# 流程定义

## 部署流程定义

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| ResultResponse deployProcess (DeploymentInfo deployInfo) |

服务发布地址

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/suspendProcessDef.freepage?+参数 |

方法参数说明

| DeploymentInfo参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| deployName | 是 | String | 部署名称 |
| wfAppId | 否 | String | 所属应用id |
| businessTypeId | 是 | String | 业务类型id |
| needConfig | 是 | String | 是否清除待办处理人和节点参数 1是 2 否 |
| upgradepolicy | 否 | String | 部署策略  0保持旧版本1升级旧版本实例到新版本2删除旧版本流程实例 |
| processDefFile | 是 | byte[] | 部署资源 zip包的字节数组 |
| paramFile | 否 | byte[] | 参数资源 zip包的字节数组 |

返回结果

| 服务方式 | 返回结果 | | | |
| --- | --- | --- | --- | --- |
| Hessian/  webservice | ResultResponse对象 | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  DeploymentInfo info = **new** DeploymentInfo();  info.setBusinessTypeId("e559e851-0142-3a21-d1b5-8ae7c56784c7");  info.setDeployName("test1");  info.setNeedConfig("1");  info.setUpgradepolicy(0);  info.setWfAppId("da8e7c0a-8ef3-46b4-a275-151d41763141");  **byte**[] bytes =  ValueObjectUtil.*getBytesFileContent*("E:\\workspace\\SanyPDP\\test\\com\\  sany\\activiti\\demo\\diagrams\\mms.return.zip");  info.setProcessDefFile(bytes);  ResultResponse hessianResult = hassianService.deployProcess(info); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  DeploymentInfo info = **new** DeploymentInfo();  info.setBusinessTypeId("e559e851-0142-3a21-d1b5-8ae7c56784c7");  info.setDeployName("test1");  info.setNeedConfig("1");  info.setUpgradepolicy(0);  info.setWfAppId("da8e7c0a-8ef3-46b4-a275-151d41763141");  **byte**[] bytes =  ValueObjectUtil.*getBytesFileContent*("E:\\workspace\\SanyPDP\\test\\com\\  sany\\activiti\\demo\\diagrams\\mms.return.zip");  info.setProcessDefFile(bytes);  ResultResponse cxfResult = cxfService.suspendProcessDef(info); |

## 挂起流程定义

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| ResultResponse suspendProcessDef(String processDefId) |

服务发布地址

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/suspendProcessDef.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| processDefId | 是 | String | 流程定义ID |

返回结果

| 服务方式 | 返回结果 | | | |
| --- | --- | --- | --- | --- |
| Hessian/  webservice | ResultResponse对象 | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |
| HTTP请求 | JSON对象：  {"resultCode":"1","resultMess":""挂起操作成功"} | | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  ResultResponse hessianResult = hassianService.suspendProcessDef(""); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  ResultResponse cxfResult = cxfService.suspendProcessDef(""); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /suspendProcessDef.freepage?processDefId=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

## 激活流程定义

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| ResultResponse suspendProcessDef(String processDefId) |

服务发布地址

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/activateProcessDef.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| processDefId | 是 | String | 流程定义ID |

返回结果

| 服务方式 | 返回结果 | | | |
| --- | --- | --- | --- | --- |
| Hessian/  webservice | ResultResponse对象 | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |
| HTTP请求 | JSON对象：  {"resultCode":"1","resultMess":""激活操作成功"} | | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  ResultResponse hessianResult = hassianService.suspendProcessDef(""); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  ResultResponse cxfResult = cxfService.suspendProcessDef(""); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /suspendProcessDef.freepage?processDefId=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

## 删除流程定义

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| ResultResponse delDeployment (String processKeys) |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/delDeployment.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| processKeys | 是 | String | 流程定义key(多个以,号分隔) |

