SMD. Práctica 3. Herramientas ETL
PDI (*Pentaho Data Integration*)

José Samos Jiménez

Departamento de Lenguajes y Sistemas Informáticos
Universidad de Granada

2020 jsamos (LSI-UGR)

Curso 2019-2020

Índice

1. Introducción **3**

2. Operaciones sobre *PostgreSQL* **3**
2.1. Crear BD y esquema . . . 3
2.2. Crear una tabla y definir su estructura . . . 4
2.2.1. Desde *PgAdmin* . . . 4
2.2.2. Desde *LibreOffice* . . . 4
2.2.3. Desde *Spoon* . . . 6
2.3. Generar sentencias SQL . . . 7
2.3.1. Desde *PgAdmin* . . . 7
2.3.2. Desde *Spoon* . . . 9

3. Operaciones sobre *Spoon* **9**
3.1. Crear transformación o trabajo . . . 9
3.2. Añadir un paso . . . 9
3.3. Conectar dos pasos . . . 9
3.4. Ejecutar y analizar una transformación o trabajo . . . 10
3.5. Detalles de transformaciones específicas . . . 10
3.5.1. Leer un archivo *Excel* . . . 10
3.5.2. Seleccionar campos . . . 10
3.5.3. Generar una tabla . . . 11

4. Transformaciones a realizar **11**
4.1. Transformaciones para obtener el resultado inmediato . . . 11
4.2. Transformaciones alternativas . . . 12

Bibliografía **12**

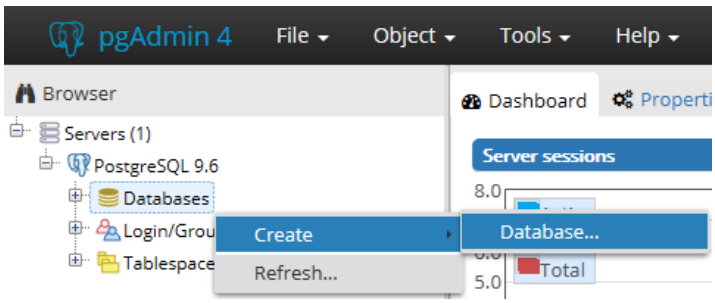

Figura 1: Crear una BD en *PgAdmin*.

Los objetivos de esta actividad son:

- Entender el componente ETL.
- Desarrollar los elementos del componente ETL para un caso sencillo (extracción, transformación y carga de datos).
- Usar una herramienta ETL profesional.
- Aprender el funcionamiento básico de la herramienta ETL *PDI* (*Pentaho Data Integration*).

1. Introducción

Esta actividad se realizará sobre *PDI* (*Pentaho Data Integration*) también conocido como *Kettle*, mediante su interfaz gráfica *Spoon*. En *PDI* hay dos elementos: transformaciones y trabajos:

- Las transformaciones se definen para trabajar con datos.
- Los trabajos se definen para organizar tareas definiendo su orden y condiciones de ejecución.

Asociadas a *PDI* se utilizan generalmente las siguientes herramientas:

- *Spoon* es la interfaz gráfica de *PDI*, permite diseñar y ejecutar transformaciones y trabajos.
- *Pan* es la herramienta que permite ejecutar transformaciones desde la línea de comandos.
- *Kitchen* es la herramienta que permite ejecutar trabajos desde la línea de comandos.

En este caso, usaremos *PDI* mediante *Spoon* y también *PostgreSQL*.
A continuación, en primer lugar, se presentan alternativas posibles sobre cómo realizar operaciones sobre *PostgreSQL* (sección 2), algunas mediante *Spoon*, y operaciones específicas sobre *Spoon* (sección 3). En la sección 4, se enuncian las operaciones que debes realizar utilizando lo que consideres más adecuado de entre lo presentado en las secciones 2 y 3, o bien con otras alternativas usando *PDI*.

2. Operaciones sobre *PostgreSQL*

2.1. Crear BD y esquema

Para crear una BD o un esquema, se puede utilizar *PgAdmin*.
Desde *PgAdmin*, para crear una BD, en el menú contextual de «Databases», pulsamos sobre [«Create», «Database»] (figura 1). Para crear un esquema en una BD, en el menú contextual asociado al nombre de la BD, pulsamos sobre [«Create», «Schema»].

