数据转换工具Sqoop

谭唐华

大数据协作框架

- “大数据协作框架”其实是一个统称，实际上就是Hadoop 2.x生态系统中几个辅助Hadoop 2.x框架。在此，主要是以下四个框架：
  - 数据转换工具Sqoop
  - 文件收集库框架Flume
  - 任务调度框架Oozie
  - 大数据WEB工具Hue

Sqoop分析数据的来源？主要有以下两种来源：

1）RDBMS：**数据大量存储在 RDBMS（Oracle，MYSQL，DB2 等等）上，如果需要对数据进行分析，需要将这些数据存储迁移到 HDFS 上去。**那么 **Sqoop 的作用**就是**将关系数据库中的某张表数据抽取到 Hadoop 的 HDFS 文件系统当中，底层运行的还是 MapReduce。**它利用 MapReduce 加快数据传输速度。批处理方式进行数据传输。也可以将 HDFS 上的文件数据或者是 Hive 表中的数据导出到关系型数据库当中的某张表中。

Sqoop 官方网址：http://sqoop.apache.org/

Sqoop分析数据的来源？主要有以下两种来源：

2)日志文件 数据存储在类似日志文件当中 如何收集这些数据到HDFS上呢？**Flume**

就是实时的收集数据，存储到HDFS中。

Flume官方网址：http://flume.apache.org/

Oozie的来由

- 当大数据分析平台中MapReduce Job和HiveQL比较多，需要定时调度，合理充分使用集群资源；此外，有很多业务，一般需要多个MapReduce 任务共同完成，那么job1、job2、job3之间的存在彼此的依赖调度。此时就需要一个调度框架来完成【多任务Job定时调度】和【多任务之间的依赖调度】，**在Hadoop 2.x生态系统中，有很多类似的框架，其中Oozie是功能最强大的，相对来说很多公司都使用的一个框架**（当然很多大公司，自身都有自己开发的调度系统，不会使用Oozie这些）。**Oozie既可以基于时间也可以基于数据可用性（调度任务运行之前首先判断要处理的数据是否在HDFS之上存在）的工作流调度框架。**当然还有很多其他开源的调度框架，比如Azkaban（简单，能实现调度，发预警，发邮件）、Zeus（阿里开源的Hadoop Job调度框架）等。

Hue的由来

前面已经讲解过很多框架了，各个框架都有自己的WEB UI监控页面，分别对应不同的端口号，比如HDFS（50070）、YARN（8088）、MapReduce（19888）以及Hive运行HiveQL语句时命令行方式等等，此时对于实际的开发人员和运维人员来说，需要一个统一的WEB UI页面，集成大多数大数据常用框架的监控和SQL运行界面，此时Hue应运而生，可以在浏览器端的Web控制台上与Hadoop集群进行交互来分析处理数据，例如操作HDFS上的数据，运行MapReduce Job等等。

- Hue官方网住：http://gethue.com/

大数据协作框架诞生

- 总上所述，在做大数据平台数据分析过程必须遇到的，因而诞生了对应的框架，并且是开源的，供各大公司使用。我们在学习时，首先要理解框架如何诞生的？能过解决什么问题？在进一步带着疑问去学习基本的使用，辅助我们大数据的分析，这样学习才更快更好。其中Sqoop和Oozie底层运行的也是MapReduce Job，所以MapReduce可以说是非常的核心关键，其优势就是分布式的并行计算所决定的。

课程大纲

- 基于CDH 5.x版本Hadoop 2.x和Hive环境搭建
- Sqoop功能及使用要点
- Sqoop如何导入数据到HDFS
- Sqoop如何导出数据到RDBMS表
- Sqoop与Hive的结合

选择CDH5.3.x版本框架

- 在Hadoop 2.x课程的高级部分，已经给大家介绍了目前世界上主流的三大HADOOP发型版本以及之间的关系，其中Cloudera公司发布的CDH 版本，为众多公司所使用，包括国内的京东、一号店、淘宝、百度等电商互联网大中小性公司。Cloudera公司发布的每一个CDH版本，其中一个最大的好处就是，帮我们**解决了大数据Hadoop 2.x生态系统中各个框架的版本兼容问题**，我们直接选择某一版本，比如CDH5.3.6版本，其中hadoop版本2.5.0，hive版本0.13.1，flume版本1.4.5；还有一点就是类似**Sqoop、Flume、Oozie等框架，在编译的时候都要依赖对应的Hadoop 2.x版本，使用CDH版本的时候，已经给我们编译好了，无需再重新配置编译**。、
- CDH 5.x版本下载地址：

