

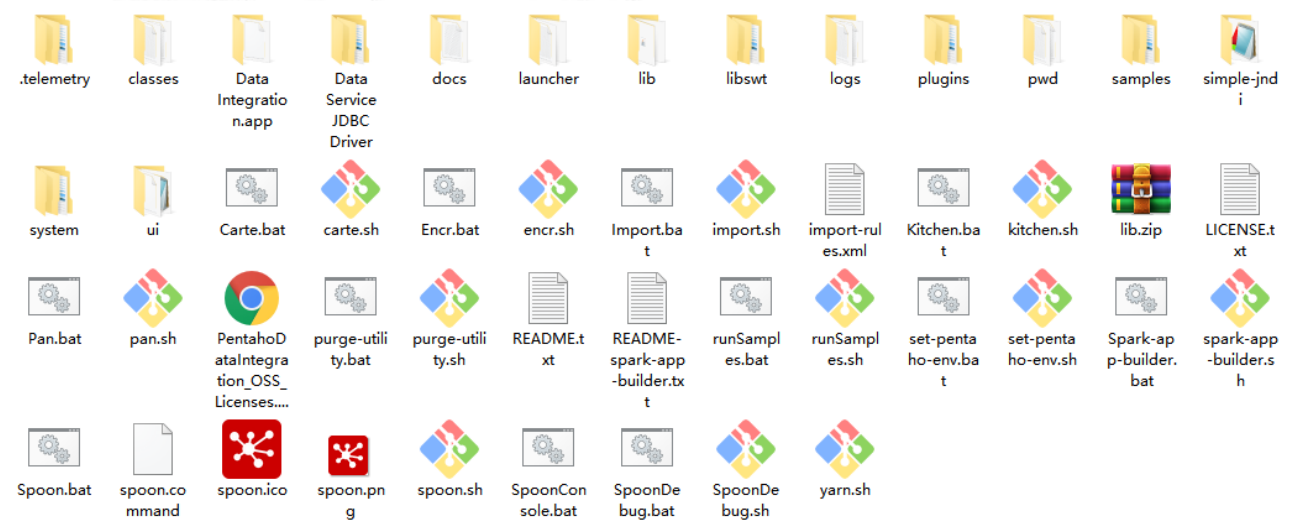
[kettle 中文网](#)

## 安装 kettle

去 [kettle 中文网](#) 下载 kettle 软件（他的官网经常崩溃，估计是没钱不维护了），下载完的解压包解压后，会发现里面有 \*.bat, \*.sh 脚本文件，你会发现，这是一个 windows 与 linux 通用的安装包



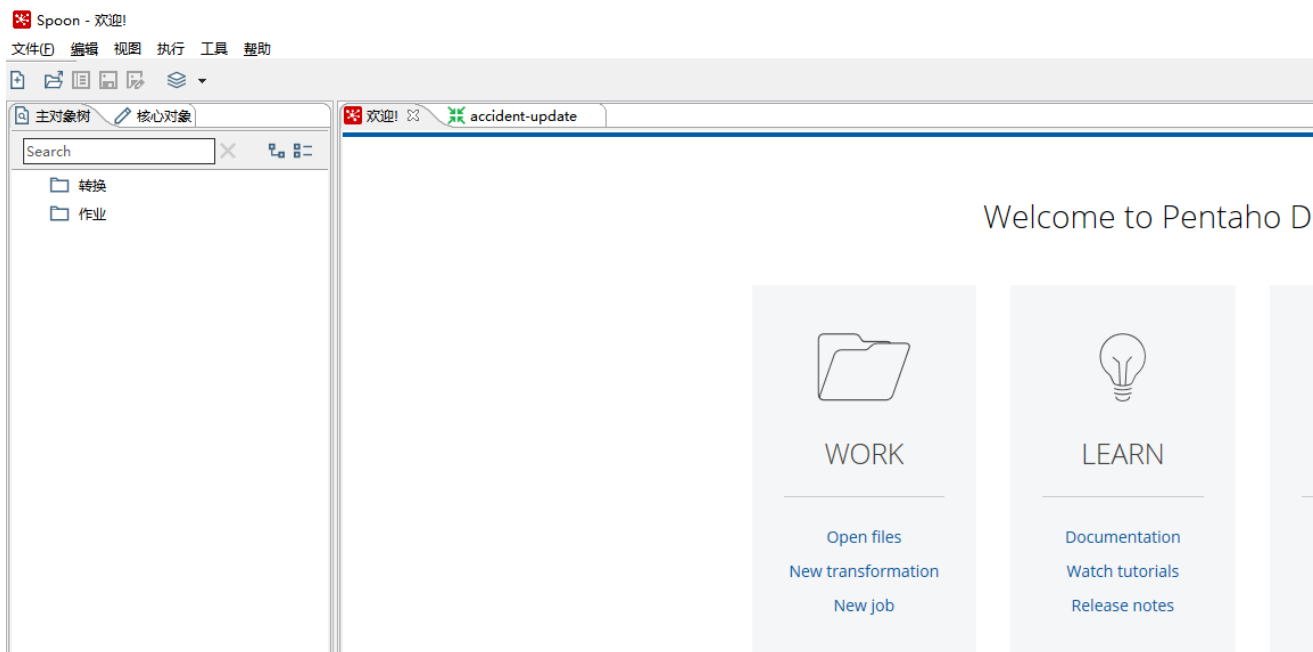
### 锈得老子心绞痛



解压完后，我们点击 `Spoon.bat` 就可以在 windows 下启动 kettle 了，推荐大家是先在 windows 下配置好任务后再上传到 linux 中，利用 linux 的定时任务去执行

## 创建转换任务

打开了 `kettle` 后，进入的页面是这样的

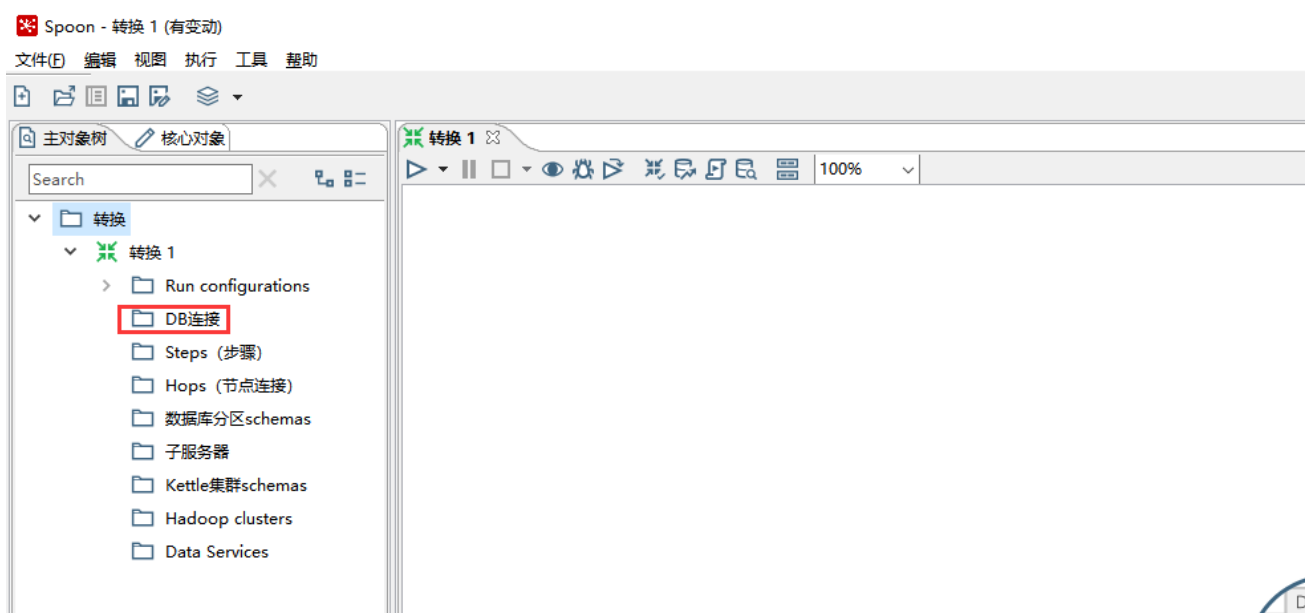


在左边的主对象树中，有两个对象，一个是转换，一个是作业，简单介绍一下，转换就是创建转换任务，作业呢就是在转换任务创建好后设置工作流之类的（例如定时器），我们在这里不需要用他的作业，所以只需要创建转换就好了，定时任务交给 `linux` 来做。

