

100%

内容丰富、权威

精通C/C++、Perl、CGI以及Shell编程的技巧

利用Emacs、gcc、make、gdb以及其他Linux编程工具

学习已在所有主要Linux发行版中证实的编程方案

美国IDG“宝典”丛书

Linux Programming Bible

丛书
累计印数
24万册



[美] John Goerzen 著 魏永明 耿岳 游华云 等译
魏永明 审校

Linux 编程 宝典

适合
实际应用
环境的可运
行示范代码



电子工业出版社

Publishing House Of Electronics Industry
URL: <http://www.phei.com.cn>

美国 IDG“宝典”丛书

Linux 编程宝典

Linux Programming Bible

[美] John Goerzen 著

魏永明 耿岳 游华云 等译

魏永明 审校

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书面向 Linux 系统的广大程序员,目的是提供一本详尽的编程教材和指南。本书的作者是 Debian GNU/Linux 发行版的主要开发者。

本书主要内容包括基本编程工具的使用、C 语言环境、Linux 文件系统及进程模型、Linux 中的进程间通信、Perl 及其在 CGI 和数据库应用中的使用、利用 Perl/Tk 和 GTK 的图形用户界面等,还包含有对 CVS(一个广泛使用的版本控制系统)、安全性以及优化等方面的描述,这些内容是目前 Linux 系统上进行开发的热点主题。

通过阅读本书,读者不仅能够学习到 Linux 如何实现一些功能,同时还能掌握如何在自己的应用程序中充分发挥这些功能。

本书非常适合于初级及中级 Linux 程序员阅读,对学习 Linux 编程以及操作系统的在校大学生来说,也不失为一本好书。

Linux Programming Bible by John Goerzen



Copyright ©2000 by Publishing House of Electronics Industry. Original English language edition copyright ©2000 by IDG Books Worldwide, Inc. All rights reserved including the right of reproduction in whole or in part in any form. This edition published by arrangement with the original publisher, IDG Books Worldwide, Inc., Foster City, California, USA.

本书中文简体专有翻译出版版权由美国 IDG Books Worldwide, Inc. 公司授予电子工业出版社及其所属今日电子杂志社。未经许可,不得以任何手段和形式复制或抄袭本书内容。该专有出版版权受法律保护,侵权必究。

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,翻版必究。

图书在版编目(CIP)数据

Linux 编程宝典/(美)格泽(Goerzen, J.)著;魏永民译

-北京:电子工业出版社,2000.10

(美国 IDG“宝典”)

ISBN 7-5053-6210-0

I. L... II. ①格... ②魏... III. 操作系统(软件), Linux IV. TP316.81

中国版本图书馆 CIP 数据核字(2000) 第 71189 号

从 书 名:美国 IDG“宝典”丛书

书 名:Linux 编程宝典

著 者:[美] John Goerzen

译 者:魏永明 耿 岳 游华云 等

审 校 者:魏永明

责任编辑:李秦华

特约编辑:李文义

印 刷 者:北京东光印刷厂

出版发行:电子工业出版社 URL: <http://www.phei.com.cn>

北京市海淀区万寿路 173 信箱 邮编 100036

经 销:各地新华书店经销

开 本:787×1092 1/16 印张:47.75 字数:1146 千字

版 次:2000 年 10 月第 1 版 2000 年 10 月第 1 次印刷

书 号:ISBN 7-5053-6210-0

TP·3345

定 价:79.00 元

著作权合同登记号:图字:01-2000-0645

凡购买电子工业出版社的图书,如有缺页、倒页、脱页者,请向购买书店调换。若书店售缺,请与本社发行部联系调换。联系电话:010-68159356、68279077

出版说明

美国 IDG“宝典”丛书是美国 IDG Books Worldwide, Inc. 编辑出版的全球最畅销的计算机系列图书之一。本套丛书编写旨在满足读者想要快速获取最全面、最准确的计算机信息的日益需求。

