



**UTN.BA**

UNIVERSIDAD TECNOLÓGICA NACIONAL  
FACULTAD REGIONAL BUENOS AIRES

Gestión de Datos

Trabajo Práctico

2º Cuatrimestre 2021

Gestión de Flota

Grupo Nro.: 17

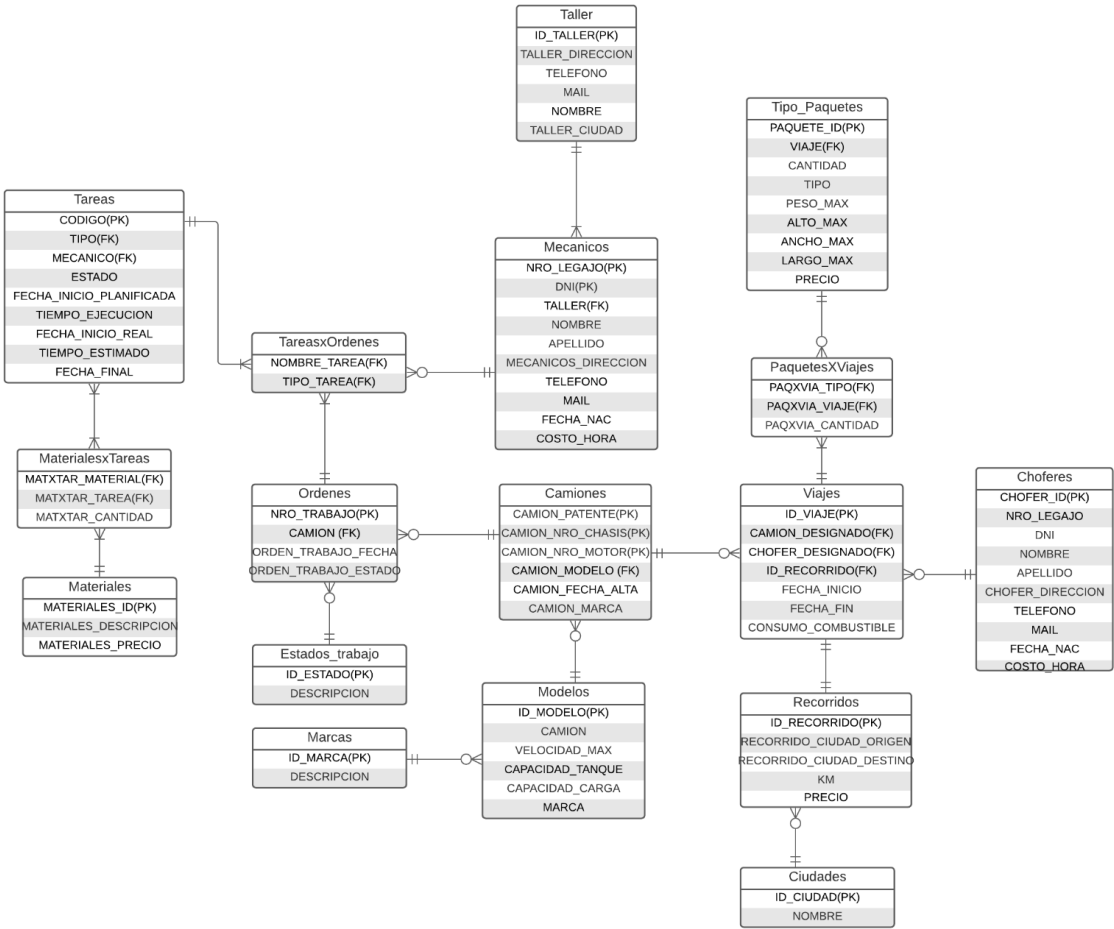
Curso: K3522

<b>Alumnos</b>	<b>Legajos</b>
Tomás Villa	149481-8
Alejo Contestabile	160549-5
Lautaro Lazo	172258-0
Santiago Fernández	173222-5

## **Introducción**

En este documento se expondrá la estrategia utilizada para desarrollar un sistema que permita la gestión de viajes y mantenimiento de camiones de una empresa de logística que se encarga de transportar paquetes entre distintas ciudades del país. Siguiendo con el diagrama de entidad relación que nos permita crear un modelo de datos que organice y normalice los datos de la Tabla Maestra para una óptima migración de estos e implementación de los nuevos requerimientos a incorporar.

