

调试快捷键

1 说明

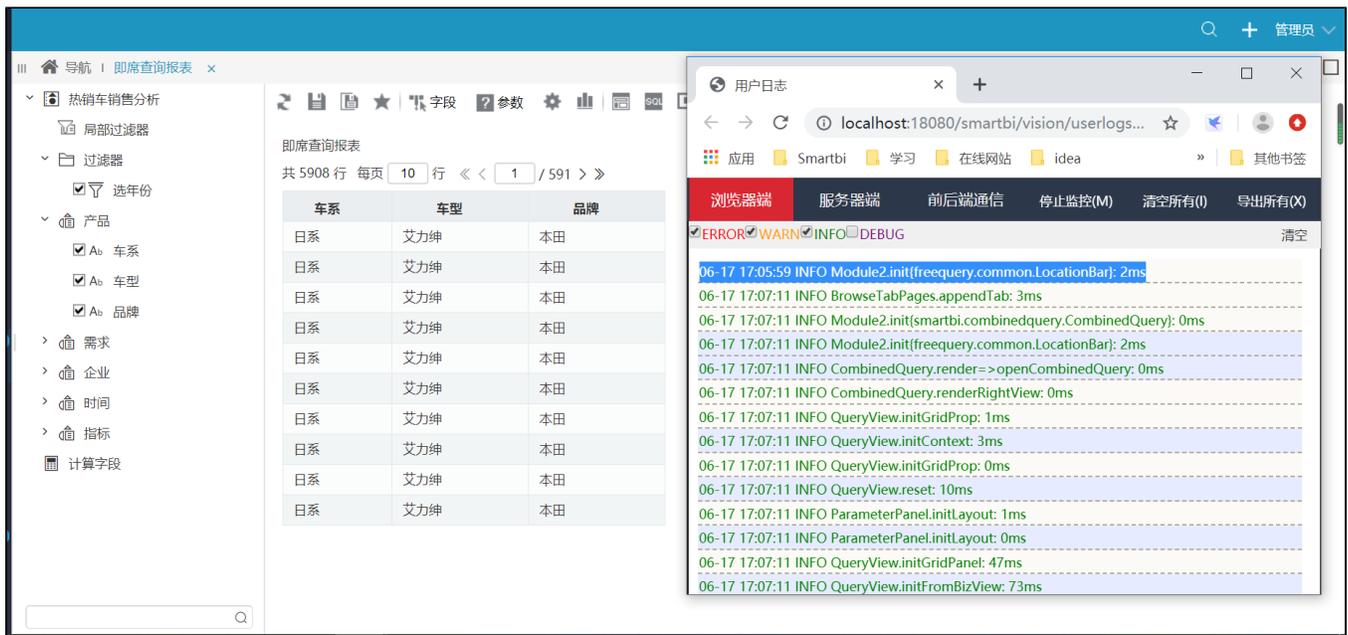
在定制开发过程中，经常需要定位 Smartbi 的代码位置，比如需要查找某个对话框的类名，又或者需要知道资源目录树上某个节点的右键菜单响应事件等。为了方便开发，Smartbi 提供了很多操作快捷键，可以便捷地定位代码位置。具体参考如下。

