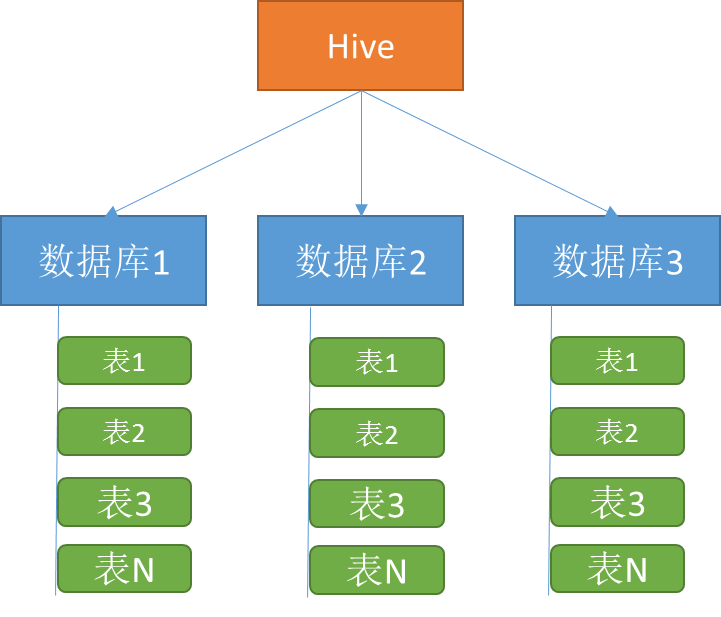
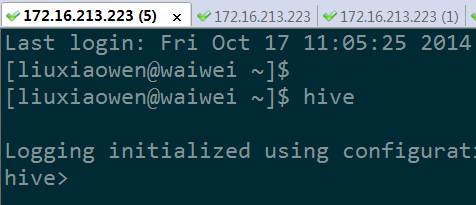
# 一、基础篇

## Hive的层次结构：

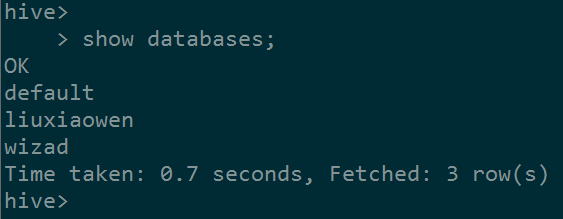


## 进入Hive

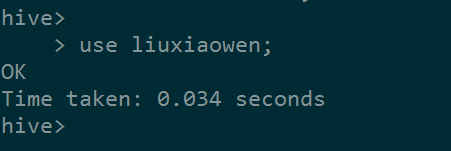
* 用自己的用户登陆网关机(172.16.213.223),运行命令**hive**,进入Hive Cli命令行：



