



机关事业单位养老保险管理信息系统

环境配置及系统开发



机关事业单位养老保险管理信息系统构架组



目录



环境配置



系统开发



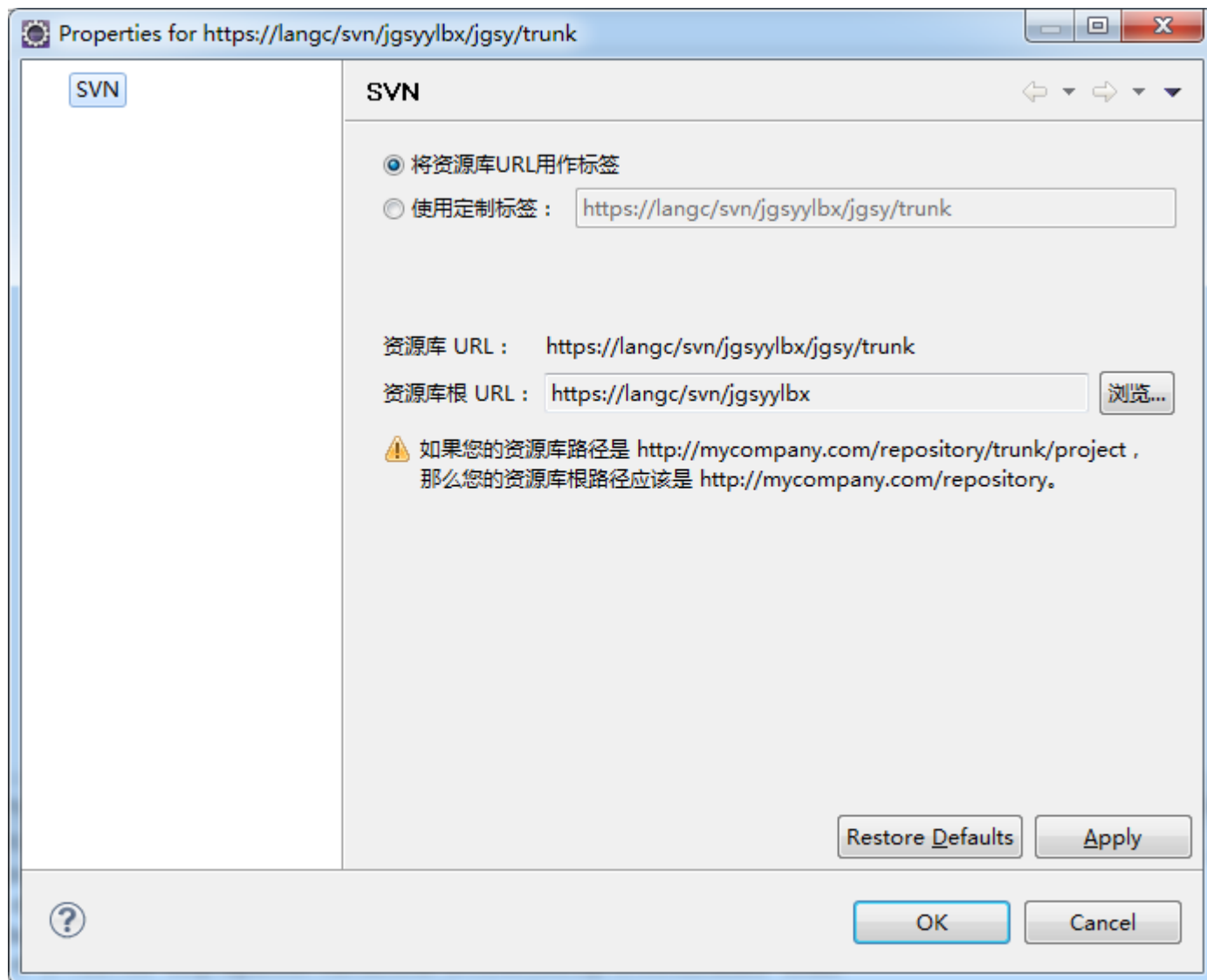
开发环境及开发工具



❖ 开发环境及工具

- JAVA IDE: Eclipse 3.5 及更高版本
- JDK 1.6.0 及更高版本
- 中间件: Tomcat 5.5 及更高版本
- 数据库: Oracle 10G 及更高版本
- 版本控制: svn





