大数据库系统

6.7 Hive中数据导入与导出

6.7 Hive中数据导入与导出

- 主要内容

  6.7.1数据导入的几种方法介绍

  6.7.2数据导出的几种方法介绍

  6.7.3清空表中的数据

6.7.1 数据导入的几种方法介绍

- 1、使用load data命令加载本地文件到hive：拷贝文件

  写法： load data local inpath 'linux_filepath' [overwrite] into table tablename;

  ‘linux_filepath’指的是linux的文件路径

  例：load data local inpath '/opt/modules/hive-0.13.1-bin/student.txt' into table tmp2_table;

  一般用于日志文件的直接导入

  注意：如果表目录下有多个数据文件，全部都会被导入

- 覆盖表中的数据

  使用load data ，在into前加个overwrite为覆盖数据，不加overwrite为追加数据

  load data local inpath '/opt/modules/hive-0.13.1-bin/student.txt' overwrite into table tmp2_table;

6.7.1 数据导入的几种方法介绍

如果是向分区表导入数据，则必须指定导入的分区

```
load data local inpath '/opt/datas/emp.txt' into table emp_part partition
(date='20170211');
```

6.7.1 数据导入的几种方法介绍

- 2、加载HDFS文件到hive：移动文件

load data inpath 'hdfs_filepath' into table tablename;

hdfs_filepath表示hdfs上的文件路径

由于放到了hdfs中，因此适用于文件比较大的情况

同样，分区表要指定导入的分区

注意：如果表目录下有多个数据文件，全部都会被导入

6.7.1 数据导入的几种方法介绍

- 3、创建表时通过as select加载数据

create table tablename as select ……;

通过子查询方式直接把数据写入表中

常用于临时表反复使用，作为数据分析结果的保存

create table tmp2_table2 as select * from tmp2_table;

```
hive (tmp2)> create table tmp2_table2 as select * from tmp2_table;
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks is set to 0 since there's no reduce operator
Starting Job = job_1495579252923_0002, Tracking URL = http://bigdata-training01.hpsk.com:8088/proxy/ap
plication_1495579252923_0002/
Kill Command = /opt/modules/hadoop-2.5.0/bin/hadoop job  -kill job_1495579252923_0002
```

实际上是个Map任务，没有Reduce任务，没涉及到合并操作

6.7.1 数据导入的几种方法介绍

- 4、创建表时通过location加载数据

create [external] tablename (col_comment……) location ‘hdfs_filepath’;

直接指定加载location指定的路径上的数据文件，即把该路径作为表的目录，目录下的文件自动作为数据文件，不涉及移动拷贝文件

常用于固定位置的数据采集时指定hdfs的数据目录

tablename和目录的名称可以不一样

如果数据已经存在于HDFS上，并且这些数据已经有其他人在使用了，没办法改变数据文件格式和位置，所以这个时候建表，就要使用extend表，并且指定location，这样就可以直接从该位置读取数据

6.7.1 数据导入的几种方法介绍

- 5、创建表以后，通过insert加载数据将查询结果加载到表

insert overwrite| into table tablename select ……

into表示追加数据

如果将into替换为overwrite，则表示覆盖表

通常用于将其他表数据分析的结果存到另外的表中，也用于临时表

每次insert into都会创建一个新的文件，存储追加内容

例：查询tmp2_table，并将查询结果追加到tmp2_table4表

```
create table tmp2_table4(
num string ,
name string
)
row format delimited fields terminated by '\t'
stored as textfile;
insert into table tmp2_table4 select * from tmp2_table;
```

