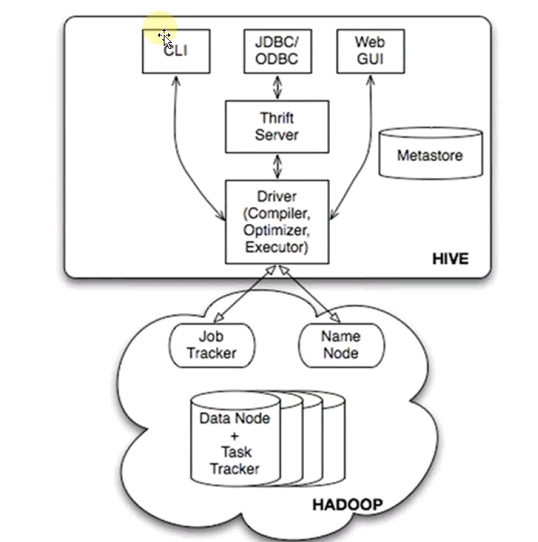
一:Hive的本质是什么?

1. Hive是分布式数据仓库，同时又是查询引擎，所以Spark SQL取代只是Hive查询引擎，在企业实际生产环境下Hive+Spark SQL是目前最为经典的数据分析组合；
2. Hive本身就是一个简单单机版本的软件，主要负责：
   1. 把HQL翻译成Mapper(s)-Reducer-Mapper(s)的代码，并且可能产生很多MapReduce的Job；
   2. 把生成的MapPedece代码及相关资源打包成为jar并发布到Hadoop集群中且进行运行（这一切都是自动的）；
3. Hive本身的架构如下所示：



1. Hive安装和配置
2. Hive 1.2.1
3. Hive默认情况下放元数据的是Derby，遗憾的是Derby是单用户，所以在生产环境下一般会支持多用户的数据库来进行MetaStore，且进行Master-Slaves主从读写分离和备份。我们最常使用MySQL
4. MySQL的安装和配置

(1)删除linux上已经安装的mysql相关库信息。rpm -e xxxxxxx --nodeps

执行命令rpm -qa |grep mysql 检查是否删除干净

(2)执行命令 rpm -i mysql-server-\*\*\*\*\*\*\*\* 安装mysql服务端

(3)启动mysql 服务端，执行命令 mysqld\_safe &

(4)执行命令 rpm -i mysql-client-\*\*\*\*\*\*\*\* 安装mysql客户端

(5)执行命令mysql\_secure\_installation设置root用户密码

1. 系统环境变量的配置：
   1. HIVE\_HOME
   2. HIVE\_CONF\_DIR=$HIVE\_HOME/conf
   3. PATH
2. cp hive-env.sh.template hive-env.sh

vim hive-env.sh

export HADOOP\_HOME=...

export HIVE\_HOME=...

export HIVE\_CONF\_DIR=...

1. cp hive-default.xml.template hive-site.xml

<name>javax.jdo.option.ConnectionURL</name>

<value>jdbc:mysql://Master:3306/hive?createDatabaseIfNotExist=true</value>

<description>JDBC connect string for a JDBC metastore</description>

<name>javax.jdo.option.ConnectionDirverName</name>

<value>com.mysql.jdbc.Driver</value>

<description>Driver class name for a JDBC metastore</description>

<name>javax.jdo.option.ConnectionUserName</name>

<value>root</value>

<description>usrname to use against metastore database</description>

<name>javax.jdo.option.ConnectionPassword</name>

<value>root</value>

<description>password to use against metastore database</description>

//hive数据仓库的数据的具体存储位置，里面放的就是简单的文件，这个文件里面有多少列数据， 每列数据的类型是什么从这个不知道，当数据放到这个里面的时候，hive已经把这个的元数据放进Mysql中了，所以你访问的时候，元数据相当于数据本身，所以就可以基于数据库中的数据表去进行各种查询了

<name>hive.metastore.warehouse.dir</name>

<value>/user/hive/warehouse</value>

<description>location of default database for the warehouse</description>

1. Hive的表有两种类型：一种内部表（这种表数据输入Hive本身，言外之意是如果原来的数据在HDFS的其他地方，此时数据会通过HDFS移动到Hive数据仓库所在的目录下，如果删除Hive中的该表的话数据和元数据均会被删除），另外一种属于外部表（这种表数据不属于Hive数据仓库，元数据中会表达具体数据在哪里，使用的时候和内部表的使用时一样的，只是如果通过Hive去删除的话，此时删除的只是元数据，并没有删除数据本身）；
2. Hive的启动

