

Proyecto Final  
Tacoste  
Universidad Nacional Autónoma de México  
Facultad de Ciencias  
Fundamentos de Bases de Datos

Ángel Iván Gladín García  
No. cuenta: 313112470  
angelgladin@ciencias.unam.mx

Luis Fernando Yang Fong Baeza  
No. cuenta: 313320679  
fernandofong@ciencias.unam.mx

Marco Antonio Hurtado Gutierrez  
No. cuenta: 313110902  
markhg@ciencias.unam.mx

María Fernanda González Chávez  
No. cuenta: 313036367  
fernandagch@ciencias.unam.mx

31 de agosto de 2017

## Modelo Relacional

Sucursal(Id.Sucursal<sub>PK</sub>, CURP\_Gerente<sub>FK</sub>, Estado, Municipio, Colonia, CP, Calle, Numero, Horario\_Ap, Horario\_Cierre, Fecha\_Inicio).

Como para una sucursal, solo tenemos un gerente, entonces tenemos que  $ID \rightarrow CURP\_G$ . También, tenemos que  $Colonia \rightarrow CP$ , no podemos tener repetidos y solo a uno, se le asigna un CP único, entonces tenemos que el conjunto de dependencias funcionales es:

$$DF = \{ID \rightarrow CURP, Col \rightarrow CP\}$$

Se crea esta tabla ya que *Telefono* es un atributo multivaluado.

TelSucursal(Id\_Sucursal<sub>FK</sub>, Telefono).

Esta relación, no puede ser otra más que la trivial, puesto que justamente para eso hicimos el atributo multivaluado, el mismo Id puede y debe ir a distintos teléfonos, entonces  $ID \text{ Tel} \rightarrow ID \text{ Tel}$ .

Empleado(CURP<sub>PK</sub>, Id\_Sucursal<sub>FK</sub>, Paterno, Materno, Nombre, Estado, Municipio, Colonia, CP, Num, RFC, Num\_Emerg, Tipo, Fech\_Ing, Tipo\_Emp, Tipo\_Sangre, Fecha\_Nac).

En esta relación, también se comparte la dependencia de la Colonia con el CP, pero podemos observar que se agrega  $CURP \rightarrow RFC$ , además de que tenemos  $CURP \rightarrow Pat, Mat, Nombre$ , entonces el conjunto de dependencias funcionales se vuelve:

$$DF = \{CURP \rightarrow RFC, CURP \rightarrow Pat \text{ Mat Nombre}, Colonia \rightarrow CP\}$$

CorreoEmpleado(CURP<sub>FK</sub>, Correo).

TelEmpleado(CURP<sub>FK</sub>, Tel).

Supervisar(CURP\_Supervisor<sub>FK</sub>, CURP\_Subordinado<sub>FK</sub>).

Estas tres relaciones son el mismo caso que para la relación TelSucursal.

Se incluye el atributo *CURP* ya que esta conectado a la entidad *Repartidor* la cual se incluye en la entidad *Empleado* y por esto se toma la llave de ésta entidad.

Licencia(Código<sub>PK</sub>, CURP<sub>FK</sub>, Tipo, Vigencia, Transporte).

Tenemos una dependencia algo obvia la que a un código de licencia, solo puede ser asignada a un CURP, o a una persona, no pueden compartir licencia dos personas, pero puede haber distintos códigos de licencia con el mismo tipo, o el mismo día de vigencia o el mismo transporte, entonces eso vuelve al conjunto de DF:

DF = {Codigo  $\rightarrow$  CURP}

Proveedor(RFC<sub>PK</sub>, Estado, Nombre, Municipio, Colonia, Calle, CP, Numero).

No podemos tener el mismo RFC para dos distintos proveedores, entonces tenemos que RFC  $\rightarrow$  Nombre y de igual manera tenemos la dependencia Colonia  $\rightarrow$  CP, entonces eso nos deja con:

DF = {RFC  $\rightarrow$  Nombre, Colonia  $\rightarrow$  CP}

Suministro(Id\_Suministro<sub>PK</sub>, RFC<sub>FK</sub>, Nombre, Marca, Precio).

El Id\_Suministro determina funcionalmente al nombre, no podemos tener distintos Id's para distintos nombres, podemos tener distintas marcas, pero es el mismo nombre, entonces Id  $\rightarrow$  Nombre, pero de distintos proveedores puede venir el mismo producto, de distinta marca o con precio variado entonces solo queda así:

DF = {Id  $\rightarrow$  Nombre}

Suministrar(No\_Lote<sub>FK</sub>, Id\_Suministro<sub>FK</sub>).

Observemos que una vez más vemos el caso de una relación en la cuál, no podemos obtener una dependencia funcional, puesto que en el mismo lote, pueden viajar muchos suministros, entonces no podemos obtener DF.

Inventario(No\_Lote<sub>PK</sub>, Dia, Cantidad).

En esta relación, el número de lote no nos asegura algo sobre el día, puede haber dos lotes que se hayan entregado el mismo día o con la misma cantidad, entonces no hay dependencias funcionales.

Generar(Id\_Sucursal<sub>FK</sub>, No\_Lote<sub>FK</sub>).

Mismo caso de una relación bivaluada.

