

计算机最新技术培训教材

PEKING UNIVERSITY PRESS

2001
新书

于 娉 王 峰 编 著

U NIX

学 习 教 程



- 优秀的操作系统
- 学习UNIX从入门到精通的教材

北京大学出版社

<http://cbs.pku.edu.cn>



计算机最新技术培训教材

UNIX 学习教程

于 娉 王 峰 编著

北 京 大 学 出 版 社

北 京

内 容 提 要

在 UNIX 操作系统不断完善, 并被越来越多的人接受的今天, 如果忽视对它的学习, 那将会是每一个电脑爱好者的一大遗憾。本书就是专门致力于讲解 UNIX 操作系统使用方法的一本学习教程, 它特别适用于有一定计算机使用经验的读者。如果读者使用过 DOS 或 Windows 等操作系统, 那么读者将能在本书的帮助下快速掌握 UNIX 操作系统的基本使用方法。

本书分 10 章介绍了 UNIX 操作系统, 内容覆盖了 UNIX 操作系统的历史、文件操作、网络功能和 Shell 等知识。在介绍时采用了结合实例的方法, 让读者既能学会如何操作, 又能马上看到操作结果的一个范例, 以期读者能够快速掌握。

图书在版编目 (CIP) 数据

UNIX 学习教程/于姆, 王峰编著. —北京: 北京大学出版社, 2001. 2

(计算机最新技术培训教材)

ISBN 7-301-01218-7

I. U… I. ① 于… ② 王峰 II. UNIX—操作系统, UNIX—技术培训
IV. TP316.81

中国版本图书馆 CIP 数据核字 (2001) 第 45817 号

书 名: UNIX 学习教程

著作责任者: 于 姆 王 峰

责任编辑: 汉 明

标准书号: ISBN 7-301-01218-7/TP·56

出 版 者: 北京大学出版社

地 址: 北京市海淀区中关村北京大学校内 100871

电 话: 010-62765013 (编辑部)

网 址: <http://cbs.pku.edu.cn>

排 版 者: 南方立德 (Leader) 信息技术中心

印 刷 者: 北京神剑印刷厂 (原国防科工委印刷厂)

发 行 者: 北京大学出版社

经 销 者: 新华书店

787 毫米×1092 毫米 16 开本 20.75 印张 515 千字

2001 年 2 月第 1 版 2001 年 2 月第 1 次印刷

定 价: 29.00 元

前 言

UNIX 从诞生到现在,已经有近 30 年的历史。经过数代人的不懈努力,UNIX 成为当代主流操作系统,几乎在各种类型的计算机中都配有 UNIX 操作系统。UNIX 系统是一个开放性的操作系统,作为操作系统平台,国际已经有统一标准。而它的应用软件应有尽有,包括了商业、科学计算以及工程设计的各个方面。而且,随着 Internet 的迅猛发展,UNIX 的高度安全性、可靠性使得它获得了更大的发展空间。为了使读者掌握 UNIX 最基础的知识,能够进行简单的系统维护、系统管理和系统开发。本书主要介绍以下内容:

UNIX 操作系统简介 介绍了 UNIX 的发展过程,以及对将来的一些展望。

UNIX 的联机帮助 介绍了联机手册的使用方法。

UNIX 基础 主要介绍了 UNIX 文件和常用命令,UNIX 目录,shell 等一些 UNIX 的基础知识。

UNIX 文件管理 主要介绍了 UNIX 文件系统的基础,文件结构,目录名和文件名,权限,目录层次设备等一些涉及 UNIX 文件管理的知识。

UNIX 文本编辑器 主要介绍了 UNIX 文本编辑器的使用方法。

shell 主要介绍了 UNIX shell 程序的知识。

UNIX 游戏 主要介绍了 UNIX 游戏的知识。

UNIX 下编写 C 程序 主要介绍了 UNIX 下编写 C 程序的原理和方法。

UNIX 系统的维护 主要介绍了 UNIX 系统维护的知识。

UNIX 总则 总结了 UNIX 的命令和 ASCII 码。

本书尽可能做到以下几点:

- (1) 着重介绍基本概念,尽可能做到简明易懂。
- (2) 着重介绍 UNIX 中的常用命令和重要命令的使用方法,并配有大量的实用示例帮助读者理解。
- (3) 力图内容详尽而叙述简洁。

经过 30 多年的发展,UNIX 的内容极其丰富,本书所涉及到的内容,可谓是沧海一粟。希望读者通过本书的学习,能在较短的时间内掌握 UNIX 基本的使用方法。

本书由于娉、王峰主编,参加本书编写工作的还有王京、李涛、谢刚、王颖、陈戈、王丽娜、曹伟、陈禹、张灿、巨磊、杨晓荣、唐前华、冉钊、吴宁、吕杰、崔羽、刘万鹏等。由于时间仓促,再加上个人的实际经验毕竟是有限的,书中难免存在一些疏漏欠妥之处,望广大读者能够谅解,并提出宝贵意见。

