

慧正 workflow 引擎 API 说明

(试用版)

北京慧正通软科技有限公司

2012 年 5 月

目录

一、 总体

1. 业务操作说明

业务操作说明		
功能	方法	参数说明
1.	<pre>XMLWork xWork = new XMLWork(String userid,String username,String deptname)</pre>	必要参数： Userid: 用户 Id UserName用户名称 DeptName: 部门名称
	<pre>int init = (" ",flowID, " ");</pre>	根据流程定义 Id ,启动流程实例 必要参数： flowID: 流程定义 Id
2.	<pre>XMLWork xWork = new XMLWork(String userid,String username,String deptname)</pre>	设置 XMLWor的基本属性 必要参数： Userid: 用户 Id UserName用户名称 DeptName: 部门名称
	<pre>int init = (workid , " ", " ");</pre>	根据流程实例 Id ,打开流程

		实例 必要参数： workid : 流程实例 Id
--	--	---------------------------------

2. 流程参数说明

具体可参考 /horizon/workflow/include2/

流程参数总体说明			
参数名	含义	格式要求	必须赋值
flowIdentifier	流程数据源标识	或者配置，具体参考	
dataIdentifier	业务数据源标识		
UserID	操作人用户 Id		
UserName	操作人用户名称		
DeptName	操作人部门名称		
WorkID	流程实例 id		
NodeID	流程实例当前节点 Id		
NodeName	流程实例当前节点名称		

TrackID	路径 Id		
		<p>果提交到下一点，只有一个节点时，请不要赋值，置为空；</p> <p>2. 当下一节点有多个节点，根据路由条件无法确定时，可以指定其中一个或者多个来确定流程走向。</p> <p>传入格式：（必须从返回格式中获取传入数据）</p> <p>rld1~nld1 rld2~nld2&rld3~nld3</p> <p>路由 ID~节点 ID。</p> <p>3. 如果从当前节点（A）到目标节点（C）之间有其他节点（B）被跳过时格式为：路由 B~节点 B 路由 C~节点 C</p> <p>4. 多个节点被选择时，使用&连接</p>	
msgSendFlag	发送消息方式	ToDo Mail Msg Note RTF Other	如果在配置 ReChgMsgFl

			ag 节点为 true 时需要传入发送消息的方式
ActionName	操作类型	Save 等,具体参考总体中的流程执行操作	
allowUnitNext	合并节点,是否允许执行提交操作	1: 允许,其他不允许	合并节点时为必填项
cancelFlag	撤销时的标志,默认删除	1=删除,2=回收站	只有执行撤销操作时为必填项
urlapp	待办链接地址	如果为空则默认使用流程框架页面打开	
Comments	意见内容		
selectFlag	当前节点如果替换处理人和转办时,是否允许选择多人	1: 允许,其他不允许	执行替换和转办时为必填项
submitflag	操作提交状态	(1) 提交操作时的含义: 1=获取下一节点信息,返回是否选择节点,是否需要选择人员,用于跟前台交互, 0=执行提交到下一节点办理人,如果处理中遇到节点不明确或者办理人不明确	

		<p>时，跟 1 效果一样。当需要跟前台操作人进行交互，比如获取是否需要选择节点或者选择办理人，是否需要操作人进行确认等，建议使用 1，处理速度会快一点。</p> <p>(2) 跳转操作时的含义： 1=获取可以跳转的节点列表，0=执行跳转到指定节点。</p> <p>(3) 退回操作时的含义： 1=获取可以退回的节点列表，0=执行退回到指定节点操作。</p> <p>(4) 启动子流程时的含义： 如果系统提供了启动子流程的 ID（参数 selectFlowID），则忽略此参数，不提供 selectFlowID 参数时，如果当前人没有启动过子流程时忽略此参数，直接根据定义判断是否有多个子流程定义，如果只有一个子流程定义，则直接启动，否则返回；如果当前人已经启动过子流程，1=获取</p>	
--	--	--	--

		已经启动的子流程列表，并返回，0=根据定义判断是否有多个子流程定义，如果只有一个子流程定义，则直接启动，否则返回。 (5) 催办操作时的含义： 1=催办次数如果已经达到上限则停止，0=催办次数已经达到上限也继续催办	
curNodeAuthorSendFlag	当前节点的作者消息发送方式，用于执行会签，加签，替换，转办	格式同 msgSendFlag	
curNodeReaderSendFlag	当前节点的读者消息发送方式，用于执行阅知操作时显示在前台供操作者选择		
selectFlowID	启动子流程的Id		启动子流程时为必填项
FormAction	流程提交操作的URL		
tmpAuthor	选择人员时，临时获取选择	格式同 SelectAuthor	