宝典丛书侧重于完成具体任务,而不是学习随机功能。这套丛书不是繁琐冗长的使用手册或枯燥乏味的大本参考书。宝典丛书的作者准确地告诉读者用计算机软硬件可以做哪些事以及如何做这些事。读者可跟着学,循序渐进。本丛书的内容全面,并且在语言和结构方面易于理解。

宝典丛书的作者都是出类拔萃的专家,他们为你提供内行意见和独到的提示和技巧。这些作者通过读者来信、培训讲座、电子邮件、用户小组以及咨询工作与用户保持联系。由于这些作者了解计算机日常实际使用情况并与读者保持直接联系,所以使这套宝典丛书具有战略优势。

本宝典丛书的作者经验丰富,他们以最有效的方式研究探讨某一题目。我们的研究表明,读者踊跃购买计算机图书的原因之一是,想从计算机行家里手那里取经并有所收获。读者定会从这套宝典丛书中受益匪浅。

此外,作者还在该宝典丛书中列出或推荐了一些有用的软件。本系列丛书大都附带有 CD-ROM 光盘,光盘中包涵有与该书的内容或主题有关的成功案例,免费、共享软件以及最新的软件测试版本。

无论是通读本书或逐段阅读,还是一次只读一个主题,读者都会从中获益不小,并能找到读者想要的信息与答案。

在宝典丛书的组织翻译中,译者大都是国内知名的专家学者,他们多年从事计算机开发与应用、测试与培训,其渊博的知识、丰富的经验,充分体现在本丛书的各个章节中。翻译过程中,在忠实原著的基础上,充分体现国内读者的需求,而且在技术名词、术语和技术内容本身上力求通用、严谨、准确,一般采用“约定俗成”的通用译法。

宝典丛书以计算机中级用户为主要对象,内容丰富、权威,解释详细,示例和练习由浅入深,对高级话题和疑难问题的讨论深入细致,以适合于中高级计算机用户阅读。

译者序

毋庸置疑, Linux 在国内已经进入稳定发展阶段。以 BluePoint Linux、Turbo Linux 为代表的中文化 Linux 发行版,已经解决了大部分 Linux 的本地化工作。但是,需要指出的是, Linux 上应用的缺乏,尤其是中文应用软件的缺乏,是 Linux 尚不能为大众所接受的主要障碍。国内 Linux 先行者已经发现这一问题,并积极为推动 Linux 上的应用开发而努力。清华 SMTH BBS (<http://smth.org>) Linux 板上与编程相关的问题越来越多;清华大学的一个学生组织,即 AKA (<http://www.aka.citf.net>)正在主办以 Linux/UNIX 编程为主题的免费系列讲座,并得到了国内著名 Linux 开发商的支持,第一次讲座就吸引了多达 300 人的爱好者参加。这一切,奏响了国内大规模 Linux 应用开发的序曲。

但是,对许多通过 Windows 平台学习编程技术的开发人员来讲,面对 Linux 编程的浩瀚海洋,的确有一些茫然不知所措。许多人不熟悉 Linux 上的开发工具,不了解 Linux 上众多的编程语言,不知道 Linux 上的编程风格,不清楚如何使用 Linux 操作系统的各种系统设施。而且,与 Linux 系统的快速发展相比,程序员可用的文档资料显得有些不足。为了给广大程序员提供一个编程教材和指导,作为 Debian GNU/Linux 发行版的主要开发者,John Goerzen 为广大程序员编写了《Linux 编程宝典》一书。通过阅读本书,读者不仅能够学习到 Linux 如何实现一些功能,同时还能掌握如何在自己的应用程序中充分发挥这些功能。作为 Linux 上的开发人员,我们(译者)也很高兴能向广大 Linux 程序员介绍这本书。

本书主要有如下特色:

- ◆ 涉及面广。本书包括基本编程工具的使用、C 语言环境、Linux 文件系统及进程模型、Linux 中的进程间通信、Perl 及其在 CGI 和数据库应用中的使用、利用 Perl/Tk 和 GTK 的图形用户界面等,还包含有对 CVS(一个广泛使用的版本控制系统)、安全性以及优化等方面的描述。这些内容,是目前 Linux 系统上进行开发的热点主题。
- ◆ 详细阐述了某些缺少技术文档的高级技术的使用方法。比如, Linux 具备如今最好的共享库系统,但是由于这一系统很新,从而很少有描述它的文档。在本书出现之前,程序员为了有效使用共享库,不得不纠缠一些细节问题,从而踟躇不前。而本书则详细描述了共享库的工作原理以及使用方法。
- ◆ 包含大量的示例代码。作为程序员,在编写程序时,不仅要知道为何使用某个特定的功能,而且要知道如何使用这些功能。通过书中的大量的示例程序及其注释,将看到如何将思想和概念付诸实施,读者也可以将这些代码作为自己程序的起点。
- ◆ 内容组织得当,既可用作教材,也可用作参考手册。因为本书深入描述了 Linux 的许多不同方面,读者几乎可以从书中找到任何需要的信息。
- ◆ 语言流畅、讲解清楚。

因此,本书非常适合于初级及中级 Linux 程序员阅读,对学习 Linux 编程以及操作系统的在校大学生来说,也不失为一本好书。

本书由魏永明、耿岳、游华云等组织翻译,全书由魏永明审校。参与本书翻译和文字校对工作的还有郑翔、陶品、张文翔、雷年胜、陆伟、蒋洪军、方伟等。另外,张立新、李涛、张华、孙登峰、李超、高寿福、徐成、甄武等也参与了部分翻译工作。由于译者水平有限,错误之处在所难免,还望读者不吝指正。

译者
2000年8月

目 录

前言	(1)
第 1 部分 Shell 和基本工具	(7)
第 1 章 Linux 编程环境简介	(9)
1.1 Linux 编程基本概念	(9)
1.2 Linux 设计	(10)
1.3 Linux 文档	(10)
1.4 手册页	(11)
1.5 Info 页	(12)
1.6 Perl 文档	(14)
1.7 程序文档	(15)
1.8 Internet 资源	(15)
1.9 小结	(16)
第 2 章 Shell 编程简介	(17)
2.1 Bash 简介	(17)
2.1.1 创建脚本	(18)
2.1.2 启动 Bash	(19)
2.1.3 返回值	(20)
2.1.4 简单命令组合	(21)
2.1.5 通配符	(26)
2.1.6 引号和转义	(26)
2.1.7 注释	(29)
2.2 重定向和管道	(30)
2.2.1 输出重定向	(31)
2.2.2 输入重定向	(35)
2.2.3 管道	(37)
2.2.4 命令置换	(40)
2.3 变量	(40)
2.3.1 环境变量	(42)
2.3.2 特殊变量	(44)
2.4 函数	(46)
2.5 条件和循环	(50)
2.5.1 if	(50)

2.5.2 使用 [...] 测试	(51)
2.5.3 case	(53)
2.5.4 while	(55)
2.5.5 for	(56)
2.6 Shell 应用	(58)
2.7 小结	(59)
第 3 章 使用正则表达式	(61)
3.1 正则表达式简介	(61)
3.1.1 模式	(62)
3.1.2 正则表达式语法	(63)
3.1.3 egrep 简介	(67)
3.1.4 sed 简介	(69)
3.1.5 Perl 中的正则表达式	(70)
3.2 理解字符类	(73)
3.3 使用量词	(75)
3.4 备选与分组	(80)
3.5 在 Linux 下对正则表达式的支持	(81)
3.5.1 Perl	(81)
3.5.2 sed 和 awk	(85)
3.5.3 C/C++	(85)
3.6 小结	(86)
第 4 章 Emacs 简介	(87)
4.1 Emacs 入门	(87)
4.1.1 Emacs 键盘操作符号	(87)
4.1.2 浏览	(89)
4.1.3 搜索	(90)
4.1.4 Emacs 区域	(91)
4.1.5 缓冲区	(91)
4.1.6 窗口	(91)
4.1.7 框架	(93)
4.1.8 语法和括弧高亮显示	(93)
4.2 主模式	(94)
4.2.1 C 模式	(95)
4.2.2 Perl 模式	(97)
4.2.3 其他模式	(98)
4.3 Emacs 是一个 IDE	(98)
4.3.1 编译程序	(99)
4.3.2 调试	(99)

