



DEPARTAMENTO  
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

# Trabajo Práctico I

---

23 de septiembre de 2015

Bases de datos  
Segundo Cuatrimestre de 2015

Integrante	LU	Correo electrónico
Ignacio Truffat	837/10	el_truffa@hotmail.com
Gaston Rocca	836/97	gastonrocca@gmail.com
Agustín Godnic	689/10	agustingodnic@gmail.com
Matías Pizzagalli	257/12	matipizza@gmail.com



**Facultad de Ciencias Exactas y Naturales**

Universidad de Buenos Aires

Ciudad Universitaria - (Pabellón I/Planta Baja)

Intendente Güiraldes 2160 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (54 11) 4576-3359

<http://www.fcen.uba.ar>

# Índice

<b>1. Introducción.</b>	<b>3</b>
<b>2. Diagrama Entidad Relacion.</b>	<b>4</b>
2.1. Restricciones adicionales. . . . .	5
2.2. Asumpciones sobre el dominio del problema. . . . .	5
<b>3. Modelo Relacional</b>	<b>6</b>
3.1. Modelado general del siniestro . . . . .	6
3.2. Modelado de las personas involucradas en el siniestro . . . . .	6
3.3. Modelado de un conductor . . . . .	7
3.4. Modelado de un vehículo . . . . .	7
3.5. Lugar de ocurrencia del siniestro . . . . .	8
3.6. Análisis, estudios, peritajes . . . . .	9
3.7. Condiciones de clima, pavimento, iluminación, etc . . . . .	9
<b>4. Diseño Físico</b>	<b>10</b>
4.1. Consulta por número de licencia. . . . .	10
4.1.1. version con sub-query . . . . .	10
4.1.2. version con Vista. . . . .	10
4.1.3. version con Trigger . . . . .	11
4.2. Consulta por modalidad de accidentes viales. . . . .	11
<b>5. Conclusiones</b>	<b>13</b>

