

报表导出不同数据并发送给对应用户

应用场景
在计划任务的应用中，对于导出的报表数据，通过邮件发给多个用户时，都是同一份数据。实际中有可能需要对数据进行控制，比如同一个报表，同时发送给不同的用户，需要限制用户只能够看到各自权限内的数据。

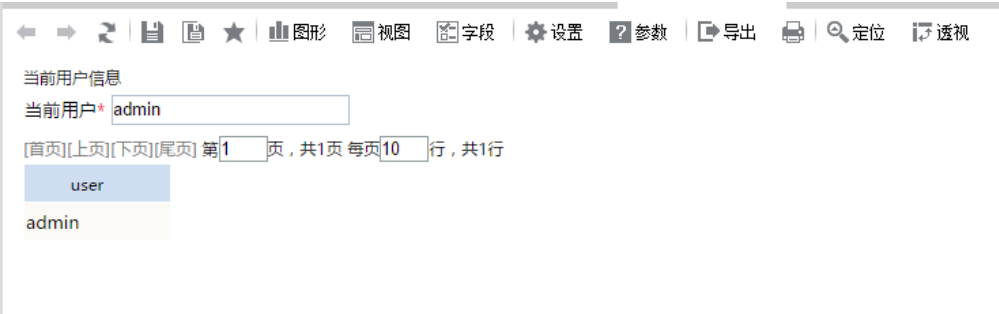
注：此示例只支持“灵活分析”报表。

- 实现步骤
 - 任务代码
 - (1) 通过SQL查询出需要发送邮件的用户名称
 - (2) 根据用户名称获取用户属性：email
 - (3) 通过设置用户名参数
 - (4) 打开报表
 - (5) 执行导出
 - (6) 发送邮件
 - (7) 批量发送邮件
 - 附完整的定制任务代码
- 更多
 - 注
- 使用自定义定制任务小技巧

实现步骤

对于导出报表的数据权限控制，具体的实现步骤是：

1. 首先创建好报表资源，报表上有一个接收用户名的参数，根据用户名参数，能够筛选出该用户所能查看的数据。



2. 在数据库中维护一张表，这张表记录了要接收邮件的用户的名称。比如创建表t_user, 其中包含username字段。这个字段记录了需要接收邮件的用户名称。

	username
☐	张三
☐	李四
☒	(NULL)

(3) 在用户管理中，为用户添加一个扩展属性，如email，并在该项中设置用户的邮箱地址

增加用户扩展属性，可参考：<https://history.wiki.smartbi.com.cn/pages/viewpage.action?pageId=27002165>

用户信息	角色列表	用户所属组
名称：*	张三	
别名：		
密码：		
确认密码：		
描述：		
启用用户：	☒ 是 ☐ 否	
email	zhangsan@smartbi.com.cn	
是否强制修改密码	☒ 是 ☐ 否	

4. 创建一个任务，任务类型选择“定制”。

通过以下代码，获取需要接收邮件的用户的名称和邮箱。打开报表时，动态设置用户名参数，并将该用户名参数的报表数据批量发送邮件给指定用户。

任务基本信息
任务名称：* AA
任务别名：
任务描述：
任务类型： 定制

自定义设置
<pre>1 /** 2 * 以下代码需要根据实际使用场景进行设置。设置的地方主要有两个： 3 * 一、数据源与SQL，此处设置是为了获取邮件发送的对象，即工号。 4 * 二、报表参数信息，此处是设置准备发送报表的参数信息。 5 */ 6 importPackage(Packages.java.io); 7 importPackage(Packages.java.lang); 8 importPackage(Packages.java.util); 9 importPackage(Packages.smartbi.usermanager); 10 importPackage(Packages.smartbi.sdk); 11 importPackage(Packages.smartbi.sdk.service.user); 12 importPackage(Packages.smartbi.sdk.service.datasource); 13 var usrManagerService = new UserManagerService(connector); 14 var datasrcService = new DataSourceService(connector); 15 //设置报表ID 16 var reportID = 'I2c9019810158fc12fc1224c00158fc2e7937006d'; 17 //开始设置数据源与SQL----- 18 var sql = "SELECT JRYH FROM dyxx"; // 该SQL语句获取需要邮件推送的用户名称 19 var dsId = "DS.邮件推送"; //上一句SQL运行的数据源ID 20 //结束设置数据源与SQL----- 21 var title = "%TASKNAME_%DATE%TIME"; //邮件标题，默认为：任务名称_日期时间 22 var text = "这是一个系统自动发送的邮件，所含附件是您订阅的报表"; //邮件正文内容 23 var dsId = "DS.邮件推送"; //邮件正文内容</pre>

