

JavaServer Faces 2.0 完全参考手册

- 全面介绍JSF 2.0
- 详述如何使用Ajax，以及按照JSF 2.0的方式构建组件
- 快速理解众多可以直接运行的代码示例



YZLI0890118113

Ed Burns 著
(美) Chris Schalk
Neil Griffin 合著
陶 克 熊淑华 译

JavaServer Faces 2.0

完全参考手册

Ed Burns
(美) Chris Schalk
Neil Griffin
陶克 熊淑华
著
合著
译



YZLI0890118113

清华大学出版社

北 京

Ed Burns Chris Schalk Neil Griffin
JavaServer Faces 2.0: The Complete Reference
EISBN: 978-0-07-162509-8

Copyright © 2010 by The McGraw-Hill Companies, Inc.

All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including without limitation photocopying, recording, taping, or any database, information or retrieval system, without the prior written permission of the publisher.

This authorized Chinese translation edition is jointly published by McGraw-Hill Education (Asia) and Tsinghua University Press. This edition is authorized for sale in the People's Republic of China only, excluding Hong Kong, Macao SAR and Taiwan.

Copyright © 2011 by McGraw-Hill Education (Asia), a division of the Singapore Branch of The McGraw-Hill Companies, Inc. and Tsinghua University Press.

版权所有。未经出版人事先书面许可，对本出版物的任何部分不得以任何方式或途径复制或传播，包括但不限于复印、录制、录音，或通过任何数据库、信息或可检索的系统。

本授权中文简体字翻译版由麦格劳-希尔(亚洲)教育出版公司和清华大学出版社合作出版。此版本经授权仅限在中华人民共和国境内(不包括香港特别行政区、澳门特别行政区和台湾)销售。

版权 © 2011 由麦格劳-希尔(亚洲)教育出版公司与清华大学出版社所有。

北京市版权局著作权合同登记号 图字: 01-2010-4105

本书封面贴有 McGraw-Hill 公司防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

JavaServer Faces 2.0 完全参考手册 / (美)伯恩斯(Burns, E.), (美)沙尔克(Schalk, C.), (美)格里芬(Griffin, N.) 著; 陶克 熊淑华 译. —北京: 清华大学出版社, 2012.1

书名原文: JavaSever Faces 2.0: The Complete Reference

ISBN 978-7-302-27446-9

I. J… II. ①伯… ②沙… ③格… ④陶… ⑤熊… III. JAVA 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2011)第 246981 号

责任编辑: 王 军 张立浩

装帧设计: 牛艳敏

责任校对: 邱晓玉

责任印制: 王秀菊

出版发行: 清华大学出版社

地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编: 100084

社 总 机: 010-62770175

邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者: 清华大学印刷厂

经 销: 全国新华书店

开 本: 185×260 印 张: 41.75 字 数: 1016 千字

版 次: 2012 年 1 月第 1 版 印 次: 2012 年 1 月第 1 次印刷

印 数: 1~3000

定 价: 98.00 元

产品编号: 038319-01

内 容 提 要

本书全面介绍了 JavaServer Faces 2.0 框架结构、扩展的 JavaServer Faces 以及 JavaServer Faces 工具和库的相关内容，包括 JSF 2.0 复合组件，JSF 2.0 支持的标准 Ajax，支持 POST、REDIRECT、GET，隐式导航，可定制的 JSF 页面，以及新版本对安全、移动渲染、本地化和表达式语言的增强等，本书为这些标准的组件提供了完整的参考和指导信息。

本书通过对概念的解释和简单的示例说明相结合的方式，详尽地介绍了 JavaServer Faces 2.0 的相关基础知识，同时还以一个虚拟教练应用程序贯穿后面部分的内容，随着内容的深入，逐步提供更多更高级的信息。本书适合具有各级经验的广大读者阅读，包括具有中级 Web 开发经验的开发人员和高级企业 Java 架构师等。

致 谢

JavaServer Faces 是一种基本技术，它建立在其他多层软件技术之上。与其他基础软件产品一样，JavaServer Faces 是多年来很多个人和组织努力工作和无私奉献的结果。记录这些技术的书籍同样需要经过这样一个过程，因此我希望能够向那些为本书出版以及 JavaServer Faces 的发布做过贡献的人表示感谢。

