AVANCE N°3 - CUBO OLAP

Caso DUOC Gourmet

**INTEGRANTES:** LILIAN JULIO **FECHA:** 25-11-2020

LIENTUR QUELEMPAN

JONATHAN MOYA

FRANCISCO JARA

**DOCENTE:** ALDO ALBERTO MARTINEZ ORDENES

ÍNDICE

[ÍNDICE 1](#_Toc57236790)

[INTRODUCCIÓN 2](#_Toc57236791)

[CONTEXTO 3](#_Toc57236792)

[PROBLEMA DEL CASO DUOC GOURMET 3](#_Toc57236793)

[DISEÑO Y CONSTRUCCION DE MODELO BI 4](#_Toc57236794)

[MODELO DE PROCESOS DE VENTA 4](#_Toc57236795)

[REGLAS DE NEGOCIO 5](#_Toc57236796)

[NIVEL DE GRANULARIDAD 5](#_Toc57236797)

[DIMENSIONES 6](#_Toc57236798)

[MÉTRICAS 8](#_Toc57236799)

[MODELO DE BASE DE DATOS TRANSACCIONAL 9](#_Toc57236800)

[DISEÑO FÍSICO DEL MODELO BI 9](#_Toc57236801)

[RUTINA DE CARGA ETL 16](#_Toc57236802)

[ETAPA STAGING AREA 16](#_Toc57236803)

[ETAPA CARGA DE DIMENSIONES 22](#_Toc57236804)

[ETAPA CARGA DE TABLA DE HECHOS 25](#_Toc57236805)

[CREACIÓN DEL CUBO OLAP 30](#_Toc57236806)

[INTRODUCCIÓN A LOS CUBOS OLAP 30](#_Toc57236807)

[PASOS PARA CONSTRUIR EL CUBO OLAP DUOC GOURMET 30](#_Toc57236808)

[CREACIÓN DE KPI INGRESOS POR VENTA 36](#_Toc57236809)

[CONSULTAS Y ANALISIS POR MEDIO DEL EXPLORADOR DEL CUBO 38](#_Toc57236810)

[CONCLUSIÓN 41](#_Toc57236811)

# ÍNDICE

# INTRODUCCIÓN

En esta tercera entrega el enfoque estará centrado en la creación del cubo OLAP del modelo dimensional del caso que tiene relación con la empresa el Restaurante DUOC Gourmet. En el informe se despliegan los requerimientos que se solicitan para la tercera fase del proyecto, el cual consisten en los siguientes puntos:

Los pasos del modelo dimensional son:

* Contexto del Caso que explica el problema de DUOC Gourmet.
* El Diseño conceptual y construcción del modelo BI físico en la base de datos.
* La rutina de la carga ETL de los datos fuentes que se cargaran a la base de datos junto con las dimensiones y la tabla BI.
* La creación del cubo OLAP del modelo explicado por pasos secuenciales

Finalmente se termina el documento con una breve conclusión sobre el proceso que se llevó a cabo para esta entrega.

# CONTEXTO

## PROBLEMA DEL CASO DUOC GOURMET

DUOC\_Gourmet es una aplicación especialmente diseñada para restaurantes de talla mediana a grande que aborda los temas de atención al público tanto en venta directa como en atención a mesas de compras.

La organización solicita desarrollar un datamart que puede capturar datos necesarios que consideren los siguientes puntos:

•Visualización gráfica de la distribución y número de las mesas en el restaurante.

•Selección de acompañamientos para aquellos platos que así requieran.

•Manejo de diferentes tipos de pago que pueden definir o no propina por garzón: Efectivo, Cheque, Tarjeta de Crédito, Cheque de Restaurante, Convenios, etc.

•Pago de cuentas.

•Descuentos

•Ventas asociadas a mesas individuales por garzón.

•Platos más vendidos

•Platos que más ganancia da

•Garzón que más vende al día, semana, mes

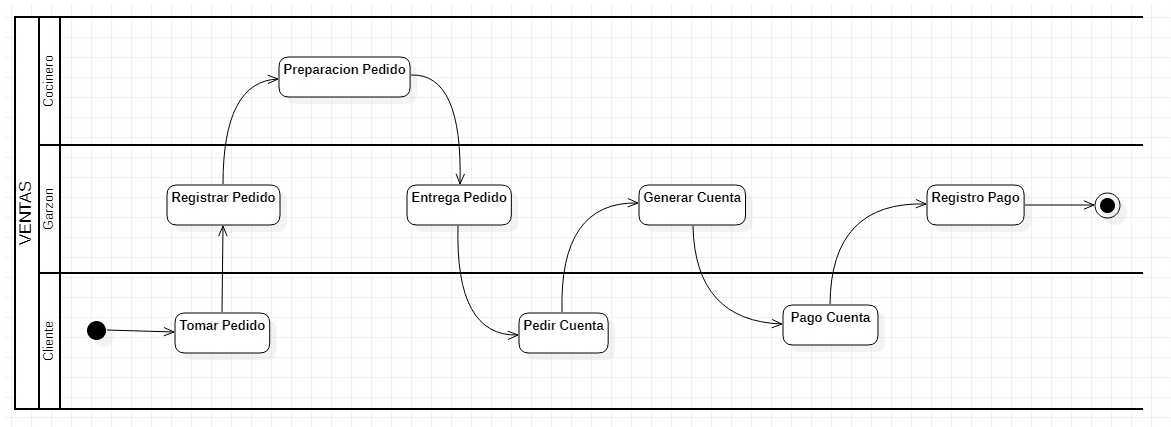
•Mesa que mas Vende

# DISEÑO Y CONSTRUCCION DE MODELO BI

## MODELO DE PROCESOS DE VENTA

La siguiente imagen ilustra el diagrama de actividad del proceso de negocio de ventas del restaurant DUOC Gourmet. En este, podemos observar que en el proceso existen 3 actores los cuales son:

* Cliente: Actor que gatilla el proceso de ventas puesto es quien es el que solicita el servicio que brinda el restaurante. Las acciones que realiza son, tomar pedidos, pedir cuenta y pago de cuenta.
* Garzón: Actor cuyo rol es principalmente atender a la clientela que llega al restaurante, entre otras labores más. En el proceso su función es, registrar el pedido de los clientes, entregar el o los pedidos, generar la cuenta a pagar por el cliente y registrar el pago de esta misma.
* Cocinero: Actor el cual su rol en el proceso es la preparación de los platos y alimentos que se registran en cada pedido. Su acción es la preparación del pedido propiamente tal.



Como se puede observar, el proceso comienza con la toma de un pedido por parte del cliente. El garzón toma el pedido y lo registra al sistema para que los cocineros se enteren de la solicitud y así comenzar a preparar los platos. Una vez listos los platos, el garzón hace la entrega a la mesa correspondiente. Eventualmente los clientes pedirán la cuenta total de los servicios entregados y el garzón se encargará de generar la cuenta que debe ser cancelada. El cliente confirma el total adeudado e informa el medio de pago. También debe notificar si desea aplicar un descuento en caso de tenerlo. El garzón confirma el pago y lo registra al sistema con todos los datos correspondiente al servicio entregado.

## 

## REGLAS DE NEGOCIO

Cabe mencionar que el gerente del restaurante nos indicó que como regla de negocios, los clientes pueden aplicar un máximo de 3 tipos de descuento por cada cuenta que deben pagar. Además, se pueden realizar solo 3 tipos de pago en una cuenta. El pago puede ser: en efectivo, Cheque, Tarjeta de Crédito, Cheque de Restaurante, Convenios y Cheques.

## NIVEL DE GRANULARIDAD

Para el nivel de detalle que se necesita de los datos que se van a capturar en el modelo BI se determinó que sean los más detallados posibles, ya sea a nivel de día por venta, por personal individual, por establecimiento y por unidad o producto que se vendió. El dato debe filtrarse de los registros que existen en las fuentes, puesto que cada transacción que se guardará será por producto vendido en el proceso de negocio al que apuntamos. La granularidad se define en los siguientes puntos:

* Ventas al día, mes
* Ventas en $$ por garzón
* Ventas en $$ por local
* Unidades vendidas
* Tiempo (fecha\_pago, hora\_pedido)

TABLA DE HECHOS

Ya que se consideró el proceso de negocios de las ventas de DUOC Gourmet, se determinó que la información que se desea almacenar para los análisis debe tener relación directa con esta actividad. Por lo tanto, para la tabla de hechos se obtiene las ventas de los platos que se concretan en el proceso mencionado. Sus atributos están conformados por las llaves foráneas de las dimensiones y por medidas que se determinaron para su análisis.

* Ventas:
  + Id\_FactVentas (PK)
  + Id\_DimPlato (FK)
  + Id\_DimMesa (FK)
  + Id\_DimGarzon (FK)
  + Id\_DimTipoPago (FK)
  + Id\_DimCliente (FK)
  + Id\_DimFecha (FK)
  + precio\_plato
  + cantidad\_plato
  + propina
  + descuento
  + total\_neto
  + IVA
  + Total\_liquido
  + total\_descuento
  + monto\_tipo\_pago
  + fecha\_pago

## DIMENSIONES

* Mesa: Dimensión que describe información sobre las mesas que dispone el restaurante. Cada mesa tiene una capacidad de personas y una ubicación dentro del establecimiento. Está relacionada con la dimensión Local, puesto que por cada local hay un número determinado de mesas. Sus Atributos son:
  + Id\_DimMesa (PK)
  + capacidad
  + descripcion\_ubicacion
  + Id\_DimLocal (FK)
* Local: Dimensión sobre la información del local. contiene la dirección y la capacidad total de las personas. Está relacionada con la dimensión Mesa: Atributos:
  + Id\_DimLocal
  + direccion\_local
  + jefe\_loca
  + cantidad\_personall
  + capacidad\_local
  + comuna
* Plato: El plato que se eligió en la orden según el menú que entrega el restaurante. Cabe mencionar que los platos tienen una categoría y según la categoría se determina si viene con acompañamiento o no. De esta manera se podrá tener conocimiento del tipo de plato que se registró según su categoría, ya sea uno con acompañamiento o uno sin él. La categoría de desglosa en otra tabla la cual está relacionada. Sus Atributos:
  + Id\_DimPlato (PK)
  + nombre\_plato
  + descripcion\_plato
  + precio\_plato
  + Id\_DimCategoria (FK)
* Categoría plato: La categoría del plato. Este puede ser: con acompañamiento, vegetariano, exóticos, parrillas, etc. Atributos:
  + Id\_DimCategoria (PK)
  + nombre\_categoria
  + descripcion\_categoria
* Garzón: El empleado que se encarga de atender a los clientes. Recibe una propina que se carga en la boleta del servicio. Se añade un historial de código para cada garzón con el motivo de poder controlar campos que cambian lentamente (SCD). Atributos:
  + Id\_DimGarzon (PK)
  + nombre\_garzon
  + rut
  + Id\_DimGarzonOld
* Tipo Pago: Para que la empresa pueda tener una idea de cómo se comporta el cliente al momento del pago se creó la dimensión Tipo de Pago el cual contiene los atributos necesarios para los análisis requeridos. Sus atributos son:
  + Id\_DimTipoPago (PK)
  + nombres\_tipo\_pago
  + descripcion
* Fecha: Esta dimensión contiene la información de tiempo, puesto que para todo tipo de modelo BI es necesario disponer de una dimensión que determine el tiempo. La llave primaria se conforma por la conversión del tiempo que se registra en la transacción a un número entero, en donde el orden de los números denotan el año, mes, dia, hora. Sus atributos son:
  + Id\_DimFecha (PK)
  + hora
  + dia
  + mes
  + año
  + fecha
* Encargado de local: son las Personas que tienen a cargo la administración del local llevando a cabo trabajos administrativos
  + id\_encargado
  + nombres
  + apellido\_paterno
  + apellido\_materno
  + local\_id\_local
* Multi Pago: esta es una tabla de quiebre que recibe diversos pagos de una misma venta
  + id\_multipago
  + fact\_venta\_id\_numero\_boleta
  + tipo\_pago\_id\_tipopago
* Tipo Pago: tabla que contendrá diferente tipos de pagos
  + id\_tipopago
  + nombre\_pago
* Multi Descuento: tabla de quiebre donde se reciben diferentes tipos de descuentos de una u otras mesas
  + id\_multidesc
  + fac\_venta\_id\_numero\_boleta
  + descuento\_id\_descuento
* Descuento: tabla donde se almacena los distintos tipos de descuentos
  + id\_descuento
  + nombre\_descuento

## MÉTRICAS

Para las métricas se tomó en consideración el nivel de granularidad de los datos que se deben capturar en las transacciones. Las medidas están enfocadas principalmente a las ventas que se consuman tanto por plato, mesa, cliente, local, las propinas del garzón, entre otros. Con estas mediciones se puede tener una visión detallada de las ventas que se realizan en un periodo determinado. Por ejemplo, si se quiere saber las ganancias que se adquirieron en la venta de un plato por día se puede concluir cuál es el producto que más se vende del menú dentro de un periodo de tiempo en específico.

