tFileExcelSheetInput

Purpose

This component reads Excel sheets.
This component needs the components tFileExcelWorkbookOpen (open a file or creates a workbook)
Advantages of this component:

- The columns to read can be set also with gaps
- The column position can automatically configured by a header line and additional by the use of regularly expressions
- Can read reliable all possible data types and tries also to convert it into the schema target column type
- Can read comments
- Can fill the output stream for empty cells with the last not empty value
- Can ignore cell read errors e.g. in case of a type conversion is impossible
- Uses always the latest Apache POI API

Talend-Integration

This component can be found in the palette under File/Spreadsheet
This component provides several return values.

Parameters for tFileExcelSheetInput

Property	Content
Workbook	Choose the tFileExcelWorkbookOpen component holding the Apache POI Excel workbook
Schema	The schema of the output
Sheet name	The name or the index of the sheet. Please take care of a valid sheet name or simply type the index of the sheet your want to write. You can take the sheet name from the return value of the tFileExcelSheetList.
Find Sheet Tolerant	Find the sheet by ignoring case, spaces and underscores, otherwise the name must fit exactly to an existing sheet.
Row start index	The component starts reading data with this row (1-based)
Stop at empty row	If no values from the row was received (only all configured columns) the component stops reading.
Skip empty rows	If the all needed values are empty, the row will be skipped.
Column start	Is visible only if no individual column configuration takes place The first column A in an excel sheet can be referenced with "A" or 0.
Use individual column configuration	If chosen it shows the configuration of columns and its position can be configured individual
User header to configure position of columns	If true the column position will be configured according to its position in the header line. The name will be found not case sensitive.
Header line	In the column configuration it is possible to configure the column position by the header line. Here set the index of the header line (1-based)
Column configuration	You can specify the columns in the Column Configuration in the column Sheet Column Name. Here you can use the Excel letter reference (“A” for the first column) or an index (0 for the first column). It is possible to have gaps between the different columns (unlike the build-in Talend components). **Column**: Name of the schema column **Sheet Column Name**: Column position as letter (starts with “A” or 0)

	Name in Header: if position should be found in header, set here the name in the header (also with the use of regex) **Read cell comment:** If checked the comment will taken as value **Use last value for empty:** If checked an empty cell will be filled with the last known value **Ignore Errors:** If something goes wrong this option avoids aborts

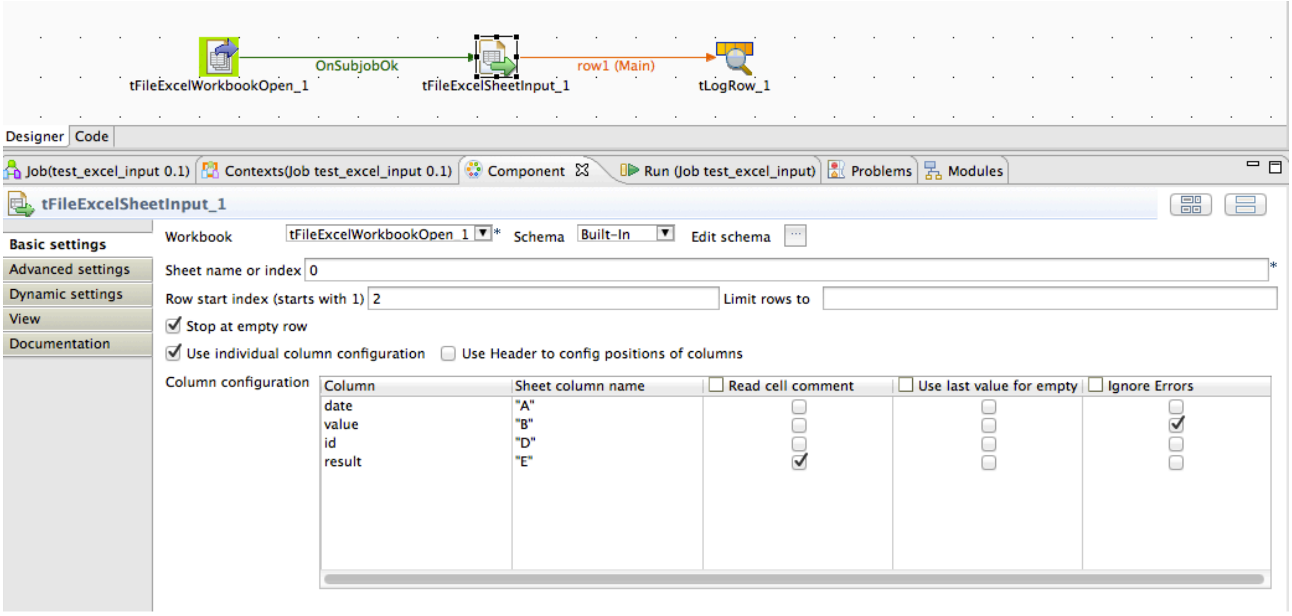
Advanced setting parameters for tFileExcelSheetInput

