# 一、需求分析

多源数据模型存储通常用于处理来自不同来源的数据，例如HTTP、DBMS、DNS等，将这些不同类型的数据存储在统一的模型文件中，便于后续的识别和分析；

输出的日志类型用于分析存储在模型中的日志数据，并将其分类到相应的模块，例如敏感信息监测、业务访问行为分析等。

通过这种方式，可以有效地管理和利用多源日志数据，支持更深入的分析和决策；

# 设计方案

## 模型设计

设计一个通用的，可扩展的数据模型，可以分析存储来自多源数据的，输出到不同类型模块的模型。

## 日志分类与标签

在数据模型中增加“日志类型”和“源日志信息”字段标签,以便在分析时对日志进行分类，并清晰的知道需要存储到什么模块。

例如为其日志类型打上”敏感信息监测”或”业务访问行为分析”等标签；为源日志信息打上“HTTP”或“DBMS”等标签；支持后续的精准分析。

# 实现过程

## 流程图

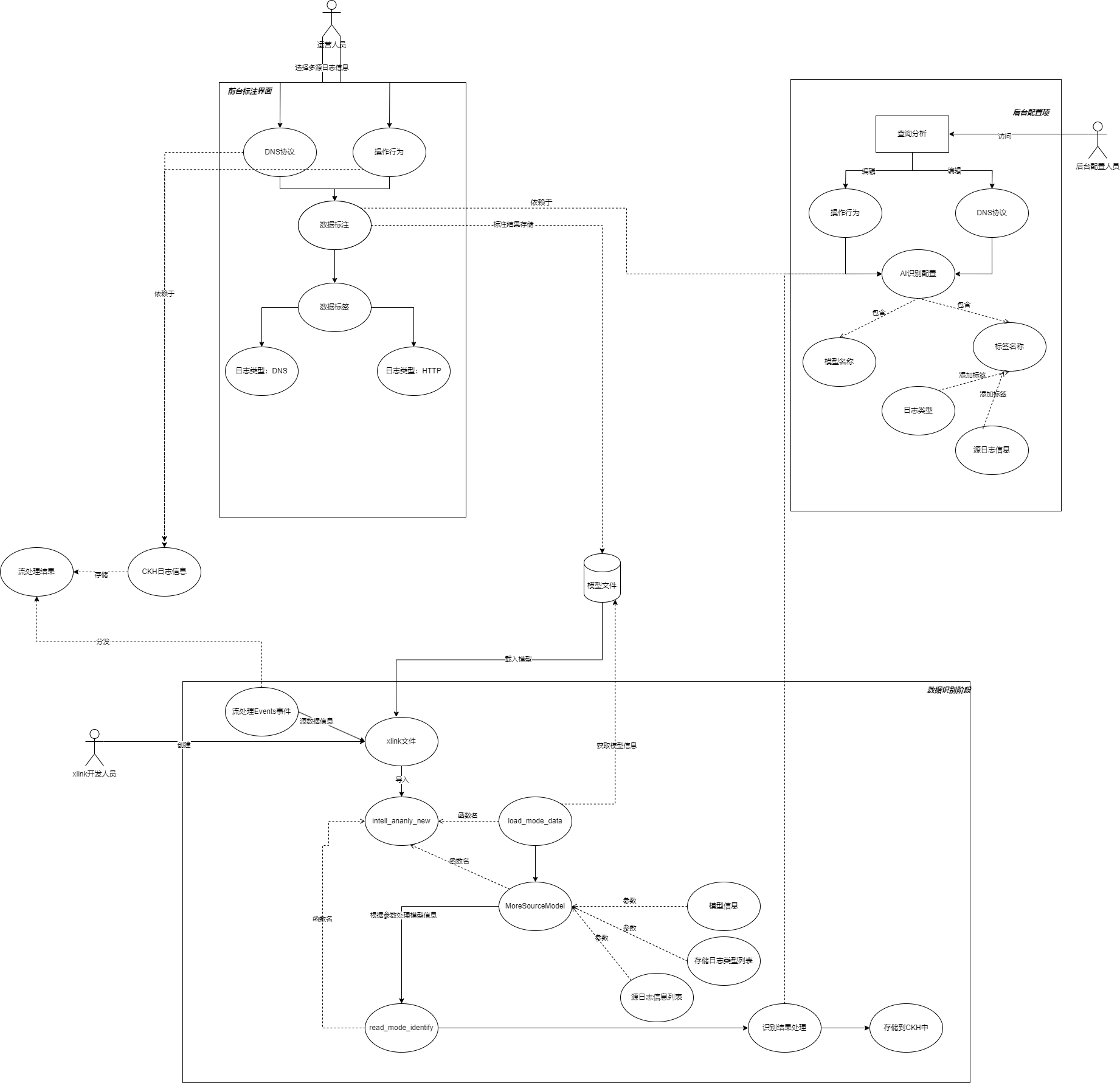
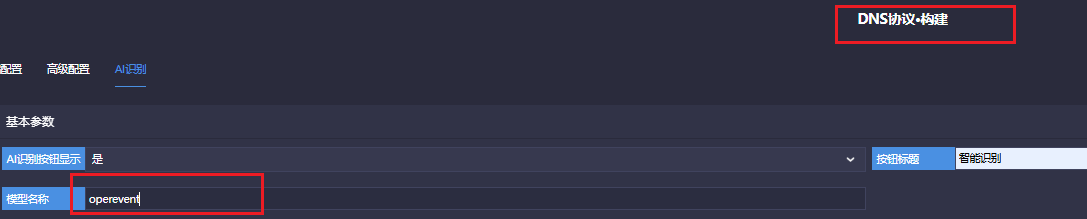


