JOB AID

Processing data

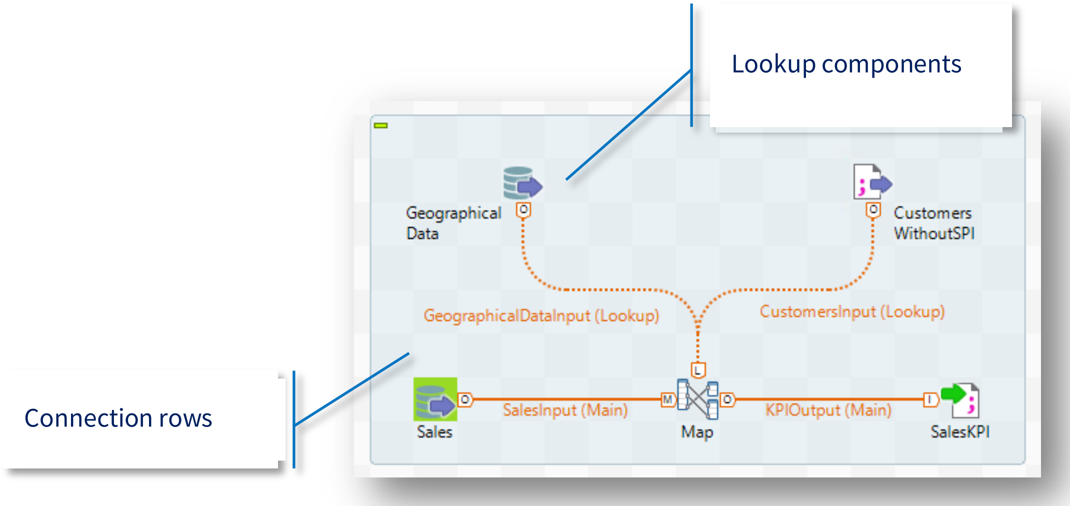
Overview

This module covered several data processing methods. The first part introduced you to data transformation using the tMap component (mapping data, joining data, and collecting join rejects, filtering data, and collecting filter rejects). The second part focused on other processing tasks, such as aggregating, sorting, and sampling.

Best practices

When using tMap, follow these best practices when placing components in the Designer:

- Place lookup components above **tMap**
- Rename connection rows

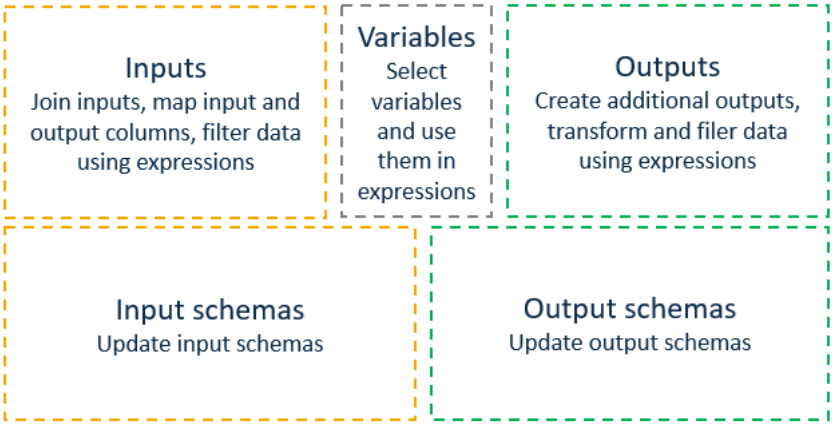
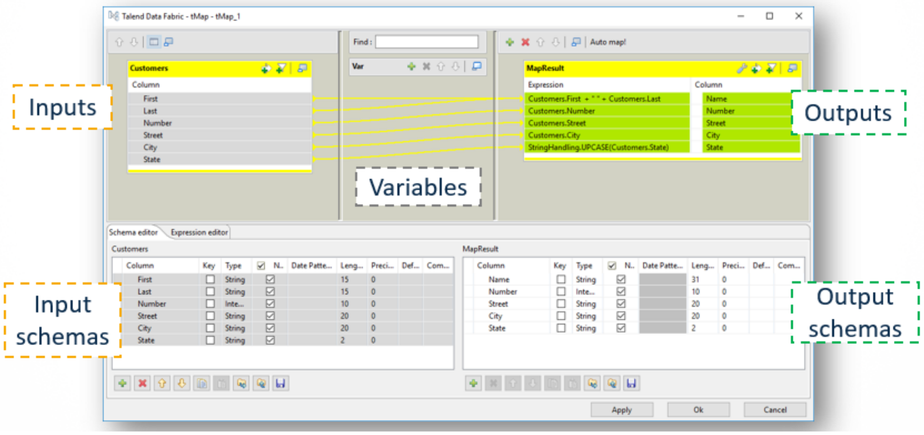


