

TUTORIAL INTELIGENCIA DE NEGOCIOS

Ing. Cristian Bigatti, Ing. Guillermo Leale, Ing. Juan Moine

Versión 2011 R4

Índice general

1. Introducción	3
1.1. Descripción General del Tutorial	3
1.2. Prerrequisitos	5
1.3. Pasos generales a seguir	5
2. Herramientas de análisis y diseño	6
2.1. Esquema starnet por cada requerimiento	6
2.2. Esquema starnet consolidado	7
2.3. Diseño de la base de datos DW (esquema estrella)	7
2.4. Red de tareas	8
2.4.1. Esquema en Nodo	8
2.4.2. Matriz de Precedencias	8
2.5. Diseño	9
2.5.1. Herramientas por cada tarea de la red de proceso	9
2.5.2. Pantallas de análisis por cada requerimiento	10
3. Extracción de Datos - Transformaciones básicas	12
3.1. Crear la Base de Datos de la Staging Area.	12
3.2. Cargar los datos desde las Fuentes de Datos a la Staging Area	13
4. Transformaciones avanzadas: Construcción del Data Warehouse	28
4.1. Creación de la base de datos Data Warehouse	28
4.2. Realizar los procesos de extracción, transformación y carga (ETL)	29
5. OLAP: Definición de una estructura multidimensional con IBM Cognos	42
5.1. Introducción	42
5.2. Prerrequisitos	42
5.3. Crear un origen de datos ODBC	43
5.4. Crear un nuevo proyecto	44
5.5. Publicar un Paquete	55
5.6. Navegar la estructura creada	57

A. Instalación	61
A.1. Instalación y configuración de componentes MySQL	61
A.1.1. Instalación del servidor MySQL	61
A.1.2. Instalación de una herramienta de administración	63
A.1.3. Restauración de una base de datos	64
A.2. Instalación y configuración de Pentaho Data Integration	65
A.2.1. JRE (Java Runtime Enviroment)	65
A.2.2. Pentaho Data Integration	65

Capítulo 1

Introducción

1.1. Descripción General del Tutorial

Nuestro objetivo es la construcción de un Data Warehouse para que TDC (The Drinking Company) pueda atender las siguientes cuestiones:

- Cantidad de litros consumidos y de productos adquiridos por cliente en el tiempo.
- Compra promedio en litros por cliente en el tiempo.

En primera instancia, debemos determinar la estructura que tendrá nuestro DW y posteriormente, los pasos que seguiremos para cargar los datos en el mismo. Para ello podremos usar diferentes técnicas como el *esquema star* para determinar la estructura DW o la *red de tareas* para identificar las fases del proceso ETL (Ver Capítulo 2).

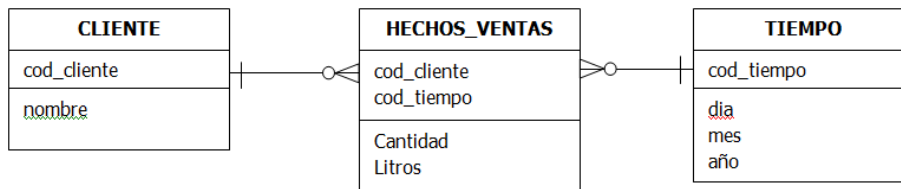
El proceso de construcción del Data Warehouse será dividido en dos grandes etapas:

- Procesos de *extracción, transformación y carga* (ETL).
- Procesamiento *analítico* de datos (OLAP).

Utilizaremos distintas herramientas para cada parte del proceso:

- El proceso ETL será llevado a cabo con la herramienta *Pentaho Data Integration* (PDI), la cual es Open Source. El producto tiene una versión comercial y otra *community* (la cual utilizaremos para este trabajo). El proceso ETL será realizado en los capítulos 3 y 4 del Tutorial
- El proceso de construcción del cubo y el procesamiento OLAP lo llevaremos a cabo con el software *Cognos* de IBM. Este proceso será visto en capítulo 5 del Tutorial.

Luego de una fase de análisis, y utilizando las herramientas propuestas en el Capítulo 2, se ha determinado que el esquema del Data Warehouse que responderá a los requerimientos planteados es el siguiente:



Esta Base de Datos poseerá las siguientes tablas:

CLIENTES	VENTAS
cod_cliente	cod_cliente
nombre	cod_tiempo
	Cantidad
	Litros

En la tabla *Ventas*, el campo “Cantidad” representa la cantidad de productos adquiridos. No es necesario crear la tabla *Tiempo* debido a que se generará en la etapa OLAP.

Para llegar a construir esta Base de Datos, los analistas determinaron que serán necesarias las siguientes Fuentes de Datos:

Products.txt (Productos)

CAMPO
PRODUCT_ID
DETAIL
PACKAGE

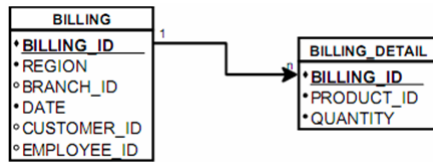
Customer_R.txt (Clientes Minoristas)

CAMPO
CUSTOMER_ID
FULL_NAME
BIRTH_DATE
CITY
STATE
ZIPCODE

Customer_W.txt (Clientes Mayoristas)

CAMPO
CUSTOMER_ID
FULL_NAME
BIRTH_DATE
CITY
STATE
ZIPCODE

Sales (Ventas) BD MySQL



La Base de Datos *History_Sales* no será utilizada en este Tutorial.

1.2. Prerrequisitos

- Servidor de bases de datos MySQL (para este tutorial se usará la versión 5.0)
- Herramienta de administración del servidor MySQL (para este tutorial se usará SQLyog Community Edition).
- Pentaho Data Integration versión 4.1.
- Fuentes de datos del Sistema de Gestión de Ventas de TDC:
 - *Customer_W.txt* (Clientes Mayoristas)
 - *Customer_R.txt* (Clientes Minoristas)
 - *Products.txt* (Productos)
 - Base de datos MySQL *Sales* (Ventas)
- Carpeta de trabajo definida: *C:\SGII + N^o comisión* (Ej. *C:\SGII501*). Usaremos este nombre de carpeta para el presente trabajo. En esta carpeta deberán estar copiadas las fuentes de datos.
- Controlador JDBC para MySQL. Este controlador nos permitirá establecer la conexión con la base de datos *Sales* (Ventas). Asegurarse que las librerías correspondientes al tipo de base de datos (MySQL, SQLServer, PostgreSQL, etc) estén instaladas en la carpeta *... \libext\JDBC*

Las instrucciones para la descarga e instalación se encuentran en el Apéndice A en la página 61

1.3. Pasos generales a seguir

Las siguientes son las tareas cubiertas por este tutorial.

Capítulo 3

1. Crear la Base de Datos de la Staging Area.
2. Cargar los datos desde las Fuentes de Datos a la Staging Area.

Capítulo 4

1. Crear la Base de Datos del Data Warehouse.
2. Realizar los procesos de Extracción, Transformación y Carga (ETL).

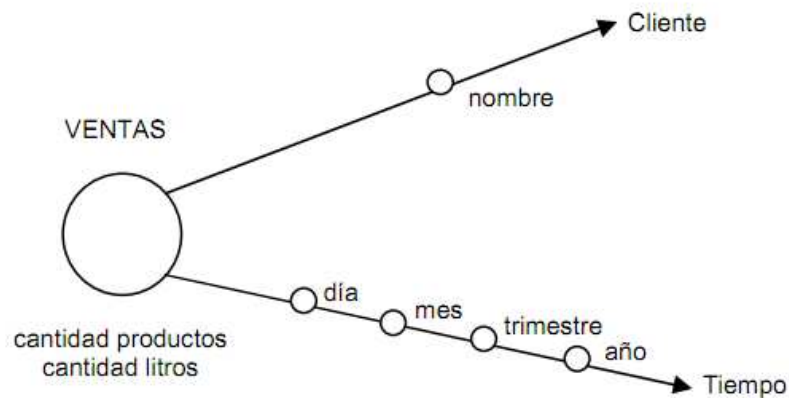
Capítulo 2

Herramientas de análisis y diseño

2.1. Esquema starnet por cada requerimiento

Para cada uno de los requerimientos diagramamos el *esquema starnet* correspondiente. Luego, unificamos las dimensiones y medidas para formar el *esquema starnet consolidado*.

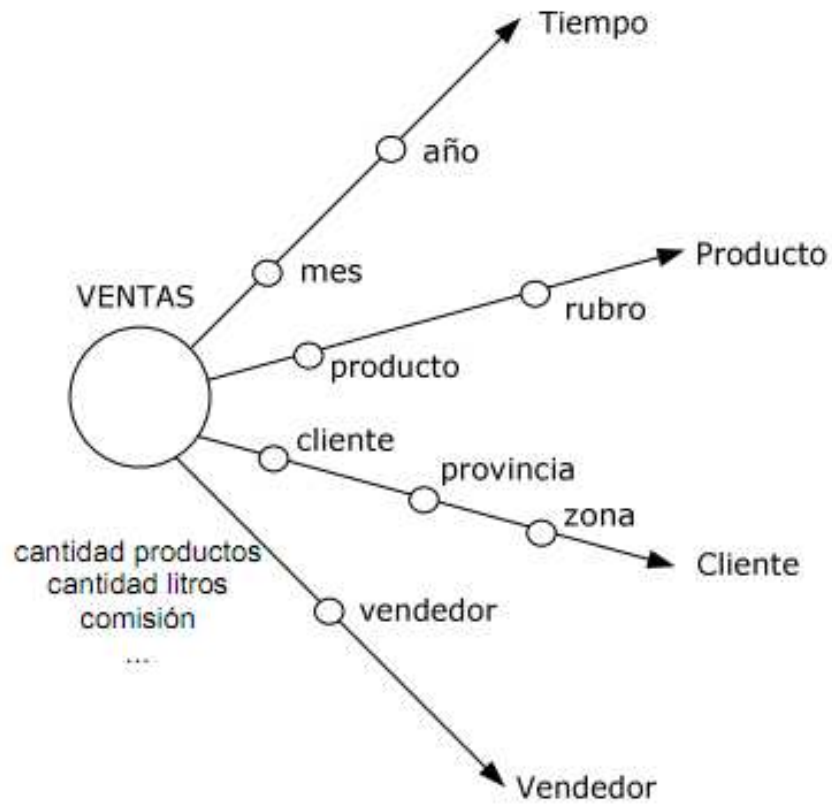
Req 1. Cantidad de litros consumidos y de productos adquiridos por cliente en el tiempo



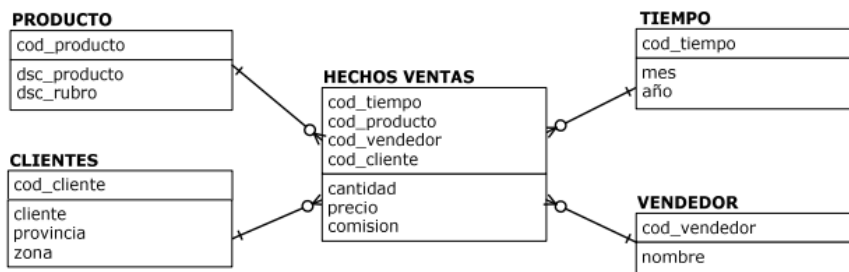
Req 2. Compra promedio en litros por cliente en el tiempo

...
...
...

2.2. Esquema starnet consolidado



2.3. Diseño de la base de datos DW (esquema estrella)

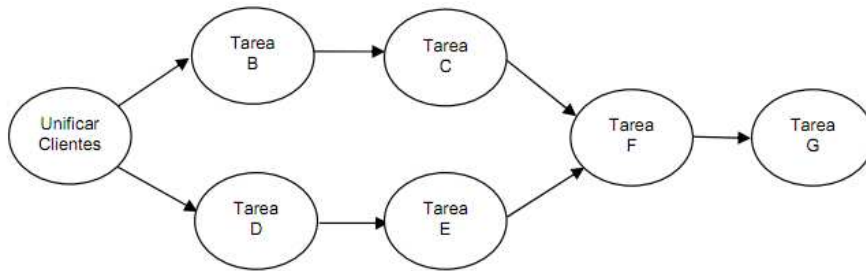


2.4. Red de tareas

Especificaremos las tareas a realizar en el proceso mediante una *Red de Tareas*. Definimos una *Tarea* como un paso a ser realizado en nuestro proceso, con entradas, transformaciones y salidas. Una tarea posee un nombre que indica una acción. Involucra una o más entradas, consistentes en datos o consultas sobre datos. Las transformaciones consisten en procesos de manipulación de datos que se valen de las entradas para transformarlas en salidas. Las salidas, así como las entradas, deben tener una ubicación en una estructura de datos.

2.4.1. Esquema en Nodo

Las Tareas se identifican y organizan en un *Esquema en Nodo* como el siguiente. Un Esquema en Nodo es simplemente un grafo dirigido cuyos nodos son las tareas, conectadas por arcos que implican precedencia.



Tarea “Unificar Clientes”: Consiste en unificar los registros de los clientes mayoristas y los clientes minoristas en una tabla de la Staging Area

Tarea B: Describir en qué consiste la tarea B

Tarea C: Describir en qué consiste la tarea C

...

...

...

2.4.2. Matriz de Precedencias

El Esquema en Nodo se complementa con una Matriz de Precedencias, donde se indica para cada una de las Tareas cuáles deben estar finalizadas antes de la ejecución de la siguiente.

Tarea	Precedencia
Unificar clientes	
B	Unificar clientes
C	B
D	Unificar clientes
E	D
F	C,E
G	F

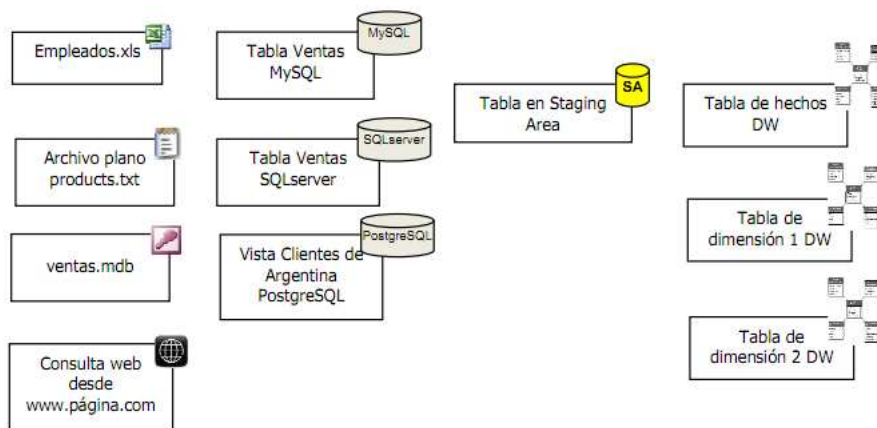
2.5. Diseño

2.5.1. Herramientas por cada tarea de la red de proceso

Esquema de proceso



En el esquema del proceso se identificarán los objetos intervinientes con íconos específicos según su tipo, por ejemplo:



Miniespecificación del Proceso

Tarea: Unificar clientes Mayoristas y Minoristas

Entrada: Tabla Clientes Mayoristas
Tabla Clientes Minoristas

Transformaciones: (Select idCliente, nombre from Minoristas)
union
(Select idCliente, nombre from Mayoristas)

Salida: Se cargaron los campos *idCliente* y *nombre* de la tabla *Clientes* de la Staging Area.

2.5.2. Pantallas de análisis por cada requerimiento

Req 1. Cantidad de litros consumidos y de productos adquiridos por cliente en el tiempo

<i>Producto</i>	▼
<i>Vendedor</i>	▼

		<i>Cliente A</i>	<i>Cliente B</i>	<i>Cliente C</i>	...
2007		
	<i>Ene</i>	
	<i>Feb</i>	
	<i>Mar</i>	
	...				
2008		
2009		
...					

La *Pantalla de Análisis* es un bosquejo de salida en pantalla, desprendido de los detalles de diseño, de tal forma de poder disponer los datos de la forma en que van a aparecer una vez codificados.

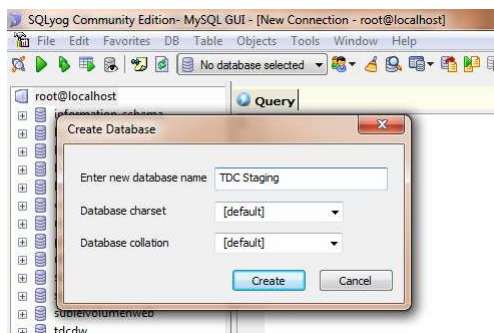
Capítulo 3

Extracción de Datos - Transformaciones básicas

3.1. Crear la Base de Datos de la Staging Area.

Un Staging Area es un área de almacenamiento intermedio, que nos permitirá una mejor manipulación de los datos a la hora de realizar los procesos ETL subsiguientes.

1. Ingresar a la herramienta de administración del servidor MySQL.
2. Autenticarse indicando el servidor (en nuestro caso *localhost*) el nombre de usuario y contraseña.
3. Crear la base de datos del Staging Area. Hacer clic con el botón derecho sobre el nodo raíz (servidor) → create database. Le daremos el nombre *TDC Staging*. Clic en create.



