

学校的理想装备

电子图书·学校专集

校园网上的最佳资源

Linux 操作系统 (一)

介绍 Linux





返回总目录

第一部分介绍 Linux

第 1 章 Linux 操作系统

- 1.1 操作系统和 Linux
- 1.2 Linux 和 Unix 的历史
- 1.3 Linux 总览
- 1.4 Internet 上的 Linux 软件及信息源
- 1.5 Linux 发行版本
- 1.6 OpenLinux
- 1.7 总结

第 2 章 安装 Linux

- 2.1 硬件,软件各信息的要求
- 2.2 为 Linux 分区开辟磁盘空间
- 2.3 创建 OpenLinux 安装盘
- 2.4 安装 Linux

- 2.5 安装和配置 X-Windows
- 2.6 安装声卡的驱动程序:声音
- 2.7 Lisa

第 3 章 Linux 系统的启动和安装

- 3.1 用户帐号
- 3.2 登录 Linux 系统
- 3.3 Linux 命令和命令行编辑
- 3.4 网络在线帮助
- 3.5 网络在线文件
- 3.6 配置你的系统:用 Lisa 设置
- 3.7 安装软件包
- 3.8 远程通讯
- 3.9 使用调制解调器和 Internet 连接 ,pppd 和 ezppp
- 3.10 电子邮件:XFMail
- 3.11 X-Windows 和网络配置
- 3.12 总结:Linux 系统的启动和退出

第 4 章 窗口管理器和桌面:X-Windows

- 4.1 窗口、文件和程序管理器
- 4.2 启动和退出 X-Windows 窗口管理器和文件桌面

- 4.3 窗口和图标
- 4.4 fvwm 窗口管理器和桌面
- 4.5 Caldera 桌面
- 4.6 安装窗口管理器和文件管理器
- 4.7 Crisplite 和 XEmacs 编辑器
- 4.8 总结: X-Windows

第二部分 Linux 操作

第 5 章 Shell 操作

- 5.1 命令行
- 5.2 特殊字符和文件名参数: * , ?, []
- 5.3 标准输入/输入和重定向
- 5.4 管道: |
- 5.5 shell 变量
- 5.6 shell 脚本: 用户定义的命令
- 5.7 作业: 后台, 杀除 (kills) 和中断
- 5.8 总结: shell 操作

第 6 章 Linux 文件结构

- 6.1 Linux 文件

- 6.2 文件结构
- 6.3 列表、显示及打印文件:ls, cat, more 和 lpr
- 6.4 目录管理:mkdir, rmdir, ls, cd 和 pwd
- 6.5 文件和目录操作:find.cp.mv.rm 和 ln
- 6.6 总结:Linux 文件系统

第 7 章 文件管理操作

- 7.1 显示文件信息:ls -l
- 7.2 文件和目录许可:Chmod
- 7.3 文件系统:安装和卸载
- 7.4 网络文件系统:NFS 和 etc/exports
- 7.5 档案文件和设备:tar
- 7.6 文件压缩:gzip
- 7.7 从压缩的档案中安装软件:tar.gz
- 7.8 编译软件
- 7.9 mtools 实用工具:msdos
- 7.10 DOS 和 Window 仿真器:DOSEmu,Wine 和 Willow
- 7.11 总结:文件管理

第一部分 介绍 Linux

第 1 章 Linux 操作系统

Linux 是用于像 IBM 兼容机等带有 386,486 或奔腾微处理器的微机上的的一种操作系统,它也有针对 DEC Alpha 系统,Macintosh 系统,及 Sun 系统的多种版本。Linux 是在 90 年代早期,被 Linus Torvald 和其它遍布世界各地的编程人员共同开发的,作为一种操作系统,它完成了像 DOS 或 Windows 的同样功能。然而,Linux 更以灵活性及有效性而著称。大部分的微机操作系统,像 DOS,一直都是在受限的微机上开发的,仅最近才变得能在更多种类的机器上使用。像这样的操作系统需要不断地升级以适应微机硬件的发展,而 Linux 的开发却是不同的。Linux 是几十年来被用在主机及单机上的,现在已成为工作站首选系统的 Unix 的微机版本。Linux 使得 Unix 的快速,高效和灵活性能够用于你的微机,并能更好地利用个人电脑所提供的资源。

Linux 以一个好的价钱完成了一切,它是免费的。不像商用的 Unix 操作系统,

它是具体由 Free Software Foundation 在公共许可下免费分配的,任何人都可以免费来获得它。Linux 是有版权的,并不是一个公共的财产,然而,GNU 公共许可和公共财产几乎是同样的效果。这个使用许可使得 Linux 保持了免费,同时又具有标准化。仅有一种 Linux 标准。

Linux 是免费的事实有时给人一种误解:Linux 不完全是一个专业的操作系统,而事实上,Linux 是 Unix 的 PC 版本,正确地评价 Linux,你应该了解 Unix 操作系统发展的特殊环境,Unix 不像大部分其它操作系统。是在一个科研和学术的环境下开发出来。在大学和实验室里,Unix 是可选择的系统,在过去几十年里,Unix 的发展和整个计算机及通讯的革命相一致。计算机专业人员经常用 Unix 开发新的计算机科技。例如为 Internet 的开发。尽管它是一个高科技的系统,从一开始,Unix 就被设计成非常灵活的系统。Unix 系统本身可以被容易地修改产生不同的版本。事实上,许多厂商坚持 Unix 的商业版本,IBM,SUN 和 HP 都销售并坚持它们自己的 Unix 版本,满足他们自己的特殊需要。Unix 系统在设计上的可继承的灵活性从不减弱它的质量,事实上,正是它的多重版本,使它可以适应任何实用的环境,Linux 就是在这种情况下开发出来的,从这个意义上说,Linux 是 Unix 的另一版本-为 PC 的版本。由计算机专业人员在科研环境下的开发,反映了 Unix 版本通常被开发的方法,Linux 具有公共权限并且是免费的事实反映了在科研机构的开发人员对公共服务的热心支持。

作为对 Linux 的介绍,这章讨论了 Linux 作为一种操作系统,Linux 和 Unix 的历史和 Linux 的整个设计过程。这章也介绍了如何更好地使用本书。人们经常带着不同背景开始学 Linux,一些特点看上去十分熟悉,而另一些会有些不同,无论你的背景如何,本书通过严谨的结构提供给读者对 Linux 操作系统的清楚理解,从

而体现出 Linux 的各种不同特点。

1.1 操作系统和 Linux

一个操作系统是为用户管理硬件和软件的一个程序,操作系统最初是设计成执行重复管理硬件的任务。这些任务集中在文件管理,运行程序和从用户得到命令。你通过一个用户界面和操作系统相联系,这个用户界面允许操作系统接收来自用户的中断指令,你只需要发送一个指令给操作系统,就可完成像阅读一个文件或打印一份文件等的任务。一个操作系统的用户界面可以用只输入命令行一样简单,也可以用选择菜单和图标一样复杂。

一个操作系统也管理软件应用程序,去执行像编辑文件或计算等不同的任务,你需要专门的应用程序。一个编辑器是一个应用程序的例子。一个编辑器允许你去编辑一个文件,改变或增加新的文本。这个编辑器本身是一个由计算机可执行的指令所组成的程序。为了应用这个程序,必须把它装进内存,然后它的指令被执行。操作系统控制了所有程序的装入和执行,包括所有应用程序。当你打算用一个编辑器时,只要简单地指示操作系统去装入编程器应用程序并执行它。

文件管理,程序管理和用户交互是所有操作系统传统的共性。Linux 像所有 Unix 的版本一样,增加了两个特点,是一个多用户和多任务的系统。作为一个多任务系统,你可以要求系统同时执行多个任务,例如,当其它文件被打印时,你可以编辑另一个文件,你编辑之前,不用等打印工作被完成。作为一个多用户系统,几个用户可以同时注册在一个系统中,每个用户通过终端和系统相连。

操作系统最初是为有效地支持硬件而设计的,当计算机首先发展了,他们可能会受操作系统的限制。操作系统必须能实现计算机的绝大部分资源,所以操作系统是为硬件,并非为用户设计的,操作系统倾向于呆板,不灵活,使用户必须适应硬件效率的要求。

