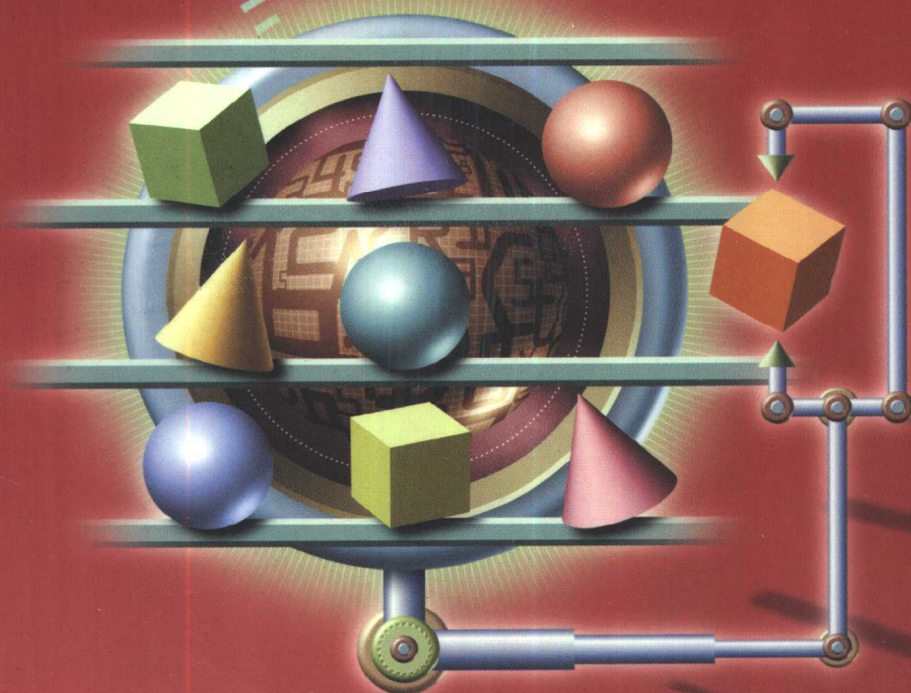


即时应用

ColdFusion 5



- ▶ 开发数据驱动的 Web 应用程序
- ▶ 获得无限的扩展性
- ▶ CD-ROM 中包含全部源代码和数据库



Jeffrey Houser 著
技桥工作室 译

人民邮电出版社
www.pptph.com.cn

麦格劳-希尔教育出版集团
www.mheducation.com

Mc
Graw
Hill

即时应用

ColdFusion 5

Jeffry Houser 著

技桥工作室 译

 人民邮电出版社 麦格劳-希尔教育出版集团 

图书在版编目 (CIP) 数据

即时应用 ColdFusion 5/ (美) 豪泽 (Houser, J.) 著; 技桥工作室译. —北京: 人民邮电出版社, 2002.2

ISBN 7-115-09935-9

I. 即... II. ①豪... ②技... III. 主页制作—应用软件, ColdFusion 5 IV. TP393.092

中国版本图书馆 CIP 数据核字 (2001) 第 088009 号

版 权 声 明

Jeffry Houser

Instant ColdFusion 5

ISBN: 0-07-213238-8

Copyright © 2001 by the McGraw-Hill Companies, Inc.

Original language published by The McGraw-Hill Companies, Inc. All Rights reserved. No part of this publication may be reproduced or distributed in any means, or stored in a database or retrieval system, without the prior written permission of the publisher.

Simplified Chinese translation edition jointly published by McGraw-Hill Education (Asia) Co. and People's Posts & Telecommunications Publishing House.

本书中文简体字翻译版由人民邮电出版社和美国麦格劳-希尔教育(亚洲)出版公司合作出版。未经出版者预先书面许可, 不得以任何方式复制或抄袭本书的任何部分。

本书封底贴有 McGraw-Hill 公司激光防伪标签, 无标签者不得销售。

即时应用 ColdFusion 5

◆ 著 Jeffry Houser
译 技桥工作室
责任编辑 陈冀康

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@pptph.com.cn
网址 <http://www.pptph.com.cn>
读者热线 010-67180876
北京顺义振华印刷厂印刷
新华书店总店北京发行所经销

◆ 开本: 787 × 1092 1/16
印张: 34.5
字数: 1 112 千字 2002 年 2 月第 1 版
印数: 1 - 3 500 册 2002 年 2 月北京第 1 次印刷

