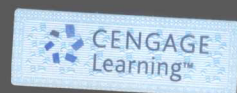


操作系统

原理、设计与应用

Brian L. Stuart 著

葛秀慧 田浩 刘展威 等译



PRINCIPLES OF OPERATING SYSTEMS

Design and Applications

清华大学出版社

世界著名计算机教材精选

操作系统

原理、设计与应用

Brian L. Stuart

著

清华大学出版社

北京

北京市版权局著作权合同登记号 图字 01-2009-3864 号

Principles of Operating Systems: Design and Applications

Brian L. Stuart 著, 葛秀慧 田浩 刘展威 等译

Copyright © 2009 by Course Technology of Cengage Learning.

Original edition published by Cengage Learning. All Rights reserved.

本书原版由圣智学习出版公司出版。版权所有,盗印必究。

Tsinghua University Press is authorized by Cengage Learning to publish and distribute exclusively this simplified Chinese edition. This edition is authorized for sale in the People's Republic of China only (excluding Hong Kong, Macao SAR and Taiwan). Unauthorized export of this edition is a violation of the Copyright Act. No part of this publication may be reproduced or distributed by any means, or stored in a database or retrieval system, without the prior written permission of the publisher.

本书中文简体字翻译版由圣智学习出版公司授权清华大学出版社独家出版发行。此版本仅限在中华人民共和国境内(不包括中国香港、澳门特别行政区及中国台湾)销售。未经授权的本书出口将被视为违反版权法的行为。未经出版者预先书面许可,不得以任何方式复制或发行本书的任何部分。

Cengage Learning Asia Pte. Ltd.

5 Shenton Way, #01-01 UIC Building, Singapore 068808

本书封面贴有 Cengage Learning 防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

操作系统原理、设计与应用/(美)斯图尔特(Stuart, B. L.)著;葛秀慧等译. —北京:清华大学出版社, 2010.6

(世界著名计算机教材精选)

书名原文: Principles of Operating Systems: Design and Applications

ISBN 978-7-302-22318-4

I. ①操… II. ①斯… ②葛… III. ①操作系统—教材 IV. TP316

中国版本图书馆 CIP 数据核字(2010)第 055944 号

责任编辑:龙啟铭

责任校对:李建庄

责任印制:杨 艳

出版发行:清华大学出版社

<http://www.tup.com.cn>

社 总 机:010-62770175

投稿与读者服务:010-62795954, jsjic@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

地 址:北京清华大学学研大厦 A 座

邮 编:100084

邮 购:010-62786544

印 刷 者:北京市清华园胶印厂

装 订 者:三河市漂源装订厂

经 销:全国新华书店

开 本:185×260 印 张:30.5

字 数:752 千字

版 次:2010 年 6 月第 1 版

印 次:2010 年 6 月第 1 次印刷

印 数:1~3000

定 价:59.00 元

产品编号:033749-01

译者序

操作系统是计算机科学课程中的核心课程。本书内容全面，实践性强。作者经过多年的构思与讲授操作系统的课程，才完成了这部内容翔实、充满启迪的教科书。学生不仅能学到基本的原理，还能通过各种不同风格操作系统的具体的代码来印证这些原理，从而即学到理论，又能编程实践，达到了理论与实践兼顾的学习目标。另外，这本书最大的特色就在于为讲授这门课程的教师提供了最大的灵活性，教师可以根据本校学生的实际来自由构成自己的讲义，这也是其他教材所不具备的。

