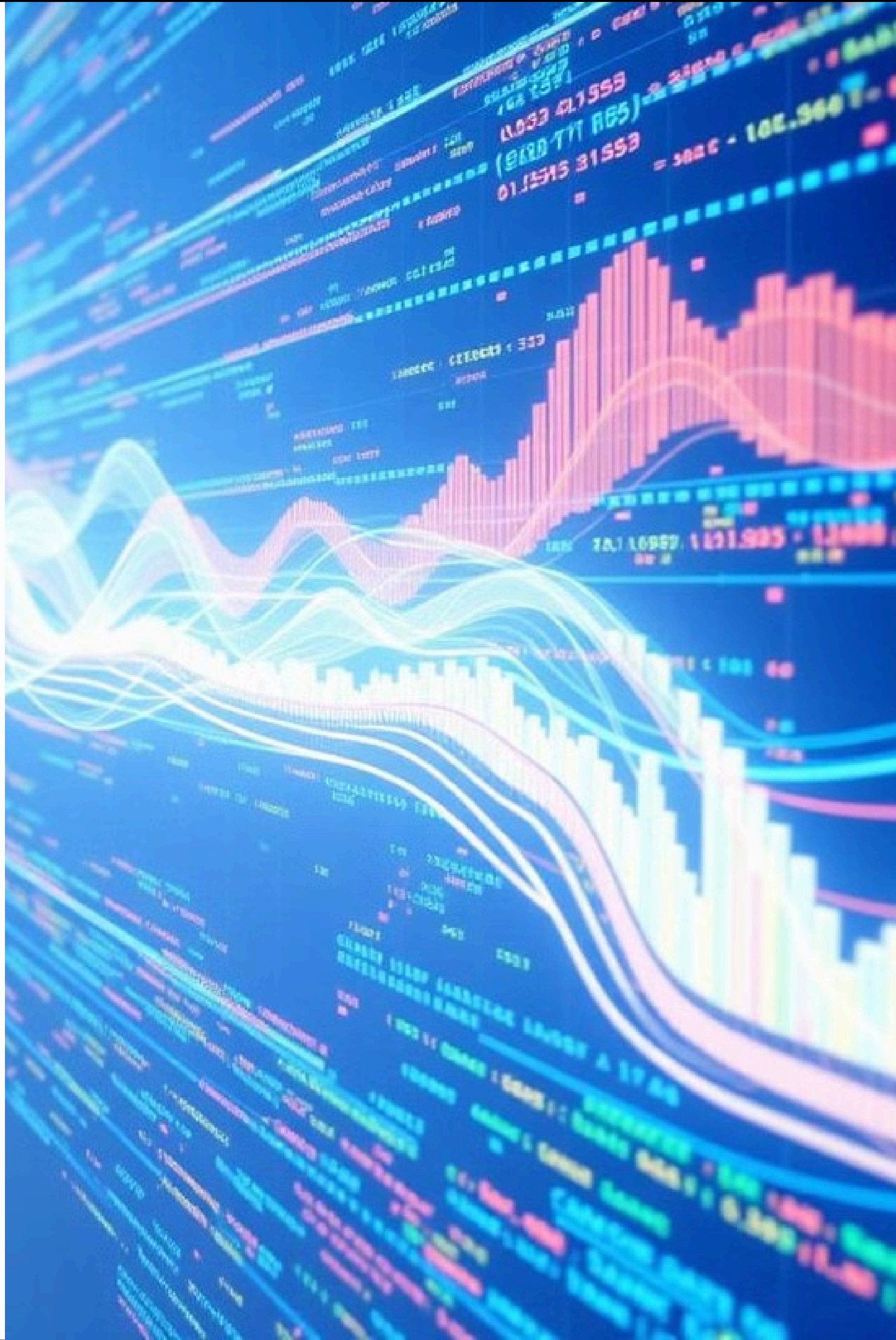
DATASTREAM

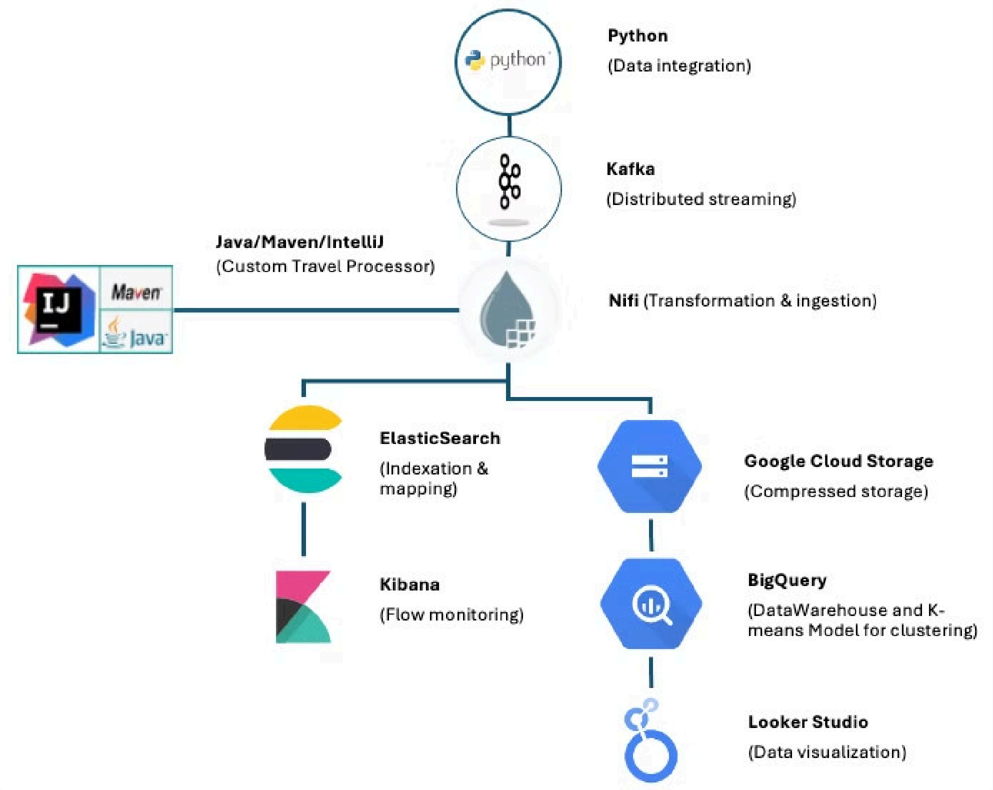
Back-end development for a real-time platform calculating taxi fare prices based on a selected comfort level.



