



Inteligencia de Negocios (Laboratorio)

Índice

Presentación	5
Red de contenidos	7
Unidad de Aprendizaje 1	
APLICACIONES MICROSOFT DE BUSINESS INTELLIGENCE	8
1.1 Tema 1 : Introducción	9
1.1.1 : SQL Server como herramienta de diseño en soluciones BI	9
1.1.2 : Revisión de los servicios SSAS, SSIS, SSRS y SQL Server	
: Agent	11
1.1.3 : Revisión del DW AdventureWorks 2014	16
1.2 Tema 2 : Conociendo un cubo y sus elementos	18
1.2.1 : Creación de cubos a través de SQL Server 2014 Data Tools	18
1.2.2 : Definición de un cubo y medidas	20
1.2.3 : Uso de Data Source y Data Source View	25
1.2.4 : Implementación de cubos en Analysis Services	25
Unidad de Aprendizaje 2	
IMPLEMENTACIÓN DE SOLUCIONES OLAP CON MICROSOFT SQL SERVER 2014	37
2.1 Tema 3 : Definición de estructura de Dimensiones	38
2.1.1 : Creando Dimensiones	38
2.1.2 : Definiendo tipos de Dimensiones	40
2.1.3 : Definiendo jerarquías	49
2.1.4 : Creando dimensión de tiempo	65
2.1.5 : Agrupando miembros de un atributo	70
2.2 Tema 4 : Definiendo Medidas	72
2.2.1 : Creando cubos, medidas y grupos de medidas	72
2.2.2 : Relacionando Dimensiones y grupos de medidas	81
2.3 Tema 5 : Definición de Miembros calculados y KPI	99
2.3.1 : Creando Miembros Calculados	99
2.3.2 : Creando KPI	107
2.4 Tema 6 : Definiendo Traducciones y perspectivas	113
2.4.1 : Creando Traducciones	113
2.4.2 : Creando Perspectivas	122
Unidad de Aprendizaje 3	
POBLANDO DATA WAREHOUSE	129
3.1 Tema 7 : Herramienta ETL SSIS	130
3.1.1 : SQL Server Data Tools y el diseñador de paquetes	130
3.1.2 : Elementos de SSIS	131
3.1.2.1 : Conexiones y tipos	131
3.1.2.2 : Task	147
3.1.2.3 : Control Flow	150
3.1.2.4 : Container	159
3.1.2.5 : Package	175

3.2 Tema 8 : Proceso de carga	200
3.2.1 : Diseño de proceso de carga	200
3.2.1.1 : Mapeo de Datos	200
3.2.1.2 : Estrategia de manejo de errores	201
3.2.1.3 : Estandarización y limpieza de datos	205
3.2.1.4 : Relacionando Dimensiones y grupos de medida	205
3.2.1.5 : Creación y Desarrollo de Proyecto SSIS	206
Unidad de Aprendizaje 4	
GENERANDO REPORTES Y CONSULTAS MDX	208
4.1 Tema 9 : MDX	209
4.1.1 : Introducción a MDX	209
4.1.2 : Consulta de cubos a través de MDX	209
4.1.3 : Concepto de tuplas y sets	213
4.1.4 : Funciones en MDX	214
4.1.5 : Filtra y ordena resultados	227
4.2 Tema 10 : Creación de Reportes utilizando SSRS	229
4.2.1 : Introducción a SSRS	229
4.2.2 : Creación de proyectos SSRS	231
4.2.2.1 : Establecer fuente de datos	232
4.2.2.2 : Establecer conjunto de datos y creación de consultas	235
4.2.2.3 : Diseño de Reporte	239
4.2.2.4 : Uso de Tablix y Matrix	239
4.3 Tema 11 : Optimización y publicación de reportes	243
4.3.1 : Optimización de reportes	243
4.3.1.1 : Uso de Parámetros	243
4.3.1.2 : Uso de Gráficos y Medidores	247
4.3.1.3 : Creación de Sub Informe	253
4.3.1.4 : Navegación entre reportes	253
4.3.2 : Publicación de Reporte	254

Presentación

Una buena toma de decisiones debe estar soportada en un análisis exhaustivo de la realidad. La importancia de tener diversas perspectivas del negocio es imprescindible.

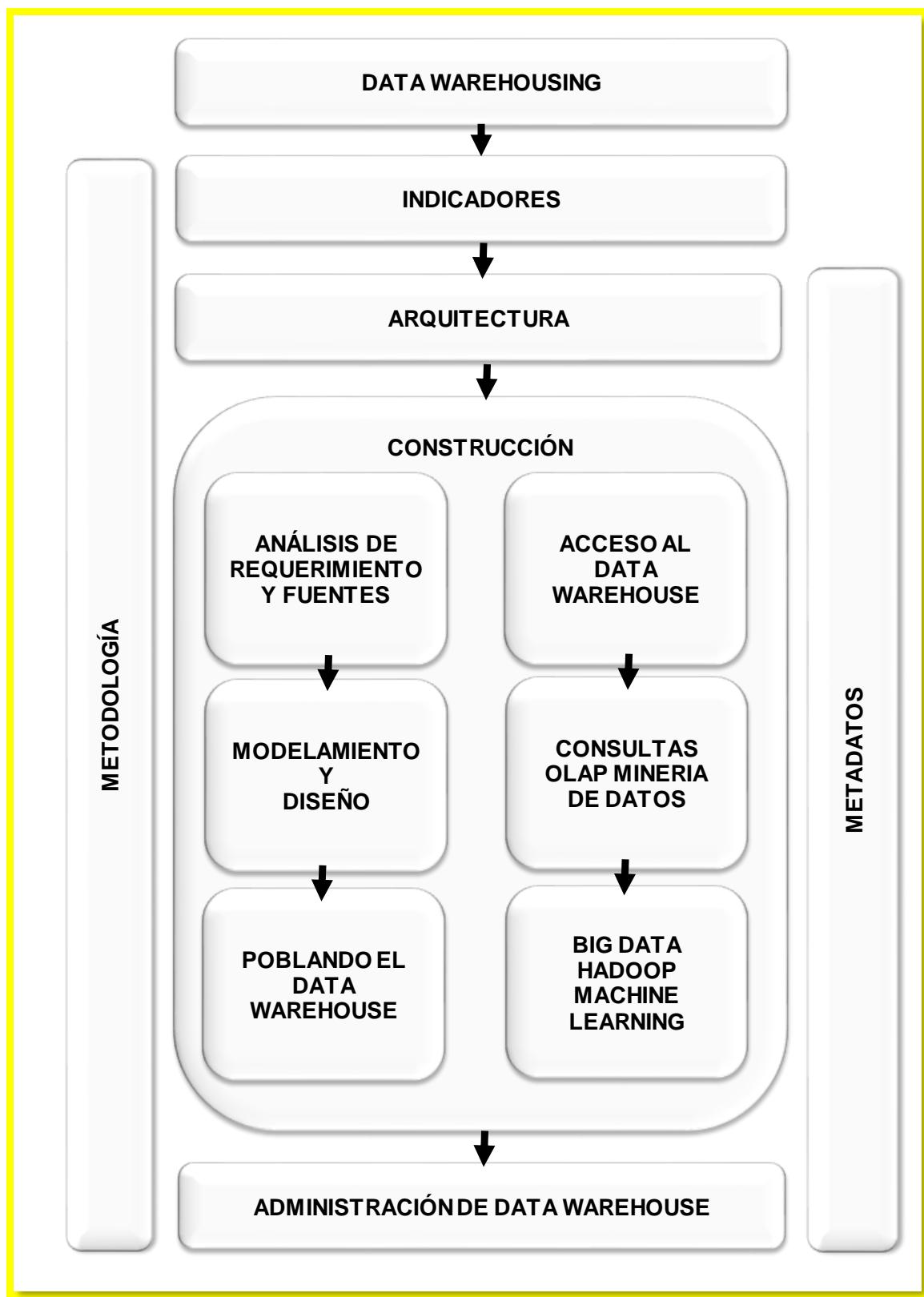
El presente manual, desarrolla la manera de implementar una solución analítica basada en un proceso de datawarehousing. Las herramientas empleadas permiten que el usuario (quien toma las decisiones), tenga diversas formas de ver su realidad y lo más importante es que le permite a él mismo crear sus propias perspectivas de su realidad. Para esto se desarrolla un ejemplo completo de implementación de una solución OLAP, desde la definición de los indicadores (signos vitales de una organización), pasando por el modelamiento analítico, la implementación de estructuras conocidas por cubos (vista analítica personalizada al usuario) hasta la explotación de datos en herramientas Office y su publicación en Internet.

Este material esta unidades de aprendizaje. Cada unidad tiene objetivos, contenidos y actividades que permitirán afianzar el logro de los objetivos.

Las unidades de aprendizaje han sido elaboradas, también, sobre la base de diversos textos sobre el tema y publicaciones en Internet que aparecen en la bibliografía del curso. Estos resúmenes, en algunos casos, contienen copias literales de fragmentos de estas fuentes, los cuales han sido complementados con comentarios del autor para un mejor entendimiento del tema.

Se recomienda la lectura del material de la unidad correspondiente antes de asistir a clase, con el fin de fomentar la discusión del tema y facilitar la comprensión del mismo.

Red de contenidos





APLICACIONES MICROSOFT DE INTELIGENCIA DE NEGOCIOS

LOGRO DE LA UNIDAD DE APRENDIZAJE

Al término de la unidad, el alumno conocerá de forma preliminar las herramientas de Business Intelligence de Microsoft.

TEMARIO

1.1 Tema 1 : Introducción

- 1.1.1 : SQL Server como herramienta de diseño de soluciones de BI
- 1.1.2 : Revisión de los servicios SSAS, SSIS, SSRS y SQL Server Agent
- 1.1.3 : Revisión del Data Warehouse AdventureWors 2014

1.2 Tema 2 : Conociendo un Cubo y sus elementos

- 1.2.1 : Creación de cubos a través de SQL Server 2014 Data Tools
- 1.2.2 : Definición de la estructura: Cubos y medidas
- 1.2.3 : Uso de Data Source y data Source View
- 1.2.4 : Implementación de cubos en Analysis Services

ACTIVIDADES PROPUESTAS

1. Los alumnos reconocen las herramientas Microsoft para BI.
2. Los alumnos reconocen el DW Adventure Works 2014
3. Los alumnos identifican Cubos OLAP.

INTRODUCCIÓN

1.1.1. SQL Server como herramienta de soluciones BI

Microsoft, empresa visionaria en Inteligencia de negocios proporciona las siguientes herramientas para soluciones de Inteligencia de Negocios:



SQL Server Integration Services.-

Herramienta ETL que permite la extracción, transformación y carga de datos.

SSIS es una plataforma para la construcción de soluciones de integración de datos de alto rendimiento, incluyendo la extracción, transformación y paquetes de carga (ETL) para el almacenamiento de datos. SSIS incluye herramientas y asistentes para la construcción y depuración gráficas de paquetes; tareas para llevar a cabo las funciones de flujo de trabajo, tales como operaciones de FTP, ejecutar sentencias SQL, y el envío de mensajes de correo electrónico; la extracción y carga de datos; transformaciones, limpieza, agregación, fusión y copia de datos.



SQL Server Analysis Services.-

Herramienta OLAP que permite la creación de Cubos de Información y Datamining.

SSAS es la herramienta de Microsoft Business Intelligence, para desarrollar soluciones de procesamiento analítico en línea (OLAP). En términos simples, se puede utilizar para crear cubos SSAS a partir de datos de data marts / almacén de datos para el análisis de datos más profundos y más rápidos. Los cubos son las fuentes de datos multidimensionales que tienen dimensiones y hechos (también conocidas como medidas) como componentes básicos. Desde una perspectiva relacional dimensiones pueden ser considerados como tablas maestras y los hechos pueden considerarse detalles medibles.



SQL Server Reporting Services.-

Herramienta Enterprise Reporting que satisface los 3 niveles del ciclo de vida de reportes empresariales: Creación, Administración y Envío.

SQL Server Reporting Services ofrece una gama completa de herramientas y servicios para crear, implementar y administrar informes para el negocio. Reporting Services incluye características de programación que permite ampliar y personalizar la funcionalidad de informes.

Reporting Services es una plataforma de informes basada en servidor que proporciona una completa funcionalidad de informes para una variedad de fuentes de datos. Reporting Services incluye un conjunto completo de herramientas para crear, administrar y entregar informes y así también APIs que permiten a los desarrolladores procesar informes con aplicaciones personalizadas.



Master Data

Services.-

Servicio de maestro de datos que consolida las tablas principales en un solo servidor.

Master Data Services (MDS) es la solución de SQL Server para la gestión de datos maestros. Gestión de datos maestros (MDM) describe los esfuerzos realizados por una organización para descubrir y definir listas no transaccionales de datos, con el objetivo de compilar listas maestras mantenibles. Un proyecto de MDM generalmente incluye una evaluación y reestructuración de los procesos de negocio internos junto con la implementación de la tecnología MDM. El resultado de una exitosa solución MDM es de datos fiables y centralizados que puede ser analizada, lo que resulta en mejores decisiones para el negocio.

Data Quality Services.-

Servicio de calidad de datos.



Power BI.-

Conjunto de herramientas Self-Services BI que permite tomar decisiones y está compuesto por Power Pivot, Power View, Power Map y Power Query.

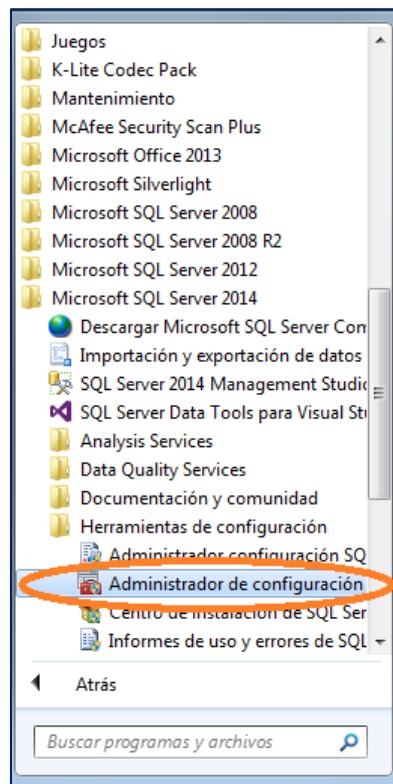
Microsoft Power BI es una colección de funciones y servicios en línea que le permite encontrar y visualizar los datos, compartir descubrimientos, y colaborar en nuevas formas intuitivas.



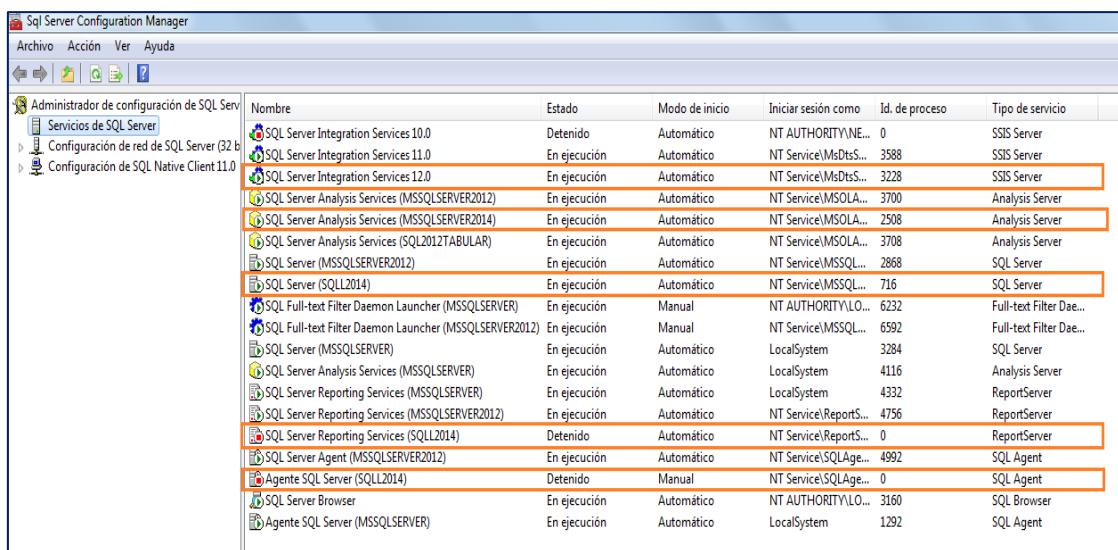
1.1.2. Revisión de los servicios SSAS, SSIS, SSRS, SQL Server Agent

Para trabajar con las herramientas Business Intelligence de Microsoft se debe tener activado los servicios:

Hacer click en Inicio, todos los programas, Microsoft SQL Server 2014, Herramientas de Configuración, Administración de configuración SQL Server 2014.



Aparecerá una ventana donde se verificará que los servicios estén iniciados o ejecutándose.



Servicio	Cuándo se inicia
Analysis Services (Nombre de la instancia predeterminada: MSSQLServerOLAPService)	Una instancia de Analysis Services debe iniciarse para que los usuarios puedan iniciar sesión en esa instancia.
Nombre de la instancia: MSOLAP\$<i>InstanceName</i>)	
SQL Server (Nombre de la instancia predeterminada: MSSQLServer)	Si el almacenamiento relacional de Analysis Services se almacena en SQL Server, o si la reescritura se ha configurado para SQL Server, el servicio de SQL Server que aloja los datos relacionales debe iniciarse antes de que Analysis Services pueda obtener acceso a los datos del almacenamiento relacional o volver a escribir datos al almacén relacional.
Nombre de la instancia: MSSQL\$<i>InstanceName</i>)	
SQL Server Agente (Nombre de la instancia predeterminada: SQLServerAgent)	Si el Agente de SQL Server está configurado para ejecutar tareas automatizadas, el Agente de SQL Server de la instancia adecuada de SQL Server debe iniciarse.
Nombre de la instancia: SQLAgent\$<i>InstanceName</i>)	
SQL Server Explorador (Nombre del servicio: SQL Server Explorador)	Si los usuarios deben conectarse a una instancia con nombre de Analysis Services sin especificar un número de puerto, debe iniciarse el Explorador de SQL Server.

Una instalación de SQL Server se compone de una o más instancias independientes. Una instancia, ya sea predeterminada o con nombre, tiene su propio conjunto de archivos de programa y de datos, así como un conjunto de archivos comunes compartidos entre todas las instancias de SQL Server del equipo.

En una instancia de SQL Server que incluya el Motor de base de datos, Analysis Services y Reporting Services, cada componente tiene un conjunto completo de datos y de archivos ejecutables, así como de archivos comunes compartidos por todos los componentes.

Para aislar las ubicaciones de instalación de cada componente, se generan identificadores de instancia únicos para cada componente de una determinada instancia de SQL Server.

Archivos compartidos para todas las instancias de SQL Server

Los archivos comunes que usan todas las instancias en un único equipo se instalan en la carpeta <unidad>:\Archivos de programa\Microsoft SQL Server\120\, donde <unidad> es la letra de unidad donde están instalados los componentes. La unidad predeterminada es normalmente la C.

Ubicaciones de archivos y asignaciones del Registro

Durante la instalación de SQL Server, se genera un identificador de instancia en cada componente de servidor. Los componentes de servidor de esta versión de SQL Server son el Motor de base de datos, Analysis Services y Reporting Services.

El identificador de instancia predeterminado se construye con el formato siguiente:

- MSSQL para el Motor de base de datos, seguido del número de versión principal, un guion bajo y la versión secundaria cuando proceda, un punto y, a continuación, el nombre de instancia.
- MSAS para Analysis Services, seguido del número de versión principal, un guion bajo y la versión secundaria cuando proceda, un punto y, a continuación, el nombre de instancia.
- MSRS para Reporting Services, seguido del número de versión principal, un guion bajo y la versión secundaria cuando proceda, un punto y, a continuación, el nombre de instancia.

A continuación se enumeran algunos ejemplos de identificadores de instancia predeterminados de esta versión de SQL Server:

- MSSQL12.MSSQLSERVER para una instancia predeterminada de SQL Server 2014.
- MSAS12.MSSQLSERVER para una instancia predeterminada de SQL Server 2014 Analysis Services (SSAS).
- MSSQL12.MyInstance para una instancia con nombre de SQL Server 2014 denominada "MyInstance".

La estructura de directorios para una instancia con nombre de SQL Server 2014 que incluyera el Motor de base de datos y Analysis Services, se denominara "MyInstance" y estuviera instalada en los directorios predeterminados sería como sigue:

- C:\Archivos de programa\Microsoft SQL Server\MSSQL12.MyInstance\
- C:\Archivos de programa\Microsoft SQL Server\MSAS12.MyInstance\

Puede especificar cualquier valor para el identificador de instancia, pero evite los caracteres especiales y las palabras clave reservadas.

Puede especificar un identificador de instancia no predeterminado durante la instalación de SQL Server. En lugar de <Archivos de programa>\Microsoft SQL Server, se usa una <ruta de acceso personalizada>\Microsoft SQL Server, si el usuario decide cambiar el directorio de instalación predeterminado. Tenga en cuenta que no se admiten identificadores de instancia que comienzan por un subrayado (_) o que contienen el signo de almohadilla (#) o el signo de dólar (\$).

Integration Services y los componentes de cliente no reconocen las instancias y, por consiguiente, no tienen asignado un identificador de instancia. De forma predeterminada, los componentes que no reconocen las instancias se instalan en un único directorio: <unidad>:\Archivos de programa\Microsoft SQL Server\120\. Si cambia la ruta de instalación de un componente compartido, cambiará también la de los demás componentes compartidos. Las instalaciones posteriores instalan componentes que no reconocen instancias en el mismo directorio que la instalación original.

SQL ServerAnalysis Services es el único componente de SQL Server que permite cambiar el nombre de las instancias después de la instalación. Si se cambia el nombre una instancia de Analysis Services, el identificador de la instancia no cambiará. Después de completarse el cambio de nombre de la instancia, los directorios y claves del Registro continuarán utilizando el identificador de instancia creado durante la instalación.

El subárbol del Registro se crea debajo de HKLM\Software\Microsoft\Microsoft SQL Server\<Instance_ID> para los componentes dependientes de la instancia. Por ejemplo,

- HKLM\Software\Microsoft\Microsoft SQL Server\MSSQL12.MyInstance
- HKLM\Software\Microsoft\Microsoft SQL Server\MSAS12.MyInstance
- HKLM\Software\Microsoft\Microsoft SQL Server\MSRS12.MyInstance

El Registro también mantiene una asignación de identificador de instancia a nombre de instancia. La asignación de identificador de instancia a nombre de instancia se mantiene de la siguiente forma:

- [HKEY_LOCAL_MACHINE\Software\Microsoft\Microsoft SQL Server\Instance Names\SQL] "InstanceName"="MSSQL12"
- [HKEY_LOCAL_MACHINE\Software\Microsoft\Microsoft SQL Server\Instance Names\OLAP] "InstanceName"="MSAS12"
- [HKEY_LOCAL_MACHINE\Software\Microsoft\Microsoft SQL Server\Instance Names\RS] "InstanceName"="MSRS12"

Especificar rutas de acceso a los archivos

Durante la instalación, puede cambiar la ruta de instalación de las siguientes características:

La ruta de instalación solo aparece para las características cuya carpeta de destino puede configurar el usuario:

Componente	Ruta de acceso predeterminada	Ruta de acceso configurable o fija
Componentes de servidor de Motor de base de datos	\Archivos de programa\Microsoft SQL Server\MSSQL12. <instanceid>\< td=""><td>Configurable</td></instanceid>\<>	Configurable
Archivos de datos de Motor de base de datos	\Archivos de programa\Microsoft SQL Server\MSSQL12. <instanceid>\< td=""><td>Configurable</td></instanceid>\<>	Configurable
Servidor Analysis Services	\Archivos de programa\Microsoft SQL Server\MSSQL12. <instanceid>\< td=""><td>Configurable</td></instanceid>\<>	Configurable
Archivos de datos de Analysis Services	\Archivos de programa\Microsoft SQL Server\MSSQL12. <instanceid>\< td=""><td>Configurable</td></instanceid>\<>	Configurable
Servidor de informes de Reporting Services	\Archivos de programa\Microsoft SQL Server\MSRS12.<InstanceID>\Reporting Services\ReportServer\Bin\	Configurable
Administrador de informes de Reporting Services	\Archivos de programa\Microsoft SQL Server\MSRS12.<InstanceID>\Reporting Services\ReportManager\	Ruta de acceso fija
Integration Services	<Directorio de instalación>\120\DTs\	Configurable
Componentes cliente (excepto bcp.exe y sqlcmd.exe)	<Directorio de instalación>\120\Tools\	Configurable
Componentes cliente (bcp.exe y sqlcmd.exe)	<Directorio de instalación>\Client SDK\ODBC\110\Tools\Binn	Ruta de acceso fija
Objetos COM del servidor y la replicación	<unidad>:\Archivos de programa\Microsoft SQL Server\120\COM ⁵	Ruta de acceso fija
DLL de componentes de Integration Services del motor en tiempo de ejecución de transformación de datos, el motor de canalización de transformación de datos y la utilidad de símbolo del sistema dtsexec	<unidad>:\Archivos de programa\Microsoft SQL Server\120\DTs\Binn	Ruta de acceso fija
DLL que proporcionan compatibilidad con la conexión administrada para	<unidad>:\Archivos de programa\Microsoft SQL Server\120\DTs\Connections	Ruta de acceso fija

Integration Services		
DLL para cada tipo de enumeración que admite Integration Services	<unidad>:\Archivos de programa\Microsoft SQL Server\120\DTS\ForEachEnumerators	Ruta de acceso fija
Servicio Explorador de SQL Server, proveedores WMI	<unidad>:\Archivos de programa\Microsoft SQL Server\120\Shared\	Ruta de acceso fija
Componentes que se comparten entre todas las instancias de SQL Server	<unidad>:\Archivos de programa\Microsoft SQL Server\120\Shared\	Ruta de acceso fija

Cuando especifique una ruta de instalación durante la instalación de los componentes de servidor o de los archivos de datos, el programa de instalación utilizará el identificador de instancia además de la ubicación especificada para el programa y los archivos de datos. El programa de instalación no utiliza el identificador de instancia para las herramientas y otros archivos compartidos. Tampoco utiliza ningún identificador de instancia para el programa y los archivos de datos de Analysis Services, aunque lo use para el depósito de Analysis Services.

Si establece una ruta de instalación para la característica Motor de base de datos, el programa de instalación de SQL Server utilizará dicha ruta como directorio raíz de todas las carpetas específicas de la instancia en dicha instalación, incluido SQL Data Files. En este caso, si establece el directorio raíz en "C:\Archivos de programa\Microsoft SQL Server\MSSQL12.<nombreDeInstancia>\MSSQL\", los directorios específicos de la instancia se agregarán al final de dicha ruta de acceso.

Los clientes que decidan usar la funcionalidad de actualización de USESYSDB en el Asistente para la instalación de SQL Server (modo de UI del programa de instalación) pueden llegar con facilidad a una situación en la que el producto se instale en una estructura de carpetas recursiva. Por ejemplo, <SQLProgramFiles>\MSSQL12\MSSQL\MSSQL10_50\MSSQL\Data\). En su lugar, para usar la característica USESYSDB, establezca una ruta de instalación para la característica de archivos de datos de SQL (SQL Data Files) y no para la característica Motor de base de datos.

1.1.3. Revisión del Data Warehouse Adventure Works



Adventure Works 2014 Bases de datos de ejemplo

Clasificación: puntuaciones	Basado en 13	Lanzamiento: 03 de julio 2015
		Actualizado: 03 de julio

Comentado: 10 opiniones	2015 por DerrickVMSFT
Descargas: 3745	Estado Dev: Estable ?

AdventureWorksDW2014 contiene un subconjunto de las tablas de la base de datos OLTP, además de la información financiera que se extrae de una fuente de datos separada. Los datos se mantienen en sintonía con la base de datos OLTP para apoyar Integration Services típicos escenarios tales como la carga y actualización del DW. El AWDataWarehouseRefresh muestra cómo utilizar un paquete de Integration Services para cargar datos en AdventureWorksDW2014 de la base de datos OLTP AdventureWorks2014.

AdventureWorksDW2014 contiene dos áreas temáticas, finanzas y ventas, que se describen en las siguientes secciones. Además, del DW contiene vistas que se utilizan para apoyar a los escenarios de minería de datos que se describen más adelante.

Finanzas

El DW de finanzas se divide en dos esquemas que tienen las siguientes características:

Finanzas

- Contiene datos sobre las finanzas de Adventure Works corporativo y sus filiales.
- Contiene datos de la moneda local de la organización a la que está asociado.
- Soporta el grupo de medida de Finanzas.

Los Tipos

- Contiene los datos de conversión de moneda, como los porcentajes promedios diarios y totales, así también las tasas de diarias en relación con el dólar estadounidense (USD). .
- Soporta el grupo de medida de tipos de cambio.

Ventas

El área temática de ventas se divide en cuatro esquemas con las siguientes características:

Reseller Sales (Ventas a Distribuidor)

- Contiene ventas sólo a distribuidores.
- Contiene órdenes solamente enviados.
- Contiene datos en USD y moneda de origen.
- Soporta el grupo de medida Reseller Sales.

Resumen de ventas

- Contiene una vista de resumen de los datos de venta a distribuidor y de ventas por Internet.
- Ha reducido dimensionalidad en comparación con los esquemas de venta a distribuidor y de ventas por Internet.

Ventas por Internet

- Contiene cliente, las ventas individuales por internet y datos al detalle.
- Contiene órdenes solamente enviados.
- Contiene datos en USD y moneda de origen.

Cuota de Ventas

- Contiene datos de cuota de ventas de los representantes de ventas.
- Soporta el grupo de medida Sales Quota.

Las tablas de ventas por Internet han tenido una tendencia a apoyar escenarios de minería de datos, mientras que las otras mesas se han tendido a apoyar escenarios OLAP.

Escenarios de Minería de Datos

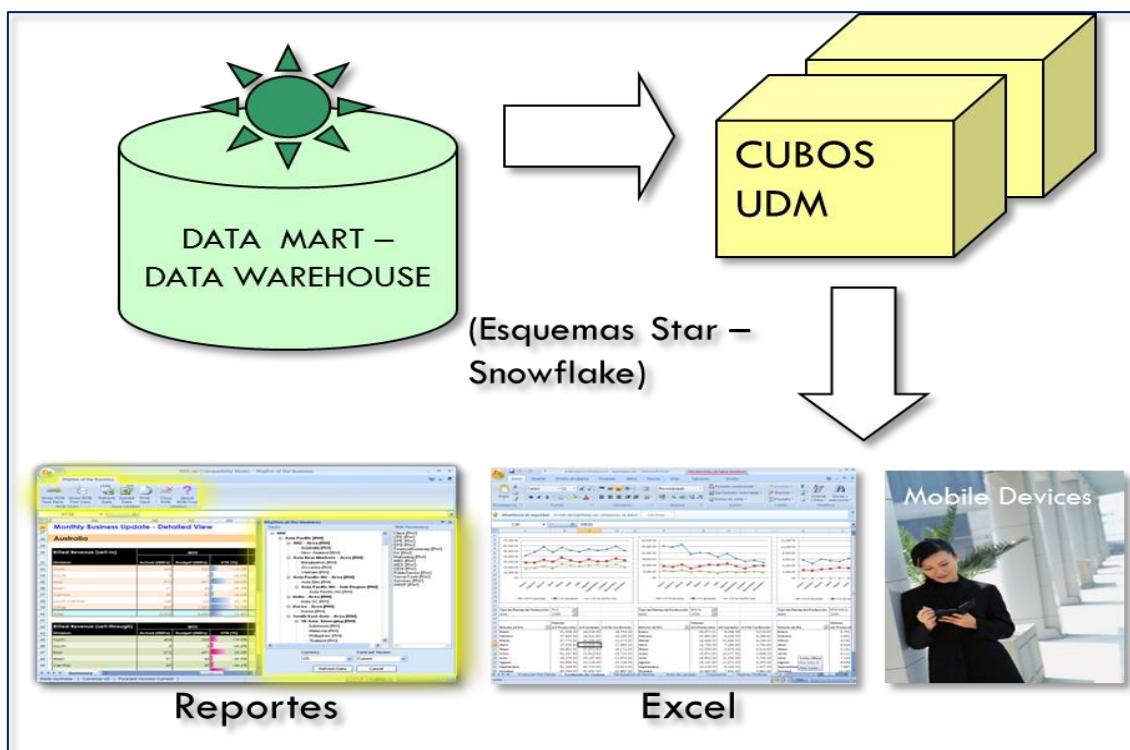
La tendencia en la base de datos AdventureWorksDW2014 soporta los siguientes escenarios de minería de datos:

- Predicción: Apoya el escenario de un analista que investiga el crecimiento de modelos de bicicletas por el tiempo y la región.
- Targeted Mailing Campaña: Apoya el escenario de un analista de la aplicación de diversos algoritmos de minería de datos para Adventure Works datos de ventas de los clientes y de Internet para determinar los atributos demográficos de los clientes que están dispuestos a comprar una bicicleta. El analista puede entonces aplicar el modelo de minería de datos a una lista de clientes potenciales con el fin de determinar qué clientes tienen más probabilidades de responder a un correo directo que promueve Adventure Works bicicletas.
- Market Basket Analysis: Apoya el escenario de un desarrollador crea una solución cesta de mercado que sugiere un producto basado en otros productos que ya existen en un carrito de la compra a los clientes.
- Clústeres de secuencia: Apoya el escenario de un analista de la investigación de la secuencia en la que los clientes ponen artículos en un carrito de compras.

Estos escenarios y la tendencia en el almacén de datos se demuestran en el tutorial de minería de datos en SQL Server Libros en pantalla y en el proyecto de ejemplo Adventure Works DW.

1.2. CONOCIENDO UN CUBO Y SUS ELEMENTOS

1.2.1. Creación de cubos a través de SQL Server 2014 Data Tools



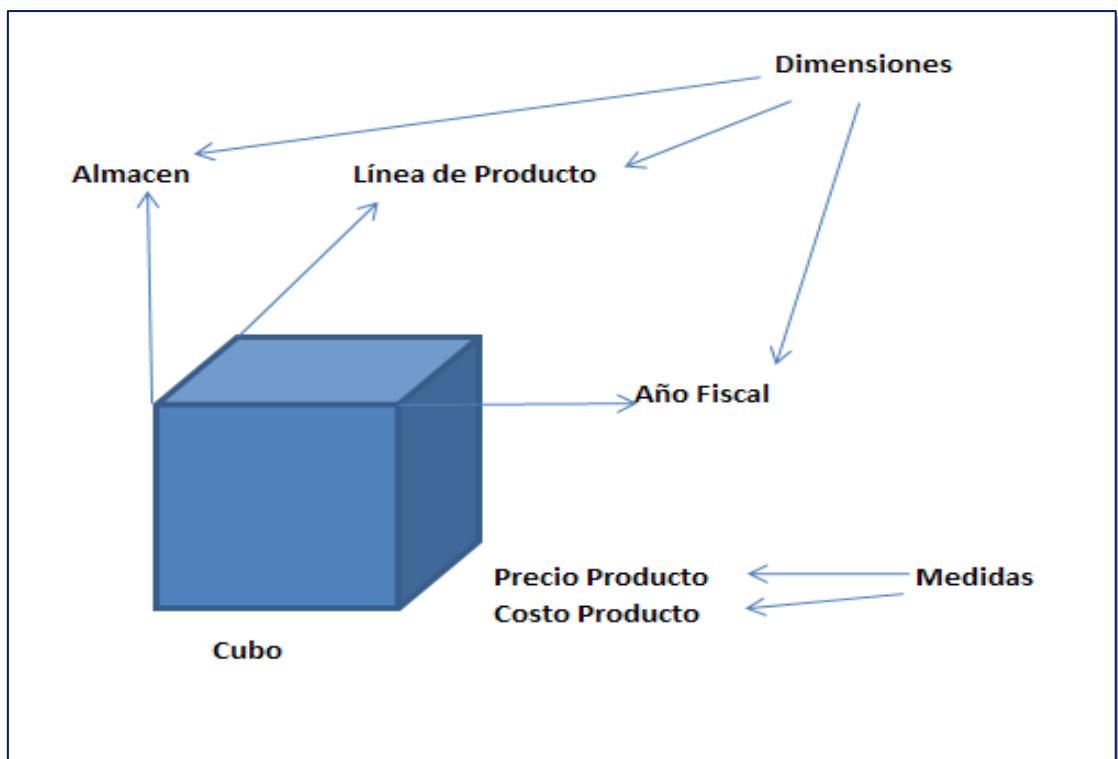
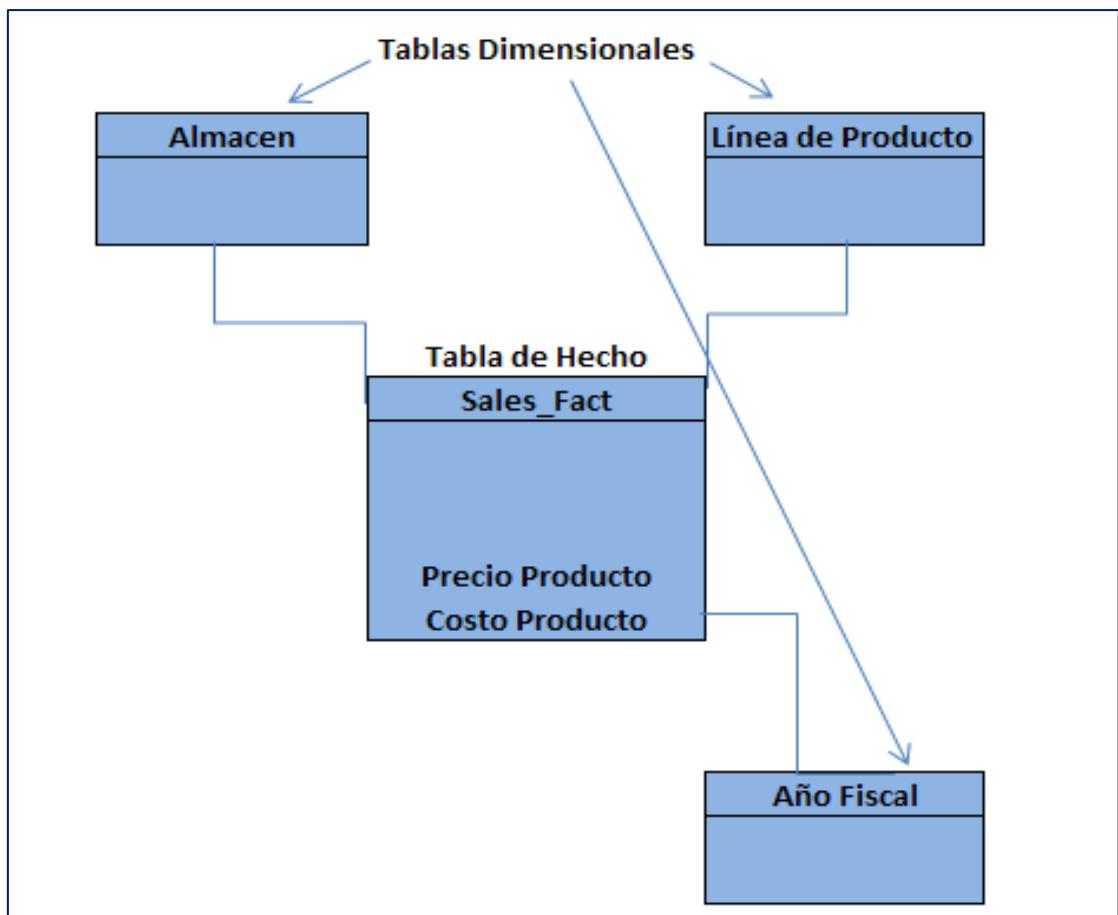
Una vez que la información ha sido consolidada y transferida al Data Mart (donde reside el esquema STAR o SNOWFLAKE), puede ser enviada a repositorios multidimensionales. SQL Server 2014 implementa el almacenamiento multidimensional a través de los Servicios de Análisis (SQL Server Analysis Services, o SSAS).

Los repositorios multidimensionales se denominan cubos. Estos permiten efectuar consultas muy complejas sobre grandes conjuntos de datos, con alto rendimiento y velocidad. Los datos se encuentran totalizados y precalculados, lo cual permite obtener un tiempo de respuesta reducido para consultas que implican cálculos sobre grandes volúmenes de información.

Un cubo se define a partir de una tabla de hechos y una o varias tablas de dimensión. Las medidas (measures) de la tabla de hechos se convertirán en las medidas del cubo. Las dimensiones del cubo se definen a partir de las tablas de dimensión. Por ejemplo, un cubo de ventas puede tener las medidas Precio_Producto y Costo_Producto, provenientes de la tabla de hechos Sales_Fact. Estas medidas se cruzan con las dimensiones Almacén,

Línea_de_Producto y Año_Fiscal. Las consultas efectuadas sobre este cubo permitirían obtener la evolución del precio de un producto a lo largo del tiempo, por almacén y línea de producto.

Veamos la ilustración:

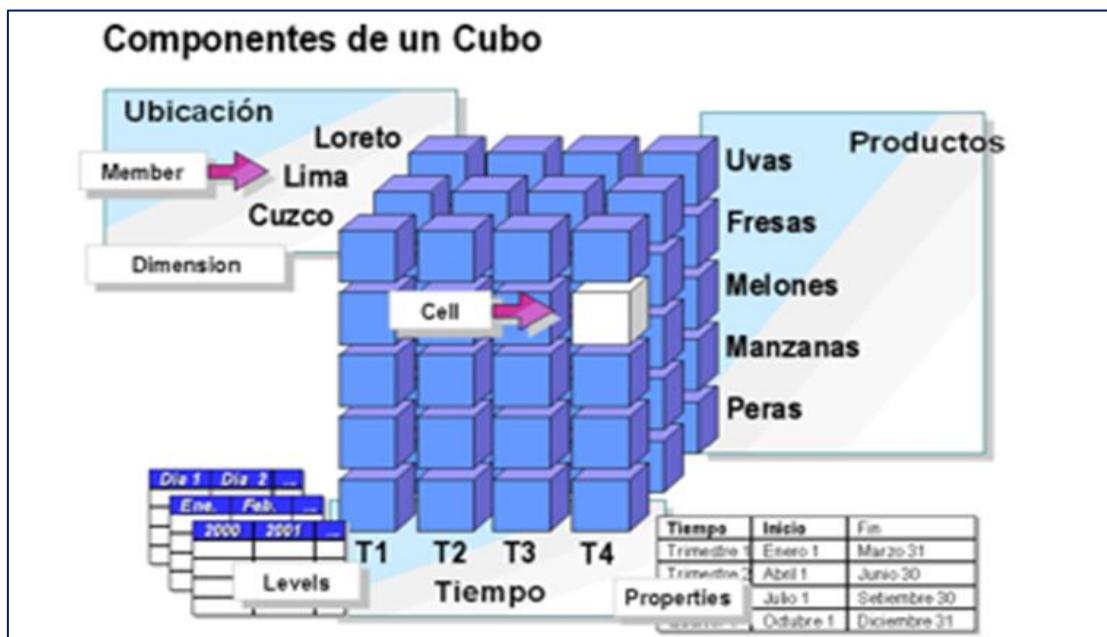


Cada dimensión en un cubo tiene uno o varios niveles (levels). Los niveles representan agrupaciones jerárquicas al interior de una dimensión; por ejemplo, la

dimensión Ubicación_Geográfica puede tener los niveles Continente, Región, País y Ciudad. Los continentes contienen regiones, las regiones contienen países, y los países ciudades.

La existencia de niveles permite obtener rápidamente resultados totalizados de información; de esta manera, los cuatro niveles mencionados en el ejemplo anterior permiten obtener las ventas totalizadas por Continente, Región, País y Ciudad.

1.2.2. Definición de la estructura de Cubos y Medidas



Los componentes básicos de un cubo son las dimensiones (dimensions), los miembros (members), (estos dos componentes provienen de las tablas dimensionales) y las celdas (cells) (que provienen de la tabla de hechos).

Dimensiones (dimensions)

Cada dimensión en un cubo representa una perspectiva desde la cual los usuarios consultan y analizan los datos. Desde un punto de vista gráfico, se puede afirmar que las dimensiones corresponden con los ejes del cubo. Por ejemplo, se desea un reporte que nos muestre las ventas por Producto, Tiempo y Ubicación, ellas representan las dimensiones de un cubo. En la figura superior, podemos ver la representación gráfica de las dimensiones Productos (en el eje vertical), Tiempo (en el eje horizontal) y Ubicación (en el eje que atraviesa perpendicularmente esta página).

Niveles (levels)

Los niveles representan los grados de sumarización de una dimensión. Por ejemplo, la dimensión Tiempo puede contener los niveles Año, Mes y Día.

Miembros (members)

Cada ocurrencia particular en una dimensión está representada por un miembro. Por ejemplo, en la figura los miembros de la dimensión Productos son Uvas, Fresas, Melones, Manzanas y Peras. Los miembros de la dimensión Ubicación son Loreto, Lima y Cuzco.

Celdas (cells)

Una celda está definida por una intersección de las dimensiones del cubo. En la figura, la celda de color blanco representa la intersección de los miembros T4 (Tiempo), Fresas (Productos) y Cuzco (Ubicación).

SQL Server 2014 introduce muchas mejoras respecto de las características existentes en Analysis Services 2000. A continuación se enumeran algunas de estas mejoras:

- Mejoras en el entorno de desarrollo de soluciones de Business Intelligence: Intellisense (identificación automática de código escrito), menús sensibles al contexto, etc.
- Utilización de UDM: Los cubos de SQL Server 2012 son, en realidad, definiciones escritas a través del Unified Dimensional Modeling (UDM). El UDM permite especificar la estructura del cubo (dimensiones y medidas) en un archivo XML que contiene tanto la data del cubo como su metadata. (Estructura).
- Mejoras en la arquitectura interna de los cubos:
 - En SSAS 2014, un cubo puede estar formado por varias tablas de hechos. En la versión 2000, cada cubo contenía solamente una tabla de hechos. Por tanto, para consolidar la información de varios cubos, era necesario utilizar cubos virtuales (no soportados en la versión SQL 2012). Esto producía una gran cantidad de cubos en cada base de datos de análisis.
 - Uso de atributos para especificar la estructura de las dimensiones. Esto permite incrementar la eficiencia en el procesamiento del cubo, y proporcionar una mejor experiencia de navegación para el usuario final.
 - Soporte para relaciones complejas entre tablas del datamart. SSAS 2014 soporta relaciones transitivas entre dimensiones y relaciones de muchos a muchos entre las tablas de hechos.
- Mejoras en el lenguaje de consulta MDX. Por ejemplo, SSAS incluye sentencias de definición de sub – cubos (esto sustituye a las celdas calculadas – calculated cells – de Analysis Services 2000).
- Mejoras en Data Mining. Contra los dos modelos de data mining soportados en la versión 2000, SSAS 2014 presenta siete modelos de data mining. Esto permite a las empresas llegar a un grado muy alto de compresión de la información del negocio. Por otro lado, SSAS incluye el lenguaje de consulta DMX (Data Mining Expressions), para consultar la información de los modelos de Data Mining.
- Inclusión de características para Business Intelligence en tiempo real. La característica de Proactive Caching permite que los cubos actualicen automáticamente su caché de información ante cambios en los orígenes de datos, o revertir el modo de almacenamiento de MOLAP a ROLAP cuando la información del cubo queda desactualizada.



Con el entorno de desarrollo SQL Server Data Tools, se crean proyectos de Analysis Services, a través de los cuales se define la estructura de los cubos en una solución de Business Intelligence.

Algunas de las mejoras respecto de la herramienta existente en la versión 2000 de SQL Server son las siguientes:

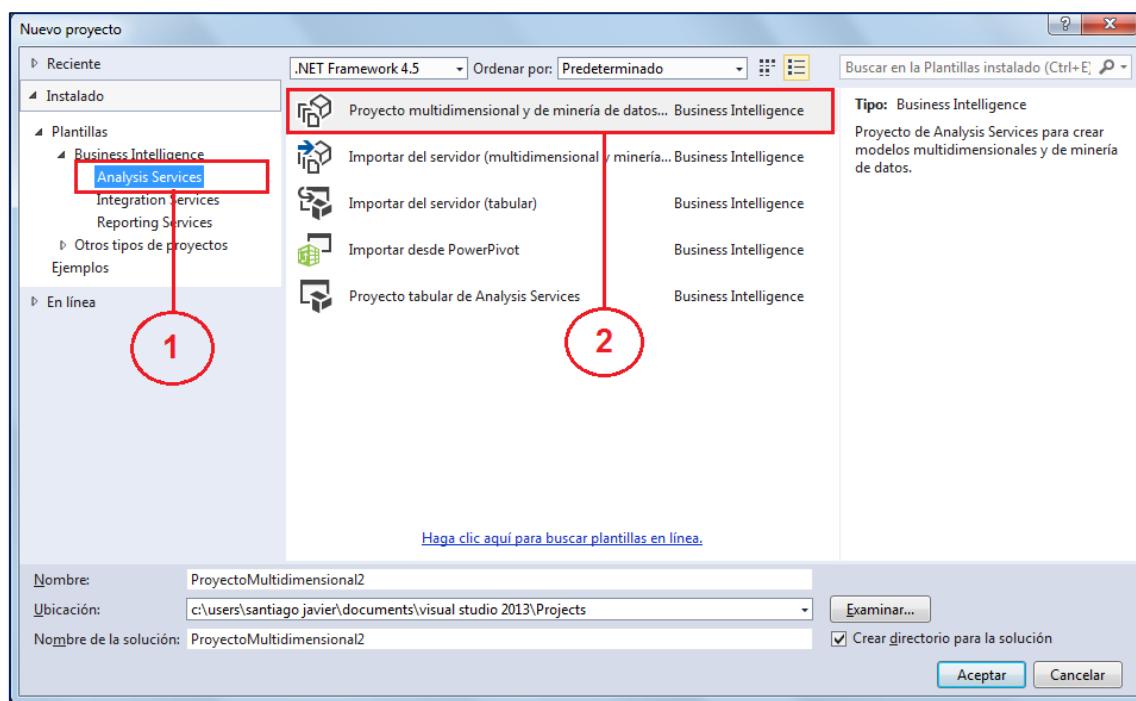
- Definición de Data Source Views

En la versión SQL Server 2000, la creación de bases de datos de análisis requiere una conexión permanente hacia el origen de datos donde residen los esquemas de datos (STAR o SNOWFLAKE). En SQL Server 2012, los Data Sources Views (vistas de origen de datos) almacenan localmente información sobre la estructura de las tablas del data mart. De esta manera, el desarrollador puede continuar trabajando sobre el cubo sin necesidad de una conexión con el origen de datos.

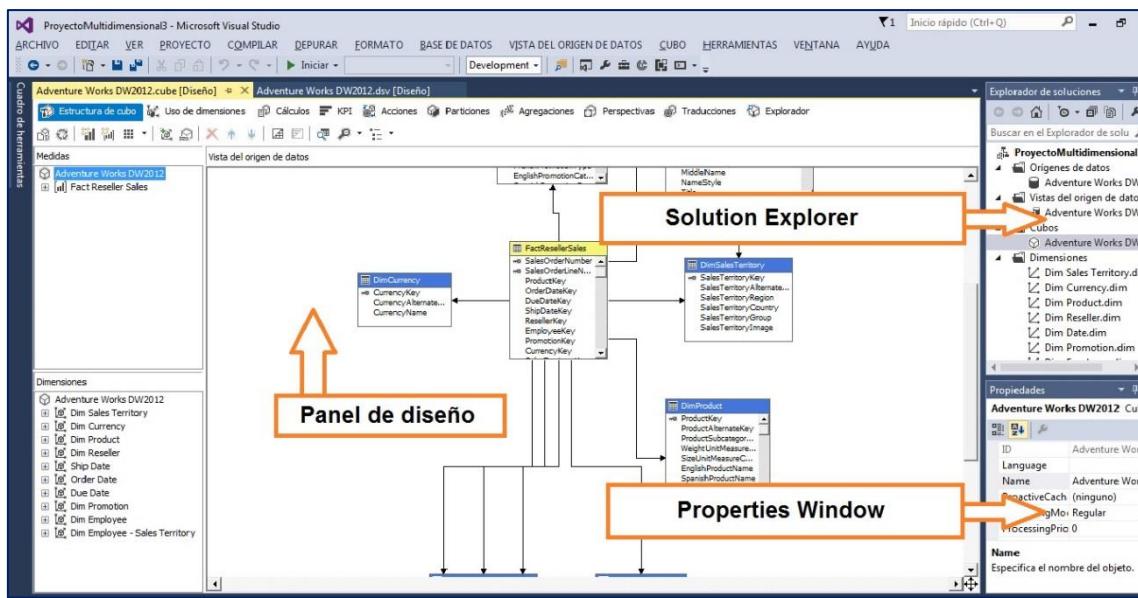
- Mejoras en la interfaz de usuario

El desarrollo de una solución de Business Intelligence a través del entorno de desarrollo de Microsoft Visual Studio .Net permite un desarrollo más rápido y una mejor definición de los componentes del cubo.

Para definir cubos en el SQL Server Data Tools, se debe crear un proyecto de tipo Analysis Services Project.



La interfaz de desarrollo tiene la siguiente apariencia:



A continuación se describen algunos de los componentes del entorno de diseño:

1. **Panel de diseño**: Muestra gráficamente los componentes que constituyen el cubo y las dimensiones.
2. **Solution Explorer Window (explorador de soluciones)**: Muestra, en una vista de árbol, a los elementos que componen el proyecto: orígenes de datos (Data Sources), vistas de origen de datos (Data Source Views), cubos, dimensiones, modelos de minería de datos, etc.
3. **Properties Window (ventana de propiedades)**: Muestra las propiedades configurables del cubo, las dimensiones y sus componentes.

1.2.3. Uso de Data Source y Data Source View

Data Sources y Data Source Views

A través de los Data Sources (orígenes de datos), se establece la ubicación de las bases de datos donde residen los esquemas STAR y SNOWFLAKE a partir de los cuales se crearán los cubos. Los data sources pueden conectarse a una gran variedad de orígenes de datos: SQL Server, Oracle, DB2, etc. Se pueden definir múltiples data sources en un proyecto de Business Intelligence,

Los Data Source Views (vistas de orígenes de datos) son objetos que contienen el esquema (estructura) de las tablas que conforman el STAR o SNOWFLAKE. Se crean a partir de los data sources. Una data source view almacena:

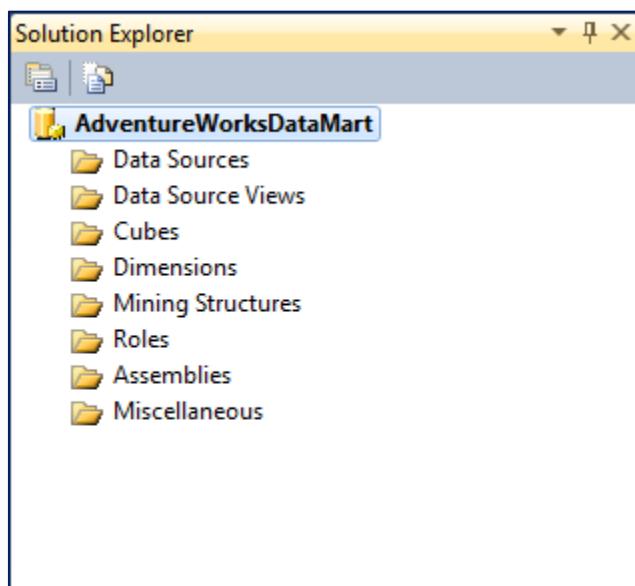
- La estructura de las tablas de conforman el STAR o SNOWFLAKE.
- Las relaciones entre las tablas. En un data source view, dichas relaciones se obtienen automáticamente a partir de las claves foráneas existentes en las tablas. Sin embargo, el desarrollador puede definir nuevas relaciones directamente en el data source view.
- Nombre amigables para las tablas y sus columnas.
- Nuevas definiciones de columnas, escritas por el desarrollador a partir de expresiones de cálculo.

Los data source views se almacenan de forma permanente en el proyecto de análisis; por tanto, permiten trabajar sin necesidad de una conexión en línea con las bases de datos de origen.

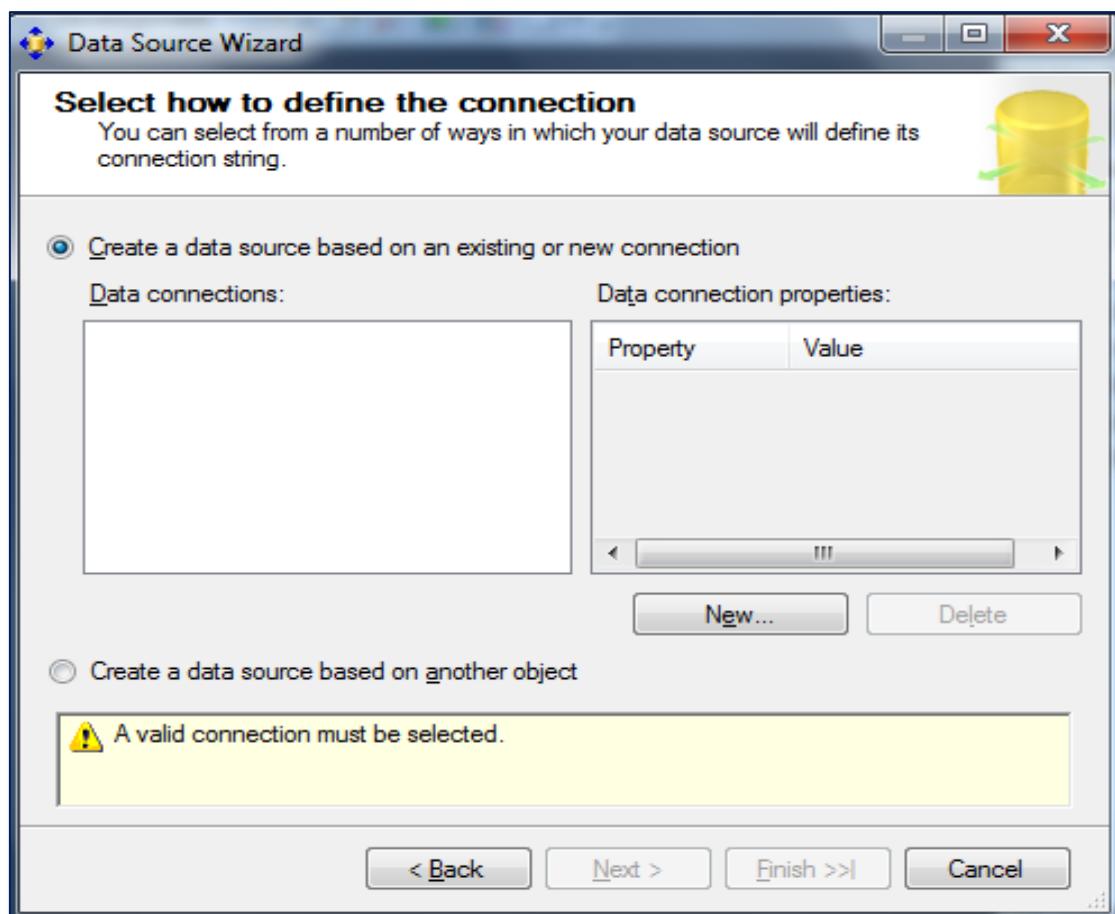
1.2.4. Implementación de cubos en Analysis Services

En el siguiente ejercicio contruiremos un proyecto Analysis Services, el cual contendrá los orígenes de datos y vista de orígenes de datos que nos permitirán posteriormente crear los cubos de información.

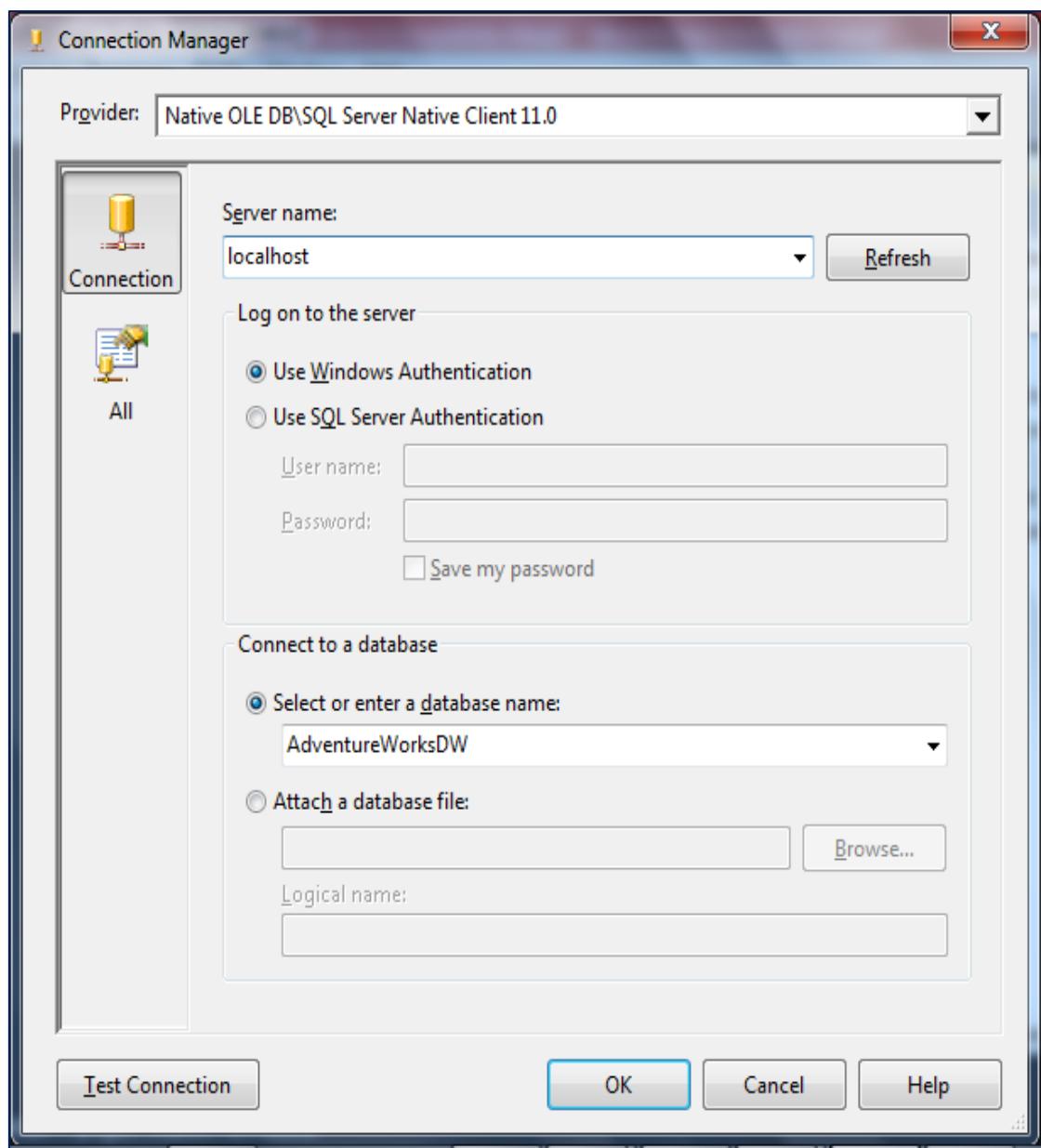
1. Desde el botón Start > All Programs > Microsoft SQL Server 2012 inicie el SQL Server Data Tools.
2. Se abrirá el entorno de desarrollo de proyectos de Business Intelligence. En el menú File > New, seleccione el ítem Project. Se abrirá una ventana que muestra los distintos tipos de templates de proyecto que se pueden crear. Seleccione el ítem Analysis Services Project. En la caja de texto Name, escriba "AdventureWorksDataMart". En la caja de texto Location, seleccione su carpeta de trabajo. Pulse el botón OK para crear el proyecto.
3. Observe la ventana del Solution Explorer, al lado derecho de su pantalla (si no ve esta ventana, en el menú View seleccione el ítem Solution Explorer). Esta ventana muestra los elementos que constituyen su solución de Analysis Services:



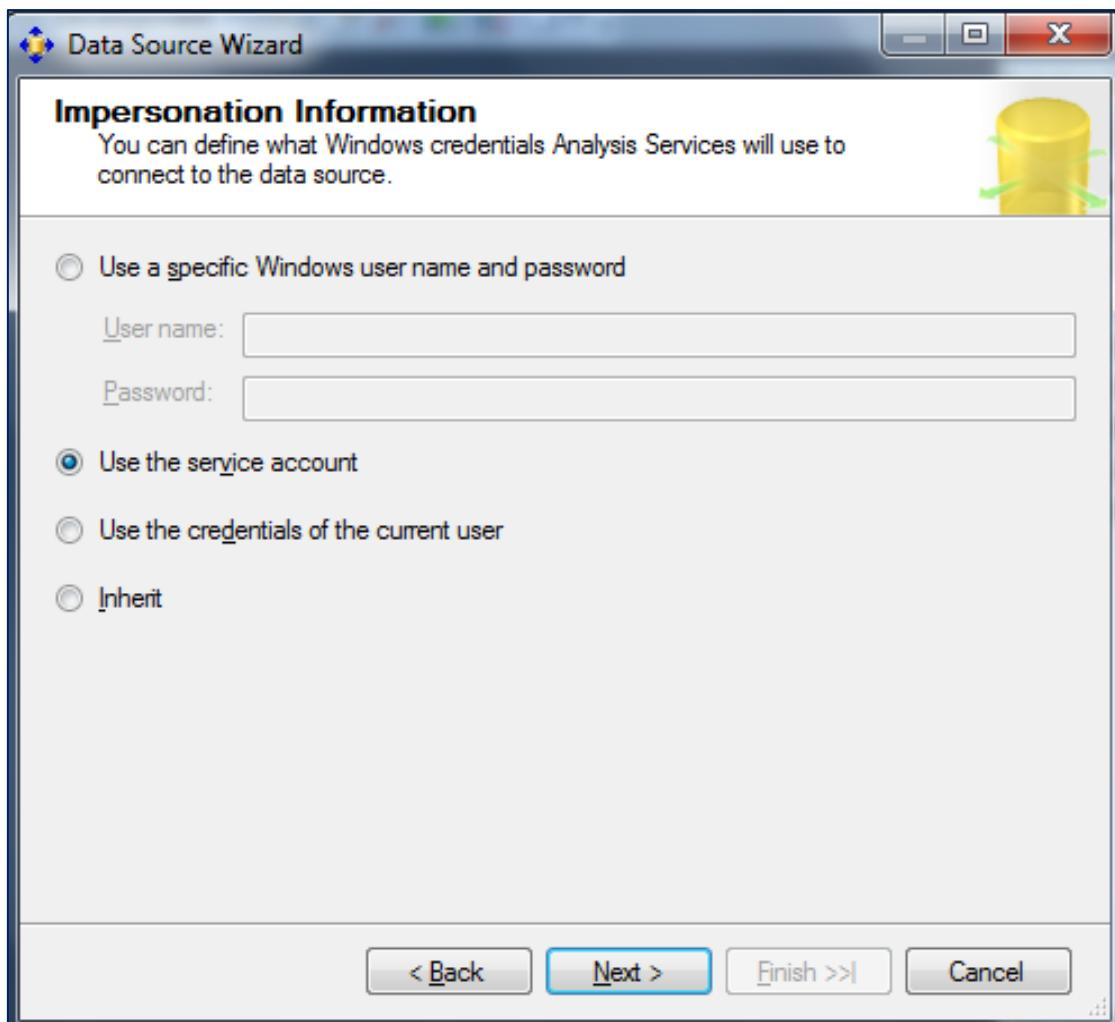
4. En el Solution Explorer, seleccione el nodo Data Sources. Haga un clic derecho, y seleccione la opción New Data Source. Aparecerá la pantalla de bienvenida del asistente de creación de data sources. Pulse el botón Next.
5. Aparecerá la ventana Select how to define the connection. Si aparecen conexiones existentes en la lista Data connections, elimínelas con el botón Delete.



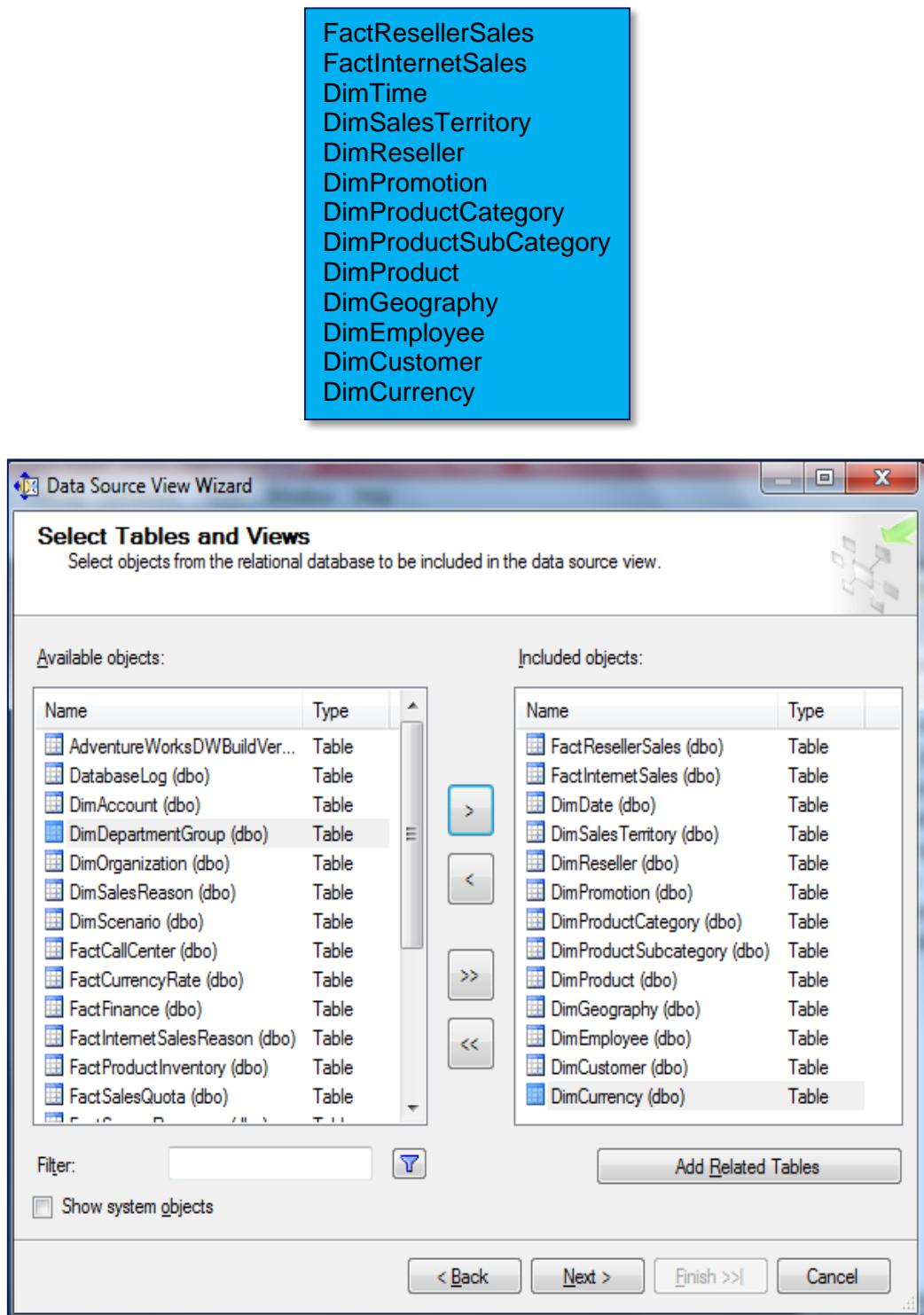
6. Pulse el botón New. Aparecerá la ventana de definición de data sources. Seleccione el proveedor Native OLE DB\SQL Native Client 11.0. Establezca la propiedad Server Name a “localhost”. Seleccione la base de datos “AdventureWorksDW”.



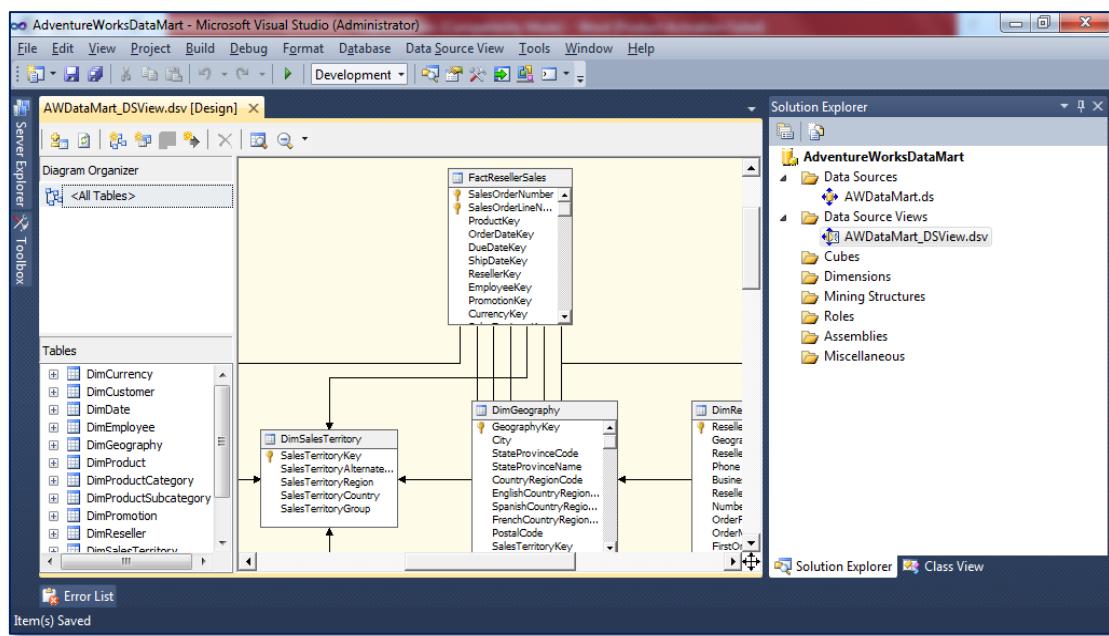
7. Pulse el botón Test Connection para verificar que la conexión se efectúa exitosamente. Presione el botón OK. A continuación, pulse el botón Next para continuar con el asistente.
8. Aparecerá la ventana Impersonation Information. A través de esta ventana, se establecen las credenciales que se utilizarán para la conexión con el origen de datos, al efectuar el procesamiento del cubo. Seleccione la opción Use the service account. De esta forma, se utilizará la cuenta de inicio de servicio de Analysis Services para efectuar la conexión con el origen de datos.



9. Pulse el botón Next. En la ventana final del asistente, establezca el nombre del nuevo data source a “AWDataMart”. Pulse el botón Finish para finalizar el asistente.
10. En el Solution Explorer, seleccione el nodo Data Source Views. Haga un clic derecho, y seleccione la opción New Data Source View.
11. Aparecerá la ventana inicial del asistente de creación de vistas de orígenes de datos. Pulse el botón Next.
12. La siguiente ventana mostrará la lista de data sources existentes en el proyecto de análisis. En el lado izquierdo de la ventana, seleccione el data source “AWDataMart”, y pulse el botón Next.
13. La siguiente ventana mostrará las tablas y vistas disponibles en el data source “AWDataMart”. Se deben seleccionar aquellas tablas y vistas que conformarán el data source view. Seleccione las siguientes tablas con el botón >.



14. Pulse el botón Next. Aparecerá la ventana final del asistente. Establezca el nombre del nuevo data source view a "AWDataMart_DSVIEW". Pulse el botón Finish.
15. Observe el entorno de diseño. Todas las definiciones de las tablas han sido copiadas al Data Source View, y se muestran de forma gráfica:



El asistente de creación de cubos (Cube Wizard)

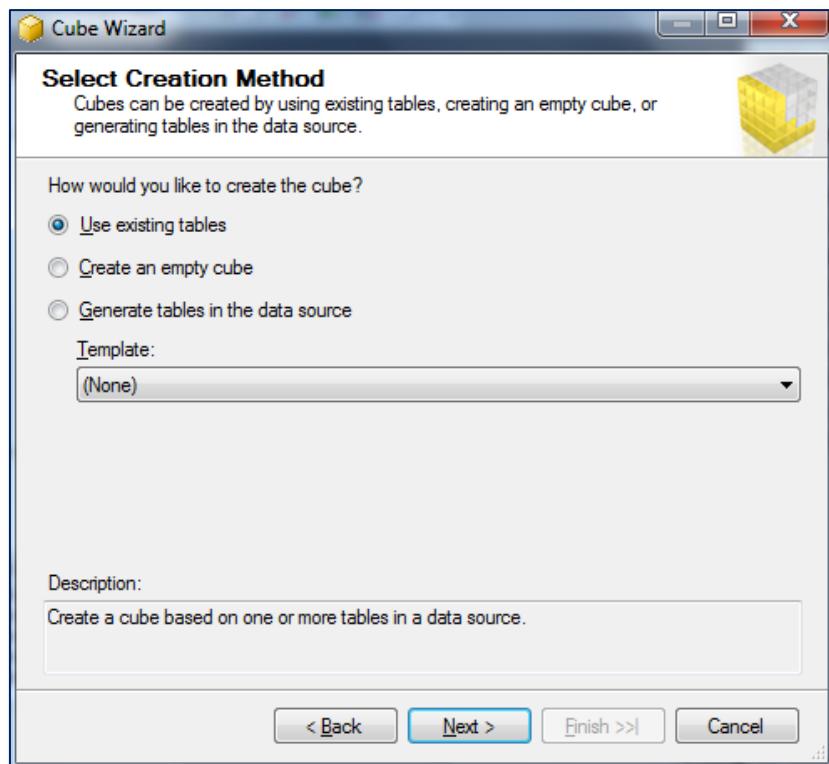
El entorno de desarrollo de proyectos de análisis incluye un asistente para la definición de cubos. Dicho asistente incluye dos opciones para la generación de cubos:

- Build the cube using a data source: Esta opción obtiene las definiciones del cubo a partir de la estructura existente en una base de datos, donde reside un esquema STAR o SNOWFLAKE. Si se selecciona la opción Auto build, el asistente analizará la estructura de las tablas, y detectará automáticamente las tablas de hechos y tablas de dimensión. Además, el asistente podrá determinar cuáles serán los niveles que se agregarán a las dimensiones. Si la opción Auto build no se selecciona, el desarrollador definirá manualmente la estructura del cubo.
- Build the cube without using a data source: Esta opción permite que el desarrollador defina la estructura del cubo sin necesidad de elegir un data source view. Posteriormente, a partir de la estructura propuesta por el desarrollador, SSAS generará las definiciones del STAR o SNOWFLAKE. SSAS provee dos plantillas de diseño: una de ellas es compatible con las características de la versión Enterprise Edition, y la otra con las opciones disponibles en la versión Standard Edition.

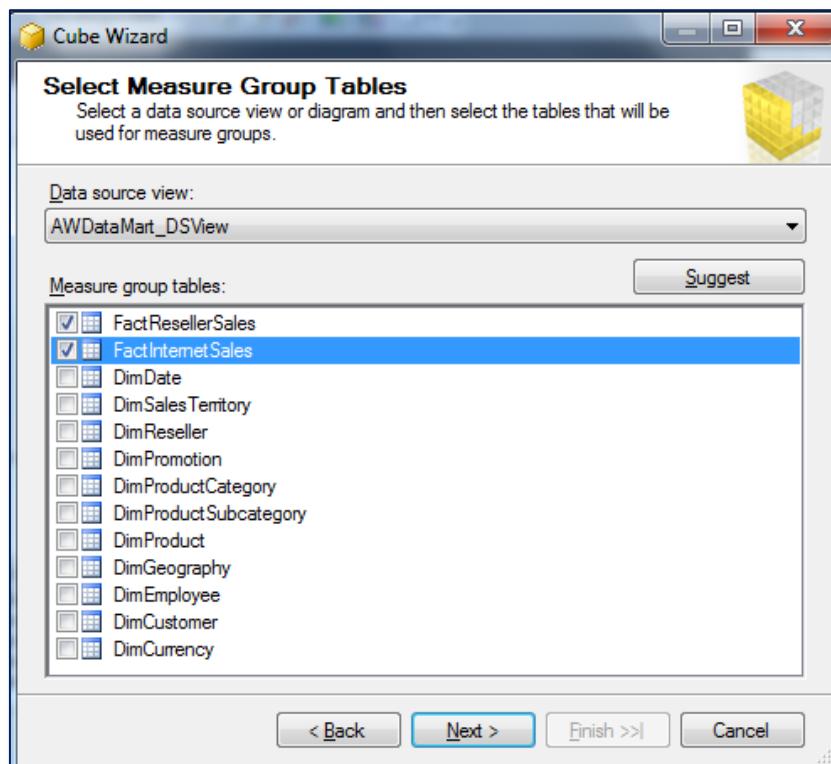
Cualquiera de las características y definiciones establecidas por el asistente pueden ser modificadas por el desarrollador en un momento posterior.

En el siguiente ejercicio construiremos un cubo de información a través del Asistente de Cubos (Cube Wizard).

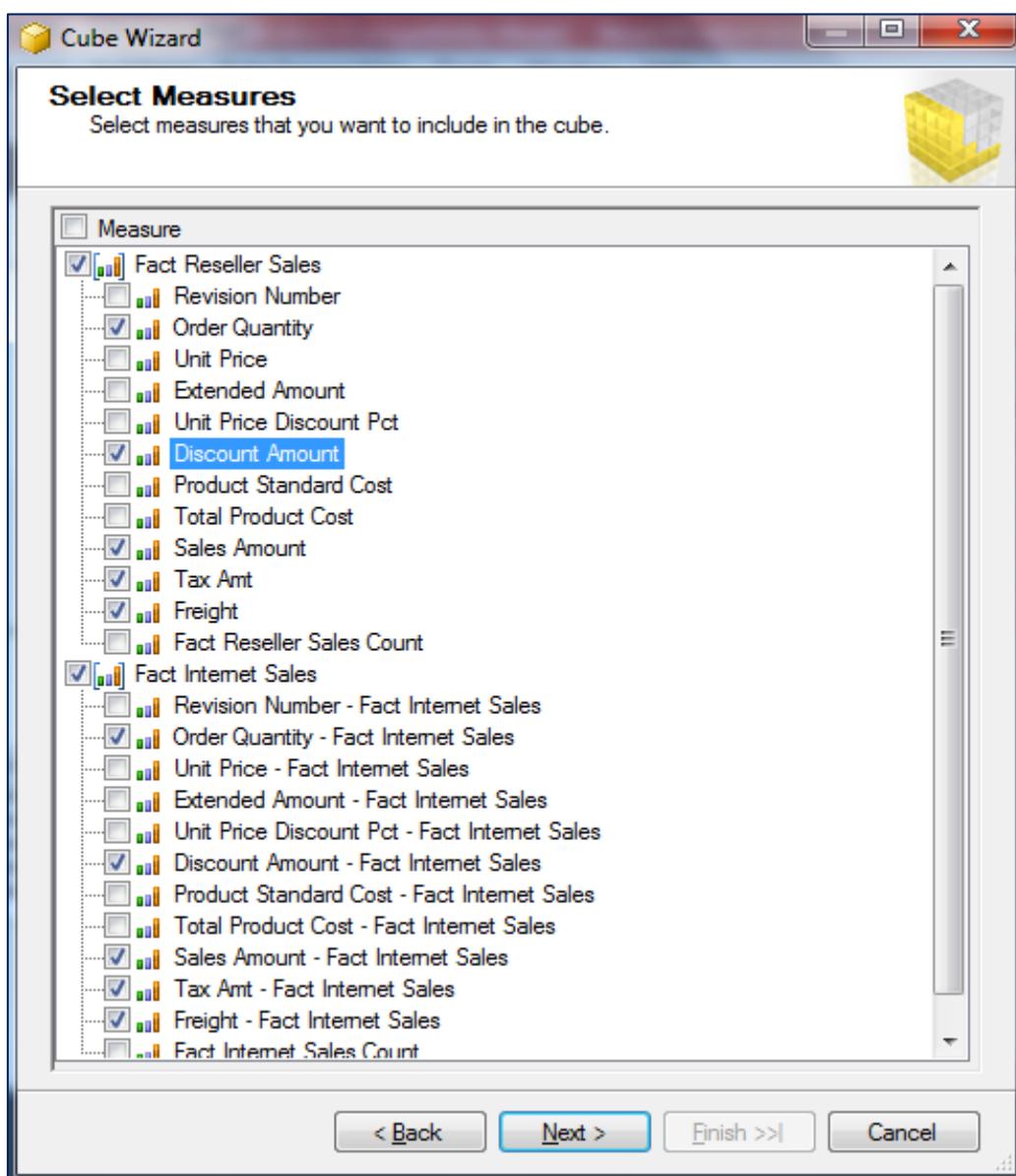
1. En el Solution Explorer, seleccione el nodo Cubes. Haga un clic derecho, y seleccione la opción New Cube.
 2. En la ventana inicial del asistente, pulse el botón Next.
 3. La siguiente ventana permite elegir el método de construcción del cubo. Observe que la opción Use existing tables se encuentra seleccionada, esto significa que el asistente analizará las tablas del data source view, y a partir de ellas inferirá la estructura del cubo.
- Pulse el botón Next para continuar.



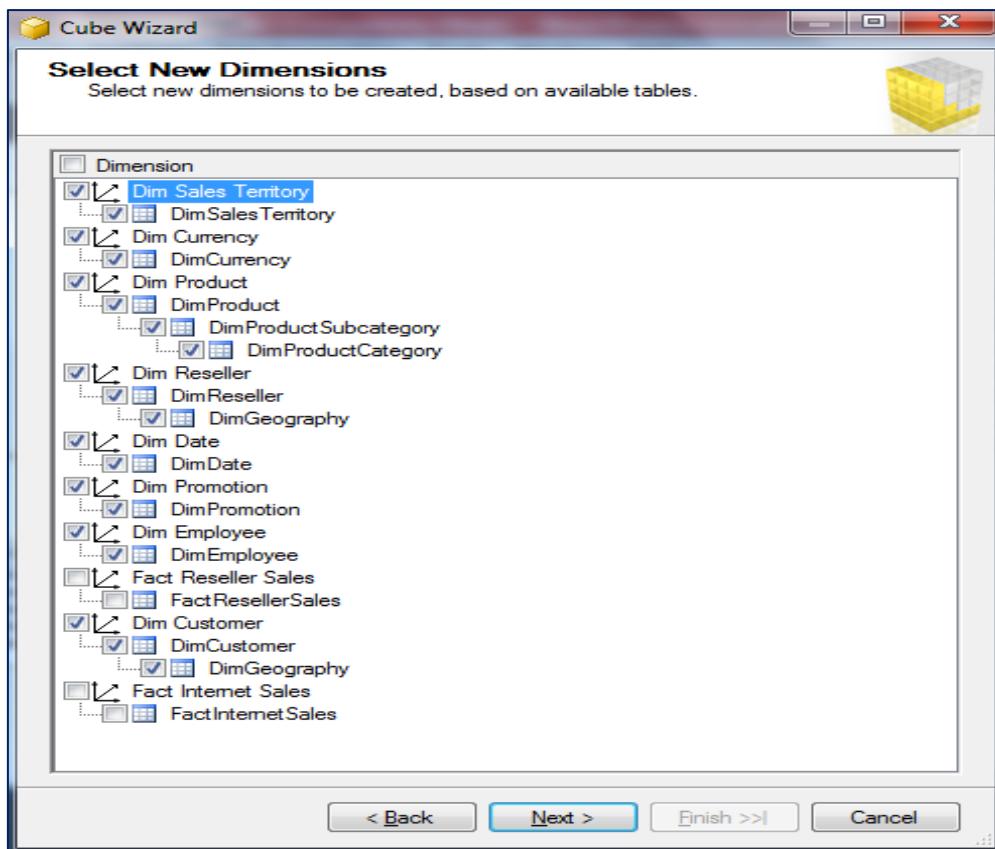
4. La siguiente ventana permite elegir el data source view que se utilizará para definir la estructura del cubo. Asimismo permite elegir las tablas que contienen las medidas de nuestro datamart. Presione el botón Suggest para que el asistente nos sugiera las tablas de hechos. Observe que la tabla “DimReseller” ha sido identificada como tabla de hechos. Para corregir esto, desmarque el checkbox. Luego pulse el botón Next.



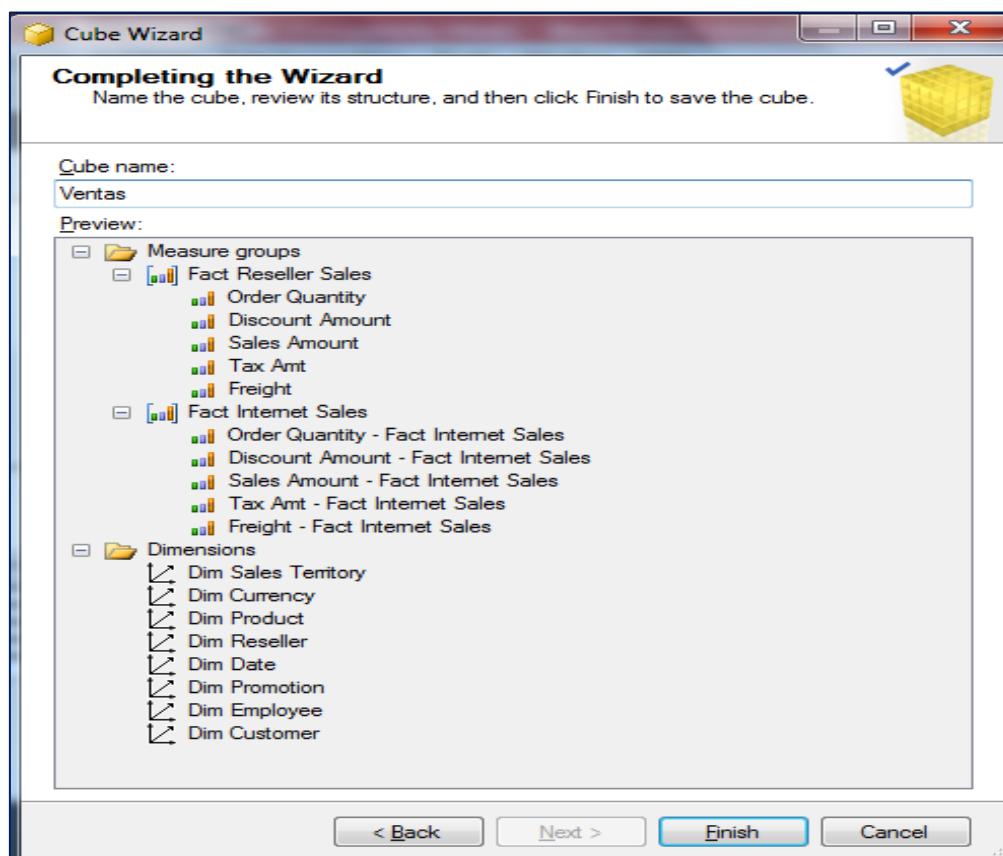
5. La siguiente ventana permite determinar cuáles son las medidas que tendrá el cubo. En SSAS, las medidas se agrupan en Measure Groups (grupos de medidas). Existe un measure group por cada tabla de hechos del cubo. Observe que, en el cubo que está construyendo, existen dos measure groups: uno para la tabla “FactResellerSales” y otro para la tabla “FactInternetSales”. Cada columna numérica de las tablas de hechos es reconocida como potenciales medidas del cubo. Desmarque todos los checkboxes de la ventana. Bajo el measure group “Fact Reseller Sales”, seleccione las medidas “Order Quantity”, “Discount Amount”, “Sales Amount”, “Tax Amt” y “Freight”. Bajo el measure group “Fact Internet Sales”, seleccione las medidas “Order Quantity – Fact Internet Sales”, “Discount Amount – Fact Internet Sales”, “Sales Amount – Fact Internet Sales”, “Tax Amt – Fact Internet Sales” y “Freight – Fact Internet Sales”.



6. La siguiente ventana muestra las dimensiones obtenidas por el asistente. Observe que las tablas Fact Reseller Sales y Fact Internet Sales han sido detectados como dimensiones. Para corregir esto, desmaque los checkboxes.



Pulse el botón Next. Aparecerá la ventana final del asistente. Establezca el nombre del nuevo cubo a “Ventas”.



Guarde los cambios.

El diseñador de cubos

El diseñador de cubos permite establecer las propiedades del cubo y sus dimensiones. Provee una interfaz gráfica de fácil utilización, a través de la cual se puede personalizar el diseño de cada cubo en la solución de análisis. Este diseñador posee las siguientes pestañas, visibles en la parte superior de la pantalla:

- Cube Structure: Permite efectuar cambios en la estructura del cubo: medidas, dimensiones, propiedades de agregación, etc.
- Dimension Usage: Permite definir las relaciones existentes entre el cubo y sus dimensiones.
- Calculations: Permite definir cálculos personalizados en el cubo a través de sentencias MDX (Multidimensional Expressions).
- KPIs: Permite definir Key Performance Indicators (KPIs) para establecer valores comparativos contra los cuales deben evaluarse las medidas.
- Actions: Permite definir las acciones que los usuarios pueden efectuar al navegar a través de la información del cubo.
- Partitions: Permite definir las particiones en que se almacena el cubo y los modos de almacenamiento de cada una de ellas.
- Perspectives: Permite definir perspectivas. Cada perspectiva constituye una vista parcial de la información, disponible para un grupo de usuarios.
- Translations: Permite traducir los nombres de los objetos en el cubo a múltiples idiomas.
- Browser: Permite observar la información almacenada en el cubo.

La figura superior muestra el diseñador de cubos y las ventanas que constituyen la vista Cube Structure.

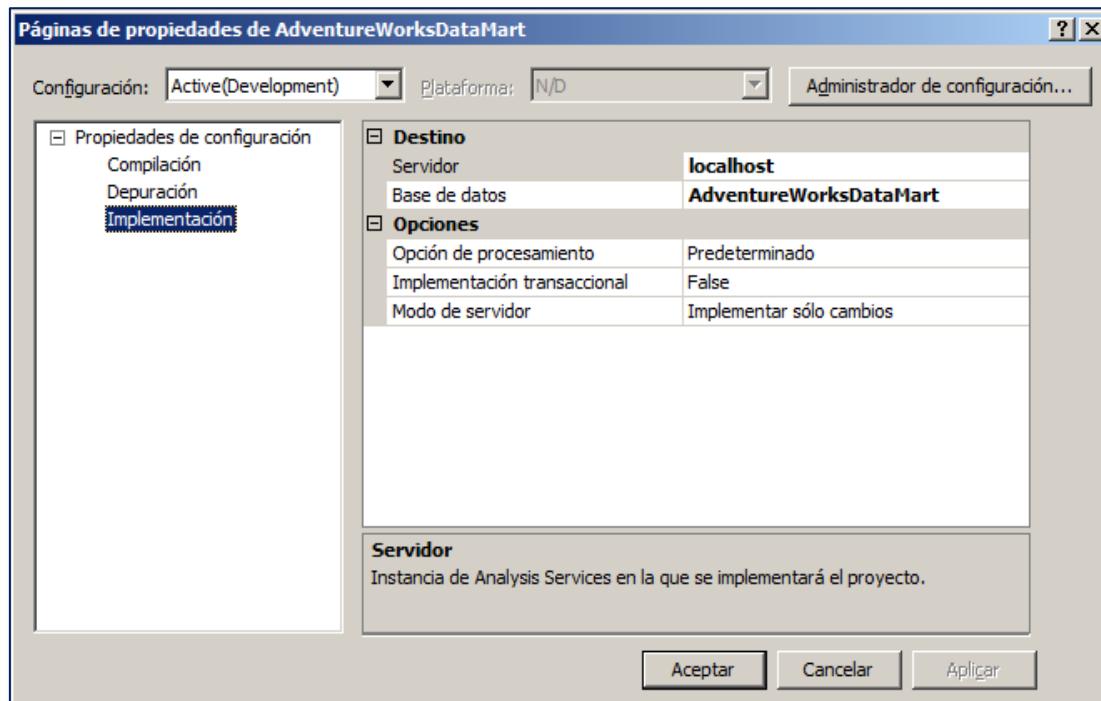
Despliegue (deployment) y procesamiento del cubo



Una vez finalizado el diseño del cubo, se debe crear físicamente el cubo en un servidor de análisis. Este proceso, conocido como deployment, consiste en la generación de un script que se enviará a un servidor de análisis, y que generará el cubo y sus dimensiones. A continuación, se efectuará el procesamiento del cubo. Durante la fase de procesamiento, el servidor de análisis se conecta con el data source donde residen las tablas de hechos y las tablas de dimensión, y extrae su información para poblar el cubo.

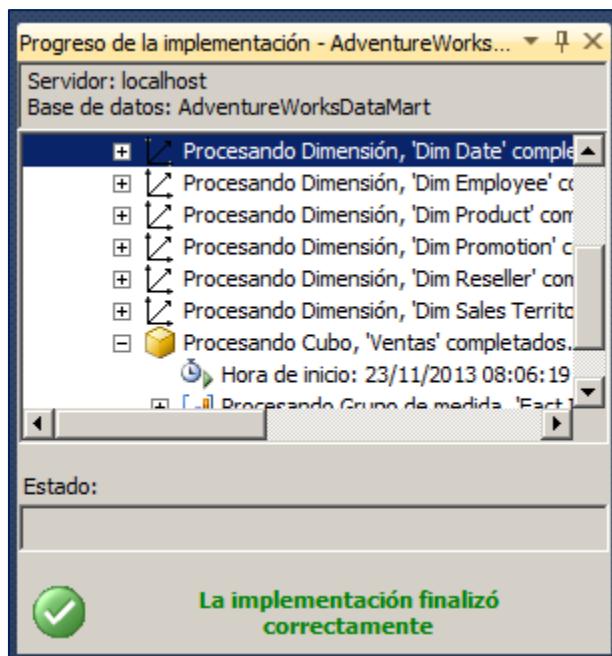
En el siguiente ejercicio implementaremos (deployment) el cubo de información a través del SQL Server Data Tools (SSDT).

1. En el Solution Explorer, seleccione el proyecto “AdventureWorksDataMart”. Haga un clic derecho, y seleccione la opción Properties.
2. Se abrirá la página de propiedades del proyecto. Existen tres categorías de propiedades: Build (compilación), Debugging (depuración) y Deployment (Implementación). En la parte izquierda de la pantalla, seleccione la opción Deployment. Observe los valores de las propiedades establecidas:



La propiedad Server está establecida al “localhost”. Esto significa que el servidor de análisis es la máquina local. Si el servidor se encuentra en otra ubicación, basta con colocar su nombre en esta propiedad. La propiedad Database determina el nombre de la base de datos de análisis que se creará al distribuir el proyecto. Observe que su valor coincide, por defecto, con el nombre del proyecto de análisis, en este caso “AdventureWorksDataMart”. No efectúe ningún cambio. Pulse el botón Aceptar para cerrar la página de propiedades.

3. En el Solution Explorer, seleccione el proyecto “AdventureWorksDataMart”. Haga un clic derecho, y seleccione la opción Deploy. Esto iniciará la distribución del proyecto de análisis. Espere unos minutos mientras finaliza el procesamiento del cubo. Al terminar, en la ventana Deployment Progress (situada en la parte inferior derecha de la ventana de diseño) aparecerá un mensaje de confirmación.



4. En la ventana de diseño de cubos, en la parte superior, seleccione el tab Browser (es el último tab, de izquierda a derecha). A través de este tab, se puede visualizar la información del cubo. En la parte izquierda de la ventana, expanda el measure group "Fact Reseller Sales". Seleccione la medida "Sales Amount" y arrástrela sobre el panel de datos (es la zona en blanco al lado derecho inferior). Haga lo mismo con la medida "Sales Amount – Fact Internet Sales" del measure group "Fact Internet Sales". A continuación, expanda la dimensión "Order Date". Seleccione el atributo "Order Date.Date Key" y arrástrelo sobre el panel de datos, al lado izquierdo de la medida Sales Amount. Luego, expanda la dimensión "Dim Product", seleccione el atributo "Product Category Key" y arrástrelo sobre el panel de datos al lado izquierdo del atributo Date Key. Observe cómo se muestra la información.

The screenshot shows the Microsoft Visual Studio interface for the AdventureWorksDataMart project. The main window displays the 'Ventas.cube [Design]' tab, which contains a data grid with the following columns: Product Category Key, Date Key, Sales Amount, and Sales Amount - Fact Internet S... . The data grid shows numerous rows of sales data. To the left of the grid, there is a navigation pane with tabs like Cube Structure, Dimension Usage, Calculations, KPIs, Actions, Partitions, Aggregations, Perspectives, Translations, and Browser. The 'Browser' tab is currently selected. The 'Solution Explorer' on the right shows the project structure, including the 'AdventureWorksDataMart' database with its data sources, views, cubes, dimensions, and measures. The 'Deployment Progress' window at the bottom indicates that the deployment was successful.

5. Efectúe otros cruces de información, según su propio criterio.



IMPLEMENTANDO SOLUCIONES OLAP CON MICROSOFT SQL SERVER

LOGRO DE LA UNIDAD DE APRENDIZAJE

Al término de la unidad, el alumno implementa cubos de análisis definiendo medidas, personalizando dimensiones e indicadores de rendimiento, utilizando MS SQL Server Analysis Services.

TEMARIO

2.1 Tema 3 : Definición de estructura de Dimensiones

- 2.1.1 : Creando Dimensiones
- 2.1.2 : Definiendo tipo de Dimensiones
- 2.1.3 : Definiendo Jerarquías
- 2.1.4 : Creando Dimensión Tiempo
- 2.1.5 : Documentación de los elementos de diseño
- 2.1.6 : Agrupando Miembros de un atributo

2.2 Tema 4 : Definiendo Medidas

- 2.2.1 : Creando cubos, medidas y grupos de medidas
- 2.2.2 : Relacionando dimensiones y grupos de medidas

2.3 Tema 5 : Definición de Miembros Calculados y KPI

- 2.3.1 : Creando Miembros calculados
- 2.3.2 : Creando KPI

2.4 Tema 6 : Definición de Traducciones y Perspectivas

- 2.4.1 : Creando Traducciones
- 2.4.2 : Creando Perspectivas

ACTIVIDADES PROPUESTAS

4. Los alumnos identifican Dimensiones.
5. Los alumnos identifican y definen medidas.
6. Los alumnos identifican miembros calculados y KPI
7. Los alumnos identifican traducciones y perspectivas.

2.1. DEFINICIÓN DE ESTRUCTURAS DE DIMENSIONES

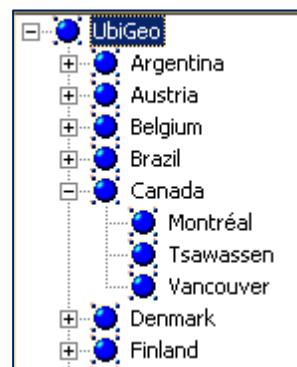
2.1.1. Creando Dimensiones

Las dimensiones son definidas a partir de los criterios utilizados por los usuarios para consultar la información. Por ejemplo, si los usuarios desean obtener las ventas por producto, región y trimestre, esto indica que se deben definir las dimensiones Producto, UbiGeo y Tiempo.

En la unidad anterior, se implementaron diversas dimensiones para los cubos Ventas y Sales, a través del Cube Wizard. En el presente capítulo, se mostrará cómo configurar las propiedades de las dimensiones a través del entorno de desarrollo de SQL Server Data Tools.

Miembros (members) y atributos de una dimensión

La figura de la derecha muestra a los elementos de la Dimensión UbiGeo.



Al formular el diseño de una dimensión, se deben identificar las propiedades relevantes para el negocio. Por ejemplo, la dimensión UbiGeo debe contener las propiedades “país” y “ciudad”. SSAS modela estas propiedades a través de atributos (attributes). Cada atributo definido en una dimensión corresponde con alguna propiedad de interés para el modelo de negocios. De acuerdo con esto, la dimensión UbiGeo contendría los atributos País y Ciudad.

Un miembro (member) representa una ocurrencia específica de una dimensión. Por ejemplo, en la figura de la derecha, Argentina, Austria, Canadá, Montreal, etc. son miembros de la dimensión UbiGeo.

Cada miembro en una dimensión corresponde con un atributo. Por ejemplo, en la dimensión UbiGeo el miembro Argentina corresponde con el atributo País, y el miembro Montréal con el atributo Ciudad.

Jerarquías y niveles

En una dimensión, un atributo puede “contener” a otro. Por ejemplo, en la dimensión UbiGeo, el atributo País “contiene” al atributo Ciudad, pues cada ciudad se encuentra en un determinado país. Este concepto permite efectuar operaciones de Drill – Down a través de las dimensiones; de esta forma, se pueden visualizar las ventas del Perú, y luego visualizar las ventas de cada una de sus ciudades.

En SSAS, esta agrupación jerárquica de los atributos se define a través de jerarquías (hierarchies). La estructura de una jerarquía se define a través de niveles (levels), cada uno de los cuales está mapeado a un atributo en particular. Por ejemplo, los atributos País y Ciudad de la dimensión UbiGeo pueden utilizarse para definir una

jerarquía llamada Ubicación, cuyos niveles son País y Ciudad. Las jerarquías se denotan con la notación nombre_de_dimensión.nombre_de_jerarquía. Por ejemplo, la jerarquía Ubicación de la dimensión Ubigeo tendría la siguiente representación:

- Ubigeo.Ubicación
 - País
 - Ciudad

Este diseño especifica que cada país contiene varias ciudades, y cada ciudad se encuentra en un solo país.

La siguiente figura muestra las relaciones existentes entre miembros y niveles en la dimensión UbiGeo:



Una dimensión puede contener varias jerarquías. Cada jerarquía definirá una forma distinta de agrupar y sumarizar la información de una dimensión. Existen entidades de negocio que no tienen un único criterio de agrupación; por ejemplo, los clientes pueden ser desglosados por su ubicación geográfica (país, región y ciudad) y también por su género (masculino o femenino).

Ejemplo

Considérese el siguiente escenario

Se desea obtener indicadores de rendimiento de los empleados de la corporación. Los reportes de evaluación de los empleados deben ser totalizados de acuerdo a las divisiones organizacionales y regiones geográficas de la corporación. Cada empleado trabaja para una división organizacional. A su vez, cada empleado pertenece a una ciudad, y cada ciudad está contenida en una región. ¿Cuál es la manera más adecuada de definir los niveles de la dimensión Empleado?

De acuerdo con el problema, se tienen dos maneras de desglosar la información de los empleados: por división organizacional y por ubicación geográfica. No puede afirmarse que las divisiones organizacionales están contenidas en ubicaciones geográficas, pues pueden existir divisiones que se encuentren dispersas a través de varios sitios geográficos. Por tanto, la estructura más adecuada para la dimensión Empleado consiste en definir dos jerarquías de la siguiente manera:

Empleado.Región

- Región
- Ciudad
- Empleado

Empleado.División

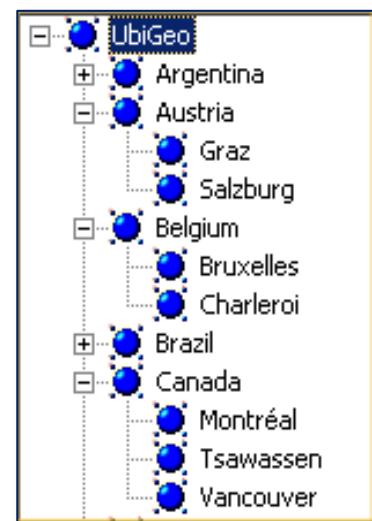
- División
- Empleado

Relaciones entre los miembros de una dimensión

Existen ciertas relaciones de parentesco entre los miembros de una dimensión. Dichas relaciones se basan en la posición relativa de los miembros.

El siguiente gráfico muestra los miembros de la dimensión UbiGeo. Las relaciones que se pueden identificar son las siguientes:

- Austria es el parent (parent) de Graz y Salzburg.
- Austria y Belgium son hermanos (siblings).
- Graz y Salzburg son hermanos (siblings).
- Graz y Salzburg son hijos (children) de Austria.
- Los ancestros (ancestors) de Graz son Ubigeo y Austria.
- Los descendientes (descendants) de Ubigeo son todos los miembros de la dimensión (Argentina, Austria, Graz, Salzburg, etc.)
- Graz y Salzburg son primos (cousin) de Bruxelles y Charleroi.



2.1.2. Definiendo tipos de Dimensiones

Dimensiones Estándar

El tipo más sencillo de dimensión está constituido por las dimensiones estándar. En estas dimensiones, cada nivel está representado por una columna en la tabla de dimensión.

Una característica de las dimensiones estándar es que todos los miembros pertenecientes a un determinado nivel tienen el mismo número de ancestros. Por ejemplo, en la dimensión UbiGeo, todos los miembros del nivel City tienen el mismo número de ancestros (su respectivo miembro padre en la dimensión Country, y el miembro All Ubigeo).