4. Como cada una de las fuentes de datos se migrarán inicialmente a una tabla dentro de la Staging Area, procedemos a crear las mismas. En la ventana de consulta ingresamos las siguientes sentencias:

```

CREATE TABLE 'mayoristas' (
  'nombre' varchar(150) default NULL,
  'cod_cliente' int(11) NOT NULL,
  PRIMARY KEY ('cod_cliente')
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE 'minoristas' (
  'cod_cliente' int(11) NOT NULL,
  'nombre' varchar(150) default NULL,
  PRIMARY KEY ('cod_cliente')
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE 'productos' (
  'cod_producto' int(10) NOT NULL,
  'descripcion' varchar(150) default NULL,
  'presentacion' varchar(100) default NULL,
  PRIMARY KEY ('cod_producto')
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE 'ventas' (
  'cod_venta' int(10) NOT NULL,
  'region' varchar(45) default NULL,
  'cod_sucursal' int(10) default NULL,
  'fecha' datetime default NULL,
  'cod_cliente' smallint(5) default NULL,
  'cod_empleado' smallint(5) default NULL,
  PRIMARY KEY ('cod_venta')
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE 'detalle_ventas' (
  'cod_venta' int(10) NOT NULL,
  'cod_producto' smallint(5) NOT NULL,
  'cantidad' smallint(5) default NULL,
  PRIMARY KEY ('cod_venta', 'cod_producto')
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

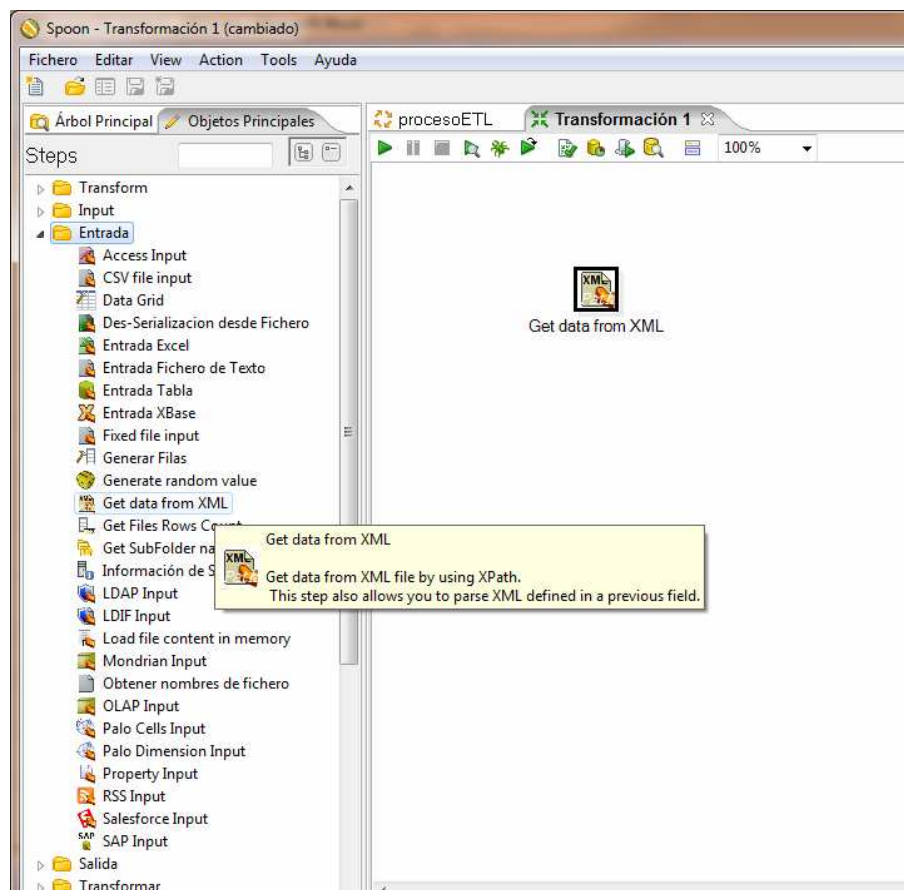
Seleccionar todas las sentencias y ejecutarlas presionando F9.

3.2. Cargar los datos desde las Fuentes de Datos a la Staging Area

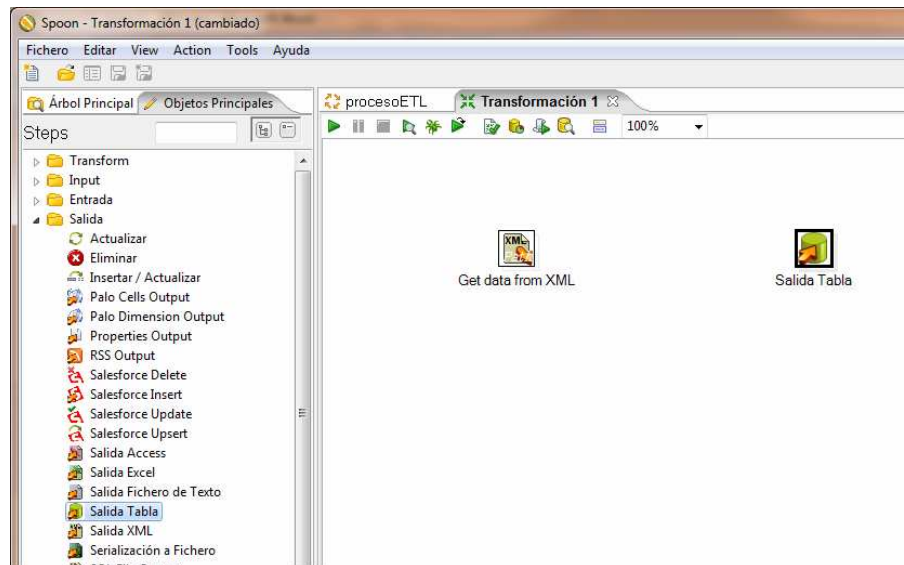
En PDI, crearemos las transformaciones para mover y transformar filas entre una fuente y un destino. Todas las acciones que se ubiquen en una transformación se ejecutarán simultáneamente. Para armar una ruta controlando el flujo de las acciones, utilizaremos los trabajos, los cuales nos permiten ejecutar transformaciones, enviar correo electrónico en caso de error, enviar archivos por FTP, etc. Mediante un trabajo

definiremos la secuencia de transformaciones que se ejecutarán durante nuestro proceso ETL.

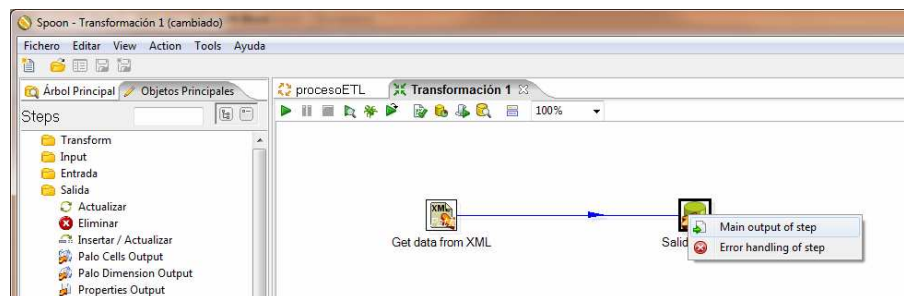
1. Abrir PDI, mediante el ejecutable spoon.bat.
2. Crear un nuevo trabajo, el cual definirá la ruta del proceso ETL. Fichero → Nuevo → Trabajo. En este trabajo se incluirán posteriormente todas las transformaciones, y las condiciones bajo las cuales se deberá ejecutar cada una. Guardar el trabajo mediante el menú Fichero → Guardar como. Llamaremos al archivo *procesoETL.kjb*.
3. Crear la transformación, en la cual se migrarán los datos desde las Fuentes de Datos a la Staging Area. Fichero → Nuevo → Transformación.
4. Migrar los clientes mayoristas a Staging.
 - a) Desde el panel izquierdo click en la solapa Objetos principales. Seleccionar la categoría Entrada → Get data from XML. Arrastrar el elemento al área de trabajo. En PDI, cada uno de estos elementos se denomina *paso*.



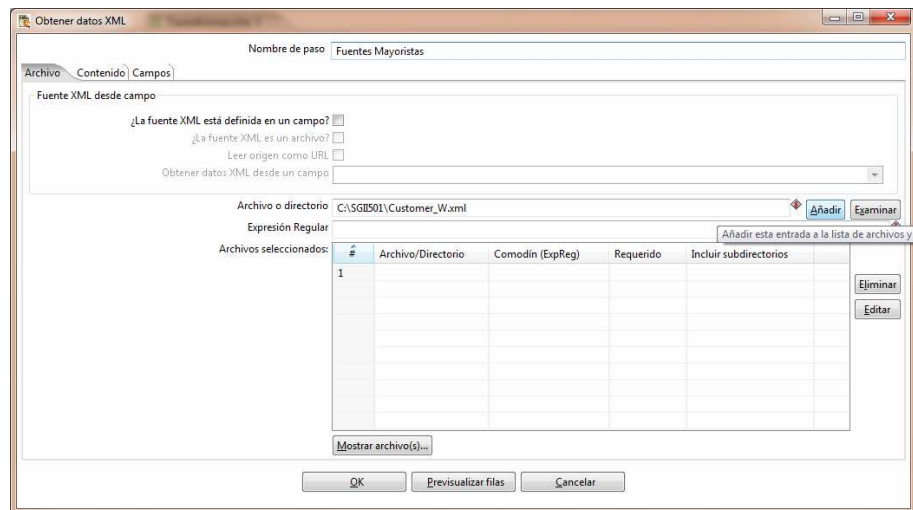
- b) Especificar el destino de los datos. Click en la solapa Objetos principales. Seleccionar la categoría Salidas → Salida tabla.



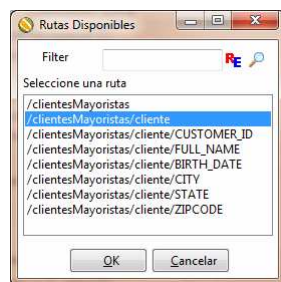
- c) La conexión entre dos pasos se denomina *salto*. Mediante un salto, indicaremos que los datos contenidos en el XML serán la información de entrada a la tabla. Para ello, seleccionar el icono Get data from XML, y manteniendo presionada la tecla “shift”, arrastrar hasta conectar con el icono Salida Tabla. Seleccionar la opción Main output of step.



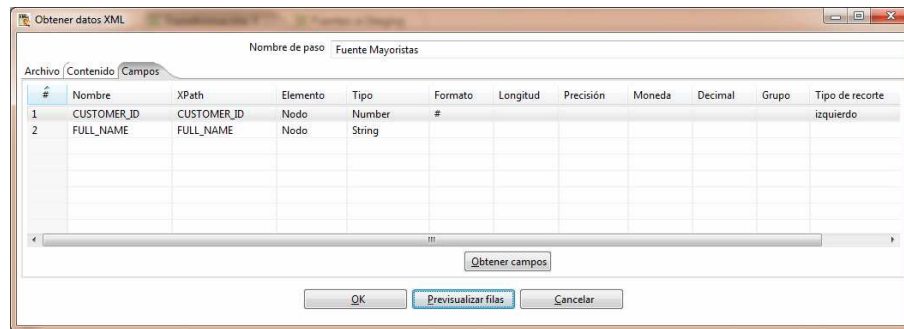
- d) Una vez realizada la conexión, establecer las propiedades de cada elemento. Configurar primero la entrada (archivo XML), haciendo doble clic sobre el mismo. En nombre del paso ingresamos *Fuentes Mayoristas*. En la solapa Archivo, indicar el archivo de origen *c:\SGII501\customer_w.xml*, y presionar Añadir.



- e) En la solapa Contenido, presionar el botón **Obtener nodos XPath** para especificar los tags que identifican las filas (es decir, cada uno de los clientes). Seleccionar *clientesMayoristas/cliente*.

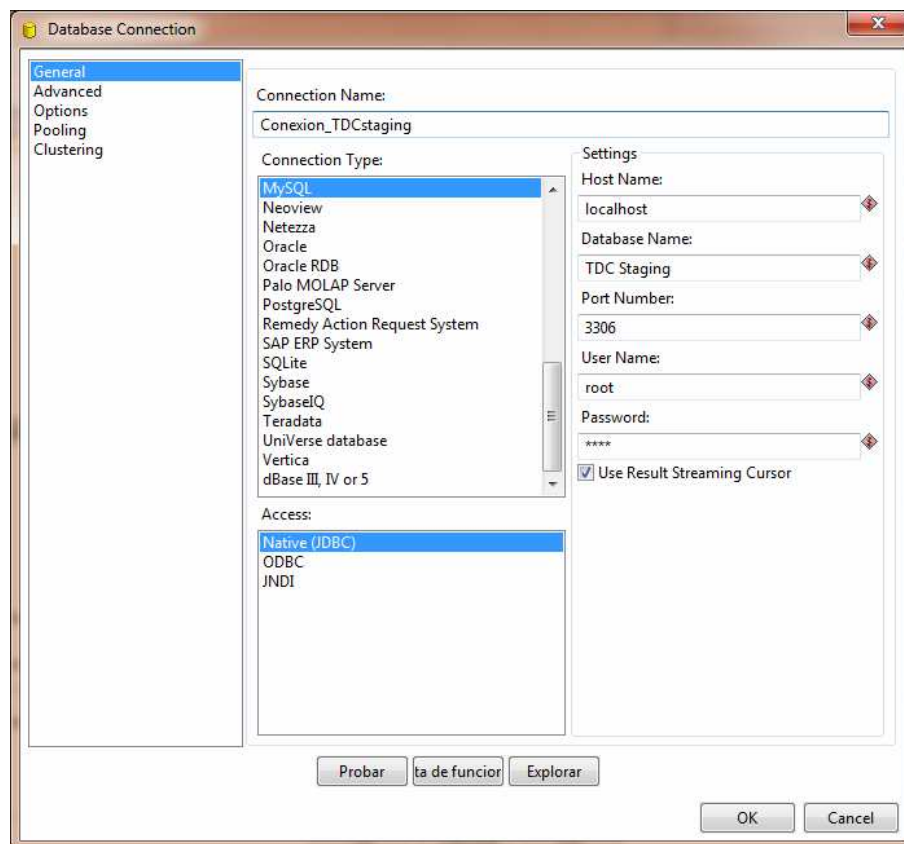


- f) En la solapa Campos, presionar el botón **Obtener campos**. Como en este momento nos interesan sólo los campos *customer_id* y *full_name*, eliminamos los restantes de la lista.
- g) Para el campo *customer_id*, especificar *tipo de recorte* = *izquierdo* (lo cual elimina espacios en blanco antes del campo) y *formato* = *#* (para indicar que es un entero). Verificar la configuración presionando el botón **Previsualizar Filas**.



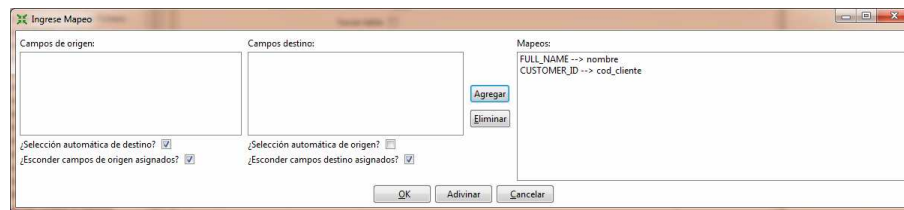
Clic en OK. La entrada de datos ya está configurada.

- h) Configurar ahora el destino haciendo doble clic sobre el mismo. Nombraremos a este paso *Mayoristas Staging*. Como la tabla de destino está en la base de datos *TDC Staging*, creamos una conexión a la misma presionando el botón **Nuevo**. Establecer los parámetros de la conexión con la base de datos TDC Staging (servidor, usuario, contraseña, puerto) y asignar el nombre *conexion_TDCstaging*. Clic en **Probar** para verificar que la conexión sea exitosa.



Clic en OK. La conexión con la base de datos *TDC Staging* fue creada.

- i) Para que la conexión pueda ser utilizada en otras transformaciones y en el trabajo principal, una vez creada la misma, dirigirse a la solapa **Arbol principal** → **Conexión a bases de datos**. Clic derecho sobre la conexión, y seleccionar **compartir**. La conexión compartida se mostrará en **negrita**.
- j) En el campo **Tabla destino** ingresar *mayoristas*. Marcar el atributo **Specify database fields**. Esto nos permitirá ingresar manualmente el mapeo entre los atributos de entrada y los de salida.
- k) En la solapa **Database fields** clic en **Enter field mapping**. Aquí seleccionaremos el campo de origen junto al campo de destino y presionamos **Agregar**.



- l) Una vez mapeados los campos, clic en **OK**. Clic en **OK** nuevamente para terminar la configuración del paso.
5. Migración de clientes minoristas a Staging. Repetir los mismos pasos que para los clientes mayoristas, seleccionando el archivo *customer_R.xml* como origen, y la tabla *minoristas* de la Staging Area como destino (para conectarse seleccionar la conexión *conexion_TDCstaging* creada en el paso anterior).
 6. Migración de productos a la Staging Area.
 - a) En el **Panel de objetos principales**, sección **entrada**, seleccionar **CSV File input**. En la sección **Salida**, seleccionar el objeto **Salida de Tabla**. Conectar la entrada con la salida manteniendo presionada la tecla “shift”.
 - b) Configurar la entrada haciendo doble clic sobre la misma. Especificar el nombre del paso como *Fuente Productos*. Indicar la ruta del archivo *C:\SGII501\Products.txt*. Ingresar como delimitador de campos el símbolo *pipe* (|). Desmarcar la opción **Fila de encabezado presente**, ya que el archivo de productos no contiene los nombres de las columnas. Presionar el botón **Recuperar campos**. Vemos que por defecto se coloca el nombre *Field_x* a cada columna. Modificar los mismos con *cod_producto*, *Descripcion* y *Presentacion*. Especificar que el primer campo es un entero seleccionando **Integer**, formato **#**, con tipo de recorte **izquierda**. Presionar **Previsualizar** para verificar la configuración realizada.

Nombre del paso: Fuente Productos

Nombre de archivo: C:\SGI501\Products.txt

Delimitador: |

Cierre de campo: "

Tamaño del buffer NIO: 50000

¿Conversión perezosa? ☒

¿Fila encabezado presente? ☐

Añadir archivos a resultado ☐

El campo conteniendo el número de línea:

¿Ejecutando en paralelo? ☐

Codificación de archivo:

#	Nombre	Tipo	Formato	Longitud	Precisión	Moneda	Decimal	Grupo	Tipo de recorte
1	cod_producto	Integer	#	17		\$,	.	izquierda
2	Descripcion	String		31		\$,	.	ninguno
3	Presentacion	String		11		\$,	.	ninguno

OK Traer Campos Previsualizar Cancelar

- c) Configurar la salida, que será la tabla *productos* de nuestra base de datos *TDC Staging Area*. Doble clic sobre la misma, se abre la ventana de propiedades. Nombrar al paso como *Productos Staging*. Especificar la conexión con la base de datos staging *conexion_TDCstaging* y el nombre de la tabla destino *productos*. Desmarcar la opción *Specify database fields* y presionando el botón **Enter field mapping** realizar el mapeo de los campos de origen con los de destino.

Nombre de paso: Productos Staging

Conexión: Conexion_TDCstaging

Esquema destino:

Tabla destino: productos

Tamaño transacción (commit): 1000

Vaciar tabla ☐

Ignorar errores de inserción ☐

Specify database fields ☐

Main options Database fields

Fields to insert:

Ingreso Mapeo

Campos de origen: Campos destino: Mapeos:

Descripcion --> descripcion

Presentacion --> presentacion

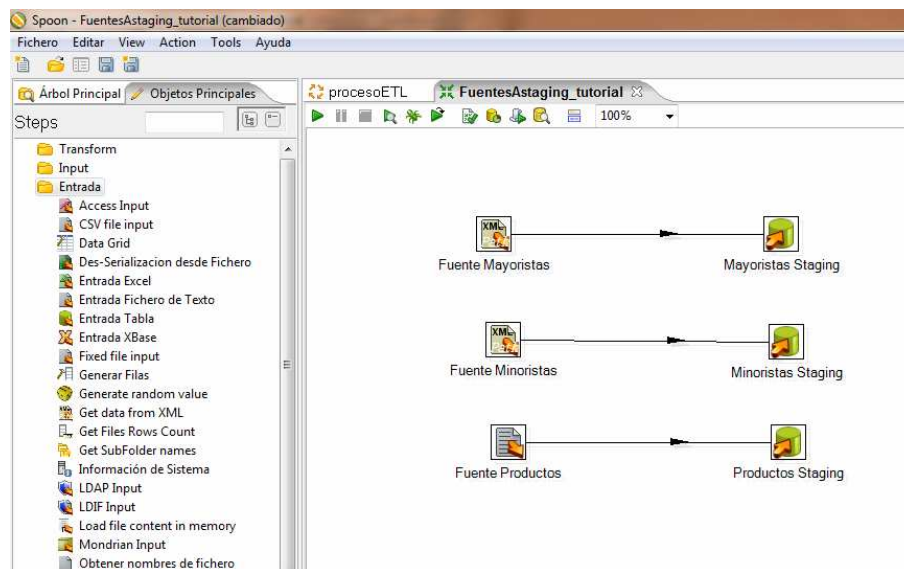
cod_producto --> cod_producto

¿Selección automática de destino? ¿acción automática de origen?

¿Esconder campos de origen asignados? campos destino asignados?

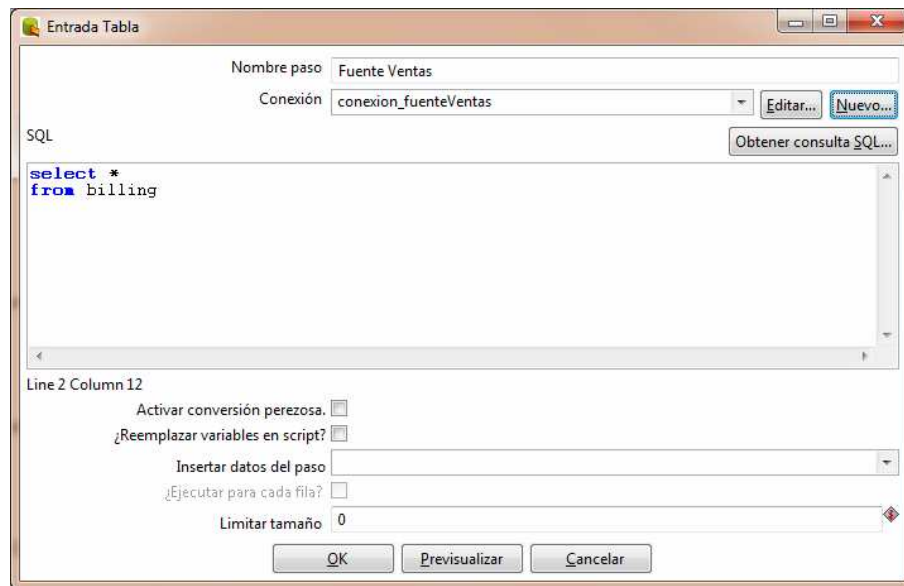
OK Adivinar Cancelar

Clic en OK para cerrar el mapeo. Clic en OK para terminar la configuración de la salida.



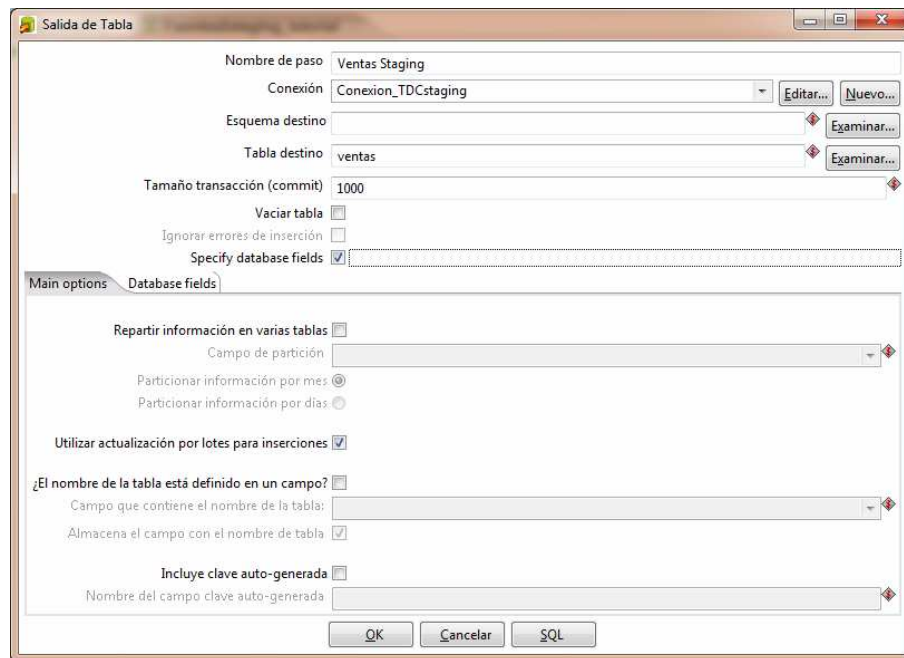
7. Migración de Ventas a la Staging Area.

- Para migrar las ventas a la base de datos *TDC Staging*, seleccionaremos en el Panel de objetos principales → Entrada → Entrada tabla. Luego, seleccionar en la sección Salida → Salida tabla. Unir la entrada con la salida manteniendo presionada la tecla “shift”.
- Doble clic sobre el icono de la tabla para acceder a su configuración. En nombre del paso especificar *Fuente Ventas*. En conexión crear una conexión a la base de datos *Sales*, la cual contiene toda la información de las ventas. Clic en Nuevo. Aquí especificamos que queremos conectarnos a una base de datos MySQL. Asignar el nombre a la conexión *conexion_FuenteVentas*. Especificar servidor, base de datos *Sales*, usuario y contraseña. Clic en Probar para verificar la conexión.
- Una vez creada la conexión, especificaremos mediante una consulta SQL la selección de todos los campos y registros de la tabla *billing*:



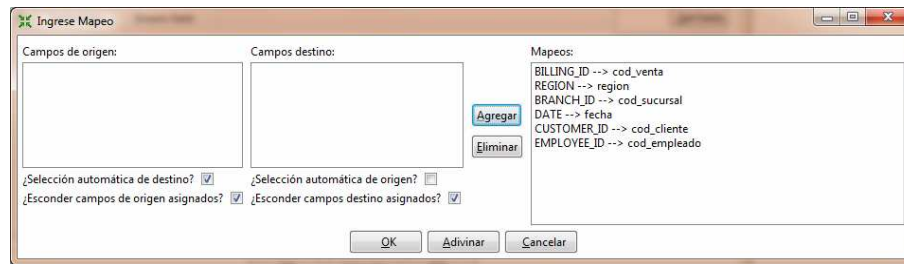
Clic en OK

- d) Configurar la salida haciendo doble clic sobre la misma. Nombrar a la misma *Ventas Staging*, especificamos que la conexión es *conexion_TDCstaging*, la tabla destino es *Ventas* y marcar la opción *Specify database fields*.



- e) En la solapa *Database fields* presionamos el botón *Enter fields mapping*. Realizamos el

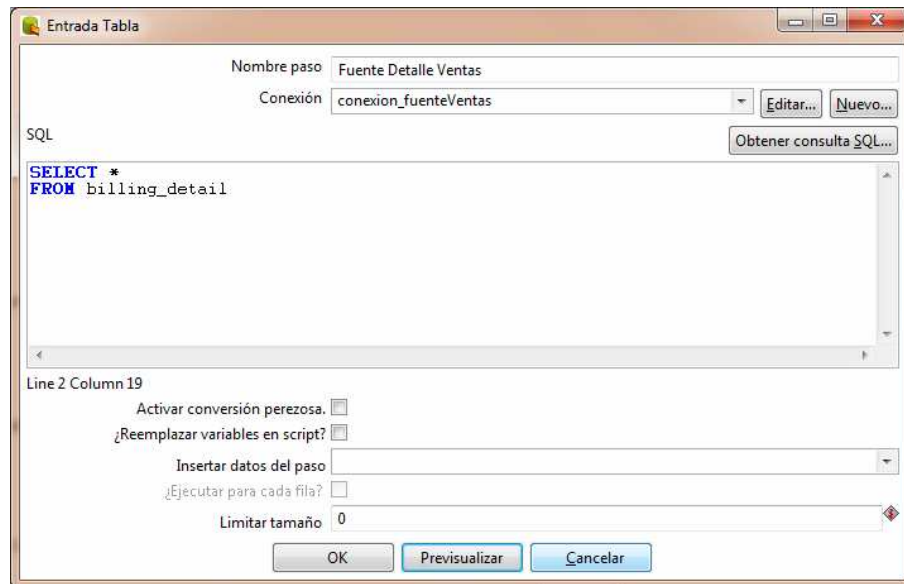
mapeo entre los campos de origen y destino. Al finalizar presionamos OK.



Clic en OK para cerrar la configuración.

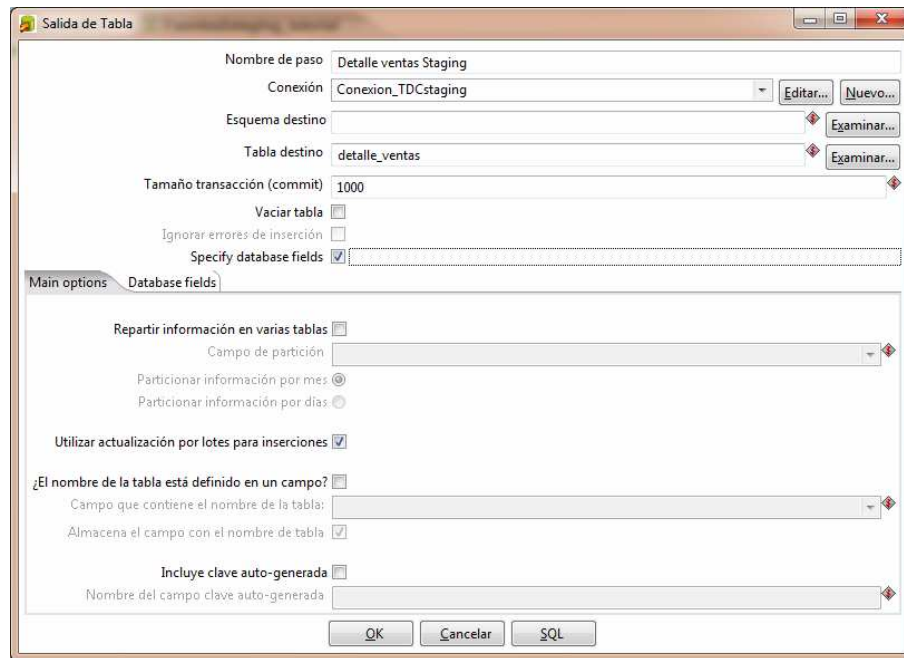
8. De la misma forma que migramos los encabezados de las ventas, migraremos las líneas de ventas.

- a) Seleccionamos del Panel de objetos principales → Entrada → Entrada tabla. Luego seleccionamos Salida → Salida tabla. Unimos ambos mediante la tecla “shift”.
- b) Configurar la entrada asignándole el nombre *Fuente Detalle Ventas*, conexión *conexion_fuenteVentas*, y en SQL ingresamos una consulta que recupere todos los registros y todos los campos:



Clic en OK para cerrar la configuración.

- c) Configurar la salida asignándole el nombre *Detalle ventas staging*. En conexión seleccionamos *conexion_TDCStaging*, en tabla especificamos *detalle_ventas*, y marcamos la casilla Specify database fields.

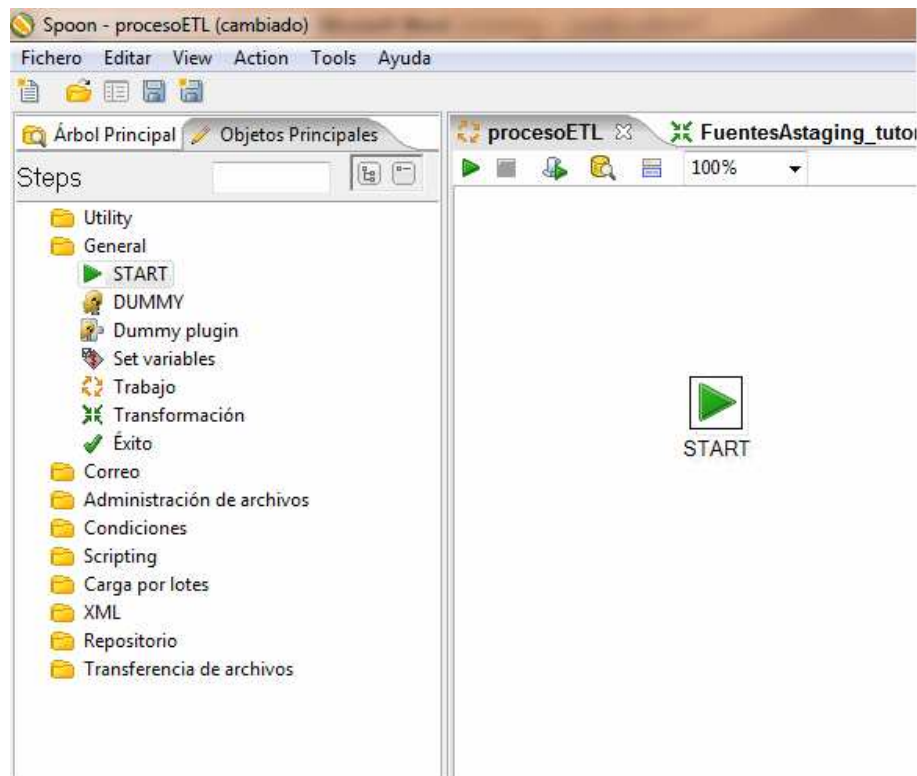


- d) En la solapa Database fields, presionamos el botón Enter field mapping para indicar el mapeo entre los campos de origen y destino. Una vez finalizado hacemos clic en OK.



9. Ya tenemos armada la transformación que migra los datos desde las Fuentes de Datos a la base de datos TDC Staging. Para guardar la misma, dirigirse al menú Fichero → Guardar como..., y en nombre le asignamos *tr_Fuentes_a_staging.ktr*. Clic en Guardar, clic en OK. Ahora incluiremos esta transformación en el trabajo que creamos en el paso 2 en la página 14, el cual representa nuestro proceso ETL completo.

- a) Abrir el archivo de trabajo *procesoETL.kjb*. Dirigirse a la solapa Objetos principales. Seleccionar General → Start.



- b) Seleccionar Scripting → SQL. Doble clic sobre el mismo para configurarlo. En este objeto SQL colocaremos las sentencias que limpiarán nuestra base de datos *TDC Staging* al comienzo del proceso ETL. En nombre ingresamos *Limpiar Staging*, en conexión seleccionamos *conexion_TDCStaging* y en SQL la sentencia:

```
delete from minoristas;
delete from mayoristas;
delete from productos;
delete from ventas;
delete from detalle_ventas;
```

Clic en OK¹

- c) Definimos el flujo de trabajo uniendo el objeto **Start** con el objeto **SQL**, manteniendo presionada la tecla “shift”. Esta ruta se denomina *salto*.

¹Si la conexión con la base de datos TDC Staging no aparece en el listado, cerrar el archivo del trabajo, y abrirlo nuevamente para refrescar el listado de conexiones compartidas (como precondition, la transformación donde reside la conexión que queremos compartir ya debió ser guardada).



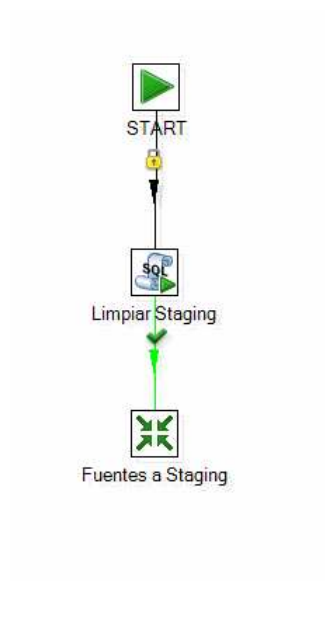
En cada salto, podremos ver un dibujo que nos indica la condición por la cual se pasa al próximo paso (Incondicional, positiva o negativa) ²

Podemos cambiar la condicionalidad para la ejecución de un salto haciendo clic sobre el ícono que lo identifica (salvo la ruta que va desde **Start** hasta la primer tarea que solo puede ser incondicional).

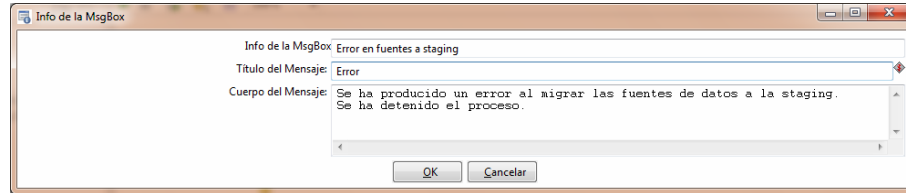
- d) Dirigirse dentro del panel de Objetos principales a General → Transformación. Hacemos doble clic sobre el mismo para especificar sus propiedades. Le asignamos el nombre *Fuentes a Staging*, y en la ruta del archivo especificamos *c:|SGII501|fuentes_a_staging.ktr*. Clic en OK.
- e) Mediante un salto por condición positiva, indicaremos que si la limpieza de la staging se ha realizado correctamente, luego se ejecutará la transformación *Fuentes a Staging*. Para ello seleccionamos el objeto **SQL**, presionamos la tecla “shift” y unimos la flecha con la transformación *Fuentes a Staging*.

2

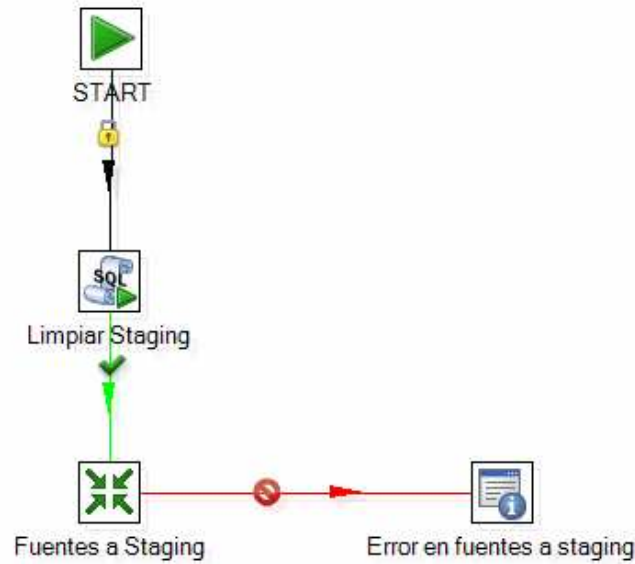
- Un candado nos indica incondicionalidad de ejecución, es decir, independientemente de que ocurran errores o no se seguirá la ruta del proceso (Salto incondicional).
- Un símbolo de correcto nos indica que se seguirá la ruta si no ocurrieron errores (Salto por condición positiva).
- Un símbolo incorrecto nos indica que se seguirá la ruta si se produce un error (Salto por condición negativa).



- f) Agregar desde nuestro panel de objetos, Utility → Mostrar información en caja de texto. Doble clic sobre la misma para configurarla. En Info de la MsgBox ingresar *Error en fuentes a staging*, en título *Error* y en cuerpo del mensaje una pequeña descripción que indique la ocurrencia de un error al migrar las fuentes de datos a la staging.



- g) Unimos la transformación *Fuentes a Staging* con la caja de texto manteniendo presionada la tecla “shift”.
- h) Determinamos el salto por condición negativa, con el símbolo *Incorrecto* (haciendo clic sobre el dibujo que identifica su tipo), para que se muestre el mensaje sólo si se produce un error en la transformación.



- i) Presionar el botón Play de la barra de herramientas para correr el proceso. Aparecerá un cuadro de diálogo en el que presionamos el botón **Ejecutar**.

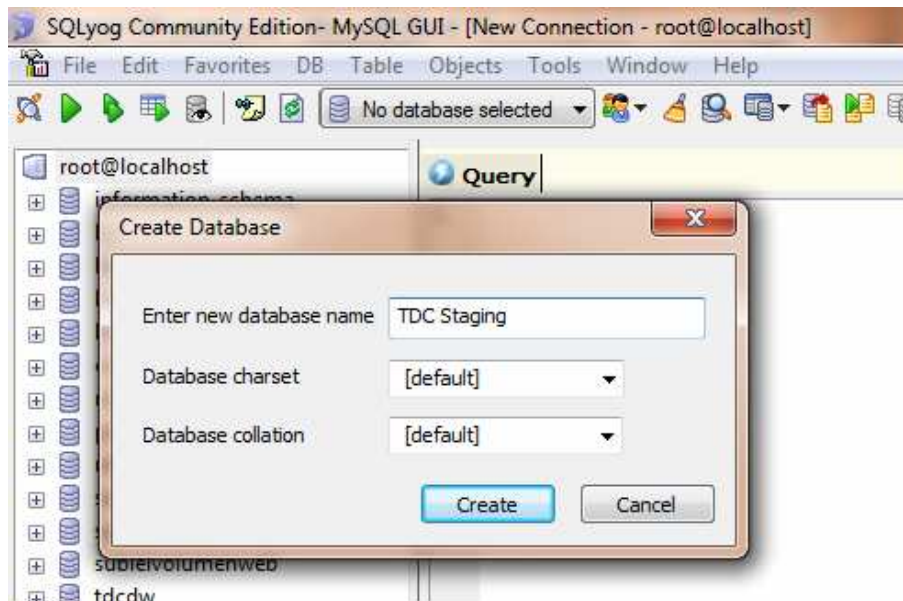
La primera parte de nuestro proceso ETL ha sido construida. Mediante la solapa **Logging** del panel inferior podremos consultar toda la información sobre la ejecución del proceso.

Capítulo 4

Transformaciones avanzadas: Construcción del Data Warehouse

4.1. Creación de la base de datos Data Warehouse

1. Ingresar a la herramienta de administración del servidor MySQL.
2. Autenticarse indicando el servidor, nombre de usuario y contraseña.
3. Una vez que hayamos ingresado, podremos observar a la izquierda un panel con las bases de datos que residen en nuestro servidor. Para crear nuestra base de datos DW, clic con el botón derecho sobre el nodo raíz (servidor) → `create database`. Le daremos el nombre *TDC DW*. Clic en `Create`.
4. Como hemos visto en la sección 1.1 en la página 3, nuestra base de datos DW estará formada por dos tablas:



En la tabla Ventas, el campo “Cantidad” representa la cantidad de productos adquiridos. Para crear las tablas, en la ventana de consulta o Query ingresamos las sentencias:

```
CREATE TABLE 'clientes' (
  'cod_cliente' int(11) NOT NULL,
  'nombre' varchar(150) default NULL,
  PRIMARY KEY ('cod_cliente')
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE 'ventas' (
  'cod_cliente' int(11) NOT NULL,
  'cod_tiempo' date NOT NULL,
  'litros' float default NULL,
  'cantidad' bigint(20) default NULL,
  PRIMARY KEY ('cod_cliente', 'cod_tiempo'),
  CONSTRAINT 'FK_ventas' FOREIGN KEY ('cod_cliente')
  REFERENCES 'clientes' ('cod_cliente')
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Seleccionamos todas las sentencias y presionamos F9 para ejecutarlas.

4.2. Realizar los procesos de extracción, transformación y carga (ETL)

En este paso, desarrollaremos las transformaciones necesarias para

- Cargar la tabla *Clientes* del Data Warehouse

- Cargar la tabla *Ventas* del Data Warehouse. Para lograr este punto, crearemos primero una tabla auxiliar en la base de datos de la Staging Area (que llamaremos *StagingVentas*). Esta tabla intermedia facilitará el proceso de transformación de los datos previamente a la carga de los mismos en el Data Warehouse. La tabla estará formada por los siguientes campos:

<u>StagingVentas</u>
Fecha (date)
<u>Cod_cliente (int)</u>
<u>LitrosEnvase (float)</u>
Cantidad (int)

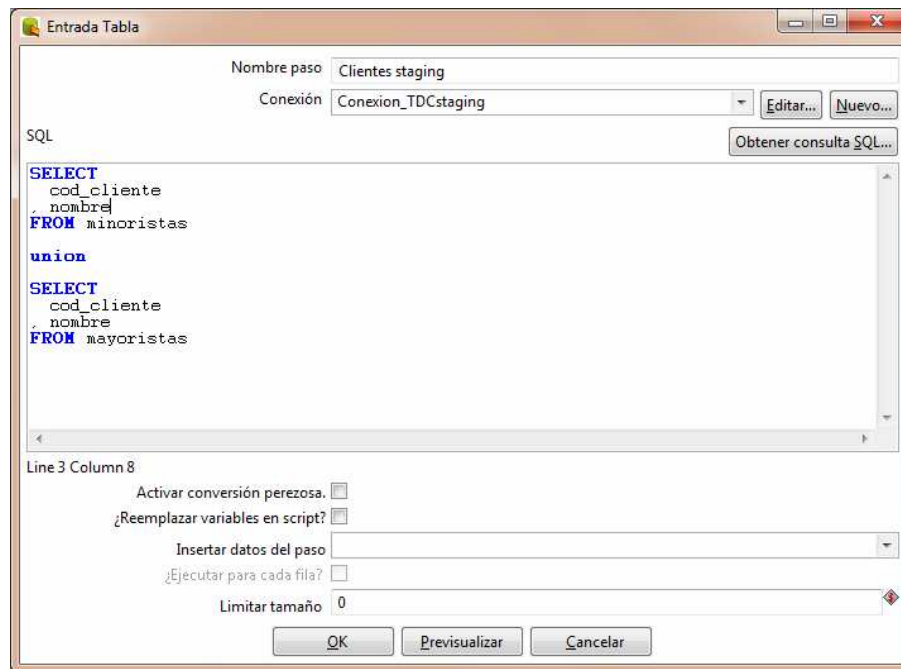
Por lo tanto, el proceso de carga de la tabla *Ventas* del Data Warehouse será dividido en dos pasos:

- Crear la transformación para cargar una nueva tabla en la Staging Area (*StagingVentas*).
- Crear la transformación para cargar los datos desde la tabla *StagingVentas* a la tabla *Ventas* del Data Warehouse

1. Cargar los clientes al Data Warehouse

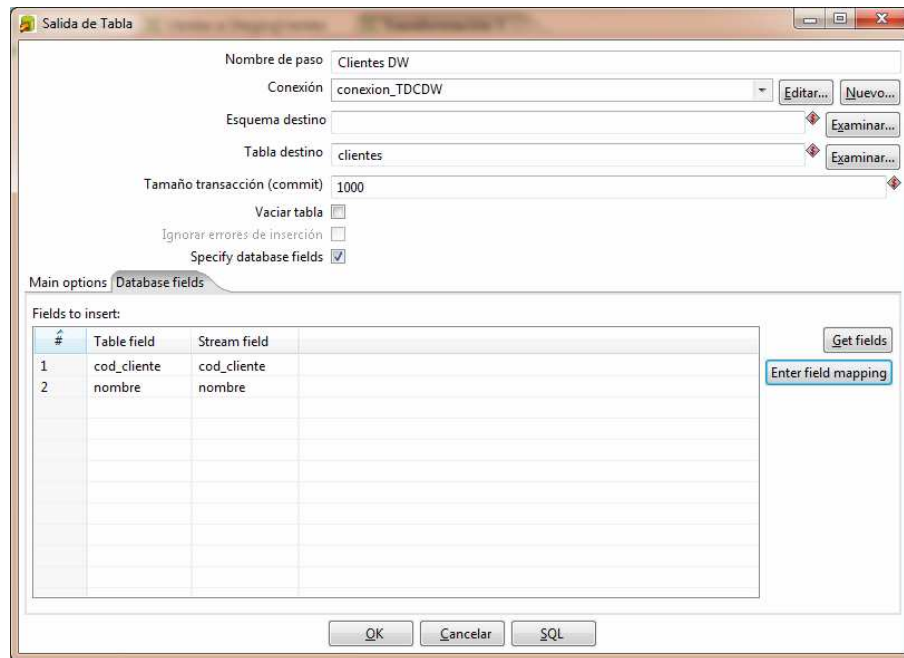
- a) En PDI, vamos a crear una nueva transformación para unificar los clientes y guardarlos en el Data Warehouse. Ir al menú Fichero → Nuevo → Transformación.
- b) En la solapa de Objetos Principales, seleccionar Entrada → Entrada tabla. Luego, seleccionar en la sección Salida → Salida tabla. Unir la entrada con la salida manteniendo presionada la tecla “shift”.
- c) Doble clic sobre el ícono de entrada para acceder a su configuración. Asignar al paso el nombre *Clientes Staging*. En conexión seleccionar *conexión_TDCStaging* y en la sección SQL especificar la unión de los clientes mayoristas con los minoristas mediante la sentencia:

```
SELECT cod_cliente , nombre FROM minoristas
union
SELECT cod_cliente , nombre FROM mayoristas
```



Clic en OK

- d) Doble clic sobre la salida para acceder a su configuración. Llamaremos al paso *Clientes DW*. Como el destino de los datos es la base de datos del Data Warehouse, debemos crear una conexión a la misma presionando el botón **Nuevo**. Asignamos a la conexión el nombre *conexion_TDCDW*, tipo de conexión *MySQL*, servidor *localhost*, nombre de la base de datos *TDC DW*, usuario y contraseña *root*. Clic en **OK** para crear la conexión.
- e) En tabla destino, especificamos la tabla *Clientes*. Marcamos la opción **Specify database fields**. En la solapa **Database fields**, clic en el botón **Enter field mapping**. Especificamos el mapeo entre los campos de origen y de destino. Clic en **OK**.



Clic en OK para finalizar la configuración de la salida.

- f) Guardamos la transformación en la carpeta de trabajo (*C:\SGII501*) mediante el menú Fichero → Guardar como. Le asignamos el nombre *tr_clientes_a_DW*. Clic en Guardar.

2. Cargar las Ventas al Data Warehouse: Carga de la tabla *StagingVentas*

- a) Primero creamos la tabla *StagingVentas* en nuestra base de datos TDC Staging. En la herramienta de administración del servidor MySQL, en la ventana de consultas o Query, ejecutamos la siguiente sentencia:

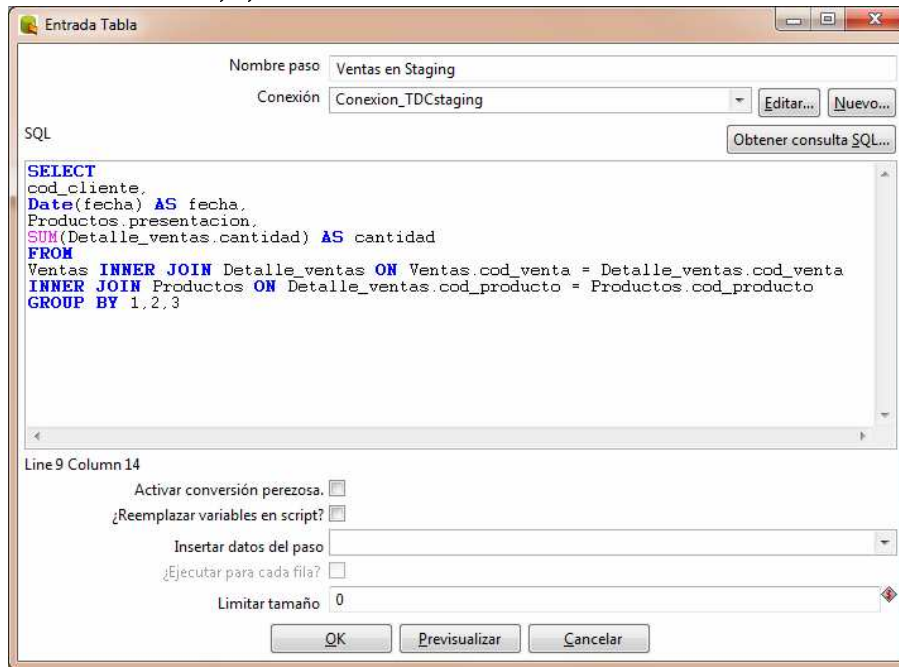
```
CREATE TABLE 'stagingventas' (
  'fecha' date NOT NULL,
  'cod_cliente' int(11) NOT NULL,
  'LitrosEnvase' float NOT NULL,
  'cantidad' int(11) default NULL,
  PRIMARY KEY ('fecha','cod_cliente','LitrosEnvase')
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

- b) En PDI, vamos al menú Fichero → Nuevo → Transformación. En esta transformación, totalizaremos las ventas por cliente / fecha / presentación (o envase). Creamos desde el panel de Objetos principales una Entrada de tabla. Doble clic sobre la misma para configurarla. En nombre del paso, colocamos *Ventas Staging*. En conexión seleccionamos *conexion_TDCStaging*, y en SQL ingresamos la siguiente sentencia:

```

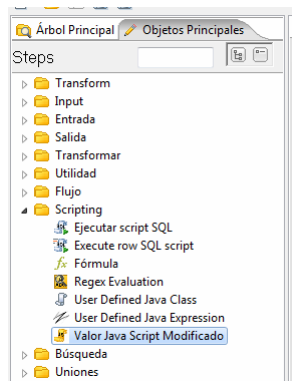
SELECT
    cod_cliente, Date(fecha) AS fecha,
    Productos.presentacion,
    SUM(Detalle_ventas.cantidad) AS cantidad
FROM
    Ventas
    INNER JOIN Detalle_ventas ON
        Ventas.cod_venta = Detalle_ventas.cod_venta
    INNER JOIN Productos ON
        Detalle_ventas.cod_producto = Productos.cod_producto
GROUP BY 1,2,3

```

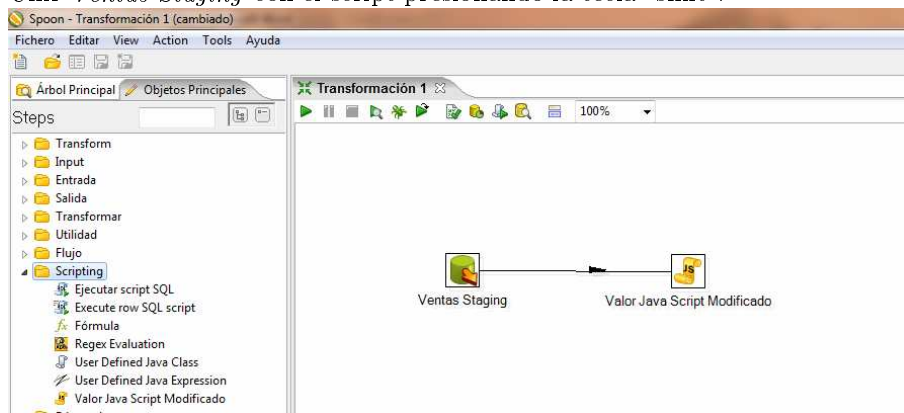


Clic en Previsualizar para verificar que la consulta se ejecute exitosamente.

- c) En la tabla *Productos* de la Staging Area, el campo *Presentación* (que representa el envase del mismo) se encuentra en formato de texto o cadena. Sin embargo, en la tabla *StagingVentas* debemos almacenarlo en formato numérico, ya que necesitamos totalizar la cantidad de litros. Para transformar el campo *Presentación* al tipo numérico, ejecutaremos un script seleccionando del panel de Objetos principales → Scripting → Valor Java Script Modificado.



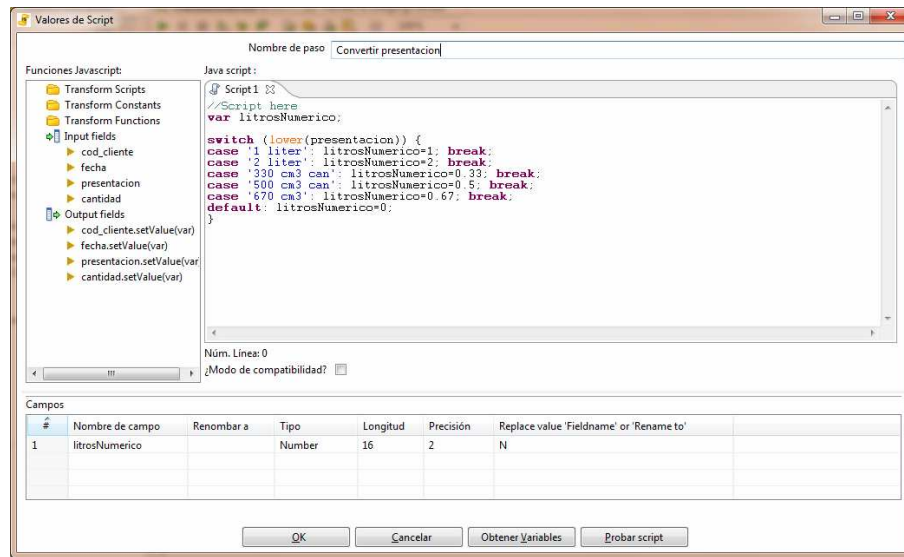
Unir *Ventas Staging* con el script presionando la tecla “shift”.



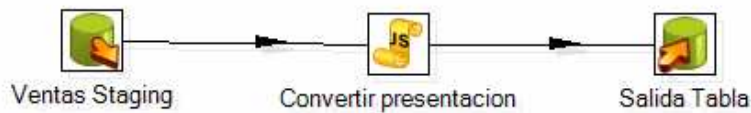
Doble clic sobre el icono del script para acceder a su configuración. Llamaremos al script *Convertir presentación*. Podemos ver en el panel izquierdo de la pantalla las variables de entrada al script (en este caso, los campos provenientes de la entrada *Ventas Staging*) y las de salida. El código del script para convertir la presentación a formato numérico es el siguiente:

```
var litrosNumerico;
switch (lower(presentacion)) {
case '1 liter': litrosNumerico=1; break;
case '2 liter': litrosNumerico=2; break;
case '330 cm3 can': litrosNumerico=0.33; break;
case '500 cm3 can': litrosNumerico=0.5; break;
case '670 cm3': litrosNumerico=0.67; break;
default: litrosNumerico=0; }
```

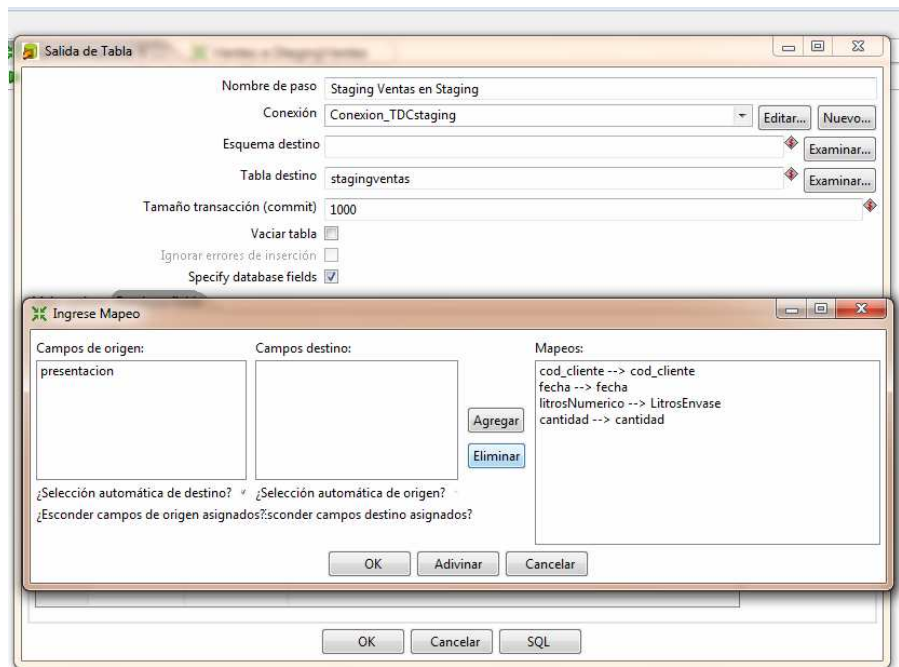
Presionamos luego el botón *Obtener variables*. De esta forma, todas las variables que hayamos definido en nuestro código (en este caso la variable *litrosNumérico*) formarán parte de la salida del script, junto con los campos de entrada originales. Clic en OK.



- d) Seleccionamos del panel de Objetos principales → Salida de Tabla. Unimos el script *Convertir Presentación* con la salida de tabla presionando la tecla “shift”.

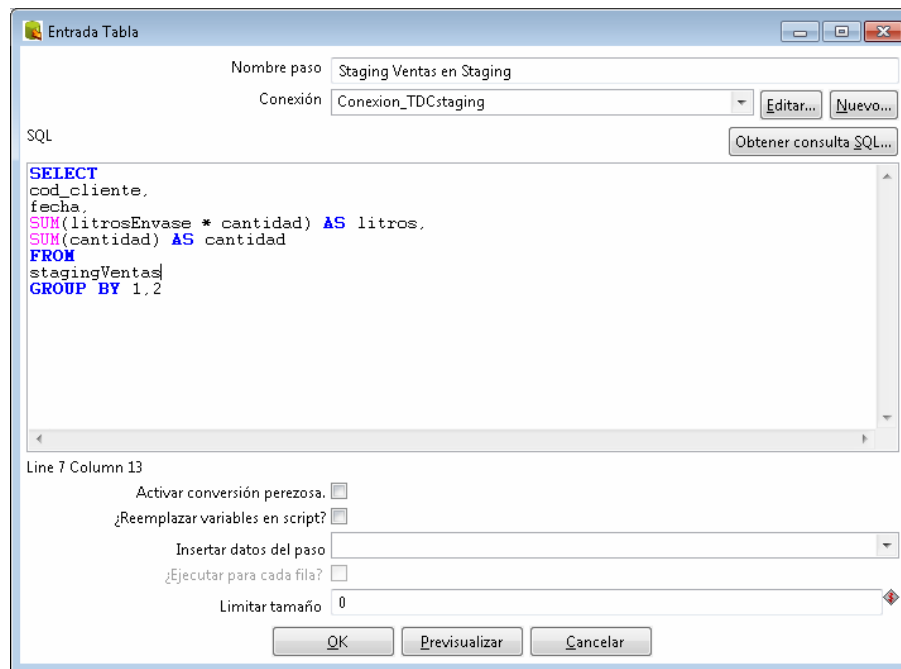


- e) Configuraremos la salida de tabla haciendo doble clic sobre la misma. Le asignamos el nombre *Staging Ventas en Staging*, seleccionamos la conexión *conexion_TDCstaging*, en tabla especificamos *stagingVentas*. Marcamos la opción **Specify database fields**. En la solapa **Database Fields**, Clic en el botón **Enter field mapping**. Mapeamos los campos de origen con los de destino. El campo *litrosNumerico* (calculado en el script) debe ser mapeado al campo *litrosEnvase* de la tabla destino. El campo de origen *Presentación* quedará sin ser mapeado, ya que fue reemplazado por *litrosNumerico*. Clic en OK. Clic en OK para cerrar la configuración.



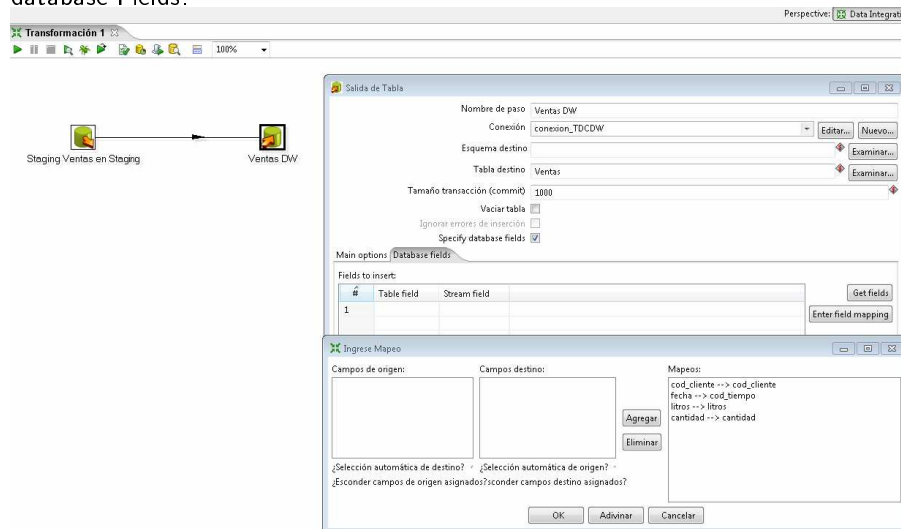
- f) Guardamos la transformación. Ir a Fichero → Guardar como. Le asignamos el nombre *tr_ventas_a_StagingVentas*. Clic en Guardar.
3. Cargar las Ventas al Data Warehouse: Carga de la tabla *Ventas* del Data Warehouse. Una vez creada la transformación para cargar los datos a la tabla intermedia *StagingVentas*, crearemos otra transformación para migrar los datos de la misma a la tabla *Ventas* en la base de datos *TDC DW*.
- En el menú Fichero → Nuevo → Transformación.
 - En la solapa de *Objetos Principales*, seleccionar **Entrada** → **Entrada tabla**. Luego, seleccionar en la sección *Salida* → **Salida tabla**. Unir la entrada con la salida manteniendo presionada la tecla “shift”.
 - Doble clic sobre el icono de entrada para acceder a su configuración. Le asignamos al paso el nombre *Staging Ventas en Staging*. En conexión seleccionamos *conexion_TDCStaging*. En el campo SQL ingresamos una sentencia mediante la cual totalizaremos la cantidad de litros y de productos vendidos por cliente y por fecha:

```
SELECT
  cod_cliente,
  fecha,
  SUM(litrosEnvase * cantidad) AS litros,
  SUM(cantidad) AS cantidad
FROM stagingVentas
GROUP BY 1,2
```



Clic en OK para finalizar la configuración de la entrada.

- d) Configuraremos ahora la salida, que será la tabla de hechos *Ventas* de nuestro Data Warehouse. Le asignamos al paso el nombre *Ventas DW*. En *Conexión*, seleccionamos *conexion_TDCDW*. En nombre de la tabla ingresamos *Ventas*. Marcamos el campo *Specify database Fields*.



- e) En la solapa *Database Fields*, clic en el botón *Enter field Mapping*. Especificamos los campos de entrada y los de salida. Clic en OK. Clic en OK para finalizar la configuración

de la salida.

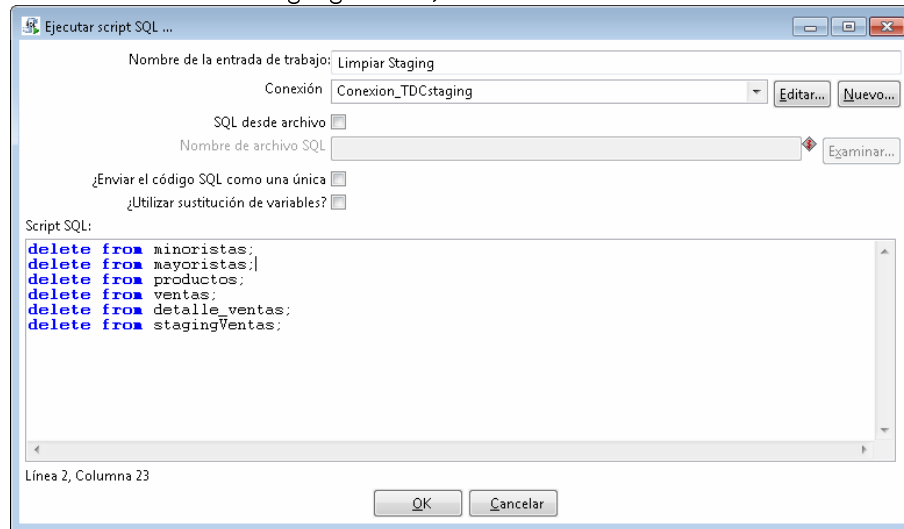
f) Guardamos la transformación. Asignamos el nombre `tr_ventas_a_DW`. Clic en Guardar.

4. Introducir las transformaciones en el proceso completo.

Debemos insertar nuestras transformaciones en el trabajo que representa nuestro proceso ETL completo.

- a) Abrimos el archivo del trabajo. Fichero → Abrir → `C:\SGII501\procesoETL.kjb`
- b) Modificamos el script *Limpiar Staging*, agregando una sentencia para limpiar los datos de la tabla intermedia *stagingVentas*:

```
delete from stagingVentas;
```



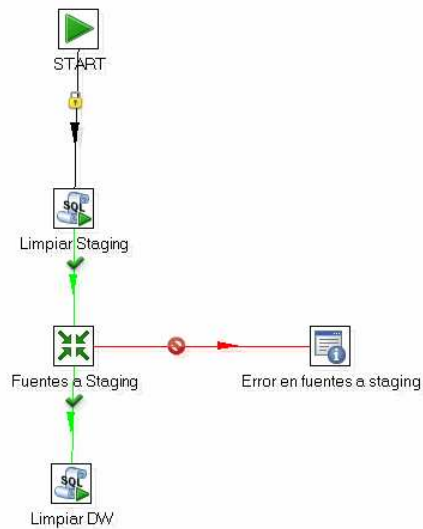
Clic en OK para guardar el cambio.

- c) Insertamos un script SQL que limpie los datos del Data Warehouse. En el panel de Objetos principales → Scripting → SQL. Doble clic para ingresar a la configuración. En nombre del paso ingresamos *Limpiar DW*. En conexión seleccionamos *conexion_TDCDW* y en SQL las siguientes sentencias:

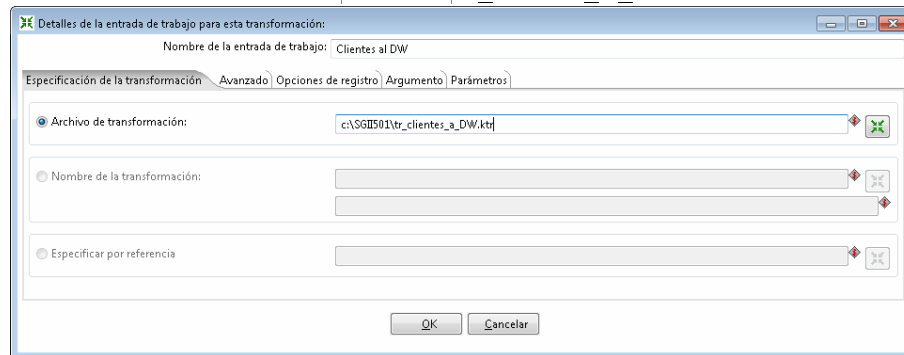
```
delete from ventas;
delete from clientes;
```

Clic en OK.

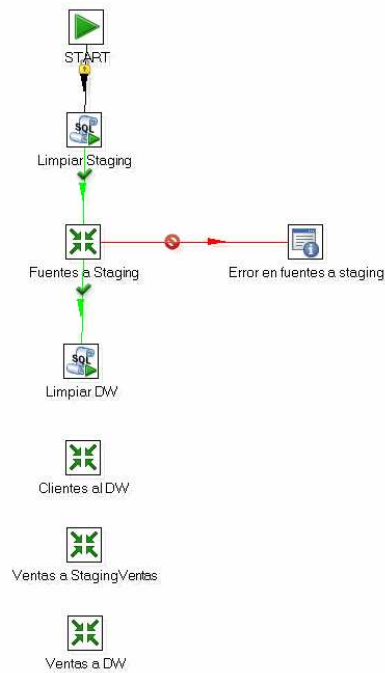
- d) Conectamos por condición de éxito la salida de la transformación *Fuentes a Staging* con el script *Limpiar DW*, para indicar que una vez que estén migradas las fuentes de datos a la Staging Area, se realice la limpieza de los datos en el Data Warehouse.



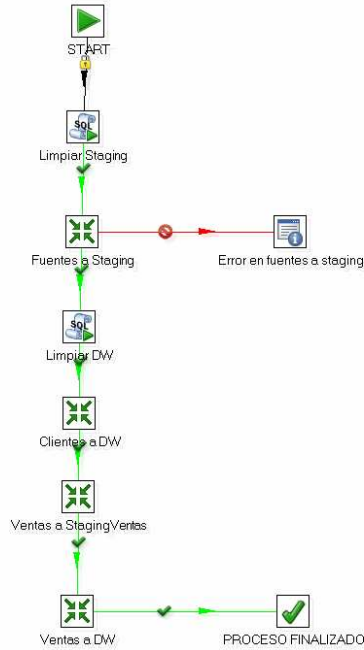
- e) Agregamos al proceso la transformación que carga los clientes al Data Warehouse. En el panel de Objetos principales → General → Transformación. Doble clic en la misma para configurarla. En nombre de entrada asignamos *Clientes al DW*. En archivo de transformación indicamos la ruta *c:\SGII501\tr_clientes_a_DW.ktr*.



- f) Repetimos el paso anterior para agregar las transformaciones *tr_ventas_a_stagingVentas.ktr* (la cual llamaremos *Ventas a StagingVentas*) y *tr_ventas_a_dw.ktr* (la cual llamaremos *Ventas a DW*).



- g) Manteniendo presionada la tecla “shift”, conectamos por condiciones de éxito las transformaciones *Limpiar DW*, *Clientes al DW*, *Ventas a Staging* y *Ventas a DW*.
- h) Agregamos un elemento que nos indique la finalización exitosa del proceso desde el menú de **Objetos principales** → **General** → **Éxito**. Le asignamos el nombre *Proceso Finalizado*.
- i) Conectamos mediante la tecla “shift” la última transformación (*Ventas a DW*) con el elemento de éxito.



De esta forma, ya hemos finalizado el diseño del proceso de carga de datos a nuestro Data Warehouse. Presionamos el botón **Play** de la barra de herramientas para correr el mismo. Aparecerá un cuadro de diálogo en el que presionamos el botón **Ejecutar**.

- j) Mediante la solapa **Logging** del panel inferior podremos consultar toda la información sobre la ejecución del proceso, como errores, mensajes de advertencia y registros cargados en cada tabla.

Capítulo 5

OLAP: Definición de una estructura multidimensional con IBM Cognos

5.1. Introducción

Nuestro objetivo es la definición de una estructura multidimensional de tal forma que pueda atender los requerimientos planteados en forma ágil y dinámica:

- Cantidad de litros consumidos y de productos adquiridos por cliente en el tiempo.
- Compra promedio en litros por cliente en el tiempo.

El esquema starnet de nuestro modelo dimensional estará formado de la siguiente manera¹:

- Medidas
 - Cantidad de Productos Vendidos
 - Cantidad de Litros vendidos
 - Cantidad de Litros Promedio (Medida Calculada)
- Dimensiones y Granularidad:
 - Tiempo: año, mes, día
 - Cliente: nombre

5.2. Prerrequisitos

- IBM Cognos BI Suite
- Base de Datos del Data Warehouse de TDC cargada a través de los procesos ETL

¹Para una descripción del esquema starnet, ver la sección 2 en la página 6

5.3. Crear un origen de datos ODBC

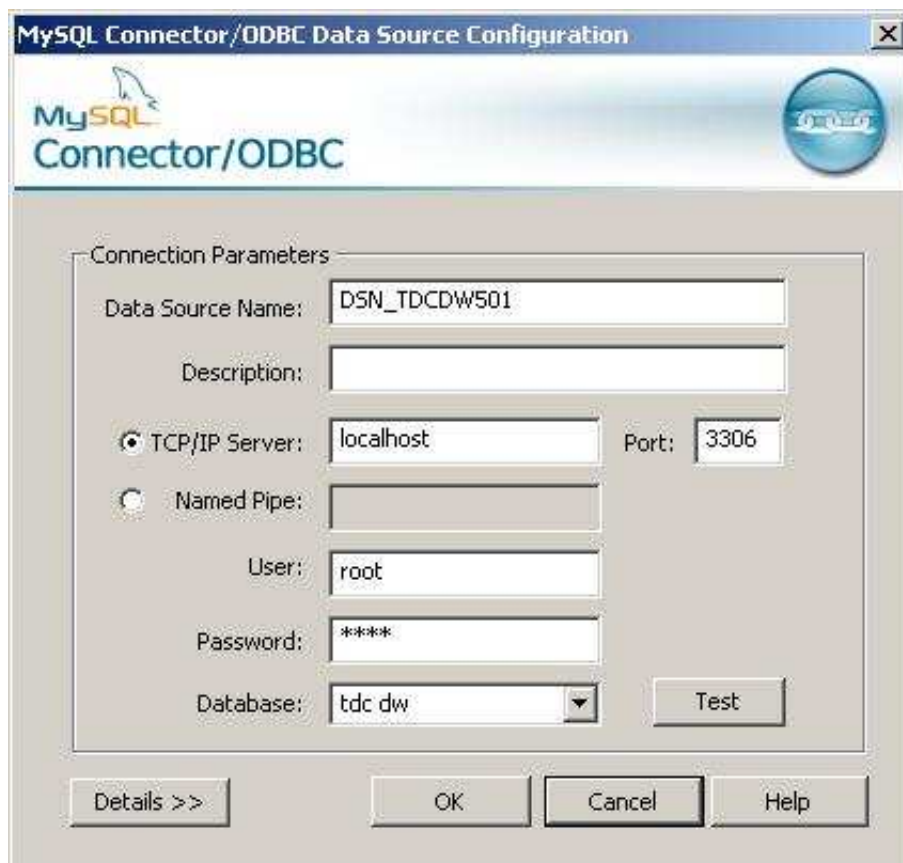
El origen de datos ODBC provee una interfaz para que Cognos pueda conectarse a una base de datos de un tipo no soportado en forma nativa, en este caso MySQL.

1. Dirigirse a Inicio → Ejecutar... . Escribir *odbcad32* para entrar al administrador de orígenes de datos ODBC.



Clic en la solapa DSN de sistema. Clic en Agregar...

2. Seleccionar *MySQL ODBC 5.1 Driver* de la lista. Clic en Finalizar. Aparece la siguiente ventana:



Completamos con los siguientes datos: Data Source Name: *DSN_TDCDW501*. En TCP/IP Server: *localhost*. En user: *root*. En password: *root*. En Database seleccionamos la base de datos del Data Warehouse.

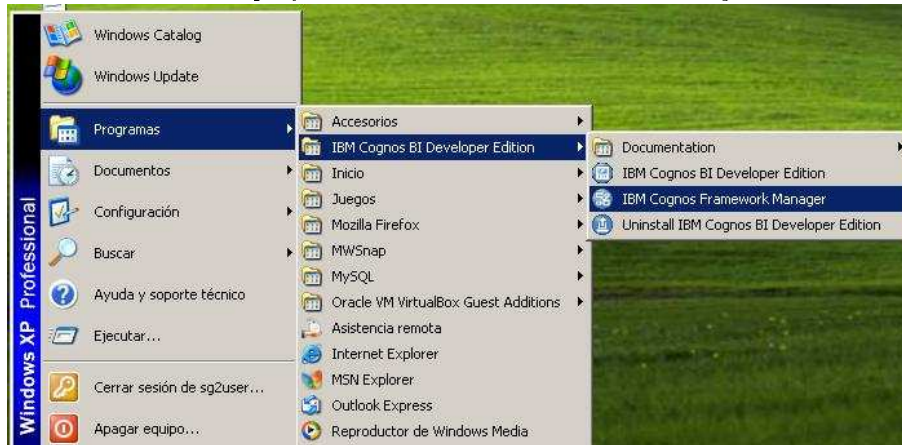