	范围		
preFlag	如果有选择节点页面，在确认页面上出现上一步按钮	1: 显示，其他不出现	
ActionClass	执行业务的 action 名称	默认	
	临时信息记录 (目前用于存储当前人填写的意见信息)		

3. 流程执行操作说明

流程执行操作		
1. (具体点击上面链接)	引入的包： (1) 传入 request 对象，调用 doAction XMLApp app = new XMLApp(); LinkedHashMap backMap = (request);	从 request 中获取的相关参数 参考流程参数总体说明，以及要执行下面具体操作的说明

(具体点击上面链接)	<pre> LinkedHashMap hashMap = new LinkedHashMap(); 2. 在自定义的类中执行 int result = doAction(null,hashMap); </pre>	组织流程操作需要的参数 具体根据流程参数总体说明以及对应的操作设置相关参数，放入到hashMap中。
------------	---	---

4. 流程实例流转操作

操作名称	英文标识	说明
提交	Submit	提交到下一节点
退回	reject	退回到某个节点
拿回	getback	拿回已提交的流程
阅知	addreader	发送待阅
撤销	cancel	撤销流程
跳转	goto	跳转到某个节点
会签	huiqian	发送给相关人员会签
加签	jiaqian	发送给相关人员加签
转办	move	转交给其他人办理
暂停	pause	暂停流程
已阅	readed	阅读待阅的信息
催办	remsg	催办流程实例

替换办理人	replace	替换流程的办理人
暂存	save	暂存流程实例
终止流程	stopflow	终止流程实例
启动子流程	subflow	启动子流程

二、 启动实例

1. 启动流程实例整体说明

引 入 类：		
执行方 法	初始化 XMLWork 设置相关属性	1. XMLWork xWork = new XMLWork(userid,username,deptname);
	启动具体的流程实例 (返回值说明见下面)	init = (" " ,flowID, " ");

2. 启动流程实例参数说明

参数名	含义	要求
userid	启动实例的操作人的用户 Id	不能为空
username	启动实例的操作人的用户名称	可以为空

deptname	启动实例的操作人的部门名称	可以为空
flowID	流程定义 Id	不能为空

3. 启动流程初始化 XMLWork 额外属性说明

初始化 XMLWork 设置相关属性（根据业务，如有需要，可以设置以下内容）		
()	设置流程数据所使用的数据源标识符。	中设置的默认数据源和中的设置不一致时，设置为空和设置成 null（或者不设置）是有区别的，设置为空时，系统使用中设置的默认数据源，设置成 null（或者不设置）时，使用的是中的设置。数据源无效时，系统默认使用中设置的默认数据源。如果 flowIdentifier 跟 dataIdentifier 一样可以直接使用（“ ”）对两个变量同时赋值
xWork .setDataIdentifier() ()	设置业务数据所使用的数据源标识符。	
xWork .setUserAuthType()	如果需要设置外部办理人获取方式。	

4. 启动流程实例返回值 [init] 说明

返回值	含义
init=	初始化成功。

init =	当前人员无权启动流程
init =	锁定当前实例失败，启动流程时不影响，可以等同成功
init =	其他原因失败从 (); 可以获取错误信息

启动流程实例成功后，可以从 xWork对象中获取相关的信息，具体参考

5. 启动实例代码样例

```

/** 启动流程实例操作，从前台获取参数，获取流程定义 Id 等信息，具体可参考
horizon/workflow/include2/ */
程定义 Id, 必须有此值
String flowID = ("flowid");
部权限设置分类 ,XMLWork 额外属性, 可以不用设置
String userAuthType = ("userAuthType");
得数据源 ,XMLWork 额外属性, 可以不用设置
String dbIdentifier = ("dbIdentifier");
前用户信息 , 根据应用系统的区别, 灵活获取。下面是产品中的自带的用户获取方式
String userid = (session);          始化 XMLWork 属性
XMLWork xWork =          new XMLWork(userid,username,deptname);
动流程实例 , 第一个参数是流程实例 Id , 第二个参数是流程定义 Id , 第三个参数是业务数据 Id ,
如果三个都有值, 获取流程数据的优先顺序:          workID > flowID > dataID
int in it = ( " ",flowID, " ");
动流程实例成功后, 对业务数据的处理
etCurForm(dataid);
for(int i=0;i<;i++) {

```

```
{
String formid = form[i].getID();
    etFormtype().equals("HTML"){
        etFormtype().equals("JSP"){
            etDataID()
            etID();
            etTableID();
            oString();
            oString();
            oString();
            oString();
            oString();
        }else{
```

6.

