Reparacion (patente, **nro**, fecha, valor)

Trabajo_Realizado (patente, **nro**, id_mec)

Venta (patente, cod_cli, id_ven, valor)

Localidad (**cod_loc**, descripcion, cod_prov)

Provincia (**cod_prov**, descripción)

Lista de FKs

Mecanico.id → Empleado.id

Mecanico.cod_esp → Especialidad.cod_esp

Vendedor.id → Empleado.id

Empleado.jefe → Empleado.id

Empleado.cod_loc → Localidad.cod_loc

Reparacion.patente → Auto.patente

Trabajo_Realizado.patente+nro → Reparacion.patente+nro

Trabajo_Realizado.id_mec → Mecanico.id

Venta.patente → Auto.patente

Venta.cod_cli → Cliente.cod_cli

Venta.id_ven → Vendedor.id

Localidad.cod_prov → Provincia.cod_prov

Regla 1 - Entidades fuertes

- Buscar tablas que tengan una PK sin claves foráneas dentro de la misma (no dependen de nadie para poder identificarse)
- Se crea una nueva entidad que tiene como atributos a los campos de la tabla, que no sean FK
- Los campos que forman la PK pasan a ser identificadores

Regla 2 - Subentidades

- Buscar tablas que tengan una PK que a su vez sea **una única** FK (no hay otros campos dentro de la PK)
- Se crea una subentidad que tiene como atributos a los campos de la tabla, que no sean ni PK ni FK
- La supraentidad quedará definida como aquella referenciada por la PK/FK

Regla 3 - Entidades débiles

- Buscar tablas que tengan una PK que contenga **una única FK + otros campos**
- Se crea una entidad débil que tiene como atributos a los campos de la tabla, que no sean ni PK ni FK
- La relación de dependencia se dará por la entidad referenciada por la FK dentro de la PK
- El resto de los campos de la PK serán discriminantes

Regla 4 - Relaciones Unarias/Binarias 1:N

- Buscar FKs dentro de una tabla, que estén fuera de la PK
- Se define una relación 1:N entre la entidad de la tabla correspondiente y aquella referenciada por la PK, quedando la N del lado donde existe la FK
- Aplica tanto para unarias como binarias

Regla 5 - Relaciones Unarias/Binarias N:N

- Buscar tablas con **2 FKs únicamente, ambas dentro de la PK**
- Se crea una relación N:N vinculando las entidades correspondientes a las tablas referenciadas por las FKs
- Si existen otros campos fuera de las FKs, se crean como atributos de relación
 - Si el campo forma parte de la PK, se lo marca como identificador

Regla 6 - Ternarias

- Buscar tablas con **3 FKs únicamente, al menos 2 de ellas dentro de la PK**
- Se crea una relación ternaria vinculando las entidades correspondientes a las tablas referenciadas por las FKs
- La cardinalidad se define poniendo una N para aquellas FK que forman parte de la PK y un 1 para el resto
- Si existen otros campos fuera de las FKs, se crean como atributos de relación
 - Si el campo forma parte de la PK, se lo marca como identificador