- 1 说明
- 2 显示和隐藏LOG界面
- 3 定位某功能对应的js类与html模板
- 4 显示当前对话框对应的js类名
- 5 查看资源树对应的类名信息

2 显示和隐藏LOG界面

快捷键：Ctrl + Shift + F2 用于显示/隐藏LOG界面，效果如下。

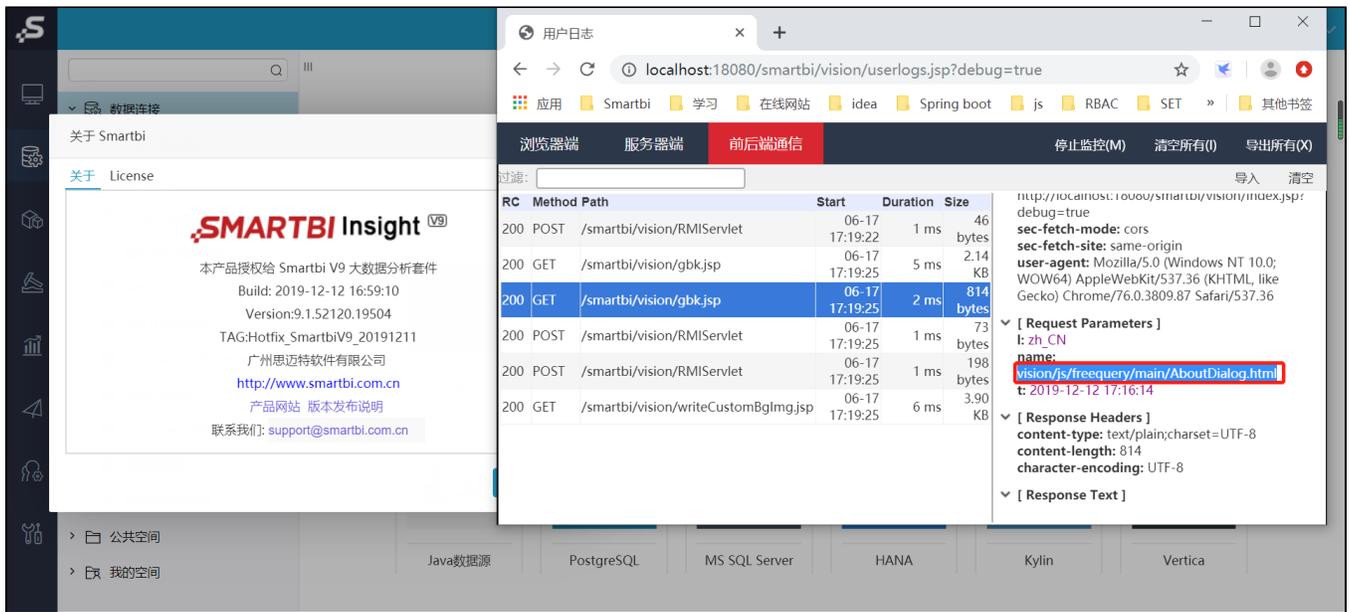
LOG界面显示之后的用户操作如果有发生与服务器的交互，都会显示出来，譬如想知道点击了刷新，执行了什么交互，就可以在点击刷新之前调出这个界面。这个LOG的功能其实与charles工具、chrome 开发者工具的network面板功能类似，只是要用后者监控网络请求的话要加上debug=true（例如http://192.168.1.10:16000/smartbi/vision/index.jsp?debug=true），否则是加密的，下面定位某功能对应的js类与html模板就是应用案例。



