



## 1) ATRIBUTOS COMPUESTOS

1.1) El atributo compuesto domicilio se resuelve generando un único atributo que concatene todos los atributos simples del atributo compuesto y los atributos simples pierden la identidad. Decidimos concatenar los atributos en uno solo porque en el discurso no se pide de forma explícita el domicilio exacto del usuario, no es un atributo que se utilice para cosas importantes.

## 2) ATRIBUTOS POLIVALENTES

2.1) El atributo polivalente TARJETA de la entidad USUARIO se convierte en una nueva entidad débil y se agrega una interrelación "posee" de cardinalidad (0,1) para la cantidad de tarjetas y otra cardinalidad (1,n) para el USUARIO.

2.2) El atributo polivalente cant\_canciones de la entidad LISTA se convierte en una nueva entidad débil y se agrega una interrelación "lista\_cant\_canciones" de cardinalidad (1,n) para la cantidad de canciones y otra cardinalidad (1,n) para LISTA.

## 3) JERARQUÍAS

3.1) Respecto a la jerarquía que existe entre la superentidad USUARIO y las subentidades INTERPRETE y CONSUMIDOR, se opta por mantener todo y hacer explícitas las relaciones "es\_un\_interprete" y "es\_un\_consumidor" de la herencia, ya que la subentidad INTERPRETE tiene comportamientos distintos a la de CONSUMIDOR.

Al eliminar las subentidades no podríamos ver claramente el comportamiento cargando a la superentidad con relaciones no obligatorias y atributos innecesarios.

Por otro lado, eliminar a la superentidad, generaría subentidades con muchos atributos y relaciones repetidas en el esquema.

## 4) ATRIBUTOS DERIVADOS

- 4.1) <cant\_likes> mantenemos el atributo derivado en el esquema, dejamos que se almacene y lo actualizamos debido a que la plataforma constantemente utiliza esta información para procesos como recomendación del contenido entre otros.
- 4.2) <reproduc\_totales> mantenemos el atributo derivado en el esquema, dejamos que se almacene y lo actualizamos debido a que la plataforma constantemente utiliza esta información para procesos como recomendación del contenido entre otros.
- 4.3) <antigüedad> este atributo calculado lo retiramos del esquema y lo calculamos debido a que no es un atributo que le proporcione mucha información de interés a la plataforma.

## **FASE II - Modelo Relacional**

### **Entidades:**

**USUARIO** (id\_usuario, correo\_electronico, nacionalidad, domicilio, genero, tipo\_membresia)

PK(id\_usuario), CK(correo\_electronico)

**TARJETA** (id\_tarjeta, num\_tarjeta, id\_usuario)

PK(id\_tarjeta), CK(nombre, id\_usuario), FK(id\_usuario, USUARIO)

**MEDIO\_DE\_PAGO** (id\_mediodepago, codigo\_op, cbu, num\_tarjeta, monto)

PK(id\_mediodepago), CK (codigo\_op)

**INTERPRETE** (id\_interprete, cant\_contenido, cant\_usuarios, cant\_suscriptores)

PK(id\_interprete), FK (id\_interprete, USUARIO)

**CONSUMIDOR** (id\_consumidor, intereses)

PK(id\_consumidor), FK (id\_consumidor, USUARIO)

**LISTA** (id\_lista, url\_lista, tipo\_lista, cant\_canciones, nombre, privacidad, id\_genero)

PK(id\_lista), CK(url\_lista), FK(id\_usuario, USUARIO), FK(id\_genero, GENERO)

**CONTENIDO** (id\_contenido, url, tipo\_contenido, nombre, duracion, comentarios, cant\_likes, reproduc\_totales id\_genero, id\_album, id\_inteprete)

PK(id\_contenido), CK(url), FK(id\_interprete, INTERPRETE), FK (id\_album, ALBUM), FK (id\_genero, GENERO)

**ALBUM** (id\_album, cant\_pistas, antigüedad, fecha\_lanzamiento, nombre, id\_interprete)

PK(id\_album), CK(nombre, id\_interprete), FK(id\_interprete, INTERPRETE)

**GENERO** (id\_genero, nombre)

PK (id\_genero), CK(nombre)

### **Interrelaciones:**

**es\_suscriptor** (es\_seguido, sigue\_a)

PK (es\_seguido + sigue\_a), FK(es\_seguido, USUARIO), FK(sigue\_a, USUARIO)

**asociados** (esta\_asociado, es\_asociado, id\_usuario)

PK (esta\_asociado + es\_asociado), FK(esta\_asociado, USUARIO),  
FK(es\_asociado, USUARIO)

**es\_subgenero** (id\_genero, id\_subgenero)

PK(id\_genero), FK(id\_genero, GENERO), FK(id\_subgenero, GENERO)

**posee** (id\_posee, id\_usuario)

PK(id\_posee), FK (id\_usuario, USUARIO)

**es\_agregado** (id\_agregado, id\_lista, id\_contenido)

PK (id\_agregado), CK (id\_lista + id\_contenido), FK(id\_lista, LISTA),

FK(id\_contenido, CONTENIDO)

**abona** (id\_abona, id\_usuario, id\_mediodepago)

PK(id\_abona), CK(id\_usuario + id\_mediodepago), FK(id\_usuario, USUARIO),

FK(id\_mediodepago, MEDIO\_DE\_PAGO)

**califica** (id\_califica, id\_usuario, id\_contenido)

PK(id\_califica), CK(id\_usuario + id\_contenido), FK(id\_usuario, USUARIO),

FK(id\_contenido, CONTENIDO)