Hive数据操作及分区分桶

高校大数据课程系列

ENTER

课程目标

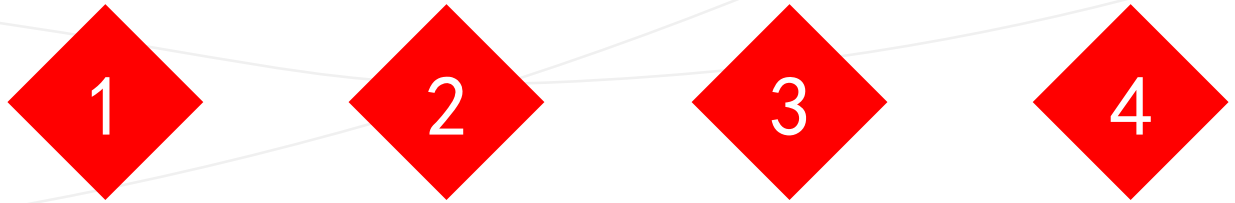
Course objectives

掌握Hive的数据操作

掌握Hive分区
分桶方法

掌握数据操作案例

掌握分区分桶
案例



本章任务

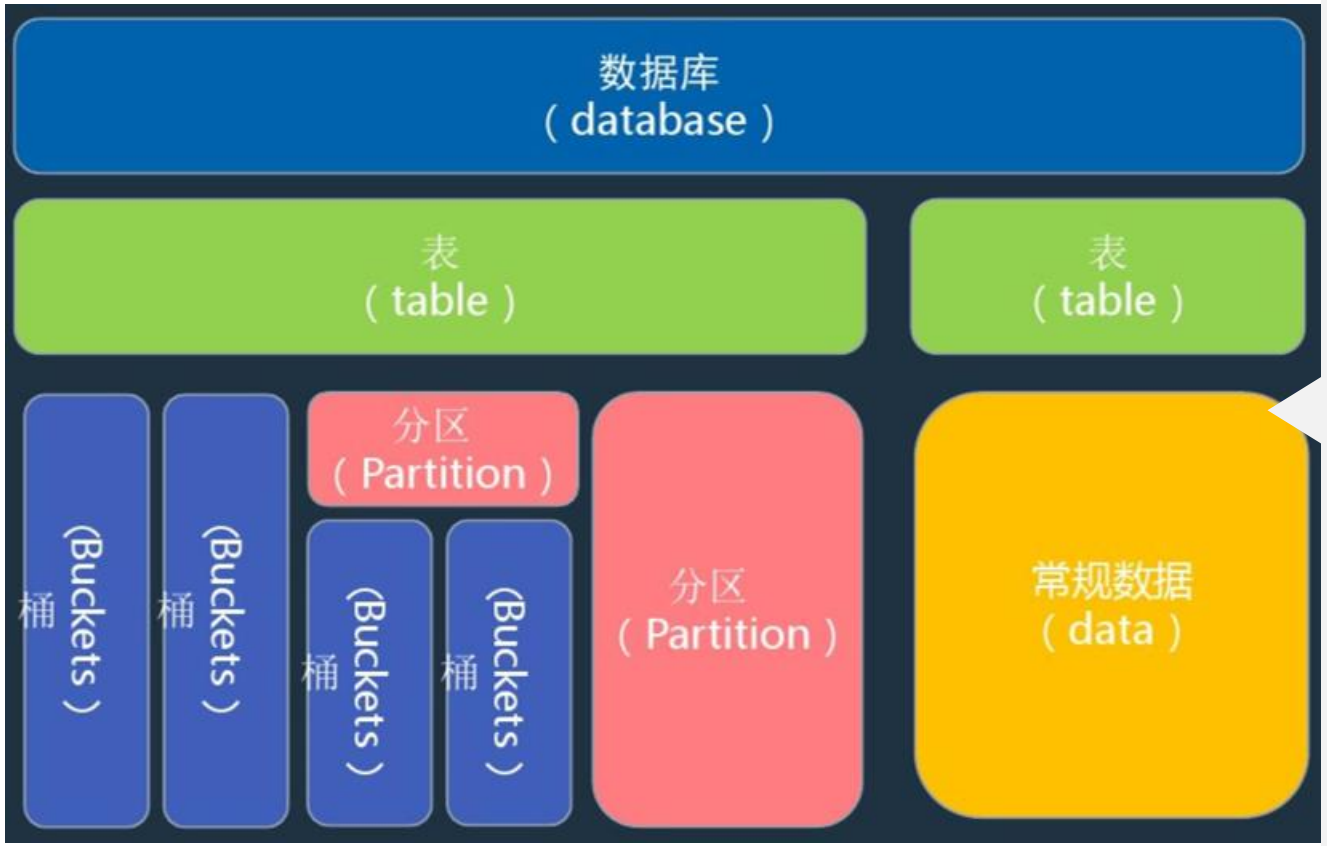
Task of this chapter

1 数据操作与分区分桶

2 数据操作案例

3 分区分桶案例

数据操作与分区分桶



数据存储模型：

- 分区按指定的格式在表下面分出若干个（有限的）文件夹，把相应的文件分到指定的文件夹下，达到从粗粒度上对表数据的划分，以此加快数据的查找速度。
- Hive提供了把表(或分区)组织成桶(bucket)的功能，它默认采用的HashPartition分区，能够满足把数据近似均匀地分配到不同的桶里。