本书的内容主要包括 7 个主题：操作系统简介、进程、存储器、I/O 设备、文件系统、操作系统安全和分布式操作系统。每一部分都从多个视角分析了主题的基本原理、应用分析、详细设计，以及 *Inferno* 和 *Linux* 的实现。在详细讨论 *Inferno* 和 *Linux* 的这些章节中，主要侧重与内核中相对较小的一部分，这些部分用于说明在原理章节中涉及的技术与原理。书中介绍的每个函数都分成了小段，然后以更详细的方式描述这些小段。这样就能详细地学习独特内核的一些关键内容。本书除了基本的原理之外，还重点介绍了 9 个操作系统 *CTSS*、*Multics*、*RT-11*、第 6 版 *UNIX*、*4.3BSD*、*VMS*、*Windows NT*、*TinyOS* 和 *Xen*。这些示例操作系统，作为主线贯穿整本书。通过重点研究这些操作系统示例，从高层次的方式来研究操作系统的开发者如何将标准技术转化为实践。从而在实践的基础上再次提升理论。使理论和实践再次相互印证。

另外值得强调的是这本书的适用性，它为需要深入研究 *Linux* 或者 *Inferno* 内部结构的专业人员提供了大量的、翔实的、来自真实世界的代码，也为初学者提供了从浅入深、循序渐进学习操作系统原理的丰富实例。本书既可以用于普通操作系统的学习，也可以作为高级操作系统的教材。同时这本书每章之后都有印证所学内容的大量、有趣的练习。要求学生自己动手来建立自己的 *Inferno* 和 *Linux*。

在本书的翻译过程中，我们力求忠实于原著，原汁原味的体现作者的内容。本书由葛秀慧、田浩、刘展威、段建勇主译，张桂香、王顶、刘秋红、刘朝晖、焦仁普、朱书敏、盖俊飞、李超、郭立甫、刘玲、李帅、程香萍、刘艳霞、黄丹雅、郑宏亮、刘丽华、杨希、晁静雅、赵倩、刘薇、张琳、陈宗斌、陈红霞、张景友、易小丽、陈婷、管学岗、王新彦、金惠敏、张海峰、徐晔、戴锋、张德福、李振国等也参加了本书的翻译与校稿。因为译者水平有限，书中的错误和不妥之处在所难免，敬请广大读者批评指正。

前言

在 20 年前，当我在研究生院时，就已萌生了写这本书的想法。1986 年夏天，在 David Cohen 博士的指导下，一群学生在一起进行高级操作系统的学习讨论。开始时，我们想要编写操作系统，但很快我们的重点发生了转移，最后决定编写操作系统的教科书。正如事情经常发生的情况一样，我们几乎没有实现我们的初衷。然而，通过这种方式，我们对关于如何组织操作系统的教材进行了许多富有成果的讨论，通过扩展，也找到了如何进行操作系统教学的方法。

这些想法与多年讲授操作系统课程的实践相结合，使我看到，讲授操作系统有几种不同的做法。同时也使我认识到现有的课本都没有真正提供这种灵活性，使教师可以根据自己的需求来讲授操作系统这门课程。编写本书的动机就是为每位教师提供这种灵活性。

组织

本书主要包括七个主题，首先介绍操作系统的重要历史、结构和组织、系统调用和引导。在此之后，较深入地分析了操作系统的主要功能：进程、存储、I/O 设备和文件系统。最后两个主题是安全性和分布式系统。

首先介绍的五个主题依次在前四章中进行介绍，每一章都从多个视角分析相关的主题。第 1 章依次介绍了与资源管理相关的基本原理。在这些章中，我介绍了相关的问题以及为了解决这些问题所使用的一些标准技术。在某些情况下，着重基本原理的章节也包括相关问题的讨论。例如，在第 5 章中的互斥和死锁主题是与进程管理一起进行讨论的，因为这两个主题都与进程间通信相关。第 2 章依次概括性地介绍了操作系统的历史和目前的操作系统。9 个操作系统的示例集合主要包括：CTSS、Multics、RT-11、第 6 版 UNIX、4.3 BSD、VMS、Windows NT、TinyOS 和 Xen。之所以重点研究这些操作系统，主要着眼点在于要以高层次的方式来研究这些操作系统的开发者如何将提供的标准技术转化为实践。在第 3 章和第 4 章的每个小节中，更深层次地研究实现的细节。在第 3 章中选取了 Inferno 中的部分代码进行讨论，在第 4 章中选取了 Linux 中的部分代码进行讨论。

