新致工作流

开发指南

**更新记录**

|  |  |  |  |
| --- | --- | --- | --- |
| **版本号** | **更新日期** | **说明** | **更新人** |
| v0.1 | 2017-06-20 | 草稿 | 余磊 |
| V0.2 | 2017-07-26 | 增加集成相关接口 | 余磊 |
| V0.3 | 2017-08-04 | 增加表单开发，修改表单数据实体接口 | 余磊 |
| v0.4 | 2017-08-08 | 整理了表单接口规范 | 余磊 |
| v0.5 | 2017-08-14 | 更新3.1.4表单接口清单，增加“表单审批框架取数接口” | 余磊 |
| v0.6 | 2017-08-23 | 增加了工作流审批接口规范 | 余磊 |
| v0.7 | 2017-10-13 | 4.2.1报账单审批页面增加isnew参数，允许指向新页面（基于jQuery） | 余磊 |
| v0.8 | 2017-10-16 | “3.1.6表单接口规范”增加“设置功能按钮可见性接口” | 余磊 |
| v0.9 | 2017-10-16 | “3.1.6表单接口规范”增加“提交后是否立刻打印接口”；调整了“设置功能按钮可见性接口”，改为“设置功能按钮属性接口”，允许设置按钮标题。 | 余磊 |
| v0.10 | 2017-11-07 | 增加“3.5会签审批” | 余磊 |
| v0.11 | 2017-12-25 | 增加“3.1.6.24. 获取操作类型接口” | 余磊 |
| v0.12 | 2017-12-27 | “4.2.2工作流审批接口”增加“获取审批人接口”、“发起环节获取审批人接口” | 余磊 |
|  |  |  |  |

**目录**

[1. 概述 4](#_Toc502156648)

[1.1. 业务表单与流程概念 4](#_Toc502156649)

[1.1.1. 业务数据 5](#_Toc502156650)

[1.1.2. 表单展现 7](#_Toc502156651)

[1.1.3. 审批流程 7](#_Toc502156652)

[1.1.4. 流程审批功能 7](#_Toc502156653)

[1.2. 业务模型 8](#_Toc502156654)

[1.2.1. 工作流引擎 8](#_Toc502156655)

[1.2.2. 系统管理 9](#_Toc502156656)

[1.2.3. 预算控制 9](#_Toc502156657)

[2. 需求分析 9](#_Toc502156658)

[2.1. 业务表单 9](#_Toc502156659)

[2.2. 业务流程 9](#_Toc502156660)

[3. 开发 10](#_Toc502156661)

[3.1. 业务表单开发 10](#_Toc502156662)

[3.1.1. 表单定义 10](#_Toc502156663)

[3.1.2. 表单界面 11](#_Toc502156664)

[3.1.3. 表单业务数据 14](#_Toc502156665)

[3.1.4. 表单接口清单 22](#_Toc502156666)

[3.1.5. 表单接口调用 23](#_Toc502156667)

[3.1.6. 表单接口规范 25](#_Toc502156668)

[3.2. 工作流概念 40](#_Toc502156669)

[3.2.1. 流程实例状态 40](#_Toc502156670)

[3.2.2. 活动实例状态 40](#_Toc502156671)

[3.2.3. 工作项状态 40](#_Toc502156672)

[3.2.4. 业务实体表数据 41](#_Toc502156673)

[3.3. 流程配置 41](#_Toc502156674)

[3.4. 个人工作台功能 41](#_Toc502156675)

[3.5. 会签审批 42](#_Toc502156676)

[3.5.1. 流程设置 42](#_Toc502156677)

[3.5.2. 提交调用改造 43](#_Toc502156678)

[4. 集成 43](#_Toc502156679)

[4.1. 集成方案1 43](#_Toc502156680)

[4.1.1. 审批待办页面 44](#_Toc502156681)

[4.2. 接口清单 44](#_Toc502156682)

[4.2.1. 表单审批页面接口 44](#_Toc502156683)

[4.2.2. 工作流审批接口 47](#_Toc502156684)

[4.2.3. 流程图接口 65](#_Toc502156685)

[4.2.4. 审批历史接口 65](#_Toc502156686)

# 概述

## 业务表单与流程概念

工作流中的业务表单，由一个包含业务数据的表单，以及与表单对应一个或一组流程共同组成。业务表单的概念如下：



### 业务数据

业务表单应当含有必要的业务数据，否则该表单和流程没有业务意义。

业务表单需要保存在数据库中。一个表单的业务实体数据，一般由一个业务主实体和一个或多个业务子实体组成。一个流程实例包含一份业务实体数据。



工作流引擎提供了业务数据保存的接口。

### 表单展现

业务表单需要把业务数据用合适的方式进行展现。

同时需要考虑在审批流程的不同环节，业务数据的展现方式可能不同。例如：

* 在发起环节，允许填写内容；但在审批环节，填写的内容只允许审批人员查看，但不允许随意修改。如果审批人员认为填写有错误，一般将退回到发起人更正。
* 有部分内容在部分审批环节不需要显示，比如业务部门审批人员不关心预算执行情况，则在这些环节不显示预算执行相关信息，仅到达预算审批员岗位才显示。

这部分都需要根据实际需求确定。

另外表单一般需要考虑打印，因此工作流支持一个公用的打印/导出功能，通过在线pdf实现。

### 审批流程

业务表单填写完成后，需要发起流程，并按照预先设置的流程进行流转。流程定义需要借助工作流的流程设计工具，按照所见即所得的方式进行定义。

另外，为了支撑工作流定义，需要按照设置参与者、流程条件、后台业务处理、流程变量等内容，这部分设置工作都可以使用流程设计工具解决。

* 参与者：每个审批环节的审批人
* 流程条件：流程的流程线和分支条件
* 流程业务处理：流程流转过程中的业务数据处理，例如预算控制、合同执行数计算、生成凭证等。
* 流程变量：在以上流程定义中可能需要流程变量参与。

### 流程审批功能

对流程审批流转的操作功能。其含义如下所述。

|  |  |  |
| --- | --- | --- |
| **序号** | **操作** | **说明** |
| 1 | 发起新流程 | 发起人发起一个新的流程实例，发起之前需要填写业务表单数据，并指定使用哪一个流程，一般也需要指定下一审批人。 |
| 2 | 提交 | 审批人打开待审批的报账单，填写审批意见后提交到下一步审批环节。 |
| 3 | 保存草稿 | 发起人填写业务表单数据后，暂时不发起新流程，保存为草稿。以后可以打开草稿填写完整数据并发起流程。 |
| 4 | 撤回 | 发起人发起了新的审批流程后，在没有审批人审批之前，自行撤回流程 |
| 5 | 回退 | 审批人审核时不同意，退回到上一步或者之前的任何一个审批环节。 |
| 6 | 拒绝 | 审批人审核时不同意，直接退回到发起人环节。 |
| 7 | 转办 | 审批人把待审批流程转给其他人，由其他人审批同意后自动提交到下一步。 |
| 8 | 转拟办 | 审批人把待审批流程转给其他人收集意见，其他人审批同意后自动转回原审批人。 |
| 9 | 删除 | 流程被退回到发起人后，由发起人自行删除流程实例。此时流程实例做逻辑删除。 |

## 业务模型

### 工作流引擎



### 系统管理

待细化

### 预算控制

待细化

# 需求分析

为了开发业务表单，需要分析以下内容：

## 业务表单

* 有哪些业务表单？
* 表单上需要展现什么信息字段？怎样布局？
* 表单上字段是什么数据类型？用什么方式输入，比如文本、下拉框还是选择框？
* 在不同审批环节中，需要展现哪一些信息字段？

## 业务流程

* 每个业务表单需要配置哪些审批流程？
* 有哪一些审批环节、每个环节由谁审批？以此形成【参与者】需求。
* 业务流程中有哪些分支条件？满足什么条件会走分支条件？以此形成【流程线与条件】需求。
* 流程流转中、或审批完成后，与外部系统有什么接口交互？流程流转中、或审批完成后，有什么内部业务数据计算？以此形成【流程业务处理】需求。

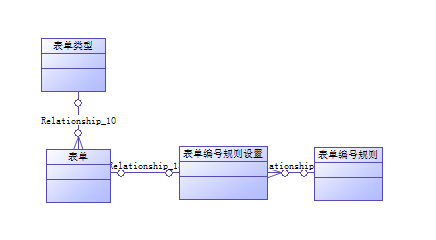
# 开发

## 业务表单开发

### 表单定义

工作流引擎中表单定义涉及到以下几张表：

|  |  |
| --- | --- |
| **表名** | **说明** |
| TWF\_FORMTYPE | 表单类型 |
| TWF\_FORM | 表单信息 |