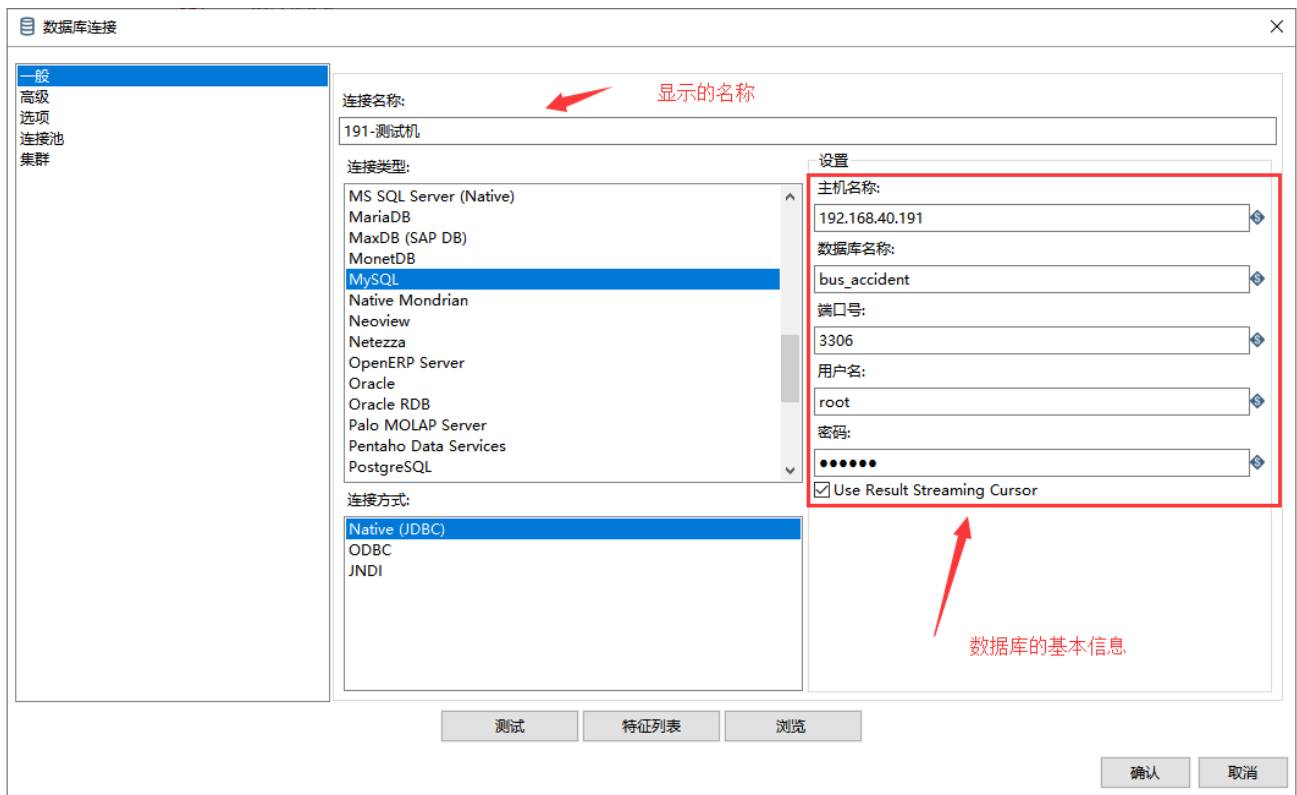
右键创建一个转换，创建完毕后会自动跳转到 `核心对象`，没有自动跳就自己点过去

## 添加数据库连接

`kettle` 获取数据与输出方式由多种，我们可以点击转换下拉框查看一下



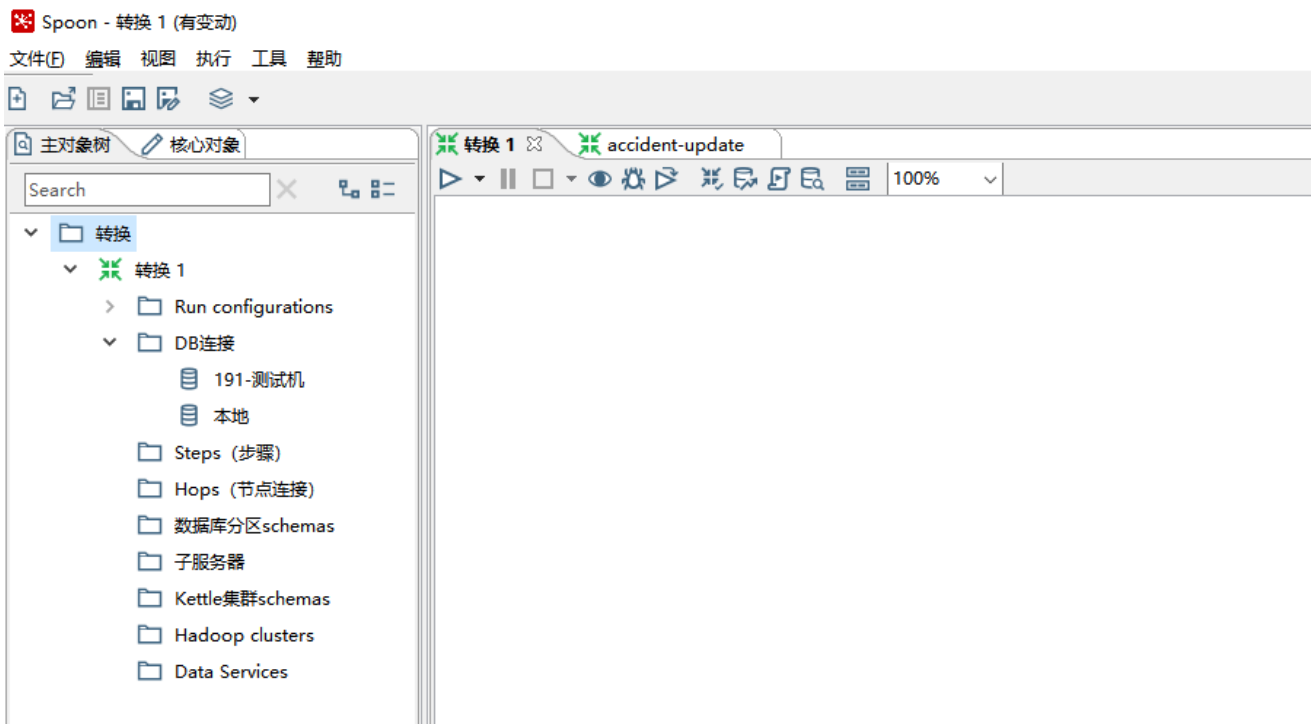
一般的转换都是从这个库到那个库的，所以我们需要创建两个 `DB连接`，我们双击它来看一哈