而 Linux,因为来源于 Unix,所以被设计的很灵活,做为 Unix 的一个版本, Linux 和 Unix 享有同样的灵活性。其灵活性来源于 Unix 的早期设计。在六十年代末和七十年代初 Unix 操作系统被工作在 AT&T 贝尔实验室的 Ken Thompson 开发出来。它结合了许多操作系统设计方面的新发展,开始,Unix 被设计成为研究人员的操作系统,一个主要目标是去产生一个能支持科研人员变换要求的系统。为了做到这一点,Thompson 必须设计能处理许多不同种类任务的操作系统,灵活性变得比硬件效率更重要。像 Unix 一样, Linux 已经能处理用户可能面临的各种任务。

这个灵活性允许 Linux 成为和用户相通的操作系统,用户不受与操作系统呆板联系的限制。取而代之,操作系统成为了提供用户能利用的一套高效工具,面向用户的科学含义是你能对系统配置或编程以满足你的特殊需要。用 Linux,操作系统变成了一种操作环境。

1.2 Linux 和 Unix 的历史

做为 Unix 的一个版本, Linux 的历史自然起始于 Unix,60 年代末期。关于新的操作系统技术有一个共同的努力。在 1968 年,一些来自通用电气公司,贝尔实

验室和麻省理工学院的研究人员开发一个名叫 Multics 的特殊操作系统的研究项目。Multicsd 在多任务文件管理和用户连接中综合了许多新概念。在 1970 年 AT&T 的贝尔实验室研究人员 Dennis Ritchie 和 Ken Thompson,在采用很多 Multics 特点的基础上开发了 Unix 操作系统。它满足了系统对科研环境的要求,使它运行在小型机上,从它的产生开始,Unix 是一个有价值的高效多用户和多任务的操作系统。

Unix 系统在贝尔实验室开始变得流行,并且越来越多的研究人员开始使用它,1973 年,Dennis Ritchie 和 Ken Thompson 一起为又用 C 语言为 Unix 编写了程序代码。Dennis Ritchie 贝尔实验室的后起之秀,开发了 C 编程语言作为程序开发的一种灵活工具,C 语言的优点之一是它能够通过一套编程命令直接访问一台计算机的硬件结构。到此为止,一个操作系统必须为每一类计算机上用硬件为基础的汇编语言,专门改写 C 语言允许 Dennis Ritchie 和 Ken Thompson 写了能用 C 编译器在不同计算机上编译的 Unix 操作系统的统一版本,在效率方面,Unix 操作系统变成了可移植的操作系统,能够不必重新编程运行在各种不同的计算机上。

Unix 从满足个人的需求设计开始,逐步成长为由许多不同开发商所支持的标准软件产品如:Novell 和 IBM,最初,Unix 被当成科研产品。第一个 Unix 版本是免费由许多知名大学的计算机系使用的。在 1972 年贝尔实验室开始发放商业版本并且给不同的用户授权使用这个系统。使用者之一是加州大学伯克莱分校的计算机系。伯克莱给系统增加了许多新的特点,后来成为了标准。1975 年伯克莱由下属部门 BSD 发行了自己的 Unix 版本。Unix 的 BSD 版本成为 AT&T 贝尔实验室版本的主要竞争者。而其它的被独立开发出的 Unix 版本也开始萌

生。

1980年微软公司开发了叫做 Xenix 的 Unix PC 版本。AT&T 几个 Unix 的科研版本 1983 年,它们发行了第一个商业版本。名叫 SYSTEM 3 后来被成为对商用软件产品重要支持 System v 所代替。

同时 Unix 的 BSD 版本不断发展,在 70 年代末期,BSD Unix 成为了国防部的高科技研究机构所使用的科研项目的基础,其结果,伯克莱发行了一个叫做 BSD Release 4.2 的有效版本。

它包括了高级的文件管理和基于 TCP/IP 网络协议的网络特点。现在,TCP/IP 被 Internet 所使用。BSD Release 4.2 被许多商所采用,例如 SUN Microsystems。

Unix 不同版本的出现导致了 Unix 标准的需要,软件开发商不知道他们和程序运行有哪个版本上比较适合,到 80 年代中期,两个竞争的标准出现了。一个是基于 AT&T 的 Unix 版本,另一个是 BSD 版本。在今天的书店里你能发现分别适用于这两个版本的不同 Unix 书,一些是 System V Unix,另一些集中在 BSD Unix。

AT&T 建立了一个叫 Unix 系统实验室的新组织,它的作用就是综合 Unix 的不同版本,集中开发一个标准系统,1991 年,Unix 系统实验室综合了所有 System V release 3, BSD release 4.3, SUN OS 和 Xenix 的所有特点。发行了 System V release 4. 为了与 System V release 4 竞争,一些其它公司如 IBM 和惠普建立了 Open Software Foundation(OSF) 去产生自己的 Unix 标准版本,继而出现了两个标准商业版本 OSF 版本和 System release 4。1993 年,AT&T 把它的 Unix 转卖给 NOVELL 公司。Unix 系统实验室成为了 Novell 的 Unix 系统小组的一部

分。Novell 发行了基于 System V release 4 的自己的 Unix 版本叫 Unixware,它可以和 Novell 公司的 Netware 系统相联。SUN 公司已经把 System V release 4 融进了它的 SUN 系统发行了 Solaris。两个相互竞争的 Unix 使用的图形用户界面,一个叫 Motif,另一个叫 Openlook 已经合并为一个新的工作平台标准,叫做共同工作平台环境(CDE)。通过不断发展,Unix 保留下来一个大的能有效运行在工作站和小型机上的操作系统。Unix 的一些版本主要被设计为工作站环境,SunOS 主要是为 SUN 工作站开发的。AIX 是为 IBM 工作站开发的,然而,由于 PC 变得更有效,因此开发 Unix 的 PC 版本就开始了。Xenix 和 System V/386 是为 IBM 及其兼容机而设计的 Unix 商业版本。AUX 是运行在 Macintosh 下的 Unix 版本。Unix 固有的可移植性使它几乎能在任何类型的计算机中被找到:工作站,小型机或者大型机。固有的可移植性使得产生 Unix 的 PC 版本成为可能。Linux 是专门为 Intel 为基础的个人计算机所设计的。它起源于荷兰赫尔辛基大学(HELSINKI)一个计算机系学生 Linus Toyvald 的个人研究项目。那时,学生们使用一个强调不同 Unix 特点的 Minix 程序,Minix 是由 Andrew Tannebaum 教授开发的,广泛分布在 Internet 上给全世界的学生使用。Linus 打算为 Minix 用户产生一个有效的 Unix PC 版本。他称它为 Linux。1991 年发行了 Linux 0.11 版本。Linux 广泛分布于 Internet,后来的几年,其它的编程人员综合现有的标准 Unix 系统中出现的大部分应用程序和特点修改并增加了它的内容。所有主要的窗口管理器已和 Linux 有了接口,Linux 有所有的 Internet 工具,像 ftp,telnet 和 SLIP,它也有整套的程序开发工具,象 C++编译器和调试器。虽然拥有全部特点,Linux 操作系统保持了小,稳定和快速。在最简单的格式下,它可以运行在仅 4MB 的内存。

尽管 Linux 是开发在免费的及 Internet 开放环境中,它仍具有商业 Unix 版本

的标准,因为前几十年里,Unix 版的大量出现,电子电气工程协会(IEEE)在美国标准化协会(ANSI)开发了一个独立的 Unix 标准,这个新的 ANSI Unix 标准被称做为计算机环境的可移植性操作系统界面(PSOIX)。这个标准限定了 Unix 系统如何进行操作,像系统调用有了专门的详述。PSOIX 限制所有 Unix 版本必须依赖大众标准,现有大部分 Unix 和流行版本都是顺从 POSIX,Linux 从一开始就是遵循 POSIX 开发出来的。

1.3 Linux 总览

像 Unix 一样,Linux 一般被分成四个主要部分:内核,Shell,文件结构和实用工具。内核是运行程序和管理像磁盘和打印机等硬件设备的核心程序。它从用户那里收到命令并把命令送给内核去执行。文件结构是文件放在磁盘等存贮设备上的组织方法。文件以目录的形式被组织每个目录可能包含许多子目录,其中有许多文件。

