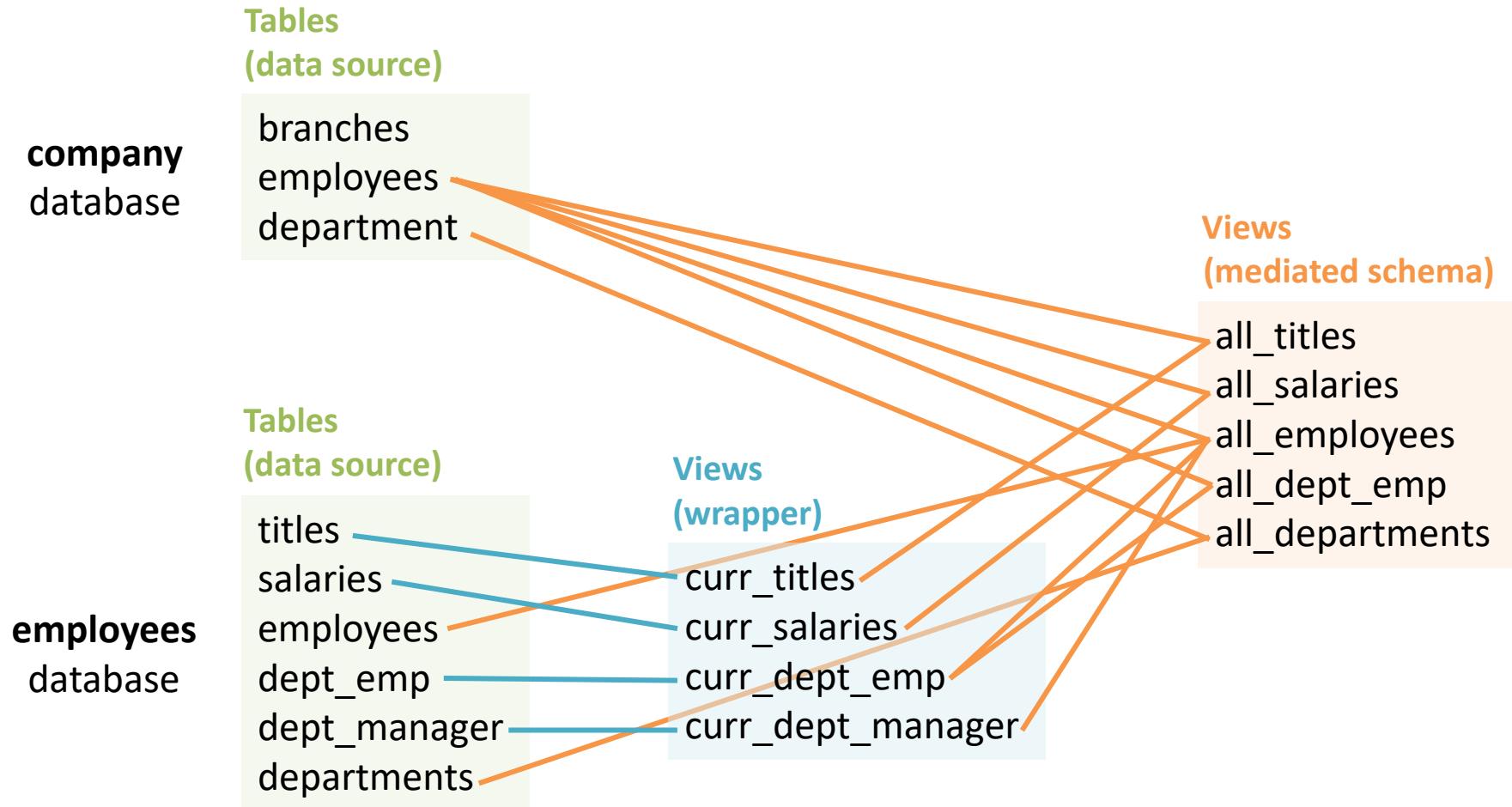


# Data Analysis and Integration

---

Introduction to ETL tools

# Data integration using views

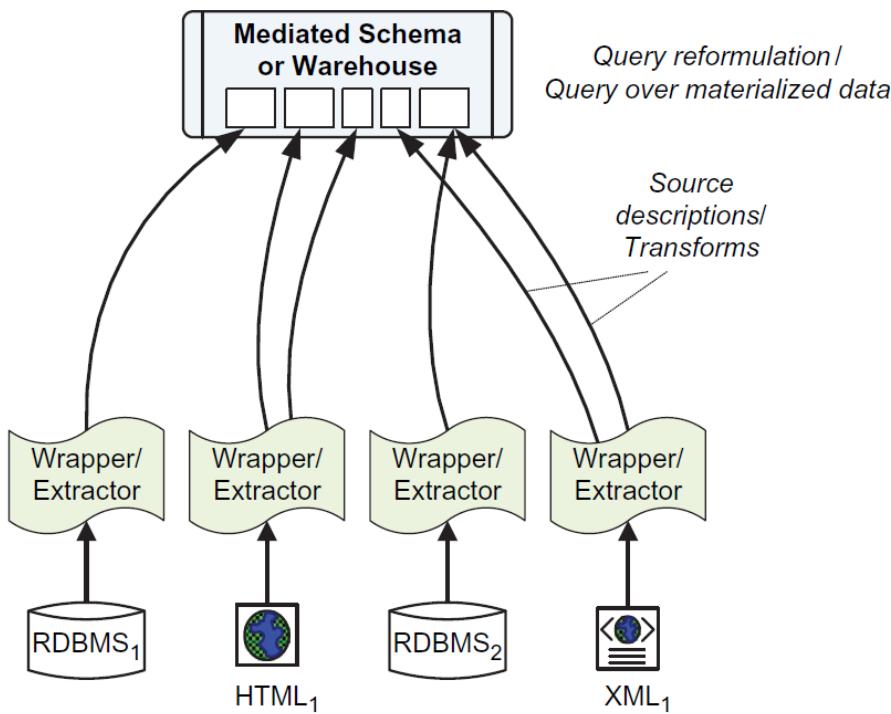


# Data integration using views

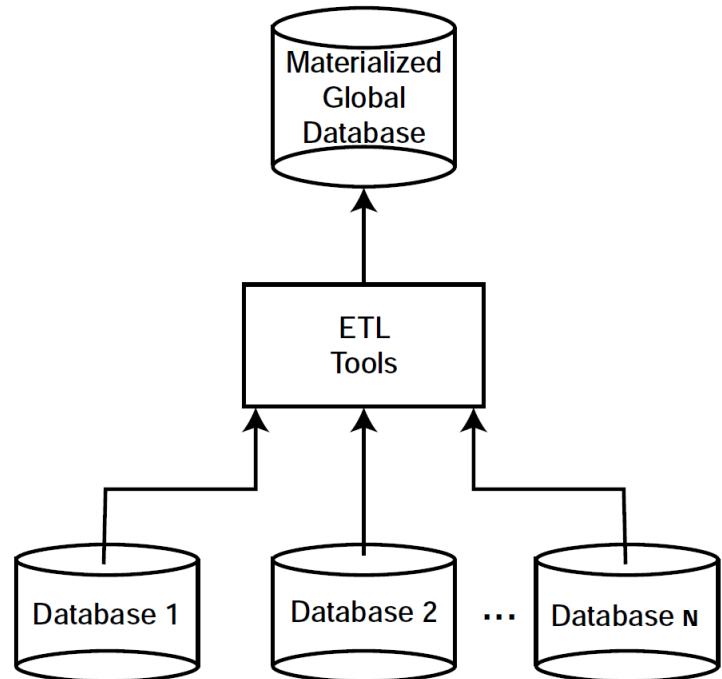
---

- Define the **mediated schema**
- Define the **schema mappings** between the **data sources** (or their **wrappers**) and the **mediated schema**
- Write query over **mediated schema**
- **Query unfolding** reformulates the query over the **mediated schema** and the **wrappers** as a query over the **data sources**
- Results are computed on-the-fly and are always up-to-date

# Data integration vs. data warehousing



A. Doan, A. Halevy, Z. Ives  
*Principles of Data Integration*  
Morgan Kaufmann, 2012



T. Özsu, P. Valduriez  
*Principles of Distributed Database Systems*  
Springer, 2011

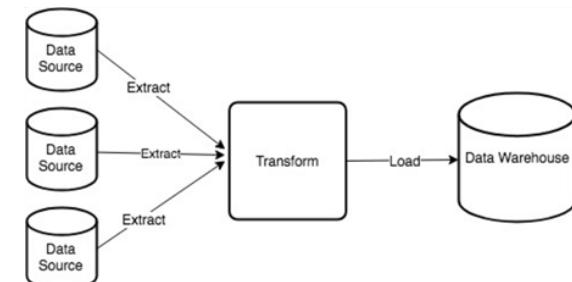
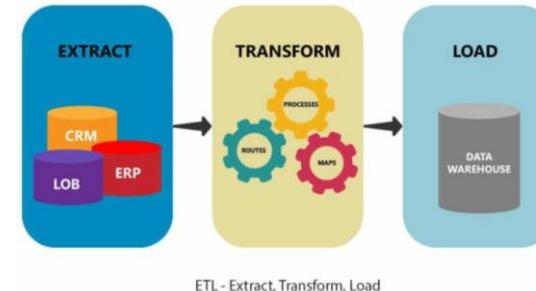
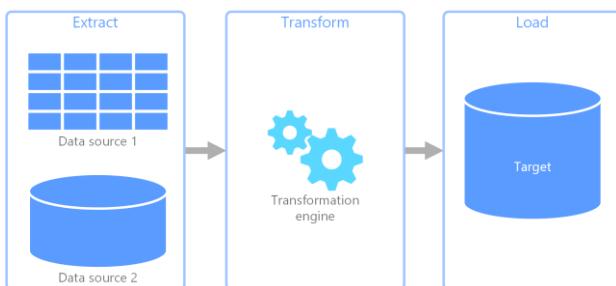
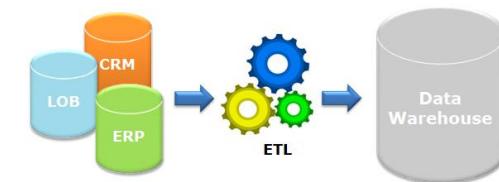
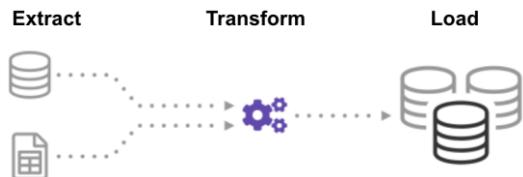
# Data warehousing

---

- Design a **data warehouse**
  - the data warehouse has its own schema, which is different from the data source schemas
- Implement an **extract-transform-load (ETL) process**
  - extract from data sources, transform data, store into data warehouse
- Write query over **data warehouse**
- Retrieve results from storage
  - run ETL process regularly to keep **data warehouse** up-to-date