* 在Hive命令行运行 **show databases;** 查看所有的数据库

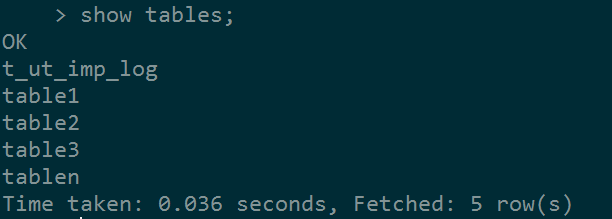


* 在Hive命令行运行 **use database\_name;** 切换到自己业务的数据库；



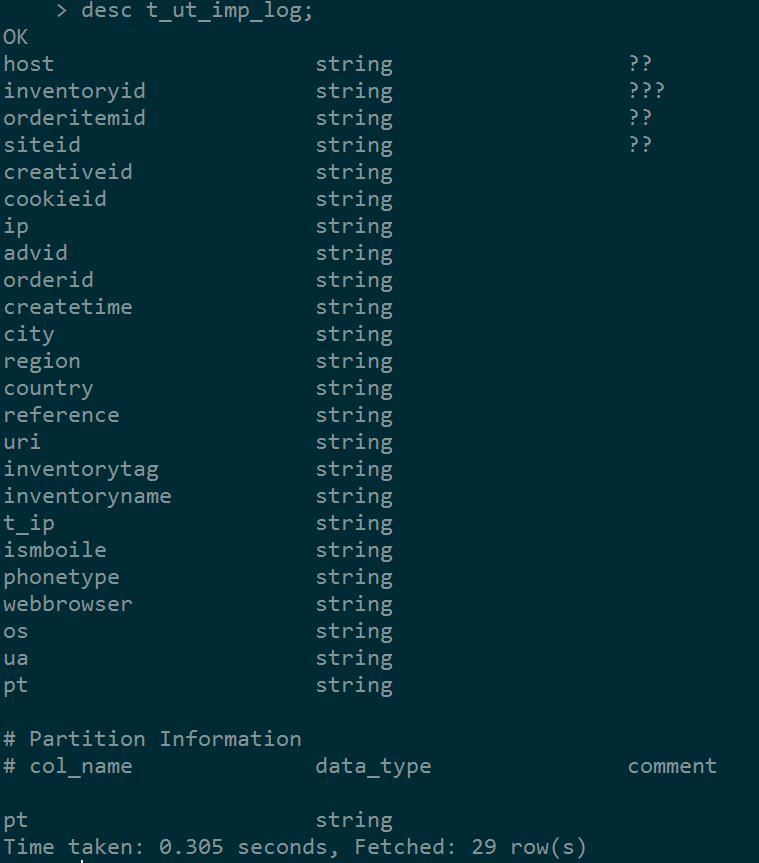
## 查看所有表

* 进入自己业务数据库后，在Hive命令行运行 **show tables;** 查看自己数据库中的所有表

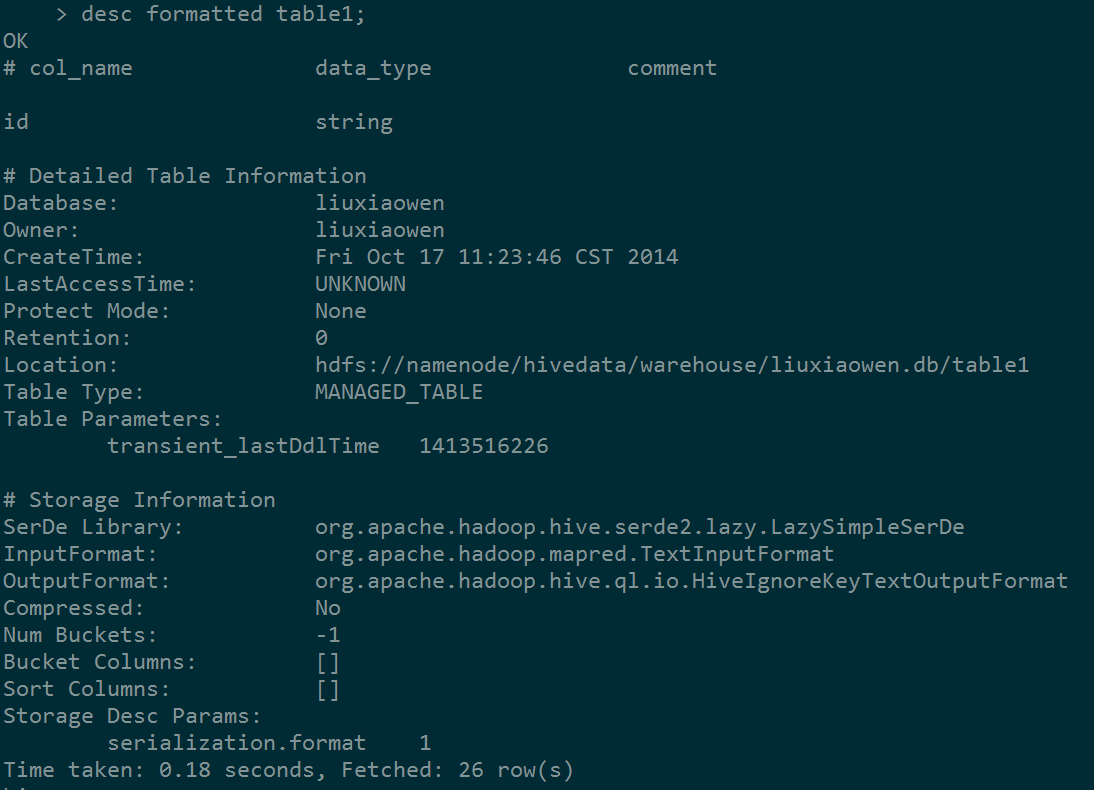


## 查看表结构

* **desc table\_name;** 查看表的基础信息(字段、分区)



