Hive：数据仓库工具，可以将hadoop下的原始结构化数据变成Hive表，支持一种几乎和SQL完全相同的语言HiveQL，但不支持更新、索引、事务等，Hive可以将HiveQL映射成MapReduce作业

Hive起源自Facebook由Jeff Hammerbacher领导的团队，是构建在hadoop上的数据仓库框架，2008年Facebook将Hive项目贡献给Apache基金会

Hive组件与体系架构：用户接口有shell，thrift，web等，Thrift服务器，元数据库MetaStore（Derby，MySQL），解析器，hadoop

Hive的解释器、编译器和优化器将HiveQL语句进行词法分析（ANTLR）、语法分析、编译、优化以及生成查询计划并放到HDFS中通过MapReduce调用执行

Hive编译器将HiveQL转换成Hive的最小处理单元操作符（Operator），每个操作符对应HDFS的一个操作或者一次MapReduce作业，Hive底层的操作符Operator使用的是树形结构

Hive安装：

内嵌模式：元数据保持在内嵌的Derby数据库模式，只允许一个会话连接

单用户模式：把元数据放到MySQL中即用用外部MySQL替换内嵌的Derby数据库，只允许一个会话连接

多用户模式：远程服务模式，客户端使用Thrift协议通过服务端启用的MetaStoreServer访问元数据库，允许多个会话连接

Hive内嵌模式，下载解压后配置：

修改hive-config.sh文件

export JAVA\_HOME=java安装路径

export HIVE\_HOME=hive安装路径

export HADOOP\_HOME=hadoop安装路径

复制hive-default.xml为hive-site.xml并修改：

hive.metastore.warehouse.dir:HDFS的数据目录，默认/user/hive/warehouse

hive.exec.scratchdir:HDFS的临时文件目录，默认/tmp/hive-${user.name}

改变hive安装目录的所有者为hadoop集群用户

复制hive安装目录/config/hive-log4j.properties.template为hive-log4j.properties文件并修改：

org.apache.hadoop.metrics.jvm.EventCounter为org.apache.hadoop.log.metrics.EventCounter

Hive单用户模式：

安装MySQL并启动服务

ALTER USER 'root'@'localhost' IDENTIFIED BY 'root-password';修改密码

create user '用户名' @ '访问主机' identified by '密码'; 新版MySQL创建用户

grant 权限列表 on 数据库 to '用户名' @ '访问主机'; 新版MySQL赋予权限，修改权限时后加with grant option

grant all privileges on \*.\* to 'root'@'%' with grant option; 旧版MySQL赋予root用户的所有表所有IP地址可以访问的权限

flush privileges;刷新权限

修改Hive安装目录/conf/hive-site.xml文件：

<configuration>

<property>（Hive数据在HDFS上存放的目录）

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

<value>/user/hive\_remote/warehouse</value>

</property>

<property>（单用户模式）

<name>hive.metastore.local</name>

<value>true</value>

</property>

<property>（元数据库连接的URL）

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

<value>jdbc:mysql://hadoop3/hive?createDatabaseIfNotExist=true</value>

</property>

<property>（元数据库连接驱动名）

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

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

</property>

<property>（元数据库用户名）

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

<value>root</value>

</property>

<property>（元数据库密码）

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

<value>root</value>

</property>

</configuration>

MySQL的jdbc连接包复制到Hive安装目录的lib目录下

高版本的Hive需要使用命令schematool -dbType mysql -initSchema手动初始化Hive的元数据库

Hive多用户模式配置：

MetaStoreServer服务端修改hive-site.xml文件配置：

<configuration>

<property>（Hive数据在HDFS上存放的目录）

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

<value>/user/hive\_remote/warehouse</value>

</property>

<property>（元数据库连接的URL）

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

<value>jdbc:mysql://hadoop3/hive?createDatabaseIfNotExist=true</value>

</property>

<property>（元数据库连接驱动名）

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

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

</property>

<property>（元数据库用户名）

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

<value>root</value>

</property>

<property>（元数据库密码）

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

<value>root</value>

</property>

</configuration>

MySQL的jdbc连接包复制到Hive安装目录的lib目录下

高版本的Hive需要在MetaStoreServer上使用命令schematool -dbType mysql -initSchema手动初始化Hive的元数据库

