HỆ THỐNG LƯU TRỮ VÀ XỬ LÝ DỮ LIỆU THÔNG TIN CÁC CHUYẾN BAY

Môn học: Lưu trữ và xử lý dữ liệu lớn

Nhóm sinh viên

1. Vũ Trung Nghĩa - 20173284
2. Lê Vũ Lợi - 20173240
3. Đặng Lâm San - 20170111

Ngày 22 tháng 12 năm 2020

Outline

Đặt vấn đề

- Dữ liệu các chuyến bay trong thực tế thường đến dưới dạng bản ghi (PNR - Passenger Name Record) theo time series với số lượng rất lớn, vì vậy các hệ thống lưu trữ và xử lý dữ liệu truyền thống không còn hiệu quả.

- Cần xây dựng một hệ thống lưu trữ và xử lý dữ liệu phân tán cho dữ liệu lớn với các ràng buộc:
  - Đáp ứng các yêu cầu về tính toán và lưu trữ
  - Có khả năng chịu lỗi tốt khi các máy trong hệ thống không tin cậy
  - Dễ dàng mở rộng
  - Cho phép người dùng có thể xem dữ liệu real-time và thực hiện các truy vấn không real-time như visulize thống kê hay anomaly detection

Đặc tả một bản ghi

ArrivalTime – local time of arrival

BusinessLeisure – if the trip is for business or leisure

CabinCategory – cabin class

CreationDate –PNR creation date (Julian day)

CurrencyCode – 3-letter currency code of payment

DepartureTime – local time of departure

Destination – IATA code of arrival airport

OfficeIdCountry – country code of office placing the reser

Origin – IATA code of departure airport

TotalAmount – total reservation cost

nPAX – number of passengers

```
{
  "ID": 1188,
  "ArrivalTime": "1453042816",
  "BusinessLeisure": "B",
  "CabinCategory": "40",
  "CreationDate": "2457373",
  "CurrencyCode": "nan",
  "DepartureTime": "1452892672",
  "Destination": "TRD",
  "OfficeIdCountry": "NO",
  "Origin": "ALC",
  "TotalAmount": "nan",
  "nPAX": "1"
}
```

Kiến trúc tổng quan
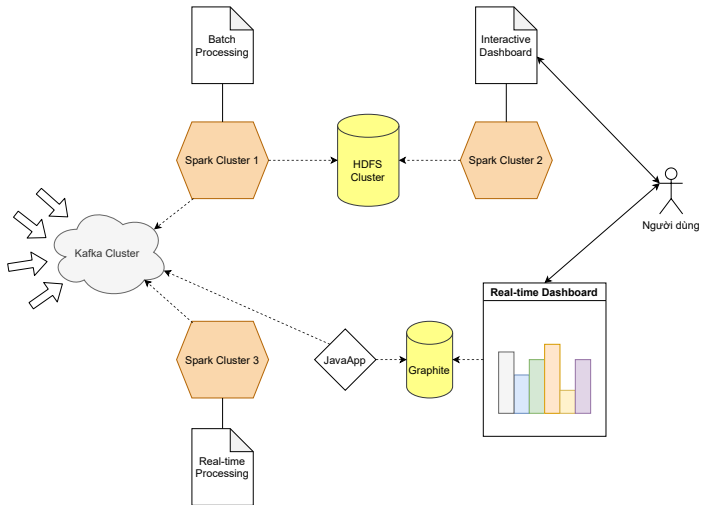
HDFS Cluster

```yaml
namenode:
  image: bde2020/hadoop-namenode:2.0.0-hadoop2.7.4-java8
  container_name: namenode
  environment:
    - CLUSTER_NAME=test
  env_file:
    - ./hadoop.env
  ports:
    - 8020:8020
    - 50070:50070
datanode-1:
  image: bde2020/hadoop-datanode:2.0.0-hadoop2.7.4-java8
  container_name: datanode-1
  environment:
    SERVICE_PRECONDITION: "namenode:50070"
  env_file:
    - ./hadoop.env
  ports:
    - 50075:50075

datanode-2:
  image: bde2020/hadoop-datanode:2.0.0-hadoop2.7.4-java8
  container_name: datanode-2
  environment:
    SERVICE_PRECONDITION: "namenode:50070"
  env_file:
    - ./hadoop.env
  ports:
    - 50076:50075
datanode-3:
  image: bde2020/hadoop-datanode:2.0.0-hadoop2.7.4-java8
  container_name: datanode-3
  environment:
    SERVICE_PRECONDITION: "namenode:50070"
  env_file:
    - ./hadoop.env
  ports:
    - 50077:50075
```