3. Clic en Test para probar la conexión. El nombre de origen de datos (Data Source Name o DSN) ya ha sido definido.

5.4. Crear un nuevo proyecto

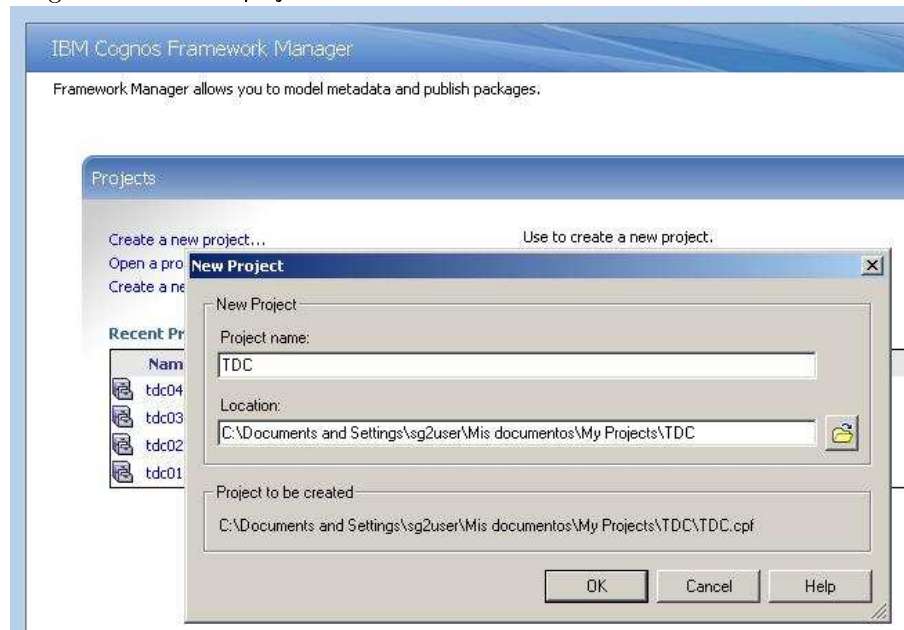
En IBM Cognos, un *Proyecto* es un elemento de datos que contiene, entre otros:

- El namespace, o contenedor de objetos de trabajo (elementos de consulta, dimensiones, medidas)
- La conexión al origen de datos (que en nuestro caso es la Base de Datos del Data Warehouse)
- Los Paquetes, que son objetos que gestionan los accesos y recursos para publicar las estructuras multidimensionales en el servidor de Cognos.

1. Crearemos un nuevo proyecto en el módulo *Framework Manager*



- a) Elegir Create a new project...

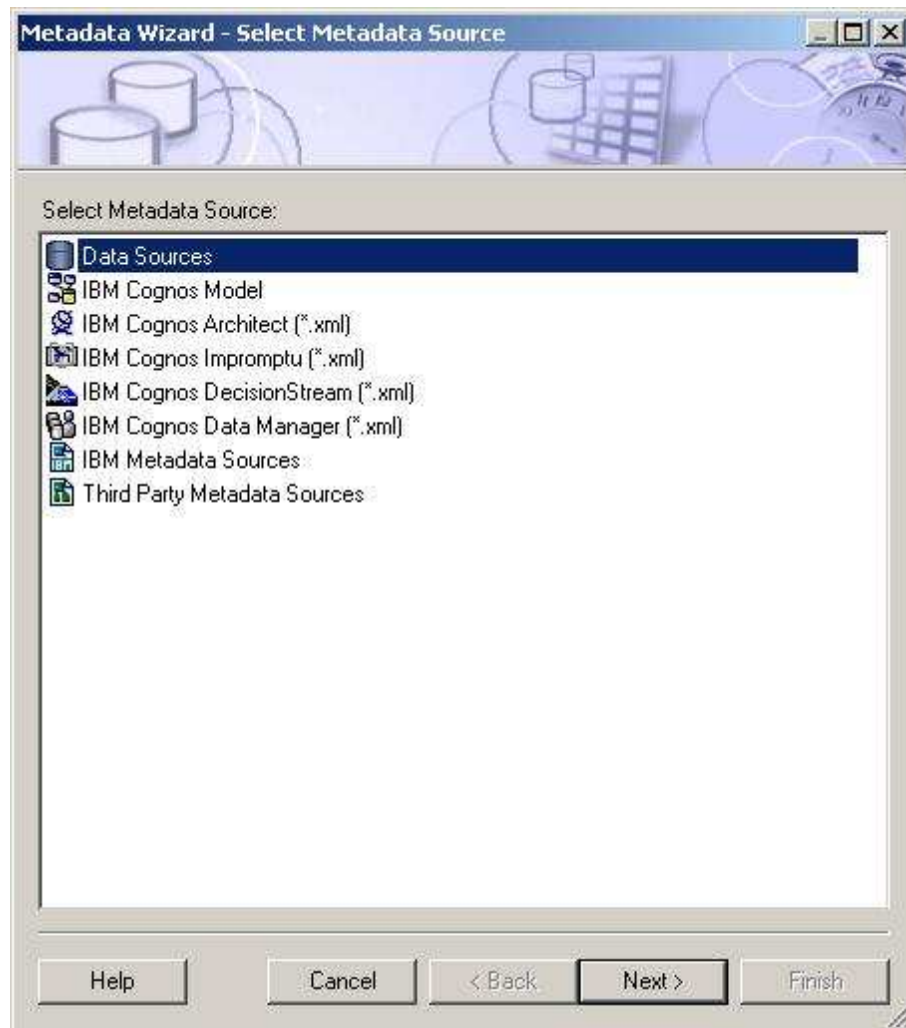


Ingresar el nombre *TDC*. Clic en OK.

- b) Aparecerá una ventana de inicio de sesión. Ingresar las credenciales proporcionadas en la instalación de Cognos

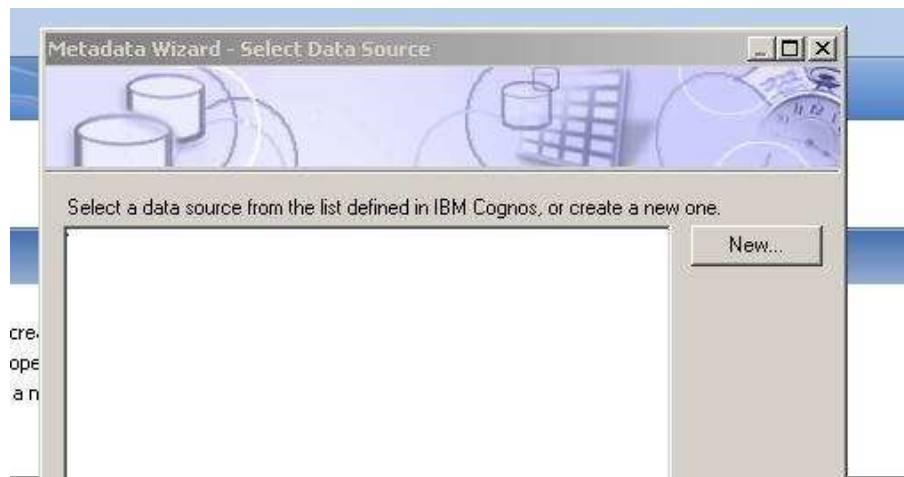


- c) Seleccionar el idioma (por defecto toma el del Sistema Operativo). Clic en OK.
- d) Seleccionar *Data Sources* como fuente de metadatos.



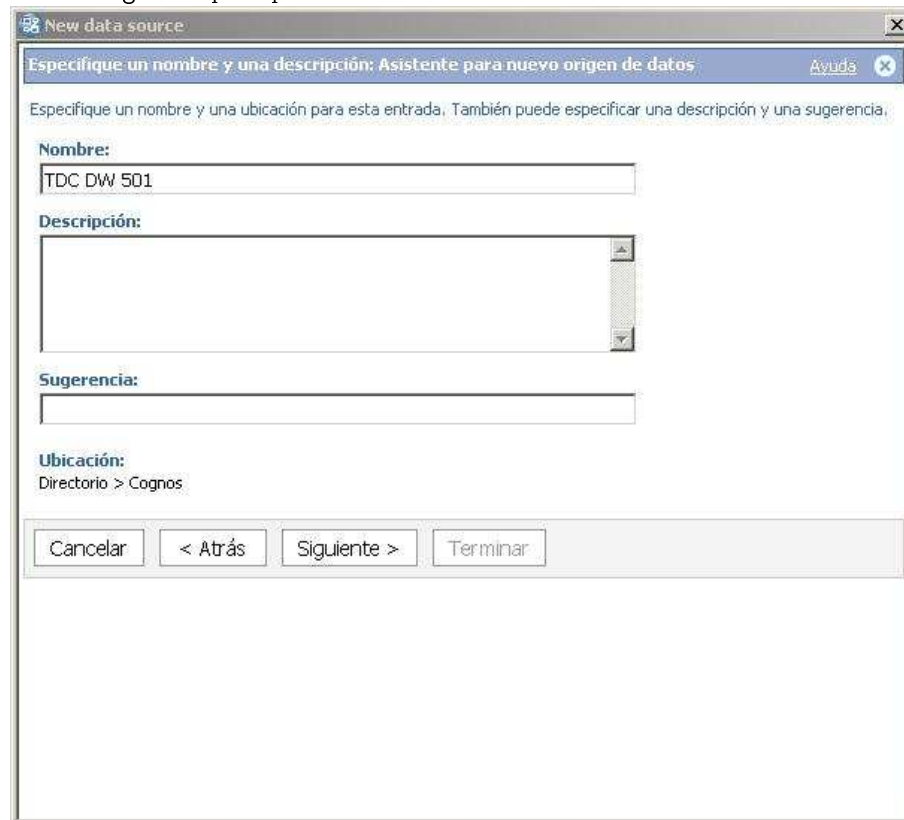
Clic en Next >.

- e) Crear una nueva Fuente de Datos, basado en el DSN definido en la sección 5.3 en la página 43



Clic en New.... Se abrirá el asistente para nuevo origen de datos de Cognos.

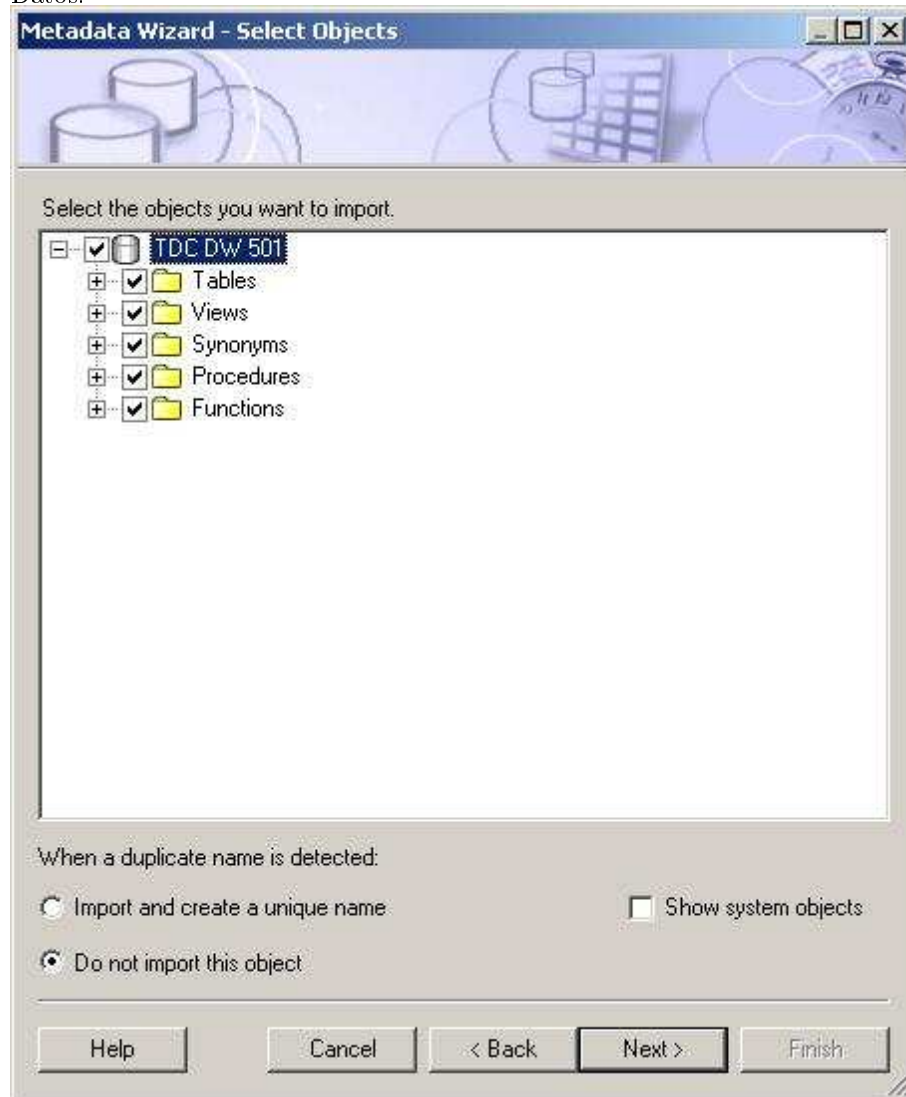
f) Clic en Siguiente para pasar la ventana de introducción.



Especificar el nombre *TDC DW 501* para la entrada. Clic en Siguiente >

g) En Tipo especificar ODBC. Clic en Siguiente >.

- h) En Origen de datos de ODBC ingresar el nombre que definimos en la sección 5.3 en la página 43. En la misma ventana, en Inicio de Sesión seleccionar Sin autenticación. En la parte inferior, clic en Probar la conexión... para verificar su correcto funcionamiento. Clic en Probar.
- i) Si la conexión es correcta (mensaje de Estado: Correcto), clic en Cerrar. Clic en Cerrar.
- j) Clic en Terminar. Clic en Cerrar. El origen de datos fue creado correctamente.
- k) Clic en Next >. El Asistente solicitará indicar los objetos a importar desde la Base de Datos.

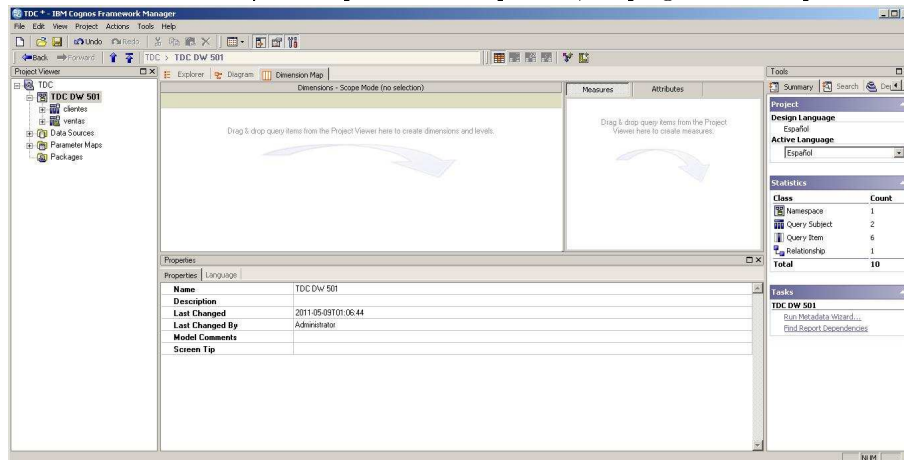


Seleccionar todos. Clic en Next >.

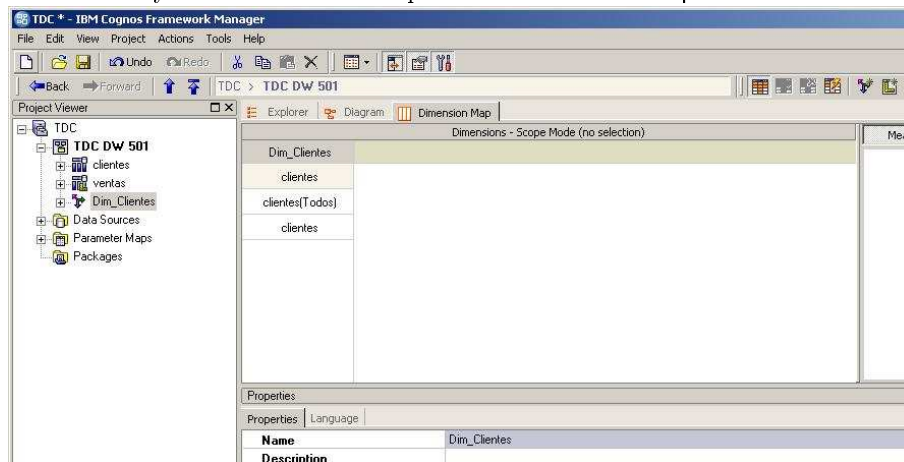
- l) Dejar las opciones de relaciones por defecto. Clic en Import.

- m) Clic en Finish. Los objetos de la base de datos del Data Warehouse han sido importados a Cognos Framework Manager.
2. Definir Dimensiones. Las dimensiones en Cognos son llamadas “Dimensiones Regulares”, en contraposición de las medidas, que en Cognos son llamadas “Dimensiones de Medida”. Tener en cuenta esto para evitar confusiones. Definiremos la dimensión Cliente.

a) Clic en Dimension Map. En el panel de la izquierda, desplegar el namespace *TDC DW 501*.



b) Notar que dentro del namespace, *clientes* y *ventas* presentan distintos íconos. Esto es porque basado en la estructura y las relaciones de la base de datos del Data Warehouse, Cognos identifica potencialmente los orígenes de las medidas y las dimensiones. Seleccionar *clientes* y arrastrarlo hasta el panel del Dimension Map.

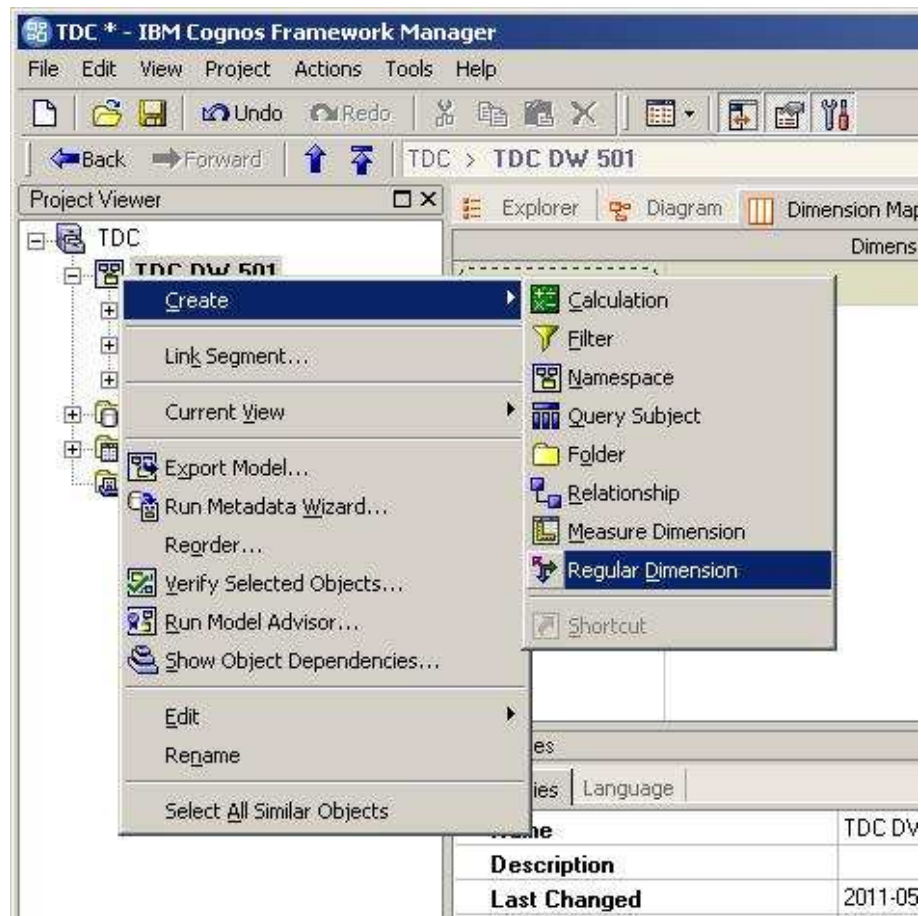


Aparece el nuevo elemento de tipo Dimensión en el panel lateral. Clic sobre el elemento y presionando “F2” lo renombraremos a *Dim_Clientes*.

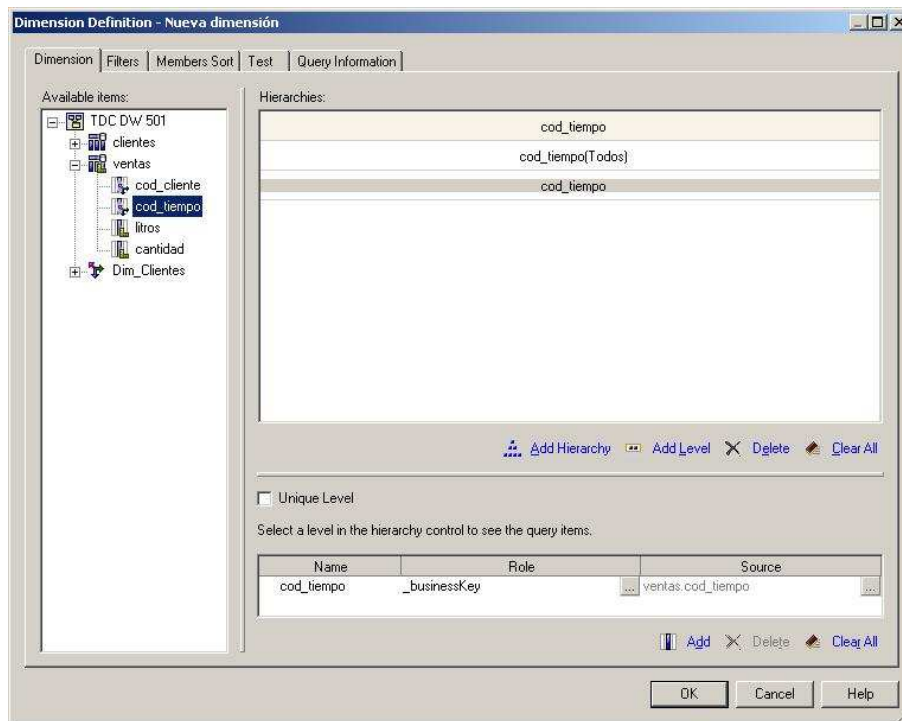
- [illegible]

Marcar Metadata. Clic en Detect. Clic en OK. Clic en OK.

- a) Clic derecho sobre el namespace. Clic en Create → Regular Dimension.

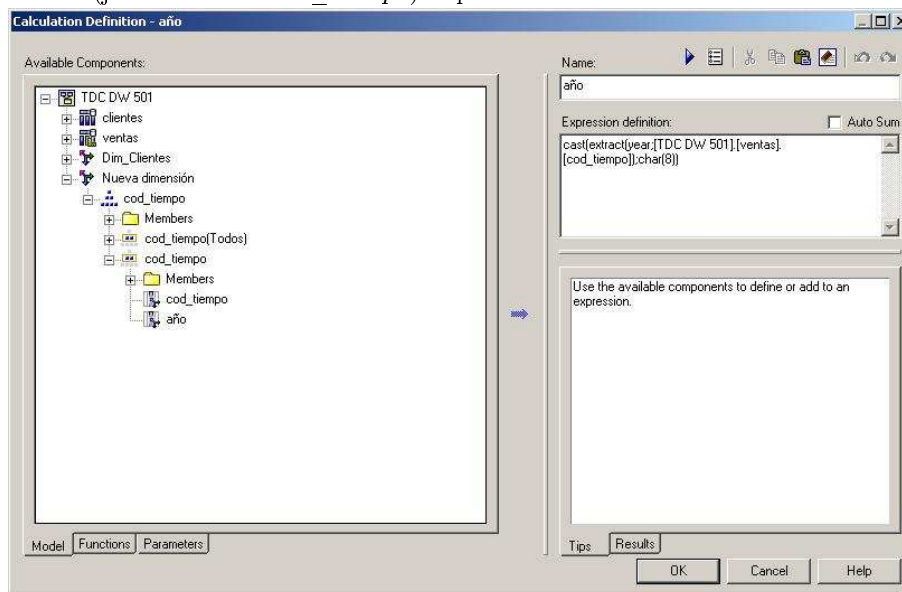


b) Aparecerá la ventana de definición de la Dimensión.



Sobre el panel de la izquierda, desplegar *ventas*. Arrastrar el ítem *cod_tiempo* hacia el panel de jerarquías, a la derecha.

- c) En el panel de niveles, en la parte inferior, clic en los puntos suspensivos de la columna Source (junto a *ventas.cod_tiempo*). Aparecerá la definición de cálculo.



En Name ingresar *año*. En Expression definition vamos a extraer el año del campo mediante el siguiente código

