



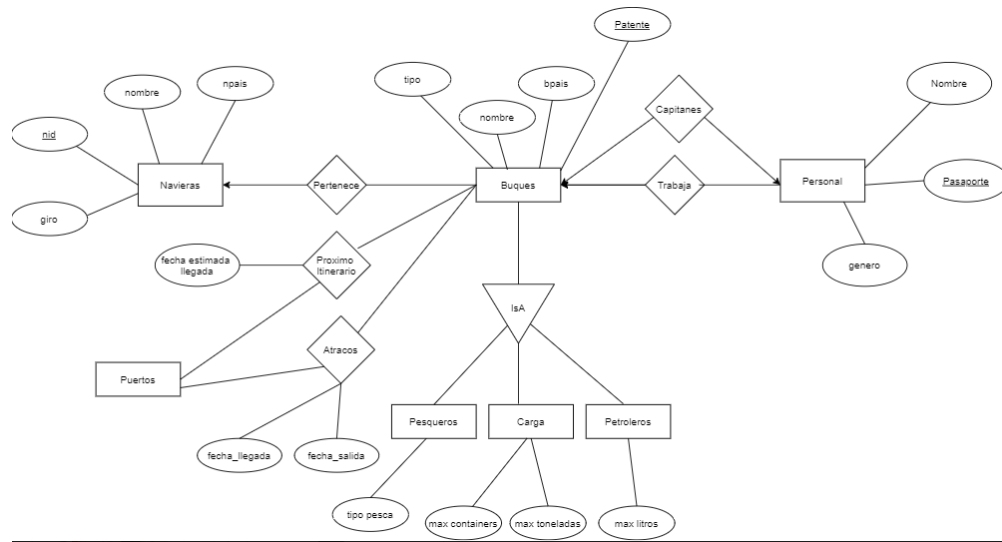
Entrega 2

28 - 09 - 2020

2 - 2020 - Juan Reutter

Gianluca Falcone, Diego Chahuan

1. Diagrama Entidad/Relacion



2. Esquemas

- Navieras(nid Primary Key int, nombre varchar(255), npais varchar(255), giro varchar(255))
- Buques(patente varchar(255) Primary key , nombre varchar(255), tipo varchar(255), bpais varchar(255))
- Pesqueros(patente varchar(255) Primary key , tipo_pesca varchar(255), foreign key patente)
- Petroleros(patente varchar(255) Primary key , max_lit int, foreign key patente)
- Carga(patente varchar(255) Primary Key, max_container int, max_ton float, foreign key patente)
- Pertenece(patente varchar(255) Primary Key, nid int, Foreign Keys (nid, patente))
- Personal(pasaporte varchar(255) Primary Key , nombre varchar(255), genero varchar(255), edad int, nacionalidad varchar(255))
- Trabaja_en (pasaporte varchar(255) primary key, patente varchar(255), Foreign Keys (patente, pasaporte))
- Capitanes(pasaporte varchar(255), patente varchar(255),Foreign Keys (patente, pasaporte))

- Atracos(puerto varchar(255),patente varchar(255), fecha_salida timestamp, fecha_llegada timestamp, Primary Key(Patente,fecha_llegada))
- Proximo_itinerario(puerto varchar(255),patente varchar(255), fecha_llegada timestamp, Primary Key(patente,fecha_llegada))

