Desafío Practico 2

DATAWAREHOUSE Y MINERÍA DE DATOS

Integrantes:

Cruz González, José Roberto CG181933

Garay Alvarado, Bryan Walberto GA181935

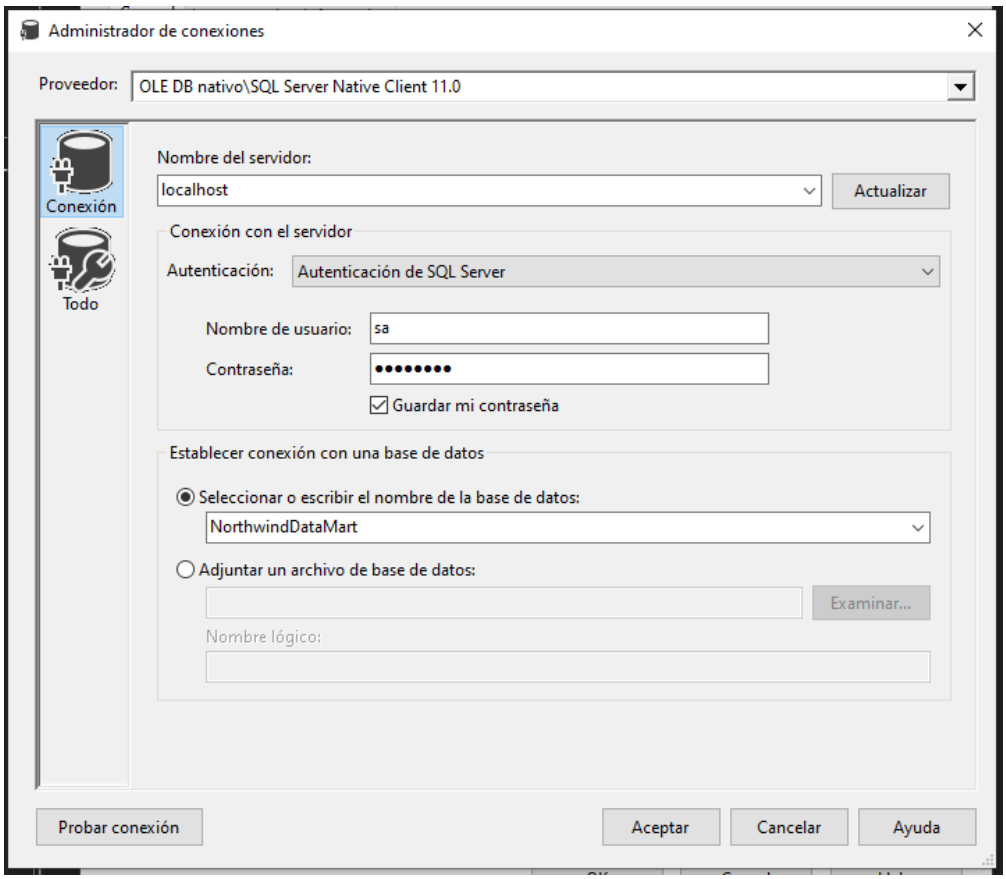
Docente:

Ing. Alexander Siguenza

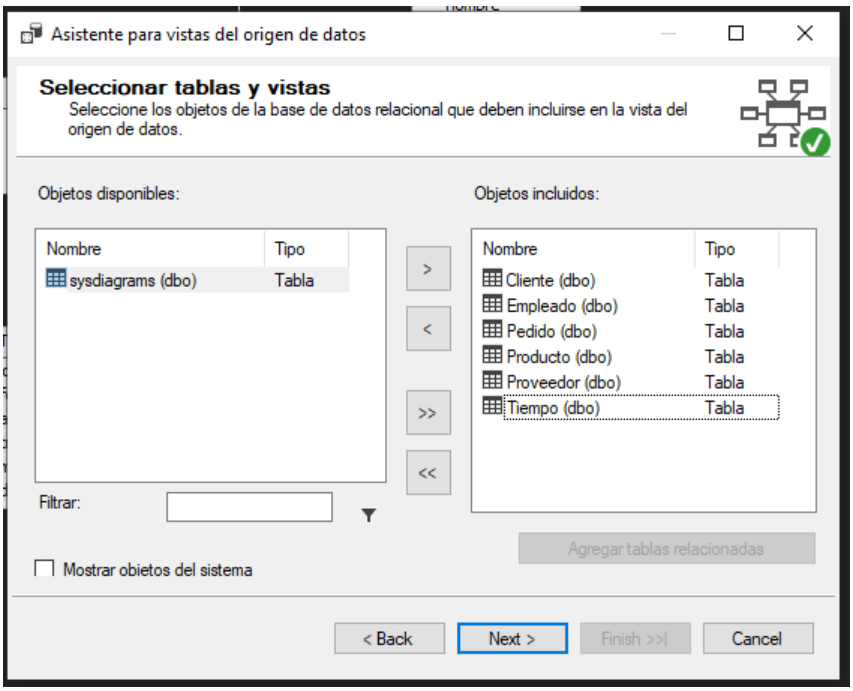
Soyapango, 04 de octubre de 2020

EJERCICIO 1

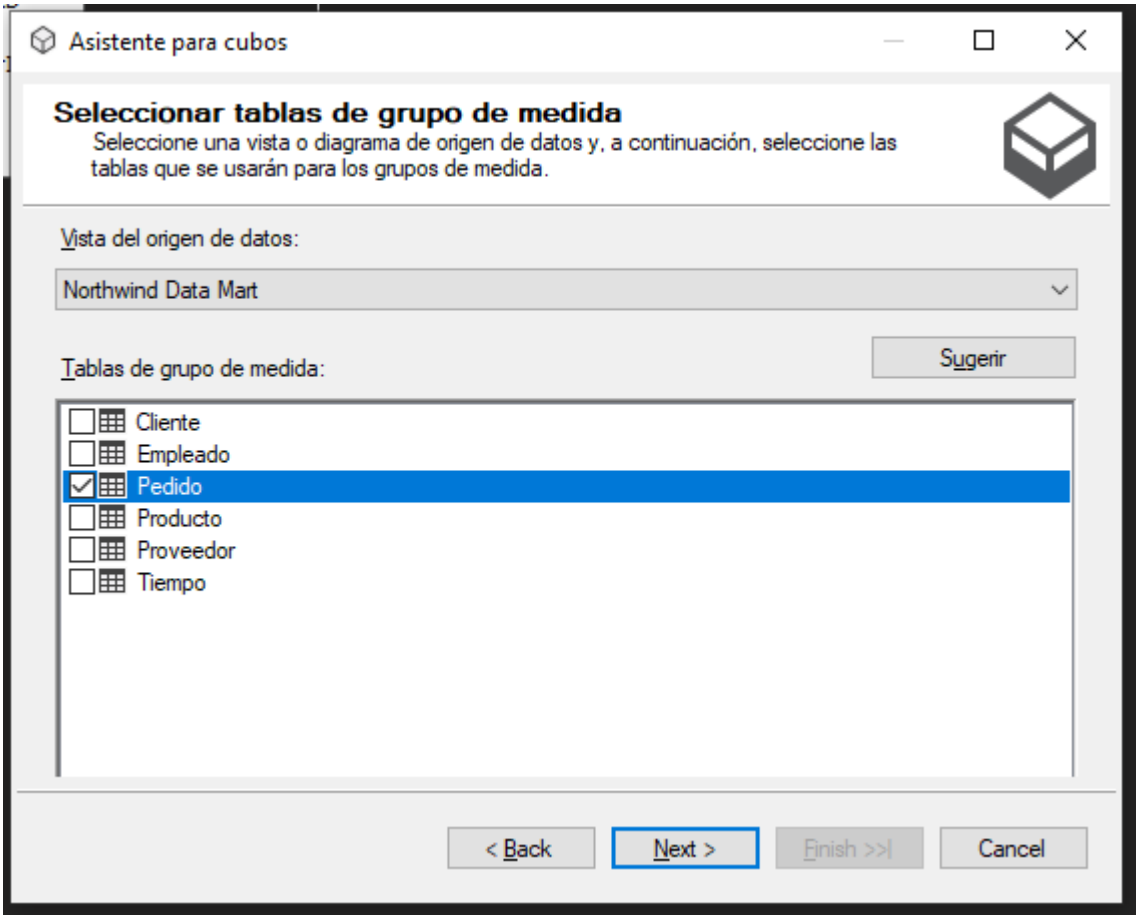
Para la creación del cubo del ejercicio numero 1 primero se debe crear un nuevo origen de datos en el cual se seleccione la base de datos “NorthwindDataMart”

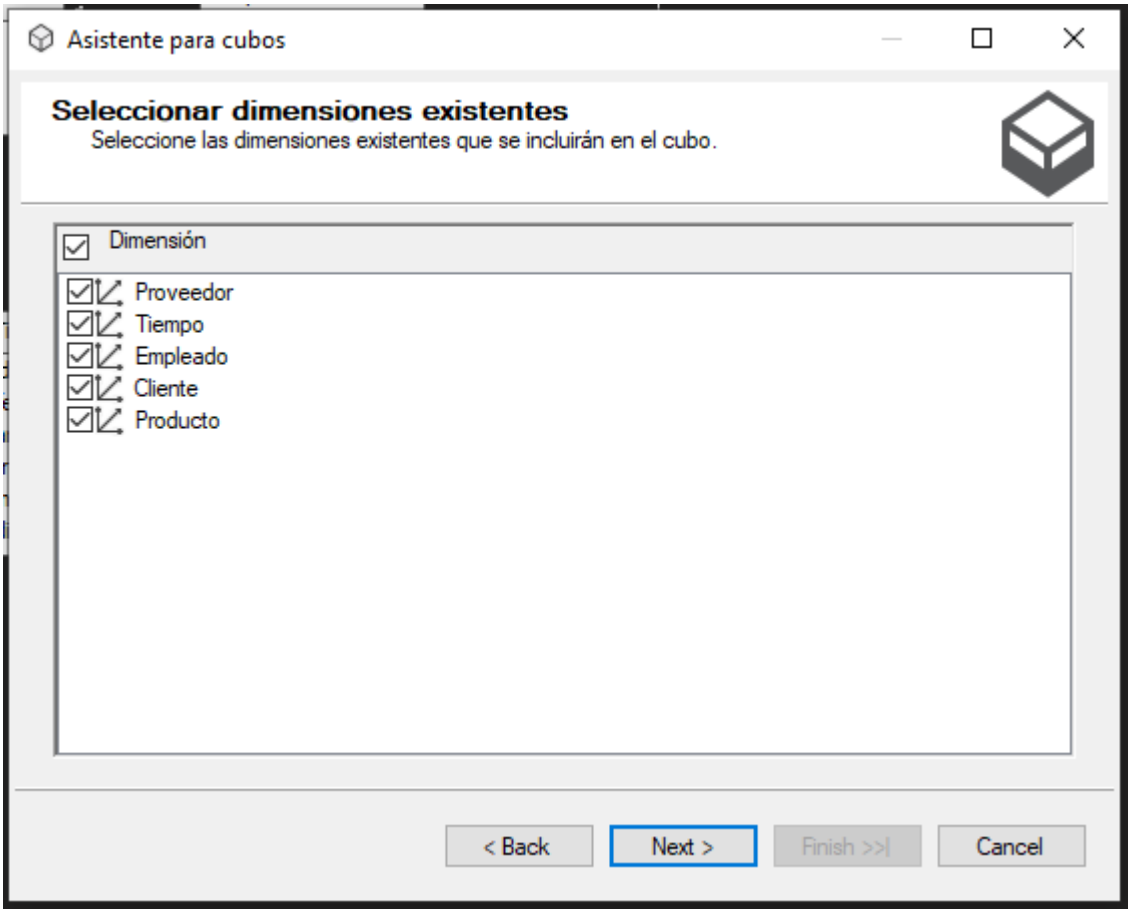


Luego creamos una vista para el origen de datos anterior seleccionando las tablas: Cliente, Empleado, Pedido, Producto, Proveedor y Tiempo; como se muestra a continuación:

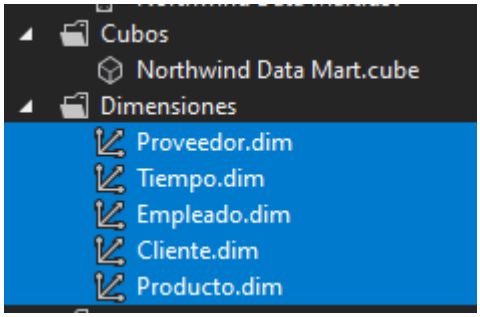


Luego procedemos a crear un nuevo cubo usando las tablas existentes y seleccionado la tabla “Pedido” como “tabla de grupo de medida”, el resto de tablas serán las dimensiones de nuestro cubo



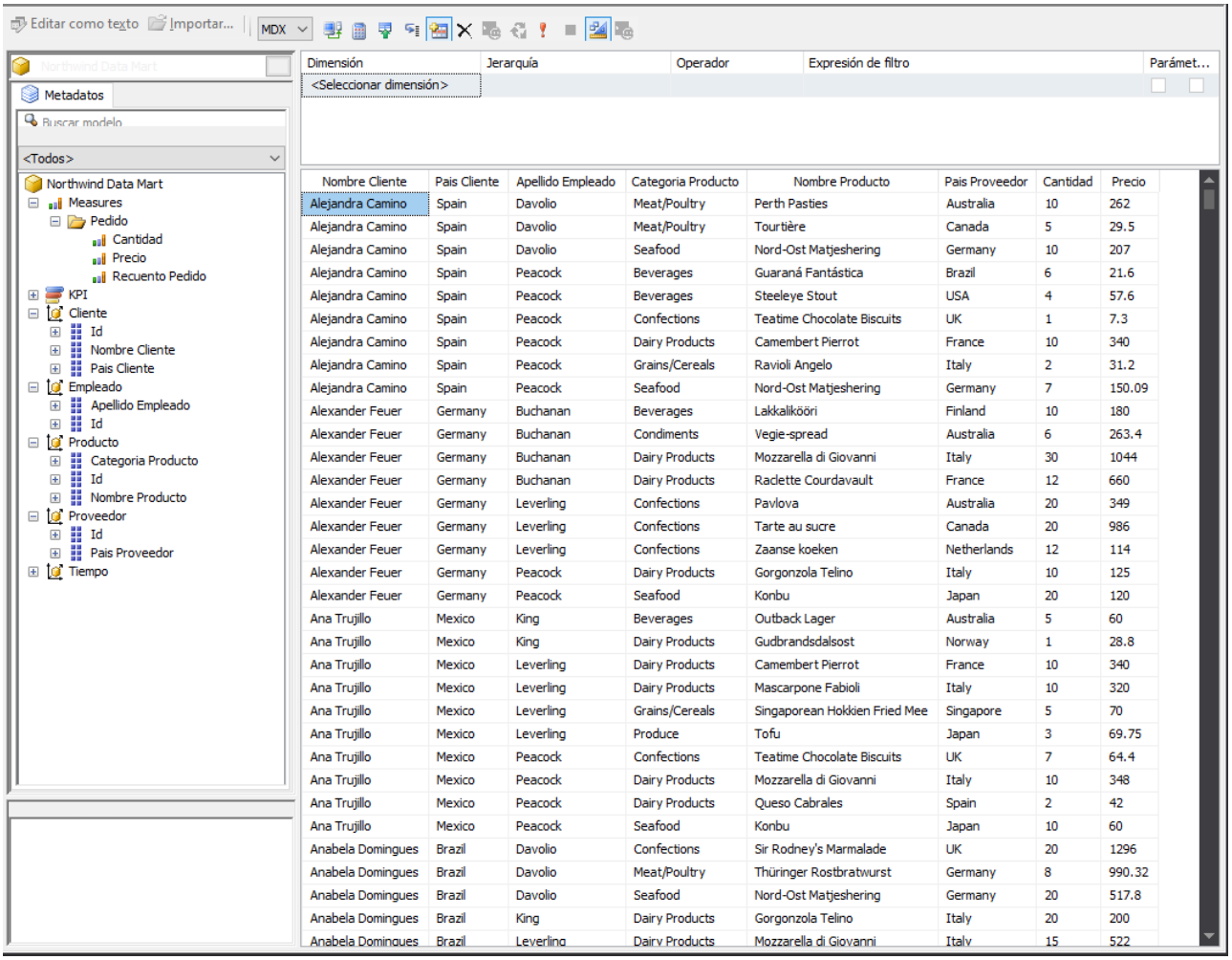


Una vez creado el cubo procedemos a ver que ya tenemos nuestras dimensiones creadas en su respectiva carpeta.



