目录

**[一 概述 1](#_Toc20638)**

[1 什么是kettle 1](#_Toc2486)

[2 Kettle核心知识点 1](#_Toc24197)

**[二 Kettle的安装和部署 2](#_Toc19652)**

[1 Kettle的官网 2](#_Toc13618)

[2 windows下安装使用 2](#_Toc14040)

[3 创建资源库 26](#_Toc17347)

**[三 Linux下安装 37](#_Toc24410)**

#### 一 概述

##### 1 什么是kettle

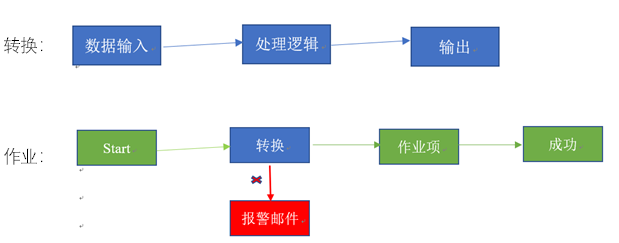
Kettle是一款开源的ETL工具，纯java编写，可以在Window、Linux、Unix上运行，绿色无需安装，数据抽取高效稳定。

##### 2 Kettle核心知识点

###### 2.1 Kettle工程存储方式

1) 以XML形式存储  
2) 以资源库方式存储(数据库资源库和文件资源库)

###### 2.2 Kettle的设计



简述：Transformation：完成针对数据的基础转换  
 Job:完成整个工作流的控制  
区别：  
 1. 作业是步骤流，转换是数据流。这是job和transformation的最大区别。  
 2. 作业的每个步骤，必须等到前面的步骤执行完毕，后面的步骤才会执行；而转换会一次性把所有控件全部先启动（一个控件对应启动一个线程）。

###### 2.3 Kettle的组成

1 勺子（Spoon.bat/Spoon.sh）:一个图形化界面，可以让我们用图形化的方式开发转换和作业。  
2 Pan.bat/Pan.sh : 利用Pan可以使用命令行来调用转换  
3 Kitchen.bat/Kitchen.sh:利用Kitchen可以使用命令行来调用作业  
4 Carte.bat/Carte.sh:Carte是一个轻量级的web容器，用于建立专用、远程的ETL服务器。

###### 2.4 Kettle的作用



#### 二 Kettle的安装和部署

##### 1 Kettle的官网

https://community.hitachivantara.com/docs/DOC-1009855  
https://sourceforge.net/projects/pentaho/files/Data%20Integration/

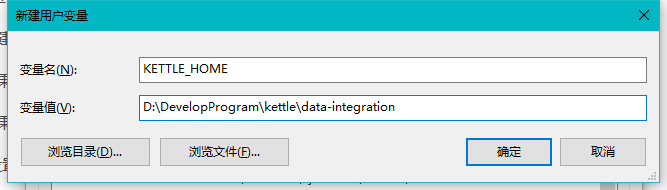
##### 2 windows下安装使用

###### 2.1 概述

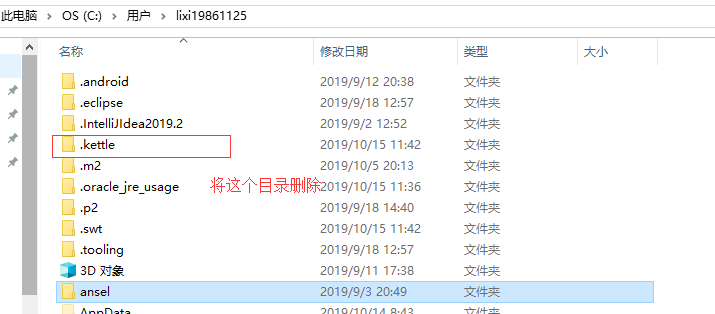
在实际企业开发中，都是在本地环境下进行kettle的job和Transformation开发的，可以在本地运行，也可以连接远程机器运行

###### 2.2 安装

1) 安装jdk  
2) 下载kettle压缩包，因kettle为绿色软件，解压缩到任意本地路径即可  
3) 双击Spoon.bat，启动图形化界面工具，就可以直接使用了  
  
3）如果出现kettle启动闪退请检查以下：  
\* 查看是否有配置好环境变量



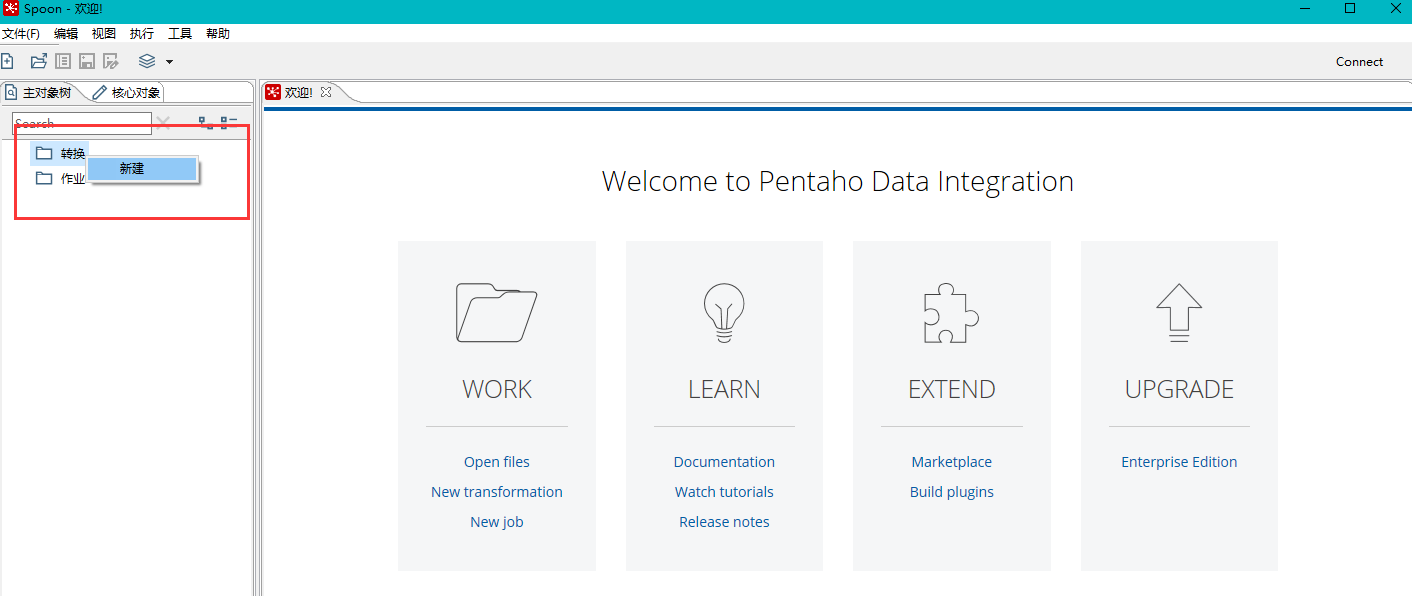
\* 配置JAVA\_HOME并且保证jdk版本至少为1.7以及以上  
\* 修改spoon.bat文件内存  
  
if "%PENTAHO\_DI\_JAVA\_OPTIONS%"=="" set PENTAHO\_DI\_JAVA\_OPTIONS="-Xms1024m" "-Xmx2048m" "-XX:MaxPermSize=256m"  
改为  
  
if "%PENTAHO\_DI\_JAVA\_OPTIONS%"=="" set PENTAHO\_DI\_JAVA\_OPTIONS="-Xms512m" "-Xmx512m" "-XX:MaxPermSize=256m"  
  
\* 然后删除如以下图示内容



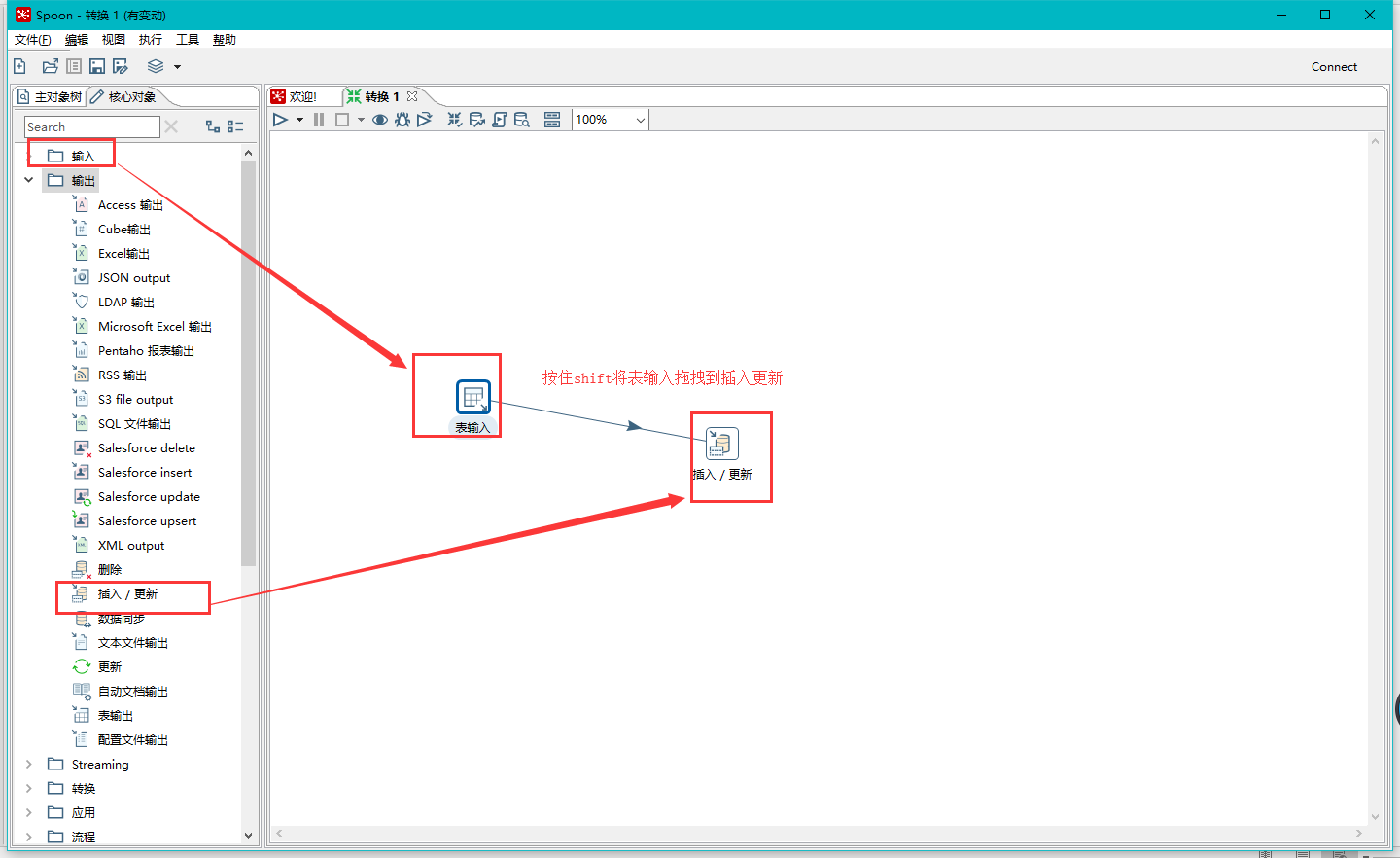
###### 2.3 案例1

1) 案例一 把stu1的数据按id同步到stu2，stu2有相同id则更新数据  
  
\* 在mysql中创建两张表  
  
drop database if exists kettle;  
create database if not exists kettle;  
  
use kettle;  
  
create table stu1(  
id int,  
name varchar(20),  
age int  
);  
  
create table stu2(  
id int,  
name varchar(20)  
);  
  
\* 往两张表中插入一些数据  
insert into stu1 values(1001,'selina',20),(1002,'hebe',18), (1003,'ella',23);  
insert into stu2 values(1001,'elva');