目前表单定义需要通过后台增加记录。增加表单后，即可在流程设计器中使用。

表单定义中最重要的字段是twf\_form.pageurl。该字段为一个url地址，用来在系统中打开业务表单。url上也可以带参数。

例如：

SMCSSSC/ssc\_smcs\_CostFormAction!toUrl.action?formTypeCode=BT0101&busiClassCode=BC0101

#### 表单编码规则

|  |  |
| --- | --- |
| **表名** | **说明** |
| TPCM\_SERIALNUMRULE | 表单编号规则 |
| TPCM\_FORMSERIALNUMRULE | 表单编号规则应用设置 |

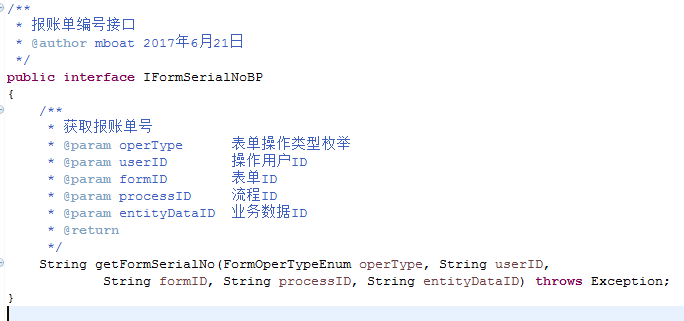
表单定义时，一般也需要定义表单编码规则。工作流引擎在新建表单时，会创建一个表单编码，作为业务表单数据的一部分。

工作流引擎有一个默认的表单编码规则，需要配置在tpcm\_serialnumrule表中，规则为前缀 + “-” + 年月 + “-” 若干位流水号。

工作流引擎也支持自定义表单编码规则，需要实现接口

com.newtouch.workflow.app.ifx.IformSerialNoBP

并把实现类作为javaBean配置在spring中。一般配置在resources/workflow/spring-context-workflow.xml文件中。



### 表单界面

表单界面需要根据业务需求自行开发。表单界面入口一般为一个web页面，例如jsp。然后配置到twf\_form.pageurl字段。

如下图所示，下图为表单审批页面，其中红框为业务表单，红框之外是表单审批框架页面。提供了公用的审批操作。实际系统中，需要将具体的业务表单嵌入在表单审批框架中（通过iframe方式）。



#### 界面权限控制

可以根据界面审批环节实现界面权限控制

#### 表单接口

实际使用过程中，业务表单页面与外部的表单审批框架页面之间需要通过接口交互数据（js形式）。

有部分接口由业务表单提供，提供给表单审批框架页面调用。表单审批框架页面也会提供接口，给业务表单或外部的主页面调用。

以下接口都是js形式的函数



#### 打印/导出

报账单打印功能，默认使用jasperreports组件导出为PDF，然后再打印。具体实现步骤如下：

* 业务表单的BP层JavaBean需要实现接口

com.newtouch.workflow.app.ifx.IFormPrintDataAdapter

该接口方法定义



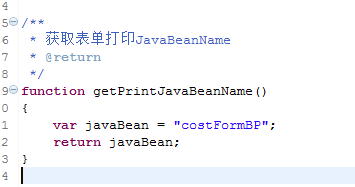
该接口负责为业务表单打印提供数据，需要返回的数据包括：报表参数、报表数据集、jasper文件名、下载文件名

该接口的实现类由Spring托管（通过注解自动扫描或xml配置）。

* 业务表单Web界面提供一个方法getPrintJavaBeanName()，返回上述javaBean的名称（应与spring初始化的beanid一致）。

注意，如果使用注解自动扫描的JavaBean，beanid为类名且第一个单词首字母小写。

例如FormCostBP，beanid = formCostBP



* 以前的getPrintData()方法作废

### 表单业务数据

表单上的业务数据需要保存在数据库表中，这些表被称为业务实体表。业务实体表根据业务需求自定定义。

#### 业务数据保存方式

工作流引擎支持以下几种方式保存业务实体数据：

* 业务实体数据持久化，完全委托给工作流引擎

通过JavaEntity实体表中的注解，描述所有的业务数据表的物理结构，并配置在工作流引擎WfEntityXmlConfigFactory中。

在发起新流程、表单提交、保存草稿时，通过表单数据接口把json格式的表单业务实体数据传入工作流引擎，由工作流引擎执行业务数据表的保存或更新操作。

* 工作流引擎仅保存业务实体数据与流程实例的关联关系表

业务表单自行完成业务数据表的持久化操作。

通过JavaEntity实体表中的注解，描述业务数据表与工作流实例关联表的物理结构，，并配置在工作流引擎WfEntityXmlConfigFactory中。

在发起新流程、表单提交时，通过表单数据接口把json格式的关联表数据传入工作流引擎，由工作流引擎执行关联表的保存或更新操作。

* 工作流引擎调用自定义的Spring Java Bean来保存实体数据

业务表单实现自己的Java Bean来完成业务数据持久化。

在发起新流程、表单提交、或保存草稿时，通过表单数据接口把json格式的表单业务实体数据，以及指定的Spring Java Bean名称传入工作流引擎。

由工作流引擎负责调用Spring Java Bean来实现业务实体数据的保存或更新。

表单打开时加载业务数据的过程，需要自行实现。

#### 定义实体表

通过java POJO类来定义实体。

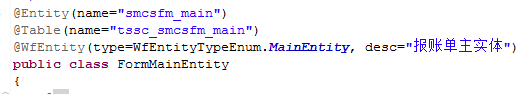


##### 实体表定义

* 使用javax.persistence.Entity注解，定义为实体类；
* 使用javax.persistence.Table注解，定义实体类的物理表名
* 使用com.newtouch.workflow.entitydata.annotation.WfEntity注解，定义该实体类的类型（主实体或子实体），以及实体中文名（可为空）

例如以下定义，定义该实体为流程引擎支持的业务实体类，实体代码为“smcsfm\_main”，

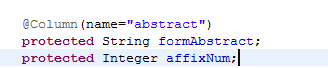
实体对应的物理表为“tssc\_smcsfm\_main”，实体类型为主实体（通过type=WfEntityTypeEnum.MainEntity），实体中文名“报账单主实体”。



##### 属性与字段

* 实体的属性自动识别为数据库表的字段，属性名称直接作为数据库字段物理名称。如果字段物理名与实体属性名不一致，需要使用@javax.persistence.Column注解。
* 静态属性忽略，不识别为数据库表字段。
* 被注解为@java.beans.Transient的属性忽略，不识别为数据库表字段。

如以下定义，实体属性formAbstract和affixNum，对应的字段物理名分别为abstract和affixnum。



实体类属性/字段类型，目前仅支持String、Integer、Double、Float这几种简单类型。

如果是java.util.Date，原始数据仅支持yyyy-MM-dd格式的日期字符串。

##### 实体特殊字段定义

使用com.newtouch.workflow.entitydata.annotation.WfColumn注解，在实体属性上定义特定的列类型。工作流将在表单提交时作处理。目前支持的列类型，包括以下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **类型** | **实体字段** | **作用** | **主实体** | **子实体** |
| 1 | WfColumnTypeEnum.EntityDataID | 数据实体ID | 在主实体中作为主键，在子实体中作为外键 | 必填 | 必填 |
| 2 | WfColumnTypeEnum.ProcessInstID | 流程实例ID | 保存时将写入流程实例ID，作为流程实例数据与业务实体数据的关联主键 | 必填 |  |
| 3 | WfColumnTypeEnum.ProcessID | 流程ID | 保存时将写入流程ID | 可选 |  |
| 4 | WfColumnTypeEnum.FormID | 表单ID | 保存时将写入表单ID | 可选 |  |
| 5 | WfColumnTypeEnum.DetailID | 子实体明细ID |  |  | 必填 |
| 6 | WfColumnTypeEnum.Status | 状态 |  | 可选 | 可选 |

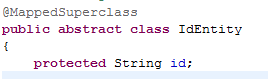
##### 实体继承

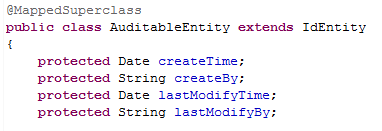
实体支持继承。父类实体需要使用@javax.persistence.Entity或@javax.persistence.MappedSuperclass注解。子类仍然按照上述方式定义。