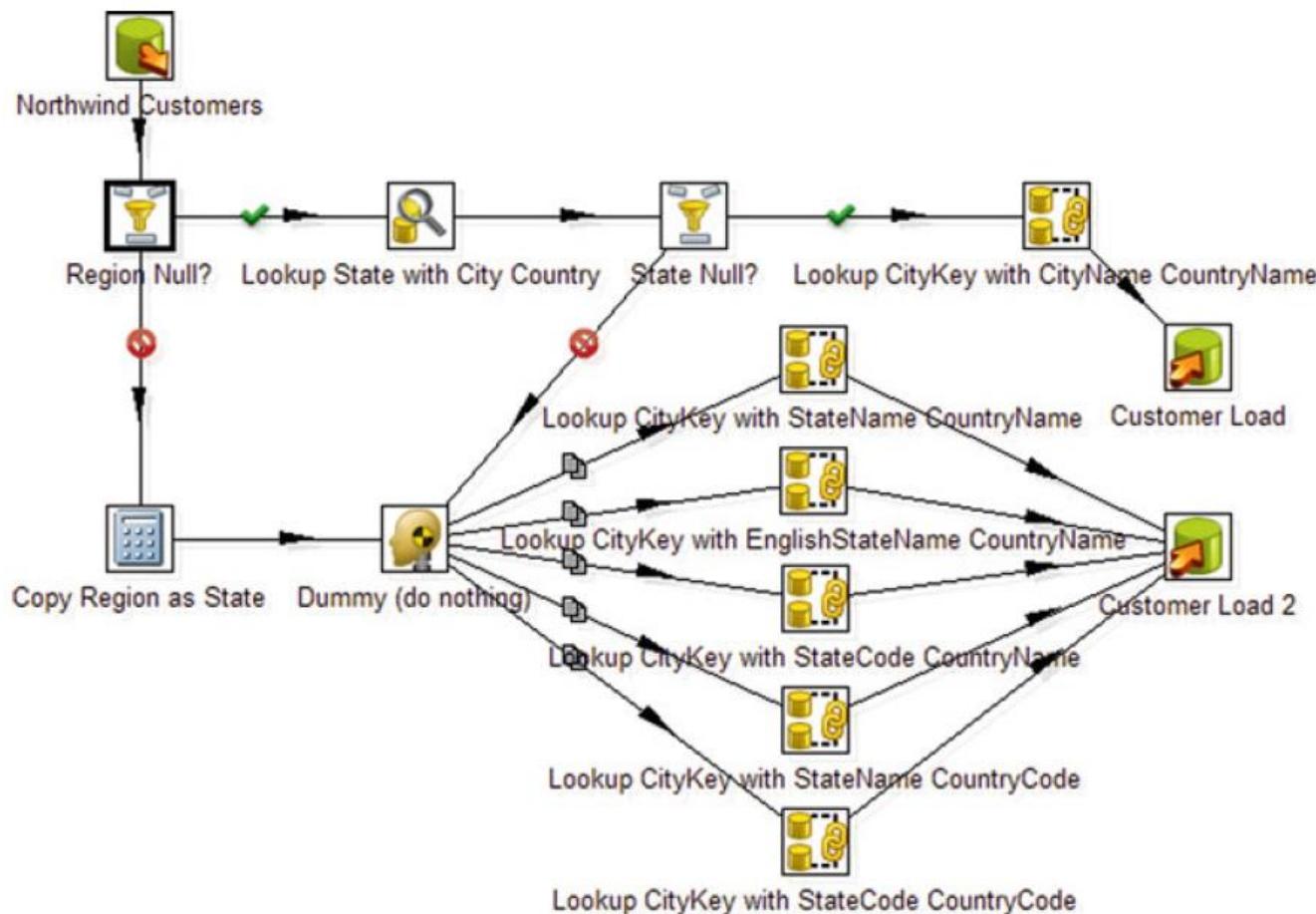
# ETL process

- Extract, transform, load



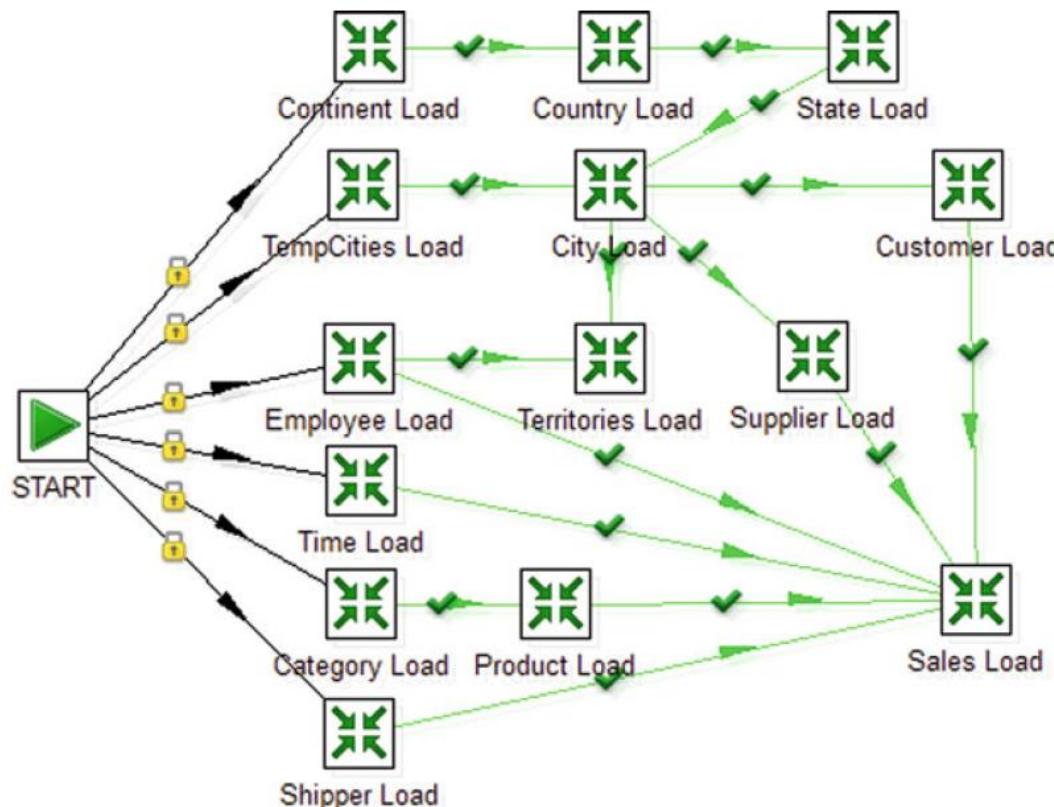
# ETL process

- Extracting, transforming and loading customers

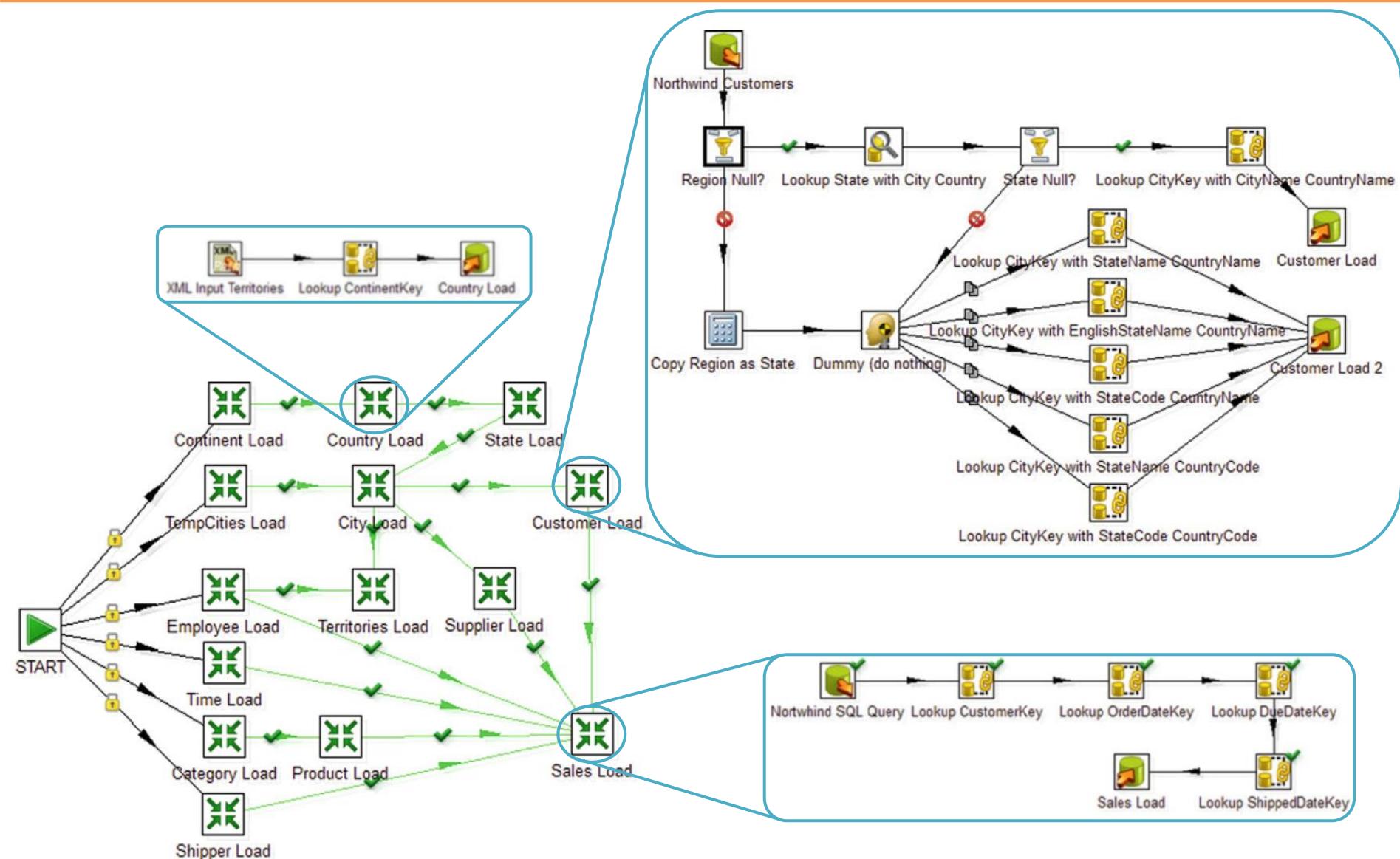


# ETL process

- An ETL process comprises many such transformations



# ETL process

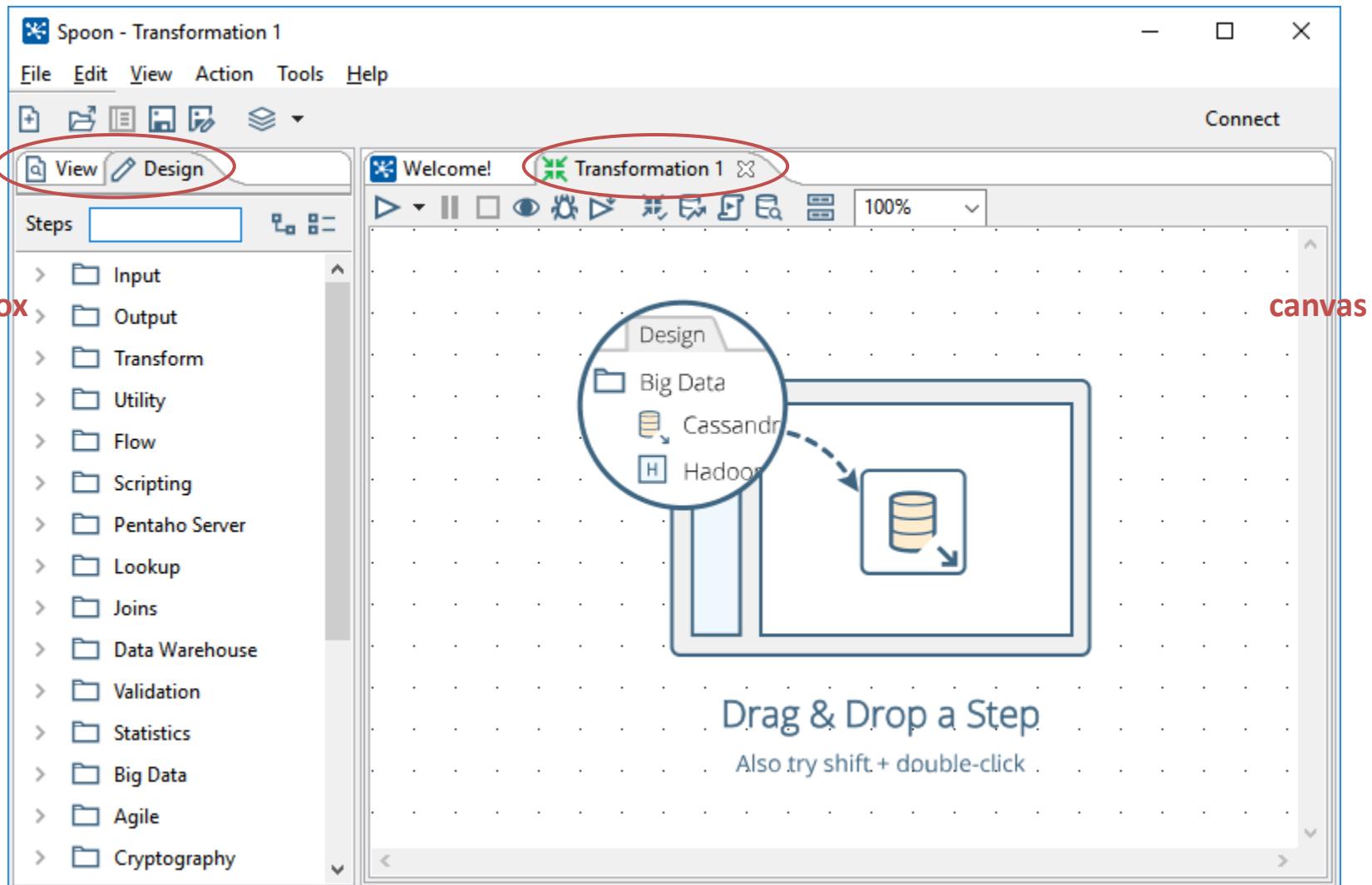


# ETL tools

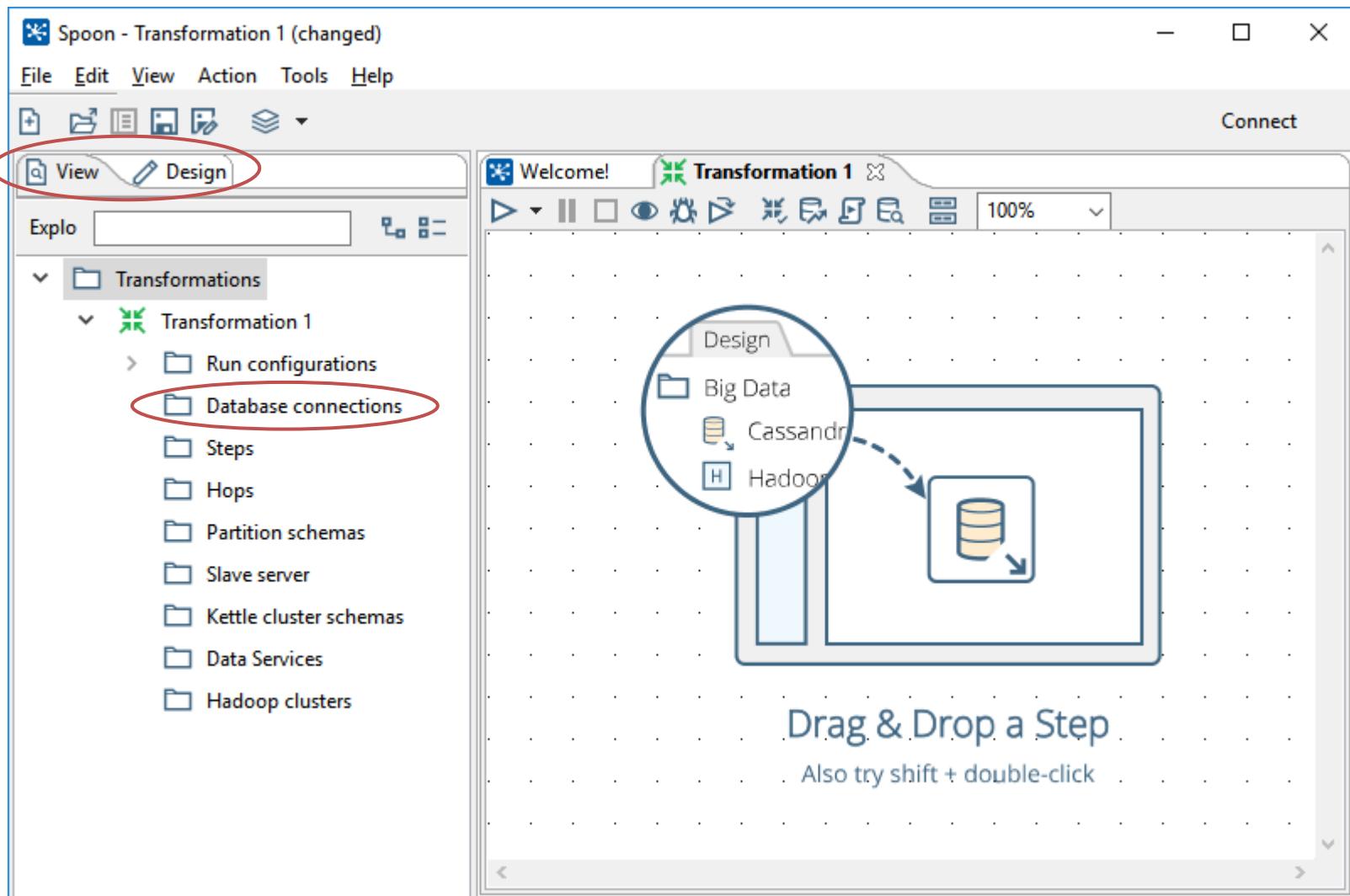
---

- ETL tools
  - what do they do?
  - how do they work?
  - how can we use them?
- The ETL tool that we will be using:
  - Pentaho Data Integration
  - also known as PDI, Kettle, or Spoon
  - competing products (e.g. SQL Server Integration Services)