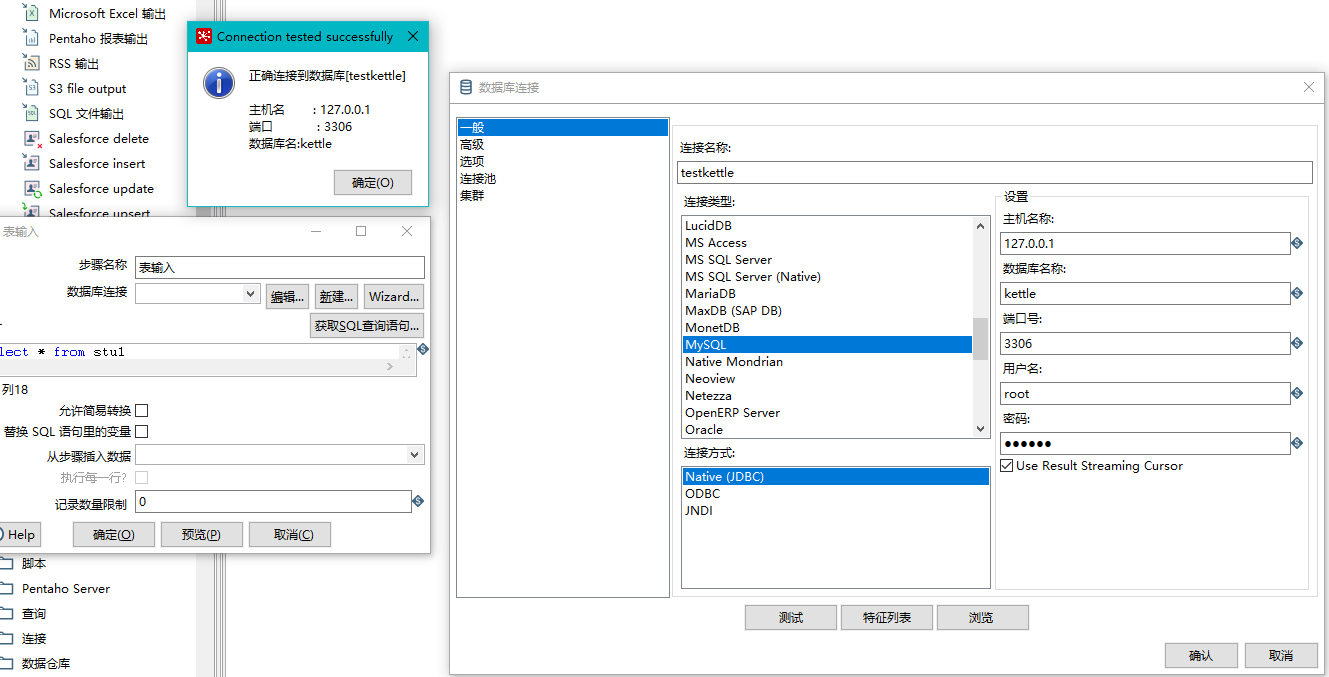
* 在kettle中新建转换



* 分别在输入和输出中拉出表输入和插入/更新

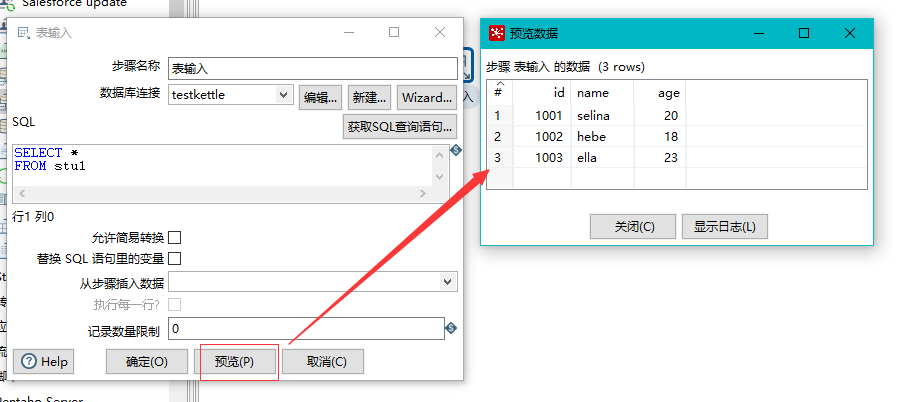


* 双击表输入对象，填写相关配置，测试是否成功

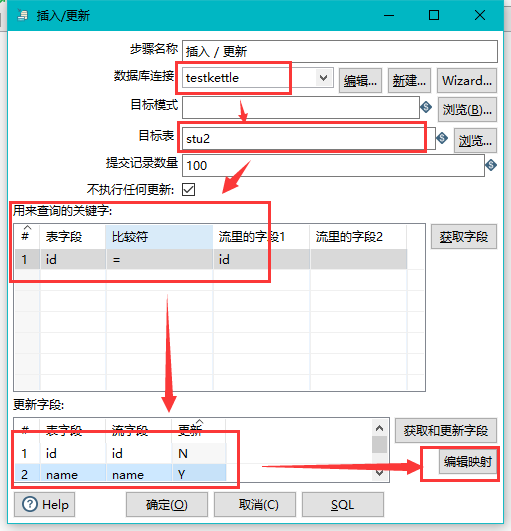


错误连接数据库 [数据库连接测试] : org.pentaho.di.core.exception.KettleDatabaseException:   
Error occurred while trying to connect to the database  
  
