

李刚J2EE系列教材

宝典丛书

100万

基于J2EE的 Ajax 宝典

本书并不停留在Ajax层的花哨交互上,而是致力于Ajax与传统J2EE的完美结合

不仅深入介绍了Ajax技术的各种基础知识,还详细介绍了Prototype.js, Dojo, DWR, JSON-RPC-Java和AjaxTags 5个Ajax框架

示例丰富,具有极好的示范性,读者可以拿来就用,轻松完成项目



電子工業出版社
Publishing House of Electronics Industry
<http://www.phei.com.cn>

李 刚 编著

宝 典 丛 书

基于 J2EE 的 Ajax 宝典

李 刚 编著

電子工業出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 简 介

Ajax 技术是目前最流行的技术,它极大地改善了传统 Web 应用的用户体验,因此也被称为对传统 Web 技术的革命。

本书介绍的内容非常全面,覆盖了 Ajax 技术的各个方面,包括 Ajax 技术的核心对象 XMLHttpRequest 对象、JavaScript 脚本的详细知识以及 DOM 和 XML 的相关知识。除了 Ajax 的这些基础知识外,本书还详细介绍了 Ajax 的 5 个相关框架: Prototype.js, Dojo, DWR, JSON-RPC-Java 和 AjaxTags。

本书介绍的不是单纯的 Ajax 技巧,而是侧重于介绍 Ajax 技术与 J2EE 技术的融合,让 Ajax 技术融入实际的 J2EE 应用。

本书适用于有一定 J2EE 编程基础的读者,因为单纯的 Ajax 技术是没有太大用处的。通过阅读本书,不但可以掌握 Ajax 的开发技巧,而且可以掌握如何让 Ajax 技术和 J2EE 应用实现完美融合。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

基于 J2EE 的 Ajax 宝典 / 李刚编著. —北京: 电子工业出版社, 2007.5

(宝典丛书)

ISBN 978-7-121-04258-4

I. 基… II. 李… III. ①计算机网络—程序设计②JAVA 语言—程序设计 IV. TP393.09 TP312

中国版本图书馆 CIP 数据核字 (2007) 第 055806 号

责任编辑: 牛晓丽

印 刷: 北京市天竺颖华印刷厂

装 订: 三河市金马印装有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编: 100036

开 本: 787×1092 1/16 印张: 53.25 字数: 1431 千字

印 次: 2007 年 5 月第 1 次印刷

定 价: 89.00 元 (含光盘一张, ISBN 978-7-900222-56-5)

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zltts@phei.com.cn, 盗版侵权举报请发邮件到 dbqq@phei.com.cn。

服务热线: (010) 88258888。

前言

Ajax 技术是 2006 年最热门的技术，无论说它是对传统 Web 技术的改进还是革命，其实质都是：它具有很强的生命力，确实代表了未来 Web 应用开发的趋势。也许 Ajax 并不是唯一的选择，但 Ajax 的功能是大家有目共睹的。

不过，Ajax 绝不是一门独立的技术，笔者在教学过程中曾遇到一个狂热的学员，他说：我就要学习 Ajax 技术，学好 Ajax 技术就不用担心找不到工作了。笔者不禁哑然：这是一个误区，单纯的 Ajax 技术没有任何用处，绝对没有。说穿了，Ajax 仅仅致力于改善用户体验，也就是说，它是表现层的技术，离开了传统 Web 应用的支持，Ajax 技术便会成为“无本之木，无源之水”。

在动手编写本书之前，笔者仔细研究了市面上介绍 Ajax 的图书，有的图书甚至宣称：本书所介绍的内容是平台无关的——这是一个歧义句：Ajax 技术本身是平台无关的，但任何一本书介绍的内容都不可能是平台无关的。因为 Ajax 技术是表现层的，而底层的技术可以采用任何 Web 编程技术：ASP，PHP，.NET，Java 或 Ruby On Rails，因此 Ajax 技术是平台无关的。但一本书的内容不可能是平台无关的，除非每段代码都提供所有语言的实现版本，这不可能。那些宣称内容平台无关的书，一会儿使用 PHP 作为底层实现，一会使用 Java 作为实现，这变成了对读者的一种折磨。

