

技术分享

目录

[1. 表格与表格之间的联动 - 2 -](#_Toc52351585)

[1.1 功能设计说明 - 2 -](#_Toc52351586)

[1.2 功能设计部署说明 - 2 -](#_Toc52351587)

[1.2.1 定义搜索表格 - 2 -](#_Toc52351588)

[1.2.2 定义联动表格 - 5 -](#_Toc52351589)

[1.2.3 表格之间的联动效果展示 - 8 -](#_Toc52351590)

[2. XWORKS框架Template定义 - 10 -](#_Toc52351591)

[2.1 WorkflowTemplateDef - 10 -](#_Toc52351592)

# 表格与表格之间的联动

## 功能设计说明

PLM中很多对象搜索页面需要对客制化的表格进行功能定制，比如多个表格之间的联动、因此针对这一块的需求统一设计后实现，以便后续项目开发使用。

## 功能设计部署说明

### 定义搜索表格

###### 定义以编码和名称为条件的搜索界面

<div class="x-box" id="contentArea">

<div class="x-box-tl">

<div class=x-box-tr>

<div class=x-box-tc></div>

</div>

</div>

<div class="x-box-bwrap">

<div class="x-box-ml">

<div class="x-box-mr">

<div class="x-box-mc" >

<div class="x-box-body">

<div id="wiz\_step\_contents">

<fieldset class="x-fieldset x-form-label-left"

id="Visualization\_and\_Attributes" style="margin:10px 5px 0px 10px;">

<legend>查询条件</legend>

<table width="100%">

<tr>

<td width="80px"><b>编号:</b></td>

<td>

<w:textBox name="resourceNumber" required="false" id="resourceNumber" maxlength="100" size="30"/>

</td>

</tr>

<tr>

<td width="80px"><b>名称:</b></td>

<td>

<w:textBox name="resourceName" required="false" id="resourceName" maxlength="100" size="30"/>

</td>

</tr>

</table>

</fieldset>

<div id="displayButton" style="margin:10px 5px 15px 10px;">

<table class="x-toolbar-ct" cellspacing="0">

<tbody>

<tr>

<td class="x-toolbar-left" align="right">

<input type="button" name="search" id="mpmResJobSkillSearch" value=" 搜索 " onClick="searchData();"/>

</td>

</tr>

</tbody>

</table>

<jsp:include page="${mvc:getComponentURL('cn.inovance.mpml.resource.mvc.builder.MpmResCutterResourceTableBuilder')}" />

<jsp:include page="${mvc:getComponentURL('cn.inovance.mpml.resource.mvc.builder.MpmResourceCompatibilityTypeTableBuilder')}" />

<jsp:include page="${mvc:getComponentURL('cn.inovance.mpml.resource.mvc.builder.MpmResourceCompatibilityTableBuilder')}" />

</div>

</div>

</div>

</div>

</div>

</div>

<div class="x-box-bl x-panel-nofooter" >

<div class=x-box-br>

<div class=x-box-bc></div>

</div>

</div>

</div>

</div>

###### 编写搜索script

<script language=JavaScript>

//搜索符合的数据

function searchData(){

var resourceNumber = document.getElementById("resourceNumber").value;

var resourceName = document.getElementById("resourceName").value;

var params ={

resourceNumber:resourceNumber,

resourceName:resourceName,

};

PTC.jca.table.Utils.reload('cn.inovance.mpml.resource.mvc.builder.MpmResCutterResourceTableBuilder', params, true);

showTable();

}

document.onkeydown = function(e){

if(!e) e = window.event;

if((e.keyCode || e.which) == 13){

document.getElementById("mpmResJobSkillSearch").click();

}

}

$2 = function(id) {

var r = $(id);

if(!r) {

var ar = Ext.DomQuery.select("[name='"+id+"']");

if(ar.length > 0) r = ar[0]; //get the first element

}

return r;

}

</script>

###### 注册action

<model name=*"ppbPRToolBar"*>

<action name=*"allocateCustMpmResCutter"* type=*"ppb"*>

<ribbonInfo buttonType=*"menuitem"* buttonScale=*"small"* />

</action>

</model>

<objecttype name="ppb" resourceBundle="cn.inovance.mpml.resource.resource.mpmCustomActionsRB">