越来越多的信息只在网络上就可以找到，很多的这类信息是个人“自行发布的”。我想要说的是，出版仍然是发布高质量有用信息的最佳方式。在与 McGraw-Hill 公司的出版团队一同工作之后，我理解了为何网络文献无法取代一本书页被翻到折角的、脏兮兮的旧书。在写作本书的过程中，编辑 Herb Schildt 为我提供了大量的指导和帮助。感谢 Herb 敏锐的洞察力。采编协调员 Joya Anthony 非常出色地将本书的不同部分整合在了一起。McGraw-Hill 公司的编辑部主任 Wendy Rinaldi 在运作这个项目的过程中经历了许多波折，却从未对其失去信心，我为自己能够将这本书交付给她而感到非常自豪。感谢编审 Robert Campbell 和项目经理 Smita Rajan。

致我的好妻子 Amy：我在这本书上花费了如此多的时间，感谢你对此情况的理解和耐心。你承担了家庭琐事的重担，没有你的帮助和支持，我是无法完成这项工作的。还要感谢我的儿子 Owen 和 Logan 对父亲总是不在家表示的理解。

我必须感谢那些引领我进入和工作在 JSF 领域并帮助我完成本书的朋友。既然 JSF 是一个 JCP 项目，那么它的专家团队显然值得着重介绍。JSF 2.0 专家团队的关键成员包括 Dan Allen、Keith Donald、Mike Freedman、David Geary、Ted Goddard、Jeremy Grelle、Roger Keays、Gavin King、Jason Lee、Ryan Lubke、Craig McClanahan、Kito Mann、Martin Marinschek、Pete Muir、Joseph Ottinger、Ken Paulsen、Andy Schwartz、Alexandr Smirnov 和 Adam Winer(以上按姓氏字母顺序排序)。JSF 不只是一个 JCP 项目，它还是一个由开发人员社区主导的项目，有许多曾在社区中帮助 JSF 获得成功的人应当得到感谢，他们是 Lincoln Baxter III、Dennis Byrne、Çağatay Çivici、Hanspeter Dünnenberger、Alexander Jesse、Max Katz、Alberto Lemos(Dr. Spock)、Imre Oßwald、Hazem Saleh、Stan Silvert、Yara 和 Vinicius Senger，以及 Matthias Weßendorf。George Drapeau 在 2001 年曾建议我参加领导团队工作的面试，感谢你，George！原始规范的主管 Amy Fowler 为 JSF 的成功做出了重要

IV JavaServer Faces 2.0 完全参考手册

的贡献，与你一同工作是一种享受。

Jim Driscoll、Mimi Hills、Tony Ng、Sridatta Viswanath 和 Barbara Louis 在 JSF 的整个开发过程中一直担任支持经理。向我最初的 JSF 团队核心成员 Jayashri Visvanathan 和 Roger Kitain 致以最诚挚的感谢。你们是这个项目的灵魂，也是我所遇到的最棒的团队。感谢 Ryan Lubke、Justyna Horwat 和 Jennifer Ball 这些 JSF 执行团队的外围成员。Ryan，特别感谢你在领导 Sun 的 JSF 执行部门时对质量与创新始终如一的坚持。我还想向 Jacob Hookom 对 JSF 在 Facelets 形式下的系统框架构建所做出的贡献以及他在将 JSF 确立为实现 Ajax 最佳方式的工作中不断展现的创造力表示感谢。

我还想特别感谢为本书撰写了第 15、16 章和附录的 Neil Griffin。

当然，我必须感谢 Chris Schalk 让我参与这个项目，并成为我伟大的合作伙伴。我明白，在如此密切的距离内，自己并不是一个容易共事的人。非常感谢你的耐心以及你对新用户及读者不断的建议。没有你，这本书是无法完成的。

最后，我想感谢始终不渝的支持者：我的父母、Brendan Burns、Lisa Lane、Diana Dean、Jeff Beckberger、Joe McCabe 和 Vern Singleton。

—Ed Burns

Altamonte Springs, FL

前言

JSF 2.0 中有什么新功能呢？

JavaServer Faces 规范的 1.2 版本和 2.0 版本之间的变化是很重要的，这种变化比 Java EE 平台中任何两种相邻版本之间规范的变化都要大得多。由于变化如此之多，因此有一个很方便的参考手册对熟悉新功能而言是很重要的。这里主要对新的功能进行分类和剖析，通过这些功能可以直接找到相应的章节。所有 JSF 子系统的图形索引如图 1 所示。浅色阴影框表示的是 JSF 2.0 中全新的功能，深色阴影框表示的是在 JSF 2.0 中变化或加强的功能。方框的排放位置表明功能之间的依赖关系。如果功能相互依赖，但是方框却不能相邻，那么使用箭头表示这种依赖关系。

