

大数据数据库系统

6.3 Hive常用属性配置、交互指令以及数据类型简介

Hive常用属性配置、交互指令以及数据类型简介

◆ 主要内容

6.3.1 Hive常用属性配置

6.3.2 Hive常用交互指令

6.3.3 Hive常见的数据类型

6.3.1 Hive常用属性配置

◆ HIVE的配置文件

/usr/local/hive/conf目录下

HIVE默认配置文件：hive-default.xml

HIVE用户自定义配置文件：hive-site.xml

注意： 用户自定义配置会覆盖默认配置；

配置文件的设定对本机启动的所有 Hive 进程都有效

如果用户在运行hive过程中或者代码中，想临时修改某个配置项，可以用set命令行传递参数，修改配置（后面会讲）

-- INSERT --

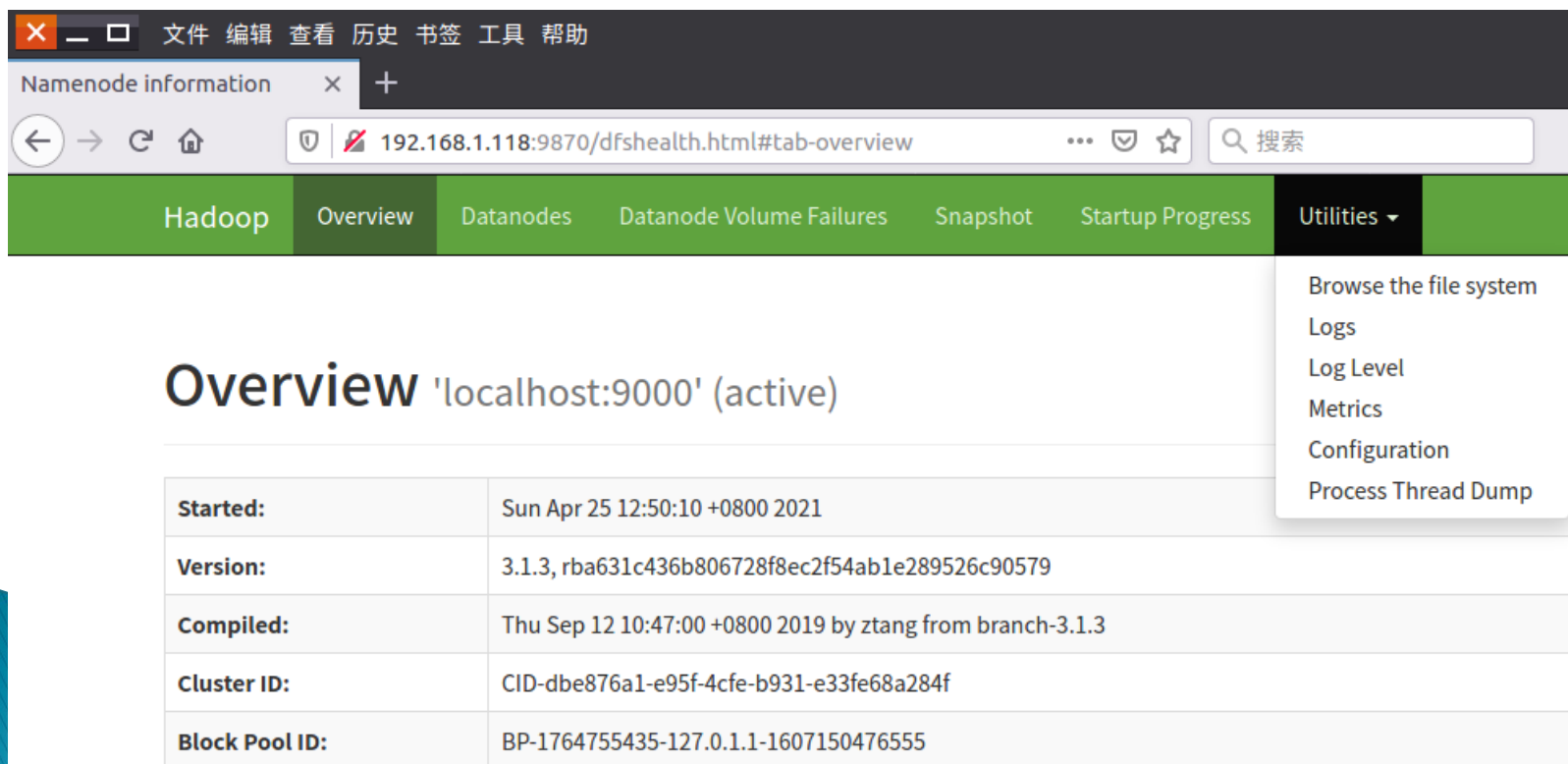
6.3.1 Hive常用属性配置

◆ 在web上浏览HDFS目录

1、关闭防火墙

`sudo ufw disable`

2、打开浏览器，输入“ip地址:9807”（例如：192.168.1.118:9807）



Overview 'localhost:9000' (active)

Started:	Sun Apr 25 12:50:10 +0800 2021
Version:	3.1.3, rba631c436b806728f8ec2f54ab1e289526c90579
Compiled:	Thu Sep 12 10:47:00 +0800 2019 by ztang from branch-3.1.3
Cluster ID:	CID-dbe876a1-e95f-4cfe-b931-e33fe68a284f
Block Pool ID:	BP-1764755435-127.0.1.1-1607150476555

6.3.1 Hive常用属性配置

◆ 数据仓库目录的配置

Hive默认数据库存放位置: /user/hive/warehouse

<input type="text" value="/user/hive/warehouse"/>							Go!
Permission	Owner	Group	Size	Replication	Block Size	Name	
drwxr-xr-x	hpsk	supergroup	0 B	0	0 B	student.db	

创建的所有表，都以目录的形式在对应的数据库目录下面

<input type="text" value="/user/hive/warehouse/student.db"/>							Go!