内核,Shell 和文件结构一起形成了基本的操作系统结构。使用这三个项目,你就可以运行程序,管理文件以及使用系统,另外,Linux 还有被称为实用工具的软件程序,实用工具被认为是一个系统的标准特征。实用工具是专门的程序,像编辑器编辑器交互程序,执行标准的计算操作等,你甚至可以产生自己的工具。

1.3.1 Shell: Bourne, Korn 和 C-Shell

Shell 提供了用户和内核的界面,它可被描绘为一个解释器,它解释由用户输入的命令并送他们到内核,内核界面是非常简单的,通常你输入一个命令然后按回车键就可完成,你输入命令的一行经常被指定为命令行,你将发现命令行能变得十分复杂。

代替命令行界面,Linux 提供了叫做 X-Windows 的图形用户界面(GUI)。它有很多供你使用的窗口管理器,窗口管理器的操作就像 Windows 和 Mac GUIs,你有窗口图标和菜单,所有的管理通过鼠标控制。两个流行的窗口管理器是 Free Virtual 窗口管理器(fvwm)和 Open Look 窗口管理器(olwm),Motif 窗口管理器也可获得,但 Motif 是一个有产权的程序,必须分开购买,另外窗口管理器,你必须使用文件和程序管理器,有各种文件和程序管理器是可获得的,例如 OpenLinux 提供了一个高级的文件管理器和程序管理器,可使用图标栏和目录夹。

尽管一个窗口管理器是一个灵活的管理界面,它真的只是一个 Shell 的界面,窗口管理可通过它收到的命令到达 Shell,实际是 Shell 解释这个命令并送它到内核。图 1-1 表明了 Shell 到内核和其它系统成员的关系。

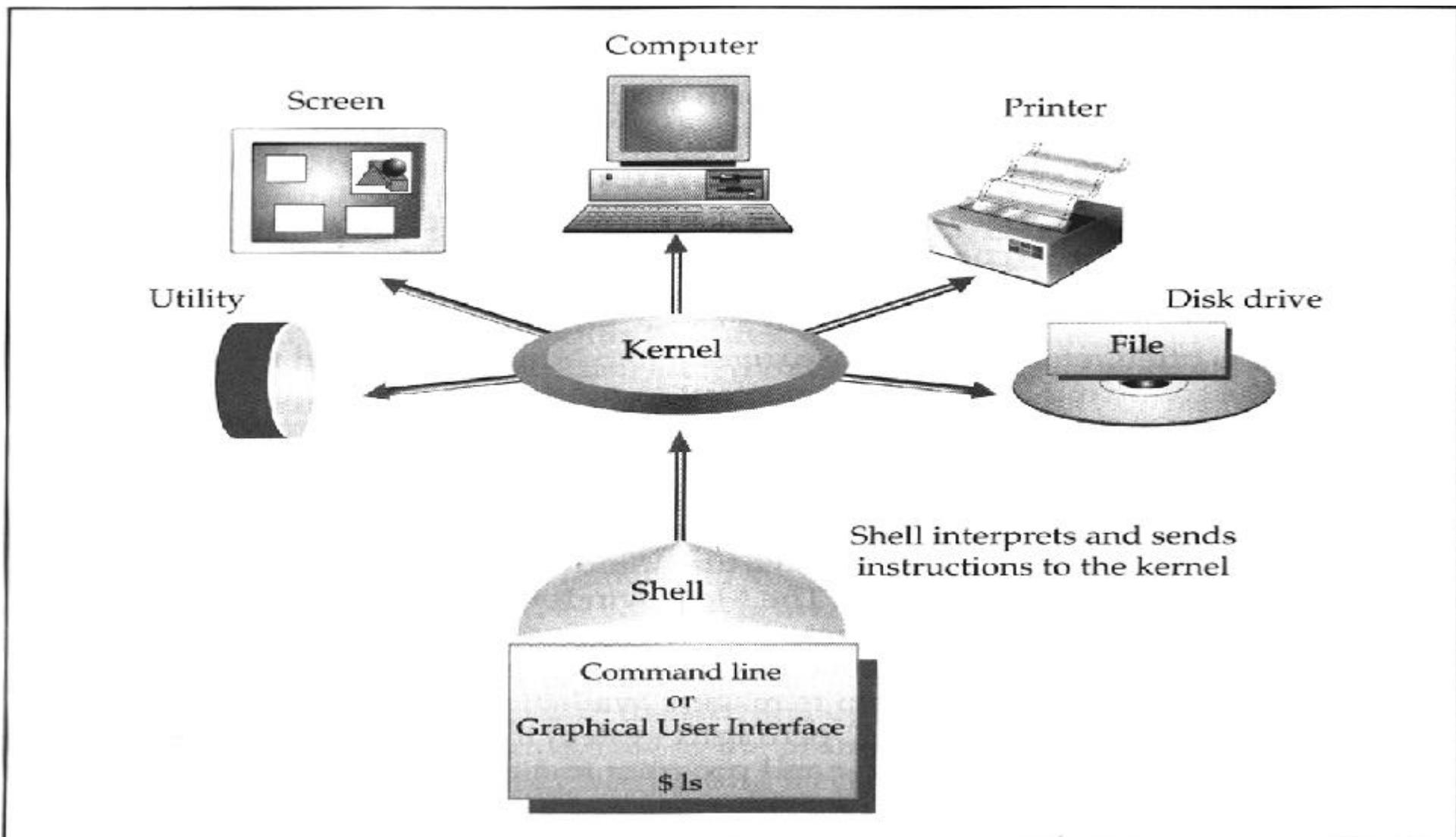


图 1-1 内核, Shell 和用户界面

Shell 不仅解释命令,它还提供了一个你可以配置和编程的环境。Shell 也有自己的编程语言,它允许你通过复杂的方法去编写 Linux 命令的程序。Shell

编程语言具有普通的编程语言很多特点,如循环结构和分支控制结构,你甚至可以产生和应用程序一样有效的 Shell 程序。

每个 Linux 系统的用户有他或她自己的用户界面或 Shell,能满足他们自己专门的 Shell 需要,从这个意义上说,一个用户的 Shell 不仅是一个用户控制的操作环境。

几年以来,几种不同的 Shell 被开发出来,当前有三种主要的 Shell: Bourne, Korn, 和 C-Shell, 其中 Bourne Shell 是贝尔实验室开发的, C-Shell 是 SUN 公司 Shell 的 BSD 版本, Korn shell 是 Bourne Shell 的进一步增强。现在的 Unix 版本,包括 Linux 在内,都包含了三个 Shell,允许你去选择所喜欢的 Shell, Linux 使用这些增强的公共版本。 Bourne Again Shell, TC-Shell 和 Public Domain Korn Shell。当你开始 Linux 系统时,系统设置在 Bourne Again Shell 下,从那里,你能够转换到其他 Shell。

1.3.2 文件结构:目录和文件

在 Linux 系统中,像 DOS 一样,文件由目录结构来组织。然而,不像 DOS 的是,你有更多的灵活性去控制你的文件,整个 Linux 文件系统是一套大的相互关联的目录。每个目录里面都包含文件,一些目录是系统保留的标准目录,你能为自己的文件产生自己的目录,也可从容易地把一个目录下的文件移至另一目录下,你甚至能移动整个目录并和系统中的其他用户共享目录和文件。用 Linux,你可以设置目录和文件的权限,以便允许和拒绝其他人访问他们。

实际上,每个用户的目录最终和其他用户的目录相连。目录是以根目录开始

被组织的等级树形结构。所有其他的目录最终是由第一个根目录派生出来。图 1-2 表面了一个树形等级结构的例子。你可以浏览整个系统,进入任何可以对你开放的目录。文件结构的相互关联性使分享数据变得容易,几个用户可以访问同一个文件。

根目录是一个当你第一次设置你的 Linux 系统时所使用的特殊目录 Linux 是一个多用户系统。操作系统本身的驻留程序存放在以根目录开始的专用目录中,有时被指定为系统目录,在图 1-2 中那些根目录下的目录就是系统目录:man,bin 和 user。其他的系统目录将在第 5 章中讨论。

