

用Web程序从Struts向Stripes框架的移植 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/252/2021\\_2022\\_\\_E7\\_94\\_A8Web\\_E7\\_A8\\_8B\\_E5\\_c104\\_252536.htm](https://www.100test.com/kao_ti2020/252/2021_2022__E7_94_A8Web_E7_A8_8B_E5_c104_252536.htm) 一、引言 把一个现有Java Web应用程序移植到一种新框架可能不是大多数开发者最感兴趣的问题。除了要花费时间学习一种新的Web框架外，例如标签、国际化系统和校验等繁重的转化过程可能会迫使每一位程序员考虑再三。我最近就面临这样的挑战-从Struts进行移植。在决定移植一个应用程序前，应该首先问一下“为何不使用现在的框架？”在我看来，Struts是一种稳定的具有良好文档的框架，并且有一大批开发者社区成员，但是其配置很麻烦，而且其表单、行为、应用程序流和校验的分离有时会带来很多麻烦。这种情形在我的Struts应用程序不断变大时越发糟糕。最后，纯粹从一种维护的角度，我决定把它移植到一种新的框架。开始，我认为没有一种框架(Java ServerFaces，Tapestry，WebWorks，Spring MVC)值得从Struts迁移向其迁移。例如JSF这样的框架看上去极不友好。其它的，例如Tapestry和WebWorks，涉及到整页整页的看上去令人麻烦的国际化系统。而从配置角度来看，Spring MVC看上去并不比Struts好多少。我选择的框架应该仅需适当的学习时间，还要与移植效益相称.而且，它还一定要使我编码、排错与维护更为容易。二、发现Stripes框架 后来，我偶然发现了Stripes框架。就象Java社区中的许多发烧友一样，我一直追随着Ruby on Rails(RoR)现象。依我看来，Stripes是最接近于RoR哲学的Java MVC框架-简单、漂亮，并且要求最小的配置。除了它的简洁外，象我这样一位Struts程序员，Stripes非

常适合我的口味。应用程序流和许多命名惯例都与之十分相似。Stripes中的ActionBeans就象Strut的Actions，而ForwardResolutions极象ActionForwards。因此，使用这一框架，我不必抛弃我所有以前的Struts知识。另外吸引我的是Stripes文档。象框架本身一样，文档也是干净、清洁而简练。其标签库文档和API都具有良好的归档，而且该框架的每一种特征几乎都有相应的示例源码。这些优秀的文档再加上我的现有Struts知识使我坚信，我可以快速地掌握这种Stripes框架。值得注意的是，Stripes还包括另外一些使其成为一种良好的AJAX平台的特征，例如它提供了一种流式方案，该方案允许对AJAX实现进行改进的错误处理。然而，对于我来说，最终的决定因素还是我能够清楚地看到它会使我的生活更容易些。我估计，在我的应用程序的行为/配置/校验部分，我只需使用约一半的代码就够了。更少的代码意味着了更少的错误、更快的开发时间和更容易的纠错。

### 三、移植过程

我从视图层开始移植，然后再向行为层移植。事实上，我也没有很明确的逻辑思路。只是必须从某处开始，而视图部分看起来更适合于作为一个起始点。

(一) JavaServer Pages 就象Struts一样，Stripes使用JSP来实现其视图层。我吃惊地发现，Stripes标签库非常类似于Struts的HTML taglib。事实上，我能够使用这种统一替换方式来升级我的许多标签。Stripes依赖于JSTL实现JSP视图中的逻辑。我在我的应用程序中混合使用了Struts逻辑标签和JSTL。通过把我的所有逻辑标签移植到JSTL，我能够利用JSTL的优越的if/else和case语句的能力处理，它们可能是很原始的或者根本不存在于Struts逻辑taglib中。

(二) 国际化

接下来，我要移植我的Struts的消息资源。在配置端，所有要

求的操作就是重命名我的Struts消息资源文件。在我的JSP中，我能够使用统一替换方式把我的所有Struts message标签(例如，)替换为JSTL格式标签(例如，)。这种JSTL格式标签还支持可用于Struts中的消息资源绑定。(三) 表单 我的移植的最有意义的部分是去掉了我的Struts Action表单，这是在Action类中进行的，要求大量的XML标记和冗长的转换，如下例所示：

```
type="org.apache.struts.validator.DynaValidatorForm"> public
ActionForward executeAction( ActionMapping mapping ,
ActionForm form , HttpServletRequest request ,
HttpServletRequest response) throws Exception {Employee
employee=new Employee(). DynaValidatorForm eaf =
(DynaValidatorForm) form.
employee.setFirstname(eaf.getString("firstname")).
employee.setLastname(eaf.getString("lastname")).
employee.setPhone (eaf.getString("phone")). employee.setEmail
(eaf.getString("email")). employee.setEmployeeid
((Integer)eaf.get("employeeid")).
employee.setSocialsecurity(Long.parseLong(eaf.getString("socialsec
urity"))).
employee.setBirthdate(MyUtils.convertStringToDate(eaf.getString("
birthdate"))).
employee.setSalary(MyUtils.convertStringToBigDecimal(eaf.getStri
ng("salary"))).
EmployeeDAOService.updateEmployee(employee). return new
ActionForward(mapping.getForward()). } 另一方面，Stripes表单
处理允许你把你的域对象用作一个表单使用： public class
```

```
UpdateEmployeeActionBean implements ActionBean { private
ActionBeanContext context. private Employee employee. public
ActionBeanContext getContext() { return context. } public void
setContext(ActionBeanContext context) { this.context = context. }
public void setEmployee(Employee employee) { this.employee =
employee. } public Employee getEmployee() { return this.employee.
} public Resolution 0update() {
EmployeeDAOService.0updateEmployee(employee). return new
ForwardResolution("/employees/0updateEmployee.jsp"). }} 100Test
下载频道开通，各类考试题目直接下载。详细请访问
www.100test.com
```