

Project cuối khóa: “Xây dựng Big Data Platform lưu trữ và phân tích dữ liệu Uber”

Đặng Văn Nam (Trưởng nhóm)

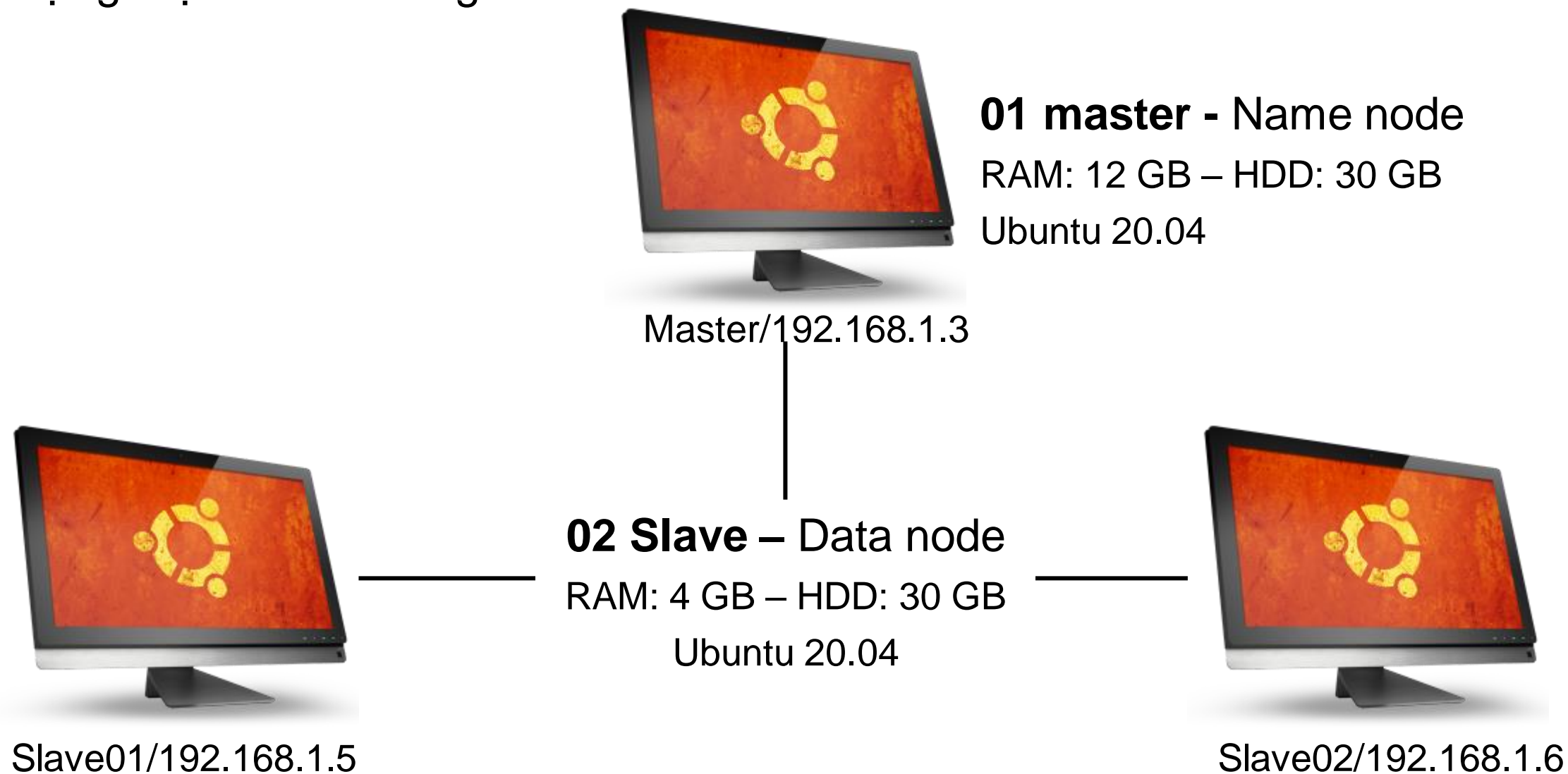
Ngô Văn Mạnh

- 1. Xây dựng được Big data platform (1 name node + 2 data node) thực hiện lưu trữ dữ liệu về thời điểm và vị trí đón khách của Uber.**
- 2. Tạo và lưu trữ dữ liệu với Hive**
- 3. Một số truy vấn dữ liệu trên bảng trong Hive**
- 4. Kết hợp SparkML và SparkSQL thực hiện phân tích tập dữ liệu, xác định các thông tin sau:**
 - 1. Các khu vực có mật độ khách book xe cao nhất?**
 - 2. Xác định các khung giờ có lượng book cao nhất, thấp nhất?**
 - 3. So sánh tỷ lệ book vé của 6 khu vực có mật độ cao nhất theo khung giờ buổi sáng, buổi chiều, buổi tối.**

1. Xây dựng Big Data Platform

1. Xây dựng Big Data Platform lưu trữ dữ liệu

Xây dựng một cluster bao gồm:



1. Xây dựng Big Data Platform lưu trữ dữ liệu

Summary

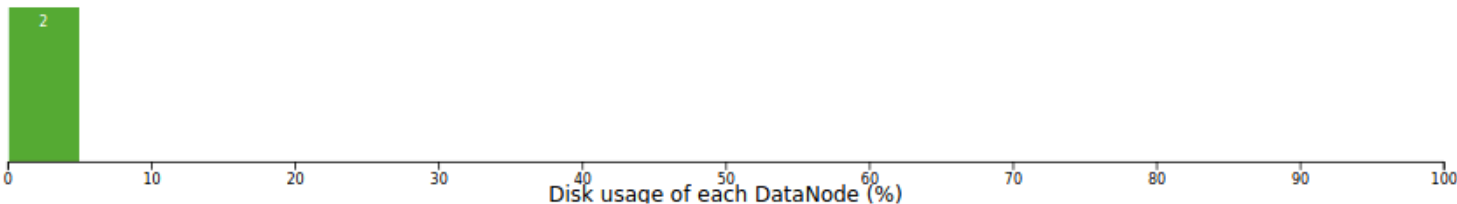
Security is off.
Safemode is off.
3,130 files and directories, 2,822 blocks (2,822 replicated blocks, 0 erasure coded block groups) = 5,952 total filesystem object(s).
Heap Memory used 203.94 MB of 313 MB Heap Memory. Max Heap Memory is 2.6 GB.
Non Heap Memory used 74.53 MB of 76.75 MB Committed Non Heap Memory. Max Non Heap Memory is <unbounded>.

Configured Capacity:	57.82 GB
Configured Remote Capacity:	0 B
DFS Used:	1.1 GB (1.9%)
Non DFS Used:	21.5 GB
DFS Remaining:	32.23 GB (55.75%)
Block Pool Used:	1.1 GB (1.9%)
DataNodes usages% (Min/Median/Max/stdDev):	1.90% / 1.90% / 1.90% / 0.00%
Live Nodes	2 (Decommissioned: 0, In Maintenance: 0)

Datanode Information

✓ In service ⚠ Down 🔄 Decommissioning 🛑 Decommissioned 🛑 Decommissioned & dead
🔧 Entering Maintenance 🔧 In Maintenance 🔧 In Maintenance & dead

Datanode usage histogram



In operation

DataNode State: All Show: 25 entries Search:

Node	Http Address	Last contact	Last Block Report	Used	Non DFS Used	Capacity	Blocks	Block pool used	Version
✓ slave01:9866 (192.168.1.5:9866)	http://slave01:9864	0s	356m	563.72 MB	10.71 GB	28.91 GB	2822	563.72 MB (1.9%)	3.3.0
✓ slave02:9866 (192.168.1.6:9866)	http://slave02:9864	2s	202m	563.72 MB	10.79 GB	28.91 GB	2822	563.72 MB (1.9%)	3.3.0