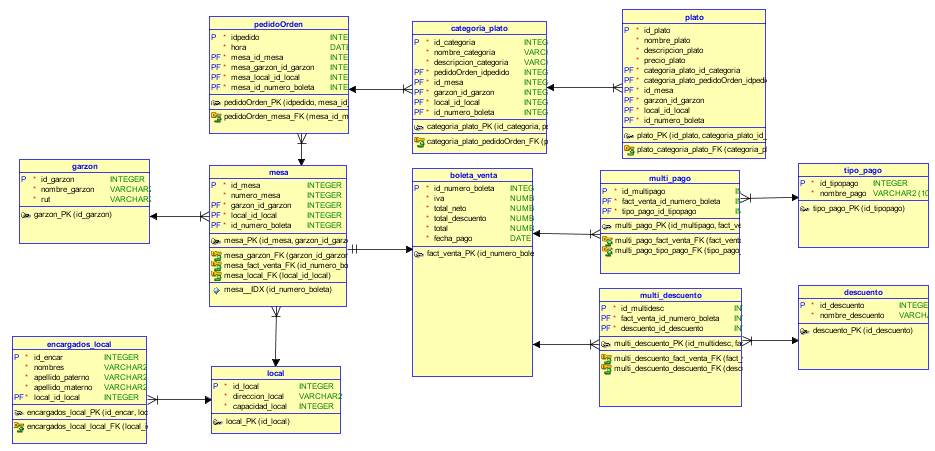
Las métricas que se identificaron se clasifican en las siguientes:

* Ventas, monto total al día en $$ por local
* Ventas monto total por plato vendido según categoría (acompañado o no) en $$ al dia o mes.
* Ventas, monto total de propinas al día en $$ por garzón
* Ventas, monto total mensual en $$ por local
* Ventas, monto total de propinas mensual en $$ por garzón
* Ventas, monto total al día que se le aplicó un descuento
* Ventas, monto total al día por concepto de tipo de pago
* Cantidad de unidades vendidas por local al día y mes.
* Tiempo, día y hora con más ventas en el local.



## MODELO DE BASE DE DATOS TRANSACCIONAL

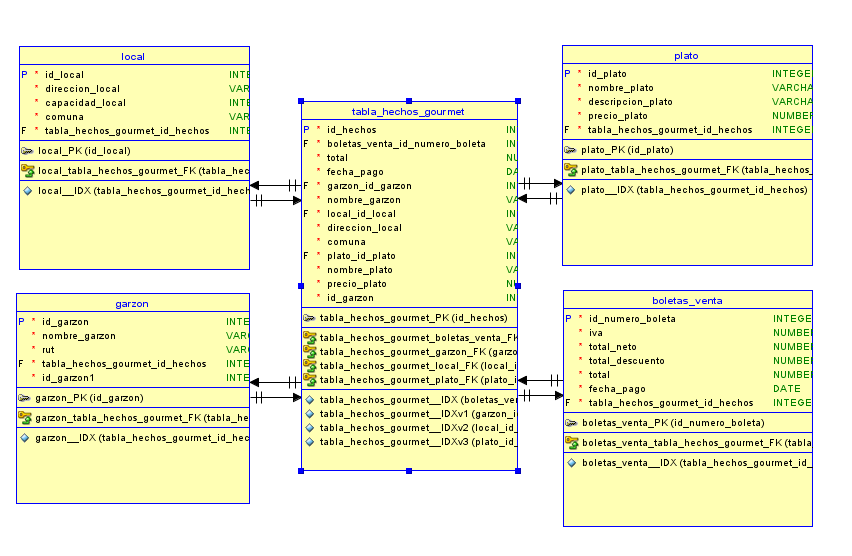
En este diagrama se ilustra el modelo de base de datos transaccional que dispone la organización DUOC Gourmet. En él se puede observar que existen 12 tablas en total y la que se encarga de registrar las transacciones es la tabla boleta\_venta dentro del modelo. Según lo detectado se puede concluir que boleta\_venta debe estar apuntando al proceso de negocios de las ventas del restaurante.



## DISEÑO FÍSICO DEL MODELO BI

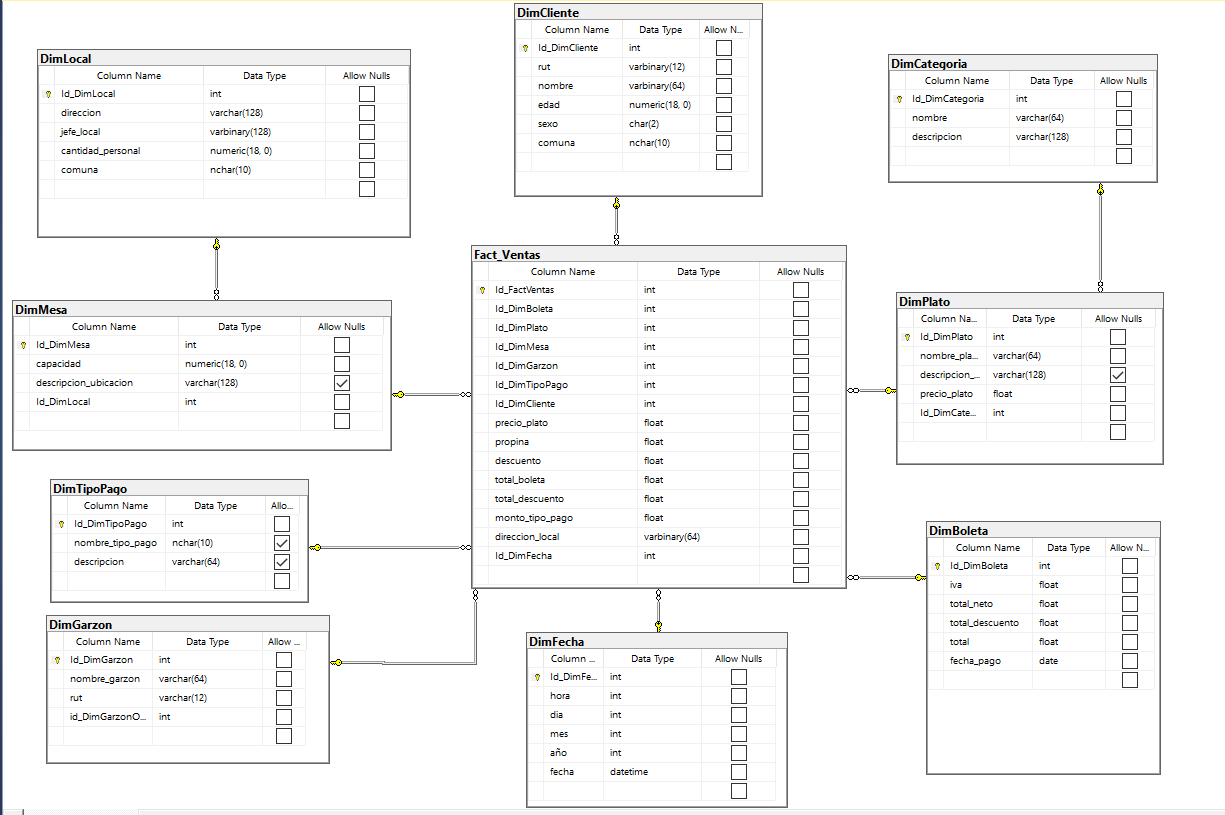
En la confección del diseño físico se realizaron 3 iteraciones. La primera se desarrolló y consultó con el docente en clases; como resultado se llegó a un segundo resultado con un modelo de copo de nieve el cual está más detallado.

* Primera iteración



En primera instancia se realizó un esquema tipo de estrella. Se pensó en esto por su funcionalidad de poder agrupar la información que se necesita como ventas total diarias y por mes, locales que han vendido, ciertos tipos de platos, los mejores garzones, etc.

* Segunda iteración

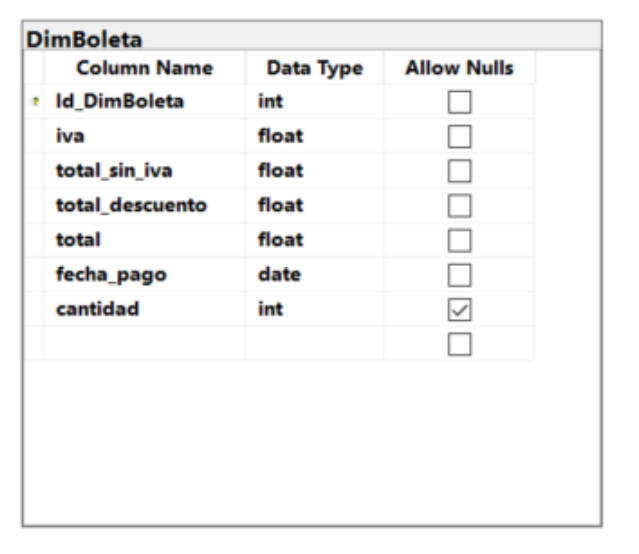


En la segunda iteración en el desarrollo del modelo BI se consideraron datos adicionales que no estaban en el modelo anterior. Esencialmente, este resultado consiste en un detalle de cada producto que se transa en el restaurante, en este caso estamos hablando de un producto en representación de un plato. Como se mencionó anteriormente, el plato se distingue de la categoría y con este dato podemos distinguir si se trata de uno con o sin acompañamiento.

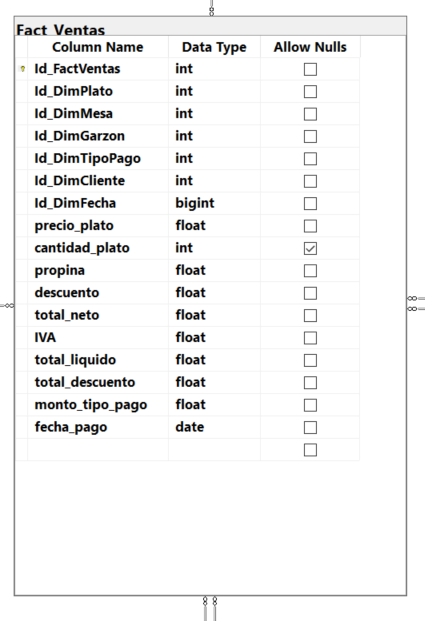
Por otro lado. Se incluye la dimensión boleta puesto que por cada transacción que se registre en la tabla de hechos, es necesario incluir la información de la venta en sí misma. ¿Por qué? Bueno el motivo es que si la tabla de hechos se registra por plato individual puede suceder que un mismo cliente pague por toda la mesa que se consumió. Esto puede generar confusión en el análisis de los datos. Es por esta razón que se integró al modelo la información de la boleta para poder filtrar los datos por orden que se realizó con sus respectivos datos, por ejemplo, el descuento en la boleta.

