

万水 Windows 2000 技术丛书

Windows 2000 Server

系统结构与设计

徐 栋 任东强 编著



中国水利水电出版社

www.waterpub.com.cn

内 容 提 要

Windows 2000 操作系统可谓家喻户晓,但多数人并不了解其内核技术的精髓所在,本书通过介绍 Windows 2000 Server 的核心技术,让您感觉其博大精深之处。对于中、高级系统开发人员来说,本书不失为一本好的参考书,尤其对微软操作系统感兴趣的读者而言,更是如此。

·本书内容虽较深,但介绍较为详细,且采用通俗易懂的语言加以叙述。

图书在版编目(CIP)数据

Windows 2000 Server 系统结构与/设计/徐栋,任东强编著. —北京:
中国水利水电出版社, 2000.8

(万水 Windows 2000 技术丛书)

ISBN 7-5084-0425-4

I. W… II. ①徐…②任… III. 服务器-操作系统, Windows 2000 Server
IV. TP316.7

中国版本图书馆 CIP 数据核字(2000)第 65673 号

书 名	Windows 2000 Server 系统结构与/设计
作 者	徐栋 任东强 编著
出版、发行	中国水利水电出版社(北京市三里河路 6 号 100044) 网址: www.waterpub.com.cn E-mail: sale@waterpub.com.cn 电话: (010) 63202266 (总机)、68331835 (发行部)
经 售	全国各地新华书店
排 版	抖斗制作中心
印 刷	北京市天竺颖华印刷厂
规 格	787×1092 毫米 16 开本 23.5 印张 531 千字
版 次	2000 年 8 月第一版 2000 年 8 月北京第一次印刷
印 数	0001—5000 册
定 价	35.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社发行部负责调换

版权所有·侵权必究

前 言

抛开对 Microsoft 公司的个人情感，有一点我们必须承认的是，Microsoft 公司向来是而且在未来的相当一段长的时间内将依然是计算机操作系统的霸主。在新千年到来时他们不失时机地推出了新一代的操作系统 Windows 2000，从其名字我们就可以感觉到微软公司要在 21 世纪的软件业继续称霸的雄心壮志。

1. Windows 2000 有什么了不起

如果稍微详细地对比一下 Windows 2000 与以往的 Windows（不管是 Windows 95/98 系列还是 Windows NT），就能感受到 Windows 2000 中发生的一系列革命性的变化。无论我们是软件开发人员还是软件管理人员，抑或只是普通的计算机用户，我们都有必要了解或理解 Windows 2000，因为 Windows 2000 带来的革命将不仅是对计算机行业的震撼，更多的是对我们的学习、工作、生活的影响，谁让我们身处信息时代呢？

Windows 2000 的发布不只是一次简单的升级，它将成为继 Windows 95 和 Windows NT 之后微软最为重要的一次产品发布。该重要性不仅是针对微软公司目前的处境，同时对整个 IT 行业也是如此。盖茨在一次大会上声称：“说微软把未来都压在了 Windows 2000 上从客观上讲并不十分偏激”。

2. Windows 2000 到底有什么不同

Windows 2000 是在 Windows NT 4.0 操作系统的基础上开发的，集 Windows NT 技术和 Windows 9X 的优点于一身，并在此基础上发展了许多新的特性和功能，如智能镜像、终端服务、分布式文件系统、磁盘定额、DNS 增强以及活动目录等等。

3. Windows 2000 安全性是否更高

回答当然是肯定的。Windows 2000 的 NTFS 文件系统支持新的“加密文件系统”功能，可以加密硬盘上的重要文件，保证用户文件的安全，只有通过安全登录的用户才能访问这些加密文件。

据说微软为了检验 Windows 2000 之安全，特地诚邀天下黑客拜访其一台安装了 Windows 2000 的桌面机，结果黑客无功而返。

如果您是 Windows 9X 的老客户，您可能还没能体会到 Windows 2000 赋予您 Administrator（系统管理员）权限所带来的喜悦。作为 Windows 2000 的最高权限使用者，您可以严格控制其他用户对您的机器的使用权，包括禁止安装、使用软件，每个用户都有