## 1. Introducción.

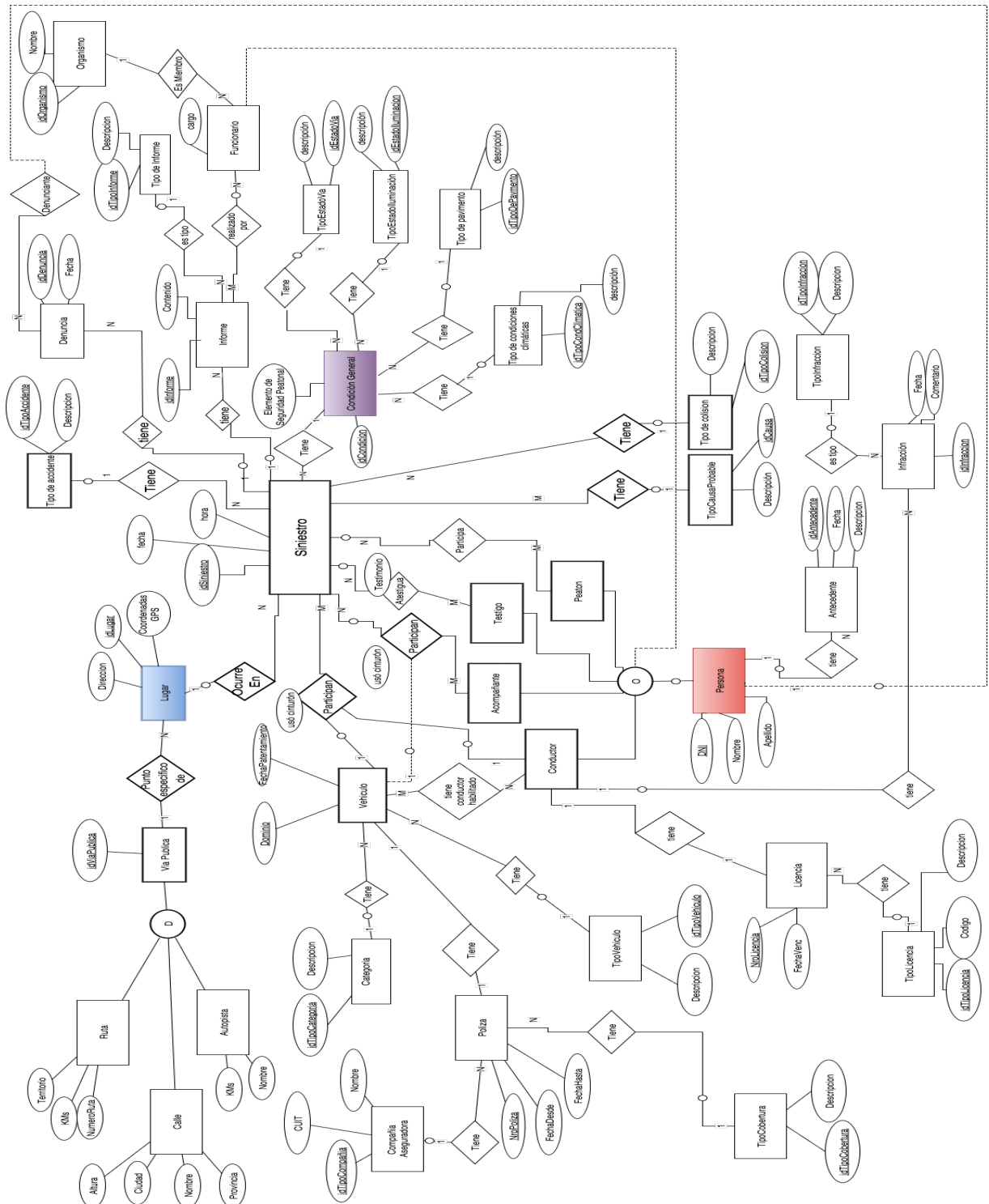
En el presente trabajo práctico nos proponemos modelar un registro de accidentes viales a nivel nacional, apartir de este modelo podremos comprender la situación actual, tener mas control sobre lo que sucede y generar posibles politicas o medidas de seguridad de vial. Todo con el fin de poder disminuir la cantidad elevada de accidentes de transito que tenemos hoy en dia, lo cual es una de las preocupaciones del gobierno.

De esta forma proponemos la creación de un Registro Único de Accidentes de Tránsito (RUAT), diseñado como presentamos a medida que desarrollamos el trabajo práctico. El RUAT almacenará toda la información que resulte relevante a nuestro problema sobre los siniestros ocurridos, como por ejemplo detalles sobre el lugar donde sucedió, la fecha y hora del suceso, condiciones generales del lugar del suceso, personas y vehiculos involucrados en el suceso, informes y peritajes correspondientes, causas probables del siniestro, entre otras.

Por otro lado, el RUAT tambien tendrá información sobre las autopistas y rutas a nivel nacional, información sobre el parque automotor que se encuentra en el país, información sobre quienes y como regulan las polizas de seguro sobre el parque automotor e información sobre los conductores habilitados para conducir y si tienen antecedentes penales.

De esta forma el RUAT presenta, desde nuestro punto de vista, información suficiente para que se pueda generar un plan de seguridad eficaz.

## 2. Diagrama Entidad Relacion.



## 2.1. Restricciones adicionales.

Las restricciones adicionales que escapan el Diagrama de Entidad Relacion son:

- No puede haber dos instancias de la entidad Autopista con mismo nombre.
- No puede haber dos instancias de la entidad Ruta con mismo nombre y territorio.
- No puede haber dos instancias de la entidad Calle con mismo nombre, ciudad y provincia.
- Dado un siniestro, una persona solo puede relacionarse con este siniestro mediante un único rol (Conductor, Acompañante, Testigo, Peaton, Funcionario).
- Dado un siniestro, el conjunto de Vehiculos que se relacionan con este siniestro y con la entidad acompañante esta incluido en el conjunto de Vehiculos que se relacionan con este mismo siniestro y con la entidad Conductor.
- Dado un Informe, todos los funcionarios que lo realizaron son miembros de la misma organización.