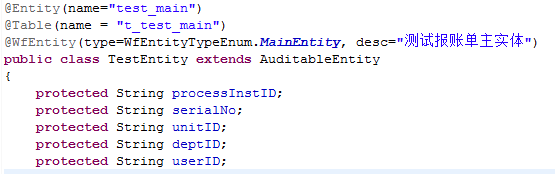
子实体会将父类的所有已声明属性一起列出。

如以下设置，TestEntity继承于AuditableEntity，而AuditableEntity继承于IdEntity。最终TestEntity识别出来的属性/字段包括：

|  |  |  |
| --- | --- | --- |
| **序号** | **字段/属性** | **定义实体** |
| 1 | id | IdEntity |
| 2 | createTime | AuditableEntity |
| 3 | createBy | AuditableEntity |
| 4 | lastModifyTime | AuditableEntity |
| 5 | lastModifyBy | AuditableEntity |
| 6 | processInstID | TestEntity |
| 7 | serialNo | TestEntity |
|  | …… | …… |

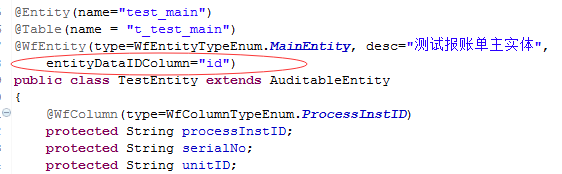






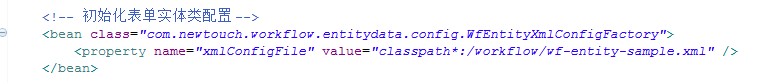
在定义特殊字段时，如果字段定义在父类，则无法进行注解，需要改为在@WfEntity中进行定义。

如下图，entityDataIDColumn对应的字段在父类中，只能定义在WfEntity标签中。



#### 表单与实体表xml配置

在配置文件中resources/webflow/spring-context-workflow.xml中，初始化表单实体类配置



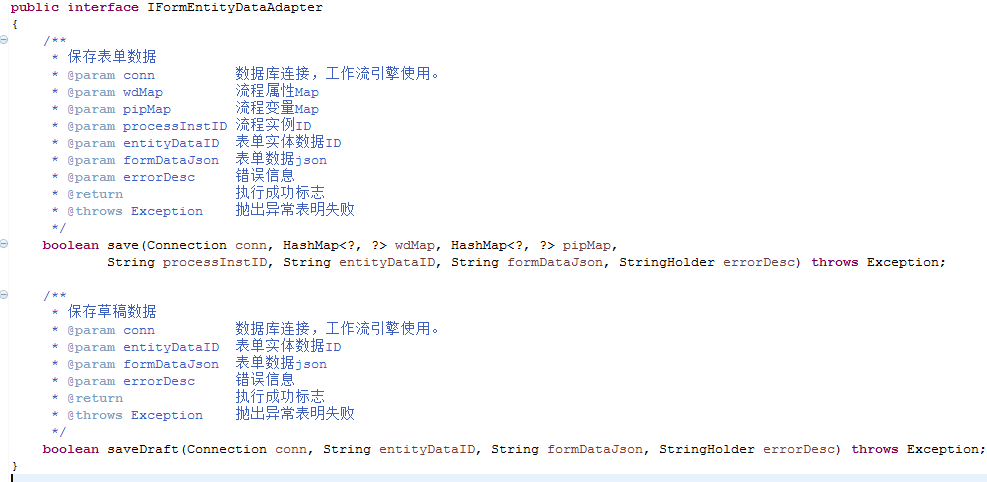
其中引用了wf-entity-sample.xml的配置文件，描述了系统中有哪一些流程业务实体类，以及每一种表单使用了哪一些流程业务实体类。





#### 业务数据保存JavaBean定义

实现com.newtouch.workflow.entitydata. IformEntityDataAdapter接口



#### 表单数据接口使用方式

按照三种业务数据保存方式，在调用表单数据接口时，有不同的用法。

##### 全部实体数据保存

var formDataObject =

{

actiontype : "entitydata", //数据保存方式。entitydata: 实体数据表配置解析； spring: 调用指定的Spring Java Bean

formcode : "cost", //表单编码，与wf-entity.xml配置文件中一致

entitydataid : entityDataID, //表达实体数据ID，可以不传入。如果为空，则工作流引擎自行生成ID

entitys : //实体数据

[

{

name : "smcsfm\_main", //主实体

data : [ createMainEntityDataObject(MainEntityControls) ]

},

{

name : "smcsfm\_cost", //子实体1

data : [ createExpandMainEntityDataObject(MainChildControls) ]

},

{

name : "smcsfm\_cost\_info", //子实体2

data : BusinessPanel.getCostInfoPanel().getFormData()

}

]

};

##### 关联表保存

var formDataObject =

{

actiontype : "entitydata", //数据保存方式。entitydata: 实体数据表配置解析； spring: 调用指定的Spring Java Bean

formcode : "cost2", //表单编码，与wf-entity.xml配置文件中一致

entitydataid : entityDataID, //表达实体数据ID，可以不传入。如果为空，则工作流引擎自行生成ID

entitys : //业务实体数据

[

{

name : "demo\_formref", //实体编码，与实体类中配置一致

data : [ {} ]

}

]

};

##### SpringJavaBean保存

var formDataObject =

{

actiontype : "spring", //数据保存方式方式。entitydata: 实体数据表配置解析； spring: 调用指定的Spring Java Bean

javabean : "cost3FormBP", //Spring Java Bean ID

entitydataid : entityDataID, //表达实体数据ID，可以不传入。如果为空，则工作流引擎自行生成ID

entitys : //业务实体数据，格式随意

{

entityDataID : entityDataID,

memo : "memo"

}

};

### 表单接口清单

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **接口** | **接口提供方** | **接口调用方** | **函数名** | **备注** |
| **业务表单提供接口** | | | | | |
| 1 | 表单加载状态接口 | 业务表单 | 表单审批框架 | checkFormDataLoaded |  |
| 2 | 表单数据接口 | 业务表单 | 表单审批框架 | getFormData |  |
| 3 | 表单草稿数据接口 | 业务表单 | 表单审批框架 | getDraftFormData |  |
| 4 | 表单打印JavaBean接口 | 业务表单 | 表单审批框架 | getPrintJavaBeanName |  |
| 5 | 表单提交接口 | 业务表单 | 表单审批框架 | OnSubmit |  |
| 6 | 表单提交完成接口 | 业务表单 | 表单审批框架 | AfterSubmit |  |
| 7 | 表单保存草稿接口 | 业务表单 | 表单审批框架 | OnSave |  |
| 8 | 表单保存草稿完成接口 | 业务表单 | 表单审批框架 | AfterSave |  |
| 9 | 表单回退接口 | 业务表单 | 表单审批框架 | OnRollback |  |
| 10 | 表单回退完成接口 | 业务表单 | 表单审批框架 | AfterRollback |  |
| 11 | 表单拒绝接口 | 业务表单 | 表单审批框架 | OnReject |  |
| 12 | 表单拒绝完成接口 | 业务表单 | 表单审批框架 | AfterReject |  |
| 13 | 表单转办接口 | 业务表单 | 表单审批框架 | OnTransfer |  |
| 14 | 表单转办完成接口 | 业务表单 | 表单审批框架 | AfterTransfer |  |
| 15 | 表单转拟办接口 | 业务表单 | 表单审批框架 | OnTransmit |  |
| 16 | 表单转拟办完成接口 | 业务表单 | 表单审批框架 | AfterTransmit |  |
| 17 | 表单获取提交后是否立刻打印标志接口 | 业务表单 | 表单审批框架 | isPrintAfterSubmit |  |
| **表单审批框架提供接口** | | | | | |
| 18 | 获取流程变量接口 | 表单审批框架 | 业务表单 | getProcData |  |
| 19 | 获取流程属性接口 | 表单审批框架 | 业务表单 | getWorkData |  |
| 20 | 获取业务实体数据ID接口 | 表单审批框架 | 业务表单 | getEntityDataID |  |
| 21 | 获取表单编号接口 | 表单审批框架 | 业务表单 | getFormSerialNo |  |
| 22 | 获取活动属性接口 | 表单审批框架 | 业务表单 | getActivityProperty |  |
| 23 | 获取环节标志接口 | 表单审批框架 | 业务表单 | getActivityBusiType |  |
| 24 | 获取操作类型接口 | 表单审批框架 | 业务表单 | getOperType |  |
| 25 | 设置功能按钮属性接口 | 表单审批框架 | 业务表单 | setButtonOption |  |
| **主界面提供接口** | | | | | |
| 26 | 回调主界面接口 | 主界面dke | 表单审批框架 | callbackAfterOperation |  |

