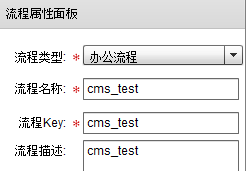
## Bpmx3整合CMS文章审批流程

# 文章审批流程设置

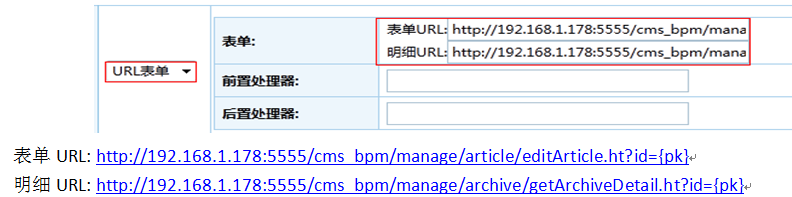
创建文章审批流程，在Bpmx3系统的【流程管理】🡪【流程定义管理】新建文章审批流程cms\_test，并设置其流程Key为cms\_test:



为了cms系统处理审批结果，应在结束节点前添加一个webservice节点，其设置将在下面的1.2中做说明。

##### 1.1设置流程表单

在Bpmx3系统【流程定义管理】🡪【设置】🡪【表单设置】中设置URL表单，如图：



表单URL：编辑文章页面的URL地址。

其代码如下：

@RequestMapping(value="/editArticle",method = {RequestMethod.*GET*,RequestMethod.*POST*})

**public** ModelAndView approveArticle(HttpServletRequest request,HttpServletResponse response) **throws** Exception

{

ModelAndView result = **new** ModelAndView();

**try**{

**long** articleId = RequestUtil.*getLong*(request, "id",0);

**long** taskId = RequestUtil.*getLong*(request, "taskId",0);

**long** archivesId = articleDao.getById(articleId).getArchiveid();

**if**(archivesId==0){

System.*out*.println("参数 arhcivesId 错误 "+archivesId);

}

Column column = columnDao.getByArchiveId(archivesId);

result.addObject("columnId", column.getColumnid());

result.addObject("column", column);

Archive chive =archiveDao.getById(archivesId);

result.addObject("chive", chive);

result.addObject("taskId", taskId);

result.addObject("status", 1);

**long** siteId = siteColumnDao.getByColumnId(column.getColumnid()).getSiteid();

com.hotent.cms.model.site.Site site=siteDao.getById(siteId);

SiteHelper.*setCurrentlySite*(request, response, siteId);

result.addObject("filterwords", site.getFilterwords());

List<com.hotent.cms.model.archive.RelateArchive> relateList=relateArchiveDao.getByArchiveId(chive.getArchivesId());

result.addObject("relateList",relateList);

String copyToClumnNames=columnArchiveDao.getClumnNamesByArchiveTitle(siteId, chive.getTitle(),chive.getArchivesId());

result.addObject("copyToClumnNames",copyToClumnNames);

Map<Long, ArchivesCustom> archivesCustomMap = archivesCustomDao.getMapByArchivesId(archivesId);

Map<String, List<ColumnCustom>> columnCustomMap = columnCustomDao.getCategoryMapByArchivesId(archivesId);

Map<Long, ColumnCustomDefine> columnCustomDefineMap = columnCustomDefineDao.getAllMap();

result.addObject("archivesCustomMap", archivesCustomMap);

result.addObject("columnCustomMap", columnCustomMap);

result.addObject("columnCustomDefineMap", columnCustomDefineMap);

result.addObject("imageList", imageDao.getByArchiveId(archivesId));

result.addObject("ticle", articleDao.getByArciveId(archivesId));

}**catch** (Exception e) {

// **TODO**: handle exception

e.printStackTrace();

log.error("准务修改文章失败："+e.getMessage());

}

result.setViewName("/manage/archive/article/updArticle.jsp");

**return** result;

}

明细URL：文章明细页面的URL地址。

具体代码如下：

@RequestMapping(value="/getArchiveDetail",method = {RequestMethod.*GET*,RequestMethod.*POST*})

**public** ModelAndView getArchiveDetail(HttpServletRequest request,HttpServletResponse response)

{

ModelAndView result = **new** ModelAndView();

**try**{

**long** archivesId = RequestUtil.*getLong*(request, "archivesId",0);

**if**(archivesId==0){

archivesId = RequestUtil.*getLong*(request, "id",0);

}

**if**(archivesId==0){

System.*out*.println("参数 arhcivesId 错误 "+archivesId);

