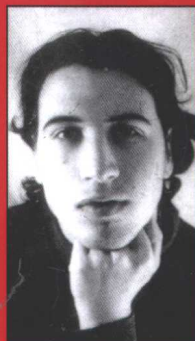
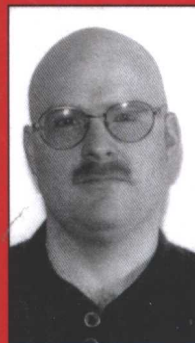
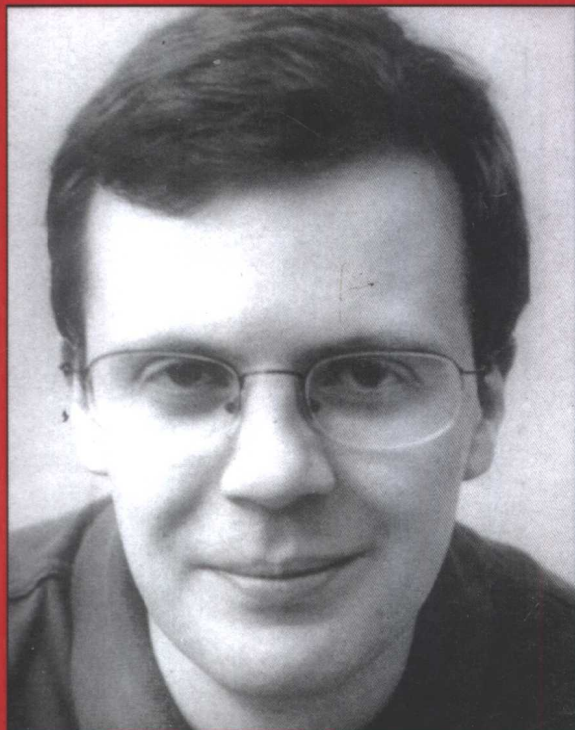




PROGRAMMER TO PROGRAMMER™

Professional Perl
Programming

Wrox 程序员参考系列



Perl

高级编程

(英) Peter Wainwright 等著

胡季红 王璐珍 丁滢 等译

p2p.wrox.com 提供在线支持



机械工业出版社
China Machine Press



乐思

Wrox 程序员参考系列

Perl 高级编程

(英) Peter Wainwright 等著

胡季红 王璐珍 丁 滢 等译



机械工业出版社
China Machine Press

本书深入浅出地介绍了利用 Perl 语言进行编程的各方面知识。首先, 本书介绍了大多数高级程序设计语言中都涉及的数据类型、特殊变量、操作符、表达式、语句和子例程等基本概念。然后, 本书逐章阐述了文件与目录处理、终端输入与输出、程序调试方法等。最后, 本书还描述了面向对象程序设计方法、Perl 解释器、进程、线程、基本的网络编程方法、Perl 语言的可移植性以及国际化等高级话题。

本书思路清晰, 逻辑性强, 方法实用性强, 是进入 Perl 编程世界的理想向导和指南。无论是 Perl 入门者, 还是有经验的 Perl 程序员, 都可得益于本书提供的丰富的 Perl 编程知识与编程技巧。

Peter Wainwright, et al: Professional Perl Programming.

Authorized translation from the English language edition published by Wrox Press.

Original copyright © 2001 by Wrox Press. All rights reserved.

Chinese simplified language edition published by China Machine Press.

本书中文简体字版由英国乐思出版公司授权机械工业出版社独家出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

版权所有, 侵权必究。

本书版权登记号: 图字: 01 - 2001 - 4401

图书在版编目 (CIP) 数据

Perl 高级编程/ (英) 温莱特 (Wainwright, P.) 等著; 胡季红等译. - 北京: 机械工业出版社, 2002.6

(Wrox 程序员参考系列)

书名原文: Professional Perl Programming

ISBN 7 - 111 - 10140 - 5

I. P... II. ①温 ... ②胡 ... III. Perl 语言 - 程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2002) 第 021165 号

机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码 100037)

责任编辑: 贾 梅

北京第二外国语学院印刷厂印刷 · 新华书店北京发行所发行

2002 年 6 月第 1 版第 1 次印刷

787mm × 1092mm 1/16 · 67.25 印张

印数: 0 001 - 4 000 册

定价: 108.00 元

凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换

前言

灵活的、可移植的、多用途的并且可以免费获得的 Perl，已经从一种 shell 脚本的简单替代物发展成为一种功能强大的面向对象的通用语言。随着每次新版本的发布，Perl 都添加了新的特性，因此这种非常流行、具有丰富特性的语言越来越普及。Perl 的一种独特的功能强大的特性是其模块库的实现，该实现使 Perl 成为了一种真正意义上的可扩充语言。

本书思路清晰，结构逻辑性强，方法实用，是进入 Perl 编程世界的理想向导和指南。

本书面向的读者

本书的覆盖面和深度使其成为一本对 Perl 的入门者和有经验的 Perl 程序员都极具帮助的图书。本书为新入门者提供了内容广泛的指导，它从这种语言的基础开始，并且不需要任何 Perl 以前的知识。本书提供了对这种语言从 shell 脚本发展到面向对象过程中很多方面的透彻理解。但是因为指导包含了很多细节内容，所以即使有经验的 Perl 程序员也会发现许多值得深入学习的地方。