* Tercera iteración

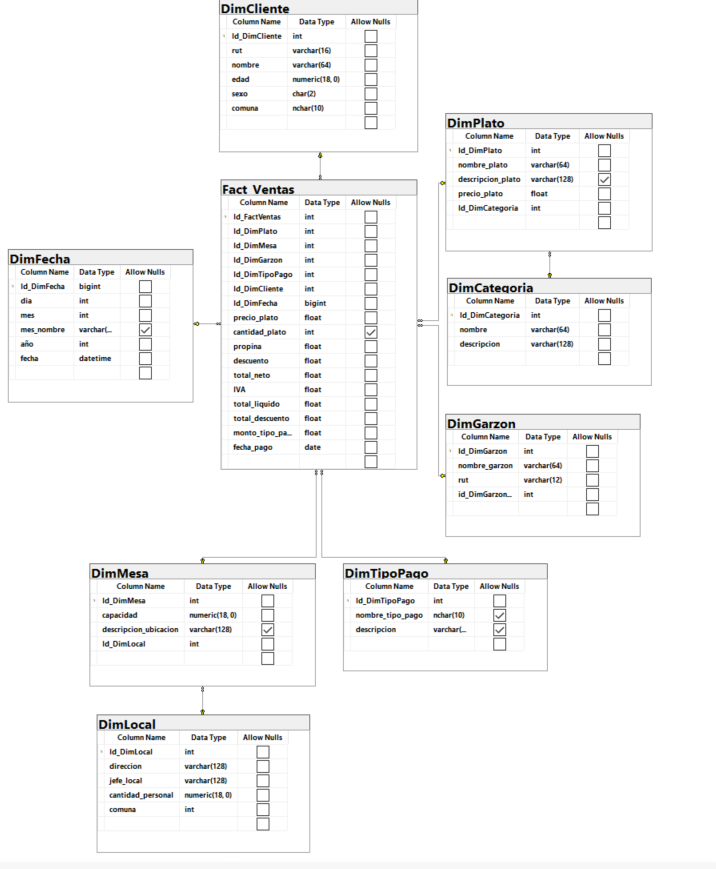
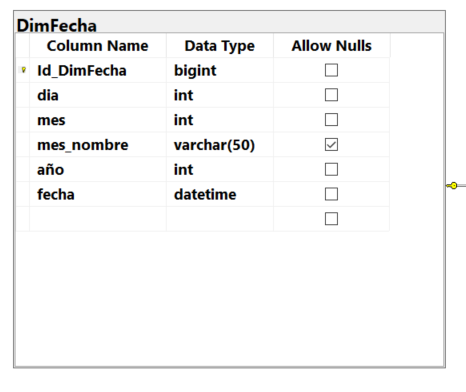
Para la tercera iteración del diseño se descartó la dimensión Boleta del modelo BI puesto que todas las transacciones de las ventas se registran en la tabla de hechos, por lo tanto, era una dimensión que estaba siendo redundante en el modelo.



Para solucionar el problema de la venta de distintos tipos de productos en una misma boleta se determinó que cada transacción debe ser distinguida por producto y por la boleta, esto quiere decir que la llave primaria de la Fact Table ventas estaría compuesta por 2 datos: la primera es el id de la boleta y la segunda el id del plato que se pagó. En cuanto a la información que tenía la dimensión boleta, se traspasaron la mayoría de los campos a la tabla de hechos con tal de colocar las métricas donde corresponden en el modelo. La Fact Table quedo de la siguiente manera:



Por otro lado, se modificó la dimensión fecha, eliminando el campo de hora. La razón de esto se debe a que el nivel de granularidad para el tiempo seria las ventas por día. Además, se añadió un campo que tuviese el nombre del mes para que las lecturas de las consultas fuesen más legibles y entendibles. Así es como quedo la dimensión DimFecha.



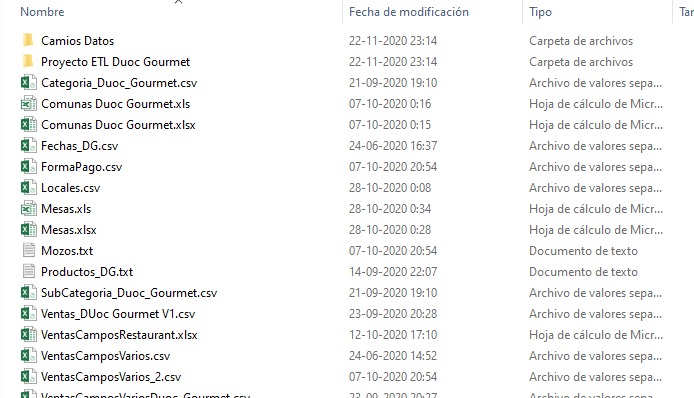
El resultado final del modelo BI consiste en uno de copo de nieve el cual asegura disponer los datos con mayor detalle a la hora de realizar el análisis de los datos registrados.

# RUTINA DE CARGA ETL

En este apartado se mostrará la construcción e implementación del proceso de extracción, transformación y carga de datos (ETL) para poblar el modelo dimensional del proyecto BI DUOC Gourmet.

En este proceso se hizo una recopilación de todas las fuentes de información que teníamos disponible para luego cargar los datos extraídos en un área de etapa de carga (Staging area) y desde allí comenzar a crear las dimensiones junto con la tabla de hechos, todo realizado de forma secuencial.

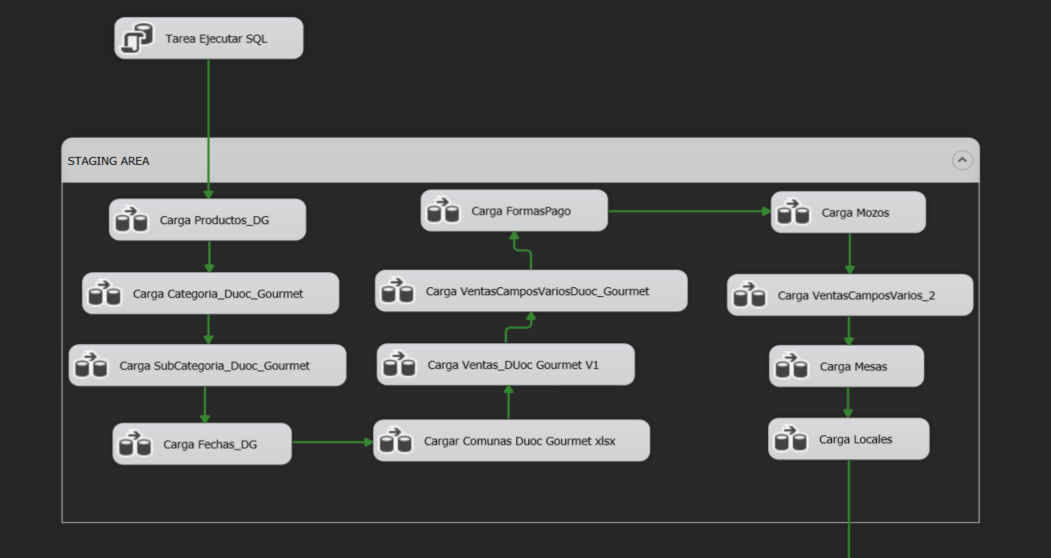
Las fuentes de los datos son extracciones de información de bases de datos transaccionales convertidos en archivos de texto plano, planillas de Excel o simplemente extraídas directamente desde bases de datos de corporaciones.



En la imagen de arriba se puede ver los distintos tipos de archivos que se recopilaron para la carga del área Staging de los procesos ETL.

## ETAPA STAGING AREA

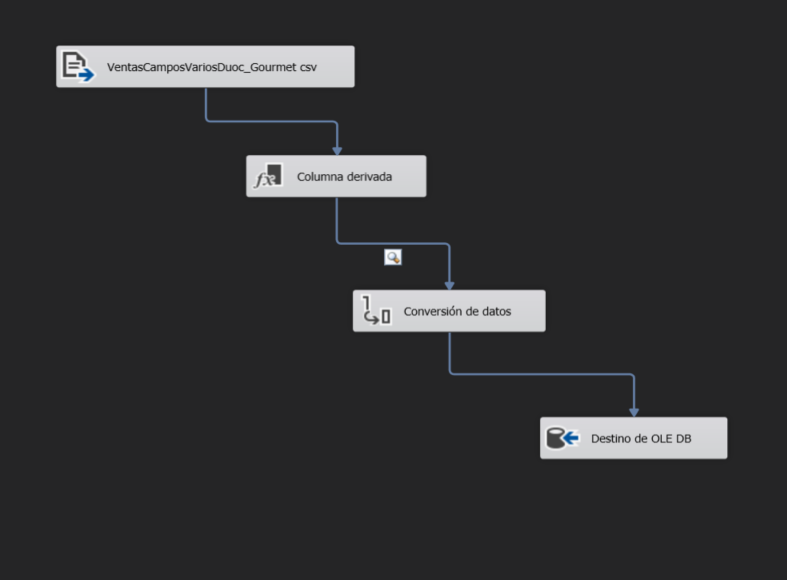
Esta etapa consiste en cargar todos los archivos fuentes, mencionados anteriormente, a una base de datos determinada. Dentro de esta base de datos se encuentra una réplica de la información de la fuente que es necesaria para alimentar el modelo BI, por lo tanto esta se encuentra ordenada, con formato correspondiente y filtrada.



Aquí se puede ver las rutinas de carga ETL para el área Staging dentro del proceso. Si bien, las tareas se van ejecutando de manera secuencial ya que es importante que los procesos se vayan ejecutando uno por uno y asi evitar posibles complicaciones que surjan al cargar múltiples tareas en paralelo.

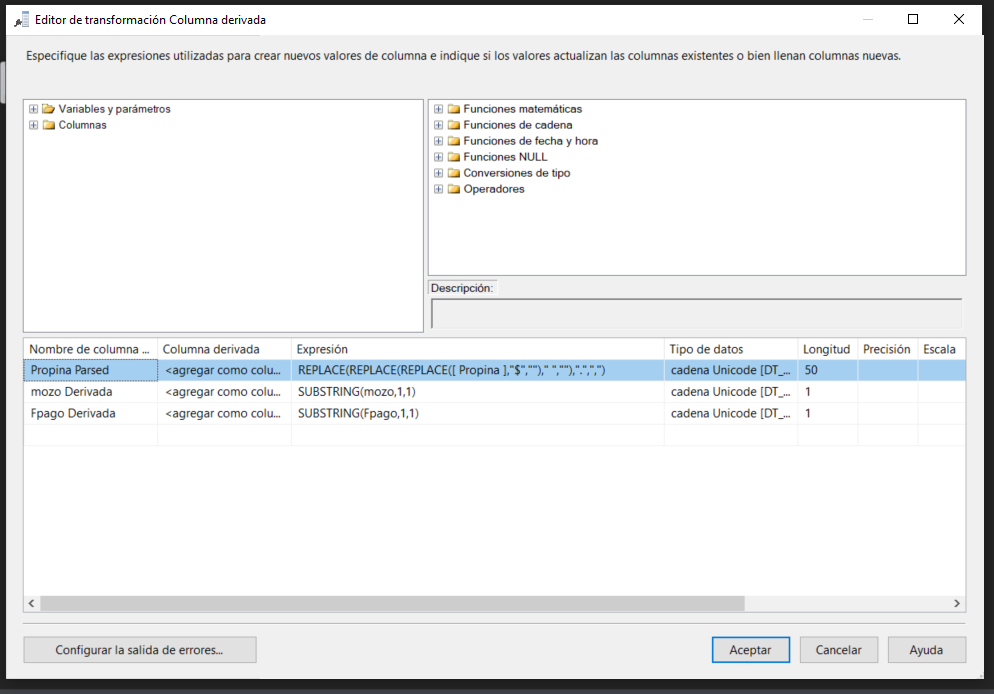
Dentro del recuadro STAGING\_AREA existen una serie de elementos que van conectados unos con otros. Cada uno representa la carga de un archivo fuente y dentro están todas las operaciones necesarias para convertir los datos con los formatos y tipos correspondientes.