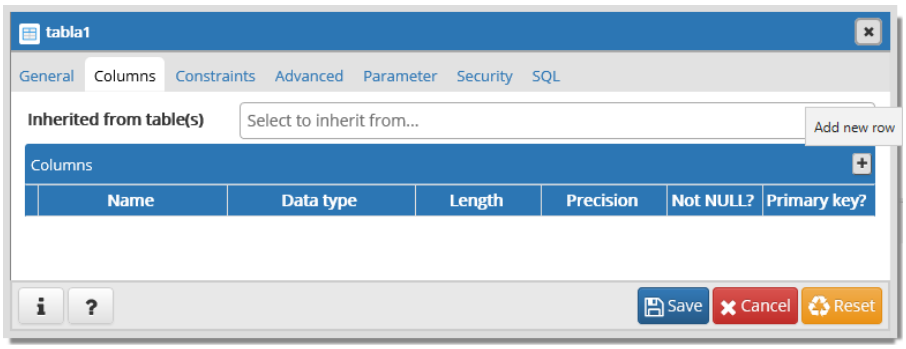

Figura 2: Crear columnas en una tabla desde *PgAdmin*.

2.2. Crear una tabla y definir su estructura

2.2.1. Desde *PgAdmin*

Desde *PgAdmin*, para crear una tabla en un esquema de una BD, en el menú contextual asociado al esquema, pulsamos sobre [«Create», «Table»]. Podemos dejar las opciones por defecto y pulsar sobre el botón «Crear»; adicionalmente, podemos definir la estructura de la tabla desde la pestaña «Columns» de la ventana de definición (figura 2).

Una vez creada una tabla, podemos definir o modificar posteriormente su estructura seleccionando la opción «Properties» en el menú contextual de la tabla. En la pestaña «Columns» podemos definir una nueva columna pulsando sobre el botón «+» que se corresponde con la operación «Add new row» (figura 2). Desde esta misma opción, desde la pestaña «Constraints», podemos definir restricciones como, por ejemplo, la clave primaria compuesta por varios campos o claves foráneas: la definición se lleva a cabo definiendo un nombre para la restricción y pulsando sobre el icono «Edit row» situado a la izquierda del nombre, para realizar la definición.

2.2.2. Desde *LibreOffice*

Para trabajar con *PostgreSQL* desde *LibreOffice*, en primer lugar, debemos definir una conexión creando una nueva BD en *LibreOffice* y definiendo que se corresponde con una BD *PostgreSQL*. Posteriormente, podemos trabajar en *LibreOffice* de manera que las transformaciones de lleven a cabo en *PostgreSQL*.

Conectar *LibreOffice* con *PostgreSQL* En *LibreOffice* pulsamos sobre «Base de datos de Base» y, en la ventana del «Asistente de bases de datos», elegimos la opción «Conectar con una base de datos existente» y seleccionamos «PostgreSQL» (figura 3)

A continuación, configuramos la conexión definiendo el host donde se encuentra *PostgreSQL* y el nombre de la BD (figura 4)

Adicionalmente, debemos definir el usuario que usaremos en la conexión y seleccionar la opción «Contraseña obligatoria». Si pulsamos sobre el botón «Probar conexión» nos pedirá la contraseña y, si todo va bien, aparecerá una ventana indicándolo (figura 5).

Por último, debemos seleccionar las opciones «Sí, registrar la base de datos» y «Abrir la base de datos para su edición» (figura 6) para trabajar en la BD *PostgreSQL* desde *LibreOffice*.

Crear una tabla y definir su estructura Una vez hemos conectado la BD actual con *PostgreSQL*, podemos ver los esquemas de la BD y, dentro de estos, las tablas definidas.

Para definir una nueva tabla, seleccionamos un esquema y pulsamos sobre la tarea «Crear tabla en modo diseño» (figura 7). Se abre una ventana donde podemos definir la lista de los nombres de campo y sus tipos, así como los índices que nos interesen.

