tFileExcelSheetOutput

cimt
let your work flow
http://www.cimt-ag.de

Table of Content

Purpose

This component creates and writes into Excel sheets.
This component needs the components tFileExcelWorkbookOpen (open a file or creates a workbook) and tFileExcelWorkbookSave (writes the workbook to the same or another file)
Advantages of this component:

- The columns to write can be set also with gaps
- The format can be set for numbers and dates
- Can create a copy of an existing sheet (templates)
- Can reuse existing styles (even alternating styles)
- Can write formulas
- Can reuse conditional formats for the written cells
- Uses always the latest Apache POI API
- Can include header names (not limited to the schema column names)
- Can set the with of the column fit to the content
- Can write comments
- Can freeze a region (e.g. to see always the header line)

Integration and Configuration

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

Basic Settings Parameters

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. If the sheet does not exist, it will be created automatically.
Create sheet as copy	This is very useful if you have a template sheet (e.g. with styles and conditional formats) and you want to create multiple sheets in the same way.

	You can specify the source sheet with an index or a name (just like in sheet name) *This option will not work if you use the Memory saving mode in the tFileExcelWorkbookOpen component.*
Exchange rows/columns	If this option is true the component writes every dataset in a transposed way. Every new row is at the end a new column and every input schema column fills a new row in the current excel column.
Append existing rows	The component detect the last row in the given workbook and starts writing at the next row.
Row start index	Enabled if Append existing rows is switched off. This row is the first row where the first dataset (or the header row) will be written. The number is 1-based (for a better understanding within excel row number)
Include header	At first row the header will be written. Normally the component takes the schema column names but you could also use self-defined names in the Column Configuration.
Use individual column positions	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 Configuration	**Column:** the schema column **Header nam**e: an optional header name, if blank the schema name will be used **Sheet column name:** the excel column where the schema column will be written at **Date or Number format:** the format of the cell in excel. This is useful to set an appropriated date format or a number format. You can see all possible formats in Excel under custom formats. This option is only enabled of you do not use the existing styles! **Auto size:** the column size will be set automatically depending on the largest size of the content **Comment:** The content will be written as comment in the excel cell. The comment will be shown with a default width: 3 column and a height: 2 rows. The cell value will not be changed. Therefore it is possible to use for such columns an already used cell column without losing the formally written value. **Hyperlink:** The content will be written as hyperlink url. The actual value of this cell will not be changed. If you do not have an explicit value for the cell set the url also as String typed value for this cell. Hyperlinks starts with an URL protocol followed by :// will be handled as URL. Hyperlinks starting with "mailto:" will be handled as email link. All other hyperlinks will be handled as file type hyperlink. **Group rows by:** If checked all rows, which has in this column the same value will be grouped. It can have side effects if you check more than one column for grouping when the value ranges overlaps. The grouping could be build unlike you expect it. Refer to the scenario 3.
Overwrite existing cell content with null allowed	If you want to keep existing content of a cell and avoid to clear it with null values, switch of this option
Remove last empty rows	After the processing of the input flow, the component can delete all rows after the last written row. This is useful if you read an existing excel file and update an existing sheet.
Freeze row	Here you can enter a row number (starting with 1), which should always be visible if the user scrolls the document in Excel.
Group columns	Configure the columns you want to group. Add groups separated with comma of column ranges separated with minus. Example: "A-D, AB-AF" will create 2 groups of columns A-D and AB-AF. Refer scenario 3.
Freeze column	Here you can enter the column name or index (0-based) which column should be keep

	visible at the left side if the user scrolls vertical.
Reuse existing style from the first written row	This works if you read an existing excel file and fill an existing sheet (even if it a template). The component memorizes the styles of all written columns and applies them to all new created rows. The first written row means the first written row with data and does not mean the row containing the header! It is recommended to design you Excel file as a template file (not a Excel template – it means a normal Excel file!) and define header and styles and so on and reuse them in your target document. *In the memory saving mode of tFileExcelWorkbookOpen this function creates sometimes a malformed content (found in Apache POI 3.10 final)*
Reuse existing style alternating	This option enhances the previous option and keeps the styles from the first and the second written data row and applies them to the even and odd new created rows. This is useful if you want to have alternating styles to better separate the rows e.g. with alternating background colours.

Reuse styles for all columns	This option copies the styles not only for the written column, instead it copies it for all existing columns in the row. But this also means, the all cells will be created if missing and this could increase the size of the document.
Reuse conditional formats from the first data row	You have to specify conditional formats in your template sheet and these conditional formats will be extended in its cell range to the written cells. The component clones the existing conditional format as new format with the full cell range. If the conditional format is already defined as full range, the component does nothing on it). This option does not work in a workbook in memory saving mode.
Reuse the row height from the first data row	The height from the first data row will be applied to all new created rows.
Extend cell range for written tables	If the component writes rows into a so-called Table (special Excel concept) the component can extend the initially configured cell range to the cell range of the written cells. Only the row number will be changed, the width of the Table keeps unchanged. This feature is very helpful to update Pivot-tables. See the chapter about Updating Pivot. Refer scenario 3. This option does not work in a workbook in memory saving mode.
Extend data validation for the written rows	If there are data validations within the first row, these data validations will be extended (or actually additional data validation covering the new rows will be added). This option does not work in a workbook in memory saving mode.

Typical examples of Date or Number formats

Format	Example value
YYYY-MM-DD	2014-04-28
DD.MM.YYYY hh:mm:ss	28.04.2014 14:36:59
#,##0.00	1,234.23
####	1234

Please take care you use always the English formats. Excel usually shows a language depending format but internal it use always the English format and the Apache POI API does not translate the formats from language depend formats to English.