1. Introduction

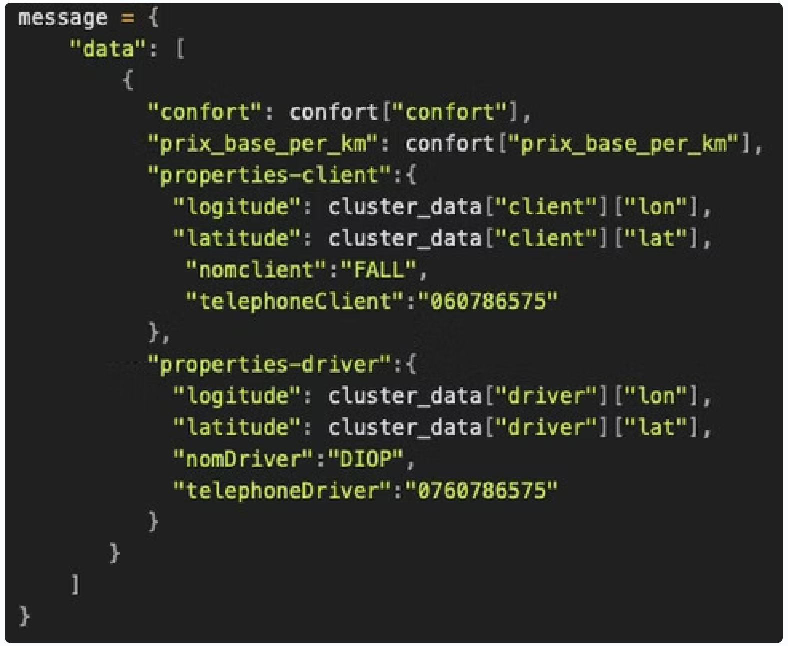
- **Objective**

Calculate the distance between a driver and a customer to determine the price of a trip based on the selected comfort type.

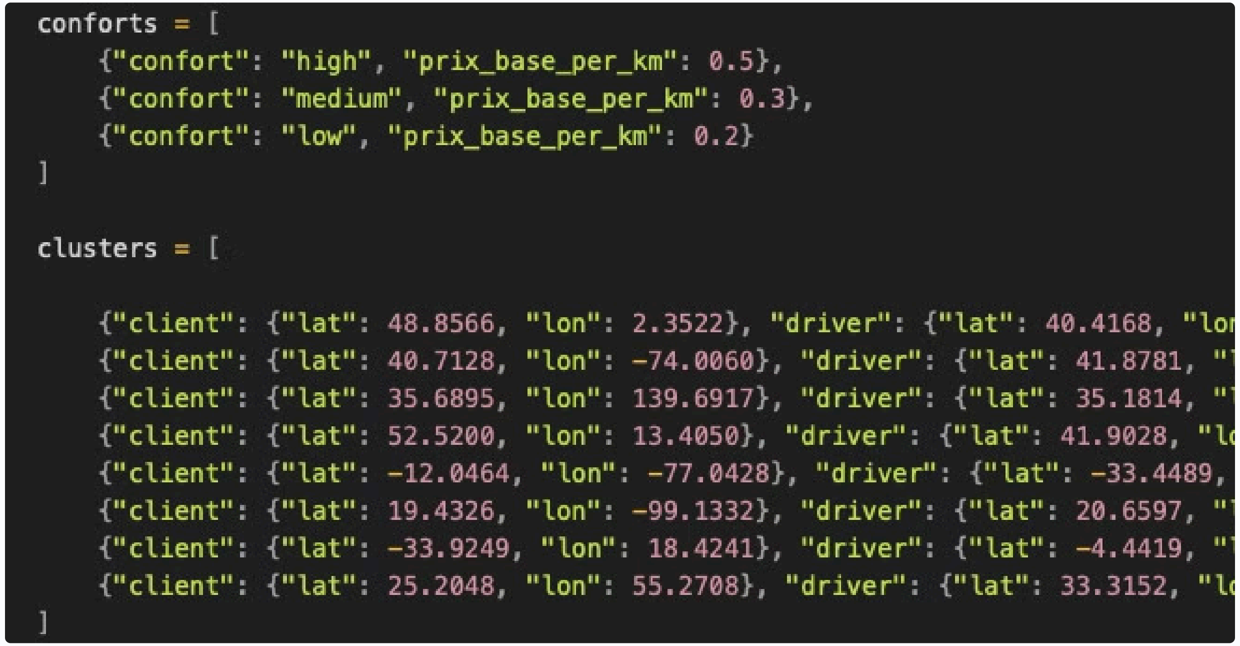


2. Data Model

Incoming Data in Kafka via Python 'KafkaProducer'