Showing 1 to 2 of 2 entries

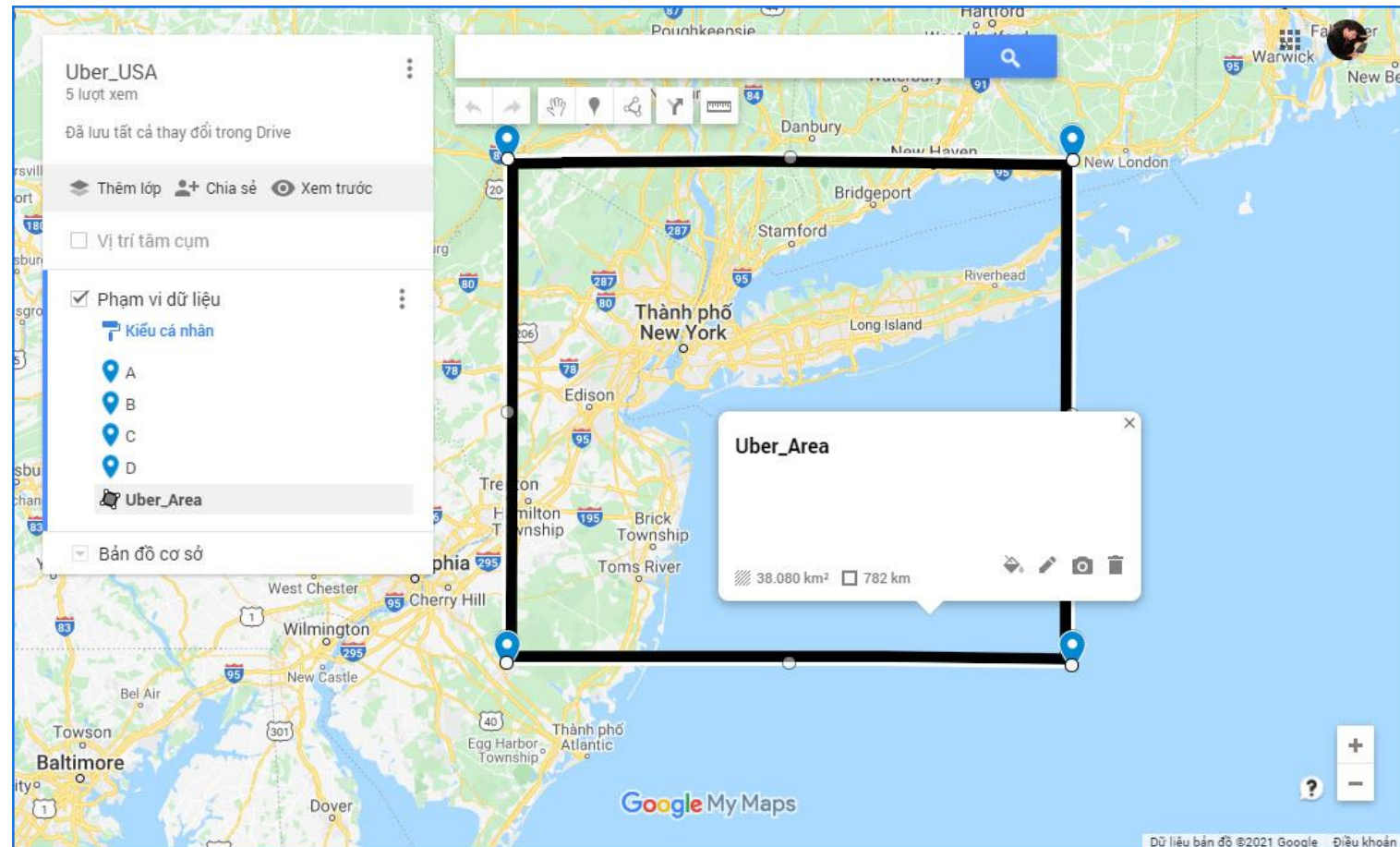
Previous 1 Next

2. Lưu trữ dữ liệu với Hive

2. Mô tả dữ liệu uber

- Dữ liệu bao gồm 829 275 bản ghi (Vị trí và thời điểm đón khách) trong khoảng thời gian từ 2014-08-01 00:00 tới 2014-08-31 23:59 (1 tháng), mỗi bản ghi bao gồm các thông tin sau:
 - datetime: Thời điểm đón khách
 - Lat: Kinh độ vị trí đón khách
 - Lon: Vĩ độ vị trí đón khách
 - Base: Mã công ty quản lý phương tiện

2014-08-01 0:00	40.729	-73.9422	B02598
2014-08-01 0:00	40.7476	-73.9871	B02598
2014-08-01 0:00	40.7424	-74.0044	B02598
2014-08-01 0:00	40.751	-73.9869	B02598
2014-08-01 0:00	40.7406	-73.9902	B02598
2014-08-01 0:00	40.6994	-73.9591	B02617
2014-08-01 0:00	40.6917	-73.9398	B02617
2014-08-01 0:00	40.7063	-73.9223	B02617
2014-08-01 0:00	40.6759	-74.0168	B02617
2014-08-01 0:00	40.7617	-73.9847	B02617
2014-08-01 0:00	40.6969	-73.9064	B02617
2014-08-01 0:00	40.7623	-73.9751	B02617
2014-08-01 0:00	40.6982	-73.9669	B02617
2014-08-01 0:00	40.7553	-73.9253	B02617
2014-08-01 0:00	40.7325	-73.9876	B02682



1. Tạo CSDL UBER, bảng Uber_raw lưu trữ dữ liệu trong Hive

```
hive>  
> CREATE TABLE IF NOT EXISTS Uber_raw(datetime Timestamp, lat decimal(8,5), lon decimal(8,5), base string)  
> COMMENT 'Data Uber Raw'  
> ROW FORMAT DELIMITED  
> FIELDS TERMINATED BY ','  
> STORED AS TEXTFILE;  
OK  
Time taken: 0.11 seconds
```

2. Load dữ liệu từ file Uber.csv vào bảng Uber_raw trong Hive:

```
hive> LOAD DATA LOCAL INPATH '/home/locnt/Downloads/uber.csv' OVERWRITE INTO TABLE uber_raw;  
Loading data to table uber.uber_raw  
OK  
Time taken: 0.613 seconds
```





```
hive> SHOW TABLES;  
OK  
uber_data  
uber_raw  
Time taken: 0.043 seconds, Fetched: 2 row(s)  
hive> DESCRIBE uber_raw;  
OK  
datetime          timestamp  
lat                decimal(8,5)  
lon                decimal(8,5)  
base               string  
Time taken: 0.072 seconds, Fetched: 4 row(s)
```


2. Tạo và lưu trữ dữ liệu với Hive


- File dữ liệu được lưu trữ trên HDFS của Hadoop:

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities ▾

Browse Directory

Go!    

Show entries Search:

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	<input type="checkbox"/>
<input type="checkbox"/>	-rwxr-xr-x	locnt	supergroup	34.62 MB	Jun 09 15:13	2	128 MB	uber.csv	

Showing 1 to 1 of 1 entries

Previous **1** Next

Hadoop, 2020.

