

# 基于 Struts 的数据校验框架的应用研究

孙 麒<sup>1,2</sup>, 郑 宁<sup>1</sup>, 周志宇<sup>2</sup>

(1. 杭州电子工业学院 计算机应用分院, 浙江 杭州 310037; 2. 浙江工程学院 信息电子学院, 浙江 杭州 310033)

**摘要:** 随着网络技术的飞速发展, 数据的有效性校验变得更加重要。基于 Struts 的数据校验框架是一种通过配置外部 XML 文件, 是一种利用可重用的校验规则和正则表达式匹配规则的校验机制。分析了实现这个框架的关键技术和工作流程, 提出了在这个框架中如何扩充数据校验的方法, 并且实现了把这种基于 Struts 框架移植到一般的 Java 系统的方案。

**关键词:** 数据校验; Struts; Plug-In 机制; XML; Resources Bundle

## Study of validator framework based on struts

SUN Qi<sup>1,2</sup>, ZHENG Ning<sup>1</sup>, ZHOU Zhi-yu<sup>2</sup>

(1. School of Computer Application Science, Hangzhou Institute of Electronics Engineering, Hangzhou 310037, China; 2. School of Information Science and Electronic Engineering, Zhejiang Institute of Science and Technology, Hangzhou 310033, China)

**Abstract:** With the development of the internet technologies, validation is more important. The validator framework based on struts is a mechanism configuring outside XML files with the reusable validation regulations and matching regulations of regular expression. Some associated technologies and the workflow of the framework are analyzed, and a method of extending the validation ways is put forward. And a scheme of validating outside of struts was represented.

**Key words:** validation; struts; plug-In; XML; resources bundle

### 1 引言

随着网络上电子商务、网上考试、电子现金、数字货币等新兴业务的兴起, 网络数据的合法性校验问题变得越来越重要。这就使我们不得不面临一个问题, 要对这么多的各种各样的从网页或图形界面接收的用户输入数据进行严格的有效性校验, 然而大多数处理有效性校验的方法不仅烦琐, 不可重用, 业务逻辑、显示逻辑与数据混合在一起, 维护困难, 而且每一次校验代码的改动都需要进行重新编译, 而 Struts 数据校验框架是只要通过配置外部的 xml 文件, 增加和修改校验方法不需要改动代码的一种新的数据校验手段。本文分析了基于 Struts 的数据校验框架, 讨论了怎样扩充数据校验方法和把这种框架移植到一般的 Java 系统的思想。

### 2 Struts 技术

Struts 技术是 Apache 基金会 Jakarta 项目组提供的一个服务器 Java 开放框架, 它采用了模型-视图-控制器 2 (Model-View-Controller2, MVC2) 模式, 能够很好地帮助 Java 开发者利用 J2EE 开发 Web 应用。它的出现, 使得在 Java 和 XML 开发应用软件时, 真正实现了用户界面和事物逻辑的分离

以及动态生成网页的逻辑部分和 XML 的分离。Struts 由一组相互协作的类(主件)、Servlet、JSP taglib 等构成。其中相互协作的类用于 XML 的解析及 Bean 的属性设定等; Servlet 用于 HTTP 请求的分配及 JSP 的调用; JSP taglib 用于页面的动态生成。Struts 有 3 个主要的类: ActionServlet、Action、ActionForm, 都包括在核心包 org.apache.struts.action 中, 它们之间的调用关系如图 1 所示。