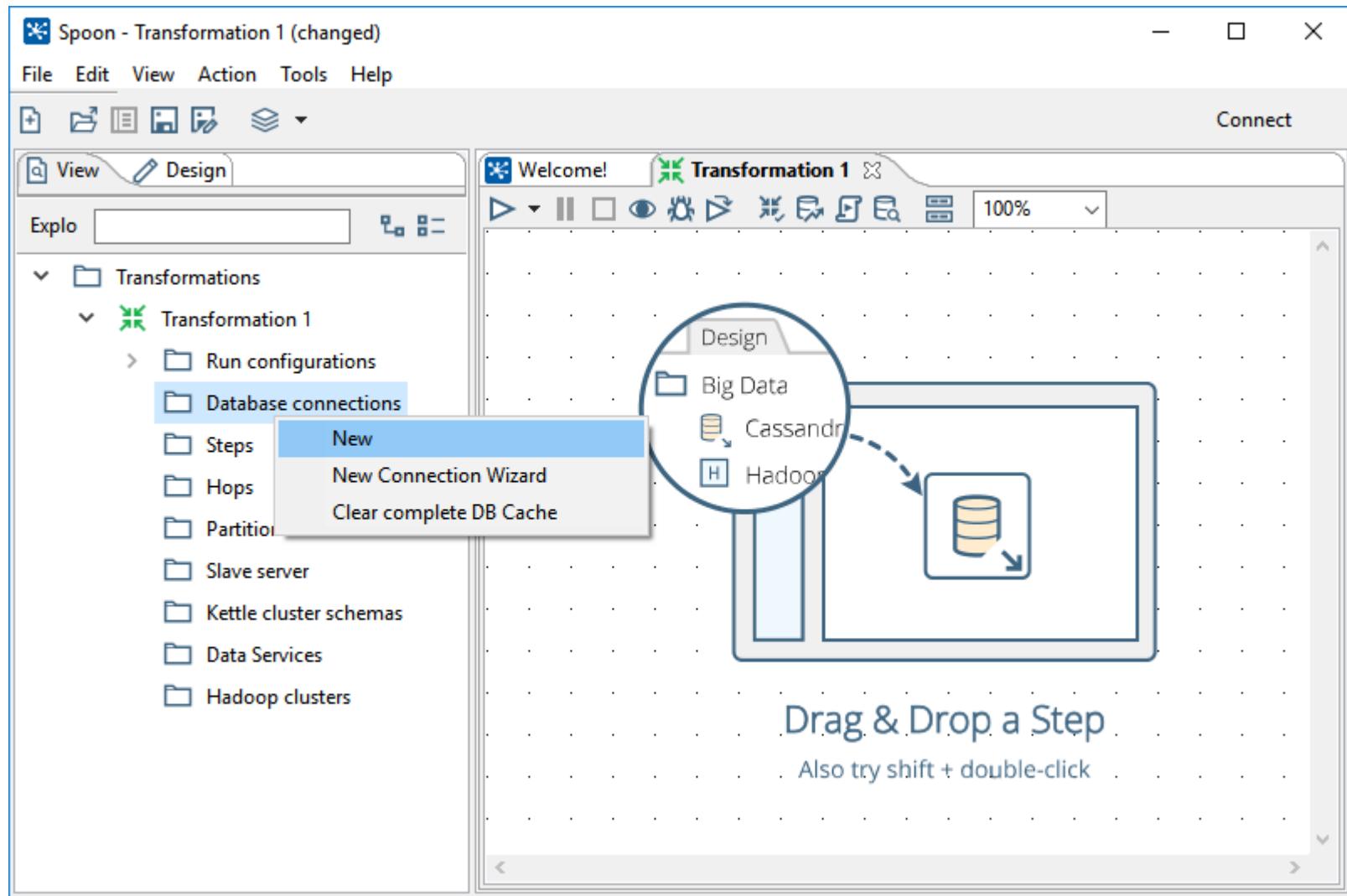
# Workspace



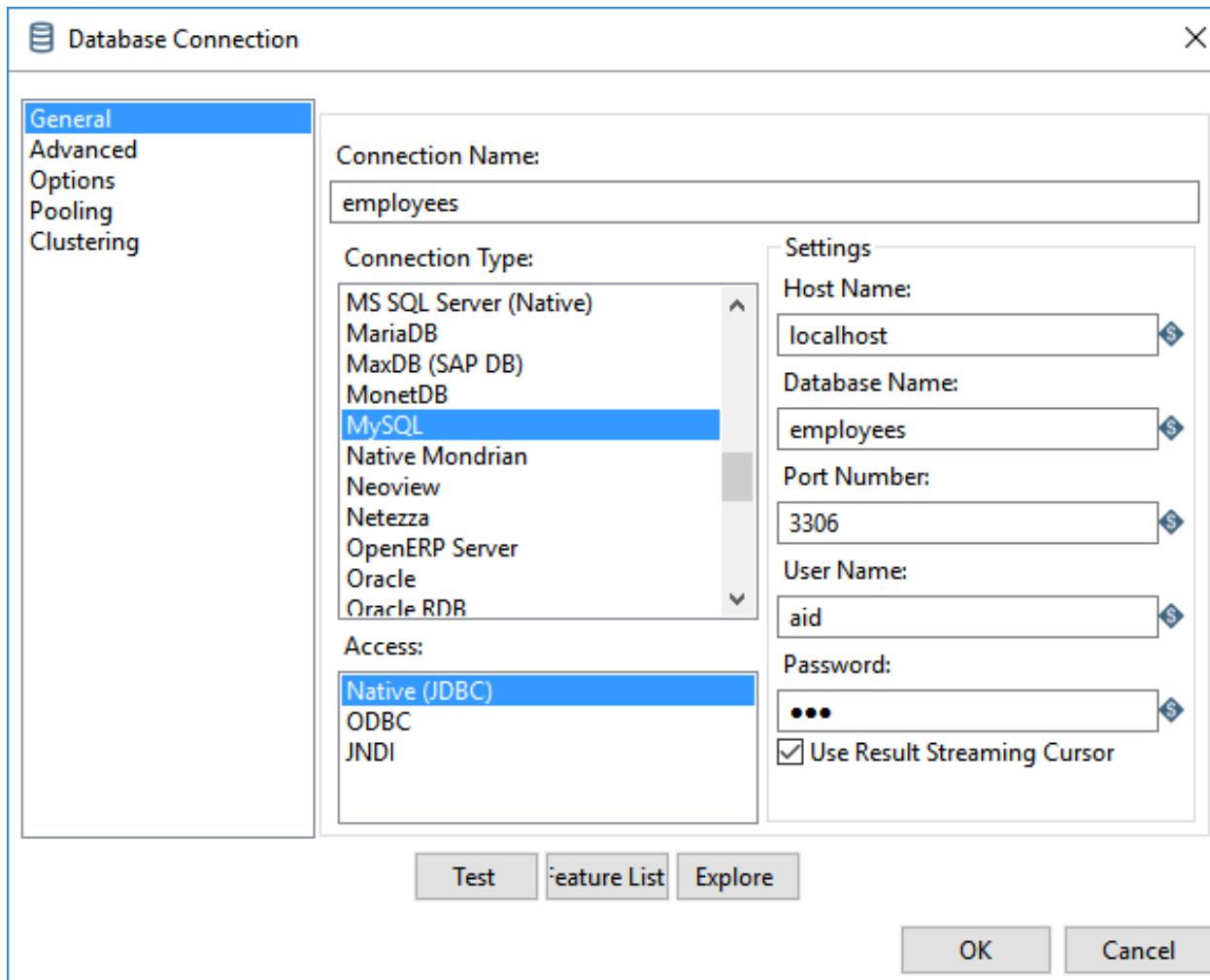
# Database connections



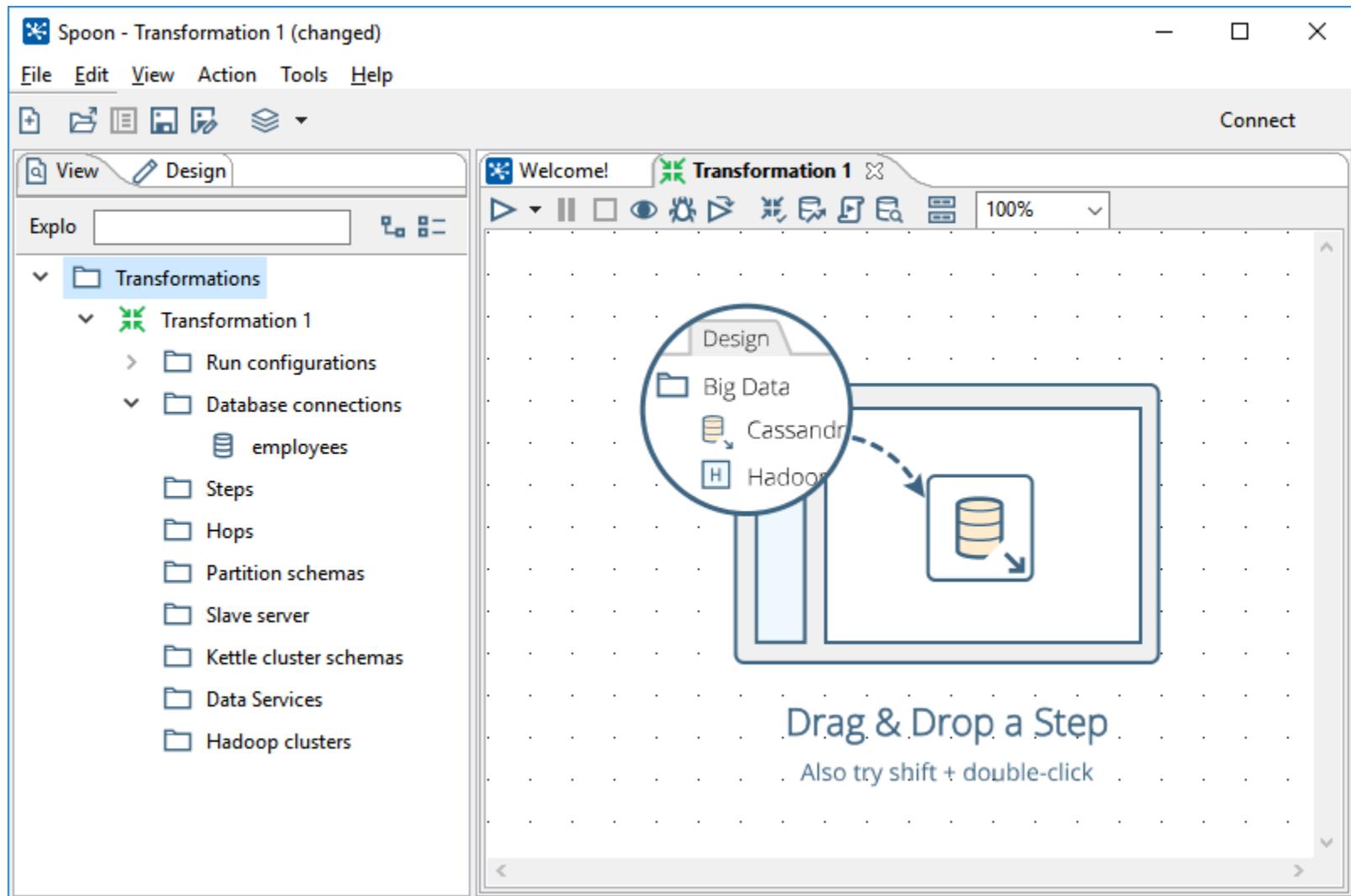
# New database connection



# New database connection

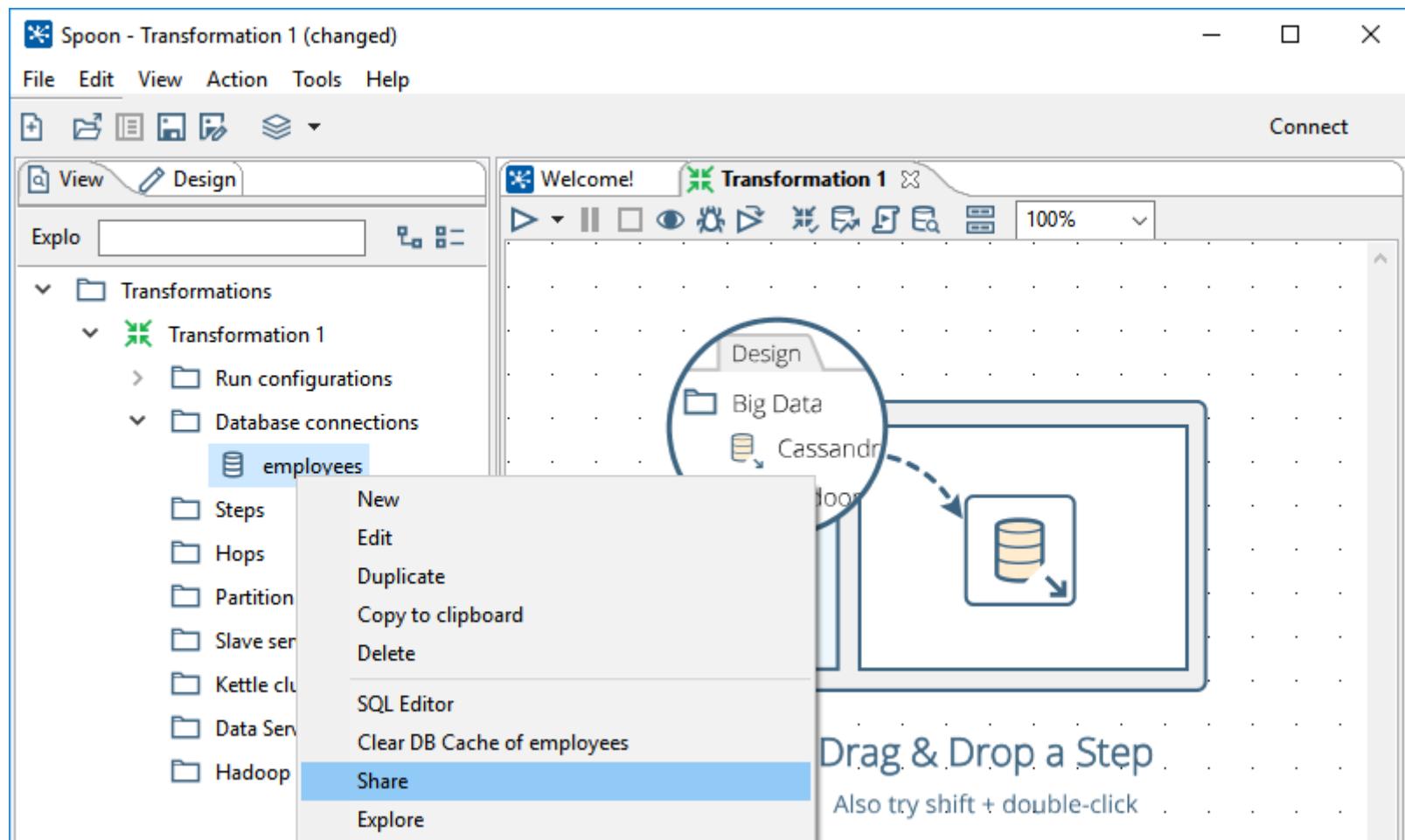


# New database connection

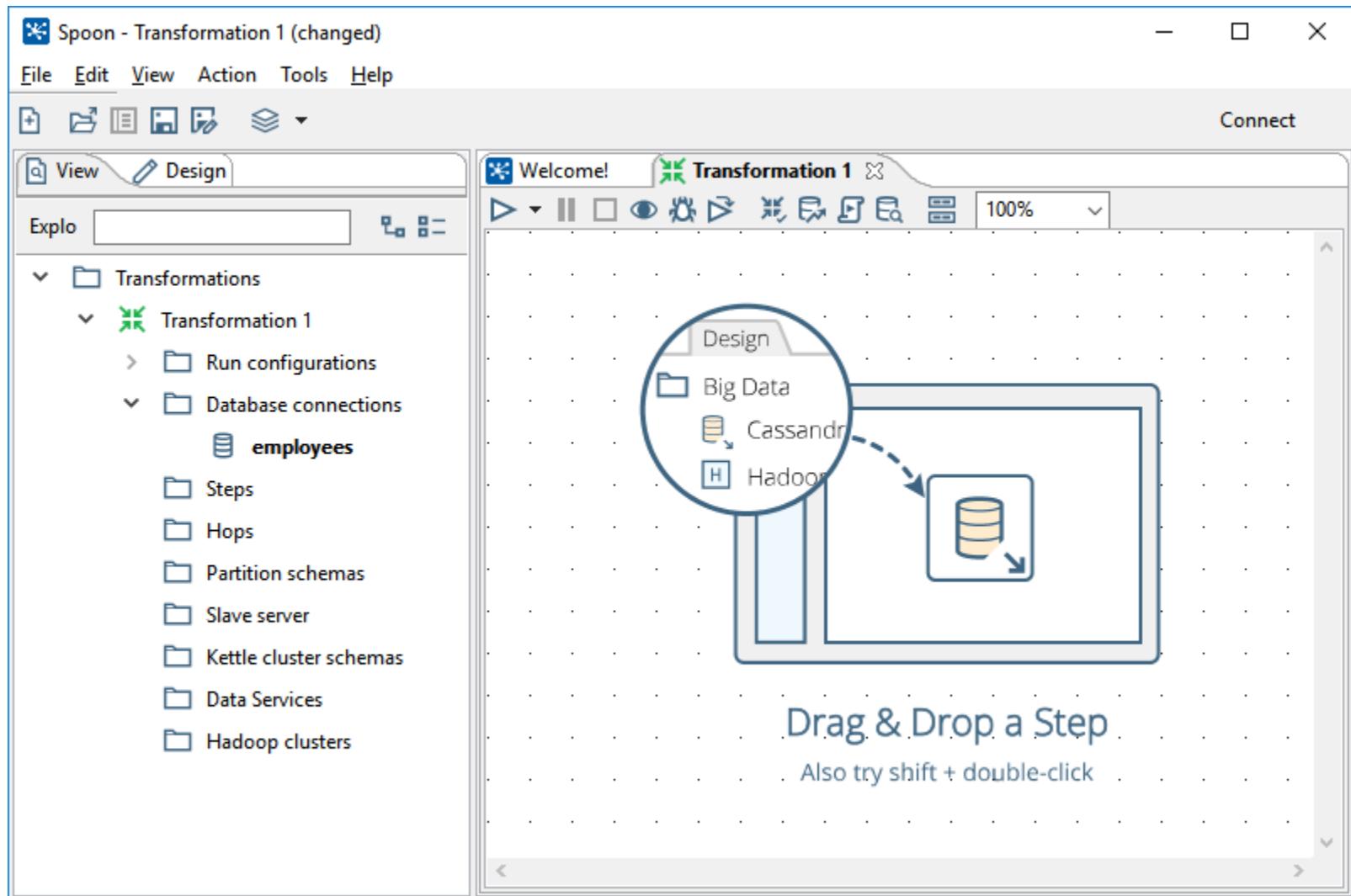


# New database connection

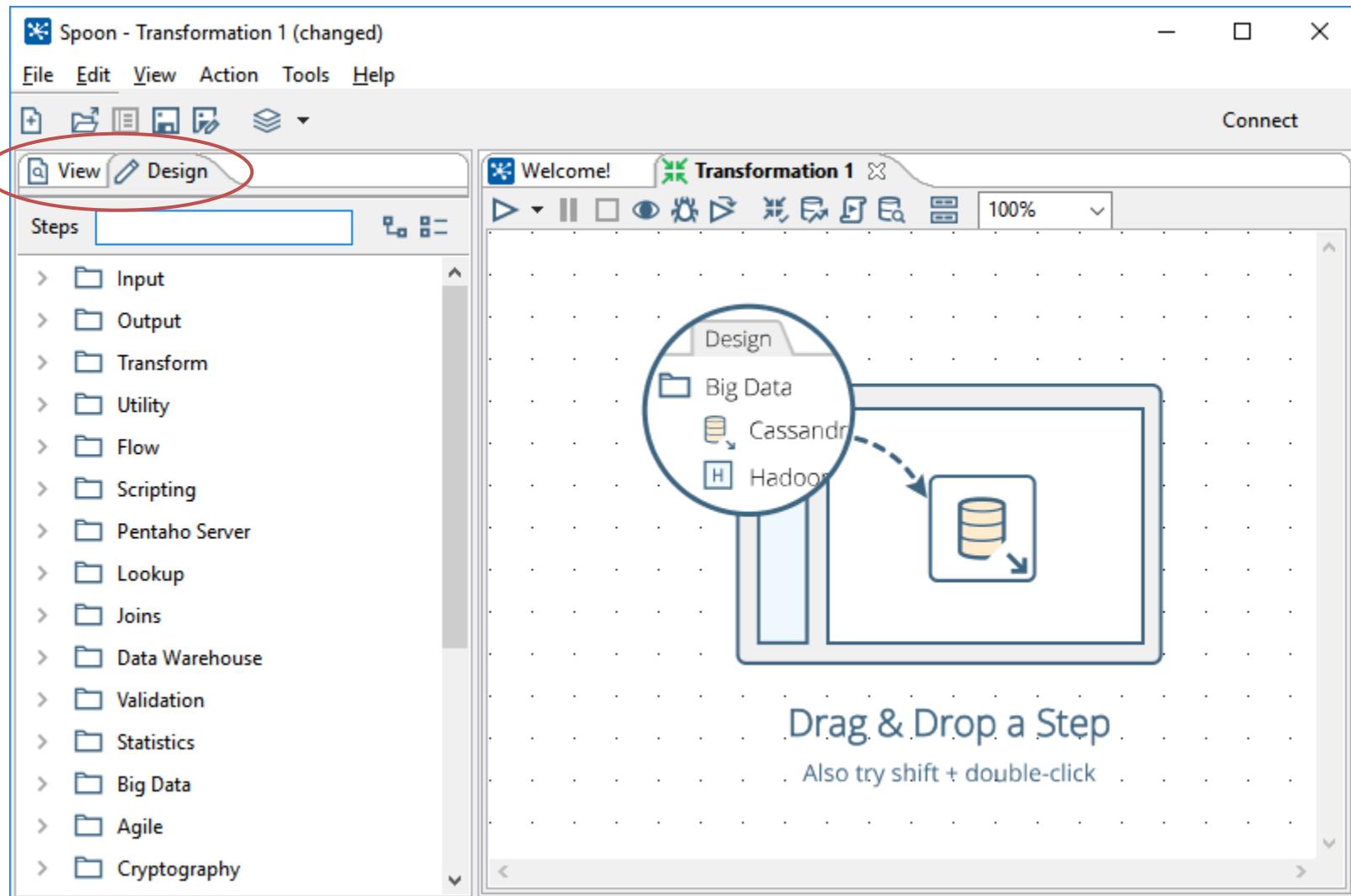
- Using the same connection in multiple transformations