}

Archive chive =archiveDao.getById(archivesId);

result.addObject("chive", chive);

List<RelateArchive> relateList=relateArchiveDao.getByArchiveId(chive.getArchivesId());

result.addObject("relateList",relateList);

Map<Long, ArchivesCustom> archivesCustomMap = archivesCustomDao.getMapByArchivesId(archivesId);

Map<String, List<ColumnCustom>> columnCustomMap = columnCustomDao.getCategoryMapByArchivesId(archivesId);

result.addObject("archivesCustomMap", archivesCustomMap);

result.addObject("columnCustomMap", columnCustomMap);

String copyToClumnNames=columnArchiveDao.getClumnNamesByArchiveTitle(SiteHelper.*getCurrentlySite*(request,response), chive.getTitle(),chive.getArchivesId());

result.addObject("copyToClumnNames",copyToClumnNames);

result.setViewName(*showArticle*);

result.addObject("imageList", imageDao.getByArchiveId(archivesId));

result.addObject("ticle", articleDao.getByArciveId(archivesId));

result.addObject("status",RequestUtil.*getLong*(request, "id",0));

}**catch** (Exception e) {

// **TODO**: handle exception

e.printStackTrace();

log.error(e.getMessage());

}

**return** result;

}

##### 1.2 设置webservice节点

###### 1.2.1 cms系统发布webservice方法处理审批结果

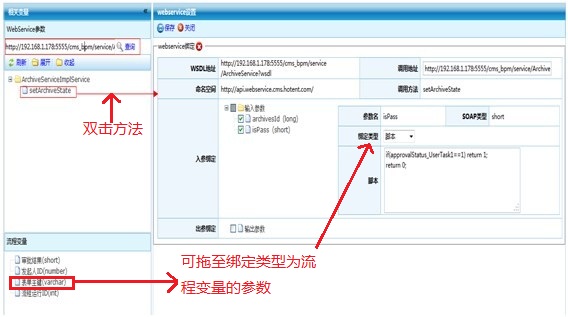
在cms系统com.hotent.cms.webservice.api.ArchiveService这个webservice接口及其实现类com.hotent.cms.webservice.impl.ArchiveServiceImpl中添加如下处理审批结果方法：

|  |
| --- |
| @Override  @WebMethod(operationName="setArchiveState")  **public** **void** setArchiveState(Long archivesId,Short isPass) {  // **TODO** Auto-generated method stub  **try** {  Archive archive = archiveDao.getById(archivesId);  **if**(isPass==1){  archive.setState(Archive.*STATE\_2*);  archive.setPublish(Archive.*publish\_0*);  }**else**{  archive.setState(Archive.*STATE\_3*);  archive.setPublish(Archive.*publish\_1*);  }  archiveDao.update(archive);    } **catch** (Exception e) {  // **TODO** Auto-generated catch block  e.printStackTrace();  } |

###### 设置webservice节点参数



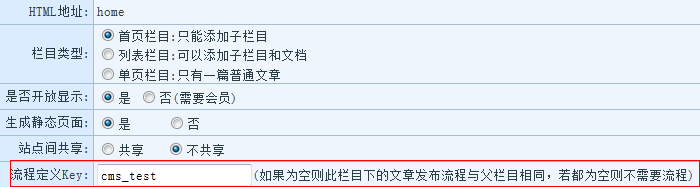
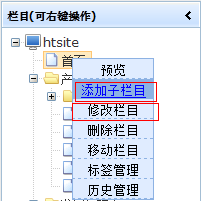
在Bpmx3系统【流程定义管理】🡪【设置】🡪【节点设置】中在webservice节点上点击webService设置：



将webservice绑定了方法并设置好参数后流程经过此webservice节点时将会调用绑定的方法。

##### 1.3设置栏目审批流程定义Key

在cms系统中【栏目管理】🡪【网站栏目管理】的【添加子栏目】和【修改栏目】可设置流程定义key



将第一步中设置的流程key：cms\_test设置到需要审批文章的栏目下。

# 编辑文章启动审批流程

在cms系统中【文档管理】🡪 【信息管理】 中启动审批流程



处于未审批状态的文章可以单击【审批】启动审批流程

处于正在审批或者以完成审批状态的文章可单击【审批明细】查看审批流程明细,如下图：



其中启动审批流程的具体代码如下：

/\*\*

\* 启动流程

\* **@param** columnid

\* **@param** article

\* **@param** archive

\* **@throws** Exception

\*/

**private** **void** startProcess(**long** columnid,Article article,Archive archive) **throws** Exception {

String defKey = columnDao.getDefKeyByColumId(columnid);