自己独立的目录，不能访问其他人的文件，对于多人共享的台式机能保护每个用户的数据安全。

4. Windows 2000 能不能将微软带上一个新的高度

就像当年的 Windows 95/98 产生的市场效应一样，相信读者在仔细阅读了我们编写的这本《Windows 2000 Server 系统结构与设计》一书之后，会有一个自己的答案。

本书由徐栋、任东强负责编写，另外参加编写的人员还有柴长锋、薛勇、石磊、黄永辉、王奇、李永、张梁、张文达、黄健兵、刘雪原、郝卫艳、刘丰、梁庆锋、李国林、张永学、石光华、王成峰、罗建军等。全书由魏红统稿。在此一并表示感谢。

由于编者水平有限，疏漏之处在所难免，欢迎广大读者和专家给予批评指正，不胜感谢。

对本书内容有疑问的读者，可以致电（010）62565533 转 3301 进行咨询，也可以访问 www.doudou.com.cn 以获得相关信息。

编者

2000 年 7 月

目 录

前言

第一章 操作系统概述	1
1.1 计算机系统的组成	1
1.2 什么是操作系统	2
1.2.1 操作系统作为虚拟机	2
1.2.2 操作系统作为资源管理器	3
1.3 操作系统发展历史	4
1.3.1 第一代计算机（1945~1955）：真空管和插板	4
1.3.2 第二代计算机（1955~1965）：晶体管和批处理系统	4
1.3.3 第三代计算机（1965~1980）：集成电路芯片和多道程序	5
1.3.4 第四代计算机（1980~现在）：个人计算机	7
1.4 Windows 2000 操作系统	9
1.4.1 Windows 2000 的市场需求	9
1.4.2 Windows 2000 设计目标	11
1.5 Windows 2000 的系统结构	15
1.5.1 Windows 2000 模型	15
1.5.2 Windows 2000 结构	20
1.5.3 环境子系统	23
1.5.4 本机服务	23
1.5.5 国际化	24
1.5.6 结构化异常处理	26
1.6 小结	28
第二章 Windows 2000 新特性	30
2.1 活动目录	30
2.2 分布式安全性扩展	31
2.3 加密	32
2.4 安全配置编辑程序	33
2.5 分布式文件服务	33
2.6 NTFS 扩展	33
2.7 Microsoft 管理控制台	34
2.8 Microsoft 软件安装程序	35
2.9 存储管理	35
2.10 IntelliMirror	36

2.11	应用程序的开发	36
2.12	作业对象	37
2.13	Alpha 上的大容量内存	38
2.14	Microsoft 终端服务器	41
2.14.1	Windows 2000 终端服务器的组件	41
2.14.2	Windows 2000 中的操作系统支持	42
2.15	即插即用和电源管理	42
2.15.1	即插即用的发展	43
2.15.2	在 Windows 2000 中的实现	43
2.15.3	驱动程序的更改	44
2.15.4	Windows 2000 即插即用结构	44
2.16	小结	47
第三章	实现 Active Directory	48
3.1	Active Directory 概述	48
3.1.1	什么是 Active Directory	48
3.1.2	Active Directory 特性	48
3.2	Active Directory 结构和同步复制	50
3.2.1	逻辑结构	50
3.2.2	物理结构	53
3.2.3	站点内的同步复制	54
3.3	Active Directory 概念	54
3.3.1	模式	54
3.3.2	全局目录	55
3.3.3	名字空间	55
3.3.4	命名约定	56
3.4	规划 Active Directory 的实现	56
3.4.1	规划名字空间	57
3.4.2	规划站点	57
3.4.3	规划组织单元	58
3.5	安装 Active Directory	58
3.5.1	Active Directory 安装向导	58
3.5.2	数据库和共享系统卷	59
3.5.3	域模式	59
3.6	管理多域环境	60
3.6.1	信任关系	60
3.6.2	安全性	60
3.6.3	规划一个树	61