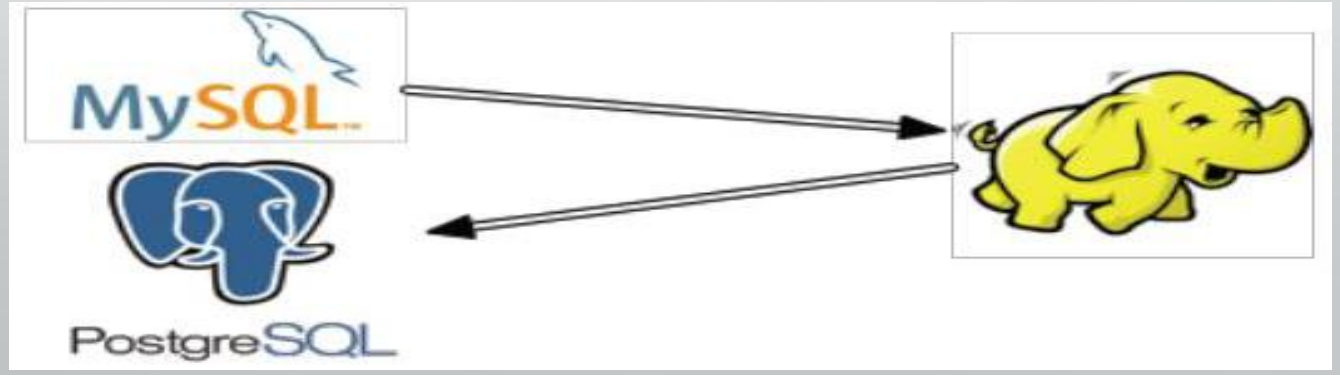
http://archive.cloudera.com/cdh5/cdh/5/

Apache Sqoop

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

Apache Sqoop

- Sqoop：SQL-to-Hadoop
- 连接传统关系型数据库和Hadoop的桥梁
  - 把关系型数据库的数据导入到Hadoop与其相关的系统(HBase和Hive)中
  - 把数据从Hadoop系统里抽取并导出到关系型数据库里
- 利用MapReduce加快数据传输速度，批处理方式进行数据传输



