

Implementación de DataWarehouse

**Nombre del Caso: “Data Warehousing y OLAP
para la Industria de Comidas Rápidas”**

Sistemas de Gestión 2 – Abraham, Botta, Fratte

‘11



Contenido

DESCRIPCIÓN DEL CASO	2
Introducción	2
Áreas Temáticas	2
Estructura de Datos	3
Requisitos Particulares	4
Índices a Resolver	4
SOLUCIÓN PROPUESTA	5
Diseño del Datawarehouse (<i>Vista de Cubos OLAP</i>)	5
Solución Propuesta (<i>Vista de la herramienta – OLAP DSV</i>)	7
Esquema	10
Dimensiones	10
Herramientas Utilizadas	10
Proceso de ETL (Extracción, Transformación y Carga)	11
1. Generación de Datos	12
2. Almacén Intermedio	12
3. Migración de Datos a SQL Server	13
4. Creación de la vista	16
Creación de Cubos	25
Creación de Dimensiones	27
Creación de Medidas	33
Navegación del Cubo	35
Presentación al Usuario final	36
Análisis de las ventas	36
Análisis de las compras	40
Ventas VS Compras	41
Encuestas de Motivación	42
ANEXO	44
Código de la Generación de Datos	44



DESCRIPCIÓN DEL CASO

Introducción

La industria de comida rápida es altamente competitiva, un cambio muy pequeño en las operaciones pueden tener un impacto significativo en la línea de operaciones. Por esta razón, el acceso rápido a la información completa para la presentación de informes estándar y bajo demanda, es esencial.

DecisionSoft ha diseñado e implementado un Data Warehouse para hacer frente a esta exigencia de Springwood Corporation, una franquicia de comida rápida que opera aproximadamente 20 Taco TABASCO y restaurantes Kentucky Fried Chicken alrededor del país. El Springwood Data Warehouse proporciona ahora soporte de decisiones estratégicas y tácticas para todos los niveles de gestión dentro de Springwood.

El almacén de datos se implementa en Microsoft SQL Server 2000, e incorpora los datos de dos fuentes principales:

- Todos los días información de ventas es suministrada de forma automática por el sistema de facturación.
- Información contable desde la base de datos de contabilidad Dynamics (Microsoft Great Plains).

Estos datos multidimensionales, se actualizan automáticamente de forma periódica (o bajo demanda, si es necesario) y se mantienen históricamente durante varios años con fines comparativos.

A efectos de información y análisis, los datos en el almacén se transforman en cubos OLAP. Los cubos se acceden a través de Excel utilizando MD BusinessQuery.

Los datos pueden ser analizados por tienda, por empresa, por área y zona, por año, trimestre y el mes, y por la marca y el concepto.

Áreas Temáticas

1. **Datos de pérdidas y ganancias y flujo de caja para Springwood.** Las cantidades se pueden ver durante un período, trimestre a la fecha, de año a la fecha, y se puede comparar con cualquiera de los dos presupuestos, en comparación con el mismo período del año anterior, o como porcentaje de las ventas.
2. **Mezcla de Ventas:** Contiene las ventas diarias de todos los elementos del menú en todas las tiendas. Además de los parámetros de análisis estándar, estos datos también puede ser analizado por marca, por tipo de posición o elemento de menú, por año calendario, mes y semana, y por el nivel de precios. Este cubo se puede utilizar para calcular los importes de ventas, los costos y la variación de precio de lista. Cubos de Ventas (partes del día). Contiene los importes de ventas y cuenta en intervalos de 15 minutos. Además de los parámetros de análisis estándar, los



datos de este cubo también se pueden analizar por año calendario, mes y semana, y por ocho horas, cuatro horas, dos horas, una hora y 15 minutos de intervalo, o específicos comida (por ejemplo, almuerzo, cena, desayuno, entre comidas, etc.)

3. **Resumen de Ventas Diarias.** Contiene resumen diario de ventas de cada tienda. Además de los parámetros de análisis estándar, estos datos también pueden ser dimensionado por un indicador comparable, por año calendario, mes y semana, y por el nivel de precios. Las ventas brutas, las ventas gravadas, las ventas no gravadas. Muchas cantidades se pueden ver, opcionalmente, como las variaciones, como un porcentaje de las ventas, o se resumen en la semana a la fecha, período, mes.
4. **Reportes de Costos de Ventas.** Muestra la cantidad y el costo promedio de artículos, considerando como unidad mínima de análisis, el ítem de menú. Debe realizarse un análisis geográfico de costos, e incluyendo la estructura corporativa.
5. **Análisis de Motivación y satisfacción de empleados,** midiendo respuestas positivas de encuestas de satisfacción. Deberá organizarse por tienda, zona, categoría de empleados, y analizar como varía el hecho en los distintos turnos.

Estructura de Datos

La estructura de datos implementada para Springwood les permite mantener varias estructuras organizativas distintas con el fin de representar adecuadamente a cada tienda en un

- La estructura corporativa, es decir, la filial de que pertenecen,
- La estructura de operaciones, es decir, la zona o área y
- El concepto de estructura, es decir, KFC, la TB (Taco TABASCO), Burger King, etc.

El almacén de datos Springwood y la consiguiente estructura de cubos OLAP puede navegarse a lo largo de cualquiera de estas jerarquías corporativas - es decir, por la compañía operadora, según la zona o zonas, o por marca o concepto. Esto permite comparaciones entre los conceptos, por ejemplo, o de todas las tiendas dentro de un concepto. Del mismo modo, es fácil hacer comparaciones de zona a zona, o para ver el desempeño de todas las tiendas dentro de un área.

El almacén de datos tiene una dimensión de tiempo. Este calendario se ha construido en el almacén, lo que permite una fácil comparación de cualquier período del ejercicio anterior o el mismo período de un año antes. En lugar de comparar a nivel período, las comparaciones y las tendencias se pueden hacer a nivel trimestral o anual. (Excepto para los casos de decisiones de más bajo nivel, por ejemplo, comparar semana a semana e incluso día a día).

En caso de Beneficios y Pérdidas se compara cada período durante los últimos cinco años, hasta el nivel de cuenta. Esto hace que sea fácil para Springwood para evaluar



las tendencias en cualquier categoría de gasto, comparando tienda a tienda, período a período, una zona a zona, o concepto-concepto.

Los cubos de Ventas Diarias se actualizan durante la noche con la información capturada por las cajas registradoras en cada tienda. Esto permite a los gerentes para evaluar y comparar las tendencias en la rapidez del servicio, el uso de mano de obra, las compras de alimentos de los empleados, etc., por el almacén, zona, área, concepto, etc. Debido a las ventas se registran en intervalos de 60 minutos, llamadas Partes del día, los administradores pueden usar esto para encontrar patrones de ventas extraños, posiblemente indicativos de robo de los empleados, durante las horas de la medianoche.

Requisitos Particulares

Las ventas diarias se analizan en el segundo Nivel de la organización y podría darse a cada gerente de Tienda la autorización para el análisis de sus propios datos.

Índices a Resolver

1. Mostrar Análisis de costos, ya sea histórico y además geográfico.
2. Analizar históricamente las ventas, podría desagregarse por Estructura Corporativa (para determinar si hay mayores o menores incidencias de tiendas)
3. Ranking de Ventas por Estructura de Productos de manera de estudiar preferencias.
4. Análisis de Tendencia para pérdidas y ganancias, Ranking de gastos por estructura corporativa.
5. Análisis de Importes de Ventas por intervalos de tiempo, para Gerentes de segundo nivel de la organización.
6. Reporte de cifras de Ventas gravadas, no gravadas, etc.
7. Tabla dinámica de Mix de Ventas, detallando hasta el Ítem de Menú, posición de menú y sus costos.
8. Relacionar en una tabla dinámica, Tiendas (y correspondiente estructura corporativa) con las estructura de productos vendidos.
9. Comparación entre distintos periodos de pérdidas y ganancias según las cuentas contables. Tabla y gráfico.
10. Total de ventas realizadas a Empleados. (horarios y productos).
11. Análisis geográfico de ventas mensuales, para estudiar posibilidades de abrir nuevas sucursales
12. Rentabilidad Económica de Productos del último mes, comparando contra un valor objetivo.



SOLUCIÓN PROPUESTA

Diseño del Datawarehouse (*Vista de Cubos OLAP*)

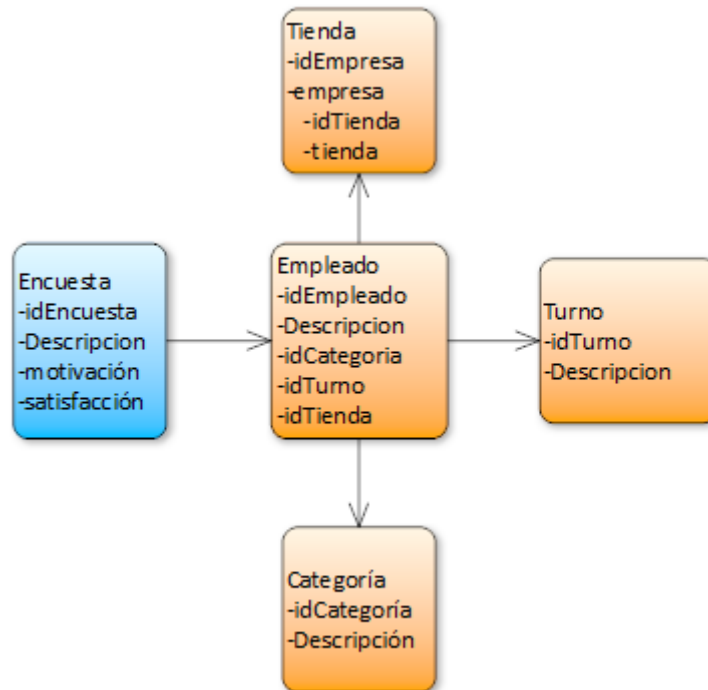


Ilustración 1 - Encuestas

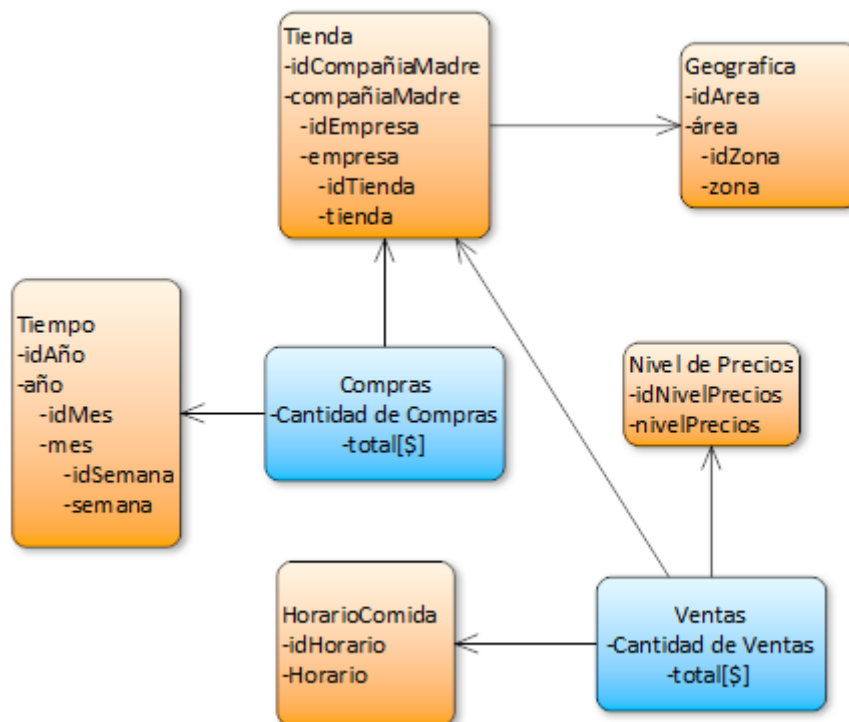


Ilustración 2 - Ingresos vs. Egresos

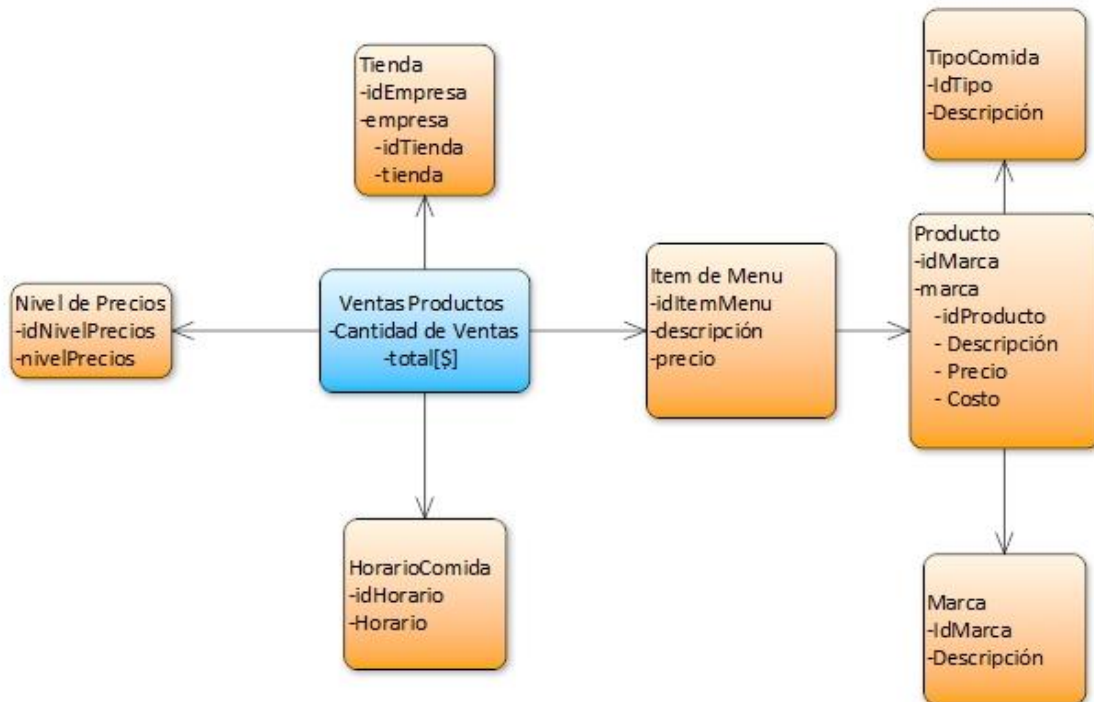


Ilustración 3 – Ventas Productos

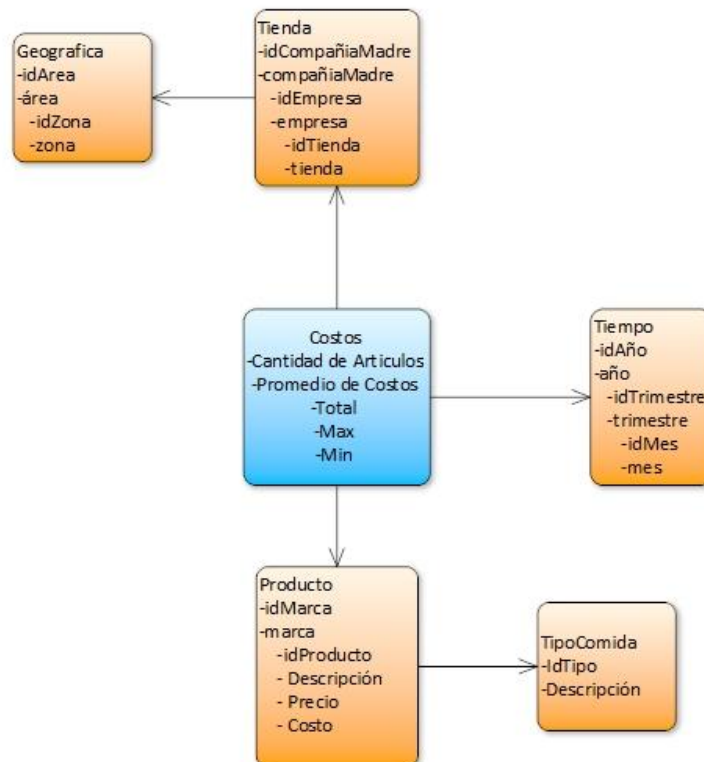


Ilustración 4 - Costos

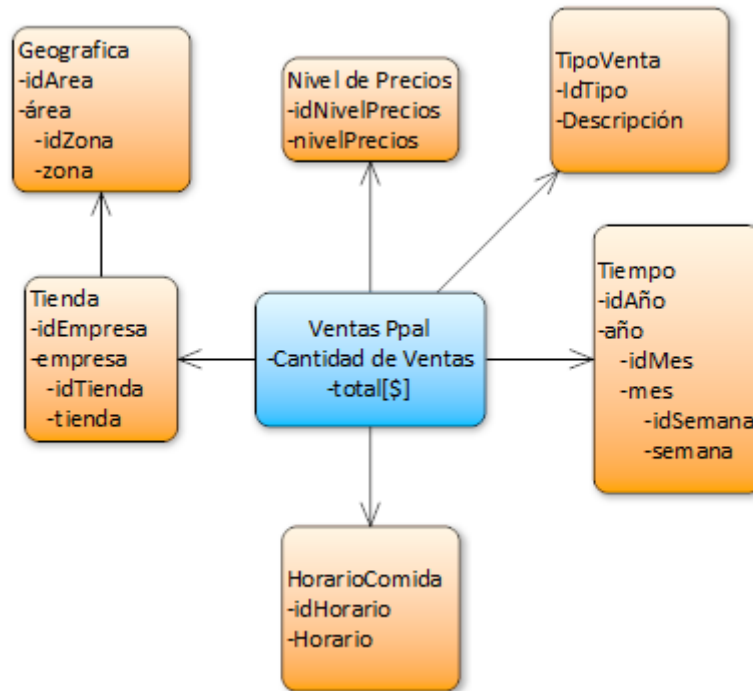


Ilustración 5 - Ventas

Solución Propuesta (Vista de la herramienta – OLAP DSV)

