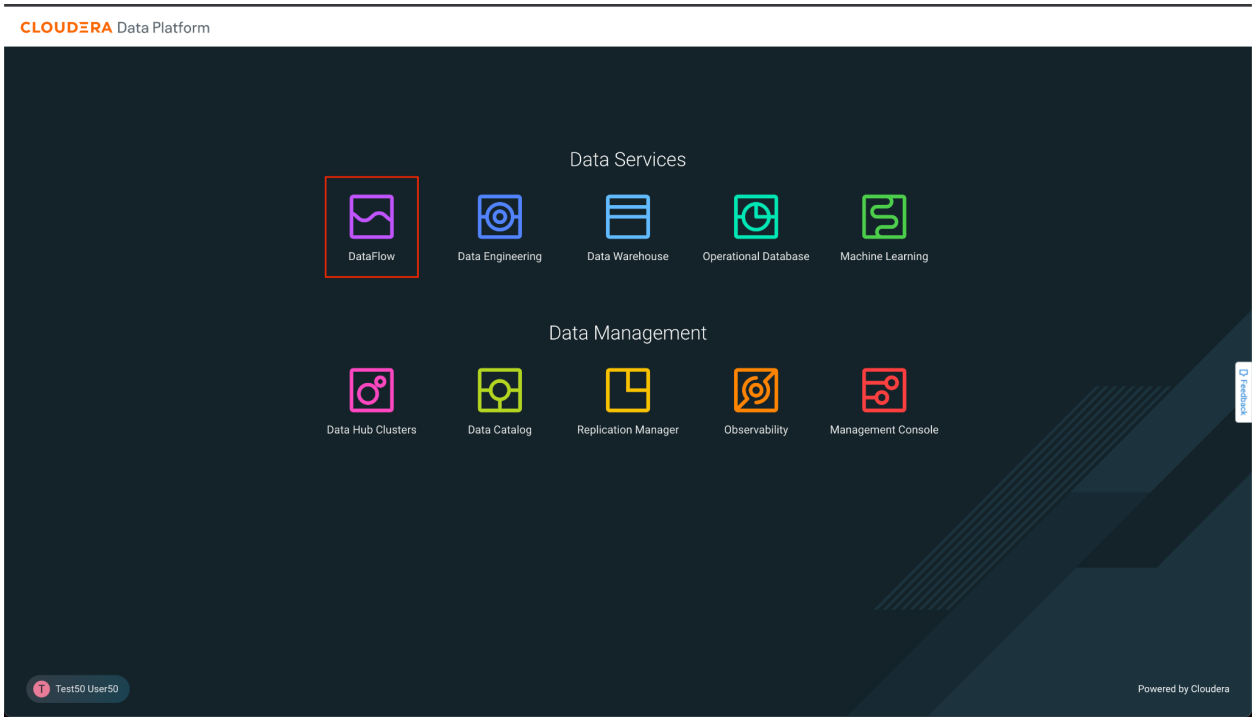
Data Lifecycle CDP Public Cloud

Data Flow Lab

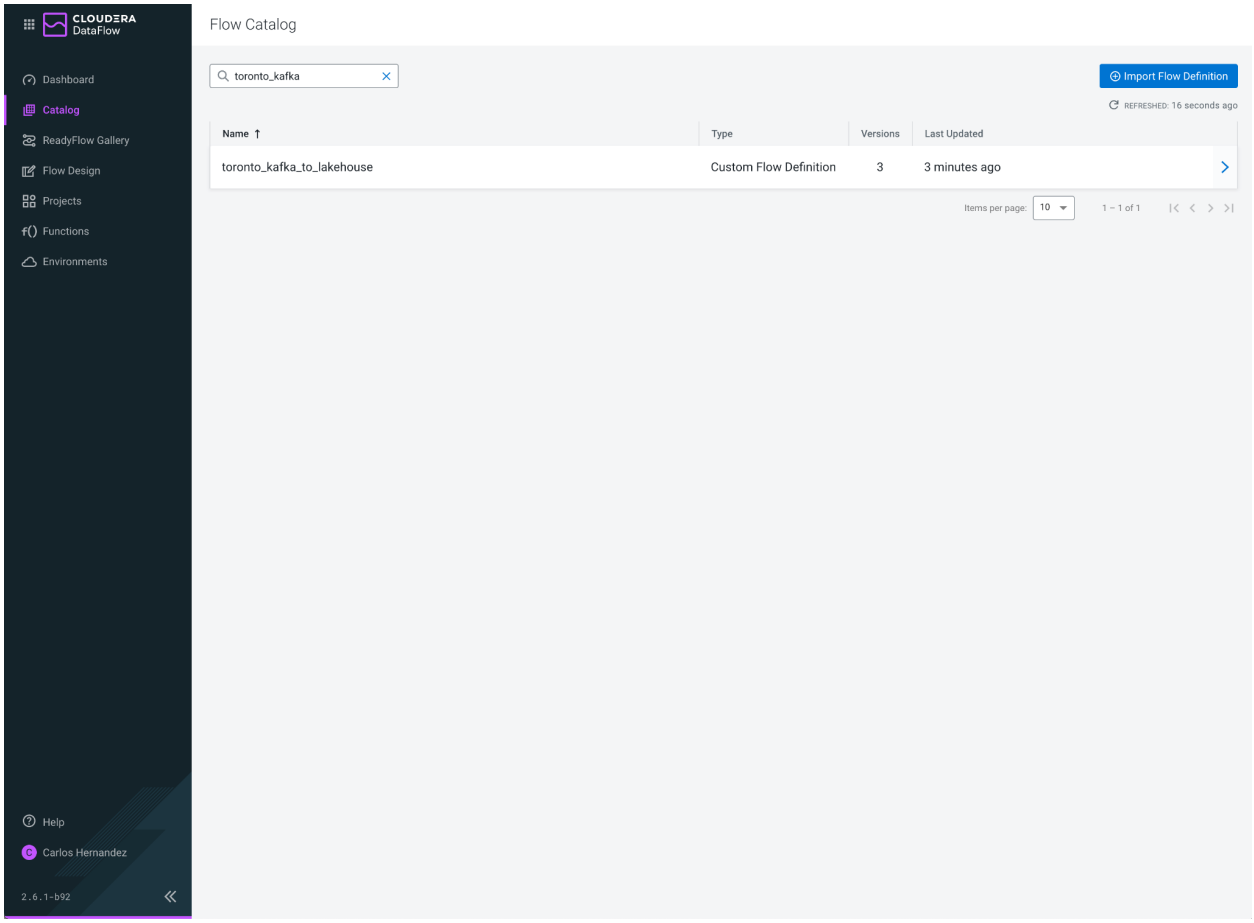
Goals:

- Consume data from a Kafka topic
- Convert the data to Parquet format
- Store the data in a table in the Lakehouse

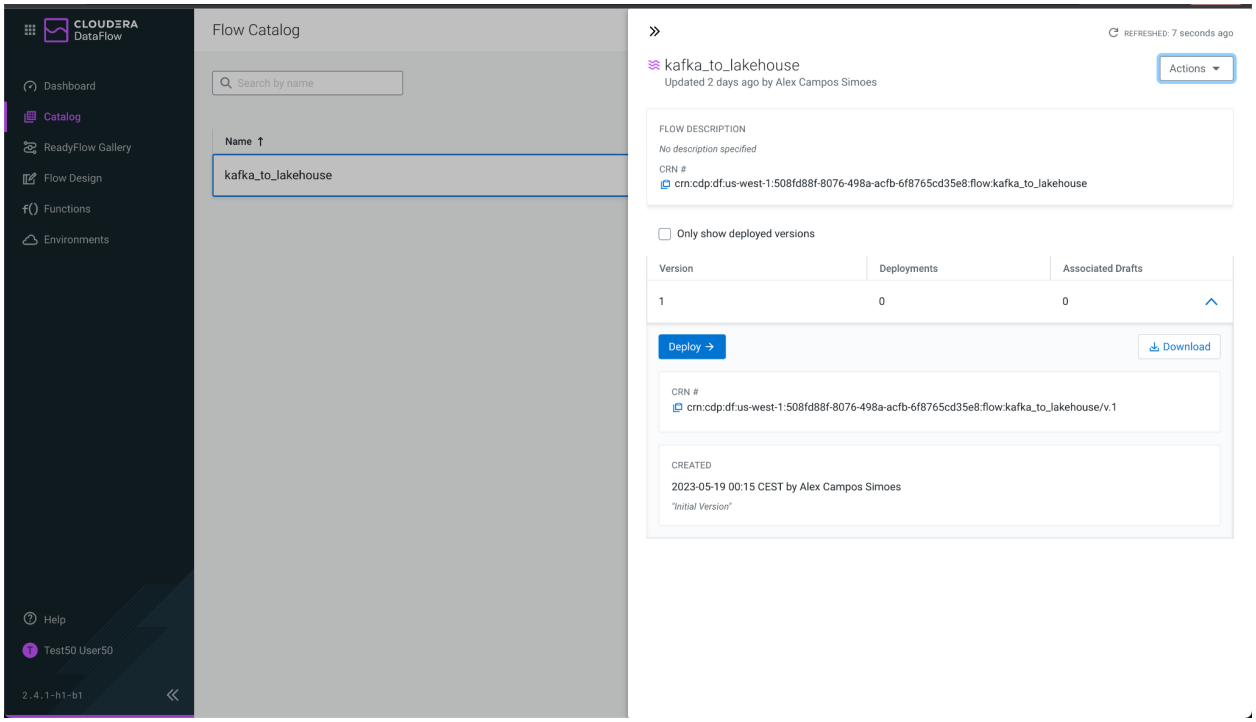
1. Click on DataFlow from CDP PC Home:



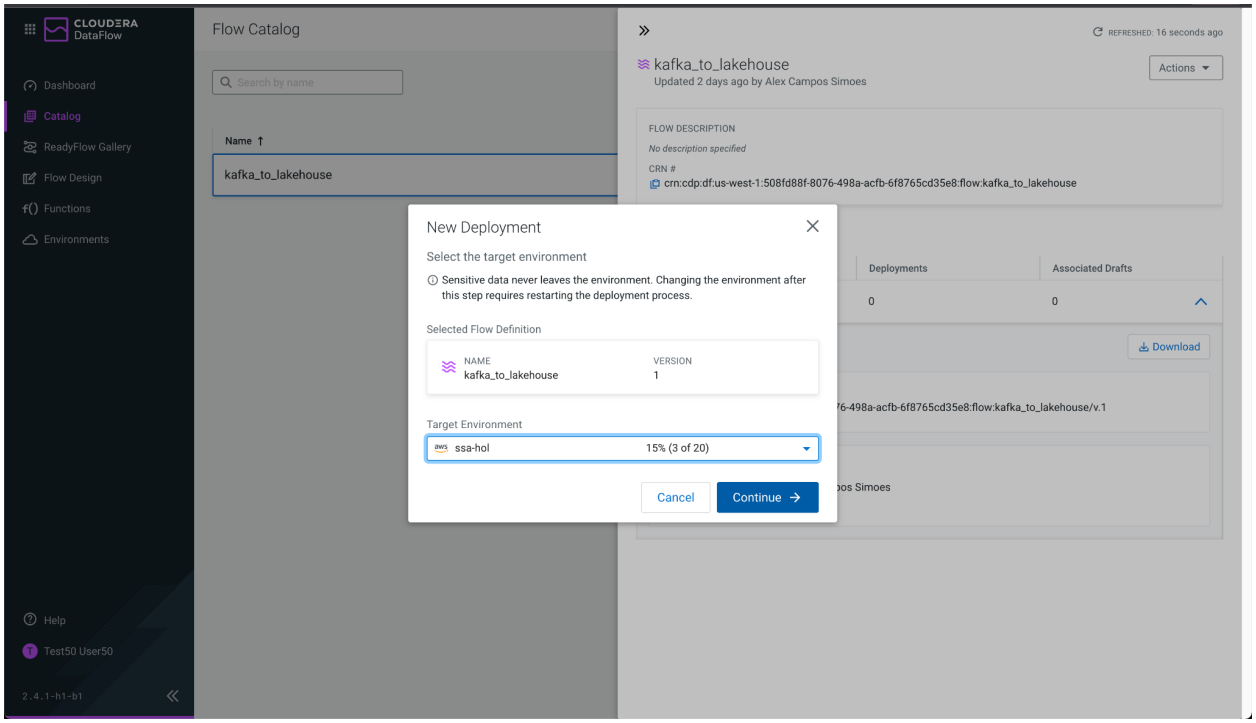
2. Once in DataFlow, click on the option **Catalog** from the left menu. The data ingestion application templates are listed here. For the purpose of this workshop, we have created and published a template that allows you to read Kafka topic data and ingest/store it in the Lakehouse provided by CDP Public Cloud. Click on the Flow called **toronto_kafka_to_lakehouse** to start deploying it.



