



高等学校本科应用型教材

操作系统原理

——Linux 技术实现

史志才 毛玉萃



高等教育出版社

图书在版编目 (CIP) 数据

操作系统原理: Linux 技术实现/史志才, 毛玉萃.
北京: 高等教育出版社, 2004.11
ISBN 7-04-015120-0

I. 操... II. ①史...②毛... III. Linux 操作系统
IV. TP316.89

中国版本图书馆 CIP 数据核字(2004)第 105750 号

策划编辑 倪文慧 责任编辑 武林晓 市场策划 陈 振
封面设计 刘晓翔 责任印制 宋克学

出版发行 高等教育出版社
社 址 北京市西城区德外大街 4 号
邮政编码 100011
总 机 010-58581000

购书热线 010-64054588
免费咨询 800-810-0598
网 址 <http://www.hep.edu.cn>
<http://www.hep.com.cn>

经 销 新华书店北京发行所
印 刷 北京凌奇印刷有限责任公司

开 本 787×960 1/16
印 张 21.25
字 数 380 000

版 次 2004 年 11 月第 1 版
印 次 2004 年 11 月第 1 次印刷
定 价 25.20 元

本书如有缺页、倒页、脱页等质量问题, 请到所购图书销售部门联系调换。

版权所有 侵权必究

物料号: 15120-00

内 容 提 要

本书以现代操作系统的各组成功能模块为主线,详尽地介绍了现代操作系统的概念、发展历史、组成结构以及进程管理、处理机管理、存储管理、设备管理和文件系统等各功能模块的实现原理,并以源代码开放的自由软件 Linux 操作系统为例介绍了现代操作系统主要功能的实现技术,最后对并行、网络、分布式以及多媒体操作系统的特点、主要功能以及实现过程中的技术难点进行了详细的介绍。

该书覆盖面广,内容丰富,理论论述和实现技术互相衬托,文字精练,可读性好,实用性强,力争反映操作系统的最新成就。本书配套的 PPT 课件将上载到高等教育出版社的高等理工教学资源网(<http://www.hep-st.com.cn>)上,供读者学习参考。本书可以作为普通高等院校计算机科学与技术专业本科的教学用书或参考书,也可作为广大计算机科学技术工作者从事相关专业研究的参考书。

前 言

操作系统是现代计算机系统中最基本、最重要的系统软件,它是最接近硬件的第一层软件,是硬件功能的首次扩充;计算机系统中的任何其他软件均运行在操作系统所构筑的软件平台上。操作系统的主要功能是管理计算机系统的各种软、硬件资源,使它们能够高效、协调地运行,以提高系统资源的利用率,同时为用户使用计算机系统提供一种方便、简单、直观的手段(即用户接口)。随着计算机体系结构以及用户需求的不断变化,对操作系统的要求越来越高,其功能越来越强,实现难度也越来越大,所以操作系统一直是计算机科学与技术领域的重点研究方向之一。

操作系统在计算机系统的高效运行和管理过程中具有相当重要的作用,同时,用户或程序开发人员要对操作系统具有一定的认识和了解后才能充分利用计算机系统的各种软、硬件资源,更好地实现所要完成的功能,所以学习和掌握现代计算机操作系统的基本原理和实现技术是十分必要的。

为了方便读者对操作系统的原理和实现技术有一个全面、系统的认识和理解,本书以操作系统的主要功能为主线,以其理论和实现技术相融合的方式进行组织;理论的论述力求简明、扼要、精练;实现技术以 Linux 操作系统为例,对理论上难以理解的概念进行解释,力求形象、直观;本书结合作者多年来在操作系统方面的研究以及教学过程中对相关理论、技术问题的理解和认识,较全面地论述了操作系统的原理及其实现技术。同时,利用 Linux 操作系统源码开放等特点,广大科技人员和学生可以非常方便地上机认证,以获得更为直观的感性认识。本书除了介绍成熟的理论和技术外,还对操作系统领域中的前瞻性问题和热点问题(例如多媒体、对称多处理、分布式系统、集群、安全操作系统等)进行了阐述,论述了这些操作系统的特殊之处和技术难点,希望起到抛砖引玉的作用。每章均附有类型丰富的大量习题,可以进一步提高读者对相关概念、理论和技术的理解。

本书内容的安排与组织情况如下:第1章为操作系统概述,主要介绍操作系统的基本概念、分类、基本特征、基本功能、组成、发展以及研究操作系统的观点,并对几种典型的操作系统进行了简单介绍。第2章为用户接口,除了作业的相关概念之外,还介绍了操作系统提供给用户的三种接口形式。第3章为进程管理,主要介绍进程的概念、特性以及与程序、作业的区别,还介绍了进程的描述、进程的状态和转换以及线程的概念。第4章为处理机管理,主要介绍计算机系

