stratebi
open business intelligence

Integración Talend Data Preparation & Talend Stewardship con Talend Studio (Enterprise edition)

DEMO

CONTENIDO TEÓRICO

Índice

OBJETIVO

talend

OBJETIVO

El **objetivo** de esta demo es utilizar la herramienta **Talend Studio 7.3.1** con licencia **Enterprise**, para la creación de un job. A través de este job, se aplicarán algunas de las funcionalidades ofrecidas por *Talend Data Preparation* y *Talend Stewardship*, servicios de *Talend Cloud*.

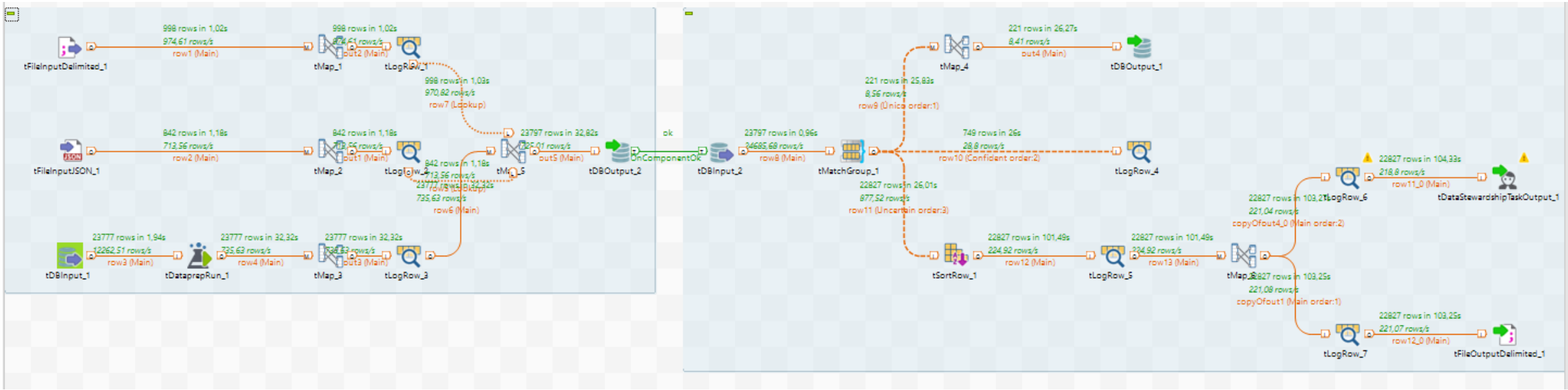
Se utilizará un dataset basado en las estadísticas obtenidas en las carreras de los principales premios de Fórmula1, desde 1950 hasta 2017. En estos datos se almacena información referente a la carrera, circuito, piloto, nacionalidad, clasificación, puntos obtenidos, mejores vueltas, etc.