目录



环境配置



系统开发



系统开发

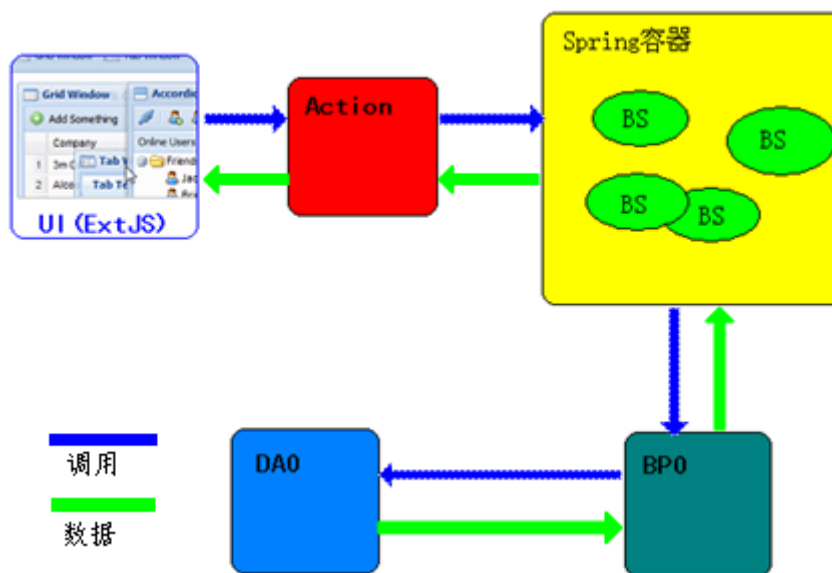


❖ Brick-spot框架介绍

❖ 框架应用及开发



架构分层 (brick-spot)





框架介绍



❖ 框架组成部分

- ExtJS 3.0
- Struts 1.3
- Spring 2.5
- Hibernate 3.6



spot



- ❖ **ExtJS**是一套后台无关的基于浏览器技术(JavaScript, HTML, DOM, CSS)的**富客户端AJAX框架**, 包括Web常用的UI组件和炫丽的页面效果, 提供了丰富的用户体验, 使用简单, **API**丰富, 近年在Web开发领域非常流行。
- ❖ 为提高工作效率, 根据业务劳动社保主要业务需要, 基于ExtJS, 我们研发了一套更加实用的标签“**Spot**”



Brick 优点



- ❖ **Brick**框架中BS不需配置，框架自动搜索BS实现并加入Spring容器实现事务支持，BS方法名要符合Spring项目统一的事务支持命名规范。
- ❖ **Brick**框架对数据库访问采用Spring对**Hibernate API**和**JDBC API**的扩展，又对其分别进行封装，简化了Hibernate和JDBC访问数据库操作。
 - `ApplicationContext.getJdbcTemplate()`
 - `ApplicationContext.getHibernateTemplate()`



基于Hibernate方式数据库访问



方法摘要

int	bulkUpdate (String queryString, Object[] paramValues, org.hibernate.type.Type[] types) 执行批量更新
int	bulkUpdate (String queryString, String[] paramNames, Object[] paramValues) 执行批量更新
int	bulkUpdate (String queryString, String[] paramNames, Object[] paramValues, org.hibernate.type.Type[] paramTypes) 执行批量更新
void	deleteAll (Object[] objs) 一次删除多个实体
List<?>	executeSQLQuery (HQLInfo<?> info) 执行SQL查询
List<?>	executeSQLQuery (String sql) 执行SQL查询
<T> List<T>	executeSQLQuery (String sql, Class<T> entity) 执行SQL语句
int	executeUpdate (HQLInfo<?> hi) 执行更新
int	executeUpdate (String updateString, Object[] paramValues)
int	executeUpdate (String updateString, String[] paramNames, Object[] paramValues, org.hibernate.type.Type[] paramTypes) 执行更新
List<?>	find (HQLInfo<?> hqlinfo) 执行查询



