**文档信息及修订记录**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **日期** | **版本** | **修订者** | | **修订说明** | **批准人** | |
| 2013-05-13 | V0.01 | | 杨金云 | 1，创建文档。2，增加按钮重命名章节。3，增加其它按钮组配置章节。 |  |
| 2013-07-09 | V0.02 | | 杨金云 | 增加组织资源扩展类使用说明 |  |
| 2013-09-20 | V0.03 | | 杨金云 | 完善子流程扩展类说明 |  |
| 2013-11-08 | V0.04 | | 杨金云 | 完善系统配置值 |  |
| 2014-07-14 | V0.05 | | 杨金云 | 1，增加点事件监听类。2，增加线条件扩展类。3，补充没有完善的章节。 |  |
|  |  | |  |  |  |
|  |  | |  |  |  |
|  |  | |  |  |  |
|  |  | |  |  |  |
|  |  | |  |  |  |
|  |  | |  |  |  |

目录

[一、 如何重命名流程按钮？ 4](#_Toc393975677)

[（1） 通过系统配置重命名流程按钮 4](#_Toc393975678)

[（2） 修改节点属性操作权限 5](#_Toc393975679)

[二、 常用系统配置 5](#_Toc393975680)

[（1） 哪些按钮显示在其它按钮组？ 6](#_Toc393975681)

[（2） 配置工作流按钮渲染模式 6](#_Toc393975682)

[（3） 分发签收之后是否显示转发按钮？ 8](#_Toc393975683)

[（4） 设置工作流返回上一个节点的『按钮名称』 8](#_Toc393975684)

[（5） 是否任意节点都能删除自己的意见？ 9](#_Toc393975685)

[（6） 流程管理员可以看到的按钮 9](#_Toc393975686)

[（7） 可以下载审批单文件的角色 10](#_Toc393975687)

[（8） 是否显示转发按钮 10](#_Toc393975688)

[（9） 是否显示退回按钮 11](#_Toc393975689)

[（10） 默认显示“打印审批单”按钮的角色 11](#_Toc393975690)

[（11） 并发流：是否只显示汇合后的最后一条待办 11](#_Toc393975691)

[三、 催办单相关系统配置 12](#_Toc393975692)

[（1） 催办单编号前缀 12](#_Toc393975693)

[（2） 以那个用户的身份来发送催办单？ 12](#_Toc393975694)

[四、 并发流多级并发设置 13](#_Toc393975695)

[（1） 多级并发功能配置 13](#_Toc393975696)

[（2） 汇合功能配置 13](#_Toc393975697)

[（3） 示例 14](#_Toc393975698)

[（4） 注意事项 15](#_Toc393975699)

[五、 扩展类 15](#_Toc393975700)

[（1） 组织资源过滤扩展类 15](#_Toc393975701)

[（2） 节点事件监听类 18](#_Toc393975702)

[（3） 线条件判断扩展类 19](#_Toc393975703)

[（4） 流程按钮权限处理扩展类 21](#_Toc393975704)

[（5） 监听流程启动办结 22](#_Toc393975705)

[（6） 线流经事件 23](#_Toc393975706)

# 如何重命名流程按钮？

在项目实施过程中，经常会遇到平台设置的按钮名称与客户要求不一致的情况，例如在平台中的按钮“分发”，在中华保险项目被称为“传阅”。

为了解决这个问题，平台提供两种方式覆盖平台配置的按钮名称。第一，增加系统配置，所有的流程都改成系统配置指定的名称。第二种，修改工作流节点定义，在操作权限上设置按钮的新名称。第二种方式优先级最高，可以覆盖第一种方式设置的名称。

## 通过系统配置重命名流程按钮

首先增加系统配置，指定哪些按钮的名称可以被重命名。

|  |  |
| --- | --- |
| **配置项名称** | **配置项值** |
| 名称 | 需要重命名的按钮ACT\_CODE |
| KEY | SY\_WF\_BTN\_RENAME\_CODE |
| 值 | cmWfTracking,cmSetAttention,cmFavorite,cmZhuanFa |
| 说明 | 多个按钮之间使用英文逗号分隔 |