## 2.2. Asumpciones sobre el dominio del problema.

A la hora de plantear nuestra solución del problema, tomamos como verdaderas ciertas supociones, las cuales son:

- Todo Conductor tiene Licencia de conducir.
- Todo Vehiculo tiene al menos un conductor habilitado.
- Todo Conductor tiene al menos un Vehiculo para el cual esta habilitado.
- Todo Vehiculo tiene una Poliza de seguro.
- Si un funcionario se vio involucrado en un siniestro entonces este no puede realizar ningun tipo de informe sobre el siniestro.
- Cuando se habla de Tipo de Lugar o Tipo de Camino, lo tomamos como equivalente y lo modelamos en la entidad Via Publica.
- La entidad Tipo de Informe representa analisis, peritaje y estudio.
- No permitimos el trabajo intra-organismos, es decir los informes son realizados completamente por el mismo organismo.

### 3. Modelo Relacional

Por cuestiones de legibilidad, se ha dividido el modelo relacional en sectores de acuerdo al comportamiento que modela cada uno.

#### 3.1. Modelado general del siniestro

**Siniestro**( idSiniestro, fecha, hora, idLugar, idTipoAccidente, idDenuncia, idInforme, idCondicion, idTipoColision, idCausa )

**PK** = **CK** = {idSiniestro}

**FK** = {idLugar, idTipoAccidente, idDenuncia, idInforme, idCondicion, idTipoColision, idCausa}

**TipoCausaProbable**( idCausa, descripcion )

**PK** = **CK** = {idCausa}

**TipoDeColision**( idTipoColision, descripcion )

**PK** = **CK** = {idTipoColision}

**TipoDeAccidente**( idTipoAccidente, descripcion )

**PK** = **CK** = {idTipoAccidente}

**SiniestroVehiculoConductor**( idSiniestro, dominio, DNI, usoCintiron )

**PK** = **CK** = {(idSiniestro, DNI, dominio)}

**FK** = {idSiniestro, DNI, dominio}

#### 3.2. Modelado de las personas involucradas en el siniestro

**Siniestro**( idSiniestro, fecha, hora, idLugar, idTipoAccidente, idDenuncia, idInforme, idCondicion, idTipoColision, idCausa )

**PK** = **CK** = {idSiniestro}

**FK** = {idLugar, idTipoAccidente, idDenuncia, idInforme, idCondicion, idTipoColision, idCausa}

**Testigo**( DNI )

**PK** = **CK** = {DNI}

**FK** = {DNI}

*El caso es análogo para las entidades Peaton y Acompañante.*

**SiniestroPeaton**( DNI, idSiniestro )

**PK** = **CK** = {(DNI, idSiniestro)}

**FK** = {DNI, idSiniestro}

*El caso es análogo para la relacion Atestigua (SiniestroTestigo).*

**SiniestroAcompañante**( DNI, idSiniestro, usoCinturon )  
**PK = CK** = {(DNI, idSiniestro)}  
**FK** = {DNI, idSiniestro}

**Persona**( DNI, nombre, apellido )  
**PK = CK** = {DNI}

**Antecedente**( idAntecedente, fecha, descripcion, DNI )  
**PK = CK** = {idAntecedente}  
**FK** = {DNI}

**Denuncia**( idDenuncia, fecha, DNI, idSiniestro )  
**PK = CK** = {idAntecedente}  
**FK** = {DNI, idSiniestro}

### 3.3. Modelado de un conductor

**Conductor**( DNI )  
**PK = CK** = {DNI}  
**FK** = {DNI}

**Licencia**( nroLicencia, fechaVencimiento, DNI, idTipoLicencia )  
**PK = CK** = {nroLicencia}  
**FK** = {DNI, idTipoLicencia}

**TipoLicencia**( idTipoLicencia, codigo, descripcion )  
**PK = CK** = {idTipoLicencia}

**Infraccion**( idInfraccion, fecha, DNI, idTipoInfraccion )  
**PK = CK** = {idInfraccion}  
**FK** = {DNI, idTipoInfraccion}