Luego de esto procesamos el cubo.

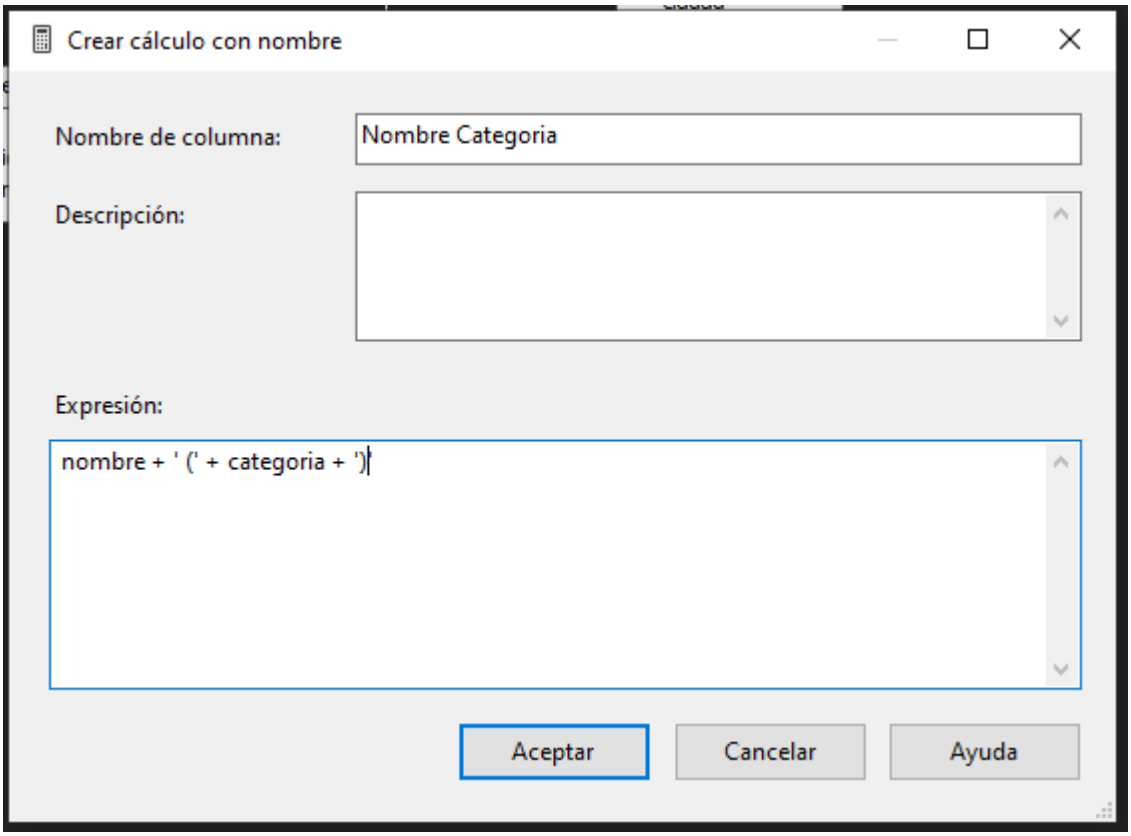
A continuacion procedemos a crear la primera consulta, para esto nos dirigimos a las dimensiones correspondientes y seleccionamos los datos requeridos, poniendole un nombre más descriptivo a cada campo para asi entender la consulta de mejor manera; posteriormente volvemos a procesar el cubo. El resultado se muestra a continuación:



Nombre Cliente	Pais Cliente	Apellido Empleado	Categoria Producto	Nombre Producto	Pais Proveedor	Cantidad	Precio
Alejandra Camino	Spain	Davolio	Meat/Poultry	Perth Pasties	Australia	10	262
Alejandra Camino	Spain	Davolio	Meat/Poultry	Tourtière	Canada	5	29.5
Alejandra Camino	Spain	Davolio	Seafood	Nord-Ost Matjeshering	Germany	10	207
Alejandra Camino	Spain	Peacock	Beverages	Guaraná Fantástica	Brazil	6	21.6
Alejandra Camino	Spain	Peacock	Beverages	Steeleye Stout	USA	4	57.6
Alejandra Camino	Spain	Peacock	Confections	Teatime Chocolate Biscuits	UK	1	7.3
Alejandra Camino	Spain	Peacock	Dairy Products	Camembert Pierrot	France	10	340
Alejandra Camino	Spain	Peacock	Grains/Cereals	Ravioli Angelo	Italy	2	31.2
Alejandra Camino	Spain	Peacock	Seafood	Nord-Ost Matjeshering	Germany	7	150.09
Alexander Feuer	Germany	Buchanan	Beverages	Lakkalikööri	Finland	10	180
Alexander Feuer	Germany	Buchanan	Condiments	Vegie-spread	Australia	6	263.4
Alexander Feuer	Germany	Buchanan	Dairy Products	Mozzarella di Giovanni	Italy	30	1044
Alexander Feuer	Germany	Buchanan	Dairy Products	Raclette Courdavault	France	12	660
Alexander Feuer	Germany	Leverling	Confections	Pavlova	Australia	20	349
Alexander Feuer	Germany	Leverling	Confections	Tarte au sucre	Canada	20	986
Alexander Feuer	Germany	Leverling	Confections	Zaanse koeken	Netherlands	12	114
Alexander Feuer	Germany	Peacock	Dairy Products	Gorgonzola Telino	Italy	10	125
Alexander Feuer	Germany	Peacock	Seafood	Konbu	Japan	20	120
Ana Trujillo	Mexico	King	Beverages	Outback Lager	Australia	5	60
Ana Trujillo	Mexico	King	Dairy Products	Gudbrandsdalsost	Norway	1	28.8
Ana Trujillo	Mexico	Leverling	Dairy Products	Camembert Pierrot	France	10	340
Ana Trujillo	Mexico	Leverling	Dairy Products	Mascarpone Fabioli	Italy	10	320
Ana Trujillo	Mexico	Leverling	Grains/Cereals	Singaporean Hokkien Fried Mee	Singapore	5	70
Ana Trujillo	Mexico	Leverling	Produce	Tofu	Japan	3	69.75
Ana Trujillo	Mexico	Peacock	Confections	Teatime Chocolate Biscuits	UK	7	64.4
Ana Trujillo	Mexico	Peacock	Dairy Products	Mozzarella di Giovanni	Italy	10	348
Ana Trujillo	Mexico	Peacock	Dairy Products	Queso Cabrales	Spain	2	42
Ana Trujillo	Mexico	Peacock	Seafood	Konbu	Japan	10	60
Anabela Domingues	Brazil	Davolio	Confections	Sir Rodney's Marmalade	UK	20	1296
Anabela Domingues	Brazil	Davolio	Meat/Poultry	Thüringer Rostbratwurst	Germany	8	990.32
Anabela Domingues	Brazil	Davolio	Seafood	Nord-Ost Matjeshering	Germany	20	517.8
Anabela Domingues	Brazil	King	Dairy Products	Gorgonzola Telino	Italy	20	200
Anabela Domingues	Brazil	Leverling	Dairy Products	Mozzarella di Giovanni	Italy	15	522

En el panel de la izquierda se pueden ver los campos que agregamos a cada dimension para poder realizar la consulta y como medida tomamos los datos: precio y cantidad de la tabla “pedidos”.

Para finalizar este ejercicio procedemos a crear un nuevo campo, concatenando el nombre y la categoria del producto. Primero nos dirigimos a la vista del origen de datos y le damos doble click. En el panel tabla procedemos a darle click izquiedo a la tabla “Producto” y seleccionamos la opción: “Nuevo cálculo con nombre” y procedemos a llenar el formulacio de la siguiente forma:



Luego de esto volvemos a darle click izquierdo a la tabla “Producto” y seleccionamos la opcion de “Explorar datos” y ya podemos ver nuestra nueva columna:

Tabla

id	nombre	categoria	Nombre Categoria
1	Alice Mutton	Meat/Poultry	Alice Mutton (Meat/Poultry)
2	Aniseed Syrup	Condiments	Aniseed Syrup (Condiments)
3	Boston Crab Meat	Seafood	Boston Crab Meat (Seafood)
4	Camembert Pierrot	Dairy Products	Camembert Pierrot (Dairy Products)
5	Carnarvon Tigers	Seafood	Carnarvon Tigers (Seafood)
6	Chai	Beverages	Chai (Beverages)
7	Chang	Beverages	Chang (Beverages)
8	Chartreuse verte	Beverages	Chartreuse verte (Beverages)
9	Chef Anton's Cajun Seasoning	Condiments	Chef Anton's Cajun Seasoning (Condiments)
10	Chef Anton's Gumbo Mix	Condiments	Chef Anton's Gumbo Mix (Condiments)
11	Chocolade	Confections	Chocolade (Confections)
12	Côte de Blaye	Beverages	Côte de Blaye (Beverages)

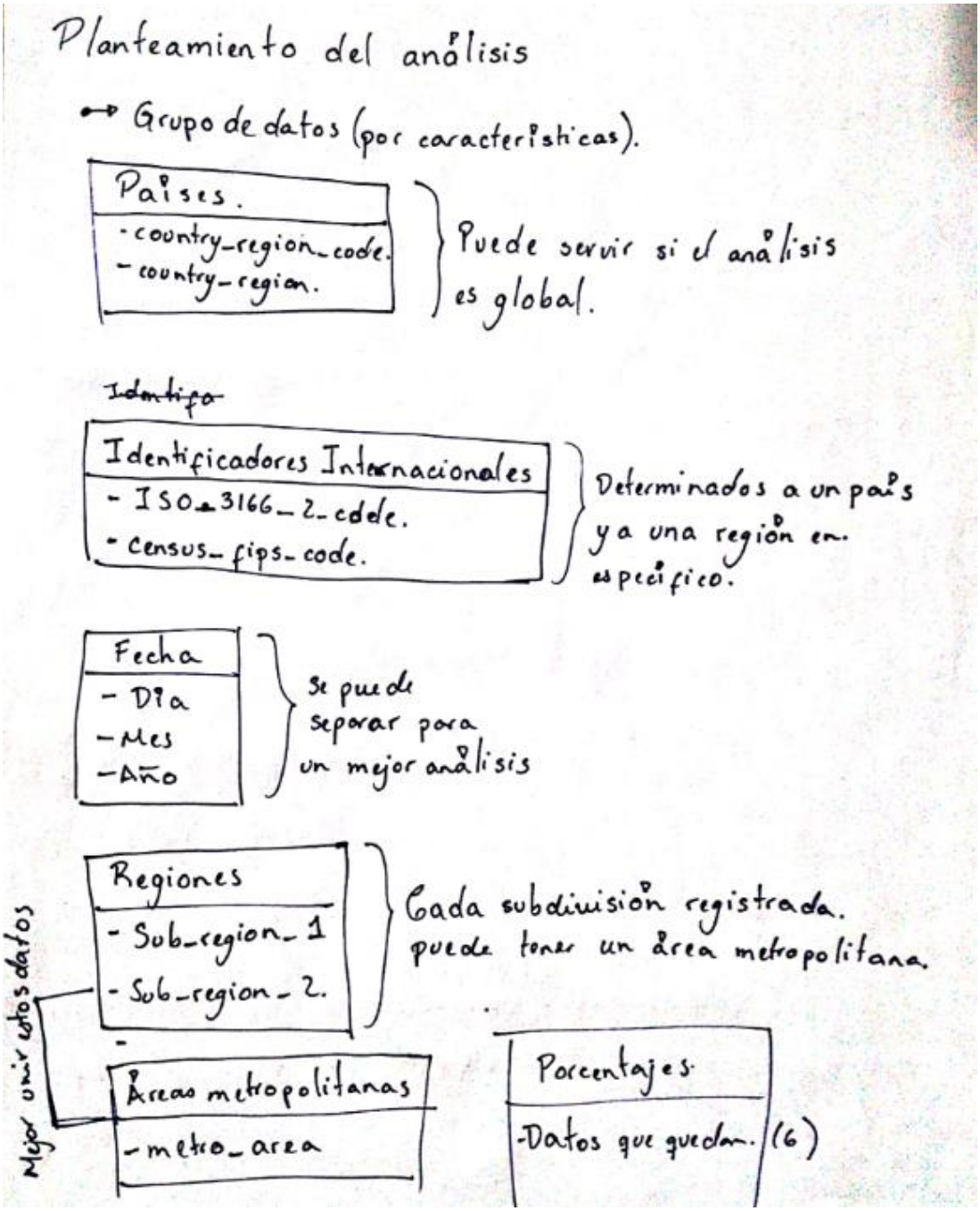
Como observamos primero tenemos el nombre del producto, seguido de la categoría entre parentesis.