本书是由经验丰富的程序员所写的，他们每个人都总结了自己使用 Perl 的经验，提供了每个人的专门知识和见解。因此本书的特点是专业的开发人员在分享其艰辛获得的知识。采用 Wrox 惯例，本书不仅给出了原始的（并且免费的）由这种语言提供的文档，而且对所涉及到的每一个主题都给出了一种重实效的方法并给出了实例。

对于 Perl 开发者而言，无论是否了解 Perl，都能从本书中发现许多新的内容。

本书的覆盖范围

这里对本书的内容做一个简略的介绍。

第 1 章先对 Perl 做概括的介绍。我们首先看看如何在 UNIX、Windows 和 Macintosh 系统上安装 Perl 的二进制发布版本和源代码发布版本，然后介绍安装 Perl 模块和第三方程序库，最后概述 Perl 命令行选项和环境变量。

在深入研究 Perl 的细节之前，第 2 章讲述基础，简要介绍这种语言的一些基本概念：数据类型、特殊变量、操作符和表达式等等。这样可以使 Perl 的入门者在我们更详细地研究这些概念之前能够熟悉它们。

在第 3 章中，我们分析 Perl 中首要的和最基本的数据类型：标量。我们首先了解了标量的三种主要类型——整数、浮点数和字符串以及它们之间互相进行转换的方式。接下来，我们分析能被用于处理字符串的函数。最后我们讨论用于字符串转换的低级函数。

使用标量需要操作符。第 4 章介绍和回顾 Perl 中所有操作符的用法，并且介绍优先级和结合性的相关概念以及在 Perl 中应该如何使用它们。对于更高级的读者，我们还介绍了如何禁用和覆盖操作符。

译者序

Perl 是 Practical Extraction and Report Language 的缩写，最初是由 Larry Wall 设计的。随着网络技术的迅猛发展，Perl 这种可用来处理文本并具有脚本语言特点、功能十分强大的高级程序设计语言成为了最常用的网络编程语言之一。

本书深入浅出地描述了利用 Perl 语言进行编程的各个方面。首先，本书介绍了大多数高级程序设计语言中都涉及的数据类型、特殊变量、操作符、表达式、语句和子例程等基本概念。然后，本书说明了文件与目录处理、终端输入与输出、程序调试方法等。最后，本书还描述了面向对象程序设计方法、Perl 解释器、创建与管理进程以及进程间通信、线程、基本的网络编程方法、Perl 语言的可移植性以及国际化等高级话题。

全书既有广泛的覆盖面，表现在对 Perl 编程的各个方面都进行了介绍；又有非常深入的关于专业 Perl 编程知识与编程技巧的讨论，表现在本书不仅仅限于介绍基本的 Perl 编程，而且深入描述了 Perl 程序调试、Perl 解释器等非常高级的 Perl 程序设计方法。因此，本书不仅适合于 Perl 入门者，而且是有经验的 Perl 程序员必备的一本 Perl 编程参考书。

本书是由经验丰富的程序员编写的，他们每个人都在自己多年从事 Perl 编程经验的基础上；提炼出各自的专业知识和见解。可以说，本书是专业开发人员为分享其宝贵的编程经验而推出的力作。本书采用 Wrox 惯例，不仅给出了原始的（并且免费的）由这种语言提供的文档，而且对所涉及到的每一个主题都给出了一种注重实效的基于例子的方法。

本书由胡季红、王璐珍、丁滢、汪任伟、畅斌、易会战、刘明、郑笛、王俊、周婷和李化进行翻译，全书最后由肖国尊统稿，辰卓工作室全体人员参加了本书的翻译、校对、录入和排版工作。在翻译过程中，我们查阅了大量资料，力求准确、详实而全面地表现出原著者的初衷。但由于水平与时间有限，错误之处在所难免，望各界同仁批评指正。

2001 年 11 月

第 5 章介绍更多的 Perl 数据类型，主要是数组、hash 和 typeglob。这一章还介绍了两类特殊的标量：引用和未定义值。我们将看到如何操作这些数据类型以及如何用它们来创建复杂的数据结构，例如多维数组和 hash 的 hash。这一章的最后一节讲述常量，包括如何声明标量常量、列表常量和 hash 常量。

第 6 章建立在前面几章的基础上，介绍 Perl 的编程结构。首先是表达式和语句，然后是条件语句，最后是循环和循环结构。

子例程是代码的可重用块，它们使脚本更容易维护和被改写。第 7 章示范如何声明参数和向子例程传递参数。这一章还介绍了如何用原型来指明子例程应该采用何种类型的参数和子例程如何返回值。这一章围绕子例程的属性做更进一步的讨论，还介绍了如何用这些属性来修改子例程的行为。

对理解变量和代码如何相互作用而言，作用域和可见性这两个概念是必不可少的。因为 Perl 既支持“包作用域”（也叫做“动态作用域”），也支持“词法作用域”，所以它的作用域规则看上去很复杂。第 8 章详细地解释和对比这两种不同的作用域。这一章还介绍了符号表和名字空间的相关概念。

这种语言的一个功能强大的特性是 Perl 模块库，它使 Perl 成为真正的可扩充语言。第 9 章举例说明使用模块的几种不同方式，并解释改变 Perl 编译器行为的语用模块。我们还讲述了两种特殊的变量：@INC 和 %INC 以及它们如何与模块使用发生联系。另外，我们展示如何检查模块的存在及其可用性，还给出一些方法，用来延迟模块的装入和编译，除非它们真正被使用或者直到它们真正被使用。