<!--刀具-->

<action name="allocateCustMpmResCutter" ajax="component" >

<command class="com.ptc.windchill.enterprise.part.psb.client.action.InsertExistingPartsJCAAction" method="execute" url="/ptc1/inovance/mpml/resource/addMpmResCutterResourceRowsWizard" windowType="popup"/>

<supportedTypes>

<type value="com.ptc.windchill.mpml.processplan.operation.MPMOperation"/>

</supportedTypes>

</action>

</objecttype>



### 定义联动表格

1. 创建表格显示

<jsp:include page=*"*${mvc:getComponentURL('cn.inovance.mpml.resource.mvc.builder.MpmResCutterResourceTableBuilder')}*"* />

<jsp:include page=*"*${mvc:getComponentURL('cn.inovance.mpml.resource.mvc.builder.MpmResourceCompatibilityTypeTableBuilder')}*"* />

<jsp:include page=*"*${mvc:getComponentURL('cn.inovance.mpml.resource.mvc.builder.MpmResourceCompatibilityTableBuilder')}*"* />

1. 编写单选按钮点击事件

<script language=JavaScript>

function getFormElements(ele\_name){

var result = [];

var allElements = document.forms.mainform.elements;

var allElementsLength = allElements.length;

for (i = 0; i < allElementsLength; i++) {

var formElement = allElements[i];

var eleName=formElement.name;

if(formElement.name!=null){

if (eleName == ele\_name){

result.push(formElement);

}

}

}

return result;

}

function changeNameFiledValue(){

var selectItems = PTC.jca.table.Utils.getTableSelectedRowsById("cn.inovance.mpml.resource.mvc.builder.MpmResCutterResourceTableBuilder");

var slength =selectItems.length;

for(var i = 0; i<slength;i++){

var selectedOid = selectItems[i].value;

//alert(selectedOid);

var record = PTC.jca.table.Utils.getTableRowFromOid("cn.inovance.mpml.resource.mvc.builder.MpmResCutterResourceTableBuilder",selectedOid);

var resourceNumber = record.data.number;

//var resourceName = record.data.name;

var params ={

resourceNumber:resourceNumber,

//resourceName:resourceName,

};

PTC.jca.table.Utils.reload('cn.inovance.mpml.resource.mvc.builder.MpmResourceCompatibilityTableBuilder', params, true);

PTC.jca.table.Utils.reload('cn.inovance.mpml.resource.mvc.builder.MpmResourceCompatibilityTypeTableBuilder', params, true);

}

}

function addChangeListener(){

var elements=getFormElements("radio\_cn.inovance.mpml.resource.mvc.builder.MpmResCutterResourceTableBuilder");

if(elements && elements.length > 0){

var elementsLength = elements.length;

for (i = 0; i < elementsLength; i++) {

var formElement = elements[i];

if(formElement){

Event.observe(formElement,'click', changeNameFiledValue);

}

}

}

setTimeout(addChangeListener,1000);

