在hive-default.xml中加入HADOOP\_HOME和HIVE\_CONF\_DIR

创建表：

Create table student(id int,name string)row format delimited fields terminated by ‘\t’;

装载数据

Load data inpath ‘/usr/local/……’ into table student;

查看表结构

Desc student

Desc formatted student

Desc extended student

查看hive支持的函数

Show functions

查看一个函数的说明

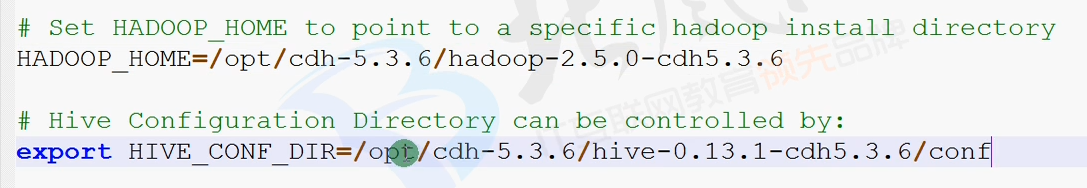
Desc function upper;

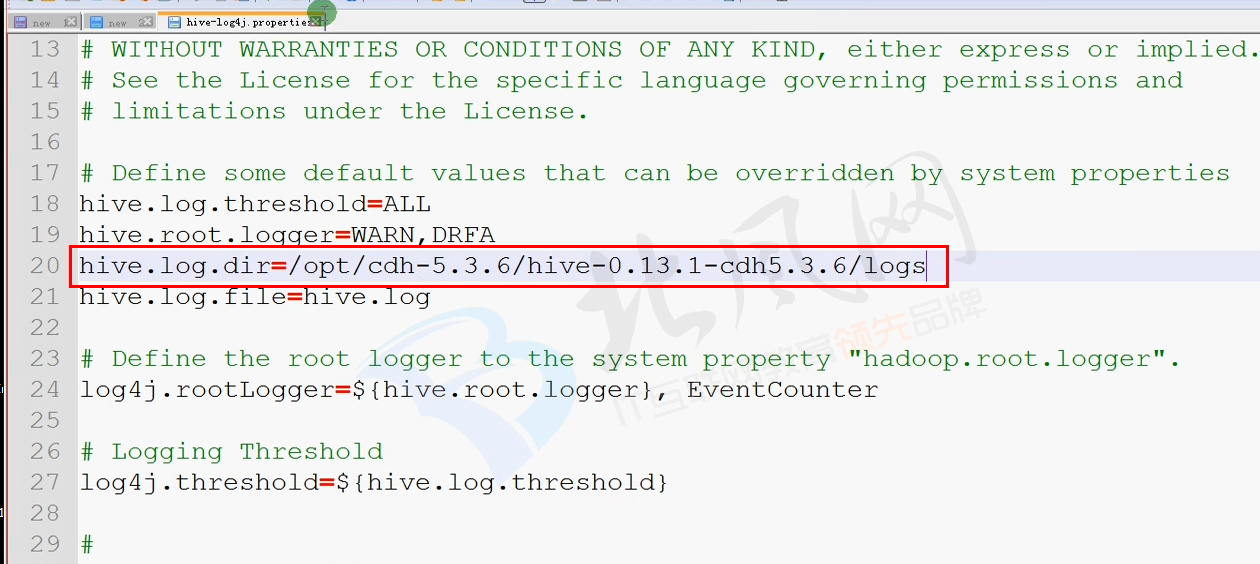
查看一个函数的说明，并带上使用例子

Desc function extended upper;

配置：

Hive-eng.sh 加入adoop的home和hive的配置文件





Touch hive-site.xml

具体见附件

两个常用配置，用于打开cli时候提示用户进入哪个数据库及其字段头的显示

<property>

<name>hive.cli.print.header</name>

<value>true</value>

<description>Whether to print the names of the columns in query output.</description>

</property>

<property>

<name>hive.cli.print.current.db</name>

<value>true</value>

<description>Whether to include the current database in the hive prompt.</description>

</property>

日志配置

# Define some default values that can be overridden by system properties

hive.root.logger=WARN,DRFA

hive.log.dir=/usr/local/hive/logs

hive.log.file=hive.log

日志级别

hive.root.logger=WARN,DRFA

查看当前所有的配置信息：在hive>set;

查看当前属性的值：set system:user.name;

设置当前属性的值：set system:user.name=momo;