Property	**Content**
Language / Country for number format	Number formats are different for different languages/countries. In case of the number is stored in a text typed cell and the schema expects a number this local will be used to find the correct format pattern for the text-to-number conversion.
Return Hyperlink URL	The normal value for a cell is always the visible cell value. If the cell has an underlying hyperlink, this option must be switched on to get the hyperlink instead of the value.
Concatenate Label | URL	The label of the hyperlink is the visible cell value and if both (label and hyperlink) are needed the component read both values and concatenates them with a pipe symbol.
Trim columns	If checked all textual content will be trimmed (leading and trailing spaces, tabulators or line breaks will be removed)
Use cached value if formula evaluation fails	It could happen especially if a formula references to external files, the formula fails. Excel usually keeps the value of the last evaluation and this value will be delivered.

Return values of the component:

Value	Content
NB_LINE	Number of lines read
ERROR_MESSAGE	Error message if something went wrong
LAST_ROW_INDEX	Index of the last read row in this sheet.
MAX_ROW_INDEX	The maximum row index available in the sheet (not necessarily read)
CURRENT_ROW_INDEX	The current absolute excel row index (available within the flow)

Scenario 1: Read from cells referenced by the Excel column names



Workbook tFileExcelWorkbookOpen_1 * Schema Built-In Edit schema

Sheet name or index 0 *

Row start index (starts with 1) 2 Limit rows to

☑ Stop at empty row

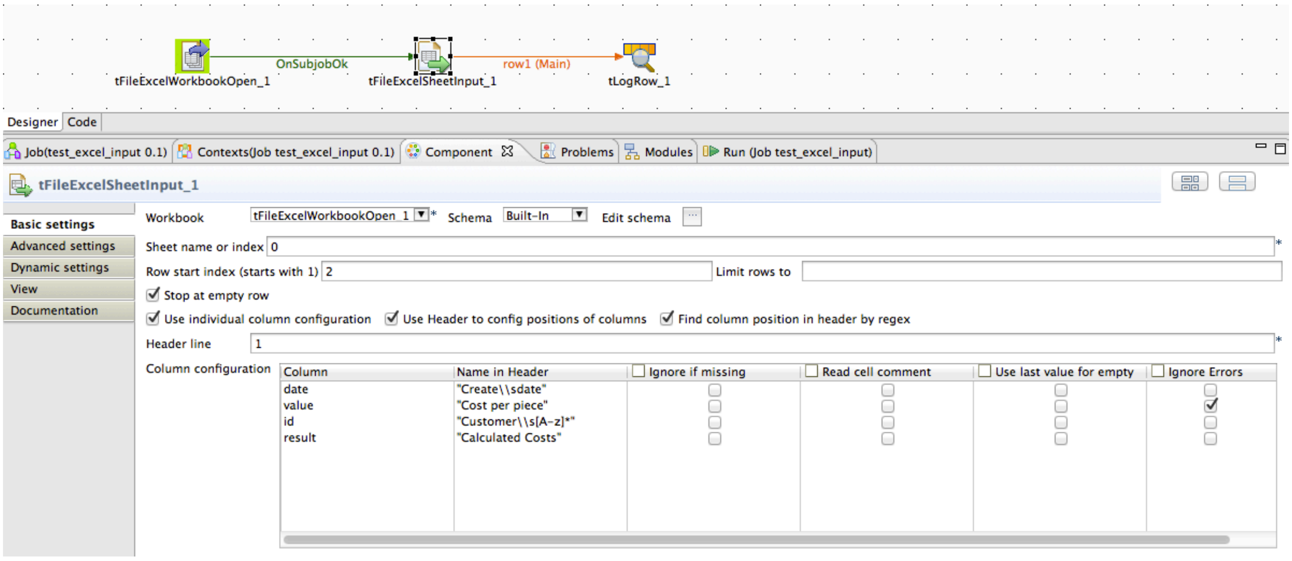
☑ Use individual column configuration ☐ Use Header to config positions of columns

