

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

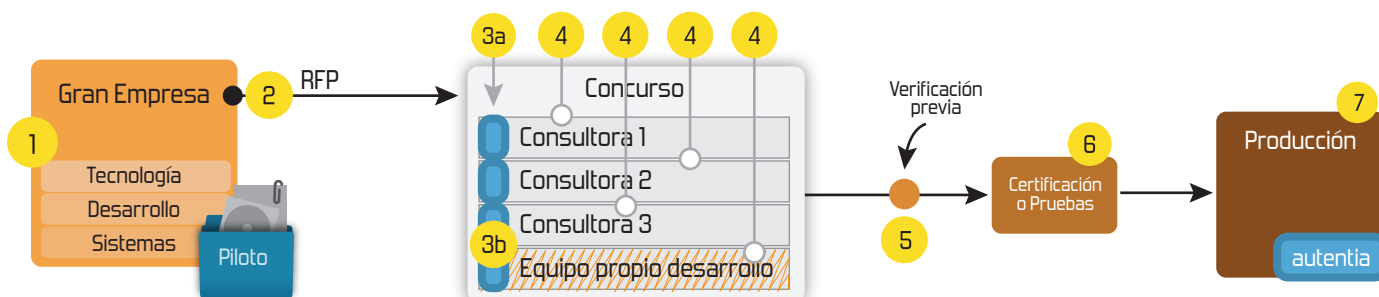
1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

Control de autenticación y
 acceso (Spring Security)
 UDDI
 Web Services
 Rest Services
 Social SSO
 SSO (Cas)

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)



» Estás en: Inicio » Tutoriales » Talend. Lectura y tratamiento de base de datos Mysql.



Daniel Casanova Frutos

Consultor tecnológico de desarrollo de proyectos informáticos.

Ingeniero Técnico En Informática De Sistemas por la Universidad Alfonso X El Sabio.

Puedes encontrarme en [Autentia](#): Ofrecemos servicios de soporte a desarrollo, factoría y formación

Somos expertos en Java/J2EE

[Ver todos los tutoriales del autor](#)



Fecha de publicación del tutorial: 2012-09-06

Tutorial visitado 6 veces [Descargar en PDF](#)

Talend. Lectura y tratamiento de base de datos Mysql.

0. Índice de contenidos.

- 1. Introducción.
- 2. Entorno.
- 3. Tratamiento de la información de una base de datos Mysql.
- 4. Referencias.
- 5. Conclusiones.

1. Introducción

Este tutorial nos enseña de manera sencilla la extracción, manipulación e inserción de información contenida en una base de datos Mysql, mediante la herramienta **Talend**.

Talend, como ya se ha enseñado en [tutoriales](#) publicados con anterioridad, es una herramienta de diseño **ETL (Extract, Transform, Load)**, es decir, su finalidad es extraer datos, transformación y carga de los mismo, a partir de distintas fuentes de datos como pueden ser ficheros, conexiones a distintas bases de datos y un sin fin de posibilidades.

2. Entorno.

El tutorial está escrito usando el siguiente entorno:

- Hardware: Portátil MacBook Pro 15' (2.2 GHz Intel Core i7 Duo, 8GB DDR3 SDRAM).
- Sistema Operativo: Mac OS X Lion 10.7.3 (11D50d)
- Talend Open Studio V5.1.1 r84309

3. Tratamiento de la información de una base de datos Mysql.

En este capítulo veremos como Talend permite realizar de una manera sencilla una de las tareas más habituales en cualquier sistema de información, veremos como obtener y manipular la información de una base de datos cualquiera y hacer uso de la misma como un flujo de salida hacia cualquier tipo de fichero o incluso hacia otra base de datos.

Debemos entender que Talend ofrece componentes específicos para cada tipo de base de datos, con un funcionamiento similar entre ellos, pero con ciertas particularidades en cada uno de ellos, por lo que en este ejemplo realizamos el procesamiento de la información sobre una base de datos MySQL. El resto de componentes de las distintas bases de datos tienen todos un comportamiento similar.

Lo primero de todo es crear un nuevo trabajo en el entorno de Talend. Para ello realizamos click derecho sobre 'Job Design' para crear el trabajo y le asignamos un nombre. Trás pulsar en finalizar ya tendremos listo nuestro entorno de trabajo en Talend.

El principal componente de manejo de información proveniente de base de datos Mysql en Talend es el componente **'tMysqlInput'**, que encontramos en la categoría de componentes *Databases/Mysql*.

Catálogo de servicios Autentia



Síguenos a través de:



Últimas Noticias

» Estreno Terrakas 1x04: "Terraka por un día"

» Nuevos cursos de gestión de la configuración en IOS y Android

» La regla del Boy Scout y la Oxidación del Software

» Autentia conquista los Alpes

» Orientación a objetos y la importancia del "Tell, Don't Ask"

[Histórico de noticias](#)

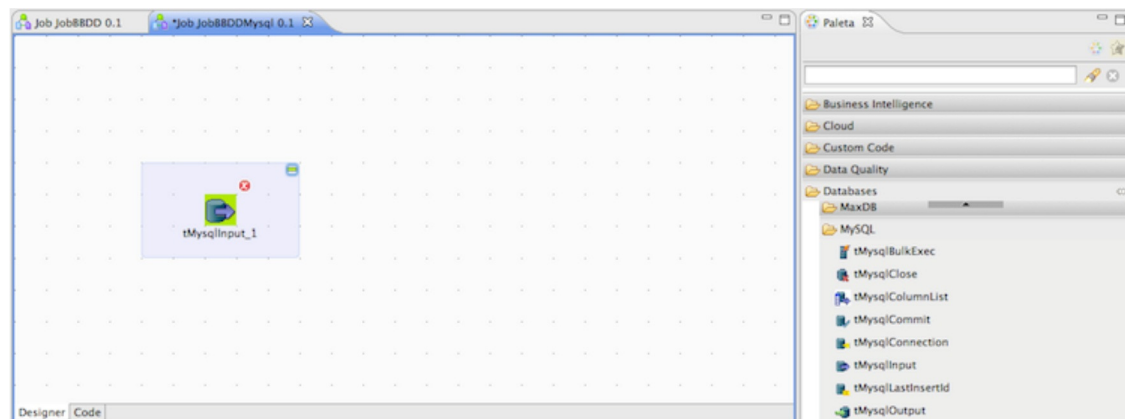
Últimos Tutoriales

» [Lectura y tratamiento de ficheros XML con Talend](#)