从外部了解了模块之后，第 10 章开始介绍如何从内部来理解它们——深入它们的实现。我们首先处理基本的模块结构和特殊的程序块 BEGIN、END 及友元。接着我们介绍自动加载并讨论各种不同的自动加载技术和策略，包括子例程的定义。然后，我们详细讨论在不同包之间的导入和导出子例程。在最后一节，我们讲述如何用 h2xs 工具创建可安装的模块包。

第 11 章的主题是 Perl 的正则表达式引擎，这个特性是使用这种语言的一个最常见原因。我们先介绍插值，它与正则表达式共用一些相同的概念，而且可以被用来单独地或者与正则表达式结合在一起产生作用。介绍完插值以后，这一章给出一个详尽的关于正则表达式的指导，包括从基本概念到高级技术的内容，我们将讨论正则表达式模式的定义和结构，以及它们的赋值。

第 12 章介绍另一种特有的数据类型：文件句柄。我们将学习创建文件句柄的几种方法，并学习如何读写它们，然后讨论锁定文件、随机存取和缺省输出文件句柄。接下来我们学习在系统级上处理文件句柄。在系统级上我们可以更高程度地控制这些文件句柄，但是要担负更大的责任。

学习完文件句柄以后，第 13 章讲述文件名，包括文件权限和所有权。我们先介绍一元文件测试操作符，并学习如何改变文件属性及通过文件系统在文件上执行操作。然后，我们看看 Perl 的 glob 操作符以及临时文件的创建与处理。该章的最后一部分集中讲述目录，介绍目录的读取、创建和删除。

第 14 章讲述命令行参数的语法分析和处理。首先我们详细介绍一个特殊的数组 @ARGV，

当启动一个 Perl 程序时，这个数组包含传递给这个程序的参数。接下来，这一章将阐述命令行处理模块 `Getopt::Std` 和 `Getopt::Long`。这一章的第二部分示范如何创建简单的 Perl shell 以及如何把这些底层 shell 集成到我们自己的 Perl 脚本中。

第 15 章讨论程序设计中的终端输入和输出。我们介绍几种输出到终端的不同方法，包括使用 `Term::ReadKey` 模块和 `Term::ReadLine` 模块。接下来，这一章将介绍如何输出到屏幕上，并介绍高级终端模块。最后，介绍由模块 `POSIX::Termios` 提供的终端低级接口。

在第 16 章中，我们介绍警告和错误，包括警告的生成、发现错误和如何处理它们。我们学习如何解释来自系统调用的错误结果。有时我们需要使非致命的错误变成致命的，`Fatal` 模块向我们展示如何做到这一点。这一章还介绍了 `Carp` 模块以及如何使用底层系统和 Perl 函数记录错误。

第 17 章讨论调试方法。这一章详细展示如何使用 Perl 的调试器，并介绍了许多调试模块，这些模块都是 Perl 标准库的一部分。然后，这一章展示如何实现自动测试，接下来解释 profiling——为应用程序收集计时统计信息的过程。

第 18 章分为主要的三节，第一节阐述 Perl 提供的用于处理文本的各种模块。这一节介绍的模块有用于语法分析单词与短语的 `Text::ParseWords` 和用于格式化段落的 `Text::Wrap`。第二节介绍 Perl 的简易老式文档（Plain Old Documentation, POD）。最后一节展示如何使用被称做格式化的特殊的布局定义生成格式化文本。

第 19 章的主题是面向对象的程序设计。在深入讨论使用 Perl 的对象进行程序设计之前，这一章首先简要介绍对象的概念。这一章向我们展示如何编写对象类并详细介绍了继承和派生类，还介绍了保持数据私有、销毁对象以及操作符重载。最后一节讨论 Perl 的一个特殊功能部件：捆绑对象和被捆绑对象。

第 20 章结束对 Perl 的泛泛之谈，带领我们深入到 Perl 的解释器中。这一章讲述 Perl 是如何被构建的、`Config.pm` 模块、Perl 使用的内部数据类型和 Perl 编译器。在后面一节，我们研究 `O` 模块和 `B::Module` 家族的工作原理。

在第 21 章中，我们看看 Perl 和其他语言的集成。首先我们看看如何在 Perl 中使用 C，在接口 XS 上可知道细节。接着我们学习如何访问一个动态库中的代码，如果这个动态库是一个 UNIX 共享对象或者一个 Windows 动态链接库。然后我们看看如何在 C 中使用 Perl。最后，我们看看 `Java - Perl Lingo (JPL)` 以及在一个 COM 环境中使用 Perl。

第 22 章的主题是创建进程和管理进程。首先是对信号和信号处理的介绍，接着我们展示如何在两个进程之间操作和通信，以及如何共享数据。最后我们看看 Perl 的一个比较新的特性：线程。

第 23 章先介绍连网的概念，然后介绍套接字和基本的网络编程。接下来介绍 TCP 连网和 UDP 连网，还有 UNIX Domain 套接字。这一章还讲述了套接字配置选项、多路复用服务器、轮询、分叉和实现一个线程服务器。在最后一节，我们看看如何获取有关网络主机、网络、协议和服务的信息。

