

Profesor:	DAVID APARICIO ESCRIBANO	Grupo	81
Alumno/a:	MARIO RAMOS SALSON	NIA:	100495849
Alumno/a:	VICTOR MARTINEZ DE LAS HERAS	NIA:	100495829
Alumno/a:	MIGUEL FIDALGO GARCIA	NIA:	100495770

1. Introducción

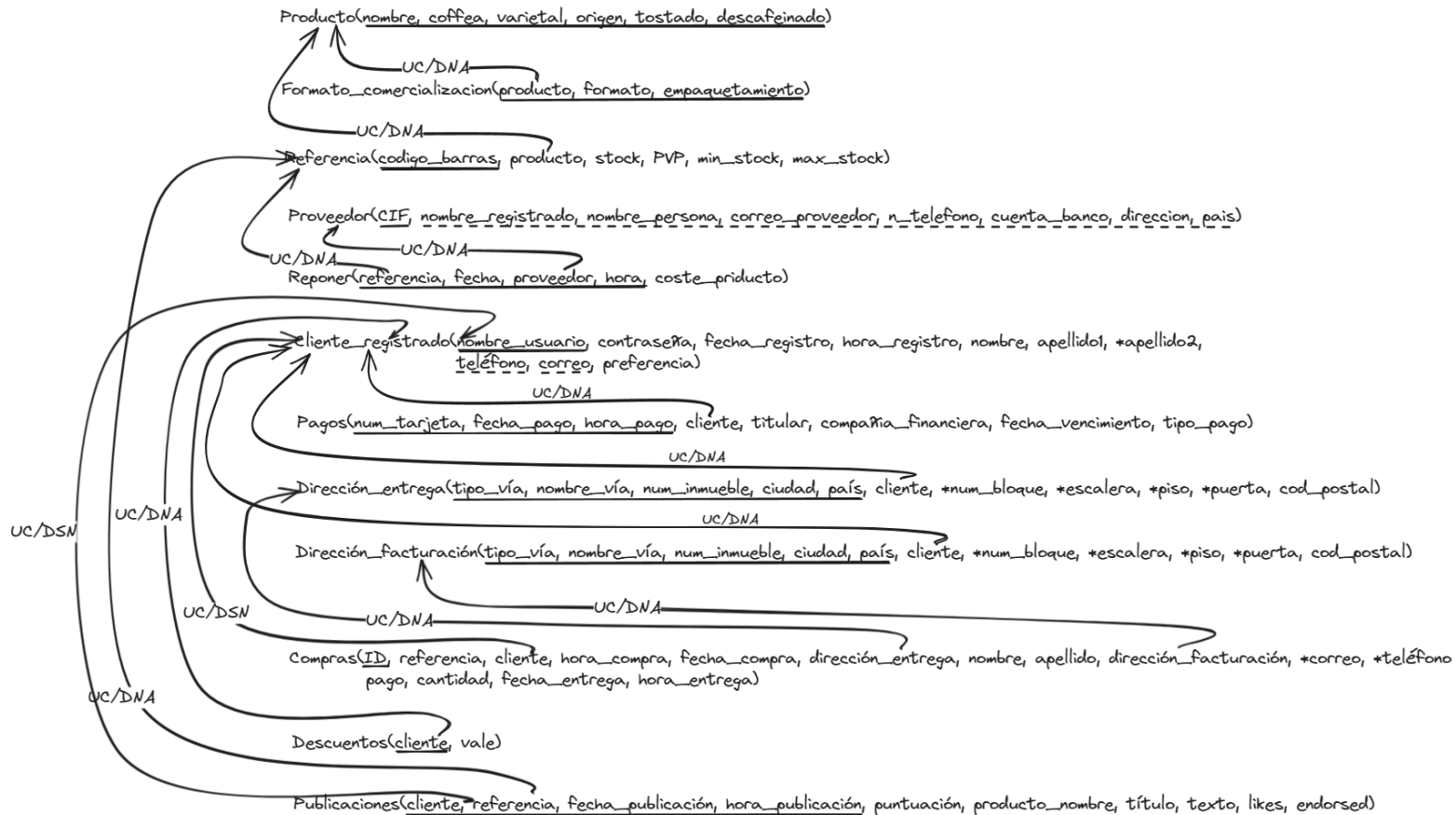
La empresa de comercio electrónico *Caffeine Maniacs* TM Inc. necesita una nueva base de datos para registrar y gestionar su información. Debido a la mala organización de su base de datos, nos han pedido implementar un modelo más actual y correcto. Para ellos hemos realizado un diseño relacional donde indicamos las tablas necesarias y la distribución de sus atributos. Además, especificamos la semántica implícita y explícita acerca del diseño.