**DER**



A continuación se procede a explicar detalles y nombrar tantos los atributos como las Primary Key y Foreign Key de cada entidad

## **Viajes**

Luego de la lectura del enunciado del Trabajo Práctico, optamos por comenzar con una de las entidades más importantes e influyentes del modelo, la entidad de **Viajes**. Los atributos que la acompañan son: ID\_VIAJE, CAMION\_DESIGNADO, CHOFER\_DESGINADO, ID\_RECORRIDO, FECHA\_INICIO, FECHA\_FIN, CONSUMO\_COMBUSTIBLE. Los atributos escogidos como Primary Key son: ID\_VIAJE, CAMION\_DESIGNADO, CHOFER\_DESGINADO, ID\_RECORRIDO

A partir de aquí comenzaremos a desarrollar las cuatro entidades que se relacionaban directamente con “Viajes”.

## **Recorrido**

Comenzamos a desarrollar la entidad *Recorrido* la cual cuenta con los atributos: ID\_RECORRIDO, RECORRIDO\_CIUADAD\_ORIGEN, RECORRIDO\_CIUADAD\_DESTINO, KM, PRECIO. La Primary Key de esta entidad es: ID\_RECORRIDO.

## **Camiones**

En la seccion de camiones nos encontramos con: CAMION\_PATENTE, CAMION\_NRO\_CHASIS, CAMION\_NRO\_MOTOR, CAMION\_MODELO, CAMION\_FECHA\_ALTA, CAMION\_MARCA. Las Primary Key escogidas en este caso son CAMION\_PATENTE, CAMION\_NRO\_CHASIS, CAMION\_NRO\_MOTOR. Y por otro lado, la Foreing Key es CAMION\_MODELO.

## **Chofer**

“Chofer” se compone de: CHOFER\_ID, NRO\_LEGajo, DNI, NOMBRE, APELLIDO, CHOFER\_DIRECCION, TELEFONO, MAIL, FECHA\_NAC, COSTO\_HORA. La Primary Key que corresponde a esta entidad es: CHOFER\_ID

## **Modelo**

Luego de chofer decidimos crear la entidad modelo, la misma nos permite normalizar el modelo, en ella modelo podemos encontrar los siguientes atributos: ID\_MODELO, CAMION,

VELOCIDAD\_MAX, CAPACIDAD\_TANQUE, CAPACIDAD\_CARGA, MARCA. Como Primary Key tomamos ID\_MODELO.

## **Tipo de Paquetes**

Los atributos que componen a esta entidad son: PAQUETE\_ID, VIAJE, CANTIDAD, TIPO, PESO\_MAX, ALTO\_MAX, ANCHO\_MAX, LARGO\_MAX, PRECIO. La única Primary Key que escogimos fue PAQUETE\_ID y la Foreign Key es el atributo VIAJE.

## **PaquetesXViajes**

Aquí desarrollamos el primer entidad compuesta del modelo, la cual consta de los paquetes y los viajes, la misma surge de una ruptura de una relación *muchos a muchos*. Los atributos que surgen de ella son: PAQXVIA\_TIPO, PAQXVIA\_VIAJE y PAQXVIA\_CANTIDAD. Cuenta con dos Foreign Key, PAQXVIA\_TIPO y PAQXVIA\_VIAJE.

## **Orden de Trabajo**

En esta entidad contamos con los siguientes atributos: NRO\_TRABAJO, CAMION, ORDEN\_TRABAJO\_FECHA, ORDEN\_TRABAJO\_ESTADO. La Primary Key corresponde a NRO\_TRABAJO. Como orden de trabajo se relaciona con la entidad Camiones la misma posee su ID como Foreign Key.

## **OrdenDeTrabajoXTareas**

Cómo debemos contar la con información que corresponde a una tarea específica que será realizada en un momento determinado, decidimos juntos Orden de trabajo con la entidad Tareas. De esta manera los atributos que componen a esta nueva entidad son: NOMBRE\_TAREA, TIPO\_TAREA, MECANICO, ESTADO, FECHA\_INICIO\_PLANIFICADA, TIEMPO\_DE\_EJECUCION, FECHA\_INICIO\_REAL, TIEMPO\_ESTIMADO, FECHA\_FINAL. Así mismo cuenta con 3 Foreign Key: La orden de trabajo NOMBRE\_TAREA, la propia tarea que se realizará TIPO\_TAREA y el mecanico designado MECANICO.

