第十章-Apache Sqoop介绍及数据迁移

Objective(本课目标)

- ☑ 了解Apache Sqoop常用命令使用
- ☑ 掌握使用Sqoop完成从RDB到HDFS的数据迁移
- ☑ 掌握使用Sqoop完成从RDB到Hive的数据迁移

What is Sqoop?

- Sqoop is a tool designed to transfer data between Hadoop and relational databases or mainframes. (Sqoop是一个用于在Hadoop和关系数据库，或商业服务器之间的数据传输的工具)
- Import Data from RDBMS to HDFS (从RDB导入数据到HDFS)
- Export Data from HDFS to RDBMS (导出数据从HDFS到RDB)
    - Sqoop uses MapReduce to import and export the data, which provides parallel operation as well as fault tolerance. (Sqoop使用MapReduce导入和导出数据，提供并行操作和容错)
- Target Users (使用者)
    - application programmers (应用开发人员)
    - System administrators (系统管理员)
    - Database administrators (数据库管理员)
- sqoop底层就是mapreduce任务，但是只有map，没有reduce
- http://sqoop.apache.org/docs/1.4.6/SqoopUserGuide.html

Import RDB TABLE to HDFS(从RDB的表到HDFS)

```
# mysql数据库的表导出到HDFS
sqoop import --connect jdbc:mysql://localhost/sqoopdb --driver com.mysql.jdbc.Driver --table customers --username root --password hadoop --target-dir /sqoop/db/customers --m 3

# connect -> 连接数据库的url
# driver -> mysql的驱动包
# table -> 指定要导出的表
# target-dir -> HDFS的目标目录
# m -> mapper的个数
# --sqoop-import is an alias of sqoop import
```

Import by WHERE

```
# 导入数据通过where进行过滤
sqoop import --connect jdbc:mysql://localhost/sqoopdb --driver
com.mysql.jdbc.Driver --table orders --where "order_date > '2014-06-10'" --
username root --password hadoop --delete-target-dir --target-dir /sqoop/db/orders
--m 1

#--where 添加过滤条件
#--target-dir 删除HDFS目标目录
```

- ## Import by COLUMNs

```
#导入指定的column
sqoop import --connect jdbc:mysql://localhost/sqoopdb --driver
com.mysql.jdbc.Driver --table products --columns
"product_id,product_name,product_price" --username root --password hadoop --
delete-target-dir --target-dir /sqoop/db/products --m 1

#columns -> 指定column列表.
```

- ## Import by Query

```
# 将查询的结果集导入到HDFS
sqoop import --connect jdbc:mysql://localhost/sqoopdb --driver
com.mysql.jdbc.Driver --query "select * from categories where category_id > 50
and \$CONDITIONS" --username root --password hadoop --split-by category_id --
delete-target-dir --target-dir /sqoop/db/categories --m 1
```

- --query -> all the queries should end up with $CONDITIONS，which is used by sqoop internally to distribute record ranges to all mappers.(所有查询都应以$ CONDITIONS结束，sqoop在内部使用它来将记录范围分配给所有mapper).
- --split-by host -> the host column is used to split work units. (切分工作单元).
- $CONDITIONS：该变量记录了 split-by指定的column，通过该字段和mapper number 来分割数据.
- Note：查询结果为一个数据集，sqoop根据split-by值的范围把数据分为若干份(mapper数量决定)，然后把slipt-by的值给$CONDITIONS变量。比如说是ID，范围1~1500，那么每个mapper处理500条数据.

- ## Incremental Import with Sqoop (增量导入)

- Incremental Mode:
  - append – append all matched records (may create duplicates in target) 追加所有匹配的记录(可能在目标中创建重复记录)

    - last-modified – append new records and update modified records from all matched records (在源表中有数据更新的时候使用，检查列就必须是一个时间戳或日期类型的字段，更新完之后，last-value会被设置为执行增量导入时的当前系统时间)

```
sqoop import --connect jdbc:mysql://localhost/sqoopdb --driver com.mysql.jdbc.Driver --table departments --username root --password hadoop --incremental append --check-column department_id --last-value '3' --target-dir /sqoop/db/departments --m 1
```

File Format(文件格式)

- Target File Format:
    - --as-avrodatafile -> Imports data to Avro Data Files
    - --as-sequencefile -> Imports data to SequenceFiles
    - --as-textfile -> Imports data as plain text (default)
    - --as-parquetfile -> Imports data to Parquet Files

```
sqoop import --connect jdbc:mysql://localhost/sqoopdb --driver com.mysql.jdbc.Driver --table order_items --username root --password hadoop --delete-target-dir --target-dir /sqoop/db/order_items --m 1 --as-sequencefile
```

Mysql -> Hive

```
sqoop import --connect jdbc:mysql://localhost/sqoopdb --driver com.mysql.jdbc.Driver --table orders --username root --password hadoop --hive-import --create-hive-table --hive-table sqoopcase.orders --hive-overwrite --m 1

# --hive-import -> 导入数据到hive
# --create-hive-table -> 创建一张新的表
# --hive-overwrite -> 覆盖目标数据
```

Mysql -> Hive Partition

```
sqoop import --connect jdbc:mysql://localhost/sqoopdb --driver com.mysql.jdbc.Driver --query "select order_id, order_status from orders where order_date >= '2013-11-03' and order_date < '2013-11-04' and \$CONDITIONS" --username root --password hadoop --target-dir /sqoop/db/orders_partition --delete-target-dir --split-by order_status --hive-import --hive-table sqoopcase.orders --hive-partition-key "order_date" --hive-partition-value "20131103" --m 1

#导入hive分区数据,目标table会被动态的创建出来.
```

MySQL -> HBase数据迁移

```
#创建hbase的table
create 'products','data','category'

#sqoop import shell
sqoop import --connect jdbc:mysql://localhost/sqoopdb --username root --password hadoop --driver com.mysql.jdbc.Driver --table products --columns "product_id,product_name,product_description,product_price,product_image" --hbase-table products --column-family data --hbase-row-key product_id --m 1

count 'products'
scan 'products' , { LIMIT => 10 }

#插入category 数据
sqoop import --connect jdbc:mysql://localhost/sqoopdb --username root --password hadoop --driver com.mysql.jdbc.Driver --query "select p.product_id,c.category_name from products p inner join categories c on p.product_category_id = c.category_id and \$CONDITIONS" --split-by product_id --hbase-table products --column-family category --hbase-row-key product_id --m 1

count 'products'
scan 'products' , { LIMIT => 10 }
```

总结 (Summary)

- 掌握mysql到HDFS数据迁移
- 掌握mysql到Hive数据迁移
- 掌握mysql到Hbase数据迁移