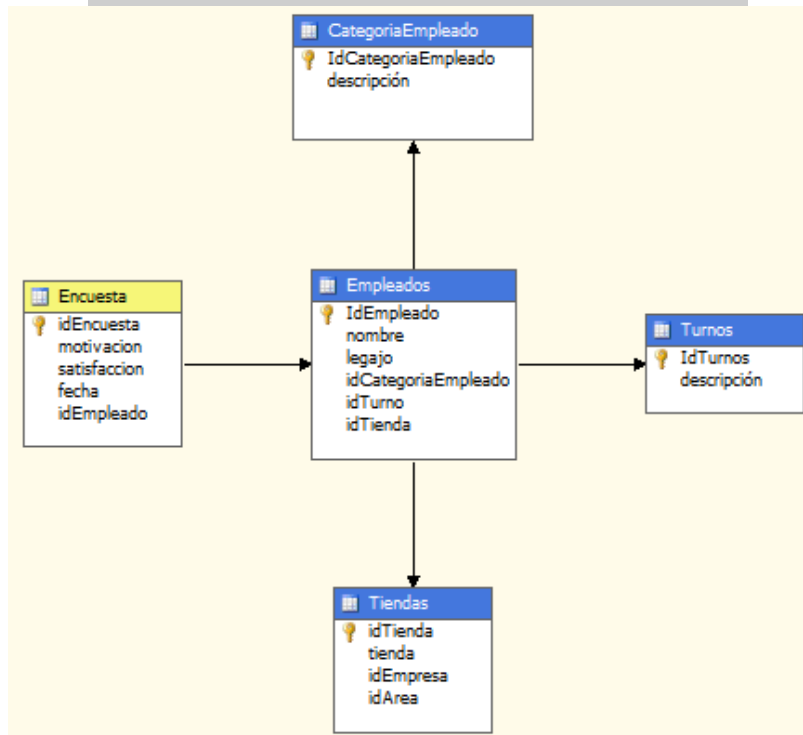


Ilustración 6 - Encuestas

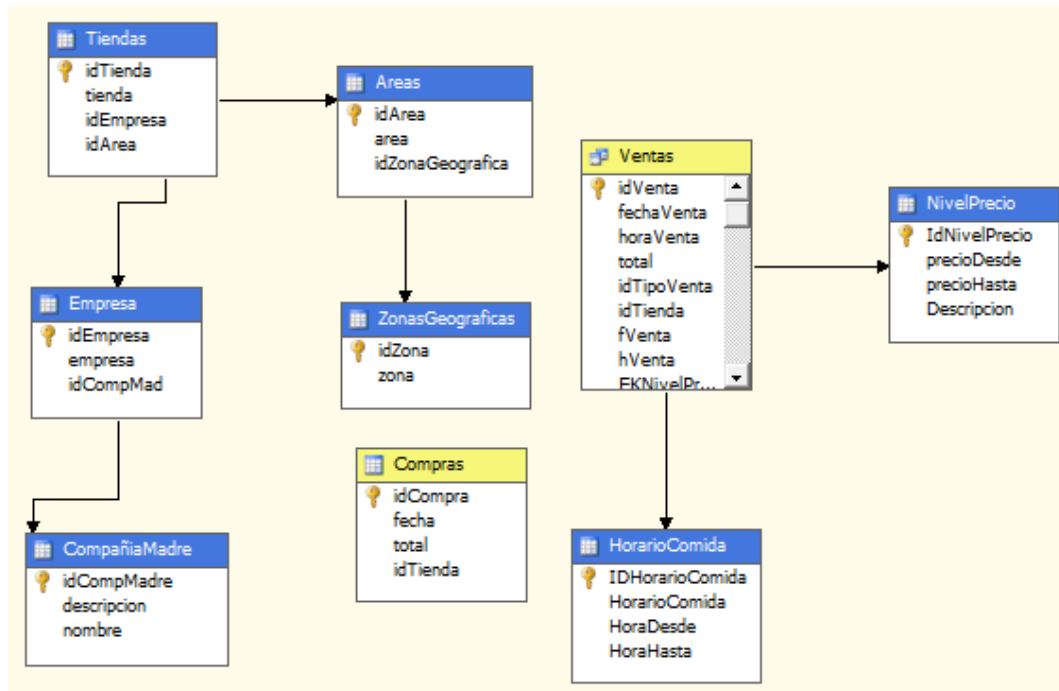


Ilustración 7 - Ingresos vs. Egresos

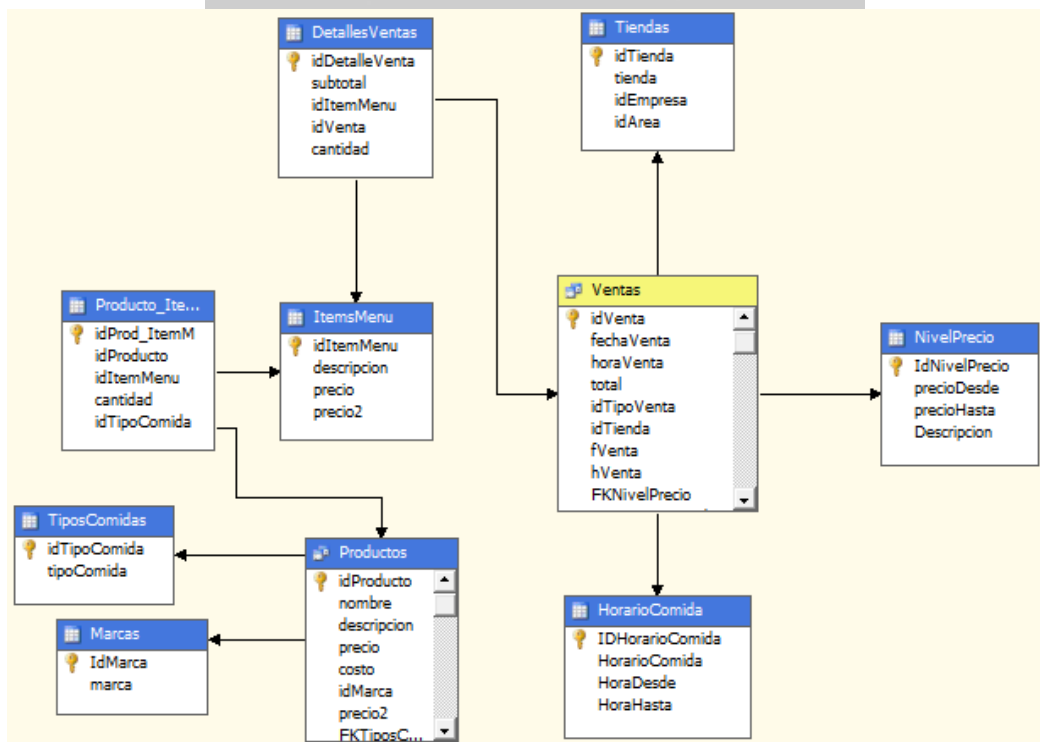


Ilustración 8 - Ventas Productos

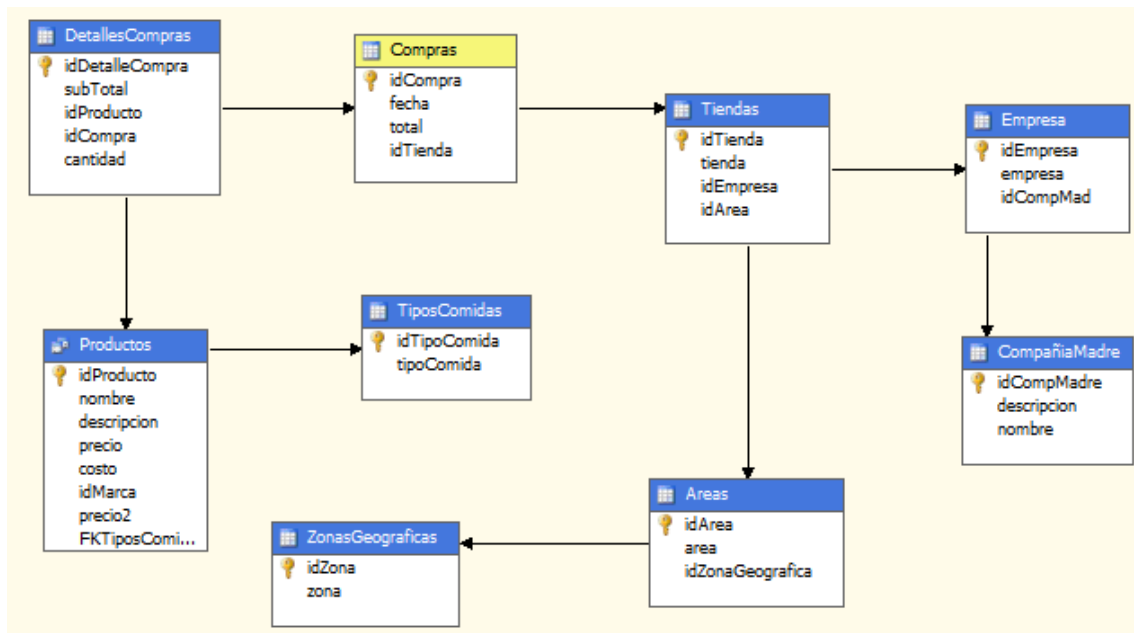


Ilustración 9 - Compras

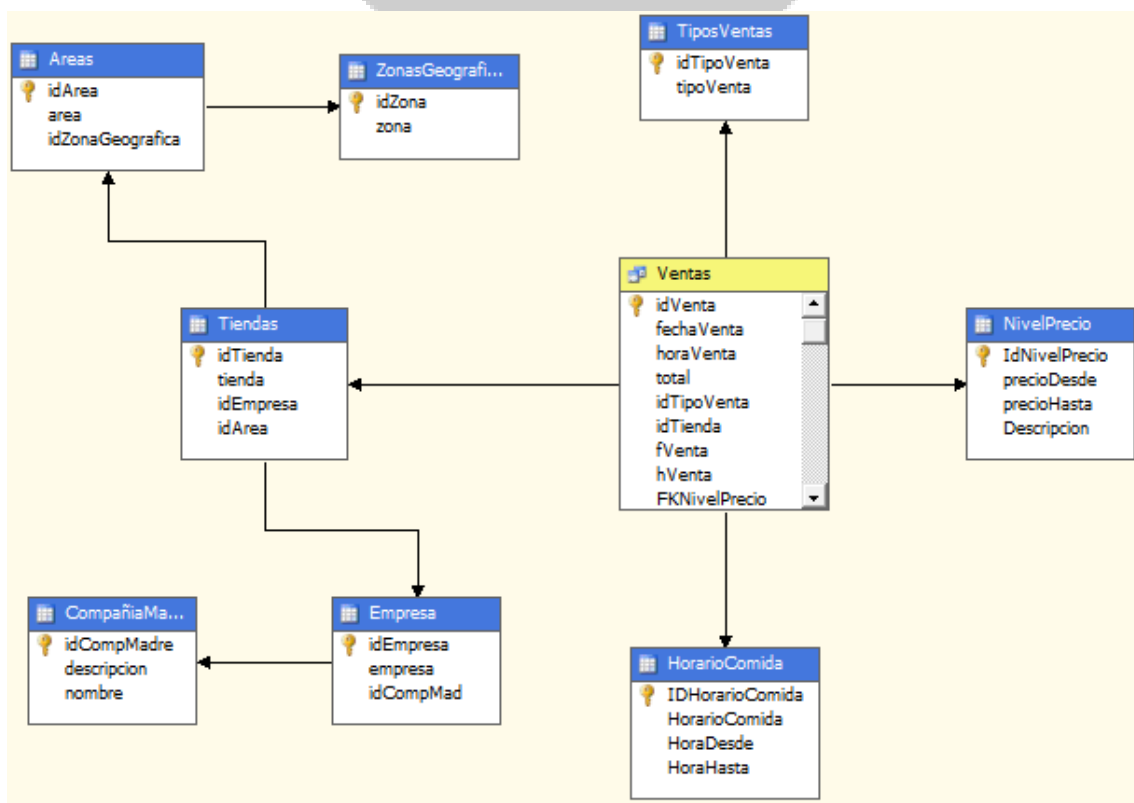


Ilustración 10 - Ventas



Esquema

Como puede observarse en los diagramas de la sección anterior, se ha optado por un esquema de Datawarehouse del tipo **Copo de Nieve**.

La implementación de este tipo de esquema fue posible debido al tamaño del modelo y datos en cuestión. Al no ser un volumen grande de datos, la performance de este tipo de esquema es aceptable.

Dimensiones

Las dimensiones definidas son:

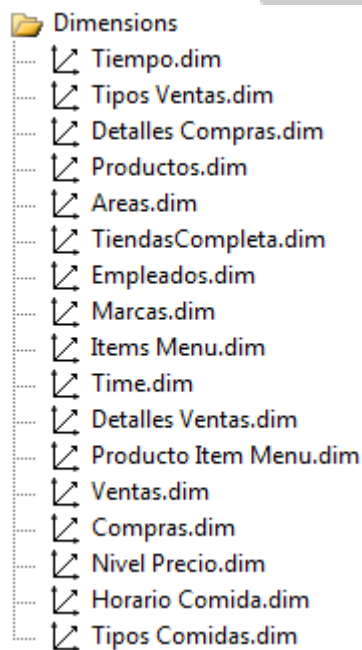


Ilustración 11 - Dimensiones

Herramientas Utilizadas

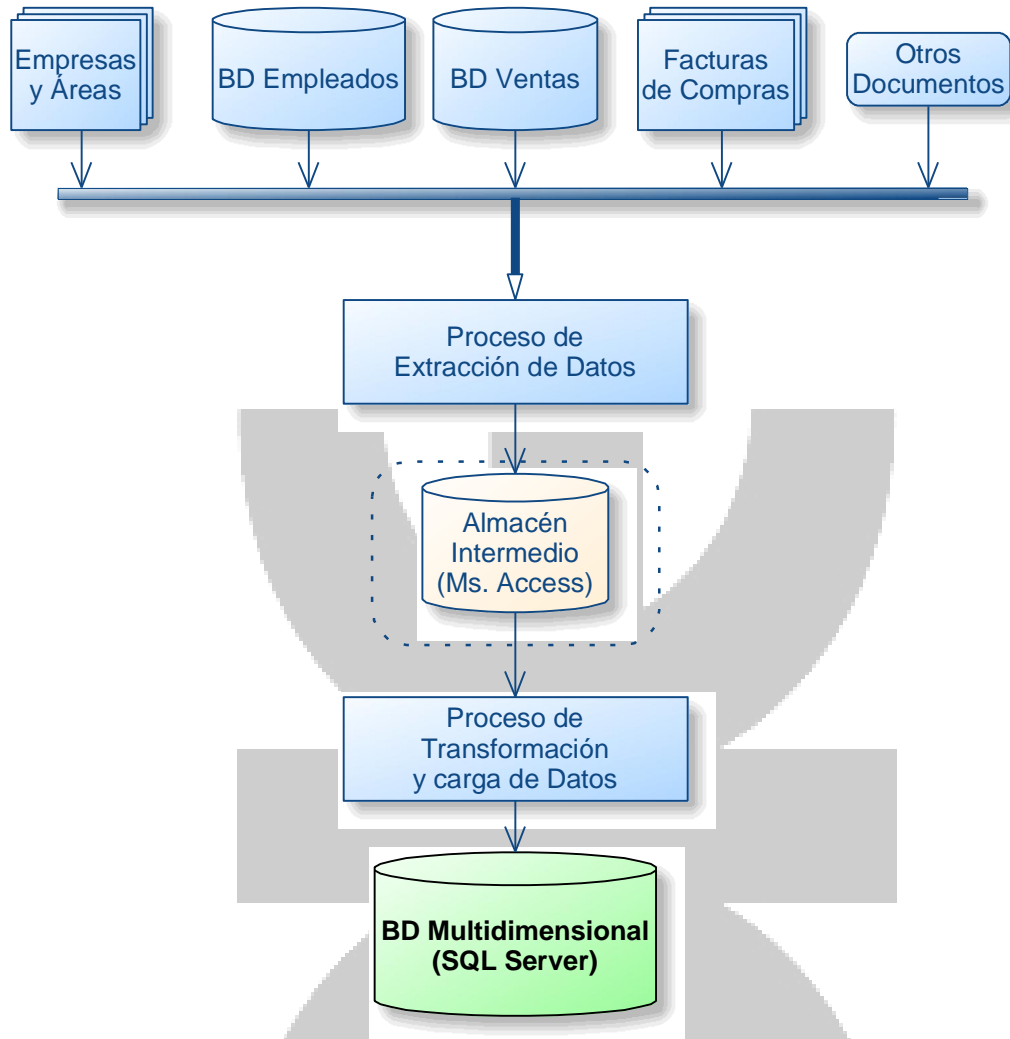
Herramienta	Propósito
RAD Studio Delphi 2007	Con este IDE se realizó una aplicación para generar los datos aleatorios sobre una Base de Datos en Microsoft Access.



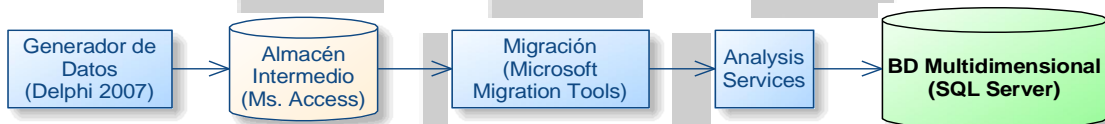
Microsoft Access 2010	Utilizado como motor de base de datos genérico. Se optó por este motor ya que la mayoría de los motores de base de datos del mercado permiten importar/exportar desde/hacia Access.
Microsoft SQL Server Migration Assistant for Access v4.2	Para migrar una base de datos Access a SQL Server
Microsoft SQL Server 2008 R2 Developers Edition: <ul style="list-style-type: none">• Analysis Services• SQL Server Business Intelligence Development Studio• SQL Server Management Studio	Creación de cubos, índices y reportes.

Proceso de ETL (Extracción, Transformación y Carga)

ETL es el proceso que organiza el flujo de los datos entre diferentes sistemas en una organización y aporta los métodos y herramientas necesarias para mover datos desde múltiples fuentes a un *almacén* de datos, reformatearlos, limpiarlos y cargarlos en otra base de datos.



Para este caso en particular, se crearon datos aleatorios en una BD Relacional en Microsoft Access utilizando una aplicación desarrollada en Delphi 2007, y luego se migraron a SQL Server, para poder crear los cubos utilizando Analysis Services.



