



**ataccama**

# ONE Desktop Workshop

Flow Control

Prepared for: v15.4.x

Prepared by: Ataccama

Dated: October 2024










## Contents of the Document

1. Introduction.....	3
2. Tasks.....	4
2.1. Create a plan and add a data source.....	5
2.2. Alter format step.....	6
2.3. Join step.....	8
2.4. Extract Filter step.....	10
2.5. Condition step.....	11
2.6. Multiplier, Union, and Union Same steps.....	12
2.7. Sort step.....	14
2.8. Text Outputs.....	16
2.10. Run the plan.....	18
3. Conclusion.....	19
Correct answers, hints, and useful tips.....	20

# 1. Introduction

This workshop will introduce the use of the Flow Control steps within the Ataccama ONE Desktop.

After data enters via an input step into a plan, typically some transformations are required before the data flow reaches the output. We will cover the following steps in this workshop:

	Alter Format		Multiplicator
	Join	-	
	Extract Filter		Union and Union Same
	Condition		Sort

We will also output our data into text files.

## 2. Tasks

All activities will include the usage of the above-mentioned steps over data coming from the **party\_full\_1.csv** sample file. To simulate a real-life scenario, let's imagine we have received some requirements to transform our incoming data. The following is needed:

- **Create new columns:**
  - **upper\_name** turn the **src\_name** column into upper case.
  - **sin\_number** strip **src\_sin** of non-digits and store the value as **LONG** type.
  - **fix\_email** replace any **src\_email** values with "gmail" to "googlemail"
- **Provide** values (English name only) into a new **client\_tier\_name** attribute received from a new data source: **client\_tiers.csv**
- **Split the data** stream as follows:
  - Stream 1: all the data in the flow sorted by the **src\_name** attribute.
  - Stream 2: only Platinum owners of credit cards with **src\_card** values starting with '4'
  - Stream 3: only Platinum owners of credit cards with **src\_card** values NOT starting with '4'

For Streams 2 and 3 only output the **src\_primary\_key**, **src\_name**, and **src\_card** attributes.

## 2.1. Create a plan and add a data source.

In the initial phase of this workshop, we need to read and prepare the ground for the upcoming changes:

- › In your existing Training Project's **plans** folder, create a new plan called **02\_party.plan**
- › Add a **Text File Reader** step for the **party\_full\_1.csv** into this plan and make sure you set the metadata correctly.
- › Don't forget to change the **type** for the **meta\_last\_update** attribute to **DATE** and set the **date format** as yyyy/MM/dd

The screenshot shows the 'Properties of Text File Reader' dialog box. On the left, a tree view shows the 'Columns (10)' section with the following items: src\_name:STRING, src\_gender:STRING, src\_birth\_date:STRING, src\_sin:STRING, src\_card:STRING, src\_address:STRING, src\_email:STRING, src\_primary\_key:INTEGER, **meta\_last\_update:DATE** (highlighted), and src\_tier:STRING. Below this are 'Shadow Columns (0)', 'Data Format Parameters', and 'Error Handling Strategy'. The main area on the right is titled 'meta\_last\_update:DATE'. It contains a 'Name\*' field with 'meta\_last\_update', a 'Type\*' dropdown menu set to 'DATE', and an 'Ignore' checkbox which is unchecked. There is a link 'delete dataFormatParameters'. Below that, 'Date Format Locale\*' is set to 'en' and 'Date format\*' is set to 'yyyy/MM/dd'. At the bottom, there is a 'Comments (Hide)' section with an 'Edit...' button. At the very bottom of the dialog are 'OK', 'Cancel', and 'Apply' buttons.



HINT

To add new steps to your plan's canvas, find the desired steps in the right section of the toolbar and use the **CTRL+I** shortcut or press the **'Insert'** key.