Column configuration

Column	Sheet column name	☐ Read cell comment	☐ Use last value for empty	☐ Ignore Errors
date	"A"	☐	☐	☐
value	"B"	☐	☐	☑
id	"D"	☐	☐	☐
result	"E"	☑	☐	☐

The cell can be addressed with the well-known Excel column name (staring with “A”) or the cell index (starting with 0).

Scenario 2: Read from cells referenced by the header line



Workbook tFileExcelWorkbookOpen_1 * Schema Built-In Edit schema

Sheet name or index 0 *

Row start index (starts with 1) 2 Limit rows to

☑ Stop at empty row

☑ Use individual column configuration ☑ Use Header to config positions of columns ☑ Find column position in header by regex

Header line 1 *

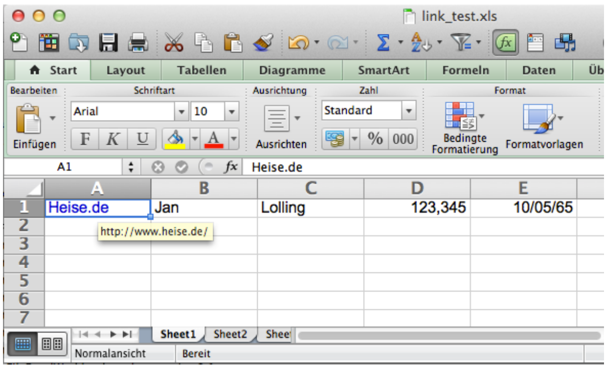
Column configuration

Column	Name in Header	□ Ignore if missing	□ Read cell comment	□ Use last value for empty	□ Ignore Errors
date	"Create\\sdate"	□	□	□	□
value	"Cost per piece"	□	□	□	☑
id	"Customer\\s[A-z]*"	□	□	□	□
result	"Calculated Costs"	□	□	□	□

In this scenario the column the header line will configure positions. The component tries to find the column by its name (case insensitive) or by regularly expressions (also case insensitive).