4.3.3 使用标记	(100)
4.3.4 在 Emacs 中的 Shell	(101)
4.3.5 Term 模式	(103)
4.3.6 Dired	(104)
4.3.7 w3	(105)
4.3.8 Gnus	(107)
4.3.9 版本控制	(107)
4.4 获取帮助	(108)
4.5 小结	(109)
第 5 章 理解 Linux 数据文件和脚本	(111)
5.1 基本概念	(111)
5.2 文件系统层次	(112)
5.2.1 根目录	(112)
5.2.2 /dev:设备文件	(114)
5.2.3 /etc:配置和启动文件	(115)
5.2.4 /usr:标准系统程序	(115)
5.2.5 /var:变量数据	(115)
5.3 passwd 和 shadow 文件	(116)
5.3.1 从 Shell 中获取账号信息	(117)
5.3.2 从 C 中获取账号信息	(118)
5.3.3 从 Perl 中获取账号信息	(120)
5.4 组文件	(121)
5.5 init 文件	(122)
5.6 网络文件	(126)
5.6.1 DNS 文件	(126)
5.6.2 安全性文件	(127)
5.6.3 超级服务器文件	(128)
5.7 小结	(129)
第 2 部分 C 环境	(131)
第 6 章 使用 gcc	(133)
6.1 使用 gcc 编译程序	(133)
6.1.1 警告	(134)
6.1.2 优化 gcc	(135)
6.1.3 产生调试标记	(139)
6.2 gcc 编译流程简介	(143)
6.2.1 C 编译器: gcc	(143)
6.2.2 C++ 编译器: g++	(143)
6.2.3 C 预处理器: cpp	(144)

6.2.4 程序员“礼品店”	(147)
6.3 管理大型项目	(148)
6.4 使用高级 gcc 选项	(150)
6.4.1 指定查找路径	(151)
6.4.2 连接库	(151)
6.4.3 使用管道加速编译过程	(152)
6.4.4 使用 gcc 的 -v 选项	(152)
6.4.5 使用 ANSI C	(155)
6.5 小结	(155)
第 7 章 使用 GNU Make 管理项目	(157)
7.1 GNU make 简介	(157)
7.1.1 Makefile 规则	(158)
7.1.2 一个简单的 Makefile 文件	(158)
7.1.3 更巧妙的 Makefile 文件	(161)
7.2 使用智能的 Makefile	(166)
7.2.1 两种变量类型	(166)
7.2.2 通配符:问题和解决办法	(167)
7.2.3 依赖性计算	(169)
7.3 创建其他文件	(181)
7.4 使用递归 make	(187)
7.5 小结	(194)
第 8 章 内存管理	(195)
8.1 静态与动态内存	(195)
8.1.1 静态分配内存	(195)
8.1.2 动态分配内存	(197)
8.2 安全性和设计着眼点	(201)
8.3 高级指针	(203)
8.3.1 解析数据	(214)
8.3.2 代码分析	(223)
8.4 发现问题	(225)
8.5 小结	(226)
第 9 章 库和连接	(227)
9.1 库简介	(227)
9.2 创建和使用静态库	(228)
9.3 创建和使用共享库	(232)
9.4 使用高级共享库特性	(234)
9.4.1 ldd 工具	(234)

