

第三课：报表宏的调试

客户端模块调试

在Smarthi中进行宏代码开发，或者开发Smarthi扩展包时，都会写大量的JavaScript代码。如何在调试模式下单步执行代码，查看变量值和调用堆栈呢，下面就是操作的过程和步骤。

客户端模块调试本质就是JavaScript的调试，如果您已经是js调试高手可以忽略本节内容，但是[使用debug模式访问smarthi](#)还是需要看看。

- 客户端模块调试
 - 安装调试器
 - 打开IE的调试选项
 - 清除IE缓存
 - 使用debug模式访问smarthi
 - 代码调试
- 服务端模块调试
 - 添加调试代码
 - 查看调试信息
- IE开发人员工具
 - IE Developer Toolbar 安装
 - IE Developer Toolbar调试界面
 - 使用IE Developer Toolbar调试JS代码

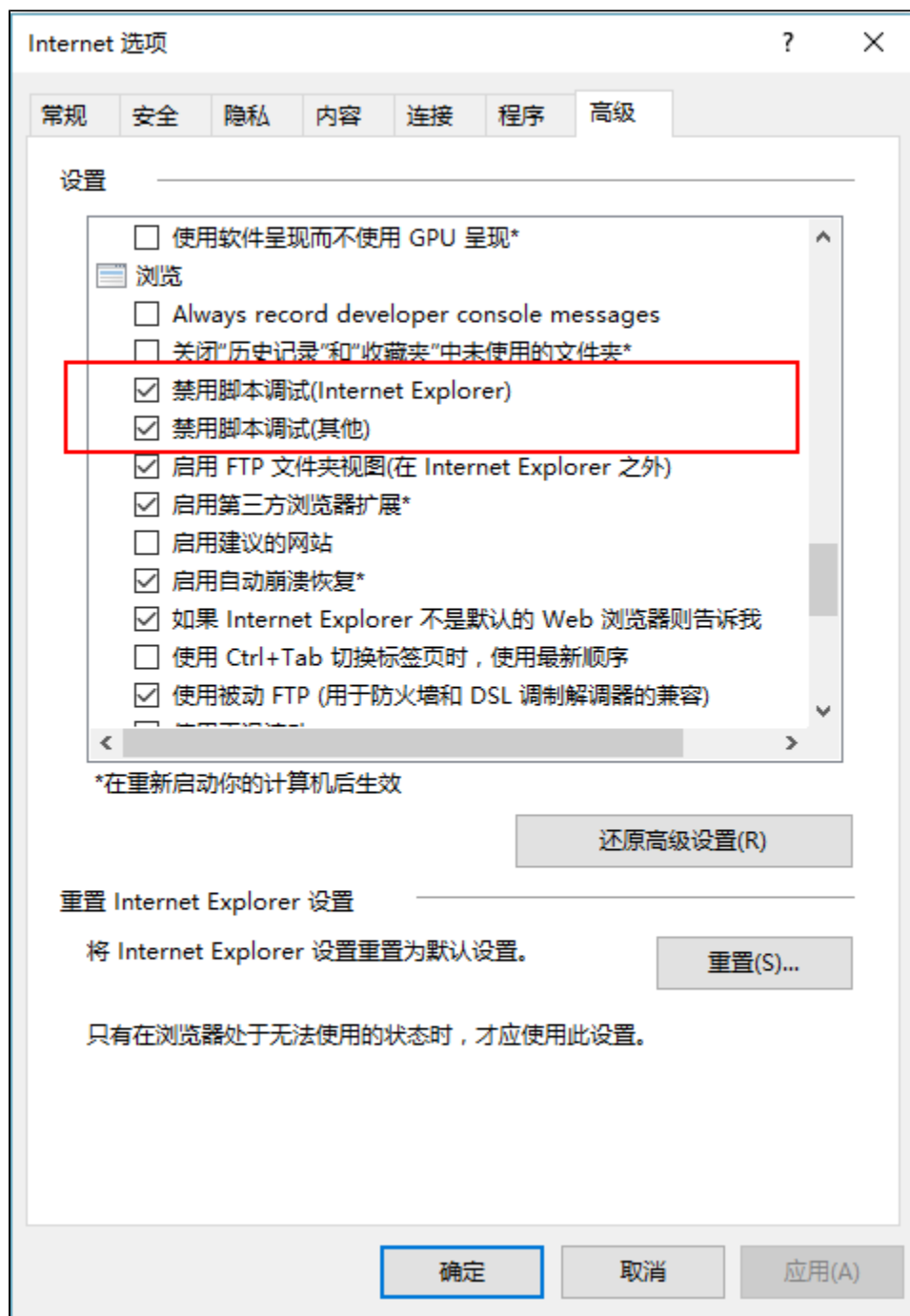
安装调试器

所谓工欲善其事，必先利其器，要调试JavaScript代码，就要先有个好用的调试工具。推荐使用 [Microsoft Visual Studio](#) 开发工具，用它来调试 JS 代码，是非常方便的。

也可以直接使用IE或谷歌浏览器的开发者工具（一般是F12可以调出），谷歌是右上角的 三竖点>更多工具>开发者工具，IE11是右上角的 设置>F12开发人员工具，如果使用浏览器本身调试工具，1.2步骤是非必选项。