需要注意的是图上的所有信息都必须填写完整，若没有填写完它是不会让你确认的，填写完后点 **测试**，若能正确连接再确认

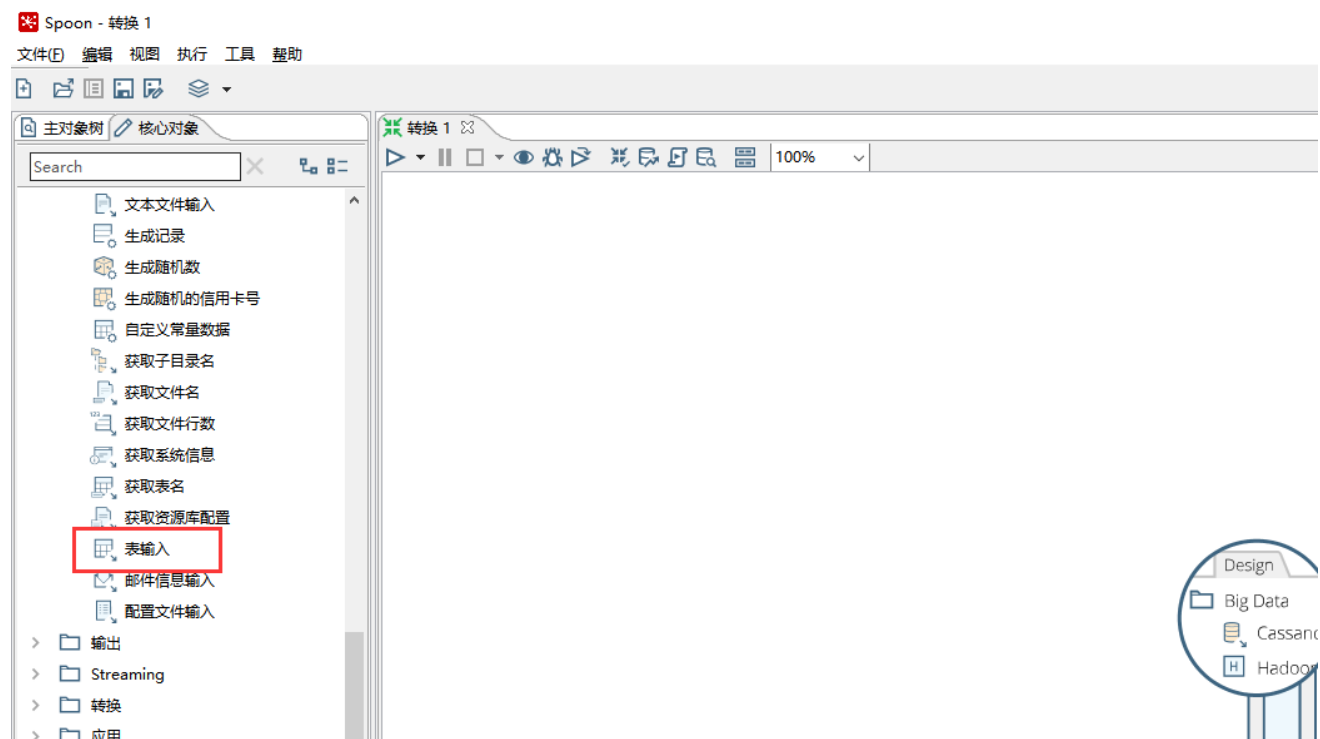
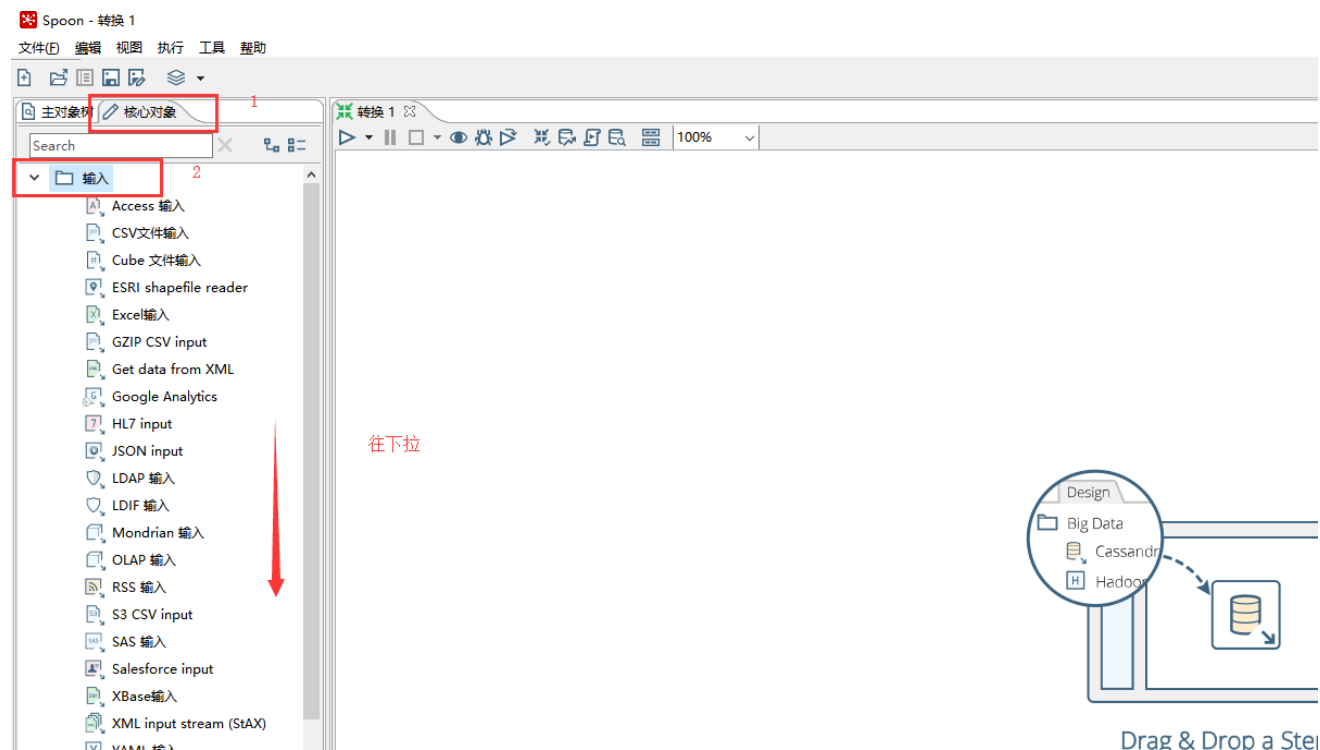
连接成功后，可能会出现写入数据出现乱码问题，这时候可以在被写入数据库连接中点击高级，设置连接参数 `characterEncoding` 为 `utf8`