* **desc formatted table\_name;** 查看表的详细信息(字段、分区、路径、格式等)



## 基本数据类型

整型：

TINYINT、SMALLINT、**INT**、**BIGINT**

布尔型：

BOOLEAN

浮点型：

FLOAT、DOUBLE

字符型：

**STRING**

[**https://cwiki.apache.org/confluence/display/Hive/Tutorial#Tutorial-TypeSystem**](https://cwiki.apache.org/confluence/display/Hive/Tutorial#Tutorial-TypeSystem)

# 表



## 内部表和外部表的区别

内部表删除时候**会**删除Hadoop上的数据;

外部表删除时候**不会**删除Hadoop上的数据;

## 内部表

* 适用场景：中间表，结果表，不需要从外部（如本地文件、HDFS上load数据）
* 创建语法：

**CREATE TABLE** tab (

column1 STRING,

column2 STRING,

column3 STRING,

column4 STRING,

columnN STRING

);

## 外部表

* 适用场景：源表，需要将外部数据映射到表中
* 创建语法：

**CREATE EXTERNAL TABLE** tab (

column1 STRING,

column2 STRING,

column3 STRING,

column4 STRING,

columnN STRING

) ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\t'

stored as textfile

**location** 'hdfs://namenode/tmp/liuxiaowen/tab/';

## 分区

* 对表按照一定的规则进行切分，如时间，每天（每小时）一个分区；
* 适用场景：数据量非常大，按照时间或其他属性周期性维护和查询；
* 创建分区语法：

CREATE TABLE tab (

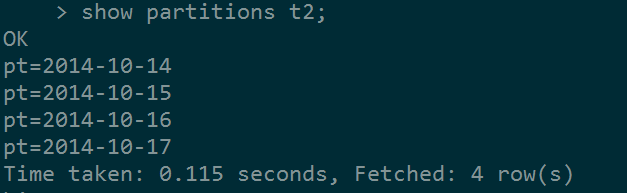
column1 STRING,

columnN STRING

) **partitioned BY (day STRING);**

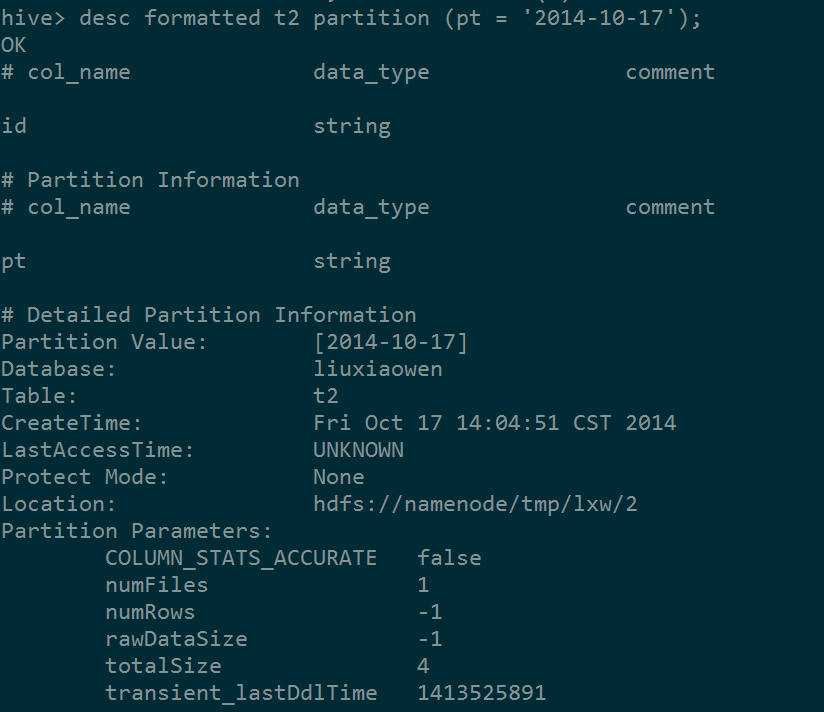
* 内部表和外部表都可以创建分区；
* 分区表必须在创建时候指定分区，对于一个已经创建好的非分区表，不能将其修改为分区表；
* 一个表可以有多个分区字段；
* 查看表有哪些分区：