3. When clicked, the following panel appears with the Flow information. It shows the available versions, creation date, creator user, and a button **Deploy** to start the deployment. Click on that button.



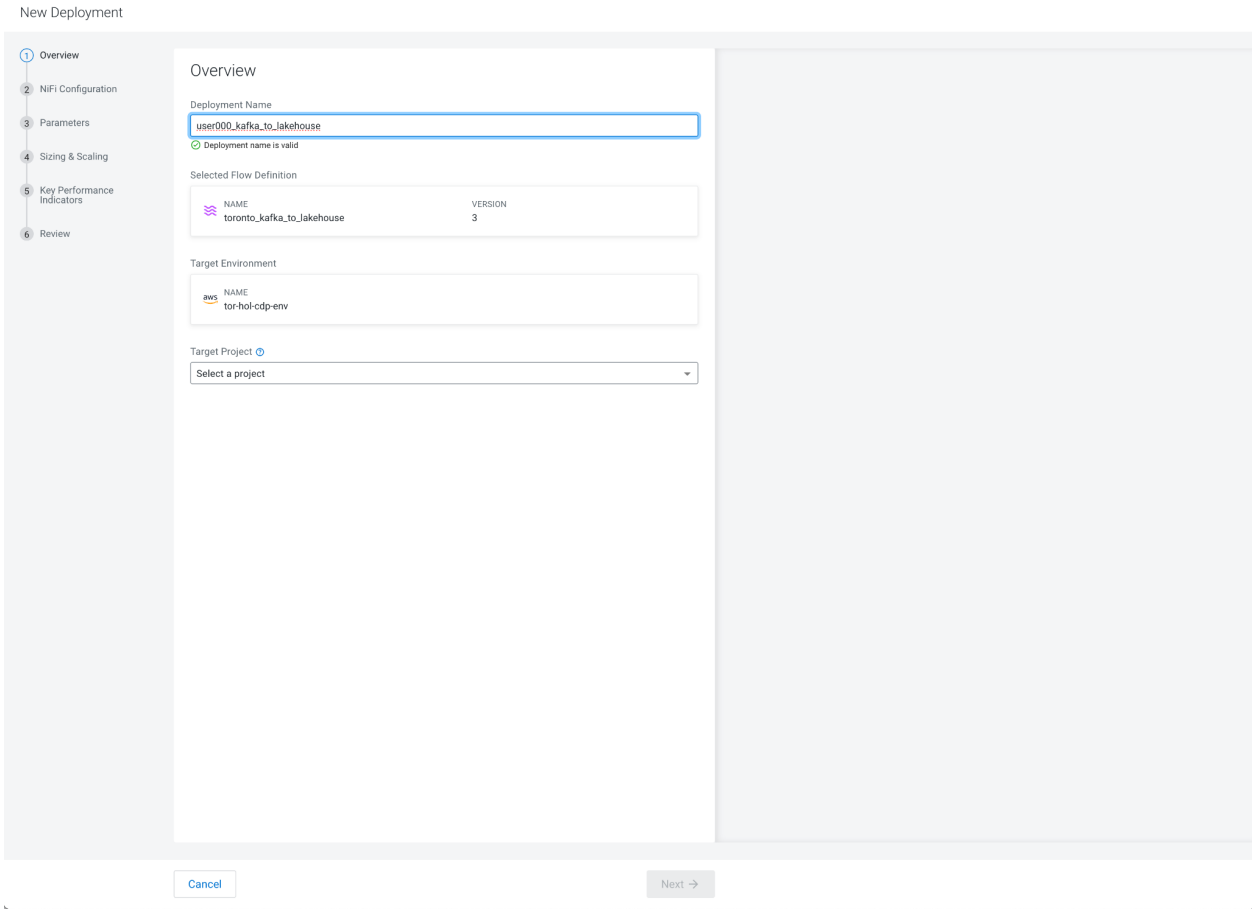
4. The following popup window allows you to select the DataFlow cluster in which you want to deploy the Flow. In this case, the cluster to be selected is **tor-hol-cdp-env**. The workshop instructor will tell you which environment to select. Once selected, click **Continue**.



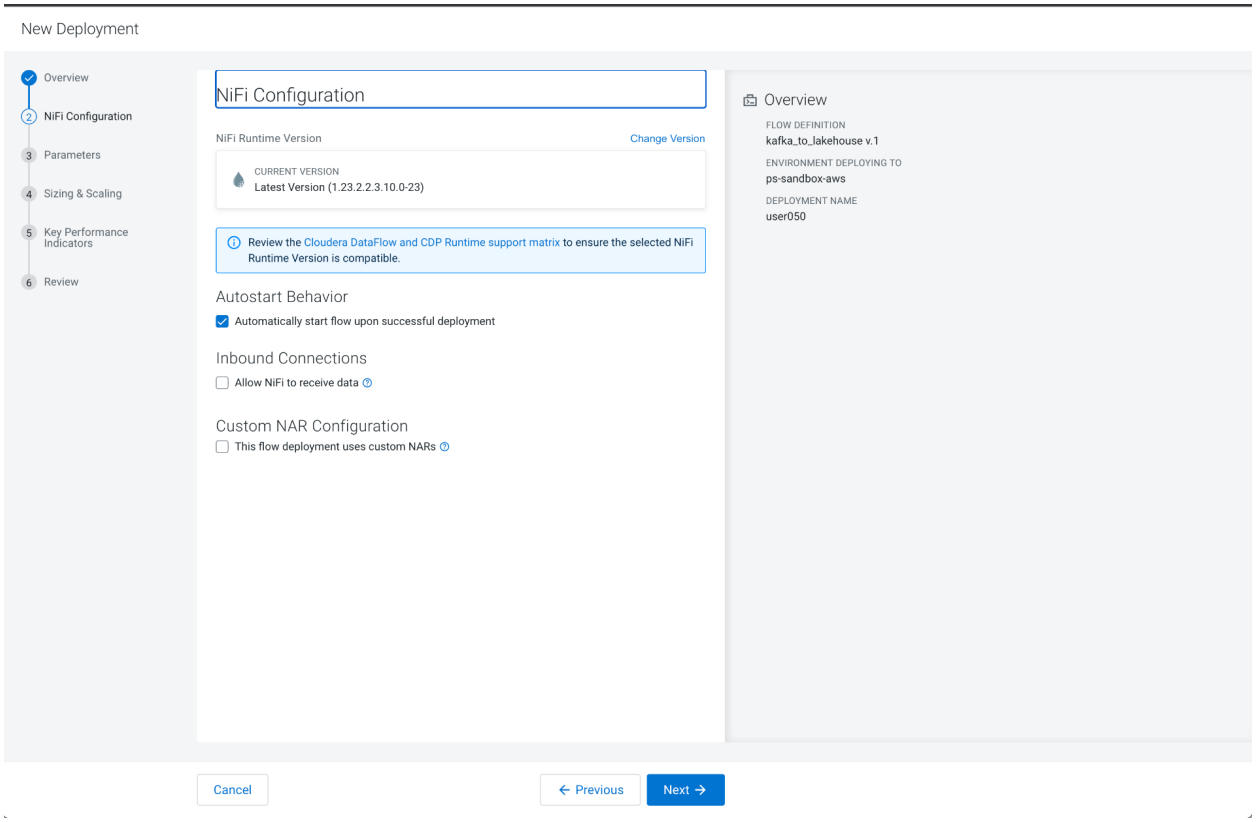
5. From this point, you will need to enter the Flow configuration. Start by assigning a **Deployment Name**, **Target Project**, and click **Next**.

For the purposes of this workshop, please name the Flow starting with your assigned username. For example, **user000**

Select **workshop** or **unassigned** for the project, which is a way to organize your flows.



6. Make sure the option **Automatically start flow upon successful deployment** is checked and click **Next.**



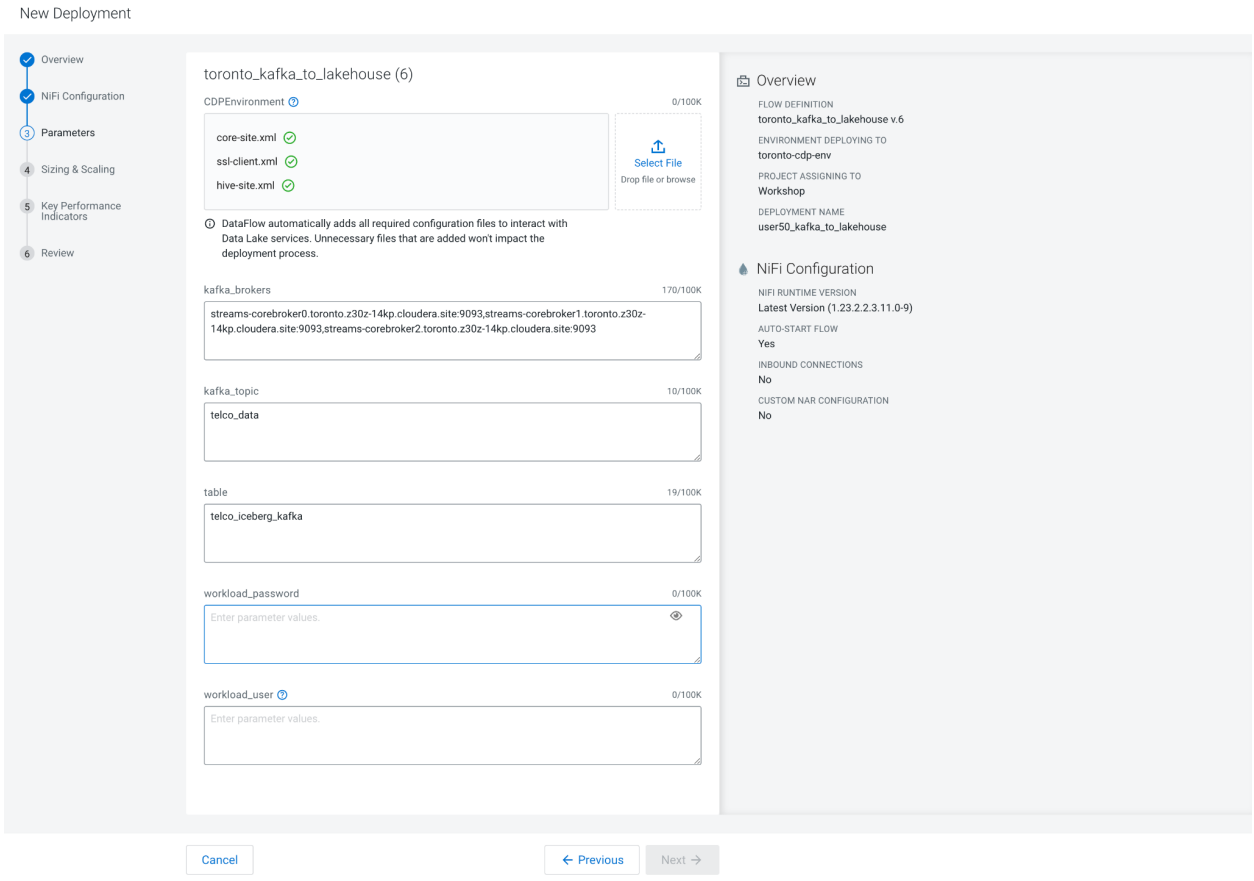
7. In this part of Parameters, you must enter the following values:

workload_password: Enter the Workload Password shared at the beginning of the workshop.
workload_user: enter the assigned user number, *user050*, for example.

