

流行软件指南丛书

UNIX

使用指南

杜维文 编著

曾 斌 审



清华大学出版社



流行软件指南丛书

unix

UNIX 使用指南

杜雍文 编著
曾 斌 审

清华大学出版社

(京)新登字 158号

内 容 简 介

几十年来 UNIX 一直是世界上的主流操作系统,它在科研、教育、商业、工业等众多领域内发挥着无可替代的巨大作用。本书从博大精深的 UNIX 中选择了其中重要和常用的内容,即 UNIX 系统基础、UNIX 系统使用、UNIX 系统管理、UNIX 系统开发进行了系统而深入的阐述。通过对本书的学习,读者不仅能够使用 UNIX 系统,而且还可以管理 UNIX 系统以及在 UNIX 系统上进行开发研究,进一步完善 UNIX 系统。本书可供大学计算机系的师生及广大的计算机使用者学习使用。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

UNIX 使用指南/杜维文编著. —北京:清华大学出版社,1997. 6

(流行软件指南丛书)

ISBN 7-302-02554-1

I. U… I. 杜… II. UNIX 操作系统-指南 IV. TP316-62

中国版本图书馆 CIP 数据核字(97)第 10640 号

出版者:清华大学出版社(北京清华大学校内,邮编 100084)

<http://www.tup.tsinghua.edu.cn>

印刷者:中国科学院印刷厂

发行者:新华书店总店北京发行所

开 本: 787×1092 1/16 印张: 26.75 字数: 633 千字

版 次: 1997 年 8 月第 1 版 1999 年 9 月第 5 次印刷

书 号: ISBN 7-302-02554-1/TP·1300

印 数: 21001~25000

定 价: 35.00 元

《流行软件指南丛书》编辑委员会

主 编 沈金发

副主编 熊桂喜

委 员 李幼哲 郑甫京 李振格 王 敏 赵仲明
王图印 郑全战 谷彦国 薛继峰

《流行软件指南丛书》序

近年来,计算机以前所未有的速度发展和普及,一些优秀的计算机软件广泛流行。人们已离不开这些流行软件。学习这些软件,掌握与使用这些软件已是广大计算机用户的迫切愿望。

围绕着流行软件,国内许多出版社已翻译出版了不少国外图书,这无疑对广大用户用好这些软件起到了积极作用。但是,由于许多软件需要或已经汉化,国内也不断自行开发出一些优秀软件,加之文化背景的不同,仅靠翻译书远不能满足广大用户的需要。国内众多出版社虽然也组织编写了不少有关计算机软件的图书,但是无论从选题规模、组织方式还是人力投入上都不能和国外知名的软件丛书相比。针对这种状况,我们在清华大学出版社的大力支持下,组织国内一些长期从事计算机教学、科研与应用的专家和技术人员,精选那些为我国广大计算机用户普遍认同的优秀流行软件,精心策划了这套《流行软件指南丛书》。

《流行软件指南丛书》面向初级和中级用户,每一本介绍一种流行软件,内容基本覆盖该软件的全部功能;全书以功能为主线,组织方式与人们学习和使用软件的方式相一致;采用通俗易懂、循序渐进的方式讲解与阐述软件的各种使用方法和所涉及背景知识;根据每种软件的特点,配有针对性的图示和实例。总之,力求使本丛书中的每一本书既是学习软件的好教材,又是可以方便查阅的好参考书。

多年来,清华大学出版社在广大读者的关心和支持下,出版了不少深受欢迎的计算机图书。现在,本丛书编委会将以极大的热情、严谨的态度来组织编写这套面向广大计算机用户的普及读物,希望通过我们的努力,使这套丛书能真正成为大家工作与学习的好帮手。

最后,衷心地希望广大读者对本丛书的不足和缺点提出批评,对本丛书今后的发展提出宝贵意见。

丛书编委会

1995年6月

前 言

UNIX 从其诞生到现在,已经有将近 30 年的历史了。30 年的风风雨雨,一代又一代的计算机工作者在其上花费了无数的心血与劳动。今天 UNIX 也当之无愧地成为世界上的主流操作系统,它在科研、教育、商业、工业等领域内发挥着无可替代的巨大作用!