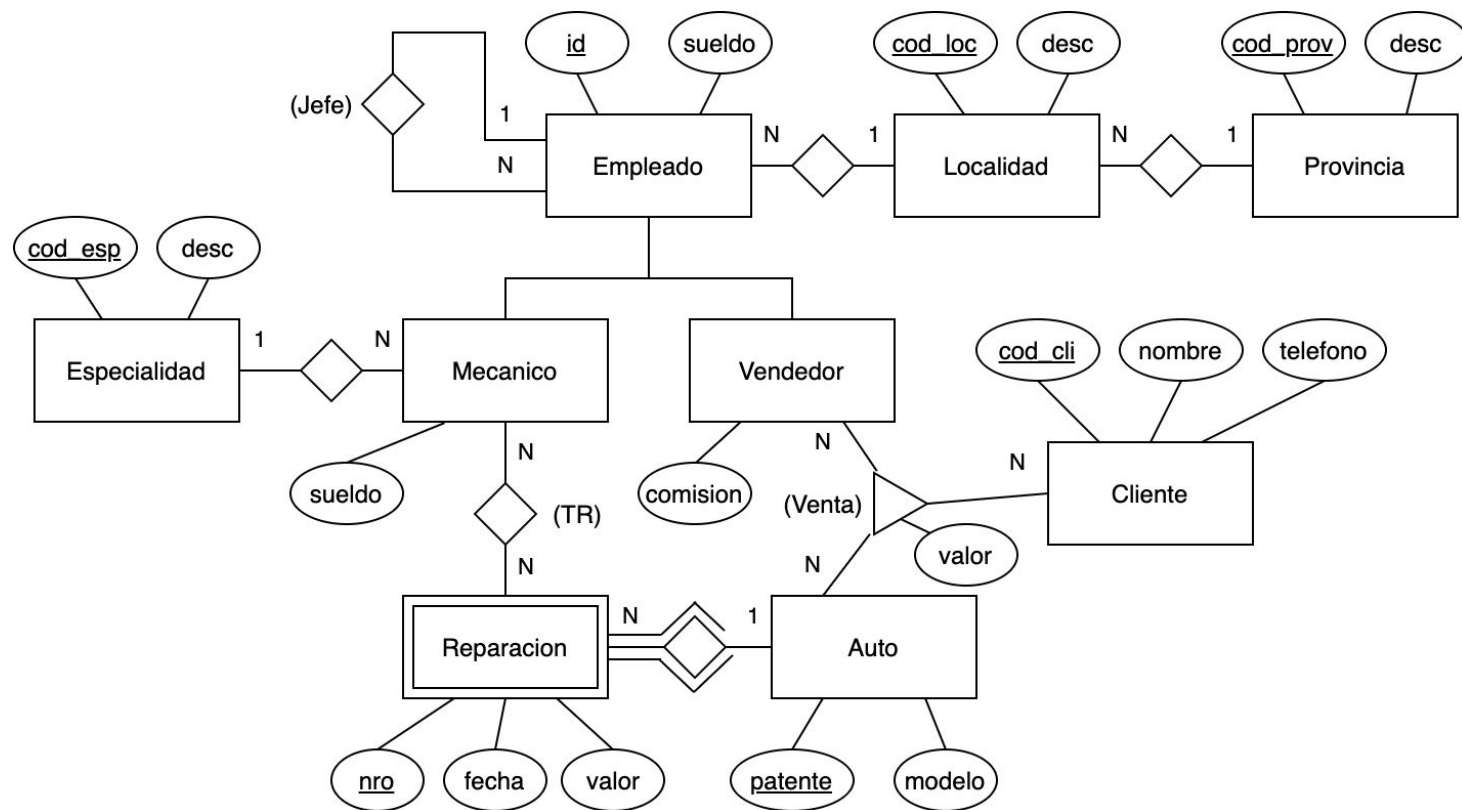
Pérdida

- Atributos calculados
- Atributos agrupadores
- Restricciones de jerarquías (solapamiento/partición)

Ambigüedad

- Binarias/Unarias 1:1 (1:N)
- Ternarias 1:1:1 y 1:1:N (1:N:N)
- Atributos de relaciones 1:1 o 1:N (atributo de entidad)
- Entidad débil con sólo un discriminante (atributo multivaluado)
- Atributo de tipo en jerarquía (atributo de supraentidad)

Ejercicio



Base de Datos I

Práctica Ingeniería Inversa (MR → DER)

Ejercicio 1

A (a1, a2, a3) FK=(a1 \rightarrow E.e1) , (a3 \rightarrow E.e1)

B (b1, **b2**, b3) FK=(b1 \rightarrow G.g1)

C (**c1**, c2, c3) FK=(c3 \rightarrow C.c1)