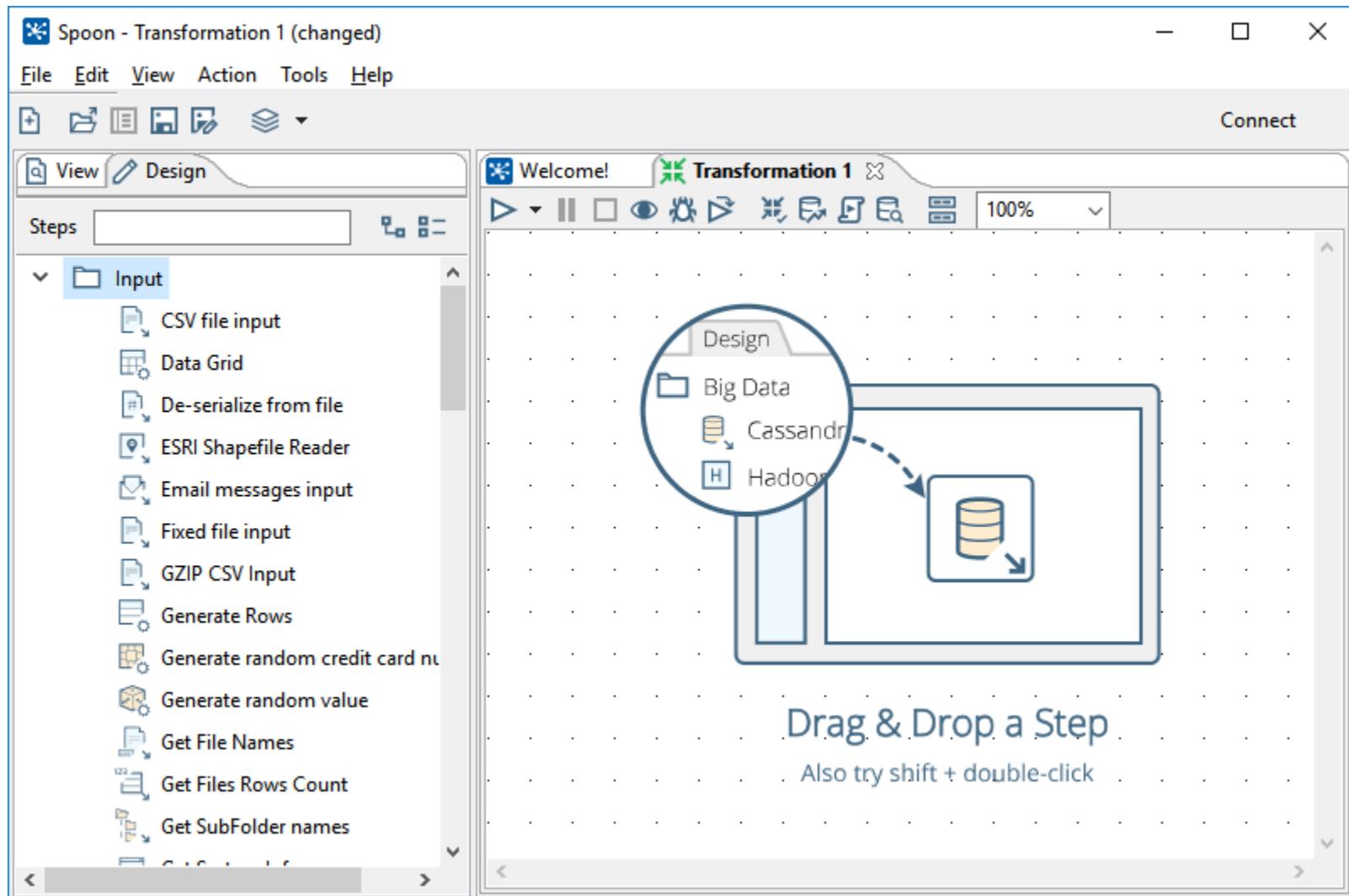
# New database connection



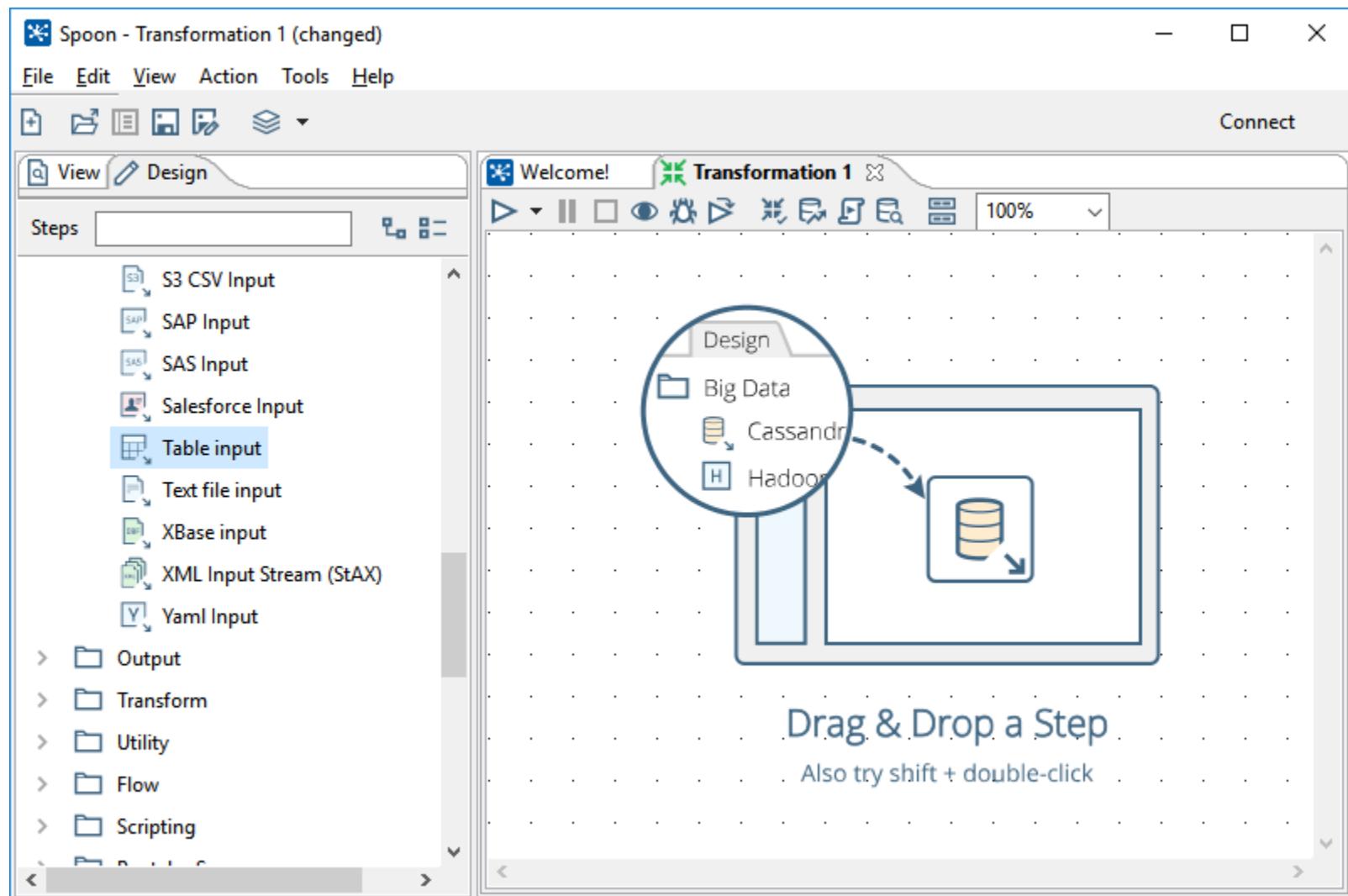
# Table input



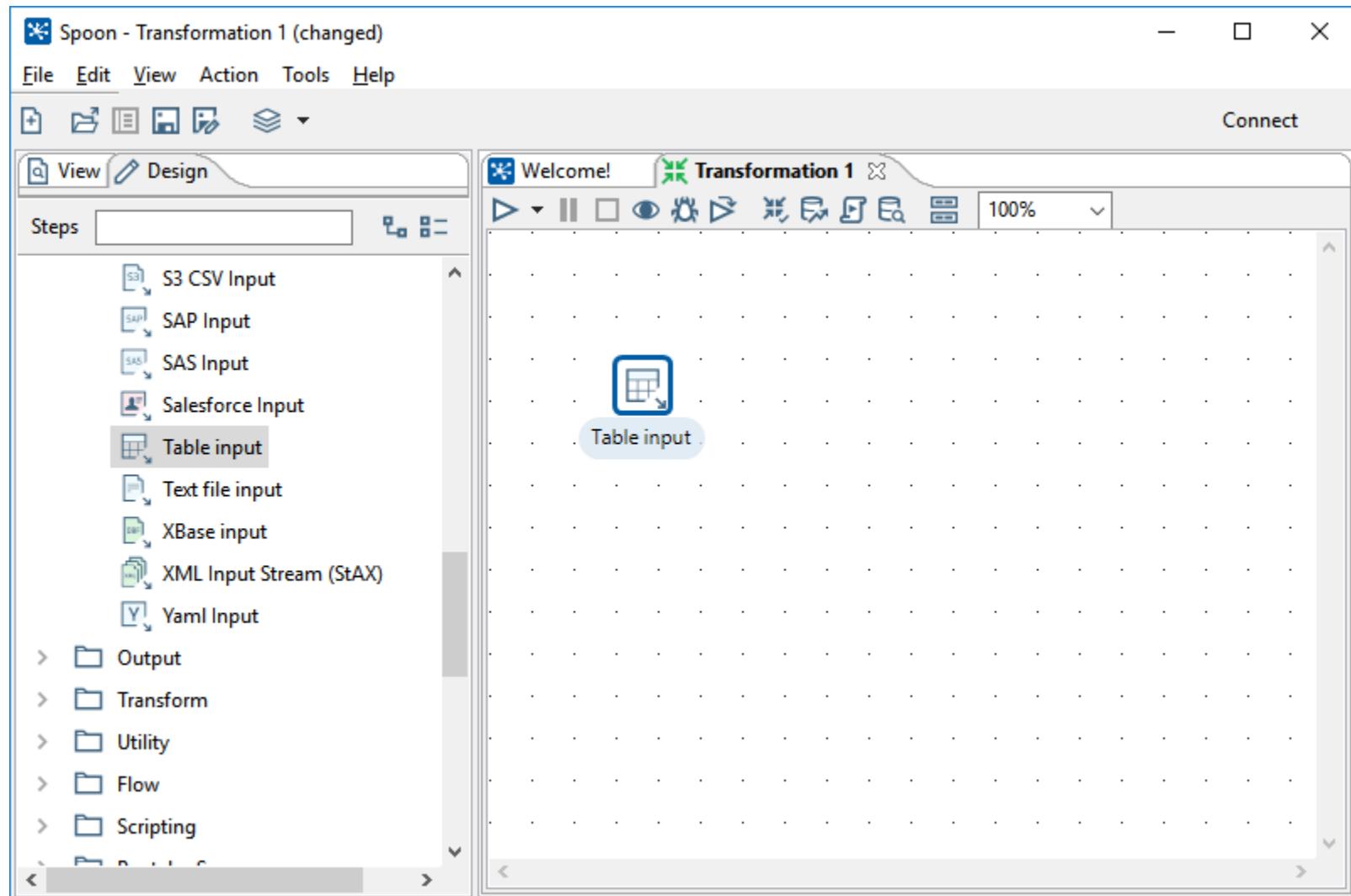
# Table input



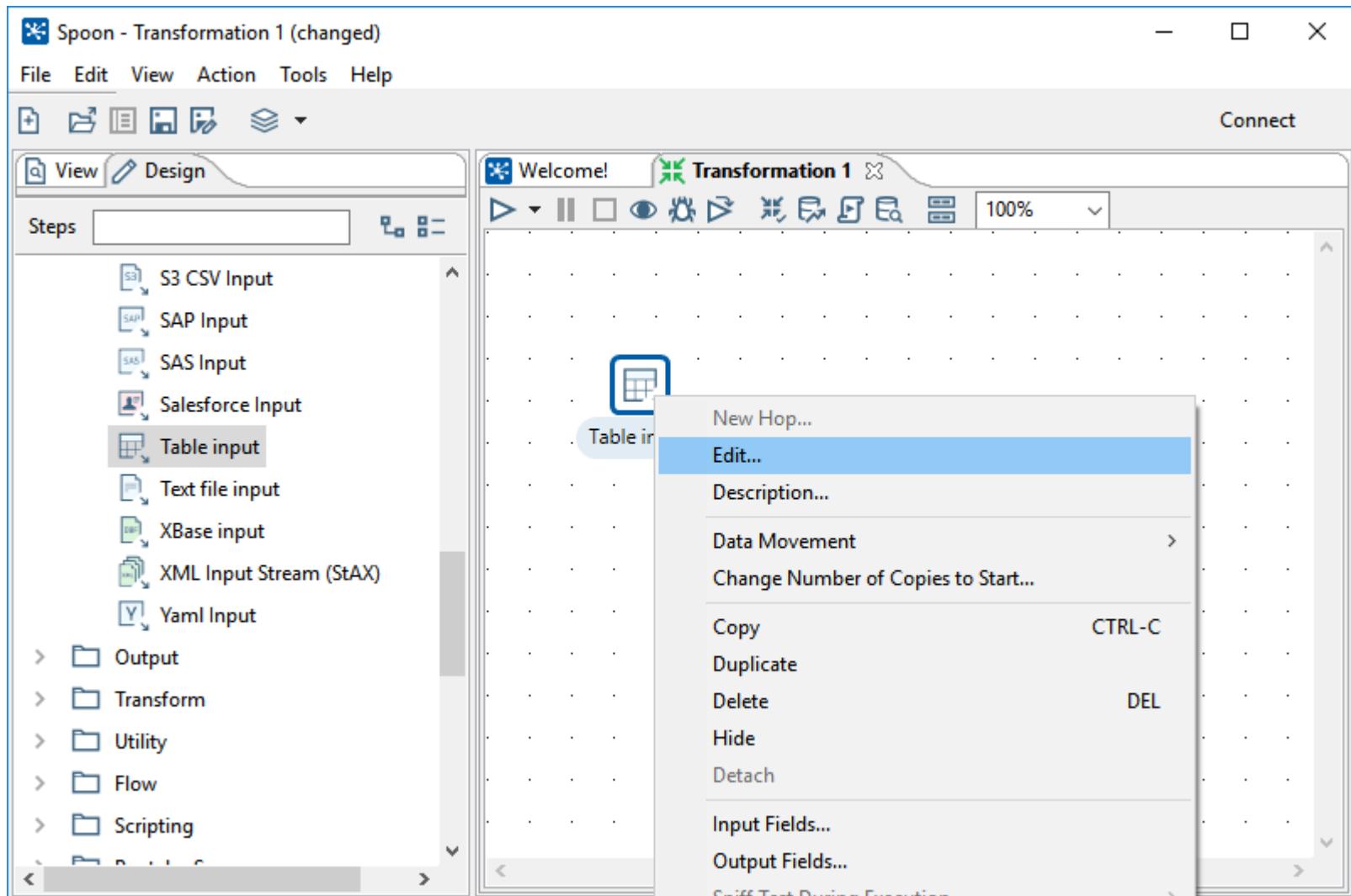
# Table input



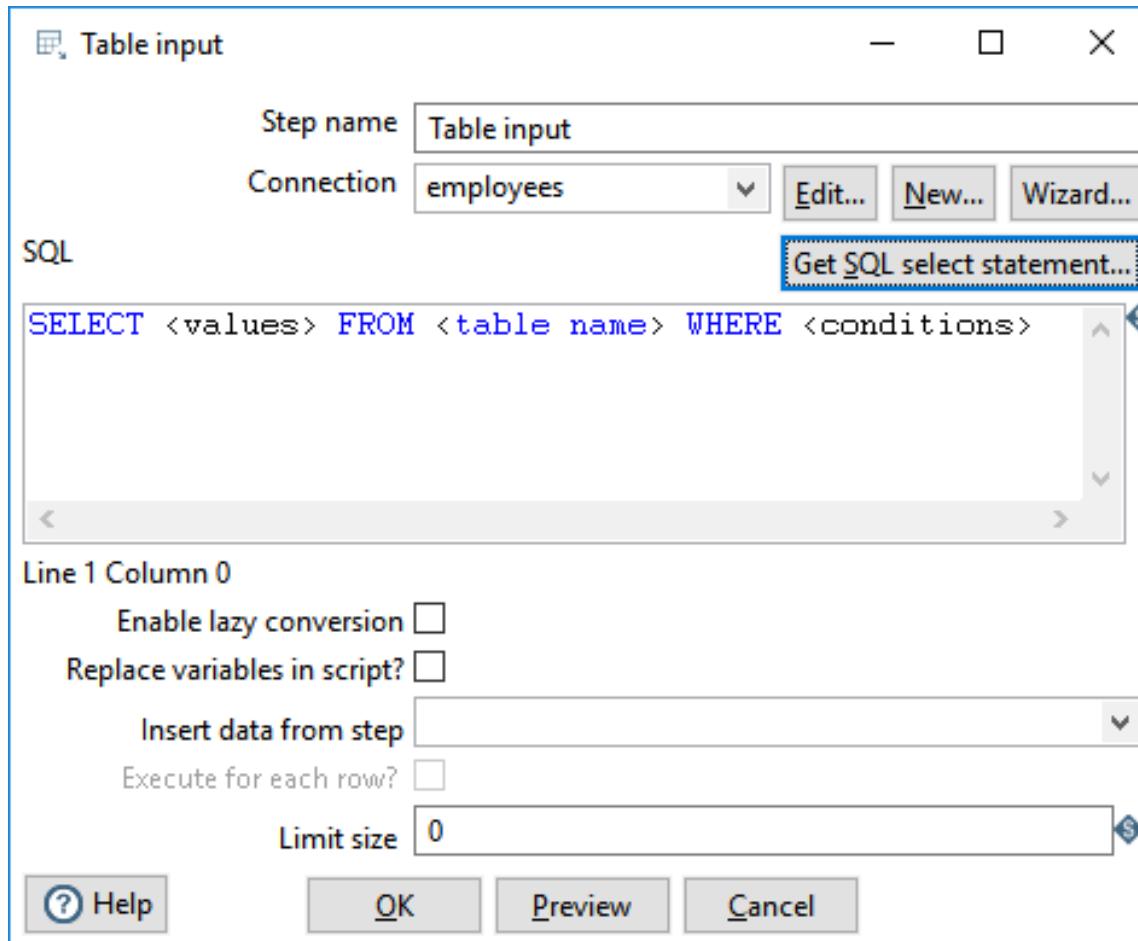
# Table input



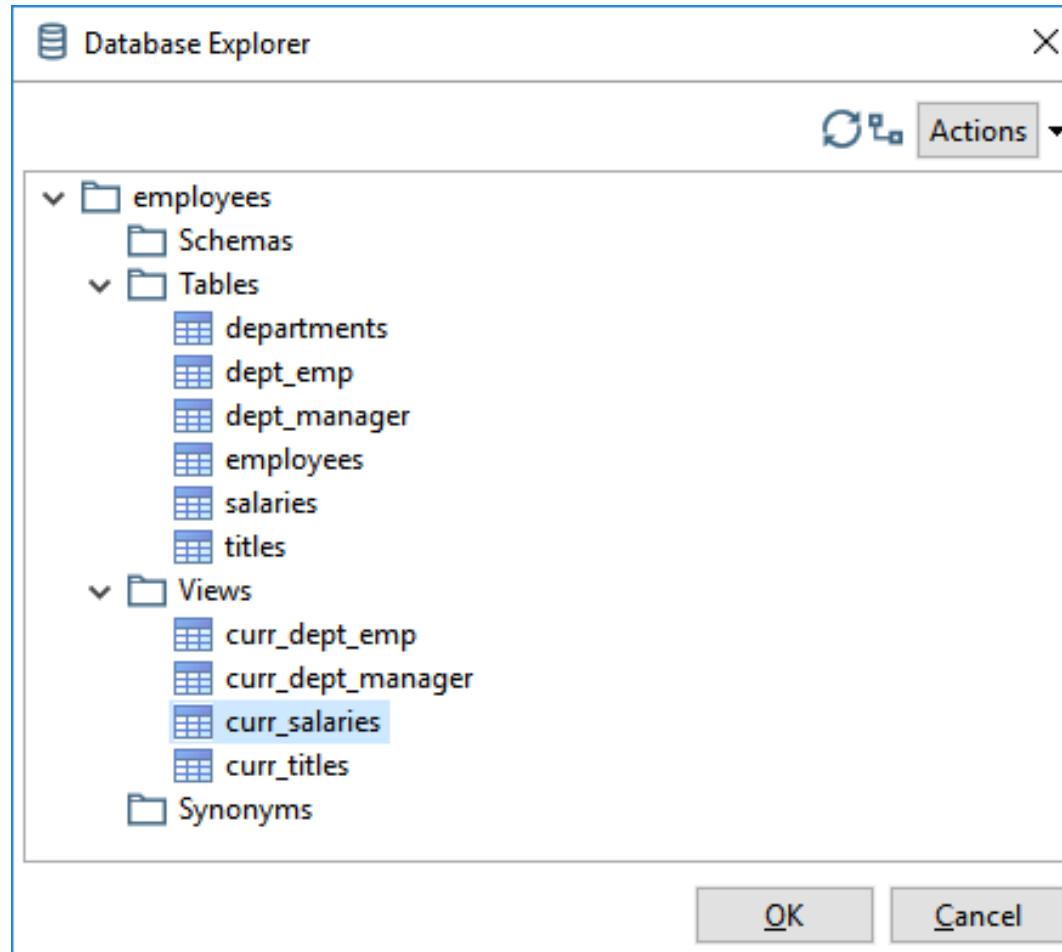
# Table input



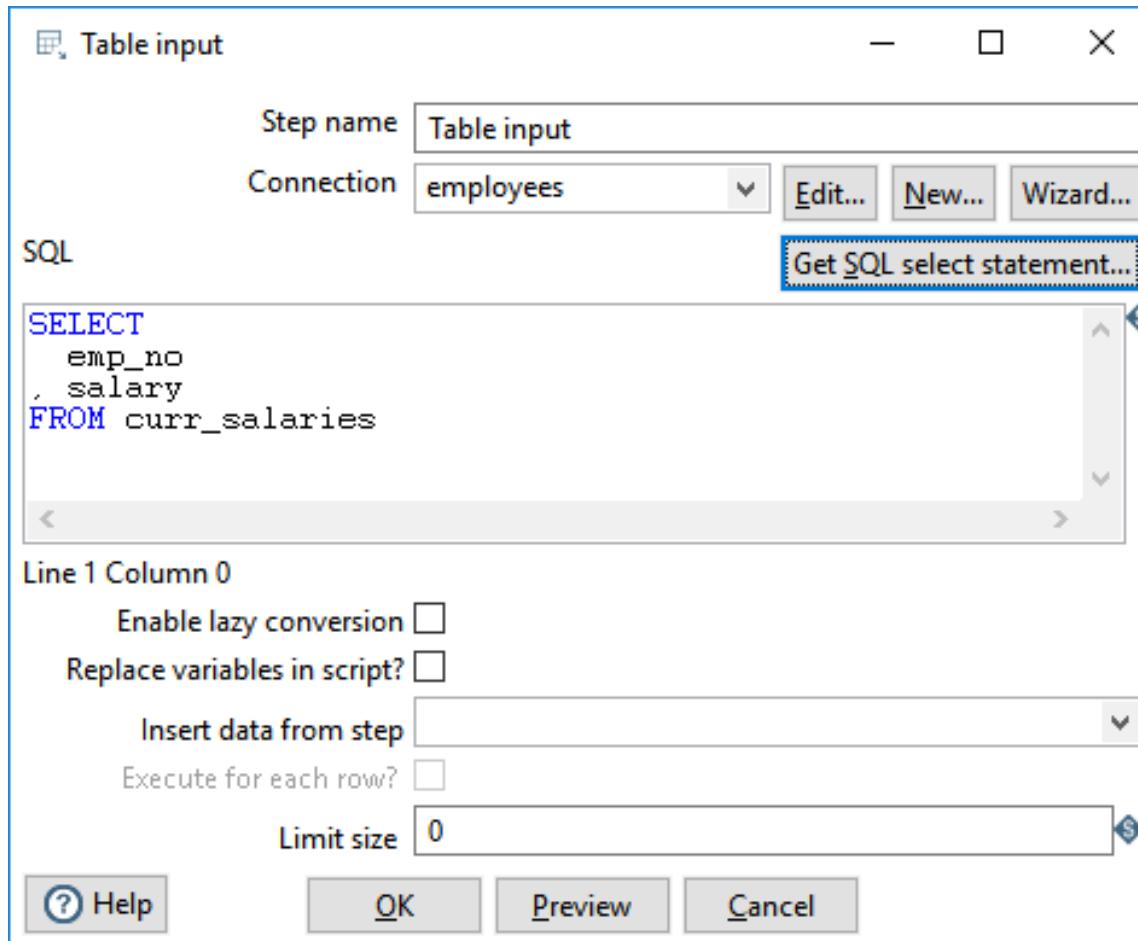
# Table input



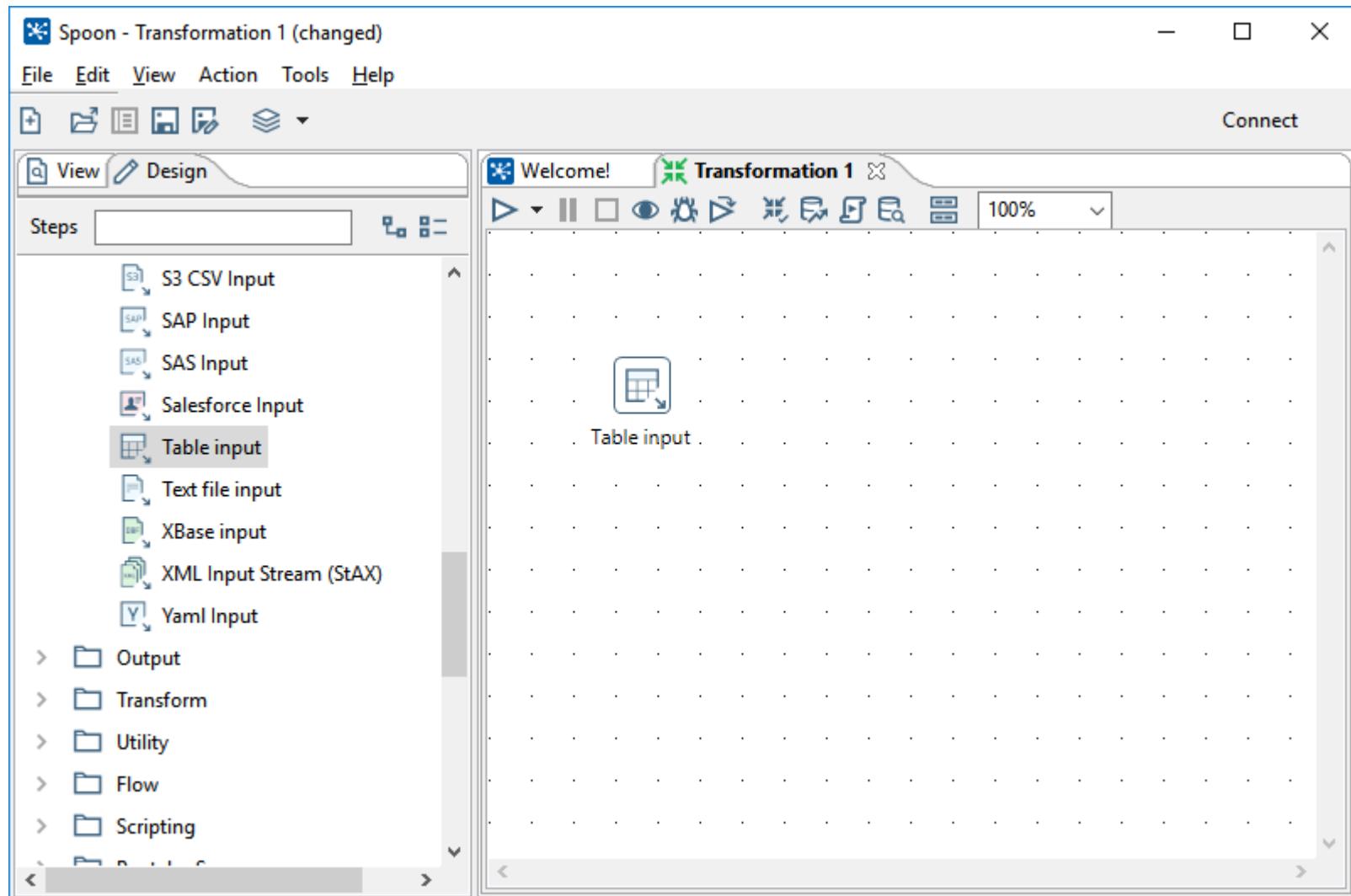
# Table input



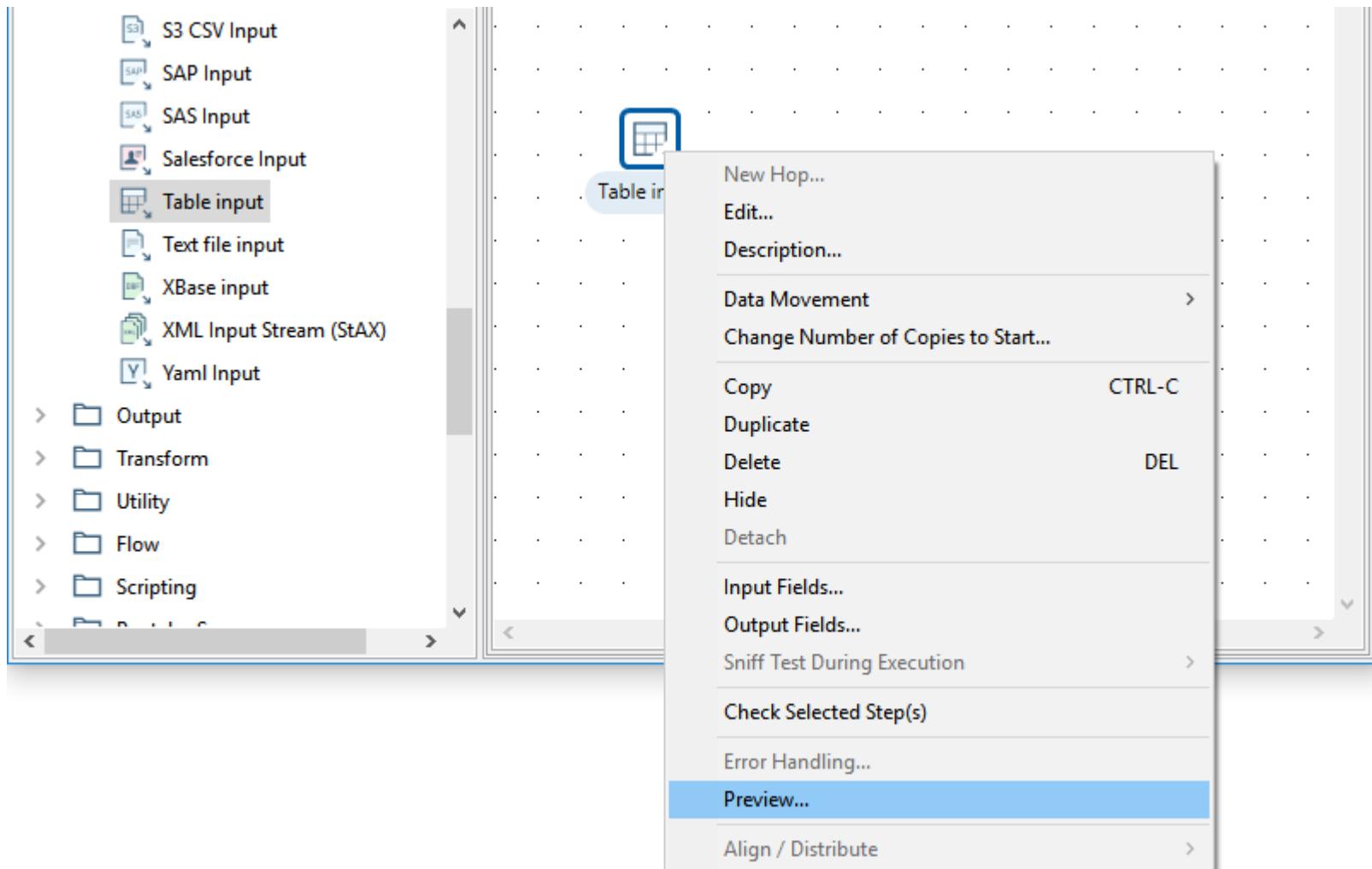
# Table input



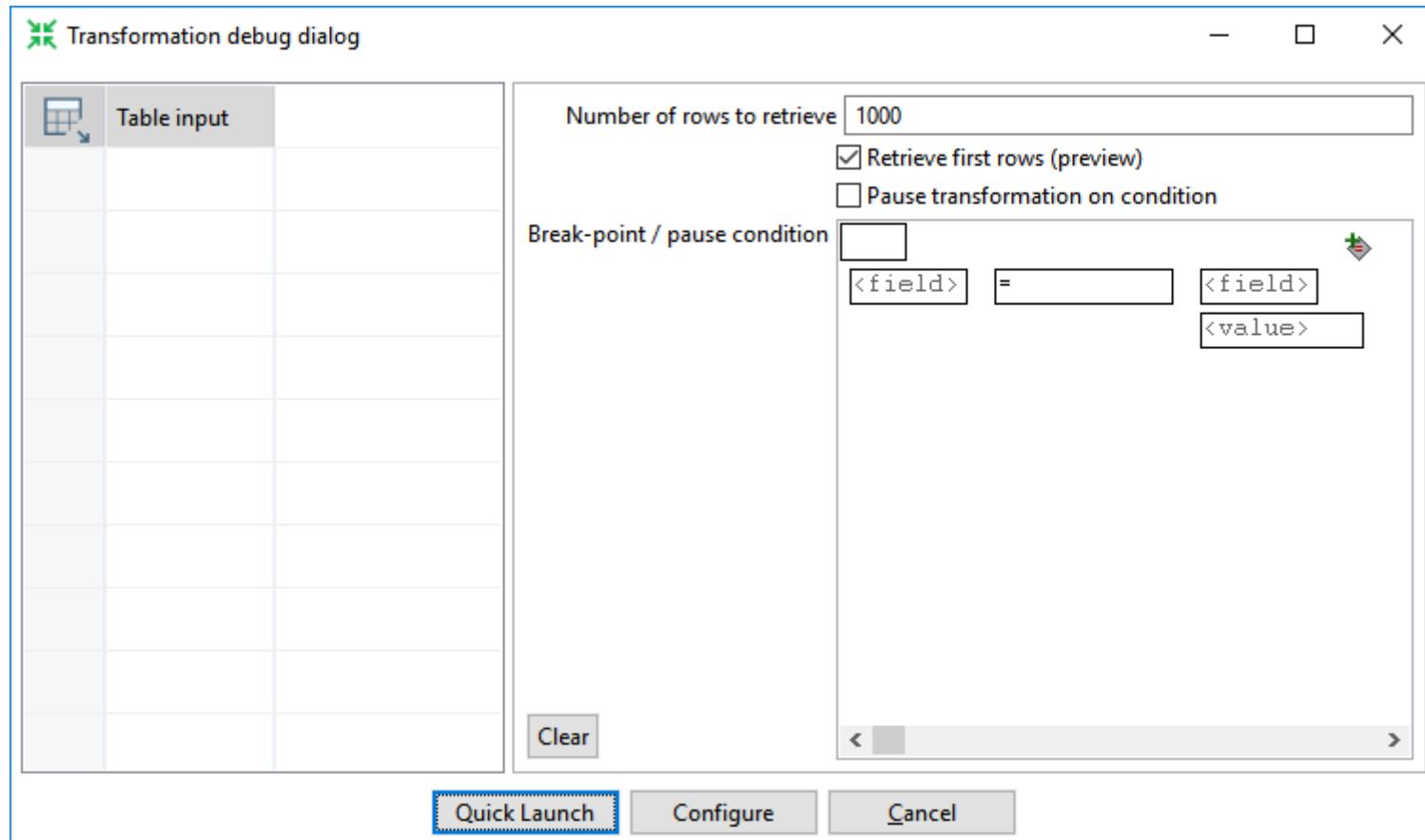
# Table input



# Table input



# Table input



# Table input

Examine preview data

Rows of step: Table input (252 rows)

#	emp_no	salary
1	10721	44812
2	11260	52435
3	11371	81461
4	11693	101179
5	13816	76104
6	14007	105453
7	14083	71350
8	14791	49249
9	17698	91443
10	17739	91836
11	17890	80046
12	18691	67677
13	19103	70313
14	19344	66406
15	19884	56851
16	19983	57499

[Close](#)

# Table input

The screenshot shows the Apache Nifi Spoon interface for a transformation named "Transformation 1".