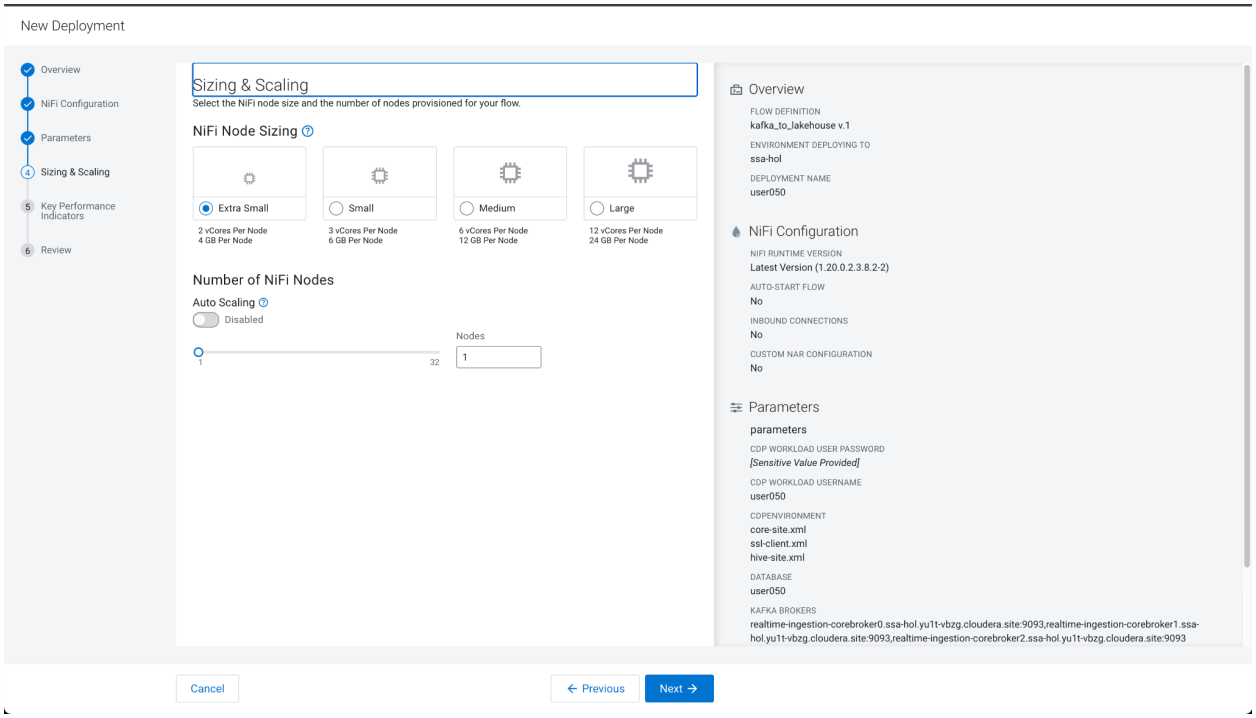
NOTE: for the purposes of the workshop, your user (e.g. user050) is also the name of the **database** where you will store the data (which has already been created for you), and the name of the **Kafka Consumer Group ID** for reading messages.

For the purposes of this workshop, the remaining values were filled out for you and don't need to change.

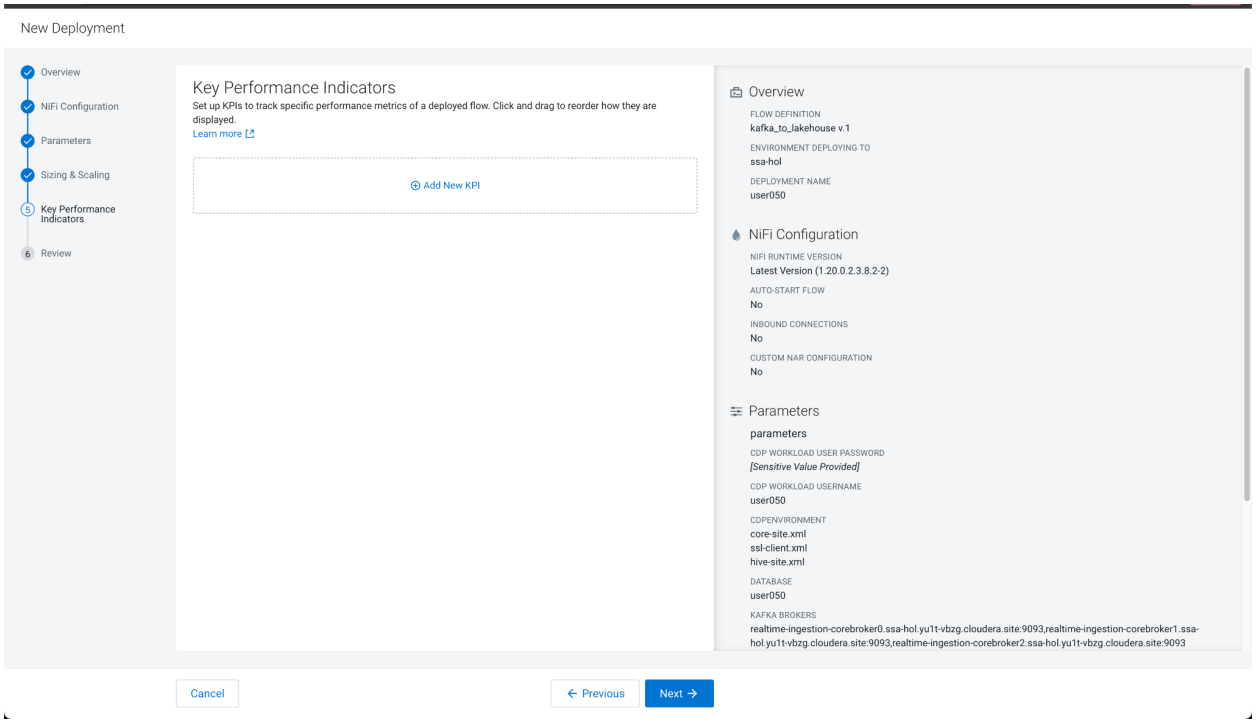
Review that the parameters were entered correctly. Then click **Next.**



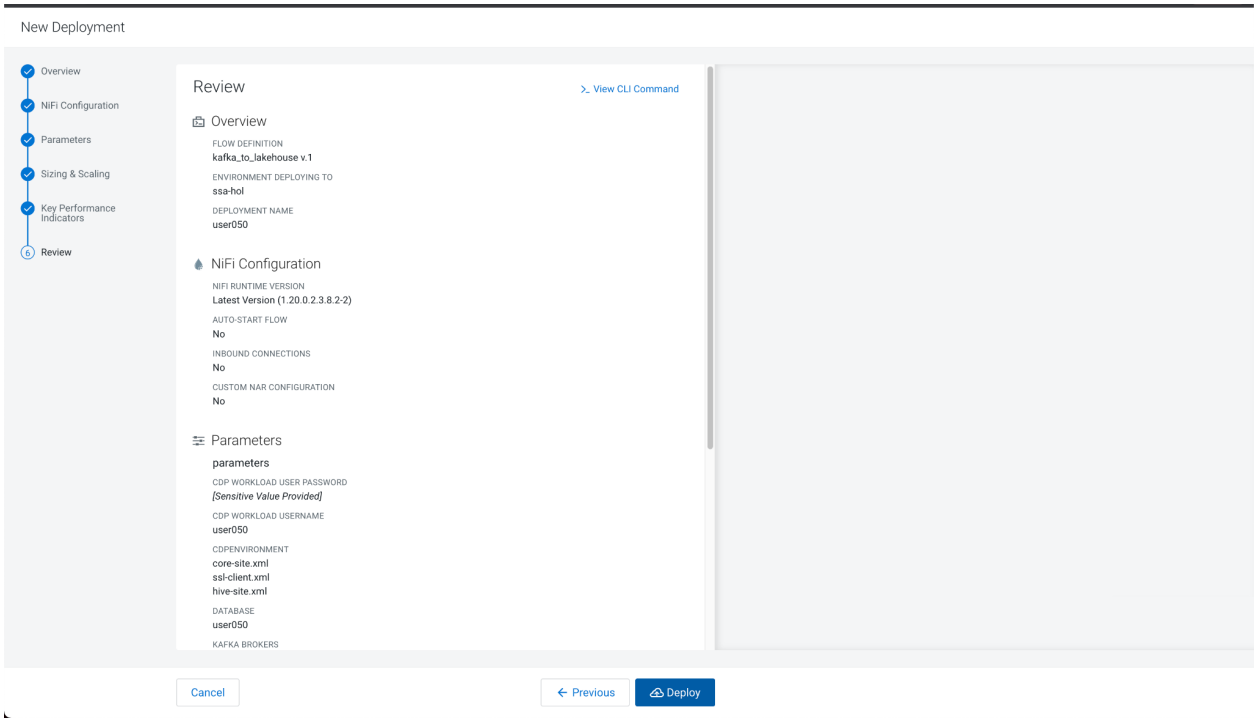
8. There is no need to configure auto-scaling parameters. Click **Next.**



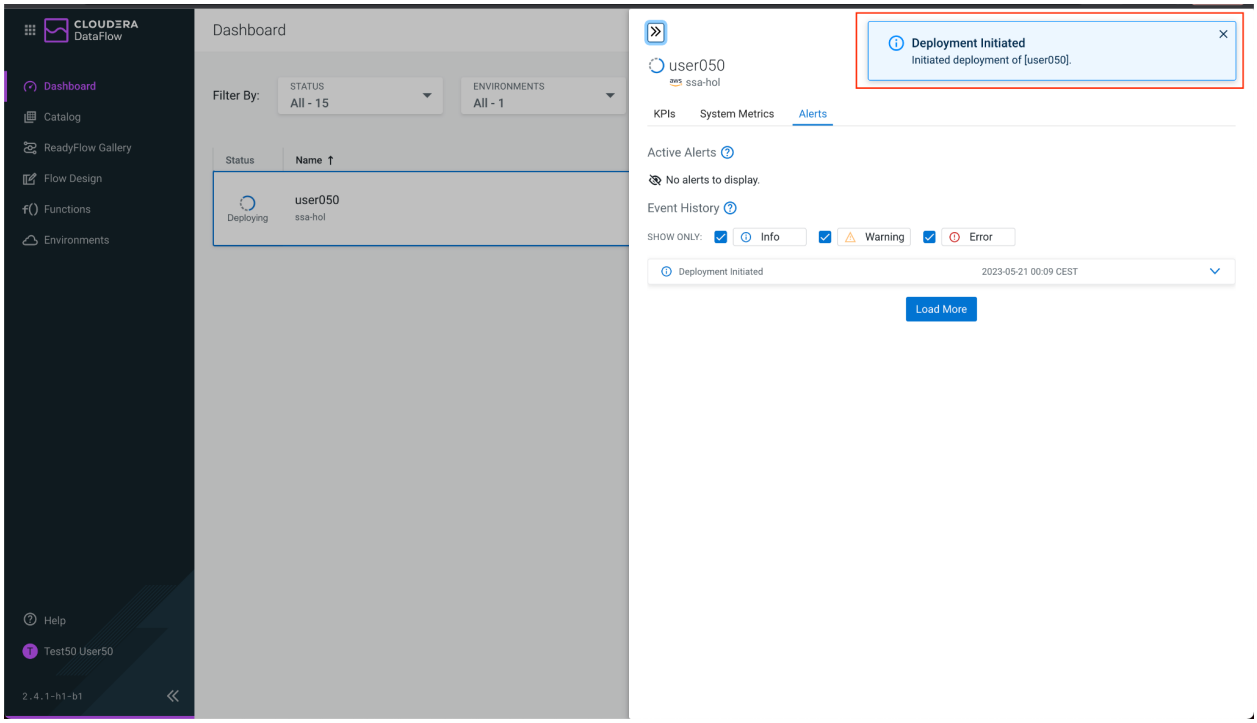
9. We are also not going to configure KPIs now. Click **Next** to continue the configuration.