此种方式设置的属性的值尽在当前会话有效。

在启动的时候设置参数

Hive -help

Hive --hiveconf hive.root.logger=INFO,console

Hive数据仓库位置配置

Default /usr/hive/warehouse

在hive-site.xml中进行设置：

<property>

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

<value>/user/hive/warehouse</value>

<description>location of default database for the warehouse</description>

</property>

设置完成以后要设置权限

Hadoop fs –chmod g+w /user/hive/warsehouse

注意事项：

* 在仓库目录下，没有对默认的数据库default创建文件夹
* 如果某张表输入default数据库，直接在数据仓库目录下创建一个文件夹

\*bin/hive –e <ueoted-query-string>

Eg：bin/hive –e “select \* from db\_test.tb\_student;”

\*bin/hive –f <filename>

Filename例如放入sql语句

Eg：

Gy\_test.sql

select \* from db\_test.tb\_student;

bin/hive –f gy\_test.sql

bin/hive –f gy\_test.sql > xx.txt sql查询的结果保存在文件中

\*bin/hive –I <filename>

与用户自定义的udf相互使用。

退出

Exit;

Quit;

如何在命令行中对hdfs操作

Hive>dfs –ls / ;

操作linux命令的时候

Hive>!ls /usr/local/

Hive>!cat /usr/local/hello.txt

查看历史命令：

$HOME/.hivehistory

既在当前用户目录下面的.hivehistory目录下

Cd ~

Ls –al

员工表

create table if not exists default.emp(

Empno int,

Ename string,

Job string,

Mgr int,

Hiredate string,

Sal double,

Comm double,

Deptno int

)

Row format delimited fields terminated by ‘\t’

部门表

create table if not exists default.dept(

deptno int,

dname string,

loc string,

)

Row format delimited fields terminated by ‘\t’

分区表注意事项：

不能直接把数据让如到对应的文件件下面，例如如下：

create table if not exists default.dept\_part(

deptno int,

dname string,

loc string,

)

Partitioned by (day string)

Row format delimited fields terminated by ‘\t’

Hdfs dfs –mkdir –p /user/hive/warehouse/dept\_part/day=20150913

Hdfs dfs –put /usr/local/hive/gy\_test/dir/dept.txt /user/hive/warehouse/dept\_part/day=20150913

此时查询表default.dept\_part，里面的数据为空。

原因就是分区没有注册到元数据中。：可以查看mysql中hive的元数据库中的partition这个表，就会发现里面没有记录

解决方法

第一种方式

>msck repair table default.dept\_part;进行修复。

就可以了，此时查询mysql中hive的元数据库中的partition就会发现增加了一条数据。

第二种方式

Hdfs dfs –mkdir –p /user/hive/warehouse/dept\_part/day=20150914

Hdfs dfs –put /usr/local/hive/gy\_test/dir/dept.txt /user/hive/warehouse/dept\_part/day=20150914

Alter table default.dept\_part add partition(day = ‘20150914’)

查询一个表有多少分区

Show partitions dept\_part;

加载数据

Load data [local] inpath ‘filepath’

[overwrite] into table tablename

[partition (partcol1=val1,…)] //分区表特殊

Overwrite对表中的数据是否覆盖；如果不写，就是追加。

从hive中导出数据

方法一：

Insert overwrite [local] directory ‘/usr/local/hive/gy\_test\_dir’

[Row format delimited fields terminated by ‘\t’ collection items terminated by ‘\n’]

Select \* from default.emp; 走mapreduce

方法二：

Bin/hive –e “select \* from default.tem” > /usr/local/……/xx.txt

方法三：sqoop

Hdfs/hive -> rdbms

Rdbms -> hdfs/hive/hbase

使用export和import可以实现数据库的备份；

Export

导出，将Hive表中的数据，导出到外部

Import

导入，将外部数据导入Hive表中

Export table tablename to ‘export\_target\_path‘；

Export\_tartget\_path：指的是HDFS上的路径；

Create table db\_test.emp like default.emp;

Import table db\_test.emp from ‘….’

Order by

对全局数据的一个排序，仅仅只有一个reduce

Select \* from emp order by emp desc;

可以想到他的mapreduce只有一个reduce

Sort by

对每一个reduce内部数据进行排序，全局结果集来说不是排序的。

Set mapreduce.job.reduces = 3;

Select \* from emp sort by empno asc;

Insert overwrite local directory ‘/usr/local/hive/gy\_test\_dir\_sortby’ select \* from emp sort by empno asc;

Distribute by

分区partition

类似于mapreduce中分区partition的功能。对数据进行分区，通常

结合sort by结合使用

Insert overwrite local directory ‘/usr/local/hive/gy\_test\_dir\_distribute\_sortby’ select \* from emp distribute by deptno sort by empno asc;

注意事项：

Distribute by 必须在sort by之前。

Cluster by

Sort by和distribute by的组合

当distribute by和sort by 字段相同时，可以使用它进行代替。

Insert overwrite local directory ‘/usr/local/hive/gy\_test\_dir\_cluseter\_sortby’ select \* from emp cluster by empno ;