**TipoInfraccion**( idTipoInfraccion, descripcion )  
**PK = CK** = {idTipoInfraccion}

### 3.4. Modelado de un vehículo

**Vehiculo**( dominio, fechaPatentamiento, idTipoCategoria, idTipoVehiculo )  
**PK = CK** = {dominio}

**FK** = {idTipoCategoria, idTipoVehiculo}

**Categoria**( idTipoCategoria, descripcion )  
**PK** = **CK** = {idTipoCategoria}

**TipoVehiculo**( idTipoVehiculo, descripcion )  
**PK** = **CK** = {idTipoVehiculo}

**Poliza**( nroPoliza, fechaDesde, fechaHasta, dominio, idTipoCobertura, idTipoCompania )  
**PK** = **CK** = {nroPoliza}  
**FK** = {dominio, idTipoCobertura, idTipoCompania}

**TipoCobertura**( idTipoCobertura, descripcion )  
**PK** = **CK** = {idTipoCobertura}

**TipoCompania**( idTipoCompania, CUIT, nombre )  
**PK** = **CK** = {idTipoCompania}

**TieneConductorHabilitado**( DNI, dominio )  
**PK** = **CK** = {(DNI, dominio)}  
**FK** = {DNI, dominio}

### 3.5. Lugar de ocurrencia del siniestro

**Lugar**( idLugar, coordenadasGPS, direccion, idViaPublica )  
**PK** = **CK** = {idLugar}  
**FK** = {idViaPublica}

**ViaPublica**( idViaPublica, coordenadasGPS, )  
**PK** = **CK** = {idViaPublica}

**Ruta**( idLugar, KMs, territorio, numeroRuta )  
**PK** = **CK** = **FK** = {idLugar}

**Autopista**( idLugar, nombre, KMs )  
**PK** = **CK** = **FK** = {idLugar}

**Calle**( idLugar, provincia, ciudad, nombre, altura )  
**PK** = **CK** = **FK** = {idLugar}



### 3.6. Análisis, estudios, peritajes

**Informe**( idInforme, contenido, idTipoInforme )  
**PK = CK** = {idInforme}  
**FK** = {idTipoInforme}

**Funcionario**( DNI, cargo, idOrganismo )  
**PK = CK** = {DNI}  
**FK** = {DNI, idOrganismo}

**FueRealizadoPor**( DNI, idInforme )  
**PK = CK** = {(DNI, idInforme)}  
**FK** = {DNI, idInforme}

**Organismo**( idOrganismo, nombre )  
**PK = CK** = {idOrganismo}

**TipoInforme**( idTipoInforme, descripcion )  
**PK = CK** = {idTipoInforme}

### 3.7. Condiciones de clima, pavimento, iluminación, etc

**CondicionGeneral**( idCondicion, elementoSeguridadPeatonal, idTipoCondClimatica,  
idTipoDePavimento, idEstadoIluminacion, idEstadoVia )  
**PK = CK** = {idCondicion}  
**FK** = {idTipoCondClimatica, idTipoDePavimento, idEstadoIluminacion, idEstadoVia}

**TipoCondClimatica**( idTipoCondClimatica, descripcion )  
**PK = CK** = {idTipoCondClimatica}

**TipoDePavimento**( idTipoDePavimento, descripcion )  
**PK = CK** = {idTipoDePavimento}

**TipoEstadoIluminacion**( idEstadoIluminacion, descripcion )  
**PK = CK** = {idEstadoIluminacion}

**TipoEstadoVia**( idEstadoVia, descripcion )  
**PK = CK** = {idEstadoVia}

## 4. Diseño Físico

El diseño físico de nuestra solución lo implementamos sobre el motor de base de datos MySQL. Constó de varias etapas, primero un modelado con la herramienta MySQLWorkbench, y luego desarrollamos scripts para la creación de tablas y para poblar las tablas con datos (situados en los directorios bd. Para crear las tablas se puede correr `mysql < create.sql`, y para llenarlas con datos, correr `mysql < llenar.sql`. Esos datos fueron necesarios para probar las queries desarrolladas.