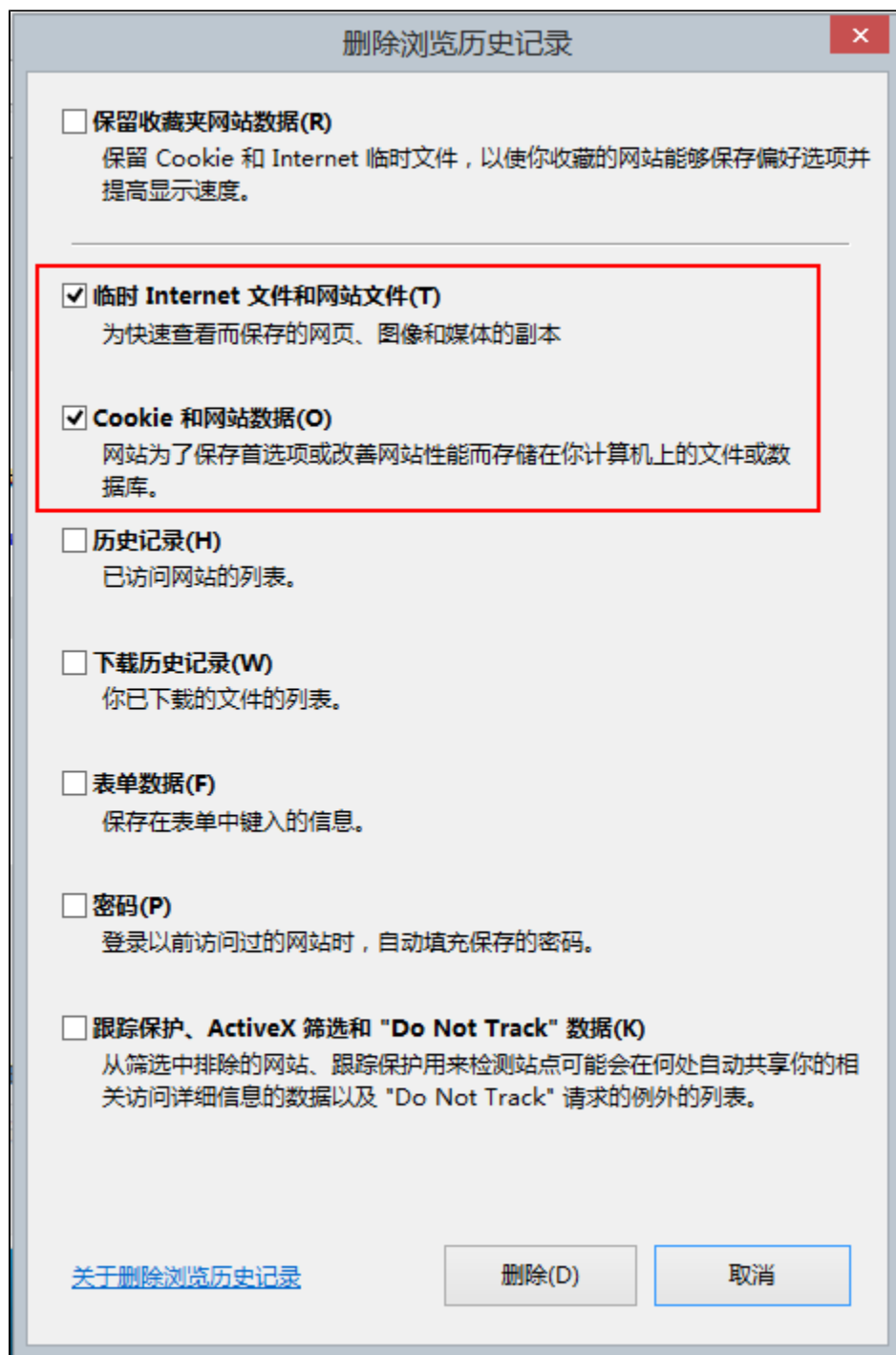
打开IE的调试选项

修改 IE 浏览器设置，打开脚本调试选项。打开Internet Explorer，选择 **工具 -> Internet选项** 命令，在弹出的对话框中选择 **高级** 选项卡，在 **浏览** 项下，取消选中 **禁用脚本调试**（Internet Explorer）和 **禁用脚本调试**（其他）复选框，单击 **确定** 按钮。



清除IE缓存

清除 IE 浏览器缓存，点击 **工具 > 删除浏览的历史记录** 菜单项，在弹出的对话框上勾选如下图所示删除项目，点击 **删除**。然后关闭IE浏览器。