Order by

全局排序，一个reduce

sort by --- sql server中没有这个，

每个reduce内部进行排序，全局不是排序

Distribute by --- 类似于sql server中的group by

类似于MR中partition，进行分区，结合sort by使用

Cluster by

当distribute和sort字段相同时，使用方式。

UDF

编程步骤：

1. 继承org.apapche.hadoop.hive.ql.UDF
2. 需要实现evaluate函数；evaluate函数支持重载；
3. UDF必须要有返回类型，可以返回null，但是返回类型不能为void；
4. UDF中常用Text/LongWritable等类型，不推荐使用java类型、

Usage；

Public class LowerUDF extend UDF{

Public Text evaluate(Text str) {

If(null == str.toString()){

Return null;

}

Return new Text(str.toString().toLowerCase());

}

}

打成jar包例如：hiveudf.jar

在hive中执行如下：

方法一：

Add jar /usr/local/hive/hiveudf.jar

Create temporary function my\_lower as ‘类的全名’；相当于注册

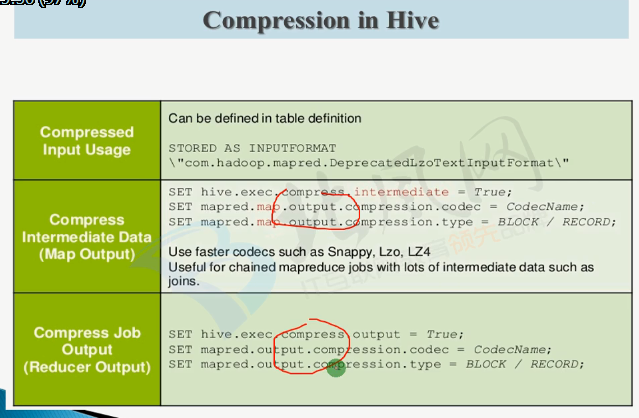
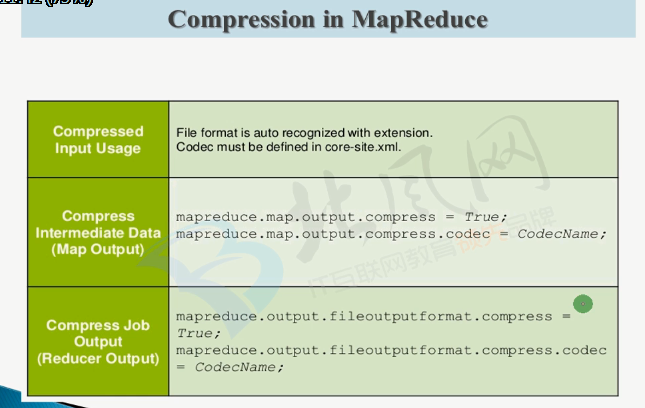
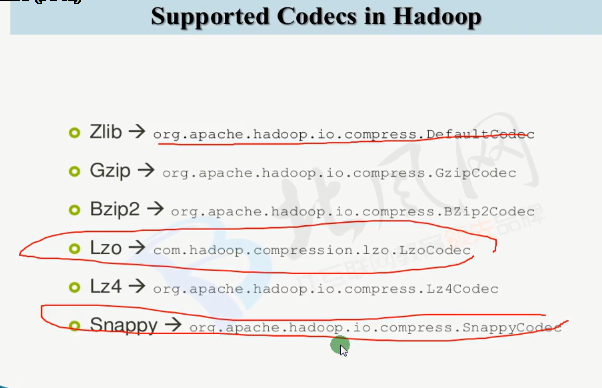
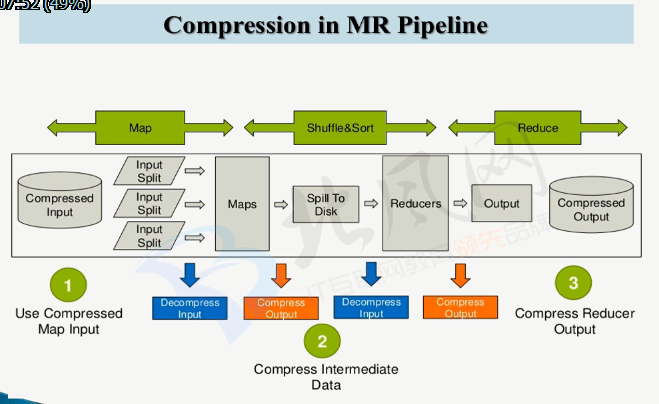
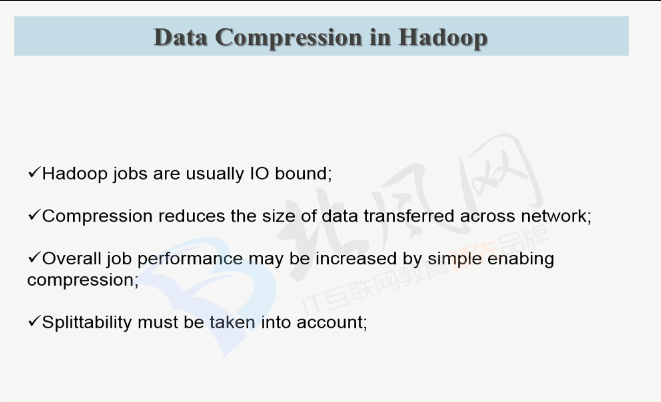
Select ename,my\_lower(ename) lowername from emp limit 5;