Permission	Owner	Group	Size	Replication	Block Size	Name	
drwxr-xr-x	hpsk	supergroup	0 B	0	0 B	stu_info	

表里面的数据会保存在表的目录下面

<input type="text" value="/user/hive/warehouse/student.db/stu_info"/>							Go!
Permission	Owner	Group	Size	Replication	Block Size	Name	
-rw-r--r--	hpsk	supergroup	49 B	1	128 MB	student.txt	

6.3.1 Hive常用属性配置

◆ 那默认的数据库存放目录该如何更改呢？

进入/hive/conf/hive-default.xml中可以找到配置项

```
<property>  
  <name>hive.metastore.warehouse.dir</name>  
  <value>/user/hive/warehouse</value>  
  <description>location of default database for the warehouse</description>  
</property>
```

该配置项决定数据库默认存放的目录

把这段配置项复制到hive-site.xml文件中，再将value值改成需要的目录即可

6.3.1 Hive常用属性配置

◆ 显示提示符表头信息

1、使用show databases显示当前所有数据库

```
hive> show databases;  
OK  
default  
student  
Time taken: 0.423 seconds, Fetched: 2 row(s)
```

有两个数据库，这时我们使用use student，使用student数据库

```
hive> use student;  
OK  
Time taken: 0.014 seconds  
hive>
```

可以看到，执行成功后，没有任何变化，看不出来我到底在那个数据库中，不方便操作

6.3.1 Hive常用属性配置

2、使用语句：select * from stu_info查看stu_info表

```
hive> select * from stu_info;  
OK  
1001      zhangsan  
1002      lisi  
1003      wangwu  
1004      zhao Liu  
Time taken: 0.889 seconds, Fetched: 4 row(s)  
hive> █
```