Entre las partes más fundamentales del trabajo está el traspasar toda la información de su base de datos antigua y desactualizada a la nueva. Esto presentará obviamente problemas a la hora de llevar a cabo la transición, debido a la gran diferencia estructural que presentan ambas bases de datos. No obstante documentaremos todos estos errores para poder explicar mejor el resultado final de la práctica.

2. Diseño Relacional

Esta sección se subdivide en tres apartados:

- Esquema relacional:



- Semántica implícita: supuestos semánticos que, por referirse a información ausente en la descripción explícita (es decir, no se encuentran en el enunciado), es necesario añadir para completar el diseño.

Sup_id	Mecanismo	Descripción
I1	Añadir ID en compras.	Hemos añadido un ID compras para poder tener una clave primaria, debido a que el valor del cliente puede ser NULL ya que existen clientes no registrados.
I2	Eliminación del DECAF PROCESS.	El proceso de descafeinado no se pedía a la hora de representar el diagrama por lo que no lo hemos añadido.
I3	Eliminación de la fecha de descuento.	El dato de la fecha de descuento no se encontraba en las bases de datos de la empresa, por lo que no hemos podido añadirla.

I4	Vales para los clientes.	Al no tener fecha para los vales, consideramos que un cliente no puede tener un vale de descuento a la vez.
I5	Clave primaria.	Las reposiciones están identificadas por el código de barras, la fecha, el CIF del proveedor y la hora a la que se realizó.
I6	Precio de reponer	A la hora de elegir el proveedor para reponer los productos, en caso de encontrar dos productos exactamente iguales pero con diferente precio, elegiremos el que sea más caro.
I7	Base_price	Como solo nos pedían guardar dos precios, el que le pagamos al proveedor y el que el cliente nos paga a nosotros, hemos decidido no añadir el atributo “base_price” ya que su valor es igual al de “retail price” o “PVP”.
I8	Borrado y actualización	Suponemos que, a menos de que digan lo contrario, el borrado será de no acción (DNA) y la actualización en cascada (UC).

Tabla 1: Semántica implícita

- Semántica explícita no contemplada en el diseño: supuestos semánticos indicados en el enunciado que no han podido representarse en el esquema relacional. Para cada uno de los supuestos, crea una fila en la tabla presentada a continuación.

Sup_id	Descripción
S ₁	A la hora de establecer el correo electrónico o el número de teléfono de un usuario registrado, no hemos podido representar la condición de “uno de ellos o los dos a la vez”. Por ello hemos tomado como decisión de diseño que a la hora de insertar esos datos, habrá algo que obligue al cliente a rellenar al menos uno de los dos campos.
S ₂	No hemos podido representar en el esquema relacional todas las actualizaciones y cambios que dependen del estado del pedido, ya que los cambios se llevan a cabo a la hora de realizar UPDATES, y estos son complicados de representar en un diagrama.
S3	No hemos podido representar en el diagrama la fecha del cupón porque esos datos no constaban en la fase de datos de la empresa.
S4	La preferencia de contacto se ha establecido a ‘SMS’ por defecto.
S5	Los likes de las publicaciones no han de superar los 1 000 000 000.
S6	La puntuación acerca de un producto tiene que estar entre 1 y 5, ambos inclusive.
S7	Los clientes registrados tienen que tener al menos una dirección.
S8	Los clientes registrados no pueden tener más de una dirección por localidad.
S9	Hemos tenido que añadir el número de teléfono y el correo electrónico como opcionales ambas para poder meter una de ellas a NULL a pesar de que son únicas en la vida real.
S10	No hemos podido añadir el atributo “estado”, ya que no se encontraba esa información en la base de datos.

Tabla 2: Semántica explícita no contemplada

3. Implementación de la Estática Relacional en SQL (LDD)

El código de SQL Oracle para la creación de las tablas se encuentra en el fichero *NEWcreation.sql*.

Semántica explícita re-incorporada: Incluir aquellos supuestos de la Tabla 2 que se han podido contemplar con las sentencias de definición de SQL.

Sup id	Descripción de la solución
S4	La preferencia de contacto está establecida a “sms”. DEFAULT ‘SMS’
S5	Los likes de las publicaciones no han de superar los 1 000 000 000. CONSTRAINT ck_publicaciones CHECK (likes < 1000000000)
S6	La puntuación acerca de un producto tiene que estar entre 1 y 5, ambos inclusive. CHECK (0 < puntuacion AND puntuacion < 6)

Tabla 3: Semántica explícita re-incorporada

Semántica implícita: (continúa la numeración donde terminó en la tabla 1).