		XMLWork xWork = new XMLWork(userid,username,deptname);
	打开具体的流程实例 (返回值说明见下面)	init = (workid , " " , " ");

7. 打开流程实例参数说明

参数名	含义	要求
userid	启动实例的操作人的用户 Id	不能为空
username	启动实例的操作人的用户名称	可以为空
deptname	启动实例的操作人的部门名称	可以为空
workid	流程实例 Id	不能为空

8. 打开流程初始化 XMLWork 额外属性说明

初始化 XMLWork 设置相关属性（根据业务，如有需要，可以设置以下内容）		
()	设置流程数据所使用的数据源标识符。	<p>(1) 中设置的默认数据源和中的设置不一致时，设置为空和设置成 null（或者不设置）是有区别的，设置为空时，系统使用中设置的默认数据源，设置成 null（或者不设置）时，使用的是中的设置。数据源无效时，系统默认使用中设置的默认数据源。</p> <p>(2) 如果 flowIdentifier 跟 dataIdentifier 一样可以直接</p>
()	设置业务数据所使用的数据源标识符。	

		使用(“ ”)对两个变量同时赋值
()	设置指定路径 Id	打开流程实例时可以设置此值 (1) 如果当前人在多个分支节点都具有权限，还需要指定路径 ID (2) 如果不指定，系统会自动获取所有分支，返回前台供选择

9. 打开流程实例返回值 [init] 说明

返回值	含义
init=	选择需要打开的节点 String allnodes = (); String[] nodes = (“\”); 选择后的结果赋值给 (“ ”); return;
init =	打开实例成功
init =	当前人员无权查看此流程实例
init =	当前人只能以只读权限打开（此时不能执行任何操作）
init =	当前实例被其他人锁定（此时不能执行任何操作）
init =	锁定当前实例失败，此时其他人仍然可以打开实例进行操作，可能会出现数据冲突问题，建议退出重新打开
init =	其他原因失败从 (); 可以获取错误信息

打开流程实例成功后，可以从 xWork对象中获取相关的信息，具体参考

10. 打开实例代码样例

```
/** 打开流程实例操作，从前台获取参数，获取流程实例 Id等信息，具体可参考
horizon/workflow/include2/ */
程实例 Id, 必须有此值
String workid = ("workid");
部权限设置分类 ,XMLWork 额外属性, 可以不用设置
String userAuthType = ("userAuthType");
得数据源 ,XMLWork 额外属性, 可以不用设置
String dbIdentifier = ("dbIdentifier");
设置指定路径 Id , XMLWork 额外属性, 如果没有分支, 可以不用设置
String openTrackID = ("openTrackID ");
前用户信息 , 根据应用系统的区别, 灵活获取。下面是产品中的自带的用户获取方式
String userid = (session); 始化 XMLWork 属性
XMLWork xWork = new XMLWork(userid,username,deptname);
setOpenTrackID(openTrackID);
开流程实例
int init = (workid, " ", " ");
开流程实例成功后, 对业务数据的处理
etCurForm(dataid);
for(int i=0;i<;i++) {
{
String formid = form[i].getID();
```



```

etFormtype().equals("HTML"){
    etFormtype().equals("JSP"){
        etDataID()
    etID();
    etTableID();
    oString();
    oString();
    oString();
    oString();
    oString();
    }else{

```

```

oUpperCase(), "");          oUpperCase(), "");          oUpperCase(), "");          oUpperCase(), "0");

```

提交操作时的含义：1=获取下一节点信息，返回是否选择节点，是否需要选择人员，用于跟前台交互，0=执行提交到下一节点办理人，如果处理中遇到节点不明确或者办理人不明确时，跟1效果一样。当需要跟前台操作人进行交互，比如获取是否需要选择节点或者选择办理人，是否需要操作人进行确认等，建议使用1，处理速度会快一点。

跳转操作时的含义：1=获取可以跳转的节点列表，0=执行跳转到指定节点。

退回操作时的含义：1=获取可以退回的节点列表，0=执行退回到指定节点操作。

启动子流程时的含义：如果系统提供了启动子流程的ID（参数 selectFlowID），则忽略此参数，不提供 selectFlowID 参数时，如果当前人没有启动过子流程时忽略此参数，直接根据定义判断是否有多个子流程定义，如果只有一个子流程定义，则直接启动，否则返回；如果当前人已经启动过子流程，1=获取已经启动的子流程列表，并返回，0=根据定义判断是否有多个子流程定义，如果只有一个子流程定义，则直接启动，否则返回。

催办操作时的含义：1=催办次数如果已经达到上限则停止，0=催办次数已经达到上限也继续催办

启动子流程时可能需要的参数，指定子流程 ID

```

("selectFlowID".toUpperCase(), "");

```

```

oUpperCase(), "");

```

```

oUpperCase(), "Submit");oUpperCase(), workID);oUpperCase(), userid);oUpperCase(), "");oUpperCase(), "");oUpperCase(), "");

```

A.

	etNodeID()); ()== , 获取当前节点已经选择的办理人, 用于赋值给

	selectAuthor (pos)

B.

```
1 " sgFlagCN();  
String msgFlag ="";  
String other ="";  
    if(xWork !=null){
```


C.

