流程监控

流程监控是对已创建的流程进行查看及监控。

- 功能入口
- 流程监控
- 流程上的宏
- 权限控制
- 其它功能
- 特殊场景

功能入口

入口1：在“系统导航栏”选择 **系统运维** ，进入“系统运维界面主菜单”选择 **流程监控。**



入口2：拥有流程权限后，在系统主界面右上角的下拉菜单中，可以看见“流程监控”菜单。点击菜单，进入流程监控主界面。



流程监控

在流程监控界面中，通过查询选择要监控的流程，展示流程对应的参数。

- 不选择参数，展示所有的和当前流程相关的流程实例；
- 选择参数，则查询展示与参数值相关的流程实例。



流程上的宏

宏事件往往用于帮助用户对数据进行处理。

例如有些用户执行汇总的时候，需要执行复杂的存储过程进行数据汇总；或者在发布的时候，将数据从一个库表复制到另外一个库表；或者用户需要将当前流程的审核状态同步到业务库中，这些需求都可以通过宏来实现。

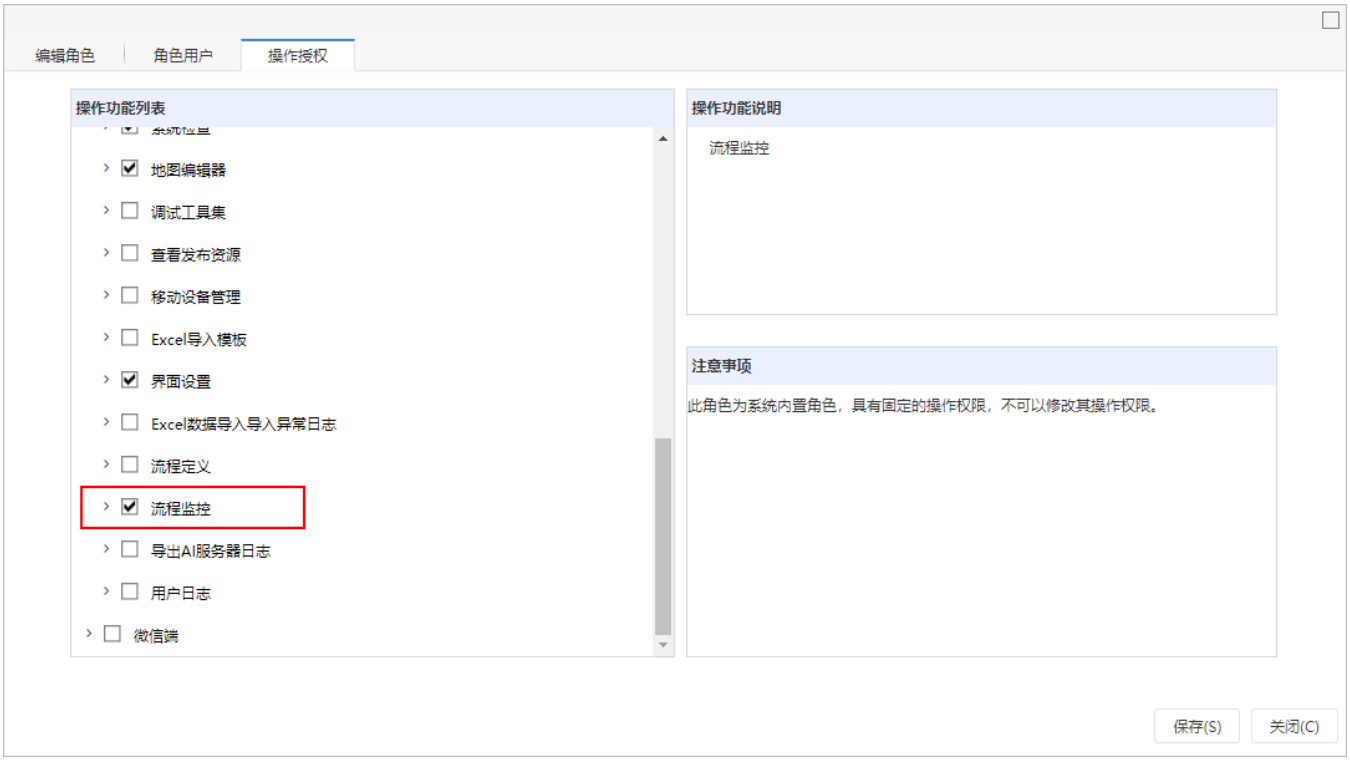
在流程定义上，右键编辑宏，可以进入到流程宏的编辑界面。流程只支持客户端宏，不支持服务器端宏。流程宏基本上与流程操作按钮对应，且按执行时间，分为执行按钮操作之前和执行操作按钮之后触发。流程支持的宏包括：