3.7 小结	61
第四章 对象管理	62
4.1 执行体对象	62
4.1.1 使用对象	63
4.1.2 对象结构	66
4.1.3 对象类型	67
4.2 管理对象	68
4.2.1 对象名	69
4.2.2 对象句柄	73
4.2.3 对象方法	75
4.3 保护对象	77
4.3.1 存取令牌	78
4.3.2 存取控制表	79
4.4 小结	80
第五章 进程和线程	81
5.1 进程	81
5.1.1 进程的定义	81
5.1.2 进程本质	82
5.1.3 CreateProcess 函数	82
5.1.4 进程的创建过程	86
5.1.5 进程间的通信	95
5.1.6 进程调度	102
5.1.7 进程的结束	110
5.2 线程	110
5.2.1 CreateThread () 函数	111
5.2.2 CreateThread () 流程	112
5.2.3 线程调度	114
5.3 小结	125
第六章 内存管理	126
6.1 Windows 2000 内存管理简介	126
6.1.1 虚拟内存	126
6.1.2 文件映像	129
6.1.3 堆管理	131
6.2 地址空间分布	132
6.2.1 用户地址空间布局	133
6.2.2 系统地址空间布局	133
6.3 深入内存管理器	136

6.3.1	组件	136
6.3.2	调整内存管理器	136
6.4	地址转换	137
6.4.1	转换虚拟地址	138
6.4.2	页目录 (page directory)	138
6.4.3	进程和系统页表	139
6.4.4	页表项	139
6.4.5	页面内的字节	141
6.4.6	转换后备缓冲区	141
6.5	页错误处理	141
6.5.1	无效的 PTE	142
6.5.2	原形 PTE	142
6.5.3	入页 I/O	143
6.5.4	冲突页错误	144
6.5.5	页面文件	144
6.6	虚拟地址描述符	144
6.7	工作集	145
6.7.1	页面调度策略	145
6.7.2	进程工作集	146
6.7.3	平衡集管理器和交换程序	148
6.7.4	系统工作集	148
6.8	页帧数据库	149
6.8.1	页面列表动态	149
6.8.2	更改页面写入程序	150
6.8.3	PFN 数据结构	151
6.9	区域对象	152
6.10	小结	153
第七章	Windows 2000 文件系统	154
7.1	NTFS 的设计目标和特性	154
7.1.1	高端文件系统需求	154
7.1.2	NTFS 的其他特性	156
7.2	NTFS 的内部结构	158
7.3	NTFS 在磁盘上的结构	159
7.3.1	卷	159
7.3.2	簇	159
7.3.3	主控文件表 (MFT)	160
7.3.4	文件引用号	161

7.3.5	文件记录.....	161
7.3.6	文件名.....	162
7.3.7	常驻属性和非常驻属性.....	163
7.3.8	文件名索引.....	164
7.3.9	数据压缩.....	165
7.4	可恢复支持.....	166
7.4.1	文件系统设计的发展.....	167
7.4.2	记录.....	169
7.4.3	恢复.....	172
7.5	容错支持.....	175
7.5.1	卷管理特性.....	175
7.5.2	容错卷.....	176
7.5.3	NTFS 坏簇恢复.....	177
7.6	小结.....	179
第八章	I/O 系统.....	180
8.1	I/O 系统结构和模型.....	180
8.1.1	I/O 管理器.....	181
8.1.2	I/O 函数.....	181
8.2	设备驱动程序.....	183
8.2.1	驱动程序结构.....	184
8.2.2	同步.....	185
8.3	数据结构.....	186
8.3.1	文件对象.....	186
8.3.2	驱动程序对象和设备对象.....	187
8.3.3	I/O 请求包.....	188
8.4	I/O 处理.....	189
8.4.1	对单层驱动程序的 I/O 请求.....	189
8.4.2	对分层驱动程序的 I/O 请求.....	190
8.5	小结.....	191
第九章	COM 体系结构.....	192
9.1	软件的发展.....	192
9.1.1	剪贴板 (Clipboard) 和动态数据交换 (DDE).....	192
9.1.2	OLE 1.0.....	193
9.1.3	VBX 组件.....	194
9.1.4	COM 和 OLE 2.0.....	194
9.1.5	OCX 组件.....	195
9.1.6	OLE 的扩展.....	195

