



高职高专 **立体化教材** 计算机系列

操作系统原理与应用

CAOZUOXITONG YUANLI YU YINGYONG

成秋华 主 编
丛佩丽 李星华 曾 振 副主编

 赠送电子课件及
其他立体化资源



清华大学出版社

读者回执卡

高职高专立体化教材 计算机系列

操作系统原理与应用

成秋华 主编

丛佩丽 李星华 曾 振 副主编

清华大学出版社
北 京

邮 处

邮 票

邮 票
1-1000
2008年6月1日
185×580
全图版并印

北京市平谷区
三河市金山公司
全图版并印

北京100084-1157信箱

内 容 简 介

本书在编写上突出内容的实用性和系统性。全书分为3个部分。第1~5章为基础篇,介绍操作系统的基础知识和基本管理功能;第6~7章为系统篇,介绍操作系统的管理功能在Windows和Linux系统中的实现方法;第8~9章为实训篇,分别在Windows和Linux操作系统中设置了8~10个实验项目,既有基础的命令操作,又有网络服务的配置操作,把理论教学和实践教学紧密结合起来。本书的特点是,理论知识内容深浅适度,实验和实训项目安排合理、步骤详细,具有很强的操作性,能提高学生对操作系统的应用能力。

本书可作为应用型、技能型人才学习“操作系统”课程的教学用书,也可供各类培训机构用作培训教材,对广大自学爱好者来说也是一本通俗易懂的自学参考书。

本书封面贴有清华大学出版社防伪标签,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

操作系统原理与应用/成秋华主编;丛佩丽,李星华,曾振副主编. —北京:清华大学出版社,2008.6
(高职高专立体化教材 计算机系列)
ISBN 978-7-302-17612-1

I. 操… II. ①成… ②丛… ③李… ④曾… III. 操作系统—高等学校:技术学校—教材 IV. TP316

中国版本图书馆CIP数据核字(2008)第069195号

责任编辑:石伟 桑任松

封面设计:山鹰工作室

版式设计:杨玉兰

责任印制:王秀菊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦A座

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

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

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

印 刷 者:北京市清华园胶印厂

装 订 者:三河市金元印装有限公司

经 销:全国新华书店

开 本:185×260 印 张:21.5 字 数:515千字

版 次:2008年6月第1版 印 次:2008年6月第1次印刷

印 数:1~4000

定 价:34.00元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:029032-01

前 言

操作系统是计算机的核心系统软件,它负责控制和管理整个系统的资源并组织用户协调使用这些资源,使计算机高效地工作。操作系统课程是计算机科学与技术类专业的核心课程,是所有从事计算机工作的技术人员必须掌握的基本知识。

近几年,随着高职高专职业技术教育的发展,对高技能型人才培养提出了更高的要求。对于操作系统这类理论性和概念性非常强的课程,怎样把握理论内容的深度和广度,怎样提高学生对理论知识的应用、提高学生的操作能力,成为操作系统课程教学改革的重点。本教材内容的取舍以“理论适度,重在应用”为主导思想,文字描述力求“概念引出自然,内容深入浅出”,配以实例分析、习题和实验,重点提高学生分析问题的能力和实践动手能力。

本书作者长期从事计算机操作系统的研究和教学工作。通过多年的教学实践,认为对高职高专学生操作系统课程的教学,要培养学生的3种能力:

- 系统管理和系统优化能力。了解操作系统怎样管理计算机系统的资源,有哪些管理技术和手法,通过这部分知识的学习,使学生具备对计算机进行系统管理的能力,了解系统工作状态及对系统进行性能优化的能力。
- 借鉴操作系统技术实现方法的能力。通过操作系统中技术实现方法的学习,使学生在后续的程序设计课程中,能借鉴一些操作系统中的方法和思想,提高软件开发的能力。通过了解操作系统为用户提供的接口和内核功能,提高学生在操作系统平台上开发应用程序的能力。
- 网络服务配置能力。操作系统最直接的应用就是管理网络和提供网络服务。通过学习,把理论知识和具体的操作系统结合起来,能运用相关的命令和软件工具配置各种网络服务并进行相应的管理。

在教材的内容策划上着重考虑了以下几个问题:

- 对传统的五大管理模块,省略了作业管理。理由是在微机操作系统中,任务的提交基本上是经过一个由进程到线程的过程,作业管理层次在微机系统中已被淡化。
- 增加了操作系统基本工作机制的介绍。讲解了操作系统中的接口、中断、指令、虚拟和时钟机制,这些知识将有助于对操作系统技术实现的理解。
- 实践环节的教学内容以目前的主流操作系统 Windows 和 Linux 为两条主线,在教学中可以根据各学校的具体情况进行选择,增加了教材的适应性和灵活性。
- 全书以操作系统的概念、原理和方法、实例分析、实验与实训为点,将操作系统理论和实践串联起来,即保留了操作系统的经典内容,在内容的处理编排上又力求创新,而且通过实例分析深入介绍了重点内容。实验与实训内容则提高了学生的动手能力。