统的分级调度及其算法,并对各种调度算法进行了详细论述。第5章为存储管理,对内部存储器的管理问题进行了讨论,主要包括各种管理方法如何实现内存的分配与回收、地址转换、共享与保护以及容量扩充等问题,并对内存的各种管理技术进行了比较。第6章为设备管理,主要介绍了设备管理的功能、作为主要外部存储设备——磁盘的驱动调度、与设备管理密切相关的技术(中断、通道和缓冲)以及设备的分配、虚拟设备等问题。第7章为文件管理,主要介绍文件管理中的基本概念、文件的逻辑组织和物理组织、磁盘存储空间的管理、磁盘的容错技术以及文件的共享、保护和保密等问题。第8章为进程的互斥、同步、通信与死锁,主要介绍进程的互斥、进程间的同步与通信以及死锁,并对有关进程通信的经典问题进行了讨论。第9章为安全和安全操作系统,一方面介绍了与操作系统有关的主要安全问题,包括影响安全的最重要方面、对系统的攻击来源、保证安全的基本措施、保护机制等,另一方面介绍了安全操作系统的概念、发展、模型、开发方法以及安全评测。第10章为并行、网络 and 分布式操作系统,主要介绍多处理器操作系统、分布式操作系统和网络操作系统的主要功能、特点和技术难点。第11章为多媒体操作系统,主要从多媒体操作系统与一般操作系统的差异出发,对多媒体操作系统进行了介绍,包括多媒体文件、文件系统、处理调度、文件在磁盘上的放置方式以及磁盘调度和缓存在多媒体中的特殊用途等。每章均结合 Linux 介绍了操作系统中各种关键技术和方法的具体实现方式。

本书第1章、第3章、第4章、第8章、第10章由史志才编写,第2章、第5章、第6章、第7章、第9章、第11章由毛玉萃编写,最后由史志才进行了全书的统稿工作。

本书覆盖面广、内容丰富、技术性强、可读性好,可以作为普通高等院校计算机科学与技术专业本科的教学用书,也可作为广大计算机科学技术工作者从事相关专业研究的参考书。

南京大学费翔林教授审阅了本书,并提出了许多宝贵意见,在此表示衷心的感谢;同时也非常感谢哈尔滨工业大学计算机学院博士后流动站的胡铭曾教授、廖宏安教授和季振洲教授对作者在站期间的关怀以及对于本书的编写所给予的支持和指导。

由于作者水平有限,书中难免存在不妥之处,敬请广大读者批评指正。

编 者

2004年8月于大连

目 录

第 1 章 操作系统概述	(1)	第 3 章 进程管理	(39)
1.1 操作系统的基本概念	(1)	3.1 进程的基本概念	(39)
1.2 操作系统的基本功能	(2)	3.1.1 程序的顺序与并发 执行	(39)
1.3 操作系统的类型	(6)	3.1.2 进程的定义及特征	(42)
1.4 操作系统的基本特征	(10)	3.2 进程的状态及其转换	(43)
1.5 操作系统的结构	(11)	3.3 进程描述	(45)
1.6 研究操作系统的几种观点 ..	(13)	3.3.1 进程的组成	(45)
1.7 操作系统的发展历程	(15)	3.3.2 进程控制块	(46)
1.8 典型操作系统简介	(17)	3.4 进程控制	(49)
1.9 本章小结	(21)	3.4.1 进程空间	(49)
习题	(22)	3.4.2 进程控制原语	(50)
第 2 章 用户接口	(23)	3.4.3 进程上下文	(51)
2.1 作业	(23)	3.5 线程	(52)
2.1.1 作业的概念	(23)	3.5.1 线程的基本概念及 分类	(52)
2.1.2 作业控制块	(24)	3.5.2 线程的状态及转换	(53)
2.1.3 作业的状态及其转换 ..	(25)	3.5.3 线程的应用	(54)
2.2 命令接口	(26)	3.6 Linux 的进程模型	(55)
2.2.1 联机用户接口	(26)	3.6.1 Linux 的进程控制块	(55)
2.2.2 脱机用户接口	(28)	3.6.2 Linux 进程的创建和 撤销	(57)
2.3 编程接口	(28)	3.6.3 Linux 进程的状态及其 转换	(58)
2.3.1 系统调用的类型	(29)	3.7 Linux 系统的线程机制	(59)
2.3.2 系统调用的实现	(29)	3.8 本章小结	(60)
2.4 图形接口	(30)	习题	(61)
2.4.1 窗口	(31)	第 4 章 处理机管理	(62)
2.4.2 图标	(32)	4.1 分级调度	(62)
2.4.3 菜单	(33)	4.1.1 作业调度	(62)
2.4.4 对话框	(33)	4.1.2 交换调度	(63)
2.5 Linux 的用户接口	(34)	4.1.3 进程调度	(63)
2.5.1 Linux 命令接口	(34)	4.1.4 线程调度	(65)
2.5.2 Linux 编程接口	(35)		
2.5.3 Linux 图形接口	(35)		
2.6 本章小结	(37)		
习题	(38)		