1、将代码输入进终端，可以看到这也是个map的任务，分为三个过程

```
hive (tmp2)> create table tmp2_table4(
           > num string ,
           > name string
           > )
           > row format delimited fields terminated by '\t'
           > stored as textfile;
OK
Time taken: 0.031 seconds
hive (tmp2)> insert into table tmp2_table4 select * from tmp2_table;
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks is set to 0 since there's no reduce operator
Starting Job = job_1495579252923_0003, Tracking URL = http://bigdata-training01.hpsk.com:8088/proxy/ap
plication_1495579252923_0003/
Kill Command = /opt/modules/hadoop-2.5.0/bin/hadoop job  -kill job_1495579252923_0003
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 0
2017-05-24 08:04:57,658 Stage-1 map = 0%,  reduce = 0%
2017-05-24 08:05:04,079 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 0.79 sec
MapReduce Total cumulative CPU time: 790 msec
Ended Job = job_1495579252923_0003
Stage-4 is selected by condition resolver.
Stage-3 is filtered out by condition resolver.
Stage-5 is filtered out by condition resolver.
Moving data to: hdfs://bigdata-training01.hpsk.com:8020/tmp/hive-hpsk/hive_2017-05-24_08-04-49_840_857
5864357310427580-1/-ext-10000
Loading data to table tmp2.tmp2_table4
Table tmp2.tmp2_table4 stats: [numFiles=1, numRows=4, totalSize=49, rawDataSize=45]
MapReduce Jobs Launched:
Job 0: Map: 1   Cumulative CPU: 0.79 sec   HDFS Read: 275 HDFS Write: 121 SUCCESS
Total MapReduce CPU Time Spent: 790 msec
OK
tmp2_table.number       tmp2_table.name
Time taken: 15.562 seconds
hive (tmp2)>
```

移动数据，加载数据到表中

2、查看表tmp2_table4的内容，可以看到跟tmp2_table一样

```
hive (tmp2)> select * from tmp2_table4;
OK
tmp2_table4.num tmp2_table4.name
1001    zhangsan
1002    lisi
1003    wangwu
1004    zhaoliu
Time taken: 0.03 seconds, Fetched: 4 row(s)
hive (tmp2)>
```

6.7.1 数据导入的几种方法介绍

- 如果想向分区表使用insert方式导入数据，同样需要指定分区

例：向student表的month=201707分区覆盖数据

insert overwrite table student partition(month='201707')

select id, name where month=‘201709’ from student;

- Insert还可以导入单条或者多条数据

insert into|overwrite table student_par values(1,'wangwu'),(2,'zhaoliu');

每添加一个一条数据，都会创建一个小文件在location指定的目录下

6.7.2 数据导出的几种方法介绍

- 1、通过insert命令进行导出

insert overwrite [local] directory 'path' select ……

将数据导出至本地目录（加local字段）：

insert overwrite local directory '/opt/datas/tmp2_table' select * from tmp2_table2;

导出到HDFS（不加local字段）：

insert overwrite directory '/tmp2_table' select * from tmp2_table2;

注意与数据导入命令insert into|overwrite table 区别开

例：1、将tmp2_table2表里的数据导出到/opt/datas/tmp2_table目录下

```
hive (tmp2)> insert overwrite local directory '/opt/datas/tmp2_table' select * from tmp2_table2;
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks is set to 0 since there's no reduce operator
Starting Job = job_1495579252923_0004, Tracking URL = http://bigdata-training01.hpsk.com:8088/proxy/ap
plication_1495579252923_0004/
Kill Command = /opt/modules/hadoop-2.5.0/bin/hadoop job  -kill job_1495579252923_0004
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 0
2017-05-24 08:10:58,788 Stage-1 map = 0%,  reduce = 0%
2017-05-24 08:11:05,247 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 0.76 sec
MapReduce Total cumulative CPU time: 760 msec
Ended Job = job_1495579252923_0004
Copying data to local directory /opt/datas/tmp2_table
Copying data to local directory /opt/datas/tmp2_table
MapReduce Jobs Launched:
Job 0: Map: 1   Cumulative CPU: 0.76 sec   HDFS Read: 273 HDFS Write: 49 SUCCESS
Total MapReduce CPU Time Spent: 760 msec
OK
tmp2_table2.number      tmp2_table2.name
Time taken: 15.663 seconds
hive (tmp2)>
```

依然是一个map任务

这里有复制数据到文件的过程

```
Last login: Wed May 24 07:51:09 2017 from 192.168.134.1
[hpsk@bigdata-training01 ~]$ cd /opt/datas/
[hpsk@bigdata-training01 datas]$ ll
total 20
-rw-rw-r-- 1 hpsk hpsk   79 May 21 22:58 dept.txt
-rw-rw-r-- 1 hpsk hpsk  656 May 21 22:58 emp.txt
-rw-rw-r-- 1 hpsk hpsk   30 May 23 15:46 hive.exec.log
-rw-rw-r-- 1 hpsk hpsk   42 May 23 15:48 hivetest.sql
drwxrwxr-x 2 hpsk hpsk 4096 May 24 08:11 tmp2_table
```