» [Desplegando una aplicación en Cloud Foundry con Maven](#)

» [Primeros pasos en Android \(II\)](#)

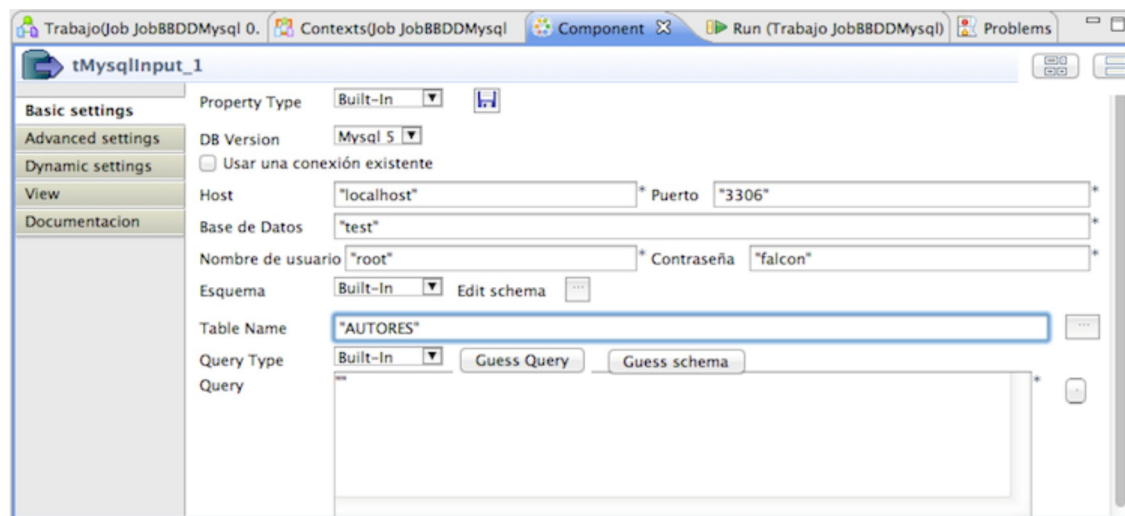
Colocamos un componente de este tipo en nuestro panel de trabajo:



A continuación con un doble click sobre el componente veremos su configuración principal:

- **Property type:** Permite seleccionar si las propiedades del componente se crean de manera dinámica o a través de metadatos ya definidos con anterioridad al trabajo
- **DB Version:** Versión de la base de datos que estamos utilizando.
- ☐ **Usar una conexión existente:** Es la manera más habitual de definir la configuración de nuestra conexión a base de datos. Esto es utilizando otro tipo de componente (*tMysqlConnection*) donde se define nuestra conexión, y posteriormente utilizamos esta misma.
- **Host:** Nombre del host de conexión
- **Puerto:** Puerto de la conexión
- **Base de Datos:** Nombre de la base de datos
- **Mysql Schema:** Como su nombre indica, nombre del esquema en la base de datos
- **Nombre del usuario:** Usuario de la base de datos
- **Contraseña:** Contraseña de la base de datos
- **Esquema:** Definición del esquema en el mismo componente mediante la opción '*Built-In*', o escoger la definición del esquema de un repositorio previamente definido en Talend mediante la opción '*Repository*'
- **Edit schema:** Definición del esquema de información origen
- **Table name:** Nombre de la tabla de la cual extraer la información
- **Query:** Query formada de manera dinámica por Talend al definir el esquema. Para actualizar dicha query usar el botón 'Guess schema'.

Estas será la configuración mínima para el funcionamiento de dicho componente. Veamos un caso practico de configuración:



Como podemos ver hemos configurado nuestra conexión a BBDD y posteriormente hemos configurado que la tabla de la que extraer la información sera la tabla de nombre 'AUTORES'.

Nuestra tabla AUTORES tiene las siguiente definición en base de datos:

Field	Type	Null	Key	Default	Extra
nombre	varchar(100)	YES			
apellido1	varchar(100)	YES			
apellido2	varchar(100)	YES			
direccion	varchar(100)	YES			

Los registros de la tabla AUTORES son:

Field	Type	Null	Key	Default	Extra
-------	------	------	-----	---------	-------

» Eclipse Juno, la versión 4.2 de Eclipse

» Double Opt-In y autologin con el soporte de Spring MVC y Spring Security.

Últimos Tutoriales del Autor

» Lectura y tratamiento de ficheros XML con Talend

» Lectura y tratamiento de ficheros Excel con Talend (!): nociones básicas.

» JMeter. Uso de funciones.

» JMeter. Gestión de usuarios

» JMeter y JSF. Extracción del parámetro ViewState

Últimas ofertas de empleo

2011-09-08
Comercial - Ventas - MADRID.

2011-09-03
Comercial - Ventas - VALENCIA.

2011-08-19
Comercial - Compras - ALICANTE.

2011-07-12
Otras Sin catalogar - MADRID.

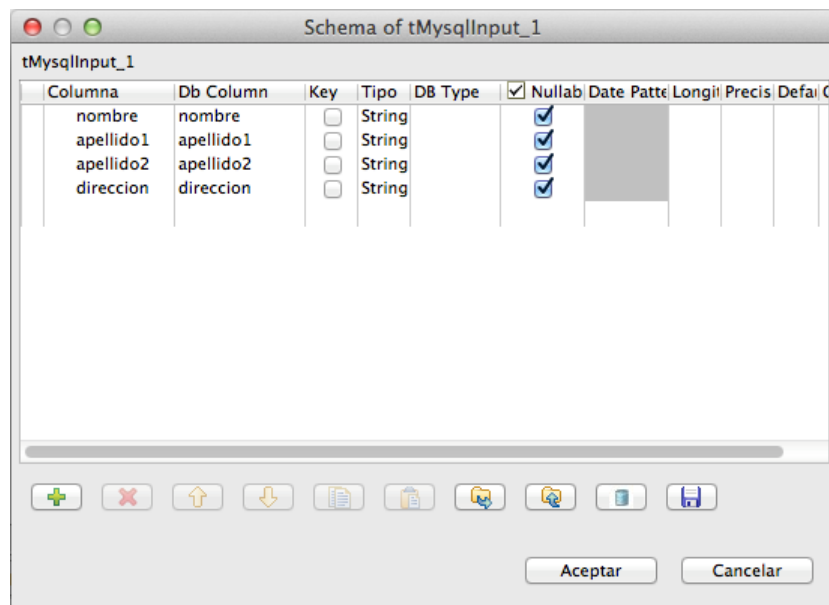
2011-07-06
Otras Sin catalogar - LUGO.

```

| nombre | apellido1 | apellido2 | direccion |
+-----+-----+-----+-----+
| Daniel | Casanova | Frutos | C/San Jeronimo 1 Portal G 5C |
| Raul | Lopez | Martín | C/Marco 45 1B |
| Mario | López | Delgado | C/Paraiso 29 F 1A |
+-----+-----+-----+-----+