Propiedades de los atributos

Los atributos tienen propiedades que pueden ser configuradas a través del editor de dimensiones. A continuación se explicarán algunas de estas propiedades:

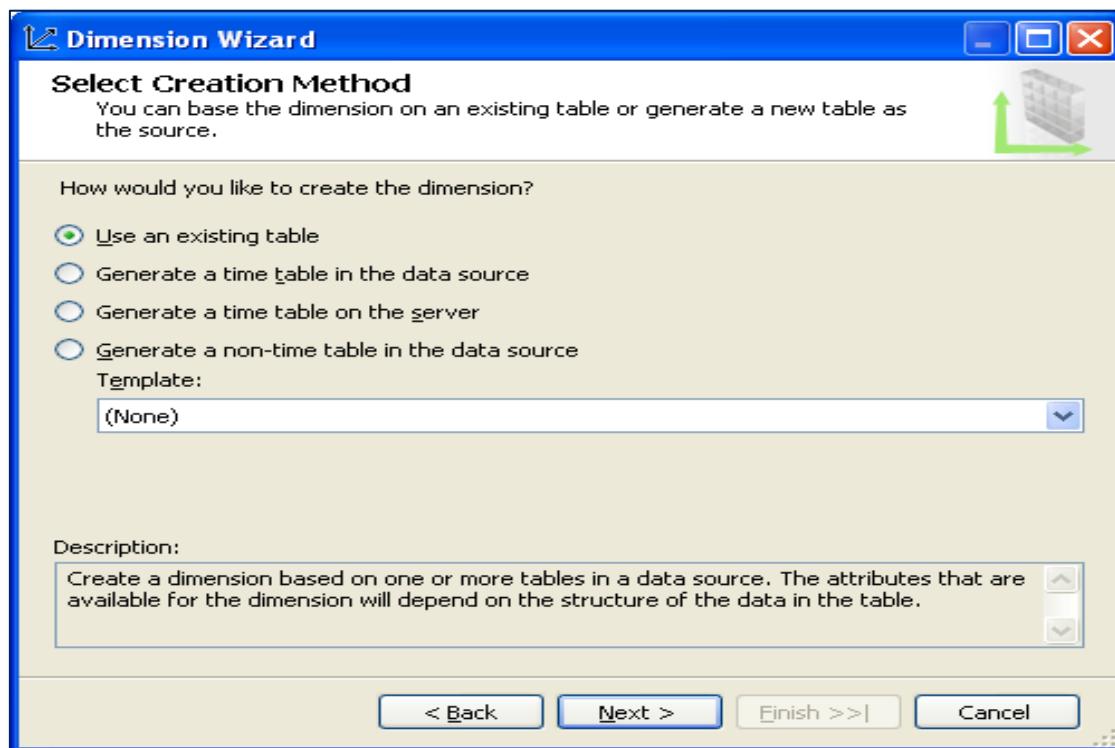
- Name: Establece el nombre del atributo.
- Usage: Establece el propósito que el atributo debe cumplir en la estructura de la dimensión. Existen tres tipos de uso y los siguientes:
 - Key: El atributo se utiliza para identificar de manera única a cada miembro de la dimensión, y se utilizará para relacionar a la dimensión

- con las tablas de hechos. Sólo puede existir un atributo de tipo Key en una dimensión, y toda dimensión debe tener un atributo Key.
- Parent: El atributo se utiliza para definir una dimensión Parent – Child.
 - Regular: El atributo se utiliza sólo para mostrar información.
 - Propiedad KeyColumns: Establece el identificador único de los miembros de un atributo. Dicho identificador puede estar compuesto por una o varias columnas de la tabla de dimensión. Por ejemplo, el identificador del atributo Mes de la dimensión Tiempo puede estar compuesto por las columnas “Año” y “NúmeroMes” de la tabla de dimensión.
 - Propiedad NameColumn: Esta propiedad determina la columna de la tabla de dimensión a partir de la cual se obtiene el valor que mostrarán los miembros de un atributo. Por ejemplo, en la dimensión Tiempo, la propiedad NameColumn del atributo Mes puede apuntar hacia la columna “NombreMes” de la tabla de dimensión.

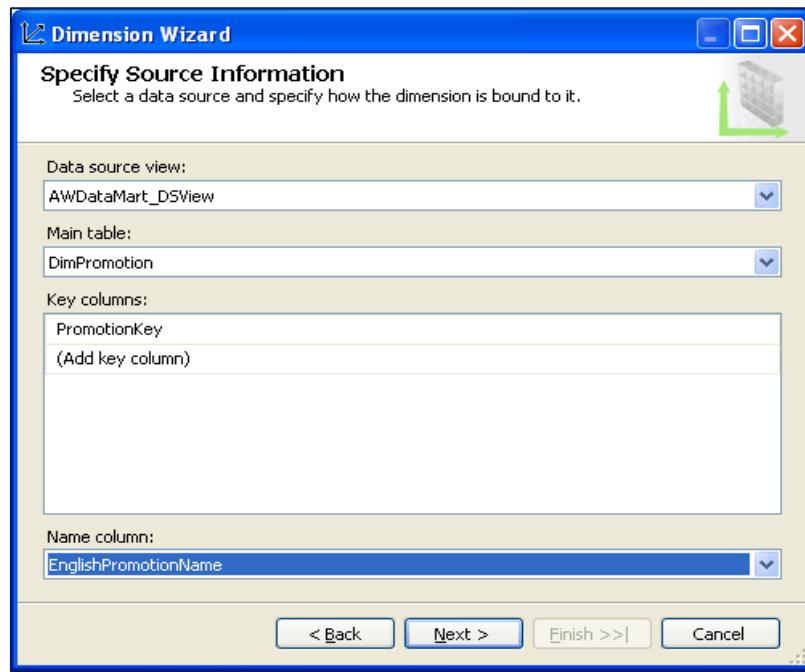
Ejercicio 1

Cree las dimensiones PROMOTION y CURRENCY.

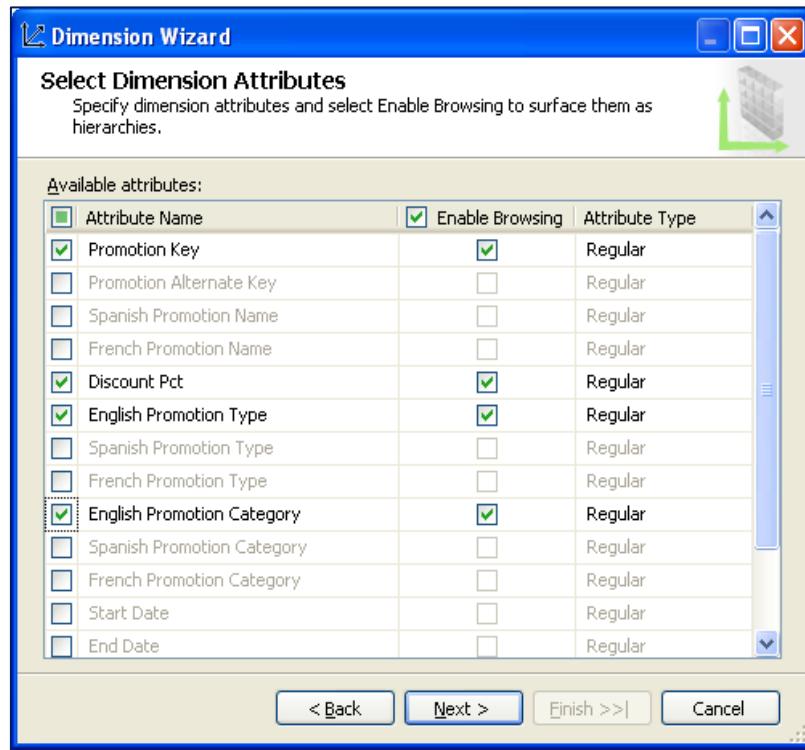
1. A través del SQL Server Data Tools, abra el proyecto “AdventureWorksDataMart”, desarrollado en el capítulo anterior.
2. En el Solution Explorer, ubique el cubo “Ventas”. Haga un clic derecho, y seleccione la opción Delete para eliminar el cubo.
3. En el Solution Explorer, seleccione la dimensión “Dim Time”. Haga un clic derecho, y seleccione la opción Delete para eliminar la dimensión. Repita este paso para todas las dimensiones restantes.
4. En el Solution Explorer, seleccione el folder Dimensions. Haga un clic derecho, y seleccione la opción New Dimension. Esto iniciará el asistente para la creación de dimensiones. En la primera ventana del asistente, presione el botón Next.
5. La siguiente ventana permite definir el método que se usará para construir la dimensión. Mantenga la opción por default Use an existing table.



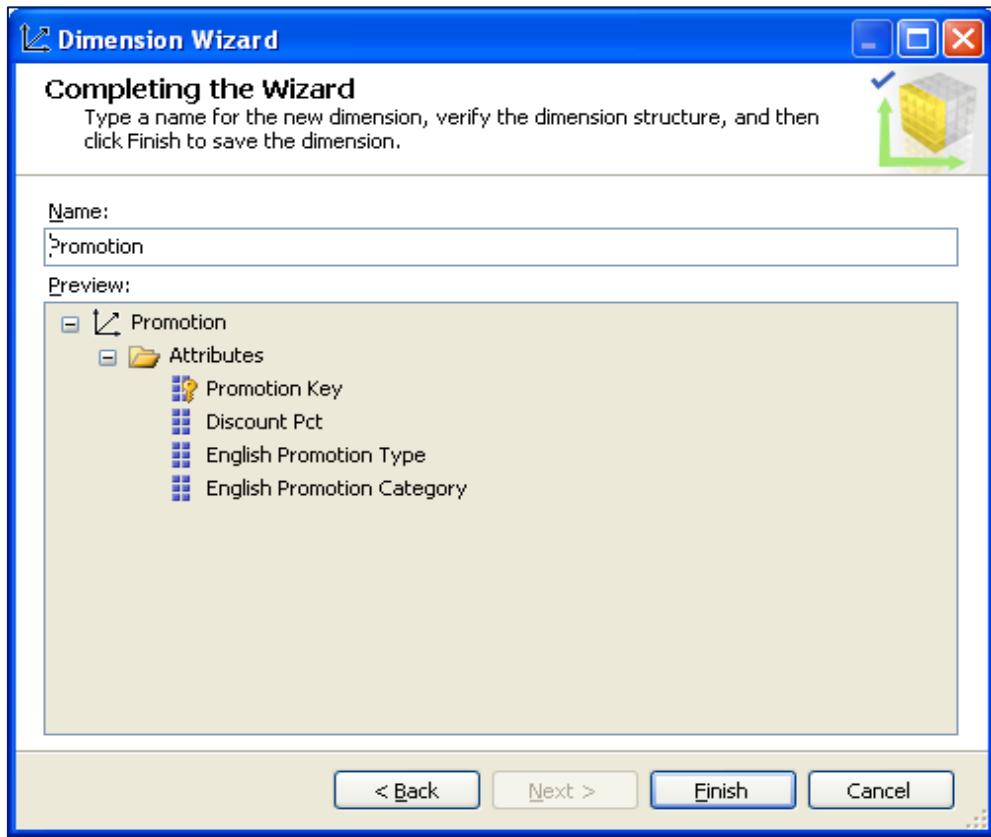
6. Pulse el botón Next. La siguiente ventana permite elegir el data source view a partir del cual se construirá la dimensión. Seleccione el data source view "AWDataMart_DSView". Adicionalmente esta ventana también permitirá definir las propiedades KeyColumns y NameColumns. En la propiedad Main Table, elija la tabla "DimPromotion". En la propiedad Key Columns, seleccione la columna PromotionKey. En la propiedad Column containing the member name, seleccione la columna EnglishPromotionName.



7. Pulse el botón Next. La siguiente ventana permite definir atributos adicionales para la dimensión. Seleccione las columnas "Discount Pct", "English Promotion Type" y "English Promotion Category".

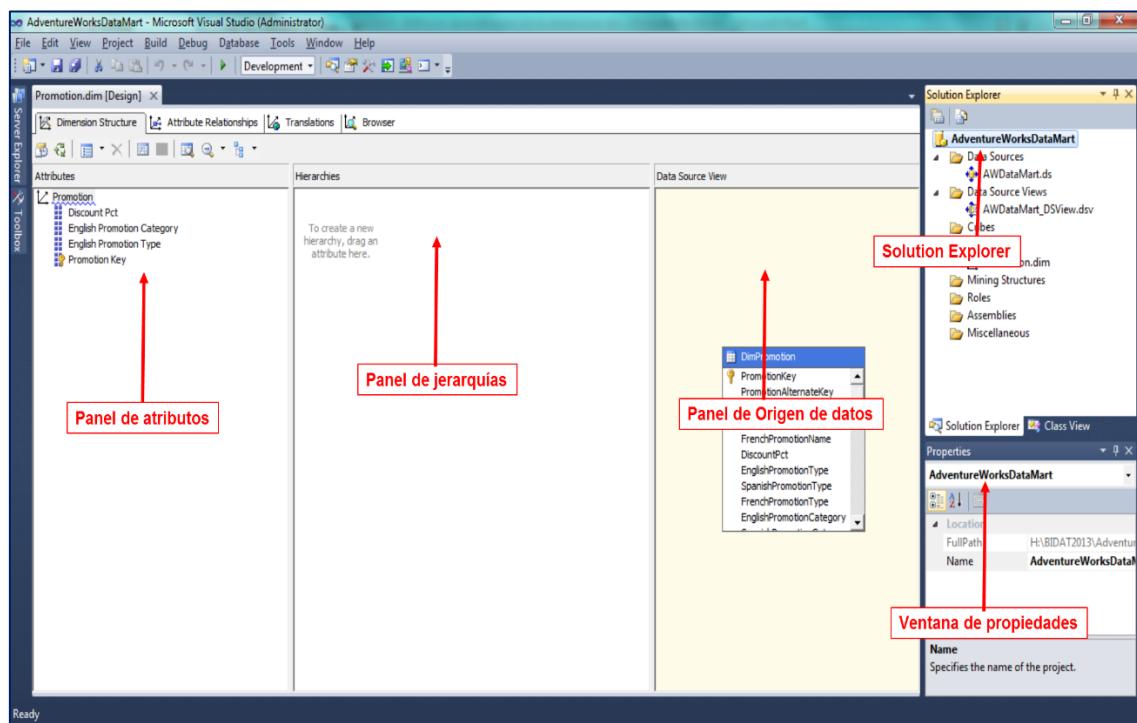


8. Se abrirá la ventana final del asistente. Establezca el nombre de la nueva dimensión a “Promotion”.

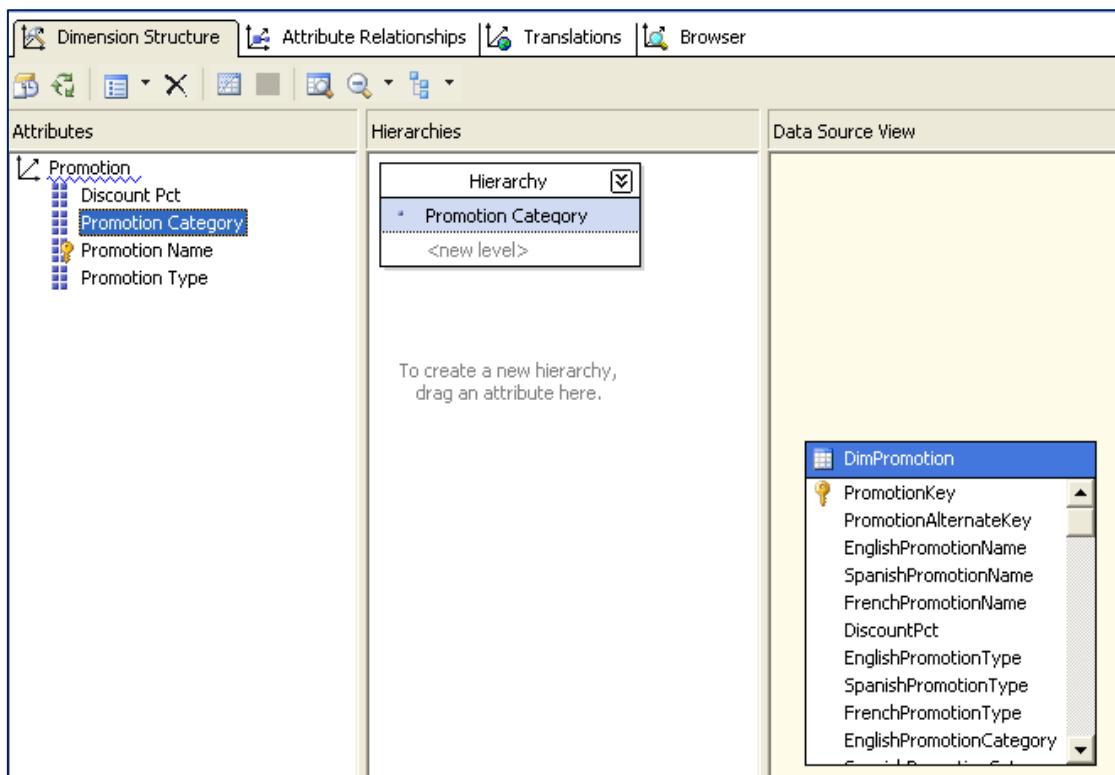


9. Pulse el botón Finish para finalizar la creación de la dimensión.

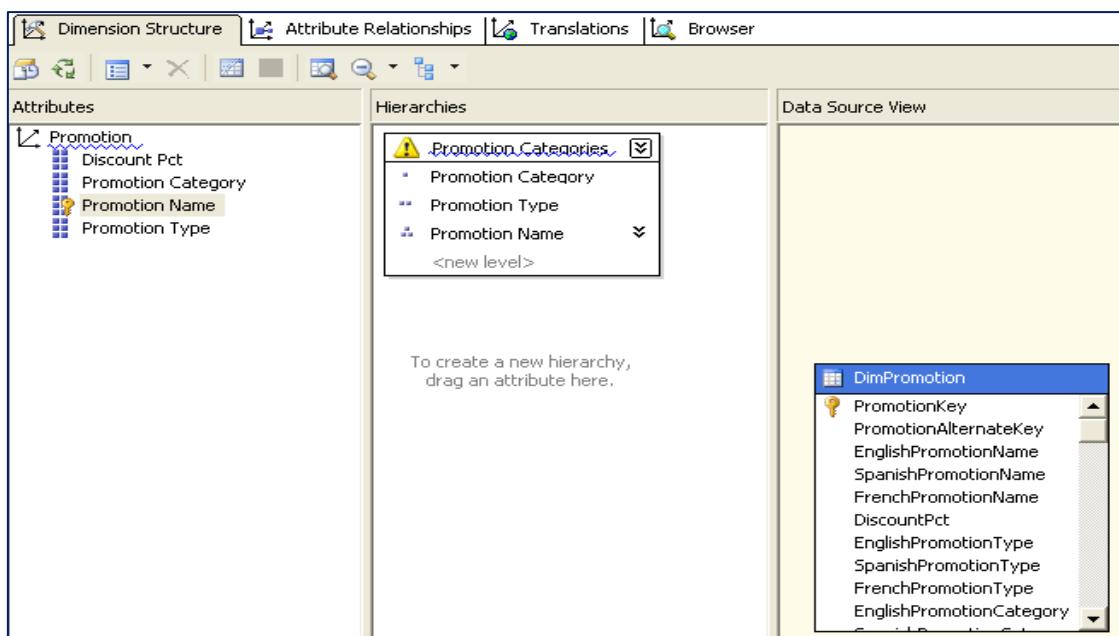
10. El entorno de desarrollo mostrará la estructura de la dimensión “Promotion”.



11. En el panel de atributos, seleccione el atributo “Promotion Key”. En la ventana de propiedades, observe su propiedad Usage. Está establecida a “Key”, lo cual significa que éste es el atributo clave para la dimensión. En la ventana de propiedades, cambie su propiedad Name a “Promotion Name”. Haga lo mismo para renombrar el atributo “English Promotion Category” a “Promotion Category”, y el atributo “English Promotion Type” a “Promotion Type”.
12. A continuación, se definirá una jerarquía en la dimensión “Promotion”. En el panel de atributos, seleccione el atributo “Promotion Category”, y arrástrelo sobre el panel de jerarquías. Inmediatamente, se creará una nueva jerarquía, cuyo nombre es “Hierarchy”. Esta jerarquía contiene un solo nivel, llamado “Promotion Category”.



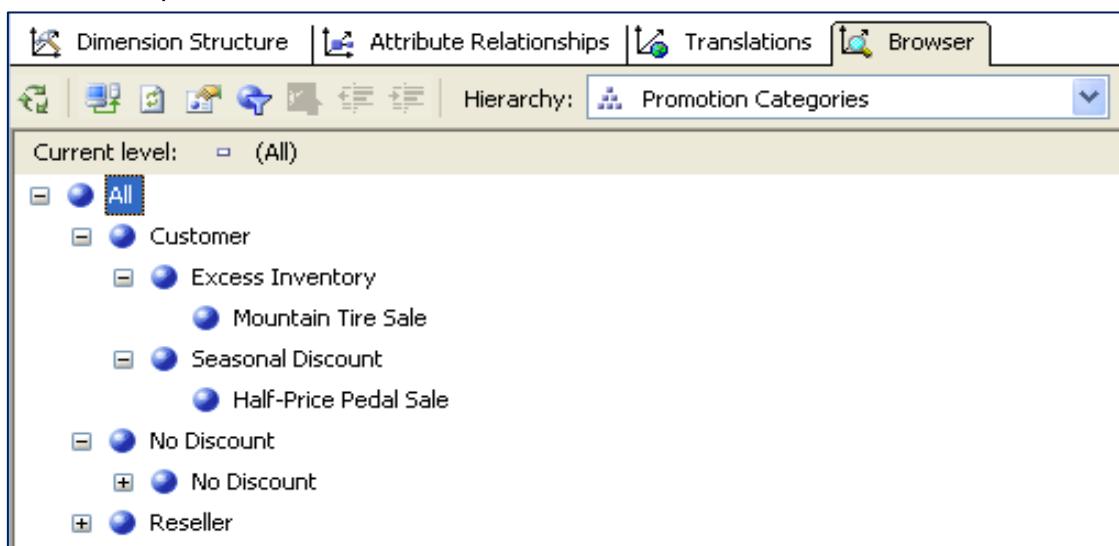
13. A continuación, seleccione el atributo “Promotion Type” en el panel de propiedades, y arrástrelo sobre la jerarquía “Hierarchy”, debajo del nivel “Promotion Category”. Después, seleccione el atributo “Promotion Name” en el panel de propiedades, y arrástrelo debajo del nivel “Promotion Type” en la jerarquía “Hierarchy”.
14. En el panel de jerarquías, seleccione la jerarquía “Hierarchy” haciendo un clic sobre la parte superior de la caja que representa la jerarquía. En la ventana de propiedades, cambie su propiedad Name a “Promotion Categories”. Al finalizar, el entorno de desarrollo debe mostrar la siguiente apariencia:



15. Repita los pasos anteriores para crear una nueva dimensión llamada “Currency” (Moneda) con las siguientes configuraciones:

Tabla de Dimensión	DimCurrency
Key Columns	Currency Key
Column containing the member name	Currency Alternate Key
Atributos adicionales	Ninguno
Jerarquía	Ninguna

16. En la dimensión “Currency”, renombre el atributo “Currency Key” como “Currency Name”.
 17. Guarde sus cambios y efectúe un deployment del proyecto
 18. En el Solution Explorer, haga un doble clic sobre “Promotion.dim”. Se abrirá el diseñador de dimensiones para la dimensión “Promotion”. Seleccione el tab Browser para visualizar su información.



19. Observe la estructura de árbol de la información. Esto representa la estructura de niveles de la jerarquía “Promotion Categories”. El miembro que se encuentra en la cúspide del árbol lleva la etiqueta “All” y se utiliza para totalizar la información de todos los miembros de la dimensión.

20. Usando el tab Browser, visualice la información de la dimensión “Currency”.

El miembro “All” y la propiedad Default Member

Las dimensiones poseen un miembro “All” (“Todos”). Este nivel se encuentra en la cúspide de todas las jerarquías y es ancestro de todos los miembros de la dimensión.

Por ejemplo, en la dimensión “Promotion” creada en la actividad anterior, puede observarse al miembro “All” en la cúspide de la jerarquía “Promotion Categories”.

La existencia del miembro All en una dimensión no es obligatoria. Cada atributo de la dimensión posee la propiedad IsAggregatable, cuyo valor por defecto es True. Si el valor de esta propiedad es False, el atributo no genera un miembro “All”.

También es posible modificar el nombre del miembro “All”, a través de la propiedad AttributeAllMemberName de la dimensión, y la propiedad AllMemberName de las jerarquías. De esta forma, se puede configurar a la dimensión “Promotion” para mostrar un nombre personalizado en el miembro “All”, como, por ejemplo, “All Promotions”.

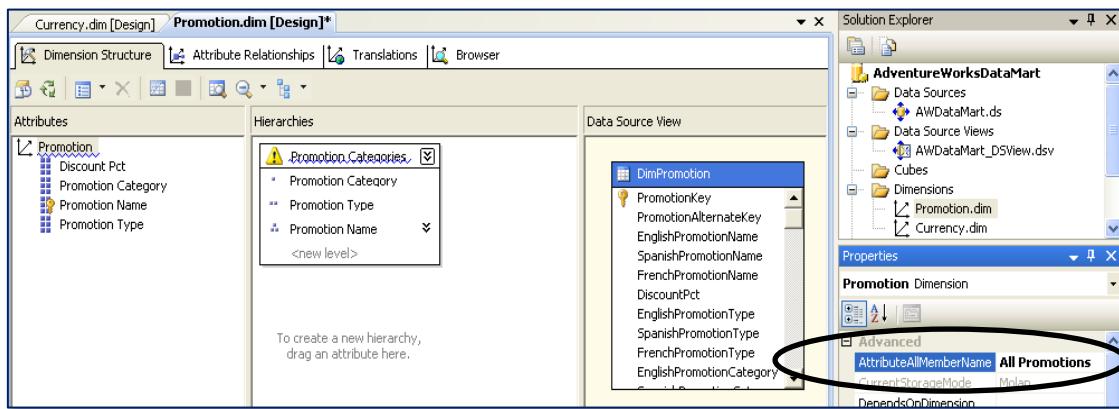
Por otro lado, toda dimensión tiene un miembro predeterminado. Si una consulta de datos no especifica en forma explícita cuál es el miembro utilizado para obtener la información asociada con una dimensión, el servidor de análisis utilizará el miembro predeterminado para efectuar la consulta. Por defecto, el miembro predeterminado es el miembro “All”. Sin embargo, puede cambiarse a través de la propiedad DefaultMember presente en cada atributo de la dimensión.

Por ejemplo, en la dimensión “Currency” creada en la actividad anterior, el miembro predeterminado es “All”. Esta dimensión contiene información de monedas. Sin embargo, esta configuración no es recomendable, pues si esta dimensión estuviese asociada a un cubo de ventas, las consultas de datos devolverían las ventas efectuando la suma de todas las monedas, lo cual no tiene sentido. Por tanto, el desarrollador debe establecer una moneda como miembro por defecto; por ejemplo, “USD” (dólares americanos).

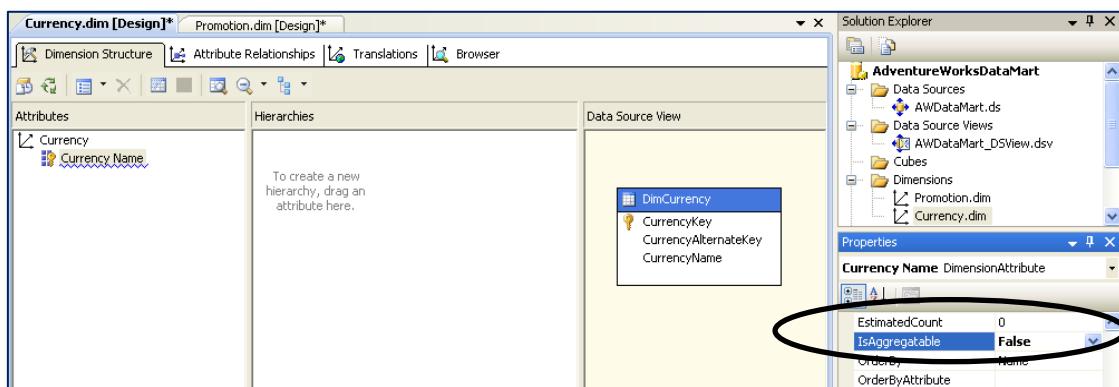
Ejercicio

Cambiar el nombre del miembro ALL y establecer el miembro predeterminado

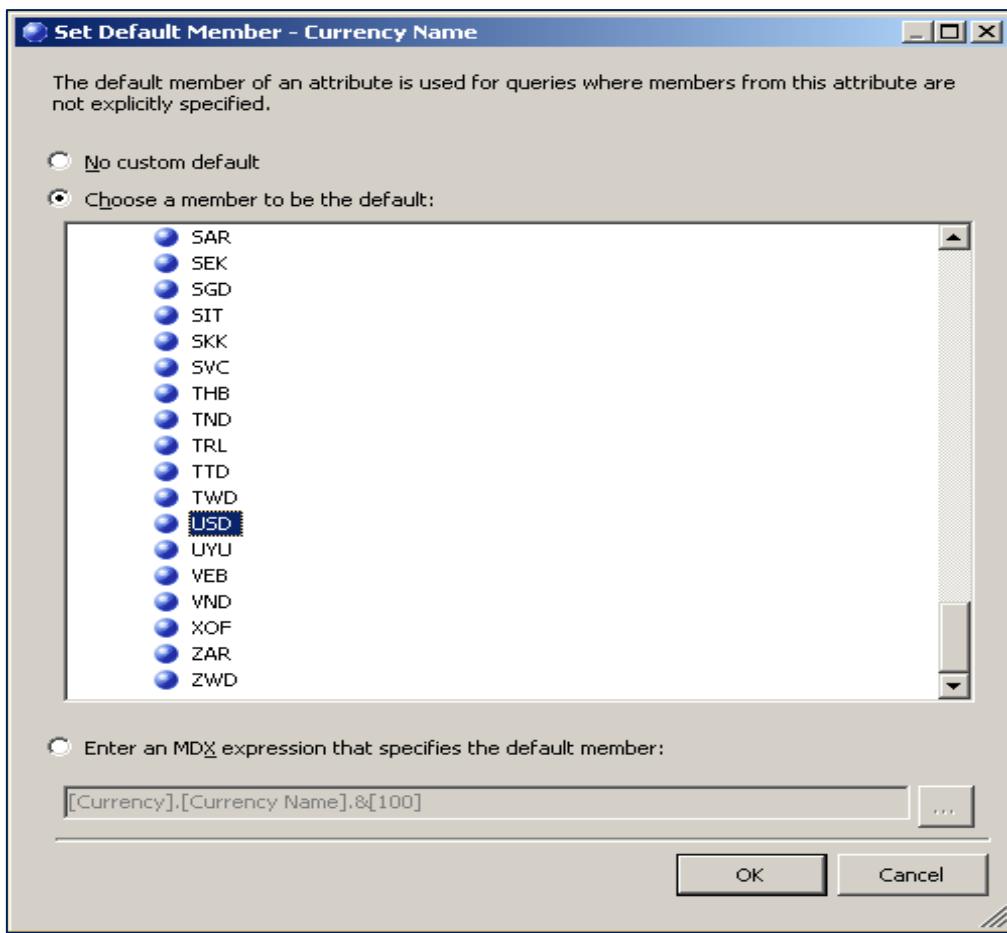
1. En el Solution Explorer, haga doble clic sobre “Promotion.dim”. A continuación, en el panel de diseño, seleccione el tab Dimension Structure. En el panel de atributos, seleccione la dimensión Promotion. En la ventana de propiedades, establezca la propiedad AttributeAllMemberName a “All Promotions”.



2. En el panel de jerarquías, seleccione la jerarquía “Promotion Categories”. En el Solution Explorer, establezca el valor de la propiedad AllMemberName a “All Promotions”.
3. En el Solution Explorer, haga doble clic sobre “Currency.dim”. A continuación, en el panel de diseño, seleccione el tab Dimension Structure. En el panel de atributos, seleccione la dimensión Currency. En la ventana de propiedades, establezca la propiedad AttributeAllMemberName a “All Currencies”.
4. Por otro lado, los miembros de la dimensión “Currency” no deben sumarizarse, pues no tiene sentido sumar ventas en distintas monedas. Esto se controla a través de la propiedad IsAggregatable. En el panel de atributos, seleccione el atributo “Currency Name”, y en la ventana de propiedades, establezca la propiedad IsAggregatable a False.



5. A continuación, se cambiará el miembro predeterminado del atributo “Currency Name” para que la moneda por defecto sea el dólar americano. Seleccione el atributo “Currency Name” en el panel de atributos y ubique la propiedad DefaultMember en la ventana de propiedades. Pulse el botón.
6. Se abrirá una ventana que permite elegir el miembro predeterminado. Marque la opción Choose a member to be the default, y en la lista de miembros ubique y seleccione el miembro “USD”.

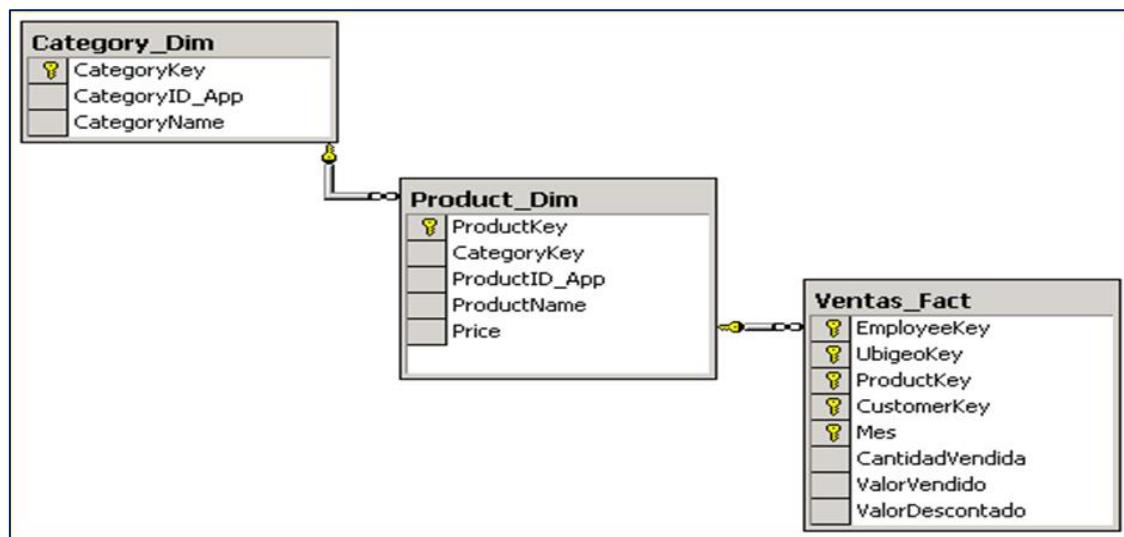


7. Pulse el botón OK para aceptar los cambios.
8. Guarde los cambios y efectúe un deployment del proyecto.
9. Utilice el tab Browser para ver los datos de la dimensión “Promotion”. Dado que la estructura de las dimensiones ha cambiado debido al deployment, debe pulsar el botón ubicado en la parte superior, para reconectarse con el servidor de análisis. Observe la data.

10. Repita el paso (9) con la dimensión “Currency” para observar su data. Puede observarse que no existe un miembro “All” en esta dimensión, debido a la configuración efectuada sobre la propiedad IsAggregatable.

2.1.3. Definiendo Jerarquías

Dimensiones Snowflake

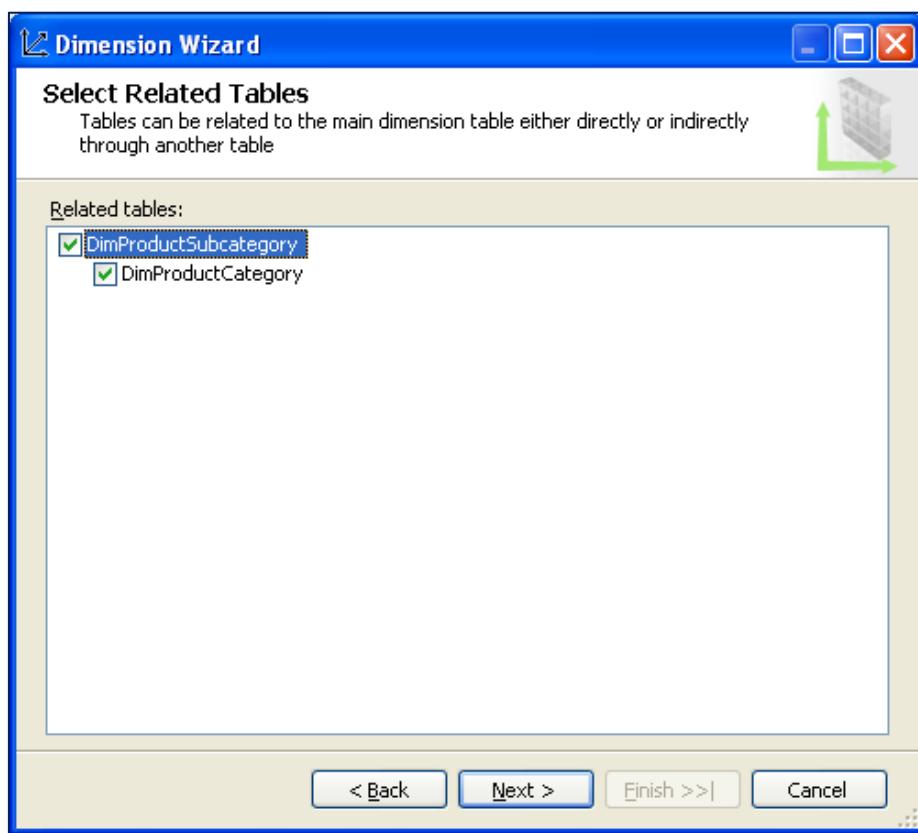


Las dimensiones SnowFlake son generadas por más de una tabla de dimensión. En la figura superior se muestran las tablas de dimensión Category_Dim y Product_Dim. Éstas están enlazadas entre sí a través de una relación. Sólo la tabla Product_Dim se enlaza con la tabla de hechos Ventas_Fact.

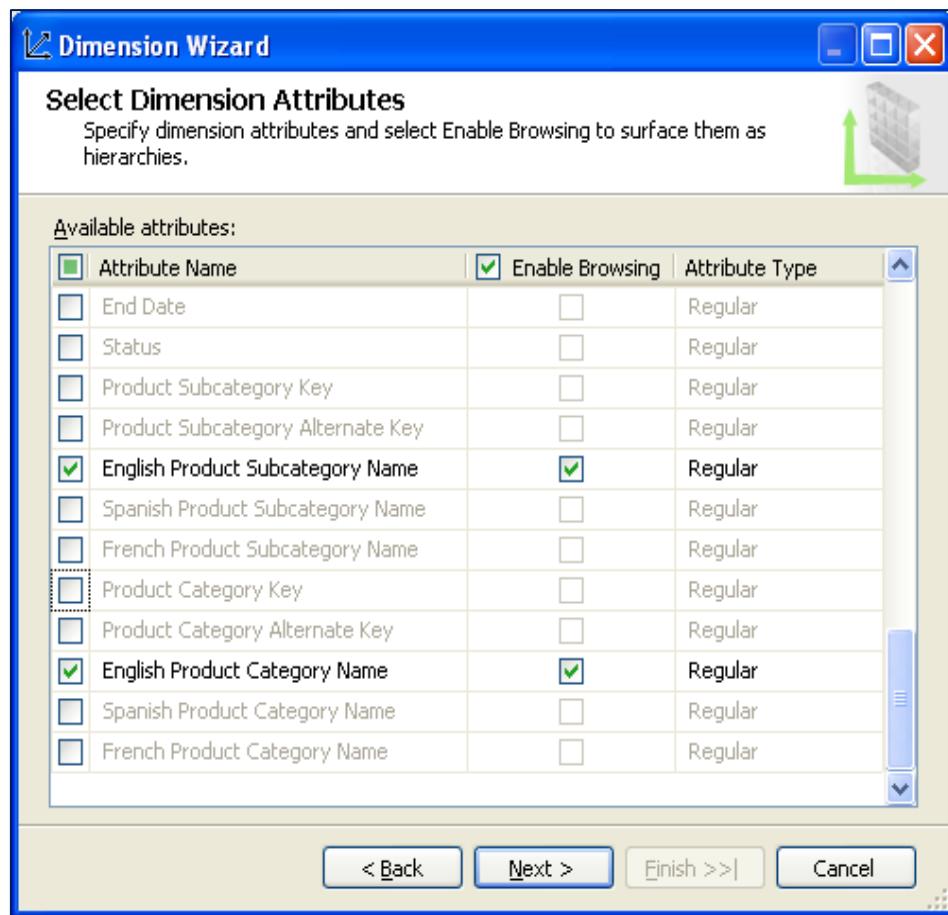
Son dimensiones con estructuras normalizadas. Permiten reducir el tamaño de las tablas dimensionales, pero generan mayor complejidad al modelo dimensional.

El siguiente ejercicio muestra la creación de dimensiones SnowFlake a través del diseñador de dimensiones.

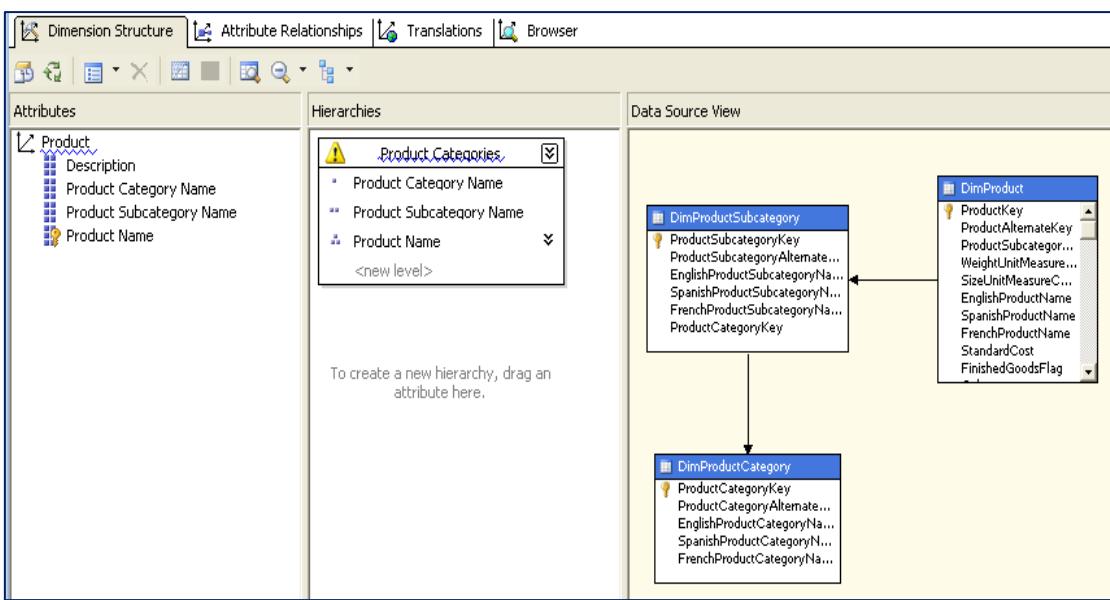
1. Inicie la creación de una nueva dimensión. En la ventana de selección del método de creación de la dimensión, asegúrese de usar la opción Use an existing table.
2. Para la creación de esta dimensión, seleccione la tabla “DimProduct”. En la propiedad Key Columns, seleccione la columna “ProductKey”. En la propiedad Name column, seleccione la columna “EnglishProductName”.
3. El asistente detectará automáticamente las relaciones existentes entre las tablas “DimProduct”, “DimProductSubCategory” y “DimProductCategory”. La siguiente ventana muestra las tablas relacionadas con “DimProduct”:



4. Continúe con el asistente. En la ventana de selección de atributos, mantenga el atributo “Product Key”, desmarque el atributo “Product Subcategory Key” y “Product Category Key”. Y a continuación seleccione los atributos “English Product Category Name”, “English Product Subcategory Name” y “English Description”:



5. Al finalizar, establezca el nombre de la nueva dimensión a “Product”.
6. En el diseñador de dimensiones, cambie respectivamente la propiedad Name de los atributos “Product Key”, “English Description”, “English Product Category Name” y “English Product Subcategory Name” a los valores “Product Name”, “Description”, “Product Category Name” y “Product Subcategory Name”.
7. A través del diseñador de dimensiones, agregue una nueva jerarquía llamada “Product Categories” con los siguientes niveles:
 - Product Category Name
 - Product Subcategory Name
 - Product Name
8. Al finalizar estos pasos, el panel de diseño de la dimensión “Product” debe mostrar la apariencia mostrada en la siguiente figura. Obsérvese que, en el panel de origen de datos, se encuentran las tablas “DimProduct”, “DimProductSubcategory” y “DimProductCategory”. Cada atributo de la dimensión “Product” pertenece a alguna de estas tablas. Por ejemplo, el atributo “Product Category Name” corresponde con la columna “EnglishProductName” de la tabla “DimProductCategory”.



9. Efectúe las configuraciones correspondientes para establecer el nombre del miembro “All” a “All Products”, en la dimensión “Product” y en la jerarquía Product Categories”.
10. Guarde sus cambios y efectúe un deployment del proyecto. Observe la información de la dimensión.

Named Calculations (Cálculos con nombre)

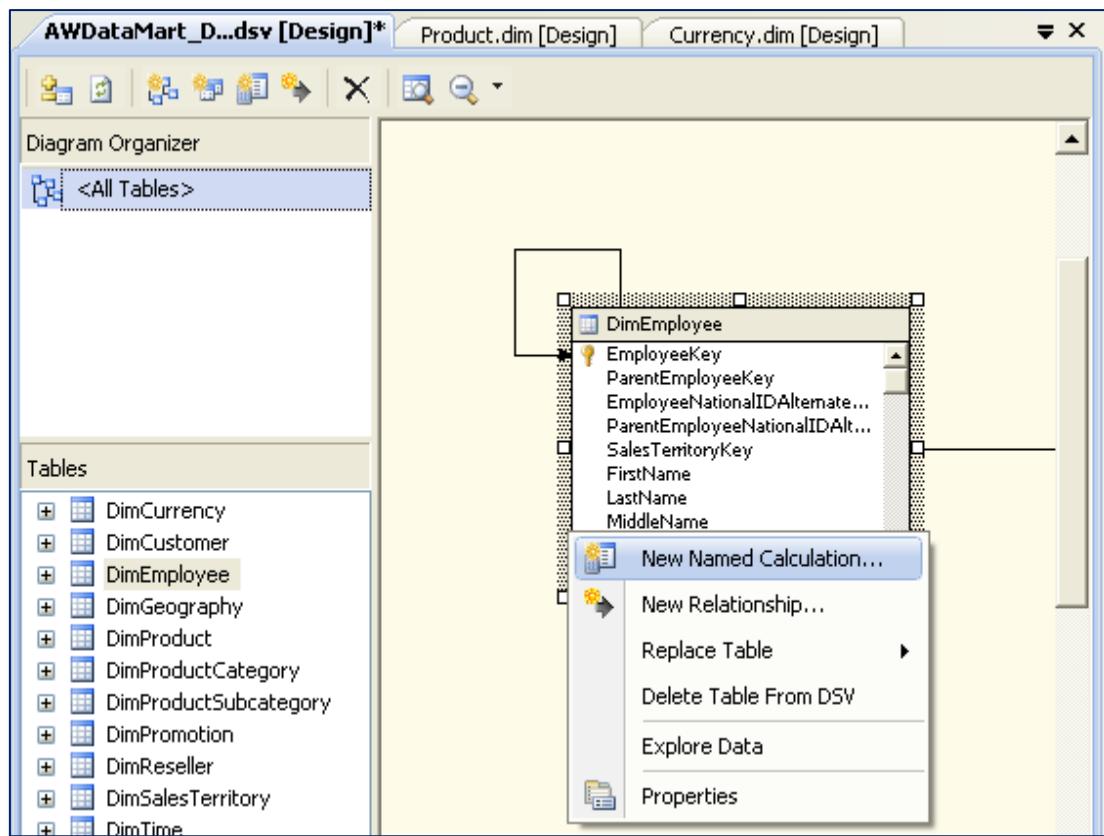
Los named calculations permiten extender el modelo de datos sobre el cual se construyen los cubos y las dimensiones, sin necesidad de modificar la estructura de las tablas en el origen de datos. Los named calculations se crean en los data source views a través de expresiones SQL.

Por ejemplo, la tabla “DimEmployee” contiene las columnas “FirstName”, “MiddleName” y “LastName”. Sin embargo, al especificar la estructura de la dimensión “Employee”, es conveniente contar con un atributo que almacene el nombre completo del empleado. Para conseguirlo, es preciso crear un named calculation en la tabla “DimEmployee” que efectúe una concatenación de las columnas antes mencionadas.

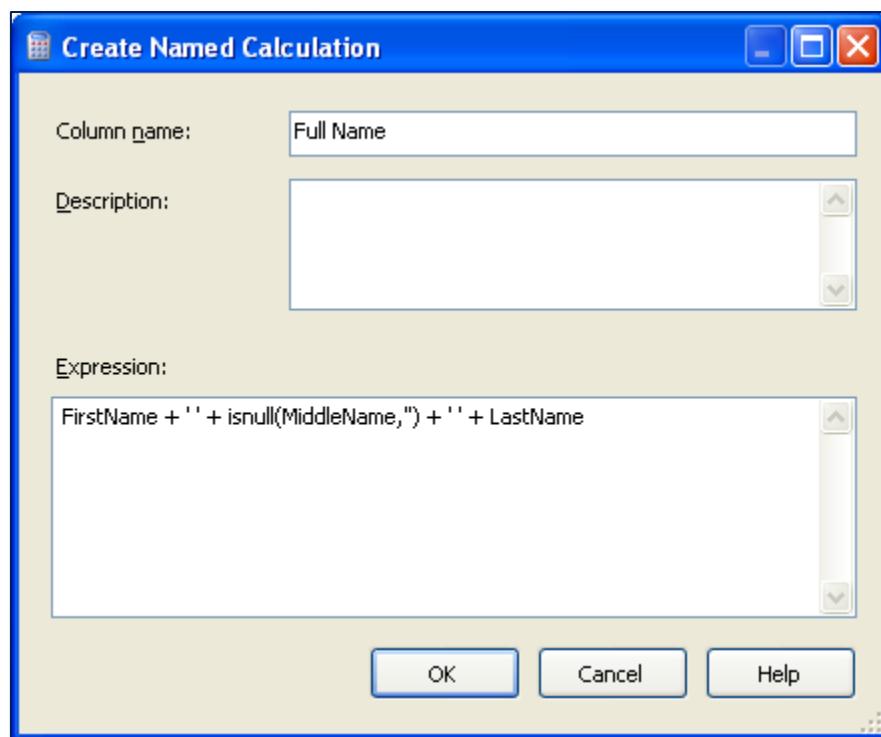
Ejercicio:

Crear la Dimensión Named Calculation.

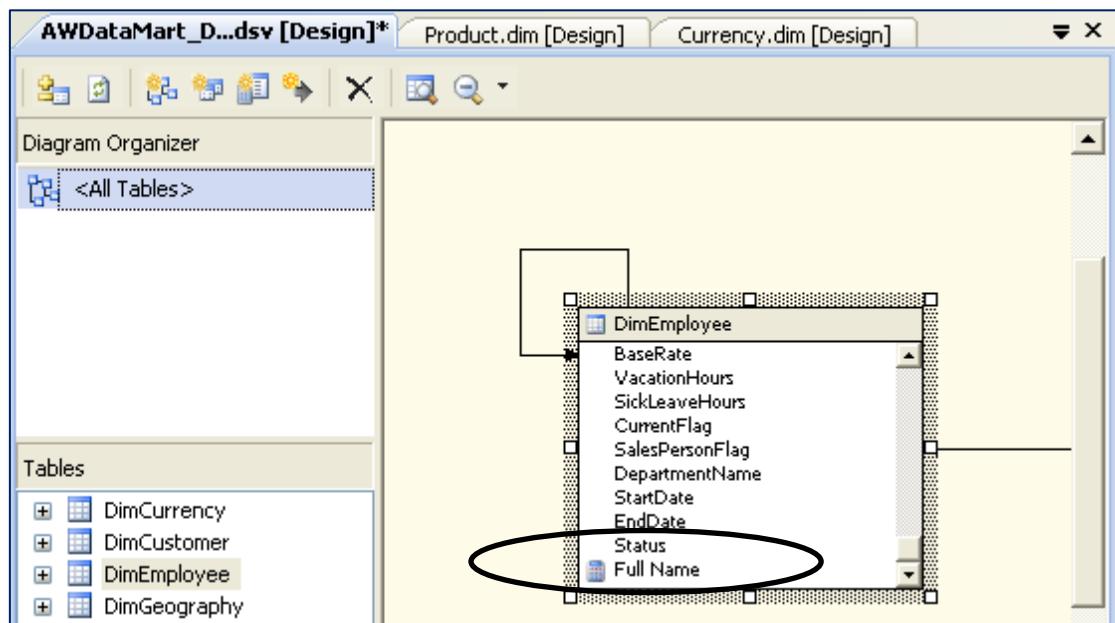
1. En el Solution Explorer, ubique el data source view “AWDataMart_DSView”. Haga doble clic sobre él. Se abrirá el diseñador de data source views.
2. En el diseñador de data source views, ubique la tabla “DimEmployee” (para ubicarla con facilidad, haga un clic derecho sobre cualquier parte del diseñador de data source views, y seleccione la opción Find Table; a continuación, ubique la tabla y pulse el botón OK).
3. En el diseñador de data source views, haga un clic derecho sobre la tabla “DimEmployee”, y seleccione la opción New Named Calculation.



4. En la ventana de creación de named calculations, escriba una expresión SQL para crear un nuevo named calculation llamado “FullName” estableciendo los siguientes valores:



5. Pulse el botón OK. Observe que el named calculation se ha adicionado a la tabla “DimEmployee” como una nueva columna.



6. En la tabla “DimEmployee”, repita los pasos anteriores para crear un nuevo named calculation llamado “MaritalStatusDescription” basado en la siguiente expresión SQL:

```
CASE
    WHEN MaritalStatus = 'M' THEN 'Married'
    WHEN MaritalStatus = 'S' THEN 'Single'
    ELSE 'Not defined'
END
```

7. En la tabla “DimEmployee”, cree un nuevo named calculation llamado “GenderDescription” basado en la siguiente expresión SQL:

```
CASE
    WHEN Gender = 'M' THEN 'Male'
    WHEN Gender = 'F' THEN 'Female'
END
```

8. En la tabla “DimCustomer”, cree un nuevo named calculation llamado “FullName” basado en la siguiente expresión SQL:

`FirstName + ' ' + isnull(MiddleName, '') + ' ' + LastName + isnull(Suffix, '') + ' - ' + CustomerAlternateKey`

9. En la tabla “DimCustomer”, cree un nuevo named calculation llamado “MaritalStatusDescription” basado en la siguiente expresión SQL:

```
CASE
    WHEN MaritalStatus = 'M' THEN 'Married'
    WHEN MaritalStatus = 'S' THEN 'Single'
    ELSE 'Not defined'
END
```

10. En la tabla “DimCustomer”, cree un nuevo named calculation llamado “GenderDescription” basado en la siguiente expresión SQL:

```
CASE
    WHEN Gender = 'M' THEN 'Male'
    WHEN Gender = 'F' THEN 'Female'
END
```

11. En la tabla “DimDate”, cree un nuevo named calculation llamado “Semester” basado en la siguiente expresión SQL:

```
'Semester ' + CAST(CalendarSemester AS CHAR(1)) + ' - ' +
CAST(CalendarYear AS CHAR(4))
```

12. En la tabla “DimDate”, cree un nuevo named calculation llamado “Trimester” basado en la siguiente expresión SQL:

```
'Trimester ' + CAST(CalendarQuarter AS CHAR(1)) + ' - ' +
CAST(CalendarYear
AS CHAR(4))
```

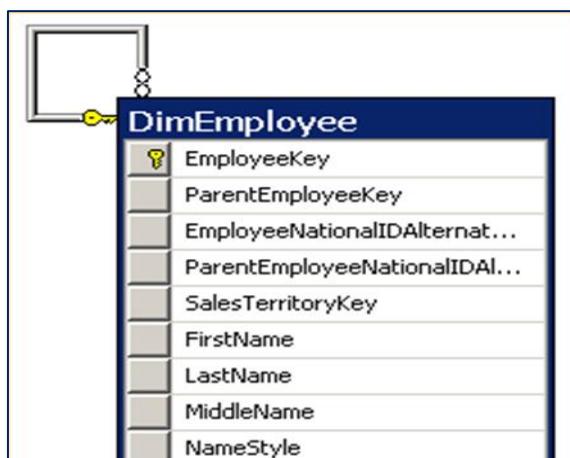
13. En la tabla “DimDate”, cree un nuevo named calculation llamado “Month” basado en la siguiente expresión SQL:

```
EnglishMonthName + ' ' + CAST(CalendarYear AS CHAR(4))
```

14. Grabe los cambios.

Dimensiones Padre – Hijo

Las dimensiones padre – hijo constituyen un caso especial en el que los niveles se basan en relaciones recursivas de tipo foreign – key existentes en una misma tabla de dimensión.



La figura superior muestra la tabla DimEmployee. Esta tabla posee una relación recursiva, en la cual participan las columnas EmployeeKey y ParentEmployeeKey.

Cada empleado (identificado a través de la columna EmployeeKey) posee un jefe (identificado a través de la columna ParentEmployeeKey). Tanto los empleados como los jefes se encuentran almacenados en la tabla DimEmployee.

La figura de la derecha muestra a los miembros de la dimensión Employee basada en la tabla DimEmployee. Ken J Sánchez es el jefe de todo el personal. A su vez, Brian S Welcker es el jefe de Amy E Alberts, Jae B Pak, Rachel B Valdez y Ranjit R Varkey Chudukatil.



Miembros con data

Una jerarquía se compone de varios niveles. Los miembros del último nivel de una jerarquía se denominan leaf members (miembros de hoja).

En una dimensión estándar, sólo los miembros del último nivel de una jerarquía, es decir, los leaf members, tienen datos en la tabla de hechos. Sean, por ejemplo, una tabla de hechos de ventas y la jerarquía Ubigeo que contiene los niveles País y Ciudad. En la tabla de hechos, los registros estarán asociados únicamente con los miembros del último nivel; es decir, la tabla de hechos contendrá ventas para las ciudades. De esta manera, se encontrarán registros de ventas para Lima, Trujillo, Bogotá, etc. El cubo obtendrá las ventas para cada país con base en la suma de las ventas para las ciudades.

A diferencia de las dimensiones estándar, en las dimensiones Padre – Hijo los miembros de niveles superiores también pueden poseer datos en la tabla de hechos. Por ejemplo, si se tienen en cuenta los miembros mostrados en la figura anterior, puede suceder que tanto Jae B Palk y Rachel B Valdez (leaf members de la jerarquía) estén asociados con registros en la tabla de hechos, y también Brian W Welcker y David M Bradley (miembros de niveles superiores o non – leaf members).

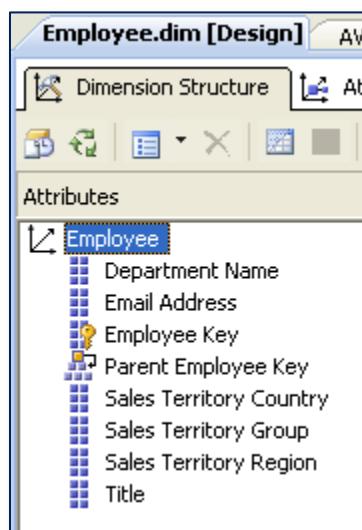
Las jerarquías de tipo Padre – Hijo tienen una propiedad llamada Members with data, que determina el tratamiento que se dará a la información de la tabla de hechos que corresponde con miembros de niveles superiores. Esta propiedad puede tomar uno de los siguientes valores:

- Non – leaf data visible: La información de la tabla de hechos asociada con miembros de niveles superiores (miembros non – leaf) se muestra en las consultas.
- Non – leaf data hidden: La información de la tabla de hechos asociada con miembros de niveles superiores (miembros non – leaf) queda oculta.

Ejercicio:

Crear una Dimensión Padre - Hijo

1. Inicie la creación de una nueva dimensión. En la ventana de selección del método de creación de la dimensión, asegúrese de que la opción Use an existing table esté marcada.
2. Para la creación de esta dimensión, seleccione la tabla “DimEmployee”. En la propiedad Key Columns, seleccione la columna “EmployeeKey”. En la propiedad Name column, seleccione la columna “FullName” (esta columna fue creada en el ejercicio anterior, a través de un named calculation).
3. El asistente detectará automáticamente la relación existente entre las tablas “DimEmployee” y “DimSalesTerritory”. Continúe con el asistente.
4. En la ventana de selección de atributos, mantenga seleccionados los atributos “Employee Key” y “Parent Employee Key”, agregue los atributos “Title”, “Email Address”, “Department Name”, “Sales Territory Region”, “Sales Territory Country” y “Sales Territory Group”. Desmarque “Sales Territory Key”.
5. Al finalizar, establezca el nombre de la nueva dimensión a “Employee”. Luego, observe en el panel de atributos la estructura de la dimensión:



Obsérvese la existencia de la jerarquía “Parent Employee Key”. La relación Padre – Hijo se efectuará a través de esta jerarquía.

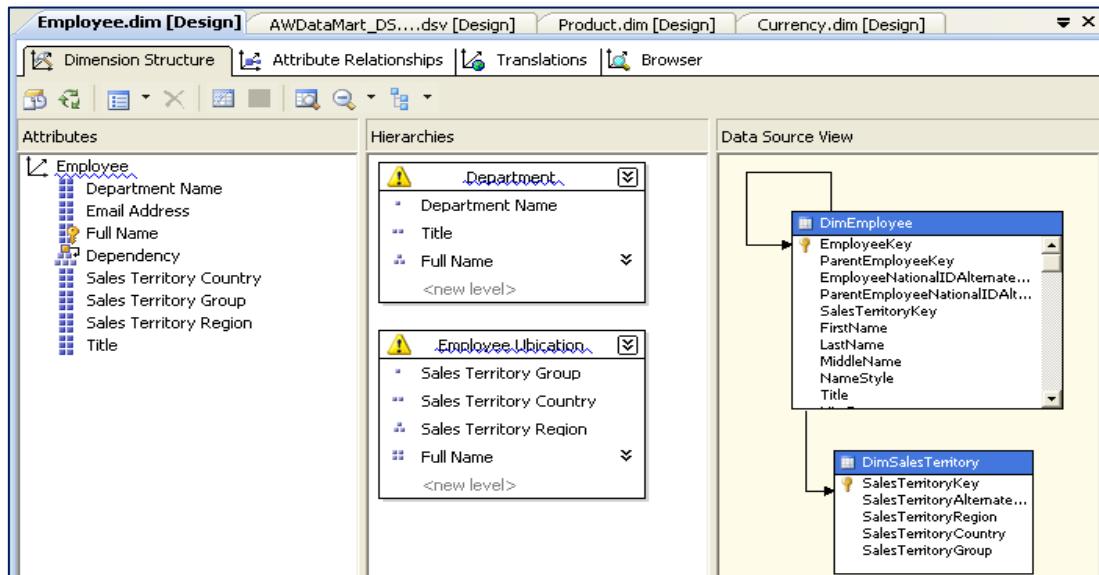
6. En la ventana de propiedades, cambie el nombre del atributo “Employee Key” a “Full Name”. Cambie el nombre de la jerarquía “Parent Employee Key” a “Dependency”.
7. A continuación, cree una nueva jerarquía llamada “Department” en la dimensión “Employee” con la siguiente estructura de niveles:

- Department Name
- Title
- Full Name

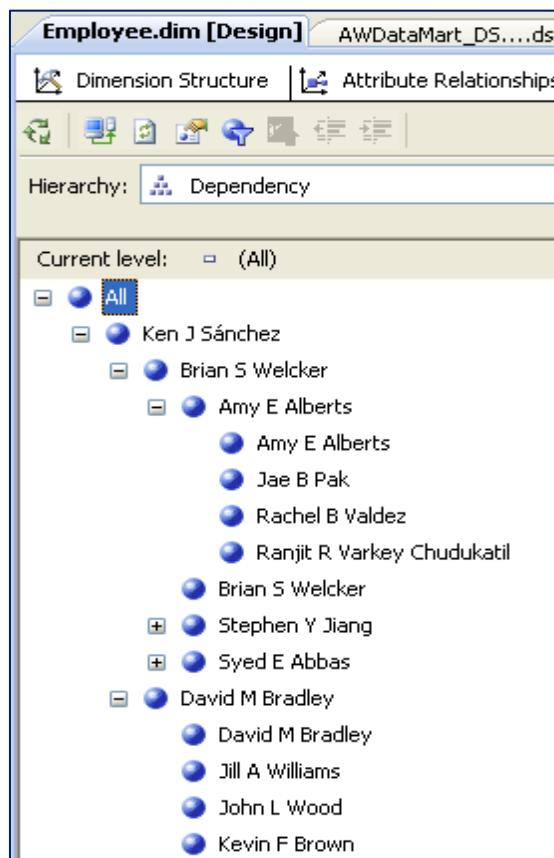
8. A continuación, cree una nueva jerarquía llamada “Employee Ubication” en la dimensión “Employee” con la siguiente estructura de niveles:

- Sales Territory Group
- Sales Territory Country
- Sales Territory Region
- Full Name

9. Al finalizar estos pasos, el panel de diseño de la dimensión "Employee" debe mostrar la apariencia mostrada en la siguiente figura. Además de la jerarquía Padre – Hijo, la dimensión "Employee" cuenta con dos jerarquías adicionales; es decir, existen múltiples maneras de navegar a través de la información de empleados.



10. Guarde los cambios y efectúe un deployment del proyecto. Observe los datos de la jerarquía "Dependency" de la dimensión "Employee":



Relaciones entre atributos (attribute relationships)

Los atributos de una dimensión pueden relacionarse entre sí. Dichas relaciones se definen a través de los attribute relationships. Cuando una dimensión es utilizada por un cubo, los attribute relationships sirven para conectar cada atributo con la tabla de hechos.

Por defecto, el atributo clave de una dimensión (el atributo cuya propiedad Usage está establecida a “Key”) contiene relaciones hacia todos los demás atributos de la dimensión. De hecho, todo atributo debe estar relacionado con el atributo clave, directa o indirectamente.

Los attribute relationships se definen para resolver tres tipos de situaciones:

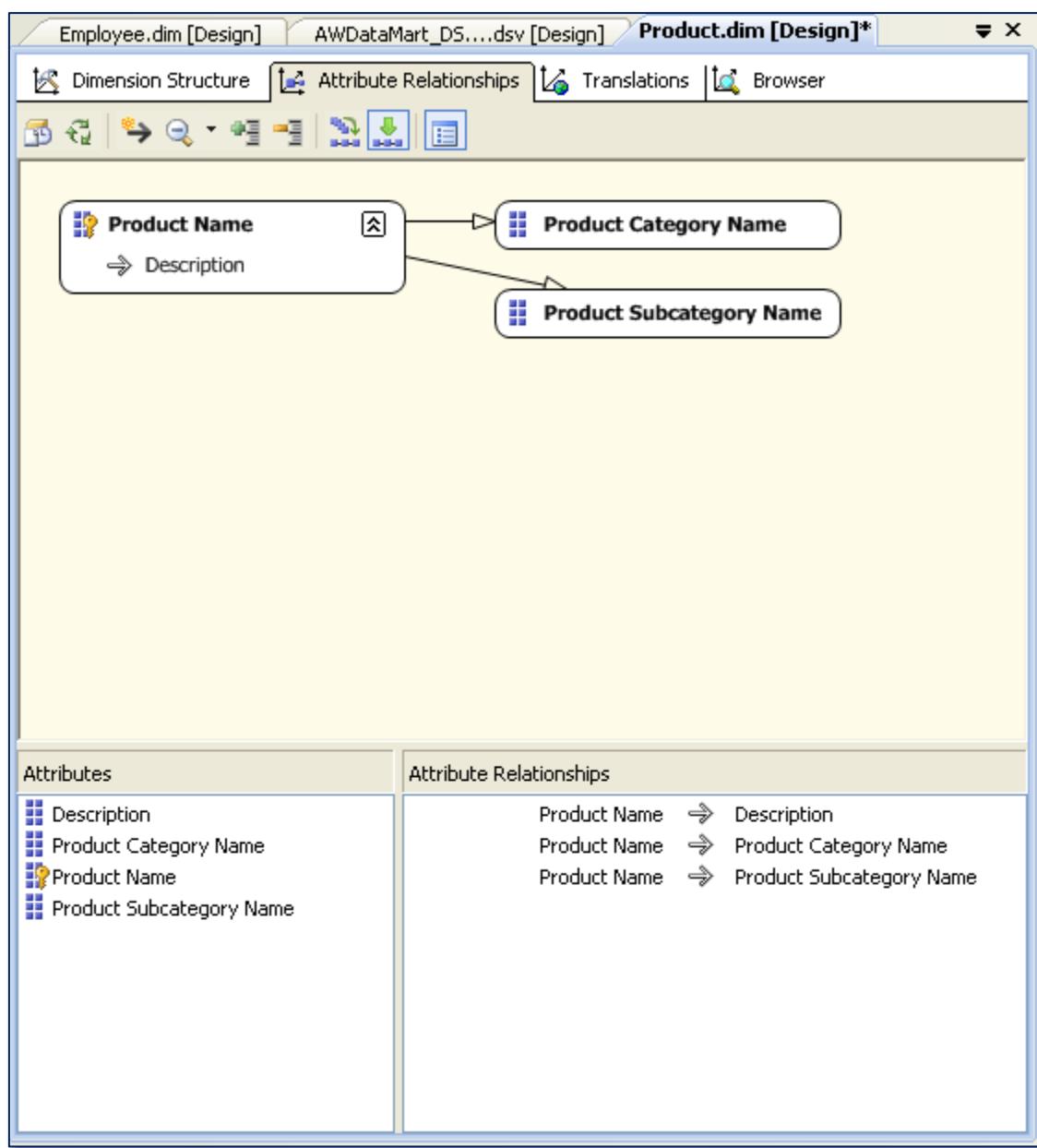
- Optimizar el funcionamiento de las jerarquías. En una jerarquía, se tiene una estructura en la cual cada nivel está contenido en el nivel inmediato superior. Por ejemplo, en la jerarquía “Product Categories” creada en la actividad (3), el nivel “Product Name” está contenido en el nivel “Product Subcategory Name”, y éste a su vez está contenido en el nivel “Product Category Name”. Al definir attribute relationships para vincular los atributos que dan origen a estos niveles, se optimiza el rendimiento de los cubos que utilicen esta dimensión, debido a la creación de estructuras de agregación e indexación, a través de las cuales los valores de las medidas asociados con los miembros superiores se obtienen por la sumarización de los valores de los miembros inferiores.
- Para determinar el ordenamiento de los miembros de un atributo. Por ejemplo, en una dimensión de fecha, es deseable que los miembros del atributo “Mes” no estén ordenados de forma alfabética, sino con base en el número correlativo de mes.
- Asociar propiedades a los miembros (member properties). Por ejemplo, se puede asociar el atributo “FullName” de la dimensión “Employee” con otro atributo llamado “Address” que contiene la dirección del domicilio del empleado. El valor del atributo “Address” podrá consultarse para cada valor del atributo “FullName” a través de sentencias MDX.

El criterio fundamental para definir attribute relationships es que la relación entre los atributos sea uno – muchos. Por ejemplo, para relacionar los atributos “Ciudad” y “País”, es necesario que cada ciudad esté contenida en un solo país. Esta relación puede representarse de la siguiente forma:

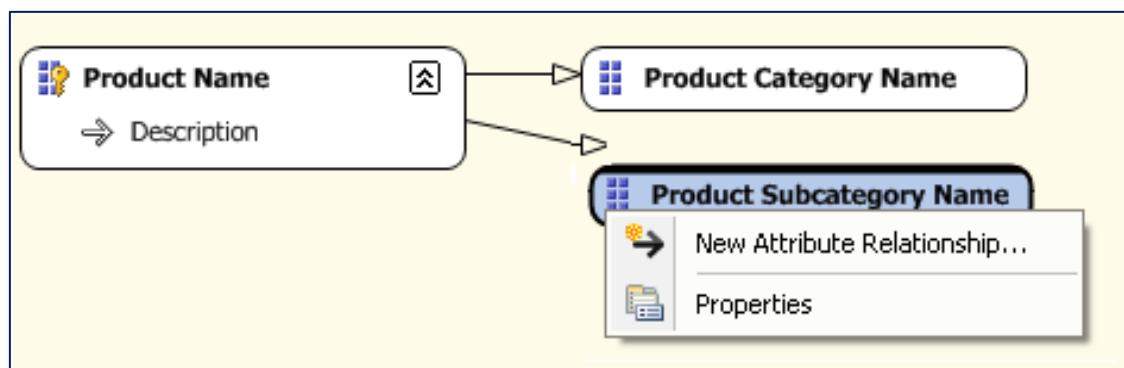


Ejercicio:

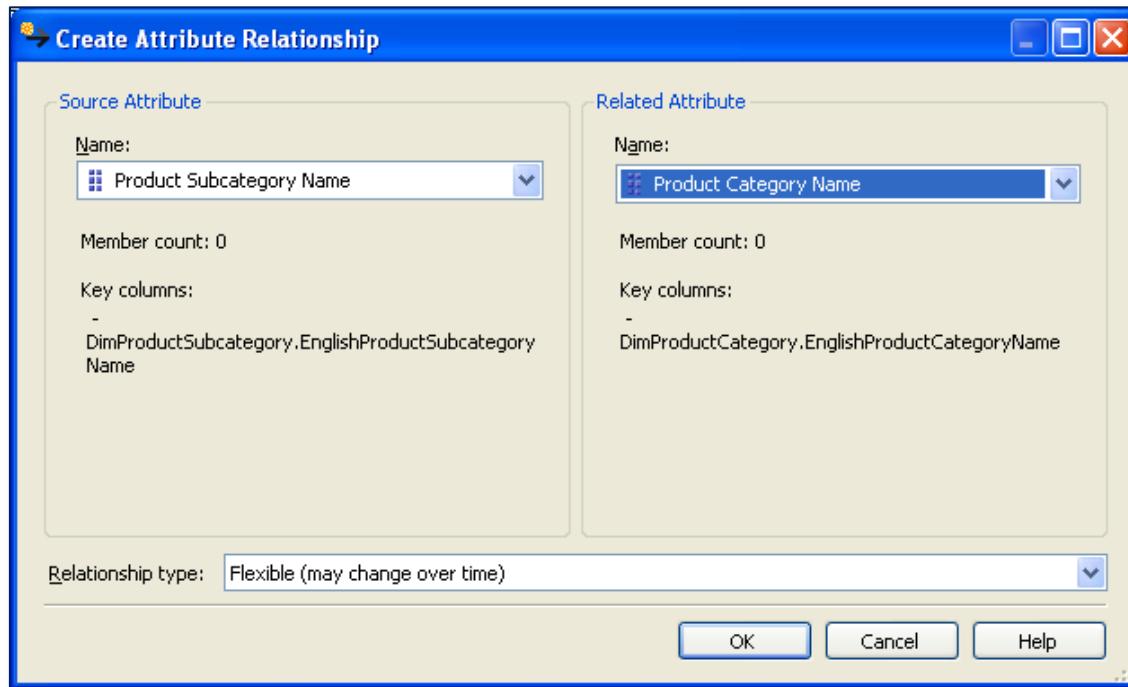
1. En el Solution Explorer, haga doble clic sobre “Product.dim”.
2. En la pestaña Attribute Relationships, expanda el atributo “Product Name”. Este es el atributo clave de la dimensión. Observe las relaciones que posee con los demás atributos.



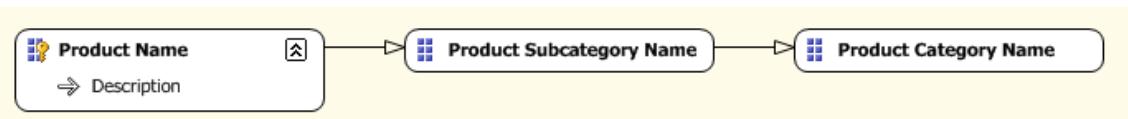
- Para optimizar el funcionamiento de las dimensiones, se definirá un attribute relationship entre la categoría de producto y la subcategoría de producto. Para esto, seleccione el atributo “Product Subcategory Name”, click derecho y elija “New Attribute Relationship”.



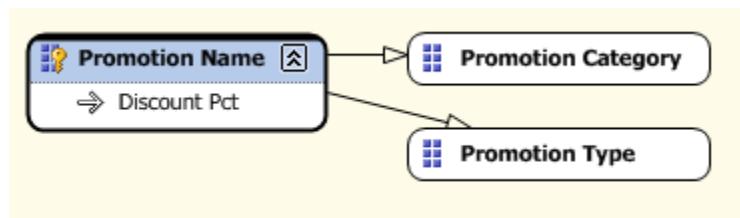
4. A continuación, cree la relación de atributo teniendo como atributo origen “Product Subcategory Name” y como atributo relacionado el atributo “Product Category Name”, como se muestra a continuación:



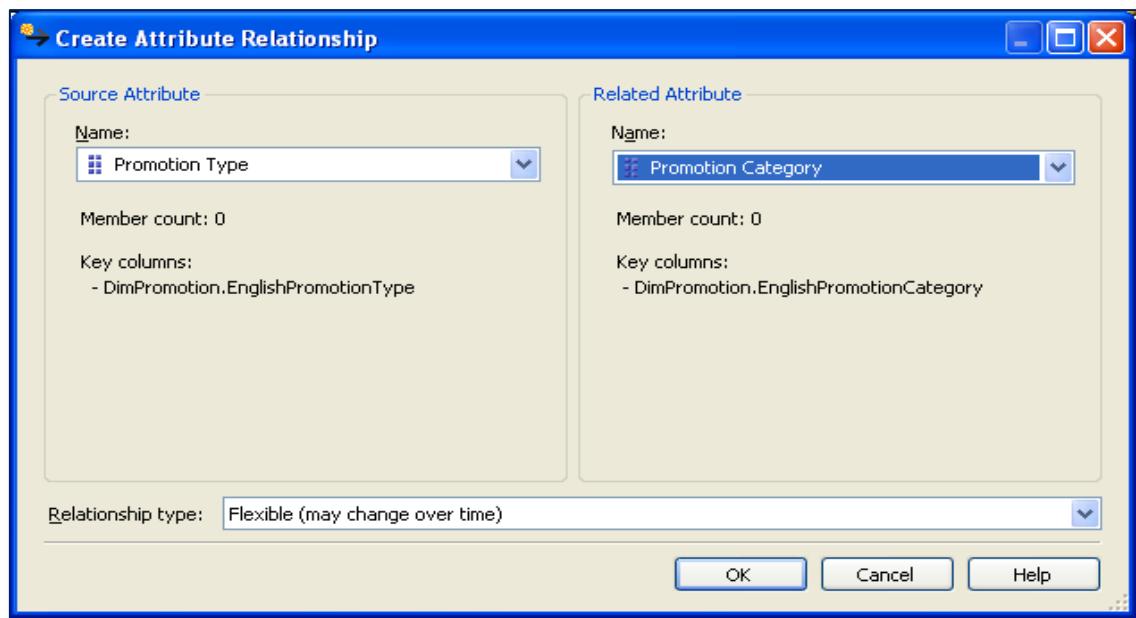
5. La relación de atributos quedará definida de la siguiente manera:



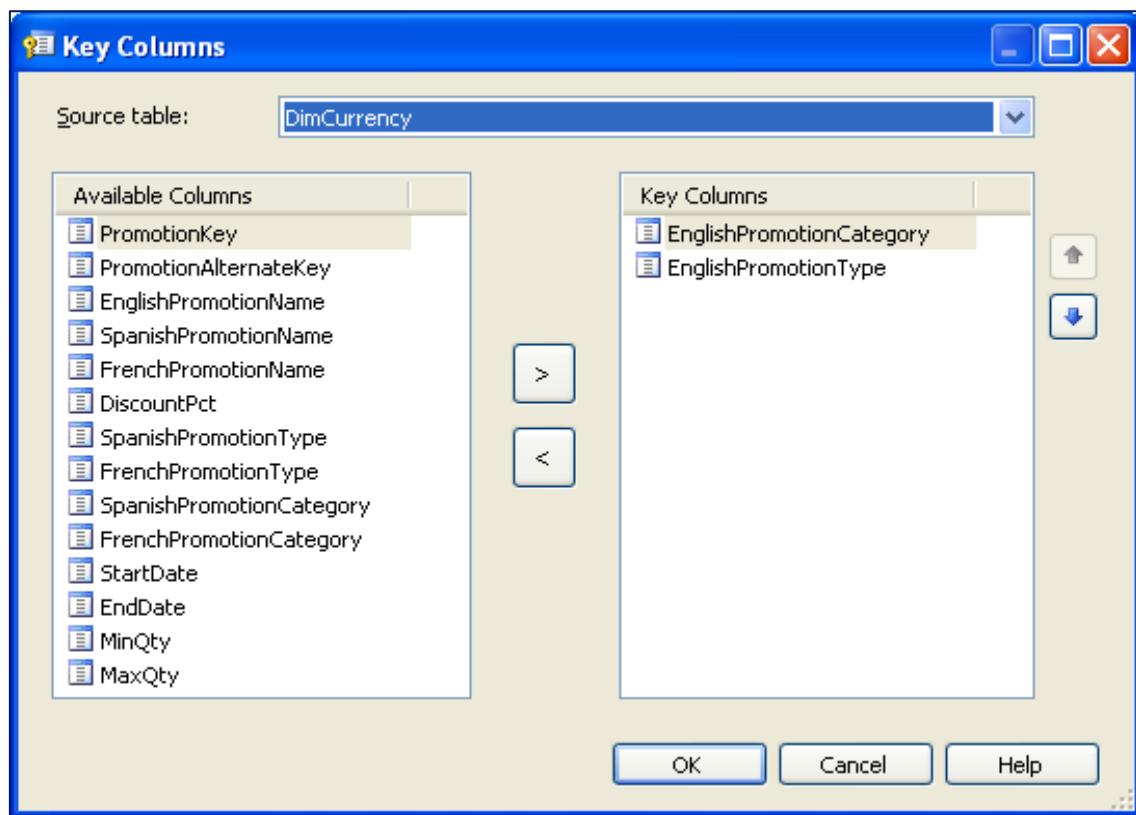
6. Grabe los cambios.
 7. A continuación, se definirán attribute relationships en la dimensión “Promotion”. Para esto, en el Solution Explorer, haga doble clic sobre “Promotion.dim”.
 8. En la pestaña Attribute Relationships, expanda el atributo “Promotion Name”. Este atributo es la clave de la dimensión, y está relacionado con todos los atributos restantes.



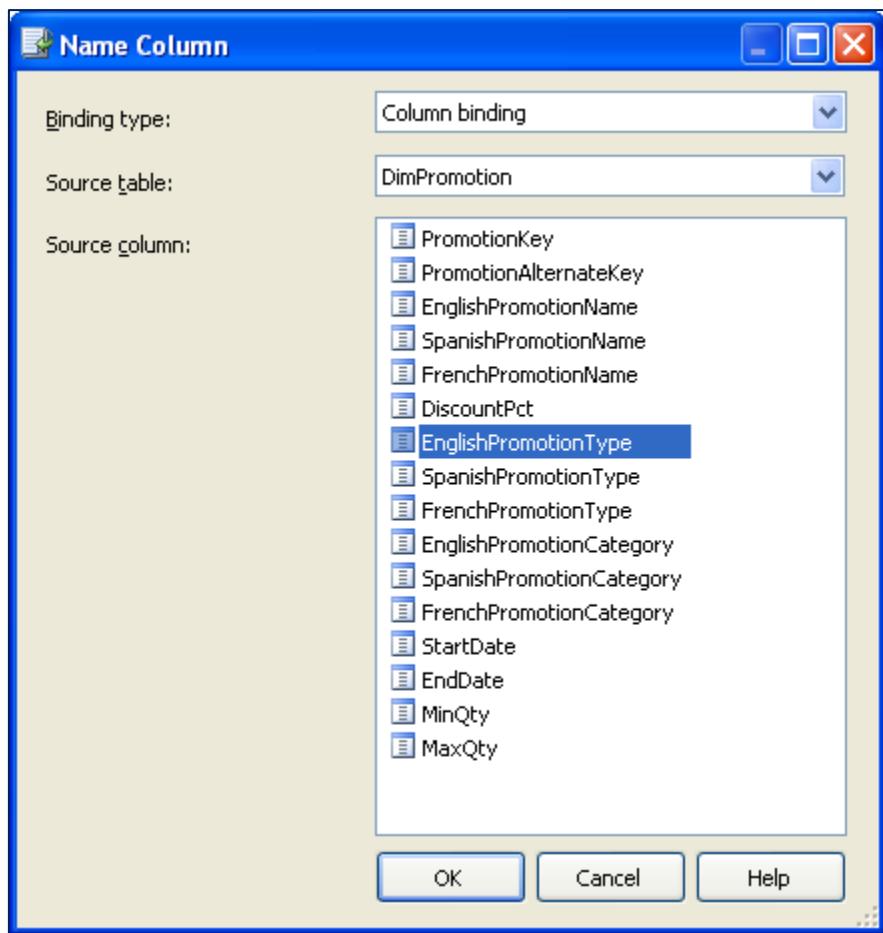
9. Observe la jerarquía “Promotion Categories”. Para optimizar el funcionamiento de esta jerarquía, se definirán las relaciones correspondientes. Seleccione el atributo “Promotion Type”. A continuación, haga clic derecho y cree un New Attribute Relationship. Escoja como atributo origen Promotion Type y como atributo relacionado Promotion Category, como se muestra a continuación:



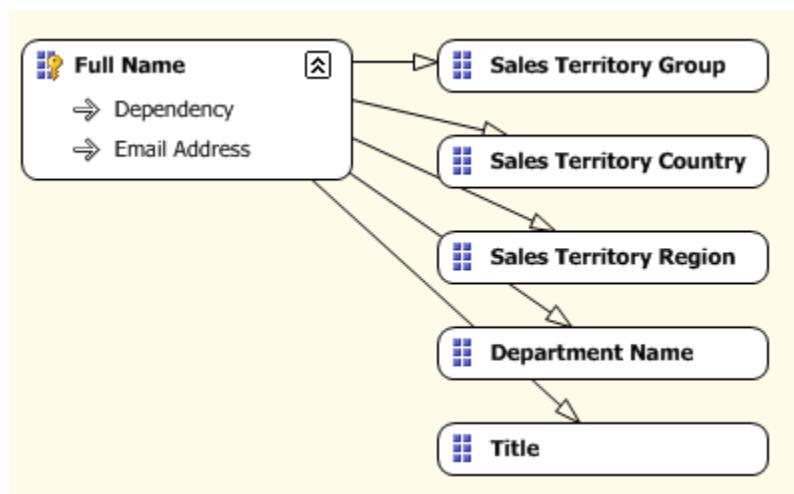
10. A continuación agregue al Key Column del atributo “Promotion Type”, el atributo “English Promotion Category”.



11. Verifique que la propiedad Name Column tenga el valor “EnglishPromotion Type”

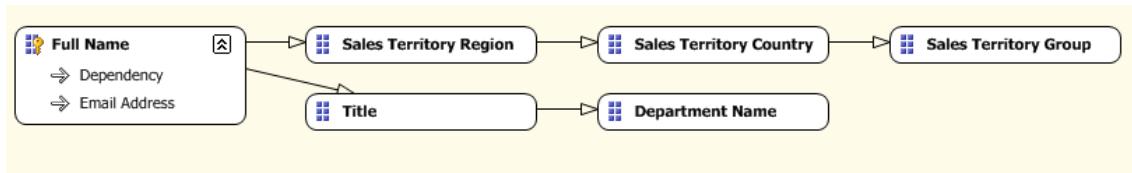


12. Grabe los cambios y cierre el editor de dimensiones.
13. A continuación, se definirán attribute relationships en la dimensión “Employee”. En el Solution Explorer, haga doble clic sobre “Employee.dim”.
14. En la pestaña Attribute Relationships, expanda el atributo “Full Name”. Este atributo es la clave de la dimensión.

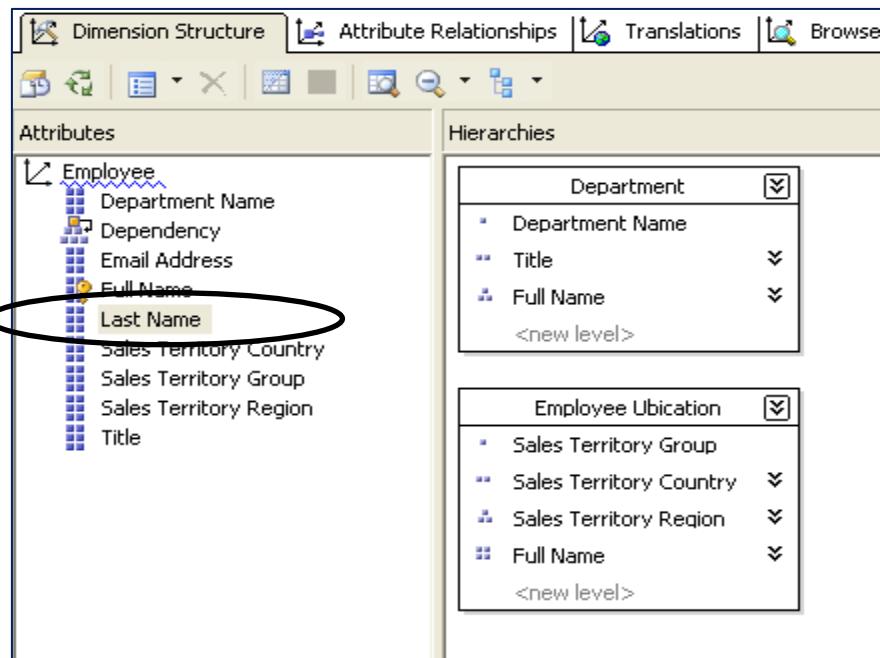


15. Se definirán relaciones que reflejen la estructura de las jerarquías “Department” y “Employee Ubication”. En primer lugar, seleccione el atributo “Title”. A continuación, cree una nueva relación con el atributo “Department Name”.

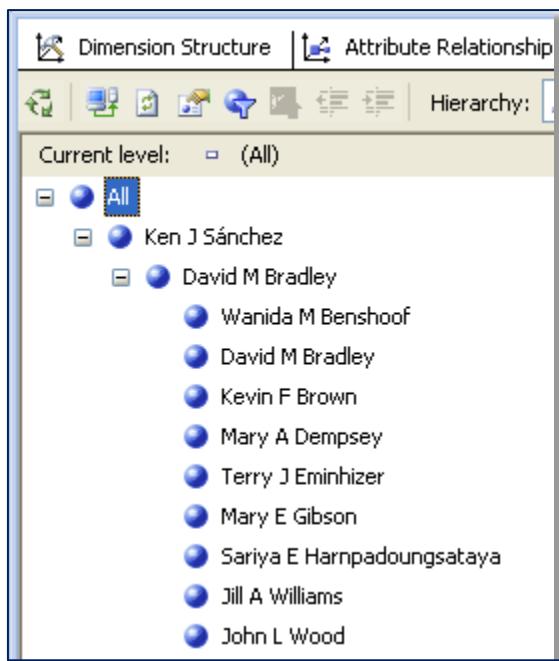
16. Seleccione el atributo “Sales Territory Region” y como siguiente paso cree una relación con el atributo “Sales Territory Country”.
17. Seleccione el atributo “Sales Territory Country” y a continuación cree una relación con el atributo “Sales Territory Group”.



18. Grabe los cambios. Las relaciones creadas hasta este momento tienen como objetivo optimizar el funcionamiento de las jerarquías definidas en las dimensiones. A continuación, se mostrará la creación de una relación para configurar el ordenamiento de un atributo.
19. El atributo “FullName” de la dimensión “Employee” está formado por la concatenación del nombre y apellidos del empleado. Es deseable que los miembros del atributo “FullName” no estén ordenados con base en el nombre, sino en el apellido del empleado. En primer lugar, en el panel de origen de datos, seleccione la columna “LastName” de la tabla “DimEmployee”, y arrástrela bajo la dimensión “Employee” en el panel de atributos. De esta manera, se agregará el atributo “Last Name” a la dimensión. Al finalizar, la estructura de la dimensión “Employee” debe quedar de la siguiente manera:



20. En el panel de atributos, seleccione el atributo “Full Name”. En la ventana de propiedades, establezca la propiedad OrderBy a “AttributeName”, y la propiedad OrderByAttribute a “Last Name”. De esta manera, la información del atributo “Full Name” se ordenará con base en el atributo “Last Name”. Guarde sus cambios, efectúe un deploy del proyecto y consulte la data de la dimensión “Employee”. Observe el ordenamiento.

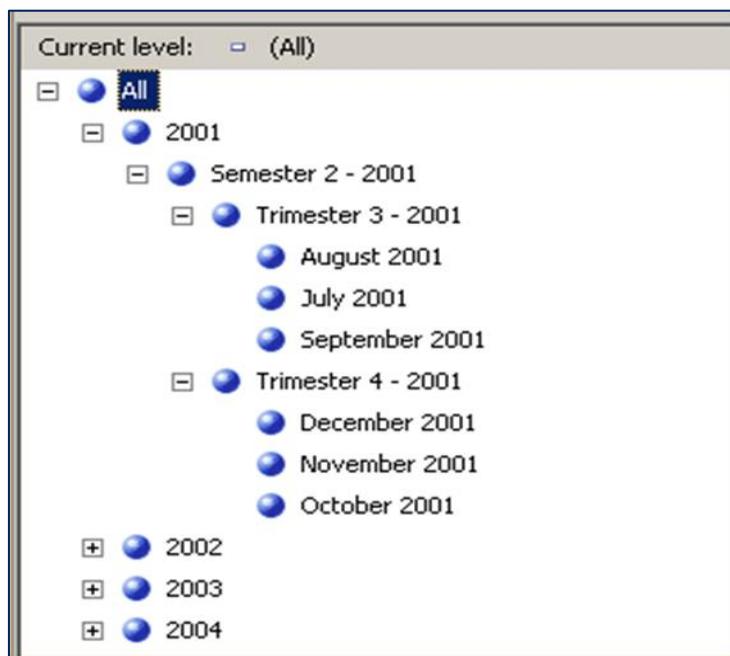


2.1.4. Creando Dimensión Tiempo

Dimensión Time

Todo proyecto de Business Intelligence requiere de la existencia de una dimensión que almacene información de fecha e intervalos temporales de tiempo. SSAS permite crear dimensiones del tipo tiempo permitiendo crear la entidad en el origen de datos o crearla en el servidor SSAS. En el siguiente ejercicio crearemos una dimensión tiempo con una entidad ya existente en el origen de datos.

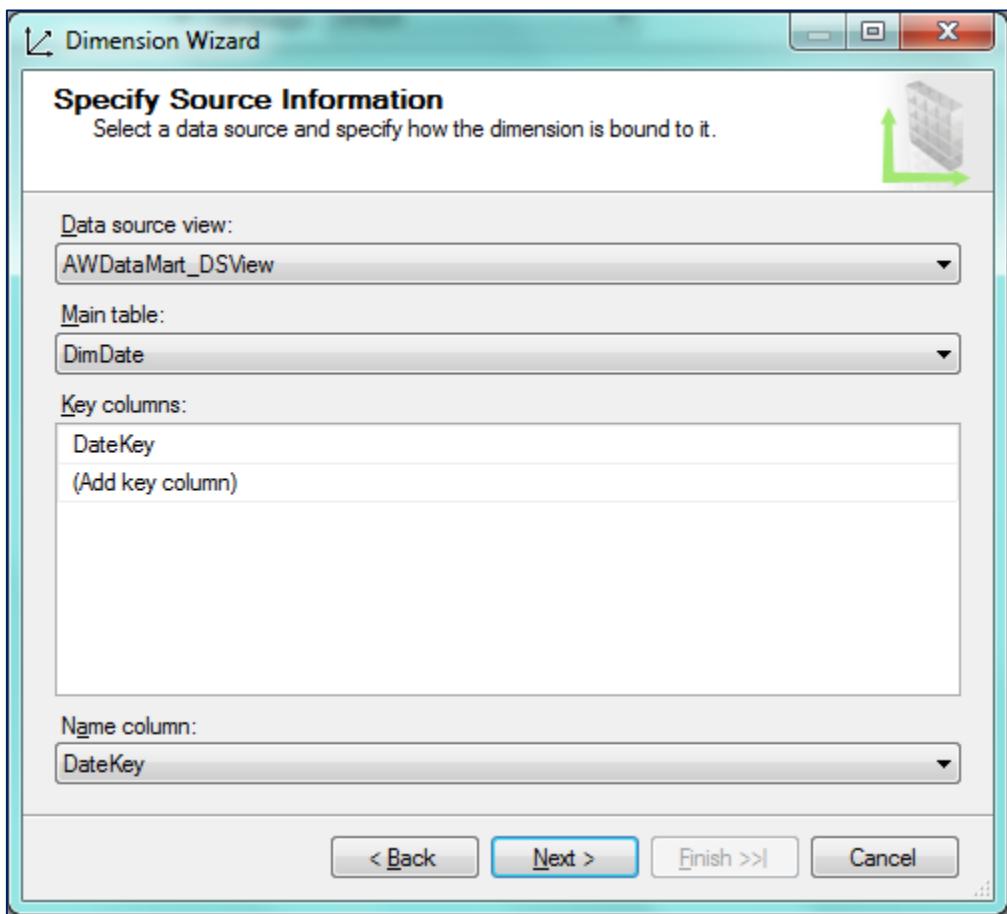
La mayoría de soluciones de BI utilizan una dimensión para almacenar intervalos de tiempo.



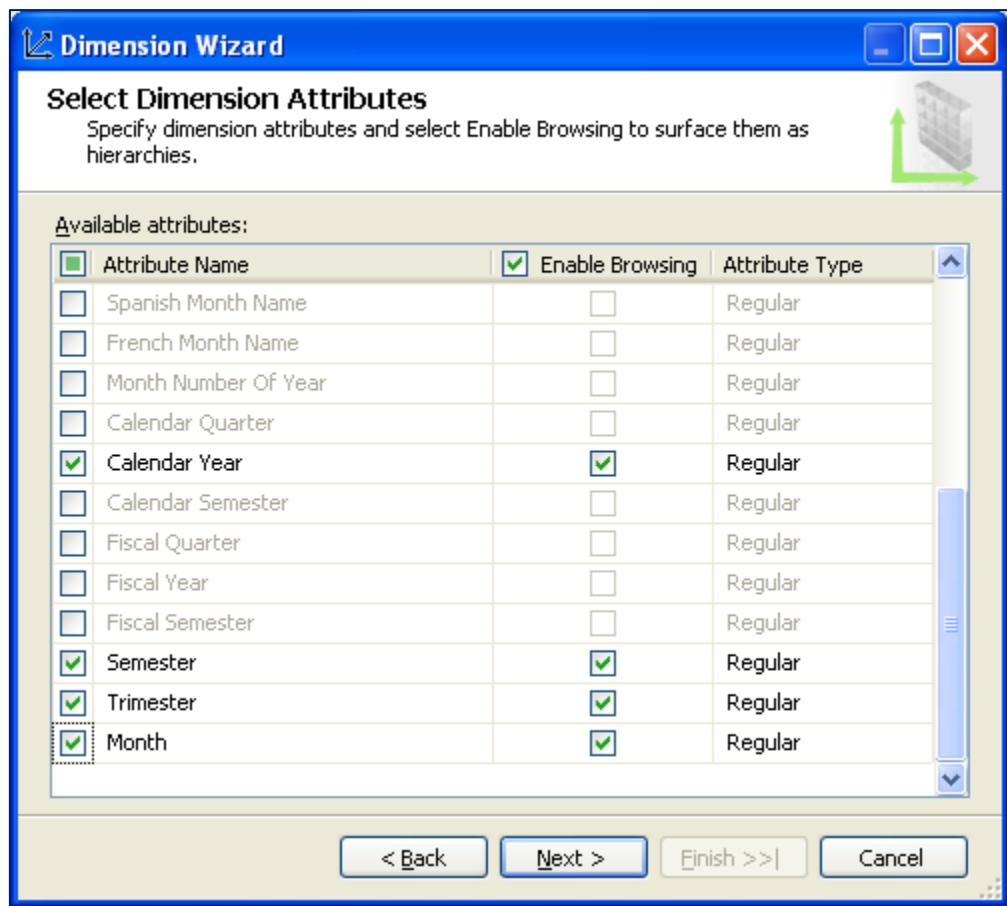
Ejercicio:

Creación de una Dimensión Time

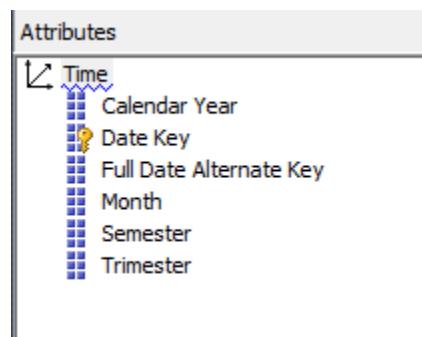
1. Inicie la creación de una nueva dimensión. En la ventana de selección del método de creación de la dimensión, asegúrese de que la opción Use an existing table esté marcada.
2. En la siguiente ventana seleccione como Main Table la tabla DimDate, como Key Column el atributo DateKey y como Name Column el atributo DateKey.



3. En la siguiente ventana mantenga el atributo DateKey, y agregue los siguientes atributos: Full Date Alternate Key, Calendar Year, Semester, Trimester, Month.



4. Continúe con el asistente. Al finalizar, establezca el nombre de la nueva dimensión a “Time”. En el panel de atributos, la estructura debe quedar de la siguiente manera:



5. Renombre el atributo “Calendar Year” como “Year”.
 6. En la dimensión “Time”, agregue una nueva jerarquía llamada “Calendar” con la siguiente estructura de niveles:

- Year
- Semester
- Trimester
- Month

Al finalizar, la estructura de la dimensión “Time” debe quedar de la siguiente manera:

The screenshot shows the 'Dimension Structure' interface. The 'Attributes' pane on the left lists various attributes under a 'Time' dimension, including 'Year', 'Date Key', 'Full Date Alternate Key', 'Month', 'Semester', and 'Trimester'. The 'Hierarchies' pane on the right displays a hierarchy named 'Calendar' with levels: Year, Semester, Trimester, and Month. A tooltip on the right side of the hierarchy pane says 'To create hierarchy attribute'.

7. Observe el atributo “DateKey”. Este es el atributo clave de la dimensión. En la ventana de propiedades, observe su propiedad NameColumn. Puede observar que, al consultar los miembros de este atributo, los usuarios visualizarán los valores del campo “DateKey” de la tabla “DimTime”. Esto no resulta de mucha utilidad. Sin embargo, el atributo “DateKey” es necesario para sostener la estructura de la dimensión. Por tanto, este atributo debe configurarse para no ser visible para las consultas. Para conseguirlo, en la ventana de propiedades, establezca su propiedad AttributeHierarchyVisible a “False”.
8. Establezca los attribute relationships necesarios para optimizar la jerarquía “Calendar” como se muestra en la siguiente figura:

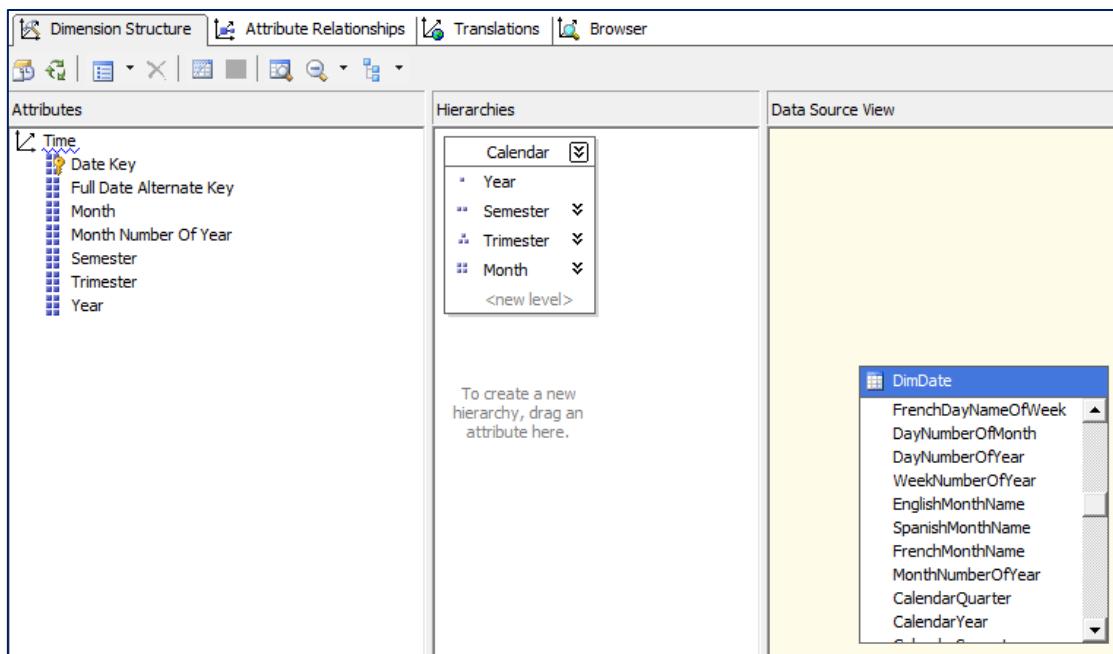


9. Guarde los cambios y efectúe un deployment del proyecto. A continuación, consulte la información de la jerarquía “Calendar”:

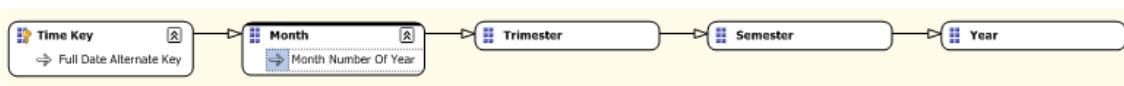
The screenshot shows the 'Dimension Structure' interface with the 'Hierarchy' pane selected. The hierarchy 'Calendar' is expanded to show the following structure: All, 2001, Semester 2 – 2001 (containing Trimester 3 – 2001, August 2001, July 2001, September 2001), Trimester 4 – 2001 (containing December 2001, November 2001, October 2001), 2002, 2003, and 2004.

Observe que los meses están ordenados de forma alfabética, lo cual constituye un error.

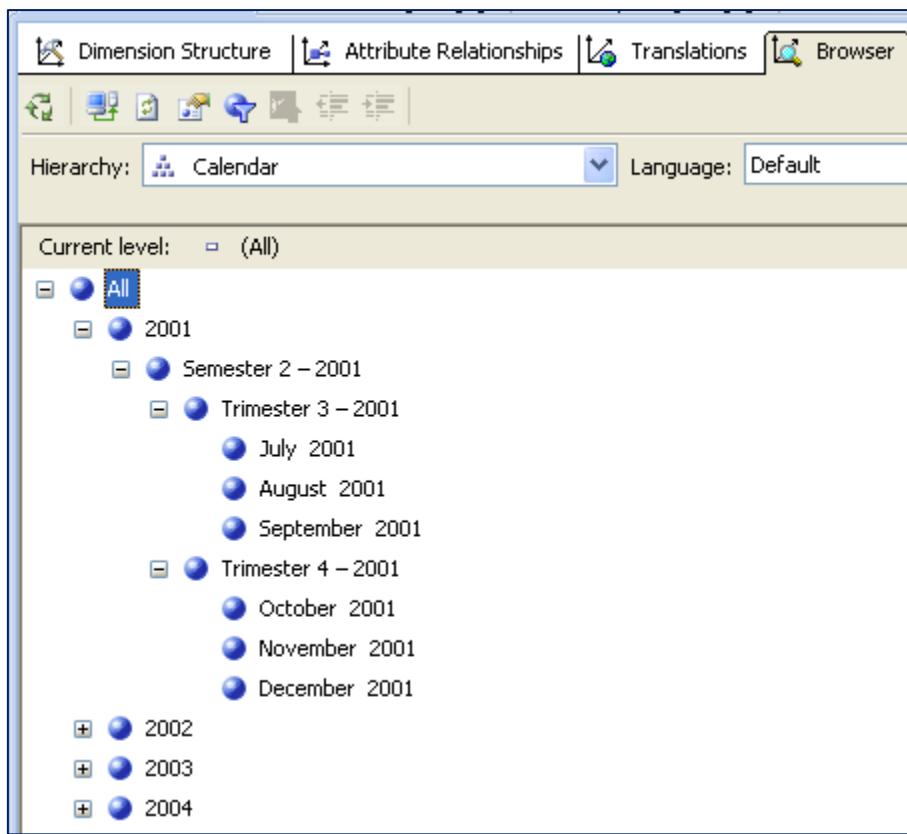
10. En el panel de origen de datos, seleccione la columna “MonthNumberOfYear” de la tabla DimDate, y arrástrela hacia el panel de atributos, debajo de la dimensión “Time”. De esta manera, se ha creado el atributo “Month Number Of Year” en la dimensión.



11. Cada atributo adicional que se agrega a una dimensión eleva el tiempo de procesamiento y el espacio en disco. Sin embargo, el atributo “Month Number Of Year” no se utilizará directamente en las consultas; su única finalidad es servir como criterio de ordenación para el atributo “Month”. Por tanto, se debe habilitar el uso del atributo “Month Number Of Year” en las consultas. Para esto, en el panel de atributos seleccione el atributo “Month Number Of Year”, y en la ventana de propiedades, establezca su propiedad AttributeHierarchyEnabled a “False”.
12. A continuación, se debe crear una relación entre el atributo “Month” y el atributo “Month Number Of Year”. Para esto, diríjase a la pestaña Attribute Relationships y cree la relación. Al final, la estructura debe quedar de la siguiente manera:



13. En el panel de atributos, seleccione el atributo “Month”. En el panel de resultados, establezca su propiedad OrderBy a “AttributeName”, y su propiedad OrderByAttribute a “Month Number Of Year”.
14. Guarde los cambios y efectúe un deployment del proyecto. Consulte la información de la jerarquía “Calendario”, y verifique que los meses están ordenados correctamente.



2.1.5. Agrupando Miembros de un atributo

Agrupar los miembros de un atributo

Una funcionalidad importante de SSAS es su capacidad de efectuar agrupaciones automáticas de los miembros de un atributo. Por ejemplo, un atributo puede almacenar los precios de los productos de la empresa. Sin embargo, expandir dicho atributo durante una consulta, y observar una extensa lista de precios puede resultar de poca utilidad para el usuario final. Para resolver esto, SSAS puede efectuar una agrupación automática, de manera que el usuario, al consultar la información, no observa cada precio en particular, sino rangos de precios.

Las características de agrupamiento de un atributo se establecen a través de las propiedades DiscretizationMethod (que establece el método de agrupamiento) y DiscretizationBucketCount (que establece el número de grupos que se generarán). Están implementados tres métodos de agrupamiento:

- Equal Areas: Examina los valores del atributo, y crea grupos que tienen igual número de valores.
- Clusters: Utiliza el algoritmo K – Means para obtener los grupos a través de distribuciones de Gauss. Sólo puede usarse con valores numéricos.
- Equal Ranges: Cada grupo obtenido corresponde con un rango de valores. Por ejemplo, si se tienen valores entre 0 y 100, y la propiedad DiscretizationBucketCount está establecida a 4, se generarán cuatro grupos: (0 – 25, 26 – 50, 51 – 75, 76 – 100).