No nos hemos visto obligados a comprobar información la cual no se pedía en el enunciado, ya que el único problema que nos ha surgido que se podía llegar a solucionar de dicha manera era el hecho de que el correo y el teléfono no podían ser NULL a la vez. Pero además esto sí se indicaba en el enunciado y lo hemos resuelto con una decisión de diseño en la cual damos por hecho que no se van a insertar nunca los dos a la vez como NULL.

Tabla 1(cont.): Semántica implícita

Semántica excluida: Al crear la base de datos en SQL específico del SGBD Oracle puede que no se hayan podido contemplar algunas restricciones semánticas explícitas (tabla 2 – tabla 3), o implícitas que no han podido incorporarse (tabla 1).

Sup_id	Descripción semántica	Motivo	Explícita/ Implícita
E1	Las modificaciones en cascada definidas en el grafo relacional.	El SGBD Oracle 11g no contempla esta regla de integridad.	Implícita
E2	Los comentarios de los clientes registrados pasan a ser comentarios de clientes no registrados, una vez que dicho cliente registrado es eliminado de la base de datos.	No hemos sido capaces de realizar dicha tarea con los conocimientos actuales.	Explícito
E3	Cuando se borra un cliente registrado de la base de datos, sus compras han de pasar a figurar como compras de cliente no registrado.	No hemos sido capaces de realizar dicha tarea con los conocimientos actuales.	Explícito

E4	Actualizar el estado del pedido en función de su progreso.	A pesar de que al final no hemos añadido el atributo estado, no hemos sido capaces de realizar dicha tarea con los conocimientos actuales.	Explícito
E5	Actualización de las unidades disponibles para la referencia.	No hemos sido capaces de realizar dicha tarea con los conocimientos actuales.	Explícito
E6	Pedido automático para reponer una referencia cuando está lleva a su stock mínimo.	No hemos sido capaces de realizar dicha tarea con los conocimientos actuales.	Explícito
E7	Establecer el stock máximo 10 unidades por encima del stock mínimo	No hemos sido capaces de realizar dicha tarea con los conocimientos actuales.	Explícito
E8	En caso de empate a la hora de elegir proveedor, distinguir por el tiempo de satisfacción de entrega.	No hemos sido capaces de realizar dicha tarea con los conocimientos actuales.	Explícito
E9	En caso de otro empate en el proveedor, se elegirá el que más pedidos haya hecho en el último año.	No hemos sido capaces de realizar dicha tarea con los conocimientos actuales.	Explícito

Tabla 4: Semántica excluida en la creación de tablas

4. Carga de datos (LMD)

El código de SQL Oracle para la inserción de información se encuentra en el fichero *NEWcreation.sql*.

El orden en el que se crean las tablas está definido por las claves ajenas de cada una de ellas. En los primeros diseños de nuestro esquema las flechas de las claves ajenas no tenían un orden definido y apuntaban a tablas por encima y debajo de ellas. Reordenándolas para que solo apunten a las que están por encima, tenemos el orden de creación de las tablas de nuestra base de datos.

Al insertar todos los datos desde otras tablas ajenas a nuestra base de datos, el orden de inserción no ha importado.

Problemas con las inserciones:

1. Problema con la inserción de producto:

Al intentar introducir los datos nos daba el error de clave principal violada. Este error ocurre cuando se intentan insertar dos filas con las mismas claves primarias. En producto nos dimos cuenta de que había filas con las mismas claves primarias pero con diferentes datos en otras columnas, por ejemplo en empaquetado. Lo solucionamos poniendo empaquetado como clave principal, ya que si hubiéramos hecho una agrupación por las antiguas primarias se habrían perdido datos.

2. Problema con la inserción de reponer:

En la tabla ‘reponer’ pasaba algo parecido que en ‘producto’, había filas con las mismas claves primarias. La diferencia era que en ‘reponer’, había filas que mostraban que un mismo producto puede ser repuesto por el mismo proveedor, el mismo día y a la misma hora con dos precios diferentes, lo que es imposible, por lo que no podíamos poner precio como clave primaria. Para resolverlo, optamos por coger el máximo del precio de las columnas con las mismas claves primarias.