}

PTC.onReady(function() {

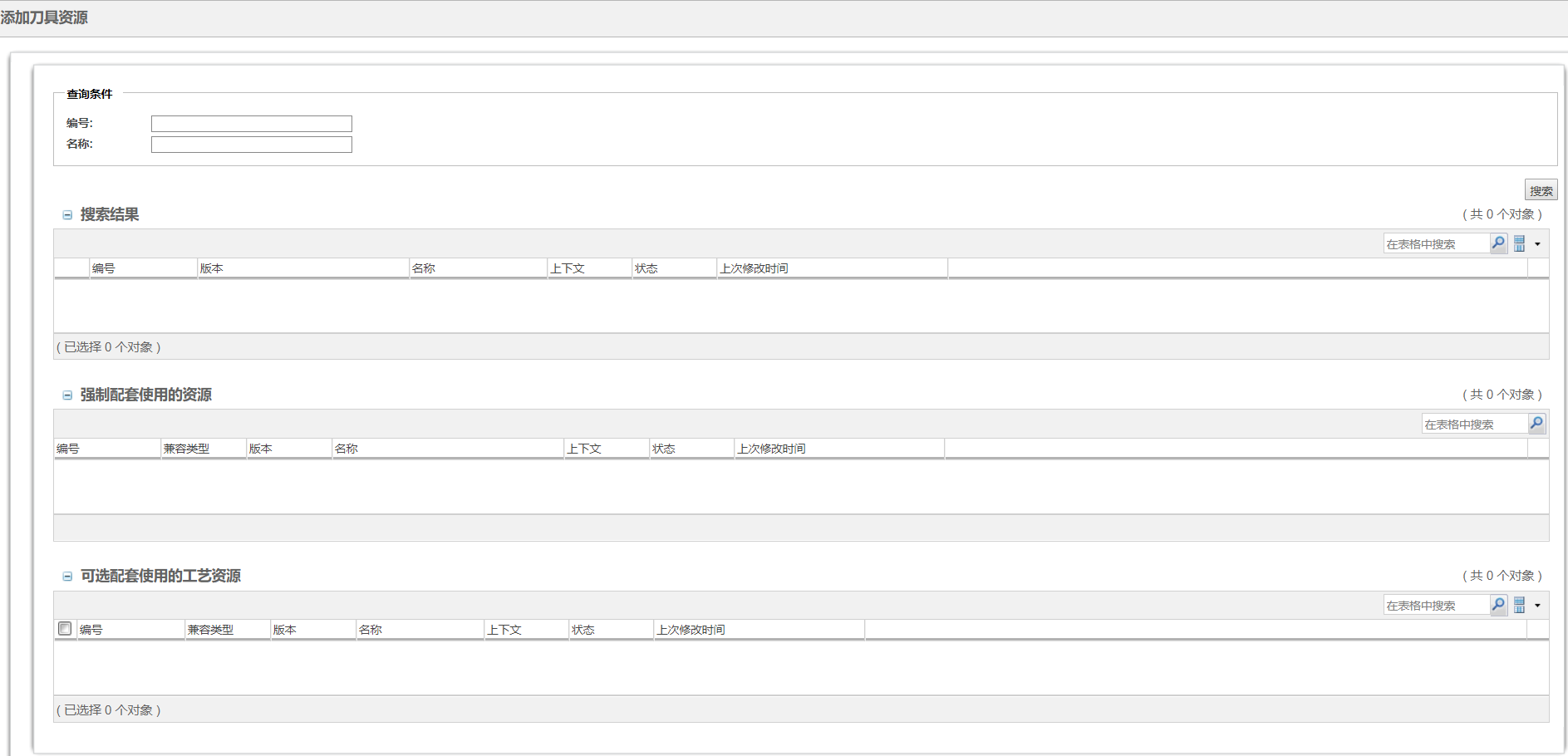
addChangeListener();