本书并不是一本单纯的 Ajax 入门书籍，也不打算为介绍 Ajax 而介绍 Ajax。除了详细介绍 Ajax 技术的各方面知识之外，本书还介绍了如何让 Ajax 技术与 J2EE 技术完美融合。换句话说，本书介绍的内容更侧重于让 Ajax 技术融入实际应用开发，而不是满足于表面的 Ajax 技巧。本书所示范的应用，底层都按照 J2EE 规范进行实现；而在用户界面上，Ajax 则大展身手，极大地改善了用户体验。

本书的写作风格与笔者在新东方的授课风格一致：平实、浅显，放弃新名词，把所有深奥的内容变得浅显易懂。笔者不喜欢罗列一堆的新名词，抄大段大段的理论来炫耀自己。作为一个有多年编程经验的程序员，笔者相信“代码就是硬道理”，因此本书中涵盖了 Ajax 的绝大部分实际应用场景。希望读者也不要仅仅“看书”，而一定要参照本书进行动手操作，将每个应用都实际做一遍，相信会有很大的提高。

本书优点

1. 专业性强

本书介绍的不是单纯的 Ajax 技术，而是详细介绍了如何开发 Ajax 应用，如何将 Ajax 技术和 J2EE 应用整合在一起。本书致力于让 Ajax 技术真正融入实际应用的开发，而不是停留在 Ajax 层，为了 Ajax 而 Ajax。通过认真阅读本书，读者不仅可以掌握如何开发 Ajax 应用，而且可以掌握如何让 Ajax 技术和 J2EE 应用实现完美融合。

2. 知识丰富

本书除介绍 Ajax 的各种相关知识（如 JavaScript，DOM 和 XML）之外，还详细介绍了 5 个目前流行的 Ajax 框架：Prototype.js，Dojo，DWR，JSON-RPC-Java 和 AjaxTags，并通过实际示例讲述了如何使用这 5 个框架。

3. 经验丰富，针对性强

笔者既担任过软件开发的技术经理，也担任过软件公司的培训导师，还担任过职业培训的专职讲师，熟悉软件开发的难点，同时也了解软件学习过程中的苦楚。因此，本书尽量针对学习过程中容易遇到的难点进行重点讲解，对开发过程中的“陷阱”给出提示。

4. 示例丰富，实用性强

本书的各个示例针对不同的知识点设计，尽量不与其他知识点掺杂在一起，以免分散读者的注意力。各知识点的示例重点突出，示范性非常强。另外，本书所有的示例都侧重于介绍 Ajax 在实际项目中的使用，例如第 13 章的全功能 Ajax 树，支持在客户端删除、创建和拖动节点，而且这些操作都将通过 Ajax 请求直接持久化到数据库，非常有实用价值。

本书内容

Ajax 的脚本语言 JavaScript，包括 JavaScript 的各种基本知识，还有 JavaScript 的面向对象知识。

XML 语言的相关知识，包括 XML 的文档规则，DTD 和 Schema 的知识，以及如何在 JavaScript 中创建、解析 XML 文档。

CSS 样式单的相关知识，包括 CSS 样式单的选择器，样式单的各种属性，以及如何在脚本中动态修改样式单，从而动态修改 HTML 页面的表现。

DOM 模型的相关知识，包括 HTML 页面和 DOM 模型的关系，以及如何操作 DOM 模型，从而让 HTML 页面内容动态更新。

JavaScript 的事件机制，包括 Internet Explorer 中的事件机制、Netscape 4 中的事件机制和 DOM 2 中的事件机制。

DHTML 模型的相关知识。

XMLHttpRequest 对象的相关知识，包括如何在不同浏览器中创建 XMLHttpRequest 对象，XMLHttpRequest 对象的各种属性、方法，以及如何通过 XMLHttpRequest 对象发送异步请求并获得和处理服务器响应。

Prototype.js 的相关知识，包括 Prototype.js 的扩展和内置函数等。当然，也重点介绍了 Prototype.js 的 Ajax 支持。

Dojo 的相关知识，包括 Dojo 的扩展、内置函数、Dojo 的 Ajax 支持、Dojo 事件机制、Dojo 的拖动功能以及 Dojo 的常用页面控件。

DWR 的各种相关知识，包括 DWR 的创建器、转换器，DWR 框架所生成的 JavaScript 函数库，DWR 和 J2EE 框架的整合，以及 DWR 2.0 的新知识：反向 Ajax。

JSON-RPC-Java 框架的各种相关知识，包括 JSONRPCServlet 和 JSONRPCBridge 等核心类的作用和使用方法，以及如何编写服务器处理类，如何通过 JSONRpcClient 调用服务器的