3 定位某功能对应的js类与html模板

使用上一小节的显示和隐藏LOG界面快捷键可以快速定位某功能对应的js类和html模板，具体步骤如下：

- 1、先在打开功能界面之前先按“Ctrl + Shift + F2”调出LOG界面，点击开始监控。
- 2、再打开“关于”弹窗，LOG记录效果如下：

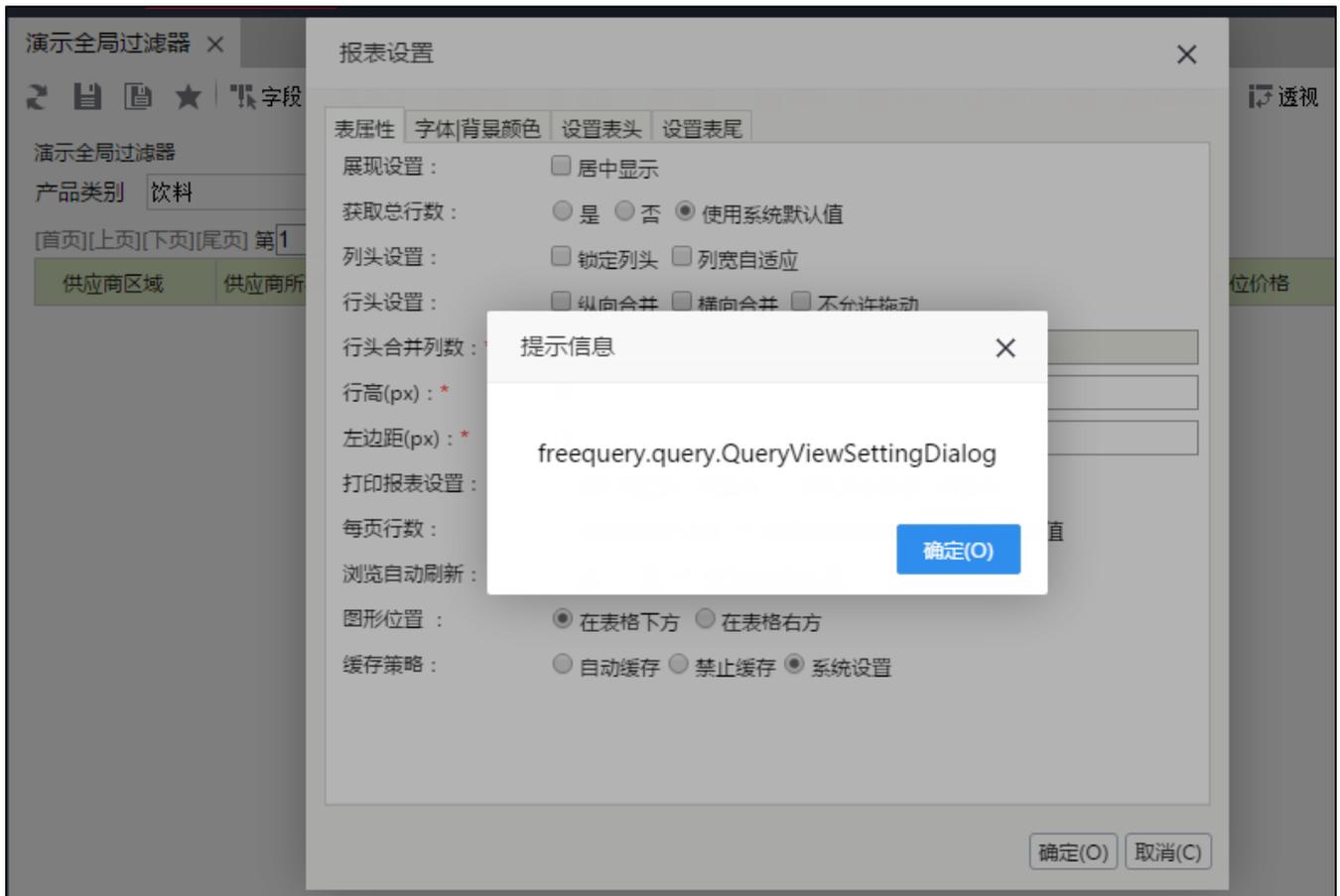


其中红色框选中的内容就是对应的html模板文件名。

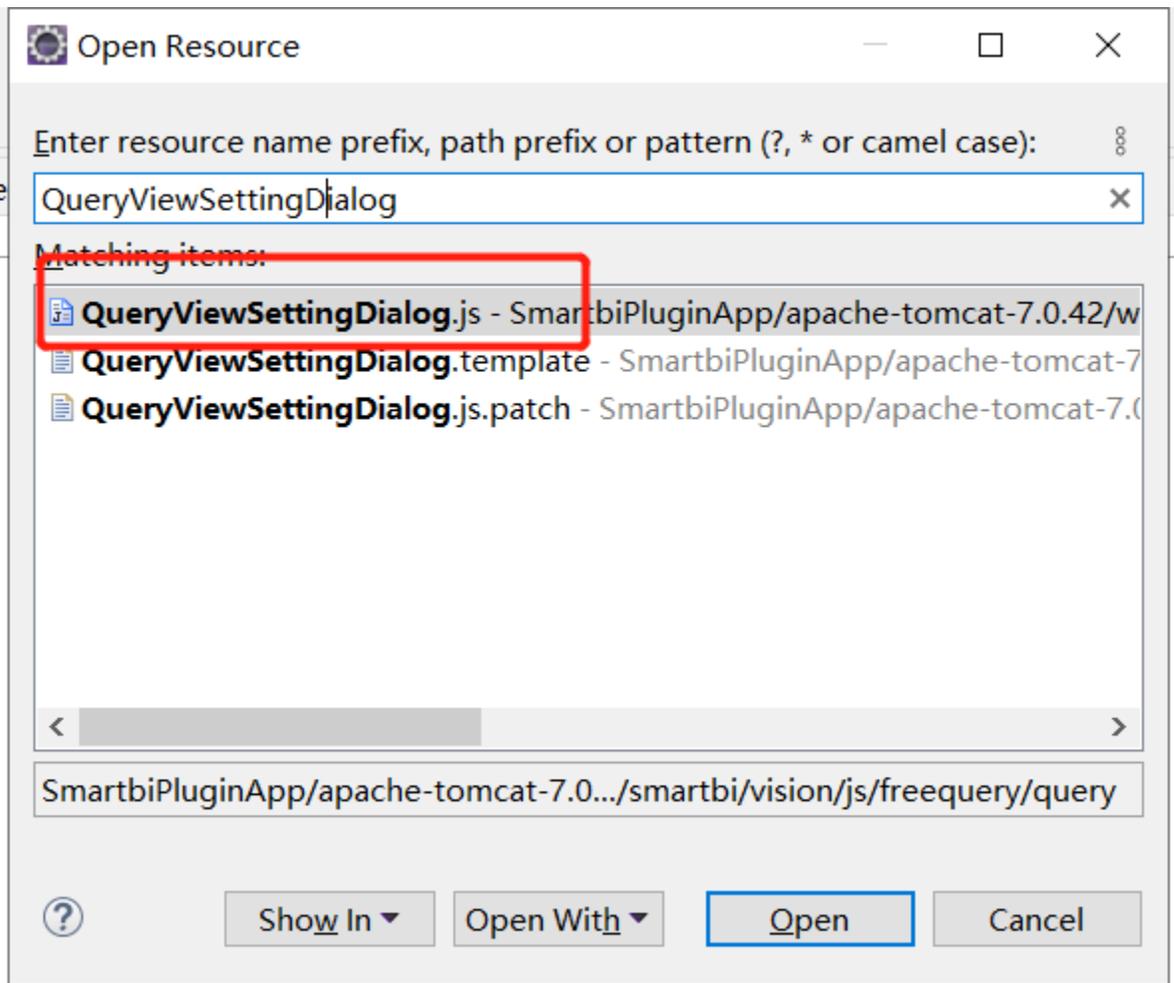
4 显示当前对话框对应的js类名

定位快捷键: `Ctrl + Shift + F4` 用于显示当前对话框对应的js类名。

系统中每个对话框, 都会对应一个js组件, 仅适用于对话框中。效果如下:



取到JS类名之后, 即可定位到JS文件。如需调试, 在文件中找到对应的入口方法, 添加debugger即可进行调试。



```
12 };
13
14 lang.extend(QueryViewSettingDialog, BaseDialogEx);
15
16 QueryViewSettingDialog.prototype.init = function(parent, data, fn, obj) {
17     debugger;
18     QueryViewSettingDialog.superclass.init.call(this, parent, data, fn, obj);
19     this.parent = parent;
20     this.dialogBody.style.border = 'none';
