**工作流使用说明**

—基础软件部

摘要

本文档主要从使用的角度介绍工作流，同时介绍了部分工作流设计层面的内容。本文将分别从数据库设计、前端页面配置、调用接口三个方面对工作流进行介绍，并说明如何进行使用。建议首先阅读《工作流业务分析文档》，这样在阅读本文档时会有更好的理解。

一、数据库设计

分别从表结构、引擎两个方面介绍工作流的数据库设计

1.表结构

分别从结构、实例两个方面介绍工作流的表结构

1.1结构部分

工作流结构三要素：流程、环节、流向，如图1

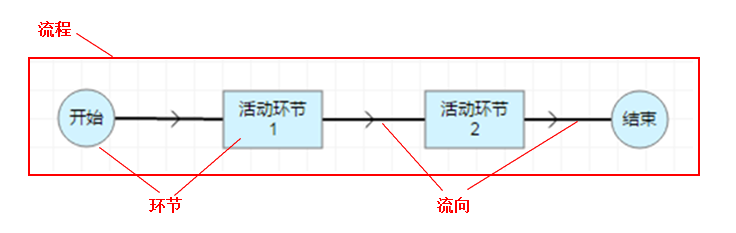


图1

--SYS\_WORKFLOW\_TYPE：流程类别表，类别分上下级

--SYS\_WORKFLOW：流程表，存放流程基本信息，流程与流程类别是多对一关系

--SYS\_WORKFLOW\_PAGE：流程页面表，流程与页面是多对多关系

--SYS\_WORKFLOW\_VARIABLE：流程变量表，流程与变量是一对多关系

--SYS\_NODE：环节表，存放环节基本信息，环节与流程是多对一关系

--SYS\_TRANSACT\_NODE：办理环节表，存放办理人

--SYS\_ACTIVITY\_NODE：活动环节表，存放办理方式

--SYS\_DECISION\_NODE：决策环节表，存放决策类型

--SYS\_NESTED\_NODE：嵌套环节表，绑定子流程

--SYS\_COMPOSITE\_NODE：复合环节表，绑定子流程

--SYS\_NODE\_PAGE：环节页面表，环节与页面是多对多关系

--SYS\_ROUTER：流向表，连接一个流程中的各个环节

1.2实例部分

工作流运行时，以实例的方式进行：流程实例、环节实例、任务实例。

流程实例：当办理业务时，每发起一个流程即产生一个该流程的实例

环节实例：业务流程运行到每个环节时，产生相应的环节实例

任务实例：环节实例结合办理角色产生基于每个人的任务实例（简称任务）

工作流的运行依赖于实例状态的改变。

--SYS\_WORKFLOW\_INSTANCE：

流程实例表，流程实例与流程是多对一关系，一个流程实例对应一个业务数据

--SYS\_WORKFLOW\_VARIABLE\_INSTANCE：