EJERCICIO 2

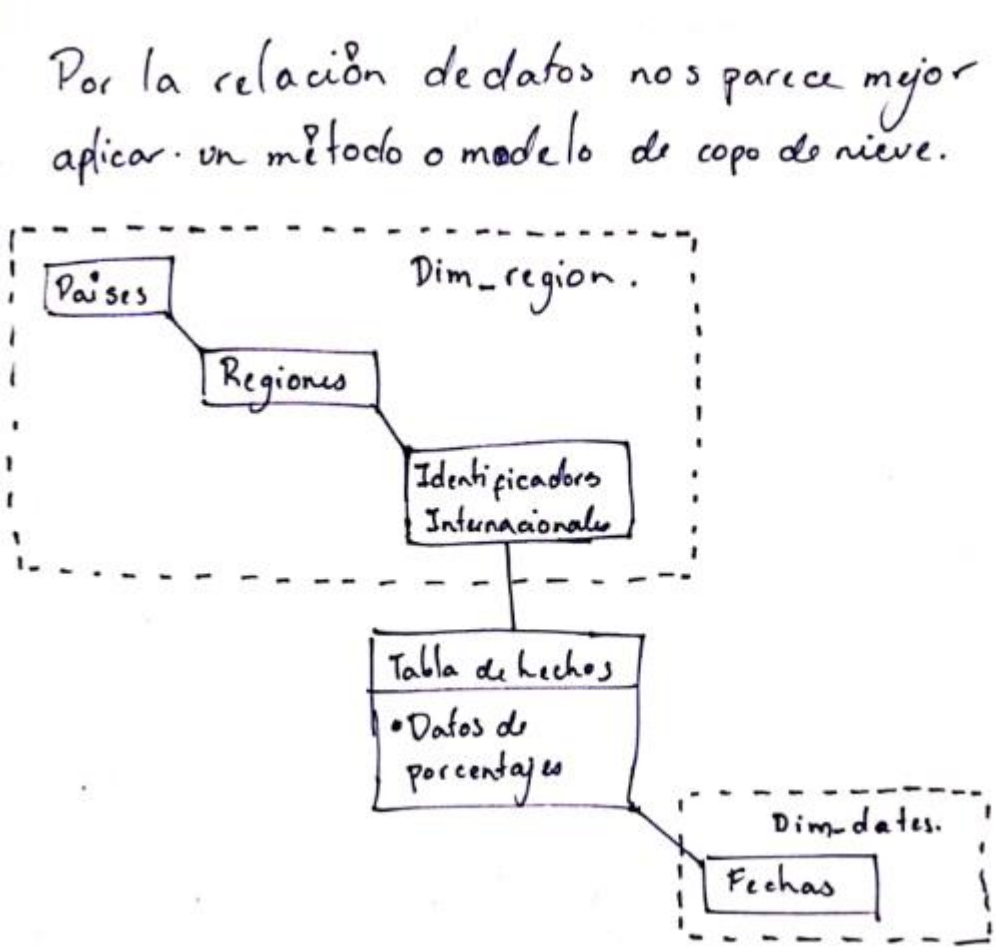
El planteamiento de este ejercicio era utilizar los datos brindados por Google acerca del porcentaje de movilidad de personas en zonas recurrentes marcadas por fecha y zona geográfica.

De este se utilizaría solamente un archivo, para este ejercicio, se ha utilizado el archivo correspondiente para Nicaragua.

Para comenzar con el planteamiento del problema y el análisis de los datos, se dio inicio con el ETL correspondiente para guardar los datos en una Base de datos relacional en SQL Server, el planteamiento fue el siguiente:



Para esta separación de los datos, se pudo acomodar en dos dimensiones concretas y una dimensión extra que no estaba del todo claro, esta sería la dimensión de lugares de movilización, pero no se terminó de integrar esta dimensión porque aun siendo nombre de columna su nombre no representaba una descripción o valor de una dimensión, es decir, tenía que ser añadida manualmente, por lo que terminó por ser parte de la tabla de hechos, la distribución planeada fue la siguiente:

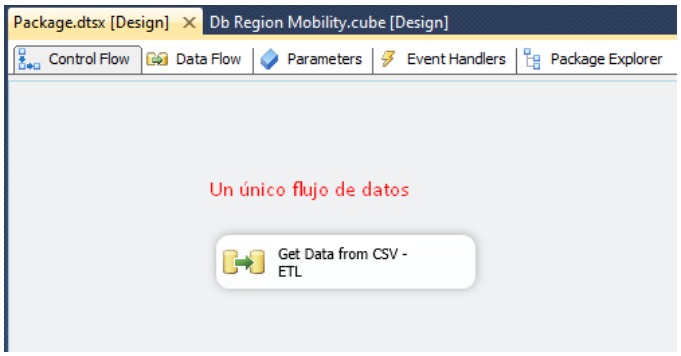


Como se menciona en el planteamiento, la relación más efectiva que se encontró ha sido el modelo de tipo “SnowFlake” (Copo de nieve), sin obtener más dimensiones innecesarias, pero sin perder datos que son relevantes y descriptivos de las dos macro dimensiones. Luego de ello se procedió a la realización del ETL.

Realización de ETL

1. Creación del flujo de datos

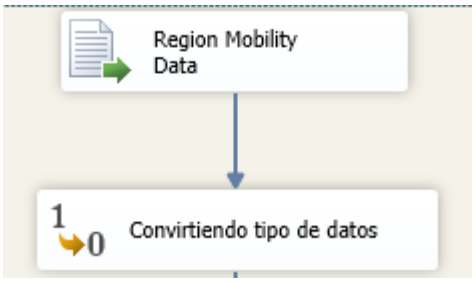
   A pesar de ser un diagrama de múltiples tablas relacionadas entre sí, muchos optan por realizar varios flujos de datos, pero esto suele consumir muchos recursos de la computadora, en un servidor esto no tiene mucha relevancia, pero siempre hay que pensar en el rendimiento y eficacia del modelo por ello el proceso se resumió en un solo flujo de datos.



Teniendo un único flujo de datos la función que describe el uso de datos siempre sería líneal (N slots de memoria) para (N datos en el archivo), mientras que al ejecutar de manera paralela múltiples flujos de datos, habría N^m slots de memoria utilizándose al mismo tiempo, donde N representa la cantidad de datos y "m" la cantidad de flujos de datos. Y en caso se ejecutarán uno por uno, no sería un proceso automatizado, y el tiempo de ejecución no sería computarizado, sino que dependería de quien esté realizando el proceso.

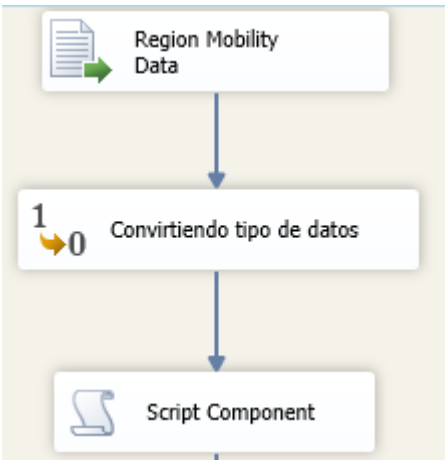
2. Extracción de los datos del archivo .csv

   En esta parte se extraen los datos para un archivo de texto plano, esto porque el archivo .csv es un simple archivo de texto, que representa relación de valores como una tabla por medio de la separación de "comas".



En esta parte extraemos los datos y convertimos algunos tipos de datos en otros, para acomodarlos desde un inicio y no tener que hacerlo después por separado, no entraremos en detalle en las configuraciones porque el tema principal del ejercicio es la utilización de las herramientas de análisis de servicios.

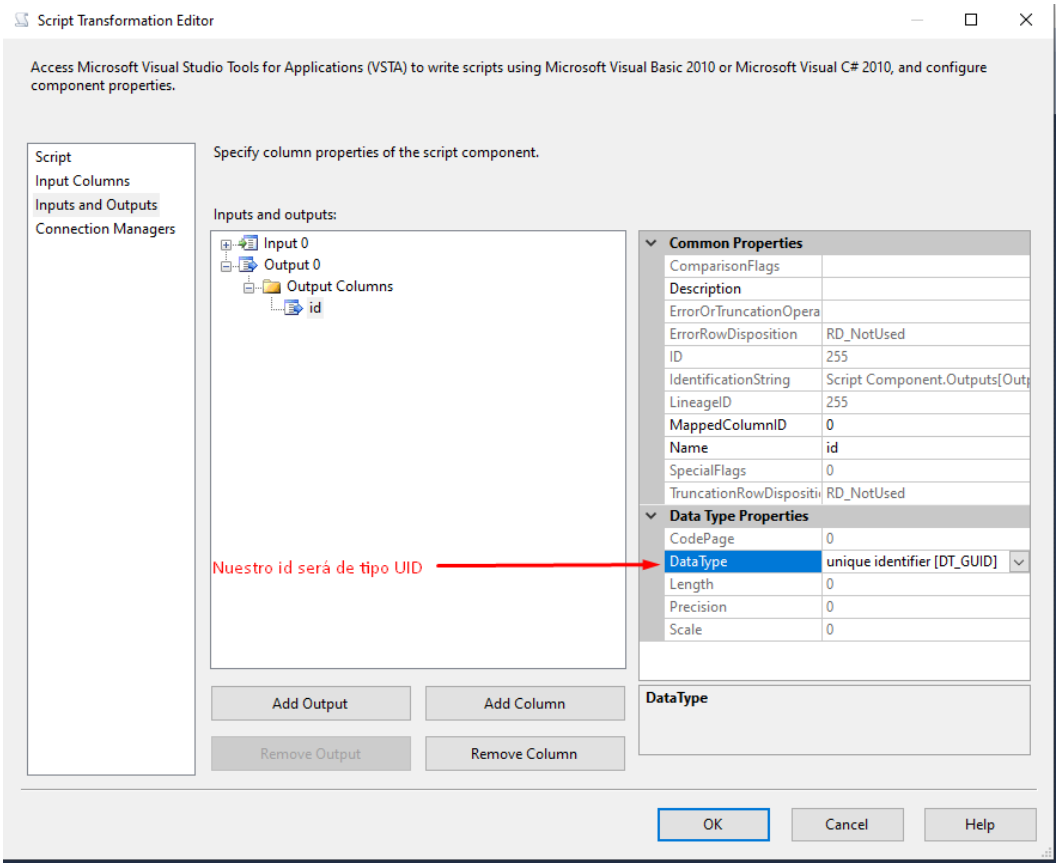
3. Agregar un identificador a cada registro



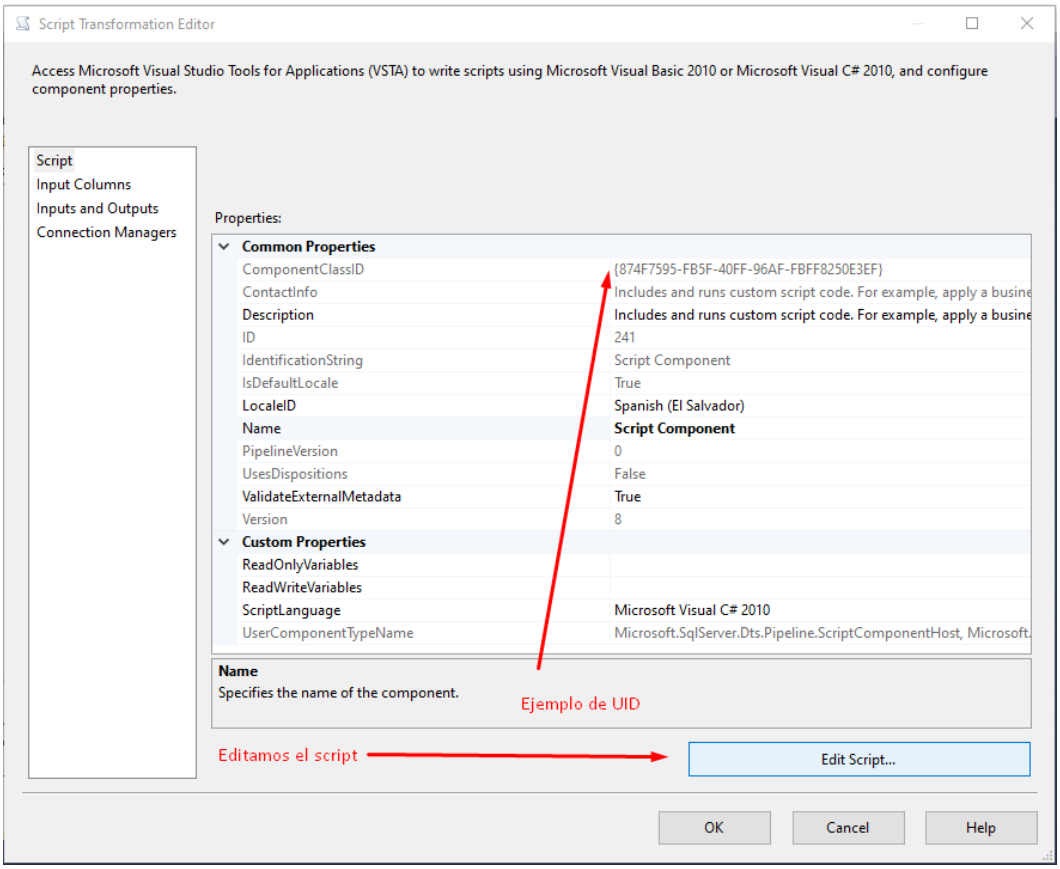
Luego de la conversión de datos, se añade un identificador por medio de un script, que contiene código C#, esto como referencia a cada fila e importante para las relaciones entre tablas.

En este script es posible añadir más columnas de datos a la salida de este componente, para este caso particular se añade la columna "id" y dentro del código se genera un Unique Identifier para que sea un identificador largo, irrepetible y estándar.

