



Proyecto Business Intelligence

Northwind

Mario González Galdames

Introducción

Durante la última década, el número de empresas alrededor del mundo ha crecido de forma casi exponencial, incrementando los volúmenes de datos que producen durante su vida útil. Es a raíz de esto que comenzaron a surgir nuevas metodologías para poder operar volúmenes de datos cada vez mayores y transformar esos datos en información de calidad, y de esta forma combinando estas metodologías con estrategias de negocios y las tecnologías de la información es que nace el concepto de Business Intelligence.

El término Inteligencia de Negocios (Business Intelligence), conocido habitualmente como “BI”, es muy utilizado hoy en día; lo utilizan tanto expertos, como las propias compañías que desarrollan productos de esta área. Hay una gran cantidad de definiciones, y tampoco es el objetivo de esta entrada hacer una definición exhaustiva del término, sino que va orientado principalmente a transmitir este concepto de una forma pragmática.

Entonces podríamos definir BI en palabras sencillas como el conjunto de estrategias, tecnologías y metodologías que nos ayuda a convertir los datos en información de calidad, y dicha información en conocimiento que nos permita una toma de decisiones más acertadas y que nos ayude así a mejorar nuestra competitividad. Los recursos BI nos permiten predecir y descubrir nuevos patrones de comportamiento para avanzarse a los cambios y, así, identificar oportunidades de mercado. Esto, por supuesto, entrega una importante ventaja competitiva a aquellas empresas que utilizan el Business Intelligence a su favor. Como mencionamos anteriormente, todo lo descrito se adoptará de forma pragmática en un caso en específico propuesto previamente por el docente, donde se aplicarán diversas metodologías, procesos ETL y recursos que nos brinda BI para la elaboración de un Datawarehouse a partir de la base de datos “Northwind”, el cual es un recurso de acceso gratuito que nos permitirá realizar las transformaciones pertinentes.

1. Metodologías de desarrollo de un Datawarehouse:

Los Data Warehouse toman cada día mayor importancia, a medida que las organizaciones pasan de esquemas de sólo recolección de datos a esquemas de análisis de los mismos. Sin embargo, a pesar de la gran difusión de los conceptos relacionados con los almacenes de datos, no existe demasiada información objetiva en cuanto a las metodologías para implementarlos, por lo que eventualmente se tendrá que seleccionar la metodología que más se adapte al modelo que se tiene y al resultado que se desea obtener.

Existen muchas metodologías de diseño y construcción de DW. Cada fabricante de software de inteligencia de negocios busca imponer una metodología con sus productos. Sin embargo, se imponen entre la mayoría dos metodologías, la de Kimball y la de Inmon.

En resumen, la metodología de Inmon se basa en conceptos bien conocidos del diseño de bases de datos relacionales, la metodología para la construcción de un sistema de este tipo es la habitual para construir un sistema de información, utilizando las herramientas habituales. Mientras que la de Kimball, se basa en un modelado dimensional (no normalizado).

1.1. Metodología a utilizar:

Pensamos que la metodología más acorde al proyecto que se va a desarrollar es la de Kimball, ya que esta nos proporcionará un enfoque de menor a mayor, muy versátil y una serie de herramientas prácticas que ayudan a la implementación de un DW. Es acorde a nuestro proyecto simplemente porque este se basa en obtener un cubo OLAP a partir de una base de datos en específico. Por tanto, se detallará esta metodología a continuación.

1.2. Metodología Kimball:

La metodología Kimball propone tener un experto en cada área de la organización que se encargue del diseño de la información más relevante para de esta forma crear los modelos de datos y evitar redundancia de información. Esta se basa en lo que él denomina “Ciclo de Vida Dimensional del Negocio” para la obtención de un Datawarehouse de calidad y consistente. Este ciclo de vida del proyecto de DW, está basado en cuatro principios básicos los cuales se describirán a continuación:

1.2.1. Siempre ofrecer la solución completa:

La finalidad de este principio es que se debe hacer una entrega de un diseño funcional, que cumpla con las expectativas de la organización, se deben entregar herramientas de consulta, aplicaciones gráficas para informes, capacitación y soporte, dependiendo de la tecnología elegida para la implementación. Esto indica que es fundamental brindar todos los elementos necesarios para entregar valor a los usuarios de negocios o en otras palabras, tener un almacén de datos sólido, bien diseñado, con calidad probada, y accesible. También se deberá entregar herramientas de consulta ad hoc, aplicaciones para informes y análisis avanzado, capacitación, soporte, sitio web y documentación

1.2.2. Construir una infraestructura de información adecuada:

Se debe hacer un análisis a fondo de la información a analizar, para así poder generar los modelos adecuados para los data marts. Esto implica diseñar una base de información única, integrada, fácil de usar, de alto rendimiento donde se refleja la amplia gama de requerimientos de negocio identificados en la empresa.

1.2.3. Realizar entregas en incrementos significativos:

Según esta metodología se deben hacer entregas en plazos acordados con la organización teniendo en cuenta que estos no sean tan extensos ni tan cortos. Se recomienda crear el almacén de datos en incrementos entregables en plazos de 6 a 12 meses. Hay que usar el valor de negocio de cada elemento identificado para determinar el orden de aplicación de los incrementos. En esto la metodología se parece a las metodologías ágiles de construcción de software.

1.2.4. Centrarse en el negocio:

Hay que tener un conocimiento profundo acerca del negocio, para poder implementar el modelo a partir de un buen esquema de requerimientos. Esto requiere Identificación de los requerimientos del negocio y su valor asociado, y usar estos esfuerzos para desarrollar relaciones sólidas con el negocio, agudizando el análisis de este y la competencia consultiva de los implementadores.

En resumen, la metodología Kimball es de gran ayuda para este tipo específico de proyectos ya que facilita la creación de un DWH ya que de acuerdo con las misiones de las diferentes áreas de la organización permite el diseño y la

construcción de los datamarts específicos, que darán forma al futuro modelo general del sistema.

No obstante, es necesario seguir una serie de tareas en específico para lograr crear un datawarehouse consistente y de calidad, por lo que serán planteados a continuación:

Metodología Ralph Kimball

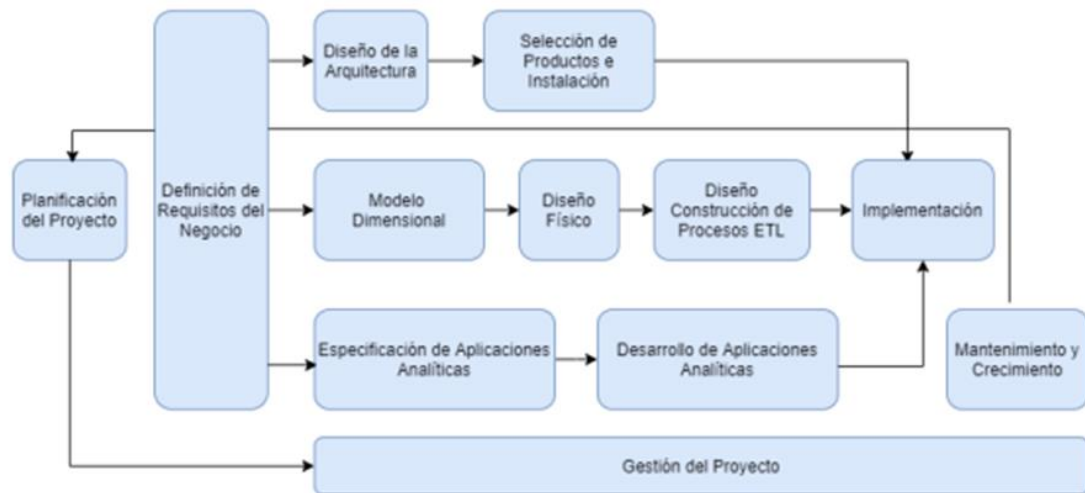


Figura 1: Diagrama de la metodología de Ralph Kimball

De acuerdo con la figura anterior la metodología Kimball se clasifica en 3 caminos los cuales contienen las fases que unidas entre sí conformarán el ciclo de vida de esta, estos caminos son los siguientes:

- Tecnología: Esta parte está asociada al diseño de la arquitectura y a la selección de las herramientas informáticas que serán utilizadas en la aplicación de la metodología.
- Datos: En esta capa se implementarán los modelos dimensionales y los modelos de transformación de los datos.
- Aplicaciones de Inteligencia de Negocios: En esta capa se diseñarán las aplicaciones de lógica de negocios que mostrarán los resultados de la metodología planteada para los usuarios de negocio.

2. Definición de los requerimientos del proyecto

Como mencionamos anteriormente, el proyecto consta en la obtención de un cubo OLAP a partir de la base de datos propuesta “Northwind Traders”.

Northwind Traders es una organización ficticia que gestiona pedidos, productos, clientes, proveedores y muchos otros aspectos de una pequeña empresa. Northwind es una conocida base de datos de ejemplo de Microsoft que se utiliza en multitud de libros, cursos y artículos para ilustrar todo tipo de conceptos de acceso a datos.

Esta base de datos simula la información acumulada del día a día en una empresa ficticia llamada Northwind Traders. Incluye tablas para clientes, proveedores, facturas, líneas de factura, regiones comerciales, etc, y a través de ellas aporta ejemplos de los principales tipos de relaciones entre tablas, tipos de datos habituales, etc. Por lo que es muy apropiada para el aprendizaje del manejo de bases de datos.

Apareció como ejemplo para Microsoft Access en el año 1994, y más adelante Microsoft creó una versión específica para su gestor empresarial de datos SQL Server.

No obstante, lo cierto es que no se deja de utilizar, y constantemente nos encontraremos ejemplos que hacen uso de ella, por lo que conviene tener claro cómo podemos obtenerla y ponerla en marcha en nuestro sistema.

Existen otras bases de datos de ejemplo (como AdventureWorks) pero Northwind sigue siendo la más utilizada por su sencillez. Ha habido intentos de crear versiones modernizadas de la misma, con más campos y tipos de datos, pero hoy por hoy no existe a disposición de la comunidad ninguna versión de la base de datos Northwind más actualizada.



Dicho esto, se utilizará la base de datos presentada para proponer una serie de indicadores de gestión (KPI) que fueron elegidos de forma arbitraria, para posteriormente ser resueltos con la metodología correspondiente. En este caso los KPI seleccionados fueron los siguientes:

KPI N°1:

Objetivo del análisis	Se necesita saber cuántas ventas realizó cada país según el vendedor
KPI	Total de ventas en cada país por vendedor

KPI N°2:

Objetivo del análisis	La empresa necesita saber cual es el empleado que vende mas productos en general
KPI	Empleado que vende mayor cantidad de productos

KPI N°3:

Objetivo del análisis	La empresa necesita saber que compañía realizó más cantidad de compras
KPI	Compañía que compró más productos

3. Base de datos fuente

En este caso la base de datos Northwind se encontraba en formato SQL, y para su instalación se debía ejecutar un código mediante una query, gracias a la cual se creaba cada tabla con sus atributos. En la situación actual, la BD contaba con un total de 13 tablas, las cuales estaban organizadas de la siguiente manera:

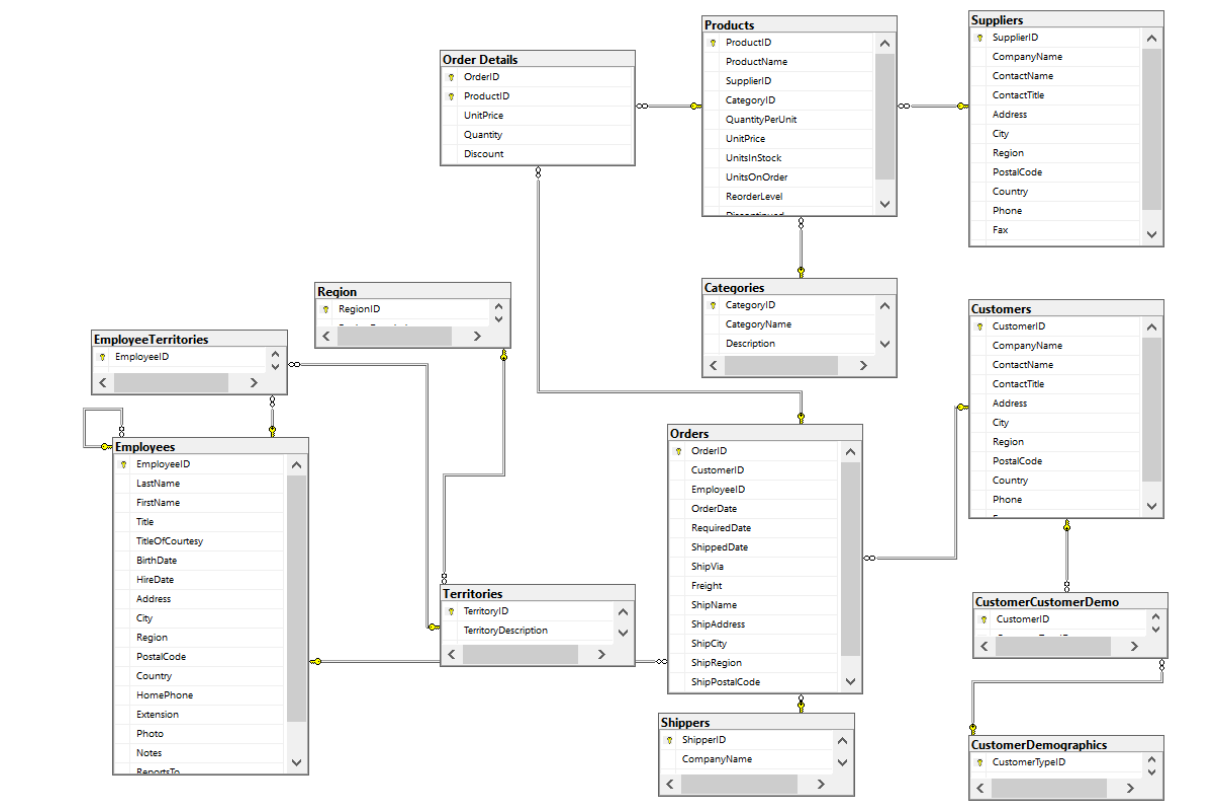


Figura 2. Modelo Crudo Northwind

3.1. Modelo Relacional

En este caso la tabla que más datos importantes contenía fue la de Orders, por lo tanto fue la utilizada para realizar la tabla de hechos del modelo relacional (MR), Utilizando para ello los distintos ID de las otras tablas, que finalmente serán las claves foráneas de la Fact Table, y en cuanto a las tablas restantes, lo que se realizó fue filtrar los datos y eliminar aquellos que no servían, esto con el fin de simplificar el trabajo y mejorar la visualización. A continuación se detalla el modelo relacional final, y como quedaron distribuidas las tablas dentro de la base de datos.