9.1.7	ActiveX	196
9.1.8	对象超集: COM+	197
9.1.9	对象技术的现状	198
9.2	COM 理论	199
9.2.1	COM 思想概述	199
9.2.2	COM 的优点	200
9.2.3	COM 的需求	204
9.2.4	COM 的接口	206
9.2.5	COM 的公共接口 IUnknown	209
9.2.6	COM 库服务	210
9.3	小结	213
第十章	网络基础	214
10.1	网络结构	215
10.1.1	ISO 开放系统互连模型	215
10.1.2	网络 API	217
10.1.3	网络术语	219
10.1.4	数据信号传输	220
10.2	局域网	223
10.2.1	局域网参考模型	224
10.2.2	逻辑链路控制协议	226
10.2.3	CSMA/CD 介质访问控制协议	231
10.2.4	标记环介质访问控制协议	236
10.2.5	FDDI 网	239
10.2.6	基于交换技术的网络	242
10.2.7	千兆位以太网	244
10.2.8	ATM 局域网	246
10.2.9	无线局域网	247
10.3	广域网	250
10.3.1	计算机交换分机 (CBX)	250
10.3.2	CBX 的结构	251
10.3.3	ISDN 交换系统	253
10.3.4	X.25 分组交换网	253
10.3.5	高速广域网发展的动因	254
10.3.6	帧中继网	255
10.3.7	ATM 网	259
10.3.8	ATM LAN 仿真	262
10.4	网络管理	264

10.4.1	局域网管理技术.....	264
10.4.2	网络管理标准.....	265
10.4.3	简单网络管理协议 (SNMP)	267
10.5	小结	267
第十一章	网络管理协议	269
11.1	NetBEUI	269
11.2	TCP/IP 协议	271
11.2.1	IP 寻址	271
11.2.2	多点传送寻址	272
11.2.3	IP 数据报	272
11.2.4	端口	274
11.2.5	用户数据报协议	275
11.2.6	传输控制协议	275
11.3	IPX/SPX 协议	292
11.4	DLC 协议	294
11.5	AppleTalk 协议	295
11.5.1	AppleTalk 低层协议	295
11.5.2	AppleTalk 中层协议	296
11.5.3	AppleTalk 上层协议	296
11.6	网络驱动程序的 NDIS 环境	296
11.7	分布式应用程序环境	297
11.8	小结	300
第十二章	网络服务技术	301
12.1	电子邮件传递	301
12.1.1	RFC 822 邮件格式	302
12.1.2	邮局协议	303
12.1.3	简单邮件传输协议	305
12.1.4	SMTP 扩展	306
12.1.5	网际消息访问协议版本 4	307
12.1.6	多用途的网际邮件扩充	310
12.1.7	安全/多用途的网际邮件扩充协议	312
12.1.8	增强的私密电子邮件	313
12.1.9	高度保密	314
12.1.10	邮件列表和列表服务器	315
12.1.11	新闻和 Usenet	315
12.2	文件传输和文件系统	318
12.2.1	文件传输协议	319

