FutureCart CRM solution – Run book

Contents

[Dimension table load 1](#_Toc42472035)

[Case & Survey History Load 2](#_Toc42472036)

[Fact & Pivot table Load 2](#_Toc42472037)

[Real time data simulation in json format for Kafka 2](#_Toc42472038)

[Kafka Producer and Stream for Case event 2](#_Toc42472039)

[Kafka Producer and Stream for Survey event 3](#_Toc42472040)

[Loading incremental data from Cassandra to Hive data warehouse 4](#_Toc42472041)

[Data Validation 4](#_Toc42472042)

[Project name & location for different jars used in runbook 5](#_Toc42472043)

# Dimension table load

Below is the list of dimension tables present in MySQL and the process to populate it to futurecart\_dw database in hive:

|  |  |
| --- | --- |
| **MySQL DIM table name** | **Hive Dim table name** |
| futurecart\_calendar\_details | dim\_futurecart\_calendar |
| futurecart\_call\_center\_details | dim\_futurecart\_call\_center |
| futurecart\_case\_category\_details | dim\_futurecart\_case\_category |
| futurecart\_case\_country\_details | dim\_futurecart\_country |
| futurecart\_case\_priority\_details | dim\_futurecart\_case\_priority |
| futurecart\_employee\_details | dim\_futurecart\_employee |
| futurecart\_product\_details | dim\_futurecart\_product |
| futurecart\_survey\_question\_details | dim\_futurecart\_questions |

execute the below command to load the dimension table by passing two arguments i.e. MySQL table name and corresponding Hive table name:

|  |
| --- |
| spark2-submit --jars /mnt/bigdatapgp/edureka\_921625/project2/solution/jars/mysql-connector-java-5.1.48-bin.jar --driver-class-path /mnt/bigdatapgp/edureka\_921625/project2/solution/jars/mysql-connector-java-5.1.48-bin.jar --class com.edureka.futurecart.dim.RetailCartDimLoad --master yarn --deploy-mode client /mnt/bigdatapgp/edureka\_921625/project2/solution/jars/futurecartbatchload\_2.11-0.1.jar futurecart\_case\_priority\_details futurecart\_dw.dim\_futurecart\_case\_priority |

# Case & Survey History Load

Below is Sale and price change table names and commands to do the history load:

1. **futurecart\_case\_dly :**

execute the below script to load the history data for the table futurecart\_case\_dly from MySQL using Sqoop:

<https://github.com/tejaswichandra/FutureCart-CRM/blob/master/Sqoop_script_for_history_load/load_case_futurecart_dly_hist.sql>

1. **futurecart\_survey\_dly:**

execute the below script to load the history data for the table futurecart\_survey\_dly from MySQL using Sqoop:

<https://github.com/tejaswichandra/FutureCart-CRM/blob/master/Sqoop_script_for_history_load/load_survey_futurecart_dly_hist.sql>

# Fact & Pivot table Load

Below is Fact & Pivot table names and commands to populate it:

1. **fact\_futurecart\_case\_survey\_dly:**

|  |
| --- |
| spark2-submit --class com.edureka.futurecart.fact.FactCaseSurveyDaily --conf spark.sql.crossJoin.enabled=true --master yarn --deploy-mode client --driver-memory 10g /mnt/bigdatapgp/edureka\_921625/project2/solution/jars/futurecartbatchload\_2.11-0.1.jar |

1. **pivot\_fact\_futurecart\_case\_survey\_dly :**

|  |
| --- |
| spark2-submit --class com.edureka.futurecart.fact.PivotFactCaseSurveyDaily --conf spark.sql.crossJoin.enabled=true --master yarn --deploy-mode client --driver-memory 10g /mnt/bigdatapgp/edureka\_921625/project2/solution/jars/futurecartbatchload\_2.11-0.1.jar |

# Real time data simulation in json format for Kafka

To generate real time data for Kafka producer, a real time simulator script is created based on python which will generate json files in the corresponding directory of below two events every 15 seconds:

* futurecart\_case\_event
* futurecart\_survey\_event

Script location: <https://github.com/tejaswichandra/retailcart/tree/master/real_time_simulator>

Command to execute: python2 real\_time\_simulator.py

# Kafka Producer and Stream for Case event

Kafka producer for case event will fetch the latest json files from the case\_event directory and push the data to topic in json format in real time.

It will use the table: fc\_case\_producer\_audit in cassandra for fetching the latest created json file.

Below are the queries to be executed before starting kafka producer:

|  |
| --- |
| create table fc\_case\_producer\_audit(id int primary key, last\_fetched\_ts int);  INSERT INTO fc\_case\_producer\_audit (id, last\_fetched\_ts) VALUES (1, 0); |

Command to start Kafka producer:

|  |
| --- |
| java -cp futurecart-kafka-ingestion-2.3.0-jar-with-dependencies.jar com.edureka.futurecart.stream.driver.cases.CaseProducer fc\_case\_topic case 45000 |

In this command, fc\_case\_topic is the topic name, case is the event name and 45000 is the time interval in ms in which it will fetch the json data from the corresponding directory.