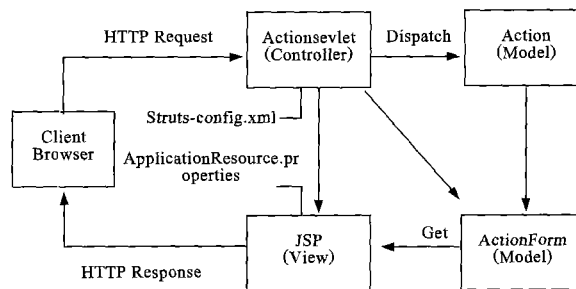


图 1 Struts 工作原理图

### 3 数据校验框架

#### 3.1 概述

数据校验框架 validator 是 Jakarta 通用包的一个部分,

收稿日期: 2003-08-11。

**作者简介:** 孙麒 (1977-), 女, 浙江诸暨人, 硕士研究生, 助教, 研究方向为电子商务和数据挖掘; 郑宁, 男, 教授, 硕士研究生导师, 研究方向为 ICCAD、电信网管和电子商务; 周志宇 (1974-), 男, 浙江诸暨人, 硕士研究生, 研究方向为计算机视觉。

是基于 Struts 技术、通过配置外部 XML 文件以引用校验规则和匹配正则表达式为基础的一种校验机制,可使用相同的校验规则同时实现服务器端和客户端的数据校验。它很好地利用了 XML 文件可以包括语法描述以及使应用程序可以对文件进行结构确认的特点,是一种对于整个 Web 校验架构在观念上的颠覆。

校验框架能够顺利运行,除了被校验数据的表单的定义,主要有以下 3 个部分组成:通用的包文件、配置文件和资源绑定文件,其中配置文件是核心,图 2 是它们之间的关系。

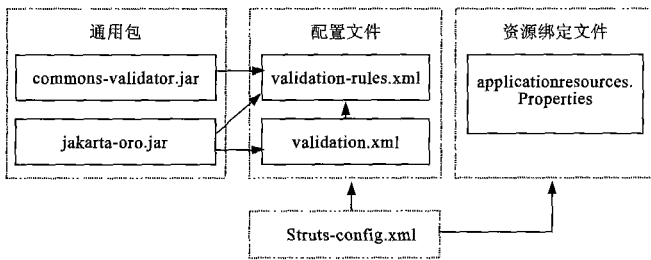


图 2 数据校验框架的部件关系图

(1) 通用的包文件 commons-validator.jar 和 jakarta-oro.jar 是校验框架的基础。commons-validator.jar 文件主要定义了校验的方法,其中包含是否为空格、是否是身份证号码、是否是 Email 地址、是否是日期、是否是整型、长型浮点型数据以及最大和最小长度校验的等方法。Jakarta 的 ORO 包,它主要包含了校验中需要的正则表达式、置换规则、文本划分等方法。

(2) 配置文件 validation-rules.xml 和 validation.xml 是框架的核心,起桥梁作用,把校验规则和数据联系起来。在 Struts 的数据校验框架中,通过 validation-rules.xml 描述校验规则,通过 validation.xml 文件配置被校验数据,整个框架围绕着对这两个 XML 文件的处理展开。

validation-rules.xml 文件具体提供了一些大多数 Web 实际应用中经常涉及的基本校验规则,包括判断数据项是否为空域,EMAIL 格式、日期、时间格式是否正确,是否落在所设置的最大值、最小值范围中等简单的校验规则。validation.xml 文件专门对于应用的,对特定的 Actionform 在这个文件配置了各自的匹配规则。因此只要在这里配置好匹配规则,就不需要在 Actionform 中重复地写复杂的数据校验类了。

(3) 资源绑定文件 applicationresources.properties,主要用于报告出错信息。利用了 Struts 中已有的信息绑定文件,在这里除了系统的一些缺省信息外,还可以根据需要自己定义错误信息和数据项。

另外,数据校验框架中还提供了 JSP 标记,主要用来进行 JavaScript 的校验,使得框架能运用相同的校验规则,实现服务器端的校验,也能够进行客户端的校验。

### 3.2 关键技术

基于 Struts 的数据校验框架主要涉及两个重要技术: plug-in 机制实现在 Struts 中引入数据校验框架, resourcesbundle 技术输出错误提示信息以及数据标签。