返回结果

| 服务方式 | 返回结果 | | | |
| --- | --- | --- | --- | --- |
| Hessian/  webservice | ResultResponse对象 | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |
| HTTP请求 | JSON对象：  {"resultCode":"1","resultMess":""删除操作成功"} | | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  ResultResponse hessianResult = hassianService.delDeployment(""); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  ResultResponse cxfResult = cxfService.delDeployment(""); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /delDeployment.freepage?processKeys=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

## 获取流程定义XML

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| DataResponse getProccessXML (String processKey, String version) |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/getProccessXML.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| processKey | 是 | String | 流程key |
| version | 是 | String | 流程版本 |

返回结果

| 服务方式 | 返回结果 | | | |
| --- | --- | --- | --- | --- |
| Hessian/  webservice | DataResponse对象 | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |
| resultData | String | 返回数据 |
| HTTP请求 | JSON对象：  {"resultCode":"1",  "resultMess":"获取流程定义XML成功",  "resultData":"<?xml version=\"1.0\" encoding=\"UTF-8\"?>\n  <definitions xmlns=\"http://www.omg.org/spec/BPMN/20100524/**MODEL**\"   xmlns:xsi=\"http://www.w3.org/2001/**XMLSchema**-**instance**\"   xmlns:activiti=\"http://activiti.org/bpmn\"   xmlns:bpmndi=\"http://www.omg.org/spec/BPMN/20100524/DI\"   xmlns:omgdc=\"http://www.omg.org/spec/DD/20100524/DC\"   xmlns:omgdi=\"http://www.omg.org/spec/DD/20100524/DI\"   typeLanguage=\"http://www.w3.org/2001/**XMLSchema**\"   expressionLanguage=\"http://www.w3.org/1999/XPath\"   targetNamespace=\"http://www.activiti.org/test\">\n   <process id=\"Mms.**return**\" name=\"物料退料流程\"   isExecutable=\"**true**\">\n   ...  ...  " } | | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  DataResponse hessianResult = hassianService.getProccessXML("",""); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  DataResponse cxfResult = cxfService.getProccessXML("",""); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /getProccessXML.freepage?processKey=&version=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

## 获取流程定义图片

接口路径

|  |
| --- |
| 无 |

接口方法

|  |
| --- |
| 无 |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| HTTP请求 | 工程上下文/workflow/webservice/getProccessPic.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| processDefId | 是 | String | 流程定义id |
| response | 是 | HttpServletResponse | 请求响应 |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| HTTP请求 | <img src=" http://localhost:8080/SanyPDP/workflow/webservice/getProccessPic.freepage? processDefId=" /> |

# 流程实例

## 开启流程实例

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| ResultResponse startInstance(String processKey, String businessKey,String currentUser, List<HashMap<String, Object>> nodeInfoList,List<HashMap<String, Object>> variableList) |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/startInstance.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 | | |
| --- | --- | --- | --- | --- | --- |
| processKey | 是 | String | 流程key | | |
| businessKey | 否 | String | 业务主题 | | |
| currentUser | 是 | String | 当前用户登录账号 | | |
| nodeInfoList | 否 | List<HashMap> | Key | Value类型 | 节点配置描述 |
| nodeKey | String | 任务节点key |
| nodeUserIds | String | 任务节点可处理人id |
| nodeGroupIds | String | 任务节点可处理组id |
| isMulti | String | 串/并多实例切换  1串行多实例  2并行多实例 |
| durationNode | String | 处理工时/小时  (最多保留一位小数) |
| variableList | 否 | List<HashMap> | Key | Value类型 | 参数描述 |
| paramName | String | 参数英文名 |
| paramValue | String | 参数值 |

返回结果

| 服务方式 | 返回结果 | | | |
| --- | --- | --- | --- | --- |
| Hessian/  webservice | ResultResponse对象 | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |
| HTTP请求 | JSON对象：  {"resultCode":"1","resultMess":"开启实例操作成功"} | | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  ResultResponse hessianResult = hassianService.startInstance("","","",null,null); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  ResultResponse cxfResult = cxfService.startInstance("","","",null,null); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /startInstance.freepage?processKey=&businessKey=&currentUser=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