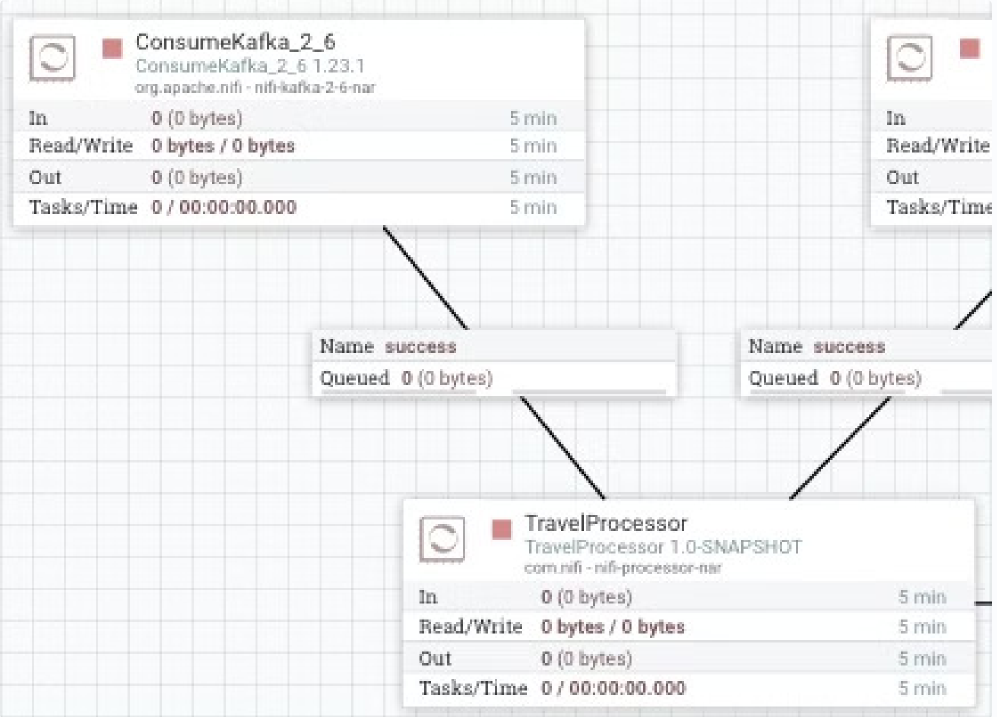
```
message = {
    "data": [
        {
          "confort": confort["confort"],
          "prix_base_per_km": confort["prix_base_per_km"],
          "properties-client":{
            "logitude": cluster_data["client"]["lon"],
            "latitude": cluster_data["client"]["lat"],
             "nomclient":"FALL",
             "telephoneClient":"060786575"
          },
          "properties-driver":{
            "logitude": cluster_data["driver"]["lon"],
            "latitude": cluster_data["driver"]["lat"],
            "nomDriver":"DIOP",
            "telephoneDriver":"0760786575"
          }
      }
    ]
}
```



```
conforts = [
    {"confort": "high", "prix_base_per_km": 0.5},
    {"confort": "medium", "prix_base_per_km": 0.3},
    {"confort": "low", "prix_base_per_km": 0.2}
]

clusters = [

    {"client": {"lat": 48.8566, "lon": 2.3522}, "driver": {"lat": 40.4168, "lon
    {"client": {"lat": 40.7128, "lon": -74.0060}, "driver": {"lat": 41.8781, "l
    {"client": {"lat": 35.6895, "lon": 139.6917}, "driver": {"lat": 35.1814, "l
    {"client": {"lat": 52.5200, "lon": 13.4050}, "driver": {"lat": 41.9028, "lo
    {"client": {"lat": -12.0464, "lon": -77.0428}, "driver": {"lat": -33.4489,
    {"client": {"lat": 19.4326, "lon": -99.1332}, "driver": {"lat": 20.6597, "l
    {"client": {"lat": -33.9249, "lon": 18.4241}, "driver": {"lat": -4.4419, "l
    {"client": {"lat": 25.2048, "lon": 55.2708}, "driver": {"lat": 33.3152, "lo
]
```

3. TravelProcessor Development

Development based on defined architecture and unit tests.



ConsumeKafka_2_6
ConsumeKafka_2_6 1.23.1
org.apache.nifi - nifi-kafka-2-6-nar

In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min

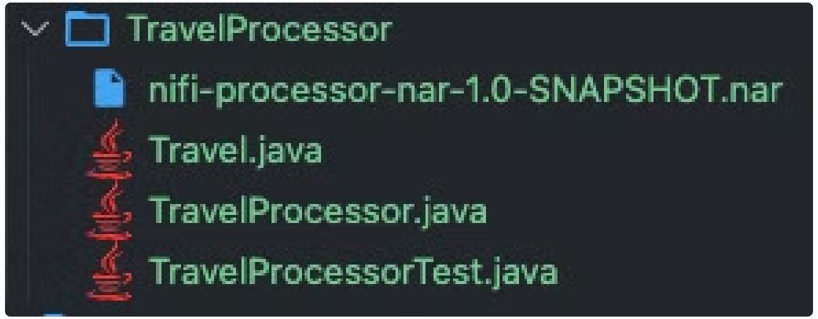
In	
Read/Write	
Out	
Tasks/Time	

Name success
Queued 0 (0 bytes)

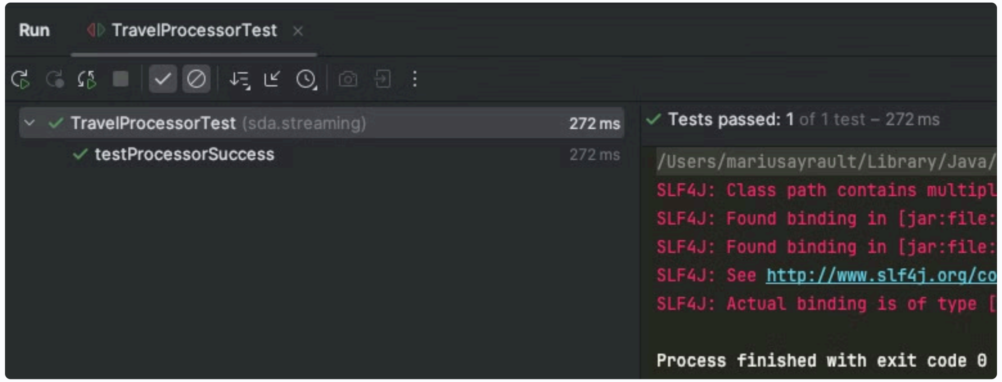
Name success
Queued 0 (0 bytes)

TravelProcessor
TravelProcessor 1.0-SNAPSHOT
com.nifi - nifi-processor-nar

In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min



```
TravelProcessor
    nifi-processor-nar-1.0-SNAPSHOT.nar
    Travel.java
    TravelProcessor.java
    TravelProcessorTest.java
```