Perl 是一种高度可移植的语言。它可以在所有的大型机和大多数小型机的操作系统上运行。但是虽然大多数 Perl 代码都是可移植的，我们仍然需要注意平台之间的不同之处和易犯的

错误。如果我们准备编写真正可移植的 Perl 代码，那么我们需要牢牢记住第 24 章中强调的要点。

第 25 章讲述 Perl 对 Unicode 的支持。这一章首先介绍 Unicode 的基本概念，接着讨论 Perl 的 Unicode 对正则表达式的支持，然后我们看看双向文本是如何被翻译的。最后这一章示范如何实现 Perl 对 Unicode 的支持。

前面一章研究 Unicode，接下来的一章讨论地区和国际化方案。第 26 章解释 Perl 对不同语言的支持和如何使用这种支持。然后这一章介绍许多和语言有关的模块以及如何设计多语言网页。

这本书的最后是一些附录，都是有用的参考资料，包括 Perl 的命令行选项、特殊变量、函数和标准库模块。

使用本书需要的基础知识

虽然许多平台都支持 Perl，但是最广泛的支持来自于 UNIX（还有 Linux 和 BSD）以及微软的 Windows 9x、2000 和 NT。这本书的一些章节中包含了特定平台的例子，不会同时被应用到 UNIX 和 Windows 上。所有例子使用的都是 Perl 5.6，虽然可以使用 Perl 5.003 以后的任意一个版本，但是安装 Perl 5.6 版本更可取。在整本书的不同地方，我们将需要分别安装 Perl 的不同模块，所有这些模块都可以从 CPAN 上获得。

虽然不做要求，但是在处理某些主题时，知道一些 UNIX 或者 Linux 的知识会有帮助。

源代码

我们已尽可能地提供了能最有效说明文章中所讨论概念的示范程序和代码片段，书中的全部源代码都可以从下面这个网站中下载：

<http://www.wrox.com>

按照 GNU 公共版权（GNU Public License）的条款，这些代码是可以获得的。虽然所有需要的代码基本上都在书中列了出来，我们仍然建议读者下载一份拷贝，这样可以节省大量的录入时间。

目 录

译者序		
前言		
第 1 章 概览	1	
1.1 简介	1	
1.1.1 关键的特性	2	
1.1.2 支持平台	3	
1.1.3 Perl 的历史和版本	3	
1.1.4 基础信息	4	
1.2 构建和安装 Perl	5	
1.2.1 安装预构建的 Perl 发布版本	6	
1.2.2 构建 Perl 源代码	8	
1.2.3 构建和安装模块	12	
1.2.4 用 CPAN 模块安装模块	12	
1.3 运行 Perl	21	
1.3.1 启动 Perl 应用程序	21	
1.3.2 命令行	23	
1.3.3 命令行语法	24	
1.3.4 Perl 环境	26	
1.4 小结	28	
第 2 章 基本概念	29	
2.1 值和变量	29	
2.2 空白区域	30	
2.3 数据类型	30	
2.4 特殊变量	32	
2.5 插值	33	
2.6 上下文	34	
2.6.1 标量上下文和列表上下文	34	
2.6.2 空白上下文	35	
2.7 操作符	35	
2.8 程序块	36	
2.9 控制结构	36	
2.10 循环修饰符	38	
2.11 子例程	39	
2.12 函数	40	
2.13 作用域	40	
第 3 章 标量	42	
3.1 值转换和高速缓存	42	
3.2 数字	43	
3.2.1 整数	43	
3.2.2 浮点数	46	
3.2.3 use integer 编译指示	50	
3.2.4 数学函数	51	
3.3 字符串	52	
3.3.1 引号和引用	52	
3.3.2 here 文档	54	
3.3.3 裸词字符串和版本号	56	
3.3.4 把字符串转换成数字	58	
3.3.5 把字符串转换成列表和 hash	58	
3.3.6 操作字符串的函数	60	
3.3.7 字符串格式化	69	
3.4 小结	72	
第 4 章 操作符	73	
4.1 操作符与函数比较	73	
4.2 操作符类型和类别	74	
4.2.1 赋值操作符	74	
4.2.2 算术操作符	75	
4.2.3 移位操作符	76	
4.2.4 字符串操作符和列表操作符	77	
4.2.5 逻辑操作符	78	
4.2.6 位操作符	79	
4.2.7 组合赋值操作符	82	
4.2.8 递增操作符和递减操作符	83	
4.2.9 比较操作符	84	
4.2.10 正则表达式绑定操作符	86	
4.2.11 逗号操作符和关系操作符	86	
4.2.12 引用操作符和间接引用操作符	87	
4.2.13 箭头操作符	87	
4.2.14 范围操作符	88	
4.2.15 三元操作符	90	
4.3 优先级和结合性	90	