## 升级流程实例

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| ResultResponse upgradeInstancesByProcessKey(String processKey) |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/upgradeInstancesByProcessKey.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| processKey | 是 | String | 流程key |

返回结果

| 服务方式 | 返回结果 | | | |
| --- | --- | --- | --- | --- |
| Hessian/  webservice | ResultResponse对象 | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |
| HTTP请求 | JSON对象：  {"resultCode":"1","resultMess":"升级操作成功"} | | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  ResultResponse hessianResult = hassianService.upgradeInstancesByProcessKey(""); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  ResultResponse cxfResult = cxfService.upgradeInstancesByProcessKey(""); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /upgradeInstancesByProcessKey.freepage?processKey=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

## 逻辑删除流程实例

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| ResultResponse delInstancesForLogic (String instancesIds, String deleteReason) |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/delInstancesForLogic.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| instancesIds | 是 | String | 流程实例IDs(多个以,号分隔) |
| deleteReason | 是 | String | 删除原因 |

返回结果

| 服务方式 | 返回结果 | | | |
| --- | --- | --- | --- | --- |
| Hessian/  webservice | ResultResponse对象 | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |
| HTTP请求 | JSON对象：  {"resultCode":"1","resultMess":"逻辑删除操作成功"} | | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  ResultResponse hessianResult = hassianService.delInstancesForLogic("",""); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  ResultResponse cxfResult = cxfService.delInstancesForLogic("",""); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /delInstancesForLogic.freepage?instancesIds=&deleteReason=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

## 物理删除流程实例

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| ResultResponse delInstancesForPhysics (String instancesIds) |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/delInstancesForPhysics.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| instancesIds | 是 | String | 流程实例IDs(多个以,号分隔) |

返回结果

| 服务方式 | 返回结果 | | | |
| --- | --- | --- | --- | --- |
| Hessian/  webservice | ResultResponse对象 | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |
| HTTP请求 | JSON对象：  {"resultCode":"1","resultMess":"物理删除操作成功"} | | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  ResultResponse hessianResult = hassianService.delInstancesForPhysics(""); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  ResultResponse cxfResult = cxfService.delInstancesForPhysics(""); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /delInstancesForLogic.freepage?instancesIds=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

## 挂起流程实例

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| ResultResponse suspendInstance (String instancesId) |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/suspendInstance.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| instancesId | 是 | String | 流程实例ID |

返回结果

| 服务方式 | 返回结果 | | | |
| --- | --- | --- | --- | --- |
| Hessian/  webservice | ResultResponse对象 | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |
| HTTP请求 | JSON对象：  {"resultCode":"1","resultMess":"挂起流程实例操作成功"} | | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  ResultResponse hessianResult = hassianService.suspendInstance(""); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  ResultResponse cxfResult = cxfService.suspendInstance(""); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /suspendInstance.freepage?instancesId=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

## 激活流程实例

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| ResultResponse activateInstance (String instancesId) |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/activateInstance.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| instancesId | 是 | String | 流程实例ID |

返回结果

| 服务方式 | 返回结果 | | | |
| --- | --- | --- | --- | --- |
| Hessian/  webservice | ResultResponse对象 | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |
| HTTP请求 | JSON对象：  {"resultCode":"1","resultMess":"激活流程实例操作成功"} | | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  ResultResponse hessianResult = hassianService.activateInstance(""); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  ResultResponse cxfResult = cxfService.activateInstance(""); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /activateInstance.freepage?instancesId=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

## 实例处理记录(详情)

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| DealInfoResponse getInstanceDealInfo (String instancesId) |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/getInstanceDealInfo.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| instancesId | 是 | String | 流程实例ID |

