

Hive数据类型及表创建

高校大数据课程系列

ENTER

课程目标

Course objectives

掌握Hive的数据类型

掌握内部表和外部表区别

1

2

3

4

掌握Hive创建表
使用方法

掌握应用案例

本章任务

Task of this chapter

1

数据类型与表创建

2

数据类型与表创建案例

数据类型与表创建

大类	类型	描述	示例
数值类型	TINYINT	1字节 有符号整数, -128~127	1
	SMALLINT	2字节 有符号整数, -32,768 ~32,767	1
	INT/INTEGER	4字节 有符号整数, -2,147,483,648 ~ 2,147,483,647	1
	BIGINT	8字节 有符号整数, -9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807	1
	FLOAT	4字节 单精度浮点数	1.0
	DOUBLE	8字节 双精度浮点数	1.0
	DECIMAL	在Hive 0.11.0中引入,精度为38位 Hive 0.13.0引入了用户可定义的精度和比例	DECIMAL(38,18)
	DOUBLE PRECISION	仅从Hive2.2.0开始有效	
	NUMERIC	从Hive3.0.0开始有效	
字符串类型	STRING	字符串	'a', 'a'
	VARCHAR	仅从Hive0.12.0开始有效	
	CHAR	仅从Hive0.13.0开始有并行	
Misc类型	BOOLEAN	true/false	TRUE
	BINARY	仅从Hive0.8.0开始有效	
日期/时间类型	TIMESTAMP	精度到纳秒的时间戳, 仅从Hive0.8.0开始有效	132550247050
	DATE	仅从Hive0.12.0开始有效	
	INTERVAL	仅从Hive1.2.0开始有效	

基本数据类型:

- Hive数据类型实现了Java的接口, 它支持类似Java中INT和FLOAT这样多种不同长度的类型, 也支持String这样无长度限制的类型。
- Hive基本数据类型主要有数值类型 (Numeric Types)、日期/时间类型、字符串类型和布尔型四种, 每种数据类型又分成许多细节的类型, 供Hive使用。具体内容如左表所示。

数据类型与表创建

类型	描述	示例
ARRAY<data_type>	从Hive 0.14开始，一组有序字段，字段的类型必须相同	array(1, 2)
MAP<primitive_type, data_type>	从Hive 0.14开始，一组无需的键值对，键的类型必须是原子的，值可以是任何类型。同一个映射的键的类型必须相同，值的类型也必须相同。	map('a' , 1, ' b' , 2)
STRUCT<col_name : data_type [COMMENT col_comment], ...>	一组命名的字段，字段的类型可以不同	struct('a' , 1, 1, 0)
UNIONTYPE<data_type, data_type, ...>	只能从Hive 0.7.0开始，支持符合数据类型（data_type）多个值的列举存储格式。	create_union(1, ' a' , '20')

复杂数据类型：

- Hive数据类型实现了Java的接口，它支持类似Java中INT和FLOAT这样多种不同长度的类型，也支持String这样无长度限制的类型。
- 复杂类型一共分为：数组类型（array_type）、映射类型（map_type）、结构类型（struct_type）和联合类型（union_type）四大类。

数据类型与表创建

ARRAY：您可以声明一个ITEMS数组来保存字符串值，如下所示：

```
ITEMS ARRAY<"Bread" , "Butter" , "Organic Eggs">
```

因为这个字符串集合有一个定义好的顺序或序列，所以可以通过一个零索引来访问这些字符串。

```
ITEMS[0] returns "Bread"
```

```
ITEMS[2] return "Organic Eggs"
```

Map：你可以声明一个Map集合，其中包含以下物品及其数量，如下所示：

```
Basket MAP<'string','int'>
```

```
Basket MAP<"Eggs",'12'>
```

复杂数据类型

您可以通过在Map函数中指定数量的对应项来打印数量值

```
Basket("Eggs") returns 12.
```

数据类型与表创建

Struct: 您可以使用以下结构定义来声明客户的地址记录:

```
address STRUCT <housenoo:STRING, street:STRING, city:STRING, zipcode:INT, state:STRING, country:STRING>
```