但是，在第 21 章和第 22 章中，并没有采用前面的这种模式。这两章分别讨论了安全性和分布式系统。因为这两个主题的每一个主题都可以写成一本书，所以必须选择性地讨论这两个主题。这两章只介绍了一些有代表性的技术。在第 21 章中，也讨论了更受限的示例集来说明如何应用安全技术。

在附录 A 中介绍了如何创建宿主 Inferno 的内核镜像。为了完成修改 Inferno 内核这样的任务，创建镜像就是解决方案的一部分。以这种方式创建的内核镜像可以像应用程序一样运行在现有的主机操作系统之上。附录 B 教给用户如何创建本地的 Inferno 内核。特别是提供了用于 x86 PC 上创建启动盘镜像的必要步骤。这个镜像可以写入到软盘或者用于创建启动 CD-ROM。

读者群

这本书的读者群主要分为两个群体。第一个群体是需要理解 Linux 或者 Inferno 内部的从业人员，这样可以加深对基本原理的理解。在这个群体中的读者可以在这本书中大量接触自己所使用操作系统的经验。他们也能看到关于操作系统操作的一些熟悉的概念和技巧。对于这个读者群，这本书讨论的一些数据结构他们可能并不感兴趣，因为任何数据结构的图书都能提供这些知识。同样，计算机组成这样的图书可以提供硬件的所有知识。对于这个读者群，各章的基本原理将有助于填补他们知识的任何空白，Linux 和 Inferno 的相应章节将提供这些操作系统的内部介绍。

这本书的第二个群体是教师和学习操作系统课程的学生。普通操作系统和高级操作系统课程都可以使用这本教材。学习操作系统课程之前，先决条件是已学习了数据结构课程和计算机组成课程。这本书的一些章节都认为，读者已经学习了这些知识。书中的一些章节也涉及程序设计语言、它的编译器和运行环境。如果学过程序设计语言及其编译的课程固然很好，但是如果没学过，也不需要再去学习这些知识。

本书的使用

在大多数情况下，读书都是从开始到结束来通读一本书。读者可以用这种方式来读这本书。然而，大多数教师不会使用这种方法。相对而言，在课堂上使用这本书最有效的方式是从每个主要章节中选择资料。全套资料的提供使每个教师都能灵活地选择最适合的课程风格及个人喜好的资料。

讲授操作系统课程的教师当然会选择支持课程一般教学方法的章节。例如，一种做法可能侧重于抽象的概念和技术。为了理论深度，在实际硬件上实现相关技术的内容可能被忽略。另一种做法可能权衡所用的时间，将用于介绍基本原理的时间通过分析一些真正的操作系统来说明这些原理的应用。介绍性课程通用的办法是基本原理的学习与操作系统的实现相结合。在这种类型的课程中，学生常常需要对他们所学习的操作系统进行修改。这种深入现有操作系统内部的经验一般是在高级操作系统课程中积累的。

现在，考虑为每种不同风格的课程推荐章节集合。对于基本原理课程，应把重点放在第 1 章、第 5 章、第 9 章、第 13 章、第 17 章、第 21 章和第 22 章。教师为了补充这些内容可以在第 2 章、第 6 章、第 10 章、第 14 章和第 18 章中选择示例。当然这些章可以指定为课外读物。这类课程的一种特别提法值得高度注意。在计算机课程 2001 中，指定课程 CS220 还确定了一些应该包括的具体主题。以下的各节可以很好地覆盖这些主题：1.1~1.4、1.6、5.1~5.8、9.1~9.3、9.5~9.6、13.1~13.4、13.7.1、17.1~17.4、17.7、21.1~21.2、21.4~21.5、22.1 和 22.3~22.4。当然，教师也可以不局限于这些章节。在其他章节中的资料也可以成为 CC2001 推荐的补充资料。