Lookup components

Connection rows

The tMap editor

The tMap editor is divided into panels from which you can perform different actions:

<table>
<tr>
<td>
Inputs
Join inputs, map input and output columns, filter data using expressions
Variables
Select variables and use them in expressions
Outputs
Create additional outputs, transform and filer data using expressions
Input schemas
Update input schemas
Output schemas
Update output schemas</td>
<td>Inputs and outputs are named using the labels you used in the Designer for connection rows.</td>
</tr>
<tr>
<td>

Customers
Column: First, Last, Number, Street, City, State
MapResult
<table>
<tr><th>Expression</th><th>Column</th></tr>
<tr><td>Customers.First + " " + Customers.Last</td><td>Name</td></tr>
<tr><td>Customers.Number</td><td>Number</td></tr>
<tr><td>Customers.Street</td><td>Street</td></tr>
<tr><td>Customers.City</td><td>City</td></tr>
<tr><td>StringHandling.UPCASE(Customers.State)</td><td>State</td></tr>
</table>
Customers
<table>
<tr><th>Column</th><th>Key</th><th>Type</th><th>☑ N..</th><th>Date Patte...</th><th>Leng...</th><th>Preci...</th><th>Def...</th><th>Com...</th></tr>
<tr><td>First</td><td>☐</td><td>String</td><td>☑</td><td></td><td>15</td><td>0</td><td></td><td></td></tr>
<tr><td>Last</td><td>☐</td><td>String</td><td>☑</td><td></td><td>15</td><td>0</td><td></td><td></td></tr>
<tr><td>Number</td><td>☐</td><td>Inte...</td><td>☑</td><td></td><td>10</td><td>0</td><td></td><td></td></tr>
<tr><td>Street</td><td>☐</td><td>String</td><td>☑</td><td></td><td>20</td><td>0</td><td></td><td></td></tr>
<tr><td>City</td><td>☐</td><td>String</td><td>☑</td><td></td><td>20</td><td>0</td><td></td><td></td></tr>
<tr><td>State</td><td>☐</td><td>String</td><td>☑</td><td></td><td>2</td><td>0</td><td></td><td></td></tr>
</table>
MapResult
<table>
<tr><th>Column</th><th>Key</th><th>Type</th><th>☑ N..</th><th>Date Patte...</th><th>Leng...</th><th>Preci...</th><th>Def...</th><th>Com...</th></tr>
<tr><td>Name</td><td>☐</td><td>String</td><td>☑</td><td></td><td>31</td><td>0</td><td></td><td></td></tr>
<tr><td>Number</td><td>☐</td><td>Inte...</td><td>☑</td><td></td><td>10</td><td>0</td><td></td><td></td></tr>
<tr><td>Street</td><td>☐</td><td>String</td><td>☑</td><td></td><td>20</td><td>0</td><td></td><td></td></tr>
<tr><td>City</td><td>☐</td><td>String</td><td>☑</td><td></td><td>20</td><td>0</td><td></td><td></td></tr>
<tr><td>State</td><td>☐</td><td>String</td><td>☑</td><td></td><td>2</td><td>0</td><td></td><td></td></tr>
</table>
</td>
<td>You can update schemas in the tMap editor. You can rename, add, and reorder columns, modify data length and type, and more.</td>
</tr>
</table>

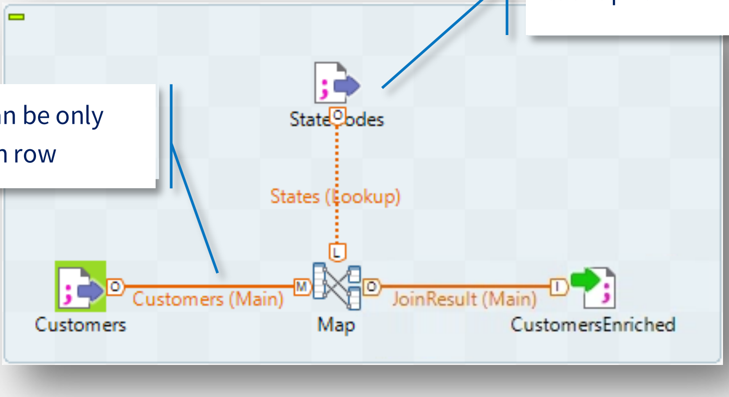
JOB AID

Key steps

Follow these steps to join two data flows using tMap.