3. Justificacion 3NF

Este esquema pertenece a BCNF, para demostrar esto vamos a ir analizando las dependencias funcionales en todas las relaciones de nuestro esquema. En la primera relación Navieras, no se nos especifica qué nombre ni giro ni país son únicos, tampoco que un país solo pueda tener una naviera con un nombre por lo que se tiene que crear un id que determina todos los otros atributos por lo que la única dependencia funcional que tenemos es $nid \rightarrow nombre, npais, giro$ y como nid es parte de la llave esta relación pertenece a BCNF. Un argumento similar se puede hacer para el personal y buques, ambos tienen solo una dependencia funcional, para buques esta es $patente \rightarrow nombre, tipo, bpais$ y para personal es $pasaporte \rightarrow nombre, género$. Pero también tenemos que hay una dependencia funcional que relaciona estas dos tablas que es que el pasaporte \rightarrow patente. Esto se debe a que si uno sabe el trabajador esté solo puede trabajar en un solo buque por lo que creamos una relación Capitan y Trabaja.en para poder tener una tabla que contenga esta dependencia y cómo estas relaciones tienen una llave foránea que contiene el atributo pasaporte está también pertenece a BCNF. De la misma manera dentro del esquema también hay una dependencia funcional que es $patente \rightarrow nid$ ya que cada buque puede pertenecer a solo una naviera para representar esta dependencia en nuestro esquema usamos la relación pertenece que tiene como atributos las llaves foráneas la patente y el nid y como primary key patente que es la parte izquierda de la dependencia funcional por lo que esta relación también pertenece a BCNF. También tenemos que el buque determina las características específicas de su tipo con esta información podemos deducir que $patente \rightarrow tipo_pesca, patente \rightarrow max_lit, patente \rightarrow max_container, max_ton$. Y al igual que las anteriores patente forma parte de las llaves de estas tablas pero como varios buques pueden tener las mismas características no se forman dependencias funcionales dentro de estas relaciones por lo que estas pertenecen a BCNF. Por último tenemos que justificar que las tablas de atracos y próximo itinerario están normalizadas. Estas relaciones tienen un poco más de complejidad que las anteriores ya que ni un atributo por sí solo determina el otro. Un buque puede haber llegado y salido muchas veces del mismo puerto, un puerto recibe muchos buques en distintas fechas. En las mismas fechas pueden llegar y salir distintos buques. Por estas razones tenemos que buscar cual es la dependencia funcional si reflexionamos podemos darnos cuenta que un buque solo puede llegar a un puerto a la vez por lo que si tenemos el buque y la fecha de llegada podemos determinar el puerto y la fecha de salida. Por lo que para la relación atraco tenemos que la única dependencia funcional que se tiene es $patente, fecha_llegada \rightarrow fecha_salida, puerto$. En todo caso en esta relación también se forma una dependencia funcional buque, $fecha_salida \rightarrow puerto$, como un buque solo puede estar en un puerto a la vez la fecha de salida puede determinar el puerto pero cómo generamos mucha repetición si generamos otra nueva relación con estos 3 atributos y si los separamos de la tabla perderíamos una dependencia funcional ya no vamos a poder justificar nuestro esquema como BCNF sino como 3NF esto se debe a que $fecha_salida$ no forma parte de la llave lo que rompe el BCNF pero depende de un atributo que sí pertenece a una llave por lo que está dentro de 3NF. Como próximo itinerario no tiene $fecha_salida$ se aplica la misma lógica para la primera dependencia funcional de la otra tabla que es $patente, fecha_entrada \rightarrow puerto$ y como patente y fecha de entrada son las llaves esta no rompe BCNF. En conclusión aunque la mayoría de las relaciones cumplen con BCNF no podemos justificar nuestro esquema en esta forma normal ya que la tabla de atracos no cumple las restricciones pero sí podemos decir que nuestro esquema se encuentra en 3NF.

4. Consultas SQL

1. `select nombre from navieras;`

2. `select distinct buques.nombre, buques.patente, buques.tipo, buques.bpais from buques, pertenece, navieras where pertenece.patente = buques.patente and pertenece.nid in (select nid from navieras where lower(nombre) like 'francis drake s.a.');`
3. `select buques.nombre, buques.patente, buques.tipo, buques.bpais from buques where patente in (select patente from atracos where lower(puerto) like '%valparaiso%' and fecha_llegada ≥ '2020-1-1' and fecha_llegada ≤ '2020-12-31');`
4. `select * from buques where patente in (select atracos.patente from (select * from atracos where patente in (select patente from buques where lower(nombre) = 'magnolia') and lower(puerto) = 'mejillones') as foo join atracos on ((atracos.fecha_llegada ≥ foo.fecha_llegada and atracos.fecha_llegada ≤ foo.fecha_salida) or (atracos.fecha_salida ≥ foo.fecha_llegada and atracos.fecha_salida ≤ foo.fecha_salida)) and lower(atracos.puerto)=lower(foo.puertos));`
5. `select * from personal where pasaporte in (select distinct pasaporte from buques, capitanes where buques.patente in (select patente from atracos where lower(puerto) like '%talcahuano%') and capitanes.patente = buques.patente) and genero like 'mujer';`
6. `select * from buques where patente in (select patente from (select * from buques natural join trabaja_en where tipo = 'pesquero' union select * from buques natural join capitanes where tipo = 'pesquero') as foo group by patente having count(*) ≥ ALL (select count(*) from (select * from buques natural join trabaja_en union select * from buques natural join capitanes) as foo group by patente));`

5. Consideraciones Especiales

- Las relacion personal no tiene ni una tupla con nombres con apostrofe, apesar de que la base de datos si los tenia lo que puede dar algunas consultas distintas.