著作权合同登记 图字: 01 - 2001 - 3340 号

ISBN 7-115-09935-9/TP·2665

定价: 60.00 元

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

内 容 提 要

本书全面地介绍了用 ColdFusion 5 进行 Web 开发的常用技术。全书分为 4 部分 21 章。第 1 部分为第 1 到 6 章，主要介绍有关后台数据库的知识，为交互应用的实现提供支持。第 2 部分为第 7 到 13 章，介绍了定制站点交互界面的方法。第 3 部分是第 14 到 18 章，介绍了各种网站服务的开发技术以及网站管理的方法。第 4 部分介绍了一些高级数据传送技术。

本书适合于中级和高級的 ColdFusion 开发者阅读，也适合有经验的 Web 开发人员学习 ColdFusion 之用。

译者序

ColdFusion 5 是 Macromedia 公司推出的一个用于建立交互式站点的强大开发工具。利用 ColdFusion，我们可以用最快的速度将服务器、浏览器和数据库技术整合在一起，建立强大的 Web 应用程序。ColdFusion 的优势主要有：

1. 作为一种服务器端语言，ColdFusion 能够实现 Java 和 JavaScript 所无法实现的很多功能。Java 和 JavaScript 是客户端语言，在处理文件和数据库的访问权限时要受到很多的限制。而使用 ColdFusion，您可以轻松地应对文件和数据库的访问。

2. 与 CGI 相比，ColdFusion 可以节省您的服务器资源。CGI 程序与 ColdFusion 程序一样，属于服务器端程序，可以在 Web 上实现广泛的功能，但其致命的弱点是：要占用宝贵的服务器资源；因此在一般的场合它还可以支持系统运行，但在电子商务领域，它不得不让位于 ColdFusion 和 ASP。

3. 与 ASP 相比，ColdFusion 易学易用。同属服务器端程序设计语言，但 ColdFusion 对设计者的要求不太高。它不需要使用传统的编程语言，而只需将标准的 HTML 语言与一种名为 CFML (ColdFusion Markup Language) 的语言结合使用即可。这样，如果您熟悉 HTML 编程，了解数据库实现的知识，就完全可以通过本书掌握 ColdFusion 网站应用程序设计的精髓，从而在很短的时间内成长为一名优秀的网站设计师。

本书是一本供读者快速掌握 ColdFusion 5 的精彩读物。文中提供了大量的应用程序实例，从数据库的设计开始，到整个软件开发的过程，直到完成编写代码的工作。本书对使用 ColdFusion 的广大中、高级开发者来说，是一本难得的好书。

参加本书翻译工作的人员有：常晓波、苏绎鸥、栗庆丰和李亚威等人。苏绎鸥对全书做了初步的统一和校正。马睿倩做了初步的排版工作。在此谨向为本书的出版付出过辛勤劳动的所有人员致以诚挚的感谢！

限于译者水平，错误和不当之处在所难免，欢迎读者不吝指正。

北京技桥工作室

前言

了解 Internet 的历史会有助于了解 ColdFusion 的功能。在此，我首先向您简单介绍一下 Internet 的发展历程和 Web 的创建。我们将讨论 Web 的发展，并讨论像 ColdFusion 这样的程序是如何得以存在的。最后，我们将 ColdFusion 与其他类似的技术相比较，并简要地给出在阅读本书之前大致应该掌握的知识。

HTML 和 Web

Tim Berners-Lee 在 20 世纪 90 年代初创建了万维网，以提供用于共享信息的公共场所。分散在各个节点的信息被组织到一起，任何一个节点可以连接到任何其他节点。如果说 Internet 拥有计算机以及连接这些计算机的网线的有形的的基础，那么 Web 就是信息的集合。

Tim 为 NeXT 系统创建了超文本编辑器，并于 1991 年完成。在之后的 3 年中，他鼓励人们为其他操作系统创建客户端和服务端。Netscape 公司的早期版本浏览器 Mosaic 的代码就是在这个时期内编写而成的。Web 上的用户数量每年增长 10 倍。Tim 在他的原程序中包含了三个定义：

Uniform Document Identifier (UDI) 用于定位网页，并成为众所周知的 URL (Uniform Resource Locators)。