ESTRUCTURA Y FUNCIONAMIENTO

talend

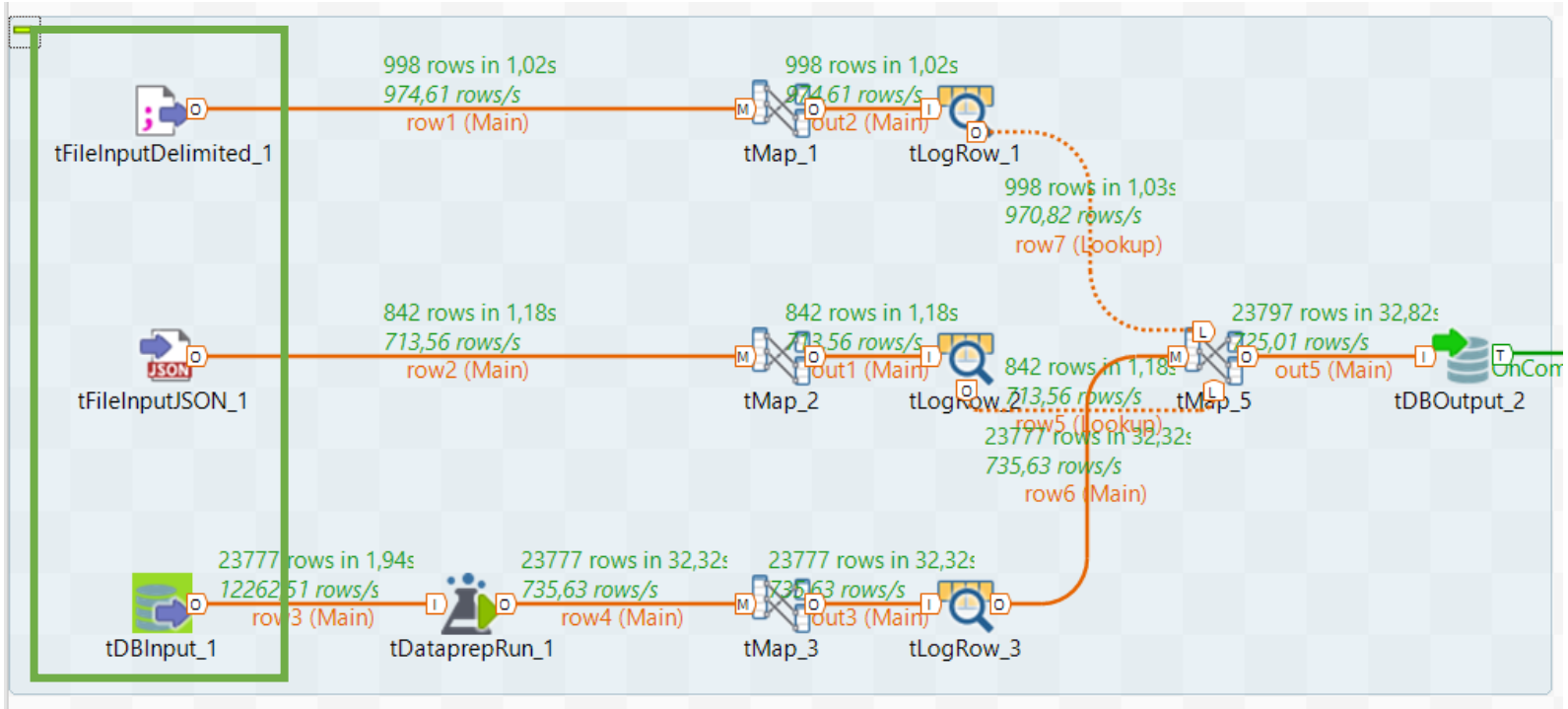
VISTA GENERAL

La estructura general del job es la siguiente:



FUENTES DE DATOS

El origen utilizado son tres fuentes de datos distintas (csv, json y una bbdd MySQL).



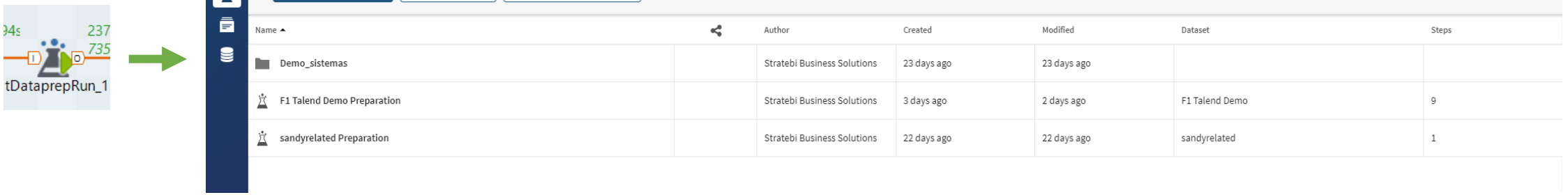
FUENTES DE DATOS

- Mediante el *csv* se importan los datos correspondientes a los grandes premios y los respectivos circuitos donde tienen lugar.
- El fichero *JSON* aporta los datos correspondientes a todos los pilotos profesionales de F1 durante los años mencionados.
- Por último, desde la base de datos de MySQL se importan los datos correspondientes a las carreras realizadas y sus resultados, es decir, posiciones, puntos obtenidos, vuelta rápida, etc. Este origen a su vez, hace uso de las funcionalidades de Talend Data Preparation, para realizar sobre los datos las transformaciones necesarias.

Con estas fuentes se crea un modelo de datos completo.