数据操作与分区分桶

分区操作

创建人员信息表person_part，列以逗号“，”分隔。 建立city为分区。

```
create table person_part
(id string,name string,sex string,age int)
partitioned by(city string)
row format delimited fields terminated by ‘，’
stored as textfile;
--加载数据：本地数据位置： / tmp/person.txt
 load data local inpath ‘file:///tmp/person.txt person_part’ into table person_part  partition(city=‘beijing’);
```

数据操作与分区分桶

Hive分区示例：

- 数据存储在city=“jinan“目录下

```
[root@ambari-agent-251 hivedb]# hadoop fs -ls /apps/hive/warehouse/liuhivedb.db/person_part/city=jinan
Found 1 items
-rwxrwxrwx   3 hdfs hdfs        280 2015-03-28 21:55 /apps/hive/warehouse/liuhivedb.db/person_part/city=jinan/person.txt
```

- 根据分区查询：hive会自动判断where语句中是否包含分区的字段。 而且可以使用大于小于等运算符

```
0: jdbc:hive2://127.0.0.1:10000> select * from person_part where city='jinan';
+-------------------+---------------------+--------------------+--------------------+---------------------+--+
| person_part.id    | person_part.name    | person_part.sex    | person_part.age    | person_part.city    |
+-------------------+---------------------+--------------------+--------------------+---------------------+--+
| 0                 | 汤姆                  | man                | 10                 | jinan               |
| 1                 | 麦克                  | man                | 12                 | jinan               |
| 2                 | 露丝                  | woman              | 9                  | jinan               |
| 3                 | tom                 | man                | 8                  | jinan               |
| 4                 | mike                | man                | 7                  | jinan               |
| 5                 | rose                | woman              | 11                 | jinan               |
| 6                 | l汤姆                 | man                | 12                 | jinan               |
| 7                 | l麦克                 | man                | 14                 | jinan               |
| 8                 | l露丝                 | woman              | 6                  | jinan               |
| 9                 | ltom                | man                | 17                 | jinan               |
| 10                | lmike               | man                | 18                 | jinan               |
| 11                | lrose               | woman              | 19                 | jinan               |
+-------------------+---------------------+--------------------+--------------------+---------------------+--+
12 rows selected (0.101 seconds)
```

数据操作与分区分桶

分桶操作

创建人员信息表person_bucket，列以逗号分隔。 年龄字段上建立5个桶。

```
create table person_bucket (id string,name string, sex string,age int) partitioned by (city string)
clustered by (age) sorted by (name) into 5 buckets
row format delimited fields terminated by ‘,’ stored as textfile;
--打开桶参数：
set hive. enforce. bucketing＝ true;
--加载数据：
 insert into table person_bucket partition (city＝ ‘Jinan ’） select * from person_inside;
```

数据操作与分区分桶

Hive分桶示例：

- 数据加载到桶表时，会对字段取Hash值，然后与桶的数量取模，把数据放到对应的目录下

```
[root@ambari-agent-251 hivedb]# hadoop fs -ls /apps/hive/warehouse/liuhivedb.db/person_bucket/city=jinan
Found 5 items
-rwxrwxrwx   3 hdfs hdfs         16 2015-03-28 02:51 /apps/hive/warehouse/liuhivedb.db/person_bucket/city=jinan/000000_0
-rwxrwxrwx   3 hdfs hdfs         34 2015-03-28 02:51 /apps/hive/warehouse/liuhivedb.db/person_bucket/city=jinan/000001_0
-rwxrwxrwx   3 hdfs hdfs         60 2015-03-28 02:51 /apps/hive/warehouse/liuhivedb.db/person_bucket/city=jinan/000002_0
-rwxrwxrwx   3 hdfs hdfs         28 2015-03-28 02:51 /apps/hive/warehouse/liuhivedb.db/person_bucket/city=jinan/000003_0
-rwxrwxrwx   3 hdfs hdfs         52 2015-03-28 02:51 /apps/hive/warehouse/liuhivedb.db/person_bucket/city=jinan/000004_0
```

- 抽样查询：查询5个桶中的第2个桶，即000001_0文件

```
0: jdbc:hive2://127.0.0.1:10000> select * from person_bucket tablesample(bucket 2 out of 5 on age);
+--------------------+----------------------+---------------------+---------------------+----------------------+--+
| person_bucket.id   | person_bucket.name   | person_bucket.sex   | person_bucket.age   | person_bucket.city   |
+--------------------+----------------------+---------------------+---------------------+----------------------+--+
| 8                  | 1聂丝                  | woman               | 6                   | jinan                |
| 5                  | rose                 | woman               | 11                  | jinan                |
+--------------------+----------------------+---------------------+---------------------+----------------------+--+
2 rows selected (0.109 seconds)
```

```
[root@ambari-agent-251 hivedb]# hadoop fs -cat /apps/hive/warehouse/liuhivedb.db/person_bucket/city=jinan/000001_0
8,1聂丝,woman,6
5,rose,woman,11
```