Hyper Text Transfer Protocol (HTTP) 定义如何找到文档。

Hypertext Markup Language (HTML) 定义页面的布局。

Tim Berners-Lee 最后决定成立一个独立的、第三方的组织来帮助制定 Web 上的标准，这个组织就是 World Wide Web Consortium (W3C)，您可以在 www.w3.org 上找到关于 Web 的更多信息。W3C 定义了组成 HTML 的标准标记，现在已经是其第四个版本了。HTML 只是定义了页面的布局，您还需要一个程序将 HTML 的结构解释成为用户所看到的最终页面。这种程序称为 Web 浏览器。当今市场上两个最常用的 Web 浏览器是 Netscape Navigator 和 Microsoft Internet Explorer。

在 Web 上提供更多功能的需要

正如前面所描述的，HTML 是一种用于定义数据的显示方式的语言。它是一种标记语言。它不包括处理数据的标记。在 Web 的早期时代，HTML 还是不错的，因为当时 Web 的用途就是共享信息。起初，Web 主要用于 Internet 学术界。而后，计算机行业开始注意到该项技术。随着时间的推移，Internet 不断发展壮大，吸引了众多商业用户和个人用户。Web 渐渐得以普及，并有了广泛的用户基础。人们开始考虑这项新技术的潜力。

现在，Web 管理员有了新任务。Web 有切实可行的方法来促进经营吗？公司可以在线销售其产品吗？可以为每个用户创建自定义的体验吗？Web 可以帮助商家改善客户支持吗？当然，如果您正在阅读本书，就应该已经知道这些问题的答案是“对”的。

很多新的公司完全基于 Web 进行经营，而传统公司（网民们称之为 brick- and- mortar 公司）也逐渐将经营转移到网络这个虚拟世界中。专业 Web 开发者非常需要将公司的宏伟计

划转变成为现实。虚拟店面、基于 Web 的用户组、以及用户的自定义，这些就是网站要为其顾客提供的一些功能。像 Outpost.com 和 Amazon.com 这样的公司作为 Web 零售商已经是家喻户晓了。传统的零售业公司，如 Barnes & Noble 和 Borders，也已经步入了电子商务。在 Web 世界中新技术更新非常频繁，能够满足新的需求的技术会很快地进入 Web 的领域。JavaScript、Java 和 ColdFusion 就是这些技术中的几个例子，它们离开 Web 和 Internet 是不能存在的。

服务器端语言和客户端语言

所有为扩展 Web 的功能而开发的语言基本上可以分为两大类：服务器端语言和客户端语言。客户端语言在 Web 浏览器中完成它的所有处理，而服务器端语言是在 Web 服务器上运行的。

JavaScript 和 Jscript（Microsoft 的变形）是最常使用的客户端语言。JavaScript 是一种脚本语言，并不是编译语言。当编写 Web 页面的代码时，通常要将未编译的 JavaScript 代码混合到 HTML 中。然后，Web 浏览器相应地解释并执行代码。客户端脚本语言有一个缺点，那就是人人都可以查看和窃取您的代码。脚本语言在访问服务器或客户端计算机时，还会有所限制。然而，对于需要即时修改网页等简单任务而言，脚本语言是非常合适的。

在客户端语言中，有一种称作 Java 的编程语言，它赋予使用者更多能力。Java 是由 Sun 微系统公司开发的。我们可以使用 Java 来编写小程序。小程序是一种设计在浏览器中运行的程序。开发 Java 语言的初衷是想让人们一旦编写了代码，就可以在任何地方运行。Java 是一种半编译语言。不像大多数编程语言要编译成为仅在特定操作系统中可以执行的机器代码，Java 小程序编译成为位代码（bit code）。位代码由浏览器下载，并在下载过程中使用即时（jase-in-time, JIT）编译器而转变为机器代码。这时，代码在被称作 Java 虚拟机的假想计算机中执行。JIT 编译器和 Java 虚拟机是独立于 Java 编程语言而实现的，这样就可以简单地将位代码传递到所需的操作系统中，从而实现以一种语言编写代码，在任何操作系统中都可运行。作为真正的编程语言而非脚本语言，Java 比 JavaScript 具有更多的灵活性和更为强大的功能。如果由用户给定了权限，小程序可以访问用户的本地资源和服务器资源。但是，用户控制着赋予 Java 小程序的权限。如果需要访问客户端系统，您最好了解 Web 浏览器上的设置，并在多控制的环境中使用 Java。在 Internet 上，您不能指望被赋予访问匿名 Web 用户的计算机资源的权限。