Como ejemplo, en la siguiente imagen se ve la tarea de flujo de datos para el archivo *VentasCamposVariosDuoc\_Gourmet.csv.*

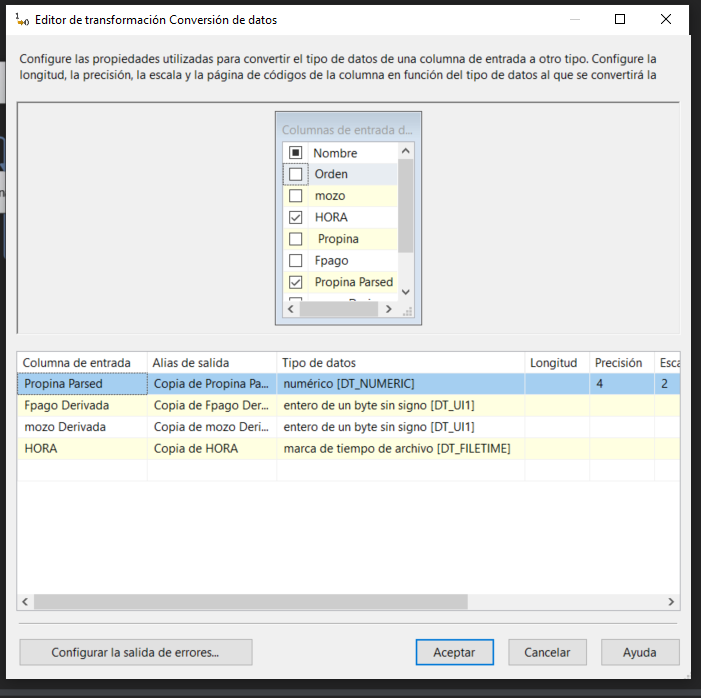


La seria de operaciones del diagrama se configuran de la siguiente manera:

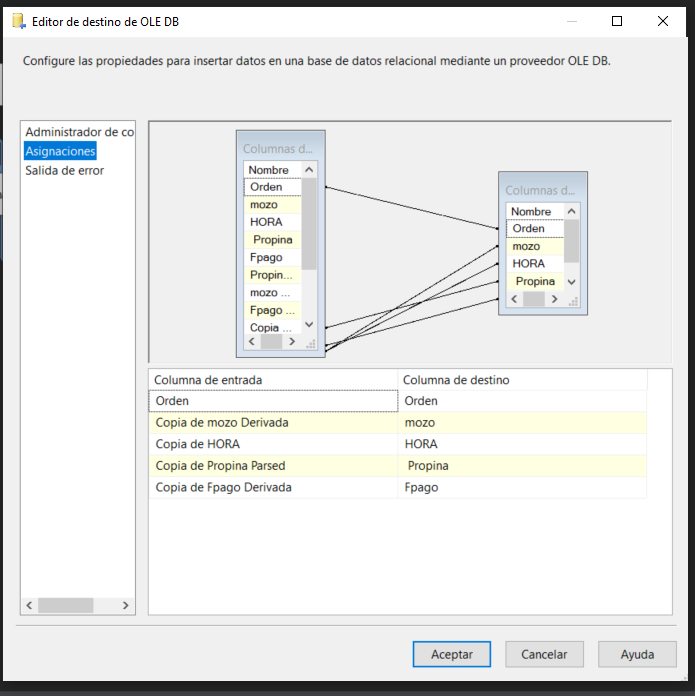
* Primero buscar la fuente de la información
* Luego se crea un campo con un formato especifico en base a un dato que se extrae de la fuente. Se hace a través de la operación *columna derivada.*



* Se convierten los datos extraídos a un tipo de dato correspondiente al campo de la tabla que se destina. Para esto se usa la operación “Conversión de datos”

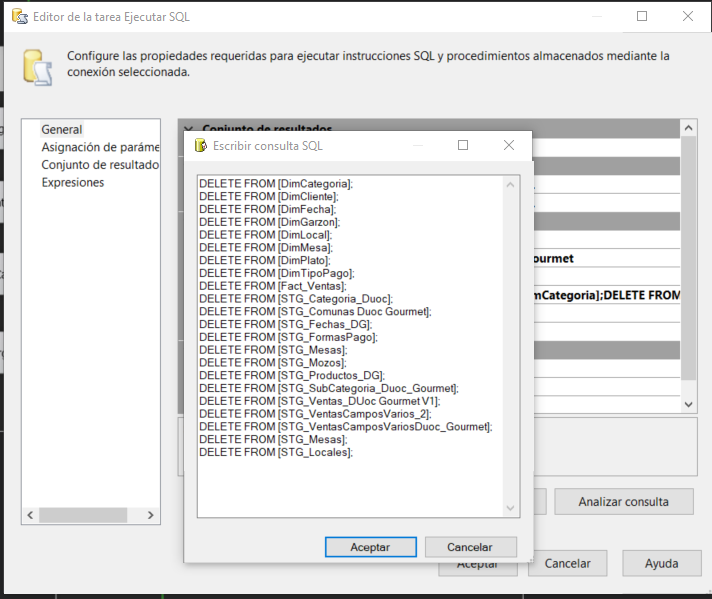


* Finalizando la tarea con la carga de la información necesaria a la tabla que está dedicada para almacenar los datos del archivo fuente.



Para cada archivo que se carga el ETL existe una tabla en la base de datos con su respectivo nombre y siguiendo la nomenclatura: *STG\_nombre\_archivo.*

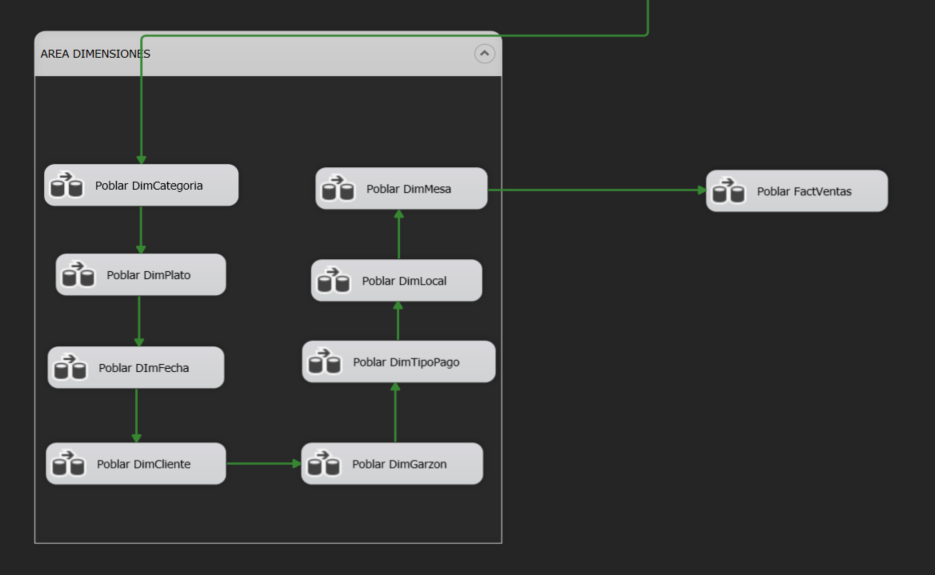
Cabe mencionar que al comienzo se ejecuta una sentencia SQL que asegure que las tablas que se van a cargar a la base de datos estén completamente vacías. Es necesario hacerlo debido a que el modelo BI tiene dependencia de llaves foráneas y así evita que haya inserciones de datos con la misma llave primara.



## ETAPA CARGA DE DIMENSIONES

Esta etapa comienza una vez que se hayan cargado todos los archivos fuentes al STAGING AREA con tal de tener una versión limpia de los datos para poder ser cargados a las dimensiones.

A continuación, se observa el diagrama del área de carga de las dimensiones en el ETL.



Al igual que en el STAGING AREA cada elemento que está en el recuadro corresponde a la carga de datos de cada dimensión que existe en el modelo BI, por lo tanto, cada uno de ellos consiste en las tareas necesarias para el procesado y extracción de información que va destinada a cada dimensión.

Como ejemplo se visualizarán las tareas de carga para las dimensiones de fecha y platos:

Flujo de datos para DimFecha:



Flujo de datos DimPlatos:



Como se puede observar en las 2 imágenes anteriores se realiza un flujo de proceso de datos de manera secuencial. La fuente para estas tareas ya no son los archivos fuentes que se utilizaron, sino que se usan las tablas que se generaron en la etapa anterior. La gracia de tener estas tablas disponibles es que los datos fuente están centralizados y ordenados en un solo lugar, ayudando a tener un proceso ETL más eficiente y ordenado.

Sin embargo, igual es necesario realizar ciertos cambios en los formatos de datos de origen para que la información que salga de estos procesos este con los tipos de datos correspondientes a los que las dimensiones contienen. También se hace necesario unir cierta información con otra para que los datos que se ingresen sean consistentes a la información que se necesita. Más adelante se explicará cómo se soluciona este problema.

## ETAPA CARGA DE TABLA DE HECHOS

Finalmente, el proceso ETL finaliza con el poblado de la tabla de hechos, la más importante de todas. En esta etapa se unen todas las dimensiones cargadas anteriormente al vincular sus llaves primarias con las llaves foráneas que conforman gran parte de la tabla de hechos. Adicionalmente se agregan todas las métricas que se definieron en la etapa de diseño del modelo. Estas métricas se llenan con las tablas del STAGING AREA principalmente con la tabla que representa las transacciones en el esquema transaccional en cual está basado el modelo BI.



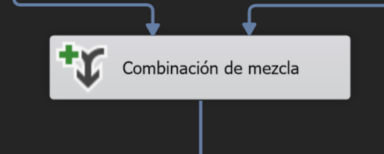
Esta etapa está separada del resto y consiste una sola tarea de flujo de datos debido a que el proceso representa la carga de datos de la tabla de hechos FactVentas.

Al abrir la tarea se puede observar lo siguiente:

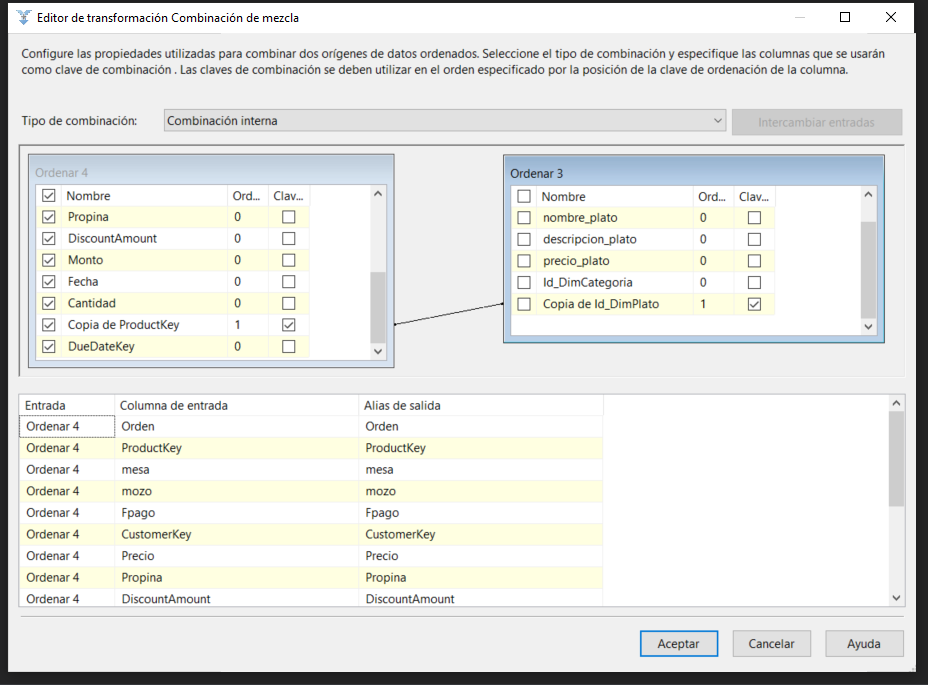


Si bien, este proceso es el más complicado de todo puesto que se necesita hacer una serie de cruces de información con tal que se unifique todas las tablas de datos en una sola. La salida de los datos debe ser lo más consistente posible y evitar posibles fugas de transacciones por errores o mal procesado en la integración de los datos.