Return values of the component

Value	Content
NB_LINE	Number of lines written
ERROR_MESSAGE	Error message if something went wrong
LAST_ROW_INDEX	Index of the last written row in this sheet. This can be used to append in a next sub

	job to the same sheet. This is more secured than use the Append rows option because you know exactly where to start with the next row.

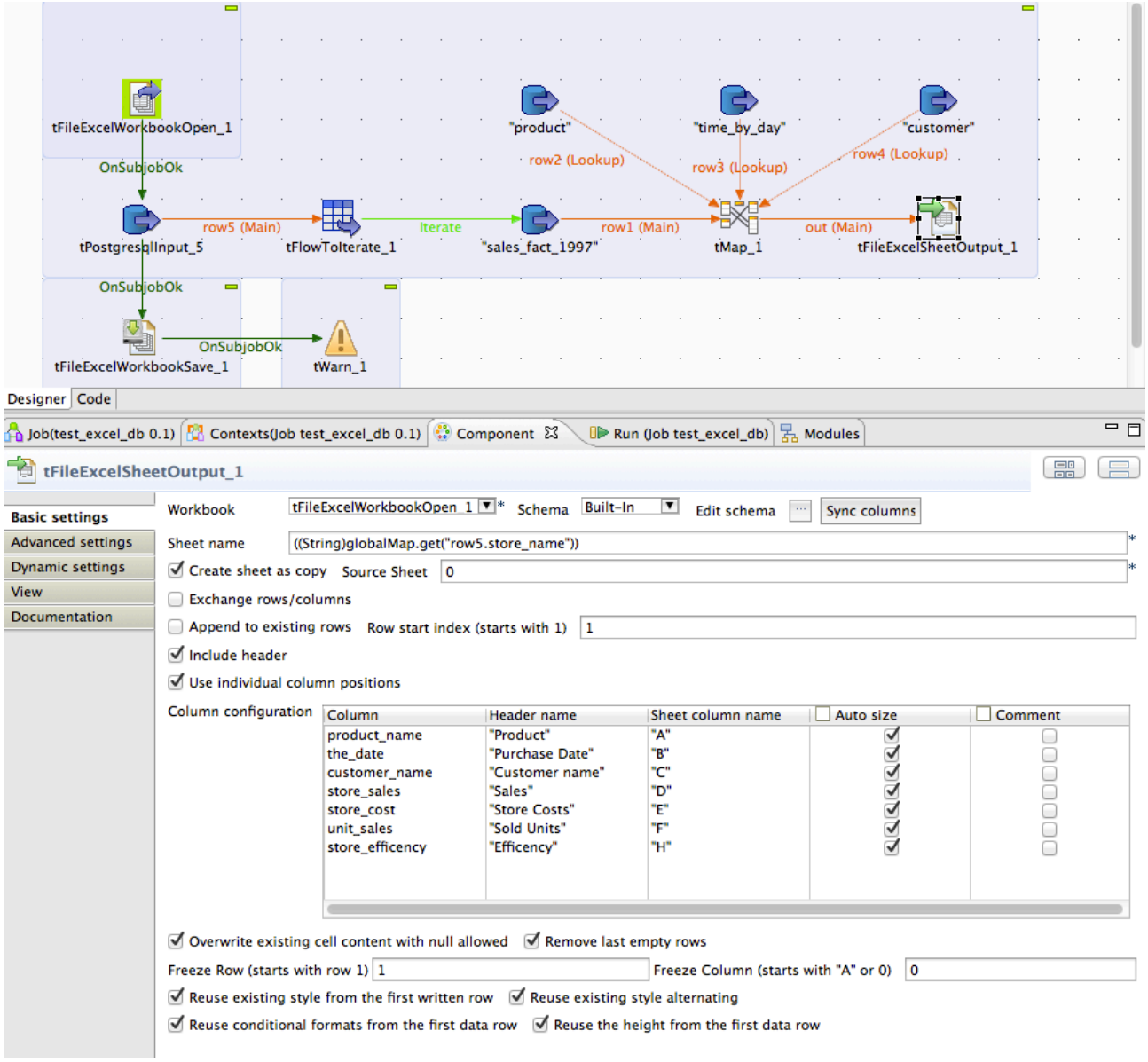
Writing formulas:

It is possible to write formulas with this component. To do this define a String typed schema column and start the content with "=". This will switch the cell type from a simple text type to a formula cell.
All functions must be written in English language. It can be sometimes a bit cumbersome to get the original English name for the functions – sorry there is currently no way to translate them automatically.
The row reference can be set with the expression "{row}" in the formula. The component will replace it with the current row number.
Examples:
"=A{row}+D{row}"
"=SUM(E{row}:H{row})"
It is generally a good idea if possible to install the English language package for Office.