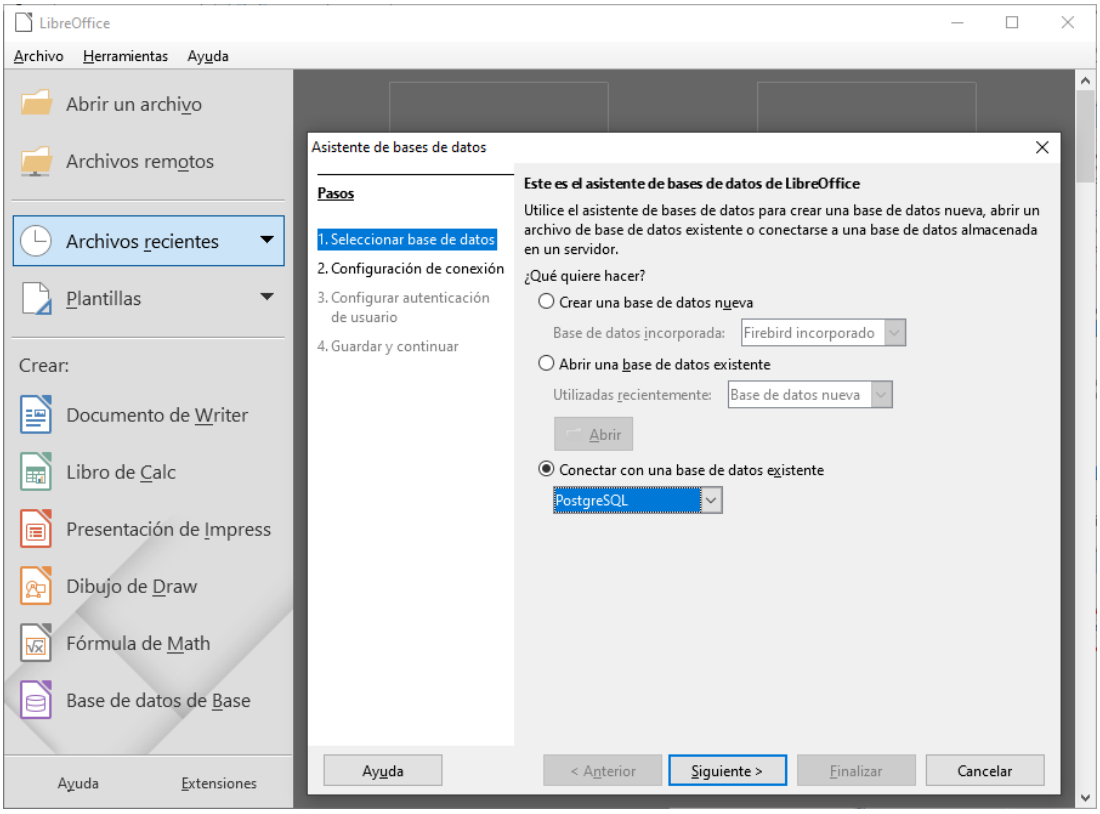

Figura 3: Conectar *LibreOffice* con *PostgreSQL*.

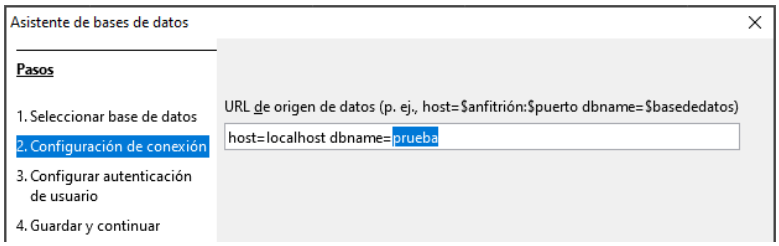

Figura 4: Configuración de la conexión.

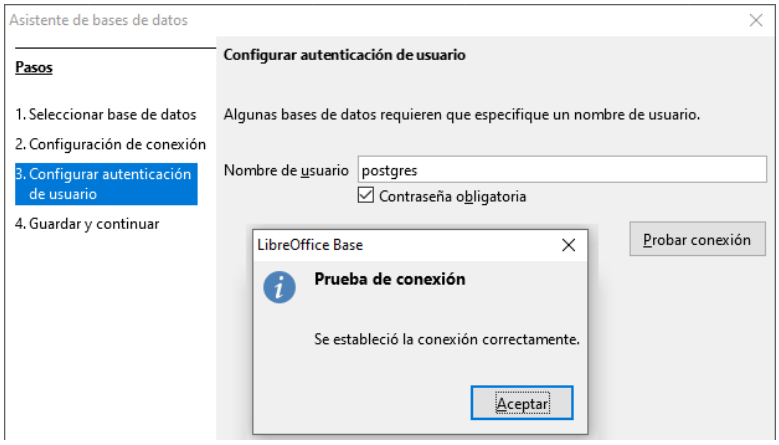

Figura 5: Probar la conexión.

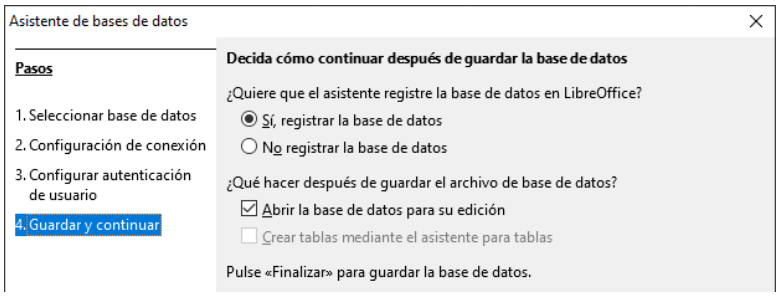

Figura 6: Registrar la BD.

