



TRABAJO PRÁCTICO GESTIÓN DE DATOS - GRUPO 3

Profesor: Reinosa Enrique

Nombre y Apellido	Legajo
Camila Kaner Loy	167771-8
Nicolas Benvenaste	167418-3
Luciano Capomolla	160848-5
Constanza González Pósleman	172206-2

Fecha de entrega: 26/10/2021

ÍNDICE

Estrategia - Modelo de datos	3
Stored Procedures:	5
Estrategia - Modelo de Inteligencia de Negocios	7
Requerimientos	7
Tablas de hechos	8
Stored Procedures	9
Funciones	10
Vistas asociadas	10
Conclusión	11

Estrategia - Modelo de datos

Analizando los datos y nombres brindados por la tabla maestra, identificamos las siguientes entidades, algunas decidimos dar una breve explicación y otras nos detendremos puntualmente luego de la presentación de todas:

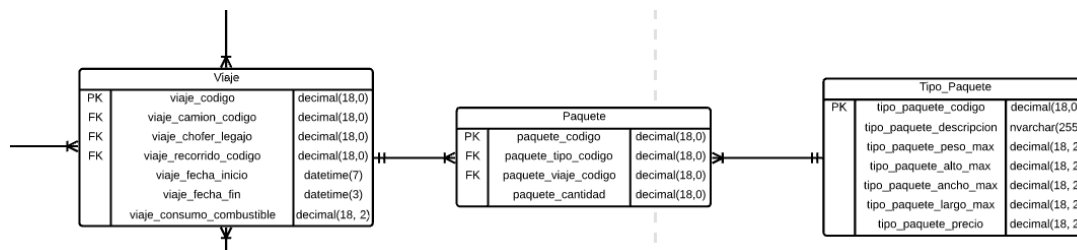
- **Viaje**
- **Paquete**
- **Tipo_Paquete**
- **Recorrido:** Además de contar con un identificador unívoco, presenta dos FK que se relacionarán con *Ciudad*, estos representarán los lugares de inicio y final del viaje.
- **Chofer:** En este caso, decidimos utilizar como PK el número de legajo del mismo, el cual es unívoco, evitando la necesidad de generar otro campo como *cod_cliente*.
- **Camion:** Decidimos asignarle a cada camión un valor de *cod_codigo*, siendo este su PK, para poder identificar unívocamente a los mismos. Además, esta entidad presenta una FK (*camion_modelo_codigo*) que se relaciona con una entidad que almacena los modelos y por otra parte un FK (*camion_marca_codigo*) que almacenará las marcas.
- **Marca**
- **Modelo**
- **Orden_Trabajo**
- **Estado**
- **Taller**
- **Ciudad**
- **Tarea_a_Realizar**
- **Mecánico:** En esta entidad, se presenta una situación similar a la de Chofer. En la cual utilizamos como PK a su número de legajo evitando tener información innecesaria.
- **Tarea**
- **Tipo_Tarea**
- **Material_x_Tarea**
- **Material**

Algunas entidades presentan un comportamiento similar en cuanto a que tienen una clave unívoca y una descripción, como por ejemplo *Marca*, *Modelo*, etc.