编 者

2001 年 1 月

目 录

第一章 UNIX 操作系统简介.....	1
1.1 什么是操作系统.....	1
1.2 操作系统.....	1
1.3 UNIX 的诞生和发展.....	2
1.4 最重要的 X 客户程序: xterm.....	9
1.5 学习使用一个图形用户界面.....	10
第二章 UNIX 的第二联机帮助.....	16
2.1 联机手册与 man 命令.....	16
2.2 在终端上显示联机手册.....	16
2.3 联机手册的组织.....	17
2.4 在 man 命令中指定节号.....	18
2.5 如何参考联机手册.....	19
2.6 手册页的格式.....	19
2.7 一种快速查寻命令作用的方法: whatis	22
2.8 查找一条命令: apropos	22
第三章 UNIX 基础.....	24
3.1 起步.....	24
3.2 文件和常用命令.....	31
3.3 目录.....	41
3.4 shell	44
3.5 UNIX 系统的其余部分.....	54
第四章 UNIX 文件管理.....	55
4.1 文件系统的基础.....	55
4.2 文件结构.....	59
4.3 目录和文件名.....	60
4.4 权限.....	64
4.5 I 节点	68
4.6 目录层次.....	74
4.7 设备.....	76

第五章 正文编辑器	81
5.1 ed 的使用	81
5.2 vi 的使用	89
5.3 emacs 的使用	105
第六章 shell	115
6.1 shell 程序	115
6.2 shell 的特殊类命令	131
6.3 C_shell 简介	136
6.4 命令行结构	141
6.5 创建新命令	147
6.6 命令参数	148
6.7 程序输出作为参数	152
6.8 shell 变量	153
6.9 shell 程序里的循环	159
6.10 bundle 合并	161
6.11 为什么说 shell 是可编程的	163
第七章 UNIX 游戏	164
7.1 UNIX 游戏程序	164
7.2 如何终止一个游戏（及一般的 UNIX 程序）	165
7.3 怎样玩游戏	165
7.4 UNIX 游戏程序概述	166
7.5 消遣与新奇游戏	167
7.6 计算器	168
7.7 万年历	170
第八章 UNIX 环境下编写 C 程序	172
8.1 一个 C 程序的例子	172
8.2 cc 编译命令	173
8.3 程序示例	174
8.4 UNIX 环境下 C 程序设计语言	176
8.5 C 程序软件包	205
8.6 ADB 导引	225
8.7 LINT: 程序检查程序	249
8.8 MAKE: 维护计算机程序的程序	258
第九章 UNIX 系统的维护	266
9.1 UNIX 系统概述	266
9.2 FSCK: UNIX 文件系统检查程序	274

9.3	UNIX I/O 系统	279
9.4	再生系统软件	285
第十章	UNIX 总则	292
10.1	UNIX 命令分类一览表	292
10.2	ASCII 码表	310

第一章 UNIX 操作系统简介

1.1 什么是操作系统

UNIX 是一个操作系统。操作系统协调、管理、指挥各硬件的工作，它简化了应用程序的构成并按用户的需要执行应用程序，这是通过控制以下系统资源来实现的：

- 中央处理器（CPU）
- 主要存储器（或随机存取存储器 RAM）
- 辅助存储器（磁盘、磁带或只读存储器 ROM）
- 其他外设（如打印机、串行通信硬件或 CD-ROM 驱动器）

一个操作系统提供一组基本服务，处理一些繁琐的工作，如任务调度、将程序装入内存并执行、读写数据、安全管理、监视资源使用情况以及与别的系统或远程设备通信等等。而用户界面是操作系统的组成部分，用于与该计算机系统的用户打交道。如人们对 UNIX 的初步了解就是它古老而高效的命令行界面，它提示用户输入命令，读取用户命令并尝试执行。

UNIX 还维护着一些数据，用以定义用户的操作环境。操作环境是包含优先级和数据值的列表，用以定义诸如程序缺省行为、用户定期检查邮箱的时间、使用的终端类型等等。安全管理是操作系统设计的一个重要内容，但有的操作系统，如现今流行的个人计算机操作系统 MS-DOS 却没考虑它。多用户系统则提供各种不同的安全机制，保护任一个用户不受系统中其他用户的侵犯。重要的是，它能保护系统免受偶然的或蓄意的破坏。