```

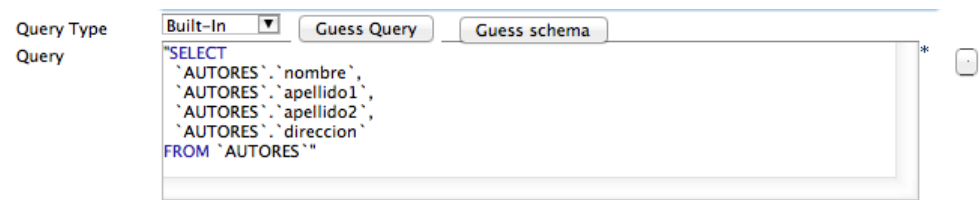
A continuación editamos el esquema definiendo el nombre y tipo de las columnas de la tabla, mediante la opción '*Edit Schema*':



Como podemos ver básicamente indicamos el nombre que queremos dar a la columna en el campo '**Columna**', el nombre real de la columna en base de datos en el campo '**Db Column**' y el tipo de flujo de salida, que en este caso definimos todo como String.

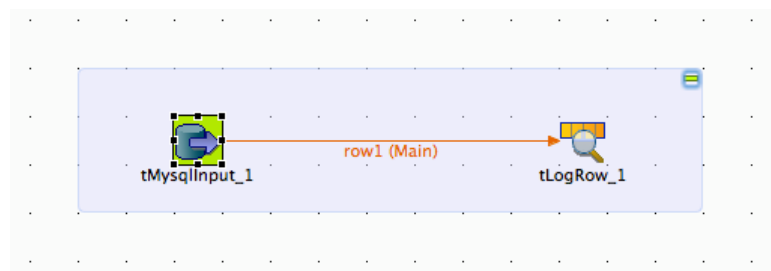
Se puede configurar también el tipo de columna en base de datos (DB Type), si es nutable o no el valor de la columna (Nullable) y una serie de características más sobre las columnas.

Al definir el esquema podemos pulsar sobre la opción 'Guess Query' y observar como el campo '**Query**' de las preferencias del componente se ha autocompletado conformando la query a través de la cual obtendra el flujo de información de la base de datos:



Se puede realizar una prueba directa de dicha query contra la base de datos pulsando en la opción '**Guess schema**'

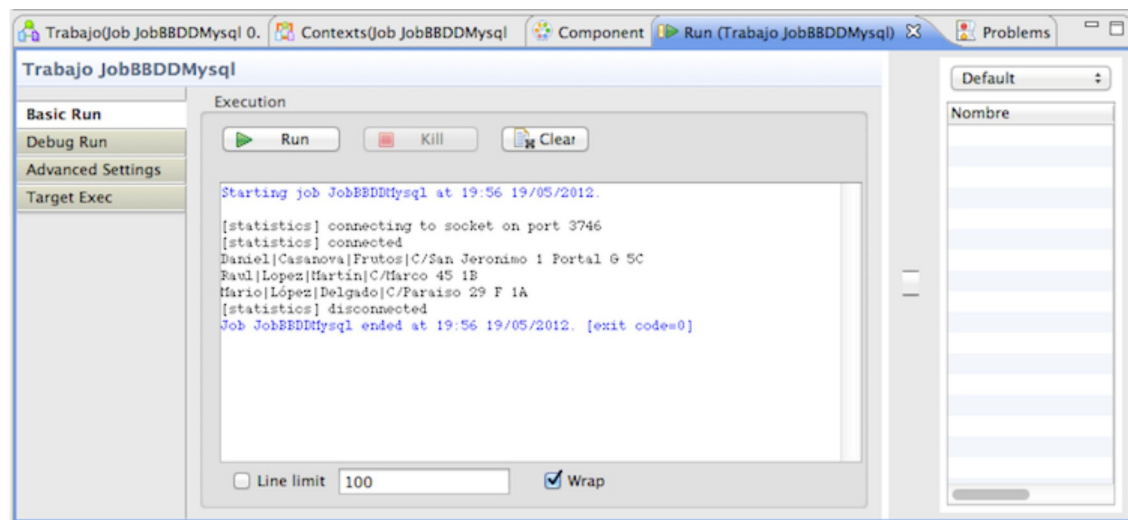
A continuación utilizamos el componente de log '*tLogRow*' para observar los resultados del flujo de salida. Para ello lo colocamos desde la paleta de componentes y propagamos el flujo del componente de entrada de base de datos hacia dicho componente. Para ello realizamos click derecho sobre el componente de base de datos, opción '**Fila/Main**' y clickamos sobre el componente de log resultando:



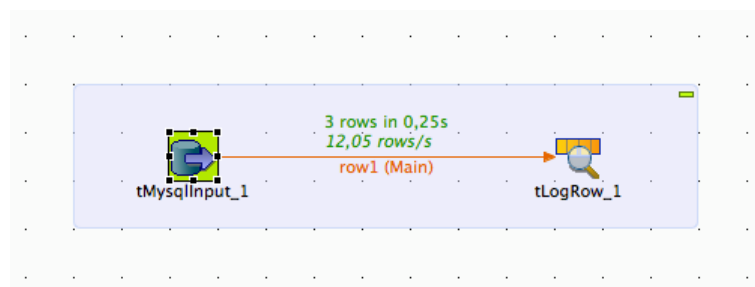
Para observar el esquema del flujo de salida pulsamos en la opción '**Edit schema**' de las preferencias del componente tLogRow.

Podemos observar como el flujo de salida corresponde por defecto con cada una de las columnas del flujo de salida que proviene del componente tMysqlInput.