File information - uber.csv ✕

Download Head the file (first 32K) Tail the file (last 32K)

Block information --

Block ID: 1073742830
Block Pool ID: BP-1302352360-192.168.1.3-1617900323427
Generation Stamp: 2006
Size: 36302337
Availability:

- slave02
- slave01

File contents

```

2014-08-01 00:00:00,40.729,-73.9422,B02598
2014-08-01 00:00:00,40.7476,-73.9871,B02598
2014-08-01 00:00:00,40.7424,-74.0044,B02598
2014-08-01 00:00:00,40.751,-73.9869,B02598
2014-08-01 00:00:00,40.7406,-73.9902,B02598
2014-08-01 00:00:00,40.6994,-73.9591,B02617
2014-08-01 00:00:00,40.6917,-73.9398,B02617
2014-08-01 00:00:00,40.7063,-73.9223,B02617

```

Close

3. Truy vấn dữ liệu với Hive

3. Truy vấn dữ liệu với Hive

1. Thống kê số lượng bản ghi trong bảng uber_raw:

```
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2021-06-09 15:54:18,841 Stage-1 map = 0%, reduce = 0%
2021-06-09 15:54:27,093 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 4.26 sec
2021-06-09 15:54:34,335 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 7.4 sec
MapReduce Total cumulative CPU time: 7 seconds 400 msec
Ended Job = job_1623210889275_0005
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 7.4 sec HDFS Read: 36310859 H
DFS Write: 106 SUCCESS
Total MapReduce CPU Time Spent: 7 seconds 400 msec
OK
829275
Time taken: 25.658 seconds, Fetched: 1 row(s)
```

2. Liệt kê danh sách 20 bản ghi đầu tiên trong ngày 15-08-2014 có mã quản lý xe B02682:

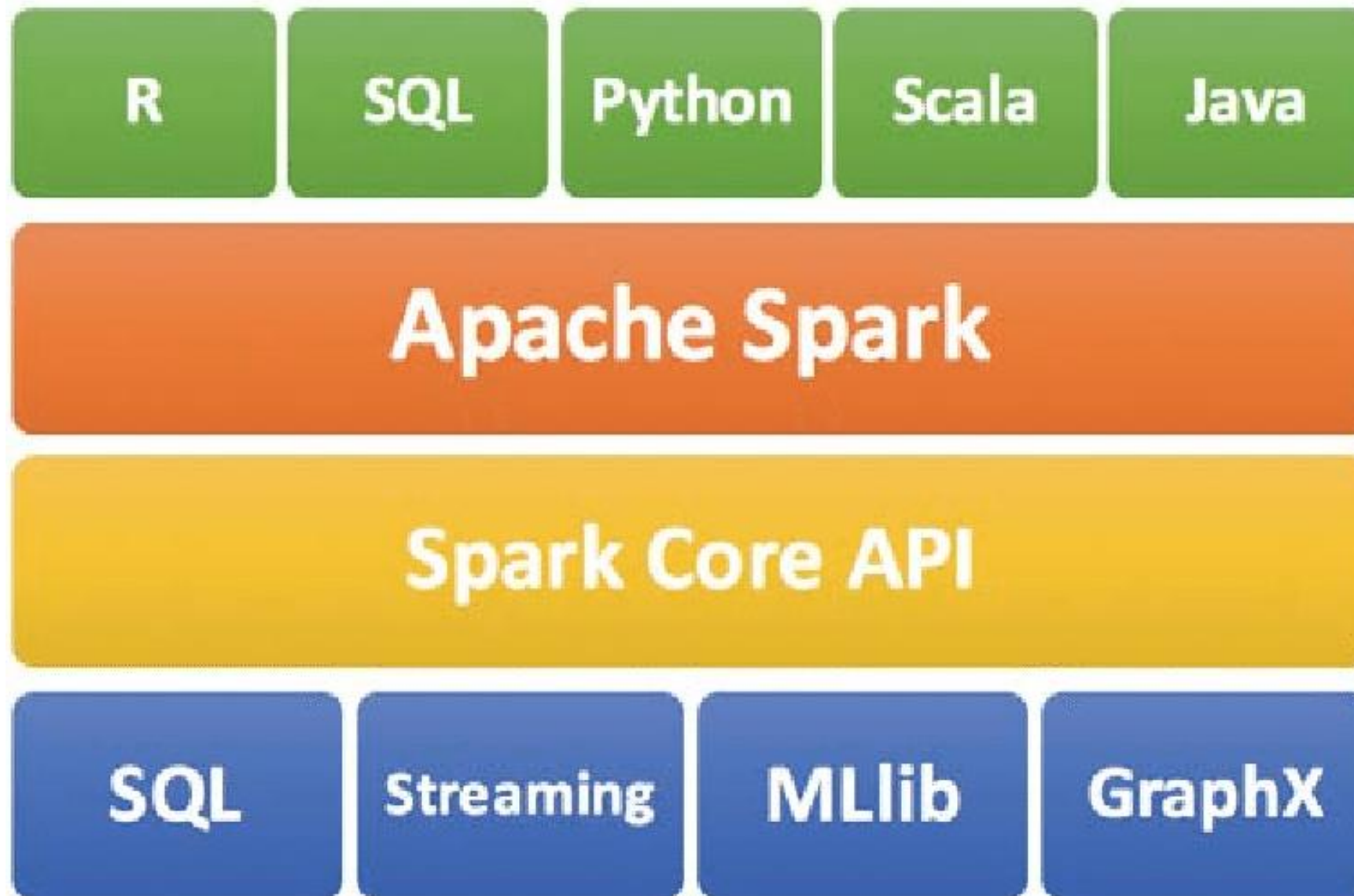
```
hive> SELECT * FROM uber_raw WHERE date(datetime) = '2014-08-15' and base='B02682' LIMIT 20;
OK
2014-08-15 00:00:00 40.75120 -74.02860 B02682
2014-08-15 00:00:00 40.82210 -73.95350 B02682
2014-08-15 00:00:00 40.73320 -74.00630 B02682
2014-08-15 00:00:00 40.71890 -73.98920 B02682
2014-08-15 00:01:00 40.69290 -73.95480 B02682
2014-08-15 00:01:00 40.75040 -73.98660 B02682
2014-08-15 00:01:00 40.69450 -73.90640 B02682
2014-08-15 00:01:00 40.68810 -73.96090 B02682
2014-08-15 00:02:00 40.72240 -73.99760 B02682
2014-08-15 00:02:00 40.76070 -73.97910 B02682
2014-08-15 00:02:00 40.72240 -73.98750 B02682
2014-08-15 00:02:00 40.77190 -73.95600 B02682
2014-08-15 00:03:00 40.74370 -73.99240 B02682
2014-08-15 00:03:00 40.95000 -73.95760 B02682
2014-08-15 00:04:00 40.75110 -74.00660 B02682
2014-08-15 00:05:00 40.74000 -74.00560 B02682
2014-08-15 00:05:00 40.64670 -73.78950 B02682
2014-08-15 00:05:00 40.75550 -73.98720 B02682
2014-08-15 00:07:00 40.76330 -73.98280 B02682
2014-08-15 00:07:00 40.64480 -73.78220 B02682
Time taken: 0.192 seconds, Fetched: 20 row(s)
```


3. Thống kê dữ liệu theo mã hãng quản lý xe (base):

SELECT base, count(base) as count FROM uber_raw GROUP BY base ORDER BY count desc;

```
Starting Job = job_1623210889275_0004, Tracking URL = http://master:8088/proxy/application_1623210889275_0004/
Kill Command = /usr/local/hadoop/bin/hadoop job -kill job_1623210889275_0004
Hadoop job information for Stage-2: number of mappers: 1; number of reducers: 1
2021-06-09 15:19:58,488 Stage-2 map = 0%, reduce = 0%
2021-06-09 15:20:05,711 Stage-2 map = 100%, reduce = 0%, Cumulative CPU 3.21 sec
2021-06-09 15:20:15,021 Stage-2 map = 100%, reduce = 100%, Cumulative CPU 6.13 sec
MapReduce Total cumulative CPU time: 6 seconds 130 msec
Ended Job = job_1623210889275_0004
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 7.05 sec HDFS Read: 36310460 HDFS Write: 234 SUCCESS
Stage-Stage-2: Map: 1 Reduce: 1 Cumulative CPU: 6.13 sec HDFS Read: 5707 HDFS Write: 215 SUCCESS
Total MapReduce CPU Time Spent: 13 seconds 180 msec
OK
B02617 355803
B02598 220129
B02682 173280
B02764 48591
B02512 31472
Time taken: 56.552 seconds, Fetched: 5 row(s)
```