9.4.2 soname	(234)
9.4.3 共享库装载器	(236)
9.4.4 使用 LD_PRELOAD	(236)
9.4.5 使用 dlopen	(239)
9.5 小结	(239)
第 10 章 使用 gdb 调试	(241)
10.1 gdb 需求	(241)
10.2 调试步骤	(242)
10.2.1 调试指南	(242)
10.2.2 调试其他进程	(249)
10.3 显示数据	(253)
10.3.1 使用 print 和 display 命令	(253)
10.3.2 内存检查	(259)
10.3.3 使用 printf 命令	(260)
10.3.4 使用 set 命令	(261)
10.4 使用断点和观察窗口	(261)
10.4.1 设置断点	(261)
10.4.2 设置观察窗口	(264)
10.5 core dump 分析	(268)
10.6 命令小结	(275)
10.7 小结	(278)
第 3 部分 Linux 模型	(279)
第 11 章 文件、目录和设备	(281)
11.1 文件的实质	(281)
11.1.1 系统调用 stat() 和 lstat()	(282)
11.1.2 opendir()、readdir()及其相关函数	(292)
11.2 I/O 方法	(299)
11.2.1 基于流的 I/O 方法	(299)
11.2.2 系统调用 I/O 方法	(300)
11.3 特殊文件	(303)
11.4 小结	(304)
第 12 章 Linux 系统中的进程	(305)
12.1 理解进程模型	(305)
12.2 进程基础知识	(305)
12.3 启动和终止进程	(307)
12.3.1 派生的基本原理	(307)
12.3.2 执行其他程序	(308)

12.3.3	等待进程	(310)
12.3.4	一起使用	(318)
12.4	使用返回代码	(320)
12.5	进程的同步措施	(323)
12.5.1	原子操作和非原子操作	(324)
12.5.2	死锁	(325)
12.5.3	竞争条件	(325)
12.5.4	自旋和忙等待	(326)
12.6	了解安全性	(327)
12.6.1	安全性的基本知识	(327)
12.6.2	安全性秘密	(327)
12.6.3	setuid 和 setgid	(329)
12.6.4	setuid 和 setgid 的负效应	(339)
12.6.5	保证 setuid 和 setgid 的安全	(340)
12.7	小结	(343)
第 13 章	了解信号	(345)
13.1	信号的使用	(345)
13.2	信号处理程序	(346)
13.2.1	基本信号处理程序	(346)
13.2.2	阻塞信号	(351)
13.2.3	高级信号处理程序	(357)
13.3	信号发送	(363)
13.4	信号和系统调用	(372)
13.5	信号处理程序带来的危险	(373)
13.6	小结	(373)
第 14 章	Linux 的 I/O 系统简介	(375)
14.1	程序库和系统调用	(375)
14.1.1	基于流的 I/O 方法	(376)
14.1.1	系统调用 I/O 方法	(379)
14.2	错误情况	(386)
14.2.1	使用封装的函数库	(387)
14.2.2	在读者自己的程序中使用封装函数库	(395)
14.3	高级 I/O	(399)
14.3.1	稀疏文件	(399)
14.3.2	非阻塞 I/O 方式	(401)
14.4	内存映射的 I/O	(402)
14.5	函数调用 select() 和 poll()	(406)
14.5.1	select() 函数	(406)

14.5.2 poll() 函数	(411)
14.6 咨询锁定	(414)
14.7 小结	(419)
第 15 章 终端编程	(421)
15.1 终端属性	(422)
15.2 伪终端	(425)
15.3 Ncurses	(429)
15.4 小结	(456)
第 4 部分 与世界交流	(457)
第 16 章 共享内存和信号量	(459)
16.1 使用共享内存	(459)
16.2 使用信号量同步	(460)
16.3 利用共享内存通信	(470)
16.4 小结	(487)
第 17 章 使用管道和 FIFO	(489)
17.1 建立管道	(490)
17.2 实现重定向	(499)
17.3 解决通信问题	(504)
17.4 使用 FIFO	(504)
17.5 小结	(509)
第 18 章 Internet 套接字	(511)
18.1 TCP/IP 简介	(511)
18.1.1 问题	(511)
18.1.2 解决方案	(512)
18.1.3 术语注解	(513)
18.2 TCP/IP 的独特挑战	(513)
18.3 协议	(514)
18.4 寻址	(515)
18.5 客户端连接	(516)
18.6 服务器端连接	(519)
18.7 一个网络函数库	(529)
18.8 小结	(542)
第 19 章 高级 TCP/IP 套接字	(543)
19.1 服务器设计和多工化	(543)
19.2 用户数据报协议	(552)