Para unificar los datos en una sola salida se utiliza la operación “*combinación de mezcla*”.



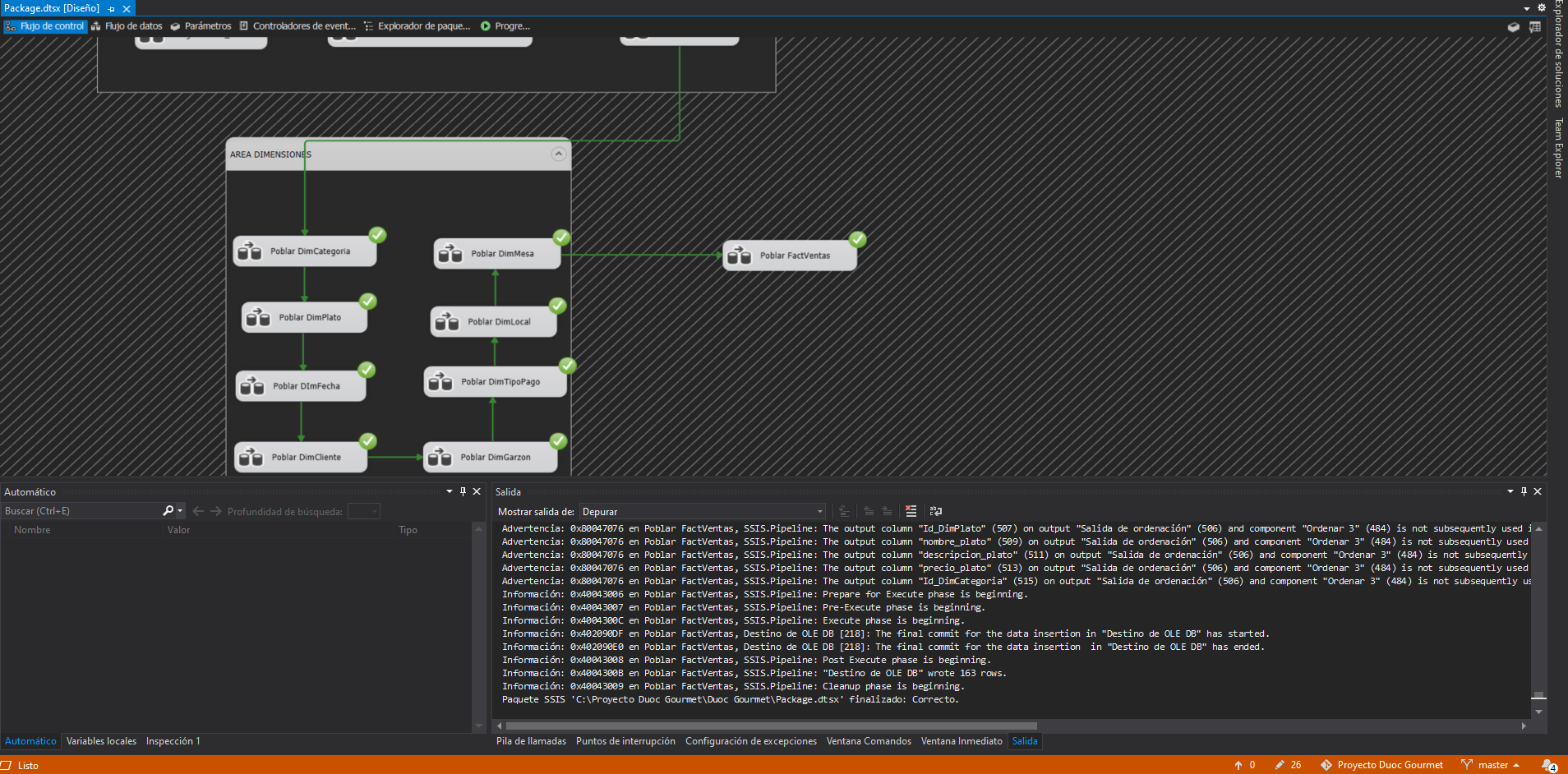
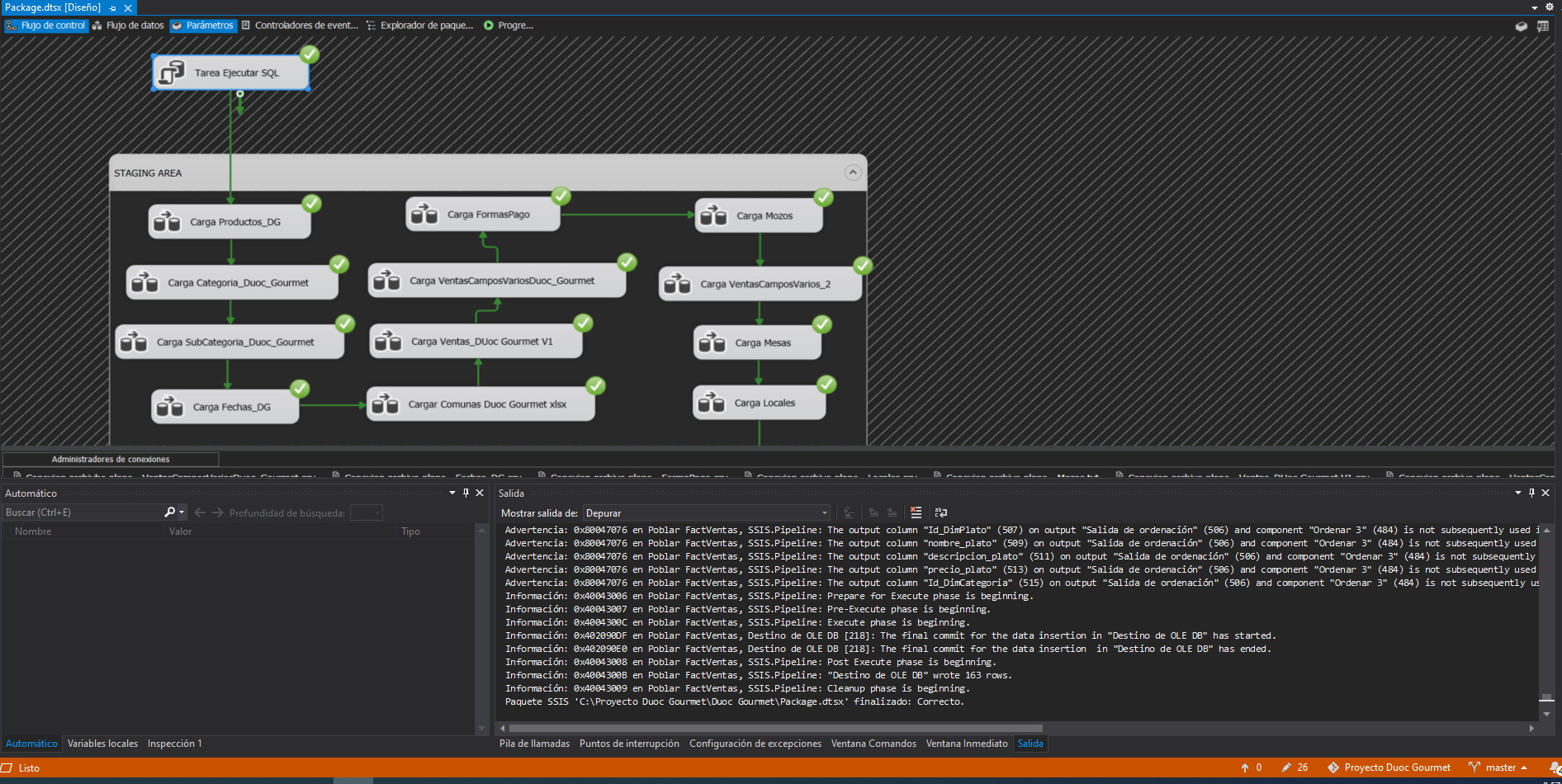
Esta operación permite hacer la unificación de datos similar a lo que sucede al tratar de unir 2 o más tablas por medio de sentencias SQL, más conocido como “*Join*”. Al hacer doble clic sobre la operación aparecerá lo siguiente:



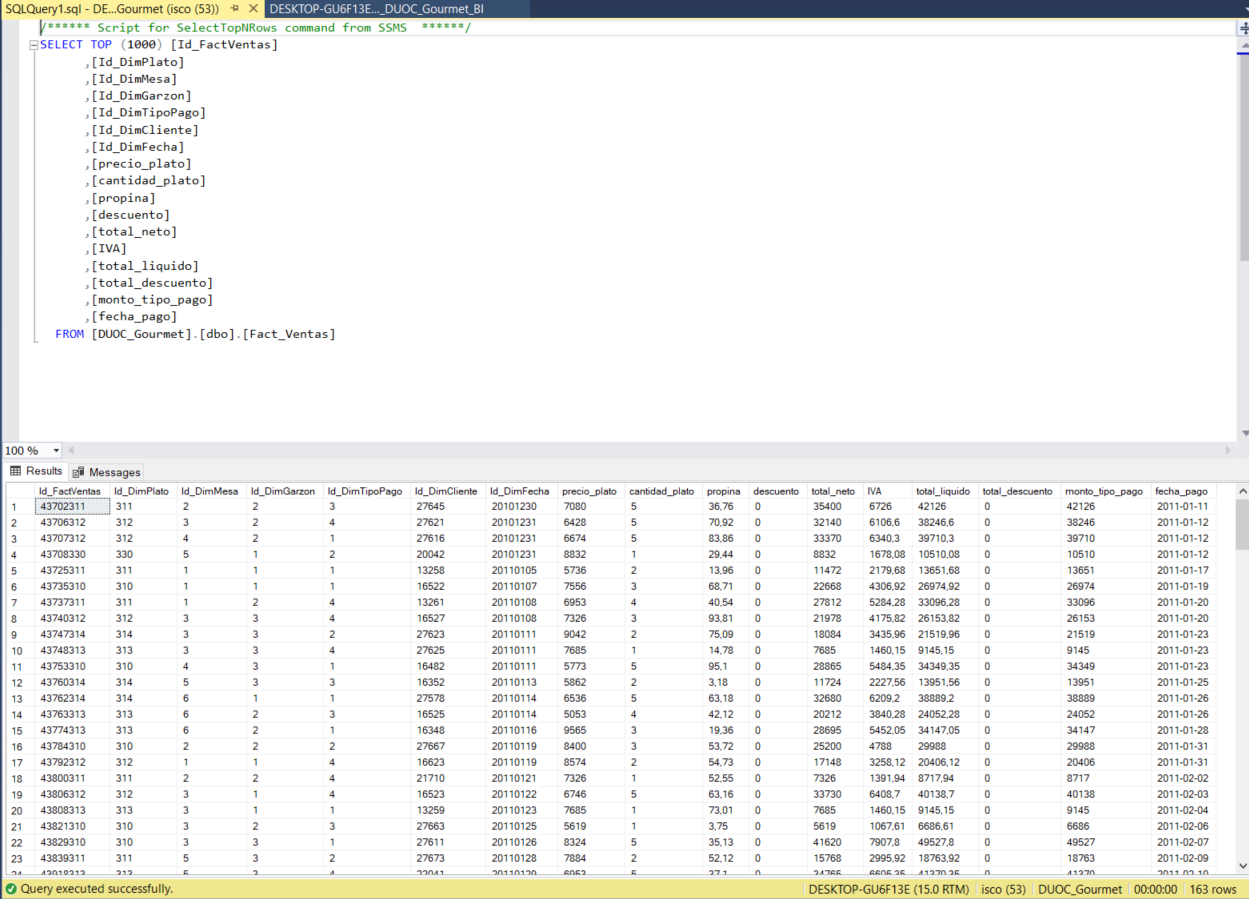
A partir de los datos se hace una conexión en base a uno o más campos que tengan relación las entidades en cuestión. De esta manera se puede recolectar y filtrar la información necesaria para cargar los campos que conforman la tabla de hechos.

