

计划任务中刷新报表

- 使用场景
- 实现思路
- 操作步骤

 示例代码只针对即席查询和透视分析有效。

使用场景

项目中具有较多的大数据量报表，初次打开这些大数据量报表耗时较长。

因需要对这些大数据量报表进行预处理，使用客户可以快速的对大数据量报表进行预览分析，无需等待。

下面以定时刷新即席查询报表“汽车销售情况分析-演示示例”为例，“汽车销售情况分析-演示示例”报表ID为：I40288453016667896789e42701668640ab395dba。我们可以选中资源右键选择属性获取节点ID。

资源属性 ✕

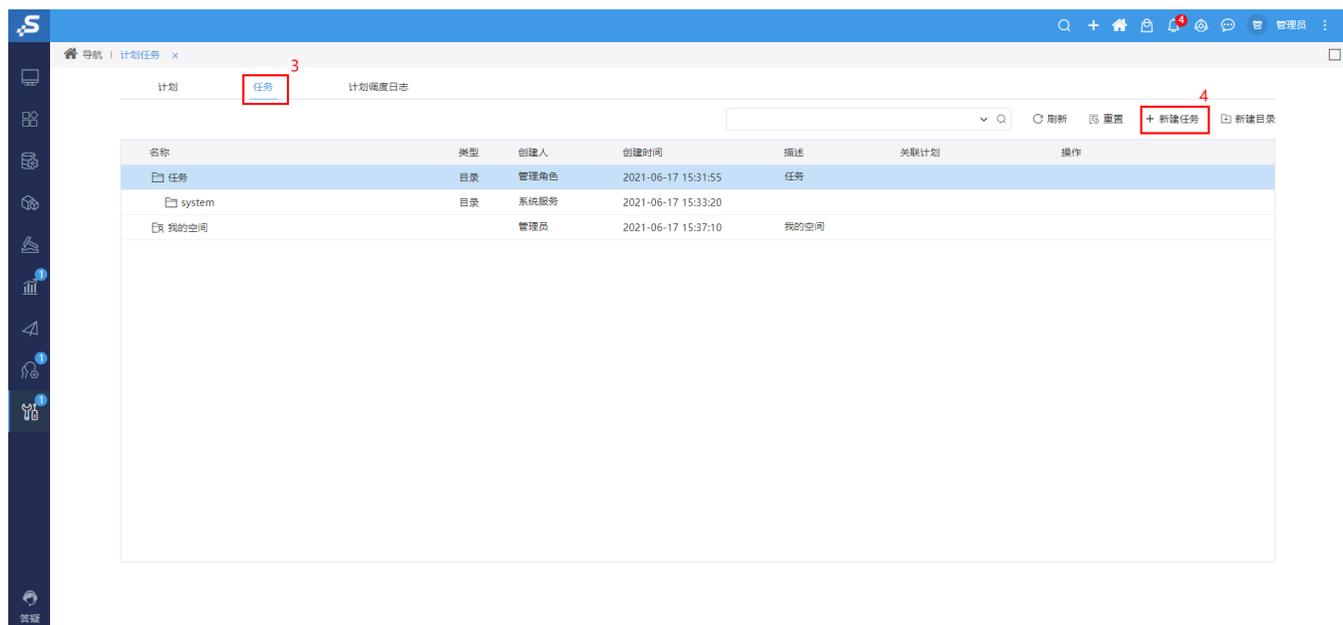
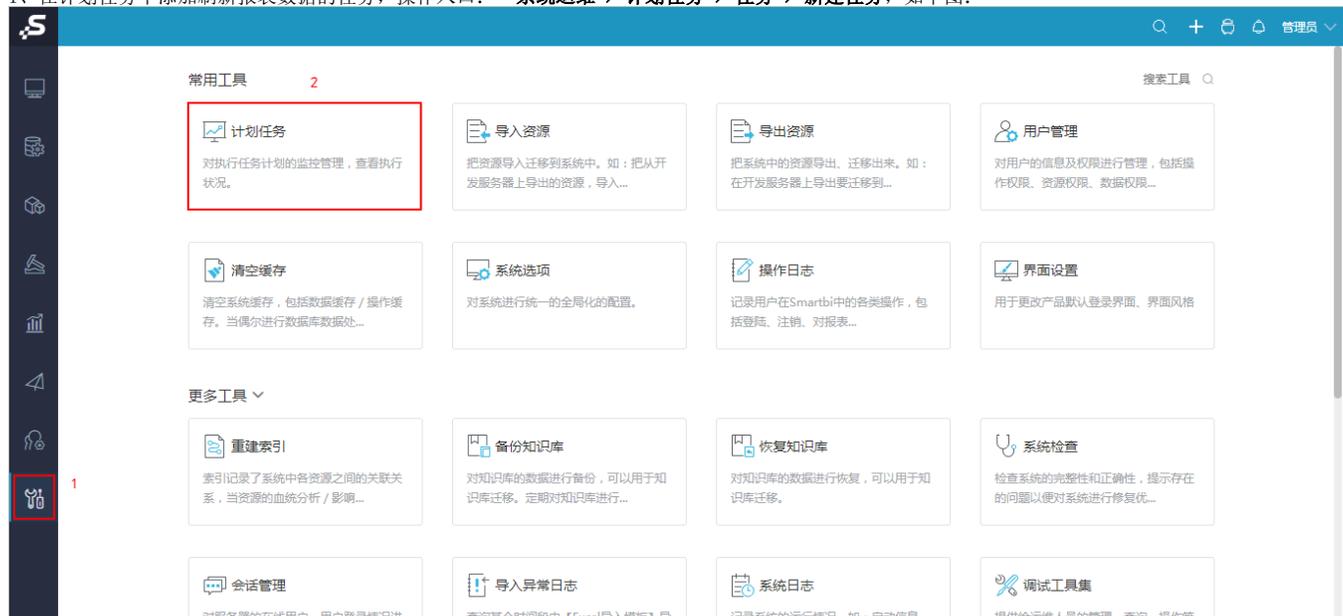
节点ID：	I40288453016667896789e42701668640ab395dba
名字：	汽车销售情况分析-演示示例
别名：	汽车销售情况分析-演示示例
路径：	公共空间\document\即席查询\汽车销售情况分析-演示示例
类型名：	即席查询
描述：	汽车销售情况分析-演示示例
创建人：	管理员
创建时间：	2019-07-31 13:54:37
显示终端：	<input checked="" type="checkbox"/> 电脑 <input type="checkbox"/> 平板 <input type="checkbox"/> 手机
图标：	 <input type="button" value="选择"/>