表里有两列，由于列数不多，我们可以看出每一列是什么字段，第一列是学号，第二列是姓名

假定有10列数据，那如何区分那一列是什么字段呢？

6.3.1 Hive常用属性配置

◆ 通过修改配置文件来解决上述问题

1、进入/hive/conf/hive-default.xml文件，找到下列配置信息

```
<property>
  <name>hive.cli.print.header</name>
  <value>>false</value>
  <description>Whether to print the names of the columns in query output.</description>
</property>

<property>
  <name>hive.cli.print.current.db</name>
  <value>>false</value>
  <description>Whether to include the current database in the Hive prompt.</description>
</property>
```

从字面就可以知道，分别为显示表头信息与显示当前数据库，但默认的value都是false

6.3.1 Hive常用属性配置

2、将配置项复制并粘贴到hive-site.xml中，并将两个配置项的value由false改成true，修改后重启hive，这时我们使用use student命令

```
hive (default)> use student;  
OK  
Time taken: 0.346 seconds  
hive (student)> █
```

命令行显示当前数据库名称，再使用 select * from stu_info查看stu_info表

```
OK  
stu_info.number stu_info.name  
1001    zhangsan  
1002    lisi  
1003    wangwu  
1004    zhaoliu  
Time taken: 1.183 seconds, Fetched: 4 row(s)  
hive (default)> █
```

成功显示每一列的字段名

6.3.1 Hive常用属性配置

◆ 还可以通过其他方式修改配置项

在启动hive的命令行添加参数-hiveconf 如：

bin/hive -hiveconf hive.cli.print.current.db=false;

```
[atguigu@hadoop102 hive]$ bin/hive -hiveconf hive.cli.print.current.db=false;
which: no hbase in (/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/opt/module
in:/opt/module/hadoop-3.1.3/bin:/opt/module/hadoop-3.1.3/sbin:/home/atguigu/.loc
uigu/bin:/opt/module/jdk1.8.0_212/bin:/opt/module/hadoop-3.1.3/bin:/opt/module/h
:/opt/module/hive/bin)
Hive Session ID = 9ec7f106-d27b-40fe-96ed-b91be2cf2b4e

Logging initialized using configuration in file:/opt/module/hive/conf/hive-log4j
nc: true
Hive-on-MR is deprecated in Hive 2 and may not be available in the future versio
ng a different execution engine (i.e. spark, tez) or using Hive 1.X releases.
Hive Session ID = b6592ad2-75db-467c-b6ab-7bb7dd19436e
hive> █
```

进入hive后发现数据库名称又消失了，注意：这种方式仅对本次 hive 启动有效

6.3.1 Hive常用属性配置

◆ 还可以通过其他方式修改配置项

进入hive以后，通过set命令修改配置项

- 例：set hive.cli.print.header=false;

```
hive> set hive.cli.print.header=false;
hive> select * from test;
OK
1001
1002
1003
Time taken: 0.184 seconds, Fetched: 3 row(s)
hive> █
```

- 注意：这种方式仅对本次 hive 启动有效


6.3.1 Hive常用属性配置

- ◆ 如果只想查看某个配置项的值，而不想修改

格式：set 配置项名称

- 例：set hive.cli.print.header;

```
hive (default)> set hive.cli.print.header;  
hive.cli.print.header=false  
hive (default)>
```



6.3.1 Hive常用属性配置

◆ Hive 常用交互命令

在启动hive前，查看帮助

bin/hive -help

```
[hpsk@bigdata-training01 hive-0.13.1-bin]$ bin/hive -help
usage: hive
  -d,--define <key=value>          Variable substitution to apply to hive
                                   commands. e.g. -d A=B or --define A=B
  --database <databasename>        Specify the database to use
  -e <quoted-query-string>         SQL from command line
  -f <filename>                     SQL from files
  -H,--help                         Print help information
  -h <hostname>                     connecting to Hive Server on remote host
  --hiveconf <property=value>      Use value for given property
  --hivevar <key=value>            Variable substitution to apply to hive
                                   commands. e.g. --hivevar A=B
  -i <filename>                     Initialization SQL file
  -p <port>                         connecting to Hive Server on port number
  -S,--silent                       Silent mode in interactive shell
  -v,--verbose                      Verbose mode (echo executed SQL to the
                                   console)
```