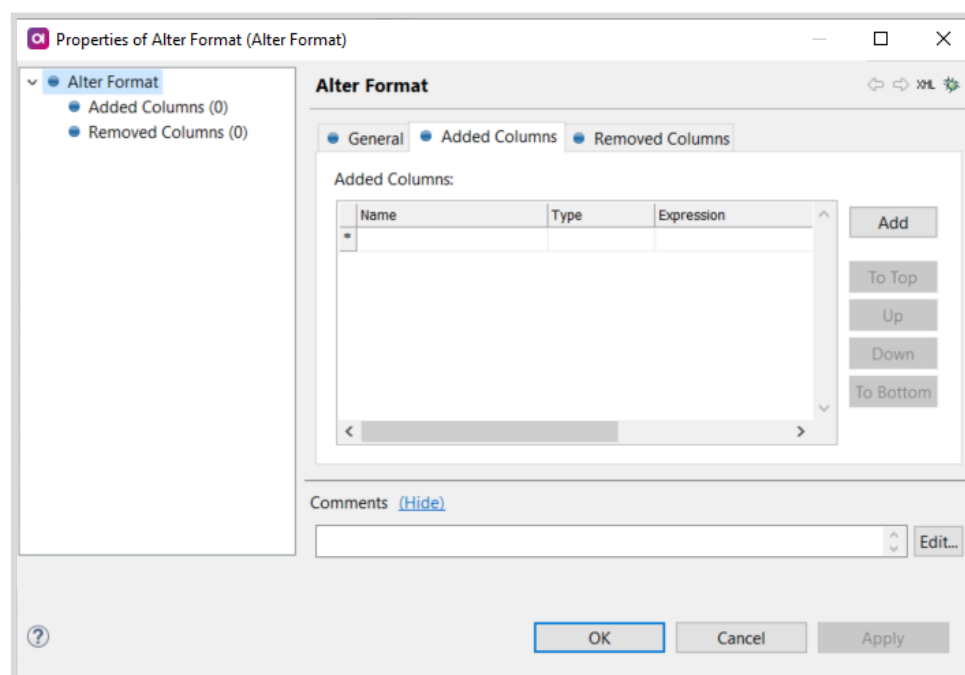
NOTE

The procedure of creating plans and reading files has already been covered. Refer to sections 3.1 to 3.4 of the **Data Profiling Workshop** for more detailed instructions.

## 2.2. Alter format step.

It's time to start working on the data flow and modifying it:

- › Find the **Alter format** step and add it to your plan.
- › Connect the **'out'** endpoint of the **Text File Reader** to the **'in'** endpoint of the **Alter Format** step.
- › Open the **Alter Format** step and select the **Added Columns** tab:



Here you can define the details of the new columns to be added – names, data types, and expressions as per requirements at the beginning of this workshop. You can add optional **Comments** to allow other people to understand what the purpose of each column is.

- › Go ahead and create the 3 new columns including their **Expression** definitions. Look at the task requirements for details of what's needed.



*It's up to you now! Try to figure out the logic inside each of the 3 new columns **upper\_name**, **sin\_number**, and **fix\_email** on your own. Write and test expressions for the right outcome.*

*If in doubt, refer to the end of this workshop for correct answers!*



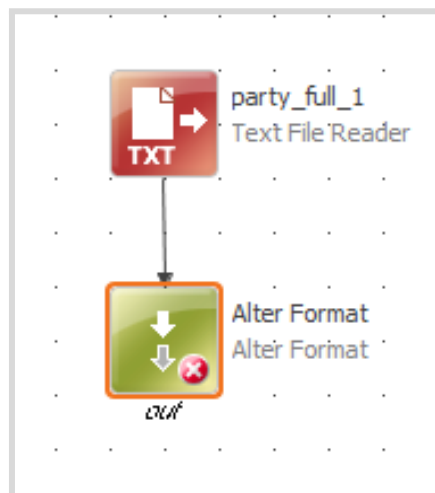
- These expressions may come in handy: **upper()**, **trashNonDigits()**, **replace()**.
- Remember to use the **CTRL+Space** key for content assist.
- Hover over the expressions for more help!


When finished, your **Alter Format** step should create the 3 new columns and apply expressions to their values at the same time.



Usually, the **Alter Format** step only creates or removes columns - it is not mandatory to define any expression right away. It is recommended to modify the data later in a separate step, so it is easy to see the changes in the plan's flow. However, you can still do both actions together in a single step just like we did.

- Your plan should now look like this:



As you can see, the **Alter Format** step shows the  symbol, because its **'out'** endpoint is not connected to anything else. Let's move on to the next requirement.

## 2.3. Join step.

The next step will incorporate a new branch with data. It is time to use a **Join** step that allows you to merge two inputs together into one while configuring the merging behavior. In our example, we need to link the values of **src\_tier** in our existing flow with the same value present in the new incoming file.



Check the contents of the **client\_tiers.csv** file and find the attribute that could be joined with the **src\_tier** values.



### JOIN Step – Join Type Options

Using the setting is much like an SQL join:

- **INNER** join will output records if they exist in both files.
- **OUTER** join will output records that only exist in one or the other file.
- **LEFT** join will output all records on the “left” side. It will provide values from the right side if they exist. If they do not exist, the values will be NULL.
- **RIGHT** join will output all records on the “right” side. It will provide values from the left side if they exist. If they do not exist, the values will be NULL.