Scenario 3: Read hyperlinks from a cell

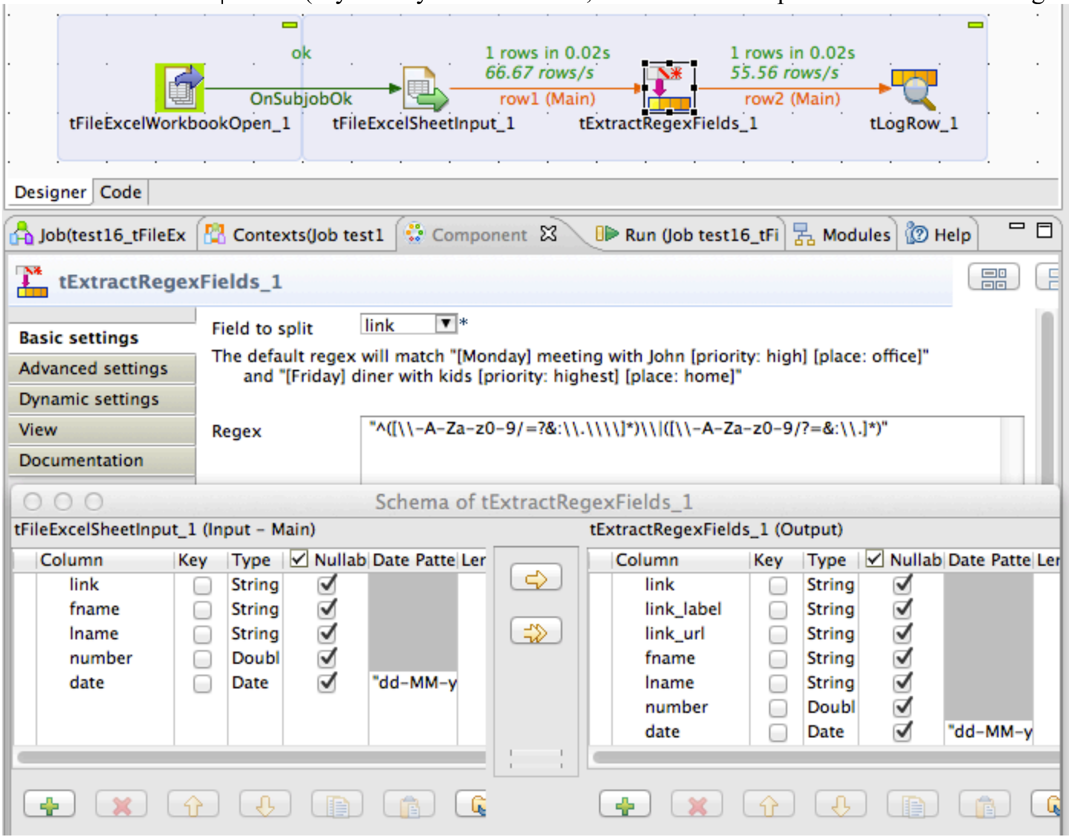
The file to read looks like this:



link_test.xls

	A	B	C	D	E
1	Heise.de	Jan	Lolling	123,345	10/05/65

Here a simple job reading the hyperlink and separate label and URL with a regex expression.
In the component tFileExcelSheetInput activate in the advanced settings the options “Return Hyperlink URL” and “Concatenate Label | URL” (if you only need the URL, disable the last option and leaf out the regex component).



Field to split: link *

The default regex will match "[Monday] meeting with John [priority: high] [place: office]"
and "[Friday] diner with kids [priority: highest] [place: home]"

Regex: "^([\\-A-Za-z0-9/=?&:\\.\\\\]*)\\|([\\-A-Za-z0-9/?=&:\\.]*)"

Schema of tExtractRegexFields_1

tFileExcelSheetInput_1 (Input – Main)

Column	Key	Type	Nullab	Date Patte	Ler
link	□	String	☑		
fname	□	String	☑		
lname	□	String	☑		
number	□	Doubl	☑		
date	□	Date	☑	"dd-MM-y	

tExtractRegexFields_1 (Output)

Column	Key	Type	Nullab	Date Patte	Ler
link	□	String	☑		
link_label	□	String	☑		
link_url	□	String	☑		
fname	□	String	☑		
lname	□	String	☑		
number	□	Doubl	☑		
date	□	Date	☑	"dd-MM-y	

The tExtractRegexFields expects after the parsed field (link) additional fields as much as you want to extract content by regex groups. It is highly recommended to check the regex expression with external tools and take care you get only one regex sequence with (in this case) to groups. Please keep in mind every regex sequence causes an output record (e.g. an additional output record).
The regex expression here is: `"^([\\-A-Za-z0-9/=?&:\\.\\\\]*)\\|([\\-A-Za-z0-9/?=&:\\.]*)"`

The output of the job:

```
.-----------------------------+----------+--------------------+-----+-------+-------+----------.
|                                          tLogRow_1                                           |
|=-----------------------------+----------+--------------------+-----+-------+-------+---------=|
|link                         |link_label|link_url            |fname|lname  |number |date      |
|=-----------------------------+----------+--------------------+-----+-------+-------+---------=|
|Heise.de|http://www.heise.de/|Heise.de  |http://www.heise.de/|Jan  |Lolling|123.345|10-05-1965|
'-----------------------------+----------+--------------------+-----+-------+-------+----------'
```

tFileExcelSheetInputUnpivot

Purpose

This component normalized pivotal structured data.
This component needs the components tFileExcelSheetInput which reads the actual row by row and tFileExcelSheetInputUnpivot normalizes these row data.