本章任务

Task of this chapter

1 数据操作与分区分桶

2 数据操作案例

3 分区分桶案例

任务1 数据操作案例

任务背景

Hive数据操纵语言（Data Manipulation Language, DML）是SQL语言中，负责对数据库对象运行数据访问工作的指令集，以INSERT、UPDATE、DELETE三种指令为核心，分别代表插入、更新与删除，是开发以数据为中心的应用程序必定会使用到的指令。Hive DML主要指表数据的加载、插入、更新、删除和合并。

任务1 数据操作案例

任务需求

1).创建表userinfo（包含基本数据类型姓名、薪水和复杂数据类型家庭成员、税金、住址）

2).将本地数据文件userinfo1.txt和userinfo2.txt数据内容导入表userinfo中

3).使用HDFS命令方式删除由userinfo2.txt导入的数据

4).查询表userinfo中的Array、Map、Struct结构的数据

5).使用函数及带条件的查询表userinfo中的数据

任务1 数据操作案例

任务分析

启动Hadoop服务，查看安全模式的状态。启动Hive服务，进入Hive命令行客户端。创建表userinfo（包含基本数据类型姓名、薪水和复杂数据类型家庭成员、税金、住址），将本地数据文件userinfo1.txt和userinfo2.txt数据内容导入表userinfo。本地数据文件userinfo1.txt和userinfo2.txt数据内容导入表userinfo。使用Hive命令查询表userinfo数据。

任务1 数据操作案例

任务步骤

1、启动Hadoop服务

2、启动Hive服务

3、创建表userinfo（包含基本数据类型姓名、薪水和复杂数据类型家庭成员、税金、住址）

4、本地数据文件userinfo1.txt和userinfo2.txt数据内容导入表userinfo

5、查询表userinfo中的数据

任务1 数据操作案例

任务结果

Hive命令行客户端正常启动，成功执行相应Hive语句。

本章任务

Task of this chapter

1 数据操作与分区分桶

2 数据操作案例

3 分区分桶案例

任务2 分区分桶案例

任务背景

Hive分区按指定的格式在表下面分出若干个（有限的）文件夹，把相应的文件分到指定的文件夹下，达到从粗粒度上对表数据的划分，以此加快数据的查找速度。但此种方法对于一些细粒度的划分，或者数据均匀分配上并不擅长。例如按userid进行划分时，会产生众多分区，从而非常容易产生众多的小文件。Hive里提供了把表(或分区)组织成桶(bucket)的功能，它默认采用的HashPartition分区，能够满足把数据近似均匀地分配到不同的桶里。具体讲分桶有如下的**好处**：

获得更高的查询处理效率。桶为表加上了额外的结构。Hive 在处理有些查询时能够利用这个结构。具体而言，连接两个在(包含连接列的)相同列上划分了桶的表，可以使用map端连接(map-side join) 高效地实现。

使"取样" (sampling)更高效。在处理大规模数据集时，在开发和修改查询的阶段，如果能在数据集的一小部分数据上试运行查询，会带来很多方便。

任务2 分区分桶案例

任务需求

1).创建表p_b_stocks（包含股票代码、股票交易日期、股票开盘价、股票最高价、股票最低价、股票收盘价、股票交易量和股票成交价）

2).将本地数据文件stocks.csv数据内容导入表p_b_stock

3).创建分区表 p_stocks（按股票代码分区），并将表p_b_stocks导入

4).创建分桶表 b_stocks（按股票代码分成3个桶），并将表p_b_stocks导入

5).查看 p_stocks和 b_stocks的结构和数据

任务2 分区分桶案例

任务分析

启动Hadoop服务，查看安全模式的状态。启动Hive服务，进入Hive命令行客户端。创建表p_b_stocks（包含股票代码、股票交易日期、股票开盘价、股票最高价、股票最低价、股票收盘价、股票交易量和股票成交价），将本地数据文件stocks.csv数据内容导入表p_b_stocks。创建分区表 p_stocks（按股票代码分区），并将表p_b_stocks导入。创建分桶表 b_stocks（按股票代码分成3个桶），并将表p_b_stocks导入。分别查看表 p_stocks和 b_stocks的结构和数据。退出Hive环境，停止Hadoop服务。

任务2 分区分桶案例

任务步骤

1、启动Hadoop服务，启动Hive服务

2、创建表p_b_stocks（包含股票代码、股票交易日期、股票开盘价、股票最高价、股票最低价、股票收盘价、股票交易量和股票成交价）

3、创建分区表 p_stocks（按股票代码分区），并将表p_b_stocks导入

4、创建分桶表 b_stocks（按股票代码分成3个桶），并将表p_b_stocks导入

5、查看 p_stocks和 b_stocks的结构和数据并停止Hadoop和Hive服务

任务2 分区分桶案例

任务结果

Hive命令行客户端正常启动，成功执行相应Hive语句。

谢谢观看

THANKS FOR WATCHING