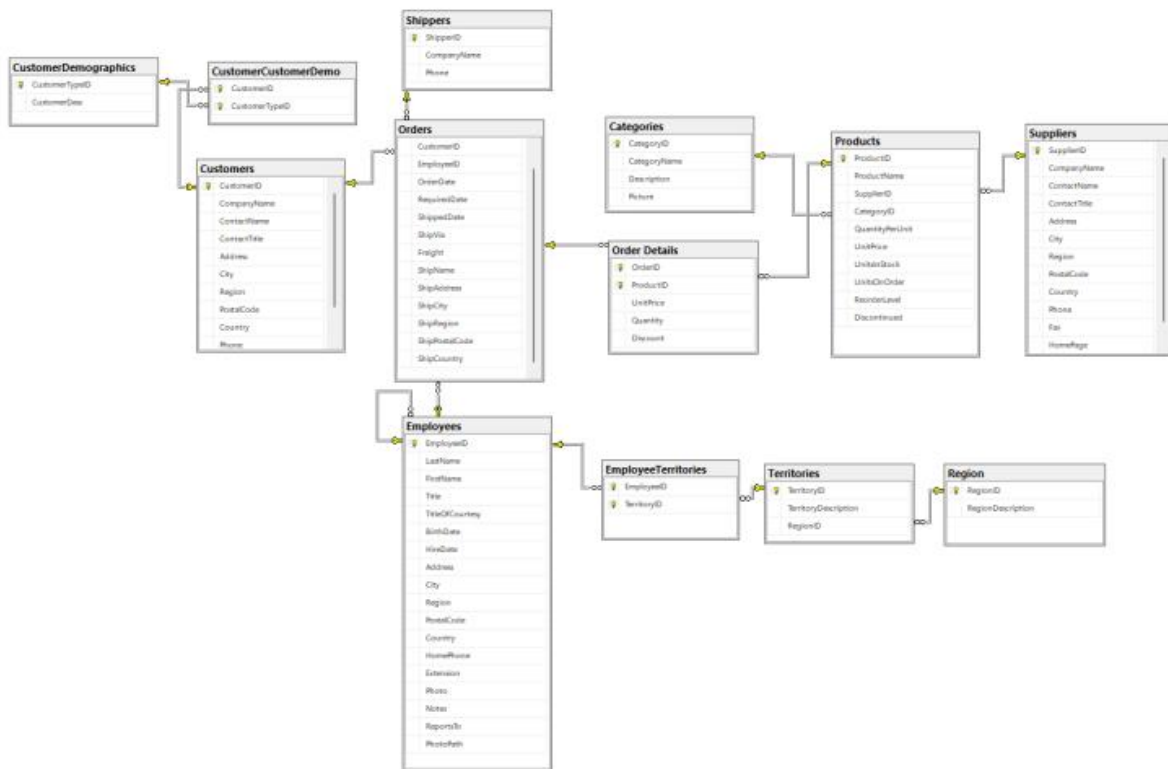


Figura 3. Modelo Relacional

3.2. Tabla de hechos

/****** Script for SelectTopNRows command from SSMS *****/

```

SELECT TOP (10) [ProductID]
, [CategoryID]
, [CustomerID]
, [Fecha]
, [EmployeeID]
, [SupplierID]
, [Quantity]
, [UnitPrice]
, [Total]
FROM [NorthwindETL].[dbo].[FactTable]
  
```

	ProductID	CategoryID	CustomerID	Fecha	EmployeeID	SupplierID	Quantity	UnitPrice	Total
1	11	4	VINET	1996-07-04	5	5	12	14,00	168
2	42	5	VINET	1996-07-04	5	20	10	9,80	98
3	72	4	VINET	1996-07-04	5	14	5	34,80	174
4	14	7	TOMSP	1996-07-05	6	6	9	18,60	167
5	51	7	TOMSP	1996-07-05	6	24	40	42,40	1696
6	41	8	HANAR	1996-07-08	4	19	10	7,70	77
7	51	7	HANAR	1996-07-08	4	24	35	42,40	1484
8	65	2	HANAR	1996-07-08	4	2	15	16,80	252
9	22	5	VICTE	1996-07-08	3	9	6	16,80	100
10	57	5	VICTE	1996-07-08	3	26	15	15,60	234

3.3. Tabla dimensión Tiempo

SQLQuery1.sql - LA...B4VFD4S\luisi (51))

```
/****** Script for SelectTopNRows command from SSMS *****/  
SELECT TOP (1000) [Fecha]  
                , [AÑO]  
                , [MES]  
FROM [etl-northwind].[dbo].[DIM Tiempo]
```

100 %

Results Messages

	Fecha	AÑO	MES
1	1996-07-04	1996	7
2	1996-07-05	1996	7
3	1996-07-08	1996	7
4	1996-07-09	1996	7
5	1996-07-10	1996	7

3.4. Tabla Customer (cliente)

```
/****** Script for SelectTopNRows command from SSMS *****/  
SELECT TOP (15) [CustomerID]  
                , [CompanyName]  
                , [City]  
                , [Country]  
FROM [NorthwindETL].[dbo].[DIM_Customer]
```

	CustomerID	CompanyName	City	Country
1	ALFKI	Alfreds Futterkiste	Berlin	Germany
2	ANATR	Ana Trujillo Emparedados y helados	México D.F.	Mexico
3	ANTON	Antonio Moreno Taquería	México D.F.	Mexico
4	AROUT	Around the Horn	London	UK
5	BERGS	Berglunds snabbköp	Luleå	Sweden
6	BLAUS	Blauer See Delikatessen	Mannheim	Germany
7	BLONP	Blondesddsl père et fils	Strasbourg	France
8	BOLID	Bólido Comidas preparadas	Madrid	Spain
9	BONAP	Bon app'	Marseille	France
10	BOTTM	Bottom-Dollar Markets	Tsawassen	Canada
11	BSBEV	B's Beverages	London	UK
12	CACTU	Cactus Comidas para llevar	Buenos Aires	Argentina
13	CENTC	Centro comercial Moctezuma	México D.F.	Mexico
14	CHOPS	Chop-suey Chinese	Bern	Switzerland
15	COMMI	Comércio Mineiro	Sao Paulo	Brazil