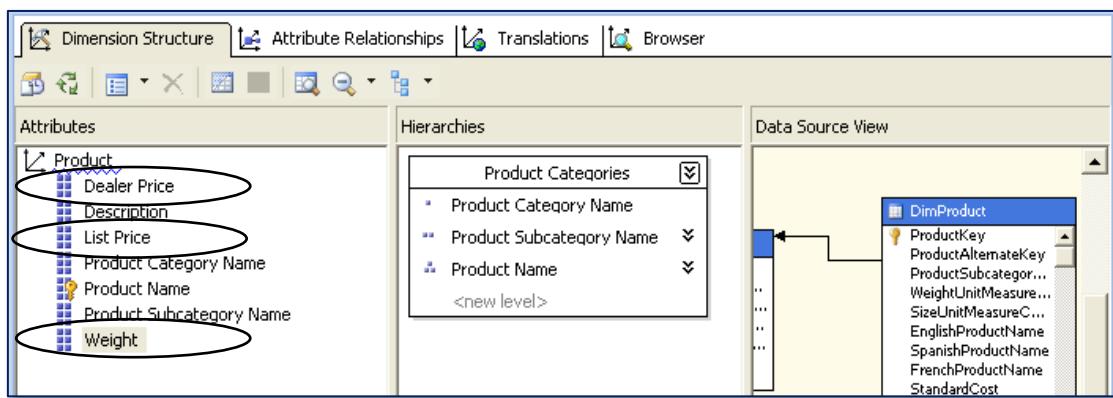
El valor por defecto para la propiedad DiscretizationBucketCount es de 0. De esta forma, SSAS determinará automáticamente el número de grupos a través de un

muestreo de los datos del atributo. Sin embargo, si se desea controlar el número de grupos generados, puede establecerse esta propiedad a un número mayor que 0. Debe tenerse en cuenta que es posible que, aun así, el algoritmo no produzca exactamente el número de grupos deseado, pues esto depende de la distribución estadística de la data.

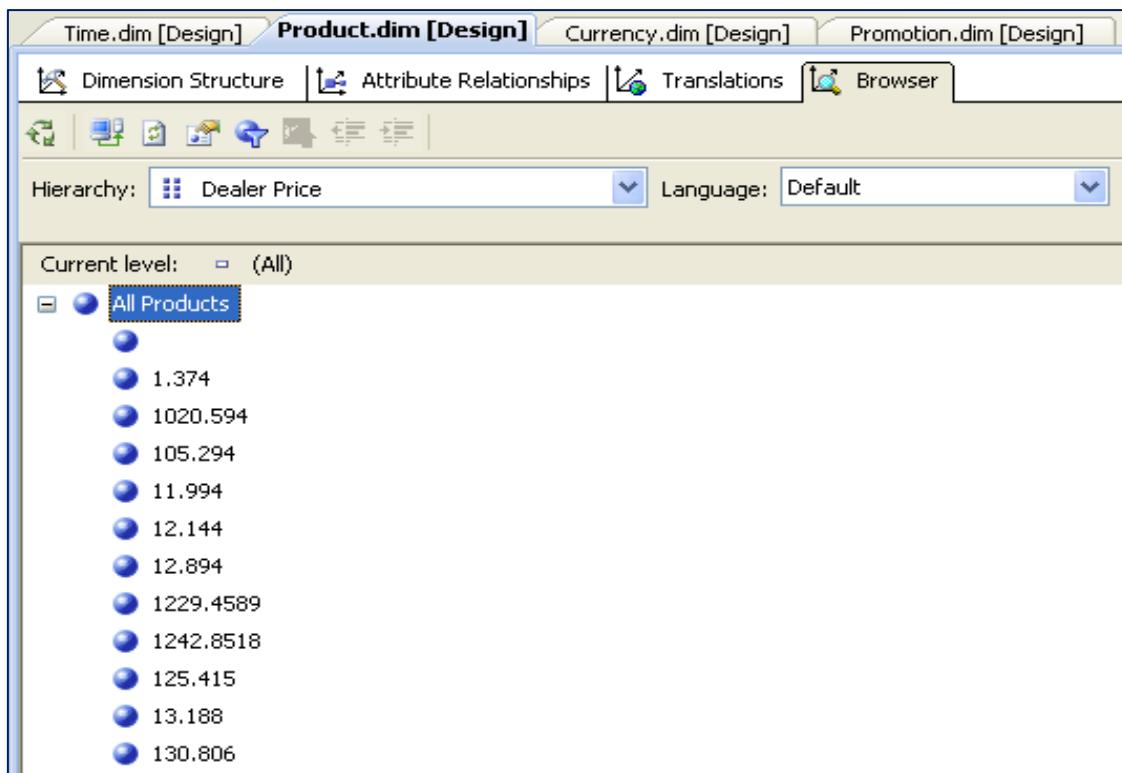
Ejercicio:

Definir propiedades de agrupamiento en la Dimensión Product.

1. En el Solution Explorer, haga doble clic sobre “Product.dim”.
2. En el panel de origen de datos, seleccione la columna “DealerPrice” y arrástrela sobre el panel de atributos, debajo de la dimensión “Product”. Haga lo mismo con las columnas “ListPrice” y “Weight”. Al finalizar, la estructura de la dimensión “Product” debe quedar de la siguiente manera:

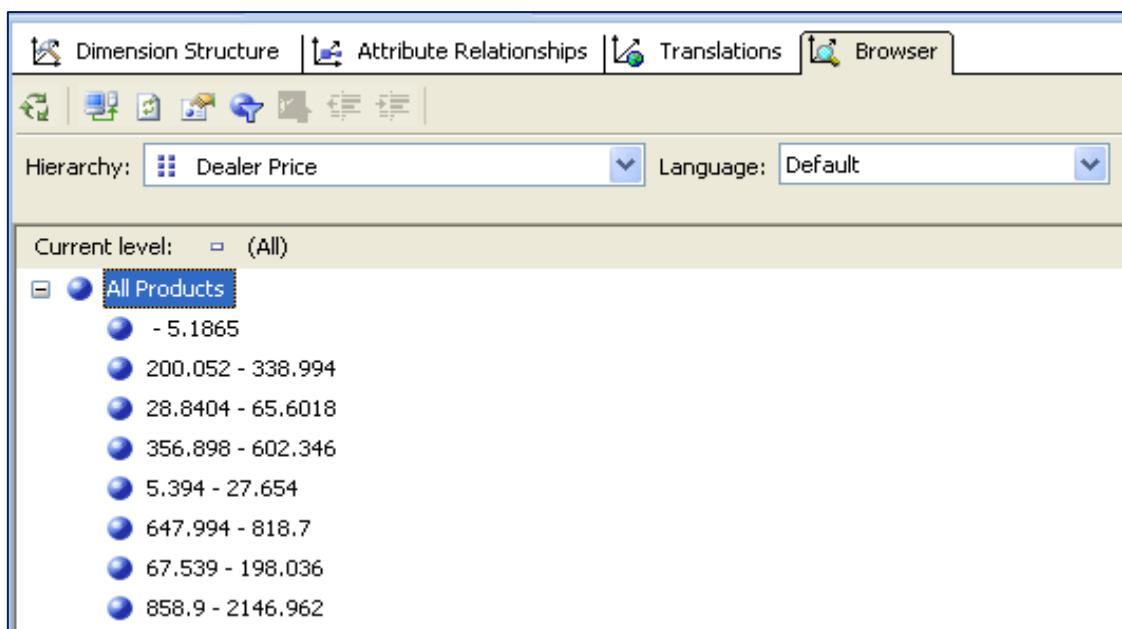


3. Guarde los cambios y efectúe un deployment del proyecto. Consulte la dimensión y observe la data del atributo “Dealer Price”:



Observe que los miembros del atributo “Dealer Price” son todos los valores que toma esta columna en la tabla de dimensión. Por supuesto, sería más conveniente que SSAS generase rangos significativos con estos valores. Los siguientes pasos configurarán este comportamiento.

4. En el panel de atributos, seleccione el atributo “Dealer Price”. En la ventana de propiedades, establezca su propiedad DiscretizationMethod a “Equal Areas”, y la propiedad DiscretizationBucketCount a 0.
5. En el panel de atributos, seleccione el atributo “List Price”. En la ventana de propiedades, establezca su propiedad DiscretizationMethod a “Equal Areas”, y la propiedad DiscretizationBucketCount a 0.
6. En el panel de atributos, seleccione el atributo “Weight”. En la ventana de propiedades, establezca su propiedad DiscretizationMethod a “Clusters”, y la propiedad DiscretizationBucketCount a 5.
7. Guarde los cambios y efectúe un deployment del proyecto. A continuación, consulte la información del atributo “Dealer Price”. Observe los grupos generados por SSAS.



2.2. DEFINIENDO MEDIDAS

2.2.1. Creando Cubos, medidas y grupos de medidas

Cubos

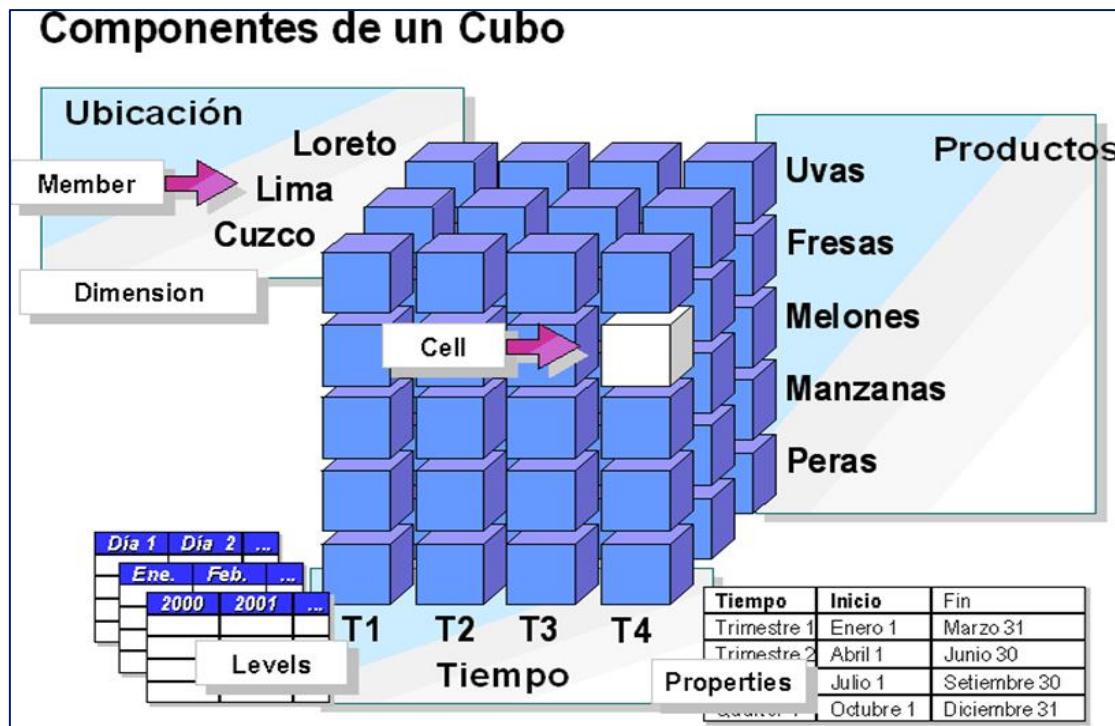
Un cubo contiene un subconjunto de la información de un Data Mart o Data Warehouse. Su información se almacena en una estructura multidimensional definida por dimensiones y medidas.

Todo cubo se construye a partir de un esquema de datos que consiste en tablas de hechos y tablas de dimensión. Una mejora importante de SSAS 2012 sobre la versión 2000 es que cada cubo puede contener múltiples sola tablas de hechos.

Cada dimensión puede contener varias jerarquías. Cada jerarquía contiene varios niveles. Los niveles permiten efectuar complejas operaciones de drill down (aumentar el nivel de detalle de las consultas) y drill up (disminuir el nivel de detalle de las consultas).

Por ejemplo, una operación de drill down puede consistir en observar la información de ventas de manzanas en todo el Perú, y luego aumentar el nivel de detalle para consultar las ventas de Cuzco y Lima. Una operación de drill up típica consistiría en observar las ventas de Lima, y disminuir posteriormente el nivel de detalle para observar las ventas en todo el Perú.

La información se sumariza en un cubo de acuerdo a la estructura de niveles y jerarquías definida en cada dimensión. Por ejemplo, en la tabla de hechos de un cubo pueden almacenarse las ventas para cada distrito. Durante el procesamiento del cubo, se calculan y almacenan las sumas parciales de las ventas en cada provincia y departamento. Estas sumas parciales reciben el nombre de agregaciones. De esta manera, se pueden efectuar consultas rápidas sobre vastos conjuntos de datos.



Propiedades de los cubos

Los cubos tienen una serie de propiedades, configurables desde el diseñador de cubos de SQL Server Data Tools. A continuación se explican algunas de estas propiedades:

- Name: Nombre del cubo.
- Visible: Determina si el cubo es visible o no.
- Default Measure: La medida devuelta por el cubo en caso de que las consultas no especifiquen de forma explícita qué medida se desea visualizar.

Medidas y Grupos de Medidas

Una medida (measure) representa un valor obtenido a partir de una columna numérica de la tabla de hechos. La definición de las medidas constituye el paso central en el análisis de soluciones de inteligencia de negocios, pues constituyen la información que los usuarios finales desean visualizar.

Las medidas se agrupan en grupos de medidas (measure groups). Cada measure group corresponde con una tabla de hechos. Un cubo de SSAS 2012 puede contener varios measure groups.

Propiedades de los measure groups

A continuación se explican las propiedades principales de los measure groups:

- Name: Nombre de measure group. Por defecto, el nombre del measure group se forma a partir de la tabla de hechos correspondiente.
- Storage location: Directorio donde se ubicará físicamente la data del measure group.

Propiedades de las medidas

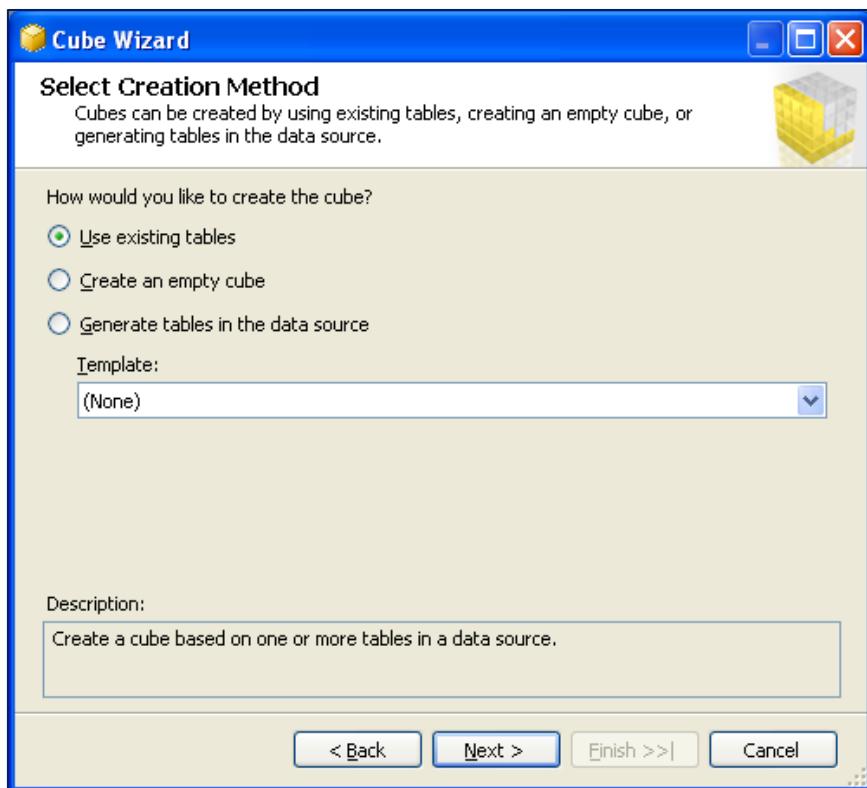
Las medidas poseen una serie de propiedades. A continuación, se explican algunas de ellas:

- Name: Nombre de la medida.
- Description: Descripción de la medida.
- Source: Columna de la tabla de hechos a partir de la cual se obtiene la medida.
- Format String: Determina el formato de visualización de la medida. Analysis Services provee una serie de formatos (por ejemplo, porcentaje y moneda), y también permite ingresar una cadena de formato personalizada.
- Visible: Determina si la medida es visible o no. Esta propiedad es útil cuando se definen medidas intermedias, a partir de las cuales se calcularán otras medidas a través de MDX.
- Data Type: Tipo de dato de la medida. Por defecto, es heredado del tipo de dato de la columna que genera la medida.
- Aggregate Function: Función que se utilizará para sumarizar la medida. Analysis Services provee las siguientes funciones de agregación:
 - SUM: Es la función de agregación por defecto. Significa que el valor de la medida para un miembro de una dimensión es igual a la suma de los valores para todos sus miembros hijos. Ejemplo: la venta del Perú es la suma de las ventas de todos sus departamentos.
 - COUNT: Recupera la cuenta de registros en la tabla de hechos. Esta función puede responder a preguntas del tipo: “¿cuántas ventas se han producido en Enero del 2005?”
 - MIN: Significa que el valor de la medida para un miembro es igual al valor mínimo encontrado entre sus miembros hijos. A través de esta función se puede saber, por ejemplo, cuál ha sido la venta más pequeña entre los distritos de Lima.
 - MAX: Significa que el valor de la medida para un miembro es igual al valor máximo encontrado entre sus miembros hijos. A través de esta función se puede saber, por ejemplo, cuál ha sido la venta más grande entre los distritos de Lima.
 - DISTINCT COUNT: Recupera la cuenta de registros en la tabla de hechos, eliminando las ocurrencias repetidas. Esta función puede responder a preguntas del tipo: “entre mis 150 clientes, ¿cuántos han efectuado compras en Enero del 2005?”
 - NONE: No se efectúa ningún tipo de agregación; es decir, la medida no es aditiva. Esta función de agregación debe usarse cuando los valores de la medida no deben ser acumulados; por ejemplo, si la medida representa un promedio o el resultado de un cociente.
 - BY ACCOUNT: Agrega la medida de acuerdo con la función de agregación asignada a las dimensiones de tipo “Account”. Si el cubo no tiene dimensiones de tipo “Account”, este valor es ignorado y se trata como una función NONE.
 - AVERAGE OF CHILDREN: Significa que el valor de la medida para un miembro de una dimensión es igual al promedio de los valores para todos sus miembros hijos no vacíos.
 - FIRST CHILD: Significa que el valor de la medida para un miembro es igual al valor de la medida para su primer miembro hijo.
 - LAST CHILD: Significa que el valor de la medida para un miembro es igual al valor de la medida para su último miembro hijo.
 - FIRST NON EMPTY: Significa que el valor de la medida para un miembro es igual al valor de la medida para su primer miembro hijo no vacío.
 - LAST NON EMPTY: Significa que el valor de la medida para un miembro es igual al valor de la medida para su último miembro hijo no vacío.

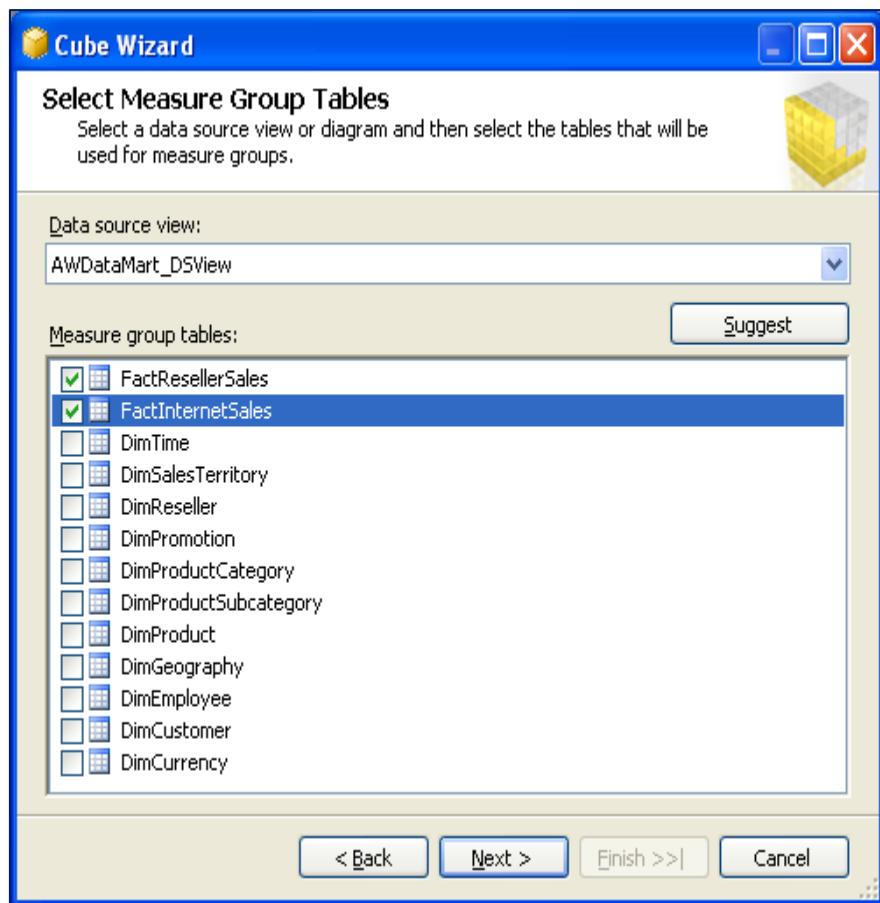
Ejercicio

Crear el cubo de ventas a través de Cube Editor

1. Desde el SQL Server Data Tools, abra el proyecto AdventureWorksDataMart. En el Solution Explorer, seleccione la carpeta Cubes. Haga un clic derecho, y seleccione la opción New Cube.
2. Se iniciará el asistente para la creación de cubos (Cube Wizard). Pulse el botón Next para continuar.
3. La ventana siguiente permite seleccionar el método de creación del cubo. Debe estar seleccionada la opción Use existing tables.



4. Pulse el botón Next. La siguiente ventana permite seleccionar el data source view a partir del cual se definirá la estructura del cubo. Seleccione el data source view "AWDataMart_DSView", y en la sección Measures group tables: establezca las tablas "FactInternetSales" y "FactResellerSales".



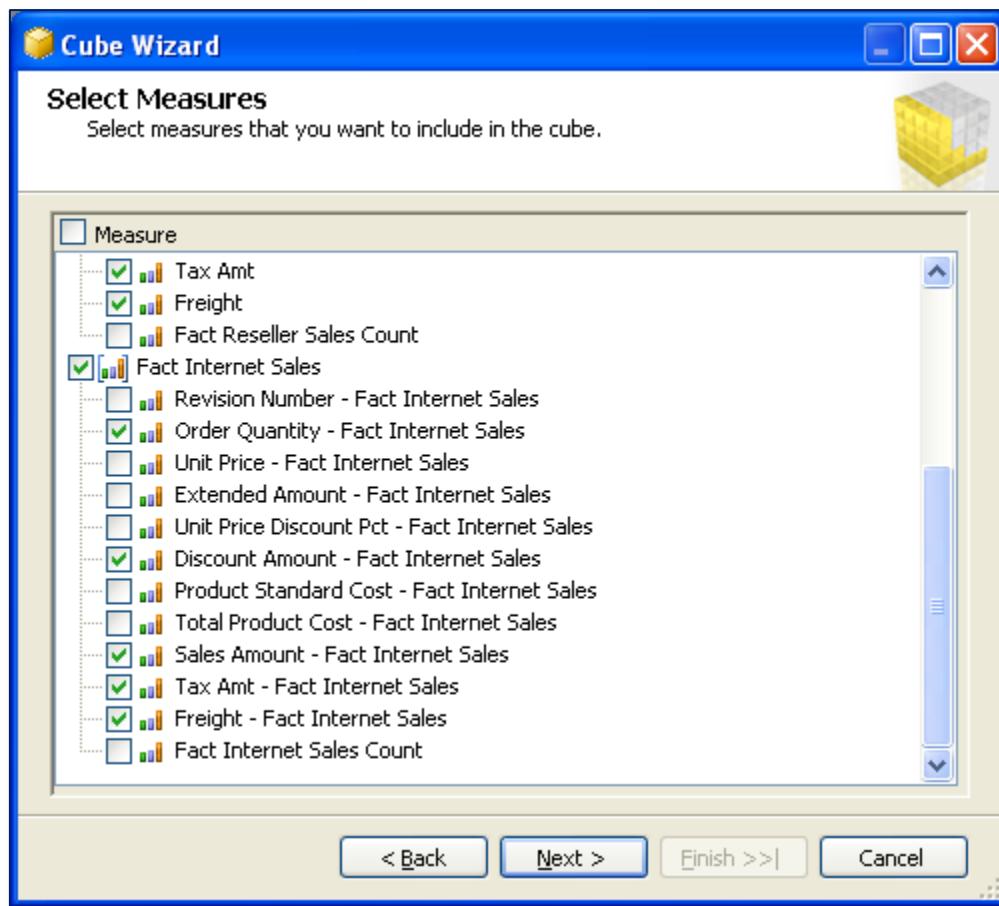
5. La siguiente ventana muestra los measure groups y las medidas que pueden definirse a partir de las columnas en las tablas de hechos. Sólo deben estar seleccionadas las columnas siguientes:

En el measure group Fact Reseller Sales

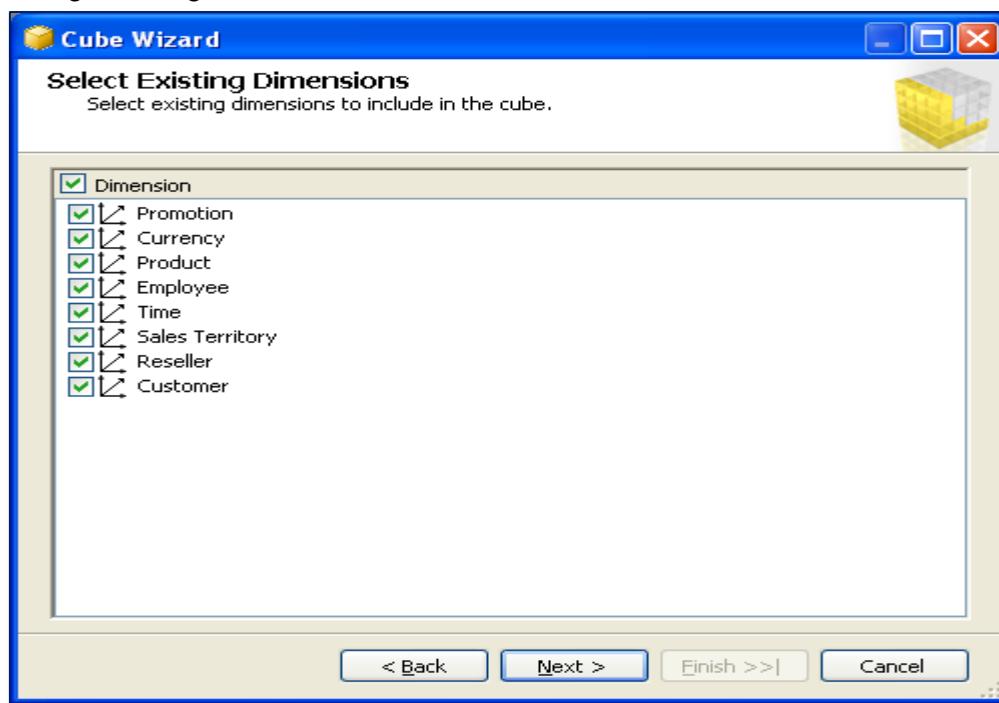
Order Quantity
Discount Amount
Sales Amount
Tax Amt
Freight

En el measure group Fact Internet Sales

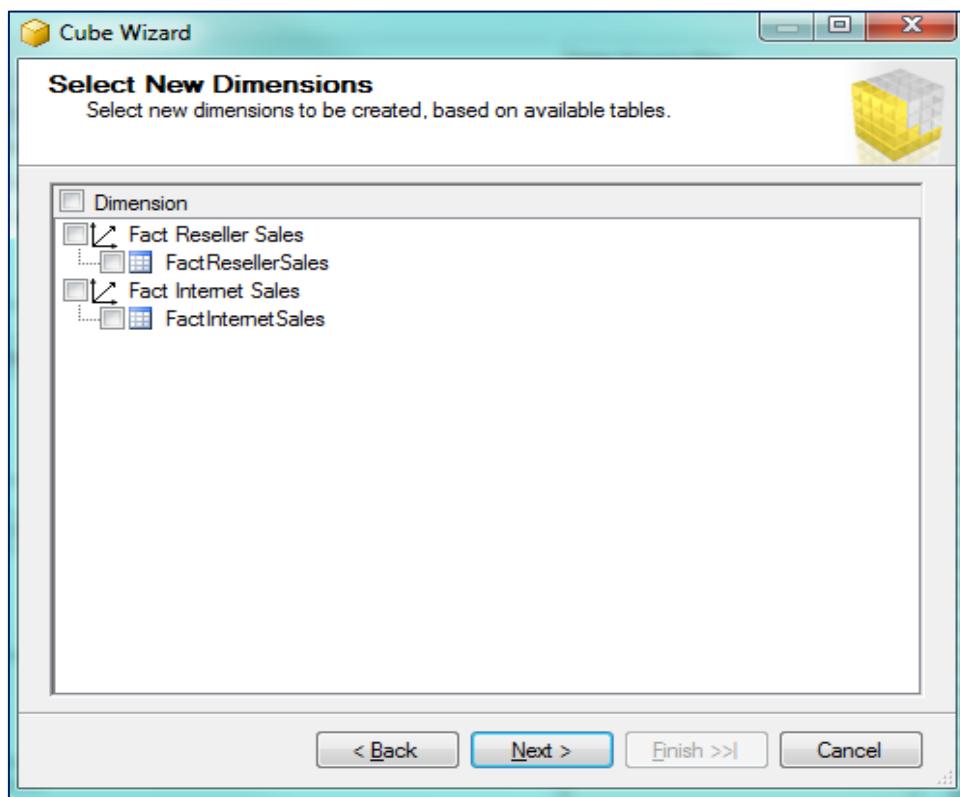
Order Quantity – Fact Internet Sales
Discount Amount – Fact Internet Sales
Sales Amount – Fact Internet Sales
Tax Amt – Fact Internet Sales
Freight – Fact Internet Sales



6. Pulse el botón Next. La siguiente ventana permite seleccionar las dimensiones que utilizará el cubo. Observe que, en la lista aparecen las dimensiones desarrolladas en el capítulo anterior. las dimensiones, como se muestra en la siguiente figura:



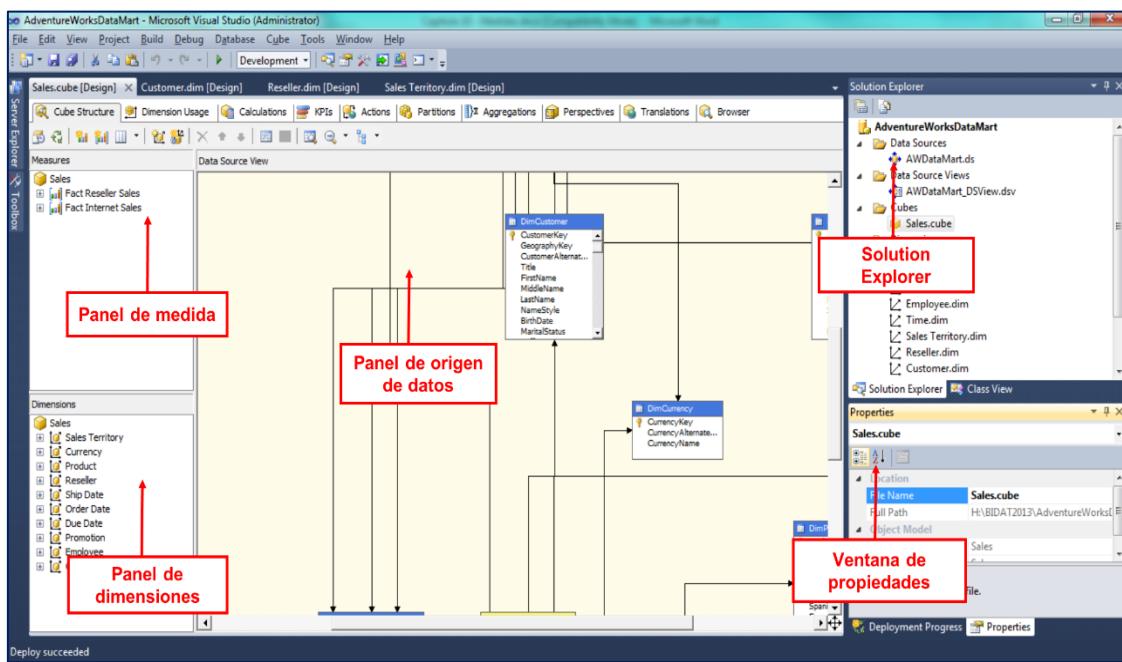
En la siguiente pantalla, desmarque todos los casilleros.



7. Pulse el botón Next. En la ventana final del asistente, establezca el nombre del nuevo cubo a "Sales":



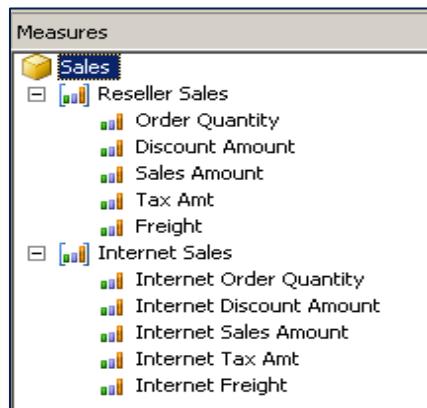
8. Pulse el botón Finish para finalizar la creación del cubo. Al finalizar, se mostrará el diseñador de cubos.



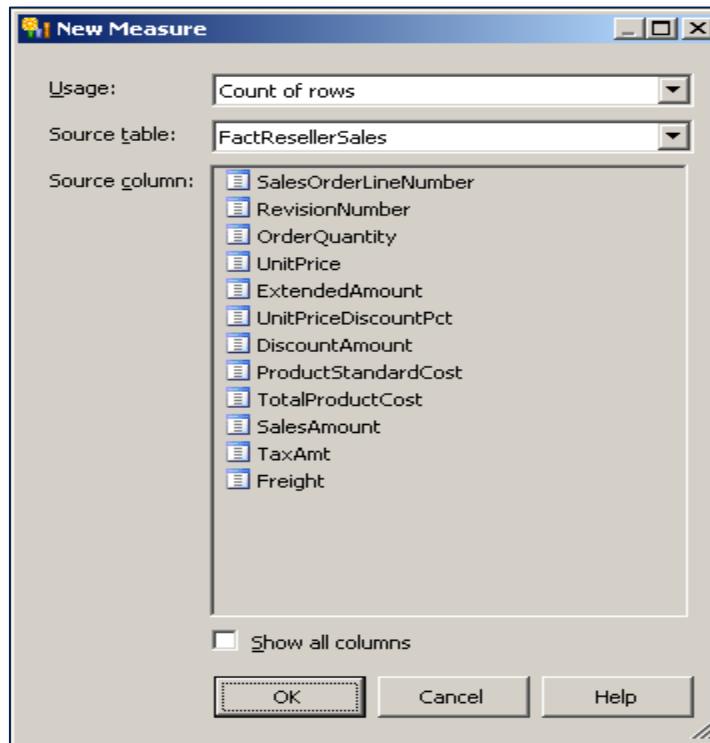
9. En el panel de medidas, expanda el measure group “Fact Internet Sales”. Seleccione la medida “Order Quantity - Fact Internet Sales”. En la ventana de propiedades, cambie la propiedad Name de esta medida a “Internet Order Quantity”. Repita estos pasos para renombrar las medidas restantes según el siguiente cuadro:

Nombre anterior	Nuevo nombre
Discount Amount - Fact Internet Sales	Internet Discount Amount
Sales Amount - Fact Internet Sales Internet Sales Amount	Internet Sales Amount
Tax Amt - Fact Internet Sales	Internet Tax Amt
Freight - Fact Internet Sales	Internet Freight

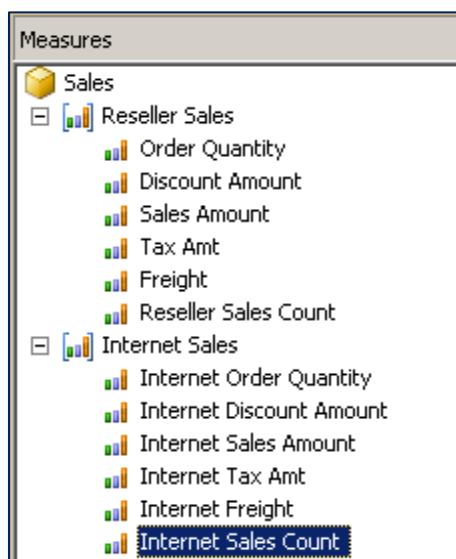
10. En el panel de medidas, seleccione el measure group “Fact Reseller Sales”. En la ventana de propiedades, cambie la propiedad Name del measure group a “Reseller Sales”.
11. En el panel de medidas, seleccione el measure group “Fact Internet Sales”. En la ventana de propiedades, cambie la propiedad Name del measure group a “Internet Sales”. Al finalizar estos pasos, la estructura de las medidas debe verse de la siguiente manera en el panel de medidas:



12. A continuación, se agregarán dos medidas de tipo Count. En el panel de medidas, seleccione el measure group “Reseller Sales”. Haga un clic derecho, y seleccione la opción “New measure”. En la propiedad Usage, seleccione el tipo de función “Count of rows”. En la propiedad Source table, seleccione la tabla “FactResellerSales”.



13. Pulse el botón OK para aceptar. Se habrá creado una nueva medida llamada “Reseller Sales Count” en el measure group “Reseller Sales”. Esta medida efectúa un conteo de las filas de la tabla de hechos FactResellerSales.
14. A continuación, repita el paso (13) para crear una nueva medida llamada “Internet Sales Count” en el measure group “Internet Sales”. Esta medida debe ser de tipo “Count of rows”. Al finalizar, la estructura de las medidas del cubo “Sales” debe quedar de la siguiente manera:



15. Grabe los cambios y efectúe un deployment del proyecto. Al finalizar el deployment, efectúe consultas sobre el cubo a través del tab Browser.

2.2.2. Relacionando dimensiones y grupos de medidas

En un cubo, las dimensiones deben estar relacionadas con los measure groups. Comúnmente, estas relaciones se efectúan a través de columnas comunes en las tablas de dimensión y las tablas de hechos.

SSAS 2014 soporta los siguientes tipos de relaciones entre las dimensiones y los measure groups:

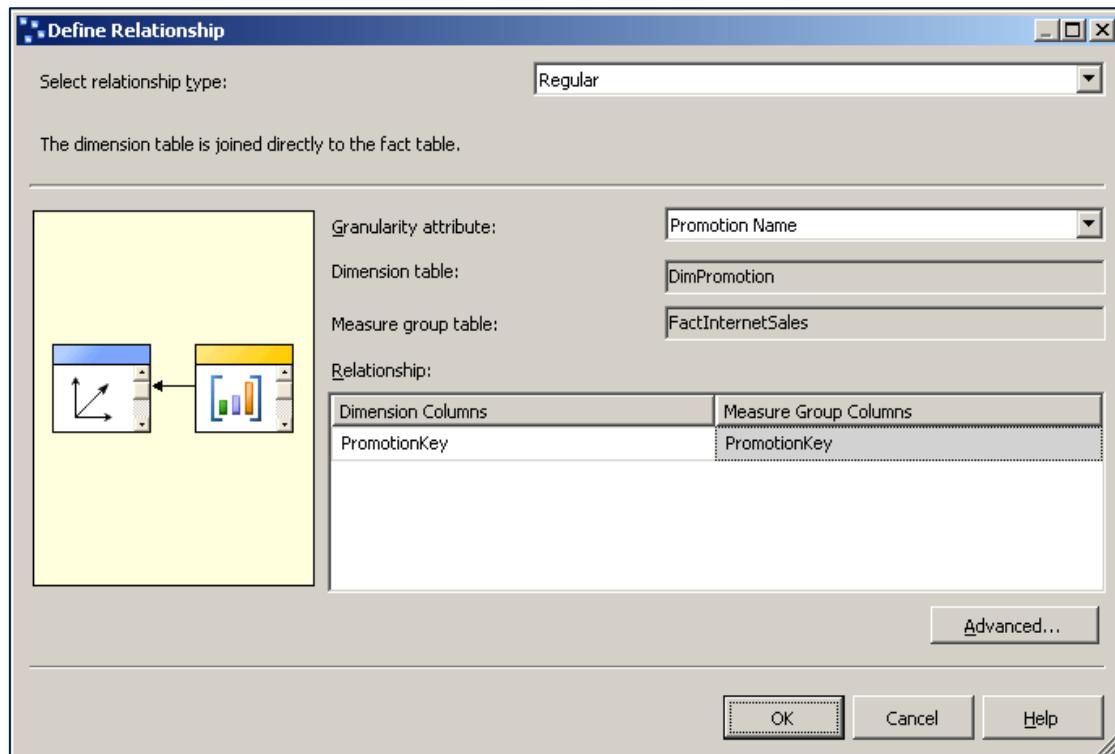
- Relación regular.
- Relación referenciada.
- Relación “fact”.
- Relación many – to – many.

El tipo de relación entre las dimensiones y los measure groups se determina a través del tab Dimension Usage en el diseñador de cubos.

Relación regular

Las relaciones regulares constituyen el tipo más frecuente de relación. En una relación regular, la tabla de dimensión se encuentra directamente relacionada a través de una relación de tipo primary key – foreign key. La siguiente figura muestra una relación de este tipo entre “DimPromotion” y “FactInternetSales”:

La siguiente figura muestra las características de la relación entre la dimensión “Promotion” y el measure group “Internet Sales”:



Un caso especial lo constituye la situación denominada “Role Playing”. En ocasiones, una dimensión participa múltiples veces en el mismo cubo, desempeñando distintos roles. Esto se presenta, por ejemplo, en las siguientes ocasiones:

- Una tabla de hechos puede contener tres columnas que lo enlacen con la dimensión de tiempo: una columna para almacenar la fecha de facturación, otra para almacenar la fecha de remisión, y otra para almacenar la fecha de entrega.
- Una tabla de hechos puede contener dos columnas enlazadas con la dimensión de empleados: una columna para representar al empleado que efectuó la venta, y otra para representar al empleado que efectuó el despacho.

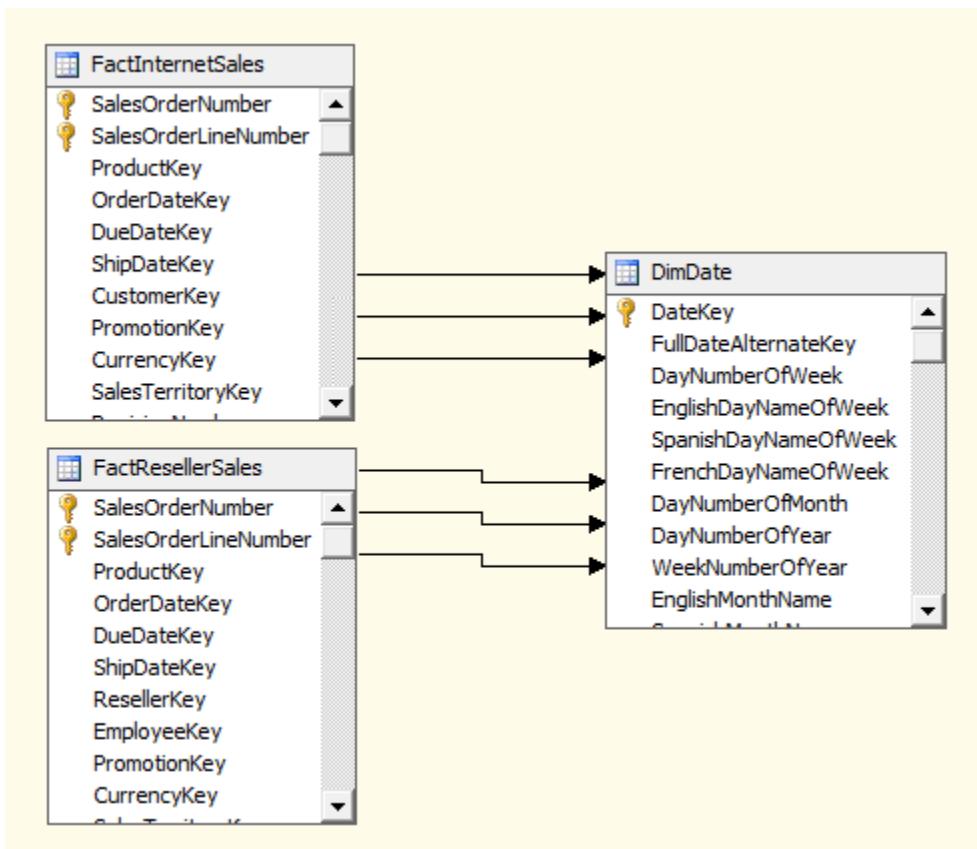
En SQL Server 2000, este tipo de requerimiento sólo podía satisfacerse duplicando las tablas de dimensión, o creando vistas a partir de la tabla de dimensión (en los ejemplos anteriores, la solución hubiera consistido en crear tres tablas de dimensión para el tiempo y dos tablas de dimensión para los empleados; o en crear tres vistas a partir de la tabla de dimensión de tiempo y dos vistas a partir de la tabla de empleados).

En cambio, SSAS 2014 provee soporte natural a las situaciones de “Role Playing”. En SSAS 2014, la misma dimensión puede participar múltiples veces en el mismo cubo. Por ejemplo, la dimensión “Empleado” puede participar dos veces en el cubo “Ventas”, con los nombres “Empleado vendedor” y “Empleado despachador”.

Ejercicio

Visualizar relaciones de tipo Role Playing en el cubo Sales

1. La siguiente figura muestra las relaciones existentes entre la tabla “DimDate” y las tablas “FactInternetSales” y “FactResellerSales”.



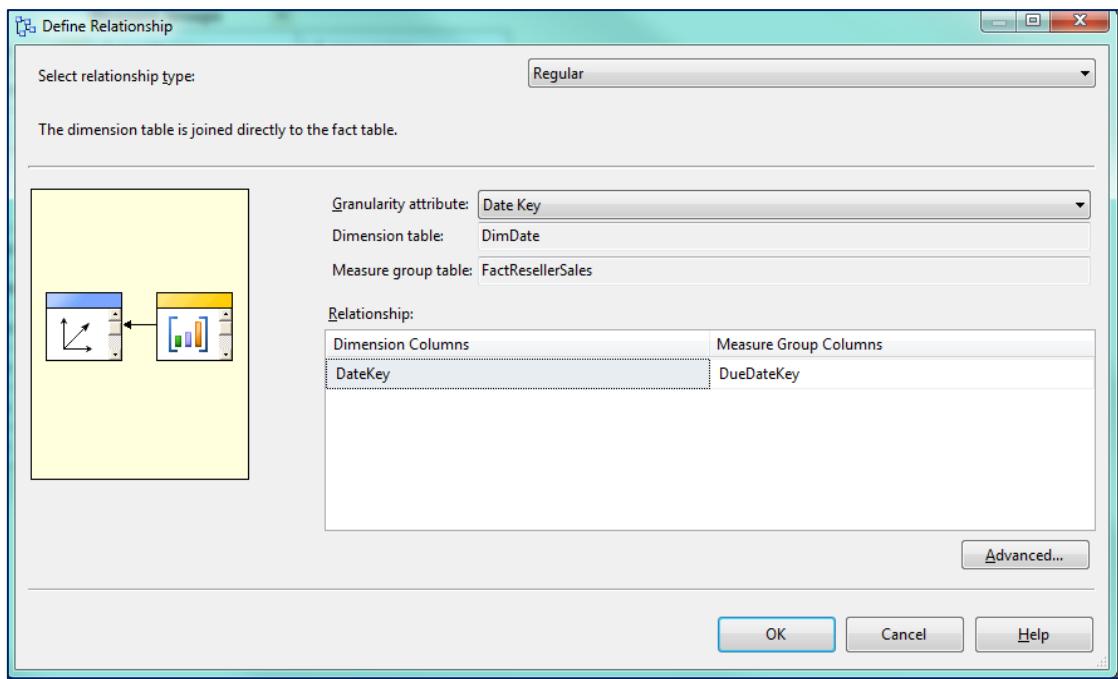
Obsérvese que cada tabla de hechos tiene tres relaciones con la tabla de dimensión “DimDate”. Estas relaciones están definidas en tres columnas distintas de cada tabla de hechos: las columnas “OrderDateKey” (fecha de emisión de orden), “DueDateKey” (fecha esperada) y “ShipDateKey” (fecha de embarque de la orden). Esto significa que la dimensión “Time” juega tres roles distintos con cada tabla de hechos.

2. En el Solution Explorer, haga doble clic sobre el cubo “Sales” para abrir el diseñador de cubos.
3. En la parte superior del diseñador de cubos, seleccione el tab Dimension Usage (es el segundo tab, inmediatamente después del tab Cube Structure). Este tab muestra las relaciones existentes entre las tablas de hechos del cubo y las dimensiones.

The screenshot shows the 'Sales cube [Design]' window. At the top, there's a toolbar with various icons: Cub..., Dim..., Calc..., KPIs, Acti..., Part..., Per..., Tra..., Bro...; followed by a set of icons for search, refresh, and navigation (A, Z, X). Below the toolbar is a menu bar with 'File', 'Edit', 'View', 'Insert', 'Format', 'Tools', 'Help'. The main area is titled 'Measure Groups' with a dropdown arrow. It contains two sections: 'Dimensions' (with a dropdown arrow) and 'Measure Groups' (with a dropdown arrow). Under 'Dimensions', there are ten rows corresponding to Sales Territory, Currency, Time (Due Date), Product, Time (Ship Date), Promotion, Reseller, Time (Order Date), Employee, and Customer. Each dimension row has two columns: the first column lists the dimension attribute (e.g., Sales Territory Region, Currency Name, TimeKey, Product Name, TimeKey, Promotion Name, Reseller Name, TimeKey, Full Name, Customer Name), and the second column shows the measure group it is associated with (e.g., Reseller Sales, Internet Sales). A small minus sign icon is located between the two columns in the first dimension row.

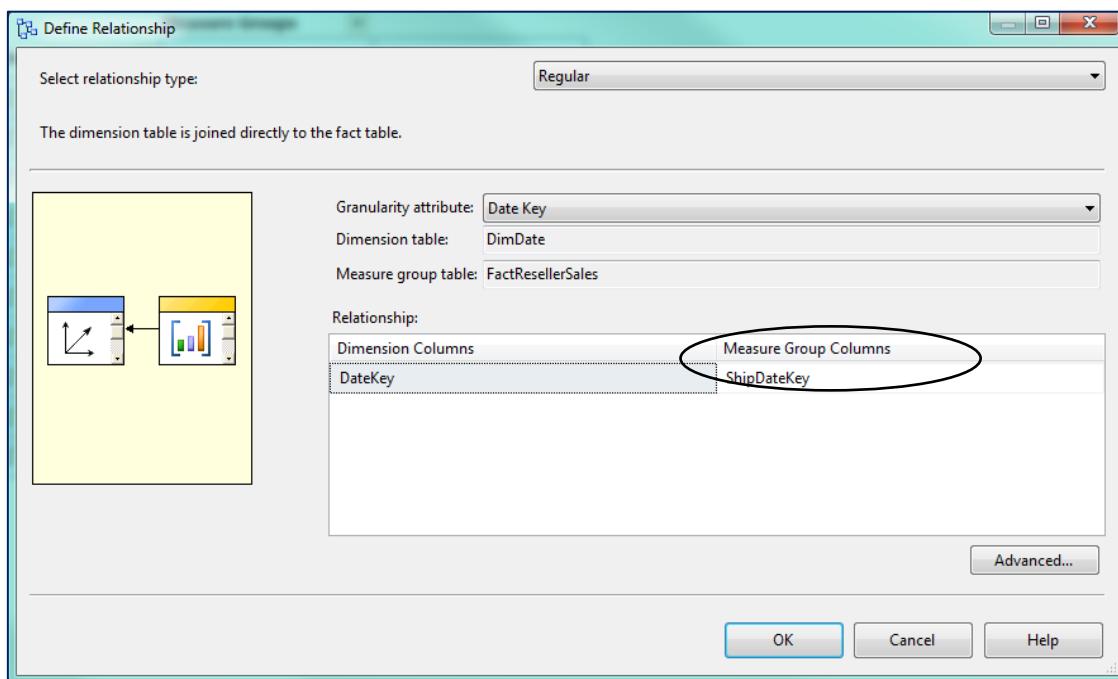
Obsérvese que cada measure group está relacionado con cada dimensión a través de un atributo que los “une”. Por ejemplo, la dimensión “Sales Territory” y el measure group “Reseller Sales” están relacionados por el atributo “Sales Territory Region”.

4. Note la presencia de las tres relaciones con la dimensión “Time”: “Time (Due Date)”, “Time (Ship Date)” y “Time (Order Date)”. Seleccione la intersección entre “Time (Due Date)” y “Reseller Sales”. Pulse el botón . Aparecerá una ventana que muestra los detalles de la relación existente entre la dimensión “Time” y el measure group “Reseller Sales”:



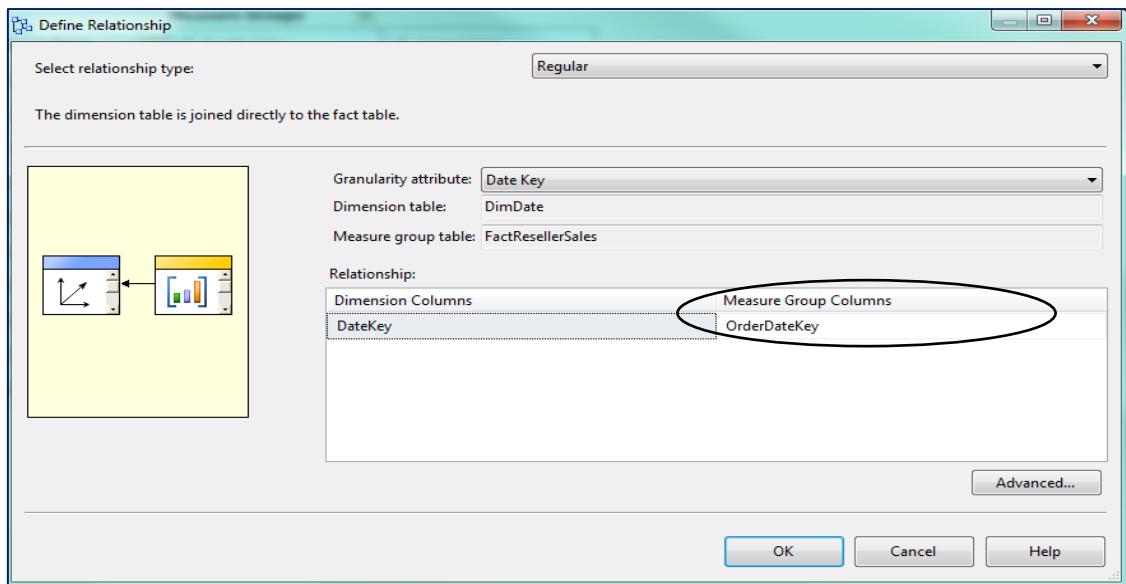
Obsérvese que, en este caso, la relación entre la dimensión “Time” y el measure group “Reseller Sales” se produce a través de la columna “DueDateKey”, de la tabla de hechos “FactResellerSales”. Pulse el botón Cancel.

5. Seleccione la intersección entre “Time (Ship Date)” y “Reseller Sales”. Pulse el botón .



Obsérvese que, en este caso, la relación entre la dimensión “Time” y el measure group “Reseller Sales” se produce a través de la columna “ShipDateKey”, de la tabla de hechos “FactResellerSales”. Pulse el botón Cancel.

6. Seleccione la intersección entre “Time (Order Date)” y “Reseller Sales”. Pulse el botón .



Obsérvese que, en este caso, la relación entre la dimensión “Time” y el measure group “Reseller Sales” se produce a través de la columna “OrderDateKey”, de la tabla de hechos “FactResellerSales”. Pulse el botón Cancel.

El asistente de creación de cubos generó automáticamente estas relaciones debido a que, en el origen de datos, existen tres relaciones de foreign key entre la tabla “DimDate” y la tabla de hechos “FactResellerSales”: una relación a través de la columna “DueDateKey”, otra a través de la columna “ShipDateKey”, y otra a través de la columna “OrderDateKey”.

7. Repita los pasos anteriores para visualizar las relaciones existentes entre la dimensión “Time” y el measure group “Internet Sales”.

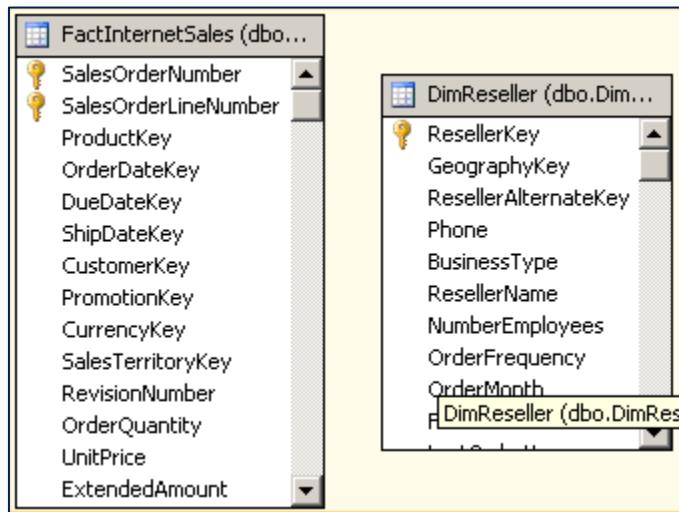
Relación referenciada (referenced relationship)

Una relación referenciada se produce cuando la tabla de dimensión no está directamente relacionada con la tabla de hechos. En esta situación, la relación se produce a través de una tabla de dimensión intermedia, como se muestra en la figura superior.

Este tipo de situaciones corresponde con lo que típicamente se llama “Dimensión Snowflake”. Sin embargo, el enfoque de relación referenciada provee mayor flexibilidad en el diseño de las dimensiones. Por ejemplo, véase el siguiente diseño:

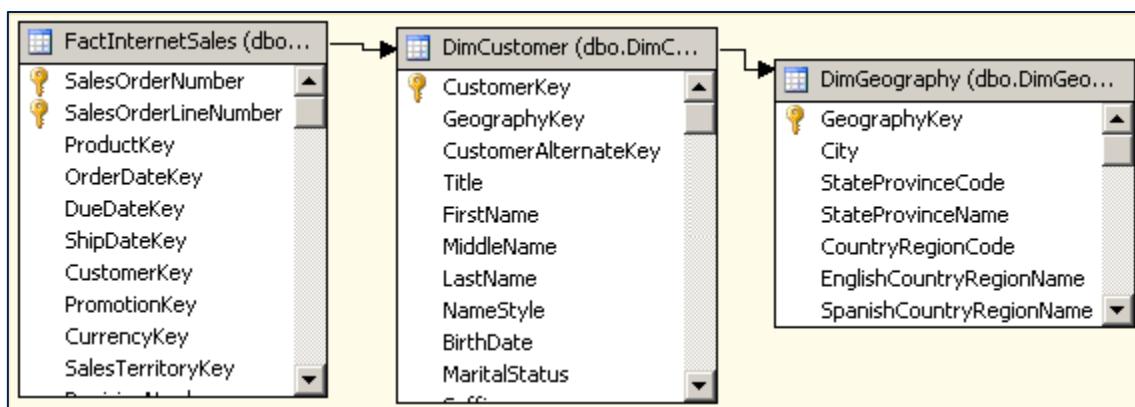


Puede construirse una dimensión de tipo Snowflake con atributos provenientes de las tablas “DimGeography” y “DimReseller”. Esta dimensión podrá conectarse sin problemas con la tabla de hechos “FactResellerSales”. Sin embargo, no podrá relacionarse con la tabla de hechos “FactInternetSales”, pues, como puede verse en la siguiente figura, no existe ninguna relación entre “DimReseller” y “FactInternetSales”:



Una solución alternativa (y más flexible) consiste en crear dos dimensiones distintas: una dimensión “Geography” basada en la tabla “DimGeography”, y una dimensión “Reseller” basada en la tabla “DimReseller”. Luego, puede usarse una relación referenciada para relacionar la dimensión “Geography” con la tabla de hechos “FactResellerSales”.

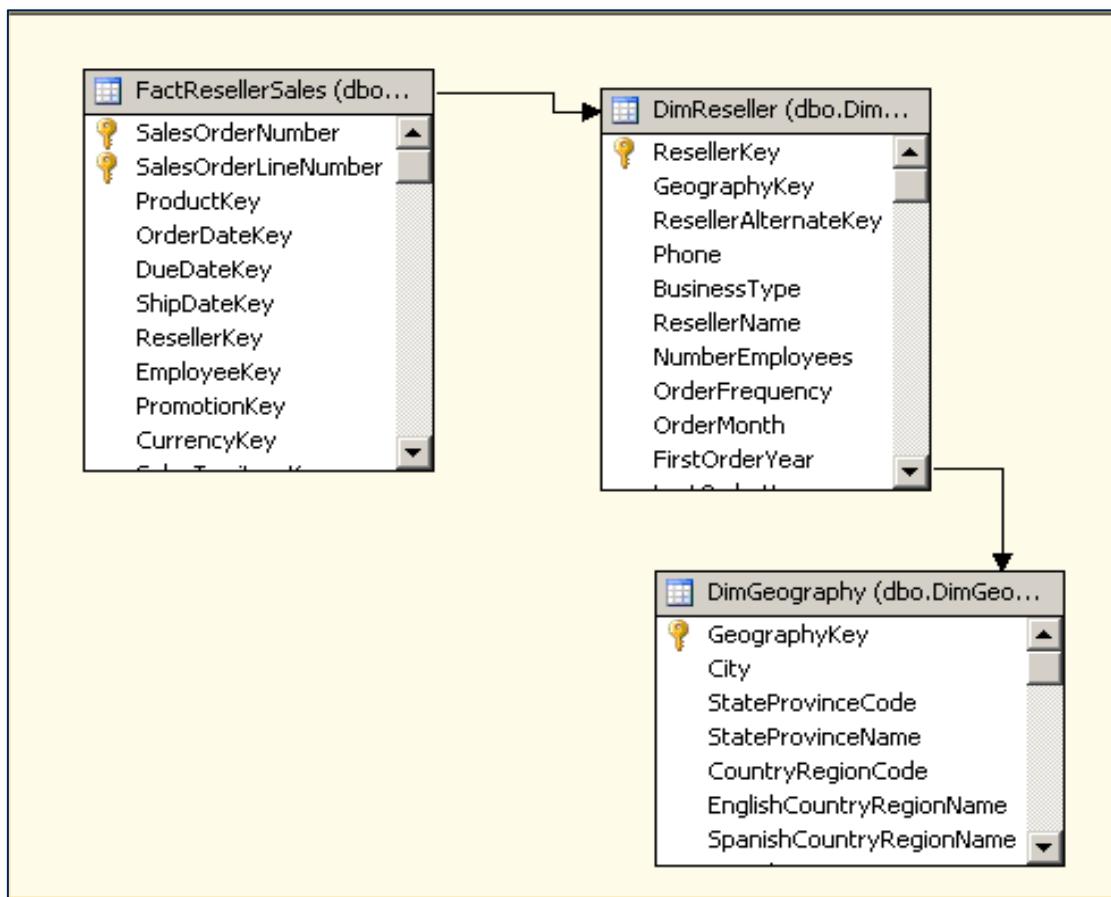
Por otro lado, la siguiente figura, muestra que la tabla “FactInternetSales” también se relaciona indirectamente con “DimGeography” a través de la tabla “DimCustomer”. Por tanto, se puede usar otra relación referenciada para relacionar la dimensión “Geography” con la tabla de hechos “FactInternetSales”, a través de la dimensión “Customer”. Este grado de flexibilidad no hubiese podido alcanzarse si se hubiese implementado la solución a través de dimensiones Snowflake.



Ejercicio

Crear relaciones referenciadas.

1. La siguiente figura muestra la relación existente entre las tablas “DimGeography”, “DimReseller” y “FactResellerSales”.



Obsérvese que no existe una relación directa entre las tablas “DimGeography” y “FactResellerSales”, sino que están relacionadas a través de una dimensión intermedia: la tabla “DimReseller”. La columna que relaciona a “DimReseller” y “DimGeography” es la columna “GeographyKey”.

2. Como primer paso, en el proyecto “AdventureWorksDataMart” cree una nueva dimensión llamada “Geography” con las siguientes especificaciones:

Tabla de dimensión principal: DimGeography

Tablas de dimensión snowflake: ninguna

KeyColumns: GeographyKey

NameColumn: City

Atributo clave: Renombrarlo como “City”.

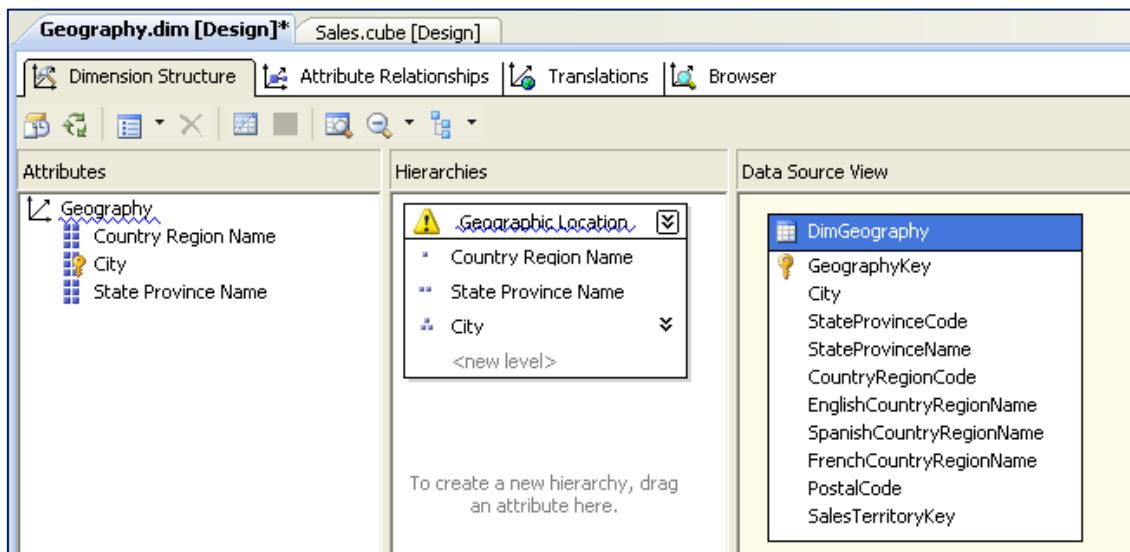
Atributos adicionales

Atributo	Columna origen
State Province Name	DimGeography.StateProvinceName
Country Region Name	DimSalesTerritory.EnglishCountryRegionName

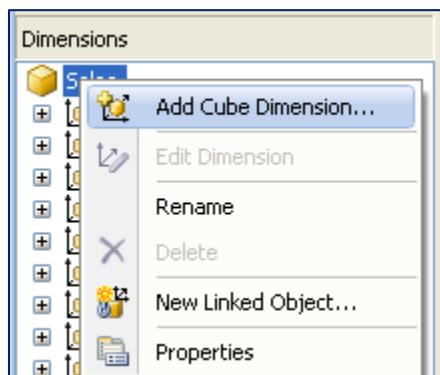
Jerarquías

Jerarquías	Niveles
Geographic Location	<ul style="list-style-type: none"> - Country Region Name -- State Province Name --- City

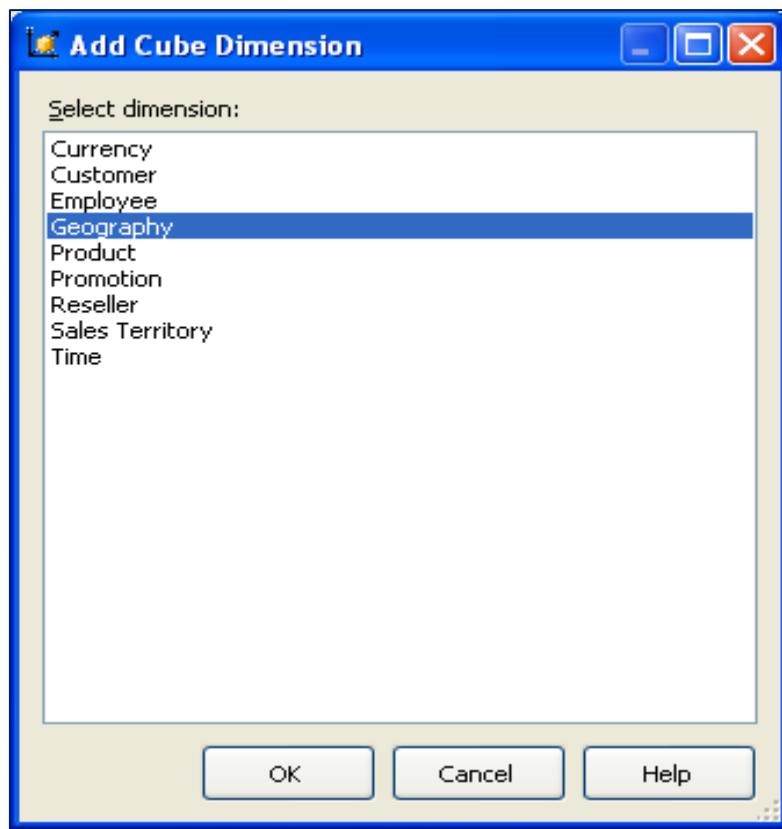
Al finalizar, la dimensión “Geography” debe tener la siguiente estructura en el diseñador de dimensiones.



3. En el Solution Explorer, haga doble clic sobre el cubo “Sales” para abrir el diseñador de cubos.
4. En el panel de dimensiones, haga clic derecho sobre el cubo “Sales”. Seleccione la opción “Add Cube Dimension”.



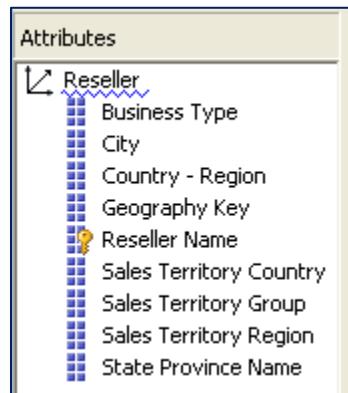
5. Se abrirá una ventana que muestra las dimensiones existentes en el proyecto. Seleccione la dimensión “Geography”.



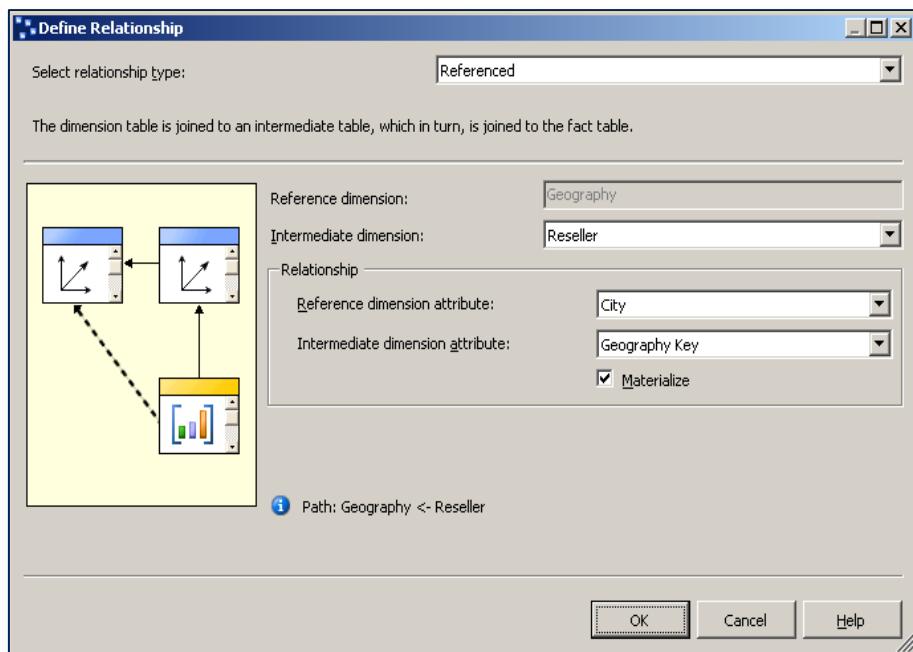
6. Pulse el botón OK para aceptar.
7. En el diseñador de cubo, abra el tab Dimension Usage. Observe que la dimensión "Geography" no tiene relaciones con ninguno de los measure groups.

Dimensions	Measure Groups
Reseller	Reseller Sales
Sales Territory	Sales Territory Region
Currency	Currency Name
Product	Product Name
Reseller	Reseller Name
Time (Ship Date)	Date Key
Time (Order Date)	Date Key
Time (Due Date)	Date Key
Promotion	Promotion Name
Employee	Full Name
Customer	Customer Name
Geography	

8. En el Solution Explorer, haga doble clic sobre la dimensión “Reseller”. En el panel de origen de datos, seleccione la columna “GeographyKey” de la tabla “DimReseller” y arrástrela hacia el panel de atributos, bajo la dimensión “Reseller”. Esto creará un nuevo atributo llamado “Geography Key” en la dimensión “Reseller”. Dicho atributo servirá para efectuar el enlace con la dimensión “Geography”. Una vez creado, establezca la propiedad AttributeHierarchyVisible del atributo “Geography Key” a False.

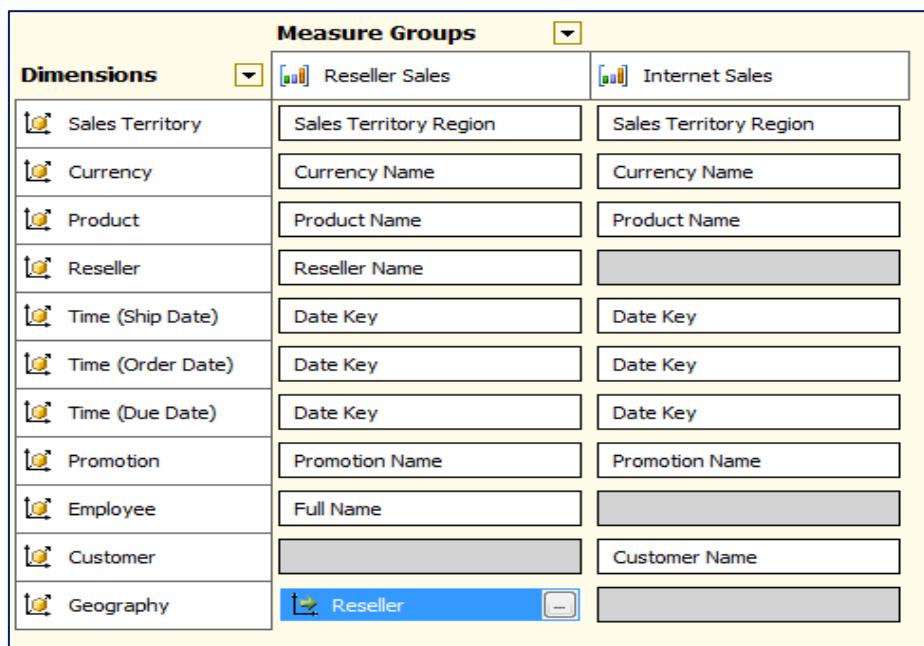


9. En el Solution Explorer, haga doble clic sobre la dimensión “Geography”. En el panel de atributos, seleccione el atributo “City”. Nótese que “City” es el atributo clave de la dimensión. Observe el valor de su propiedad KeyColumns en la ventana de propiedades: la clave de este atributo es la columna “GeographyKey” de la tabla “DimGeography”.
10. En el Solution Explorer, haga doble clic sobre el cubo “Sales”. En el tab Dimension Usage, seleccione la intersección entre la dimensión “Geography” y el measure group “Reseller Sales”. Pulse el botón . En la ventana de definición de relaciones, defina el tipo de relación a Referenced, luego establezca la propiedad Intermediate dimension a “Reseller”. En la propiedad Reference dimension attribute, seleccione el atributo “City”; y en la propiedad Intermediate dimension attribute seleccione el atributo “Geography Key” que se muestran en la figura siguiente:

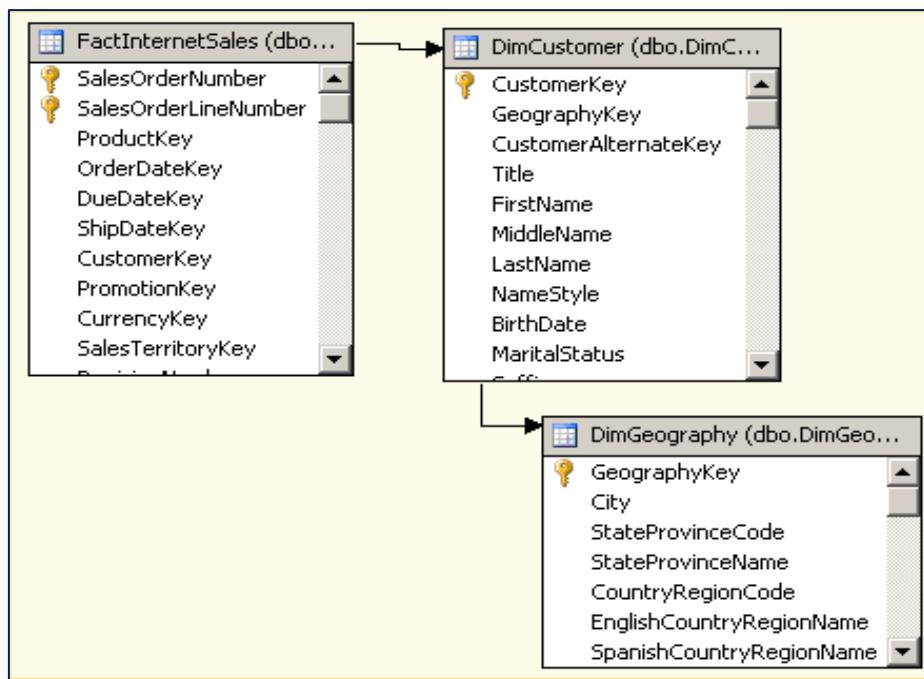


Observe que la relación ha sido definida entre el atributo “City” (dimensión “Geography”) y el atributo “Geography Key” (dimensión “Reseller”). Esto se debe a que la columna de clave para “City” es “DimGeography.GeographyKey”, y la columna de clave para “Geography Key” es “DimReseller.GeographyKey”. Por lo tanto, ambos atributos pueden usarse para relacionar las dimensiones “Geography” y “Reseller”.

11. Pulse el botón OK para aceptar. Observe que se ha creado una relación entre la dimensión “Geography” y la tabla de hechos “Reseller Sales”, representada por el símbolo



12. La siguiente figura muestra las relaciones existentes entre las tablas “DimGeography”, “DimCustomer” y “FactInternetSales”:



13. Con la guía de los pasos anteriormente efectuados, relacione la dimensión “Geography” con el measure group “Fact Internet Sales”.
14. Grabe los cambios y efectúe el deployment del proyecto. Al finalizar, consulte la información del cubo “Sales”.

Relación “Fact”

Frecuentemente, las tablas de hechos contienen, además de columnas de medidas y foreign keys, columnas adicionales que almacenan información relevante para el negocio. Por ejemplo, una tabla de hechos con información de ventas puede contener columnas que almacenen el número de factura y la dirección de envío del despacho.

En estas situaciones, puede resultar muy útil definir una dimensión a partir de la tabla de hechos. A este tipo de dimensiones se les denomina “dimensiones degeneradas”.

La siguiente actividad mostrará la creación de dos dimensiones nuevas, a partir de las de las tablas “FactInternetSales” y “FactResellerSales”, que contienen columnas que almacenan el número de la orden de venta.

Ejercicio:

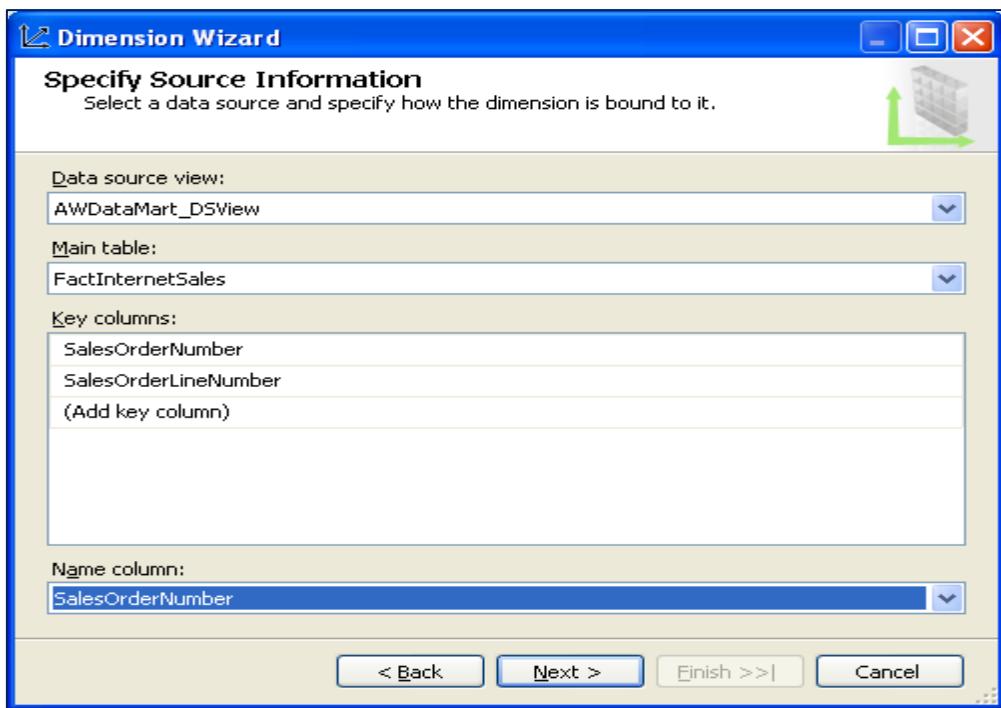
Crear una relación del tipo “FACT” en el cubo “SALES”

1. La siguiente figura muestra la estructura de la tabla “FactInternetSales”.

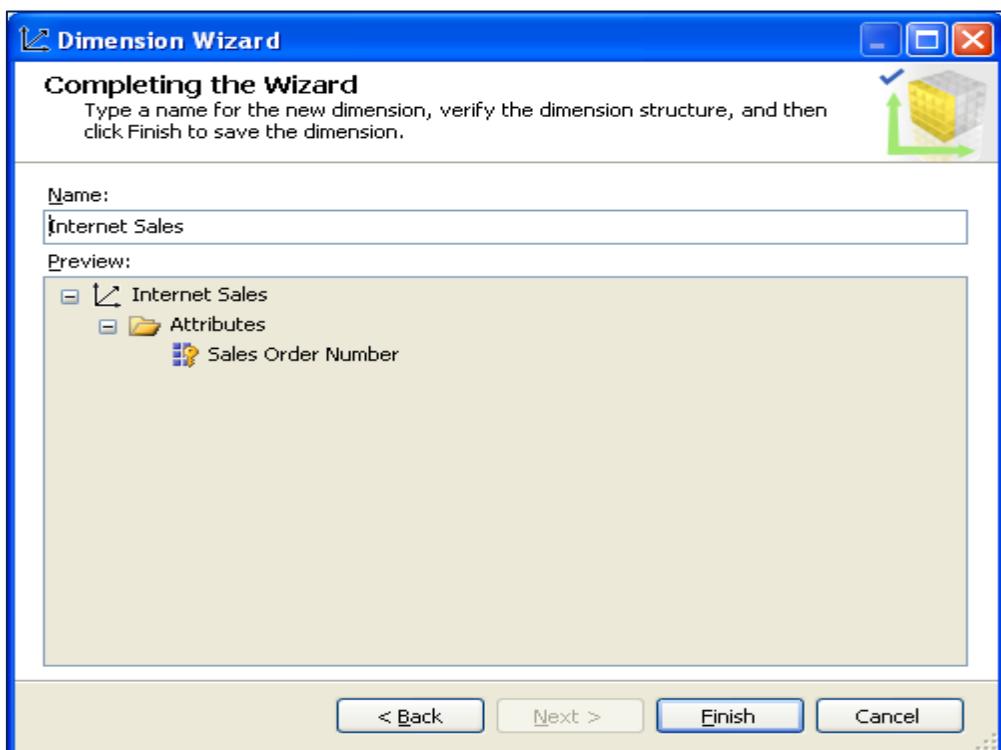
SalesOrderNumber
SalesOrderLineNumber
ProductKey
OrderDateKey
DueDateKey
ShipDateKey
CustomerKey
PromotionKey
CurrencyKey
SalesTerritoryKey
RevisionNumber
OrderQuantity
UnitPrice
ExtendedAmount
UnitPriceDiscountPct
DiscountAmount
ProductStandardCost
TotalProductCost
SalesAmount
TaxAmt

Observe que la clave primaria está formada por las columnas “SalesOrderNumber” y “SalesOrderLineNumber”. La columna “SalesOrderNumber” almacena el número del documento de orden de venta. Este es un caso de “dimensión degenerada”; es decir, en la tabla de hechos existe información que no corresponde con medidas ni tampoco con claves foráneas provenientes de tablas de dimensión.

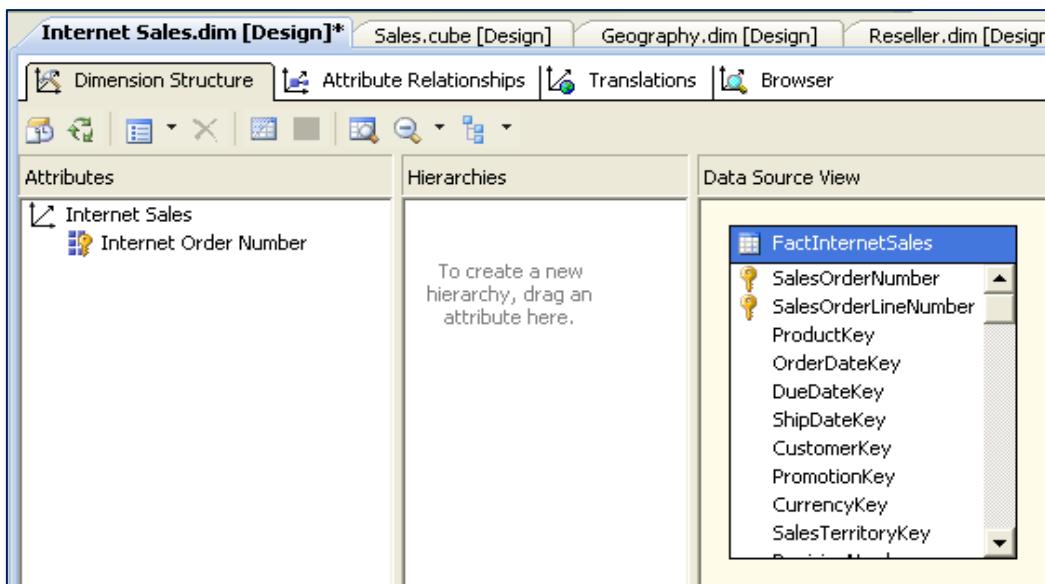
2. En el proyecto “AdventureWorksDataMart”, inicie la creación de una nueva dimensión. En la ventana de selección del método de creación de la dimensión, asegúrese de que el checkbox Use an existing table esté marcado.
3. Continúe con el asistente. En la ventana de selección de la tabla de dimensión, seleccione la tabla “FactInternetSales”. En la propiedad KeyColumns, seleccione las columnas “SalesOrderNumber” y “SalesOrderLineNumber”. En la propiedad Name Column, seleccione la columna “SalesOrderNumber”.



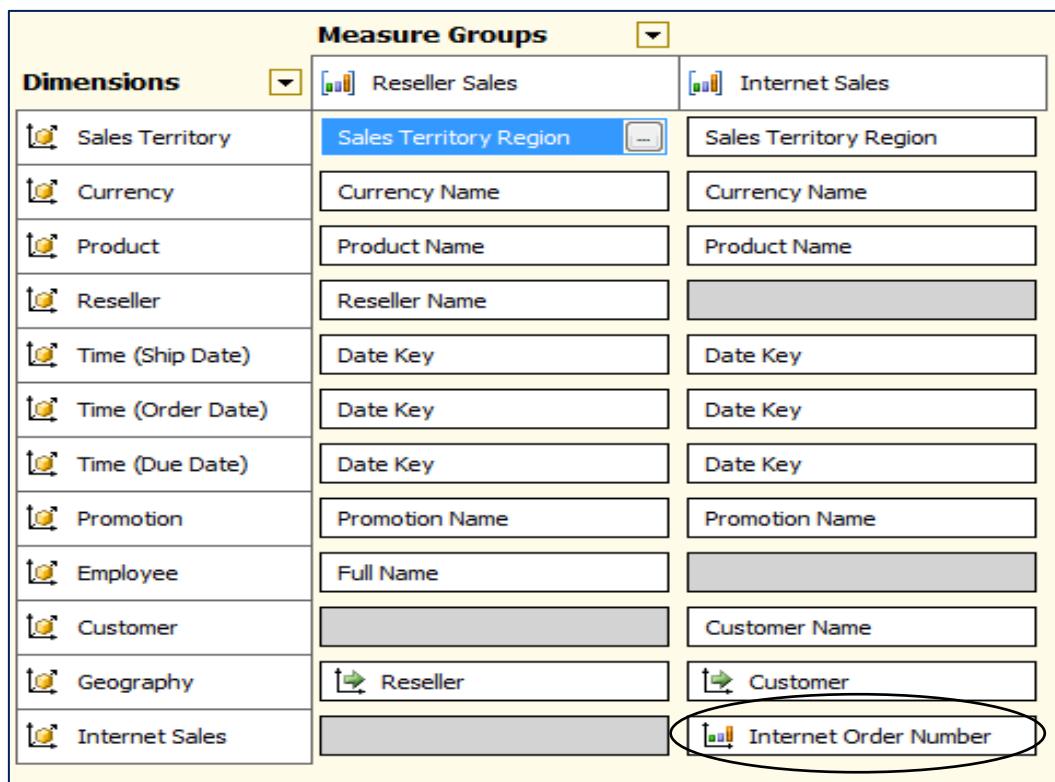
- Continúe con el asistente, pulsando el botón Next. Deseleccione las tablas relacionadas de la pantalla siguiente y no agregue ningún atributo adicional. Al finalizar, establezca el nombre de la nueva dimensión a “Internet Sales”.



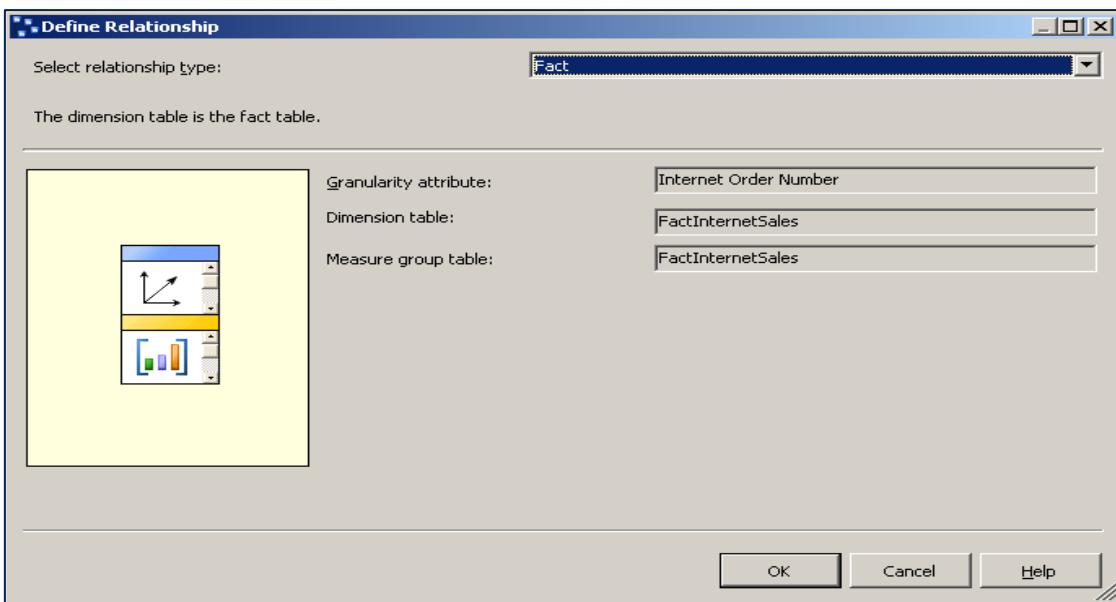
- En el Solution Explorer, abra la nueva dimensión “Internet Sales”. En el panel de atributos, seleccione el atributo “Sales Order Number” y, en la ventana de propiedades, cambie su nombre a “Internet Order Number”. Al finalizar, la estructura de la dimensión debe quedar tal como se muestra a continuación:



6. En el Solution Explorer, haga doble clic sobre el cubo "Sales". En el panel de jerarquías, seleccione el cubo "Sales", haga clic derecho y seleccione la opción Add Cube Dimension. Seleccione la dimensión "Internet Sales".
7. Seleccione el tab Dimension Usage. Observe que se ha creado automáticamente una relación entre la dimensión "Internet Sales" y el measure group "Internet Sales", con el símbolo



8. Seleccione la intersección entre la dimensión "Internet Sales" y el measure group "Internet Sales". Pulse el botón



Observe que se ha generado de manera automática una relación de tipo Fact. Pulse el botón Cancel.

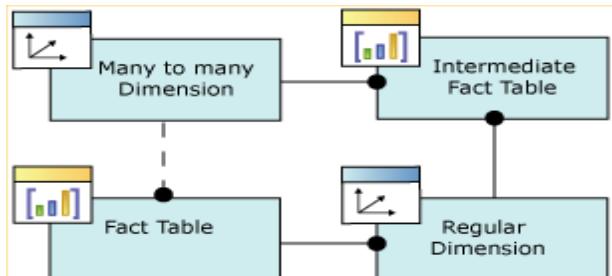
Relación muchos a muchos (many – to – many)

En las relaciones estudiadas hasta este momento, se asume que una dimensión tiene una relación uno – muchos con las tablas de hechos. Por ejemplo, un cliente (un registro en “DimCustomer”) puede tener muchas ventas (varios registros en “FactResellerSales”); pero cada venta corresponde con un solo cliente.

Sin embargo, existen situaciones en las cuales las dimensiones tienen relaciones de tipo muchos – muchos con las tablas de dimensión. Por ejemplo, véase la imagen superior.

La tabla “DimSalesReason” almacena motivos de venta. Como puede deducirse de la figura, la relación entre “DimSalesReason” y “FactInternetSales” es de tipo muchos – muchos: cada venta (un registro en “FactInternetSales”) puede tener múltiples motivos de venta (registros en “DimSalesReason”). Y, análogamente, cada motivo de venta puede estar relacionado con múltiples ventas. En este diseño, la relación muchos – muchos se implementa a través de la tabla “FactInternetSalesReason”, que relaciona los registros de “FactInternetSales” con los registros de “DimSalesReason”.

SSAS 2014 provee un tipo especial de relación que permite implementar relaciones de muchos a muchos entre las tablas de hechos y las tablas de dimensión. Véase la siguiente figura:



En general, la implementación de una relación muchos – muchos requiere de los siguientes componentes representados en la figura anterior:

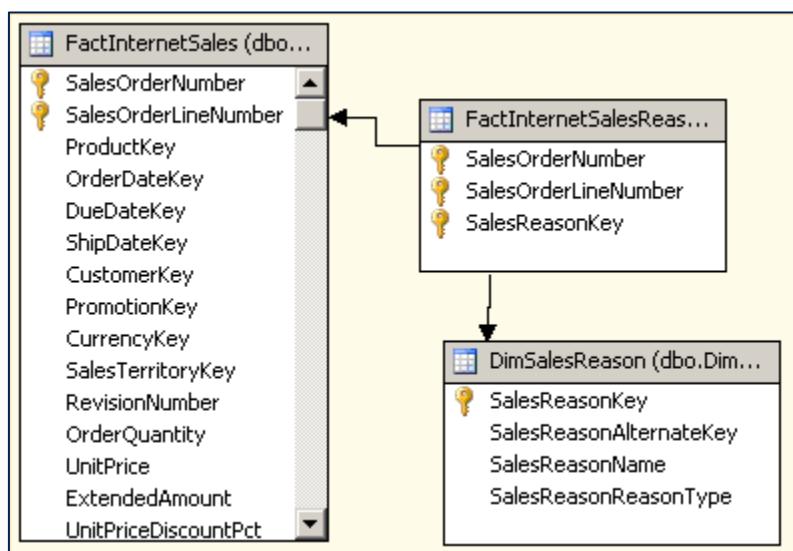
- Fact Table: La tabla de hechos principal.
- Many to many dimension: La tabla de dimensión con la cual se desea relacionar la tabla de hechos principal.
- Regular Dimension: Una dimensión relacionada con la tabla de hechos principal a través de una relación regular.
- Intermediate Fact Table: Una tabla de hechos relacionada con las dos dimensiones mencionadas más arriba.

Las tablas Fact Table, Regular Dimension e Intermediate Fact Table deben estar relacionadas en el data source view. De lo contrario, no se podrá definir la relación muchos – muchos.

Ejercicio:

Crear una relación “many to many” en el cubo “Sales”.

1. En el proyecto “AdventureWorksDataMart”, en el Solution Explorer, haga doble clic en el data source view “AWDataMart_DSVView”.
2. Haga un clic derecho sobre cualquier lugar del diseñador de data source views. Seleccione la opción Add/Remove Tables. Añada al data source view las tablas “FactInternetSalesReason” y “DimSalesReason”. Al finalizar, grabe los cambios.
3. La siguiente figura muestra las relaciones existentes entre las tablas “FactInternetSales”, “FactInternetSalesReason” y “DimReason”:



La tabla “FactInternetSalesReason” enlaza a la tablas “FactInternetSales” y “DimSalesReason” en una relación muchos – muchos. Por cada registro en “DimSalesReason” existen muchos registros en la tabla “FactInternetSales” (es decir, por cada motivo de venta existen muchas ventas), y por cada registro en “FactInternetSales” existen muchos registros en “DimSalesReason” (es decir, por cada venta pueden existir varios motivos de venta).

4. A continuación, en el proyecto “AdventureWorksDataMart”, cree una nueva dimensión llamada “Sales Reason” con las siguientes especificaciones:

Tabla de dimensión principal: DimSalesReason

Tablas de dimensión snowflake: ninguna

KeyColumns: SalesReasonKey

NameColumn: SalesReasonName
 Atributo clave: Renombrarlo como “Reason”.

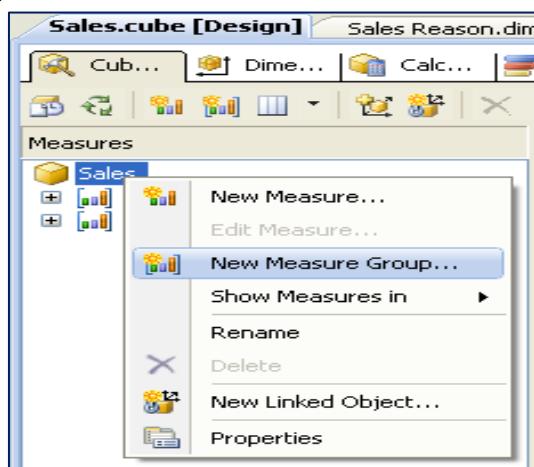
Atributos adicionales

Atributo	Columna origen
Type	DimSalesReason.SalesReasonReasonType

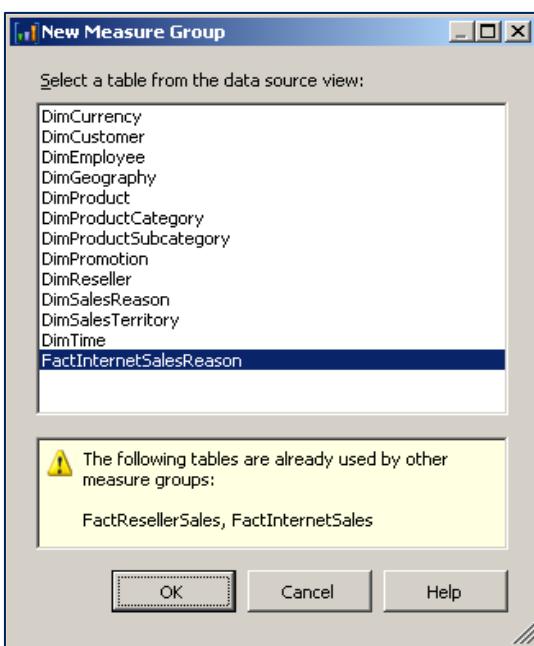
Jerarquías

Jerarquías	Niveles
Sales Reasons	<ul style="list-style-type: none"> ▪ Type ▪▪ Reason

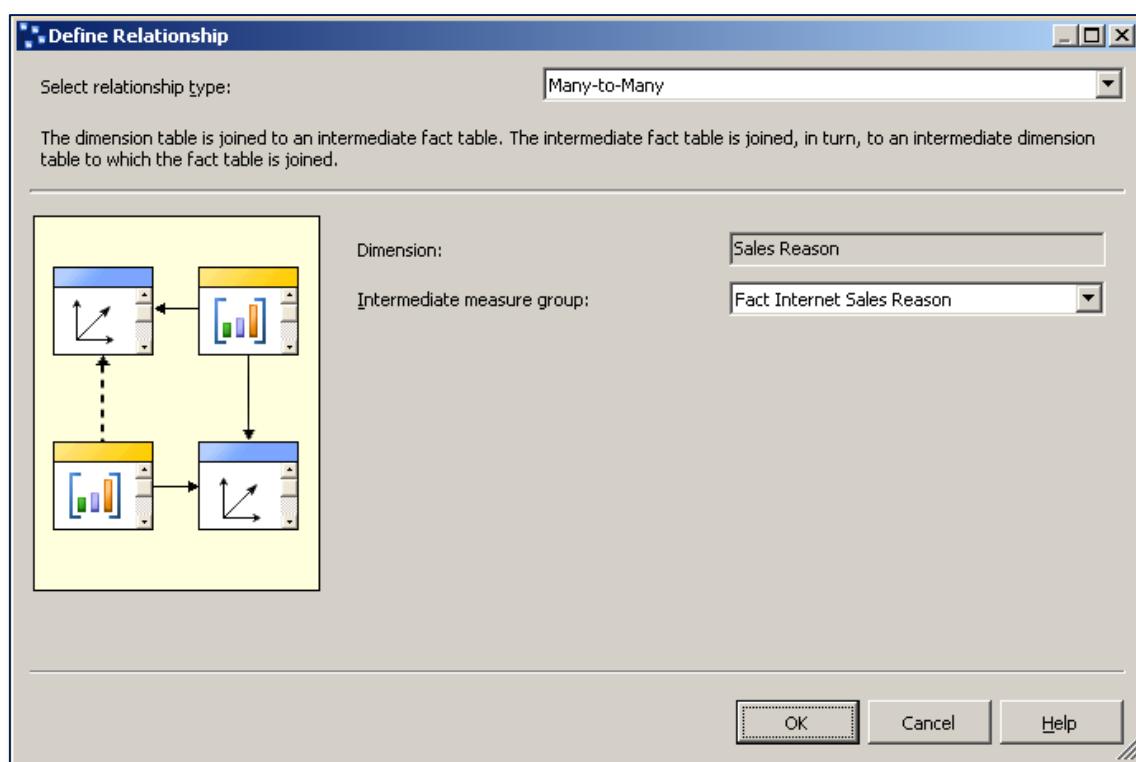
- En el Solution Explorer, haga doble clic sobre el cubo “Sales”. En el panel de medidas, haga clic derecho sobre el cubo “Sales” y seleccione la opción New Measure Group.



- Se abrirá una ventana que permite seleccionar la tabla sobre la cual se construirá el nuevo measure group. Seleccione la tabla “FactInternetSalesReason”.



7. Pulse el botón OK para aceptar. Este measure group tendrá como único objetivo relacionar al measure group “Internet Sales” (basado en la tabla “FactInternetSales”) con la dimensión “Sales Reason” (basado en la tabla “DimSalesReason”).
8. En el panel de medidas, seleccione el measure group “Fact Internet Sales Reason”. Seleccione la medida “Fact Internet Sales Reason Count” y establezca, en la ventana de propiedades, su propiedad Visible a False.
9. En el panel de Dimensions, haga clic derecho sobre el cubo “Sales” y seleccione la opción Add Cube Dimension. Seleccione la dimensión “Sales Reason”.
10. Seleccione el tab Dimension Usage. Obsérvese que se ha creado automáticamente una relación regular entre la dimensión “Sales Reason” y el measure group “Fact Internet Sales Reason”. Sin embargo, no hay relación entre la dimensión “Sales Reason” y el measure group “Internet Sales”.
11. Seleccione la intersección entre la dimensión “Sales Reason” y el measure group “Internet Sales”. Pulse el botón . Se abrirá la ventana de definición de relaciones. En la propiedad Select relationship type seleccione el tipo “Many – to – Many”. En la propiedad Intermediate measure group, seleccione el measure group “FactInternetSalesReason”.



12. Pulse el botón OK. Observe que se ha creado una relación de tipo many – to – many entre la dimensión “Sales Reason” y el measure group “Internet Sales”, representada por el símbolo :

The screenshot shows the 'Measure Groups' section of the Dimension Designer. It lists various dimensions on the left and their corresponding members in three measure groups: Reseller Sales, Internet Sales, and Fact Internet Sales. A circled area highlights the 'Sales Reason' dimension, which is mapped to the 'Fact Internet Sales Reason' measure group.

Dimensions	Measure Groups		
	Reseller Sales	Internet Sales	Fact Internet Sales Re...
Employee	Full Name		
Time (Order Date)	Time Key	Time Key	
Reseller	Reseller Name		
Promotion	Promotion Name	Promotion Name	
Time (Ship Date)	Time Key	Time Key	
Product	Product Name	Product Name	
Time (Due Date)	Time Key	Time Key	
Currency	Currency Name	Currency Name	
Sales Territory	Sales Territory Region	Sales Territory Region	
Customer		Customer Name	
Geography	Reseller	Customer	
Internet Sales		Internet Order Number	Internet Order Number
Reseller Sales	Reseller Order Number		
Sales Reason		Fact Internet Sales Re...	Reason

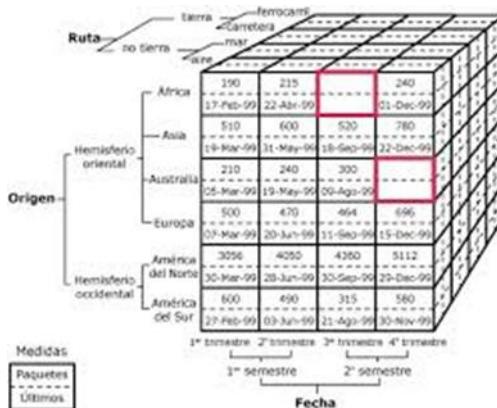
13. Grabe los cambios y efectúe el deployment del proyecto. Consulte la información del cubo, cruzando medidas del measure group “Internet Sales” con la dimensión “Sales Reason”, y la dimensión “Order Date”.

2.3. DEFINICIÓN DE MIEMBROS CALCULADOS Y KPI

2.3.1. Creando Miembros Calculado

Analysis Services provee soporte para definir miembros calculados (calculated members) en tiempo de diseño. Estos miembros son persistentes; es decir, se almacenan en la base de datos de análisis.

Adicionalmente, Analysis Services permite alcanzar un grado mayor de flexibilidad a través de las celdas calculadas (calculated cells). Las celdas calculadas consisten en expresiones MDX que se aplican únicamente a un slicer específico del cubo. De esta manera, una determinada medida puede tener distintas funciones de cálculo, dependiendo de las dimensiones, niveles y miembros que participan en una consulta.



Una característica nueva de SSAS 2012 es la implementación de Key Performance Indicators (KPIs). A través de los KPIs, una empresa puede comparar las métricas principales del negocio con las metas establecidas y evaluar de esta manera el éxito de la organización.

Miembros calculados (calculated members)

Un miembro calculado es un miembro que se calcula en tiempo de ejecución a través de una sentencia MDX. El servidor de análisis sólo almacena la definición de los miembros calculados; sus valores son calculados en tiempo de ejecución durante las consultas de datos.

SSAS provee un editor para diseñar miembros calculados. Durante su creación, se debe especificar la dimensión a la cual pertenecerá el nuevo miembro y la expresión MDX que se utilizará para calcularlo.

Name:
[Valor Descontado]

Parent Properties

Parent hierarchy: Measures
Parent member:

Expression

```
Measures.[Sales Amount] - Measures.[Discount Amount]
```

Additional Properties

Format string:

Visible: True
Non-empty behavior:

Associated measure group: (Undefined)
Display folder:

Color Expressions
Font Expressions

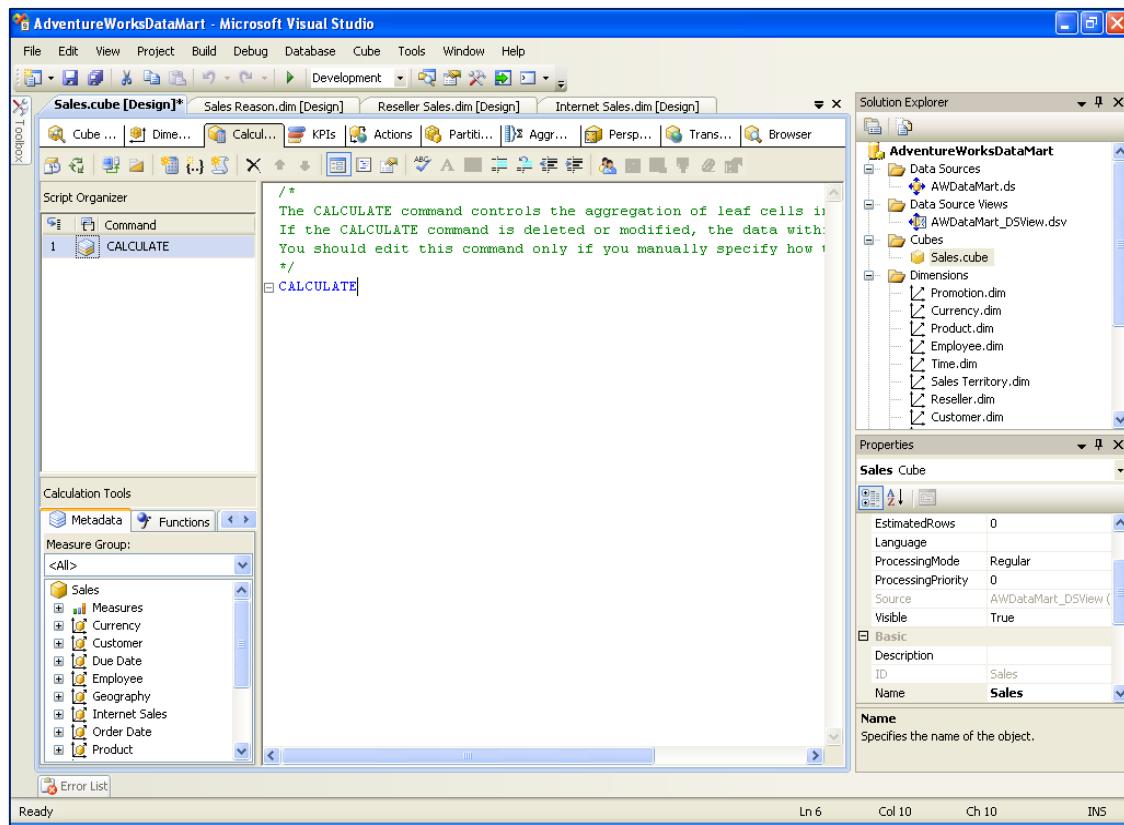
Los miembros calculados son muy útiles para definir nuevas medidas con funciones personalizadas de cálculo. En estos casos, los miembros calculados deben asignarse a la dimensión Measures. Por ejemplo, puede utilizarse la siguiente expresión MDX en la definición del miembro calculado Valor Descontado en la dimensión Measures.