Talend-Integration

This component can be found in the palette under File/Spreadsheet
This component multiplies potentially the incoming records

Parameters for tFileExcelSheetInputUnpivot

Property	**Content**
tFileExcelSheetInput component	Choose the tFileExcelSheetInput which is the actual input for this component
Schema	The schema of the output
Row index for the header	Config here to the header row
Start column index of the pivot data	Config here the start column at which the pivot data starts (it is 0-based or starts with “A”)
End column index of the pivot data	Config here the last column for the pivot data. If you leave it free the column range is defined by the existing filled cells of the header row
Column for the normalized header	Choose here the schema column in which you want to have the header values
Column for the normalized value	Choose here the schema column in which you want to have the actual value of the pivot data
Ignore errors while getting values	If true the component ignores errors while extracting the values and set the affected values to null
Original row index of the value	Choose here an integer typed schema column in which the component writes the original excel row index from which the current values if taken
Original column index of the value	Choose here an integer typed schema column in which the component writes the original excel column (0-based) index from which the current values if taken

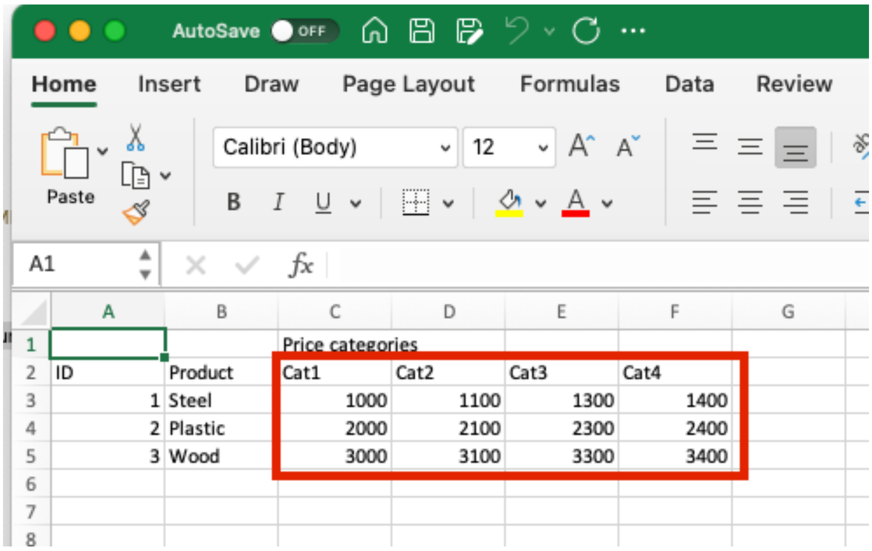
The component replicates the incoming row as much as it can provide normalized values.
If there is no normalized value, the component returns zero rows.

The component transfers the incoming values to the outgoing values for all schema columns which exist in the incoming flow as well in the outgoing flow.

This way the tFileExcelSheetInput component provides the values for the row header(s)

