 **Talend Component tGoogleSheetInput**

**Purpose and procedure**

This component reads Google Sheets.

The component provides these capabilities:

1. Read the cells of a sheet
2. Detect the position of columns by a header row
3. Tolerant date parsing
4. 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 |

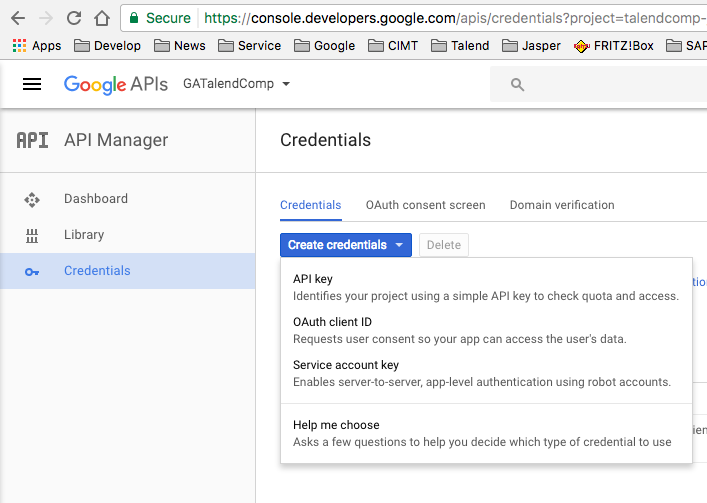
The usage of the “OAuth Client-ID” expects on the first run a user interaction with the Google web page and after finishing the form to approve the access right you need to close the browser to let the component continue, otherwise the authentication process will not complete.

**How to enable and authenticate the drive client**

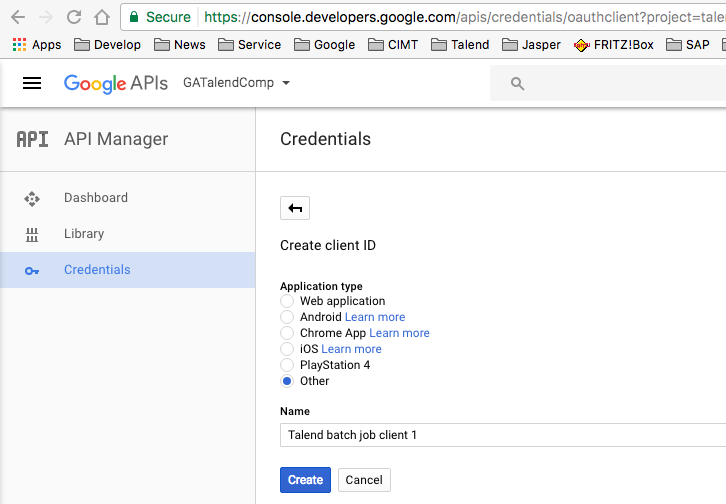
1. Create credentials to allow you job to get access to your files

Go to the Google API console: <https://console.developers.google.com/apis/dashboard>

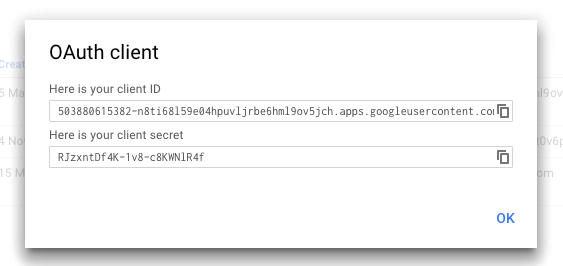
And enable here the Google Drive API and the Google Sheet API.



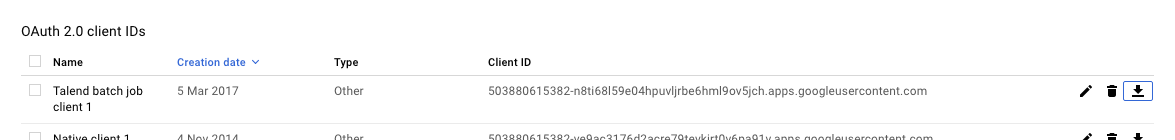
… now choose the application type:



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).