3.5. Tabla Supplier (proveedor)

```
/****** Script for SelectTopNRows command from SSMS *****/
SELECT TOP (1000) [SupplierID]
    , [CompanyName]
    , [City]
    , [Country]
FROM [etl-northwind].[dbo].[DIM Suppliers]
```

100 %

Results Messages

	SupplierID	CompanyName	City	Country
1	1	EXOTIC LIQUIDS	LONDON	UNITED KINGDOM
2	2	NEW ORLEANS CAJUN DELIGHTS	NEW ORLEANS	USA
3	3	GRANDMA KELLY'S HOMESTEAD	ANN ARBOR	USA
4	4	TOKYO TRADERS	TOKYO	JAPAN
5	5	COOPERATIVA DE QUESOS 'LAS CABRAS'	OVIEDO	SPAIN
6	6	MAYUMI'S	OSAKA	JAPAN
7	7	PAVLOVA, LTD.	MELBOURNE	AUSTRALIA
8	8	SPECIALTY BISCUITS, LTD.	MANCHESTER	UNITED KINGDOM

3.6. Tabla Product

```
/****** Script for SelectTopNRows command from SSMS *****/
SELECT TOP (1000) [ProductID]
    , [Producto]
    , [Categoria]
FROM [etl-northwind].[dbo].[DIM Producto]
```

100 %

Results Messages

	ProductID	Producto	Categoria
1	1	CHAI	BEVERAGES
2	2	CHANG	BEVERAGES
3	3	ANISEED SYRUP	CONDIMENTS
4	4	CHEF ANTON'S CAJUN SEASONING	CONDIMENTS
5	5	CHEF ANTON'S GUMBO MIX	CONDIMENTS
6	6	GRANDMA'S BOYSENBERRY SPREAD	CONDIMENTS
7	7	UNCLE BOB'S ORGANIC DRIED PEARS	PRODUCE
8	8	NORTHWOODS CRANBERRY SAUCE	CONDIMENTS
9	9	MISHI KOBE NIKU	MEAT/POULTRY
10	10	IKURA	SEAFOOD

3.7. Tabla Employee (empleado)

```
/****** Script for SelectTopNRows command fr
```

```
SELECT TOP (1000) [EmployeeID]
```

```
        ,[FullName]
```

```
FROM [etl-northwind].[dbo].[DIM Employee]
```

100 %

Results Messages

	EmployeeID	FullName
1	1	NANCY DAVOLIO
2	2	ANDREW FULLER
3	3	JANET LEVERLING
4	4	MARGARET PEACOCK
5	5	STEVEN BUCHANAN
6	6	MICHAEL SUYAMA
7	7	ROBERT KING
8	8	LAURA CALLAHAN
9	9	ANNE DODSWORTH

3.8. Distribución de las tablas en SQL

NorthwindETL

- Database Diagrams
- Tables
 - System Tables
 - FileTables
 - External Tables
 - Graph Tables
 - dbo.DIM_Customer
 - dbo.DIM_Employee
 - dbo.Dim_Product
 - dbo.DIM_Supplier
 - dbo.DIM_Tiempo
 - dbo.FactTable

4. DISEÑO DE LA ARQUITECTURA BI

Para el desarrollo del proyecto se utilizaron las herramientas que dispone Microsoft SQL Server Management 18.

Microsoft SQL Server Management Studio (SSMS), es un entorno integrado para administrar cualquier infraestructura de SQL, desde SQL Server a Azure SQL Database. SSMS proporciona herramientas para configurar, supervisar y administrar instancias de SQL Server y bases de datos. SSMS es usado para implementar, supervisar y actualizar los componentes de nivel de datos que usan las aplicaciones, además de compilar consultas y scripts.

Microsoft SQL Server está compuesto por:

- Database Engine: Es el motor de la base de datos relacional, Es el componente o servicio principal, encargado de administrar las base de datos que da respuestas a las peticiones de uso de las tablas, campos, etc.
- Analysis Services: Es el motor de bases de datos multidimensionales. Este servicio o componente nos permite diseñar, distribuir, explorar los cubos de información y modelos de minería de datos.
- Integration Services: Este servicio permite la implementación de servicios de ETL (Extracción, Transformación y Carga), posibilitando migrar datos de un lugar a otro, hacer conexiones entre diversas bases de datos para realizar las tareas de limpieza, normalizacion, y estandarizacion de datos, o simplemente trasladar datos de un lugar a otro.
- Reporting Services: permite diseñar, almacenar y ejecutar reportes que han sido diseñados mediante SQL Server y que acceden a la información de las bases de datos relacionales.

Como podemos ver en la imagen superior, un proceso bi se inicializa recogiendo los datos almacenados en la base de datos relacional (Datos relacionados entre sí) (se entiende como un conjunto de tablas o relaciones) OLTP (procesamiento de

transacciones en línea es un tipo de procesamiento de datos que consiste en ejecutar una serie de transacciones de manera simultánea en cualquier entorno ya sea de compras, banca en línea , etc), los cuales mediante una serie de consultas utilizando la herramienta de gestión de bdd de microsoft, es posible aislar la información necesaria. Es posible por lo demás encontrar los datos de distintas formas, ya sea en archivos planos u aplicaciones empresariales para el almacenamiento de datos.

El proceso que disecciona la información de origen y la exporta al almacén de datos datawarehouse es el ETL, entendido como extraer, transformar y cargar. Se extrae los elementos necesarios mediante las consultas previamente mencionadas y se exporta la información a la bdd de destino, la cual tendrá la estructura que se estime conveniente, en este caso es de tipo estrella puesto que cada dimensión no tiene relación con las otras, pero todas individualmente forman parte de una tabla de hechos.

Una vez recolectados los datos y trabajados, estos son almacenados en el datawarehouse para su futura utilización en el motor de análisis de datos conocido como cubo olap, limpiando así el almacén de datos. Cada cubo distribuye los datos en sus dimensiones y estas se completan con miembros organizados jerárquicamente.

Para este caso se utiliza un sistema olap de tipo multidimensional osea Molap, puesto que los análisis se realizan directamente en una base de datos multidimensional mediante estructuras de datos diversas, normalmente multidimensionales conocidas como cubos olap que contienen datos resumidos de grandes bases de datos o sistemas transaccionales (oltp). Este análisis de datos se resume en agilizar la consulta de grandes cantidades de datos .

Por último se realiza la conexión con power bi, que es una solución para el análisis empresarial de datos, con el cual es posible generar los dashboards en los que se verán representados los kpi.

Todo lo expuesto anteriormente se puede representar mediante un esquema por capas nivelada desde la capa 1 a la 5.

Capa 1: Carga de datos y uso de los mismos mediante la herramienta de administración de bases de datos de Microsoft.

Capa 2: Realización del Etl, que consiste como ya se explicó en la extracción desde una bd de origen, transformación de los datos y carga en una bd de destino. Se utiliza para esto Visual Studio y Sql Management Studio.

Capa 3: Trabajo en el cubo, en donde se distribuye la información a usar en cada dimensión (y hechos), jerarquizando la misma si así se estima conveniente.