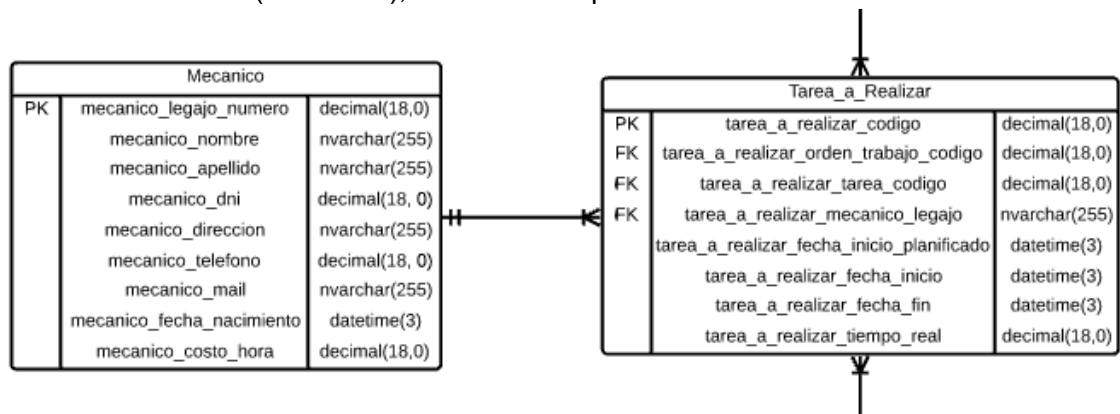
Comenzamos con las relaciones asociadas al *Viaje*. Decidimos crear 2 tablas para poder representar por un lado los tipos de paquetes que puede tener un paquete y por el otro los paquetes en sí mismo, que además, nos permiten saber su cantidad. Esto nos permite evitar la repetición de información, tener disponible en una tabla los diversos tipos de paquetes y nos ofrece la capacidad de que cada paquete, mediante una FK, se asocie con un tipo paquete. Además, es destacable que el tipo de un paquete es una característica que no se consulta frecuentemente.

Posterior a lo relacionado con el viaje, nos enfocaremos en las reparaciones de los camiones. Para evitar la repetición de información, creamos una tabla llamada *Tarea* donde se encuentran las distintas tareas predeterminadas que se pueden realizar en un camión. En esta tabla se encuentra información propia y relaciones con otras tablas, como la de materiales (el cual será desarrollado luego).

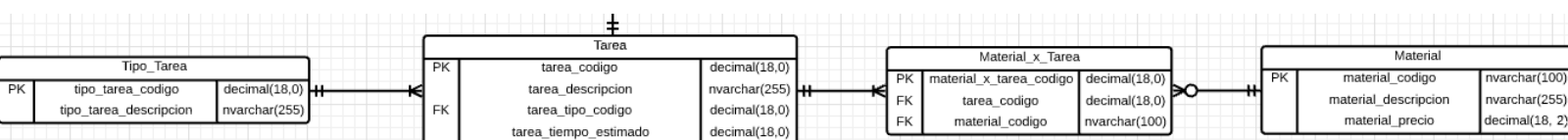
También disponemos de una tabla donde se guardaran las tareas que se realizarán específicamente por cada camión (*Tarea_a_Realizar*). Esta dispone de información puntual sobre la tarea que se realizará, desde el mecánico hasta los tiempos y las fechas.



Además, mediante el uso de FK, para los mecánicos, se dispondrá una única tabla donde se encontrarán todos (*Mecánico*), evitando la repetición de datos.



A continuación detallamos la relación de la tarea con su tipo, creamos una tabla *Tipo_Tarea* para evitar la repetición de información en Tarea. Volviendo a los materiales, disponemos de 2 tablas, la tabla del material (*Material*) en sí y otra que representa el material que se usa por cada tarea (*Material_x_Tarea*) ya que la relación de la tarea con su material es de muchos a muchos y para su implementación, se recurre a una tabla intermedia. Por otra parte, la Tarea también se relaciona mediante un FK a una entidad donde se almacenan los diversos tipos de tareas que existen.



Stored Procedures:

```
CREATE PROCEDURE UTNIX.Migrar_Material
CREATE PROCEDURE UTNIX.Migrar_Material_x_Tarea
CREATE PROCEDURE UTNIX.Migrar_Tipo_Tarea
CREATE PROCEDURE UTNIX.Migrar_Tarea
CREATE PROCEDURE UTNIX.Migrar_Tipo_Paquete
CREATE PROCEDURE UTNIX.Migrar_Paquete
CREATE PROCEDURE UTNIX.Migrar_Chofer
CREATE PROCEDURE UTNIX.Migrar_Modelo
CREATE PROCEDURE UTNIX.Migrar_Marca
CREATE PROCEDURE UTNIX.Migrar_Mecanico
CREATE PROCEDURE UTNIX.Migrar_Taller
CREATE PROCEDURE UTNIX.Migrar_Estado
CREATE PROCEDURE UTNIX.Migrar_Orden_Trabajo
CREATE PROCEDURE UTNIX.Migrar_Tarea_a_Realizar
CREATE PROCEDURE UTNIX.Migrar_Ciudad
CREATE PROCEDURE UTNIX.Migrar_Recorrido
CREATE PROCEDURE UTNIX.Migrar_Viaje
CREATE PROCEDURE UTNIX.Migrar_Camion
```