客户端语言最大的好处在于把执行代码的职责交给浏览器，这样，负担已经很重的服务器就可以减轻 CPU 的负担了。最大的缺点是功能性的限制。在使用 JavaScript 时，您没有任何到文件系统或数据库的访问连接。您也许可以使用 Java 来访问文件系统或数据库，但是却不可依赖它，因为用户比您更有控制权。

服务器端语言与客户端语言相反。服务器端语言在服务器上执行，并把生成的 HTML 页面返回给用户。当今，有两种不同的方法来执行服务器端语言。一种是编写独立的应用程序，从浏览器开始运行。另一种方法是编写程序，通过 Web 服务器的应用程序编程接口（API）来访问 Web 服务器。当从浏览器上运行独立应用程序时，使用称为公共网关接口（CGI）的规范，该规范是由 W3C 控制的。

最初，所有的服务器端程序都使用 CGI 进行调用。CGI 不是编程语言，但它定义了数据

如何在服务器和客户端之间通过 HTTP 协议进行传送。CGI 程序是服务器端应用程序，可以使用任何能为服务器上的操作系统编译机器代码的编程语言来编写。一般使用 Perl 或者 C 语言来编写 CGI，但使用任何编译语言都可以进行编写。这些程序从 Web 客户端取得数据，处理数据，并将信息返回到 Web 浏览器。通常通过表单将数据输入到浏览器，表单信息通过提交表单传送到服务器。CGI 可以执行任何任务，从简单的发送电子邮件消息到复杂的数据库操作。

当运行 CGI 程序时，您对服务器比在运行客户端代码时具有更强大的控制权。CGI 程序可以访问服务器的文件结构和其他服务器资源，如数据源名称。而其主要的缺点在于，运行 CGI 程序会占用宝贵的服务器资源。您每次由提交表格调用程序时，服务器上的程序被启动。如果站点的通信量很繁重，则经常会出现同一时间运行该程序的多份例程的情况。然而，CGI 程序的功能比客户端脚本有相当程度的提高。ColdFusion 最初是作为 CGI 程序实现的，但随着软件功能的扩展，程序变得越来越大，每次装载到内存时变得越来越困难。这样，服务器 API 就应运而生了。

另一种实现服务器端语言的方法是使用 Web 服务器的应用程序编程接口来编写代码。Microsoft Active Server Pages (ASP) 和 ColdFusion 就是这种实现方法的程序实例。一个解释程序被写入到 Web 服务器的接口。当 Web 服务器试图处理包含特殊 ASP 或 ColdFusion 指令的时候，解释程序执行指令，并将代码执行的结果返回到浏览器。让解释程序持续运行比在每次需要时装载单个程序更为有效。无论网站上有多少个用户，只需运行一个解释程序，这对于每小时都有大量访问者的电子商务站点来说是个很大的优点。

Web 开发者的真正技巧是应该知道每种语言的优势，以及对每种任务应该选用哪种语言。企业级开发者经常需要同时使用两种语言来建立一个功能全面的站点，而一般初学者很难做到这一点。如果要实时改变用户环境的状态，如图像转换和实时表单合法性检验，最好使用客户端脚本。但是，如果需要采用某些安全特性和后台数据库，最好考虑使用在服务器端解决方案。

什么是 ColdFusion，我们为什么要使用它？

J. J. 和 Jeremy Allaire 两兄弟在 1995 年成立了 Allaire 公司。一开始，Jeremy 一直在寻找一种方法，来方便地定期更新一个出版社的 Web 站点。他不想让更新工作因为需要懂得 HTML 或其他技术而犯难。他向搞计算机编程的兄弟 J.J. 寻求帮助，创造一种方法来简化该项工作。当工作完成后，两兄弟认识到它们所创造的技术的潜力，很快 ColdFusion 的第一个正式版本出现了。现在，Allaire 已经与 Macromedia 合并，后者是一家开发 Web 图形软件的公司。

从核心上讲，ColdFusion 是用于数据库到 Web 连接的程序。ColdFusion Markup Language (CFML) 是一种基于标记的语言，最好被描述为是 HTML 的扩展。它是 ColdFusion 的编程语言。ColdFusion 的最初版本是作为 CGI 程序运行的，但是，随着时间的推移和功能的生长，ColdFusion 开始通过服务器的 API 运行。