Por último podemos ejecutamos el trabajo mediante el botón '**Play**' y observamos como el flujo de salida obtenido es el adecuado:



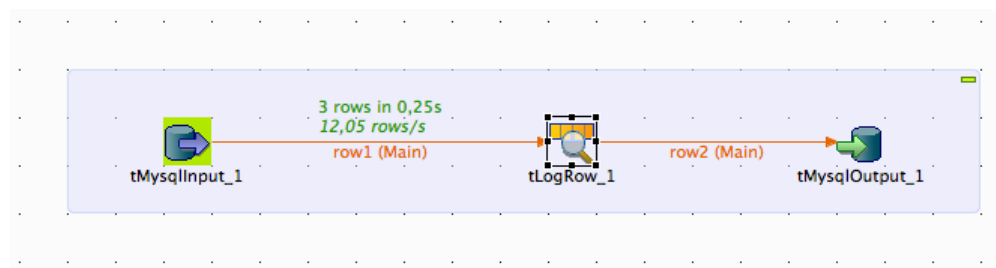
Podemos observar como en el area de trabajo nos marca el número de filas del flujo de salida recuperadas, así como el tiempo que ha tardado, y la media de filas por segundo:



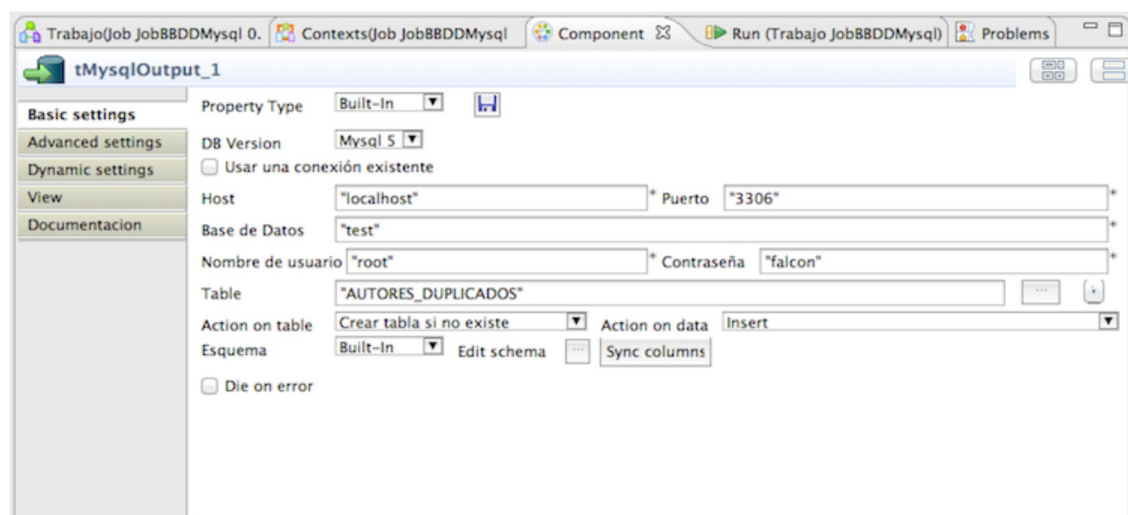
En resumen, el componente **tMysqlInput** nos permite recuperar información de cualquier base de datos del tipo Mysql, sea cual se la tabla, y nos permite filtrar las columnas así como manipular la información de salida (tipología, nombres...).

A continuación veremos uno de los componentes más interesantes en cuanto a tratamiento de la información de base de datos. Se trata del componente **tMysqlOutput**. Como su nombre indica sirve para volcar un flujo de información en una tabla en concreto de base de datos.

Colocamos un compoente de este tipo en nuestro area de trabajo y propagamos el flujo de salida del componente tlogRow hacia dicho componente (click derecho sobre tLogRow, Fila, Main y click sobre el componente tMysqlOutput):



De momento configuramos el componente con la misma conexión que el componente tMysqlInput (Ya que la base de datos donde volcar el flujo de salida es el mismo) y con la particularidad de configurar la propiedad **'Table'** con el nombre de **'AUTORES_DPLICADOS'**, es el nombre de la tabla donde volcamos la información:



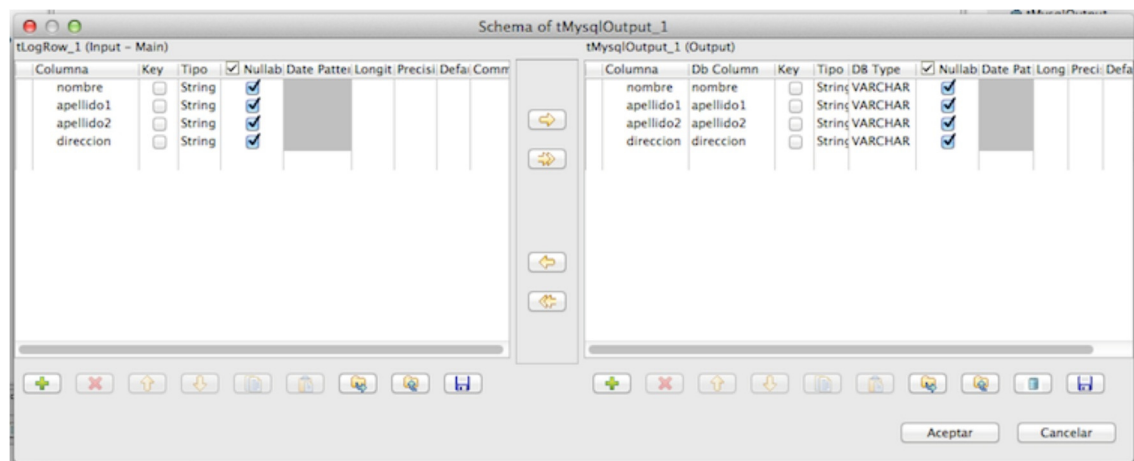
Otra de las propiedades interesantes de este componente son:

- **Action on table:** Acción a realizar sobre el esquema acerca de la tabla de destino. Por ejemplo crear la tabla sino existe, limpiarla antes de insertar...