Measures.[Sales Amount] – Measures.[Discount Amount]

Los miembros calculados proveen gran flexibilidad para implementar lógica personalizada de cálculo; sin embargo, penalizan la performance de las consultas, debido a que son calculados por el servidor en tiempo de ejecución. Es una práctica recomendable evitar la complejidad en la escritura de miembros calculados, y tener en cuenta la cantidad de celdas que serán procesadas durante la ejecución del cálculo.

Ejercicios

1. Con el Sql Server Data Tools, abra el proyecto “AdventureWorksDataMart”. En el Solution Explorer, haga doble clic en el cubo “Sales”.
2. En el diseñador de cubos, seleccione el tab Calculations (es el tercero empezando por la izquierda). Los miembros calculados se crean en este tab.



3. En el toolbar del tab Calculations, pulse el botón para iniciar la creación de un nuevo miembro calculado. Se abrirá un editor que permitirá efectuar la definición de la fórmula de cálculo y las propiedades del nuevo miembro. En la propiedad Name, escriba: “[Sales Last Period]”. En la propiedad Parent Hierarchy, seleccione “MEASURES”, y en propiedad Expression escriba la siguiente expresión MDX:

```
([Measures].[Sales Amount],  
parallelperiod([Order Date].[Calendar].currentmember.level, 1,  
[Order Date].[Calendar].currentmember))
```

Name: [Sales Last Period]

Parent Properties:

- Parent hierarchy: Measures
- Parent member: [Change]

Expression:

```
([Measures].[Sales Amount],  
parallelperiod([Order Date].[Calendar].currentmember.level,  
1,[Order Date].[Calendar].currentmember))
```

Additional Properties:

- Format string: [dropdown]
- Visible: True
- Non-empty behavior: [dropdown]
- Associated measure group: (Undefined)
- Display folder: [dropdown]

Color Expressions: [dropdown]

Font Expressions: [dropdown]

4. Los miembros calculados pueden utilizarse para definir otros miembros calculados. Repita el paso (3) para crear un nuevo miembro calculado llamado “[Sales Growth]” en la jerarquía “MEASURES” con la siguiente expresión:

```
CASE  
WHEN ISEMPTY(Measures.[Sales Last Period]) THEN 1  
ELSE (Measures.[Sales Amount] - Measures.[Sales Last Period])/  
Measures.[Sales Last Period]  
END
```

En la propiedad Format String, seleccione “Percent”.

Name: [Sales Growth]

Parent Properties:

Parent hierarchy: Measures

Parent member: Change

Expression:

```
CASE
    WHEN ISEMPTY(Measures.[Sales Last Period]) THEN 1
    ELSE
        (Measures.[Sales Amount] - Measures.[Sales Last Period]) /
        Measures.[Sales Last Period]
    END
```

Additional Properties:

- Format string: "Percent"
- Visible: True
- Non-empty behavior:
- Associated measure group: (Undefined)
- Display folder: |

Color Expressions

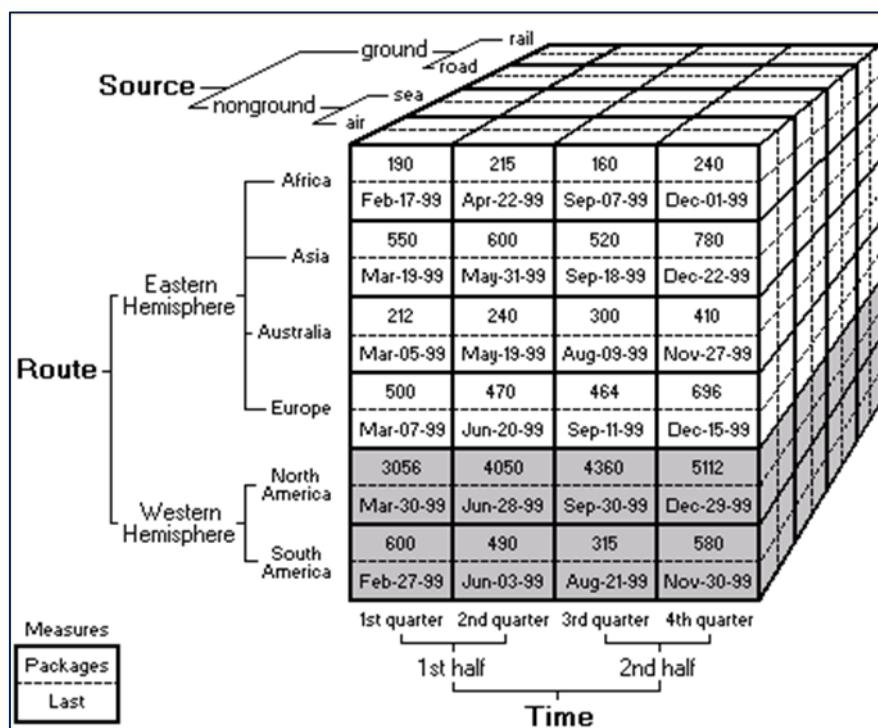
Font Expressions

5. Guarde los cambios y efectúe un deployment del proyecto. Consulte las medidas “Sales Amount”, “Sales Last Period” y “Sales Growth” del cubo “Sales” en el tab Browser colocando la jerarquía “Order Date.Calendar” en las columnas y el nivel “Product Category Name” de la jerarquía “Product.Product Categories” en las filas. Observe que el valor de “Sales Last Period” para el año 2002 coincide con el valor de “Sales Amount” para el año 2001.

Drop Filter Fields Here								
Product Category Name	Year		2002					
	2001	2002	Sales Amount	Sales Last Period	Sales Growth	Sales Amount	Sales Last Period	Sales Growth
Accessories	15087.8071	15087.8071	100.00%	65107.014600000	15087.8071	331.52%		
Bikes	6024627.35299994	6024627.35299994	100.00%	15309837.3263	6024627.35299994	154.12%		
Clothing	26463.0040999999	26463.0040999999	100.00%	338878.99800000	26463.0040999999	1180.58%		
Components	485897.684799999	485897.684799999	100.00%	2685552.1211000	485897.684799999	452.70%		
Grand Total	6552075.84899996	6552075.84899996	100.00%	18399375.460000	6552075.84899996	180.82%		

Celdas calculadas (calculated cells)

Al igual que los miembros calculados, las celdas calculadas (calculated cells) obtienen su valor en tiempo de ejecución, a través de una fórmula MDX. Los miembros calculados aplican su fórmula de cálculo a todas las celdas del cubo. En cambio, las celdas calculadas pueden aplicar su fórmula a un subconjunto específico del cubo, un determinado rango de celdas, o incluso a una celda específica. Al igual que los miembros calculados, las celdas calculadas pueden definirse también sobre cubos virtuales.



Usando celdas calculadas, es posible determinar una fórmula que se aplique únicamente a las celdas que corresponden con el miembro Western Hemisphere del cubo como se muestra en la figura superior.

En SSAS 2012, la funcionalidad de las celdas calculadas se implementa a través de la sentencia SCOPE. Su sintaxis es:

```
SCOPE (Expresión de subcubo)
    [ Sentencias MDX ... ]
END SCOPE
```

En este ejercicio, se creará una celda calculada en el cubo “Sales” para modificar el comportamiento de la medida “Discount Amount”. En el tab Browser, verifique el valor de esta medida para el año 2004, y los miembros de los niveles “Country Region Name” y “State Province Name” como se muestra en la figura siguiente:

		Year ▾			
		2004		Grand Total	
Country Region Name ▾	State Province Name City	Sales Amount	Discount Amount	Sales Amount	Discount Amount
France	+ Essonne	57837.6708	271.1952	152420.3249	300.0661
	+ Garonne (Haute)	175358.3955	711.7497	391040.5897	10681.6048
	+ Hauts de Seine	233561.0498	9.7426	485791.5132	6792.6241
	+ Loir et Cher	85404.4674	377.4736	420190.2599	1310.132
	+ Loiret	213256.0515	1815.7076	989510.130199999	4083.933
	+ Nord	16907.802	0	34735.2737	822.7199
	+ Pas de Calais	5339.586	0	54633.6577	0
	+ Seine (Paris)	253088.2061	150.5292	1164554.3922	4926.9204
	+ Seine et Marne	23949.324	0	41185.4658	491.3773
	+ Seine Saint Denis	84560.67	0	173233.51	3762.8393
United States	+ Val d'Oise	80513.455	375.1061	209695.1458	556.8593
	Total	1229776.6781	3711.506	4116990.26309999	33729.0762
+ Alabama	15618.894	0	45429.029	0	
+ Arizona	442701.4685	2051.8749	1432585.5698	13956.4555	
+ California	1639637.87719999	5460.1291	9764270.7083	94150.9917000001	
+ Colorado	445802.8904	3407.7737	2395913.84299999	8142.0050999999	
+ Connecticut	199509.9508	38.6537	1125700.407	2197.0571	
+ Florida	398446.6817	314.2738	2300107.32199999	12594.7603	
+ Georgia	179522.8995	587.4315	1044577.7962	3035.6486	
+ Idaho	119610.504	0	227301.751	2647.6124	
+ Illinois	32886.0384	374.1796	607939.968799999	412.4569	
+ Indiana	74025.1152	80.5357	1073388.5872	965.595	
+ Kentucky	68648.178	0	163713.8398	1655.1488	
+ Maine	27643.1662	53.447	479865.5299	2043.6334	
+ Massachusetts	23001.114	0	107353.8749	150.3258	
+ Michigan	262116.9134	612.7645	1447882.2101	7719.664	

Observe que existen varias celdas para las cuales el valor de “Discount Amount” es 0. Se desea modificar esta medida de acuerdo a la siguiente regla de negocio: si el valor de “Discount Amount” para algún miembro de la jerarquía “Geography.Geographic location” en el año 2004 es 0, su valor cambiará automáticamente para ser el 10% del valor de la medida “Sales Amount”.

Efectúe los siguientes pasos:

1. En el diseñador de cubos, seleccione el tab Calculations.
2. Pulse el botón del toolbar del tab Calculations. Este botón muestra la vista de script de todas las expresiones de cálculo creadas en el cubo. Note que en el script mostrado están las expresiones correspondientes con los miembros calculados creados en el ejercicio anterior.

```
/*
The CALCULATE command controls the aggregation of leaf cells in
If the CALCULATE command is deleted or modified, the data with
You should edit this command only if you manually specify how
*/
CALCULATE;
CREATE MEMBER CURRENTCUBE.[MEASURES].[Sales Last Period]
    AS ([Measures].[Sales Amount],
parallelperiod([Order Date].[Calendar].currentmember.level,
1, [Order Date].[Calendar].currentmember)),
VISIBLE = 1;
CREATE MEMBER CURRENTCUBE.[MEASURES].[Sales Growth]
    AS CASE
        WHEN isempty(Measures.[Sales Last Period]) THEN 1
        ELSE
            (Measures.[Sales Amount] - Measures.[Sales Last Period]) /
            Measures.[Sales Last Period]
        END,
FORMAT_STRING = "Percent",
VISIBLE = 1;
```

3. Al final del script, escriba las siguientes sentencias MDX:

```

SCOPE(ROOT());
SCOPE([Measures].[Discount Amount],[Order Date].[Calendar].[Year].&[2004],
      [Geography].[Geographic location].MEMBERS,
      [Currency].[Currency Name].MEMBERS);
THIS=CASE
    WHEN [Measures].[Discount Amount] = 0 THEN [Measures].[Sales Amount]*0.1
    ELSE [Measures].[Discount Amount]
    END;
END SCOPE;

```

```

/*
The CALCULATE command controls the aggregation of leaf cells in the cube.
If the CALCULATE command is deleted or modified, the data within the cube is
You should edit this command only if you manually specify how the cube is ag-
*/
CALCULATE;
CREATE MEMBER CURRENTCUBE.[MEASURES].[Sales Last Period]
    AS ([Measures].[Sales Amount],
        parallelperiod([Order Date].[Calendar].currentmember.level,
                      1, [Order Date].[Calendar].currentmember)),
    VISIBLE = 1;
CREATE MEMBER CURRENTCUBE.[MEASURES].[Sales Growth]
    AS CASE
        WHEN isempty(Measures.[Sales Last Period]) THEN 1
        ELSE
            (Measures.[Sales Amount] - Measures.[Sales Last Period])/
            Measures.[Sales Last Period]
        END,
    FORMAT_STRING = "Percent",
    VISIBLE = 1 ;
    SCOPE(ROOT());
SCOPE([Measures].[Discount Amount],[Order Date].[Calendar].[Year].&[2004],
      [Geography].[Geographic location].MEMBERS,
      [Currency].[Currency Name].MEMBERS);
THIS=CASE
    WHEN [Measures].[Discount Amount] = 0 THEN [Measures].[Sales Amount]*0.1
    ELSE [Measures].[Discount Amount]
    END;
END SCOPE;

```

La sentencia SCOPE(ROOT()) tiene como objetivo fijar el “ámbito” del cubo; es decir, a todos los miembros “All” o miembros por defecto en el caso de dimensiones no agregables (por ejemplo, “Currency”). La segunda sentencia SCOPE expresa un subcubo, es decir, un subconjunto del cubo. Este subcubo está formado por la intersección de la medida “Discount Amount”, el miembro “2004” de la jerarquía “Order Date.Calendar”, todos los miembros de la jerarquía “Geography.Geographic location” y todos los miembros de la jerarquía “Currency.Currency Name”. La expresión que sigue:

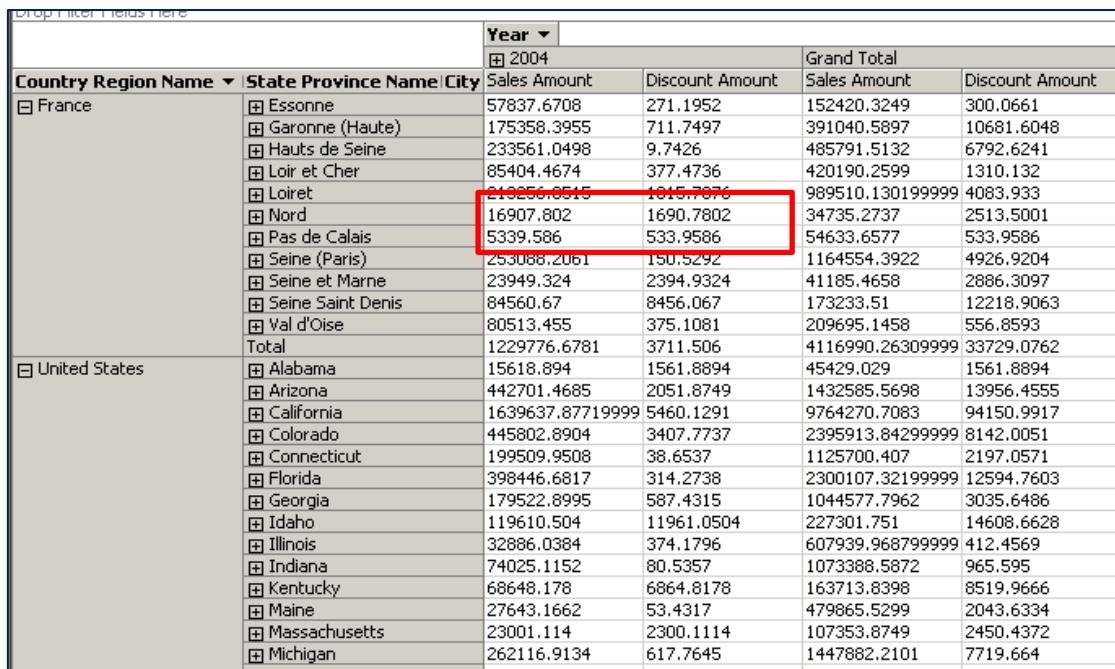
```

THIS=CASE
    WHEN [Measures].[Discount Amount] = 0 THEN [Measures].[Sales Amount]*0.1
    ELSE [Measures].[Discount Amount]
    END;

```

Define que, en el subcubo formado por la segunda sentencia SCOPE, el valor de “Discount Amount” será el 10% del valor de la medida “Sales Amount”. Por último, la sentencia END SCOPE pone fin al subcubo, restableciendo el ámbito anterior.

4. Guarde los cambios y efectúe un deployment del proyecto.
5. En el tab Browser, verifique el valor de la medida “Discount Amount” para el año 2004, y los miembros de los niveles “Country Region Name” y “State Province Name” como se muestra en la figura siguiente:



Country Region Name	State Province Name	City	Year	Grand Total	
			2004	Sales Amount	Discount Amount
France	Essonine		57837.6708	271.1952	152420.3249
	Garonne (Haute)		175358.3955	711.7497	391040.5897
	Hauts de Seine		233561.0498	9.7426	485791.5132
	Loir et Cher		85404.4674	377.4736	420190.2599
	Loiret		215256.6515	1015.7878	989510.130199999
	Nord		16907.802	1690.7802	34735.2737
	Pas de Calais		5339.586	533.9586	54633.6577
	Seine (Paris)		253088.2061	150.5292	1164554.3922
	Seine et Marne		23949.324	2394.9324	41185.4658
	Seine Saint Denis		84560.67	8456.067	173233.51
United States	Val d'Oise		80513.455	375.1081	209695.1458
	Total		1229776.6781	3711.506	4116990.263099999
	Alabama		15618.894	1561.8894	45429.029
	Arizona		442701.4685	2051.8749	1432585.5698
	California		1639637.87719999	5460.1291	9764270.7083
	Colorado		445802.8904	3407.7737	2395913.842999999
	Connecticut		199509.9508	38.6537	1125700.407
	Florida		398446.6817	314.2738	2300107.32199999
	Georgia		179522.8995	587.4315	1044577.7962
	Idaho		119610.504	11961.0504	227301.751
	Illinois		32886.0384	374.1796	607939.968799999
	Indiana		74025.1152	80.5357	1073388.5872
	Kentucky		68648.178	6864.8178	163713.8398
	Maine		27643.1662	53.4317	479865.5299
	Massachusetts		23001.114	2300.1114	107353.8749
	Michigan		262116.9134	617.7645	1447882.2101
	Total		1229776.6781	3711.506	4116990.263099999
	Alabama		15618.894	1561.8894	45429.029
	Arizona		442701.4685	2051.8749	1432585.5698
	California		1639637.87719999	5460.1291	9764270.7083

Observe el cumplimiento de la expresión de cálculo en las celdas en las que “Discount Amount” tenía un valor de 0 antes de la creación de la celda calculada.

2.3.2. Creando KPI

Key Performance Indicators (KPIs)

Los Key Performance Indicators (KPIs) son utilizados para medir el grado de cumplimiento de los objetivos de la empresa, a través de la comparación de las principales métricas del negocio con las metas establecidas. Por ejemplo, para evaluar el rendimiento del equipo de ventas, es necesario comparar los montos vendidos con las cuotas de ventas.

Típicamente, los KPIs son utilizados para construir tableros de control, a través de los cuales los usuarios pueden visualizar semáforos de control e información de tendencia sobre los aspectos críticos del negocio. Todo KPI se basa en la comparación entre un valor actual y un valor objetivo. El grado de diferencia entre el valor actual y el valor objetivo determina si los objetivos trazados se están cumpliendo o no.

Display Structure	Value	Goal	Status
 KPI Ventas Internet	11851296.66	28862504.96	
 KPI Ventas Internet Francia	2462361.69	2058495.13	
 KPI Ventas Internet USA	9388934.97	26804009.83	

Al definir KPIs con SSAS 2014, se debe proporcionar la siguiente información:

- Nombre: Nombre del KPI
- Value Expression: Expresión MDX que establece el valor actual del KPI
- Goal Expression: Expresión MDX que establece el valor objetivo del KPI
- Status Indicator: Determina el símbolo que se utilizará para mostrar visualmente el KPI. Se puede elegir una variedad de formas de representación, como termómetros, semáforos, etc.
- Status Expression: Expresión MDX que devuelve un valor normalizado entre -1 y 1, dependiendo de la diferencia entre el valor actual (Value Expression) y el valor objetivo (Goal Expression). Si el valor devuelto es -1, el indicador de status (Status Indicator) del KPI mostrará una alarma (por ejemplo, un semáforo en rojo). Si el valor es 1, el indicador mostrará un signo de conformidad (por ejemplo, un semáforo en verde).

Opcionalmente, puede ingresarse la siguiente información:

- Trend: Expresión MDX que establece la tendencia de evolución del valor actual. Generalmente, esta expresión se basa en la diferencia del valor actual entre el período de tiempo actual y un período anterior. También debe devolver un valor normalizado entre -1 y 1.
- Trend indicator: Determina el símbolo que se utilizará para mostrar visualmente si la tendencia es positiva o negativa.
- Parent KPI: Los KPIs pueden visualizarse y procesarse de manera jerárquica; es decir, un KPI puede contener a otros KPIs. Por ejemplo, un KPI llamado "Ventas mundiales" puede contener a los KPIs de "Ventas en USA", "Ventas en Alemania", "Ventas en Canadá", etc.

SSAS 2014 posee un visor de KPIs, integrado con el SQL Server Data Tools. Sin embargo, debe tenerse en cuenta que la capacidad de los usuarios finales para visualizar los KPIs depende de la herramienta de consulta que estén utilizando. Por ejemplo, sólo a partir de la versión 2007 de Office, las herramientas de oficina de Microsoft pueden mostrar visualmente los KPIs creados con SSAS 2014.

Ejercicio:

Crear KPIs en el cubo SALES.

El cubo "Sales" contiene medidas de ventas por distribuidor (Reseller Sales) y ventas por Internet (Internet Sales). En este escenario, se asumirá que el objetivo de la corporación es aumentar las ventas por Internet hasta que sean el 50% de las ventas por distribuidor. Por tanto, se construirá un KPI para evaluar el cumplimiento de este objetivo.

Efectúe los siguientes pasos:

1. En el diseñador de cubos, seleccione el tab KPIs (es el cuarto tab empezando por la izquierda). En el toolbar del tab KPIs, pulse el botón  para agregar un nuevo KPI.

2. Se abrirá el editor de KPIs. En la propiedad Name del nuevo KPI, escriba “KPI Ventas Internet USA”. En la propiedad Value Expression, escriba la siguiente expresión MDX para obtener el valor de “Internet Sales Amount” en los Estados Unidos:

$([Measures].[Internet Sales Amount], [Geography].[Geographic location].[Country Region Name].[United States])$

3. En la propiedad Goal Expression, escriba la siguiente expresión MDX para obtener el valor de “Sales Amount” para los Estados Unidos, multiplicado por 0.5:

$0.5 * ([Measures].[Sales Amount], [Geography].[Geographic location].[Country Region Name].[United States])$

4. En la propiedad Status Indicator, seleccione el símbolo  .
5. En la propiedad Status Expression, escriba la siguiente expresión MDX para devolver valores entre -1 y 1 dependiendo del grado de cumplimiento de la meta de ventas por Internet. Note el uso de las funciones MDX KPIVALUE y KPIGOAL.

```
case
    when KPIVALUE("KPI Ventas Internet USA") >= KPIGOAL("KPI Ventas Internet USA") then 1
    when KPIVALUE("KPI Ventas Internet USA") >= 0.9 * KPIGOAL("KPI Ventas Internet USA") then 0.75
    when KPIVALUE("KPI Ventas Internet USA") >= 0.7 * KPIGOAL("KPI Ventas Internet USA") then 0.5
    when KPIVALUE("KPI Ventas Internet USA") >= 0.5 * KPIGOAL("KPI Ventas Internet USA") then 0
    else -1
end
```

Al finalizar, el editor de KPIs debe quedar de la siguiente manera:

Name: KPI Ventas Internet USA

Associated measure group: <All>

Value Expression: ([Measures].[Internet Sales Amount], [Geography].[Geographic location].[Country Region Name].[United States])

Goal Expression: 0.5*([Measures].[Sales Amount], [Geography].[Geographic location].[Country Region Name].[United States])

Status:

Status indicator: Gauge

Status expression:

```
case
    when KPIVALUE("KPI Ventas Internet USA") >= KPIGOAL("KPI Ventas Internet USA") then 1
    when KPIVALUE("KPI Ventas Internet USA") >= 0.9 * KPIGOAL("KPI Ventas Internet USA") then 0.75
    when KPIVALUE("KPI Ventas Internet USA") >= 0.7 * KPIGOAL("KPI Ventas Internet USA") then 0.5
    when KPIVALUE("KPI Ventas Internet USA") >= 0.5 * KPIGOAL("KPI Ventas Internet USA") then 0
    else -1
end
```

6. Repita los pasos anteriores para crear un nuevo KPI con nombre “KPI Ventas Internet Francia”. En la propiedad Value Expression, escriba la siguiente expresión MDX para obtener el valor de “Internet Sales Amount” en Francia:

[Measures].[Internet Sales Amount],
[Geography].[Geographic location].[Country Region Name].[France])

7. En la propiedad Goal Expression, escriba la siguiente expresión MDX para obtener el valor de “Sales Amount” para Francia, multiplicado por 0.5:

0.5*([Measures].[Sales Amount],
[Geography].[Geographic location].[Country Region Name].[France])

8. En la propiedad Status Indicator, seleccione el símbolo  .

9. En la propiedad Status Expression, escriba la siguiente expresión MDX:

```
case
    when KPIVALUE("KPI Ventas Internet Francia") >= KPIGOAL("KPI Ventas Internet Francia") then 1
    when KPIVALUE("KPI Ventas Internet Francia") >= 0.9 * KPIGOAL("KPI Ventas Internet Francia") then 0.75
    when KPIVALUE("KPI Ventas Internet Francia") >= 0.7 * KPIGOAL("KPI Ventas Internet Francia") then 0.5
    when KPIVALUE("KPI Ventas Internet Francia") >= 0.5 * KPIGOAL("KPI Ventas Internet Francia") then 0
    else -1
end
```

Al finalizar, el editor de KPIs debe mostrar la siguiente información:

Name: KPI Ventas Internet Francia

Associated measure group: <All>

Value Expression: `([Measures].[Internet Sales Amount], [Geography].[Geographic location].[Country Region Name].[France])`

Goal Expression: `0.5 * ([Measures].[Sales Amount], [Geography].[Geographic location].[Country Region Name].[France])`

Status:

Status indicator: Gauge

Status expression:

```
case
    when KPIVALUE("KPI Ventas Internet Francia") >= KPIGOAL("KPI Ventas Internet Francia")
        then 1
    when KPIVALUE("KPI Ventas Internet Francia") >= 0.9 * KPIGOAL("KPI Ventas Internet Francia")
        then 0.75
    when KPIVALUE("KPI Ventas Internet Francia") >= 0.7 * KPIGOAL("KPI Ventas Internet Francia")
        then 0.5
    when KPIVALUE("KPI Ventas Internet Francia") >= 0.5 * KPIGOAL("KPI Ventas Internet Francia")
        then 0
    else -1
end
```

10. Cree un nuevo KPI llamado “KPI Ventas Internet”. En la propiedad Value Expression, escriba la siguiente expresión MDX para obtener la suma de los valores de los KPIs creados en los pasos anteriores:

KPIVALUE("KPI Ventas Internet USA") + KPIVALUE("KPI Ventas Internet Francia")

11. En la propiedad Goal Expression, escriba la siguiente expresión MDX para obtener el valor de la suma de los objetivos de los KPIs creados en los pasos anteriores:

KPIGOAL("KPI Ventas Internet USA") + KPIGOAL("KPI Ventas Internet Francia")

12. En la propiedad Status Indicator, seleccione el símbolo  .

13. En la propiedad Status Expression, escriba la siguiente expresión MDX:

```
case
    when KPIVALUE("KPI Ventas Internet") >= KPIGOAL("KPI Ventas Internet")
        then 1
    when KPIVALUE("KPI Ventas Internet") >= 0.9 * KPIGOAL("KPI Ventas Internet")
        then 0.75
    when KPIVALUE("KPI Ventas Internet") >= 0.7 * KPIGOAL("KPI Ventas Internet")
        then 0.5
    when KPIVALUE("KPI Ventas Internet") >= 0.5 * KPIGOAL("KPI Ventas Internet")
        then 0
    else -1
end
```

Al finalizar, el editor de KPIs debe mostrar la siguiente información:

The screenshot shows the KPI Editor interface with the following details:

- Name:** KPI Ventas Internet
- Associated measure group:** <All>
- Value Expression:** KPIVALUE("KPI Ventas Internet USA") + KPIVALUE("KPI Ventas Internet Francia")
- Goal Expression:** KPIGOAL("KPI Ventas Internet USA") + KPIGOAL("KPI Ventas Internet Francia")
- Status:**
 - Status indicator: Gauge
 - Status expression:

```
case
when KPIVALUE("KPI Ventas Internet") >= KPIGOAL("KPI Ventas Internet") then 1
when KPIVALUE("KPI Ventas Internet") >= 0.9 * KPIGOAL("KPI Ventas Internet") then 0.75
when KPIVALUE("KPI Ventas Internet") >= 0.7 * KPIGOAL("KPI Ventas Internet") then 0.5
when KPIVALUE("KPI Ventas Internet") >= 0.5 * KPIGOAL("KPI Ventas Internet") then 0
else -1
end
```

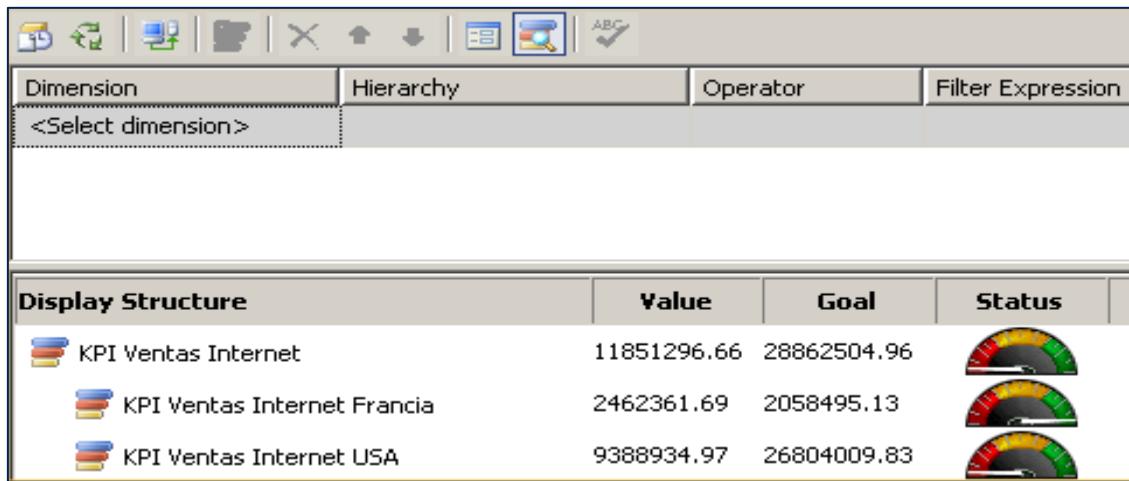
14. En el lado izquierdo del editor de KPIs se encuentra el panel del KPI Organizer. Seleccione el ítem “KPI Ventas Internet USA” y, en el editor, expanda el cuadro Additional Properties. En la propiedad Parent KPI, seleccione “KPI Ventas Internet”.

The screenshot shows the KPI Organizer interface with the following details:

KPI Organizer	<pre>KPIGOAL("KPI Ventas Internet USA") then 0.75 when KPIVALUE("KPI Ventas Internet USA") >= 0.7 * KPIGOAL("KPI Ventas Internet USA") then 0.5 when KPIVALUE("KPI Ventas Internet USA") >= 0.5 * KPIGOAL("KPI Ventas Internet USA") then 0 else -1 end</pre>
	Trend Additional Properties Display folder: Parent KPI: KPI Ventas Internet
Calculation Tools	

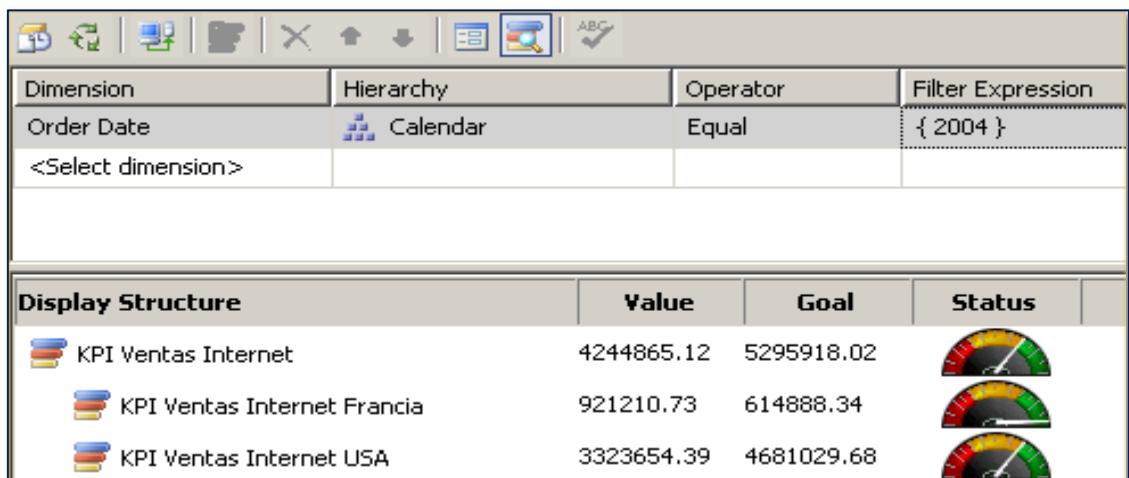
Haga caso omiso del símbolo de advertencia .

15. Repita el paso (14) para “KPI Ventas Internet Francia”.
16. Guarde los cambios y efectúe el deployment del proyecto.
17. En el toolbar del tab KPIs, pulse el botón para abrir el visor de KPIs de SSAS 2014:



Observe que en USA el indicador de status muestra que los objetivos de crecimiento de ventas por Internet están lejos de cumplirse. Sin embargo, esto muestra información evaluada sobre la totalidad del cubo "Sales". En la parte superior, en el grid de selección de criterios, sobre la etiqueta <Select dimension> seleccione la dimensión "Order Date".

En la columna Hierarchy, seleccione la jerarquía "Calendar". En la columna Operator, seleccione el operador "Equal", y en la columna Filter Expression, seleccione el año 2004. Luego, haga un clic sobre el gráfico que muestra los KPIs. Los valores cambiarán de la siguiente forma:



Observe que la información de status mostrada por los KPIs muestra una clara mejora en el año 2004, respecto de los objetivos trazados para las ventas por Internet.

2.4. DEFINICIÓN DE TRADUCCIONES Y PERSPECTIVAS

2.4.1. Creando Traducciones

El soporte multi – lenguaje en SQL Server Analysis Services 2012 se implementa a través del concepto de traducciones (translations). A través de las traducciones se puede definir los siguientes elementos:

- Los nombres de cubos, medidas y dimensiones para diferentes idiomas.

- Los nombres de los atributos y jerarquías de cada dimensión para diferentes idiomas.
- Los datos que se muestran al navegar a través de los atributos y jerarquías de cada dimensión para diferentes idiomas.

El servidor de Analysis Services detecta el lenguaje utilizado por el cliente que efectúa la consulta. Acto seguido, define los valores que se mostrarán a dicho cliente con base en las traducciones definidas en la solución. Por ejemplo, si un cliente se conecta desde una PC configurada para el idioma español, visualizará los nombres de cubos, medidas, dimensiones, etc. traducidos al español, de acuerdo con las equivalencias definidas en las traducciones creadas en el proyecto de análisis.

Se pueden crear dos tipos de traducciones.

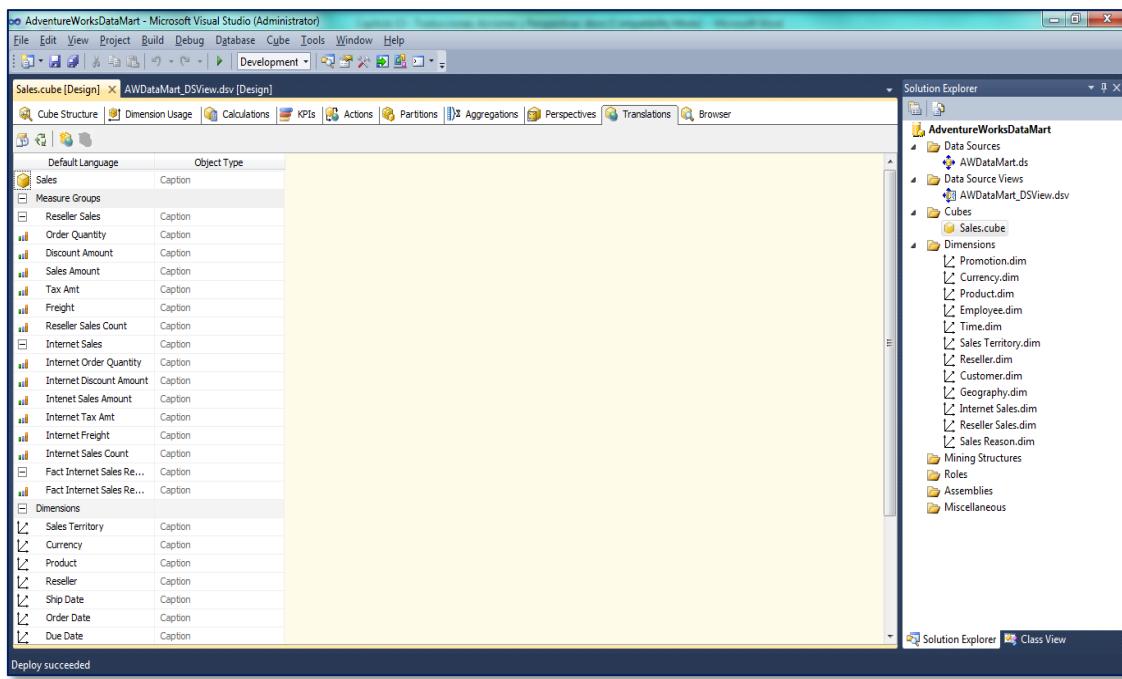
- Traducciones de cubo (Cube Translations).
Se definen a través del tab Translations del editor de cubos. Se pueden traducir los siguientes elementos:
 - El nombre del cubo
 - El nombre de las medidas
 - El nombre de los folders donde se muestran las medidas
 - Los nombres de las dimensiones que participan en el cubo
 - Nombres de perspectivas
 - Nombres, descripciones y folders de los KPIs
 - Acciones (actions)
 - Sets con nombre (named sets)
 - Miembros calculados (calculated members)
- Traducciones de dimensión (Dimension Translations)
Se definen a través del tab Translations del editor de dimensiones. Se pueden traducir los siguientes elementos:
 - El nombre de la dimensión
 - Nombre del miembro All
 - Nombre del miembro Unknown (desconocido)
 - Nombres de jerarquías
 - Los nombres de los folders donde se muestran las jerarquías
 - Nombres de niveles
 - Nombres de atributos
 - Valores mostrados al navegar a través de los atributos y jerarquías de la dimensión

Ejercicio:

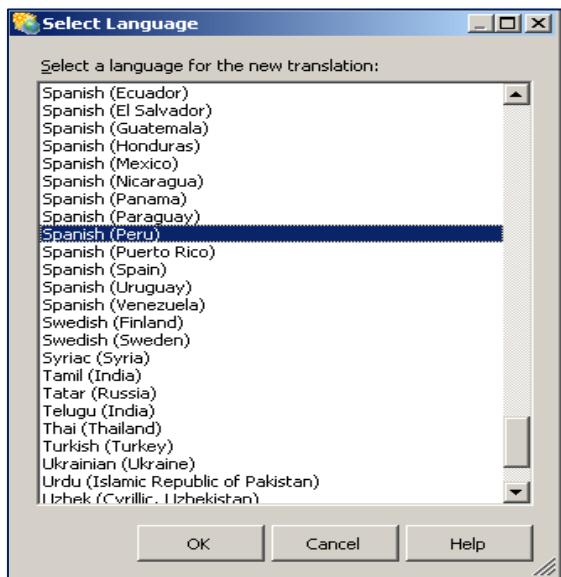
Crear Traducciones en la solución AdventureWorksDatamart

En el siguiente ejercicio, se definirá una traducción al español para el cubo “Sales” y la dimensión “Product”. También se definirán traducciones de los valores del atributo “Product Name”.

1. Con el Sql Server Data Tools, abra el proyecto “AdventureWorksDataMart”. En el Solution Explorer, haga doble clic en el cubo “Sales”.
2. En el diseñador de cubos, seleccione el tab Translations (es el noveno empezando por la izquierda). Las traducciones se crean en este tab.



3. En el toolbar del tab Translations, pulse el botón para iniciar la creación de una nueva traducción.
Se abrirá una ventana que permitirá seleccionar el idioma de la traducción. Seleccione el idioma “Spanish (Peru)”.



4. Pulse el botón OK. Observe que se ha creado una traducción llamada “Spanish (Peru)”.

Object Type	Object Name	Spanish (Peru)
Sales	Caption	Ventas
Measure Groups		
Reseller Sales	Caption	Ventas por Distribuidor
Order Quantity	Caption	Cantidad por Distribuidor
Discount Amount	Caption	Descuento por Distribuidor
Sales Amount	Caption	Monto por Distribuidor
Tax Amt	Caption	Impuestos por Distribuidor
Freight	Caption	Flete por Distribuidor
Reseller Sales Count	Caption	Cantidad de Ventas por Distribuidor
Internet Sales	Caption	Ventas por Internet
Internet Order Quantity	Caption	Cantidad por Internet
Internet Discount Amount	Caption	Descuento por Internet
Internet Sales Amount	Caption	Monto por Internet
Internet Tax Amt	Caption	Impuestos por Internet
Internet Freight	Caption	Flete por Internet
Internet Sales Count	Caption	Cantidad de Ventas por Internet
Fact Internet Sales Re...	Caption	
Fact Internet Sales Re...	Caption	

5. En la traducción “Spanish (Peru)”, asigne las siguientes traducciones:

Objeto	Nombre	Traducción
Cubo	Sales	Ventas
Grupo de Medidas	Reseller Sales	Ventas por Distribuidor
Medida	Order Quantity	Cantidad por Distribuidor
Medida	Discount Amount	Descuento por Distribuidor
Medida	Sales Amount	Monto por Distribuidor
Medida	Tax Amt	Impuestos por Distribuidor
Medida	Freight	Flete por Distribuidor
Medida	Reseller Sales Count	Cantidad de Ventas por Distribuidor
Grupo de Medidas	Internet Sales	Ventas por Internet
Medida	Internet Order Quantity	Cantidad por Internet
Medida	Internet Discount Amount	Descuento por Internet
Medida	Internet Sales Amount	Monto por Internet
Medida	Internet Tax Amt	Impuestos por Internet
Medida	Internet Freight	Flete por Internet
Medida	Internet Sales Count	Cantidad de Ventas por Internet

6. Al finalizar, el editor tendrá la siguiente apariencia:

Object Type	Object Name	Spanish (Peru)
Sales	Caption	Ventas
Measure Groups		
Reseller Sales	Caption	Ventas por Distribuidor
Order Quantity	Caption	Cantidad por Distribuidor
Discount Amount	Caption	Descuento por Distribuidor
Sales Amount	Caption	Monto por Distribuidor
Tax Amt	Caption	Impuestos por Distribuidor
Freight	Caption	Flete por Distribuidor
Reseller Sales Count	Caption	Cantidad de Ventas por Distribuidor
Internet Sales	Caption	Ventas por Internet
Internet Order Quantity	Caption	Cantidad por Internet
Internet Discount Amount	Caption	Descuento por Internet
Internet Sales Amount	Caption	Monto por Internet
Internet Tax Amt	Caption	Impuestos por Internet
Internet Freight	Caption	Flete por Internet
Internet Sales Count	Caption	Cantidad de Ventas por Internet
Fact Internet Sales Re...	Caption	
Fact Internet Sales Re...	Caption	

7. Haga scroll hacia abajo en el editor, y proceda a definir las siguientes traducciones para las dimensiones utilizadas por el cubo “Sales”:

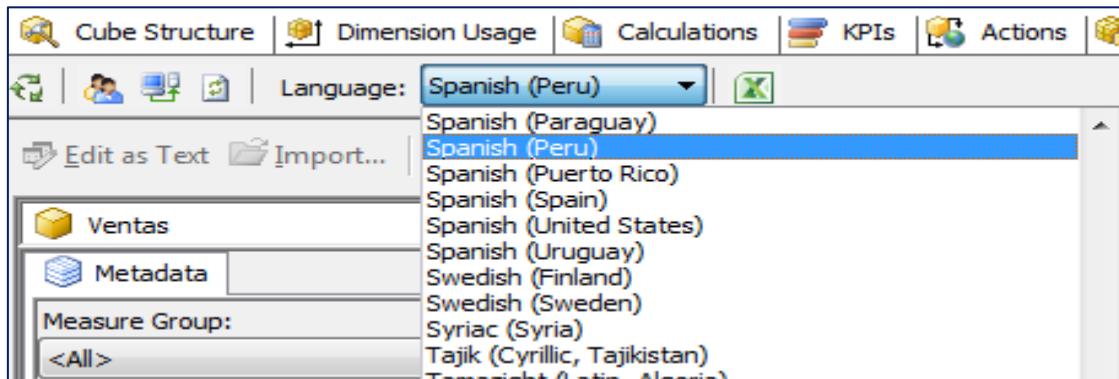
Nombre de Dimensión	Traducción
Sales Territory	Territorio de Ventas
Currency	Moneda
Due Date	Fecha Programada
Product	Producto
Ship Date	Fecha de Despacho
Promotion	Promoción
Reseller	Distribuidor
Order Date	Fecha de Orden
Employee	Empleado
Customer	Cliente
Geography	Geografía
Internet Sales	Ordenes por Internet
Reseller Sales	Ordenes por Distribuidor
Sales Reason	Motivo de Compra

8. Al finalizar, el editor tendrá la siguiente apariencia:

Object Type	Spanish (Peru)
Reseller Sales Count	Cantidad de Ventas por Distribuidor
Internet Sales	Ventas por Internet
Internet Order Quantity	Cantidad por Internet
Internet Discount Amount	Descuento por Internet
Internet Sales Amount	Monto por Internet
Internet Tax Amt	Impuestos por Internet
Internet Freight	Flete por Internet
Internet Sales Count	Cantidad de Ventas por Internet
Fact Internet Sales Re...	Cantidad de Ventas por Internet
Fact Internet Sales Re...	Cantidad de Ventas por Internet
Dimensions	
Sales Territory	Territorio de Ventas
Currency	Moneda
Product	Producto
Reseller	Distribuidor
Ship Date	Fecha de Despacho
Order Date	Fecha de Orden
Due Date	Fecha Programada
Promotion	Promoción
Employee	Empleado
Customer	Cliente
Geography	Geografía
Internet Sales	Ordenes por Internet
Reseller Sales	Ordenes por Distribuidor
Sales Reason	Motivo de Compra

9. Guarde los cambios y despliegue la solución.

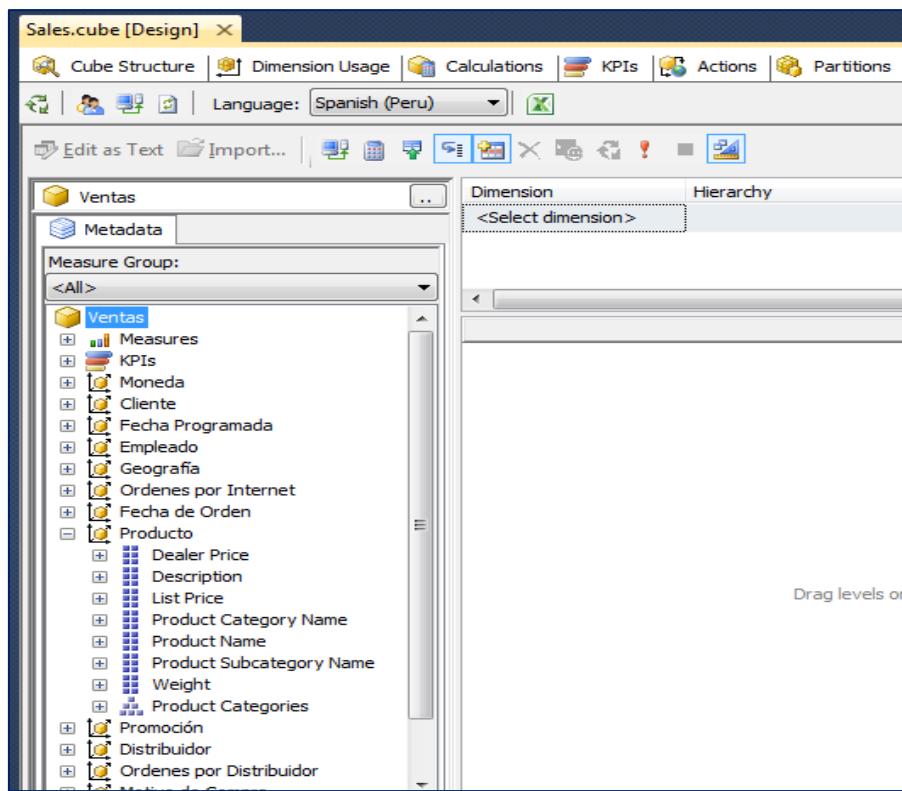
10. Consulte el cubo “Sales” a través del tab Browser. En la propiedad Language, seleccione el idioma “Spanish (Peru)” como se muestra en la siguiente figura:



11. Observe los cambios en los nombres del cubo, las medidas y las dimensiones.

A screenshot of the Sales.cube [Design] window. The top navigation bar is identical to the previous screenshot. The main area shows the cube structure. On the left, the Measures section is expanded, displaying various measures under the Ventas dimension, such as Cantidad de Ventas por Distribuidor, Descuento por Distribuidor, Flete por Distribuidor, Impuestos por Distribuidor, and Monto por Distribuidor. Other dimensions listed include Moneda, Cliente, Fecha Programada, Empleado, Geografía, Ordenes por Internet, Fecha de Orden, Producto, Promoción, Distribuidor, and Ordenes por Distribuidor. A large empty area on the right is labeled "Drag levels or hierarchies here".

12. Expanda la dimensión “Producto”. Observe que no ha cambiado el nombre de ninguno de los atributos y jerarquías de la dimensión.



13. En el Solution Explorer, haga doble clic en la dimensión “Product”.
14. En el editor de dimensiones, seleccione el tab Translations (es el tercero empezando por la izquierda).
15. En el toolbar del tab Translations, pulse el botón para iniciar la creación de una nueva traducción. Se abrirá una ventana que permitirá seleccionar el idioma de la traducción.
Seleccione el idioma “Spanish (Perú)” y pulse el botón OK. Observe que se ha creado una nueva traducción llamada “Spanish (Perú)”:

Default Language	Object Type	Spanish (Peru)
Product	Caption	
All Products	AttributeAllMemberNa...	
Attributes		
Product Name	Caption	
Description	Caption	
Product Subcategory N...	Caption	
Product Category Name	Caption	
Dealer Price	Caption	
List Price	Caption	
Weight	Caption	
Hierarchies		
Product Categories	Caption	
All Products	AllMemberName	
Product Category N...	Caption	
Product Subcategor...	Caption	
Product Name	Caption	

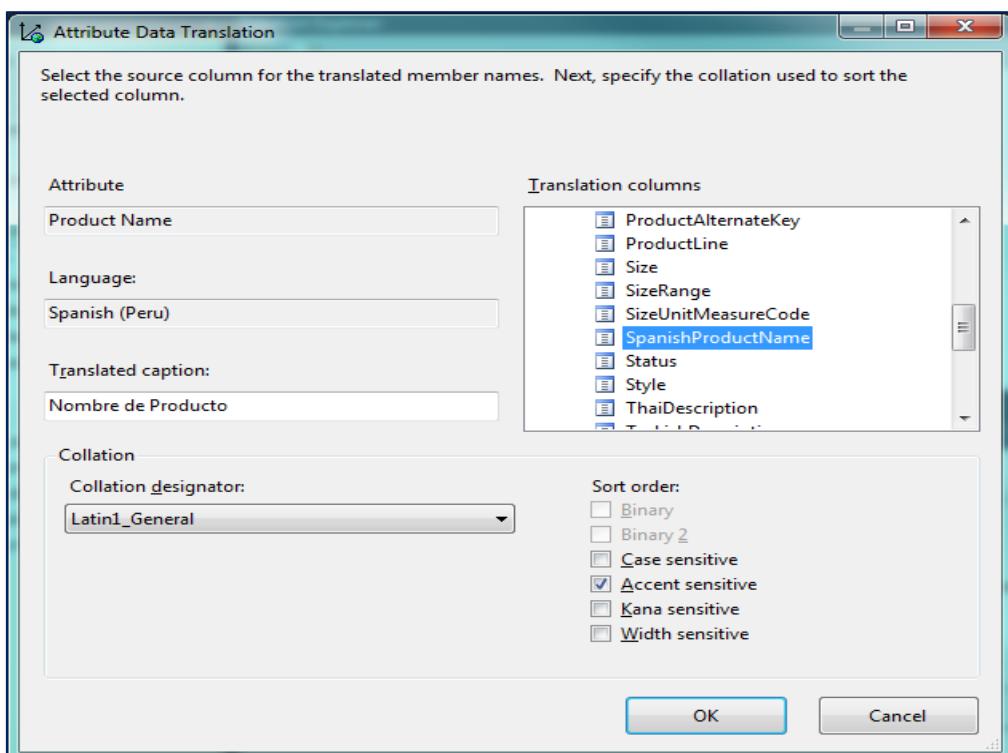
16. En la traducción “Spanish (Perú)”, asigne las siguientes traducciones:

Objeto	Nombre	Traducción
Dimensión	Product	Producto
Miembro All	All Products	Todos los productos
Atributo	Product Name	Nombre de Producto
Atributo	Description	Descripción
Atributo	Product Subcategory Name	Subcategoría de Producto
Atributo	Product Category Name	Categoría de Producto
Atributo	Dealer Price	Precio en Distribuidor
Atributo	List Price	Precio de Lista
Atributo	Weight	Peso
Jerarquía	Product Categories	Productos por Categoría
Miembro All	All Products	Todos los productos
Nivel	Product Category Name	Categoría de Producto
Nivel	Product Subcategory Name	Subcategoría de Producto
Nivel	Product Name	Nombre de Producto

17. Al finalizar, el editor tendrá la siguiente apariencia:

Sales.cube [Design]		
	Dimension Structure	Attribute Relationships
	Translations	Browser
Default Language	Object Type	Spanish (Peru)
Product	Caption	Producto
All Products	AttributeAllMemberNa...	Todos los productos
Attributes		
Product Name	Caption	Nombre de Producto
Description	Caption	Descripción
Product Subcategory N...	Caption	Subcategoría de Producto
Product Category Name	Caption	Categoría de Producto
Dealer Price	Caption	Precio en Distribuidor
List Price	Caption	Precio de Lista
Weight	Caption	Peso
Hierarchies		
Product Categories	Caption	Productos por Categoría
All Products	AllMemberName	Todos los productos
Product Category N...	Caption	Categoría de Producto
Product Subcategor...	Caption	Subcategoría de Producto
Product Name	Caption	Nombre de Producto

18. Ubique el cursor en la traducción que corresponde con el atributo “Product Name”. Observe la aparición de un botón [.] y púlselo. Aparecerá la ventana Attribute Data Translation que permite traducir los datos del atributo. En la lista Translation columns, seleccione la columna “SpanishProductName”. Ésta es una columna perteneciente a la tabla de dimensión “DimProduct” que contiene el nombre de cada producto en español.



19. Pulse el botón OK. Al finalizar, el editor de dimensiones tendrá la siguiente apariencia:

Default Language	Object Type	Spanish (Peru)
Product	Caption	Producto
All Products	AttributeAllMemberNa...	Todos los productos
Attributes		
Product Name	Caption	Nombre de Producto
Description	Caption	Descripción
Product Subcategory N...	Caption	Subcategoría de Producto
Product Category Name	Caption	Categoría de Producto
Dealer Price	Caption	Precio en Distribuidor
List Price	Caption	Precio de Lista
Weight	Caption	Peso
Hierarchies		
Product Categories	Caption	Productos por Categoría
All Products	AllMemberName	Todos los productos
Product Category N...	Caption	Categoría de Producto
Product Subcategor...	Caption	Subcategoría de Producto
Product Name	Caption	Nombre de Producto

20. Despliegue la solución. Al finalizar el procesamiento, consulte el cubo “Sales” con el tab Browser.

Cambie el lenguaje a “Spanish (Perú)”. Expanda la dimensión “Producto”; observe que los nombres de los atributos y jerarquías han cambiado. Arrastre el atributo “Nombre de Producto” y del grupo de medidas Ventas por Distribuidor arrastre la medida Cantidad de Ventas por Distribuidor a la sección de datos. Observe que aparecen los nombres en español para los productos (los valores etiquetados como “(Blank)” corresponden con productos para los cuales la columna “SpanishColumnName” de la tabla “DimProduct” tiene valor nulo).

Nombre de Producto	Cantidad de Ventas po...
Lavado de bicicleta...	230
Cable antirrobo	307
Cadena	268
Camiseta clásica, G	248
Camiseta clásica, M	180
Camiseta clásica, P	205
Frenos delanteros	300
Desviador delantero	168
Guantes completos, G	152
Guantes completos, M	250
Guantes completos, P	248
(Blank)	158
(Blank)	65
(Blank)	74
(Blank)	176
(Blank)	187
(Blank)	99

2.4.2. Creando Perspectivas

Un cubo de SQL Server Analysis Services 2012 puede ser definido a partir de múltiples tablas de hechos y tablas de dimensión. Como resultado, se pueden construir cubos de alta complejidad que combinen un gran número de medidas y dimensiones. En estos casos, los usuarios finales pueden encontrar difícil navegar a través de la información del cubo. SSAS 2014 provee la capacidad de definir perspectivas (perspectives) para facilitar esta interacción.

Una perspectiva es un subconjunto de las medidas y dimensiones que componen un cubo. Cada perspectiva es vista por el usuario final como un cubo distinto, en el cual sólo participan las medidas y dimensiones de su interés. De esta manera, la complejidad del cubo subyacente queda oculta.

Las perspectivas controlan la visibilidad de los siguientes elementos:

- Dimensiones
- Atributos
- Jerarquías
- Grupos de medidas
- Medidas
- KPIs
- Miembros calculados (calculated members) y sets con nombre (named sets)

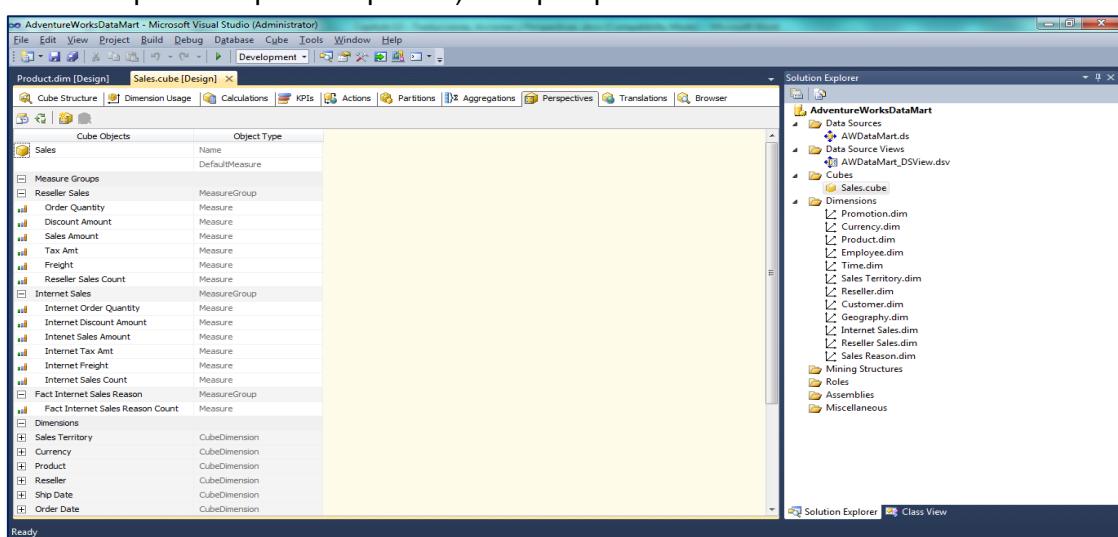
Las perspectivas sirven para mejorar la interacción del usuario final. No constituyen un mecanismo de seguridad.

Ejercicio:

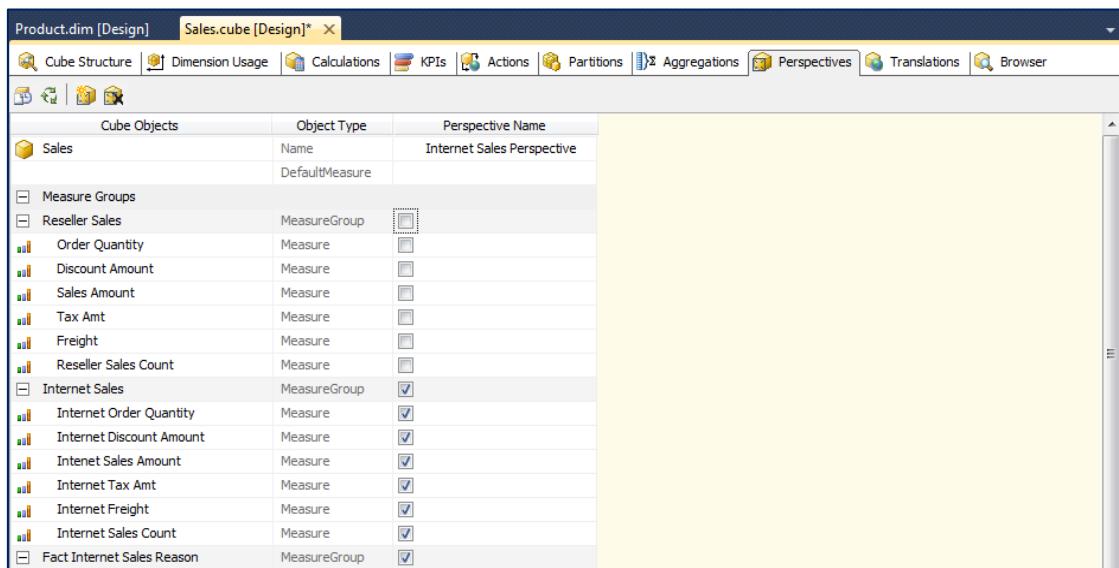
Crear Perspectivas en el cubo Sales.

En este ejercicio, se creará una perspectiva que contendrá únicamente a las medidas y dimensiones que están relacionadas con las ventas a través de Internet. Efectúe los siguientes pasos:

1. En el Solution Explorer, haga doble clic en el cubo “Sales”.
2. En el diseñador de cubos, seleccione el tab Perspectives (es el octavo empezando por la izquierda). Las perspectivas se crean en este tab.



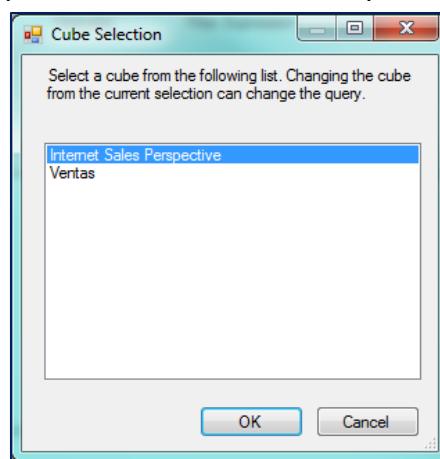
3. En el toolbar del tab Perspectives, pulse el botón  para iniciar la creación de una nueva perspectiva.
4. En la primera fila, en la columna Perspective Name, asigne el nombre “Internet Sales Perspective” a la nueva perspectiva.
5. A continuación, desmarque el checkbox correspondiente con el grupo de medidas “Reseller Sales”. Al finalizar, el editor debe tener la siguiente apariencia:



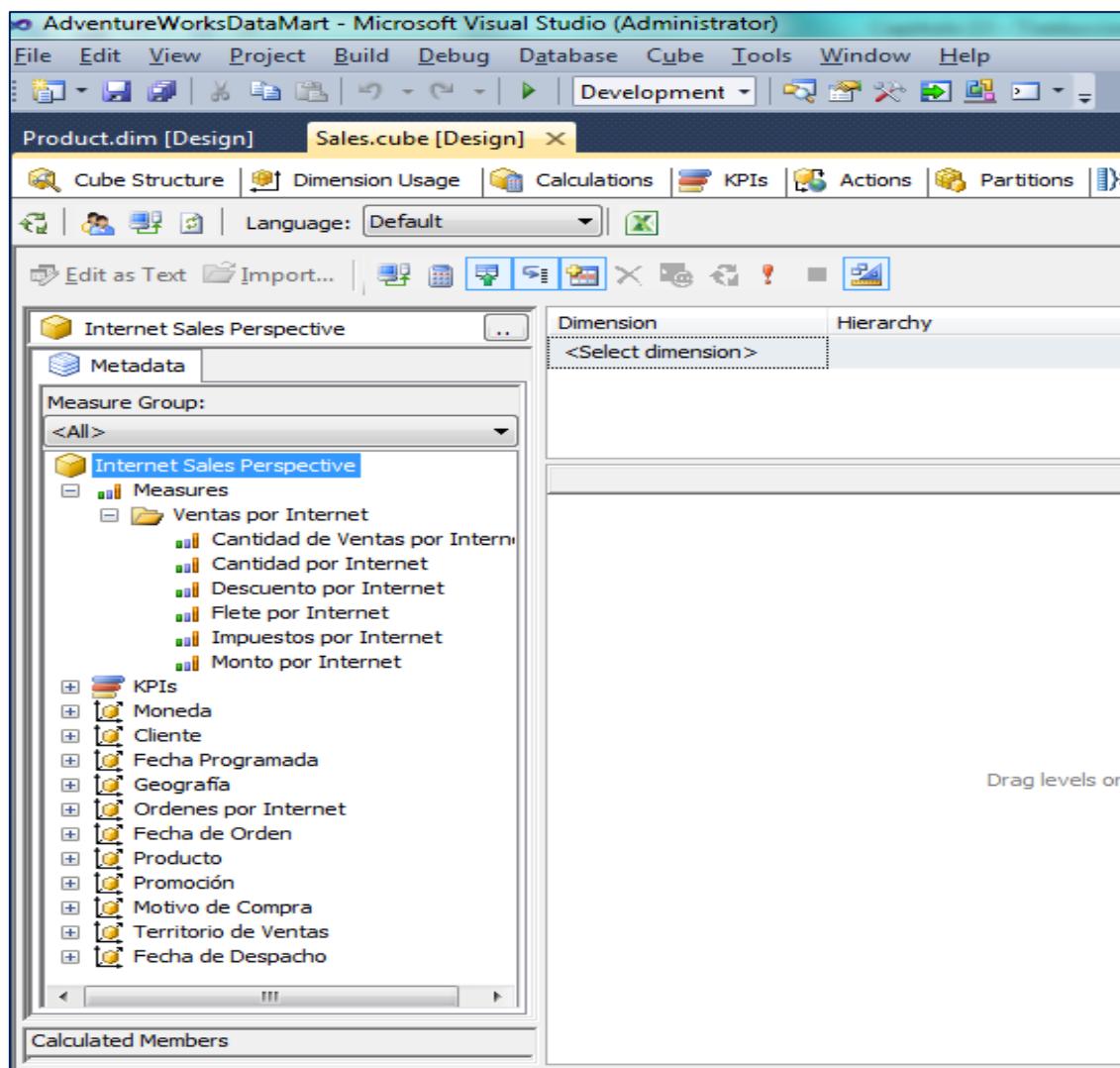
Cube Objects	Object Type	Perspective Name
Sales	Name	Internet Sales Perspective
	DefaultMeasure	
Measure Groups		
Reseller Sales	MeasureGroup	<input type="checkbox"/>
Order Quantity	Measure	<input type="checkbox"/>
Discount Amount	Measure	<input type="checkbox"/>
Sales Amount	Measure	<input type="checkbox"/>
Tax Amt	Measure	<input type="checkbox"/>
Freight	Measure	<input type="checkbox"/>
Reseller Sales Count	Measure	<input type="checkbox"/>
Internet Sales	MeasureGroup	<input checked="" type="checkbox"/>
Internet Order Quantity	Measure	<input checked="" type="checkbox"/>
Internet Discount Amount	Measure	<input checked="" type="checkbox"/>
Internet Sales Amount	Measure	<input checked="" type="checkbox"/>
Internet Tax Amt	Measure	<input checked="" type="checkbox"/>
Internet Freight	Measure	<input checked="" type="checkbox"/>
Internet Sales Count	Measure	<input checked="" type="checkbox"/>
Fact Internet Sales Reason	MeasureGroup	<input checked="" type="checkbox"/>

6. Haga scroll hacia abajo en el editor para continuar definiendo la perspectiva “Internet Sales Perspective”. Desmarque los checkboxes correspondientes con las dimensiones “Reseller”, “Employee” y “Reseller Sales”. Desmarque también los checkboxes correspondientes a los miembros calculados “Sales Last Period” y “Sales Growth”.
7. Al finalizar, el editor debe tener la siguiente apariencia:

8. Guarde los cambios y despliegue la solución. Consulte el cubo a través del tab Browser.
9. En la lista de cubos, observe que se encuentran las perspectivas “Ventas” (correspondiente con la totalidad del cubo) y “Internet Sales Perspective”. Seleccione la perspectiva “Internet Sales Perspective”.



10. Observe que sólo están presentes las medidas y dimensiones que fueron marcadas como pertenecientes a la perspectiva “Internet Sales Perspective”.



Acciones

Las acciones (actions) son comandos que pueden ser ejecutados para presentar información adicional a los usuarios mientras navegan en un cubo. Permiten profundizar el nivel de análisis de la información. Por ejemplo, un usuario puede observar un valor bajo en una medida de ventas al consultar el cubo. Para encontrar la causa de esto, puede disparar una acción que muestre el detalle de las facturas que están detrás de dicha cifra, o un reporte que muestre información detallada sobre la evolución de las ventas.

Las acciones son implementadas a través de sentencias MDX que producen comandos, los cuales pueden ser ejecutados desde el cliente que efectúa la consulta sobre el cubo. SQL Server Analysis Services 2014 permite crear los siguientes tipos de acciones:

- URL: Muestra una página web, utilizando un URL definido por el implementador del cubo.
- Statement: Corre un comando OLE DB.
- Rowset: Retorna un conjunto de filas.
- Dataset: Retorna un dataset.
- Drillthrough: Muestra filas de datos provenientes de una o varias dimensiones del cubo.

- Reporting: Muestra un reporte parametrizable de Reporting Services.
- Proprietary: Ejecuta una acción personalizada definida por el usuario, no soportada por los tipos de acción ya definidos en esta lista.

Para utilizar acciones, es necesario que la herramienta de consulta utilizada para consultar el cubo provea soporte para acciones. Por ejemplo, Microsoft Excel 2007 soporta el uso de acciones.

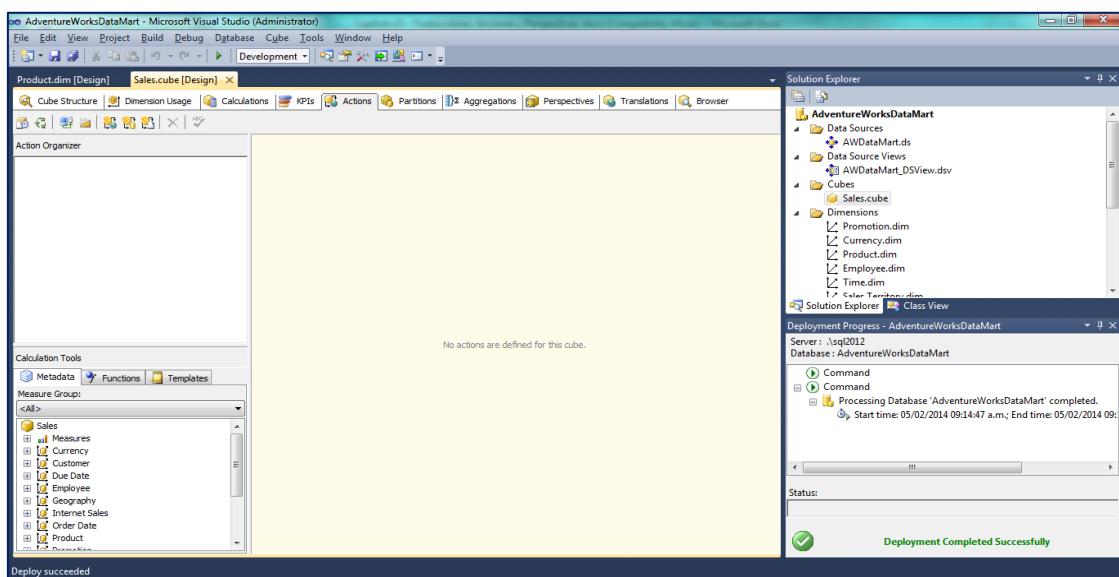
Ejercicio:

Cree Acciones en el cubo Sales.

En este ejercicio, se creará una acción en el cubo “Sales” que permitirá ver las órdenes relacionadas con la medida “Internet Sales Amount”. Cuando el usuario navegue a través de la data del cubo “Sales”, podrá disparar la acción haciendo click derecho sobre cualquier celda que contenga valores de la medida “Internet Sales Amount”.

Realice los siguientes pasos:

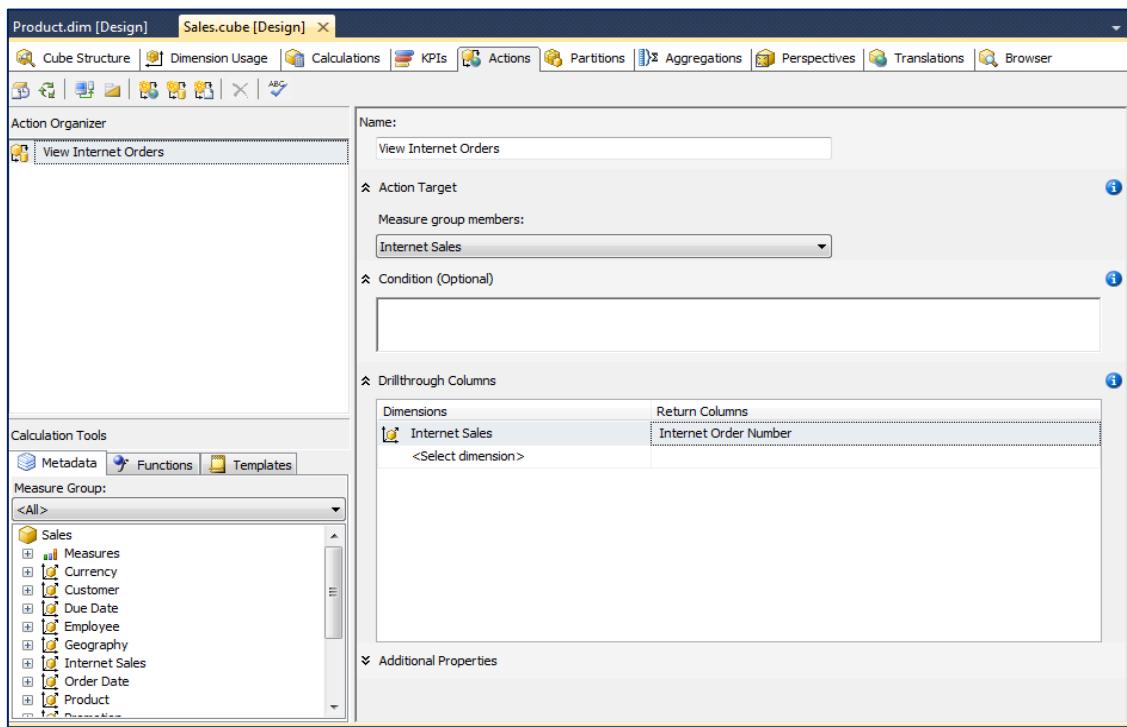
1. En el Solution Explorer, haga doble clic en el cubo “Sales”.
2. En el diseñador de cubos, seleccione el tab Actions (es el quinto empezando por la izquierda). Las acciones se crean en este tab.



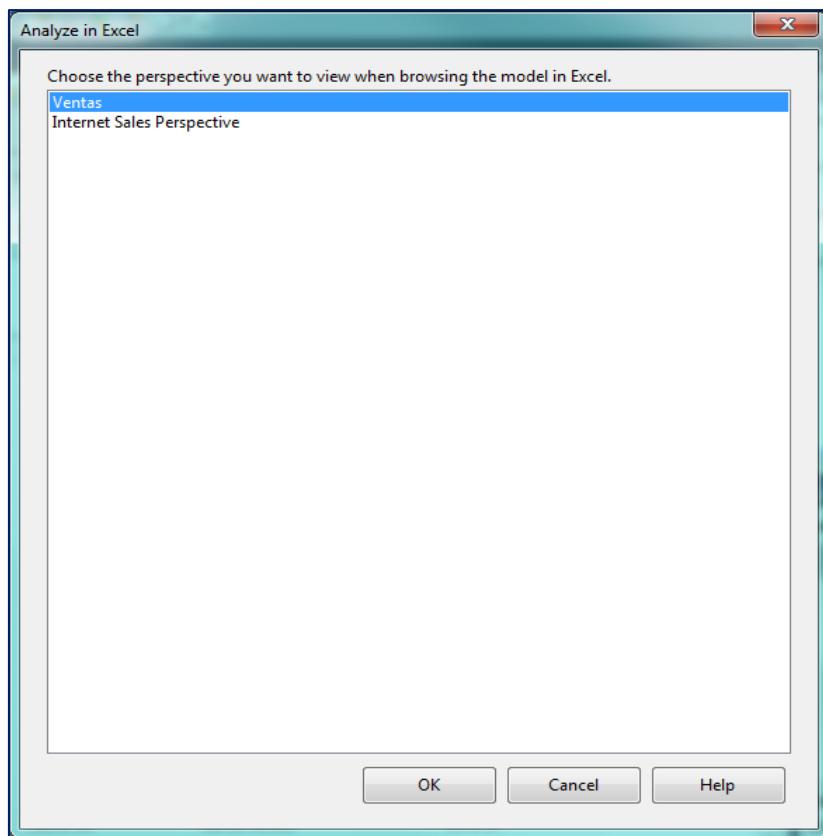
3. En el toolbar del tab Actions, pulse el botón para iniciar la creación de una nueva acción de tipo Drillthrough.
4. Establezca las siguientes propiedades a la acción que acaba de crear:

Propiedad	Valor
Name	View Internet Orders
Measure group members	Seleccione el grupo de medidas “Internet Sales”
Drillthrough Columns	En la columna Dimensions, seleccione la dimensión "Internet Sales". En la columna Returns Columns, seleccione el atributo "Internet Order Number"

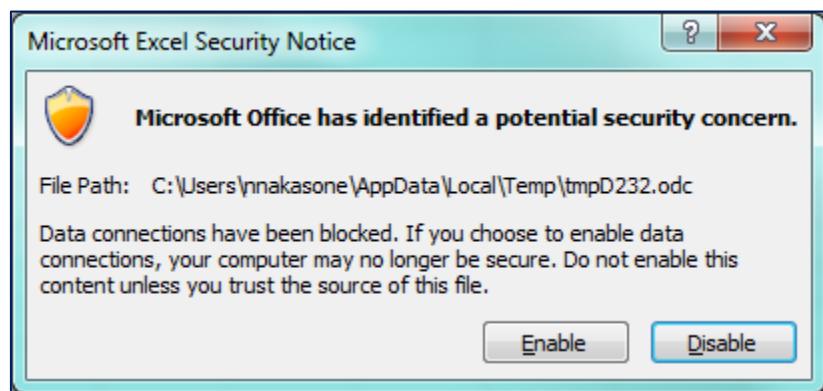
5. Al finalizar, el tab Actions debe tener la siguiente apariencia:



6. Guarde los cambios y despliegue la solución.
7. Consulte al cubo "Sales" utilizando el tab Browser. Asegúrese de que la perspectiva "Sales" está seleccionada. Para visualizar la acción debemos hacerlo desde Excel; para ello, seleccionamos el ícono de Excel y luego escogemos la perspectiva Ventas y le damos clic en el botón Ok.



8. Si a continuación nos muestra una venta de Microsoft Excel Security, haga clic en Enable.



9. Luego del panel derecho de datos, seleccione del grupo de medidas Ventas por Internet, la medida: Monto por Internet.
10. Haga clic derecho, Addional Actions y elija la acción creada View Internet Orders.

11. Luego, mostrará el detalle de todas las órdenes que corresponden a esa medida.

[SInternet Sales].[Internet Order Number]
SO43699
SO43700
SO43702
SO43706
SO43707
SO43711
SO43713
SO43718
SO43719
SO43723
SO43726
SO43728
SO43730
SO43733
SO43734
SO43741
SO43747
SO43748
SO43749

12. Adicione la dimensión Fecha de Orden y coloque en filas la jerarquía Calendar y muestra el detalle de las órdenes para el año 2005.



POBLANDO DATA WAREHOUSE

LOGRO DE LA UNIDAD DE APRENDIZAJE

Al término de la unidad, el alumno puebla un datawarehouse utilizando SQL Server Integration Services.

TEMARIO

3.1 Tema 7 : Herramienta ETL

- 3.1.1 : SQL Data Tools y el diseñador de paquetes
- 3.1.2 : Elementos de SSIS
- 3.1.2.1 : Conexiones y tipos
- 3.1.2.2 : Tasks
- 3.1.2.3 : Control Flow
- 3.1.2.4 : Containers
- 3.1.2.5 : Package

3.2 Tema 8 : Procesos de Carga

- 3.2.1 : Diseño de proceso de carga
- 3.2.1.1 : Mapeo de Datos
- 3.2.1.2 : Estrategia de manejo de errores
- 3.2.1.3 : Estandarización y limpieza de datos
- 3.2.1.4 : Creación y desarrollo de proyectos SSIS

ACTIVIDADES PROPUESTAS

8. Los alumnos identifican el proceso ETL y sus etapas.
9. Los alumnos identifican los elementos de SSIS.
10. Los alumnos identifican conexiones y tipos.
11. Los alumnos identifican y definen procesos de carga

3.1. HERRAMIENTA ETL SSIS

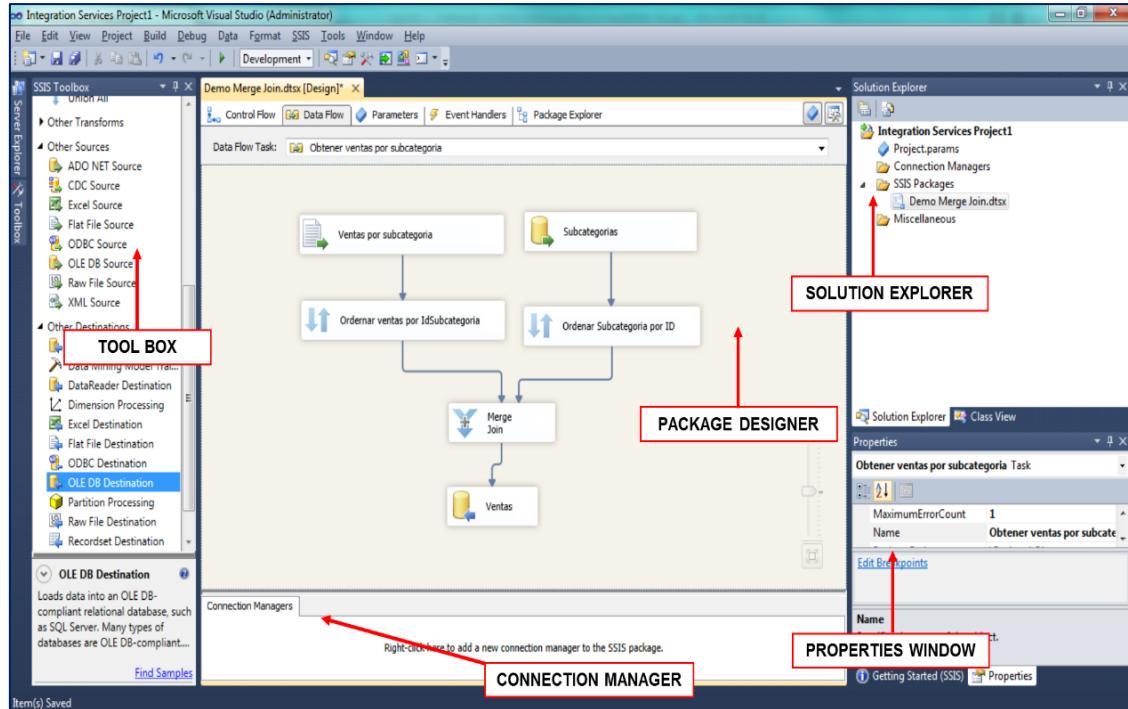
SQL Server Integration Services (SSIS) no constituye la evolución de los Data transformation Services (DTS) de SQL Server 2000. Es una plataforma completamente nueva para implementar procesos ETL de alcance empresarial.

3.1.1. SQL Server Data Tools y el Diseñador de paquetes

SQL Server Data Tools es el nuevo entorno de desarrollo de Business Intelligence de Microsoft basado en la versión de Visual Studio 2013 partner. Este entorno posee varias plantillas para el desarrollo de BI, en particular en esta unidad usaremos la plantilla de Integration Services.

A continuación se muestra los paneles correspondientes al entorno de desarrollo:

- Tool Box: Caja de herramientas que posee las tareas y componente para realizar los procesos ETL
- Diseñado de Paquete: entorno de trabajo donde se diseña el paquete a través tareas y componentes.
- Connection Manager: almacena las cadenas de conexión a las fuentes de datos.
- Solution Explorer: panel que almacena los componentes de la solución: Project Parameters, Connection Manager, Packages, Miscellaneous.
- Properties: Panel para configurar las propiedades de cada elemento de un proyecto Integration Services.



3.1.2. Elementos de SSIS

SSIS está constituido por varios elementos que permiten construir aplicaciones de carga y transformación de datos altamente complejos y flexibles. A continuación se presentan los elementos básicos de SSIS:

- Connection Managers: Definen conexiones hacia los orígenes de datos. SSIS es capaz de conectarse con cualquier origen de datos accesible a través de proveedores .Net y OLE DB.
- Tasks: Una tarea (task) define una unidad de transformación o procesamiento de datos. Las tareas pueden ejecutar diversas acciones: ejecutar sentencias SQL, ejecutar scripts .Net, enviar correos, etc. Un tipo especial de tarea es el Data Flow Task (tarea de flujo de datos) que efectúa transferencias de registros entre varios tipos de orígenes de datos.
- Control flow: Permite definir secuencias lógicas de transferencia de información. Por ejemplo, a través de los flujos de control, se pueden definir bucles que repitan varias veces una transferencia de datos.
- Containers: Las tareas de transformación se pueden agrupar en contenedores (containers). El uso de contenedores hace más sencillo definir la secuencia lógica de ejecución de las tareas.
- Package: Todas las tareas y contenedores se hallan dentro de un package (paquete) de SSIS. Un proyecto SSIS puede contener varios paquetes.

3.1.2.1. Conexiones y tipos

Conexiones y Transformaciones

Los Connection Manager son conexiones privadas o compartidas que usan los paquetes SSIS. Las privadas las encontramos en el connection manager del paquete y las compartidas son definidas en el proyecto.

La tarea Data Flow Task permite efectuar transferencias y transformaciones de datos, desde objetos Data Flow Sources hacia objetos Data Flow Destination. SSIS implementa diversos tipos de transformaciones: conversiones de tipos de datos, uso de tablas de lookup, agregaciones, etc. A través de estas transformaciones, la información puede ser trabajada antes de llegar a su destino. Cada transformación posee columnas de entrada (inputs) y columnas de salida (outputs).

En el siguiente ejercicio, se creará un proyecto SSIS que transferirá las filas de un join entre las tablas Product y ProductSubCategory en la base de datos AdventureWorks, hacia una tabla llamada ResumenProductos, ubicada en la base de datos AWorks_MartDemo. Para esto, se utilizará la transformación Data Flow Task.

Ejercicio:

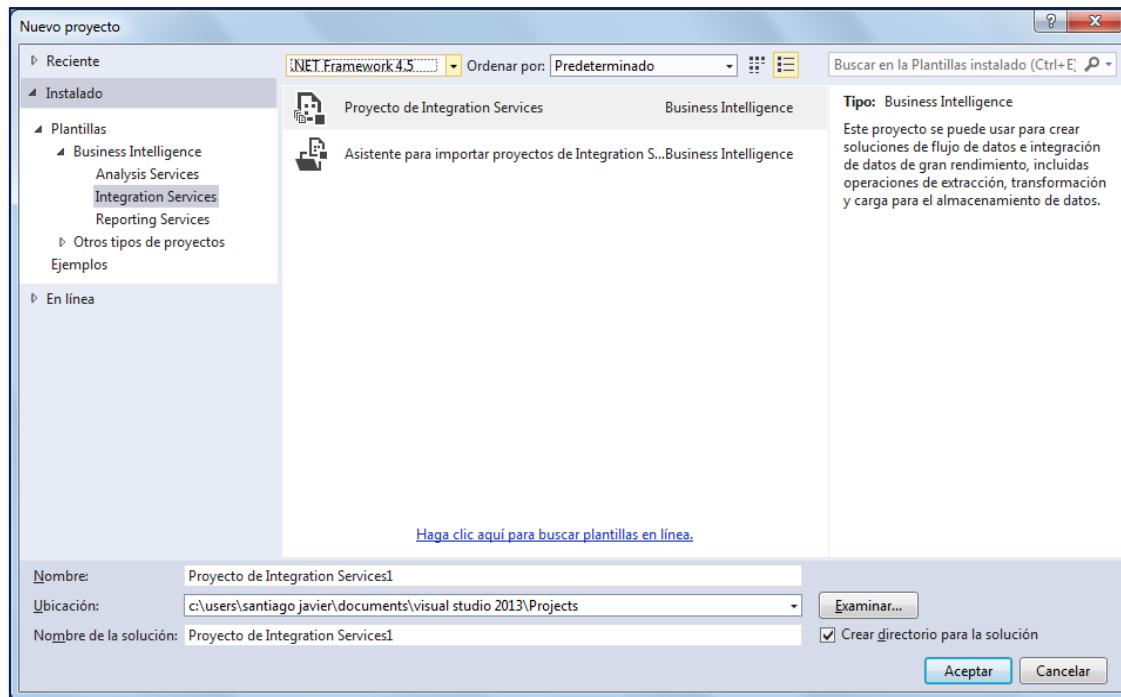
Crear conexiones y transformaciones en un proyecto SQL Server Integration Services.

1. Descargue la base de datos AdventureWorks2014 del siguiente enlace:

<https://msftdbprodsamples.codeplex.com/releases/view/125550>

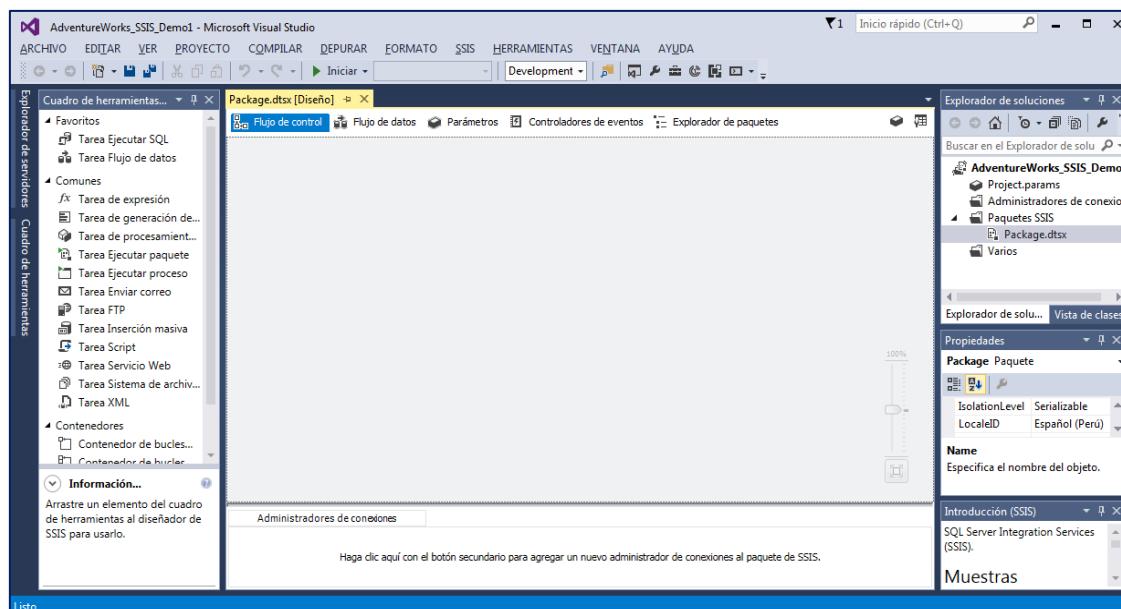
2. Adjuntarlo al servidor de base de datos SQL Server 2014.

3. En el menú Inicio, seleccione All Programs > Microsoft SQL Server 2014 > SQL Server Data Tools. Se abrirá el entorno de desarrollo de .Net.
4. En el menú File, seleccione New y luego Project. Aparecerán los distintos tipos de proyecto que se pueden crear.

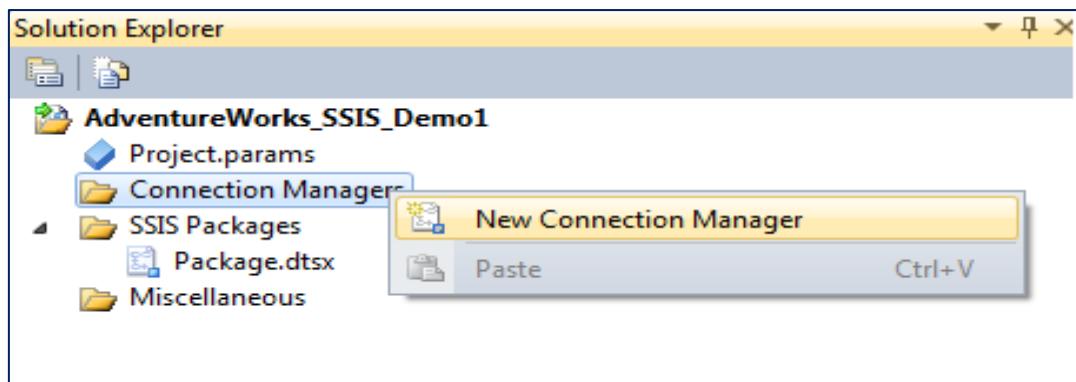


Escoja la plantilla Integration Services Project. En la casilla Location, elija la ruta donde desea crear el proyecto. En la casilla Name, escriba "AdventureWorks_SSIS_Demo1". La ventana debe quedar como se muestra en la figura anterior. Presione el botón OK para crear el proyecto.

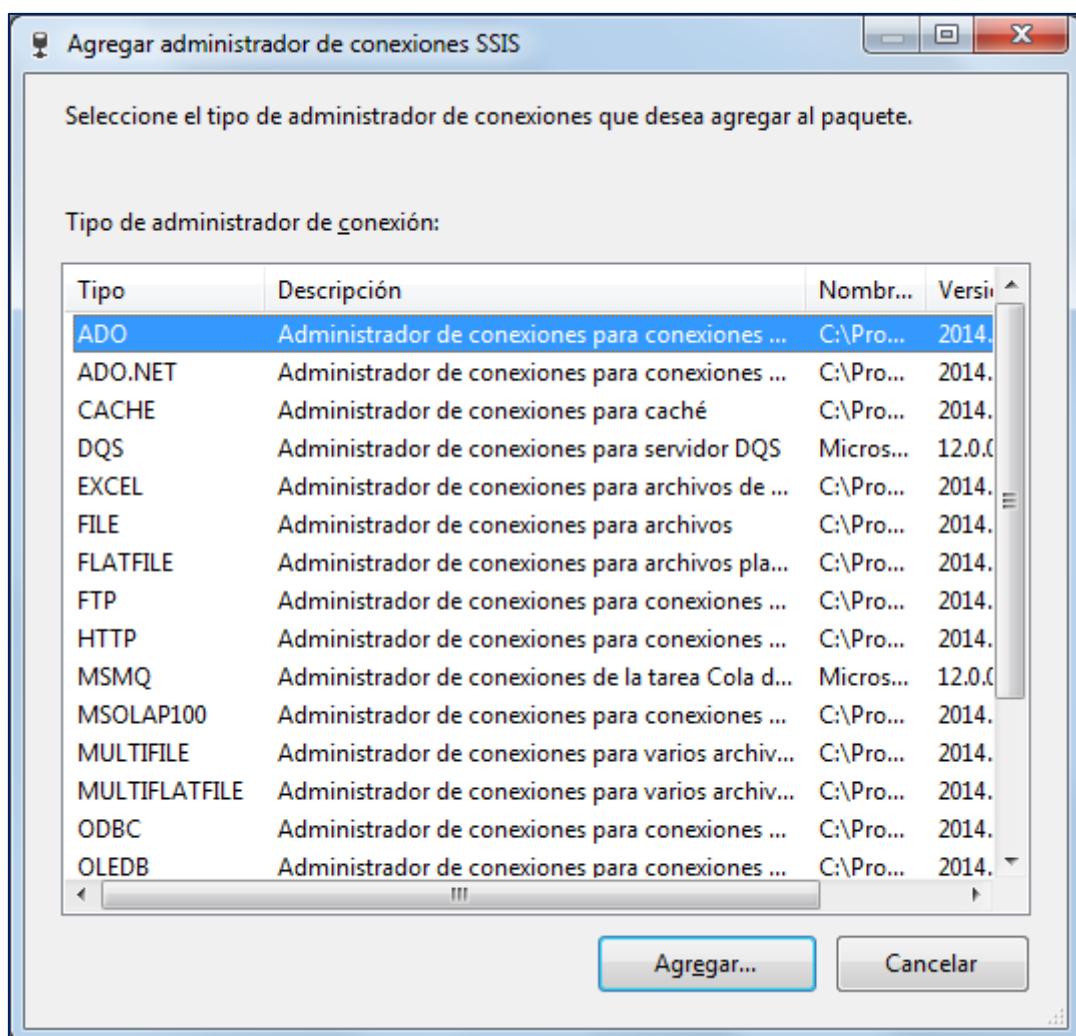
5. Aparecerá el entorno de desarrollo de proyectos de Integration Services. En la ventana del Solution Explorer, observe los elementos constituyentes del proyecto: Project.Params, Connection Managers, SSIS Packages y Miscellaneous.



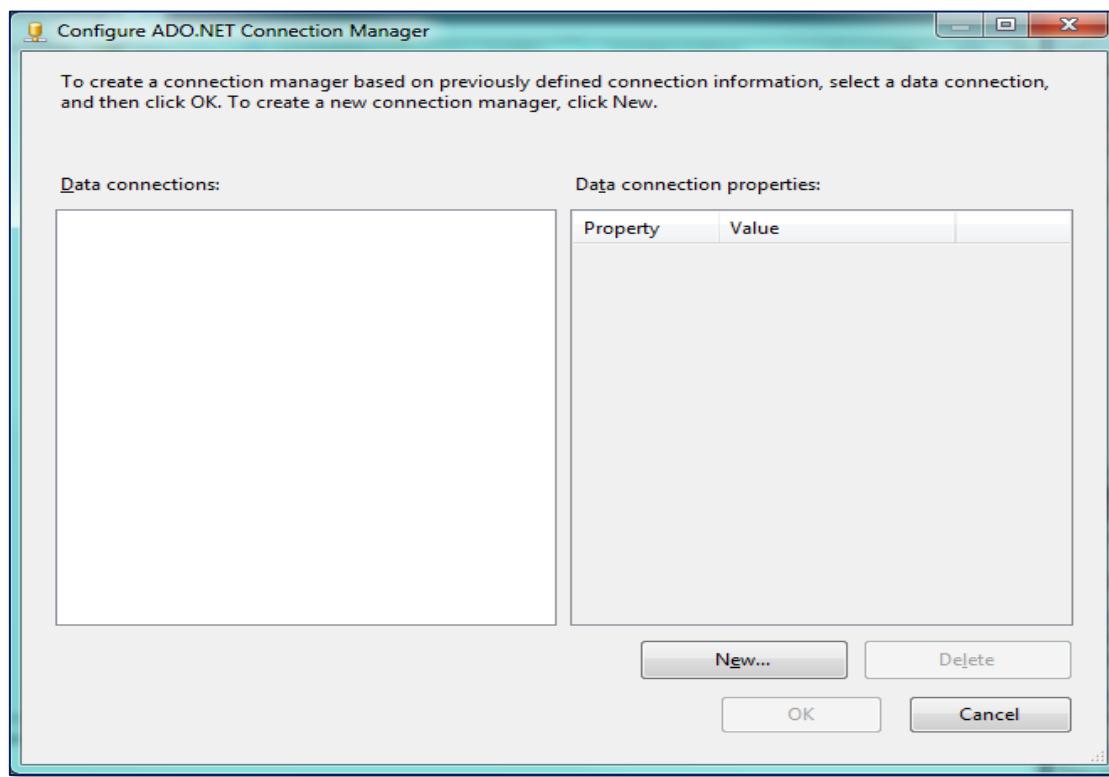
6. El primer paso es la definición de un Connection Managers. En el Solution Explorer, seleccione el nodo Connection Managers, dé clic derecho y seleccione New Connection Manager.



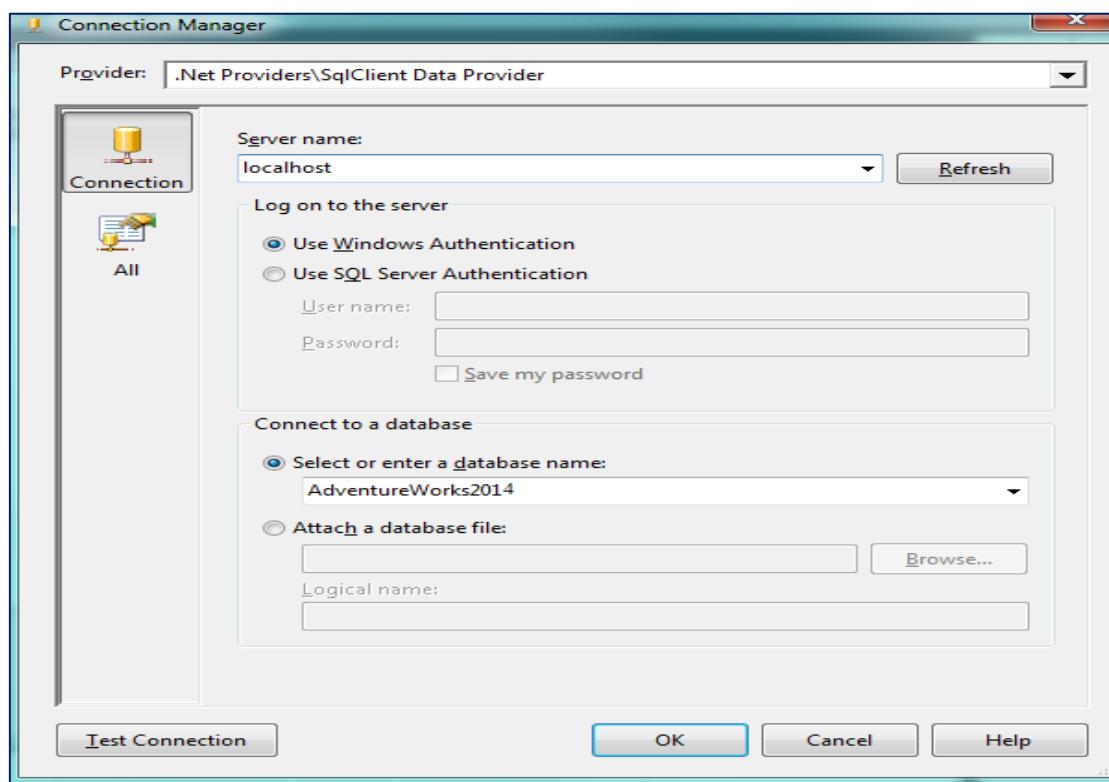
7. Aparecerá la pantalla Add SSIS Connection Manager para la definición de orígenes de datos. Elija ADO.Net como tipo de conexión y haga clic en el botón Add.



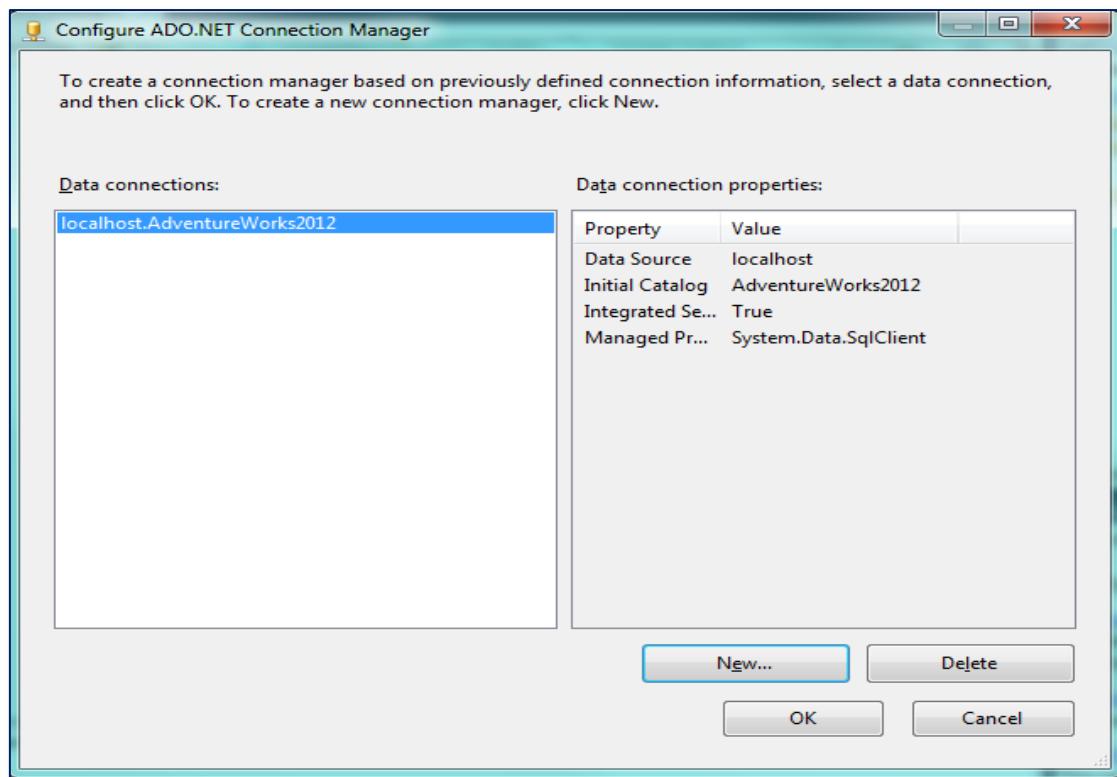
8. En la siguiente pantalla, haga clic en el botón New para definir la cadena de conexión:



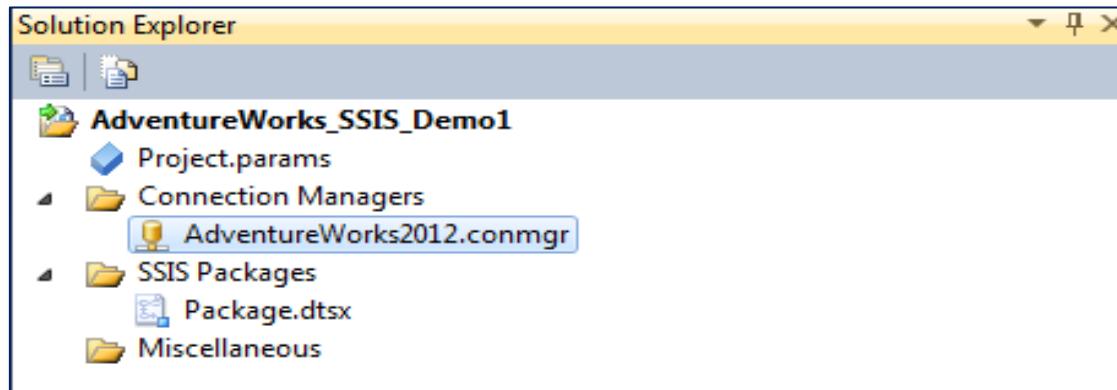
9. Aparecerá la ventana del Connection Manager. Elija el servidor local y la base de datos AdventureWorks2014 como se muestra en la siguiente figura:



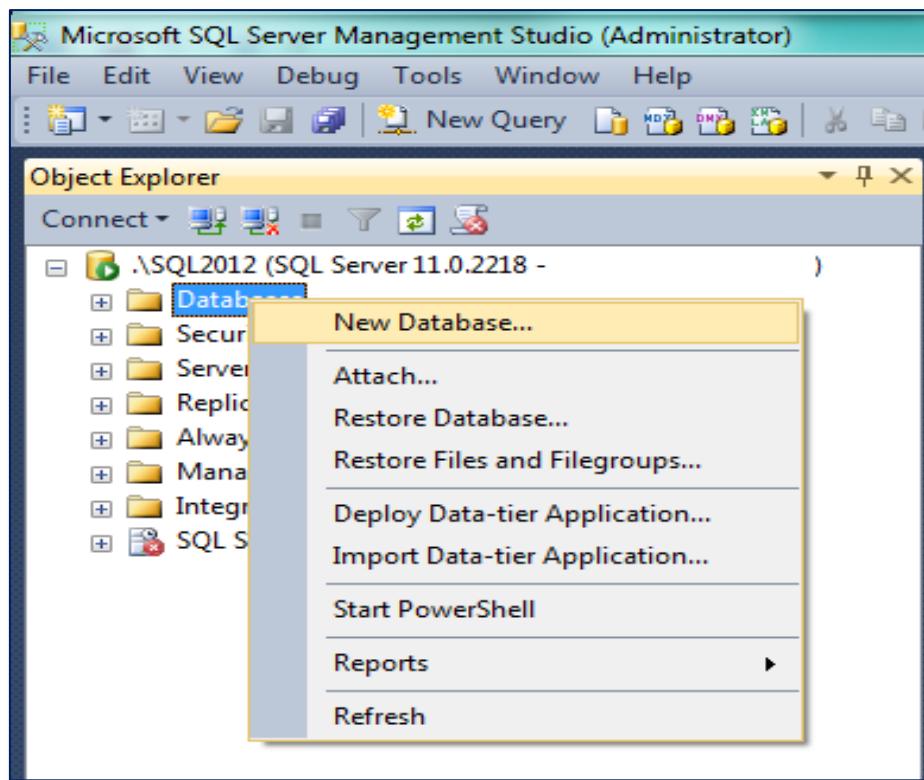
10. Presione el botón OK y nuevamente OK para finalizar el asistente:



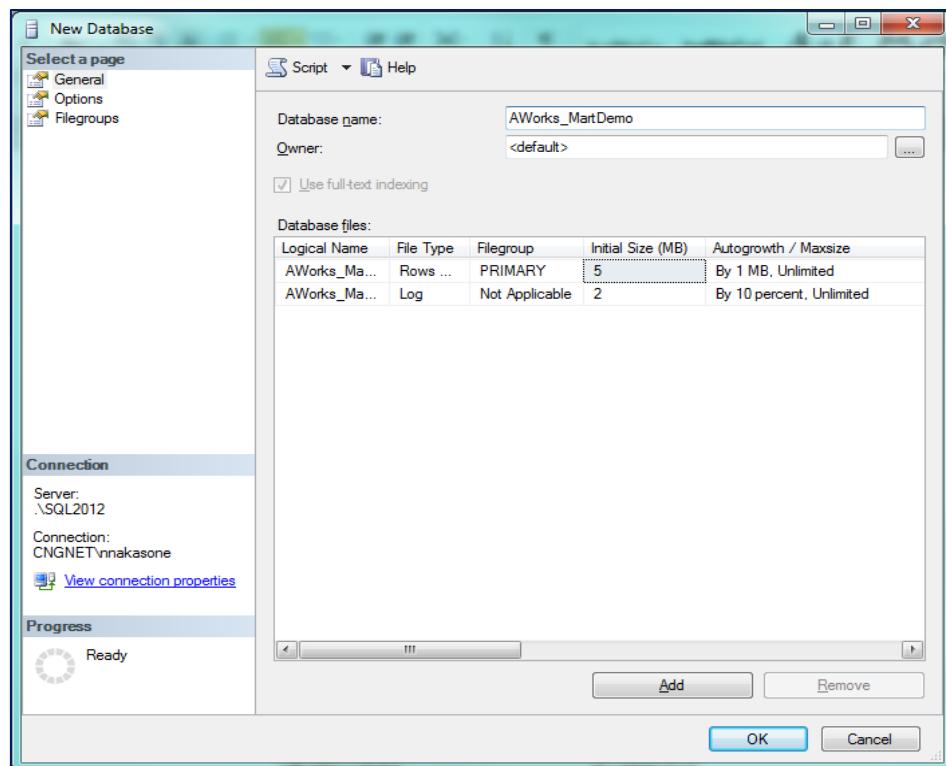
11. Luego, renombre el connection manager a AdventureWorks2014.