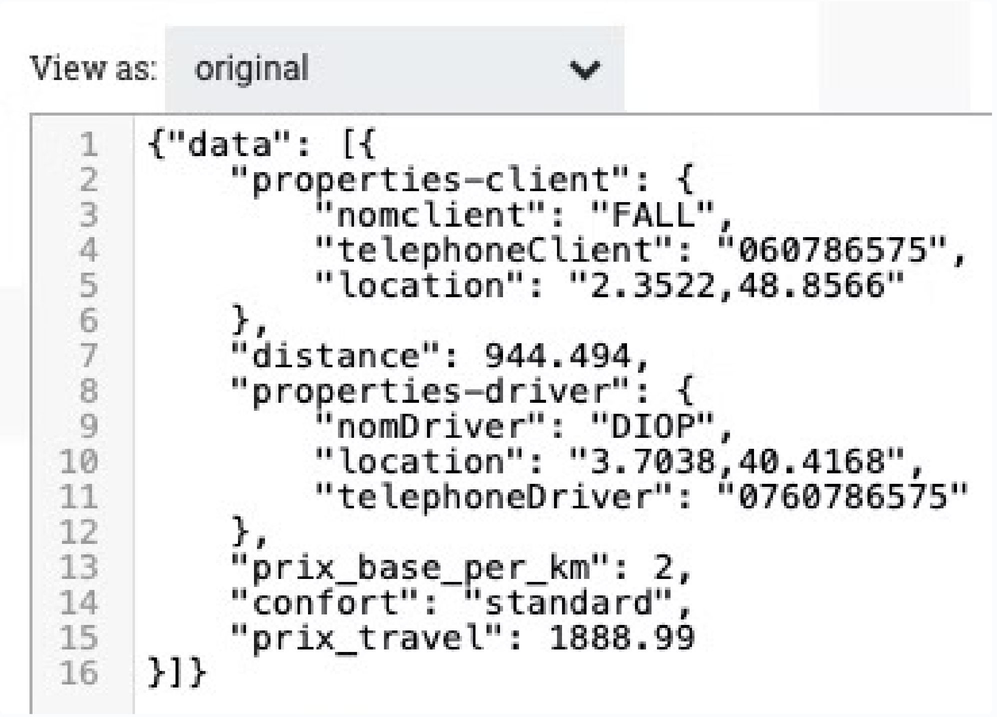
```
TravelProcessorTest (sda.streaming)    272 ms
    testProcessorSuccess               272 ms
```

```
✓ Tests passed: 1 of 1 test – 272 ms

/Users/mariusayrault/Library/Java/
SLF4J: Class path contains multipl
SLF4J: Found binding in [jar:file:
SLF4J: Found binding in [jar:file:
SLF4J: See http://www.slf4j.org/co
SLF4J: Actual binding is of type [

Process finished with exit code 0
```

4. Data Model

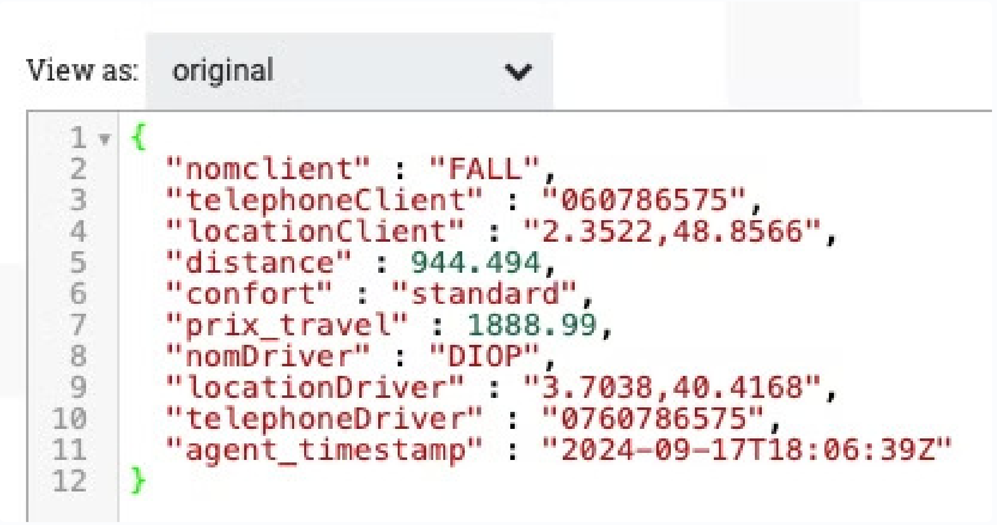
Transformation Examples

Custom TravelProcessor



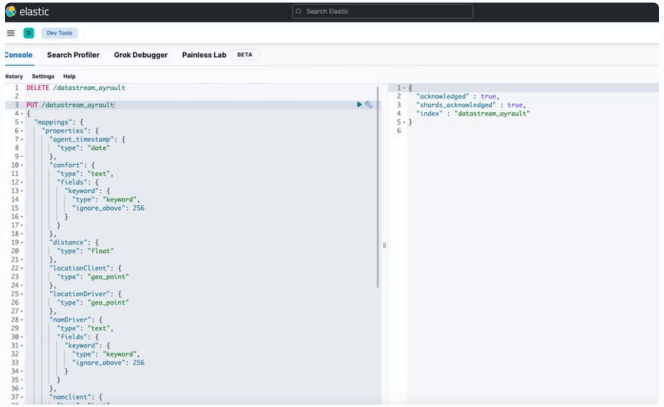
```
{"data": [{
    "properties-client": {
        "nomclient": "FALL",
        "telephoneClient": "060786575",
        "location": "2.3522,48.8566"
    },
    "distance": 944.494,
    "properties-driver": {
        "nomDriver": "DIOP",
        "location": "3.7038,40.4168",
        "telephoneDriver": "0760786575"
    },
    "prix_base_per_km": 2,
    "confort": "standard",
    "prix_travel": 1888.99
}]}
```