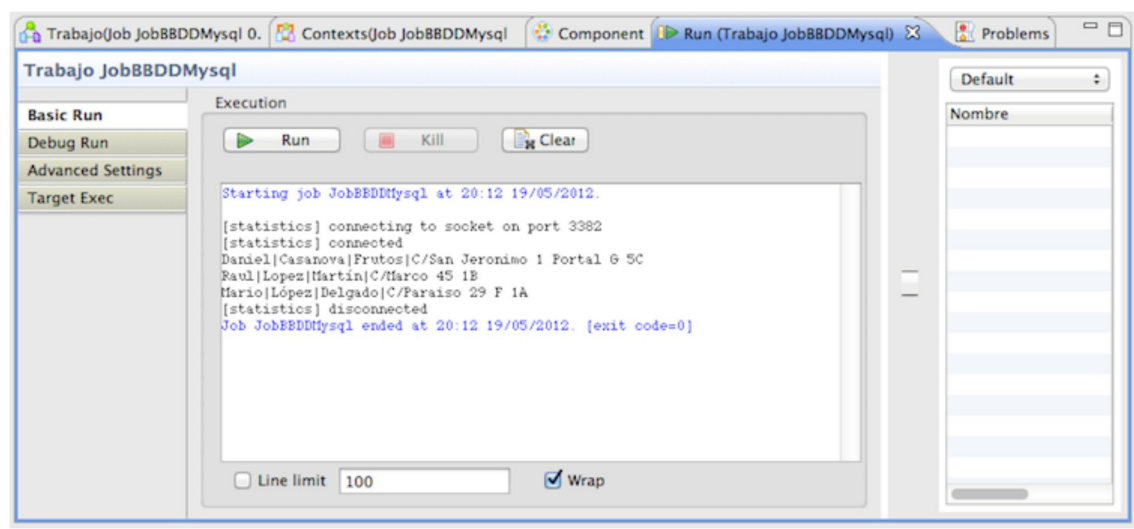
Talend. Lectura y tratamiento de base de datos MySQL.

- **Action on data:** Acción a realizar con el flujo de información, por ejemplo la acción de insert o insertar los registros
- **Die on error:** Interesante acción ya que al checkearla quiere decir que si en algún momento el trabajo falla, este se pare por completo, es decir, no siga volcando el resto de información.

Pulsamos en la opción 'Edit Schema' para poder ver como el flujo de datos definido es el adecuado, es decir, por defecto se creará la tabla AUTORES_DUPLICADOS con las mismas columnas y registros que la tabla de la que proviene el flujo de datos, AUTORES:

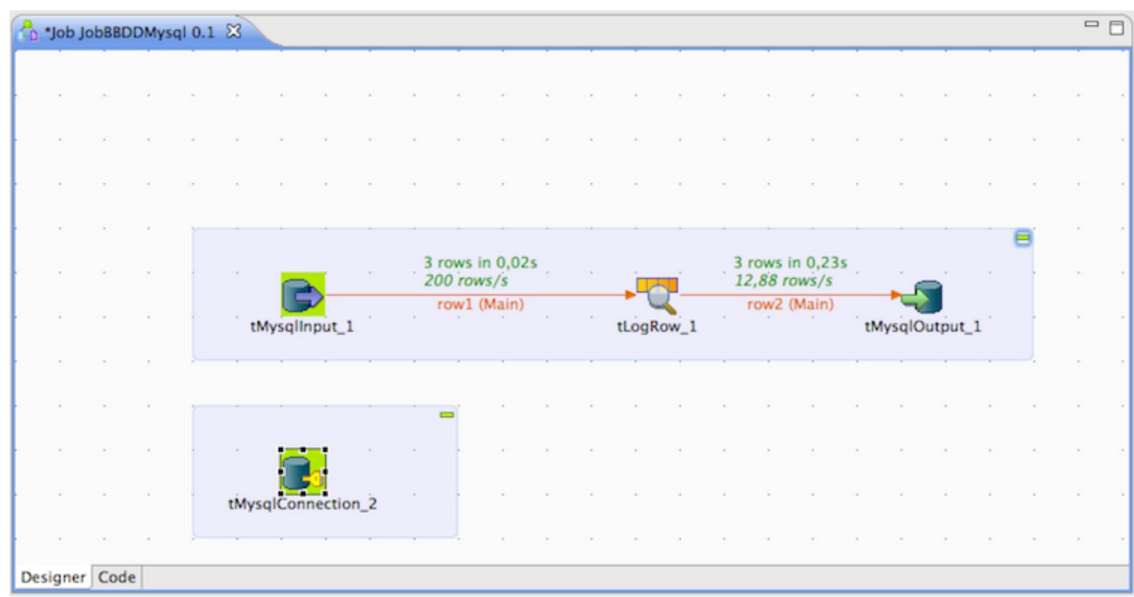


A continuación ejecutamos el trabajo y podemos observar en nuestro esquema como la tabla AUTORES_DUPLICADOS se ha creado de manera adecuada y se ha rellenado con los datos de manera correcta:

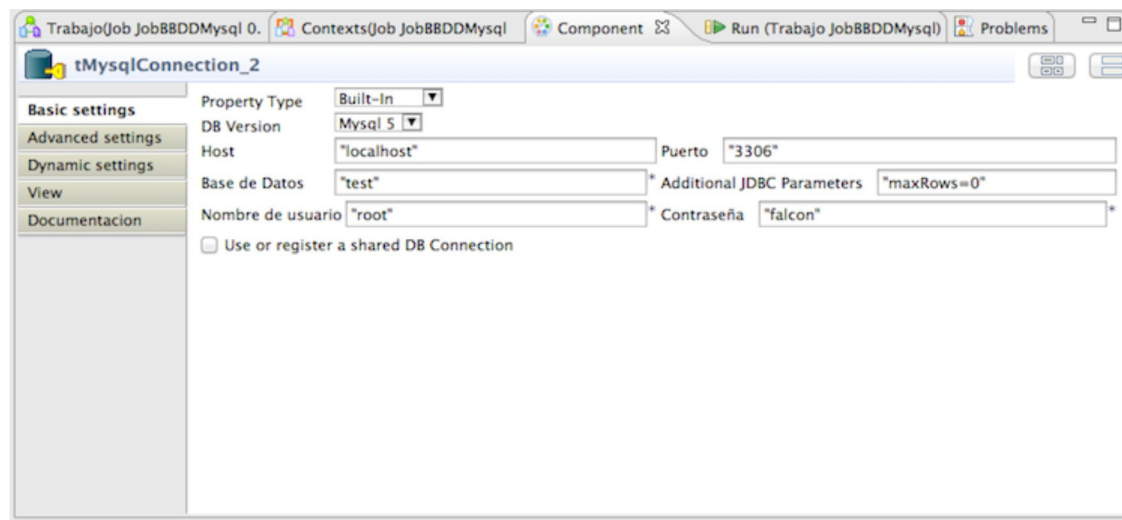


Veamos ahora el componente **tMySQLConnection**. Dicho componente nos permite configurar una conexión a base de datos, y reutilizar dicha conexión en otros componentes, como pueden ser del tipo tMySQLInput o tMySQLOutput.