另一个可选的功能是用户记账程序，它负责跟踪每个用户对系统资源（CPU 时间、磁盘空间等等）的使用情况，它还能监视哪些命令正在执行及其执行的频率。记账程序的数据常被系统管理员用作诊断工具，有时它也被作为付账的依据。提供记账程序的系统也常常向管理员提供将此功能封锁掉的方式，因为此功能能在极短的时间内产生大量的数据。

1.2 操作系统

操作系统是管理计算机硬件的一个或多个程序，它允许用户运行程序、在磁盘文件中存储数据、打印信息等等。当计算机电源打开时，首先运行的程序是操作系统，只有在操作

系统运行后, 用户才能使用计算机。MS-DOS 和 UNIX 是两个使用最广泛的操作系统, MS-DOS 是基于个人计算机的操作系统, 一次只允许一个用户使用; 相反, UNIX 是一个多用户操作系统, 这意味着两个或多个 (甚至上百个) 用户能同时使用同一台计算机。

UNIX 操作系统从诞生到现在已经有 20 多年了。在这 20 多年中, UNIX 从一个只供实验室内部使用的默默无闻的操作系统发展成为适用于个人计算机、工作站、小型、中型、大型及巨型计算机的著名的操作系统, 使得运行 UNIX 系统的计算机越来越多。本章将向读者介绍这个系统的诞生、发展过程及其特点, 并与其他一些常用的操作系统进行比较, 使读者初步了解 UNIX 的优点及不足。最后, 本章还将向读者介绍一下 UNIX 今后的发展方向, 使读者对 UNIX 的未来也有所了解。

1.3 UNIX 的诞生和发展

谈到 UNIX 的诞生, 就不能不提到另一个操作系统 —— Multics。Multics 操作系统是美国国防部委托 MIT (麻省理工学院) 和 AT&T (美国电话电报公司) 的贝尔实验室研制的一个多用户交互式分时操作系统。该系统从 1965 年开始开发, 但由于它规模庞大, 开发进展缓慢, 所以 1969 年初贝尔实验室退出了该项研究, 这也导致了 Multics 操作系统最终未能达到原定的设计目标。

虽然 Multics 操作系统最终未能成功, 但它却极大地推动了操作系统的发展。UNIX 操作系统从根本上讲, 借鉴了 Multics 操作系统设计时的一些思想。最初的 UNIX 操作系统是在 1969 年由贝尔实验室的系统程序员 K.Thompson 和 D.M.Ritchie 开发成功的。这两人曾经参与了 Multics 操作系统的开发, 并且当时已经取得了一定的成绩, 实现了一种简单的、规模很小但用户非常容易接受的操作系统。

在贝尔实验室参与 Multics 操作系统的设计工作时, Thompson 和 Ritchie 等人就在运行 Multics 的机器上编写了一个名叫 “Space Travel” 的游戏程序, 这个游戏模拟了星体在太阳系中的运动。当贝尔实验室退出了 Multics 的研制工作后, Thompson 和 Ritchie 等在试图把这个游戏移植到 PDP-7 机器上时, 发现若使用单用户交互式操作系统时花费的成本太高。1970 年 Thompson 用 PDP-7 机器专用的汇编语言编写了一个供单个用户使用的操作系统, 该系统继承了 Multics 中文件系统的树型结构等一些特点。为了区别 Multics 操作系统能满足多个用户要求的多用户操作特点, 这个操作系统被命名为 UNIX。其中, “Uni” 与 “Multi” 相对应, 表明只用于单用户。但是, 随着 UNIX 的逐步发展, 它现在已经成为了一个支持多用户的通用操作系统, 这倒和当初的设想相背了。下面我们就简要介绍一下 UNIX 的发展过程。

1970 年, 采用 PDP-7 机器上专用的汇编语言编写的 UNIX 操作系统获得成功, 并安装在 PDP-11/20 上运行。

1971 年, UNIX 系统版本 1 正式形成, 并开始安装在贝尔实验室内部的机器上运行。

1972 年, UNIX 版本 2 形成, 首次实现了管道功能。从这一年开始, UNIX 开始被移植到 PDP-11/34、40、45、60 等机器上运行。

1973 年, 为了将 UNIX 移植到不同类型的机器上运行, Ritchie 研制成功高级语言 C 语言, 并成功地用 C 语言改写了 UNIX 系统。由于 UNIX 原来是用 PDP-7 机器上专用的汇编语言写成的, 所以它原来只能在特定的机器上运行, 而这次经过用 C 语言改写后, UNIX 能够被移植到任何计算机上, 大大增强了通用性。实际上, Thompson 为了便于移植 UNIX 曾研制了一种 B 语言, 但从整体上看 B 语言的应用并不广泛。C 语言是基于 B 语言的, 由于用它成功地改写了 UNIX 而闻名于世, 并开始在全世界获得广泛应用。用 C 语言改写后的 UNIX 形成了 UNIX 版本 5。