Scenarios

Scenario 1: Creating multiple sheets according the input data

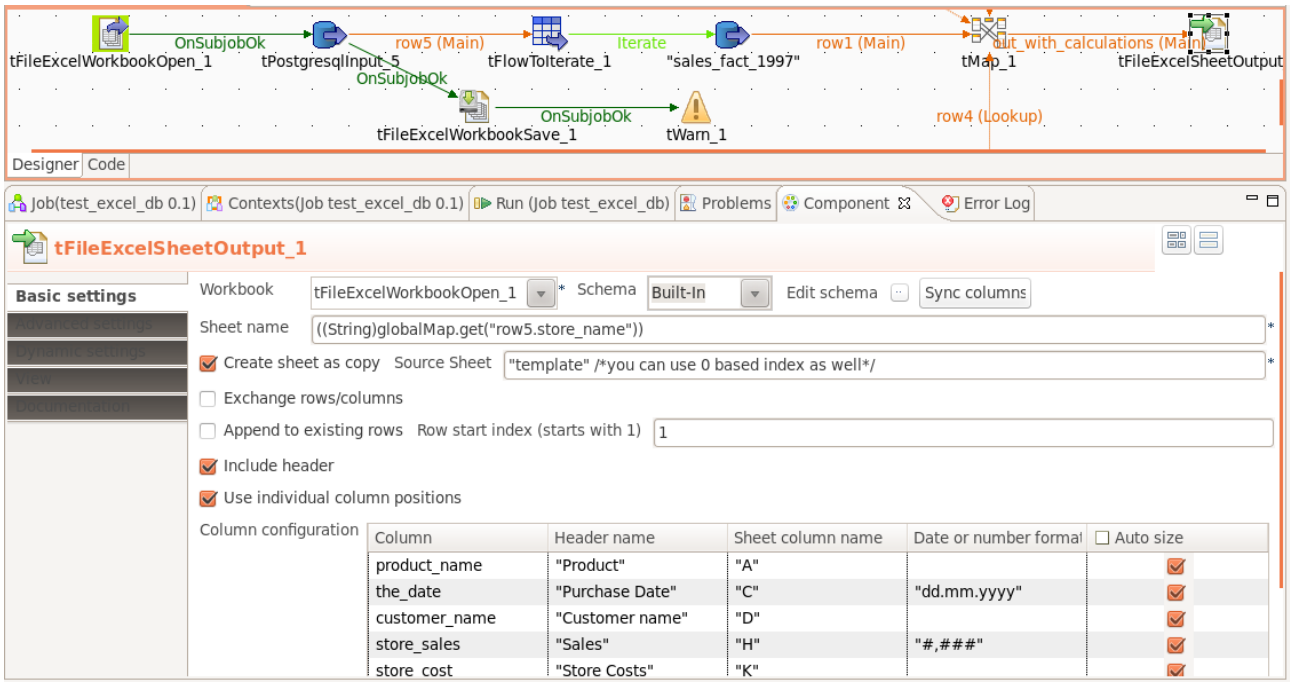
Read a template excel file and create a lot of new sheets base on a template sheet:



In this example the tFileExcelWorkbookOpen reads an existing file.
This file will get as much as iterations happens new sheets as a copy of the first sheet.
At the end a new excel file will be written with tFileExcelWorkbookSave.

Scenario 2: Write Excel file with self defined header and formats

Create a new Excel file without using a template and define the data formats for columns.



As template for the new sheets a sheet with the name “template” will be used here. You can name it as you want.
You will get the column Date or number format if you switch off the option “Reuse existing style from the first row”.

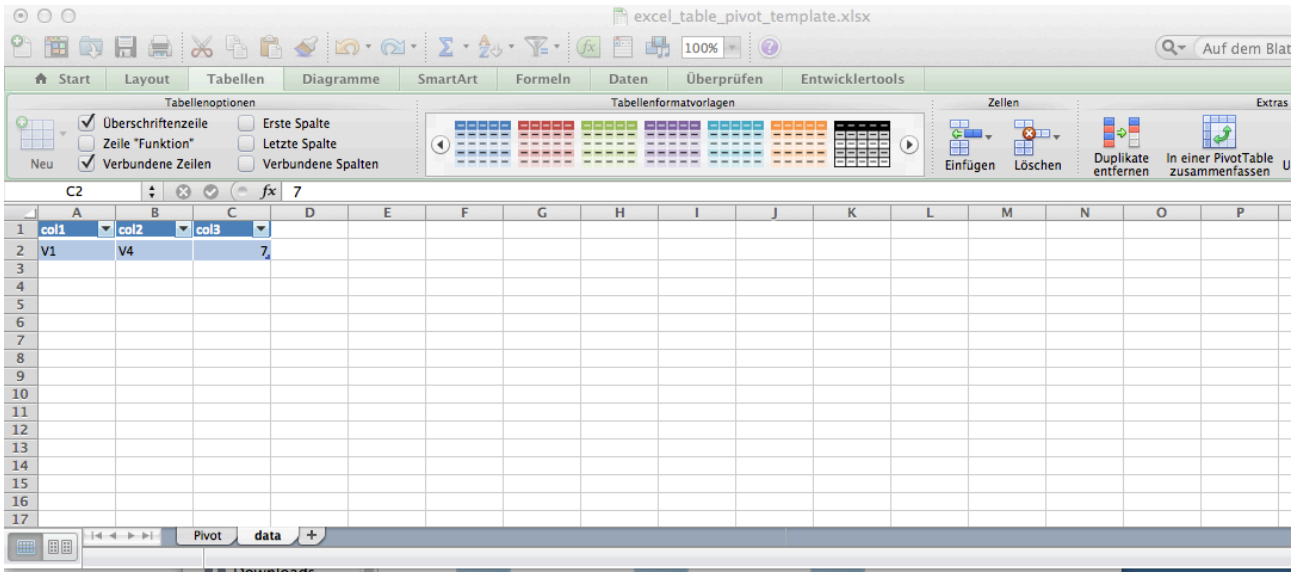
Scenario 3: Updating Pivot-tables

A normal Pivot-table is based on a preconfigured cell range. Unfortunately we cannot configure or create the Pivot-table directly. If a Pivot-table is based on an Excel-table the Pivot-table use always all data in this Excel-table even when the Excel-table has more rows.
The way to update a Pivot-table is to update and reconfigure the underlying Excel-table.
The Pivot-table should be configured as “Update data when the document will be opened”