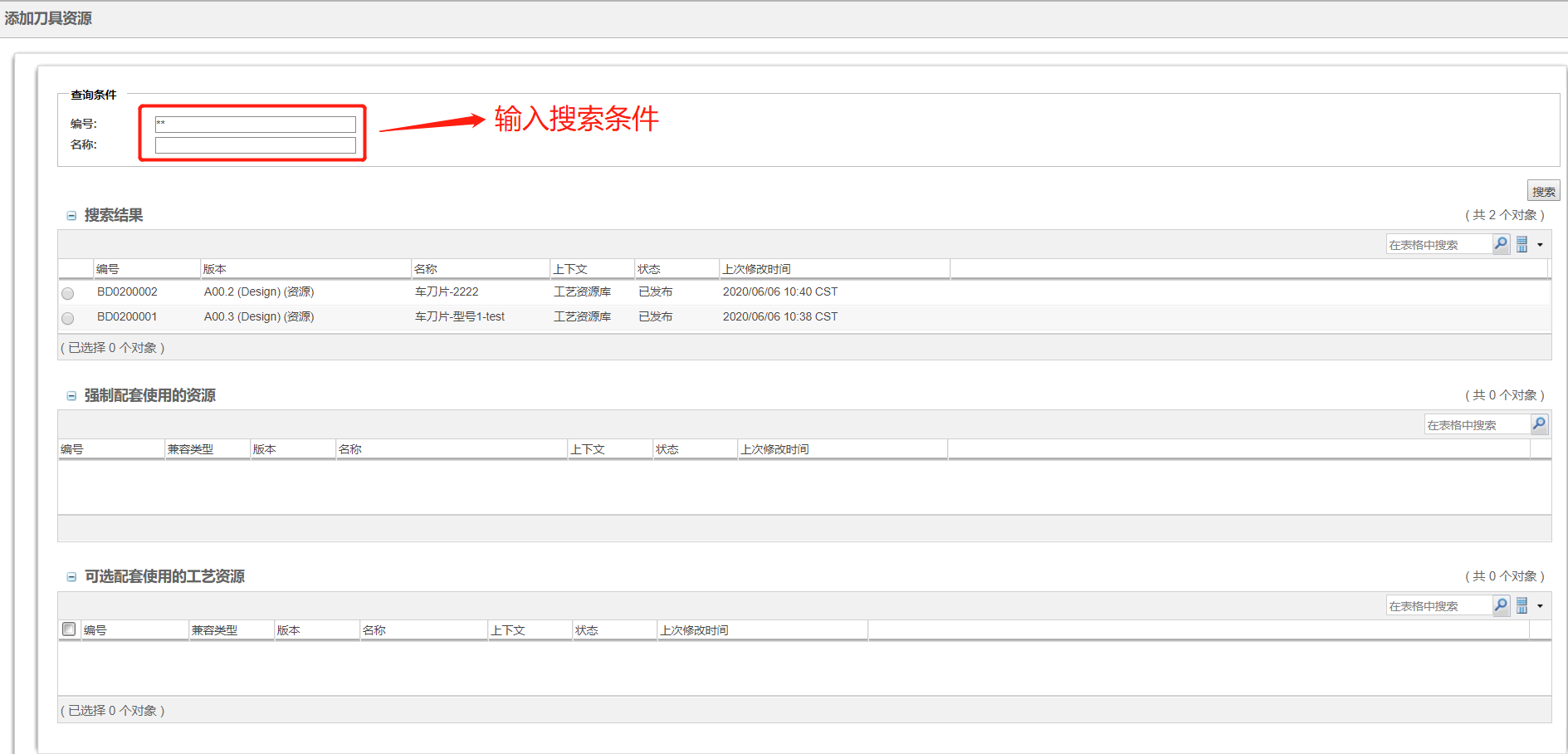
});