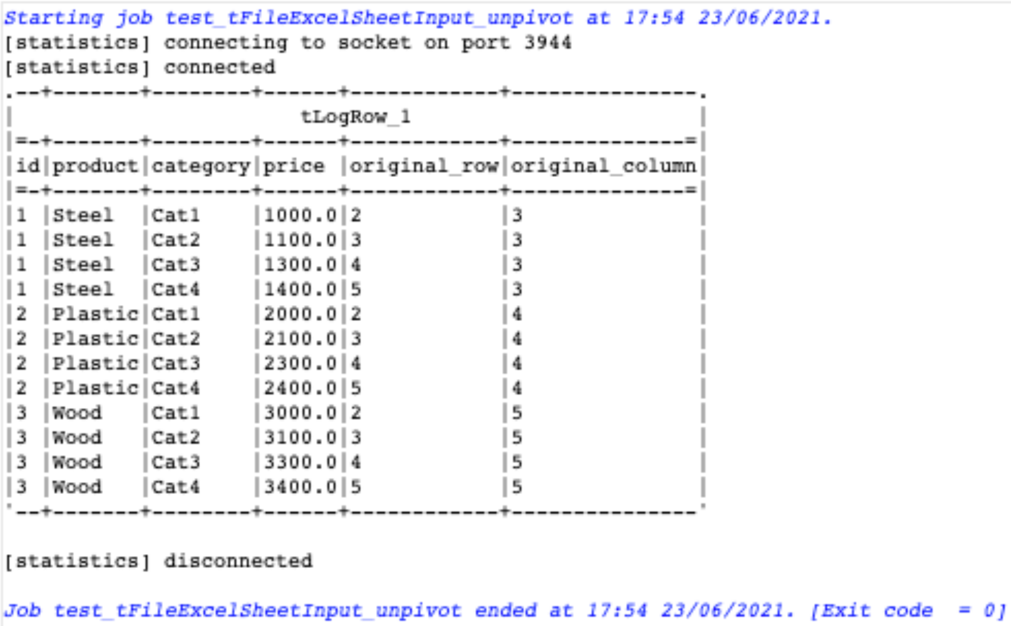
Scenario: Unpivot price data with product and price categories

Input data:



	A	B	C	D	E	F	G
1			Price categories				
2	ID	Product	Cat1	Cat2	Cat3	Cat4	
3	1	Steel	1000	1100	1300	1400	
4	2	Plastic	2000	2100	2300	2400	
5	3	Wood	3000	3100	3300	3400	
6							
7							
8							

Expected normalized output data:

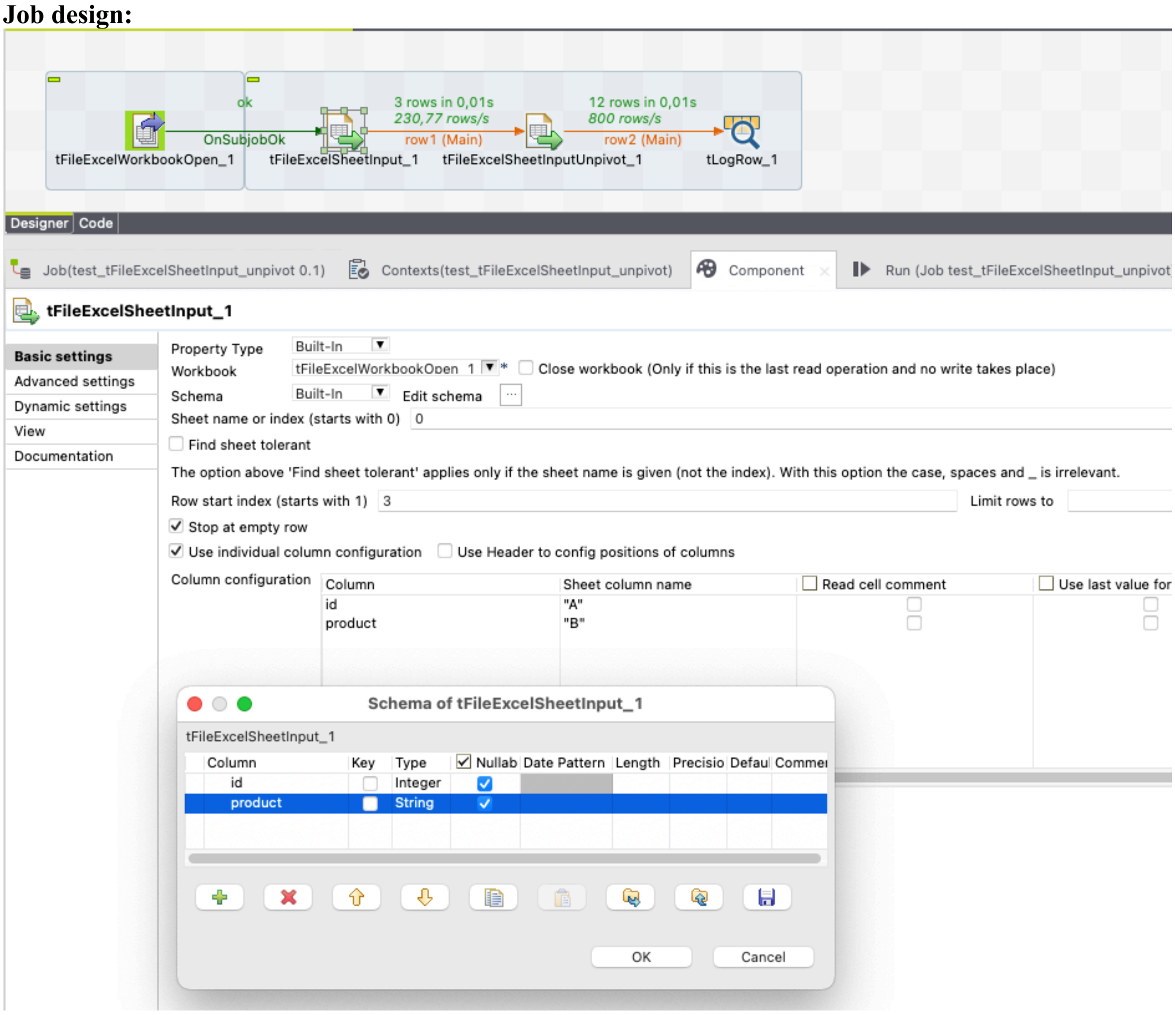


```
Starting job test_tFileExcelSheetInput_unpivot at 17:54 23/06/2021.
[statistics] connecting to socket on port 3944
[statistics] connected
.--+-------+--------+------+------------+---------------.
|                       tLogRow_1                        |
|=-+-------+--------+------+------------+--------------=|
|id|product|category|price |original_row|original_column|
|=-+-------+--------+------+------------+--------------=|
|1 |Steel  |Cat1    |1000.0|2           |3              |
|1 |Steel  |Cat2    |1100.0|3           |3              |
|1 |Steel  |Cat3    |1300.0|4           |3              |
|1 |Steel  |Cat4    |1400.0|5           |3              |
|2 |Plastic|Cat1    |2000.0|2           |4              |
|2 |Plastic|Cat2    |2100.0|3           |4              |
|2 |Plastic|Cat3    |2300.0|4           |4              |
|2 |Plastic|Cat4    |2400.0|5           |4              |
|3 |Wood   |Cat1    |3000.0|2           |5              |
|3 |Wood   |Cat2    |3100.0|3           |5              |
|3 |Wood   |Cat3    |3300.0|4           |5              |
|3 |Wood   |Cat4    |3400.0|5           |5              |
'--+-------+--------+------+------------+---------------'

[statistics] disconnected

Job test_tFileExcelSheetInput_unpivot ended at 17:54 23/06/2021. [Exit code  = 0]
```

Job design:



tFileExcelSheetInput_1

Property Type: Built-In

Workbook: tFileExcelWorkbookOpen_1 * □ Close workbook (Only if this is the last read operation and no write takes place)

Schema: Built-In   Edit schema ...

Sheet name or index (starts with 0): 0

□ Find sheet tolerant

The option above 'Find sheet tolerant' applies only if the sheet name is given (not the index). With this option the case, spaces and _ is irrelevant.

Row start index (starts with 1): 3   Limit rows to

☑ Stop at empty row

☑ Use individual column configuration   □ Use Header to config positions of columns