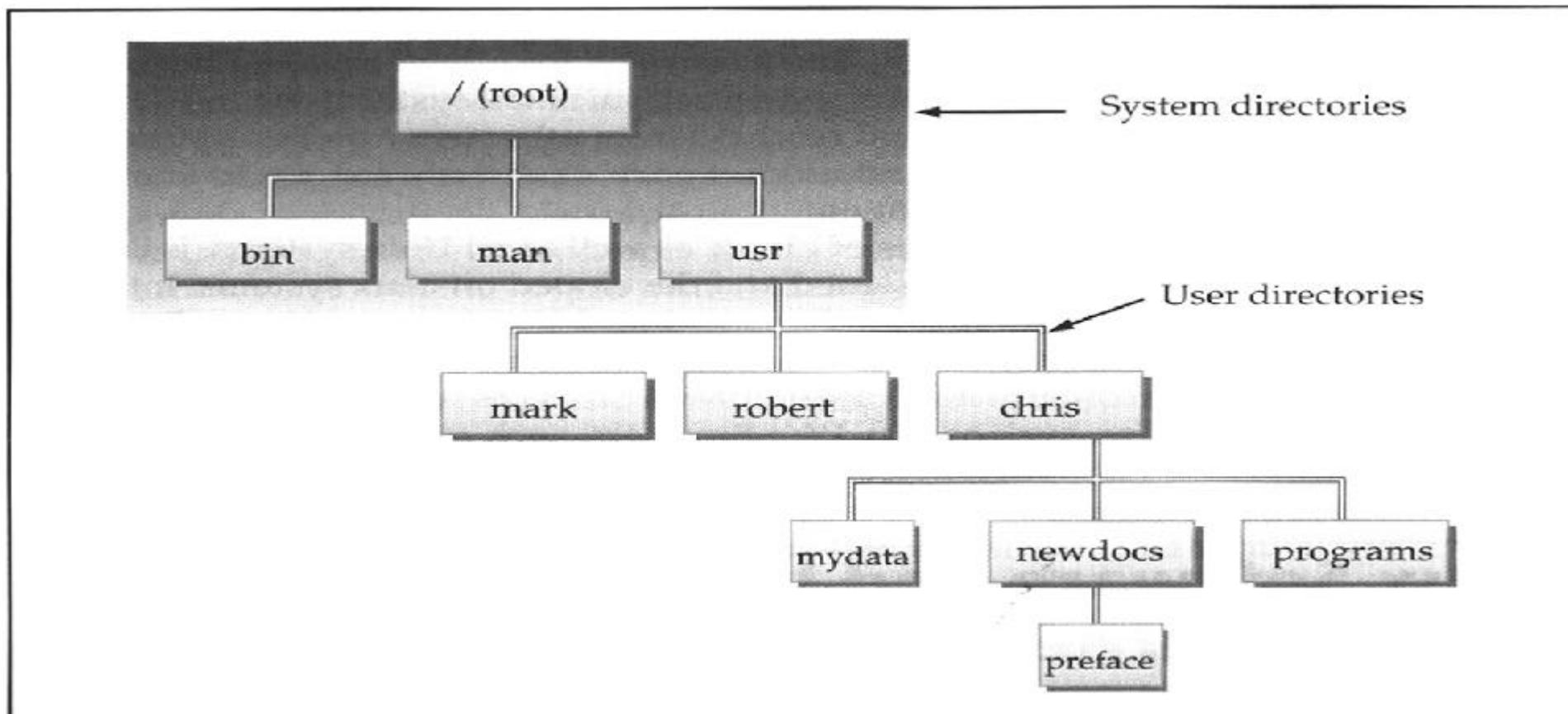


图 1-2Linux 的文件和目录结构

1.3.3 实用工具:编辑器、过滤器和交互程序

Linux 包含大量的工具。一些工具执行简单的操作,另一些是带有他们自己整套命令的复杂程序,从一开始这些使用工具被分为三类:编辑器,过滤器,和交互程序。当然不是所有的工具适合这三类。有的工具是执行文件操作和程序管理。

所有的 Unix 版本中都有有一些标准编辑器可获得。包括:Ed,Ex,Vi 和 Emacs。Ed 和 Ex 是行编辑器,Vi 和 Emacs 是全屏目编辑器。所有的标准编辑器都是由早期不能处理像今天字处理程序中光标移动的机器中发展而来。尽管 Vi 和 Emacs 有全屏编辑能力,但他们不能像由鼠标驱动的字处理器一样容易使用。Ed 和 Ex 是每次只能显示和编辑一行的行编辑器。尽管有这些限制,但编辑器的效率是很高的,他有大量的命令和复杂的操作相结合。

另一套工具最好被描述为过滤器(filter),一个过滤器读取从用户文件或其他地方的输入,检查和处理数据然后输出结果。从这个意义上说,他们过滤经过他们的数据有不同类型的过滤器,一些用行编辑命令输出一个被编辑的文件。另外的一个是模式寻找文件并以这种模式输出部分数据。还有一些执行字处理操作,检测一个文件中的格式的命令输出一个格式化的文件,过滤器的输入可以不是一个文件,它可以是用户从键盘敲入的数据,也可以是另一个过滤器的输出,过滤器能相互连接,因此一个过滤器的输出能成为另一个过滤器的输入,甚至你可以编写自己的过滤器程序,一个过滤器编程语言叫做 awk,能使你编写自己的过滤器。Linux 也有一套允许你和本系统或其他系统用户相交互的工具,作为一个多用户系统 Linux 必须和所有用户保持联系。显示的基本要求和文件系统的基本互联性使得电子邮件系统容易实现。信息可由系统上的不同用户发送和接收,你甚至可以一次对多个用户同时广播一个住处或直接与另一个相连进行一个实时对话。

Linux 和 Unix 系统最重要的特点之一是它的多套 Internet 工具,Internet 是在 Unix 系统的基础上设计和开发出来的,像 ftp 和 telnet 等 Internet 工具最先实现于 Unix 和 BSD 版本。Darpanet 是 Internet 的前身,用来联接美国国内各大学的 Unix 系统,Linux 包含了全套的 Internet 工具和他们提供了与 Internet 的直

接相连像 SLIP 网络浏览器像 Netscape 都在 Linux 可获得的。Internet 上的 Linux 软件和信息资源 Linux 是在 Internet 上通过共同合作开发出来的,没有一家公司和机构控制 Linux。

1.4 Internet 上的 Linux 软件及信息源

当 Linux 用户决定共同完成一个项目的开发就开始进行,当完成以后,软件由 Internet 上的一个 ftp 地址来发布。任何 Linux 用户可以到达这个网址来下载软件。

软件是你需要编译的源代码形式或执行的二进制代码。软件包通常包括文件和配置文件,OpenLinux 使用叫做 Redht Package Manager(RPM)的专门安装工具。你能通过一个简单的命令 `rpm -i` 和所有的文件和配置文件去安装 RPM 软件包(详见第 3 章)。大部分当前的软件可在 RPM 软件包中获得。在 `ftp.redhat.com/pub/contrib/i386` 中可发现大量的 RPM 软件包。在这儿,你应该首先核对你想使用的软件的 RPM 版本,对于 Linux 软件和文件的主要 ftp 地址之一是 `/pub/Linux` 目录下的 `Sunsite.unc.edu`。Homesites 是 Linux 软件最初被邮寄的地方 Mirror Sites 是这些 Homesites 的复制。`ftp.caldera.com` 和 `ftp.redhot.com` 包含了 Linux 文件的扩展档案。表 1-1 列出了不同的 ftp 位置,通过与网址相联,你可以获得其他的地址。例如 Linux 在 World Wide Wed 中是在 `www.sss.com/linux/resoures/web.html`。其中特别有趣的是 Linux Game Tome 中列出了为 Linux 的大部分流行游戏。在第 3 章 3-2 表中列出了几个资源网页。

大部分 LINUX 软件具有 Free Software Foundation 支持的 GNU 公共权限下版权,经常被指定为 GNU 软件,GNU 软件已经证明是可信的且是有效的。许多 Linux 使用的工具,像 C 编译器,Shell 和编辑器都是 GNU 软件的应用程序。你将发现你的 OpenLinux 系统上安装 GNU C++和 Lisp 编译器,Vi和 Emacs 编辑器 Bash 和 Tcsh Shell 以及 Tex 和 Ghostscript 文件格式器,许多其他的 GNU 软件应用程度也可在 Internet 的不同 ftp 地址上获得。表 1-2,详细描述了从 Internet 上下载软件应用程序及安装他们的过程。