基于JDBC方式的数据库访问



方法摘要

<u>Object</u>	<u>callProcedure</u> (<u>String</u> callString) 执行存储过程
<u>Object</u>	<u>callProcedure</u> (<u>String</u> callString, <u>ICallableStatementProcessor</u> csp) 执行存储过程
<u>Object</u>	<u>callProcedure</u> (<u>String</u> callString, <u>ICallableStatementProcessor</u> csp, <u>ICallableResultProcessor</u> crp) 执行存储过程
<u>Object</u>	<u>executePreparedStatement</u> (<u>String</u> sql) 执行预编译的sql语句
<u>Object</u>	<u>executePreparedStatement</u> (<u>String</u> sql, <u>IPreparedStatementProcessor</u> psp) 执行预编译的sql语句
<u>Object</u>	<u>executePreparedStatement</u> (<u>String</u> sql, <u>IPreparedStatementProcessor</u> psp, <u>IpreparedResultProcessor</u> prp) 执行预编译的sql语句
<u>Integer</u>	<u>getCount</u> (<u>HQLInfo</u> <?> hqlInfo) 获得查询的总记录数
<u>Integer</u>	<u>getCount</u> (<u>String</u> SQL) 获得查询的总记录数
<u>Date</u>	<u>getDBDate</u> () 获得数据库日期 (仅用于Oracle)
<u>Timestamp</u>	<u>getDBTimestamp</u> () 获得数据库 timestamp (仅用于Oracle)
<u>Long</u>	<u>getSequence</u> (<u>String</u> sequenceName) 获得Long值sequence (仅用于Oracle)
<T> T	<u>getSequence</u> (<u>String</u> sequenceName, <u>Class</u> <T> type) 获得sequence (仅用于Oracle)



事务控制



- ❖ 根据配置由BS方法名称自动提供事务支持
- ❖ 编程方式控制事务

```
TransUtil.trans(new ICommand<ResultType>(){  
    public ResultType execute() {  
        // TODO 数据库操作;  
        return result;  
    }  
});
```



系统开发



❖ 1、数据库及事务控制配置

❖ 2、系统开发

- 后台业务开发BS、BSImp、BPO、DAO
- Action开发
- 前台UI开发

❖ 3、license更新



数据库及事务控制配置



❖ 配置文件路径 `src/conf/context-persistence.xml`

- 数据库配置
- 事务控制配置

```
<!-- jdbc数据源配置 -->
<bean id="defaultDataSource" class="org.apache.commons.dbcp.BasicDataSource"
    destroy-method="close" autowire-candidate="true" primary="true">
    <property name="driverClassName" value="oracle.jdbc.driver.OracleDriver" />
    <property name="url" value="jdbc:oracle:thin:@192.168.1.156:1521:orcl" />
    <property name="username" value="test" />
    <property name="password" value="test" />
</bean>
```




系统开发——业务处理



- 业务处理开发主要包含四部分
 - BS 开发
 - BSImp 开发
 - BPO 开发
 - DAO 开发

```
package com.lbs.lrs.la.apps.agent.bp;  
  
import org.springframework.stereotype.Service;  
  
import com.lbs.brick.sys.ServiceBase;  
  
@Service("TestBS")  
public class TestBSImp extends ServiceBase implements TestBS {  
  
}
```




业务处理及数据库操作



❖ 常用数据库操作方法：

- `hibernate.save(entity);`
- `hibernate.update(entity);`
- `hibernate.delete(entity);`
- `hibernate.executeUpdate(sql);`
- `hibernate.executeUpdate(sql, new para[]);`
- `jdbc.query(sql, new BeanPropertyRowMapper(Bean.class));`
- `.....`



分页查询——Action层



- ❖ 执行跟request和response相关的数据交互，包括从前台读取数据和返回数据到前台。
- ❖ 连接BS和UI

```
public class TestAction extends BizAction {
    //BS 业务处理
    private TestBS bs = (TestBS) getService(TestBS.class);

    public ActionForward query(final ActionMapping mapping,
        final ActionForm actionForm, final HttpServletRequest request,
        final HttpServletResponse response) throws Exception {
        final ActionForward af = mapping.findForward("query");

        try {
            HQLInfo hi = bs.query();
            hqlPageQuery(hi, request);
        } catch (Exception e) {
            saveErrors(request, e, "查询出错");
        }
        return queryResponse(af, request, response);
    }
}
```



分页查询



❖ Brick分页查询写法

- `sqlPageQuery(sql, request);`
- `hqlPageQuery(hql, request);`



Action异常处理及数据返回机制



```
try {  
    //.....  
    //.....  
    saveSuccessfulMsg(request, "成功");  
} catch (Exception e) {  
    saveErrors(request, e, "失败");  
}  
return ajaxResponse(request, response);
```

将业务操作的成功失败标示及业务数据传递给前台UI（Spot）。

saveSuccessfulMsg(request,"成功");	//返回成功标识
saveErrors(request,"失败");	//返回失败标识
saveResponseData(request, map);	//返回业务数据



Action 返回 (Return)