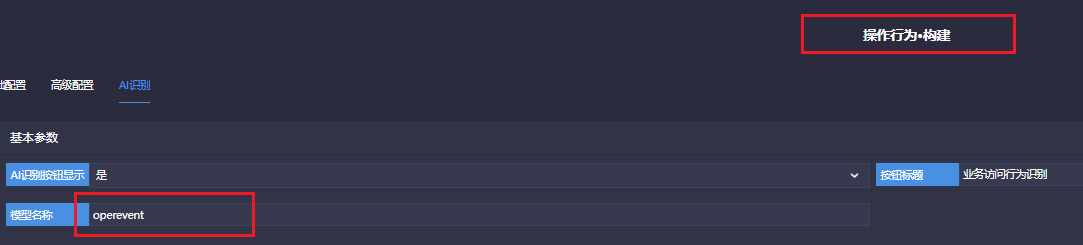
图1-1多源模型流程图

## 2.配置信息

① 为需要进行模型识别的模块，添加 AI识别->模型名称(名称相同)

例如：需要对操作行为和DNS协议进行识别，只需形成一个模型名称 “operevent”

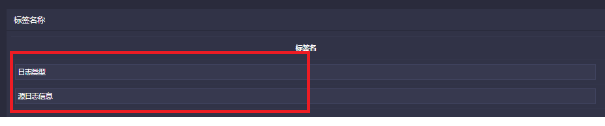




② 添加标签名 AI识别->标签名称

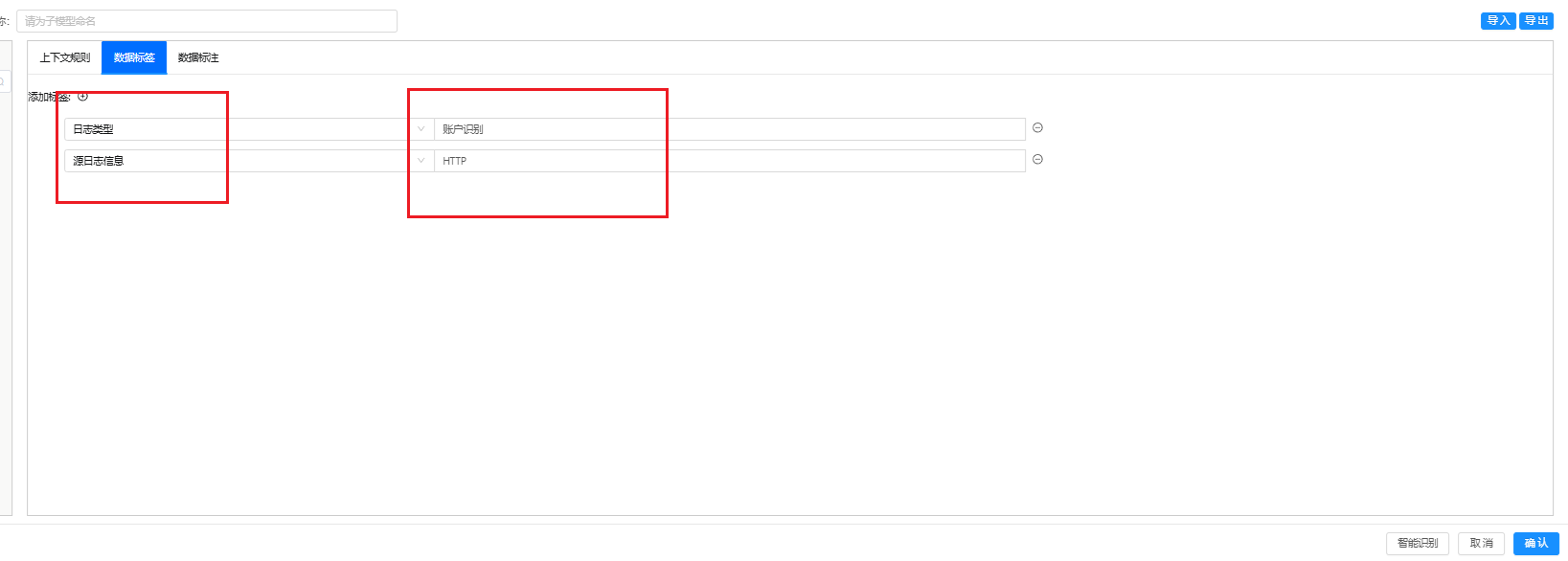
日志类型：表示模型识别后日志信息需要存储到什么模块（例如现有的敏感监测，业务访问行为）