我们需要从本机库同步数据到测试服务器，所以我们还需要添加本地连接进去，最后效果图

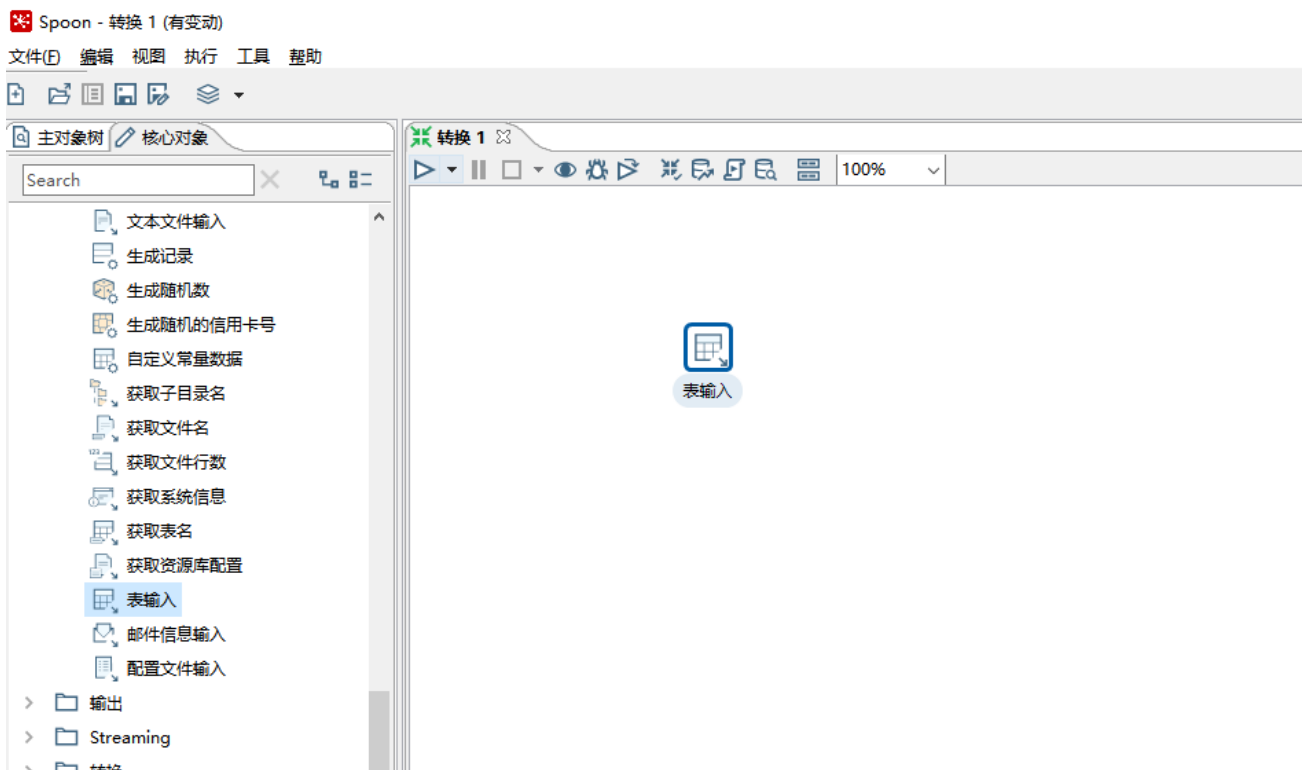


# 从数据源中获取数据

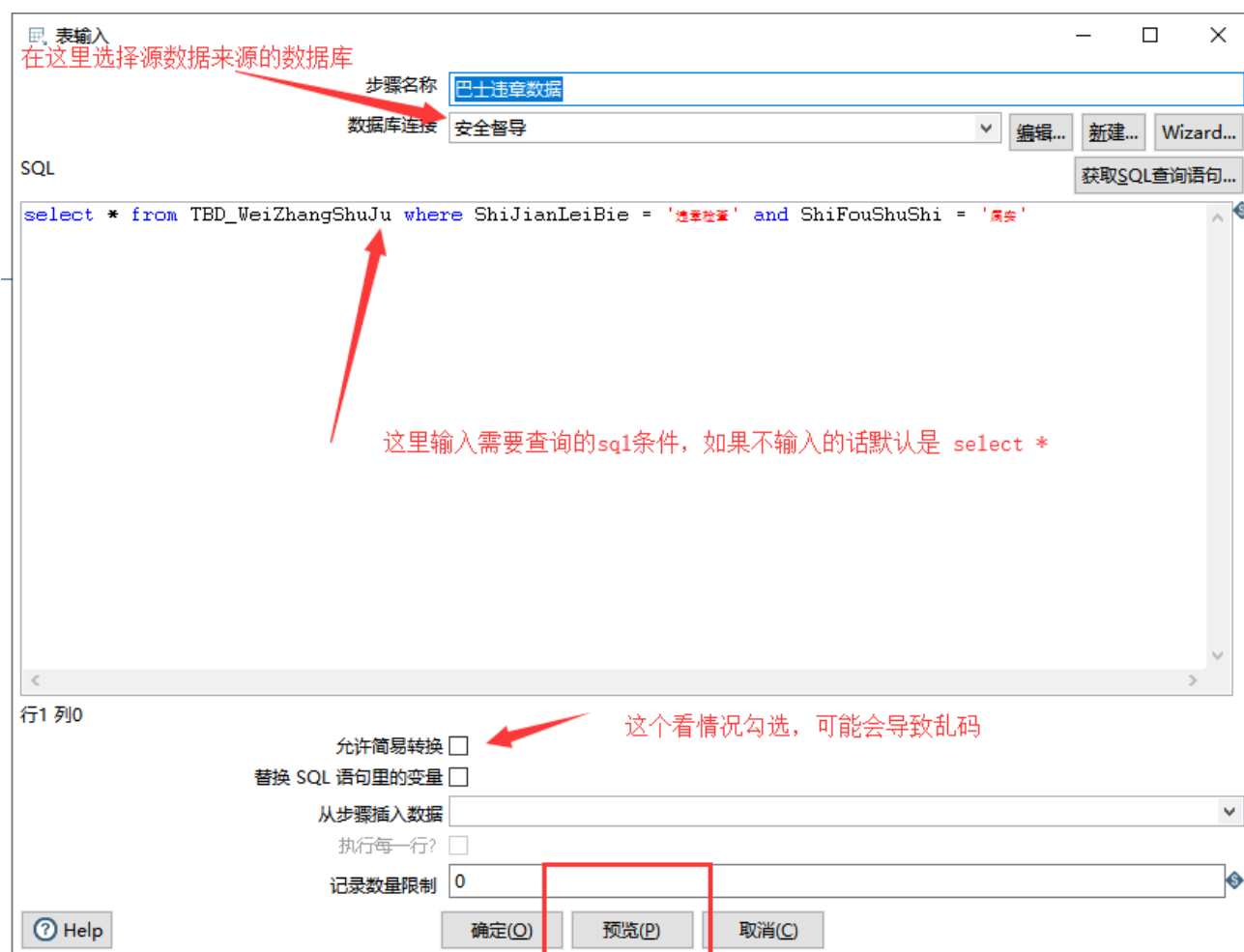
在 核心对象 里面点击 输入 下拉框，找到表输入



双击它后我们会在右边的大框框里看到一个表输入，双击后可以配置他



看一下我配置过的一个例子



需要注意的有3点：

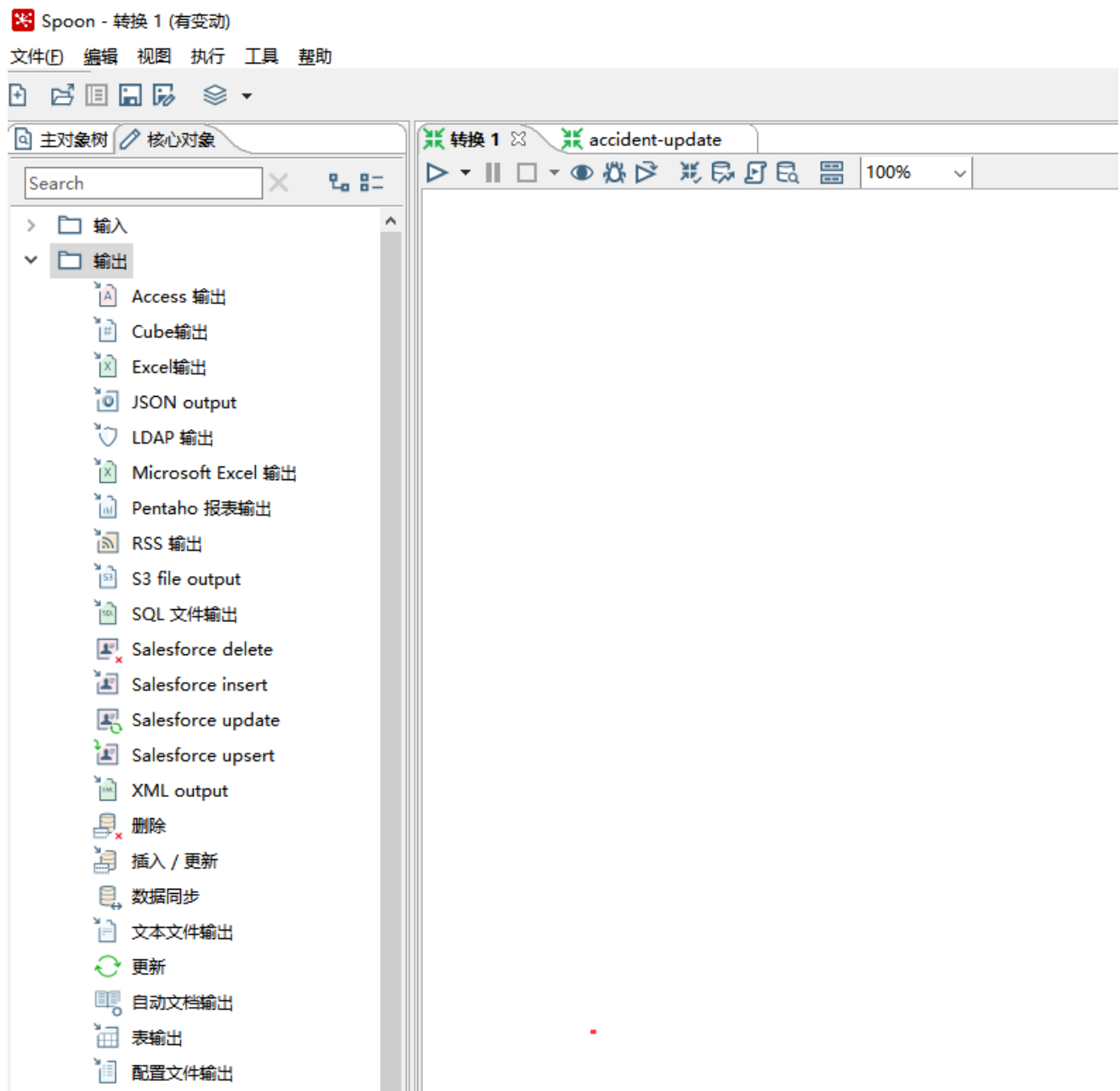
- 数据库选择不要选择错了，表输入中的数据库选择是源数据的数据库
- 中间的大 SQL 框可以不修改，默认是抓取全部数据，也可以点击 获取SQL查询语句 来生成 SQL
- 允许建议转换这个选项，网上有教程说需要勾选，不然会导致乱码，我测试的情况是勾了之后反而导致了乱码，具体需要看使用情况而定

配置好后点击 预览 按钮，若可以查询出数据则代表配置成功

这样，获取源数据就完成了

## 将获取的源数据保存起来

在 核心对象 中，除了有输入外，还有输出，我们先看一下输出列表中有哪些选项，他们有什么不一样



emm，一大堆看不懂的东西



富婆叫我了，告辞

我们都不要管它，表输出？是把刚获取的表输入直接整个输出出去，适合一次性使用的导表功能。

我们只要看 **插入/更新** 如果你需要删除的话，那就再看多一个 **删除** 吧

输出栏里还有一个 **数据同步** 选项，这个点进去会发现和 **插入/更新** 很像，具体的区别在哪，我查文档也没发现，这里我们也用不上

### 配置插入/更新

我们将一个插入更新拖到中间的版版中，双击看一下

插入/更新

步骤名称

插入 / 更新

1、存储数据的库

数据库连接

编辑...

新建...

Wizard...

目标模式

浏览(B)...

目标表

lookup table

浏览...

提交记录数量

100

2、这个值可以设定同步的速度

不执行任何更新:

☐

用来查询的关键字:

#	表字段	比较符	流里的字段1	流里的字段2
1				

获取字段

更新字段:

#	表字段	流字段	更新
1			

获取和更新字段

编辑映射

主要有4点内容，其中第四点是最麻烦的

- 先设置目标数据库与目标表，目标表需要在设置了目标数据库后点击浏览选择
- 提交记录量，这个值如果设置的太小就会造成多次插入，太大会担心连接中断
- 第三点是设置用来比较的字段，例如 ID，当 ID 存在的时候就更新，不存在的时候就插入
- 第四点是最复杂的一点，因为目标数据表可以和源数据表字段不一样，可以改名，改大小写，所以这里需要人工一一对应，感觉整个工作量都在这里了。

注意，有一个很关键的点！若是直接把这个控件拖到了白板中，那么他是和前面设定的 表输入 没有关联起来的，看有么有关联起来，就要看他们之间是否有一条箭头连线，若没有连线，在 插入/更新 中是无法匹配上字段的

