Talend Component tGoogleSheetOutput

Purpose and procedure

This component writes Google Sheets.
The component provides these capabilities:

1. Insert/Update or Append rows
2. Can delete rows after last written row to cleanup documents
3. Compressed data load saves bandwith

Talend-Integration

This component can be found in the palette under Cloud->Google
This component provides an input flow and several return values (depending on the operational mode).

Parameters and Usage

Parameters to establish the client and connect to Google Drive (the home of your sheets)

These settings will be used in all modes and therefore explained only once here.
Google suggest 2 different modes for authentication for backend processes and native applications:

1. Service Account: A service account is a new account and has the advantage it does not need any user interaction while the job runtime. If the job is supposed to manage real person drives, this mode does not help because the files have as owner the service account and you cannot access them directly.
2. Client-ID for native applications: A Client-Id is needed if you want managing files on real person accounts. The disadvantage is, it depends on a user interaction (only for the first time any arbitrary job using this account runs the first time, all other jobs using he same account does not need any interaction anymore).

Property	Content	Data types
Application Name	Not necessary, but recommended by Google. Simple provide the name of your application gathering data. ***Required***	String
Use existing client	Choose here the tGoogleDrive component which client do you want to reuse in this component instance.	Boolean
Authentication Method	Choose the method to authenticate: Service Account or OAuth Client-ID (for native applications)	

Properties to use the service account

Property	Content	Data types
Service Account Email	The email address of the service account. Google creates this address within the process of creating a service account. Only for service accounts! ***Required***	String
Key File (p12)	The Service Account Login works with private key file for authentication. In the process of creating a service account you download this file. Only for service accounts ***Required***	String

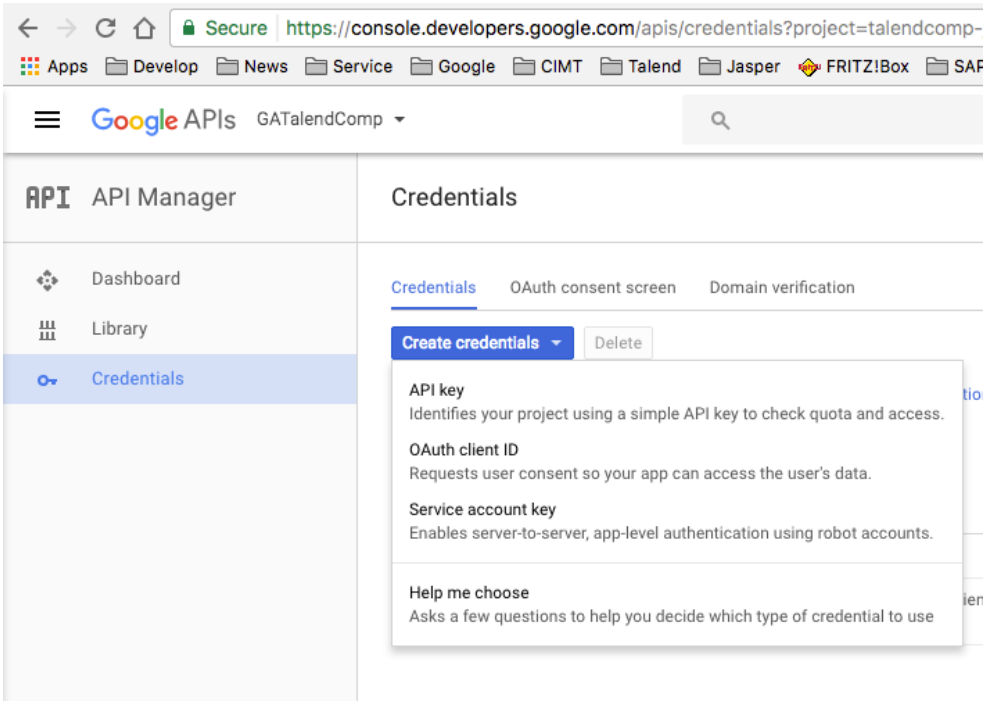
Properties to use the OAuth Client-ID authentication

Property	Content	Data types
User Account Email	Email of the user account or the Client-ID	String
Client secret file (json)	This json file downloaded for the Client-ID	String