UNIX 系统的广泛应用迫切要求广大用户能够熟练地使用它、管理它,并能够充分发挥其作为一个良好的软件开发环境所具有的各种功能。同时用户对于 UNIX 系统的内部实现机制应当有一定程度的了解,以“知其所以然”,这样才能在使用、管理、开发过程中做到心中有数,以不变应万变。

考虑到这些问题,我们结合自己长期在 UNIX 环境下进行开发的经验,精心地从 UNIX 博大精深的庞杂内容中选择如下部分加以详细介绍:

- **UNIX 系统基础** 讨论 UNIX 的文件系统及进程系统的内部实现机制和使用;同时对 UNIX 的历史、现在和将来作了简要介绍和讨论。
- **UNIX 系统使用** 这一部分我们从用户使用的角度出发,详细讨论了最常用的各种 UNIX 工具,如 Shell、文本编辑程序、存储介质,以及 UNIX 环境下的网络(主要是 Internet)实用程序的使用。
- **UNIX 系统管理** 系统管理是最为庞杂的内容。在这一部分我们将讨论系统的启动与关闭、安全性管理、文件系统管理、时间与作业调度等,这些最为常见的管理内容,以期读者能够对系统管理获得一般性的了解。
- **UNIX 系统开发** UNIX 系统是一个极好的软件开发环境。在这一部分我们对使用得最为广泛的若干开发工具如 C 编译系统、调试系统、程序维护系统(make)和源程序版本维护工具进行了详细讨论。

在有针对性地选取所介绍内容的基础上,我们还尽量使本书内容更加实用。在每一章节的后面,我们都给出了对所涉及到的命令的详细描述,以期读者对所讨论的问题作更深入的了解;同时,这些内容也可以作为参考手册而进行查阅。实际上,阅读有关命令的详细解释是深入了解 UNIX 系统的一种极好方法。

注重理论性是本书的另一大特色。UNIX 系统之所以显得比较高深莫测,很大一部分原因是由于用户对其内部实现不甚了解。书中我们尽可能地揭开覆盖在 UNIX 脸上的神秘面纱,以还其本来面目。我们相信,读者在了解了这些内容之后,既会领略到 UNIX 的精巧,同时又会有有一种顿悟的感觉:原来是这样!内容实现机制与使用在本书中并不是脱节的,而是紧紧相连。

UNIX 是一部博大精深的大书。本书想要穷尽其所有内容是很困难的,而是选择其最基本、最常用的内容进行了深入的阐述。因学识所限,加之时间紧促,书中缺点和不当之处在所难免,敬请读者多多指教。

编著者

1996 年 9 月

目 录

第 1 部分 UNIX 系统基础

第 1 章 UNIX 简介	1
1.1 UNIX 的发展历史	1
1.2 UNIX 现状	4
1.3 UNIX 的未来	6
第 2 章 文件系统	7
2.1 概述	7
2.2 用户观点中的文件系统	9
2.2.1 UNIX 的树型目录结构	9
2.2.2 目录和路径	11
2.2.3 命令附录	14
2.2.4 文件	15
2.3 文件系统的内部实现	31
2.3.1 概述	31
2.3.2 文件子系统	32
2.3.3 文件系统高速缓冲	54
2.4 小结	64
第 3 章 进程系统	65
3.1 用户观点中的进程系统	65
3.1.1 进程概念	65
3.1.2 进程状态	69
3.1.3 进程控制	74
3.1.4 命令附录	78
3.2 进程系统的内部实现	80
3.2.1 进程结构	80
3.2.2 进程存储管理	86
3.2.3 进程上下文	92
3.2.4 进程控制	98
3.2.5 进程调度	108
3.2.6 关于线程	110
3.3 小结	112