Capa 4: Se procesa y crea el cubo, con este será posible acceder a los datos de la forma más eficiente.

Capa 5: Se crean los reportes en función a lo almacenado en el cubo Olap. Esto para presentar la información de la forma más clara y sin complicaciones posible, facilitando el análisis de los datos y la toma de decisiones y es requerida.

5.1.1. Análisis de la tabla de hechos propuesta.

La tabla de hechos que se utilizará para el modelo multidimensional propuesto consta de 8 hechos los cuales proporcionarán la información correspondiente para dar respuesta a los KPI establecidos, los cuales se enfocan principalmente a los pedidos realizados en la empresa.

FactTable	
	ProductID
	CategoryID
	CustomerID
	Fecha
	EmployeeID
	SupplierID
	Quantity
	UnitPrice
	Total

Figura 4: Representación de la tabla de hechos.

5.1.2. Análisis de las tablas de dimensiones propuestas

Las tablas de dimensiones propuestas representarán aquello que se quiere almacenar respecto a un tema concreto. En una misma base de datos multidimensional puede haber numerosas tablas de dimensiones asociadas entre sí. Las tablas de dimensiones pueden ser generadas de forma automática, o creadas manualmente por los propios usuarios, lo cual fué lo que sucedió en este caso.

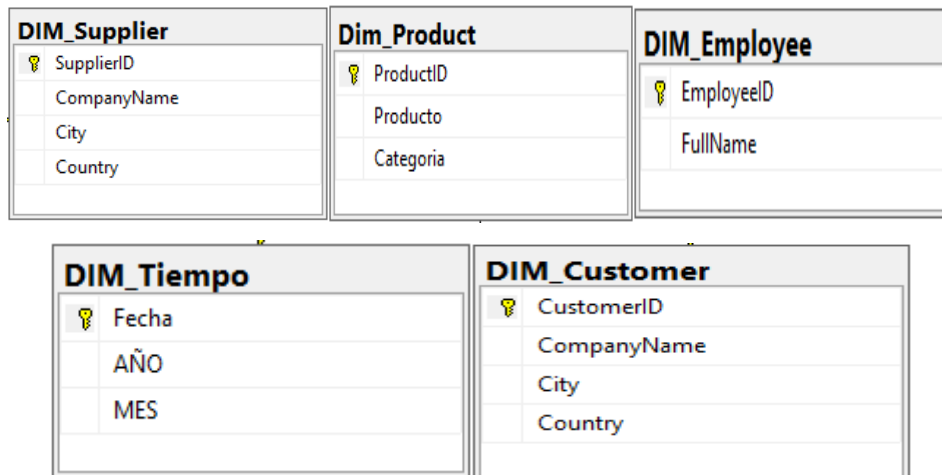


Figura 5: Representación de tablas de dimensiones.

5.1.3. Modelo gráfico de alto nivel

El esquema en estrella, que está compuesto por una tabla central (Facts Table), la que abarca datos cuantitativos, también conocidos como medidas, y una serie de dimensiones que rodea la tabla principal, cada una de estas dimensiones están compuestas por solo una tabla.

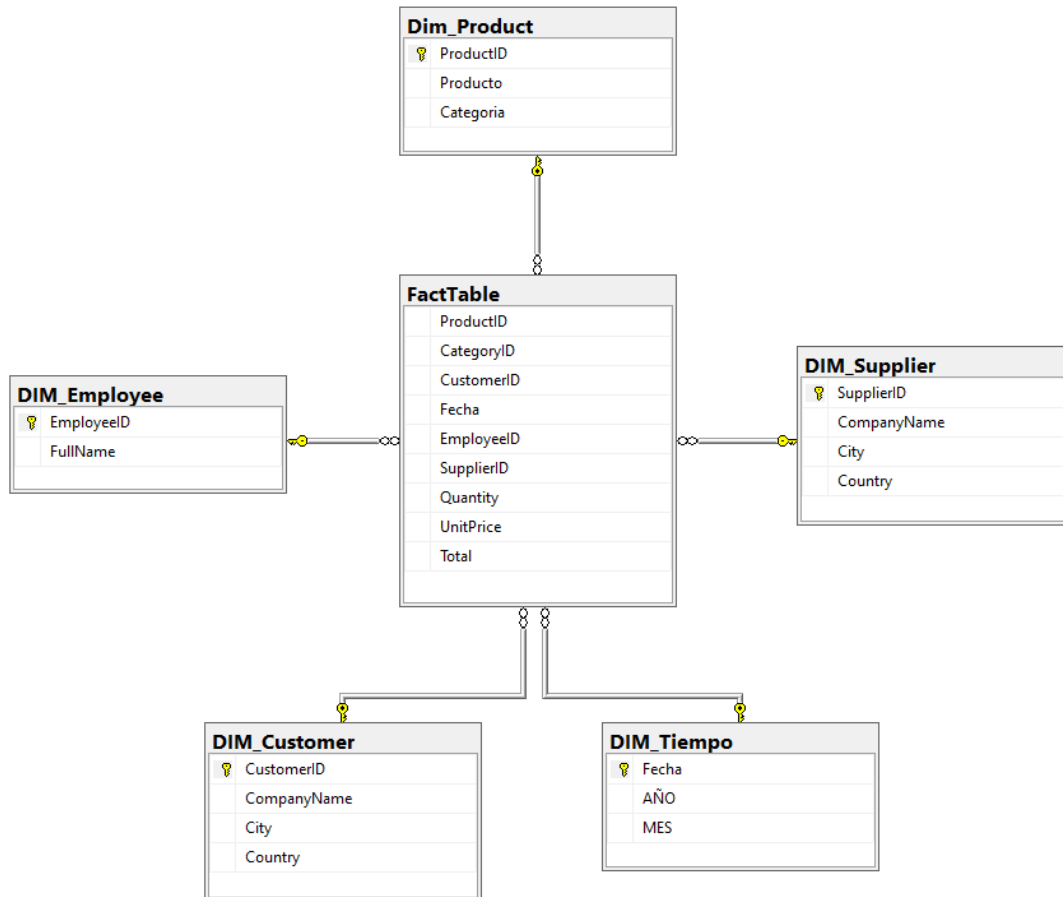


Figura 6: Esquema de estrella.

5.1.4. Prueba de modelo

En este caso, los KPI pedidos fueron los siguientes:

- a) Total de ventas en cada país por vendedor.
- b) Empleado que vende más cantidad de productos.
- c) Compañía que compró más productos.