使用hive --service metastore命令启动，一般使用hive --service metastore &命令后台启动

客户端修改hive-site.xml文件配置：

<configuration>

<property>

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

<value>/user/hive\_remote/warehouse</value>

</property>

<property>（多用户模式）

<name>hive.metastore.local</name>

<value>false</value>

</property>

<property>（metastore的thrift连接uri）

<name>hive.metastore.uris</name>

<value>thrift://hadoop2:9083</value>

</property>

</configuration>

HiveQL的语言标准大约相当于SQL92标准，Hive shell可以查看或临时设置配置Hive的参数，但只对当前会话有效

Hive可以像传统数据库一样使用JDBC或者ODBC连接，但需要将Hive和Hadoop的lib下的所有包复制到项目的类路径下

Hive的WEB的浏览器UI访问为<http://Hive>所在的主机的IP地址:9999/hwi/（端口默认9999）

Hive的数据存放在HDFS的warehouse目录下，一般每个表对应一个子目录，Hive桶会对应HDFS的warehouse目录下的一个子目录，Hive桶与Reduce相对应，本地的/tmp目录存放Hive的日志和执行计划

Hive的基本数据类型：TINYINT、SMALLINT、INT、BIGINT、BOOLEAN、FLOAT、DOUBLE、STRING、array、map和struct

CREATE [EXTERNAL（是否为外部表，表的数据存放在指定的HDFS路径名，Hive管理数据表结构不管理数据）] TABLE 表名(域名 域类型[,域名 域类型（字段类型）]) ROW FORMAT DELIMITED [FIELDS TERMINATED BY '域分隔符（支持正则表达式）' [ESCAPED BY char]] [COLLECTION ITEMS TERMINATED BY '集合类型分隔符'] [MAP KEYS TERMINATED BY 'map类键值对分隔符'] [LINES TERMINATED BY '行分隔符默认\n'] LOCATION '外部表的HDFS保存路径，默认为Hive的HDFS数据保存目录'

Create table 表名 like 表名; 创建指定表复制指定表名的结构

LOAD DATA [LOCAL] INPATH 'filepath' [OVERWRITE] INTO TABLE tablename [PARTITION (partcol1=val1, partcol2=val2 ...)] Hive以加载文件的方式将文件数据以表逻辑形式插入到Hive表中，Hive将文件上传到HDFS并移动到指定的目录下，文件内容不发生变化，只是Hive表读取数据时以指定的表逻辑形式读取

ALTER TABLE 表名 ADD/DROP PARTITION (分区名=分区值); 添加/删除指定分区到指定表名上

HiveServer2之后可以使用Beeline连接方式进行连接，支持嵌入模式和远程模式

在Hive服务端的Hive安装目录/bin/hiveserver2启动服务后使用beeline进行连接

默认配置文件hive.server2.authentication为NONE用户名和密码不验证，但由于数据存放在HDFS上会有HDFS的权限认证（用户名）

Beeline没有密码显示不允许连接可能是HDFS认证问题需要在Hadoop安装目录/etc/hadoop/core-site.xml中加入

<property>

<name>hadoop.proxyuser.用户名.hosts</name>

<value>\*</value>

</property>

<property>

<name>hadoop.proxyuser.用户名.groups</name>

<value>\*</value>

</property>

Hive的自定义函数有是三种：一进一出UDF（User-Defined-Function）、多进一出聚合函数UDAF（User-Defined Aggregation Function）和一进多出UDTF（User-Defined Table-Generating Function）

Hive自定义函数UDF需要继承org.apache.hadoop.hive.ql.UDF，实现支持重载的evaluate函数，将程序打jar包上传至Hive机器节点，使用Hive客户端命令add jar jar包路径名;进行添加自定义函数需要的jar包，使用create temporary function 临时函数方法名 as '自定义方法的全类名';添加临时自定义函数，可以使用命令drop temporary function 临时函数方法名;删除指定临时函数