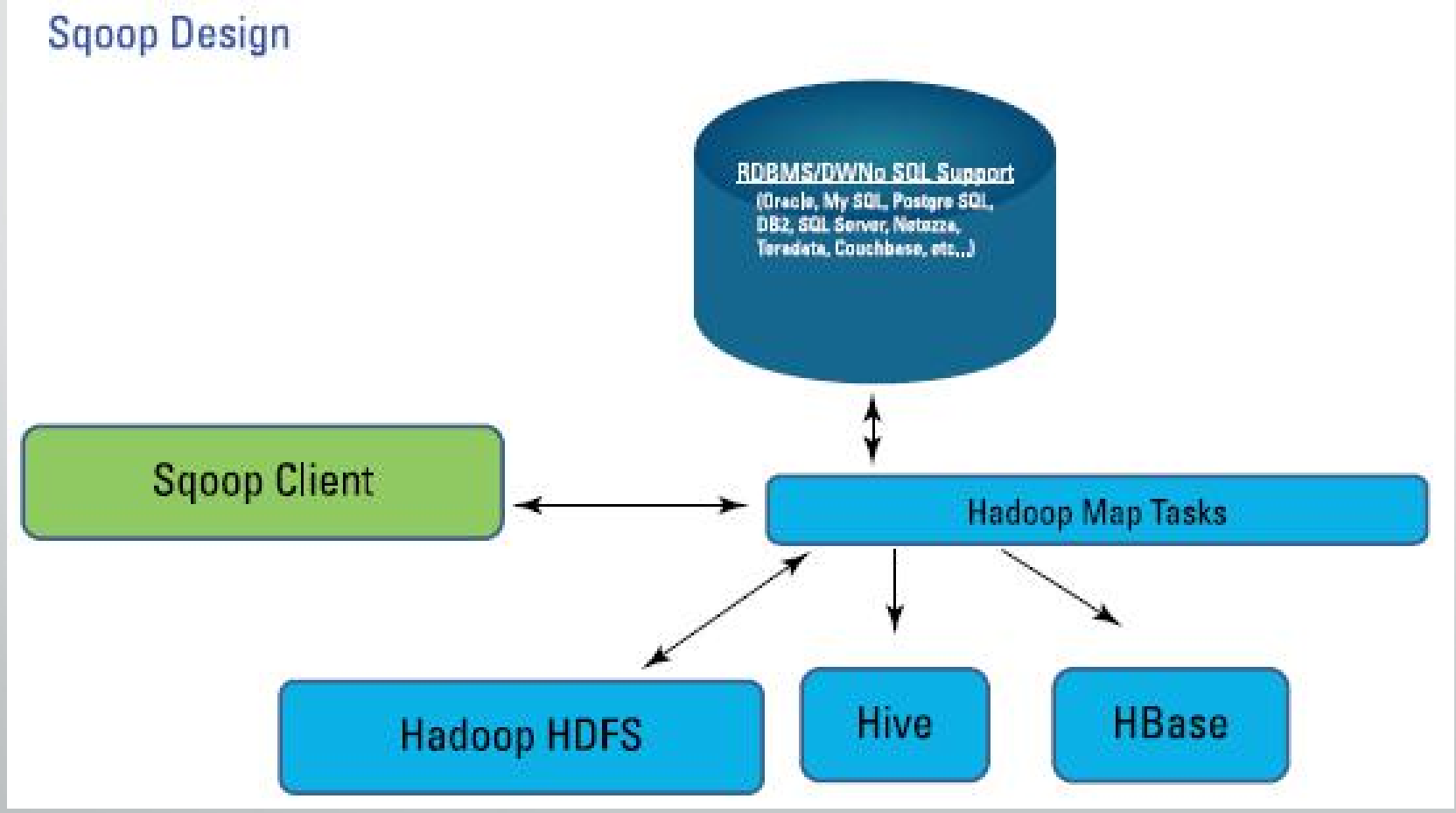
Sqoop1 & Sqoop2

- 两个不同版本，完全不兼容
- 版本号划分方式
  - Apache：1.4.x~ ，1.99.x~
- Sqoop2比Sqoop1的改进
  - 引入sqoop server，集中化管理Connector等
  - 多种访问方式：CLI，Web UI，REST API
  - 引入基于角色的安全机制

课程大纲

- 基于CDH 5.x版本Hadoop 2.x和Hive环境搭建
- Sqoop功能及使用要点
- Sqoop如何导入数据到HDFS
- Sqoop如何导出数据到RDBMS表
- Sqoop与Hive的结合