* 1、hive 命令行模式，直接输入#/hive/bin/hive的执行程序，或者输入 #hive --service cli
* 2、 hive web界面的 (端口号9999) 启动方式

#hive --service hwi &

用于通过浏览器来访问hive

http://hadoop0:9999/hwi/

* 3、 hive 远程服务 (端口号10000) 启动方式

#hive --service hiveserver &

Service mysqld start

#hive –service metastore &

1. 内部表

* 与数据库中的 Table 在概念上是类似
* 每一个 Table 在 Hive 中都有一个相应的目录存储数据。例如，一个表 test，它在 HDFS 中的路径为：/ warehouse/test。 warehouse是在 hive-site.xml 中由 ${hive.metastore.warehouse.dir} 指定的数据仓库的目录
* 所有的 Table 数据（不包括 External Table）都保存在这个目录中。
* 删除表时，元数据与数据都会被删除

CREATE TABLE t1(id int);

LOAD DATA LOCAL INPATH ‘/root/id’ INTO TABLE t1;

CREATE TABLE t2(id int,name string) ROW FORMAT DELIMITED FIELDS TERMINATED BY ‘\t’;

加载数据有两种方法：

第一种就是LOAD DATA LOCAL INPATH ‘/root/id’ INTO TABLE t1;

这种形式。如果是LOAD DATA INPATH ‘/root/id’ INTO TABLE t1; 则加载的是HDFS上面的数据；

第二种方式：直接通过hads命令上传文件到hive数据仓库中；

1. 分区表

* Partition 对应于数据库的 Partition 列的密集索引
* 在 Hive 中，表中的一个 Partition 对应于表下的一个目录，所有的 Partition 的数据都存储在对应的目录中

例如：test表中包含 date 和 city 两个 Partition，

则对应于date=20130201, city = bj 的 HDFS 子目录为：

/warehouse/test/date=20130201/city=bj

对应于date=20130202, city=sh 的HDFS 子目录为；

/warehouse/test/date=20130202/city=sh

CREATE TABLE t2(id int) PARTITIONED BY (day int);

LOAD DATA LOCAL INPATH ‘/root/id’ INTO TABLE t3 PARTITION (day=22);

1. 桶表

* 桶表是对数据进行哈希取值，然后放到不同文件中存储。
* 创建表

create table bucket\_table(id string) clustered by(id) into 4 buckets;

* 加载数据

set hive.enforce.bucketing = true;

insert into table bucket\_table select name from stu;

insert overwrite table bucket\_table select name from stu;

* 数据加载到桶表时，会对字段取hash值，然后与桶的数量取模。把数据放到对应的文件中。
* 抽样查询

select \* from bucket\_table tablesample(bucket 1 out of 4 on id);

create table t4(id int) clustered by(id) into 4 buckets;

set hive.enforce.bucketing = true;

insert into table t4 select id from t3;

1. 外部表

* 指向已经在 HDFS 中存在的数据，可以创建 Partition
* 它和 内部表 在元数据的组织上是相同的，而实际数据的存储则有较大的差异
* 内部表 的创建过程和数据加载过程（这两个过程可以在同一个语句中完成），在加载数据的过程中，实际数据会被移动到数据仓库目录中；之后对数据对访问将会直接在数据仓库目录中完成。删除表时，表中的数据和元数据将会被同时删除
* 外部表 只有一个过程，加载数据和创建表同时完成，并不会移动到数据仓库目录中，只是与外部数据建立一个链接。当删除一个 外部表 时，仅删除该链接

CREATE EXTERNAL t5(id int) location ‘/external’;

1. java客户端

* Hive远程服务启动#hive --service hiveserver >/dev/null 2>/dev/null &
* JAVA客户端相关代码

Class.*forName("org.apache.hadoop.hive.jdbc.HiveDriver");*

Connection con = DriverManager.*getConnection("jdbc:hive://192.168.1.102:10000/wlan\_dw", "", "");*

Statement stmt = con.createStatement();

String querySQL="SELECT \* FROM wlan\_dw.dim\_m order by flux desc limit 10";

ResultSet res = stmt.executeQuery(querySQL);

**while (res.next()) {**

System.*out.println(res.getString(1) +"\t" +res.getLong(2)+"\t" +res.getLong(3)+"\t" +res.getLong(4)+"\t" +res.getLong(5));*

}