StartCmd startCmd=**new** StartCmd();

startCmd.setFlowKey(defKey);

startCmd.setAccount(ContextUtil.*getCurrentUser*().getAccount());

startCmd.setBusinessKey(Long.*toString*(article.getArticleid()));

//通过webservice获取启动流程的xml

StartResult startResult = ProcessService.*startFlow*(startCmd);

archive.setProcessRunId(Long.*valueOf*(startResult.getRunId()));

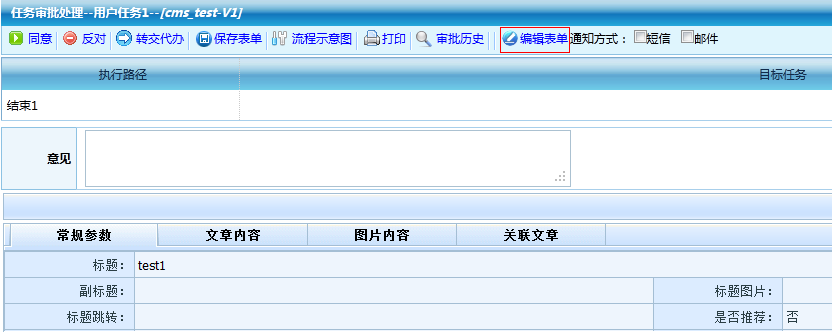
archive.setState(Archive.*STATE\_4*);

archiveDao.update(archive);

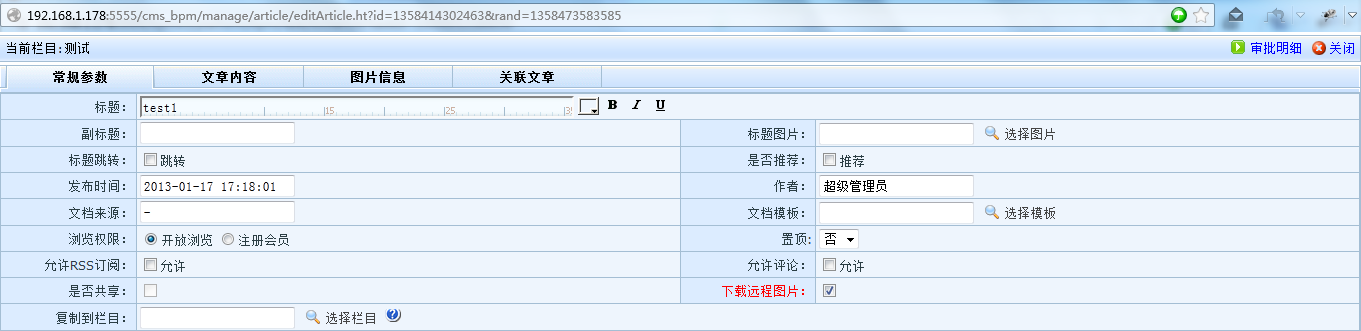
}

# 审批文章

在Bpmx3 系统中【个人办公】🡪【我的代办事项】中，点击【主办】，进入审批流程页面：



【编辑表单】按钮可进入文章编辑页面：



当整个流程完成的时候会调用步骤一中设置的流程结束脚本事件，对cms中的文章审批状态进行更新，完成文章的审批。

其中执行下一个节点的具体代码如下：

/\*\*

\*

\* **@param** archive

\* **@param** request

\* **@throws** Exception

\*/

**private** **void** nextProcess(Archive archive,HttpServletRequest request) **throws** Exception {

NextCmd nextCmd=**new** NextCmd();

String taskId = RequestUtil.*getString*(request, "taskId");

nextCmd.setTaskId(taskId);

nextCmd.setAccount("admin");

**int** status = RequestUtil.*getInt*(request, "status");

String state =Archive.*STATE\_4*;

nextCmd.setVoteAgree(String.*valueOf*(status));

**switch**(status){

**case** 1://审批通过

state = Archive.*STATE\_2*;

**break**;

**case** 2://审批不通过

state = Archive.*STATE\_3*;

**break**;

**case** 3://驳回发起人

nextCmd.setTaskType(TaskType.*BACKTOSTART*);

**break**;

}

List<BpmNode> bpmNode = ProcessService.*getTaskOutNodes*(taskId);

ProcessService.*doNext*(nextCmd);

**if**(bpmNode.size()==1){

archive.setState(state);

archive.setPublish(Archive.*publish\_0*);

archiveDao.update(archive);

}

}