### 表单接口调用

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **操作** | **接口** | **主界面** | **表单审批框架页面** | **业务表单** |
| 1 | 发起新流程 | 表单提交接口 |  | ＝＝＝＞ | |
| 2 | 表单数据接口 |  | ＝＝＝＞ | |
| 3 | 表单提交完成接口 |  | ＝＝＝＞ | |
| 4 | 回调刷新主界面接口 | ＜＝＝＝ | |  |
| 5 | 提交 | 表单加载状态接口 |  | **＝＝＝＞** | |
| 6 | 表单提交接口 |  | **＝＝＝＞** | |
| 7 | 表单数据接口 |  | **＝＝＝＞** | |
| 8 | 表单提交完成接口 |  | **＝＝＝＞** | |
| 9 | 回调主界面接口 | **＜＝＝＝** | |  |
| 10 | 保存草稿 | 表单保存草稿接口 |  | **＝＝＝＞** | |
| 11 | 表单草稿数据接口 |  | **＝＝＝＞** | |
| 12 | 表单保存草稿完成接口 |  | **＝＝＝＞** | |
| 13 | 回调主界面接口 | **＜＝＝＝** | |  |
| 14 | 打印 | 表单打印JavaBean接口 |  | **＝＝＝＞** | |
| 15 | 撤回 | 表单加载状态接口 |  | **＝＝＝＞** | |
| 16 | 表单回退接口 |  | **＝＝＝＞** | |
| 17 | 表单回退完成接口 |  | **＝＝＝＞** | |
| 18 | 回调主界面接口 | **＜＝＝＝** | |  |
| 19 | 回退 | 表单加载状态接口 |  | **＝＝＝＞** | |
| 20 | 表单回退接口 |  | **＝＝＝＞** | |
| 21 | 表单数据接口 |  | **＝＝＝＞** | |
| 22 | 表单回退完成接口 |  | **＝＝＝＞** | |
| 23 | 回调主界面接口 | **＜＝＝＝** | |  |
| 24 | 拒绝 | 表单加载状态接口 |  | **＝＝＝＞** | |
| 25 | 表单拒绝接口 |  | **＝＝＝＞** | |
| 26 | 表单数据接口 |  | **＝＝＝＞** | |
| 27 | 表单拒绝完成接口 |  | **＝＝＝＞** | |
| 28 | 回调主界面接口 | **＜＝＝＝** | |  |
| 29 | 转办 | 表单加载状态接口 |  | **＝＝＝＞** | |
| 30 | 表单转办接口 |  | **＝＝＝＞** | |
| 31 | 表单数据接口 |  | **＝＝＝＞** | |
| 32 | 表单转办完成接口 |  | **＝＝＝＞** | |
| 33 | 回调主界面接口 | **＜＝＝＝** | |  |
| 34 | 转拟办 | 表单加载状态接口 |  | **＝＝＝＞** | |
| 35 | 表单转拟办接口 |  | **＝＝＝＞** | |
| 36 | 表单数据接口 |  | **＝＝＝＞** | |
| 37 | 表单转拟办完成接口 |  | **＝＝＝＞** | |
| 38 | 回调主界面接口 | **＜＝＝＝** | |  |

### 表单接口规范

业务表单需要具备以下接口，让表单审批框架页面能够操作业务表单，进行验证或加载。

以下接口都是js形式的函数

#### 表单加载状态接口

**函数声明：**

function boolean checkFormDataLoaded ()

**说明：**

进行流程审批操作时，将通过该接口检查业务表单数据加载是否完成。如未完成，后续操作将无法继续，避免出现异步加载失败导致表单数据不一致的情况。

涉及到的操作包括：

|  |  |
| --- | --- |
| **序号** | **操作** |
| 1 | 提交 |
| 2 | 撤回 |
| 3 | 回退 |
| 4 | 拒绝 |
| 5 | 转办 |
| 6 | 转拟办 |

**参数：**

无

**返回值：**

boolean类型。业务表单是否加载完成。true，加载完成；false加载未完成，后续操作将无法进行。

#### 表单数据接口

**函数声明：**

function object getFormData ()

**说明：**

表单提交、回退、拒绝等流程审批操作时，将通过该接口获取业务表单数据，用于提交工作流流程接口。最终由工作流引擎保存到业务实体表。

|  |  |
| --- | --- |
| **序号** | **操作** |
| 1 | 发起新流程 |
| 2 | 提交 |
| 3 | 回退 |
| 4 | 拒绝 |
| 5 | 转办 |
| 6 | 转拟办 |

**参数：**

无

**返回值：**

string类型。表单数据，json格式

完整格式

{

actiontype : "entitydata", //数据保存方式方式。entitydata: 实体数据表配置解析； spring: 调用指定的Spring Java Bean

javabean : "costFormBP", //Spring Java Bean ID

formcode : "cost", //表单代码，同配置文件wf-entity.xml中formcode

entitydataid : "xxxx", //表达实体数据ID，可以不传入。如果为空，则工作流引擎自行生成ID

entitys : //业务实体数据

[

//第一个实体表。如果是主实体表，只有一行记录

{

name : entityname1, //实体1名称

data : //实体值

[

{

field1 : value1,

field2 : value2,

field3 : value3

}

]

},

//第二个实体表

{

name : entityname2, //实体2名称

data : //实体值

[

{

field1 : value1,

field2 : value2,

field3 : value3

},

{

field1 : value1,

field2 : value2,

field3 : value3

},

....

]

},

...

]

}

#### 草稿表单数据接口

**函数声明：**

function object getDraftFormData ()

**说明：**

表单保存草稿时，将通过该接口获取业务表单数据，由工作流引擎保存到业务实体表。

**参数：**

无

**返回值：**

string类型。表单数据，json格式。格式同表单数据获取接口。

#### 表单打印JavaBean接口

**函数声明：**

function string getPrintJavaBeanName ()

**说明：**

表单打印时，将通过该接口打印javaBeanName。

**参数：**

无

**返回值：**

string类型。打印java bean name名称。表单打印的JavaBean应当交由Spring托管，javabean名称为spring中bean id。

如java bean采用注解方式，注意使用spring beanid的默认名称。例如注解的java b ean的类名为CostFormBP，则默认beanid = comFormBP。

#### 表单提交接口

**函数声明：**

function boolean OnSubmit()

**说明：**

表单发起新流程或提交时之前，将调用该接口，做业务表单提交前的处理。包括但不限于：

1. 业务数据计算
2. 设置自定义的流程属性（如@DeptID），以提流程条件、流程接口或提交调用使用。
3. 判断是否满足提交条件，例如数据填写是否完整，数据填写是否正确等。

如果返回false，则表单不会提交。

**参数：**

无

**返回值：**

boolean类型。true，验证通过继续提交；false验证不通过，将不提交表单。

#### 表单提交完成接口

**函数声明：**

function void AfterSubmit(processInstID, entityDataID)

**说明：**

表单发起新流程或提交以后，调用该接口。

**参数：**

|  |  |  |
| --- | --- | --- |
| **序号** | **参数** | **说明** |
| 1 | processInstID | 流程实例ID，引用于(twf\_processinst.processinstid) |
| 2 | entityDataID | 实体数据ID |

**返回值：**

无

#### 表单保存草稿

**函数声明：**

function boolean OnSave()

**说明：**

表单保存草稿之前，将调用该接口，做业务表单保存之前的。

如果返回false，则表单不会提交。

**参数：**

无

**返回值：**

boolean类型。true，验证通过继续保存；false验证不通过，将不保存草稿。

#### 表单保存草稿完成接口

**函数声明：**

function void AfterSave (entityDataID)

**说明：**

表单保存草稿以后，调用该接口。

**参数：**

|  |  |  |
| --- | --- | --- |
| **序号** | **参数** | **说明** |
| 1 | entityDataID | 实体数据ID |