返回结果

| 服务方式 | 返回结果 | | | | |
| --- | --- | --- | --- | --- | --- |
| Hessian/  webservice | DealInfoResponse对象 | | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |
| dataList | List<NodeTaskInfo> | 返回数据集 |
| HTTP请求 | JSON对象：  {"resultCode":"1",  "resultMess":"获取处理记录操作成功",  "dataList":[{"duration":"20分钟34秒",  "endTime":"",  "startTime":"2014-08-05 10:27:20",  "userName":"test1(test1)",  "nodeName":"申请退料",  "alertTime":"",  "overTime":"2014-08-05 15:27:20",  "remark":null,  "isAlertTime":"0",  "delegateTaskList":[{"toUserName":"test2(test2)",  "changeTime":"2014-08-05 10:28:33",  "fromUserName":"test2(test2)",  "taskRelation":"2"},  {"toUserName":"test3(test3)",  "changeTime":"2014-08-05 10:28:33",  "fromUserName":"test3(test3)",  "taskRelation":"1"}],  "claimTime":"2014-08-05 10:28:33",  "isOverTime":"0",  "durationNode":"5小时",  "isContainHoliday":0,  "advanceSend":0,  "overtimeSend":0,  "assigneeName":"test1(test1)"}  ]  } | | | | |
| NodeTaskInfo字段说明 | | | | | |
| duration | String | 耗时 | | | |
| endTime | String | 任务结束时间 | | | |
| startTime | String | 任务开始时间 | | | |
| userName | String | 处理人 | | | |
| nodeName | String | 节点名称 | | | |
| alertTime | String | 预警时间点 | | | |
| overTime | String | 超时时间点 | | | |
| remark | String | 备注 | | | |
| isAlertTime | String | 是否到达预警时间点 | | | |
| delegateTaskList | List<NodeTaskRelation> | 委托/转办关系记录 | | | |
| claimTime | String | 任务签收时间点 | | | |
| isOverTime | String | 是否到达超时时间点 | | | |
| durationNode | String | 任务处理工时 | | | |
| isContainHoliday | int | 节假日策略  0不考虑节假日/作息时间  1考虑节假日,不考虑作息时间  2考虑节假日/作息时间 | | | |
| advanceSend | int | 预警提示返回结果  0短信/邮件预警未发送  1短信/邮件预警发送成功  2短信预警发送成功  3短信预警发送失败  4邮件预警发送成功  5邮件预警发送失败  6短信预警发送成功,邮件预警发送失败  7短信预警发送失败,邮件预警发送成功  8短信、邮件预警失败 | | | |
| overtimeSend | int | 超时提醒返回结果  0短信/邮件超时未发送  1短信/邮件超时发送成功  2短信超时发送成功  3短信超时发送失败  4邮件超时发送成功  5邮件超时发送失败  6短信超时发送成功,邮件超时发送失败  7短信超时发送失败,邮件超时发送成功  8短信、邮件超时失败 | | | |
| assigneeName | String | 签收人 | | | |
| NodeTaskRelation字段说明 | | | | | |
| taskRelation | String | 1转办关系 2 委托关系 | | | |
| fromUserName | String | 转办/委托人姓名 | | | |
| toUserName | String | 被 转办/委托人姓名 | | | |
| changeTime | String | 转办/委托 时间 | | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  DealInfoResponse hessianResult = hassianService.getInstanceDealInfo(""); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  DealInfoResponse cxfResult = cxfService.getInstanceDealInfo(""); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /getInstanceDealInfo.freepage?instancesId=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

# 任务

## 签收任务

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| ResultResponse signTask(String taskId, String userId) |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/signTask.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| taskId | 是 | String | 任务id |
| userId | 是 | String | 签收人id |

返回结果

| 服务方式 | 返回结果 | | | |
| --- | --- | --- | --- | --- |
| Hessian/  webservice | ResultResponse对象 | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |
| HTTP请求 | JSON对象：  {"resultCode":"1","resultMess":"签收任务操作成功"} | | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  ResultResponse hessianResult = hassianService.signTask("",""); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  ResultResponse cxfResult = cxfService.signTask("",""); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /signTask.freepage?taskId=&userId=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