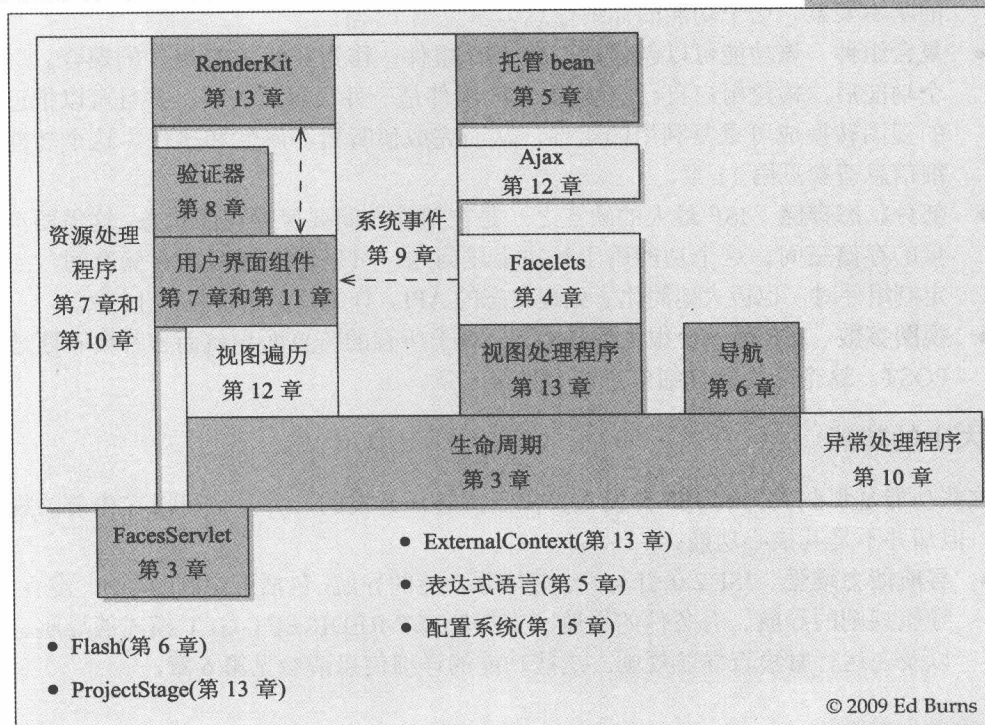


图 1 JSF 子系统的图形索引

在接下来的内容中, 这些功能按照复杂程度来衡量并进行分组, 其中大的功能与小的功能相比较而言设计起来更为困难且更为重要。

基本功能

这些功能由专家组开发, 使得 JSF 2.0 中的其他功能得以实现; 然而, 这些功能也是同样有用的, 因此对公共 API 是可用的:

- **系统事件** 这个功能在 JSF 运行时处理请求, 提供了细粒度级的观察和执行动作的细节。详细信息请参见第 9 章。
- **资源** 这个功能使 JSF 运行时可以提供静态资源(如样式表、脚本和图像), 此外, 也可以像之前版本一样提供 JSF 页面。这些都将在第 7 章中进行介绍, 并将在第 10 章中详细说明。
- **Facelets** Facelets 首先是作为一个开放源代码的 JSF 扩展, 它提供了一流的模板, 并且可以很容易地把标记和 JSF API 集成在一起。这使得编写 JSF 页面的过程变得越来越容易, 并且可维护性也得到增强。Facelets 包含在核心的 JSF 2.0 版本规范中, 详细信息请参阅第 4 章。

大的功能

这些功能是“新规范推出的动因”, 即在先前发布规范的基础上添加的新功能:

- **Ajax** 这个功能使得 JSF 视图可以通过浏览器与服务器直接进行交互, 而不需要对浏览器窗口中的整个页面请求刷新。Ajax 是当前大多数产品质量级 Web 应用程序的基本要素。这个功能的详细信息请参见第 12 章。
- **复合组件** 该功能可以创建真正的 JSF 组件, 作为其他 JSF 组件的聚合。具备这个功能后, 构建用户自己定制的 JSF 组件是一件很容易的事, 并且可以把已存在的视图转换成可重复利用的组件, 从而完成侦听器、属性和事件。这个功能的详细信息请参见第 11 章。
- **部分状态存储** JSF 最大的缺点之一是在请求之间要保存视图状态, 这需要消耗大量的存储空间。这个功能用于解决上述问题, 可以动态地减少存储空间, 当开发定制组件时, 也极大地简化了处理状态的 API。详细信息请参见第 11 章。
- **视图参数** JSF 另一个很大的缺点是: 对于所有的跨页面导航而言都要不断地使用 POST。这个功能的详细信息请参见第 6 章。