12.2.2	一般文件传输协议.....	321
12.2.3	公用网际文件系统.....	322
12.2.4	WebNFS.....	323
12.3	文本会谈.....	323
12.3.1	Internet 在线聊天.....	324
12.3.2	多用户网络游戏.....	326
12.3.3	基于 Web 的聊天.....	327
12.4	高级服务.....	327
12.4.1	Microsoft 事务服务器.....	327
12.4.2	Microsoft 消息队列.....	334
12.4.3	Microsoft 群集服务器.....	340
12.5	公司范围内的联网和分布式安全性.....	345
12.6	小结.....	347
第十三章	Windows 2000 的安全性.....	348
13.1	安全基础.....	349
13.1.1	安全机制.....	349
13.1.2	安全模型.....	350
13.2	安全保护对象.....	351
13.2.1	安全保护对象.....	351
13.2.2	安全描述器和访问控制.....	352
13.2.3	访问令牌与模仿.....	354
13.3	安全审核.....	355
13.4	登录.....	356
13.4.1	安全问题的产生.....	356
13.4.2	登录.....	356
13.4.3	存取控制.....	359
13.4.4	NTFS 文件系统安全功能.....	360
13.5	提高安全的策略.....	360
13.5.1	用户口令的设置.....	361
13.5.2	访问许可权的设置.....	361
13.5.3	RAS 安全.....	362
13.5.4	服务器的安全性.....	363
13.6	小结.....	363

第一章 操作系统概述

计算机如果离开了软件将成为一堆无用的废铜烂铁。有了软件，计算机可以对信息进行存储、处理和检索，可以显示多媒体文档、搜索 Internet 并完成其他工作。计算机软件大致分为两类：系统软件和应用软件。

系统软件管理计算机本身及应用程序，它是计算机硬件和应用程序之间实现交互的接口；应用软件执行用户最终所需要的功能，例如日常使用的子处理软件和表格处理软件。而操作系统（Operating System）则是最基本的系统软件，它控制计算机的所有硬件和软件资源并提供开发应用程序的基础。本章主要讲述操作系统在整个计算机系统中的地位和发展历史。

1.1 计算机系统的组成

现代计算机的硬件系统是非常复杂的，它包括：

- 一个或多个处理器
- 若干内存（常称为 RAM——随机存取存储器）
- 磁盘
- 打印机
- 网络接口
- 其他输入/输出设备

编写一个程序来管理所有这些器件以正确地使用它们，即使不考虑性能优化也是一件很困难的事情。如果每个程序员都必须处理磁盘如何工作，再加上每读一个磁盘块都有几十种因素可能导致操作出错，那么对于一般的程序员即使仅仅编写一个最基本的磁盘的输入/输出应用程序也是相当困难的。

许多年以前人们就认识到必须找到某种方法将硬件的复杂性同程序员分离开来。经过不断探索和改进，目前采用的方法是裸机上加载一层软件来管理整个系统，同时给用户提供一个更容易理解和进行程序设计的接口，这被称为虚拟机（Virtual machine）。这样一层软件就是操作系统。这种处理方式如图 1-1 所示。

接着是微程序（microprogram）通常存放在只读存储器中，它是一层很原始的软件，用来控制设备并向上一层提供一个更清晰的接口。微程序实际上是一个解释器，它先取得机器语言指令，如 ADD、MOVE 和 JUMP 等，然后通过一个动作序列来执行这些指令。例如，为了执行一条 ADD 指令，微程序必须按以下步骤进行：

确定运算数据的位置→取数→相加→存放得数。

由微程序解释执行的这一套指令称为机器语言。机器语言并不是硬件的组成部分，但硬件制造商通常在手册中给出机器语言的完整描述，所以许多用户将其认定为真正的“计

算机”。

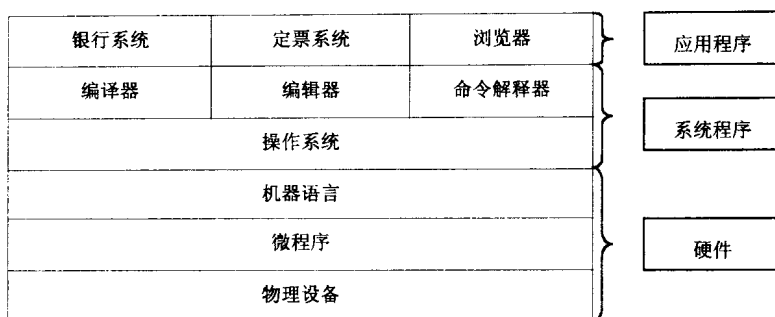


图 1-1 计算机的层次结构

采用精简指令集计算机（RISC）技术的计算机没有微程序层，其机器指令通过硬件逻辑直接执行。例如 Motorola 680x0 有微程序，而 IBM Power PC 则没有。