解决方案：  
将对应的mysql驱动包放到下kettle的目录：data-integration\lib下。

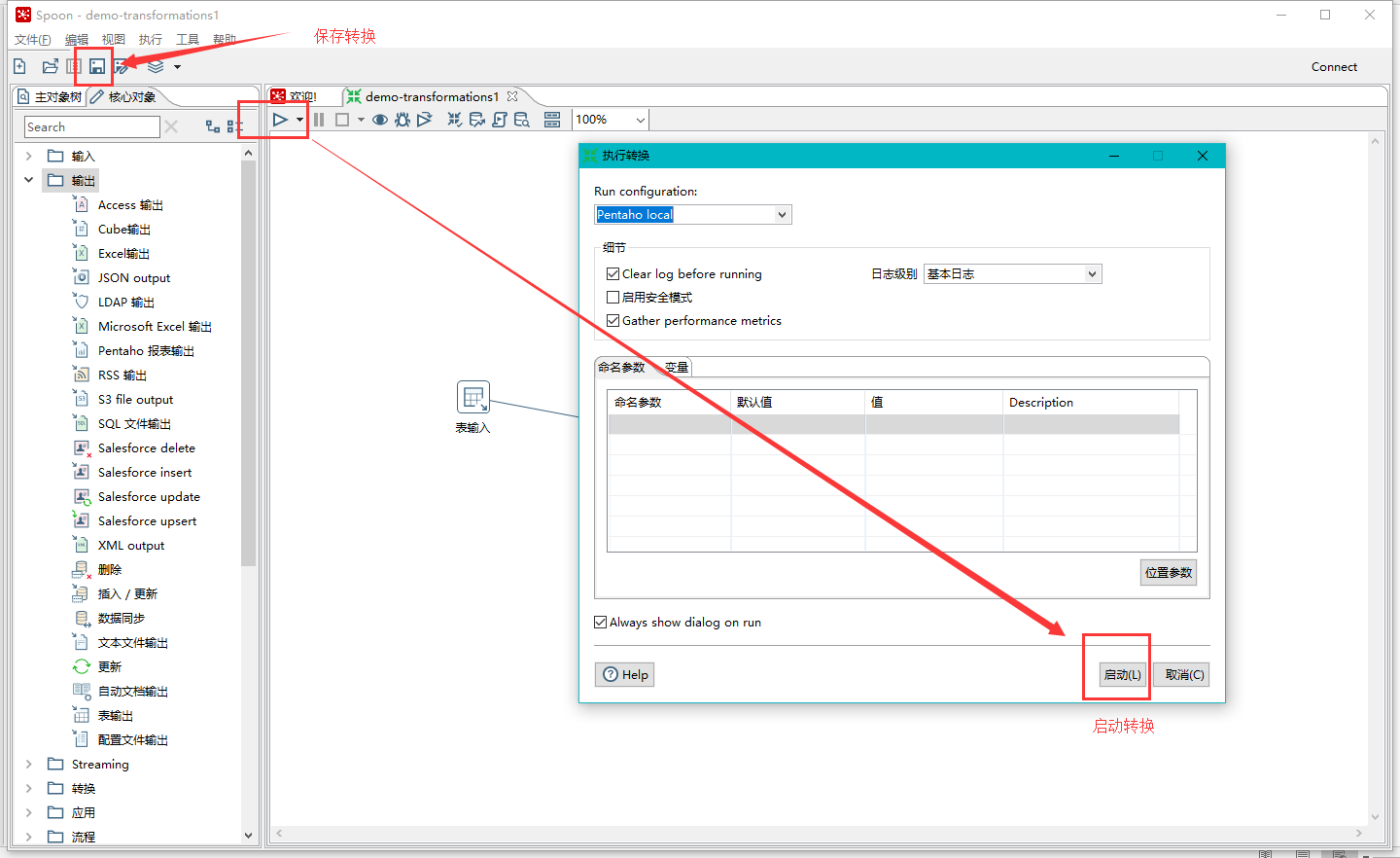
* 预览

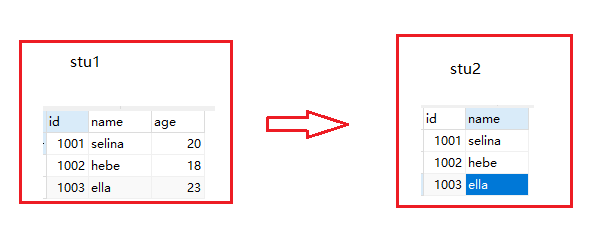


* 双击 更新/插入对象，填写相关配置



* 保存转换，启动运行，去mysql表查看结果



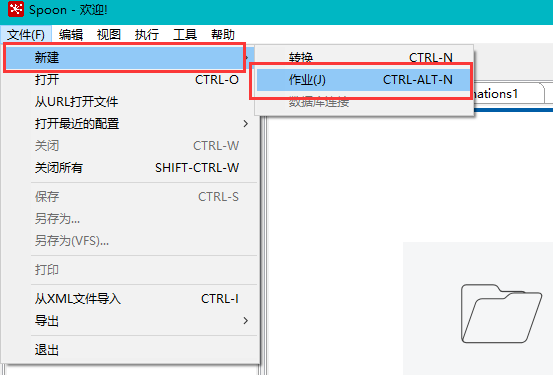


###### 2.4 案例2

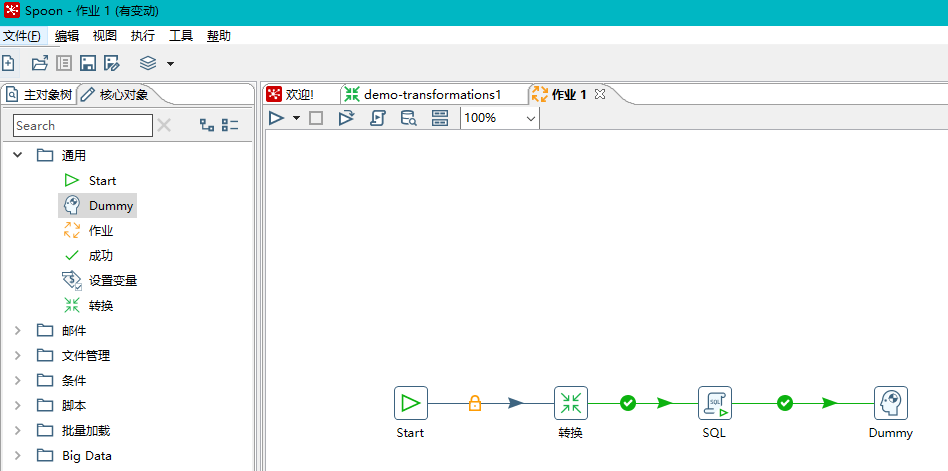
* 需求

使用作业执行上述转换，并且额外在表stu2中添加一条数据

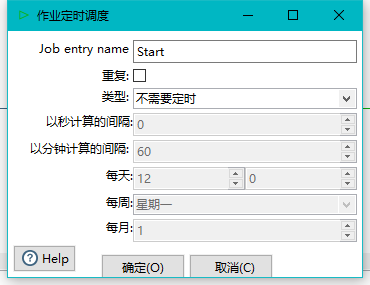
* 新建一个作业



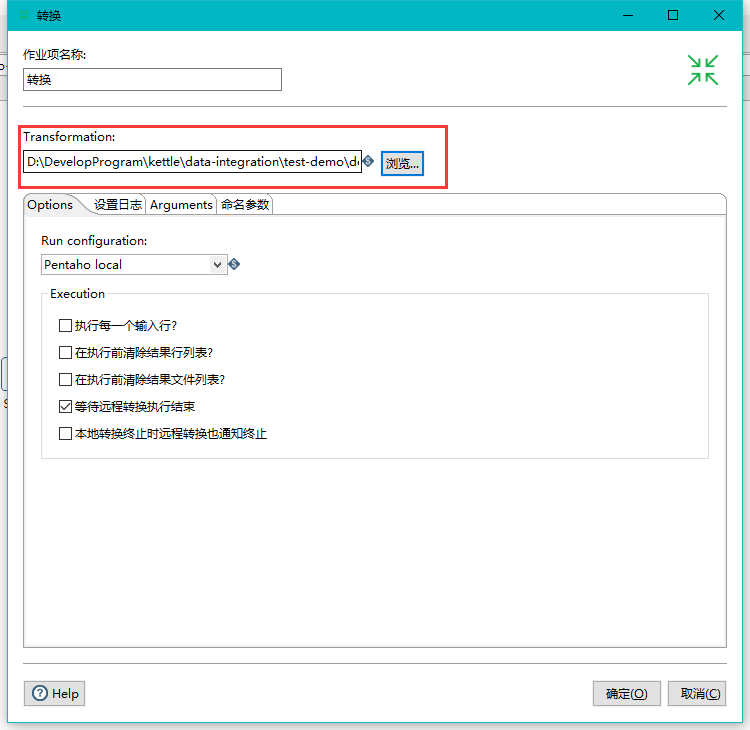
* 按图示拉取组件



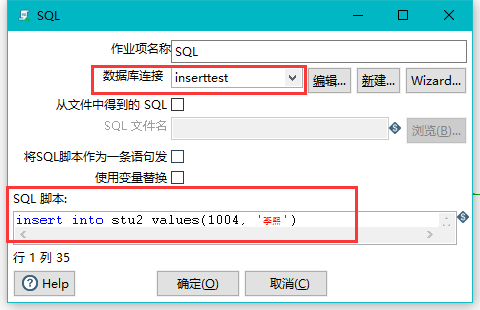
* 双击Start编辑Start



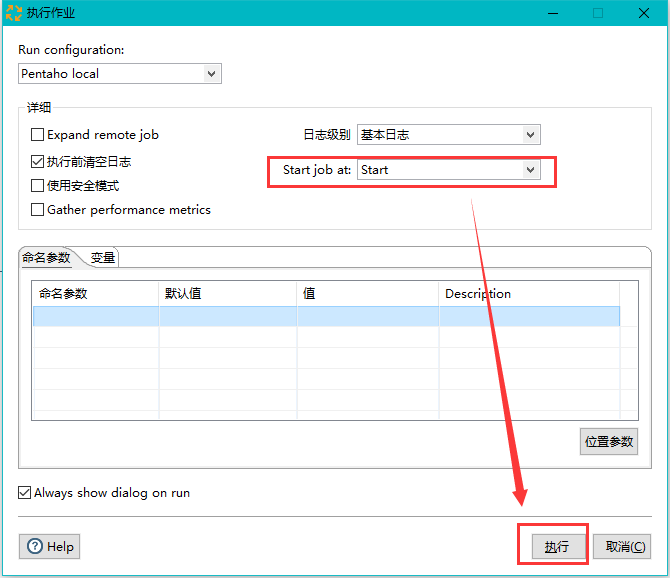
* 双击转换，选择案例1保存的文件



* 双击SQL，编辑SQL语句



* 保存执行



###### 2.5 案例3

* 需求

将hive表的数据输出到hdfs

* 修改解压目录下的data-integration\plugins\pentaho-big-data-plugin下的plugin.properties

active.hadoop.configuration=hdp26

* 将hbase/hadoop/hive的\*-site.xml拷贝到指定目录下

data-integration\plugins\pentaho-big-data-plugin\hadoop-configurations\hdp26

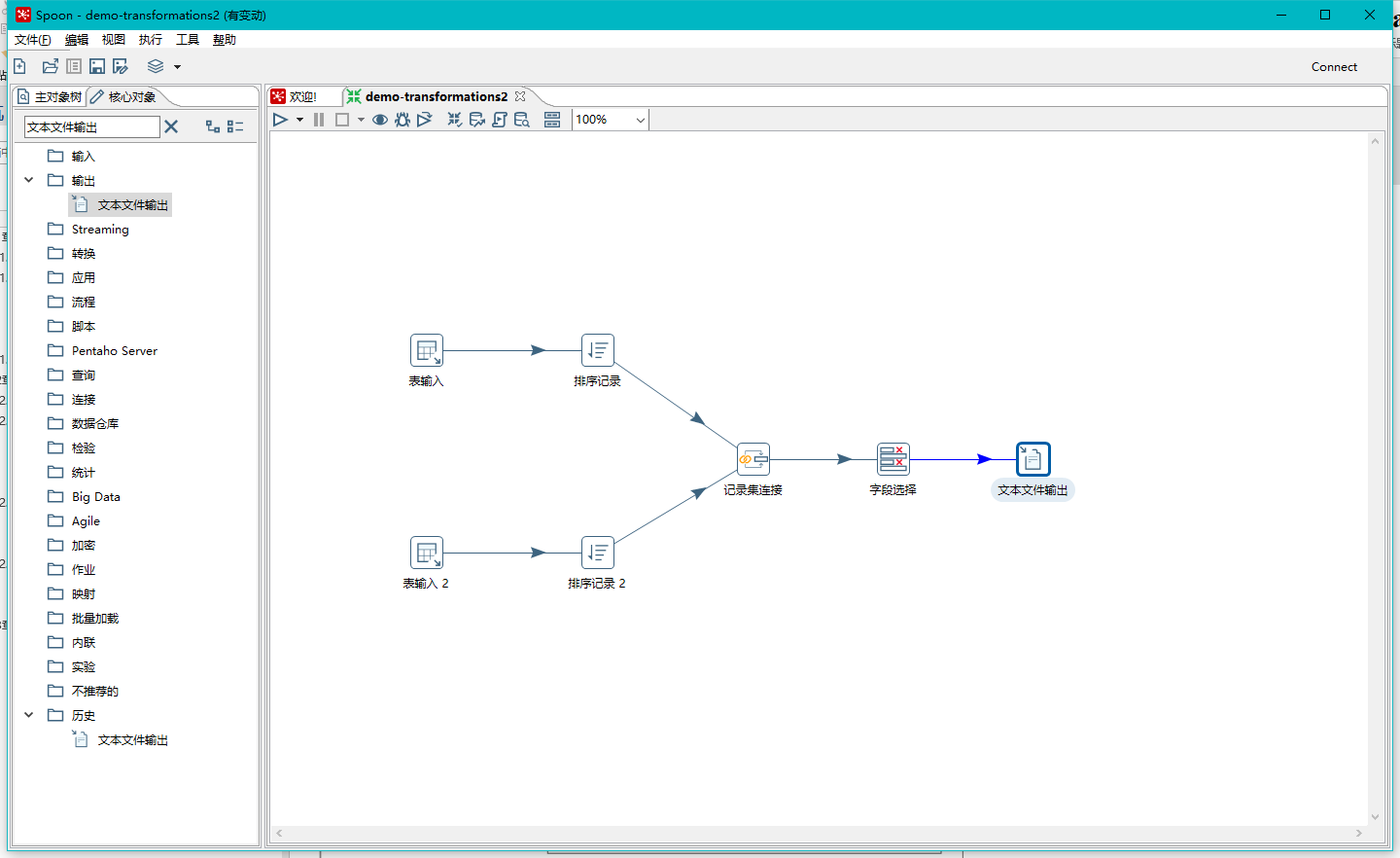
* 启动启动hdfs，yarn，hbase集群的所有进程，启动hiveserver2服务
* 进入beeline，查看10000端口开启情况
* 创建两张表dept和emp

DROP DATABASE IF EXISTS hue CASCADE;  
CREATE DATABASE IF NOT EXISTS hue;  
  
CREATE TABLE hue.dept(  
deptno int,   
dname string,  
loc string  
)  
ROW FORMAT DELIMITED  
FIELDS TERMINATED BY '\t';  
  
CREATE TABLE hue.emp(  
empno int,  
ename string,  
job string,  
mgr int,  
hiredate string,  
sal double,  
comm int,  
deptno int  
)  
ROW FORMAT DELIMITED  
FIELDS TERMINATED BY '\t';

* 插入数据

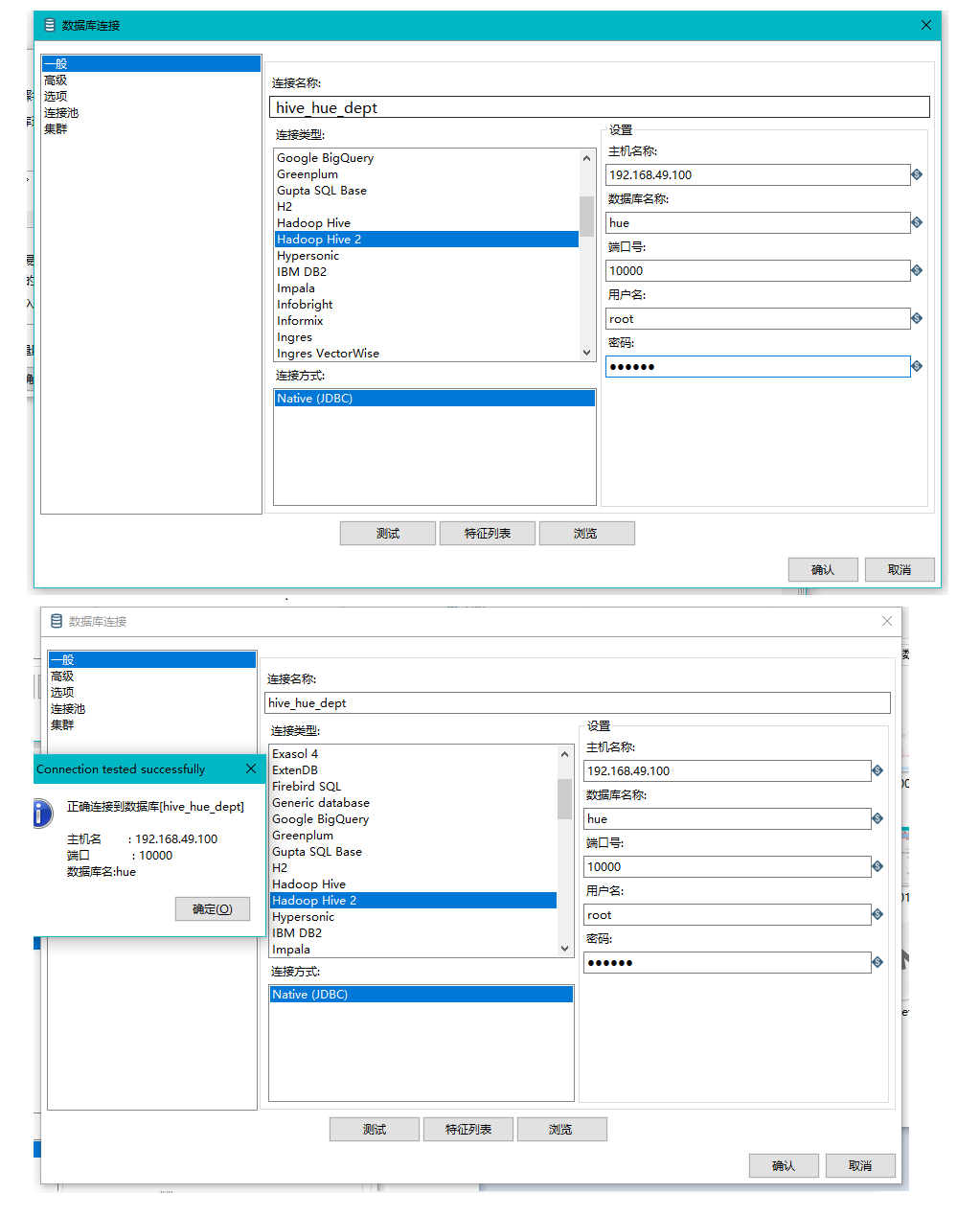
insert into hue.dept values(10,'Java','Beijing'),(20,'Bigdata','Shanghai'),(30,'Html5','Hangzhou'),(40,'UI','Chongqing');  
  
insert into hue.emp values(7369,'lixi','CLERK',7902,'1980-12-17',800,NULL,20),(7499,'rock','SALESMAN',7698,'1980-12-17',1600,300,30),(7521,'lee','SALESMAN',7698,'1980-12-17',1250,500,30),(7566,'narudo','MANAGER',7839,'1980-12-17',2975,NULL,20);