Following steps are necessary to update a Pivot-table:

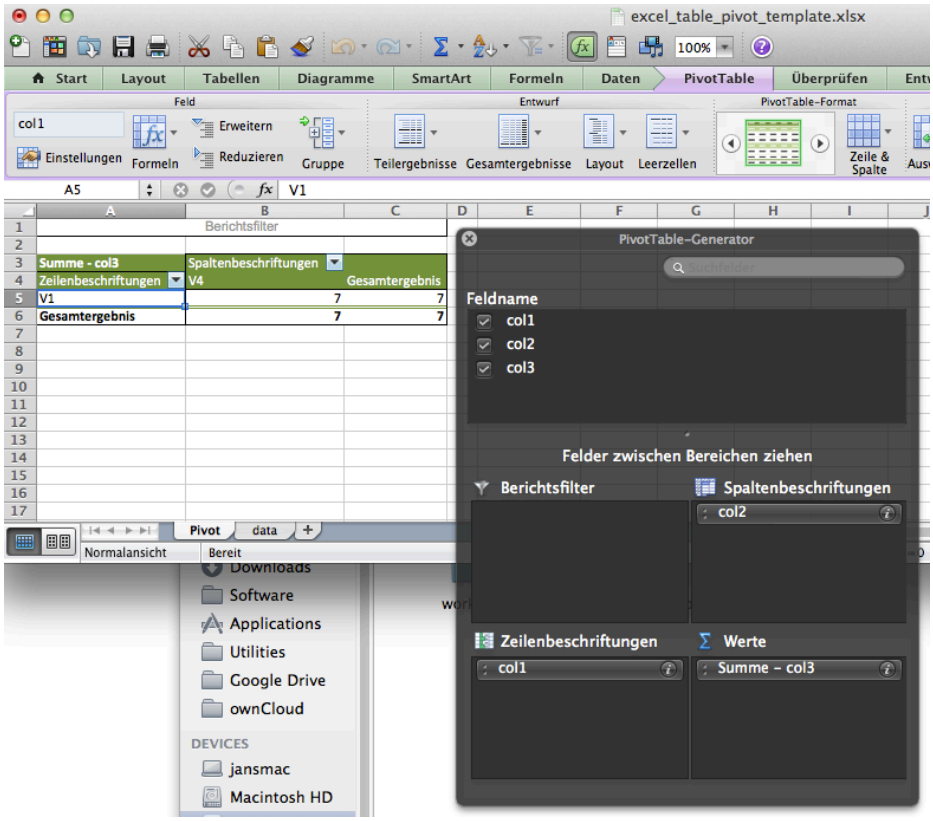
1. Create an Excel file and add an Excel-table
2. Fill this Excel-table with some example data and create a Pivot-table based on this Excel-table
3. Use this file (we now call it the excel template) in your job in the tFileExcelWorkbookOpen component
4. Configure the tFileExcelSheetOutput to write with the first data row (a header line does not make sense if you write into an Excel-table) in the first data row of the Excel-table.
5. Check the option “Extend cell range for written tables”

Step 1 Create the Excel file with the Excel-table



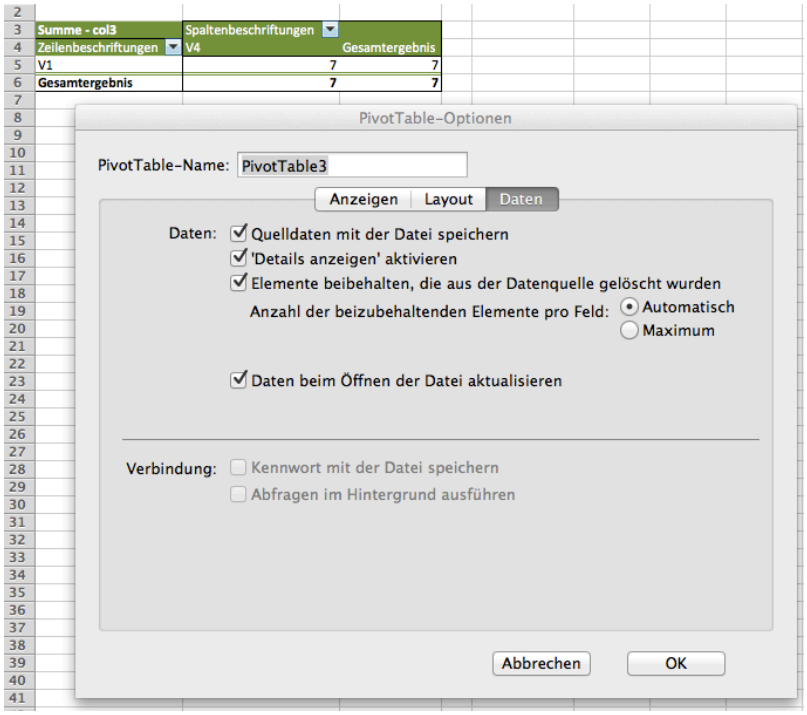
This is the sheet containing the Excel-table. Address this sheet in the tFileExcelSheetOutput component.
All overdue rows will be automatically deleted.
In workbook with the memory saving mode of tFileExcelWorkbookOpen this function creates sometimes a malformed content (found in Apache POI 3.10 final) or simply does not work!