1. Generación de Datos

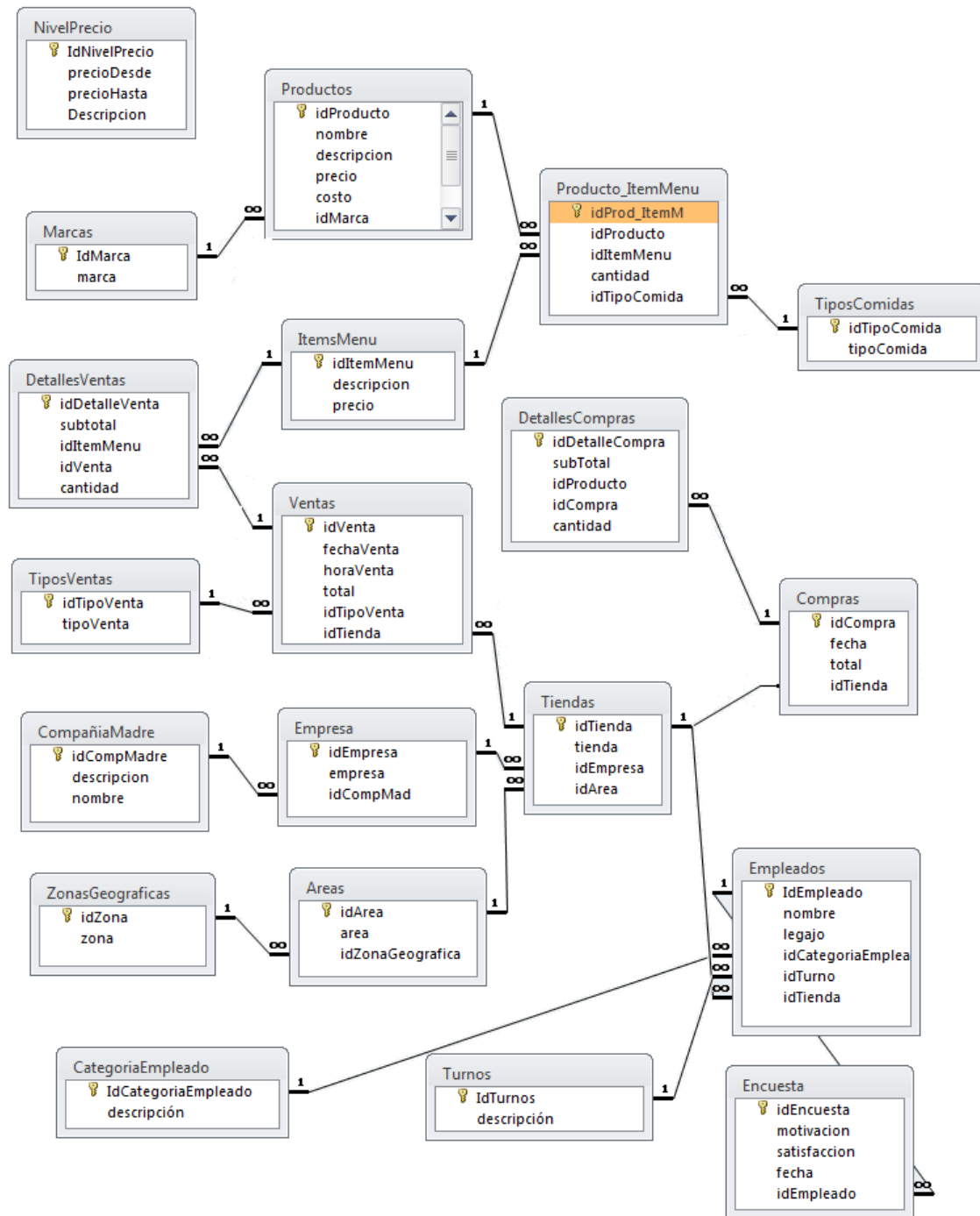
El código para generar los datos puede verse en el Anexo Adjunto al final del presente informe. En el mismo se muestran las funciones y porciones de código más representativas del generador. (Nota: El generador no tiene interfaz gráfica)

2. Almacén Intermedio

El diseño de la Base de Datos intermedia, realizado en Access, puede apreciarse en la



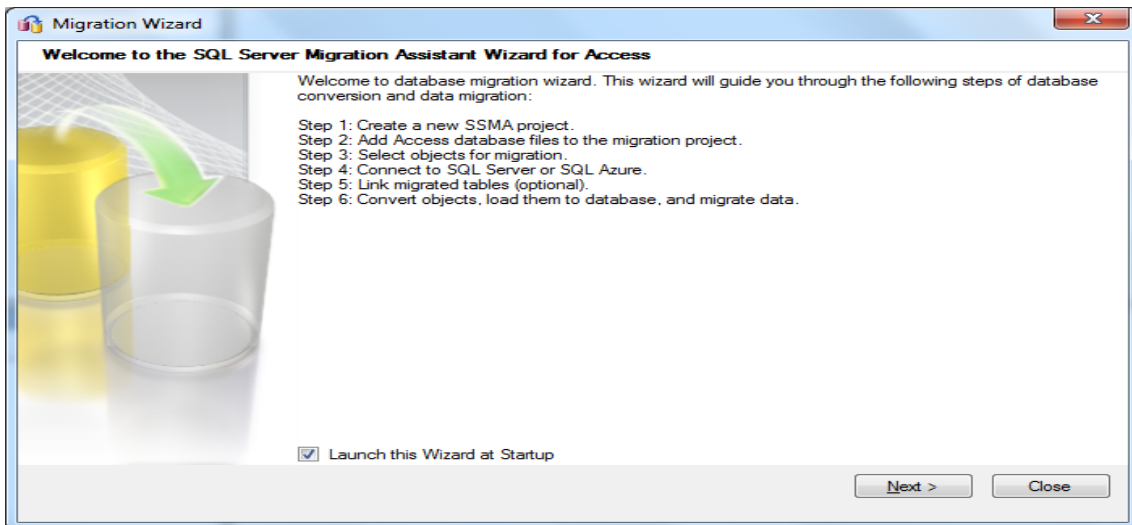
figura siguiente:



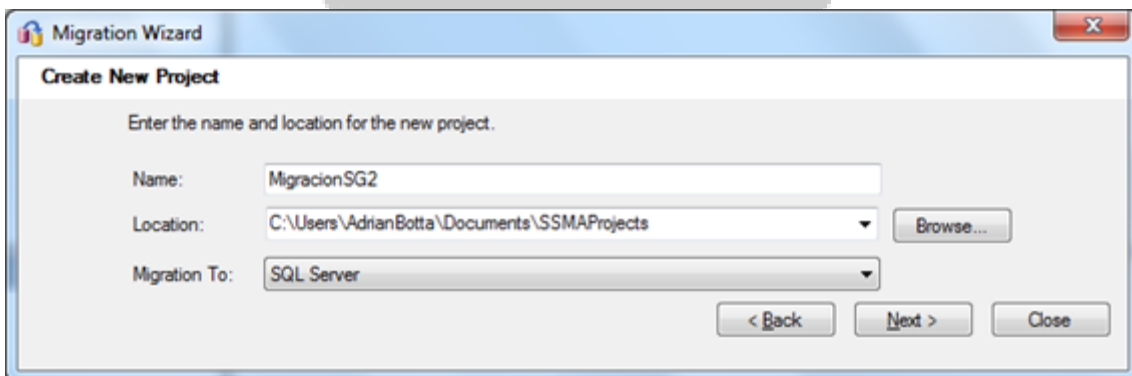
3. Migración de Datos a SQL Server

Para este paso, utilizaremos la herramienta Microsoft SQL Server Migration Assistant for Access v4.2, que permite migrar la base de datos relacional de Access a SQL server.

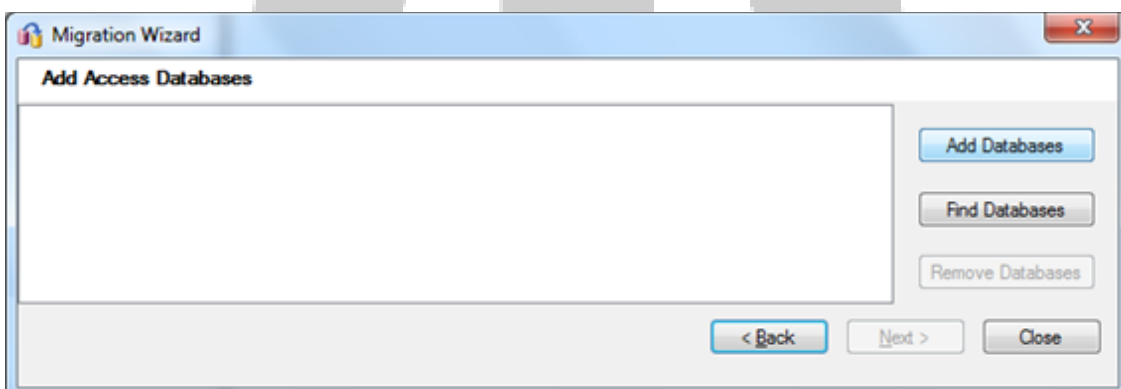
Primero, abrimos el programa, y automáticamente nos abrirá un asistente como se muestra a continuación.



Hacemos clic en “Next”, y configuramos el nombre y ubicación del proyecto de migración. En nuestro caso, el proyecto se llama “MigracionSG2”. Elegimos SQL server como destino, y hacemos clic en “Next”.

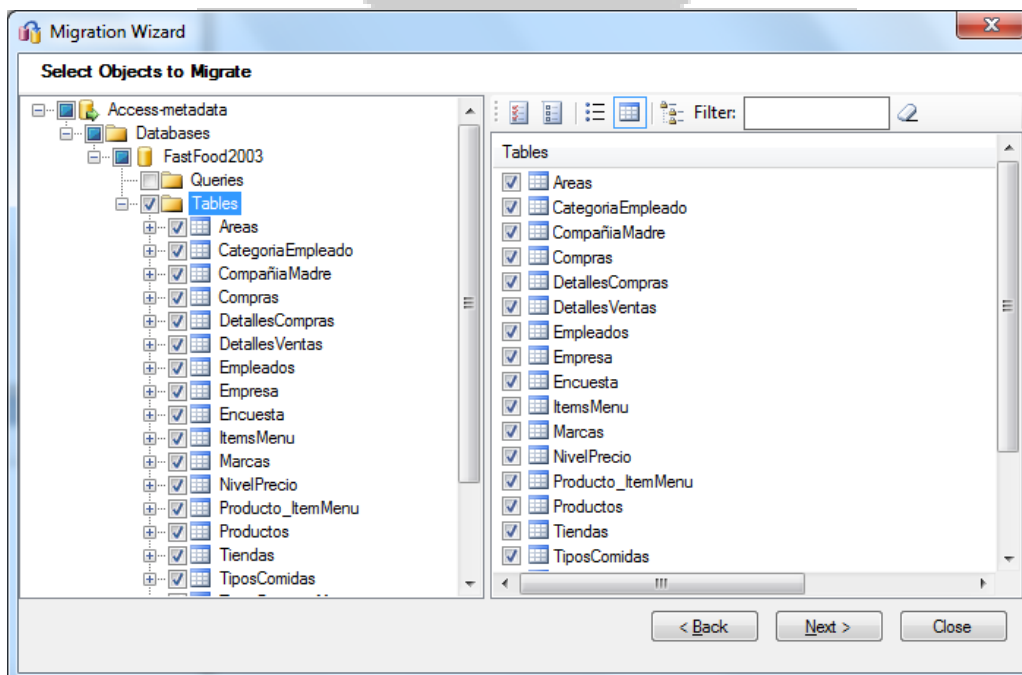


A continuación, debemos seleccionar el origen de los datos. Para ello, vamos a “Add Databases”, y elegimos la base de datos de Access creada anteriormente.

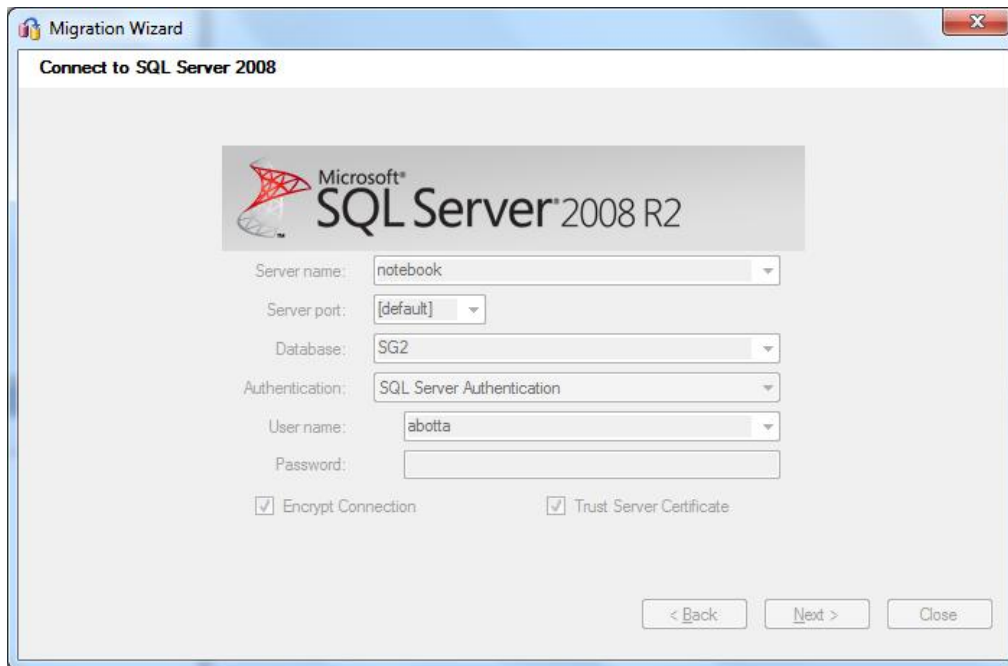




Luego, debemos seleccionar qué tablas y datos migrar. En este caso, seleccionamos todo y hacemos clic en “Next”.



Posteriormente, configuramos el destino. En este caso, una Base de datos en red, en el servidor “notebook”. Una vez ingresada la Base de Datos de destino y las credenciales de acceso, hacemos clic en “Next”.

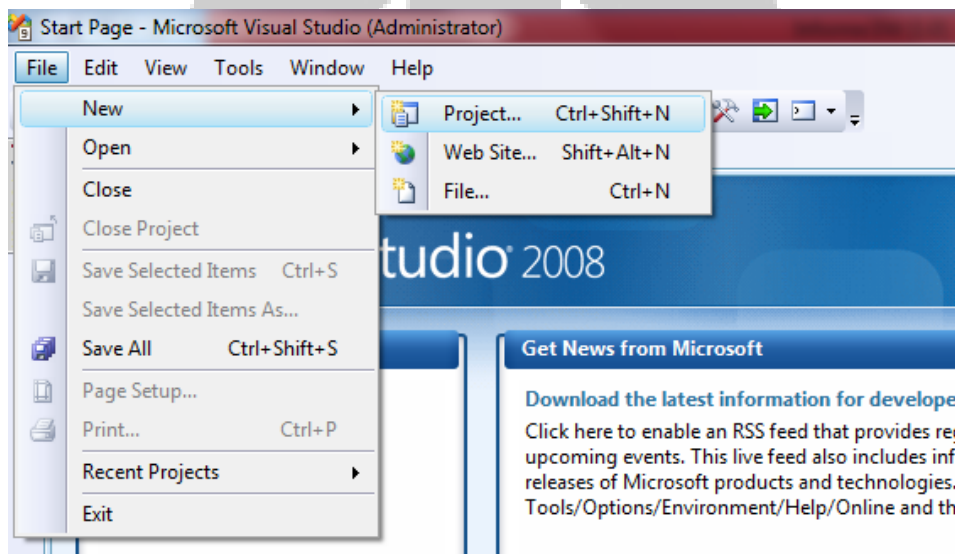


Finalmente, luego de esperar unos momentos, la Base de Datos es migrada a SQL Server.

4. Creación de la vista

Para esto, se utilizará la herramienta “SQL Server Business Intelligence Development Studio” (BIDS).

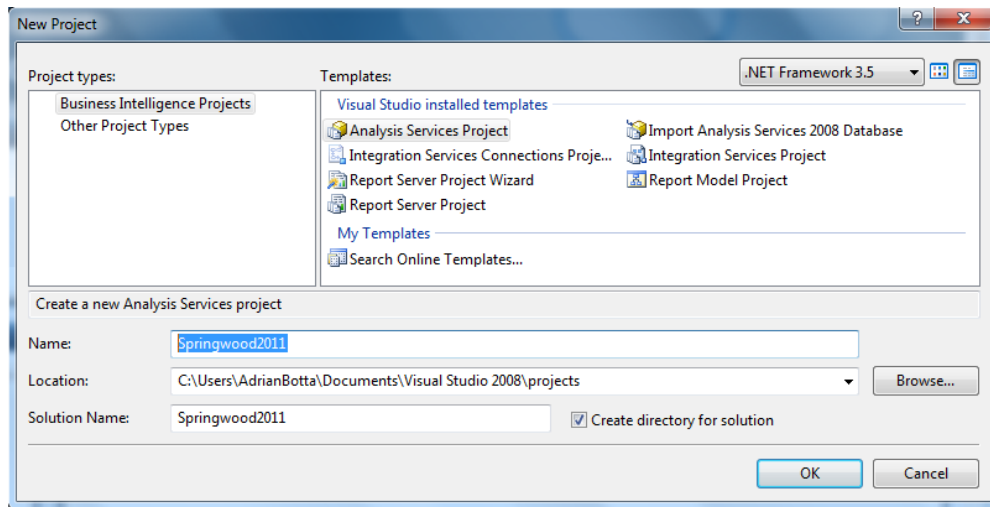
Primero, abrimos BIDS, y vamos a **File → New → Project**



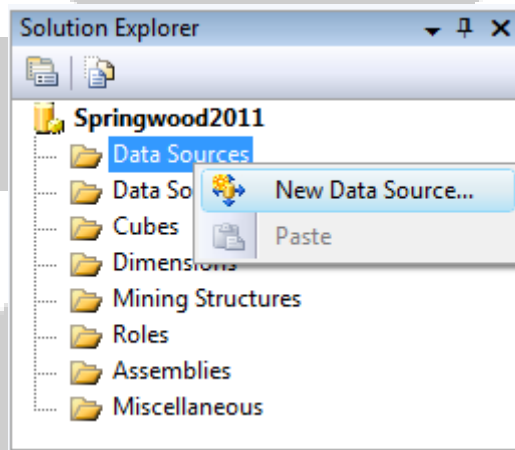
Y elegimos un proyecto de “Analysis Services”, llamado “Springwood2011”. Hacemos



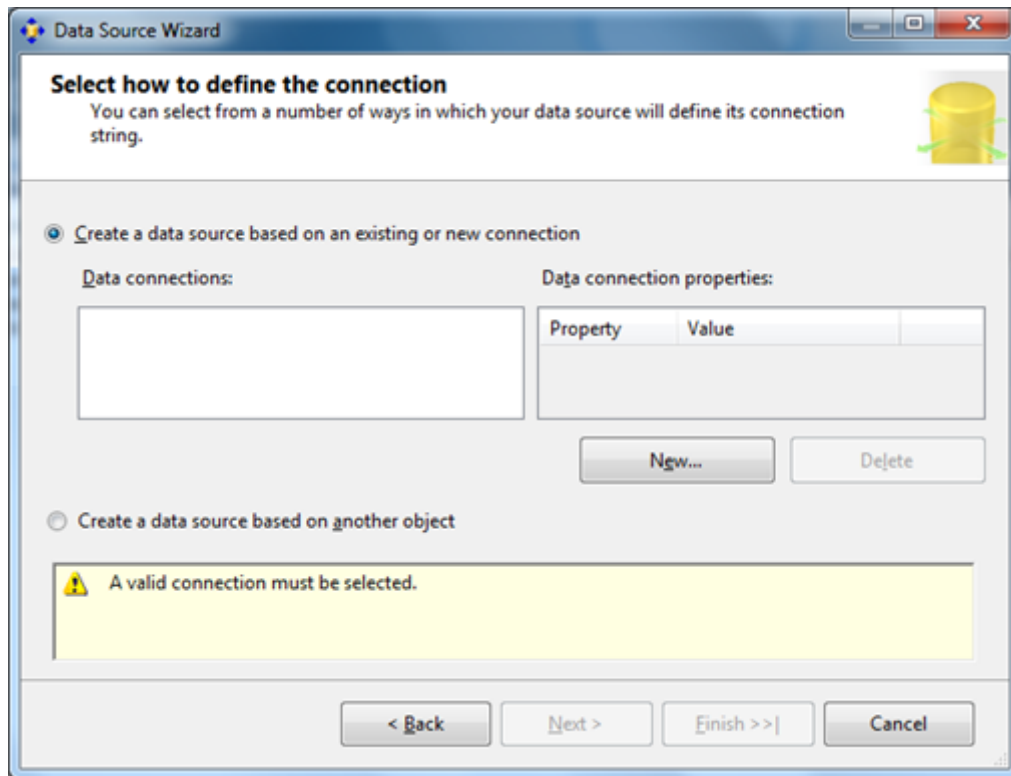
clic en “OK” para confirmar.