1974 年, UNIX 操作系统在美国《Communication of ACM》杂志上以《The UNIX Time-Sharing System》(UNIX 分时系统)一文发表, UNIX 操作系统正式向外界披露, 标志着 UNIX 操作系统的诞生。此时 UNIX 操作系统已经在大学里得到较多的使用并获得好评。

1975 年, UNIX 版本 6 发表, 其中实现了多通道技术, 从而使 UNIX 成为一个真正的多用户分时系统。从这一年开始, UNIX 向外界提供, 并被广泛安装在各大学的 PDP-11 系列机上运行。

1979 年, UNIX 版本 7 发表, UNIX 开始支持巨型文件(能支持多达 1000 兆字节的文件), 并被移植到 32 位机 Interdata 8/32 上, 得到了西部电气公司的正式承认。同年, 贝尔实验室发布了适用于 VAX-11 计算机的操作系统 UNIX 32V。

1982 年, 美国电话电报公司发表了 UNIX 系统 III。这期间由于 UNIX 系统的源代码已经开放, 使得 UNIX 开始出现了许多变种。开发这些变种最著名的是加利福尼亚大学的伯克利分校。1980 年伯克利分校发布了用于 VAX-11 机上的 BSD 4.0 版和 BSD 4.1 版。从那时起, UNIX 操作系统走上了以 AT&T 和伯克利分校两家为主的不同的发展道路。

1983 年, AT&T 公司发布了 UNIX 系统 5 Release 1 版本, 同年, 伯克利分校发布了 BSD 4.2 版。

1984 年, 美国 IBM 公司(国际商用机器公司)发布了用于 IBM/PC 机器上的 UNIX。

1985 年, Microsoft 公司(微软公司)发布了用于 16 位 PC 机上的 UNIX 变体 Xenix 操作系统。接着, Cray 公司发布了用于超大型机的 UNIX 系统, AT&T 则推出了 UNIX 系统 5 Release 2 版本。

1986 年, UNIX 系统 5 Release 3 版本发表。同年, BSD 4.2 升级到了 BSD 4.3。Sunmicro 公司则开发了基于 BSD 4.2/4.3 系统的 SUN 工作站操作系统 SUNOS。同年, MIT(麻省理工大学)为 UNIX 操作系统开发了图形界面系统 X-Window。

由于同时存在各种不同版本和种类的 UNIX 系统, 就有必要对 UNIX 系统进行标准化。于是, 美国 IEEE 协会(电气和电子工程师协会)成立了 POSIX 委员会, 用来标准化 UNIX 系统的工作, 定义一个统一的 UNIX 系统。但是, 由于种种原因, 这个标准化工作事实上

是由 UI (UNIX International) 和 OSF (Open Software Foundation) 两大集团来进行的。其中, UI 主要由 AT&T 和 Sunmicro 作为支持, DEC、HP 和 IBM 公司则是支持 OSF 的主要力量。这两个组织虽然所做的工作不一样, 但是他们给出了 UNIX 系统统一的定义, 那就是能运行 UNIX 应用软件的操作系统就是 UNIX。这样定义之后就要求各家 UNIX 系统不论其内部实现方法和原理怎样, 只要他们能提供一个标准的 UNIX 用户界面就可以了。从用户的角度来看, 无需确切地知道是哪一家的 UNIX 系统, 操作方法都一样。

在这个标准的基础上, UI 于 1989 年推出了 UNIX 系统 5 Release 4.0 版本。这个版本实现了 BSD 4.3 和系统 5 Release 4.0 版本在界面上的统一, 而且它集中了前面各个版本的优点。而 OSF 则在 1990 年推出了基于 BSD 4.3 的 OSF/1。事实上, 这两个组织的 UNIX 系统还在竞争, 但是竞争已经转入到如何提高可靠性和友好的用户界面等方面。

在微机领域, SCO 公司在 1988 年推出了适用于 X86 系列机的 SCO UNIX 系统。这个系统投放市场后受到用户的欢迎, 这是由于它具有支持多任务处理和安全性高等特点。

上面我们介绍了 UNIX 系统的诞生和发展情况, 从中读者很容易发现, 由于 UNIX 在源代码上实行开放, 因此在短短的 20 多年中得到飞速发展, 包含技术不断增加, 性能不断提高, 这一切都体现了开放源代码的好处。但是, 我们也看到, 由于开放了源代码, 使得同时存在多家公司都在开发 UNIX 系统的现象, 这就极容易导致“分流”, 最后还要重新对它进行统一, 这就使得 UNIX 的发展走了一些弯路。