JoltTransformJSON



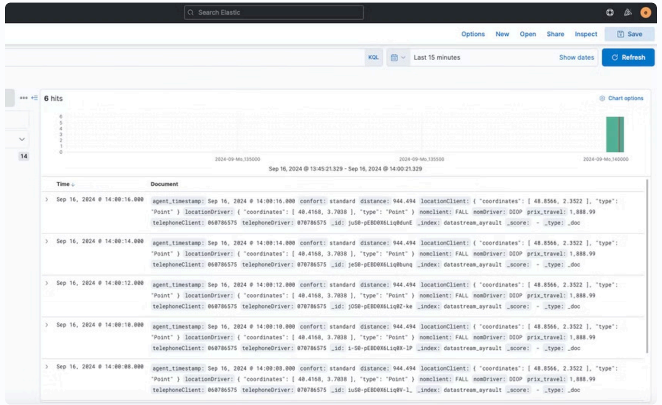
```
{
  "nomclient" : "FALL",
  "telephoneClient" : "060786575",
  "locationClient" : "2.3522,48.8566",
  "distance" : 944.494,
  "confort" : "standard",
  "prix_travel" : 1888.99,
  "nomDriver" : "DIOP",
  "locationDriver" : "3.7038,40.4168",
  "telephoneDriver" : "0760786575",
  "agent_timestamp" : "2024-09-17T18:06:39Z"
}
```

5. Indexing & monitoring

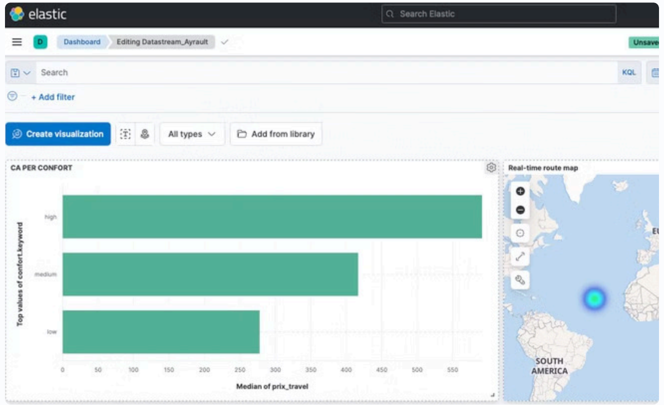


```
DELETE /datastream_ayrault

PUT /datastream_ayrault
{
  "mappings": {
    "properties": {
      "agent_timestamp": {
        "type": "date"
      },
      "confort": {
        "type": "text",
        "fields": {
          "keyword": {
            "type": "keyword",
            "ignore_above": 256
          }
        }
      },
      "distance": {
        "type": "float"
      },
      "locationClient": {
        "type": "geo_point"
      },
      "locationDriver": {
        "type": "geo_point"
      },
      "nomDriver": {
        "type": "text",
        "fields": {
          "keyword": {
            "type": "keyword",
            "ignore_above": 256
          }
        }
      },
      "nomclient": {
```

```
{
  "acknowledged" : true,
  "shards_acknowledged" : true,
  "index" : "datastream_ayrault"
}
```



```
Sep 16, 2024 @ 14:00:16.000  agent_timestamp: Sep 16, 2024 @ 14:00:16.000 confort: standard distance: 944.494 locationClient: { "coordinates": [ 48.8566, 2.3522 ], "type": "Point" } locationDriver: { "coordinates": [ 40.4168, 3.7038 ], "type": "Point" } nomclient: FALL nomDriver: DIOP prix_travel: 1,888.99 telephoneClient: 060786575 telephoneDriver: 070786575 _id: juS0-pEBD0X6Liq0dunE _index: datastream_ayrault _score: - _type: _doc
Sep 16, 2024 @ 14:00:14.000  agent_timestamp: Sep 16, 2024 @ 14:00:14.000 confort: standard distance: 944.494 locationClient: { "coordinates": [ 48.8566, 2.3522 ], "type": "Point" } locationDriver: { "coordinates": [ 40.4168, 3.7038 ], "type": "Point" } nomclient: FALL nomDriver: DIOP prix_travel: 1,888.99 telephoneClient: 060786575 telephoneDriver: 070786575 _id: jeS0-pEBD0X6Liq0bunq _index: datastream_ayrault _score: - _type: _doc
Sep 16, 2024 @ 14:00:12.000  agent_timestamp: Sep 16, 2024 @ 14:00:12.000 confort: standard distance: 944.494 locationClient: { "coordinates": [ 48.8566, 2.3522 ], "type": "Point" } locationDriver: { "coordinates": [ 40.4168, 3.7038 ], "type": "Point" } nomclient: FALL nomDriver: DIOP prix_travel: 1,888.99 telephoneClient: 060786575 telephoneDriver: 070786575 _id: jOS0-pEBD0X6Liq0Z-ke _index: datastream_ayrault _score: - _type: _doc
Sep 16, 2024 @ 14:00:10.000  agent_timestamp: Sep 16, 2024 @ 14:00:10.000 confort: standard distance: 944.494 locationClient: { "coordinates": [ 48.8566, 2.3522 ], "type": "Point" } locationDriver: { "coordinates": [ 40.4168, 3.7038 ], "type": "Point" } nomclient: FALL nomDriver: DIOP prix_travel: 1,888.99 telephoneClient: 060786575 telephoneDriver: 070786575 _id: i-S0-pEBD0X6Liq0X-lP _index: datastream_ayrault _score: - _type: _doc
Sep 16, 2024 @ 14:00:08.000  agent_timestamp: Sep 16, 2024 @ 14:00:08.000 confort: standard distance: 944.494 locationClient: { "coordinates": [ 48.8566, 2.3522 ], "type": "Point" } locationDriver: { "coordinates": [ 40.4168, 3.7038 ], "type": "Point" } nomclient: FALL nomDriver: DIOP prix_travel: 1,888.99 telephoneClient: 060786575 telephoneDriver: 070786575 _id: iuS0-pEBD0X6Liq0V-l_ _index: datastream_ayrault _score: - _type: _doc
```



Indexing and mapping via *ElasticSearch*

Definition of data indexing and mapping.