Kafka Cluster

```yaml
zookeeper:
  image: zookeeper:3.4.10
  container_name: zookeeper
  environment:
    ZOO_MY_ID: 1
    ZOO_SERVERS: server.1=0.0.0.0:2888:3888
    ZOO_TICK_TIME: 15000
  ports:
    - 2181:2181
kafka-broker-1:
  image: wurstmeister/kafka:2.12-2.2.1
  container_name: kafka-broker-1
  depends_on:
    - zookeeper
  ports:
    - "9092:9092"
  environment:
    - KAFKA_ZOOKEEPER_CONNECT=zookeeper:2181
    - ALLOW_PLAINTEXT_LISTENER=yes
    - KAFKA_ADVERTISED_LISTENERS=INSIDE://kafka-broker-1:9093,OUTSIDE://nghiavt:9092
    - KAFKA_LISTENER_SECURITY_PROTOCOL_MAP=INSIDE:PLAINTEXT,OUTSIDE:PLAINTEXT
    - KAFKA_LISTENERS=INSIDE://kafka-broker-1:9093,OUTSIDE://0.0.0.0:9092
    - KAFKA_INTER_BROKER_LISTENER_NAME=INSIDE

kafka-broker-2:
  image: wurstmeister/kafka:2.12-2.2.1
  container_name: kafka-broker-2
  depends_on:
    - zookeeper
  ports:
    - "9094:9094"
  environment:
    - KAFKA_ZOOKEEPER_CONNECT=zookeeper:2181
    - ALLOW_PLAINTEXT_LISTENER=yes
    - KAFKA_ADVERTISED_LISTENERS=INSIDE://kafka-broker-2:9093,OUTSIDE://nghiavt:9094
    - KAFKA_LISTENER_SECURITY_PROTOCOL_MAP=INSIDE:PLAINTEXT,OUTSIDE:PLAINTEXT
    - KAFKA_LISTENERS=INSIDE://kafka-broker-2:9093,OUTSIDE://0.0.0.0:9094
    - KAFKA_INTER_BROKER_LISTENER_NAME=INSIDE
kafka-broker-3:
  image: wurstmeister/kafka:2.12-2.2.1
  container_name: kafka-broker-3
  depends_on:
    - zookeeper
  ports:
    - "9095:9095"
  environment:
    - KAFKA_ZOOKEEPER_CONNECT=zookeeper:2181
    - ALLOW_PLAINTEXT_LISTENER=yes
    - KAFKA_ADVERTISED_LISTENERS=INSIDE://kafka-broker-3:9093,OUTSIDE://nghiavt:9095
    - KAFKA_LISTENER_SECURITY_PROTOCOL_MAP=INSIDE:PLAINTEXT,OUTSIDE:PLAINTEXT
    - KAFKA_LISTENERS=INSIDE://kafka-broker-3:9093,OUTSIDE://0.0.0.0:9095
    - KAFKA_INTER_BROKER_LISTENER_NAME=INSIDE
```

Spark Cluster

```yaml
pre-batch-processing-spark-master:
  image: spark-master:spark2.4.1-python3.7-hadoop2.7
  container_name: pre-batch-processing-spark-master
  ports:
    - "8082:8080"
    - "7077:7077"
  environment:
    - INIT_DAEMON_STEP=setup_spark
pre-batch-processing-spark-worker-1:
  image: spark-worker:spark2.4.1-python3.7-hadoop2.7
  container_name: pre-batch-processing-spark-worker-1
  depends_on:
    - pre-batch-processing-spark-master
  environment:
    - "SPARK_MASTER=spark://pre-batch-processing-spark-master:7077"
    - "SPARK_WORKER_CORES=1"
    - "SPARK_WORKER_MEMORY=1G"
    - "SPARK_DRIVER_MEMORY=128m"
    - "SPARK_EXECUTOR_MEMORY=256m"
pre-batch-processing-spark-worker-2:
  image: spark-worker:spark2.4.1-python3.7-hadoop2.7
  container_name: pre-batch-processing-spark-worker-2
  depends_on:
    - pre-batch-processing-spark-master
  environment:
    - "SPARK_MASTER=spark://pre-batch-processing-spark-master:7077"
    - "SPARK_WORKER_CORES=1"
    - "SPARK_WORKER_MEMORY=1G"
    - "SPARK_DRIVER_MEMORY=128m"
    - "SPARK_EXECUTOR_MEMORY=256m"
```

Notebook

```yaml
dashboard-notebook:
  image: dashboard-notebook:spark2.4.1-python3.7-hadoop2.7-hive
  container_name: dashboard-notebook
  ports:
    - "8888:8888"
  volumes:
    - $PWD/src/post-batch-processing-spark:/home/jovyan/work
  environment:
    - JUPYTER_TOKEN=admin

spark-notebook-batch-branch:
  image: pyspark-notebook:spark2.4.1-python3.7-hadoop2.7
  container_name: spark-notebook-batch-branch
  ports:
    - "8889:8888"
  volumes:
    - $PWD/src/pre-batch-processing-spark:/home/jovyan/work
  environment:
    - JUPYTER_TOKEN=admin

spark-notebook-speed-branch:
  image: pyspark-notebook:spark2.4.1-python3.7-hadoop2.7
  container_name: spark-notebook-speed-branch
  ports:
    - "8890:8888"
  volumes:
    - $PWD/src/speed-processing-spark:/home/jovyan/work
  environment:
    - JUPYTER_TOKEN=admin
```