1.3.1 UNIX 系统

首先, 让我们来研究 UNIX 这个名字。UNIX 并不是一个缩略词, 它更近似于 Multics 的双关语。Multics 是一个早期的操作系统, UNIX 许多重要的设计概念和系统特征都从它而来。事实上, UNIX 最初被认为是 Multics 的一个较高级的版本(称作 Unics), 但后来它在市场上取代了 Multics。因此 UNIX 也可写作 Unics。通常用大写字母表示这个名词, 就如本书一样, 是为了与 AT&T 和多数流行系统软件发行商在文档中建立的传统表示相一致。

当前的 UNIX 系统实现的功能比 70 年代在贝尔实验室产生的最初的 UNIX 要多得多和复杂得多, 但是, 它仍贯彻着力求优雅和功能齐备的宗旨。

1.3.2 UNIX 系统的设计

UNIX 与其众多竞争对手的重要区别是它的灵活性。当前的许多操作系统都与固定的软件生产商捆绑在一起, 而且通常还与固定的处理器家族联姻, DOS 对 Intel 处理器的依赖就是一例。70 年代的 UNIX 已经可以在广泛的硬件平台上运行了。它的开发者使用 C 语言编写大多数源代码, 高级语言的使用使 UNIX 具备可移植性(即容易在不同的计算机平台上实现)。虽然也有一定的工作量, 但转移到新的平台是可能的。UNIX ports, 如它的名称一样, 几乎可以在所有的计算机平台上运行, 有时它甚至可以在别的操作系统上运行。

UNIX 操作系统最基本的核心称为内核，这一软件层与计算机硬件的联系最为紧密，具体情况如下所示：

外壳 (shell)	实用程序	应用程序
UNIX 内核		
硬件		

UNIX 软件的第二层是用户程序集，包括命令解释程序、编程工具、实用程序和应用程序。请注意命令解释程序，它实现用户界面，通常称为 shell，是一个简单的用户程序。具有不同的功能及不同屏幕显示的 shell 可以相互取代。

本书介绍的是标准 UNIX 命令行界面格式，因为多数用户通过远程终端或 PC 机上的仿真程序访问 UNIX 系统。而且即使你的系统在直接连接的控制台上为你提供了图形用户界面，但知道怎么使用命令行、怎么使用传统工具创建和编辑文件仍然非常重要。

1.3.3 UNIX 的多样性

在贝尔系统的基础上，UNIX 在全世界的大学和研究机构中，被成千上万的编程人员修改和增添了许多功能。近来，许多商业组织、用户团体和政府实体都在致力于对 UNIX 进行整理以建立一个标准，这是件喜忧参半的事情。

对 UNIX 的经常性修改大大增加了它对于最终用户的使用价值，并能一直保持新鲜感。但这也产生了缺乏一致性的系统，引起市场混乱。这种混乱是由占领桌面操作系统领域的几个低档操作系统商制造的。尽管如此，当商业计算机用户发现它能力强、功能完善，以及 UNIX 适于商务的需要时，UNIX 便继续扩大它在专业工作站和大型计算机系统以外的市场份额。

许多操作系统虽然名称并不是 UNIX，但事实上，它们只是 UNIX 操作系统的分支版本。如属于 SUN 的 Berkely UNIX (BSD)、Xenix、SUN OS 和 Interactive UNIX 以及大量的其他操作系统，都是基于 AT&T UNIX 源代码的，该代码现由 Novell 控制。其他一些类似 UNIX 的操作系统也是模拟一些版本的 UNIX 行为编写出的。这些系统包括 Coherent 和 QNX。通过使用 Hamilton C Shell 或 MKS Toolkit 及基于 Windows 的系统，也可以通过命令行对 UNIX 有所感知。

1.3.4 GUI: 图形用户接口

如果你曾经使用过运行 Microsoft Windows 或 OS / 2 操作系统的 Machintosh 或 PC 机，你就知道什么是图形用户接口 (GRAPHICAL USER INTERFACE)。该系统不但带有键盘，还有鼠标；屏幕上不仅显示字符，还有窗口以及图像。你可以将窗口和图像作为操作对象，

有些图形用户接口被称为 GUI。

在使用 GUI 之前,有几个基本概念需搞清楚。首先,需要学会使用两种输入设备:键盘和点设备 (POINTING DEVICE)。点设备,顾名思义,就是使用它点取屏幕上的一部分。最通用的点设备是鼠标,但也有轨迹、触摸屏等等。我们假设使用的是鼠标,但差别都不大。

典型的情况是,在桌面上移动鼠标,屏幕上的指针也随之移动。指针是一个小图像,通常是一个箭头。如果想指向一个图像,就要移动鼠标直到屏幕上的指针停在那个图像上。使用有些 GUI 会发现,在你从屏幕上一个区域移向另一个区域时指针会发生变化。