TALEND DATA PREPARATION

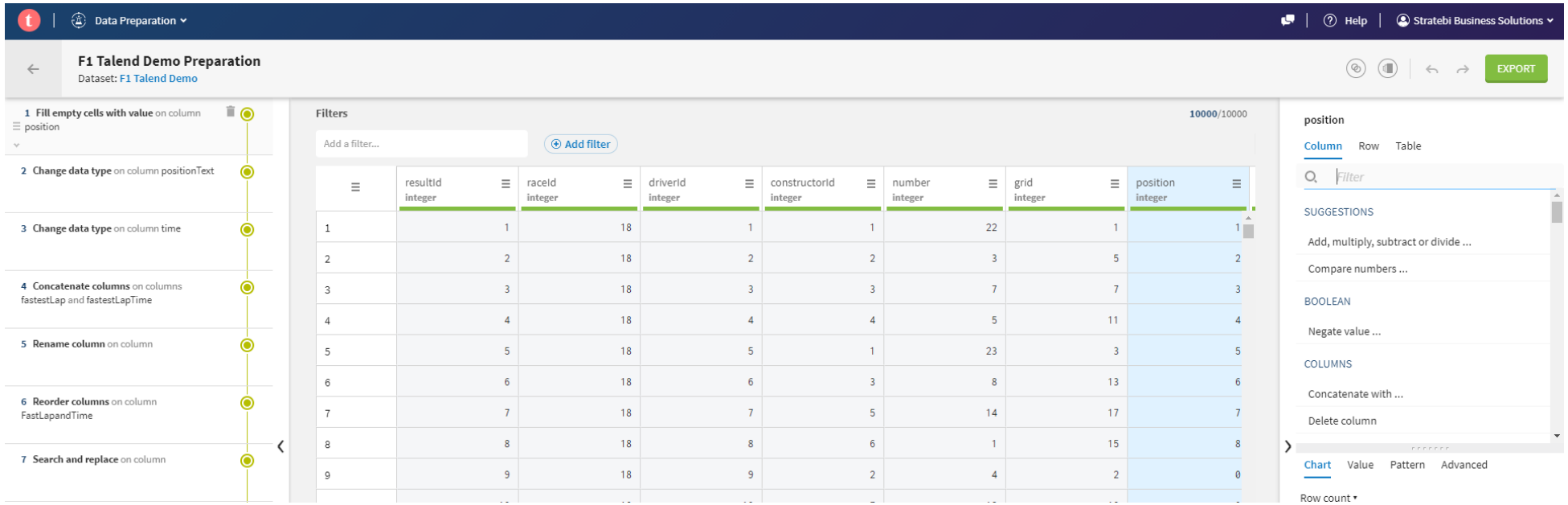
Con el componente **tDataPrepRun** se llama al servicio de **Talend Data Preparation** en la nube para ejecutar sobre el conjunto de datos entrante una *Preparación* en base al *Dataset* entrante. La preparación y dataset se crea previamente en el servicio y se la llama desde Talend Studio. Al "llamar" a esta preparación del servicio, se realizan sobre el conjunto de datos con el que se está trabajando en Talend Studio, una serie de transformaciones del dato como modificar el tipo de un campo, unificar dos columnas, reemplazar valores null, etc.



TALEND DATA PREPARATION

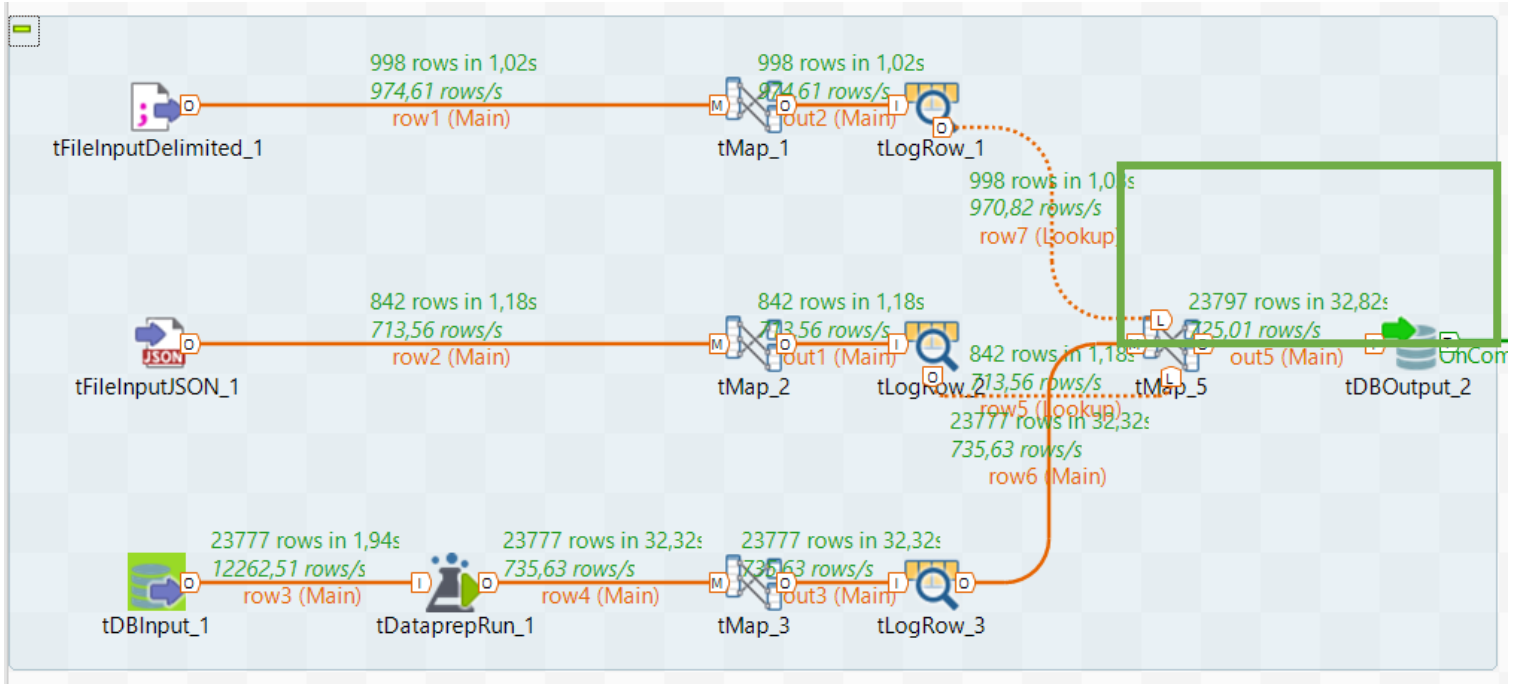
En esta demo los pasos realizados sobre el conjunto de los resultados de la carrera de Fórmula 1 son:

- Reemplazar los valores nulos de todos campos.
- Modificar la estructura de la tabla ordenando los campos.
- Modificar el tipo de dato asociado a los campos.
- Crear una columna fruto de la concatenación de otras dos. En este caso se crea una columna que indica en qué vuelta y el mejor tiempo que consigue el piloto en cada carrera.



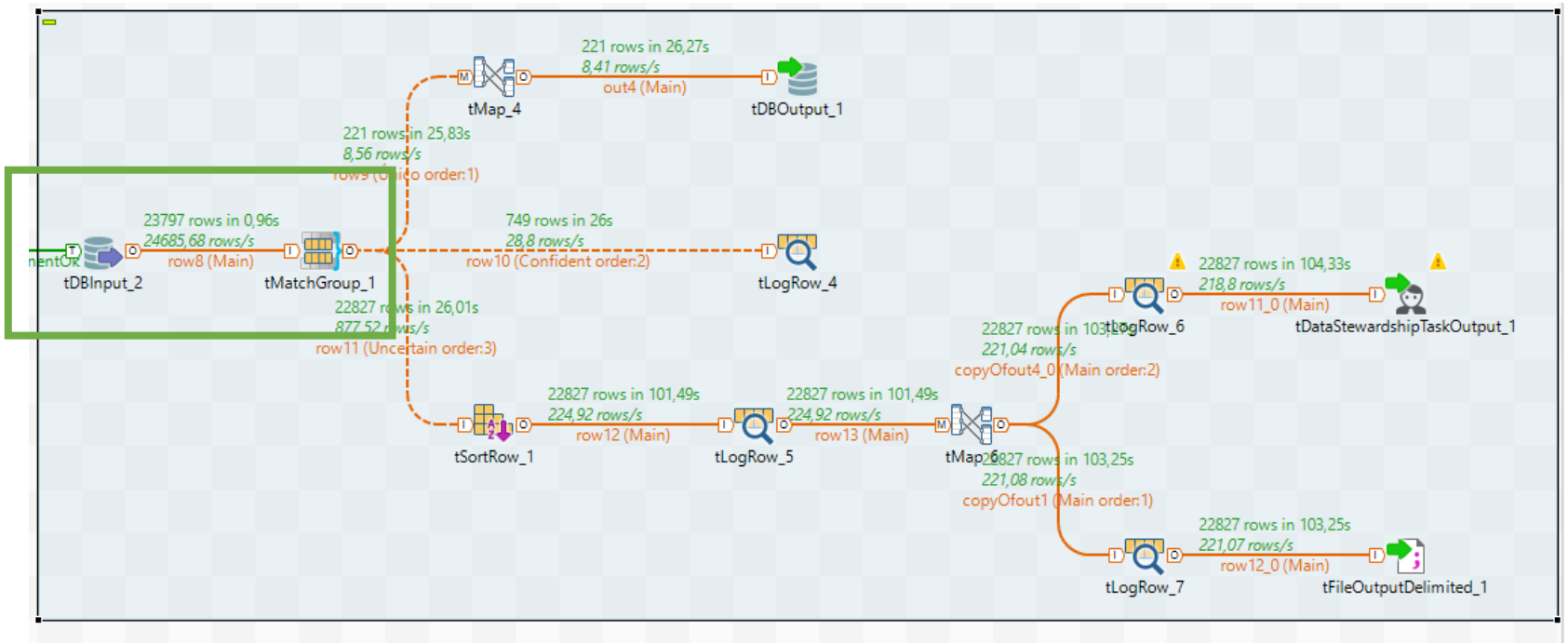
STG

El último paso del primer subjob es unificar todos los flujos en una única tabla para crear un área de Staging. En esta tabla se almacenarán todos los registros correspondientes a los resultados de cada gran premio de fórmula 1 con sus respectivos pilotos, circuitos, marcas personales, información personal, número de vueltas, etc.



