在Hive中使用Avro

Avro（读音类似于[ævr]）是Hadoop的一个子项目，由Hadoop的创始人Doug Cutting牵头开发。Avro是一个数据序列化系统，设计用于支持大批量数据交换的应用。它的主要特点有：支持二进制序列化方式，可以便捷，快速地处理大量数据；动态语言友好，Avro提供的机制使动态语言可以方便地处理Avro数据。

在Hive中，我们可以将数据使用Avro格式存储，本文以avro-1.7.1.jar为例，进行说明。

如果需要在Hive中使用Avro，需要在$HIVE_HOME/lib目录下放入以下四个工具包：avro-1.7.1.jar、avro-tools-1.7.4.jar、 jackson-core-asl-1.8.8.jar、jackson-mapper-asl-1.8.8.jar。当然，你也可以把这几个包存在别的路径下面，但是你需要把这四个包放在CLASSPATH中。

为了解析Avro格式的数据，我们可以在Hive建表的时候用下面语句：

```
hive> CREATE EXTERNAL TABLE tweets
    > COMMENT "A table backed by Avro data with the
    >       Avro schema embedded in the CREATE TABLE statement"
    > ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.avro.AvroSerDe'
    > STORED AS
    > INPUTFORMAT  'org.apache.hadoop.hive.ql.io.avro.AvroContainerInputFormat'
    > OUTPUTFORMAT 'org.apache.hadoop.hive.ql.io.avro.AvroContainerOutputFormat'
    > LOCATION '/user/wyp/examples/input/'
    > TBLPROPERTIES (
    >   'avro.schema.literal'='{
    >       "type": "record",
    >       "name": "Tweet",
    >       "namespace": "com.miguno.avro",
    >       "fields": [
    >          { "name":"username",  "type":"string"},
    >          { "name":"tweet",     "type":"string"},
    >          { "name":"timestamp", "type":"long"}
    >       ]
    >   }'
    > );
OK
Time taken: 0.076 seconds

hive> describe tweets;
OK
username            string              from deserializer
tweet               string              from deserializer
timestamp           bigint              from deserializer
```

然后用Snappy压缩我们需要的数据，下面是压缩前我们的数据：

```
{
  "username": "miguno",
  "tweet": "Rock: Nerf paper, scissors is fine.",
  "timestamp": 1366150681
},
{
  "username": "BlizzardCS",
  "tweet": "Works as intended.  Terran is IMBA.",
  "timestamp": 1366154481
},
{
  "username": "DarkTemplar",
  "tweet": "From the shadows I come!",
  "timestamp": 1366154681
},
{
  "username": "VoidRay",
  "tweet": "Prismatic core online!",
  "timestamp": 1366160000
}
```

压缩完的数据假如存放在/home/wyp/twitter.avsc文件中，我们将这个数据复制到HDFS中的/user/wyp/examples/input/目录下：

```
hadoop fs -put /home/wyp/twitter.avro  /user/wyp/examples/input/
```

然后我们就可以在Hive中使用了：

```
hive> select * from tweets limit 5;;
OK
miguno Rock: Nerf paper, scissors is fine. 1366150681
BlizzardCS Works as intended.  Terran is IMBA. 1366154481
DarkTemplar From the shadows I come! 1366154681
VoidRay Prismatic core online! 1366160000
```

Time taken: 0.495 seconds, Fetched: 4 row(s)

当然，我们也可以将avro.schema.literal中的

```
{
  "type": "record",
  "name": "Tweet",
  "namespace": "com.miguno.avro",
  "fields": [
    {
      "name": "username",
      "type": "string"
    },
    {
      "name": "tweet",
      "type": "string"
    },
    {
      "name": "timestamp",
      "type": "long"
    }
  ]
}
```

存放在一个文件中，比如：twitter.avsc,然后上面的建表语句就可以修改为：

```
CREATE EXTERNAL TABLE tweets
   COMMENT "A table backed by Avro data with the Avro schema stored in HDFS"
   ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.avro.AvroSerDe'
   STORED AS
   INPUTFORMAT  'org.apache.hadoop.hive.ql.io.avro.AvroContainerInputFormat'
   OUTPUTFORMAT 'org.apache.hadoop.hive.ql.io.avro.AvroContainerOutputFormat'
   LOCATION '/user/wyp/examples/input/'
   TBLPROPERTIES (
      'avro.schema.url'='hdfs:///user/wyp/examples/schema/twitter.avsc'
   );
```

效果和上面的一样。