中等大小的功能

这些功能是那些优秀的 JSF 开发人员出于兴趣而开发的, 并且也可以在先前的规范中扩展, 但却并不是其核心功能。

- **导航能力增强** JSF 2.0 引入了一些增强功能的导航, 包括可定制的导航、没有 XML 导航规则的导航、有条件的导航、支持 POST-REDIRECT-GET 模式的导航、flash 以及在运行时检查导航规则。这些功能的详细信息请参见第 6 章。

- **异常处理** JSF 2.0 现在具备一个中央 `ExceptionHandler`，通过这里时，所有的异常处理都会被筛选。这使得构建一个使用 JSF 组件的错误页面变得很容易。这个功能的详细信息请参见第 10 章。
- **表达式语言增强** 引入了一些新的隐式对象，并且 EL 现在对任意的 Java 方法都支持方法调用，包括参数传递。这个功能的详细信息请参见第 5 章。
- **验证器** 一个完整的新 Java 规范已经发展成为地址验证器，这是一种 JSR-303 Bean 验证器。这种规范在 JSF 2.0 中整合良好，详细信息请参见第 8 章。
- **新的作用域** 除了在第 5 章中描述的 `flash` 作用域外，JSF 2.0 还为声明定制作用域提供了一种机制。详细信息请参见第 13 章。

小的新功能

这些功能主要用于 bug 修复，在该规范的次要修订版本中无法修复这些严重的 bug：

- 现在可以在应用程序启动和关闭期间访问 `FacesContext`。这个功能的使用请参见第 10 章。
- `UISelectItems` 是从 `UISelectOne` 中扩展得到的子组件，并且 `UISelectMany` 可以引用任意对象的集合。
- JSF 现在允许开发人员告诉运行时处于软件开发生命周期中的哪个阶段，即运行时正在执行哪个阶段。这个“项目状态”功能使运行时可以输出更多提示性的错误消息，并且在项目正在开发时做出提示。这个功能的使用将在第 13 章介绍。
- 内注可以避免对 XML 的需求。几乎每一个 XML 元素都可以保存在 `faces-config.xml` 文件中，现在可以使用 Java 语言内注来代替它。这种内注功能与“不需要 XML 导航规则的导航”相结合，可以在完全没有 `faces-config.xml` 文件的情况下构建 JSF 应用程序。有关内注的详细信息请参见第 15 章。

微小的新功能

这些功能用于充实新功能，并且包括一些简单的 bug 修复，这些 bug 修复已经整合到 JSF 2.0 规范发布中：

- 关于新方法 `ExternalContext` 和 `FacesContext` 的详细信息，请参见第 13 章。
- 通过装饰设计模式来帮助扩展 JSF 的封装器，请参见第 13 章。
- 新的 `DataModel.iterator` 功能将在第 10 章中进行介绍。
- 有一些新的 `context-param` 设置，详细信息请参见第 15 章。

如果您是一个很有经验的 JSF 用户，那么可以使用新功能前的引导索引直接跳到这些新的资料。如果您是第一次接触 JSF，那么并不需要了解 JSF 2.0 中的这些新功能，以及它们是如何从先前的版本中移植过来的。所有这些新功能都将很自然地展现在本书的内容中。事实上，我们认为，JSF 2.0 保留了 JSF 一贯的风格。

本书深入阐述了如下内容：

- 教程内容涵盖了 `JavaServer Faces` 技术的每一个方面；
- 完全覆盖了 JSF 2.0 中的所有新功能，并且不是采用补充说明的方式，而是完全“按照 JSF 2.0 应用使用的说明方式”进行阐述；