附上一张完整的图



插入/更新

步骤名称

插入 / 更新

数据库连接

正式环境

编辑...

新建...

Wizard...

目标模式

浏览(B)...

目标表

driving\_record\_of\_violation\_bus

浏览...

提交记录数量

300

不执行任何更新:

☐

用来查询的关键字:

#	表字段	比较符	流里的字段1	流里的字段2	获取字段
1	serial_number	=	WeiZhangLiuShuiHao		

更新字段:

#	表字段	流字段	更新	获取和更新字段
1	entertime	entertime	Y	
2	owner_id	OwnerID	Y	
3	id	id	N	
4	create_date	create_date	N	
5	modify_date	modify_date	Y	
6	serial_number	WeiZhangLiuShuiHao	N	
7	affiliated_organization	SuoShuZuZhi	Y	
8	fleet	SuoShuCheDui	Y	
9	line	SuoShuXianLu	Y	
10	plate_number	WeiZhangCheHao	Y	
11	driver_name	WeiZhangRen	Y	
12	driver_code	WeiZhangRenGongZuoKaHao	Y	
13	violation_date	WeiZhangRiQi	Y	
14	violation_time	WeiZhangShiJian	Y	
15	violation_locale	WeiZhangDiDian	Y	
16	driving_direction	CheLiangHangShiFangXiang	Y	
17	violation_code	WeiZhangDaiMa	Y	
18	violation_behavior	WeiZhangHangWei	Y	
19	violation_describe	WeiZhangMiaoShu	Y	
20	source_of_investigation	ChaChuLaiYuan	Y	
21	investigation_time	JianChaShiJian	Y	
22	examiner_code	JianChaRenGongZuoKaHao	Y	
23	examiner	JianChaRen	Y	
24	division_time	FenGongShiJian	Y	
25	division_code	FenGongRenGongZuoKaHao	Y	
26	division_user	FenGongRen	Y	

Help

确定(O)

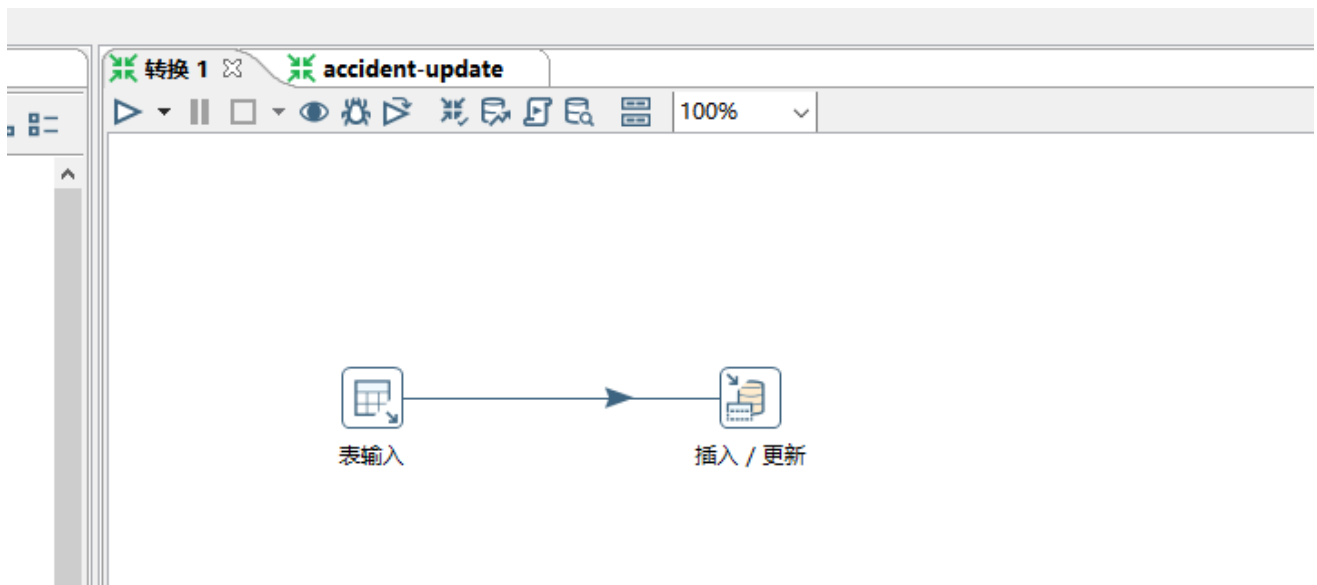
取消(C)