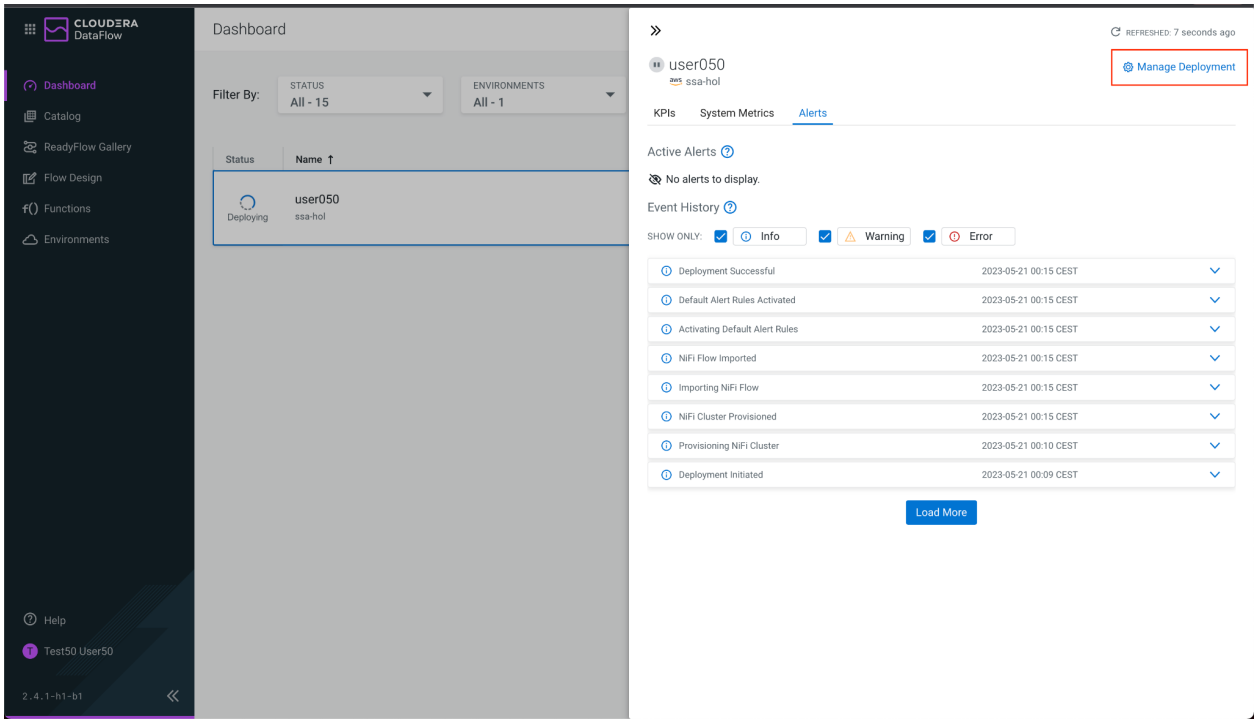
10. Review all the information entered for your Flow, then click on **Deploy** to start the deployment process.



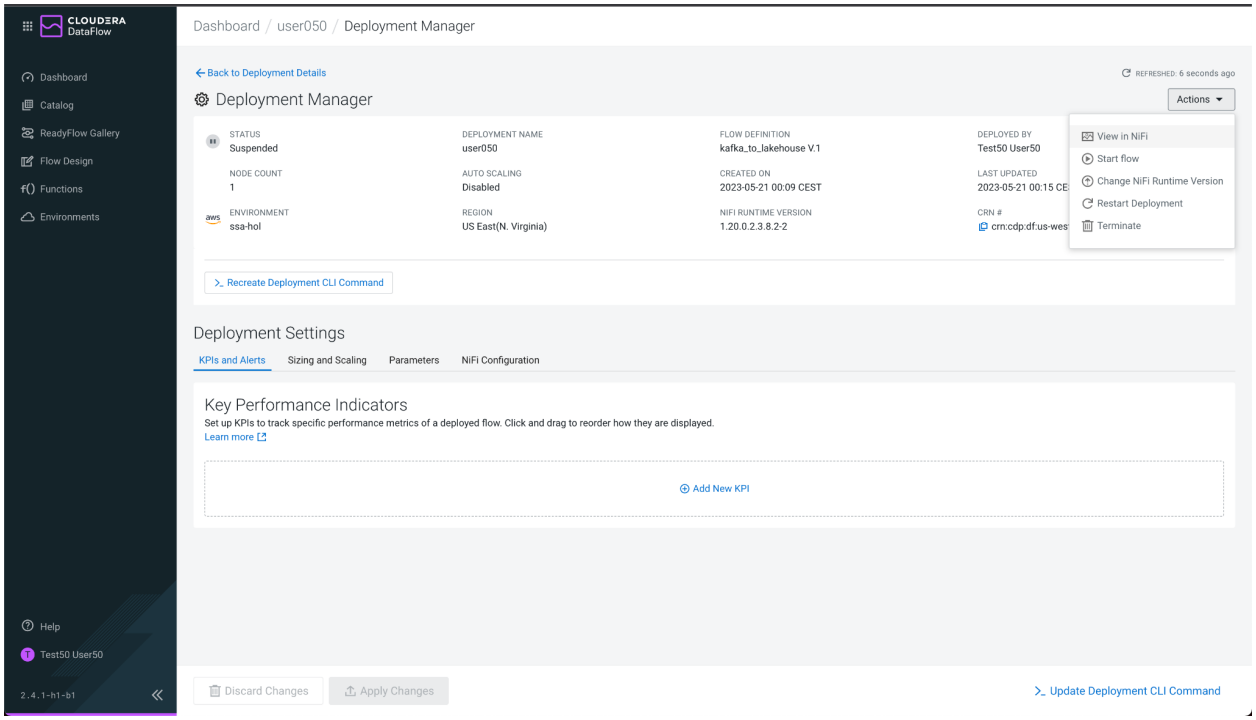
11. The blue box indicates that the Flow deployment process has been started. By clicking on the button **Load More** you will be able to see the different stages of the deployment. After about 60 to 90 seconds approximately, the last event should be *Deployment Successful*.



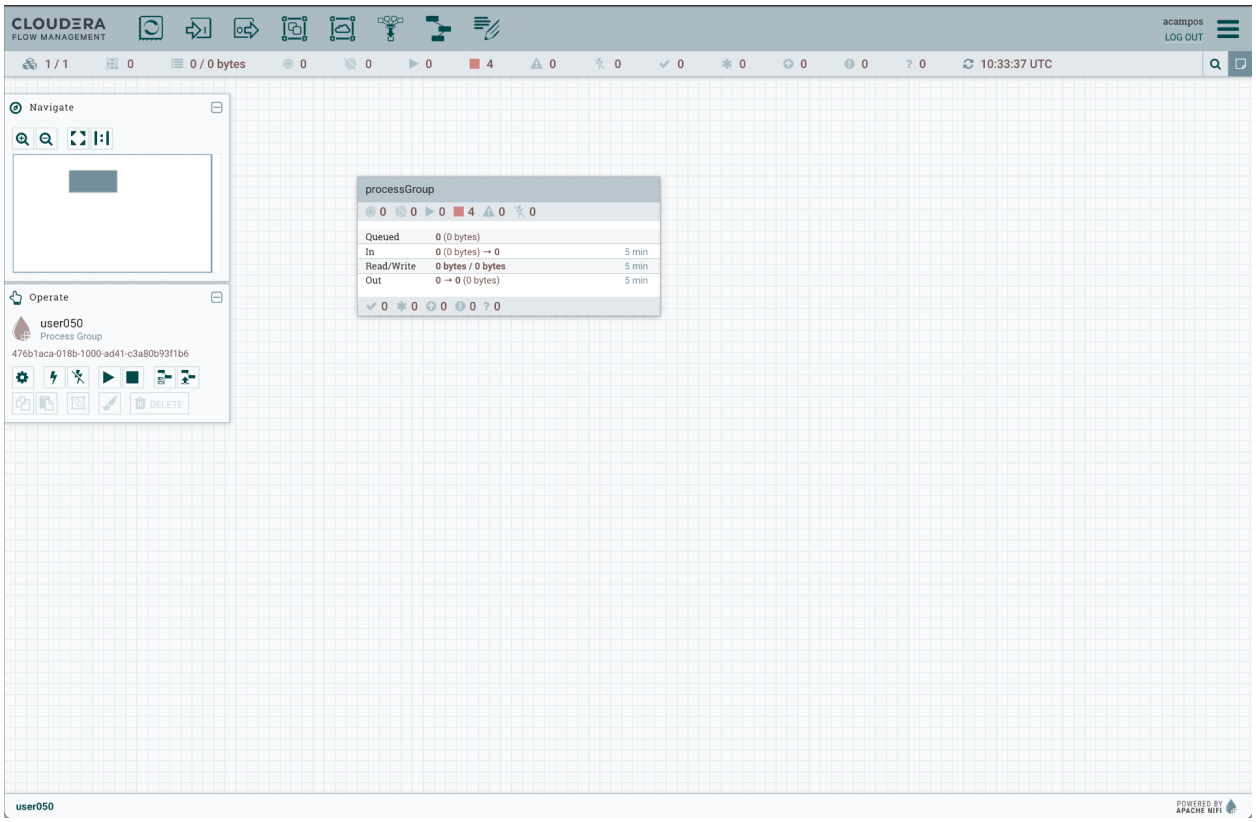
12. Once the deployment is finished, click on **Manage Deployment** to see the details of the recently deployed Flow.



13. In this window you will see the Flow information displayed. It is time to execute the application processes from the graphical Flow Management interface. Click on **Actions** -> **View in NiFi**, to open Cloudera Flow Management canvas in a new window/tab.