为什么我们使用像 ColdFusion 这样的程序，而不用 Perl 或 C 来编写自己的自定义 CGI 程序呢？ColdFusion 已经问世 5 年多了，目前是其第 5 个版本。如果使用 Perl 或 C 从草稿开始编写，您将花费时间重复地编写其他人已经编写和测试过的功能。譬如，您要开发一项产

品，它通过许多 Web 开发者和 Macromedia 商业伙伴获得大量的支持。如果使用定制编写的应用程序，一旦开发者遇到难题，您将陷入困境。ColdFusion 就没有这个风险。简单地讲，如果其他人已经为您做了大量的工作，您再从草稿开始编写已经没有任何意义了。您的老板和客户可不想等上 5 年，让您编写一个与 ColdFusion 同样健壮和可缩放的软件。

为什么我们选择在 ColdFusion 中编写应用程序，而不选择其他类似的技术呢（如 Microsoft Active Server Pages）？ColdFusion 是一种基于标记的语言，与 HTML 很相似。对于非编程人员，从创建 HTML 页面转向创建 ColdFusion 模板很容易。Active Server Pages 的语言是 VBScript——一种 Visual Basic 派生语言，对于非编程人员是不推荐使用它的。许多在 ASP 中必须明确地编写的功能是在使用 ColdFusion 的情况下完成的。ASP 模板的常见问题是在查询数据库后忘记关闭数据库，这样用于数据库对象的内存空间就永远不会释放。对于每分钟内有多次点击打开的 Web 页面来说，过多的内存使用会使服务器瘫痪。ColdFusion 则会自动关闭数据库。需要编写的代码较少，这会使 Web 站点的开发和维护更为高效。由于 ColdFusion 的工作是在后台自动完成的，因此它是快速开发和配置的理想工具。ColdFusion 还提供跨平台的环境，可用于各种不同的 Web 服务器和操作系统。而 Microsoft ASP 只适用于 Microsoft Windows NT 服务器和 Microsoft Internet Information Server。

读者指南

本书适用于中级和高级 ColdFusion 开发者。每章带领读者逐步完成应用程序的创建，从数据库的设计开始，到整个软件开发的过程，直到完成编写代码的工作。我假设您已经精通 HTML，而且已经接触过 ColdFusion 开发工作。了解简单的 ColdFusion Markup Language (CFML) 语句，以及 CFML 如何与标准的 HTML 相互联系将会大有裨益。

推荐先掌握一些基础的编程结构知识，如 if 语句和循环结构。您应该非常熟悉结构化查询语言 (SQL)，并能够编写 select、insert、update 和 delete 查询。本书将复习数据库设计方法，但是建议您事先熟悉它们。了解其他的 Web 技术基础知识，如 JavaScript 和 Java，也是很有用的。本书涵盖了如何将 ColdFusion 数据与各种 Java 小程序和 JavaScript 相集成的内容。如果您对这些知识的概念不熟悉，请不必担忧，我们将在开始学习这部分内容时复习这些知识。

我们不会在 Microsoft SQL Server、Sybase SQL Anywhere、Oracle 或 Microsoft Access 中深入探讨数据库实现的细节问题。本书中的应用程序实例将使用 Microsoft Access，因为它的开发很容易。本书将不涉及服务器管理问题。如果要练习本书中的实例（我们强烈推荐您这样做），您需要在计算机上安装 ColdFusion 和 Web 服务器，或者访问装有 ColdFusion 的计算机。尽管不是必需的，但您最好在阅读本书之前有些实际编程工作的经验。

在随书的 CD-ROM 中的相应章节目录下有为本书配备的文件。当在某章中引用一个文件时，将会有有一个 CD-ROM 图标为您指出文件目录在章节文件夹中的具体位置。

本书中所使用的命名规则非常简单易懂。HTML 和 CFML 标记都是大写字母。变量作用域，如表单、会话或应用程序等，都是小写字母。变量名、文件名、目录和数据库表格名混合使用大写和小写字母。例如，如果数据库表格名是由两个单词组合而成，那么每个单词的第一个字母将是写大的，而其他字母是小写的。