Danh sách containers

```
PORTS                                                                              NAMES
8081/tcp, 8888/tcp                                                                 pre-batch-processing-spark-worker-1
8081/tcp, 8888/tcp                                                                 pre-batch-processing-spark-worker-2
8081/tcp, 8888/tcp                                                                 post-batch-processing-spark-worker-2
8081/tcp, 8888/tcp                                                                 post-batch-processing-spark-worker-1
8081/tcp, 8888/tcp                                                                 speed-processing-spark-worker-2
8081/tcp, 8888/tcp                                                                 speed-processing-spark-worker-1
0.0.0.0:9095->9095/tcp                                                             kafka-broker-3
0.0.0.0:9092->9092/tcp                                                             kafka-broker-1
0.0.0.0:9094->9094/tcp                                                             kafka-broker-2
6066/tcp, 8888/tcp, 0.0.0.0:7078->7077/tcp, 0.0.0.0:8083->8080/tcp                 post-batch-processing-spark-master
6066/tcp, 8888/tcp, 0.0.0.0:7077->7077/tcp, 0.0.0.0:8082->8080/tcp                 pre-batch-processing-spark-master
0.0.0.0:8020->8020/tcp, 0.0.0.0:50070->50070/tcp                                   namenode
0.0.0.0:8890->8888/tcp                                                             spark-notebook-speed-branch
0.0.0.0:8888->8888/tcp                                                             dashboard-notebook
0.0.0.0:50076->50075/tcp                                                           datanode-2
0.0.0.0:50077->50075/tcp                                                           datanode-3
0.0.0.0:50075->50075/tcp                                                           datanode-1
0.0.0.0:8889->8888/tcp                                                             spark-notebook-batch-branch
2888/tcp, 0.0.0.0:2181->2181/tcp, 3888/tcp                                         zookeeper
6066/tcp, 8888/tcp, 0.0.0.0:7079->7077/tcp, 0.0.0.0:8084->8080/tcp                 speed-processing-spark-master
```

Spark cluster 1

Spark Master at spark://ddb96567726c:7077

2.4.1

URL: spark://ddb96567726c:7077
Alive Workers: 2
Cores in use: 2 Total, 2 Used
Memory in use: 2.0 GB Total, 2.0 GB Used
Applications: 1 Running, 0 Completed
Drivers: 0 Running, 0 Completed
Status: ALIVE

▼ **Workers (2)**

Worker Id	Address
worker-20201221141801-172.18.0.21-34867	172.18.0.21:34867
worker-20201221141802-172.18.0.20-46809	172.18.0.20:46809

▼ **Running Applications (1)**

Application ID		Name	Cores ▲
app-20201221142802-0000	(kill)	SparkBatchStreamingKafka	2

▼ **Completed Applications (0)**

Application ID	Name	Cores	Memory per Executor

Spark cluster 2

Spark Master at spark://b23b29c5ddcf:7077

URL: spark://b23b29c5ddcf:7077
Alive Workers: 2
Cores in use: 2 Total, 2 Used
Memory in use: 2.0 GB Total, 2.0 GB Used
Applications: 1 Running, 0 Completed
Drivers: 0 Running, 0 Completed
Status: ALIVE

▼ **Workers (2)**

Worker Id	Address
worker-20201221141801-172.18.0.18-33609	172.18.0.18:33609
worker-20201221141801-172.18.0.19-46763	172.18.0.19:46763

▼ **Running Applications (1)**

Application ID		Name	Cores	Memory pe
app-20201221142928-0000	(kill)	DashBoard	2	1024.0 MB

▶ **Completed Applications (0)**

HDFS

Datanode Information
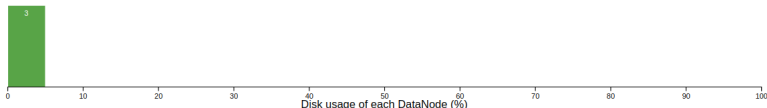
Datanode usage histogram

Disk usage of each DataNode (%)

In operation

Node	Last contact	Admin State	Capacity	Used	Non DFS Used	Remaining	Blocks	Block pool used	Failed Volumes	Version
b5f5307eae55:50010 (172.18.0.8:50010)	1	In Service	355.6 GB	28 KB	83.27 GB	254.2 GB	0	28 KB (0%)	0	2.7.4
d3d6d5b8d26e:50010 (172.18.0.6:50010)	1	In Service	355.6 GB	28 KB	83.27 GB	254.2 GB	0	28 KB (0%)	0	2.7.4
58b814f6f2c3:50010 (172.18.0.5:50010)	0	In Service	355.6 GB	28 KB	83.27 GB	254.2 GB	0	28 KB (0%)	0	2.7.4

HDFS

Tạo topic và gửi 10000 bản ghi

```
Topic: trips    PartitionCount: 3      ReplicationFactor: 3    Configs: segment.bytes=1073741824
        Topic: trips    Partition: 0    Leader: 1003    Replicas: 1003,1002,1001      Isr: 1003,1002,1001
        Topic: trips    Partition: 1    Leader: 1002    Replicas: 1002,1001,1003      Isr: 1002,1001,1003
        Topic: trips    Partition: 2    Leader: 1001    Replicas: 1001,1003,1002      Isr: 1001,1003,1002
```

```
Sent 10000 records in 25.57376456260681 seconds
Sending rate: 391.02573168369975 records/s
```

Partition 0 trên 3 brokers

%H!◆_v◆◆xuv◆◆xu◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆{"ID": 9985, "ArrivalTime": "1444902656",
"BusinessLeisure": "nan", "CabinCategory": "40", "CreationDate": "24571
62", "CurrencyCode": "nan", "DepartureTime": "1445120256", "Destination
": "JNB", "OfficeIdCountry": "PL", "Origin": "PRG", "TotalAmount": "nan",
&H`a◆◆v◆◆x~v◆◆x-◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆{"ID": 9987, "ArrivalTime": "1440208000"
, "BusinessLeisure": "nan", "CabinCategory": "40", "CreationDate": "2457
247", "CurrencyCode": "nan", "DepartureTime": "1440858112", "Destination
": "HLA", "OfficeIdCountry": "ZA", "Origin": "INN", "TotalAmount": "nan"
'FGL◆ev◆◆x◆v◆◆x◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆{"ID": 9988, "ArrivalTime": "1444797696"
, "BusinessLeisure": "B", "CabinCategory": "40", "CreationDate": "245735
1", "CurrencyCode": "nan", "DepartureTime": "1445876864", "Destination":
"BOM", "OfficeIdCountry": "IN", "Origin": "BLR", "TotalAmount": "nan",
"nPAX": "1"}bash-4.4# ☐

