TP3 : Kafka

1. Installation kafka

```
root@hadoop-master:~# jps
2000 Jps
532 SecondaryNameNode
249 NameNode
828 ResourceManager
1007 QuorumPeerMain
root@hadoop-master:~#
```

2. Première utilisation d'apache Kafka

Liste des topics créés

```
root@hadoop-master:~# kafka-topics.sh --list --bootstrap-server localhost:9092
Hello-Kafka
root@hadoop-master:~#
```

Description d'un topic

```
root@hadoop-master:~# kafka-topics.sh --describe --topic Hello-Kafka --bootstrap-server localhost:9092
Topic: Hello-Kafka      TopicId: FZ7Cq1oOQjyrn9gMWJCRHQ PartitionCount: 1       ReplicationFactor: 1    Configs:
        Topic: Hello-Kafka      Partition: 0    Leader: 0       Replicas: 0     Isr: 0
root@hadoop-master:~#
```

Ecrire des évènements dans un topic

```
root@hadoop-master:~# kafka-console-producer.sh --bootstrap-server localhost:9092 --topic Hello-Kafka
>ahmed
>qais
>tp3
>
```

A chaque fois qu'un ajout se fait sur le client producer, on le reçoit sur le consumer.

```
>> docker exec -it hadoop-master bash
root@hadoop-master:~# kafka-console-consumer.sh --bootstrap-server localhost:9092 --topic Hello-Kafka --from-beginning
hello
kafka
hadoop
hive
ahmed
qais
tp3
```

3. Création d'une application kafka

- ## Création du Producer

```
PS C:\Users\ahmed\Downloads\LabsBigData\lab3_kafka\kafka_lab> mvn clean package
[INFO] Scanning for projects...
[INFO]
[INFO] ----------------------< edu.ensias.kafka:kafka_lab >----------------------
[INFO] Building kafka_lab 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] --------------------------------[ jar ]---------------------------------
[INFO]
[INFO] --- clean:3.2.0:clean (default-clean) @ kafka_lab ---
[INFO] Deleting C:\Users\ahmed\Downloads\LabsBigData\lab3_kafka\kafka_lab\target
[INFO]
[INFO] --- resources:3.3.1:resources (default-resources) @ kafka_lab ---
[INFO] Copying 0 resource from src\main\resources to target\classes
[INFO]
[INFO] --- compiler:3.11.0:compile (default-compile) @ kafka_lab ---
[INFO] Changes detected - recompiling the module! :source
[INFO] Compiling 2 source files with javac [debug target 1.8] to target\classes
[WARNING] bootstrap class path not set in conjunction with -source 8
[INFO]
[INFO] --- resources:3.3.1:testResources (default-testResources) @ kafka_lab ---
[INFO] skip non existing resourceDirectory C:\Users\ahmed\Downloads\LabsBigData\lab3_kafka\kafk
a_lab\src\test\resources
[INFO]
[INFO] --- compiler:3.11.0:testCompile (default-testCompile) @ kafka_lab ---
[INFO] Changes detected - recompiling the module! :dependency
[INFO]
[INFO] --- surefire:3.1.2:test (default-test) @ kafka_lab ---
[INFO]
[INFO] --- jar:3.3.0:jar (default-jar) @ kafka_lab ---
[INFO] Building jar: C:\Users\ahmed\Downloads\LabsBigData\lab3_kafka\kafka_lab\target\kafka_lab
-1.0-SNAPSHOT.jar
[INFO]
[INFO] --- assembly:3.6.0:single (make-assembly) @ kafka_lab ---
[INFO] Building jar: C:\Users\ahmed\Downloads\LabsBigData\lab3_kafka\kafka_lab\target\kafka_lab
-1.0-SNAPSHOT-jar-with-dependencies.jar
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  19.318 s
```

Après le lancement du code du producer, on copie le jar dans shared_volume en spécifiant le nom du topic et de cette façon, on obtient le résultat en temps réel sur le « consumer ». à chaque fois, on affiche la boucle de 1 jusqu'à 9 comme présenté dans le producer

```
root@hadoop-master:~# kafka-console-consumer.sh --bootstrap-server localhost:9092 --topic Hello-Kafka --from-beginning
hello
kafka
hadoop
hive
ahmed
qais
tp3
0
1
2
3
4
5
6
7
8
```

- ## Création du consumer

On apporte des modification au fichier « pom.xml » pour comporter le consumer et le producer à la fois.