2、在Linux下进入/opt/datas目录下发现多了个tmp2_table文件夹

6.7.2 数据导出的几种方法介绍

3、进入/opt/datas/tmp2_table目录下，可以看到生成了一个数据文件

```
[hpsk@bigdata-training01 datas]$ cd tmp2_table/
[hpsk@bigdata-training01 tmp2_table]$ ll
total 4
-rw-r--r-- 1 hpsk hpsk 49 May 24 08:11 000000_0
```

4、Linux下使用more命令查看这个文件，就是表tmp2_table的内容

```
[hpsk@bigdata-training01 tmp2_table]$ more 000000_0
1001^Azhangsan
1002^Alisi
1003^Awangwu
1004^Azhaoliu
[hpsk@bigdata-training01 tmp2_table]$
```

这里我们发现，每行的数据之间分割符变成了奇怪的符号，可否由用户自己制定分隔符呢？

6.7.2 数据导出的几种方法介绍

- 指定分隔符导出数据

在insert指定路径之后加上row format delimited fields terminated by字段即可指定分隔符，例：

insert overwrite local directory ‘/opt/datas/tmp2_table’ row format delimited fields terminated by ‘\t’ select * from tmp2_table2;

执行结果：

```
[hpsk@bigdata-training01 datas]$ cd tmp2_table/
[hpsk@bigdata-training01 tmp2_table]$ ll
total 4
-rw-r--r-- 1 hpsk hpsk 49 May 24 08:12 000000_0
[hpsk@bigdata-training01 tmp2_table]$ more 000000_0
1001    zhangsan
1002    lisi
1003    wangwu
1004    zhaoliu
[hpsk@bigdata-training01 tmp2_table]$
```

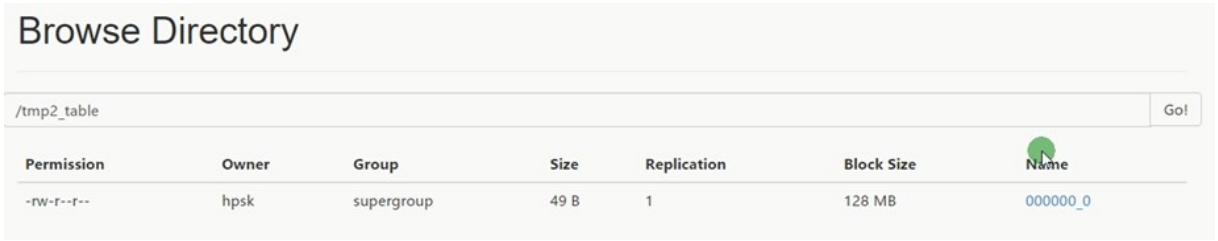
奇怪的方框符号消失了

6.7.2数据导出的几种方法介绍

- 2、将数据导出至HDFS目录，不加local字段

insert overwrite directory '/tmp2_table' select * from tmp2_table2;

执行后可以在/目录下找到‘tmp2_table目录，在/tmp2_table/目录下可以找到从tmp2_table2下导出的数据文件



在linux终端下查看hdfs中/tmp2_table/下的数据文件，又看到了奇怪的方框

```
[hpsk@bigdata-training01 hadoop-2.5.0]$ bin/hdfs dfs -text /tmp2_table/0*
1001□zhangsan
1002□lisi
1003□wangwu
1004□zhaoliu
[hpsk@bigdata-training01 hadoop-2.5.0]$
```

6.7.2数据导出的几种方法介绍

问：那将数据导出至HDFS目录是否可以支持用户自定分隔符呢?