D.

```
1 " substring(nodes[i].indexOf("~" +1);
  substring(0,nodes[i].indexOf("~"));
}
}
}else{
```


E.

F.

--

G.

H.

要把 selectAuthor 变量赋值，此值是阅知人员 Id
if(cp == {

I.

J.

要将 ("ActionName ", "cancel ");
if(cp == {

--	--	--

K.

L.

```
要把 selectAuthor 变量赋值，此值是会签人员 Id  
if(cp>{  
    1 " substring(nodes[i].indexOf("~")+1) ;  
    substring(0,nodes[i].indexOf("~"));  
}  
}  
}else{0 " sgFlagCN();
```

```
String msgFlag = ?():      "" ;  
String other = ?(msgFlag):  "" ;
```


M.

N.

```
要把 selectAuthor 变量赋值，此值是会签人员 Id  
if(cp == {
```


O.

P.

要把 selectAuthor 变量赋值，此值是加签人员 Id
if(cp == {

Q.

R.

要把 selectAuthor 变量赋值，此值是转办人员 Id
if(cp == {

S.

T.

--

U.

V.

--

W.

--	--

X.

要将催办的内容 Comments 赋值，放到 dataHashMap 的 Comments 对应的值
if(cp == {

Y.

Z.

要把 selectAuthor 变量赋值，此值是替办人员 Id

```
if(cp == {
```


AA.

BB.

CC.

DD.

--

EE.

FF.

```
要将 selectFlowID 选择的子流程放入到 dataHashMap中.  
if(cp == {  
    plit("\\|");  
    for ( int i=0,n=i<n;i++){  
        if (list[i].length()==0) continue ;  
        substring(0,list[i].indexOf("~"));  
    }  
}
```


	ode_Parallel : 多人处理 StaticVar .Node_Order : 多人顺序 StaticVar .Node_Everyone : 多人单一签核
	() 获取当前节点的办理方式
	() 获取当前节点事件列表
	() 获取当前节点当前人所具有的表单信息 (包含表单域和操作信息)
	dataid) 获取当前节点当前人所具有的表单信息 (包含表单域和操作信息) 在已有数据上启动流程时, 增加查找主表单, 并判断是否有 dataid, 如果没有则使用参数赋值
	() 获取当前节点 ID
	() 获取当前节点名称
	() 获取当前节点的读者消息发送方式

	() 获取当前节点所有未办理人员列表中文名称，多个值用“ ”分隔。
	backSplit) 获取当前节点所有未办理人员列表中文名称，使用指定分隔符
	() 获取当前节点所有未办理人员列表 ID，多个值用“ ”分隔。
	() 获取当前节点的可能路由条件
	() 获取当前执行人 ID
	() 获取当前执行人的中文名称
	() 业务数据源名称
	(int i) 获取当前表单域信息，用于表单引擎解析
	formID) 获取当前表单域信息，用于表单引擎解析
	() 获取流程 ID
	() 获取流程的数据源

		() 获取流程名称
	int	() 获取当前分支的流程状态 返回值： StaticVar .FlowStatus_Start : 开始 StaticVar .FlowStatus_Normal : 正常 StaticVar .FlowStatus_HuiQian : 会签 StaticVar .FlowStatus_Reject : 退回 StaticVar .FlowStatus_Getback : 拿回 StaticVar .FlowStatus_End : 正常结束 StaticVar .FlowStatus_ErrorEnd : 异常结束 StaticVar .FlowStatus_Pause : 暂停 StaticVar .FlowStatus_Cancel : 撤销 StaticVar .FlowStatus_Delete : 删除
		() 获取当前分支的流程状态名称
		varName) 必须在打开实例后才能获取到值
		button) 获取表单按钮数组
		button) 根据指定表单按钮生成的 html 内容