源日志信息：表示日志信息来源于什么协议（HTTP,DNS）



③ 前台标注

针对当前日志信息，可以给出日志类型和源日志信息，由下图可见，分析出来的数据是要存储到账户识别模块，源数据来源与HTTP协议



## 3.Xlink处理模型数据

### ①导入函数



### ② 初始化载入模型数据



Outcome\_log : 输出识别结果存储位置信息(例如：想要将识别结果存储到敏感监测中)

Source\_log: 多源日志数据

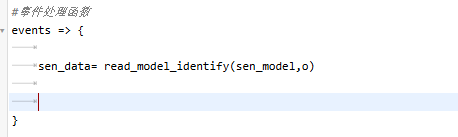
载入模型数据分四种情况：

#### 无输出日志类型、无源日志类型 默认(二者为空，输出全部子模型)

##### 初始化



##### Events处理

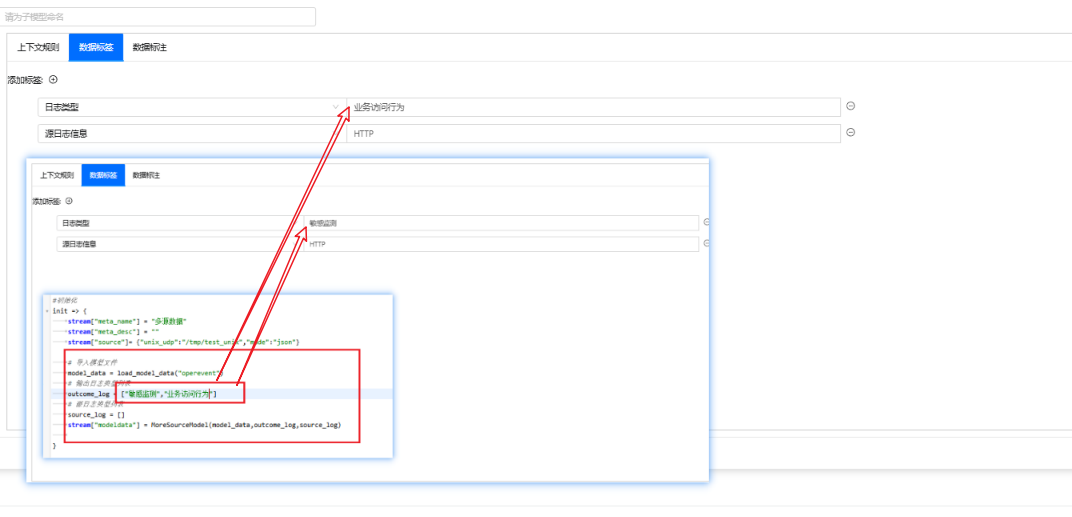