19.3 小结	(552)
第 5 部分 利用 Perl 协同工作	(553)
第 20 章 Perl 简介	(555)
20.1 Perl 的设计思想	(555)
20.1.1 变量	(556)
20.1.2 数组	(557)
20.1.3 哈希表	(558)
20.2 第一步	(559)
20.3 数据结构	(565)
20.3.1 引用	(565)
20.3.2 匿名引用	(566)
20.3.3 符号引用	(567)
20.3.4 数组	(567)
20.3.5 嵌套哈希表	(572)
20.4 子例程和作用域	(577)
20.5 流控制	(579)
20.6 OOP 特征	(580)
20.7 小结	(588)
第 21 章 利用 Perl 操作数据	(589)
21.1 读取数据	(589)
21.2 分析和处理数据	(591)
21.2.1 使用 split	(593)
21.2.2 使用 grep	(595)
21.3 保存数据	(596)
21.3.1 使用数据库	(596)
21.3.2 使用 Data Dumper	(607)
21.3.3 Data Dumper 选项	(611)
21.4 输出和特别考虑的问题	(612)
21.4.1 基本输出	(612)
21.4.2 输出到文件或命令	(613)
21.4.3 open 的选项	(613)
21.4.4 传递文件句柄	(614)
21.4.5 文件句柄的作用域	(614)
21.5 小结	(615)
第 22 章 CGI 编程	(617)
22.1 CGI 与 Web	(617)
22.2 编写 CGI 脚本	(618)

22.3 处理无连接问题	(630)
22.4 解决性能问题	(632)
22.5 小结	(632)
第 23 章 利用 DBI 访问 SQL 数据库	(633)
23.1 数据库简介	(633)
23.2 使用 DBI 的第一步	(634)
23.3 使用 SQL	(638)
23.3.1 创建表	(638)
23.3.2 插入数据	(639)
23.3.3 读取数据	(640)
23.3.4 更新表	(643)
23.3.5 删除信息	(644)
23.3.6 连接表	(645)
23.4 在应用程序中使用数据库	(648)
23.4.1 DBI 和命令行	(648)
23.4.2 DBI 和 CGI	(652)
23.5 小结	(666)
第 6 部分 使用 X 的图形界面	(667)
第 24 章 利用 Perl/Tk 的 GUI	(669)
24.1 Linux 中的 GUI 编程	(669)
24.1.1 X Window 系统基础	(670)
24.1.2 位置无关性	(670)
24.2 客户解剖	(670)
24.2.1 Xlib	(671)
24.2.2 构件集	(671)
24.2.3 层次化窗口	(671)
24.2.4 配置	(671)
24.3 基于事件的编程	(672)
24.4 使用 Perl/Tk 的第一步	(672)
24.5 对象属性	(674)
24.6 特殊对象	(679)
24.6.1 窗框	(679)
24.6.2 菜单	(679)
24.6.3 文本和画布	(684)
24.7 几何管理器	(686)
24.8 特别考虑的问题	(691)
24.9 SpecTcl/SpecPerl	(691)
24.10 小结	(694)

第 25 章 利用 GNOME 建立 GUI	(695)
25.1 使用 GNOME 组件	(695)
25.2 第一步	(696)
25.3 绘制窗口	(699)
25.4 其他 GNOME 注解	(703)
25.5 小结	(704)
第 7 部分 将一切组织起来	(705)
第 26 章 利用 CVS 管理归档和协作	(707)
26.1 CVS 简介	(707)
26.2 建立仓库	(708)
26.3 CVS 的日常使用	(709)
26.4 使用标记和分支	(713)
26.4.1 标记	(714)
26.4.2 分支	(715)
26.5 访问网络	(717)
26.5.1 设置服务器	(717)
26.5.2 生成 ssh 键	(717)
26.5.3 环境变量	(718)
26.6 技巧和诀窍	(719)
26.6.1 关键词	(719)
26.6.2 二进制文件	(720)
26.6.3 使用子目录	(721)
26.6.4 CVSROOT 文件	(721)
26.7 小结	(722)
第 27 章 理解安全性和代码	(723)
27.1 好代码的重要性	(723)
27.2 Linux 安全性概览	(724)
27.2.1 验证	(724)
27.2.2 访问控制	(724)
27.3 安全性的指导方针	(726)
27.3.1 安全性原则	(726)
27.3.2 常见问题	(728)
27.3.3 编写安全代码	(729)
27.4 小结	(732)
第 28 章 优化性能	(733)
28.1 更快代码的原则	(733)