## **Taller**

La entidad taller cuenta con ID\_TALLER, TALLER\_DIRECCION, TELEFONO, MAIL, NOMBRE, TALLER\_CIUADAD. La Primary Key correspondiente es ID\_TALLER

## **Mecanicos**

Continuamos con la entidad Mecanicos que está integrada por: NRO\_LEGAJOS, DNI, TALLER, NOMBRE, APELLIDO, MECANICOS\_DIRECCION, TELEFONO, MAIL, FECHA\_NAC, COSTO\_HORA. Las Primary Key son NRO\_LEGAJOS, DNI e incluimos la Foreign Key del taller mecánico donde trabaja cada uno de los mecánicos: TALLER.

## **Materiales**

Creamos la tabla Materiales, donde la misma nos permite normalizar el modelo. Se compone de los atributos: MATERIALES\_ID escogida como Primary Key, MATERIALES\_DESCRIPCION y MATERIALES\_PRECIO

## **Tarea**

Se compone de CÓDIGO como Primary Key y TIPO como Foreign Key

## **MaterialesXTarea**

Creamos la tabla MaterialesXTareas que nos permite relacionar los materiales con las tareas en las que se los requiere y la cantidad requerida para realizar cada una de las tareas. La cual cuenta con: MATXTAR\_MATERIALES, MATXTAR\_TAREA, Y MATXTAR\_CANTIDAD. Las Foreign Key escogidas son MATXTAR\_MATERIALES y MATXTAR\_TAREA.

Y para finalizar esta primera parte, decidimos crear tres tablas, donde en cada una de ellas encontraremos atributos como el ID de la misma y el DESARROLLO o NOMBRE correspondiente.

## **Ciudades**

Sus atributos son ID\_CIUDAD, su Primary Key, y NOMBRE

## **Estados\_Trabajo**

Cuenta con atributos como ID\_ESTADO y DESCRIPCION, ID\_ESTADO es la Primary Key

## **Marca**

Y por último, la entidad marca con ID\_MARCA como Primary Key y DESCRIPCION

## **Migración**

La segunda parte de este documento consta del desarrollo de la etapa de migración de datos. Como primer acercamiento, establecimos las distintas tablas que vamos a utilizar con sus respectivos tipos de datos que deben ser consistentes con los de la tabla maestra. Teniendo también en consideración sus Primary Keys y Foreign Keys. Por ejemplo, la entidad de Choferes la creamos de la siguiente manera:

```
CREATE TABLE [GRAN_EXCEL].[Choferes] (  
    [nro_legajo] INT IDENTITY,  
    [dni] DECIMAL(18, 0) NOT NULL,  
    [nombre] NVARCHAR(255) NOT NULL,  
    [apellido] NVARCHAR(255) NOT NULL,  
    [direccion] NVARCHAR(255) NOT NULL,  
    [telefono] INT NOT NULL,  
    [mail] NVARCHAR(255) NOT NULL,  
    [fecha_nacimiento] DATETIME2(3) NOT NULL,  
    [costo_hora] INT NOT NULL,  
    [rango_edad] NVARCHAR(10) NOT NULL,  
    PRIMARY KEY(nro_legajo)  
)
```

Ademas, para evitar problemas a la hora de hacer la migración, a algunas entidades que lo necesitaban, añadimos los Foreign Key correspondientes a tal entidad. Por ejemplo, en este caso con Modelos:

```
ALTER TABLE [GRAN_EXCEL].[Modelos] ADD CONSTRAINT modelos_id_marca FOREIGN KEY (id_marca) REFERENCES [GRAN_EXCEL].[Marcas](id_marca)
```

Una vez creadas todas las tablas, definimos las variables que se utilizarán para realizar los inserts y updates en las tablas creadas anteriormente.