Kafka stream for case event will fetch the latest message from the kafka topic and will write it into the Cassandra table ‘fc\_case\_event’ in real time.

Below is the Cassandra table DDL which needs to be executed before starting the kafka stream:

<https://github.com/tejaswichandra/FutureCart-CRM/blob/master/Cassandra-DDL/cassandra_ddl.sql>

Command to start kafka stream:

|  |
| --- |
| java -cp futurecart-kafka-ingestion-2.3.0-jar-with-dependencies.jar com.edureka.futurecart.stream.driver.cases.CaseConsumer fc\_case\_topic |

In this command, fc\_case\_topic is the topic name and it will write the data to the table fc\_case\_event in Cassandra.

# Kafka Producer and Stream for Survey event

Kafka producer for survey event will fetch the latest json files from the survey\_event directory and push the data to topic in json format in real time.

It will use the table: fc\_survey\_producer\_audit in cassandra for fetching the latest created json file.

Below are the queries to be executed before starting kafka producer:

|  |
| --- |
| create table fc\_survey\_producer\_audit(id int primary key, last\_fetched\_ts int);  INSERT INTO fc\_survey\_producer\_audit (id, last\_fetched\_ts) VALUES (1, 0); |

Command to start Kafka producer:

|  |
| --- |
| java -cp futurecart-kafka-ingestion-2.3.0-jar-with-dependencies.jar com.edureka.futurecart.stream.driver.survey.SurveyProducer fc\_survey\_topic survey 45000 |

In this command, fc\_survey\_topic is the topic name, survey is the event name and 45000 is the time interval in ms in which it will fetch the json data from the corresponding directory.

Kafka stream for survey event will fetch the latest message from the kafka topic and will write it into the Cassandra table ‘fc\_survey\_event’ in real time.

Below is the Cassandra table DDL which needs to be executed before starting the kafka stream:

<https://github.com/tejaswichandra/FutureCart-CRM/blob/master/Cassandra-DDL/cassandra_ddl.sql>

Command to start kafka stream:

|  |
| --- |
| java -cp futurecart-kafka-ingestion-2.3.0-jar-with-dependencies.jar com.edureka.futurecart.stream.driver.survey.SurveyConsumer fc\_survey\_topic |

In this command, fc\_survey\_topic is the topic name and it will write the data to the table fc\_survey\_event in Cassandra.

# Loading incremental data from Cassandra to Hive data warehouse

To load the incremental data from Cassandra which is populated by kafka stream, below steps needs to be perfomed:

1. Execute the below commands to load the data from Cassandra to hive tables:

|  |
| --- |
| spark2-submit --class com.edureka.futurecart.FCCaseDaily --conf spark.cassandra.connection.host=cassandradb.edu.cloudlab.com --conf spark.cassandra.auth.username=edureka\_921625 --conf spark.cassandra.auth.password=edureka\_9216255ar56 --packages com.datastax.spark:spark-cassandra-connector\_2.11:2.4.3 --master yarn --deploy-mode client /mnt/bigdatapgp/edureka\_921625/project2/solution/jars/futurecartbatchload\_2.11-0.1.jar  spark2-submit --class com.edureka.futurecart.FCSurveyDaily –conf spark.cassandra.connection.host=cassandradb.edu.cloudlab.com --conf spark.cassandra.auth.username=edureka\_921625 --conf spark.cassandra.auth.password=edureka\_9216255ar56 --packages com.datastax.spark:spark-cassandra-connector\_2.11:2.4.3 --master yarn --deploy-mode client /mnt/bigdatapgp/edureka\_921625/project2/solution/jars/futurecartbatchload\_2.11-0.1.jar |

1. Re-execute the spark jobs to populate the fact & pivot table as mentioned in the step:

[fact\_pivot\_table\_load](#_Fact_&_Pivot)

# Data Validation

Below is the location for data validation queries and results after performing the history load:

# Project name & location for different jars used in runbook

**Jar file:** futurecartbatchload\_2.11-0.1.jar

**Project location:** <https://github.com/tejaswichandra/FutureCartCRM/tree/master/futurecartbatchload>

**Uses:** dim table load from MySQL to hive, cases & surveys data history load from MySQL, cases & survey data incremental data load from Cassandra, fact and pivot table load

**Jar file:** futurecart-kafka-ingestion-2.3.0-jar-with-dependencies.jar

**Project location:**

<https://github.com/tejaswichandra/FutureCart-CRM/tree/master/futurecart-realtime-ingestion>

**Uses:** kafka producer & stream for case and survey event