可以看到书写hive启动命令时，可以在“hive”后跟上许多参数

6.3.2 Hive常用交互指令

◆ 启动时指定进入某个数据库

--database dbname dbname表示数据库名称

例:

- 1、bin/hive --database student

```
[hpsk@bigdata-training01 hive-0.13.1-bin]$ bin/hive --database student
Logging initialized using configuration in file:/opt/modules/hive-0.13
s
OK
Time taken: 0.47 seconds
hive (student)>
```

- 2、show tables查看数据库中的表

```
hive (student)> show tables;
OK
tab_name
stu_info
Time taken: 0.106 seconds, Fetched: 1 row(s)
hive (student)> █
```


6.3.2 Hive常用交互指令

◆ 在命令行执行SQL语句:单条SQL语句

-e <quoted-query-string> SQL from command line

可以不进入hive shell执行单条sql语句，即在linux命令行执行单条hive语句

- 例：bin/hive -e “show databases”

```
[hpsk@bigdata-training01 hive-0.13.1-bin]$ bin/hive -e "show databases"
Logging initialized using configuration in file:/opt/modules/hive-0.13.1-
s
OK
database_name
default
student
Time taken: 0.568 seconds, Fetched: 2 row(s)
[hpsk@bigdata-training01 hive-0.13.1-bin]$
```

语句记得加双引号，语句结尾可以不用分号

结尾可以看到会自动退出hive回到linux命令行

6.3.2 Hive常用交互指令

◆ 如果想将结果保存到文件中，可以使用重定向

例：1、输入bin/hive -e "show databases" >> /opt/datas/hive.exec.log

```
[hpsk@bigdata-training01 hive-0.13.1-bin]$ bin/hive -e "show databases" >> /opt/datas/hive.exec.log
Logging initialized using configuration in file:/opt/modules/hive-0.13.1-bin/conf/hive-log4j.properties
OK
Time taken: 0.596 seconds, Fetched: 2 row(s)
[hpsk@bigdata-training01 hive-0.13.1-bin]$
```

发现结果并没有输出到屏幕上

2、看下文件中有没有结果

```
[hpsk@bigdata-training01 hive-0.13.1-bin]$ more /opt/datas/hive.exec.log
database_name
default
student
[hpsk@bigdata-training01 hive-0.13.1-bin]$
```

文件结果输出到了文件中

6.3.2 Hive常用交互指令

◆ 在命令行执行一个包含多条SQL语句的文件

由于-e只能执行一条语句

因此可以使用-f参数，执行一个包含多条SQL语句的文件

-f <filename> SQL from files

- 例：我们先创建一个.sql文件：vim /opt/datas/hivetest.sql，输入如下三条指令，记得添加分号，保存

```
show databases;  
use student;  
show tables;■
```

在linux命令行输入

bin/hive -f /opt/datas/hivetest.sql, 执行结果

```
[hpsk@bigdata-training01 hive-0.13.1-bin]$ bin/hive -f /opt/datas/hivetest.sql
Logging initialized using configuration in file:/opt/modules/hive-0.13.1-bin/conf/hive-default.properties
OK
database_name
default
student
Time taken: 0.557 seconds, Fetched: 2 row(s)
OK
Time taken: 0.028 seconds
OK
tab_name
stu_info
Time taken: 0.038 seconds, Fetched: 1 row(s)
[hpsk@bigdata-training01 hive-0.13.1-bin]$
```

第一个ok是show databases;

第二个ok是use student;

第三个ok是show tables;

