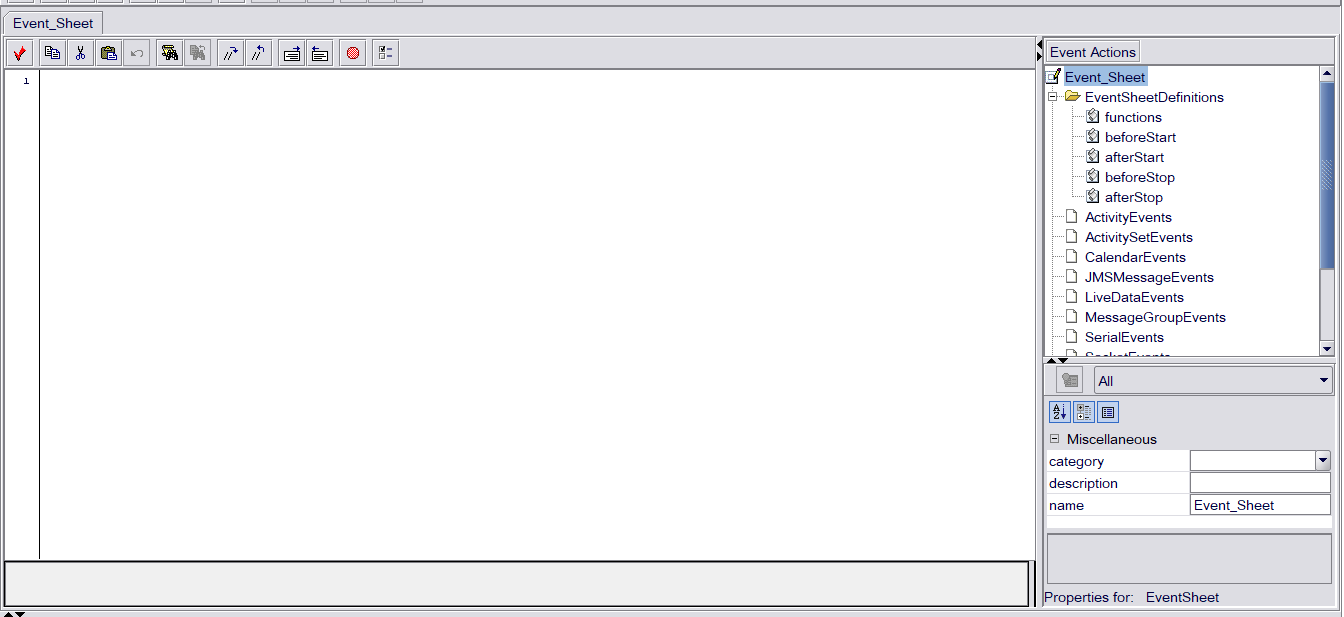
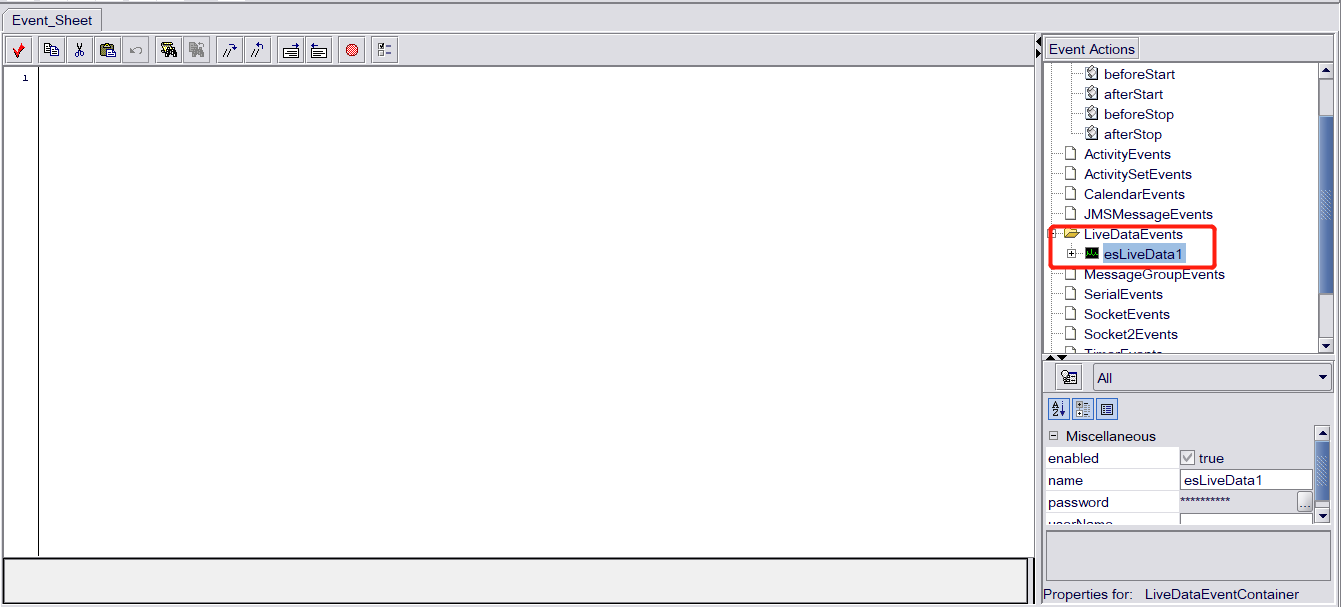
**普通模式**

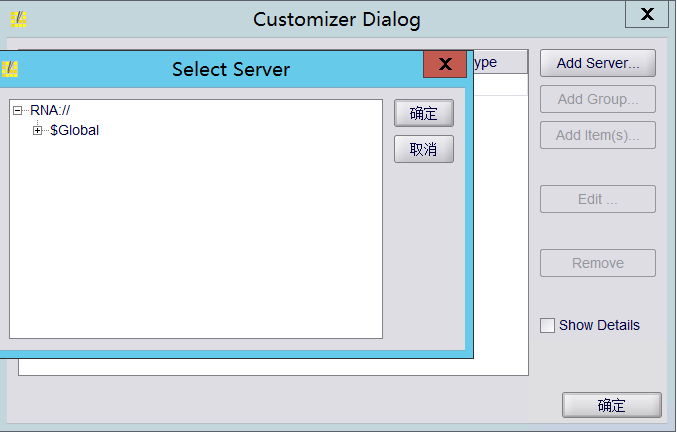
1:创建EvenSheets



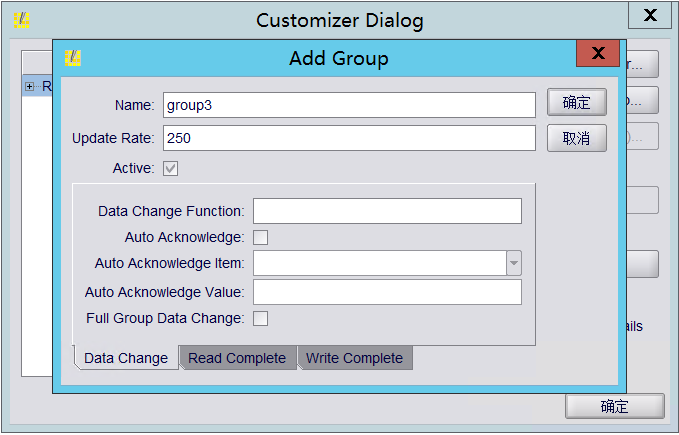
创建LiveDate



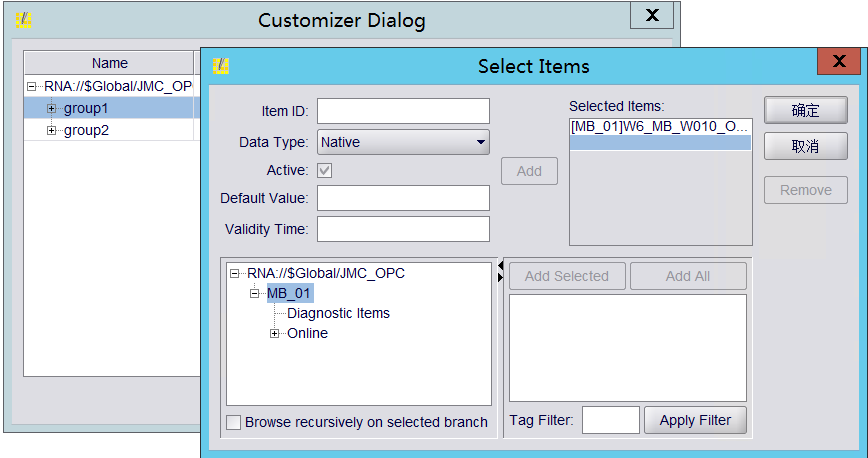
创建service



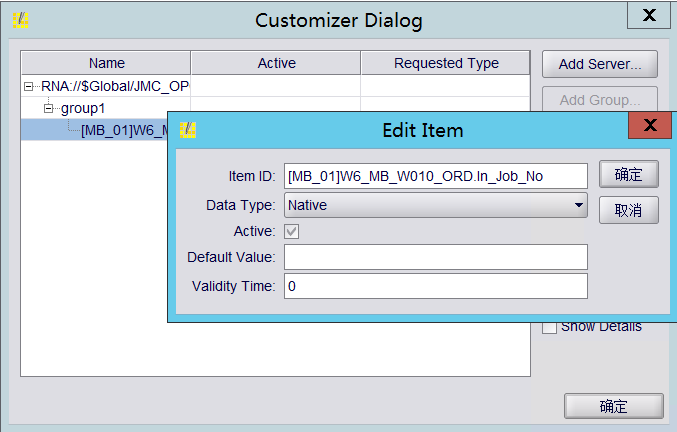
创建Group：如果需要有DateChange 方法 则需要把Active方法勾上