En el panel “Solution Explorer”, hacemos clic derecho sobre “Data Sources”, y luego en “New Data Source...”.

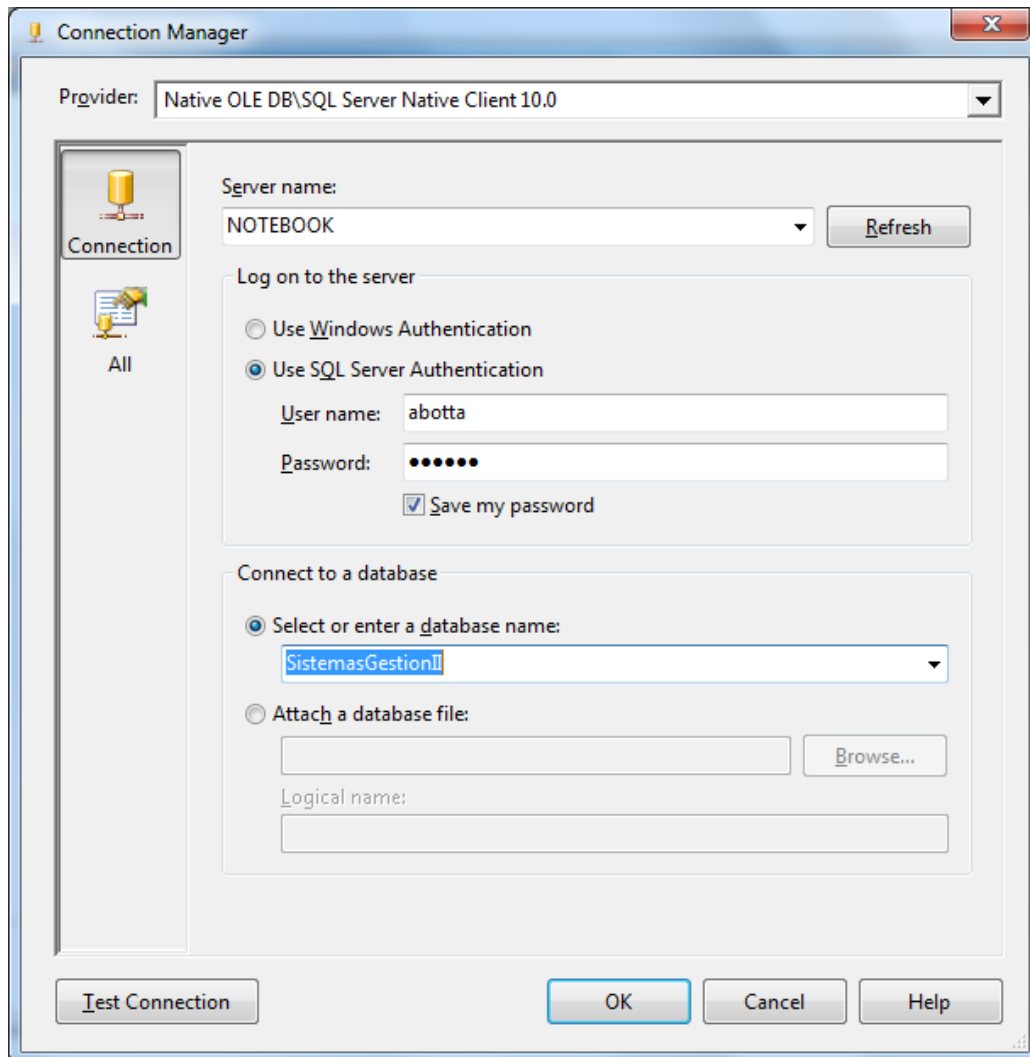


A continuación, se presentará un asistente como el siguiente.

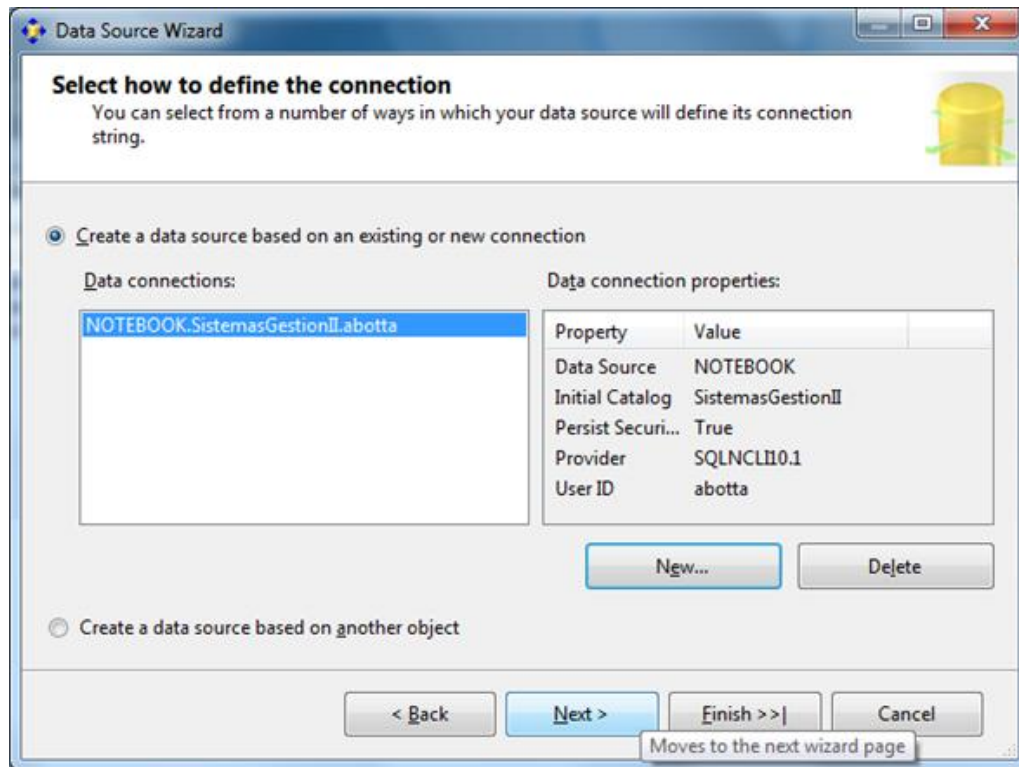


Hacemos clic en “New” para agregar una conexión.

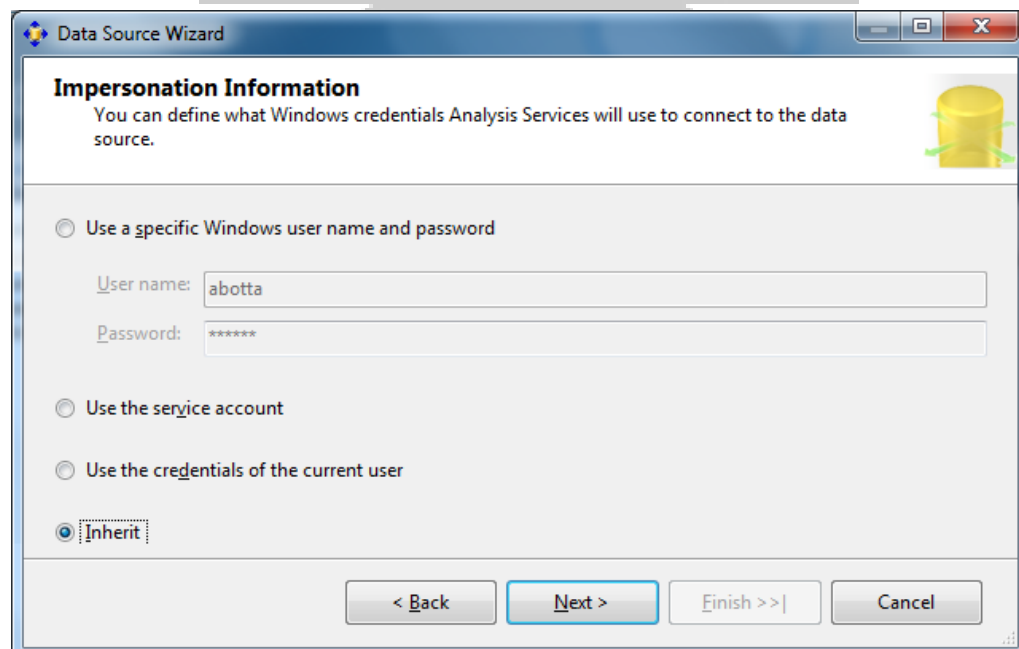
Configuramos el nombre del servidor y las credenciales, como lo hicimos en el paso anterior. Luego, hacemos clic en “OK”.



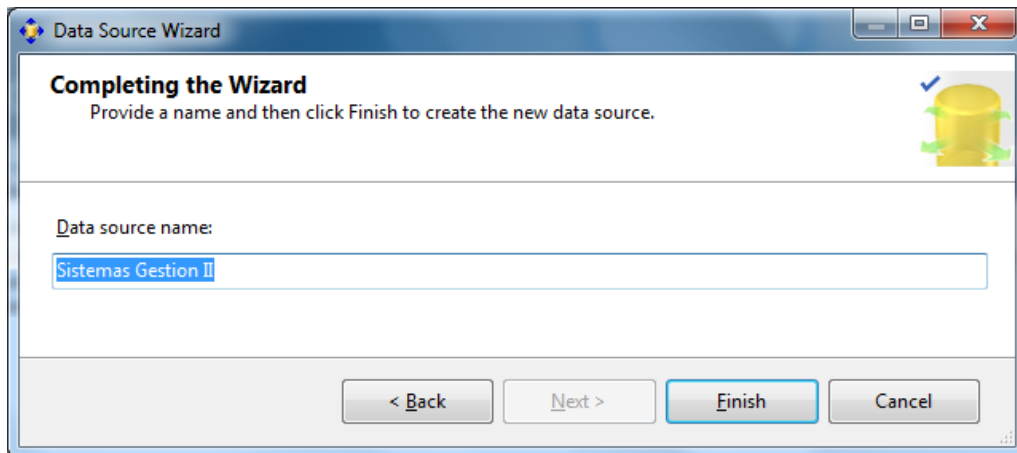
Se mostrará una pantalla como la siguiente. Hacemos clic en “Next”.



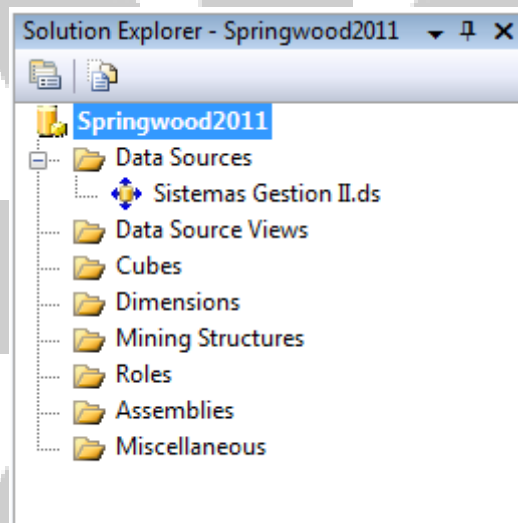
En la siguiente pantalla, heredamos las credenciales de seguridad, y hacemos clic en “Next”.



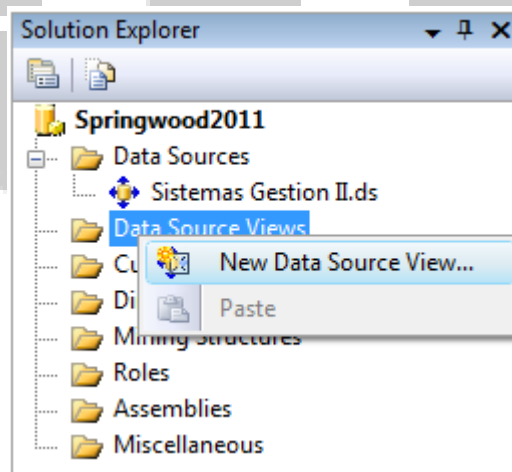
Finalmente, nombramos el DataSource como “Sistemas Gestion II” y hacemos clic en “Finish”.



Podemos ver que ahora, en el panel “Solution Explorer” figura nuestro Data Source.



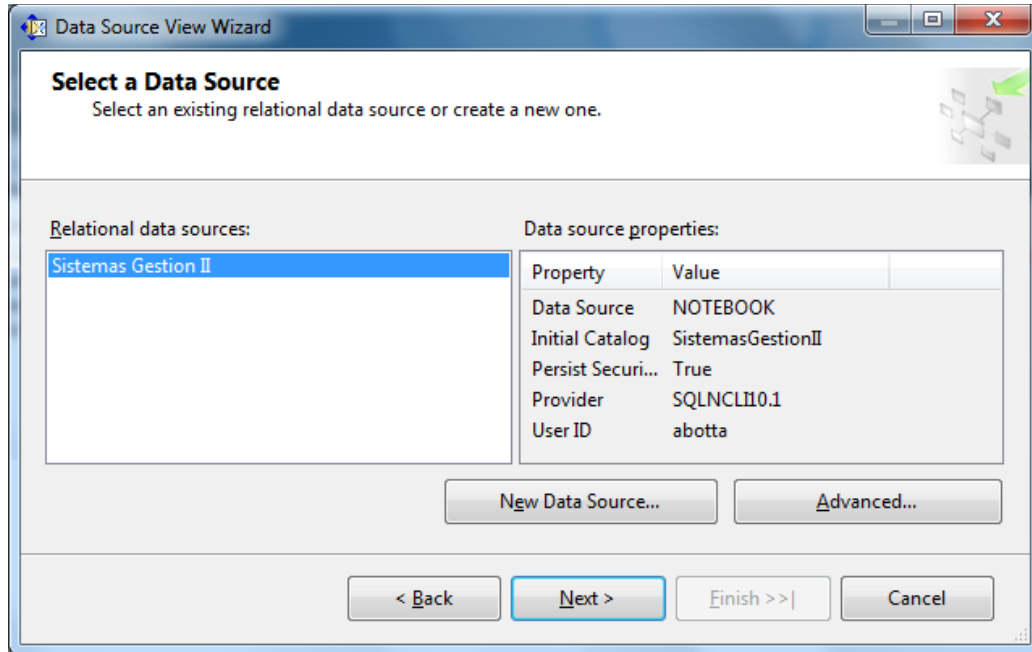
Posteriormente, en “Solution Explorer”, hacemos clic derecho sobre “Data Source Views” y luego en “New Data Source View...” como indica la figura.



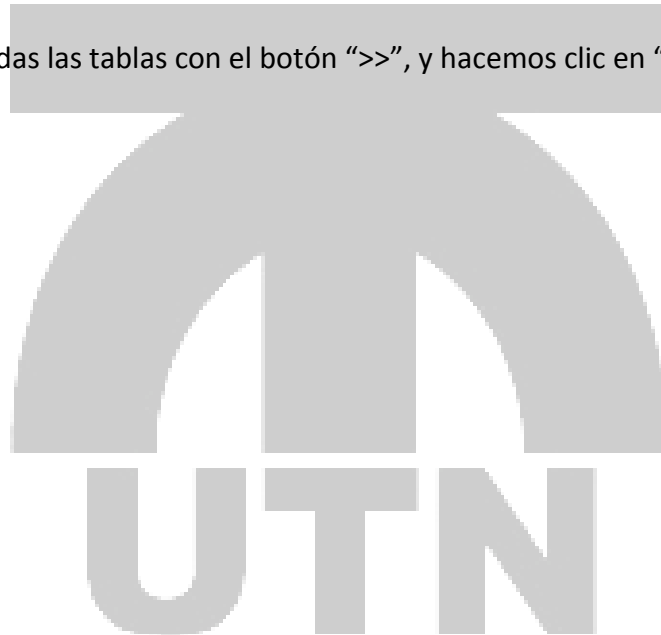


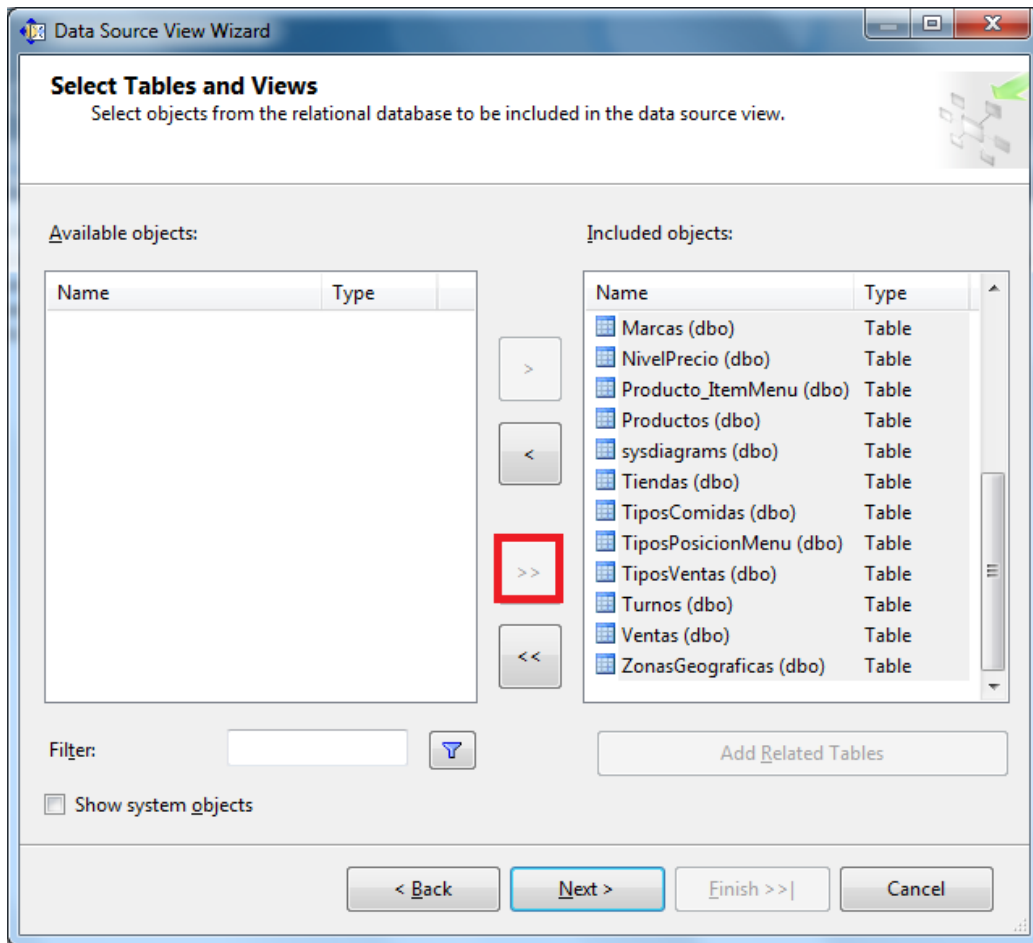
Se abrirá un Wizard, que deberemos seguir.