A continuación, presentaremos el DER completo con todas sus relaciones y entidades:



Estrategia - Modelo de Inteligencia de Negocios

Requerimientos

1) Máximo tiempo fuera de servicio de cada camión por cuatrimestre Se entiende por fuera de servicio cuando el camión está en el taller (tiene una OT) y no se encuentra disponible para un viaje.

2) Costo total de mantenimiento por camión, por taller, por cuatrimestre. Se entiende por costo de mantenimiento el costo de materiales + el costo de mano de obra insumido en cada tarea (correctivas y preventivas)

3) Desvío promedio de cada tarea x taller.

4) Las 5 tareas que más se realizan por modelo de camión.

5) Los 10 materiales más utilizados por taller 6) Facturación total por recorrido por cuatrimestre. (En función de la cantidad y tipo de paquetes que transporta el camión y el recorrido)

7) Costo promedio por rango etario de choferes.

8) Ganancia por camión (Ingresos – Costo de viaje – Costo de mantenimiento)

- ❖ Ingresos: en función de la cantidad y tipo de paquetes que transporta el camión y el recorrido.
- ❖ Costo de viaje: costo del chofer + el costo de combustible. Tomar el precio del litro de combustible \$100.
- ❖ Costo de mantenimiento: costo de materiales + costo de mano de obra.

Tablas de dimensiones

En base a lo requerido y a las peticiones que se le harán al sistema, inicialmente nombraremos las dimensiones necesarias que se encontraran en el mismo, siendo las mismas utilizadas en diversas tablas de hechos:

- **BI_Recorrido**
- **BI_Camion**
- **BI_Modelo**
- **BI_Tarea**
- **BI_Tiempo**
- **BI_Paquete**
- **BI_Mecánico**
- **BI_Taller**
- **BI_Material**
- **BI_Rango_Etario**
- **BI_Chofer**

Tablas de hechos

Estas dimensiones serán utilizadas para crear las diversas tablas de hechos y además, algunas de ellas, son utilizadas en más de una de estas. Dado los requerimientos al modelo del BI y evitando tener demasiados campos en valor NULL. Es destacable que cada tabla de hechos contiene sus PRIMARY KEYS en los valores que harán el acceso unívoco a la información como sus FOREIGN KEYS a sus dimensiones. Decidimos realizar varias, otorgando las funcionalidades a:

- **BI_Hechos_Mantenimiento_Camion:** se agregaron los campos “maximo_tiempo_fuera_de_servicio” y “costo_total_de_mantenimiento”, dado que esta tabla de hechos responderá a los requerimientos 1 y 2.
- **BI_Hechos_Facturaciones:** En esta tabla de hechos se le agregan los campos: “facturacion_total_recorrido” y “bi_tarea”, colocaran los resultados a los requerimientos 4 y 6.
- **BI_Hechos_Taller_x_Tareas:** Se agregara el campo “desvio_promedio” y aquellas variables relacionadas para almacenar el TOP 10 de las tareas más realizadas por taller, respondiendo a 3 y 5.
- **BI_Costo_Promedio_Etario:** Para ser capaz de responder al requerimiento 7, se agrega la variable “costo_promedio”, donde se almacenará el valor
- **BI_Hechos_Ganancias:** Por último, para responder al requerimiento restante, es decir el 8, decidimos crear una tabla de hechos independiente ya que toda la funcionalidad está relacionada a temas económicos. En esta, agregamos el campo “ganancias” que almacenará las ganancias mismas de cada camión.