选择Items （Tage）



查看点位

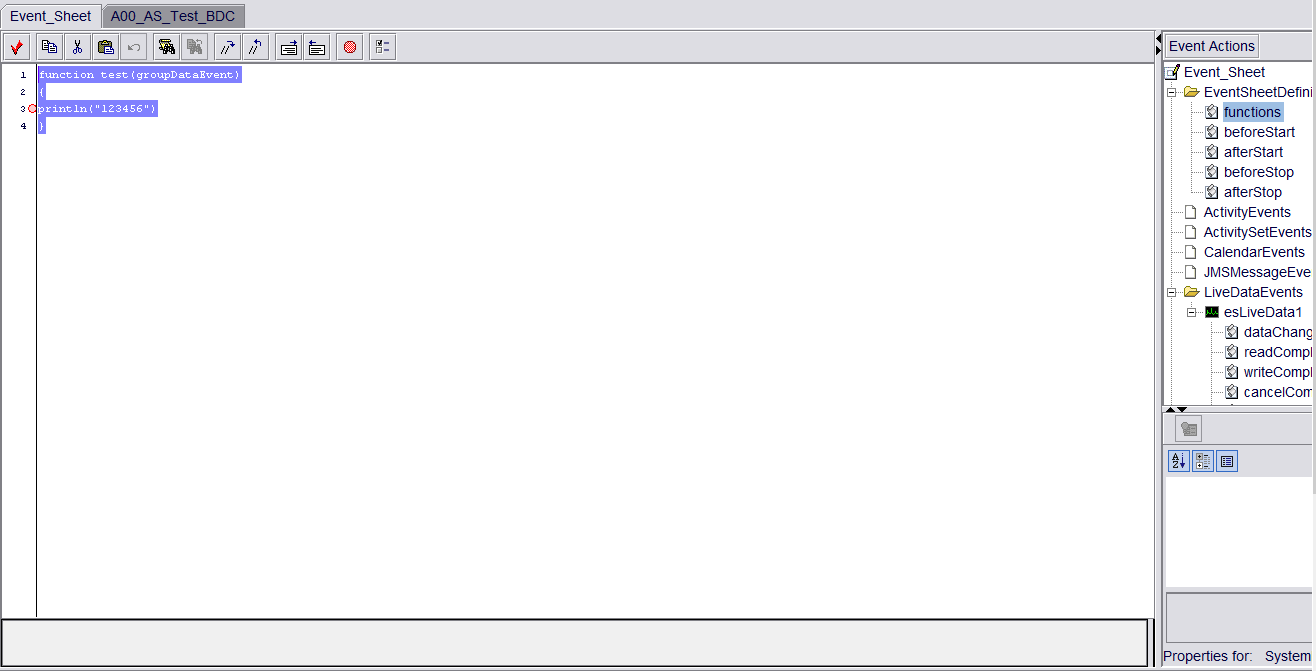


编写DateChange 方法：

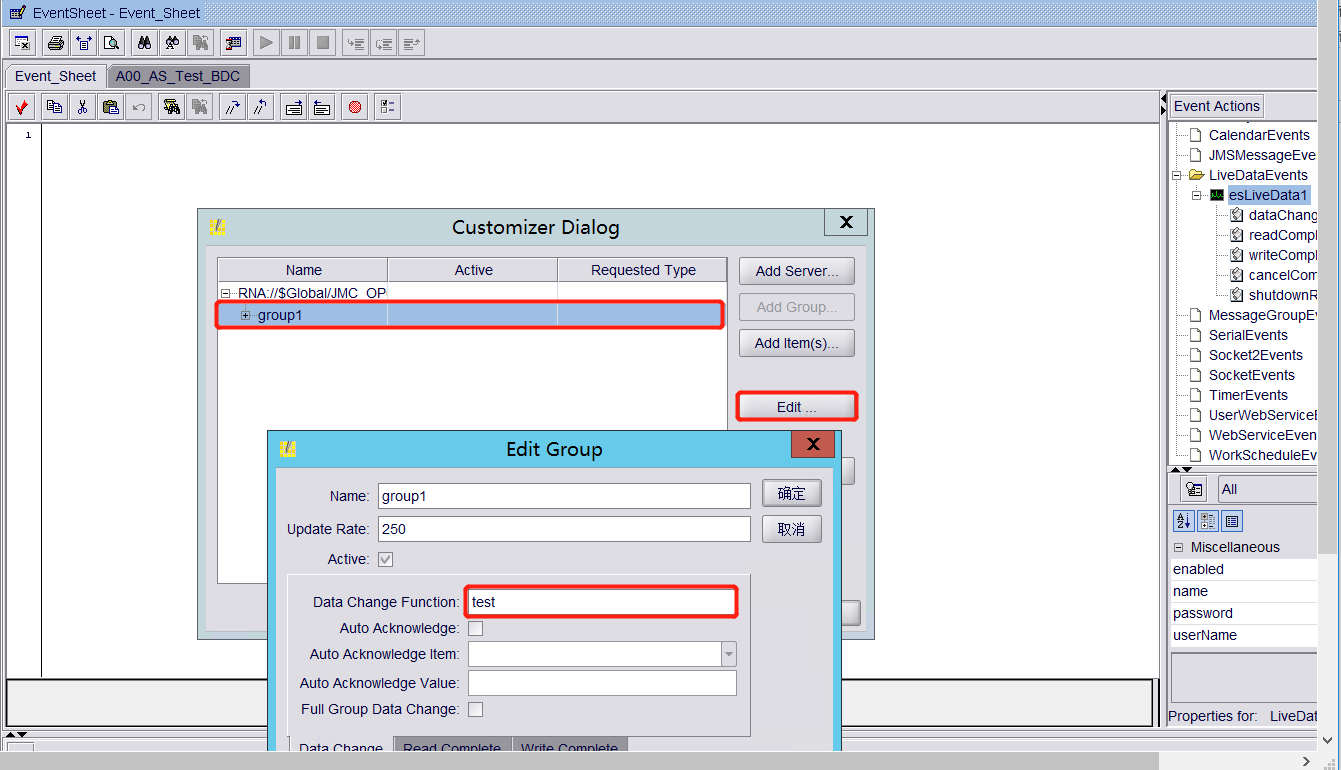
function test(groupDataEvent)

{

println("123456")