D (d1, d2, d3, **d4**) FK=(d1 \rightarrow E.e1), (d2 \rightarrow C.c1), (d3 \rightarrow G.g1)

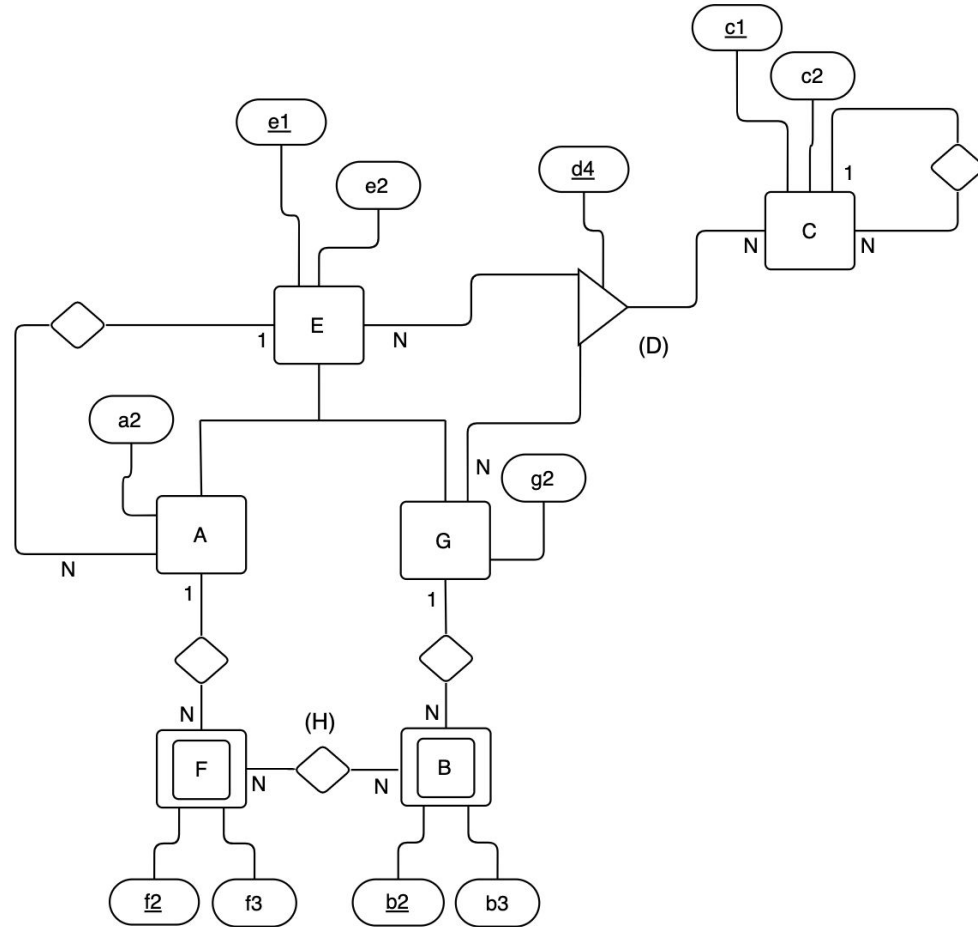
E (**e1**, e2)

F (f1, **f2**, f3) FK=(f1 \rightarrow A.a1)

G (g1, g2) FK=(g1 \rightarrow E.e1)

H (h1, h2, h3, h4) FK=(h1 + h2 \rightarrow F.f1 + F.f2), (h3 + h4 \rightarrow B.b1 + B.b2)