Step 2 Create Pivot based on the Excel-table



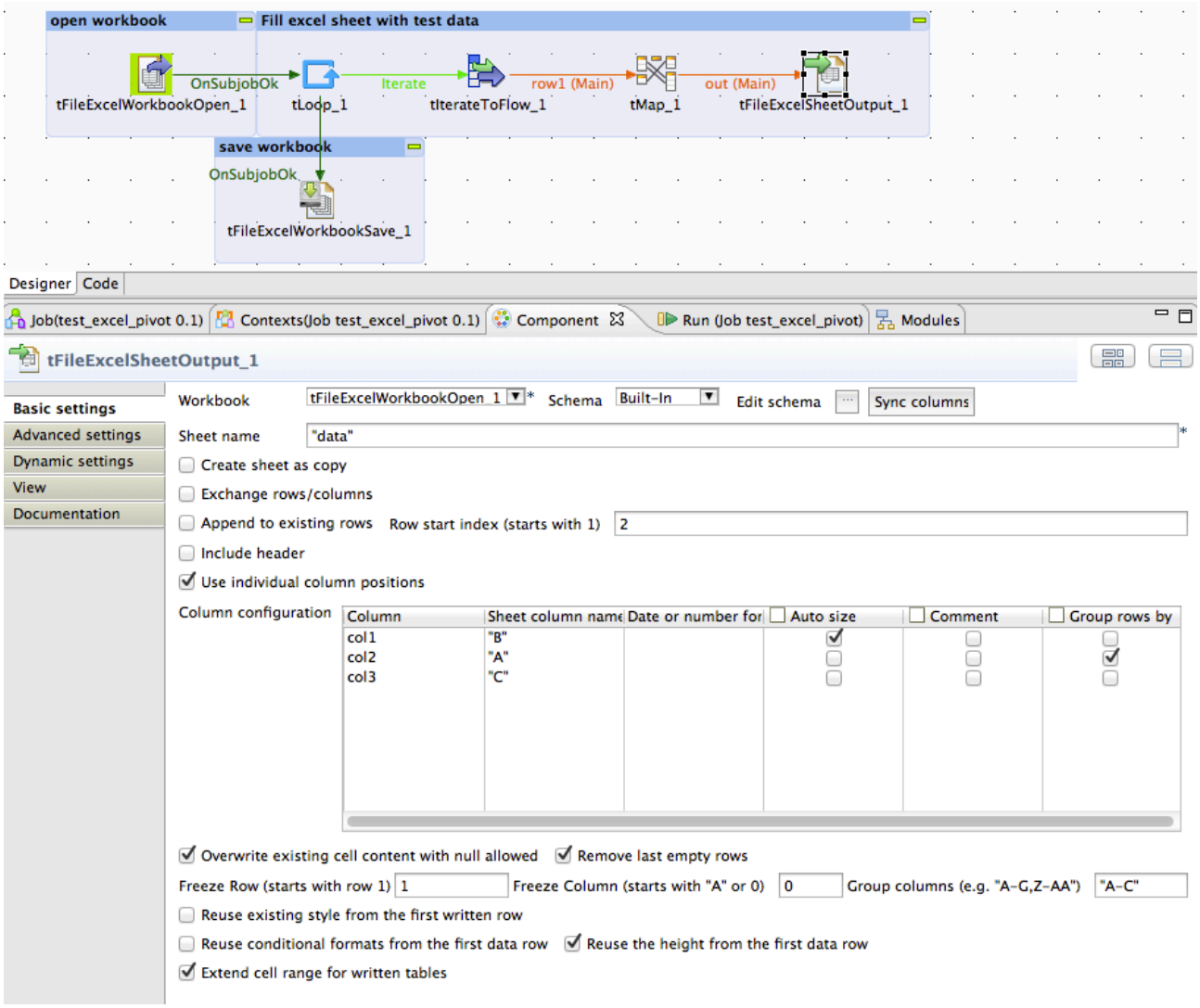
Configure the Pivot-table with the columns from the Excel-table.

In the options of the Pivot-table setup the table to “Update pivot when document opened”



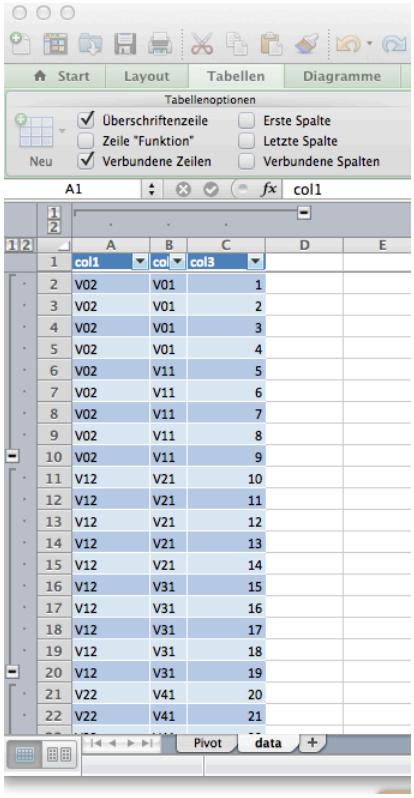
“Daten beim Öffnen der Datei aktualisieren” -> Refresh data while opening the file…