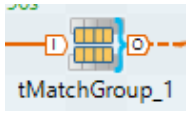
STG

Lo siguiente es extraer del área de Staging todos los registros, para poder aplicar una serie de algoritmos de reconocimiento de campos (tMatchGroup), que sirven para distinguir los datos únicos, de los correctos y de los duplicados o no identificados.



tMatchGroup

El componente tMatchGroup permite aplicar los siguientes algoritmos a los campos seleccionados, para la criba de datos. Esto se refleja en Talend Studio como la creación de grupos de registros de datos similares, mediante el uso de una o varias reglas de coincidencia.



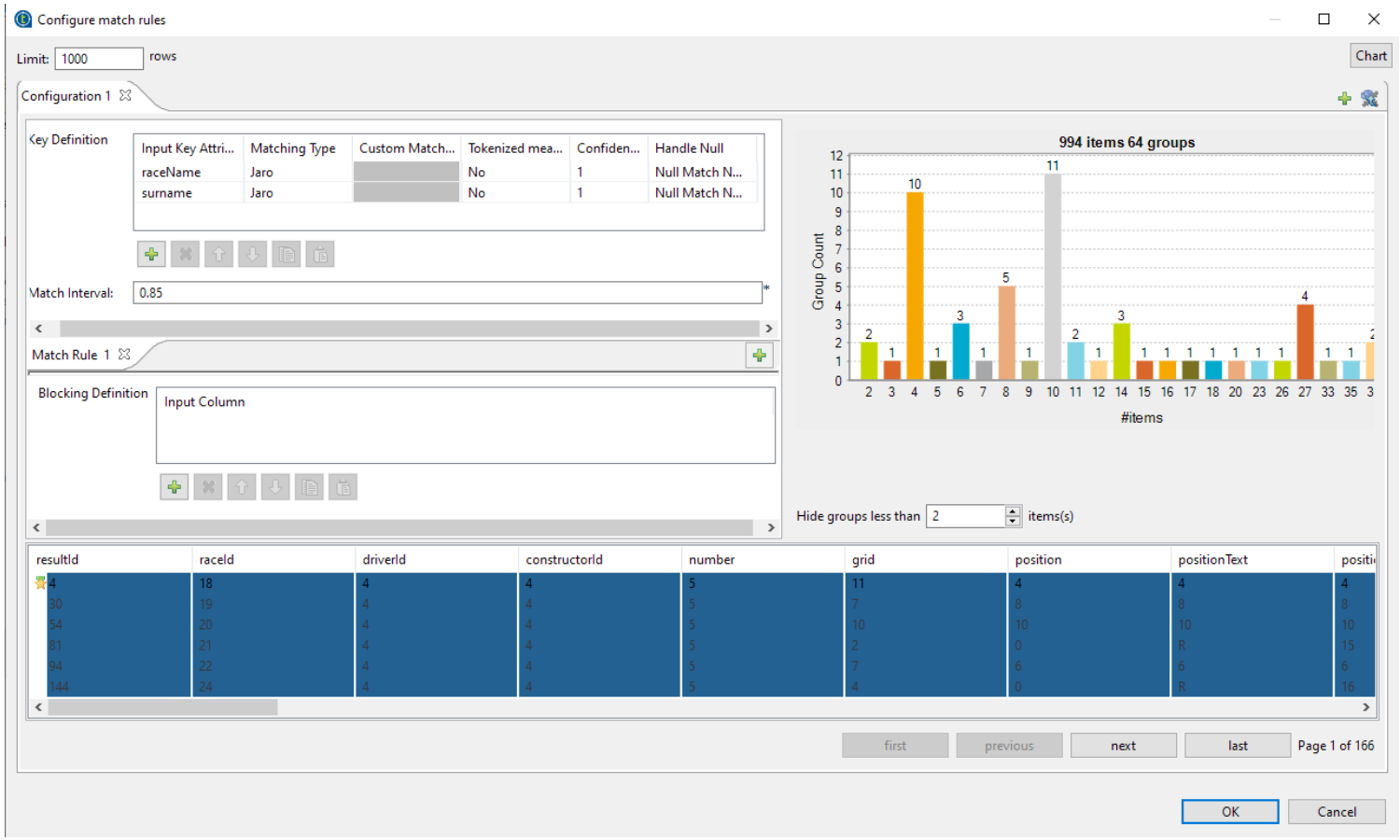
- **Exact**: hace coincidir cada entrada procesada con todas las entradas de referencia posibles con exactamente el mismo valor.
- **Exact (ignorar mayúsculas y minúsculas):** hace coincidir cada entrada procesada con todas las posibles entradas de referencia con exactamente el mismo valor, ignorando las mayúsculas y minúsculas del valor.
- **Soundex**: hace coincidir las entradas procesadas de acuerdo con un algoritmo fonético estándar en inglés. Indiza las cadenas por sonido, como se pronuncia en inglés, por ejemplo, "Hello": "H400".
- **Levenshtein** (distancia de edición): calcula el número mínimo de ediciones (inserción, eliminación o sustitución) necesarias para transformar una cadena en otra. El componente calcula automáticamente un porcentaje de coincidencia basado en la distancia. Esta puntuación de coincidencia se utilizará para el cálculo de coincidencia global, según la ponderación que asigne en el campo Ponderación de confianza .
- **Metaphone**: Basado en un algoritmo fonético para indexar entradas por su pronunciación. Primero carga la fonética de todas las entradas de la referencia de búsqueda y compara todas las entradas del flujo principal con las entradas del flujo de referencia.