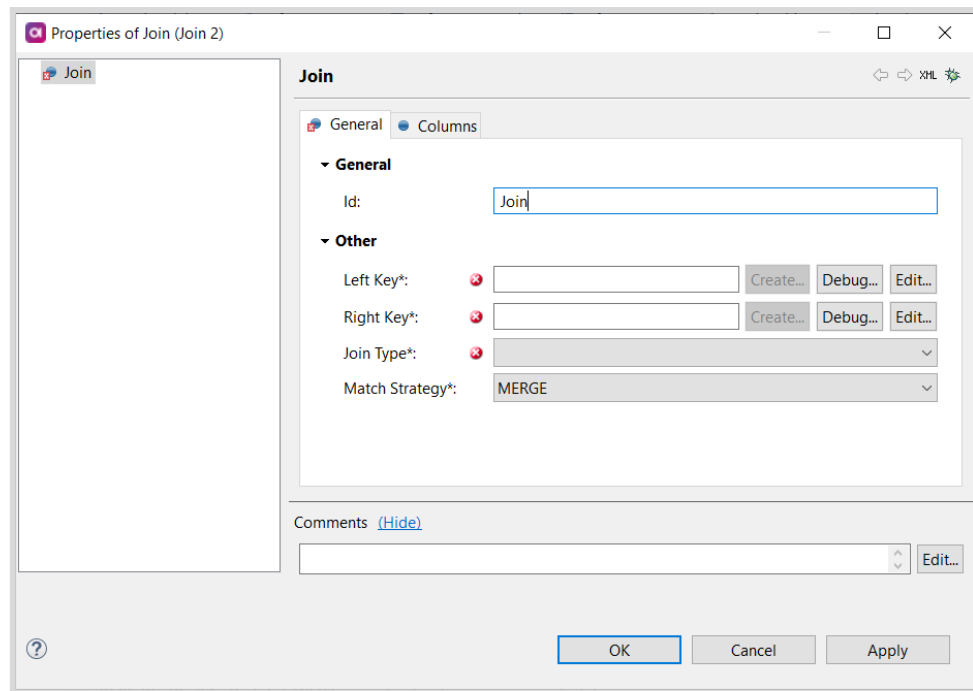
It seems like the **LEFT** type of join would be best to obtain the client tier name values.

- › Locate and add a new file – **client\_tiers.csv** into your plan.
- › Find and add a **Join** step into your plan.
- › Connect the output from the **Alter Format** into the **‘in\_left’** input of the **Join** step.
- › Connect the output from the **client\_tiers.csv** to the **‘in\_right’** input of the **Join** step.

The **Join** step should now be receiving all the necessary data. It is time to configure the details of how they will be joined together:

- › Open the **Join** step properties!
- › Make sure the **General** tab is selected.
- › Configure the **Left** and **Right** joining keys – which ones did you find out in the task earlier?
- › Also choose the right type of join – **LEFT**:



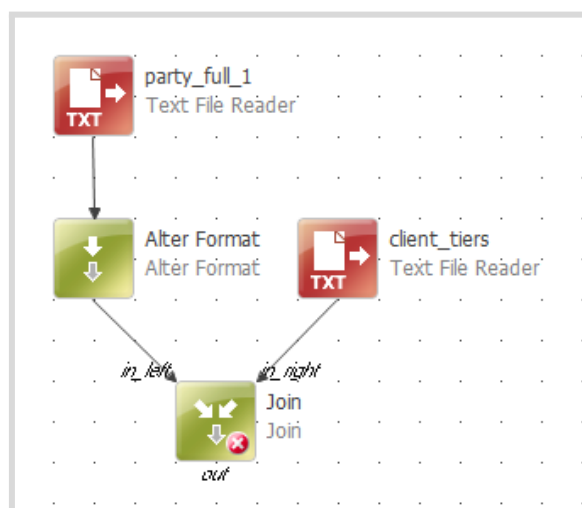


- › Switch to the **Columns** tab to configure which columns should be on the output of the **Join** step. You can use the '**Fill Columns...**' button to assist you.

Remember – we only need the English name of the tier.

- › Choose which attribute from the '**in\_right**' input would be required. Call this column as per requirement – **client\_tier\_name**.

Your plan should look something like this:



## 2.4. Extract Filter step

The next part of the workshop is to split the data streams into several according to different conditions as listed in the requirements.

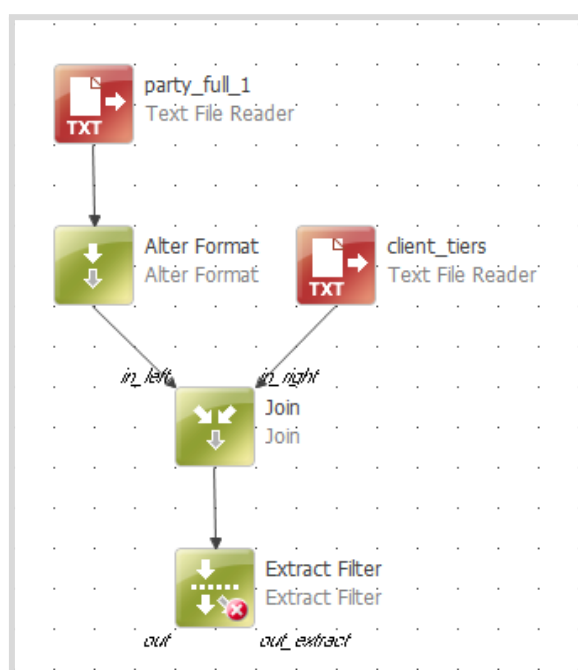
Firstly, we need to separate one stream with all the data and another stream with just Platinum clients. This stream will then be divided further:

- › Add the **Extract Filter** step to your plan.
- › Connect the end point from the **Join** step to the input of the **Extract Filter** step.
- › Open the **Extract Filter** step properties and fill in the expression to be the output to the **'out\_extract'** endpoint.
- › The whole data stream will be kept intact and continue to the **'out'** endpoint.



*If we did not need to keep the full stream, but only needed Platinum clients, we could have used the **Filter** step. The condition expression would be the same.*

Your plan should now look like this:



## 2.5. Condition step

The second part of the stream segregation is to split the data stream based on the Platinum clients. Values of the **src\_card** attribute starting with '4' will be used as the dividing condition.

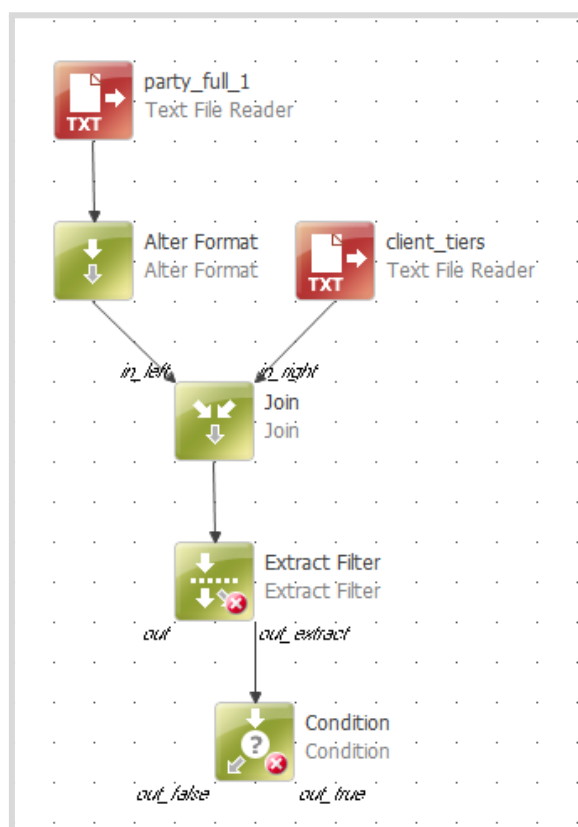


Can you work out how to add the **Condition** step into the plan and configure it?



The expression **left()** can be used here. Can you think of any other expressions that could be used to achieve the same result?

Your plan should now look like this:



## 2.6. Multiplier, Union, and Union Same steps

Surprise! You have just received a new requirement!

- **Produce an extract for contact information** with the following columns:
  - **src\_primary\_key** (STRING)
  - **contact\_type** (STRING) with either 'ADDRESS' or 'EMAIL' values as appropriate.
  - **contact\_info** (STRING) store either the values of **src\_address** or **fix\_email** here.

This output (Stream 4) should only contain **src\_primary\_key**, **contact\_type**, and **contact\_info**.