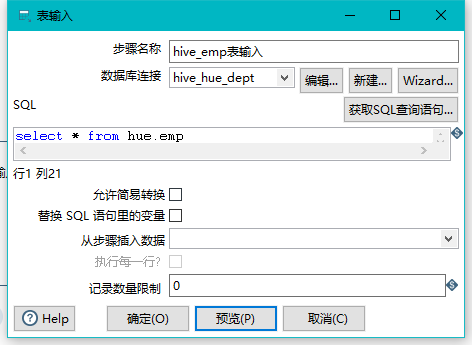
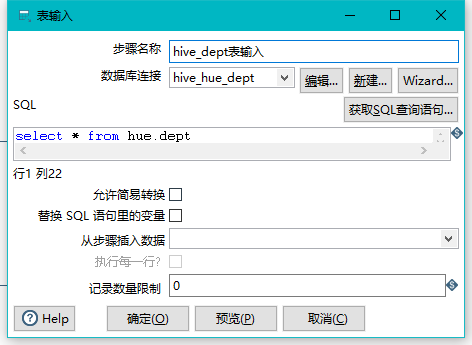
* 按下图建立流程图



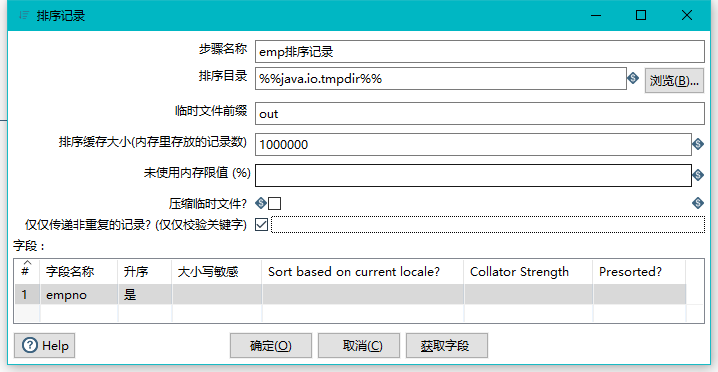
* 设置表输入，连接hive

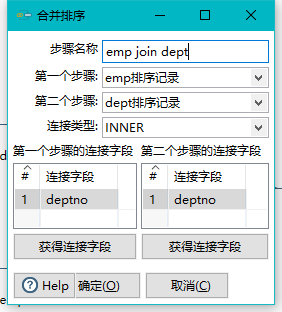
1. 这里连接hive一定直接死翘翘。因为我们的cdh版本是5.7.6。我们使用的kettle默认的cdh版本5.14.0。所以，里面的某些jar包需要我们自己手动指定。  
2. 第一先将kettle的D:\DevelopProgram\kettle\data-integration\plugins\pentaho-big-data-plugin\hadoop-configurations\cdh514下的\*-site.xml使用服务器的配置文件替换  
3. 修改D:\DevelopProgram\kettle\data-integration\plugins\pentaho-big-data-plugin目录下的plugins.properties文件：active.hadoop.configuration=cdh514  
4. 下载cdh5.7.6的hadoop的jar和hive的jar，替换lib目录下的hive和hadoop

