Version-Hadoop-2.7.2 Version-Hive-2.2.0 Version-Hbase-1.2.6

PICT, PUNE

SL-VI Assignment 4

4. Write an application using HBase and HiveQL for flight information system which will include

1. Creating, Dropping, and altering Database tables
2. Creating an external Hive table to connect to the HBase for Customer Information Table
3. Load table with data, insert new values and field in the table, Join tables with Hive
4. Create index on Flight information Table
5. Find the average departure delay per day in 2008.

# Creating, Dropping, and altering Database tables Using Hbase

### #Create Table:

hbase(main):002:0> create 'flight','finfo','fsch'

**#Table Created-list** hbase(main):003:0> list

**#Insert records in created table** hbase(main):004:0> put 'flight',1,'finfo:source','pune'

hbase(main):008:0> put 'flight',1,'finfo:dest','mumbai'

hbase(main):010:0> put 'flight',1,'fsch:at','10.25a.m.'

hbase(main):011:0> put 'flight',1,'fsch:dt','11.25 a.m.'

hbase(main):012:0> put 'flight',1,'fsch:delay','5min' hbase(main):015:0> put 'flight',2,'finfo:source','pune'

hbase(main):016:0> put 'flight',2,'finfo:dest','kolkata'

hbase(main):017:0> put 'flight',2,'fsch:at','7.00a.m.'

hbase(main):018:0> put 'flight',2,'fsch:dt','7.30a.m.'

hbase(main):019:0> put 'flight',2,'fsch:delay','2 min'

hbase(main):021:0> put 'flight',3,'finfo:source','mumbai'

hbase(main):022:0> put 'flight',3,'finfo:dest','pune'

hbase(main):023:0> put 'flight',3,'fsch:at','12.30p.m.'

hbase(main):024:0> put 'flight',3,'fsch:dt','12.45p.m.'

hbase(main):025:0> put 'flight',3,'fsch:delay','1 min'

hbase(main):026:0> put 'flight',4,'finfo:source','mumbai'

hbase(main):027:0> put 'flight',4,'finfo:dest','delhi'

hbase(main):028:0> put 'flight',4,'fsch:at','2.00p.m.'

hbase(main):029:0> put 'flight',4,'fsch:dt','2.45p.m.'

hbase(main):030:0> put 'flight',4,'fsch:delay','10 min'

### #Display Records from Table ‘flight’

hbase(main):031:0> scan 'flight'

**#Alter Table (add one more column family)** hbase(main):036:0> alter 'flight',NAME=>'revenue'

hbase(main):037:0> scan 'flight'

**#Insert records into added column family** hbase(main):038:0> put 'flight',4,'revenue:rs','45000' 0 row(s) in 0.0100 seconds

**#Check the updates** hbase(main):039:0> scan 'flight'

### #Delete Column family

hbase(main):040:0> alter 'flight',NAME=>'revenue',METHOD=>'delete'

**#changes Reflected in Table** hbase(main):041:0> scan 'flight'

**#Drop Table**

**#Create Table for dropping** hbase(main):046:0\* create 'tb1','cf'

hbase(main):047:0> list

**#Disable table** hbase(main):049:0> disable 'tb1'

hbase(main):050:0> drop 'tb1'

hbase(main):051:0> list

### #Read data from table for row key 1:

hbase(main):052:0> get 'flight',1

### Read data for particular column from HBase table:

hbase(main):053:0> get 'flight','1',COLUMN=>'finfo:source'

### Read data for multiple columns in HBase Table:

hbase(main):054:0> get 'flight','1',COLUMN=>['finfo:source','finfo:dest']

hbase(main):055:0> scan 'flight',COLUMNS=>'finfo:source'

# Creating an external Hive table to connect to the HBase for Customer Information Table

## Covers===>

1. **Load table with data, insert new values and field in the table, Join tables with Hive**

### Add these Jar files in Hive(on hive prompt)

add jar file:///usr/local/HBase/lib/zookeeper-3.4.6.jar; add jar file:///usr/local/HBase/lib/guava-12.0.1.jar;

add jar file:///usr/local/HBase/lib/hbase-client-1.2.6.jar; add jar file:///usr/local/HBase/lib/hbase-common-1.2.6.jar; add jar file:///usr/local/HBase/lib/hbase-protocol-1.2.6.jar; add jar file:///usr/local/HBase/lib/hbase-server-1.2.6.jar; add jar file:///usr/local/HBase/lib/hbase-shell-1.2.6.jar; add jar file:///usr/local/HBase/lib/hbase-thrift-1.2.6.jar;

add jar file:///usr/local/hive/lib/hive-hbase-handler-2.2.0.jar;

### Set the values of variables in Hive

set hbase.zookeeprt.quorum=localhost; set hive.metastore.client.setugi=true;

set hive.exec.stagingdir=/tmp/.hivestage; set hive.exec.dynamic.partition=true;

set hive.exec.dynamic.partition.mode=nonstrict; set hive.auto.convert.join=false;

set hive.hbase.wal.enabled=false;

SET hive.exec.dynamic.partition = true;