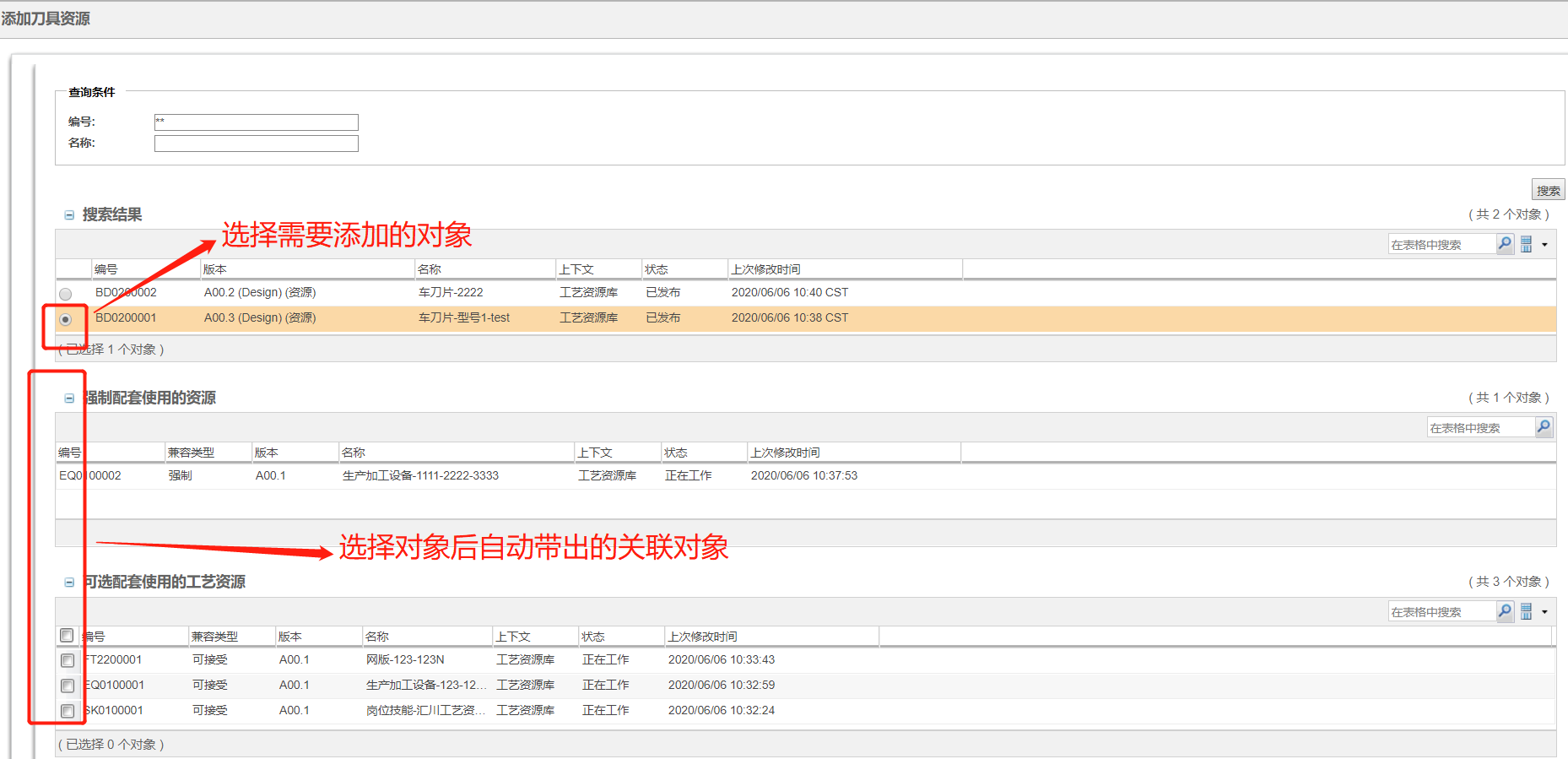
</script>



### 表格之间的联动效果展示







描述：1）如图1为表格显示界面

2）如图2填写搜索条件点击搜索按钮得到搜索结果为原对象

3）如图3选择搜索结果中的对象联动显示下面两个表格的对象

# XWORKS框架Template定义

## WorkflowTemplateDef

@WorkflowTemplateDef(templateId = ProcessConst.***INOVANCE\_ProcessArchiving\_ID***, templateName = "INOVANCE\_产品工艺归档流程", description = "产品工艺归档流程",

attachmentGroupDefs = {

@AttachmentGroupDef(

groupId = "attachment01",

name = @ResourceBundleKey(resourceBundle = ProcessConst.***RESOURCE***, key = ProcessResource.***attachment\_group1\_description***),

contentRoleType = "WORKFLOW\_ATTACHMENT")

},

relatedObjectGroupDefs = {

@RelatedObjectGroupDef(

groupId = "relatedObjectGroup01",

name = @ResourceBundleKey(

resourceBundle = ProcessConst.***RESOURCE***,

key = ProcessResource.***relatedObject\_group1\_description***

),

supportedTypes = {@SupportedType(type = "cn.inovance.Inovance\_ProduceProcess", includeSubType = **false**)},//设置添加对象类型

validators = CurrentLoopDocRelatedObjectGroupMemberValidator.**class**,//添加校验

tableBuilderConfig = @TableBuilderConfig(actionCommandConfigs = {

@ActionCommandConfig(actionName = "addProductTechnologyRows", commandClass = ProductTechnologyAddRowsProcessor.**class**),

@ActionCommandConfig(actionName = "deleteProductTechnologyRows", commandClass = ProductTechnologyDeleteRowsProcessor.**class**),

},

actionModelResolver = ProductTechnologyTableActionModelResolver.**class**,

configurableTableCreator = ProductTechnologyTableBuilder.**class**,

tableConfigBuilder = ProductTechnologyTableBuilder.**class**),

annotationAttributes = {

@WebFormAttribute(name = "annotation1",

label = @ResourceBundleKey(resourceBundle = DocumentConst.***INOVANCE\_CUSTRESOURCEBUNDLE***, key = CustProResource.***custDistributionOfProfessionals\_label***),

dataType = String.**class**, createGui=GuiComponentType.***COMBOBOX***, viewGui=GuiComponentType.***TEXT***,

editGui=GuiComponentType.***COMBOBOX***,

enumerationProvider = @EnumerationProvider(provierType = EnumerationProviderType.***STRING\_LIST***, value=": |软件:软件|硬件:硬件|结构:结构")),