点设备通常有按键。尽管有时用得较少, X Window 仍使用三个按键。为方便起见,这几个按键从左至右都编了号,左边的是 1 号键,中间的是 2 号键,3 号键在右边。使用 GUI 时,1 号键也就是最左边的键用得较多 (如果你是左撇子,可以将顺序调换一下)。

关于 GUI 的另一个比较重要的特点是:它把屏幕分成若干个被称为窗口 (WINDOWS) 的并带有边框的区域。窗口的边界通常 (但不总是) 是矩形的。窗口可以像桌面上的纸一样可以重叠,而且可以根据需要改变窗口的大小和位置。

在每个窗口中都可以做不同的事。比如,你可以使用几个不同的窗口,每个窗口运行不同的程序。工作时,可以很容易地从一个窗口转换到另一个窗口,这样就允许你根据需要使用每个程序。普通字符型终端一次只能看见一个程序 (除非具有特殊的软件)。事实上, X Window 和一般意义的窗口系统发展的主要动力就是允许一个用户运行多道程序。

为了使用 GUI,需要理解许多重要的概念并掌握其他一些知识,但在此之前,我们还是先讨论一下 X Window。

1.3.5 什么是 X Window

X Window 是一个支持图形用户接口的系统。为了方便起见,通常将 X Window 称做“X”。这样,你就可以问:“你的计算机能运行 X 吗?”。

X 的含义是为显示图形数据的程序提供标准服务。例如,你可以在这台计算机上运行程序,而在另一台机器的屏幕上输出, X 隐藏在背后,将所有细节组织起来,提供对这种结构的支持。

X 是 Project Athena 的一个部门 MIT 开发的。现在, X 是由一个被称为 X Consortium 的独立组织来维护的。1987 年 MIT 发行了 X 的 11 版,被称为 X 11。X 11 的不同版本有 X 11.4, X 11.5 等。顺便提醒一下,正确的名字是“X Window”而不是“X Windows”。

1.3.6 窗口管理器: mwm, olwm, twm

实际的图形用户界面不是由 X 本身,而是由被称为窗口管理器 (WINDOW MANAGER) 的程序提供的。窗口管理器控制窗口的形状、字符以及图像。

X 有几种窗口管理器,你的系统中可能仅用一个;否则,你可以选择一个。工作时,

真正打交道的不是 X，而是窗口管理器。系统的外观和感觉怎样，依赖于你所选用的窗口管理器。

虽然窗口管理器种类较多，但有三种最通用，它们是 mwm (Motif 窗口管理器)、olwm (Open Look 窗口管理器) 以及 twm (Tab 窗口管理器)。Motif 是开放软件基金组织 (The Open Software Foundation) 的产品，应用在很多系统上。Open Look 是由 AT&T 和 SUN 开发的，应用于许多 SUN 系统之上。Tab 源于 X 系统。如果你要选择一个，最好选用 Motif 或 Open Look，因为它们使用起来更容易些，而且外观比 Tab 好看。

使用图形用户界面就像开车，会使用这个就很容易学会使用另一个。我们进一步打个比喻，如果把 twm 比喻成一个手动传输系统的汽车，那么 mwm 和 olwm 就是一个自动传输系统的汽车。

如果你曾经用过运行 Microsoft Windows 或 OS / 2 操作系统的 Machintosh 或 PC 机，你就会使用图形用户界面，再学习基于 X 的一种 GUI 就没有什么难点了。实际上，如果你用过 OS / 2 并熟悉 Presentation 管理器的 GUI，你就会发现用 mwm 特别容易。Motif 就是参照 Presentation 管理器的外观和工作方式设计的。

1.3.7 X 服务器与 X 客户

可以想象，如果你自己写一个提供图形用户界面的程序，那工作量一定很大。你要完成画窗口的许多细节，移动图像、追踪鼠标等等，而且，你的程序还要非常清楚屏幕、键盘和鼠标这些硬件的特性。

如果你的程序非常流行，一定有许多人要在各种机器上使用它，你就不得不对你的程序做适当的修改，以适用于各种不同的屏幕、键盘和鼠标。绝大多数的程序都会致力于怎样保持图形用户界面的详细内容。

X Window 系统的设计目的就是为了使编写和使用图形程序都很容易。为实现这个目标，X 提供了标准的服务程序，即显示服务器 (DISPLAY SERVER)。该显示服务器负责与图形用户界面打交道的所有细节。

我们花点时间复习一下术语：能通过网络来提供资源的程序或计算机叫服务器，使用资源的程序叫客户。

举例来说，假如你运行一个程序，它使用存储在另一台计算机上的文件，这个程序就是客户，而文件是由文件服务器通过网络传送到你的程序中的。相类似地，你要向另一台机器发送一个被打印的文件，另一台计算机就是打印服务器。