同理, 如果想将结果存到文件中, 指令如下

bin/hive -f /opt/datas/hivetest.sql >> /opt/datas/hive.exec.log

6.3.2 Hive常用交互指令

◆ 启动hive时通过命令行，临时修改配置参数

`-hiveconf <property=value>` Use value for given property

- 例： `bin/hive -hiveconf hive.cli.print.header=false`
- 临时修改配置参数，重启hive后就失效

6.3.2 Hive常用交互指令

◆ 退出hive

命令：quit、exit、ctrl+C

◆ “!”用于用于在hive shell中执行Linux命令

进入hive shell，输入：!ls /;

```
boot
cmd2
data
dev
etc
home
hosts
lib
lib64
lost+found
media
misc
mnt
net
```

ls为linux shell的查看指定目录下所有文件的命令，再按一次分号回车返回
hive shell

6.3.2 Hive常用交互指令

- 例：在hive shell 下查看某个文件的内容

```
hive (default)> !cat /opt/modules/hive-0.13.1-bin/student.txt;  
hive (default)> 1001      zhangsan  
1002      lisi  
1003      wangwu  
1004      zhaoliu  
█
```

- 这种方法可以避免hive shell 和linux shell来回切换的不便

- 例：在hive shell中，查看在 hive 中输入的所有历史命令

在hive shell中，刚刚输入不久的命令按下键盘上箭头就可以显示出来，那如何知道以前输入过的历史命令呢？

hive将所有历史命令存储在当前用户的根目录下的.hivehistory文件中

在hive shell下输入：!cat ~/.hivehistory

```
create table aa(id int);
show tables;
create table aa(id int);
show tables;
show databases;
insert into aa values(1);
select * from aa;
show tables;
show databases;
show tables;
create table test(id string);
show tables;
insert into test values('1001');
select * from test;
show tables;
create table test(id string);
select * from test;
show tables;
```


6.3.2 Hive常用交互指令

◆ dfs命令

用于在hive shell中执行hdfs的命令

- 例: `dfs -ls /;`

```
hive (default)> dfs -ls /;
Found 2 items
drwxrwx---   - hpsk supergroup          0 2017-05-23 15:09 /tmp
drwxr-xr-x   - hpsk supergroup          0 2017-05-23 11:19 /user
hive (default)> █
```

- `dfs -ls /user/hive/warehouse;`

```
hive (default)> dfs -ls /user/hive/warehouse;
Found 2 items
drwxr-xr-x   - hpsk supergroup          0 2017-05-23 15:30 /user/hive/warehouse/stu_tmp
drwxr-xr-x   - hpsk supergroup          0 2017-05-23 15:07 /user/hive/warehouse/student.db
hive (default)> █
```

- 在hive shell里执行hdfs命令比在linux shell下执行的速度快很多

原因: 在hive中已经一直与hdfs保持着持续的通信联系

6.3.3 HIVE中常见的数据类型

HIVE中常见的数据类型，类似于常见的的编程用开发软件和常见的数据库

Hive数据类型	长度	例子
TINYINT	1byte有符号整数	20
SMALINT	2byte有符号整数	20
INT	4byte有符号整数	20
BIGINT	8byte有符号整数	20
BOOLEAN	布尔类型，true 或者false	TRUE FALSE
FLOAT	单精度浮点数	3.14159
DOUBLE	双精度浮点数	3.14159
STRING	字符系列	可以使用单引号或者双引号 ‘ now is the time ’“for all good men”
TIMESTAMP	时间类型	

6.3.3 HIVE中常见的数据类型

HIVE中复杂的数据类型，简单了解下

数据类型	描述	语法示例
STRUCT	和 c 语言中的 struct 类似，都可以通过“点”符号访问元素内容。例如，如果某个列的数据类型是 STRUCT(first STRING, last STRING),那么第 1 个元素可以通过字段.first 来引用。	struct() 例如 struct<street:string, city:string>
MAP	MAP 是一组键-值对元组集合，使用数组表示法可以访问数据。例如，如果某个列的数据类型是 MAP，其中键->值对是' first' ->' John' 和' last' ->' Doe' ，那么可以通过字段名['last']获取最后一个元素	map() 例如 map<string, int>
ARRAY	数组是一组具有相同类型和名称的变量的集合。这些变量称为数组的元素，每个数组元素都有一个编号，编号从零开始。例如，数组值为['John' , 'Doe']，那么第 2 个元素可以通过数组名[1]进行引用。	Array() 例如 array<string>