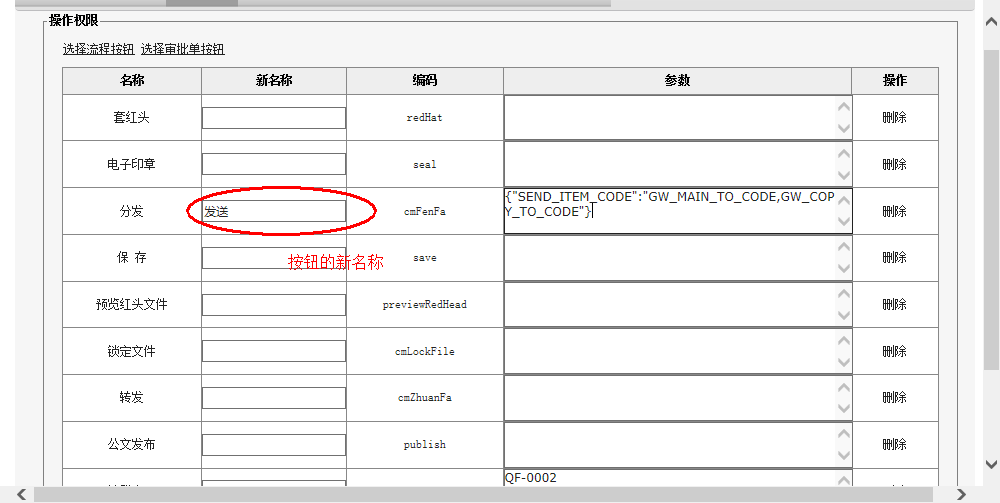
其次，增加将指定按钮重命名成其它名称。

|  |  |
| --- | --- |
| **配置项名称** | **配置项值** |
| 名称 | 需要重命名按钮的ACT\_CODE和新名称 |
| KEY | SY\_WF\_BTN\_RENAME\_cmFenFa |
| 值 | 传阅 |
| 说明 | 绿色标识部分为固定值，是KEY的前缀，不能修改；黄色标识的部分为按钮的Act\_Code。配置值为按钮的新名称。 |

## 修改节点属性操作权限

打开工作流指定节点的的“操作权限”卡片，操作列表有一列“新名称”，在新名称列对应的输入框中，填写节点的新名称即可。如果不需要改变按钮名称，则可以不填写，让输入框保持为空。

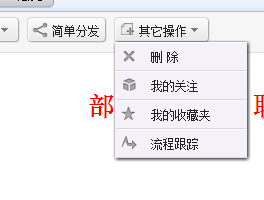
这种重命名按钮方式的优先级最高，可以覆盖第一种方式设置的名称。



# 常用系统配置

## 哪些按钮显示在其它按钮组？

在审批单流转过程中，有的节点有很多操作按钮，有些按钮不一定必须处理，例如：流程跟踪，我的收藏夹，我的关注，补登意见，删除等按钮。按照现有的按钮显示方式，不用处理的按钮与需要处理的按钮混杂在一起了，不需要处理的按钮对用户操作有些干挠。为了突出用户需要办理的按钮，增加一个功能，可以将不必须处理的按钮，合在一起，组成“其它操作”按钮，点击“其它操作”按钮，展开下拉菜单，显示组内的其它按钮。



如果要启动此功能，只需在系统配置中设置指定按钮的Act\_CODE即可。

|  |  |
| --- | --- |
| **配置项名称** | **配置项值** |
| 名称 | 放在其它操作按钮组中按钮 |
| KEY | SY\_WF\_BTN\_GROUP\_OTHER |
| 值 | cmWfTracking,cmSetAttention,cmFavorite,cmZhuanFa |
| 说明 | 配置值为指定按钮的ACT\_CODE，多个CODE之间使用英文逗号分隔 |

## 配置工作流按钮渲染模式

《工作流按钮渲染模式》说明作者：王晨

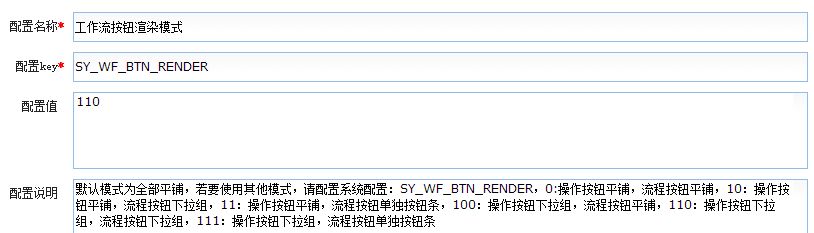
(1)概念定义：

操作按钮：一些常用操作，如转发，我的关注，删除，我的收藏夹，流程跟踪

流程按钮：如节点名称，返回某某，某某节点

(2)目前共支持六种渲染方式：

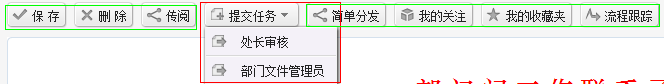
默认模式为全部平铺，若要使用其他模式，请配置系统配置：SY\_WF\_BTN\_RENDER，如下：