如果使用原理-分析方法，在课堂上介绍时，应该包括第 2 章、第 6 章、第 10 章、第 14 章和第 18 章。如果在课堂上对其他资料需求太大，教师可以选择在原理章节应该介绍那些主题。例如，最短作业优先最优的证明和名称空间的形式都可以放心地忽略。本书不包括的其他现有操作系统的示例都可以作为课外阅读作业。

操作系统课程具有实践部分，这也是很常见的结构。在此方法中，学生们普遍期望熟悉和修改现有的操作系统。通常情况下，使用的操作系统应该是相对较小而又便于管理。本书也为这种类型课程提供了资料。教师可以在现有的两个操作系统 Inferno 和 Linux 中进行选择。教师使用 Inferno，除了基本原理之外，还应该选择第 3 章、第 7 章、第 11 章、第 15 章和第 19 章。同样，选择 Linux，就应该选择第 4 章、第 8 章、第 12 章、第 16 章和第 20 章。在涉及代码分析的章节中，在课堂上重要的不是分析太多的代码。经验表明，不需要在开始时分析所有的代码，这样做没有任何益处。最好是分析较少量的代码，这部分代码说明重要的知识点，然后让学生将代码其余部分作为课外阅读和作业。能够阅读和理解真正的代码在这种组织方式是最有效的。

最后要考虑的课程是高级操作系统课程。考虑到高级课程的本质，可以使用本书的几个章节。如果学生已经学习过入门课程，但并没有涵盖在本书中讨论的所有基本原理，那么在课堂上，可以抽时间讲解这些基本原理并可以更深入地剖析任何基本原理。同样，概括性章节是一个良好的出发点，可以对其中任何一个基本原理进行全面的分析或者更多地围绕实际操作系统进行分析。最后，对于那些已学习过入门课程，但没有学习操作系统内部知识的经验的学生，Inferno 和 Linux 详细内容将是提供这种经验的一个起点。这些不同的课程设计的总结，如下表所示：

章	只选基本原理	原理与示例	原理与 Inferno	原理与 Linux
1	√	√	√	√
2	•	√		
3			√	
4				√
5	√	√	√	√
6	•	√		
7			√	
8				√
9	√	√	√	√
10	•	√		
11			√	
12				√
13	√	√	√	√
14	•	√		
15			√	
16				√
17	√	√	√	√
18	•	√		
19			√	
20				√
21	√	√	√	√
22	√	√	√	√

√：包括此章。

•：包括此章选择的主题。

本书特色

本书的每一部分都从多个视角分析了不同的主题：基本原理、应用分析、详细设计和 **Inferno** 和 **Linux** 的实现。基本原理的章节包括了一些重要特性。以半正式化算法方式分析的一些技术非常适用于实现。这些算法在印刷上与众不同。在一些情况下，使用详细示例说明的技术也有不同的格式。最后，这些章节都包括一些知识回顾可以帮助读者建立上下文的联系。在详细讨论 **Inferno** 和 **Linux** 的这些章中，主要侧重与内核中相对较小的一部分，这些部分用于说明在原理章节中涉及的技术与原理。书中介绍的每个函数都分成了小段，然后以更详细的方式描述这些小段。这样就能详细地学习独特内核的一些关键内容。这些章节也包括练习，要求学生自己动手来建立自己的 **Inferno** 和 **Linux**。除了这些基本特性之外，还讨论了下面的一些重要主题：

第 1 章：背景、历史、组织、引导与系统调用。

第 3 章与第 4 章：**Inferno** 和 **Linux** 的历史、结构、初始化和系统调用。

第 5 章：进程表示、进行调度、上下文切换、互斥和死锁。

第 7 章与第 8 章：在 **Inferno** 和 **Linux** 中的进程表示、创建和调度，还包括在 **Linux** 中新的 $O(1)$ 调度程序。