4.4 禁用函数和操作符	95	6.1.1 声明	155
4.5 覆盖操作符	97	6.1.2 表达式和简单语句	156
4.6 小结	98	6.1.3 程序块和复合语句	157
第 5 章 标量之外的更多数据类型	99	6.2 条件语句	162
5.1 列表和数组	99	6.2.1 真值的概念	163
5.1.1 操作数组	101	6.2.2 if、else 和 elsif	164
5.1.2 把列表和数组转换为标量	108	6.2.3 unless	166
5.2 hash	110	6.2.4 使用逻辑操作符写条件	167
5.2.1 操作 hash	112	6.2.5 三元操作符	168
5.2.2 把列表和数组转换为 hash	114	6.2.6 开关和多分支条件	171
5.2.3 把 hash 转换为标量	118	6.2.7 从多分支条件中返回值	173
5.2.4 把 hash 转换为数组	119	6.3 循环和循环过程	174
5.2.5 特殊的 hash 变量 %ENV	120	6.3.1 使用 for 编写 C 样式的循环	174
5.3 引用	124	6.3.2 使用 foreach 写出更好的循环	175
5.4 复杂的数据结构	132	6.3.3 条件循环——while、until 和 do	179
5.4.1 嵌套带来的问题——列表被展开	132	6.3.4 控制循环执行	183
5.4.2 列表的列表和多维数组	133	6.3.5 goto 语句	187
5.4.3 hash 的 hash 和其他复杂的 数据结构	134	6.3.6 map 和 grep	189
5.4.4 向复杂的数据结构添加元素和 修改复杂的数据结构	135	6.4 小结	191
5.4.5 编程实现复杂的数据结构	136	第 7 章 子例程	192
5.4.6 遍历复杂的数据结构	139	7.1 声明子例程和调用子例程	192
5.5 typglob	142	7.1.1 匿名子例程和子例程引用	193
5.5.1 定义 typglob	142	7.1.2 严格子例程和 use strict subs 编译指示	195
5.5.2 操作 typglob	143	7.1.3 预声明子例程	196
5.5.3 访问 typglob	144	7.1.4 子例程栈	198
5.6 未定义值	145	7.1.5 检查子例程和使 Perl 自动 定义子例程	202
5.6.1 存在测试	147	7.2 传递参数	204
5.6.2 使用未定义值	148	7.2.1 传递列表和 hash	206
5.6.3 把 undef 作为一个函数使用	149	7.2.2 把标量子例程转换为列表 处理程序	208
5.7 常量	150	7.2.3 直接把 @_ 传递给子例程	208
5.7.1 使用 constant 编译指示声明 标量常量	150	7.2.4 命名参数	209
5.7.2 声明列表常量和 hash 常量	152	7.3 原型	212
5.7.3 常量引用	152	7.3.1 定义参数的个数和它们的 作用域	213
5.7.4 列举和检查常量的存在	153	7.3.2 使用原型设计代码引用	215
5.8 小结	153	7.3.3 把子例程作为标量操作符	215
第 6 章 结构、流和控制	155	7.3.4 请求变量而不是值	216
6.1 表达式、语句和程序块	155		

7.3.5 可选参数	218	9.3.1 语用模块的工作原理	253
7.3.6 禁用原型	218	9.3.2 语用模块的作用域	254
7.4 从子例程中返回值	219	9.4 特殊 hash 变量 %INC	254
7.4.1 返回未定义值	220	9.5 特殊数组变量 @INC	255
7.4.2 对调用上下文的检测和响应	222	9.5.1 直接修改 @INC	256
7.4.3 闭包	224	9.5.2 使用 lib 编译指示修改 @INC	257
7.4.4 可赋值的子例程	226	9.6 定位与脚本相关的库	257
7.5 属性列表	227	9.7 检查一个模块的可用性	258
7.5.1 定义子例程属性	227	9.8 找出被安装的模块	259
7.5.2 访问属性	228	9.9 将模块的载入推迟至使用时	262
7.5.3 特殊属性	228	9.10 小结	264
7.5.4 包属性	229	第 10 章 模块与包内幕	265
7.6 小结	230	10.1 BEGIN 块、END 块和其他程序块	265
第 8 章 作用域和可见性	231	10.1.1 BEGIN 块	267
8.1 包变量	231	10.1.2 END 块	269
8.1.1 定义包变量	231	10.1.3 CHECK 和 INIT	269
8.1.2 使用“严格的”变量	233	10.2 操作包	270
8.1.3 声明全局包变量	233	10.2.1 删除包	272
8.1.4 使用 use vars 声明全局包变量	234	10.2.2 禁止包变量	273
8.1.5 使用 our 词法声明全局包变量	234	10.3 按程序方式查找包名	273
8.1.6 在 Perl 中自动局部化	235	10.4 自动加载	274
8.1.7 使用 local 局部化包变量	236	10.4.1 自动加载子例程	274
8.2 词法变量	238	10.4.2 自行定义而不是自动加载子 例程	278
8.2.1 声明词法变量	238	10.4.3 自动加载模块	280
8.2.2 在词法变量作用域外使用它们	239	10.5 导入与导出	285
8.3 符号表	240	10.5.1 导入机制	286
8.3.1 main 包	241	10.5.2 导出	288
8.3.2 typeglob 和 main 包	242	10.6 创建可安装的模块	299
8.3.3 符号表层次	243	10.6.1 编写良好的模块	300
8.3.4 直接操作符号表	243	10.6.2 创建一个工作目录	301
8.3.5 访问符号表	245	10.6.3 建立可安装的包	302
8.4 小结	246	10.6.4 添加一个测试脚本	303
第 9 章 使用模块	248	10.6.5 将模块上载到 CPAN	304
9.1 模块和包	248	10.7 小结	304
9.2 使用 do、require 和 use 装载代码	249	第 11 章 正则表达式	306
9.2.1 导入列表	250	11.1 字符串插值	306
9.2.2 禁止缺省导入	251	11.1.1 Perl 的插值语法	306
9.2.3 使用 no 禁用特性	251	11.1.2 插入元字符和字符代码	307
9.2.4 测试模块版本和 Perl 的版本	252	11.1.3 对变量进行插值	308
9.3 语用模块	253		

