**Automating the Process with Oozie**

**Credit Card System**

**Req-2.2.3 Automating the Process with Oozie Functional Requirements 1) Create an Oozie Workflow that will automate the processes of steps 2.2.1 and 2.2.2.  Each of the files in step 2.2.1 should be deleted before the workflow is executed in order to prevent storage of redundant data  The tables created in step 2.2.2 should be dropped before executing the hive workflow in order to prevent redundancy.**

1. Create file by the name of creditcard.sql then put the following script inside the creditcard.sql file

drop table cdw\_branch;

create table cdw\_branch

( BRANCH\_CODE INT,BRANCH\_NAME VARCHAR(25), BRANCH\_STREET VARCHAR(30),

BRANCH\_CITY VARCHAR(30), BRANCH\_STATE VARCHAR(30), BRANCH\_ZIP INT,

BRANCH\_PHONE VARCHAR(13), LAST\_UPDATED TIMESTAMP )

row format delimited fields terminated by ','

STORED AS TEXTFILE

Tblproperties("skip.header.line.count"="3");

LOAD DATA INPATH '/user/maria\_dev/Credit\_card\_System/CDW\_SAPP\_BRANCH/part-m-00000'

OVERWRITE INTO TABLE cdw\_branch;

drop table CDW\_SAPP\_D\_BRANCH;

create external table CDW\_SAPP\_D\_BRANCH

(BRANCH\_CODE INT,BRANCH\_NAME VARCHAR(25), BRANCH\_STREET VARCHAR(30),

BRANCH\_CITY VARCHAR(30), BRANCH\_STATE VARCHAR(30), BRANCH\_ZIP INT,

BRANCH\_PHONE VARCHAR(13))

PARTITIONED BY (LAST\_UPDATED TIMESTAMP)

row format delimited fields terminated by ','

LINES TERMINATED BY '\n'

location '/user/maria\_dev/credit\_card\_system/CDW\_SAPP\_D\_BRANCH/';

SET hive.exec.dynamic.partition=true;

SET hive.exec.dynamic.partition.mode=nonstrict;

INSERT OVERWRITE TABLE CDW\_SAPP\_D\_BRANCH

PARTITION(LAST\_UPDATED)

SELECT BRANCH\_CODE,BRANCH\_NAME, BRANCH\_STREET, BRANCH\_CITY, BRANCH\_STATE,

COALESCE(BRANCH\_ZIP,99999),concat('(',substring(branch\_phone,1,3),')',

substring(branch\_phone,4,3),'-',substring(branch\_phone,7,4)),

last\_updated

FROM cdw\_branch;

drop table cdw\_CREDITCARD;

create table cdw\_creditcard

( TRANSACTION\_ID INT, DAY INT, MONTH INT, YEAR INT,

CREDIT\_CARD\_NO VARCHAR(16), CUST\_SSN INT,BRANCH\_CODE INT,

TRANSACTION\_TYPE VARCHAR(30),TRANSACTION\_VALUE DECIMAL(20,3))

row format delimited fields terminated by ','

STORED AS TEXTFILE

Tblproperties("skip.header.line.count"="3");

LOAD DATA INPATH '/user/maria\_dev/Credit\_card\_System/CDW\_SAPP\_CREDITCARD'

OVERWRITE INTO TABLE cdw\_creditcard;

drop table CDW\_SAPP\_F\_CREDIT\_CARD;

create external table CDW\_SAPP\_F\_CREDIT\_CARD

( TRANSACTION\_ID INT, CUST\_CC\_NO STRING,CUST\_SSN INT,

BRANCH\_CODE INT,TRANSACTION\_TYPE VARCHAR(30),

TRANSACTION\_VALUE DECIMAL(20,3), TIMEID VARCHAR(8))

PARTITIONED BY (YEAR VARCHAR(4))

row format delimited fields terminated by ','

LINES TERMINATED BY '\n'

location '/user/maria\_dev/credit\_card\_system/CDW\_SAPP\_F\_CREDIT\_CARD/';

SET hive.exec.dynamic.partition=true;

SET hive.exec.dynamic.partition.mode=nonstrict;

INSERT OVERWRITE TABLE CDW\_SAPP\_F\_CREDIT\_CARD

PARTITION (YEAR)