本书由成秋华主编、策划和统稿。各章编写分工为：第1、2、3、7章和第5章的5.6节由成秋华编写，第4章和第5章的5.1~5.5节由丛佩丽编写，第6、8章由李星华编写，第9章由曾振编写。

由于时间仓促和编者水平有限，书中难免有疏漏和欠妥之处，敬请各位专家、读者批评指正。

编者

目 录

第 1 章 概论	1
1.1 操作系统的形成与发展	1
1.1.1 无操作系统阶段	2
1.1.2 监控程序阶段	2
1.1.3 执行系统阶段	2
1.1.4 操作系统的形成与成熟阶段	3
1.2 操作系统的基本概念	3
1.2.1 操作系统的定义和特征	3
1.2.2 操作系统的功能服务	4
1.2.3 操作系统的结构与性能指标	5
1.3 操作系统的分类	7
1.3.1 多道批处理操作系统	7
1.3.2 分时操作系统	8
1.3.3 实时操作系统	8
1.3.4 网络操作系统	9
1.3.5 分布式操作系统	10
1.3.6 多处理机操作系统	11
1.3.7 嵌入式操作系统	11
1.4 操作系统的基本功能	12
1.4.1 处理机管理	12
1.4.2 存储器管理	13
1.4.3 设备管理	13
1.4.4 文件管理	13
1.5 操作系统的基本工作机制	14
1.5.1 接口机制	14
1.5.2 中断机制	15
1.5.3 指令机制	16
1.5.4 虚拟机制	17
1.5.5 时钟机制	17
1.6 现代主流操作系统简介	18
1.6.1 Windows 操作系统	18
1.6.2 UNIX 操作系统	19
1.6.3 Linux 操作系统	20
习题	22
第 2 章 进程管理	25
2.1 进程	25
2.1.1 进程的引入和定义	25
2.1.2 进程的状态和进程控制块	28
2.1.3 进程的调度	31
2.2 进程的同步与互斥	37
2.2.1 进程之间的制约关系	38
2.2.2 信号量机制	39
2.2.3 同步与互斥的实现	39
2.2.4 互斥与同步实例分析	43
2.3 进程的通信	46
2.3.1 消息缓冲通信	46
2.3.2 信箱通信	48
2.3.3 管道通信	49
2.4 死锁	50
2.4.1 死锁的基本概念	50
2.4.2 死锁的预防与避免	52
2.4.3 死锁的检测与解除	55
2.5 线程	58
2.5.1 线程的概念	58
2.5.2 线程实现机制	60
2.5.3 线程与进程的比较	62
2.6 系统调用	63
2.6.1 系统调用的基本概念	63
2.6.2 系统调用的实现过程	63
2.6.3 系统调用与一般过程调用、库函数调用的主要区别	64
2.6.4 系统调用的主要种类	65
2.7 实例分析	65
习题	72

第3章 存储管理	77	第5章 文件管理	144
3.1 存储管理的基本概念	77	5.1 文件管理的基本概念	144
3.1.1 存储管理的任务	77	5.1.1 文件与文件名	144
3.1.2 覆盖与交换技术	82	5.1.2 文件系统	145
3.2 实存管理	84	5.1.3 文件的分类	146
3.2.1 分区存储管理	84	5.2 文件的组织与文件目录	148
3.2.2 分页存储管理	88	5.2.1 文件的逻辑结构	148
3.2.3 分段与段页式存储管理	93	5.2.2 文件的物理结构	148
3.3 虚拟存储管理	96	5.2.3 文件目录	151
3.3.1 请求分页存储管理	96	5.3 外存空间的管理	155
3.3.2 调页与页面淘汰	97	5.3.1 磁盘调度算法	155
3.3.3 页面置换算法	98	5.3.2 磁盘存储空间的管理	160
3.4 实例分析	101	5.4 文件共享与安全	162
习题	105	5.4.1 文件共享的控制与实现	162
第4章 设备管理	109	5.4.2 文件存取控制	163
4.1 设备管理概述	109	5.5 文件操作的实现过程	166
4.1.1 设备管理的基本任务和 功能	109	5.5.1 文件的建立与删除	166
4.1.2 设备的类型	110	5.5.2 文件的打开与关闭	167
4.1.3 设备的分配	112	5.5.3 文件的读写操作	169
4.2 数据传输控制方式	118	5.6 实例分析	169
4.2.1 程序直接控制方式	118	习题	172
4.2.2 中断控制方式	119	第6章 Windows 2000 操作系统	175
4.2.3 DMA 方式 (Direct Memory Access)	120	6.1 Windows 2000 系统简介	175
4.2.4 通道方式	122	6.2 Windows 2000 的进程管理机制	176
4.3 设备管理技术	122	6.2.1 Windows 2000 线程调度	176
4.3.1 缓冲技术	122	6.2.2 Windows 2000 进程的 通信	179
4.3.2 中断技术	125	6.3 Windows 2000 的存储管理机制	180
4.3.3 通道技术	129	6.3.1 Windows 2000 的 地址变换	180
4.3.4 SPOOLing 技术	131	6.3.2 Windows 2000 的 页面调度策略	184
4.4 设备驱动程序	132	6.3.3 Windows 2000 的 高速缓存管理	184
4.4.1 设备驱动程序的功能	133	6.3.4 Windows 2000 的 磁盘管理	188
4.4.2 设备驱动程序的结构	133	6.4 Windows 2000 的文件管理机制	190
4.5 实例分析	134		
习题	141		