第 2 部分 UNIX 系统使用

第 4 章 UNIX Shell	114
4.1 概述	114
4.2 输入、输出的重定向与管道	115
4.2.1 重定向	116
4.2.2 管道	120
4.3 环境变量	120
4.4 命令的生成	125
4.4.1 命令文件的查找	126
4.4.2 通配符	127
4.4.3 特殊字符的处理	128
4.5 Shell 编程	129
4.5.1 Shell 命令文件	129
4.5.2 Shell 变量	130
4.5.3 命令控制结构	132
4.5.4 Shell 内部命令	142
4.5.5 Shell 执行方式控制	143
4.6 其他 Shell 介绍	145
4.6.1 层次化 Shell	145
4.6.2 csh 和 ksh	146
4.7 小结	147
4.8 命令附录	147
第 5 章 文本编辑程序	150
5.1 vi 入门	150
5.1.1 进入 vi	151
5.1.2 插入文本(Insert)	152
5.1.3 重复前一命令(Redo)	152
5.1.4 取消上一命令(Undo)	153
5.1.5 移动光标	153
5.1.6 删除文本>Delete)	154
5.1.7 搜索某个字符串(Search)	155
5.1.8 串替换(Substitute)	155
5.1.9 读入其他文件中的内容	157
5.1.10 执行其他 Shell 命令	158
5.1.11 行号	159
5.1.12 退出 vi	160
5.2 Vi 使用详述	160

5.2.1	启动 vi	161
5.2.2	移动光标	161
5.2.3	屏幕翻滚	163
5.2.4	插入文本	163
5.2.5	文件内容的复制	164
5.2.6	文件内容的删除	167
5.2.7	文本的取代	169
5.2.8	文本行的移动	170
5.2.9	串搜索	171
5.2.10	串替换	172
5.2.11	行的合并	174
5.2.12	编辑多个文件	174
5.2.13	退出 vi	175
5.3	vi 配置	176
5.3.1	终端类型的设置	176
5.3.2	选项设置	176
5.3.3	vi 中的宏功能	179
5.3.4	字符串缩写	180
5.3.5	永久性的环境设置	180
5.4	命令汇总	181
第 6 章	存储介质	185
6.1	存储空间使用情况	185
6.1.1	文件系统空闲情况统计	185
6.1.2	磁盘空间占用情况	187
6.1.3	用户占用文件系统空间统计	188
6.1.4	命令附录	188
6.2	存储介质上的文件系统	190
6.2.1	存储介质的组织方式	190
6.2.2	存储介质上文件系统的使用	193
6.2.3	命令附录	197
6.3	文件备份工具	201
6.3.1	dd 命令	201
6.3.2	cpio 命令	202
6.3.3	tar 命令	206
6.3.4	命令附录	208
6.4	DOS 格式软盘存取工具	214
6.5	小结	215

第 7 章 UNIX 环境下的网络	217
7.1 关于 Internet	217
7.1.1 概述	217
7.1.2 什么是网络互联	218
7.1.3 Internet 地址	220
7.1.4 域名服务	223
7.1.5 Internet 应用服务	225
7.2 文件传输	226
7.2.1 Internet 文件传输服务内部实现机制	227
7.2.2 文件传输命令 ftp 的使用	228
7.2.3 命令附录	236
7.3 远程登录	236
7.3.1 Internet 远程登录内部实现机制	236
7.3.2 远程登录命令 telnet 的使用	238
7.3.3 命令附录	241
7.4 电子邮件	242
7.4.1 Internet 电子邮件的内部实现机制	242
7.4.2 UNIX 环境下电子邮件系统的使用	245
7.4.3 命令附录	255
7.5 小结	255

第 3 部分 UNIX 系统管理

第 8 章 系统的启动与关闭	258
8.1 系统的启动过程	258
8.1.1 操作系统装入程序的加载	260
8.1.2 UNIX 系统核心程序的装入	260
8.1.3 系统内核的初始化	261
8.1.4 硬件的检测和配置	261
8.1.5 系统进程的生成	262
8.1.6 系统初始化 Shell 命令文本的执行	262
8.1.7 各终端进程的生成	266
8.1.8 启动过程中的人工干预	268
8.2 系统的关闭	268
8.3 小结	269
8.4 命令附录	269
第 9 章 安全性管理	271
9.1 安全性问题概述	272
9.1.1 系统登录	272