%H!◆_v◆◆xuv◆◆xu◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆{"ID": 9985, "ArrivalTime": "1444902656",
"BusinessLeisure": "nan", "CabinCategory": "40", "CreationDate": "24571
62", "CurrencyCode": "nan", "DepartureTime": "1445120256", "Destination
": "JNB", "OfficeIdCountry": "PL", "Origin": "PRG", "TotalAmount": "nan",
&H`a◆◆v◆◆x~v◆◆x-◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆{"ID": 9987, "ArrivalTime": "1440208000"
, "BusinessLeisure": "nan", "CabinCategory": "40", "CreationDate": "2457
247", "CurrencyCode": "nan", "DepartureTime": "1440858112", "Destination
": "HLA", "OfficeIdCountry": "ZA", "Origin": "INN", "TotalAmount": "nan"
'FGL◆ev◆◆x◆v◆◆x◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆{"ID": 9988, "ArrivalTime": "1444797696"
, "BusinessLeisure": "B", "CabinCategory": "40", "CreationDate": "245735
1", "CurrencyCode": "nan", "DepartureTime": "1445876864", "Destination":
"BOM", "OfficeIdCountry": "IN", "Origin": "BLR", "TotalAmount": "nan",
"nPAX": "1"}bash-4.4# ☐

%H!◆_v◆◆xuv◆◆xu◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆{"ID": 9985, "ArrivalTime": "1444902656",
"BusinessLeisure": "nan", "CabinCategory": "40", "CreationDate": "24571
62", "CurrencyCode": "nan", "DepartureTime": "1445120256", "Destination
": "JNB", "OfficeIdCountry": "PL", "Origin": "PRG", "TotalAmount": "nan",
&H`a◆◆v◆◆x~v◆◆x-◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆{"ID": 9987, "ArrivalTime": "1440208000"
, "BusinessLeisure": "nan", "CabinCategory": "40", "CreationDate": "2457
247", "CurrencyCode": "nan", "DepartureTime": "1440858112", "Destination
": "HLA", "OfficeIdCountry": "ZA", "Origin": "INN", "TotalAmount": "nan"
'FGL◆ev◆◆x◆v◆◆x◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆{"ID": 9988, "ArrivalTime": "1444797696"
, "BusinessLeisure": "B", "CabinCategory": "40", "CreationDate": "245735
1", "CurrencyCode": "nan", "DepartureTime": "1445876864", "Destination":
"BOM", "OfficeIdCountry": "IN", "Origin": "BLR", "TotalAmount": "nan",
"nPAX": "1"}bash-4.4# █

Partition 0,1,2 trên brokers

%H!◊_v◊◊xuv◊◊xu◊◊◊◊◊◊◊◊◊◊◊◊◊◊{"ID": 9985, "ArrivalTime": "1444902656",
"BusinessLeisure": "nan", "CabinCategory": "40", "CreationDate": "24571
62", "CurrencyCode": "nan", "DepartureTime": "1445120256", "Destination
": "JNB", "OfficeIdCountry": "PL", "Origin": "PRG", "TotalAmount": "nan",
&H`◊◊v◊◊x~v◊◊x-◊◊◊◊◊◊◊◊◊◊◊◊◊◊{"ID": 9987, "ArrivalTime": "1440208000"
, "BusinessLeisure": "nan", "CabinCategory": "40", "CreationDate": "2457
247", "CurrencyCode": "nan", "DepartureTime": "1440858112", "Destination
": "HLA", "OfficeIdCountry": "ZA", "Origin": "INN", "TotalAmount": "nan"
'FGL◊ev◊◊x◊◊x◊◊◊◊◊◊◊◊◊◊◊◊◊◊{"ID": 9988, "ArrivalTime": "1444797696"
, "BusinessLeisure": "B", "CabinCategory": "40", "CreationDate": "245735
1", "CurrencyCode": "nan", "DepartureTime": "1445876864", "Destination":
"BOM", "OfficeIdCountry": "IN", "Origin": "BLR", "TotalAmount": "nan",
"nPAX": "1"}bash-4.4# ▯