6.4.1	Windows 2000 文件系统的层次结构	190
6.4.2	Windows 2000 的 FAT 文件系统	191
6.4.3	Windows 2000 的 NTFS 文件系统	193
6.5	Windows 2000 的用户接口管理机制	196
6.5.1	Windows 2000 的用户界面	196
6.5.2	Win32 API	198
	习题	200
第 7 章	Red Hat Linux 操作系统	201
7.1	Linux 的进程管理机制	201
7.1.1	Linux 进程结构	202
7.1.2	Linux 进程调度策略	206
7.1.3	Linux 进程优先级的调整	207
7.1.4	进程管理的常用命令和系统调用	208
7.2	Linux 的内存管理机制	210
7.2.1	Linux 操作系统中的虚拟存储空间	210
7.2.2	Linux 的页面调度策略	211
7.2.3	Linux 的交换文件和交换分区	213
7.2.4	Linux 的内存操作命令	214
7.3	Linux 的文件和设备管理机制	215
7.3.1	Linux 的文件系统	215
7.3.2	Linux 的虚拟文件系统	217
7.3.3	Linux 文件系统注册和装卸	220
7.3.4	Linux 的设备文件和设备驱动程序	223
7.3.5	文件管理的常用命令	225
7.4	Linux 的用户管理机制	226
7.4.1	用户的管理	226
7.4.2	文件与目录的权限管理	229
7.4.3	常用的文件与目录命令	231
7.5	shell 与文本编辑器 vi	233

7.5.1	shell	233
7.5.2	vi 文本编辑器	235
	习题	237

第 8 章 Windows 2000 Server

	实验与实训	239
8.1	Windows 2000 Server 的安装	239
8.2	Windows 2000 Server 磁盘管理	248
8.2.1	分区的创建和管理	248
8.2.2	文件的压缩、加密与磁盘整理	252
8.3	Windows 2000 Server 文件系统管理	254
8.3.1	查看与更改文件、文件夹的访问许可权	254
8.3.2	添加与管理共享文件夹	256
8.4	DNS 服务器的安装和配置	257
8.5	WWW 服务器的配置	262
8.5.1	安装 IIS	263
8.5.2	配置、管理 Windows 2000 WWW 服务器	264
8.6	FTP 服务器配置	267
8.6.1	配置、管理 Windows 2000 FTP 服务器	267
8.6.2	测试 FTP 服务器	270
8.7	DHCP 服务器配置	270
8.7.1	添加 DHCP 服务器	271
8.7.2	DHCP 服务器配置与管理	271
8.8	搭建一个基于 Windows 的局域网	276
8.8.1	组建一个基于 Windows 的小型局域网	276
8.8.2	设置文件夹共享和打印及共享, 映射网络硬盘	278