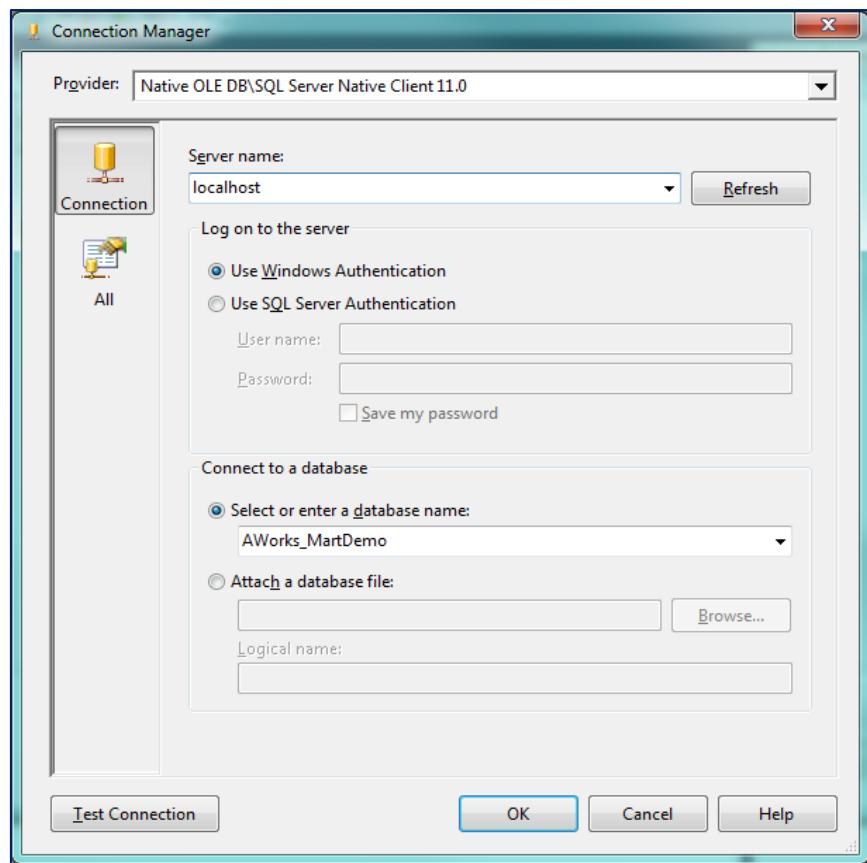
12. La base de datos AdventureWorks2014 será utilizada como origen de datos. En este ejercicio, el destino de datos será una nueva base de datos llamada AWorks_MartDemo. Esta nueva base de datos será creada a través del SQL Server Management Studio. Ingrese al SSMS y haga clic derecho en la carpeta Databases, New Databases.



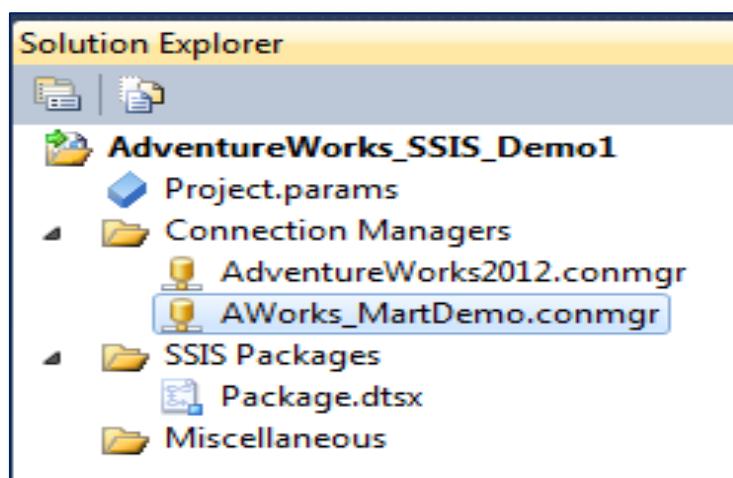
13. En la ventana de creación de bases de datos, ingrese como nombre AWorks_MartDemo y haga clic en Ok.



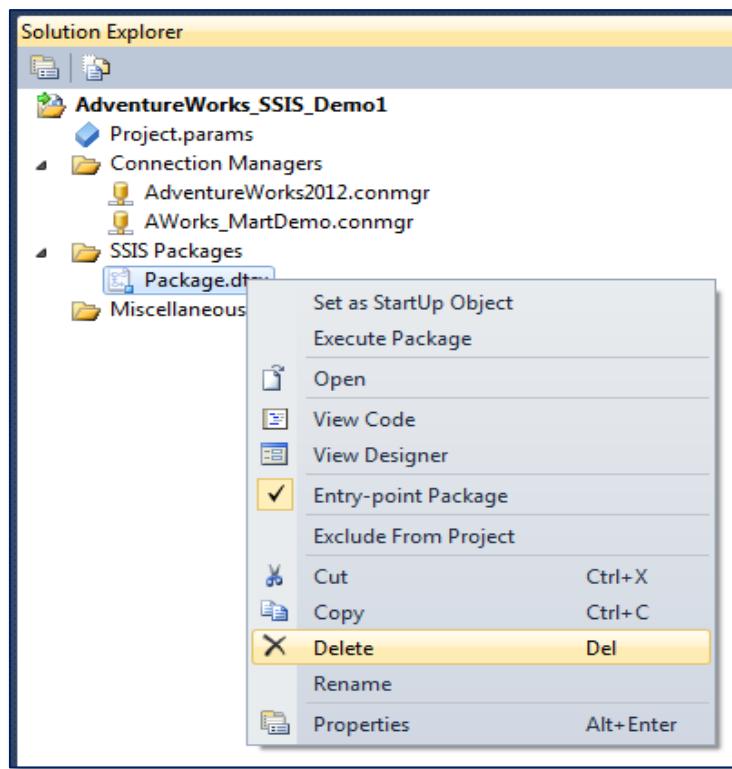
14. Repita los pasos del (6) al (11) para crear un nuevo Connection Manager llamado AWorksMartDemo, dirigido hacia la base de datos AWorksMartDemo. Para este data source, elija el tipo OLEDB.



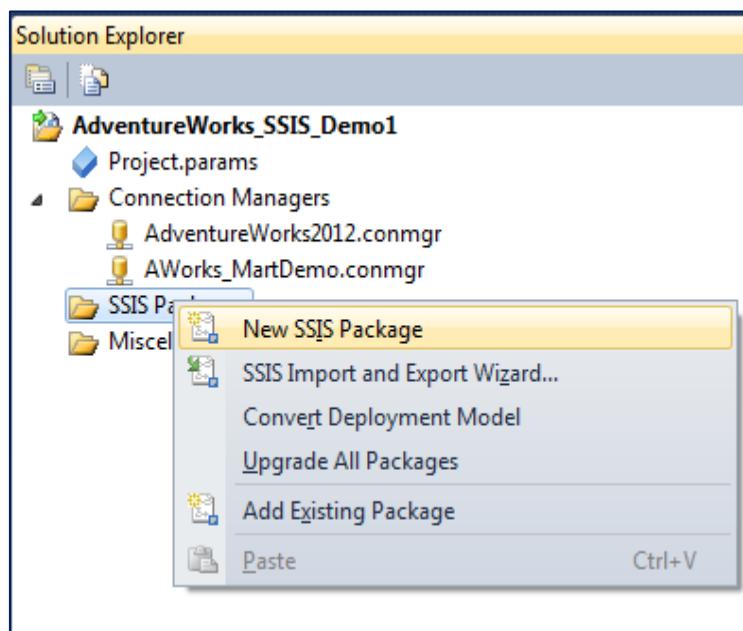
Finalmente, la ventana del Solution Explorer debe quedar de la siguiente manera:



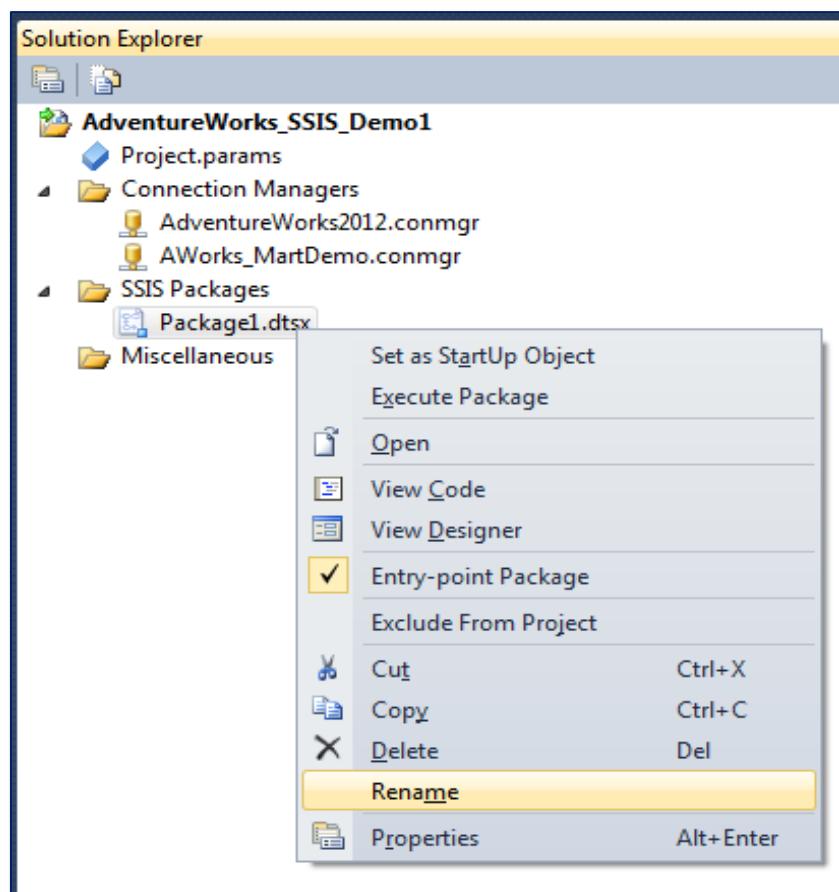
15. El siguiente paso es la definición del paquete de Integration Services para implementar la lógica de la transformación de datos. Los paquetes de Integration Services se ubican en el nodo SSIS Packages del Solution Explorer. Obsérvese que en el proyecto se ha agregado automáticamente un paquete llamado Package.dtsx. Haga clic derecho sobre el paquete y seleccione la opción Delete.



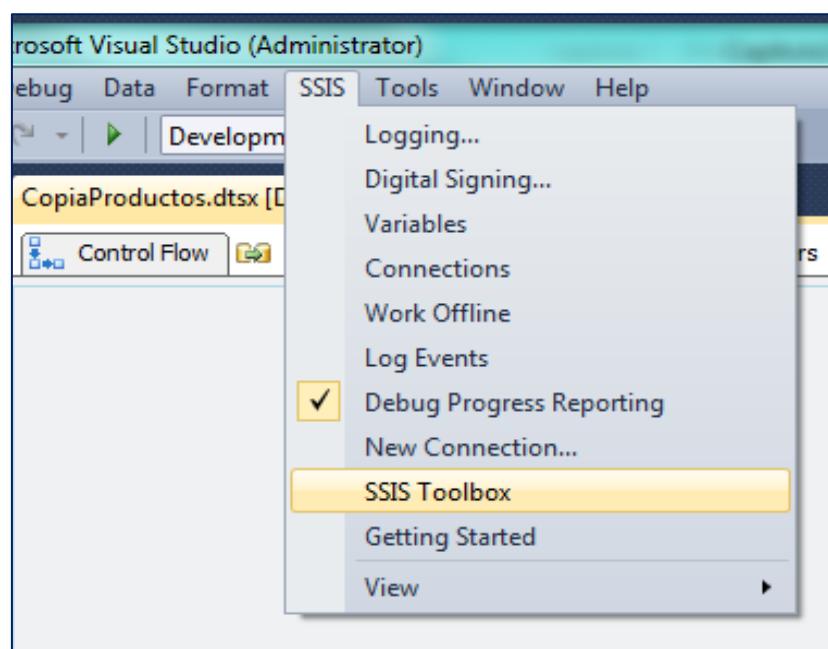
16. Pulse el botón OK en la ventana de confirmación para confirmar la eliminación del paquete.
17. A continuación se creará el primer paquete SSIS. Para esto, seleccione el nodo SSIS Packages en el Solution Explorer, haga clic derecho y seleccione la opción New SSIS Package.



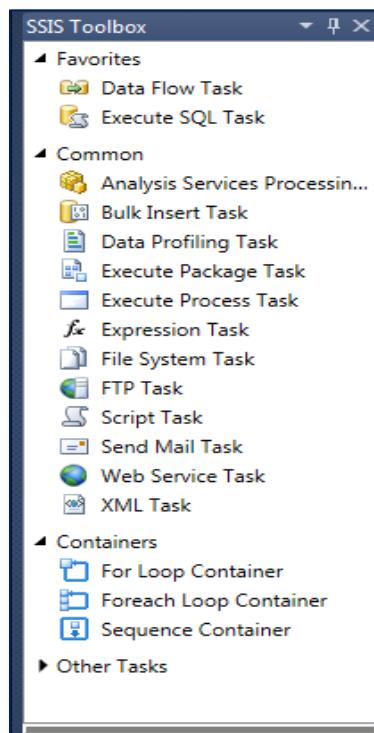
18. Obsérvese que se ha creado un nuevo paquete llamado Package1.dtsx. Para renombrarlo, haga clic derecho sobre el paquete en el Solution Explorer y seleccione la opción Rename.



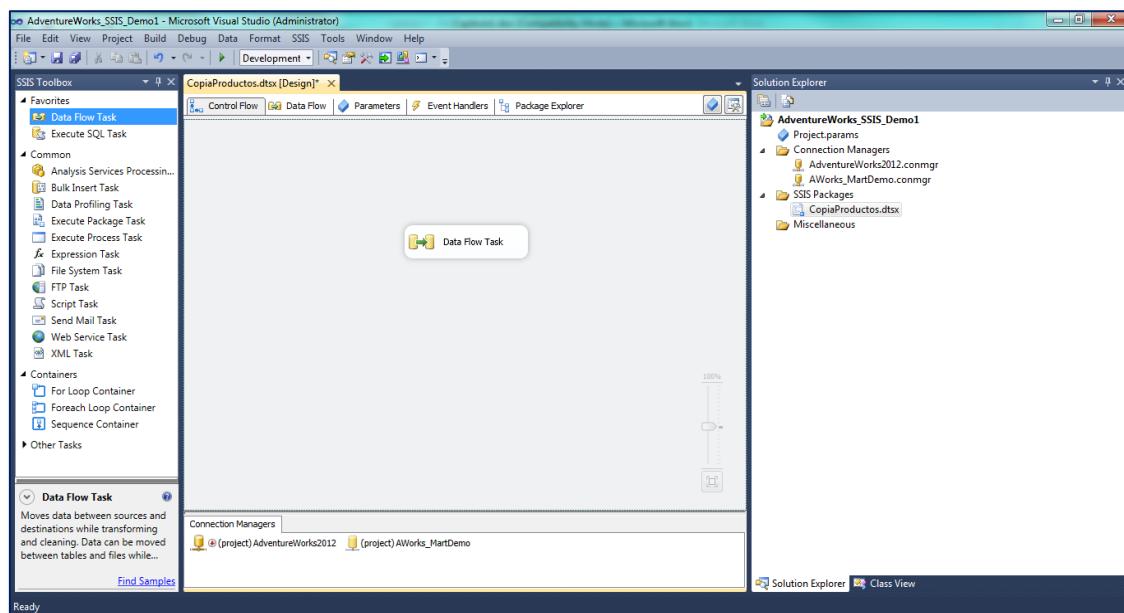
19. Renombre el paquete como “CopiaProductos.dtsx”, y presione la tecla Enter.
20. El siguiente paso es definir las tareas de transformación. En Integration Services, estas tareas están representadas en los Tasks, cada uno de los cuales puede efectuar una labor específica de procesamiento o transformación de datos. Estas tareas pueden visualizarse a través de la caja de herramientas (Toolbox) del entorno de desarrollo. Para visualizar el Toolbox, expanda el menú SSIS, y seleccione la opción SSIS Toolbox.



21. Se abrirá el Toolbox de SQL Server Integration Services. La siguiente figura muestra una vista de la caja de herramientas:

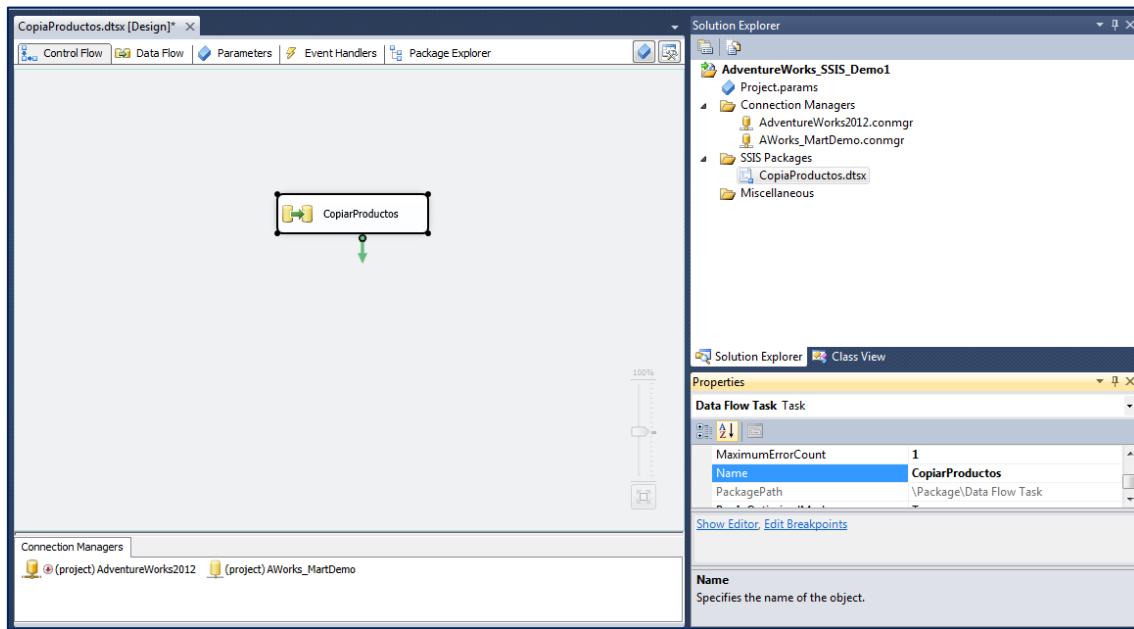


22. En el toolbox, seleccione la tarea Data Flow Task y arrástrela hacia la ventana de diseño del paquete (en la parte central del entorno de desarrollo). Se creará una tarea llamada Data Flow Task. El entorno de desarrollo quedará de la siguiente manera:

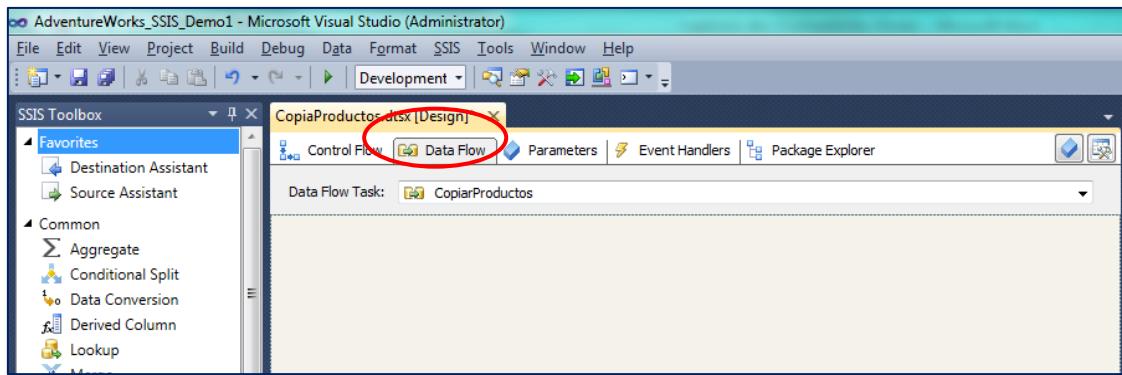


23. A continuación, se renombrará la tarea de transformación. Primero, seleccione la tarea “Data Flow Task” en el diseñador del paquete, y cambie la propiedad Name a “Copiar productos” en la ventana de propiedades (para abrir la ventana de propiedades, expande el menú View y seleccione la opción Properties Window. La

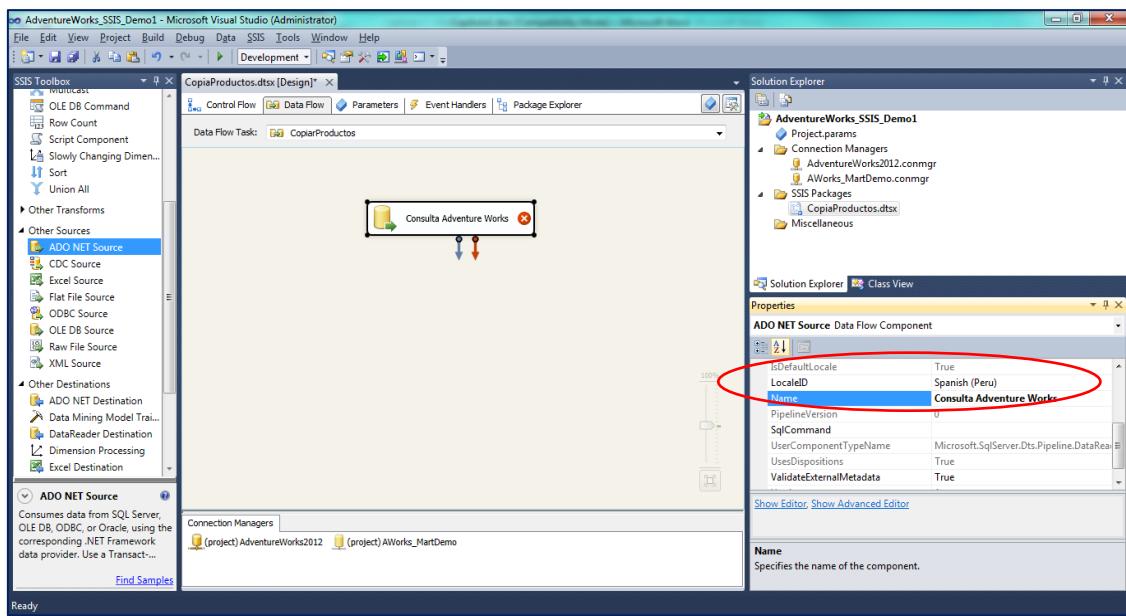
ventana de propiedades se abrirá en la parte inferior derecha, debajo del Solution Explorer).



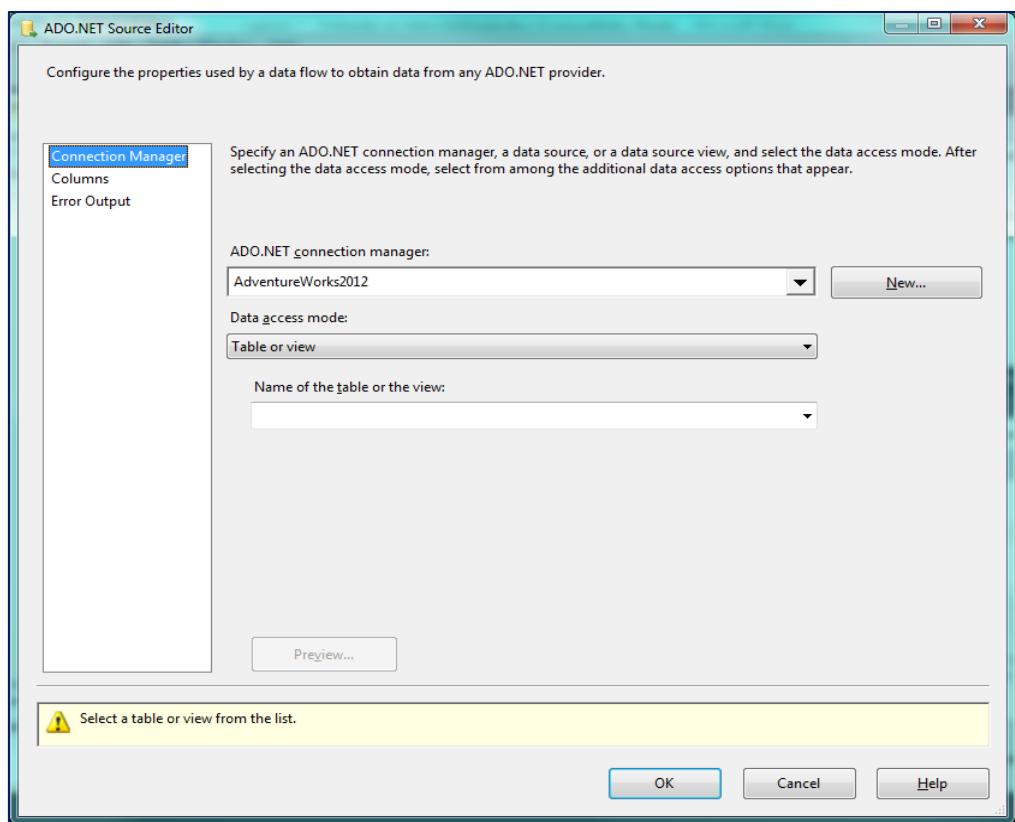
24. Cada tarea de flujo de datos consiste en varias transferencias de información entre orígenes de datos. A continuación, se definirán los flujos de información al interior de la tarea “Copiar productos”. Para iniciar el proceso, haga un doble clic sobre la tarea “Copiar productos”. Se abrirá en el entorno de desarrollo la pestaña “Data Flow”.



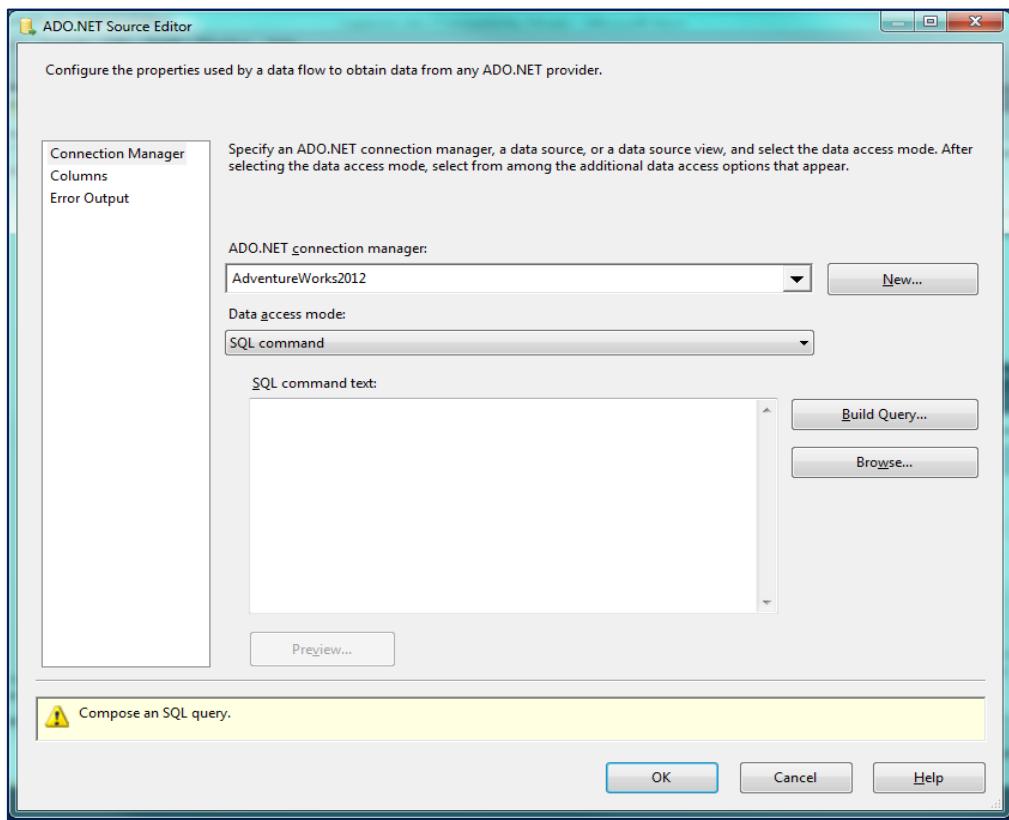
25. En el Toolbox, expanda la pestaña Other Sources, seleccione el ítem ADO Net Source y arrástrelo sobre la ventana de diseño. Se creará un nuevo origen de registros llamado “ADO NET Source”. Selecciónelo y, en la ventana de propiedades, cambie su propiedad Name a “Consulta Adventure Works”. La pantalla debe quedar de la siguiente manera:



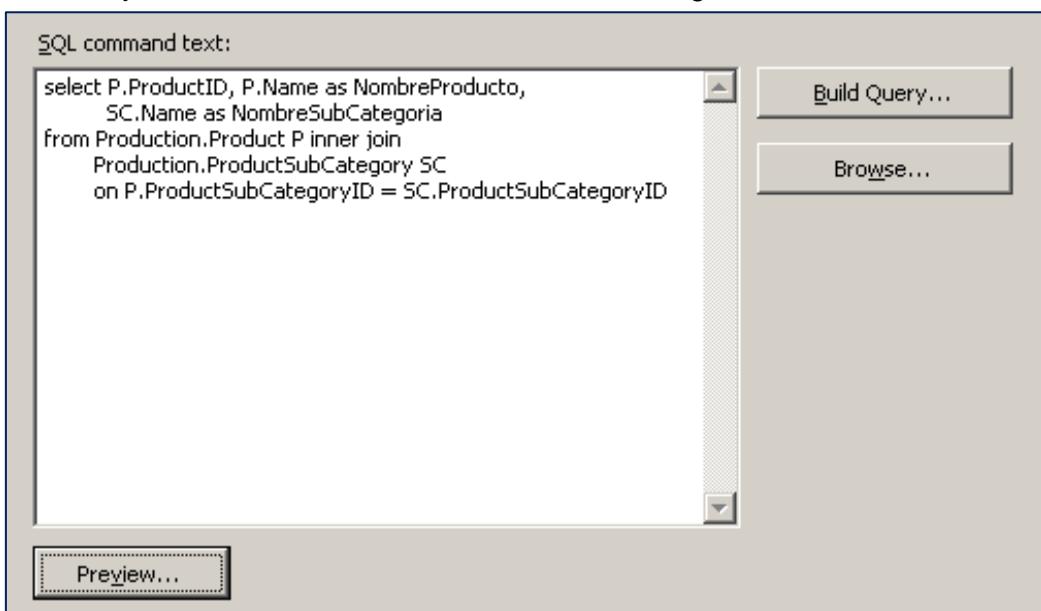
26. Haga doble clic sobre el origen “Consulta Adventure Works”. Aparecerá el editor de propiedades del ADO.NET Source. En la opción ADO.NET connection managers, seleccione el data source Adventure Works2012, como muestra la figura.



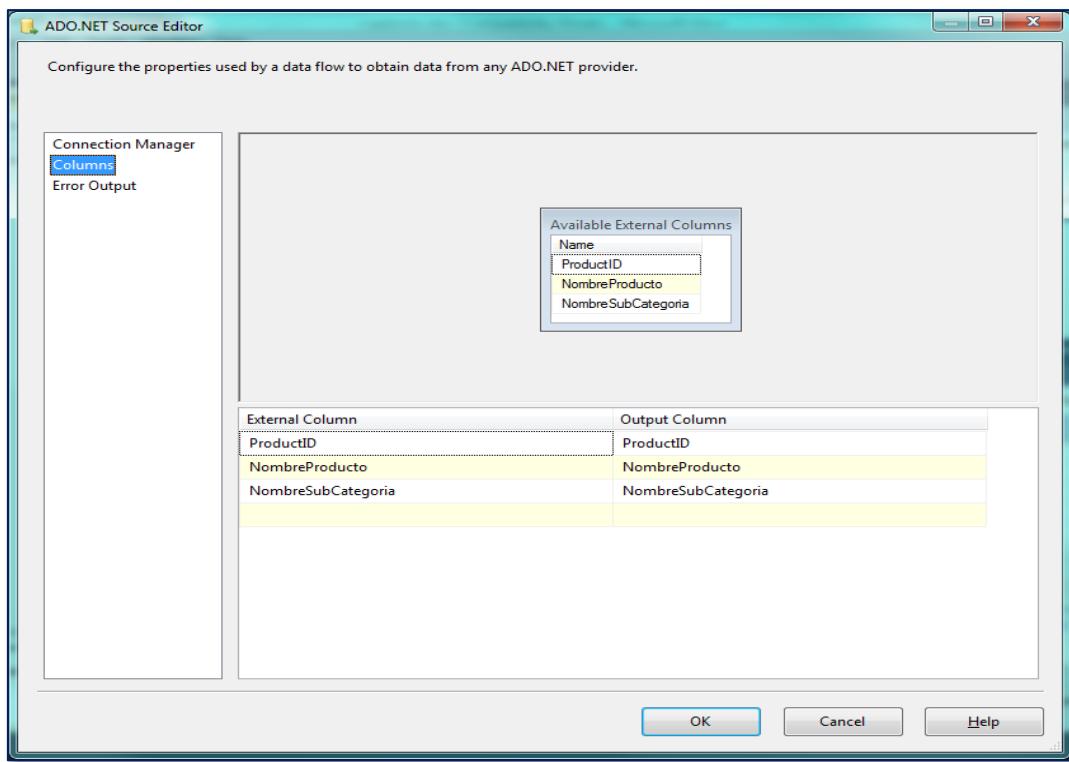
27. En la opción Data Access mode seleccione SQL command como muestra la figura.



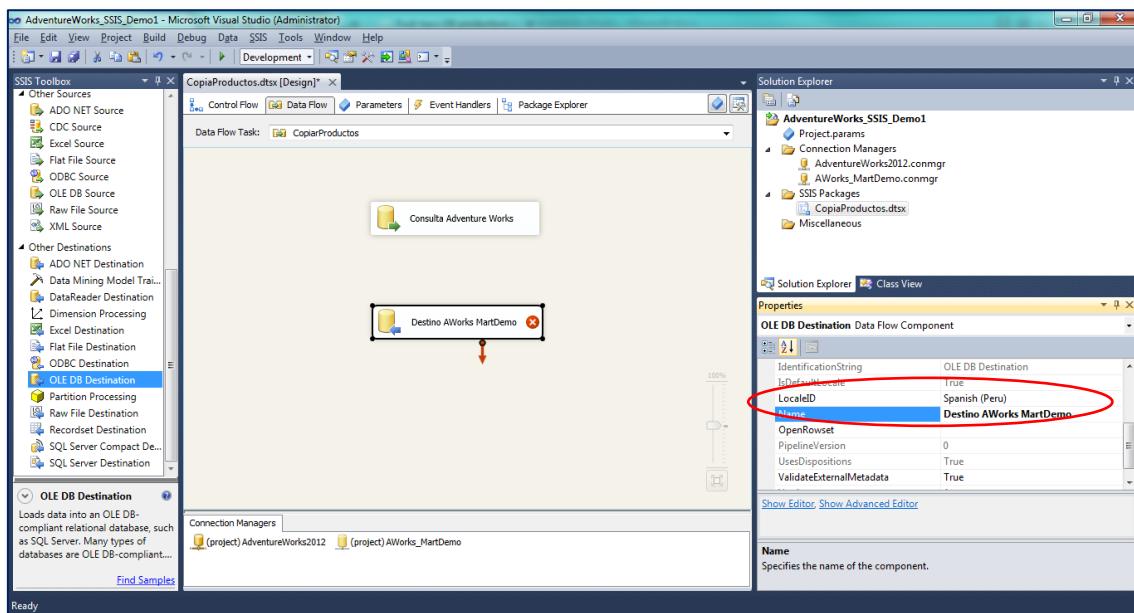
28. En la caja de texto SQL command text, escriba la siguiente sentencia:



29. Al lado izquierdo del panel, seleccione la pestaña Columns, observe que las columnas de la consulta han sido detectadas:



30. Pulse el botón “OK”. Con estos pasos, se ha completado la creación de un origen de registros, basado en una consulta SQL.
31. En el Toolbox, expanda la pestaña Other Destinations, elija el ítem OLE DB Destination y arrástrelo sobre la ventana de diseño. Se creará un nuevo destino de registros llamado “OLE DB Destination”. Selecciónelo y, en la ventana de propiedades, cambie su propiedad Name a “Destino AWorks MartDemo”. La pantalla debe quedar de la siguiente manera:

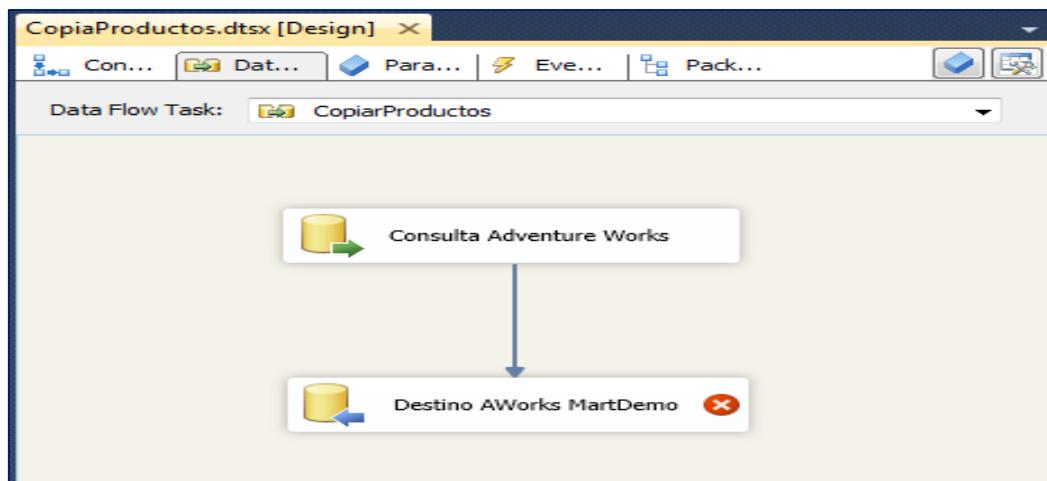


32. Seleccione el origen “Consulta Adventure Works”. Observe que aparece una flecha azul y una flecha roja saliendo del origen de registros. La flecha verde representa el flujo de datos que se produce en caso de que la lectura de registros transcurra

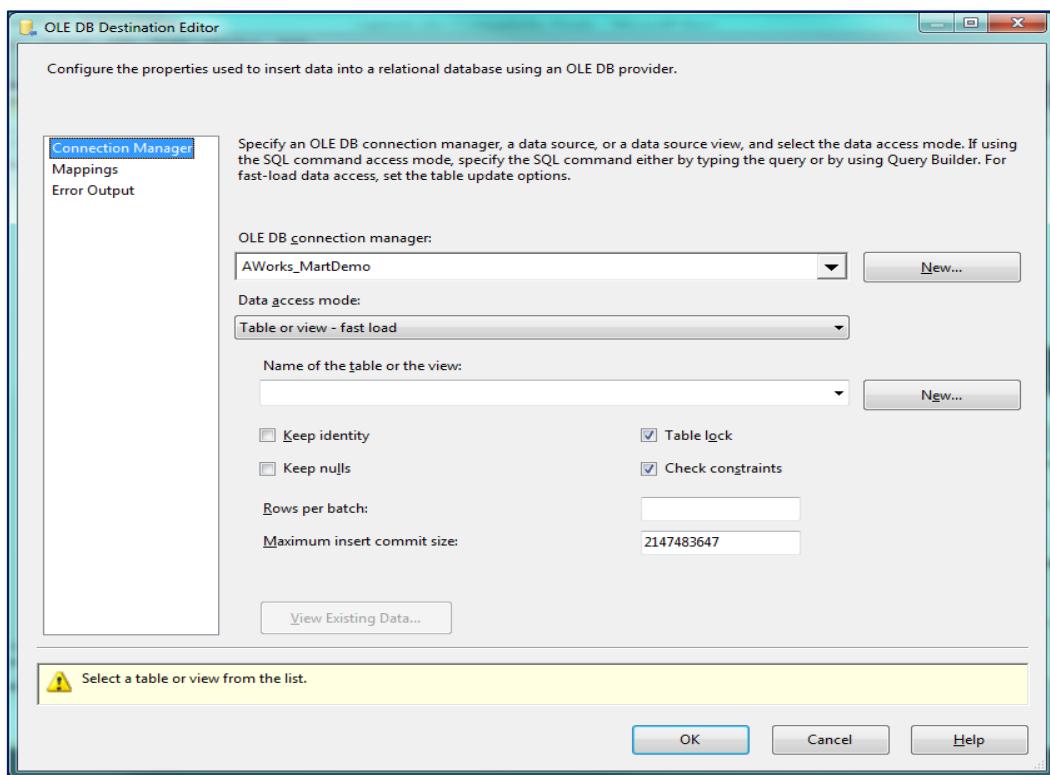
sin problemas; la flecha roja representa el flujo que tendrá lugar ante un error en la lectura de los registros del origen.



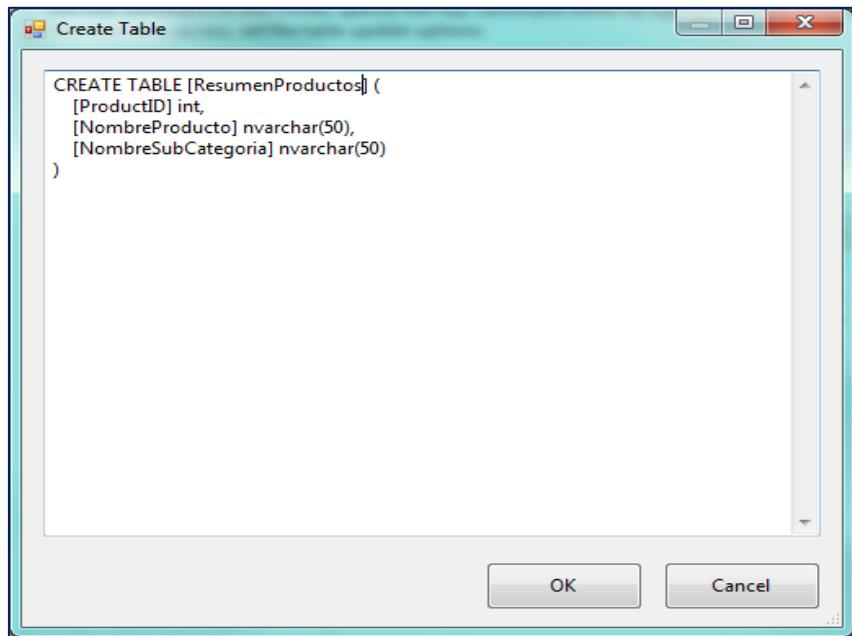
33. Presione la flecha verde que sale de “Consulta Adventure Works”, y arrástrela con el mouse sobre el destino “Destino AWorks MartDemo”. Esta acción define el flujo de los datos entre el origen y el destino de registros. Al finalizar, la ventana debe mostrarse de la siguiente forma:



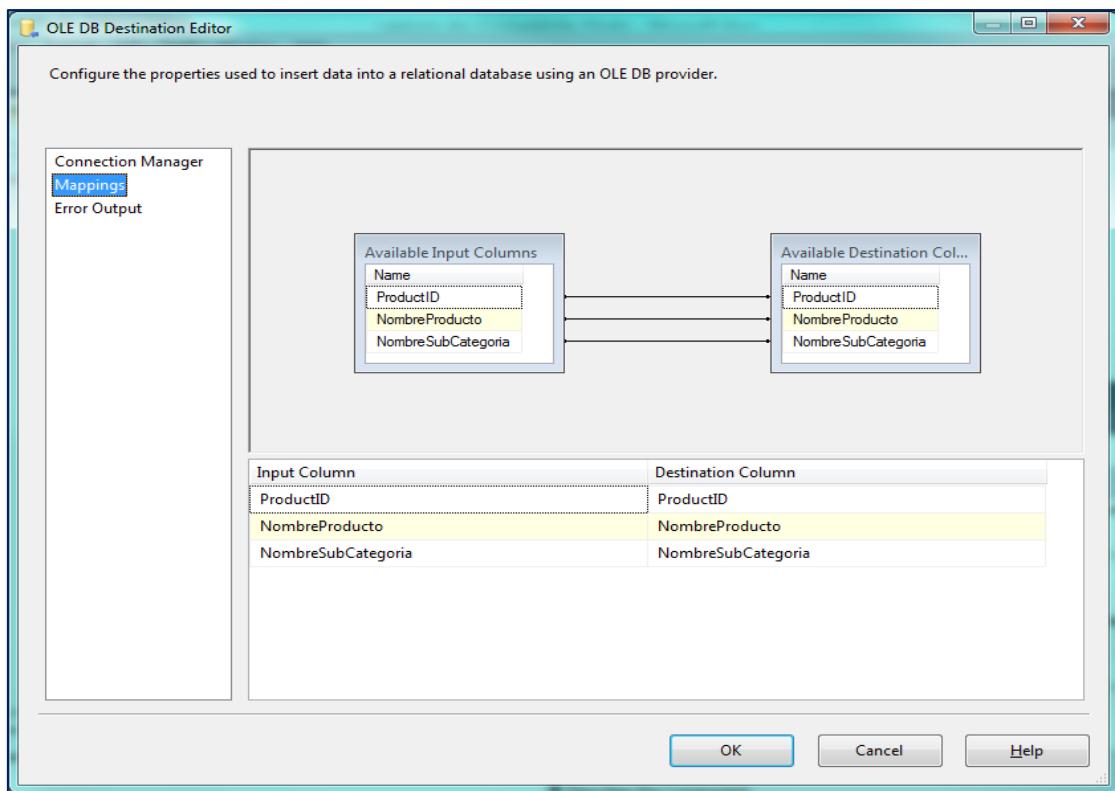
34. Haga doble clic sobre el destino “Destino AWorks MartDemo”. Se abrirá el editor de propiedades del destino de registros. La ventana debe quedar de la siguiente manera:



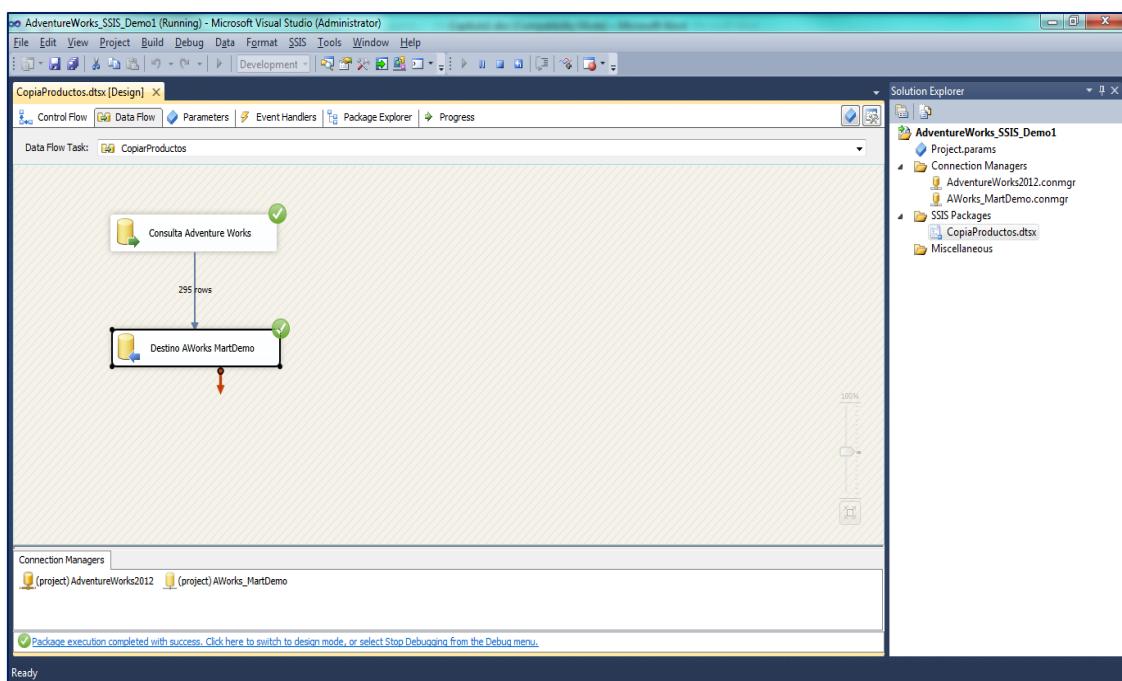
35. A continuación, se creará una tabla en el destino AWorks MartDemo. En la propiedad Name of the table or the view, pulse el botón New. Se abrirá una ventana que permite escribir una sentencia CREATE TABLE. Modifique la sentencia de la siguiente manera:



36. Pulse el botón OK para crear la tabla. A continuación, seleccione la opción Mappings en el lado izquierdo del editor de propiedades del destino de registros. Esto generará automáticamente los mapeos de columnas entre el origen (la consulta SQL escrita en el paso 28) y el destino de registros (la tabla ResumenProductos).



37. Pulse el botón OK para aceptar los cambios y finalizar la configuración del destino de registros.
38. Para depurar el paquete, haga clic en el botón en la barra de herramientas del entorno de desarrollo, o seleccione la opción Start Debugging en el menú Debug. Al finalizar la ejecución, el entorno de desarrollo mostrará la transferencia de datos de la siguiente forma:



39. Para finalizar la depuración, presione el botón  en la barra de herramientas del entorno de desarrollo, o seleccione la opción Stop Debugging en el menú Debug.

3.1.2.2. Tasks

Definir Precedencias

En el ejercicio anterior, se creó una tarea de flujo de datos para transferir datos hacia la tabla ResumenProductos. Es posible que, antes de efectuar la transferencia de los registros, se desee limpiar esta tabla a través de una sentencia DELETE o TRUNCATE. Para ello, SSIS tiene una tarea llamada Execute SQL Task, a través de la cual se puede ejecutar un batch SQL. Bastará con agregar esta tarea al paquete, y escribir en ella la sentencia SQL que elimine la información de la tabla ResumenProductos.

Sin embargo, al ejecutar el paquete, es posible que la tarea de eliminación de los registros se efectúe después de la transferencia de registros. Como resultado, al finalizar el paquete, la tabla ResumenProductos estará vacía. Por tanto, es necesario implementar algún mecanismo de precedencia que garantice que la tarea SQL se efectuará antes de la transferencia de datos.

SSIS permite conectar dos tareas a través de una precedencia. De esta forma, se garantiza que la segunda tarea se efectuará después de la primera. A través de las precedencias, se puede determinar el orden en que se efectuarán las transformaciones y tareas. Los principales tipos de precedencia son los siguientes:

- On Success: La tarea o transformación destino se efectúa sólo si la transformación origen termina exitosamente.
- On Failure: La tarea o transformación destino se efectúa sólo si la transformación origen falla.
- On Completion: La tarea o transformación destino se efectúa al terminar la transformación origen, sin importar si acabó exitosamente o si produjo un error.

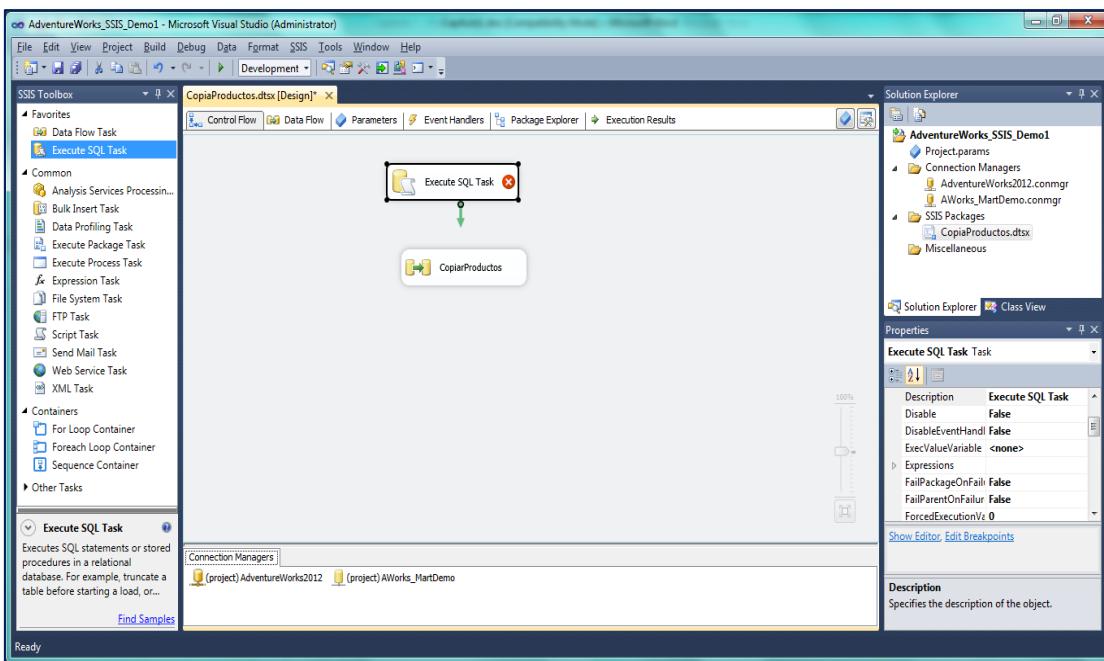
Es posible, también, configurar una precedencia para evaluar una expresión, a través de la cual se verifique el valor de una variable. De esta manera, si la expresión devuelve True, la tarea destino se efectuará.

Ejercicio:

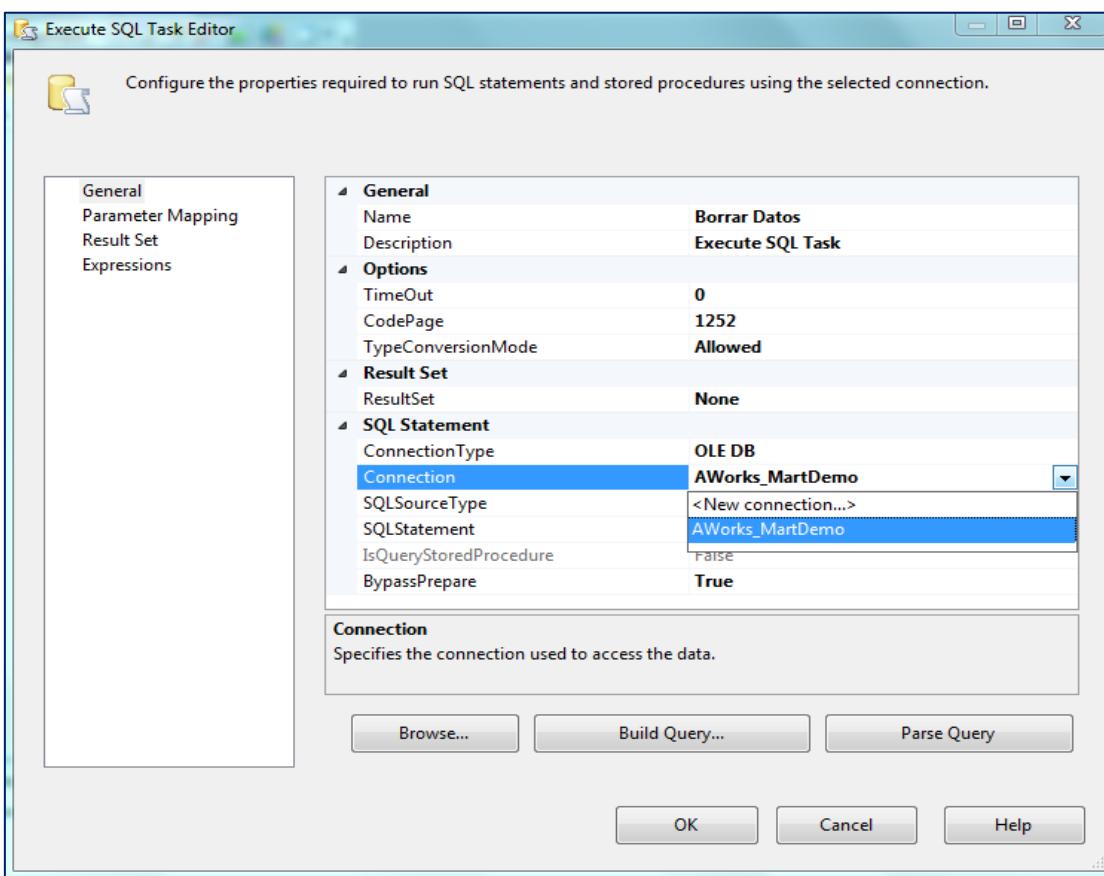
Definir precedencias en un proyecto de SQL Server Integration Services.

En el siguiente ejercicio, se implementará una tarea Execute SQL Task que limpiará la tabla Resumen Productos y además, estableceremos precedencias con el Data Flow Task “Copiar productos”.

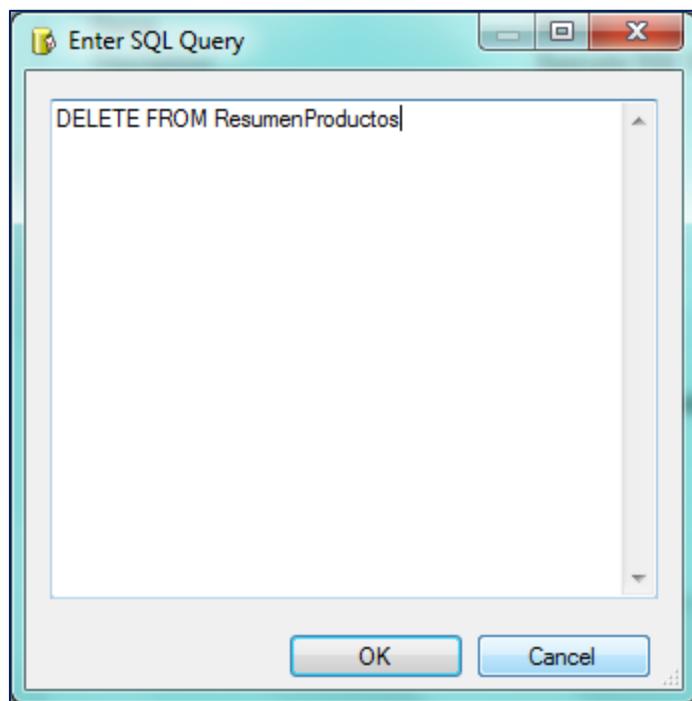
1. En el entorno de desarrollo, en el paquete “CopiaProductos.dtsx”, se debe regresar a la pestaña Control Flow. En grupo Favorites del Toolbox, seleccione el ítem Execute SQL Task, y arrástrelo sobre la ventana de diseño. La pantalla debe quedar de la siguiente manera:



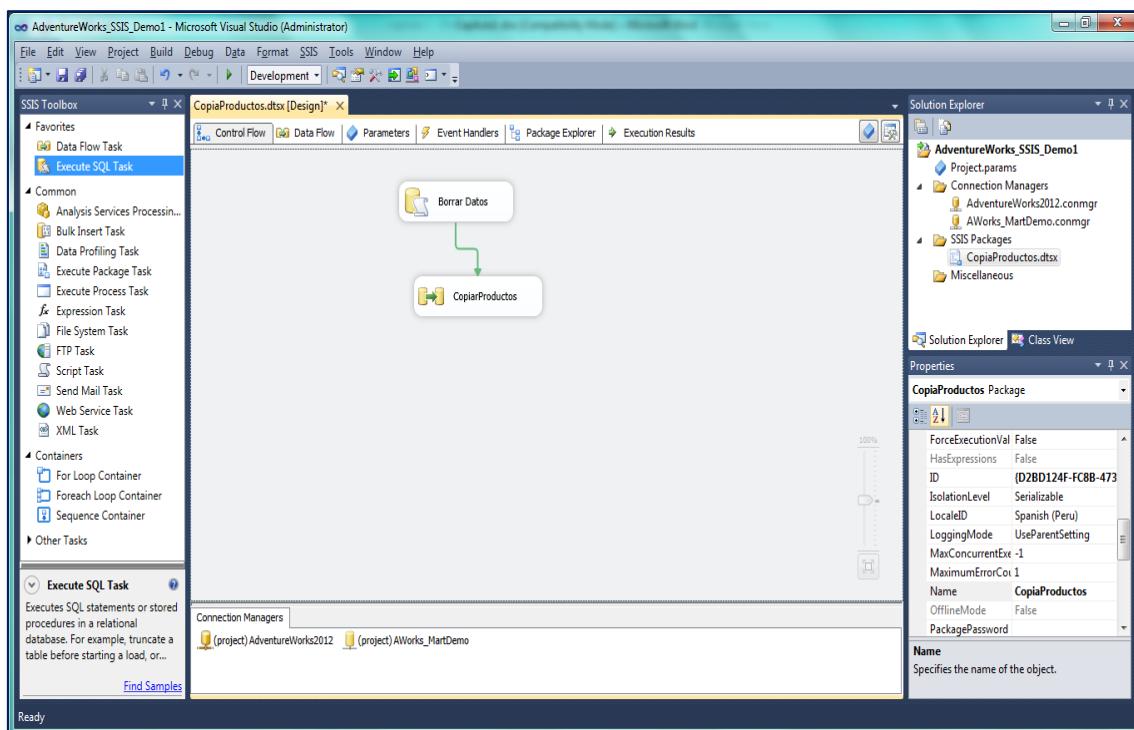
2. En la ventana de propiedades, seleccione la propiedad Name de la tarea que se acaba de crear, y cámbiela por “Borrar Datos”.
3. Haga doble clic sobre la tarea “Borrar Datos”. Se abrirá el editor de propiedades de la tarea. Seleccione la propiedad Connection, y cambie su valor para que apunte hacia la conexión “AWork MartDemo”:



4. A continuación, seleccione la propiedad SQLStatement, y presione el botón . Escriba la siguiente sentencia SQL:



5. Pulse el botón OK para confirmar la sentencia escrita. Pulse nuevamente el botón OK para aceptar los cambios en la tarea.
6. En la ventana de diseño, seleccione la tarea “Borrar datos”. Seleccione la flecha verde que sale de esta tarea, y arrástrela sobre la tarea “Copiar productos”. Esto define una precedencia en el flujo de control del paquete que establece que la tarea “Borrar datos” debe ejecutarse antes de la tarea “Copiar productos”:



7. Ejecute el paquete para probar su funcionamiento.

3.1.2.3 Control Flow

Escribir código Net para personalizar las transformaciones: Script Component

SSIS incluye un tipo de transformación que permite al desarrollador implementar componentes avanzados, escritos a través de .Net: el Script Component. El código de un Script Component puede aprovechar toda la funcionalidad disponible en el .Net Framework, lo cual permite crear paquetes muy complejos y personalizados. Los lenguajes soportados son Visual Basic .Net y C#. Existen tres tipos de componentes de script:

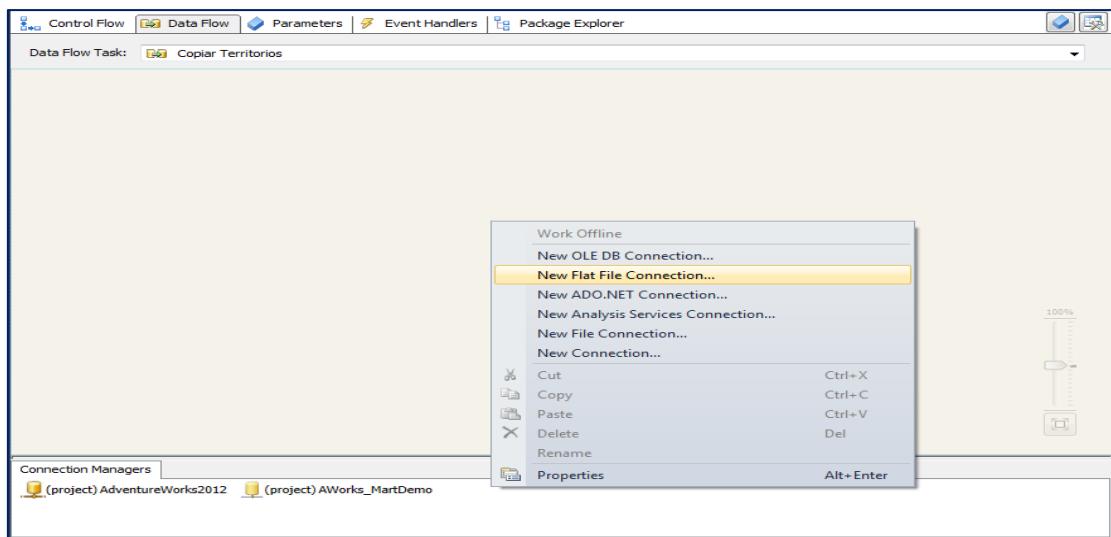
- Source Components: No poseen columnas de input. A través de estos componentes, se pueden escribir orígenes personalizados de datos. De esta manera, es posible desarrollar orígenes sobre formatos de datos no soportados de forma nativa por SSIS.
- Destination Components: No poseen columnas de output. A través de estos componentes, se pueden escribir destinos personalizados de datos para escribir la información hacia formatos no soportados de forma nativa por SSIS.
- Transformation Components: Poseen columnas de input y columnas de output. A través de estos componentes, se puede escribir lógica compleja para efectuar transformaciones personalizadas de datos.

En el siguiente ejercicio, se extraerá información de un archivo de texto, cuya última columna contiene códigos de regiones: 1, 2, 3 y 4. Dichos códigos serán transformados por un script component para ser grabados en el destino con sus equivalencias en inglés: Eastern, Western, Northern y Southern.

Ejercicio:

Escribir una transformación personalizada a través de script component.

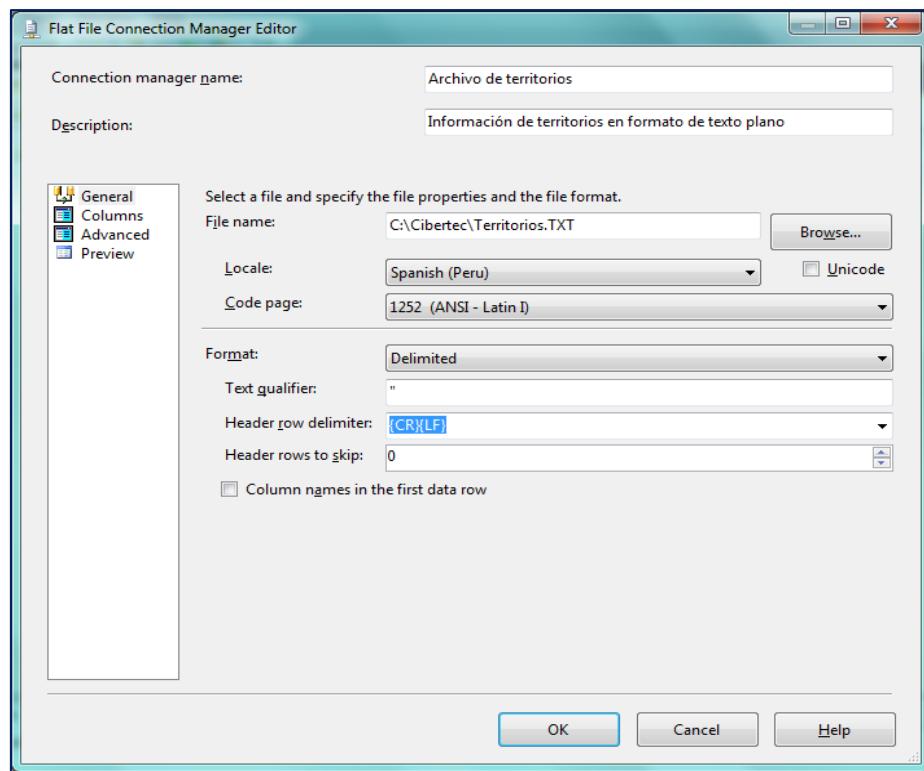
1. Abra el paquete CopiaProductos.dtsx. Agregue una nueva tarea de tipo Data Flow Task, y asígnale el nombre “Copiar territorios”. A continuación, haga un doble clic sobre la tarea “Copiar territorios” para definir el detalle del flujo de datos.
2. En la pestaña Connection managers, agregue una nueva conexión de tipo Flat File. Para esto, haga un doble clic sobre cualquier lugar de la ventana Connection Managers y seleccione la opción New Flat File Connection.



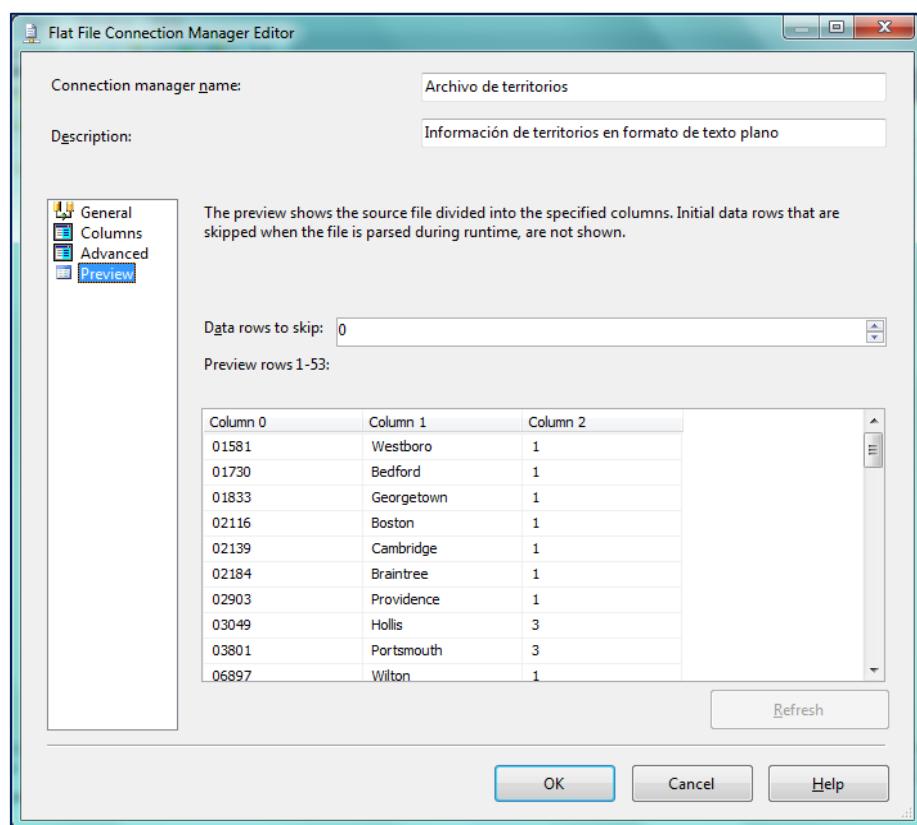
3. En la pestaña Connection managers, agregue una nueva conexión de tipo Flat File. Para esto, haga un doble clic sobre cualquier lugar de la ventana Connection Managers y seleccione la opción New Flat File Connection.

Se abrirá la ventana Flat File Connection Manager Editor. Establezca los siguientes valores para las opciones de esta ventana:

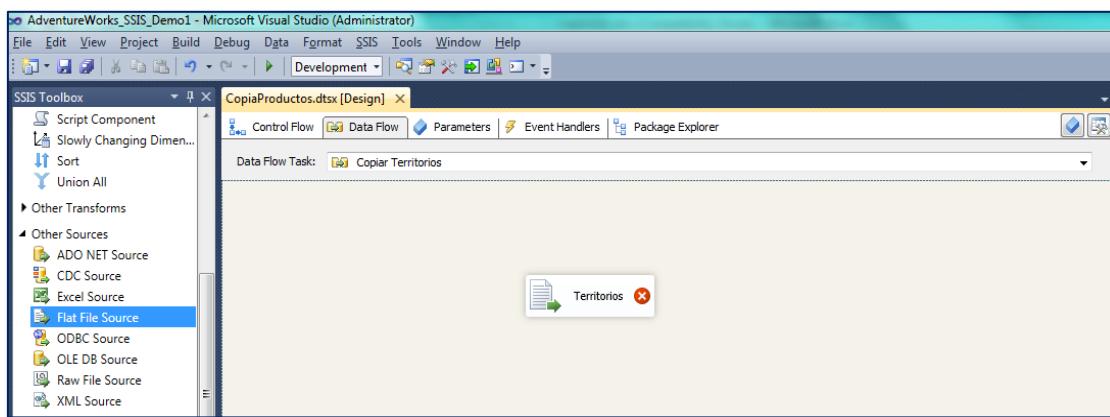
Opción	Valor
Connection manager name	Archivo de territorios
Descripción	Información de territorios en formato de texto plano
File name	Utilice el botón Browse para ubicar el archivo Territorios.txt
Format	Delimited
Text qualifier	(comillas dobles)
Column names in the first data row	Desmarcado



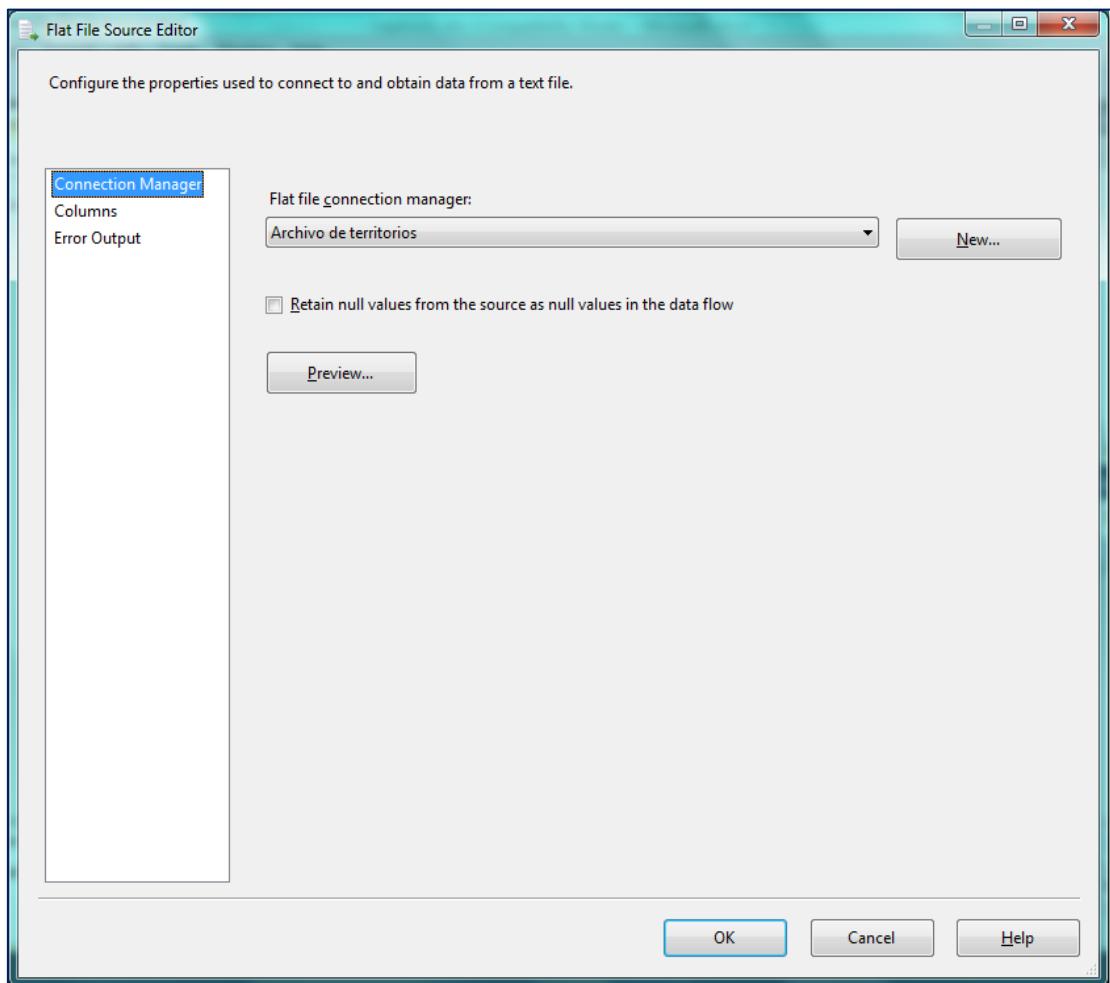
4. En el panel izquierdo de la ventana Flat File Connection Manager Editor, seleccione la opción Preview para observar una vista preliminar de la información. Observe que se han asignado automáticamente los nombres “Column 0”, “Column 1” y “Column 2” a las columnas del archivo de texto.



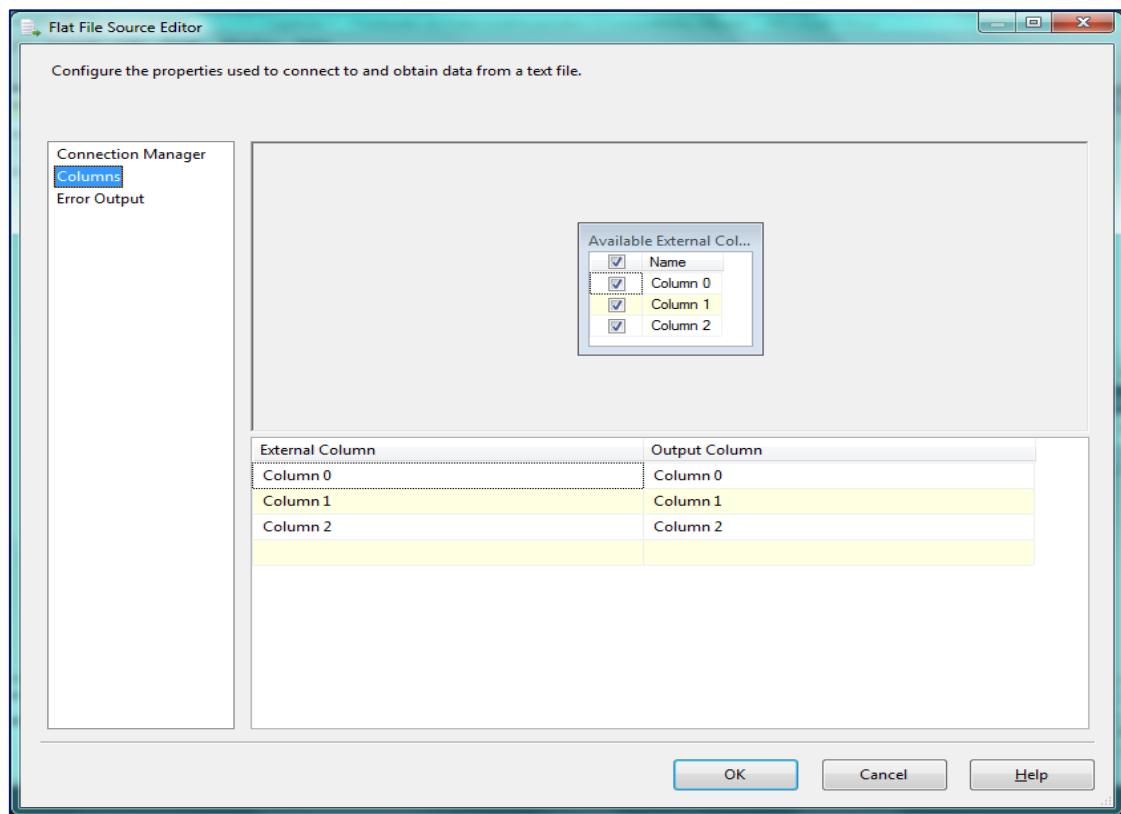
5. Pulse el botón OK para finalizar la configuración de la conexión al archivo de texto.
6. En el Toolbox, expanda la pestaña Other Sources. Seleccione el ítem Flat File Source y arrástrelo sobre la ventana de diseño. Renombre el origen resultante como “Territorios”.



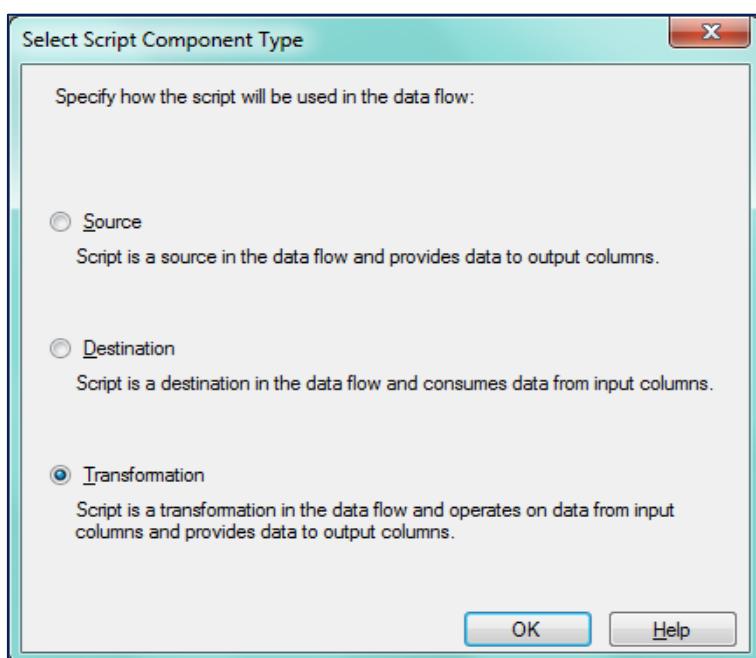
7. Haga doble clic sobre “Archivo de territorios”. Se abrirá la ventana Flat File Source Editor. En la propiedad Flat file connection manager, seleccione “Archivo de territorios”.



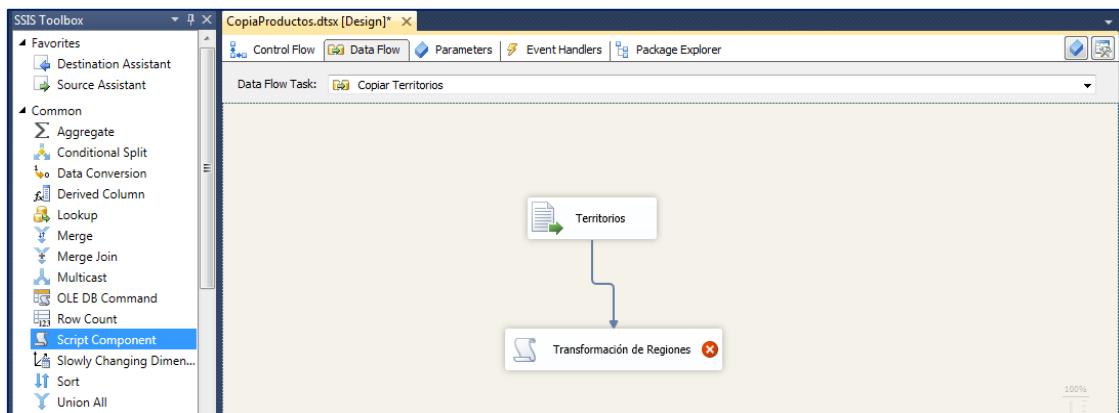
8. En el panel izquierdo de la ventana Flat File Source Editor, seleccione la opción Columns. Observe que cada columna del archivo de texto ha sido reconocida.



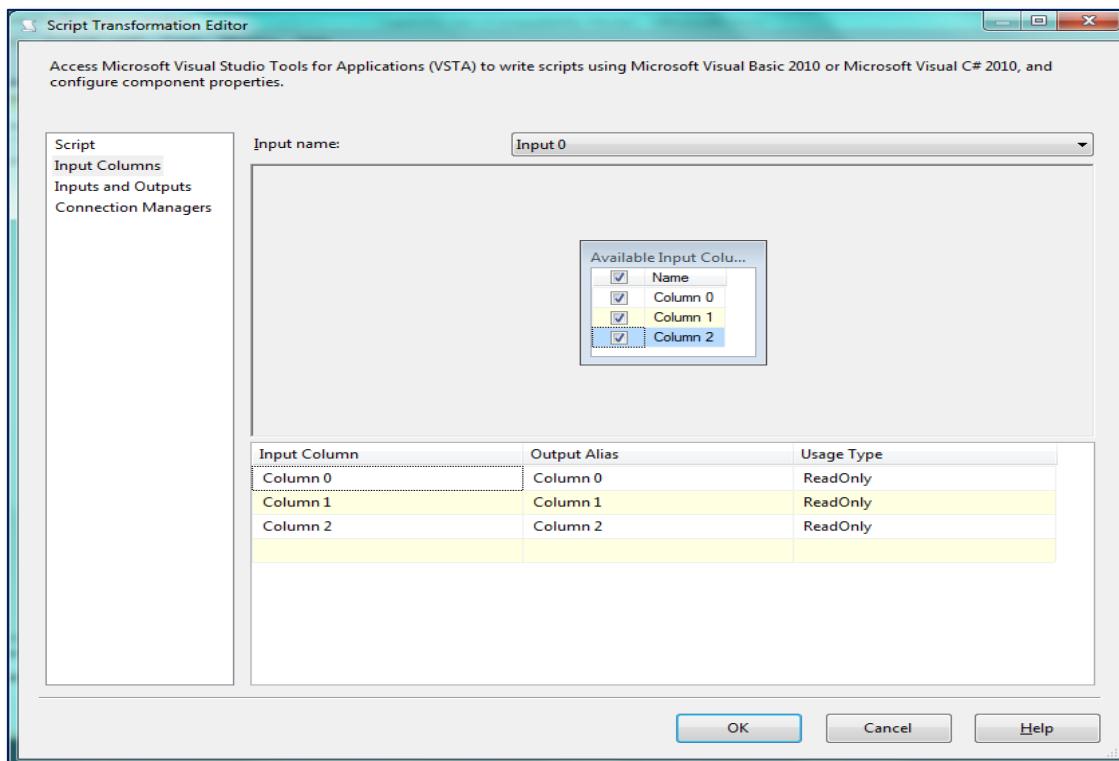
9. Pulse el botón OK para finalizar la configuración del origen del Flat File Source.
10. En el ToolBox, expanda la pestaña Common Transformations. Seleccione el ítem Script Component y arrástrelo sobre la ventana de diseño. Se abrirá una ventana que permite determinar el tipo de componente que se desea crear. Elija la opción Transformation.



11. Pulse el botón OK. Renombre la transformación como “Transformación de regiones”. A continuación, conecte el origen “Territorios” con “Transformación de regiones”. Al finalizar, su editor debe quedar de la siguiente manera:

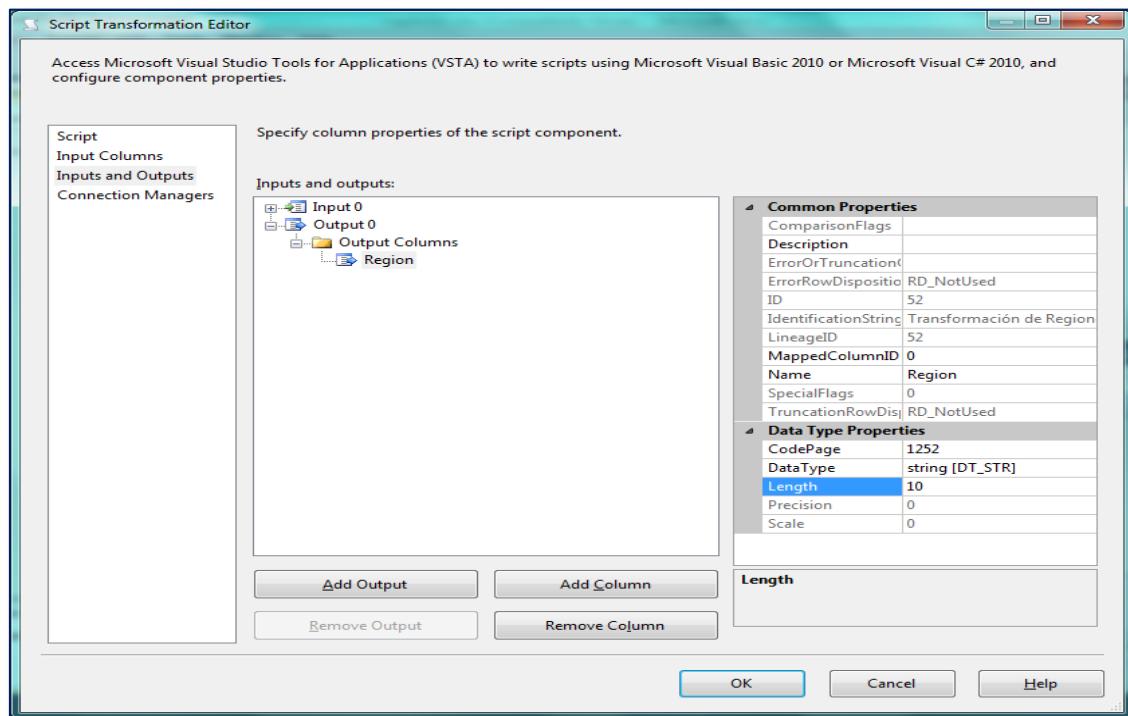


12. A continuación, haga doble clic sobre “Transformación de regiones” para abrir su ventana de propiedades.
 13. En la opción Input Columns del lado izquierdo de la ventana de propiedades, marque los checkboxes que señalan las columnas “Column 0”, “Column 1” y “Column 2” (estas columnas representan los inputs provenientes del origen “Territorios”).

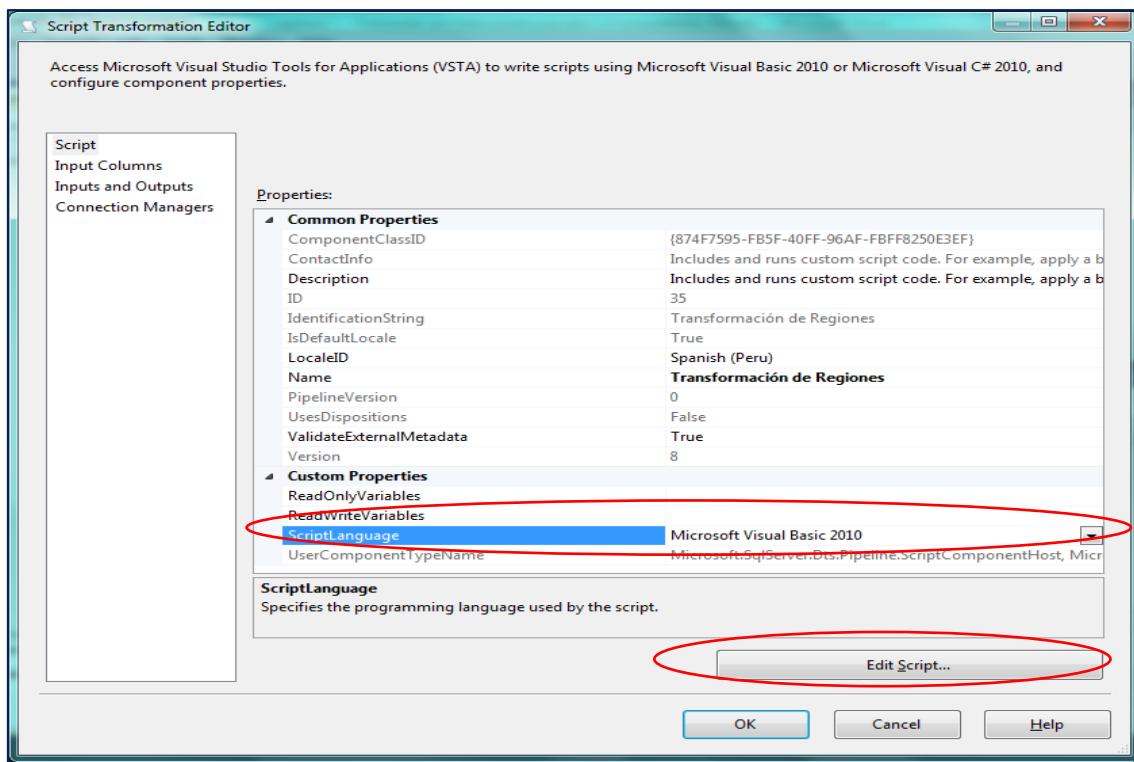


14. Seleccione la opción Inputs and Outputs del lado izquierdo de la ventana de propiedades. Expanda Output 0. Expanda Output Columns. Presione el botón Add Column para agregar un nuevo Output a la transformación. Establezca las siguientes propiedades para el output:

Opción	Valor
Name	Región
Data Type	Seleccione STRING [DT_STR]
Length	10



15. Seleccione la opción Script del lado izquierdo de la ventana de propiedades; luego en la opción ScriptLanguage, seleccionar Microsoft Visual Basic 2010 y presione el botón Design Script.



16. Se abrirá el entorno de desarrollo de .Net para scripts de SSIS. Escriba el siguiente código para definir la transformación en la función Input0_ProcessInputRow:

```

Public Overrides Sub PostExecute()
    MyBase.PostExecute()
    '
    ' Add your code here
    '
End Sub

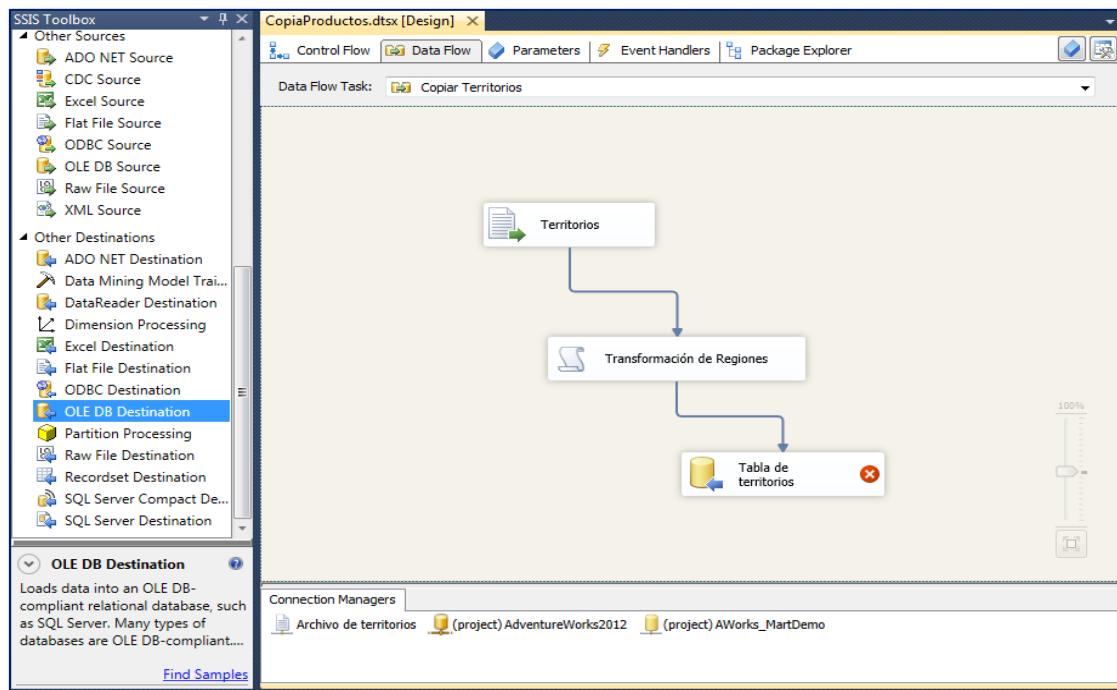
'This method is called once for every row that passes through the component from Input0.
'
'Example of reading a value from a column in the the row:
' zipCode = Row.ZipCode

'Example of writing a value to a column in the row:
' Row.ZipCode = zipCode
Public Overrides Sub Input0_ProcessInputRow(ByVal Row As Input0Buffer)
    '
    ' Add your code here
    '

    Select Case CType(Row.Column2, Integer)
        Case 1
            Row.Region = "Eastern"
        Case 1
            Row.Region = "Western"
        Case 1
            Row.Region = "Northern"
        Case 1
            Row.Region = "Southern"
    End Select
End Sub
End Class

```

17. Grabe el script presionando el botón . Cierre el entorno de desarrollo de scripts y pulse el botón OK para finalizar la edición de la transformación.
18. En el Toolbox, expanda la pestaña Other Destinations. Seleccione el ítem OLE DB Destination y arrástrelo sobre la ventana de diseño. Renombre el destino como “Tabla de territorios”. A continuación, conecte “Transformación de regiones” con “Tabla de territorios”. La ventana del editor debe quedar como sigue:

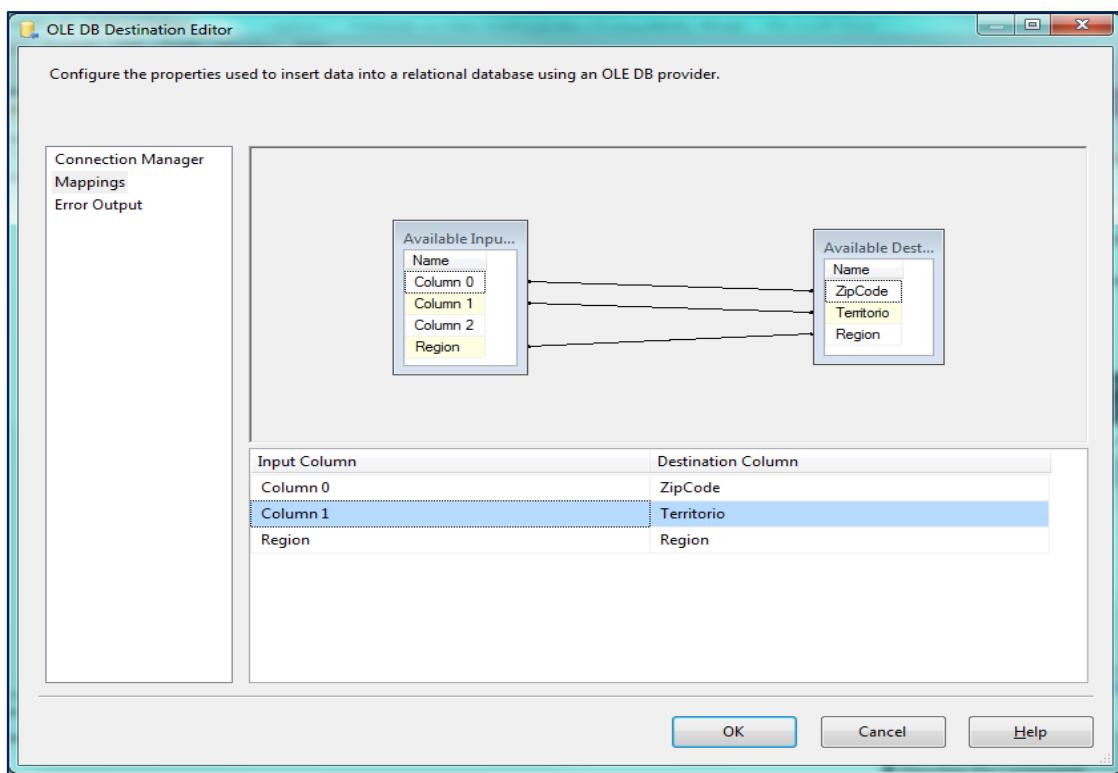


19. Haga doble clic sobre el destino “Tabla de territorios” para abrir su ventana de propiedades. Seleccione la opción Connection Manager del lado izquierdo de la pantalla, y establezca los siguientes valores para las propiedades:

Opción	Valor
OLE DB Connection Manager	Seleccione AWorks MartDemo
Data Access Mode	Table or view – Fast load
Name of the table or the view	Pulse el botón New para crear una nueva tabla. Escriba la siguiente sentencia SQL: CREATE TABLE Territorios (ZipCode char(5), Territorio varchar(50), Region varchar(10))

20. Seleccione la opción Mappings del lado izquierdo de la pantalla. Establezca los siguientes mapeos entre las columnas definidas en “Transformación de regiones” y el destino “Tabla de territorios”:

Column 0	ZipCode
Column 1	Territorio
Region	Region



21. Presione el botón OK para finalizar la edición del destino.
22. Guarde el paquete y ejecútelo. Verifique que en la columna Región de la tabla Territorios se han insertado los valores transformados.
23. Actividad sugerida: Modifique la tarea “Borrar Datos” para eliminar los datos de la tabla Territorios antes de iniciar las transferencias de datos.

3.1.2.4 Container

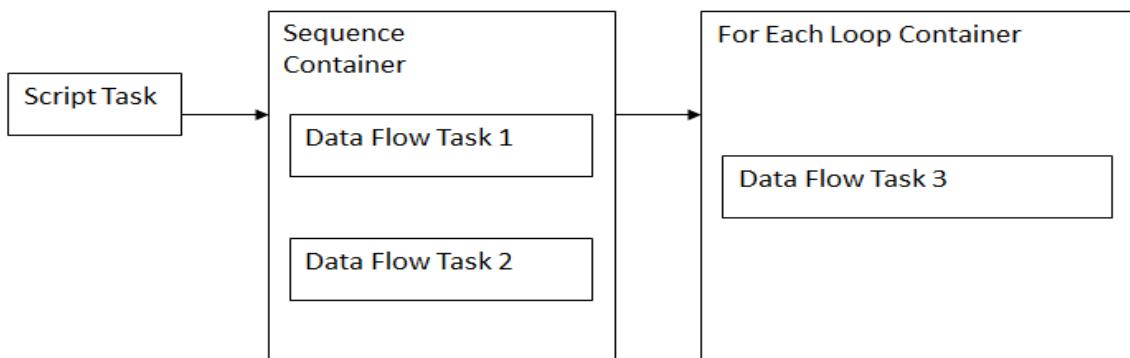
Containers (Contenedores) y variables

Containers (Contenedores)

Los contenedores permiten agrupar tareas relacionadas. El uso de contenedores hace más sencillo definir el flujo de las tareas al interior de un paquete. La siguiente figura muestra un ejemplo de la definición de tareas agrupadas en contenedores:

Como puede observarse en la figura, resulta más sencillo representar el flujo de ejecución de las tareas utilizando contenedores. Existen tres tipos de contenedores y son los siguientes:

- Sequence Container: Agrupa tareas relacionadas entre sí. Las tareas en el interior de este tipo de contenedor se ejecutarán en paralelo, a menos que se hayan definido precedencias entre ellas.
- For Each Loop: Este contenedor efectúa una iteración a través de los elementos de una colección. Esta colección puede provenir de diversas fuentes: la lista de archivos dentro de una carpeta del sistema operativo, una colección de ADO .Net, etc. Las tareas dentro del contenedor For Each Loop repetirán su ejecución, por cada elemento encontrado en la colección.
- For Loop: Evalúa una expresión, y repite la ejecución de las tareas hasta que dicha expresión sea verdadera.



Variables

A menudo, es necesario que los paquetes SSIS utilicen valores que deben calcularse en tiempo de ejecución. Por ejemplo, una variable en un paquete SSIS puede ser utilizada para almacenar el nombre de un servidor de base de datos. De esta manera, si una base de datos cambia de servidor, no sería necesario actualizar los connection managers o data sources en el proyecto SSIS.

Las propiedades de los objetos de SSIS (data sources, connection managers, tasks, containers, etc.) pueden ser obtenidas desde variables. El uso de variables resulta particularmente útil cuando se utilizan Script Components o Script Tasks. Una variable en SSIS tiene siempre el ámbito o alcance del container donde fue declarada.

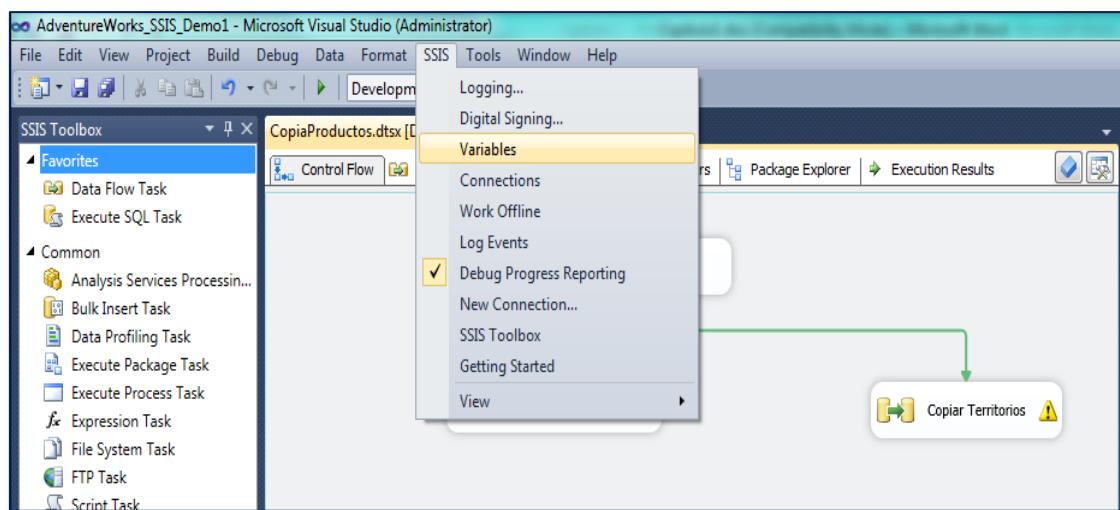
Por ejemplo, si se declara una variable en el paquete SSIS, dicha variable será visible desde cualquier lugar del paquete. Si la variable es declarada en un contenedor For Each Loop, será visible sólo en las tareas que se encuentren en dicho contenedor.

En el siguiente ejercicio, leeremos una serie de archivos de texto correspondiente a datos enviados por diferentes plantas, para ello definiremos una variable varTerritoryFile y un contenedor del tipo For Each Loop que nos permitirá recorrer cada uno de los archivos y cargarlos a la tabla “Despachos”.

Ejercicios:

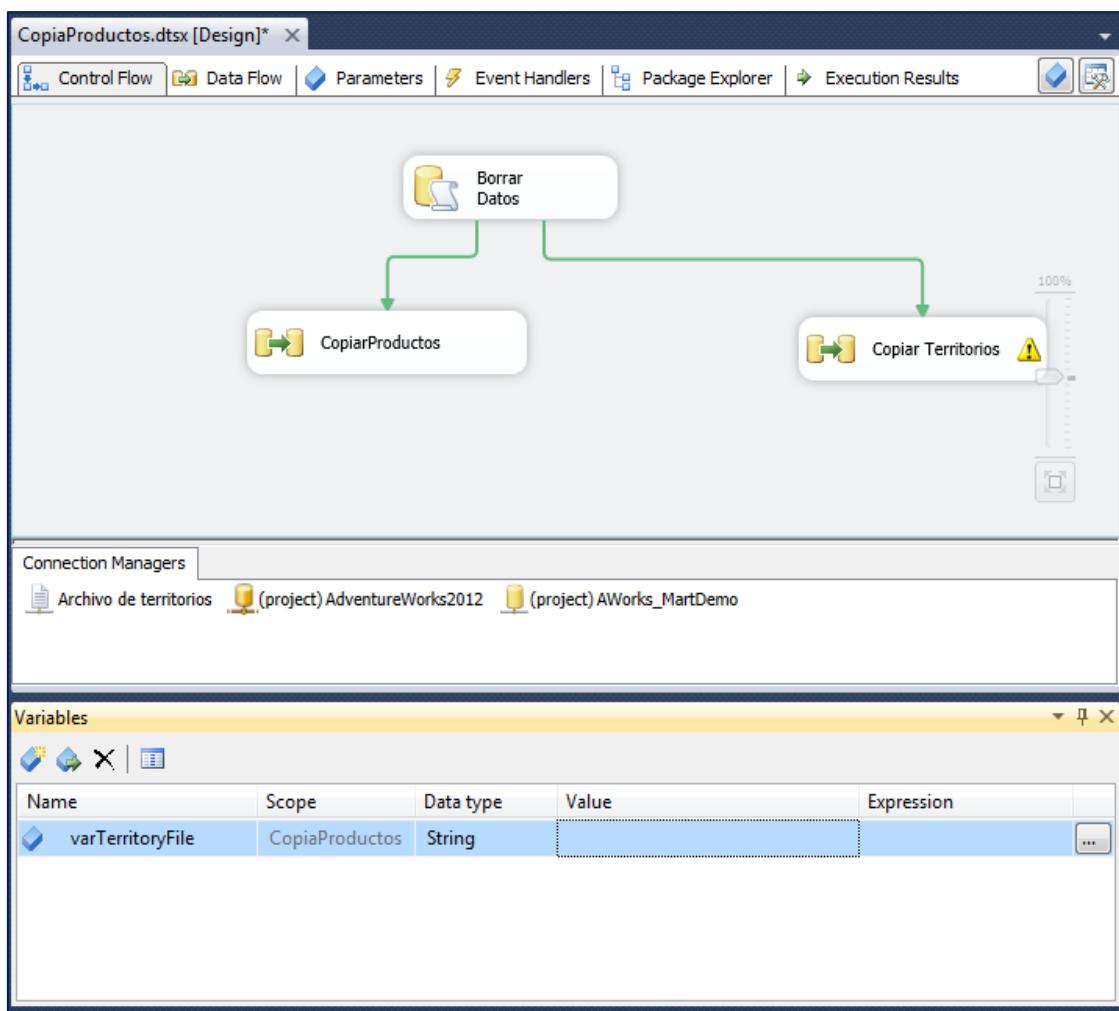
Definir variables en el paquete. Uso del contenedor For Each Loop.