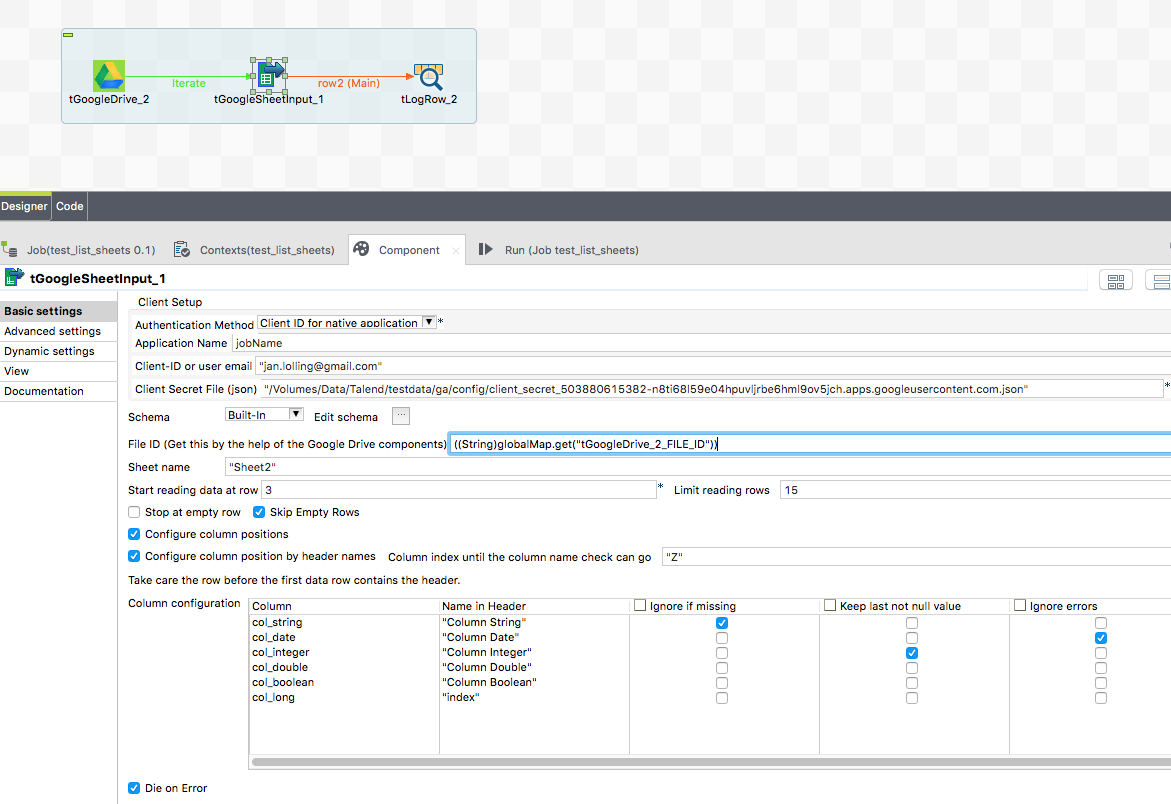
If you want to relocate your job, simple take care the created directory is next to the json file and the component points to this json file.

**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) |
| Sheet name | The name of the sheet you want to read. |
| Start reading data at row | The first row of the data (without the header line). In case we expect a header line, the component takes the row above) |
| Limit reading rows | You can specify a limit of data rows to read. |
| Stop at empty row | If the component detects an empty row (it takes only the columns in account you are about to read) it stops the reading |
| Skip empty rows | Skip over empty rows. The return value CURRENT\_ROW\_INDEX is always correct and shows the real row index. |
| Configure column positions | This option enables the individual column positioning. This is the default option. |
| Configure column position be header names | The component tries to find the header names in the header row and takes their position as new position. If you want to see the result of this discovery, switch on the debug mode. |
| Column Configuration | **Column**: the schema column  **Name in header** (alternative to Index or Name): Name of the column in the header row  **Ignore if missing:** Ignore missing columns (the component nulls the values of missing columns) |
|  | **Index of Name**: Name ("A" or "Z") or index (starts with 0) of the column in the sheet  Keep last not null value: The component uses the last not-null value for null values of the next row  **Ignore errors:** In case of transformation errors (like exceptionally a text instead of a date or numeric value) this value will be ignored and set to null |
| Die on error | In case the component detects an error, the job will fail. |