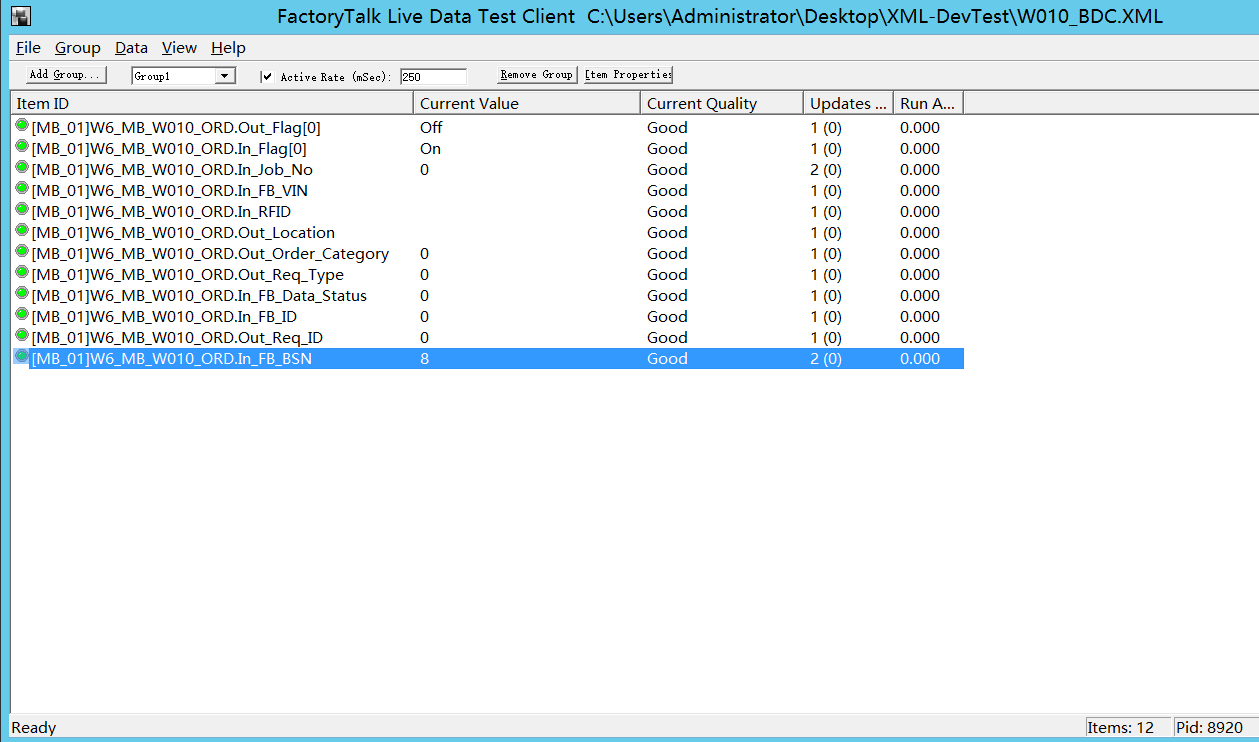
}

方法名与Group的Active方法名对应

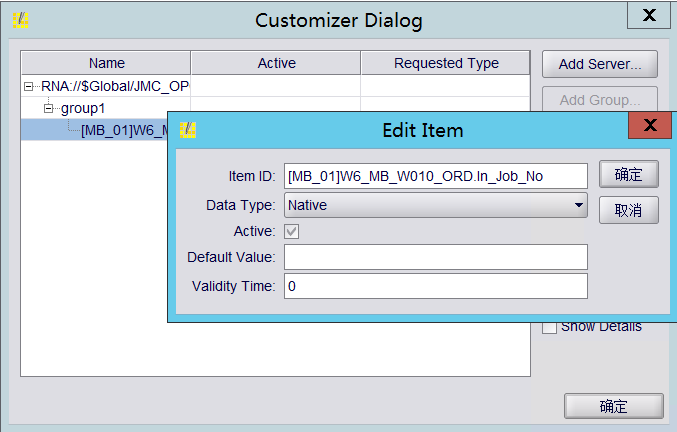


**测试：**

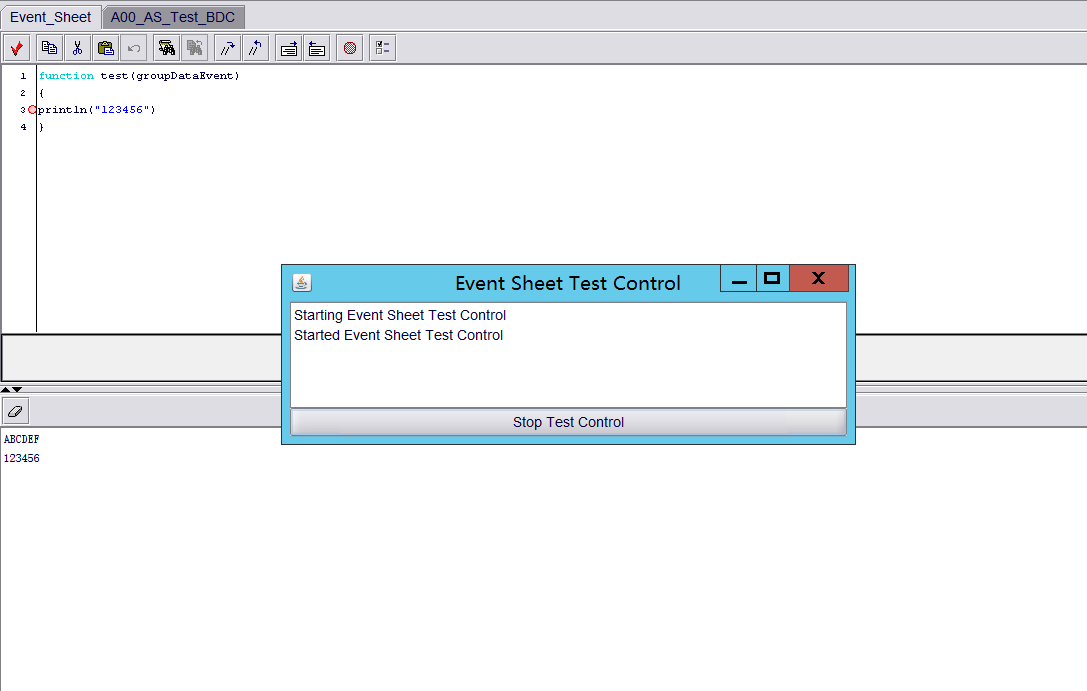
1：启动该站点的监控信息



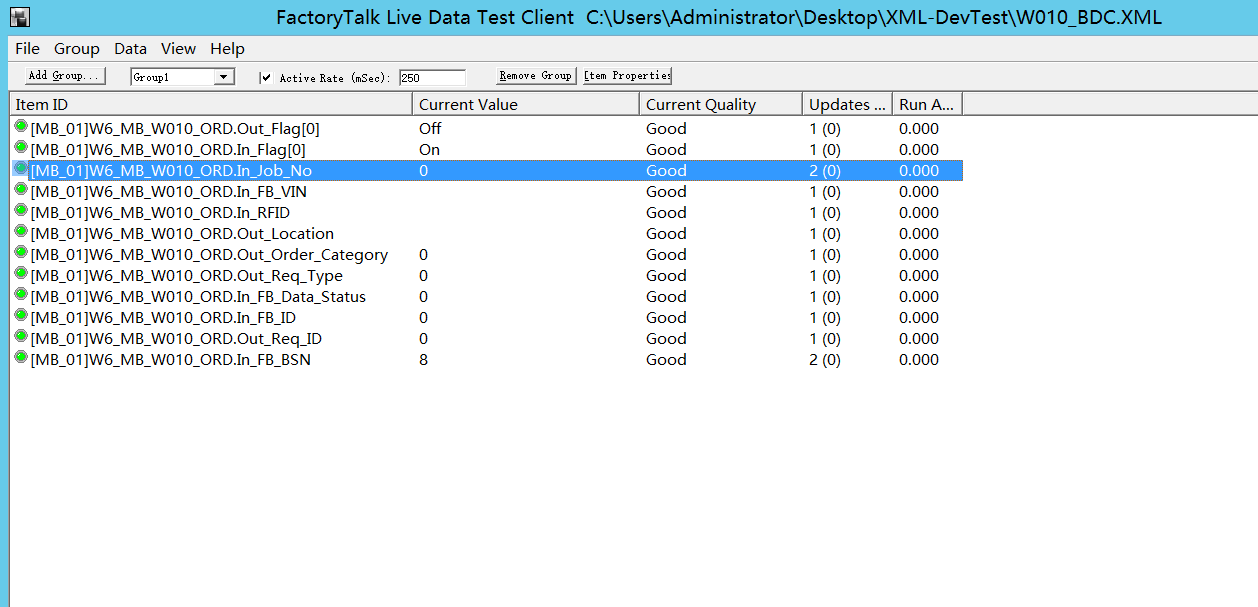
找到刚才PD里配置的站点信息



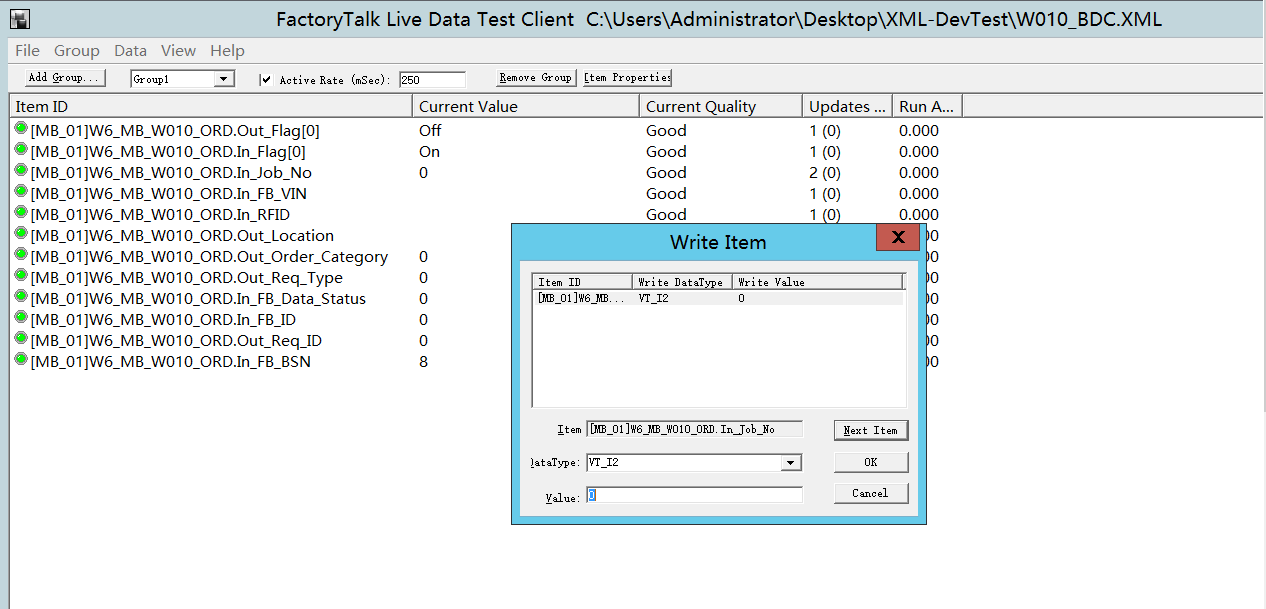