4.2 调度算法	(65)	5.5.1 段式存储管理	(106)
4.2.1 先来先服务调度算法 ..	(65)	5.5.2 段页式存储管理	(110)
4.2.2 优先级调度算法	(66)	5.6 Linux 的存储管理	(111)
4.2.3 轮转调度算法	(67)	5.6.1 物理内存的管理	(111)
4.2.4 分级轮转调度算法	(68)	5.6.2 进程空间的管理	(114)
4.2.5 分级反馈轮转调度 算法	(69)	5.6.3 Linux 虚存的保护	(116)
4.2.6 最短作业优先算法	(70)	5.7 本章小结	(117)
4.2.7 响应比高者优先算法 ..	(71)	习题	(118)
4.3 实时调度算法	(72)	第6章 设备管理	(120)
4.4 选择调度方式和评价调 度算法的若干准则	(73)	6.1 设备管理概述	(120)
4.5 Linux 的进程调度	(75)	6.1.1 设备的种类	(120)
4.5.1 调度的时机	(75)	6.1.2 设备管理的功能和 任务	(121)
4.5.2 调度算法	(76)	6.1.3 数据传送控制方式	(122)
4.6 本章小结	(78)	6.2 磁盘的驱动调度	(123)
习题	(78)	6.2.1 磁盘的结构	(123)
第5章 存储管理	(80)	6.2.2 磁盘的驱动调度	(125)
5.1 存储管理的功能	(80)	6.3 中断技术	(128)
5.1.1 内存的分配与回收	(80)	6.3.1 中断及其基本概念	(129)
5.1.2 地址转换	(81)	6.3.2 中断处理过程	(129)
5.1.3 内存信息的共享与 保护	(82)	6.3.3 中断优先级与多重 中断	(130)
5.1.4 内存的扩充	(83)	6.4 通道技术	(131)
5.2 覆盖和交换技术	(83)	6.4.1 通道的引入	(131)
5.2.1 覆盖技术	(84)	6.4.2 通道类型	(133)
5.2.2 交换技术	(85)	6.4.3 通道指令和通道 程序	(135)
5.3 分区存储管理	(85)	6.4.4 通道的工作过程	(137)
5.3.1 单分区存储管理	(86)	6.5 缓冲技术	(138)
5.3.2 多分区存储管理	(87)	6.5.1 缓冲的引入	(139)
5.3.3 分区存储管理的评价 ..	(94)	6.5.2 缓冲的种类	(139)
5.4 页式存储管理	(95)	6.5.3 缓冲池的管理	(140)
5.4.1 页式存储管理的基本原理 和使用的数据结构	(95)	6.6 设备分配	(142)
5.4.2 静态页式存储管理	(97)	6.6.1 设备的独立性	(142)
5.4.3 动态页式存储管理	(99)	6.6.2 设备分配的原则	(143)
5.4.4 页式管理的优缺点	(105)	6.6.3 设备分配策略	(143)
5.5 段式和段页式管理	(106)	6.6.4 设备分配所使用的数据 结构和分配算法	(144)

6.7 虚拟设备	(147)	7.4.3 树形目录结构	(179)
6.7.1 虚拟设备的引入	(147)	7.4.4 文件目录管理	(180)
6.7.2 虚拟设备的实现	(147)	7.5 磁盘存储空间的管理	(181)
6.8 I/O 进程控制	(151)	7.5.1 位示图	(181)
6.8.1 I/O 控制	(151)	7.5.2 空闲块表	(182)
6.8.2 I/O 控制的功能	(151)	7.5.3 空闲块链	(182)
6.8.3 I/O 控制的实现	(152)	7.6 磁盘容错技术	(184)
6.9 设备驱动程序	(153)	7.6.1 第一级容错技术	(184)
6.9.1 设备驱动程序的功能 和特点	(153)	7.6.2 第二级容错技术	(185)
6.9.2 设备驱动程序的处理 过程	(154)	7.6.3 廉价磁盘冗余阵列	(186)
6.9.3 设备驱动程序的管理	(155)	7.6.4 后备系统	(188)
6.9.4 Linux 的设备驱动 程序	(156)	7.7 文件的使用	(190)
6.10 Linux 的设备管理	(157)	7.7.1 文件的操作	(190)
6.10.1 设备文件的概念	(157)	7.7.2 文件的使用	(192)
6.10.2 相关数据结构	(158)	7.8 文件的共享、保护和保密	(193)
6.10.3 中断和异常	(160)	7.8.1 文件的共享	(193)
6.11 本章小结	(160)	7.8.2 文件的保护	(195)
习题	(161)	7.8.3 文件的保密	(197)
第7章 文件管理	(163)	7.9 文件的层次模型	(197)
7.1 文件和文件系统	(163)	7.10 Linux 的文件管理	(199)
7.1.1 文件	(163)	7.10.1 虚拟文件系统	(199)
7.1.2 文件的分类	(164)	7.10.2 EXT2 文件系统	(204)
7.1.3 文件系统	(165)	7.11 本章小结	(206)
7.2 文件的逻辑组织	(165)	习题	(207)
7.2.1 流式文件	(166)	第8章 进程的互斥、同步、通信 与死锁	(209)
7.2.2 记录式文件	(166)	8.1 进程互斥	(209)
7.2.3 存取方法	(168)	8.1.1 临界区与进程互斥	(210)
7.3 文件的物理组织	(169)	8.1.2 互斥的加锁实现	(212)
7.3.1 磁带文件的组织	(169)	8.1.3 信号量和 P、V 原语	(213)
7.3.2 磁盘文件的组织	(171)	8.1.4 利用 P、V 原语实现进程 互斥	(215)
7.3.3 记录的成组与分解	(175)	8.2 进程同步	(216)
7.4 文件目录	(177)	8.3 经典的进程同步问题	(219)
7.4.1 一级目录结构	(177)	8.3.1 生产者和消费者 问题	(219)
7.4.2 二级目录结构	(178)	8.3.2 其他经典的进程同步 问题	(221)