❖ 普通返回

- `return mapping.findForward(“xxx”);`

❖ 分页查询返回

- `return queryResponse(Actionforward, request, response);`

❖ AJAX返回

- `return ajaxResponse(request, response);`





分页查询——Action配置



❖ 配置Struts-config文件

- FormBean配置
- Action配置
- Forward配置

❖ 在web.xml中配置Struts-config, 使其生效

```
<action path="/testAction"
        type="com.xxx.xxx.TestAction"
        name="LrUserForm"
        parameter="method"
        scope="request">
    <forward name="query" path="/pages/lr/lruser/userQuery.jsp"></forward>
    <forward name="add" path="/pages/lr/lruser/lrUserAdd.jsp"></forward>
    <forward name="modify" path="/pages/lr/lruser/lrUserModify.jsp"></forward>
</action>
```




分页查询——JSP页面



```
<%@ page contentType="text/html; charset=UTF-8"%>
<%@ taglib uri="http://brick.bjlbs.com.cn/tags-spot" prefix="s"%>
<html>
  <s:head title="申请登记">
    <s:onready>
      <s:viewport layout="border" >
        <s:grid title="组庭管理" id="Grid1" tbar="tbar" pageSize="12" region="center" viewConfig="{forceFit:true}"
          rowNumber="true" autoLoad="true" remoteSort="true" sm="sm"
          url="/courtAction.do?method=query">
          <s:textcolumn name="aez030" header="组庭ID" hidden="true"/>
          <s:textcolumn name="aez010" header="申请登记编号" hidden="true"/>
          <s:codecolumn name="aeb078" header="庭审类型" editable="false"/>
          <s:textcolumn name="aeb080" header="开庭日期" editable="false" align="center"/>
          <s:textcolumn name="aeb082" header="首席仲裁员姓名" editable="false" />
          <s:textcolumn name="aeb085" header="书记员姓名" editable="false"/>
          <s:codecolumn name="aee002" header="审批状态" editable="false"/>
        </s:grid>
        <s:form id="approveverify" hidden="true" url="/courtAction.do">
          <s:hidden name="aez010" id="caseid"/>
        </s:form>
      </s:viewport>
    </s:onready>
```



JSP必要标签



❖ JSP头

- `<%@ page contentType="text/html; charset=UTF-8"%>`

❖ 标签库

- `<%@ taglib uri="http://brick.bjlbs.com.cn/tags-spot" prefix="s"%>`

❖ Title、JS、CSS设置及引入

- `<s:head title="测试页面" />`

❖ OnReady

- `<s:onready><s:viewport layout="border" >`
- `</s:viewport></s:onready>`





分页查询标签



❖ Grid标签

- tbar
中的toolbar控制
- Sm
中选择模式及监听
- Textcolumn
//文本列
- Codecolumn
//代码列

//grid

//grid

//文本列

//代码列

```
<s:grid title="标题" id="Grid1" tbar="tbar"  
  pageSize="12" region="center"  
  viewConfig="{forceFit:true}"  
  rowNum="true" autoLoad="true" remoteSort="true" sm="sm"  
    url="/courtAction.do?method=query">  
    <s:textcolumn name="aez030" header="组庭ID" hidden="true"/>  
    <s:textcolumn name="aez010" header="申请登记编号"/>  
    <s:codecolumn name="aee002" header="审批状态" editable="false"/>  
</s:grid>
```



Grid中Tbar属性



❖ Tbar属性

- Text
- id
- Icon
- handler



```
<script type="text/javascript">
var tbar = [{
    text : '组庭',
    icon : jsapp.ctxPath+'/res/images/icons/table/add.gif',
    handler : function(){
        var sr = getGridSelected('Grid1');
        var callback = function(resObj){
            if(resObj.success){
                var srurl = '/courtAction.do?method=loadAdd&aez010='+Ext.getCmp('caseid').value;
                popWin('组庭信息',srurl,'Grid1',800,450);
            }else{
                Ext.Msg.alert('错误提示',resObj.msg);
            }
        }
        ajaxRequest('/la/approveAction.do?method=verify&aez010='+Ext.getCmp('caseid').value,callback);
    }
}, {
```



Grid中sm属性



❖ sm属性

- singleSelect
- listeners



```
<script type="text/javascript">
var disabled = false;
var sm = new Ext.grid.RowSelectionModel( {
    singleSelect : true,
    listeners : {
        'selectionchange' : function(sm){
            Ext.getCmp('deleteBtn').setDisabled(disabled||sm.getCount() < 1);
            Ext.getCmp('updateBtn').setDisabled(disabled||sm.getCount() < 1);
            Ext.getCmp('printBtn').setDisabled(disabled||sm.getCount() < 1);
        }
    }
});
</script>
```