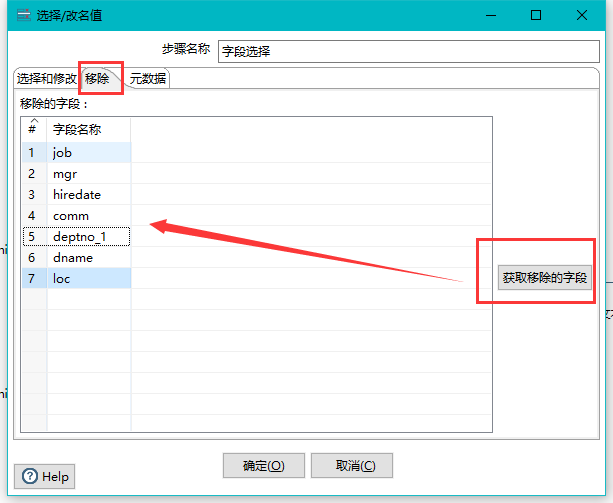




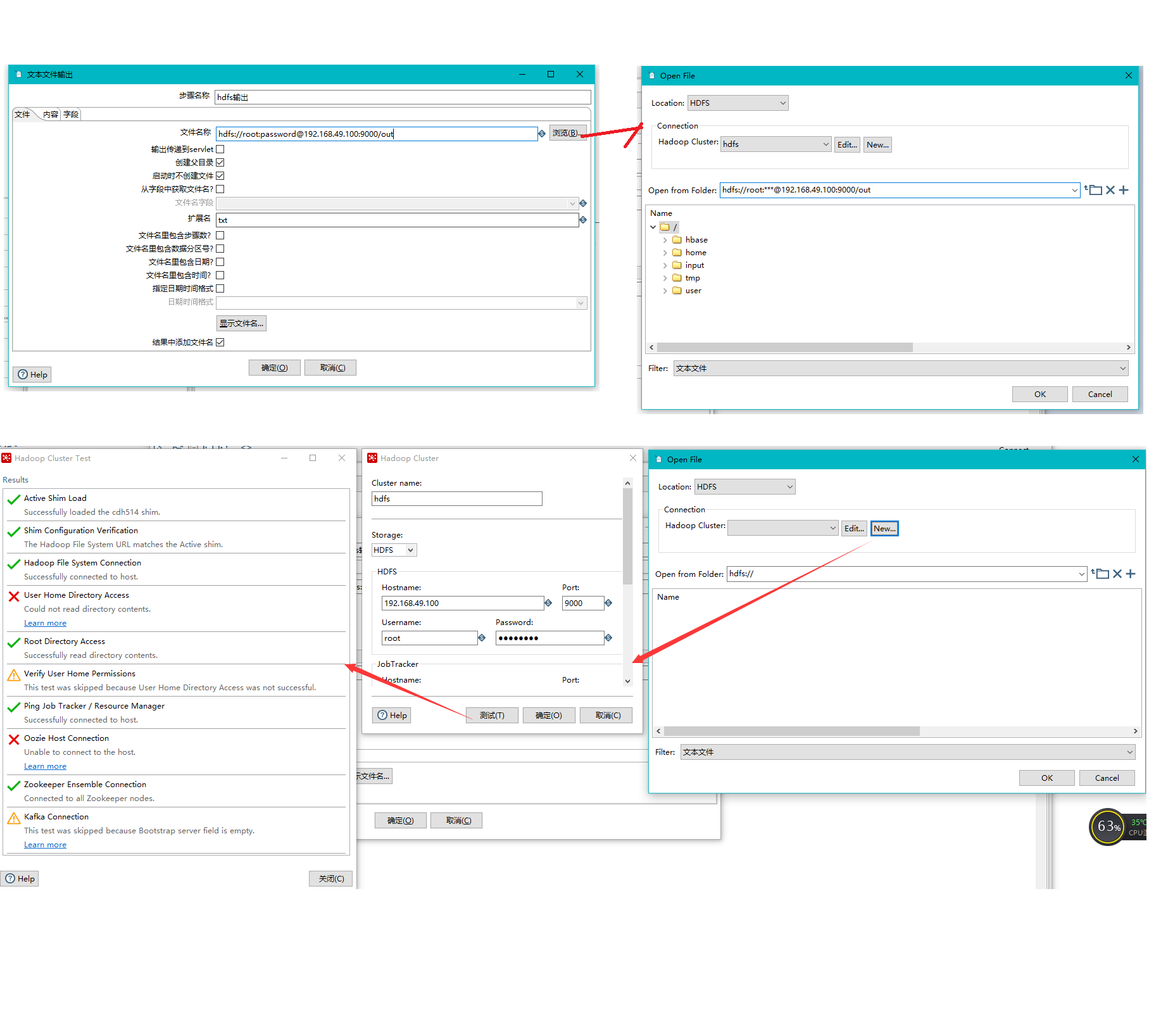
* 设置排序记录

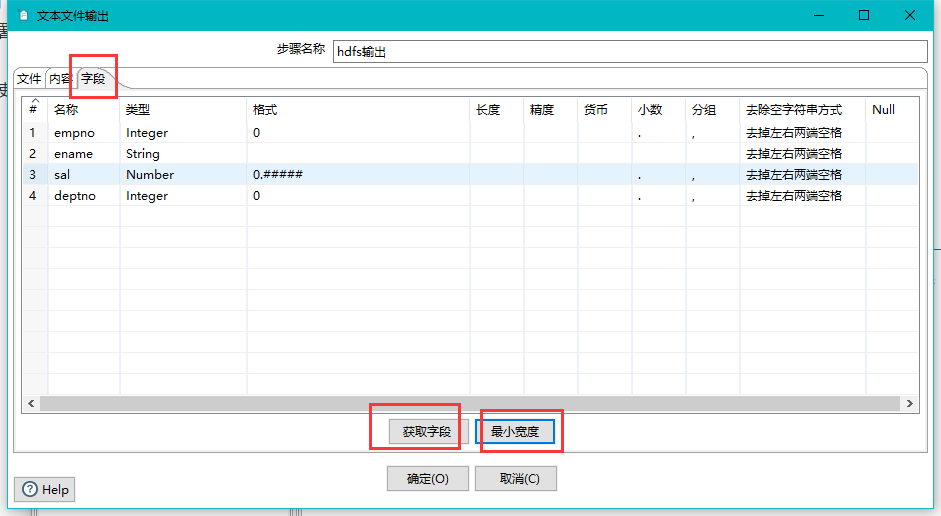


* 设置记录集连接
* 
* 设置字段选择

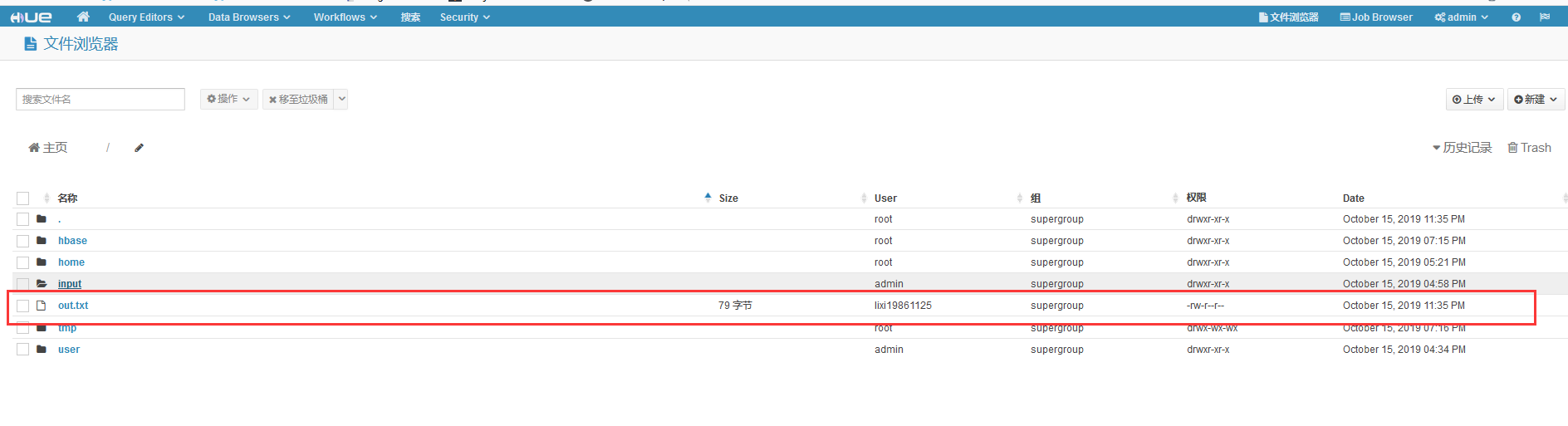


* 设置文件输出





* 保存并运行查看hdfs



###### 2.6 案例4

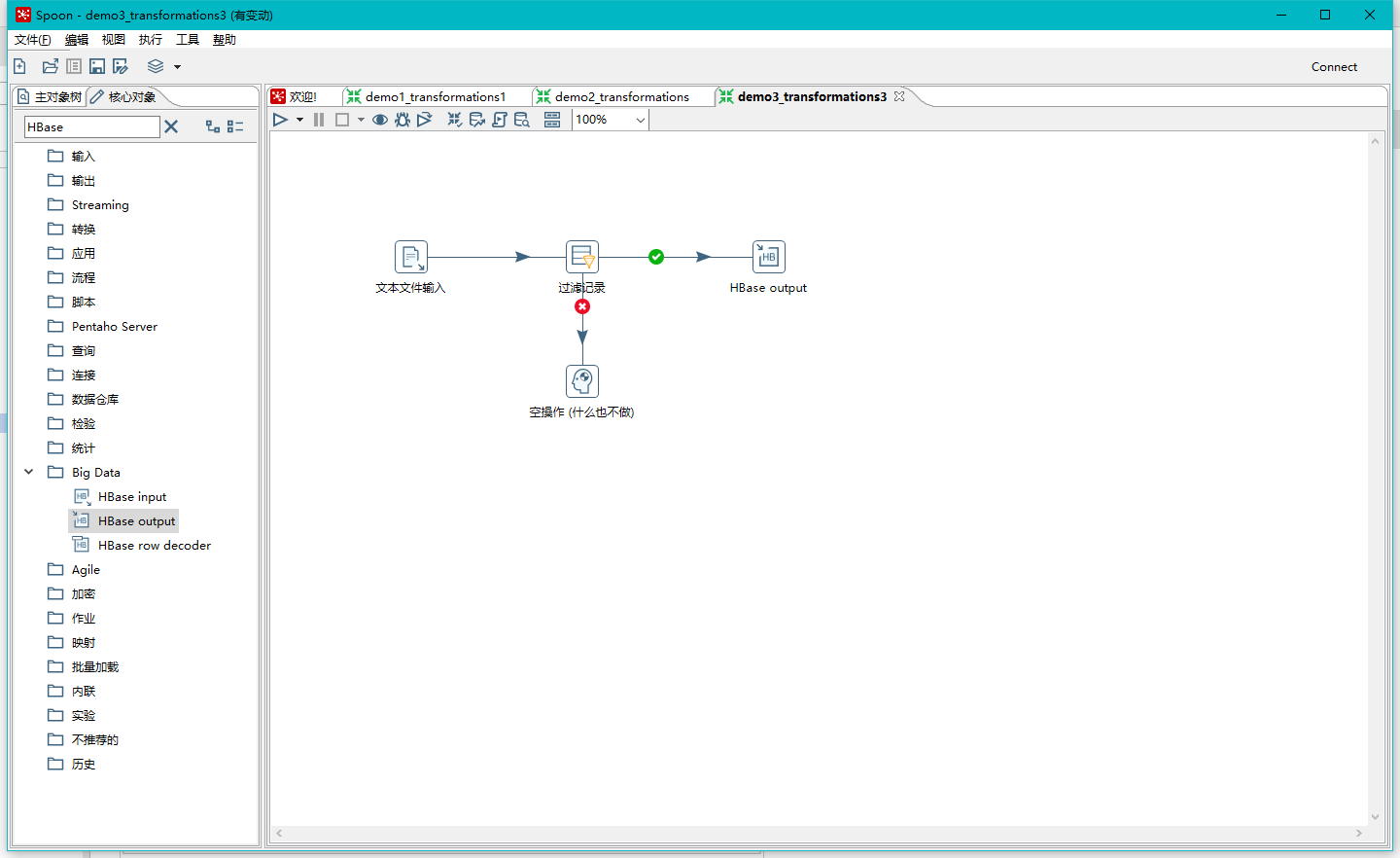
* 需求

读取hdfs文件并将sal大于1000的数据保存到hbase中

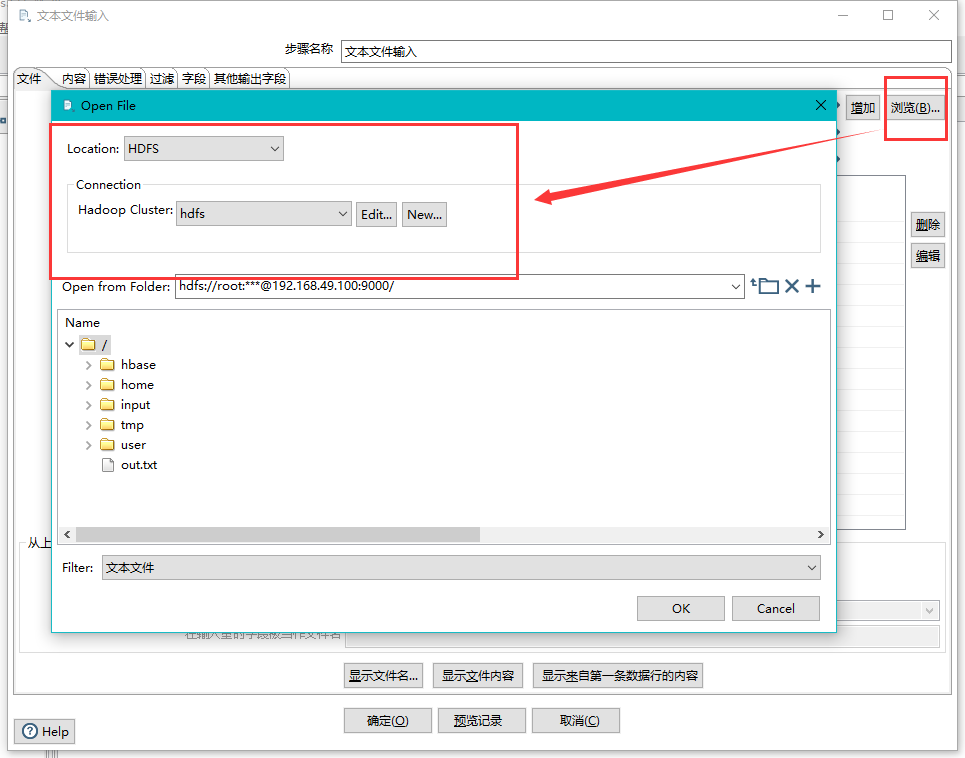
* 在HBase中创建一张表用于存放数据

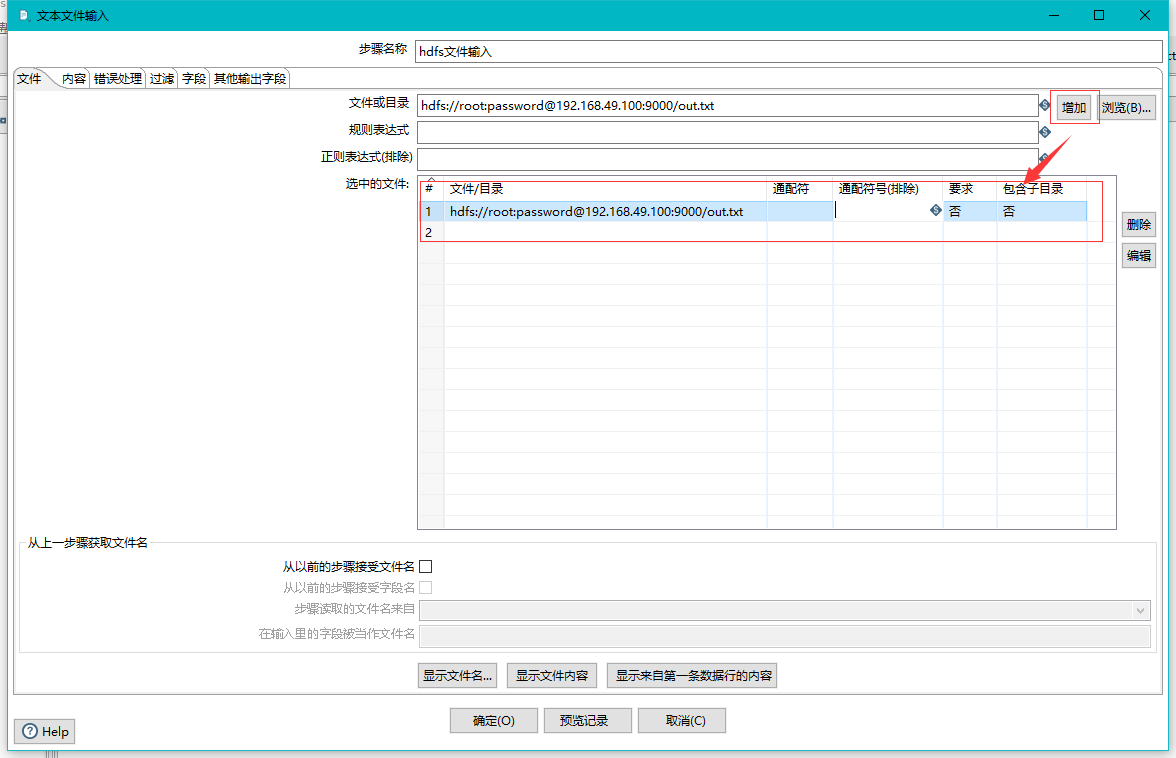
hbase shell  
create 'emp','info'