第 9 章：地址变换技术、动态分配技术（包括兼容性示例）、交换和分页。

第 11 章：在 **Inferno** 中的池/块分配和垃圾回收。

第 12 章：在 **Linux** 中的区域/slab 分配、页表和缺页。

第 13 章：I/O 硬件概述、控制设备技术和选择性设备管理技术。

第 15 章：在 **Inferno** 中的设备驱动程序结构、并行端口驱动程序、键盘驱动程序和 IDE 磁盘驱动程序。

第 16 章：在 **Linux** 中的两部分中断处理程序、并行端口驱动程序和软盘驱动程序。

第 17 章：名称空间、存储管理技术和日志文件系统。

第 19 章：**Inferno** 的文件服务器设计、Styx 协议和 kfs 文件系统。

第 20 章：**Linux** 的虚拟文件系统和 EXT3 文件系统。

第 21 章：基本的安全技术与威胁、橙皮书、加密和 **Multics** 的保护环。

第 22 章：资源共享、同步操作、集群、网格、分布式时钟和选举算法。

操作系统示例

这本书详细讨论的两个操作系统每一个都有自己的优势。**Inferno** 是一个相对较小的操作系统，从而更容易把握其细节。**Inferno** 的设计还有些独特，它不仅是易用的本地操作系统，而且还可以作为应用运行在宿主操作系统之上。宿主的兼容性使学生更容易地在自己的计算机上安装此操作系统。当学生将调试这个操作系统的新版本作为作业时，这也将简化测试过程。另一方面，**Linux** 也是非常相似的系统，它一点也不比 **Inferno** 逊色，同样，研究 **Linux** 可以使学生积累更多应用方面的经验。它也能提供一些在 **Inferno** 中不存在的更复杂技术的示例。

本书使用的 Inferno 版本是 2007 年 5 月 10 日发布的版本。在 Vita Nuova 网站上可以找到最近发布的 Inferno 版本，网址是：<http://www.vitanuova.com>。目前开发和最新的修订记录在 Google code-hosting 网站上，网址是：<http://code.google.com>。我讨论的 Inferno 的编译与运行过程在附录中。

本书中使用的 Linux 内核版本是 2.6.18。最新和以前的 Linux 内核版本的主要网站的网址是：<http://www.kernel.org>。建立 Linux 内核的两个优秀信息源分别是 Brian Ward 的“Linux Kernel HOWTO”和 Kwan Lowe 的“Kernel Rebuild Guide”。

在第 2 章、第 6 章、第 10 章、第 14 章和第 18 章中的示例的一些源代码在网上也有。CTTS 操作系统的全部源代码的网址是：<http://www.piercefuller.com/library/ctss.html>。Multics 的部分源代码的网址是：<http://www.multicians.org>。第 6 版 UNIX 及以前版本的网址是：<http://tuhs.org>。UNIX 的 4.3BSD 版本是由国际自由计算任务组 (International Free Computing Task Force, IFCTF) 维护 4.3BSD-Quasijarus。本项目的网址是：<http://ifctfvax.harhan.org/Quasijarus>，在网站上也有可用的更新及以前的版本。TinyOS 的资源主要网址是：<http://www.tinyos.net>。最后，Xen 虚拟机管理器的主页是：<http://www.cl.cam.ac.uk/research/srg/netos/xen>。

源代码格式

本书的源代码格式使用 Knuth 和 Levy 的 CWEB 系统的结构文档。关键字与数据类型的排版是粗体。不在字母表中的标识符都排斜体；在字母表中的标识符的排版都是粗体。一些 C 语言的操作符是多字符序列，使用的是 ASCII 字符集中的字符。当进行介绍时，这些字符中的一些字符用一些通用的数学符号进行了替代，这样在某些情况下，能更直接表达具体的含义。C 语言的操作符的 ASCII 与在正文中的符号排版的对照表如下所示：

ASCII	符号
->	→
NULL	∧
~	~
^	⊕
==	≡
!=	≠
>=	≥
<=	≤
!	¬
&&	∧
	∨
>>	>>
<<	<<

除了这些惯例之外，CWEB 格式的八进制和十六进制常数是不同的。在 ASCII 中八进制常量表示为 0123，印刷时表示为 ° 123，十六进制常量 0x123 印刷表示为 # 123。