流程变量实例表，对流程变量赋值

--SYS\_NODE\_INSTANCE：

环节实例表，环节实例与环节、流程实例均是多对一关系

--SYS\_TASK\_INSTANCE：

任务实例表，任务实例与环节实例、流程实例均是多对一关系

--SYS\_TASK\_WF\_PAGE\_INSTANCE：

任务表单实例表，任务实例与表单是多对多关系

补充说明：以上主要从功能的角度进行了介绍，并未详细介绍表中每个具体字段，且以上只列举了对理解工作流最有帮助的表，详细表结构设计文件可从svn获取。

2.引擎

分别从存储过程、存储函数两个方面介绍工作流引擎

2.1存储过程

工作流引擎由PKG\_TASK于PKG\_WF这两个包及包体中的存储过程构成

--PKG\_TASK：此包中存放了多个与任务相关的存储过程

--PKG\_WF：此包中存放了多个与流程相关的存储过程

--PKG\_WF\_DAMIC\_USER：此包主要用于存放与动态角色相关的存储过程

2.2存储函数

--SPLITSTR：返回由指定符号分隔的数据集合

--SPLITSTR\_QC：对逗号分隔的字符串去重再拼接

--RX\_COUNT：统计指定字符串出现次数

--RX\_GETSTR：获取指定位置的字符串

--RX\_INSTR：判断一个字符串是否出现

以上存储函数在工作流引擎的存储过程已被使用，当然也可以用于非工作流的地方。

二、前端页面配置

1.页面布局（图2）

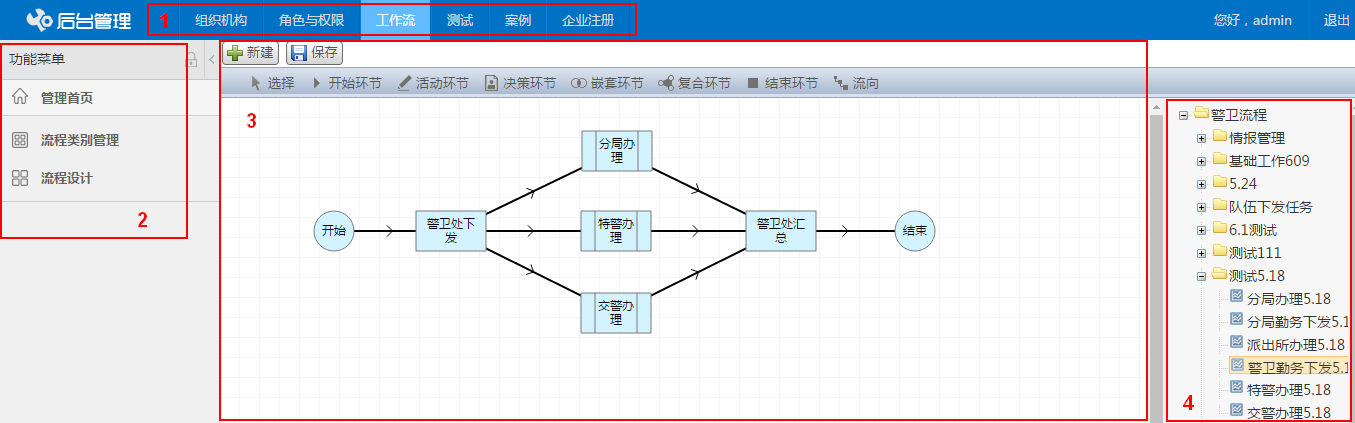


图2

--1菜单区：点击菜单区切换功能区页面展示内容

--2功能区：点击功能区切换到流程图绘制区

--3流程图绘制区：在此区域绘制具体的业务流程图

--4流程树展示区：点击具体流程，展示该流程结构

2.参数配置

2.1流程配置

依次点击“流程设计”、“新建”进入流程设置页面，如图3

2.1.1基本信息

--流程名称：结合业务进行设置

--流程类别：弹出层，从流程类别树上选择一个分类即可（必选项）

--编码：一般设置为流程名称的拼音首字母（必填项，不可重复）

--流程类型：当一个流程中有复合环节（参考2.2.5）时，此项设为复合，否则设为普通

--流程实例标题：一般设置为与流程名称一致（必填项）