在 X 系统中，“DISPLAY”这个词不仅仅指你的屏幕显示。DISPLAY 是一个专用术语，它指屏幕、键盘以及鼠标。也就是说，DISPLAY 是由所有的与你的程序打交道的设备组成的。

显示服务器是一个为其他程序管理屏幕、键盘和鼠标的程序。当你写 X 程序时，无需关心输入 / 输出的细节，你的程序能够调用显示服务器完成这些工作。例如，一个 X

程序需要在屏幕上画一个窗口，只要告诉显示服务器所画窗口的大小和位置即可，由显示服务器具体实施这些细节，而且 X 程序也无需了解所用屏幕的类型。

这就是说 X 程序具有较高的可移植性。只要有基于 X 系统的显示服务器来完成实际的输入和输出任务，它可以与任何计算机上的图形用户接口打交道，使用任何的屏幕、键盘和鼠标。

借用网络术语，显示服务器通常被称为 X SERVER；同样地，任一使用显示服务器的程序都被称为 X CLIENT。换句话说，X 客户是一个在 X 下运行的程序，它使用 X 服务器的资源来处理输入和输出问题。

X 系统提供了 50 多个在 X 下运行的应用程序。用 X 专用语讲，系统提供了 50 多个客户 (client)。比如有一个程序可以显示一个时钟，告诉人们现在的时间。

当你使用 X 的时候，你的计算机在执行一个 X 服务器的程序，X 服务器再与你想使用的 X 客户程序打交道。举例来说，你在同时执行 5 个 X 客户程序，可以根据需要，从一个程序到另一个程序来回切换。屏幕上，每个程序都在自己的窗口区域内显示。所有 X 客户程序的输入 / 输出都是由机器上运行的一个 X 服务程序来处理的。

1.3.8 使用 X 执行远程计算机程序

你可能要说，所有这些东西都很不错（甚至很有趣），但我真的仅仅为了要使用图形用户接口而要去了解这么多的东西吗？实际上，没有必要陷入到学习怎样使用 Machintosh、Microsoft Windows 或 OS / 2 之中。

你必须搞清楚 X CLIENT 与 SERVER 的关系是因为它可以提供非常理想的服务。既然显示（屏幕+键盘+鼠标）是由一个 X 服务器替所有的 X 客户处理的，我们就可以有效地将输入 / 输出从该过程中分离出来。这样，运行在你的机器上的 X 服务器程序就可以服务于在其他机器上执行的 X 客户程序。

当然，你的 X 服务器程序能为你机器上的 X 客户程序服务，但只要有网络将两台机器连接起来，你的 X 服务器也可以为其他机器上的客户程序服务。

这是一个很完美的 X 系统：它使用图形用户界面，你可同时与几个窗口打交道，其中每个窗口运行一独立的程序。虽然 X 服务程序在你的机器上运行，但窗口中的程序可能正在网络中其他的机器上运行。

比如，有三个程序正在你自己的计算机上运行，有一个程序是你的一位朋友的，它运行在大厅的一台计算机上，还有另外一个程序运行在另一座楼内的大型机上。每个程序都在自己的窗口内运行，你可以使用一个图形用户界面来控制所有内容。

这样，你可以在一台计算机上使用基于 X 的图形用户界面，运行网络中其他机器上的程序和任何 X 客户程序。

最后还要说明一点：由于 X 有几种窗口管理器，如果使用的图形用户接口不同，屏幕显示的画面就有所不同。

比如，你和你的朋友肩并肩在工作，你们分用两个图形终端。你决定使用 `mwm` (Motif)，而她用的是 `olwm` (Open Look)，你们都在运行刚才所描述的 5 个程序，但是由于你们使用不同的 GUI，你们的屏幕函数和内部动作就不一样。

1.4 最重要的 X 客户程序：xterm

一个 X 客户是一个运行在 X Window 系统中的一个程序。作为一个图形系统，X 被设计成只支持图形显示。可是，有不少时间要在 shell 提示符下一个接一个地输入 UNIX 命令，这样就需要明确的字符终端。

由于这是一个基本的需要，所以 X Window 系统提供了一个专为此目的而设计的 X 客户程序，在窗口内运行，仿真一个终端。这个程序就叫 `xterm`。`xterm` 运行时，建立一个小窗口，就像一个小终端。这样，如果你有 4 个不同的 `xterm`，每个都在自己的窗口内运行，看起来就像 4 个不同的终端，而且都已注册。你可以根据需要从一个移向另一个。

`xterm` 能模仿两种类型的终端：作为字符终端而广泛使用的 VT-100；作为图形终端使用的 Tektronix 4014。绝大多数情况是使用 `xterm` 仿真 VT-100 (缺省值)。