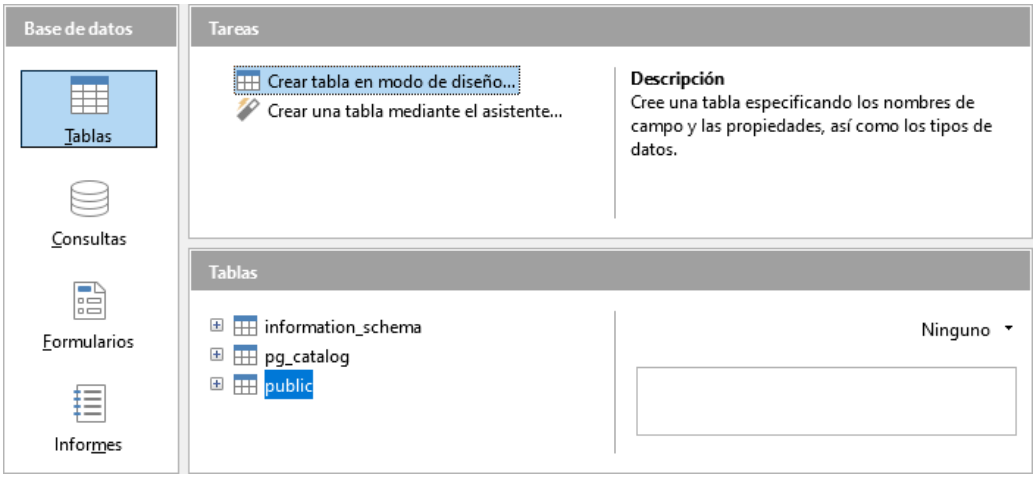

Figura 7: Crear tabla en modo diseño.

2.2.3. Desde *Spoon*

Desde *Spoon* podemos crear una tabla y definir su estructura en un solo paso. En primer lugar debemos definir una conexión con la BD; una vez establecida la conexión, se pueden definir transformaciones asociadas a la BD: en la definición de algunas transformaciones nos permite definir la estructura de la tabla de manera que pueda almacenar los datos que se produzcan.

Conectar con *PostgreSQL* Partimos de una transformación ya creada, podemos ver varios de sus elementos en la pestaña «View», entre ellos están las conexiones a BD («Database connections»). Podemos crear una nueva conexión desde el menú contextual de este apartado (figura 8).

Se abre la ventana «Database Connection» (figura 9) donde podemos definir los parámetros de la

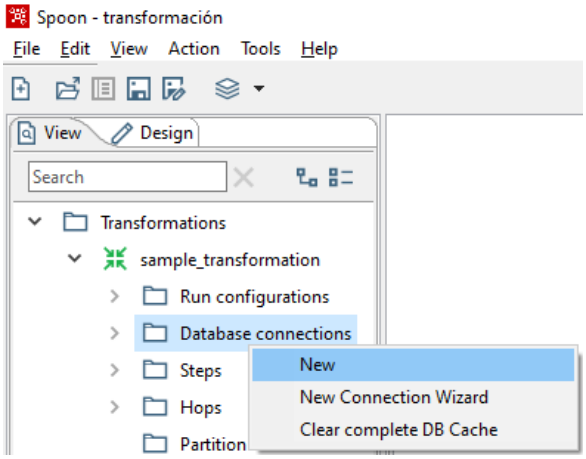

Figura 8: Crear una conexión con una BD.