启动PD中的EvenSheet



在Test 软件中选择刚才配置的站点信息

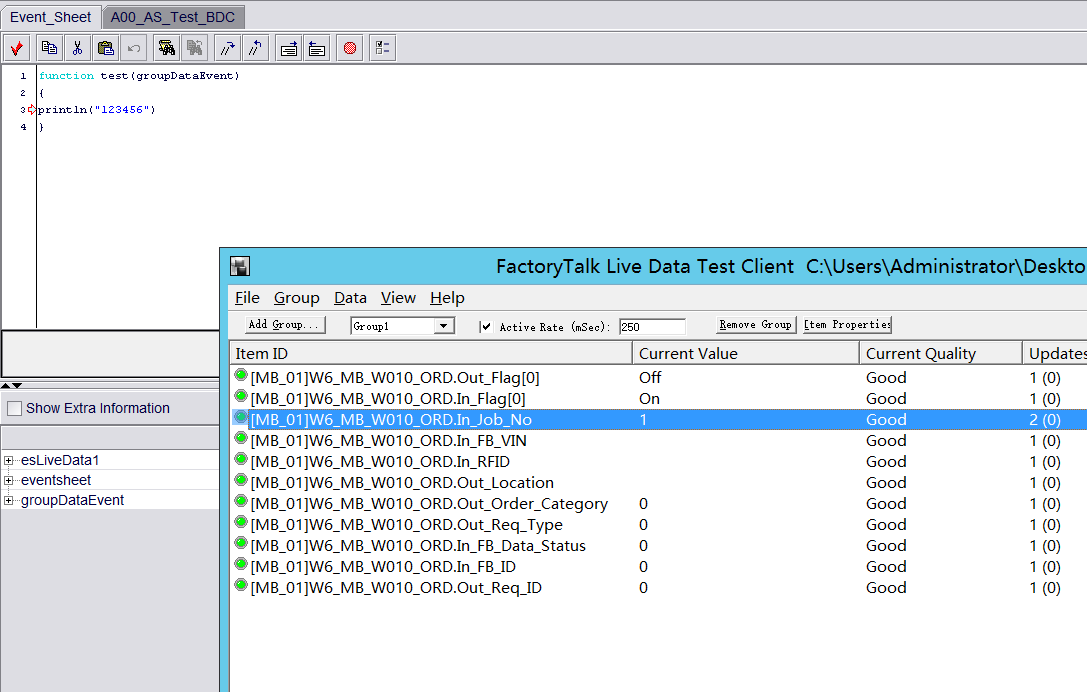


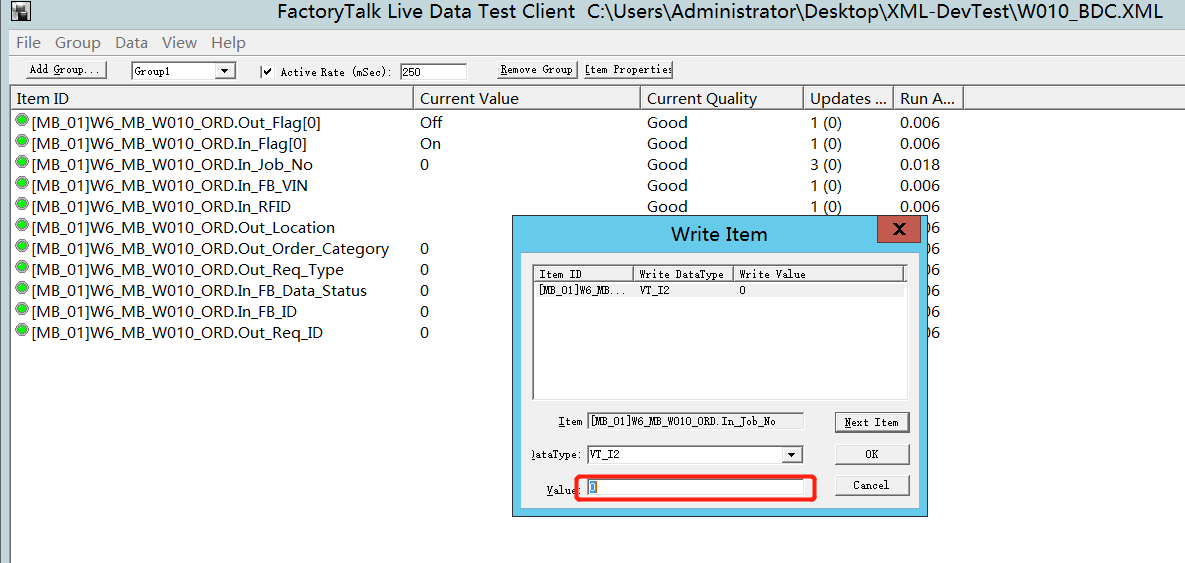
右键选择：writ\_items----Asyn\_Write  点击OK



打开PD 查看是否运行了DateChange方法：

当





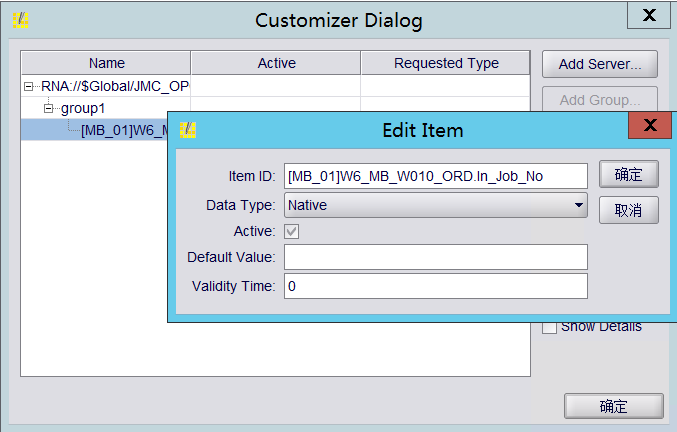