## 通过任务

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| ResultResponse completeTask(HashMap<String, Object> taskMap,  List<HashMap<String, Object>> nodeInfoList, List<HashMap<String, Object>> variableList) |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/completeTask.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | Key | Value类型 | 任务基本信息描述 |
| --- | --- | --- | --- | --- | --- |
| taskMap | 否 | HashMap | createUser | String | 委托人id |
| 否 | entrustUser | String | 被委托人id |
| 否 | completeReason | String | 完成任务原因 |
| 是 | currentUser | String | 当前用户id |
| 是 | taskId | String | 任务id |
| 是 | processKey | String | 流程key |
| 是 | instancesId | String | 流程实例id |
| 是 | taskState | String | 任务状态1未签收 2已签收 |
| 否 | taskKey | String | 任务key |
| nodeInfoList | 否 | List<HashMap> | Key | Value类型 | 节点配置描述 |
| nodeKey | String | 任务节点key |
| nodeUserIds | String | 任务节点可处理人id |
| nodeGroupIds | String | 任务节点可处理组id |
| isMulti | String | 串/并多实例切换  1串行多实例  2并行多实例 |
| variableList | 否 | List<HashMap> | Key | Value类型 | 参数描述 |
| paramName | String | 参数英文名 |
| paramValue | String | 参数值 |

返回结果

| 服务方式 | 返回结果 | | | |
| --- | --- | --- | --- | --- |
| Hessian/  webservice | ResultResponse对象 | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |
| HTTP请求 | JSON对象：  {"resultCode":"1","resultMess":"通过任务操作成功"} | | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  ResultResponse hessianResult = hassianService.completeTask("","",""); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  ResultResponse cxfResult = cxfService.completeTask("","",""); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /completeTask.freepage?taskMap=&nodeInfoList=&variableList=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

## 驳回任务

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| ResultResponse rejectToPreTask(String taskId, String rejectedReason, **int** rejectedType, List<HashMap<String, Object>> nodeInfoList, List<HashMap<String, Object>> variableList) |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/rejectToPreTask.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 | | |
| --- | --- | --- | --- | --- | --- |
| taskId | 是 | String | 任务id | | |
| rejectedReason | 否 | String | 驳回理由 | | |
| rejectedType | 是 | int | 驳回类型0 上一个任务对应的节点 1当前节点的上一个节点 | | |
| nodeInfoList | 否 | List<HashMap> | Key | Value类型 | 节点配置描述 |
| nodeKey | String | 任务节点key |
| nodeUserIds | String | 任务节点可处理人id |
| nodeGroupIds | String | 任务节点可处理组id |
| isMulti | String | 串/并多实例切换  1串行多实例  2并行多实例 |
| variableList | 否 | List<HashMap> | Key | Value类型 | 参数描述 |
| paramName | String | 参数英文名 |
| paramValue | String | 参数值 |

返回结果

| 服务方式 | 返回结果 | | | |
| --- | --- | --- | --- | --- |
| Hessian/  webservice | ResultResponse对象 | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |
| HTTP请求 | JSON对象：  {"resultCode":"1","resultMess":"驳回任务操作成功"} | | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  ResultResponse hessianResult = hassianService.rejectToPreTask("","",1,null,null); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  ResultResponse cxfResult = cxfService.rejectToPreTask("","",1,null,null); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /rejectToPreTask.freepage?taskId=&rejectedReason=&rejectedType=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

## 转办任务

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| ResultResponse delegateTask(String taskId, String fromUserId, String toUserId,  String instancesId, String processKey) |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/delegateTask.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| taskId | 是 | String | 任务id |
| fromUserId | 是 | String | 当前处理人的id |
| toUserId | 是 | String | 转办人的id |
| instancesId | 是 | String | 流程实例id |
| processKey | 是 | String | 流程key |