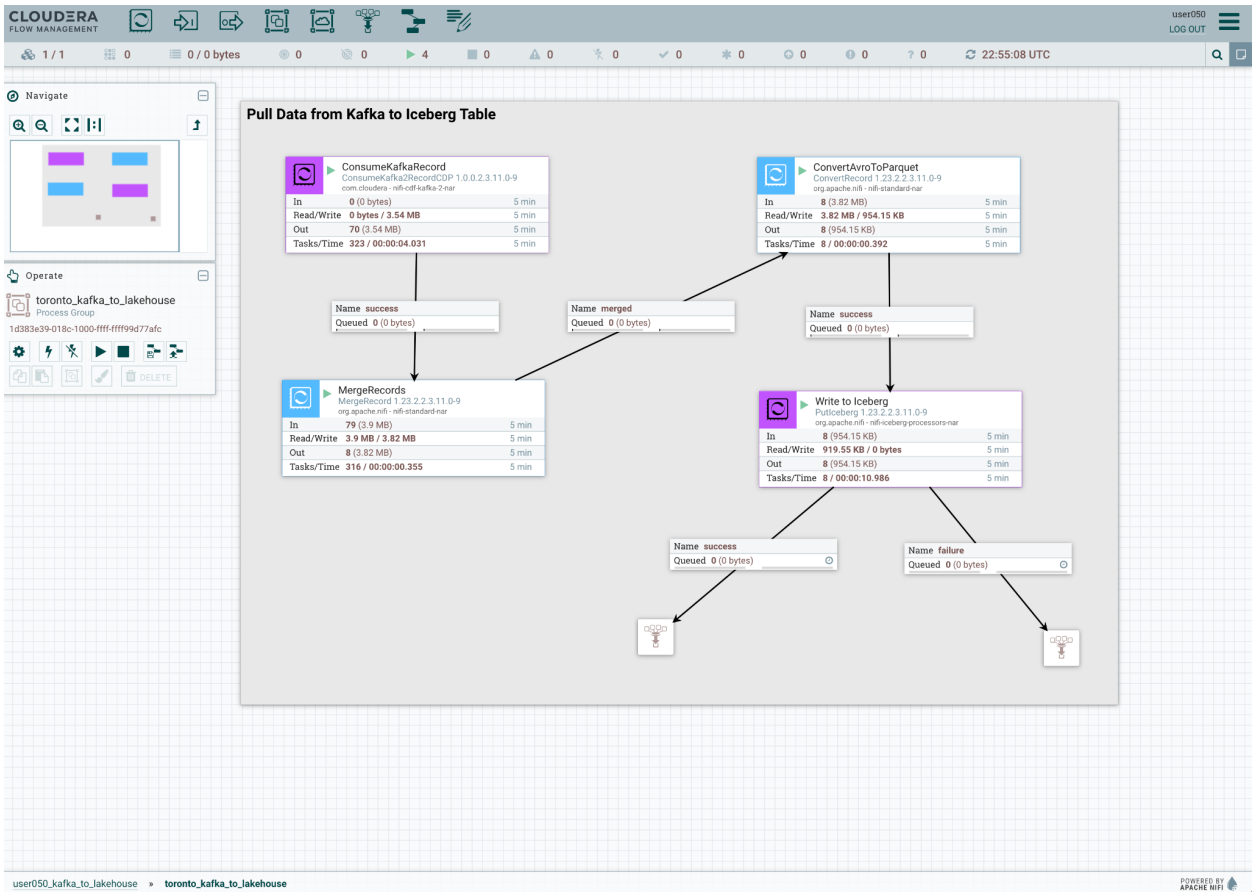
14. Double-click on the Process Group to open it.



16. When opening the Process Group, you should be able to see the Processors that compose the Flow application. To summarize, there are four Processors:

ConsumeKakfaRecord, consumes data from the Kafka topic, reading the data in JSON and outputting in AVRO.
MergeRecords, to group the flow files and streamline the data flow.
ConvertAvroToParquet, conversion needed to store the data in PARQUET format.
PutIceberg, to insert the data into the table in the Lakehouse. The destination table is called *telco_kafka_iceberg*, and each user has an assigned database (user_id is the name of the database).

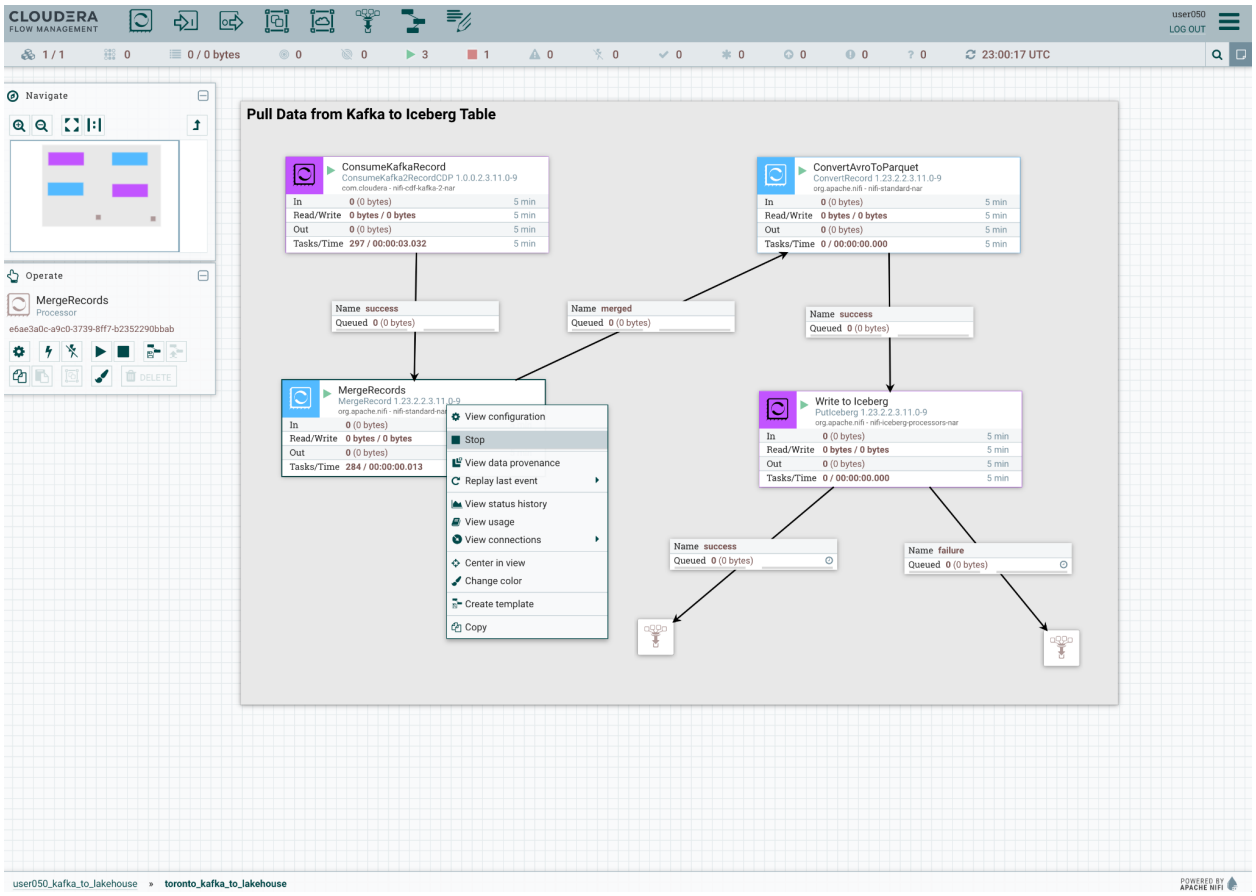
As you can see, the Processors are not started, they are paused.



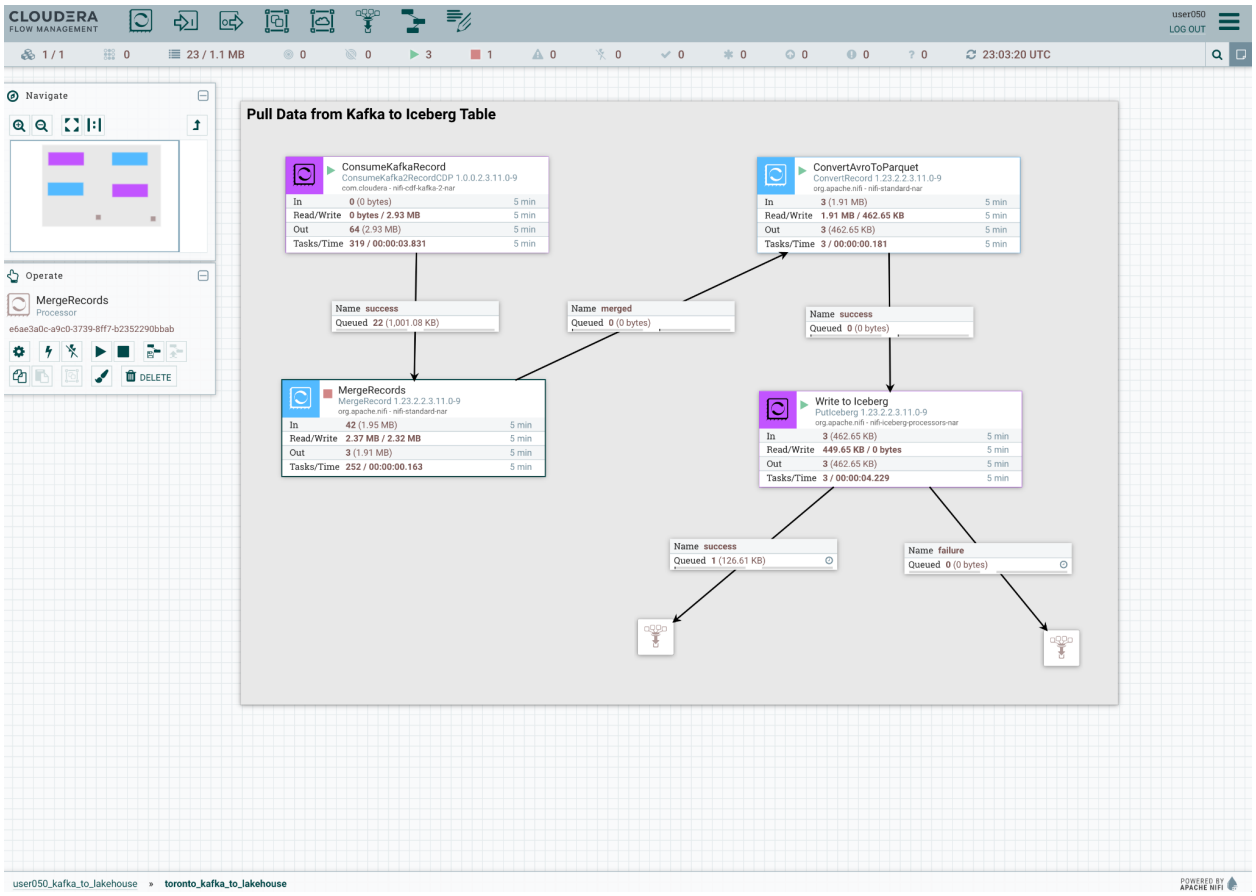
From the Out field in every processor, you can see that data has flowed through in the past 5 minutes. You have already consumed data from Kafka and to Iceberg!

17. Flow Management allows us to see and access data in motion during the execution of the data flow. Between Processors **ConsumeKafkaRecord** (just started) and **MergeRecords**, there is a connection. This connection is what joins the Processors and transmits data from one to the other, and you can check how much data is queued at every step of the process.

Let’s see this in action by building up the queue. First, right-click on **MergeRecords** processor and click **Stop.**