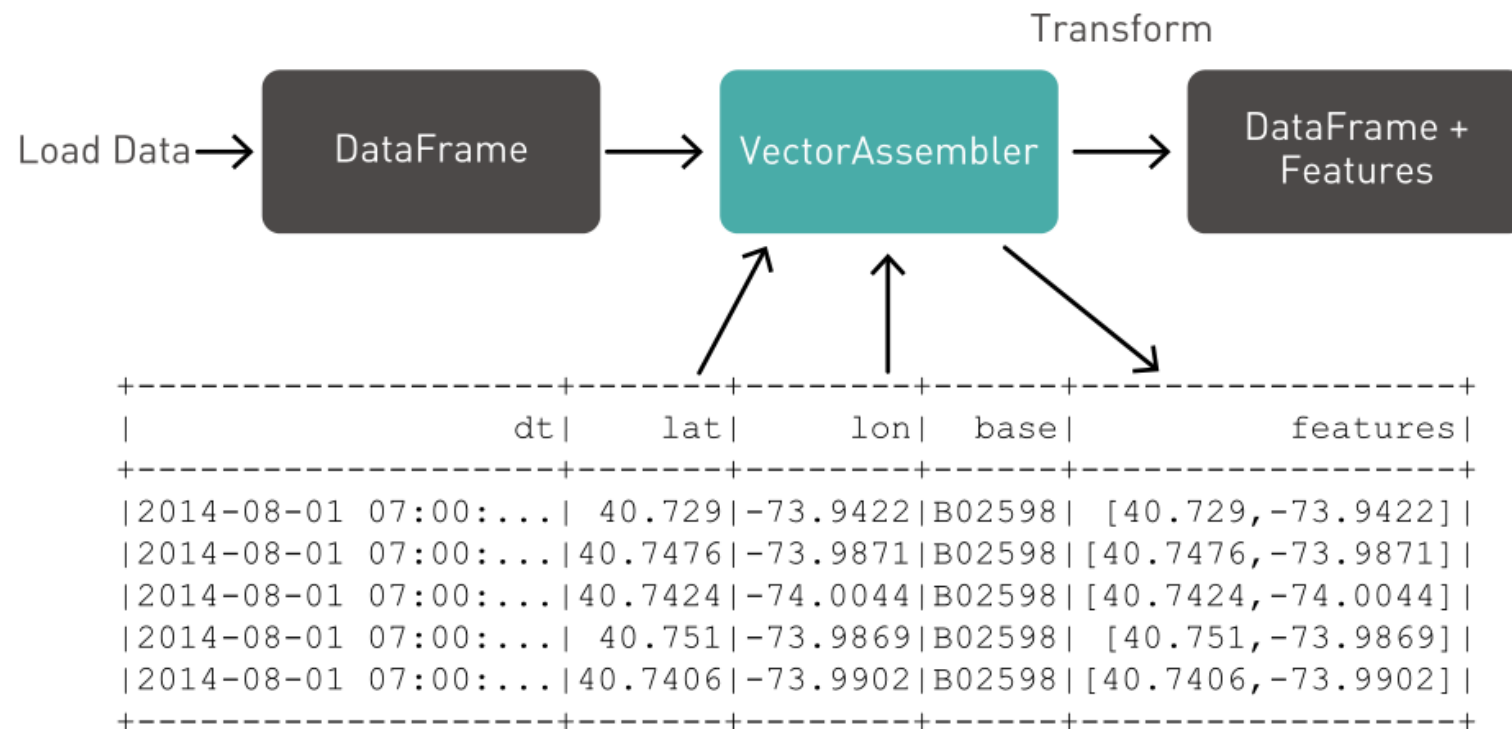
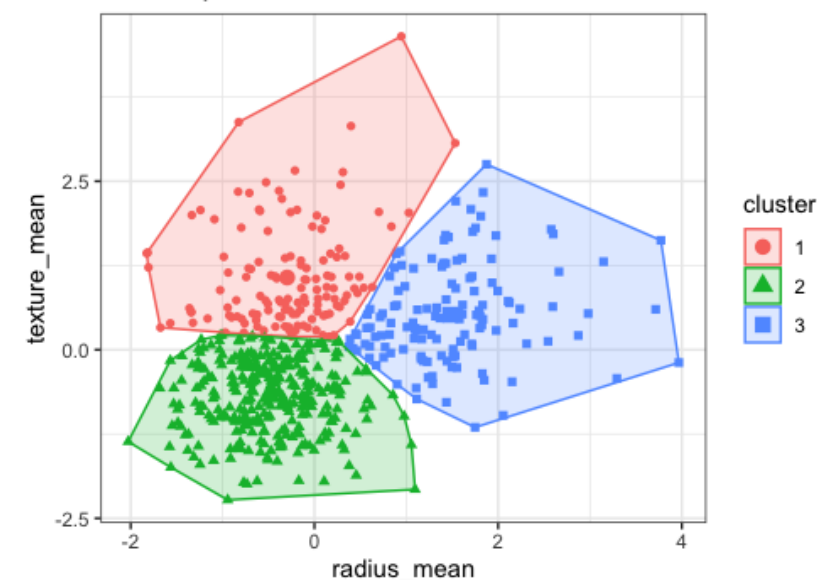
4. Sử dụng Spark phân tích dữ liệu uber



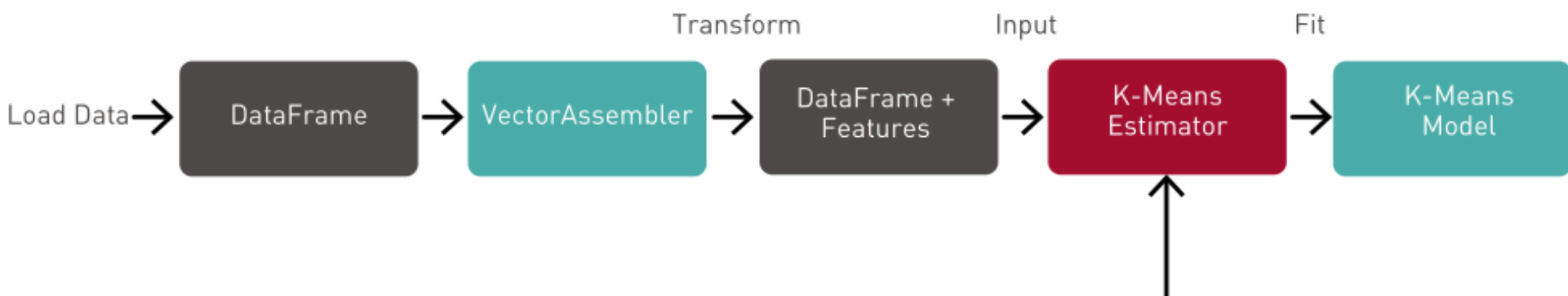
4.1 Phân cụm các điểm đón khách với Kmeans

- Dựa vào vị trí đón khách (lat, lon), thực hiện gom nhóm các điểm gần nhau vào 20 cụm với thuật toán Kmeans:
- Mục tiêu:** Xác định được các khu vực có mật độ khách đặt xe cao nhất?