**Left Panel (Steps):**

- S3 CSV Input
- SAP Input
- SAS Input
- Salesforce Input
- Table input** (highlighted)
- Text file input
- XBase input
- XML Input Stream (StAX)
- Yaml Input
- Output
- Transform
- Utility
- Flow
- Scripting

**Middle Panel (Transformation 1):**

- Icon: Table input
- Label: Table input

**Bottom Panel (Execution Results):**

#	Stepname	Copynr	Read	Written	Input	Output	Update
1	Table input	0	0	252	252	0	0

# Filter rows

Spoon - Transformation 1 (changed)

File Edit View Action Tools Help

Connect

View Design

Steps

- > Input
- > Output
- > Transform
- > Utility
- Flow**
- > Scripting
- > Pentaho Server
- > Lookup
- > Joins
- > Data Warehouse
- > Validation
- > Statistics
- > Big Data
- > Agile
- > Cryptography

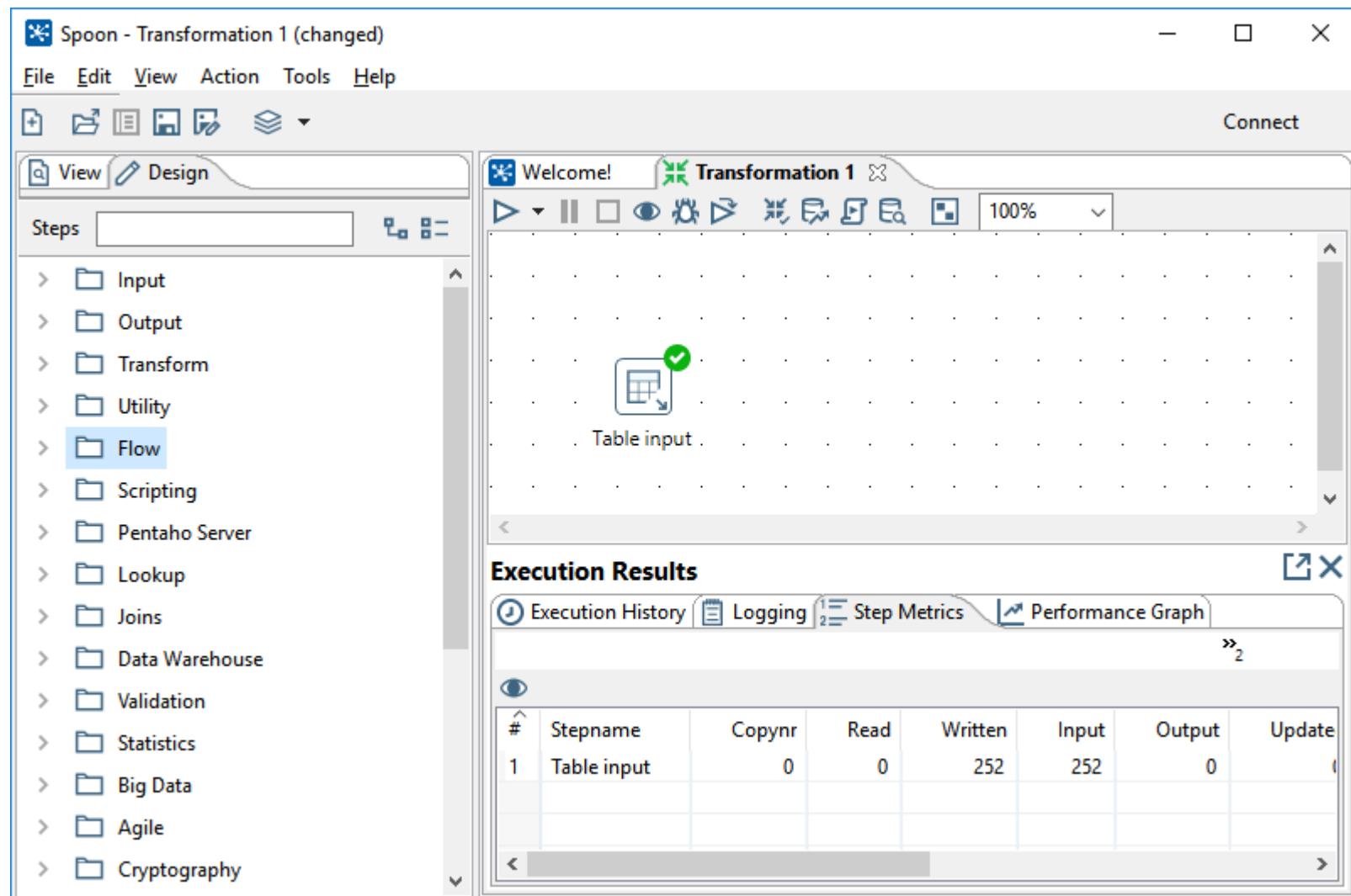
Welcome! Transformation 1

Table input

Execution Results

Execution History Logging Step Metrics Performance Graph

#	Stepname	Copynr	Read	Written	Input	Output	Update
1	Table input	0	0	252	252	0	0



# Filter rows

The screenshot shows the Apache Nifi Spoon interface for a transformation named "Transformation 1 (changed)".

**Left Panel (View):**

- Shows the "Design" tab selected.
- Contains a "Steps" search bar and a "Flow" tree view.
- The "Flow" tree is expanded, showing various step types:
  - Abort
  - Annotate Stream
  - Append streams
  - Block this step until steps finish
  - Blocking Step
  - Detect empty stream
  - Dummy (do nothing)
  - ETL Metadata Injection
  - Filter rows** (highlighted in blue)
  - Identify last row in a stream
  - Java Filter
  - Job Executor
  - Prioritize streams

**Middle Panel (Transformation 1):**

- Shows a single "Table input" step in the canvas.
- Step status: **Success** (green checkmark).
- Execution Results panel below the canvas:

  - Execution History tab is active.
  - Logging tab is present.
  - Step Metrics tab is active, showing the following data:

#	Stepname	Copynr	Read	Written	Input	Output	Update
1	Table input	0	0	252	252	0	0

# Filter rows

The screenshot shows the Apache Nifi Spoon interface for a transformation named "Transformation 1".

**Left Panel (View):**

- Shows the "Flow" section with various step icons and names.
- The "Filter rows" step is highlighted with a yellow border.

**Middle Panel (Transformation 1):**

- Shows the "Table input" and "Filter rows" steps connected in a flow.
- The "Filter rows" step is selected, indicated by a blue selection bar.

**Bottom Panel (Execution Results):**

- Shows the "Step Metrics" tab.
- A table displays the following metrics for the "Table input" step:

#	Stepname	Copynr	Read	Written	Input	Output	Update
1	Table input	0	0	252	252	0	0

# Filter rows

Spoon - Transformation 1 (changed)

File Edit View Action Tools Help

View Design

Steps

Flow

- Abort
- Annotate Stream
- Append streams
- Block this step until steps finish
- Blocking Step
- Detect empty stream
- Dummy (do nothing)
- ETL Metadata Injection
- Filter rows**
- Identify last row in a stream
- Java Filter
- Job Executor
- Prioritize streams

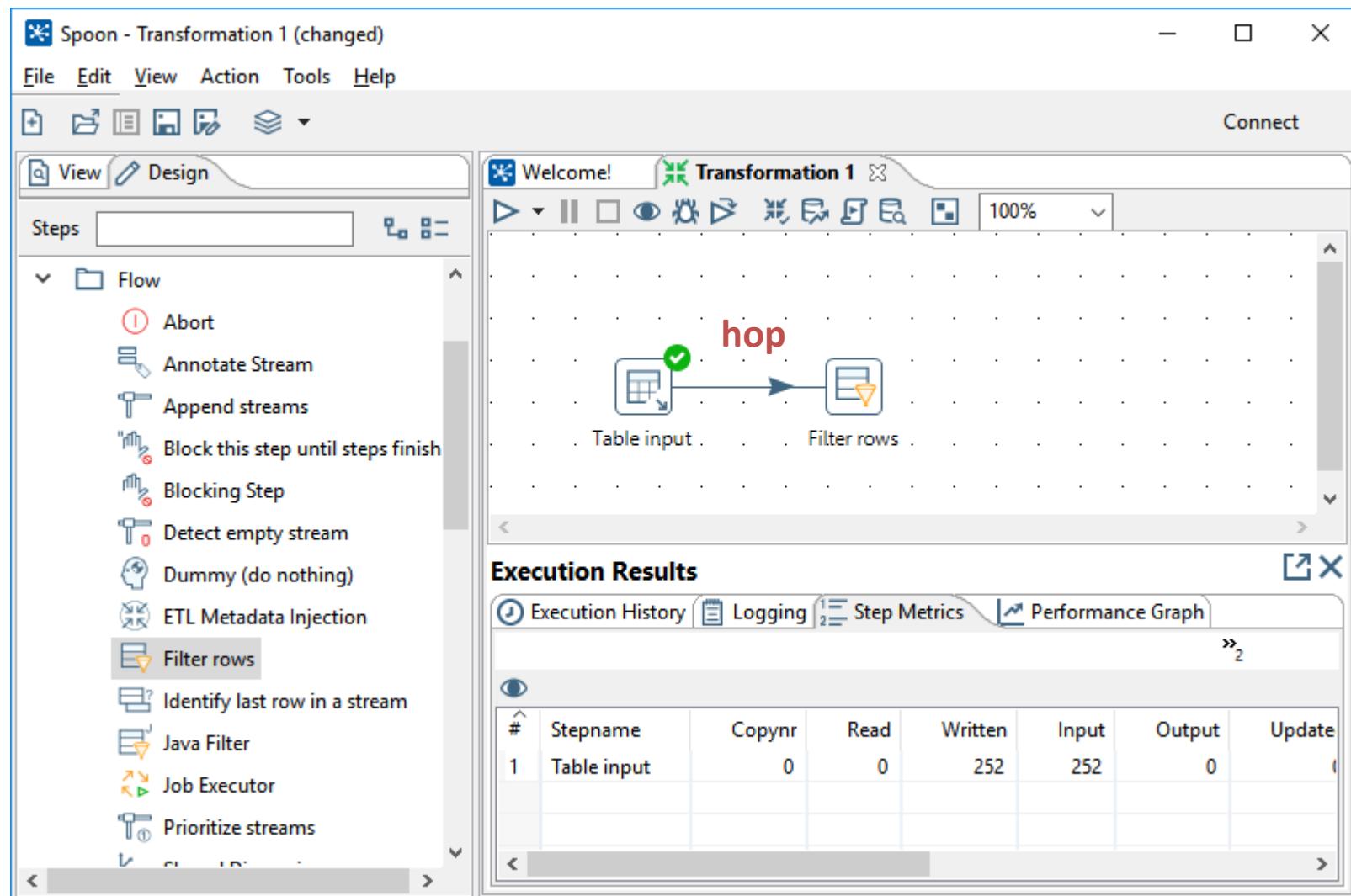
Welcome! Transformation 1

Table input hop Filter rows

Execution Results

Execution History Logging Step Metrics Performance Graph

#	Stepname	Copynr	Read	Written	Input	Output	Update
1	Table input	0	0	252	252	0	0



# Filter rows

The screenshot shows the Apache Nifi Spoon interface for a transformation named "Transformation 1". The left panel displays a list of available steps under the "Flow" category, including "Abort", "Annotate Stream", "Append streams", "Block this step until steps finish", "Blocking Step", "Detect empty stream", "Dummy (do nothing)", "ETL Metadata Injection", "Filter rows" (which is highlighted), "Identify last row in a stream", "Java Filter", "Job Executor", and "Prioritize streams". The main workspace shows a flow starting with a "Table input" step followed by a "Filter" step. A context menu is open over the "Filter" step, listing options such as "New Hop...", "Edit..." (which is selected and highlighted in blue), "Description...", "Data Movement", "Change Number of Copies to Start...", "Copy", "Duplicate", "Delete" (with a keyboard shortcut "CTRL-D" shown), "Hide", "Detach", "Input Fields...", "Output Fields...", and "Sniff Test During Execution".

Spoon - Transformation 1 (changed)

File Edit View Action Tools Help

View Design

Steps

Flow

- Abort
- Annotate Stream
- Append streams
- Block this step until steps finish
- Blocking Step
- Detect empty stream
- Dummy (do nothing)
- ETL Metadata Injection
- Filter rows**
- Identify last row in a stream
- Java Filter
- Job Executor
- Prioritize streams

Welcome! Transformation 1

Table input Filter

New Hop...

**Edit...**

Description...

Data Movement

Change Number of Copies to Start...

Copy

Duplicate

Delete **CTRL-D**

Hide

Detach

Input Fields...

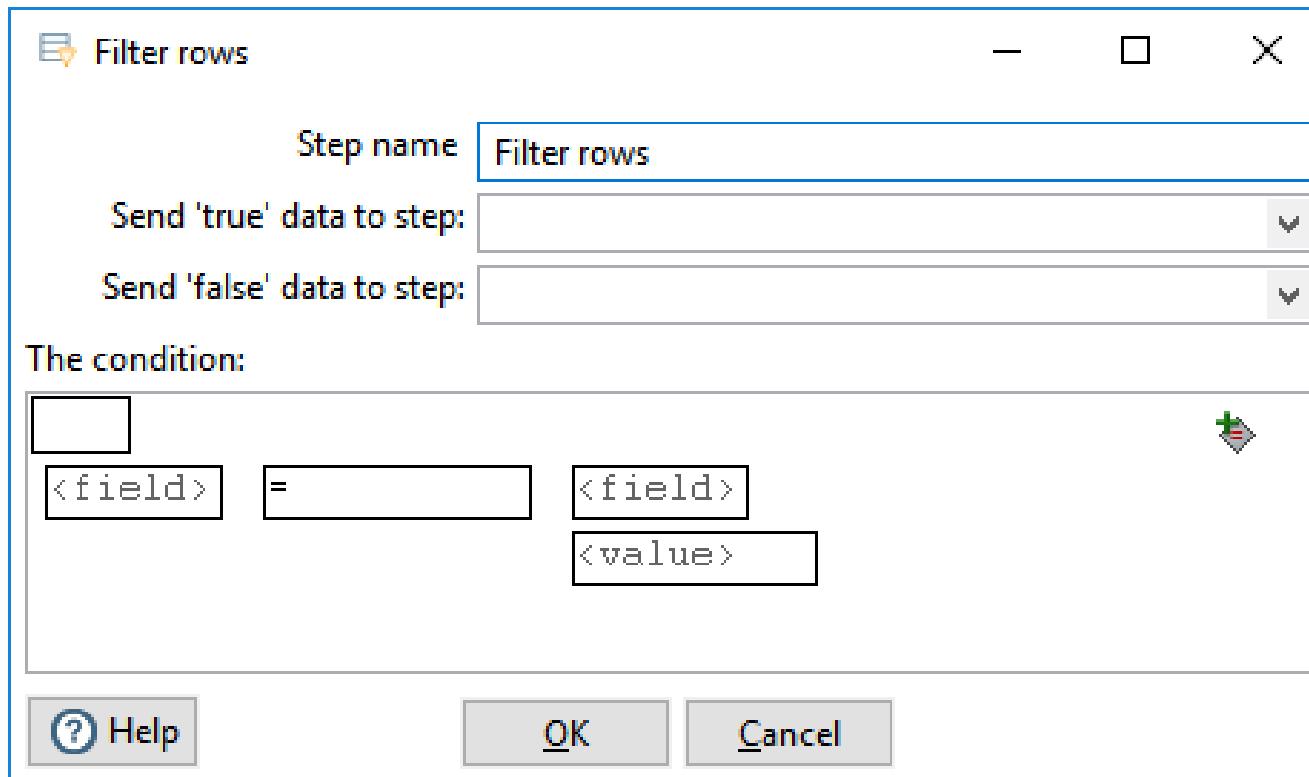
Output Fields...