3. Problemas con la inserción de direcciones:

En la primera implementación del diseño relacional, pensamos que sería mejor unir las direcciones de pedido y las de facturación en una sola tabla llamada ‘direcciones’. Intentamos hacer una unión de las columnas para que salieran todas las tuplas que contenían las direcciones de pedido tanto como las de facturación, pero salía violación de clave primaria. Para quitar los duplicados hicimos una agrupación, pero había algunas tuplas que a pesar de tener las mismas claves primaria no se agrupaban, como se muestra a continuación:

```

WITH aux AS (SELECT DISTINCT DLIV_WAYTYPE, DLIV_WAYNAME, USERNAME, DLIV_GATE, DLIV_BLOCK, DLIV_STAIRW, DLIV_FLOOR,
DLIV_DOOR, DLIV_ZIP, DLIV_TOWN, DLIV_COUNTRY
FROM fsdb.trolley

WHERE DLIV_WAYTYPE is NOT NULL and DLIV_WAYNAME is NOT NULL and DLIV_TOWN is NOT NULL and
DLIV_COUNTRY is NOT NULL)
UNION
SELECT DISTINCT BILL_WAYTYPE, BILL_WAYNAME, USERNAME, BILL_GATE, BILL_BLOCK, BILL_STAIRW, BILL_FLOOR,
BILL_DOOR, BILL_ZIP, BILL_TOWN, BILL_COUNTRY
FROM fsdb.trolley

WHERE BILL_WAYTYPE is NOT NULL and BILL_WAYNAME is NOT NULL and BILL_TOWN is NOT NULL and
BILL_COUNTRY is NOT NULL)
select * from aux GROUP BY DLIV_WAYTYPE, DLIV_WAYNAME, USERNAME, DLIV_GATE, DLIV_BLOCK, DLIV_STAIRW, DLIV_FLOOR,
DLIV_DOOR, DLIV_ZIP, DLIV_TOWN, DLIV_COUNTRY
  
```

	DLIV_WAYTYPE	DLIV_WAYNAME	USERNAME	DLIV_GATE	DLIV_BLOCK	DLIV_STAIRW	DLIV_FLOOR	DLIV_DOOR	DLIV_ZIP	DLIV_TOWN	DLIV_COUNTRY
1	Plaza	Hyacinths	abby	143	(null)	(null)	Fifth	D	59145	Valnieves	Cayman Islands
2	Plaza	Orange Tree	abby	92	(null)	(null)	5	5	44112	Ciudad del Robledal	Denmark
3	Road	Reservoir	abby	54	(null)	(null)	First	C	63664	Valtorres	Cook Islands
4	Street	Community	abelarda	16	(null)	(null)	3	6	73115	Sotosolana	Rwanda
5	Street	Community	abelarda	16	(null)	(null)	3	6	73115	Sotosolana	Rwanda
6	Avenue	San Jose	abencia	179	(null)	(null)	(null)	4	95091	Vallobos del Canto Rodao	Ile of Man
7	Avenue	San Jose	abencia	179	(null)	(null)	(null)	4	95091	Vallobos del Canto Rodao	Ile of Man
8	Plaza	Holy Ghost	abo	43	(null)	(null)	Fourth	D	87648	Sotosombas	Fiji
9	Road	Our Lady at Butarque	abo	97	(null)	(null)	Second	4	58781	Valcantos	Azerbaijan
10	Square	Fishermen	acindina	159	(null)	(null)	9	(null)	22493	Sotoguijarros	Philippines
11	Plaza	Bullring	acindina	199	(null)	(null)	Fourth	(null)	32652	Nava de las Tripas	Algeria
12	Road	Cloth	acindina	26	(null)	(null)	(null)	6	96016	Valvacas de las Ardillas	Lao People's Democratic Republic

Al final decidimos hacer dos tablas, una de direcciones de pedido y otra de direcciones de facturación. De esta manera podríamos diferenciar el tipo de direcciones y facilitar la inserción de datos.

4. Problemas con la inserción de compras:

Pusimos como clave primaria de compras un ID, ya que no sabíamos qué combinación de claves primarias poner porque en este sistema las compras pueden ser realizadas por clientes registrados y no registrados. Al insertar los datos, nos decía que no habíamos insertado ninguna clave primaria, porque en la antigua base de datos no había un identificador para cada compra que pudiéramos insertar. Al final hicimos que se asignará un ID para cada compra en la creación de la tabla con el código *“ID_compra NUMBER GENERATED ALWAYS as IDENTITY(START with 1 INCREMENT by 1”*

5. Problema general de otras tablas:

El fallo de violación de clave primaria ha sido el que más ha salido a la hora de las inserciones. La mayoría se solucionan incluyendo más claves primarias en la tabla o agrupando algunas tuplas con el GROUP BY, pero otras requerían una solución más compleja, como las mencionadas anteriormente.