**返回值：**

无

#### 表单回退接口

**函数声明：**

function boolean OnRollback()

**说明：**

表单回退之前，将调用该接口，做业务表单回退前的处理。

如果返回false，则表单不会回退。

**参数：**

无

**返回值：**

boolean类型。true，验证通过继续回退；false验证不通过，将不回退表单。

#### 表单回退完成接口

**函数声明：**

function void AfterRollback ()

**说明：**

表单回退完成以后，调用该接口。

**参数：**

无

**返回值：**

无

#### 表单拒绝接口

**函数声明：**

function boolean OnReject()

**说明：**

表单拒绝之前，将调用该接口，做业务表单拒绝前的处理。

如果返回false，则表单不会拒绝。

**参数：**

无

**返回值：**

boolean类型。true，验证通过继续拒绝；false验证不通过，将不拒绝表单。

#### 表单拒绝完成接口

**函数声明：**

function void AfterReject ()

**说明：**

表单拒绝完成以后，调用该接口。

**参数：**

无

**返回值：**

无

#### 表单转办接口

**函数声明：**

function boolean OnTransfer ()

**说明：**

表单转办之前，将调用该接口，做业务表单转办前的处理。

如果返回false，则表单不会转办。

**参数：**

无

**返回值：**

boolean类型。true，验证通过继续转办；false验证不通过，将不转办表单。

#### 表单转办完成接口

**函数声明：**

function void AfterTransfer ()

**说明：**

表单转办完成以后，调用该接口。

**参数：**

无

**返回值：**

无

#### 表单转拟办接口

**函数声明：**

function boolean OnTransmit ()

**说明：**

表单转拟办之前，将调用该接口，做业务表单转拟办前的处理。包括但不限于：

如果返回false，则表单不会转拟办。

**参数：**

无

**返回值：**

boolean类型。true，验证通过继续转拟办；false验证不通过，将不转拟办表单。

#### 表单转拟办完成接口

**函数声明：**

function void AfterTransmit ()

**说明：**

表单转拟办完成以后，调用该接口。

**参数：**

无

**返回值：**

无

#### 表单获取提交后是否立刻打印标志接口

**函数声明：**

function boolean isPrintAfterSubmit()

**说明：**

业务表单提供，由表单审批页面在申请完成或者提交完成后调用。获取是否要打印的标志位。如果返回true，则立刻打印。

**参数：**

无

**返回值：**

返回boolean。如果返回true，在申请完成或审批完成后，立刻打印；返回false则不打印。

#### 获取流程变量接口

**函数声明：**

function object getProcData ()

**说明：**

业务表单调用，获取系统内置的流程变量Map。

在业务表单中，使用以下方式调用：

parent.getProcData();

获取流程变量：

parent.getProcData().getValue(“ProcessInstID”);

**参数：**

无

**返回值：**

返回流程变量Map。流程变量清单，参见流程工具使用指南相关章节。

#### 获取流程属性接口

**函数声明：**

function object getWorkData ()

**说明：**

业务表单调用，获取自定义的流程属性Map。

在业务表单中，使用以下方式调用：

parent. getWorkData ();

获取流程属性：

parent.getProcData().getValue(“G\_SSC\_DETPID”);

设置流程属性：

parent.getProcData().setValue(“G\_SSC\_DETPID”, “xxxxxx”);

**参数：**

无

**返回值：**

返回流程属性Map。

#### 获取业务实体数据ID接口

**函数声明：**

function string getEntityDataID()

**说明：**

业务表单调用，获取业务实体数据ID。

在业务表单中，使用以下方式调用：

parent. getEntityDataID();

**参数：**

无

**返回值：**

返回业务实体数据ID。

#### 获取表单编号接口

**函数声明：**

function string getFormSerialNo()

**说明：**

业务表单调用，获取表单编号。

在业务表单中，使用以下方式调用：

parent. getFormSerialNo();

**参数：**

无

**返回值：**

返回表单编号。

#### 获取活动属性接口

**函数声明：**

function string getActivityProperty()

**说明：**

业务表单调用，获取活动属性。

在业务表单中，使用以下方式调用：

parent. getActivityProperty();

**参数：**

无

**返回值：**

返回表单编号。

#### 获取环节标志接口

**函数声明：**

function string getActivityBusiType()

**说明：**

业务表单调用，获取活动（即审批环节）上设置的属性。

在业务表单中，使用以下方式调用：

parent. getActivityBusiType();

**参数：**

无

**返回值：**

返回表单编号。

#### 获取操作类型接口

**函数声明：**

function string getOperType()

**说明：**

业务表单调用，获取当前操作业务类型。

在业务表单中，使用以下方式调用：

parent. getOperType ();

**参数：**

无

**返回值：**

返回操作类型。

1, 新建

2, 审批

3, 个人跟踪

4, 我已审批

5, 审批完成

6, 草稿

7, 模板

#### 设置功能按钮属性接口

**函数声明：**

function void setButtonOption (option)

**说明：**

业务表单调用，设置表单审批框架上的功能按钮可见性。

在业务表单中，使用以下方式调用：

var option =

{

transfer : /\* 转办 \*/

{

visible : false

},

transmit : /\* 转拟办 \*/

{

visible : false

},

print : /\* 流程图 \*/

{

caption : "打印1"

}

}

parent.setButtonOption(option);

**参数：**

option ： 功能显示参数，格式为json对象

var option =

{

apply : //按钮标志

{

caption : “申请”, //按钮标题

visible : true //按钮可见性

},

savedraft : //下一个功能按钮属性

{

visible : false //按钮可见性

}

}

不同的表单审批界面，允许不同的按钮。具体功能按钮及其标志对应如下：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **序号** | **功能** | **按钮标志** | **审批界面** | | | | |
| **新建** | **待办审批** | **个人跟踪** | **审批完成** | **草稿表单** |
| 1 | 申请 | apply | **〇** | **〇** |  |  | **〇** |
| 2 | 保存草稿 | savedraft | **〇** |  |  |  | **〇** |
| 3 | 同意 | submit |  | **〇** |  |  |  |
| 4 | 退回 | rollback |  | **〇** |  |  |  |
| 5 | 转办 | transfer |  | **〇** |  |  |  |
| 6 | 转拟办 | transmit |  | **〇** |  |  |  |
| 7 | 拒绝/不同意 | reject |  | **〇** |  |  |  |
| 8 | 撤回 | undo |  |  | **〇** |  |  |
| 9 | 打印 | print |  | **〇** | **〇** | **〇** | **〇** |
| 10 | 流程图 | procchart | **〇** | **〇** | **〇** | **〇** | **〇** |
| 11 | 关闭 | close | **〇** | **〇** | **〇** | **〇** | **〇** |
| 12 | 习惯用语 | opinion | **〇** | **〇** | **〇** | **〇** | **〇** |
| 13 | 附件上传 | attach\_upload | **〇** | **〇** | **〇** | **〇** | **〇** |
| 14 | 附件下载 | attach\_download | **〇** | **〇** | **〇** | **〇** | **〇** |
| 15 | 附件删除 | attach\_delete | **〇** | **〇** | **〇** | **〇** | **〇** |

**返回值：**

返回表单编号。

#### 回调主界面接口

**函数声明：**

function void callbackAfterOperation ()

**说明：**

表单审批、回退等操作完成以后，在关闭表单窗口之前，调用该接口回调主页面。可以执行主页面的数据刷新操作。

**参数：**

无

**返回值：**

无

## 工作流概念

### 流程实例状态

流程实例状态被记录在twf\_processinst.status字段中。

|  |  |  |
| --- | --- | --- |
| **序号** | **状态** | **说明** |
| 1 | 1: 运行 | 表示流程正在运行中。当流程实例被发起，但还未审批完成时后，状态即为【运行】。 |
| 2 | 4: 完成 | 表示流程已审批完成（已流转到终点），不能继续流转。当流程实例审批后，状态变为【完成】。 |
| 3 | 5: 终止 | 表示流程非正常结束。当流程实例在中途被退回到发起人，并被删除后，状态变为【终止】。 |

### 活动实例状态

活动实例状态记录在twf\_activityinst.status字段中。

|  |  |  |
| --- | --- | --- |
| **序号** | **状态** | **说明** |
| 1 | 0：初始 |  |
| 2 | 1: 就绪 | 活动实例已创建，但尚未分配工作项 |
| 3 | 2：执行 | 活动实例已创建并分配工作项，等待审批 |
| 4 | 3：挂起 |  |
| 5 | 4：过期 |  |
| 6 | 5：完成 | 活动实体审批通过 |
| 7 | 6：终止 | 活动实例审批不通过或删除 |
| 8 | 7：废止 |  |