Step 3-5 will be done in Talend in the Job

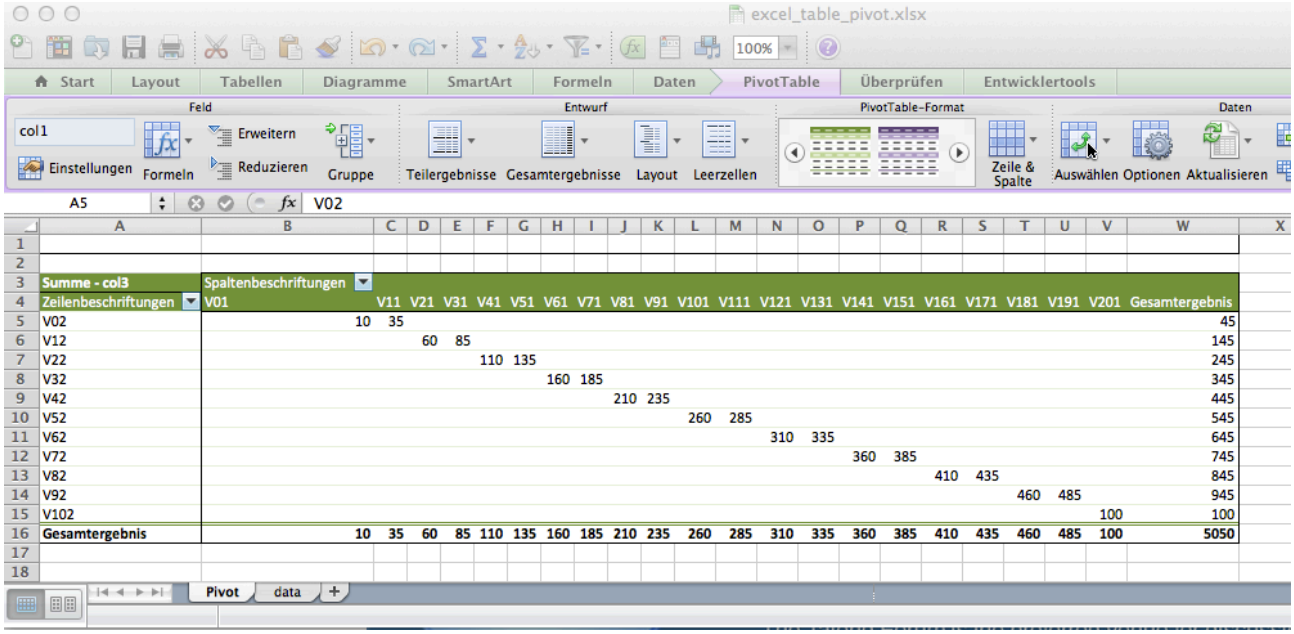


In the component tFileExcelWorkbookOpen choose your just created excel template file.
In the tFileExcelSheetOutput set the option Extend cell range for written tables – means all Excel-tables affected by the writing of this component will be extended in its cell range to the amount of the written rows.
The tLoop component will be used to create dummy example data. Normally you will find here a database input component or similar inputs from another data source probably also processed with a tMap.

As the result of this job the Excel-table is filled with data and the Pivot-table refreshes it self when Excel open this result file. This output file shows also the grouping function.



… here the result pivot table:

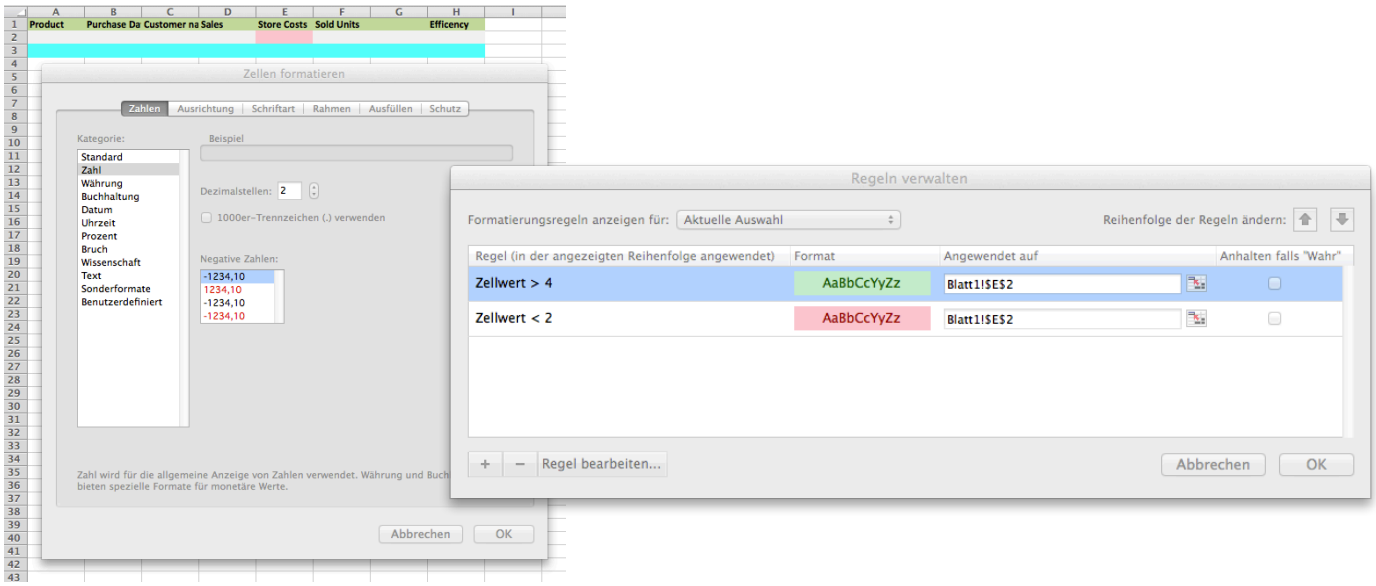


Summe - col3	Spaltenbeschriftungen																						
Zeilenbeschriftungen	V01	V11	V21	V31	V41	V51	V61	V71	V81	V91	V101	V111	V121	V131	V141	V151	V161	V171	V181	V191	V201	Gesamtergebnis
V02	10	35																				45
V12			60	85																		145
V22					110	135																245
V32							160	185														345
V42									210	235												445
V52											260	285										545
V62													310	335								645
V72															360	385						745
V82																	410	435				845
V92																			460	485		945
V102																					100	100
Gesamtergebnis	10	35	60	85	110	135	160	185	210	235	260	285	310	335	360	385	410	435	460	485	100	5050

Scenario 4: Reuse conditional formats

In this scenario we use also a template excel file in which all formats will be configured.
It is recommended to create for every related excel column its own conditional format.
If you want alternating colours define them in the first two data rows in excel.