SELECT TRANSACTION\_ID, CREDIT\_CARD\_NO, CUST\_SSN,

BRANCH\_CODE ,TRANSACTION\_TYPE ,

TRANSACTION\_VALUE , concat(YEAR,lpad(MONTH,2,'0'),lpad(DAY,2,'0')) AS TIMEID , YEAR

FROM cdw\_creditcard;

DROP TABLE cdw\_CUSTOMER;

create table cdw\_customer

( CUST\_F\_NAME VARCHAR(40), CUST\_M\_NAME VARCHAR(40), CUST\_L\_NAME VARCHAR(40),

CUST\_SSN INT,CUST\_CC\_NO STRING,CUST\_APT\_NO STRING, CUSTO\_STREET VARCHAR(38),

CUST\_CITY VARCHAR(30), CUST\_STATE VARCHAR(30), CUST\_COUNTRY VARCHAR(30),

CUST\_ZIP INT, CUST\_PHONE VARCHAR(8), CUST\_EMAIL VARCHAR(40), LAST\_UPDATED TIMESTAMP )

row format delimited fields terminated by ','

STORED AS TEXTFILE

Tblproperties("skip.header.line.count"="3");

LOAD DATA INPATH '/user/maria\_dev/Credit\_card\_System/CDW\_SAPP\_CUSTOMER/part-m-00000'

OVERWRITE INTO TABLE cdw\_customer;

drop table CDW\_SAPP\_D\_CUSTOMER;

create external table CDW\_SAPP\_D\_CUSTOMER

( CUST\_F\_NAME VARCHAR(40), CUST\_M\_NAME VARCHAR(40), CUST\_L\_NAME VARCHAR(40),

CUST\_SSN INT,CUST\_CC\_NO STRING, CUSTO\_STREET VARCHAR(38), CUST\_CITY VARCHAR(30),

CUST\_COUNTRY VARCHAR(30), CUST\_ZIP INT, CUST\_PHONE VARCHAR(8),

CUST\_EMAIL VARCHAR(40), LAST\_UPDATED TIMESTAMP )

PARTITIONED BY (CUST\_STATE VARCHAR(30))

row format delimited fields terminated by ','

LINES TERMINATED BY '\n'

location '/user/maria\_dev/credit\_card\_system/CDW\_SAPP\_D\_CUSTOMER/';

SET hive.exec.dynamic.partition=true;

SET hive.exec.dynamic.partition.mode=nonstrict;

INSERT OVERWRITE TABLE CDW\_SAPP\_D\_CUSTOMER

PARTITION (cust\_state)

SELECT CONCAT(UCASE(SUBSTRING(cust\_f\_name, 1, 1)),LOWER(SUBSTRING(cust\_f\_name, 2))),

lcase(CUST\_M\_NAME),

CONCAT(UCASE(SUBSTRING(cust\_l\_name, 1, 1)),LOWER(SUBSTRING(cust\_l\_name, 2))),

cust\_cc\_no, concat(cust\_apt\_no," ",custo\_street) as cust\_street, cust\_city,

cust\_ssn, cust\_country,cust\_zip,

concat(SUBSTRING(cust\_phone, 1, 3),'-',SUBSTRING(cust\_phone, 4, 4)),

cust\_email, last\_updated, cust\_state

FROM CDW\_CUSTOMER;

drop table CDW\_SAPP\_D\_TIME;

create external table CDW\_SAPP\_D\_TIME

( TIMEID VARCHAR(8), DAY INT, MONTH INT,

YEAR INT,QUARTER VARCHAR(8))

row format delimited fields terminated by ','

LINES TERMINATED BY '\n'

location '/user/maria\_dev/credit\_card\_system/CDW\_SAPP\_D\_TIME/';

SET hive.exec.dynamic.partition=true;

SET hive.exec.dynamic.partition.mode=nonstrict;

INSERT OVERWRITE TABLE CDW\_SAPP\_D\_TIME

SELECT concat(YEAR,lpad(MONTH,2,'0'),lpad(DAY,2,'0')) AS TIMEID, DAY, MONTH,

YEAR, case month when 1 then 1 when 2 then 1 when 3 then 1 when 4 then 2

when 5 then 2 when 6 then 2 when 7 then 3 when 8 then 3 when 9 then 3

when 10 then 4 when 11 then 4 when 12 then 4

end as QUARTER

FROM cdw\_creditcard;