Sniff Test During Execution

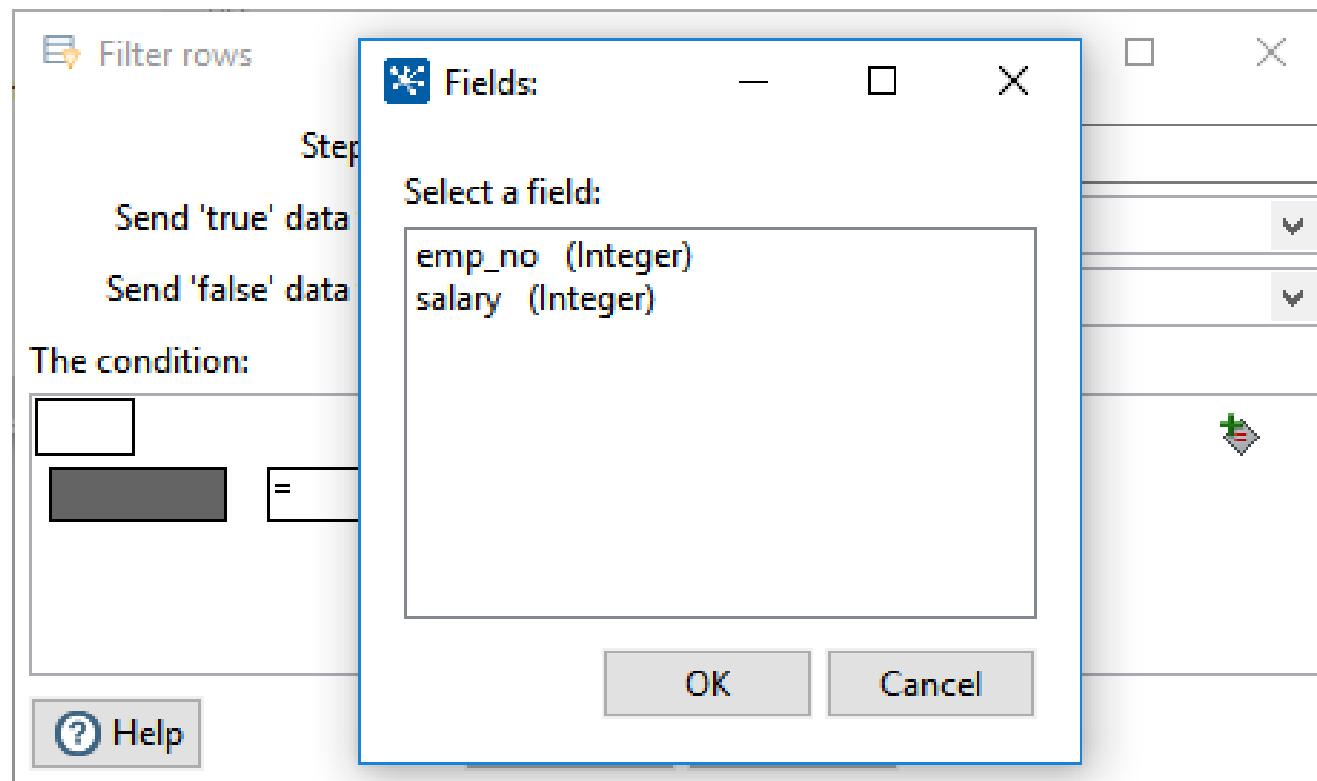
Execution Results

#	Stepname	Copynr	R
1	Table input	0	

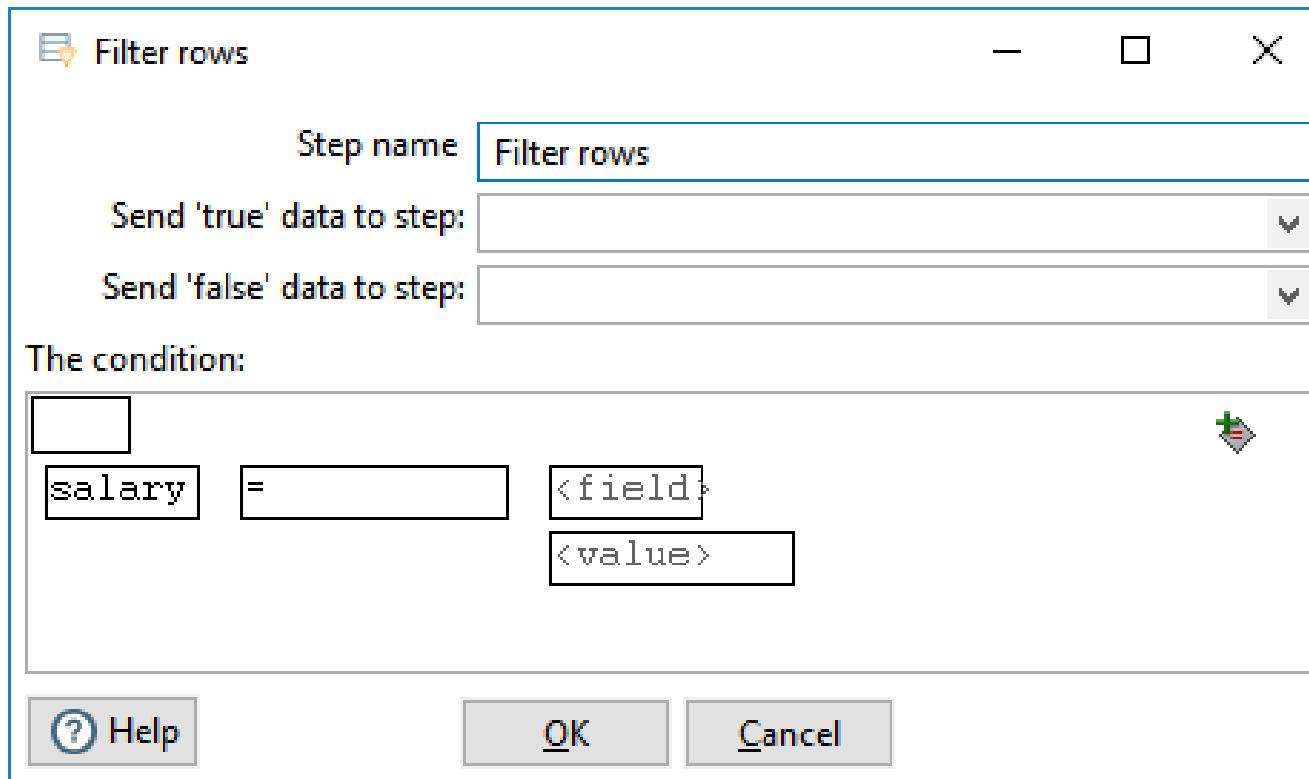
# Filter rows



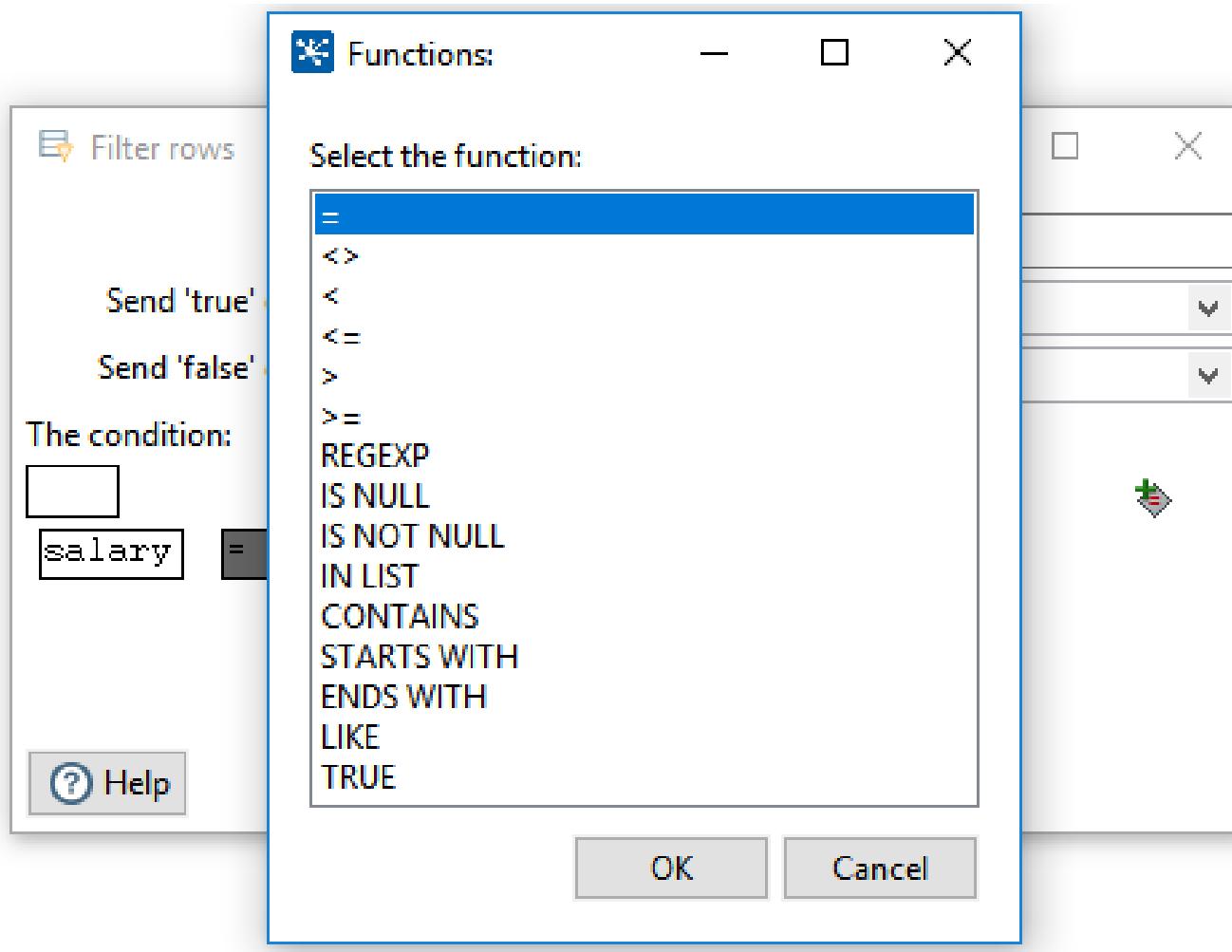
# Filter rows



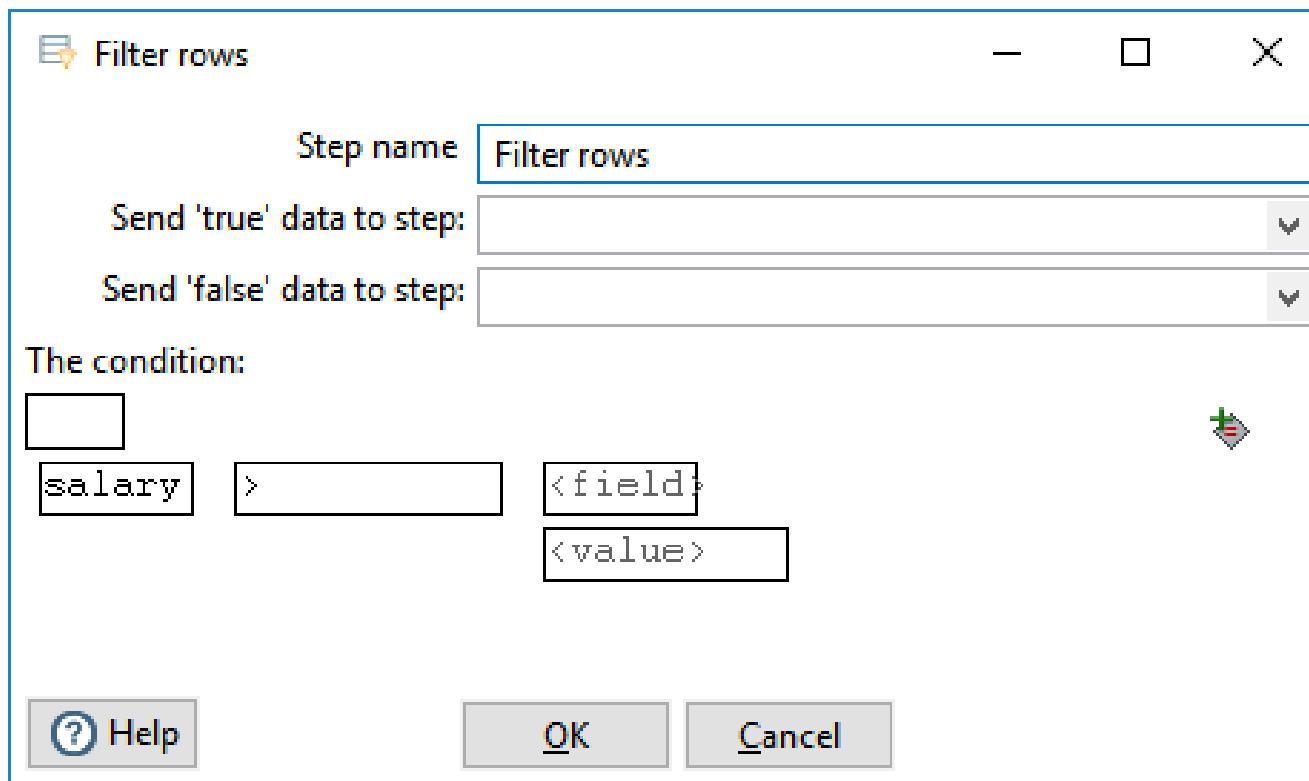
# Filter rows



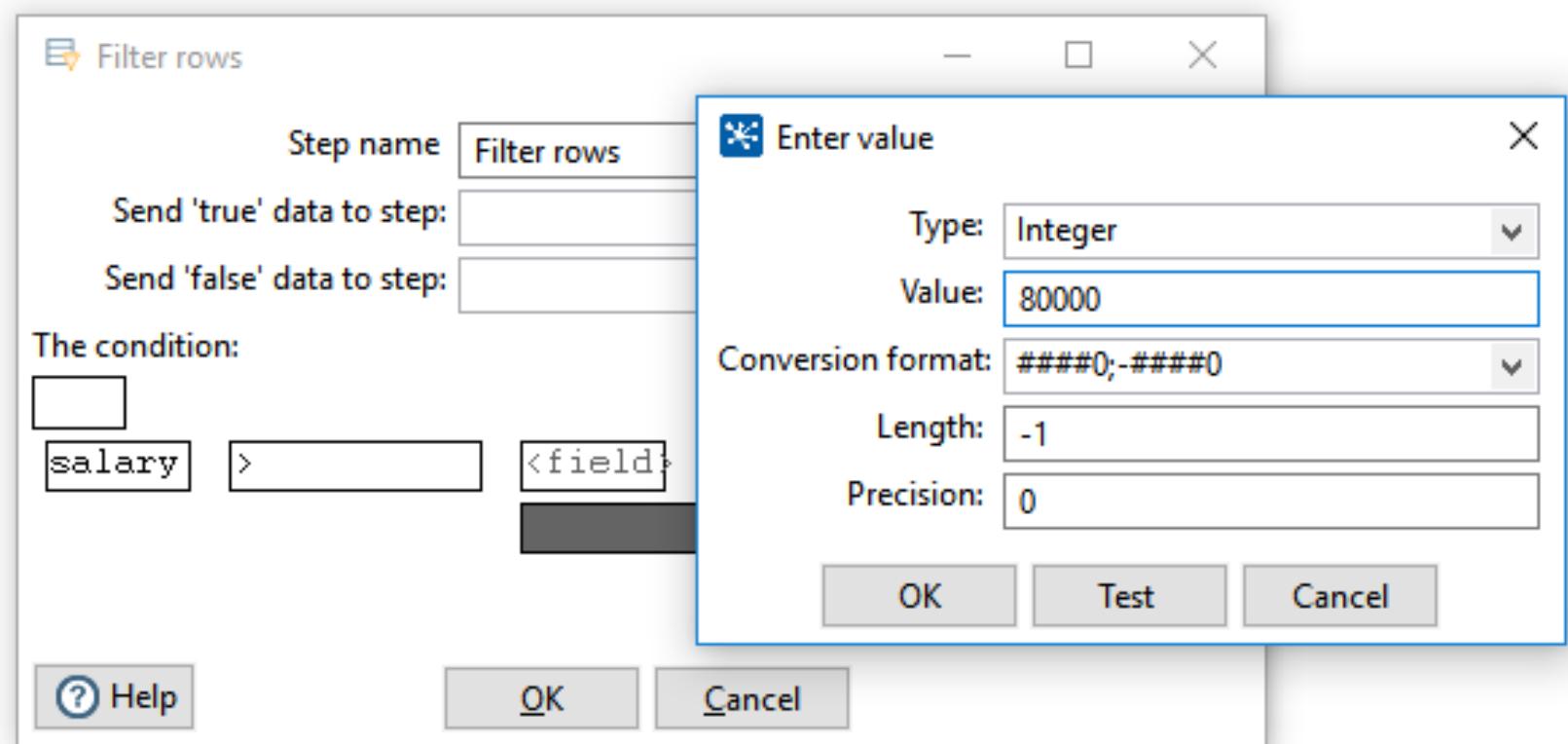
# Filter rows



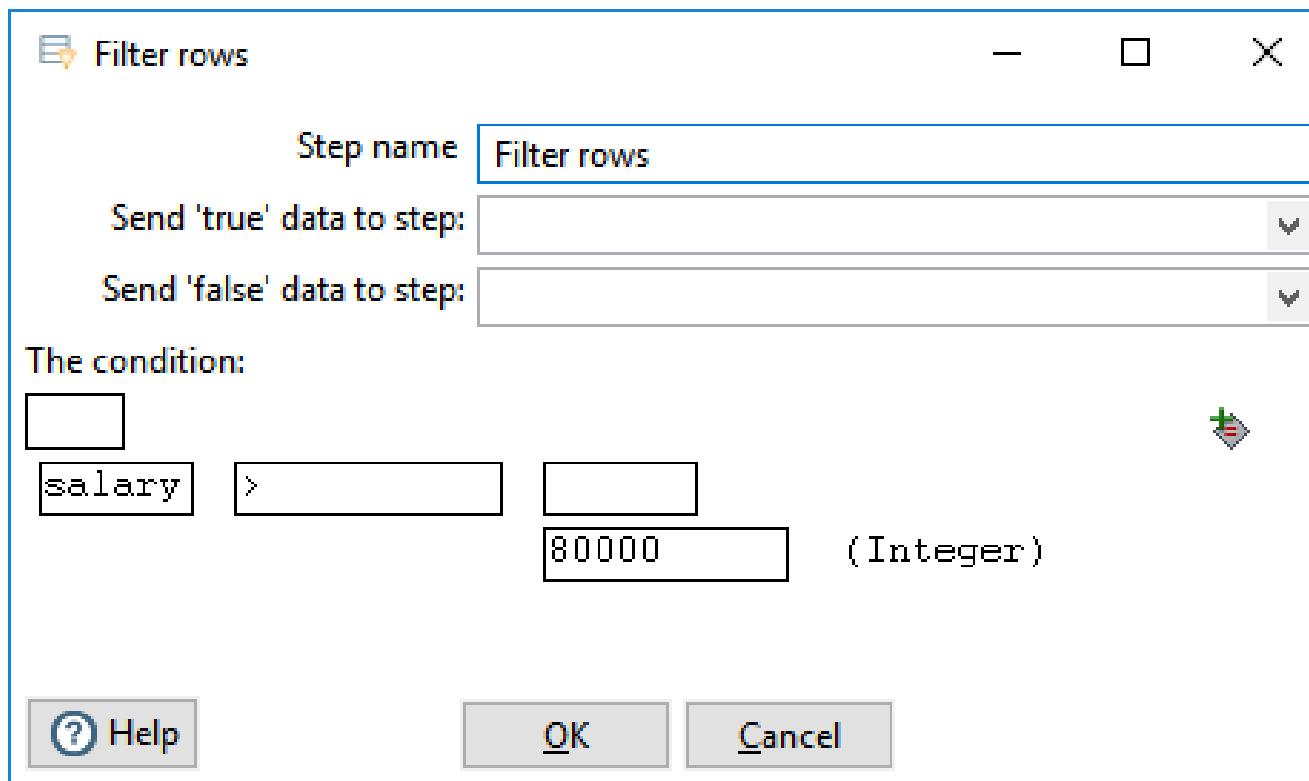
# Filter rows



# Filter rows



# Filter rows



# Filter rows

Spoon - Transformation 1 (changed)

File Edit View Action Tools Help

View Design

Steps

Flow

- Abort
- Annotate Stream
- Append streams
- Block this step until steps finish
- Blocking Step
- Detect empty stream
- Dummy (do nothing)
- ETL Metadata Injection
- Filter rows
- Identify last row in a stream
- Java Filter
- Job Executor
- Prioritize streams

Welcome! Transformation 1

Table input Filter rows

Execution Results

Execution History Logging Step Metrics Performance Graph

#	Stepname	Copynr	Read	Written	Input	Output	Update
1	Table input	0	0	252	252	0	0

The screenshot shows the Apache Nifi Spoon interface. On the left, the 'Flow' panel lists various step types, with 'Filter rows' selected. In the main workspace, a simple flow is visible: a 'Table input' step is connected to a 'Filter rows' step. The 'Execution Results' panel at the bottom shows a single step named 'Table input' with zero reads and writes, and 252 inputs and outputs. The 'Step Metrics' tab is active.

# Filter rows

The screenshot shows the Talend Data Integration environment. On the left, a sidebar titled 'Flow' lists various step types, with 'Filter rows' currently selected. The main workspace displays a flow diagram with a 'Table input' step followed by a 'Filter' step. A context menu is open over the 'Filter' step, listing options such as 'New Hop...', 'Edit...', 'Description...', 'Data Movement', 'Copy' (with a keyboard shortcut of 'CTRL-C'), 'Duplicate', 'Delete' (with a keyboard shortcut of 'DEL'), 'Hide', 'Detach', 'Input Fields...', 'Output Fields...', 'Sniff Test During Execution', 'Check Selected Step(s)', 'Error Handling...', 'Preview...' (which is highlighted in blue), and 'Align / Distribute'. Below the flow diagram, an 'Execution Results' panel shows a table with one row: # Stepname Copynr 1 Table input 0.

New Hop...

Edit...

Description...

Data Movement

Change Number of Copies to Start...

Copy CTRL-C

Duplicate

Delete DEL

Hide

Detach

Input Fields...

Output Fields...

Sniff Test During Execution

Check Selected Step(s)

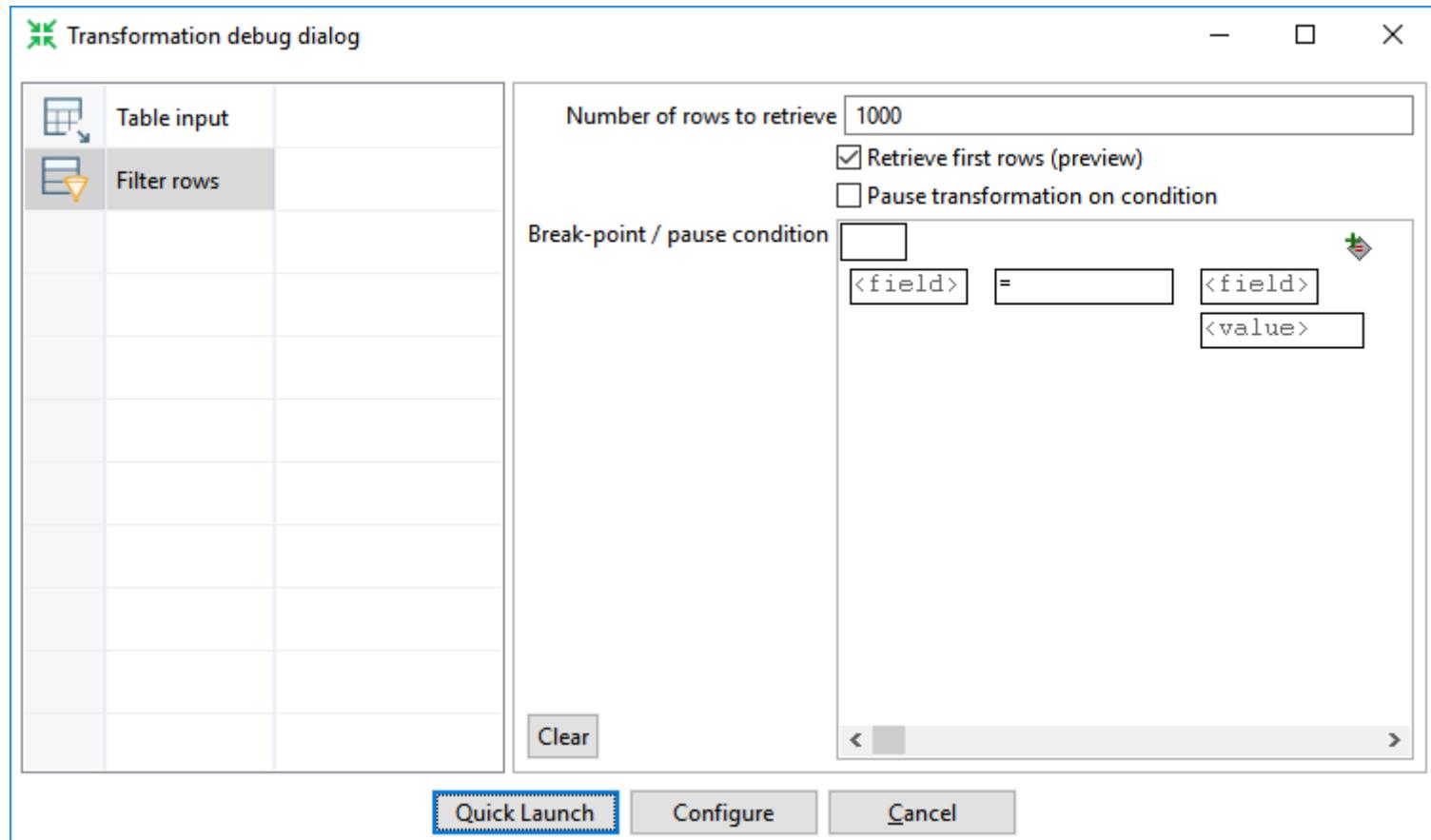
Error Handling...

Preview...

Align / Distribute

#	Stepname	Copynr
1	Table input	0

# Filter rows



# Filter rows

Examine preview data

Rows of step: Filter rows (82 rows)

#	emp_no	salary
1	11371	81461
2	11693	101179
3	14007	105453
4	17698	91443
5	17739	91836
6	17890	80046
7	25730	82887
8	25949	80946
9	26002	94825
10	30851	104788
11	40676	95940
12	43941	112704
13	44474	84378
14	47000	90163
15	49487	89924
16	52227	91021

[Close](#)

# Filter rows

Spoon - Transformation 1 (changed)

File Edit View Action Tools Help

View Design

Steps

Flow

- Abort
- Annotate Stream
- Append streams
- Block this step until steps finish
- Blocking Step
- Detect empty stream
- Dummy (do nothing)
- ETL Metadata Injection
- Filter rows**
- Identify last row in a stream
- Java Filter
- Job Executor
- Prioritize streams

Welcome! Transformation 1

Table input → Filter rows

Execution Results

#	Stepname	Copynr	Read	Written	Input	Output	Update
1	Table input	0	0	252	252	0	0
2	Filter rows	0	252	82	0	0	0

# Text file output

Spoon - Transformation 1 (changed)

File Edit View Action Tools Help

View Design

Steps

- > Input
- > Output
- > Transform
- > Utility
- > Flow
- > Scripting
- > Pentaho Server
- > Lookup
- > Joins
- > Data Warehouse
- > Validation
- > Statistics
- > Big Data
- > Agile
- > Cryptography

Welcome! Transformation 1

Table input → Filter rows

Execution Results

Execution History Logging Step Metrics Performance Graph

#	Stepname	Copynr	Read	Written	Input	Output	Update
1	Table input	0	0	252	252	0	0
2	Filter rows	0	252	82	0	0	0

```
graph LR; A[Table input] --> B[Filter rows]
```

# Text file output

The screenshot shows the Spoon interface for a transformation named "Transformation 1".

**Left Panel (Steps):**

- Output
  - Automatic Documentation Out
  - Delete
  - Insert / Update
  - JSON Output
  - LDAP Output
  - Microsoft Access Output
  - Microsoft Excel Output
  - Microsoft Excel Writer
  - Pentaho Reporting Output
  - Properties Output
  - RSS Output
  - S3 File Output
  - SQL File Output
  - Table File Output

**Middle Panel (Transformation View):**

Workflow diagram:

```
graph LR; A[Table input] --> B[Filter rows]
```

**Bottom Panel (Execution Results):**

#	Stepname	Copynr	Read	Written	Input	Output	Update
1	Table input	0	0	252	252	0	0
2	Filter rows	0	252	82	0	0	0

# Text file output

The screenshot shows the Spoon interface for a transformation named "Transformation 1".

**Left Panel (Steps):**

- Table of available steps:
  - Serialize to file
  - Synchronize after merge
  - Table output
  - Text file output** (selected)
  - Update
  - XML Output
- Category tree:
  - Transform
  - Utility
  - Flow
  - Scripting
  - Pentaho Server
  - Lookup
  - Joins
  - Data Warehouse

**Middle Panel (Transformation View):**

Workflow diagram:

```
graph LR; A[Table input] --> B[Filter rows]
```

**Bottom Panel (Execution Results):**

**Execution Results**

#	Stepname	Copynr	Read	Written	Input	Output	Update
1	Table input	0	0	252	252	0	0
2	Filter rows	0	252	82	0	0	0

# Text file output

The screenshot shows the Spoon interface for a transformation named "Transformation 1".

**Left Panel (Steps):**

- Table of contents:
  - Serialize to file
  - Synchronize after merge
  - Table output
  - Text file output** (highlighted)
  - Update
  - XML Output
- Transform, Utility, Flow, Scripting, Pentaho Server, Lookup, Joins, Data Warehouse.