*Can you think of how to enhance the plan to provide this extract?*

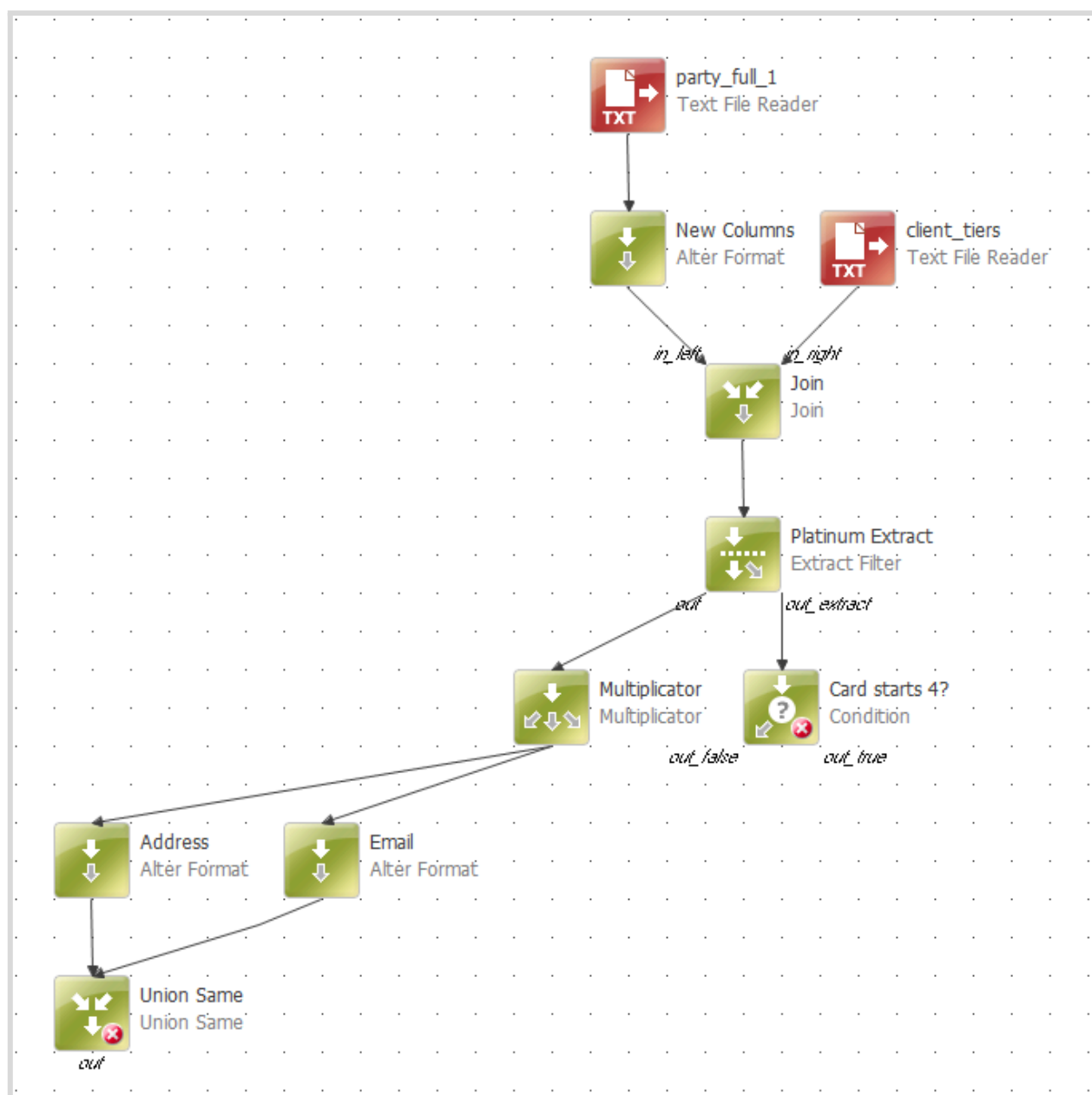


- Use a **Multiplier** step to duplicate the data into multiple streams.
- Use an **Alter Format** step to add new columns.
- Use **Union** or **Union Same** to combine the data back into one stream.
- You can rename steps and/or add comment boxes to give more explanation to what the steps do.



- The **Union** step allows 2 sources to be combined and you can pick which columns are used.
- The **Union Same** step allows more than 2 sources to be combined. They must all have the same column names and data types.
- You can click on the individual incoming connectors and check the **Properties** window at the bottom of the screen to see the incoming columns of each stream.

The plan could now look like this. Note that some steps' names have been changed to indicate what is happening:



## 2.7. Sort step

Let's focus on your original requirement for Stream 1. The full data extract needs to be sorted by the **src\_name** attribute. This can be easily achieved by the **Sort** step.