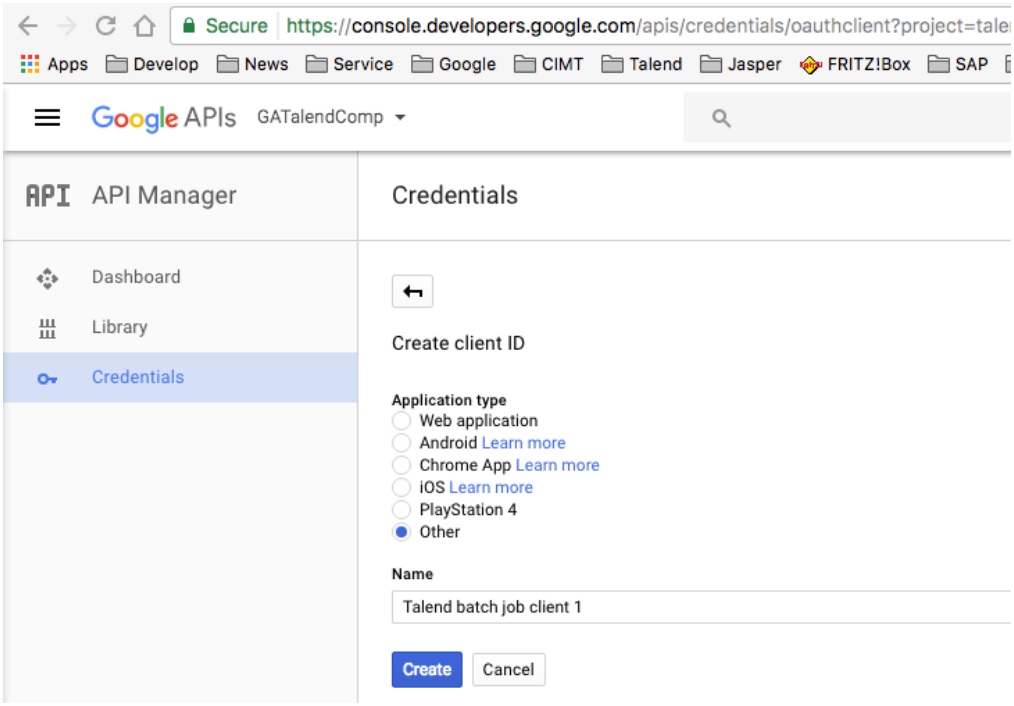
The usage of the “OAuth Client-ID” expects on the first run a user interaction with the Google web page and after finishing the form to approve the access right you need to close the browser to let the component continue, otherwise the authentication process will not complete.

How to enable and authenticate the drive client

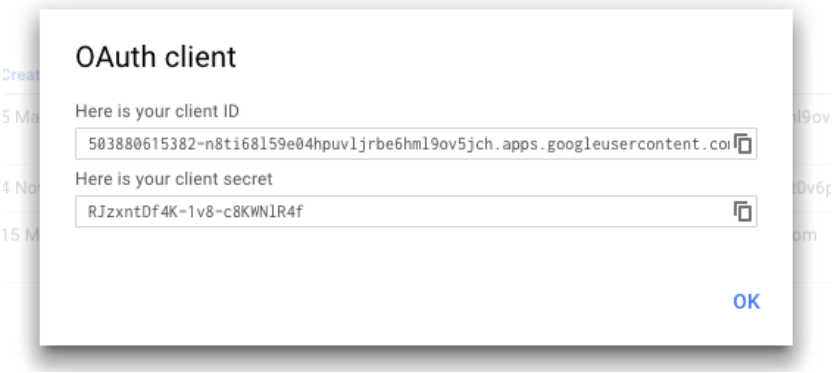
1. Create credentials to allow you job to get access to your files
Go to the Google API console: https://console.developers.google.com/apis/dashboard
And enable here the Google Drive API and the Google Sheet API.



… now choose the application type:



Ok, here the essentials of the new credentials. But you do not need to memorize that here.



Download the credentials as json file (click on the download icon)

You will get a json file. This file is what you need in the component.
If you start yur job first time the component opens a browser and you must approve the access of the application (the Client-ID) to your data. After you have approved the access and have closed the browser(!!), the component creates a directory with the name of the json file (without the extension).
If you want to relocate your job, simple take care the created directory is next to the json file and the component points to this json file.

