

Hive从概念到安装使用总结

6 回复 66 查看



(<https://www.shiyanlou.com/user/8490>) 实验楼管理员



(<https://www.shiyanlou.com/vip>)

2016-01-19 15:47

来自: HIVE教程 (<https://www.shiyanlou.com/questions/courses/38>)

技术分享 (<https://www.shiyanlou.com/questions/?tag=技术分享>)

文章从Hive的概念到安装使用都做了较为详细的总结~

分享到微博

全部回答



实验楼管理员 (<https://www.shiyanlou.com/user/8490>)



(<https://www.shiyanlou.com/vip>)

一、Hive的基本概念

1.1 hive是什么？

- (1) Hive是建立在hadoop数据仓库基础之上的一个基础架构；
- (2) 相当于hadoop之上的一个客户端，可以用来存储、查询和分析存储在hadoop中的数据；
- (3) 是一种SQL解析引擎，能够将SQL转换成Map/Reduce中的Job在hadoop上执行。

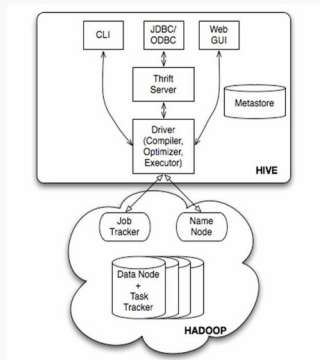
1.2 hive的数据存储特点

- (1) 数据存储是基于hadoop的HDFS；
- (2) 没有专门的数据存储格式；
- (3) 存储结构主要有：数据库、文件（默认可以直接加载文本文件）、表、视图、索引；

说明：hive中的表实质就是HDFS的目录，按表名将文件夹分开，若是分区表，则分区值是子文件夹。这些数据可以直接在M/R中使用。hive中的数据是存放在HDFS中的。

二、hive的系统结构

存储hive的元数据（表及表的属性、数据库名字等）



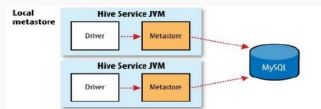
分析执行hive QL语句，将执行计划投递给hadoop，转到map/reduce执行

2.1 hive的系统结构– metastore存储方式

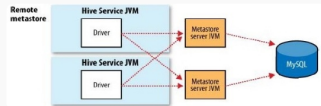
默认情况，元数据使用内嵌的derby数据库作为存储引擎



将存储数据独立出来，支持多用户同时访问



将metastore独立出来，远程方法调用



三、hive的安装与使用

3.1 下载hive源文件，解压hive文件

进入 `$HIVE_HOME/conf/` 修改文件

```
cp hive-default.xml.template hive-site.xml
cp hive-env.sh.template hive-env.sh
```

修改 `$HIVE_HOME/bin` 的 `hive-env.sh`，增加以下三行

```
HADOOP_HOME=    --hadoop的home目录
export HIVE_CONF_DIR=    --hive的conf目录
export HIVE_AUX_JARS_PATH=    --hive的lib目录
```

生效文件：

```
source /hive-env.sh(生效文件)
```

2016-01-19 15:48



实验楼管理员 (<https://www.shiyanlou.com/user/8490>) 💛 (<https://www.shiyanlou.com/vip>)

(<https://www.shiyanlou.com/user/8490>)

3.2 配置MySQL的metastore

修改 `$HIVE_HOME/conf/hive-site.xml`

```
<property>

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

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

</property>

<property>

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

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

</property>

<property>

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

<value>root</value>

</property>

<property>

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

<value>123456</value>

</property>
```

3.3hive临时目录的配置

修改 \$HIVE_HOME/conf/hive-site.xml

(1)设定数据目录

```
<property>

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

<value>/usr/local/hive/warehouse</value>

</property>
```

(2)设定临时文件目录

```
<property>

<name>hive.exec.scratchdir</name>

<value>/usr/local/hive/tmp</value>

</property>
```

(3)hive相关日志的目录

```
<property>

<name>hive.querylog.location</name>

<value>/usr/local/hive/log</value>

</property>
```

3.4hive的运行模式的指定

Hive的运行模式即任务的执行环境，分为本地与集群两种，我们可以通过 `mapred.job.tracker` 来指明 本地模式设置方式：

```
hive > set mapred.job.tracker=local;
hive > set hive.exec.mode.local.auto=true;
hive.exec.mode.local.auto.inputbytes.max默认128M
```

2016-01-19 15:48



实验楼管理员 (<https://www.shiyanlou.com/user/8490>) (<https://www.shiyanlou.com/vip>)

(<https://www.shiyanlou.com/user/8490>)