1. Abra el paquete CopiaProductos.dtsx, y sitúese en la pestaña Control Flow. Haga un clic sobre cualquier parte de la ventana de diseño, asegurándose de no seleccionar ninguna tarea. A continuación, seleccione el menú SSIS, y elija la opción Variables.

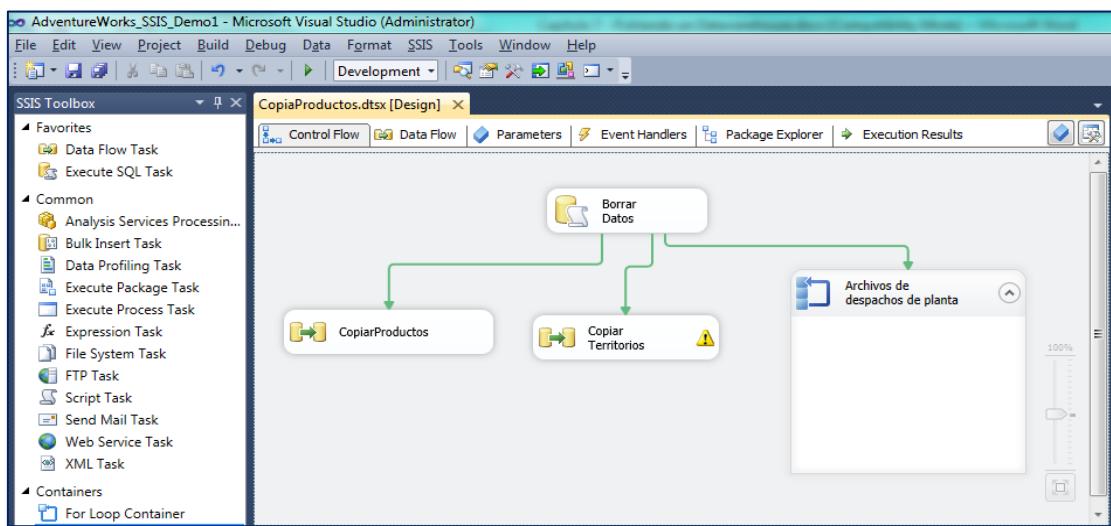


2. Aparecerá la ventana Variables en el lado inferior de la pantalla. Pulse el botón para agregar una nueva variable al paquete. Establezca las siguientes propiedades para la variable:

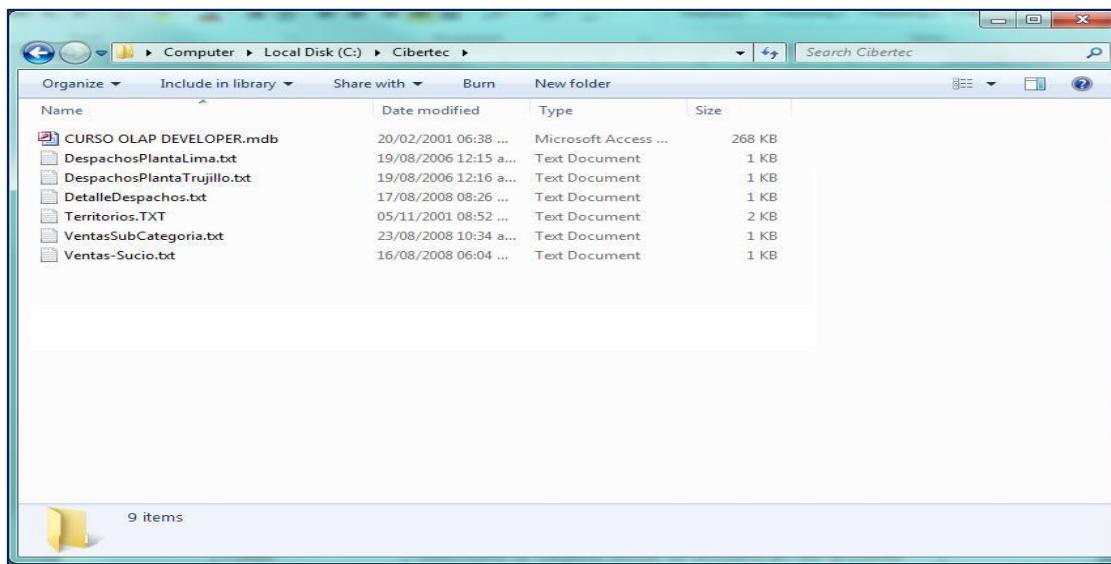
Propiedad	Valor
Name	varTerritoryFile
Scope	CopiaProductos
Data Type	String
Value	



3. Abra el Toolbox, seleccione del grupo Containers el ítem Foreach Loop Container, y arrástrelo sobre la ventana de diseño del paquete. Renombre el elemento creado como “Archivos de despachos de planta”, y agréguele una precedencia con origen en la tarea “Borrar datos”. El entorno de diseño debe quedar de la siguiente manera:

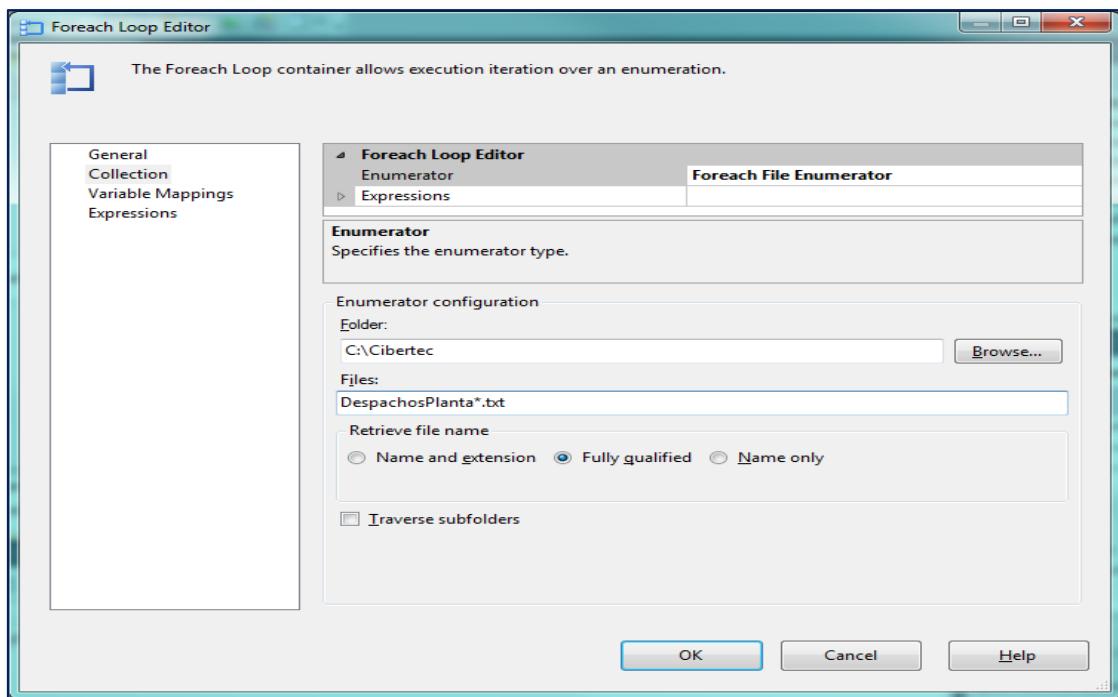


4. El bucle creado en el paquete tiene como objetivo recorrer el directorio de trabajo de esta unidad, para encontrar los archivos de texto cuyo nombre empieza con “DespachosPlanta”. Por cada archivo encontrado, su contenido se copiará a una tabla de SQL Server.

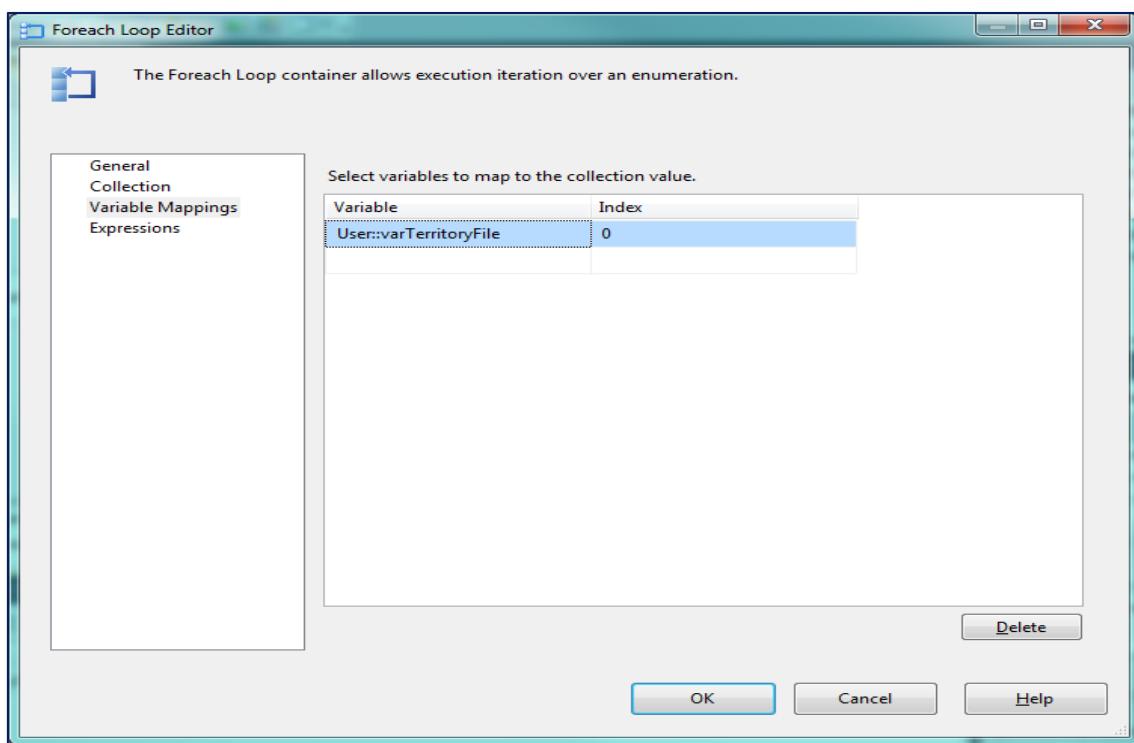


5. Haga doble clic sobre el bucle “Archivos de despachos de planta”. Se abrirá su ventana de propiedades. En la parte izquierda de la ventana, seleccione la opción Collection. Establezca las siguientes opciones para hacer que el bucle recorra el directorio de los archivos de texto con los datos de los despachos de planta:

Propiedad	Valor
Enumerator	Foreach File Enumerator
Folder	Seleccione la carpeta donde se encuentran los archivos “DespachosPlantaLima.txt” y “DespachosPlantaTrujillo.txt”
Files	DespachosPlanta*.txt
Retrieve file name	Seleccione la opción Fully Qualified



6. A continuación, se configurará el bucle para que, por cada vez que encuentre un archivo, almacene su nombre en la variable varTerritoryFile. En el lado izquierdo de la ventana de propiedades, seleccione la opción Variable Mappings. En la columna Variables, seleccione la variable varTerritoryFile, y asígnela al elemento de posición 0 de la colección de archivos (el bucle Foreach recorre colecciones, y las colecciones tienen filas y columnas; en el caso de una colección que recorre una lista de archivos, sólo hay una columna que corresponde con la posición 0 de la colección).

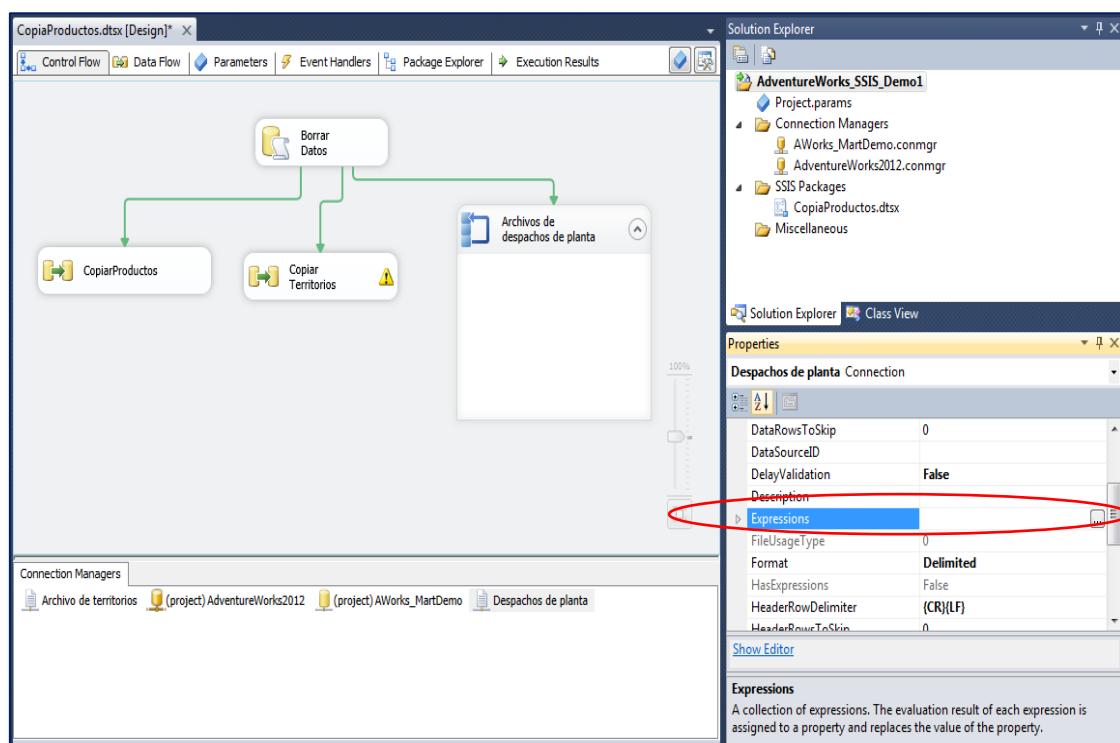


7. Pulse el botón OK para finalizar la edición del bucle.

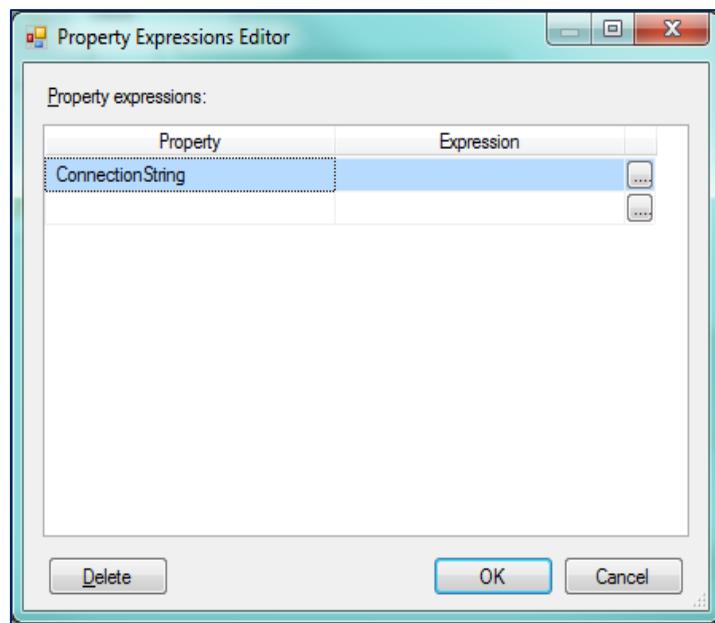
8. Agregue un nuevo connection manager de tipo Flat File Connection, y establezca los siguientes valores para sus propiedades:

Propiedad	Valor
Connection manager name	Despachos de planta
File name	Seleccione el archivo “DespachosPlantaLima.txt”
Format	Delimited
Text qualifier	(comillas dobles)
Column names in the first data row	Seleccionado

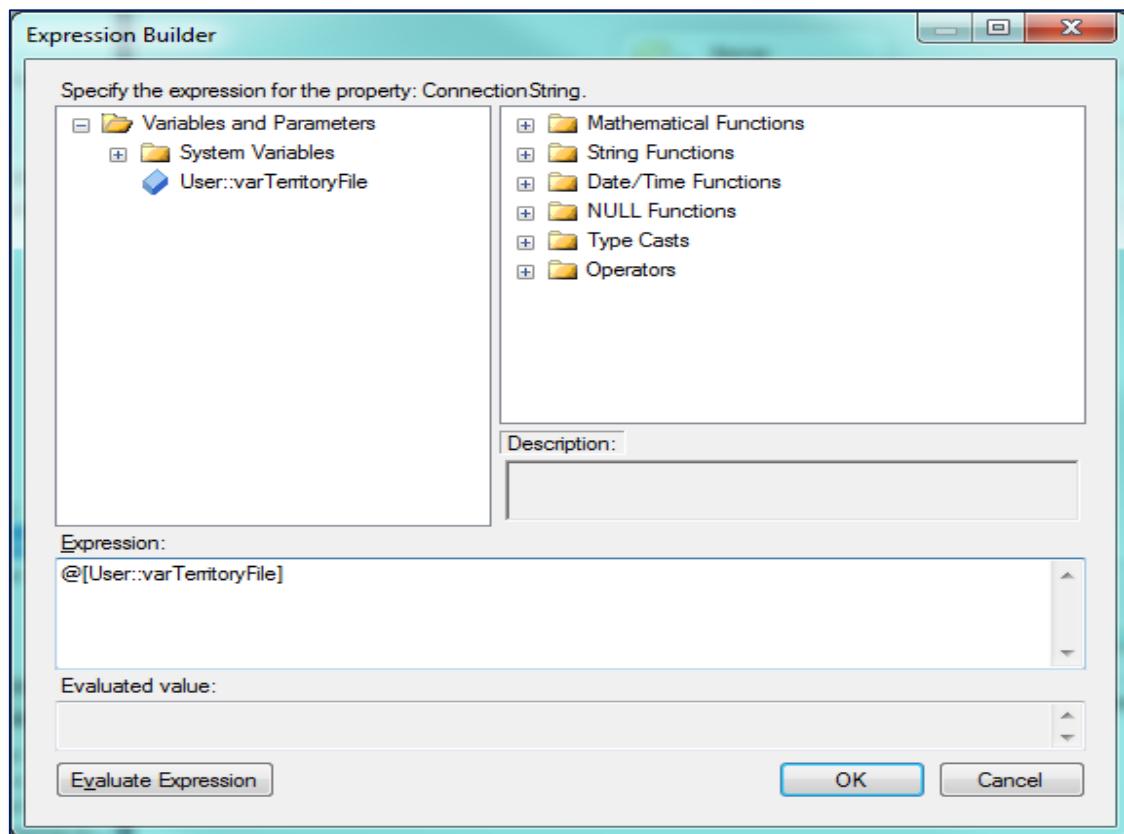
9. Obsérvese que, en el connection manager que acaba de crear, la propiedad File name apunta únicamente al archivo “DespachosPlantaLima.txt”. Para hacer que dicha propiedad obtenga sus valores a partir de la variable varTerritoryFile, seleccione el connection manager “Despachos de planta”. En el menú View, seleccione la opción Properties Window para visualizar la ventana Properties. Seleccione la propiedad Expressions.



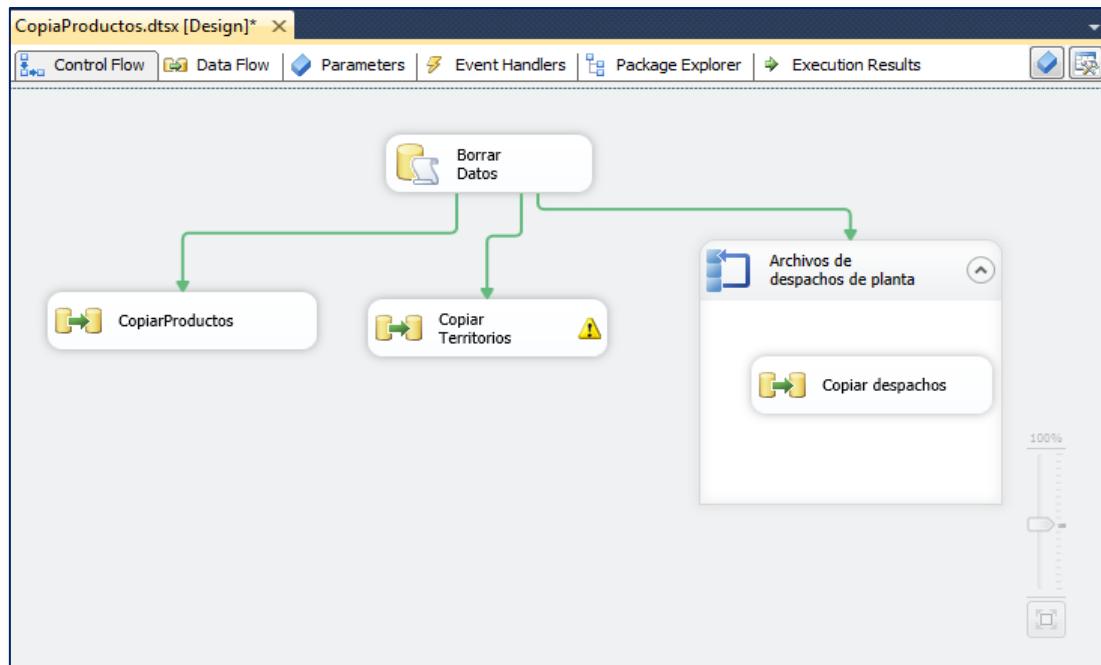
10. En la propiedad Expressions, presione el botón . Se abrirá el editor de expresiones. En la columna Property, seleccione la propiedad ConnectionString.



11. Presione el botón al costado de la columna Expression. Se abrirá el constructor de expresiones. En el lado izquierdo de la pantalla, expanda el nodo Variables, seleccione la variable **User::varTerritoryFile** y arrástrela sobre la caja de texto Expresión. Al finalizar, la ventana debe quedar de la siguiente manera:



12. Pulse el botón OK para cerrar el constructor de expresiones. Pulse nuevamente el botón OK para cerrar la ventana del Property Expressions Editor. Guarde el paquete.
13. Dentro del bucle “Archivos de despachos de planta”, agregue un nuevo Data Flow Task (asegúrese de arrastrar el Data Flow directamente desde el Toolbox y colocarlo dentro del bucle). Renómbrelo como “Copiar Despachos”. Al finalizar, la ventana de diseño debe quedar de la siguiente manera:



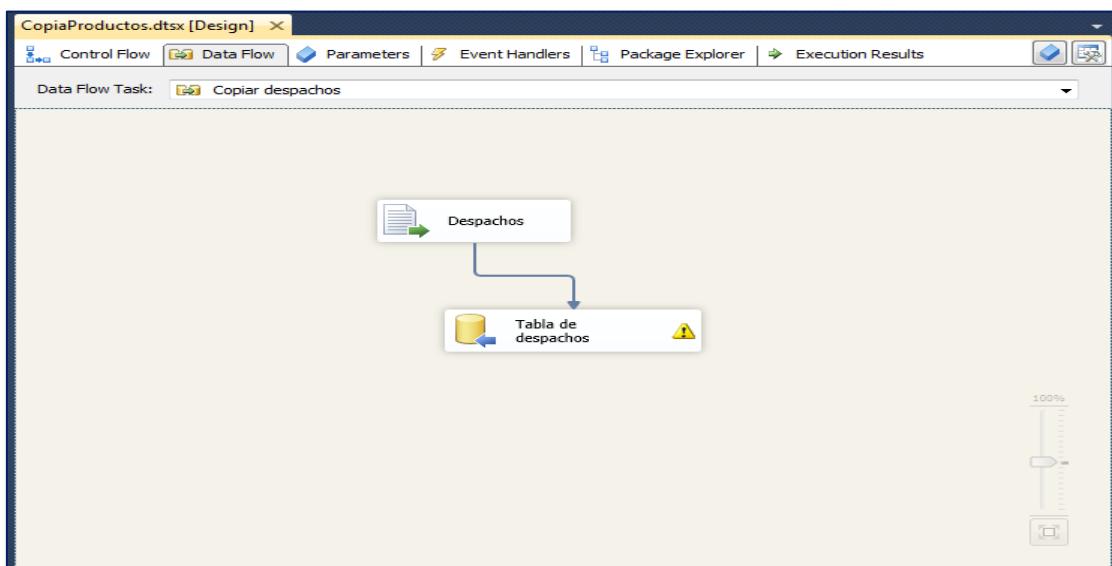
14. Haga doble clic sobre la tarea “Copiar Despachos” para diseñar el flujo de datos. Realice los siguientes pasos:

- Desde el Toolbox, agregue un Flat File Source y renómbrelo como “Despachos”. Asócielo con el connection manager “Despachos de planta”.
- Desde el Toolbox, agregue un OLE DB Destination y renómbrelo como “Tabla de Despachos”. A continuación, una “Despachos” con “Tabla de Despachos”.
- Edite las siguientes propiedades del destino “Tabla de despachos”:

Propiedad	Valor
OLE DB Connection Manager	AWorks MartDemo
Data access mode	Table or view – fast load
Name of the table or the view	Presione el botón New para crear una tabla, con la siguiente sentencia SQL: CREATE TABLE [Despachos]([IdPlanta] int, [IdProducto] int, [IdCliente] char(11), [Cantidad] smallint)

- En la parte izquierda del editor de propiedades del destino OLEDB, seleccione la opción Mappings, y observe los mapeos de columnas que se han creado automáticamente.

Al finalizar, la ventana de diseño debe quedar de la siguiente manera:

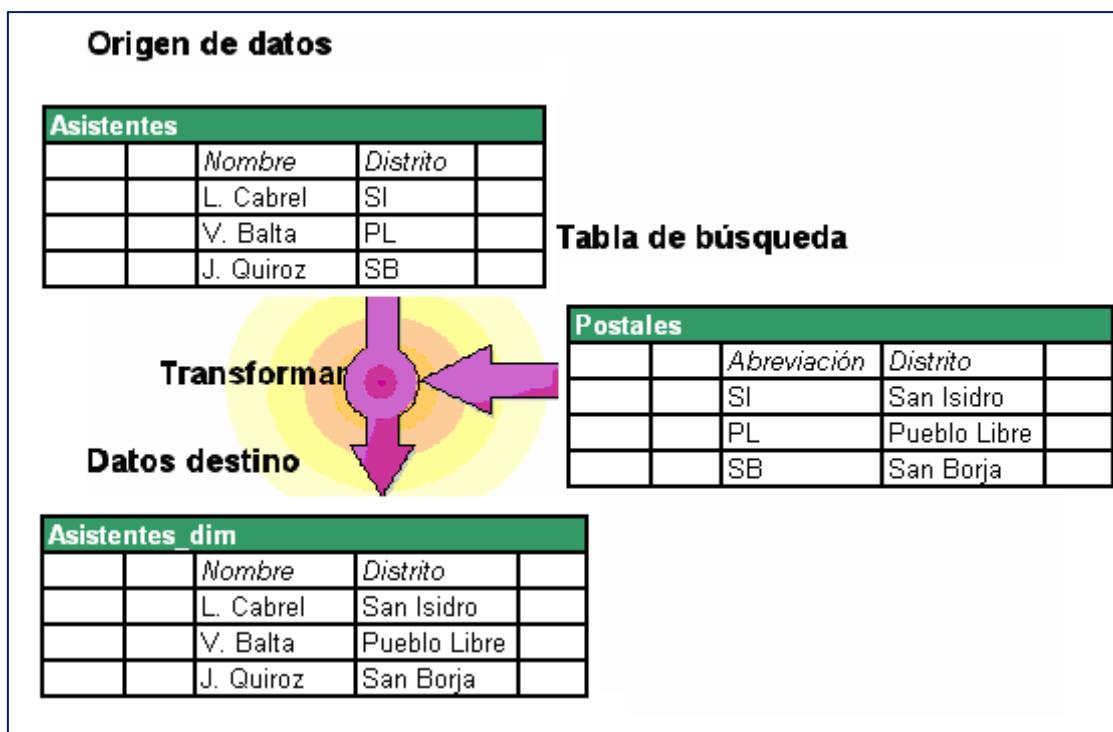


15. Regrese al Control Flow, y edite la tarea “Borrar Datos” para efectuar un DELETE sobre la tabla “Despachos” al inicio de la ejecución del paquete.
16. Grabe los cambios y pruebe el paquete

Lookup y Data Conversions

Lookup

En el ejercicio previo al anterior, se utilizó un Script Component para traducir los códigos numéricicos a nombres de regiones. En dicho ejemplo, debido a que sólo existen cuatro regiones, basta con escribir una sentencia SELECT CASE en el código .Net del Script Component. Sin embargo, si la transformación involucra una cantidad grande de valores distintos (por ejemplo, los códigos de los distritos de un país), el script puede hacerse muy extenso y difícil de escribir.



Los Lookups implementan un mecanismo útil para efectuar transformaciones utilizando tablas de equivalencia. La figura superior muestra la utilización de una tabla de búsqueda para convertir los códigos de distritos de Lima en sus respectivos nombres.

Data Conversion

Los procesos ETL deben leer información de múltiples tipos de fuentes, y transferirla a diversos destinos. Es muy posible, por tanto, que sea necesario efectuar conversiones de datos. SSIS incorpora una transformación llamada Data Conversion, cuya labor es efectuar conversiones de tipos de datos. De esta forma, es posible convertir información de texto de un formato a otro (por ejemplo, convertir cadenas de formato UNICODE a no-UNICODE), convertir tipos de datos de fecha, o establecer la precisión de información numérica.

En el siguiente ejercicio, se usará la transformación Data Conversion para transformar tipos de datos UNICODE de un archivo Access a no-UNICODE; adicionalmente, se implementará una transformación Lookup para buscar la equivalencia de códigos de ciudades y obtener sus descripciones.

Ejercicio:

Utilizar las transformaciones Data Conversion y Lookup

1. Ejecute el siguiente bloque Transact – SQL desde el SQL Server Management Studio:

```
USE AWorks_MartDemo
GO

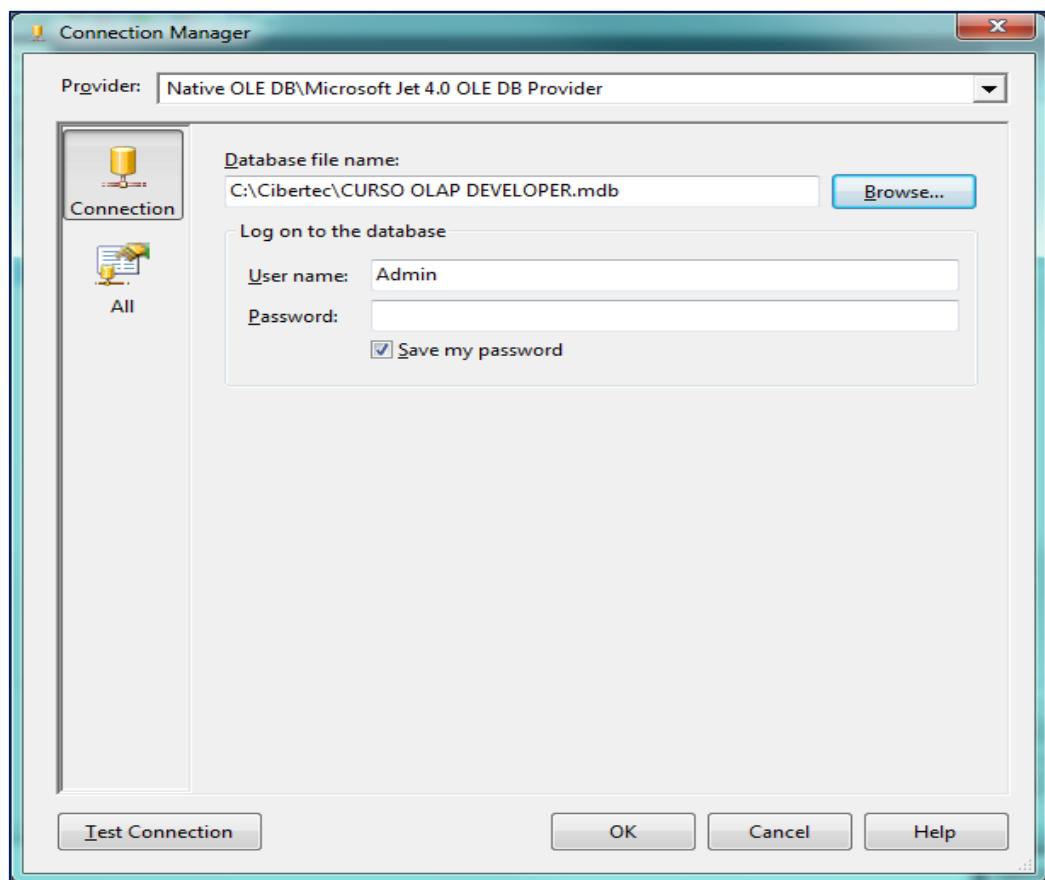
CREATE TABLE Ciudades
(IdCiudad char(2),
Descripcion varchar(20))
```

2. A través del SQL Server Management Studio, ingrese los siguientes registros en la tabla Ciudades:

IdCiudad	Descripción
AN	ANCASH
AQ	AREQUIPA
CJ	CAJAMARCA
CZ	CUZCO
HU	HUANUCO
LI	LIMA

3. En el proyecto de Integration Services, agregue un nuevo Connection Manager que utilice el proveedor Native OLE DB\Microsoft Jet 4.0 OLE DB Provider apuntando hacia el archivo Access llamado “CURSO OLAP DEVELOPER.mdb”.

Asegúrese de que el checkbox Save my password esté seleccionado.

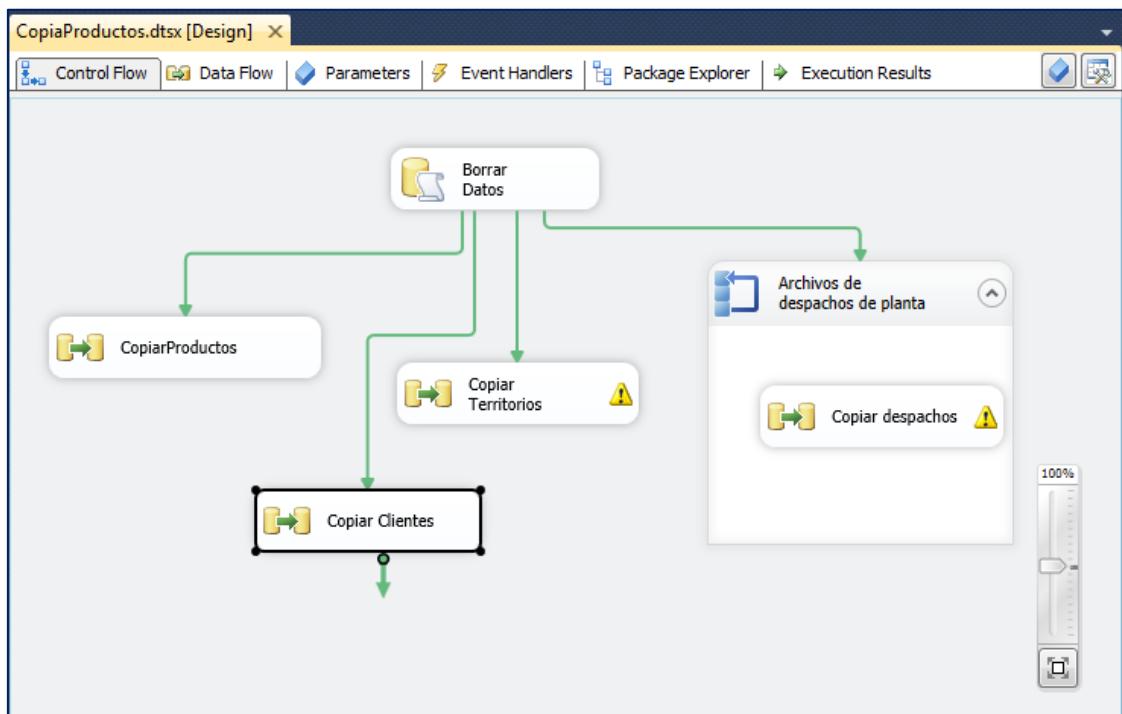


4. El archivo Access “CURSO OLAP DEVELOPER.mdb” contiene una tabla llamada “ClientesPeru” con los siguientes datos:

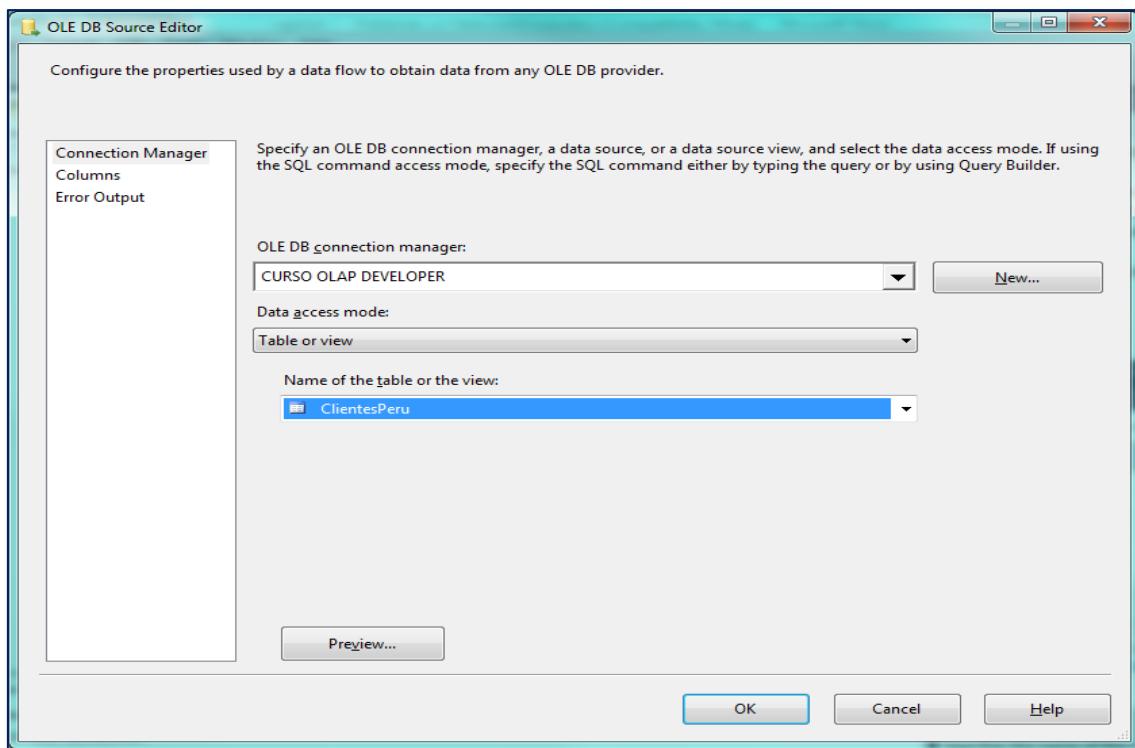
CuentasPeru : Table				
	idCuenta	NomCuenta	TelCuenta	Ciudad
1	Carlos Acosta Rojas	475-0074	LI	
2	Karina Palacios Ramos	555-4729	CJ	
3	Jessica Linares Hidalgo	555-3932	LI	
4	Karen Mostafan Urrutia	555-7788	AQ	
5	Carlos Ruiz Flores	921-1234	CZ	
6	Paola Marian Yepez	621-0846	AQ	
7	Jorge Torres Miranda	564-7575	AQ	
8	George Mario Rosales	555-2282	AQ	
9	Percy Arellano Hernandez	425-8575	AQ	
10	Susana Leon Jimenez	555-4729	LI	
11	Susan Leon Cavessa	555-1212	LI	
12	Rina Gamarra Flores	135-5555	LI	
13	Julio Sanchez Reyes	555-3392	CJ	
14	Daniel Sanchez Medina	452-0765	AN	
15	Sandro Diaz Medina	555-7647	AN	
16	Hernan Asturias Torres	555-2282	AQ	
17	Viviana Cardenas Flores	241-0391	LI	
18	Hugo Perez Polo	325-2658	HU	
19	Walter Perez Davalos	555-0297	HU	
20	Gerardo Cardenas Polo	675-3425	HU	

Observe que la columna “Ciudad” contiene códigos de ciudades (“LI”, “CJ”, etc.).

5. Agregue al paquete (con cuidado de NO colocarlo dentro del bucle “Archivos de despachos de planta”) un nuevo Data Flow Task y renómbrelo como “Copiar clientes”. Agréguele una precedencia con origen en la tarea “Borrar Datos”.



6. Dentro del Data Flow Task “Copiar clientes”, agregue un nuevo OLE DB Source estableciendo las siguientes propiedades:



Renombre a este OLE DB Source como “Access Clientes”.

7. La información de las columnas de texto en el archivo Access se encuentra almacenada en formato internacional (UNICODE). En SQL Server, los tipos de datos NCHAR, NVARCHAR, NVARCHAR (max) y NTEXT soportan este formato de texto. Sin embargo en este ejercicio, la información del archivo Access será transferida a una tabla con columnas CHAR y VARCHAR que no soportan el formato UNICODE. Por tanto, es necesario que los datos del Access pasen por una conversión de tipo antes de ser copiados a SQL Server 2014. La tarea Data Conversion permite efectuar este tipo de operación. En el Data Flow Task “Copiar Clientes”, agregue desde el Toolbox el ítem llamado Data Conversion, y renómbrelo como “Convertir a no-UNICODE”. Finalmente, una la salida del OLE DB Source “Access Clientes” con la conversión “Convertir a no-UNICODE”.

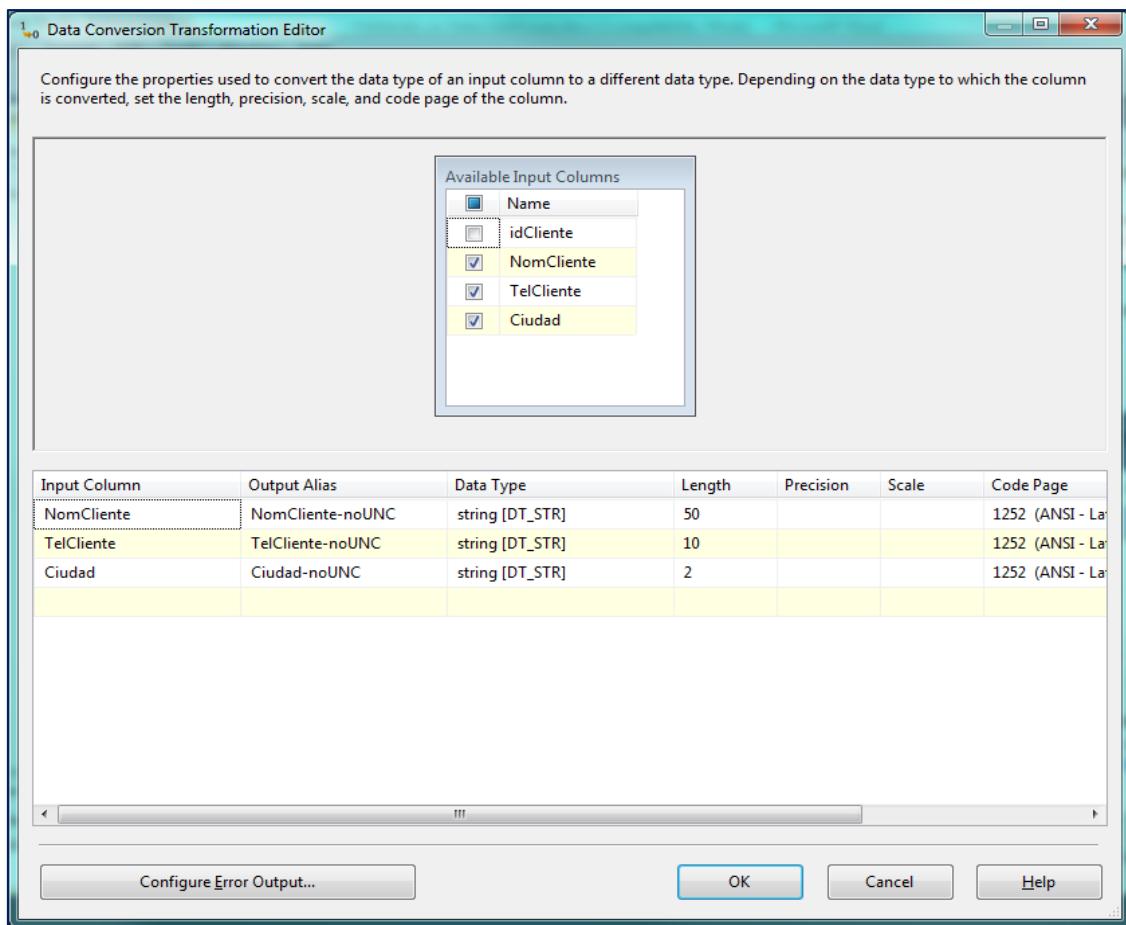


8. Haga doble clic en “Convertir a no-UNICODE” para editar las propiedades de la conversión de datos. Se abrirá el Data Conversion Transformation Editor. Esta ventana muestra los inputs provenientes del origen “Access Clientes” y permite

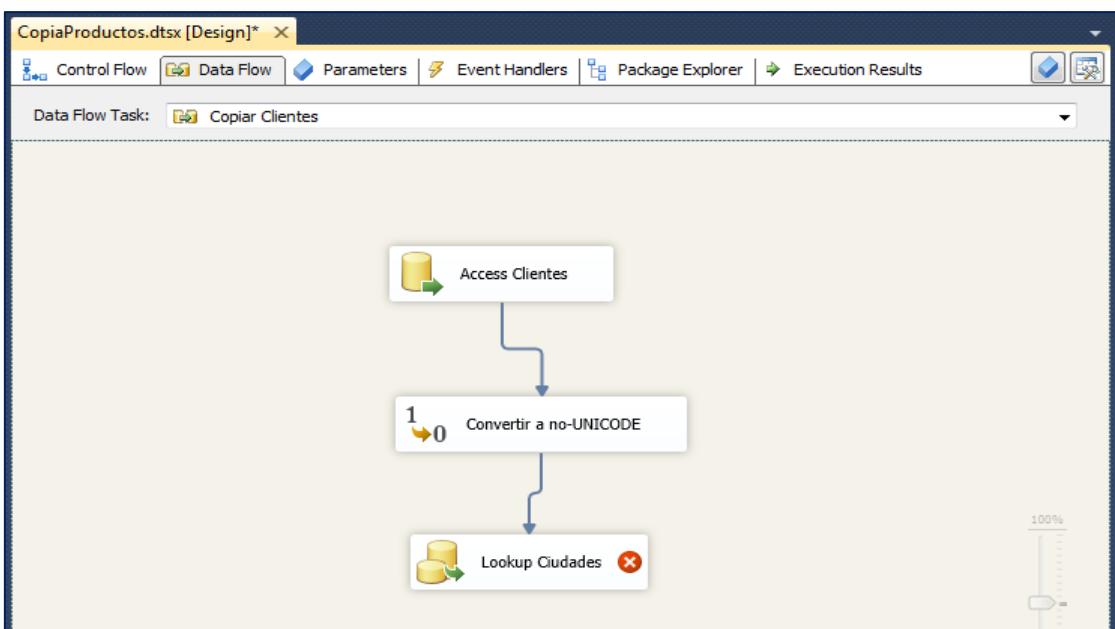
efectuar conversiones de datos. En el panel inferior, configure las siguientes propiedades para convertir las columnas de texto del Access a formato normal:

Input Column	OutPut Alias	Data Type	Length
NomCliente	NomCliente-noUNC	String [DT_STR]	50
TelCliente	TelCliente-noUNC	String [DT_STR]	10
Ciudad	Ciudad-noUNC	String [DT_STR]	2

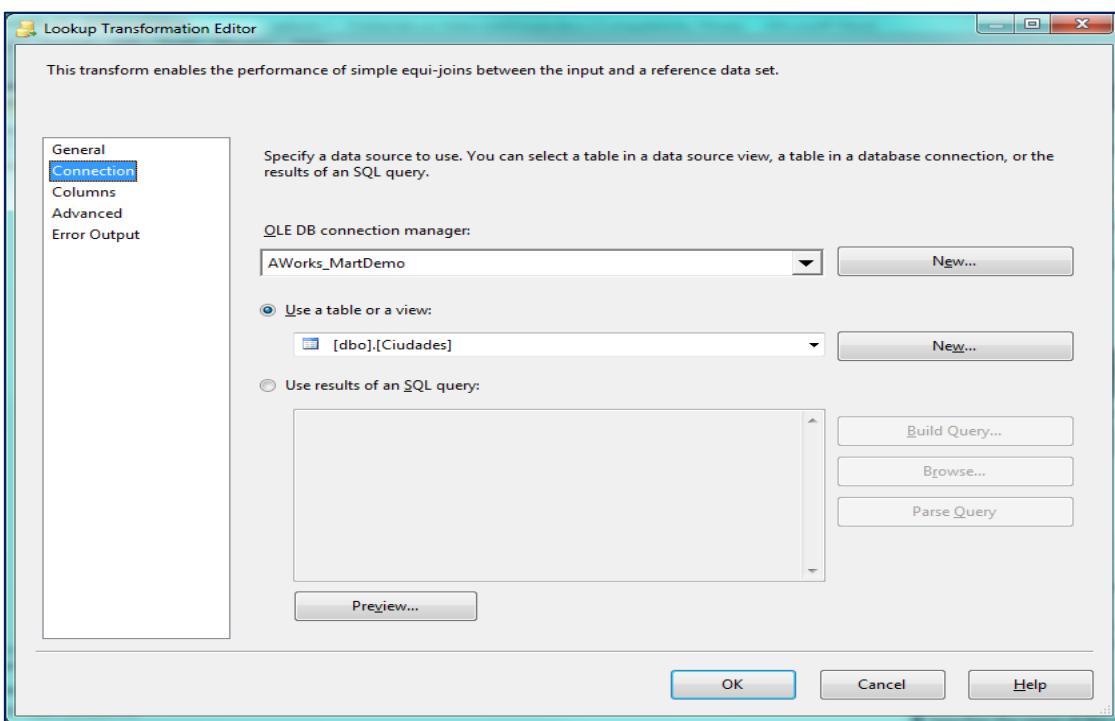
Esto significa que, al output proveniente de “Access Clientes”, se agregan tres columnas adicionales llamadas NomCliente-noUNC, TelCliente-noUNC y Ciudad-noUNC, que contienen los datos transformados a cadena normal (string).



9. Pulse el botón OK para finalizar la edición del Data Conversion. A continuación, agregue desde el Toolbox una transformación de tipo Lookup, y renómbrela como “Lookup Ciudades”. Finalmente, conecte la salida de “Convertir a no-UNICODE” con la transformación “Lookup Ciudades”.

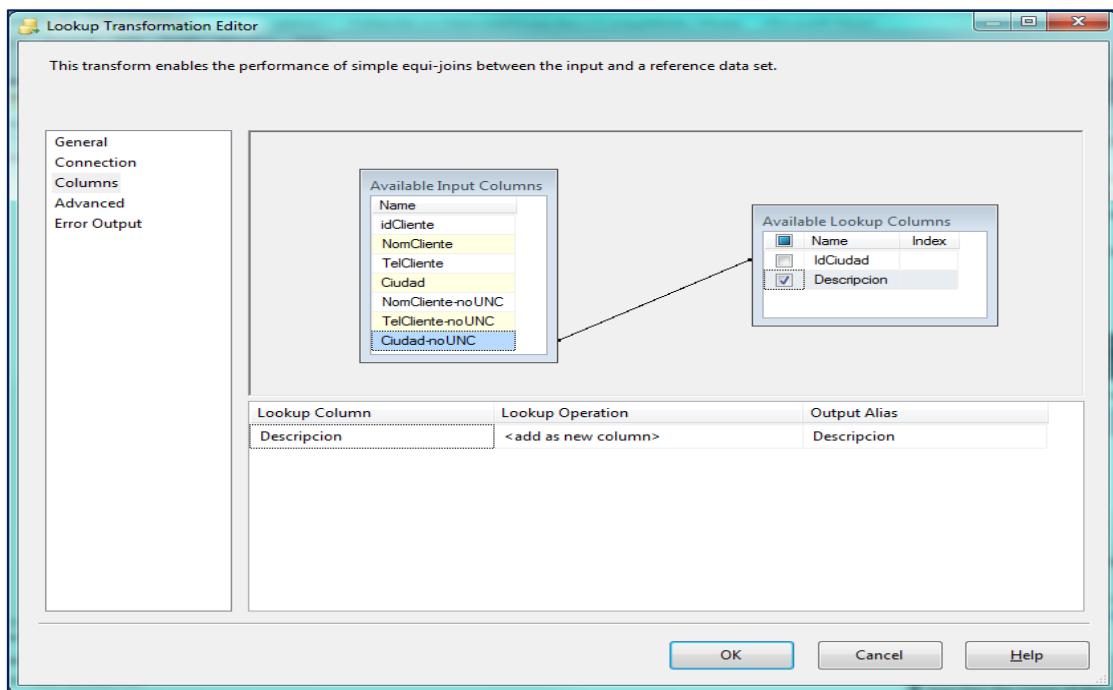


10. Haga doble clic sobre “Lookup Ciudades” para editar sus propiedades. Se abrirá la ventana del Lookup Transformation Editor. En la parte izquierda del editor, seleccione Connection, la propiedad OLE DB Connection Manager debe apuntar al connection manager “AWorks MartDemo”. La tabla de referencia debe ser “dbo.Ciudades”.



11. Seleccione en la opción Columns de la ventana del Lookup Transformation Editor. La tabla Available Lookup Columns contiene las columnas de la tabla dbo.Ciudades que servirá para relacionar el código de la ciudad (columna “IdCiudad”) con el nombre de la ciudad (columna “Descripcion”). En la tabla Available Lookup Columns, seleccione el campo “IdCiudad” y arrástrelo sobre el campo “Ciudad-noUNC” de la tabla Available Input Columns. Luego, marque el checkbox de la

columna “Descripcion” de la tabla Available Lookup Columns. Al finalizar, la ventana debe quedar de la siguiente manera:



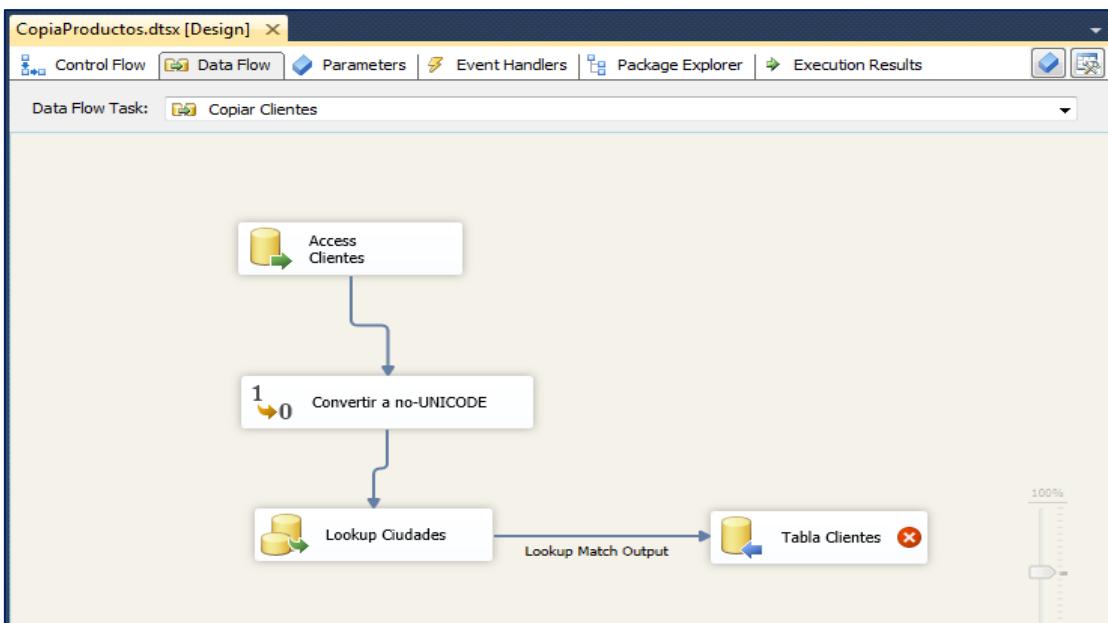
De esta manera, la transformación Lookup efectúa una operación de join con la tabla dbo.Ciudades, basada en la igualdad de las columnas “IdCiudad” y “Ciudad-noUNC”. El

join efectuado permite obtener, para cada registro en la tabla “ClientesPeru” del archivo

Access de origen, el valor de la columna “Descripcion” que le corresponde en la tabla dbo.Ciudades.

12. Pulse el botón OK para confirmar los cambios.

13. Desde el Toolbox, en la pestaña Other Destinations, agregue un nuevo OLE DB Destination, y renómbrelo como “Tabla Clientes”. Conecte la salida de “Lookup Ciudades” con “Tabla Clientes” y en la ventana Input Output Selection seleccione en Output: Lookup Match Output.

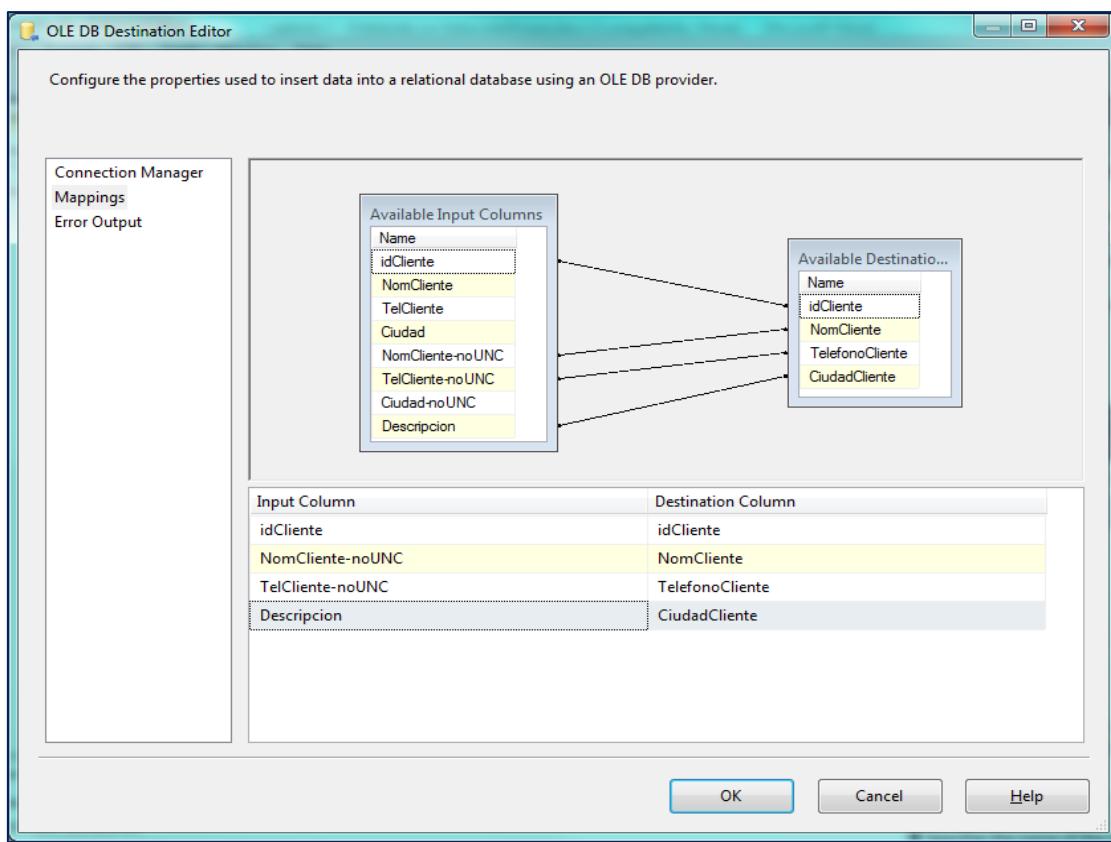


14. Haga doble clic sobre el destino “Tabla clientes” para abrir su ventana de propiedades. Seleccione la opción Connection Manager del lado izquierdo de la pantalla y establezca los siguientes valores para las propiedades:

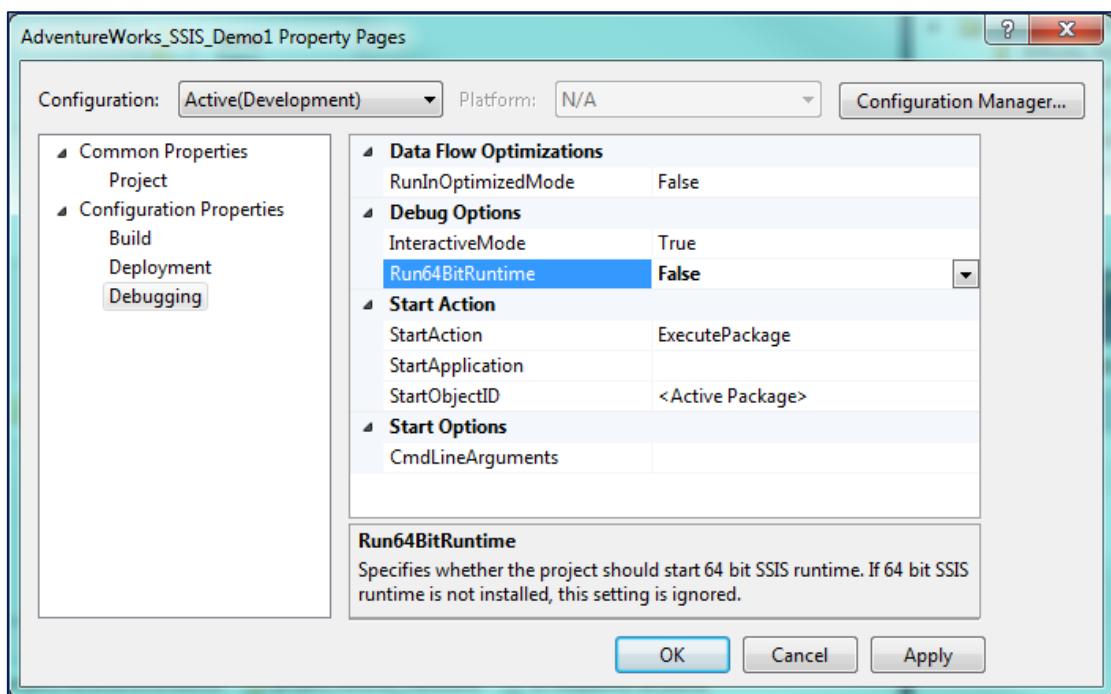
Propiedad	Valor
OLE DB Connection Manager	Seleccione AWworks MartDemo
Data Access Mode	Table or view – Fast load
Name of the table or the view	Pulse el botón New para crear una nueva tabla. Escriba la siguiente sentencia SQL: CREATE TABLE Clientes (IdCliente integer, NombreCliente varchar(50), TelefonoCliente varchar(10), CiudadCliente varchar(20))

15. Seleccione la opción Mappings del lado izquierdo de la pantalla. Establezca los siguientes mapeos entre las columnas de “Lookup Ciudades” y el destino “Clientes”:

idCliente	idCliente
NomCliente-noUNC	NombreCliente
TelCliente-noUNC	TelefonoCliente
Descripcion	CiudadCliente



16. Pulse el botón OK para confirmar los cambios.
17. Regrese al Control Flow y edite la tarea “Borrar Datos” para efectuar un DELETE sobre la tabla “Clientes” al inicio de la ejecución del paquete.
18. Grabe los cambios y ejecute el paquete. Al finalizar, efectúe una consulta sobre la tabla “Clientes” en la base de datos AWWorks_MartDemo.
19. Si tuviera error al leer el archivo Access, deshabilite el Run-time de 64 bits.



3.1.2.5 Package

Package Configurations

A menudo, para aumentar la flexibilidad de un paquete SSIS, es necesario pasarle ciertos valores de entrada. Por ejemplo, si una base de datos se mueve hacia otro servidor, es deseable que el nombre del servidor pueda ser establecido en tiempo de ejecución sin necesidad de modificar el paquete SSIS.

Los Package Configurations implementan mecanismos para pasarle valores de entrada a un paquete SSIS. Los package configurations permiten definir información de entrada a través de:

- Un archivo XML: Los valores quedan almacenados en un archivo XML. El paquete SSIS cargará estos valores al iniciarse su ejecución.
- Variables de entorno: Los valores de entrada del paquete se obtienen de las variables de entorno de Windows.
- Entrada de registro: Los valores de entrada del paquete se obtienen desde el registro de Windows.
- Parent package variable: Los valores de entrada del paquete se encuentran definidos como variables en un paquete de nivel superior que contiene al paquete actual.
- SQL Server: Los valores de entrada del paquete se encuentran almacenados en una tabla de SQL Server.

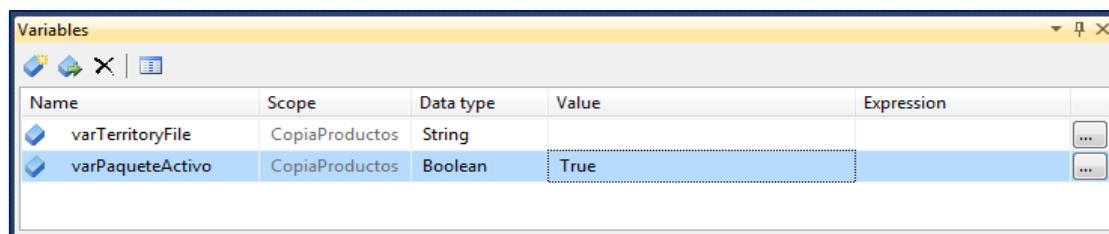
En el siguiente ejercicio, se agregará un package configuration que almacenará una configuración para determinar si el paquete debe ejecutarse o no. Dicho valor será asignado a una variable, y evaluado en una tarea de tipo Script Task para definir si la ejecución del paquete debe interrumpirse.

Ejercicio:

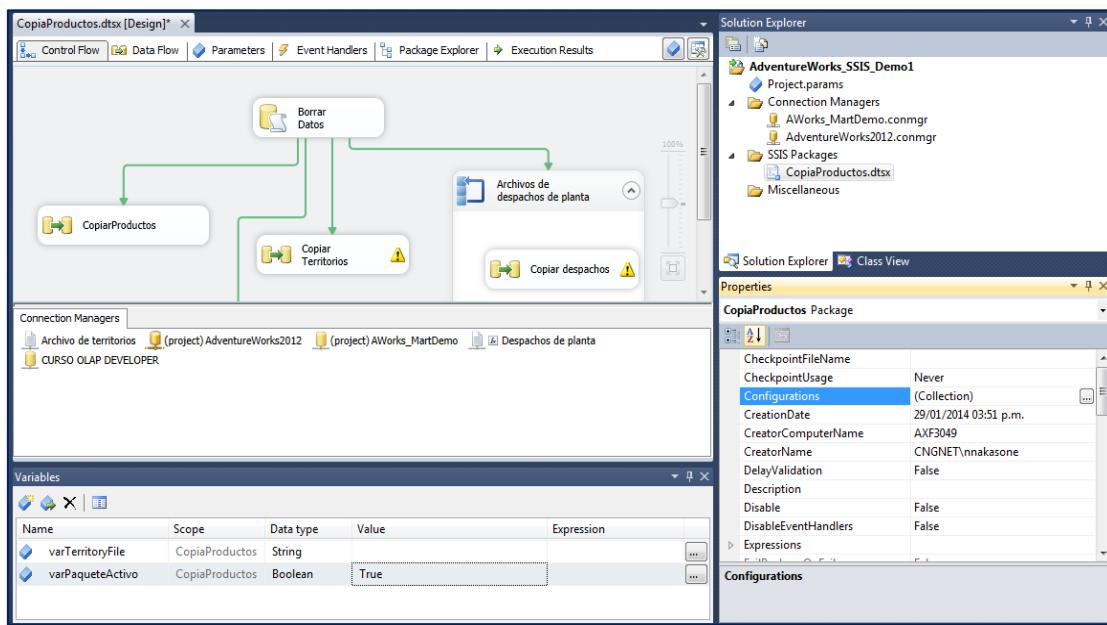
Utilizar Package Configurations, en el proyecto SSIS desarrollado

1. En el paquete “CopiarProductos.dtsx”, agregue una variable llamada “varPaqueteActivo” con las siguientes características:

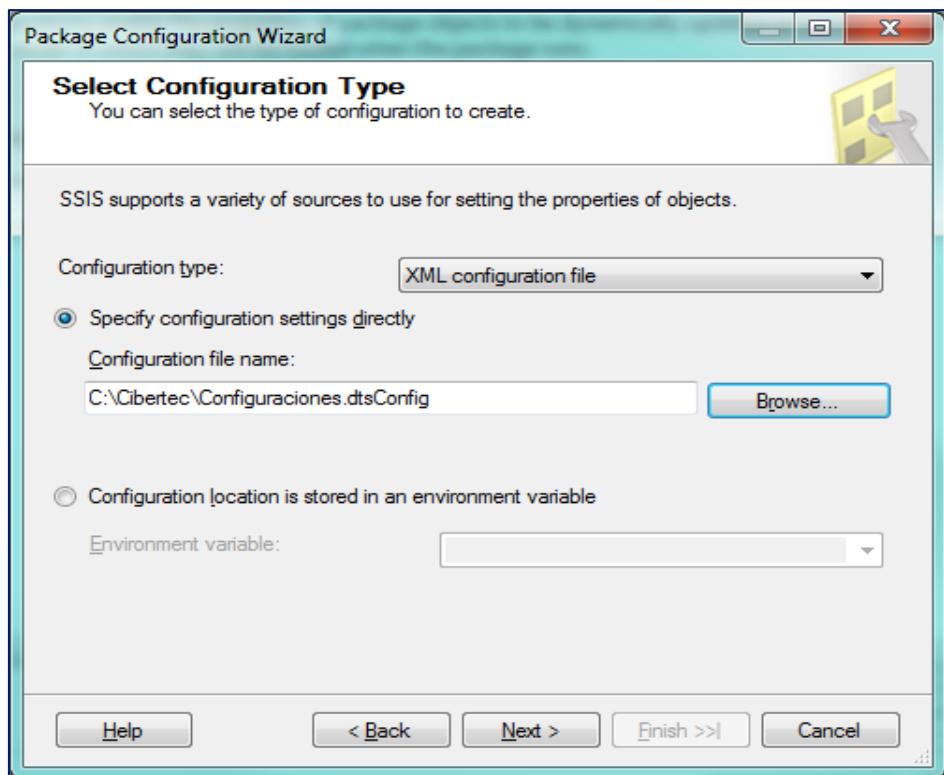
Propiedad	Valor
Name	varPaqueteActivo
Scope	CopiaProductos
Data Type	Boolean
Value	True



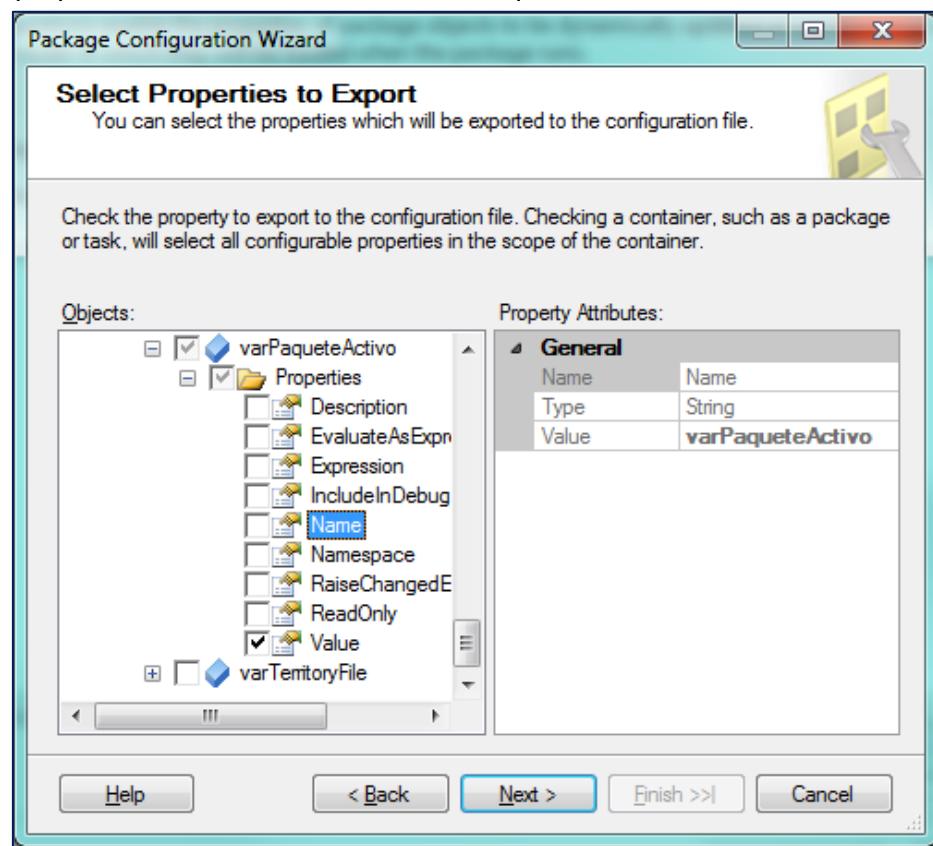
2. Haga un clic sobre la pestaña Control Flow en el entorno de diseño. Y luego en las propiedades del paquete, ubique la propiedad Configurations y haga clic en la elipse.



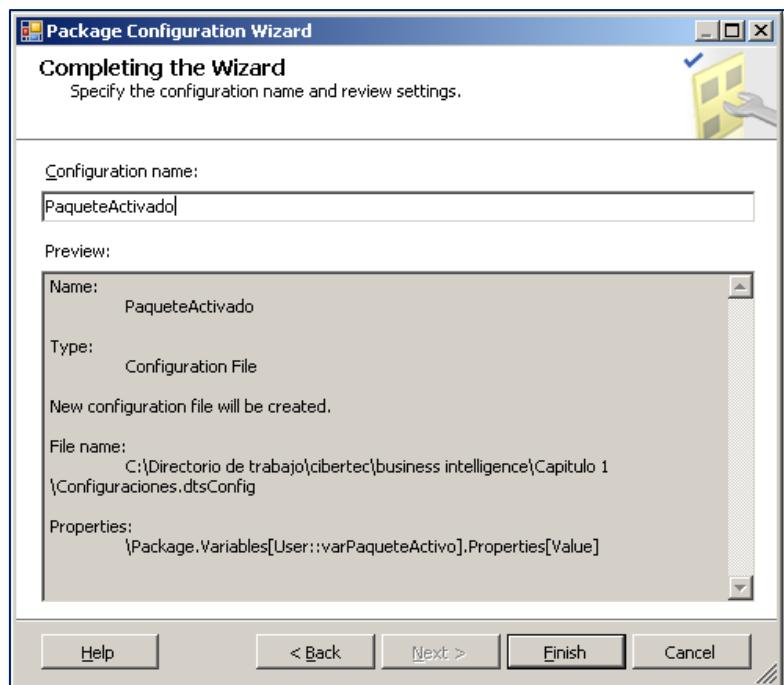
3. Aparecerá la ventana del Package Configurations Organizer. Marque el check box Enable Package Configurations. Para crear una nueva configuración, pulse el botón Add.
4. Aparecerá la ventana inicial del Package Configuration Wizard. Pulse el botón Next.
5. En la siguiente ventana, en la propiedad Configuration Type, seleccione la opción XML Configuration File. Marque la opción Specify configuration settings directly, y utilice el botón Browse para definir la ruta donde desea crear el archivo de configuración. Asigne al archivo el nombre "Configuraciones.dtsConfig".



6. Pulse el botón Next. En la siguiente ventana, en la lista Objects, seleccione la propiedad Value de la variable “varPaqueteActivo”.



7. Pulse el botón Next. En la ventana final del asistente, en la propiedad Configuration name, escriba “PaqueteActivado”.



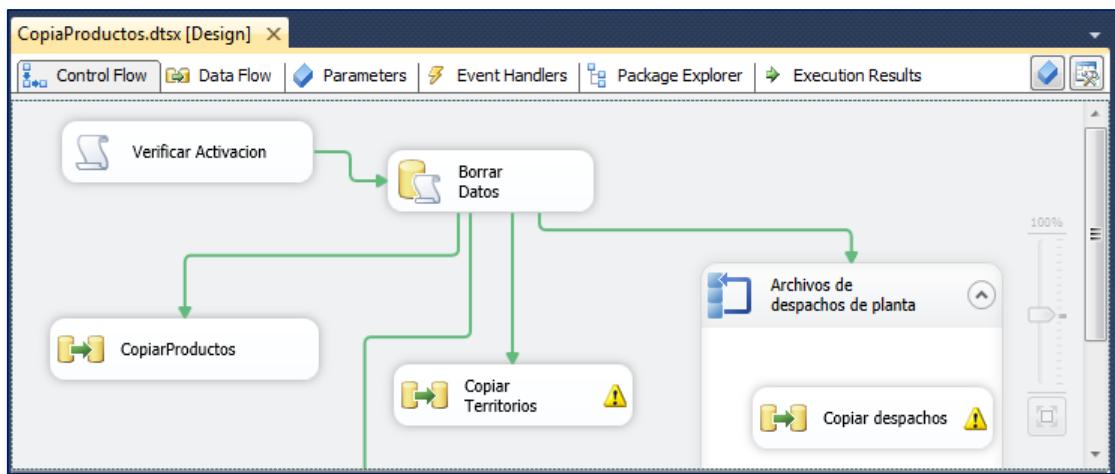
8. Pulse el botón Finish para finalizar el asistente. Pulse el botón Close para cerrar el Package Configurations Organizer.
9. Observe que se ha creado un archivo XML llamado “Configuraciones.dtsConfig” en la ruta seleccionada durante la ejecución del asistente. Abra el archivo utilizando el Internet Explorer. Observe la existencia de un elemento Configuration, de tipo booleano, establecido a -1 (true).

```

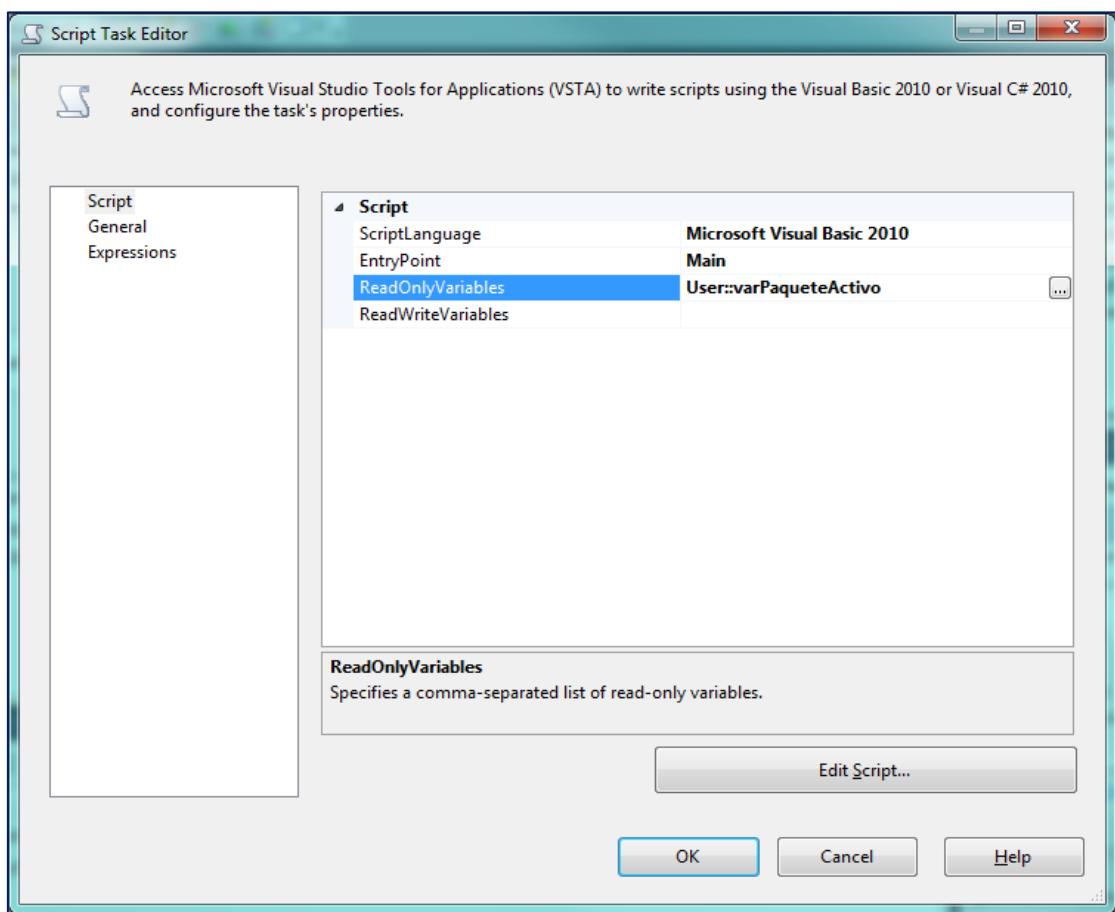
<?xml version="1.0"?>
- <DTSConfiguration>
  - <DTSConfigurationHeading>
    <DTSConfigurationFileInfo GeneratedDate="30/01/2014 04:04:10 p.m." GeneratedFromPackageID="{D2BD124F-FC8B-4736-BEF8-EF57132AD9EE}" GeneratedFromPackageName="CopiaProductos" GeneratedBy="CNGNET\nnakasone"/>
  </DTSConfigurationHeading>
  - <Configuration ValueType="Boolean" Path="\Package.Variables[User::varPaqueteActivo].Properties[Value]" ConfiguredType="Property">
    <ConfiguredValue>-1</ConfiguredValue>
  </Configuration>
</DTSConfiguration>

```

10. Desde el Toolbox, agregue al paquete una tarea de tipo Script Task, y renómbrela como “Verificar activación”. A continuación, agregue a la tarea “Borrar datos” una precedencia con origen en “Verificar activación”.



11. Haga doble clic en la tarea “Verificar Activación”. Se abrirá la ventana del Script Task Editor. En la parte izquierda de la ventana, seleccione la opción Script. En la propiedad ScriptLanguage seleccione Microsoft Visual Basic 2013, luego en la propiedad ReadOnlyVariables, escriba “varPaqueteActivo”. Esto indica que el script tiene acceso de sólo lectura sobre la variable varPaqueteActivo.



12. Pulse el botón Edit Script para escribir el código .Net que define el comportamiento de la tarea. Escriba el siguiente código en el método Main() de la clase ScriptMain:

```

<Microsoft.SqlServer.Dts.Tasks.ScriptTask.SSIScriptTaskEntryPointAttribute()> _
<System.CLSCompliantAttribute(False)> _
Partial Public Class ScriptMain
    Inherits Microsoft.SqlServer.Dts.Tasks.ScriptTask.VSTARTScriptObjectModelBase

    Help: Using Integration Services variables and parameters in a script
    Help: Firing Integration Services events from a script
    Help: Using Integration Services connection managers in a script

    'This method is called when this script task executes in the control flow.
    'Before returning from this method, set the value of Dts.TaskResult to indicate success or failure.
    'To open Help, press F1.

    Public Sub Main()
        '
        ' Add your code here
        '

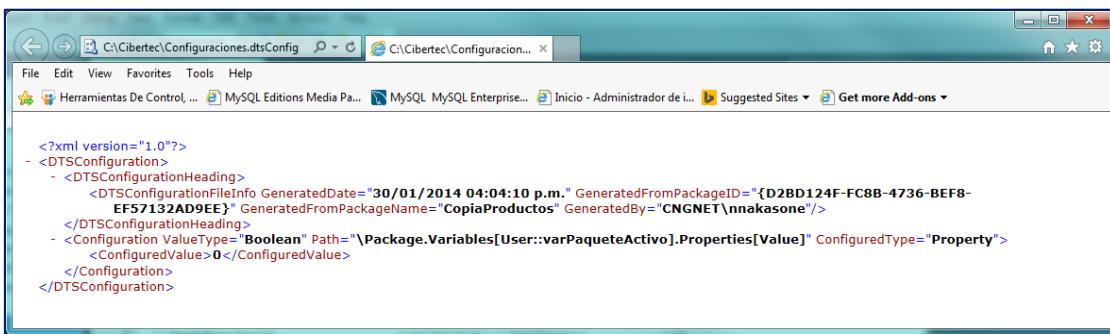
        If CType(Dts.Variables("varPaqueteActivo").Value, Boolean) Then
            Dts.TaskResult = ScriptResults.Success
        Else
            Dts.TaskResult = ScriptResults.Failure
        End If
    End Sub

    ScriptResults declaration

End Class

```

13. Guarde el script y cierre la ventana de .Net. En la ventana del Script Task Editor, pulse el botón OK.
14. Guarde el paquete y ejecútelo. Observe que, como el valor de la variable “varPaqueteActivo” es True, el script task finaliza notificando éxito, y da paso a la ejecución de las tareas siguientes.
15. Abra el archivo “Configuraciones.dtsConfig” utilizando el Notepad. Modifíquelo, para establecer el valor de la configuración para la variable “varPaqueteActivo” a 0 (False). Al finalizar, el archivo debe quedar de la siguiente manera:



16. Ejecute nuevamente el paquete. Observe que, como el valor de la variable “varPaqueteActivo” es False, el script task finaliza notificando fallo, y todo el paquete finaliza su ejecución.

Derived Column

La transformación Derived Column se utiliza para generar nuevas columnas, con base en las columnas de input ya existentes. Esto se hace a través de expresiones, es decir, fórmulas que se aplican a las columnas de input para generar un nuevo valor de output.

El lenguaje de expresiones de SSIS posee un vasto conjunto de funciones y operadores que permite efectuar transformaciones avanzadas en la información de una fila.

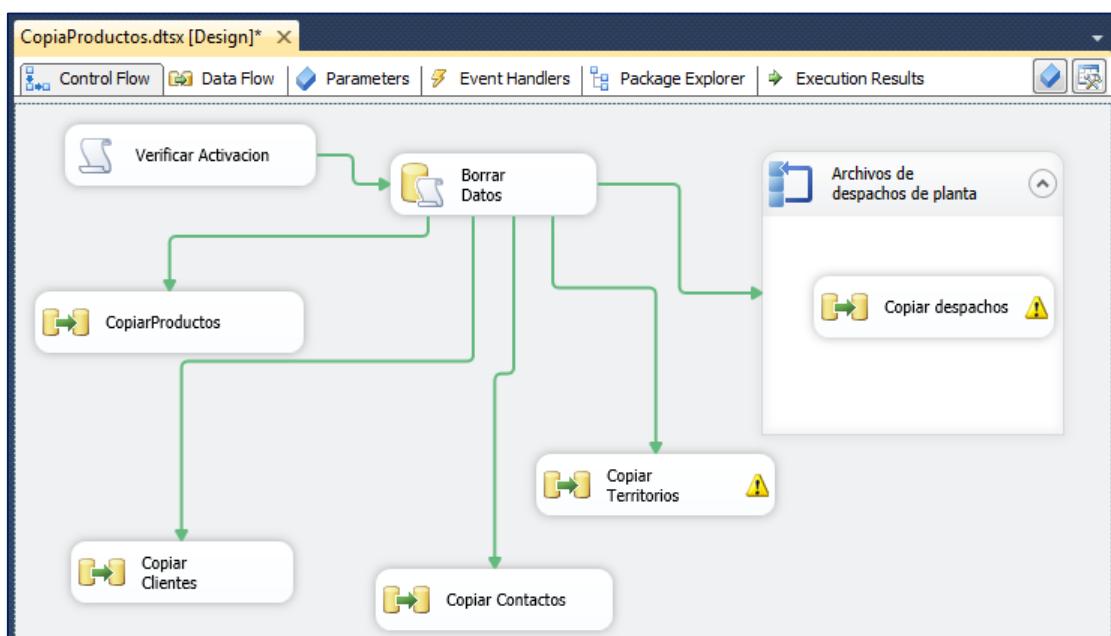
El Derived Column es muy útil para situaciones en las cuales se debe efectuar concatenaciones, extraer una porción de una cadena, obtener partes de una fecha, aplicar funciones matemáticas, etc.

En el siguiente ejercicio, se creará un DerivedColumn para concatenar los valores de las columnas “FirstName” y “LastName” de la tabla Person.Contact, y enviar el resultado hacia una nueva tabla llamada “ResumenContactos”.

Ejercicio

Utilizar Transformación Derived Column en el proyecto SSIS desarrollado.

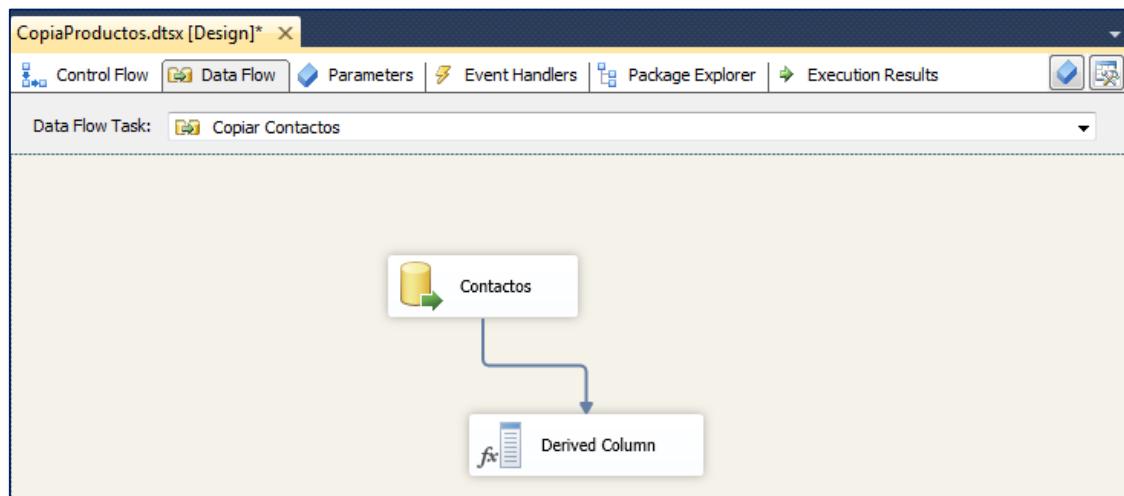
1. En el paquete “CopiarProductos.dtsx”, agregue un Data Flow Task y renómbrelo como “Copiar Contactos”.



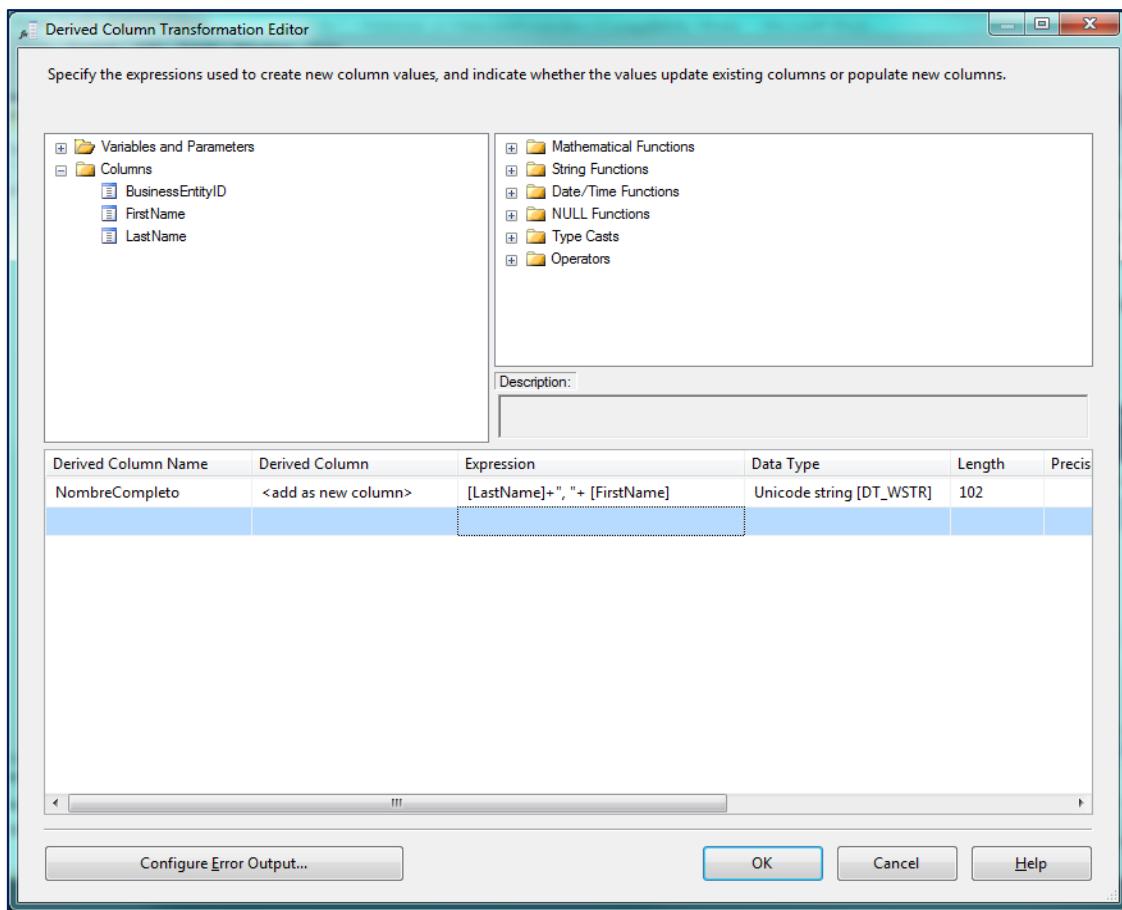
2. En el data flow “Copiar Contactos”, agregue un ADO NET Source con las siguientes propiedades:

Propiedad	Valor
Name	Contactos
ADO.NET Connection Manager	Adventure Works2014
Data access mode	SQL command
SQL command text	Select BusinessEntityID, FirstName, LastName From Person.Person

3. En data flow “Copiar Contactos”, agregue una transformación de tipo Derived Column. Conecte la salida del origen “Contactos” con la transformación que acaba de añadir.



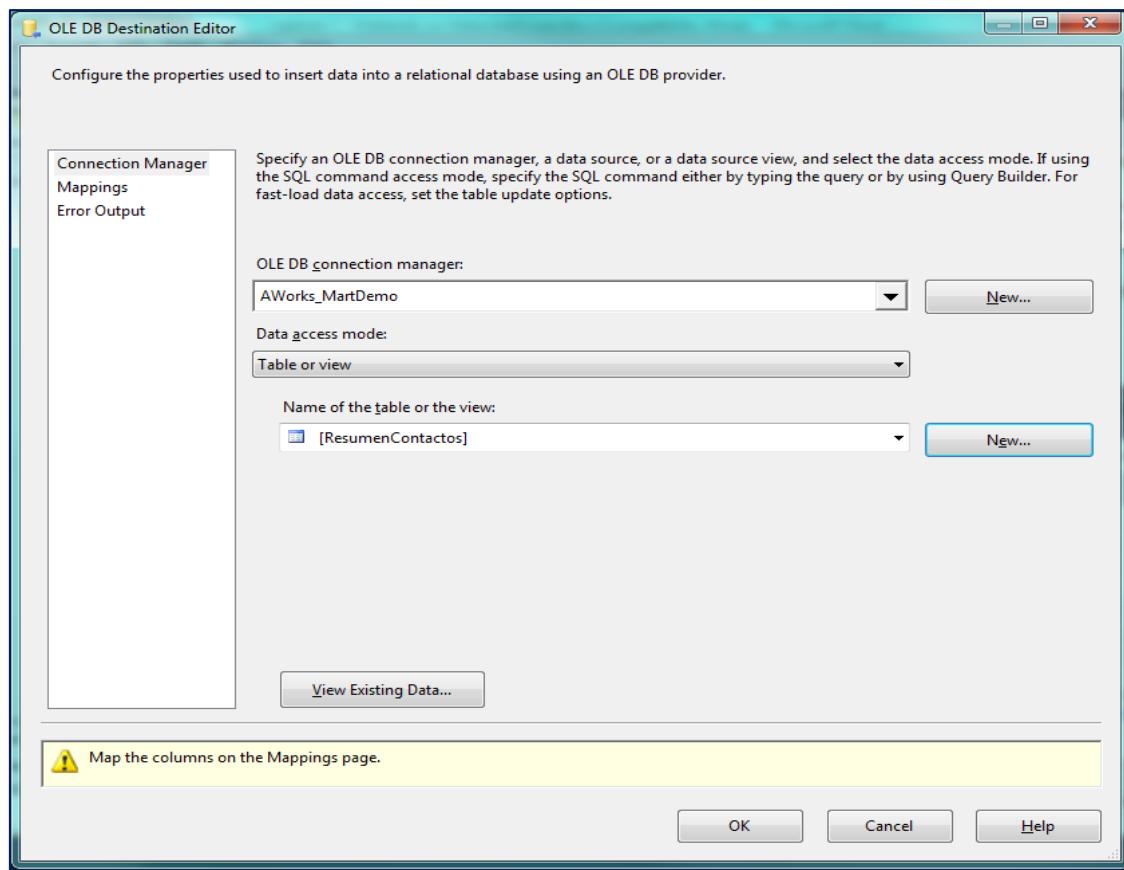
4. Haga doble clic sobre la transformación “Derived Column” para editar sus propiedades. Aparecerá la ventana del Derived Column Transformation Editor. En el grid de la parte inferior, agregue nueva columna derivada con las siguientes propiedades:



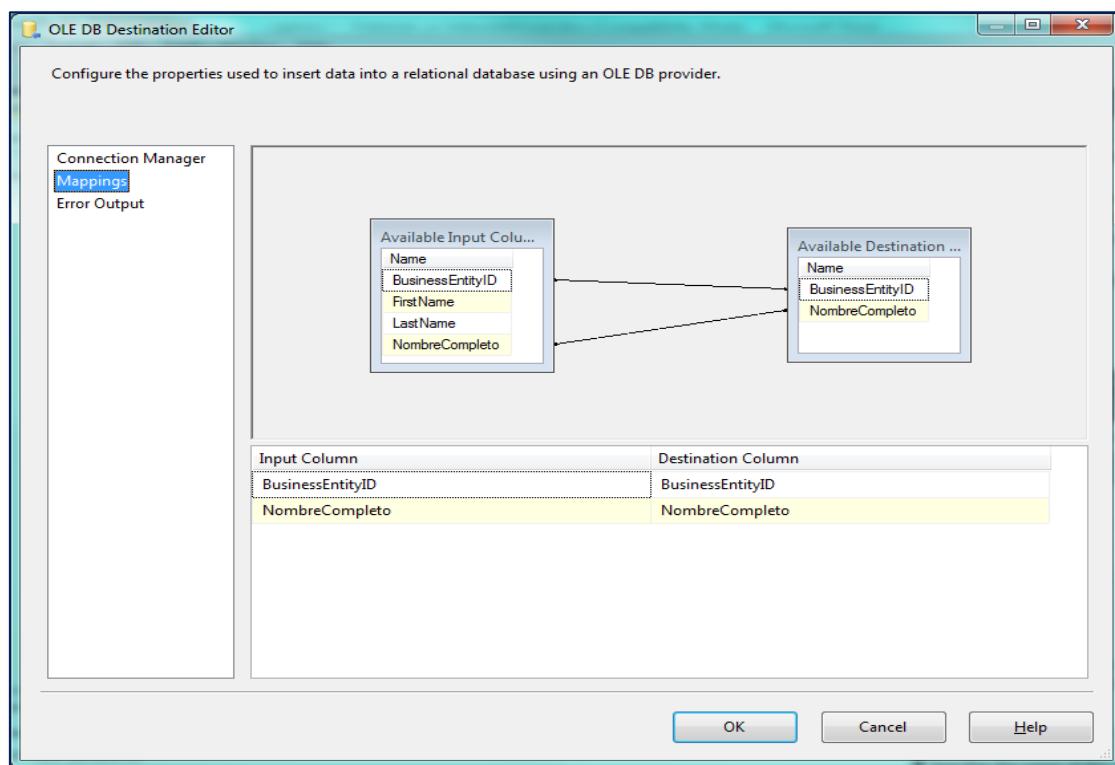
Propiedad	Valor
Derived Column Name	NombreCompleto
Derived Column	<add as new column>
Expression	[LastName] + ", " + [FirstName]
Data Type	Unicode string [DT_WSTR]

5. Pulse el botón OK para confirmar los cambios. A continuación, agregue un OLE DB Destination llamado “Resumen Contactos”. Conecte la salida de la transformación “Derived Column” con la transformación “Resumen Contactos”. Haga doble clic sobre “ResumenContactos” y establezca las siguientes propiedades:

Propiedad	Valor
OLE DB Connection Manager	AWorks MartDemo
Data access mode	Table or view – fast load
Name of the table or the view	Presione el botón New para crear una tabla, con la siguiente sentencia SQL: CREATE TABLE [ResumenContactos](BusinessEntityID Integer, NombreCompleto Nvarchar(150))



6. En la parte izquierda de la ventana, seleccione la opción **Mappings**, y establezca los siguientes mapeos de columnas:



7. Pulse el botón OK para aceptar los cambios. Guarde el paquete y ejecútelo. Compruebe los resultados.

Fuzzy Lookup

Al igual que el Lookup, la transformación Fuzzy Lookup compara los datos de entrada con una tabla de equivalencias. Sin embargo, en Fuzzy Lookup no es necesario que las equivalencias sean exactas; basta con que exista cierta similitud para que Integration Services decida que existe una correspondencia.

La transformación Fuzzy Lookup crea columnas adicionales que almacenan indicadores numéricos de semejanza. Esta característica permite que el desarrollador del paquete establezca políticas de tipo “si la semejanza es mayor que 90%, la decisión tomada por Integration Services es correcta. Para las filas restantes, el administrador tomará la decisión final”.

Ejercicio:

Utilizar la transformación Fuzzy Lookup en el proyecto SSIS desarrollado.

En este ejercicio, se implementa el siguiente escenario: periódicamente, los vendedores remiten información en archivos de texto. Esta información consiste en el nombre de cada cliente, y la venta correspondiente. Sin embargo, es común que estos nombres contengan errores ortográficos y de digitación.

Adicionalmente, se tiene una tabla llamada “Clientes”, que contiene la información de los clientes de la empresa. Este ejercicio consistirá en construir un paquete de Integration Services que lea la información de un archivo de texto, y efectúe correctamente el mapeo entre los clientes en este archivo y los clientes en la tabla “Clientes”, basado en la similitud de sus valores.

Por ejemplo, en esta tabla se tienen los siguientes registros:

IdCliente NombreCliente

2 Karina Palacios Ramos
3 Jessica Linares Hidalgo

En el archivo de ventas, la información viene en la siguiente forma:

Cliente, Ventas
Karina Palasios Ramos, 124500
Jesica Linares Hidalgo, 89076

Obsérvese que en el archivo de texto, el apellido en la primera fila es “Palasios” en lugar de “Palacios”. En la segunda fila, el nombre es “Jesica” en lugar de “Jessica”. El paquete que se construirá tendrá la capacidad de determinar el cliente correcto para cada venta en el archivo de texto con base en la similitud de su nombre con los registros existentes en la tabla “Clientes”.

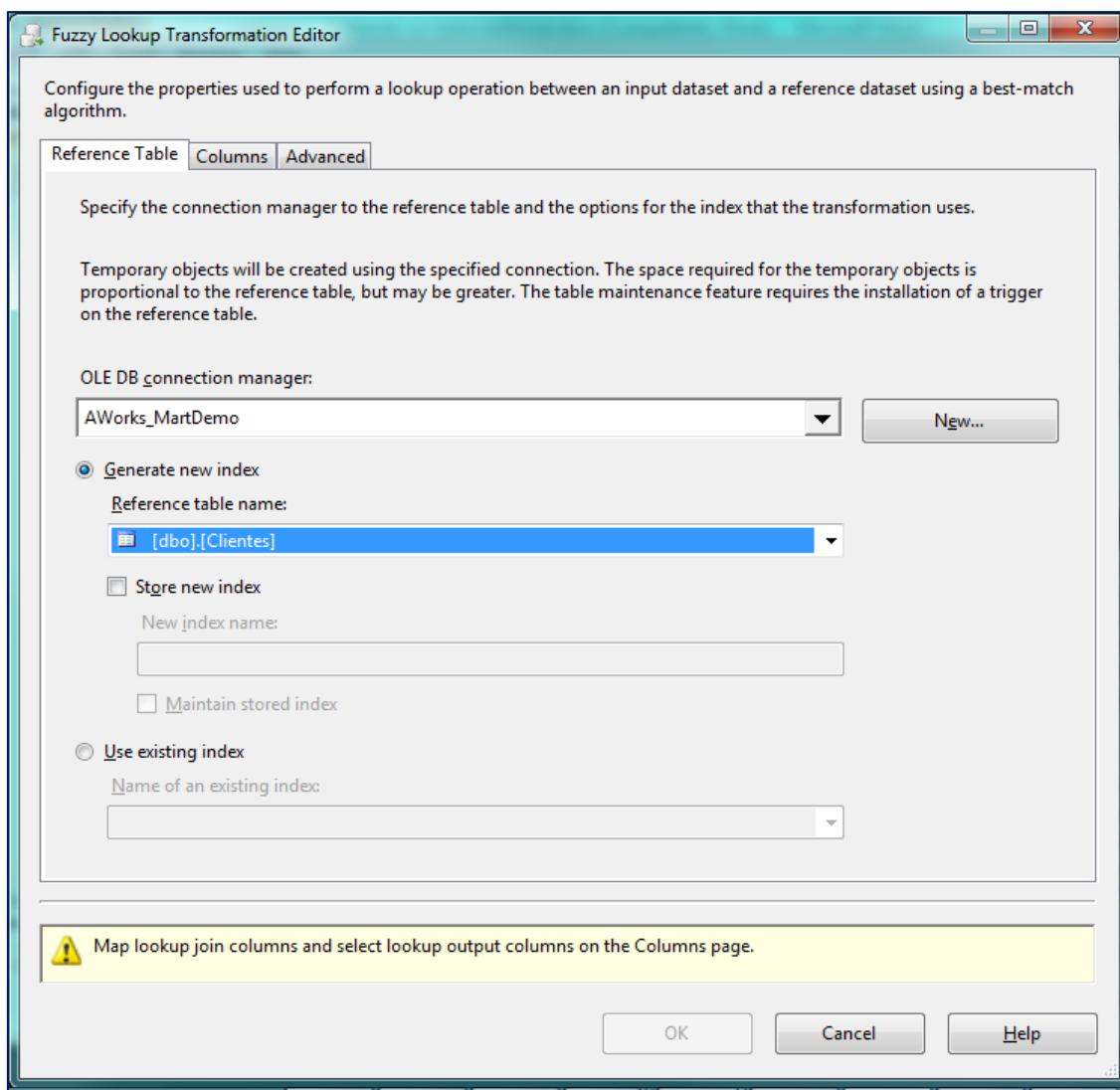
1. Agregue al proyecto un nuevo paquete llamado “Demo Fuzzy Lookup”.
2. Agregue un nuevo Flat File Connection Manager con las siguientes características:

Propiedad	Valor
Name	Ventas Sucio
Archivo	Seleccione el archivo “Ventas-sucio.txt”
Text Qualifier	Ninguno
Column names in the first data row	Sí

3. En el paquete, cree una tarea de tipo Data Flow Task. Asignele el nombre “Limpiar Datos de Ventas”.
4. Dentro de la tarea “Limpiar Datos de Ventas”, agregue un Flat File Source con las siguientes propiedades:

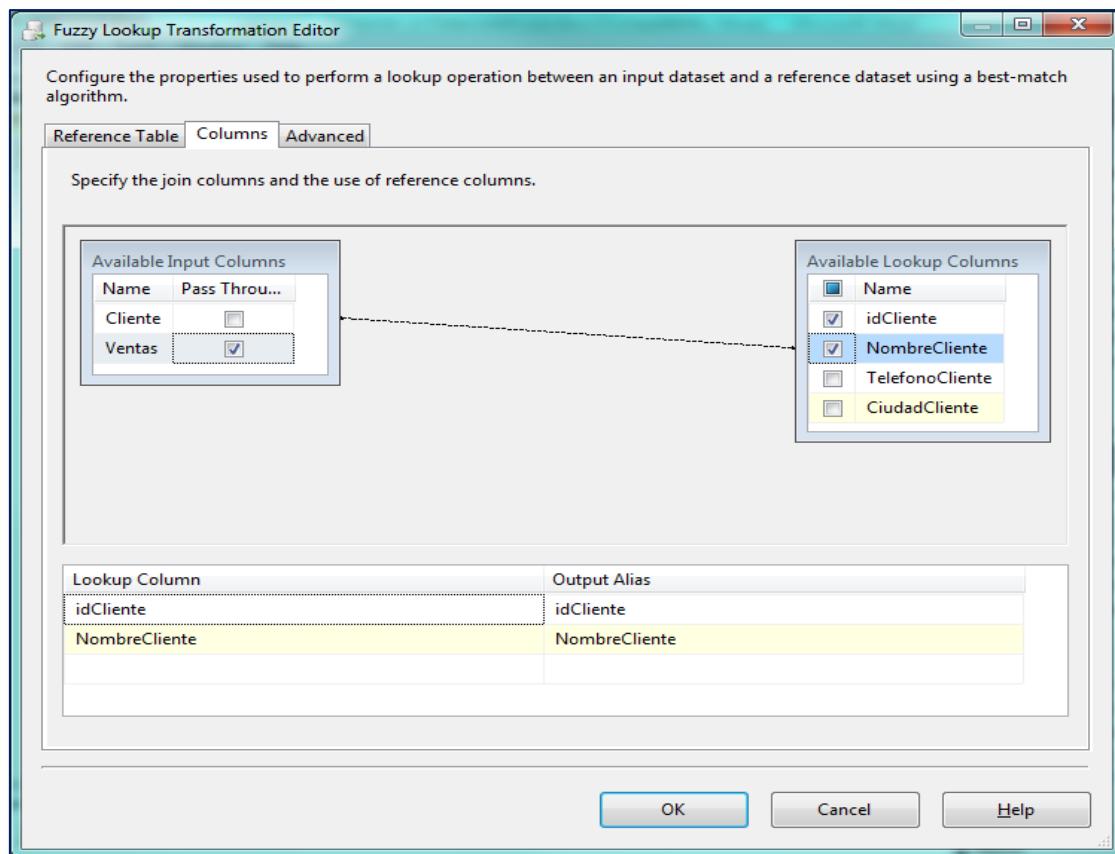
Propiedad	Valor
Name	Archivo de Ventas
Flat File Connection Manager	Seleccione el connection manager “Ventas Sucio”

5. Dentro de la tarea “Limpiar Datos de Ventas”, agregue una transformación de tipo Fuzzy Lookup, y establezca su nombre a “Corregir datos”. Luego, seleccione el origen “Archivo de Ventas” creado en el paso anterior, y conecte su flecha de salida con la transformación “Corregir datos”.
6. Haga doble click sobre la transformación “Corregir datos”. Seleccione la pestaña Reference Table. En la propiedad OLE DB connection manager, seleccione el connection manager “AWorks MartDemo”. En la propiedad Reference table name, seleccione la tabla “Clientes”. Al finalizar, la pantalla debe tener la siguiente apariencia:

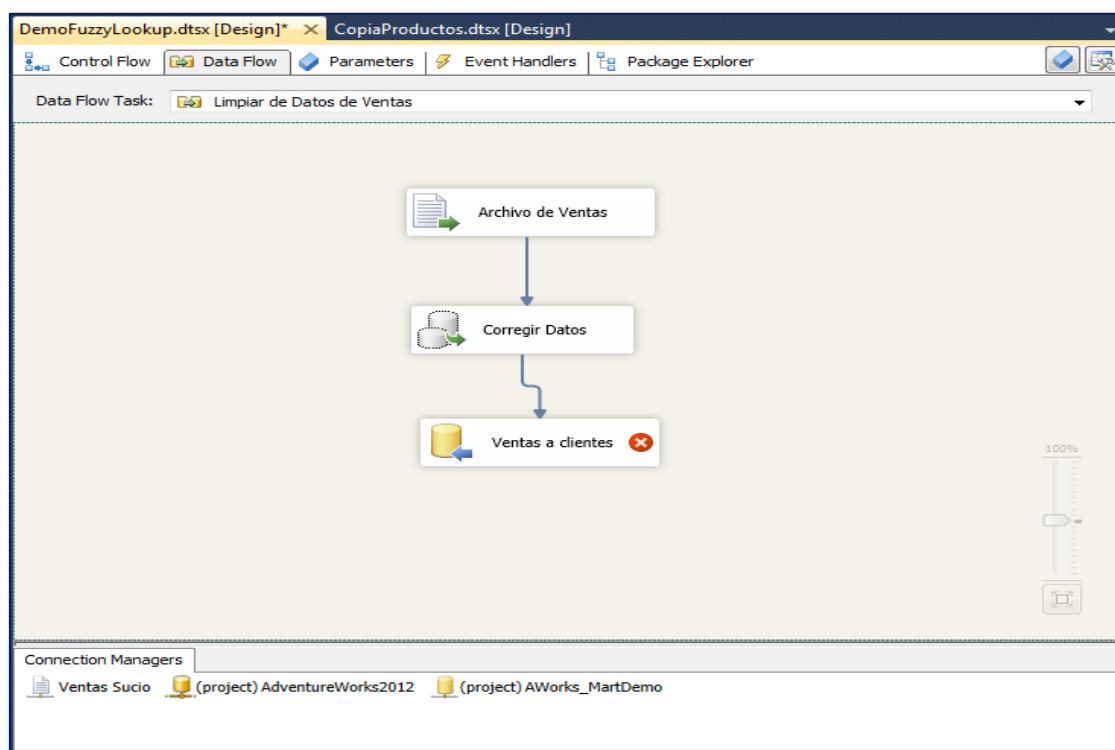


7. Seleccione la pestaña Columns. Observe que se muestran dos tablas: una llamada "Available Input Columns" que corresponde con el archivo de texto "Ventas-Sucio.txt"; y otra llamada "Available Lookup Columns" que corresponde con la tabla "Clientes". Realice los siguientes pasos:
 - Ligue la columna "Cliente" de "Available Input Columns" con la columna "NombreCliente" de "Available Lookup Columns".
 - En "Available Input Columns", desactive la casilla de verificación de la columna "Cliente".
 - En "Available Lookup Columns", active la casilla de verificación de las columnas "IdCliente" y "NombreCliente".