1 Define multiple input sources

- Following best practices, organize components and connection rows in the **Designer**
- Carefully choose connections:
  - Data channeled through the Main row is on the left side of the join
  - When using a left outer join, data from the Main row is always protected, even when you have nonmatching records
- Double-click the **tMap** component to open the editor



Other inputs are lookup rows

There can be only one Main row

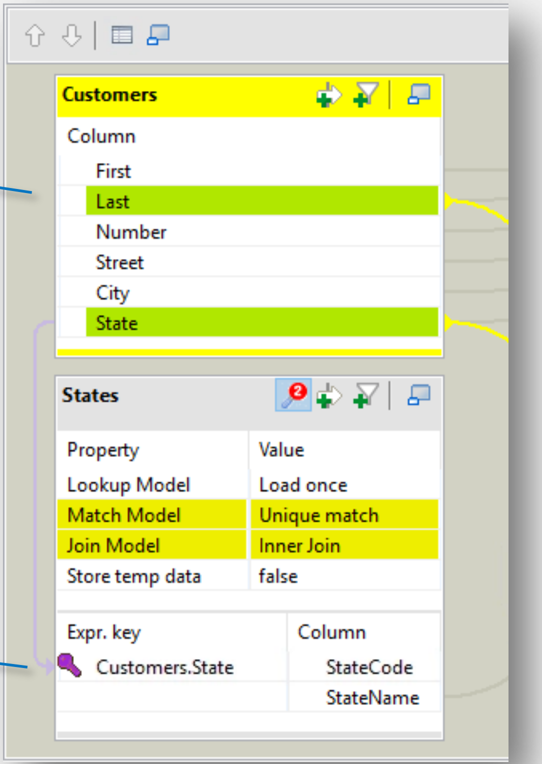
2 Configure a join

To create a join, in the **tMap** input panel:

1. Drag the key join column from the main input table to the corresponding column in the lookup table.
2. In the lookup table, click the **Table settings** button.
3. For **Join Model**, select **Inner Join** or **Left Outer Join**.

You can utilize advanced settings for **Lookup Model** and **Match Model**.

Main input table



Customers

Column
First
Last
Number
Street
City
State

States

Property	Value
Lookup Model	Load once
Match Model	Unique match
Join Model	Inner Join
Store temp data	false

Expr. key	Column
Customers.State	StateCode
	StateName

Lookup table

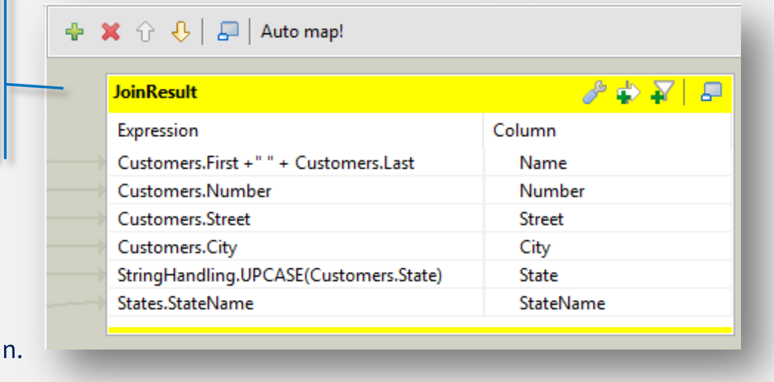
3 Define multiple output sources

To configure output flows, in the **tMap** output panel:

1. Click **[+]** to create as many output tables as you need.
2. Create the output schemas by connecting the columns from the input tables to those in the output tables.

You can utilize advanced settings by editing a mapping expression for each column.

To create outputs, click **[+]**

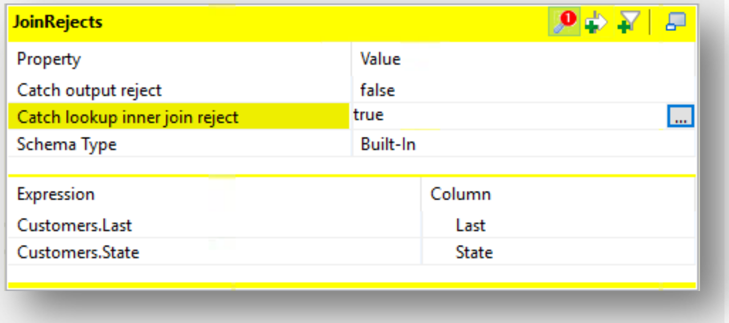


Expression	Column
Customers.First + " " + Customers.Last	Name
Customers.Number	Number
Customers.Street	Street
Customers.City	City
StringHandling.UPCASE(Customers.State)	State
States.StateName	StateName

4 Additional setting for inner joins: handle rejects

To channel rejects to a separate output for additional processing:

1. Create a specific output table.
2. Configure the output schema.
3. Click the **Table settings** button.
4. To the right of **Catch lookup inner join reject**, in the **Value** column, select **true**.