--业务状态字典：弹出层，从列表中选择存放业务状态信息的字典

--其他配置项：流程版本号、原始流程名称、优先级别 使用默认配置即可

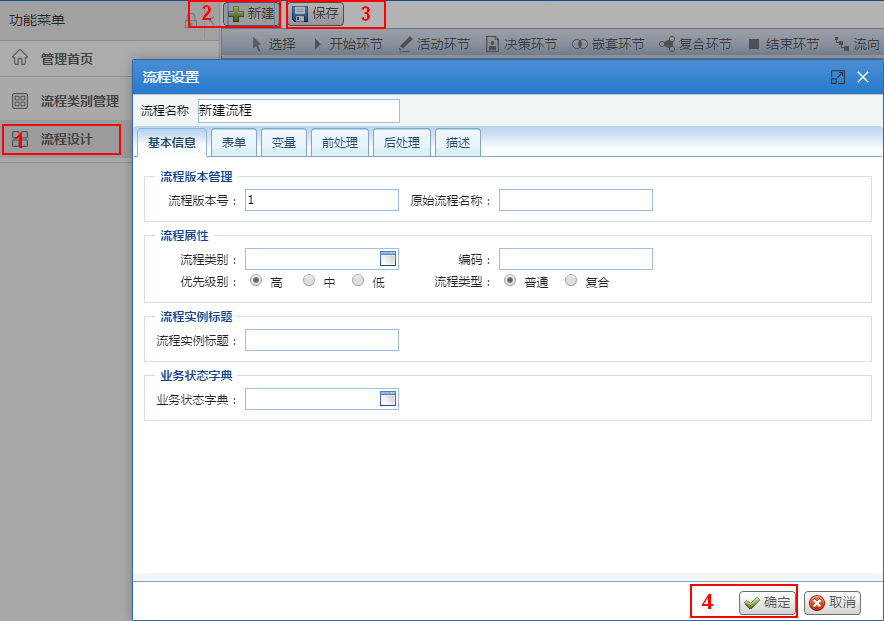


图3

2.1.2表单



图4

--选择表单：弹出层，选择流程中需要使用的业务表单，可选择多个

--标题、序号：选择表单后自动生成

每选择一个表单后需要点击“保存”。

2.1.3变量



图5

--变量名、默认值：设置流程变量及其默认值，一般用于决策环节选择不同流程走向， 可配置多个变量

每配置一个流程变量后需要点击“保存”。

2.1.4前处理、后处理、描述

--前处理：配置流程的前置执行程序，可配置为sql语句、存储过程

--后处理：配置流程的后置执行程序，可配置为sql语句、存储过程

--描述：可存放流程相关描述信息

2.1.5保存流程

上述流程配置均完成后，依次点击图3 中“确定”、“保存”，新建的流程将被保存，可 在图2中的流程树展示区查看。

2.2环节配置

环节配置前，需要首先绘制流程图（假设已完成），下面进行不同环节配 置介绍。

2.2.1开始环节、结束环节

一个流程只有一个开始环节和一个结束环节，在流程图中绘出即可，不需进行其他配置

2.2.2活动环节

2.2.2.1环节设置

--多人办理方式：当配置为抢占时，某项工作经由多个有权限的人办理，但只需一人办理完成即可；当配置为会签时，则需要结合会签参数及值进行判断。

--会签参数及值：多人办理方式为会签时生效，下拉选择会签参数（**全部**：某项工作需有办理权限的所有人员办理 **比例**：某项工作需要有办理权限的人中，实际办理人数达到设定的比例 **固定**：某项工作需要有办理权限的人中，实际办理人数达到设定的固定人数）。

--环节序号：配置该环节在流程中的序号，各环节进行配置时注意不要重复即可

--不同意跳转至：设置退回环节，默认情况下退回上一环节

--是否显示保存按钮：配置任务办理页面“保存”按钮的显隐

--是否显示机构名称：配置是否显示办理机构的名称

--环节属性：通常设置为普通；当流程运行过程中当前环节后有多个需要办理的环节或当前环节后有嵌套环节时，设置为分支；当流程运行过程中当前环节后前多个需要办理的环节或当前环节前有嵌套环节时，设置为聚合。