Ejercicio 1



Ejercicio 2

$R_1(\mathbf{A1}, A2)$

$R_2(\mathbf{A1}, B1, B2)$

$FK=(A1 \rightarrow R1.A1)$

$R_3(\mathbf{C1}, C2, \mathbf{A1})$

$FK=(A1 \rightarrow R1.A1)$

$R_4(\mathbf{D1}, D2)$

$R_5(\mathbf{L1}, L2, L3)$

$R_6(\mathbf{C1}, \mathbf{K1}, \mathbf{L1})$

$FK=(C1 \rightarrow R3.C1), (K1 \rightarrow R11.K1), (L1 \rightarrow R5.L1)$

$R_7(\mathbf{A1}, E2, E3)$

$FK=(A1 \rightarrow R1.A1)$

$R_8(\mathbf{A1}, F1, F2)$

$FK=(A1 \rightarrow R1.A1)$

$R_9(\mathbf{D1}, \mathbf{C1})$

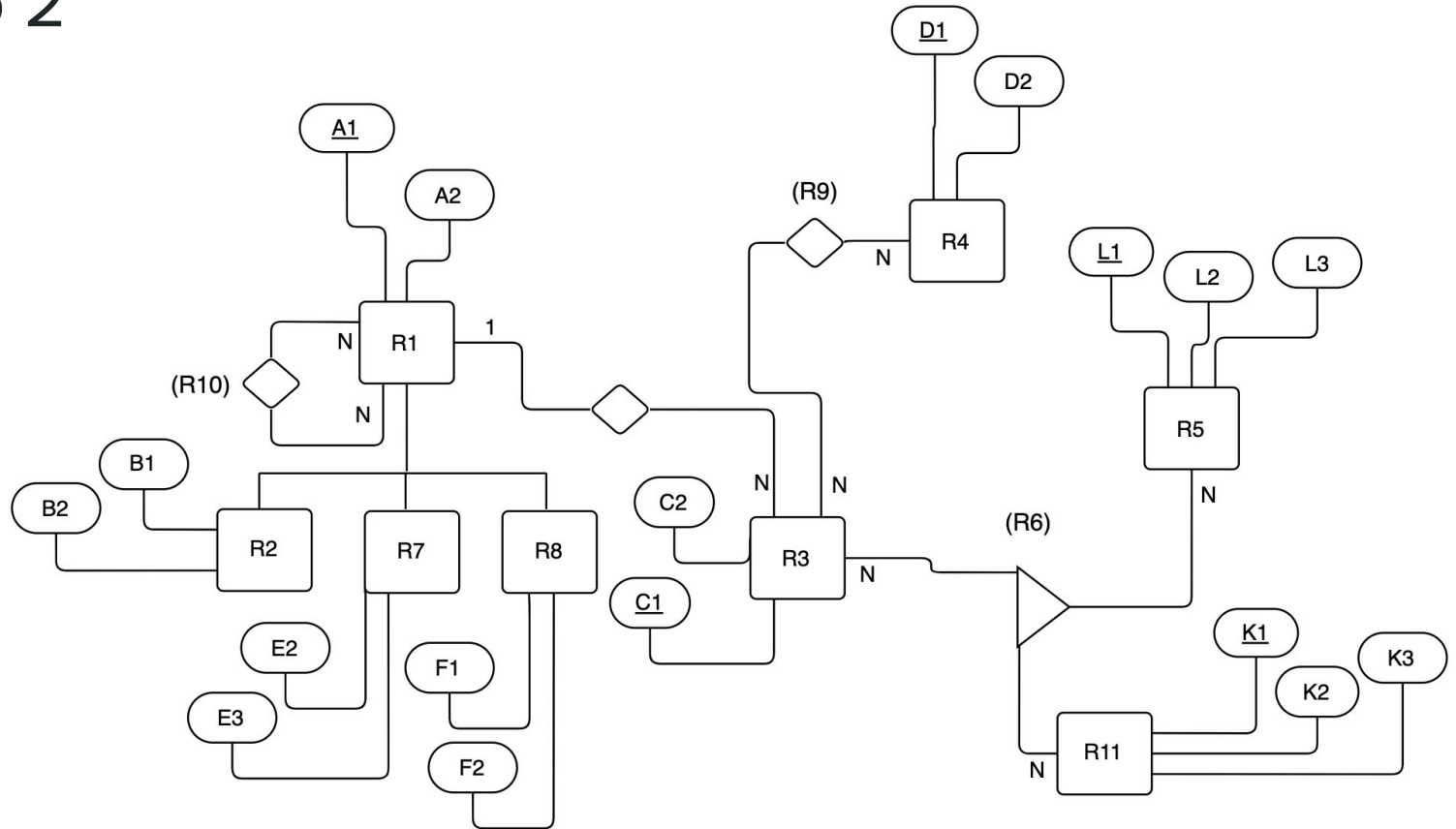
$FK=(C1 \rightarrow R3.C1), (D1 \rightarrow R4.D1)$