### 4.1. Consulta por número de licencia.

#### 4.1.1. version con sub-query

```
USE 'tp1';
DROP procedure IF EXISTS 'HistorialLicencia';

DELIMITER $$
USE 'tp1' $$
/*Consulta por número de licencia: obtener, con un número de licencia específico,
información sobre los accidentes en los que ha participado el conductor propietario de
la misma, con detalles de fecha, lugar, tipo de accidente, participación y modalidad.
También se deberá indicar la cantidad de automóviles que está habilitado a conducir*/
CREATE PROCEDURE 'HistorialLicencia' (nrolicencia int)
BEGIN
    select S.Fecha, Lugar.Direccion, TA.Descripcion as TipoAccidente, Participacion.Participacion
    from licencia L
    inner join Conductor C on L.nroLicencia = C.nroLicencia
    inner join Persona P on C.dni = P.dni
    /*Filtrar por Dni*/
    inner join
        (
            select Dni, idSiniestro, 'Acompañante' as Participacion
            from siniestroacompañante

            union all
            select dni, idSiniestro, 'Testigo' as Participacion
            from siniestroTestigo

            union all
            select dni, idSiniestro, 'Peaton' as Participacion
            from siniestroPeaton

            union all
            select dni, idSiniestro, 'Conductor' as Participacion
            from siniestroVehiculoConductor
        ) as Participacion
    on Participacion.dni = P.dni
    inner join Siniestro S on S.idSiniestro = Participacion.idSiniestro
    inner join Lugar on Lugar.idLugar = S.idLugar
    inner join TipoAccidente TA on TA.idTipoAccidente = S.idTipoAccidente
    where L.nroLicencia = nrolicencia;

    /*También se deberá indicar la cantidad de automóviles que está habilitado a conducir*/
    /*Detalle de los automoviles Habilitados*/
    select V.Dominio, V.FechaPatentamiento, C.Descripcion as Categoria, TV.Descripcion as TipoVehiculo
    from Licencia L
    inner join Conductor on L.nroLicencia = Conductor.nroLicencia
    inner join conductorhabilitado CH on CH.DNI = Conductor.dni
    inner join vehiculo V on V.idVehiculo = CH.idVehiculo
    inner join Categoria C on C.idCategoria = V.IdCategoria
    inner join TipoVehiculo TV on TV.idTipoVehiculo = V.idTipoVehiculo
    where L.nroLicencia = nrolicencia;
END
$$

DELIMITER ;
```

#### 4.1.2. version con Vista.

```
CREATE DEFINER='root'@'localhost' PROCEDURE 'HistorialLicenciaConVista'(nrolicencia int)
BEGIN
```

```

select S.Fecha, Lugar.Direccion, TA.Descripcion as TipoAccidente, Participantes.Participacion
from licencia L
inner join Conductor C on L.nroLicencia = C.nroLicencia
inner join Persona P on C.dni = P.dni
/* Filtrar por Dni */
inner join Participantes on Participantes.dni = P.dni
inner join Sinistro S on S.idSinistro = Participantes.idSinistro
inner join Lugar on Lugar.idLugar = S.idLugar
inner join TipoAccidente TA on TA.idTipoAccidente = S.idTipoAccidente
where L.nroLicencia = nrolicencia;

/* También se deberá indicar la cantidad de automóviles que está habilitado a conducir */
/* Detalle de los automoviles Habilitados */
select V.Dominio, V.FechaPatentamiento, C.Descripcion as Categoria, TV.Descripcion as TipoVehiculo
from Licencia L
inner join Conductor on L.nroLicencia = Conductor.nroLicencia
inner join conductorhabilitado CH on CH.DNI = Conductor.dni
inner join vehiculo V on V.idVehiculo = CH.idVehiculo
inner join Categoria C on C.idCategoria = V.IdCategoria
inner join TipoVehiculo TV on TV.idTipoVehiculo = V.idTipoVehiculo
where L.nroLicencia = nrolicencia;
END