具体代码可见下方

任务代码

以下为代码中的关键代码片段说明

（1）通过SQL查询出需要发送邮件的用户名称

```
var dsId = "DS.user";//数据源ID

var sql = "SELECT userName FROM t_user";

var gridDataGroup = datasrcService.executeNoCacheable(dsId, sql);

for (var i = 0; i < gridDataGroup.getRowsCount(); i++) {

    var userName = gridDataGroup.get(i, 0).getValue();

    .....

}
```

（2）根据用户名称获取用户属性：email

```
//根据用户名获取用户对象

var user = usrManagerService.getUserByName(userName);

//根据key值获取用户扩展属性，此处获取用户的email地址

var email = usrManagerService.getUserAttribute(user.getId(), "email").getValue();
```

（3）通过设置用户名参数

```
//{id:"OutputParameter.ID.",value:,displayValue:}

var reportParamSetting = [{id:"OutputParameter.I2c9019810158fc12fc1224c00158fc2e21c50069.当前用户",value:userName,displayValue:userName}]

//"$ $"

如： {id:"OutputParameter.I2c9019810158fc12fc1224c00158fc2e21c50069.测试",value:"$默认值$",displayValue:"$默认值$"}


```

（4）打开报表

//打开报表的方法基本固定，无需作出更改。

```
var openReportOutput = execute('openResource', {
    connector: connector,
    reportId: reportID
});
```

（5）执行导出

```
var exportReportOutput = execute('exportResource', {
    connector: connector,
    resourceHandle: openReportOutput.resourceHandle,
    paramSettingIterator: enumerateParamValuesOutput.getParamValues(),
    exportSetting: {"delimiter": "", "pathId": "", "fileNameSufParams": false},
    taskName: taskName,
    exportType: "EXCEL2007"
});
```

（6）发送邮件

```
var title = "%TASKNAME_%DATE%TIME";//_
var text = "";//
var doZip = true;//
var picInMail = false;//
```

```

var sendToMailOutput = execute('sendToMail', {
    connector: connector,
    taskName: taskName,
    paramValueMap: null,
    files: [outputFile],
    //设置邮件
    sendSetting: {"mailList":email,"doZip":doZip,"title":title,"text":text,"picInMail":picInMail}
});

```

(7) 批量发送邮件

通过遍历第1步获取到的用户名称，重复执行第2、3、4、5、6步即可完成批量发送邮件。

附完整的定制任务代码

```

/**
 *
 * SQL
 *
 */
importPackage(Packages.java.io);
importPackage(Packages.java.lang);
importPackage(Packages.java.util);
importPackage(Packages.smartbi.usermanager);
importPackage(Packages.smartbi.sdk);
importPackage(Packages.smartbi.sdk.service.user);
importPackage(Packages.smartbi.sdk.service.datasource);
var usrManagerService = new UserManagerService(connector);
var datasrcService = new DataSourceService(connector);
//ID
var reportID = 'I4028818a015c345e345e04fd015c8076cbda0c41';
//SQL-----
var sql = "SELECT userName FROM t_user"; // SQL
var dsId = "DS._"; //SQLID
//SQL-----
var title = "%TASKNAME_%DATE%TIME";//_
var text = "";//
var doZip = true;//
var picInMail = false;//
var gridDataGroup = datasrcService.executeNoCacheable(dsId, sql);
for (var i = 0; i < gridDataGroup.getRowsCount(); i++) {
    var userName = gridDataGroup.get(i, 0).getValue();

    //
    var user = usrManagerService.getUserByName(userName);
    //keyemail
    var email = usrManagerService.getUserAttribute(user.getId(), "email").getValue();

    //-----
    //{id:"OutputParameter.ID.",value:,,displayValue:}
    //"$$"
    var reportParamSetting = [
        {id:"OutputParameter.I4028818a015c345e345e04fd015c8076bf830c3d.",value:userName,displayValue:userName}
    ]
    //-----

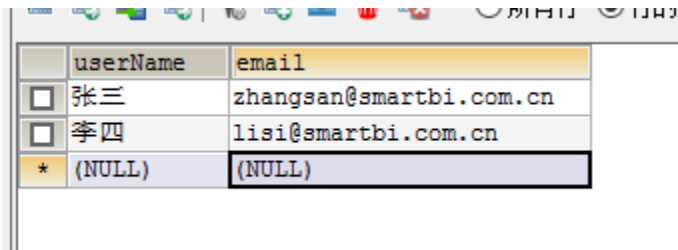
    //-----
    var openReportOutput = execute('openResource', {
        connector: connector,
        reportId: reportID
    });
    var enumerateParamValuesOutput = execute('enumerateParamValues', {
        resourceHandle: openReportOutput.resourceHandle,
        byEveryParam: false,
        paramsSetting: reportParamSetting
    });
    var exportReportOutput = execute('exportResource', {
        connector: connector,