Cluster plot



4.1 Phân cụm các điểm đón khách với Kmeans



dt	lat	lon	base	features
2014-08-01 07:00:...	40.729	-73.9422	B02598	[40.729, -73.9422]
2014-08-01 07:00:...	40.7476	-73.9871	B02598	[40.7476, -73.9871]
2014-08-01 07:00:...	40.7424	-74.0044	B02598	[40.7424, -74.0044]
2014-08-01 07:00:...	40.751	-73.9869	B02598	[40.751, -73.9869]
2014-08-01 07:00:...	40.7406	-73.9902	B02598	[40.7406, -73.9902]

```
#Sử dụng thuật toán KMeans để phân cụm dữ liệu
#Vị trí (kinh độ, vĩ độ) - features được sử dụng để phân thành 20 cụm

from pyspark.ml.clustering import KMeans
from pyspark.ml.evaluation import ClusteringEvaluator

# Trains a k-means model.
kmeans = KMeans().setK(20).setFeaturesCol("features").setPredictionCol("cid").setSeed(1)
model = kmeans.fit(df_uber_2)

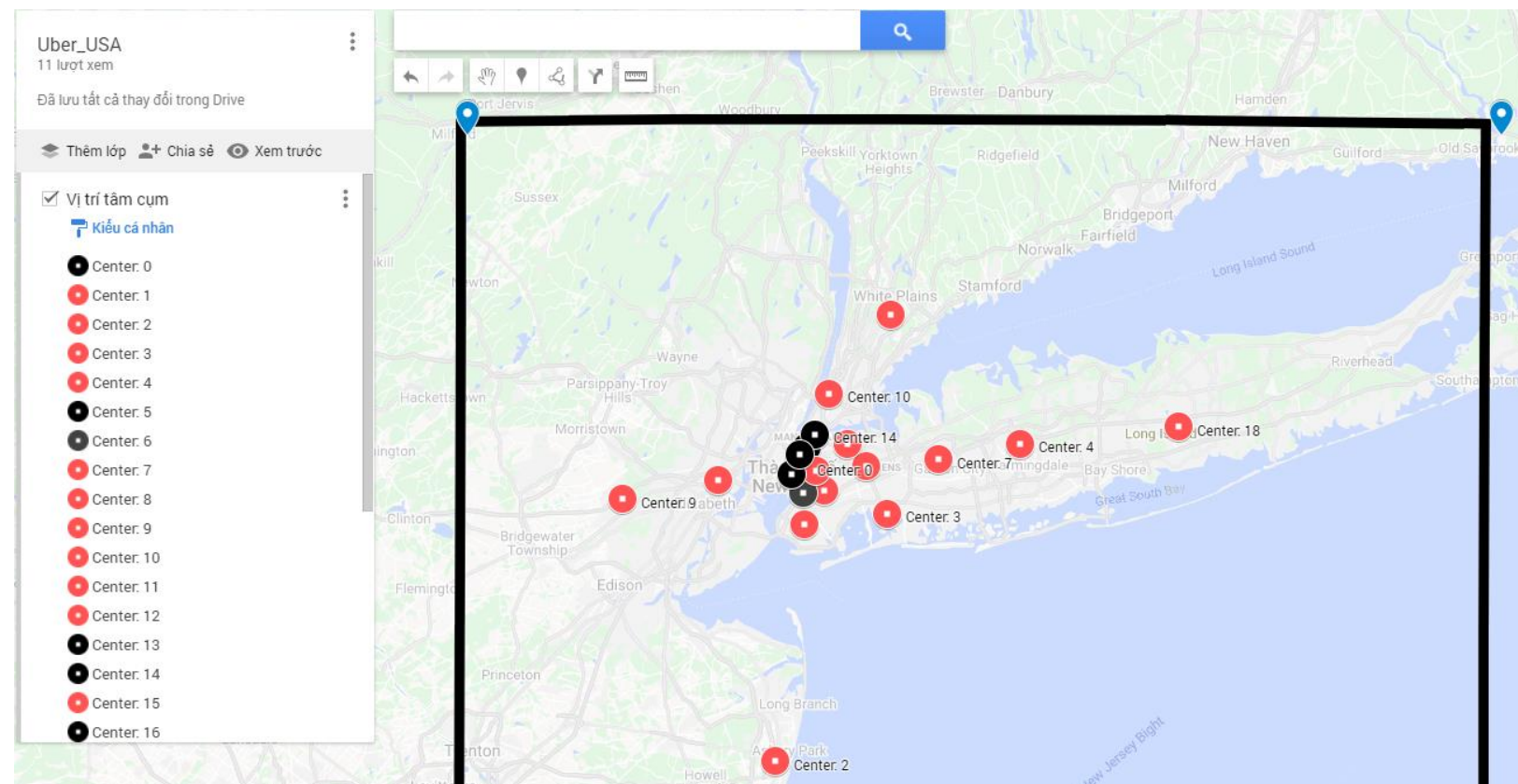
# Make predictions
predictions = model.transform(df_uber_2)
```

4.1 Phân cụm các điểm đón khách với Kmeans

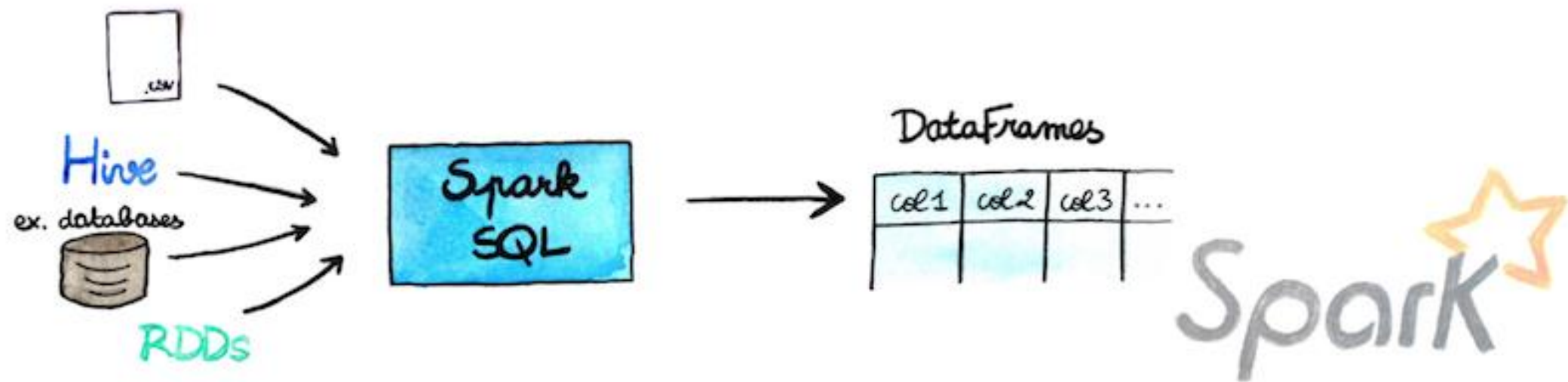
- Tâm của 20 cụm:

Cluster Centers:

```
0 [ 40.73029505 -73.99401699 ]
1 [ 40.6869184 -73.93251392 ]
2 [ 40.20140148 -74.04681999 ]
3 [ 40.64618963 -73.78285196 ]
4 [ 40.77051196 -73.47089168 ]
5 [ 40.76340085 -73.97456554 ]
6 [ 40.6828684 -73.98276235 ]
7 [ 40.74266896 -73.66292005 ]
8 [ 40.72192509 -73.95204341 ]
9 [ 40.6717815 -74.40639041 ]
10 [ 40.86049186 -73.9226221 ]
11 [ 40.99980858 -73.77604296 ]
12 [ 40.62560232 -73.97997814 ]
13 [ 40.71617648 -74.01075134 ]
14 [ 40.78690096 -73.95471837 ]
15 [ 40.70563962 -74.18181722 ]
16 [ 40.75089386 -73.99102045 ]
17 [ 40.73114395 -73.83256963 ]
18 [ 40.80073212 -73.09776341 ]
19 [ 40.7692754 -73.87837059 ]
```