@WebFormAttribute(name = "annotation2",

label = @ResourceBundleKey(resourceBundle = DocumentConst.***INOVANCE\_CUSTRESOURCEBUNDLE***, key = CustProResource.***modifyDesc\_label***),

dataType = String.**class**, createGui=GuiComponentType.***TEXTAREA***, viewGui=GuiComponentType.***TEXT***,

editGui=GuiComponentType.***TEXTAREA***, textSize = 30, maxLength = 500, textRows = 2)

},

showAnnotationColumn = **true**,

eventAdapter = com.ptc.xworks.promotion.builders.PromotionNoticeRelatedObjectGroupEventAdapter.**class**,

objectSearchingConfig = @ObjectSearchingConfig(baseWhereClause ="(state.state='INV\_WAIT\_IMPL')|(state.state='INV\_IN\_CHANGE')")//设置添加对象状态

),

@RelatedObjectGroupDef(

groupId = "relatedObjectGroup02",

name = @ResourceBundleKey(

resourceBundle = ProcessConst.***RESOURCE***,

key = ProcessResource.***relatedObject\_group2\_description***

),

supportedTypes = {@SupportedType(type = "cn.inovance.OperationDocument", includeSubType = **false**),

@SupportedType(type = "cn.inovance.DensoProgramDoc", includeSubType = **false**),

@SupportedType(type = "cn.inovance.ProcessProductionDoc", includeSubType = **false**)},

tableBuilderConfig = @TableBuilderConfig(actionCommandConfigs = {

@ActionCommandConfig(actionName = "addProcessDocumentationRows", commandClass = ProcessDocumentationAddRowsProcessor.**class**),

@ActionCommandConfig(actionName = "deleteProcessDocumentationRows", commandClass = ProcessDocumentationDeleteRowsProcessor.**class**),

},

actionModelResolver = ProcessDocumentationTableActionModelResolver.**class**,

configurableTableCreator = DefaultRelatedObjectTableConfigBuilder.**class**,

tableConfigBuilder = DefaultRelatedObjectTableConfigBuilder.**class**),

eventAdapter = com.ptc.xworks.promotion.builders.PromotionNoticeRelatedObjectGroupEventAdapter.**class**,

objectSearchingConfig = @ObjectSearchingConfig(baseWhereClause ="(state.state='INWORK')")//设置添加对象状态

)

},

processPhaseDefs = {

@ProcessPhaseDef(

name = ProcessConst.***INOVANCE\_ProcessArchiving\_submit***,

description = @ResourceBundleKey(resourceBundle = ProcessConst.***RESOURCE***, key = ProcessResource.***INOVANCE\_ProcessArchiving\_submit***)

),

@ProcessPhaseDef(

name = ProcessConst.***INOVANCE\_ProcessArchiving\_update***,

description = @ResourceBundleKey(resourceBundle = ProcessConst.***RESOURCE***, key = ProcessResource.***INOVANCE\_ProcessArchiving\_update***)

),

@ProcessPhaseDef(

name = ProcessConst.***INOVANCE\_ProcessArchiving\_huiQian***,

description = @ResourceBundleKey(resourceBundle = ProcessConst.***RESOURCE***, key = ProcessResource.***INOVANCE\_ProcessArchiving\_huiQian***)

),

@ProcessPhaseDef(

name = ProcessConst.***INOVANCE\_ProcessArchiving\_piZhun***,

description = @ResourceBundleKey(resourceBundle = ProcessConst.***RESOURCE***, key = ProcessResource.***INOVANCE\_ProcessArchiving\_piZhun***)

)

}

)