Java 方法。

AjaxTags 的各种常用标签，以及如何编写 AjaxTags 的各种服务器处理类。

本书介绍的 Ajax 知识虽然也适用于其他 Web 开发平台，但还是希望读者在阅读本书之前具有一定的 J2EE 编程基础；有 J2EE 编程基础的读者可以更好地将 Ajax 技术应用到实际的 J2EE 开发中，切记：仅有 Ajax 技术是做不了任何事情的。

编者
2007.4

目 录

第 1 部分 初识 Ajax	1
第 1 章 Ajax 入门	2
1.1 重新思考 Web 应用	2
1.1.1 应用系统的发展史	2
1.1.2 传统 Web 应用的优势和缺点	4
1.2 重新设计 Web 应用	5
1.2.1 RIA 应用	5
1.2.2 异步发送请求并避免等待	7
1.2.3 使用 Ajax	7
1.3 Ajax 简介	8
1.3.1 Ajax 的工作方式	8
1.3.2 XMLHttpRequest	9
1.3.3 JavaScript 语言	10
1.3.4 HTML 页面的 DOM	10
1.3.5 数据交换和显示	11
1.4 Ajax 的基本特征	12
1.5 Ajax 的替代技术	13
1.5.1 Sun 的 Java Web Start 技术	13
1.5.2 Microsoft 的 ClickOnce 技术	13
1.5.3 基于 Flash 的 Flex	14
1.6 搭建开发运行环境	15
1.6.1 本书的 Ajax 环境	15
1.6.2 Windows 下 JDK 的安装	15
1.6.3 Linux 下 JDK 的安装	19
1.6.4 安装 Tomcat 服务器	20
1.6.5 配置 Tomcat 的服务端口	22
1.6.6 进入控制台	22
1.6.7 部署 Web 应用	24
1.6.8 配置 Tomcat 的数据源	25
1.6.9 安装 Ant 工具	28
1.6.10 安装 Eclipse 开发环境	28
1.6.11 在线安装	29
1.6.12 手动安装	30
1.7 小结	31
第 2 章 Ajax 初体验	32
2.1 Ajax 带来的优势	32
2.2 传统的 JSP 聊天室	34



2.2.1	实现业务逻辑组件.....	35
2.2.2	实现控制器.....	38
2.2.3	实现视图.....	40
2.2.4	JSP 聊天室的问题.....	41
2.3	Ajax 聊天室.....	42
2.3.1	异步发送请求.....	42
2.3.2	解决多余刷新的问题.....	44
2.3.3	解析服务器响应.....	47
2.3.4	何时发送请求.....	48
2.3.5	Ajax 聊天室的特点.....	52
2.4	传统 Web 应用与 Ajax 的对比.....	52
2.5	小结.....	53
第 2 部分 Ajax 基本技术		55
第 3 章 JavaScript 关键语法详解		56
3.1	JavaScript 简介	56
3.2	数据类型和变量	57
3.2.1	定义变量的方式	57
3.2.2	类型转换	58
3.2.3	变量	60
3.3	基本数据类型	62
3.3.1	数字类型	62
3.3.2	字符串类型	67
3.3.3	布尔类型	70
3.3.4	undefined 和 null	71
3.4	复合类型	71
3.4.1	对象	72
3.4.2	数组	72
3.4.3	函数	73
3.5	运算符	75
3.5.1	赋值运算符	75
3.5.2	算术运算符	76
3.5.3	位运算符	78
3.5.4	加强的赋值运算符	79
3.5.5	比较运算符	80
3.5.6	逻辑运算符	81
3.5.7	三目运算符	82
3.5.8	逗号运算符	84
3.5.9	void 运算符	84
3.5.10	typeof 运算符	85
3.6	语句	85
3.6.1	语句块	85
3.6.2	空语句	86
3.6.3	异常抛出语句	86
3.6.4	异常捕捉语句	87
3.6.5	with 语句	88