11.1.4 插值代码	310	12.6.3 检测文件结束	396
11.1.5 插值上下文	310	12.6.4 读取一个字符	396
11.1.6 正则表达式中的插值	311	12.7 向文件句柄中写数据	397
11.1.7 在字符串变量中插入文本	312	12.7.1 缓冲方式与自动刷新方式	398
11.1.8 保护字符串防止被插值	312	12.7.2 print 的其他替代函数	399
11.2 正则表达式	314	12.8 处理二进制文件与文本文件	399
11.2.1 正则表达式出现在什么位置	315	12.8.1 binmode 函数	400
11.2.2 正则表达式分隔符	318	12.8.2 open 编译指示	400
11.2.3 正则表达式的元素	319	12.9 随机访问	401
11.2.4 更高级的模式	321	12.9.1 利用 seek 函数定位到文件中的 某个特定位置	401
11.2.5 模式匹配修饰符	330	12.9.2 利用 seek 函数清除文件 结束条件	402
11.2.6 正则表达式与通配符	332	12.9.3 在文件结束位置写入数据	403
11.2.7 元字符	333	12.9.4 查找当前位置	404
11.2.8 提取匹配的文本	337	12.9.5 面向对象的随机访问	404
11.2.9 不止一次匹配	346	12.10 截去文件与调整文件大小	404
11.2.10 扩展模式	356	12.11 文件上锁	406
11.2.11 文档化正则表达式	357	12.11.1 建立文件锁	407
11.2.12 书写有效的正则表达式	358	12.11.2 文件上锁的问题与防止误解 的说明	409
11.2.13 检查正则表达式的有效性	367	12.12 修改缺省的输出文件句柄	409
11.2.14 正则表达式、感染变量和调试	369	12.12.1 使用特殊变量来配置其他 文件句柄	410
11.2.15 置换	375	12.12.2 自动恢复缺省文件句柄	410
11.2.16 音译	377	12.12.3 使用缺省变量与 IO::Handle 方法	411
11.3 小结	380	12.13 文件句柄的复制与别名	411
第 12 章 利用文件句柄进行输入 与输出	381	12.14 重定向文件句柄	412
12.1 IO 与文件句柄	381	12.15 Cache 多个文件句柄	414
12.2 文件句柄数据类型	382	12.16 IO::Handle 方法与特殊变量	415
12.3 标准文件句柄	382	12.17 系统级 IO	419
12.4 创建文件句柄	382	12.17.1 在系统级打开文件句柄	419
12.4.1 利用 open 函数创建文件句柄	383	12.17.2 不带缓冲的读	422
12.4.2 为读、写以及更新打开文件	384	12.17.3 不带缓冲的写	423
12.4.3 打开任意文件名	385	12.17.4 系统级文件定位	424
12.4.4 打开标准输入与标准输出	385	12.17.5 fcntl 与 ioctl	425
12.4.5 利用 IO::File 创建文件句柄	386	12.17.6 POSIX IO	429
12.4.6 数据文件句柄	388	12.18 小结	431
12.4.7 其他文件句柄	390	第 13 章 操作文件与目录	432
12.5 引用文件句柄	390		
12.6 从文件句柄读数据	392		
12.6.1 读行操作符	392		
12.6.2 读数据操作的精细控制	395		

13.1 文件与文件名	432	15.2.3 整行读	528
13.1.1 获取用户信息与组信息	432	15.2.4 利用 Term::ReadLine 进行 高级行输入	535
13.1.2 一元文件测试操作符	439	15.2.5 创建终端对象	536
13.1.3 询问文件	448	15.2.6 单词自动完成	540
13.1.4 改变文件属性	449	15.3 写屏幕	541
13.1.5 Fcntl 模块	452	15.3.1 终端功能	542
13.1.6 链接、断开链接与重命名文件	454	15.3.2 彩色输出	545
13.1.7 复制与移动文件	457	15.4 高级终端模块	549
13.1.8 比较文件	459	15.4.1 Term::Screen	549
13.1.9 查找文件	460	15.4.2 Curses 库	551
13.1.10 解释文件路径	465	15.5 通过 POSIX 直接进行终端编程	556
13.1.11 文件名 globbing	466	15.6 小结	558
13.1.12 glob 语法	467	第 16 章 警告与错误	559
13.1.13 临时文件	471	16.1 启用警告	559
13.2 操作目录	475	16.2 启用诊断	560
13.2.1 读目录	476	16.3 产生警告与错误	562
13.2.2 创建与删除目录	480	16.4 截取警告与错误	563
13.2.3 目录间的转移	483	16.5 译解系统调用所产生的错误结果	564
13.3 小结	485	16.5.1 错误编号与名称	564
第 14 章 命令行与 shell 交互	486	16.5.2 设置错误编号	565
14.1 解析命令行参数	486	16.5.3 求值代码的错误	565
14.1.1 命令行约定	486	16.5.4 扩展的错误消息	566
14.1.2 @ARGV 数组	487	16.5.5 Errno 与 POSIX 模块	567
14.1.3 利用 Getopt::Std 进行简单的 命令行处理	493	16.5.6 检查子进程与外部命令的 退出状态	567
14.1.4 利用 Getopt::Long 进行更为复杂的 命令行处理	496	16.6 将非致命错误转换成致命错误	568
14.2 shell、shell 命令与 Perl	512	16.7 利用 Carp 在上下文中返回警告信息与 错误信息	569
14.2.1 创建一个简单的 Perl shell	512	16.8 错误日志与系统日志	571
14.2.2 编写更为有用的 shell	513	16.9 高级警告	572
14.2.3 将 shell 集成到 Perl 中	516	16.10 小结	574
14.2.4 在 Windows 系统上模拟 UNIX 命令	518	第 17 章 调试	575
14.3 小结	520	17.1 语用调试支持	575
第 15 章 终端输入与输出	521	17.2 自行调试的应用程序	576
15.1 与终端进行交互	521	17.2.1 一个简单的调试系统	576
15.2 从键盘读取数据	522	17.2.2 一个更好的调试系统	577
15.2.1 简单输入	523	17.2.3 创建调试日志	578
15.2.2 利用 Term::ReadKey 控制 终端输入	523	17.2.4 添加调用上下文到调试消息中	579
		17.3 Perl 调试器	580