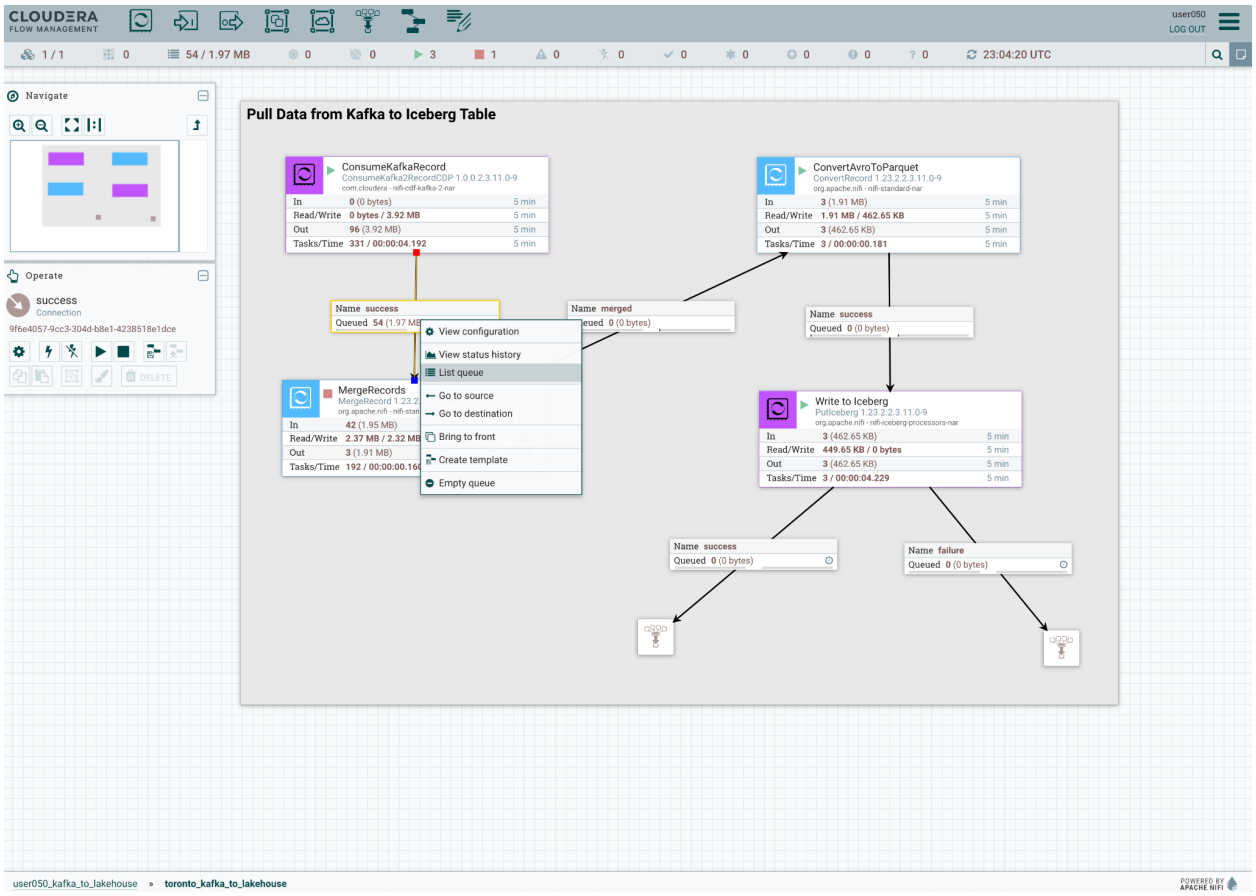


18. You will see data start to queue up in the connector shortly after stopping the MergeRecords processor.

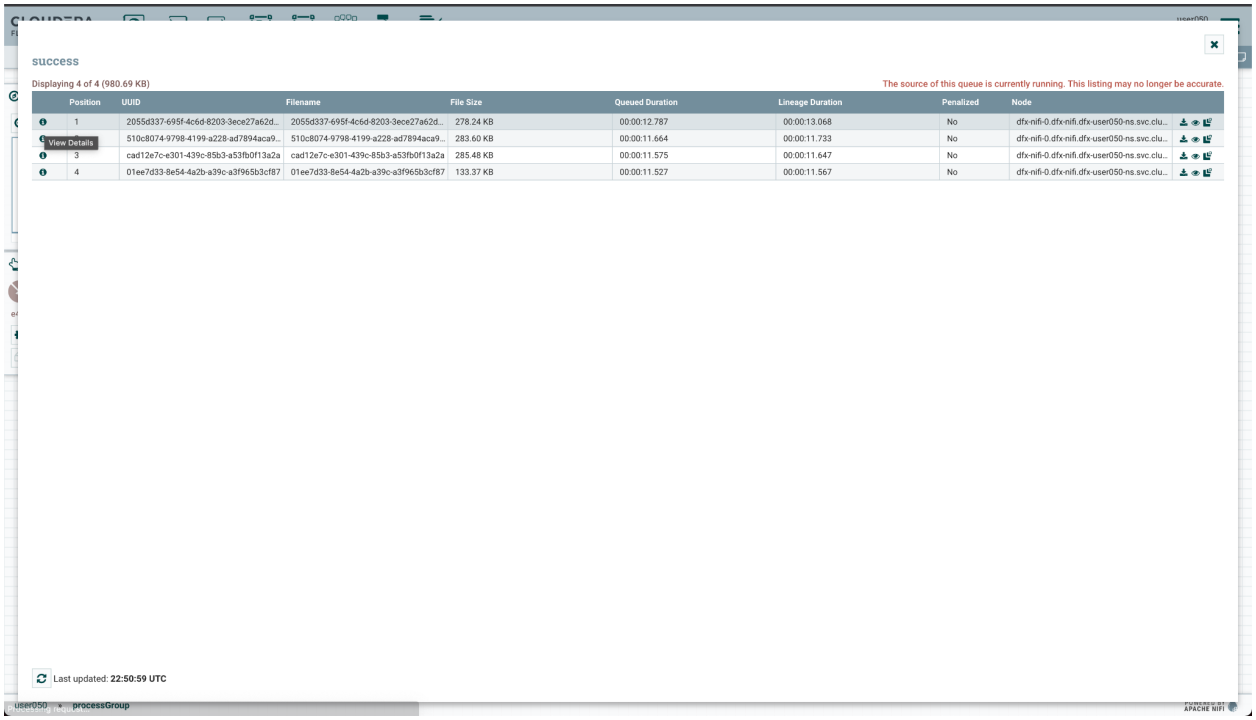


You can refresh the counter by pressing the Ctrl+R (Windows) or Command+R (Mac) combination on the keyboard.

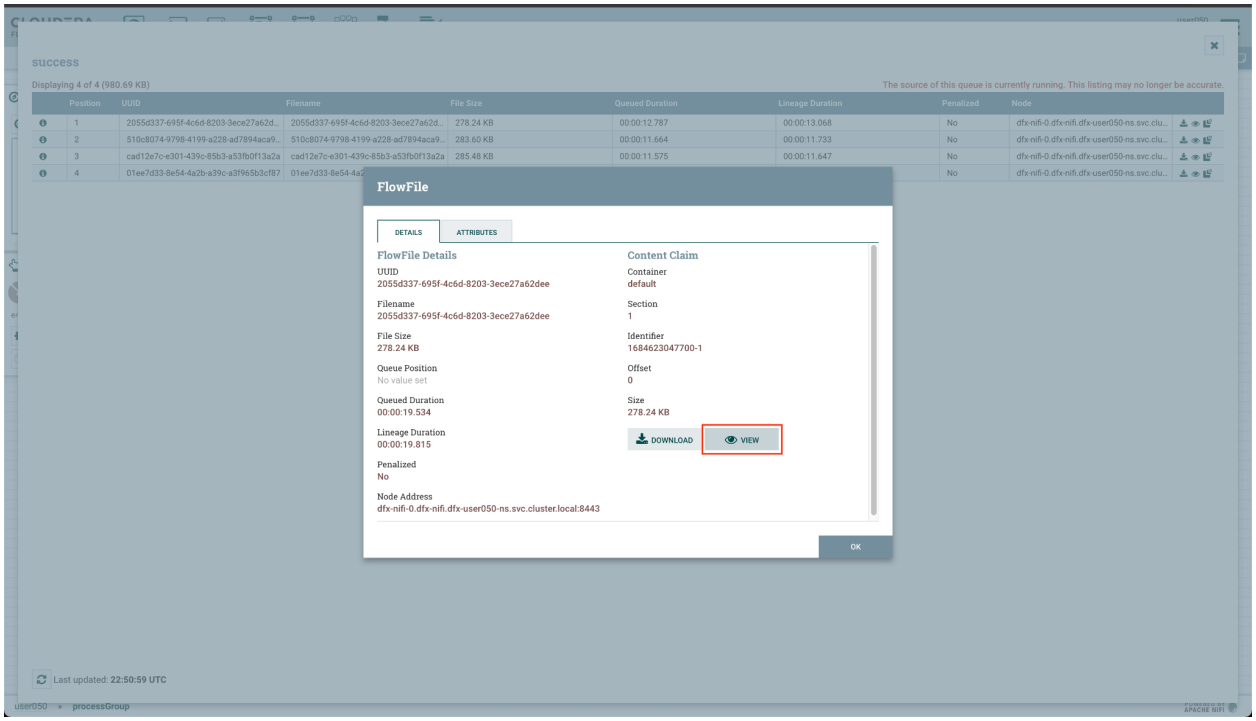
This will allow the current metrics of the entire data stream to be updated. At some point there should be a number next to the legend **Queued** in the connection between **ConsumeKafkaRecord** and **MergeRecords.** To see the queued data, right-click on the connection and click on the option **List Queue**, opening a popup window.



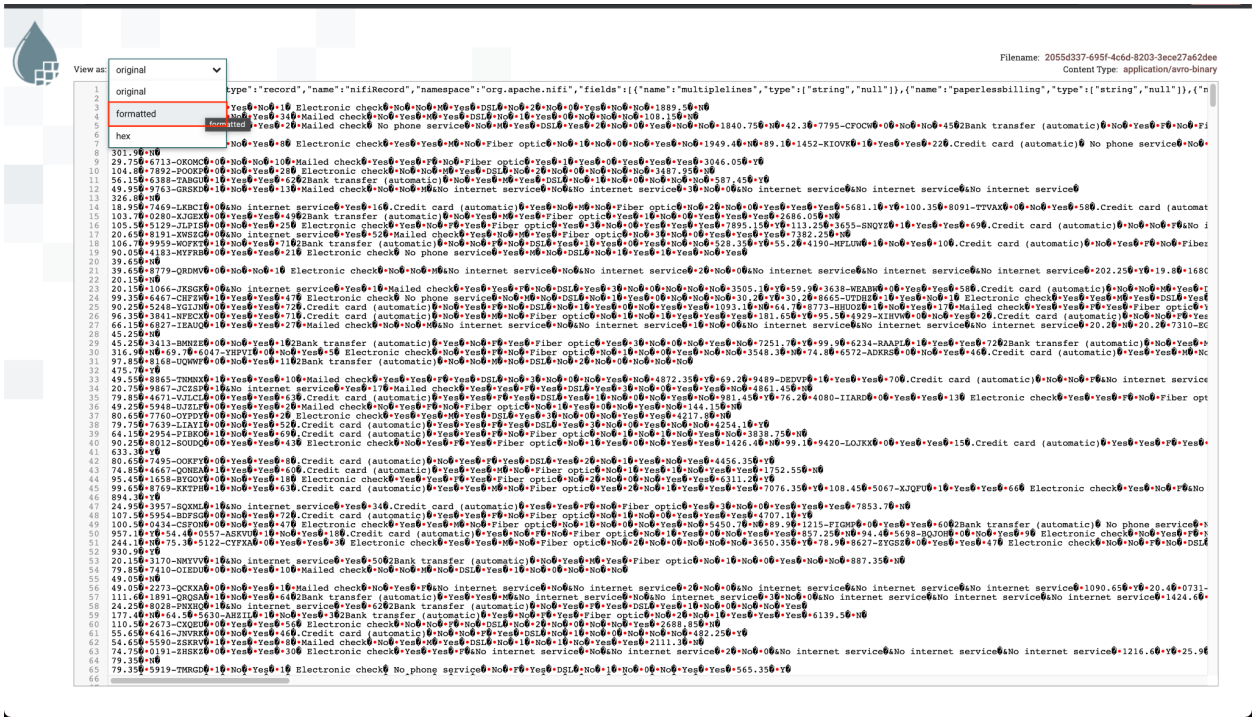
19. The next popup window lists the queued data. Click on the information icon (i) that appears on the left side to view the events.