第 9 章 Red Hat 9.0 实验与实训

9.1	Red Hat 9.0 系统安装	280
9.1.1	实验目的	280
9.1.2	实验环境	280
9.1.3	实验预备知识	280

9.1.4 实验内容与步骤.....	282	9.7.1 实验目的.....	308
9.2 Linux 常用命令和工具.....	288	9.7.2 实验环境.....	308
9.2.1 实验目的.....	288	9.7.3 实验预备知识.....	308
9.2.2 实验环境.....	288	9.7.4 实验内容与步骤.....	309
9.2.3 实验预备知识.....	288	9.8 DNS 服务器的配置.....	310
9.2.4 实验内容和步骤.....	289	9.8.1 实验目的.....	310
9.3 网络配置和管理.....	289	9.8.2 实验环境.....	311
9.3.1 实验目的.....	289	9.8.3 实验预备知识.....	311
9.3.2 实验环境.....	290	9.8.4 实验内容与步骤.....	311
9.3.3 实验预备知识.....	290	9.9 WWW 服务器的配置.....	318
9.3.4 实验内容与步骤.....	291	9.9.1 实验目的.....	318
9.3.5 操作练习.....	296	9.9.2 实验环境.....	318
9.4 进程和磁盘管理.....	296	9.9.3 实验预备知识.....	318
9.4.1 实验目的.....	296	9.9.4 实验内容与步骤.....	319
9.4.2 实验环境.....	296	9.9.5 httpd.conf 文件解析.....	324
9.4.3 实验预备知识.....	297	9.10 FTP 服务器的配置.....	325
9.4.4 实验内容与步骤.....	300	9.10.1 实验目的.....	325
9.5 用户管理.....	301	9.10.2 实验环境.....	325
9.5.1 实验目的.....	301	9.10.3 实验预备知识.....	325
9.5.2 实验环境.....	301	9.10.4 实验内容与步骤(在 Windows	
9.5.3 实验预备知识.....	301	客户机上进行).....	327
9.5.4 实验内容与步骤.....	303	9.10.5 vsftpd.conf 文件解析.....	329
9.6 Samba 服务器的配置.....	303	9.11 网络服务实验环境的准备和	
9.6.1 实验目的.....	303	IP 地址规划.....	330
9.6.2 实验环境.....	303	9.11.1 实验环境网络拓扑图及	
9.6.3 实验预备知识.....	304	IP 地址规划.....	330
9.6.4 实验内容与步骤.....	305	9.11.2 实验项目的设置.....	330
9.7 NFS 服务器的配置.....	308	9.11.3 说明.....	331

第1章 概 论

内容提示：本章以操作系统的发展过程为主线，阐述了操作系统在计算机系统中的地位 and 作用，多道程序设计的思想引入，并发、共享和虚拟的概念，操作系统的结构与性能指标，操作系统的基本功能和操作系统的基本工作机制，并介绍了现代主流操作系统。

教学目标：掌握操作系统的概念与作用，操作系统的功能与结构，了解操作系统的基本工作机制，理解并发、共享、虚拟等基本概念，为以后各章的学习与理解奠定基础。

1.1 操作系统的形成与发展

现代计算机系统由硬件系统和软件系统两部分构成。硬件系统是指构成计算机系统的物理设备，比如控制器、运算器、存储器、外部设备等。软件系统是指在计算机系统中使用的各种程序。没有任何软件支持的计算机称为“裸机”，让用户直接面对裸机工作是十分困难的，同时用户编程时也不想涉足硬件的具体细节。那么，隐蔽对硬件的复杂操作，建立起一个服务体系，为用户提供良好的操作环境和服务功能就成为计算机系统的主要任务之一。在计算机的硬件之上覆盖一层层的管理软件，内层向外层提供某种服务，每经过一层覆盖，系统功能便会增强一个级别。到了用户层，“裸机”就扩展成了一台操作界面简单、功能强大的计算机。在裸机上覆盖的第一层系统软件就是操作系统。操作系统是最基本的系统软件，也是对硬件系统功能的第一次扩充。操作系统直接控制和管理所有的系统硬件，也为其他系统软件和应用软件提供基本的支持环境。当代的计算机都离不开操作系统。

从图 1-1 所示的计算机系统的层次结构来看，从最底层的裸机开始直到计算机用户，操作系统是最靠近硬件的一个层次，它控制和管理着在它内层的硬件系统，也控制和管理

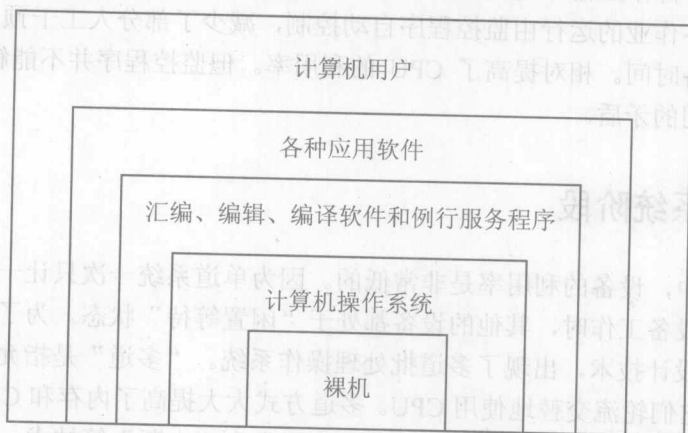


图 1-1 计算机系统的层次结构

着在它外层的系统实用软件和各种应用软件, 为其他软件提供开发与运行环境, 并与各种系统实用软件协作, 使各种应用软件能正常、高效率地运行。

1.1.1 无操作系统阶段

20 世纪 50 年代末以前, 也就是第一台计算机问世后的若干年, 并没有操作系统。那时的计算机运行速度慢, 外部设备少。程序的装入、调试及控制程序的运行等工作都由操作人员手工完成。人们以手工操作的方式来使用计算机。计算机工作时, 由操作员把程序纸带(或卡片)装入输入机, 输入机把程序和数据输入计算机后, 操作员启动程序运行, 运行完后取走结果, 卸下纸带(或卡片)。然后下一个用户才能上机。