近来,主要的软件公司,也开始开发他们的最流行应用程序的 linux 版本,你可以从 Internet 的 ftp 地址:ftp.caldera.com 来下载它,SUN 公司的 Java Linux 版本也可从 ftp.Blackcown.org 上获得一个 Wordprefect 的 Linux 版本也可获得。

Linux 支持资料也在 Internet 上开发的,目前大部分支持资料可以从 Internet 的 ftp 地址处下载,一个叫 Linux Documentation Project 的 Linux 项目由 Matt Welsh 带领,正在开发一套 Linux 说明,这个资料可由以下 URL 获得:<http://sunsite.unc.edu/mdw> 这份支持资料包括了用户指导介绍和管理指导,它们可以用超文本或文本的格式获得。表 1-3 列出了这些指导,你能发现对于什么是 How-To 文件的简单解释,How-To 文件包括如安装打印和电子邮件等多个方面。这个文件通常可从/pub/linux/doc/how-to 目录下的 Linux ftp 地址处获得。

在/pub/linux/doc目录下你能发现 Meta-fao 文件包含不同的 Linux 信息位置的列表,在相同的位置和目录,你也可以下载 Linux 软件图(LSM),这是当前为 Linux 所使用的大部分软件的列表。许多软件公司也有提供他们 Linux 应用程序的网址,这些被列在表 1-4 中。

除了 ftp 网址,也有 Linux Usenet newsgroups.通过互连网,你能登录 Linux

Newsgroups 去阅读其他用户的内容并邮寄你的信息。有很多的 Linux Newsgroups,都是以 comp.os.linux 开始的,特别能兴起初学者兴趣的是 com.os.linux.help,从那里你可以发出问题,表 1-5 列出了在 Usenet 上的不同 Linux Newsgroups。

大部分当前的标准 Linux 软件和文件已经包含在你的 OpenLinux Lite CD-ROM 中。How-To 文件都是以 HTML 格式可获得的,所以你可以用网浏览器去看他们,但是,在将来,你可能必须直接到达 Linux 和 Internet 网址去获得最新的信息。

当前,唯一处理 Linux 的出版物是 Linux Journal,你可以从下面的网址获取关于它的信息:www.sss.com/lj。也可以联系以下地址:

Linux Journal
P.O. BOX 85867
Seattle WA 98145
206-527-3385

1.5 Linux 发行版本

尽管有唯一的 Linux 标准版本,但有几个不同的发行公司用略有不同的方法 Linux 包装和 Linux 软件,每个公司然后用 CD-ROM 发行 Linux 软件包,后来发行的可能包含程序的最新版本和新软件,其中一些是 Slackware,Redhat,TAMU,Yggdrasil 和 Infomagic。Open Linux Lite CD-Rom

包含 Caldera 的完整 Open Linux Linux 系统。一些供应部门,像 Caldera 和 Redhat 也用像字处理的商业软件或像 Netscape 的商用 Internet 服务器包在它们的系统中。

1.6 OpenLinux

OpenLinux Linux Lite 光盘包含了 Caldera 的完全 OpenLinux Linux 系统,并包含所有 GNU 软件包的软件包以及 X-Windows 窗口管理器 Internet 工具,它是和 POSIX 相一致的 Caldera 提供它的免费的 OpenLinux 系统,光盘被指定为 "lite",因为 Caldera 也提供一些商用的,有产权的软件包,这些软件包包括一些像 Wordperfect, netscape 服务器和 Novell Netware 用户等产品,因此,不是免费的.详见 caldera 的网址 www.caldera.com。

1.7 总结

Linux 是由 Linus Torvald 首先开发的,为 386,486 或奔腾处理器的 PC 系列使用的一个 Unix 操作系统的版本,Linux 是以它的高效性和灵活性而著称,带给你的 PC 全部的 Unix 特点,包括多任务,多用户的能力,Linux 是在 GNU 公共权限下免费获得的,它是一个符合 POSIX 标准的操作系统。

你的 Linux 软件包不仅 Linux 操作系统,而且还包括了像编译器编辑器等的应用软件。它也包括带有几个窗口管理器的 X-Windows 图形用户界面(GUI)允许你用窗口,图标和菜单与系统相互交流。

Linux 系统由内核,Shell 文件结构和实用工具组成,内核是系统的核心,它包含直接管理计算机硬件的控制程序,Shell 是用户界面,Shell 收到来自用户的命令并把它送入内核去执行。Shell 能满足个别用户的需要。Shell 甚至有自己的编程语言用于对命令的编辑。文件结构包括放置文件的目录,目录提供管理文件的方便途径,你能从一个目录移到另一个目录并且设置权限和其他用户共享其中的文件。

标准的 Linux 系统都有一套叫做实用工具的程序,实用工具可分三类:编辑器,过滤器,或交互程序,很多编辑器虽然效率很高,但使用起来下如现有 PC 编辑器一样容易,过滤器是接收数据并产生数据被修改格式的专用工具。交互工具允许你发送信息或收到来自其他用户的信息。usenet 和 web 等 Internet 具允许你容易地实现信息服务。

表 1-1 Linux ftp 网址

Linux ftp 网址	目录
sunsite.unc.edu	/pub/linux
tsx-11.mit.edu	/pub/linux
nic.funet.fi	/pub/os/linux
Mirror.Sites	
ftp.mcc.ac.uk	/pub/linuxBritain
ftp.ibp.fr	/pub/linuxFrance
ftp.dfv.rwth-aachen.de	/pub/linuxGermany
kirk.bu.oz.au	/pub/linuxAustralia
ftp.uu.net	/systems/unix/linux
ftp.stack.nl	/pub/linux
ftp.caldera.com	/pub/linux
ftp.redhat.com	/pub
ftp.cdrom.com	/pub/linux/Concord,CA
ftp.siriuscc.com	/pub/linux/Dallas,TX
lss.afit.af.mil	/pub/linux/Dayton,OH
ftp.engr.uark.edu	/pub/linux/Fayetteville,AR
ftp.spin.ad.jp	/pub/linux/sunsite.unc.eduTokyo

续表

ftp.funet.fi	/pub/linux/
ftp.ba-mannheim.de	/pub/linux/Mannheim
ftp.nuri.net	/pub/linux/Seoul

表 1-2Linux 的应用程序

网 址	应 用 程 序
ftp://ftpredhat.com/ pub/contrib/i386	Linux applications in RPM format(shareware)
ftp://ftp.ncsa.edu	Mosaic Web Browser and Web Server software for Linux
ftp://ftp4.netscape.com	Netscape Web Browser for Linux
ftp://ftp.caldera.com	
ftp://ftp.blackdown.org	Sun's Java Development Kit for Linux(link to mirror sites)
http://www.redhat.com	Applixware Office Suite
http://www.sdcorp.com	WordPerfect 7
wplinux.htm	
http://www.caldera.com	Netscape Web Browser for Linux,Caldera Office Suite,StarOffice
http://sunscript.sun.com	Tk/Tcl Products
http://www.uk.linux.org/	Linux Commercial Vendors InderLxCommercial.html
http://www.xnet.com/ ~blatura/linapps.shtml	Linux Applications and Utilites Page

表 1-3Linux 的参考资料

网 址	描 述
http://sunsite.unc.edu/mdw	LDP Web site
ftp://wunsite.unc.edu/pub/	LDP ftp site
Linux/docs/linux-doc-project	
http://www.iki.fi/liw/linux/sag/	System Administrators' Guide Homepage
ftp://sunsite.unc.edu/pub	LDP HOW-TO documents
Linux/docs/HOWTO	
Guides	Document Format and Web Sites
Linux Installation and Getting Started Guide	DVI, PostScript, and HTML(online)
Linux User's Guide	DVI, PostScript, and PDF
Linux System Administrator's Guide	DVI, PostScript, and HTML(online)
Linux Network Adiminstrator's Guide	DVI, PostScript, and HTML(online)
Linux Programmer's Guide	linuxwww.db.erau.edu/LPG (link through LDP Web site), DVI, PostScript, and HTML(online)