### 工作项状态

工作项状态记录在twf\_workitem.status字段中。

|  |  |  |
| --- | --- | --- |
| **序号** | **状态** | **说明** |
| 1 | 0: 初始 | 表示该工作项还未审批。 |
| 2 | 1: 完成 | 表示该工作项已审批完成（不可重复审批）。 |
| 3 | 2: 关联完成 | 表示由于其他人的工作项目审批完成，导致该工作项也同时完成。一般出现在同一活动实例下多个工作项目的情况。 |
|  |  |  |
|  |  |  |
|  |  |  |

### 业务实体表数据

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | **操作** | **流程实例数据** | **业务实体表** |
| 1 | 新建报账单提交 | 创建流程实例数据 | 保存业务实体表数据 |
| 2 | 审批报账单并提交/回退/拒绝/转办/转拟办 | 更新流程实例数据 | 更新业务实体表数据 |
| 3 | 删除报账单 | 更新流程实例状态为【5终止】 | 不更新业务实体表数据。  如果需要物理删除业务表数据，可以在表单的终止调用中执行（因为流程实例数据并未物理删除，不建议如此处理）。 |
| 4 | 新建报账单并保存草稿 | 无 | 保存业务实体表数据，但业务实体主表中流程实例ID为空（因为尚未发起流程实例） |
| 5 | 打开草稿并保存草稿 | 无 | 更新业务实体表数据 |
| 6 | 打开草稿并提交 | 创建流程实例数据 | 更新业务实体表数据，并更新流程实例ID |
| 7 | 删除草稿 | 无 | 删除业务实体表数据（物理删除） |

## 流程配置

参见《新致工作流流程设计工具使用手册.docx》

## 个人工作台功能

个人工作台功能一般包括：

* 新建流程/报账单
* 工作项审批
* 个人跟踪
* 审批完成
* 工作项草稿
* 工作项查询

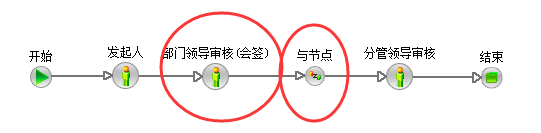
这些功能已经提供默认实现，可以在默认实现的基础上修改，或参考后重新实现。

## 会签审批

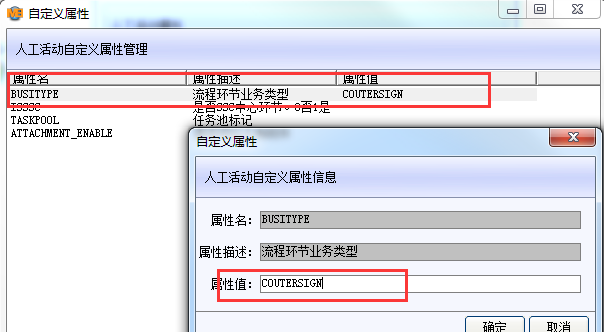
会签审批的概念：一个审批节点有多个审批人，需要等这些审批人全部审批通过，流程才能下到一步。

对于一个审批节点有多个审批人，工作流引擎默认处理模式为，任何一人审批通过，则流程马上流转到下一步，其他人不需要审批。因此为了实现会签的模式，需要对流程进行特定操作处理。

### 流程设置



* 会签节点后需要跟一个与节点。（图中会签节点是【部门领导审核(会签)】，之后跟随了一个与节点。）
* 会签节点的自定义属性BUSITYPE，应当设置为“COUTERSIGN”



* 会签节点后第一个人工审批节点的任务分配方式应当为：自动分配。（图中是指【分管领导审核】节点。）

### 提交调用改造

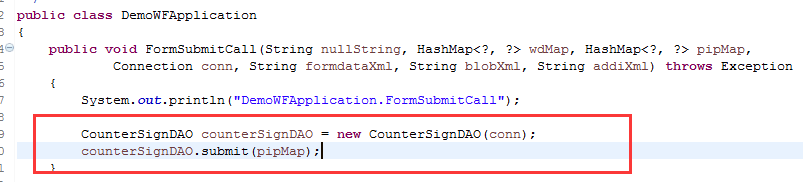
会签节点需要设置提交调用，该提交调用内需要引入class：

com.newtouch.workflow.engine.dao.CounterSignDAO

并调用submit方法。

下面是样例代码：

其中创建CounterSignDAO类时需要传入Connection



# 集成

工作流审批功能，如果前端Web页面采用集成方式集成到其他系统中。

## 集成方案1

<待细化>

### 审批待办页面



## 接口清单

### 表单审批页面接口

允许系统打开表单页面，执行报账单新建、审批、查看等功能。

|  |  |  |
| --- | --- | --- |
| **序号** | **接口** | **url** |
| 1 | 新建表单接口 | [basePath] + wf/formmgr/newProcess |
| 2 | 待办审批表单接口 | [basePath] + wf/formmgr/formApproveForm |
| 3 | 个人跟踪表单接口 | [basePath] + wf/formmgr/viewInTransForm |
| 4 | 审批完成表单接口 | [basePath] + wf/formmgr/viewFinishedForm |
| 5 | 草稿表单接口 | [basePath] + wf/formmgr/openDraftForm |

#### 新建表单那接口

**接口形式：**

action，返回jsp页面

**url:**

[basePath] + wf/formmgr/newProcess

**参数:**

|  |  |  |
| --- | --- | --- |
| **序号** | **参数** | **说明** |
| 1 | processID | 流程ID，引用于(twf\_process.processid) |
| 2 | isnew | 参数值=1，使用新审批页面（基于jQuery）；否则使用老审批页面（基于Ext） |

**返回值：**

/wf/formmanager/wizardItemSubmit.jsp

#### 待办审批表单接口

**接口形式：**

action，返回jsp页面

**url:**

[basePath] + wf/formmgr/formApproveForm

**参数:**

|  |  |  |
| --- | --- | --- |
| **序号** | **参数** | **说明** |
| 1 | workItemID | 流程实例当前工作项ID，引用于(twf\_workitem.workitemid) |
| 2 | isnew | 参数值=1，使用新审批页面（基于jQuery）；否则使用老审批页面（基于Ext） |

**返回值：**

/wf/formmanager/formApprove.jsp

#### 个人跟踪表单接口

**接口形式：**

action，返回jsp页面

**url:**

[basePath] + wf/formmgr/viewInTransForm

**参数:**

|  |  |  |
| --- | --- | --- |
| **序号** | **参数** | **说明** |
| 1 | workItemID | 流程实例当前工作项ID，引用于(twf\_workitem.workitemid) |
| 2 | isnew | 参数值=1，使用新审批页面（基于jQuery）；否则使用老审批页面（基于Ext） |

**返回值：**

/wf/formmanager/viewInTransForm.jsp

#### 审批完成表单接口

**接口形式：**

action，返回jsp页面

**url:**

[basePath] + wf/formmgr/viewFinishedForm

**参数:**

|  |  |  |
| --- | --- | --- |
| **序号** | **参数** | **说明** |
| 1 | processInstID | 流程实例ID，引用于(twf\_processinst.processinstid) |
| 2 | isnew | 参数值=1，使用新审批页面（基于jQuery）；否则使用老审批页面（基于Ext） |

**返回值：**

/wf/formmanager/viewInTransForm.jsp

#### 草稿表单接口

**接口形式：**

action，返回jsp页面

**url:**

[basePath] + wf/formmgr/openDraftForm

**参数:**

|  |  |  |
| --- | --- | --- |
| **序号** | **参数** | **说明** |
| 1 | entityDataID | 业务表单ID，引用于业务表单的唯一ID。将保存到业务实体表中。 |
| 2 | processID | 流程ID，引用于(twf\_process.processid) |
| 3 | isnew | 参数值=1，使用新审批页面（基于jQuery）；否则使用老审批页面（基于Ext） |