1. Create Ozzie workflow.xml as below

<?xml version="1.0" encoding="UTF-8"?>

<workflow-app xmlns="uri:oozie:workflow:0.2" name="Credit-card-System">

<start to="sqoop-node"/>

<action name="sqoop-node"> <!--node name-->

<sqoop xmlns="uri:oozie:sqoop-action:0.2">

<job-tracker>${jobTracker}</job-tracker> <!--job tracking node is http://sandbox.hortonworks.com:8050 which is define in job properties-->

<name-node>${nameNode}</name-node> <!--name node is http://sandbox.hortonworks.com:8020 which is defined in job properties-->

<prepare>

<delete path="${nameNode}/user/maria\_dev/Credit\_card\_System"/>

</prepare>

<archive>/user/oozie/share/lib/lib\_20161025075203/sqoop/java-json.jar#java-json.jar</archive>

<configuration>

<property>

<name>mapred.job.queue.name</name>

<value>${queueName}</value>

</property>

</configuration>

<command>import-all-tables --connect jdbc:mysql://localhost/CDW\_SAPP --driver com.mysql.jdbc.Driver --warehouse-dir /user/maria\_dev/Credit\_card\_System -m 1

</command> <!-- get data from mysql using sqoop and store data at hive database Credit\_card\_System -->

</sqoop>

<ok to="hive-node"/> <!-- if everything is ok then go to hive node -->

<error to="fail"/>

</action>

<action name="hive-node">

<hive xmlns="uri:oozie:hive-action:0.2">

<job-tracker>${jobTracker}</job-tracker>

<name-node>${nameNode}</name-node>

<prepare>

<delete path="${nameNode}/apps/hive/warehourse/CDW\*"/> <!--if old directory is present in hadoop then this line will delete old directory because when creditcard.sql file is run, this directry will be created again for new dataset -->

</prepare>

<configuration>

<property>

<name>mapred.job.queue.name</name>

<value>${queueName}</value>

</property>

</configuration>

<script>creditcard.sql</script>

<param>OUTPUT=/user/maria\_dev/creditcard\_system/</param> <!-- this output varibale is define inside the creditcard.sql file -->

</hive>

<ok to="end"/>

<error to="fail"/>

</action>

<kill name="fail">

<message>Sqoop failed, error message[${wf:errorMessage(wf:lastErrorNode())}]</message>

</kill>

<end name="end"/>

</workflow-app>

1. Create job.properties file

nameNode=hdfs://sandbox.hortonworks.com:8020

jobTracker=http://sandbox.hortonworks.com:8050

queueName=default

oozie.use.system.libpath=true

oozie.wf.application.path=${nameNode}/user/maria\_dev/credit\_card\_system/workflow.xml

1. Upload workflow.xml and creditcard.sql files under this path "user/maria\_dev/credit\_card\_system"
2. Upload job.properties file inside VMware local directory e.g Documents/job.properties

**Run Oozie Job**

1. Type: oozie job -oozie http://localhost:11000/oozie -config .Documents /job.properties -run
2. Type your sandbox ip on the browser : IP:11000/oozie (e.g 192.168.567.0:11000)/oozie

// You can oozie view and oozie job)

Reference:

https://oozie.apache.org/docs/3.1.3-incubating/DG\_CommandLineTool.html

<https://www.slideshare.net/martyhall/hadoop-tutorial-oozie>

2) Incorporate that workflow into an Oozie Coordinator that will execute with the following conditions:  Every weekday between 08:00 and 18:00 EST  Executes once every 20 minutes  Starts on April 2nd 2018 at 08:00 EST Ends on March 29th 2025 at 18:00 EST

<coordinator-app xmlns = "uri:oozie:coordinator:0.2" name =

"hcoord\_copydata\_from\_external\_orc" frequency = "0/20 8-18 ? \* 2-6" start ="2018-04-02T08:00Z" end = "2025-03-29T18:00Z" timezone = "America/New\_York" >

<controls>

<timeout>1</timeout>

<concurrency>1</concurrency>

<execution>FIFO</execution>

<throttle>1</throttle>

</controls>

<action>

<workflow>

<app-path>${nameNode}/user/maria\_dev/credit\_card\_system/workflow.xml</app-path>

</workflow>

</action>

</coordinator-app>