#### 有输出日志类型，无源日志类型

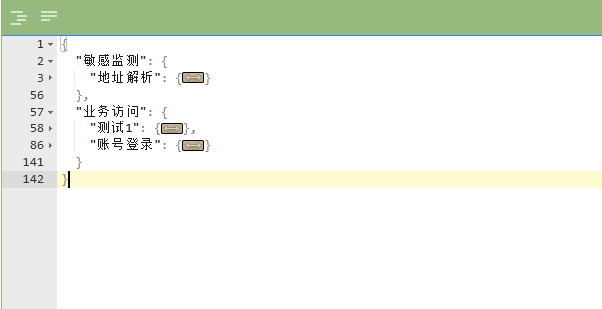
##### 初始化



##### 前台标签标注



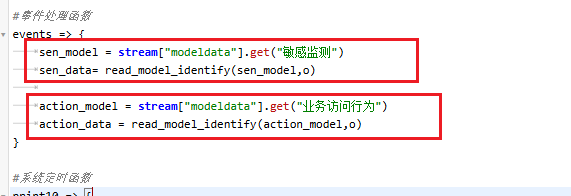
##### 获取结果



##### Events处理

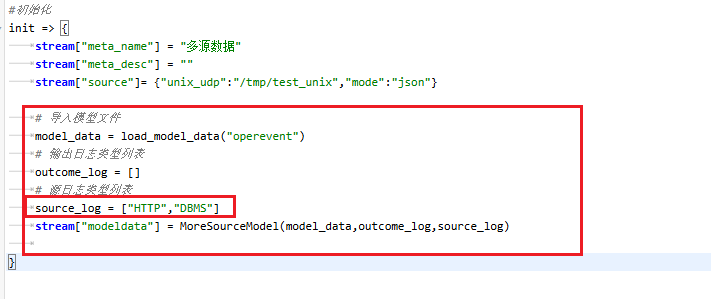
若需要在同一文件中进行存储多模块数据，就需要获取不同存储位置的子模型：

分两个模型识别分别存储到不同模块当中；

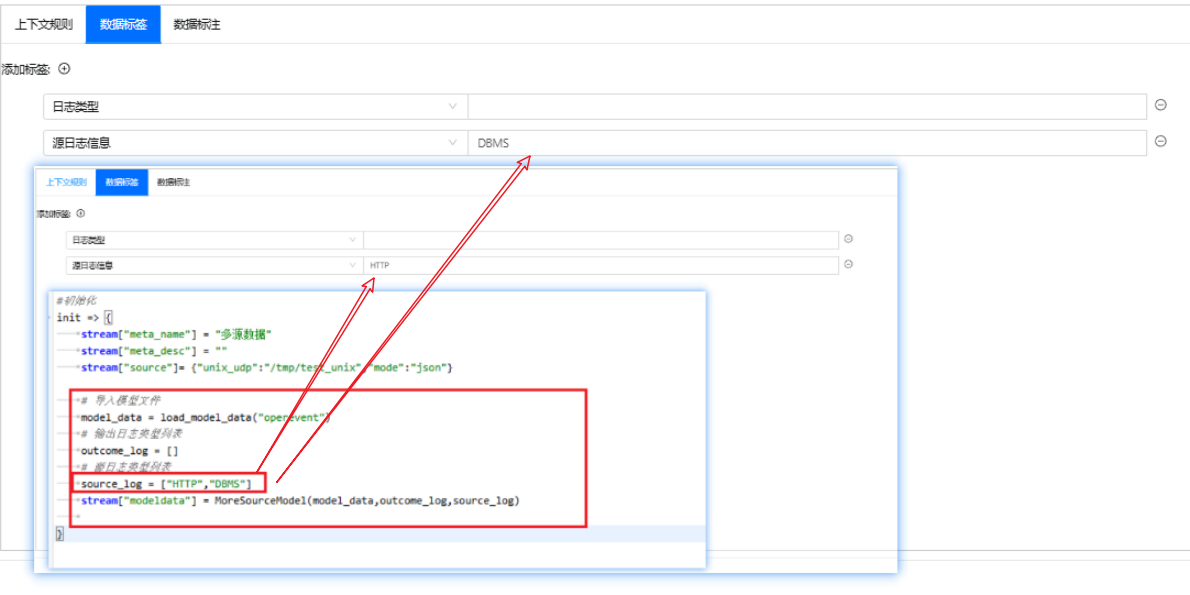


#### 无输出日志类型，有源日志类型

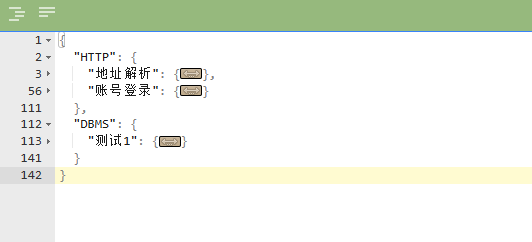