机器语言典型地有 50 到 100 条指令，大多数用来完成数据传送、算术运算和数值比较等操作。在这个层次上，通过向特殊的设备寄存器写特定的数值来控制输入/输出设备。例如将磁盘地址、内存地址、读字节数和操作类型（读/写）等值写入特定的寄存器便可完成硬件操作。实际操作往往需要更多的参数，而操作完成后的返回状态也非常复杂。进一步而言，对于许多 I/O 设备，时序在程序设计中的作用非常重要。

操作系统的主要功能之一就是将所有这些复杂性隐藏起来，同时为程序员提供一套更加方便的指令。比如，“从文件中读一个数据块”在概念上比低层的“移动磁头臂，等待旋转延迟”之类的细节来得简单、方便。

在操作系统之上是其他系统软件，包括命令解释器（shell）、窗口系统、编译器、编辑器及类似的独立于应用的程序。要注意它们本身并不是操作系统的组成部分，尽管它们通常由计算机厂商提供。这一点很重要，操作系统专指在核心态（kernel mode），或称管态（supervisor mode）下运行的软件，它受硬件保护而免遭用户的篡改。编译器和编辑器运行在用户态（user mode）。如果用户不喜欢某一个编译器，他可以自己重写一个，但他却不可以写一个磁盘中断处理程序，因为这是操作系统的一部分，而且硬件阻止用户对它进行修改。

系统软件之上是应用软件，这些软件可以是购买的或者是用户自行开发的，它们用来解决特定的问题，如字处理、表格处理、工程计算机或者电子游戏等。

1.2 什么是操作系统

对于大多数计算机用户使用计算机就是使用该计算机的操作系统，但要精确地给出操作系统的定义却很困难，原因是操作系统完成两项相对独立的任务，下面将逐项进行讨论。

1.2.1 操作系统作为虚拟机

对多数计算机而言，在机器语言一级的体系结构（指令集、存储组织、I/O 和总线结

构)上编程是很困难的,尤其是输入/输出操作。例如考虑使用多数 PC 机采用的 NEC PD765 控制器芯片(或功能等价的芯片)来进行软盘 I/O 操作。PD765 有 16 条命令,它通过向一个设备寄存器装入特定的数据来执行这些命令。命令数据长度从 1 到 9 字节不等,其中包括:读写数据、移动磁头臂、格式化磁道、初始化、检测磁盘状态、复位、校准控制器及设备。

最基本的命令是读数据和写数据。它们均需要 13 个参数,所有这 13 个参数被封装在 9 个字节中。这些参数指定的信息有:欲读取的磁盘块地址、每条磁道的扇区数、物理介质的记录格式、扇区间隙、以及对已删除数据地址标识的处理方法等。当磁盘操作结束时,控制器芯片返回 23 个状态及出错信息,它们被封装在 7 个字节中。此外,程序员还要注意步进电机的开关状态。如果电机关闭,则在读写数据前要先启动它(有一段较长的加速时间)。还要注意电机不能长时间处于开启状态,否则将损坏软盘,所以程序员必须在较长的启动延迟和可能对软盘造成损坏之间作出折衷。

显然,程序员不想涉及硬件的这些具体细节(也包括硬盘,它与软件不同,但同样很复杂)。他需要的是一种简单的高度抽象的设备。一种典型的抽象是一张磁盘包含了一组命名的文件,每个文件可以被打开,然后进行读写,最后被关闭。其中的一些细节如数据记录格式,当前步进电机的开启状态等则对用户隐藏。