返回结果

| 服务方式 | 返回结果 | | | |
| --- | --- | --- | --- | --- |
| Hessian/  webservice | ResultResponse对象 | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |
| HTTP请求 | JSON对象：  {"resultCode":"1","resultMess":"转办任务操作成功"} | | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  ResultResponse hessianResult = hassianService.delegateTask("","","","",""); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  ResultResponse cxfResult = cxfService.delegateTask("","","","",""); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /delegateTask.freepage?taskId=&fromUserId=&toUserId=&instancesId=  &processKey=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

## 废弃任务

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| ResultResponse discardTask (String instancesId, String deleteReason) |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/discardTask.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| instancesId | 是 | String | 实例id |
| deleteReason | 是 | String | 废弃原因 |

返回结果

| 服务方式 | 返回结果 | | | |
| --- | --- | --- | --- | --- |
| Hessian/  webservice | ResultResponse对象 | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |
| HTTP请求 | JSON对象：  {"resultCode":"1","resultMess":"废弃任务操作成功"} | | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  ResultResponse hessianResult = hassianService.discardTask("",""); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  ResultResponse cxfResult = cxfService.discardTask("",""); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /discardTask.freepage?instancesId=&deleteReason=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

## 撤销任务

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| ResultResponse cancelTask(String instancesId, String processKey, String cancelReason) |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/cancelTask.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| instancesId | 是 | String | 实例id |
| processKey | 是 | String | 流程key |
| cancelReason | 是 | String | 撤销原因 |

返回结果

| 服务方式 | 返回结果 | | | |
| --- | --- | --- | --- | --- |
| Hessian/  webservice | ResultResponse对象 | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |
| HTTP请求 | JSON对象：  {"resultCode":"1","resultMess":"撤销任务操作成功"} | | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  ResultResponse hessianResult = hassianService.cancelTask("","",""); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  ResultResponse cxfResult = cxfService.cancelTask("","",""); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /cancelTask.freepage?instancesId=&processKey=&cancelReason=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

## 获取待办任务总记录数

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| DataResponse countTaskNum(String pernr, String sysId) |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/countTaskNum.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| pernr | 是 | String | 实例id |
| sysId | 否 | String | 应用编号 |

返回结果

| 服务方式 | 返回结果 | | | |
| --- | --- | --- | --- | --- |
| Hessian/  webservice | DataResponse对象 | 字段 | 类型 | 描述 |
| resultCode | String | 返回代码 |
| resultMess | String | 返回信息 |
| resultData | String | 返回结果数据 |
| HTTP请求 | JSON对象：  {"resultCode":"1","resultMess":"任务代办数统计操作成功","resultData":"1"} | | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  DataResponse hessianResult = hassianService.countTaskNum("",""); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  DataResponse cxfResult = cxfService.countTaskNum("",""); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /countTaskNum.freepage?pernr=&sysId=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

## 获取待办任务数据

接口路径

|  |
| --- |
| com.sany.workflow.webservice.action.WorkflowService |

接口方法

|  |
| --- |
| NoHandTaskResponse getNoHandleTask (String pernr, String sysId) |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| hessian | 工程上下文/hessian?service=workflowService |
| webservice | 工程上下文/cxfservices/workflowService |
| HTTP请求 | 工程上下文/workflow/webservice/getNoHandleTask.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| pernr | 是 | String | 实例id |
| sysId | 否 | String | 应用编号 |