这种人工操作方式的缺点如下。

- (1) 用户独占计算机的全部系统资源。
- (2) CPU 的利用率低, 大量的时间出现空闲等待。
- (3) 作业运行过程需人工干预。

人工操作方式下, 系统资源的利用率低, 系统中没有任何管理软件, 用户直接承担所有的运行管理和具体操作。

1.1.2 监控程序阶段

随着计算机运行速度的提高, 快速的 CPU 和慢速的外设、人工操作的矛盾, 特别是人机工作速度差异太大的矛盾日益突出, 严重制约着整个作业的执行速度。想办法去掉人工干预, 实现作业的自动过渡, 提高 CPU 的利用率成为迫切需要解决的问题。

为了摆脱人工干预作业的运行, 人们把计算机的输入/输出、运行控制、出错处理等工作编成常驻内存的一个监控程序, 通过执行监控程序来完成对计算机的使用、管理和操作, 实现了作业建立和作业过渡的自动化。这种监控程序就是操作系统的雏形。为了提高 CPU 的利用率, 减少等待作业输入/输出的时间, 采用了把作业成批送入计算机的批处理方式。作业间的转接和各作业的运行由监控程序自动控制, 减少了部分人工干预, 有效地缩短了作业运行前的准备时间。相对提高了 CPU 的利用率。但监控程序并不能解决高速的 CPU 和低速的外设之间的矛盾。

1.1.3 执行系统阶段

在单道系统中, 设备的利用率是非常低的。因为单道系统一次只让一个作业程序进入系统运行。某个设备工作时, 其他的设备都处于“闲置等待”状态。为了改善这种状况, 引入了多道程序设计技术, 出现了多道批处理操作系统。“多道”是指允许若干个作业同时进入内存, 使它们轮流交替地使用 CPU。多道方式大大提高了内存和 CPU 的利用率。进入 20 世纪 60 年代以后, 在硬件方面出现了“通道”和“中断”等技术, 实现了主机 CPU 和输入/输出设备的并行工作, 使操作系统进入了“执行系统”阶段, 对操作系统的真正形

成起到了重要的推动作用。

1.1.4 操作系统的形成与成熟阶段

20 世纪 60 年代,在早期的批处理系统中引入多道程序设计技术后产生了“多道批处理系统”,随后又出现了“分时系统”、“实时系统”。这三种系统的推出标志着“操作系统”的形成。随后,操作系统迅速完善、普及,直至进入了同时具有多道批处理、分时、实时等两种或三种处理能力的“通用操作系统”阶段。随着计算机科学技术的飞速发展,个人微型计算机、多处理器计算机和计算机网络的出现,促使操作系统的理论和实践进一步发展。从 20 世纪 70 年代后期开始,出现了微机操作系统、网络操作系统、分布式操作系统、嵌入式操作系统等。

一个成熟的操作系统,其形成是一个渐进的过程。可以说,随着计算机应用范围的不断扩大,计算机运行速度的不断提高,以及计算机系统功能的不断增强,系统可同时运行的程序由单道到多道,管理和控制系统的软件由简单到复杂,用户对硬件工作的细节了解由多到少,这些变化都推动了操作系统的发展。操作系统作为核心的系统软件,为其他系统软件和应用软件的运行构筑了一个平台,而且其本身在使用中也不断发展着,并趋于成熟。

1.2 操作系统的基本概念

1.2.1 操作系统的定义和特征

1. 操作系统的定义

操作系统是计算机系统最重要的系统软件,是其他软件的支撑软件。它管理着计算机系统资源,并通过这种管理为用户使用计算机提供公共的和基本的服务,从而成为用户与计算机之间的接口。操作系统追求的主要目标如下。

- (1) 充分地控制和有效地利用各种硬、软件资源,使整个系统能保持高效率的运转。
- (2) 提供一个计算机硬件系统与用户之间的接口,为用户提供一个简捷清楚的操作界面。
- (3) 合理地组织计算机系统的工作流程,改善系统性能。
- (4) 遵循国际标准,设计和构筑开放式的环境,支持可扩展的体系结构,支持应用程序的可移植性和可互操作性。

综上所述:操作系统是一个由大量的程序模块和数据结构集合而成的大型系统软件。它全面地控制和有效地管理着计算机系统的所有硬、软件资源,使用户有一个功能强大且可扩展的工作环境。合理地组织计算机的工作流程,向用户提供方便友好的操作界面。

2. 操作系统的特征

不同的操作系统具有不同的特点,但它们都有以下4个基本特征。

(1) 并发性。并发是指让多个程序同时在系统中运行。

在单CPU系统中,每一时刻仅能执行一道程序,并发性是宏观上的,而微观上,这些程序是在CPU上交替执行的。在多CPU系统中,并发程序不仅在宏观上是并发的,而且在微观上也是并行运行的。在分布式系统中,多台计算机并存,使程序的并发特征得到更充分的体现。