SQL

图中被我红圈圈起来的是标注该字段是否更新（这样子对那些接过来的数据又可以改的很友好啊）

## 下一步该干嘛

当我们设置好了输入输出后，来看一下大框框有了什么

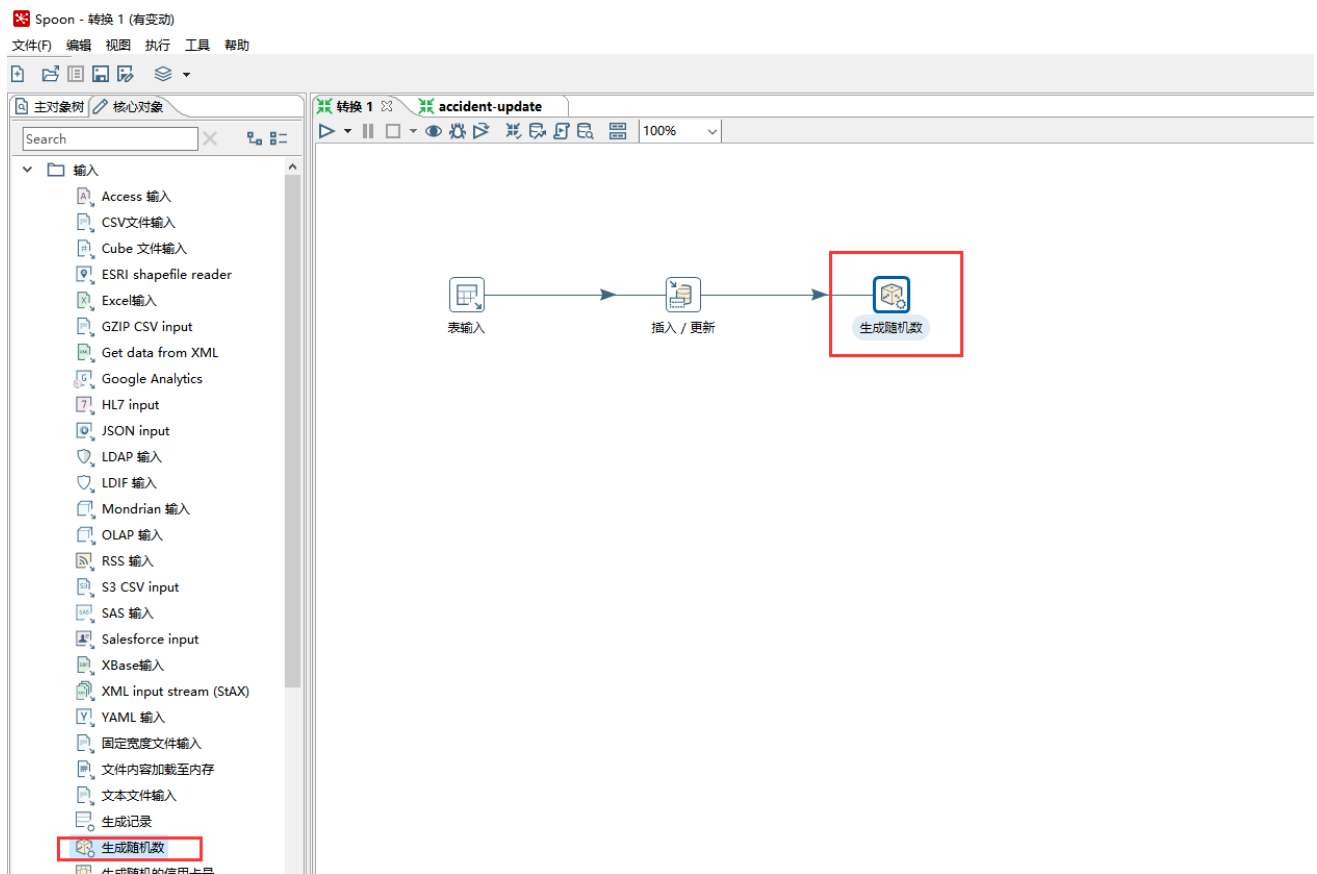


这样的時候已經是可以啟動轉換的了！不過我們好像，還漏了點什麼

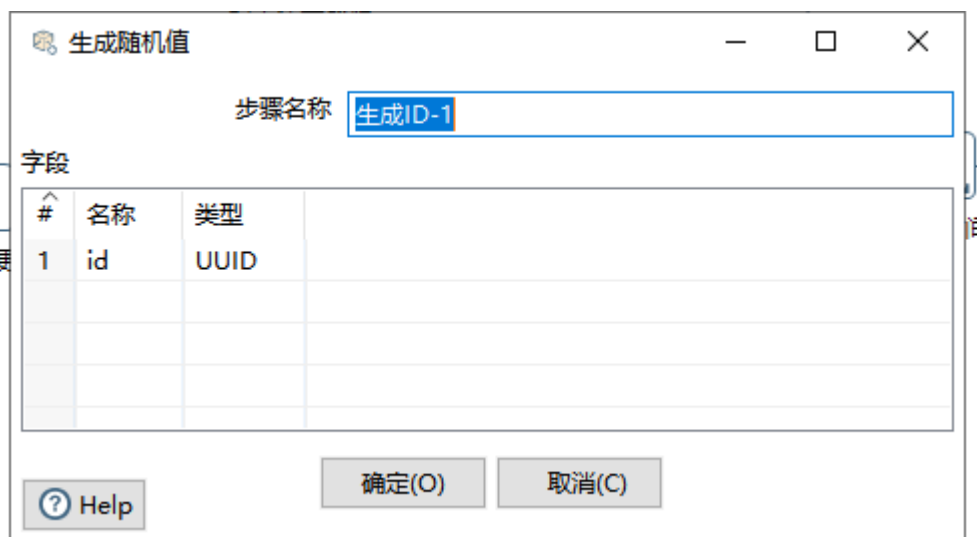
## 需要另外設置 ID 要怎麼辦？

我們獲取數據轉換的時候，可以將 ID 設置為跟源數據一樣的，這時候只要設置轉換就好了，那麼我們想配置新的 ID 呢，或者說源數據裡面用了某個很蠢的字段做了主鍵，我們想生成一個 UUID 怎麼辦？