De la siguiente manera se añade la nueva columna a la salida del componente (También es posible determinar más salidas)



Edición del script y ejemplo de uid:

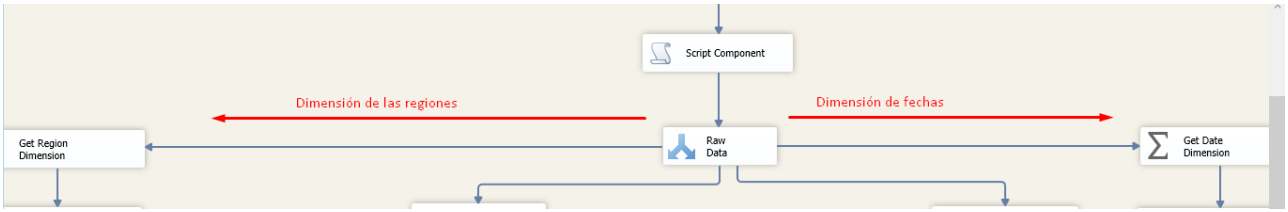


Código del script:

```
main.cs*
ScriptMain                                  Input0_ProcessInputRow(Input0Buffer Row)

    /// <summary>
    /// This method is called once for every row that passes through the component from Input0.
    ///
    /// Example of reading a value from a column in the the row:
    ///  string zipCode = Row.ZipCode
    ///
    /// Example of writing a value to a column in the row:
    ///  Row.ZipCode = zipCode
    /// </summary>
    /// <param name="Row">The row that is currently passing through the component</param>
    public override void Input0_ProcessInputRow(Input0Buffer Row)
    {
        /*
         * Generate UID for new column id
         */
        Row.id = Guid.NewGuid();

    }

}
```

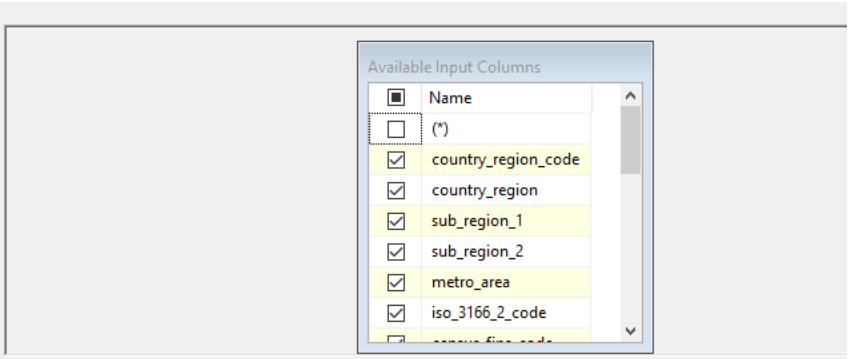
4. Obtener las dimensiones de los registros

   En esta parte comienza la separación de los datos en las respectivas tablas que hemos visto, como hay múltiples divisiones, los datos obtenidos del script se pasan por un multicast, lo cual duplicará los datos cuantas veces sea necesario:



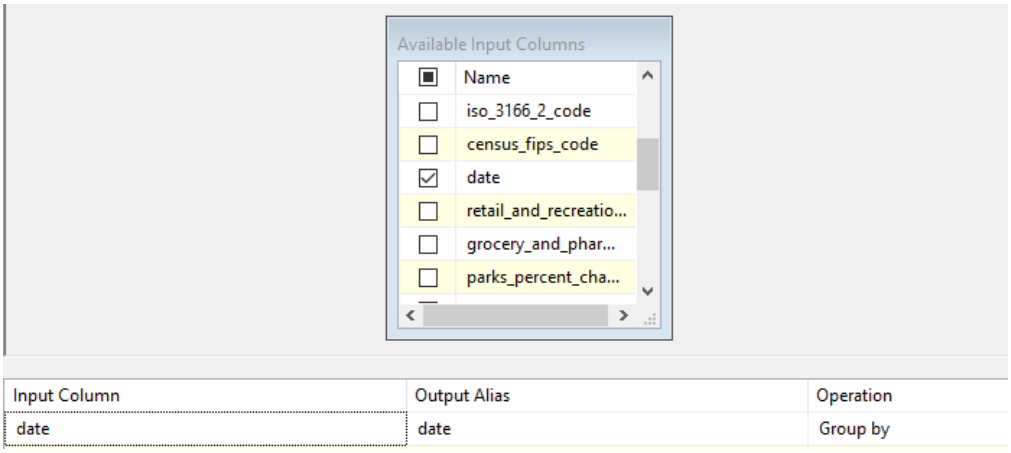
Como se puede ver hay cuatro divisiones, pero las dos principales son las superiores, las inferiores luego se serán explicadas.

Para obtener los datos de la región lo que se hace es agrupar todos los registros por cada combinación diferente en base a las columnas pertenecientes a la dimensión, esto por medio del componente de agrupamiento:



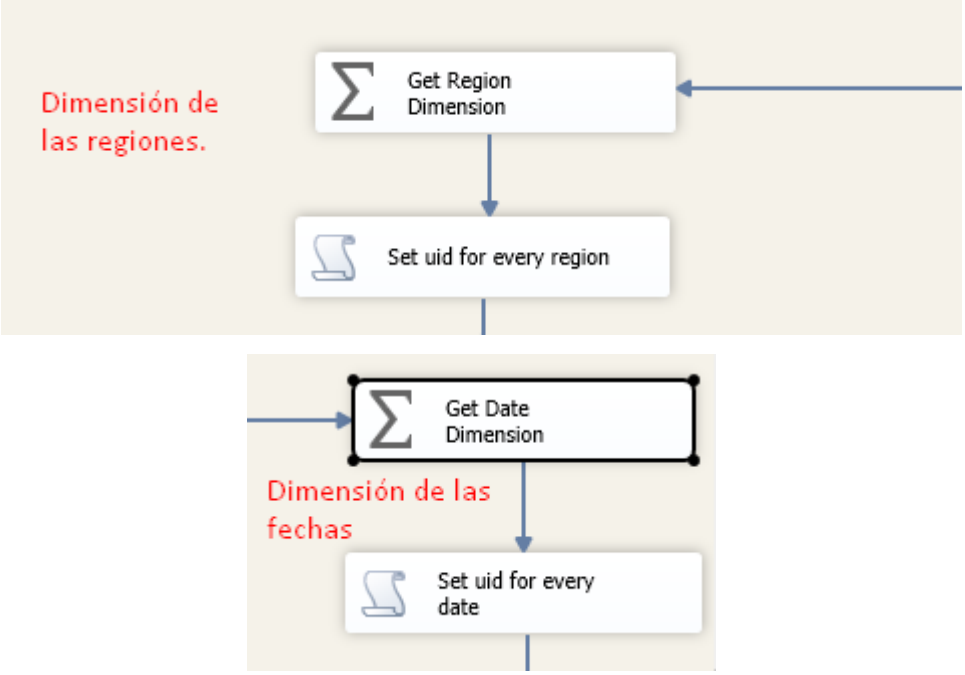
Input Column	Output Alias	Operation
country_region_code	country_region_code	Group by
country_region	country_region	Group by
sub_region_1	sub_region_1	Group by
sub_region_2	sub_region_2	Group by
metro_area	metro_area	Group by
iso_3166_2_code	iso_3166_2_code	Group by
census_fips_code	census_fips_code	Group by

E igualmente para la dimensión de las fechas:



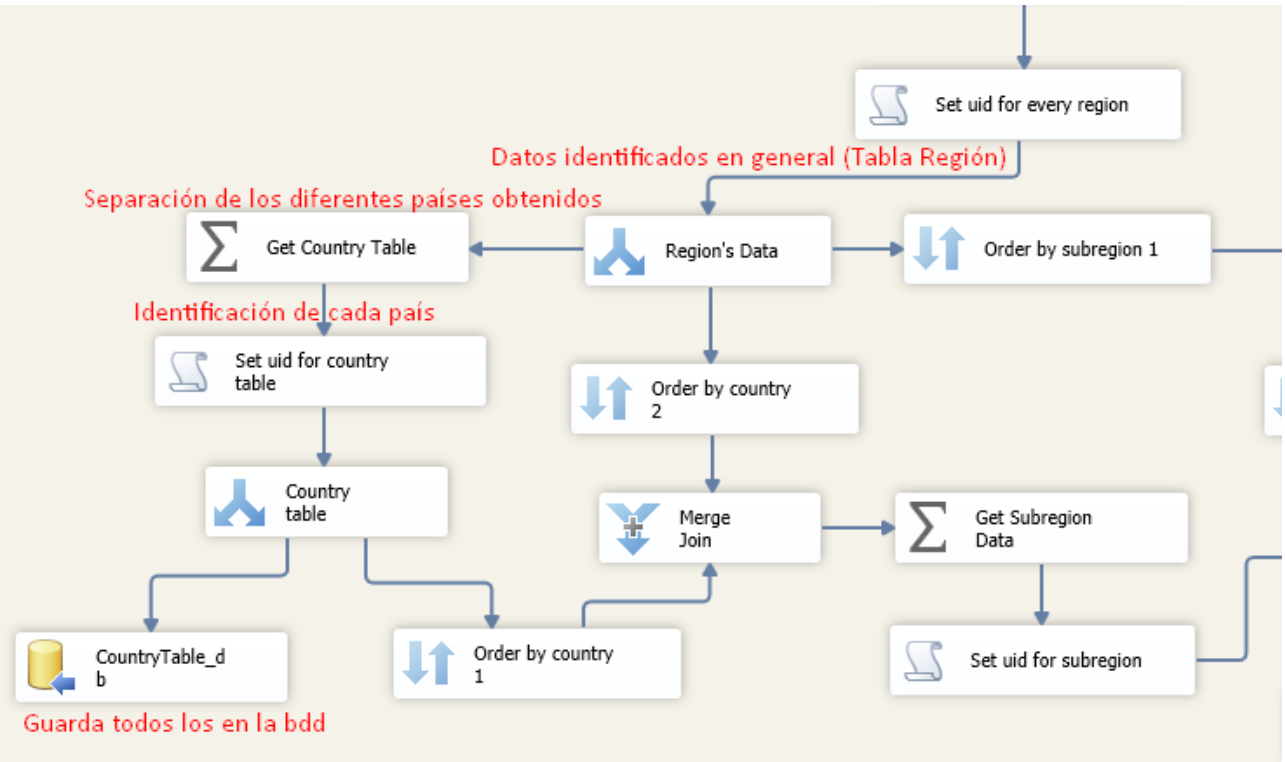
5. Identificar registros de cada dimensión

   Como se hizo anteriormente con cada registro en general, se realizará igualmente para ambas dimensiones, por lo que la estructura queda de la siguiente manera:



6. Repetir proceso para sub-tablas

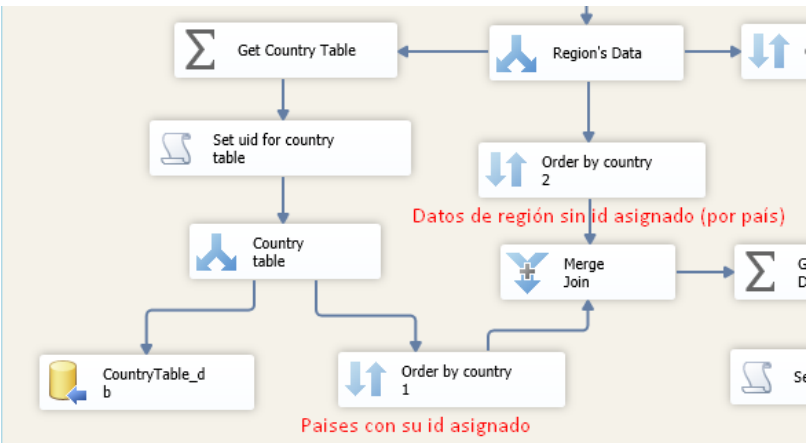
   Este paso se tuvo que realizar por el modelo copo de nieve que se implementó, por lo qué luego de identificar cada región en general, ahora se divide en cada país (Tabla primaria) y en cada subregión (Tabla intermedio) y luego se procede a identificarlos como se explicó anteriormente:



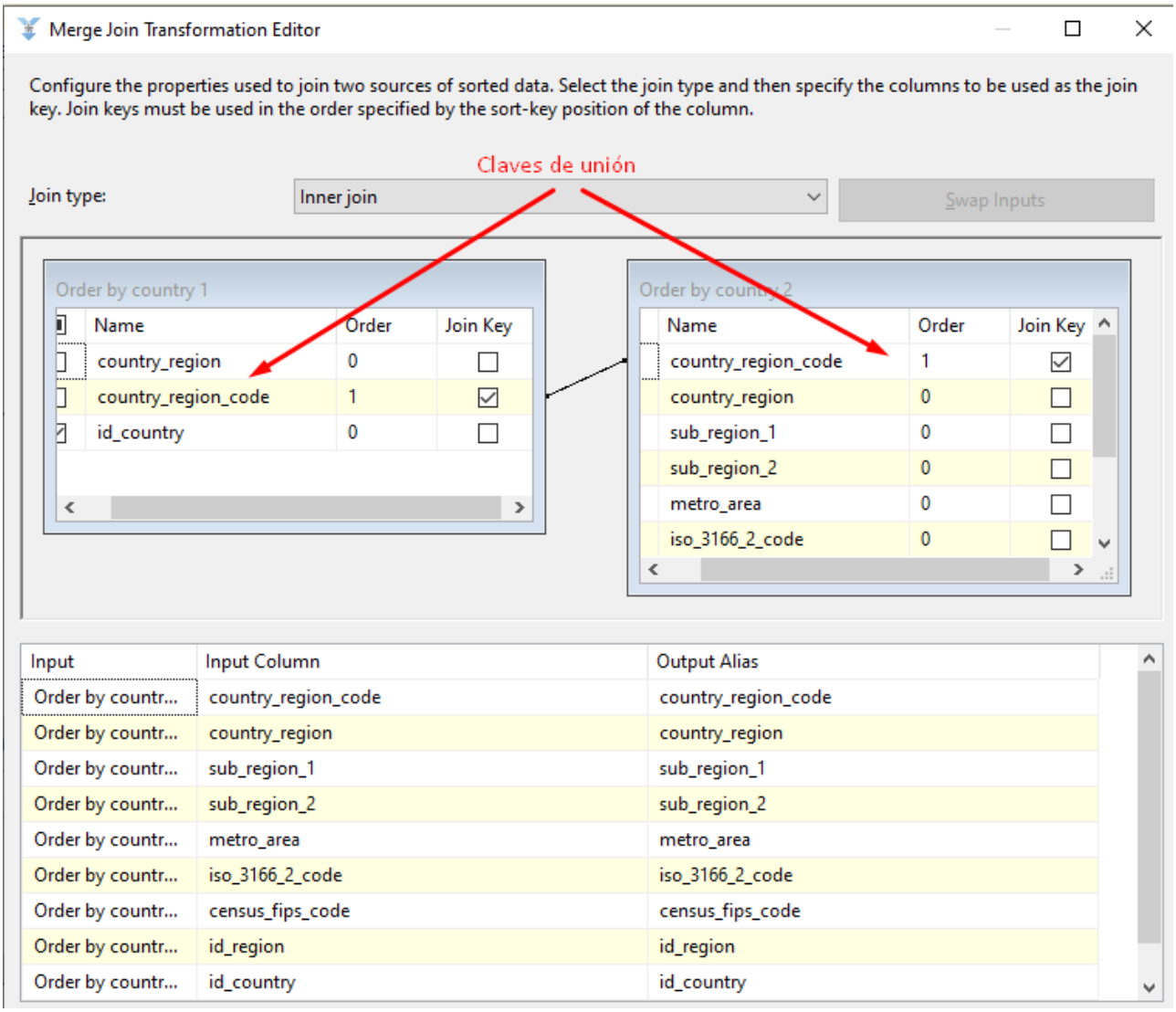
En la imagen anterior se detalla la obtención de los diferentes países, este ETL ha sido pensado para el procesamiento de múltiples datos, sin importar el país, por lo que, si al inicio de todo el ETL juntamos los demás datos, el ETL se encargará sin problemas de relacionarlos todos en la misma base de datos.