8.4 进程通信	(222)	及重要性	(254)
8.4.1 进程通信的类型	(222)	9.6.2 安全操作系统的	
8.4.2 消息传递通信	(224)	发展	(255)
8.5 死锁	(226)	9.7 安全操作系统的一般模型	
8.5.1 死锁的基本概念	(226)	及开发方法	(256)
8.5.2 进程死锁的解决		9.7.1 安全操作系统的一般	
方案	(228)	模型	(256)
8.5.3 典型的进程死锁解决方		9.7.2 安全操作系统的开发	
案——银行家算法	(229)	方法	(258)
8.6 Linux 中的线程同步	(232)	9.7.3 安全操作系统的开发	
8.7 Linux 中的进程通信机制	(233)	过程	(258)
8.7.1 管道	(233)	9.8 操作系统的安全评测	(260)
8.7.2 System V 的 IPC 通信		9.8.1 操作系统安全评测的	
机制	(234)	基础	(260)
8.8 本章小结	(238)	9.8.2 操作系统安全评测	
习题	(239)	方法	(261)
第 9 章 安全和安全操作系统	(242)	9.8.3 国内外计算机系统安全	
9.1 计算机安全最重要的三个		评测准则概况	(262)
方面	(242)	9.9 国内外安全操作系统	
9.1.1 威胁	(242)	研究的发展	(263)
9.1.2 入侵者	(243)	9.10 本章小结	(264)
9.1.3 意外数据丢失	(243)	习题	(265)
9.2 保证安全的基本措施	(243)	第 10 章 并行、网络 and 分布式操作	
9.2.1 密码术	(243)	系统	(266)
9.2.2 用户认证	(245)	10.1 计算机系统的分类	(266)
9.3 对系统的攻击	(246)	10.2 并行计算机操作系统	(267)
9.3.1 来自系统内的攻击	(246)	10.2.1 并行计算机系统	(267)
9.3.2 来自系统外的攻击	(247)	10.2.2 多处理机操作系统	(268)
9.4 保护机制	(248)	10.3 计算机网络操作系统	(272)
9.4.1 保护域	(248)	10.3.1 计算机网络简介	(272)
9.4.2 访问控制列表	(250)	10.3.2 计算机网络体系结构	
9.4.3 权能列表	(251)	与协议	(273)
9.5 Linux 的安全问题	(252)	10.3.3 网络操作系统的发展	
9.5.1 Linux 的安全措施	(252)	及分类	(275)
9.5.2 Linux 的安全漏洞	(254)	10.3.4 网络操作系统	
9.6 安全操作系统的概念及		的功能	(276)
发展	(254)	10.3.5 网络操作系统提供	
9.6.1 安全操作系统的概念		的服务	(279)