3.7 流程控制.....	89
3.7.1 分支.....	89
3.7.2 while 循环.....	91
3.7.3 do while 循环.....	92
3.7.4 for 循环.....	93
3.7.5 for in 循环.....	94
3.7.6 break 和 continue.....	95
3.8 函数.....	97
3.8.1 函数定义.....	98
3.8.2 局部变量和局部函数.....	99
3.8.3 匿名函数.....	100
3.8.4 函数的成员属性和静态属性.....	102
3.8.5 递归函数.....	104
3.9 函数的参数传递.....	106
3.9.1 基本类型参数和复合类型参数.....	106
3.9.2 空参数.....	107
3.9.3 参数类型.....	108
3.10 对象.....	109
3.10.1 面向对象的概念.....	109
3.10.2 JavaScript 中的对象.....	110
3.10.3 继承和 prototype.....	110
3.11 创建对象.....	114
3.11.1 使用 new 关键字创建对象.....	115
3.11.2 使用 Object 直接创建对象.....	115
3.11.3 使用 JSON 语法创建对象.....	118
3.12 小结.....	119
第 4 章 XML 详解.....	120
4.1 XML 概述.....	120
4.1.1 标记语言.....	120
4.1.2 XML 的定义和发展史.....	121
4.2 XML 的文档规则.....	122
4.2.1 XML 文档分类.....	122
4.2.2 XML 文档的整体结构.....	124
4.2.3 XML 声明.....	125
4.2.4 XML 元素.....	127
4.2.5 字符数据.....	129
4.2.6 注释.....	131
4.2.7 处理指令.....	132
4.3 XML 数据岛.....	133
4.3.1 数据岛概述.....	133
4.3.2 在 HTML 文档中载入 XML 文档.....	134
4.3.3 将 XML 数据绑定到表格输出.....	134
4.4 DTD 与 Schema.....	137
4.4.1 内部 DTD.....	137
4.4.2 外部 DTD.....	138
4.4.3 公用 DTD.....	139



4.5 DTD 的语法	139
4.5.1 定义元素	140
4.5.2 定义子元素	142
4.5.3 定义元素属性	146
4.5.4 定义实体引用	149
4.6 Schema 的语法	151
4.6.1 Schema 的基本语法	152
4.6.2 定义简单元素	152
4.6.3 定义元素的父子结构关系	153
4.6.4 引用元素	154
4.6.5 Schema 中的内置类型	156
4.6.6 自定义简单数据类型	156
4.6.7 自定义复杂数据类型	160
4.6.8 定义空元素	163
4.6.9 定义混合内容元素	164
4.6.10 定义属性	165
4.6.11 元素组与属性组	166
4.7 使用 JavaScript 解析 DOM	168
4.7.1 DOM 简介	168
4.7.2 解析 XML 文档	169
4.7.3 创建 XML 文档	172
4.8 小结	173
第 5 章 使用 CSS 级联样式单	174
5.1 样式单概述	174
5.2 CSS 样式单的基本使用方式	175
5.2.1 引入外部样式文件	176
5.2.2 使用内部样式定义	177
5.2.3 使用内联样式	179
5.3 使用 CSS 属性	180
5.3.1 文字相关属性	181
5.3.2 整体段落相关属性	182
5.3.3 背景相关属性	184
5.3.4 表格相关属性	186
5.3.5 大小相关属性	189
5.3.6 定位相关属性	189
5.3.7 边框相关属性	191
5.3.8 轮廓相关属性	194
5.3.9 常用属性	194
5.4 选择器定义	196
5.4.1 属性选择器	196
5.4.2 ID 选择器	198
5.4.3 class 选择	199
5.4.4 包含选择器和子元素选择器	199
5.4.5 超链接相关选择器	200
5.5 在脚本中修改显示样式	201
5.5.1 随机改变页面的背景色	201



5.5.2	导航菜单效果	202
5.5.3	卷帘效果	203
5.5.4	动态增加立体效果	205
5.6	小结	206
第 6 章	DOM 模型详解	207
6.1	DOM 模型简介	207
6.1.1	DOM 模型的概念	207
6.1.2	DOM 模型的作用	208
6.2	XML 和 DOM 模型	208
6.2.1	XML 和 DOM	209
6.2.2	使用 DOM 解析 XML 文档	209
6.2.3	使用 SAX 解析 XML 文档	212
6.2.4	使用 DOM 解析器创建 XML	215
6.3	HTML 文档和 DOM 模型	217
6.3.1	HTML 文档简介	217
6.3.2	HTML 文档的 DOM 结构	218
6.4	DOM 中的 HTML 元素	219
6.4.1	HTML 元素之间的继承图	219
6.4.2	HTML 元素之间的常见包含关系	220
6.5	使用 JavaScript 控制 DOM	221
6.5.1	创建节点	221
6.5.2	修改节点	222
6.6	添加节点	224
6.6.1	为下拉列表增加选项	225
6.6.2	动态生成表格	226
6.7	删除节点	227
6.7.1	删除下拉列表的选项	229
6.7.2	删除表格的行或单元格	230
6.8	两个常用范例	232
6.8.1	可编辑表格	232
6.8.2	导航菜单	234
6.9	小结	239
第 7 章	事件处理机制	240
7.1	基本事件模型	240
7.1.1	绑定 HTML 元素属性	240
7.1.2	绑定对象属性	243
7.1.3	使用 addEventListener 方法绑定	246
7.2	深入了解事件模型	247
7.2.1	事件处理函数的范围	247
7.2.2	使用返回值改变默认行为	248
7.2.3	JavaScript 中的 MVC	249
7.2.4	在代码中触发事件	253
7.3	Netscape 4 的事件模型	255
7.3.1	事件对象	255
7.3.2	事件捕捉	257