Performance analysis

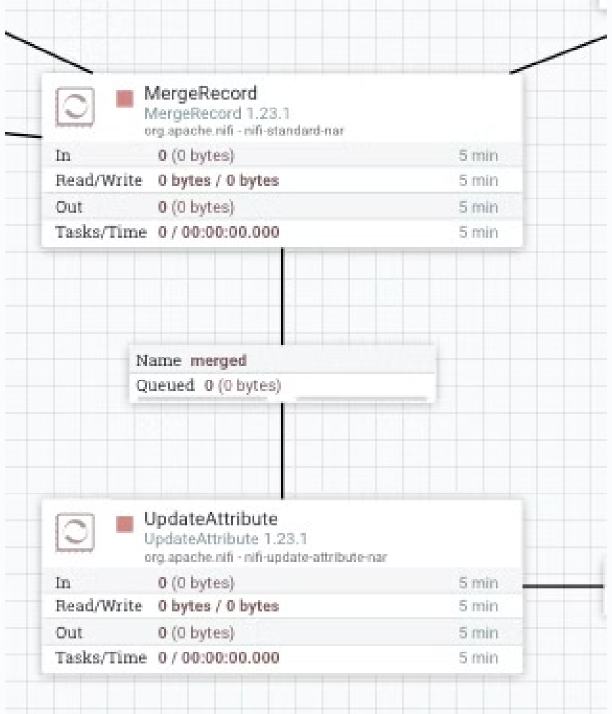
Monitoring of data streams.

Data analysis via *Kibana*

Detailed real-time visualization.

6. Datawarehouse and BigQuery

Merge all 10,000 records and compress data (.parquet*) with timestamp agent.



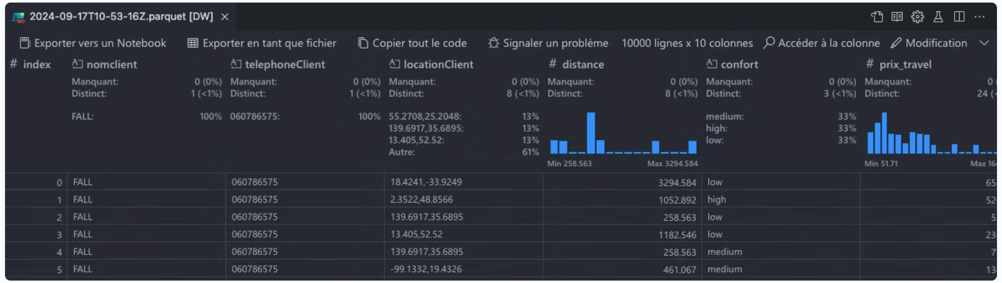
MergeRecord
MergeRecord 1.23.1
org.apache.nifi - nifi-standard-nar

In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min

Name merged
Queued 0 (0 bytes)

UpdateAttribute
UpdateAttribute 1.23.1
org.apache.nifi - nifi-update-attribute-nar

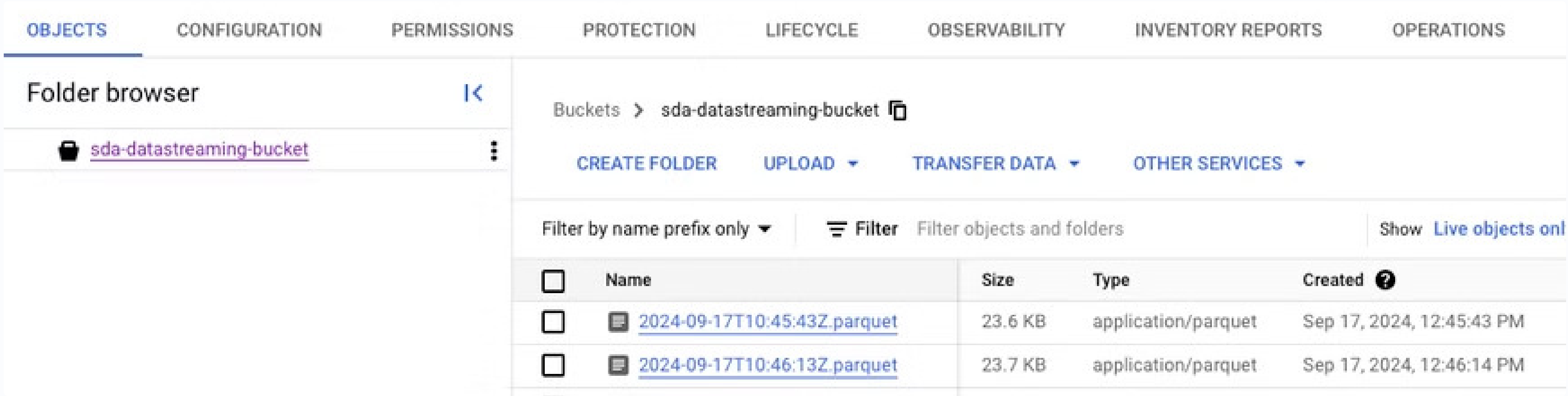
In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min