9.1.2 文件的访问权限	272
9.1.3 文件加密	273
9.1.4 受限制的 Shell	274
9.1.5 网络安全性	274
9.1.6 设备的安全性	275
9.2 用户帐号的管理	275
9.2.1 帐号管理	276
9.2.2 组的管理	280
9.2.3 口令维护	281
9.3 超级用户	284
9.3.1 超级用户的权限	284
9.3.2 转变至超级用户角色	285
9.4 小结	286
9.5 命令附录	287
第 10 章 文件系统管理	292
10.1 本地文件系统的管理	292
10.1.1 文件系统存储空间的管理	292
10.1.2 文件系统完整性管理	296
10.1.3 命令附录	298
10.2 分布式文件系统管理	300
10.2.1 分布式文件系统的实现机制	302
10.2.2 分布式文件系统的管理	304
10.2.3 命令附录	309
10.3 小结	313
第 11 章 时间和作业调度	314
11.1 时间	314
11.1.1 系统时间	314
11.1.2 文件时间	316
11.1.3 命令附录	318
11.2 作业调度	320
11.2.1 cron 机制	320
11.2.2 at 和 batch	322
11.2.3 命令附录	324
11.3 小结	326
 第 4 部分 UNIX 系统开发	
第 12 章 C 编译系统	329
12.1 编译过程概述	329

12.2 cc 命令	333
12.2.1 cc 基本用法	333
12.2.2 常用选项	336
12.3 链接处理	339
12.3.1 缺省设置	339
12.3.2 标准库函数的链接	340
12.3.3 静态库及动态库的建立	341
12.3.4 链接程序搜索目录	344
12.3.5 动态链接程序搜索目录	346
12.4 小结	347
12.5 命令附录	348
第 13 章 调试系统	350
13.1 程序调试概述	350
13.2 sdb 的启动	351
13.3 sdb 命令的使用	354
13.3.1 源程序的显示和搜索	354
13.3.2 函数调用层次显示	357
13.3.3 存储单元内容的显示和设置	357
13.3.4 程序执行控制	361
13.3.5 汇编级的程序调试	365
13.3.6 其他命令	366
13.4 小结	367
13.5 命令附录	367
第 14 章 程序维护工具 make	369
14.1 make 工作机制	369
14.1.1 make 基本工作机制	370
14.1.2 make 描述文件格式	371
14.1.3 依赖关系的确定	373
14.2 make 变量	376
14.2.1 用户自定义变量	377
14.2.2 make 预定义变量	378
14.2.3 make 变量同 Shell 环境变量的交互作用	381
14.2.4 变量的条件赋值	381
14.2.5 变量值的替换	382
14.3 make 的后缀规则	383
14.3.1 后缀规则	383
14.3.2 可识别后缀名表	384
14.3.3 make 的标准后缀规则	385

14.4	静态库的维护.....	387
14.5	小结.....	389
14.6	命令附录.....	390
第 15 章	源文件版本维护工具 SCCS	392
15.1	概述.....	393
15.2	历史文件的生成和管理.....	395
15.2.1	历史文件的生成.....	395
15.2.2	历史文件的管理.....	397
15.3	文件的提取.....	401
15.3.1	只读方式的提取.....	401
15.3.2	修改方式的提取.....	403
15.3.3	新生成的 delta 的撤消	405
15.4	新版本的加入.....	405
15.5	其他辅助工具.....	406
15.5.1	历史文件信息显示.....	406
15.5.2	delta 合并	407
15.5.3	delta 的删除	407
15.5.4	注释的修改.....	408
15.5.5	what 命令	408
15.5.6	版本间差别的显示.....	409
15.5.7	Help 命令	409
15.6	SCCS 和 make 之间的交互	410
15.7	小结.....	411
15.8	命令附录.....	411