Una vez generado la secuencia completa y rescatado los datos que se destinara para la tabla de hechos se ejecuta el ETL completo para que carguen toda información procesada a la base de datos del modelo.

A continuación, se ilustran todas las tares ETL procesadas correctamente:



Evidencia de la tabla de hechos FactVentas cargada en la BD del modelo BI:



# CREACIÓN DEL CUBO OLAP

## INTRODUCCIÓN A LOS CUBOS OLAP

Definición: On-line Analytical Processing “Procesamiento analítico en línea” Es una solución cuyo objetivo es agilizar la consulta de grandes cantidades de datos.

Este paso es menos complejo que el proceso del ETL en el proyecto. Simplemente consiste en construir el modelo BI de base de datos en una estructura de datos orientada al manejo de información en función a la estadística descriptiva. Esto quiere decir que las consultas se hacen a través de planillas parecidas a las que se hacen en Excel sin la necesidad de ejecutar sentencias SQL.

Por otro lado, los datos no se tratan como en una base de datos relacional, sino que cada tabla en el modelo representa una dimensión distinta como si se trátese de una matriz multidimensional. De ahí viene la denominación de cubo, puesto que se caracteriza por las dimensiones X, Y y Z que describe el objeto tridimensional. A partir de lo dicho, los datos se disponen en una matriz de elementos multidimensionales, en donde por cada eje se tiene la información de una dimensión en particular.

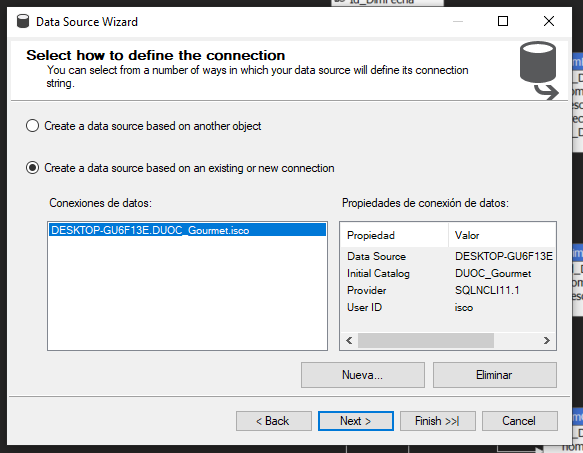
Imagen ejemplo de un cubo OLAP:



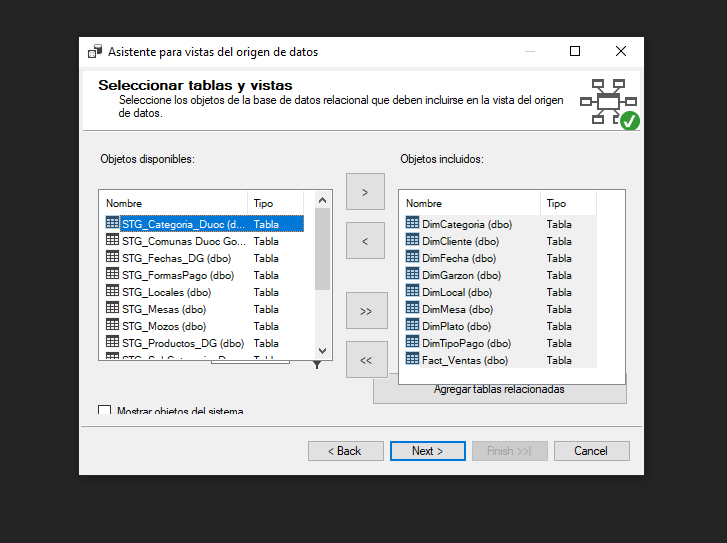
## PASOS PARA CONSTRUIR EL CUBO OLAP DUOC GOURMET

A continuación, se detallan los pasos que se siguieron para construir el cubo OLAP DUOC Gourmet.

1.- En primer lugar, se debe cargar la base de datos fuente que contiene el modelo BI con sus respetivas dimensiones y tabla de hechos cargadas.

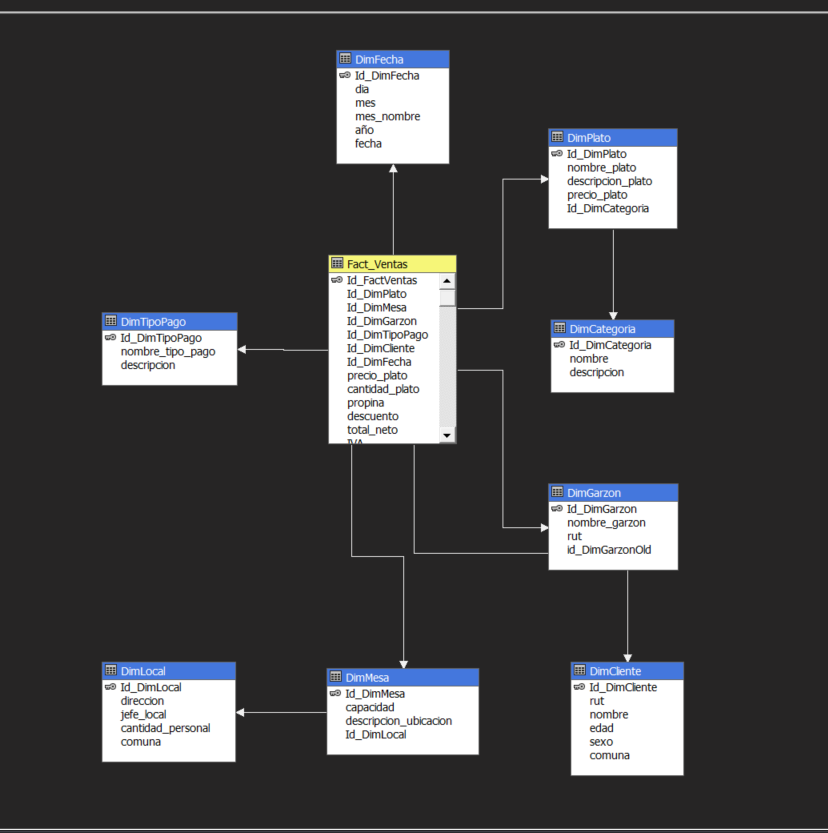


2.- Luego, se crean las vistas de origen de datos a partir del modelo BI que está en la base de datos seleccionada.

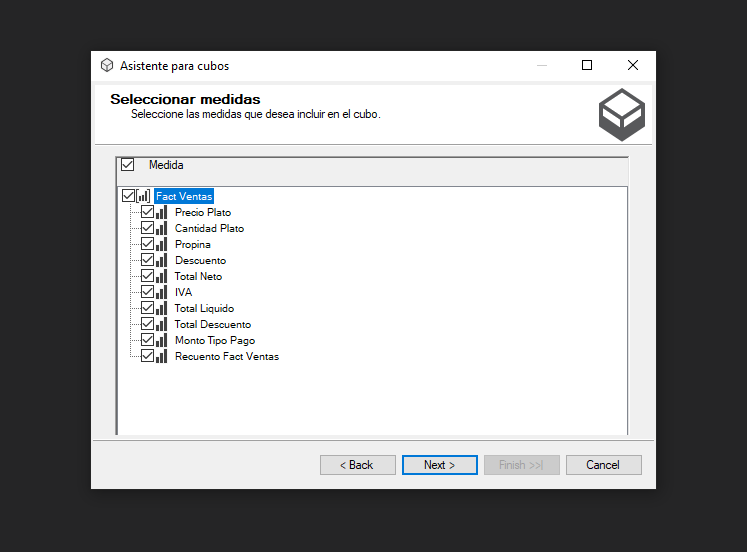


En este caso, se seleccionan solo las tablas correspondientes al modelo debido a que en la misma base de datos se encuentra el STAGING AREA de proyecto, lo cual en esta etapa no es necesario.

Como resultado, se desplegará en pantalla el diagrama completo de las vistas en base al modelo.

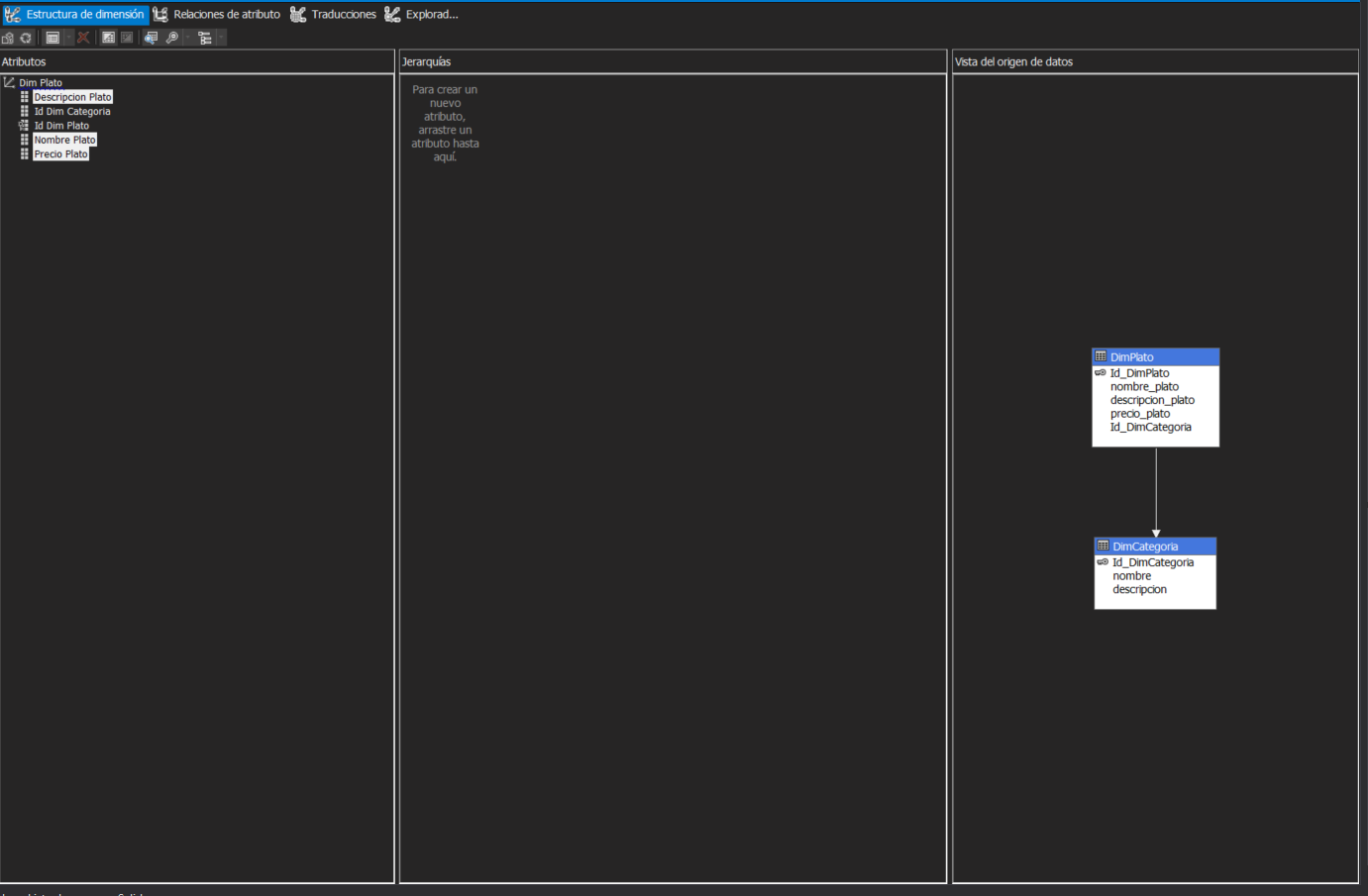


3.- Una vez creadas las vistas del modelo BI se procede a la creación del cubo propiamente tal.