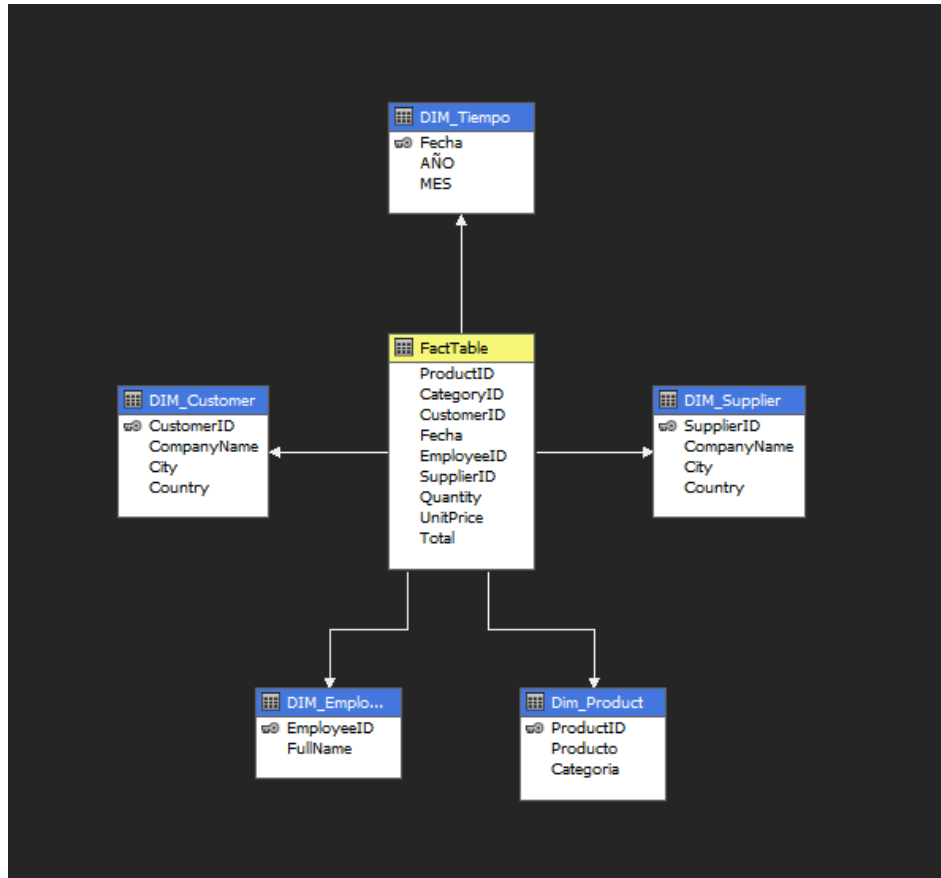


Figura 6: Esquema de estrella previo a la creación del cubo Olap (Visual Studio)

Se puede comprobar a través del modelo multidimensional que los KPI descritos anteriormente son respondidos efectiva y claramente con el modelo.

5.2. Diseño Físico y ETL: Extracción, Transformación y Carga.

El proceso de extracción, transformación y carga (ETL) es entendido como una canalización o enfoque de los datos, el cual se usa para recopilar datos de varias fuentes. A continuación, se transforman los datos en función a las reglas de negocio para ser cargados en un almacén de datos. La fase de transformación tiene su desarrollo en un motor especializado, se utilizan frecuentemente tablas temporales para el almacenamiento provisional de los datos que se están transformando, hasta que sean cargados en su lugar de llegada.

Esta transformación conlleva a menudo varias operaciones de filtrado, ordenación, agregación, combinación de datos, limpieza de estos, desaplicación y validación de datos. La idea es ejecutar las tres fases del proceso ETL en paralelo para poder ahorrar tiempo. Ejemplo de esto es que mientras se extraen los datos, se emplea un proceso de transformación sobre los datos ya recibidos y de preparación para su futura carga, así no se espera a que el proceso de extracción termine del todo.

Para poder realizar todo lo descrito se comenzará con la conexión a la fuente de datos original, esto mediante la herramienta Visual Studio. Con los datos ya capturados estos se tratarán en el apartado de flujo de datos, aquí la lógica empresarial será la encargada de hacernos ver cómo proceder y de qué manera llevar los datos a la fuente de destino.

En este punto se resuelve la segunda capa del modelo a presentar, el cual se encarga de preparar los datos crudos provenientes de la base de datos “Northwind” para posteriormente aplicar las transformaciones pertinentes para obtener los datos que se utilizarán para elaborar el cubo OLAP.

5.2.1 Tabla de hechos



Figura 7: Obtención tabla de hechos en ETL

Para obtener la tabla de hechos necesaria, con los datos consistentes y sin errores, no fué necesario aplicar una gran variedad de transformaciones, puesto que los datos ya venían tratados en la base de datos utilizada, por lo que simplemente se utilizan los objetos para capturar sus datos, ordenarlos y poblar la tabla de hechos

que se utilizará posteriormente para el cubo OLAP, asignándole en este apartado sus respectivas claves foráneas.

5.2.2 Dimensiones Producto, Cliente, Supplier, Employee, Tiempo

Para crear las dimensiones pertinentes, se utilizó el mismo proceso en sí para todas las dimensiones, el cual consistió en utilizar los objetos necesarios para la extracción de datos desde la fuente de origen, utilizar el objeto “UPPER” para transformar los datos a mayúsculas y así agilizar las consultas en el resultado final, el objeto “Columna derivada” para cambiar strings en ciertos apartados y finalmente el objeto “Destino” para crear la dimensión y poblarse con los datos listos para ser procesados.