我们以前提到，X 不提供图形用户界面的全貌，它是由所选取的窗口管理器提供的。因此，窗口管理器本身也是一个 X 的客户，也就是说，该程序运行在 X 下。一会儿我们将看到，窗口管理器也需要在一个 `xterm` 窗口内被启动。

换句话说，在随系统提供的所有 X 客户程序中，`xterm` 是必不可少的。你不仅需要它仿真一个终端以便输入 UNIX 命令，而且还需要它启动一个窗口管理器来产生图形用户界面。

下面我们开始使用 X Window: `xinit`、`xterm`、`twm`、`mwm`、`olwm`。

首先，要清楚只有你的终端支持 X，你才能用 X。也就是说，你需要有一个带有图形显示屏的工作站，或一个特别终端。它被称为 X 终端 (X TERMINAL)，是专门为运行 X 而设计的，而且无论你使用什么主机，X Window 软件必须已经安装且是可访问的。

需要你做多少工作，要根据系统管理员是否为你建立了一个能自动启动的 X 及一个窗口管理器的系统来决定。如果没有，你自己就不得不输入适当的命令。

有三种基本可能性，我们从自动的无需我们做什么的一种说起。

为了启动 X 更容易些，有些系统使用一个被称为显示管理器 (DISPLAY MANAGER) 的程序，该程序能减缓你的注册过程。当你开始工作时，会看到一条信息，比如 “Welcome to the X Window System.” 以及一个注册窗口，你可以输入你的用户标识符和口令。

注册成功后，显示管理器就自动开始做每件事情，包括 X 的自启动以及窗口管理器的启动。也就是说，如果你的系统有一个显示管理器，你真正要做的只有注册一件事。

如果没有显示管理器，UNIX 将为你提供一个标准的 `login` 提示。你输入用户标识符和口令之后就注册了。此时，会出现下面两种情况之一：若系统管理员已经为你建了账号，一

一旦注册成功，X 就会启动，窗口管理器也将开始工作。由于有显示管理器，你就不会再遇到什么麻烦，可以准备工作了。可是，如果所有东西还没有被自动建立，你就不得不手工建立系统了。如果注册后，X 没有启动，你仍处在一般的 shell 提示符下，就知道是出现了这种情况。这时你要输入两个命令：一个是用于启动 X Window，另一个是用于启动窗口管理器。

在 shell 提示符下，输入命令：

`xinit`

等一会儿，`xinit` 将开启 X Window 系统产生第一个 `xterm`，它是一个窗口（在有些系统中，可能有另外比 `xinit` 好的命令）。

一旦 X 启动并看到了 `xterm` 窗口，你就要指出希望使用哪个特殊的窗口工作，这时，只需简单地移动鼠标，让指针指在这个窗口里。

现在你已经处于 `xterm` 窗口里，可输入命令启动窗口管理器了。命令是窗口管理器的名字后加上“&”字符。你需要问问周围的人，系统使用的是哪种窗口管理器。

你还可以使用 `twin` (Tab)，它是随 X 一起提供的。但如果有 `mwm` (Motif) 或 `olwm` (Open Look)，最好还是选用它们。

比如，你使用 Motif，就在 `xterm` 窗口中输入下面的命令：

`mwm &`

过一会儿，窗口管理器就启动了，然后，你就可以开始工作。

概括起来，在 shell 提示符下，输入 `xinit` 命令，一旦 X 启动，就移到 `xterm` 窗口，输入窗口管理器名并加上“&”字符，比如，`mwm &`。

为什么要将“&”加到命令行尾？“&”字符是告诉 UNIX，我们启动的这个程序应自己运行，它处于“后台”。换句话说，窗口管理器应有自己的生命周期。

如果你忘记了“&”字符，UNIX 会启动窗口管理器，但不会释放键盘和屏幕。一旦出现这种情况，按下 `intr` 键（通常是 `^C` 或 `DELETE`）终止程序，然后再重新正确输入命令。

最后一点：因为你使用 X，你可以根据需要创建或删除一个窗口。当你删除一个窗口时，其中的每个程序都将被终止。所以，如果你删除了原始的 `xterm` 窗口，运行在其中的窗口管理程序就将终止，你的图形用户界面就消失了！

为了避免这种情况发生，有个好方法就是把原始的 `xterm` 窗口单独留下来。如果你想输入 UNIX 命令，你可以再创建一个 `xterm`。事实上，你想创建几个就能创建几个。有些人喜欢将原始的 `xterm` 窗口最小化以便将其移开。

1.5 学习使用一个图形用户界面

学习使用一个图形用户界面是容易的，但学习的方法是通过在自己的系统上练习和实践，而不能只是读书。