做个测试，输入代码：

insert overwrite directory '/tmp2_table' row format delimited fields terminated by '\t'

select * from tmp2_table2;

```
        at org.apache.hadoop.hive.ql.parse.HiveParser.statement(HiveParser.java:1036)
        at org.apache.hadoop.hive.ql.parse.ParseDriver.parse(ParseDriver.java:199)
        at org.apache.hadoop.hive.ql.parse.ParseDriver.parse(ParseDriver.java:166)
        at org.apache.hadoop.hive.ql.Driver.compile(Driver.java:404)
        at org.apache.hadoop.hive.ql.Driver.compile(Driver.java:322)
        at org.apache.hadoop.hive.ql.Driver.compileInternal(Driver.java:975)
        at org.apache.hadoop.hive.ql.Driver.runInternal(Driver.java:1040)
        at org.apache.hadoop.hive.ql.Driver.run(Driver.java:911)
        at org.apache.hadoop.hive.ql.Driver.run(Driver.java:901)
        at org.apache.hadoop.hive.cli.CliDriver.processLocalCmd(CliDriver.java:268)
        at org.apache.hadoop.hive.cli.CliDriver.processCmd(CliDriver.java:220)
        at org.apache.hadoop.hive.cli.CliDriver.processLine(CliDriver.java:423)
        at org.apache.hadoop.hive.cli.CliDriver.executeDriver(CliDriver.java:792)
        at org.apache.hadoop.hive.cli.CliDriver.run(CliDriver.java:686)
        at org.apache.hadoop.hive.cli.CliDriver.main(CliDriver.java:625)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:606)
        at org.apache.hadoop.util.RunJar.main(RunJar.java:212)
FAILED: ParseException line 1:41 cannot recognize input near 'row' 'format' 'delimited' in select clau
se
```

出现了报错，得出结论：将数据导出至HDFS目录不支持用户自定分隔符

6.7.2 数据导出的几种方法介绍

- 3、通过Hadoop的hdfs命令中的get操作导出数据

本质上数据就是文件，直接使用get命令即可取得所需表的数据

例：将表tmp2_table的数据内容进行导出至目录/opt/datas中

```
[hpsk@bigdata-training01 hadoop-2.5.0]$ bin/hdfs dfs -get /tmp2_table/000000_0 /opt/datas/
```

进入/opt/datas目录下，查看数据文件

```
[hpsk@bigdata-training01 datas]$ ll
total 24
-rw-r--r-- 1 hpsk hpsk   49 May 24 08:16 000000_0
-rw-rw-r-- 1 hpsk hpsk   79 May 21 22:58 dept.txt
-rw-rw-r-- 1 hpsk hpsk  656 May 21 22:58 emp.txt
-rw-rw-r-- 1 hpsk hpsk   30 May 23 15:46 hive.exec.log
-rw-rw-r-- 1 hpsk hpsk   42 May 23 15:48 hivetest.sql
drwxrwxr-x 2 hpsk hpsk 4096 May 24 08:12 tmp2_table
[hpsk@bigdata-training01 datas]$ more 000000_0
1001zhangsan
1002lisi
1003wangwu
1004zhaoliu
[hpsk@bigdata-training01 datas]$
```

6.7.2 数据导出的几种方法介绍

- 4、通过hive -e 或者 -f 执行hive的语句，将数据执行的结果进行重定向保存

```
bin/hive -e 'select * from default.student;' >
    /opt/module/hive/data/export/student4.txt;
```

- 5、 import和export命令

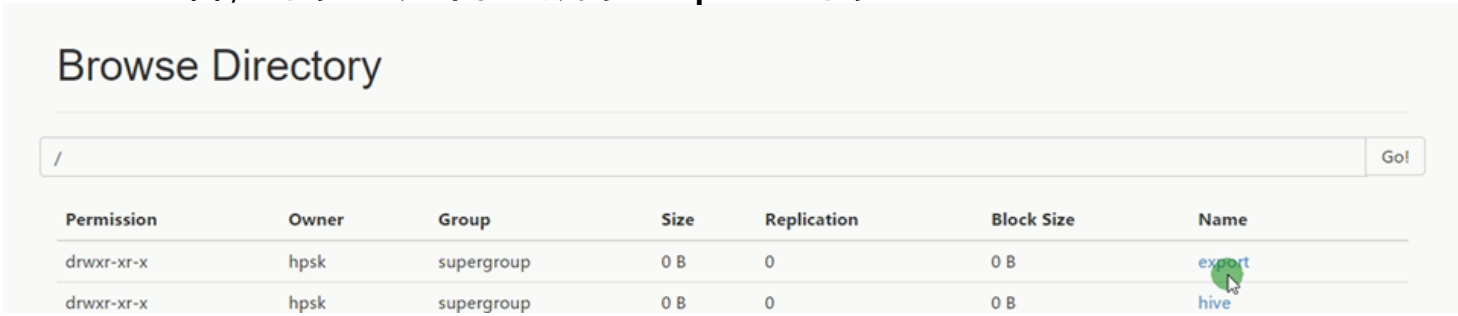
通常用于hive表的备份

```
export table tmp2_table to '/export‘;
import table tmp2_table5 from '/export';
```