EH◊%◊◊v◊◊x◊v◊◊x◊◊◊◊◊◊◊◊◊◊◊◊◊{"ID": 9992, "ArrivalTime": "1441638784"
, "BusinessLeisure": "nan", "CabinCategory": "20", "CreationDate": "2457
152", "CurrencyCode": "EUR", "DepartureTime": "1439553536", "Destination
": "JFK", "OfficeIdCountry": "ES", "Origin": "MAD", "TotalAmount": "0.0"
FFz◊◊Tv◊◊x◊v◊◊x◊◊◊◊◊◊◊◊◊◊◊◊◊{"ID": 9996, "ArrivalTime": "1441516928"
, "BusinessLeisure": "L", "CabinCategory": "40", "CreationDate": "245720
7", "CurrencyCode": "nan", "DepartureTime": "1440920448", "Destination":
"BLR", "OfficeIdCountry": "MA", "Origin": "CMN", "TotalAmount": "nan",
GH0◊◊öv◊◊x◊v◊◊x◊◊◊◊◊◊◊◊◊◊◊◊◊{"ID": 9999, "ArrivalTime": "1438053120",
"BusinessLeisure": "nan", "CabinCategory": "40", "CreationDate": "24571
69", "CurrencyCode": "nan", "DepartureTime": "1438486912", "Destination
": "TNR", "OfficeIdCountry": "KE", "Origin": "LPA", "TotalAmount": "nan",
"nPAX": "1"}bash-4.4# ▯
ssLeisure": "nan", "CabinCategory": "40", "CreationDate": "2457058", "Cu
rrencyCode": "nan", "DepartureTime": "1426283392", "Destination": "MAD",
"OfficeIdCountry": "BE", "Origin": "BIO", "TotalAmount": "nan", "nPAX":
"1"}
◊]&0◊v◊◊x◊v◊◊x◊◊◊◊◊◊◊◊◊◊◊◊◊{"ID": 9997, "ArrivalTime": "1452245
376", "BusinessLeisure": "nan", "CabinCategory": "40", "CreationDate": "
2457297", "CurrencyCode": "nan", "DepartureTime": "1449478144", "Destina
tion": "CCS", "OfficeIdCountry": "DE", "Origin": "FRA", "TotalAmount": "
nan", "nPAX": "1"}◊◊{"ID": 9998, "ArrivalTime": "1417529216", "BusinessL
eisure": "L", "CabinCategory": "40", "CreationDate": "2456918", "Currenc
yCode": "nan", "DepartureTime": "1415165568", "Destination": "MLE", "Off
iceIdCountry": "DK", "Origin": "CPH", "TotalAmount": "nan", "nPAX": "2"}
bash-4.4# ▯

Kết quả xử lý và nhận trên Batch Processing Notebook

```python
ks = KafkaUtils.createDirectStream(
    ssc, ['trips'], {'metadata.broker.list': 'kafka-broker-1:9093,kafka-broker-2:9093,kafka-broker-3:9093'})
lines = ks.map(lambda x: x[1])

transform1 = lines.map(lambda tripInfo: json_to_list(tripInfo))
transform1.foreachRDD(handle_rdd1)

transform2 = lines.map(lambda tripInfo: json_to_processed_data(tripInfo))
transform2.foreachRDD(handle_rdd2)

ssc.start()
ssc.awaitTermination()
```

```
Recieved 1191 records - transfrom 1
Recieved 1191 records - transfrom 2
Recieved 1907 records - transfrom 1
Recieved 1907 records - transfrom 2
Recieved 1874 records - transfrom 1
Recieved 1874 records - transfrom 2
Recieved 1904 records - transfrom 1
Recieved 1904 records - transfrom 2
Recieved 2050 records - transfrom 1
Recieved 2050 records - transfrom 2
Recieved 1074 records - transfrom 1
Recieved 1074 records - transfrom 2
```

Dữ liệu lưu trong HDFS

Hadoop   Overview   Datanodes   Snapshot   Startup Progress   Utilities ▾

Browse Directory

/trips   Go!

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
drwxr-xr-x	jovyan	supergroup	0 B	12/21/2020, 9:43:41 PM	0	0 B	processed_trips.parquet
drwxr-xr-x	jovyan	supergroup	0 B	12/21/2020, 9:43:40 PM	0	0 B	trips.parquet

Hadoop, 2017.

Dữ liệu lưu trong HDFS

Đọc dữ liệu từ HDFS và visualize

Read data from parquet file "trips.parquet" in hdfs

```
df = ss.read.parquet("hdfs://namenode:9000/trips/trips.parquet")
```

```python
print(f"Number of records: {df.count()}")
df = df.sort('ArrivalTime')
```

```
Number of records: 10000
```

Data Mining

Distribution of trips over time

```python
arrivalTime = df.select('ArrivalTime').toPandas()['ArrivalTime'].astype('int64')
sns.distplot(arrivalTime)
```
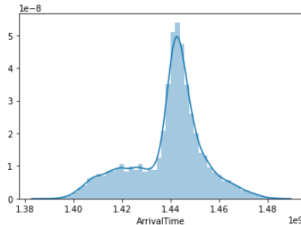
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fee8f9b4850>
```

Đưa dữ liệu real time từ kafka => kafka

- Chương trình spark đọc dữ liệu từ kafka, xử lý và đẩy vào kafka

```python
ss = SparkSession.Builder() \
    .appName("SparkBatchStreamingKafka") \
    .master("spark://speed-processing-spark-master:7077") \
    .config("spark.jars", "./spark-streaming-kafka-0-10-assembly_2.11-2.4.1.jar,./kafka-clients-0.10.1.0.jar,./spark-sql-kafka-0-10_2.11-2.4.1.jar") \
    .config("spark.sql.warehouse.dir", "hdfs://namenode:9000/") \
    .getOrCreate()

df = ss \
    .readStream \
    .format("kafka") \
    .option("kafka.bootstrap.servers", "kafka-broker-1:9093,kafka-broker-2:9093,kafka-broker-3:9093") \
    .option("partition.assignment.strategy", "none") \
    .option("subscribe", "trips") \
    .load()

import random

def transform_window(s):
    """
    s = Row(start=datetime.datetime(2020, 12, 21, 17, 9, 30), end=datetime.datetime(2020, 12, 21, 17, 9, 40))
    """
    return str(int(s.end.timestamp()))