Stored Procedures

Luego de la creación de las diversas tablas tanto de dimensiones como las de hechos implementamos diversos stored procedures para realizar la migración de los datos del Modelo de Datos Transaccional hacia nuestras nuevas tablas del Modelo BI:

- [UTNIX].BI_Migrar_Rango_Edad
- [UTNIX].BI_Migrar_Tiempo
- [UTNIX].BI_Migrar_Tipo_Tarea
- [UTNIX].BI_Migrar_Tarea
- [UTNIX].BI_Migrar_Material
- [UTNIX].BI_Migrar_Material_x_Tarea
- [UTNIX].BI_Migrar_Ciudad
- [UTNIX].BI_Migrar_Recorrido
- [UTNIX].BI_Migrar_Modelo
- [UTNIX].BI_Migrar_Marca
- [UTNIX].BI_Migrar_Camion
- [UTNIX].BI_Migrar_Estado
- [UTNIX].BI_Migrar_Mecanico
- [UTNIX].BI_Migrar_Taller
- [UTNIX].BI_Migrar_Chofer
- [UTNIX].BI_Migrar_Orden_Trabajo
- [UTNIX].BI_Migrar_Tarea_a_Realizar
- [UTNIX].BI_Migrar_Viaje
- [UTNIX].BI_Migrar_Tipo_Paquete
- [UTNIX].BI_Migrar_Paquete

Es destacable que también se migrarán los datos hacia las tablas de hechos:

- [UTNIX].BI_Migrar_Hechos_Mantenimiento_Camion
- [UTNIX].BI_Migrar_Hechos_Costo_Promedio_Etario
- [UTNIX].BI_Migrar_Hechos_Facturaciones

Funciones

Dentro del script decidimos crear varias funciones que nos ayudarán en distintos procedimientos o incluso, dentro de otras funciones. Nombraremos cada una de ellas junto a un pequeño desarrollo para dar contexto a que funcionamiento le otorga al sistema:

- **UTNIX.Obtener_Cuatrimestre(DATETIME2(3))**: A esta función le ingresa una fecha y será capaz de responder un código de cuatrimestre. Cada cuatrimestre dentro del sistema tendrá asignado un número identificador, es decir, por ejemplo, el primer cuatrimestre del año 2019 podría tener asignado el ID 1, el segundo ID 2 y manteniendo la secuencia, el primer cuatrimestre del 2020 tendrá asignado el ID 5.
- **UTNIX.Obtener_ID_Tiempo (DATETIME2(3))**: Al integrarle una fecha, responderá a qué cuatrimestre corresponde, habiendo 4 posibles por año como su nombre lo indica.
- **UTNIX.Obtener_Nombre_Ciudad (DECIMAL(18,0))**: Al ingresar un código que referencia a una ciudad, esta función responderá el nombre de la misma.
- **UTNIX.Obtener_Costo_Materiales_x_OT (DECIMAL(18,0))**: Tendrá como funcionalidad retornar el valor gastado en materiales dentro de una orden de trabajo
- **UTNIX.Obtener_Costo_Materiales_x_Camion (DECIMAL(18,0))**: Es capaz de devolver los costos de los materiales que fueron utilizados en un camión
- **UTNIX.Obtener_Costo_Mano_De_Obra_De_OT (DECIMAL(18,0))**: Al ingresar una orden de trabajo, devolverá la mano de obra asociada a una orden de trabajo
- **UTNIX.Obtener_Costo_Mano_de_Obra_por_camion (DECIMAL(18,0))**: Al ingresarle un código que hace referencia a un camión, devolverá la suma de los costos asociados a la misma de mano de obra teniendo en cuenta tanto el costo por hora individual de cada trabajador y el tiempo implicado para la realización de los respectivos arreglos.
- **UTNIX.Obtener_Rango_Etario (DATETIME2(3))**: Esta función, dada una de las funcionalidades, es capaz de decir, en función del día de nacimiento de un empleado, a que categoría etaria pertenece. Esta categorización se realiza en base a: Personas entre 18 y 30 años, personas entre 31 a 50 y por último mayores de 50.
- **UTNIX.Obtener_Costo_Chofer (DECIMAL(18,0))** : Al ingresar un camión, devolverá el costo total de los choferes que lo han utilizado.