Seleccionamos el Datasource creado anteriormente, y hacemos clic en “Next”.

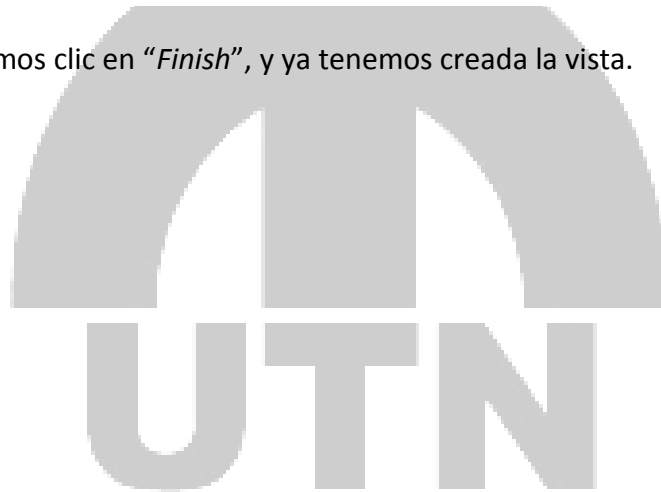


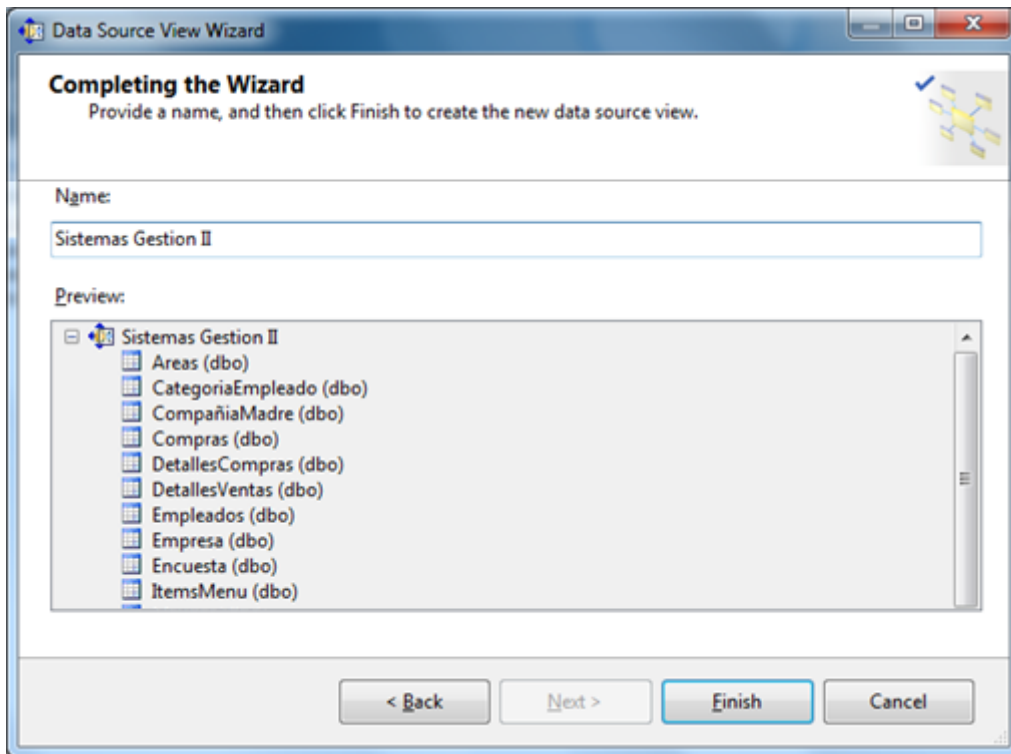
Seleccionamos todas las tablas con el botón “>>”, y hacemos clic en “Next”.



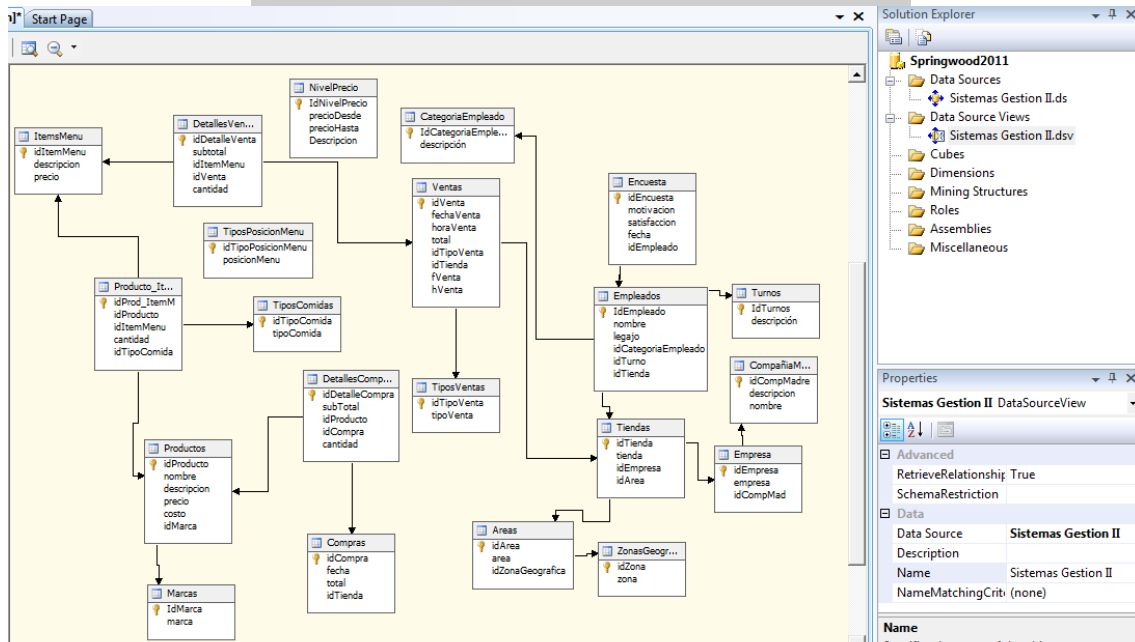


Finalmente, hacemos clic en “*Finish*”, y ya tenemos creada la vista.





En la pantalla principal aparecerá un diagrama similar al de la base de datos relacional.

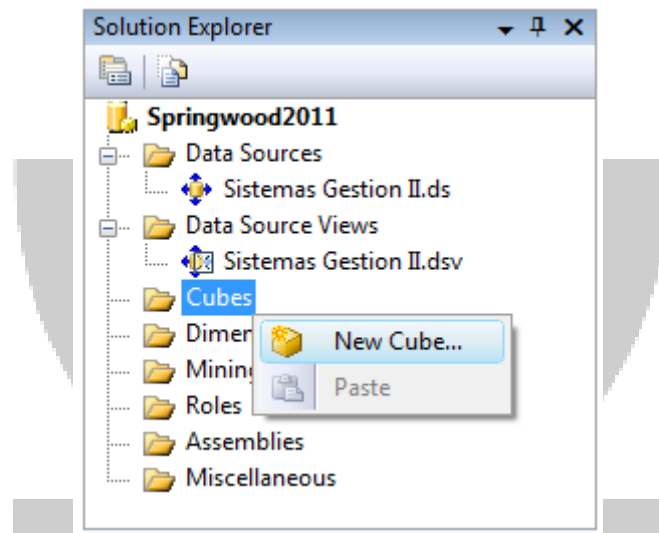




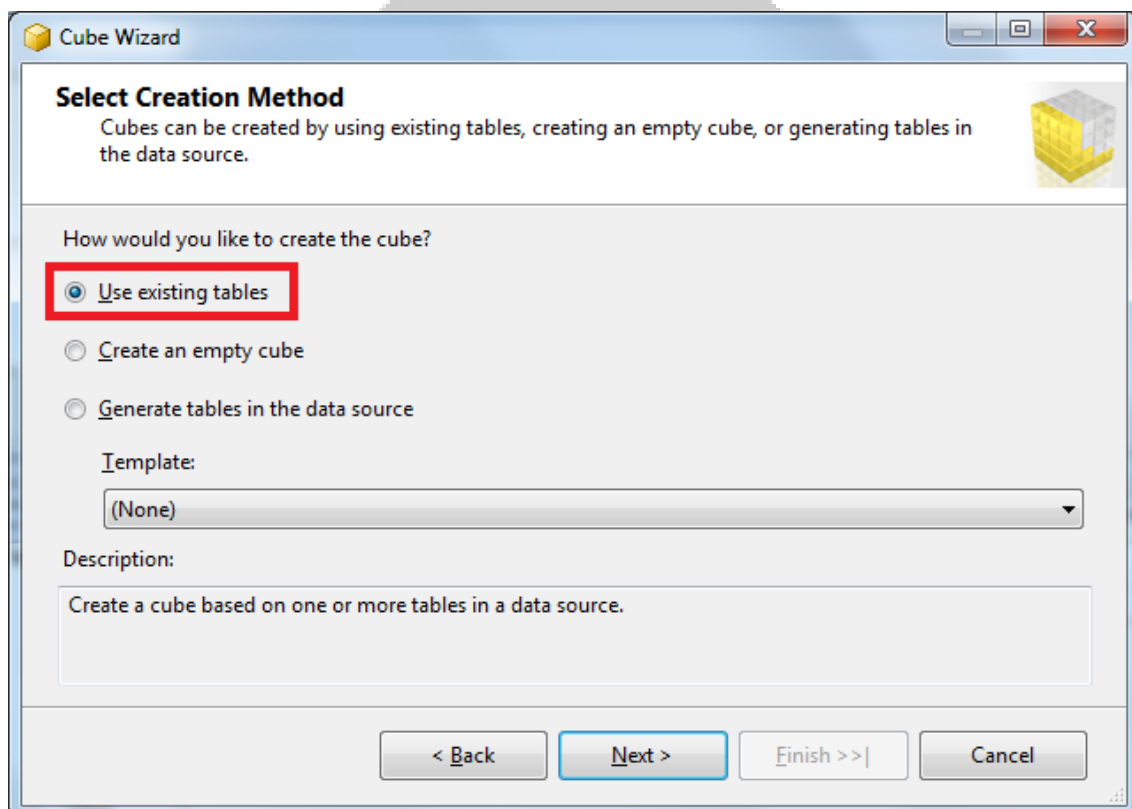
Creación de Cubos

A fin de ejemplificar el proceso de creación de cubos, utilizaremos el cubo las de “Encuestas” mencionado en el enunciado.

Para crear un cubo, en el “Solution Explorer”, hacemos clic derecho sobre “Cubes” y luego en “New Cube...”.

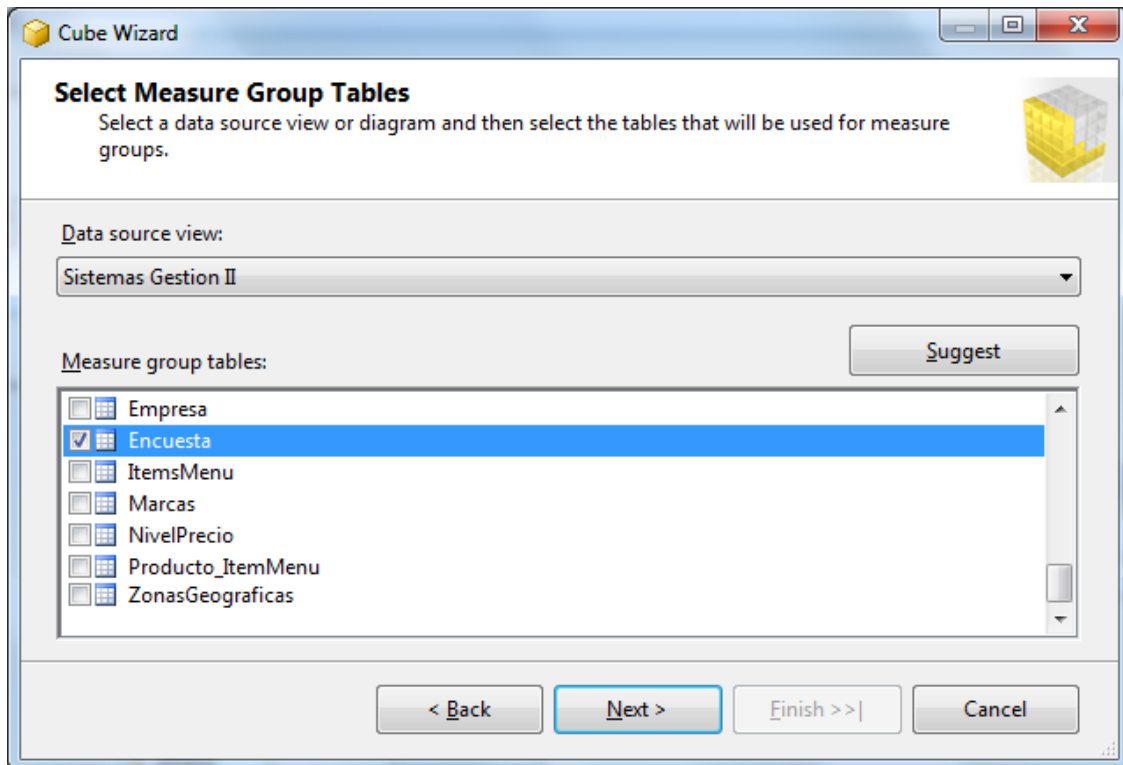


En el asistente para crear cubos, hacemos clic en la primera opción (Usar tablas existentes) y luego en “Next”

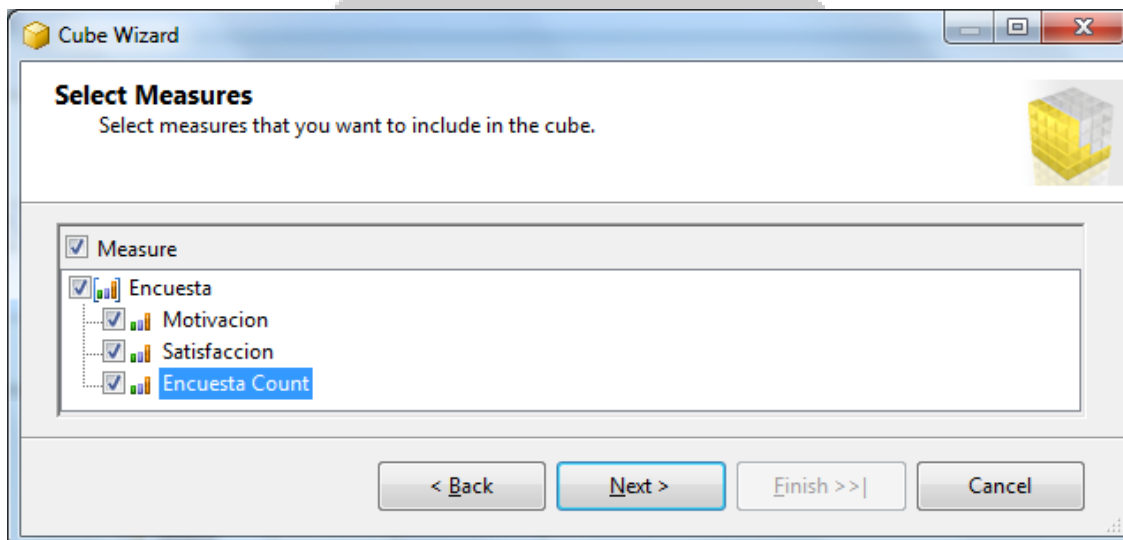




Seleccionamos las tablas a utilizar. En este caso, "Encuesta".



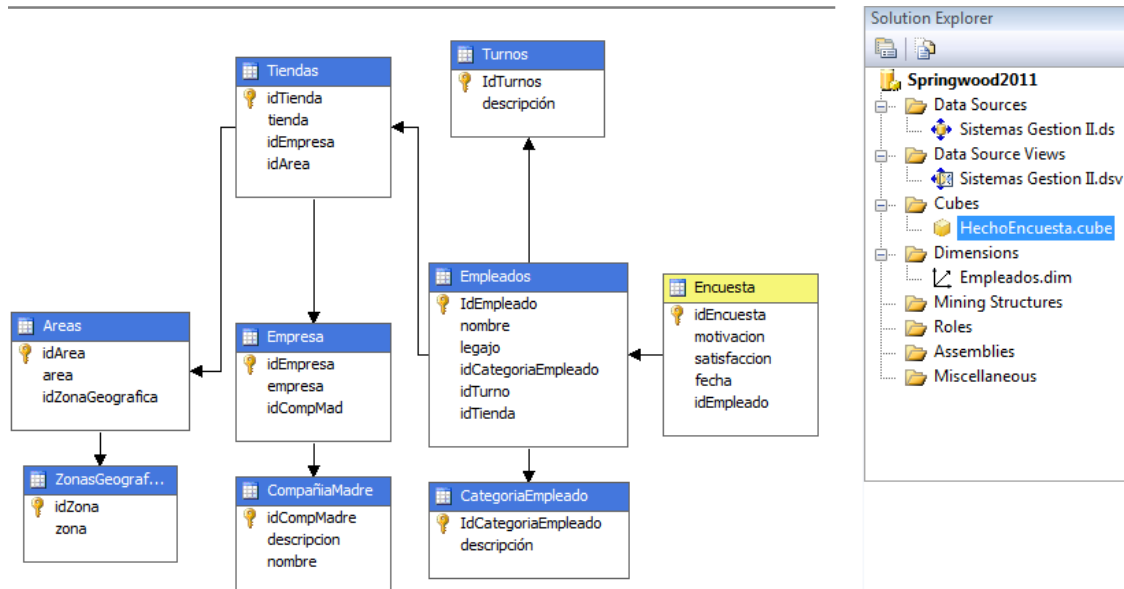
Seleccionamos las medidas que necesitamos. En este caso, seleccionamos todas. Luego se pueden agregar nuevas medidas.



A continuación se mostrarán todas las tablas relacionadas con la encuesta. Seleccionamos las que deseemos para que se conviertan en dimensiones, y hacemos clic en "Next". En este caso, no seleccionamos ninguna.