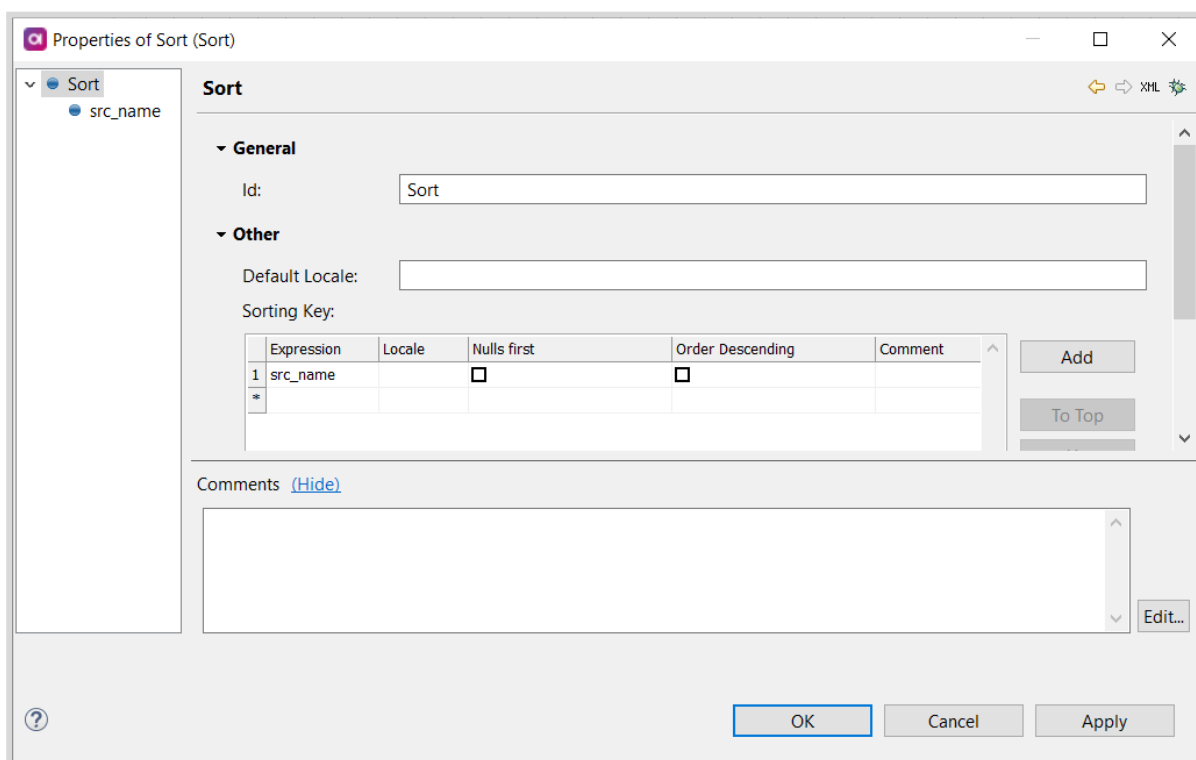
Remember this is a separate request from the remaining streams – it needs to be a new flow branch from the **Multiplicator** step.

- › Add a **Sort** step to the plan.
- › Connect the '**out**' end point from the **Multiplicator** step to the **Sort** step.
- › Define the value for the **Sorting Key** attribute – enter **src\_name** in the expression field.



NOTE

- Select the appropriate checkboxes if you need to **Order Descending** or **Nulls First**
- The sorting of key expressions can be complex – no need to restrict to attribute names only.



Properties of Sort (Sort)

Sort

Id: Sort

Default Locale:

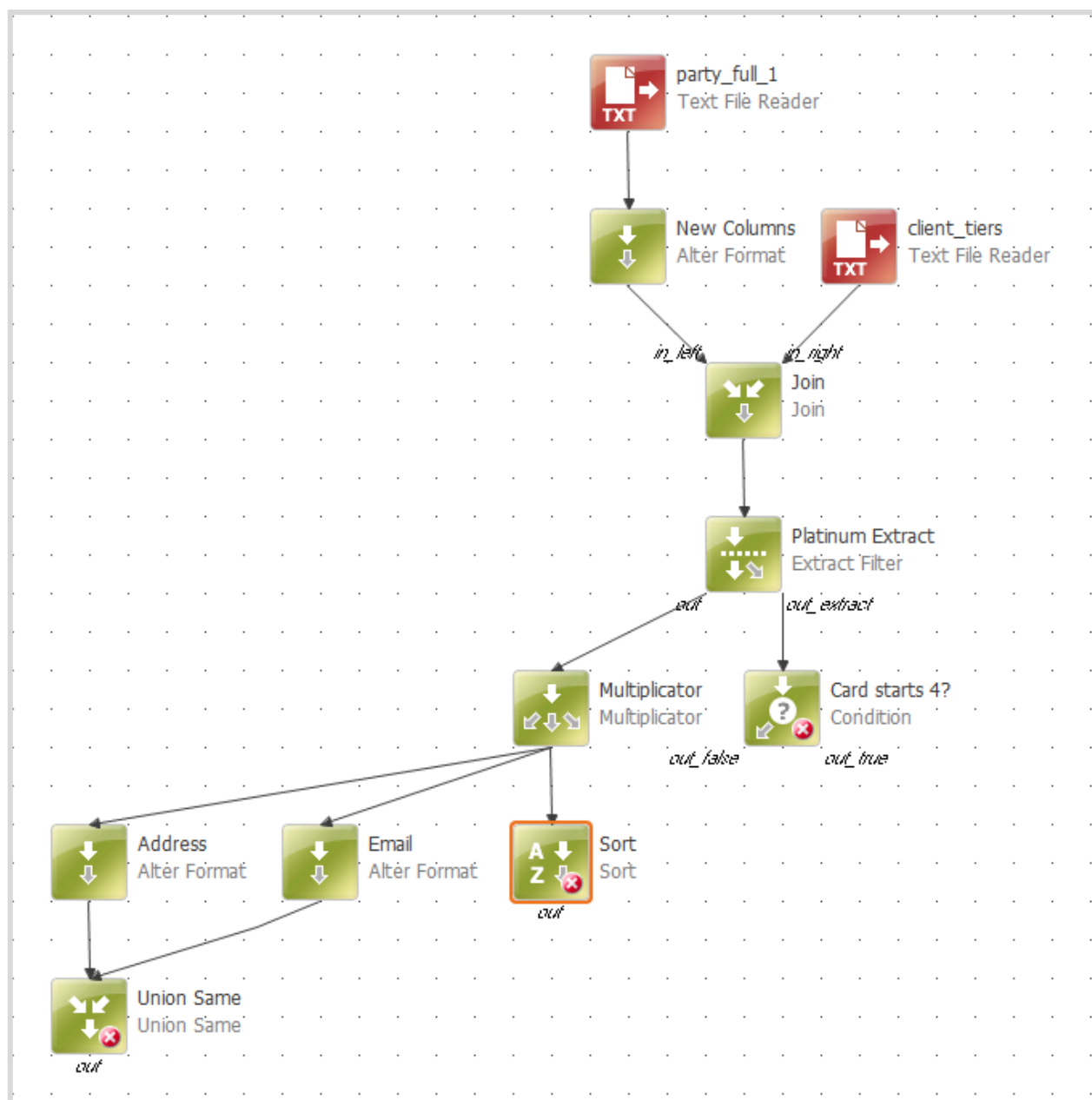
Sorting Key:

	Expression	Locale	Nulls first	Order Descending	Comment
1	src_name		<input type="checkbox"/>	<input type="checkbox"/>	
*					

Comments [\(Hide\)](#)

OK Cancel Apply

At this point, your plan should look similar to this:



## 2.8. Text Outputs

It's time to output our results! All outputs can be created as text file outputs.

- › create the following outputs:

- Stream 1:** all the data is sorted by `src_name`. Output this to `data \ out \ party_out_all.csv`.
- Stream 2:** only Platinum clients with `src_card` starting with '4'. Output this to `data \ out \ party_out_plat_4.csv`.
- Stream 3:** only Platinum clients with `src_card` NOT starting with '4'. Output this to `data \ out \ party_out_plat_not4.csv`.
- Stream 4:** the contact information extract requested in **Step 2.6**. Output this to `data \ out \ party_contact_info.csv`.

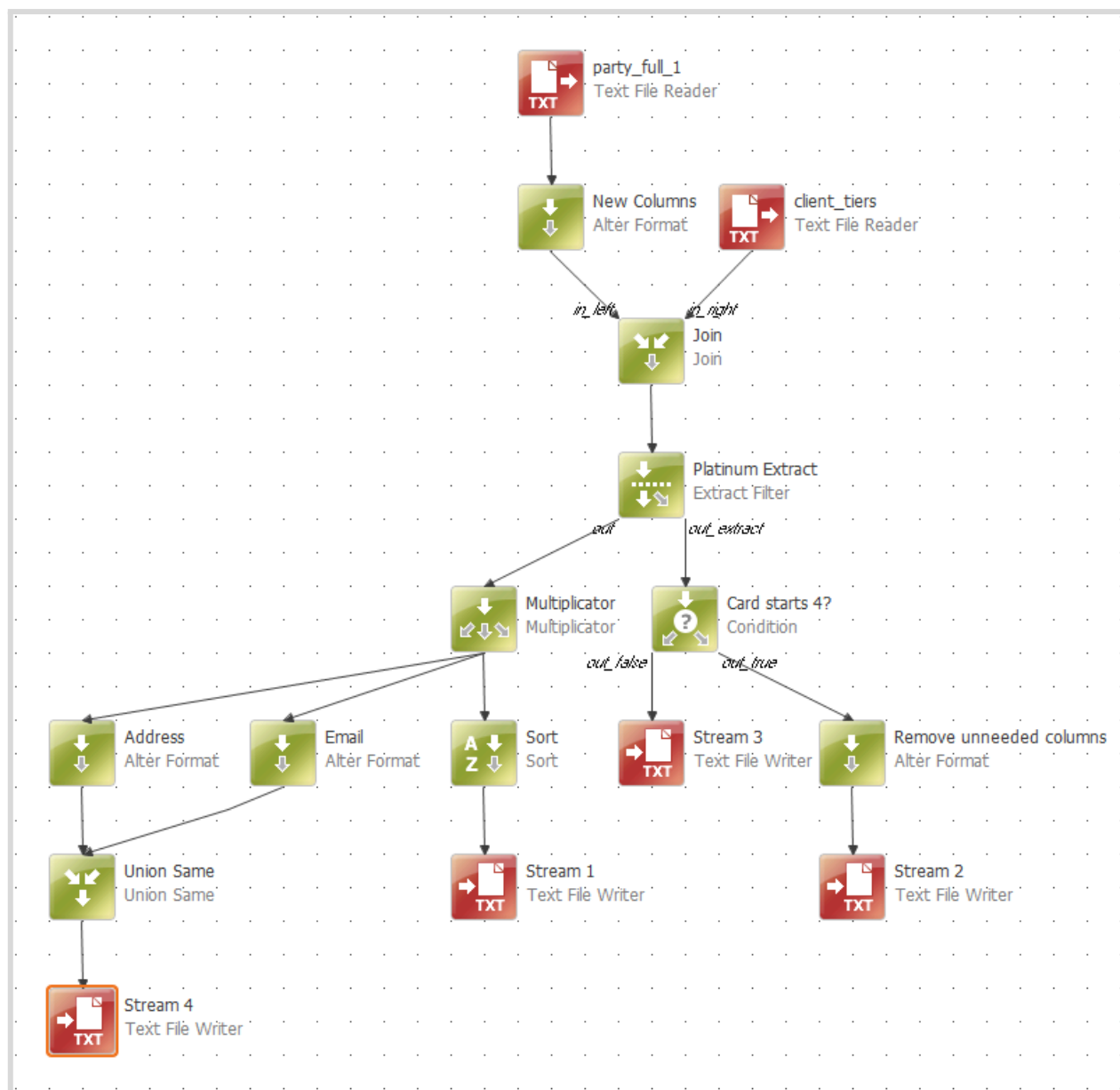
For Streams 2 and 3, only output `src_name`, `src_card`, and `src_primary_key`, while for Stream 4 only output `src_primary_key`, `contact_type`, and `contact_info`.