$R_{10}(\mathbf{A11}, \mathbf{A12})$

$FK=(A11 \rightarrow R1.A1), (A12 \rightarrow R1.A1)$

$R_{11}(\mathbf{K1}, K2, K3)$

Ejercicio 2



Base de Datos I

Normalización

Normalización

Factura

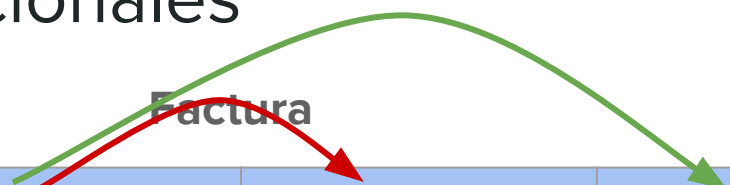
<u>Nro</u>	Fecha	CUIT	Nombre	Telefono	Importe
1	12/01/2020	11111111	Juan Perez	1144552445	1000
2	12/01/2020	2222222	María Sanchez	1156467254	2000
3	13/01/2020	3333333	Javier Rodriguez	1134227685	2000
4	13/01/2020	11111111	Juan Perez	1144552445	1500
5	14/01/2020	11111111	Juan Perez	1144552445	4000

Normalización

- Problemas del modelo anterior
 - Redundancia
 - Inconsistencia

“Mediante la normalización buscamos obtener un esquema de base de datos con redundancia mínima”

Dependencias Funcionales



<u>Nro</u>	Fecha	CUIT	Nombre	Telefono	Importe
1	12/01/2020	11111111	Juan Perez	1144552445	1000
2	12/01/2020	2222222	María Sanchez	1156467254	2000
3	13/01/2020	3333333	Javier Rodriguez	1134227685	2000
4	13/01/2020	11111111	Juan Perez	1144552445	1500
5	14/01/2020	11111111	Juan Perez	1144552445	4000

CUIT → Nombre

CUIT → Telefono

Normalización

Factura

<u>Nro</u>	Fecha	<u>CUIT</u>	Importe
1	12/01/2020	11111111	1000
2	12/01/2020	22222222	2000
3	13/01/2020	33333333	2000
4	13/01/2020	11111111	1500
5	14/01/2020	11111111	4000

Cliente

<u>CUIT</u>	Nombre	Telefono
11111111	Juan Perez	1144552445
22222222	María Sanchez	1156467254
33333333	Javier Rodriguez	1134227685

Normalización

- Se aplica sobre el modelo relacional
- El proceso requiere conocer las **dependencias funcionales**
- La **forma normal** establece el nivel de redundancia requerido
 - 1FN
 - 2FN
 - 3FN
 - FNBC (Boyce-Codd)

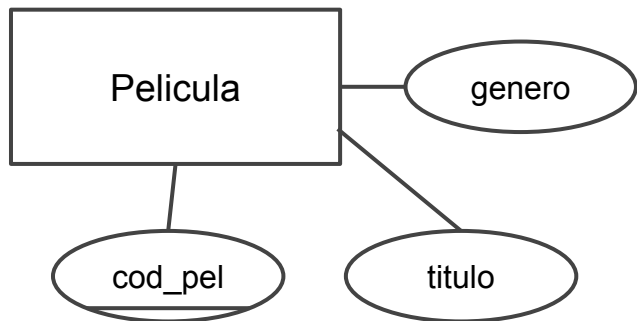
Claves primarias no numéricas

- Redundancia en espacio de almacenamiento cuando posee referencias (FKs)
- Se puede solucionar eligiendo una clave numérica