Al finalizar, la ventana debe tener la siguiente apariencia:



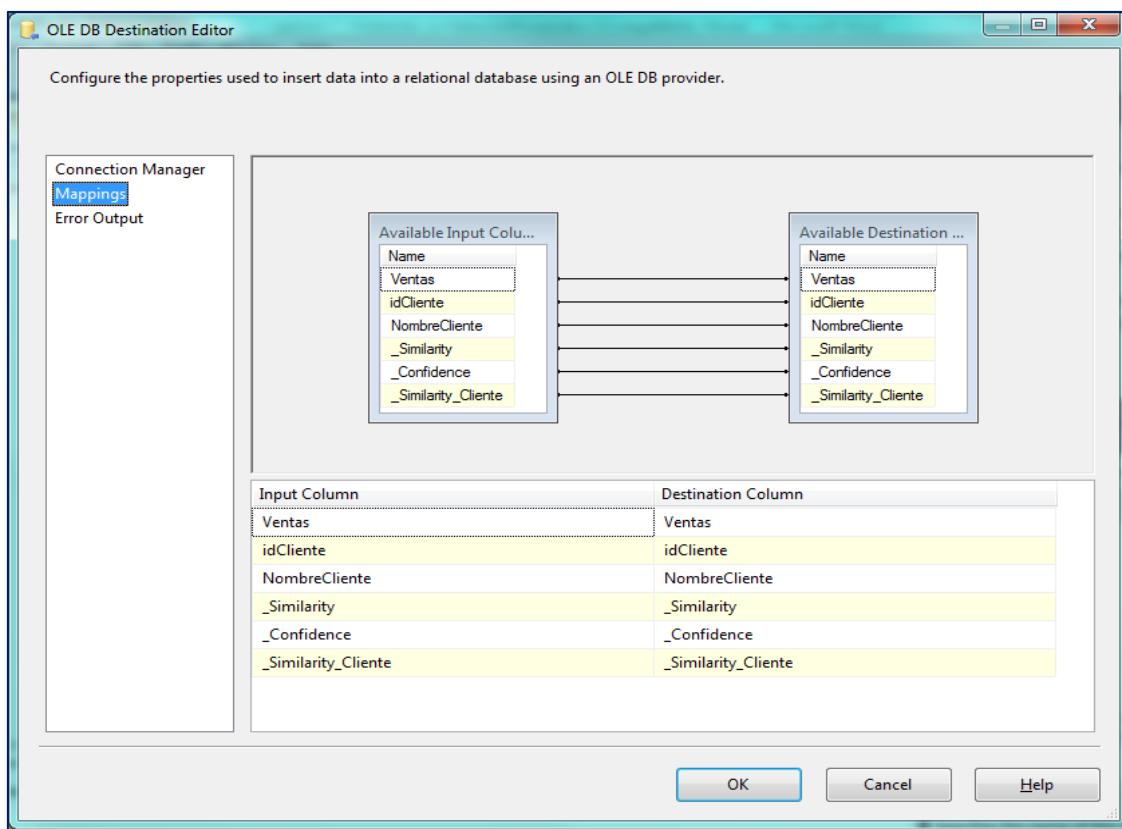
8. Pulse el botón OK.
9. A continuación, agregue un OLE DB Destination y establezca su nombre a “Ventas a Clientes”. Seleccione la transformación “Corregir datos” y conecte su flecha de salida con el destino “Ventas a Clientes”. Al finalizar, la ventana Data Flow debe tener la siguiente apariencia:



10. Configure las propiedades del destino “Ventas a clientes” de la siguiente manera:

Propiedad	Valor
OLE DB Connection Manager	AWorks MartDemo
Data access mode	Table or view – fast load
Name of the table or the view	Presione el botón New para crear una tabla, con la siguiente sentencia SQL: CREATE TABLE [VentasClientes] ([Ventas] VARCHAR(50), [IdCliente] INTEGER, [NombreCliente] VARCHAR(50), [_Similarity] REAL, [_Confidence] REAL, [_Similarity_Cliente] REAL)

11. En el destino “Ventas a clientes”, establezca los siguientes mapeos de columnas:



12. Guarde el paquete y ejecútelo. Consulte los registros de la tabla VentasClientes. Verifique que la limpieza de los datos provenientes del archivo de texto “Ventas-sucio.txt” se ha efectuado correctamente. Observe los valores de las columnas “_Similarity”, “_Confidence” y “_Similarity_Cliente”.

Agregate y Sort

Al igual que el Lookup, la transformación Fuzzy Lookup compara los datos de entrada con una tabla de equivalencias. Sin embargo, en Fuzzy Lookup no es necesario que

las equivalencias sean exactas; basta con que exista cierta similitud para que Integration Services decida que existe una correspondencia.

La transformación Fuzzy Lookup crea columnas adicionales que almacenan indicadores numéricos de semejanza. Esta característica permite que el desarrollador del paquete establezca políticas de tipo “si la semejanza es mayor que 90%, la decisión tomada por Integration Services es correcta.

Para las filas restantes, el administrador tomará la decisión final”. Una necesidad común durante las extracciones de datos consiste en efectuar agregaciones (sumas, promedios, etc.), y ordenar la información. Este tipo de operación puede efectuarse desde la misma consulta de datos (usando operadores como ORDER BY, GROUP BY y SUM). Sin embargo, existen varias circunstancias en las cuales se necesitará efectuar estas operaciones durante el proceso de transferencia de datos y no en la consulta de origen.

- Cuando el origen de datos no soporta operaciones de agregación y ordenamiento. Por ejemplo: archivos de texto, Excel.
- Si se requiere agregar y ordenar datos que provienen de operaciones previas sobre múltiples orígenes de datos.
- El uso de operadores ORDER BY, GROUP BY, etc., hacen que la consulta sobre el origen de datos sea muy lenta.

Las transformaciones Aggregate y Sort sirven para efectuar agregaciones y ordenaciones sobre la data presente en el flujo de datos del paquete SSIS.

Ejercicio:

Utilizar la transformación Aggregate y Sort en el proyecto SSIS desarrollado.

En este ejercicio, se efectuará una agregación sobre un archivo de texto que contiene información como la siguiente:

IdPlanta, IdProducto, IdCliente, Cantidad
1, 178, "87123490112", 4
1, 179, "76130920114", 2
...
4, 178, "87123490112", 9
4, 179, "76130920114", 5

Obsérvese que cada cliente tiene información en dos plantas (1 y 4). Se desea obtener el total de la columna “Cantidad” para cada cliente. Por ejemplo, el total de “Cantidad” para el primer cliente sería 13 (la suma de 4 y 9). La salida de datos estará ordenada descendente por cantidad.

1. Agregue al proyecto un nuevo paquete llamado “Demo Aggregate y Sort”.
2. Agregue un nuevo Flat File Connection Manager con las siguientes características:

Propiedad	Valor
Name	Despachos Detallados
Archivo	Seleccione el archivo “Detalle Despachos.txt”
Text Qualifier	“ (comilla doble)
Column names in the first data row	Sí

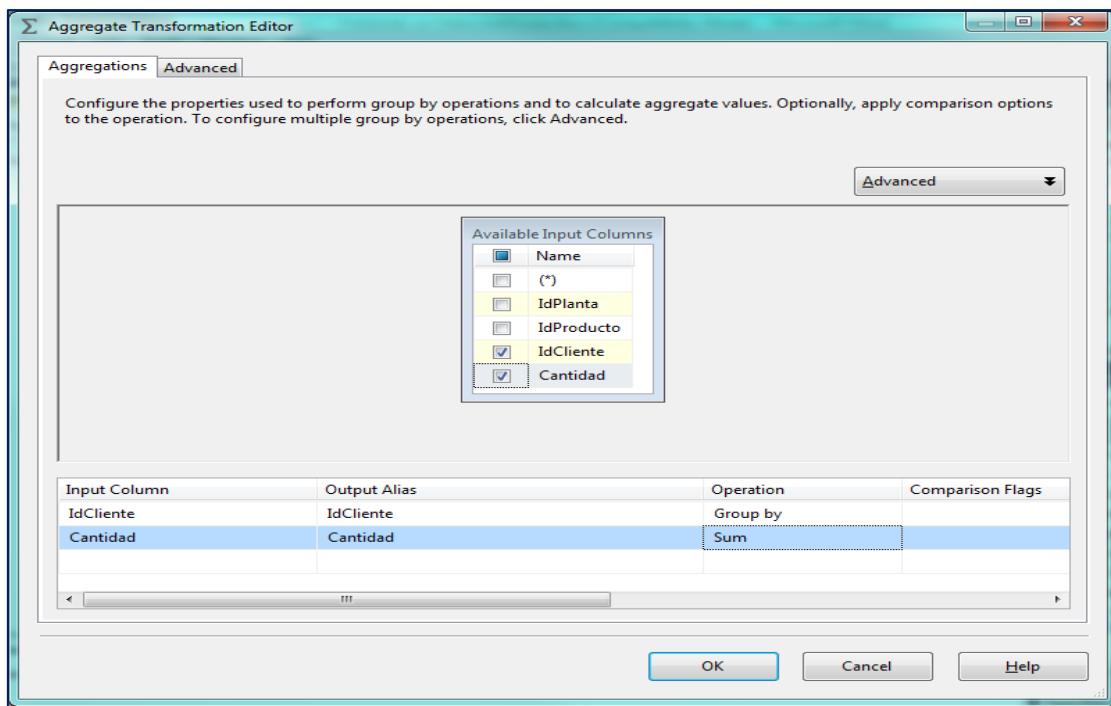
3. Continúe la edición del connection manager “Despachos Detallados”, seleccionando la opción Advanced en el panel izquierdo. Establezca las siguientes propiedades para cada columna en el archivo de texto:

Propiedad	Valor
IdPlanta	DataType: four-byte signed integer [DT_I4]
IdProducto	DataType: four-byte signed integer [DT_I4]
IdCliente	DataType: string [DT_STR] OutputColumnWidth: 11
Cantidad	DataType: decimal [DT_DECIMAL]

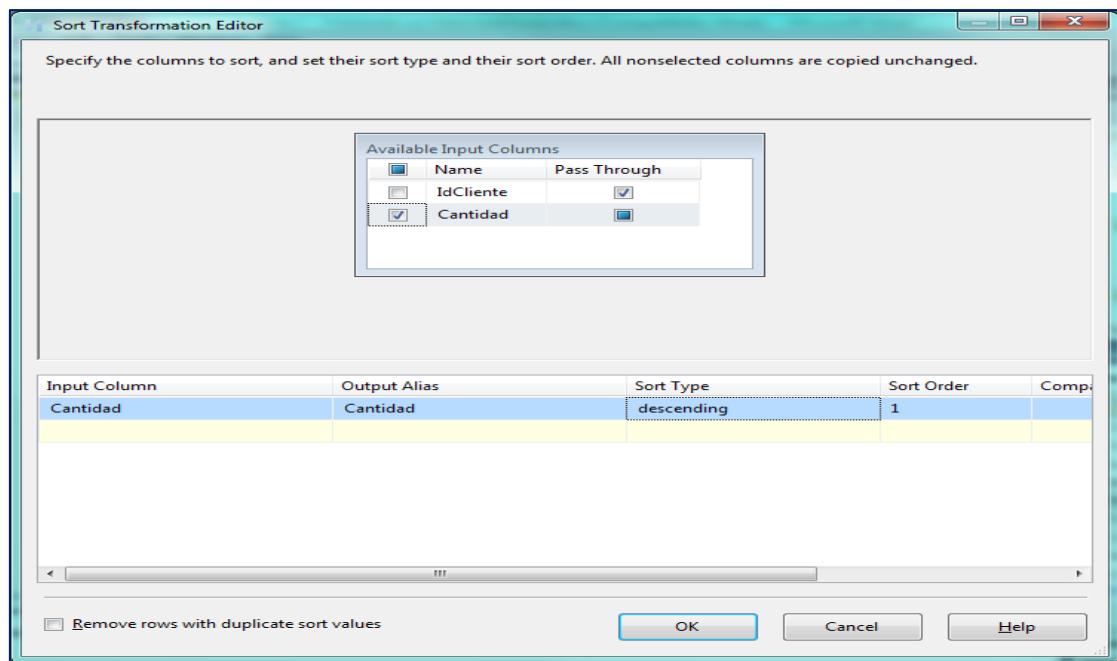
4. Pulse OK para finalizar la edición del connection manager “Despachos Detallados”.
 5. En el paquete, cree una tarea de tipo Data Flow Task. Asígnele el nombre “Agrupar y ordenar despachos por clientes”.
 6. Dentro de la tarea “Agrupar y ordenar despachos por clientes”, agregue un Flat File Source con las siguientes propiedades:

Propiedad	Valor
Name	Archivo de despachos
Flat File Connection Manager	Seleccione el connection manager “Despachos Detallados”

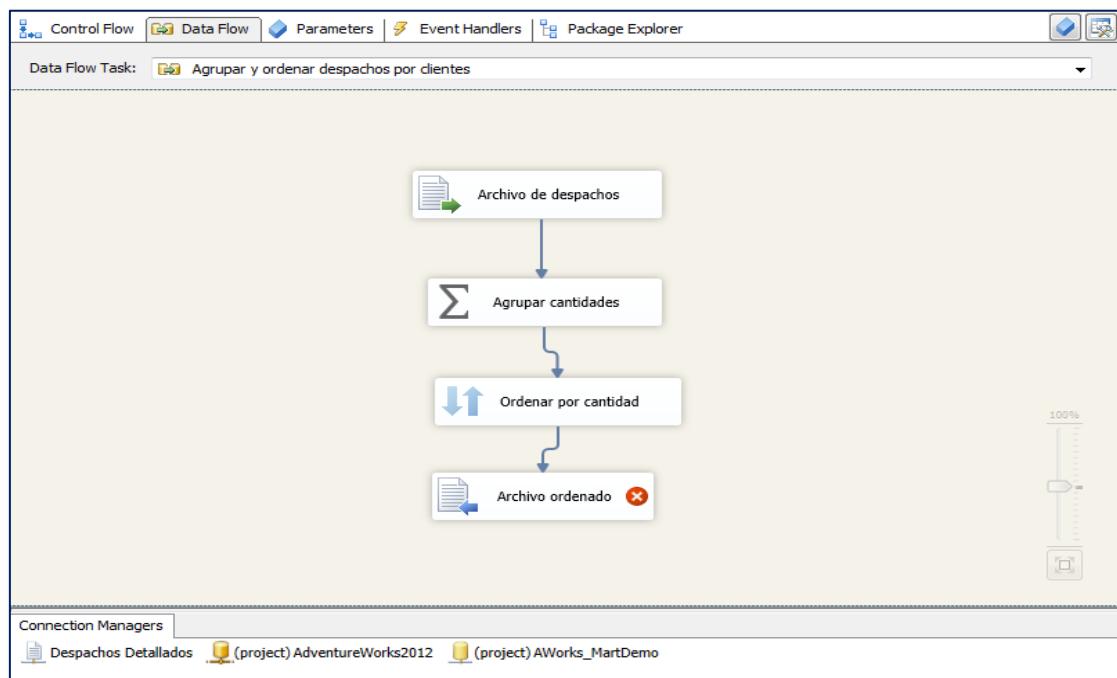
7. Dentro de la tarea “Agrupar y ordenar despachos por clientes”, agregue una transformación de tipo Aggregate, y establezca su nombre a “Agrupar cantidades”. Luego, seleccione el origen “Archivo de despachos” creado en el paso anterior, y conecte su flecha de salida con la transformación “Agrupar cantidades”.
 8. Haga doble clic sobre la transformación “Agrupar cantidades” para editar sus propiedades. En la tabla Available Input Columns, active la casilla de verificación de las columnas “IdCliente” y “Cantidad”. En la parte inferior del editor, establezca el valor de Operation para la columna “IdCliente” a Group by. Establezca también el valor de Operation para la columna “Cantidad” a Sum. Esto significa que la transformación obtendrá la suma de la columna “Cantidad”, agrupada por cada cliente. Al finalizar, el editor debe tener la apariencia que se muestra a continuación:



9. Pulse el botón OK para finalizar la edición de la transformación “Agrupar cantidades”.
10. A continuación, agregue una transformación de tipo Sort, y establezca su nombre a “Ordenar por cantidad”. Seleccione la transformación “Agrupar cantidades” y conecte su flecha de salida con la transformación “Ordenar por cantidad”.
11. Haga doble clic sobre la transformación “Ordenar por cantidad” para editar sus propiedades.
12. En la tabla Available Input Columns, active la casilla de verificación a la izquierda de la columna “Cantidad”. En la parte inferior del editor, establezca el valor de Sort Type a descending para la columna “Cantidad”. Esto significa que esta transformación ordenará los datos descendenteamente por la columna “Cantidad”. Al finalizar, el aspecto del editor debe ser el siguiente:



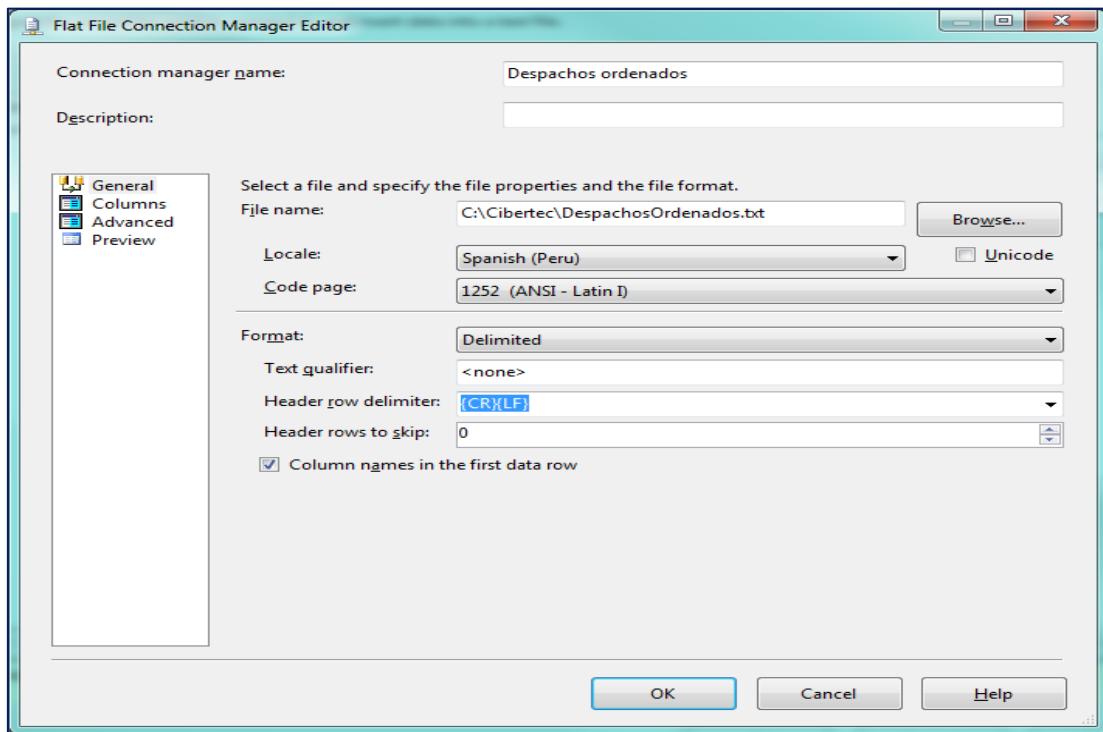
13. Pulse el botón OK para finalizar la edición de la transformación. A continuación, agregue un Flat File Destination, y establezca su nombre a “Archivo ordenado”. Seleccione la transformación “Ordenar por cantidad” y conecte su flecha de salida con el destino “Archivo ordenado”. Al finalizar, la ventana Data Flow debe tener la siguiente apariencia:



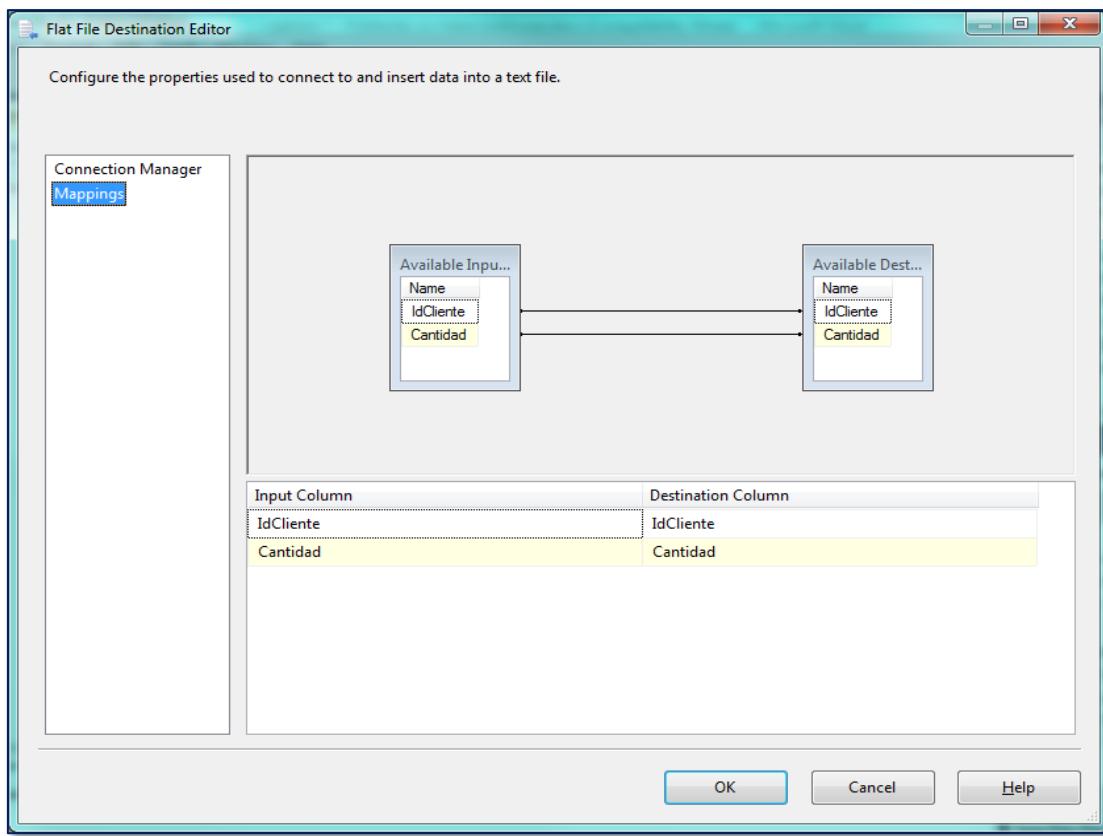
14. Haga doble clic sobre el destino “Archivo ordenado” para editar sus propiedades. En el editor, pulse el botón New para crear un nuevo connection manager. En la ventana Flat File Format, seleccione la opción Delimited. Pulse el botón OK.
 15. En la ventana de edición del nuevo connection manager, establezca las siguientes propiedades:

Propiedad	Valor
Connection Manager Name	Despachos ordenados
File Name	Seleccione una ruta cualquiera. Establezca el nombre del archivo a "DespachosOrdenados.txt"
Column Names in the First data row	Sí

Al finalizar, el editor debe quedar de la siguiente manera:



16. Pulse el botón OK. A continuación, seleccione la opción Mappings en el panel izquierdo. Los mapeos de columnas deben quedar como se muestra a continuación:



17. Pulse el botón OK. Guarde el paquete y ejecútelo. El paquete generará un archivo de texto llamado “DespachosOrdenados.txt” donde la información estará sumarizada por cantidad y agrupada por cliente. Verifique los resultados.

Merge Join

La transformación Merge Join permite combinar dos filas provenientes de orígenes de datos distintos en una sola fila con base en una o más columnas que tengan valores comunes. Por ejemplo, pueden tenerse las siguientes filas en un archivo de texto:

1, 67000
2, 78000

Donde la primera columna representa el ID del cliente, y la segunda columna representa las ventas totales. Por otro lado, en una tabla de clientes se tienen las siguientes filas:

IdCliente	NombreCliente
1	Jorge González Barrera
2	María Flores Herrera

A través de la transformación Merge Join, las filas provenientes del archivo de texto pueden combinarse con las filas de la tabla con base en los valores de las columnas que contienen el ID del cliente para producir un único conjunto de resultados con la siguiente estructura:

IdCliente	NombreCliente	Ventas
1	Jorge González Barrera	67000
2	María Flores Herrera	78000

En este caso, las columnas “IdCliente” del archivo de texto y la tabla reciben el nombre de Join Keys debido a que son usadas para definir la regla de combinación.

La implementación de una transformación Merge Join tiene las siguientes condiciones:

- Las columnas que cumplen el papel de Join Keys deben tener el mismo tipo de datos.
- Las filas que ingresan a la transformación Merge Join deben estar ordenadas con base en los valores de las columnas que cumplen el papel de Join Keys.

Ejercicio

Utilizar la transformación Merge Join en el proyecto SSIS desarrollado.

En este ejercicio, se leerán las filas existentes en un archivo de texto que contiene información como la siguiente:

IdSubCategoria, Ventas
 1, 879765
 2, 678596
 3, 987432

Estas filas se combinarán con los registros provenientes de la tabla “Production.ProductSubCategory” que tienen la siguiente estructura:

ProductSubCategoryID Name
 1 Mountain Bikes
 2 Road Bikes
 3 Touring Bikes

La salida producida por la transformación Merge Join contendrá el Id de la subcategoría, las ventas y el nombre de la subcategoría. Se utilizará el Id de la subcategoría como condición de combinación (join key).

Realice los siguientes pasos:

1. En el proyecto “AdventureWorks_SSIS_Demo1” cree un nuevo Connection Manager con nombre “AdventureWorks – OLE DB” con las siguientes propiedades:

Propiedad	Valor
Provider	NATIVE OLE DB\SQL Native Client 11.0
Server Name	localhost
Database Name	AdventureWorks2014
Name	Adventure Works2014 – OLE DB

2. Agregue al proyecto un nuevo paquete llamado “Demo Merge Join”.
3. Agregue un nuevo Flat File Connection Manager con las siguientes características:

Propiedad	Valor
Name	Archivo de Ventas
Archivo	Seleccione el archivo “VentasSubCategoria.txt”
Text Qualifier	Ninguno
Column names in the first data row	Sí

4. Continúe la edición del connection manager “Archivo de Ventas”, seleccionando la opción Advanced en el panel izquierdo. Establezca las siguientes propiedades para cada columna en el archivo de texto:

Columna	Valor
IdSubCategoria	DataType: four-byte signed integer [DT_I4]
Ventas	DataType: decimal [DT_DECIMAL]

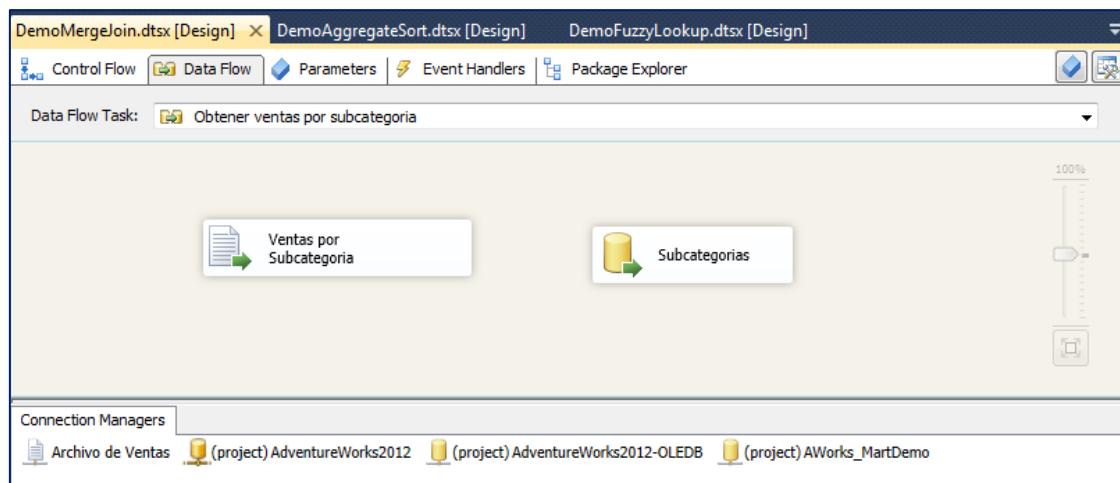
5. Pulse OK para finalizar la edición del connection manager “Archivo de Ventas”. En el paquete, cree una tarea de tipo Data Flow Task. Asígnale el nombre “Obtener ventas por subcategoría”.
6. Dentro de la tarea “Obtener ventas por subcategoría”, agregue un Flat File Source con las siguientes propiedades:

Propiedad	Valor
Name	Ventas por Subcategoria
Flat File Connection Manager	Seleccione el connection manager “Archivo de Ventas”

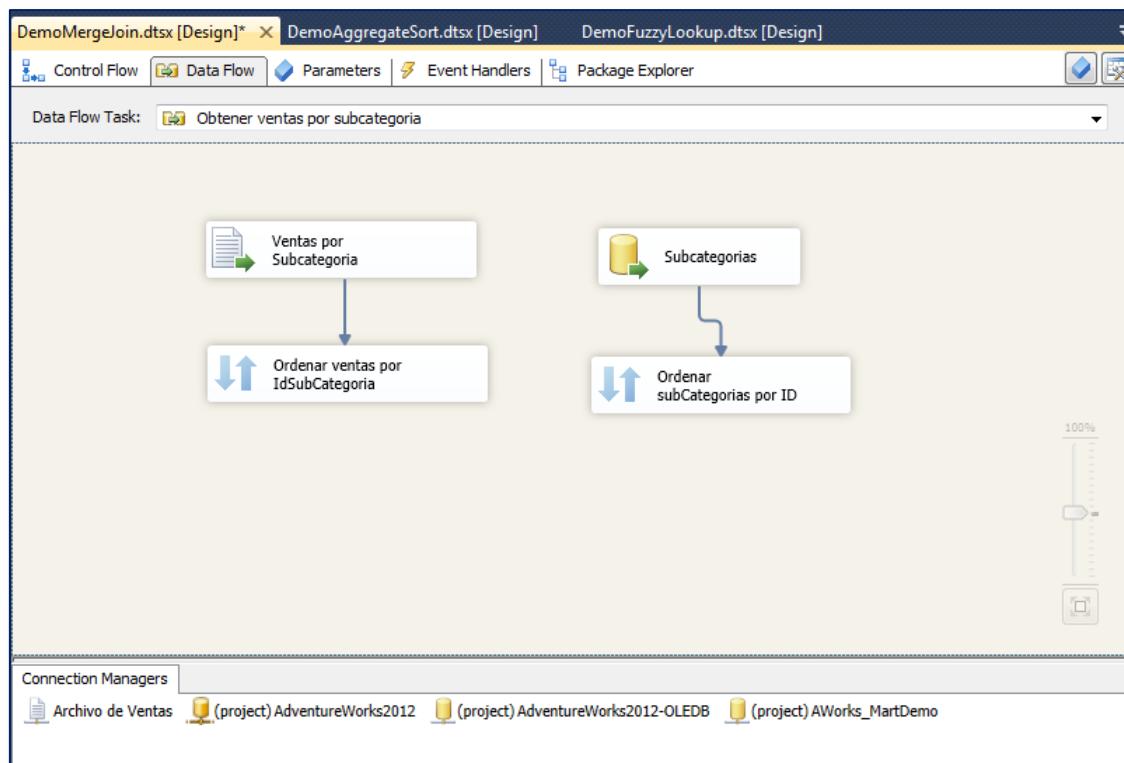
7. Dentro de la tarea “Obtener ventas por subcategoría”, agregue un OLE DB Source con las siguientes propiedades:

Propiedad	Valor
Name	Subcategorias
OLE DB Connection Manager	Seleccione el connection manager “AdventureWorks - OLEDB”
Data Access Mode	Table or view
Name of the table or the view	[Production].[ProductSubCategory]
Columns	Marque únicamente las columnas “ProductSubCategoryID” y “Name”.

Al terminar, el diseñador debe tener la siguiente apariencia:

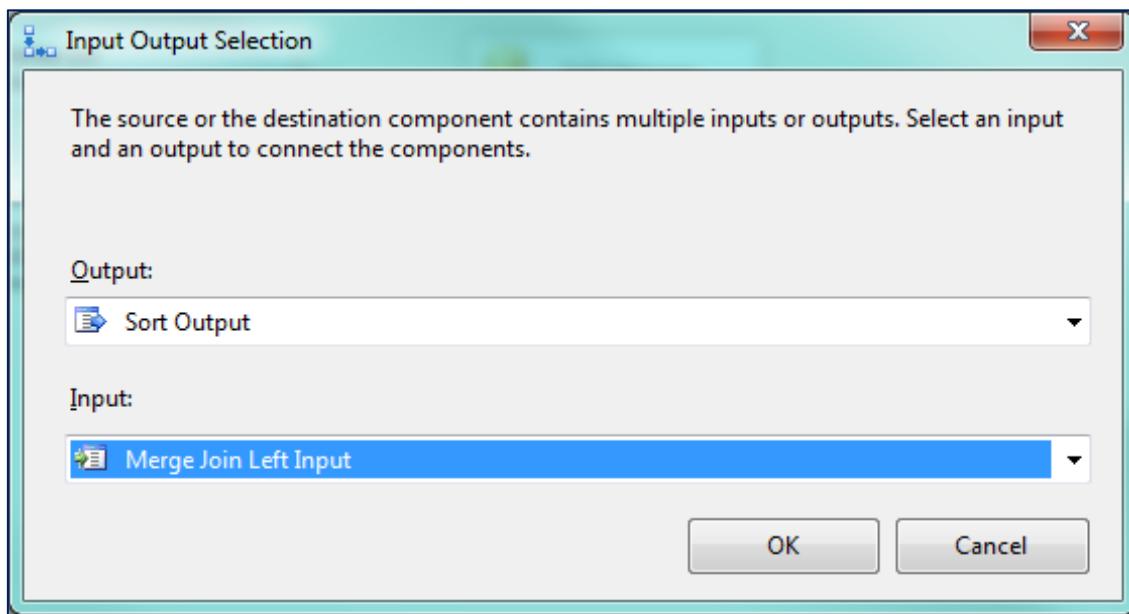


8. En la tarea “Obtener ventas por subcategoría”, agregue una transformación de tipo Sort con nombre “Ordenar ventas por IdSubCategoria”. Luego, seleccione el origen “Ventas por subcategoría” y una su salida con la transformación Sort que acaba de crear. Configure la transformación Sort para ordenar los datos ascendenteamente por la columna “IdSubCategoria”.
9. Agregue otra transformación de tipo Sort con nombre “Ordenar subcategorías por ID”. Luego, seleccione el origen “Subcategorias”, y una su salida con la transformación Sort que acaba de crear. Configure la transformación Sort para ordenar los datos ascendenteamente por la columna “ProductSubCategoryID”. Al terminar, el diseñador debe tener la siguiente apariencia:

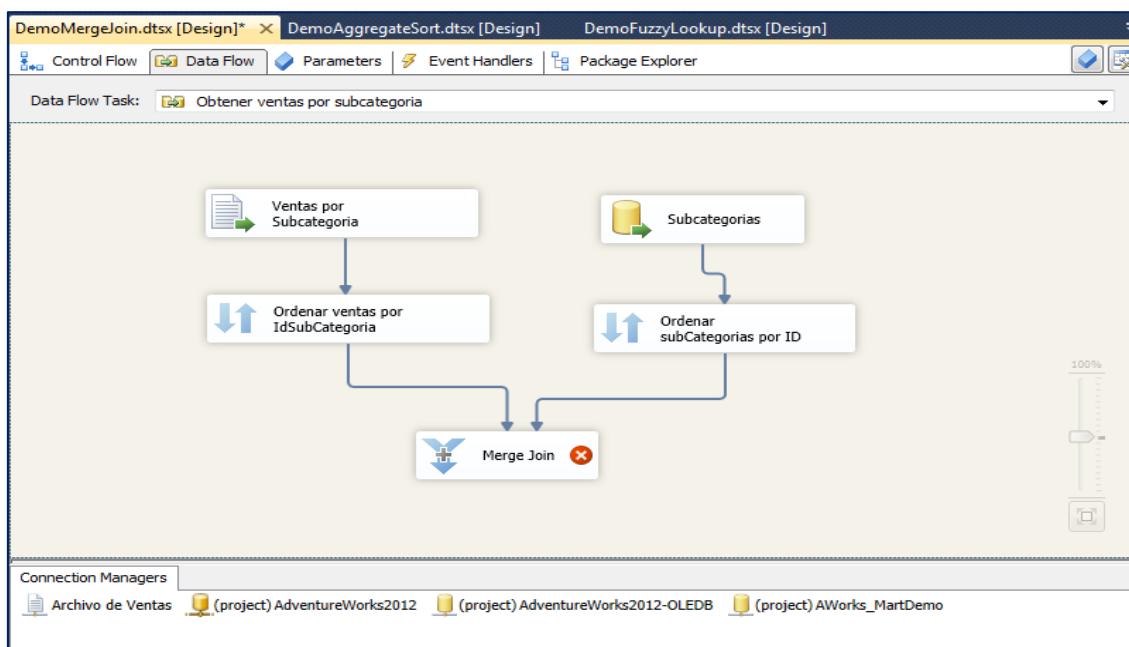


10. Agregue una transformación de tipo Merge Join. Luego, seleccione la transformación Sort llamada “Ordenar ventas por IdSubCategoria” y una su salida con la transformación Merge Join que acaba de crear. Aparecerá la ventana Input

Output Selection; establezca el valor de la propiedad Input a “Merge Join Left Input” como se muestra en la siguiente figura:

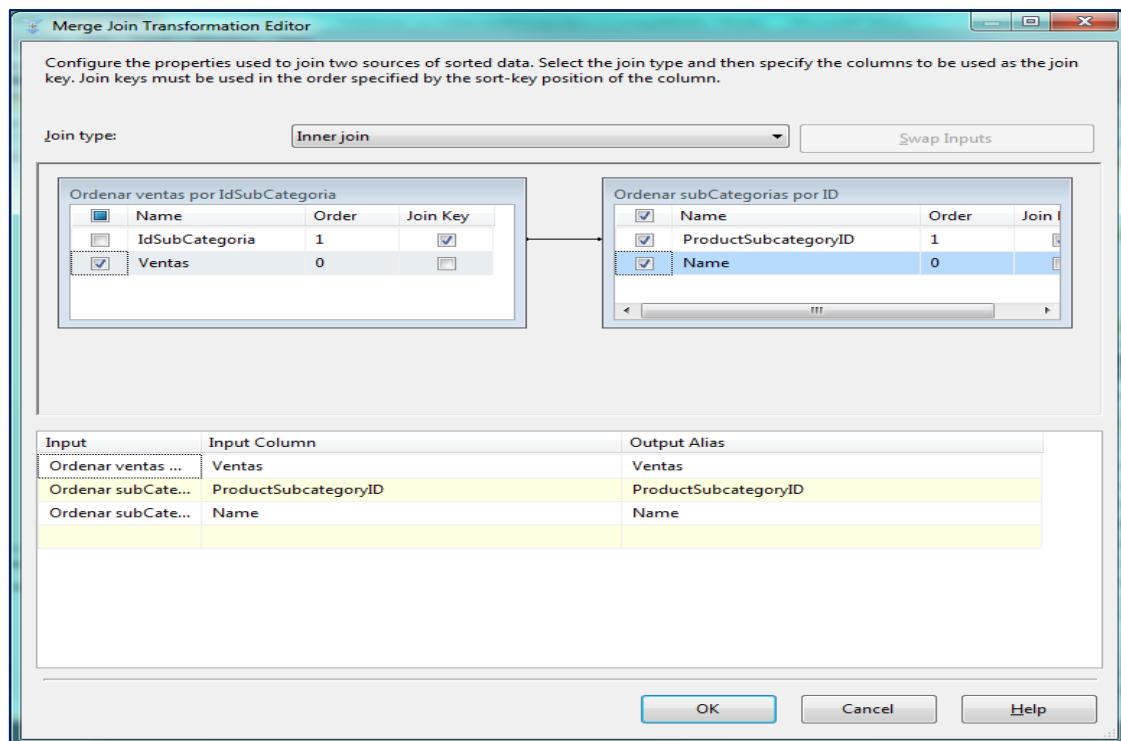


11. Pulse el botón OK. A continuación, seleccione la transformación Sort llamada “Ordenar subcategorías por ID” y una su salida a la transformación Merge Join. Al finalizar, el diseñador debe tener la siguiente apariencia:



12. Haga doble clic sobre la transformación “Merge Join”. Observe que la condición de join consiste en el enlace entre la columna “IdSubCategoria” (proveniente del archivo de texto) y la columna “ProductSubCategoryID” (proveniente de la tabla ProductSubCategory). Este enlace se ha creado automáticamente, debido a que estas son las dos únicas columnas que tienen el mismo tipo de dato (Integer). Sin embargo, esta relación puede ser cambiada por el desarrollador, si es necesario.

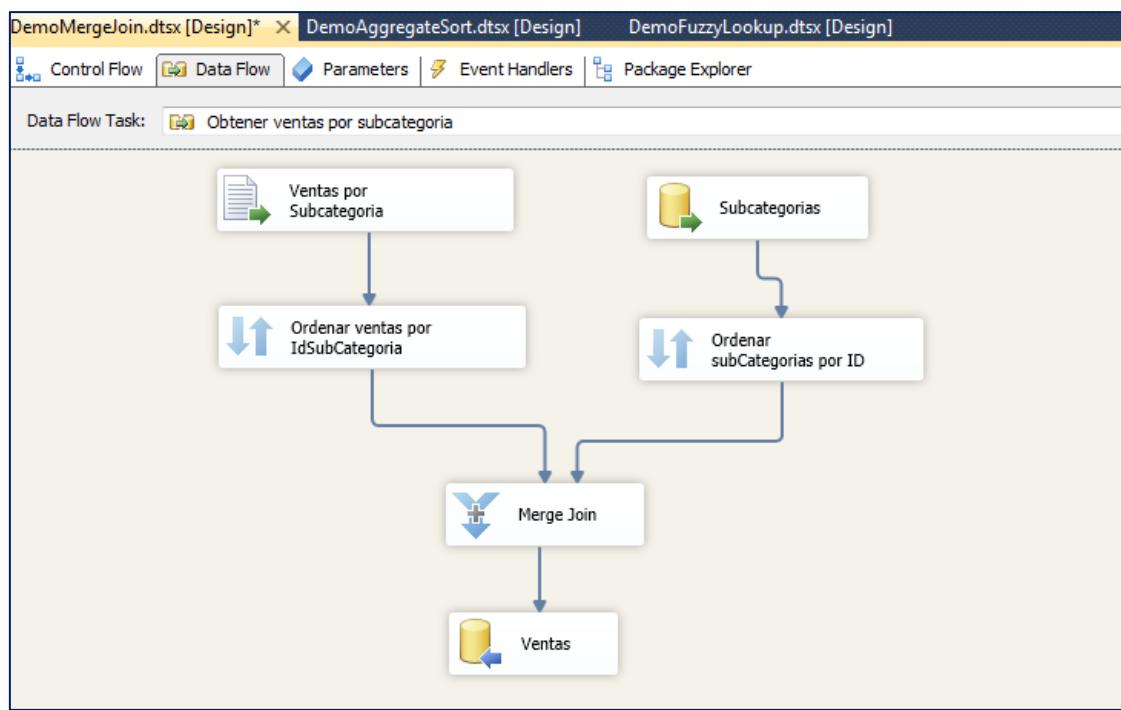
En el editor, marque los check boxes que se encuentran a la izquierda de las columnas “ProductSubCategoryID”, “Name”, y “Ventas”. Al finalizar, el editor debe tener la siguiente apariencia:



13. Pulse el botón OK para finalizar la edición.
14. Agregue al paquete un destino de tipo OLE DB Destination con nombre “Ventas”. Luego, seleccione la transformación “Merge Join” y una su salida con el destino que acaba de crear. Establezca las propiedades del destino “Ventas” como se muestra a continuación:

Propiedad	Valor
OLE DB Connection Manager	Seleccione AWorks MartDemo
Data Access Mode	Table or view – Fast load
Name of the table or the view	Pulse el botón New, y cree una nueva tabla con la siguiente definición: CREATE TABLE [VentasSubcategoria] ([ProductSubcategoryID] INTEGER, [Name] NVARCHAR(50), [Ventas] DECIMAL)

15. Una vez finalizada la edición del destino “Ventas”, el diseñador debe tener la siguiente apariencia:



16. Guarde el paquete y ejecútelo. Compruebe los resultados haciendo una selección de los registros en la tabla “VentasSubCategoria” de la base de datos AWorks_MartDemo.

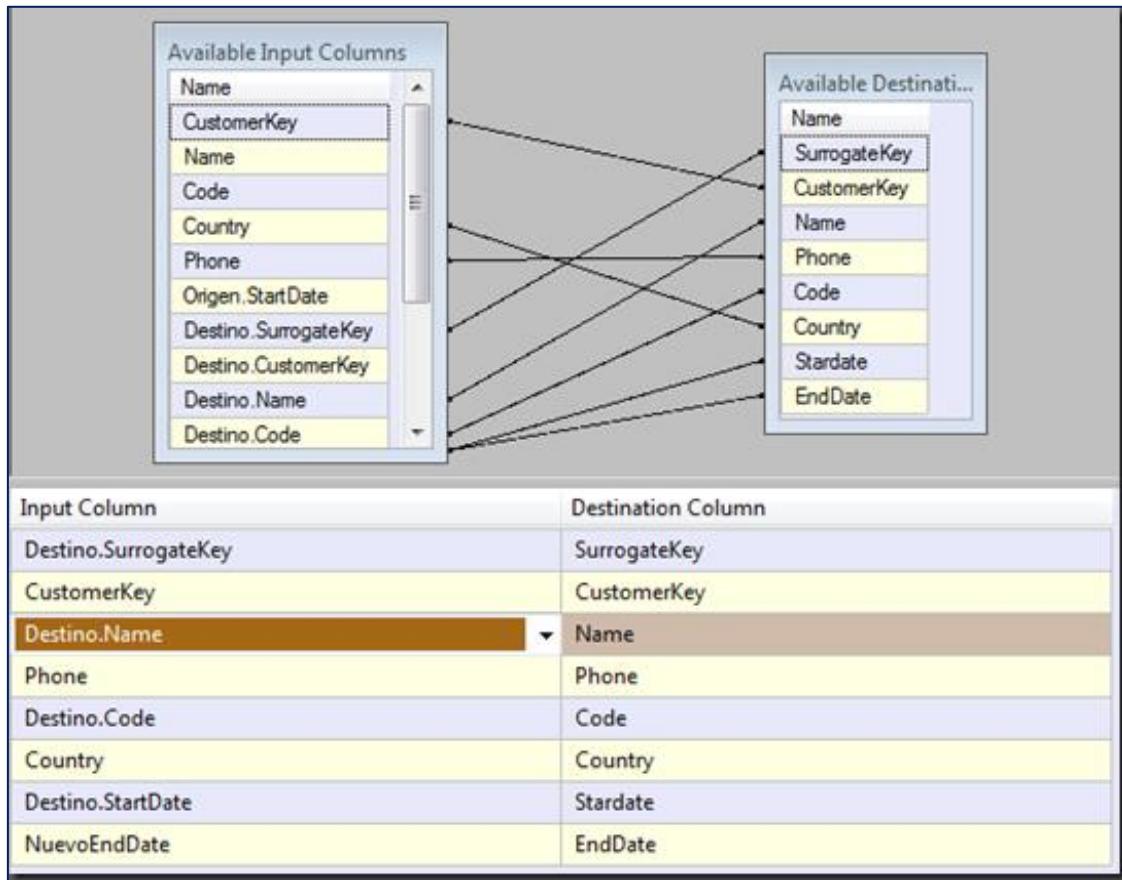
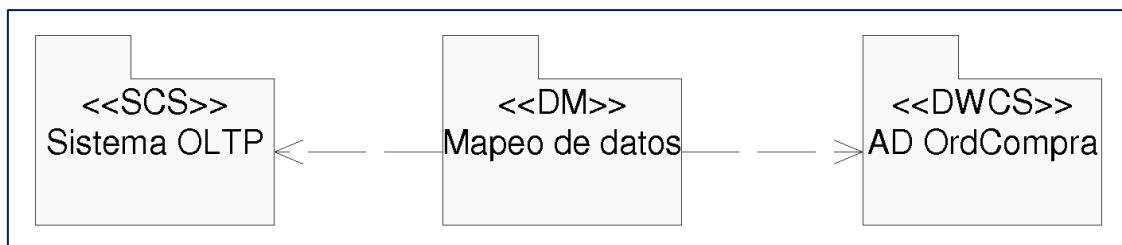
3.2. PROCESO DE CARGA

El proceso de carga es el proceso que permite a las organizaciones mover datos desde múltiples fuentes, reformatearlos y limpiarlos, y cargarlos en otra base de datos, data mart, o data warehouse para analizar, o en otro sistema operacional para apoyar un proceso de negocio..

3.2.1. Diseño de un proceso de carga

3.2.1.1 Mapeo de Datos

Mediante herramientas y técnicas de ETL extraemos los datos de las distintas fuentes en los que pudieran estar alojados (ERP, CRM, bases de datos tradicionales, planillas Excel, etc) para posteriormente depuramos, consolidamos y cargarlos en un almacén de datos.



3.2.1.2 Estrategia de Manejo de Errores

En algún momento tendremos que utilizar las opciones de depuración y gestión de errores que nos proporciona SSIS. Normalmente todo es mucho más sencillo cuando no hay problemas, pero en caso de errores conocer el funcionamiento de estas herramientas nos hará la vida un poco más fácil.

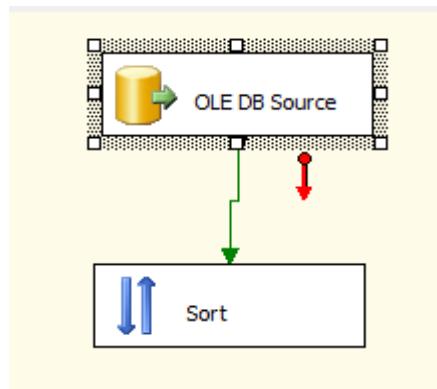
Podemos agrupar estas herramientas como:

- Gestión del control de flujo en los DataFlows.
- Visualización de “los datos” en un punto determinado de nuestro DataFlow.
- Uso de “Event Handlers” de SSIS.
- Seguimiento y adicción de logs o trazas de las ejecuciones de un determinado SSIS.

Veremos el primer caso.

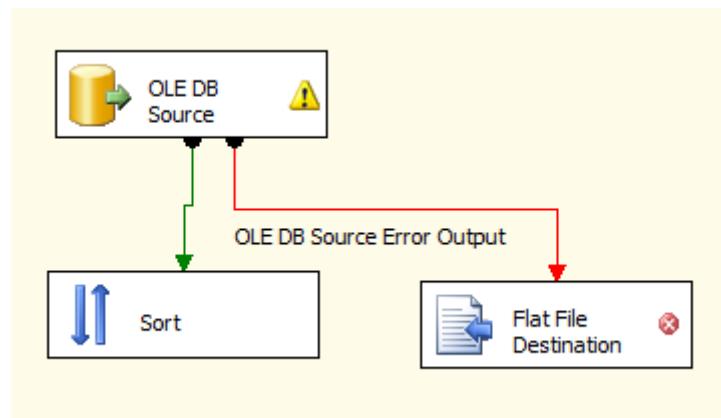
Gestión del control de flujo en los DataFlows

Cualquier componente de SSIS que colocamos en nuestros SSIS tiene al menos dos "caminos" posibles durante su ejecución: correcta e incorrecta. SSIS representa esto mediante dos "flechas" que nos servirán de conectores para el siguiente paso del SSIS.

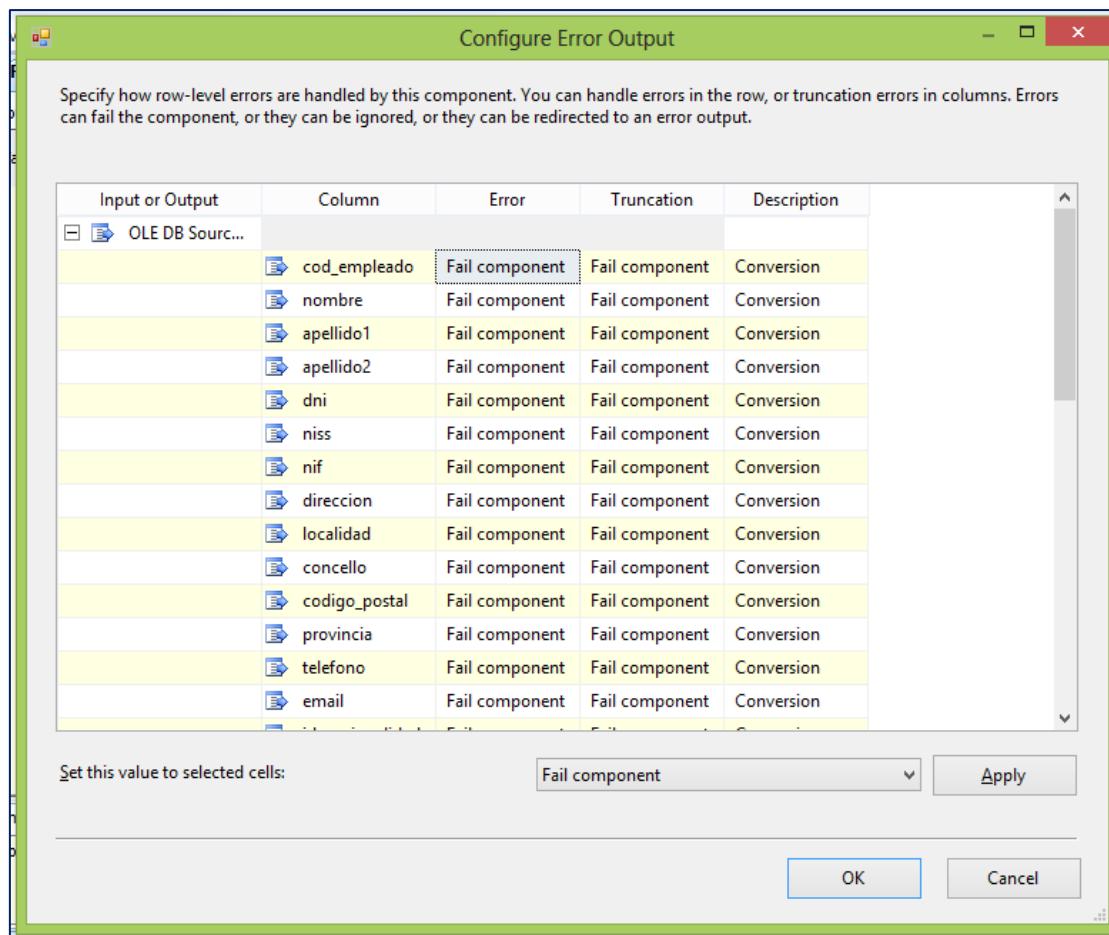


Por ejemplo en este caso mientras que la lectura de los datos en “Ole DB Source” sea correcta, se ejecutará la transformación Sort. En caso contrario el SSIS fallará y no realizará nada.

Esto se puede cambiar si por ejemplo, añadimos un componente “Flat File Destination” y arrastramos la “flecha roja” sobre este nuevo elemento.



Al conectar estos dos elementos nos aparece una pantalla de propiedades donde se podrá configurar qué queremos que haga SSIS en caso de producirse un error en este punto.

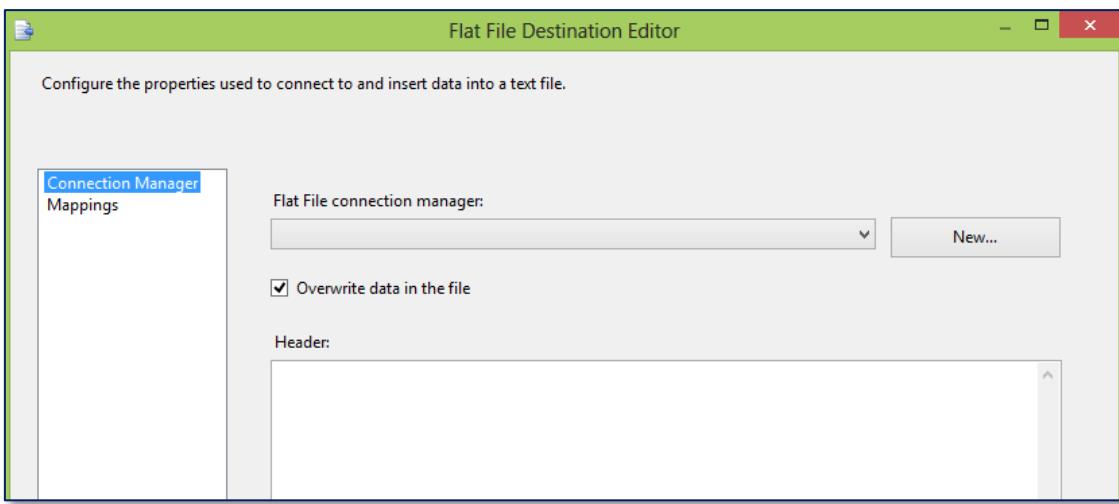


SSIS a nivel de cada campo contempla las posibilidades de “Error” y de “Truncation” y en este cuadro de diálogo le diremos qué debe hacer:

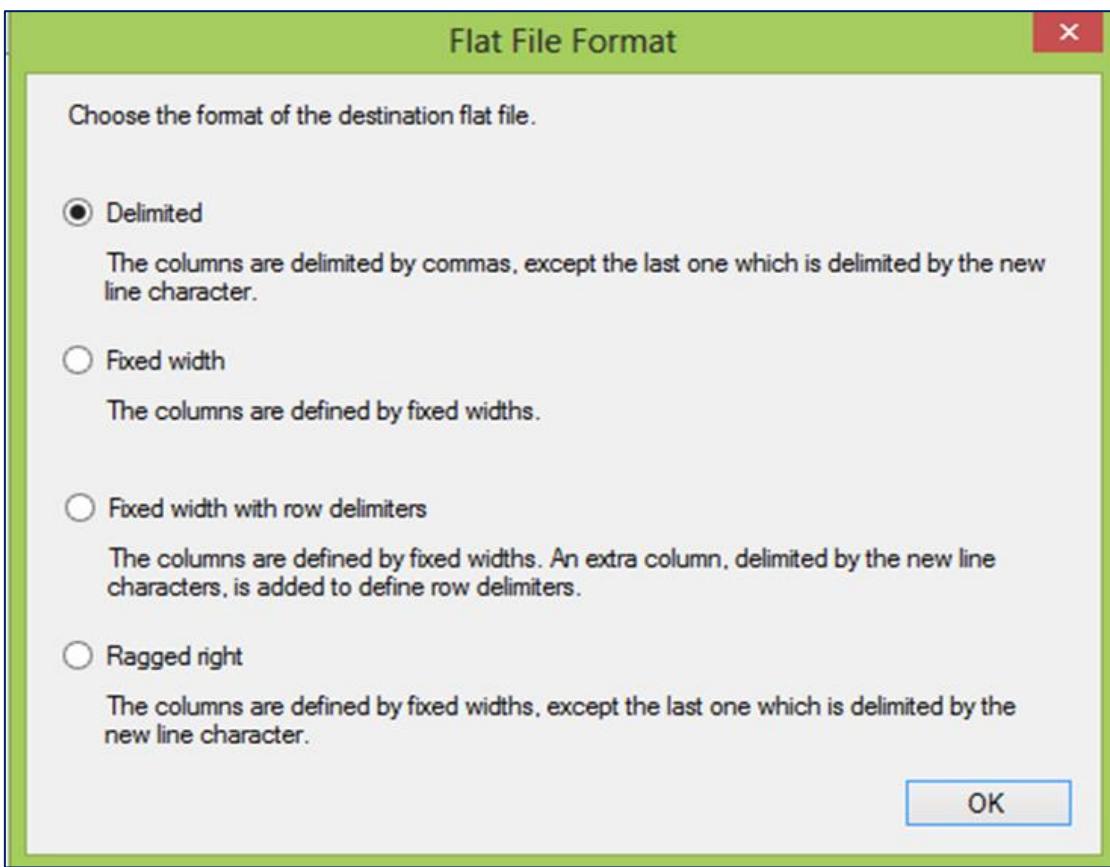
- Fail component: es decir, que pare la ejecución.
- Ignore failure: que omita el fallo y prosiga con la ejecución del SSIS:
- Redirect Row: que redirija la fila al destino que le indicamos (en este caso, un fichero de texto).

Para nuestro caso seleccionaremos en todos los campos la opción “Redirect row” para que en caso una fila tenga en un campo un valor erróneo o que se produzca un truncamiento de datos, nos guarde ese fila o registro en un fichero de texto.

Ahora sólo nos falta especificar el formato del fichero de logs. Para hacemos doble clic sobre el componente “Flat File Destination”.

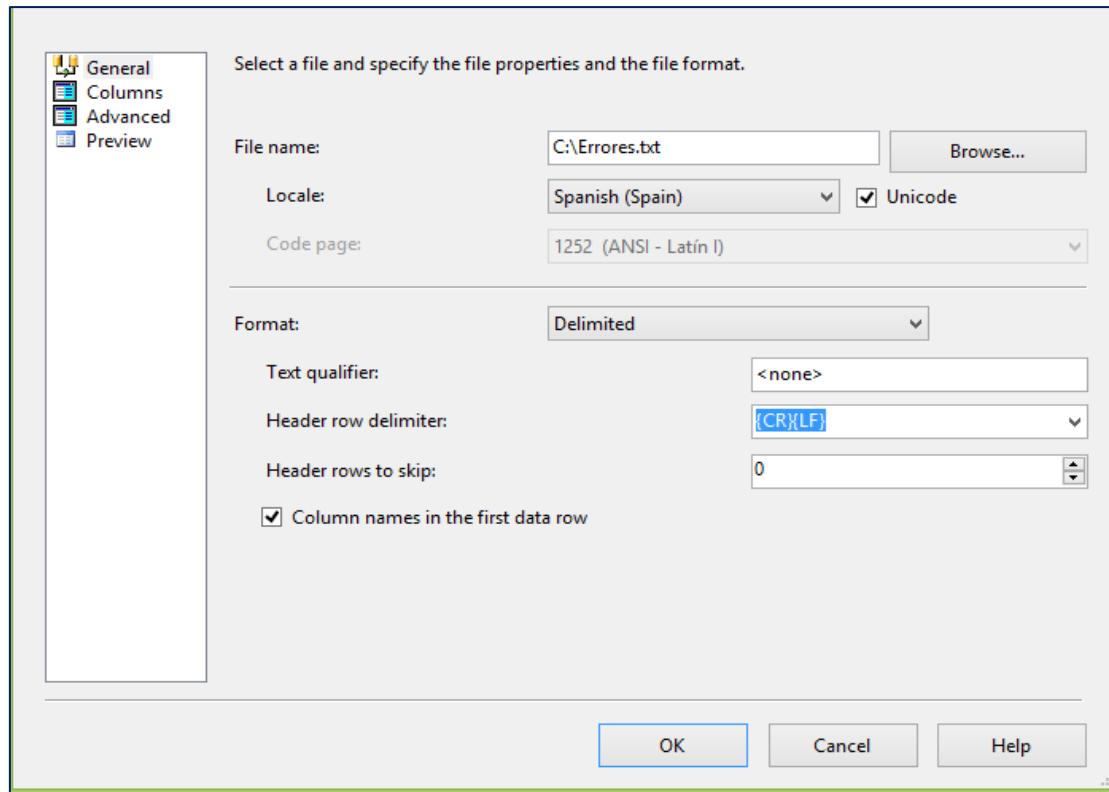


Haremos clic en “New”

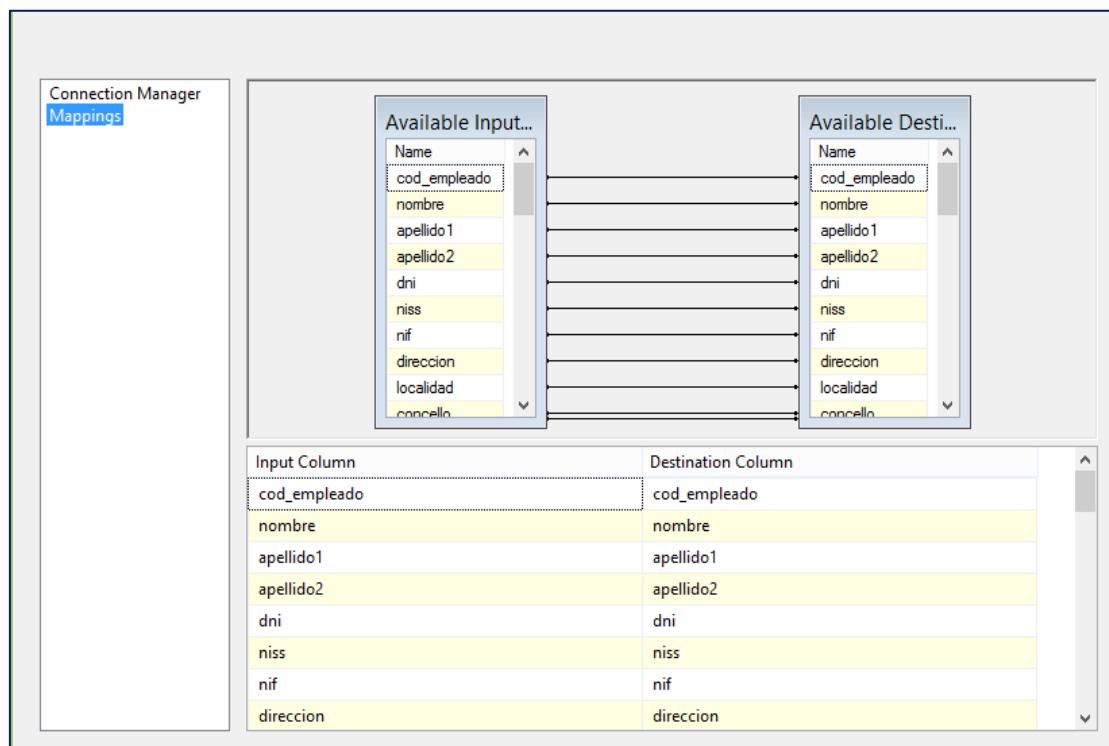


Indicaremos el formato del fichero de texto (en nuestro caso “Delimited”).

A continuación seleccionaremos la ruta de destino, la codificación, así como los campos que queremos que tenga el fichero. Para ello nos moveremos por cada una de las opciones disponibles.



Una vez finalizado, aceptaremos y revisaremos en “Mappings” que la asignación de los campos sea correcta en el momento que se guarde una determinada fila en el fichero. Además en caso de que no queramos guardar un determinado campo en el fichero podemos ignorarlo.



Con esto ya hemos finalizado.

3.2.1.3 Estandarización y limpieza de datos

En este tema se describe cómo limpiar los datos mediante un proyecto de calidad de datos de Data Quality Services (DQS). La limpieza de datos se realiza en los datos de origen utilizando una base de conocimiento generada en DQS a partir de un conjunto de datos de alta calidad. Para obtener más información, vea Crear una base de conocimiento.

La limpieza de datos se realiza en cuatro fases: una fase de asignación en la que se identifica el origen de datos que se va a limpiar y se asigna a los dominios requeridos de una base de conocimiento, una fase de limpieza asistida por PC en la que DQS aplica la base de conocimiento a los datos que se van a limpiar y propone o realiza cambios en los datos de origen, una fase de limpieza interactiva en la que los administradores de datos pueden analizar los cambios en los datos, así como aceptarlos o rechazarlos, y, por último, la fase de exportación que le permite exportar los datos limpios. Cada uno de estos procesos se realiza en una página distinta del asistente para la actividad de limpieza, lo que le permite desplazarse de una página a otra, volver a ejecutar el proceso, y cerrar un proceso de limpieza específico y volver a la misma fase del proceso. DQS proporciona estadísticas sobre los datos de origen y los resultados de limpieza que permiten tomar decisiones fundadas sobre la limpieza de datos.

Requisitos previos

Es necesario especificar valores de umbral apropiados para la actividad de limpieza. Para obtener más información acerca de cómo hacerlo, vea Configurar los valores de umbral para la limpieza y coincidencia.

Debe estar disponible una base de conocimiento de DQS en Servidor de calidad de datos con la que comparar y limpiar los datos de origen. Además, la base de conocimiento debe contener conocimiento sobre el tipo de datos que desea limpiar. Por ejemplo, si desea limpiar datos de origen que contienen direcciones de EE. UU., debe tener una base de conocimiento creada a partir de datos de ejemplo de "alta calidad" para las direcciones de EE. UU.

Si los datos de origen implicados en el proceso de limpieza están en un archivo de Excel, es necesario tener instalado Microsoft Excel en el equipo de Cliente de calidad de datos. De lo contrario, no podrá seleccionar dicho archivo en la fase de asignación. Los archivos creados por Microsoft Excel pueden tener la extensión .xlsx, .xls o .csv. Si se utiliza la versión de 64 bits de Excel, solo se admitirán los archivos de Excel 2003 (.xls); los archivos de Excel 2007 o 2010 (.xlsx) no son compatibles. Si utiliza la versión de 64 bits de Excel 2007 o 2010, guarde el archivo como un archivo .xls o .csv, o instale una versión de 32 bits de Excel en su lugar.

Permisos

Debe disponer del rol dqs_kb_editor o dqs_kb_operator en la base de datos DQS_MAIN para realizar la limpieza de datos.

3.2.1.4 Relacionando Dimensiones y grupo de medidas

Este tema pertenece a la aplicación SSAS.

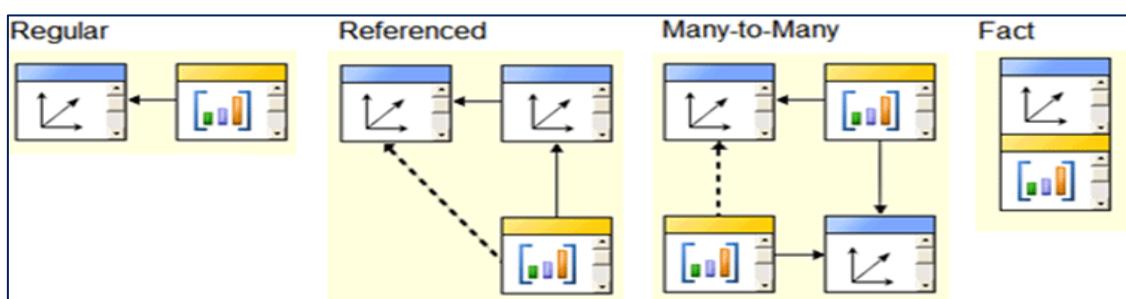
Relaciones entre Dimensiones y Grupos Medidas (Dimension Usage)

A partir de Analysis Services 2005 se produce un gran cambio en el producto. Entre otras muchas mejoras, ya permite que un cubo pueda tener más de una tabla de hechos, y aparece un nuevo concepto que es el de las relaciones entre los grupos de medidas y las dimensiones.

Dimensions	Internet Sales	Reseller Sales	Sales Summary	Sales Targets
Date	Date	Date	Date	Calendar Quarter
Date (Ship Date)	Date	Date	Date	
Date (Delivery Date)	Date	Date	Date	
Customer	Customer			
Reseller		Reseller		
Geography		Reseller		
Employee		Employee		Employee
Promotion	Promotion	Promotion	Promotion	
Product	Product	Product	Product	
Sales Territory	Sales Territory Region	Sales Territory Region	Sales Territory Region	Employee
Sales Reason	Sales Reasons			
Internet Sales Order Details	Internet Sales Order			
Reseller Sales Order Details		Reseller Sales Order		

Como hemos comentado anteriormente, no vamos a entrar a detalle, pero sí que vamos a explicar los tipos de relaciones que tenemos disponibles:

- Normal (Regular). Relaciona directamente una tabla de hechos con una tabla de dimensiones, a través de una o más columnas. Es el caso más habitual.
- Refenciada (Referenced). Una dimensión se relaciona con un grupo de medidas a través de otra dimensión. Se suele utilizar cuando modelamos en copo de nieve (snowflake).
- Varios a Varios (Many-to-Many). La tabla de dimensiones se combina con una tabla de hechos intermedia, y ésta a su vez se combina con una tabla de dimensiones intermedia con la que la tabla de hechos está combinada.
- Hecho (Fact). Se basan en la propia tabla de hechos, y el atributo clave de la dimensión será una columna de la tabla de hechos. Por ejemplo, si en la tabla de hechos tenemos el número de factura y queremos una dimensión Factura.

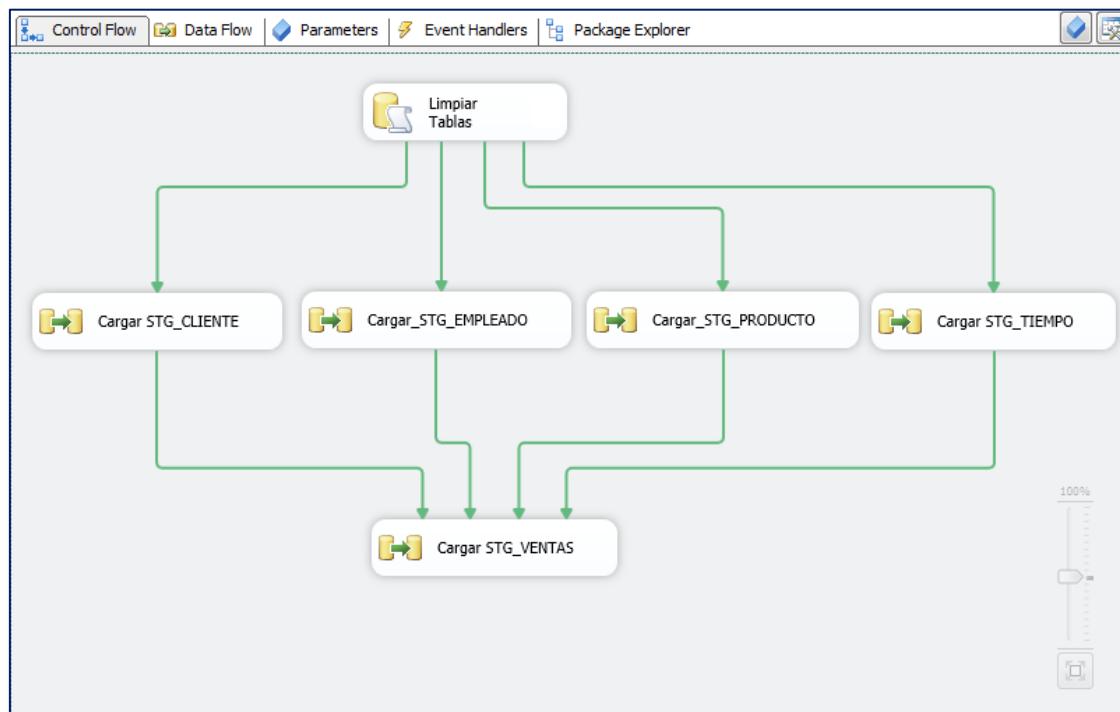


3.2.1.5 Creación y desarrollo de proyecto SSIS

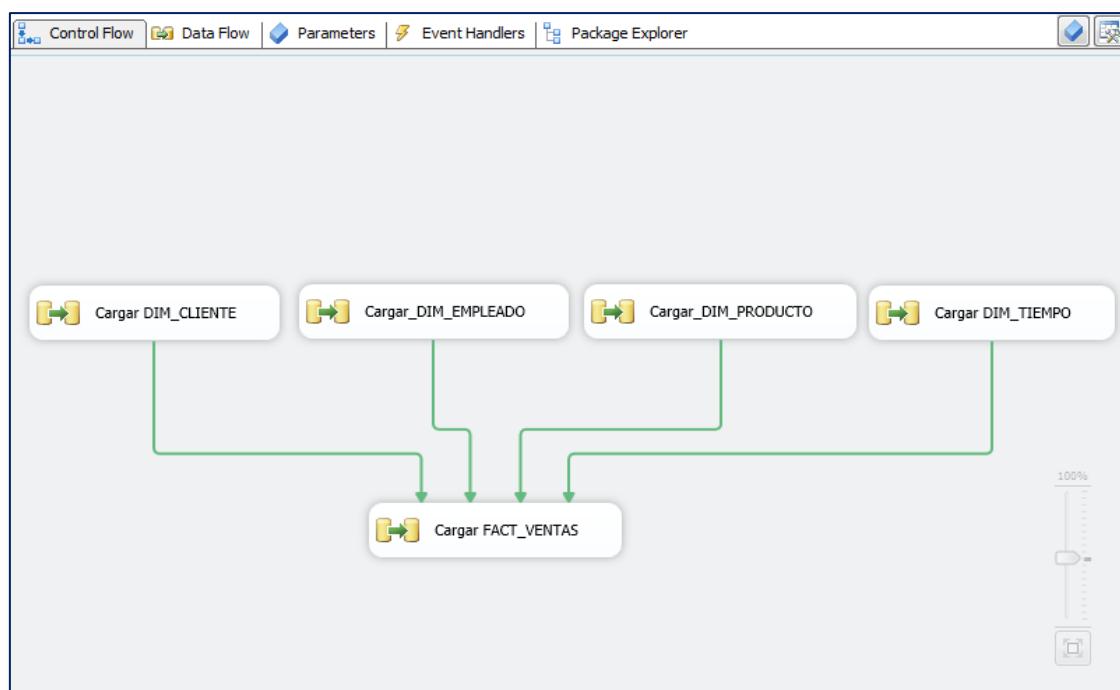
Creación de un proceso ETL desde el origen de datos hasta el DataWarehouse

- Utilice una tarea que permita mover los archivos de texto utilizados en el presente o directorio.
- Desarrolle un ejemplo usando el componente DataProfiling utilizando la base de datos AdventureWorks2012, vista Sales.vSalesPerson.
- Restaure las bases de datos Northwind, Northwind_Stage, Northwind_Mart y cree toda la solución de carga al Datawarehouse.

Carga al STAGE:



Carga DATAMART:





GENERANDO REPORTES Y CONSULTAS MDX

LOGRO DE LA UNIDAD DE APRENDIZAJE

Al término de la unidad, el alumno genera consultas MDX y reportes a partir de fuentes de datos transaccionales y analíticas utilizando SQL Server Reporting Services.

TEMARIO

4.1 Tema 9 : MDX

- 4.1.1 : Introducción a MDX
- 4.1.2 : Consulta de cubos a través de MDX
- 4.1.3 : Concepto de Tuplas y Sets.
- 4.1.4 : Funciones de MDX
- 4.1.5 : Filtra y ordena resultados

4.2 Tema 10 : Creación de Reportes utilizando SSRS

- 4.2.1 : Introducción a SSRS
- 4.2.2 : Creación de proyectos SSRS
- 4.2.2.1 : Establecer Fuente de Datos
- 4.2.2.2 : Establecer Conjunto de datos y creación de consultas
- 4.2.2.3 : Diseño de reportes
- 4.2.2.4 : Uso de Tablix y Matrix

4.3 Tema 11 : Optimización y publicación de reportes

- 4.3.1 : Optimizar reportes
- 4.3.1.1 : Uso de parámetros
- 4.3.1.2 : Uso de gráfico y medidores
- 4.3.1.3 : Creación de subinformes
- 4.3.1.4 : Navegación entre reportes
- 4.3.2 : Publicación de reportes

ACTIVIDADES PROPUESTAS

12. Los alumnos identifican los conceptos de MDX.
13. Los alumnos identifican y desarrollan reportes utilizando SSRS

4.1 MDX

4.1.1 Introducción a MDX

MDX es el acrónimo de MULTIDIMENSIONAL EXPRESSIONS. Constituye un lenguaje de consulta que soporta la definición y manipulación de datos almacenados en estructuras multidimensionales. No es una extensión del SQL (Structured Query Language).

MDX emplea las cláusulas SELECT, FROM y WHERE para extraer información de un cubo. También provee un conjunto de funciones para manipular y recuperar datos.

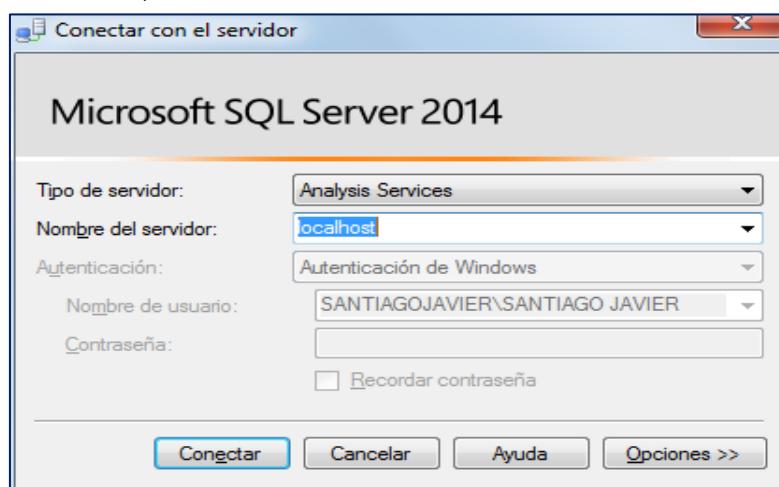
Funciones MDX

Analysis Services provee alrededor de 100 funciones MDX para definir manipular la información almacenada en un cubo. Estas funciones pueden utilizarse para construir miembros calculados (calculated members) en los cubos. Las funciones MDX están categorizadas de la siguiente manera:

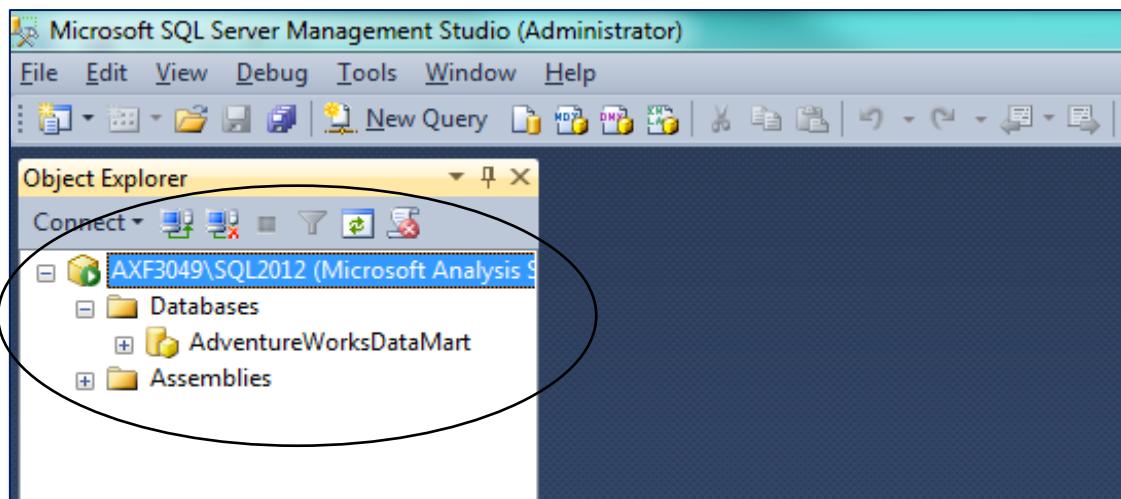
- Funciones de Array
- Funciones de Dimensión
- Funciones de Jerarquías
- Funciones de Levels
- Funciones Lógicas
- Funciones Member
- Funciones Numéricas
- Funciones de Set
- Funciones de cadenas
- Funciones de Tuplas

4.1.2 Consultas de cubos a través de MDX

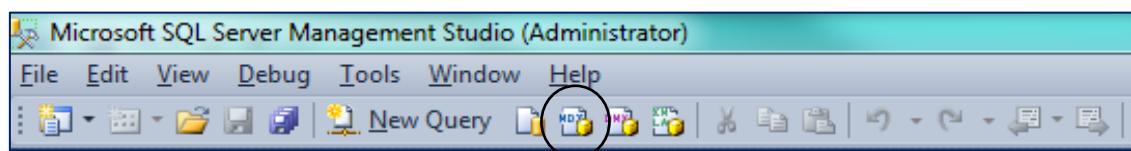
Pueden efectuarse consultas MDX dirigidas a los cubos de SSAS 2014 a través del SQL Server Management Studio. Para iniciarla, en el menú Start del sistema operativo, seleccione All Programs → Microsoft SQL Server 2014 → SQL Server Management Studio. Aparecerá una ventana en la cual se debe ingresar la información de autenticación. En la lista Server type, seleccione “Analysis Services”. En la lista Server name, escriba “localhost”.



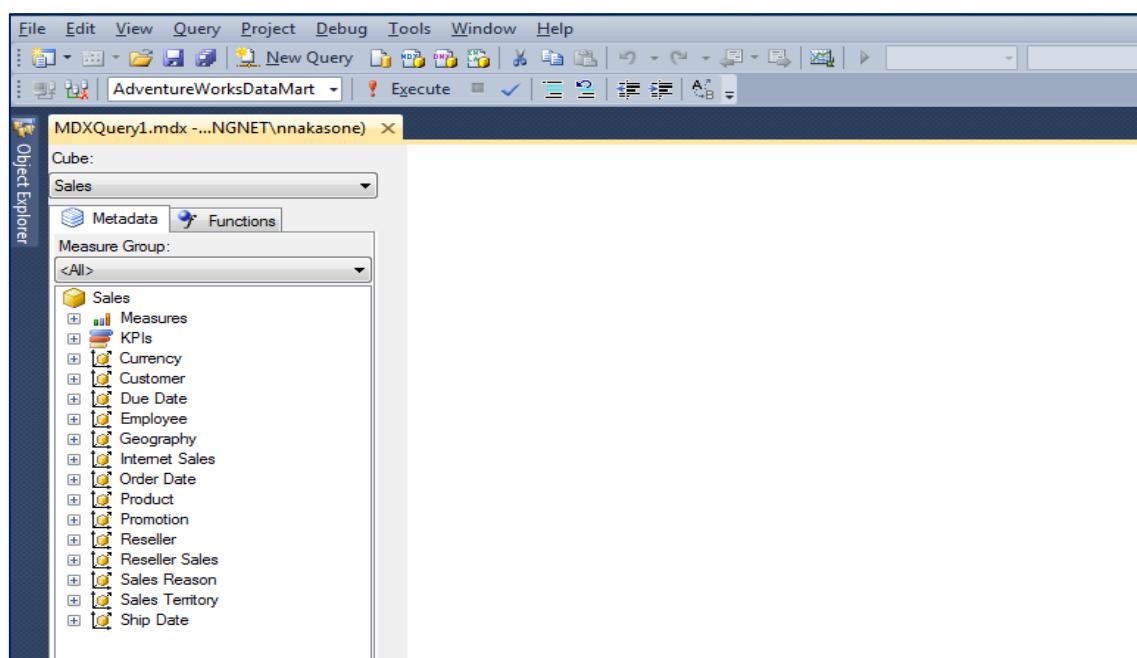
Pulse el botón Connect. El SQL Server Management Studio mostrará la lista de bases de datos de análisis en el Object Explorer (si no puede visualizar esta ventana, seleccione el menú View y la opción Object Explorer).



Para escribir una nueva consulta MDX, pulse el botón situado en el toolbox del SQL Server Management Studio:



Cada vez que se cree una nueva consulta MDX, se abrirá nuevamente la ventana de autenticación. Se deben aceptar los valores por defecto y pulsar el botón Connect. En la lista de la parte superior del toolbox, seleccione la base de datos "AdventureWorksDataMart":



La siguiente consulta MDX muestra la estructura básica de la sentencia SELECT:

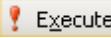
```
SELECT especificación_eje ON COLUMNS, especificación_eje ON ROWS
FROM Nombre_Cubo
WHERE especificación_de_filtro
```

Donde especificación_eje es una expresión de set que representa los miembros que se utilizarán en las consultas. Esta consulta utiliza dos ejes representados por las expresiones ON COLUMNS y ON ROWS. Las consultas MDX admiten hasta 128 ejes. Los cinco primeros ejes pueden ser expresados a través de ON COLUMNS, ON ROWS, ON PAGES, ON CHAPTERS y ON SECTIONS. Los ejes restantes se expresan a través de ON AXIS (índice_del_eje). La especificación del filtro WHERE es opcional.

Escriba la siguiente sentencia MDX en la ventana de consulta:

```
SELECT Measures.Members ON COLUMNS,
[Promotion].[Promotion Categories].Members ON ROWS
FROM Sales
```

Obsérvese el uso de la expresión MEMBERS. Esta expresión devuelve un set que contiene a todos los miembros de una dimensión. La consulta devuelve a todos los miembros de la dimensión Measures (que contiene a todas las medidas del cubo), y todos los miembros del atributo Promotion Categories de la dimensión Promotion.

Presione el botón  o presione la tecla F5. El resultado aparecerá en el panel de resultados:

The screenshot shows the Microsoft SQL Server Management Studio interface. In the center, there is a query editor window titled "MDXQuery1.mdx...ministrador)*" containing the following MDX code:

```
SELECT Measures.Members ON COLUMNS,
{[Promotion].[Promotion Category].[Customer],
[Promotion].[Promotion Category].[Reseller]} ON ROWS
FROM Sales
```

To the left of the query editor is a cube browser pane for the "Sales" cube. The "Measure Group" dropdown is set to "<All>". The "Promotion" measure group is expanded, showing categories like "Discount Pct", "Promotion Category", "Promotion Name", "Promotion Type", and "Promotion Categories". Below these are "Reseller", "Reseller Sales", "Sales Reason", and "Sales Territory".

The bottom of the screen shows the status bar with "Ready", "Ln 4", "Col 1", "Ch 1", and "INS".

El uso de corchetes es opcional, a menos que los identificadores de dimensión o de miembros contengan espacios en blanco.

La siguiente consulta contiene una expresión de set en el eje ROWS. Esto permite seleccionar algunos miembros específicos para la consulta. Obsérvese el uso de las llaves ({}):

```
SELECT Measures.Members ON COLUMNS,
{[Promotion].[Promotion Category].[Customer],
[Promotion].[Promotion Category].[Reseller]} ON ROWS
FROM Sales
```

Los resultados serán los siguientes:

	Order Quantity	Discount Amount	Sales Amount	Tax Amt	Freight	Reseller Sales Count	Internet Order Quantity
Customer	(null)	(null)	(null)	(null)	(null)	(null)	(null)
Reseller	21179	344861.5852	3596577.8164	287726.2219	89914.4513	2035	1121

La siguiente consulta muestra el uso de la función CHILDREN. Esta función, aplicada sobre un miembro específico, devuelve un set que contiene a todos sus miembros hijos.

En este ejemplo, se consultará el valor de todas las medidas del cubo, para todos los miembros hijos de las categorías de promoción Customer y Reseller, según la jerarquía Promotion Categories:

```
SELECT Measures.Members ON COLUMNS,
{[Promotion].[Promotion Categories].[Promotion
Category].[Customer].children,
[Promotion].[Promotion Categories].[Promotion Category].[Reseller].children} ON ROWS
FROM Sales
```

El resultado es el siguiente:

The screenshot shows a Windows application window titled 'Results' from Microsoft Analysis Services. It has tabs for 'Messages' and 'Results'. The table contains data with the following columns: Order Quantity, Discount Amount, Sales Amount, Tax Amt, Freight, Reseller Sales Count, and Internet. The rows represent different promotion categories. The data is as follows:

	Order Quantity	Discount Amount	Sales Amount	Tax Amt	Freight	Reseller Sales Count	Internet
Excess Inventory	(null)	(null)	(null)	(null)	(null)	(null)	
Seasonal Discount	(null)	(null)	(null)	(null)	(null)	(null)	
Discontinued Product	629	118557.5342	215819.7823	17265.5811	5395.4952	226	
Excess Inventory	216	15221.3256	35516.4264	2841.314	887.9112	47	
New Product	1233	119577.5143	546958.8674	43756.707	13673.9689	393	
Seasonal Discount	721	1474.526	10239.72	819.1793	255.9945	133	
Volume Discount	18380	90030.6851000001	2788043.0203	223043.4405	69701.0815	1236	

Un miembro puede ser identificado a través de su nombre completo: dimensión.jerarquía.nivel.miembro; o de forma resumida mediante su nombre corto: dimensión.jerarquía.miembro. La consulta anterior puede modificarse de la siguiente manera:

```
SELECT Measures.Members ON COLUMNS,
{[Promotion].[Promotion Categories].[Customer].children,
[Promotion].[Promotion Categories].[Reseller].children} ON ROWS
FROM Sales
```

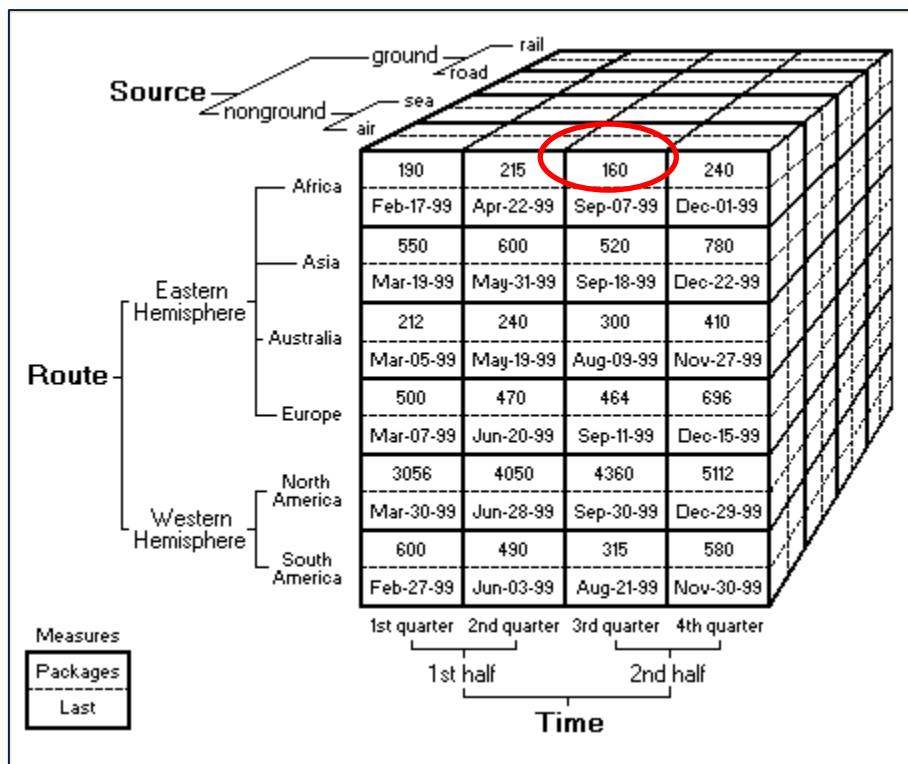
El nombre completo de los miembros debe ser utilizado en los casos en que existan varios miembros con el mismo nombre en una consulta. Por ejemplo, en una dimensión geográfica puede haber dos ciudades llamadas Birmingham, una en Inglaterra y otra en USA. Utilizar el nombre completo evita la ambigüedad.

4.1.3 Concepto de tuplas y sets

La figura muestra una representación gráfica de un cubo.

Una celda (cell) es una intersección específica de los miembros de las dimensiones del cubo. La información de las medidas se encuentra almacenada en las celdas. MDX permite la extracción de información desde una sola celda o desde un conjunto de ellas a través de un sistema de referencias denominado T – uplas.

Las T – uplas identifican celdas individuales y conjuntos de celdas a través de la especificación de las dimensiones y los miembros cuya intersección da origen a la celda (o celdas).



Las medidas de un cubo son tratadas como una dimensión privada denominada Measures. Por ejemplo, en la figura anterior, la celda encerrada en la elipse puede identificarse a través de la siguiente Tupla:

(Route.[Eastern Hemisphere].Africa, Time.[2nd half].[4th quarter],
Source.nonground.Air, Measures.Packages)

La Tupla mostrada identifica únicamente a una celda del cubo. La siguiente Tupla identifica a todas las celdas que corresponden al miembro Eastern Hemisphere de la dimensión Route:

(Route.[Eastern Hemisphere])

La siguiente T – upla identifica a todas las celdas que corresponden al miembro Eastern Hemisphere de la dimensión Route, y el miembro 2nd half de la dimensión Time:

(Route.[Eastern Hemisphere], Time.[2nd half])

Estas secciones parciales del cubo se denominan slicers. Cada slicer contiene una o más celdas.

Un set es una colección de T – uplas. Los sets se escriben utilizando llaves al inicio y al final de la expresión de set. Por ejemplo:

{ (Time.[1st half].[1st quarter]), Time.[2nd half].[3rd quarter] }

4.1.4 Funciones en MDX

La función DESCENDANTS devuelve un set que contiene todos los hijos de un miembro determinado, en un nivel determinado. Su sintaxis es la siguiente:

```
DESCENDANTS(member, level [, flags])
```

Donde “member” representa al miembro cuyos descendientes se desea obtener, “level” es el nivel donde se ubican los descendientes, y “flags” puede tomar uno de los siguientes valores:

- SELF: Valor por defecto. Retorna a los descendientes de “member” que se encuentran dentro del nivel especificado por “level”.
- AFTER: Retorna a todos los descendientes en los niveles que se encuentran debajo de “level”.
- BEFORE: Retorna a todos los descendientes que se encuentran entre “member” y “level”, sin incluir a los miembros de “level”.
- BEFORE_AND_AFTER: Retorna a todos los descendientes de “member”, excepto a los miembros del nivel “level”.
- SELF_AND_AFTER: Retorna a todos los descendientes de “member” en el nivel “level” y todos sus niveles inferiores.
- SELF_AND_BEFORE: Retorna a todos los descendientes de “member” en el nivel “level” y todos los niveles entre “member” y “level”.
- SELF_BEFORE_AFTER: Retorna a todos los descendientes de “member” en todos los niveles que se encuentran debajo del nivel de “member”.
- LEAVES: Retorna todos los miembros leaf (sin descendientes) que descienden de “member” y se encuentran entre el nivel de “level” y el nivel que contiene a “member”

La siguiente consulta obtiene los valores de todas las medidas del cubo Sales, para el miembro United States de la dimensión Geography, y para todos sus descendientes en el nivel City según la jerarquía Geographic location:

```
SELECT Measures.Members ON COLUMNS,
 $\{[Geography].[Geographic location].[Country Region Name].[United States],$ 
 $\text{DESCENDANTS}([Geography].[Geographic location].[Country Region}$ 
 $\text{Name].[United States],$ 
 $[Geography].[Geographic location].[City])\} \text{ ON ROWS}$ 
 $\text{FROM Sales}$ 
```

Los resultados son los siguientes:

The screenshot shows a table with the following columns: Order Quantity, Discount Amount, Sales Amount, Tax Amt, Freight, and Reseller Sales Count. The data includes rows for cities like Birmingham, Florence, Huntsville, Mobile, Montgomery, Chandler, Gilbert, Mesa, Phoenix, Scottsdale, and Surprise.

	Order Quantity	Discount Amount	Sales Amount	Tax Amt	Freight	Reseller Sales Count
United States	132748	311132.509	53607801.2101999	4288623.9891	1340195.6562	38809
Birmingham	53	0	20510.735	1640.8584	512.7687	23
Florence	5	0	1772.4316	141.7945	44.3108	4
Huntsville	29	0	23145.8624	1851.669	578.647	14
Mobile	(null)	(null)	(null)	(null)	(null)	(null)
Montgomery	(null)	(null)	(null)	(null)	(null)	(null)
Chandler	509	10388.0152	356081.5688	28486.5255	8902.0412	112
Gilbert	45	391.1602	28016.698	2241.336	700.418	23
Mesa	288	5.7289	114294.4289	9143.5536	2857.3617	91
Phoenix	2616	1054.1517	685695.2091	54855.6152	17142.3916	670
Scottsdale	269	190.7256	140088.7526	11207.0991	3502.2215	118
Surprise	94	0	11739.8794	939.1903	293.4975	34

La siguiente sentencia retorna a United States y todos sus descendientes hasta el nivel anterior a City:

```
SELECT Measures.Members ON COLUMNS,
DESCENDANTS([Geography].[Geographic location].[Country Region Name].[United States],
[Geography].[Geographic location].[City], before_and_after) ON ROWS
FROM Sales
```

Los resultados son los siguientes:

The screenshot shows a table with the same columns as the previous one, but the data is now grouped by state. The states listed are Alabama, Arizona, California, Colorado, Connecticut, Florida, Georgia, Idaho, Illinois, Indiana, and Kentucky.

	Order Quantity	Discount Amount	Sales Amount	Tax Amt	Freight	Reseller Sales Count
United States	132748	311132.509	53607801.2101999	4288623.9891	1340195.6562	38809
Alabama	87	0	45429.029	3634.3219	1135.7265	41
Arizona	4010	13956.4555	1432585.5698	114606.8425	35814.6597	1151
California	24302	94150.9917000001	9764270.70830001	781141.644599992	244106.881799998	6920
Colorado	6703	8142.0051	2395913.84299999	191673.105599999	59897.8741	1673
Connecticut	2421	2197.0571	1125700.407	90056.0298	28142.5251000001	913
Florida	6117	12594.7603	2299888.86799999	183991.1029	57497.2501	1851
Georgia	2601	3035.6486	1044577.7962	83566.22	26114.4596	862
Idaho	465	2647.6124	227301.751	18184.14	5682.54689999999	170
Illinois	1511	412.4569	607939.968799999	48635.1968	15198.5075	470
Indiana	2886	965.595	1073388.5872	85871.083299999	26834.7254	905
Kentucky	462	1655.1488	163713.8398	13097.1066	4092.8501	211

La cláusula WHERE

La cláusula WHERE se utiliza para limitar el conjunto de celdas que se totalizan para dar origen a la información de la consulta. La cláusula WHERE define un slicer conformado por dimensiones y miembros específicos, que restringen el universo de la consulta.

Por ejemplo, la siguiente sentencia muestra la medida Internet Order Quantity, para todos los años de la jerarquía Calendar en la dimensión Order Date, y para todos los miembros en el nivel Country Region Name de la jerarquía Geographic location:

```
SELECT [Order Date].[Calendar].[Year].Members ON COLUMNS,
[Geography].[Geographic location].[Country Region Name].Members ON ROWS
FROM Sales
```

where ([Measures].[Internet Order Quantity])

El resultado es el siguiente:

	2001	2002	2003	2004
Australia	(null)	(null)	(null)	357
Canada	(null)	53	(null)	458
France	(null)	233	2283	2944
Germany	(null)	233	2254	3062
United Kingdom	(null)	(null)	2	203
United States	341	861	8498	11618
Unknown	(null)	(null)	(null)	(null)

Cada celda muestra el valor de Internet Order Quantity para todas las categorías de productos. Si se desea restringir más la consulta para que sólo considere la categoría "Bikes", se puede utilizar la siguiente expresión slicer:

**([Measures].[Internet Order Quantity],
[Product].[Product Categories].[Product Category Name].[Bikes])**

```
SELECT [Order Date].[Calendar].[Year].Members ON COLUMNS,
[Geography].[Geographic location].[Country Region Name].Members ON
ROWS
FROM Sales
where ([Measures].[Internet Order Quantity],
[Product].[Product Categories].[Product Category Name].[Bikes])
```

Los resultados serán los siguientes:

	2001	2002	2003	2004
Australia	(null)	(null)	(null)	(null)
Canada	(null)	53	(null)	(null)
France	(null)	233	592	560
Germany	(null)	233	620	671
United Kingdom	(null)	(null)	(null)	(null)
United States	341	861	1738	1967
Unknown	(null)	(null)	(null)	(null)

Observe que los resultados son distintos en las dos consultas.