Vistas asociadas

- Máximo tiempo fuera de servicio de cada camión por cuatrimestre. Se entiende por fuera de servicio cuando el camión está en el taller (tiene una OT) y no se encuentra disponible para un viaje

v_Maximo_Tiempo_Fuera_de_Servicio

- Costo total de mantenimiento por camión, por taller, por cuatrimestre. Se entiende por costo de mantenimiento: el costo de materiales + el costo de mano de obra insumido en cada tarea (correctivas y preventivas)

v_Mantenimiento_x_Camion

- Desvío promedio de cada tarea por taller con respecto a los tiempos de inicio de las tareas a realizar

v_Desvio_Promedio_Tarea

- Las 5 tareas que más se realizan por modelo de camión

v_Tareas_x_Modelo

- Los 10 materiales más utilizados por taller

v_Materiales_x_Taller

- Facturación total por recorrido por cuatrimestre. (En función de la cantidad y tipo de paquetes que transporta el camión y el recorrido)

v_Facturacion_x_Recorrido

- Costo promedio por rango etario de choferes

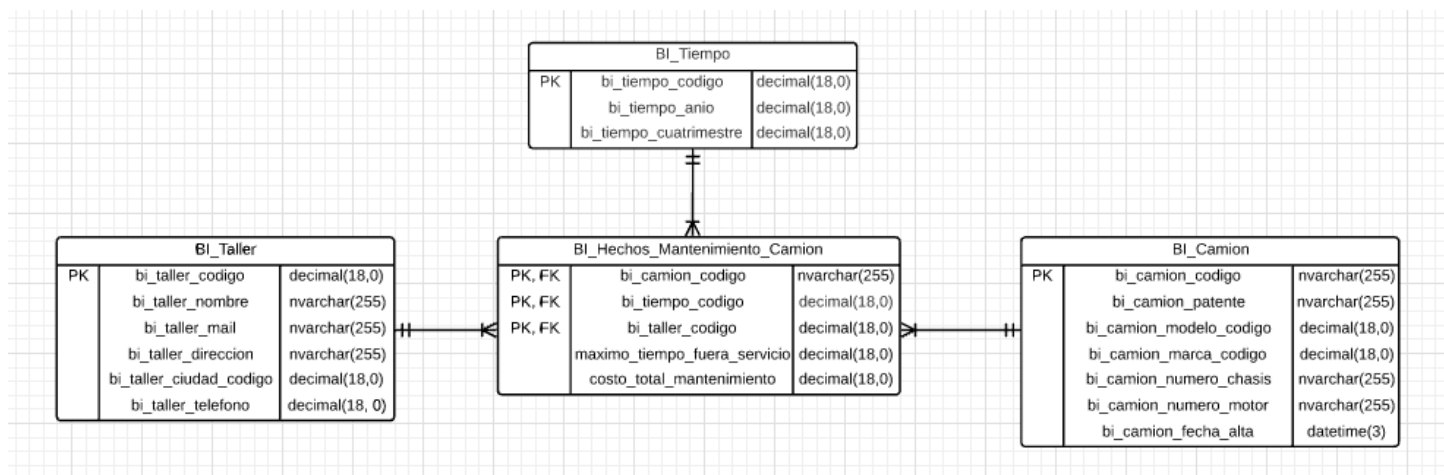
v_Promedio_x_Rango_etario

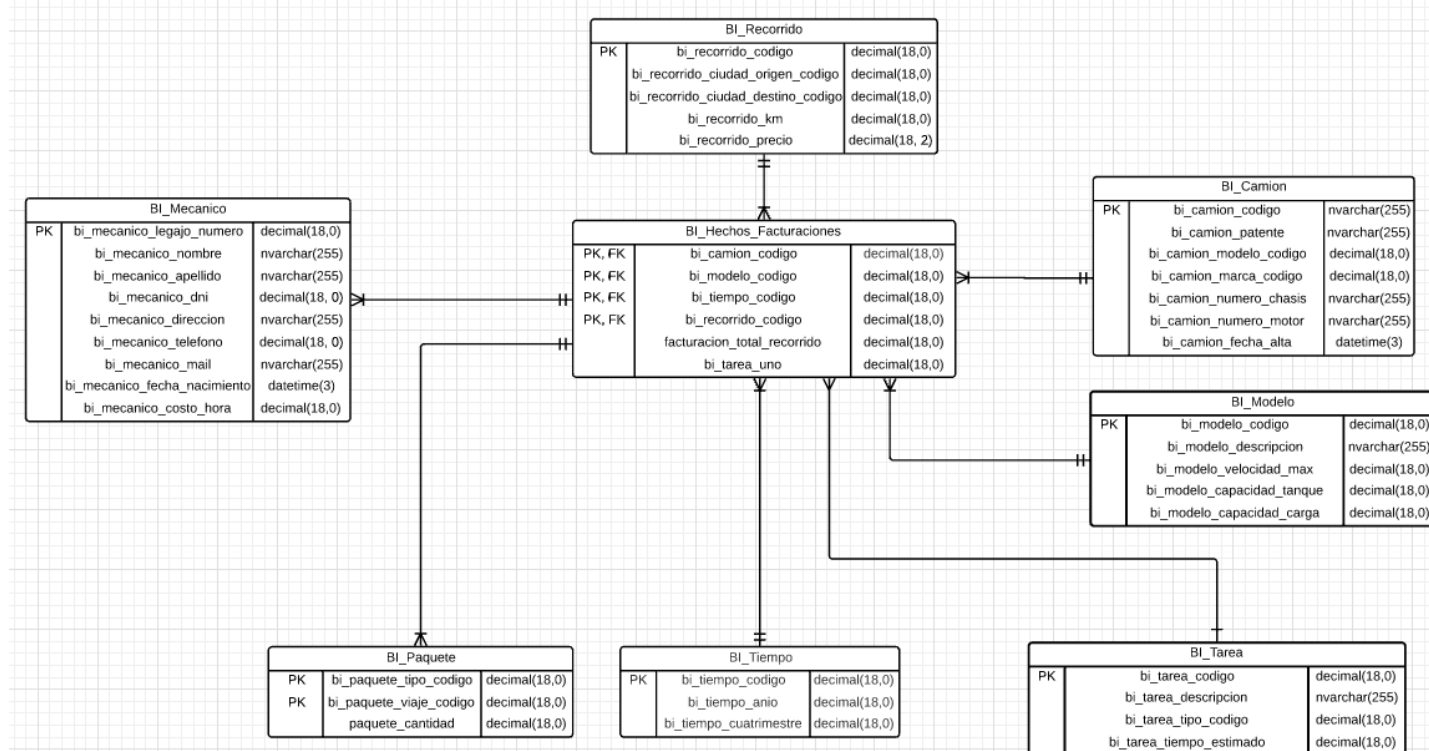
- Ganancia por camión (Ingresos – Costo de viaje – Costo de mantenimiento)