很奇怪的是，我们的印刷表示形式与在编译器输入形式并不相同，编译器的输入一般是文本编辑器中完成。虽然目前在 WEB 和 CWEB 之外，我们并没有看到这种差异，但这种情况已有了悠久的历史。一些语言允许在字符集中的变量用于操作符，这是因为并不是所有的安装都使用了相同的字符集。某些环境只有非常有限的字符集，一些甚至不包括小写字母。另一些环境具有极其丰富的字符集，允许程序员直接输入数学符号，如我们在这里使用的“不等于”、“小于等于”和“大于等于”。C 语言本身定义的 *trigraphs* 用来表示所有的字符集中没有的三个字符组合，并允许其在自身的环境中使用。除此之外，印刷代码与打印出来的代码也有不同这种情况也十分普遍，就像印刷文字与打字机的输出也有不同一样。本着这种精神，Algol 68 将印刷格式与编译器输入之间的差异形式化，印刷形式称为严格语言，编译器输入形式称为参考语言。如果注意到这一点就是很有趣的，Algol 是一种影响 C 设计的语言，而用 Algol 排版的实践影响了 Knuth 的 WEB 开发。这两种影响力已融合在 CWEB 中，在本书中我使用的印刷表示也是 CWEB。

致谢

这种规模的项目没有其他人的帮助是不可能完成的。有许多人的支持、鼓励和帮助是非常宝贵的。我首先要感谢我的妻子 Mary 和我的女儿 Rachel。她们毫无怨言地忍受了我的沮丧和无数个在办公室度过的夜晚。她们一直相信我和这个项目。当我写书时，她们尽量保持周边安静。其他家人和朋友的关心和支持也是我写作动力的源泉。

我还要感谢我在 FedEx 和 Memphis 大学的同事。无论作者如何想方设法安排时间（或者无论我如何试着安排时间），写这样一本书都会影响到其他工作。我尤其要感谢 FedEx 实验室的 Miley Ainsworth 对这个项目的支持。我还要感谢我在 FedEx 的同事 Tim Gregory，通过他的帮助，使我加深了对 TinyOS 的理解，并且还帮我修订了与 TinyOS 相关的内容。

我还要感谢在过去几年中听我讲授操作系统的学生。他们使用了这本书的测试主题。他们使用了部分的草稿、容忍了打字排版错误和尴尬的语法。不管怎样，许多人都表示支持和鼓励这个项目。我也要感谢少数学生提供了有益反馈。这些学生包括 Bob Bradley、Jim Greer、Taliesin Penfound 和 Debbie Travis。

我也与几个操作系统社区进行了联系，他们提供了宝贵的帮助和反馈。他们是 Vita Nuova Holding Ltd. 的 Charles Forsyth。前 Multics 开发组成员和 multicians.org 网站维护者 Tom Van Vleck。HP OpenVMS 组的 Stephen Hoffman。New South Wales 大学的 Digby Tarvin。我还要感谢 Saint Xavier 大学的 Jim Aman、Western Oregon 大学的 Charles Andeson、Tennessee at Martin 大学的 Bob Bradley、New Jersey City 大学的 Thomas Liu、Carleton 大学的 Chung-Horng Lung、Minnesota 大学的 Jon Weissman 和 Purdue 大学的 Dongyan Xu。他们仔细审查早期的草稿使这些资料的质量更高。

最后，我还要感谢 Course Technology 的每个成员。我没有办法列出对这个项目做出贡献

献的人员名单。但是，必须提到几个名字。首先，我要感谢 Mary Franz，他首先看到了这本书的潜力，并帮助我完成了早期的准备过程。还有三个人与我一起共同完成了整个项目，我最感谢 Alyssa Pratt、Jim Markham 和 Matt Hutchinson 为这本书所做的一切工作，他们的付出，使这本书能得以出版。每个人的投入与参与使这本书变得更好。书中的任何不足，都是与我自己有关。我感谢所有人的支持，才使这本书得以出版。