* 申请单流程模板的相关配置

|  |  |  |
| --- | --- | --- |
| **模块** | **字段** | **描述** |
| WorkflowTemplateDef | templateId | 流程模板的ID,配置在Windchill流程模板的描述字段中,使用这样的格式<!--//TEMPLATE\_ID=<template id>-->,用来唯一标示一个流程.必须为字符数字或下划线; |
| templateName | Windchill流程模板的名称 |
| description | 对该流程的描述,仅用来增强配置的可读性,没有实际作用 |
| attachmentGroupDefs | AttachmentGroupDef | 流程所定义的相关附件组的定义 |
| relatedObjectGroupDefs | RelatedObjectGroupDef | 流程所定义的相关对象组的定义 |
| processPhaseDefs | ProcessPhaseDef | 流程阶段的定义,用于在流程任务界面显示一个进度条 |
|  |  |  |

* 表格和附件批注申明

|  |  |  |
| --- | --- | --- |
| **模块** | **字段** | **描述** |
| AttachmentGroupDef | groupId | 在一个流程范围内,相关附件组的唯一标示 |
| name | 相关附件在界面上显示名称时所使用的ResourceBundle |
| contentRoleType | 附件分组默认的上传的Role,默认为SECONDARY |
| RelatedObjectGroupDef | groupId | 相关对象组的唯一标示 |
| name | 相关对象组在界面上显示时,所使用的ResourceBundle |
| supportedTypes | 用户在查询相关对象时,允许选择的对象类型 |
| validators | 通过UI项相关对象分组中添加或粘贴对象时,用于检查对象是否符合要求所使用的RelatedObjectGroupMemberValidator.默认的配置将使用RelatedObjectTypeValidator,用来验证对象的类型是否符合相关对象分组的要求. |
| tableBuilderConfig | 配置用于构造相关对象分组表格的TableConfigBuilder的信息 |
| annotationAttributes | 用于配置在相关对象分组的批注的属性(RelatedObjectGroupMember的annotation1~annotation10),控制在编辑批注界面批注属性的界面呈现方式 |
| objectSearchingConfig | 配置该相关对象分组的搜索界面的一些控制参数，如设置添加对象状态 |
|  |  |  |

* 表格

|  |  |  |
| --- | --- | --- |
| **模块** | **字段** | **描述** |
| tableBuilderConfig | actionCommandConfigs | 配置用于构造相关对象分组表格的TableConfigBuilder的信息 |
| actionModelResolver | 配置决定关联对象分组的工具条的ActionModel的class |
| configurableTableCreator | 配置显示相关对象的表格时所使用的ConfigurableTableCreator的具体实现类 |
| tableConfigBuilder | 配置显示相关对象的表格时所使用的TableConfigBuilder的具体实现类 |
| annotationAttributes | WebFormAttribute | 使用此Annotation来对XmlObject对象的属性进行标注,描述该属性在UI上使用何种UI控件进行显示、编辑; 同时用来配置UI控件的各种属性、取值约束等信息 |

* 表格按钮定义

|  |  |  |
| --- | --- | --- |
| **模块** | **字段** | **描述** |
| ActionCommandConfig | actionName | Action的名称 |
| commandClass | Action逻辑的具体实现类 |
|  |  |  |

@RobotNodeDef(nodeId = ProcessConst.INOVANCE\_INNITROLE, expressions = {ProcessArchivingRobotExpressions.class})

/\*\*

\* 提交产品工艺

\*/

@ActivityNodeDef(

nodeId = ProcessConst.INOVANCE\_ProcessArchiving\_submit,

name = @ResourceBundleKey,

phaseName = ProcessConst.INOVANCE\_ProcessArchiving\_submit,

description = "提交产品工艺",

enableApplicationForm = true,

applicationFormJspPath = "/netmarkets/jsp/inovance/mpml/plan/ProcessArchivingFrom.jsp",

attachmentGroupTab = false,

relatedObjectTab = false,

permissionToRelatedObject = {

@PermissionToRelatedObject(

groupId = "relatedObjectGroup01",

role = ProcessConst.SUBMITTER,

permissions = {AccessPermissionEnum.READ, AccessPermissionEnum.DOWNLOAD}),

@PermissionToRelatedObject(

groupId = "relatedObjectGroup02",

role = ProcessConst.SUBMITTER,

permissions = {AccessPermissionEnum.READ, AccessPermissionEnum.DOWNLOAD})