3.5 sqoop的安装

(1)下载、解压：

```
tar -zxvf sqoop-1.4.4.bin__hadoop-2.0.4-alpha.tar.gz /root
cd /root
ln -s sqoop-1.4.3.bin sqoop
```

(2)配置sqoop：

```
vi ~/.bash_profile
export SQOOP_HOME=/usr/local/sqoop
export PATH=$SQOOP_HOME/bin:$PATH
```

(3)测试连接数据库并列出数据库：

```
sqoop list-databases --connect jdbc:mysql://localhost:3306/ --username root --password 123456
```

(4)将mysql中的表导入到hive中：

```
sqoop import --connect jdbc:mysql://localhost:3306/gwifi --username root --password 123456 --table thi
nk_access --hive-import -m 1;
```

3.6 hive的命令行方式

1、输入 `#!/hive/bin/hive` 执行应用程序， 或者

```
#!/hive
hive> create table test(id int, name string);
hive> show tables;
hive>quit;
```

查看并修改表与目录之间的关系

```
#!/hadoop fs -ls /user/hive/warehouse/
```

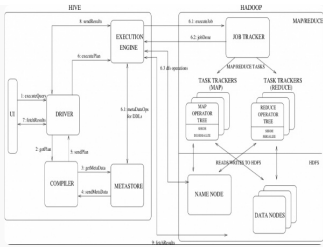
修改参数： `hive.metastore.warehouse.dir` 表与目录的对应关系

3.6 命令行方式

显示或修改参数值

在代码中可以使用 `${...}` 来使用

命名空间	使用权限	描述
hivevar	可读写	\$ hive -d name=zhangsan;
hiveconf	可读写	\$ hive -hiveconf hive.cli.prin t.current.db=true\$ hive -hiv econf hive.cli.print.header=t rue;
system	可读写	java运行时的配置属性，如yste m.user.name
env	只读	shell环境变量，如nameUSER



3.7 hive的脚本运行

```
$>hive -i /home/my/hive-init.sql  
$hive>source file
```

与linux交互命令！

```
!ls  
!pwd
```

与hdfs交互命令

```
dfs -ls /  
dfs -mkdir /hive
```

3.8 hive的jdbc模式

JAVA API交互执行方式

hive 远程服务 (端口号10000) 启动方式

3.9 hive常用的命令– set命令

hive控制台set命令:


```
set hive.cli.print.current.db=true;  
set hive.metastore.warehouse.dir=/hive
```

hive参数初始化配置set命令:

```
~/hive.rc
```

2016-01-19 15:49



实验楼管理员 (<https://www.shiyanlou.com/user/8490>)  (<https://www.shiyanlou.com/vip>)

(<https://www.shiyanlou.com/user/8490>)

四、HiveQL数据操作

4.1数据类型

- 1、基本数据类型：与mysql等数据库中基本数据类型类似；
- 2、复合数据类型：
 - (1) array 数组类型 如： array[int] 下标访问
 - (2) struct 结构类型 如： struct{name:STRING,age:INT} .访问
 - (3) Map结构

4.2 数据库/表的定义、操作

默认使用的是“default”数据库，使用命令选择数据库：

```
hive> use <数据库名>
```

创建数据库：

```
create database <数据库名>
```

查看所有数据库：

```
show databases;
```

查看/删除数据库：

```
desc/drop database <数据库名>;
```

注：Hive没有 行级别的插入，更新和删除操作，往表中插入数据的唯一方法就是使用成批载入操作

```
hive>create table 表名(字段名 字段类型,.....)
hive>show tables;
hive>create table t2 like t1;
hive> drop table 表名      -删除表
```

增加列

```
hive>ALTER TABLE t3 ADD COLUMNS(gender int);
```

在mysql中hive数据库中 show tables ； 在TBLS表中可以查看到hie创建的表。

4.3 数据库/表的定义、操作

插入数据：

```
insert overwrite table t_table1 select * from t_table1 where XXXX;
```

删除数据：

```
insert overwrite table test select * from test where 1=0;
```

数组类型的表的操作：

定义复合数据类型的表：

```
create table demo_array(id int, mydata array[string])  PARTITIONED BY (dt STRING)  row format delimit
d fields terminated by '\t' collection items terminated by '|';
```

-id 与mydata之间是 '\t' 隔开，其后的mydata数据之间用 '|' 隔开

4.3.1 Hive的数据模型-管理表

管理表，也称作内部表或受控表

特点：

- （1）数据全部保存在warehouse目录中；
- （2）删除表时元数据和表中的数据都会被删除；

- (3) 创建表和数据加载可以在同一条语句中实现；
- (4) 每个表在HDFS中都有相应的目录用来存储表的数据
- (5) 加载数据的过程，实际数据会被移动到数据仓库目录中；对数据的访问是在数据仓库目录中完成。