0:操作按钮平铺，流程按钮平铺，如下图



10：操作按钮平铺，流程按钮下拉组，如下图



11：操作按钮平铺，流程按钮单独按钮条，如下图

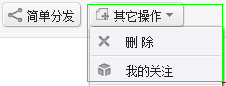




100：操作按钮下拉组，流程按钮平铺，如下图



110：操作按钮下拉组，流程按钮下拉组，如下图



111：操作按钮下拉组，流程按钮单独按钮条，如下图



注意：以上操作按钮启用下拉组渲染方式时，未定义分组的操作按钮仍将按平铺显示

(3)按钮生成顺序原则：

1、平铺按钮时按照后台送过来的顺序依次渲染。

2、下拉组按钮时多个组之间按照平铺的顺序，单个组包含的组按钮按照平铺时的顺序依次从上到下渲染，单个组在按钮条的顺序按照其包含的第一个组按钮平铺时出现的位置渲染（特例：“其他操作”组始终保持在按钮条的最后一个）。

3、独立按钮条中的按钮也按平铺时的顺序渲染。

## 分发签收之后是否显示转发按钮？

系统配置(公共)中设置指定。

|  |  |
| --- | --- |
| **配置项名称** | **配置项值** |
| 名称 | 分发签收之后是否显示转发按钮 |
| KEY | SY\_WF\_FENFA\_TRANS |
| 值 | true/false |
| 说明 | 分发签收之后是否显示转发按钮  true 显示  false 不显示 |

## 设置工作流返回上一个节点的『按钮名称』

系统配置(公共)中设置指定。

|  |  |
| --- | --- |
| **配置项名称** | **配置项值** |
| 名称 | 工作流返回上一个节点的 按钮名称 |
| KEY | SY\_WF\_BACK\_NODE\_NAME |
| 值 | 返回#NODE\_NAME#(#USER\_NAME#) |
| 说明 | #NODE\_NAME# 会被替换成上一个节点的名称  #USER\_NAME# 会被替换成该节点最后办理的人的名字  如果不填，按钮就是 返回#USER\_NAME# |

## 是否任意节点都能删除自己的意见？

系统配置(公共)中设置指定。

|  |  |
| --- | --- |
| **配置项名称** | **配置项值** |
| 名称 | 是否任意节点都能删除自己的意见 |
| KEY | CM\_MIND\_DEL\_SELF |
| 值 | true/false |
| 说明 | 当前自己正在办理的文件，能否删除之前自己所填写的意见  true: 能  false： 不能 |

## 流程管理员可以看到的按钮

有些单位的流程管理员，嫌自己看到的按钮太多了，不想显示那么多，通过这个配置就可以设置管理员能看到的按钮。

|  |  |
| --- | --- |
| **配置项名称** | **配置项值** |
| 名称 | 流程管理员能够显示的按钮 |
| KEY | SY\_WF\_ADMIN\_BTN |
| 值 | delete,cmWfDataMgr |
| 说明 | 流程管理员能看到的与流程管理有关的管理按钮。值为按钮的code，多个按钮之间使用英文的逗号。 |

目前流程管理员可以使用的管理按钮为：

|  |  |
| --- | --- |
| **按钮Code** | **按钮名称** |
| cmWfDataMgr | 数据管理 |
| save | 保存 |
| delete | 删除 |
| undoFinish | 取消办结 |
| finish | 办结 |
| cmPrintAudit | 打印审批单 |
| cmFenFa | 分发 |

## 可以下载审批单文件的角色

|  |  |
| --- | --- |
| **配置项名称** | **配置项值** |
| 名称 | 流程管理员能够显示的按钮 |
| KEY | SY\_WFE\_FILE\_DOWNLOAD\_ROLE |
| 值 | SYS\_ADMIN |
| 说明 | 配置值为角色ID。此角色用户可以下载所有审批单的文件。 |

## 是否显示转发按钮

|  |  |
| --- | --- |
| **配置项名称** | **配置项值** |
| 名称 | 是否显示转发按钮 |
| KEY | SY\_WF\_FENFA\_TRANS |
| 值 | false |
| 说明 | 默认为true |