Cuando se terminan de declarar todas las variables, se arranca a ejecutar los stored procedures. Y lo que consideramos para ejecutarlo es seleccionando todos los datos de la tabla maestra fila por fila, para lo cual se utiliza un cursor que permite que se realice de manera óptima. Y todos estos datos seleccionados se irán migrando a las tablas creadas que mencionamos anteriormente a través de INSERTs o UPDATEs. Aquí podemos ver algunos ejemplos de los Store Procedures creados, solo mostramos una cantidad limitada ya que en aspecto de desarrollo de los Store, todos son similares:

Esta fue la migración de la tabla Tareas

```
3 CREATE PROCEDURE [GRAN_EXCEL].[sp_carga_tareas]
AS
    SET IDENTITY_INSERT [GD2C2021].[GRAN_EXCEL].[Tareas] ON

3     INSERT INTO [GRAN_EXCEL].[Tareas](codigo, id_tipo_tarea, tiempo_estimado, descripcion)
    SELECT DISTINCT m.[TAREA_CODIGO], t.id_tipo_tarea, m.[TAREA_TIEMPO_ESTIMADO], m.[TAREA_DESCRIPCION]
    FROM [gd_esquema].[Maestra] m, [GRAN_EXCEL].[Tipos_tareas] t
    WHERE m.[TAREA_CODIGO] IS NOT NULL
    AND m.[TAREA_FECHA_INICIO_PLANIFICADO] IS NOT NULL
    AND m.[TAREA_TIEMPO_ESTIMADO] IS NOT NULL
    AND m.[TAREA_DESCRIPCION] IS NOT NULL
    AND t.descripcion = m.[TIPO_TAREA]

    SET IDENTITY_INSERT [GD2C2021].[GRAN_EXCEL].[Tareas] OFF
GO
```

La carga de ordenes

```
3 CREATE PROCEDURE [GRAN_EXCEL].[sp_carga_ordenes]
AS
3     INSERT INTO [GRAN_EXCEL].[Ordenes](trabajo_fecha, id_camion, id_trabajo_estado)
    SELECT DISTINCT m.[ORDEN_TRABAJO_FECHA], c.id_camion, e.id_estado
    FROM [gd_esquema].[Maestra] m, [GRAN_EXCEL].[Camiones] c, [GRAN_EXCEL].[Estados_trabajo] e
    WHERE m.[ORDEN_TRABAJO_FECHA] IS NOT NULL
    AND c.patente = m.[CAMION_PATENTE]
    AND e.descripcion = m.[ORDEN_TRABAJO_ESTADO]
GO
```

## **preguntar qué hizo**

```
3 CREATE PROCEDURE [GRAN_EXCEL].[sp_carga_materiales_x_tarea]
AS
3     INSERT INTO [GRAN_EXCEL].[MaterialesXTareas](id_material, id_tarea, cantidad)
    select mat.id_material, t.codigo, COUNT(m2.[MATERIAL_COD]) / (select count(distinct convert(varchar, m.[TAREA_FECHA_FIN])
    +convert(varchar, m.[TAREA_FECHA_INICIO])
    +convert(varchar, m.[TAREA_FECHA_INICIO_PLANIFICADO])
    +str( m.[TAREA_TIEMPO_ESTIMADO])+ m.[CAMION_PATENTE])
    from [gd_esquema].[Maestra] m
    WHERE m.[TAREA_CODIGO]= m2.[TAREA_CODIGO] and m.[TAREA_DESCRIPCION] <> 'null')
    from [gd_esquema].[Maestra] m2, [GRAN_EXCEL].[Materiales] mat, [GRAN_EXCEL].[Tareas] t
    WHERE m2.[TAREA_DESCRIPCION] IS NOT NULL
    AND mat.codigo = m2.[MATERIAL_COD]
    AND t.codigo = m2.[TAREA_CODIGO]
    group by mat.id_material, m2.[TAREA_CODIGO], t.codigo
GO
```