Basic settings in the example job:



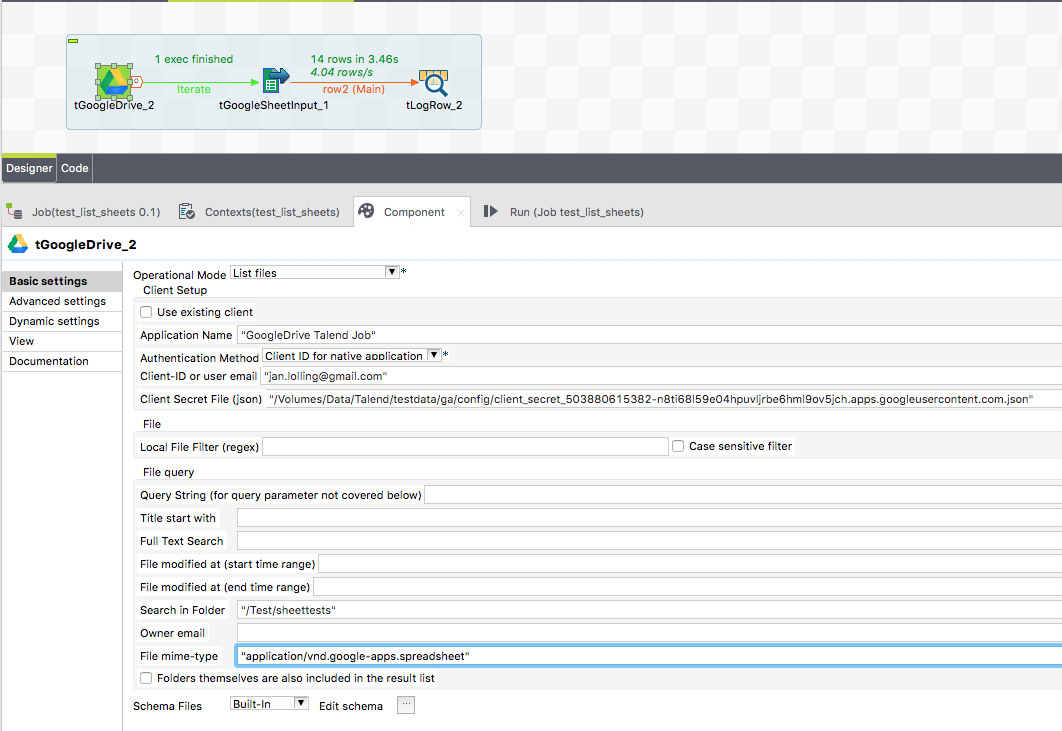
To get the id of the Sheet the best way is to use the tGoogleDrive component.

To get only Sheets you must filter the list of files in your Google Drive by the mime-type:

"application/vnd.google-apps.spreadsheet"

Here the settings of the tGoogleDrive component: (next page)

In the tGoogleDrive component you can use exactly the same authentication settings as in tGoogleSheetInput.



## Advanced Settings

The advanced settings only apply to clients. If a component uses an existing client they do not apply.

|  |  |
| --- | --- |
| **Property** | **Content** |
| Timeout in s | How long should the component wait for getting the first result and fetching all result with one internal iteration |
| Static Time Offset (to past) | Within the process of login, the component requests an access token and use the local time stamp (because these tokens will expire after a couple of seconds)  Google rejects all requests to access tokens when the request is in the future compared to the timestamp in Google servers. If you experience such kind of problems, this options let the requests appear to be more in the past (5-10s was recognized as good time shift) |
| Reuse Client for Iterations | If you use this component in iterations it is strongly recommended to set this option. It saves time to authenticate unnecessary often and avoids problems because of max amount of connects per time range. |
| Distinct Name Extension | The client will be kept with an automatically created name:  Talend-Name-Component name + job name. In case this is not distinct enough, you can specify an additional extension to the name. |