7. Relacionar los datos

Una vez se han guardado los datos primarios (Tabla de países), es decir la tabla por la que nace la relación, es momento de relacionar este dato a la siguiente tabla, en este caso a la tabla de Sub Regiones, para ello se realiza un Merge Join, el cual nos permitirá hacer un **"Inner join"** de los datos,y de esta manera asignar a cada país su id respectivo, para ello como en todo **"inner join"** es necesario tener una clave para relacionarlos,esta clave son las columnas por la que dividimos los datos, en el caso de país la clave de unión es el nombre de país y el código asignado de la siguiente manera:

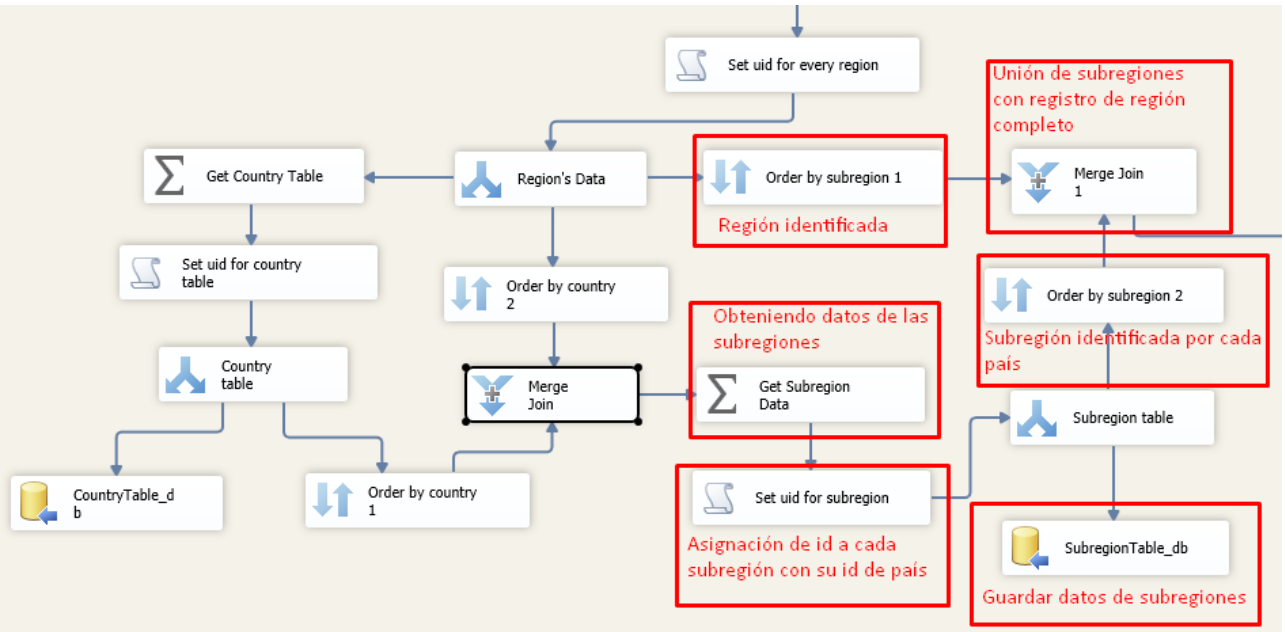


En la imagen anterior se puede ver que hay componentes de ordenamiento, y como su nombre lo indican ambos ordenan los datos por país y código país, de esta manera se le indica al componente **"Merge Join"** que las columnas por las que se han ordenado esas serán las claves de unión:



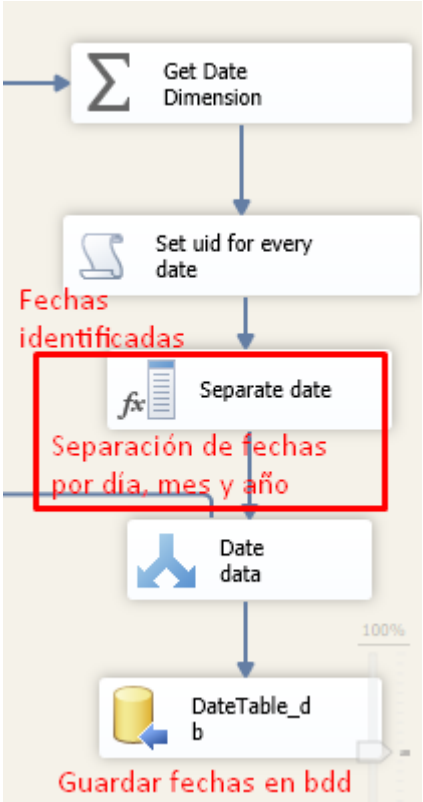
La imagen anterior muestra cómo se relacionan las tablas (en el ETL) y de esta manera ahora ya se puede asignar el código de país a cada registro de la región, puede verse en la parte inferior de la imagen.

Ahora se repiten los datos para la tabla intermedio



Ahora en el componente **"Merge Join 1"** se obtiene cada región identificada y relacionada a cada subregión y país con su id respectivo.

Y de la misma manera se aplica para la dimensión de país:



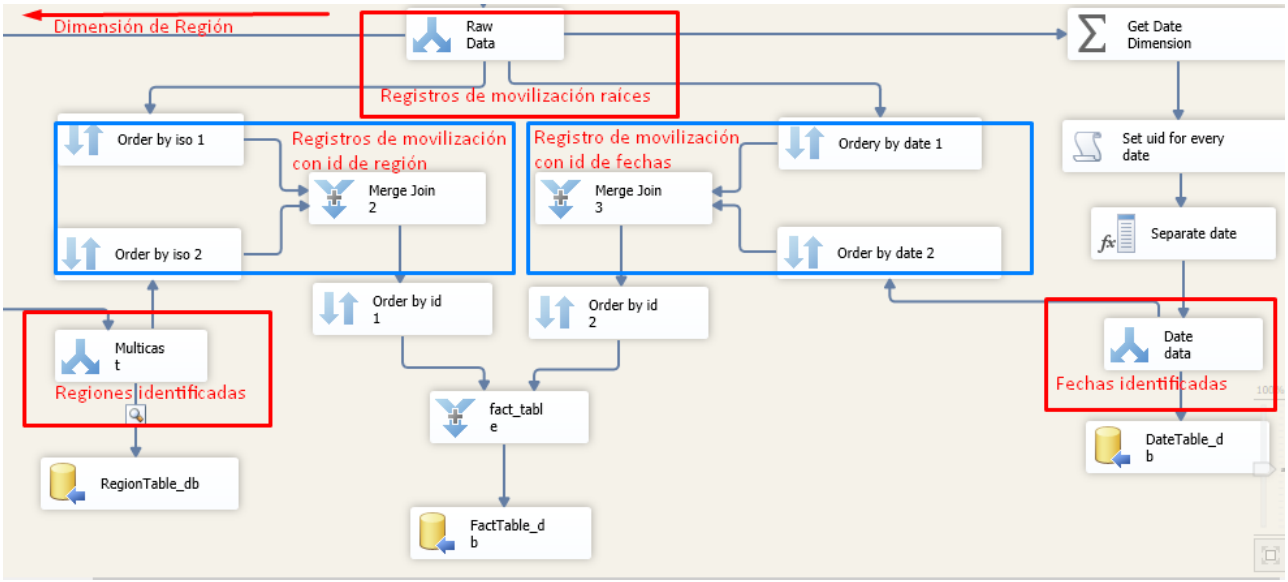
La única diferencia ahora es que se separan las fechas por día, mes y año para tener un análisis más detallado de los registros y luego se guarda en la base de datos.

Esta estructura es mucho más fácil de analizar esto debido a que es una sola tabla, no se divide en más y por lo tanto no hay que dividir más los datos, se podría dividir en meses y días, pero al ser un ejercicio no se subdividió más.

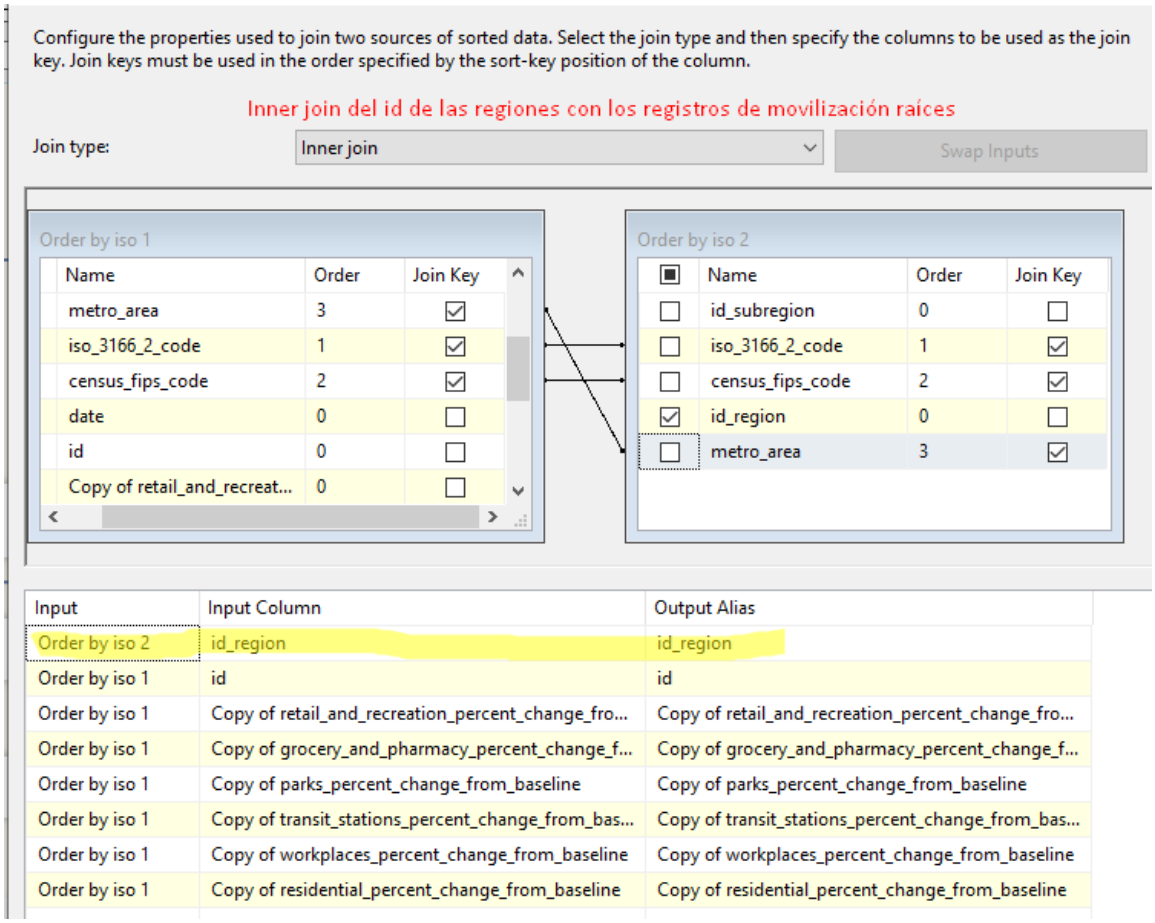
8. Juntar las dimensiones a los datos raíz

   Este paso es similar a los que vimos en la unión de las subtablas, al ser 2 dimensiones los datos raíz deberán ser dividos y ordenados respecto a cada clave de unión de las dimensiones, en el caso de la dimensión de región las claves de unión serán las mismas columnas por las que se separó, en ese caso sería

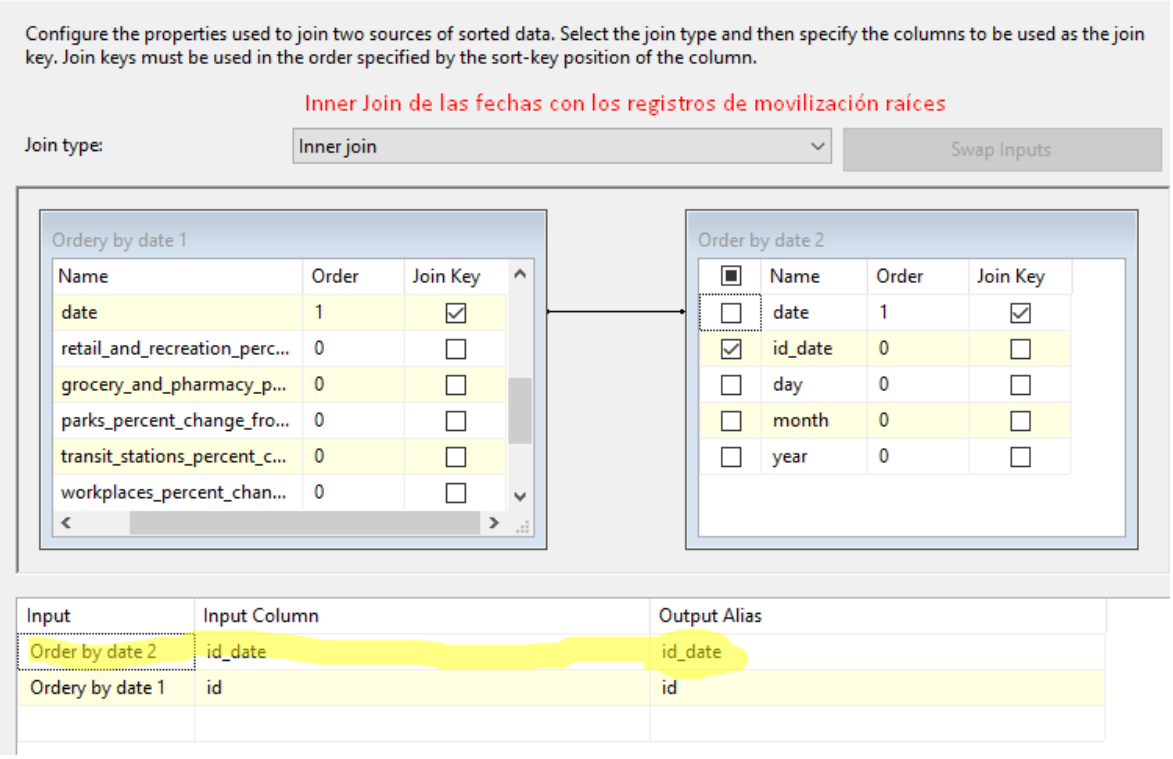
country_region_code, country_region, sub_region_1, sub_region_2, metro_area, iso_3166_2_code y census fips_code. En el caso de la dimensión de fechas la única clave de unión será la fecha en sí.