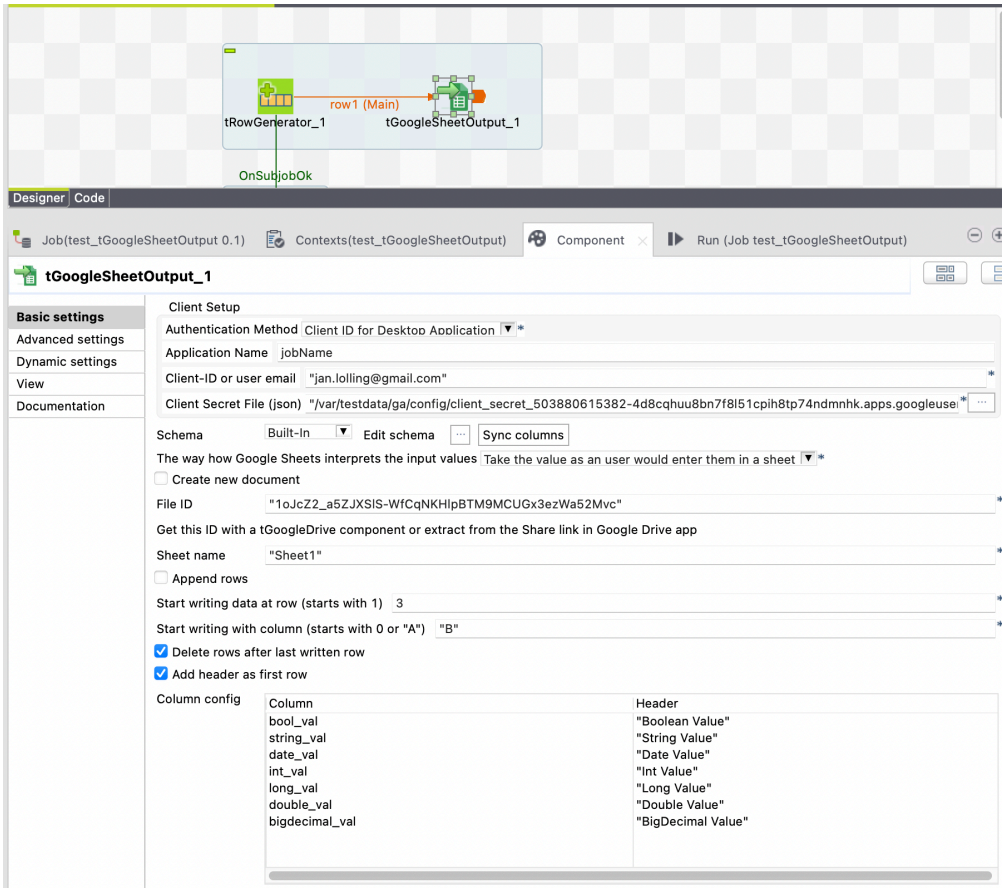
Basic settings
(Settings to work with Sheets)

Property	Content
Schema	The schema of the component data flow
File-ID	The ID of the sheet. A sheet is simply a file in your Google Drive and has an ID (alfa-numeric String)
Create new document	A new spread sheet document will be created
Title	If a new document should be created, this it the title from this new document. Please take note, a title is unlike file names not unique. You can have multiple files with the same title.
Sheet Name	The sheet to be created if not exists and to write into
Append Rows	Append Rows to the last row of the sheet. Please take care the sheet does not contains empty rows at the end of the sheet because these will remain. You can clean up the sheet in the web app or later with the option “Delete rows after last written row”
Start writing data at row	This is the row where the component starts writing. Also, a header will be placed here if a header is wanted. This option disappears if “Append Rows” is chosen.
Start writing data at column	The start column where the component starts.
Delete rows after last written row	If the sheet contains more rows you actually have written, this option delete all rows after the last written row.
Add Header as first written row	A Header will be written as first row.
Column Configuration	**Column**: the schema column **Header** Name of the column in the header row
Die on error	In case the component detects an error, the job will fail.

Return Variables

Property	Content
NB_LINE	Number lines written
LAST_ROW_INDEX	The index of the last row changed
FILE_ID	The Id of the file created of given per input control

Basic settings in the example job:



This example writes into an existing file using the ID of the file.
You can get this ID with the help of the Google Drive Webapp or with tGoogleDrive and here the List function with a filter of the title. Please take note a title is unlike in a real file system not unique in Google Drive.