10.4 分布式操作系统	(280)	11.3.2 调度相同参数的 进程	(303)
10.4.1 分布式操作系统的特点	(281)	11.4 多媒体文件系统	(303)
10.4.2 分布式操作系统的构成	(282)	11.4.1 VCR 控制功能	(304)
10.4.3 分布式系统的通信	(282)	11.4.2 准点播	(306)
10.4.4 分布式操作系统的 资源管理	(285)	11.4.3 带有 VCR 功能的准 点播	(307)
10.4.5 分布式进程管理	(286)	11.5 文件在磁盘上的放置	(309)
10.4.6 分布式进程的同步、互斥 与死锁	(287)	11.5.1 单磁盘上文件放置	(309)
10.4.7 分布式文件系统	(289)	11.5.2 文件的放置策略	(310)
10.4.8 集群系统	(289)	11.5.3 准点播的文件放置	(313)
10.5 本章小结	(291)	11.5.4 多磁盘上文件的 放置	(314)
习题	(291)	11.6 缓存	(316)
第 11 章 多媒体操作系统	(293)	11.6.1 块缓存	(316)
11.1 多媒体的引入	(293)	11.6.2 文件缓存	(318)
11.2 多媒体文件及视频压缩	(295)	11.7 多媒体磁盘调度	(318)
11.2.1 多媒体文件	(295)	11.7.1 静态磁盘调度	(318)
11.2.2 视频压缩	(296)	11.7.2 动态磁盘调度	(320)
11.3 多媒体处理调度	(301)	11.8 本章小结	(321)
11.3.1 多媒体处理调度常用 算法简介	(301)	习题	(322)
		参考文献	(324)

第1章 操作系统概述

1.1 操作系统的基本概念

众所周知,计算机系统由硬件(hardware)和软件(software)两部分组成。硬件包括 CPU、存储器、输入/输出设备等,是用户直接可见的部分;而软件是存储、运行在存储器、CPU 中的程序,是用户直接观察不到的部分,具有一定的抽象性,难以理解。软件一般分为系统软件、支撑软件和应用软件。应用软件是为了完成某种特定应用功能的专用程序,如 Office 办公软件,它运行在系统软件或系统软件和支撑软件所构筑的软件平台之上。支撑软件运行于系统软件之上,为应用软件提供开发环境和手段,以方便应用软件系统的开发,如各种集成软件开发环境和中间件(middleware)等。系统软件是用于对计算机的软、硬件资源进行管理并为应用程序提供服务的程序集合,如编译程序、解释程序等。操作系统(operating system)是最基本的系统软件,是最接近硬件的第一层软件,它负责管理计算机系统的各种软、硬件资源,并为其他软件的运行提供支撑。

计算机系统中的硬件常被称为裸机(bare machine),它通常由电子、磁、机械、光学等部件所组成。按照 von Neumann 结构,计算机系统可划分为五个组成部分:中央处理单元(Center Processing Unit, CPU)、控制器、存储器、输入设备和输出设备。如果没有操作系统等系统软件的支撑,程序员只能采用机器语言来编写程序,使用二进制代码编程非常烦琐,不直观,可读性和移植性差,而且还需要程序员考虑程序运行过程中计算机各个组成部分如何工作的具体细节,这对程序员的要求很高,负担很重,难以充分发挥计算机硬件的效率。针对这种情况,人们在计算机硬件的基础上增加一层软件来自动管理计算机系统的软、硬件资源和控制程序的运行,将硬件的复杂性同程序员分离开来,并为用户提供使用计算机系统资源的简便手段,使用户能够高效、方便地使用计算机;通过这层软件,用户仅需将要计算机所做的工作用直观、简单、形象的语言(如高级语言)编好程序并提交给计算机,而程序的运行以及在其运行过程中所涉及的系统资源的分配和使用等完全由该层软件来负责,这样,用户使用计算机就显得相当方便,而且轻松自如。人们把位于裸机上面的这层系统软件称之为操作系统,由计算机硬件和操作系统所组成的计算机系统称为“虚拟机”(virtual machine),它具

有比裸机更强的功能和更好的易用性,计算机系统的组成如图 1.1 所示。

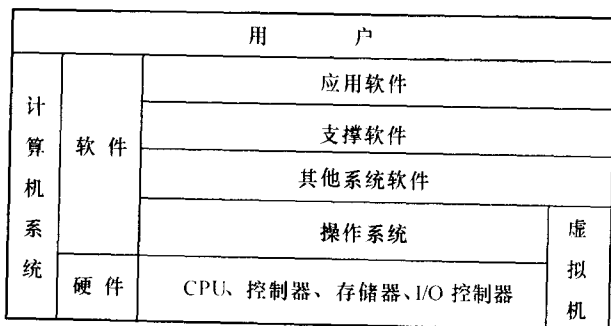


图 1.1 计算机系统的组成

由上面的分析可知,操作系统是一种系统软件,是由若干程序所组成的集合,它负责计算机系统的全部软、硬件资源的分配、调度和管理,使系统高效、安全地运行,并为用户提供简单、直观、灵活的接口,以方便用户对计算机系统的使用。

1.2 操作系统的基本功能

如上所述,操作系统负责分配、调度和管理计算机系统的全部软、硬件资源,并为用户使用计算机系统提供手段。操作系统管理的对象中,硬件资源有 CPU、内存、寄存器、堆栈、辅助存储器和输入/输出设备等,而软件资源有系统软件、应用软件和数据等。总而言之,操作系统的功能包括如下五部分:处理机管理、存储器管理、设备管理、文件管理和用户接口。