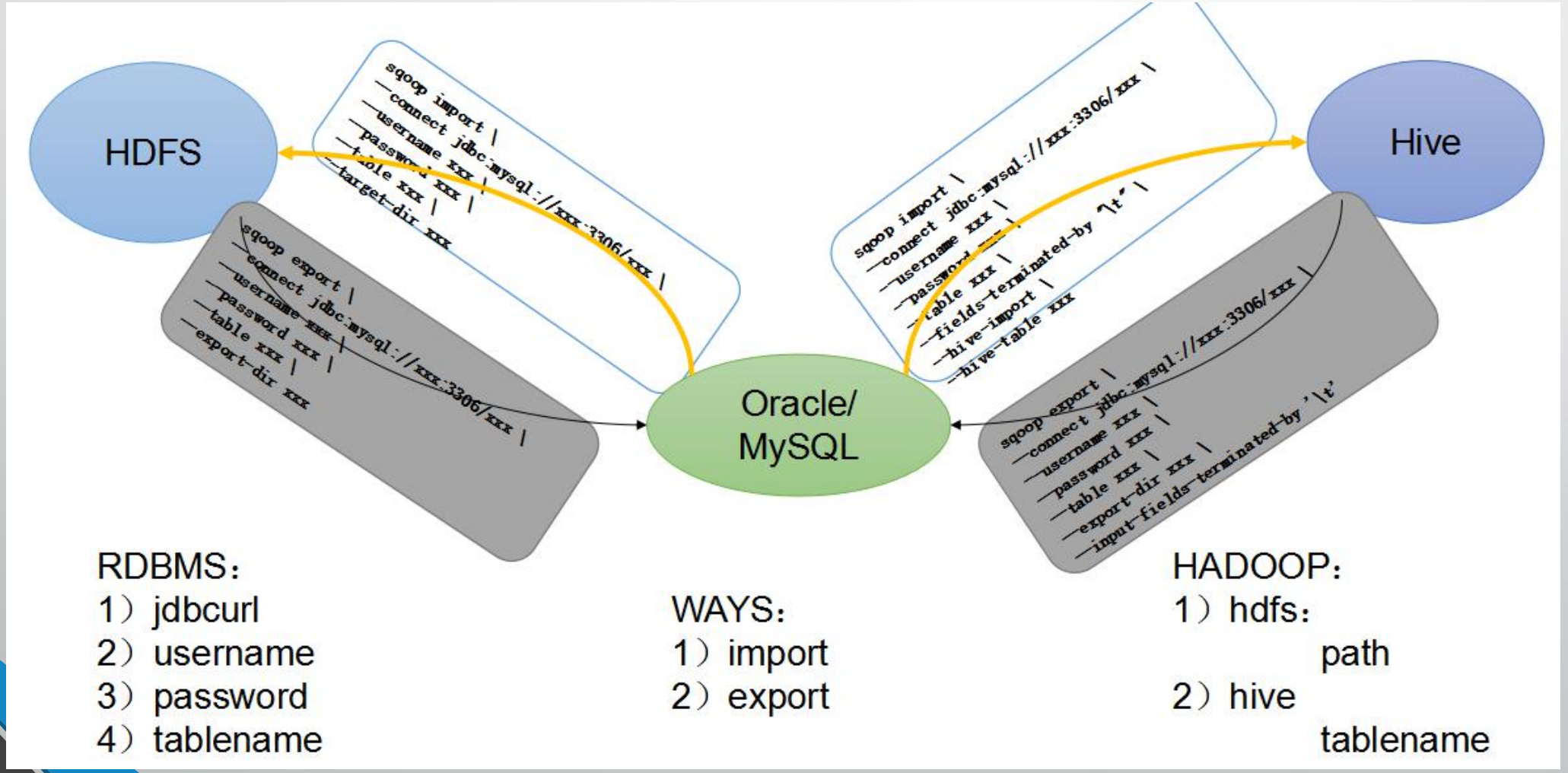
Apache Sqoop



Sqoop1 架构

- You can use Sqoop to import data from a relational database management system (RDBMS) such as MySQL or Oracle into the Hadoop Distributed File System (HDFS), transform the data in Hadoop MapReduce, and then export the data back into an RDBMS.
- Sqoop automates most of this process, relying on the database to describe the schema for the data to be imported. Sqoop uses MapReduce to import and export the data, which provides **parallel operation as well as fault tolerance**.

Sqoop 使用要点



RDBMS：
1）jdbcurl
2）username
3）password
4）tablename

WAYS：
1）import
2）export

HADOOP：
1）hdfs：
path
2）hive
tablename

Sqoop Installation

- 下载解压Sqoop 1.x
  - 下载地址：http://archive.cloudera.com/cdh5/cdh/5/sqoop-1.4.5-cdh5.3.6/
  - 解压

```
$ tar -zxf sqoop-1.4.5-cdh5.3.6.tar.gz -C /opt/cdh5.3.6
```