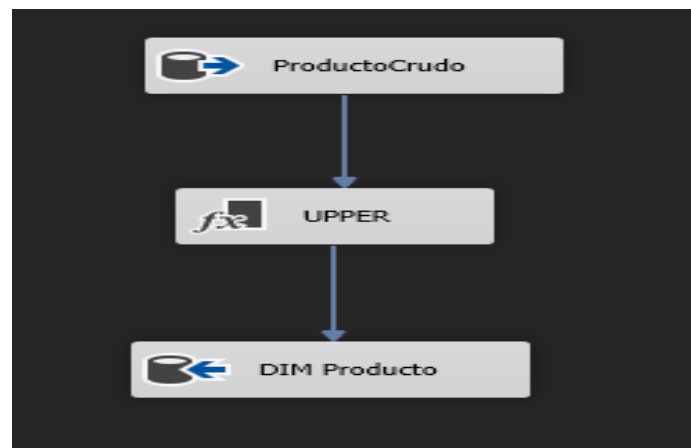


Figura 8: Dimensión Producto

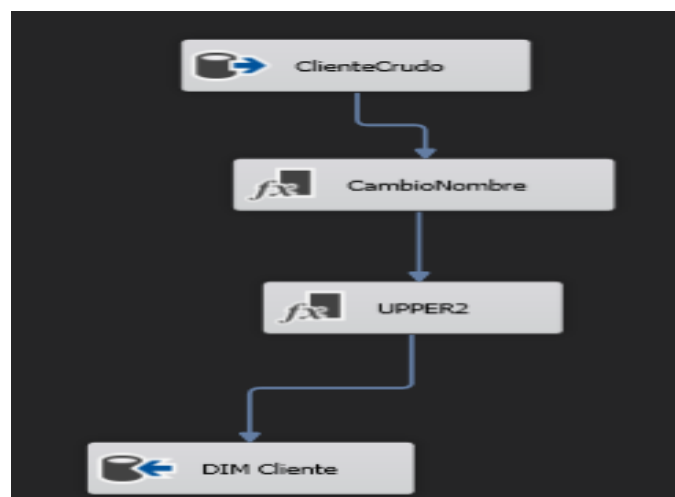


Figura 9: Dimensión Cliente

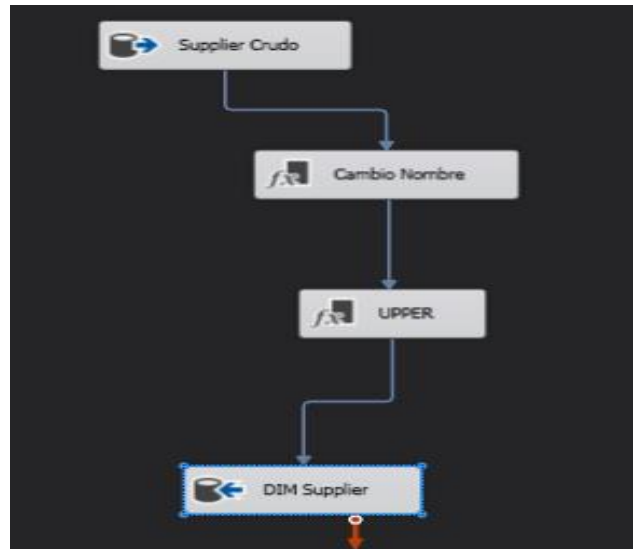


Figura 10: Dimensión Supplier

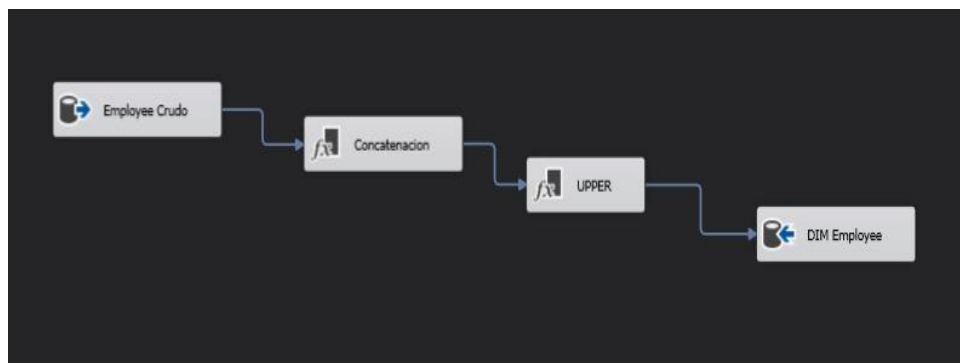


Figura 11: Dimensión Employee

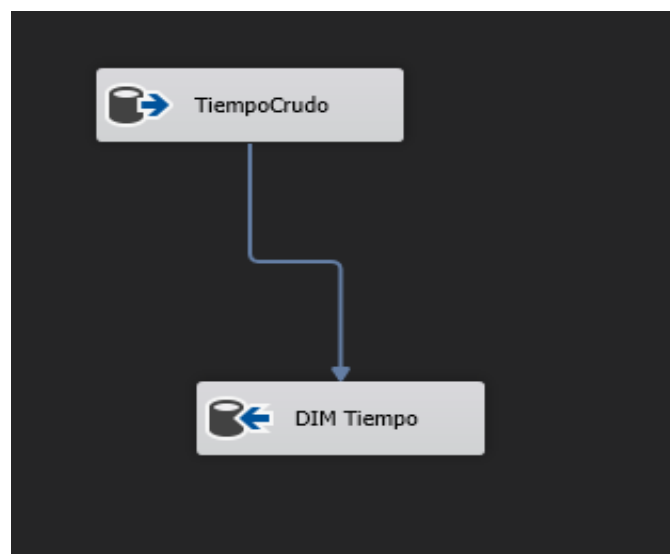


Figura 12: Dimensión Tiempo

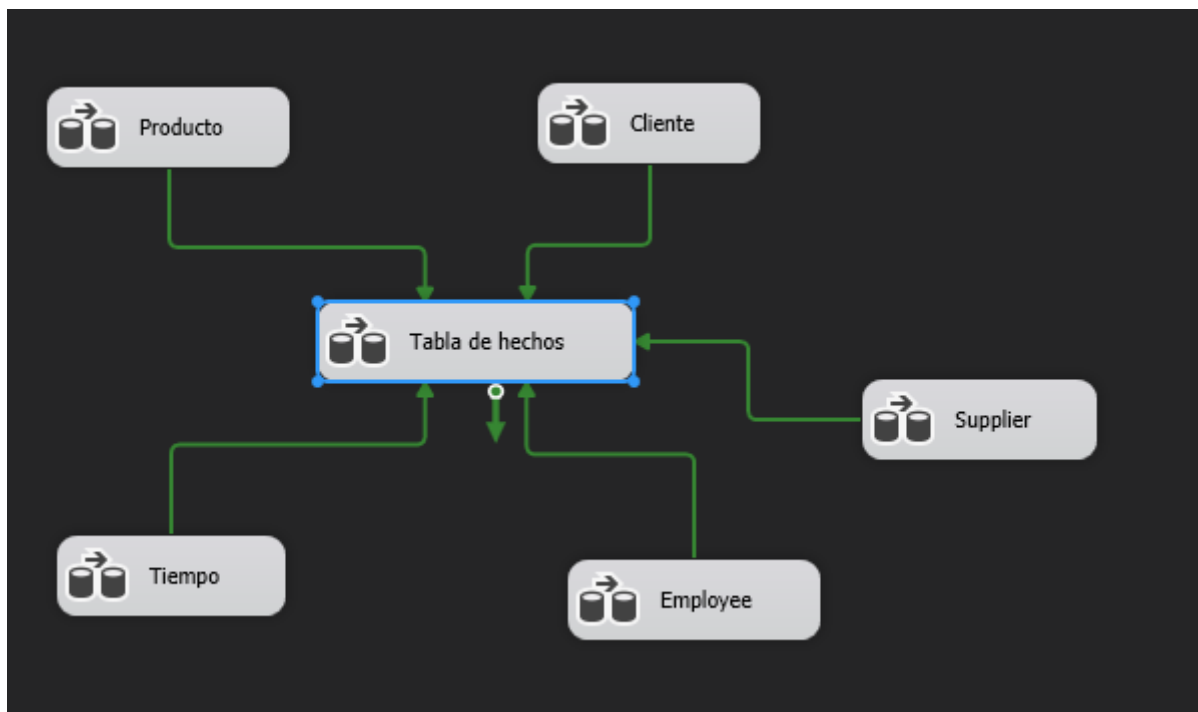


Figura 13: Modelo Estrella Final.

5.3. Diseño del cubo

Para la implementación del cubo se utilizó la herramienta Analysis Services de SQL server, la cual podemos encontrar en Visual Studio 2019, gracias a esta herramienta podemos cargar la tabla de hechos y las dimensiones con los datos que están incluidos en el Data warehouse.