- 图形索引快速展示了在 JSF 子系统中的什么地方可以迅速找到信息；
- JSF 2.0 提示概述了 JSF 最新版本之间的差异以及如何最好地使用它们；
- 专家组意见提供了技术背后的设计原理，让读者更好地理解在自己的工作中如何使用 JSF；
- 用多个示例详细阐述了定制用户界面组件的开发，包括 JSF 2.0 复合组件；
- 完整地覆盖了 JSF 2.0 支持的标准 Ajax；
- 真实应用示例，包括支持 POST、REDIRECT、GET，隐式导航以及可定制的 JSF 页面；
- 扩展 JSF 框架的一个完整指南，如安全、移动渲染、本地化和表达式语言的增强；
- 对 Neil Griffin、JSF 2.0 专家组的成员和 Liferay Portal JSF 团队领导人引入的 JSF 配置文件的一个完整指导；
- 为规范的标准组件提供完整的参考和指导信息。

开发环境

当提供支持 Java 和 JSF 的 IDE 时，本书不要求读者使用任何特定的 IDE。一个简单的基本开发环境应该包括：

- Java SE 6
- 一个 Servlet 2.5 的容器，如 Apache Tomcat 5.x
- Apache Maven

读者如果尝试使用书中提到的代码示例，那么需要使用上述 3 种环境。运行 JSF 2.0 最简单的环境是 Sun 公司的 Glassfish Application Server，网址为 www.glassfish.org。

网上示例代码资源

整本书中经常引用网上的代码，有时候只是指明网址或者称之为“网上扩展”。书中的所有代码示例都可以在 McGraw-Hill 的网站下载，网址为 www.mhprofessional.com/computingdownload，也可以从 www.tupwk.com.cn/downpage 网站下载。此外，本书自己的网站 www.jsfcompref.com 也提供可下载的源代码。

目 录

第 I 部分 JavaServer Faces 框架	
第 1 章 JavaServer Faces 简介	3
1.1 什么是 JavaServer Faces	3
1.2 JavaServer Faces 的历史	4
1.2.1 公共网关接口	4
1.2.2 Servlet API	4
1.2.3 Java 服务器页面	5
1.2.4 Apache Struts	5
1.2.5 Spring 框架和 Spring MVC	6
1.2.6 JavaServer Faces 的诞生	6
1.3 JavaServer Faces 设计目标	7
1.4 JSF 应用程序架构	8
1.4.1 JSF 请求处理生命周期	10
1.4.2 JSF 导航模型	11
第 2 章 构建一个简单的 JavaServer Faces 应用程序	15
2.1 应用程序概述	15
2.1.1 JSFReg 应用程序文件	16
2.1.2 JSF 软件栈	17
2.1.3 装配 JSFReg 应用程序	17
2.1.4 配置文件	18
2.1.5 Facelets 页面	19
2.2 构建和运行应用程序	28
2.3 应用程序关键部分回顾	30
第 3 章 JavaServer Faces 请求处理生命周期	31
3.1 JSF 请求处理生命周期概述	31
3.1.1 请求处理生命周期的功能	32
3.1.2 与其他 Web 技术的区别	32
3.1.3 服务器端视图的自动管理与同步	33
3.1.4 请求处理生命周期的各阶段	34
3.2 观察请求处理生命周期	40
3.3 与请求处理生命周期有关的高级主题	43
3.3.1 使用 immediate 属性	43
3.3.2 阶段侦听器	45
3.3.3 异常处理程序	45
3.4 关键的生命周期概念	46
第 4 章 Facelets 视图声明语言	47
4.1 JSF 中使用模板化的威力	47
4.2 JSP 与 Facelets 的异同	48
4.3 使用 Facelets 执行模板化	50
4.4 Facelets 模板化标签使用指南	53
4.4.1 ui:composition	53
4.4.2 ui:decorate	54
4.4.3 ui:define	55
4.4.4 ui:insert	55
4.4.5 ui:include	55
4.4.6 ui:param	56
4.5 Facelets 非模板化标签使用指南	56
4.5.1 ui:component	56
4.5.2 ui:fragment	57