--环节业务状态：下拉选择该环节对应的业务状态，其选项来自于流程设置中的业务状态字典



图6

2.2.2.2办理人

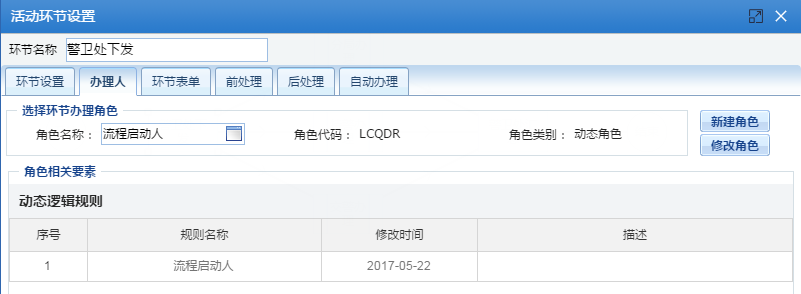


图7

--角色名称：弹出层，选择该环节的办理角色，工作流运行时会根据角色查找具体办理人（必选项）

2.2.2.3环节表单



图8

--选择表单：下拉选择流程配置中已配置的具体表单

--序号、标题：选择表单时，自动生成

--表单控制：查看（不可编辑） 办理（可编辑） 审批（可编辑） 签章（建委业务个性需求）

2.2.2.4前处理、后处理、自动办理

--前处理：配置环节的前置执行程序，可配置为sql语句、存储过程

--后处理：配置环节的后置执行程序，可配置为sql语句、存储过程

--自动办理：警保项目个性需求

2.2.3决策环节

--决策类型：设置为自动决策即可，手动决策供扩展新需求时使用



图9

2.2.4嵌套环节、复合环节

--活动名称：设置环节名称

--选择子流程：从流程树选择子流程

--退回的默认节点：设置退回到子流程中的默认环节

--环节业务状态：下拉选择该环节对应的业务状态，其选项来自于流程设置中的业务状态字典

--是否显示机构名称：设置是否显示机构名称

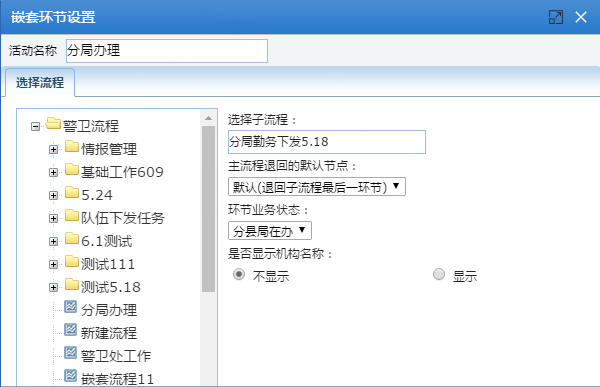


图10

嵌套环节：绑定子流程，主业务流程执行到在嵌套环节时，产生多个子业务流程，主流程与子流程业务上差异大，且有依赖关系，如上下级关系，如：警卫处某工作流程需要下发到各个分县局办理，警卫处工作的完成依赖于各个分县局工作的完成，警卫处业务为主流程，各分县局业务为子流程，在警卫处主流程中有一个“分县局办理”的嵌套环节。嵌套环节绑定的子流程会产生流程实例。

复合环节：绑定子流程，主业务流程执行到在复合环节时，并没有产生子业务流程，而是继续依次执行子流程中包含的若干环节，将若干环节组织为一个子流程并绑定到主流程中复合环节的目的在于，可以使得主流程不用关注子流程中各环节具体执行过程，主流程只需关注复合环节本身。实际上，复合环节只是以流程的方式组织了若干环节并将该流程绑定于自身，复合环节只是为了绑定那若干环节。使用复合环节可以实现将若干环节合并展示为一个环节的前端效果。复合环节绑定的子流程不会产生流程实例，仍然使用了主流程的流程实例。

2.3流向配置

流向连接环节，可根据流向设置查找环节

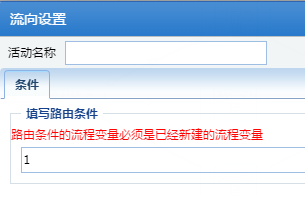


图11

--活动名称：设置流向名称

--路由条件：设置流程变量的值，如[wf:IS\_OPEN] ='1'

三、调用接口

主要介绍工作流开发过程中，使用到的前后端接口。

1.前端接口

1.1启动流程接口

*confirmWorkflowStart*(dataId, flowCode, buildParam, sourceData, title, toTag)

// dataId:业务数据ID，启动时为null

// flowCode:流程编码，必填

// buildParam:传递的参数

// sourceData:null

// title:流程实例标题

// toTag:不使用该参数

一般在前端页面通过点击事件手动调用此接口，调用后会产生流程实例，第一办理环节实例及任务。

1.2工作流任务办理接口

*handleWorkflowByCodeAndDataId*(dataId, flowCode, title, par, toTag)

// dataId:业务数据ID

// flowCode:流程编码

// title:标题

// par:个性需求参数

1.3 保存草稿接口

saveDraft():保存草稿，不保存业务数据

1.4 保存业务数据接口

sheetSave():保存表单业务数据，不提交任务

1.5 表单验证接口

checkSheetSubmit():提交任务前验证表单输入项

1.6 提交接口

sheetSubmit():保存业务数据，提交当前任务

1.7 删除接口

sheetDelete():删除业务数据及流程数据

1.8页面获取流程参数接口

var param = top.winData(window);// 获取流程参数

param.WdataId //业务数据ID

param.wid //流程实例ID

param.rwId //任务ID

param.lcId //流程ID

param.sort //环节序号

param.tmpData //草稿json数据

2.后端接口

2.1 更新业务数据ID接口：

void updateWorkflowInstanceData(FlowParam param, Long dataId, String title);

//将业务数据ID更新到流程实例表中，与流程实例建立对应关系

// param:流程参数

// dataId:业务数据ID

// title:流程实例标题

2.2 更新流程变量实例接口

boolean initVariable(FlowParam param, String name, String value);

//初始化流程变量

// param:流程参数

// name:流程变量名

// value:流程变量值