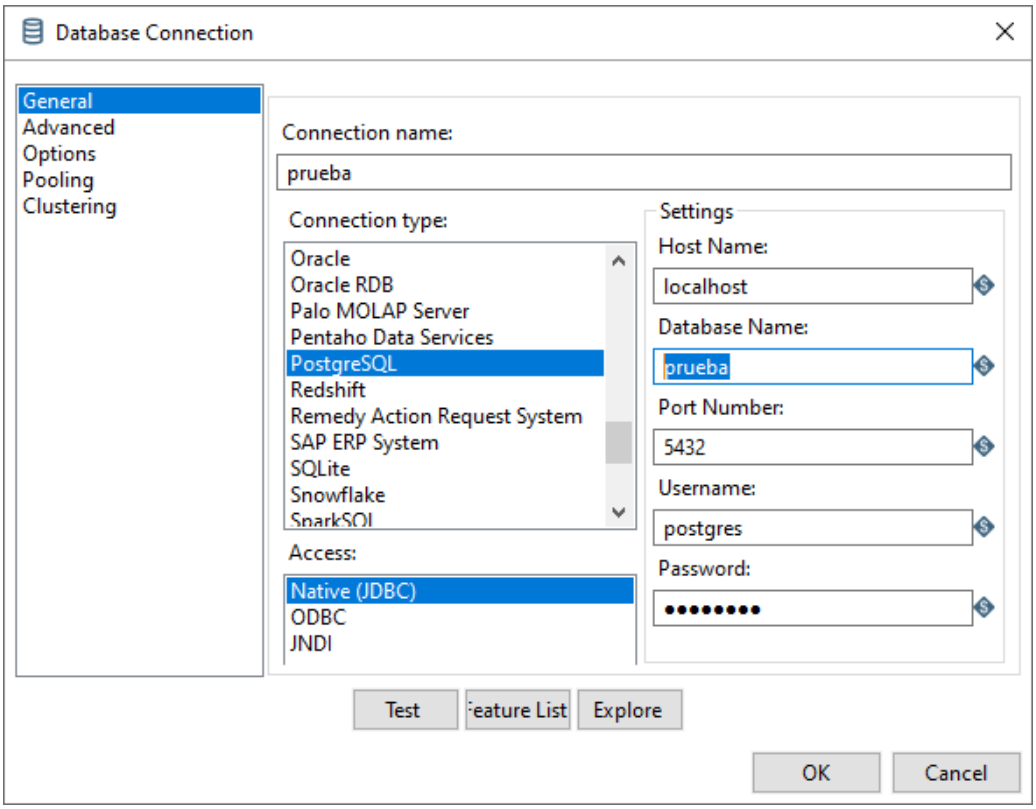

Figura 9: Parámetros de la conexión con *PostgreSQL*.

conexión. Es suficiente con completar los datos de la sección «General» que se muestran en la figura 9, asignándole un nombre adecuado a la conexión e indicando el nombre de la BD que vayamos a utilizar.

Una vez creada una conexión en una transformación, podemos compartirla con otras transformaciones seleccionando la opción «Share» del menú contextual de la conexión, de esta forma evitamos definir los mismos datos de la conexión repetidas veces.

Definir la estructura de una tabla en función del resultado Una posibilidad para adaptar o definir la estructura de una tabla a la del resultado que obtengamos en el proceso de transformación es utilizar el tipo de transformación «Table output» de la sección «Output» en la pestaña «Design».

En la configuración de la transformación «Table output» podemos seleccionar la conexión (en la conexión se define la BD) y, asociada de esta, el esquema y la tabla (figura 10). Si la tabla tiene campos creados, en la pestaña «Database fields» se puede definir la correspondencia entre los datos que llegan a la transformación y los campos de la tabla. Si la tabla que indicamos no existe o los campos que tiene no son adecuados para almacenar el resultado, podemos pulsar sobre el botón «SQL» y se abre la ventana «Simple SQL editor» en la que ha generado las sentencias SQL necesarias para crear o modificar a tabla indicada de manera que se adapte al resultado que queremos almacenar.

Podemos modificar la sentencia y después ejecutarla pulsando sobre el botón «Execute» de dicha ventana (figura 11). Otra posibilidad es ejecutar directamente la sentencia y después modificar lo necesarios en la estructura de la tabla desde *PgAdmin*, con la opción «Properties» del menú contextual de la tabla.

2.3. Generar sentencias SQL

2.3.1. Desde *PgAdmin*

Desde *PgAdmin*, una vez tenemos una tabla definida en la BD, podemos generar las sentencias SQL asociadas a la creación, borrado, inserción, selección o modificación desde el menú contextual de la tabla, pulsando sobre [«Scripts»] y seleccionando la opción de la operación correspondiente (figura 12).