**返回值：**

/wf/formmanager/newItemSubmit.jsp

### 工作流审批接口

工作流审批接口提供了流程实例的发起、审批、退回、删除等操作。

工作流审批接口提供了Spring Bean、与Http API两种形式。

#### 接口清单

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | **接口** | **说明** | **操作界面** |
| 1 | 获取审批人接口 | 获取下一环节审批人 | 待办审批 |
| 2 | 发起环节获取审批人接口 | 发起环节获取下一环节审批人 | 新建/草稿管理 |
| 3 | 新建接口 | 发起一个新流程实例 | 新建 |
| 4 | 提交接口 | 将流程实例提交到下一步 | 待办审批 |
| 5 | 回退接口 | 退回流程实例到之前任意环节 | 待办审批 |
| 6 | 撤回接口 | 在流程实例未被审批之前，由发起人自行撤回到发起人环节。 | 个人跟踪 |
| 7 | 拒绝接口 | 将流程实例退回到发起人环节 | 待办审批 |
| 8 | 删除接口 | 删除自己发起的流程实例。流程实例必须已退回到第一个人工环节（发起人环节）。 | 待办审批 |
| 9 | 转办接口 |  | 待办审批 |
| 10 | 转拟办接口 |  | 待办审批 |
| 11 | 草稿保存接口 |  | 新建 |
| 12 | 草稿删除接口 |  | 草稿管理 |

#### 接口参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **接口** | **参数** | **参数名称** | **必填** | **说明** |
| 1 | 获取审批人接口 | userID | 发起人ID | 必填 |  |
| 2 | processInstID | 流程实例ID | 必填 |  |
| 3 | activityInstID | 当前活动实例ID | 必填 |  |
| 4 | workDataJson | 流程属性Json | 必填 |  |
| 5 | 返回值 | 下一环节审批人列表 |  |  |
| 6 | 发起环节获取审批人接口 | userID | 发起人ID | 必填 |  |
| 7 | processID | 流程ID | 必填 |  |
| 8 | activityID | 当前活动ID | 必填 |  |
| 9 | workDataJson | 流程属性Json | 必填 |  |
| 10 | 返回值 | 下一环节审批人列表 |  |  |
| 11 | 新建接口 | userID | 发起人ID | 必填 |  |
| 12 | processID | 流程ID | 必填 |  |
| 13 | activityID | 活动ID | 必填 |  |
| 14 | formID | 表单ID | 必填 |  |
| 15 | approveUserJson | 提交审批人json | 必填 |  |
| 16 | checkDesc | 审批意见 | 必填 |  |
| 17 | formSerialNo | 表单编号 | 必填 |  |
| 18 | formDataJson | 表单数据Json |  |  |
| 19 | workDataJson | 流程属性Json |  |  |
| 20 | 返回值 | 新流程的流程实例ID |  |  |
| 21 | 提交接口 | workItemID | 当前待办工作项ID | 必填 |  |
| 22 | approveUserJson | 提交审批人json | 必填 |  |
| 23 | checkDesc | 审批意见 | 必填 |  |
| 24 | formDataJson | 表单数据Json |  |  |
| 25 | workDataJson | 流程属性Json |  |  |
| 26 | 回退接口 | workItemID | 当前待办工作项ID | 必填 |  |
| 27 | activityID | 退回到活动ID | 必填 |  |
| 28 | userIdJson | 退回到审批人ID | 必填 |  |
| 29 | checkDesc | 审批意见 | 必填 |  |
| 30 | formDataJson | 表单数据Json |  |  |
| 31 | workDataJson | 流程属性Json |  |  |
| 32 | 撤回接口 | processInstID | 流程实例ID | 必填 |  |
| 33 | userID | 操作用户ID | 必填 |  |
| 34 | 拒绝接口 | workItemID | 当前待办工作项ID | 必填 |  |
| 35 | checkDesc | 审批意见 | 必填 |  |
| 36 | formDataJson | 表单数据Json |  |  |
| 37 | workDataJson | 流程属性Json |  |  |
| 38 | 删除接口 | processInstID | 流程实例ID | 必填 |  |
| 39 | userID | 操作用户ID | 必填 |  |
| 40 | 转办接口 | workItemID | 当前待办工作项ID | 必填 |  |
| 41 | userID | 转办审批人ID | 必填 |  |
| 42 | checkDesc | 审批意见 | 必填 |  |
| 43 | formDataJson | 表单数据Json |  |  |
| 44 | workDataJson | 流程属性Json |  |  |
| 45 | 转拟办接口 | workItemID | 当前待办工作项ID | 必填 |  |
| 46 | userID | 转拟办审批人ID | 必填 |  |
| 47 | checkDesc | 审批意见 | 必填 |  |
| 48 | formDataJson | 表单数据Json |  |  |
| 49 | workDataJson | 流程属性Json |  |  |
| 50 | 草稿保存接口 | processID | 流程ID | 必填 |  |
| 51 | formID | 表单ID | 必填 |  |
| 52 | formSerialNo | 表单编号 | 必填 |  |
| 53 | formDataJson | 表单数据Json |  |  |
| 54 | 返回值 | 新草稿的业务实体数据ID |  |  |
| 55 | 草稿删除接口 | entityDataID | 业务实体数据ID | 必填 |  |

##### 参数：表单数据Json

表单数据Json提供了流程实例对应的业务表单数据。在流程实例流过程中，工作流引擎根据此参数，保存或更新业务表单数据。

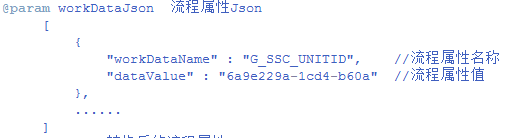
该参数格式与3.1.6.2“表单数据接口”格式相同，为一个Json对象形式的字符串。



##### 参数：流程属性Json

该参数用来在工作流审批时更新该流程的流程属性值。（流程属性值存放在twf\_workdatavalue表中）。

该参数为json数组形式的字符串。数组内每个json对象代表一个流程属性值。



#### Spring Bean接口

引用com.newtouch.workflow.engine.bp.WfEngineApproveInterfaceBP，由Spring管理和依赖注入。

##### 接口方法

各接口对应方法如下：

|  |  |  |
| --- | --- | --- |
| **序号** | **接口** | **method** |
| 1 | 获取审批人接口 | getNextActivityUser |
| 2 | 发起环节获取审批人接口 | getSMANextActivityUser |
| 3 | 新建接口 | start |
| 4 | 提交接口 | submit |
| 5 | 回退接口 | rollback |
| 6 | 撤回接口 | undo |
| 7 | 拒绝接口 | refuse |
| 8 | 删除接口 | delete |
| 9 | 转办接口 | transform |
| 10 | 转拟办接口 | transmit |
| 11 | 草稿保存接口 | saveDraft |
| 12 | 草稿删除接口 | deleteDraft |

##### 获取审批人接口

**接口声明：**



**参数说明：**

参见“4.2.2.2接口参数”

**返回值：**

下一环节审批人列表

##### 发起环节获取审批人接口

**接口声明：**



**参数说明：**

参见“4.2.2.2接口参数”

**返回值：**

下一环节审批人列表

##### 新建接口

**接口声明：**



**参数说明：**

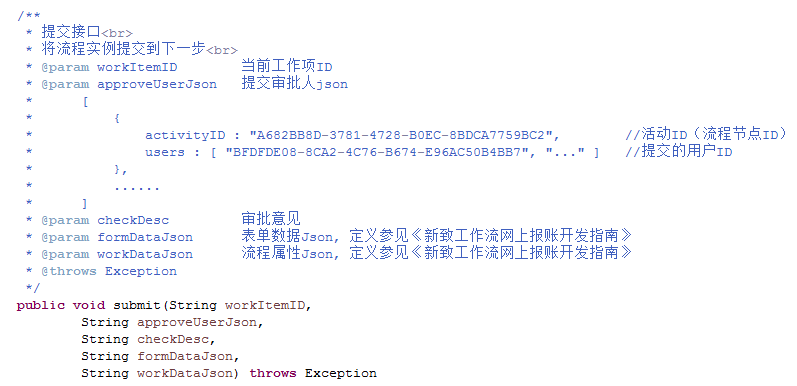
参见“4.2.2.2接口参数”

**返回值：**

新发起流程的流程实例ID

##### 提交接口

**接口声明：**



**参数说明：**

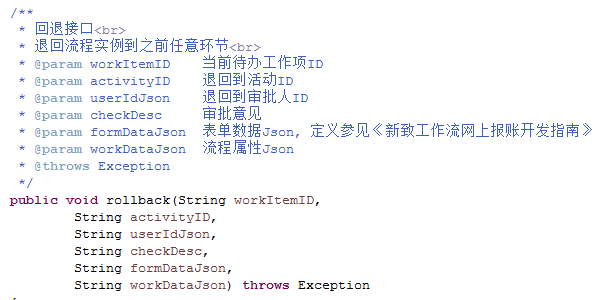
参见“4.2.2.2接口参数”