Colocamos un componente de este tipo en nuestro espacio de trabajo y lo configuramos con los datos de conexión utilizados anteriormente en los componentes de entrada y salida de datos:



Preferencias:

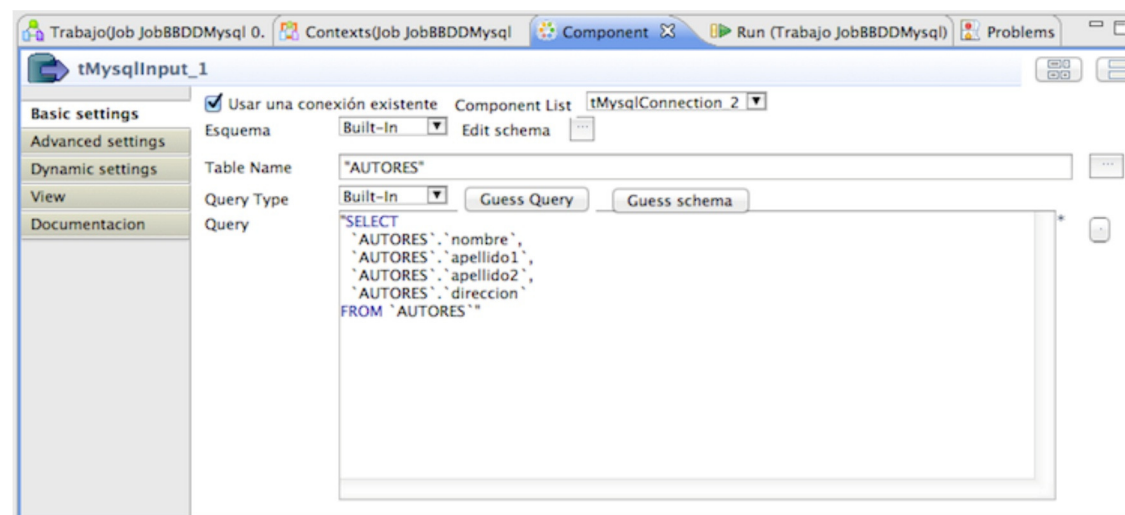


Como podemos observar las propiedades del componente son practicamente las mismas que los componentes de entrada y salida de Mysql. Añade las siguientes propiedades:

- **Additional JDB Parameters:** Permite especificar parámetros adicionales para la conexión a la base de datos, en nuestro ejemplo añadimos el parámetro 'maxRows=0', para comprobar que las preferencias funcionan de manera adecuada. Dicho parámetro indica que nos devuelva todas las filas de la base de datos.
- **Use or register a shared DB Connection:** Permite definir una conexión compartida de manera que se pueda utilizar en otros Jobs (trabajos).

Una vez configurado de manera adecuado dicho componente, nos disponemos a usarlo en el resto de componentes de nuestro trabajo, evitando así tener que configurar la conexión a la base de datos en cada uno de ellos.

Para ello accedemos a la configuración del componente tMysqlInput y tMysqlOutput para indicar '**Usar una conexión existe**', marcando dicho check:

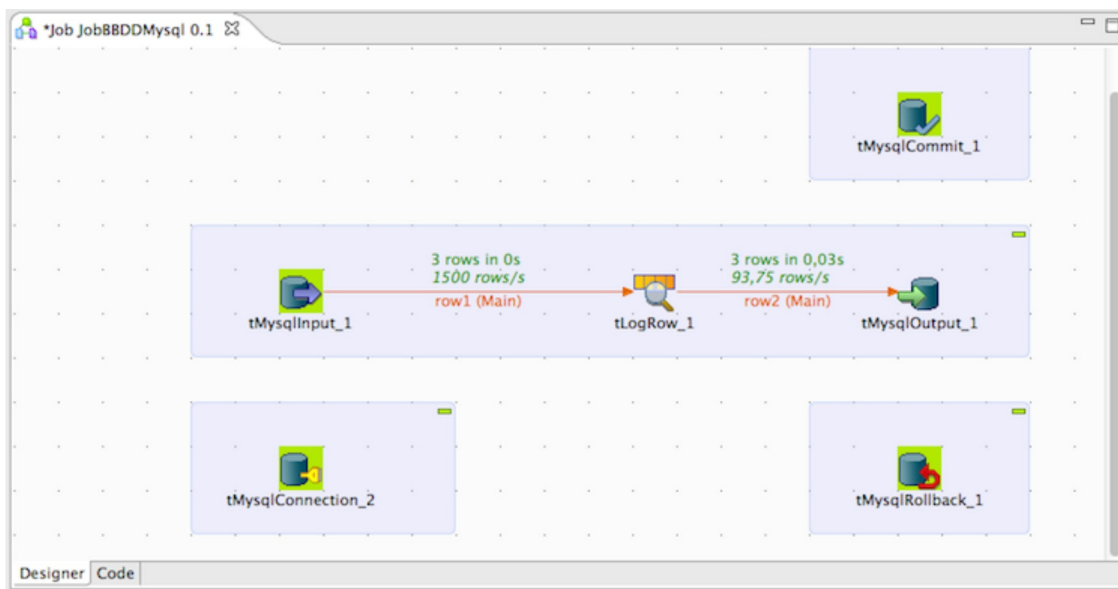


A continuación, y tras ejecutar el trabajo de nuevo podemos observar que los resultados obtenidos son los adecuados, es decir que los registros se siguen leyendo de la tabla 'autores' de base de datos, y se siguen volcando en la tabla 'AUTORES_DUPLICADOS', al igual que cuando cada componente definía su propia conexión.

En el siguiente paso del tutorial explicamos dos componentes utiles a la hora de trabajar con conexiones a base de datos en la herramienta Talend. Dichos componentes son **tMysqlCommit** y **tMysqlRollback**.

Dichos componentes, como su nombre indica, son los encargados de realizar la acción de commit o rollback, tras realizar una acción determinada sobre la base de datos. Como todos deberíamos saber al trabajar con bases de datos, la acción de commit consiste en confirmar los cambios realizados en la base de datos; y la acción de rollback consiste en deshacer todos los cambios realizados en una base de datos durante la ejecución de un proceso en la misma.