Brian L. Stuart

目 录

第 1 章 操作系统简介	1
1.1 什么是操作系统	1
1.1.1 资源管理者	2
1.1.2 服务提供者	2
1.1.3 虚拟机	2
1.2 操作系统的功能	3
1.2.1 进程	3
1.2.2 存储器	4
1.2.3 I/O 设备	4
1.2.4 文件系统	4
1.2.5 安全性	5
1.2.6 联网	5
1.2.7 用户接口	6
1.3 操作系统的历史	6
1.3.1 裸机	7
1.3.2 批处理操作系统	7
1.3.3 分时操作系统	8
1.3.4 分布式操作系统	9
1.4 组织操作系统的技术	9
1.4.1 单块设计	10
1.4.2 分层设计	10
1.4.3 微内核设计	10
1.4.4 虚拟机设计	11
1.5 引导	12
1.6 系统调用	13
1.6.1 系统调用示例	13
1.6.2 系统调用机制	14
1.7 本章小结	14
1.8 练习	15
第 2 章 操作系统示例	16
2.1 兼容分时系统	16
2.1.1 组织结构	16
2.1.2 引导	17

2.2	多路信息和计算服务	17
2.2.1	组织结构	17
2.2.2	系统调用	19
2.3	RT-11	19
2.3.1	组织结构	19
2.4	第 6 版 UNIX	20
2.4.1	组织结构	20
2.4.2	系统调用	21
2.5	虚拟内存系统	21
2.5.1	组织结构	21
2.5.2	引导	22
2.5.3	系统调用	22
2.6	4.3BSD	22
2.6.1	组织结构	23
2.6.2	系统调用	23
2.7	Windows NT	23
2.7.1	组织结构	24
2.8	TinyOS	24
2.8.1	组织结构	25
2.9	Xen	25
2.9.1	组织结构	25
2.10	本章小结	26
2.11	练习	26
第 3 章	Inferno 的结构与初始化	27
3.1	Inferno 的起源	27
3.2	基本概念	28
3.3	组织结构	30
3.3.1	基本体系结构	30
3.3.2	源代码组织结构	31
3.4	初始化	32
3.4.1	启动 Inferno	33
3.4.2	宿主操作系统的特定初始化	34
3.4.3	与宿主操作系统无关的初始化	37
3.4.4	启动分时	39
3.5	系统调用	40
3.6	本章小结	41
3.7	练习	41
第 4 章	Linux 的结构与初始化	42
4.1	Linux 的起源	42

4.2	组织结构	43
4.2.1	基本体系结构	43
4.2.2	模块	44
4.2.3	源代码组织结构	45
4.3	初始化	46
4.3.1	引导	47
4.3.2	特定处理器初始化	49
4.3.3	与处理器无关的初始化	52
4.3.4	启动分时	56
4.3.5	初始化管理级的初始化	56
4.4	系统调用	58
4.4.1	处理应用方的系统调用	59
4.4.2	处理内核方的系统调用	60
4.5	本章小结	60
4.6	练习	61
第 5 章	进程管理原理	62
5.1	进程的概念	62
5.2	实现进程	62
5.2.1	进程操作	63
5.2.2	进程状态	64
5.2.3	进程表	64
5.3	线程	65
5.4	调度	66
5.4.1	先来先服务	66
5.4.2	最短作业优先	66
5.4.3	轮转法	68
5.4.4	优先级调度	69
5.4.5	调整调度参数	71
5.4.6	两级调度	71
5.4.7	实时调度	72
5.4.8	嵌入式系统的调度	74
5.5	上下文切换	74
5.6	进程的创建与终止	76
5.7	临界区	77
5.7.1	中断控制	78
5.7.2	原子操作指令	78
5.7.3	Peterson 算法	79
5.7.4	信号量	80
5.7.5	管程	81