4.5.3	ui:remove	57	6.4.3	使用重定向	108
4.5.4	ui:debug	57	6.4.4	视图参数的 XML 配置	121
第 5 章	托管 bean 与 JSF 表达式语言	59	6.4.5	在 Servlet 错误页上使用 JSF 组件	122
5.1	托管 bean 概念	60	第 7 章	用户界面组件模型	123
5.1.1	简单托管 bean 示例	60	7.1	什么是用户界面组件	123
5.1.2	初始化托管 bean 属性	62	7.1.1	基于组件的 Web 开发的兴起	124
5.1.3	把 List 和 Map 声明为托管 bean	68	7.1.2	JavaServer Faces 用户界面组件的目标	125
5.1.4	托管 bean 的相互依赖	69	7.2	JSF 用户界面组件架构介绍	126
5.1.5	使用 EL 设置托管属性	71	7.2.1	用户界面组件树(视图)	129
5.2	控制托管 bean 生命周期	71	7.2.2	用户界面组件和相关 的“活动部分”	131
5.3	JSF 表达式语言	74	7.3	组件资源	133
5.3.1	JSF 1.1 与 JSF 1.2 之间表达式 语言的关键区别	74	7.4	用户界面组件和 Facelets	133
5.3.2	统一 EL 概念	76	7.4.1	用编程方式访问用户 界面组件	133
5.3.3	值表达式	76	7.4.2	在 JSF 视图中绑定用户 界面组件的有用建议	136
5.3.4	表达式操作符	82	第 8 章	数据转换与数据验证	139
5.3.5	方法表达式	83	8.1	验证和转换的示例	140
5.4	托管 bean 的 Web 应用程序 开发细节	86	8.2	转换和验证揭秘	142
5.4.1	采用编程方式访问托管 bean	86	8.3	Faces 转换器系统	145
5.4.2	使用托管 bean 作为 JSF 页面 的支撑 bean	88	8.3.1	DateTimeConverter	146
第 6 章	导航模型	93	8.3.2	NumberConverter	147
6.1	使用隐式导航	94	8.3.3	关联转换器与 UIComponent 实例	147
6.2	JSF 导航系统概述	96	8.3.4	转换器的生命周期	152
6.2.1	回顾 MVC——控制器	96	8.3.5	定制转换器	153
6.2.2	NavigationHandler——幕后 主管	97	8.4	Faces 验证系统	157
6.2.3	Faces 动作方法说明	98	8.4.1	LongRangeValidator	158
6.3	构建导航规则	99	8.4.2	DoubleRangeValidator	159
6.3.1	静态导航示例	100	8.4.3	LengthValidator	159
6.3.2	动态导航示例	101	8.4.4	必需的工具 Required- Validator	160
6.4	更复杂的导航示例	105	8.4.5	RegExValidator	160
6.4.1	使用通配符	105	8.4.6	BeanValidator	161
6.4.2	使用条件导航	106			

8.5 关联验证器与 UIComponent 实例	161	9.3 编写定制动作和值修改侦听器	190
8.5.1 使用标记关联验证器与 UIComponent 实例	161	9.4 更多的 Faces 事件示例	192
8.5.2 利用标记和验证器属性关联验证器与 UIComponent 实例	162	9.4.1 使用值修改事件自动填充字段	193
8.5.3 采用编程方式关联验证器与 UIComponent 实例	163	9.4.2 扩展值修改示例	196
8.5.4 验证器的生命周期	164	9.5 使用阶段事件和侦听器	198
8.5.5 定制验证器	165	9.6 使用系统事件和侦听器	201
8.6 使用 JSF 的 bean 验证	166	9.6.1 怎样发布系统事件	204
8.6.1 什么是 bean 验证	167	9.6.2 怎样订阅系统事件	205
8.6.2 JSF 验证和 bean 验证之间的区别	167	9.7 创建定制应用程序事件和侦听器的一般规则	207
8.6.3 使用 JSF 的 bean 验证的细节	168		
8.6.4 使用 bean 验证方式验证 JSF 托管 bean 属性	170	第 II 部分 扩展 JavaServer Faces	
8.7 汇总: 视图上的消息	171	第 10 章 应用 JSF: 虚拟教练	
8.7.1 FacesContext 上与 FacesMessage 相关的方法	171	应用程序简介	211
8.7.2 UIViewRoot 及其 Locale 属性	173	10.1 虚拟教练应用程序概观	212
8.7.3 何时、如何创建 FacesMessage 实例并把它添加到 FacesContext	173	10.1.1 注册和登录到虚拟教练应用程序	212
8.7.4 FacesMessages 的渲染	174	10.1.2 创建新的训练赛事检测计划	213
8.7.5 消息和 bean 验证	177	10.1.3 选择教练	214
		10.1.4 只有教练可以执行的动作	214
第 9 章 JSF 事件模型	179	10.2 虚拟教练应用程序需求	216
9.1 JSF 事件模型概述	179	10.3 虚拟教练应用程序架构	217
9.2 JSF 事件的工作方式	183	10.4 Facelet 页面	217
9.2.1 Faces 事件侦听器接口和事件类	184	10.4.1 Web 应用程序根目录中的页面	217
9.2.2 处理 Faces 事件的时机	185	10.4.2 /user 目录中的页面	218
9.2.3 应用程序事件	187	10.4.3 /trainer 目录中的页面	218
9.2.4 剖析值修改事件	189	10.4.4 /resources 目录中的页面	218
		10.5 支撑 bean	219
		10.6 持久性和事务架构	221
		10.7 与导航相关的问题	225
		10.8 创建简单的验证系统	228
		10.9 重温 JSFReg: 构建注册系统	232