在IE删除浏览历史记录对话框上，第一项“保留收藏夹网站数据”不要勾选；下面的“Internet 临时文件”、“Cookie”两项，最好全部勾选。然后点击“删除”按钮。

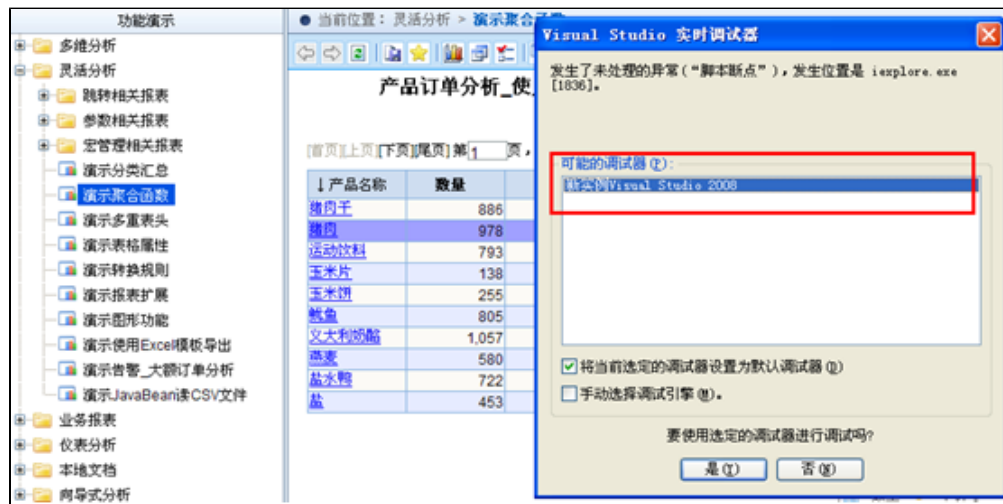
使用debug模式访问smartbi

重新打开 IE 浏览器，在浏览器地址栏输入 `http://localhost:8080/smartbi/vision/index.jsp?debug=true` 地址访问Smartbi。Smartbi服务器默认会对返回到客户端的JavaScript代码进行压缩，同时对HTTP Request、Response信息进行加密，为了便于调试，需要在debug模式下访问Smartbi，即在URL中添加debug=true参数。