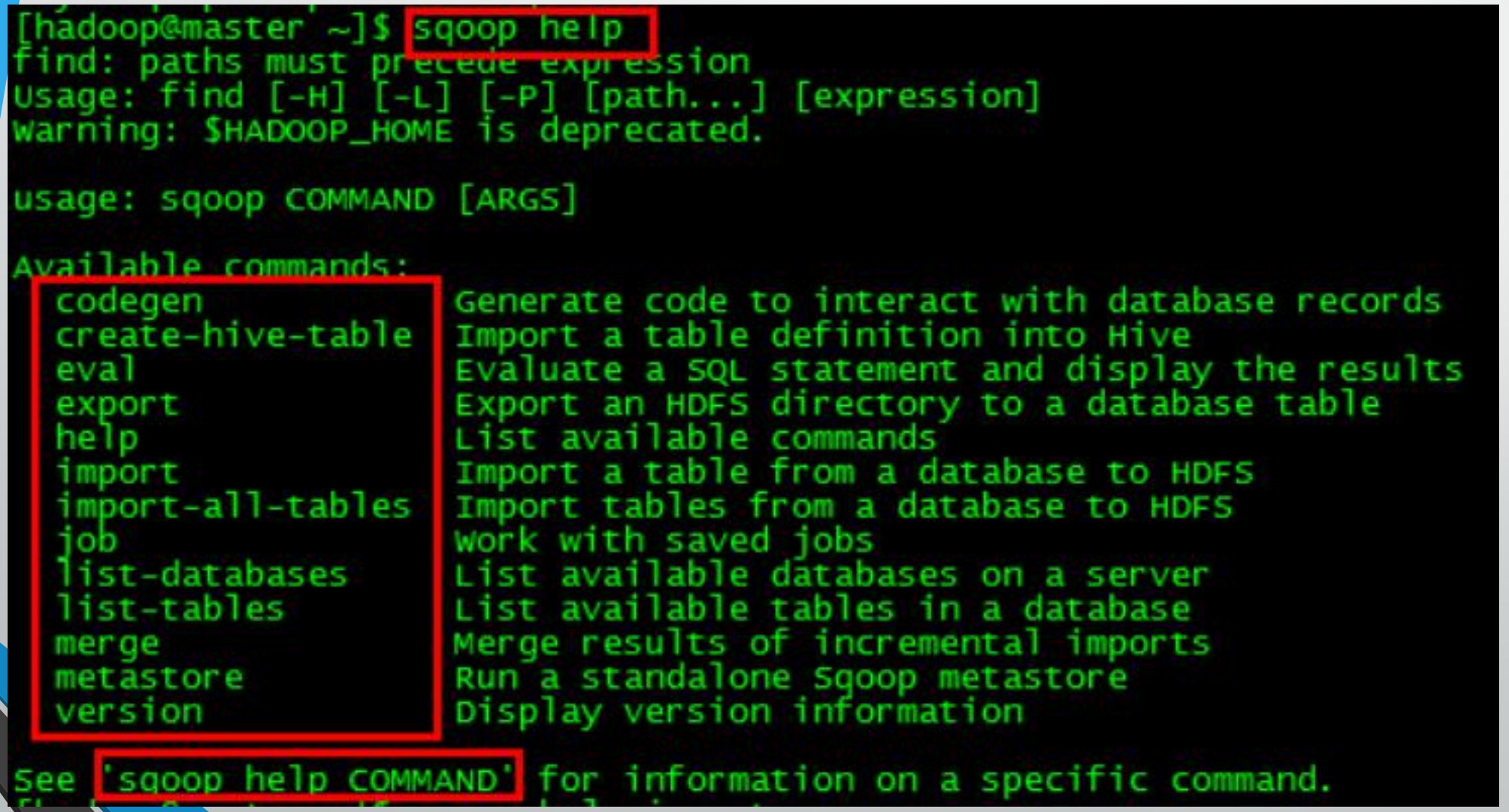
- 配置Sqoop 1.x

Sqoop安装目录下的conf目录，重命名sqoop-env-template.sh为sqoop-env.sh，配置环境变量：

```
#Set path to where bin/hadoop is available
export HADOOP_COMMON_HOME=/opt/cdh5.3.6/hadoop-2.5.0-cdh5.3.6

#Set path to where hadoop-*-core.jar is available
export HADOOP_MAPRED_HOME=/opt/cdh5.3.6/hadoop-2.5.0-cdh5.3.6
#Set the path to where bin/hive is available
export HIVE_HOME=/opt/cdh5.3.6/hive-0.13.1-cdh5.3.6
```

SQOOP HELP


[hadoop@master ~]$ sqoop help
find: paths must precede expression
Usage: find [-H] [-L] [-P] [path...] [expression]
Warning: $HADOOP_HOME is deprecated.
usage: sqoop COMMAND [ARGS]
Available commands:
codegen Generate code to interact with database records
create-hive-table Import a table definition into Hive
eval Evaluate a SQL statement and display the results
export Export an HDFS directory to a database table
help List available commands
import Import a table from a database to HDFS
import-all-tables Import tables from a database to HDFS
job Work with saved jobs
list-databases List available databases on a server
list-tables List available tables in a database
merge Merge results of incremental imports
metastore Run a standalone Sqoop metastore
version Display version information
See 'sqoop help COMMAND' for information on a specific command.

测试Sqoop 1.x

使用Sqoop 连接MySQL数据库，显示有哪些DataBase进行测试。

第一步、放置驱动包

将 MySQL 数据库 JDBC 驱动包复制到 sqoop 中的 Lib 目录下

```
cd /opt/modules/mysql-connector-java-5.1.27/
cp mysql-connector-java-5.1.27-bin.jar /opt/cdh-5.3.6/sqoop-1.4.5-cdh5.3.6/lib
```

第二步、Sqoop 测试

链接 mysql 数据库，并 list 数据库中的 databases。

```
bin/sqoop list-databases \
--connect jdbc:mysql://hadoop-senior.ibeifeng.com:3306 \
--username root \
--password 123
```