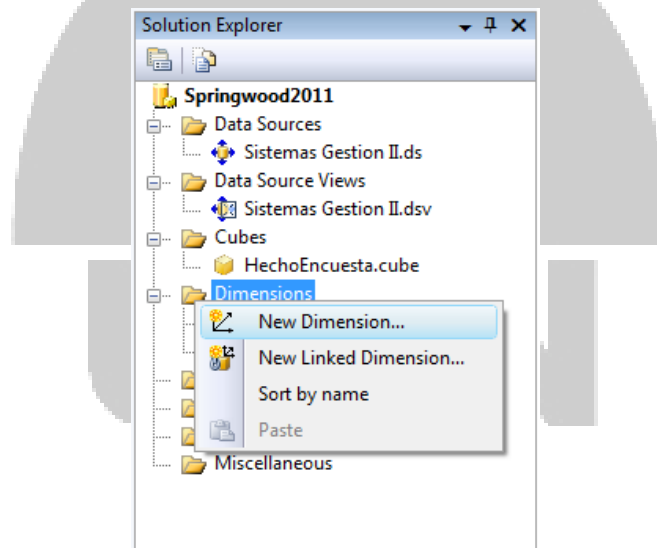


Finalmente, ingresamos el nombre del cubo. En este caso, “HechoEncuesta”, y hacemos clic en finish. Podemos ver el cubo creado en el “Solution Explorer”. En la pantalla central, podemos ver las tablas relacionadas.



Creación de Dimensiones

Para esto, vamos a “Solution Explorer”, botón derecho sobre “Dimensions”, “New Dimension”.



Primero, vamos a crear la **dimensión Tiempo**. Para ello, elegimos “Generate a time table on the server”, y hacemos Clic en “Next”



Dimension Wizard

Select Creation Method
You can base the dimension on an existing table or generate a new table as the source.

How would you like to create the dimension?

☐ Use an existing table

☐ Generate a time table in the data source

☒ **Generate a time table on the server**

☐ Generate a non-time table in the data source

Template:
(None)

Description:
Create a time dimension directly on the server, without using an underlying data source. The dimension will contain data for the date range, attributes, and calendars you specify.

< Back **Next >** Finish >> | Cancel

Seleccionamos las fechas de inicio, fin, y los periodos de tiempos deseados y hacemos clic en "Next".

Dimension Wizard

Define Time Periods
Select the time periods to use when generating the hierarchies.

First calendar day: lunes , 01 de enero de 2001

Last calendar day: sábado , 31 de diciembre de 2011

First day of the week: Sunday

Time periods:

- ☒ Year
- ☐ Half Year
- ☐ Quarter
- ☒ Trimester
- ☒ Month
- ☐ Ten Days
- ☒ **Week**
- ☒ Date

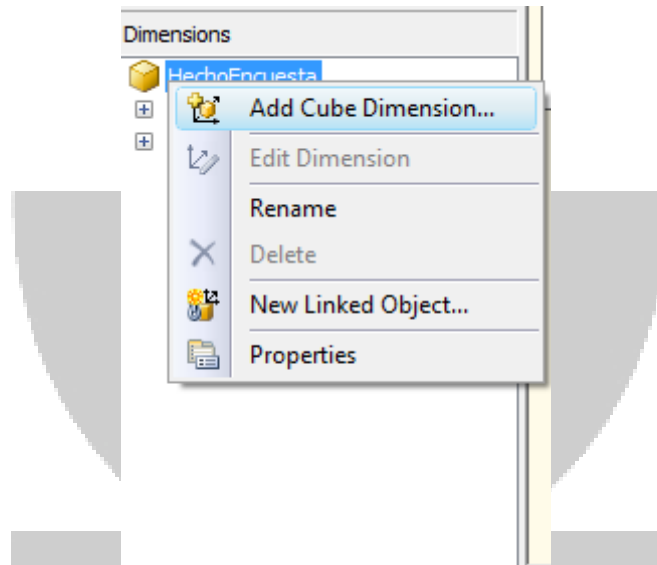
Language for time member names: English (United States)

< Back **Next >** Finish >> | Cancel



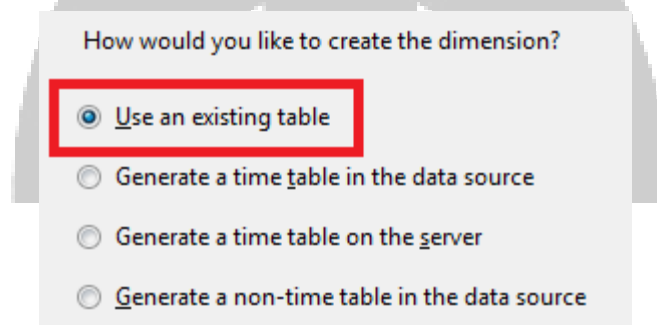
Seguimos colocando “Next” hasta que aparezca un campo para ingresar el nombre de la dimensión, y le colocamos “TiempoE”

Posteriormente, en el panel “Dimensions”, sobre nuestro cubo, hacemos clic derecho, y agregamos la dimensión recién Creada.



Ahora, crearemos la **Dimensión Empleado**. Para ello, en el “Solution Explorer”, hacemos clic con el botón derecho sobre “Dimensions”, y luego “New Dimension”.

Ahora, a diferencia de con la dimensión tiempo, elegimos “Use an existing table” y hacemos clic en Next.



Elegimos el Data Source del cual leer información, la tabla a partir de la cual crear la dimensión, y el ID. En este caso, la tabla empleados con su ID, y hacemos clic en “Next”.



Specify Source Information
Select a data source and specify how the dimension is bound to it.

Data source view:
Sistemas Gestion II

Main table:
Empleados

Key columns:
IdEmpleado
(Add key column)

Name column:
IdEmpleado

< Back Next > Finish >>| Cancel

A continuación podremos elegir cuáles de las tablas relacionadas con empleados deseamos tener para hacer drill-down/roll-up. En este caso, seleccionamos todas.

Select Related Tables
Tables can be related to the main dimension table either directly or indirectly through another table

Related tables:

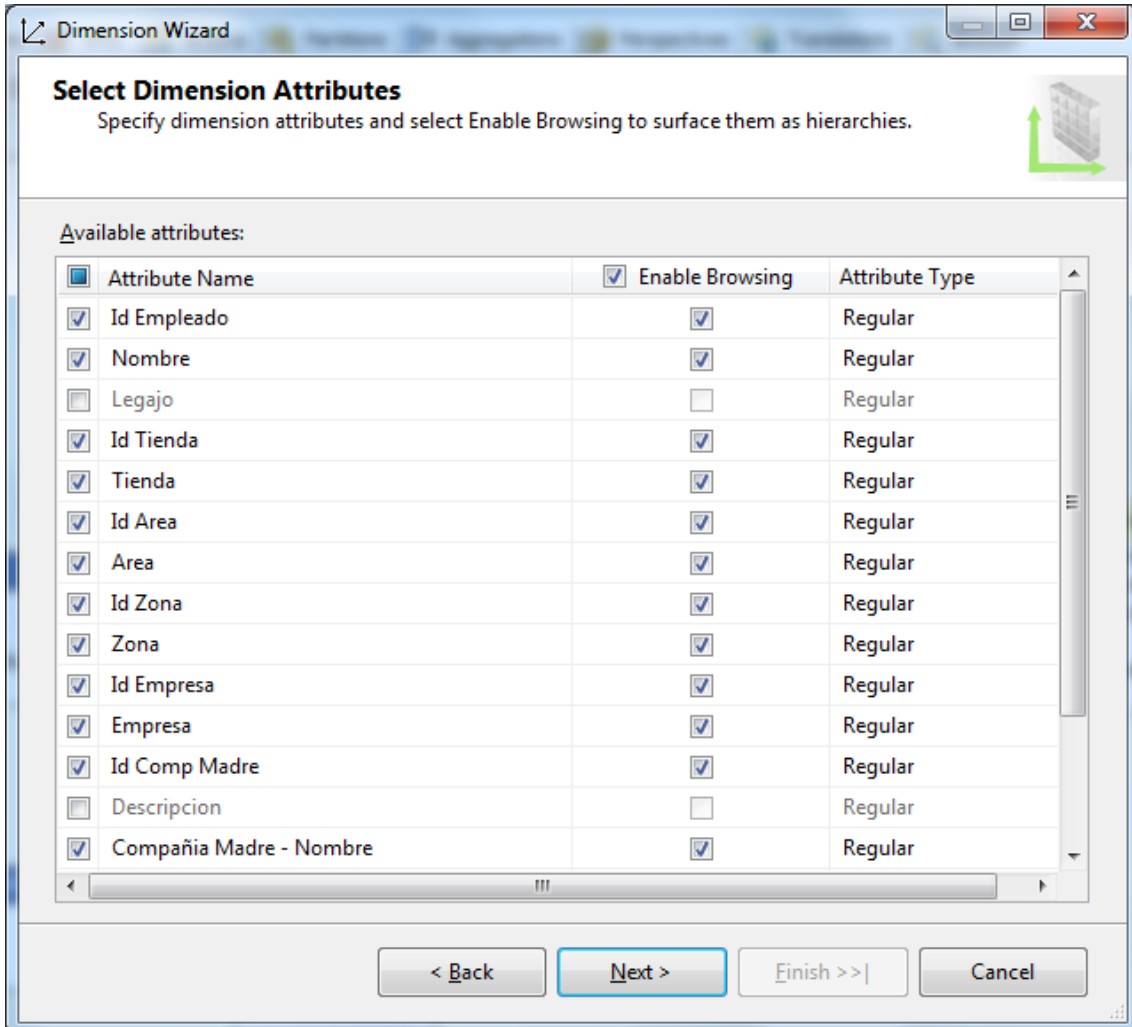
- ☒ Tiendas
- ☒ Areas
- ☒ ZonasGeograficas
- ☒ Empresa
- ☒ CompañiaMadre
- ☒ CategoriaEmpleado
- ☒ Turnos

< Back Next > Finish >>| Cancel

Luego, debemos seleccionar los atributos que deseamos. Por defecto, se muestran los ID's. Sin embargo, el ser humano no comprende fácilmente los ID's, por lo que vamos a



agregar las descripciones/nombres de cada una de las tablas relacionadas.



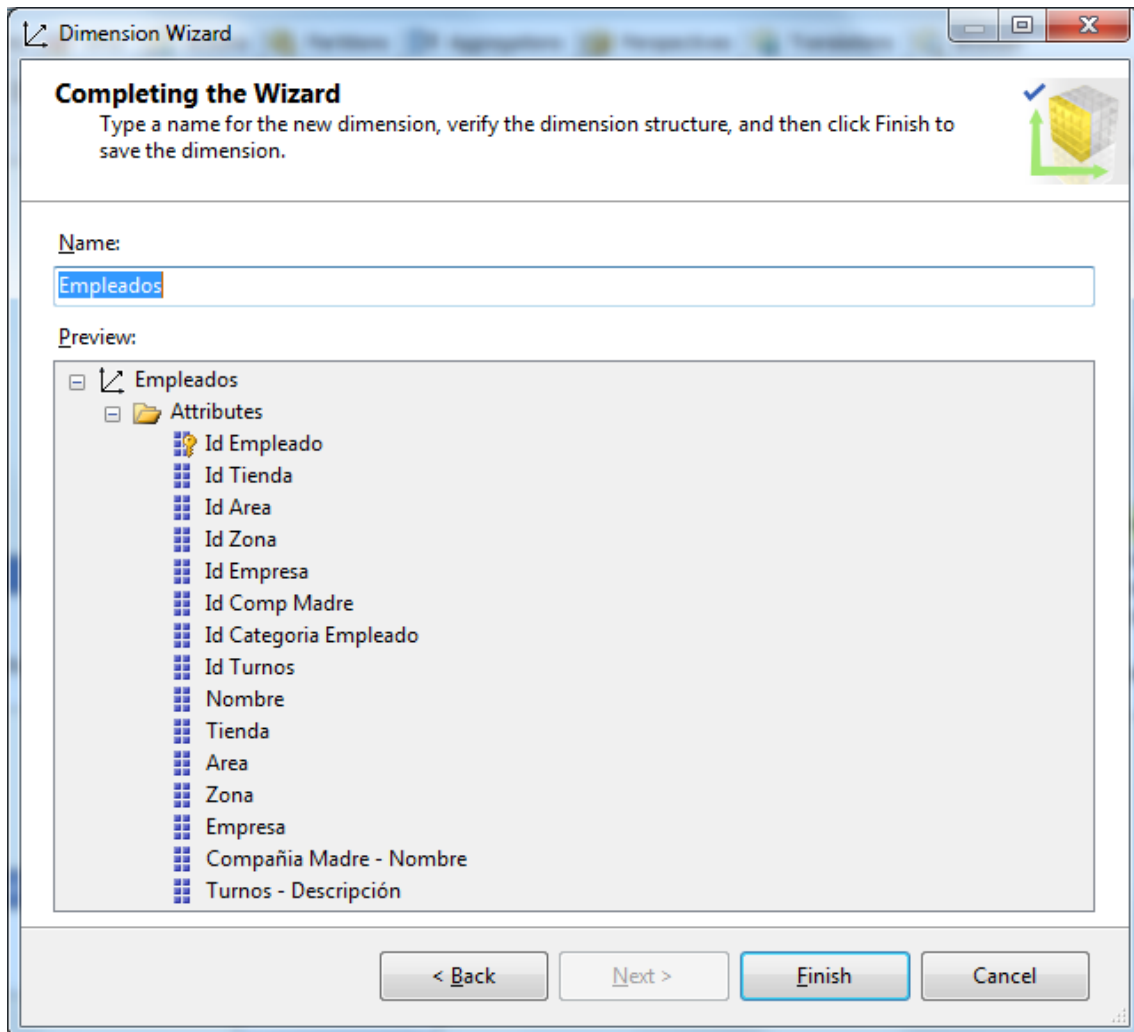
Dimension Wizard

Select Dimension Attributes
Specify dimension attributes and select Enable Browsing to surface them as hierarchies.

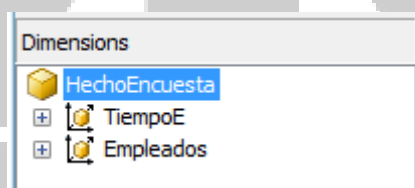
Available attributes:

<input checked="" type="checkbox"/> Attribute Name	<input checked="" type="checkbox"/> Enable Browsing	Attribute Type
<input checked="" type="checkbox"/> Id Empleado	<input checked="" type="checkbox"/>	Regular
<input checked="" type="checkbox"/> Nombre	<input checked="" type="checkbox"/>	Regular
<input type="checkbox"/> Legajo	<input type="checkbox"/>	Regular
<input checked="" type="checkbox"/> Id Tienda	<input checked="" type="checkbox"/>	Regular
<input checked="" type="checkbox"/> Tienda	<input checked="" type="checkbox"/>	Regular
<input checked="" type="checkbox"/> Id Area	<input checked="" type="checkbox"/>	Regular
<input checked="" type="checkbox"/> Area	<input checked="" type="checkbox"/>	Regular
<input checked="" type="checkbox"/> Id Zona	<input checked="" type="checkbox"/>	Regular
<input checked="" type="checkbox"/> Zona	<input checked="" type="checkbox"/>	Regular
<input checked="" type="checkbox"/> Id Empresa	<input checked="" type="checkbox"/>	Regular
<input checked="" type="checkbox"/> Empresa	<input checked="" type="checkbox"/>	Regular
<input checked="" type="checkbox"/> Id Comp Madre	<input checked="" type="checkbox"/>	Regular
<input type="checkbox"/> Descripcion	<input type="checkbox"/>	Regular
<input checked="" type="checkbox"/> Compañia Madre - Nombre	<input checked="" type="checkbox"/>	Regular

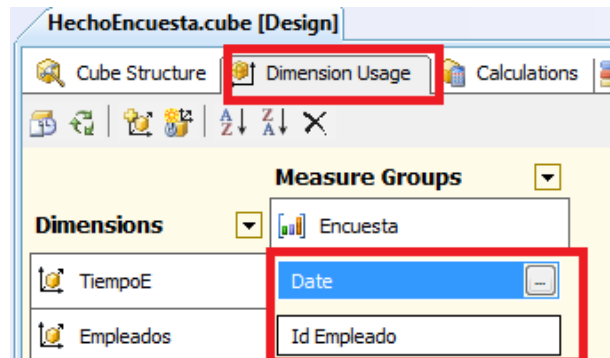
Finalmente, ingresamos el nombre de la Dimensión (empleados) y hacemos clic en Finish.



Al igual que con la dimensión Tiempo, agregamos la dimensión Empleados al Cubo



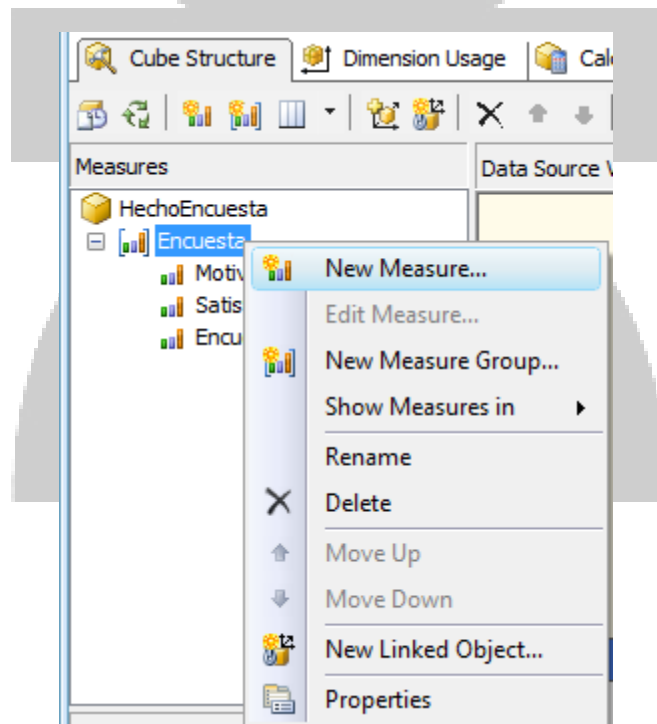
Dentro de la exploración del cubo, ingresamos a la solapa "Dimension Usage" y configuramos los ID's de las dimensiones como sigue:



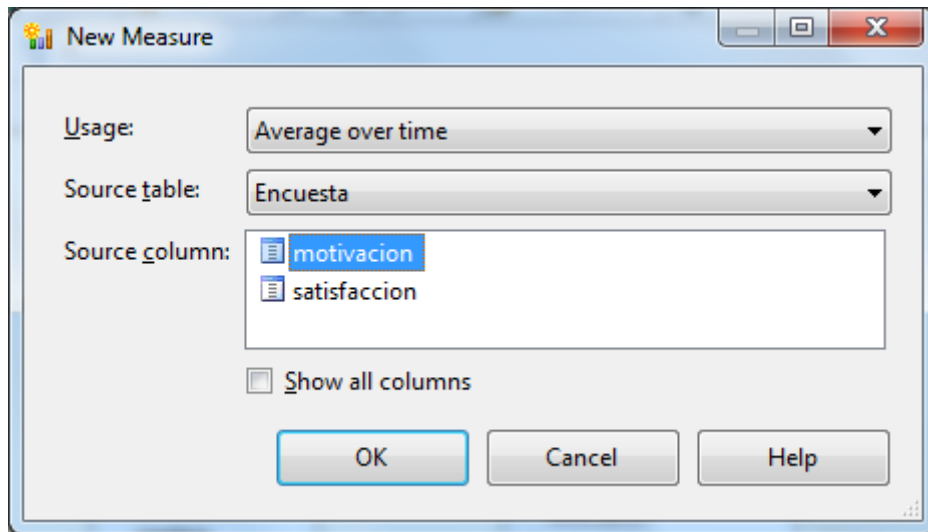
Creación de Medidas

Luego de crear un cubo y sus dimensiones, podemos agregar medidas (measures) al hecho.