tMatchGroup

- **Double Metaphone**: una nueva versión del algoritmo fonético Metaphone, que produce resultados más precisos que el algoritmo original. Puede devolver un código primario y secundario para una cadena. Esto explica algunos casos ambiguos, así como múltiples variantes de apellidos con ascendencia común.
- **Soundex FR:** hace coincidir las entradas procesadas de acuerdo con un algoritmo fonético estándar francés.
- **Jaro**: empareja las entradas procesadas según las desviaciones ortográficas. Cuenta el número de caracteres coincidentes entre dos cadenas. Cuanto mayor sea la distancia, más similares serán las cuerdas.
- **Jaro-Winkle**r: una variante de Jaro, pero le da más importancia al comienzo de la cuerda.
- **Q-gramos:** hace coincidir las entradas procesadas dividiendo las cadenas en bloques de letras de longitud qpara crear una cantidad de q gramos de longitud. El resultado de la coincidencia se da como el número de coincidencias de q-gramos sobre los posibles q-gramos.
- **Hamming**: calcula el número mínimo de sustituciones necesarias para transformar una cuerda en otra cuerda de la misma longitud. Por ejemplo, la distancia de Hamming entre " m a sking " y " p a iring " es 3.
- **Custom**: le permite cargar un algoritmo de coincidencia externo desde una biblioteca Java utilizando la columna Matcher personalizada .

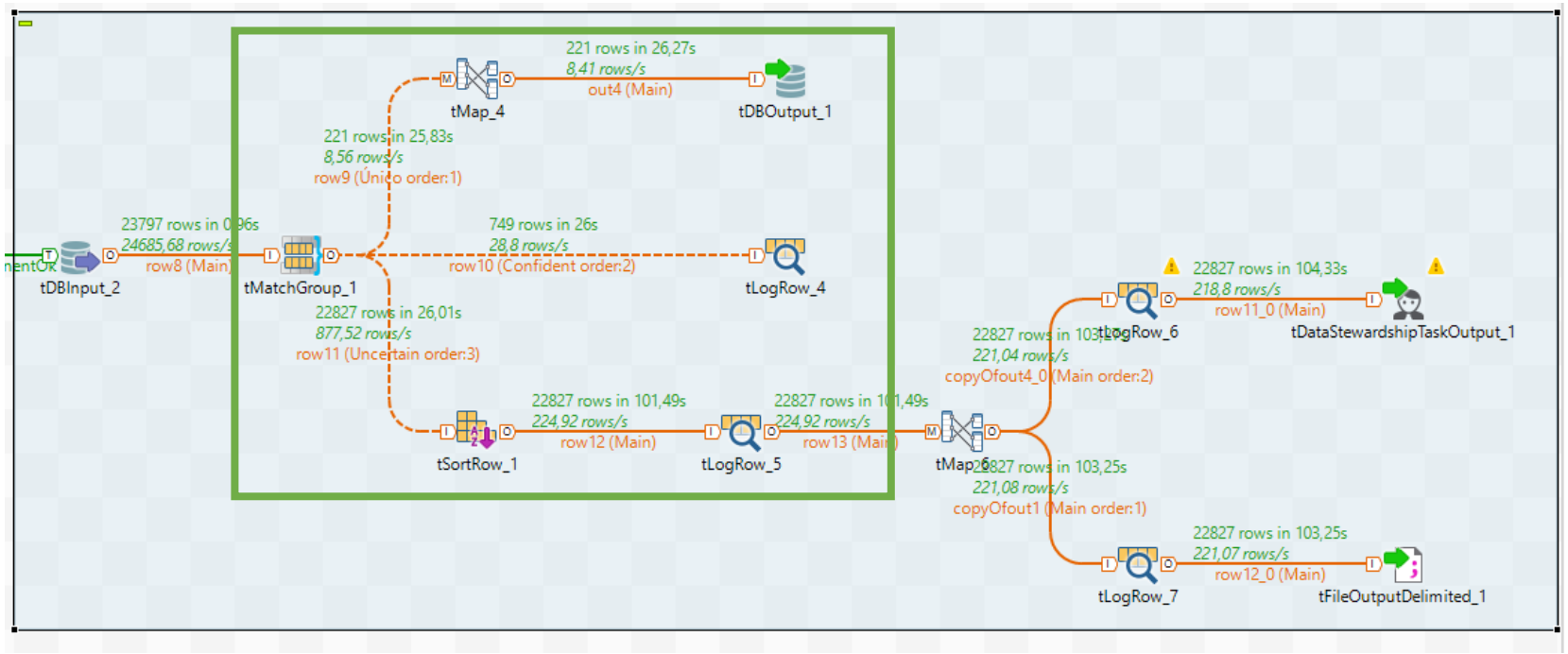
tMatchGroup

En este caso se configura el componente tMatchGroup como sigue:



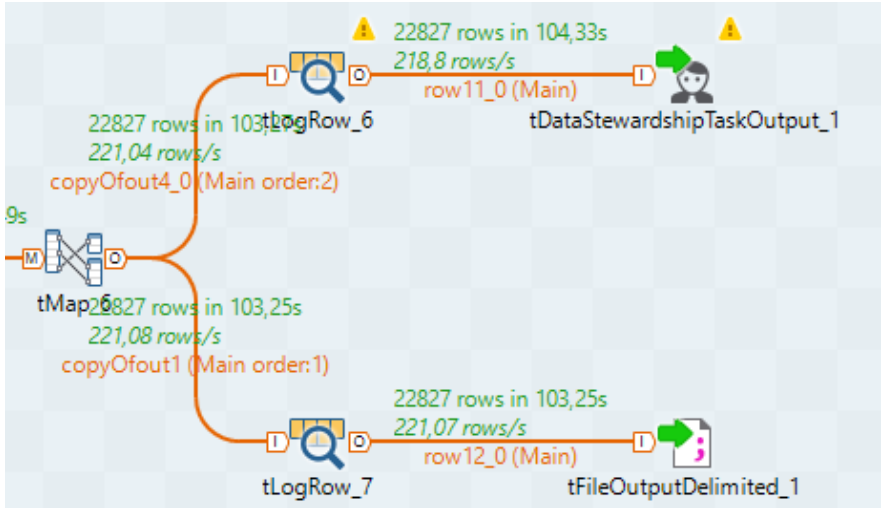
SALIDAS

En este caso se obtienen tres salidas: valores únicos, valores correctos y valores duplicados o no aceptados.



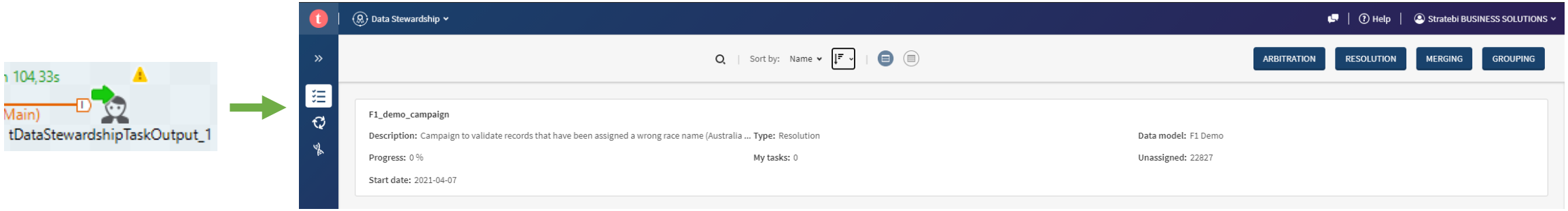
SALIDAS

- En el caso de los valores únicos se almacenan en una nueva tabla de la bbdd utilizada.
- Los valores correctos simplemente se muestran por pantalla con el componente tLogRow.
- Por último, los valores no admitidos o duplicados se duplican en dos salidas. Una de ellas escribe estos valores no admitidos, en un fichero csv para dejar constancia y registrados los valores incorrectos o duplicados, con el fin de poder acceder a ellos en otro momento. La otra salida de los datos asignados como inciertos lanza al servicio en la nube Talend Stewardship el conjunto de datos para ser analizados mediante una campaña.



TALEND DATA STEWARDSHIP

Con el componente **tDataStewardshipTaskOutput** se llama al servicio de **Talend Stewardship** en la nube para ejecutar sobre el conjunto de datos entrante una *Campaña* en base al *Data Model* entrante. La campaña se crea previamente en el servicio y se la llama desde Talend Studio. Al "llamar" a esta campaña del servicio, se realizan sobre el conjunto de datos con el que se está trabajando en Talend Studio, una serie de tareas establecidas anteriormente. Esto sirve para que un usuario de negocio pueda validar la calidad de los datos y seleccionar qué hacer con cada registro en función de la campaña asignada.