4.3.1 Hive的数据模型-管理表

创建数据文件 inner_table.dat

创建表

```
hive>create table inner_table (key string)
row format delimited fields terminated by '\t';
//这个要指定，否则load的时候数据为NULL;
```

加载数据


```
hive>load data local inpath '/root/inner_table.dat' into table inner_table;
```

查看数据

```
select * from inner_table
select count(*) from inner_table
删除表 drop table inner_table
```

2016-01-19 15:50



实验楼管理员 (<https://www.shiyanlou.com/user/8490>)  (<https://www.shiyanlou.com/vip>)

(<https://www.shiyanlou.com/user/8490>)

4.3.2 Hive的数据模型-外部表

包含externable的表叫做外部表

特点：

- (1) 删除外部表只删除metastore的元数据，不删除hdfs中的表数据；
- (2) 加载数据和创建表是同时完成的，并不会移动到数据，只是与外部数据建立一个链接；删除一个外部表，只是删除了该链接
- (3) 指向已经在 HDFS 中存在的数据

4.3.2 Hive的数据模型-外部表语法

```

CREATE EXTERNAL TABLE page_view

( viewTime INT,

userid BIGINT,

page_url STRING,

referrer_url STRING,

ip STRING COMMENT 'IP Address of the User',

country STRING COMMENT 'country of origination'

)

COMMENT 'This is the staging page view table'

ROW FORMAT DELIMITED FIELDS TERMINATED BY '44'  LINES  TERMINATED BY '12'

STORED AS TEXTFILE

LOCATION 'hdfs://centos:9000/user/data/staging/page_view';

```

4.3.3 Hive的数据模型-分区表

分区可以理解为分类，通过分类把不同类型的数据放到不同的目录下；

分类的标准就是分区字段，可以一个，也可以多个；

分区表的意义在于优化查询，查询时尽量利用分区字段；如果不使用分区字段，就会全部扫描。

创建数据文件 partition_table.dat

创建表

```

create table partition_table(rectime string,msisdn string) partitioned by(daytime string,city string)
row format delimited fields terminated by '\t' stored as TEXTFILE;

```

加载数据到分区

```

load data local inpath '/home/partition_table.dat' into table partition_table partition (daytime='2013-02-01',city='bj');

```

查看数据

```

select * from partition_table
select count(*) from partition_table
删除表 drop table partition_table

```

4.3.4 Hive的数据模型-分区表


```
CREATE TABLE tmp_table #表名

(

title    string, # 字段名称 字段类型

minimum_bid    double,

quantity    bigint,

have_invoice    bigint

) COMMENT '注释: XXX' #表注释

PARTITIONED BY(pt STRING) #分区表字段（如果你文件非常之大的话，采用分区表可以快过滤出按分区字段划分的数据）


ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\001'    # 字段是用什么分割开的

STORED AS SEQUENCEFILE; #用哪种方式存储数据，SEQUENCEFILE是hadoop自带的文件压缩格式
```

2016-01-19 15:50



实验楼管理员 (<https://www.shiyanlou.com/user/8490>)  (<https://www.shiyanlou.com/vip>)

(<https://www.shiyanlou.com/user/8490>)

4.4 装载数据

4.4.1 装载数据

从文件中装载数据

```
hive>LOAD DATA [LOCAL] INPATH '...' [OVERWRITE] INTO TABLE t2 [PARTITION (province='beijing')];
```

通过查询表重新装载数据

```
hive>INSERT OVERWRITE TABLE t2 PARTITION (province='beijing') SELECT * FROM xxx WHERE xx
```

设置job并行数量 `hive.exec.parallel=true`;

```
hive.exec.parallel.thread.number =3;
```

4.4.2 动态分区装载数据

开启动态分区支持

```
hive>set hive.exec.dynamic.partition=true;

hive>set hive.exec.dynamic.partition.mode=nostrict;

hive>set hive.exec.max.dynamic.partitions.pernode=1000;
```

查询字段一样

```
hive>INSERT OVERWRITE TABLE t3 PARTITION(province, city)

SELECT t.province, t.city FROM temp t;

hive>INSERT OVERWRITE TABLE t3 PARTITION(province='bj', city)

SELECT t.province, t.city FROM temp t WHERE t.province='bj';
```

```
hive>CREATE TABLE t4 AS SELECT ....
```

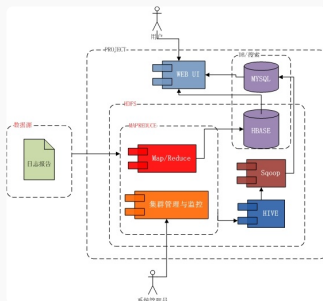
```
select count(0) from (select id from test where name like 'zh%') a join (select id from test where name like '%i%') b on a.id = b.id;
```