可以使用点符号访问结构的字段。邮政编码可以使用`address.zipcode`访问各种地址。

```
address <"17","MAIN ST", "SEATTLE", 98104, "WA","USA">
```

复杂数据类型

Union: 如果客户的联系信息存在于数据文件中, 但它们由多个电话号码或多个电子邮件地址, 您可以声明一个要存储的联系人变量信息如下所示:

```
contact UNIONTYPE <Int,array<int>, string, array<string>>
```

数据类型与表创建

您可以使用create database命令在Hive中创建一个数据库， CREATE DATABASE命令的完整语法是

```
CREATE (DATABASE | SCHEMA) [ IF NOT EXISTS ] database_name [ COMMENT database_comment ] [
LOCATION hdfs_path ] [ WITH DBPROPERTIES (property_name = property_value,...) ] ;
```

下面是一个使用完整语法的示例

创建数据库

```
CREATE DATABASE IF NOT EXISTS shopping COMMENT 'stores all shopping basket data' LOCATION
'/user/retail/hive/SHOPPING.db' WITH DBPROPERTIES ('purpose' = 'testing') ;
```

该命令将创建一个名为shopping的数据库并指定数据库的位置为/user/retail/hive/ shopping.db。使用WITH DBPROPERTIES子句，可以将任何自定义属性分配给数据库。

数据类型与表创建

- 创建数据库之后，可以使用以下命令修改其元数据属性(DBPROPERTIES)或所有者
- ALTER DATABASE命令如下:

```
ALTER DATABASE shopping SET DBPROPERTIES ('department' = 'SALES');
```

您可以使用drop database命令来删除一个Hive数据库。

```
DROP DATABASE database_name [RESTRICT|CASCADE];
```

例如

修改与删除数据库

```
DROP DATABASE shopping CASCADE;
```

数据类型与表创建

创建表语法格式如下

```
CREATE [TEMPORARY] [EXTERNAL] TABLE [IF NOT EXISTS] [db_name.]table_name  -- (Note:
TEMPORARY available in Hive 0.14.0 and later)
[(col_name data_type [COMMENT col_comment], ... [constraint_specification])]
[COMMENT table_comment]
[PARTITIONED BY (col_name data_type [COMMENT col_comment], ...)]
[CLUSTERED BY (col_name, col_name, ...) [SORTED BY (col_name [ASC|DESC], ...) ] INTO num_buckets
BUCKETS]
[SKEWED BY (col_name, col_name, ...) -- (Note: Available in Hive 0.10.0 and later)]
  ON ((col_value, col_value, ...), (col_value, col_value, ...), ...)
[STORED AS DIRECTORIES]
[
  [ROW FORMAT row_format]
  [STORED AS file_format]
  | STORED BY 'storage.handler.class.name' [WITH SERDEPROPERTIES (...)] -- (Note: Available in Hive 0.6.0
and later)
]
[LOCATION hdfs_path]
[TBLPROPERTIES (property_name=property_value, ...)] -- (Note: Available in Hive 0.6.0 and later)
[AS select_statement];  -- (Note: Available in Hive 0.5.0 and later; not supported for external tables)
```

数据类型与表创建

内部表与关系数据库中的Table在概念上类似。每一个Table在Hive中都有一个相应的目录存储数据。所有的Table数据（不包括External Table）都保存在这个目录中。删除表时，无数据与数据都会被删除。

- 元数据库中查询数据列表



```
1 • select * from tbls where db_id='116';
```

	TBL_ID	CREATE_TIME	DB_ID	LAST	OWNER	RENTI	SD_ID	TBL_NAME	TBL_TYPE
▶	263	1427482286	116	0	hdfs	0	424	person_bucket	MANAGED_TABLE
	267	1427552209	116	0	hdfs	0	432	person_inside	MANAGED_TABLE
	268	1427552417	116	0	hdfs	0	433	person_part	MANAGED_TABLE
	269	1427552758	116	0	hdfs	0	435	person_ext	EXTERNAL_TABLE
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