Claves primarias compuestas

- Traslada complejidad cuando posee referencias (FKs compuestas)
- Se puede solucionar eligiendo una clave simple

Normalización de atributos



Pelicula

<u>cod_pel</u>	titulo	genero
1000	Volver al futuro	C. Ficción
1001	Volver al futuro 2	Ciencia Ficción
1002	La llamada	Terror
1003	Duro de matar	Acción
1004	Duro de matar 2	Accion
1005	Duro de matar 3	Axión

Normalización de atributos

- Problemas del modelo anterior
 - Redundancia
 - Inconsistencia

Normalización de atributos

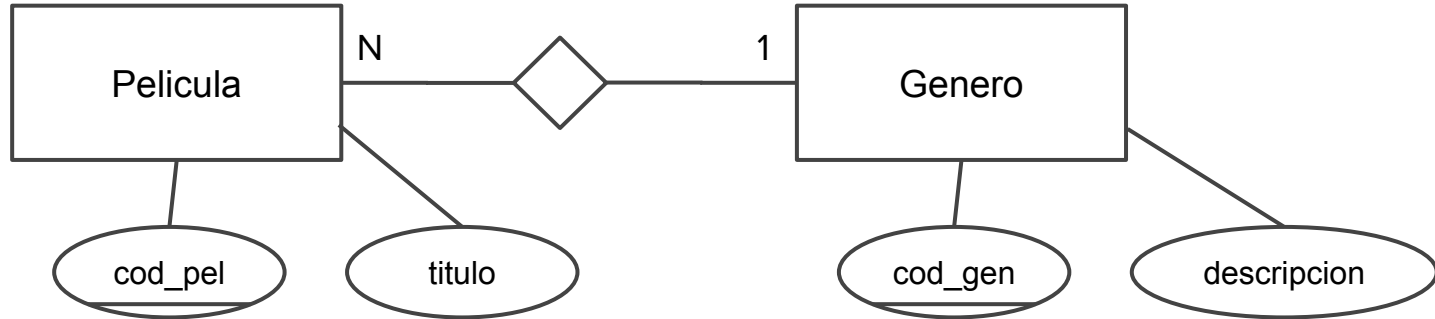
Pelicula

<u>cod_pel</u>	titulo	<u>cod_gen</u>
1000	Volver al futuro	1
1001	Volver al futuro 2	1
1002	La llamada	2
1003	Duro de matar	3
1004	Duro de matar 2	3
1005	Duro de matar 3	3

Genero

<u>cod_gen</u>	descripcion
1	Ciencia Ficción
2	Terror
3	Acción

Normalización de atributos



Normalización de atributos

- Determinar el grado de repetición o probabilidad de encontrar el mismo valor para el atributo en dos instancias diferentes del conjunto de entidades
 - Medio/Alto → Candidato a ser normalizado como entidad
 - Bajo/Nulo → Queda como atributo
- Incrementa la complejidad para la resolución del dato

Desnormalización

- Proceso por el cual se incorpora redundancia a un esquema de base de datos
- Normalmente se efectúa sobre bases de datos estáticas (históricas)
- Ventajas de uso
 - Reducción de complejidad en consultas
 - Velocidad de procesamiento
- Pueden realizarse sobre bases de datos relacionales o específicas (ej: OLAP)
- Comúnmente utilizado en sistemas ad-hoc
 - Inteligencia de negocios
 - Data Warehouse
 - Data Mining

Ejercicio a Resolver

Normalizar los atributos del ejemplo de DER extendido del videoclub