28.1.1 三种度量	(733)
28.1.2 循环	(733)
28.1.3 帮助优化器	(736)
28.1.4 避免浮点数	(736)
28.1.5 重新编写时间关键代码块	(736)
28.1.6 增加块大小	(736)
28.2 高价操作和廉价操作	(737)
28.2.1 系统调用	(737)
28.2.2 fork	(737)
28.2.3 exec	(737)
28.2.4 system	(737)
28.3 编译器优化	(738)
28.4 使用 gprof	(738)
28.5 小结	(748)

前 言

人们已经认识到, Linux 的确有与众不同之处。Linux 这一操作系统从 Linus Torvalds 上大学时的家庭作业开始, 已经发展成为市场上一支强有力的生力军, 说白了, 正在不断地创造商业价值。

但 Linux 的特殊之处在哪里呢? 为什么我们作为程序员应该关注它呢? 读者可以在本书中找到这些问题的答案。Linux 不仅仅是一个新的操作系统, 它同时为全世界的程序员呈现出了最好的东西, 而这些东西是程序员最想看到的。它丰富的多任务能力以及强大的通信功能, 可以帮助读者快速编写强大和快速的应用程序。Linux 支持多种编程语言, 其中包括 C、C++、Perl、Java、LISP、Prolog、Scheme、Pascal、BASIC、两种 Shell 语言、汇编、Ada、Smalltalk 以及 FORTRAN。Linux 的编程环境是第一流的; 在 Linux 存在之前, 许多工具就已经精雕细琢, 这归功于 UNIX。

在 Linux 还算是一个新系统时, 笔者就发现缺少一些 Linux 程序员必备的信息, 这也是编写本书的来由。通过阅读本书, 用户不仅能够学习到 Linux 如何实现一些功能, 同时还能掌握如何在自己的应用程序中充分发挥这些功能。书中大量的示例代码实际是一些完整的程序, 编译之后就可以运行, 其中某些代码还可从 Internet 下载。

为什么需要这本书

Linux 强大功能的一部分便是其多功能性。例如, 我们可以选择五种不同的方式在程序之间进行通信。或者, 我们也可以选择各种不同的编程语言来编写自己的代码。对这种灵活性有一些争议。一方面我们需要有多种选择, 这样在解决问题的时候, 我们可以选择最简单易行的方法; 而另一方面, 我们需要一些信息, 例如, 哪种通信方式适合我自己? 已有的文档很少就这些问题阐述相关问题。另外, 在准备编写自己的程序时, 不仅要知道为何使用特定功能, 而且要知道如何使用这些功能。通过书中的示例程序以及注释, 将看到如何将思想和概念付诸实施, 用户也可以将这些代码作为自己程序的起点。

Linux 也可以帮助我们进入计算机技术的新领域。它具备现今最好的共享库系统之一, 但是, 同样由于这一系统很新, 从而很少有描述它的文档。在本书出现之前, 程序员为了有效使用共享库, 不得不纠缠一些细节问题, 从而踟躇不前。而《Linux 编程宝典》则详细描述了共享库的工作原理以及使用方法。

除将本书作为教材之外, 《Linux 编程宝典》还可以作为参考手册。因为本书深入描述了 Linux 的许多不同方面, 用户几乎可以从书中找到任何需要的信息。