- Configure the outputs using the **Text File Writer** steps.
- Use another **Alter Format** step to remove some columns that are not required for Stream 2
- For Streams 3 & 4, select the required columns in the **Text File Writer** step



Your final plan should look like this:



## 2.10. Run the plan.

- › **Run** the plan and verify it completes without any errors.
- › Inspect your output files to check the results of all streams.



*Alternatively, you can create profiles of the text file outputs to verify your data flow is correct.*

### 3. Conclusion

We have come to the end of this workshop. We have covered some ways in which to transform data and output it to a database.

It is important to bear in mind that there are multiple ways to transform data to arrive at the same outcome. Our various workshop tasks above were put together to illustrate just one way of achieving our result and to show some different steps available in ONE Desktop. It is by no means the only way and often there are many ways how to achieve the same results. By the time you start dealing with large volumes of data, there could be performance implications depending on the logic and features you choose, so it's good to know the different tools you have available.

In this workshop, we started with input from several data sources, added their transformations, and configured multiple different outputs. For bigger projects, you may find advantages in configuring input and output first and working out the logic later. This ensures you can run a plan as you make changes and test your solution continuously. You can catch mistakes (if any!) as you go rather than at the end of the project when you would need to go back through the entire plan to find your mistakes.

## Correct answers, hints, and useful tips

Find answers and solutions to various tasks across this workshop:

The first **Alter Format** step should look something like this:

Alter Format

General

Added Columns

Removed Columns

Added Columns:

	Name	Type	Expression	Comment
1	upper_name	STRING	upper(src_name)	
2	sin_number	LONG	tolong(trashNonDigits(src_sin))	
3	fix_email	STRING	replace(src_email,'gmail','googlemail')	
*				

**Join** step should look like this:

Join

General

Columns

General

Other

Id:

Join

Left Key\*:

src\_tier

Right Key\*:

code

Join Type\*:

LEFT

Match Strategy\*:

MERGE

Join

General

Columns

Column Definitions:

	Name	Type	Expression	Comment
1	src_name		in_left.src_name	
2	src_gender		in_left.src_gender	
3	src_birth_date		in_left.src_birth_date	
4	src_sin		in_left.src_sin	
5	src_card		in_left.src_card	
6	src_address		in_left.src_address	
7	src_email		in_left.src_email	
8	src_primary_key		in_left.src_primary_key	
9	meta_last_update		in_left.meta_last_update	
10	src_tier		in_left.src_tier	
11	upper_name		in_left.upper_name	
12	sin_number		in_left.sin_number	
13	fix_email		in_left.fix_email	
14	client_tier_name		in_right.name	
*				

**Extract Filter** step:

Extract Filter

▼ General

Id:

▼ Other

```
client_tier_name = 'Platinum'
```

Condition\*:

**Condition** step:

Condition

▼ General

Id:

▼ Other

```
left(src_card,1)='4'
```

Condition\*:

**Alter Format** for Address:

Alter Format

● General

● Added Columns

● Removed Columns

Added Columns:

	Name	Type	Expression
1	contact_type	STRING	'ADDRESS'
2	contact_info	STRING	src_address
*			

**Alter Format** for Contact:

Alter Format

● General

● Added Columns

● Removed Columns

Added Columns:

	Name	Type	Expression
1	contact_type	STRING	'EMAIL'
2	contact_info	STRING	fix_email
*			