例：1、将tmp2_table表导出至目录/export下(/export目录不存在)

```
hive (tmp2)> export table tmp2_table to '/export';
Copying data from file:/tmp/hpsk/hive_2017-05-24_08-19-34_340_4107071912136800599-1/-local-10000/_meta
data
Copying file: file:/tmp/hpsk/hive_2017-05-24_08-19-34_340_4107071912136800599-1/-local-10000/_metadata
Copying data from hdfs://bigdata-training01.hpsk.com:8020/hive/tmp2/tmp2_table
Copying file: hdfs://bigdata-training01.hpsk.com:8020/hive/tmp2/tmp2_table/student.txt
OK
Time taken: 0.108 seconds
hive (tmp2)>
```

2、查看/目录下是否生成了export目录



3、查看/export目录里的内容：元数据_metadata，数据data

4、查看/export/data目录里可以找到student.txt

6.7.2 数据导出的几种方法介绍

- 6、import导入（前面导入讲了5种方法）

首先，我们先创建一个与表tmp2_table（刚刚export导出的）相同的表结构的新表tmp2_table5

Create table tmp2_table5 like tmp2_table;

```
hive (tmp2)> create table tmp2_table5 like tmp2_table;
OK
Time taken: 0.06 seconds
hive (tmp2)> show tables;
OK
tab_name
tmp2_table
tmp2_table2
tmp2_table4
tmp2_table5
Time taken: 0.018 seconds, Fetched: 4 row(s)
hive (tmp2)>
```

6.7.2 数据导出的几种方法介绍

接着讲刚刚export导出的数据导入到tmp2_table5中

import table tmp2_table5 from '/export';

(from后面跟的是路径)

```
hive (tmp2)> import table tmp2_table5 from '/export';
Copying data from hdfs://bigdata-training01.hpsk.com:8020/export/data
Copying file: hdfs://bigdata-training01.hpsk.com:8020/export/data/student.txt
Loading data to table tmp2.tmp2_table5
OK
Time taken: 0.165 seconds
hive (tmp2)>
```

查看tmp2_table5的内容，现在有数据了

```
hive (tmp2)> select * from tmp2_table5;
OK
tmp2_table5.number      tmp2_table5.name
1001    zhangsan
1002    lisi
1003    wangwu
1004    zhaoliu
Time taken: 0.025 seconds, Fetched: 4 row(s)
hive (tmp2)>
```

6.7.3清空表中的数据

- 使用truncate清除表中数据

truncate tablename

例：清除student表中的数据

truncate table student;

注意： truncate 只能删除管理表中的数据，不能删除外部表中数据

总结

- 数据导入：

1、使用load data local inpath命令加载本地文件到hive：拷贝文件

2、使用load data inpath加载HDFS文件到hive：移动文件

3、创建表时通过as select加载数据

4、创建表时通过location加载数据

5、创建表以后，通过create ….insert into|overwrite加载数据将查询结果加载到表

6、import导入

总结

- 数据的导出：

  1、通过insert overwrite local directory……select…命令进行导出至本地目录

  row format delimited fields terminated by字段可指定分隔符

  2、通过insert overwrite directory……select…命令进行导出至HDFS目录，不支持指定分隔符

  3、通过Hadoop的hdfs命令中的get操作导出数据

  4、通过hive -e 或者 -f 执行hive的语句，将数据执行的结果进行重定向保存

  5、通过export命令导入导出

- 使用truncate清除表中数据