```

```
        resourceHandle: openReportOutput.resourceHandle,
        paramSettingIterator: enumerateParamValuesOutput.getParamValues(),
        exportSetting: {"delimiter":"","pathId":"","fileNameSufParams":false},
        taskName: taskName,
        exportType: "EXCEL2007"
    });
    openReportOutput.resourceHandle.close();
    var outputFile = exportReportOutput.file;
    var sendToMailOutput = execute('sendToMail', {
        connector: connector,
        taskName: taskName,
        paramValueMap: null,
        files: [outputFile],
        //
        sendSetting: {"mailList":email,"doZip":doZip,"title":title,"text":text,"picInMail":picInMail}
    });
}
```

更多

另外，需要接收邮件的用户及其邮箱地址也可直接存放在同一张表中。如下：



	userName	email
☐	张三	zhangsan@smartbi.com.cn
☐	李四	lisi@smartbi.com.cn
☒	(NULL)	(NULL)

完整代码如下：

```

/**
 *
 * SQL
 *
 */
importPackage(Packages.java.io);
importPackage(Packages.java.lang);
importPackage(Packages.java.util);
importPackage(Packages.smartbi.usermanager);
importPackage(Packages.smartbi.sdk);
importPackage(Packages.smartbi.sdk.service.user);
importPackage(Packages.smartbi.sdk.service.datasource);
var usrManagerService = new UserManagerService(connector);
var datasrcService = new DataSourceService(connector);
//ID
var reportID = 'I4028818a015c345e345e04fd015c8076cbda0c41';
//SQL-----
var sql = "SELECT userName,email FROM t_user"; // SQL
var dsId = "DS._"; //SQLID
//SQL-----
var title = "%TASKNAME_%DATE%TIME";//_
var text = "";//
var doZip = true;//
var picInMail = false;//
var gridDataGroup = datasrcService.executeNoCacheable(dsId, sql);
for (var i = 0; i < gridDataGroup.getRowsCount(); i++) {
    var userName = gridDataGroup.get(i, 0).getValue();
    var email = gridDataGroup.get(i,1).getValue();

//-----
    //{id:"OutputParameter.ID.",value:,displayValue:}
    //"$ $"
    var reportParamSetting = [
        {id:"OutputParameter.I4028818a015c345e345e04fd015c8076bf830c3d.",value:userName,displayValue:userName}
    ]
//-----

//-----
    var openReportOutput = execute('openResource', {
        connector: connector,
        reportId: reportID
    });
    var enumerateParamValuesOutput = execute('enumerateParamValues', {
        resourceHandle: openReportOutput.resourceHandle,
        byEveryParam: false,
        paramsSetting: reportParamSetting
    });
    var exportReportOutput = execute('exportResource', {
        connector: connector,
        resourceHandle: openReportOutput.resourceHandle,
        paramSettingIterator: enumerateParamValuesOutput.getParamValues(),
        exportSetting: {"delimiter":"","pathId":"","fileNameSufParams":false},
        taskName: taskName,
        exportType: "EXCEL2007"
    });
    openReportOutput.resourceHandle.close();
    var outputFile = exportReportOutput.file;
    var sendToMailOutput = execute('sendToMail', {
        connector: connector,
        taskName: taskName,
        paramValueMap: null,
        files: [outputFile],
        //
        sendSetting: {"mailList":email,"doZip":doZip,"title":title,"text":text,"picInMail":picInMail}
    });
}

```

注

发送邮件前需要注意在系统选项中配置发件箱

定制管理 定制发邮件 系统选项 ×

公共设置 用户管理设置 查询设置 灵活分析组合分析设置 多维分析设置 页面设置 缓存设置 电子表格设置 透视分析设置

移动端首页目录id:

移动端跑马灯: 设置

邮件设置

SMTP服务器地址: 示例值(smtp.live.com)

发送用户地址:

帐号:

密码:

文件名源字符集:

文件名目标字符集:

索引设置

使用自定义定制任务小技巧

在使用可视化界面配置任务时，如果大部分都能通过配置达到需求，而只有一小部分无法配置实现，可以考虑先通过可视化配置界面配置好，然后通过查看运行脚本，将里面的脚本代码复制出来，然后即可根据里面的脚本代码作定制修改。

(1) 配置好以后查看运行脚本

定制管理 定制发邮件 系统选项 发邮件 ×

任务基本信息

任务名称: * 发邮件

任务别名: 发邮件

任务描述:

任务类型: 报表导出

报表设置 导出设置

导出格式: * EXCEL

导出类型: * 邮件

动态文件名: ☒ 是否添加参数值 (枚举参数时文件名必须添加参数值)

收信人: * 925787152@qq.com (请以分号分隔多个收件人) 压缩文件: ☒

邮件标题: %TASKNAME_%DATE%TIME

邮件正文: 这是一个系统自动发送的邮件，所含附件是您订阅的报表

图片作为正文内容: ☐

配置好以后查看运行脚本

查看运行脚本(V) 测试运行(T) 保存(S) 关闭(Q)

(2) 将运行脚本复制出来

运行脚本

```
1
2 var reportID = 'I2c9019810158fc12fc1224c00158fc2e7937006d';
3 var reportParamSetting = [
4   {id:"OutputParameter.I2c9019810158fc12fc1224c00158fc2e21c50069.当前用户",value:"admin",displayValue:"admin"},
5   {id:"OutputParameter.I2c9019810158fc12fc1224c00158fc2e21c50069.测试",value:"yuexiu",displayValue:"yuexiu"}
6 ];
7 var openReportOutput = execute('openResource', {
8   connector: connector,
9   reportId: reportID
10 });
11 var enumerateParamValuesOutput = execute('enumerateParamValues', {
12   resourceHandle: openReportOutput.resourceHandle,
13   byEveryParam: false,
14   paramsSetting: reportParamSetting
15 });
16 var exportReportOutput = execute('exportResource', {
17   connector: connector,
18   resourceHandle: openReportOutput.resourceHandle,
19   paramSettingIterator: enumerateParamValuesOutput.getParamValues(),
20   exportSetting: {"delimiter": ",", "pathId": "", "fileNameSufParams": true},
21   taskName: taskName,
22   exportType: "EXCEL2007"
23 });
24 openReportOutput.resourceHandle.close();
25 var outputFile = exportReportOutput.file;
26 var sendToMailOutput = execute('sendToMail', {
27   connector: connector,
28   taskName: taskName,
29   paramValueMap: null
30 }
```

取消(C)

(3) 新建一个任务，将上一步复制出来的脚本代码粘贴到定制代码中，然后作出定制修改即可。

定制管理 定制发邮件 × 系统选项 发邮件

任务基本信息

任务名称: * 定制发邮件

任务别名: 定制发邮件

任务描述:

任务类型: 定制

自定义设置

```
1 // **
2 * 以下代码需要根据实际使用场景进行设置。设置的地方主要有两个:
3 * 一、数据源与SQL，此处设置是为了获取邮件发送的对象，即工号。
4 * 二、报表参数信息，此处是设置准备发送报表的参数信息。
5 */
6
7 importPackage(Packages.java.io);
8 importPackage(Packages.java.lang);
9 importPackage(Packages.java.util);
10 importPackage(Packages.smartbi.usermanager);
11 importPackage(Packages.smartbi.sdk);
12 importPackage(Packages.smartbi.sdk.service.user);
13 importPackage(Packages.smartbi.sdk.service.datasource);
14 var userManagerService = new UserManagerService(connector);
15 var datasrcService = new DataSourceService(connector);
16
17 //设置报表ID
18 var reportID = 'I2c9019810158fc12fc1224c00158fc2e7937006d';
19
20 //开始设置数据源与SQL-----
21 var sql = "SELECT TRIM FROM dyxx"; // 该SQL语句数据需要邮件推送的用户名称
22 var dsId = "DS_邮件推送"; //上一句SQL运行的数据源ID
23 //结束设置数据源与SQL-----
24
```

查看运行脚本(V) 测试运行(T) 保存(S) 关闭(C)