实现思路

在客户预览分析报表之前，通过计划任务定时对报表进行预处理。

操作步骤

1、在计划任务中添加刷新报表数据的任务，操作入口：**系统运维 > 计划任务 > 任务 > 新建任务**，如下图：



2、输入任务名称“定时刷新报表”，“任务类型”选择 **定制**，在“自定义设置”中添加刷新报表的代码，如下图：

新建任务
— □ ×

任务名称: * 定时刷新报表

任务别名: 定时刷新报表

任务类型: 定制

任务描述:

自定义设置

```

1 //导入将会使用的java类与java包
2 importClass(java.lang.System);
3
4 //打开报表资源。
5 var tempResourceHandle = execute('openResource', {
6   connector: connector,
7   reportId: "I40288453016667896789e42701668640ab395dba" //报表id
8 }).resourceHandle;
```

查看运行脚本(V)
测试运行(T)
保存(S)
关闭(C)

```

//javajava
importClass(java.lang.System);

//
var tempResourceHandle = execute('openResource', {
  connector: connector,
  reportId: "I40288453016667896789e42701668640ab395dba" //id
}).resourceHandle;
```

3、再新建计划，实现定时刷新报表。操作入口 **系统运维 > 计划任务 > 计划 > 新建计划**，如下图：

名称	任务类型	简述	当前状态	下次执行时间	上次执行信息	执行时长	操作
清除报表耗时分析日志	自定义	从 2018-06-04 起，每 1 天，02:00	成功	2022-02-15 02:00	2022-02-14 02:00 执行 成功。	0s	操作
每天凌晨6点执行	多任务	从 2010-01-10 起，每 1 天，06:00	停止			0s	
每天自动备份	自定义	从 2015-06-17 起，每 1 天，05:00	停止			0s	
清理透视分析临时表	自定义	从 2015-08-27 起，每 1 天，06:00	成功	2022-02-14 06:00	2022-02-13 06:00 执行 成功。	0s	
统计资源访问次数	自定义	从 2012-07-29 起，每 1 天，02:00	成功	2022-02-15 02:00	2022-02-14 02:00 执行 成功。	0s	
清除SQL日志	自定义	从 2018-06-04 起，每 1 天，02:00	成功	2022-02-15 02:00	2022-02-14 02:00 执行 成功。	0s	
每天凌晨2点更新字段缓存	自定义	从 2018-06-04 起，每 1 天，02:00	成功	2022-02-15 02:00	2022-02-14 02:00 执行 成功。	0s	
清除垃圾数据	自定义	从 2021-11-01 起，每 1 天，02:00	成功	2022-02-15 02:00	2022-02-14 02:00 执行 成功。	0s	
每天下午2点更新社区活动信息	自定义	从 2021-11-24 起，每 1 天，14:00	成功	2022-02-14 14:00	2022-02-13 14:00 执行 成功。	1s	

4、如计划名称为“定时刷新报表_计划”，“待执行任务”中添加已建好的任务 **定时刷新报表_计划**，如下图所示。再根据项目需求设置定时计划的“触发类型”、“间隔类型”、“运行设置”。

新建计划

计划基本信息

计划名称：* 定时刷新报表_计划

计划别名： 定时刷新报表_计划

计划描述：

待执行任务：* 定时刷新报表 选择任务

触发类型：* 时间

间隔类型：* 一次性 是否启用 是否指定生效范围

生效范围： 从 到

运行设置：* 触发时间：* 12 : 00 (例如13:30)

运行日期：* 2019-11-21 12:00

执行设置

执行用户： 计划创建者 特定用户

失败重试机制： 重试次数 0 次 重试间隔： 0 分钟

手动执行(M) 复制(D) 保存(S) 关闭(C)

选择任务

任务

- 任务
 - system
 - workflow
 - 示例
 - 定时刷新报表
 - 我的空间

资源名称：
定时刷新报表

确定(O) 取消(C)

注：如果资源是透视分析，以上任务代码不适用，请使用以下代码。

```
importPackage(Packages.smartbi.net.sf.json);
importPackage(Packages.java.lang);
//I4028839a015f575f575f7f5f015f576d868d0058id
var insight = connector.remoteInvoke("InsightService","openQuery",
["I4028839a015f575f575f7f5f015f576d868d0058",""]);
var result = JSONObject.fromObject(insight.getResult());
var clientId = result.getString("clientId");
logger.info(clientId);
var pageNo = new Integer(0);
connector.remoteInvoke("InsightService","refreshData",[clientId,pageNo]);
connector.remoteInvoke("InsightService","closeQuery",[clientId]);//;
```