**show partitions** table\_name;



* 查看分区属性：

**desc formatted** t2 **partition** (pt = '2014-10-17');



* 添加分区及数据：

ALTER TABLE tab ADD PARTITION (day = '2014-10-16')

location 'hdfs://namenode/tmp/liuxiaowen/tab/day=2014-10-16/';

ALTER TABLE tab ADD PARTITION (day = '2014-10-17')

location 'hdfs://namenode/tmp/liuxiaowen/tab/day=2014-10-17/';

* 删除一个分区：

ALTER TABLE tab DROP PARTITION (day = '2014-10-16');

## 修改表

* 重命名

ALTER TABLE old\_table\_name RENAME TO new\_table\_name;

* 添加字段

ALTER TABLE tab1 ADD COLUMNS (c1 INT, c2 STRING);

## 删除表

* DROP TABLE tab;

# 三、加载数据

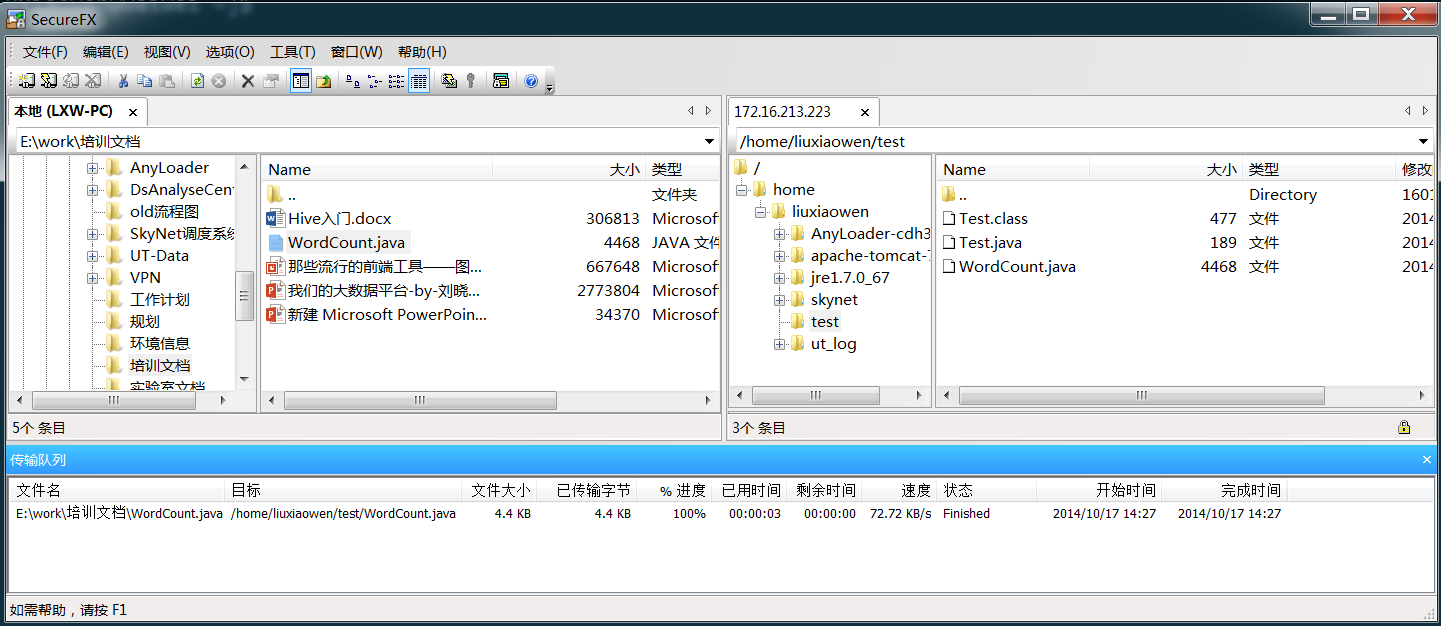
## 3.1 将本地（windows）上的文件上传至网关机（linux）

以linux终端工具SecureCRT为例，登陆网关机，点击SecureFX按钮：



弹出文件传输框，左侧为本地，右侧为网关机；

选择左侧和右侧目录，将左侧本地文件拖至右侧区域即可完成文件上传。



## 3.2 将网关机本地的文件上传至Hadoop

**hadoop fs -put** **/tmp/f.txt** **/user/data/staging/page\_view/**

## 3.3 加载本地数据到外部表

* 创建时候指定location

CREATE EXTERNAL TABLE tab (

column1 STRING,

columnN STRING

) ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\t'