续表

The Linux Kernel	DVI and PostScript
Linux Kernel Hacker's Guide	www.redhat.com/(link through LDP Web site),DVI,PostScript,and HTML(online)

表 1-4 Web 网址

网 址	描 述
www.ssc.com/linux	Linux Resources Web site
www.caldera.com	Caldera Web site
www.redhat.com	Redhat Web site
sunsite.unc.edu/mdw	Web site for Linux DocumentationProject
www.ssc.com/lj	Linux Journal
www.linux.org	Linux Organization
www.li.org	Linux International Web site
www.uk.linux.org	Linux European Web site
www.blackdown.org	Web site for Linux Java
www.netscape.com	Netscape Web site
www.fokus.gmd.de/linux	Wover Goods for Linux
www.ssc.com/linux/resources/web.html	Linux on the World Wide Web

表 1-5 Usenet Newsgroups

新闻组	描述
comp.os.linux.announce	Announcements of Linux developments
comp.os.linux.development.apps	For programmers developing Linux applications
comp.os.linux.development.system	For programmers working on the Linux operating system
comp.os.linux.hardware	Linux hardware specifications
comp.os.linux.admin	System administration questions
comp.os.linux.misc	Special questions and issues
comp.os.linux.setup	Installation problems
comp.os.linux.answers	Answers to command problems
comp.os.linux.help	Questions and answers for particular problems
comp.os.linux.networking	Linux network questions and issues

第 2 章 安装 Linux

本章描述了从 CD-ROM 安装 OpenLinux lite 系统的安装过程。这个安装包包括了 Linux 操作系统、大量的 Linux 应用程序、一整套 Internet 服务器。不同的 Linux 软件包通常有他们自己的安装程序。例如, Linux 的 Slackware 软件包部分在 OpenLinux 下有不同的安装程序。Caldera Open Linux 安装程序是简洁有效的。具体的方面,像 web 服务器,一般要求专业的很复杂的配置操作。OpenLinux 自动安装并配置许多文件。

安装 Linux 包括几个步骤:首先,你必须决定是否你的计算机满足硬件要求。然后你要寻找你使用的硬件方面的具体类型。例如你的显示器类型和显示卡类型。这些信息可以从硬件的说明书上获得。

如果你打算让 Linux 系统和其它系统共用一张硬盘,你必须给你的硬盘分区,怎样分区你的硬盘有不同的选择方式,主要依靠是否它包含你要保护的数据。

你开始安装程序的时候需要准备一张启动盘,一旦安装程序开始,你只要简单地跟随屏幕的指示,大部分时间你只要做一些是或否的选择就可以了。安装程序要经过几个阶段。首先你要在你的硬盘上产生 Linux 分区。然后你安装软件包,接下来能配置网络联接,图形用户界面(GUI)支持的 X-Windows 服务器。X-Windows 和网络配置可以在以后的时间独立执行。

一旦你的系统安装结束,你就可以准备开始和注册帐号。你将通过注册命令

来注册,从命令行你可以激活 X-Windows 安装提供给你一个完全的图形用户界面(GUI)。

你可以选择只安装操作系统,还是带有标准应用程序的系统或者在 CD-ROM 上全部软件系统。如果你选择了标准安装,以后你可以增加未安装的软件包。第 3 章描述了你怎样使用 glint 工具和用 Redhat Package Manager(RPM)去安装或卸载软件包。

2.1 硬件,软件各信息的要求

安装 Linux 以前,你必须确信你的计算机满足最小硬件要求。你还需要具体的软件硬件分区程序。所有 DOS 系统的标准也存在于你的 OpenLinux 光盘上。对于你的显示器,显示卡,鼠标,光驱等,你要知道确定的具体信息。下面的部分将详细描述所有要求,安装以前要仔细阅读他们。安装期间,你必须对基于你的计算机配置做出选择。

2.1.1 硬件要求

下面是安装 Linux 的最小系统硬件要求。

- 一个 32 位的 Intel 为基础的个人计算机。要求一个 Intel 或兼容机,386,486 或奔腾微处理器 SX 和 DX CPU 被接受。

- 一个 3.5 寸软驱。
- 至少 8MB 内存,推荐使用 16MB。
- 至少 300MB 的硬盘空间,推荐使用 300MB ~ 400MB,如果要装入并使用 CD-ROM 上的所有软件包,至少需要 750MB,基本软件包的标准安装要占用 212MB,还要加上 16 到 64MB 的交换空间,你可以选择最小安装,只安装 LINUX 的内核和少量应用程序,以后你可以每次加一个应用程序。
- 一张 3.5 寸 DOS 格式化的高密软盘,以便产生一个启动盘。
- 一个 CD-ROM 驱动器。
- 两张 3.5 寸 DOS 格式化高密软盘。

如果你打算使用 X-Windows 的图形用户界面,你还必须有。

- 一个显示卡。
- 一个鼠标或其它定位设备。

2.1.2 软件要求

软件的要求是很少的,基本上你需要一个能产生 Linux 启动盘的系统,如果你打算使 DOS 和 Linux 运行在同一张硬盘上,你需要 DOS 分区软件,以此类推,如果你有 OS/2,你就需要 OS/2 分区软件。

为了你能准备安装盘,操作系统是必要的,用 DOS 系统你可以访问 OpenLinux CD-ROM,并且发出像 DOS 的命令去产生安装盘,任何 DOS 的类型都可以,甚至你可以在 OS/2 上使用相同的命令。然而,运行 Linux 时不需要 DOS,它是一个独立的操作系统。如果你想让 Linux 和其它操作系统共享硬盘,例如

DOS,你将需要为分享硬盘的具体工具。对 DOS 来说,你需要 defrag 和 fips 工具或者 fdisk 工具,fips 工具提供在你的 OpenLinux CD-ROM 里,它可以执行一个非破坏性硬盘分区,为 Linux 系统扩展空间。defrag 和 fdisk 是标准的 DOS 工具,一般挂在 DOS 目录下,defrag 和 fips 是在 Fips 分区前整理磁盘碎片,当然,有一些其它的商用工具可以整理磁盘碎片。fdisk 是代替 fips 的工具,它执行破坏性硬盘分区,删除硬盘上的所有文件,并要求一个完全的格式化的 DOS 安装。

2.1.3 信息要求

PC 采用 Linux 这样有效的操作系统能更好地使用计算机的硬件设备,为了做到这一点,Linux 要求计算机各部件的具体信息。例如,特殊的 Linux 配置文件能够满足显示卡及显示器、具体生产厂家和型号的要求。安装 Linux 以前。手边需要有这方面的信息,这些信息一般可以从硬件或计算机的说明书上获得。

CD-ROM、硬盘和鼠标信息

如果你有一个 SCSI 的光盘驱动器,你需要制造厂商的名字和型号。如果你有一个 IDE 的光盘驱动器,也要注意这些事实。

决定在你的硬盘上给你的 Linux 系统多少空间,如果你打算和 DOS 共享硬盘,决定 DOS 和 Linux 各占多少?