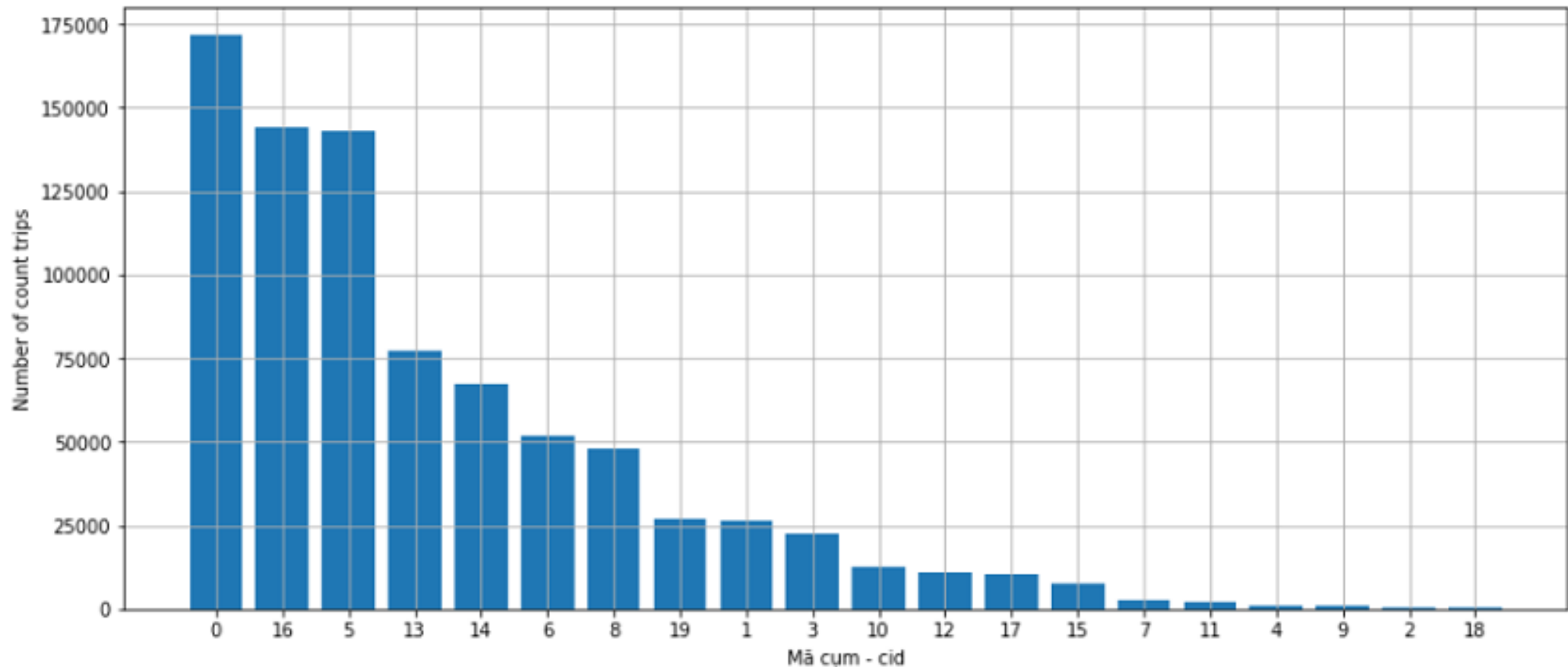
4.2 Phân tích dữ liệu với sparkSQL



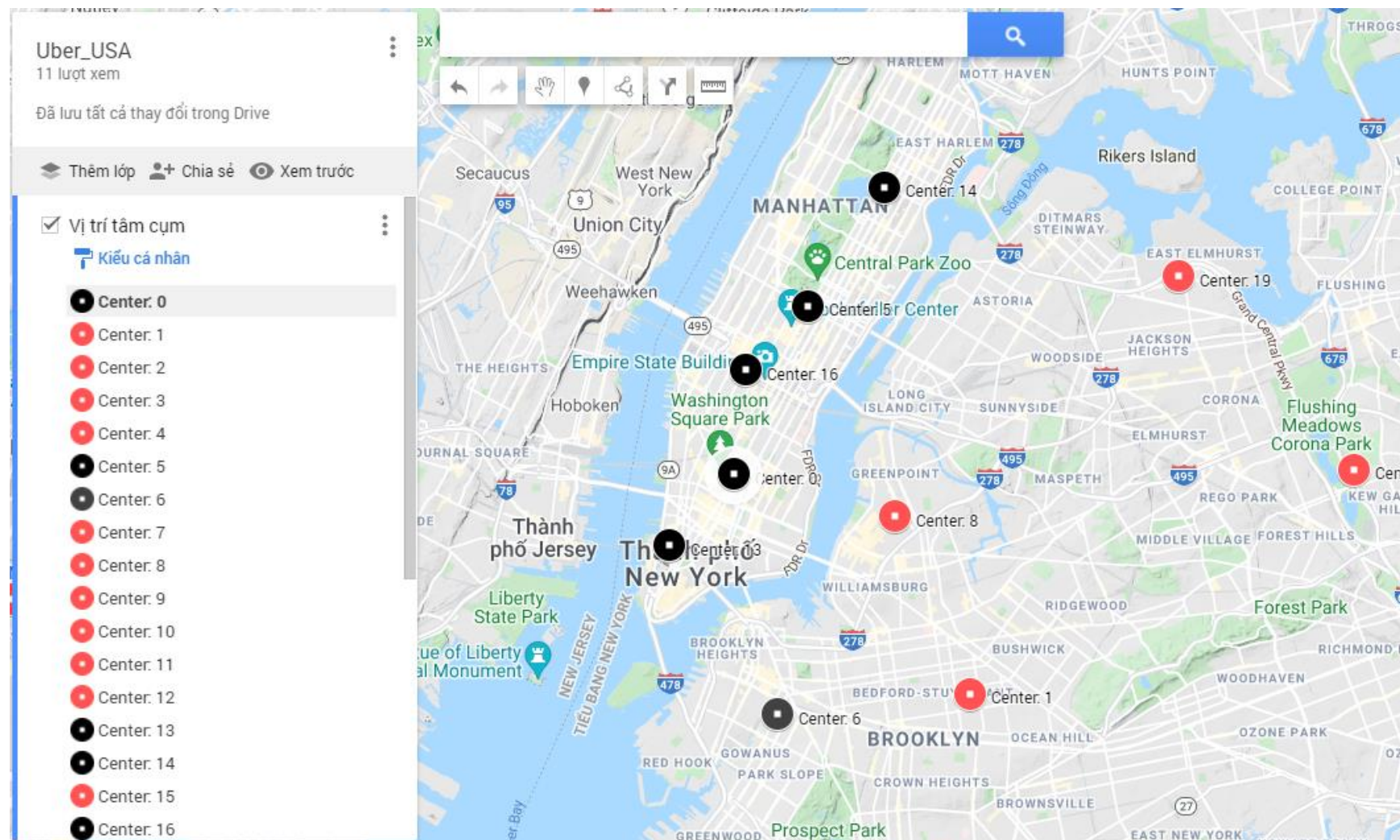
4.2 Phân tích dữ liệu

Các khu vực nào có lượng book xe cao nhất thuộc các cụm 0, 16, 5, 13, 14, 6 (>50.000 tháng 08)