7.4	Internet Explorer 的事件模型	259
7.4.1	在 Internet Explorer 中绑定事件处理器	259
7.4.2	使用 script for 绑定	259
7.4.3	使用 attachEvent()方法执行绑定	260
7.4.4	事件对象	260
7.4.5	事件传递	264
7.4.6	重定向	266
7.5	DOM 级别 2 的事件模型	268
7.5.1	绑定事件处理函数	269
7.5.2	事件传播和转发	270
7.5.3	取消事件的默认行为	271
7.5.4	控制事件传播	272
7.5.5	转发事件	273
7.5.6	DOM 2 中的事件	274
7.6	小结	275
第 8 章	使用 DHTML 动态操作 HTML 文档	276
8.1	DHTML 和 DOM 两种模型	276
8.2	使用 DHTML	277
8.2.1	DHTML 的包含体系	277
8.2.2	使用 Window 对象	278
8.3	文档对象	284
8.4	表单和表单元素	286
8.4.1	表单和表单元素对象	286
8.4.2	Form 对象的方法和属性	287
8.4.3	Text 和 Textarea	288
8.4.4	Radio 和 ChechBox	289
8.4.5	Select 和 Option	290
8.5	小结	294
第 9 章	XMLHttpRequest 对象详解	295
9.1	XMLHttpRequest 对象概述	295
9.2	XMLHttpRequest 的方法和属性	296
9.2.1	XMLHttpRequest 的方法	296
9.2.2	XMLHttpRequest 的属性	299
9.3	发送请求	304
9.3.1	发送简单请求	304
9.3.2	发送 GET 请求	306
9.3.3	发送 POST 请求	310
9.3.4	发送请求时的编码问题	311
9.3.5	发送 XML 请求	316
9.4	处理服务器响应	318
9.4.1	处理的时机	318
9.4.2	使用文本响应	319
9.4.3	使用 XML 响应	319
9.4.4	使用 DOM 模型生成页面	321
9.5	XMLHttpRequest 对象的运行周期	322



9.6 Ajax 必须解决的问题	322
9.6.1 跨浏览器问题	323
9.6.2 安全问题	324
9.6.3 性能问题	326
9.7 小结	330
第 3 部分 Ajax 常用框架	331
第 10 章 Prototype 框架详解	332
10.1 Prototype 的下载和安装	332
10.1.1 Prototype 概述	332
10.1.2 下载 Prototype	333
10.1.3 安装 Prototype	333
10.2 使用 Prototype 的基本函数	334
10.2.1 使用 \$() 函数	334
10.2.2 使用 \$A() 函数	337
10.2.3 使用 \$F() 函数	339
10.2.4 使用 \$H() 函数	340
10.2.5 使用 \$R() 函数	341
10.2.6 使用 Try.these() 函数	342
10.3 Prototype 的自定义对象和类	343
10.3.1 使用 Element 对象	343
10.3.2 使用 Enumerable 类	346
10.3.3 使用 Field 对象操作表单域	350
10.3.4 使用 Form 对象操作表单	351
10.3.5 使用 Form.Element 对象	353
10.3.6 使用 Hash 对象	354
10.3.7 使用 Class 对象	355
10.3.8 用于操作 HTML 元素的类	356
10.3.9 常用的监听器	357
10.4 Prototype 的常用扩展	359
10.4.1 扩展 Array	359
10.4.2 扩展 Document	361
10.4.3 扩展 String	361
10.4.4 扩展 Event	363
10.5 Prototype 为 Ajax 增加的类	364
10.5.1 使用 Ajax.Request 类	364
10.5.2 使用 Ajax.Responders 对象	368
10.5.3 使用 Ajax 对象	369
10.5.4 使用 Ajax.Updater 类	369
10.5.5 使用 Ajax.PeriodicalUpdater 类	374
10.6 小结	376
第 11 章 基于 Prototype 的应用：实现自动完成功能	377
11.1 应用的基本分析和设计	377
11.1.1 数据要求	377
11.1.2 数据表结构	378