## 是否显示退回按钮

|  |  |
| --- | --- |
| **配置项名称** | **配置项值** |
| 名称 | 是否显示转发按钮 |
| KEY | WFE\_IS\_SHOW\_TUIHUI |
| 值 | false |
| 说明 | 默认为true |

## 默认显示“打印审批单”按钮的角色

|  |  |
| --- | --- |
| **配置项名称** | **配置项值** |
| 名称 | 默认显示“打印审批单”按钮的角色 |
| KEY | SY\_WFE\_PRINT\_AUDIT\_ROLE |
| 值 |  |
| 说明 | 配置值为：角色ID。此角色用户能打印所有审批单。 |

## 并发流：是否只显示汇合后的最后一条待办

|  |  |
| --- | --- |
| **配置项名称** | **配置项值** |
| 名称 | 是否只显示汇合后的最后一条待办 |
| KEY | CONF\_WF\_CONVERGE\_LAST\_TODO |
| 值 | true |
| 说明 | 默认值为false。False表示汇合点一直只显示最后一条待办，true表示汇合点等所有并发都返回之后才显示待办。 |

## 并发会签时，汇合点的提示信息

|  |  |
| --- | --- |
| **配置项名称** | **配置项值** |
| 名称 | 并发会签时，汇合点的提示信息 |
| KEY | SH\_WF\_HEAD\_MSG |
| 值 | 会签尚未完成，不能进行流转操作，可视情况手动选择终止会签，继续流转。 |
| 说明 |  |

## 是否只显示汇合后的最后一条待办

|  |  |
| --- | --- |
| **配置项名称** | **配置项值** |
| 名称 | 是否只显示汇合后的最后一条待办 |
| KEY | WF\_CONVERGE\_LAST\_TODO |
| 值 | true |
| 说明 | 值为true则表示最后一条才返回。False表示自动合并 |

# 催办单相关系统配置

## 催办单编号前缀

例如催办单的编号为：催办（2013）10号，则前缀为“催办”。

|  |  |
| --- | --- |
| **配置项名称** | **配置项值** |
| 名称 | 催办单编号前缀 |
| KEY | SY\_WFE\_REMIND\_CODE |
| 值 | 催办 |
| 说明 | 催办单编号的前缀，默认为“催办”。 |

## 以那个用户的身份来发送催办单？

|  |  |
| --- | --- |
| **配置项名称** | **配置项值** |
| 名称 | 以哪个用户身份来发送催办单 |
| KEY | SY\_WFE\_REMIND\_USER |
| 值 | admin |
| 说明 | 值为用户的登录名。 |

# 并发流多级并发设置

软虹工作流支持单次并发和多级并发功能，多级并发是指一个审批单并发给多个用户办理之后，在未汇合的情况下，还能再次并发给其它用户办理，而且可以为第二次并发设置独立的汇合点，只汇合第二次并发的节点实例。程序对并发及汇合的次数没有限制，由于数据库字段设置为200Byte，因此可以支持连续30次无汇合的并发。