10.10	构建虚拟教练应用程序 的核心页面.....	236	11.5.2	<code>#{cc}</code> 隐式对象	278
10.10.1	创建 <code>allEvents.xhtml</code> 页面	236	11.6	用户界面组件的各部分	279
10.10.2	<code>editTrainingEvent.xhtml</code> 的教练版本	239	11.7	非组合 JSF 定制用户界面 组件	280
10.10.3	创建 <code>viewTrainees.xhtml</code> 和 <code>viewTrainingSessions- ForUser.xhtml</code> 页面	242	11.7.1	构建 <code>HtmlHelloWorld</code> 示例	280
10.11	<code>UserRegistry</code> 和 <code>EventRegistry</code>	245	11.7.2	接收表单输入的 <code>Hello- World</code> 用户界面组件	284
10.11.1	访问和初始化 <code>UserRegistry</code> 实例	245	11.7.3	JSF 股票报价组件	287
10.11.2	读写 <code>User</code> 实例	246	11.8	定制界面组件的状态 管理	289
10.11.3	读写 <code>TrainingSession</code> 实例	248	11.8.1	对 <code>StateHelper</code> 的写操作	289
10.12	访问和初始化 <code>EventRegistry</code> 实例	249	11.8.2	对 <code>StateHelper</code> 的读操作	290
10.13	读写 <code>Event</code> 实例	250	11.8.3	从 <code>StateHelper</code> 中删除值	290
10.14	虚拟教练应用程序中的 JPA 和实体类	252	11.9	将渲染代码提取到 渲染器中	291
10.14.1	JPA 的 XML 配置文件 <code>persistence.xml</code>	254	11.9.1	创建定制 <code>Facelet</code> 标签库 <code>TagHandler</code>	292
10.14.2	定制 <code>ExceptionHandler</code> 示例	255	11.9.2	使用 <code>RenderKit</code> 支持多种 客户端设备类型	294
10.15	虚拟教练应用程序 的国际化	257	11.10	高级复合组件的开发	302
10.16	对于虚拟教练应用程序 的最后评论	267	11.10.1	为 <code><vt:loginPanel></code> 创建 支撑类	302
第 11 章	构建定制用户界面组件	269	11.10.2	回顾复合组件特性	307
11.1	构建定制用户界面组件 的时机	269	11.11	将 JSF 组件打包成自包含 的 Jar 文件	307
11.2	用户界面组件定义	270	11.12	将复合组件打包成 jar 文件	307
11.3	简单的 JSF 用户界面组件	271	11.12.1	将复合组件 <code>Facelet</code> 文件打包	307
11.4	简单 JSF 用户界面组件 标记背后的代码	273	11.12.2	为定制组件库声明 定制名称空间 URL	307
11.5	为用户界面组件添加行为	274	11.12.3	将复合组件的支撑 类打包	308
11.5.1	复合组件细节初探	276	11.13	将非复合组件打包成 Jar 文件	308