stored as textfile

location 'hdfs://namenode/tmp/liuxiaowen/tab/';

* 将网关机本地文件上传至指定的location

**hadoop fs -put** **/tmp/f.txt** **hdfs://namenode/tmp/liuxiaowen/tab/**

* 文件格式要和表定义的格式一致

## 3.4 加载本地数据到内部表（分区）

* 加载网关机本地文件到内部表

LOAD DATA **LOCAL** INPATH /tmp/f.txt **INTO** TABLE tab PARTITION (day = ‘2014-10-17’);

LOAD DATA **LOCAL** INPATH /tmp/f.txt **OVERWRITE** **INTO** TABLE tab PARTITION (day = ‘2014-10-17’);

**\*** INTO 为追加，OVERWRITE INTO 为覆盖

* 加载Hadoop文件到内部表

**LOAD DATA INPATH** '/tmp/liuxiaowen/tab/f.txt' **INTO** TABLE tab PARTITION (day = ‘2014-10-17’);

**LOAD DATA INPATH** '/tmp/liuxiaowen/tab/f.txt' **OVERWRITE INTO** TABLE tab PARTITION (day = ‘2014-10-17’);

**\*** LOAD DATA LOCAL INPATH 为网关机本地路径，LOAD DATA INPATH为Hadoop路径

* 文件格式要和表定义的格式一致

## 3.5 从一个查询结果集加载数据（INSERT）

INSERT **INTO** TABLE tab PARTITION (day = ‘2014-10-17’)

SELECT \* FROM source\_table;

INSERT **OVERWRITE** TABLE tab PARTITION (day = ‘2014-10-17’)

SELECT \* FROM source\_table;

* INTO 为追加，OVERWRITE为覆盖

# 四、下载数据

## 4.1 从Hadoop上下载文件到本地

**hadoop fs –get** **/hivedata/warehouse/liuxiaowen/1.txt** **/home/liuxiaowen/data/**

## 4.2 将一个查询结果集保存到本地文件

INSERT OVERWRITE LOCAL DIRECTORY '**/home/liuxiaowen/data/**'

ROW FORMAT DELIMITED FIELDS TERMINATED BY ','

SELECT \* FROM source\_table;

# 五、查询

## 5.1 简单查询

SELECT \*

FROM user

WHERE userid = ‘u1’;

SELECT u.userid,u.age,u.sex

FROM user u

WHERE userid = ‘u2’;

## 5.2 基于分区的查询

SELECT \*

FROM user

WHERE userid = ‘u1’ AND day = ‘2014-10-17’;

SELECT u.userid,u.age,u.sex

FROM user u

WHERE userid = ‘u2’ AND u.day >= ‘2014-10-16’ AND u.day <= ‘2014-10-17’

## 5.3 关联查询(JOIN)

用户表：

|  |  |  |
| --- | --- | --- |
| userid | age | sex |
| u1 | 29 | f |
| u2 | 25 | m |
| u3 | 24 | m |
| u4 | 30 | f |

PV表：

|  |  |  |
| --- | --- | --- |
| userid | pv | day |
| u1 | 13 | 2014-10-17 |
| u2 | 15 | 2014-10-17 |
| u1 | 15 | 2014-10-16 |
| u3 | 16 | 2014-10-16 |
| u4 | 20 | 2014-10-16 |

SELECT u.userid,u.age,u.sex,p.pv

FROM user u **JOIN** pv p

ON (u.userid = p.userid)

WHERE p.day = '2014-10-17';

|  |  |  |  |
| --- | --- | --- | --- |
| userid | age | sex | pv |
| u1 | 29 | f | 13 |
| u2 | 25 | m | 15 |