21     this.dialogBody.style.backgroundColor = '#cccccc';
22 }
```

在入口处添加debugger, 即可进行调试

5 查看资源树对应的类名信息

快捷键: 鼠标“中键” —》用于查看资源树对应的类名信息, 以便快速定位资源树对应的代码。

系统中大量使用资源树的地方, 譬如定制管理左侧, 新建报表会有让选择数据源等, 这些树组件因为逻辑不同会对应不同的js组件, 该快捷方式就是快速定位到对应组件。

操作步骤为在“资源树”的“展开/收起”图标位置处点击鼠标“中键”。效果如下:

SMARTBI 导航 | 业务主题 | 区域销售时间维 x

显示名称 显示别名 主题名: 区域销售时间维 主题别名: 区域销售时间维 描述:

northwind (!!! 不可修改) 属性区 表关系视图

DEFAULT

- 过滤器
- 表关系视图
- 计算字段
- 过滤器
- 业务视图
- 参数定义
- 函数列表
- 用户属性

过滤器

t_time2

类型: ID: 名称: * 别名: 描述:

提示信息

Tree: freequery.businesstheme.BusinessThemeViewTree

TreeNode: freequery.businesstheme.BusinessThemeViewTreeNode

确定(O)

视图分析有效) 选择... 清除 (只对透视分析和即席查询有效)

选择... 清除 (只对透视分析有效)

视图分析新建时有效)

表达式: * <表达式编辑器, 拖动对象支持: 表、视图、参数、字段、函数、过滤器、业务属性、用户属性>

是否生成SQL: 是 否

地理维: 是 否

保存(S) 关闭(C)