##### 初始化



##### 前台标签标注

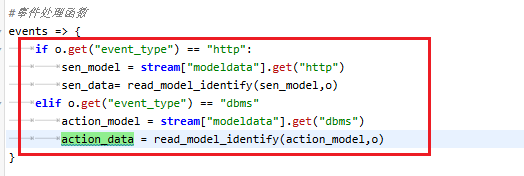


##### 获取结果



##### Events处理

需要对多源日志进行处理，需要在Events中进行源日志判断，并且获取相应的模型数据并进行模型识别分析；



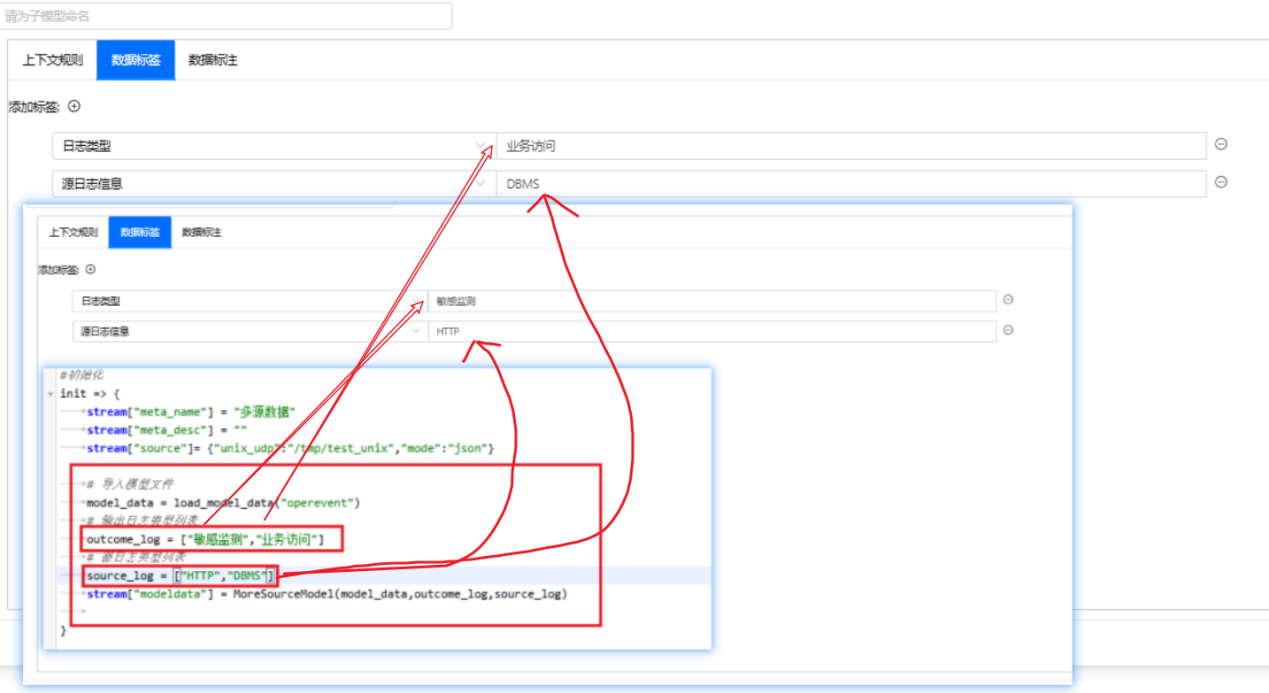
#### 4）有输出日志类型，有源日志类型

该情况属于合并状态，二者必须同时存在才能输出模型文件

##### 初始化



##### 前台标签标注



##### 获取结果

就会出现嵌套结果 源数据为第一层，第二层为需要存储的模块，第三层为模型信息



##### Events处理



# 总结

通过标签多源数据和输出日志类型，可以处理不同类型不同场景下同一模型文件识别分析存储问题