当Value值发生变化的时候  会调用PD里面的test 方法：

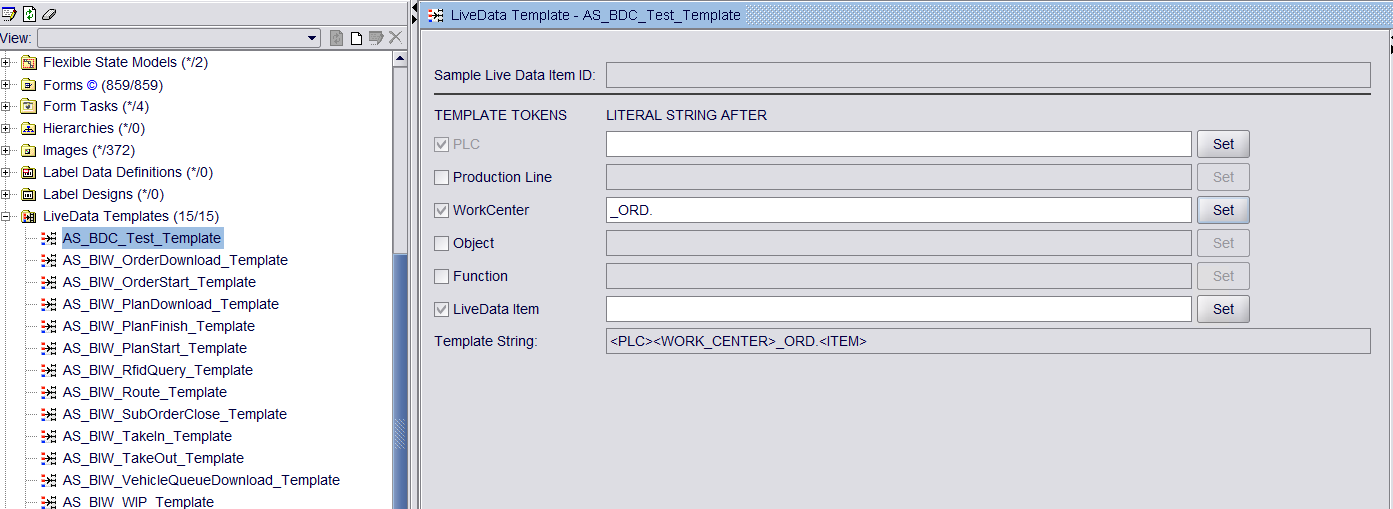
测试成功

**复杂模式**

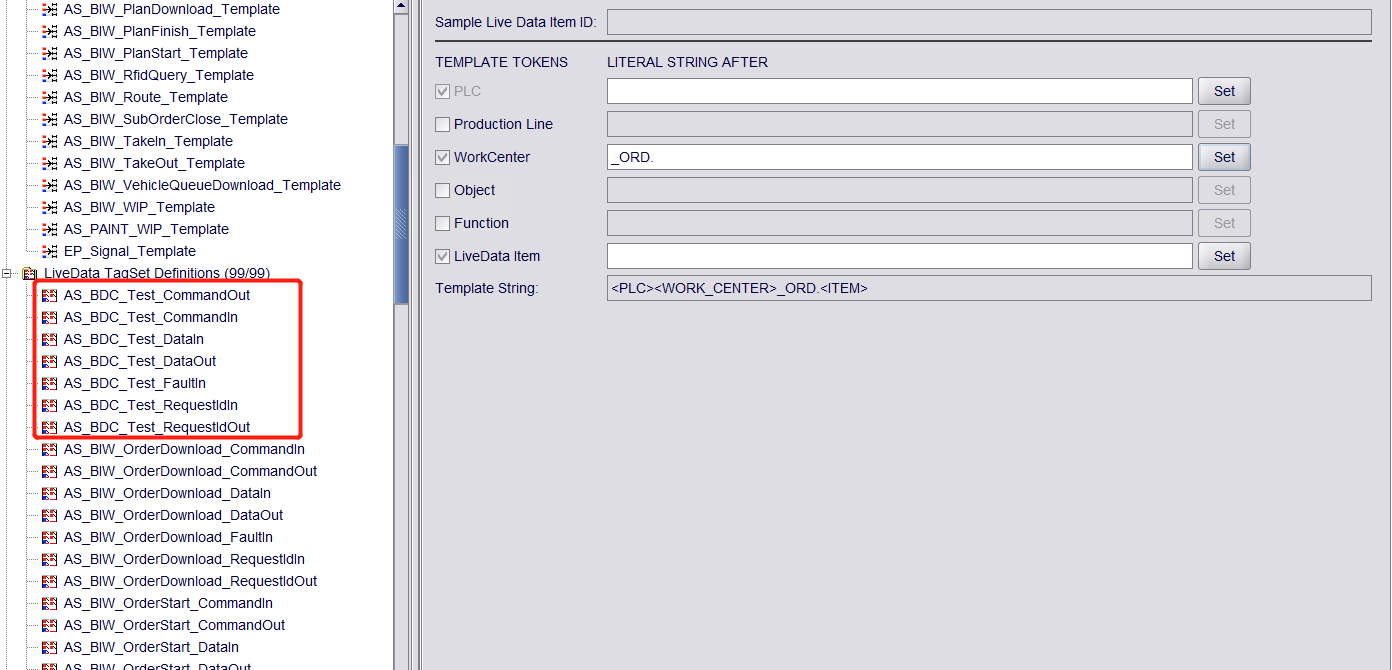
根据站点配置信息配置liveDate模板

建议复制已经调试好的模板 进行修改

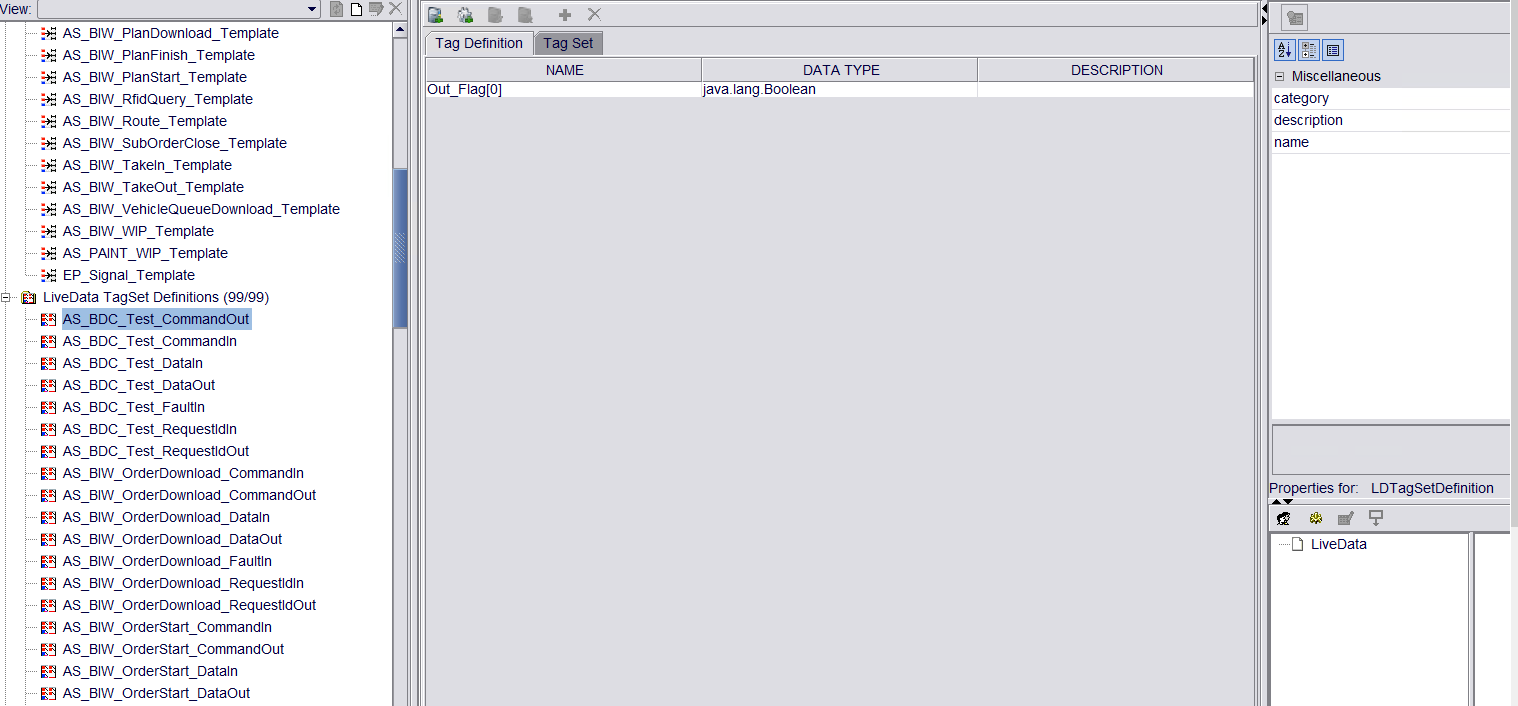




创建Group：



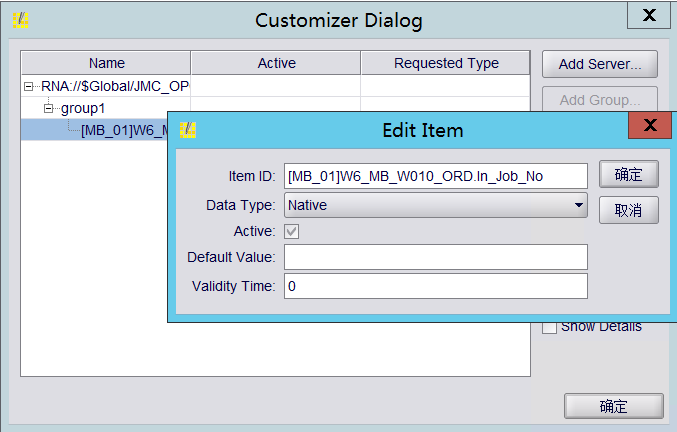