现在，让我们从头开始学习这本书。

目 录

第 1 部分 实现 Internet 的交互性

第 1 章 设计应用程序时需要考虑的一般事项	3
1.1 ColdFusion 所用的后台数据库	3
1.1.1 Microsoft Access	3
1.1.2 客户端/服务器数据库	5
1.2 ColdFusion 模板的结构	6
1.2.1 如何编写好的文档头	7
1.2.2 模板的处理区段	9
1.3 变 量	10
1.3.1 变量类型	10
1.3.2 变量类别	11
1.4 Application.cfm 文件	12
1.4.1 应用程序作用域	15
1.4.2 Cookie 并不可怕	16
1.4.3 标准变量	17
1.4.4 错误处理	18
1.5 本章所学的内容	24
第 2 章 构建 Product 区段	25
2.1 规划应用程序	25
2.1.1 一对一关系	26
2.1.2 一对多关系	27
2.1.3 关于数据建模的一些最后说明	28
2.2 浏览 Product 区段	29
2.2.1 SQL select 语句	30
2.2.2 Application.cfm	31
2.2.3 我们的产品主页	32
2.2.4 产品列表	35
2.2.5 我们的产品描述页面	37
2.3 本章所学的内容	41
第 3 章 动态 Web 站点导航	42
3.1 代码模块化	42
3.1.1 副作用	43
3.1.2 CFINCLUDE 语句	43

3.1.3 自定义标记	47
3.2 使用包含文件来改进我们的 Web 站点	50
3.2.1 重建 Application.cfm	50
3.2.2 创建动态导航栏	55
3.3 本章所学的内容	58
第 4 章 通过数据库生成所需信息	59
4.1 自引用表格	59
4.1.1 表格结构	60
4.1.2 导航结构	61
4.1.3 数据库驱动部分	61
4.2 创建导航栏	63
4.3 基于数据库驱动的内容	67
4.3.1 主页的默认部分	67
4.3.2 创建 News 部分	70
4.3.3 Support 部分	78
4.4 本章所学的内容	81
第 5 章 用户调查	82
5.1 组织数据	82
5.2 创建调查	83
5.2.1 提出调查问题	83
5.2.2 处理调查结果	87
5.2.3 调查存档	92
5.3 本章所学的内容	94
第 6 章 为站点添加搜索引擎	95
6.1 搜索产品	95
6.2 创建全文搜索	102
6.2.1 创建一个 Verity 集合	102
6.2.2 填充集合	103
6.2.3 对查询结果集建立索引	105
6.3 创建一个 Verity 搜索界面	108
6.4 本章所学的内容	114

第 2 部分 创建自定义的用户体验

第 7 章 创建用户注册表单	119
7.1 我们需要收集什么样的用户信息	119

7.2 创建注册表单	121
7.2.1 注册表单	121
7.2.2 注册处理	127
7.2.3 CFINSERT 标记	132
7.3 本章所学的内容	132
第 8 章 为 ColdFusion 应用程序添加安全性	134
8.1 创建安全性方案	134
8.1.1 安全性的普通实现	134
8.1.2 数据结构	135
8.2 数据项	138
8.3 创建安全管理员	142
8.3.1 命名规则	142
8.3.2 维护组	142
8.3.3 用户维护	150
8.3.4 关联组和用户	171
8.3.5 维护组对数据的访问	177
8.4 本章所学的内容	185
第 9 章 登录和注销	186
9.1 创建登录页面	186
9.2 再访 Application.cfm	193
9.2.1 Internet Application.cfm	193
9.2.2 Admin Application.cfm	195
9.3 改写的注册处理	196
9.4 再访导航	202
9.5 注 销	206
9.6 本章所学的内容	207
第 10 章 ColdFusion 和 Java	208
10.1 关于 Java 的一点知识	208
10.2 滚动标题小程序	208
10.3 创建 Java 导航	212
10.4 本章所学的内容	222
第 11 章 增强顾客服务部分	223
11.1 表格结构	223
11.2 再访用户注册	224
11.2.1 注册用户自定义标记	224