17.3.1 启动调试器	580	18.3.4 组合报告和规则输出	668
17.3.2 输入命令	581	18.4 小结	670
17.3.3 调试命令	584	第 19 章 面向对象 Perl	672
17.3.4 配置调试器	594	19.1 对象简介	672
17.3.5 跟踪与终端	599	19.2 使用对象编程	676
17.3.6 通过程序进入调试器	600	19.2.1 创建对象	676
17.3.7 使用调试器异常分支	601	19.2.2 使用对象	676
17.4 调试模块与信息模块	603	19.2.3 确定一个对象是什么	679
17.4.1 Dumpvalue 模块	604	19.2.4 确定通过继承得到的特征	679
17.4.2 Safe 模块	606	19.3 编写对象类	681
17.5 调试 Perl 解释器	609	19.3.1 构造器	681
17.6 自动测试	610	19.3.2 调试对象类	701
17.6.1 编写测试脚本	611	19.4 继承与派生类	707
17.6.2 自动化测试	613	19.4.1 从父类继承	707
17.7 Profile	614	19.4.2 编写可继承的类	709
17.7.1 Profile Perl 应用程序	615	19.4.3 私有方法	713
17.7.2 产生 Profile 报告	615	19.4.4 扩展与重定义对象	714
17.7.3 直接收集计时信息	619	19.4.5 多继承	717
17.7.4 性能测试	621	19.4.6 UNIVERSAL 构造器	721
17.8 小结	627	19.5 自动加载方法	726
第 18 章 文本处理与文档生成	628	19.6 保持数据私有	730
18.1 文本处理	628	19.6.1 私有类数据	730
18.1.1 使用 Text::Tabs 扩充和缩 短制表符	629	19.6.2 私有对象数据	731
18.1.2 使用 Text::Abbrev 计算缩写	630	19.7 销毁对象	734
18.1.3 使用 Text::ParseWords 解析单词 和短语	632	19.7.1 析构器和继承	735
18.1.4 使用 Text::Wrap 格式化段落	635	19.7.2 析构器与多继承	736
18.1.5 使用 Text::Soundex 匹配近似的 发声单词	638	19.8 重载操作符	737
18.1.6 其他文本处理模块	641	19.8.1 基本重载	737
18.2 文档化 Perl	641	19.8.2 确定操作数顺序和操作符名称	739
18.2.1 注释	641	19.8.3 重载比较操作符	739
18.2.2 简易老式文档	641	19.8.4 重载转换操作	740
18.2.3 pod 工具和实用程序	647	19.8.5 回退到未重载的操作	743
18.2.4 对 pod 编程	650	19.8.6 重载与继承	745
18.3 报告——在 Perl 中的 r	655	19.8.7 自动生成的操作	745
18.3.1 格式与格式数据类型	655	19.8.8 可重载的操作	747
18.3.2 格式结构	659	19.9 自动化对象类开发	747
18.3.3 页面控制	665	19.10 捆绑和被捆绑的对象	751
		19.10.1 使用被捆绑的对象	751
		19.10.2 编写被捆绑的对象	754
		19.10.3 标准捆绑对象类	754