第 1 部分 UNIX 系统基础

为时

在现代计算机系统中,操作系统的重要性是不言而喻的。一台没有操作系统的机器同一堆废铜烂铁没有什么两样,不能完成任何工作,起不到任何作用。而一种优秀的操作系统,则能充分发挥计算机硬件强大的功能,使之能方便地、高效地完成用户所需完成的工作。UNIX 操作系统就是这样一种系统。从 20 世纪 60 年代末 UNIX 诞生以来,经过近 30 年的发展,今天的 UNIX 已经在计算机技术领域获得了极为广泛的应用,在科研、商业、工业、办公自动化等诸多方面都发挥着极其重要的作用。这些都从各个不同的侧面反映出 UNIX 系统是一种非常重要、也是一种非常优秀的系统。

虽然现在预测在计算机领域中 UNIX 将一统天下尚为时过早或过于乐观,但随着信息时代的到来,UNIX 的重要性确是日渐突出。广大用户应能熟练地使用 UNIX 系统,充分发挥其提供的强大功能,进而应能对整个系统进行维护和管理,并熟练地在此种环境下进行应用程序的开发。为此,用户对 UNIX 的基础知识至少应有一个大致的了解,这样在使用、管理和开发时就能做到胸有成竹,兵来将挡。不管 UNIX 系统怎么复杂,怎么千变万化,有了 UNIX 系统的基础知识,就能够做到以不变应万变,任何情况下,总能“逢凶化吉”、“左右逢源”。

本部分的内容分为 3 章:

第 1 章“UNIX 简介”,本章主要对 UNIX 的过去、现在和将来作一简要介绍和讨论,从中读者可以获得对 UNIX 的一个大致认识。

第 2 章“文件系统”,UNIX 最让人称道的一个重要方面就是其高效的、强大的文件系统。这种层次化树型的文件系统是 UNIX 的基石。系统程序、用户数据、各种外围设备等都通过文件系统这一概念而得到统一。本章将介绍 UNIX 文件系统的结构、维护和管理等。

第 3 章“进程系统”,用户使用计算机的目的是为完成某种任务。而进程是 UNIX 系统中进行任务调度的基本结构。本章将对 UNIX 的进程结构、调度等进行介绍。

第 1 章 UNIX 简介

本章将对 UNIX 的发展历史、现状及其未来的发展趋势作一介绍,希望借此能帮助读者建立起对 UNIX 的一个整体的认识,对 UNIX 的过去、现在和将来有一个比较好的了解,为用好、管好 UNIX 系统打下一个良好的基础。

1.1 UNIX 的发展历史

提起玩计算机游戏,大部分人肯定会热血沸腾,激动不已。但是当他们得知游戏这种人类的娱乐形式对计算机技术的发展起过极为重要的作用时肯定会感到异常惊讶!计算机技术的发展和游戏有什么关系?但情况确实是这样,游戏对计算机技术的发展起过极为重要的作用。UNIX 就是从开发一个游戏程序而发展起来的。

1964 年,美国 AT&T Bell 实验室、通用电气公司和麻省理工学院(MIT)合作开发一个名为 Multics 的、用于大型机的操作系统。Multics 系统的主要目标是要提供一种多用户多任务的计算环境,方便地实现硬软件资源的共享。经过 5 年的努力,1969 年 Multics 终于开始在一种称为是 GE645 的机器上运行。但 Multics 所追求的目标实在是太庞大、太复杂,以至于它的开发者们都不知道到底要把它做成一个什么样的东西。在这种情况下,Bell 实验室的开发人员退出了这个系统的开发。

回到 Bell 实验室的 Computing Science Research Center 后,这批极富才华的开发人员开始陷入一种困境:没有一个方便的交互计算服务可用了。于是他们着手设计自己的计算服务环境。Ken Thompson 和 Dennis Ritchie 在中心其他人的协助之下给出了这种环境的文件系统的规范说明。然后 Thompson 开始着手实现这个文件系统的模拟软件,并对请求调页的存储管理技术进行了模拟。同时他还开始编制一个名为 Space Travelling 的游戏程序。这个游戏程序是在 Honeywell 635 机器上用 Fortran 语言实现的。在编制这个程序的过程中,Thompson 遇到了几个麻烦。一是 Honeywell 635 机器图形能力较差,游戏中的飞船难以控制;二是 Honeywell 635 的处理能力、速度较慢,不能很好地运行这个游戏。Thompson 为了编好这个游戏,玩好这个游戏,到处寻找,希望能找到一台比较合适的机器。功夫不负苦心人,最后他终于找到了一台堆在角落里无人理睬的 PDP-7 机器。