多级并发功能的配置可以分为并发配置和汇合设置2部分：

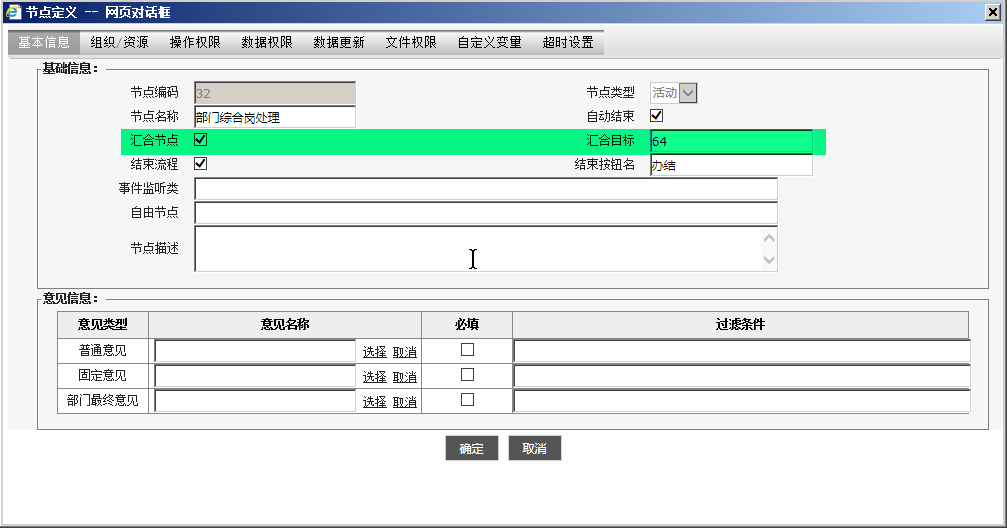
## 多级并发功能配置

多级并发配置与之前的一级并发配置方式并没有不同，只需要将线属性页面上的“是否并发”选中即可。

## 汇合功能配置

汇合功能配置与之前的配置方式有一点区别，除了将汇合节点属性的“汇合节点”选中之外，如果要分级汇合，还需要设置“汇合目标”的值。

“汇合目标”的值为需要被汇合的并发节点的ID，这个值如果为空，表示未指定，则汇合整个流程所有活动节点实例，如果不为空，则只汇合指定并发节点并发的活动节点实例。例如下图所示的配置，将会汇合并发节点64并发的所有实例。

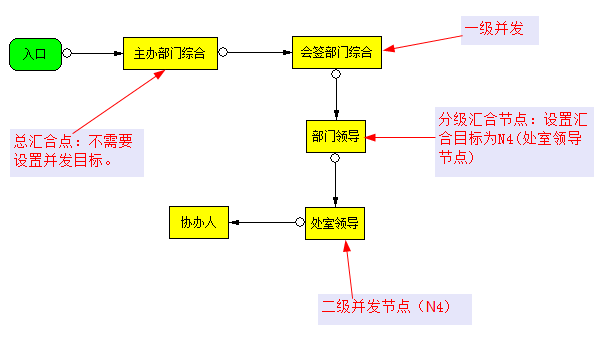


## 示例

下图为一个非常简单的二级并发流及汇合配置的例子。它有两个并发点“会签部门综合”和“处室领导”和两个汇合节点“主办部门综合”和“部门领导”。两个并发点的设置与一级并发的设置完全相同。

“主办部门综合”为总汇合点，只需要配置节点的“汇合节点”选中，“汇合目标”为空，表示可以汇合所有活动的节点实例。

“部门领导”节点为分级汇合点，配置了节点的“汇合节点”选中，“汇合目标”为N4。表示只汇合并发给处室领导节点的活动节点实例。



## 注意事项

配置工作流时，存在以下注意事项。

1. 所有的并发流程必须经过总汇合点之后才能发送给非并发区。如果有的节点实例不经过汇合点直接送到非并发区，可能会造成程序异常。

2， 总的汇合点最好不要设置“汇合目标”。可以避免由于流程定义混乱引起的并发流标记错误。

# 扩展类

## 组织资源过滤扩展类

当工作流组织资源定义功能提供的按照角色、部门、用户定义方式不能满足复杂业务需要时，允许开发人员使用扩展类，来处理特殊业务逻辑，返回特定的办理用户。

组织资源扩展类有2个接口类，它们实现的功能略有差别。

第一个接口：com.rh.core.wfe.resource.ExtendBinder。实现类要实现接口类的run方法，并返回结果对象（com.rh.core.wfe.resource.ExtendBinderResult）。

ExtendBinder接口类的详细说明如下：

|  |
| --- |
| /\*\*  \* 组织机构扩展过滤接口，用于处理不能通过“组织资源定义”功能设置接收用户的情况。 <br>  \* 在工作流送下一个节点时，先判断是否设置了扩展过滤接口，如果有则执行扩展过滤类，并取得最终接收人 ，<br>  \* 否则使用组织资源定义的结果，取得最终接收人。  \*  \* **@author** yangjy  \*/  **public** **interface** ExtendBinder {  /\*\*  \* 执行run方法，返回组织机构过滤结果。  \* **@param** currentWfAct 当前节点实例  \* **@param** nextNodeDef 下一节点的定义  \* **@return** 组织机构绑定结果对象ExtendBinderResult  \*/  ExtendBinderResult run(WfAct currentWfAct , WfNodeDef nextNodeDef);  } |

参考类

#从审批单的会签部门字段中选择用户

com.rh.core.wfe.resource.HuiQianExtWfBinder

#秘书找领导：从领导秘书 对应关系中选择用户

com.rh.core.wfe.resource.LeaderExtWfBinder

#领导找秘书：从领导秘书对应关系中选择用户

com.rh.core.wfe.resource.SecretorExtWfBinder

第二个接口com.rh.core.wfe.resource.GroupExtendBinder。通过该接口过滤的用户直接作为任务处理人向下流转，不送到前台选择。会为每个GroupBean生成一个任务实例，多用于抢占式任务。