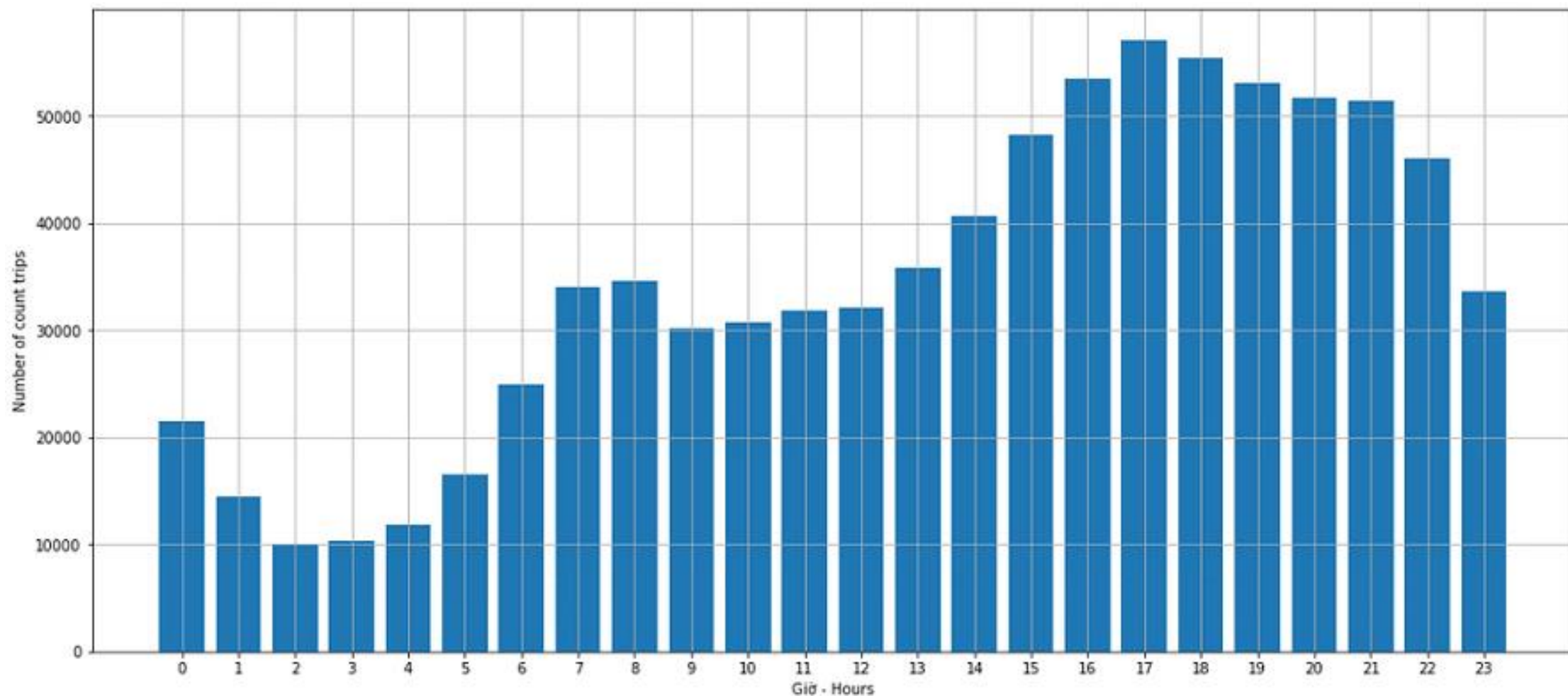
cid	count
0	171704
16	143994
5	143125
13	77351
14	67268
6	52053
8	48184
19	27148
1	26628
3	22481
10	12779
12	10829
17	10394
15	7786
7	2560
11	2321
4	1145
9	730
2	486
18	309



1. Khu vực nào có lượng book xe cao nhất?

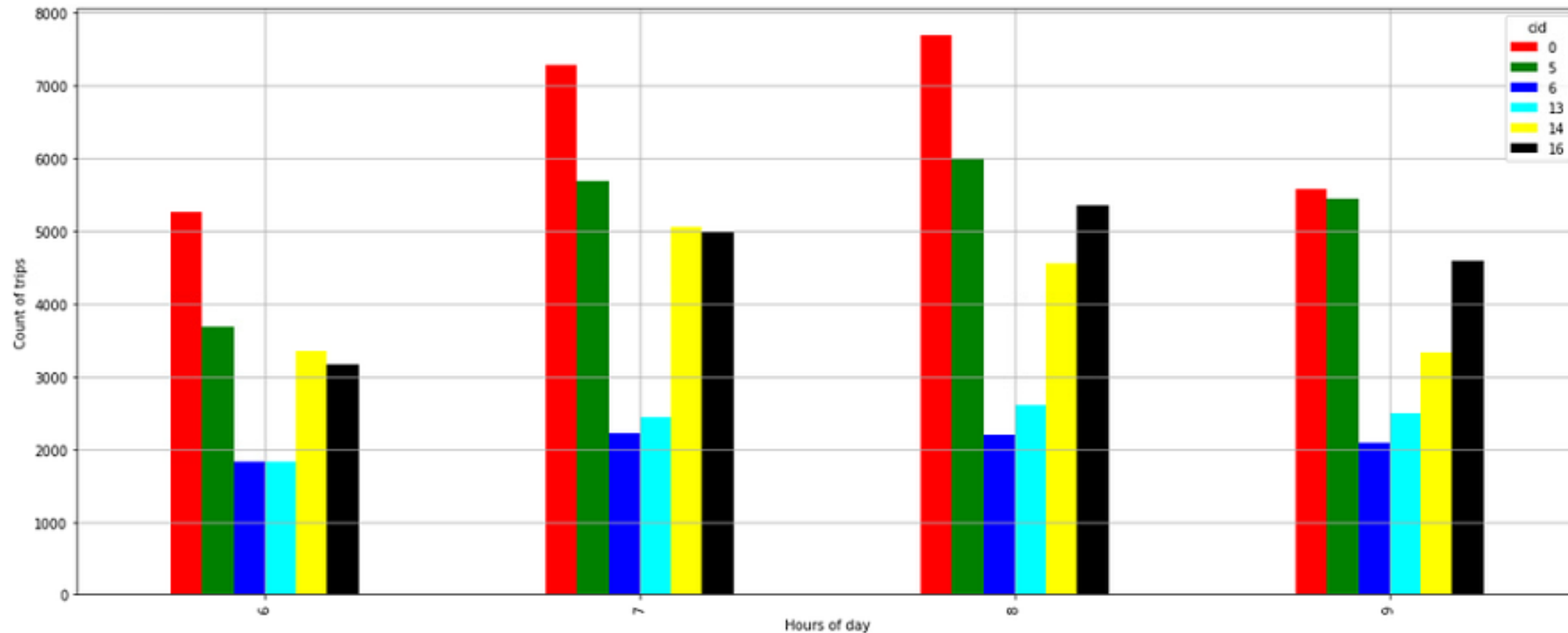


2. Thời gian nào nào trong ngày khách đi xe nhiều nhất?



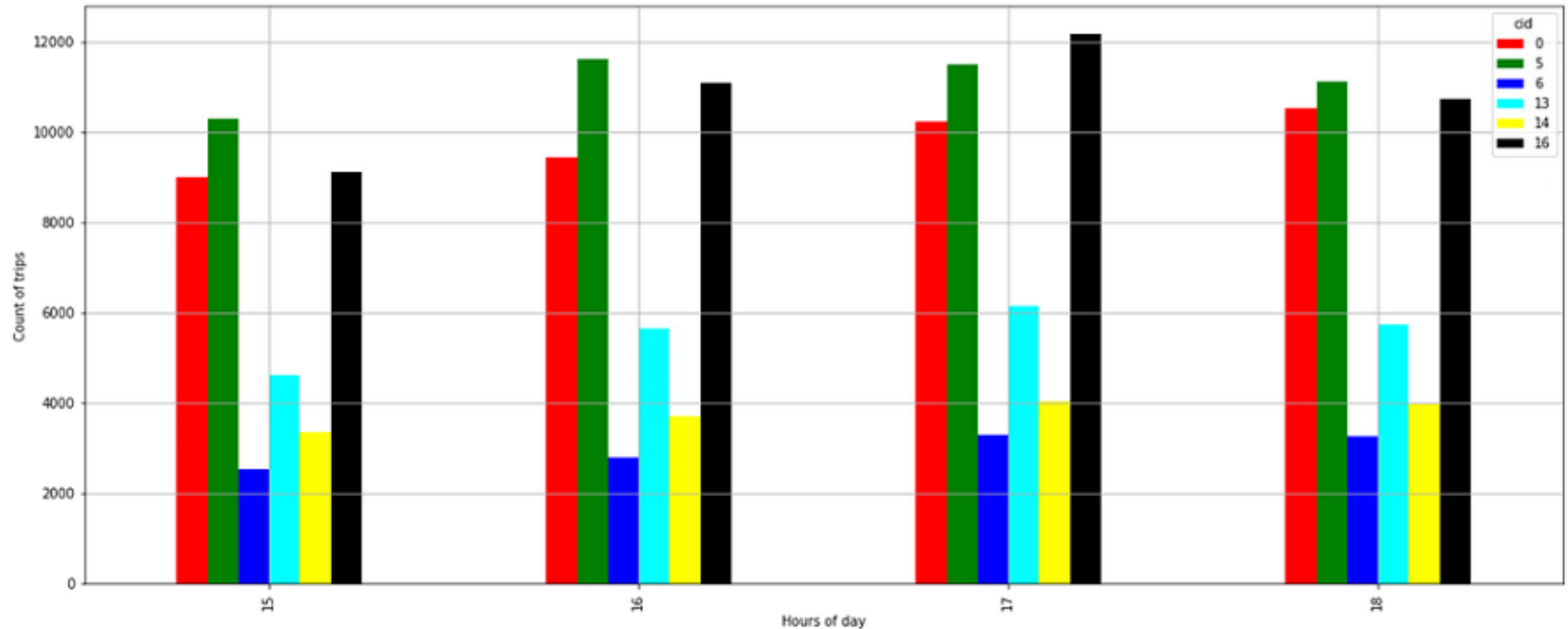
3. Số lượng book xe theo khung giờ của 6 cụm có mật độ cao nhất?