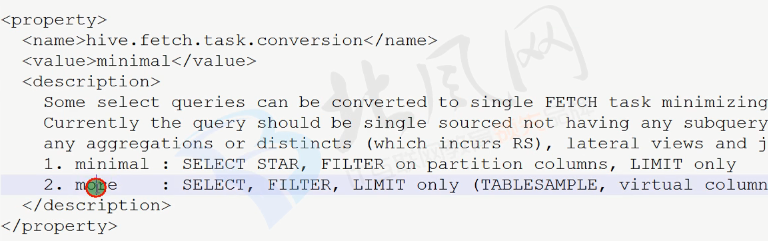
方法二：

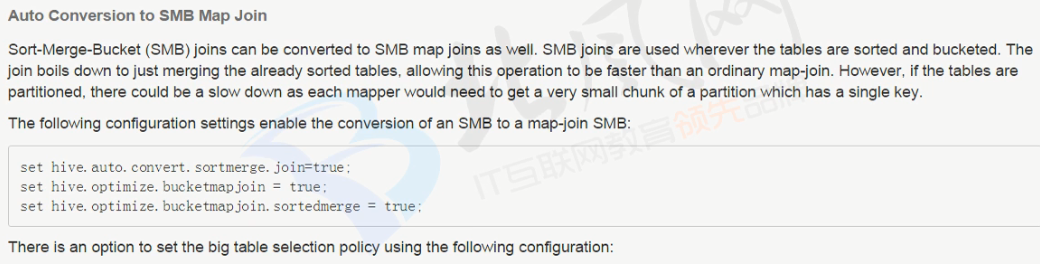
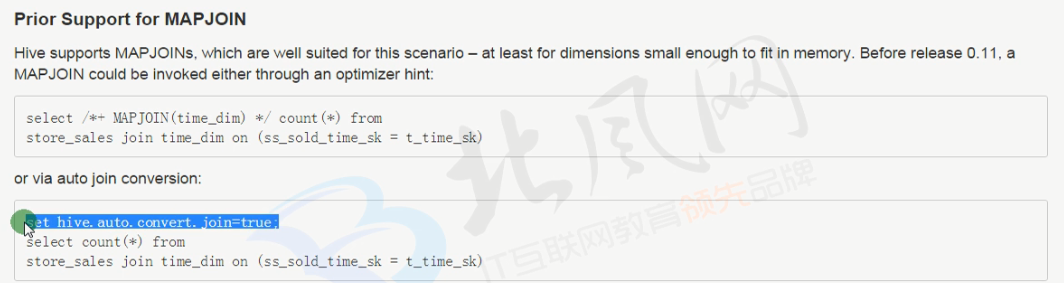
Add jar /usr/local/hive/hiveudf.jar

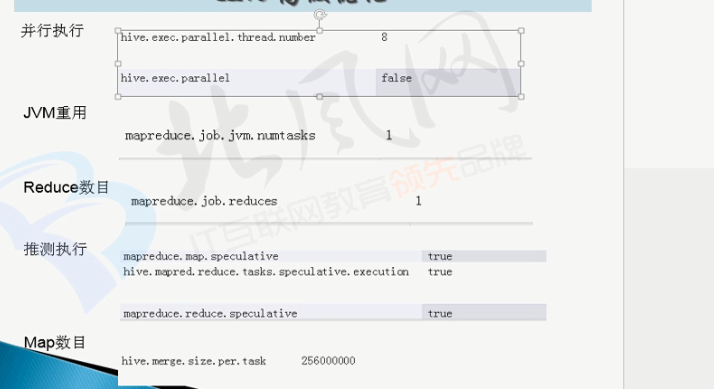
Create function self\_lower as ‘类的全名’ using ‘hdfs上面jar包的路径’;

Select ename,self\_lower(ename) lowername from emp limit 5;









建议：hive.exec.parallel.thread.number为10到20个之间，例如15

Mapreduce.job.jvm.numtasks : <10 例如9

Reduce数目设置就是看reduce任务大体上是否同时完成。

推测执行建议都关闭：全部设置成false