参考类：

com.rh.oa.icbc.wfe.DajySpecialTypeExtWfBinder



**配置方式**

打开工作流节点（或线）属性，选中组织资源选项卡，在处理人扩展输入框中，输入实现类的类名。例如：com.rh.core.wfe.resource.HuiQianExtWfBinder。



如果想要传递参数给实现类，那么可以在类名后面加上2个逗号，逗号后面的值在实现类中可以取得。取得参数的方式如下：

|  |
| --- |
| String extCls = nextNodeDef.getStr("NODE\_EXTEND\_CLASS");  String configStr = "";  String[] classes = extCls.split(",,");  **if** (classes.length == 2) {  configStr = classes[1];  }  //如果参数为JSON字符串，则可以将json字符串转换成Bean  Bean configBean = JsonUtils.*toBean*(configStr); |

## 节点事件监听类

当流程从其它节点进入本节点时，会触发配置在本节点上的事件监听实现类。事件监听实现类的为抽象类的子类（com.rh.core.wfe.util.AbstractNodeEvent），具体的业务逻辑可以实现在抽象方法内。

AbstractNodeEvent抽象类有2个抽象方法，afterEnter和afterFinish。afterEnter方法在进入节点之后触发，afterFinish方法在节点结束之后触发。

|  |
| --- |
| /\*\*  \* 进入节点  \*  \* **@param** preWfAct 前一节点实例  \* **@param** currWfAct 当前节点实例  \* **@param** wfParam 参数Bean  \*/  **public** **abstract** **void** afterEnter(WfAct preWfAct ,  WfAct currWfAct ,  WfParam wfParam);    /\*\*  \* 节点结束  \*  \* **@param** currWfAct 当前节点  \* **@param** wfParam 参数Bean  \*/  **public** **abstract** **void** afterFinish(WfAct currWfAct ,  WfParam wfParam); |

**参考类**

#流程起草完成之后，根据用户身份自动送到不同的点。如：普通人员还在拟稿点，处长则送处长核稿节点。

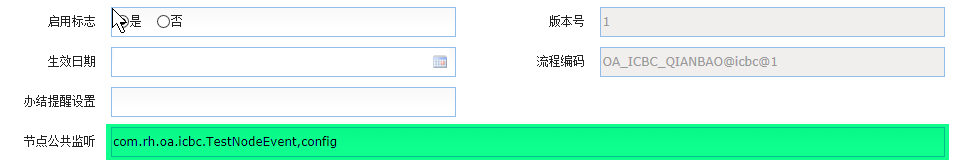
com.rh.oa.icbc.DsWfStartupLsn



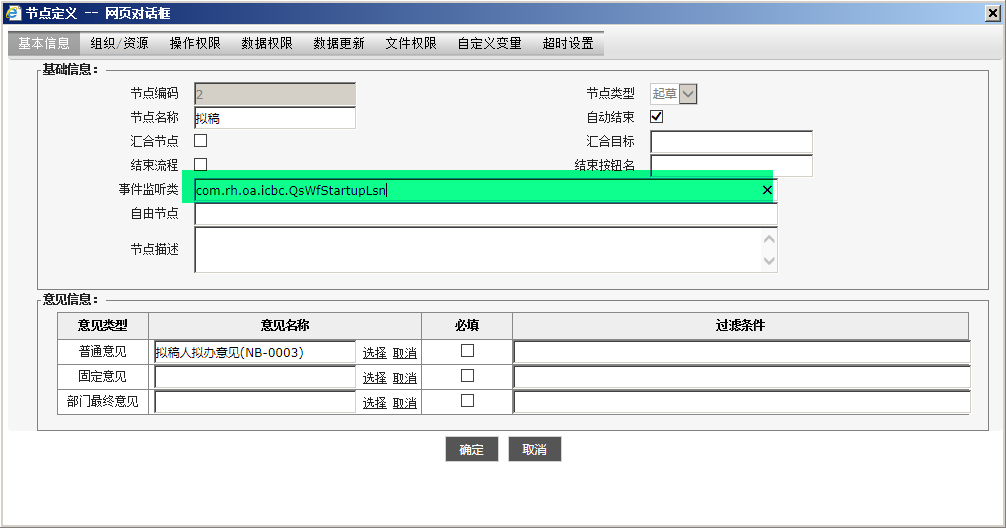
**配置方式**

节点事件监听类有两个配置位置：

第一个位置为流程定义界面，设置节点公共监听（如下图所示）。这个位置配置的监听类，在本流程的所有节点上都会执行。



第二个位置为：节点属性界面对话框🡪基本信息卡片页的事件监听类输入框中，配置事件监听实现类，这个位置配置的监听类只在本节点上执行。



## 线条件判断扩展类

如果简单条件表达式和复杂条件表达式不能满足需求时，工程师可以通过实现LineConditionExt（com.rh.core.wfe.condition.LineConditionExt）接口，来实现非常复杂的条件判断。