```

### 4.1.3. version con Trigger

Los triggers van cargando la tabla participa, hay un trigger por cada tipo de participante.

```

CREATE DEFINER='root'@'localhost' PROCEDURE `HistorialLicenciaConTriggers` (nrolicencia int )
BEGIN

select S.Fecha, Lugar.Direccion, TA.Descripcion as TipoAccidente, Participante.TipoParticipacion as Participacion
from licencia L
inner join Conductor C on L.nroLicencia = C.nroLicencia
inner join Persona P on C.dni = P.dni
/* Filtrar por Dni */
inner join Participante on Participante.dni = P.dni
inner join Sinistro S on S.idSinistro = Participante.idSinistro
inner join Lugar on Lugar.idLugar = S.idLugar
inner join TipoAccidente TA on TA.idTipoAccidente = S.idTipoAccidente
where L.nroLicencia = nrolicencia;

/* También se deberá indicar la cantidad de automóviles que está habilitado a conducir */
SELECT
    V.Dominio,
    V.FechaPatentamiento,
    C.Descripcion AS Categoria,
    TV.Descripcion AS TipoVehiculo
FROM
    Licencia L
    INNER JOIN
    Conductor ON L.nroLicencia = Conductor.nroLicencia
    INNER JOIN
    conductorhabilitado CH ON CH.DNI = Conductor.dni
    INNER JOIN
    vehiculo V ON V.idVehiculo = CH.idVehiculo
    INNER JOIN
    Categoria C ON C.idCategoria = V.IdCategoria
    INNER JOIN
    TipoVehiculo TV ON TV.idTipoVehiculo = V.idTipoVehiculo
WHERE
    L.nroLicencia = nrolicencia;
END

```

## 4.2. Consulta por modalidad de accidentes viales.

```

/*
Consulta por modalidad de accidentes viales: Obtener, dado una modalidad de
accidente (atropello, vuelco, incendio, etc), un listado de licencias de
conducir y la cantidad de veces que cada una de estas licencias incurrio en
la modalidad consultada.
*/

USE 'tp1';
DROP procedure IF EXISTS `DetalleDeAccidentesPorModalidad`;

DELIMITER $$
USE 'tp1'$$
CREATE DEFINER='root'@'localhost' PROCEDURE `DetalleDeAccidentesPorModalidad` (modalidad varchar(200))
BEGIN

```

```
select Licencia.NroLicencia, count(Siniestro.idSiniestro)
from TipoColision
inner join Siniestro on Siniestro.idTipoColision = TipoColision.idTipoColision
inner join siniestrovehiculoconductor SVC on SVC.idSiniestro = Siniestro.idSiniestro
inner join Conductor on Conductor.DNI = SVC.DNI
inner join Licencia on Conductor.nroLicencia = Licencia.nroLicencia
where TipoColision.Descripcion = modalidad
group by Licencia.NroLicencia;

END$$

DELIMITER ;
```

## 5. Conclusiones

Nos encontramos con varias soluciones para resolver el problema. Fueron muy valiosas las discusiones de ideas sobre diseños en el DER. También nos percatamos de la complejidad escondida que tienen los diseños, y como algo que parece simple puede ser muy complejo. Es muy importante invertir tiempo en el diseño para simplificar las consultas y en general el modelo en la base de datos. Tanto fue así, que con el modelo propuesto en el DER logramos soluciones a las consultas bastantes simples, incluso sin necesidad de Triggers.

Para la resolución de las consultas surgieron distintas opciones. La consulta HistorialLicencia se realizó inicialmente con una subquery. Luego transformamos esta subquery en una Vista. Para finalizar realizamos la consulta generando una nueva tabla de Participantes en un siniestro. Esto generó un profundo aprendizaje en el uso de los Triggers.

En la generación de los datos tuvimos algunos inconvenientes para cargar tablas con muchas FKs. Muchas de las cuales fueron creadas manualmente y con poca cantidad de registros. También nos hubiera gustado generar casos más complejos, pero por la cantidad de dependencias se hacía difícil.