La imagen anterior muestra el flujo de datos después de los pasos mostrados, a continuación, se muestran los **"inner join"** y sus claves de unión:



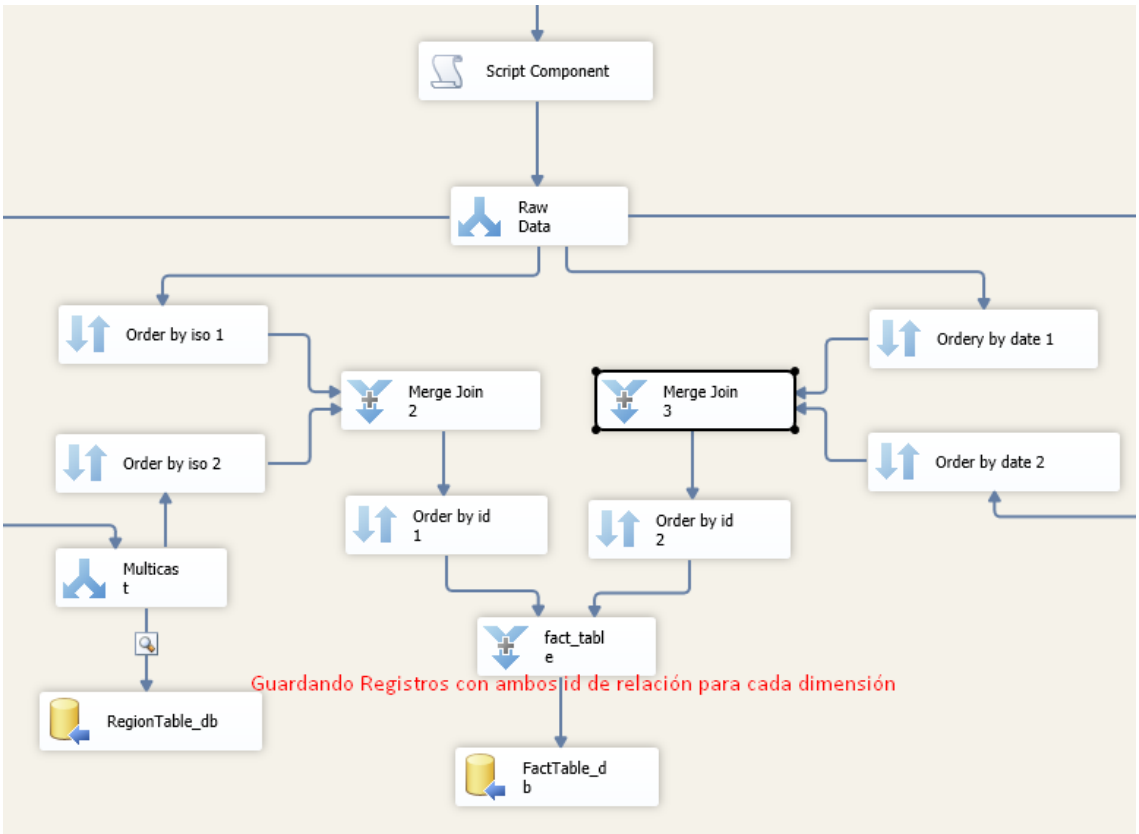
Como se puede observar en la imagen anterior se muestran las claves de unión y subrayado el motivo del porqué el inner join, añadir al registro el id de la región.

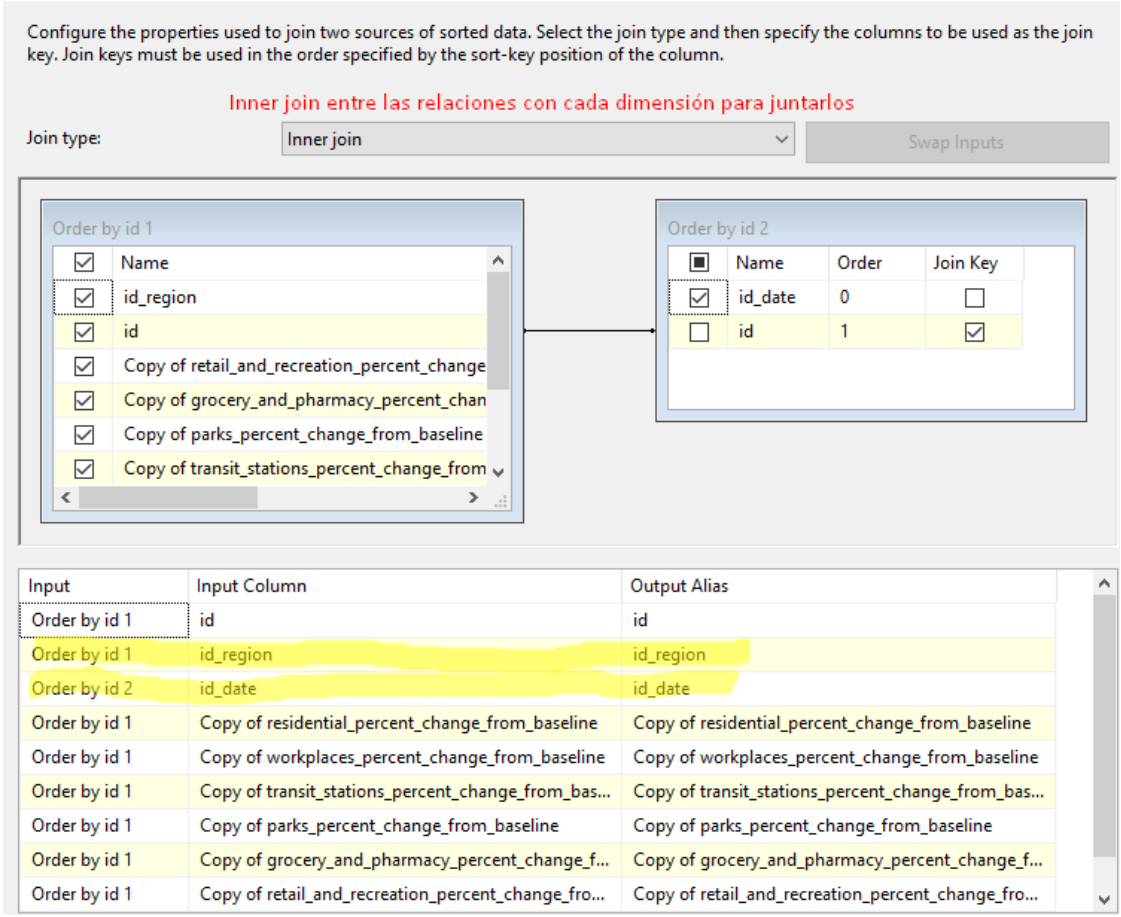


En la imagen anterior se puede ver la unión entre los registros raíces y la dimensión de fecha, y abajo el id de la fecha de la relación, en este caso ya no se agregan los datos de los porcentajes porque se añaden en la unión anterior.

9. Unir los registros con su relación separada, en una sola para obtener la tabla de hechos

   Esta ya es la parte final del ETL, ya se tienen los datos de los porcentajes (Tabla de Hechos) y sus id de relación a las dos dimensiones principales y como están por separadas, se realiza una vez más el **“inner join”**, en este caso ordenando por el id principal.





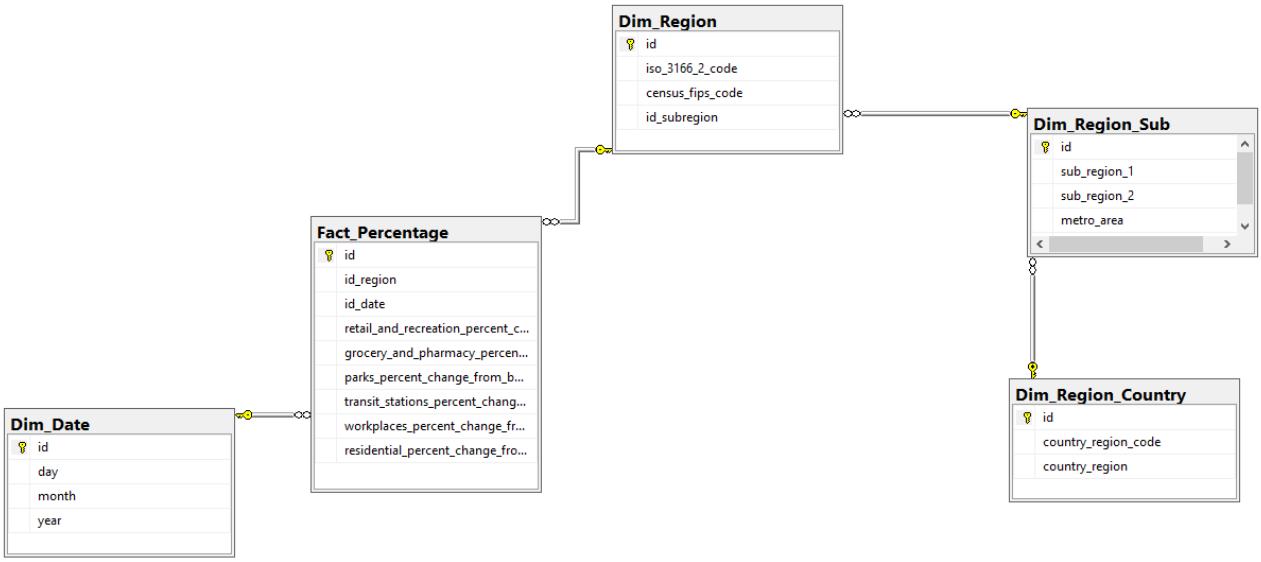
En la imagen anterior se puede ver el **“inner join”** entre los anteriores para juntar los id de relación.

Creación de base de datos relacional

Se necesita crear la base de datos para que el ETL pueda guardar la información, la creación de la base de datos está dada por el siguiente SQL:

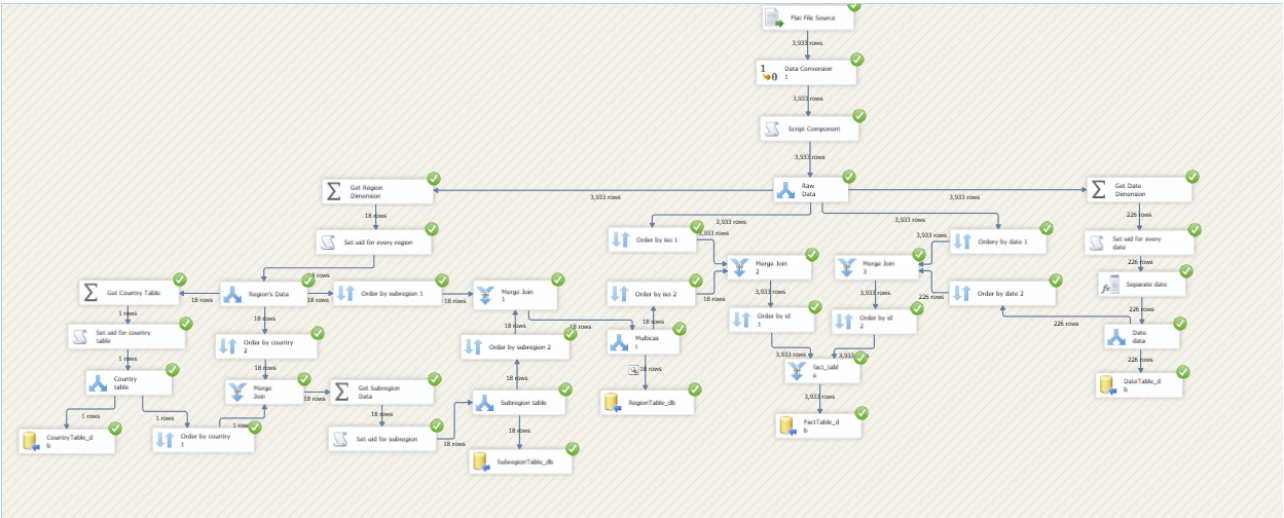
```sql
use db_region_mobility;

create table Dim_Region_Country (
       id uniqueidentifier primary key,
       country_region_code char(2),
       country_region varchar(100)
) ;

create table Dim_Region_Sub (
       id uniqueidentifier primary key,
       sub_region_1 varchar(100),
       sub_region_2 varchar(100),
       metro_area varchar(100),
       id_country uniqueidentifier foreign key references Dim_Region_Country(id)
);

create table Dim_Region (
       id uniqueidentifier primary key,
       iso_3166_2_code varchar(10),
       census_fips_code varchar(20),
       id_subregion uniqueidentifier foreign key references Dim_Region_Sub(id)
);

create table Dim_Date (
       id uniqueidentifier primary key,
       "day" int,
       "month" int,
       "year" int
);

create table Fact_Percentage (
       id uniqueidentifier primary key,
       id_region uniqueidentifier foreign key references Dim_Region(id),
       id_date uniqueidentifier foreign key references Dim_Date(id),
       retail_and_recreation_percent_change_from_baseline int,
       grocery_and_pharmacy_percent_change_from_baseline int,
       parks_percent_change_from_baseline int,
       transit_stations_percent_change_from_baseline int,
       workplaces_percent_change_from_baseline int,
       residential_percent_change_from_baseline int
);
```

Al ejecutar la consulta se puede observar el diagrama de la base de datos:



Resultados de ETL

Se procede a ejecutar el ETL para ver el resultado:



Y se puede ver cómo funciona al cien por ciento, luego para terminar la comprobación en SQL SERVER se ha preparado la siguiente consulta:

```
select drc.country_region_code, drc.country_region, drs.sub_region_1, drs.sub_region_2,
drs.metro_area,dr.iso_3166_2_code, dr.census_fips_code, dd.year, dd.month, dd.day,
retail_and_recreation_percent_change_from_baseline,
grocery_and_pharmacy_percent_change_from_baseline, parks_percent_change_from_baseline,
transit_stations_percent_change_from_baseline, workplaces_percent_change_from_baseline,
residential_percent_change_from_baseline from Fact_Percentage fp
join Dim_Date dd on fp.id_date = dd.id
join Dim_Region dr on fp.id_region = dr.id
join Dim_Region_Sub drs on dr.id_subregion = drs.id
join Dim_Region_Country drc on drs.id_country = drc.id
order by drs.sub_region_1, drs.sub_region_2, drs.metro_area,  dd.month,dd.day, dd.year
```