LineConditionExt接口只有一个check方法，如果允许送交则返回true，否则返回false。

|  |
| --- |
| /\*\*  \* **@param** wfAct 当前节点实例  \* **@param** doUser 办理用户  \* **@param** config 类名后面配置的字符串  \* **@return** 检验是否能通过当前点  \*/  **public** **boolean** check(WfAct wfAct ,  UserBean doUser ,  Bean lineBean ,  String config); |

**参考类**

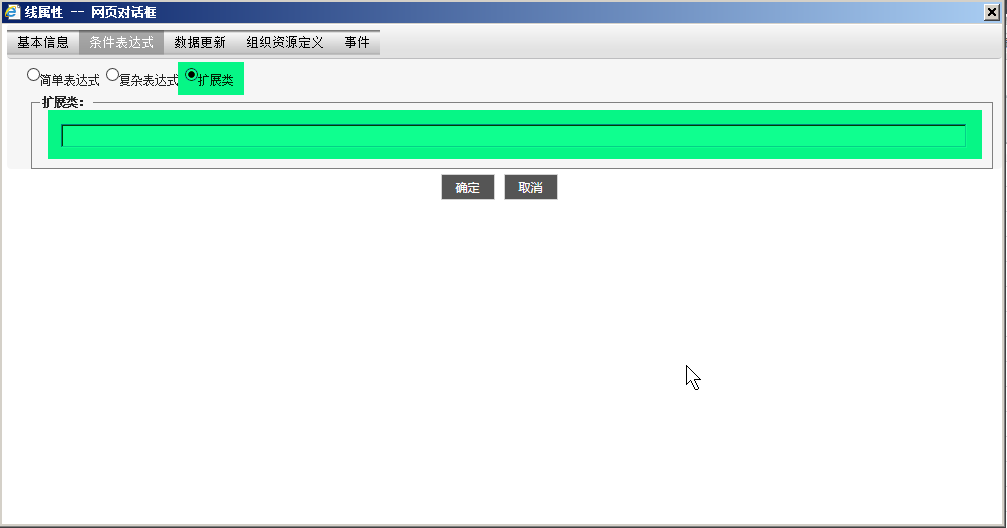
#根据档案借阅的种类来区分是否能走下一个节点。

com.rh.oa.icbc.serv.DajyLineCondition



**配置方式**

在线点属性界面对话框🡪条件表达式卡片页，选中扩展类，然后在事件监听类输入框中，配置事件监听实现类。实现类后面可以加上逗号，逗号后面的字符串会当作config参数传递给监听方法。



## 流程按钮权限处理扩展类

为了让审批单可以在非办理节点（例如已办、查看分发给我的文件等）也可以控制按钮的权限，流程引擎增加了一个扩展类com.rh.core.wfe.WfFilter接口，这个扩展类每次打开审批单时都会执行。在扩展类中可以根据流转状态（在办、已办、分发）和用户权限控制按钮在客户端的显示和隐藏。

|  |
| --- |
| /\*\*  \*  \* **@param** wfAct 当前流程节点实例  \* **@param** WfOutBean 工作流输出给前台的所有数据Bean。  \*  \*/  **void** doButtonFilter(WfAct wfAct, WfOut WfOutBean); |