Column configuration

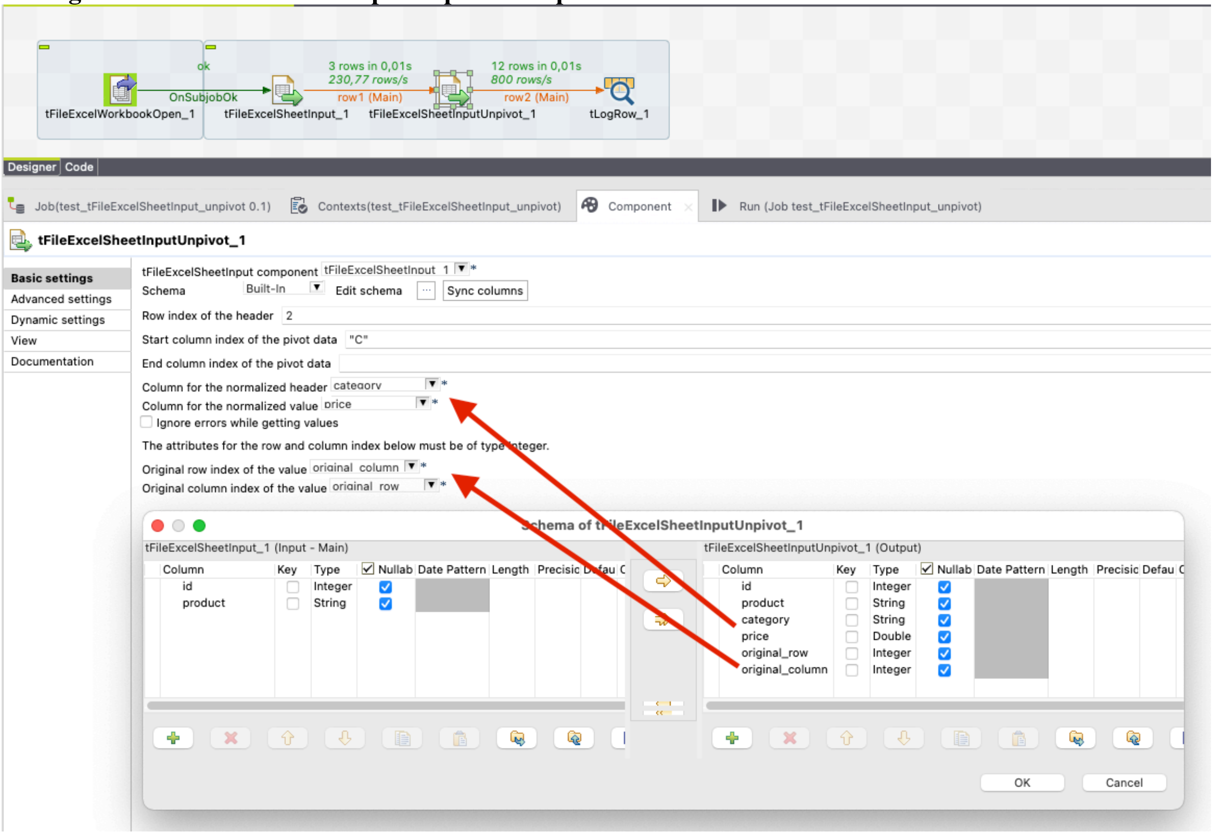
Column	Sheet column name	□ Read cell comment	□ Use last value for
id	"A"	□	□
product	"B"	□	□

Schema of tFileExcelSheetInput_1

tFileExcelSheetInput_1

Column	Key	Type	☑ Nullab	Date Pattern	Length	Precisio	Defaul	Commei
id	□	Integer	☑					
product	□	String	☑					

Settings of the tFileExcelSheetInputUnpivot component:



tFileExcelSheetInputUnpivot_1

tFileExcelSheetInput component: tFileExcelSheetInput_1 *

Schema: Built-In   Edit schema ...   Sync columns

Row index of the header: 2

Start column index of the pivot data: "C"

End column index of the pivot data:

Column for the normalized header: category *

Column for the normalized value: price *

□ Ignore errors while getting values

The attributes for the row and column index below must be of type Integer.

Original row index of the value: original_column *

Original column index of the value: original_row *

Schema of tFileExcelSheetInputUnpivot_1

tFileExcelSheetInput_1 (Input - Main)

Column	Key	Type	☑ Nullab	Date Pattern	Length	Precisic	Defau	C
id	□	Integer	☑					
product	□	String	☑					

tFileExcelSheetInputUnpivot_1 (Output)

Column	Key	Type	☑ Nullab	Date Pattern	Length	Precisic	Defau	C
id	□	Integer	☑					
product	□	String	☑					
category	□	String	☑					
price	□	Double	☑					
original_row	□	Integer	☑					
original_column	□	Integer	☑					