De même, les fichiers jar sont créés.

```
igned partitions: Hello-Kafka-0
[main] INFO org.apache.kafka.clients.consumer.internals.ConsumerCoordinator - [Consumer clientId=consumer-test-1, groupId=test] Found no committ
ed offset for partition Hello-Kafka-0
[main] INFO org.apache.kafka.clients.consumer.internals.SubscriptionState - [Consumer clientId=consumer-test-1, groupId=test] Resetting offset f
or partition Hello-Kafka-0 to position FetchPosition{offset=29, offsetEpoch=Optional.empty, currentLeader=LeaderAndEpoch{leader=Optional[hadoop-
master:9092 (id: 0 rack: null)], epoch=0}}.
offset = 29, key = 0, value = 0
offset = 30, key = 1, value = 1
offset = 31, key = 2, value = 2
offset = 32, key = 3, value = 3
offset = 33, key = 4, value = 4
offset = 34, key = 5, value = 5
offset = 35, key = 6, value = 6
offset = 36, key = 7, value = 7
offset = 37, key = 8, value = 8
offset = 38, key = 9, value = 9
```

Ingestion des données d'une source vers une destination(sink) HDFS avec Kafka Connect

Après creation des fichiers de configuration des connecteur, on crée aussi le topic

```
root@hadoop-master:~# kafka-topics.sh --list --bootstrap-server localhost:9092
Hello-Kafka
Hello-Kafka-
__consumer_offsets
connect-topic
root@hadoop-master:~#
```

```
root@hadoop-master:~# more /tmp/test-sink.txt
Bonjour Kafka
Bienvenue dans le monde du streaming
root@hadoop-master:~#
```

```
root@hadoop-master:~# echo "Exercice Kafka Connect simple" >> /tmp/test-source.txt
root@hadoop-master:~# more /tmp/test-sink.txt
Bonjour Kafka
Bienvenue dans le monde du streaming
Exercice Kafka Connect simple
root@hadoop-master:~#
```

Application Word Count avec Kafka Streams

On crée le jar pour wordcount. Puis, on crée les topics input et output

```
root@hadoop-master:~# kafka-topics.sh --bootstrap-server localhost:9092 --create --topic input-topic --partitions 1 --replication-factor 1
Created topic input-topic.
root@hadoop-master:~# kafka-topics.sh --bootstrap-server localhost:9092 --create --topic output-topic --partitions 1 --replication-factor 1
Created topic output-topic.
```

```
i       2
am      2
ahmed   2
tp      1
word    2
count   2
wordcount       1
```

4. Mise en place d'un cluster Kafka à deux brokers et application WordCount interactive

Partie 1 : Configuration de plusieurs brokers

```
root@hadoop-master:/usr/local/kafka/config# $KAFKA_HOME/bin/kafka-topics.sh --bootstrap-server localhost:9092,localhost:9093,localhost:9094 --de
scribe --topic WordCount-Topic
Topic: WordCount-Topic  TopicId: feIYf2VjTHSU-1yAH2oSiQ PartitionCount: 3       ReplicationFactor: 2    Configs:
        Topic: WordCount-Topic  Partition: 0    Leader: 0       Replicas: 0,2   Isr: 0,2
        Topic: WordCount-Topic  Partition: 1    Leader: 2       Replicas: 2,1   Isr: 2,1
        Topic: WordCount-Topic  Partition: 2    Leader: 1       Replicas: 1,0   Isr: 1,0
root@hadoop-master:/usr/local/kafka/config#
```

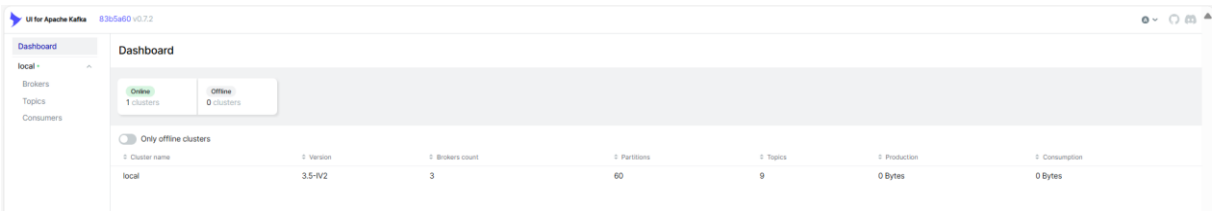
Partie 2 : Création de l'application word count

Dans le jar de « dependencies » se trouve le word producer et word consumer donnant le résultat suivant après exécution