1. 处理机管理

处理机管理就是对 CPU 进行管理,即:如何分配处理机?当系统中存在多个程序要运行时分配给谁?分配多长时间?何时收回?等等。众所周知,CPU 是计算机系统的核心,是最宝贵的硬件资源。如何调度程序以使 CPU 尽可能地忙起来,减少其空闲时间,提高其利用率,相对提高系统的处理能力,是操作系统所要重点解决的问题。例如,当某个用户程序 A 进行输入/输出操作时,CPU 处于空闲状态,是否可以将 CPU 暂时分配给用户程序 B,当用户程序 A 完成输入/输出操作时再中断用户程序 B 并返回到用户程序 A 的断点处继续执行,从而减少 CPU 的空闲时间,提高 CPU 的利用率。

为了方便处理机的调度、分配和管理,引入了进程和线程的概念。进程是对处于运行状态下程序的动态描述,而线程是进程内部的一个控制流,这些概念的

具体含义在以后的相关章节中再详细解释。有了上述概念后,就可以引出多任务、多进程和多线程的概念,即在一定时间内(宏观上)计算机同时执行多个任务、多个进程和多个线程。对于单 CPU 系统,由于仅有一个 CPU,某一时刻(即微观上)不可能有多个任务或多个进程同时占有 CPU,所以只能采用某种调度策略将 CPU 轮流分配给各任务或进程,使它们在一段时间间隔内均能得到执行的机会,从而实现宏观上的同时运行(人们称之为并发执行)。

处理机的管理最终可以归结为对进程的管理,因为处理机的调度、分配都是以进程为基本单位的;故这一部分还包括进程管理、同步、互斥、通信和死锁等内容。

2. 存储器管理

存储器管理的对象为主存储器,简称为内存或主存。因受成本等方面条件的限制,主存储器的容量有限。众所周知,程序若要运行需首先装入内存,然后待分配到 CPU 等必要的系统资源后便可以执行了,那么用户程序如何申请到内存?申请到内存后又怎样从外存(磁盘等)装入内存?具体装入到内存的哪个位置?是否会与内存中已有的程序发生冲突?如果这些问题均需用户来考虑,用户编程的难度简直不可想像;如果上述功能可以通过软件自动实现,则可以减轻用户很大负担。出于上述考虑,操作系统中设计了存储器管理模块,用来实现存储器的自动管理和高效使用。

存储器管理是对内存资源进行管理,功能之一是为位于外存中的程序分配一定的内存空间,并将之装入内存,此时存在三种情况:其一,若内存中有足够的空闲空间,程序便可以顺利装入内存。其二,当要装入内存中的程序大于空闲内存空间的大小时,还需根据一定的算法将内存中的某些程序交换到外存,以空出内存空间。其三,当大于内存总容量的某个大型程序需要运行或多个用户程序要同时并发执行时,即使内存全部处于空闲状态仍满足不了需要;在这种情况下就需要提供某种机制,仅将要运行的那部分程序装入内存,而其他部分暂时存在外存,在运行时再装入内存,从而实现小内存运行大程序的目的;这样不需扩大物理内存的容量,而是借助于软件等技术把内存和外存结合起来,从逻辑上扩充内存的相对容量,并通过存储器管理功能为用户提供透明服务(即用户无需知道如何分配内存,程序具体放在内存的哪个位置等),使用户感觉到存在一个比实际内存空间大得多的存储空间供其使用,这就是虚拟存储技术。

由上述内容可以看出,存储器管理的主要任务是内存的分配和回收、内存扩充、存储共享与保护、地址变换等,并以提高内存的利用率、扩大内存的相对容量、为用户提供透明服务为目的。实现上述任务的具体内容如下:

(1) 内存的分配和回收:即为要装入内存的程序分配内存空间,在程序运行结束后要释放所占用的内存资源,并由系统进行登记,以便分配给将要运行

的程序。

(2) 内存扩充:为了解决分配给作业或进程的内存空间不足问题,达到小内存运行大程序的目的,同时也为了提高进程的并行性,使系统资源得以充分利用,均需要对内存的容量进行相对扩充。操作系统主要采用软件技术(如覆盖技术、交换技术和虚拟技术等)对存储系统进行管理,利用大容量的辅存来弥补内存空间的不足(详见第5章存储管理)。

(3) 存储共享与保护:这里有两方面含义:一是指存储共享,即当多个程序包括一个公用程序段时,这个公用程序段在内存中仅存一个副本,为一共享程序段;存储管理要解决这一公用程序段的共享问题;二是指存储保护,即内存中存在多个运行程序时,存储管理要保证每个用户程序只能访问自己的存储空间,不能对驻留在内存中的操作系统等系统软件或其他应用程序造成破坏。存储保护通常需要硬件的支持,常用的方法有界限地址寄存器、存储保护键等。