def transform_count(s):
    """
    s = 941
    """
    return str(s)
udf_transform_window = udf(transform_window)
udf_transform_count = udf(transform_count)

query = df.withWatermark("timestamp", "15 seconds") \
    .groupBy(window("timestamp", "5 seconds", "5 seconds")) \
    .count() \
    .withColumn("count", udf_transform_count("count")) \
    .withColumn("window", udf_transform_window("window")) \
    .withColumn('value', sf.concat(sf.col('window'),sf.lit('_'), sf.col('count'))) \
    .writeStream \
    .format("kafka") \
    .option("kafka.bootstrap.servers", "kafka-broker-1:9093,kafka-broker-2:9093,kafka-broker-3:9093") \
    .option("topic", "real-time-statistic") \
    .option("checkpointLocation", "/tmp/checkpoint") \
    .outputMode("append") \
    .option("truncate", False) \
    .start()
query.awaitTermination()
```

Đưa dữ liệu real time vào graphite

- Chương trình chạy java để đẩy dữ liệu từ kafka vào graphite

Dữ liệu sau khi được đưa vào graphite
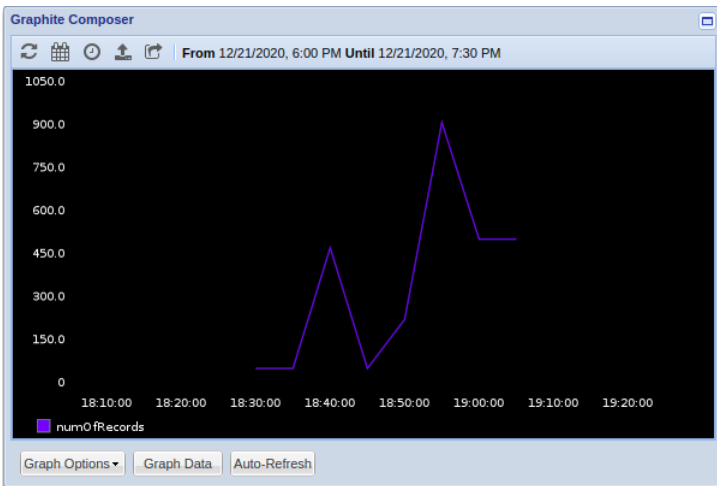
Visualize dữ liệu của graphite với grafana
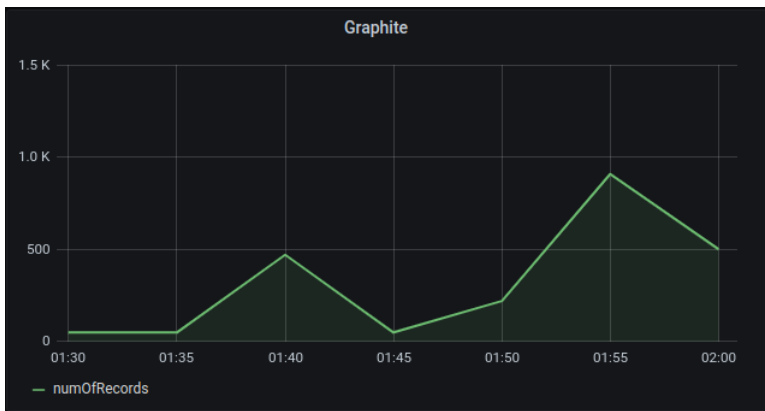
Đánh giá khả năng chịu tải của kafka

Delay (s)	3 brokers			2 brokers		
	Tốc độ gửi (records/s)	Số gói tin nhận	Số gói tin bị mất	Tốc độ gửi (records/s)	Số gói tin nhận	Số gói tin bị mất
0	4695.9	9980	20	4245.8	9896	104
	4399.7	10000	0	4705.8	10000	0
	4698.1	9868	132	4979.7	10000	0
0.0000001	798.8	9989	11	1228.1	10000	0
	1044	10000	0	1420	10000	0
	1092.9	10000	0	1448.8	10000	0
0.000001	1207	9997	3	1342.9	10000	0
	1133.5	9998	2	1432.1	10000	0
	1216.3	10000	0	1215	10000	0
0.00001	1034.3	9998	2	1354.2	10000	0
	1084.4	9997	3	1296.1	10000	0
	1100.6	9999	1	1363.1	10000	0
0.0001	834.4	10000	0	884.5	10000	0
	900.1	10000	0	939.7	10000	0
	908.3	10000	0	923.3	10000	0

Đánh giá khả năng chịu lỗi của kafka

```
Topic: trips    PartitionCount: 3      ReplicationFactor: 3    Configs: segment.bytes=1073741824
        Topic: trips    Partition: 0    Leader: 1002    Replicas: 1002,1001,1003    Isr: 1002,1003,1001
        Topic: trips    Partition: 1    Leader: 1003    Replicas: 1001,1003,1002    Isr: 1003,1002,1001
        Topic: trips    Partition: 2    Leader: 1003    Replicas: 1003,1002,1001    Isr: 1003,1002,1001


Topic: trips    PartitionCount: 3      ReplicationFactor: 3    Configs: segment.bytes=1073741824
        Topic: trips    Partition: 0    Leader: 1002    Replicas: 1002,1001,1003    Isr: 1002,1003
        Topic: trips    Partition: 1    Leader: 1003    Replicas: 1001,1003,1002    Isr: 1003,1002
        Topic: trips    Partition: 2    Leader: 1003    Replicas: 1003,1002,1001    Isr: 1003,1002
```

Read data from parquet file "trips.parquet" in hdfs