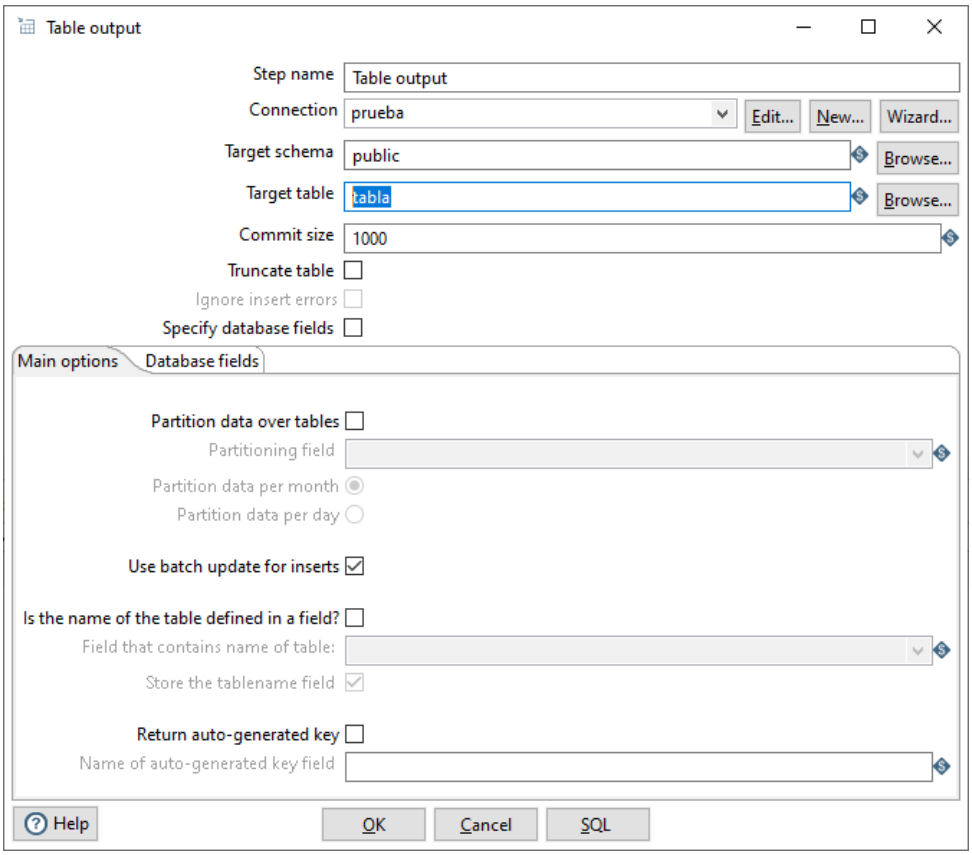

Figura 10: Almacenar el resultado de la transformación en una tabla.

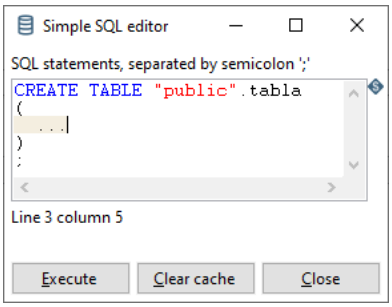

Figura 11: Crear una tabla generando su estructura automáticamente.

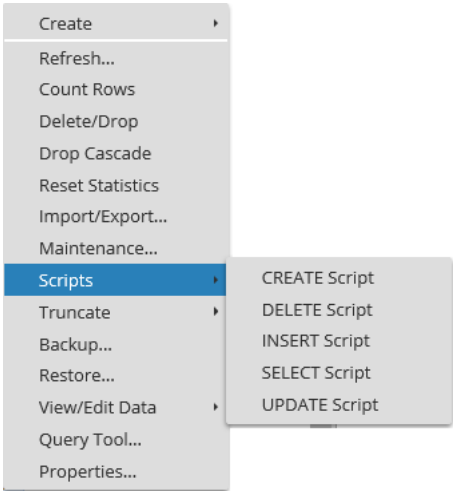

Figura 12: Generar SQL para una tabla desde *PgAdmin*.

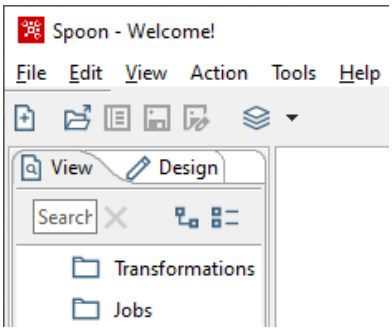

Figura 13: Pantalla de inicio de *Spoon*.

En particular, nos puede interesar generar el script de creación de la tabla, pulsando sobre [«Scripts», «CREATE Script»].

2.3.2. Desde *Spoon*

Desde *Spoon* podemos generar las sentencias SQL para insertar los datos del resultado en una tabla e incluso para crear previamente la tabla. Se puede llevar a cabo mediante la transformación «SQL file output» del apartado «Output» en la pestaña «Design». El resultado el un archivo con las sentencias SQL adecuadas a la configuración que hayamos definido.

3. Operaciones sobre *Spoon*

Se pueden consultar más detalles sobre *PDI* y *Spoon* en [Rol17] y [Rol18] (disponibles como recurso electrónico en la biblioteca de la UGR). A continuación, se incluye solo lo imprescindible para desarrollar esta actividad.

3.1. Crear transformación o trabajo

Podemos crear una transformación o un trabajo desde el menú contextual de las carpetas «Transformations» y «Jobs», respectivamente, de la pestaña «View» en la pantalla de inicio de *Spoon* (figura 13). Una vez creado uno cualquiera de ellos, se abre una ventana con el área de diseño y se activa la pestaña «Design» con carpetas con los distintos pasos que podemos añadir a la nueva transformación o nuevo trabajo creados.

Las carpetas organizan los elementos por temas, por ejemplo, para una transformación, en la carpeta «Input» se encuentran las posibilidades de entrada de datos que ofrece la herramienta.

3.2. Añadir un paso

Para añadir un paso a una transformación o a un trabajo, localizamos el paso que necesitamos en las carpetas de la pestaña «Design» y lo «pulsamos-arrastramos-soltamos» sobre el área de diseño. El resultado es que se añade un icono que representa el paso seleccionado. Pulsando «doble-click» sobre el icono se abre la ventana de configuración para definir su nombre y las características asociadas a él.

3.3. Conectar dos pasos

Para conectar dos pasos, al situar el cursor del ratón sobre un paso, en la parte inferior del mismo se abre un menú asociado, una de las opciones representa una flecha de salida (figura 14), al pulsar sobre ella y, a continuación, pulsar sobre otro paso, se establece una conexión entre ambos pasos. Mediante una conexión en el paso destino se puede acceder a los datos obtenidos como resultado del paso origen.