#### (1) plug-in 机制

plug-in 机制即插拔的概念。在实际的系统开发中,对于主体的应用程序来说,每个插件就是原来应用程序的一个功能扩展。用户结合自己的具体需要,编写出可被主程序调用的模块代码,这个模块代码不需要在主程序发布的时候同主程序一起编译和连接,而是在主程序启动的时候自动寻找和加载。Plug-in 机制使应用程序具有更大的可扩展性,具有更强的生命力,是面向对象概念在实际软件中的一个体现。

Struts 的 plug-in 机制是通过类 org.apache.struts.validator.ValidatorPlugIn 实现的。在利用 Struts 开发的系统中,若要进行校验时,系统就调用类 ValidatorPlugIn 中的方法 init() 把数据校验框架的两个重要的 XML 文件就加载到内存当中了,当校验结束时调用方法 destory(), 进行释放动作。

如下代码是通过 Struts 的 PlugIn 机制把校验框架的 validation-rules.xml 文件和 validation.xml 文件到插入 Struts 框架中去,在配置文件 struts-config.xml 中设置如下:

```

<plug-in className="org.apache.struts.validator.ValidatorPlugIn"> //引入 PlugIn 机制
    <set-property
        property="pathnames"
        value="/Web-INF/validator-rules.xml,/Web-INF/validation.xml"/>
</plug-in>
  
```

#### (2) Resources Bundle

Resources Bundle 即资源绑定,有利于在 Web 应用开发中根据不同的地区来定义不同地区的信息和相关的文本信息,以避免在代码中重复定义文本信息,造成不必要的代码冗余。在 Struts 的数据校验的错误提示信息通过文件 applicationresources.properties 文件定义,文件本身还提供一些经常要用到的错误提示信息。这样既很好地支持了国际化问题,而且也节省了错误信息的维护时间。

### 3.3 校验流程

本文以一个简单的用户注册信息的校验为例,被校验数据为用户的密码,不能为空,而且长度大于 6 小于 10 的数字串组成,来说明基于 Struts 的数据校验的校验流程。图 3 是当密码数据项为空时,输出的校验信息。要做的工作是配置 validation.xml 以及信息绑定文件 applicationresources-

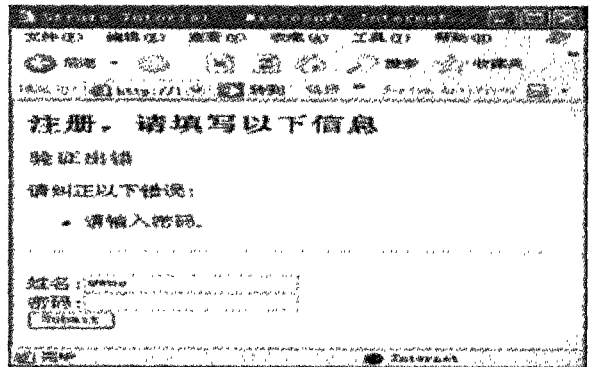


图 3 校验实例图

ces.properties。

(1) 文件 validation.xml

用户名 username 只能由数字字母组成, 密码只能由数据组成, 且长度要大于等于 6 小于等于 10, 配置如下:

```

<formset>
  <form name="StudentForm"> //表单名称, 对应应在 truts-config.xml 中的定义
    <field
      property="password" //需要校验的数据项: 密码
      depends="required,mask,range"> //满足 3 个校验规则
      <msg name="mask" key=" StudentForm. password.maskmsg "/>
      <arg0 key=" StudentForm.password"/>
      <arg1 name="range" key="$ {var: minlength}" resource="false"/>
      <arg2 name="range" key="$ {var: maxlength}" resource="flase"/> //设置密码最大和最小长度
      <var>
        <var-name>mask</var-name>
        <var-value>^[0-9]*$</var-value> //密码有 0-9 的数字组成
      </var>
      <var>
        <var-name>minlength</var-name> //定义最小长度为 6
        <var-value>6</var-value>
      </var>
      <var>
        <var-name>maxlength</var-name> //定义最大长度为 10
        <var-value>10</var-value>
      </var>
    </field>
  </form>
</formset>