这种将硬件细节与程序员隔离开来,同时提供一个简捷的命名文件方式的程序就是操作系统。与磁盘抽象类似,它还隐藏了其他许多低层硬件的特性,包括中断、时钟、存储器等。总之,操作系统提供的每一种抽象都较低层硬件本身更简单、更易用。

从这个角度看,操作系统的作用是为用户提供一台等价的扩展计算机,或称虚拟机,它比低层硬件更容易编程。本书的内容正是详细说明操作系统如何做到这一点。

1.2.2 操作系统作为资源管理器

上述虚拟机模型是一种自顶向下的观点。按照自底向上的观点,操作系统则用来管理一个复杂系统的各个部分。现代计算机都包含处理器、时钟、磁盘、鼠标、网络接口、激光打印机以及其他许多设备,从这个角度看,操作系统的任务是在相互竞争的程序间有序地控制这些设备的分配。

设想在一台计算机上运行的三个程序同时试图在一台打印机上输出计算机结果,那么可能头几行是程序 1 的输出,接下来几行是程序 2 的输出,然后又是程序 3 的输出等等,最终打印结果将是一团糟。操作系统采用将打印输出缓冲到磁盘上的方法可以避免这种混乱。当一个程序结束后,操作系统暂存在磁盘文件上的输出结果送到打印机,同时其他程序可以继续运行产生新的输出结果,而这些程序并不知道这些输出没有立即送至打印机。

当一台计算机(或网络)有多个用户时,因为用户可能相互影响,所以管理和保护存储器、I/O 设备以及其他设备的需求随之增加。而且用户往往不仅需要共享硬件,还要共享信息(文件、数据库等)。总之,此时操作系统的首要任务是跟踪资源的使用状态、满足资源请求、提高资源利用率,以及协调各程序和用户对资源的使用冲突。

1.3 操作系统发展历史

操作系统经历了一个漫长的发展过程，下面对此进行简要的回顾。由于在历史上操作系统与计算机体系结构存在非常密切的联系，我们将按照计算机的换代历程讲述操作系统的发展状况。

第一台真正的数字计算机是英国数学家 Charles Babbage(1792~1871)设计的。Babbage 投入毕生精力去建造他的“分析机”，但却没能让它成功地运行起来。因为它是纯机械式的，而当时的技术不可能使分析机的零部件达到他所需要的精度。很显然，分析机没有操作系统。

有趣的是，Babbage 认识到他的分析机需要软件，于是他雇佣了一个年轻的女子，英国著名诗人拜伦的女儿 Ada Lovelace 为他工作。Ada 由此成了世界上第一位程序员，Ada 程序设计语言就是用她的名字命名。

1.3.1 第一代计算机（1945~1955）：真空管和插板

从 Babbage 之后一直到二战，数字计算机几乎没有什么进展。在 40 年代中期，哈佛大学的 Howard Aiken、普林斯顿高等研究院的 John Von Neumann（冯·诺依曼）、宾夕法尼亚大学的 J. Presper Eckert 和 William Mauchley、德国电话公司的 Konrad Zuse、以及其他一些人都成功地使用真空管建造了计算机。这些机器非常庞大，往往使用数万个真空管，占据几个房间，然而其运算速度却不如现在最便宜的个人计算机。

在计算机出现的早期，每台机器都有一个小组专门来设计、制造、编程、操作和维护。编辑全部的机器语言，通过在一些插板上的硬连线来控制其基本功能，这时没有程序设计语言（甚至没有汇编语言），操作系统更是闻所未闻。机器的使用方式是程序员提前在墙上的机时表上预约一段时间，然后到机房将他的插板插到计算机里，在接下来的几小时里计算自己的题目。这时的计算机很不可靠，因为几万个真空管中任何一个发生故障，计算机就无法运行。这个阶段基本上所有的题目都是数值计算机问题，例如计算正弦和余弦函数表。