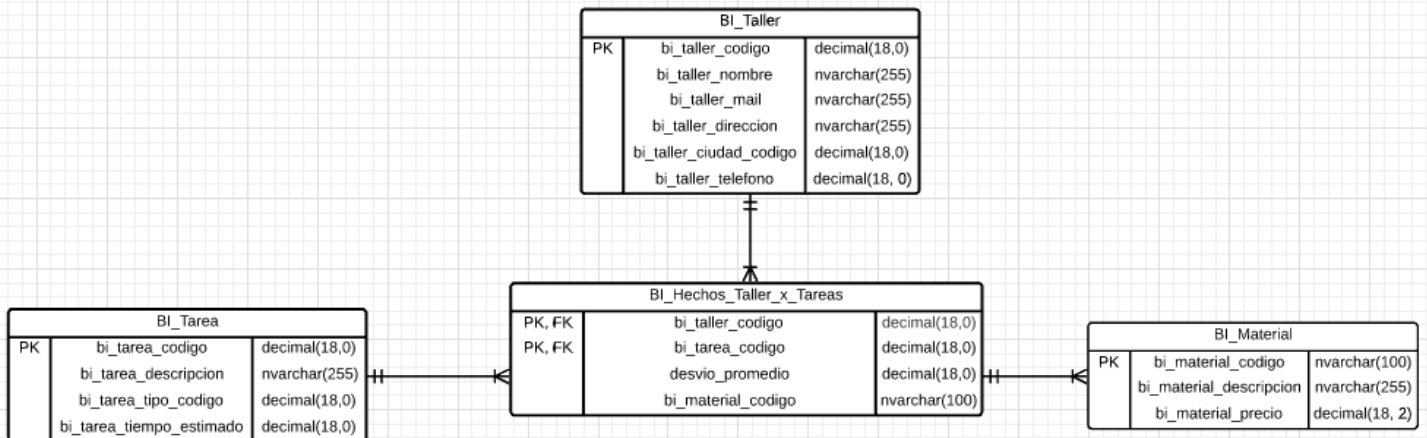
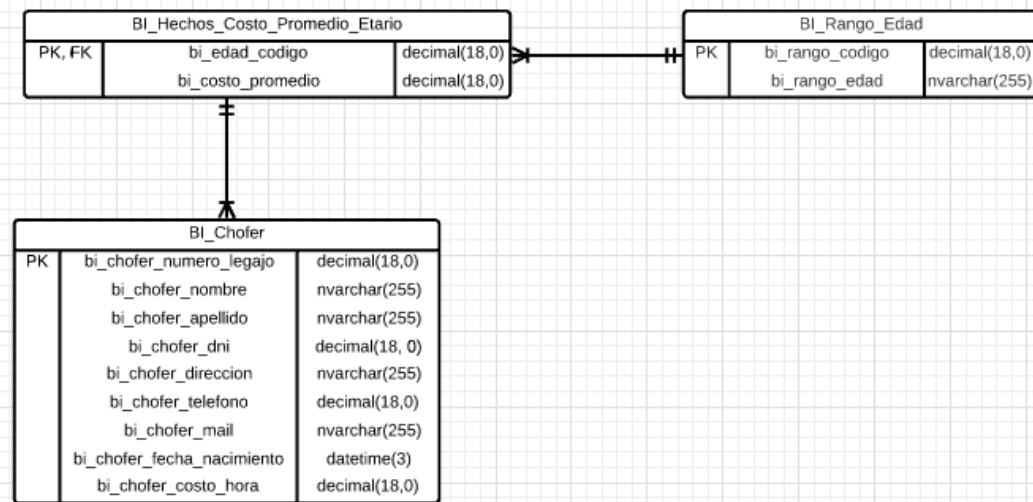
v_Ganancias_x_Camion

Diagrama de entidad - modelo de inteligencia de negocios

Por último, adjuntamos todas las imágenes que reflejan tanto las dimensiones como los hechos que creímos relevantes para realizar las solicitudes pedidas por consigna:







Conclusión

Gracias al modelo de BI creado teniendo en cuenta desde las tablas implicadas (desde dimensiones y hechos) hasta las funciones y procedimientos, se es capaz de analizar tanto patrones como tendencias para la toma de decisiones futuras. Es destacable que gracias a las vistas implicadas, se gana tanto seguridad como simpleza en los resultados, siendo así el sistema más práctico para el cliente.

El proceso utilizado para la creación del mismo parte desde la migración a tablas denominadas dimensiones que aportaran sus datos a tablas llamada hechos, que serán las encargadas de calcular aquellos datos que serán beneficiarios para el negocio.



Es notable que las tablas de dimensiones no deben coincidir exactamente con las tablas del modelo relacional, siendo estas capaces de almacenar y calcular datos mucho más oportunos y beneficiosos al modelo BI. Además, queríamos destacar la capacidad que presentan las tablas de hechos, que en base a las distintas vistas creadas, son capaces de responder a varios requerimientos cuyas funcionalidades pueden o no estar relacionadas.

Optamos por crear varias tablas de hechos, ya que, a pesar de que pudiésemos hacer una sola que sea capaz de responder a todos los ítems solicitados, presentaría muchos campos en valor NULL; por lo que, se crearon varias, asociando en cada una de ellas, aquellos requerimientos que presenten cierta similitud entre ellos, sea en base a las dimensiones que se necesitan para implementarse como en el porqué de su creación (por ejemplo, la tabla C presenta funcionalidades y dimensiones asociadas x taller).