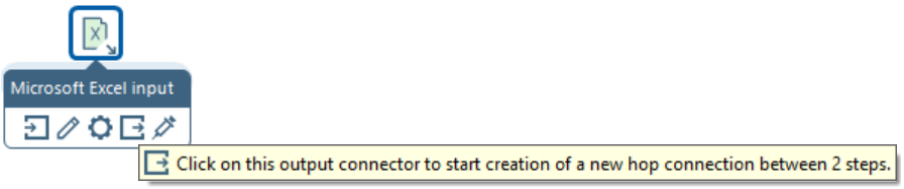

Figura 14: Conectar pasos de una transformación o trabajo.

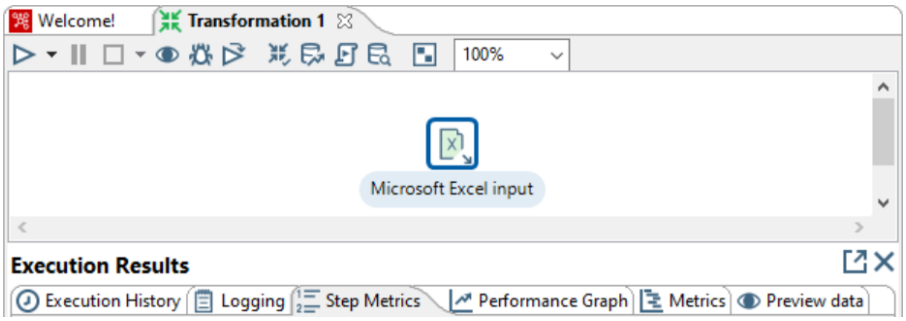

Figura 15: Ejecutar una transformación o trabajo.

3.4. Ejecutar y analizar una transformación o trabajo

Podemos ejecutar y analizar una transformación o trabajo mediante la barra de iconos de la parte superior del área de diseño (figura 15).

Cuando pulsamos sobre «Run» (primer icono por la izquierda), se abre una ventana donde se pueden definir parámetros y variables, junto a varias opciones de ejecución. Podemos dejar las opciones por defecto y pulsar el botón «Run» de esta ventana.

Los resultados obtenidos se pueden ver en las pestañas que se abren en la parte inferior de dicha ventana. En caso de no abrirse automáticamente estas pestañas, podemos mostrarlas pulsando sobre el icono «Show the execution results pane» de la barra de iconos (el primero por la derecha); al estar las pestañas activas, el icono se transforma en «Hide the execution results pane» para ocultarlas.

3.5. Detalles de transformaciones específicas

En este apartado se presentan detalles de algunas transformaciones que seguramente será necesario usar en esta actividad.

3.5.1. Leer un archivo *Excel*

La transformación «Microsoft Excel input» permite leer a la vez varios archivos con la misma estructura: los archivos seleccionados se añaden a una lista en la pestaña «Files». Hay que configurar algunos datos adicionales en el resto de pestañas, por ejemplo, las hojas que se leen: si se leen varias hojas de un archivo, todas han de tener la misma estructura. Respecto a los campos, en la pestaña «Fields», pulsando sobre el botón «Get fields from header row» se obtienen los nombres y los tipos que deduce; los tipos se pueden modificar según nuestro conocimiento de los datos; una vez definida la estructura podemos probar el resultado pulsando sobre el botón «Preview rows» en esa misma pestaña.

3.5.2. Seleccionar campos

Podemos seleccionar un subconjunto de los campos de entrada mediante la transformación «Select values». En la pestaña «Select & Alter» hay que definir los campos que se desean incluir cambiando el nombre o directamente (si se incluyen sin indicar un nuevo nombre). En la pestaña «Remove» se definen los campos que se desea excluir del resultado de entre los que se hayan indicado en la pestaña «Select & Alter».

Es decir, podemos limitarnos a indicar en la pestaña «Select & Alter» los campos que queremos incluir en el resultado.

3.5.3. Generar una tabla

La transformación «Table output» nos permite generar el contenido de una tabla de una BD. En la ventana de configuración, seleccionamos la conexión con la BD, el esquema y la tabla. Si todavía no hemos definido una conexión con la BD, tenemos que definirla previamente (apartado 2.2.3). Si la tabla no existe o no tiene la estructura adecuada para almacenar los datos, pulsando sobre el botón «SQL» genera las sentencias SQL necesarias para generar la tabla adecuada. Para ejecutar esas sentencias, hemos de pulsar sobre el botón «Execute» de la ventana que las muestra.