到 50 年代早期，出现了穿孔卡片，这时就可以不用插板，而将程序写在卡片上然后读入计算机，但其他过程则依然如故。

1.3.2 第二代计算机（1955~1965）：晶体管和批处理系统

50 年代发明的晶体管极大地改变了计算机的状况。这时的计算机已经很可靠，厂商可以成批地生产计算机并卖给客户，客户可以长时间地使用它来完成一些有用的工作。至此，第一次将设计人员、生产人员、操作员、程序员和维护人员分开。

这个时期，计算机安装在空调房间里，有专人操作。由于其价格昂贵，仅有少数大公司、主要的政府部门和大学才买得起。运行一个作业（一个或一个组程序）时，程序员首先将程序写在纸上（用 FORTRAN 或汇编语言），然后用穿孔机制成卡片，最后将这些卡

片交给操作人员。

计算机运行完当前任务后，其计算结果从打印机上输出，操作员从打印机上取得运算结果并送到输出室，程序员就可从该处取到运算结果，然后，操作员再从卡片上读入另一个任务。如果需要 FROTRAN 编译器，操作员还要从别处取来读入计算机。当操作员在机房里走来走去时，许多机时被浪费掉了。

由于当时计算机非常昂贵，很自然地人们开始想办法减少机时的浪费，答案就是批处理程序（现代操作系统的前身），它从磁盘带上将第一个作业读入并运行，其输出写到第一盘输出磁带上，而不是打印出来。每个作业结束后，操作系统自动地读入下一个作业运行。当这一批作业完全结束后，操作员取下输入和输出磁带，将输入磁带换成下一批作业，然后把输出磁带拿到一台 1401 机器上进行脱机打印。

下面我们来考察一个典型的输入作业结构。该系统由一张 \$JOB 卡片开始，该卡标识出所需的最大运行时间（分钟）、计费标识、以及程序员的名字。随后是一张 \$FORTRAN 卡片，它通知操作系统从系统磁带上装入 FROTRAN 语言编译器。在此之后是待编译的源程序，然后是 \$LOAD 卡片，它通知操作系统装入刚编译好的目标程序（编译好的目标程序通常写到暂存磁带上，需要显式装入）。接着是 \$RUN 卡片，它告诉操作系统运行该程序并使用其后的数据。最后，\$END 卡片标识作业结束。这些原始的控制卡片是现代作业控制语言和命令解释的先驱。

第二代计算机主要用于科学计算，例如解编微分方程。这些题目大多用 FORTRAN 语言和汇编语言编程。典型的操作系统是 FMS（FORTRAN Monitor System, FORTRAN 监控系统）和 IBSYS（IBM 为 7094 机配备的操作系统）。

1.3.3 第三代计算机（1965~1980）：集成电路芯片和多道程序

在 60 年代初期，多数计算机厂商都有两条完全不同并且互不兼容的生产线：一条是面向字的复杂科学计算机和工程计算机的计算机，如 IBM 7094；另一条是面向字符的商用计算机，如 IBM1401，主要被银行和保险公司用于磁带归档和打印服务。

对厂商来说，开发和维护两种完全不同的产品是很昂贵的。同时，许多新的计算机用户开始只需要一台小计算机，到后来则可能需要一台较大的计算机，而且要求能够更快地执行原有程序。

IBM 公司试图通过引入 360 系统来解决这两个问题。IBM360 是一个软件兼容的计算机系列，在该系列中，低档机与 1401 相当，高档机则比 7094 功能强很多。这些计算机只在价格和性能（最大存储器容量、处理器速度、允许的 I/O 设备数量等）上有差异。由于所有的计算机都有相同的体系结构和指令集，因此为一种型号机器编写程序可以在其他所有型号的机器上运行（起码在理论上可行），而且 360 被设计成既可用于科学计算，又可用于商业计算，这样一个系列的计算机便可以满足所有用户的要求。在随后的几年里，IBM 陆续推出了 360 的后续机型，如用户熟知的 370、4300、3080 和 3090 系列。

360 是第一种采用集成电路（小规模）芯片的主流机型。与采用分立晶体管制造的第二代计算机相比，其性能价格均有很大提高。360 很快就获得了成功，由此其他主要厂商