},

permissionToCollectedObject = {

@PermissionToCollectedObject(permissionTargetObjectCollector = GetECNObjectsAsPermissionTarget.class, permissions = {

AccessPermissionEnum.READ, AccessPermissionEnum.DOWNLOAD }, role = PermissionToCollectedObject.ANY\_ROLE),

@PermissionToCollectedObject(

permissionTargetObjectCollector = GetTeamInstanceAsPermissionTarget.class,

permissions = {AccessPermissionEnum.READ, AccessPermissionEnum.MODIFY},

role = PermissionToCollectedObject.ANY\_ROLE)

},

operationToRelatedObjectGroup = {

@AllowedOperation(

groupId = "relatedObjectGroup01",

role = ProcessConst.SUBMITTER,

allowedOperations = {UserOperation.VIEW}),

@AllowedOperation(

groupId = "relatedObjectGroup02",

role = ProcessConst.SUBMITTER,

allowedOperations = {UserOperation.VIEW})

},

transitionTriggers = {

@TransitionTriggerConfig(transition = ProcessArchivingTransitionTrigger.TRANSITION\_SUBMIT, transitionTriggers = {ProcessArchivingTransitionTrigger.class}),

@TransitionTriggerConfig(transition = ProcessArchivingTransitionTrigger.TRANSITION\_CANCEL, transitionTriggers = {ProcessArchivingTransitionTrigger.class})

},

resourcePoolConfigs = {

@ResourcePoolConfig(role = ProcessConst.INOVANCE\_COUNTERSIGN\_PERSONNEL, option = ResourcePoolOption.ALL\_USER),

@ResourcePoolConfig(role = ProcessConst.APPROVER, option = ResourcePoolOption.ALL\_USER)

}

)

String INOVANCE\_ProcessArchiving\_submit = "INOVANCE\_ProcessArchiving\_submit"; // 该常量仅用于承载Annotation,并没有任何实际用途

* 注册任务表达式节点

|  |  |  |
| --- | --- | --- |
| **模块** | **描述** | **备注** |
| @RobotNodeDef | 注册表达式节点 | 表达式ID唯一 |
| @ActivityNodeDef | 注册任务节点 | 节点ID唯一 |

* 节点描述

|  |  |  |
| --- | --- | --- |
| **模块** | **字段** | **描述** |
| RobotNodeDef | nodeId | 一个流程模板范围内,自动机节点的唯一标示符,这个标示符配置在Windchill流程模板中 描述中,使用这样的格式<!--//NODE\_ID=<nodeid of activity node>-->必须为字母数字或下划线; |
| expressions | 配置自动机需要执行的RobotExpressions |
| ActivityNodeDef | nodeId | 一个流程模板范围内,人工活动节点的唯一标示符,这个标示符配置在Windchill流程模板中 描述中,使用这样的格式<!--//NODE\_ID=<nodeid of activity node>-->必须为字母数字或下划线; |
| name | 流程人工节点的名称 |
| phaseName | 该活动节点所属的阶段的名称 |
| description | 对该流程节点的描述,仅用来增强配置的可读性,没有实际作用 |
| enableApplicationForm | 是否在流程任务详细页面嵌入一个表单 |
| applicationFormJspPath | 流程任务详细页面内嵌的表单页面的JSP路径 |
| attachmentGroupTab | 在该流程节点是否显示附件页签, 默认为false |
| relatedObjectTab | 在该流程节点是否显示相关对象Tab, 默认为false |
| permissionToRelatedObject | 在该流程节点,当WorkItem创建时,自动给对应的角色的人员,指定的相关对象组中的所有对象,赋予指定的权限, 当完成WorkItem时,自动将先前赋予的权限回收 |
| permissionToCollectedObject | 在该流程节点,当WorkItem创建时,自动给对应的角色的人员,指定的PermissionToCollectedObject中配置的PermissionTargetObjectCollector实现类中返回的所有对象, 赋予指定的权限, 当完成WorkItem时,自动将先前赋予的权限回收 |
| operationToRelatedObjectGroup | 在该流程节点,用来定义某个角色的用户,允许用户对相关对象组进行哪些操作 |
| transitionTriggers | 任务路由完成校验 |
| resourcePoolConfigs | 配置任务角色访问表格权限 |