(4) 地址变换:由于一般情况下内存空间较外存空间小得多,所以两者编址是不一样的。当程序从外存装入内存时,需将程序中每条指令的地址(相对地址或逻辑地址)转换为内存中的地址(物理地址)。该过程可以在程序装入内存时完成(静态地址变换),也可以在程序执行时进行(动态地址变换),这个过程分别称为静态地址重定位和动态地址重定位,通常需要硬件的支持。

3. 设备管理

设备管理是指对计算机中除了 CPU 和内存以外的输入/输出设备的管理。输入/输出设备也称为外部设备,其种类繁多,功能差异很大,使用方法各异。设备管理的主要任务就是通过驱动程序和控制程序自动实现对输入/输出设备的管理和调度以及数据传输,以使用户不必了解输入/输出设备的具体细节就可以完成输入/输出操作,实现输入/输出操作与设备的无关性。设备管理的另一任务就是通过中断技术、DMA(Direct Memory Access)技术、通道技术和缓冲技术等使 CPU 与外部设备能够并行、高效地工作,解决高速 CPU 与低速外设间速度不匹配的矛盾,从而提高系统资源的利用率。

由上面分析可见,设备管理应包括如下功能:响应用户进程提出的 I/O 请求;为用户进程分配 I/O 设备;对系统的 I/O 设备进行管理;控制 CPU 与外设间的数据交换。

4. 文件管理

计算机系统所处理的对象为程序和数据,这些程序和数据通常以文件的形式存放在磁盘、光盘和磁带等外部存储器上,需要时再装入内存。外存的存储容量相当大,存有大量的系统软件、应用程序和数据等,这些文件在外存中如何组织才能方便用户对文件的查找和透明存取均是文件管理模块所要解决的问题。

实现对文件的“按名存取”是操作系统所要完成的另一项重要功能,即用户只要知道文件名就可以方便地存取文件,具体如何组织和查找文件均由文件管理模块来完成,用户无需关心。此外,文件管理还应提供文件的共享和保护、文件操作等功能。其中,文件的共享对于多道程序系统、多用户系统或多任务系统是相当重要的,它使被多个程序共享的对象在外存上仅保留一个备份,从而提高了外部存储器的利用率;文件保护功能用于对外部存储器中的文件进行保护,防止文件受到其他用户非法读取或破坏;文件操作为用户对文件进行操作提供手段,如文件的读、写、复制、删除等。此外,文件系统还应提供磁盘的容错功能,以保证所存储文件的可靠性。

5. 用户接口

操作系统作为计算机系统资源的直接管理者,应提供相应的手段供用户使用,使用户可以方便地使用系统的各种资源,这种手段就是用户接口。用户接口通常分为两种,一种供用户通过键盘和鼠标直接对计算机进行操作时使用,称为操作级接口;另一种供用户在编程过程中使用,称为编程级接口。这两种接口体现为三种形式:命令接口、图形接口和编程接口。

命令接口和图形接口为用户提供了直接控制计算机或使用系统资源的手段。命令接口为用户提供若干条命令,用户通过键盘和显示器完成想要的操作。用户所要完成的操作可以由若干条命令组成,这些命令可以通过键盘逐条键入系统,此称为联机用户接口。这些命令也可以通过编辑程序录入到计算机内,形成一个文件,系统通过运行这个文件来完成指定的功能,此称为脱机用户接口,如批处理等。图形接口(Graphics User Interface, GUI)为用户提供了功能图标或菜单并显示在显示器上,用户若想完成某项功能,就可以用鼠标点击相应的图标或者选择相应的菜单,如现在流行使用的视窗操作系统 Windows 家族系列、Linux 等均提供这样的方式。

编程接口是为用户程序在执行过程中访问系统资源而设置的,是用户程序访问系统资源的惟一手段。操作系统通过系统功能调用来实现它的编程接口。开发软件时,用户通过编辑程序将系统功能调用嵌入到程序中来访问系统的软、硬件资源,从而对用户屏蔽了功能实现的具体细节。实际上,命令控制界面也是通过系统功能调用来实现的。每个系统调用具有不同的功能号,由专门的指令嵌入到程序中,以完成特定的功能。对于不同的操作系统有不同的系统调用,一般有几十至几百条。系统调用可分为如下几类:设备管理、文件管理、进程管理、存储管理和线程管理。

操作系统设计的目标之一就是为用户提供直观、简单、高效、易用、透明的使用接口。

1.3 操作系统的类型

在计算机技术发展的不同时期产生了不同种类的操作系统,以满足该时期特定的硬件条件下用户对计算机的使用。同时针对不同的应用场合,也开发了相应的操作系统以满足用户的特殊需求。

1. 批处理系统(batch processing operating system)