15.5.7	attribute-class 元素	407	15.5.37	from-action 元素	431
15.5.8	attribute-name 元素	407	15.5.38	from-outcome 元素	432
15.5.9	base-name 元素	408	15.5.39	from-view-id 元素	433
15.5.10	before 元素	409	15.5.40	if 元素	433
15.5.11	behavior 元素	410	15.5.41	key 元素	434
15.5.12	behavior-class 元素	411	15.5.42	key-class 元素	435
15.5.13	behavior-id 元素	411	15.5.43	lifecycle 元素	436
15.5.14	client-behavior-renderer 元素	412	15.5.44	lifecycle-factory 元素	437
15.5.15	client-behavior-renderer- class 元素	413	15.5.45	list-entries 元素	437
15.5.16	client-behavior-renderer- type 元素	414	15.5.46	locale-config 元素	439
15.5.17	component 元素	414	15.5.47	managed-bean 元素	439
15.5.18	component-class 元素	415	15.5.48	managed-bean-class 元素	441
15.5.19	component-family 元素	416	15.5.49	managed-bean-name 元素	441
15.5.20	component-type 元素	417	15.5.50	managed-bean-scope 元素	442
15.5.21	converter 元素	418	15.5.51	managed-property 元素	443
15.5.22	converter-class 元素	419	15.5.52	map-entries 元素	445
15.5.23	converter-for-class 元素	419	15.5.53	map-entry 元素	446
15.5.24	converter-id 元素	420	15.5.54	message-bundle 元素	447
15.5.25	default-locale 元素	421	15.5.55	name 元素	448
15.5.26	default-render-kit-id 元素	422	15.5.56	navigation-case 元素	450
15.5.27	default-validators 元素	423	15.5.57	navigation-handler 元素	450
15.5.28	default-value 元素	423	15.5.58	navigation-rule 元素	451
15.5.29	el-resolver 元素	425	15.5.59	null-value 元素	452
15.5.30	exception-handle-factory 元素	425	15.5.60	ordering 元素	454
15.5.31	external-context-factory 元素	426	15.5.61	others 元素	455
15.5.32	faces-config 元素	427	15.5.62	partial-view-context- factory 元素	456
15.5.33	faces-context-factory 元素	428	15.5.63	phase-listener 元素	456
15.5.34	facet 元素	428	15.5.64	property 元素	457
15.5.35	facet-name 元素	430	15.5.65	property-class 元素	459
15.5.36	factory 元素	431	15.5.66	property-name 元素	459
			15.5.67	property-resolver 元素	460
			15.5.68	redirect 元素	461

15.5.69	referenced-bean 元素	462	15.5.99	view-param 元素	488
15.5.70	referenced-bean-class 元素	462	15.5.100	visit-context-factory 元素	489
15.5.71	referenced-bean-name 元素	463	15.6	扩展元素	489
15.5.72	render-kit 元素	464	第 16 章	标准的 JSF 组件库	493
15.5.73	render-kit-class 元素	465	16.1	JSF、Facelet 和 JSP 标签术语 概览	493
15.5.74	render-kit-factory 元素	466	16.2	获取并安装标准库	494
15.5.75	render-kit-id 元素	466	16.2.1	下载二进制	494
15.5.76	renderer 元素	467	16.2.2	下载源代码	495
15.5.77	renderer-class 元素	468	16.2.3	标准核心库	495
15.5.78	renderer-type 元素	469	16.2.4	f:actionListener 标签	497
15.5.79	resource-bundle 元素	470	16.2.5	f:ajax 标签(仅限于 2.0)	498
15.5.80	resource-handler 元素	471	16.2.6	f:attribute 标签	499
15.5.81	source-class 元素	471	16.2.7	f:convertDateTime 标签	500
15.5.82	state-manager 元素	472	16.2.8	f:convertNumber 标签	501
15.5.83	suggested-value 元素	473	16.2.9	f:converter 标签	502
15.5.84	supported-locale 元素	474	16.2.10	f:event 标签(2.0)	503
15.5.85	system-event-class 元素	475	16.2.11	f:facet 标签	504
15.5.86	system-event-listener 元素	476	16.2.12	f:loadBundle 标签	504
15.5.87	system-event-listener- class 元素	477	16.2.13	f:metadata 标签(2.0)	505
15.5.88	tag-handler-delegate- factory 元素	478	16.2.14	f:param 标签	506
15.5.89	to-view-id 元素	478	16.2.15	f:phaseListener 标签 (1.2、2.0)	507
15.5.90	validator 元素	479	16.2.16	f:selectItem 标签	507
15.5.91	validator-class 元素	480	16.2.17	f:selectItems 标签	508
15.5.92	validator-id 元素	481	16.2.18	f:setPropertyActionListener 标签(仅限于 1.2、2.0)	510
15.5.93	value 元素	481	16.2.19	f:subview 标签	510
15.5.94	value-class 元素	483	16.2.20	f:validateBean 标签 (2.0)	512
15.5.95	var 元素	485	16.2.21	f:validateDoubleRange 标签	514
15.5.96	variable-resolver 元素	486	16.2.22	f:validateLength 标签	514
15.5.97	view-declaration-language- factory 元素	486	16.2.23	f:validateLongRange 标签	515
15.5.98	view-handler 元素	487	16.2.24	f:validateRegex 标签 (2.0)	516