**返回值：**

无

##### 回退接口

**接口声明：**



**参数说明：**

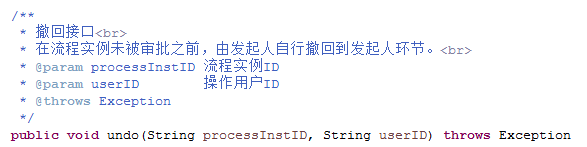
参见“4.2.2.2接口参数”

**返回值：**

无

##### 撤回接口

**接口声明：**



**参数说明：**

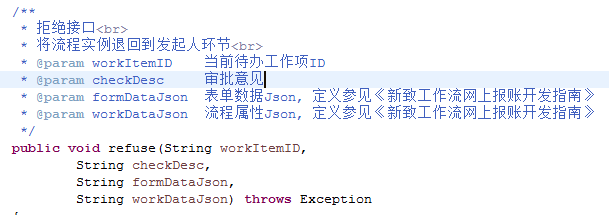
参见“4.2.2.2接口参数”

**返回值：**

无

##### 拒绝接口

**接口声明：**



**参数说明：**

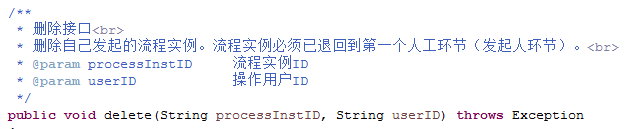
参见“4.2.2.2接口参数”

**返回值：**

无

##### 删除接口

**接口声明：**



**参数说明：**

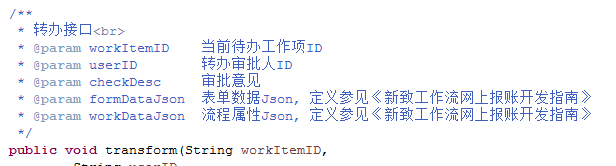
参见“4.2.2.2接口参数”

**返回值：**

无

##### 转办接口

**接口声明：**



**参数说明：**

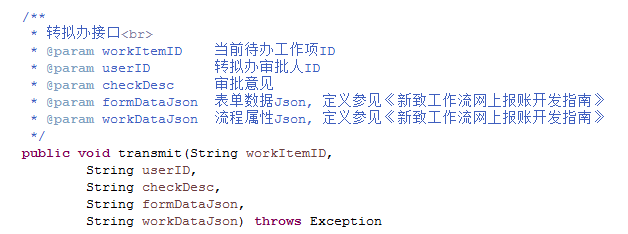
参见“4.2.2.2接口参数”

**返回值：**

无

##### 转拟办接口

**接口声明：**



**参数说明：**

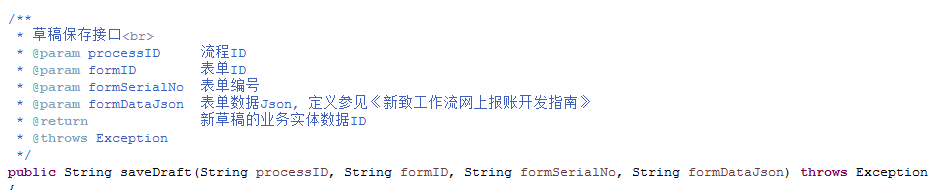
参见“4.2.2.2接口参数”

**返回值：**

无

##### 草稿保存接口

**接口声明：**



**参数说明：**

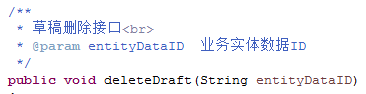
参见“4.2.2.2接口参数”

**返回值：**

已保存草稿的业务数据ID

##### 草稿删除接口

**接口声明：**



**参数说明：**

参见“4.2.2.2接口参数”

**返回值：**

无

#### http API接口

##### 接口URL

http各个接口url如下所示：

|  |  |  |
| --- | --- | --- |
| **序号** | **接口** | **url** |
| 1 | 获取审批人接口 | /wf/approve/getnextactivityuser |
| 2 | 发起环节获取审批人接口 | /wf/approve/getsmanextactivityuser |
| 3 | 新建接口 | /wf/approve/start |
| 4 | 提交接口 | /wf/approve/submit |
| 5 | 回退接口 | /wf/approve/rollback |
| 6 | 撤回接口 | /wf/approve/undo |
| 7 | 拒绝接口 | /wf/approve/refuse |
| 8 | 删除接口 | /wf/approve/delete |
| 9 | 转办接口 | /wf/approve/transform |
| 10 | 转拟办接口 | /wf/approve/transmit |
| 11 | 草稿保存接口 | /wf/approve/savedraft |
| 12 | 草稿删除接口 | /wf/approve/deletedraft |

##### 获取审批人接口

**接口URL:**

参见“4.2.2.4.1”接口URL

**参数说明：**

参见“4.2.2.2接口参数”

**返回值：**

##### 发起环节获取审批人接口

**接口URL:**

参见“4.2.2.4.1”接口URL

**参数说明：**

参见“4.2.2.2接口参数”

**返回值：**

##### 新建接口

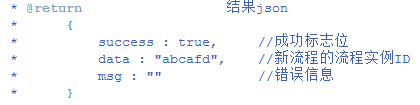
**接口URL:**

参见“4.2.2.4.1”接口URL

**参数说明：**

参见“4.2.2.2接口参数”

**返回值：**



##### 提交接口

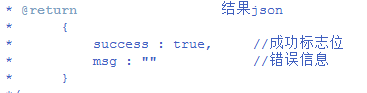
**接口URL:**

参见“4.2.2.4.1”接口URL

**参数说明：**

参见“4.2.2.2接口参数”

**返回值：**



##### 回退接口

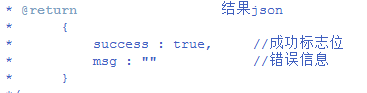
**接口URL:**

参见“4.2.2.4.1”接口URL

**参数说明：**

参见“4.2.2.2接口参数”

**返回值：**



##### 撤回接口

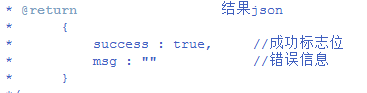
**接口URL:**

参见“4.2.2.4.1”接口URL

**参数说明：**

参见“4.2.2.2接口参数”

**返回值：**



##### 拒绝接口

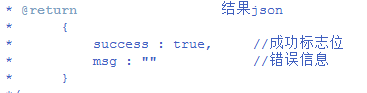
**接口URL:**

参见“4.2.2.4.1”接口URL

**参数说明：**

参见“4.2.2.2接口参数”

**返回值：**



##### 删除接口

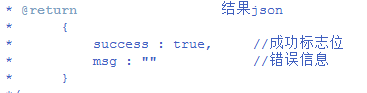
**接口URL:**

参见“4.2.2.4.1”接口URL

**参数说明：**

参见“4.2.2.2接口参数”

**返回值：**



##### 转办接口

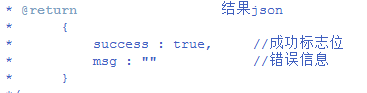
**接口URL:**

参见“4.2.2.4.1”接口URL

**参数说明：**

参见“4.2.2.2接口参数”

**返回值：**



##### 转拟办接口

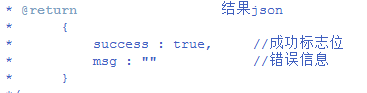
**接口URL:**

参见“4.2.2.4.1”接口URL

**参数说明：**

参见“4.2.2.2接口参数”

**返回值：**



##### 草稿保存接口

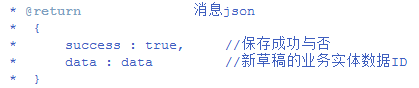
**接口URL:**

参见“4.2.2.4.1”接口URL

**参数说明：**

参见“4.2.2.2接口参数”

**返回值：**



##### 草稿删除接口

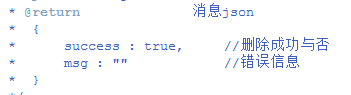
**接口URL:**

参见“4.2.2.4.1”接口URL

**参数说明：**

参见“4.2.2.2接口参数”

**返回值：**



### 流程图接口

待定

### 审批历史接口

待定