11.2	Domain Object 和持久层	378
11.2.1	Domain Object	378
11.2.2	实现 DAO 组件	381
11.3	实现 Service 组件	385
11.4	使用 Servlet 提供服务器响应	388
11.4.1	实现根据前缀查询品牌	389
11.4.2	实现根据品牌查询型号	390
11.4.3	实现根据型号查询具体描述	391
11.5	客户端 HTML 页面实现	392
11.6	增加 HTML 页面的事件响应能力	394
11.6.1	实现品牌输入框的事件处理函数	395
11.6.2	实现键盘事件处理函数	397
11.6.3	根据品牌提示型号	399
11.6.4	根据型号显示描述	401
11.6.5	注册 Ajax 事件监听器	403
11.7	小结	404
第 12 章 使用 Dojo		405
12.1	初识 Dojo	405
12.1.1	Dojo 概述	405
12.1.2	下载和安装 Dojo	406
12.2	了解 Dojo 的体系	408
12.2.1	通用库	408
12.2.2	Ajax 相关库	409
12.2.3	页面控件相关库	409
12.2.4	其他相关库	409
12.3	使用 Dojo 的通用库	410
12.3.1	使用 djConfig 对象	410
12.3.2	do 对象的函数	412
12.3.3	字符串相关函数	414
12.3.4	扩展函数	416
12.3.5	DOM 相关函数	421
12.4	Dojo 的 Ajax 支持	425
12.4.1	使用 Dojo 开发 Ajax 应用	425
12.4.2	发送请求的几种形式	427
12.4.3	使用 queueBind 函数发送请求队列	431
12.4.4	使用 dojo.io.setIFrameSrc 函数代替 XMLHttpRequest	431
12.4.5	使用 dojo.io.argsFromMap 函数	432
12.5	Dojo 的事件机制	433
12.5.1	简单的 connect 绑定	433
12.5.2	使用对象的方法作为事件处理器	434
12.5.3	为非 DOM 元素绑定事件处理器	435
12.6	基于 connect 的 AOP 实现	436
12.6.1	简单的 before 绑定	437
12.6.2	简单的 around 绑定	437
12.6.3	访问目标方法参数的绑定	439



12.6.4	与 around 绑定在一起使用的监听器	440
12.7	connect 绑定其他函数	441
12.7.1	使用 connectBefore	441
12.7.2	使用 connectAround	441
12.7.3	使用 connectOnce	442
12.7.4	使用 kwConnect	442
12.7.5	使用 disconnect	443
12.7.6	使用 kwDisconnect	443
12.8	基于 Topic 事件机制	443
12.9	Dojo 的拖拉功能	444
12.9.1	自由拖动	445
12.9.2	有相对“坐标”的移动	446
12.9.3	带手柄的移动	448
12.10	Dojo 的常用 widget	449
12.10.1	按钮	449
12.10.2	对话框	451
12.10.3	日历	454
12.10.4	树	457
12.10.5	Tab 页	459
12.11	Dojo 与 Prototype 的对比	461
12.12	小结	461
第 13 章	基于 Dojo 的动态树	462
13.1	自关联 Domain Object 的设计	462
13.2	基于 Hibernate 的 DAO 实现	464
13.2.1	DAO 组件的接口	464
13.2.2	DAO 组件的实现类	465
13.3	基于 Spring 的业务逻辑组件	467
13.3.1	业务逻辑组件的接口	467
13.3.2	业务逻辑组件的实现类	468
13.3.3	配置 Spring 容器	471
13.4	基于 Spring 的集成测试	473
13.5	Dojo 树的相关 Widget	475
13.5.1	使用 TreeContextMenu	475
13.5.2	使用 TreeRPCController	478
13.6	Dojo 树的 HTML 页面实现	479
13.7	实现服务器响应 Servlet	484
13.7.1	实现 getAllRoot	484
13.7.2	实现 updateNode	486
13.7.3	实现 createChild	487
13.7.4	实现 removeNode	489
13.7.5	实现 getChildren	490
13.8	与其他树的对比	491
13.9	小结	493
第 14 章	使用 DWR	494
14.1	DWR 的下载和安装	494