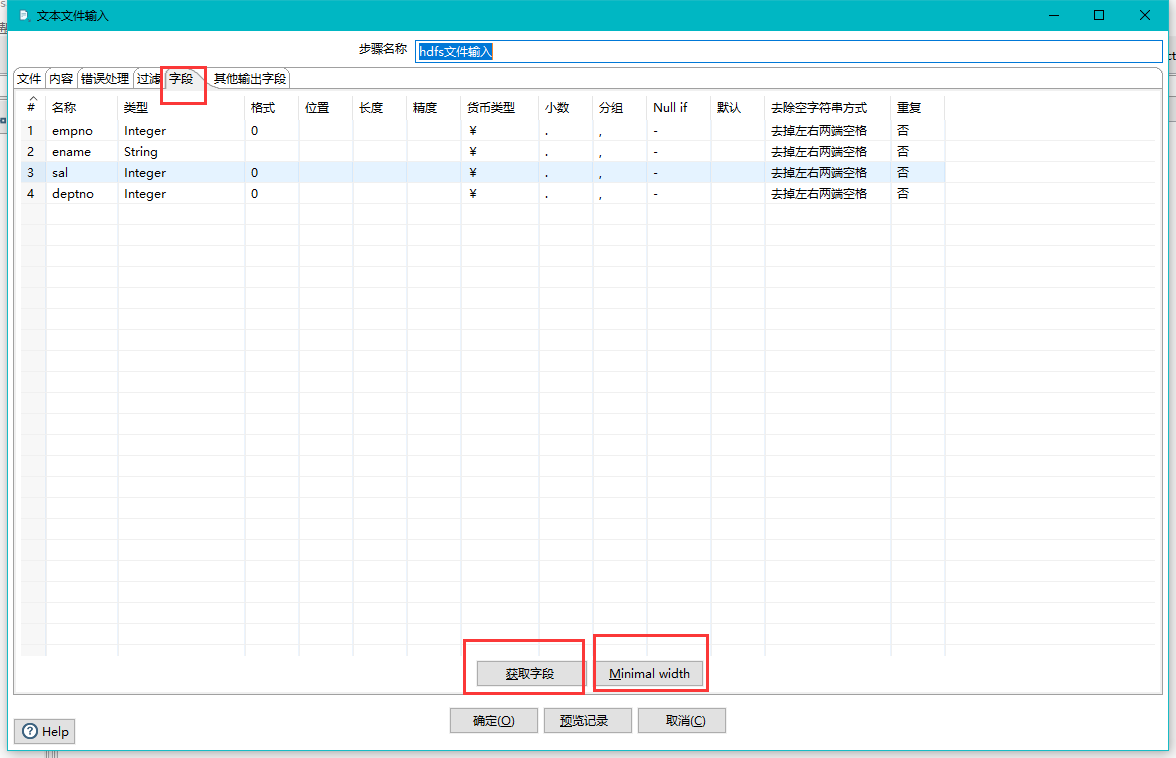
* 按下图建立流程图



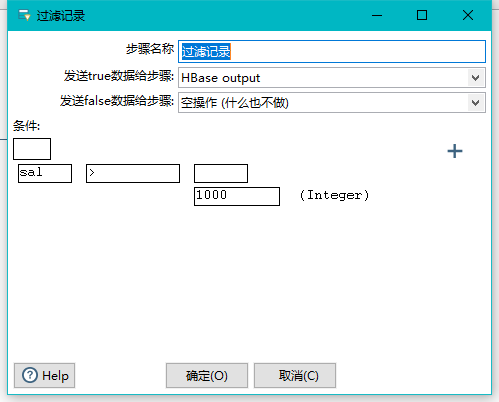
* 设置文件输入，连接hdfs



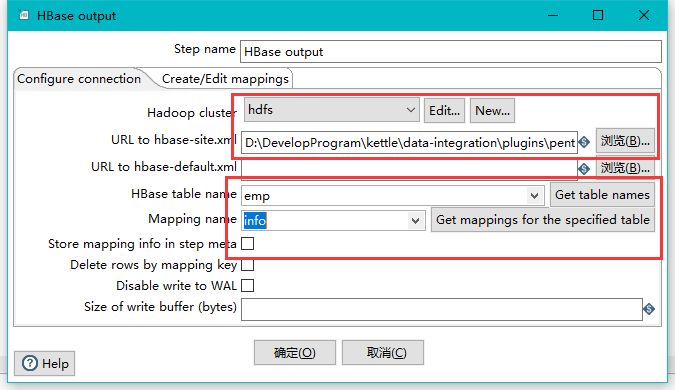
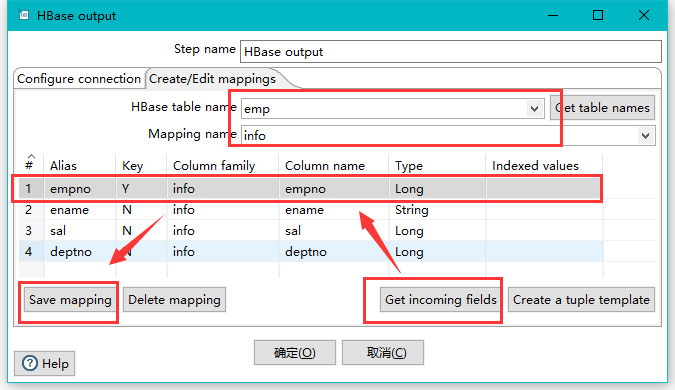




* 设置过滤记录



* 设置HBase output



异常：org.pentaho.big.data.kettle.plugins.hbase.HBaseConnectionException  
解决方案：  
将hbase的cdh5.7.6版本的jar包下载好，拷贝到D:\DevelopProgram\kettle\data-integration\plugins\pentaho-big-data-plugin\hadoop-configurations\cdh514\lib\pmr  
  
异常：Caused by: java.lang.NoClassDefFoundError: Could not initialize class org.apache.hadoop.hbase.protobuf.ProtobufUtil（检查kettle日志）  
上网找了说有三种情况  
1. 缺少protobuf-java 2.5.0的jar包  
2. 多个protobuf jar包互相冲突  
3. hbase-site.xml中hbase.rootdir配置项写错成一个不存在的hdfs的host了  
  
异常：Caused by: java.lang.RuntimeException: Failed to create local dir \\home\apps\hbase-1.2.0-cdh5.7.6\tmp\local\jars, DynamicClassLoader failed to init  
是因为我的hbase-site.xml的hbase.tmp.dir多配置了个\造成的  
  
异常：Caused by: java.net.UnknownHostException: hue1  
在windows的hosts中配置hue1的ip映射关系即可，这里建议将hue2、hue3一起配置  
  
tip:  
注意：若报错没有权限往hdfs写文件，在Spoon.bat中第119行添加参数  
"-DHADOOP\_USER\_NAME=root" "-Dfile.encoding=UTF-8"

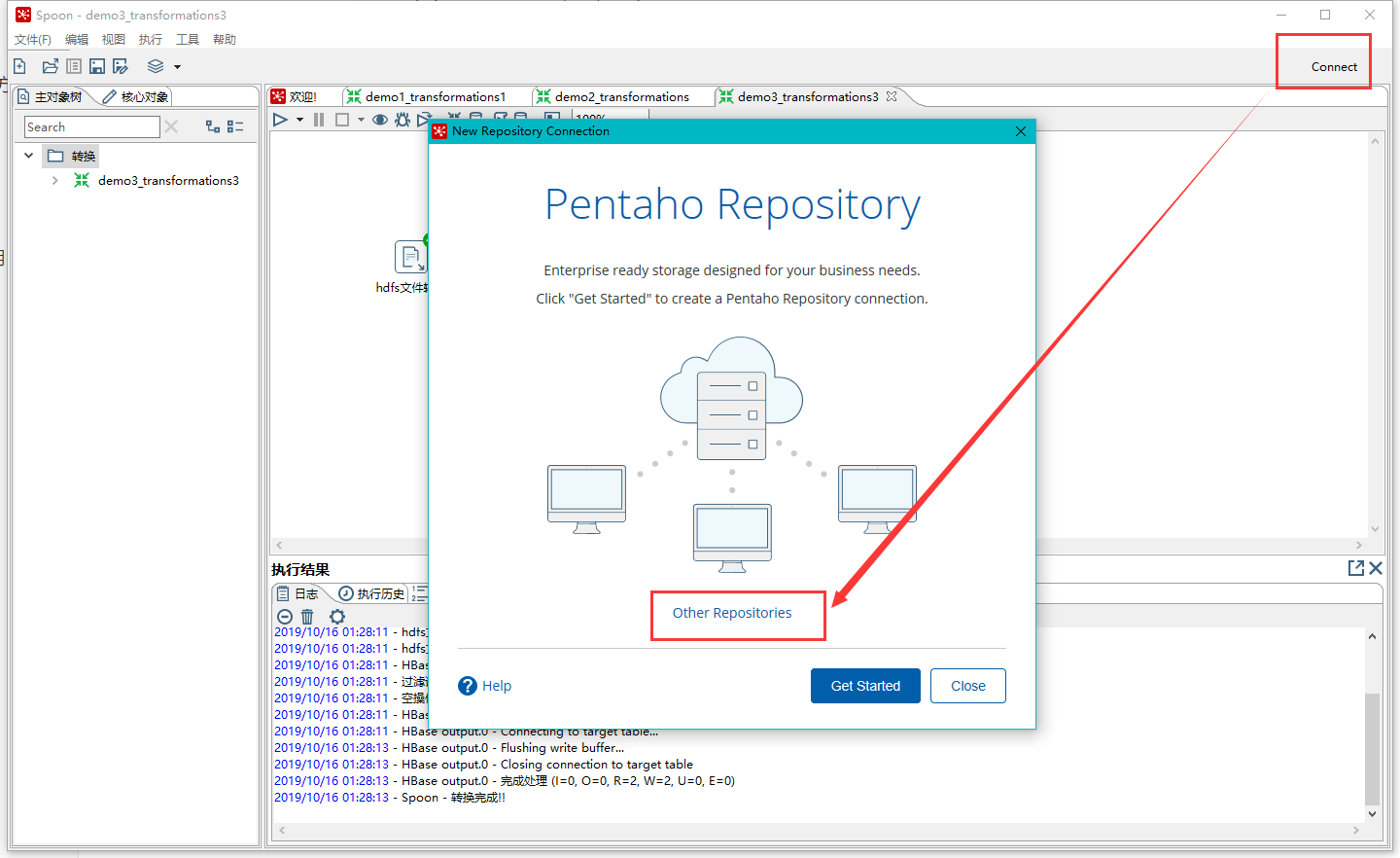
##### 3 创建资源库

###### 3.1 数据库资源库

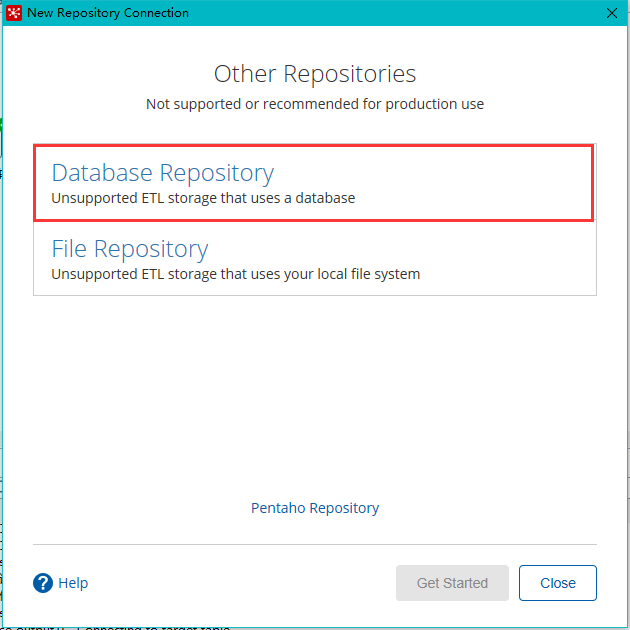
数据库资源库是将作业和转换相关的信息存储在数据库中，执行的时候直接去数据库读取信息，很容易跨平台使用