Here an example template file screenshot:



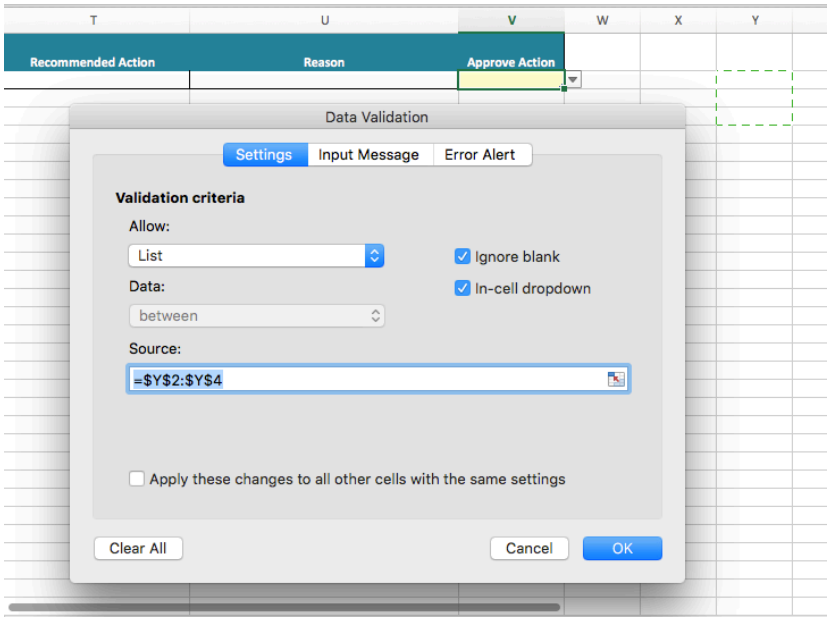
Here the result after filling the sheet with the option “Reuse existing style from the first row” and “Reuse the existing style alternating” and “Reuse the conditional formats from the first data row”:

	A	B	C	D	E	F	G	H
1	Product	Purchase Date	Customer name	Sales	Store Costs	Sold Units		Efficency
2	Big Time Frozen Carrots	28.1.2012	Gobel Buss	3,28	1,1808	2,00		2,78
3	Big Time Lemon Popsicles	28.1.2012	Gobel Buss	2,94	1,1466	2		2,5641
4	Carlson Blueberry Yogurt	28.1.2012	Gobel Buss	5,52	2,3184	2,00		2,38
5	Tri-State Mixed Nuts	28.1.2012	Polly Lira	5,28	1,9008	3		2,77778
6	Landslide Decaf Coffee	28.1.2012	Polly Lira	6,93	3,1185	3,00		2,22
7	Choice Bubble Gum	28.1.2012	Polly Lira	2,12	0,8692	4		2,43902
8	Nationeel Golden Raisins	28.1.2012	Polly Lira	10,1	3,7296	4,00		2,70
9	Walrus White Zinfandel Wine	28.1.2012	Polly Lira	4,83	1,9803	3		2,43902
10	Super Creamy Peanut Butter	28.1.2012	Polly Lira	5,92	2,96	4,00		2,00
11	Imagine Frozen Pancakes	28.1.2012	Shelley Tedrick	5,61	2,3001	3		2,43902
12	Consolidated Tartar Control Toothpaste	28.1.2012	Shelley Tedrick	6,39	2,8116	3,00		2,27
13	Tell Tale Golden Delcious Apples	28.1.2012	Gladys Lindenberg	15,2	6,3672	4		2,38095
14	High Top Fresh Lima Beans	28.1.2012	Gladys Lindenberg	8,61	3,3579	3,00		2,56
15	Moms Sliced Chicken	28.1.2012	Pattie Brumfield	7,23	2,3859	3		3,0303
16	Nationeel Dried Dates	28.1.2012	Pattie Brumfield	9,85	4,8265	5,00		2,04
17	James Bay City Map	28.1.2012	Pattie Brumfield	8,6	2,666	5		3,22581
18	Bird Call 200 MG Acetominifen	28.1.2012	Pattie Brumfield	9,54	3,2436	3,00		2,94
19	Landslide Columbian Coffee	28.1.2012	Carolyne Nelson	8,34	3,2526	3		2,5641
20	PigTail Beef TV Dinner	28.1.2012	Carolyne Nelson	8,56	3,5096	4,00		2,44
21	Red Wing C-Size Batteries	28.1.2012	Bonnie Crawford	8,64	3,024	3		2,85714
22	Genteel Seasoned Hamburger	28.1.2012	Bonnie Crawford	4,6	1,518	4,00		3,03
23	Great Blueberry Muffins	28.1.2012	Bonnie Crawford	7,89	3,2349	3		2,43902
24	Red Wing D-Size Batteries	28.1.2012	Bonnie Crawford	4,5	1,35	3,00		3,33
25	Better Canned Tuna in Water	28.1.2012	Bonnie Crawford	8,31	3,8226	3		2,17391
26	Cutting Edge Low Fat Cole Slaw	28.1.2012	Bonnie Crawford	8,55	3,6765	3,00		2,33
27	Jeffers Oatmeal	28.1.2012	Dolores Convance	4,62	2,1714	3		2,12766
28	Golden Frozen Carrots	28.1.2012	Dolores Convance	2,92	1,3432	2,00		2,17
29	Sunset Room Freshener	28.1.2012	Dolores Convance	9,76	3,1232	4		3,125
30	Landslide Sesame Oil	28.1.2012	Dolores Convance	1,74	0,7656	3,00		2,27
31	PigTail Lemon Popsicles	28.1.2012	Dolores Convance	5,12	1,8944	4		2,7027
32	Ebony Mixed Nuts	28.1.2012	Dolores Convance	2,85	1,026	3,00		2,78
33	ADJ Rosy Sunglasses	28.1.2012	Dolores Convance	8,28	3,3948	3		2,43902