TALEND DATA STEWARDSHIP

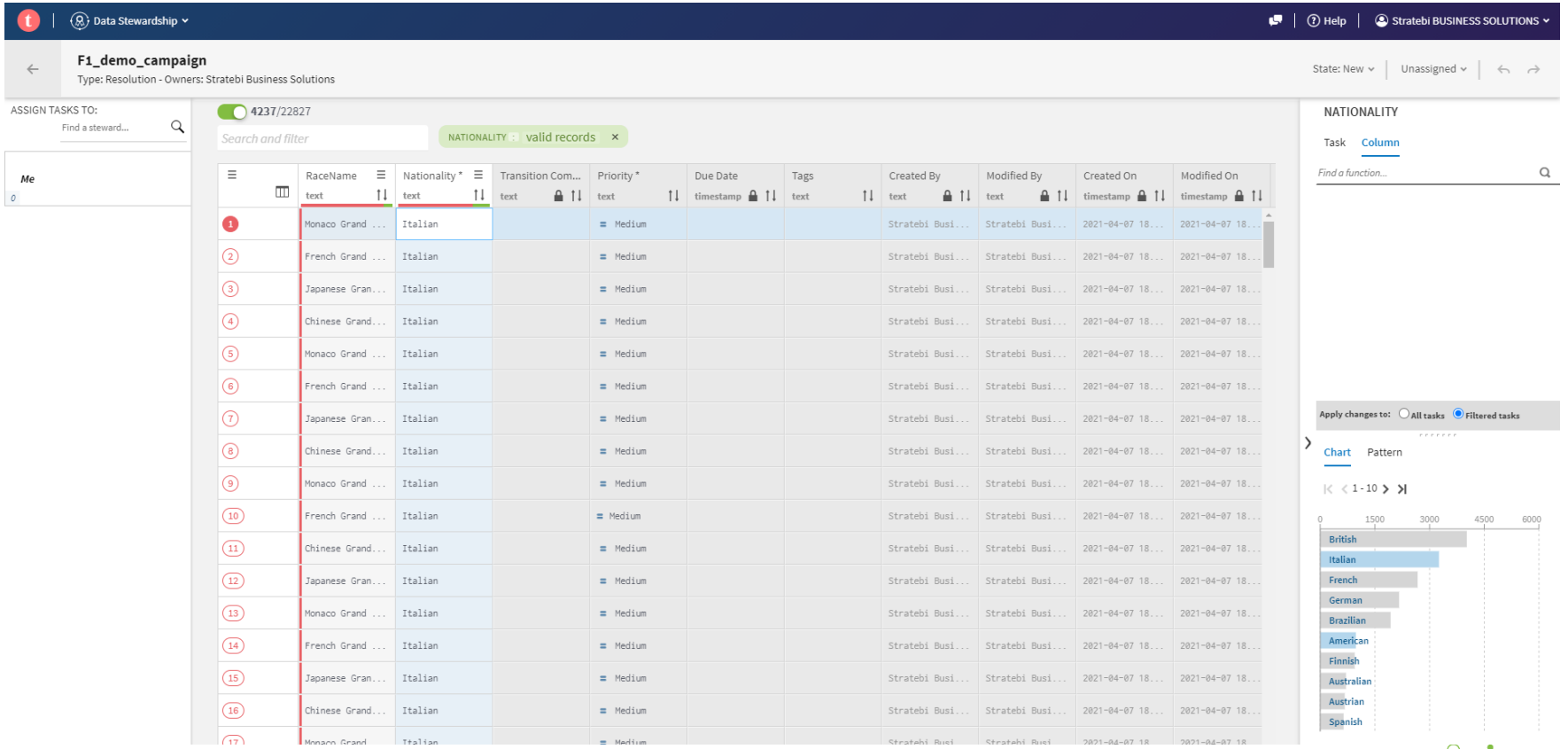
Para crear una campaña es necesario definir la estructura del modelo de datos a utilizar, para así poder tomar decisiones sobre los datos que proceden de este modelo. Es decir, esta herramienta permite al usuario de negocio que conoce el valor de los datos reales, poder decidir si quiere fusionar un registro, aceptarlo, rechazarlo, o modificarlo en función del valor que quiera establecer en cada caso. Se debe recordar que estos registros han sido previamente declarados como no admitidos o duplicados para el modelo final.

El conjunto de tareas que sirve para decidir que hacer con cada campo o registro, forman las campañas. A su vez, cada tarea puede asignarse a un steward distinto.

Los tipos de campaña se definen a continuación:

TALEND DATA STEWARDSHIP

Para esta demo se quieren revisar aquellos registros pertenecientes a pilotos Italianos y Americanos, así como aquellos nombres de las carreras que exceden el tamaño recomendado. Se crea entonces una campaña de Resolución para ambos campos.



FIN

Por último, el usuario de negocio final, decidirá que registros o campos acepta como parte del conjunto final de datos y cuales rechaza. De esta forma, junto al conjunto de datos únicos almacenados previamente, se crearía el conjunto de datos definitivo con su respectivo modelo de datos.