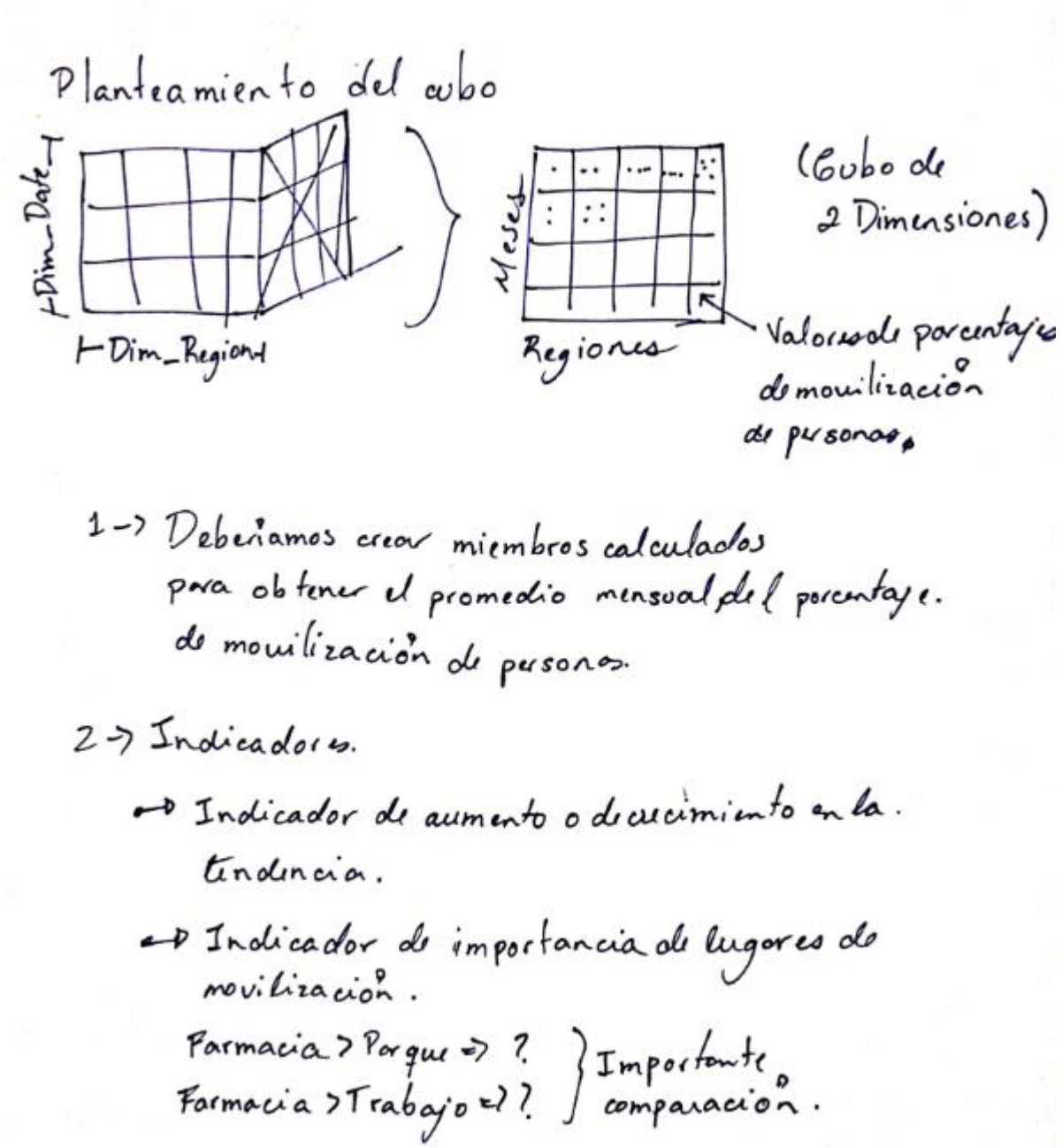
Esta consulta nos dará como resultado el archivo .csv del inicio:

Results | Messages

	country_region_code	country_region	sub_region_1	sub_region_2	metro_area	iso_3166_2_code	census_fips_code	year	month	day	retail_and_recreation_percent_change_from_baseline	grocery_and_pharmacy_percent_change_from_ba
1	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	2	15	6	5
2	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	2	16	4	7
3	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	2	17	2	5
4	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	2	18	0	0
5	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	2	19	-5	-5
6	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	2	20	-1	-2
7	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	2	21	2	0
8	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	2	22	0	0
9	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	2	23	-1	-1
10	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	2	24	-1	0
11	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	2	25	-4	-3
12	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	2	26	-5	-6
13	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	2	27	-3	-4
14	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	2	28	4	2
15	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	2	29	4	7
16	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	3	1	7	7
17	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	3	2	1	5
18	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	3	3	0	2
19	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	3	4	-4	-2
20	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	3	5	1	2
21	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	3	6	2	4
22	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	3	7	2	4
23	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	3	8	2	4
24	NI	Nicaragua	NULL	NULL	NULL	NULL	NULL	2020	3	9	-2	3

Planteamiento del Cubo

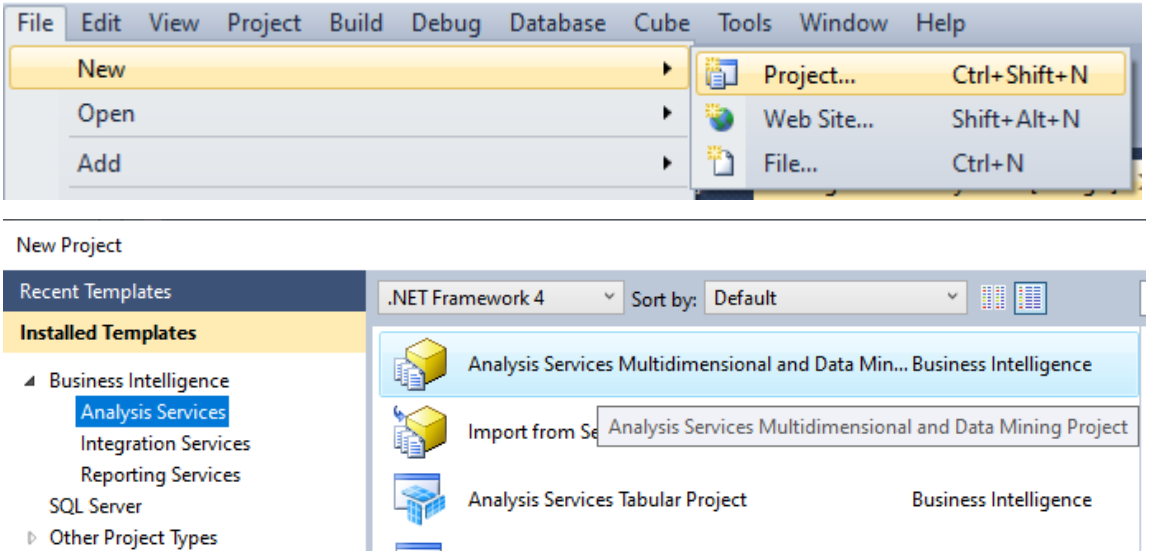
El planteamiento del cubo final está dado en la siguiente imagen:



Realización del cubo

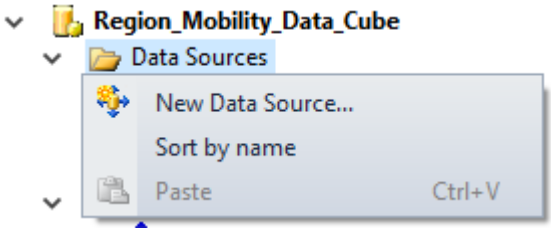
1. Añadir proyecto de análisis de servicios

   En la misma solución se debe agregar un nuevo proyecto de análisis de servicios



2. Creación del DataSource

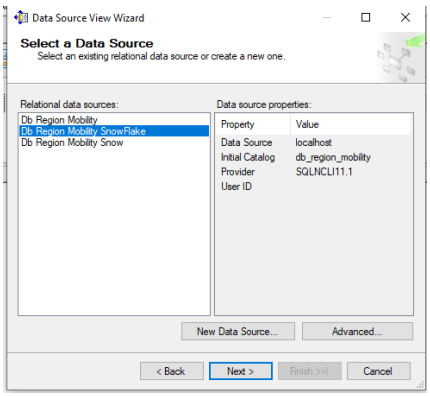
   En primera instancia es necesario definir el origen de datos del cual se va a extraer la información necesaria para su correspondiente análisis

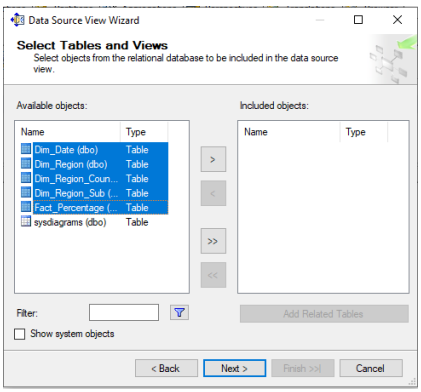


   Luego utilizar la conexión existente hacia la base de datos en la que se guarda la información recopilada en el archivo .csv

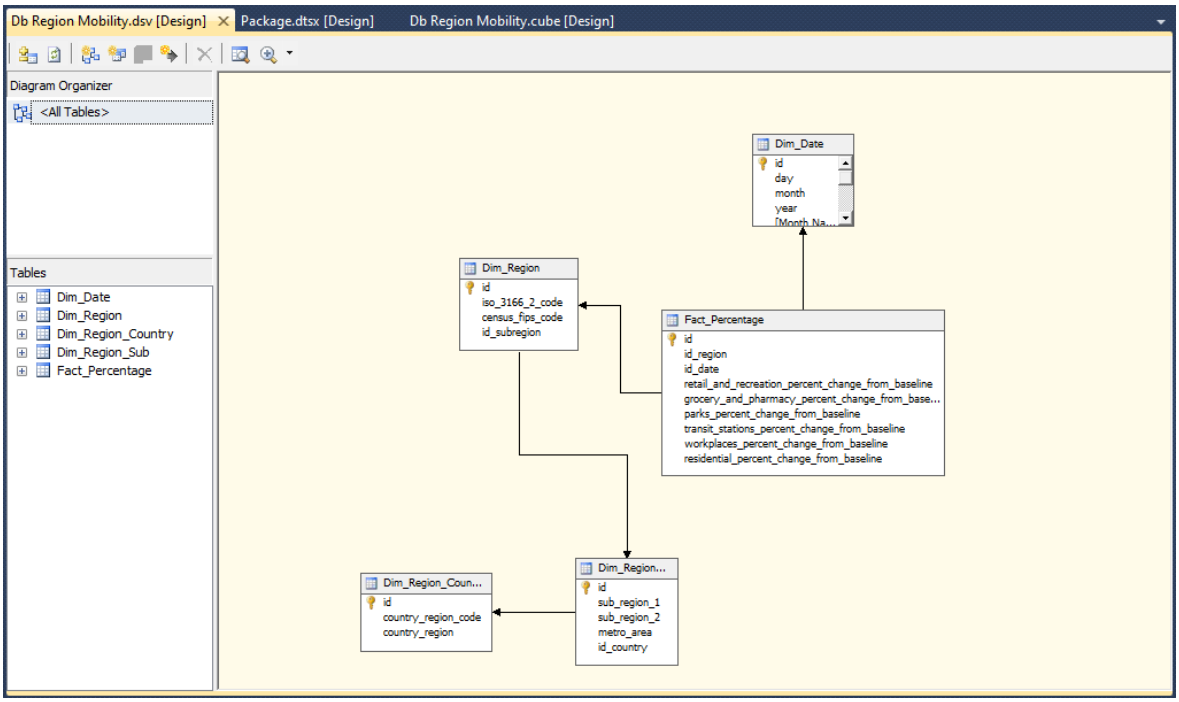
3. Creación de un un Data Source View

   Se procede a crear una vista para el data source, de esta manera indicamos las tablas que podremos utilizar para el análisis e indicar tanto la tabla de hechos como las tablas de dimensiones.





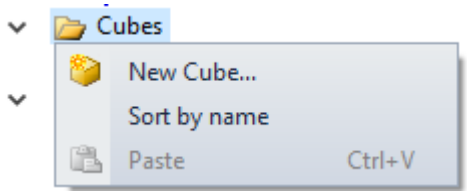
La vista del Data Source quedaría de la siguiente manera:



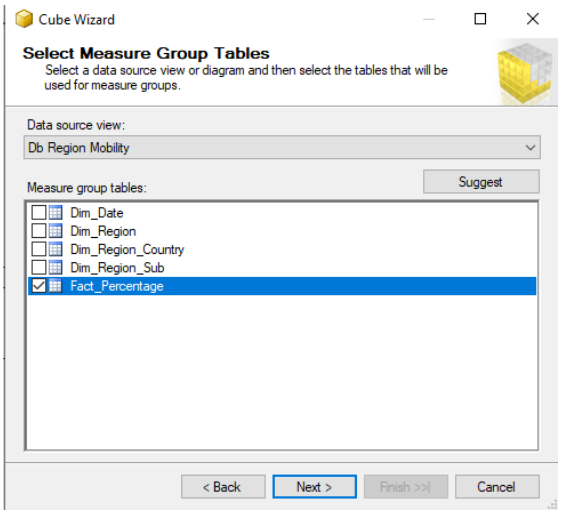
4. Creación del Cubo

Ahora procederemos a crear el cubo, usando tablas existentes (el data source)