###### 3.2 数据库资源库操作

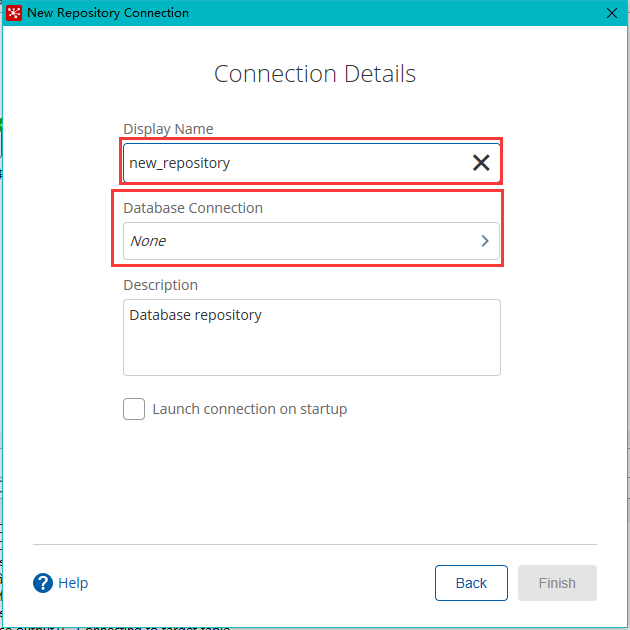
1. 点击右上角connect，选择Other Resporitory

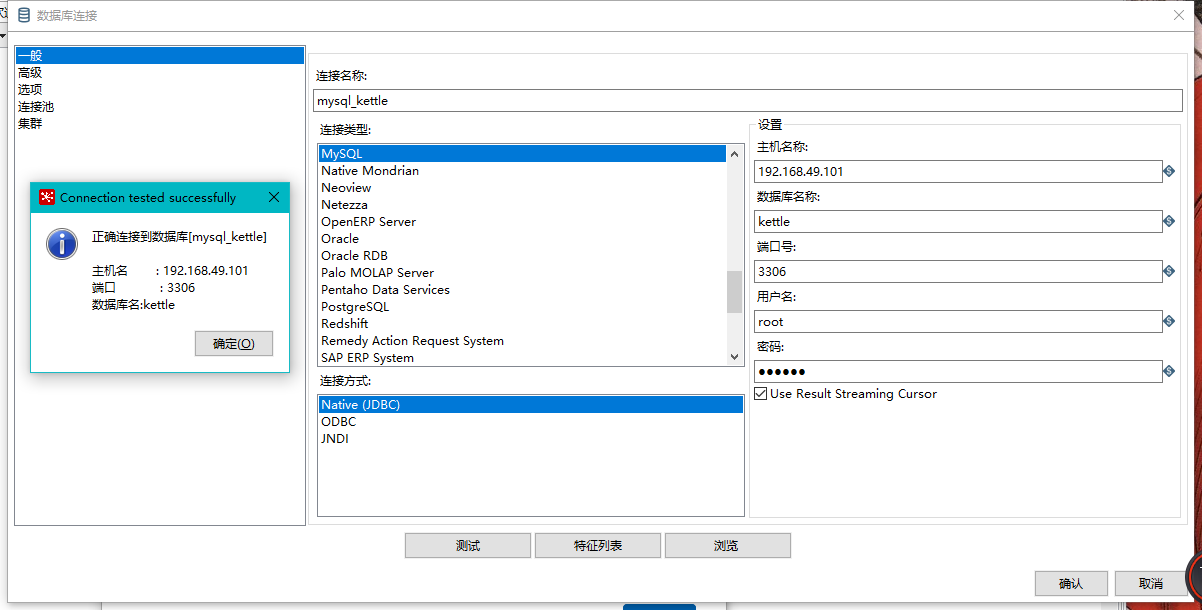


2. 选择Database Repository

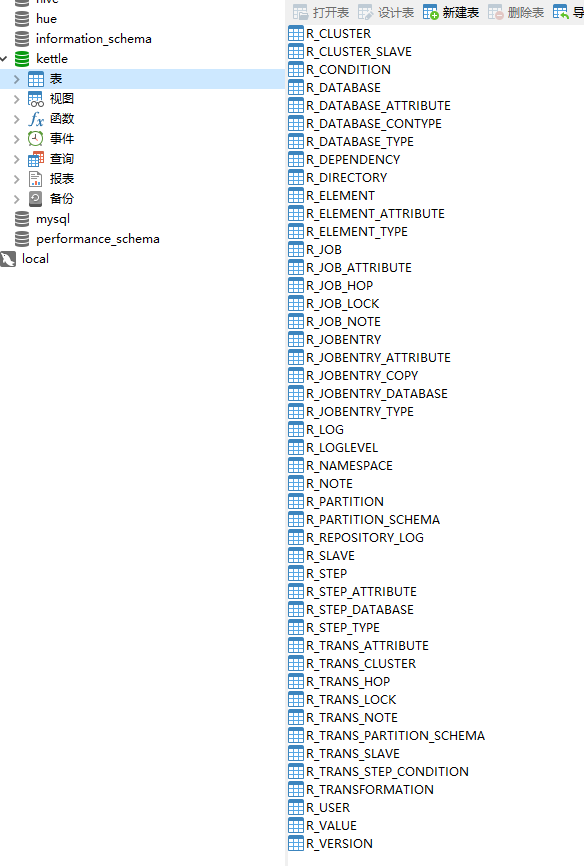


3. 建立新连接

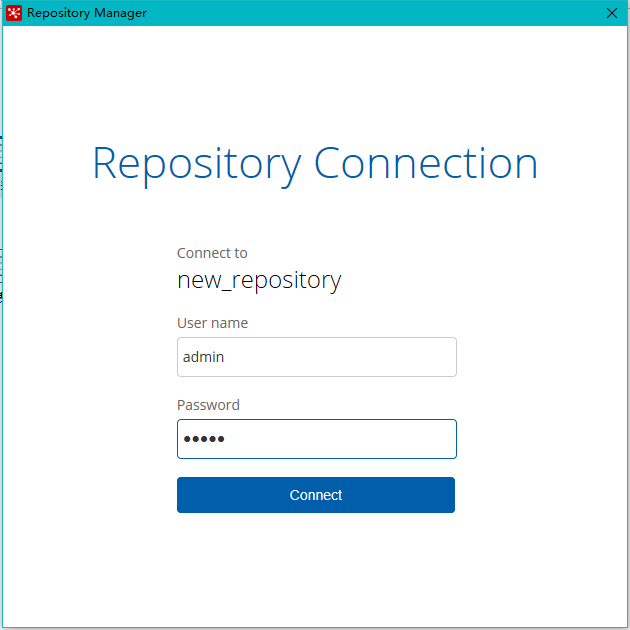




错误：  
Error connecting to database: (using class org.gjt.mm.mysql.Driver)  
解决方案：  
如果转换中可以使用连接数据库，但是创建资源库报错，一般情况下是jdbc驱动jar包版本的问题。尽量选用高一点的版本  
  