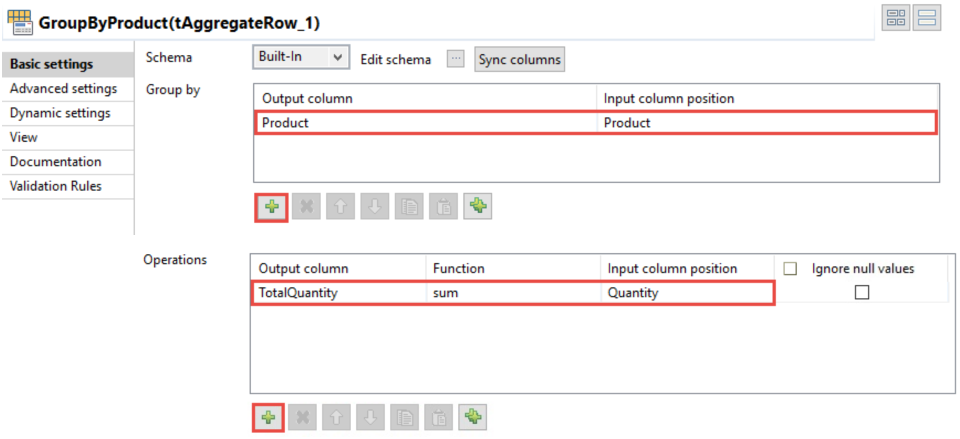
Property	Value
Catch output reject	false
Catch lookup inner join reject	true
Schema Type	Built-In

Expression	Column
Customers.Last	Last
Customers.State	State

Tips

Other elements you can use for different types of data processing tasks:

Aggregation



GroupByProduct(tAggregateRow_1)

Schema: Built-In · Edit schema · Sync columns

Group by

Output column	Input column position
Product	Product

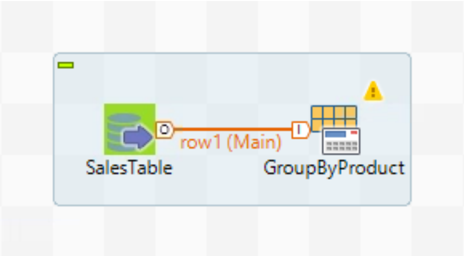
Operations

Output column	Function	Input column position	□ Ignore null values
TotalQuantity	sum	Quantity	□

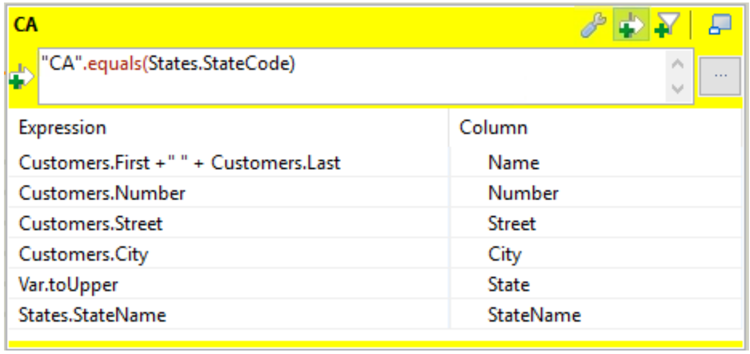
The tAggregateRow component aggregates data using a numeric function (sum, count, count distinct, max, min).

Based on the results of the aggregation you want:

1. Define the output schema.
2. In the **Group by** section, select the output columns.
3. In the **Operations** section, select the aggregation function and the column where it applies.



Filtering data using tMap



CA

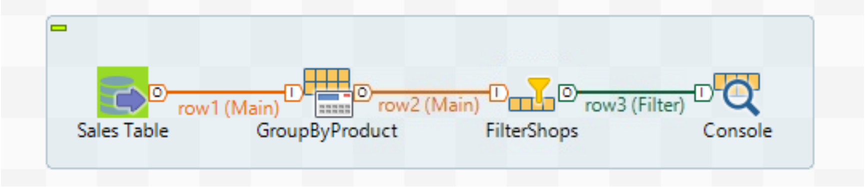
"CA".equals(States.StateCode)

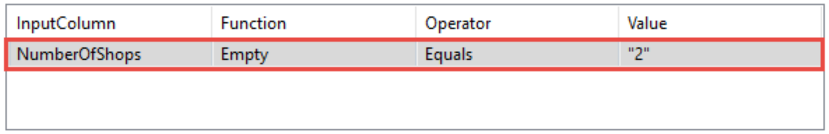
Expression	Column
Customers.First + " " + Customers.Last	Name
Customers.Number	Number
Customers.Street	Street
Customers.City	City
Var.toUpper	State
States.StateName	StateName

You can:

- Apply filter expressions to input and output tables
- Create an output to catch rows rejected by the filter

Filtering data using tFilterRow





InputColumn	Function	Operator	Value
NumberOfShops	Empty	Equals	"2"

You can use the tFilterRow component to graphically set up simpler filters. The component has two outputs, one for filtered and one for rejected rows.