PDP-7 机器的图形功能在当时还算不错,其速度及性能也还过得去。Thompson 在这台机器上继续研究他的“游戏事业”期间,对这台机器的硬件配置有了充分详细的了解,这为他后来的工作打下了一个良好的基础。机器条件虽然好了些,但在编程时却有一个非常大的麻烦。Thompson 找到的这台 PDP-7 机器没有一个好的开发支撑环境,在这台机器上写程序要先在原先的那台 Honeywell 635 机器上进行交叉汇编,然后把结果纸带送到 PDP-7 机上输入。真是麻烦至极。Thompson 和 Ritchie“一怒之下”,下定决心要在这台 PDP-7 机器上实现他们先前所规划的系统。经过一番艰苦的努力,这个系统终于实现了,

这就是最初的 UNIX。

UNIX 这个名字是由 Bell 实验室的 Brian Kernighan 想出来的。联想一下前面提到的 Multics 系统,再想一想英文中前缀 Multi 和 Uni 的中文意义,我们能够猜想出 Kernighan 的良苦用心。Multi 是大的意义,大而且繁,这是 Multics 的特点;Uni 是小的意义,小而且巧,则是 UNIX 系统初期开发者的目标和希望。当然在初期,UNIX 也确实是一个非常小巧的系统。

之后,UNIX 开始在 Bell 实验室内逐渐普及开来,同时它的设计者们也逐步对之进行完善、扩充,并提供了许多有用的实用程序,使之更好用、功能更强大。到 1972 年 6 月,《UNIX Programmer's Manual》第二版上,开发者们高兴地宣布:The number of UNIX installations has grown to 20, with more expected. 1973 年,开发者们用 Ritchie 新发展起来的 C 语言重写了 UNIX。这虽然导致了其速度有所下降(据估计是 20%~40%),但这一举动却是操作系统开发中的一个重要里程碑,它向人们宣布,操作系统的开发中汇编语言已不再是主宰了。改写后的 UNIX 系统获得了极好的可移植性,并逐渐被移植到 PDP-11 等当时新出现的机器上。此时,Bell 实验室内 UNIX 的装机量已达到 25 台。

随后,Bell 实验室开始向大学、科研机构免费发放 UNIX 的源代码。这一举措的直接后果是导致大批优秀的计算机技术人才借助于 UNIX 这个系统而被培养出来。同时在 AT&T 沿着商业应用方向发展 UNIX 系统的同时,加州大学 Berkeley 分校的面向科研和工业应用的 BSD(Berkeley Software Distribution)版 UNIX 也发展起来了。从此开始,UNIX 的发展就出现了两种不同的技术走向,同时也造成了两种不同版本的 UNIX 之间的一些不兼容。但他们都能很快地在自己的系统中溶进对方新发展起来的新技术,或者开发出与对方某种新技术相应的新技术。这些又导致 UNIX 获得更充分的发展。

1977 年,UNIX 的装机量达到 500 台,这个数目是 1973 年的 20 倍,其中的 $\frac{1}{4}$ 是在大学中的计算机上运行。也就是在这一年,UNIX 开始在非 PDP 系列的其他机种上运行,这个系统就是 Interdata 8132。从此开辟了 UNIX 向其他巨型机、大型机、中型机、小型机、工作站和微机移植的新局面。

UNIX 系统版本的演变比较复杂。从 AT&T 的 UNIX 发展来看,早期比较著名的版本有第六版和第七版。各厂商根据客户的不同要求,分别对其所销售的系统进行改造,加入特定的功能和应用,这样就形成了 UNIX 的许多变体。1977 年到 1982 年,AT&T 把这些变体综合成一个单一的系统,这就是 UNIX System III。随后于 1983 年初,AT&T 宣布了 UNIX System V。而 System IV 是 Bell 实验室内使用的一个过渡性产品,一直没有被公布。20 世纪 80 年代末,AT&T 对 System V 的命名重新标准化,分别是 System V Release X,简记为 SVRx,如 SVR3,SVR4 等。这些版本均是以第七版为基础而发展起来的。