	<p>() 获取办理记录 XML 片段,由应用在前台进行显示格式转换</p>
	<p>formID, type) 根据 formID 获取其他属性,一般是获取 TableID 和 DataID 参数: formID : 表单 Id type : TableID: 库表 ID , 业务数据 DataID:</p>
int	<p>() 获取实例化 XMLWork 对象的结果</p>
	<p>() 获取当前锁定人中文名称</p>
	<p>() 获取当前锁定人 ID</p>
	<p>() 获取流程图对应的 XML 数据,用于流程图显示</p>
	<p>parentid) 获取继承文档所在的库表和数据 id</p>
	<p>() 获取上一节点办理人</p>
	<p>backSplit) 获取上一节点办理人,使用指定分隔符</p>
	<p>()</p>

	获取上一节点办理人 id
	() 获取上一节点名称
	(int pos) 获取给下一节点读者发送消息的方式
	(int pos) 获取给下一节点办理人发送消息的方式
int	() 获取选择人员方式 , 单选还是多选 (当前节点)
	id) 获取静态表单内容
	() 获取系统按钮数组
	() 根据对应系统按钮生成的 html 内容
int	() 获取当前人的类型 (为拿回时 , 需要区分是否为起草人) StaticVar .User_MainDo : 主办人 StaticVar .User_SecondDo : 协办办理人 StaticVar .User_Manager : 管理员 StaticVar .User_SecondDone : 协办已办理人 StaticVar .User_CurDone : 当前节点办理人 StaticVar .User_PreDone : 上一节点办理人

	StaticVar .User_Creator : 流程创建者 ,流程起草人 StaticVar .User_Done : 已办理人 StaticVar .User_Huiqian : 会签提出人 StaticVar .User_Reader : 流程读者 ,节点读者 ,阅知人 StaticVar .User_CReader : 读者 ,由其他权限降级或者不参与读者权限 (比如转办人) StaticVar .User_NoRole : 无权限
	() 用于前台显示当前用户类型
	() 获取实例 ID
boolean	() 判断是否为当前操作人锁定
下面的方法用于执行操作，需要跟前台交互时，获取对应信息进行下一步处理	
boolean	() 判断下一节点是否跟当前节点相同，相同为 false，不同为 true。
	() 获取给下一节点作者发送消息的方式。多值时用“ ” 隔开。
	() 获取当前节点办理人的消息发送方式。多值时用“ ” 隔开。
	() 返回从定义列表中选择人员 ，，用于前台从流程节点定义中选择人员。多值时用“ ” 隔开。

Int	getMaySelectNodeNumber() 返回可以选择的节点数量 (包含已经选择的节点)
	获取下一办理人中文名称 ,用于前台显示 ,让操作人最后确认。
	pos) 获取下一办理人中文名称 ,用于前台显示 ,让操作人最后确认。
	backSplit) 获取下一办理人中文名称 ,用于前台显示 ,使用指定分隔符
	backSplit,int pos) 获取下一办理人中文名称 ,用于前台显示 ,使用指定分隔符
	() 获取下一办理人 id ,多值用 “ ” 分隔。
	(int pos) 获取下一办理人 id ,多值用 “ ” 分隔。
	(int pos) 获取下一办理人 ,用于最终完成提交操作提供参数 ,多值用 “ ” 分隔。
	(int pos) 获取协办人 ,用于最终完成提交操作提供参数 ,多值用 “ ” 分隔。
int	() 获取下一节点办理方式 返回值 : : 单一签核

	StaticVar .Node_Parallel : 多人处理 StaticVar .Node_Order : 多人顺序 StaticVar .Node_Everyone : 多人单一签核
int	(int pos) 获取下一节点办理方式
	()后台调用时使用方法
	() 获取下一节点 ID
	(nodeid) 根据节点名称获取在下一节点列表中的位置
	() 获取下一节点名称
int	() 获取下一节点类型 StaticVar .Node_Start: 开始节点 StaticVar .Node_Single : 单一签核 StaticVar .Node_Parallel : 多人处理 StaticVar .Node_Order : 多人顺序 StaticVar .Node_Everyone : 多人单一签核 StaticVar .Node_Unite : 合并节点,用于多路并发中的合并处理,或者其他需要业务自行处理的节点 StaticVar .Node_Robot : 自动节点 StaticVar .Node_End : 结束节点

	StaticVar .Node_Condition : 条件节点 StaticVar .Node_Cluster : 多引擎交互节点
String	getNextNodeTrackID(int pos) 成功提交后获取路径 ID,用于成功后重新打开文档时调用
boolean	() 获取待办是否自动打开。流程操作成功后获取
int	(int pos) 获取选择人员方式 ,单选还是多选 (下一节点)
	() 获取需要选择节点时的对应路由列表
	() 后台代码获取使用
	() 获取当前发送待办页面链接
boolean	(int pos) 返回是否允许自由选择人员