## **preguntar qué es eso**



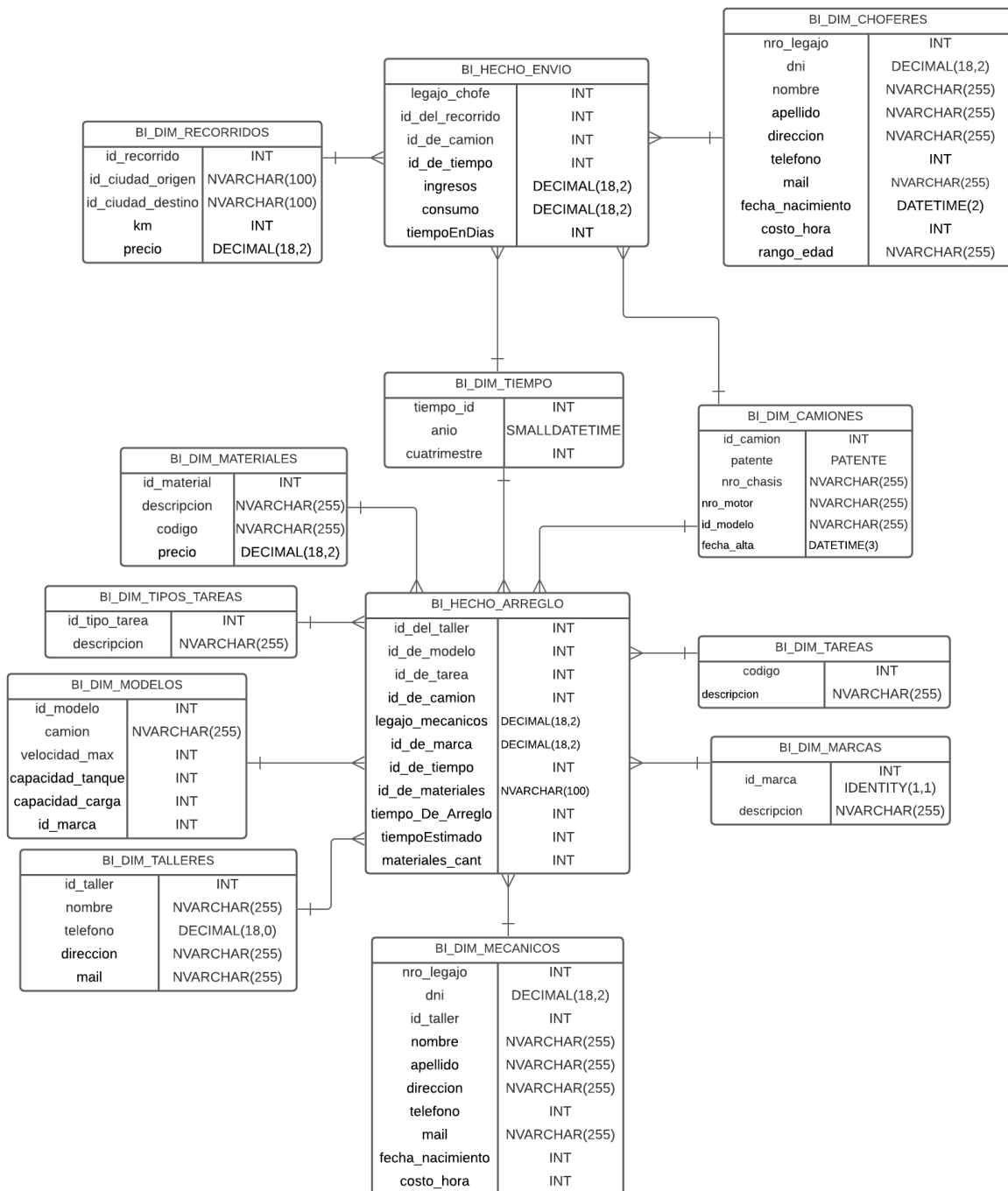
```
CREATE PROCEDURE [GRAN_EXCEL].[sp_migrate_data]
AS
EXEC [GRAN_EXCEL].[sp_carga_materiales]
EXEC [GRAN_EXCEL].[sp_carga_ciudades]
EXEC [GRAN_EXCEL].[sp_carga_talleres]
EXEC [GRAN_EXCEL].[sp_carga_mecanicos]
EXEC [GRAN_EXCEL].[sp_carga_marcas]
EXEC [GRAN_EXCEL].[sp_carga_modelos]
EXEC [GRAN_EXCEL].[sp_carga_camiones]
EXEC [GRAN_EXCEL].[sp_carga_recorridos]
EXEC [GRAN_EXCEL].[sp_carga_choferes]
EXEC [GRAN_EXCEL].[sp_carga_viajes]
EXEC [GRAN_EXCEL].[sp_carga_estados_trabajo]
EXEC [GRAN_EXCEL].[sp_carga_ordenes]
EXEC [GRAN_EXCEL].[sp_carga_tipos_paquetes]
EXEC [GRAN_EXCEL].[sp_carga_paquetes_x_viaje]
EXEC [GRAN_EXCEL].[sp_carga_tipos_tareas]
EXEC [GRAN_EXCEL].[sp_carga_tareas]
EXEC [GRAN_EXCEL].[sp_carga_tareas_x_ordenes]
EXEC [GRAN_EXCEL].[sp_carga_materiales_x_tarea]

GO

EXEC [GRAN_EXCEL].[sp_migrate_data]
GO
```