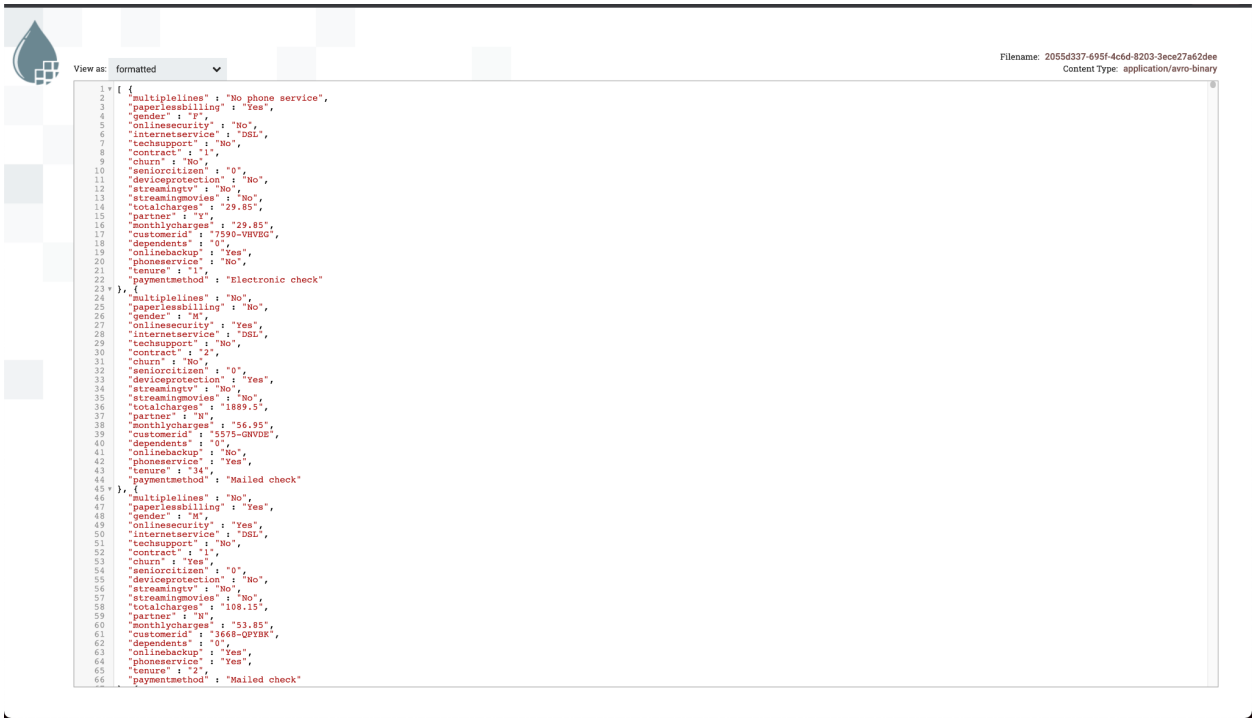
20. Once the FlowFile detail window appears, click on the button **VIEW** to open the content of consumed events.



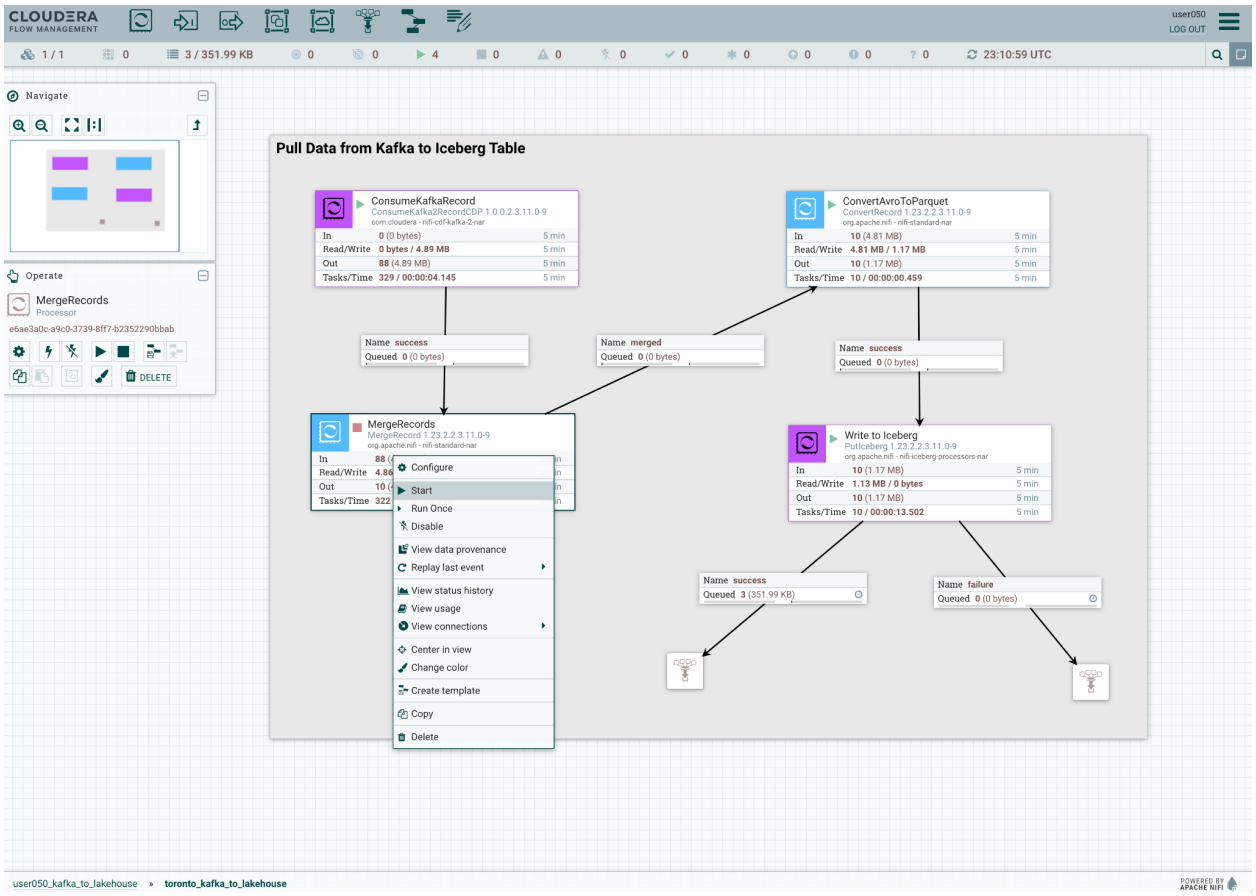
21. The new window that opens shows the data of the FlowFile content. Being in AVRO format, it is not fully readable. A deserializer must be selected to correctly display the data. For this, in the upper left, select the option **formatted** from the menu **View as**.



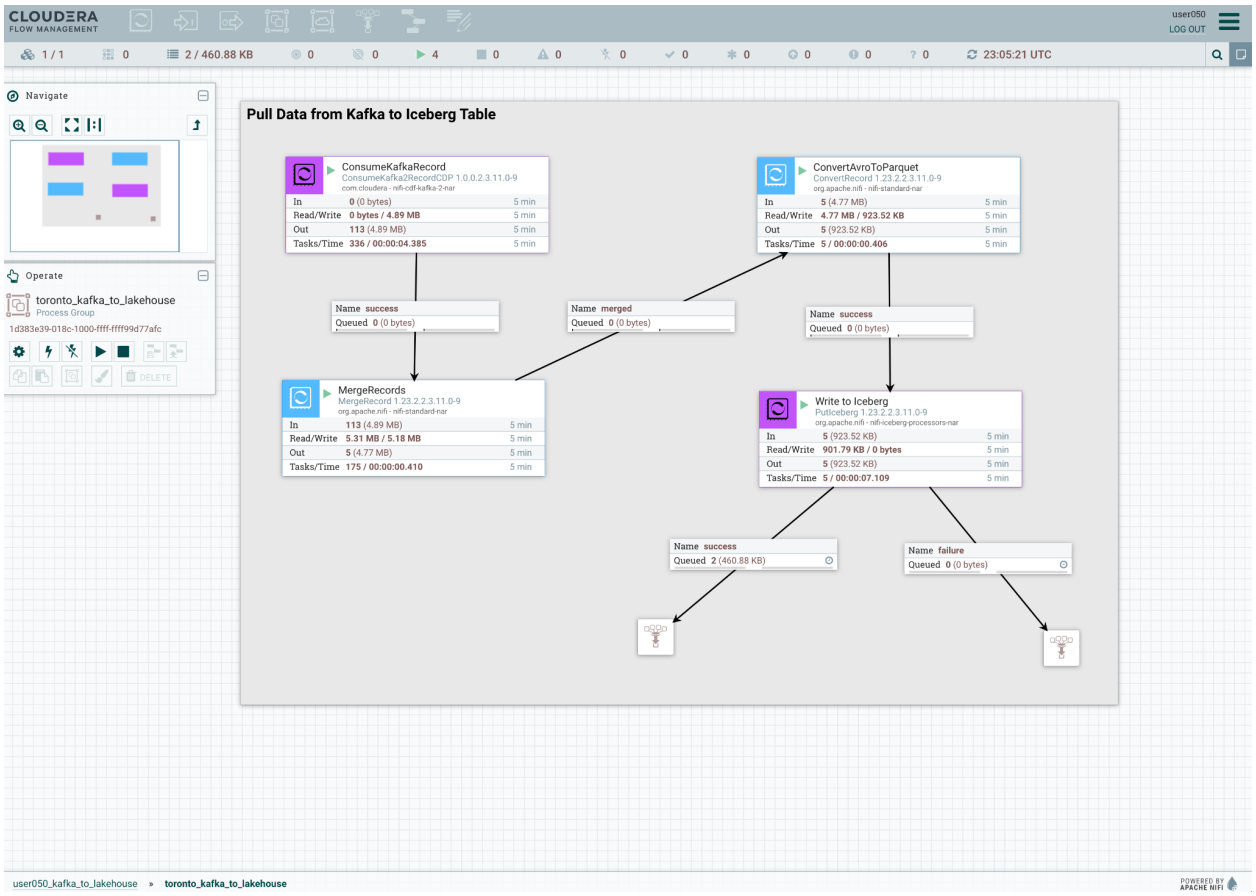
22. Now you can display the data correctly. Notice that the fields or attributes indicated at the beginning of the workshop appear. You can close that FlowFile window and the popups, returning to the canvas with the four Processors.



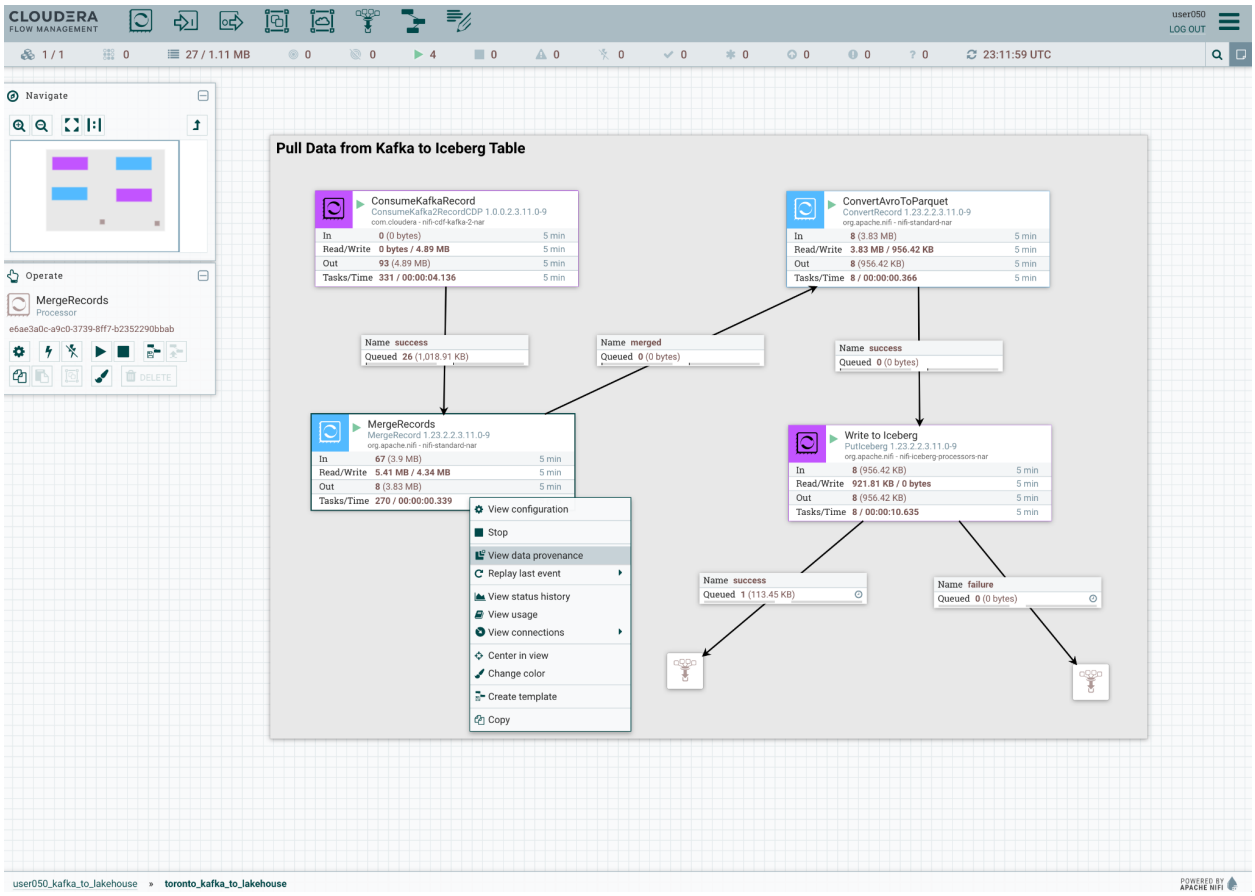
23. Start the stopped: **MergeRecords** processor again to resume the flow. Remember that you can refresh the flow counters with the combination Control+R or Command+R.