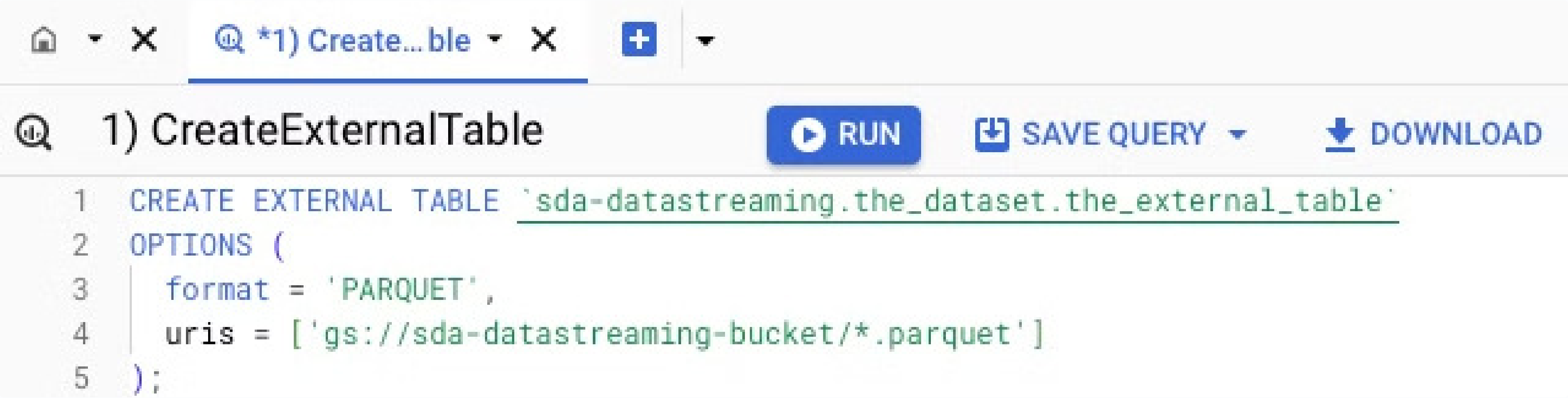
10000 lignes x 10 colonnes

# index	nomclient	telephoneClient	locationClient	# distance	confort	# prix_travel
	Manquant: 0 (0%) Distinct: 1 (<1%)	Manquant: 0 (0%) Distinct: 1 (<1%)	Manquant: 0 (0%) Distinct: 8 (<1%)	Manquant: 0 (0%) Distinct: 8 (<1%)	Manquant: 0 (0%) Distinct: 3 (<1%)	Manquant: 0 Distinct: 24 (
	FALL: 100%	060786575: 100%	55.2708,25.2048: 13% 139.6917,35.6895: 13% 13.405,52.52: 13% Autre: 61%		medium: 33% high: 33% low: 33%	
0	FALL	060786575	18.4241,-33.9249	3294.584	low	65
1	FALL	060786575	2.3522,48.8566	1052.892	high	52
2	FALL	060786575	139.6917,35.6895	258.563	low	5
3	FALL	060786575	13.405,52.52	1182.546	low	23
4	FALL	060786575	139.6917,35.6895	258.563	medium	7
5	FALL	060786575	-99.1332,19.4326	461.067	medium	13

7. DataWarehouse Configuration



□	Name	Size	Type	Created
□	2024-09-17T10:45:43Z.parquet	23.6 KB	application/parquet	Sep 17, 2024, 12:45:43 PM
□	2024-09-17T10:46:13Z.parquet	23.7 KB	application/parquet	Sep 17, 2024, 12:46:14 PM



```
CREATE EXTERNAL TABLE `sda-datastreaming.the_dataset.the_external_table`
OPTIONS (
  format = 'PARQUET',
  uris = ['gs://sda-datastreaming-bucket/*.parquet']
);
```

8. Revenue Calculation by Cluster and Comfort Level

Analysis process with BigQuery ML

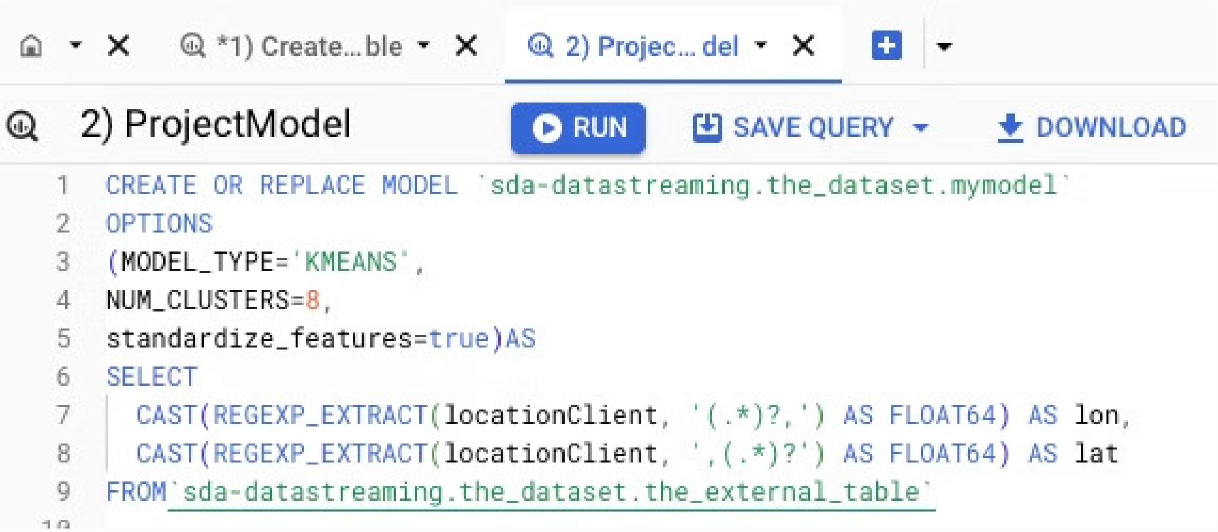
Using BigQuery ML to create a K-Means model.

Clustering Methodology

Identification of 8 clusters based on geographical coordinates.

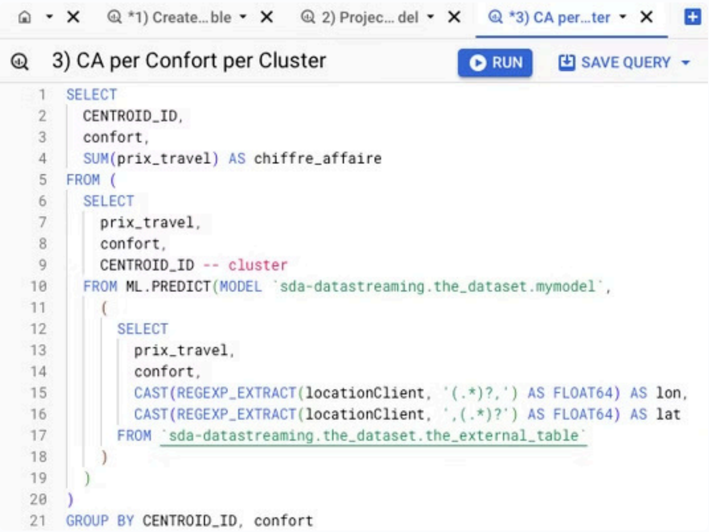
Revenue Calculation

Calculation of revenue per cluster and comfort type.