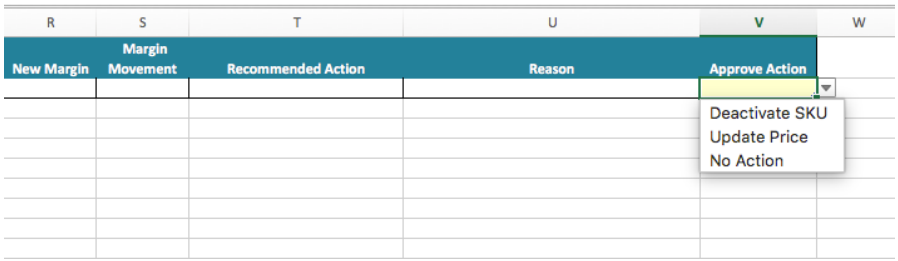
Scenario 5: Reuse data validations

Data validation are beside to the conditional formats an important feature if you have to create Excel files for the accounting department e.g.
The key here is like in conditional formats the template file. The component it self cannot create data validations from scratch but can use existing one and replicate them to the new written rows.

This is a typical setup of a data validation. Do this in the file you will read as template.

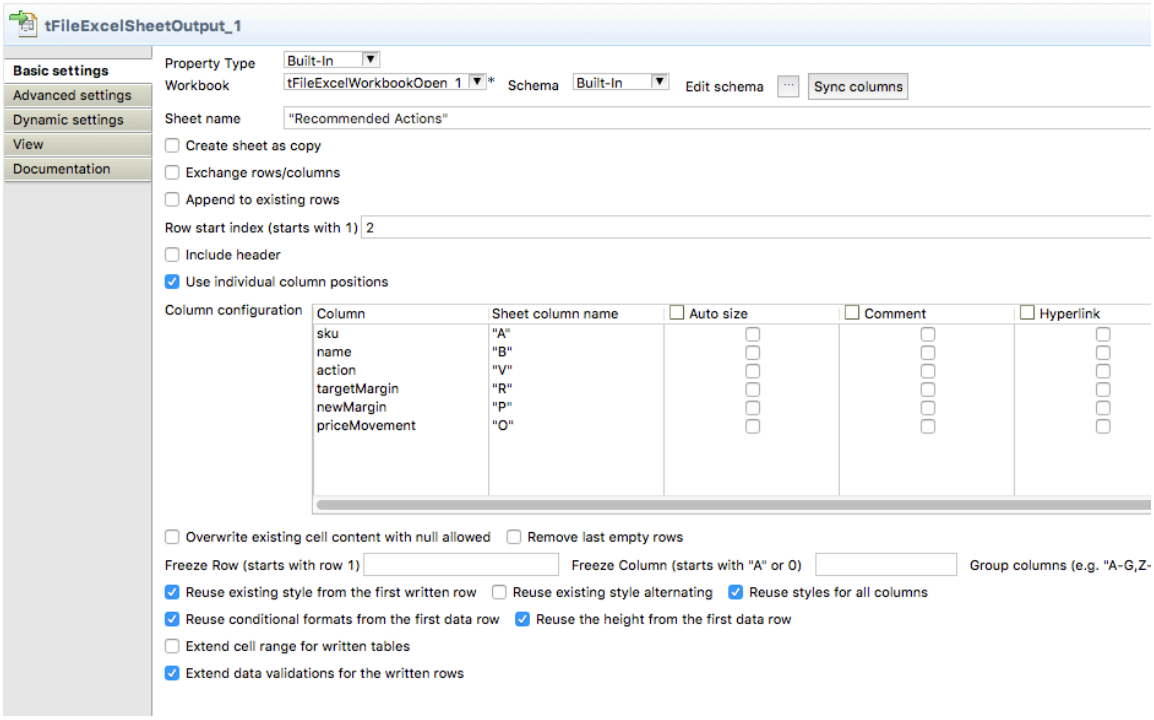


This is how it looks like:



… see in the next page…

Here the necessary setup in the component:
See the checked options “Reuse styles for all columns” and “Extend data validations for written rows”.



… and this how it looks like in the result file:

P	Q	R	S	T	U	V	W
Target Margin	Current Margin	New Margin	Margin Movement	Recommended Action	Reason	Approve Action	
59%		46,00%					
49%		63,00%					
29%		22,00%				Deactivate SKU	
29%		72,00%				Update Price	
65%		71,00%					
12%		40,00%				Deactivate SKU	
55%		86,00%					
35%		49,00%				Update Price	
86%		24,00%				Deactivate SKU	
23%		2,00%					
63%		81,00%					
5%		89,00%				Update Price	
28%		1,00%					
11%		97,00%					
11%		63,00%				Deactivate SKU	
46%		72,00%				Update Price	
8%		87,00%					
67%		0,00%				Deactivate SKU	
82%		51,00%					
92%		13,00%				Update Price	
3%		33,00%				Deactivate SKU	

The result file will have for column V a data validation configured for the first row and an additional data validation for the all rows below.