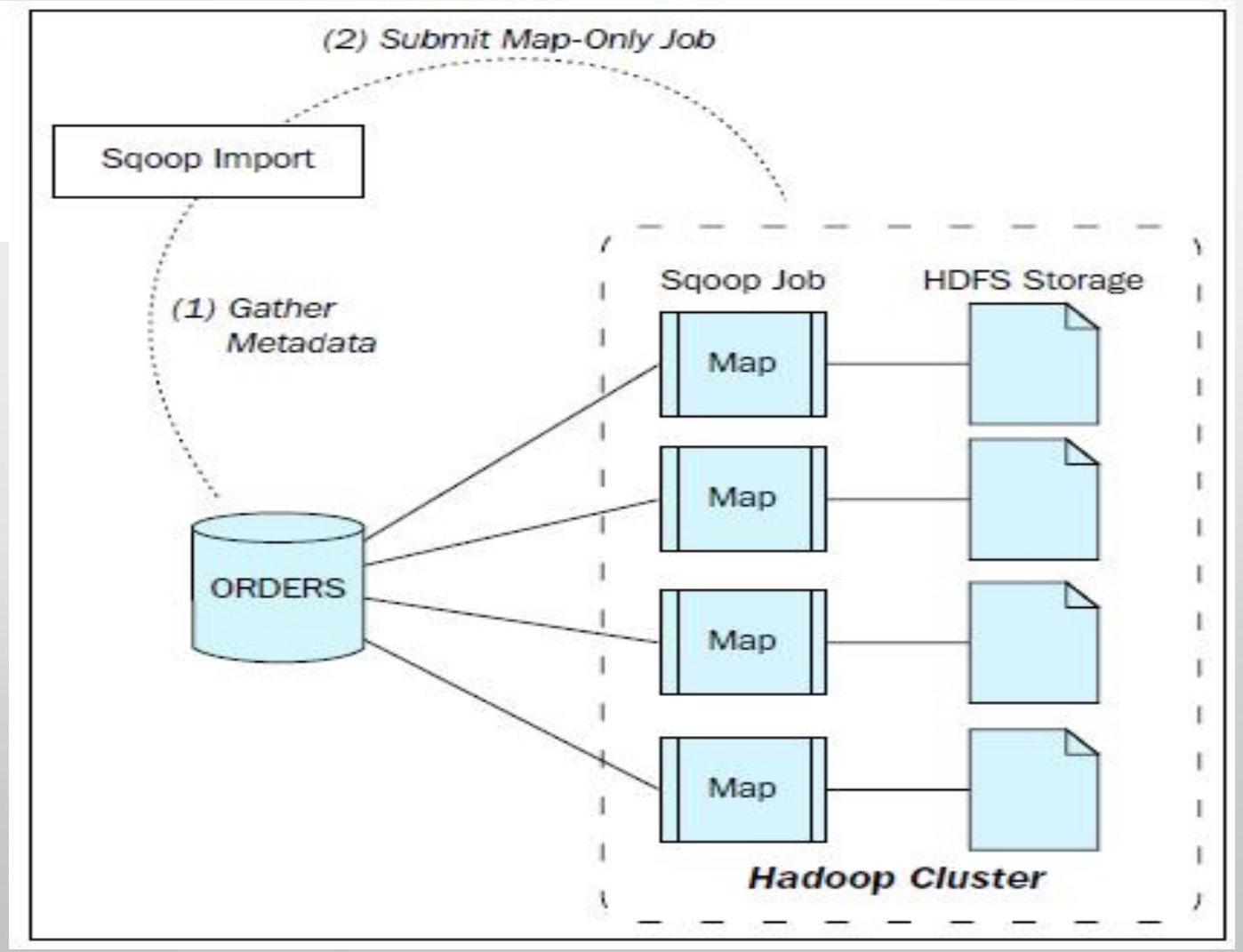
课程大纲

- 基于CDH 5.x版本Hadoop 2.x和Hive环境搭建
- Sqoop功能及使用要点
- Sqoop如何导入数据到HDFS
- Sqoop如何导出数据到RDBMS表
- Sqoop与Hive的结合

导入数据HDFS

Sqoop import is executed in two steps:

1. Gather metadata
2. Submit map only job



准备测试数据

- 以MySQL数据库为例，按照如下步骤准备数据：
  - **第一步**、进入mysql数据库

```
sudo mysql -uroot -p123
```

  - 第二步、创建数据库与表

```
create database cc_1209;
use cc_1209;
CREATE TABLE `my_user` (
  `id` tinyint(4) NOT NULL AUTO_INCREMENT,
  `account` varchar(255) DEFAULT NULL,
  `passwd` varchar(255) DEFAULT NULL,
  PRIMARY KEY (`id`)
);
```

准备测试数据

- 以MySQL数据库为例，按照如下步骤准备数据：
  - **第三步**、导入数据到表中

```
1.  INSERT INTO `my_user` VALUES ('1', 'admin', 'admin');
2.  INSERT INTO `my_user` VALUES ('2', 'pu', '12345');
3.  INSERT INTO `my_user` VALUES ('3', 'system', 'system');
4.  INSERT INTO `my_user` VALUES ('4', 'zxh', 'zxh');
5.  INSERT INTO `my_user` VALUES ('5', 'test', 'test');
6.  INSERT INTO `my_user` VALUES ('6', 'pudong', 'pudong');
7.  INSERT INTO `my_user` VALUES ('7', 'qiqi', 'qiqi');
```

  - **第四步**、显示表中数据

```
mysql> show tables;
+--------------------+
| Tables_in_cc_1209 |
+--------------------+
| my_user            |
+--------------------+
1 row in set (0.00 sec)
```

```
mysql> select * from my_user;
+----+---------+--------+
| id | account | passwd |
+----+---------+--------+
|  1 | admin   | admin  |
|  2 | pu      | 12345  |
|  3 | system  | system |
|  4 | zxh     | zxh    |
|  5 | test    | test   |
|  6 | pudong  | pudong |
|  7 | qiqi    | qiqi   |
+----+---------+--------+
7 rows in set (0.00 sec)
```

默认情况下导入数据至HDFS

- 命令：
  - bin/sqoop import \
  - --connect jdbc:mysql://192.168.10.1:3306/cc_123 \
  - --username root \
  - --password 123456 \
  - --table my_user
- 结果：

指定目录和Mapper个数

- 创建目录
  - bin/hdfs dfs -mkdir sqoop/input
- 设置map个数为1，指定目录为/user/beifeng/sqoop/input/,并且如果已存在目标目录则先删除
  - bin/sqoop import \
  - --connect jdbc:mysql://192.168.10.1:3306 /cc_123 \
  - --username root \
  - --password 123456 \
  - --table my_user \
  - --num-mappers 1 \
  - --target-dir /user/hadoop/sqoop/input \
  - --delete-target-dir