并行和并发是既相似又有区别的两个概念。并行是指两个或多个事件在同一时刻发生,而并发是指两个或多个事件在同一时间间隔内发生。

(2) 共享性。资源共享是指系统中的软硬件资源不再为某个程序所独占,而是供多个用户程序共同使用。操作系统要在多个并发程序间通过调度来分配处理器时间,并且保证对系统数据共享的正确性以及数据的完整性。

根据资源的属性不同,共享可分为互斥共享和同时共享两种方式。互斥共享是指系统中的资源虽能提供给多个进程使用,但在一段时间内却只允许一个进程访问该资源。同时共享允许在一段时间内,由多个进程同时对资源进行访问。当然,“同时”仍然是宏观上的,微观上,这些进程可能是交替地对该资源进行访问。

(3) 虚拟性。仅由硬件构成的计算机是个“裸机”,不仅功能有限,用户也无法直接使用。操作系统则扩充了“裸机”的功能。操作系统以硬件提供的基本功能为基础,采用不断扩充、逐层虚拟的分层结构。在分层结构中,上层依赖于下层功能,并对下层功能进行扩展后向再上一层提供服务。操作系统的各个程序模块分别对硬件逐层进行扩充和改造,最终形成了一个功能强大的、虚拟的计算机,出现在用户面前。

(4) 随机性。操作系统运行在一个随机的环境中。也就是说,各程序在何时被执行,多个程序间的执行顺序以及完成每道程序所需的时间都具有不确定性。系统无法对所运行程序的行为以及硬件设备的情况做任何假定,运行着的程序会在什么时候做什么事情是不可预知的。这就是随机性的含义。随机性并不是说操作系统无法很好地控制资源的使用和程序的运行,而是强调操作系统的设计与实现要充分考虑各种可能性,才能稳定、可靠、安全、高效地达到程序并发和资源共享的目的。

并发和共享是操作系统的两个最基本的特征,而这两者之间又是互相依存的。程序的并发执行带来资源共享的问题,而资源共享为程序的并发提供了支持。资源共享是以程序的并发执行为条件的,假若系统不允许程序并发执行,就不存在资源共享问题。另一方面,若系统不能对资源共享实施有效的管理,势必影响到程序的并发执行。

1.2.2 操作系统的功能服务

操作系统作为用户与计算机之间的接口,要为用户提供良好的操作界面,让用户感觉到易用、易学,这是操作系统一个最主要的职能。早期的计算机没有操作系统这个用户接口,只能直接通过机器语言使用计算机,这对没有计算机专业知识的人非常困难。有了操作系统,用户通过操作系统提供的操作命令和系统调用命令就可以方便、快捷、安全、可

靠地操纵计算机,运行自己的程序。随着图形用户接口的出现,使用户尤其是普通用户操作计算机变得更为简单。

下面就从不同角度来看操作系统能为我们提供什么服务。

(1) 从使用角度看,操作系统是一台比裸机功能更强,使用更方便,服务质量更高的虚拟机,为用户使用计算机提供了一个窗口和界面。

(2) 从功能角度看,操作系统是计算机资源的管理者,它负责对计算机全部软硬件资源的分配、控制、调度和回收。操作系统跟踪资源的使用情况,满足用户对资源的请求,合理分配资源,解决用户程序在执行过程中由于资源竞争而引发的各类事件,并尽可能使得系统中的各类资源处于“忙碌”状态,缩短任务完成时间,提高资源利用率。为用户程序的运行提供了良好的环境。

(3) 从组成结构看,操作系统是一个大型的系统软件,是由控制和管理系统运行的程序和数据集合组成。操作系统这个大型系统软件能为其他软件提供开发与运行环境,能与各种系统实用软件协作,使用户在操作系统平台上开发的各种应用软件能正常、高效率地运行。

1.2.3 操作系统的结构与性能指标

1. 操作系统的结构设计

操作系统是由很多相关程序模块组成的一个大型程序。每个模块的质量及它的整体结构对操作系统的性能都起着决定性的作用。人们通过软件开发的实践逐步认识到,软件结构是影响软件质量的内在因素,良好的结构可以提高软件的正确性、可维护性和工作效率。

在操作系统的开发过程中,操作系统的设计有以下几种主流的结构形式。

(1) 模块结构。早期的操作系统和现在的一些小型系统采用无序模块式结构。将一个大系统分成若干个相对独立的模块,每个模块又可以分成更小的模块,模块组成“积木式”的结构。每个模块都有定义良好的接口,相互间的调用不受约束。接口简单、结构紧凑且组合方便。系统具有很大的灵活性。存在的问题是:模块间数据联系过多,模块独立性差,系统结构不清晰,难以保证可靠性;当操作系统规模庞大、结构复杂时,这种不分层次、不加约束的结构模式很难保证系统的正确性和有效性。