```python
df = ss.read.parquet("hdfs://namenode:9000/trips/trips.parquet")
```

```python
print(f"Number of records: {df.count()}")
df = df.sort('ArrivalTime')
```
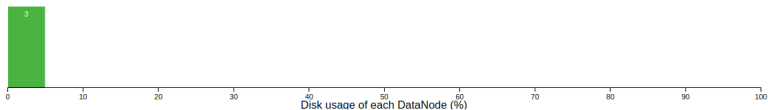
```
Number of records: 10000
```

Đánh giá khả năng chịu lỗi của HDFS

Datanode Information

Datanode usage histogram

Disk usage of each DataNode (%)

In operation

Node	Last contact	Admin State	Capacity	Used	Non DFS Used	Remaining	Blocks	Block pool used	Failed Volumes	Version
75ef3df56e50:50010 (172.18.0.9:50010)	2	In Service	355.6 GB	1.85 MB	83.93 GB	253.54 GB	36	1.85 MB (0%)	0	2.7.4
c4851725fa5e:50010 (172.18.0.12:50010)	2	In Service	355.6 GB	1.85 MB	83.93 GB	253.54 GB	36	1.85 MB (0%)	0	2.7.4
1d7debe0eb02:50010 (172.18.0.11:50010)	2	In Service	355.6 GB	1.85 MB	83.93 GB	253.54 GB	36	1.85 MB (0%)	0	2.7.4

Đánh giá khả năng chịu lỗi của HDFS

Spark 2.4.1 **stderr log page for app-20201222020708-0000/1**

Back to Master

Showing 57501 Bytes: 0 - 57501 of 57501

```
        at java.lang.Thread.run(Thread.java:748)
20/12/22 02:07:57 WARN DFSClient: Failed to connect to /172.18.0.12:50010 for block, add to deadNodes and continue.
java.net.NoRouteToHostException: No route to host
java.net.NoRouteToHostException: No route to host
        at sun.nio.ch.SocketChannelImpl.checkConnect(Native Method)
        at sun.nio.ch.SocketChannelImpl.finishConnect(SocketChannelImpl.java:716)
        at org.apache.hadoop.net.SocketIOWithTimeout.connect(SocketIOWithTimeout.java:206)
        at org.apache.hadoop.net.NetUtils.connect(NetUtils.java:531)
        at org.apache.hadoop.hdfs.DFSClient.newConnectedPeer(DFSClient.java:3436)
        at org.apache.hadoop.hdfs.BlockReaderFactory.nextTcpPeer(BlockReaderFactory.java:777)
        at org.apache.hadoop.hdfs.BlockReaderFactory.getRemoteBlockReaderFromTcp(BlockReaderFactory.java:694)
        at org.apache.hadoop.hdfs.BlockReaderFactory.build(BlockReaderFactory.java:355)
        at org.apache.hadoop.hdfs.DFSInputStream.blockSeekTo(DFSInputStream.java:673)
        at org.apache.hadoop.hdfs.DFSInputStream.readWithStrategy(DFSInputStream.java:882)
        at org.apache.hadoop.hdfs.DFSInputStream.read(DFSInputStream.java:934)
        at org.apache.hadoop.hdfs.DFSInputStream.read(DFSInputStream.java:735)
        at java.io.FilterInputStream.read(FilterInputStream.java:83)
        at org.apache.parquet.io.DelegatingSeekableInputStream.read(DelegatingSeekableInputStream.java:61)
        at org.apache.parquet.bytes.BytesUtils.readIntLittleEndian(BytesUtils.java:80)
        at org.apache.parquet.hadoop.ParquetFileReader.readFooter(ParquetFileReader.java:520)
        at org.apache.parquet.hadoop.ParquetFileReader.readFooter(ParquetFileReader.java:505)
        at org.apache.parquet.hadoop.ParquetFileReader.readFooter(ParquetFileReader.java:499)
        at org.apache.parquet.hadoop.ParquetFileReader.readFooter(ParquetFileReader.java:448)
```

# Đánh giá khả năng chịu lỗi của HDFS