定义字段用制表符隔开

- 默认的情况下，导入到HDFS上的文件中每行数据的列与列之间的分隔符是【逗号】隔开，可以通过【--fields-terminated-by】属性指定分隔符，测试命令如下：
  - bin/sqoop import \
  - --connect jdbc:mysql://192.168.10.1:3306 /cc_123 \
  - --username root \
  - --password 123456 \
  - --table my_user \
  - --num-mappers 1 \
  - --target-dir /user/hadoop/sqoop/input \
  - --delete-target-dir \
  - --fields-terminated-by “\t”

增量导入

- 增量导入数据到HDFS文件中，可以通过下面三个参数进行设置：
  - --check-column
  - --incremental
  - --last-value
- 命令如下：
  - bin/sqoop import \
  - --connect jdbc:mysql://192.168.10.1:3306 /cc_123 \
  - --username root \
  - --password 123456 \
  - --table my_user \
  - --num-mappers 1 \
  - --target-dir /user/hadoop/sqoop/input \
  - --fields-terminated-by "\t"
  - --check-column id \
  - --incremental append \
  - --last-value 4 //表示从第5位开始导入

指定文件格式

- 默认情况下，导入数据到HDFS文件存储格式为textfile，可以通过属性进行指定，比如文件存储格式为parquet，命令如下：
  - bin/sqoop import \
  - --connect jdbc:mysql://192.168.10.1:3306 /cc_123 \
  - --username root \
  - --password 123456 \
  - --table my_user \
  - --num-mappers 1 \
  - --target-dir /user/hadoop/sqoop/input \
  - --fields-terminated-by “\t”
  - --as-parquetfile