```

(2) 文件 applicationresources.properties

以下定义了在校验密码时可能出现的出错信息:

```

errors.invalid=<li>{0} 不正确, 请重新输入.</li>
errors.range=<li>{0} 位数应位于 {1} 和 {2} 之间.</li>
errors.required=<li>请输入 {0}.</li> <!-- 数据项为空时的出错信息-->

```

<!-- {0} 对应 arg0, {1} 对应 arg1, {2} 对应 arg2-->

```

StudentForm.password=密码 <!-- 被校验数据项-->

```

### 4 扩充校验规则的设计思想

虽然基于 Struts 的数据校验框架提供了很多校验规则, 已经能够满足一般的应用, 但是有时发现对于一些特殊的

Web 应用, 需要的校验规则没有包含在框架中, 那么就迫切需要自己来定义校验规则。能否实现把自己的校验规则溶入到基于 Struts 的数据校验框架中, 而且同样能通过配置文件实现校验, 通过信息绑定文件输出提示信息? 本文提供了方便的的扩充校验规则的设计思想, 一般需要通过以下步骤完成。

#### 4.1 特殊校验方法的描述

在数据校验框架中的每个方法都对应一个包 commons-validator.jar 的类 StrutsValidator 中校验方法, 这些校验方法实现根据 validation.xml 中一个数据的一种校验配置, 进行工作, 返回校验结果。校验结果统一为布尔值, 返回一个 true, 表示此数据项合格, 返回 flase, 则表示不合格, 并要求返回错误信息。

此处以在实际的项目 Web 应用开发过程中, 通常会遇到的对两个数据项的比较是否相等, 增加这个校验的方法为例。

首先根据统一的格式定义方法名和相关的参数。

```

public static boolean validateTwoFields(Object bean, ValidatorAction va, Field field,
    ActionErrors errors, HttpServletRequest request,
    ServletContext application) {
    .....
}

```

然后根据 bean 和数据项得到被校验的数据的值, 此处包括需要比较的两个数据项的值。

```

String value = ValidatorUtil.getValueAsString (bean, field.getProperty());
String sProperty2 = field.getVarValue("secondProperty");
String value2 = ValidatorUtil.getValueAsString (bean, sProperty2);

```

最后在取出值后, 进行真正的校验, 校验不合格, 报告出错信息。

```

if (! GenericValidator.isBlankOrNull(value)) {
    try {
        if (! value.equals(value2)) {
            errors.add(field.getKey(), ValidatorUtil.get Action-Error(application, request, va, field));
            return false;
        }
    } catch (Exception e) {
        errors.add(field.getKey(), ValidatorUtil.get Action-Error(application, request, va, field));
        return false;
    }
}
return true;
}

```

#### 4.2 特殊校验方法的定义

一个校验方法对应文件 validation-rules.xml 中的一个校验元素, 各个校验元素各有一个互不相同的名字。增加一

个校验方法的定义,即增加一个校验元素,具体包含这个校验方法的名字、所在的类、类中的方法名以及错误信息的定义。例如两个数据项比较是否相等:

```
<validator name="twofields"
  classname=" org.apache.struts.util.StrutsValidator "
  method="validateTwoFields"
  msg="errors.twofields"/>
```

这样就完成框架中的校验规则的扩充,如果要完成其他校验规则的扩充,可按相同方式进行,这对于一个实际应用的柔韧性是十分有利的。

## 5 移植 Struts 数据框架的设计思想

前面所提到的数据校验框架是基于 Struts 的,那么可否把这种框架移植到一般的 Java 系统中,在系统校验数据时也能够做到一样的简单方便呢?作者发现需要做的工作是除了把相关的通用包文件 commons-validator.jar 和 jakarta-oro.jar 加到系统中,还要解决 3 个问题:① validation-rules.xml 和 validation.xml 文件在 Struts 中是通过 Plug-In 机制自动被应用程序发现和加载的,而一般的 Java 系统不能够自动进行;② Action 和数据项在 Struts 中是自动传到数据校验框架中的;③ 错误信息的输出在 Struts 中是通过 Struts 固有的资源绑定文件,在一般的 Java 中通常没有资源绑定文件。