5.7.6	消息传递	82
5.7.7	示例	82
5.8	死锁	85
5.8.1	充分必要条件	86
5.8.2	处理死锁	86
5.9	本章小结	91
5.10	练习	91
第 6 章	进程管理示例	93
6.1	CTSS	93
6.1.1	进程状态	93
6.1.2	系统调用	94
6.1.3	调度	94
6.2	Multics	95
6.2.1	系统调用	95
6.2.2	进程状态	95
6.2.3	调度	95
6.3	RT-11	96
6.3.1	系统调用	96
6.3.2	进程状态	97
6.3.3	进程表	97
6.3.4	调度	97
6.4	第 6 版 UNIX	97
6.4.1	系统调用	97
6.4.2	进程状态	98
6.4.3	进程表	98
6.4.4	调度	99
6.5	4.3BSD	99
6.5.1	系统调用	99
6.5.2	进程状态与进程表	100
6.5.3	调度	100
6.6	VMS	101
6.6.1	系统调用	101
6.6.2	线程状态	101
6.6.3	调度	102
6.7	Windows NT	102
6.7.1	系统调用	103
6.7.2	线程状态	103
6.7.3	进程表与线程表	103
6.7.4	调度	104

6.8	TinyOS	105
6.9	Xen	105
6.10	本章小结	106
6.11	练习	106
第 7 章	Inferno 中的进程管理	108
7.1	Inferno 中的进程	108
7.2	进程的状态	109
7.2.1	内核进程	109
7.2.2	用户进程	110
7.3	进程的数据结构	111
7.3.1	内核进程表	111
7.3.2	内核进程表项	112
7.3.3	用户进程表	113
7.3.4	用户进程表项	113
7.4	进程的创建	115
7.4.1	解释进程创建指令	116
7.4.2	实现进程创建	116
7.5	进程的终止	119
7.6	进程调度	121
7.6.1	插入就绪表	121
7.6.2	从就绪表中删除	122
7.6.3	分时	122
7.6.4	运行时间片	124
7.7	本章小结	126
7.8	练习	126
第 8 章	Linux 中的进程管理	128
8.1	进程与线程	128
8.1.1	Linux 中的内核线程	128
8.1.2	进程间的关系	128
8.2	系统调用	129
8.3	进程状态	130
8.4	进程表	131
8.5	进程的创建	134
8.5.1	处理系统调用	134
8.5.2	创建进程	137
8.5.3	特定体系结构的步骤	139
8.6	进程调度	141
8.6.1	优先级	141
8.6.2	队列结构	142

8.6.3	时钟计时单元	143
8.6.4	调度程序	146
8.7	本章小结	151
8.8	练习	152
第 9 章	存储管理原理	153
9.1	存储层次结构	153
9.2	地址变换	154
9.2.1	基址/上下界寄存器	155
9.2.2	分段存储	155
9.2.3	分页存储	156
9.3	存储相关的服务	158
9.4	存储布局	159
9.5	内存分配技术	160
9.5.1	空闲空间管理	161
9.5.2	碎片	162
9.5.3	分区	162
9.5.4	选择策略	163
9.5.5	伙伴系统管理	165
9.6	过度分配技术	166
9.6.1	交换	167
9.6.2	段交换	168
9.6.3	分页	168
9.6.4	段页式	175
9.6.5	内存映射文件	175
9.6.6	写时复制	176
9.6.7	性能问题	176
9.7	嵌入式系统的存储管理	178
9.8	本章小结	178
9.9	练习	179
第 10 章	存储管理示例	181
10.1	CTSS	181
10.2	Multics	181
10.2.1	存储相关的系统调用	182
10.2.2	存储布局	182
10.2.3	段式管理与页式管理	182
10.3	RT-11	183
10.3.1	存储相关的系统调用	183
10.3.2	存储布局	183
10.3.3	USR 与 KMON 交换	184