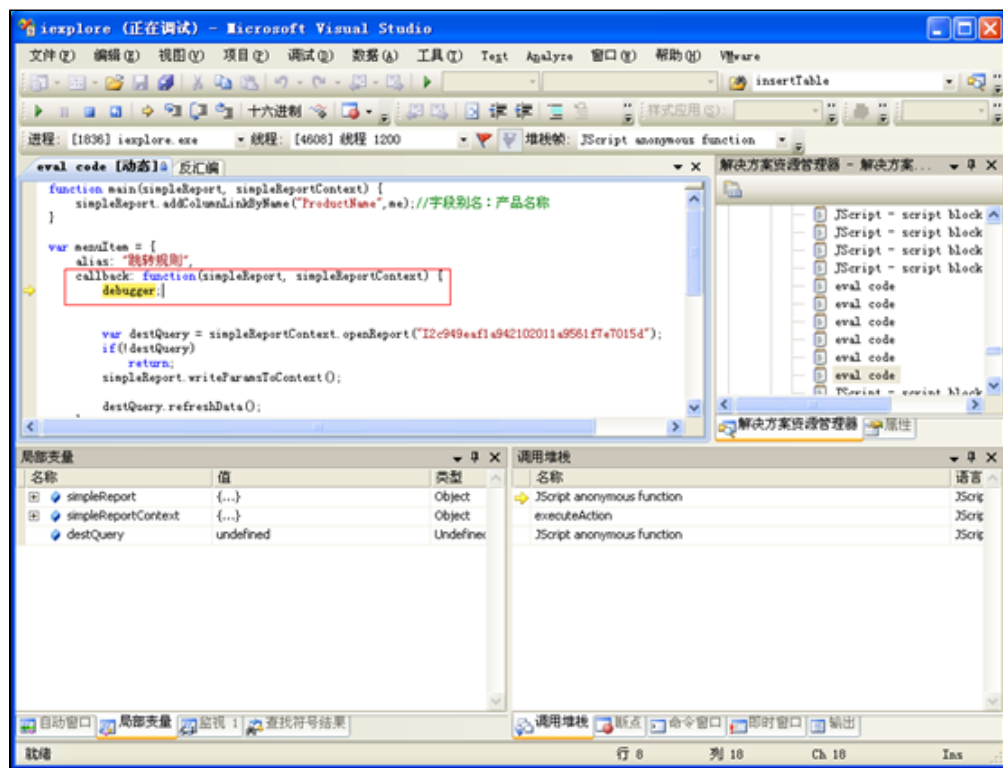
```
http://localhost:8080/smartbi/vision/index.jsp?debug=true
```

代码调试

接下来，在您的宏代码或者插件代码中添加 JS 断点 `debugger`，然后访问Smartbi，当程序执行到断点位置时，浏览器会弹出“实时调试器”选择对话框，在“可能的调试器”列表框中选择 **新示例Visual Studio 2008**，点击 **是** 按钮。



这时会打开“Microsoft Visual Studio”，并定位到代码中 `debugger` 断点的位置，此时您就可以单步执行代码，并随时查看各种变量值，进行开发调试了。



服务端模块调试

添加调试代码

宏控制台仅用于对“服务端模块”的报表宏进行调试。进行调试时需要在宏脚本中通过“logger”对象打印出服务端的日志信息。

```

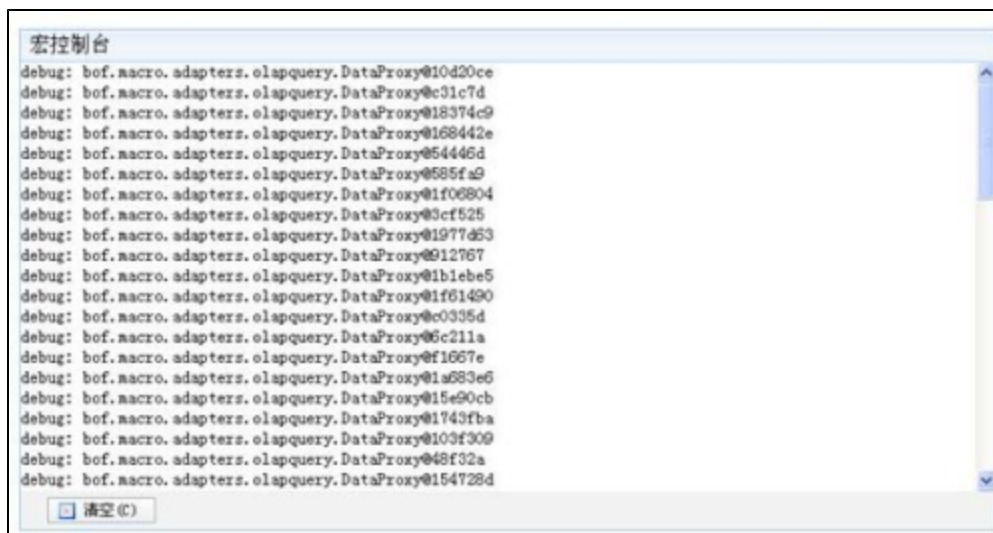
1 var jmUtils = use("system.utils.JMServerUtils");
2 function main(olapTable){
3     if(!olapTable)
4         return;
5
6     setDataCellsClickable(olapTable.data);
7 }
8
9 function setDataCellsClickable(datas){
10     if(!datas)
11         return;
12     for(var i=0; i<datas.length; i++){
13         var data = datas[i];
14         var cell = {data: data, cellType: "Data"};
15         setClickable(cell);
16     }
17 }
18 function setClickable(cell){
19     if(!cell)
20         return;
21     logger.debug(cell.data);
22
23     if(jmUtils.dataCellMemberAliasEquals(cell, "[Measures]", "销售量")){
24         if(cell.data && cell.data.text != ""){
25             cell.data.clickable = true;
26             cell.data.groupName = cell.data.groupName + ", [销售明细表]";
27             return;
28         }
29     }
30 }
31
32 }
33
34 }
35
36
37
38
39 }