```
org.apache.spark.sql.execution.datasources.FileScanRDD$$anon$1.org$apache$spark$sql$execution$datasources$File
FileScanRDD.scala:124)
        at org.apache.spark.sql.execution.datasources.FileScanRDD$$anon$1.nextIterator(FileScanRDD.scala:177)
        at org.apache.spark.sql.execution.datasources.FileScanRDD$$anon$1.hasNext(FileScanRDD.scala:101)
        at org.apache.spark.sql.catalyst.expressions.GeneratedClass$GeneratedIteratorForCodegenStage1.scan_nex
        at org.apache.spark.sql.catalyst.expressions.GeneratedClass$GeneratedIteratorForCodegenStage1.agg_doAg
Source)
        at org.apache.spark.sql.catalyst.expressions.GeneratedClass$GeneratedIteratorForCodegenStage1.processN
        at org.apache.spark.sql.execution.BufferedRowIterator.hasNext(BufferedRowIterator.java:43)
        at org.apache.spark.sql.execution.WholeStageCodegenExec$$anonfun$13$$anon$1.hasNext(WholeStageCodegenE
        at scala.collection.Iterator$$anon$11.hasNext(Iterator.scala:409)
        at org.apache.spark.shuffle.sort.BypassMergeSortShuffleWriter.write(BypassMergeSortShuffleWriter.java:
        at org.apache.spark.scheduler.ShuffleMapTask.runTask(ShuffleMapTask.scala:99)
        at org.apache.spark.scheduler.ShuffleMapTask.runTask(ShuffleMapTask.scala:55)
        at org.apache.spark.scheduler.Task.run(Task.scala:121)
        at org.apache.spark.executor.Executor$TaskRunner$$anonfun$10.apply(Executor.scala:403)
        at org.apache.spark.util.Utils$.tryWithSafeFinally(Utils.scala:1360)
        at org.apache.spark.executor.Executor$TaskRunner.run(Executor.scala:409)
        at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149)
        at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624)
        at java.lang.Thread.run(Thread.java:748)
20/12/22 02:08:00 INFO DFSClient: Successfully connected to /172.18.0.11:50010 for BP-1005430138-172.18.0.10-
1608602506248:blk_1073741856_1032
20/12/22 02:08:00 INFO Executor: Finished task 1.0 in stage 1.0 (TID 2). 1759 bytes result sent to driver
```

Đánh giá khả năng chịu lỗi của HDFS
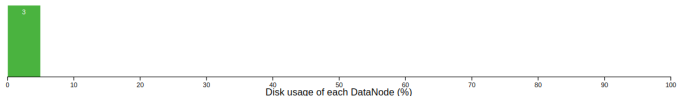
Datanode Information

Datanode usage histogram

Disk usage of each DataNode (%)

In operation

Node	Last contact	Admin State	Capacity	Used	Non DFS Used	Remaining	Blocks	Block pool used	Failed Volumes	Version
75ef3df56e50:50010 (172.18.0.9:50010)	0	In Service	355.6 GB	2.07 MB	83.93 GB	253.53 GB	36	2.07 MB (0%)	0	2.7.4
c4851725fa5e:50010 (172.18.0.12:50010)	507	In Service	355.6 GB	1.85 MB	83.93 GB	253.54 GB	36	1.85 MB (0%)	0	2.7.4
1d7debe0eb02:50010 (172.18.0.11:50010)	0	In Service	355.6 GB	2.07 MB	83.93 GB	253.53 GB	36	2.07 MB (0%)	0	2.7.4

Decommissioning

Đánh giá khả năng chịu lỗi của HDFS
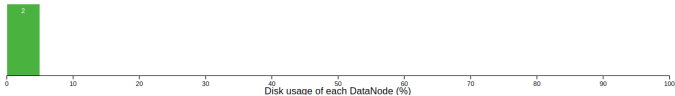
Datanode Information

Datanode usage histogram

Disk usage of each DataNode (%)

In operation

Node	Last contact	Admin State	Capacity	Used	Non DFS Used	Remaining	Blocks	Block pool used	Failed Volumes	Version
75ef3df56e50:50010 (172.18.0.9:50010)	0	In Service	355.6 GB	2.07 MB	83.94 GB	253.53 GB	36	2.07 MB (0%)	0	2.7.4
1d7debe0eb02:50010 (172.18.0.11:50010)	0	In Service	355.6 GB	2.07 MB	83.94 GB	253.53 GB	36	2.07 MB (0%)	0	2.7.4
c4851725fa5e:50010 (172.18.0.12:50010)	Tue Dec 22 2020 09:06:48 GMT+0700 (Indochina Time)	Dead	-	-	-	-	-	-	-	-

Đánh giá khả năng xử lý gói tin đến sai thứ tự của Spark

(*) source: https://spark.apache.org
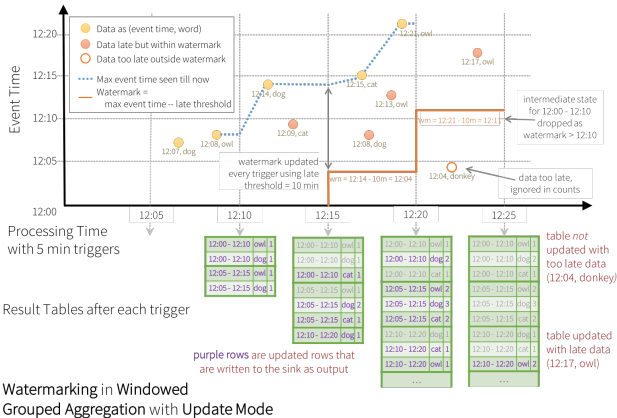
Đánh giá khả năng xử lý gói tin đến sai thứ tự của Spark

```python
query = df.withWatermark("timestamp", "20 seconds") \
        .groupBy(window("timestamp", "10 seconds", "10 seconds")) \
        .count() \
```

- Để đưa các gói tin về đúng khoảng thời gian chính xác mà nó được gửi, Structured Spark sử dụng Window Grouped Aggregation.

- Đối với các dạng event đến quá trễ và trở nên vô giá trị, Watermaking bỏ qua các gói tin đến trễ so với khoảng thời gian cho trước.

Đánh giá khả năng xử lý gói tin đến sai thứ tự của Spark

Hình: Bài toán đến số lượng bản ghi trong 10s gần nhất

- Sử dụng KafkaProducer để giả lập luồng dữ liệu có các events đến trễ.
- Cứ sau 20s, một nửa các gói tin được gửi đi từ Producer trong 10s tiếp theo được đánh dấu là trễ 10s, nửa còn lại gửi đi theo đúng thời gian thực.