我們回過頭看一下 輸入 裡面有一個 生成隨機數， 双击一下

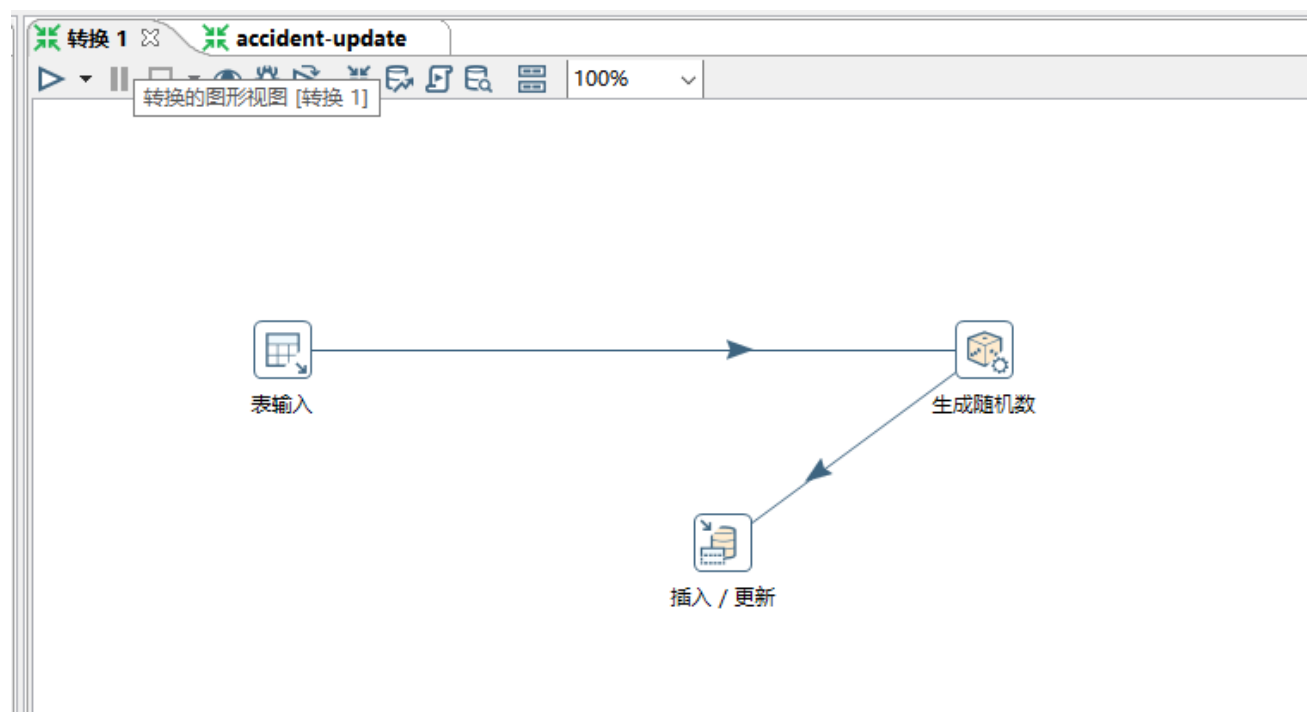


就是這個玩意，我們打開設置一下



因为都是 输入 选项，所以需要调整一下它的位置

创建连接线条是点击源组件 按着 shift 键，然后拉过去就好了

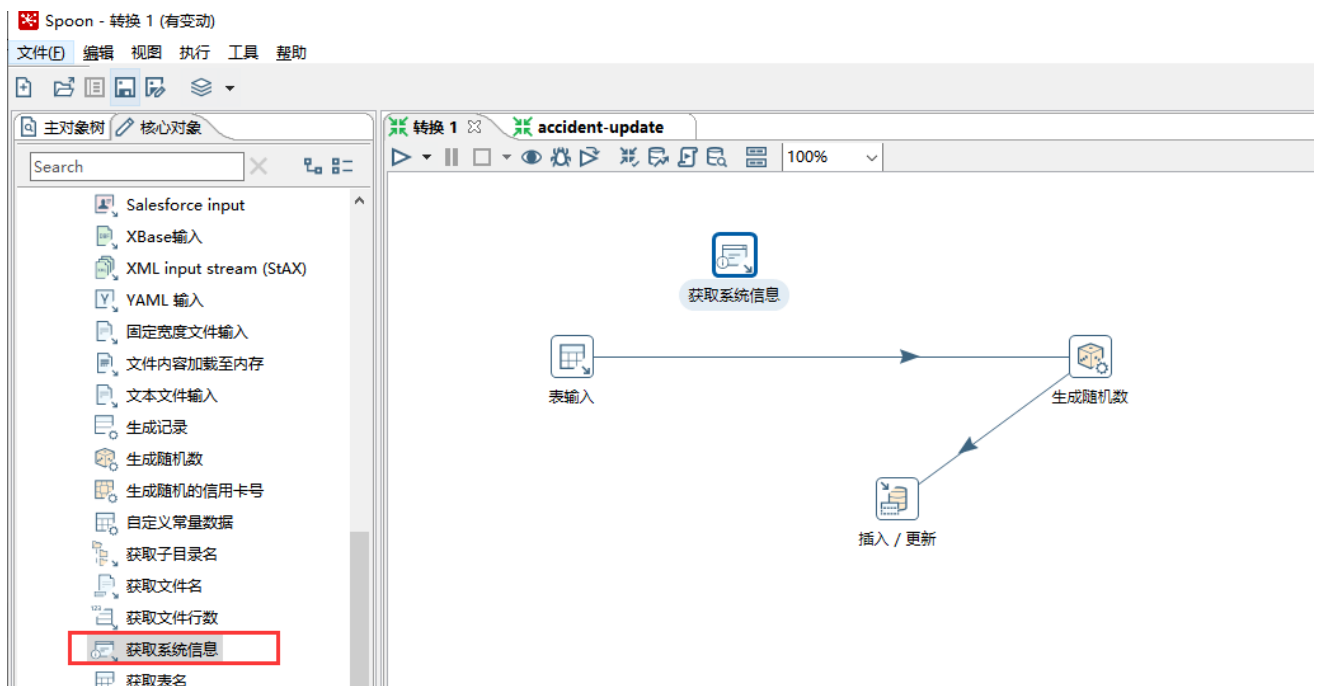


大功告成!

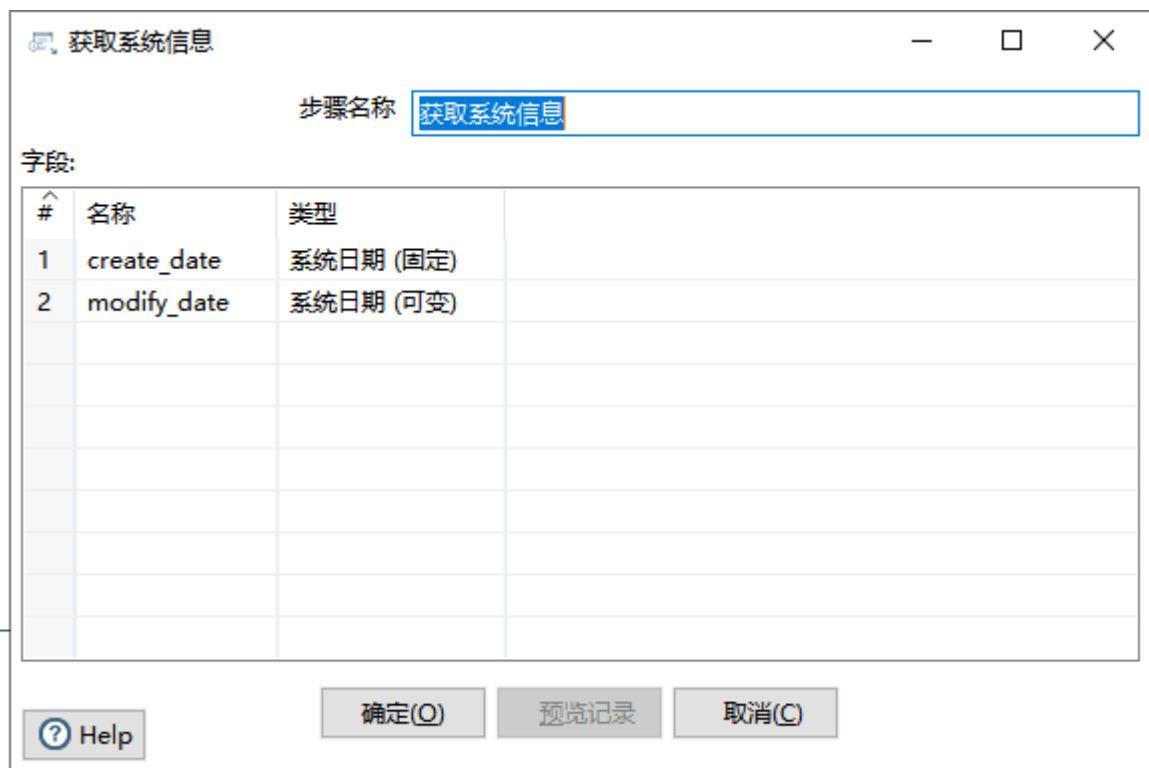
## 好像还是漏了点什么

按照 sibat 的编程习惯，每条数据都是由 `create_day` 和 `modify_date` 的，而数据源中不一定会有，刚面一个讲到的是一个随机数，那么时间怎么添加？