## **Desarrollo del Modelo de Inteligencia de Negocios (BI)**

Comenzamos exponiendo el DER correspondiente al modelo de inteligencias de negocios:



Las Entidades de la implementación de la primera parte que escogimos para hacer las dimensiones en esta parte del Trabajo Práctico fueron: Recorridos, Choferes, Camiones, Materiales, Tipos\_tareas, Modelos, Tareas, Marcas, Talleres, Mecanicos. Y por otro lado creamos la dimensión Tiempo en el script del BI directamente.

Luego de cargar por completo todas las dimensiones, se procede a implementar las tablas de hecho:

**BI\_hecho\_arreglo:** para este primero hecho, lo que pensamos fue poner todos los valores que tengan relación con el camión y su correspondiente arreglo. Aquí podemos encontrar a los atributos: id\_del\_taller, id\_de\_modelo, id\_de\_tarea, id\_de\_camion, legajo\_mecanicos, id\_de\_marca, id\_de\_tiempo, id\_de\_materiales, tiempo\_De\_Arreglo, tiempoEstimado, materiales\_cant

```
CREATE TABLE GRAN_EXCEL.[BI_hecho_arreglo](
    [id_del_taller] int,
    [id_de_modelo] int,
    [id_de_tarea] int,
    [id_de_camion] int,
    [legajo_mecanicos] int,
    [id_de_marca] int,
    [id_de_tiempo] int,
    [id_de_materiales] nvarchar(100),
    [tiempo_De_Arreglo] int,
    [tiempoEstimado] int,
    [materiales_cant] int
)

insert into GRAN_EXCEL.BI_hecho_arreglo
select distinct [id_taller], mdl.[id_modelo], [id_tarea], c.[id_camion], m1.[nro_legajo], [id_marca], tiempo_id, [id_material], [tiempo_real_dias], [tiempoEstimado], [cantidad]
from GRAN_EXCEL.[TareasXOrdenes]
join GRAN_EXCEL.BI_DIM_TIPO_TAREA on [id_tarea] = [id_tarea]
join GRAN_EXCEL.[Tareas] t1 on t1.[id_tarea] = [id_tarea]
join GRAN_EXCEL.BI_DIM_MECHANICO m1 on m1.[nro_legajo] = [legajo_mecanico]
join GRAN_EXCEL.[Mecanicos] m on m1.[nro_legajo] = m.[nro_legajo]
join GRAN_EXCEL.BI_DIM_TALLER on m.[id_taller] = [id_taller]
join GRAN_EXCEL.BI_DIM_TIEMPO on year([fecha_inicio]) = anio and DATEPART(quarter,[fecha_inicio]) = cuatrimestre
join GRAN_EXCEL.[Ordenes] on [id_orden] = [nro_trabajo]
join GRAN_EXCEL.[Camiones] c on [id_camion] = c.[id_camion]
join GRAN_EXCEL.[Modelos] mdl on mdl.[id_modelo] = c.[id_modelo]
join GRAN_EXCEL.BI_DIM_MARCA on mdl.[id_marca] = [id_modelo]
join GRAN_EXCEL.[MaterialesXTareas] on [id_tarea] = [id_tarea]
```

