**kafka常用命令：**

1. **查看有哪些主题：**

./kafka-topics.sh --list --zookeeper 127.0.0.1:2181

**1.查看topic的详细信息**

./kafka-topics.sh -zookeeper 127.0.0.1:2181 -describe -topic tx

**2、为topic增加副本**

./kafka-reassign-partitions.sh -zookeeper 127.0.0.1:2181 --reassignment-json-file replication.json -execute

{

"version": 1,

"partitions": [

{

"topic": "tx",

"partition": 0,

"replicas": [

1001,

1002,

1003

]

},

{

"topic": "tx",

"partition": 1,

"replicas": [

8,

2,

3

]

}

]

}

**3、创建topic**

./kafka-topics.sh --create --zookeeper localhost:2181 --replication-factor 1 --partitions 1 --topic tx

**4、为topic增加partition**

./bin/kafka-topics.sh –zookeeper 127.0.0.1:2181 –alter –partitions 20 –topic testKJ1

**5、kafka生产者客户端命令**

./kafka-console-producer.sh --broker-list 10.167.222.105:6667 --topic tx

**6、kafka消费者客户端命令**

./kafka-console-consumer.sh -zookeeper 10.167.222.106:2181 --from-beginning --topic tx

**7、kafka服务启动**

./kafka-server-start.sh ../conf/server.properties &

**8、下线broker**

./kafka-run-class.sh kafka.admin.ShutdownBroker --zookeeper 127.0.0.1:2181 --broker #brokerId# --num.retries 3 --retry.interval.ms 60

shutdown broker

**9、删除topic**

./kafka-run-class.sh kafka.admin.DeleteTopicCommand --topic testKJ1 --zookeeper 127.0.0.1:2181

./kafka-topics.sh --zookeeper localhost:2181 --delete --topic testKJ1

**10、查看consumer组内消费的offset**

./kafka-run-class.sh kafka.tools.ConsumerOffsetChecker --zookeeper localhost:2181 --group test-consumer-group --topic tx

修改偏移量(进入zookeeper设置)

set /consumers/test-consumer-group/offsets/tx/0 500

-------------------------------

=============================

---------------------------------

# Run on a YARN cluster

export HADOOP\_CONF\_DIR=XXX

./bin/spark-submit \

--class org.apache.spark.examples.SparkPi \

--master yarn-cluster \ # can also be `yarn-client` for client mode

--executor-memory 20G \

--num-executors 50 \

/path/to/examples.jar \

1000

# Run on a Spark Standalone cluster in cluster deploy mode with supervise

./bin/spark-submit \

--class org.apache.spark.examples.SparkPi \

**--master spark://207.184.161.138:7077** \

**--deploy-mode cluster**

--supervise

--executor-memory 20G \

--total-executor-cores 100 \

/path/to/examples.jar \

1000

删除kafka topic

su rm -r /var/kafka/log/tmp/test\*

/home/kafka/bin/kafka-topics.sh --delete --zookeeper localhost:2181 --topic tx

删除zookeeper相关的路径

打开zookeeper client

/home/ZooKeeper/bin/zkCli.sh

#删除topic test的consumer group，如果有消费记录的话

rmr /kafka/consumers/test-group

rmr /kafka/config/topics/test

rmr /kafka/brokers/topics/test

rmr /kafka/admin/delete\_topics/test

/home/kafka/bin/kafka-topics.sh --list --zookeeper localhost:2181

用下面命令可以查询到topic:DynamicRange broker:SparkMaster:9092的offset的最小值：

$ ./kafka-run-class.sh kafka.tools.GetOffsetShell --broker-list localhost:6667 --topic tx --time -2

输出

DynamicRange:0:1288

查询offset的最大值：

$ ./kafka-run-class.sh kafka.tools.GetOffsetShell --broker-list localhost:6667 --topic tx --time -1

输出

DynamicRange:0:7885

从上面的输出可以看出topic:DynamicRange只有一个partition:0 offset范围为:[1288,7885]

设置consumer group的offset

启动zookeeper client

$ /opt/cloudera/parcels/CDH/lib/zookeeper/bin/zkCli.sh

通过下面命令设置consumer group:DynamicRangeGroup topic:DynamicRange partition:0的offset为1288:

set /consumers/DynamicRangeGroup/offsets/DynamicRange/0 1288

注意如果你的kafka设置了zookeeper root，比如为/kafka，那么命令应该改为：

set /kafka/consumers/DynamicRangeGroup/offsets/DynamicRange/0 1288

生效

重启相关的应用程序，就可以从设置的offset开始读数据了。

kafka.Kafka /usr/hdp/2.6.0.3-8/kafka/config/server.properties