19.10.4 被捆绑的对象方法	755	21.2.2 使用 C::DynaLib 模块	818
19.10.5 被捆绑的 hash 类的一个例子	758	21.2.3 使用 Win32::API 模块	818
19.10.6 使用 Tie::StdHash 的类 的一个例子	760	21.3 在 C 中使用 Perl	819
19.11 小结	762	21.3.1 第一步	819
第 20 章 深入 Perl	764	21.3.2 实现 Perl 解释器	821
20.1 分析 Perl 二进制文件——Config.pm	764	21.3.3 嵌入 perl 代码	823
20.1.1 perl - V	764	21.3.4 得到 Perl 值	824
20.1.2 工作原理	765	21.3.5 使用 Perl 子例程	826
20.2 Perl 进一步分析	767	21.3.6 与 Perl 内核协同工作	828
20.2.1 源文件树	767	21.3.7 使用模块	829
20.2.2 构建 Perl	769	21.4 Java-Perl Lingo	831
20.3 Perl 工作原理	770	21.5 Perl 与 COM	832
20.3.1 语法分析	770	21.5.1 PerlCOM	832
20.3.2 编译	772	21.5.2 PerlCtrl	833
20.3.3 解释	773	21.6 其他语言	834
20.4 内部变量类型	774	21.7 小结	836
20.4.1 PV	775	第 22 章 创建进程与管理进程	837
20.4.2 IV	775	22.1 信号	837
20.4.3 NV	776	22.1.1 信号处理	839
20.4.4 数组与 hash	777	22.1.2 发送信号	846
20.4.5 使用 Devel::Peek 分析原始 数据类型	777	22.1.3 报警	846
20.5 Perl 编译器	780	22.2 启动新进程	848
20.5.1 O 模块	781	22.2.1 替换当前进程	849
20.5.2 B 模块	781	22.2.2 进程 ID	850
20.5.3 B::模块家族	783	22.2.3 进程、组及端口监控程序	850
20.5.4 编写 Perl 编译器后端	796	22.3 处理子进程并返回退出代码	852
20.6 小结	799	22.3.1 等待一个子进程	852
第 21 章 Perl 与其他程序设计语言 的集成	800	22.3.2 获得退出状态	852
21.1 在 Perl 中使用 C	800	22.3.3 处理多个子进程	853
21.1.1 XS 概述	801	22.3.4 POSIX 标志及函数	855
21.1.2 从 h2xs 启动	802	22.4 进程间通信	856
21.1.3 XS 文件	804	22.4.1 十分简单的解决方法	856
21.1.4 XS 函数	806	22.4.2 管道	857
21.1.5 TYPemap	811	22.4.3 打开并运行外部进程	859
21.1.6 Makefile	814	22.4.4 双向管道	860
21.2 动态链接	815	22.4.5 使用分叉的 open 避开 shell	862
21.2.1 使用 FFI 模块	816	22.4.6 连接外部进程的双向管道	863
		22.5 进程间共享数据	865
		22.5.1 IPC::SysV	866
		22.5.2 消息队列	867

22.5.3 信号量	870	24.13 安全性	953
22.5.4 共享内存段	873	24.14 风格	953
22.6 线程	875	24.15 平台	956
22.6.1 检查线程支持	876	24.15.1 UNIX	957
22.6.2 创建线程	876	24.15.2 DOS 及其派生系统	957
22.6.3 标识线程	877	24.15.3 Mac OS	958
22.6.4 线程特定数据	877	24.15.4 其他 Perl	959
22.6.5 线程管理	878	24.16 函数实现	960
22.6.6 变量锁	880	24.17 小结	964
22.6.7 条件变量、信号量及队列	880	第 25 章 Unicode	965
22.6.8 线程安全与锁定代码	890	25.1 Unicode 所影响的用户群	965
22.7 小结	890	25.2 结论	965
第 23 章 Perl 网络编程	891	25.3 字符和 Unicode 简介	965
23.1 网络简介	891	25.4 Perl 中的数据	968
23.1.1 协议层	891	25.5 Unicode 和正则表达式	969
23.1.2 网际协议	893	25.6 双向脚本	972
23.1.3 UDP&TCP	896	25.6.1 显示 bidi	973
23.1.4 ICMP	898	25.6.2 bidi 算法做了什么	974
23.1.5 其他协议	898	25.6.3 Perl 和 bidi	975
23.1.6 表示层/应用层	898	25.7 Perl、I18n 和 Unicode	976
23.1.7 匿名、广播与回送地址	898	25.7.1 安装 Unicode 字体	977
23.2 Perl 网络编程	899	25.7.2 安装 Unicode 编辑器	977
23.3 多路复用服务器	921	25.7.3 创建 HTML 模板	977
23.4 获得网络信息	928	25.7.4 处理资源文件	978
23.4.1 系统网络文件	928	25.7.5 运行脚本	979
23.4.2 判断本地主机名	939	25.7.6 输出文件	979
23.5 小结	940	25.7.7 正在进行的工作	981
第 24 章 编写可移植 Perl	942	25.8 小结	981
24.1 可移植的必要性	942	第 26 章 本地化和国际化	982
24.2 新行	942	26.1 走向地区的原因	983
24.3 文件与文件系统	945	26.2 关于时间：时区	995
24.4 字节存放次序与数字宽度	949	26.3 看上去像外语	997
24.5 系统交互	949	26.4 葡萄牙语的词性变化	999
24.6 进程间通信	950	26.5 Lingua::* 模块	1001
24.7 外部子例程	951	26.6 编写多语种网页	1005
24.8 模块	951	26.7 创建自己的本地化 Perl 模块	1009
24.9 时间及日期	952	26.8 小结	1011
24.10 字符集与字符编码	952	附录 A 命令行选项	1012
24.11 国际化	953	附录 B 特殊变量	1014
24.12 系统资源	953	附录 C 函数	1020

附录 D 正则表达式	1036	附录 G Perl 资源	1050
附录 E 标准语用模块	1041	附录 H 与本书相关的网站:	
附录 F 标准功能模块	1043	p2p.wrox.com	1052