```

查看调试信息

打开该宏所在的报表，刷新后，点开 **定制管理 > 系统运维 > 开发工具集**，选择点击 **宏控制台** 即可看到相关的输出信息。

运行监控	用于监控服务器目前运行情况
GC监控	用于监控服务器GC运行情况
Log4j配置	用于获取、修改Log4j的级别配置
页面离线	进入页面离线界面
IP & Mac	IP & Mac
连接池信息	连接池信息
对象池信息	对象池信息
Mondrian 管理页面	Mondrian 管理页面
Mondrian SQL日志	Mondrian SQL日志
宏控制台	查看宏代码的后台日志输出。
宏向导重生	宏向导的一键重生成



IE开发人员工具

IE Developer Toolbar是微软专门为 Web 开发人员设计的一款免费产品，该产品让开发人员能够深入探索和理解Web页面，帮助开发者更好地创建Web应用。安装后可以在 IE 中快速分析网页的软件。

IE Developer Toolbar 安装

在早期的 IE 浏览器中，需要自己手动安装该工具，但是 IE 8 以后的版本中，已经自带该功能。在 IE 浏览器的“工具”菜单中有一个“F12 开发人员工具”菜单项，即是该工具。



IE Developer Toolbar调试界面

在IE Developer Toolbar调试界面中，通过“DOM资源管理器”功能，我们可以点击选择界面元素，查看或调试该元素的页面信息。在很多项目上我们经常要写一些报表宏去改造Smartbi里面的报表界面，比如：调整报表的结构或者屏蔽一些功能按钮之类的操作。而报表宏的帮助文档不可能列出所有对象及方法，只列出一些常用的重要的对象及方法供报表宏的开发人员使用，这远远无法满足各式各样的定制要求。这时可以使用IE Developer Toolbar工具弥补这些不足，可以通过该工具找到报表DOM对象，并对报表界面进行定制。

为了方便用户去操作DOM对象，Smartbi产品做了一些封装。在DOM结点里面如果看到bofid这样的定义，那么就可以很方便的获取该结点的对象。一个简单报表为例，如图：



在调试窗口中找到报表标题DOM结点对象信息，比如“<div class='_reportHeader no_margin_p' bofid='_reportHeader'>”，这时候可以拿到它的bofid和class信息，在报表宏中要想获取该结点对象有二种办法：

- simpleReport.elem_reportHeader;
- domutils.findElementByClassName(simpleReport.panel, "_reportHeader");

使用IE Developer Toolbar调试JS代码

请参照以下步骤对报表的宏代码进行调试。

- 1、首先在报表的宏代码中添加 debugger 断点。
- 2、打开“F12 开发人员工具”，切换到“调试程序”TAB页，如果在其下的工具栏上有“启动调试”按钮，就点击一下该按钮启动调试，IE11 之后的浏览器默认是启动的，没有该按钮，不需要点击。
- 3、然后打开对应的报表，刷新。

当程序代码执行到所设置的断点位置时，程序自动停止了，光标定位到 debugger 位置。从这里，您就可以通过单步执行的方式，调试您的代码了，随时查看所有变量、对象的值。