- HDFS对应存储目录

```
[root@ambari-agent-251 hivedb]# hadoop fs -ls /apps/hive/warehouse/liuhivedb.db/
Found 3 items
drwxrwxrwx - hdfs hdfs      0 2015-03-28 02:51 /apps/hive/warehouse/liuhivedb.db/person_bucket
drwxrwxrwx - hdfs hdfs      0 2015-03-28 22:17 /apps/hive/warehouse/liuhivedb.db/person_inside
drwxrwxrwx - hdfs hdfs      0 2015-03-28 22:21 /apps/hive/warehouse/liuhivedb.db/person_part
```

数据类型与表创建

外部表

外部表指向已经在HDFS中存在的数据，可以创建Partition。包含内部表在无数据的组织上是相同的，而实际数据的存储则较大的差异。内部表的创建过程和数据加载过程这两个过程可以分别独立完成，也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。而外部表只有一个过程。加载数据和创建表同时完成。实际数据是存储在LOCATION后面指定的 HDFS路径中，并不会移动到数据仓库目录中。当删除一个外部表时，只删除一个链接。

本章任务

Task of this chapter

1

数据类型与表创建

2

数据类型与表创建案例

任务 数据类型与表创建案例

任务背景

Hive是一个基于Apache Hadoop的数据仓库基础架构。Hive建表的语法与MySQL类似，需要指定表名、表中的字段名及字段对应的数据类型。所以在学习Hive建立表前，明确下数据类型、关键字的问题显得很必要。

同时，Hive与传统数据库相比，格式较为宽松，它在建立表时，可以由用户指定表的字段间隔符及换行符，以及存储位置等，形式较自由。由于Hive的存储是建立在HDFS基础之上的。较复杂的HiveQL也会解释成MapReduce（Hadoop分布式计算框架）后计算大数据集的统计结果。

数据类型，它能够很好地与Hadoop平台（框架源码由Java开发）融合，HiveQL数据类型都是对Java中接口的实现。大体上讲，Hive支持的数据类型（data_type）可分为基本数据类型（primitive_type）和复杂类型（Complex Types）二种。

任务 数据类型与表创建案例

任务需求

- 1). 创建用户表1（包含姓名和薪水）并导入数据
- 2). 创建用户表2（包含姓名、薪水、家庭成员、税金、住址）表并导入数据
- 3). 查看用户表1中姓名和薪水数据
- 4). 查看用户表2中家庭成员、税金、住址数据

参考

Hive数据类型-实验
手册

任务 数据类型与表创建案例

任务分析

查看本地文件file1数据内容及格式（包含用户名和薪水），查看本地文件file2数据内容及格式（包含用户名、薪水、家庭成员、税金、住址）。启动Hadoop服务，启动Hive服务进入到Hive命令行客户端。创建表t_user1（包含基本数据类型用户名和薪水）。把file1的数据导入表t_user1。查看表t_user1数据内容及格式。创建表t_user2（包含基本数据类型用户名和薪水，复杂数据类型家庭、税金和住址），导入file2数据进入到t_user2，查看表t_user2中的家庭、税金和住址数据。

任务 数据类型与表创建案例

任务步骤

- 1、启动Hadoop服务
- 2、启动Hive服务
- 3、创建表t_user1(包含用户名和薪水)并导入数据
- 4、创建表t_user2(包含用户名和薪水、家庭、税金和住址)并导入数据
- 5、查看表t_user1和t_user2数据内容及格式

任务 数据类型与表创建案例

任务结果

Hive命令行客户端正常启动，成功执行相应Hive语句。

其他任务参考实验

任务列表

- Hive数据类型-实验手册
- Hive创建表及数据导入-实验手册

谢谢观看

THANKS FOR WATCHING