除了上面所说的 AT&T 的 UNIX 版本之外,还有许多其他版本和分支的 UNIX,如 BSD 版的 UNIX,Microsoft 的 Xenix。Xenix 主要运行在 x86 系列的机器上,其功能同 AT&T 的 SVR2 类似。

UNIX 系统从一开始就显示出其强大的优越性和生命力,从而使得众多的高校、科研

单位和产业集团在 UNIX 系统的研究和发展上投入了大量的人力、物力和财力。这些给 UNIX 的发展注入了巨大的活力,同时也因为机构太多而不容易协调,形成了多种不同的版本和标准。从 20 世纪 80 年代末到 90 年代初,逐渐形成了两个主要的集团。其一为 UI (即 UNIX International),该集团的主要成员为 AT&T, Sun MicroSystem, 东芝, 富士通等;另一为 OSF (Open Software Foundation), 主要有 DEC, IBM, HP, NEC 等。它们所奉行的是不同的技术路线,在技术和商业方面进行着激烈的竞争。竞争一方面促进了 UNIX 的发展,增强了其功能;另一方面也因其不兼容而给用户带来许多困难。但事物的发展是曲折的,“分久必合,合久必分”,在 UNIX 的发展过程中也得到了充分的体现。最近,我们高兴地注意到,两大集团正向“合”的方向发展着。

1.2 UNIX 现状

上一节,我们对 UNIX 的发展历史作了一个较为详尽的介绍。本节我们将向读者介绍一下当前 UNIX 系统在功能和技术方面的一些特点。

前面说过,操作系统在整个计算机系统中起着至关重要的作用。那么它到底有哪些作用呢? UNIX 系统是一个多用户、多任务的分时系统。也就是说,在它的管理之下,一台机器可供多个用户“同时”使用,每一个用户都能执行一个或多个程序。为实现这一点,UNIX 必须对整个计算机进行控制,在此基础上,为每个用户所执行的各个程序分配相应的资源。这些资源中最重要当然是 CPU,即中央处理器。程序是由一条条指令构成的,而没有 CPU,这些指令将无法执行,只能是“死”的、静态的,完成不了任何工作。另一些资源有内存(RAM)、外存(磁盘)等,它们都是程序正常运行所必不可少的。围绕这些资源的分配,发展出一整套的技术,后面我们对此还要进行介绍。

计算机要对整台机器的各方面进行控制,如前面说过的 CPU、内存、外存,以及其他许多外围设备,如软盘驱动器、光盘驱动器、打印机、磁带、终端、网卡等。当然,进行这种控制的目的是为了向用户提供一个高效而又方便地使用这些资源的接口。所有这些控制程序就构成了整个 UNIX 的核心——内核。程序完成的所有工作,都要直接地或间接地使用这个内核所提供的服务来完成。UNIX 提供这些服务的形式是系统调用(同函数调用类似,但又不是普通的函数调用)。

仅有内核还不行,因为仅这样还不能为用户提供一个方便的开发、使用环境。为此 UNIX 系统还提供了大量的系统程序,也就是一些标准的服务,如编译程序、文本编辑程序、命令语言、文件打印服务、记帐服务、系统管理服务,等等,不胜枚举,但他们都是以内核提供的服务为基础的。

在随系统自带的标准服务之上,根据各行各业用户具体工作需求的不同,可以在标准的系统之上加上一些独立厂商开发的应用软件或应用开发软件。如各种数据库管理系统、图形应用和开发环境、网络应用和开发环境,等等。在这些开发环境的基础上,我们可以开发出自己的大量满足不同需求的应用。

综上所述,我们可以清楚地看出当今 UNIX 系统在结构上的一些特点,如图 1.1 所示。