宏代码库

主要用来定义一些公共的宏方法。将一些使用频率较高的方法定义为公共的方法，便于代码重用，消除重复。

- 宏代码库中声明的公共方法不通过事件触发，因此需要通过报表宏或跳转规则引用宏代码库来调用这些公共方法。
- 系统默认内置的工具宏为：system > utils，分别针对服务端模块和客户端模块定义了一些常用的公共方法。
- 使用时用use(完整的路径)，譬如var jmUtils = use("system.utils.JMClientUtils")，这样JMClientUtils中的所有方法都可以jmUtils.方法名引用。

- 操作入口
- 操作步骤
  - 创建宏代码库
  - 给宏代码库定义公共方法
  - 调用工具宏定义的公共方法

系统中的宏代码库主要包括两项内容：

- **目录**：用来存储不同的宏代码文件。如内置的system就是一个目录。
- **宏代码库**：等同于一个代码文件，文件中定义了许多公共的方法。如内置的utils就是一个宏代码库。

操作入口

入口1：在“系统导航栏”选择 **公共设置**，展开资源目录区，在“宏代码库”或其节点下文件夹的更多操作选择 **新建 > 宏代码库**，进入“新建宏代码库”界面。

入口2：在“公共设置”界面主菜单选择 **宏代码库。**

操作步骤

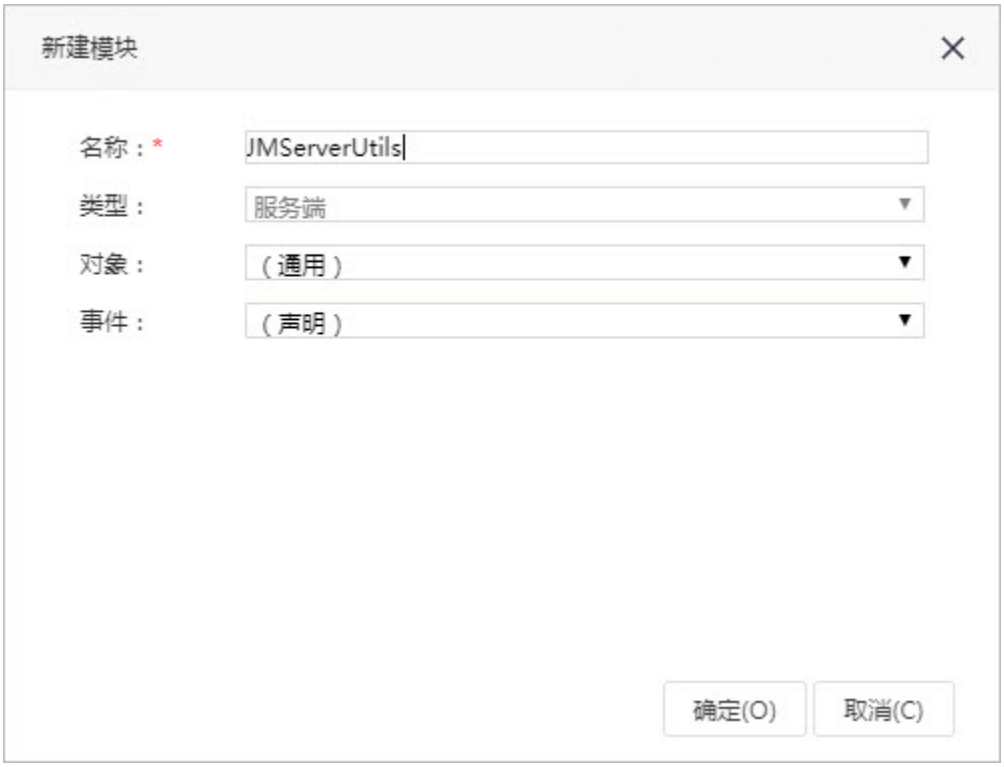
创建宏代码库

在"公共设置"界面的主菜单，直接点击 **宏代码库** ，进入“新建宏代码库”设置界面，输入"名称"为"utils"。

给宏代码库定义公共方法

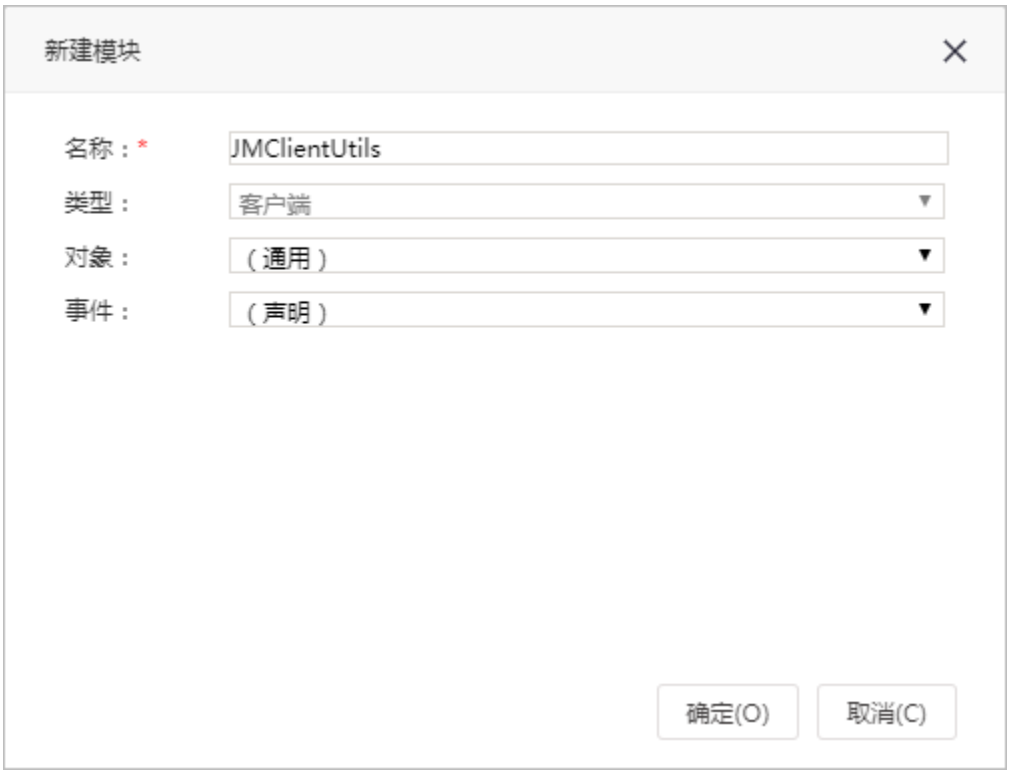
1、在“资源目录区”展开“宏代码库”节点，找到已有的宏代码库，在其更多操作中选择　打开，或双击现有宏代码库，打开宏编辑界面。

2、对模块区的“服务端模块”节点，在其更多操作中选择　**新建模块**，弹出“新建模块”窗口，输入名称，并在“对象”中选择“（通用）”，“事件”中选择“（声明）”。



3. 在编辑区中，编写对应的宏代码。

4. 对模块区的“客户端模块”节点，在其更多操作 选择 **新建模块** ，弹出“新建模块”窗口，输入名称，并在"对象"中选择"（通用）"，"事件"中选择"（声明）"。



5. 在编辑区中，编写对应的宏代码。

调用工具宏定义的公共方法

工具宏定义的公共方法按照如下方式被调用：