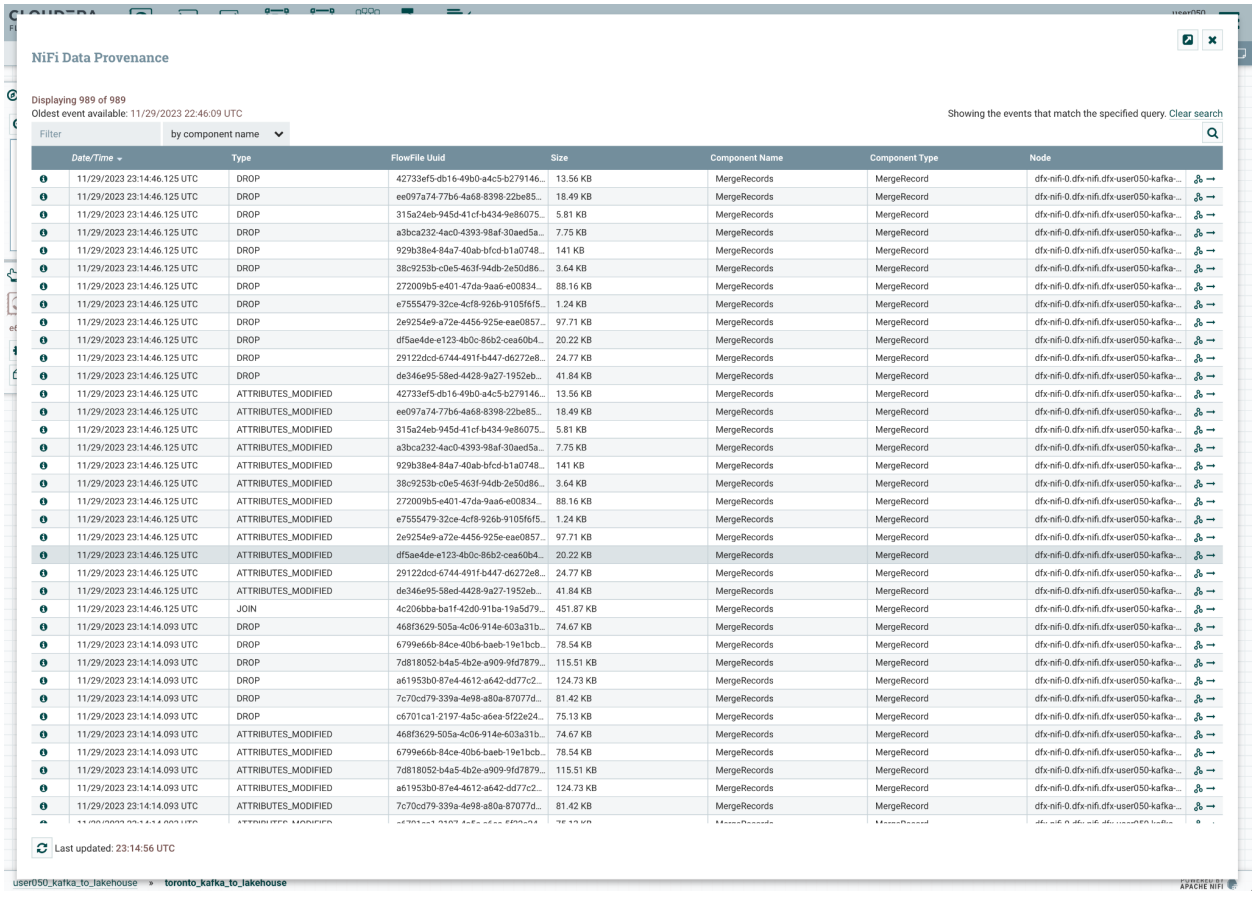


If the previous steps were executed correctly, the connection of the Processor **PutIceberg** to a funnel should be of type **success**.



24. BONUS: NiFi is a powerful ingestion tool that gives you granular visibility into everything that's done to the data - for example, right-click on any processor and then click on **View data provenance** to see this in action


CLOUDERA
FLOW MANAGEMENT
user050
LOG OUT
1 / 1
0
27 / 1.11 MB
0
0
4
0
0
0
0
0
0
0
23:11:59 UTC
Navigate
Operate
MergeRecords
Processor
e6ae3a0c-a9c0-3739-8ff7-b2352290bbab
DELETE
Pull Data from Kafka to Iceberg Table
ConsumeKafkaRecord
ConsumeKafka2RecordCDP 1.0.0.2.3.11.0-9
com.cloudera - nifi-cdf-kafka-2-nar
In 0 (0 bytes) 5 min
Read/Write 0 bytes / 4.89 MB 5 min
Out 93 (4.89 MB) 5 min
Tasks/Time 331 / 00:00:04.136 5 min
ConvertAvroToParquet
ConvertRecord 1.23.2.2.3.11.0-9
org.apache.nifi - nifi-standard-nar
In 8 (3.83 MB) 5 min
Read/Write 3.83 MB / 956.42 KB 5 min
Out 8 (956.42 KB) 5 min
Tasks/Time 8 / 00:00:00.366 5 min
Name success
Queued 26 (1,018.91 KB)
Name merged
Queued 0 (0 bytes)
Name success
Queued 0 (0 bytes)
MergeRecords
MergeRecord 1.23.2.2.3.11.0-9
org.apache.nifi - nifi-standard-nar
In 67 (3.9 MB) 5 min
Read/Write 5.41 MB / 4.34 MB 5 min
Out 8 (3.83 MB) 5 min
Tasks/Time 270 / 00:00:00.339 5 min
Write to Iceberg
PutIceberg 1.23.2.2.3.11.0-9
org.apache.nifi - nifi-iceberg-processors-nar
In 8 (956.42 KB) 5 min
Read/Write 921.81 KB / 0 bytes 5 min
Out 8 (956.42 KB) 5 min
Tasks/Time 8 / 00:00:10.635 5 min
View configuration
Stop
View data provenance
Replay last event
View status history
View usage
View connections
Center in view
Change color
Create template
Copy
Name success
Queued 1 (113.45 KB)
Name failure
Queued 0 (0 bytes)
user050_kafka_to_lakehouse
»
toronto_kafka_to_lakehouse
POWERED BY
APACHE NIFI



NiFi Data Provenance

Displaying 989 of 989
Oldest event available: 11/29/2023 22:46:09 UTC

Showing the events that match the specified query. Clear search

Filter | by component name

Date/Time	Type	FlowFile Uuid	Size	Component Name	Component Type	Node
11/29/2023 23:14:46.125 UTC	DROP	42733ef5-db16-49b0-a4c5-b279146...	13.56 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	DROP	ee097a74-77b6-4a68-8398-22be85...	18.49 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	DROP	315a24eb-945d-41cf-b434-9e86075...	5.81 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	DROP	a3bca232-4ac0-4393-98af-30aed5a...	7.75 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	DROP	929b38e4-84a7-40ab-bfcd-b1a0748...	141 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	DROP	38c9253b-c0e5-463f-94db-2e50d86...	3.64 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	DROP	272009b5-e401-47da-9aa6-e00834...	88.16 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	DROP	e7555479-32ce-4cf8-926b-9105f6f5...	1.24 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	DROP	2e9254e9-a72e-4456-925e-eae0857...	97.71 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	DROP	df5ae4de-e123-4b0c-86b2-cea60b4...	20.22 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	DROP	29122dcd-6744-491f-b447-d6272e8...	24.77 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	DROP	de346e95-58ed-4428-9a27-1952eb...	41.84 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	ATTRIBUTES_MODIFIED	42733ef5-db16-49b0-a4c5-b279146...	13.56 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	ATTRIBUTES_MODIFIED	ee097a74-77b6-4a68-8398-22be85...	18.49 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	ATTRIBUTES_MODIFIED	315a24eb-945d-41cf-b434-9e86075...	5.81 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	ATTRIBUTES_MODIFIED	a3bca232-4ac0-4393-98af-30aed5a...	7.75 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	ATTRIBUTES_MODIFIED	929b38e4-84a7-40ab-bfcd-b1a0748...	141 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	ATTRIBUTES_MODIFIED	38c9253b-c0e5-463f-94db-2e50d86...	3.64 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	ATTRIBUTES_MODIFIED	272009b5-e401-47da-9aa6-e00834...	88.16 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	ATTRIBUTES_MODIFIED	e7555479-32ce-4cf8-926b-9105f6f5...	1.24 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	ATTRIBUTES_MODIFIED	2e9254e9-a72e-4456-925e-eae0857...	97.71 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	ATTRIBUTES_MODIFIED	df5ae4de-e123-4b0c-86b2-cea60b4...	20.22 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	ATTRIBUTES_MODIFIED	29122dcd-6744-491f-b447-d6272e8...	24.77 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	ATTRIBUTES_MODIFIED	de346e95-58ed-4428-9a27-1952eb...	41.84 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:46.125 UTC	JOIN	4c206bba-ba1f-42d0-91ba-19a5d79...	451.87 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:14.093 UTC	DROP	468f3629-505a-4c06-914e-603a31b...	74.67 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:14.093 UTC	DROP	6799e66b-84ce-40b6-baeb-19e1bcb...	78.54 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:14.093 UTC	DROP	7d818052-b4a5-4b2e-a909-9fd7879...	115.51 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:14.093 UTC	DROP	a61953b0-87e4-4612-a642-dd77c2...	124.73 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:14.093 UTC	DROP	7c70cd79-339a-4e98-a80a-87077d...	81.42 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:14.093 UTC	DROP	c6701ca1-2197-4a5c-a6ea-5f22e24...	75.13 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:14.093 UTC	ATTRIBUTES_MODIFIED	468f3629-505a-4c06-914e-603a31b...	74.67 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:14.093 UTC	ATTRIBUTES_MODIFIED	6799e66b-84ce-40b6-baeb-19e1bcb...	78.54 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:14.093 UTC	ATTRIBUTES_MODIFIED	7d818052-b4a5-4b2e-a909-9fd7879...	115.51 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:14.093 UTC	ATTRIBUTES_MODIFIED	a61953b0-87e4-4612-a642-dd77c2...	124.73 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...
11/29/2023 23:14:14.093 UTC	ATTRIBUTES_MODIFIED	7c70cd79-339a-4e98-a80a-87077d...	81.42 KB	MergeRecords	MergeRecord	dfx-nifi-0.dfx-nifi.dfx-user050-kafka-...

Last updated: 23:14:56 UTC

user050_kafka_to_lakehouse » toronto_kafka_to_lakehouse