返回结果

| 服务方式 | 返回结果 | | | | |
| --- | --- | --- | --- | --- | --- |
| Hessian/  webservice | NoHandTaskResponse  对象 | 字段 | | 类型 | 描述 |
| resultCode | | String | 返回代码 |
| resultMess | | String | 返回信息 |
| dataList | | List<NoHandTaskInfo> | 返回结果数据 |
| HTTP请求 | JSON对象：  {"resultCode":"1",  "resultMess":"获取任务待办数据操作成功",  "dataList":[{"taskType":0,  "url":"http://localhost:8080/SanyPDP/workflow/taskManage/  toDealTask.page?processKey=Mms.return  &processInstId=2f9b0603-190f-11e4-9627-4437e6999a31  &taskId=2f9fc0fd-190f-11e4-9627-4437e6999a31&taskState=1  &suspensionState=1&sysid=pdp",  "title":"物料退料流程",  "userAccount":"admin",  "createTime":"2014-08-04 15:59:34",  "sender":"系统管理员",  "taskId":"2f9fc0fd-190f-11e4-9627-4437e6999a31",  "businessKey":"测试",  "suspensionState":"1",  "processKey":"Mms.return",  "taskDefKey":"usertask1",  "taskState":"1",  "fromUser":null,  "fromUserName":null,  "appUrl":"http://localhost:8080/SanyPDP",  "instanceId":"2f9b0603-190f-11e4-9627-4437e6999a31",  "dealButtionName":"处理",  "defId":"Mms.return:1:eeffae14-1888-11e4-8a1d-4437e6999a31"}]  } | | | | |
| NoHandTaskInfo字段说明 | | | | | |
| taskType | int | | 任务类型0自己任务1委托任务2转办任务 | | |
| url | String | | 跳转到处理页面链接地址 | | |
| title | String | | 流程标题 | | |
| userAccount | String | | 发起人账号 | | |
| createTime | Long | | 任务创建时间 | | |
| sender | String | | 发起人姓名 | | |
| taskId | String | | 任务id | | |
| businessKey | String | | 业务主题 | | |
| suspensionState | String | | 流程状态1运行中2 挂起 | | |
| processKey | String | | 流程key | | |
| taskDefKey | String | | 任务key | | |
| taskState | String | | 任务状态1未签收2已签收 | | |
| fromUser | String | | 转办/委托人 | | |
| fromUserName | String | | 转办/委托人姓名 | | |
| appUrl | String | | 应用上下文 | | |
| instanceId | String | | 流程实例id | | |
| dealButtionName | String | | 按钮名称 | | |
| defId | String | | 流程定义id | | |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| hessian | HessianProxyFactory factory = **new** HessianProxyFactory();  String url = "http://localhost:8080/SanyPDP/hessian?service=workflowService";  WorkflowService hassianService = (WorkflowService) factory.create(  WorkflowService.**class**, url);  DataResponse hessianResult = hassianService.getNoHandleTask("",""); |
| webservice | String cxfUrl = "http://localhost:8080/SanyPDP/cxfservices/workflowService";  JaxWsProxyFactoryBean WSServiceClientFactory = **new** JaxWsProxyFactoryBean();  WSServiceClientFactory.setAddress(cxfUrl);  WSServiceClientFactory.setServiceClass(WorkflowService.**class**);  WorkflowService cxfService = (WorkflowService) WSServiceClientFactory.create();  DataResponse cxfResult = cxfService.getNoHandleTask("",""); |
| HTTP请求 | String url = "http://localhost:8080/SanyPDP/workflow/webservice  /getNoHandleTask.freepage?pernr=&sysId=";  String httpResult = org.frameworkset.spi.remote.http.HttpReqeust  .*httpPostforString*(url);  注意：  如果是后台采用HTTP请求调用服务时，参数含有中文需要转码  例如：java.net.URLEncoder.encode("HTTP测试", "UTF-8");  如果是通过前台浏览器直接请求服务，则不需要转码 |

## 获取任务追踪图片

接口路径

|  |
| --- |
| 无 |

接口方法

|  |
| --- |
| 无 |

服务发布方式

| 发布方式 | 发布地址 |
| --- | --- |
| HTTP请求 | 工程上下文/workflow/webservice/getProccessActivePic.freepage?+参数 |

方法参数说明

| 参数 | 是否必须 | 类型 | 描述 |
| --- | --- | --- | --- |
| instancesId | 是 | String | 流程实例id |
| response | 是 | HttpServletResponse | 请求响应 |

实例

| 服务方式 | 调用代码 |
| --- | --- |
| HTTP请求 | <img src="http://localhost:8080/SanyPDP/workflow/webservice/ getProccessActivePic.freepage?instancesId=" /> |