6.3.3 HIVE中常见的数据类型

◆ 部分数据类型可以隐式转换

- Hive 的数据类型是可以进行隐式转换的，比如HIVE会根据语句表达式，自动地将TINYINT 会自动转换为 INT 类型、FLOAT 和 某些STRING 类型都可以隐式地转换成 DOUBLE类型等
- 如果需要强制转换，可以使用cast语句，后面章节会介绍

总结

◆ Hive常用属性配置

用户配置文件

默认配置文件

默认配置文件模板

如何修改HIVE的配置配置信息

通过直接修改配置文件

通过启动hive是添加参数（临时）

在hive shell下通过set查看和修改（临时）

◆ Hive常用交互指令

-help 帮助

-database 进入指定数据库

-e 执行一条sql

-f 执行一个文件中的sql

-hiveconf 修改配置项

退出hive shell: quit、exit

在shell中用! 使用linux命令

使用dfs使用hdfs命令

◆ Hive常见的数据类型

◆ Struct使用

数据: test5.txt

1,zhou:30

2,yan:30

3,chen:20

4,li:80

建表:

```
hive> create table student_test(id INT, info struct<name:STRING, age:INT>)
```

```
> ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
```

```
> COLLECTION ITEMS TERMINATED BY ':';
```

‘FIELDS TERMINATED BY’: 字段与字段之间的分隔符 ‘COLLECTION ’ ‘ITEMS
TERMINATED BY’: 一个字段各个item的分隔符

Struct查询:

```
hive> select info.age from student_test;
```

Total MapReduce jobs = 1

Total MapReduce CPU Time Spent: 490 msec

OK

30

30

20

80

Time taken: 21.677 seconds

数据: test5.txt

1,zhou:30

2,yan:30

3,chen:20

4,li:80

建表:

```
hive> create table student_test(id INT, info struct<name:STRING, age:INT>)
```

```
> ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
```

```
> COLLECTION ITEMS TERMINATED BY ':';
```


◆ Array使用

test6.txt

034,1:2:3:4

035,5:6

036,7:8:9:10

建表:

```
hive> create table class_test(name string, student_id_list array<INT>)
```

```
> ROW FORMAT DELIMITED > FIELDS TERMINATED BY ','
```

```
> COLLECTION ITEMS TERMINATED BY ':';
```

test6.txt

034,1:2:3:4

035,5:6

036,7:8:9:10

建表:

```
hive> create table class_test(name string, student_id_list array<INT>)
```

```
> ROW FORMAT DELIMITED > FIELDS TERMINATED BY ','
```

```
> COLLECTION ITEMS TERMINATED BY ':';
```

Array查询

```
hive> select student_id_list[3] from class_test;
```

Total MapReduce jobs = 1

.....

Total MapReduce CPU Time Spent:

480 msec


OK

4

NULL

10

Time taken: 21.574 seconds



◆ Map的使用

test7.txt

1 job:80,team:60,person:70

2 job:60,team:80

3 job:90,team:70,person:100

建表

```
hive> create table employee(id string, perf map<string, int>)
```

```
> ROW FORMAT DELIMITED > FIELDS TERMINATED BY '\t'
```

```
> COLLECTION ITEMS TERMINATED BY ','
```

```
> MAP KEYS TERMINATED BY ':'; OK
```

Map查询

```
hive> select perf['person'] from employee;
```

Total MapReduce jobs = 1

.....

Total MapReduce

CPU Time Spent: 460 msec

OK

70

NULL

100

Time taken: 20.902 seconds

test7.txt

1 job:80,team:60,person:70

2 job:60,team:80

3 job:90,team:70,person:100

建表

```
hive> create table employee(id string, perf map<string, int>)
```

```
> ROW FORMAT DELIMITED > FIELDS TERMINATED BY '\t'
```

```
> COLLECTION ITEMS TERMINATED BY ','
```

```
> MAP KEYS TERMINATED BY ':'; OK
```