(2) 层次结构。这种系统是把一个大系统分解成若干个基本模块,并按照一定的原则将这些模块排列成若干层,各层之间单向调用,即只能由上层调用下层,不能由下层调用上层。把最靠近硬件、直接与硬件有关的模块,比如 I/O 启动、中断管理、进程管理等,放在最内层。最靠近用户、与用户管理有关的模块,比如命令解释模块、作业管理模块等,放在最外层。其余部分按照文件管理、设备管理、存储管理模块的调用顺序,从外向内放在中间层次。层次结构的优点是各层之间单向调用。在设计低层软件时,可以不考虑高层软件的实现方法。

高层软件中的错误也不会影响到底层软件,这给调试和维护操作系统提供了较大方便。同时,在增加、修改或替换一个层次时不会影响到其他层次,有利于系统的维护和扩充。

(3) 微内核结构。微内核是 20 世纪 90 年代发展起来的操作系统开发新技术。设计的

思路是尽量缩小运行在系统态下的操作系统规模,使其只留下极小的内核。①内核与涉及硬件的操作运行在系统态,是整个操作系统最核心的部分。微内核减少了操作系统中与硬件直接作用的部分,最大限度地隐蔽硬件特性,以提高系统的安全可靠性和灵活性。②操作系统的服务功能则放在核外并运行在用户态。各种服务功能分解成若干相对独立的服务器(Server),每个服务器只提供一个方面的功能,例如文件服务、进程服务、终端服务或存储器服务等。而且,所有的服务器都以用户进程的形式运行,不直接访问硬件。

当称作客户(Client)的用户进程,需要操作系统的服务时,发送服务请求给相应的服务器进程,服务器进程收到请求后为用户服务。客户与服务器之间的消息传递由内核实现。整个操作系统的服务功能由能够“独立工作”的小单元群组而成。可以提高可靠性和灵活性,也简化了整体结构。

微内核中仅提供操作系统中最主要、最基本、最底层的功能,如进程通信、低级 I/O、低级进程管理、某些存储器管理等,因而它不是一个完整的操作系统。内核所做的主要工作仅仅是实现客户进程与服务器进程之间的消息传递。由于所有服务器都作为用户态进程运行,它们不直接使用硬件,因此若某个服务器发生了错误,那么相应服务也许失效,但是整个系统一般不会崩溃。微内核为通用操作系统的开发提供底层支持。在内核的基础上和模块化、层次化设计方法以及面向对象技术相结合,可以非常方便地开发出具有各种特点的操作系统。微内核结构能有效地支持多处理机运行,故非常适合于并行操作系统和分布式操作系统的开发。目前比较流行的、能支持多处理机运行的操作系统大多都采用微内核结构。这种结构也顺应了计算机网络化和分布式的发展潮流。

2. 操作系统的性能指标

一个好的操作系统应该有可靠性高、方便维护、并发性好、资源利用率高、适应性强等一些性能特征,常用如下指标来描述。

(1) 可靠性。系统可靠性是指系统发现、诊断、恢复硬件或软件故障的能力,以及对用户误操作的容错能力。可靠性实际上是计算机系统的正常使用率,该使用率称为可服务性。

可服务性=平均无故障时间/(平均无故障时间+平均故障修复时间)

平均无故障时间:系统能正常工作的无故障时间的平均值。该时间越长,可靠性越高。

平均故障修复时间:从故障发生至故障修复所需的平均时间,该时间越短,可维护性越好。

(2) 易用性。易用性是指计算机系统使用的简单性、可操作性、可携带性等。随着计算机的普及,人们对系统的人性化设计要求越来越高。

(3) 可移植性。可移植性是指操作系统在发展过程中,进行硬件环境移植,功能扩充、改进和完善所需要的工作量。通常用人年(或人月)来表示。系统的结构设计和软件开发工具都会影响系统的可移植性。

(4) 开放性。操作系统的开放性是指能够集成不同厂家生产的设备,且能正确有效地协同工作,支持应用程序的可移植性和互操作性。

(5) 系统资源利用率。在给定时间内,系统中某种资源实际使用时间所占的比例。操作系统应使系统资源尽可能被高效地应用。

(6) 系统响应时间。系统从接受数据到输出结果的时间间隔。

(7) 系统吞吐量。系统在单位时间内所处理的信息量。以每小时或每天所处理的任务数来度量。

1.3 操作系统的分类

1.3.1 多道批处理操作系统

多道批处理系统诞生于 20 世纪 60 年代初期,至今仍是主要的操作系统类型之一。“多道”是指内存中可以同时存放多个作业,多个作业交替使用 CPU 和系统中的各种资源。宏观上并行微观上串行。“批处理”是指作业成批进入系统,成批地处理,成批地退出。