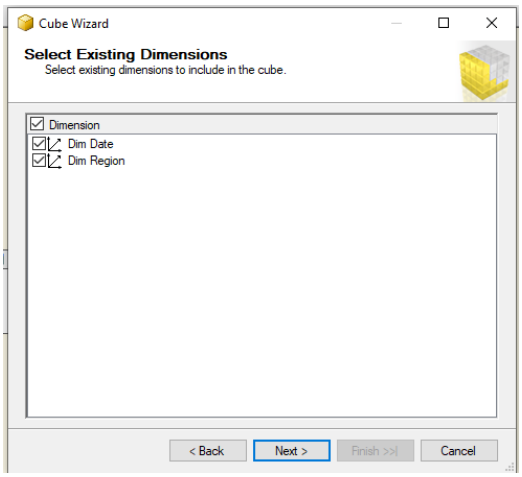
- Se crea el cubo



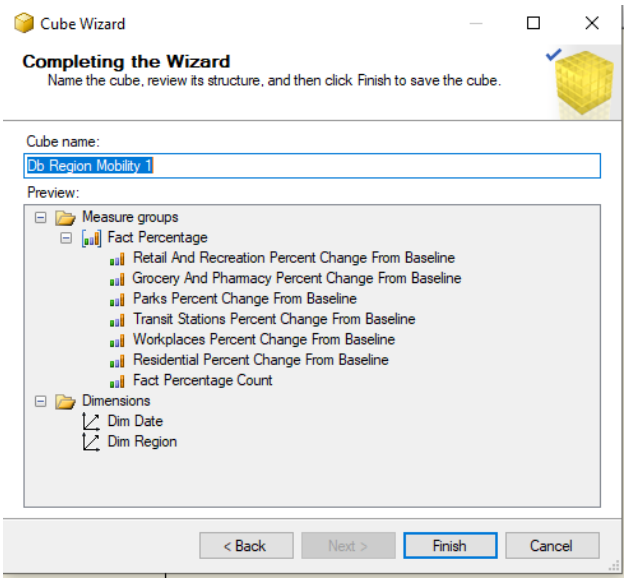
- Se indica la tabla de hechos



- Confirmamos las dimensiones

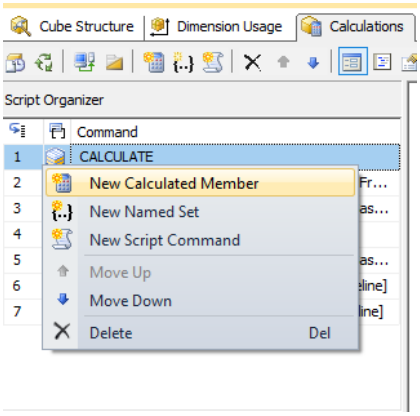


El cubo quedaría de la siguiente manera:



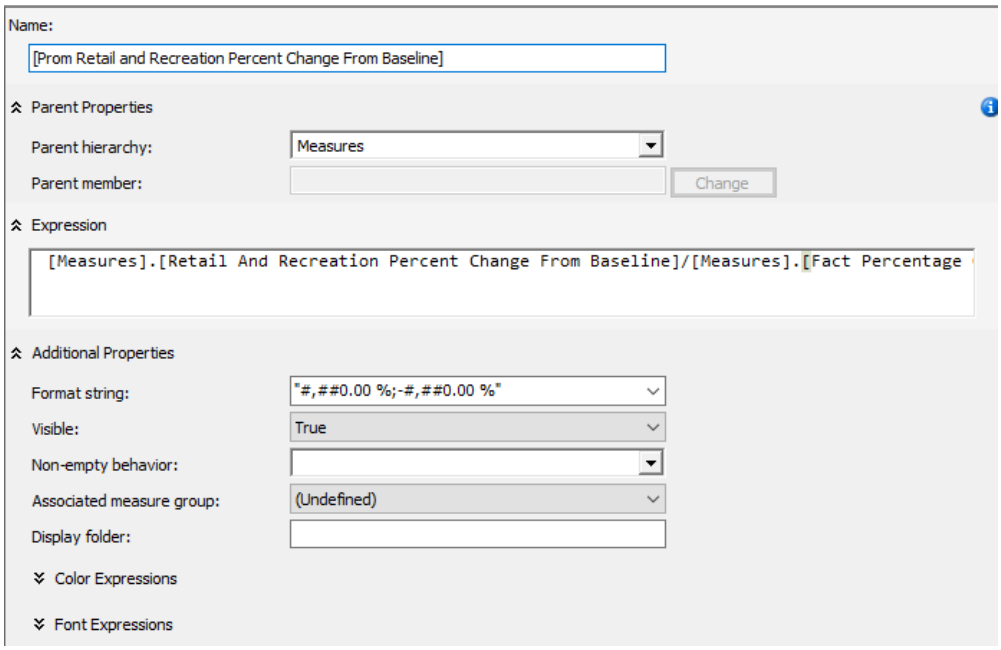
5. Agregar miembros calculados

En esta parte agregamos los promedios de los porcentajes de cada sitio

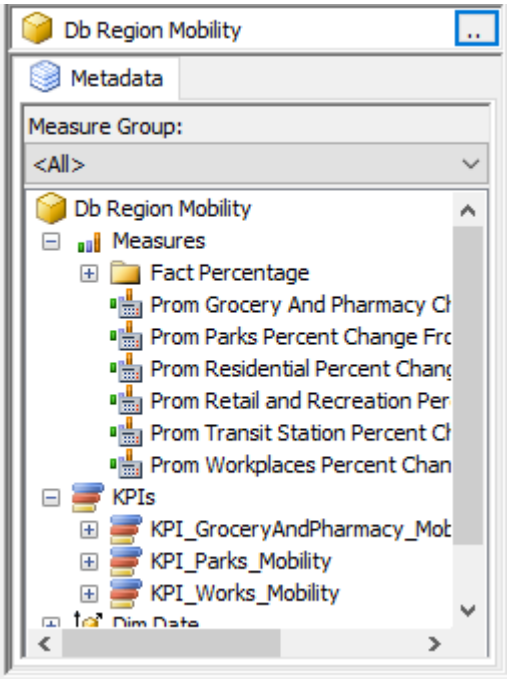


En esta parte agregamos un nuevo miembro calculado y definimos el cálculo a realizar, este será una nueva medida para el cubo.

En la siguiente imagen se realiza el cálculo del promedio de movilidad en comercio minorista y recreación.



Y se realiza de la misma manera para los demás promedios de otras localidades. Al finalizar se procesa el cubo y los promedios se pueden ver en el navegador:



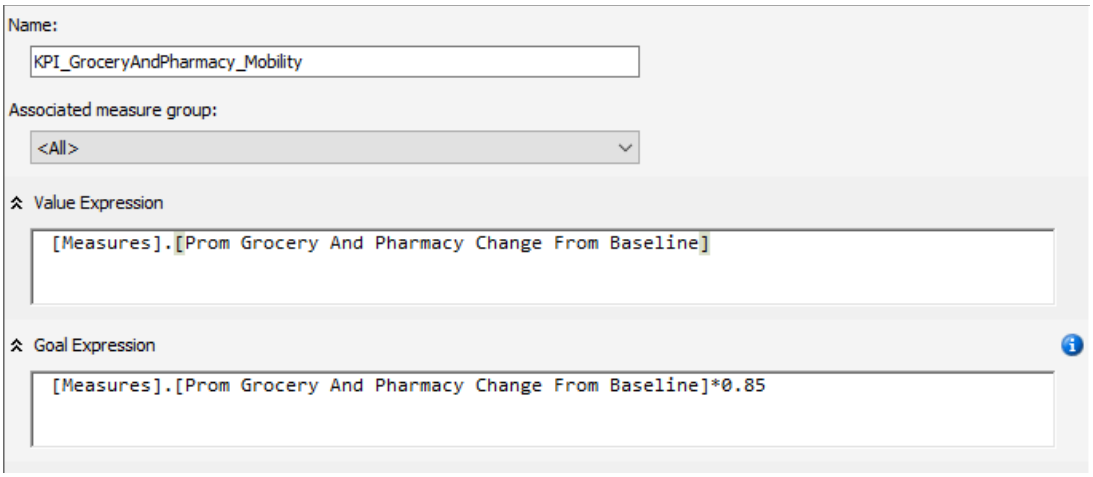
6. Crear KPI's

Estos indicadores se crearán para definir 2 cosas:

- Indicar en qué momento la movilidad incrementa o decrementa en las zonas indicadas
- Indicar la importancia de ciertas zonas respecto a otras

Para indicar cuándo la movilidad incrementa o decrementa, se realiza en base a la KPI de tendencia, restando al registro actual el anterior, si la diferencia es negativa entonces ha disminuido la movilidad si es positivo ha aumentado.

Para el ejercicio se puede hacer con cada zona, en este caso se muestra con las zonas de Farmacias:



Con lo anterior indicamos cuál será el valor de la expresión y el objetivo del valor

Luego en tendencia indicamos el siguiente código:

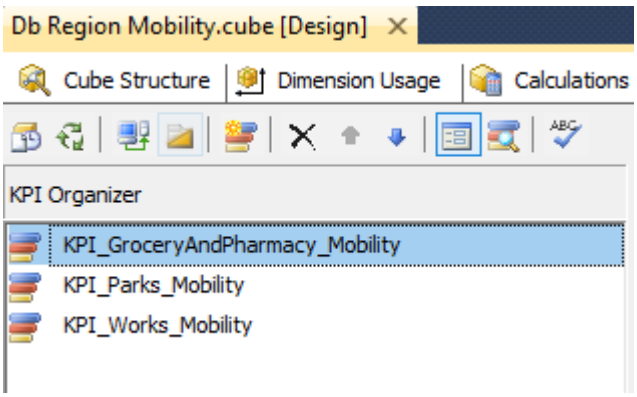
Trend

Trend indicator: Standard arrow

Trend expression:

```
CASE
    WHEN (KpiValue("KPI_GroceryAndPharmacy_Mobility")
    ,[Dim Date].[Month].CurrentMember)
    > (KpiValue("KPI_GroceryAndPharmacy_Mobility")
      ,[Dim Date].[Month].PrevMember) THEN 1
    WHEN (KpiValue("KPI_GroceryAndPharmacy_Mobility")
    ,[Dim Date].[Month].CurrentMember)
    < (KpiValue("KPI_GroceryAndPharmacy_Mobility")
      ,[Dim Date].[Month].PrevMember) THEN -1
    else 0
end
```

Para el siguiente indicador realizamos indicadores para diferentes localizaciones, por este motivo no se realizó la tercera dimensión, en este indicador haremos diferentes KPI's como se generó el de la farmacia, por ejemplo se realizará una comparación de Farmacia vs Parque y Farmacia vs Trabajo, por lo que hay que generar las KPI's para Parque y Trabajo, quedaría de la siguiente manera:



Ahora en la expresión de estado indicamos la comparación con los demás zonas de la siguiente manera:

Status

Status indicator: Shapes

Status expression:

```
CASE
    WHEN kpivalue("KPI_GroceryAndPharmacy_Mobility") > kpivalue("KPI_Parks_Mobility")
    THEN 1
    WHEN kpivalue("KPI_GroceryAndPharmacy_Mobility") < kpivalue("KPI_Parks_Mobility")
    THEN -1
    ELSE 0
END
```

En la imagen anterior se puede observar la evaluación de un KPI con respecto a otro y esto indicará cuál de los dos ha sido más visitado respecto a otro.

Resultados del cubo:

El primer análisis es indica el porcentaje de movilidad en cada mes en cada subregión (departamento) de Nicaragua

Month	Month Name	Sub Region 1	Prom Grocery And Pharma...
2	February		0.766666666666667
2	February	Boaco	0
2	February	Carazo	0
2	February	Chinandega	-1.2
2	February	Chontales	0.933333333333333
2	February	Esteli	-2.06666666666667
2	February	Granada	0.333333333333333
2	February	Jinotega	0
2	February	Leon	0.666666666666667
2	February	Madriz	0
2	February	Managua	1.2
2	February	Masaya	-3.13333333333333
2	February	Matagalpa	-0.133333333333333
2	February	North Carib...	0
2	February	Nueva Segovia	0
2	February	Rivas	0
2	February	South Carib...	0

El segundo análisis utiliza un KPI para indicar cuando aumentó o decrementó la movilidad en una específica región durante cada mes:

Month	Sub Region 1	Prom Grocery And Pharma...	KPI_GroceryAndPharmacy...
2	Chinandega	-1.2	-1
3	Chinandega	-3.64516129032258	-1
4	Chinandega	-20.0666666666667	-1
5	Chinandega	-25.0645161290323	-1
6	Chinandega	-20.5333333333333	1
7	Chinandega	-15.1290322580645	1
8	Chinandega	-5.70967741935484	1
9	Chinandega	0	1

Se puede ver como el KPI nos muestra la tendencia de cuando se ha sufrido un decremento (-1) o incremento (1) de la movilidad, esto evidentemente queda mucho más visible y entendible con reportes o herramientas como Power BI.

Y por último Se utiliza un KPI para la evaluación entre zonas, determinando si una ha sido más importante o visitada por otra:

Dimension	Hierarchy	Operator	Filter Expression	Parame...
Dim Region	Iso 3166 2 Code	Equal	{ NI-GR }	☐
<Select dimension>				

Month	Sub Region 1	Prom Grocery And Pharma...	Prom Workplaces Percent ...	KPI_Works_Mobility Status
2	Granada	0.333333333333333	12.6666666666667	1
3	Granada	-4.09677419354839	2.54838709677419	1
4	Granada	-13	-21.1	-1
5	Granada	-14.3548387096774	-19	-1
6	Granada	-12.4333333333333	-23.8333333333333	-1
7	Granada	-9.35483870967742	-22.1935483870968	-1
8	Granada	-2.58064516129032	-20.5483870967742	-1
9	Granada	0	-23.1111111111111	-1
Unkn...	Granada	(null)	(null)	0

Si es (1) ha tenido mayor movilidad las zonas de trabajo
Si es (-1) las farmacias han tenido mayor afluencia de personas

De lo anterior se puede concluir que Nicaragua tuvo restricciones de movilidad en las empresas desde el mes de abril y se mantiene hasta la fecha.

Dimension	Hierarchy	Operator	Filter Expression	Parame...
Dim Region	Iso 3166 2 Code	Equal	{ NI-GR }	☐
<Select dimension>				

Month	Sub Region 1	Prom Grocery And Pharma...	Prom Parks Percent Chang...	KPI_GroceryAndPharmacy...
2	Granada	0.333333333333333	-8.46666666666667	1
3	Granada	-4.09677419354839	-22.6451612903226	1
4	Granada	-13	-32.6	1
5	Granada	-14.3548387096774	-33.1290322580645	1
6	Granada	-12.4333333333333	-31.6333333333333	1
7	Granada	-9.35483870967742	-15.3225806451613	1
8	Granada	-2.58064516129032	-10.9032258064516	1
9	Granada	0	0	0
Unkn...	Granada	(null)	(null)	0

Al ser todos un valor de (1) las farmacias siempre han sido más visitados por las personas que los parques

Ocurre igual con la imagen anterior donde se muestra la comparación entre farmacias y parques, donde se puede concluir que una gran mayoría de Nicaragüenses atendieron al llamado de quedarse en casa por lo que durante los meses más fuertes del confinamiento (Abril, Mayo y Junio) las farmacias fueron más visitadas que los parques y los lugares de trabajo.