Miembros calculados (calculated members) y sets con nombre (named sets)

Los miembros calculados (calculated members) permiten la definición de fórmulas personalizadas, las cuales pueden ser tratadas como nuevos miembros en una dimensión específica. La sintaxis de creación de miembros calculados es la siguiente:

WITH MEMBER dimensión.nombre AS 'expresión'

Donde “dimensión” representa la dimensión a la cual pertenecerá el nuevo miembro calculado. El siguiente ejemplo muestra la sintaxis de definición de un miembro calculado llamado ProfitPercent, ubicado dentro de la dimensión Measures:

```
WITH MEMBER Measures.PorcentajeVentasInternet AS
'Measures.[Internet Sales Amount] /

$$([Measures].[Sales Amount] + [Measures].[Internet Sales Amount])'$$
,
FORMAT_STRING='PERCENT'
```

La propiedad FORMAT_STRING determina el formato que se empleará para la presentación de la información. Por ejemplo, el empleo del símbolo % indica que el valor devuelto será un porcentaje. El siguiente ejemplo muestra la creación de un miembro calculado en Measures llamado TotalVentas. Con base en este miembro calculado, se crea otro miembro llamado PorcentajeVentasInternet:

```
WITH MEMBER Measures.TotalVentas AS
'[Measures].[Sales Amount] + [Measures].[Internet Sales Amount]'  

MEMBER Measures.PorcentajeVentasInternet AS
'Measures.[Internet Sales Amount] / Measures.TotalVentas',  

FORMAT_STRING='PERCENT'  

select {[Order Date].[Calendar].Year.MEMBERS} ON COLUMNS,  

[Geography].[Geographic location].[Country Region Name].MEMBERS ON  

ROWS  

FROM SALES  

WHERE (Measures.PorcentajeVentasInternet
```

El resultado es el siguiente:

	2001	2002	2003	2004
Australia	(null)	(null)	(null)	100.00%
Canada	(null)	100.00%	(null)	100.00%
France	(null)	39.86%	32.72%	42.83%
Germany	(null)	100.00%	100.00%	100.00%
United Kingdom	(null)	(null)	100.00%	100.00%
United States	14.38%	10.77%	12.39%	26.20%
Unknown	(null)	(null)	(null)	(null)

También se pueden usar expresiones de cálculo sobre miembros de una dimensión Time. Por ejemplo, la siguiente expresión define dos miembros calculados llamado Primer Bienio y Segundo Bienio en la jerarquía Calendar de la dimensión Order Date:

```
MEMBER [Order Date].[Calendar].[Primer Bienio] AS
'[Order Date].[Calendar].[2001] + [Order Date].[Calendar].[2002]'  

MEMBER [Order Date].[Calendar].[Segundo Bienio] AS
'[Order Date].[Calendar].[2003] + [Order Date].[Calendar].[2004]'
```

El siguiente ejemplo muestra estos miembros calculados:

```
WITH MEMBER Measures.TotalVentas AS
'[Measures].[Sales Amount] + [Measures].[Internet Sales Amount]'  

MEMBER Measures.PorcentajeVentasInternet AS
'Measures.[Internet Sales Amount] / Measures.TotalVentas',  

FORMAT_STRING='PERCENT'  

MEMBER [Order Date].[Calendar].[Primer Bienio] AS
```

```
[Order Date].[Calendar].[2001] + [Order Date].[Calendar].[2002]
MEMBER [Order Date].[Calendar].[Segundo Bienio] AS
[Order Date].[Calendar].[2003] + [Order Date].[Calendar].[2004]
select {[Order Date].[Calendar].[Primer Bienio],
[Order Date].[Calendar].[Segundo Bienio]} ON COLUMNS,
[Geography].[Geographic location].[Country Region Name].MEMBERS ON
ROWS
FROM SALES
WHERE (Measures.PorcentajeVentasInternet)
```

Los resultados son: los siguientes

	Primer Bienio	Segundo Bienio
Australia	(null)	100.00%
Canada	100.00%	100.00%
France	39.86%	36.83%
Germany	100.00%	100.00%
United Kingdom	(null)	100.00%
United States	11.78%	17.31%
Unknown	(null)	(null)

La propiedad SOLVE_ORDER determina el orden en el cual se evalúan y calculan los miembros calculados. Se puede dar a SOLVE_ORDER un valor de 0 para el miembro calculado PorcentajeVentasInternet, y un valor de 1 para los miembros Primer Bienio y Segundo Bienio, de manera que PorcentajeVentasInternet se calcule primero:

```
WITH MEMBER Measures.TotalVentas AS
'[Measures].[Sales Amount] + [Measures].[Internet Sales Amount]'
MEMBER Measures.PorcentajeVentasInternet AS
'Measures.[Internet Sales Amount] / Measures.TotalVentas',
FORMAT_STRING='PERCENT', SOLVE_ORDER=0
MEMBER [Order Date].[Calendar].[Primer Bienio] AS
[Order Date].[Calendar].[2001] + [Order Date].[Calendar].[2002],
SOLVE_ORDER=1
MEMBER [Order Date].[Calendar].[Segundo Bienio] AS
[Order Date].[Calendar].[2003] + [Order Date].[Calendar].[2004],
SOLVE_ORDER=1
select {[Order Date].[Calendar].[Primer Bienio], [Order
Date].[Calendar].[Segundo Bienio]}
ON COLUMNS,
[Geography].[Geographic location].[Country Region Name].MEMBERS ON
ROWS
FROM SALES
WHERE (Measures.PorcentajeVentasInternet)
```

Observe la variación en el resultado de la consulta.

	Primer Bienio	Segundo Bienio
Australia	(null)	100.00%
Canada	100.00%	100.00%
France	39.86%	75.54%
Germany	100.00%	200.00%
United Kingdom	(null)	200.00%
United States	25.15%	38.59%
Unknown	(null)	(null)

Si se cambian los valores de SOLVE_ORDER para que se evalúen primero los miembros calculados Primer Bienio y Segundo Bienio, y después el miembro calculado PorcentajeVentasInternet, los valores coincidirán con los resultados de la primera consulta.

```

WITH MEMBER Measures.TotalVentas AS
'[Measures].[Sales Amount] + [Measures].[Internet Sales Amount]'  

MEMBER Measures.PorcentajeVentasInternet AS
'Measures.[Internet Sales Amount] / Measures.TotalVentas',  

FORMAT_STRING='PERCENT', SOLVE_ORDER=1  

MEMBER [Order Date].[Calendar].[Primer Bienio] AS
'[Order Date].[Calendar].[2001] + [Order Date].[Calendar].[2002]',  

SOLVE_ORDER=0  

MEMBER [Order Date].[Calendar].[Segundo Bienio] AS
'[Order Date].[Calendar].[2003] + [Order Date].[Calendar].[2004]',  

SOLVE_ORDER=0  

select {[Order Date].[Calendar].[Primer Bienio], [Order  

Date].[Calendar].[Segundo Bienio]}  

ON COLUMNS,  

[Geography].[Geographic location].[Country Region Name].MEMBERS ON  

ROWS  

FROM SALES  

WHERE (Measures.PorcentajeVentasInternet)

```

	Primer Bienio	Segundo Bienio
Australia	(null)	100.00%
Canada	100.00%	100.00%
France	39.86%	36.83%
Germany	100.00%	100.00%
United Kingdom	(null)	100.00%
United States	11.78%	17.31%
Unknown	(null)	(null)

En los ejemplos anteriores, los miembros calculados se han asignado directamente a una dimensión. Es posible, además, definir un miembro calculado como hijo de otro miembro existente, como por ejemplo

```

WITH MEMBER [Order Date].[Calendar].[2003].[Bimestre 1 - 2003] AS
'[Order Date].[Calendar].[January 2003] + [Order Date].[Calendar].[February

```

```

2003]'  

MEMBER [Order Date].[Calendar].[2003].[Bimestre 2 - 2003] AS  

'[Order Date].[Calendar].[March 2003] + [Order Date].[Calendar].[April 2003]'  

select {[Order Date].[Calendar].[Bimestre 1 - 2003],  

[Order Date].[Calendar].[Bimestre 2 - 2003]} ON COLUMNS,  

[Geography].[Geographic location].[Country Region Name].MEMBERS ON  

ROWS  

FROM SALES  

WHERE (Measures.[Sales Amount])

```

En este ejemplo, los miembros calculados Bimestre 1 – 2003 y Bimestre 2 – 2003 se han designado como hijos del miembro 2003. El resultado es el siguiente:

	Bimestre 1 - 2003	Bimestre 2 - 2003
Australia	(null)	(null)
Canada	(null)	(null)
France	162125.8126	106295.7954
Germany	(null)	(null)
United Kingdom	(null)	(null)
United States	2752353.99649999	2314131.4748
Unknown	(null)	(null)

Un set con nombre (named set) permiten crear sets en tiempo de ejecución y utilizarlos en consultas MDX. La sintaxis para definir sets con nombre es la siguiente:

```
WITH SET nombre_set AS 'expression'
```

Por ejemplo, se puede crear un set llamado Norteamerica que contenga a los miembros Canada y United States de la dimensión Geography como se muestra a continuación:

```

WITH SET [Norteamerica] AS  

'{[Geography].[Geographic location].[Canada], [Geography].[Geographic  

location].[United  

States]}'  

select [Norteamerica] ON COLUMNS,  

[Product].[Product Categories].[Product Category Name].MEMBERS ON ROWS  

FROM SALES  

WHERE (Measures.[Sales Amount])

```

El resultado es el siguiente:

	Canada	United States
Accessories	(null)	303515.227900003
Bikes	(null)	44832751.728801
Clothing	(null)	1037436.94729996
Components	(null)	7434097.30620015

Las expresiones desarrolladas en estos ejemplos definen miembros calculados y sets con nombre que están disponibles sólo en tiempo de consulta. Con cada consulta, vuelven a evaluarse y calcularse. Si se requiere que persistan a lo largo de una sesión MDX, se debe emplear el comando CREATE con la cláusula de ámbito SESSION.

Navegación por jerarquías

Con frecuencia, es necesario hacer referencia a cada miembro de una determinada dimensión, y utilizarlo en una operación aritmética con un miembro distinto en la misma dimensión. Por ejemplo, calcular el cociente entre las ventas de cada mes y las ventas del mes precedente. Este tipo de operaciones pueden implementarse utilizando las funciones PREV MEMBER (miembro anterior), NEXT MEMBER (miembro siguiente), CURRENT MEMBER (miembro actual) y PARENT (miembro padre). También puede utilizarse la función FIRST CHILD que toma el primer elemento hijo del miembro especificado; y la función LAST CHILD que toma el último elemento hijo.

A través de estas funciones se puede escribir, por ejemplo, una expresión de cálculo que divide las ventas de un producto entre las ventas totales de su subcategoría. La siguiente sentencia obtiene el valor de la medida Sales Amount para cada producto, y lo divide entre el valor de Sales Amount para su miembro padre, es decir, la subcategoría a la que pertenece:

```
WITH MEMBER MEASURES.PercentageSales AS
'([Product].[Product Categories].CURRENTMEMBER, [Measures].[Sales
Amount]) /
([Product].[Product Categories].CURRENTMEMBER.PARENT,
[Measures].[Sales
Amount])',
FORMAT_STRING = 'PERCENT'
SELECT {MEASURES.[Sales Amount], MEASURES.PercentageSales} ON
COLUMNS,
[Product].[Product Categories].[Product Name].MEMBERS ON ROWS
FROM [Sales]
```

El resultado es el siguiente:

	Sales Amount	PercentageSales
Hitch Rack - 4-Bike	112147.836	100.00%
All-Purpose Bike Stand	(null)	(null)
Mountain Bottle Cage	(null)	(null)
Road Bottle Cage	(null)	(null)
Water Bottle - 30 oz.	4316.6231	100.00%
Bike Wash - Dissolver	6465.86720000001	100.00%
Fender Set - Mountain	(null)	(null)
Sport-100 Helmet, Black	8942.61949999999	5.47%
Sport-100 Helmet, Black	21749.0490999999	13.30%
Sport-100 Helmet, Black	23388.4431	14.30%
Sport-100 Helmet, Blue	10384.3526	6.35%
Sport-100 Helmet, Blue	22770.1004999999	13.92%
Sport-100 Helmet, Blue	25470.2742	15.58%
Sport-100 Helmet, Red	8599.44899999999	5.26%
Sport-100 Helmet, Red	20840.9605	12.74%
Sport-100 Helmet, Red	21384.5482	13.08%
Hydration Pack - 70 oz.	35192.1869	100.00%

En el numerador del miembro calculado, la función CURRENTMEMBER se ha utilizado para obtener una referencia al miembro actual de la jerarquía Product Categories de la dimensión Product durante la ejecución de la consulta. En el denominador, se ha utilizado la función PARENT para obtener una referencia al miembro padre del producto actual. En esta expresión, el PARENT del CURRENTMEMBER de cada producto es la subcategoría del producto.

Nota:

La expresión del numerador:

`(([Product].[Product Categories].CURRENTMEMBER, [Measures].[Sales Amount]))`

Obtiene el valor de la medida Sales Amount para el miembro actual de la dimensión Product.

La expresión del denominador:

`(([Product].[Product Categories].CURRENTMEMBER.PARENT, [Measures].[Sales Amount]))`

Obtiene el valor de la medida Sales Amount para el padre del miembro actual de la dimensión Product.

Si se desea comparar las ventas de cada producto con las ventas de su respectiva categoría de producto, se puede modificar la expresión anterior de la siguiente manera:

```
WITH MEMBER MEASURES.PercentageSales AS
'([Product].[Product Categories].CURRENTMEMBER, [Measures].[Sales
Amount]) /
([Product].[Product Categories].CURRENTMEMBER.PARENT.PARENT,
[Measures].[Sales
Amount])',
FORMAT_STRING = 'PERCENT'
SELECT {MEASURES.[Sales Amount], MEASURES.PercentageSales} ON
COLUMNS,
[Product].[Product Categories].[Product Name].MEMBERS ON ROWS
FROM [Sales]
```

En este caso, se ha escrito la función PARENT dos veces para llegar hasta la categoría del producto. Una alternativa es utilizar la función ANCESTOR que devuelve para un miembro determinado, a su ancestro ubicado en un nivel específico.

```
WITH MEMBER MEASURES.PercentageSales AS
'([Product].[Product Categories].CURRENTMEMBER, [Measures].[Sales
Amount]) /
(Ancestor([Product].[Product Categories].CURRENTMEMBER,
[Product].[Product Categories].[Product Category Name]), [Measures].[Sales
Amount])',
FORMAT_STRING = 'PERCENT'
SELECT {MEASURES.[Sales Amount], MEASURES.PercentageSales} ON
COLUMNS,
[Product].[Product Categories].[Product Name].MEMBERS ON ROWS
FROM [Sales]
```

Si se desea mostrar en las filas de una consulta todas las categorías de productos (nivel Product Category Name de la jerarquía Product Categories), excluyendo a los miembros Clothing y Unknown se puede utilizar la función EXCEPT para efectuar esta exclusión.

```
SELECT {Measures.[Order Quantity], Measures.[Sales Amount]} ON
COLUMNS,
EXCEPT([Product].[Product Categories].[Product Category Name].MEMBERS,
{[Product].[Product Categories].[Product Category Name].[Clothing],
[Product].[Product Categories].[Product Category Name].[Unknown]}) ON
ROWS
FROM SALES
```

Los resultados son los siguientes:

	Order Quantity	Sales Amount
Accessories	15792	342540.3456
Bikes	53544	48031621.1084
Components	33937	8210376.132

La función EXCEPT tiene la siguiente sintaxis:

```
EXCEPT(set1, set2 [, ALL])
```

Esta función devuelve todos los miembros de set1, eliminando a los miembros de set2. Además, si en set1 hay elementos duplicados, quita las repeticiones. Si se desea retener las repeticiones, se debe especificar la cláusula ALL.

Un requerimiento común es mostrar el crecimiento en ventas entre dos períodos consecutivos de tiempo. La siguiente sentencia muestra el crecimiento de ventas de cada mes del año 2004, respecto del mes anterior:

```
WITH MEMBER Measures.CrecimientoVentas AS
'([Measures].[Sales Amount], [Order Date].[Calendar].CURRENTMEMBER) -
([Measures].[Sales Amount], [Order
Date].[Calendar].CURRENTMEMBER.PREVEMBER)'
SELECT {[Measures].[Sales Amount], Measures.CrecimientoVentas} ON
COLUMNS,
DESCENDANTS([Order Date].[Calendar].[2004],[Order
Date].[Calendar].[Month]) ON ROWS
FROM Sales
```

La función PREVEMBER se ha utilizado para obtener el miembro previo al actual en la jerarquía Calendar de la dimensión Order Date. Por ejemplo, Si el miembro actual es

February 2004, la función PREVEMBER hace referencia a January 2004. Los resultados obtenidos son los siguientes:

	Sales Amount	CrecimientoVentas
January 2004	1158360.8956	-843838.695699994
February 2004	1957392.74009999	799031.844499993
March 2004	1627296.0947	-330096.645399996
April 2004	1464971.7857	-162324.308999998
May 2004	2346755.17699999	881783.391299993
June 2004	2037059.35069999	-309695.826300002
July 2004	(null)	-2037059.35069999
August 2004	(null)	(null)

Si se desea comparar las ventas actuales contra las ventas del siguiente mes, se debe utilizar la función NEXTMEMBER.

La función GENERATE

La función GENERATE tiene la siguiente sintaxis:

GENERATE(set1, set2 [, ALL])

Esta función aplica el set2 a cada miembro de set1 y retorna el set generado por esta operación. Por ejemplo, la siguiente cláusula:

GENERATE({Perú, México, Ecuador}, {Bebidas, Condimentos}, ALL)

Retorna el set {Bebidas, Condimentos, Bebidas, Condimentos, Bebidas, Condimentos}. Para cada miembro del primer set (países) se ha generado repetidamente el segundo set (productos). Las repeticiones se han conservado debido a la inclusión de la cláusula ALL. Si se desea eliminar las repeticiones, la cláusula correcta es la siguiente:

GENERATE({Perú, México, Ecuador}, {Bebidas, Condimentos})

En este caso, el resultado es {Bebidas, Condimentos}

La función GENERATE es muy útil cuando se combina con la función CURRENTMEMBER. La siguiente sentencia genera, por cada miembro en el nivel Year de la jerarquía Calendar en la dimensión Order Date, un set conformado por dicho miembro y sus hijos directos. El resultado de esta operación se muestra en las columnas de la consulta.

```
SELECT GENERATE([Order Date].Calendar.Year.MEMBERS,
{[Order Date].Calendar.CURRENTMEMBER,
[Order Date].Calendar.CURRENTMEMBER.CHILDREN}) ON COLUMNS,
[Product].[Product Categories].[Product Category Name].MEMBERS ON ROWS
FROM Sales
WHERE Measures.[Sales Amount]
```

Los resultados obtenidos son los siguientes:

	2001	Semester 2 - 2001	2002	Semester 1 - 2002	Semester 2 - 2002
Accessories	15087.8071	15087.8071	65107.0146	12838.614	52268.4006
Bikes	6024627.35299994	6024627.35299994	15309837.3262999	6184983.18349992	9124854.14280001
Clothing	26463.0040999999	26463.0040999999	338878.998	24534.2016999999	314344.7963
Components	485897.684799999	485897.684799999	2685552.12110002	434203.104399999	2251349.01670002

En el siguiente ejemplo, se utiliza la función GENERATE para mostrar a los descendientes en el nivel City de los países United States y France:

```
SELECT [Product].[Product Categories].[Product Category Name].MEMBERS
ON
COLUMNS,
GENERATE({[Geography].[Geographic location].[United States],
[Geography].[Geographic location].[France]},
DESCENDANTS([Geography].[Geographic location].CURRENTMEMBER,
[Geography].[Geographic location].[City])) ON ROWS
FROM Sales
WHERE Measures.[Sales Amount]
```

Los resultados son los siguientes:

	Accessories	Bikes	Clothing	Components
Birmingham	(null)	18546.521	503.928	1460.286
Florence	(null)	939.588	(null)	832.8436
Huntsville	(null)	22372.506	(null)	773.3564
Mobile	(null)	(null)	(null)	(null)
Montgomery	(null)	(null)	(null)	(null)
Chandler	(null)	297707.3519	117.582	58256.6349
Gilbert	(null)	20940.718	(null)	7075.98
Mesa	(null)	81021.6498	1354.4577	31918.3214
Phoenix	8415.1535	503940.613500001	29066.7825	144272.6596
Scottsdale	261.831	122751.5578	424.572	16650.7918
Surprise	161.492	8632.9981	1864.4828	1080.9065
Tucson	(null)	85451.3684	11.4	11206.2646
Alhambra	2.994	24915.324	345.2088	4264.4708
Alpine	(null)	(null)	(null)	(null)
Auburn	(null)	(null)	(null)	2803.9816

Funciones TIME

Las funciones PARALLELPERIOD, CLOSINGPERIOD, OPENINGPERIOD, y PERIODSTODATE proveen potentes funcionalidades de manipulación de fechas.

La función PARALLELPERIOD permite comparar los valores para un miembro específico con otros valores en la misma posición, relativos a un período anterior. La siguiente sentencia compara el valor de la medida Sales Amount para un mes con el valor de Sales Amount para el mismo mes en el año anterior.

```

WITH MEMBER Measures.Crecimiento AS
'[Measures].[Sales Amount] - 
([Measures].[Sales Amount], PARALLELPERIOD([Order
Date].[Calendar].[Year]))'
SELECT {[Measures].[Sales Amount], [Measures].Crecimiento} ON COLUMNS,
GENERATE({[Order Date].[Calendar].[2002], [Order Date].[Calendar].[2003]}, 
DESCENDANTS([Order Date].[Calendar].currentmember, [Order
Date].[Calendar].Month))
ON ROWS
FROM Sales

```

Los resultados son los siguientes:

	Sales Amount	Crecimiento
January 2002	54936.861100001	175369.1784
February 2002	1624157.7723	402730.5340999999
March 2002	1119935.9265	159679.723200001
April 2002	654832.2089	45018.4640999996
May 2002	1884474.0929	-57855.2253000003
June 2002	823822.2419	-620459.4199
July 2002	1632449.1672	1632449.1672
August 2002	2749773.24250001	2749773.24250001
September 2002	2112945.929	2112945.929
October 2002	1323525.3167	1323525.3167
November 2002	2394160.7018	2394160.7018
December 2002	1529961.9992	1529961.9992
January 2003	1015458.9832	466122.122099997
February 2003	1899020.82589039	274863.053599996
March 2003	1053856.879	-66079.0475000027
April 2003	366570.3912	711738.182299997
May 2003	290530.4746	406056.381700006

Observe que $C = A - B$; es decir, la medida calculada Crecimiento para enero 2003 es igual a la diferencia de la medida Sales Amount entre enero 2003 y enero 2002.

CROSSJOIN

MDX soporta el empleo de la función CROSSJOIN. Esta función produce todas las combinaciones de dos sets. Por ejemplo, la siguiente sentencia muestra, por cada miembro del nivel Promotion Category de la dimensión Promotion, a todos los miembros del nivel Product Category Name de la dimensión Product.

Los resultados son los siguientes:

		Sales Amount	Order Quantity
Customer	Accessories	(null)	(null)
Customer	Bikes	(null)	(null)
Customer	Clothing	(null)	(null)
Customer	Components	(null)	(null)
No Discount	Accessories	255502.392000003	11881
No Discount	Bikes	44912443.618801	48673
No Discount	Clothing	837160.163599971	29851
No Discount	Components	8123107.48250015	33440
Reseller	Accessories	87037.9536	3911
Reseller	Bikes	3119177.4896	4871
Reseller	Clothing	303093.7237	11900
Reseller	Components	87268.6495	497

4.1.5 Filtra y ordena Resultados

Los filtros en MDX reducen el número de miembros sobre los cuales se evalúa y calcula la información.

Un método sencillo para filtrar los miembros vacíos consiste en utilizar la cláusula NON EMPTY. Por ejemplo, la siguiente sentencia elimina a los miembros del crossjoin que tienen valores vacíos para las medidas Sales Amount y Order Quantity:

```
select {Measures.[Sales Amount], Measures.[Order Quantity]} on columns,
non empty crossjoin([Promotion].[Promotion Categories].[Promotion Category].members,
[Product].[Product Categories].[Product Category Name].members) on rows
from Sales
```

Los resultados de esta consulta son los siguientes:

		Sales Amount	Order Quantity
No Discount	Accessories	255502.392000003	11881
No Discount	Bikes	44912443.618801	48673
No Discount	Clothing	837160.163599971	29851
No Discount	Components	8123107.48250015	33440
Reseller	Accessories	87037.9536	3911
Reseller	Bikes	3119177.4896	4871
Reseller	Clothing	303093.7237	11900
Reseller	Components	87268.6495	497

Para filtros más específicos se debe utilizar la función FILTER:

```
FILTER(set1, condición_de_búsqueda)
```

Esta función aplica al set1 todos los miembros que cumplan el criterio de búsqueda. Por ejemplo, si se desea observar las ciudades cuyas unidades vendidas exceden a 25000, se puede escribir la siguiente sentencia:

```
SELECT {Measures.[Sales Amount], Measures.[Order Quantity]} ON
COLUMNS,
FILTER([Geography].[Geographic location].[State Province Name].MEMBERS,
Measures.[Sales Amount] > 25000) ON ROWS
FROM Sales
```

Los resultados de la consulta son los siguientes:

	Sales Amount	Order Quantity
Essonne	152420.3249	381
Garonne (Haute)	391040.5897	578
Hauts de Seine	485791.5132	1002
Loir et Cher	420190.2599	1523
Loiret	989510.130199999	3020
Nord	34735.2737	158
Pas de Calais	54633.6577	304
Seine (Paris)	1164554.3922	3793
Seine et Marne	41185.4658	78
Seine Saint Denis	173233.51	315

La función ORDER se emplea para ordenar los resultados de la consulta. Su sintaxis es:

```
ORDER(set, expresión [, ASC | DESC | BASC | BDESC])
```

La expresión de ordenamiento puede ser numérica o de cadena. El ordenamiento por defecto es ascendente. Por ejemplo, la siguiente consulta ordena los miembros del nivel

State Province Name de la dimensión Geography alfabéticamente por su nombre:

```
SELECT {Measures.[Sales Amount], Measures.[Order Quantity]} ON
COLUMNS,
ORDER([Geography].[Geographic location].[State Province Name].MEMBERS,
[Geography].[Geographic location].CURRENTMEMBER.NAME, ASC) ON
ROWS
FROM Sales
```

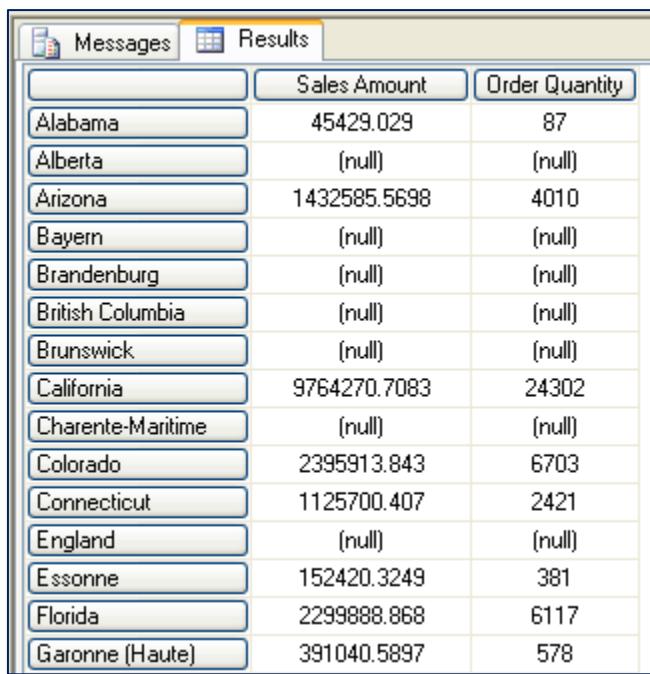
Los resultados son los siguientes:

	Sales Amount	Order Quantity
New South Wales	(null)	(null)
Queensland	(null)	(null)
South Australia	(null)	(null)
Tasmania	(null)	(null)
Victoria	(null)	(null)
Alberta	(null)	(null)
British Columbia	(null)	(null)
Brunswick	(null)	(null)
Manitoba	(null)	(null)
Ontario	(null)	(null)
Quebec	(null)	(null)
Charente-Maritime	(null)	(null)
Essonne	152420.3249	381

Observe que el resultado no se visualiza con un orden alfabético estricto. Esto se debe a que el parámetro ASC de la función ORDER ordena a los miembros del nivel State Province Name dentro de cada miembro padre en el nivel Country Region Name. Es decir, primero ordena alfabéticamente a New South Wales, Queensland, South Australia, Tasmania y Victoria, que son hijos de la región Australia. Luego, repite nuevamente la ordenación para los hijos de Canadá, y así sucesivamente. Si se desea evitar este comportamiento, debe especificarse el parámetro BASC:

```
SELECT {Measures.[Sales Amount], Measures.[Order Quantity]} ON
COLUMNS,
ORDER([Geography].[Geographic location].[State Province Name].MEMBERS,
[Geography].[Geographic location].CURRENTMEMBER.NAME, BASC) ON
ROWS
FROM Sales
```

En este caso, el ordenamiento sí es absoluto.

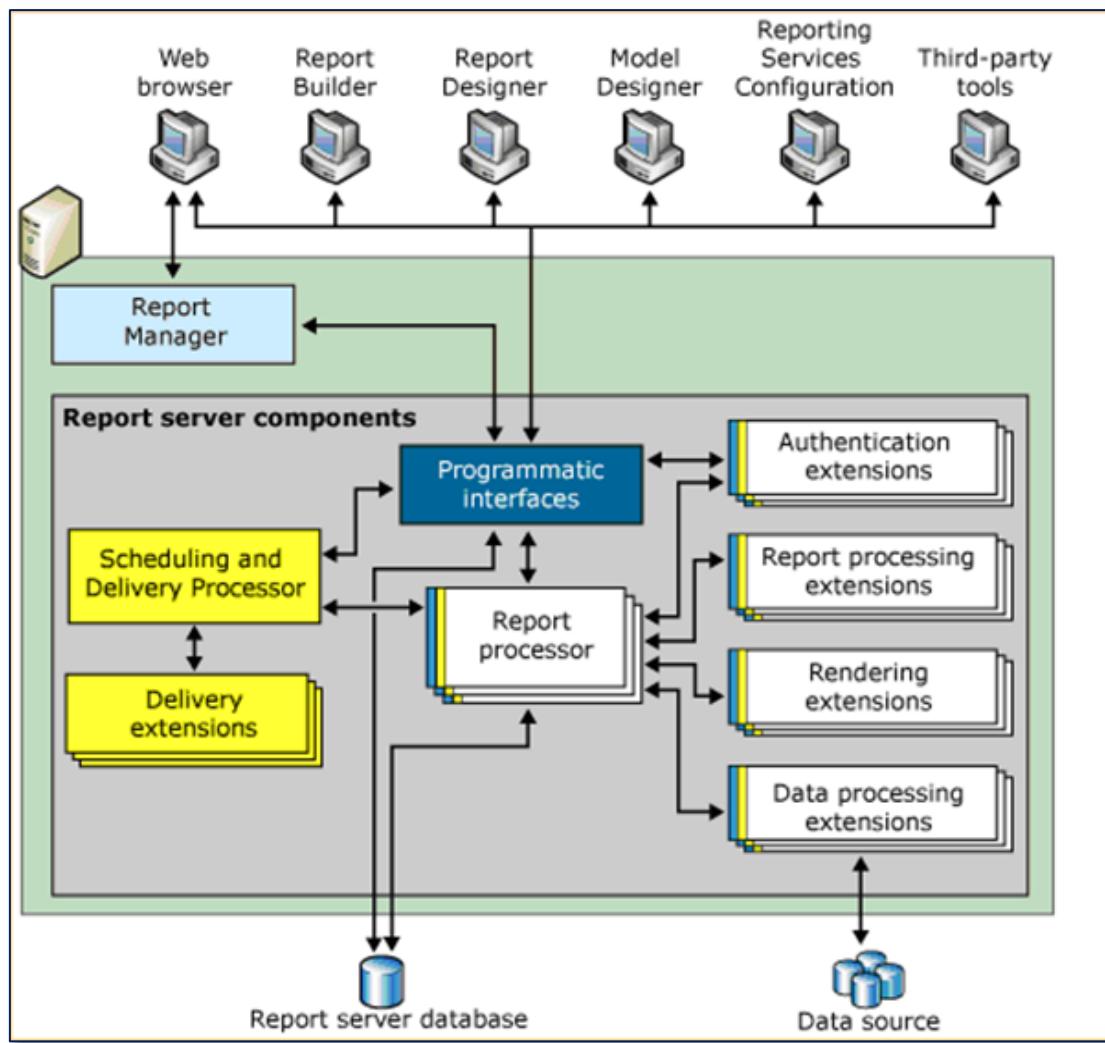


	Sales Amount	Order Quantity
Alabama	45429.029	87
Alberta	(null)	(null)
Arizona	1432585.5698	4010
Bayern	(null)	(null)
Brandenburg	(null)	(null)
British Columbia	(null)	(null)
Brunswick	(null)	(null)
California	9764270.7083	24302
Charente-Maritime	(null)	(null)
Colorado	2395913.843	6703
Connecticut	1125700.407	2421
England	(null)	(null)
Essonne	152420.3249	381
Florida	2299888.868	6117
Garonne (Haute)	391040.5897	578

4.2 Creación de reportes utilizando SSRS

4.2.1 Introducción a SSRS

SSRS 2014 es una plataforma de reportes basada en un servidor de capa intermedia (middle – tier Server). Permite construir reportes complejos que extraen información de múltiples fuentes de datos, de manera sencilla e intuitiva. El gráfico muestra la arquitectura de Reporting Services.



Los principales componentes de la arquitectura de Reporting Services:

Report Server (Servidor de Reportes): Su papel es administrar las conexiones con los orígenes de datos, procesar los reportes y enviarlos hacia los clientes. Es el componente principal en la arquitectura de Reporting Services.

- **Report Manager (Administrador de Reportes):** Es una herramienta que permite administrar el servidor de reportes. Provee una interfaz basada en la Web, a través de la cual se pueden visualizar y agrupar los reportes. Permite también configurar restricciones de visualización en los reportes.
- **Report Designer (Diseñador de Reportes):** Es una herramienta que permite construir y publicar reportes a través del SQL Server Data Tools.
- **Report Server Database (Base de datos del Servidor de Reportes):** Es una base de datos SQL Server 2014, que almacena las definiciones y la meta data de los reportes. No debe ser modificada manualmente por los usuarios.
- **Data Source (Origen de datos):** Reporting Services es capaz de extraer información de SQL Server, Oracle, Analysis Services y cualquier origen OLEDB y ODBC.

4.2.2 Creación de proyectos SSRS

El diseñador de reportes de Reporting Services permite construir y publicar reportes, a través del entorno de desarrollo de SQL Data Tools.

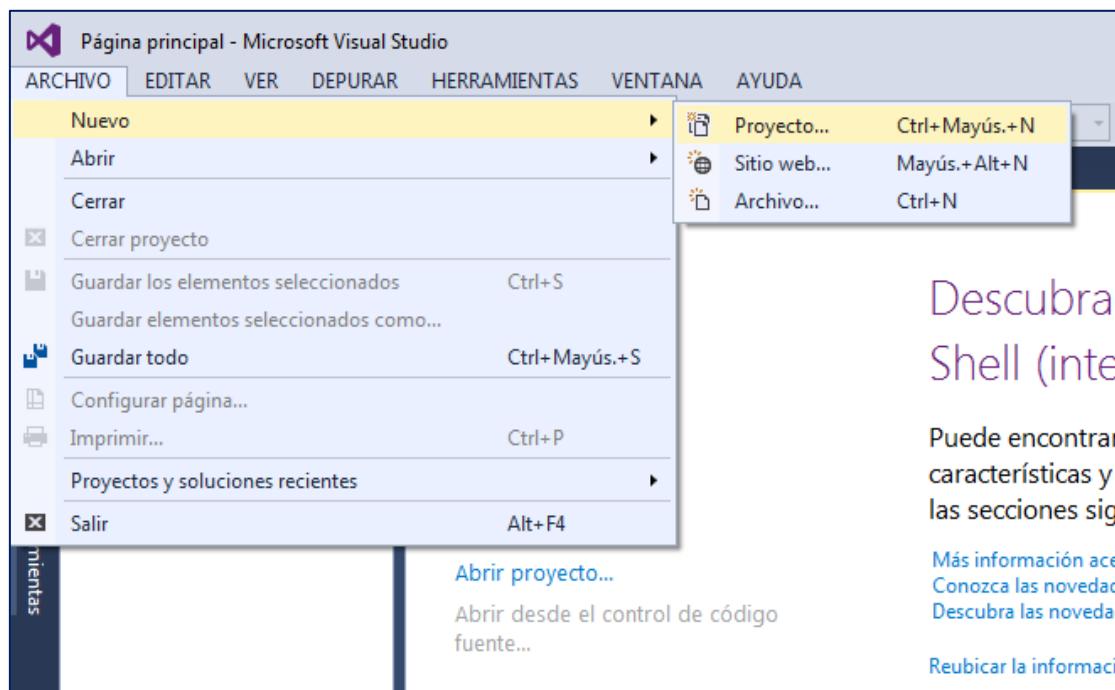
Los principales componentes de un reporte de Reporting Services son:

- Data Sources: Definen los orígenes de datos del reporte. Un reporte puede conectarse con múltiples orígenes de datos.
- Datasets: Contienen vistas de datos, obtenidos a partir de los data sources. Un reporte puede obtener su información desde múltiples datasets.
- Fields: Contienen las definiciones de las columnas de los datasets. Permiten diseñar gráficamente los reportes.
- Regiones de datos: Una región de datos define un formato de visualización. Reporting Services cuenta con tablas, matrices, listas, subreportes, etc.

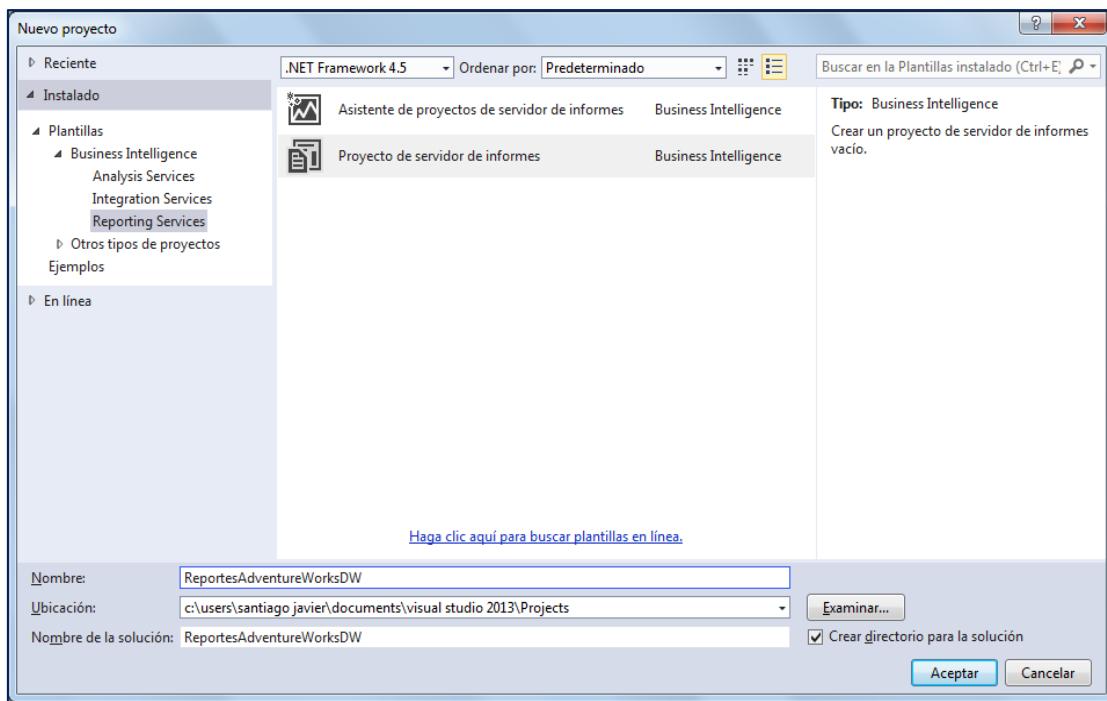
Ejercicio:

Crear Reporte Tabular sobre el cubo “Sales”

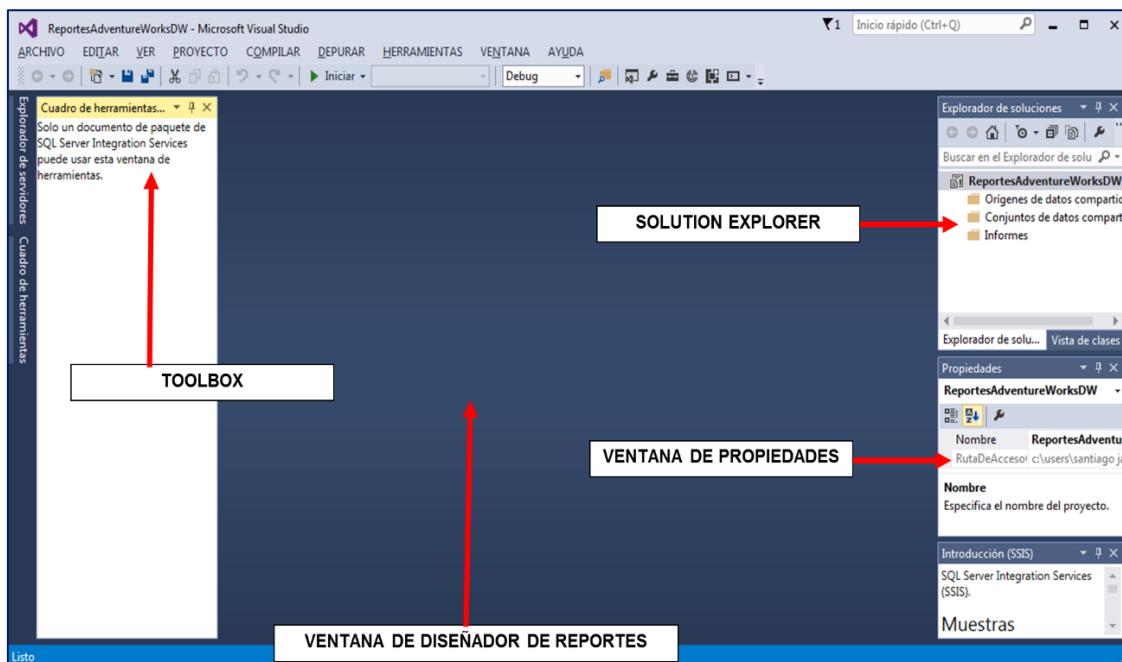
1. Abra el entorno de desarrollo del SQL Server Data Tools.
2. En el menú File, seleccione la opción New → Project:



3. Se abrirá la ventana de creación de nuevos proyectos .Net. Seleccione la plantilla de proyecto Report Server Project. En la casilla Name se debe escribir el nombre del nuevo proyecto de reportes; establezca esta propiedad a “ReportesAdventureWorksDW”. La casilla Location permite establecer el directorio del proyecto. Utilice el botón Browse para ubicar la carpeta en la que desea ubicar su nuevo proyecto de reportes:

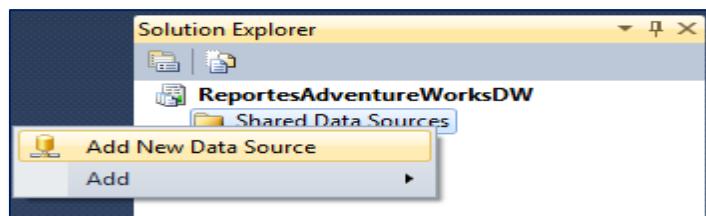


4. Presione el botón OK. Se abrirá el entorno del Diseñador de Reportes. La siguiente figura muestra los componentes del entorno de desarrollo de .Net:

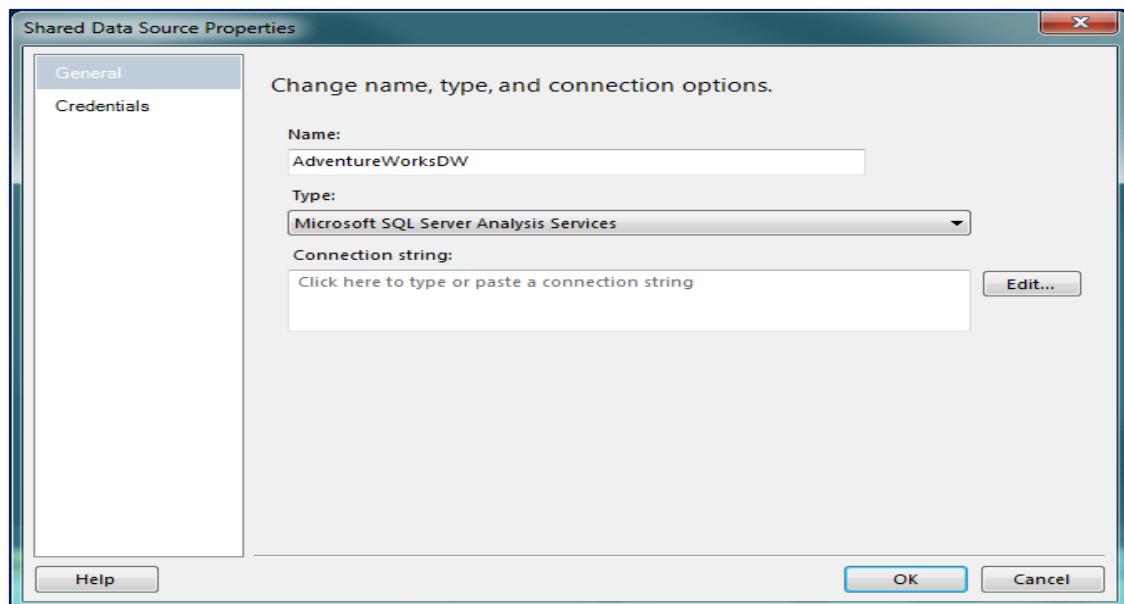


4.2.2.1 Establecer Fuente de Datos

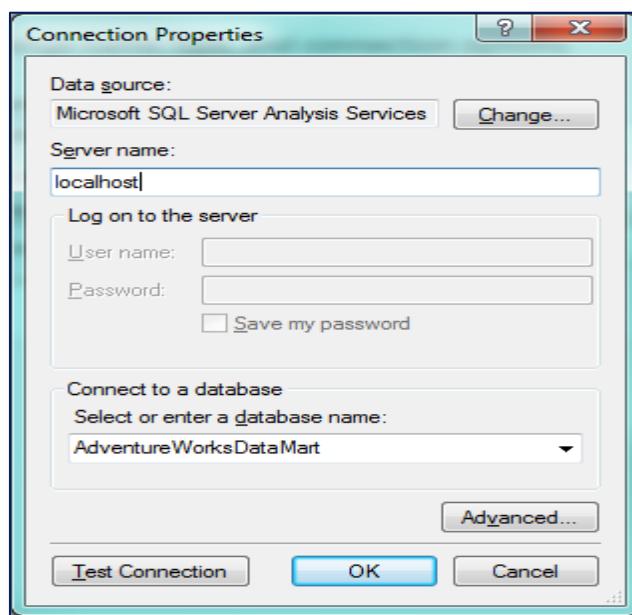
5. Agregue un nuevo origen de datos al proyecto. Para esto, haga un clic derecho sobre el folder Shared Data Sources en el explorador de soluciones, y seleccione la opción Add New Data Source:



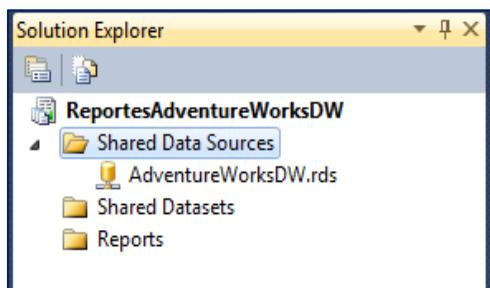
6. Aparecerá la ventana de propiedades del data source. En la propiedad Name, escriba “AdventureWorksDW”. En la propiedad Type, seleccione el tipo “Microsoft SQL Server Analysis Services”:



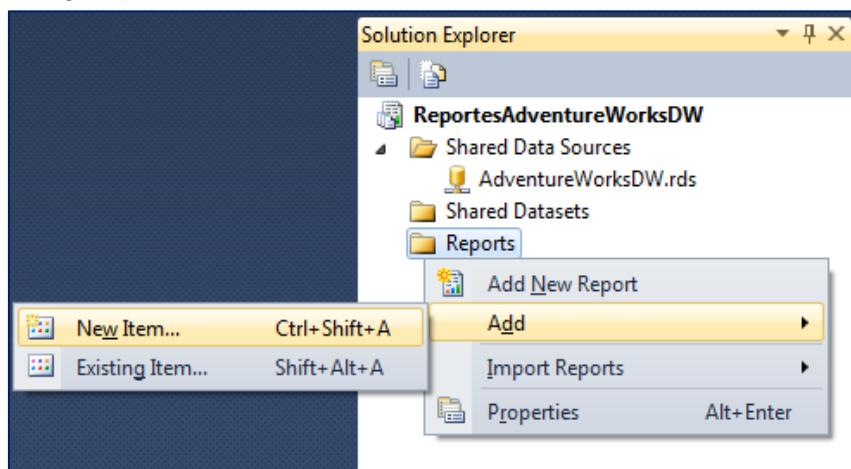
7. Pulse el botón Edit. Aparecerá la ventana de configuración de la conexión. En el nombre del servidor, escriba “localhost” para designar el servidor local. En la propiedad Select o enter a database name, seleccione la base de datos “AdventureWorksDataMart”:



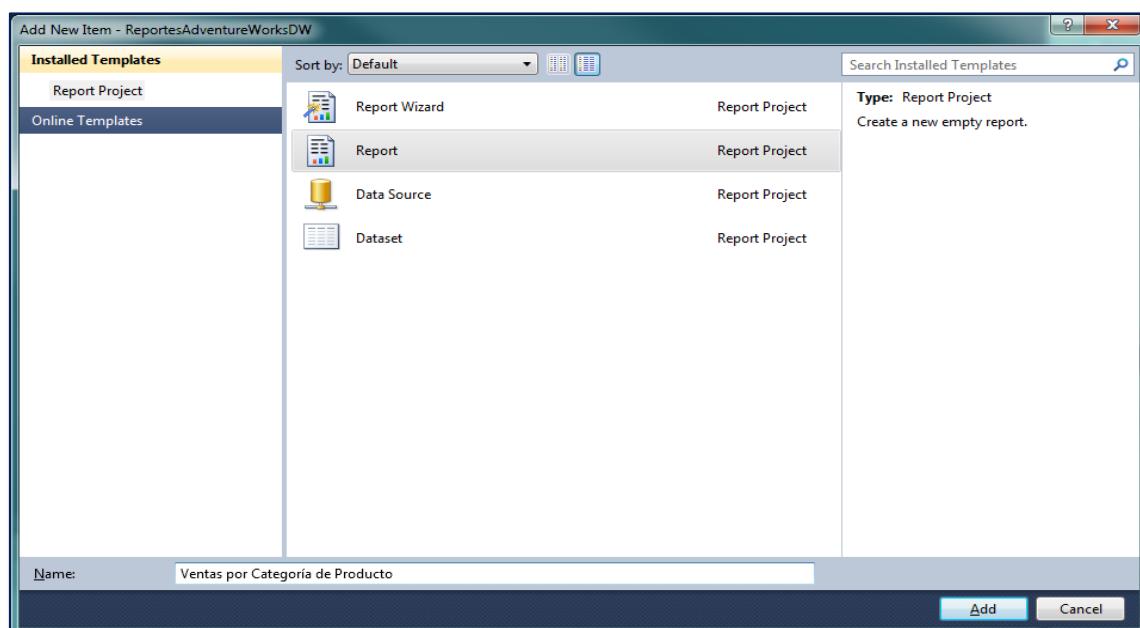
8. Prueba la conexión con el botón Test Connection. Si la prueba es satisfactoria, presione el botón OK. Luego, pulse nuevamente el botón OK. Esto finalizará la creación del nuevo origen de datos.



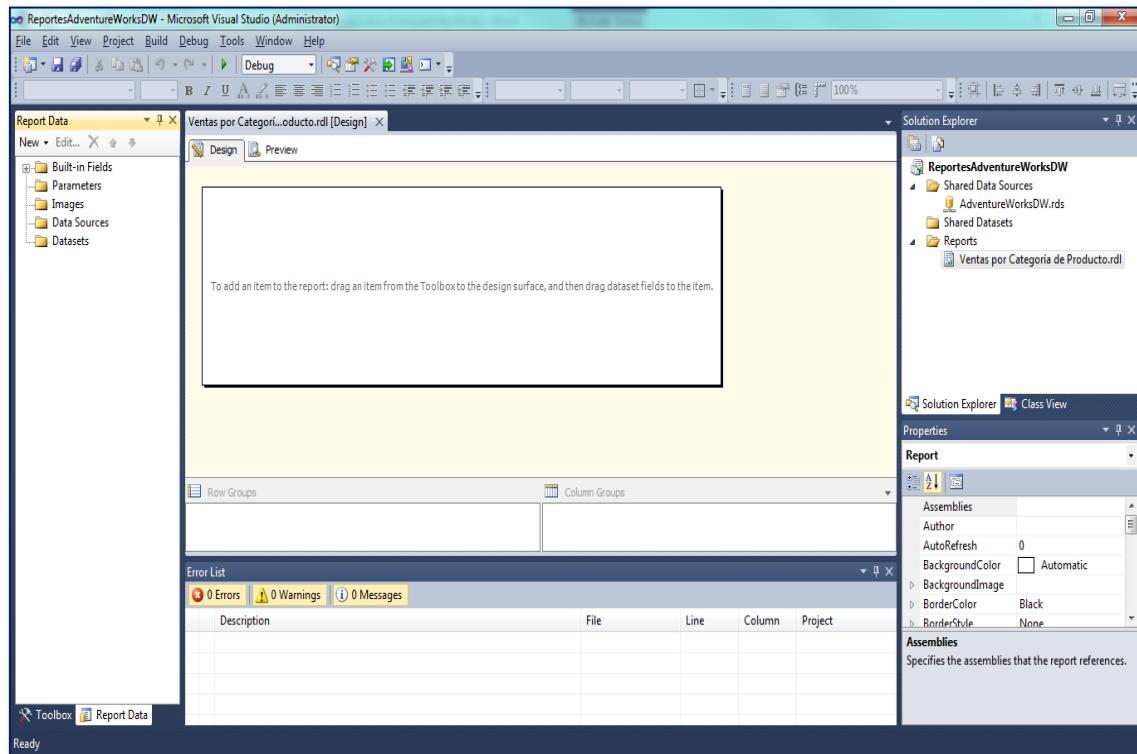
9. Para crear un nuevo reporte, haga un clic derecho sobre el folder Reports en el explorador de la solución, y seleccione la opción Add New Item (observe la siguiente figura):



10. Aparecerá la ventana de configuración del nuevo reporte. En el cuadro Templates, seleccione la opción Report. Nombre al reporte como “Ventas por Categoría de Producto”:

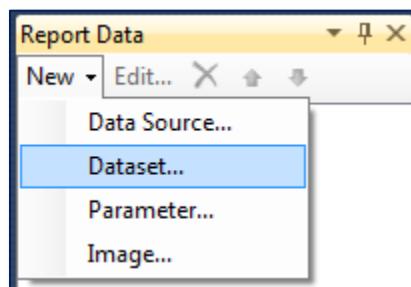


11. Presione el botón Add. Aparecerá el diseñador de reportes. Este diseñador consta de un panel Report Data y dos tabs: Design y Preview. El panel Report Data permite configurar los datasets que utilizará el reporte. La pestaña Design permite diseñar gráficamente el reporte. La pestaña Preview ejecuta el reporte y proporciona una vista preliminar de éste (ver la siguiente figura).

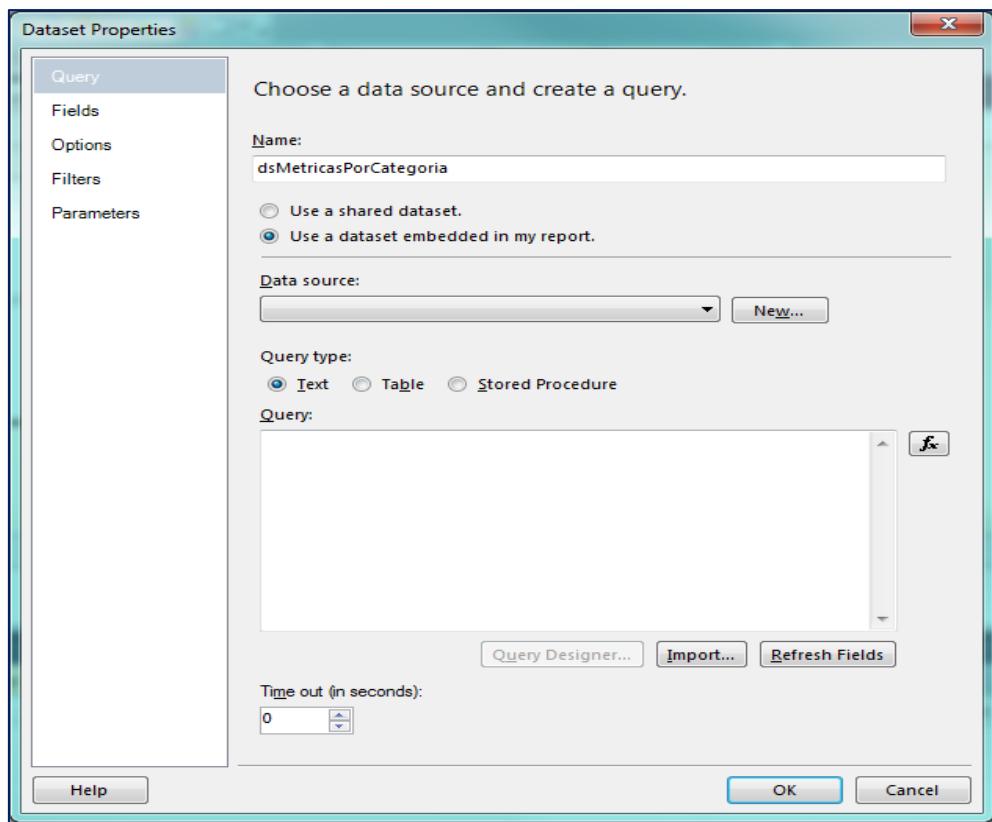


4.2.2.2 Establecer conjunto de Datos y Creación de consultas

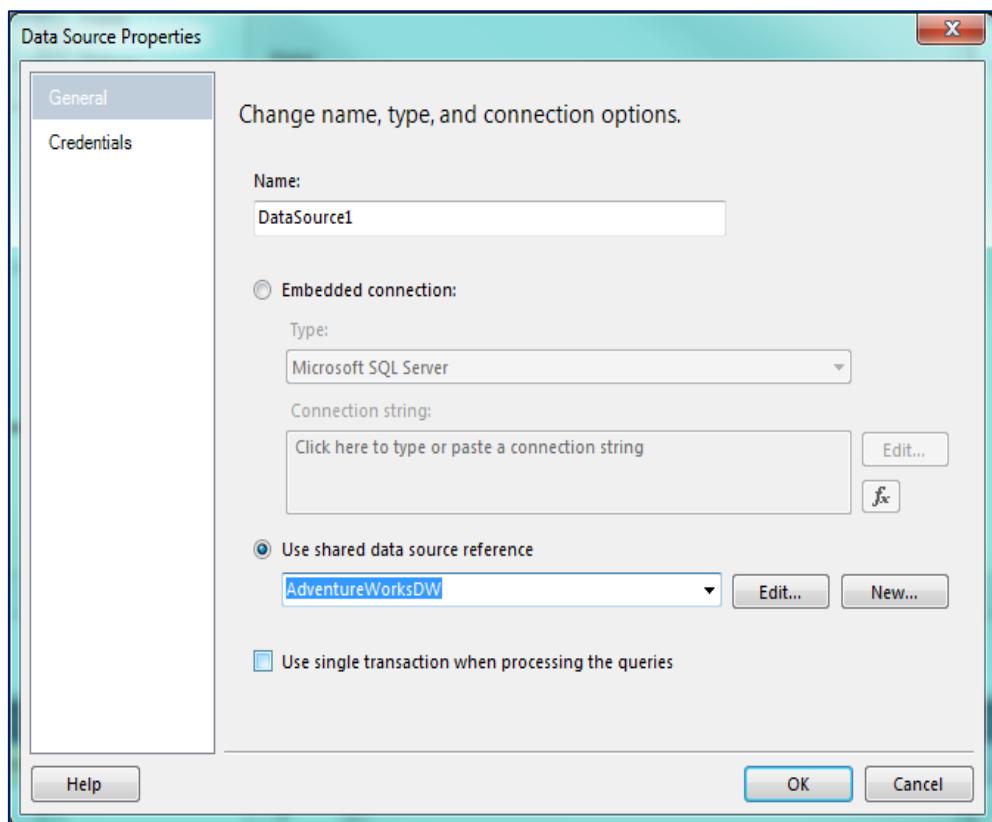
12. En la pestaña Report Data, seleccione la lista New y haga clic sobre la opción Dataset:



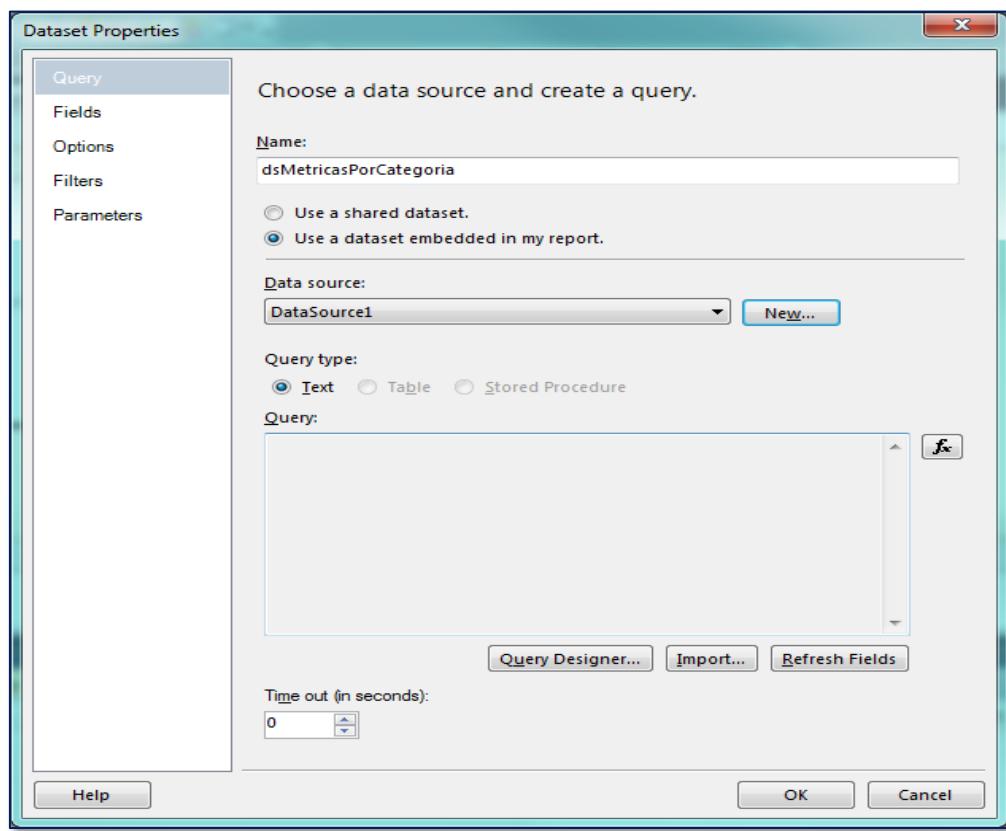
13. En la ventana de edición del nuevo dataset, establezca como nombre dsMetricasPorCategoria, seleccione la opción Use a dataset embedded in my report y de clic en el botón New:



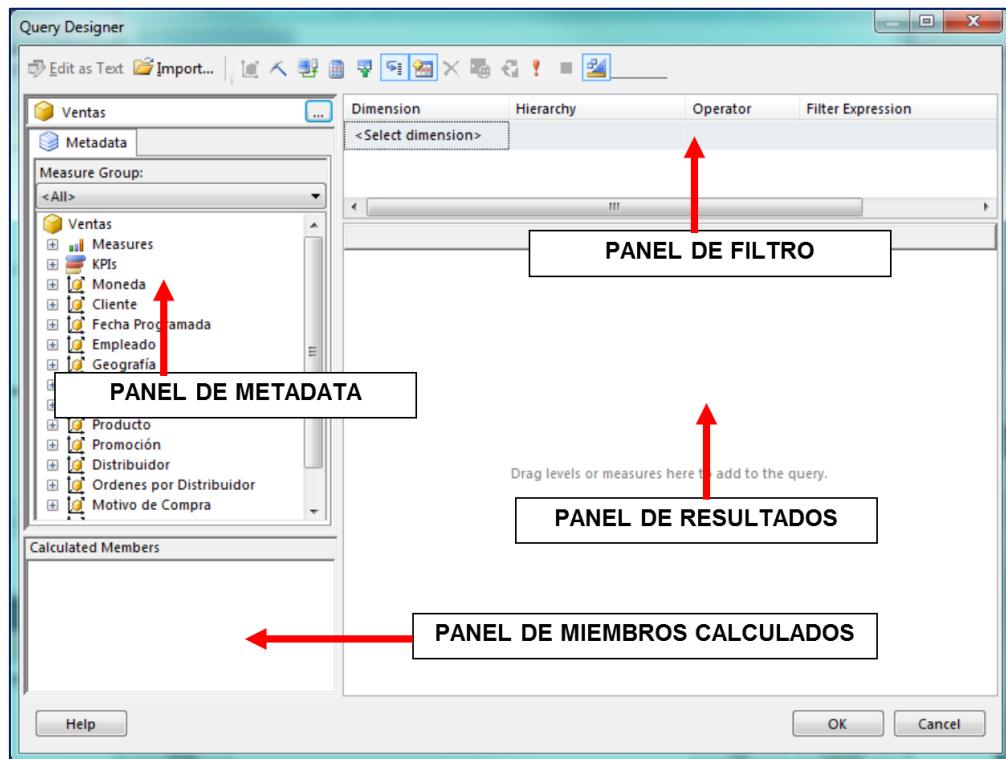
14. En la siguiente pantalla, en la opción Use Shared data source reference, elija AdventureWorksDW y luego el botón Ok:



15. En la siguiente pantalla, presione el botón Query Designer.



16. A continuación, aparecerá la pantalla del Query Designer, elija el cubo Ventas (Sales).



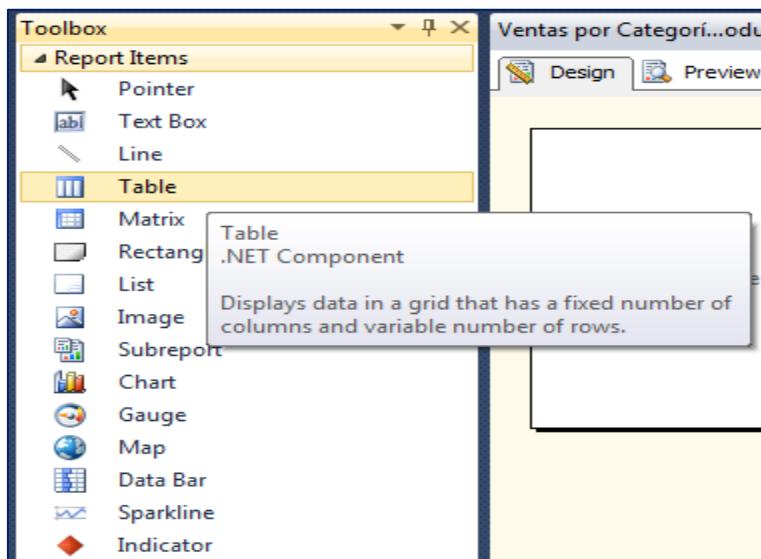
17. Desde el panel de metadata, arrastre las medidas Monto por Distribuidor (“Sales Amount”) y Monto por Internet (“Internet Sales Amount”), y el atributo Categoría de Producto (“Product Category Name”) de la dimensión Producto (“Product”) sobre el panel de resultados. Las columnas elegidas y sus resultados aparecerán sobre el panel de resultados.

Categoría de Producto	Monto por Distribuidor	Monto por Internet
Accessories	342540.345600002	396772.139999902
Bikes	48031621.1083984	14105393.3186008
Clothing	1140253.88729997	191299.859999993
Components	8210376.13199995	(null)

18. Haga dos veces clic en el botón Ok. Observe que se muestra la lista de campos del dataset con las columnas seleccionadas para la consulta. Las consultas MDX experimentan un proceso de “aplanamiento” (flatenning), que permite tratar los miembros y niveles referenciados por la consulta como si fueran campos de una tabla relacional.

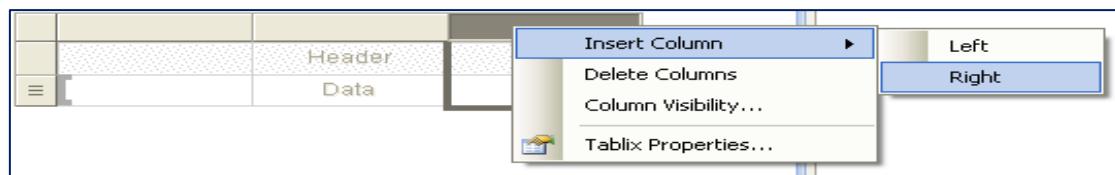
4.2.2.3 Diseño de Reportes

19. Seleccione el menú View y elija la opción Toolbox. A continuación, seleccione el tab Design. En el toolbox, elija la herramienta Table, y arrástrela sobre el panel de diseño:

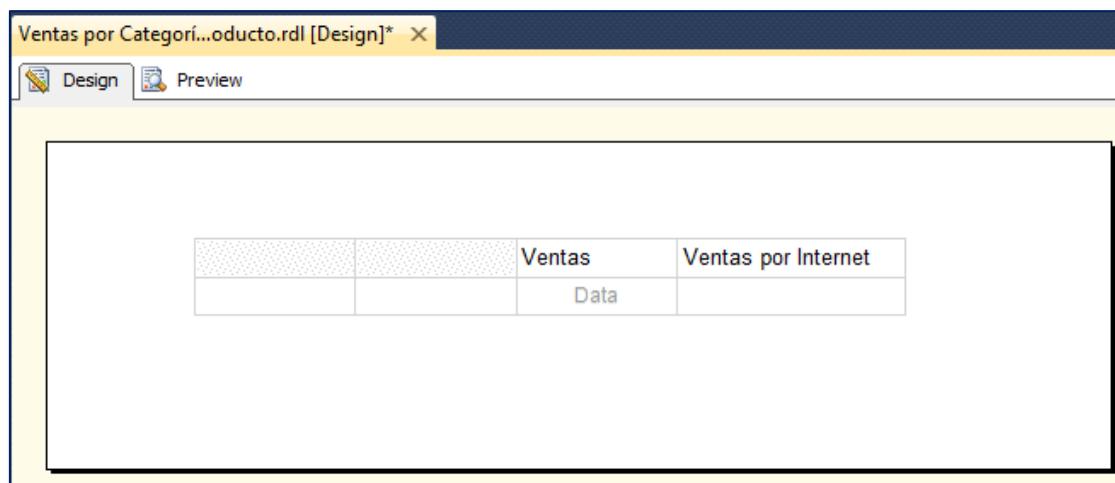


4.2.2.4 Uso de Tablix y Matrix

20. Aparecerá una tabla con dos filas y tres columnas. Para agregar una nueva columna a la tabla, haga un clic derecho sobre la cabecera de la tercera columna, y seleccione la opción Insert Column y Right:



21. Escriba sobre las cabeceras de las dos últimas columnas, como se muestra en la siguiente figura:



22. Seleccione el menú View y elija la opción Report Data para abrir la ventana de datasets. Seleccione el campo Categoría de Producto y arrástrelo sobre la segunda fila de la tabla, en la segunda celda. Esto hará que el reporte cargue la lista de categorías de productos, en la segunda columna de cada fila. Renombre la cabecera de la columna a “Categorías de productos” si fuera necesario. Al finalizar, la ventana de diseño tendrá la siguiente apariencia:



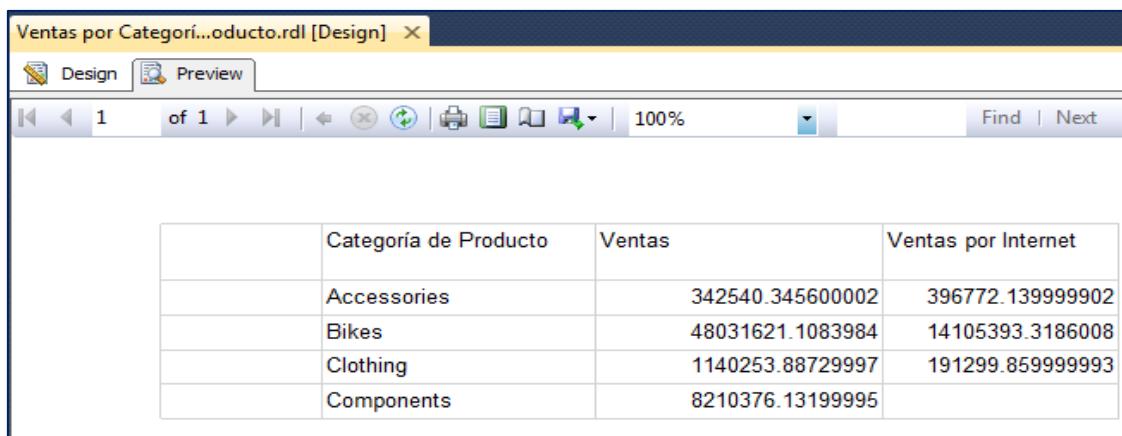
The screenshot shows the 'Ventas por Categoría de Producto.rdl [Design]' window. At the top, there are tabs for 'Design' and 'Preview'. Below the tabs is a table with four columns. The first column has a dotted pattern. The second column is labeled 'Categoría de Producto' and contains the expression '[Categoría_de_Producto]'. The third column is labeled 'Ventas' and contains the expression 'Data'. The fourth column is labeled 'Ventas por Internet'. The table is currently empty.

23. Debajo de la columna Ventas, seleccione el ícono  y elija el campo Monto por Distribuidor (Sales_Amount); igualmente debajo de la columna Ventas por Internet seleccione el campo Monto por Internet (Internet_Sales_Amount). Al finalizar, el entorno de desarrollo tendrá la siguiente apariencia:



The screenshot shows the same 'Ventas por Categoría de Producto.rdl [Design]' window. The table now has four columns. The first column has a dotted pattern. The second column is labeled 'Categoría de Producto' and contains the expression '[Categoría_de_Producto]'. The third column is labeled 'Ventas' and contains the expression '[Monto_por_Distribuidor]'. The fourth column is labeled 'Ventas por Internet' and contains the expression '[Monto_por_Internet]'. The table is currently empty.

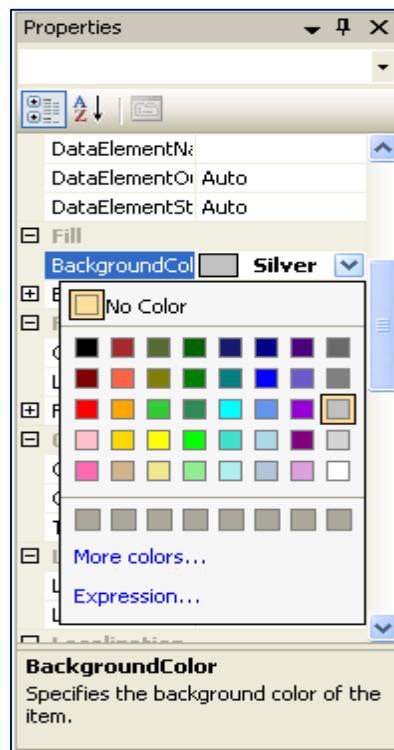
24. Seleccione la pestaña Preview para obtener una vista preliminar del reporte:



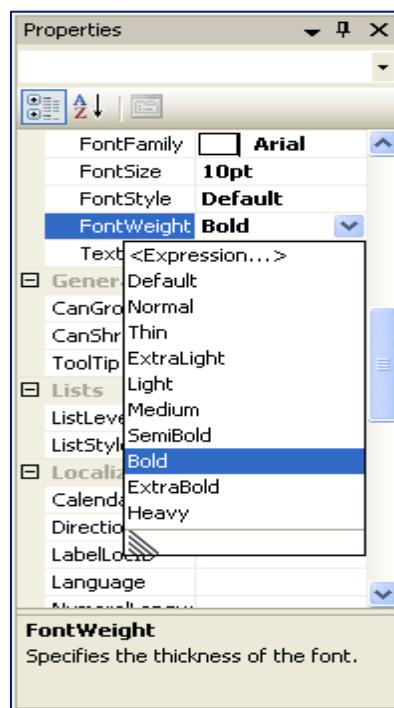
The screenshot shows the 'Ventas por Categoría de Producto.rdl [Design]' window with the 'Preview' tab selected. At the top, there are navigation buttons for pages, orientation, and zoom levels (100%). Below the preview area is a table with four columns. The first column has a dotted pattern. The second column is labeled 'Categoría de Producto' and contains the values 'Accessories', 'Bikes', 'Clothing', and 'Components'. The third column is labeled 'Ventas' and contains the values '342540.345600002', '48031621.1083984', '1140253.88729997', and '8210376.13199995'. The fourth column is labeled 'Ventas por Internet' and contains the values '396772.139999902', '14105393.3186008', '191299.8599999993', and an empty cell.

	Categoría de Producto	Ventas	Ventas por Internet
	Accessories	342540.345600002	396772.139999902
	Bikes	48031621.1083984	14105393.3186008
	Clothing	1140253.88729997	191299.8599999993
	Components	8210376.13199995	

25. A continuación, se modificará la apariencia del reporte, para que los encabezados de columna aparezcan con fondo gris. Para esto, regrese al tab Design y seleccione las cuatro celdas de la primera fila. En la ventana de propiedades, cambie el valor de la propiedad BackgroundColor de la siguiente manera:

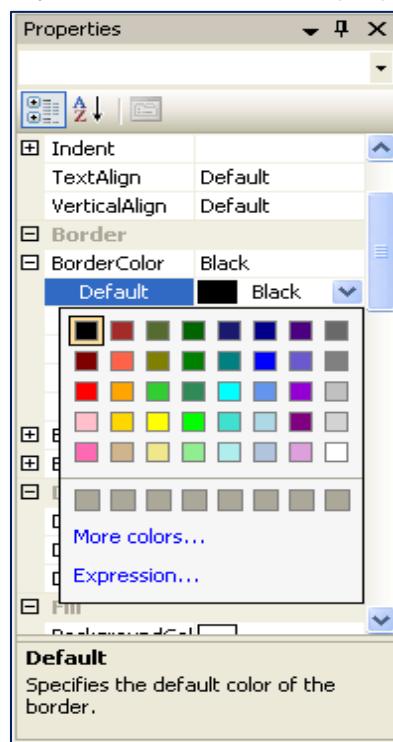


26. Para que los encabezados de columna aparezcan en negrita, seleccione la primera fila del reporte. En la ventana de propiedades, expanda la propiedad Font, y cambie el valor de la propiedad FontWeight a Bold:



Al finalizar, la vista de diseño debe mostrar la siguiente apariencia:

27. Para colocar bordes en la tabla de color negro, seleccione todas las celdas de la tabla utilizando el mouse. A continuación, en la ventana de propiedades expanda la propiedad BorderColor, y cambie el valor de la propiedad Default a Black:



28. Seleccione la pestaña Preview para obtener la vista preliminar del reporte:

4.3 Optimización y publicación de Reportes

4.3.1 Optimizar Reportes

4.3.1.1 Uso de parámetros

Usos comunes de Parámetros

Estas son algunas de las formas más comunes de utilizar parámetros.

Control de datos de informe

- Filtro informe de datos en el origen de datos al escribir consultas de conjuntos de datos que contienen variables.
- Filtrar los datos de un conjunto de datos compartido. Cuando se agrega un conjunto de datos compartido a un informe, no se puede cambiar la consulta. En el informe, se puede añadir un filtro de conjunto de datos que incluye una referencia a un parámetro de informe que cree.
- Permite a los usuarios especificar valores para personalizar los datos de un informe. Por ejemplo, proporcionar dos parámetros para la fecha de inicio y fecha de finalización de los datos de ventas.

Conecete Informes Relacionados

- Usar parámetros de relacionar los informes principales de informes detallados, subinformes a, y para informes vinculados. Cuando se diseña un conjunto de informes, puede diseñar cada informe para contestar ciertas preguntas. Cada informe puede proporcionar una visión diferente o distinto nivel de detalle de la información relacionada. Para proporcionar un conjunto de informes relacionados entre sí, crear parámetros para los datos relativos a los informes de destino.

Para obtener más información, consulte los informes de obtención de detalles (Generador de informes y SSRS), subinformes (Generador de informes y SSRS) , y crear un informe vinculado .

- Personaliza conjuntos de parámetros para múltiples usuarios. Cree dos informes vinculados con base en un informe de ventas en el servidor de informes. Un informe vinculado utiliza valores de los parámetros predefinidos para las personas de ventas y el segundo informe vinculado utiliza valores de los parámetros predefinidos para los gerentes de ventas. Ambos informes utilizan la misma definición de informe.

En un informe, los parámetros se usan para filtrar datos, modificar su apariencia y conectarse a informes relacionados. Cuando se crea un parámetro de informe y se ejecuta un informe, se da al usuario la posibilidad de elegir valores para el parámetro de informe. En el informe, se pueden escribir expresiones que hacen referencia en tiempo de ejecución al valor del parámetro y cambian los datos o el aspecto del informe. También se usan parámetros cuando se incluye un vínculo a otro informe o a un subinforme del informe principal.

Agregar parámetros para filtrar informes por fecha

Al incluir parámetros para las fechas de inicio y de finalización en una consulta del informe, se puede especificar un intervalo de fechas que limite los datos recuperados del origen de datos. Se pueden crear parámetros adicionales para filtrar los datos recuperados del origen de datos.

En esta lección, agregará los parámetros @StartDate y @EndDate a la consulta para limitar los datos recuperados del origen de datos. En el panel Datos de informe, se crean y aparecen automáticamente los parámetros de informe StartDate y EndDate. Los parámetros distinguen entre mayúsculas y minúsculas. Los parámetros de consulta comienzan por el símbolo @, mientras que los parámetros de informe no.

Establecerá el tipo de datos de los parámetros en DateTime y observará que aparece un control de calendario con el cuadro de texto del parámetro en la barra de herramientas del visor de informes. Establecerá valores predeterminados para los parámetros de manera que el informe se ejecute automáticamente. Finalmente, creará un parámetro de informe DayofWeek que no esté enlazado a ningún parámetro de consulta y lo usará para filtrar los datos recuperados desde el origen de datos.

Para abrir un proyecto de servidor de informes existente:

- Haga clic en Inicio, seleccione Todos los programas, seleccione Microsoft SQL Server 2014 y, a continuación, haga clic en SQL Server Data Tools.
- En el menú Archivo, seleccione Abrir y haga clic en Proyecto o solución.
- Haga clic en Tutorial y, a continuación, elija Tutorial.sln. Este es el tutorial que se creó en Tutorial: Crear un informe de tabla básico. ([https://technet.microsoft.com/es-es/library/ms167305\(v=sql.105\).aspx](https://technet.microsoft.com/es-es/library/ms167305(v=sql.105).aspx))
- Haga clic en Aceptar para abrir el proyecto. El proyecto Tutorial se muestra en el Explorador de soluciones con un informe denominado Sales Orders.rdl.

Para convertir un origen de datos incrustado en compartido

- En el panel Datos de informe, haga clic con el botón secundario en el origen de datos AdventureWorks y seleccione Convertir a origen de datos compartidos. Se agrega el origen de datos AdventureWorks.rds al Explorador de soluciones.
- En el panel Datos de informe, haga clic con el botón secundario en el origen de datos AdventureWorks y seleccione Propiedades del origen de datos.
- En Nombre, escriba AdventureWorks_Ref.
- Haga clic en Aceptar.

Para reemplazar el conjunto de datos existente

- En el panel Datos de informe, haga clic con el botón secundario en el conjunto de datos AdventureWorksDataset y, a continuación, haga clic en Propiedades del conjunto de datos.

Nota

Si no ve el panel Datos de informe, en el menú Ver, haga clic en Datos de informe.

- En Origen de datos, compruebe que AdventureWorks_Ref está seleccionado.
- En Tipo de consulta, compruebe que está seleccionada la opción Texto.
- Haga clic en el botón Diseñador de consultas para abrir el Diseñador de consultas.
- Reemplace el texto del cuadro de texto por la consulta siguiente:

```

SELECT
    soh.OrderDate AS [Date],
    soh.SalesOrderNumber AS [Order],
    pps.Name AS Subcat, pp.Name as Product,
    SUM(sd.OrderQty) AS Qty,
    SUM(sd.LineTotal) AS LineTotal
FROM Sales.SalesPerson sp
INNER JOIN Sales.SalesOrderHeader AS soh
    ON sp.BusinessEntityID = soh.SalesPersonID
INNER JOIN Sales.SalesOrderDetail AS sd
    ON sd.SalesOrderID = soh.SalesOrderID
INNER JOIN Production.Product AS pp
    ON sd.ProductID = pp.ProductID
INNER JOIN Production.ProductSubcategory AS pps
    ON pp.ProductSubcategoryID = pps.ProductSubcategoryID
INNER JOIN Production.ProductCategory AS ppc
    ON ppc.ProductCategoryID = pps.ProductCategoryID
GROUP BY ppc.Name, soh.OrderDate, soh.SalesOrderNumber,
    pps.Name, pp.Name, soh.SalesPersonID
HAVING (ppc.Name = 'Clothing'
    AND (soh.OrderDate BETWEEN (@StartDate) AND (@EndDate)))

```

Esta consulta es la misma que la anterior, con la única excepción de que se ha agregado una condición con dos parámetros de límite:

AND (soh.OrderDate BETWEEN (@StartDate) AND (@EndDate))

- Haga clic en Ejecutar (!) en la barra de herramientas. Se abre el cuadro de diálogo Definir parámetros de consulta para solicitarle los valores de los parámetros.
- Proporcione dos valores para ver un conjunto de resultados filtrado:
 - En la columna Valor de parámetro, especifique un valor para @StartDate, por ejemplo, 1/31/2001.
 - En la columna Valor de parámetro, especifique un valor para @EndDate, por ejemplo, 1/31/2003.
- Haga clic en Aceptar.
- El conjunto de resultados muestra un conjunto de datos filtrado para los pedidos de los años 2001 y 2002.
- Haga clic en Aceptar dos veces. El panel Datos de informe se rellena con los campos de conjunto de datos. Observe también que, debajo del nodo Parámetros, se crean y aparecen automáticamente dos parámetros de informe, StartDate y EndDate.

Después de haber definido los parámetros de consulta para el informe, deberá cambiar el tipo de datos de los parámetros de informe para que coincida con el tipo de datos de los datos de origen. El valor predeterminado es Texto, que se asigna al tipo de datos String en la mayoría de los orígenes de datos. Si un origen de datos es numérico, booleano o de fecha y hora, deberá cambiar el tipo de datos del parámetro de informe.

Para cambiar el tipo de datos y los valores predeterminados de un parámetro de informe

- En el panel Datos de informe, expanda Parámetros y haga doble clic en StartDate. Se abrirá el cuadro de diálogo Propiedades de parámetro de informe.
- Compruebe que el nombre del parámetro es StartDate y que el mensaje de petición de datos es Start Date.
- En Tipo de datos, seleccione Date/Time.
- Haga clic en Aceptar.
- En el panel Datos de informe, haga doble clic en EndDate. Compruebe los valores para el nombre y para el mensaje de petición de datos.
- En Tipo de datos, seleccione Date/Time.
- Haga clic en Aceptar.
- Haga clic en Vista previa. Los parámetros StartDate y EndDate aparecen en la barra de herramientas de informe con un control de calendario. Los controles de calendario aparecen automáticamente cuando el tipo de datos del parámetro es Date/Time y no se ha definido ninguna lista de valores disponibles. Si define una lista de valores disponibles, en su lugar aparece una lista desplegable de valores.
- Proporcione dos valores de parámetro para ejecutar el informe:
 - En el cuadro de texto del parámetro StartDate, escriba la fecha 1/31/2001.
 - En el cuadro de texto del parámetro EndDate, escriba la fecha 1/31/2003.
- Haga clic en Ver informe. El informe muestra solo los datos comprendidos entre los valores de los parámetros de informe.

Una vez que haya creado los parámetros para el informe, podrá agregar valores predeterminados para dichos parámetros. Los parámetros predeterminados permiten que el informe se ejecute automáticamente; en caso contrario, será el usuario quien deba especificar los valores de los parámetros para ejecutar el informe.

Para establecer los valores predeterminados para los parámetros

- En la vista Diseño, en el panel Datos de informe, expanda Parámetros y haga doble clic en StartDate. Se abrirá el cuadro de diálogo Propiedades de parámetro de informe.
- Haga clic en Valores predeterminados.
- Seleccione la opción Especificar valores. Aparecerá el botón Agregar y una cuadrícula Valor vacía.
- Haga clic en Agregar. Se agrega una fila vacía a la cuadrícula.
- Haga clic en el cuadro de texto Valor y elimine el texto predeterminado (Null).
- Escriba 1/31/2001. Haga clic en Aceptar.
- En la vista Diseño, en el panel Datos de informe, expanda Parámetros y haga doble clic en EndDate.
- Haga clic en Valores predeterminados.
- Seleccione la opción Especificar valores.
- Haga clic en Agregar.
- Escriba 1/31/2003. Haga clic en Aceptar.
- Haga clic en Vista previa. El informe se ejecuta de forma inmediata porque hay valores predeterminados definidos para todos los parámetros.

Para agregar un nuevo campo a la consulta a fin de usarlo para filtrar

- Cambie a la vista Diseño.

- Haga clic con el botón secundario en el conjunto de datos AdventureWorksDataset y seleccione Propiedades del conjunto de datos. Abra el Diseñador de consultas y reemplace la consulta por esta consulta nueva:

```
SELECT
    soh.OrderDate AS [Date], DATENAME(weekday, soh.OrderDate) as Weekday,
    soh.SalesOrderNumber AS [Order],
    pps.Name AS Subcat, pp.Name as Product,
    SUM(sd.OrderQty) AS Qty,
    SUM(sd.LineTotal) AS LineTotal
FROM Sales.SalesPerson sp
INNER JOIN Sales.SalesOrderHeader AS soh
    ON sp.BusinessEntityID = soh.SalesPersonID
INNER JOIN Sales.SalesOrderDetail AS sd
    ON sd.SalesOrderID = soh.SalesOrderID
INNER JOIN Production.Product AS pp
    ON sd.ProductID = pp.ProductID
INNER JOIN Production.ProductSubcategory AS pps
    ON pp.ProductSubcategoryID = pps.ProductSubcategoryID
INNER JOIN Production.ProductCategory AS ppc
    ON ppc.ProductCategoryID = pps.ProductCategoryID
GROUP BY ppc.Name, soh.OrderDate, soh.SalesOrderNumber,
    pps.Name, pp.Name, soh.SalesPersonID
HAVING (ppc.Name = 'Clothing' AND (soh.OrderDate BETWEEN (@StartDate) AND
(@EndDate)))
```

En la consulta, se ha definido una columna calculada adicional para el día de la semana en que tuvo lugar una venta; para ello, se ha agregado el siguiente comando a la instrucción SELECT:

DATENAME (weekday, soh.OrderDate) as Weekday.

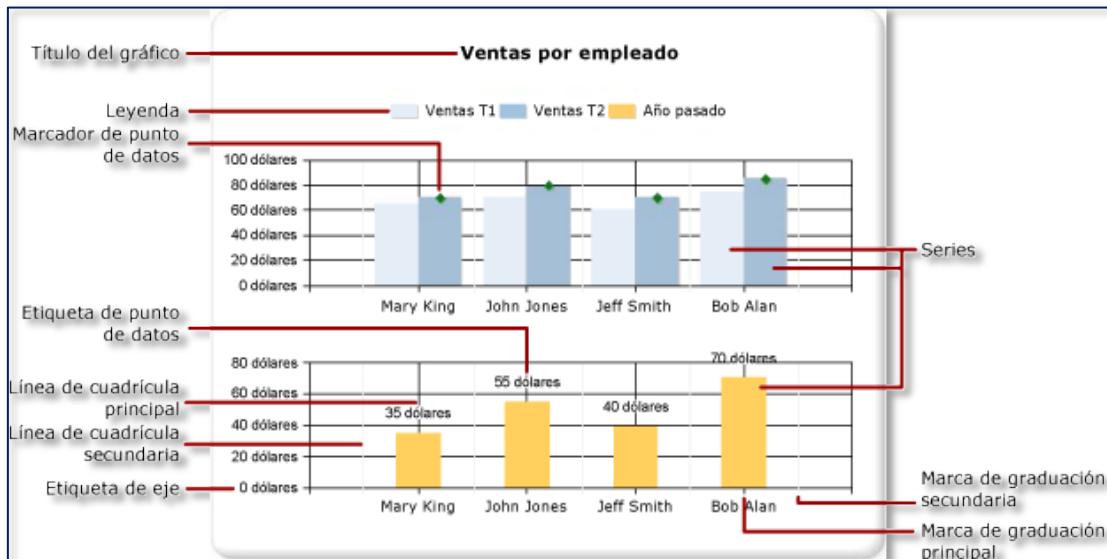
- Haga clic en Ejecutar (!). Se abre el cuadro de diálogo Definir parámetros de consulta.
- En la columna Valor de parámetro, especifique un valor para @StartDate, por ejemplo, 1/31/2001.
- En la columna Valor de parámetro, especifique un valor para @EndDate, por ejemplo 1/31/2003.
- Haga clic en Aceptar. Debe ver una nueva columna en el conjunto de resultados con la etiqueta Weekday.
- Haga clic en Aceptar dos veces. En el panel Datos del informe, compruebe que Weekday es un campo.

4.3.1.2 Uso de Gráficos y Medidores

Si desea resumir datos y presentarlos con un formato visual, use la región de datos de gráfico. Los gráficos le permiten visualizar de forma rápida grandes volúmenes de información agregada. Es importante que prepare cuidadosamente los datos y que los

comprenda antes de crear los gráficos; esto le ayudará a diseñarlos de manera rápida y eficaz.

En la ilustración siguiente se muestran muchos de los distintos elementos que se usan en un gráfico.



Puede publicar gráficos por separado de un informe como elementos de informe. Los elementos de informe son elementos de informe independientes que se almacenan en el servidor de informes y se pueden incluir en otros informes. Use el Generador de informes para examinar y seleccionar elementos de la Galería de elementos de informe para agregarlos a los informes. Use el Diseñador de informes o el Generador de informes para guardar los elementos de informe en la galería de elementos de informe.

Agregar un gráfico a un informe (Generador de informes y SSRS)

La manera más simple de agregar una región de datos de Gráfico a su informe es ejecutar el Asistente para nuevo gráfico. El asistente proporciona gráficos de columna, línea, circular, barra y área. Para éstos y otros tipos de gráfico, puede agregar también un gráfico manualmente.

Después de agregar una región de datos Gráfico a la superficie de diseño, puede arrastrar los campos de conjunto de datos de informe para los datos numéricos y no numéricos hasta el panel de Datos del gráfico en el gráfico. Haga clic en el gráfico para mostrar el panel Datos del gráfico con sus tres áreas: Grupos de series, Grupos de categorías y Valores

Para agregar un gráfico a un informe utilizando el Asistente para gráficos

- En la pestaña Insertar, haga clic en Gráfico y, a continuación, haga clic en Asistente para gráficos.
- Siga los pasos en el Asistente para Nuevo gráfico.
- En la pestaña Inicio, haga clic en Ejecutar para ver el informe representado.
- En la pestaña Ejecutar, haga clic en Diseño para seguir trabajando en el informe.

Para agregar un gráfico a un informe

- Cree un informe y defina un conjunto de datos. Para obtener más información, vea Agregar datos a un informe (Generador de informes y SSRS).
- En la pestaña Insertar, haga clic en Gráfico y, a continuación, haga clic en Insertar gráfico.
- Haga clic en la superficie de diseño en la que desea que se encuentre la esquina superior izquierda del gráfico y arrastre hasta donde desee que se encuentre la esquina inferior derecha del gráfico.
Aparece el cuadro de diálogo Seleccionar tipo de gráfico.
- Seleccione el tipo de gráfico que desea agregar. Haga clic en Aceptar.
- Haga clic en el gráfico para mostrar el panel Datos del gráfico.
- Agregue uno o más campos al área Valores. Esta información se representará en el eje de valores.
- Agregue un campo de agrupación al área Grupos de categorías. Al agregar este campo al área Grupos de categorías, se crea un campo de agrupación automáticamente. Cada grupo representa un punto de datos de la serie.
- Para resumir los datos por categoría, haga clic con el botón secundario en el campo de datos y, a continuación, haga clic en Propiedades de la serie. En el cuadro Categoría, seleccione el campo de categoría en la lista desplegable. Haga clic en Aceptar.
- En la pestaña Inicio, haga clic en Ejecutar para ver el informe representado.
- En la pestaña Ejecutar, haga clic en Diseño para seguir trabajando en el informe.

En los gráficos con ejes, como los gráficos de barras y de columnas, es posible que el eje de categorías no muestre todas las etiquetas de categoría.

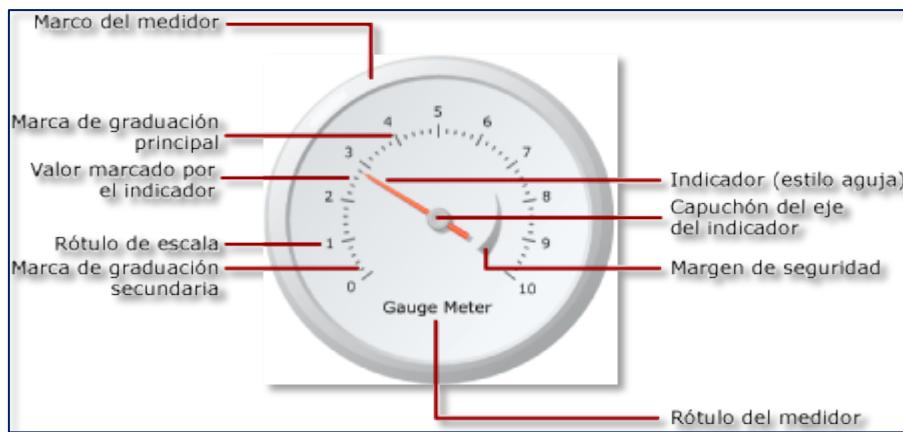
Medidores

La región de datos de medidor es una región de datos unidimensional que muestra un solo valor del conjunto de datos. Los medidores siempre se sitúan dentro de un panel de medidores, en el que pueden agregarse medidores secundarios o adyacentes. En el mismo panel de medidores, puede crear varios medidores que comparten funciones comunes como el filtrado, la agrupación o la ordenación.

En un informe, puede usar los medidores para realizar muchas tareas:

- Mostrar indicadores clave de rendimiento (KPI) en un único medidor radial o lineal.
- Situar un medidor dentro de una tabla o matriz para ilustrar los valores de cada celda.
- Usar varios medidores en un solo panel de medidores para comparar los datos entre los campos.

Existen dos tipos de medidores: radial y lineal. En la ilustración siguiente, se muestran los elementos básicos de un medidor radial del panel de medidores.



Tipos de medidor

Reporting Services proporciona dos tipos de medidores: radial y lineal. Normalmente, el medidor radial se utiliza cuando los datos se quieren expresar como una velocidad. El medidor lineal se usa para expresar los datos como una temperatura o valor de escala.

Las diferencias principales entre los dos tipos son la forma del medidor y los punteros de medidor disponibles. Los medidores radiales son circulares, o partes de un círculo, y se parecen a los indicadores de velocidad. Los punteros del medidor suelen ser agujas, pero pueden ser marcadores o barras.

Los medidores lineales son rectangulares, orientados horizontal o verticalmente, y se parecen a las reglas. Los punteros del medidor suelen ser termómetros, pero pueden ser marcadores o barras. Debido a su forma, este tipo de medidor resulta útil para su integración en las regiones de datos de la tabla o matriz para mostrar los datos de progreso.

Aparte de estas diferencias, los dos tipos de medidor son intercambiables. Sin embargo, si tiene que utilizar un medidor simple en su informe, podría interesarle utilizar un indicador en lugar de un medidor.

En las siguientes ilustraciones se muestran los medidores radiales y lineales. El medidor radial es redondo y usa el puntero de aguja. El medidor lineal es horizontal y usa el puntero de termómetro.

Medidor radial



Las opciones de medidor radial: Radial, Radial con medidor mini, Dos escalas, 90 grados Noreste, 90 grados Noroeste, 90 grados Suroeste, 90 grados Sureste, 180 grados Norte, 180 grados Sur, 180 grados Oeste, 180 grados Este e Indicador.

Medidor lineal



Las opciones de medidor lineal: Horizontal, Vertical, Varios punteros de barra, Dos escalas, Rango de tres colores, Logarítmico, Termómetro, Termómetro fahrenheit/celsius y Gráfico de viñetas.

Agregar datos a un medidor

Una vez que haya agregado un medidor a la superficie de diseño, arrastre un campo de conjunto de datos hasta el panel de datos del medidor. De forma predeterminada, el medidor agrega los valores del campo en un valor que se muestra en el medidor. Ese valor se adjunta al puntero utilizando la propiedad Value. Dependiendo del tipo de datos del campo, el medidor utiliza el agregado SUM o COUNT. Al usar datos numéricos, indicados para agregar, el medidor usa la función SUM. De lo contrario, usa el agregado COUNT. El valor del puntero puede utilizar otro agregado distinto o ninguno.

Puede agregar agrupación al medidor para ver grupos individuales o filas individuales en el mismo. Cuando se aplican la agrupación y el filtrado, el medidor usa el valor del puntero para mostrar el último grupo o la última fila del conjunto de datos devuelto.

Puede agregar varios valores a un mismo medidor agregando otro puntero. Este puntero puede pertenecer a la misma escala o puede agregar otra escala y, a continuación, asociar el puntero a la misma.

A diferencia de los tipos de gráficos del cuadro de diálogo Seleccionar tipo de gráfico, los tipos de medidores del cuadro de diálogo Seleccionar tipo de medidor se crean mediante una combinación de propiedades de medidor. Por tanto, no se puede cambiar de tipo de medidor de la misma manera que se cambia de tipo de gráfico. Para cambiar el tipo de medidor, primero debe quitar el medidor y, a continuación, debe volver a agregarlo a la superficie de diseño. Un medidor tiene al menos una escala y un puntero.

Si desea tener varias escalas, haga clic con el botón secundario en el medidor y, a continuación, seleccione Agregar escala. De forma predeterminada, se crea una escala más pequeña que se sitúa dentro de la primera escala. La escala muestra etiquetas y marcas de graduación. Hay dos conjuntos de marcas de graduación: secundarias y principales.

Si desea tener varios punteros, haga clic con el botón secundario en el medidor y, a continuación, seleccione Agregar puntero. Se crea otro puntero en la misma escala, pero si tiene varias escalas, puede asociar un puntero a cualquiera de ellas en el medidor.

Consideraciones al agregar datos al medidor

Al igual que todas las demás regiones de datos, la región de datos de medidor solo se puede enlazar a un conjunto de datos. Si tiene varios conjuntos de datos, considere la posibilidad de usar JOIN o UNION para crear un conjunto de datos o use medidores independientes para cada conjunto de datos.

Los tipos de datos numéricos se agregan con la función SUM. Los tipos de datos no numéricos se agregan con la función COUNT, que cuenta el número de instancias para un valor o campo determinado perteneciente al conjunto de datos o al grupo.

Una vez agregados los datos, al hacer clic con el botón secundario en el puntero, obtendrá las opciones Borrar valor de puntero y Eliminar puntero. La opción Borrar valor de puntero quita el campo adjuntado al medidor, pero el puntero seguirá apareciendo en el medidor. La opción Eliminar puntero quita el campo del medidor y hace que el puntero deje de verse. Si vuelve a agregar un campo al medidor, reaparece el puntero predeterminado. Después de haber agregado el campo al medidor, debe establecer los valores máximo y mínimo en la escala correspondiente para dar contexto al valor en el medidor. También puede establecer los valores mínimo y máximo en un intervalo, que muestra un área crítica en la escala. El medidor no establecerá automáticamente los valores mínimo o máximo en la escala o el intervalo porque no puede determinar cómo se debería percibir el valor.

Métodos para agregar datos a un medidor

Después de definir un conjunto de datos para el informe, puede agregar un campo de datos al medidor con uno de los métodos siguientes:

- Arrastre un campo desde el conjunto de datos al panel de datos. Haga clic en el medidor y arrastre un campo hasta él. Puede abrir panel de datos haciendo clic en el medidor o arrastrando un campo por él. Si no había ningún puntero en el medidor, se agrega uno y se enlaza al campo agregado.
- Muestre el panel de datos y señale al marcador de posición de campo. Haga clic en la flecha hacia abajo que hay al lado del marcador de posición de campo y seleccione el campo que desea usar. Si ya hay un campo seleccionado, haga clic en la flecha hacia abajo y, a continuación, seleccione otro campo.

Nota

Este método no es aplicable cuando no hay ningún puntero en el medidor o el informe contiene más de un conjunto de datos y el panel de medidores no está asociado a un conjunto de datos.

- Haga clic con el botón secundario en el puntero del medidor y seleccione Propiedades de puntero. Para Valor, seleccione un campo en la lista desplegable o defina una expresión de campo haciendo clic en el botón Expresión (fx).

Agregar campos en un solo valor

De forma predeterminada, cuando se agrega un campo a un medidor, Reporting Services calcula a un agregado para dicho campo. Los tipos de datos numéricos se agregan con la función SUM. Los tipos de datos no numéricos se agregan con la función COUNT, que cuenta el número de instancias para un valor o campo determinado perteneciente al conjunto de datos o al grupo. Si el tipo de datos del campo de valores es String, el medidor no puede mostrar un valor numérico, aun cuando haya números en los campos. En su lugar, el medidor agrega los campos con cadenas mediante la función COUNT. Para evitar este comportamiento, asegúrese de que los campos que usa tienen tipos de datos numéricos, en lugar de cadenas que

contienen números con formato. Puede usar una expresión de Visual Basic para convertir los valores de cadena en un tipo de datos numérico usando la constante CDbl o CInt. Por ejemplo, la expresión siguiente convierte un campo de cadena llamado MyField en valores numéricos.

```
=Sum (CDbl (Fields!MyField.Value))
```

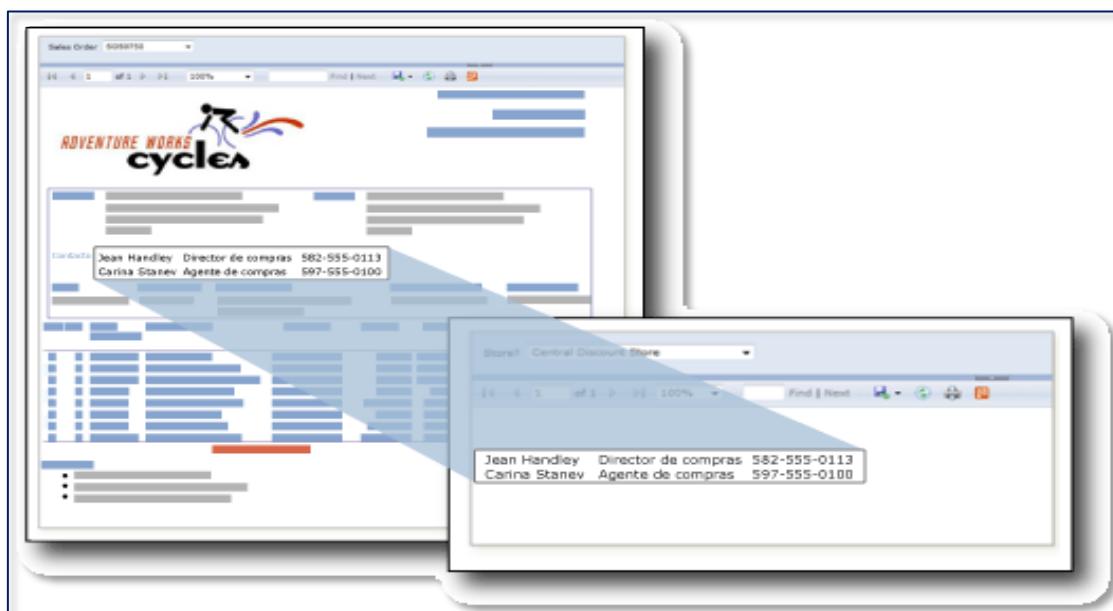
Definir un grupo en un medidor

Después de haber agregado un campo al medidor, puede agregar un grupo de datos. El medidor difiere de todas las demás regiones de datos de Reporting Services, que pueden mostrar varios grupos en una región de datos. El hecho de agregar un grupo definiendo una expresión de grupo en el medidor, equivale a agregar un grupo de filas en la región de datos Tablix. Sin embargo, cuando se agrega el grupo, solo se muestra el valor del último grupo como valor del puntero en el medidor. Por ejemplo, si agrega una expresión de agrupación según el año, el puntero señalará al valor que representa el valor de ventas agregado para el último año del conjunto de datos.

Es posible que desee agregar un grupo al medidor si, por ejemplo, está mostrando varios medidores en una tabla o una lista y desea mostrar datos agregados por grupo.

4.3.1.3 Creación de Subinformes

Un subinforme es un elemento de informe que muestra otro informe dentro del cuerpo del informe principal. Como concepto, un subinforme de un informe es como un marco en una página web. Se utiliza para incrustar un informe dentro de un informe. Cualquier informe puede utilizarse como subinforme. El informe que se muestra como el subinforme se almacena en un servidor de informes, normalmente en la misma carpeta que el informe primario. Es posible diseñar el informe primario para que pase sus parámetros al subinforme. Este tipo de informe puede repetirse dentro de las regiones de datos mediante un parámetro que filtre los datos de cada instancia del subinforme.



4.3.1.4 Navegación entre reportes

Un mapa del documento proporciona un conjunto de vínculos de navegación a los elementos de informe de un informe representado. Cuando se ve un informe que incluya un mapa del documento, aparece un panel lateral separado junto al informe. Un usuario puede hacer clic en los vínculos del mapa del documento para saltar a la página del informe que muestra el elemento. Las secciones y los grupos del informe se organizan en una jerarquía de vínculos. Cada vez que se hace clic en un elemento del mapa del documento, se actualiza el informe y se muestra el área del mismo correspondiente a dicho elemento en el mapa del documento.

Para agregar vínculos al mapa del documento, se establece la propiedad DocumentMapLabel del elemento de informe en un texto o en una expresión que se evalúa como el texto que se desea mostrar en el mapa del documento. También puede agregar los valores únicos para un grupo de tablas o de matrices al mapa del documento. Por ejemplo, para un grupo basado en colores, cada color único es un vínculo a la página del informe que muestra la instancia de grupo para ese color.

También puede crear una dirección URL a un informe que invalide la presentación del mapa del documento, lo que le permite ejecutar el informe sin mostrar el mapa del documento y, a continuación, hacer clic en el botón Mostrar u ocultar mapa de documento de la barra de herramientas del visor de informes para alternar la presentación.

4.3.2 Publicación de Reportes Reportes

Después de diseñar y probar un informe o un conjunto de informes, puede usar las características de implementación integradas en SQL Server Data Tools (SSDT) para publicar los informes en un servidor de informes.

Puede publicar informes individuales o un proyecto del servidor de informes. Publicar un proyecto del servidor de informes es la manera más fácil de publicar varios informes.

En SQL Server Data Tools (SSDT) se usa el término implementación, en lugar del término publicación. Las dos condiciones son intercambiables. Para poder publicar un informe, debe tener permiso para hacerlo. El permiso se determina a través de la seguridad basada en roles que define el administrador del servidor de informes.

Normalmente, las operaciones de publicación se conceden a través del rol Publisher. SQL Server Data Tools (SSDT) proporciona las configuraciones de proyecto para administrar la publicación de informes.

La configuración especifica la ubicación del servidor de informes, la versión de SQL Server Reporting Services instalada en el servidor de informes, si los orígenes de datos publicados en el servidor de informes se sobrescriben y otras opciones.

Además de utilizar las configuraciones que SQL Server Data Tools (SSDT) proporciona, puede crear otras adicionales.