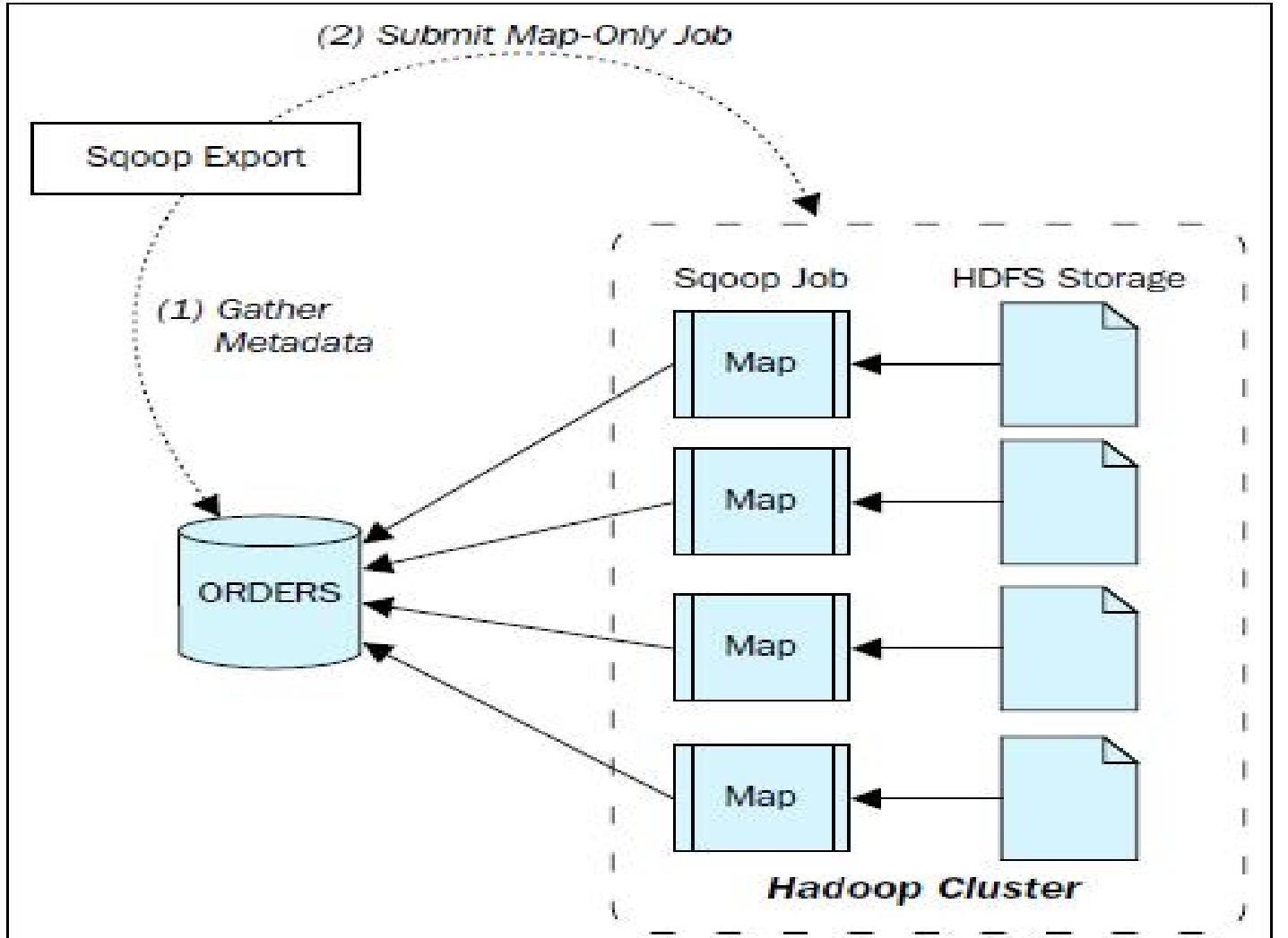
课程大纲

- 基于CDH 5.x版本Hadoop 2.x和Hive环境搭建
- Sqoop功能及使用要点
- Sqoop如何导入数据到HDFS
- Sqoop如何导出数据到RDBMS表
- Sqoop与Hive的结合

导出数据RDBMS

Sqoop Export is also in a similar process, only the source will be HDFS. Export is performed in two steps;

- Gather metadata
- Submit map-only job



课程大纲

- 基于CDH 5.x版本Hadoop 2.x和Hive环境搭建
- Sqoop功能及使用要点
- Sqoop如何导入数据到HDFS
- Sqoop如何导出数据到RDBMS表
- Sqoop与Hive的结合

Hive数据导入导出

- 使用Sqoop 将Hive表的数据与RDBMS表中数据，互为导入导出，其实对于Hive来说，数据本身就是存储在HDFS的目录下，所以Hive的数据导入导出实质还是RDBMS与HDFS数据导入导出。

从MySQL导入数据到Hive

- **原理与过程：**将数据从MySQL数据库中先放到HDFS用户主目录下，再将它从主目录下放到Hive在HDFS的目录中。

- **第一步、**在Hive中创建表

```
drop table if exists cc_1209.h_user ;
create table db_1206.h_user(
id int,
account string,
password string
)
ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t' ;
```

- **第二步、导出数据Hive表**

  - bin/sqoop import \
  - --connect jdbc:mysql://192.168.10.1:3306 /cc_123 \
  - --username root \
  - --password 123456 \
  - --table my_user \
  - --num-mappers 1 \
  - --fields-terminated-by "\t"
  - --delete-target-dir \
  - --hive-database cc_123 \
  - --hive-import \
  - --hive-table h_user

从Hive导出数据到MySQL

- 企业实际应用中，使用HiveQL分析数据时，常常将分析结果存储到Hive临时表中，然后使用Sqoop export将表中的数据导出到RDBMS对应表中，所以这个大家必须要会。从Hive表中将数据导出到MYSQL表中，实质就是HDFS文件数据导出到MYSQL表。

从Hive导出数据到MySQL

- **第一步、**在mysql中创建表

```
CREATE TABLE `user_export` (
  `id` tinyint(4) NOT NULL AUTO_INCREMENT,
  `account` varchar(255) DEFAULT NULL,
  `passwd` varchar(255) DEFAULT NULL,
  PRIMARY KEY (`id`)
);
```

- **第二步、**将Hive中的表导出到MYSQL中

```
bin/sqoop export \
--connect jdbc:mysql://hadoop-senior.ibeifeng.com:3306/cc_1209 \
--username root \
--password 123 \
--table user_export \
--num-mappers 1 \
--input-fields-terminated-by "\t" \
--export-dir /user/hive/warehouse/cc_1209.db/h_user
```

Sqoop 的--options-file使用

- 可以将Sqoop的命令选项写在文件，通过【--options-file】指定文件job.opt，进行运行程序。

- **第一步**、编写文件，写入程序

```
vim sqoop_script

export
--connect
jdbc:mysql://hadoop1:3306/sang
--username
root
--password
123456
--table
user_export
--num-mappers
1
--input-fields-terminated-by
"\t"
--export-dir
/user/hive/warehouse/test.db/h_user
```

第二步、使用—optins-file执行

```
sqoop --options--file ./sqoop_script
```

总结

- 大数据协作框架在大数据Hadoop 2.x生态系统中举足轻重，非常重要，针对不同需求诞生，辅助海量数据的存储和处理。
- Cloudera 发布的Hadoop版本CDH 5.x，有各种优势，其一各个框架版本间兼容问题，其二修复很多BUG和封装更好的接口以供实际项目中直接使用。
- Sqoop 就是RDBMS与HDFS直接数据转换工具，分为导入数据和导出数据，把握架构的核心及使用三要素，其底层是仅仅只有Map Task的MapReduce，充分利用分布式计算的功能。

作业练习

- 依据下面需求，导入MySQL表中数据到HDFS中：

  1）指定RDBMS表中部分字段导入

  --columns <col,col,col...>

  2）直接查询语句query

  --query

  3）条件查询

  --where <where clause>