早期的计算机系统非常昂贵,为了能充分利用,应尽量保持计算机系统处于连续运行状态。此时,用户不直接接触计算机,而是将由程序和数据组成的用户作业提交给系统操作员。当接收多个用户作业后,系统操作员将它们成批地装入计算机,然后由操作系统进行组织,并按照一定的算法选择一道作业装入系统内运行。作业选择算法的优劣一般以作业平均周转时间等作为评判标准。在多道程序设计技术出现以后,批处理操作系统由单道批处理发展成多道批处理。

早期的批处理操作系统是单道的,一个作业单独装入系统并独占系统的所有资源,直到其运行结束后下一个作业才能装入系统。此时 CPU 等系统资源的利用率较低,特别是对于 I/O 操作较多的作业。为了提高 CPU 的利用率,引入了多道程序设计(multiprogramming)技术,即内存中同时装入多个作业,它们通过一定的调度算法轮流占有 CPU,这样可以使 CPU 尽量处于运行状态,显著提高了 CPU 的利用率。

批处理操作系统一般运行在较大型的计算机系统上,它的特点是用户脱机使用计算机,多道程序同时运行、成批处理。目的是为了提高设备利用率和作业吞吐率。其缺点是作业周转时间长,无交互性,用户无法直接进行必要的干预。

2. 分时系统(time sharing operating system)

批处理系统环境下,用户以脱机方式工作,不直接和计算机接触,程序在运行过程中出现问题用户不能直接干预,这样使用起来非常不方便,因此产生了分时操作系统。

分时操作系统将规定的系统运行周期划分成若干个时间段,每个时间段称为时间片,然后按时间片将 CPU 轮流分配给系统内的每个程序。这样连接到计算机的每个用户或系统内的每个程序在系统运行周期内均能得到服务。当该周期选择合理或计算机的速度足够快时,各用户或程序并没有感觉到其他用户或程序的存在,好像自己在独享计算机系统的所有资源,而实际上是各个用户或程序轮流地使用计算机。显然,此时一台计算机可以有多个终端用户,各用户各自的时间片内能够和计算机进行交互作用,系统对用户的要求能够及时响应。

分时操作系统具有多路性、独立性、及时性、交互性等特征。多路性指宏观

上多个用户同时使用计算机,而微观上是各用户轮流使用;独立性是指各用户均感觉是独占计算机的所有资源,而感觉不到其他用户的存在;及时性是指各用户对计算机的操作均能得到及时响应;交互性是指用户和计算机能够进行人机交互,干预计算机的运行。

由此可见,分时系统是一个联机 (on-line)、多用户 (multi-user)、交互性 (interactive) 的操作系统。

目前,常用的通用操作系统为分时系统和批处理系统的结合。对响应时间要求较高的用户一般工作在分时状态下,称之为前台;而对时间性要求不高的作业一般工作在批处理状态下,称之为后台,如打印作业。

3. 实时系统 (real time operating system)

实时操作系统是指计算机能够及时响应外部事件的请求,在规定的时间内完成对事件的处理,并有效地控制被控对象和实时任务协调地运行。实时操作系统一般是为了满足某一类实时系统的应用而设计的,如工业生产线的控制、武器控制系统等。在设计上,首先要保证它的实时性和可靠性,其次才是系统效率。它常有两种类型:实时控制系统和实时信息处理系统,前者能够实时采集测量数据,并对数据进行实时加工、处理和输出,这种系统主要用于军事和工业生产过程的自动控制。后者能够对用户的请求及时做出响应,并能及时修改、处理系统中的数据,主要用于银行业务、订票系统等实时事务管理。

4. 通用操作系统

批处理操作系统、分时操作系统和实时操作系统是操作系统的三种基本类型,在此基础上发展了具有多种类型操作系统特征的通用操作系统,它可以兼有批处理、分时和实时操作系统的部分特征,具有一定的通用性,能够适合使用于较宽的应用领域。

5. 个人计算机操作系统 (personal computer operating system)

个人计算机操作系统一般运行于个人计算机环境下,是一种单用户、单任务或单用户、多任务的操作系统。它提供联机交互功能,而且采用图形用户界面,操作直观、友好,易学易用。这类操作系统具有虚存、并发、多任务、联网等特征或功能。这类操作系统有 DOS、Windows 3.x/95/98、OS/2、Linux 等。

6. 嵌入式操作系统 (embedded operating system)

嵌入式操作系统是运行在嵌入式系统环境中,对整个嵌入式系统以及它所操作、控制的对象进行有效管理的系统软件。嵌入式系统是以应用为中心,软、硬件可裁减,能够满足应用系统对功能、可靠性、成本、体积、功耗等综合性要求的专用计算机系统。它主要由嵌入式处理器、相关支撑硬件、嵌入式操作系统及应用软件系统等组成,它是集软、硬件于一体的可独立工作的“器件”。其中,嵌