14.1.1	DWR 概述	494
14.1.2	下载和安装 DWR	495
14.2	使用 DWR	500
14.2.1	编写处理类	500
14.2.2	配置 DWR	502
14.3	使用 DWR 的转换器	504
14.3.1	基本转换器	504
14.3.2	对象转换器	504
14.3.3	数组转换器	507
14.3.4	集合类型转换器	508
14.4	方法声明	509
14.5	使用 DWR 的创建器	510
14.5.1	创建器的配置	510
14.5.2	使用 new 创建器	512
14.5.3	使用 none 创建器	512
14.5.4	使用 script 创建器	513
14.6	调用服务器端的方法	514
14.6.1	调用服务器端方法的通用配置	514
14.6.2	使用简单回调	516
14.6.3	使用 JSON 格式的回调	524
14.6.4	将客户端参数传递到回调函数	526
14.7	使用 engine.js	526
14.7.1	设置调用顺序	527
14.7.2	设置全局超时时长	527
14.7.3	设置全局 Hook 函数	527
14.7.4	设置全局处理函数	528
14.7.5	设置常用的全局选项	528
14.7.6	批处理	528
14.8	使用 util.js	529
14.8.1	使用 \$()	530
14.8.2	处理列表	530
14.8.3	处理表格	535
14.8.4	访问 HTML 元素值	540
14.8.5	工具函数	542
14.9	整合第三方 J2EE 框架	545
14.9.1	访问 Servlet API	545
14.9.2	整合 Spring	546
14.9.3	整合 Struts	549
14.10	异常处理	551
14.11	反向 Ajax	553
14.11.1	配置使用反向 Ajax	553
14.11.2	在 Java 方法中操作 Web 页	554
14.11.3	在客户端调用反向 Ajax 方法	556
14.12	小结	557
第 15 章	基于 DWR 的 Ajax 应用: 即时消息系统	558
15.1	实现 Hibernate 持久层	558



15.1.1	Hibernate 持久层的 POJO	558
15.1.2	将 POJO 映射为持久化对象	560
15.2	实现 DAO 组件	562
15.2.1	扩展 HibernateDaoSupport 来实现分页	562
15.2.2	实现 DAO 组件	565
15.3	实现业务逻辑组件	570
15.3.1	业务逻辑组件的接口	570
15.3.2	业务逻辑组件的实现类	572
15.3.3	部署业务逻辑组件	576
15.4	调用业务逻辑组件	578
15.4.1	将 Spring 容器中的 Bean 创建成 JavaScript 对象	578
15.4.2	处理用户登录	580
15.4.3	处理用户注册	582
15.4.4	处理消息发布	583
15.4.5	获取消息列表	585
15.4.6	处理分页	587
15.4.7	查看消息内容	587
15.4.8	页面加载函数	589
15.4.9	客户端 HTML 页面	589
15.5	小结	592
第 16 章	JSON-RPC-Java 框架详解	593
16.1	使用 JSON	593
16.1.1	JSON 的语法	593
16.1.2	发送 JSON 请求	596
16.2	JSON-RPC-Java 的下载和安装	599
16.2.1	JSON-RPC-Java 概述	600
16.2.2	下载和安装 JSON-RPC-Java	600
16.3	使用 JSON-RPC-Java 完成 Ajax 交互	601
16.3.1	映射服务器端 JSONRPCServlet	601
16.3.2	编写处理类	601
16.3.3	使用 JSONRPCBridge	602
16.3.4	在 JSP 中使用 JSONRPCBridge	603
16.3.5	在 Servlet 中使用 JSONRPCBridge	603
16.3.6	暴露处理类对象	604
16.3.7	暴露处理类的实例方法	604
16.3.8	暴露处理类的静态方法	604
16.3.9	调用服务器端方法	605
16.3.10	JSON-RPC-Java 的基本流程	606
16.3.11	基本应用的代码实现	606
16.4	使用 Java 对象作为参数和返回值	611
16.4.1	使用对象作为参数	611
16.4.2	使用对象作为返回值	617
16.4.3	使用集合对象作为返回值	619
16.5	访问 Web 环境	624
16.6	JSON-RPC-Java 与 DWR 的对比	626
16.7	小结	627