查询条件



查询条件

basicfield

显示查询条件

additionfield

隐藏查询条件

```
<s:query region="north" tbar="tbar" >
  <s:basicfield>
    <s:text name="aeb003" fieldLabel="申请人"/>
    <s:text name="aeb004" fieldLabel="被申请人" />
  </s:basicfield>
  <s:additionfield>
    <s:fieldgroup>
      <s:date name="aeb008Begin" fieldLabel="登记日期 从"/>
      <s:date name="aeb009Begin" fieldLabel="申请日期 从"/>
    </s:fieldgroup>
    <s:fieldgroup>
      <s:date name="aeb008End" fieldLabel="至"/>
      <s:date name="aeb009End" fieldLabel="至"/>
    </s:fieldgroup>
  </s:additionfield>
</s:query>
```



增改删

```
<s:onready>
  <s:viewport layout="fit">
    <s:rowform id="form1" url="/courtAction.do?method=add"
      baseCls="x-plain">
      <s:hidden name="aez010" id="aez010" />
      <s:fieldgroup>
        <s:combobox name="aeb078" fieldLabel="庭审类型" allowBlank="false" />
        <s:date name="aeb080" fieldLabel="开庭日期" allowBlank="false" />
        <s:text name="aeb082" fieldLabel="首席仲裁员姓名" allowBlank="false" />
      </s:fieldgroup>
      <s:fieldgroup>
        <s:text name="aeb085" fieldLabel="书记员姓名" />
        <s:text name="aeb083" fieldLabel="仲裁员1姓名" />
        <s:text name="aeb084" fieldLabel="仲裁员2姓名" />
      </s:fieldgroup>
      <s:textarea name="aeb079" fieldLabel="开庭地址"
        itemCls="ext-form-item-1-per" anchor="98%" />
      <s:textarea name="aeb081" fieldLabel="通知事项"
        itemCls="ext-form-item-1-per" anchor="98%" />
      <s:toolbar>
        <s:submit handler="afterSubmitCloseWindow" />
        <s:reset />
      </s:toolbar>
    </s:rowform>
  </s:viewport>
```


标签



❖ 常用标签元素：

- `<s:combobox name="aeb078" fieldLabel="类型" allowBlank="false" />`
- `<s:date name="aeb080" fieldLabel="日期" allowBlank="false" />`
- `<s:text name="aeb082" fieldLabel="文本" allowBlank="false" />`
- `<s:textarea name="aeb079" fieldLabel="文本域" itemCls="ext-form-item-1-per" anchor="98%" />`





Spot按钮



❖ Toolbar

- Button
- Submit
- Reset

自定义按钮

提交按钮

重置按钮

```
<s:toolbar>  
  <s:submit handler="afterSubmitCloseWindow" />  
  <s:reset />  
  <s:button icon="dddd" text="名字" id="deletid" handler="function"/>  
</s:toolbar>
```



Spot按钮



❖ Button中的handler属性

- 定义点击按钮后事件
- 定义响应后处理事件



```
<script type="text/javascript">
    function afterSubmitCloseWindow() {
        submitForm('form1', '/caseRegAction.do?method=add', function(rdata) {tbarActive(rdata);});
    }
var tbarActive = function(rdata) {
    //根据返回的结果集进行操作
    if(rdata.data.casedisabled == "1") {
        closeWindow();
    }
}
```




AJAX请求响应数据



❖ resObj



- `resObj.success`
Boolean
- `resObj.failure`
- `resObj.data.xxx`

获取返回成功标识

获取返回错误标识

获取返回数据



license更新



❖ 将新的license文件（`license.lbs`）更新到src目录下。

```
WARN : 2010-05-25 17:02:07,580 : LicenseManagerImpl[line:?]:  
      License(brick|过期时间: 2010-12-20|IP地址: any) 授权给: 利博赛研发部  
WARN : 2010-05-25 17:02:07,580 : LicenseManagerImpl[line:?]:  
      您当前使用的是非零售版brick License, 请在正式使用本产品前更新为零售版License!  
WARN : 2010-05-25 17:02:07,611 : LicenseManagerImpl[line:?]:  
      License(spot|过期时间: 2010-12-20|IP地址: any) 授权给: 利博赛研发部  
WARN : 2010-05-25 17:02:07,611 : LicenseManagerImpl[line:?]:  
      您当前使用的是非零售版spot License, 请在正式使用本产品前更新为零售版License!  
.....
```



参考资料



- ❖ **Ext 3.0 - API Documentation**
 - ❖ **Hibernate 3.0 API DOC**
 - ❖ **Spring Framework API 2.5**
 - ❖ **Brick API DOC**
 - ❖ **Spot tag DOC**
-



请多提宝贵意见!