决定你要给交换区多少空间,你的交换区必须在 16MB 的 64MB 之间,对于大部分系统,32MB 是恰当的,它被 Linux 用来扩展你的计算机内存。

发现你的鼠标厂家和型号,Linux 支持串口和总线鼠标,大部分的鼠标被支持,包括微软 Logitech 和 Mouse 系统。

了解你所在的时区。如果你的计算机直接连在 Internet 上,你可能用格林威治时间(GMT)作为你的时区,然后,如果你是用一台单机且和 DOS 共用,你可能想用当地的时区。

了解你的鼠标用在哪个串行端口,COM1,COM2,或用 PS/2 鼠标端口。

显示卡和显示器信息

注意下面关于显示卡和显示器的信息,X-Windows 已为大部分显示卡设置好了,所以你可以不必使用显示卡信息。但是,你绝对要用到显示器信息,包括垂直和水平的、刷新频率。显示或信息:

- 你的显示卡生产厂家及型号。
- 你的显示卡用什么系列芯片集。
- 你的显示卡上存贮器有多大。

显示器信息:

· 了解你的显示器的制造厂商及型号。Linux 支持几种显示器,如果你用的不属于他们,要选择一个 GENERIC PROFILE。在一个习惯的 PROFILE 文件中键入信息。为了做到这一点,你需要下面信息。

- 水平的刷新频率是多少
- 垂直的刷新频率是多少

网络配置信息

除决定你的主计算机名称外,安装期间你不必去配置你的网络。你可以在以后用 netcfg 工具去完成网络配置,但是如果信息已准备好,安装过程将自动配置你的网络,在恰当的配置文件中放置必要入口。如果你正有一个 Internet 服务提供商,他们将提供你大部分信息,如果你亲自设置网络,你将必须决定每个信

息。安装程序将提示你考虑以下几个方面：

- 决定你的计算机的名字,Internet将通过此名识别你的计算机。
- 你的域名。
- 分配给你的计算机的 IP 地址,Internet上的每个主机被分配一个 IP 地址,是用点号分开的四位数字。做为一个区域内对计算机的识别,允许从来自其它区域的信息访问这台计算机。
- 你的网络 IP 地址,通常与 IP 地址相一致,但增加一个 0。
- 子网屏蔽,通常是 255,255,255,0。但如果你是一个大网络中的一部分,要与你的管理员来核实。
- 网络广播地址,通常你的广播地址与 IP 地址同名,只是在末尾加上 255。
- 如果你有一个网关,你需要为你的网络分配网关的 IP 地址。
- 网络使用的任何服务器的 IP 地址。
- 如果你的网络使用一个 NIS 服务器,要选择 NIS 域和 IP 地址。

2.1.4 已安装 LINUX 系统的升级信息

如果你已经安装了像 Caldera network desktop 或 Redhat 等另一个 LINUX 版本,你可能保留你喜欢的不同设置来使你的系统更加个性化。这些设置可以放在你的配置文件里并存入一张软盘,再用到你的新的 OpenLinux 系统。通过这个方法,可以保留你的最初配置(如果你用 mcopy,要记住用 -t 选项)。你也可能要保留一些像网页等目录和数据文件或者你已经下载的软件包。用下面的 tar 操作将是最好的方法。

```
tar cvMf/dev/fd0 directory-or-package
```

的 拷贝下面你打算恢复的配置文件和其它文件。你反需要拷贝那些你想恢复

文件	描述
/etc/XF86Config	X-Windows 配置文件
/etc/lilo.conf	启动管理器配置文件
/etc/hosts	联接系统的 IP 地址
/etc/resolv.conf	域名服务器地址
/etc/fstab	安装在系统上的文件系统
/etc/passwd	系统上用户的名字和口令
/home/user	在系统中,用户的 home 目录,user 将是用户的名字(对于大量的文件,要使 用 tar cfM/dev/fd0/home/user)
.netscape	每个 home 目录有自己的带有 netscape 配置文件的 .netscape 子目录。
web 网页和 ftp 文件	你可能要存储一些为 web site 使用的网页和 ftp site 的文件在用 CND 版本中,这些放在 /home/httpd 和 /home/ftpd 下。

一旦你已经安装了 OpenLinux 你就能安装软盘,并从上面给你的系统复制文件,改变那些最初的设置。如果你使用先前系统中的 /etc/xF86 config 文件,你就不用运行 xF86setup 去设置 X-Windows。/etc/xF86 config 文件包括了全部 X-WINDOWS 的设置信息。如果你想从你先前的系统中恢复 /etc/lilo.conf 文件,

就必须用下面的命令来安装它：

```
# lilo/etc/lilo.conf
```

为了恢复存在多个磁盘上的文档,要用 TAR 操作,在软驱里放入第一张盘,然后用下面的命令。

```
tar xumf/dev/fd0
```

2.2 为 Linux 分区开辟磁盘空间

如果你打算把 Linux 运行在你的整张硬盘或用同一个分区升级一个 Linux 系统,你可以跳过这部分直接看 Linux 的安装。但是,如果你要使 Linux 和 DOS 共享一张硬盘,你将必须分区并格式化你的硬盘,以便使它一部分给 DOS 系统,另一部分给 Linux 系统。你如何完成这一过程依靠你当前硬盘的状态。如果你有一个的硬盘,你就需要确信把 DOS 安装在硬盘的一部分上,另一部分留给 Linux。如果你已经在使用这张硬盘,你也必须删除已经存在的分区并对硬盘重新分区。留下空余的给 Linux,每个情况的目的都是为 Linux 开辟自由空间,当安装 Linux 时,你必须为 Linux 分区和格式化自由的硬盘空间。

一个硬盘被组织成各个分区。这个分区再被给定的操作系统进行进一步的格式化。当你安装 DOS 的时候,你首先需要把你的硬盘分成不同的区,然后用 DOS 格式化命令去格式化每个分区形成 DOS 盘,分别由一个字母来识别。例如,你可能分你的硬盘为两个区。一个作为 C 盘,另一个作为 D 盘,你也可以把你的硬盘只分成一个分区,格式化成 C 盘。为了和 Linux 分享硬盘,你必须减少他们的大

小或者删除一些分区来开辟一些自由空间。

首先,要决定你打算给你的 Linux 系统分配多少空间。你最少需要 300MB。像前面所述的那样,基本的 Linux 软件包占据 212MB,而整套件包,包括所有的源代码文件占据 667MB。另外,你必须为 Linux 留出交换区,用于实际的存贮单元,这要求 16 到 32MB 之间用 300MB 字节,如果你需要安装其它的软件包时,你很快会发现空间是太有限了。一个实用的最小空间应是 400MB 到 500MB,一旦你决定了你的 LINUX 系统所用的空间,你就可以为硬盘开辟自由空间,根据不同的硬盘分区,开辟空间会采取稍有不同的步骤。如果你已经有几个分区,你可能反需要删除他们中的几个。

先了解什么是最适合你的选择,你应首先决定你的分区是什么及它们的大小。你能用 FDISK 工具去做这一点。开始这个工具,在 DOS 提示符下键入 FDISK,然后按回车。

```
C: />fdisk
```

然后会出现一个 fdisk 选项的菜单,选择第 4 页去显示当前所有的分区及他们的大小,你可以减少或再分区你的硬盘,或者你可以删一些无用的分区。

如果你已经有一个大的区覆盖了整个硬盘,你有两个选择。你可以用 DOS 的 defrag 和 Linux 的 fips 工具在你硬盘的无用空间上非破坏地产生自由空间。为了安全,你应该对重要的数据做备份,但不必再安装 DOS 和恢复你的数据,每件东西都被保护。另一个选择,你可以再分区你的硬盘,删除所有的数据,然后你必须再安装 DOS,从你的备份中恢复数据。也要再格式化你的硬盘。如果你想把 DOS 和 Linux 安装在同一张新的硬盘上时,你将跟随下面对你的硬盘进行分区的相同步骤。

2.2.1 删除分区

在大多数情况下,一个用户在一张硬盘上会有几分区。如果你有一个数据非常少的 DOS 分区。你可以考虑删除他们,为 Linux 系统开辟自由空间。在这种情况下,只要决定你想删除哪一个及多少空间就行了。删除一个分区将删掉上面的所有数据,因此,首先要确定对数据做备份。

记住,DOS 分区和 Linux 分区有着严格的不同,如果在你的硬盘上已经有一个 Linux 系统,你打算删除这个分区安装一个新系统,你必须用 Linux 系统的 fdisk 工具,而不是 DOS 的 fdisk 工具,一个 Linux 分区不能由 Linux 的 fdisk 工具完全删除。

```
C: />fdisk
```

从删除菜单,选择 "Delete partition or logical dos drive",这样列出当前的驱动器并提示你键入分区的号码,每个分区入口将列出类型,卷标,它所占用的字节数,不要删除类型是 PRI DOS 的分区,它通常是第一个分区。一个安全的建议是从最后一个分区开始。

键入分区号以后,你将被提示键入卷标(卷标列在 fdisk 菜单中的分区入口里)。当你确定后,键字 Y 去除分区,如果你回过头选择第 4 选项。"Display partition information"你将看到这个分区已不存在。取代它的是一个自由空间的入口。它告诉你 Linux 可以使用多少自由空间。

如果你需要更多的空间,你可以重复先前的步骤,再删除一个分区,每删除一个分区,硬盘就增加了自由空间。如果删除的分区对于 Linux 系统来说又太多了,你可以用 fdisk 菜单的增加选项增加一个新的较小的 DOS 分区。