SET hive.exec.dynamic.partition.mode = nonstrict; SET hive.exec.max.dynamic.partitions = 10000;

SET hive.exec.max.dynamic.partitions.pernode = 1000;

### # Create the external table emp using hive

hive>create external table empdata2 ( ename string, esal int)

row format delimited fields terminated by "," stored as textfile location "/home/hduser/Desktop/empdata2";

hive>load data local inpath '/home/hduser/Desktop/empdb.txt' into table empdata2;

## #Create External Table in hive referring to hbase table

**# create hbase table emphive first** hbase(main):003:0> create 'emphive', 'cf'

### #create hive external table

CREATE external TABLE hive\_table\_emp(id int, name string, esal string) STORED BY 'org.apache.hadoop.hive.hbase.HBaseStorageHandler'

WITH SERDEPROPERTIES ("hbase.columns.mapping" = ":key,cf:name,cf:esal") TBLPROPERTIES ("hbase.table.name" = "emphive");

## # load data into hive\_table\_emp

**(Hive doesn’t allow directly inserting data into external hive table)**

### #for that create one hive table(managed table in hive)

**Managed table** and External **table in Hive**. There are two types of **tables in Hive** ,one is **Managed table** and second is external **table**. the difference is , when you drop a **table**, if it is **managed table hive** deletes both data and meta data,if it is external **table Hive** only deletes metadata.

hive>create table empdbnew(eno int, ename string, esal int) row format delimited fields terminated by ',' stored as textfile;

### #load data in managed table

hive>load data local inpath '/home/hduser/Desktop/empdbnew.txt' into table empdbnew;

### #Load data in external table from managed table.

hive>INSERT INTO hive\_table\_emp select \* from empdbnew;

hive> select \* from hive\_table\_emp;

**#display records where salary is greater than 40000** hive> select \* from hive\_table\_emp where esal>40000;

### #Check hbase for updates(The records are available in associated Hbase table)

hbase(main):008:0> scan 'emphive'

6 row(s) in 0.0700 seconds

## # Creating external table in Hive referring to Hbase #referring to flight table created in Hbase

CREATE external TABLE hbase\_flight\_new(fno int, fsource string,fdest string,fsh\_at string,fsh\_dt string,fsch\_delay string,delay int)

STORED BY 'org.apache.hadoop.hive.hbase.HBaseStorageHandler' WITH SERDEPROPERTIES ("hbase.columns.mapping" = ":key,finfo:source,finfo:dest,fsch:at,fsch:dt,fsch:delay,delay:dl") TBLPROPERTIES ("hbase.table.name" = "flight");

hive> CREATE external TABLE hbase\_flight\_new(fno int, fsource string,fdest string,fsh\_at string,fsh\_dt string,fsch\_delay string,delay int)

* STORED BY 'org.apache.hadoop.hive.hbase.HBaseStorageHandler'
* WITH SERDEPROPERTIES ("hbase.columns.mapping"

=":key,finfo:source,finfo:dest,fsch:at,fsch:dt,fsch:delay,delay:dl")

* TBLPROPERTIES ("hbase.table.name" = "flight");

**#table created in hive** hive> show tables; OK

abc ddl\_hive emp empdata empdata1 empdata2 empdbnew hbase\_flight

hbase\_flight1 hbase\_flight\_new hbase\_table\_1 hive\_table\_emp

Time taken: 0.036 seconds, Fetched: 12 row(s)

**# Display records from that table** hive> select \* from hbase\_flight\_new;

e) Find the average departure delay per day in 2008.

### #calculate average delay

hive> select sum(delay) from hbase\_flight\_new;

hive>

# d) Create index on Flight information Table

## #create index on hbase\_flight\_new

CREATE INDEX hbasefltnew\_index ON TABLE hbase\_flight\_new (delay)

AS 'org.apache.hadoop.hive.ql.index.compact.CompactIndexHandler' WITH DEFERRED REBUILD;

SHOW INDEX ON hbase\_flight\_new;

### #create index on table hbase\_flight\_new

hive> CREATE INDEX hbasefltnew\_index

* ON TABLE hbase\_flight\_new (delay)
* AS 'org.apache.hadoop.hive.ql.index.compact.CompactIndexHandler'
* WITH DEFERRED REBUILD;

**#show index on table hbase\_flight\_new** hive> SHOW INDEX ON hbase\_flight\_new;

## #join two tables in Hive

### #create table B for join

hive> create table empinfo(empno int, empgrade string) row format delimited fields terminated by ',' stored as textfile;

### #Load Data into table

hive> load data local inpath '/home/hduser/Desktop/empinfo.txt' into table empinfo; Loading data to table default.empinfo

### #insert data into the table

hive> load data local inpath '/home/hduser/Desktop/empinfo.txt' into table empinfo;

### # Table A empdbnew

hive> select \* from empdbnew;

### # Table B empinfo

hive> select \* from empinfo;

### #Join two tables(empdbnew with empinfo on empno)

hive> SELECT eno, ename, empno, empgrade FROM empdbnew JOIN empinfo ON eno = empno;

hive> SELECT eno, ename, empno, empgrade

* FROM empdbnew JOIN empinfo ON eno = empno;