五、hive的存储形式比较

hive在建表时，可以通过 'STORED AS FILE_FORMAT' 指定存储文件格式。有以下几种：

- 1.TextFile：存储空间较大，压缩之后的文件不能分割与合并，查询效率低；可直接存储，加载速度最快；
- 2.sequencefile：hadoop API提供的一种二进制文件支持，存储空间最大，可分割与合并，查询效率高，需要text文件转换来加载
- 3.RcFile:是一种行列存储相结合的存储方式。
 - （1）将数据按行分块，保证同一个record在一个块上，避免读一个记录需要读取多个block；
 - （2）块数据列式存储，有利于数据压缩和快速的列存取。查询效率最高、存储空间最小、但加载最慢

总结



本文链接：<http://www.codeceo.com/article/hive-usage.html> (<http://www.codeceo.com/article/hive-usage.html>)

本文作者：码农网 (<http://www.codeceo.com/>) – 朱鹏飞

2016-01-19 15:51

登录后才能回答问题哟~

我要提问

标签

Linux (<https://www.shiyanlou.com/questions/?tag=Linux>) Python (<https://www.shiyanlou.com/questions/?tag=Python>)

C/C++ (<https://www.shiyanlou.com/questions/?tag=C/C++>) 实验环境 (<https://www.shiyanlou.com/questions/?tag=实验环境>)

技术分享 (<https://www.shiyanlou.com/questions/?tag=技术分享>) 功能建议 (<https://www.shiyanlou.com/questions/?tag=功能建议>)

课程需求 (<https://www.shiyanlou.com/questions/?tag=课程需求>) Java (<https://www.shiyanlou.com/questions/?tag=Java>)

其他 (<https://www.shiyanlou.com/questions/?tag=其他>) SQL (<https://www.shiyanlou.com/questions/?tag=SQL>)

NodeJS (<https://www.shiyanlou.com/questions/?tag=NodeJS>) Hadoop (<https://www.shiyanlou.com/questions/?tag=Hadoop>)

Web (<https://www.shiyanlou.com/questions/?tag=Web>) 常见问题 (<https://www.shiyanlou.com/questions/?tag=常见问题>)

Shell (<https://www.shiyanlou.com/questions/?tag=Shell>) PHP (<https://www.shiyanlou.com/questions/?tag=PHP>)

Git (<https://www.shiyanlou.com/questions/?tag=Git>) HTML (<https://www.shiyanlou.com/questions/?tag=HTML>)

HTML5 (<https://www.shiyanlou.com/questions/?tag=HTML5>) 信息安全 (<https://www.shiyanlou.com/questions/?tag=信息安全>)

网络 (<https://www.shiyanlou.com/questions/?tag=网络>) GO (<https://www.shiyanlou.com/questions/?tag=GO>)

NoSQL (<https://www.shiyanlou.com/questions/?tag=NoSQL>) 训练营 (<https://www.shiyanlou.com/questions/?tag=训练营>)

Android (<https://www.shiyanlou.com/questions/?tag=Android>) Ruby (<https://www.shiyanlou.com/questions/?tag=Ruby>)

Perl (<https://www.shiyanlou.com/questions/?tag=Perl>)

相关问题

bug解决 (<https://www.shiyanlou.com/questions/3026>)

MySQL之终端（Terminal）管理数据库、数据表、数据的基本操作 (<https://www.shiyanlou.com/questions/3019>)

C++静态库与动态库 (<https://www.shiyanlou.com/questions/3017>)

谈Runtime机制和使用的整体化梳理 (<https://www.shiyanlou.com/questions/3010>)

JavaScript：彻底理解同步、异步和事件循环(Event Loop) (<https://www.shiyanlou.com/questions/3009>)

动手做实验，轻松学IT。

实验楼-通过动手实践的方式学会IT技术。

公司简介 (<https://www.shiyanlou.com/aboutus>) 联系我们 (<https://www.shiyanlou.com/contact>) 常见问题 (<https://www.shiyanlou.com/faq#howtostart>)
我要开课 (<https://www.shiyanlou.com/labs>) 隐私协议 (<https://www.shiyanlou.com/privacy>) 会员条款 (<https://www.shiyanlou.com/terms>)
友情链接 (<https://www.shiyanlou.com/friends>)
站长统计 (http://www.cnzz.com/stat/website.php?web_id=5902315) 蜀ICP备13019762号 (<http://www.miibeian.gov.cn/>)



QQ群



微信



微博
(<http://weibo.com/shiyanlou2013>)