a. Buổi sáng (6, 7, 8, 9h)



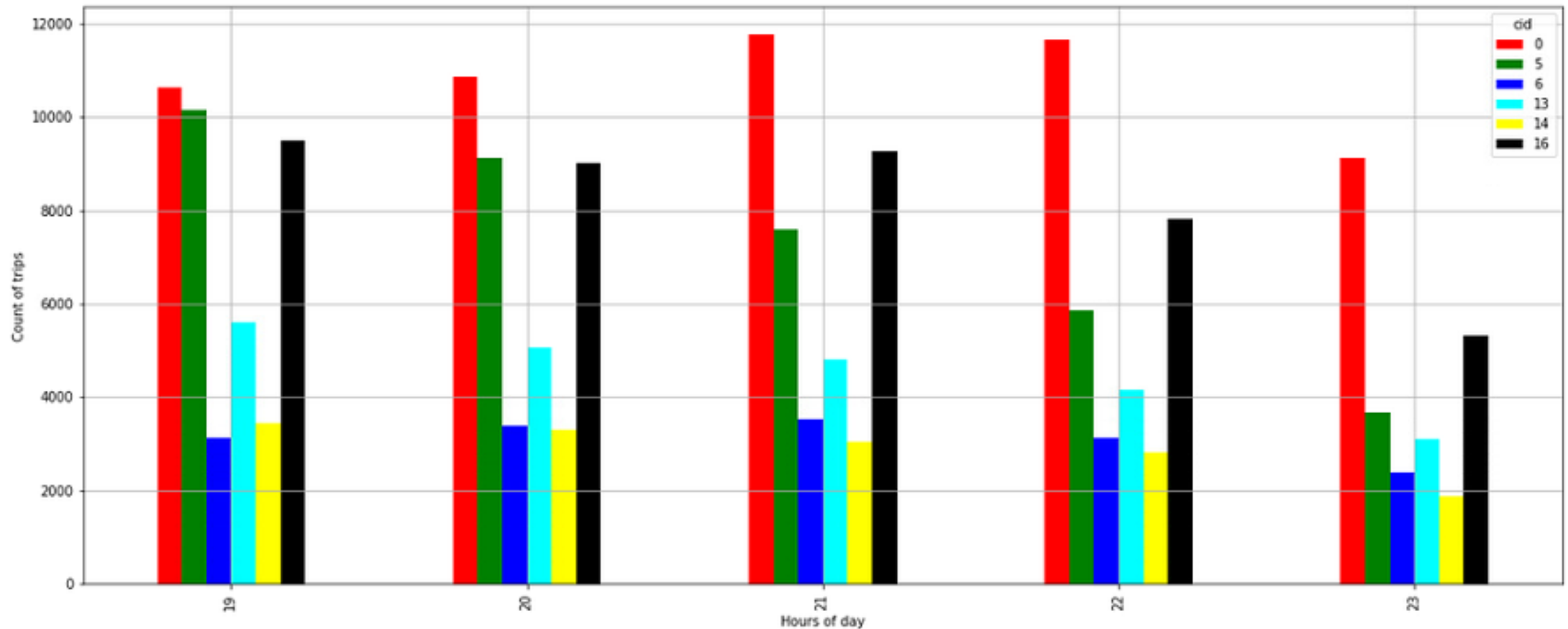
3. Số lượng book xe theo khung giờ của 6 cụm có mật độ cao nhất?

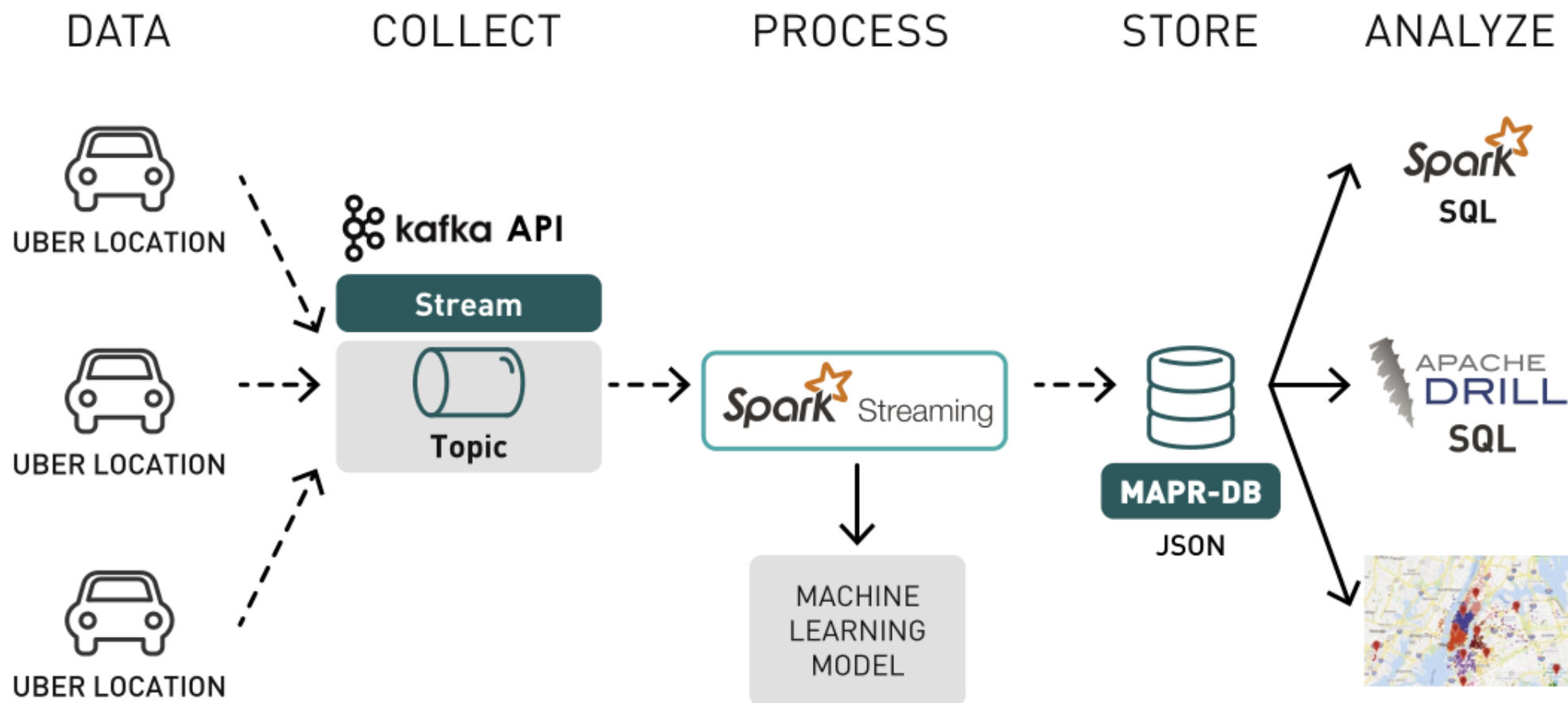
b. Buổi Chiều (15, 16, 17, 18h)



3. Số lượng book xe theo khung giờ của 6 cụm có mật độ cao nhất?

c. Buổi Tối (19, 20, 21, 22, 23h)





Thank you!