**BI\_Hecho\_Envio:** seguimos con el siguiente hecho, el cual va a relacionarse con los Viajes. Es por eso que decidimos insertar los atributos relacionados con tal evento (viaje), ellos son: legajo\_chofe, id\_del\_recorrido, id\_de\_camion, id\_de\_tiempo, ingresos, consumo, tiempoEnDias. En el código se desarrollaron de la siguiente manera:

```
CREATE TABLE GRAN_EXCEL.[BI_hecho_envio](
    [legajo_chofe] int,
    [id_del_recorrido] int,
    [id_de_camion] int,
    [id_de_tiempo] int,
    [ingresos] decimal(18,2),
    [consumo] decimal(18,2),
    [tiempoEnDias] int
)
```

```

insert into GRAN_EXCEL.BI_hecho_envio
select distinct [legajo_chofer_designado], [id_recorrido], [id_camion_designado], tiempo_id, sum([cantidad] * [precio]+[precio]), [consumo_combustible] , datediff(day,[fecha_inicio], [fecha_fin])
from GRAN_EXCEL.[Viajes]
join GRAN_EXCEL.BI_DIM_TIEMPO on year([fecha_inicio]) = anio and DATEPART(quarter,[fecha_inicio]) = cuatrimestre
join GRAN_EXCEL.[PaquetesViajes] on [id_viaje] = [id_viaje]
join GRAN_EXCEL.[Tipos_paquetes] on [id_tipo_paquete] = [id_tipo]
join GRAN_EXCEL.[Choferes] on [legajo_chofer_designado] = [nro_legajo]
join GRAN_EXCEL.BI_DIM_RECORRIDO on [id_recorrido] = [id_recorrido]
group by [legajo_chofer_designado], [id_recorrido], [id_camion_designado], tiempo_id, [consumo_combustible],[fecha_inicio], [fecha_fin]

```

Para finalizar, lo último desarrollado en esta etapa del Trabajo Práctico fueron las vistas, las mismas se implementaron de la siguiente forma:

### **BI\_tiempo\_maximo\_fuera\_de\_servicio**

Un camión se encuentra fuera de servicio cuando está en el taller, impidiéndole realizar los viajes designados.

```

IF OBJECT_ID ('GRAN_EXCEL.BI_tiempo_maximo_fuera_de_servicio', 'V') IS NOT NULL
DROP view GRAN_EXCEL.BI_tiempo_maximo_fuera_de_servicio;
GO
create view GRAN_EXCEL.BI_tiempo_maximo_fuera_de_servicio
as

select distinct id_de_camion Camion , cuatrimestre Cuatrimestre, max(tiempo_De_Arreglo) AS 'Tiempo Maximo'
from GRAN_EXCEL.BI_hecho_arreglo
join GRAN_EXCEL.BI_DIM_TIEMPO on id_de_tiempo = tiempo_id
group by cuatrimestre ,id_de_camion

go

```

### **BI costo total mantenimiento por camion**

Se entiende por costo lo siguiente: costo de materiales + el costo de mano de obra que se utilizó en cada tarea

```

IF OBJECT_ID ('GRAN_EXCEL.BI_costo_total_mantenimiento_por_camion', 'V') IS NOT NULL
DROP view GRAN_EXCEL.BI_costo_total_mantenimiento_por_camion;
GO
create view GRAN_EXCEL.BI_costo_total_mantenimiento_por_camion
as

select id_de_camion, [id_del_taller], cuatrimestre, sum([cantidad] * [precio]) + (sum( [costo_hora] * 8 * tiempo_De_Arreglo)/ count(distinct id_de_materiales)) AS 'Costo Total'
from GRAN_EXCEL.BI_hecho_arreglo
join GRAN_EXCEL.BI_DIM_TIEMPO on id_de_tiempo = tiempo_id
join GRAN_EXCEL.BI_DIM_MATERIAL on id_de_materiales = [id_material]
join GRAN_EXCEL.BI_DIM_MECANICO on [nro_legajo] = legajo_mecanicos
join GRAN_EXCEL.BI_DIM_MATERIALESXTAREA on [id_tarea] = id_de_tarea
group by id_de_camion, [id_del_taller], cuatrimestre

go

```

### **BI\_desvio\_promedio\_tarea\_por\_taller**

```

IF OBJECT_ID ('GRAN_EXCEL.BI_desvio_promedio_tarea_por_taller', 'V') IS NOT NULL
DROP view GRAN_EXCEL.BI_desvio_promedio_tarea_por_taller;
GO
create view GRAN_EXCEL.BI_desvio_promedio_tarea_por_taller
as

select id_de_tarea, [id_del_taller], avg(abs(tiempo_De_Arreglo - tiempoEstimado)) AS 'desvio'
from GRAN_EXCEL.BI_hecho_arreglo
group by id_de_tarea, [id_del_taller]

go