Comensal(Id.Comensal<sub>PK</sub>, Pat, Mat, Nombre, Estado, Municipio, Colonia, CP, Num, Correo, Telefono,Puntos).

Tenemos que el Id del comensal, por obvias razones nos va a determinar su nombre y su correo, puesto que tampoco puede haber correos repetidos en cualquier dominio entonces  $\text{Id} \rightarrow \text{Pat Mat Nombre Correo}$ , pero puede haber dos clientes que habiten en la misma casa, entonces no hay dependencia ahí, más que una vez más,  $\text{Colonia} \rightarrow \text{CP}$ , entonces el conjunto final queda:

$$\text{DF} = \{\text{ID} \rightarrow \text{Pat Mat Nombre Correo}, \text{Colonia} \rightarrow \text{CP}\}$$

Pedido(No\_Pedido<sub>PK</sub>, Id\_Cliente<sub>FK</sub>, Fecha, Forma\_Pago).

Observemos que distintos pedidos pueden coincidir tanto en cliente, como en fecha o en forma de pago, y no hay una forma exacta de encontrar una dependencia funcional para esta relación, puesto que si probamos con las combinaciones entre atributos, ninguna tiene mucho sentido, por lo limitado que tenemos los dominios de los atributos y lo grandes que pueden ser otros campos, entonces esta relación no tiene Dependencias Funcionales.

Mesa(Id.Mesa<sub>PK</sub>,Id.Cliente<sub>FK</sub>,Id.PayMaster).

Se crea la tabla Pertenecer ya que *Id.Comensal* es multivaluado en la relación *Pertenecer*.

En esta relación, es aún más fácil de observar que muchos clientes pueden estar en la misma mesa y que pudo haber pagado el mismo cliente para esa mesa, entonces rompe con cada parte de las dependencias funcionales, entonces no es posible tampoco tener DF.

Contener(No\_Pedido<sub>FK</sub>, Id\_Producto<sub>FK</sub>,Cantidad).

En esta relación tenemos que en un solo pedido y un producto puede venir en un única cantidad, entonces tenemos como dependencia funcional  $\text{No Id} \rightarrow \text{Cantidad}$ , no podemos tener en un mismo pedido y un producto en específico, dos cantidades distintas. Al final el conjunto de DF queda:

$$\text{DF} = \{\text{No Id} \rightarrow \text{Cantidad}\}$$

Pertenecer(Id.Mesa<sub>FK</sub>,Id.Comensal<sub>FK</sub>).

Hemos tenido varios casos de relaciones de dos atributos, entonces procedemos de la misma manera, no hay otra dependencia que la trivial.

Producto(Id.Producto<sub>PK</sub>, Nombre, Categoria, Precio).

En esta relación tenemos que un id, justamente nos identifica una el nombre de un producto, entonces  $\text{Id} \rightarrow \text{Nombre}$ , aunque no su categoría, puesto que muchos productos pueden ir a la misma categoría, además, no podemos tener el mismo producto con distinto precio, entonces también tenemos  $\text{Nombre} \rightarrow \text{Precio}$ .

Participar(Id\_Producto<sub>FK</sub>, Nombre\_Ingrediente<sub>FK</sub>).

Otra vez, una relación de dos atributos, simplemente procedemos.

Ingrediente(Nombre<sub>PK</sub>).

Aquí, no puede haber otra dependencia que no sea la trivial, entonces de igual manera procedemos.

Consumir(Id\_Comensal<sub>FK</sub>, Id\_Pedido, Ticket).

A un comensal con un pedido, corresponde únicamente con un ticket, entonces eso nos da una dependencia funcional, Id\_Com Id\_Ped  $\rightarrow$  Ticket, entonces:

DF = {Id\_Com, Id\_Ped  $\rightarrow$  Ticket}

Se crea esta tabla ya que *Etiqueta* es un atributo multivaluado de la entidad *Producto*.

EtiquetaProd(Id\_Producto<sub>FK</sub>, Etiqueta).

Salsa(Id\_Producto<sub>PK</sub>, Nombre, Presentacion, Nivel\_Picor).

En este punto tenemos que el Id, nos determina funcionalmente al nombre de la salsa, puesto que no tiene sentido identificar la misma salsa con un Id distinto, entonces Id  $\rightarrow$  Nombre, pero también observemos que sigue siendo la misma salsa, no importa en qué presentación venga, sigue siendo lo mismo, entonces Nombre  $\rightarrow$  Nivel\_Picor, dejando:

DF = {Id  $\rightarrow$  Nombre, Nombre  $\rightarrow$  Nivel\_Picor}

Se crea una nueva tabla a la que llamaremos *Emparejar* ya que el atributo *Id\_Producto* en la tabla *Salsa* es multivaluado.

Emparejar(Id\_Salsa<sub>FK</sub>, Id\_Producto<sub>FK</sub>).

Por último, tenemos la relación repartidores, en la que vamos a almacenar todos los empleados que son repartidores y como son repartidores, tienen asociada una única licencia, puede que tengamos nulos, ya que habrá repartidores que vayan en bicicleta.

Repartidor(CURP\_Empleado<sub>FK</sub>, Código\_Licencia(\_FK)).