2.7.1 Sqoop简介以及使用

2.7.1.1 产生背景

基于传统关系型数据库的稳定性，还是有很多企业将数据存储在关系型数据库中；早期由于工具的缺乏，Hadoop与传统数据库之间的数据传输非常困难。基于前两个方面的考虑，需要一个在传统关系型数据库和Hadoop之间进行数据传输的项目，Sqoop应运而生。

2.7.1.2 Sqoop是什么

Sqoop是一个用于Hadoop和结构化数据存储（如关系型数据库）之间进行高效传输大批量数据的工具。它包括以下两个方面：

可以使用Sqoop将数据从关系型数据库管理系统(如MySQL)导入到Hadoop系统(如HDFS、Hive、HBase)中

将数据从Hadoop系统中抽取并导出到关系型数据库(如MySQL)

Sqoop的核心设计思想是利用MapReduce加快数据传输速度。也就是说Sqoop的导入和导出功能是通过基于Map Task（只有map）的MapReduce作业实现的。所以它是一种批处理方式进行数据传输，难以实现实时的数据进行导入和导出。

官网介绍：

```
Apache Sqoop(TM) is a tool designed for efficiently transferring bulk
data between Apache Hadoop and structured datastores such as relational
databases.
```

2.7.1.3 特点

优点：它可以将跨平台的数据进行整合。

缺点：它不是很灵活。

```
mysql <--- > hdfs
mysql ---> hive
mysql ---> hbase
```

sqoop的重要的几个关键词？？

import ： 从关系型数据库到hadoop

export ： 从hadoop到关系型数据库。

2.7.2 Sqoop的安装

```
1、解压配置环境变量
tar -zxvf /home/sqoop... -C /usr/local/sqoop...
vi /etc/profile
2、mv ./conf/sqoop-env-template.sh ./conf/sqoop-env.sh

3、配置文件：vi ./conf/sqoop-env.sh
export HADOOP_COMMON_HOME=/usr/local/hadoop-2.7.1/
export HADOOP_MAPRED_HOME=/usr/local/hadoop-2.7.1/
export HIVE_HOME=/usr/local/hive-1.2.1/
export ZOOCFGDIR=/usr/local/zookeeper-3.4.7/

4、将mysql的驱动包导入到sqoop安装目录下的lib包下面
```

```
cp /home/mysql-connector-java-5.1.18.jar ./lib/
5、启动测试：
sqoop version
sqoop help
```

```
查看数据库：
sqoop list-databases \
--connect jdbc:mysql://hdp01:3306 \
--username root --password root;

sqoop list-tables \
--connect jdbc:mysql://hdp01:3306/test \
--username root --password mysql;
```

2.7.3 Sqoop-import

案例1：表没有主键，需要指定map task的个数为1个才能执行

```
bin/sqoop import --connect jdbc:mysql://hdp01:3306/userdb \
--username root --password mysql \
--table emp -m 1
```

2.7.4 DBMS-hdfs

案例2：表没有主键，使用--split-by指定执行split的字段

```
bin/sqoop import --connect jdbc:mysql://hdp01:3306/userdb \
--username root --password mysql \
--table emp \
--split-by id \
--target-dir hdfs://hdp01:9000/sqoopdata/3
```

```
出错--

Caused by: java.sql.SQLException: null,  message from server: "Host 'hdp03' is
not allowed to connect to this MySQL server"

解决方案：
先连接mysql：mysql -uroot -p
    #(执行下面的语句  .:所有库下的所有表    %：任何IP地址或主机都可以连接)
    GRANT ALL PRIVILEGES ON . TO 'root'@'%' IDENTIFIED BY 'mysql' WITH GRANT
OPTION;
    FLUSH PRIVILEGES;
    grant all privileges on . to root@"localhost" identified by "mysql" with
grant option;
    FLUSH PRIVILEGES;
```

案例3：需要导入的数据不是全部的，而是带条件导入

```
bin/sqoop import --connect jdbc:mysql://hdp01:3306/userdb \
--username root --password mysql \
--table emp \
--split-by id \
--where 'id > 1203' \
--target-dir hdfs://hdp01:9000/sqoopdata/5
```

案例4：要导入的数据，不想包含全部字段，只需要部分字段

```
bin/sqoop import --connect jdbc:mysql://hdp01:3306/userdb \
--username root --password mysql \
--split-by id \
--query 'select id,name,dept from emp where id < 1203 and $CONDITIONS' \
--target-dir hdfs://hdp01:9000/sqoopdata/7
```

2.7.5 DBMS-hive

案例5：将数据导入到hive中

```
bin/sqoop import --connect jdbc:mysql://hdp01:3306/userdb --username root --
password mysql --table emp --hive-import -m 1
```

2.7.6 增量导入数据

案例6：增量append方式导入数据：

```
bin/sqoop import --connect jdbc:mysql://hdp01:3306/userdb \
--username root --password mysql \
--table emp \
--incremental append \
--check-column id \
--last-value 1205 \
-m 1
```

2.7.7 Sqoop-export

案例7：数据导出：

```
bin/sqoop export \
--connect jdbc:mysql://hdp01:3306/userdb \
--username root \
--password mysql \
--table employee \
--export-dir hdfs://hdp01:9000/sqoopdata/5
```

mysql表的编码格式做为utf8，hdfs文件中的列数类型和mysql表中的字段数一样
导出暂不能由hive表导出mysql关系型数据库中
--export-dir是一个hdfs中的目录，它不识别_SUCCESS文件
--query导入的时候注意设置问题。