```

## **BI 5 tareas mas realizadas por modelo camion**

```
IF OBJECT_ID ('GRAN_EXCEL.BI_5_tareas_mas_realizadas_por_modelo_camion', 'V') IS NOT NULL
    DROP view GRAN_EXCEL.BI_5_tareas_mas_realizadas_por_modelo_camion;
GO
create view GRAN_EXCEL.BI_5_tareas_mas_realizadas_por_modelo_camion
as
    select b.id_de_tarea, b.id_de_modelo
    from GRAN_EXCEL.BI_hecho_arreglo b
    where b.id_de_tarea in (
        select top 5 id_de_tarea
        from GRAN_EXCEL.BI_hecho_arreglo
        where id_de_modelo = b.id_de_modelo
        group by id_de_modelo, id_de_tarea
        order by count(id_de_tarea) desc
    )
    group by b.id_de_modelo, b.id_de_tarea

go
```

## **BI los 10 materiales mas utilizados**

```
IF OBJECT_ID ('GRAN_EXCEL.BI_los_10_materiales_mas_utilizados', 'V') IS NOT NULL
    DROP view GRAN_EXCEL.BI_los_10_materiales_mas_utilizados;
GO
create view GRAN_EXCEL.BI_los_10_materiales_mas_utilizados
as
    select id_de_materiales, [id_del_taller] from GRAN_EXCEL.BI_hecho_arreglo b
    join GRAN_EXCEL.BI_DIM_MATERIAL on [id_material] = id_de_materiales
    where id_de_materiales in (
        select top 10 id_de_materiales
        from GRAN_EXCEL.BI_hecho_arreglo
        where [id_del_taller] = b.[id_del_taller]
        group by id_de_materiales
        order by sum(materiales_cant) desc
    )
    group by [id_del_taller], id_de_materiales

go
```

## **BI facturacion total por recorrido**

Esto depende de la cantidad y los tipos de paquetes que el camión transporta y además del recorrido que se efectúe

```
IF OBJECT_ID ('GRAN_EXCEL.BI_facturacion_total_por_recorrido', 'V') IS NOT NULL
    DROP view GRAN_EXCEL.BI_facturacion_total_por_recorrido;
GO
create view GRAN_EXCEL.BI_facturacion_total_por_recorrido
as
    select id_del_recorrido, cuatrimestre, sum(ingresos) AS 'Facturacion Total'
    from GRAN_EXCEL.BI_hecho_envio
    join GRAN_EXCEL.BI_DIM_TIEMPO on tiempo_id = id_de_tiempo
    group by id_del_recorrido, cuatrimestre

go
```

## **BI costo promedio por rango etario de choferes**

```
IF OBJECT_ID ('GRAN_EXCEL.BI_costo_promedio_por_rango_etario_de_choferes', 'V') IS NOT NULL
    DROP view GRAN_EXCEL.BI_costo_promedio_por_rango_etario_de_choferes;
GO
create view GRAN_EXCEL.BI_costo_promedio_por_rango_etario_de_choferes
as
    select (select sum([costo_hora])
    from GRAN_EXCEL.BI_DIM_CHOFER
    where [rango_edad_chofer] = c.[rango_edad_chofer])/ count(distinct [nro_legajo]) costo, [rango_edad_chofer]
    from GRAN_EXCEL.BI_hecho_envio
    join GRAN_EXCEL.BI_DIM_CHOFER c on c.[nro_legajo] = legajo_chofe
    group by [rango_edad_chofer]

go
```

## **BI ganancia por camion**

```
IF OBJECT_ID ('GRAN_EXCEL.BI_ganancia_por_camion', 'V') IS NOT NULL
    DROP view GRAN_EXCEL.BI_ganancia_por_camion;
GO
create view GRAN_EXCEL.BI_ganancia_por_camion
as
    select e.id_de_camion, sum(e.ingresos) -
    sum((e.consumo*100)+(e.tiempoEnDias * 8 * [costo_hora]))
    - sum(materiales_cant * [precio]) + (sum( [costo_hora] * 8 * tiempo_De_Arreglo) /count(distinct [id_material])) AS 'Ganancia'
    from GRAN_EXCEL.BI_hecho_envio e
    join GRAN_EXCEL.BI_DIM_CHOFER on e.legajo_chofe = [nro_legajo]
    join GRAN_EXCEL.BI_hecho_arreglo a on a.id_de_camion = e.id_de_camion
    join GRAN_EXCEL.BI_DIM_MATERIALES on a.id_de_materiales = [id_material]
    join GRAN_EXCEL.BI_DIM_MECANICO on [nro_legajo] = a.legajo_mecanicos
    group by e.id_de_camion

go
```