```
cast(extract(year;[TDC DW 501].[ventas].[cod_tiempo]);char(8))
```

Clic en OK. Clic en los puntos suspensivos de la columna Role. Clic en `_memberCaption`, para que quede marcada junto a `_businessKey`. Clic en Close.

d) Clic sobre la jerarquía `cod_tiempo`, última en la lista de jerarquías. Presionar “F2” para renombrarla a *año*.

e) Hacer lo propio para el mes. Clic en Add Level. Renombrar el nivel a *mes*. Clic en Add en la parte inferior para añadir un nivel. En la definición de cálculo ingresar *mes* como nombre e ingresar el siguiente código

```
cast(extract(month;[TDC DW 501].[ventas].[cod_tiempo]);char(8))
```

Clic en OK. Clic en los puntos suspensivos de la columna Role. Marcar `_memberCaption` y `_businessKey`. Clic en Close.

f) Repetir el paso anterior para el día. En la parte de código ingresar

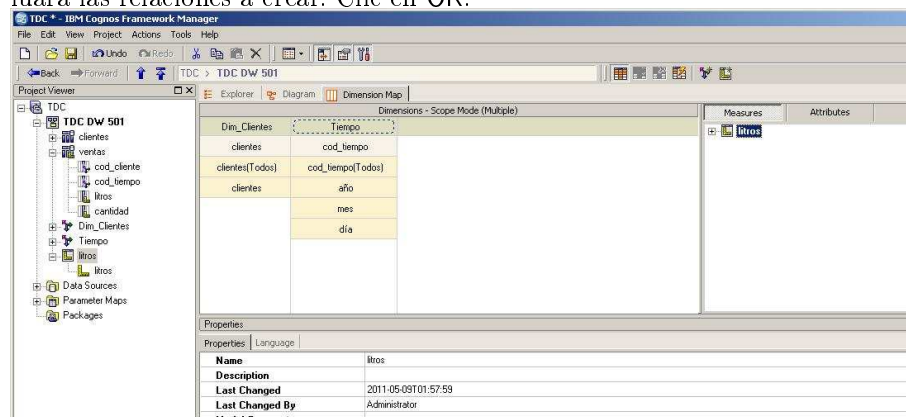
```
cast(extract(day;[TDC DW 501].[ventas].[cod_tiempo]);char(8))
```

g) Clic en OK. Clic en la solapa Members Sort para definir la granularidad. Marcar Metadata. Clic en Detect. Clic en OK. Clic en OK.

h) Renombrar la dimensión a *Tiempo*. La dimensión Tiempo ha sido creada

4. Definir Medidas.

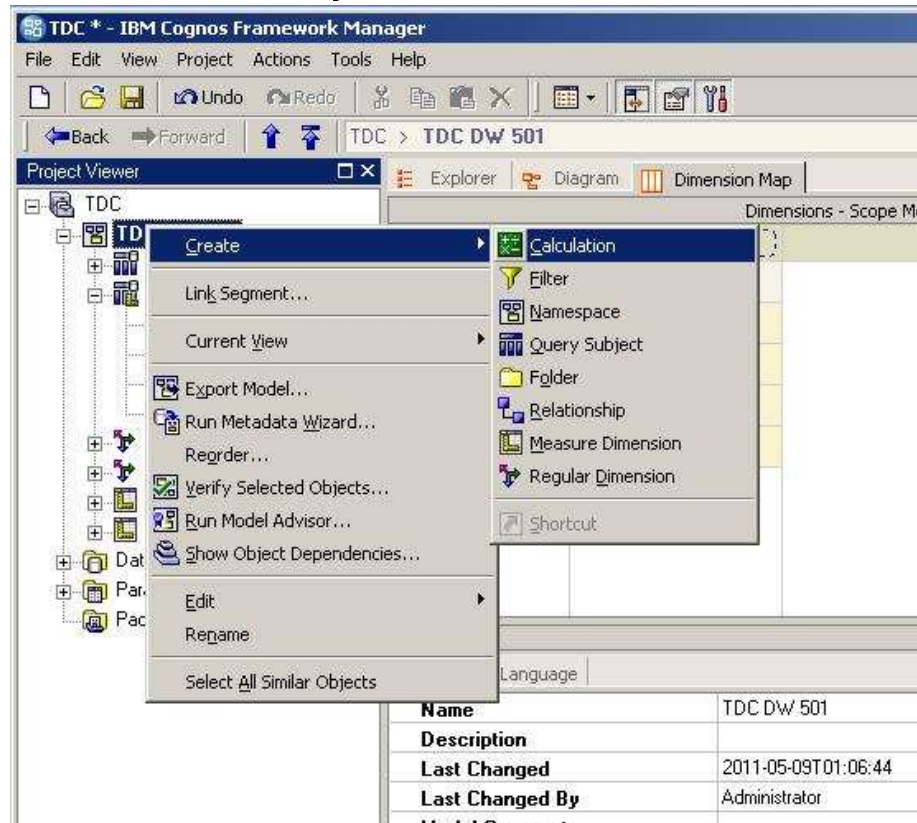
a) En el panel de la izquierda, desplegar el elemento *ventas*. Seleccionar *litros*. Arrastrar el elemento hasta el panel derecho, donde está activada la sección Measures. Cognos evaluará las relaciones a crear. Clic en OK.



b) Repetir el paso anterior con *cantidad*. Las medidas han sido creadas.

5. Definir una Medida Calculada. Una medida calculada surge en forma derivada de los objetos creados en la estructura multidimensional. Aquí incluiremos los litros promedio.

- a) Clic derecho sobre el namespace. Clic en Create → Calculation.



- b) Aparecerá la ventana de cálculo. En Name ingresar *litros promedio*. En la expresión ingresar el siguiente cálculo

`[TDC DW 501].[litros].[litros] / count([TDC DW 501].[litros].[litros])`

- c) Clic en OK. La medida calculada ha sido creada

6. La estructura multidimensional ha sido creada

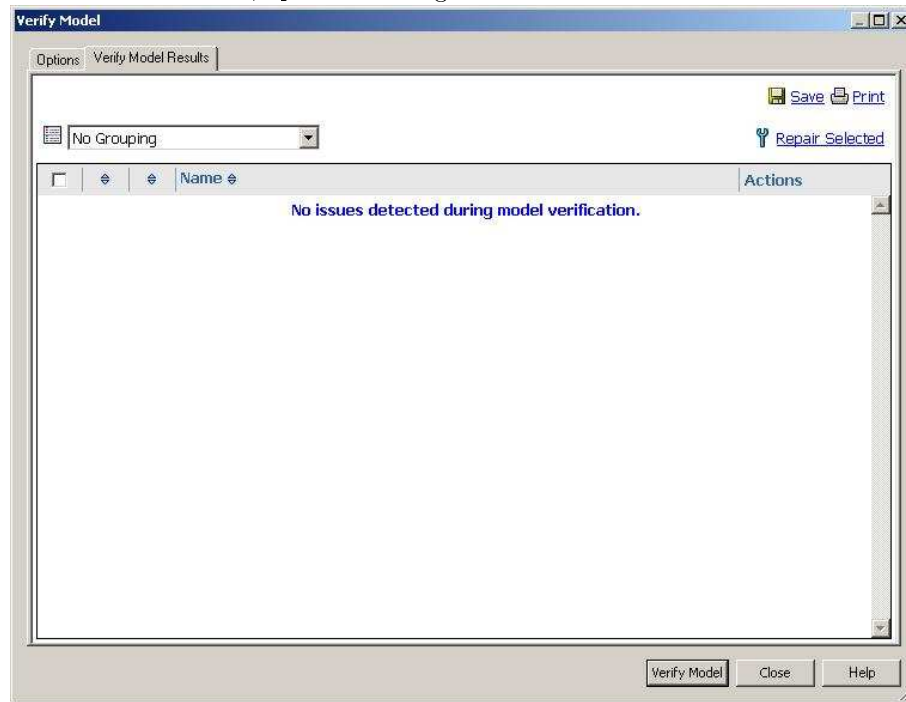
5.5. Publicar un Paquete

La publicación del paquete es necesaria para poder acceder a la estructura creada.

1. Preliminares

- a) Para poder realizar consultas sin restricciones, dirigirse en el menú principal a Project → Edit Governors.... en Cross-Product joins seleccionar Allow. Clic en OK.
- b) Verificar la estructura creada. En el menú principal, dirigirse a Project → Verify Model. Aparece una lista de lo que será verificado. Clic en Verify Model

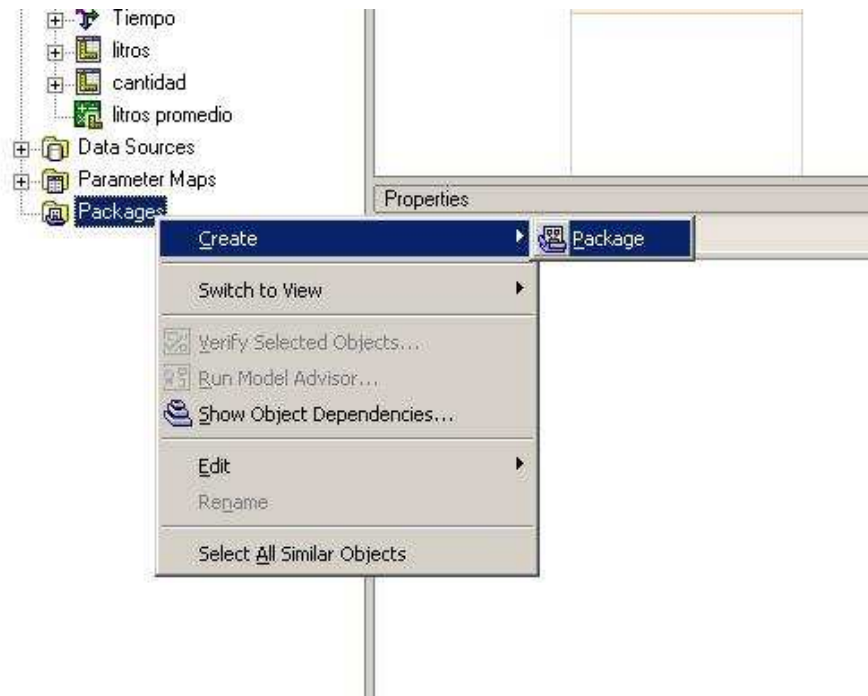
c) Si todo está correcto, aparecerá la siguiente ventana



Clic en Close.

2. Publicar el paquete

a) En el panel izquierdo, clic derecho sobre Packages. Seleccionar **Create** → **Package**.



- b) Ingresar como nombre *TDC 501*. Clic en Next >.
 - c) Dejar seleccionados todos los objetos. Clic en Next >.
 - d) No hacer modificaciones sobre las opciones de funciones. Clic en Finish.
 - e) Definido el paquete, aparece el Asistente de Publicación. Clic en Yes.
 - f) Dejar las siguientes opciones por defecto. Clic en Next >. Clic en Next >. Clic en Publish.
 - g) Clic en Finish. El paquete ha sido publicado.
3. Guardar el paquete. Cerrar Framework Manager.

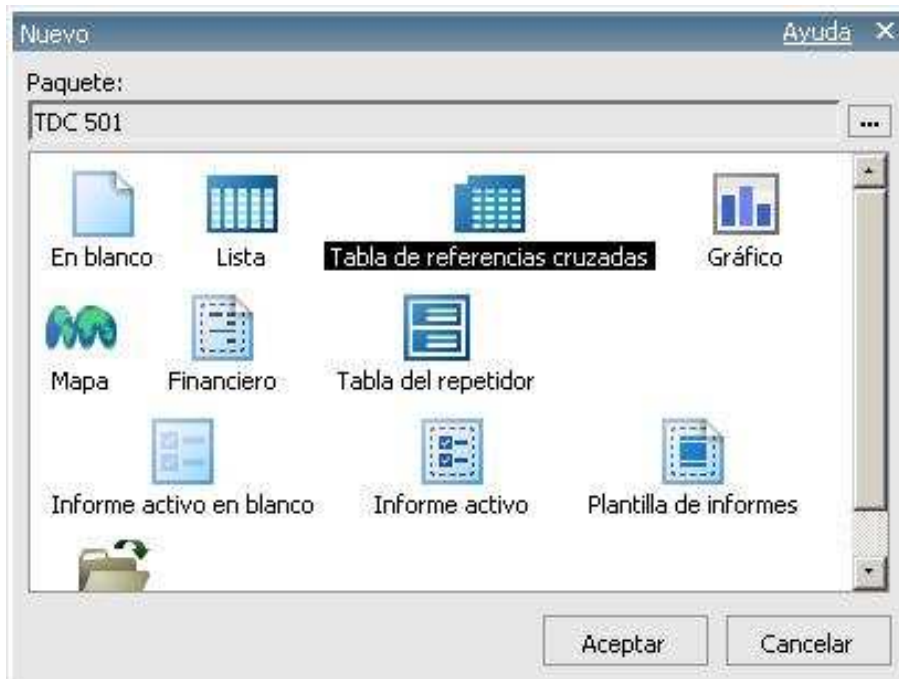
5.6. Navegar la estructura creada

1. Crear un informe

Ingresar al Cognos Manager desde un navegador de Internet ingresando la dirección http://localhost:9300/cognos_developer/manager/main.html. Sobre la parte superior derecha, clic en Iniciar → Report Studio

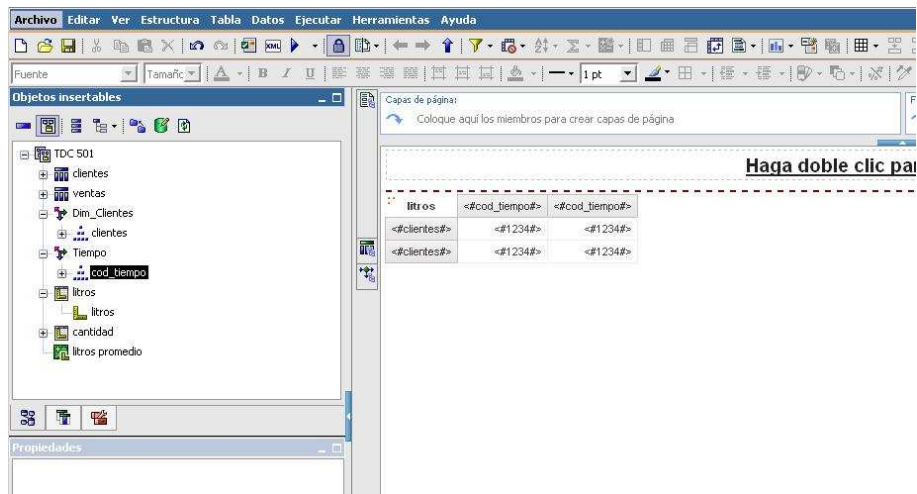


- a) Seleccionar el paquete creado, *TDC 501*. Se abrirá el módulo IBM Cognos Report Studio. Clic en Crear nuevo.
- b) Seleccionar Tabla de referencias cruzadas



Clic en Aceptar

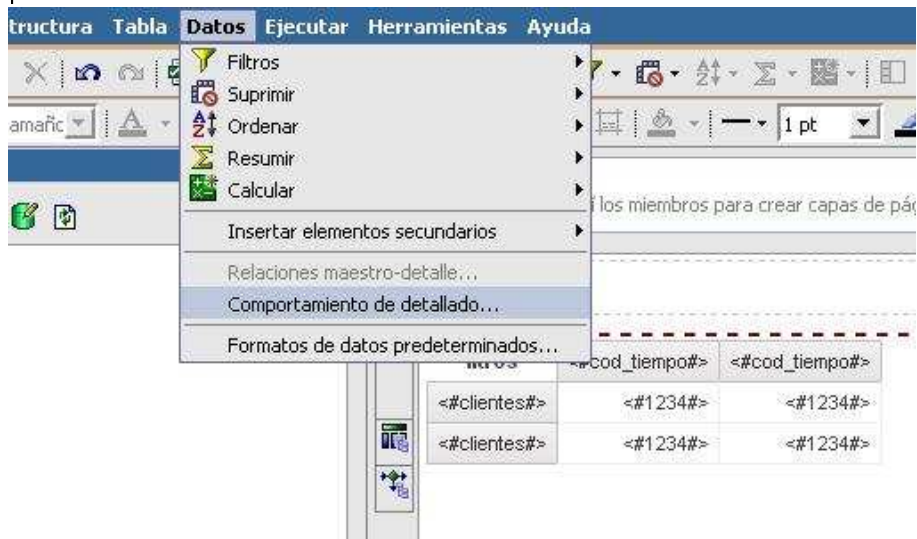
- c) En el panel de la izquierda, desplegar el elemento *litros*. Arrastrar la medida *litros* al panel central, en el área de Datos.
- d) Desplegar la dimensión *Dim_Clientes*. Seleccionar el elemento *clientes* y arrastrarlo sobre el área de Filas. Seleccionar Todos los miembros. Clic en Aceptar.
- e) Desplegar la dimensión *Tiempo*. Seleccionar el elemento *cod_tiempo* y arrastrarlo sobre el área de Columnas. Seleccionar Todos los miembros. Clic en Aceptar.



El informe ha sido definido

2. Navegar el informe

- a) Para permitir la ejecución de las acciones de drill-down y roll-up (llamadas en Cognos *profundizar* y *replegar* respectivamente), en el menú principal dirigirse a Datos → Comportamiento de detallado....



Marcar Permitir profundizar y replegar. Clic en Aceptar.

- b) En el menú principal, dirigirse a Ejecutar → Ejecutar informe (HTML). El informe se mostrará en el navegador.

litros	cod. tiempo(Todos)	2009	1	1	10	11	12	13	14
clientes(Todos)	4.863,118,5100649	4.863,118,5100649	851,432,1599526	26,218,0199803	32,119,3099088	26,909,340118	21,087,4600038	24,529,0900888	29,614
Spence, Michael	3.669,3100041	3.669,3100041	727,8400039	31,5	41,1699982		6,5		
Damstra, Robert	4.482,7100017	4.482,7100017	578,2400018	17,8500004	25,5				
Brunner, Kim	2.929,7599956	2.929,7599956	278,0200005	18	24,5	15,8800003			
Murray, Jonathan	2.002,0500039	2.002,0500039	288	36,5				45	
Rutledge, Bryan	1.374,369997	1.374,369997	177,1800003	72,6800003					
Wyllie, Jacqueline	2.814,1299924	2.814,1299924	439,1699982	16,7000008				29,5	
Conley, Lin	1.980,4199964	1.980,4199964	374,6399994	5,5	38,5		40,6399994		
Wood, Lois	3.166,4500127	3.166,4500127	356,7900009	19,5	23,2999992				
Armstrong, Kevin	3.293,420019	3.293,420019	647,4400024	30,5	37,5	28,9699993	64,6500015	26	
Gooley, Cody	1.978,8400108	1.978,8400108	145,0300007	30					
Nickell, Paula	3.179,0300133	3.179,0300133	692,0300006	35				60	
Long, Eric	1.004,4099904	1.004,4099904	103,1399996	2,8199999					
Young, Ed	5.671,0799745	5.671,0799745	505,5000006	8,3000002		36			
Arnold, Donna	2.893,8600093	2.893,8600093	506,449995	30,8199997		21		32	
Bechard, Howard	1.854,4800042	1.854,4800042	214,7999983	10,3000002		10,5			
Sims, Rose	3.965,7000054	3.965,7000054	431,6200019	71	8,5	28,8400002	31		
De Carlo, Lauretta	3.301,5800183	3.301,5800183	305,2200004	47	26,5			28	45

Apéndice A

Instalación

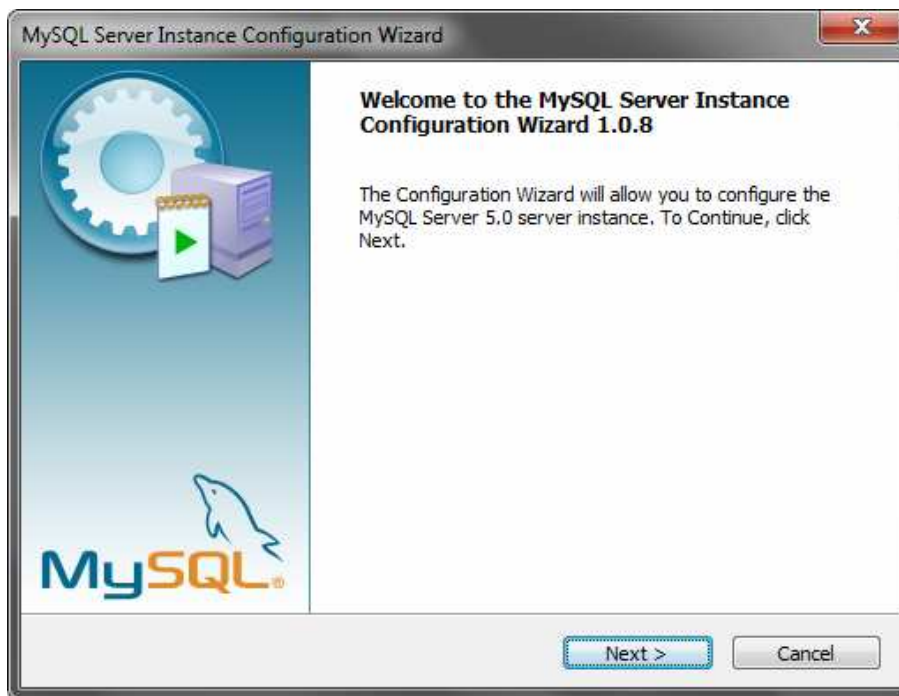
A.1. Instalación y configuración de componentes MySQL

A continuación, presentamos una guía para la instalación y configuración de los componentes MySQL.

A.1.1. Instalación del servidor MySQL

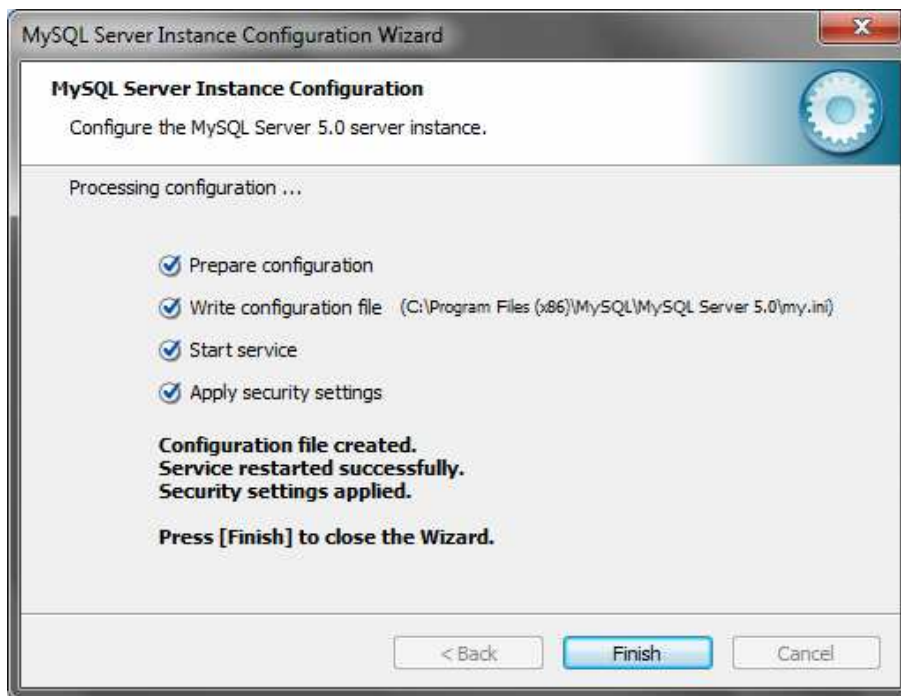
Para instalar el servidor MySQL Community Edition versión 5 en nuestro equipo, procedemos a descargar el instalador desde <http://dev.mysql.com/downloads/mysql/>

Descargamos el paquete *essential* (componentes básicos) que corresponda a nuestro sistema operativo. Una vez ejecutado el instalador, se nos preguntará si deseamos configurar la instancia del servidor, a lo que respondemos afirmativamente. Se iniciará *MySQL Server Instance Configuration Wizard*:



Se inicia el asistente de configuración. Presionamos next, dejando los valores que nos propone por defecto. En el cuarto paso, se nos solicita un nombre de usuario y contraseña. Establecemos Usuario: *root*, Contraseña: *root*.

Una vez finalizados los pasos, presionamos el botón Execute.



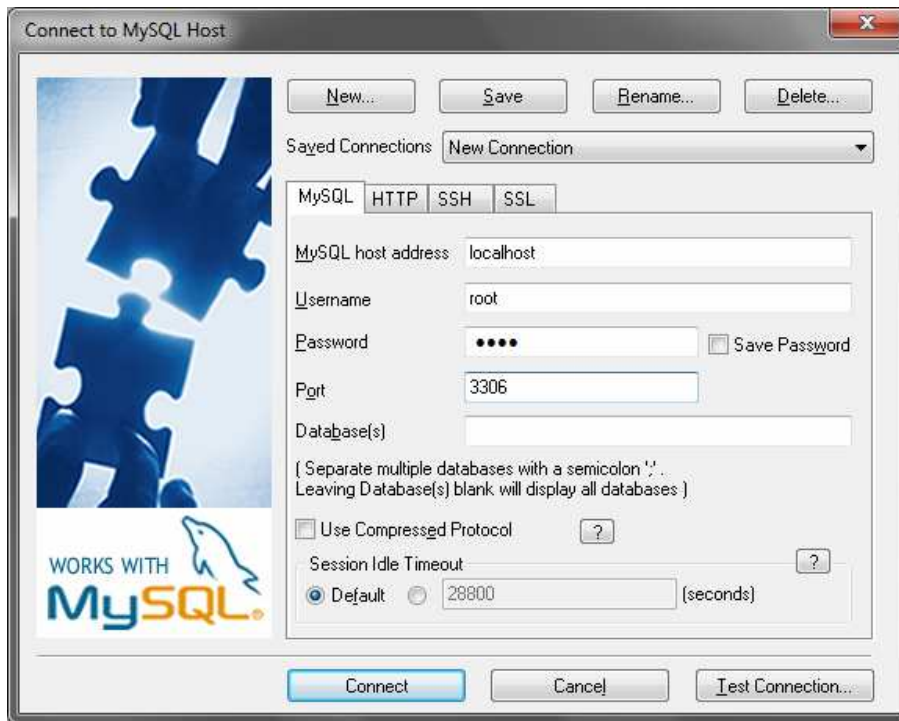
Clic en Finish. Nuestro servidor ya está configurado y ejecutándose.¹

A.1.2. Instalación de una herramienta de administración

Instalamos una herramienta de administración grafica de nuestro servidor. Una alternativa es *SQLYog Community Edition*. Podemos descargar su versión 8.4 desde <http://code.google.com/p/sqlyog/downloads/list>.

Finalizada la instalación, al ingresar a la herramienta SQLYog, nos aparecerá la ventana **Connect to MySQL Host**. Presionamos el botón **New** para establecer una nueva conexión

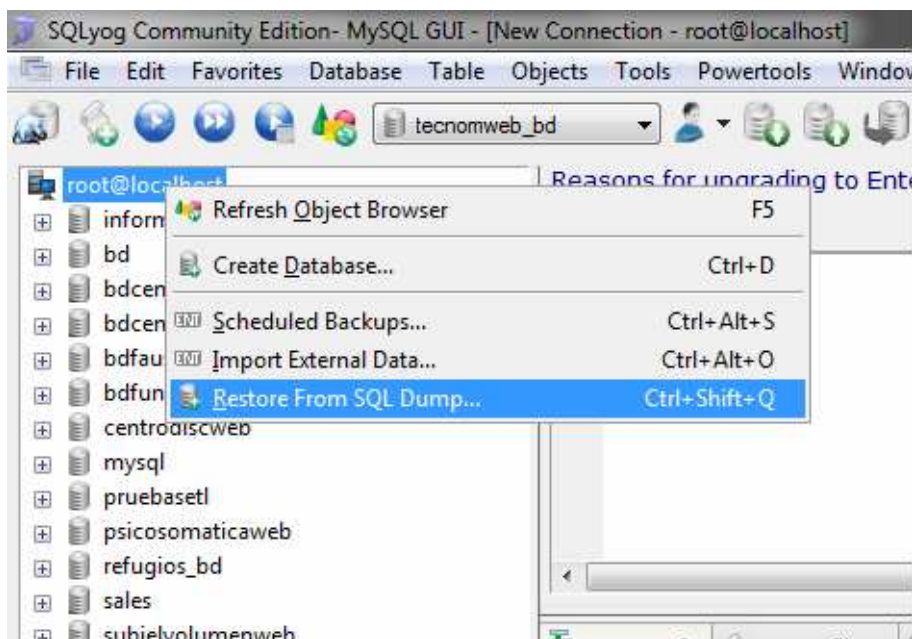
¹Nota: Si al aplicar la configuración obtuviéramos un error en la tarea **Apply security settings**, probablemente hubo un conflicto con nuestro firewall. Procedemos a deshabilitarlo temporalmente para poder finalizar la instalación.



Completamos los campos de conexión: Host *localhost*, Username *root*, Password *root* y Port *3306*. Presionamos el botón **Connect**.

A.1.3. Restauración de una base de datos

Para utilizar las fuentes de datos de Ventas que se nos ha proporcionado para el caso de estudio de la empresa TDC, debemos utilizar el archivo *sales.bak*. Una vez conectados al servidor, clic con el botón derecho sobre el icono del mismo (en panel izquierdo) y seleccionamos **Restore From SQL Dump...**



Especificamos la ubicación del archivo *sales.bak* y presionamos **Execute**. Una vez finalizada la restauración, presionamos **F5** para actualizar la lista de base de datos en nuestro servidor.

A.2. Instalación y configuración de Pentaho Data Integration

Para realizar el proceso de extracción, transformación y carga (ETL) de los datos al DW, utilizaremos la versión *community* de Pentaho Data Integration (conocida como *Kettle*).



Pasos para su instalación:

A.2.1. JRE (Java Runtime Enviroment)

Si no lo tenemos instalado en nuestro equipo, lo podemos descargar desde <http://www.java.com/en/download/manual.jsp>

A.2.2. Pentaho Data Integration

Descargamos el proyecto PDI en su versión 4.0.1 desde el sitio oficial <http://sourceforge.net/projects/pentaho/files/Data%20Integration/4.0.1-stable/pdi-ce-4.0.1-stable.zip/download>.

La descarga corresponde a un archivo comprimido, el cual contiene la carpeta *data-integration*. Descomprimos la misma. No es necesario correr un instalador. Ejecutamos el archivo `spoon.bat`²

²Nota: Para poder utilizar conexiones JDBC con bases de datos, es necesario que los controladores estén instalados en la carpeta `.../libext/JDBC`. Como las bases de datos del tutorial son de tipo MySQL, descargamos el controlador desde <http://dev.mysql.com/get/Downloads/Connector-J/mysql-connector-java-5.1.15.zip> /from/ <http://mysql.cce.usp.br/>. Copiamos el mismo en `.../libext/JDBC`.