**参考类**

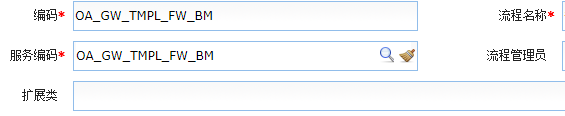
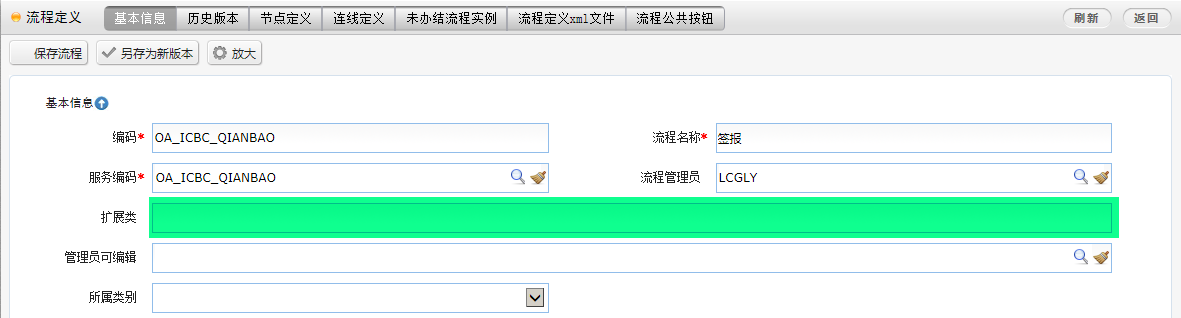
#从流程定义的节点上取出的 表单按钮 ， 根据实际情况进行过滤

com.rh.oa.gw.GwWfFliter



**配置方式**

在流程定义界面上输入扩展接口的实现类，如下图所示。



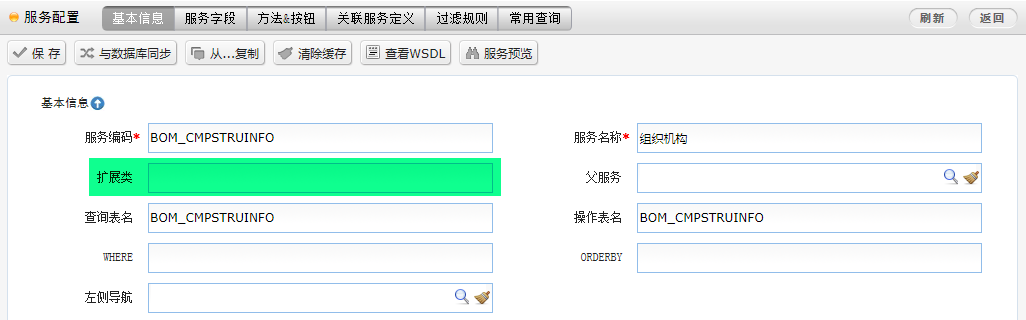
## 监听流程启动办结

在流程启动或办结时，在这个时刻如果需要特殊特殊业务逻辑，那么，可以在服务定义上配置的服务扩展类，在扩展类中override父类的beforeStart和beforeFinish。

|  |
| --- |
| /\*\*  \* 启动流程之前的拦截方法，由子类重载  \* **@param** paramBean 参数信息  \* **@param** dataBean 数据信息  \*/  **protected** **void** beforeStart(ParamBean paramBean, Bean dataBean) {  }    /\*\*  \* 办结流程之前的拦截方法，由子类重载  \* **@param** paramBean 参数信息  \*/  **protected** **void** beforeFinish(ParamBean paramBean) {  } |

**配置方式**

在服务配置🡪基本信息页面设置服务扩展类，服务扩展类必须是com.rh.core.serv.CommonServ类的子类。



## 线流经事件

当工作流从线指定的开始节点送到目标节点（或从目标点返回开始点）时，会触发配置在线上的事件监听类。

线事件监听类的父类为：com.rh.core.wfe.util.AbstractLineEvent，它有两个方法，一个forward负责监听从开始点送目标节点，另外一个backward负责监听从目标单返回开始点。

|  |
| --- |
| /\*\*  \* 顺序送下一个节点时触发此事件  \* **@param** preWfAct 前一个节点的实例  \* **@param** nextWfAct 下一个节点的实例  \* **@param** lineDef 线定义Bean  \*/  **public** **abstract** **void** forward(WfAct preWfAct, WfAct nextWfAct, Bean lineDef);  /\*\*  \* 返回操作送下一个节点时触发此事件  \* **@param** preWfAct 前一个节点的实例  \* **@param** nextWfAct 下一个节点的实例  \* **@param** lineDef 线定义Bean  \*/  **public** **abstract** **void** backward(WfAct preWfAct, WfAct nextWfAct, Bean lineDef); |

**参考类**

#根据流经的线设置审批单的状态

com.rh.oa.mt.wf.MeetingNoticeLineEvent



**配置方式**

在线属性页🡪事件选项卡的流转事件上，配置AbstractLineEvent的实现类。