El pantallazo ilustra que la herramienta Analysis Services identifica las medidas creadas en la tabla de hechos, sin tomar en cuenta las llaves foráneas. El programa sabe que las llaves se vinculan con las dimensiones del modelo, por eso que no aparecen como medidas.

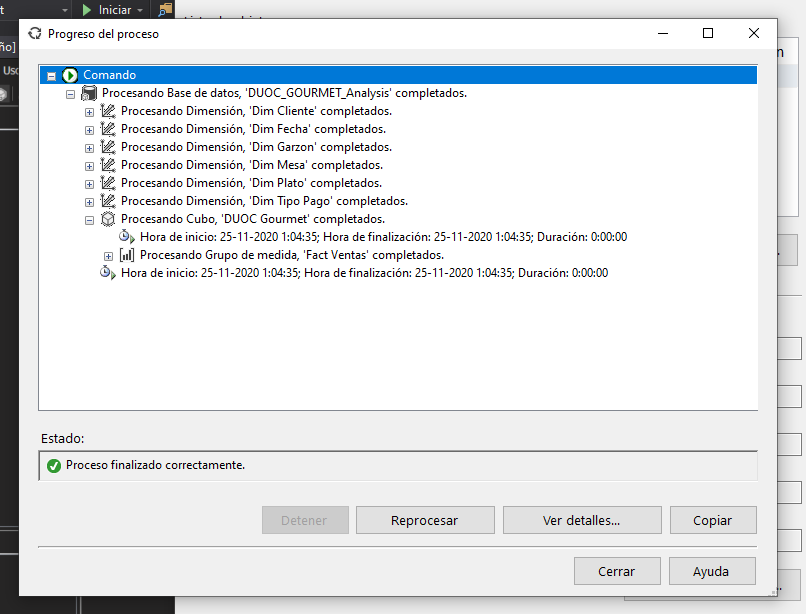
4.- Por defecto, las vistas se generan con solo los atributos de la llave primaria de cada dimensión, por lo que se hace necesario, para efecto de análisis, agregar algunos de los campos que caracteriza cada dimensión.



En la imagen se ve cómo se agregan algunos atributos de la dimensión DimPlato tales como: el nombre del plato, su descripción y precio.

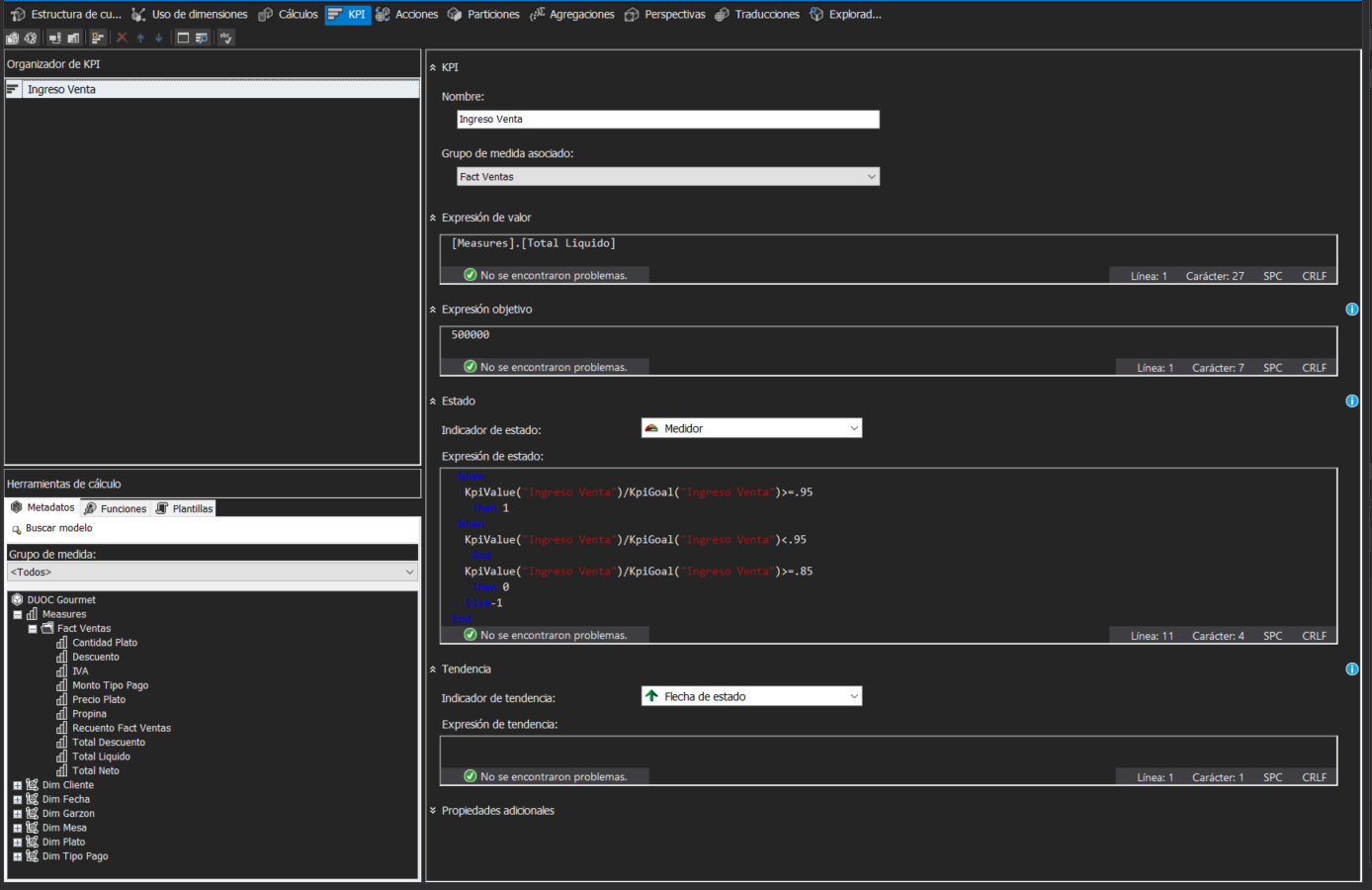
5.- Una vez completado los pasos anteriores de debe procesar el cubo por completo para que la información y configuración sea procesada por la herramienta Analisys Services.

Aquí podemos observar como el proceso del cubo fue completado exitosamente



## CREACIÓN DE KPI INGRESOS POR VENTA

Se creó un KPI que evalúa los ingresos por venta en un tiempo determinado. Para efectos de testeo se determinó que el KPI funcionaria a nivel de mes con un valor de 500.000 CPL como objetivo para las evaluaciones.



El KPI simplemente trata de evaluar las ventas por mes según un objetivo determinado, que en este caso sería en relación a un margen de 500.000 CLP. Se aplicó una lógica que evalúa el monto de cada mes y según este valor se hace un cálculo que revela si el objetivo fue cumplido o no. Por ejemplo: Si en el mes de marzo hubo un ingreso 450.000 CLP en las ventas estaríamos en el escenario de que la meta no fue cumplida; por lo tanto, mostraría un valor negativo en los indicadores del KPI.

El KPI se representa en forma de medidor en donde los valores posibles oscilan entre -1 y 1. Si la meta fue cumplida, dependiendo de qué tan cerca este del margen, el valor será entre 0 y 1; mientras que si no fue cumplida mostraría el valor -1.

En las siguientes líneas se muestra con detalle la lógica que se aplicó para controlar el KPI.