Para ello colocamos dichos componentes en nuestra zona de trabajo:



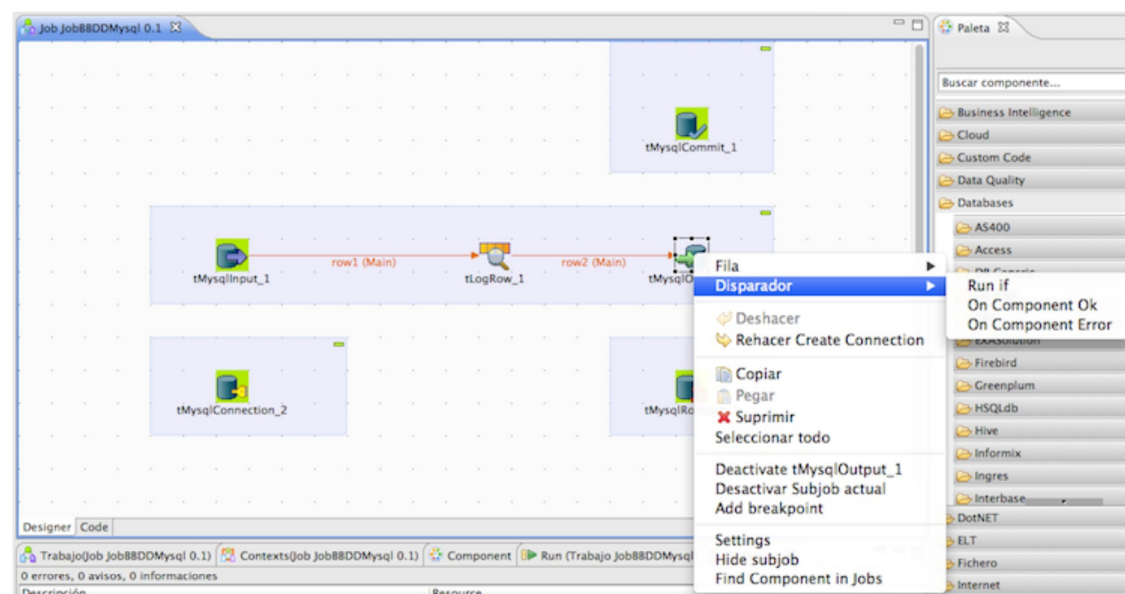
A continuación, utilizaremos una funcionalidad de la que se encuentran dotados ciertos componentes en la herramienta Talend. Dicha característica son los denominados '**Disparadores**'. Los disparadores permiten o no desviar el flujo de ejecución de nuestro Job (trabajo), en función de si se ha producido o no un error en el desarrollo de la acción a realizar por el componente donde se configura el disparador.

Tenemos tres tipos de disparadores condicionales en el componente:

- **Run if:** Dejar pasar el flujo en caso de que se cumpla una condición.
- **On component ok:** Dejar pasar el flujo en caso de que todo haya ido bien.
- **On component error:** Dejar pasar el flujo en caso de que haya habido un error.

En nuestro ejemplo configuraremos el componente de salida a base de datos 'tMysqlOutput' para que en función del resultado de su ejecución sea correcto o no, realice una acción u otra. El disparador 'On component ok' desviará el flujo para realizar un commit, es decir, hacia el componente tMysqlCommit. Y el disparador 'On component error' desviará el flujo hacia el componente tMysqlRollback, para realizar un rollback sobre base de datos, es decir, deshacer los cambios realizados en la misma en caso de error.

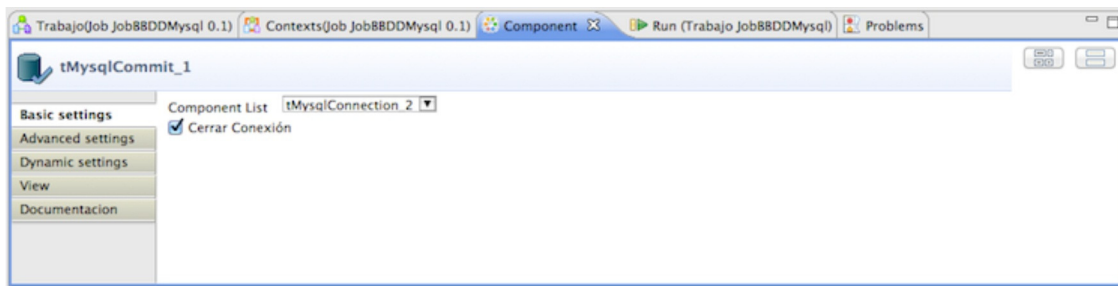
Para ello realizamos un click derecho sobre el componente tMysqlOutput seleccionando la opción 'Disparador/On component ok' y pulsando sobre el componente tMysqlCommit:



Realizamos la misma acción para establecer el disparador 'On component error' hacia el componente tMysqlRollback.

En el último paso del tutorial, veremos que siempre que realicemos una acción sobre una base de datos, es aconsejable cerrar la conexión hacia la misma, de lo contrario se quedarán abiertas conexiones que aumentan el consumo de memoria de nuestra base de datos.

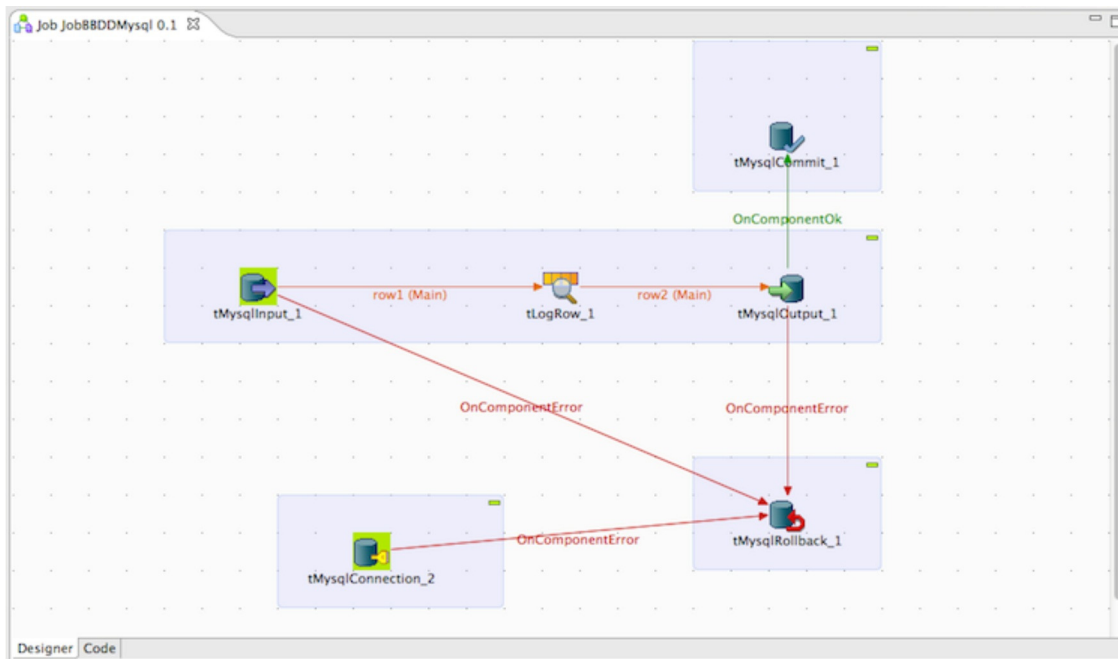
Si nos fijamos en las preferencias de los componentes encargados de realizar commit o rollback tienen una opción para cerrar la base de datos tras realizar su trabajo:



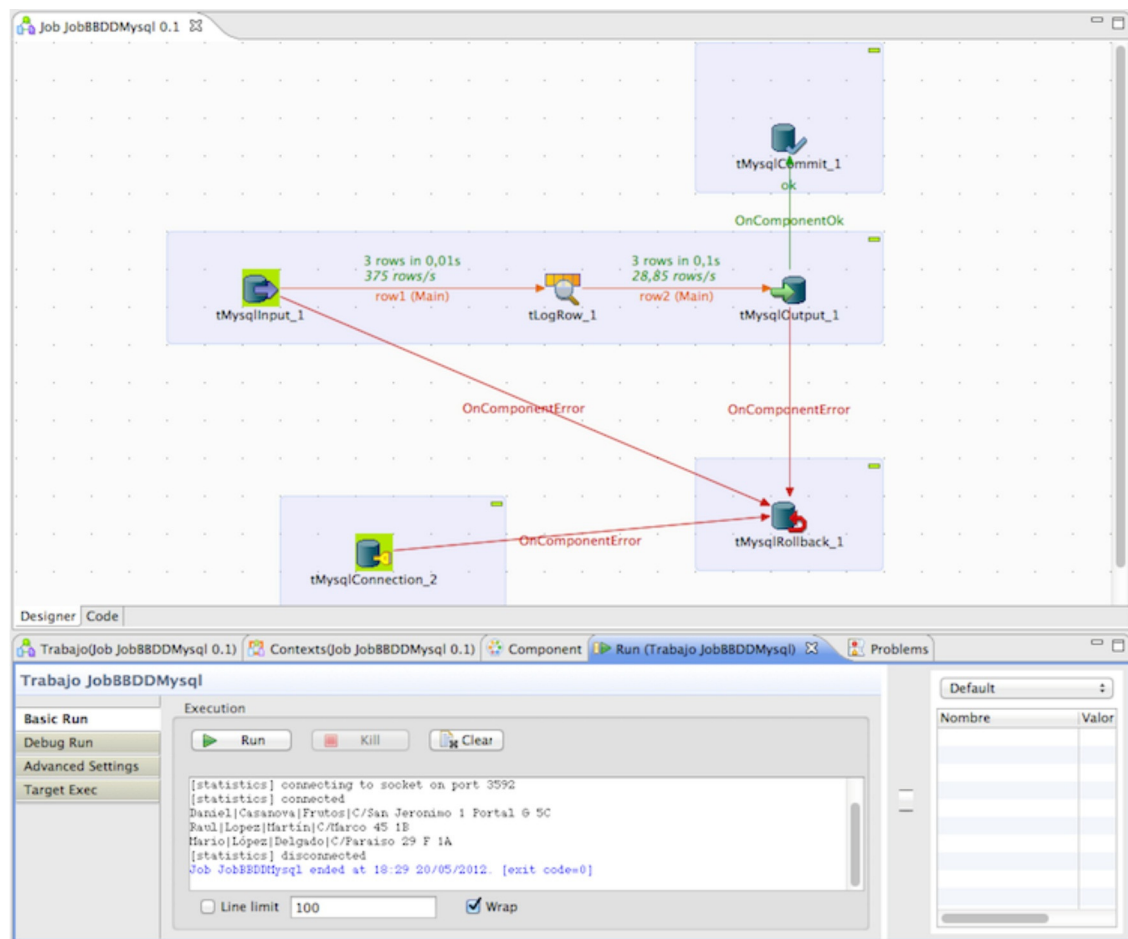
Otra manera de realizar esta acción es mediante el componente **tMysqlClose**, cuya finalidad es cerrar una conexión de base de datos. Para ello se configura con el nombre del componente sobre el cual debe cerrar la conexión. En este ejemplo no es necesario su uso.

Aún así, es aconsejable que todos los componentes que realicen una acción sobre base de datos, y por lo tanto, abran o utilicen una conexión contra la misma, deberían en caso de producir algún error, cerrar la conexión a base de datos, por ejemplo utilizando disparadores 'On component error'.

De esta manera el esquema resulta:



Para finalizar, ejecutamos nuestro trabajo y observamos como el flujo sigue el camino adecuado desde el componente que lee los datos de origen hasta el componente que realiza commit y cierra la conexión en nuestra base de datos:



4. Referencias.

- <http://www.talendforge.org/components/>

5. Conclusiones.

Como podemos concluir la herramienta TALEND nos permite realizar de manera sencilla e intuitiva la extracción, manejo y volcado de información de una base de datos hacia otra base de datos.

La lista de componentes para tratar información de bases de datos Mysql en la herramienta es muy extensa:



Estos son sólo algunos de los componentes referentes a base de datos de los que nos ofrece talend. En futuros tutoriales iremos profundizando más en el uso de esta herramienta open source, así como en el resto de componentes que la misma ofrece (<http://www.talendforge.org/components/>).

Un saludo.

Daniel Casanova

dcasanova@autentia.com

A continuación puedes evaluarlo:

[Regístrate para evaluarlo](#)

Por favor, vota +1 o compártelo si te pareció interesante

[Share](#) |

0

Anímate y coméntanos lo que pienses sobre este TUTORIAL:

» **Registrate** y accede a esta y otras ventajas «



Esta obra está licenciada bajo [licencia Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5](#)

IMPULSA

Impulsores

Comunidad

[¿Ayuda?](#)

0 personas han traído clicks a esta página

sin clicks

+ + + + + + + +

powered by [karmacracy](#)

Copyright 2003-2012 © All Rights Reserved | [Texto legal y condiciones de uso](#) | [Banners](#) | [Powered by Autentia](#) | [Contacto](#)

[W3C](#) XHTML 1.0

[W3C](#) CSS

[XML](#) RSS

[XML](#) RDF