例如：系统中有utils工具宏，该路径为"工具宏>system>utils"。在该工具宏中定义了JMClientUtils宏模块，模块中实现了headerMemberAliasEquals(cell, memberName, value)方法。

- 在宏代码中通过"var jmUtils = use("system.utils.JMClientUtils")"声明调用文件的路径。
- 在具体的宏代码中通过"变量名.方法"的方式，如"jmUtils.dataCellMemberAliasEquals(cell,"[Measures]","Profit")"，来调用相应的方法。

应用方法如下所示：

```
var jmUtils = use("system.utils.JMServerUtils");

function main(olapTable){
    if(!olapTable)
        return;

    //遍历数据区,满足条件的单元格添加链接
    setDataCellsClickable(olapTable.data);
}

function setDataCellsClickable(datas){
    if(!datas)
        return;

    for(var i=0; i<datas.length; i++){
        var data = datas[i];
        var cell = {data: data, cellType: "Data"};
        setClickable(cell);
    }
}

function setClickable(cell){
    if(!cell)
        return;

    //数据区链接显示处理
    if (jmUtils.dataCellMemberAliasEquals(cell,"[Measures]","销售量")){
        if(cell.data && cell.data.text != ""){
            cell.data.clickable = true;
            cell.data.groupName = cell.data.groupName + ",[销售明细表]";
            return;
        }
    }
}
```