4. 填好之后，点击finish，会在指定的库中创建很多表，至此数据库资源库创建完成  
tip:  
这里表必须一次性完成连接建立，否则表可能无法自动创建



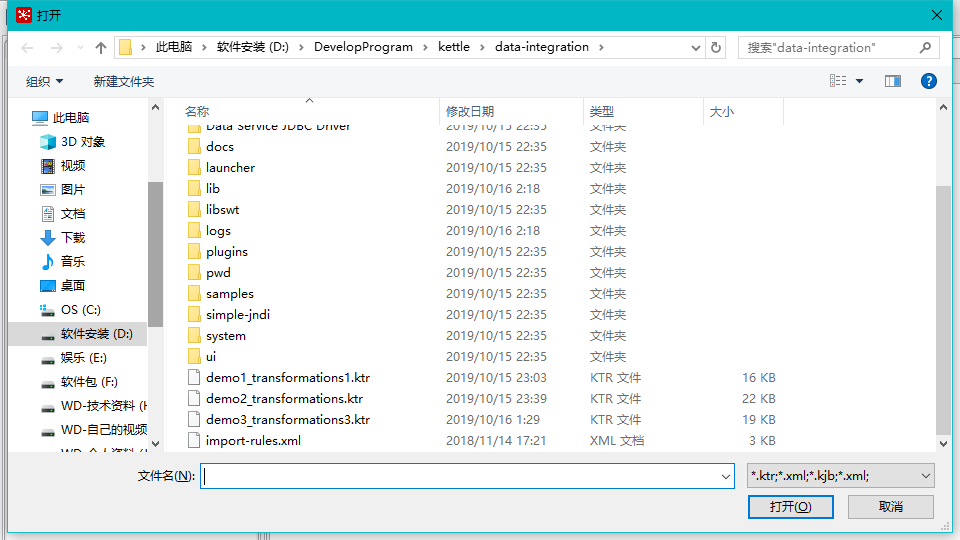
5. 连接资源库,默认账号密码为admin



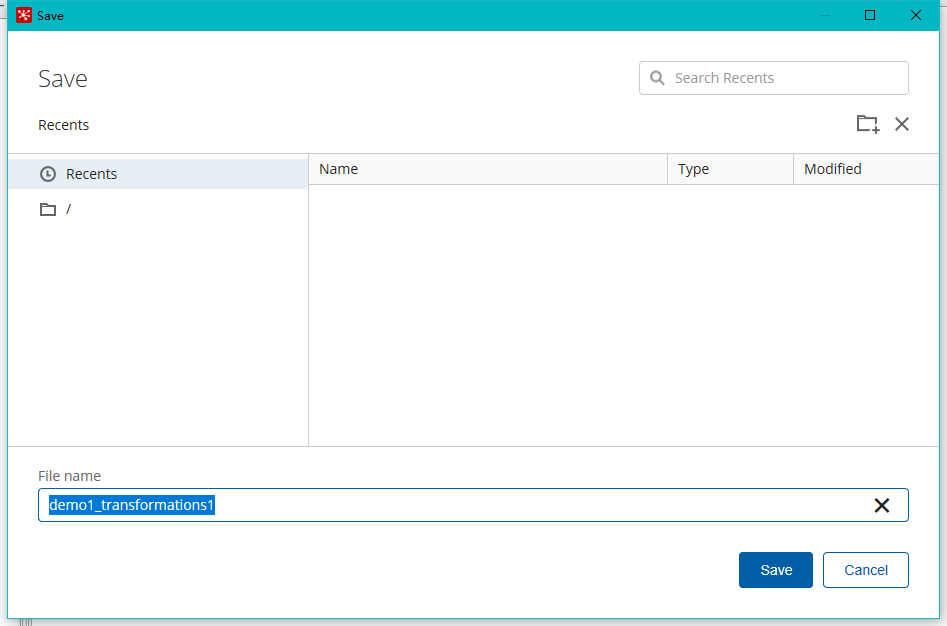
6. 将之前做过的转换导入资源库  
(1)选择从xml文件导入



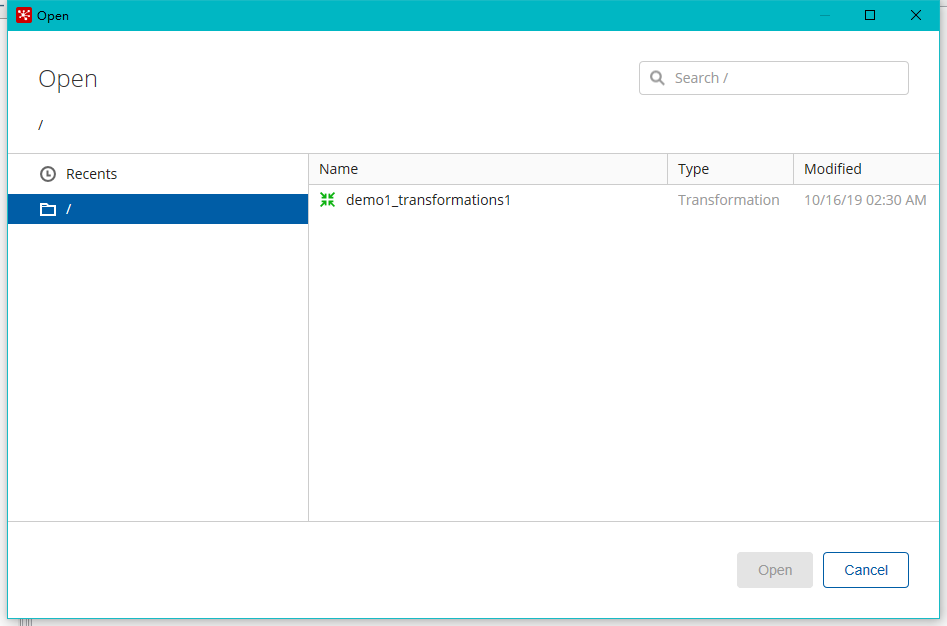
(2)随便选择一个转换



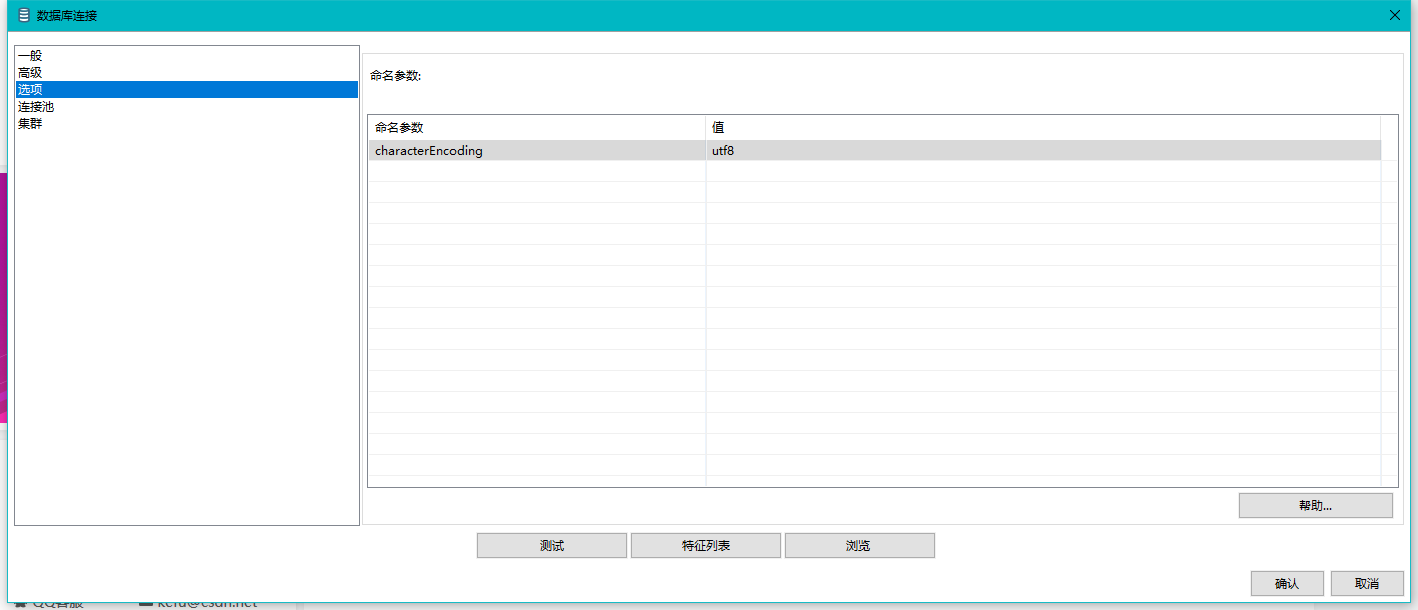
(3)点击保存，选择存储位置及文件名

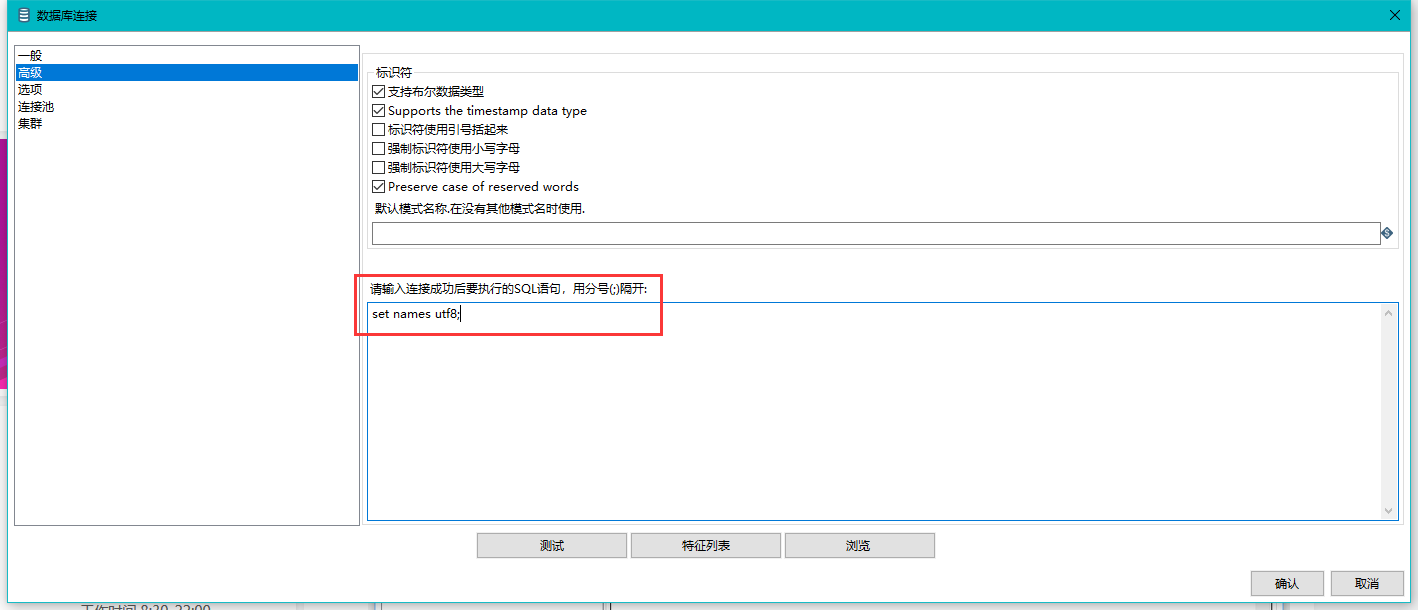


(4)打开资源库查看保存结果

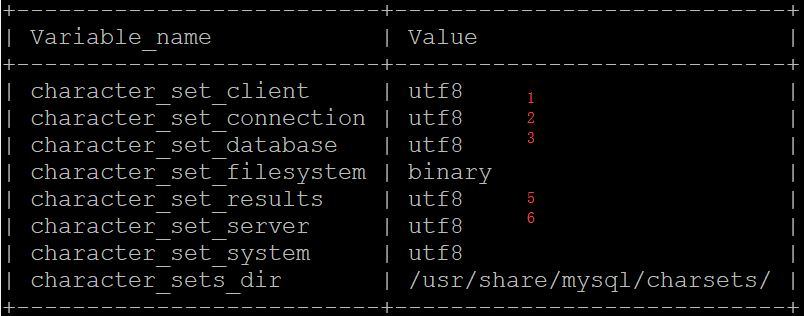


* 乱码问题





将作业或者转换信息存入mysql后，查询是发现是乱码的。这时可以修改mysql的编码  
1. 第一种方式如上图：  
  
2. 第二种方式如下：  
(1) service mysqld stop  
(2) 找到mysql的配置文件my.cnf（安装方式不同目录可能不同，我的在/etc/my.cnf），添加如下配置：  
character\_set\_server = utf8  
init\_connect='SET NAMES utf8'  
(3) 重启mysql服务，查看编码  
service mysqld restart  
mysql -uroot -p123456  
  
show variables like '%char%'

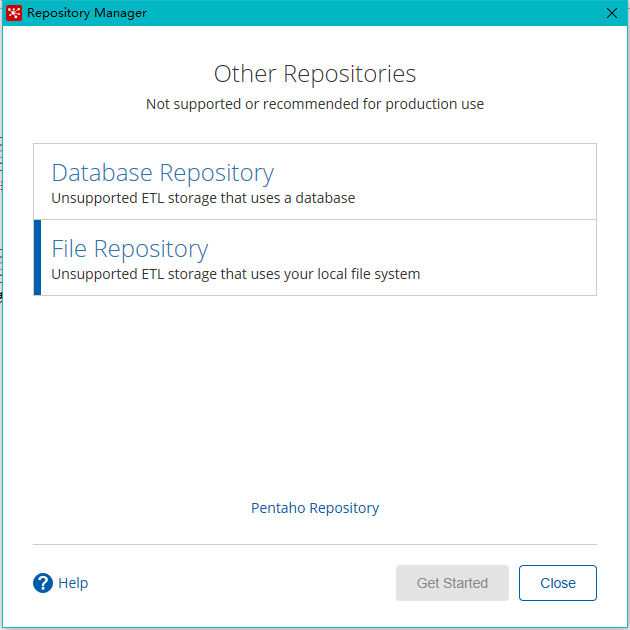


###### 3.3 文件资源库

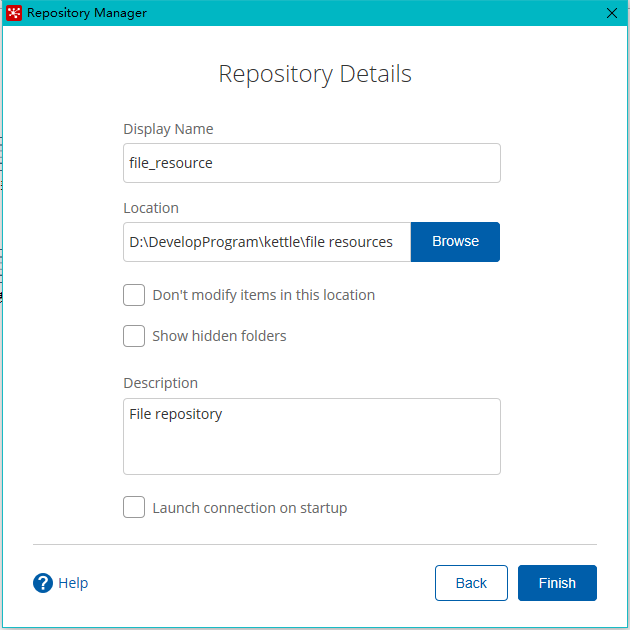
将作业和转换相关的信息存储在指定的目录中，其实和XML的方式一样。创建方式跟创建数据库资源库步骤类似，只是不需要用户密码就可以访问，跨平台使用比较麻烦

###### 3.4 文件资源库操作

1)选择connect,选择文件资源库



2)填写信息



#### 三 Linux下安装

##### 1 单机安装

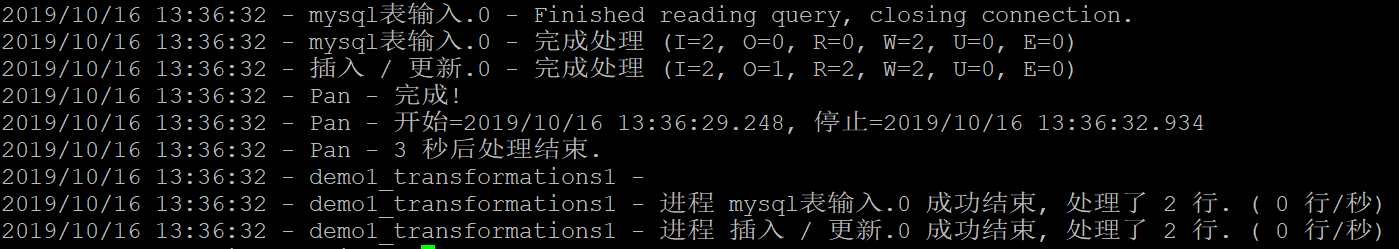
###### 1.1 安装jdk并且配置环境变量

###### 1.2 将安装包上传到服务器并解压(直接将windows中处理好的安装目录拷贝上去，不然得重新处理)

vi /etc/profile  
  
#java  
export JAVA\_HOME=/home/apps/jdk1.8.0\_45  
export CLASS\_PATH=$JAVA\_HOME/lib  
export HADOOP\_HOME=/home/apps/hadoop-2.6.0-cdh5.7.6  
export HADOOP\_PREFIX=$HADOOP\_HOME  
export HADOOP\_COMMON\_HOME=$HADOOP\_PREFIX  
export HADOOP\_CONF\_DIR=$HADOOP\_PREFIX/etc/hadoop  
export HADOOP\_HDFS\_HOME=$HADOOP\_PREFIX  
export HADOOP\_MAPRED\_HOME=$HADOOP\_PREFIX  
export HADOOP\_YARN\_HOME=$HADOOP\_PREFIX  
export HIVE\_HOME=/home/apps/hive-1.1.0-cdh5.7.6  
export ZK\_HOME=/home/apps/zookeeper-3.4.5-cdh5.7.6  
export HBASE\_HOME=/home/apps/hbase-1.2.0-cdh5.7.6  
export M2\_HOME=/home/apps/maven-3.3.9  
export ANT\_HOME=/home/apps/ant-1.8.1  
export KETTLE\_HOME=/home/apps/data-integration  
export PATH=$PATH:$JAVA\_HOME/bin:$HADOOP\_HOME/sbin:$HADOOP\_HOME/bin:$HIVE\_HOME/bin:$HBASE\_HOME/bin:$FLUME\_HOME/bin:$SQOOP\_HOME/bin:$OOZIE\_HOME/bin:$ZK\_HOME/bin:$M2\_HOME/bin:$ANT\_HOME/bin:$$KETTLE\_HOME  
  
tip:  
1. 把mysql驱动拷贝到lib目录下  
2. 将本地用户家目录下的隐藏目录C:\Users\自己用户名\.kettle，整个上传到linux的家宿主目录下

###### 1.3 运行数据库资源库中的转换

chmod +x pan.sh  
chmod +x spoon.sh  
  
./pan.sh -rep=mysql -user=admin -pass=admin -trans=demo1\_transformation1 -dir=/  
  
参数说明：  
-rep 资源库名称  
-user 资源库用户名  
-pass 资源库密码  
-trans 要启动的转换名称  
-dir 目录(不要忘了前缀 /)



###### 1.4 运行资源库里的作业

./kitchen.sh -rep=mysql -user=admin -pass=admin -job=demo2\_jobs -logfile=./logs/log.txt -dir=/  
  
-rep - 资源库名  
-user - 资源库用户名  
-pass – 资源库密码  
-job – job名  
-dir – job路径  
-logfile – 日志目录