Case

When

KpiValue("Ingreso Venta")/KpiGoal("Ingreso Venta")>=.95

Then 1

When

KpiValue("Ingreso Venta")/KpiGoal("Ingreso Venta")<.95

And

KpiValue("Ingreso Venta")/KpiGoal("Ingreso Venta")>=.85

Then 0

Else-1

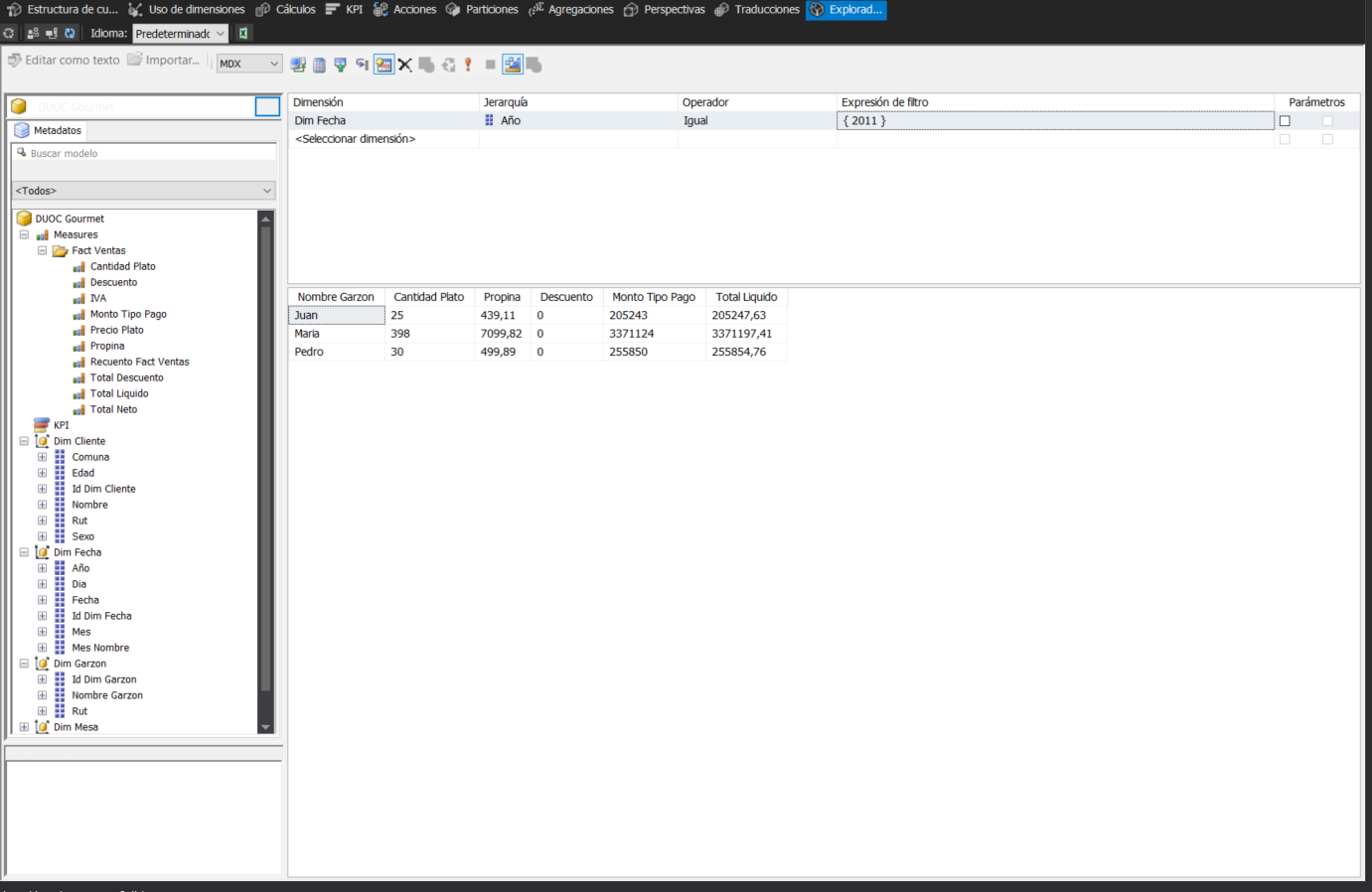
End

## CONSULTAS Y ANALISIS POR MEDIO DEL EXPLORADOR DEL CUBO

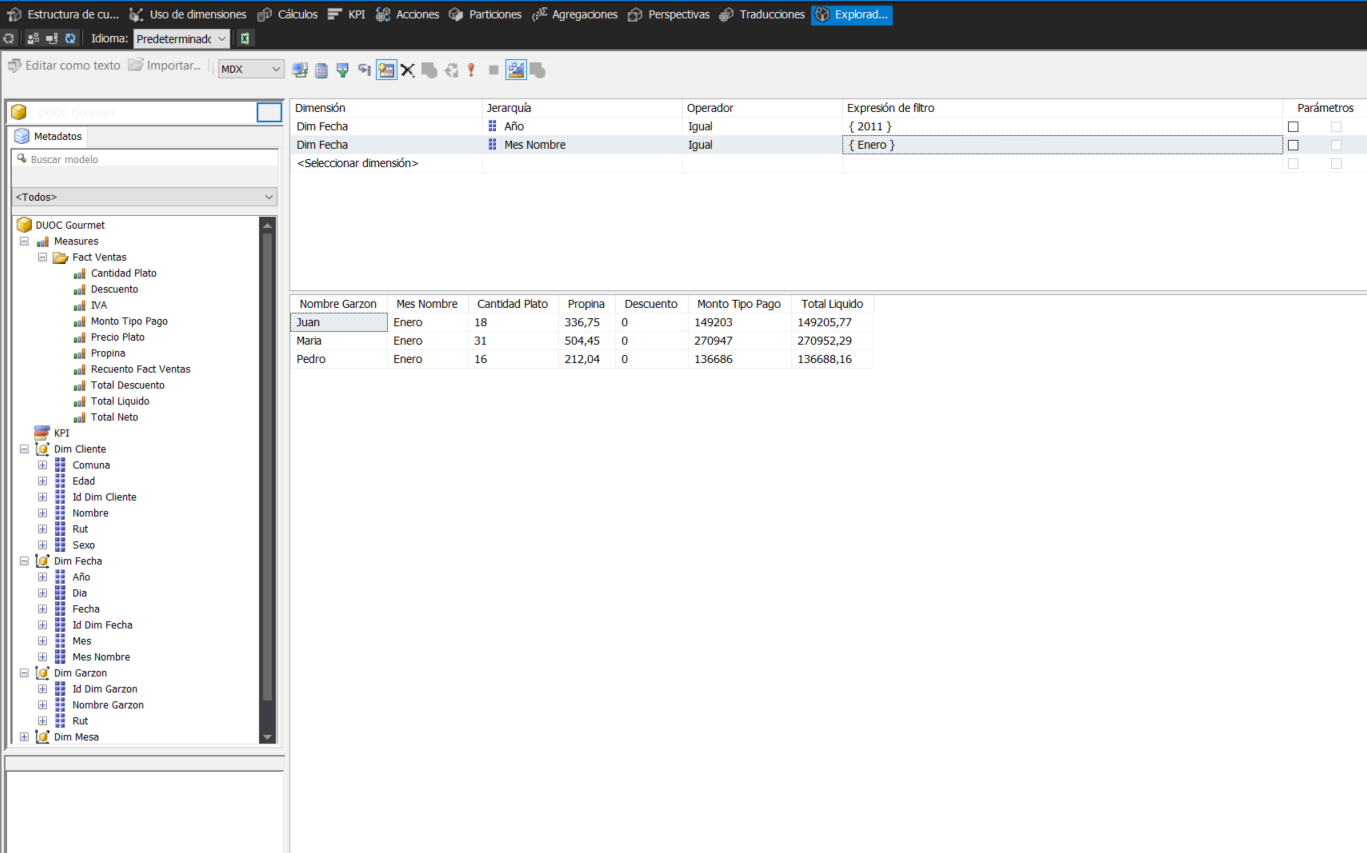
Al finalizar el cubo, la siguiente actividad seria en realizar consultas analíticas del cubo a través del explorador que ofrece Analysis Services. Este consiste en realizar un filtrado de información según una lógica que sea pertinente al negocio en cuestión. Por ejemplo, cuantos platos se vendieron en una determinada fecha, cuáles fueron las ventas por año, etc. Esto es posible gracias a que los datos del modelo, que fueron procesados en el cubo, se disponen de manera pertinente para realizar consultas analíticas de grandes cantidades de información.

Las preguntas que pudimos detectar según la necesidad del Restaurante DUOC Gourmet fueron las siguientes:

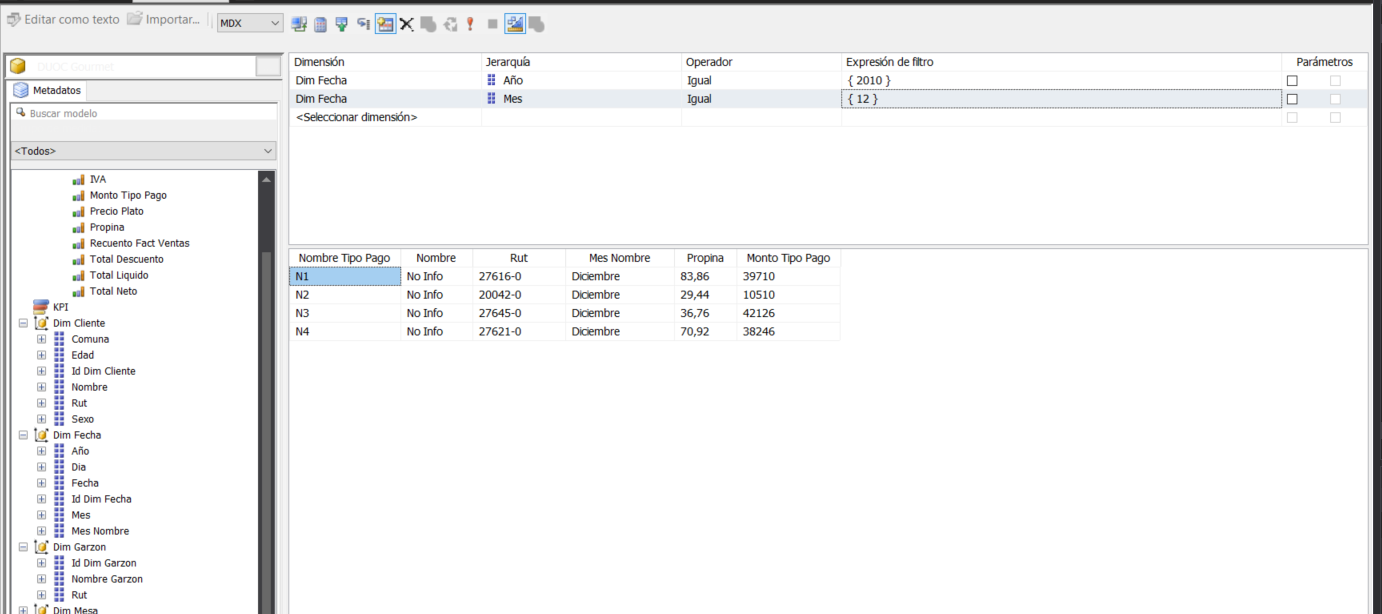
* Total de ventas en CLP por garzón en el año 2011.



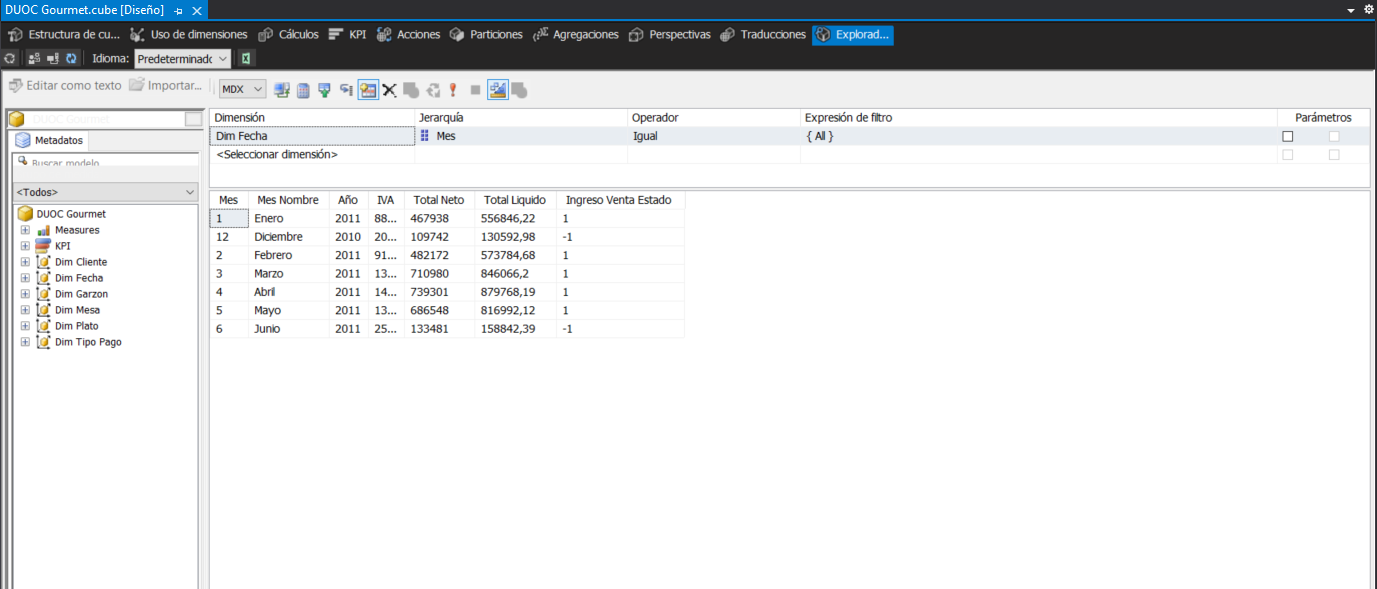
* Cantidad de ventas en CLP por garzón en el mes de enero del 2011.



* Cantidad de ventas en CLP por tipo de pago en diciembre del 2010.



* Estado en el KPI de Ingresos por mes.



# CONCLUSIÓN

Los pasos que se siguieron para desarrollar esta tercera fase del proyecto fueron:

* En primer lugar, se abordó el problema que se presentó en el documento del primer examen, analizando las evidencias que revelaba el caso y discutiendo las soluciones con el grupo de trabajo.
* Una vez comprendido el problema se diseñó el modelo de negocio que se identificó en el proceso de ventas del Restaurante DUOC Gourmet.
* Como siguiente paso se identificaron las dimensiones esenciales junto con la tabla de hechos que debía disponer el modelo BI.
* Se determinaron las métricas a las que apunta la solución con el fin de tener el monitoreo exacto que necesita la organización para mejorar la toma de decisiones con respecto a los objetivos que apunta el negocio.
* Para materializar los resultados conseguidos de todos los puntos abordados anteriormente se diagramó el diseño físico del modelo BI especificando todos sus antecedentes correspondientes.
* Se realizaron las cargas de los datos transaccionales a la base de datos en el STAGING AREA.
* Poblar las dimensiones junto con la tabla de hechos secuencialmente en base a las tablas del STAGING AREA, las cuales están formateadas y acomodadas a los respectivos tipos de datos que se necesitan.
* Ejecutar el proceso ETL completo sin errores.
* Construir los cubos con Analysis Services en base al modelo BI se cargó con el ETL generado anteriormente.
* Configurar el cubo según las necesidades del negocio para las consultas analíticas que serán desplegadas a futuro.
* Realizar las consultas pertinentes al negocio usando las métricas y los atributos que se disponen para el análisis de la información cargada y procesada en el cubo.

Luego de pasar por todos estos procesos se puede concluir que para poner en marcha un proyecto BI de la mejor manera se debe tener conocimiento del negocio el cual está sumergido la organización, así como también orientar la solución a los problemas a las necesidades del mismo. También es importante tener conocimiento de los conceptos básicos de la inteligencia de negocios para poder diseñar los modelos lo mejor posible. Así como también manejar las herramientas de *Integration Services* y *Analysis Services* que provee el IDE Visual Studio.

Para finalizar, se detectó que es muy importante poner en práctica las habilidades críticas en distintos ejercicios BI. Siguiendo esta práctica se puede alcanzar un mayor entendimiento en lo que consiste la Inteligencia de Negocios.