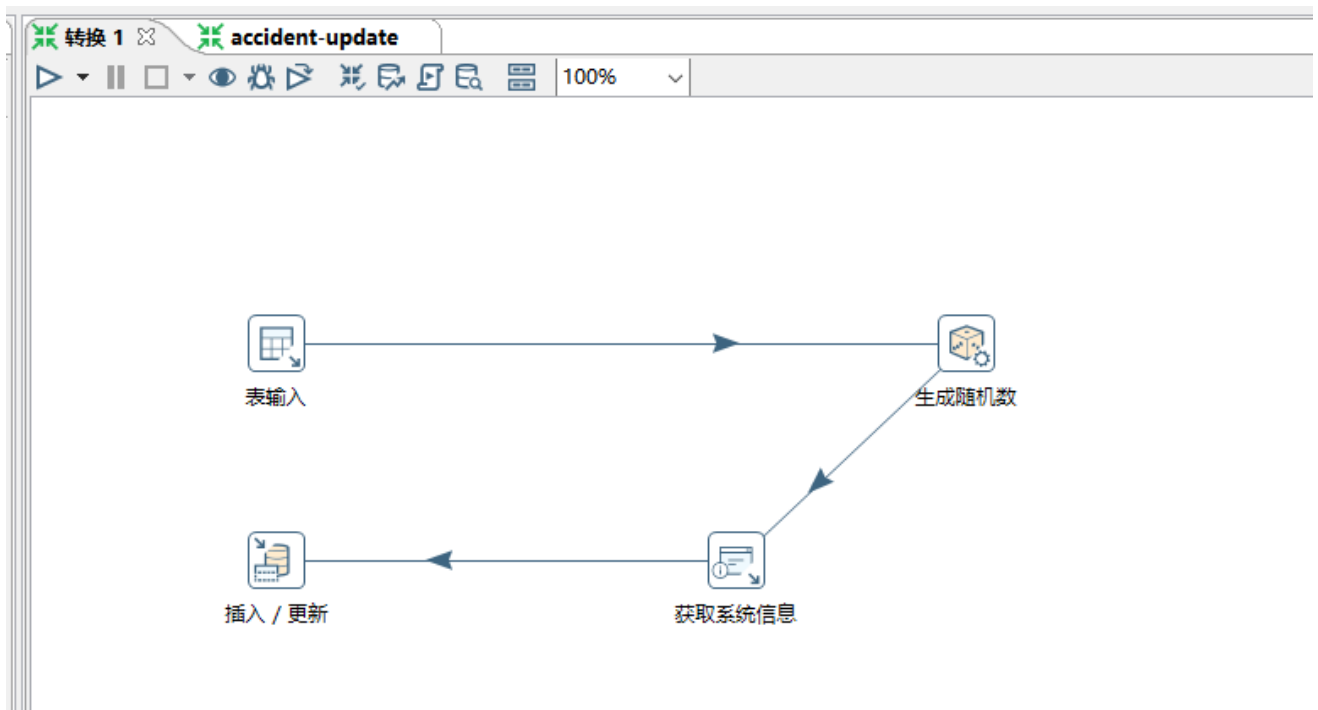
还有一个组件，也是在 输入 里面的 获取系统信息



我们来配置一下

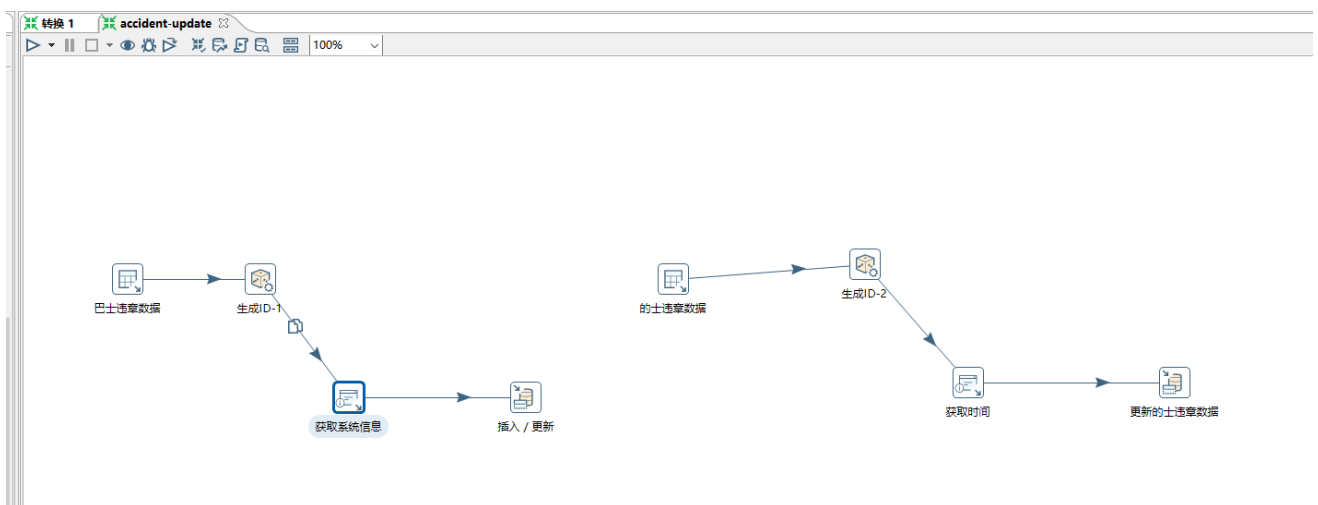


然后再调整一下顺序



这样，符合我们的一个转换流程就弄好了！

贴出一张正在运行的图



图中我们可以看出，多个转换可以放在一个面板了，他们是并发的

## 如何部署到 linux 并设置定时器

### 先把我们的软件包传到 linux 下，整个包，1点多G

上面我们讲到，在 windows 下，我们启动的是 Spoon.bat，在 linux 中，我们需要用到做定时任务的，不需要打开它的图形界面，要用到的指令是 pan.sh

我们将我们在 windows 下创建好的转换任务也上传到 linux 中，他的后缀名是 .ktr 文件

### 编写转换脚本

kettle 的转换需要用到 JDK 环境，所以 linux 需事先装好 JDK 并且记住他的环境变量

看一下我们的脚本

```
export JAVA_HOME=/usr/local/java/jdk1.8.0_181
export JRE_HOME=/usr/local/java/jdk1.8.0_181/jre
export CLASSPATH=.:$JAVA_HOME/lib:$JRE_HOME/lib:$CLASSPATH
# 这里开始时 kettle 的脚本，指定他的 pan.sh 目录
export PATH=$JAVA_HOME/bin:$JRE_HOME/bin:$PATH
/usr/local/tools/data-integration/pan.sh -file=/opt/kettle/ktr/转换.ktr >
/opt/kettle/ktr/log.log
```

然后将其放入定时器中

在 linux 下输入 crontab -e 查看定时任务 然后添加刚编写的 shell 脚本就好（这部分不会的童鞋百度去）

## 可能会出现的问题

下载的 kettle 可能会没有 mysql jdbc 的连接驱动包，需要手动下载放入安装目录的 lib 下，我使用的版本就没有