2) ProjectModel

```sql
CREATE OR REPLACE MODEL `sda-datastreaming.the_dataset.mymodel`
OPTIONS
(MODEL_TYPE='KMEANS',
NUM_CLUSTERS=8,
standardize_features=true)AS
SELECT
  CAST(REGEXP_EXTRACT(locationClient, '(.*)?,') AS FLOAT64) AS lon,
  CAST(REGEXP_EXTRACT(locationClient, ',(.*)?') AS FLOAT64) AS lat
FROM`sda-datastreaming.the_dataset.the_external_table`
```

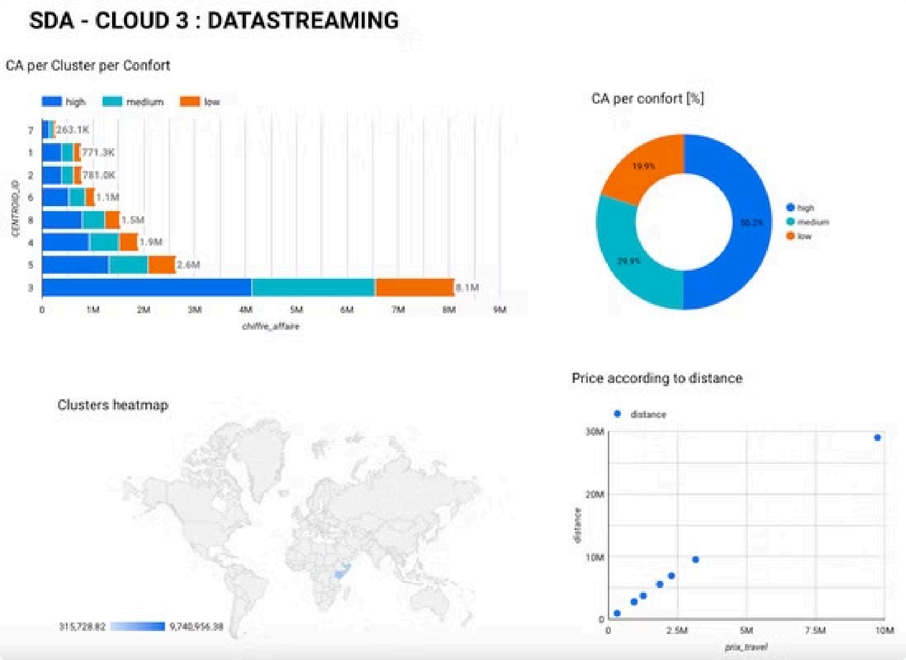


3) CA per Confort per Cluster

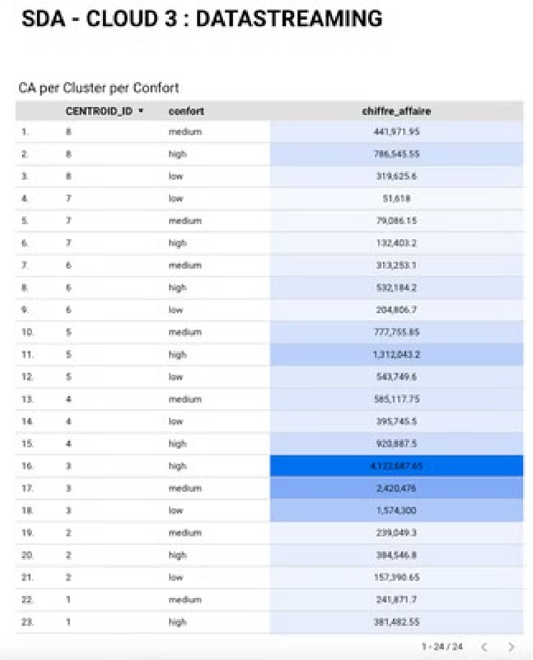
```sql
SELECT
  CENTROID_ID,
  confort,
  SUM(prix_travel) AS chiffre_affaire
FROM (
  SELECT
    prix_travel,
    confort,
    CENTROID_ID -- cluster
  FROM ML.PREDICT(MODEL `sda-datastreaming.the_dataset.mymodel`,
    (
      SELECT
        prix_travel,
        confort,
        CAST(REGEXP_EXTRACT(locationClient, '(.*)?,') AS FLOAT64) AS lon,
        CAST(REGEXP_EXTRACT(locationClient, ',(.*)?') AS FLOAT64) AS lat
      FROM `sda-datastreaming.the_dataset.the_external_table`
    )
  )
)
GROUP BY CENTROID_ID, confort
```

9. Data visualization

Visualizing the model results via Looker Studio



SDA - CLOUD 3 : DATASTREAMING



SDA - CLOUD 3 : DATASTREAMING

CA per Cluster per Confort

	CENTROID_ID ▾	confort	chiffre_affaire
1.	8	medium	441,971.95
2.	8	high	786,545.55
3.	8	low	319,625.6
4.	7	low	51,618
5.	7	medium	79,086.15
6.	7	high	132,403.2
7.	6	medium	313,253.1
8.	6	high	532,184.2
9.	6	low	204,806.7
10.	5	medium	777,755.85
11.	5	high	1,312,043.2
12.	5	low	543,749.6
13.	4	medium	585,117.75
14.	4	low	395,745.5
15.	4	high	920,887.5
16.	3	high	4,122,687.65
17.	3	medium	2,420,476
18.	3	low	1,574,300
19.	2	medium	239,049.3
20.	2	high	384,546.8
21.	2	low	157,390.65
22.	1	medium	241,871.7
23.	1	high	381,482.55