Para este caso, vamos a agregar el **promedio de satisfacción y de motivación**. Vamos a la solapa “Cube Structure”, nos posicionamos sobre encuesta y hacemos clic derecho, “New Measure”.

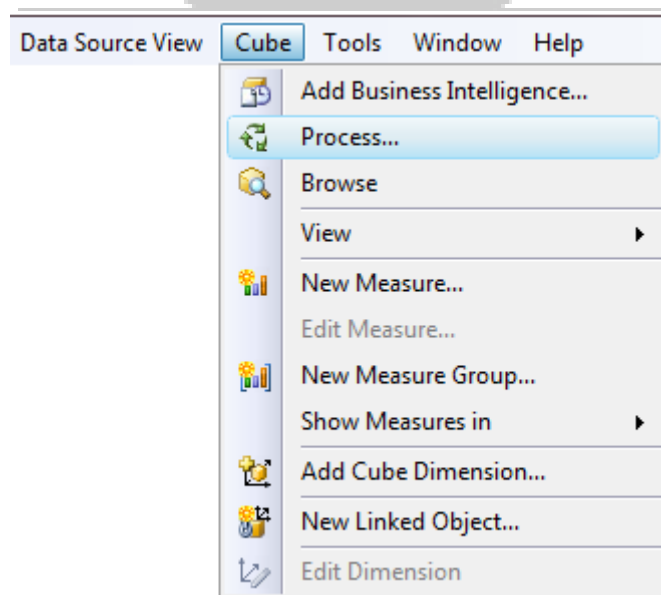


Elegimos en “Usage” → “Average over time” (Promedio), la tabla de Origen, y la columna de origen. Para este caso, elegimos motivación de la taba Encuesta, y hacemos clic en OK.



Ya tenemos una medida creada, ahora hacemos lo mismo para satisfacción.

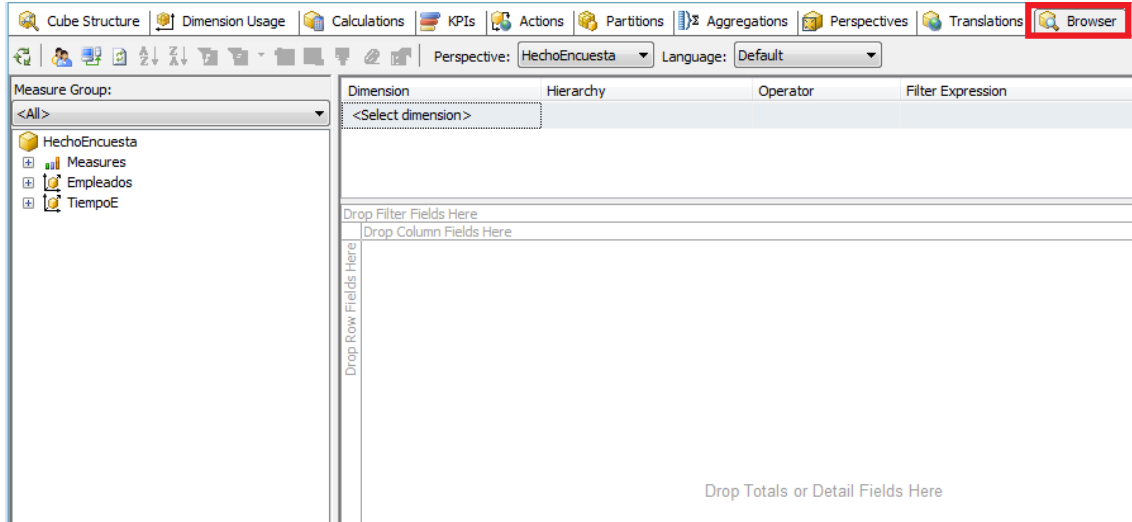
Finalmente, Procesamos el cubo que hemos creado, desde el Menú “Cube” → “Process”.





Navegación del Cubo

Una vez procesado el cubo, éste se puede navegar desde la solapa “Browser”.



Para navegar el cubo, debemos arrastrar:

- Al centro de la pantalla (*Drop Totals or Detail Fields Here*), las medidas (*measures*) que se encuentran en el panel de la izquierda bajo el hecho
- En las Columnas, las Dimensiones que NO queremos desagregar
- En las Filas, las dimensiones que queremos desagregar

Por ejemplo

				Turnos - Descripción ▼			
				Mañana	Noche	Tarde	Grand Total
Year	Trimester	Month	Date	Motivacion - Encuesta	Motivacion - Encuesta	Motivacion - Encuesta	Motivacion - Encuesta
Calendar 2000	Trimester 1, 2000			2.8	2.75	1.66666666666667	2.5
	Trimester 2, 2000	May 2000		2	3	4	3
		June 2000		3.5	4	2.33333333333333	3.14285714285714
		July 2000	Saturday, July 01 2000			5	5
			Thursday, July 20 2000			1	1
			Friday, July 28 2000		5		5
			Total		5	3	3.66666666666667
	August 2000					3	3
	Total			3	4	2.85714285714286	3.21428571428571
	Trimester 3, 2000			1.83333333333333	3.55555555555556	2	2.59090909090909
	Total			2.42857142857143	3.47058823529412	2.2	2.74
Calendar 2001				2.78947368421053	3.31818181818182	3.13333333333333	3.32692307692308
Calendar 2002				2.66666666666667	3.38461538461538	3.42105263157895	3.27083333333333
Calendar 2003				2.45	2.85714285714286	3	2.85964912280702
Calendar 2004				3.34782608695652	2.875	2.85714285714286	3.21875
Calendar 2005				2.93333333333333	2.5625	3.18181818181818	3.1
Calendar 2006				3.04761904761905	2.88	3.57142857142857	3.20689655172414
Calendar 2007				3.5625	2.84210526315789	2.95238095238095	3.53061224489796
Calendar 2008				3.47368421052632	3.33333333333333	3.5	3.6530612244898
Calendar 2009				3.15789473684211	3.16	3.45454545454545	3.52459016393443
Calendar 2010				3.2	2.88235294117647	3.36	3.28333333333333
Grand Total				3.01960784313725	3.03738317757009	3.13953488372093	3.24581939799331



Presentación al Usuario final

Finalmente se diseñó una capa de presentación de manera de poder mostrar al usuario final todos los informes y la información que desea desde las distintas perspectivas que fueron planteadas.

Para esto se realizó un archivo de Excel el cual fue conectado a la solución desarrollada en Ansisys Services y donde se extrae la información de este DataWarehouse. Aquí podemos ver la información de los cubos que también es posible ver a través del explorador de la herramienta BIDS pero esta opción es más amigable y permite realizar los filtros y cambios de dimensiones con mayor facilidad para el usuario final.

Análisis de las ventas

En un principio mostraremos ejemplos de la información que es posible ver de los cubos que involucran a las ventas desde las distintas dimensiones de análisis.

En la figura que se muestra a continuación es posible ver un ejemplo de la información del total de Ventas por Tienda y área de la compañía



Ilustración 12 – Ventas por tienda y area [\$]

También presentamos un ejemplo de un reporte en forma de gráfico de torta de las ventas analizadas con la dimensión “Tipo de ventas” donde se distinguen claramente la cantidad vendida según el tipo de venta (Gravadas y No Gravadas)



Ilustración 13 – Ventas por tipo de ventas

También se presenta un análisis de los datos desde la dimensión tiempo en monto vendido en pesos y en cantidad vendida (Ilustraciones 14 y 15 resp.)

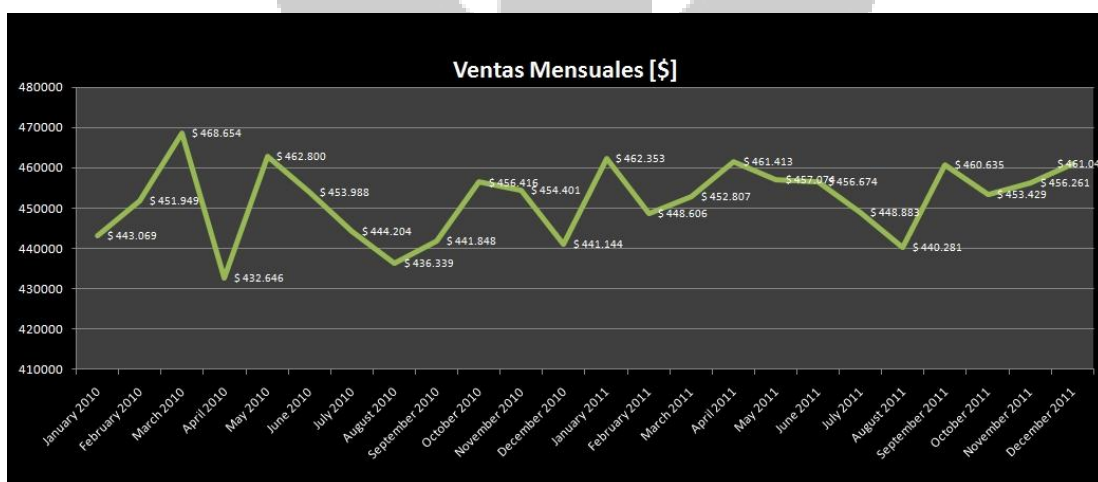


Ilustración 14 – Ventas mensuales [\$]

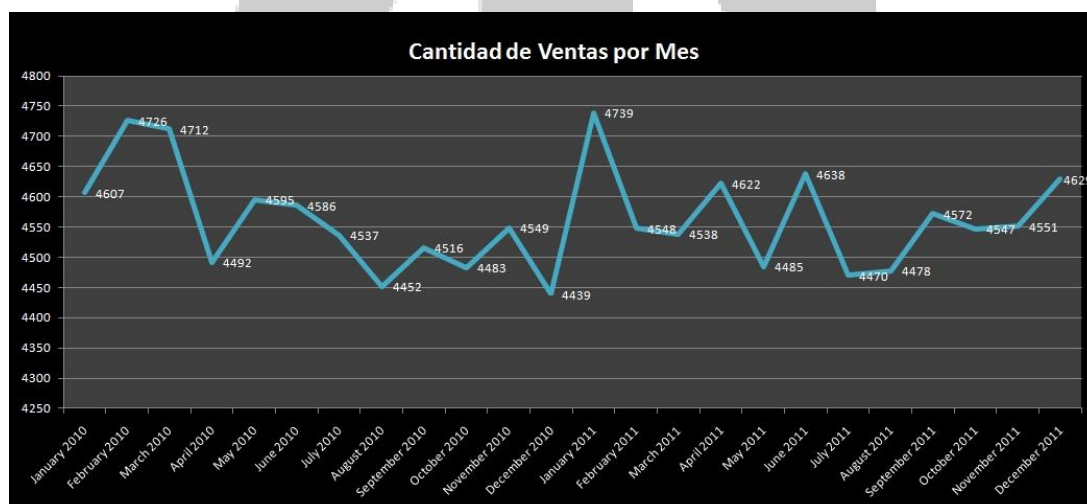


Ilustración 15 – Cantidad vendida mensual



Es posible analizar el total de ventas en la dimensión producto, es decir por “Item de Menu”

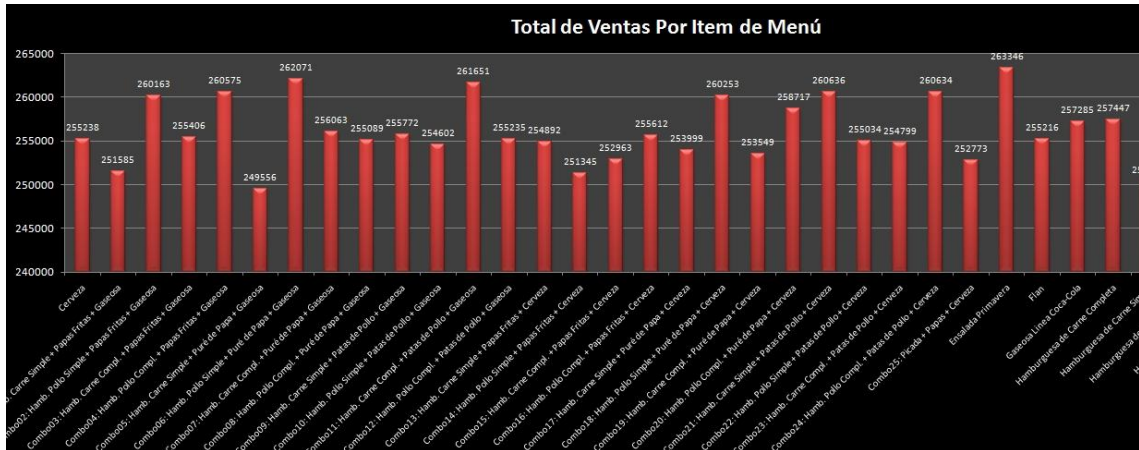


Ilustración 16 – Ventas por Item de Menu

Y de esta misma consulta se puede obtener un ranking Top 10 de las ventas por “Item de Menu”



Ilustración 17 – Top 10 ventas por Item de Menu

Podemos ver en la siguiente figura un ejemplo de un reporte que muestra el incremento de los precios de los productos para los periodos de análisis:



Precios del Año 2010	Precios del Año 2011	Porcentaje Incremento
Cerveza	Cerveza	40,00%
10	14	
Combo01: Hamb. Carne Simple + Papas Fritas + Gaseosa	Combo01: Hamb. Carne Simple + Papas Fritas + Gaseosa	26,67%
15	19	
Combo02: Hamb. Pollo Simple + Papas Fritas + Gaseosa	Combo02: Hamb. Pollo Simple + Papas Fritas + Gaseosa	20,00%
15	18	
Combo03: Hamb. Carne Compl. + Papas Fritas + Gaseosa	Combo03: Hamb. Carne Compl. + Papas Fritas + Gaseosa	23,53%
17	21	
Combo04: Hamb. Pollo Compl. + Papas Fritas + Gaseosa	Combo04: Hamb. Pollo Compl. + Papas Fritas + Gaseosa	17,65%
17	20	
Combo05: Hamb. Carne Simple + Puré de Papa + Gaseosa	Combo05: Hamb. Carne Simple + Puré de Papa + Gaseosa	20,00%
15	18	
Combo06: Hamb. Pollo Simple + Puré de Papa + Gaseosa	Combo06: Hamb. Pollo Simple + Puré de Papa + Gaseosa	13,33%
15	17	
Combo07: Hamb. Carne Compl. + Puré de Papa + Gaseosa	Combo07: Hamb. Carne Compl. + Puré de Papa + Gaseosa	23,53%
17	21	
Combo08: Hamb. Pollo Compl. + Puré de Papa + Gaseosa	Combo08: Hamb. Pollo Compl. + Puré de Papa + Gaseosa	17,65%
17	20	
Combo09: Hamb. Carne Simple + Patas de Pollo + Gaseosa	Combo09: Hamb. Carne Simple + Patas de Pollo + Gaseosa	15,00%
20	23	
Combo10: Hamb. Pollo Simple + Patas de Pollo + Gaseosa	Combo10: Hamb. Pollo Simple + Patas de Pollo + Gaseosa	10,00%
20	22	

Ilustración 18 – Evolución temporal de precios

También podemos observar en un indicador graficado en forma de gráfico de torta las ventas correspondientes analizando a través de la dimensión de “Compañía Madre”



Ilustración 19 – Ventas por compañía madre



Análisis de las compras

De los cubos de compras se utilizó la información para presentar reportes variados que ayuden a los gerentes o usuarios destinatarios de los mismos a tomar decisiones estratégicas con la mayor cantidad de información relevante posible.

En un primer ejemplo podemos ver el monto de compras realizadas por tienda y área en pesos

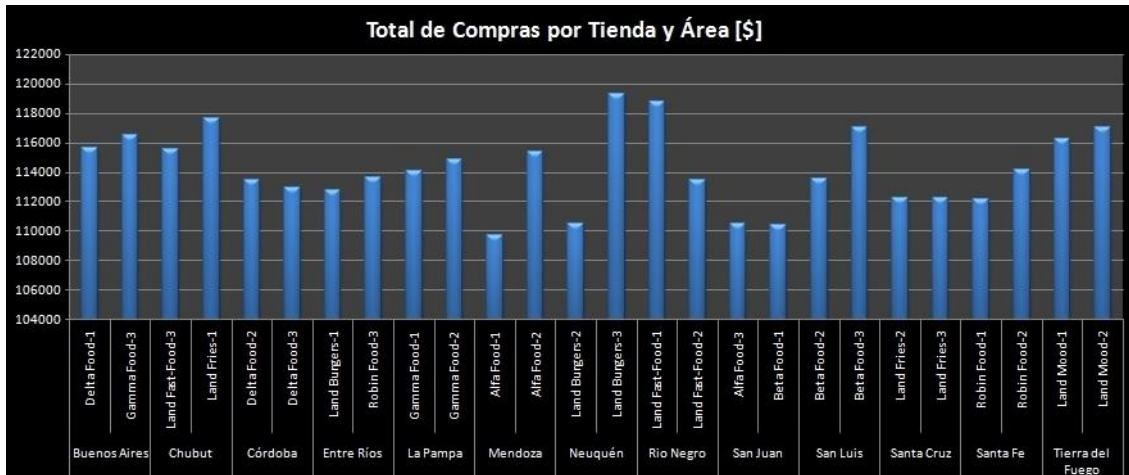


Ilustración 20 – Compras por tienda y área [\$]

También en este va a ser posible para los usuarios hacer el análisis de la evolución de las compras en pesos a través de la dimensión tiempo



Ilustración 21 – Evolución temporal de las compras [\$]



Ventas VS Compras

Puede ser necesario para el usuario visualizar las compras realizadas a través del tiempo, comparadas con las ventas para poder observar a simple vista por ejemplo si la situación de la empresa es favorable en caso de que las ventas sean mayores que las compras.