配置Tgae



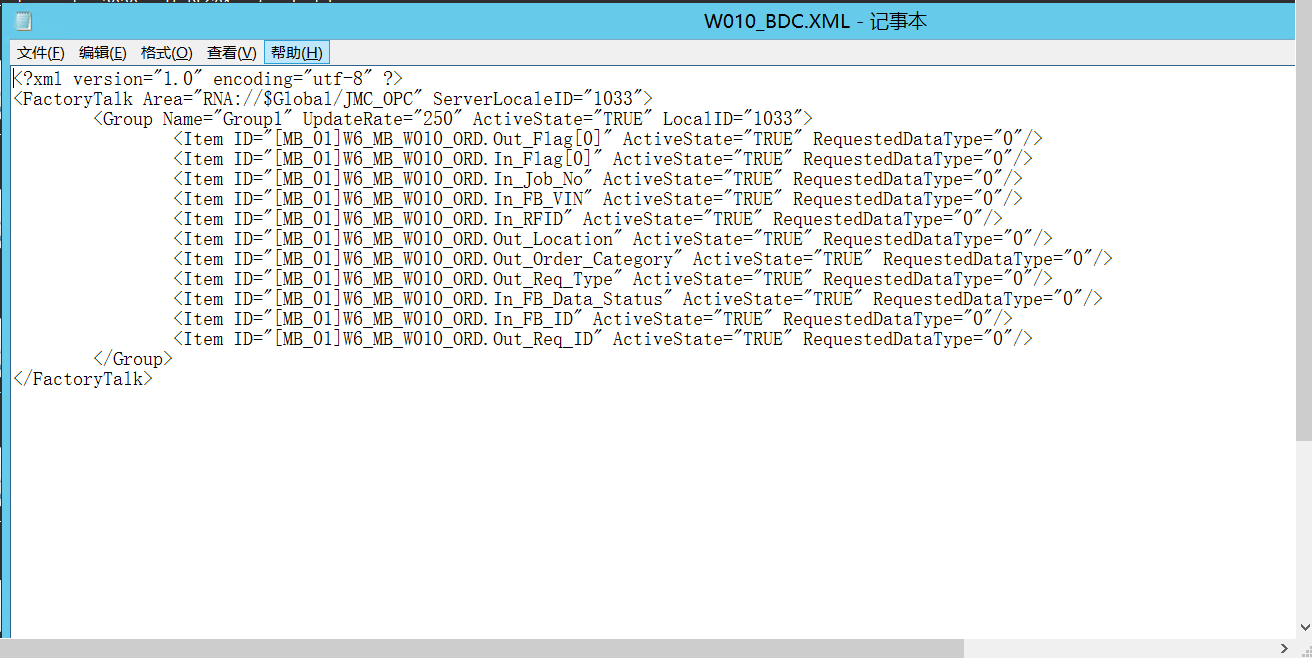
在C盘C:\SOS\TAG\_XML 复制一份原有的xml文件进行修改



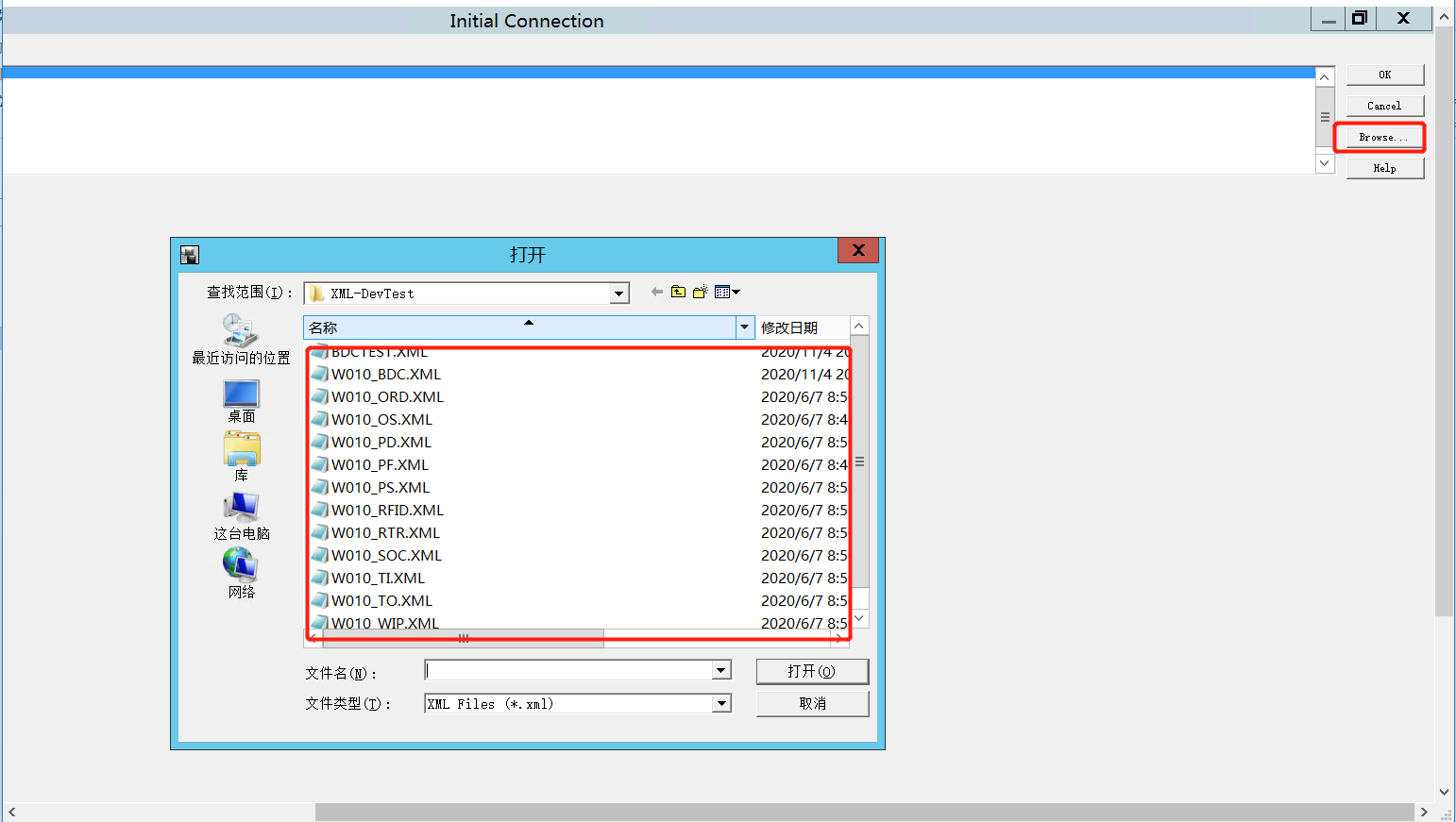
根据站点信息配置 XMl文件



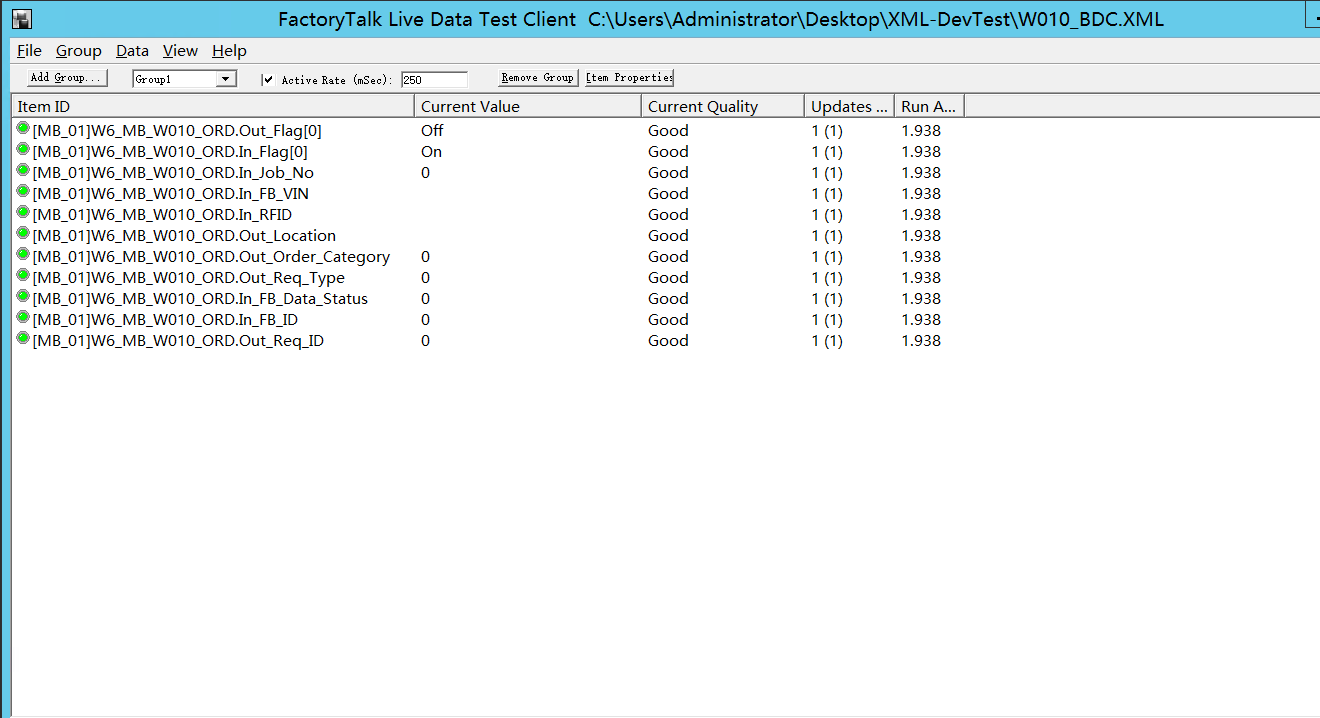
  拷贝C:\Users\Administrator\Desktop\XML-DevTest目录下的XML文件进行修改：



