Below details will give you the clear explanation of how to evolve the dynamic schema model in Hive.

* If we don’t know the schema upfront or particularly load some data into hive table.
* First of all, we have to check what are all the data available in dataset and try to identify the schema of the data and mentioning the column name and everything then I go and define the column name in create table statement by mentioning the column names. Once I done then I create the table and do the load activity.
* With the help dynamic schema, avoiding all the above steps because each and every time we are getting the data, table was created based on the data automatically without the manual involvement or human involvement.

**AVRO** is one of the serialized file format, which has partial schema in the data itself.

For eg. If we storing the data in avro format. If I check the avro data in the location, data is in binary format but however we can use set of programs called avro tools jar to read the avro schema data, we are able to see the header data in json format

**In the below use case, we can directly import the DB data using sqoop in avro format and create hive table without defining the columns by using the avro schema file (avsc) created in the below step, which helps hive create dynamic schema if we don’t know the schema upfront.**

**This is a very good feature in Hive.**

1. To achieve this dynamic schema evolution, First, I have to download the avro-tools jar from the site MVN REPOSITORY.

<https://mvnrepository.com/artifact/org.apache.avro/avro-tools/1.8.1>

1. Performing the sqoop import to get the data in AVRO format from MYSQL database

sqoop import -Dmapreduce.job.user.classpath.first=true

--connect jdbc:mysql://localhost/custpayments

--username root

--password root

-table customers

-m 3

--split-by customernumber

--target-dir /user/hduser/custavro

--delete-target-dir

--as-avrodatafile;

1. And then convert the avro data into avsc(avroschema) format with the help of avro tools jar.

hadoop jar avro-tools-1.8.1.jar getschema /user/hduser/custavro/part-m-00000.avro > /home/hduser/customer.avsc

1. If we see customer.avsc avro schema file , able to see the schema header details are stored in the json format.
2. keep this file in Hadoop location and point the location in create external table query

hadoop fs -put -f customer.avsc /tmp/customer.avsc

1. Using the avroschema file, I can create a table on top of that, as an external table. Why Because today I am getting 10 columns, tomorrow I am getting new column added or some columns removed, for that I have to extract again that new avro schema file. So, I will drop and recreate the table. This is run on daily basis.

create external table customeravro

stored as AVRO

location '/user/hduser/custavro'

TBLPROPERTIES('avro.schema.url'='hdfs:///tmp/customer.avsc');

1. This is the way we can evolve the schema, whenever it gets changed without knowing what changes happen.