11.2.2 处理注册页面	230
11.2.3 更新用户自定义标记	235
11.3 报告问题	238
11.3.1 报告问题的输入页面	238
11.3.2 报告问题的处理页面	248
11.4 表单合法性检验	254
11.4.1 使用 JavaScript 的客户端合法性检验	255
11.4.2 使用 ColdFusion 的服务器端合法性检验	258
11.5 本章所学的内容	272
第 12 章 添加讨论组	273
12.1 组织数据	273
12.2 讨论组主页	274
12.3 创建一个新的讨论主题	285
12.4 创建主题的响应	289
12.5 本章所学的内容	294
第 13 章 创建用户配置文件	295
13.1 用户配置文件页面	295
13.1.1 数据结构	295
13.1.2 用户配置文件输入页面	299
13.1.3 用户配置文件验证页面	310
13.1.4 用户配置文件处理页面	326
13.2 用数据填充主页	330
13.3 本章所学的内容	335
第 3 部分 在 ColdFusion 中创建 Intranet 站点	
第 14 章 文档共享	339
14.1 变更数据库	339
14.2 添加新文档	340
14.2.1 文档输入页面	341
14.2.2 文档处理页面	344
14.3 浏览文档	348
14.4 本章所学的内容	351
第 15 章 创建公司日历	353
15.1 组织数据	353
15.1.1 Calendar 表格	353

15.1.2 一些数据项	354
15.2 创建日历项	355
15.2.1 输入页面	355
15.2.2 处理页面	360
15.3 显示日历	364
15.3.1 算法	364
15.3.2 显示页面	365
15.3.3 项细节页面	372
15.4 本章所学的内容	375
第 16 章 创建地址簿	376
16.1 组织数据	376
16.2 组的维护	377
16.2.1 组输入页面	377
16.2.2 组处理页面	379
16.3 创建地址簿项	380
16.3.1 创建地址簿项的输入页面	381
16.3.2 验证地址簿项	388
16.3.3 处理新项	399
16.4 给用户分组	410
16.4.1 用户组输入页面	411
16.4.2 用户组处理页面	413
16.5 显示地址簿项	415
16.5.1 组列表	415
16.5.2 用户列表页面	416
16.5.3 用户详细资料页面	420
16.6 本章所学的内容	422
第 17 章 为 Internet 站点创建管理部分	424
17.1 导航栏	424
17.1.1 导航维护主页	424
17.1.2 增加新导航项	426
17.1.3 编辑导航项	434
17.1.4 删除导航项	446
17.2 维护 What's New 区	447
17.2.1 What's New 维护主页	447
17.2.2 在 What's New 中创建新文档	449
17.2.3 编辑新闻发布	456
17.2.4 删除新闻发布	467

17.3 本章所学的内容	468
第 18 章 创建用户报告	470
18.1 组织数据	470
18.2 在用户上网浏览时跟踪用户	471
18.3 创建用户报告	472
18.3.1 显示用户列表	472
18.3.2 用户日期列表	480
18.3.3 在指定日期的用户详细资料	482
18.4 导航报告	484
18.4.1 导航列表	484
18.4.2 导航日期列表	486
18.4.3 导航日期的详细资料	488
18.5 本章所学的内容	490

第 4 部分 高级数据传送

第 19 章 高级错误检查	493
19.1 处理运行时的异常	493
19.1.1 捕获数据库错误	493
19.1.2 生成自定义错误	495
19.2 ColdFusion 日志	501
19.2.1 CFLOG	501
19.2.2 在 ColdFusion Administrator 里浏览日志	502
19.3 本章所学的内容	503
第 20 章 高级查询	504
20.1 缓存州列表	504
20.2 重访讨论组	507
20.2.1 在讨论组主页上显示总数	507
20.2.2 主题列表	511
20.3 将两种查询联合使用	516
20.4 本章所学的内容	523
第 21 章 使用 WDDX 和 XML 共享数据	524
21.1 共享 Web 数据	524
21.2 使用他人的数据	528
21.2.1 获取新闻供给	528
21.2.2 在站点中集成新闻供给	529
21.3 本章所学的内容	536

第 1 部分

实现 Internet 的 交互性

目标:

- ▶ 理解可以与 ColdFusion 交互作用的各种类型的数据库。
- ▶ 学习一些基本的编程指南。
- ▶ 回顾数据库的基本理论。
- ▶ 探究代码模块化的优点。
- ▶ 创建动态导航栏。
- ▶ 在主页上添加用户调查。
- ▶ 创建调查档案。
- ▶ 通过查询或者 Verity 搜索实现搜索功能。