选择刚才配置好的XML文件并打开

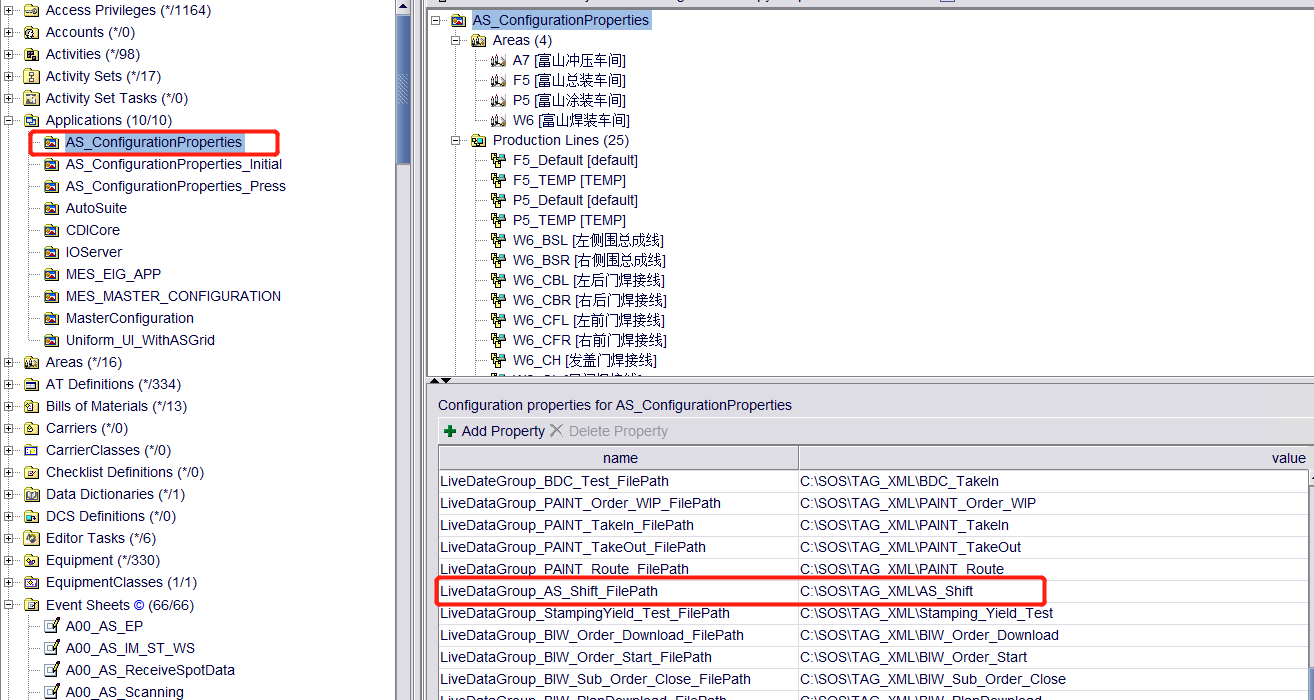


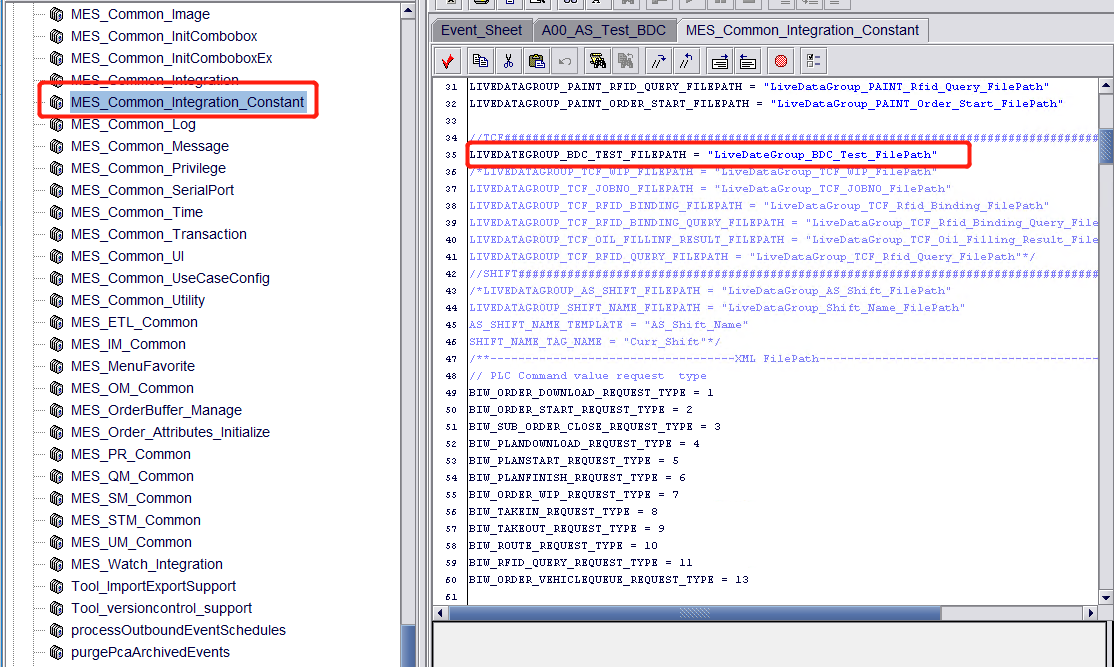
站点测试成功：



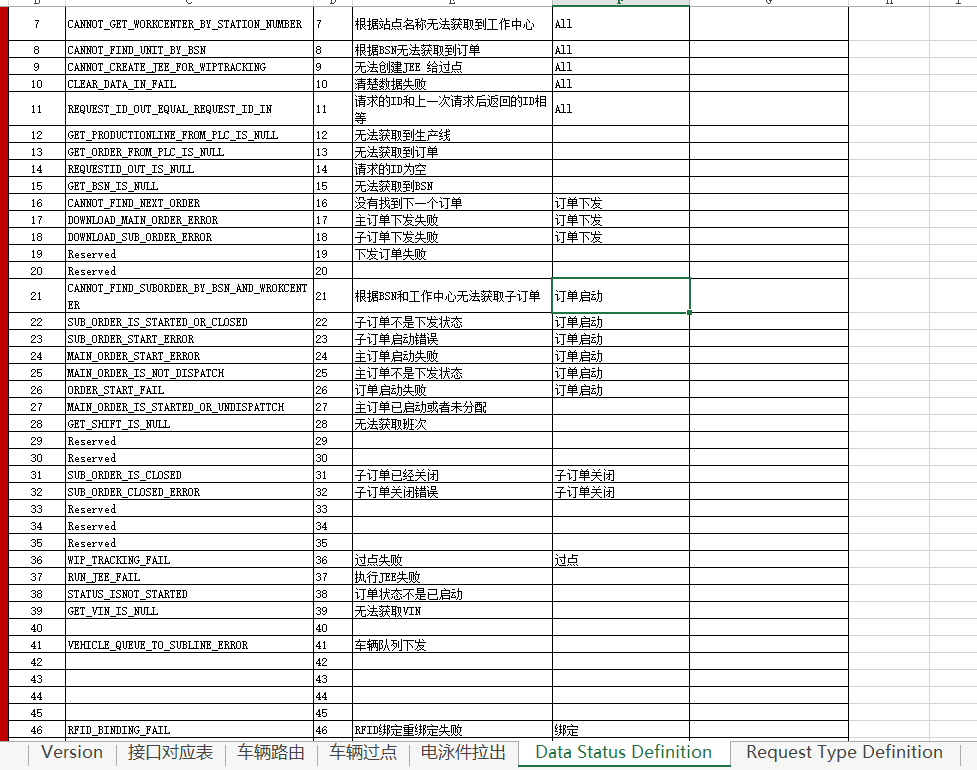
PD LiveDate代码：

配置Application

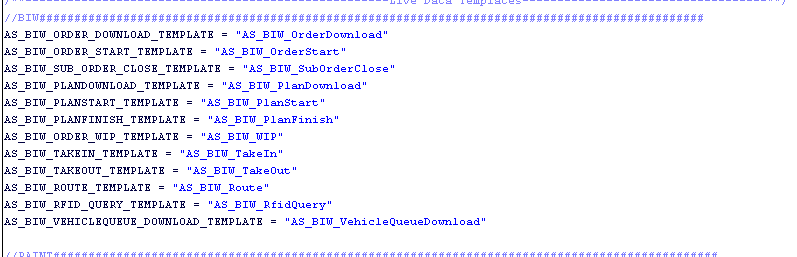


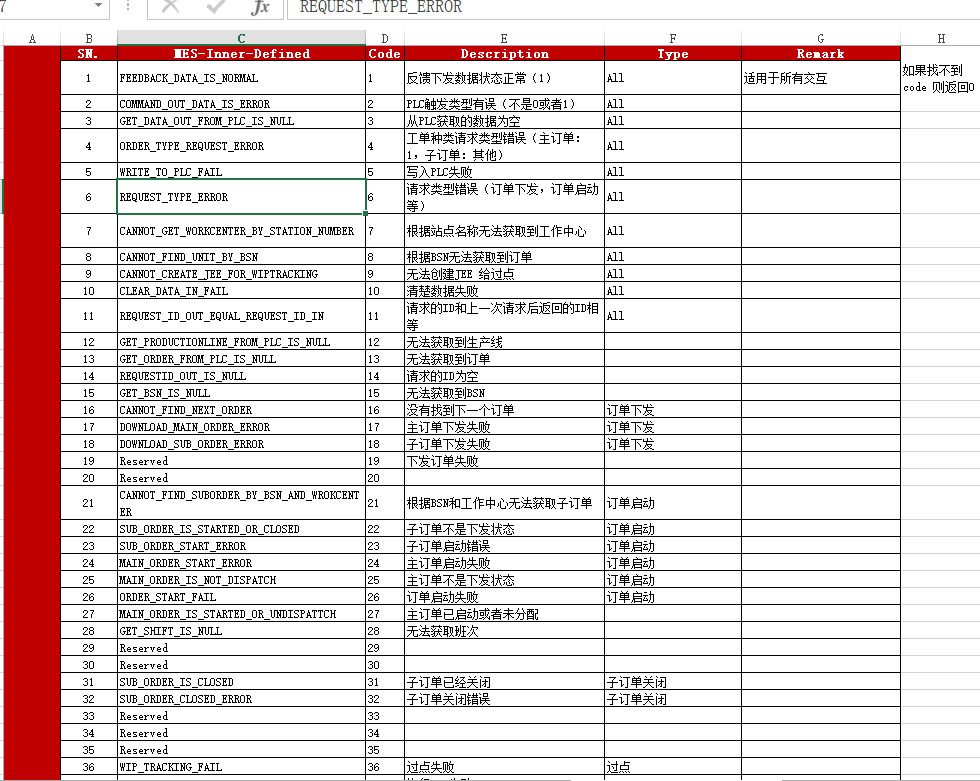


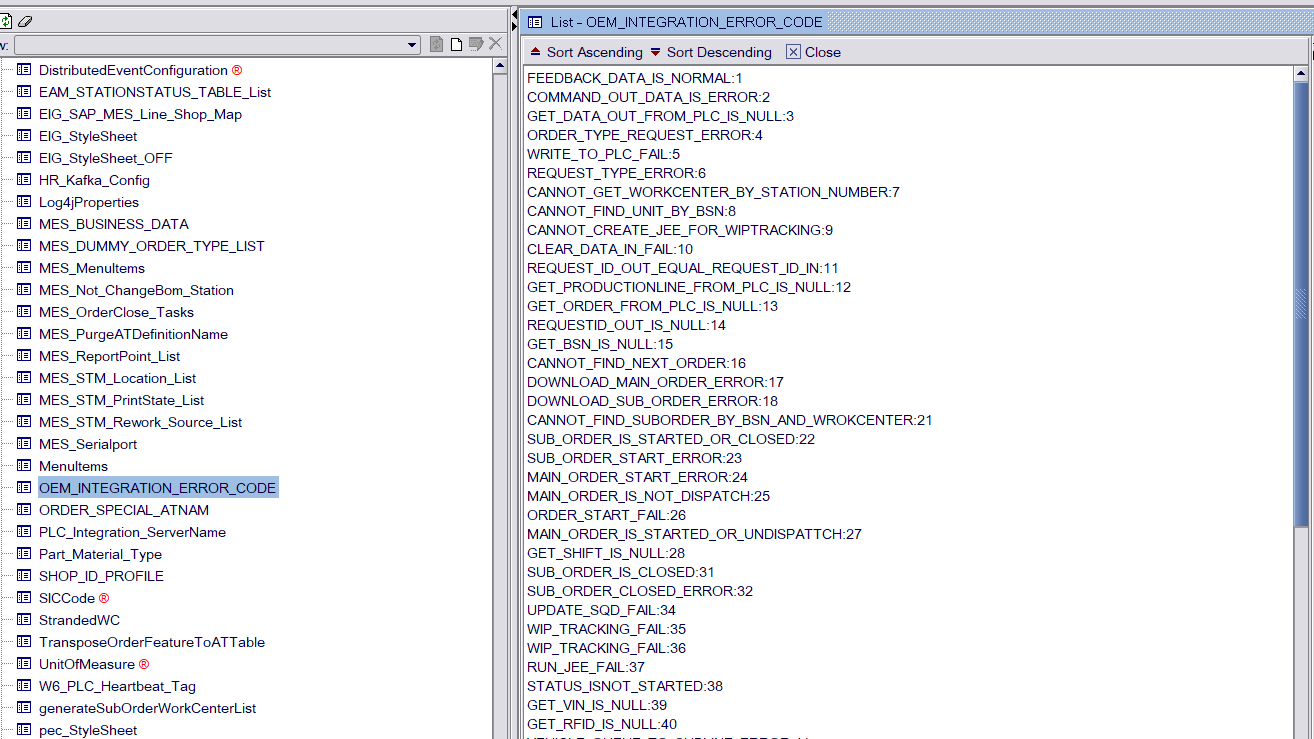
在List中OEM\_INTEGRATION\_ERROR\_CODE根据接口的文档配置字段；中英文：