- beforeCommit:执行提交之前
- afterCommit:执行提交之后
- beforeVerify:执行校验之前
- afterVerify:执行校验之后
- beforeSummary:执行汇总之前
- afterSummary:执行汇总之后
- beforeAudit:执行审核之前
- afterAudit:执行审核之后
- beforeBackToPre:执行回退上一步之前
- afterBackToPre:执行回退上一步之后
- beforeBackToStart:执行回退到开始之前
- afterBackToStart:执行回到到开始之后
- beforeRelease:执行发布之前
- afterRelease:执行发布之后

特别的，在before类型的事件中，可以针对操作做一些条件判断，例如对业务库进行检查，当不满足条件的时候，可以通过抛出异常（throw “error”）的方法，阻止按钮操作的执行。

权限控制

要新建流程定义，需要流程定义操作权限，在“系统运维界面主菜单”上选择 **用户管理** ，点击 **操作授权> 流程监控**。



另外，流程定义属于资源，本身受资源权限控制。

其它功能

流程定义支持资源导入、导出操作，支持在资源上复制、拖动换目录，支持影响性分析和血统分析。

特殊场景

1、业务库中查询已经发布的数据

很多情况下，除了审核流程相关的用户可以看到未经过审核的数据，其它用户是不允许查询这些未审核数据的。而在业务库中，这两种数据往往混合在一起存放。这个时候需要在查询中加上对流程状态的判断。可是由于流程状态是存在于知识库中的，跨库的查询是无法支持的。

遇到这种情况，可以由实施人员，在发布事件中编写宏代码，使用DataSourceService的executeUpdate方法，执行sql语句或者执行存储过程，将流程状态同步到业务库中。

```
var util = jsloader.resolve("freequery.common.util");

var ret = util.remoteInvoke("DataSourceService", "executeUpdate", ["DS.数据源名称", "update XXXXX"]);
```

2、发布时将数据写入另一个库

有些时候，用户要求审核前的数据和审核后的数据不能同库。这和4.1的处理类似，可以由实施人员，在发布事件中编写宏代码，使用DataSourceService的executeUpdate方法，执行sql语句或者执行存储过程，将审核后的数据同步到业务库中。

3、汇总时执行运算逻辑

系统中的【汇总】操作，只是做了一个状态标志的处理，并没真正的对业务数据做汇总处理。业务数据的汇总处理逻辑，往往逻辑复杂且不尽相同。

因此，可以由实施人员，在流程的汇总事件中编写宏代码，使用DataSourceService的executeUpdate方法，执行sql语句或者执行存储过程，完成响应的汇总处理。示例代码如下：

```
var p1 = spreadsheetReport.paramPanelObj.findParamByName('DeptTree');
var p2 = spreadsheetReport.paramPanelObj.findParamByName('');
var util = jsloader.resolve("freequery.common.util");
//
var sql = "delete from factyear where dateid='"  + p2.value + "' and deptid='"  + p1.value + "'";
var ret = util.remoteInvoke("DataSourceService", "executeUpdate", ["DS.", sql]);
if  (ret && ret.succeeded) {
    //
var sql = "insert into factyear (deptid,dateid,metricid,val) "  +
        "select '"  + p1.value + "','"  + p2.value + "',metricid,sum(val) "  +
        " from factyear where dateid='"  + p2.value + "' and deptid<>'"  + p1.value +
        "'  group by metricid ";


    var ret = util.remoteInvoke("DataSourceService", "executeUpdate", ["DS.", sql]);
    if  (ret && ret.succeeded) {
        spreadsheetReport.doRefresh();
        return  true;
    }
}
```