Una vez creada la tabla con la estructura que la herramienta ha considerado adecuada, podemos refinarla desde *PgAdmin*. Por ejemplo, los campos numéricos los suele generar de tipo `DOUBLE` aunque los datos sean de tipo entero: desde PgAdmin, seleccionando la opción «Properties» del menú contextual de la tabla correspondiente, en la pestaña «Columns» de la ventana que se abre podemos redefinir adecuadamente los tipos de los campos; en el resto de pestañas podemos definir otras características que nos pueden interesar de la tabla.

Si vamos a hacer pruebas (como es el caso) es recomendable seleccionar la opción «Truncate table» en la ventana de configuración de la transformación «Table output» para que, cada vez que se ejecute la transformación, elimine previamente los datos de la tabla. Esta opción tiene el inconveniente de que si definimos llaves externas, el sistema bloquea la eliminación de los registros. Por este motivo, se recomienda no definir llaves externas entre las tablas. No hay problema con las llaves primarias, se recomienda definirlas.

4. Transformaciones a realizar

Para definir las transformaciones utiliza *PDI* mediante *Spoon*.

4.1. Transformaciones para obtener el resultado inmediato

En primer lugar, vamos a definir las transformaciones para obtener el resultado final reutilizando todo lo posible de la actividad anterior. Necesitamos disponer en *PosgreSQL* de una BD con tablas con la estructura de los hechos y las dimensiones definidas, con los datos correspondientes introducidos.

1. Crea una BD *PosgreSQL* cuyo nombre sea el nombre de la provincia que tienes asignada y tu nombre de usuario de correo UGR (en mi caso se llamará `granada_jsamos`). En el esquema `public` de esa BD crea las tablas `cuando`, `donde` y `padron` añadiendo al nombre el sufijo de tu nombre de usuario de correo UGR (en mi caso se llamarán `cuando_jsamos`, `donde_jsamos` y `padron_jsamos`). La estructura de estas tablas ha de ser similar a la de las hojas correspondientes del archivo obtenido con *Power Query* en la actividad anterior, **usa estas hojas como origen de datos** (en mi caso el archivo es `granada-ETL-jsamos.xlsx`) pero, para los nombres de los campos, usa minúsculas sin tilde y, en lugar del criterio *Camel Case* (`CamelCase`), usa el carácter «_» como separador (`camel_case`).

   - Para cada tabla define una transformación distinta cuyo nombre sea el nombre de la provincia que tienes asignada, el literal «-ETL-», tu nombre de usuario de correo UGR y el nombre de la tabla (en mi caso, por ejemplo, una transformación se llamará `granada-ETL-jsamos-cuando`).
   - El nombre de cada paso de las transformaciones ha de tener como sufijo tu nombre de usuario de correo UGR (por ejemplo, en mi caso, un paso se llamará `escribe-tabla-jsamos`).
   - Define un tabajo («Jobs») que controle la ejecución de las transformaciones definidas previamente tanto en caso de que todo funcione bien como si se producen errores.

4.2. Transformaciones alternativas

2. Crea una BD *PosgreSQL* cuyo nombre sea `prueba` y tu nombre de usuario de correo UGR (en mi caso se llamará `prueba_jsamos`). En el esquema `public` de esa BD crea la tabla `cuando` añadiendo al nombre el sufijo de tu nombre de usuario de correo UGR (en mi caso se llamará `cuando_jsamos`). La estructura de esta tabla ha de ser similar a la de la hoja correspondiente del archivo obtenido con *Power Query* en la actividad anterior (en mi caso el archivo es `granada-ETL-jsamos.xlsx`) pero, para los nombres de los campos, usa minúsculas sin tilde y, en lugar del criterio *Camel Case* (`CamelCase`), usa el carácter «_» como separador (`camel_case`). Define el contenido de esa tabla mediante una transformación **usando como origen la hoja `Provincia`** del archivo generado mediante *Power Query* en la actividad anterior.

   - Llama a la transformación `cuando` más tu nombre de usuario de correo UGR (en mi caso, por ejemplo, una transformación se llamará `cuando-jsamos`).
   - El nombre de cada paso de las transformaciones ha de tener como sufijo tu nombre de usuario de correo UGR (por ejemplo, en mi caso, un paso se llamará `escribe-tabla-jsamos`).

3. **Ejercicio libre**: para realizar la transformación 2, en lugar de la hoja `Provincia` del archivo generado con *Power Query*, considera el archivo original de tu provincia. Para transformar la tabla dinámica de partida se puede usar la transformación «Row normaliser» ubicada en la carpeta «Transform» de la pestaña «Design».

Bibliografía

[Rol17] María Carina Roldán. *Learning Pentaho Data Integration 8 CE (Third Edition).* Packt Publishing, 2017.

[Rol18] María Carina Roldán. *Pentaho Data Integration Quick Start Guide.* Packt Publishing, 2018.