问题 1 解决方法:要实现这两个重要文件的加载,一般的 Java 系统可在系统初始化时,通过 ValidatorResourcesInitializer 把两个文件装载,并且返回一个 resources 的基于 XML 的实例。

```
首先生成实例 resources,用于装载这两个 XML 文件
.....
```

```
ValidatorResources resources = new ValidatorResources();
.....
```

然后读入 validation-rules.xml,通过 ValidatorResourcesInitializer 装载

```
InputStream rules = ValidateExample.class.getResourceAsStream("validation-rules.xml");
```

```
ValidatorResourcesInitializer.initialize(resources, rules);
```

最后读入 validation.xml,通过 ValidatorResourcesInitializer 装载

```
InputStream forms = ValidateExample.class.getResourceAs-
```

```
Stream("validation.xml");
```

```
ValidatorResourcesInitializer.initialize(resources, forms);
```

问题 2 解决方法:通过创建一个 validator 的实例,把 XML 文件和需要校验的表单联系起来,然后用 validator 的方法 addResource 把表单实例加载到系统中,最后执行 validator 的方法 validate 实现校验。

```
.....
```

```
StudentForm form = new StudentForm ();
```

```
Validator validator = new Validator (resources, " StudentForm ");
```

```
validator.addResource(Validator.BEAN_KEY, form);
```

```
ValidatorResults results = validator.validate();
```

```
.....
```

问题 3 解决方法:利用本身已装载入系统的 validation-rules.xml 文件,在标记 <validator>

的属性 msg 中直接定义对应校验规则的出错信息

```
<validator
```

```
.....
```

```
msg="用户名输入错误,请重新输入."/>
```

通过以上的,在一般的 Java 系统同样具有基于 Struts 的数据校验的灵活性和扩展性,所有的校验规则在外部 XML 文件配置完成,使应用更加方便。

## 6 结束语

Struts 数据校验框架是一种功能强大的数据校验框架,采用这种框架,程序员的代码量会显著减少,好扩展,工作效率会提高。这种框架不仅可以在 Struts 的项目开发中应用,它的那种模式思想还可以移植到一般的 Java 和 XML 相结合的 Web 应用开发中,甚至于基于 J2EE 的大型系统的应用开发中。

## 参 考 文 献:

- [1] 何成万,余秋惠. MVC 模型 2 及软件框架 Struts 的研究[J]. 计算机工程, 2002, 28(6): 274-281.
- [2] 葛瀛龙. 用 Java 来设计组件重用的 Query 方法[J]. 计算机工程与应用, 2002, 38(20): 103-106.
- [3] Struts Validator [CP/OL]. <http://home.earthlink.net/~dwinterrfeldt/>.

(上接第 1312 页)

根据实际情况使用串口,设置通信协议;经过多次试验,调整了缓冲区大小,避免内存浪费以及数据丢失;对数据的读取操作做了优化。从系统运行结果看,数据通信实时、稳定、可靠。

## 参 考 文 献:

- [1] 李现勇. VC++ 串口通信技术与工程实践[M]. 北京:人民邮电出版社, 2002.

- [2] 鲁沐浴. C 语言最新编程技巧 200 例[M]. 北京:电子工业出版社, 1997.
- [3] 李朝青. PC 机及单片机数据通信技术[M]. 北京:北京航空航天大学出版社, 2000.
- [4] 李鹏. 计算机通信技术及其程序设计[M]. 西安:西安电子科技大学出版社, 1998.
- [5] 周术诚, 蒋萌辉, 宁正元. 微机与单片机串行通信中断方式的实现[J]. 福州:福州大学学报, 2001, 29(8): 79-82.