El explorador de soluciones de este proyecto está compuesto de la siguiente manera:

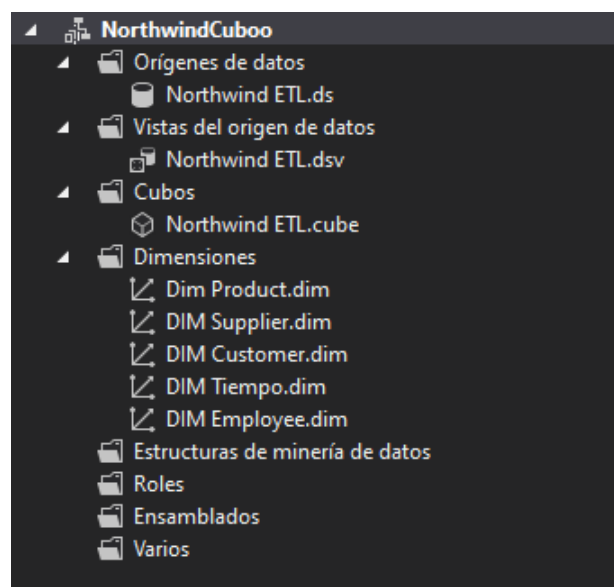
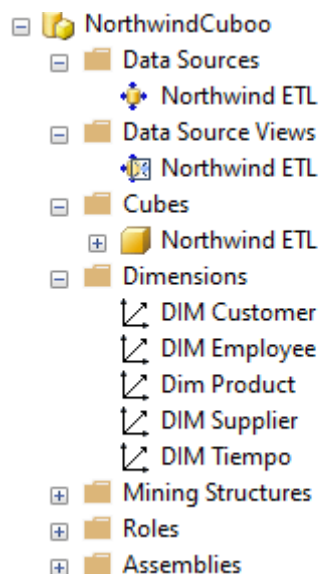


Figura 9: Vista Cubo Olap Visual

- **Orígenes de datos:** Es el primer paso que debemos realizar para poder crear el cubo, es donde se cargan los datos de origen provenientes de SQL Server.

- **Vistas del origen de datos:** Podemos ver la estructura que tendrá nuestro cubo, seleccionando las tablas correspondientes a las dimensiones y a la tabla de hechos.
- **Cubos:** Es donde toma forma el cubo, se debe señalar qué tabla es la de hechos y cuáles son las dimensiones. El modelo del cubo ya creado se puede ver en la figura N°7.
- **Dimensiones:** en este apartado podemos revisar las estructuras de las dimensiones creadas por el asistente.

Una vez realizados los procedimientos antes mencionados, queda crear una base de datos en el SQL server, en el motor de Analysis Services. A continuación, se presenta el cubo ya creado en el motor de Analysis Services.



5.4. Reporte de datos

Para la visualización, la herramienta utilizada fue Power BI de Microsoft, la cual permite realizar vistas tipo dashboard para así tener una mejor visualización de datos. En este caso, realizamos la siguiente dashboard con los 3 KPI descritos anteriormente:

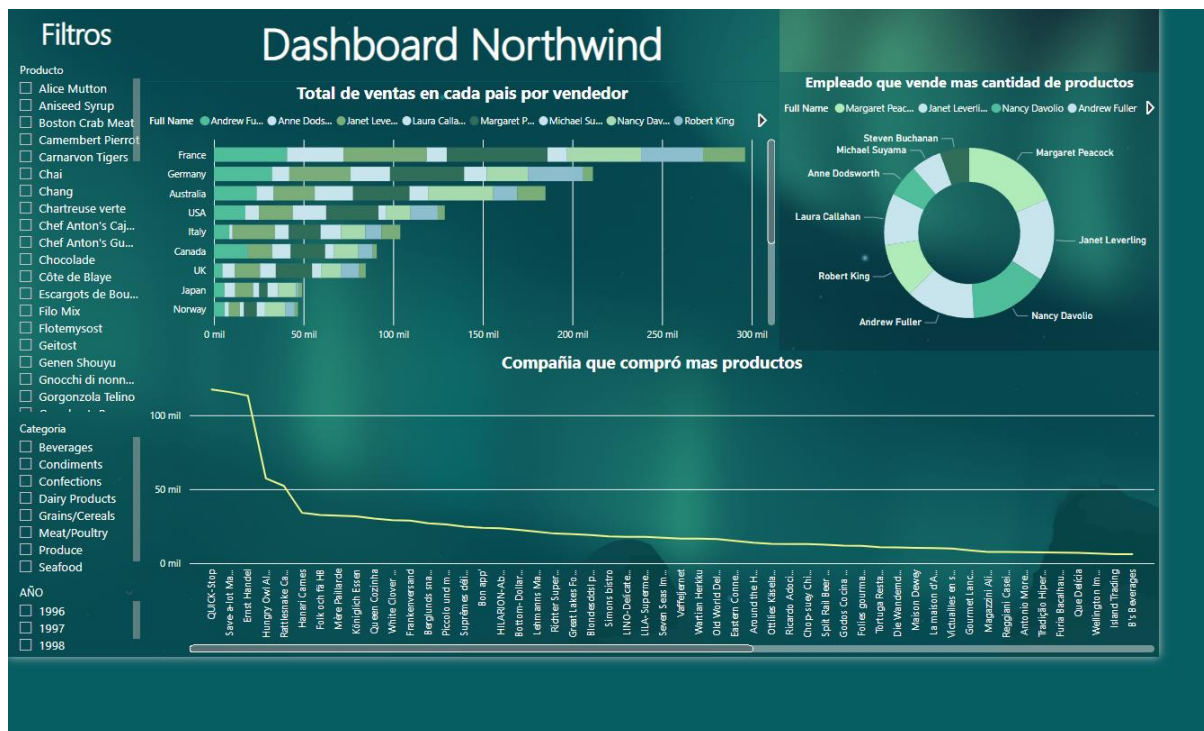


Figura 10: Reportes Power Bi

En la cual podemos observar los distintos gráficos que dan a conocer los datos y estadísticas solicitadas. Además de ello, tenemos en el lado izquierdo los filtros de producto, categoría y año, que nos ayudarán a dar una mejor visualización del modelo de datos confeccionado.

6. Conclusiones

De manera general, la realización de este proyecto de Business Intelligence, nos acercó de manera explosiva a lo que existe hoy en día en el mercado, aplicando conceptos clave en su desarrollo tales como el modelo relacional, del cual podemos rescatar la simplicidad de datos que finalmente quedaron enlazados, siguiendo con el diseño de capas, en donde vimos una clara secuencia de capas enlazadas una con la otra, pasando por el proceso ETL, el cual a nuestro parecer es la parte fundamental del proyecto, ya que dependiendo de si cumplimos con los estándares correcto, nuestros pasos siguientes tendrán éxito o no. Finalmente, en cuanto al diseño del cubo, nos encontramos con ciertos inconvenientes que se resolvieron rápidamente. En cuanto a la visualización de datos, creemos que la manera en la que se despliegan los datos es de un fácil entendimiento tanto como para desarrollador y consumidor, esto debido a que utilizamos una temática y paleta de colores coincidentes.

En resumen, el realizar un proyecto de Business Intelligence requiere un estricto procedimiento para cada una de sus fases, esto es debido a que cualquier error en su desarrollo puede llevar a una visualización mal ejecutada.