多道批处理系统中,作业成批地输入系统并暂存在外存中,组成后备作业队列,系统按一定的调度原则从后备作业中选择一个或多个装入主存进行处理。成批作业在系统中形成了一个能自动转接的、连续的作业流,系统自动依次执行每个作业,从而缩短了作业之间的转换时间。当一批作业运行完毕,输出结果后,系统便接收下一批作业。

在多道批处理系统中,用户给系统提交作业时,除了提交作业有关的程序和数据外,还必须提交作业说明书或作业控制卡,用来表明用户对其作业的控制意图(作业运行时的资源请求、对可能产生的运行错误的相应处理等)。由作业控制程序或命令解释程序解释执行,并且提供相应的服务。用户不能直接干预作业的运行过程。

多道批处理系统的主要特点是:

- 系统资源利用率和吞吐量都得到了提高。
- 作业执行过程中不能进行人-机交互。
- 作业完成时间难以估计,周转时间较长。

在多道批处理系统中,一个作业大体经历了从提交、后备、执行、完成到退出几个阶段。实现多道批处理的主要输入/输出手段是 Spooling 技术。

批处理系统以脱机为标志,用户不能干预作业的运行,因而调试程序困难。多道批处理比较适用于程序已反复检查无误、执行中预计不会出现异常、对完成时间要求不严的作业。

多道批处理系统主要通过以下 4 种途径来提高系统效率:

- 允许多道作业并发工作,减少了处理器的空闲时间,即提高了处理器的利用率。
- 作业调度可以按一定的组合去选择装入主存储器的作业,如果能把请求使用不同设备的作业搭配在一起,比如计算型作业和 I/O 型作业搭配,则可使各种设备同时工作,从而充分提高计算机系统的资源利用率。
- 作业执行过程中,可以不再访问低速的设备,而是直接在高速的磁盘上存取信息,从而缩短了作业执行时间,使单位时间内的处理能力得到提高。
- 作业成批输入,自动选择和控制作业执行,减少了人工操作时间和作业交接时间,有利于提高系统的吞吐量。

1.3.2 分时操作系统

多道批处理系统的形成,提高了资源的利用率和系统的吞吐量。但是多道批处理系统的一个不足之处是不提供人-机交互能力。成批作业在运行过程中,完全不允许用户干预,这会给用户造成一些不便。将多道程序设计技术和分时技术结合在一起,在同时兼顾系统效率和方便用户的理念指导下,产生了分时操作系统。

所谓“分时”,就是多个用户对系统资源(主要是 CPU)进行时间上的分享。分时技术把 CPU 的运行时间分成很短的时间片。时间片是系统规定进程一次使用处理机的最长时间。时间片的长短可以因不同系统而异,通常是几十到几百毫秒左右。

操作系统以时间片为单位,轮流为每个终端用户服务。每个用户程序进入 CPU 后,使用一个时间片,若在一个时间片内用户程序未能完成,就会被暂时中断执行,等待下一轮循环调度再继续执行。让出的 CPU 给另一个用户程序使用。由于计算机的处理速度快,只要时间片的间隔取得适当,用户不会察觉轮转过程中的“停顿”,每个用户的感受好像是独占一台计算机。同时,分时操作系统允许用户通过终端与系统进行广泛的人-机对话来获得系统的各种服务,并控制作业程序的运行。这种交互性为用户调试、修改及控制程序的执行提供了极大的便利。分时系统以联机操作为标志。微观上,每个用户作业轮流运行一个时间片。宏观上,多个用户同时工作,共享系统资源。

分时系统有以下 4 个基本特征。

(1) 交互性。用户能与系统进行人-机对话,即用户可以通过键盘或鼠标输入命令,请求系统服务和控制程序的运行。

(2) 及时性。终端用户的请求能在几秒甚至更短时间内获得响应。

(3) 多路性。指同时有多个程序并发执行。系统可同时为多个用户终端提供服务。多个用户同时工作,共享系统资源。

(4) 独占性。系统对多个用户的快速轮转调度,使得每个终端用户感觉就像是独占了 CPU。这种独占是逻辑上的,而物理上是多用户共享 CPU。

分时系统追求的主要目标是“同时响应”,因此响应时间是衡量一个分时系统的重要性能指标。而进程交换时调进和换出的系统开销,用户数量的多少,时间片值的大小这些因素都会影响“响应时间”。在单道分时系统中,进程频繁地换进换出和交换信息量过大都会增加系统的响应时间。如果系统同时有 n 个用户,时间片为 q ,则每个用户作业轮转一次所需时间 $T=nq$ 。用户数 n 和时间片值 q 都正比于响应时间 T 。因而适当地减少时间片长度和减少用户数,可改善对用户的响应时间。当然,这是以降低系统性能为代价的。还可以通过减少对换信息量来改善响应时间。采用重入码文件和引入虚拟存储技术都可减小对换信息量。

1.3.3 实时操作系统

在军事、工业控制、金融证券等领域,对计算机系统有着特殊的要求,那就是对用户