**Middle Panel (Transformation View):**

Workflow diagram:

```
graph LR; A[Table input] --> B[Filter rows]; B --> C[Text file output]
```

**Bottom Panel (Execution Results):**

**Execution Results**

#	Stepname	Copynr	Read	Written	Input	Output	Update
1	Table input	0	0	252	252	0	0
2	Filter rows	0	252	82	0	0	0

# Text file output

Spoon - Transformation 1 (changed)

File Edit View Action Tools Help

View Design

Steps

- Serialize to file
- Synchronize after merge
- Table output
- Text file output**
- Update
- XML Output

Transform

Utility

Flow

Scripting

Pentaho Server

Lookup

Joins

Data Warehouse

Welcome! Transformation 1

100%

Table input → Filter rows → Text file output

hop

Result is TRUE

Result is FALSE

Main output of step

Execution Results

#	Stepname	Copynr	Read	Written	Input	Output	Update
1	Table input	0	0	252	252	0	0
2	Filter rows	0	252	82	0	0	0

# Text file output

The screenshot shows the Spoon interface for a transformation named "Transformation 1".

**Left Panel (Steps):**

- Table of contents:
  - Serialize to file
  - Synchronize after merge
  - Table output
  - Text file output** (highlighted)
  - Update
  - XML Output
- Transform
- Utility
- Flow
- Scripting
- Pentaho Server
- Lookup
- Joins
- Data Warehouse

**Middle Panel (Transformation View):**

The transformation consists of three main steps connected sequentially:

- Table input
- Filter rows
- .Text file output

**Bottom Panel (Execution Results):**

**Execution History:**

#	Stepname	Copynr	Read	Written	Input	Output	Update
1	Table input	0	0	252	252	0	0
2	Filter rows	0	252	82	0	0	0

# Text file output

Spoon - Transformation 1 (changed)

File Edit View Action Tools Help

View Design

Steps

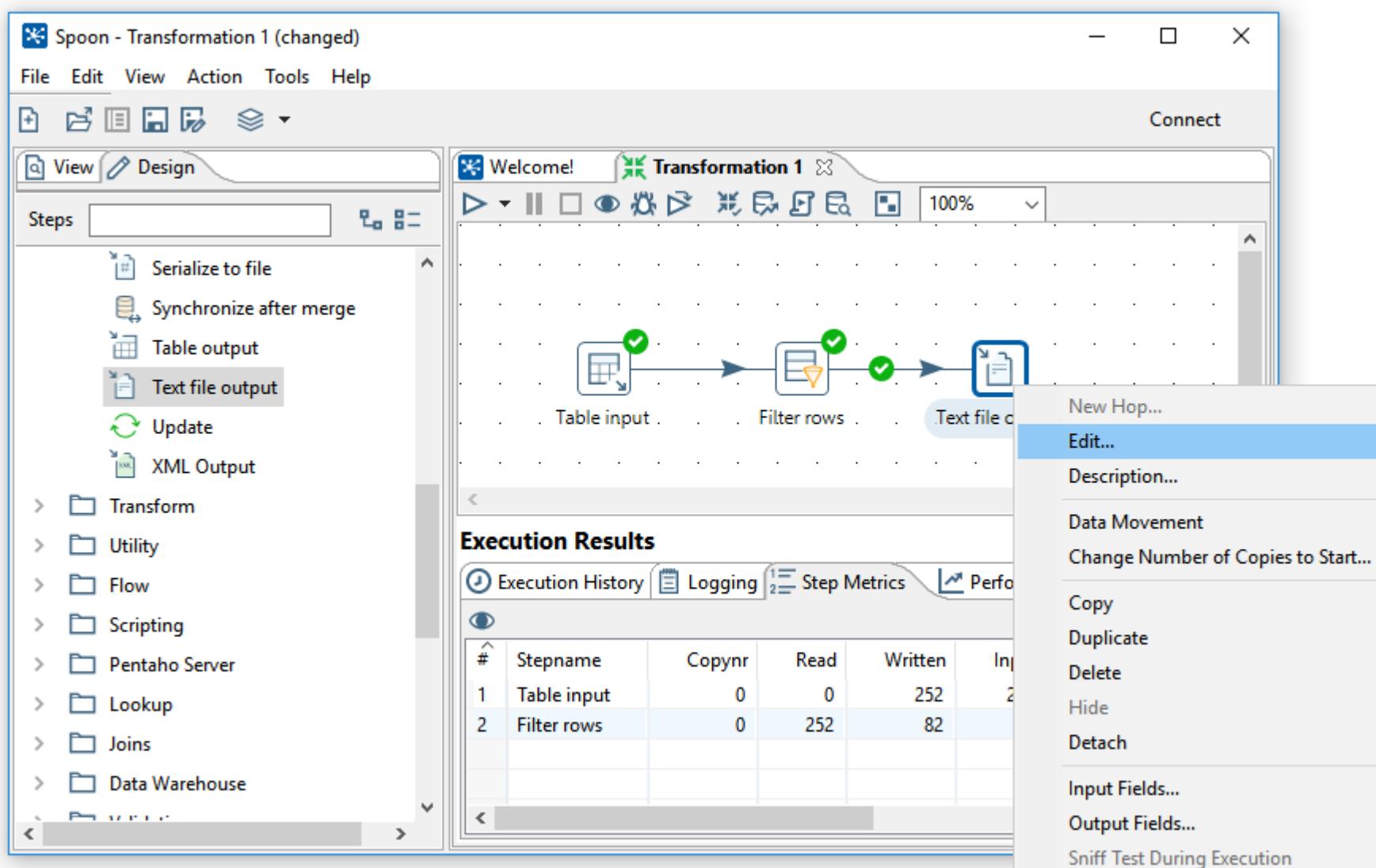
- Serialize to file
- Synchronize after merge
- Table output
- Text file output**
- Update
- XML Output

Table input → Filter rows → Text file output

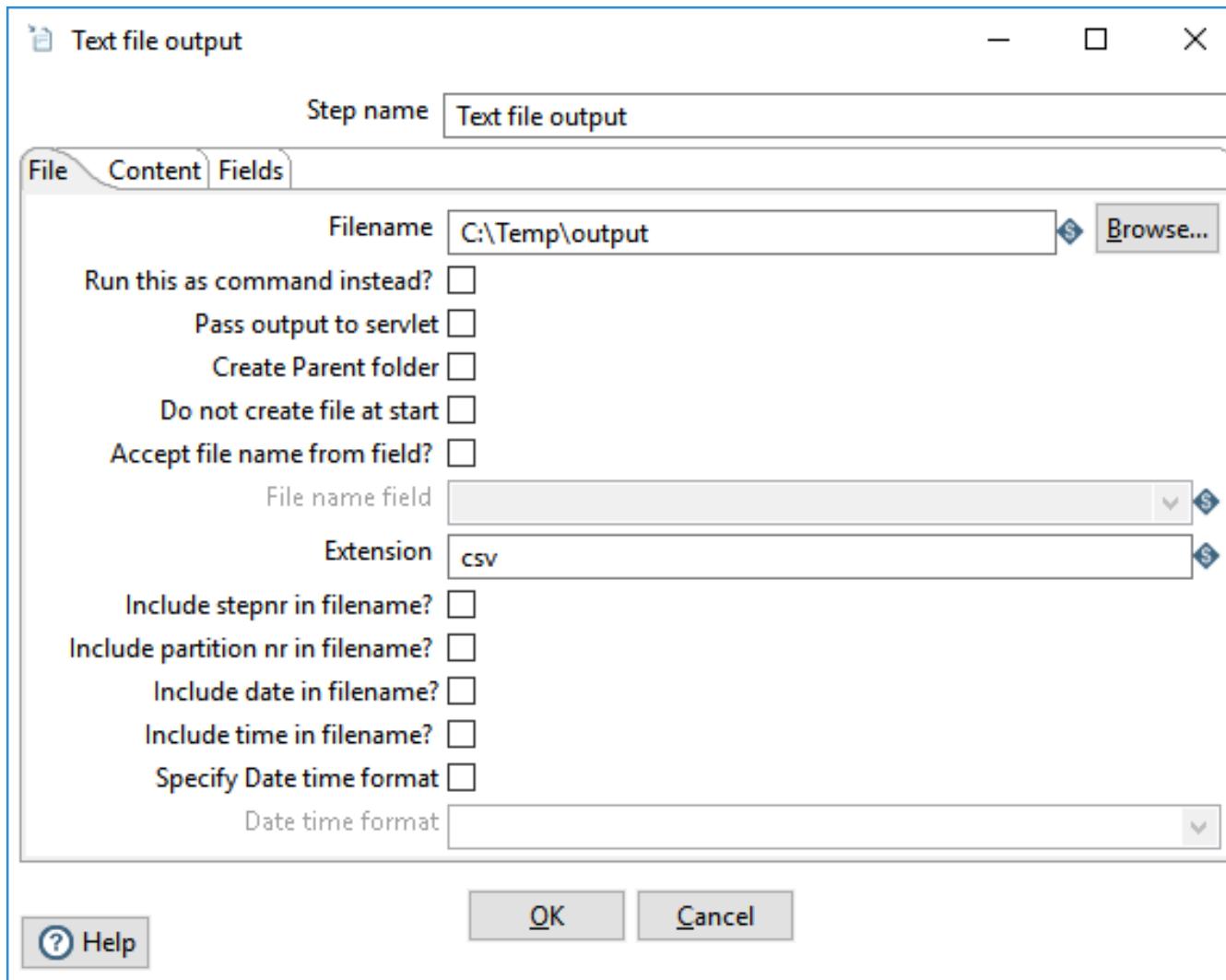
New Hop...  
Edit...  
Description...  
Data Movement  
Change Number of Copies to Start...  
Copy  
Duplicate  
Delete  
Hide  
Detach  
Input Fields...  
Output Fields...  
Sniff Test During Execution

Execution Results

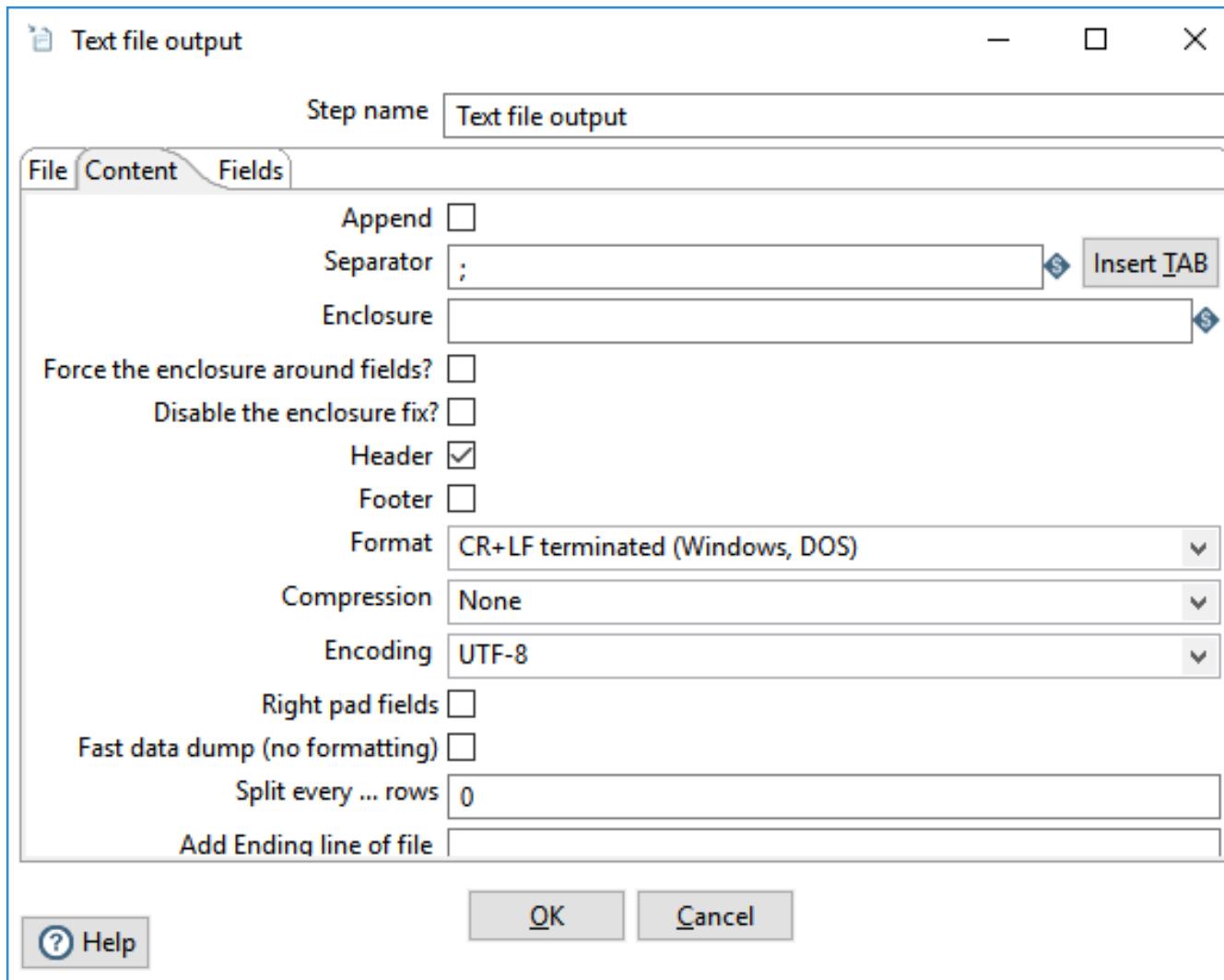
#	Stepname	Copynr	Read	Written	Inp
1	Table input	0	0	252	2
2	Filter rows	0	252	82	2



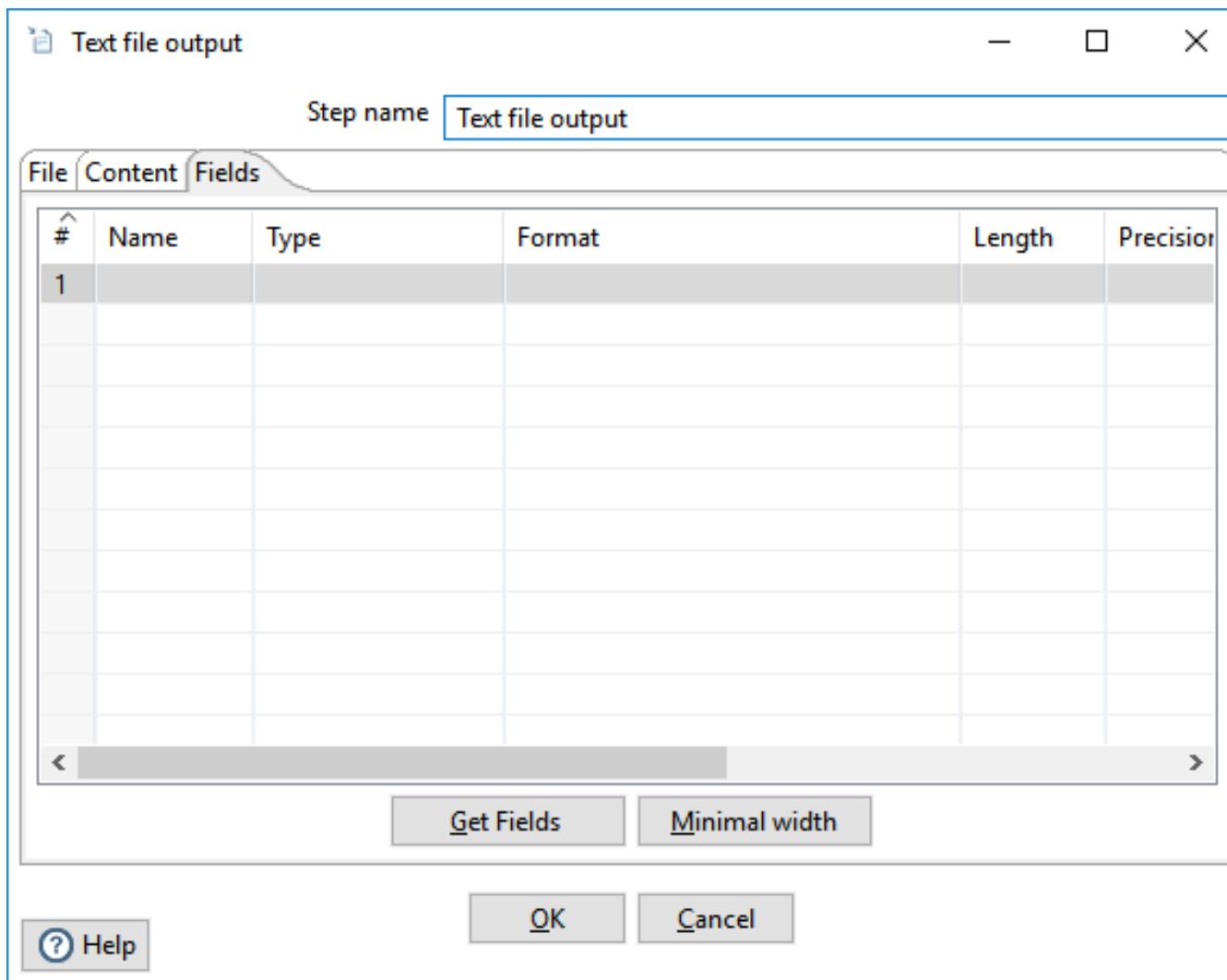
# Text file output



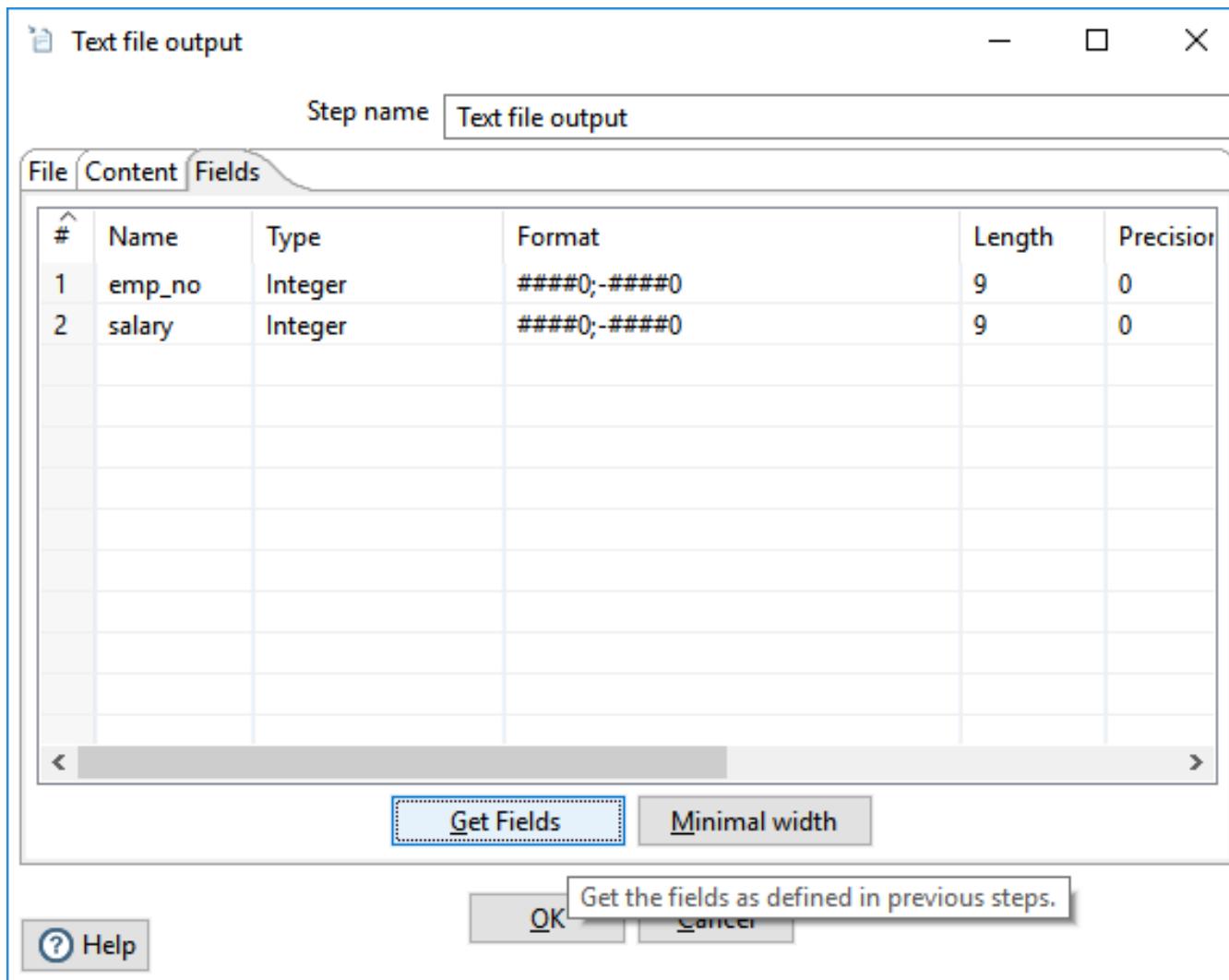
# Text file output



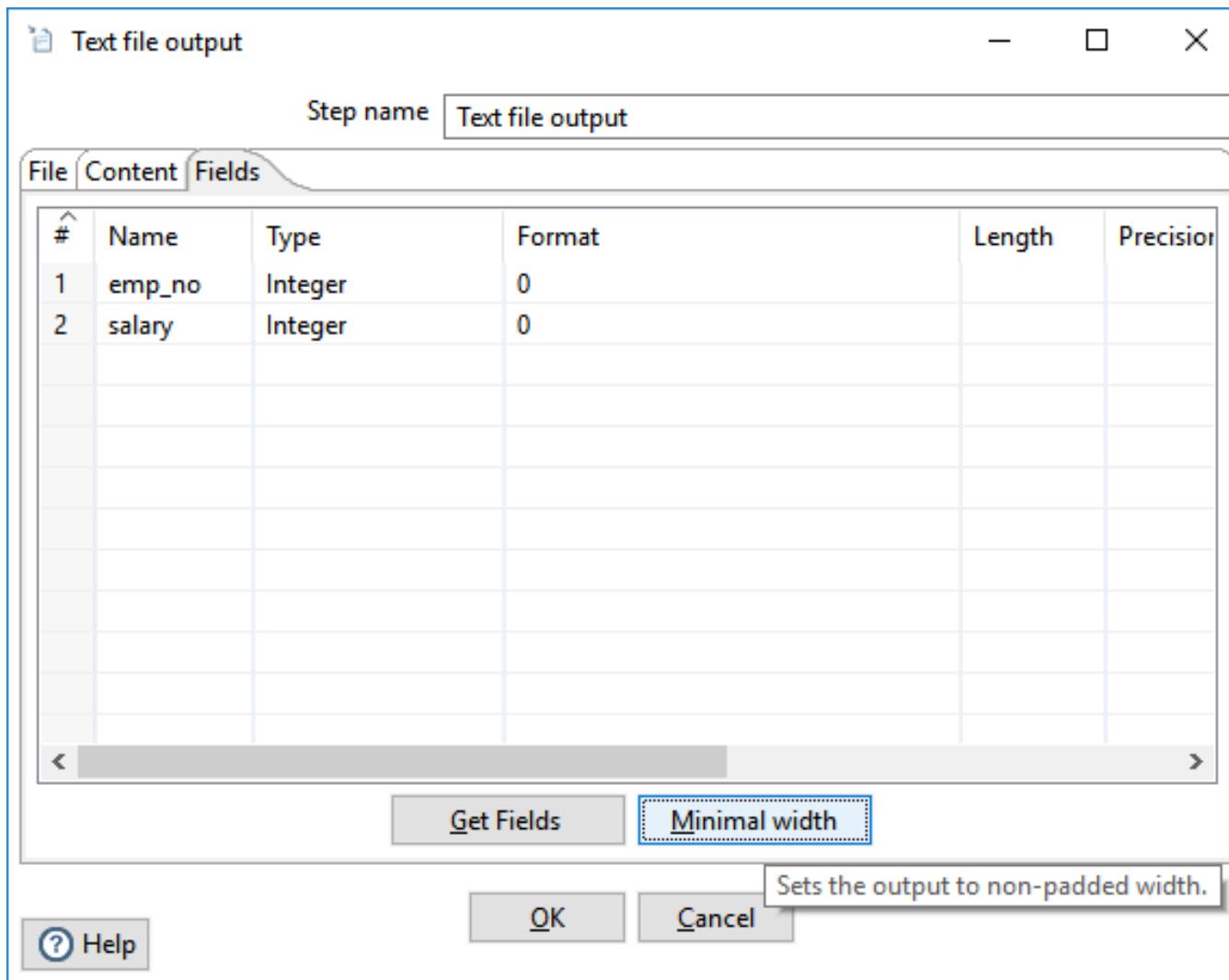
# Text file output



# Text file output



# Text file output



# Text file output

Spoon - Transformation 1 (changed)