SELECT u.userid,u.age,u.sex,p.pv

FROM user u **LEFT OUTER JOIN** pv p

ON (u.userid = p.userid AND p.day = '2014-10-17');

|  |  |  |  |
| --- | --- | --- | --- |
| userid | age | sex | pv |
| u1 | 29 | f | 13 |
| u2 | 25 | m | 15 |
| u3 | 24 | m | NULL |
| u4 | 30 | f | NULL |

SELECT u.userid,u.age,u.sex,**IF(p.pv IS NULL,0,p.pv)**

FROM user u LEFT OUTER JOIN pv p

ON (u.userid = p.userid AND p.day = '2014-10-17');

|  |  |  |  |
| --- | --- | --- | --- |
| userid | age | sex | pv |
| u1 | 29 | f | 13 |
| u2 | 25 | m | 15 |
| u3 | 24 | m | 0 |
| u4 | 30 | f | 0 |

## 5.4 聚合汇总

* 查询用户一段时间内的总PV：

SELECT userid,

**SUM**(pv)

FROM pv

WHERE day >= '2014-10-16' AND day <= '2014-10-17'

**GROUP BY** userid;

---------------------------------------------

u1 28

u2 15

u3 16

u4 20

* 查询每一天的总PV

SELECT day,

SUM(pv)

FROM pv

GROUP BY day;

---------------------------------------------

2014-10-16 51

2014-10-17 28

<https://cwiki.apache.org/confluence/display/Hive/Tutorial#Tutorial-UsageandExamples>

## 5.5 基于函数的复杂查询

* 域名解析：

SELECT

a.t\_ip,

regexp\_extract(t\_ip,'//(.\*?)/',1),

parse\_url(t\_ip,'HOST')

FROM t\_ut\_imp\_log a

WHERE a.pt = '2014-09-01'

AND t\_ip LIKE 'http://%'

limit 10;

---------------------------------------------

http://v.admaster.com.cn/i/a21771,b200492904,c3527,i0,m202,h v.admaster.com.cn v.admaster.com.cn

http://imp.gentags.net/imp/iv-5175/st-47/oi-183866/cr-2/adv-9/or-3271/pcon-0/ imp.gentags.net imp.gentags.net

http://v.admaster.com.cn/i/a21771,b200492904,c3527,i0,m202,h v.admaster.com.cn v.admaster.com.cn

http://imp.gentags.net/imp/iv-4496/st-47/oi-183837/cr-2/adv-9/or-3271/pcon-0/ imp.gentags.net imp.gentags.net

* 访问路径、页面停留时间、是否新的SESSION

**SELECT**

cookieid,

SUM(1) OVER(PARTITION BY cookieid) AS pv,

ROW\_NUMBER() OVER(PARTITION BY cookieid ORDER BY createtime) AS depth,

createtime,

LAG(createtime) OVER(PARTITION BY cookieid ORDER BY createtime) AS last\_view\_time,

LEAD(createtime) OVER(PARTITION BY cookieid ORDER BY createtime) AS next\_view\_time,

uri,

FIRST\_VALUE(uri) OVER(PARTITION BY cookieid ORDER BY createtime) AS landing\_page

**FROM** t\_ut\_imp\_log

**WHERE** pt = '2014-09-01'

**AND** cookieid = '24D7776A00023753F1ACB9'

ORDER BY createtime;

---------------------------------------------

结果见前面的tab.xlsx



# 附录A

## A.1 Hive基础教程

<https://cwiki.apache.org/confluence/display/Hive/Tutorial>

## A.2 Hive操作和函数

<https://cwiki.apache.org/confluence/display/Hive/LanguageManual+UDF>

## A.3 Hive DDL语法

DDL：数据定义语言。

创建、修改、删除（数据库、表、分区、列、视图、索引等数据库对象）。

<https://cwiki.apache.org/confluence/display/Hive/LanguageManual+DDL>

## A.4 Hive查询语法

<https://cwiki.apache.org/confluence/display/Hive/LanguageManual+Select>

## A.5 Hive命令行使用说明

<https://cwiki.apache.org/confluence/display/Hive/LanguageManual+Cli>