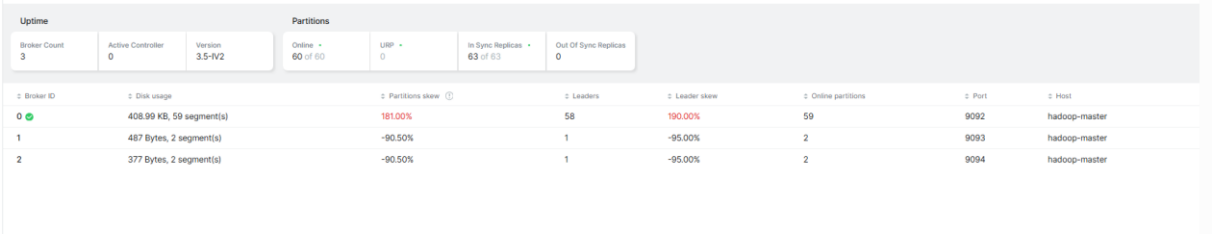
```
h{leader=Optional[hadoop-master:9094 (id: 2 rack: null)], epoch=0}}.
word=texte; count=1 partition=1 offset=0
word=sera count=1 partition=1 offset=1
word=au count=1 partition=1 offset=2
word=topicahmed count=1 partition=1 offset=3
word=# count=1 partition=2 offset=0
word=du count=1 partition=2 offset=1
word=tape count=1 partition=0 offset=0
word=chaque count=1 partition=0 offset=1
word=mot count=1 partition=0 offset=2
word=envoy?? count=1 partition=0 offset=3
word=wordcount count=1 partition=2 offset=2
word=hi count=1 partition=2 offset=3
word=lab count=1 partition=2 offset=4
word=tp count=1 partition=0 offset=4
word=hi count=2 partition=2 offset=5
word=ahmed count=1 partition=0 offset=5
```

Partie 3 : Kafka-ui

On crée le « docker-compose.kafka-ui.yml »,



Brokers



Uptime

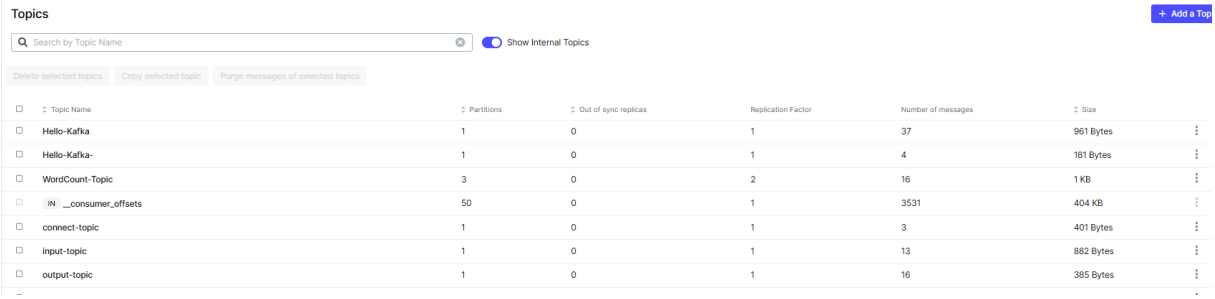
Broker Count	Active Controller	Version
3	0	3.5-IV2

Partitions

Online	URP	In Sync Replicas	Out Of Sync Replicas
60 of 60	0	63 of 63	0

Broker ID	Disk usage	Partitions skew	Leaders	Leader skew	Online partitions	Port	Host
0	408.99 KB, 59 segment(s)	181.00%	58	190.00%	59	9092	hadoop-master
1	487 Bytes, 2 segment(s)	-90.50%	1	-95.00%	2	9093	hadoop-master
2	377 Bytes, 2 segment(s)	-90.50%	1	-95.00%	2	9094	hadoop-master

Topics



+ Add a Top

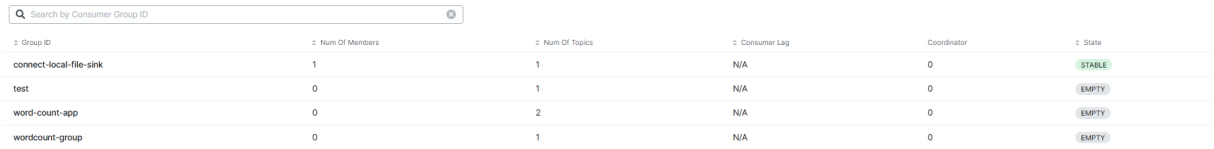
Search by Topic Name

Show Internal Topics

Delete selected topics | Copy selected topic | Purge messages of selected topics

Topic Name	Partitions	Out of sync replicas	Replication Factor	Number of messages	Size
Hello-Kafka	1	0	1	37	961 Bytes
Hello-Kafka-	1	0	1	4	181 Bytes
WordCount-Topic	3	0	2	16	1 KB
IN __consumer_offsets	50	0	1	3531	404 KB
connect-topic	1	0	1	3	401 Bytes
input-topic	1	0	1	13	882 Bytes
output-topic	1	0	1	16	385 Bytes

Consumers



Search by Consumer Group ID

Group ID	Num Of Members	Num Of Topics	Consumer Lag	Coordinator	State
connect-local-file-sink	1	1	N/A	0	STABLE
test	0	1	N/A	0	EMPTY
word-count-app	0	2	N/A	0	EMPTY
wordcount-group	0	1	N/A	0	EMPTY