File Edit View Action Tools Help

View Design

Steps

- Serialize to file
- Synchronize after merge
- Table output
- Text file output**
- Update
- XML Output

Transform

Utility

Flow

Scripting

Pentaho Server

Lookup

Joins

Data Warehouse

Table input

Filter rows

Text file output

Execution Results

Execution History Logging Step Metrics Performance Graph

#	Stepname	Copynr	Read	Written	Input	Output	Update
1	Table input	0	0	252	252	0	0
2	Filter rows	0	252	82	0	0	0

```
graph LR; A[Table input] --> B[Filter rows]; B --> C[Text file output]
```

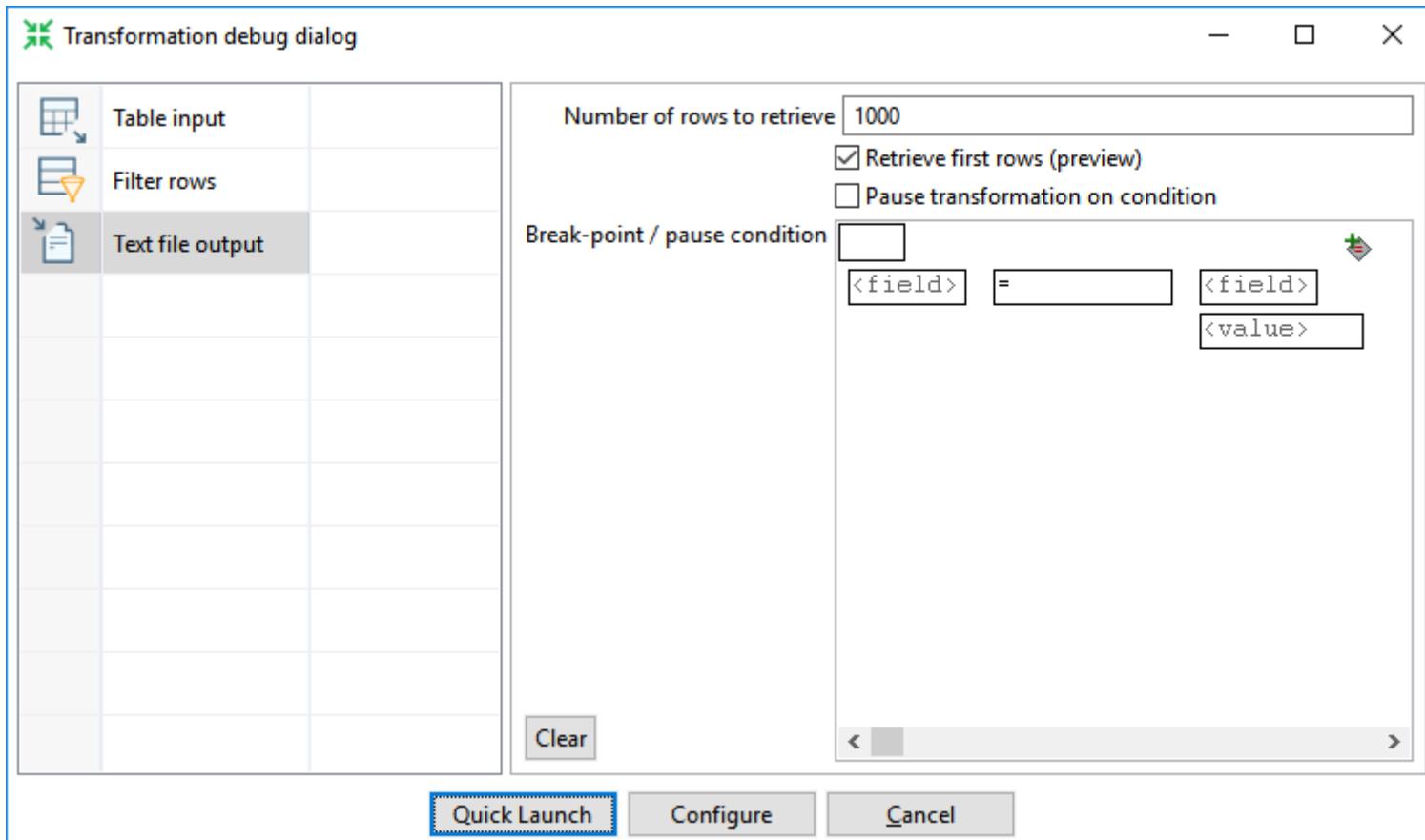
# Text file output

The screenshot shows the Pentaho Data Integration (Kettle) interface. On the left, the 'Step Catalog' sidebar is open, displaying various output step types: Serialize to file, Synchronize after merge, Table output, **Text file output**, Update, XML Output, Transform, Utility, Flow, Scripting, Pentaho Server, Lookup, Joins, and Data Warehouse. The 'Text file output' step is currently selected. The main workspace shows a transformation with three steps connected sequentially: 'Table input' (green checkmark), 'Filter rows' (orange triangle icon), and 'Text file' (blue document icon). Below the workspace is the 'Execution Results' panel, which includes tabs for Execution History, Logging, Step Metrics, and Perf. The Step Metrics tab displays the following data:

#	Stepname	Copynr	Read	Written	Ir
1	Table input	0	0	252	
2	Filter rows	0	252	82	

A context menu is open over the 'Text file' step, listing various options: New Hop..., Edit..., Description..., Data Movement, Change Number of Copies to Start..., Copy, Duplicate, Delete, Hide, Detach, Input Fields..., Output Fields..., Sniff Test During Execution, Check Selected Step(s), Error Handling..., **Preview...**, and Align / Distribute.

# Text file output



# Text file output

Examine preview data

Rows of step: Text file output (82 rows)

#	emp_no	salary
1	11371	81461
2	11693	101179
3	14007	105453
4	17698	91443
5	17739	91836
6	17890	80046
7	25730	82887
8	25949	80946
9	26002	94825
10	30851	104788
11	40676	95940
12	43941	112704
13	44474	84378
14	47000	90163
15	49487	89924
16	52227	91021

[Close](#)

# Text file output

Spoon - Transformation 1 (changed)

File Edit View Action Tools Help

View Design

Steps

- Serialize to file
- Synchronize after merge
- Table output
- Text file output**
- Update
- XML Output

Transform

Utility

Flow

Scripting

Pentaho Server

Lookup

Joins

Data Warehouse

Welcome! Transformation 1

Table input → Filter rows → Text file output

Execution Results

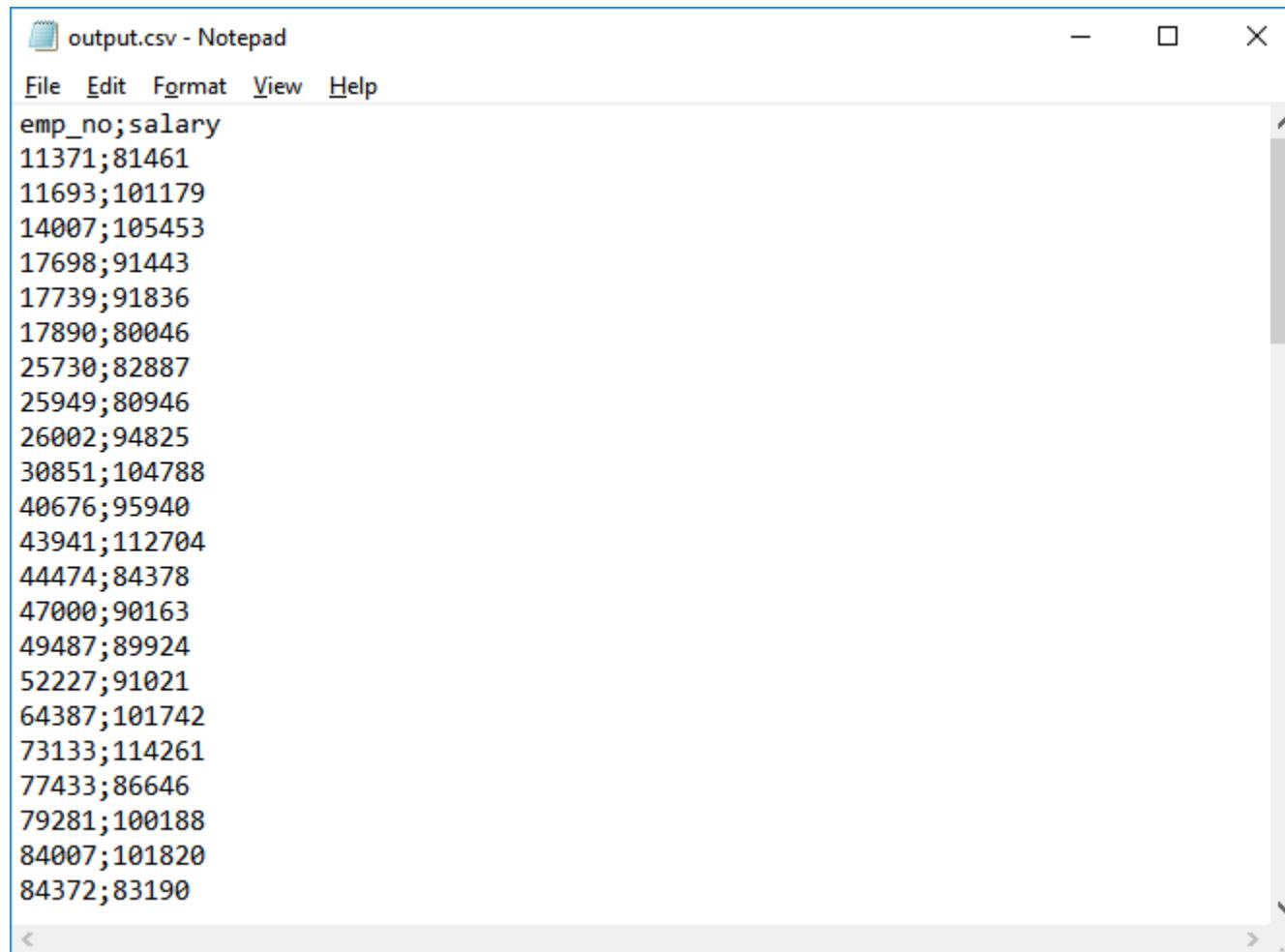
Execution History Logging Step Metrics Performance Graph

#	Stepname	Copynr	Read	Written	Input	Output	Upc
1	Table input	0	0	252	252	0	
2	Filter rows	0	252	82	0	0	
3	<b>Text file output</b>	0	82	82	0	<b>83</b>	

The screenshot shows the Spoon interface for Apache Pentaho Data Integration. The left sidebar lists various step types, with 'Text file output' selected. The main workspace displays a transformation named 'Transformation 1' with three steps: 'Table input', 'Filter rows', and 'Text file output'. The 'Text file output' step has a red oval around its 'Output' value of 83 in the execution results table. The execution results table also shows data for the other steps.

#	Stepname	Copynr	Read	Written	Input	Output	Upc
1	Table input	0	0	252	252	0	
2	Filter rows	0	252	82	0	0	
3	<b>Text file output</b>	0	82	82	0	<b>83</b>	

# Output



The screenshot shows a Windows Notepad window titled "output.csv - Notepad". The window contains a list of employee numbers and salaries separated by a semicolon. The data is as follows:

emp_no;salary
11371;81461
11693;101179
14007;105453
17698;91443
17739;91836
17890;80046
25730;82887
25949;80946
26002;94825
30851;104788
40676;95940
43941;112704
44474;84378
47000;90163
49487;89924
52227;91021
64387;101742
73133;114261
77433;86646
79281;100188
84007;101820
84372;83190

# Output



**Text Import - [output.csv]**

**Import**

Character set: Unicode (UTF-8)

Language: Default - English (USA)

From row: 1

**Separator Options**

Fixed width  Separated by

Tab  Comma  Semicolon  Space  Other

Merge delimiters Text delimiter: "

**Other Options**

Quoted field as text  Detect special numbers

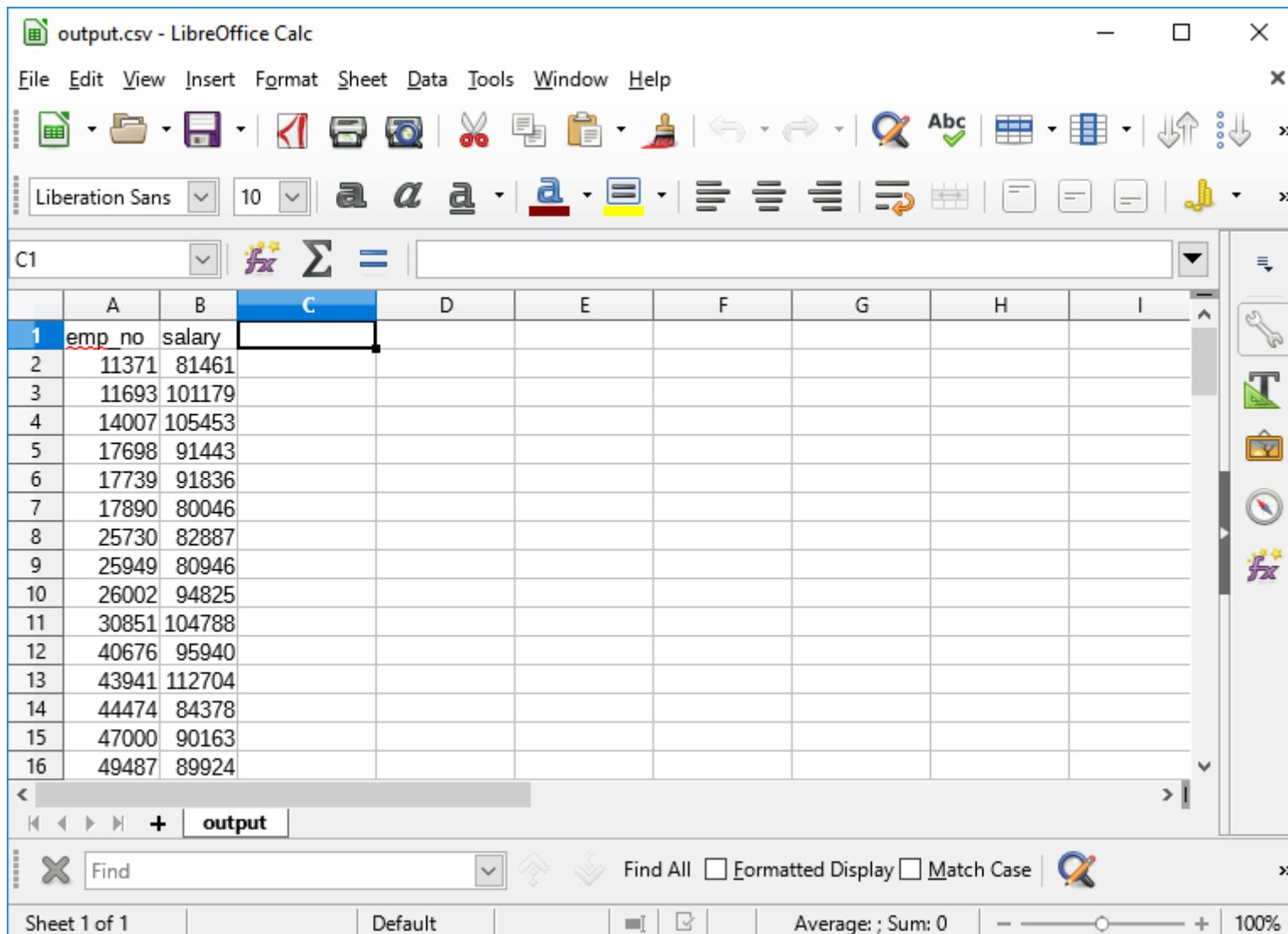
**Fields**

Column type:

	Standard	Standard
1	emp_no	salary
2	11371	81461
3	11693	101179
4	14007	105453
5	17698	91443
6	17739	91836
7	17890	80046
8	25730	82887
9	25949	80946

Help OK Cancel

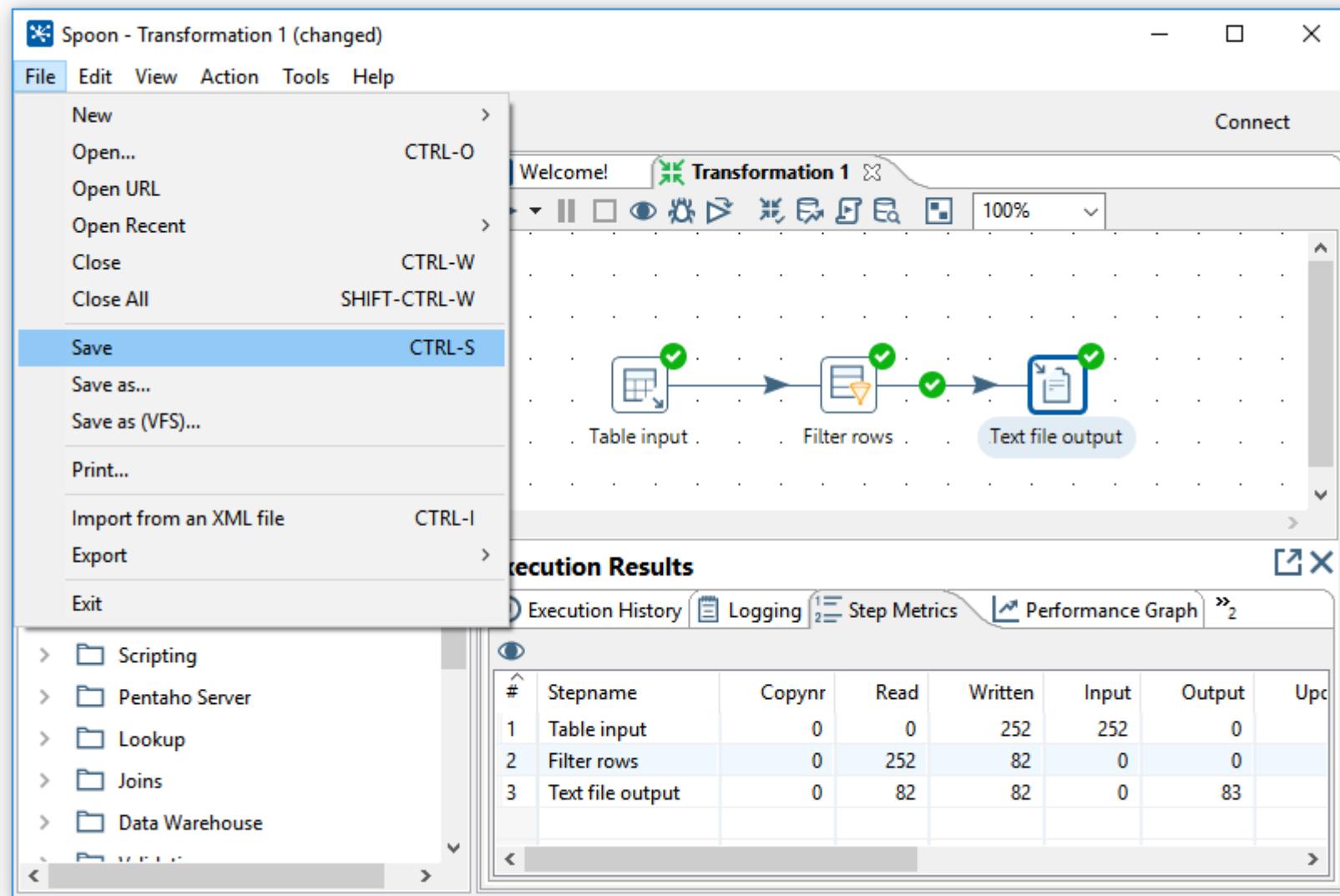
# Output



The screenshot shows a LibreOffice Calc spreadsheet window titled "output.csv - LibreOffice Calc". The menu bar includes File, Edit, View, Insert, Format, Sheet, Data, Tools, Window, and Help. The toolbar contains various icons for file operations, text styling, and data manipulation. The ribbon below the toolbar shows font style (Liberation Sans), font size (10), and a set of icons for text and alignment. The main worksheet area displays a table with 16 rows and 3 columns. The first row is a header with "C1" in the formula bar. The second column (A) contains employee numbers, and the third column (B) contains salaries. The fourth column (C) is empty and selected. The status bar at the bottom shows "Sheet 1 of 1", "Default", "Average: ; Sum: 0", and a zoom level of 100%.

	A	B	C	D	E	F	G	H	I
1	emp_no	salary							
2	11371	81461							
3	11693	101179							
4	14007	105453							
5	17698	91443							
6	17739	91836							
7	17890	80046							
8	25730	82887							
9	25949	80946							
10	26002	94825							
11	30851	104788							
12	40676	95940							
13	43941	112704							
14	44474	84378							
15	47000	90163							
16	49487	89924							

# Save transformation



# Run transformation