En la Ilustración 22 vemos la evolución mensual de este indicador

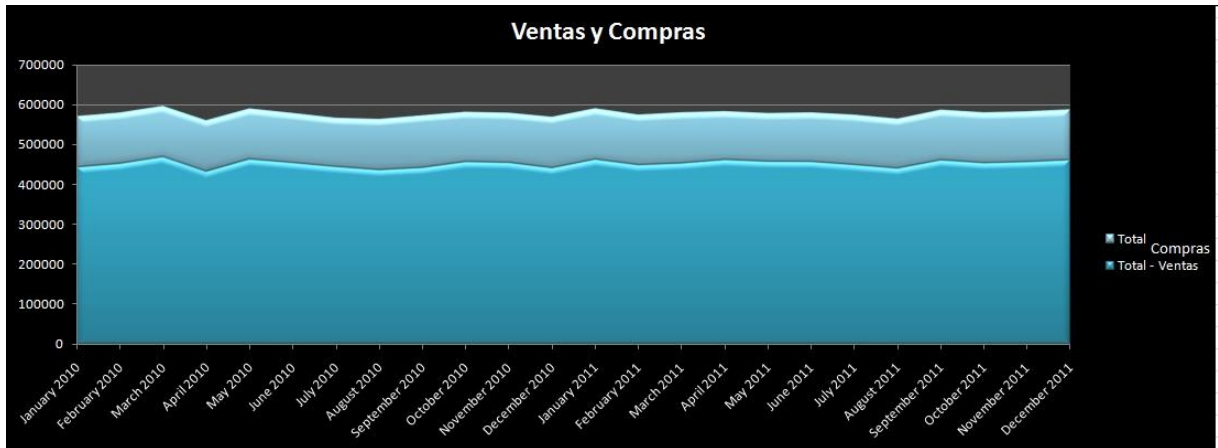


Ilustración 22 – Evolución temporal de ventas vs compras

También podemos analizar este índice comparado solamente con el periodo (año) anterior de ejercicio en la perspectiva de un grafico de barras como se ve en la ilustración a continuación.

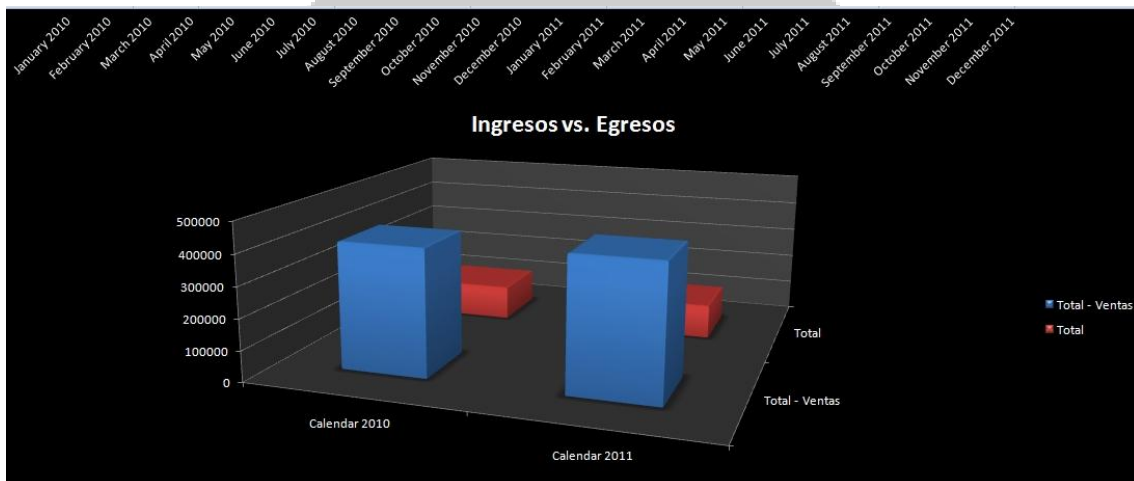


Ilustración 23 – Ventas Vs Compras 2010-2011

Gracias a este análisis también podemos obtener los indicadores de rentabilidad [%] con respecto al periodo anterior y la rentabilidad [\$] en el periodo en curso (2011)



Encuestas de Motivación

Fue necesario también realizar un análisis motivacional de los empleados por lo que es posible visualizar estos datos desde distintas dimensiones.

Mostraremos un ejemplo a continuación del análisis de la evolución en el tiempo de la satisfacción y la motivación a partir del año desde el que se tiene información en la empresa



Ilustración 24 – Evolución de motivación y satisfacción

También fue planteado como requisito de la empresa poder analizar estos hechos desde las dimensiones de las empresas y de las tiendas por lo que es posible visualizar esta información en un reporte como el que se ejemplifica a continuación



Ilustración 25 – Motivación y satisfacción por empresa y tienda

Otro análisis que se muestra en esta área temática es el promedio por zona de la motivación y la satisfacción

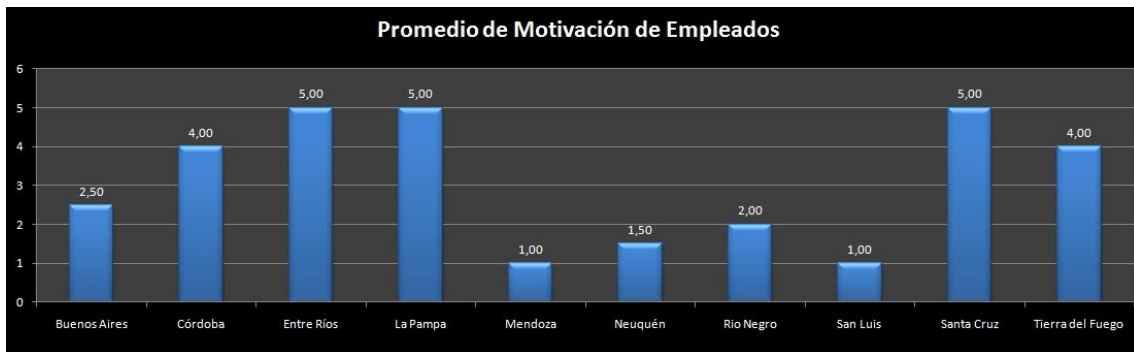


Ilustración 26 – Motivación promedio por zona

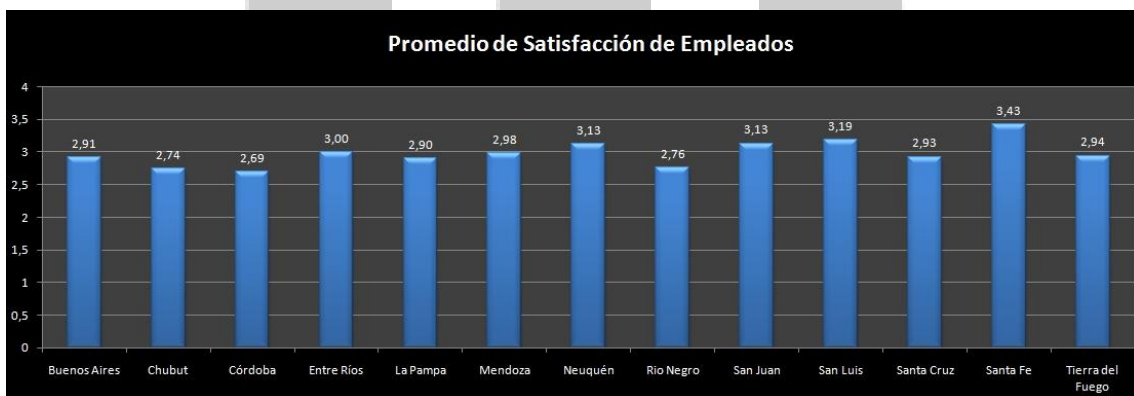


Ilustración 27 – Satisfacción promedio por zona

De estas medidas se pueden obtener índices de Satisfacción y Motivación general y mostrarlos en forma de semáforo o tacómetro.





ANEXO

Código de la Generación de Datos

```

1.
    const
2.  Apellidos: array[1..70] of string = (
3.    'Rodríguez','Gómez','Fernández','López','Díaz','Martínez','Pérez','García',
4.    'Sosa','Álvarez','Torres','Ruiz','Ramírez','Flores','Acosta','Benítez','
5.    'Medina','Suárez',
6.    'Herrera','Aguirre','Pereyra','Gutiérrez','Giménez','Molina','Silva','Ca
7.    'stro','Rojas','Ortíz',
8.    'Núñez','Luna','Juárez','Cabrera','Ríos','Ferreyra','Godoy','Morales','D
9.    'omínguez','Moreno',
10.   'Peralta','Vega','Carrizo','Quiroga','Castillo','Ledesma','Muñoz','Ojeda
11.   ','Ponce','Vera',
12.   'Vázquez','Villalba','Cardozo','Navarro','Ramos','Arias','Coronel','Córd
13.   'oba','Figueroa','Correa',
14.   'Cáceres','Vargas','Maldonado','Mansilla','Farías','Rivero','Paz','Miran
15.   'da','Roldán','Lucero');
16.
17.  Nombres: array[1..70] of string = (
18.    'Santiago','Sebastián','Diego','Nicolás','Samuel','Alejandro','Daniel','
19.    'Mateo','Ángel','Matías',
20.    'Gabriel','Tomás','David','Emiliano','Andrés','Joaquín','Carlos','Alexan
21.    'der','Adrián','Lucas',
22.    'Benjamín','Leonardo','Rodrigo','Felipe','Francisco','Pablo','Martín','F
23.    'ernando','Isaac','Manuel',
24.    'Juan
25.    Pablo','Emmanuel','Emilio','Vicente','Eduardo','Juan','Javier','Jorge','
26.    'Aaron','José',
27.    'Erick','Luis','Cristian','Ignacio','Christopher','Jesús','Kevin','Juan
28.    'José','Agustín','Juan David',
29.    'Simón','Joshua','Maximiliano','Miguel Ángel','Juan
30.    'Sebastián','Bruno','Iván','Gael','Miguel','Thiago',
31.    'Jerónimo','Hugo','Ricardo','Antonio','Ian','Anthony','Pedro','Rafael','
32.    'Jonathan','Esteban');
33.  Categoria: array[1..7] of integer = (1,2,3,2,3,2,3) ;
34.
35.  Marca: array [1..50] of string=(
36.    'AppleCrings','Arnott','ArnottTimTam','Ben&Jerry','Bisc&Mars','BlueRiban
37.    'd',
38.    'Bob','Brannigans','Bugles','BurtonMaryland','Specials','Burtons','Cadbu
39.    'ry',
40.    'Calypso','Cheeselets','CheesyToolBag','Cremosa','Discos','Disney','Domi

```



```
no',
25. 'DoubleTake','EskimoPie','FarmerBrown','Fish','Fishfingers','FlaminHotWo
    tsits',
26. 'FootballCrazy','FoxBiscuits','Frazzles','Gold','GoldenWonder','GoodCrun
    chy',
27. 'Griddles','HobNob','HorrorBags','HulaHoops','Humdinger','JaffaCake','Ju
    icyDropPop',
28. 'Kinder','Kins','KitKat','Kits','KPNuts','Kracks','LayStax','LemonCheese
    cakeKitKat',
29. 'Loacker','LyonsMaid','Magnum');
30.
31. ---COMPRAS-----
32.
33. forIdCompra:=1;
34.
35. for forIdTienda:=1 to 26 do // 26 tiendas
36. begin
37. for forFechaCompraAño:=2010 to 2011 do // del 2010 a 2011
38. for forFechaCompraMes:=1 to 12 do // Enero a Diciembre
39. begin
40. totalCompra:= 0;
41. // Siempre se compra el día 1
42. fechaCompra := StrToDate(''+inttostr(forFechaCompraMes)+'/01/'+inttostr(
    forFechaCompraAño));
43.
44. with ADOQuery_Compras do
45. begin
46. Close;
47. Parameters.ParamByName('idCompra').Value:= forIdCompra;
48. Parameters.ParamByName('fecha').Value:= fechaCompra;
49. Parameters.ParamByName('total').Value:= 0; // Despues hacemos Update...
50. Parameters.ParamByName('idTienda').Value:= forIdTienda;
51. ExecSQL;
52. end;
53.
54. // Detalles de la Compra, 1 PARA CADA PRODUCTO
55. for forIDproducto:=1 to 17 do
56. begin
57.
58. with ADOQuery_DetallesCompras do
59. begin
60. Close;
61. Parameters.ParamByName('idCompra').Value:= forIdCompra;
62. Parameters.ParamByName('idProducto').Value:= forIDproducto;
63.
64. if forIDproducto<=8 then // 540 al mes
```



```

65.  cantidadD:= random(400)+300
66.  else
67.  cantidadD:= random(200)+100;
68.
69.
70.
71.  ////////////////////////////////////////////ACA
    CUANTOS====???
72.  Parameters.ParamByName('cantidad').Value:= cantidadD;
73.
74.  //Busco el costo en la tabla Productos
75.  ADOTable_Productos.Locate('idProducto',forIDproducto,[]);
76.  costo:= ADOTable_Productos.FieldByName('costo').AsFloat;
77.
78.  subtotal:= costo*cantidadD;
79.
80.  Parameters.ParamByName('subtotal').Value:= FormatFloat('0.00', subtotal)
    ;
81.
82.  ExecSQL;
83.  end;
84.
85.  totalCompra:= totalCompra + costo * CantidadD;
86.  end;
87.
88.  with ADOQuery_UpdateCompras do
89.  begin
90.  Close;
91.  Parameters.ParamByName('total').Value:= totalCompra;
92.  Parameters.ParamByName('idCompra').Value:= forIdCompra;
93.  ExecSQL;
94.  end;
95.
96.  inc (forIdCompra);
97.
98.  BarraProgresoFechas.Progress:= BarraProgresoFechas.Progress+1;
99.
100. end; // end For Fechas
101. end; // end for IDtienda
102.
103.
104.
105. -----VENTAS-----
106.
107. inicialForIDVentas:= 1;
108. randomize;
    
```



```
109. ADOConnection1.Connected:= true;
110. ADOTable_ItemsMenu.Active:= true;
111.
112.
113. for forIdTienda:=1 to 26 do // 26 tiendas
114. begin
115. for forFechaVentaAño:=2010 to 2011 do // del 2010 a 2011
116. for forFechaVentaMes:=1 to 12 do // Enero a Diciembre
117. for forFechaVentaDia:=1 to 28 do // Días 1 a 28
118. begin
119. if (forFechaVentaDia mod 6=0) or (forFechaVentaDia mod 7=0) then //fin
    de semana + ventas
120. RNDcantVentasDia:= random(30)+1
121. else
122. RNDcantVentasDia:= random(15)+1;
123.
124. for forIdVenta := inicialForIDVentas to (inicialForIDVentas+RNDcantVentasDia-1) do // Uno por cada venta diaria
125. begin
126.
127. totalVenta:= 0;
128. fechaVenta := StrToDate(inttostr(forFechaVentaMes)+'/'+inttostr(forFechaVentaDia)+'/'+inttostr
129.
130. (forFechaVentaAño));
131.
132. //Calculo de la hora
133. if random(100)>30 then
134. begin
135. // 70% de probabilidades de que sea al mediodia o noche
136. if random(100)>50 then
137. // Venta al mediodia
138. horaVenta:= StrToTime(inttostr(random(2)+12)+':' +inttostr(random(50)+10)
    )
139. else
140. // Venta a la noche
141. horaVenta:= StrToTime(inttostr(random(2)+21)+':' +inttostr(random(50)+10)
    );
142. end
143. else
144. begin
145. // 30 % de probabilidades que sea a la tarde
146. horaVenta:= StrToTime(inttostr(random(15)+9)+':' +inttostr(random(50)+10)
    );
147. end;
148.
```




```
149.
150. with ADOQuery_Ventas do
151. begin
152. Close;
153. Parameters.ParamByName('idVenta').Value:= forIdVenta;
154. Parameters.ParamByName('fechaVenta').Value:= fechaVenta;
155. // Parameters.ParamByName('horaVenta').Value:=
    StrToTime(inttostr(random(15)+9)+':'+inttostr(random(50)+10));
156. Parameters.ParamByName('horaVenta').Value:= horaVenta;
157.
158. Parameters.ParamByName('total').Value:= 0; // Despues hacemos Update...
159. Parameters.ParamByName('idTipoVenta').Value:= random(2)+1; //gravadas o
    no gravadas
160. Parameters.ParamByName('idTienda').Value:= forIdTienda;
161. ExecSQL;
162. end;
163.
164.
165. RNDcantElementos:= random(3)+1; // Cantidad de elementos que compra
166.
167. // Detalles de la venta
168. for i:=1 to RNDcantElementos do
169. begin
170. with ADOQuery_DetallesVentas do
171. begin
172. Close;
173. Parameters.ParamByName('idVenta').Value:= forIdVenta;
174.
175. IDitemMenu:= random(40)+1;
176. Parameters.ParamByName('idItemMenu').Value:= IDitemMenu;
177.
178. cantidadD:= random(3)+1;
179. Parameters.ParamByName('cantidad').Value:= cantidadD;
180.
181. //Busco el precio en la tabla ItemsMenu
182. ADOTable_ItemsMenu.Locate('idItemMenu',IDitemMenu,[]);
183. precio:= ADOTable_ItemsMenu.FieldName('precio').AsInteger;
184.
185. Parameters.ParamByName('subtotal').Value:= precio*cantidadD;
186.
187. ExecSQL;
188. end;
189. totalventa:= totalventa + precio * CantidadD;
190. end;
191.
192. with ADOQuery_UpdateVentas do
```



```
193. begin
194. Close;
195. Parameters.ParamByName('total').Value:= totalventa;
196. Parameters.ParamByName('idVenta').Value:= forIdVenta;
197. ExecSQL;
198. end;
199.
200.
201.
202. end; // FIN for forIdVenta := 1 to RNDcantVentasDia do
203.
204. inicialForIDVentas:= inicialForIDVentas+RNDcantVentasDia;
205.
206. BarraProgresoFechas.Progress:= BarraProgresoFechas.Progress+1;
207. // Showmessage('Venta Cargada'); exit;
208. end;
209. BarraProgresoFechas.Progress:= 0;
210. BarraProgresoVentas.Progress:= BarraProgresoVentas.Progress+1;
211. end;
212.
213.
214.
215. -----EMPLEADOS-----
216.
217. t:= 3+random(5);
218. for forTienda:=1 to 75 do
219. for forTurno:=1 to 3 do
220. for forCategoria:=1 to t do
221. begin
222.
223. with ADOQuery_Empleados do
224. begin
225. Close;
226. Parameters.ParamByName('nombre').Value:= Nombres[random(70)+1]+'
      '+Apellidos[random(70)+1];
227. Parameters.ParamByName('legajo').Value:= random(20000)+1;
228. Parameters.ParamByName('idCategoriaEmpleado').Value:= Categoria[forCateg
      oria];
229. Parameters.ParamByName('idTurno').Value:= forTurno;
230. Parameters.ParamByName('idTienda').Value:= forTienda;
231. ExecSQL;
232. end;
233.
234. end;
235.
236.
```



```
237. -----ENCUESTA-----
238. for forEmpleados:=901 to 1800 do
239. BEGIN
240.
241. cant:= random(3)+1;
242. for t := 1 to cant do
243. begin
244. with ADOQuery_Encuesta do
245. begin
246. Close;
247. Parameters.ParamByName('motivacion').Value:= random(5)+1;
248. Parameters.ParamByName('satisfaccion').Value:= random(5)+1;
249. Parameters.ParamByName('fecha').Value:=StrToDate(inttostr(random(28)+1)+
    '/' +inttostr(random(12)+1)+'/' +inttostr
250.
251. (random(11)+2000));
252. Parameters.ParamByName('idEmpleado').Value:= forEmpleados;
253. ExecSQL;
254. end;
255.
256. end;
257. END;
```