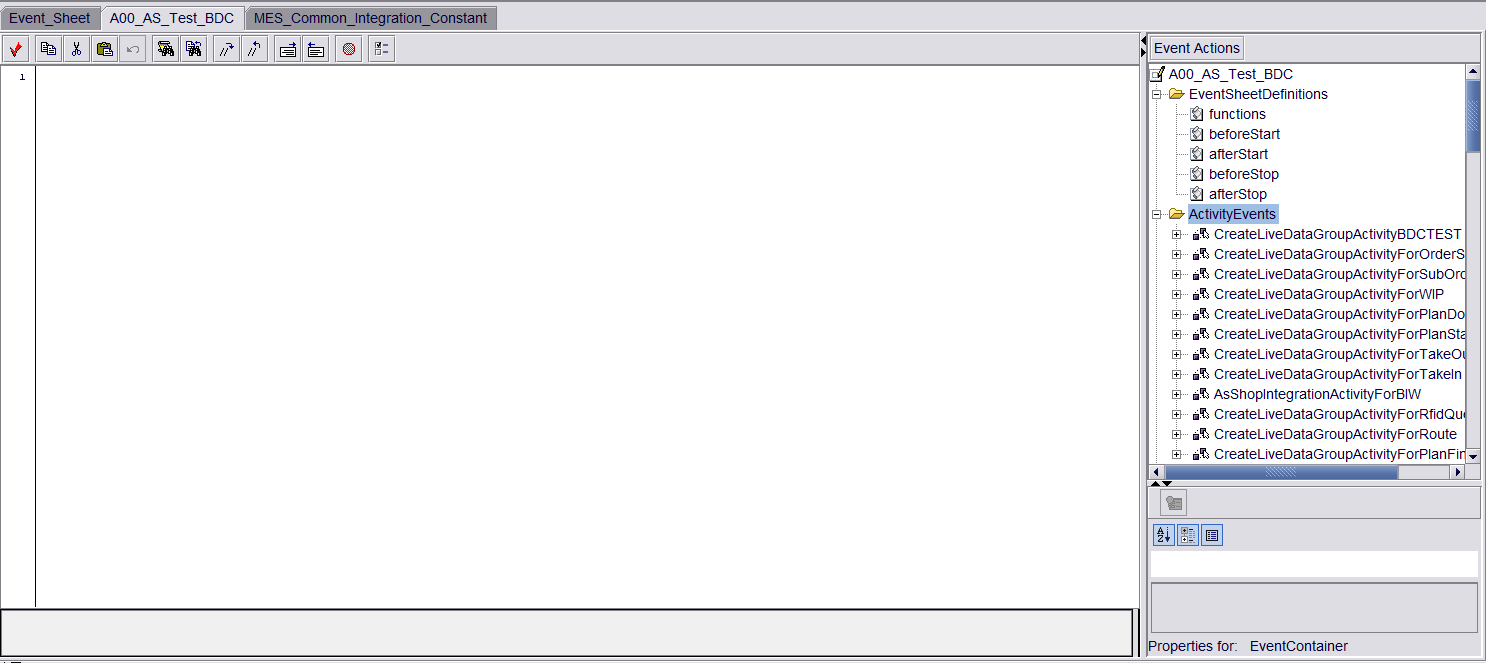
根据模块定义对以应的ID

  
根据文档信息 编写相应代码

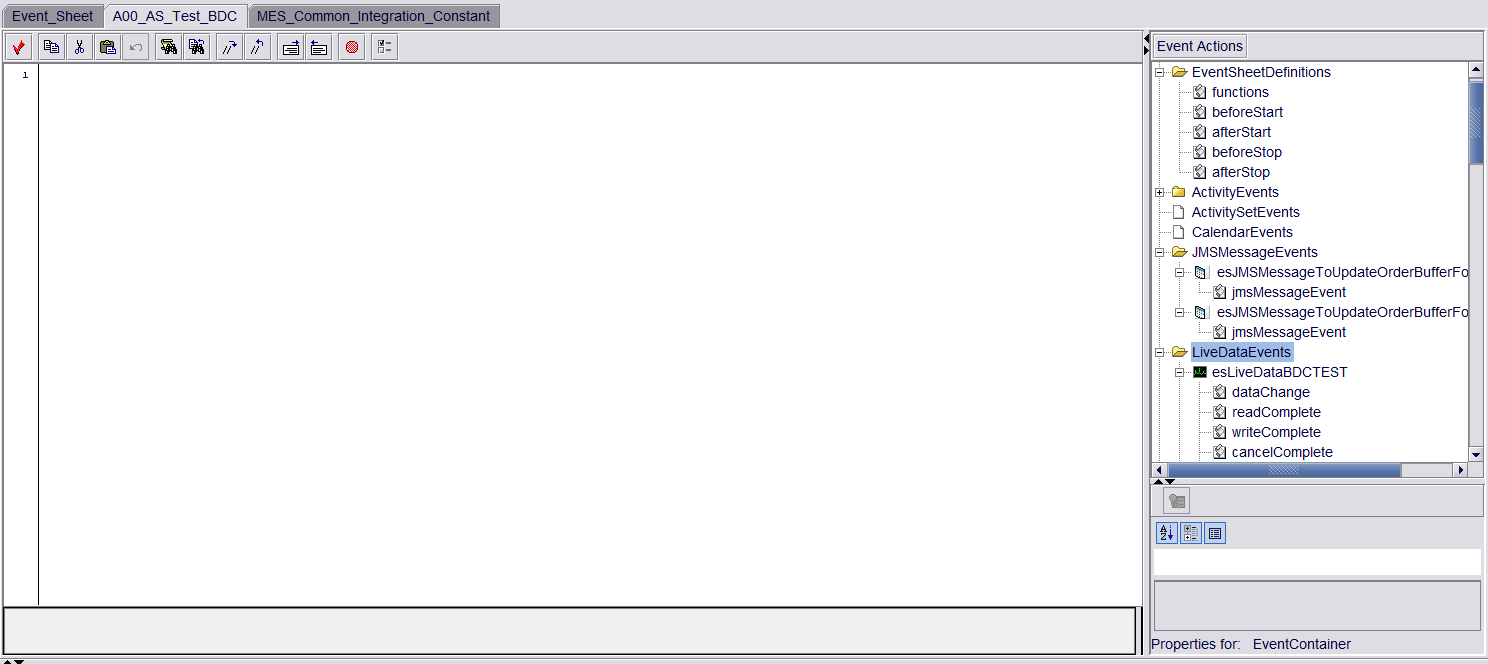
  
在list---OEM\_INTEGRATION\_ERROR\_CODE中编写对于的代码



创建ActivityEvents

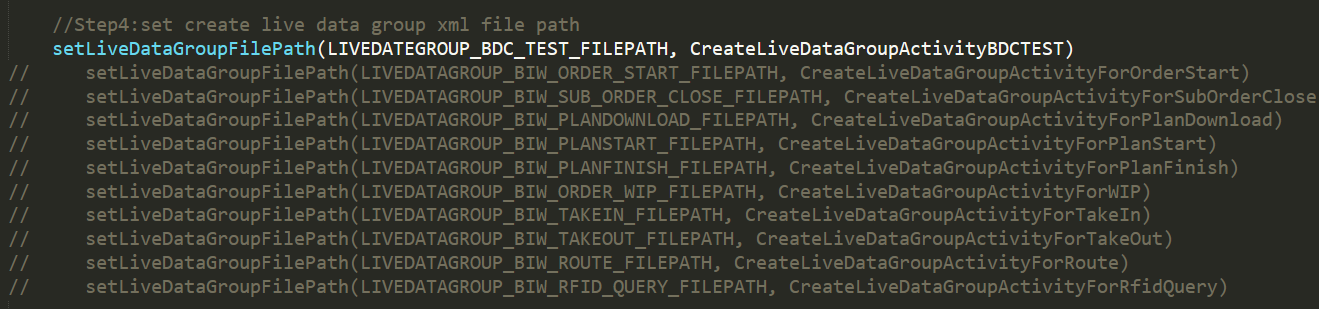


创建LiveDates

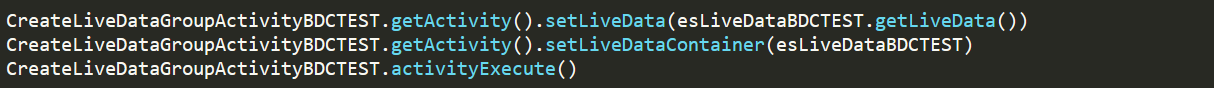


代码部分：

加载模板



运行配置文件赋予esLiveDateBDCTEST中



编写DateChangeBDCTEST方法：

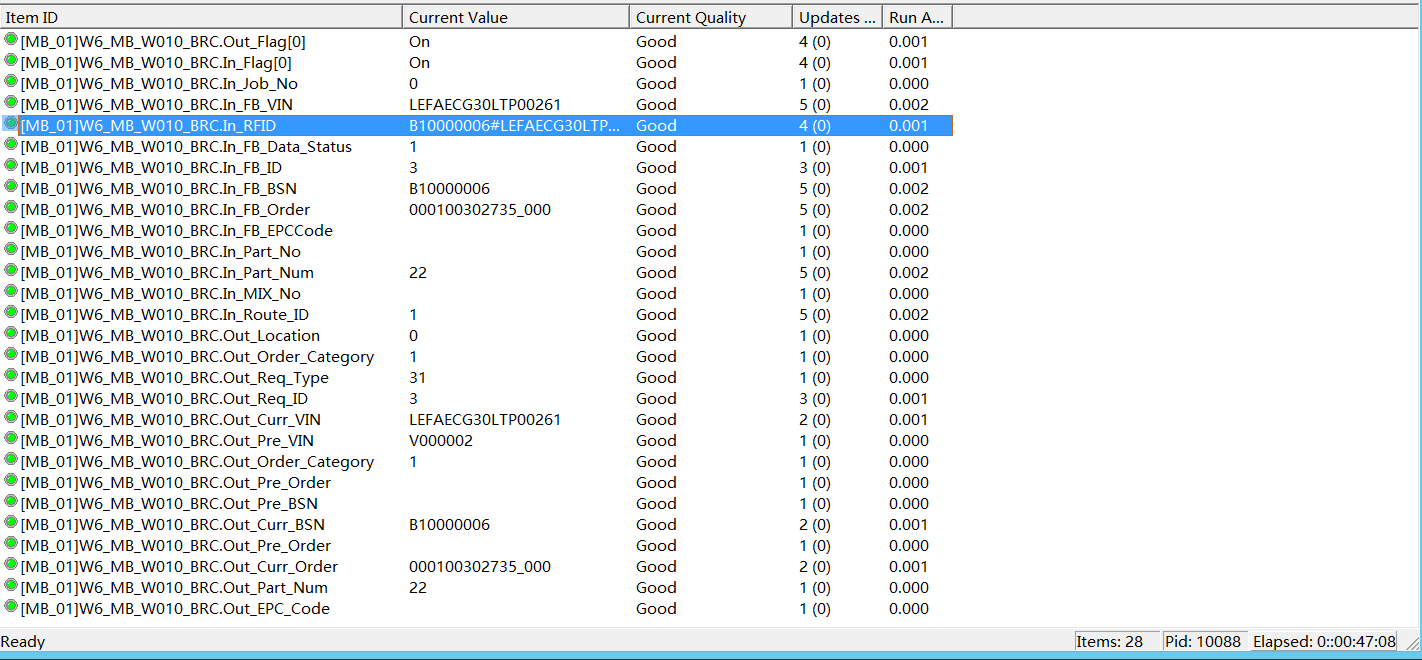


与BDCTEST方法：



**测试：**

通过Test软件更改 该配置项的值 判断是否进入了DateChange方法：



改变Out\_Flag 判断是否进入dateChange方法 是的话则初步调试成功；

在Out里面填写PLC--MES发送的值；参考接口文档的需求字段；



修改完后 修改Out\_Req\_iD  ID不能重复；如果ID重复则根据文档提示 向In\_FB\_Data\_Status 反馈对以应的异常代码

在PTPC中编写后台对应的逻辑代码；

测试对于的逻辑代码功能是否正常

进入的话则判断测试成功;