去增加一个分区,在 fdisk 菜单中选择"分区创建"选项,这会显示分区创建菜单,选择第 2 选项后,将被提示分区的大小,增加分区后,核对 fdisk 选项菜单中的第 4 项。最后的选项将显示自由空间的数量。确认为你的 Linux 系统留下足够的空间。

2.2.2 非破坏性再分区

如果在你的微机上有大量不想重新安装的数据和程序,你可以用 defrag 和 fips 实用工具来进行非破坏性再分区。但是,为了安全起见,你应该对你的数据做备份。执行非破坏性再分区,你首先了解是否有足够的剩余空间为 Linux 使用,如果没有,就必须删除一些文件。你能用 chkdsk 来核对你的硬盘有多少空间可用,看是否有必要增加更多的空间,如果需要增加,删除一些不需要的文件再次重复检查,直到有了足够的空间。

当 DOS 创建和存贮文件时,它放置这些文件在硬盘上的不同部分,你的文件是散布于硬盘上。它们之间会有一些自由空间,剩余的未使用的空间变成由文件割开更小的部分。整理磁盘碎片(defrag)工具会执行碎片整理过程,调整硬盘上所有文件为连续空间,因此,留下所有自由空间作为大的连续过程。一旦你已经整理了硬盘,你将能使用 fips 工具去使用未用空间的一部分或全部的自由空间。fips 是 fdisk 的一个版本,被设计用来探测连续的未用空间,并且从当前的 DOS 分区中删除掉,然后 Linux 系统可使用未被分区的自由空间。

运行整理磁盘碎片 defrag 工具,键入 defrag 命令,它是一种放在 DOS 目录下的命令

```
C:\dos\defrag
```

Defrag 将用带颜色的区块代表硬盘的不同部分来显示一个硬盘。它把所有的使用过的部分(包括你的数据和程序),进行最佳的整理,在硬盘中融为一体。这一过程需要几分钟的时间,这一切结束后,你将在屏幕上看到使用过的部分被安排到一起,然后退出磁盘整理工具。现在,你要准备运行 fips 工具,去开辟自由空间,Fips 安放在你的 Linux 光盘上 \col\tools\fips15 目录下。转入光区,开始运行 fips 工具。在下面例子中,驱动区 e 代表光驱。

```
C:\e:
```

```
E:\>\col\tools\fips15\fips
```

Fips 将在屏幕上显示自由空间的数量,如果你不需要全部的自由空间,你可以用你的光标移动键调整,使空间变小,并留下这些自由空间为你的 DOS 系统程序使用,按回车键,完成自由空间的开辟。

2.2.3 对整个硬盘进行破坏性再分区

代替减少你当前的 DOS 分区的另一种方法是删除它们,重新开始分区。这一过程将删除硬盘上的所有数据。但很多情况下你必须这样做。

Fips 工具只是简单地缩减了分区的尺寸,并不能删除分区。你可将现有分区变成多个小分区,或将现有分区变成大分区。如果你喜欢大分区,你必须对你的硬盘进行破坏性再分区,产生新的分区,然后能从备份中恢复你的 DOS 系统和数据。如果你有一个刚买的硬盘,并立即安装,那么你必须用相同步骤对硬盘进行破坏性分区(当然不用任何的数据备份和恢复,因为一个新硬盘没有任何数据)。如

果你正在使用一个已经有大量文件的硬盘,那么必须备份你的文件,然后对这一硬盘进行破坏性再分区。这样你就可以在以后恢复你想要的文件。你对整个硬盘分区以前,须把所有数据备份到软盘或磁带上。完成这些以后,你即可对硬盘进行破坏性分区。再次提醒读者注意的是:对硬盘进行分区,将删除所有的数据。

下面的步骤描述了怎样运用 format 和 fdisk 工具来创建一个 DOS 启动盘。首先,在软驱内插入一个空白的软盘,用 format 命令和 /s 选项来格式化磁盘,并同时把系统文件复制在软盘上,还要确定软盘驱动器的提示符。下面以驱动器 A 为例,复制 command.com, format.com 和 fdisk.exe 到 A 盘上。Format 和 fdisk 这两个工具存放在 DOS 目录下。

```
C:\>format a:/s
```

```
C:\>copy command.com a:
```

```
C:\>copy dos/format.com a:
```

```
C:\>copy dos/fdisk.exe a:
```

你现在已经有了 DOS 启动盘,接着用 DOS 启动盘重新启动计算机,系统将代替硬盘而用启动盘启动,屏幕将显示 A:\>提示符,现在你可以准备用 fdisk 对硬盘进行再分区,键入 fdisk 命令:

```
A:\>fdisk
```

这时屏幕上将显示带有分区选项的 fdisk 菜单,你可以建立删除或者排列分区,如果安装的盘不是新盘,则必须将已存在的分区删除,这时,选取菜单中的第三选项删除一个分区,而当前的分区将被显示出来屏幕将提示键入是否真的将要删除这一分区,按"Y"键,这时屏幕自动回到 fdisk 选项菜单状态下,选择第三项并重复以上步骤,则可删除另一个分区。继续这一过程,直到所有分区被删除。每次删除

一个分区时,你会注意到列出的分区将较小,自由空间变得较大。

一旦你已经删除所有分区,就可以准备对硬盘进行 DOS 分区,并给 Linux 留出足够的自由空间,选择 Fdisk 选项菜单的第一项,屏幕会显示不同类型分区的选项菜单,一个主 DOS 分区,一个扩展的 DOS 分区和一个逻辑 DOS 分区,你的第一个分区必须是主 DOS 分区,所以选择第一选项。

然后,必须把这个 DOS 分区设置为 active 分区,以便它能用于启动 DOS,选择 fdisk 菜单中的第二选项,然后选择分区 1 为一个 active 分区。

现在你可以增加另一个分区,为了增加分区,选择"Partition Creation"菜单中的 Extended DOS 分区选项,增加分区后可用 fdisk 选项菜单的第 4 项在核对分区列表,最后一项将显示自由空间的数量。要保证为 Linux 系统留出足够空间。

一旦完成了硬盘的 DOS 分区,就要对它们进行格式化,用 format 命令和 /S 选项来格式化主分区。使系统文件能复制到上面以启动计算机。主分区通常用 C: 标称。

```
A:\>format c:/s
```

其它的分区只用 format 命令来格式化,以标称为 D 的分区为例:

```
A:\>format d:
```

一旦你已经完成了格式化分区,就可以取出启动盘,再重新启动计算机,你可以安装 DOS 和 Windows 和你已备份的数据和程序。

2.3 创建 OpenLinux 安装盘

在你的 OpenLinux 光盘上存放着创建安装盘的文件,你必须在运行 DOS 的计算机上创建安装盘。首先启动计算机并进入 DOS 状态,然后执行下面步骤:

1. 在光盘驱动器中放入 OpenLinux Life 光盘;
2. 在 DOS 提示符下,转到光盘驱动器,下面的例子中 E:驱代表光驱

```
C:\>e:
```

```
E:\>
```

3. 进入 \co\launch\floppy 目录,键入 rawrite3 命令,就可以创建安装盘了,你需要复制两张分别包含文件 INSTALL.IMG 和 MODULE.IMG 的两张软盘。

```
E:\>cd co\launch\floppy
```

```
E:\co\launch\floppy>
```

4.在软驱内插入一张 3.5 寸 DOS 格式化的空白软盘。

5.现在开始 rawrite 3 命令。rawrite3 命令是在 \co\launch\floppy 下的 DOS 文件

rawrite3 提示你想要复制的文件名子,键入 INSTALL.IMG 然后键入代表软盘驱动器的字母,通常为 A 驱动器。

```
E:\co\launch\floppy>rawrite3
```

```
Enter source file name: INSTALL.IMG
```

```
Enter destination drive (A or B) and press enter: A
```

6. 确定软盘已插入回按回车键,rawrite 3 将复制文件到你的软盘,创立安装

盘,完成以后,将软盘取出,安装过程将在下面描述。

重复相同的步骤完成带有 MODULE.IMG 文件的软盘的创建。MODULE.IMG 磁盘的用于配置你的硬件。

2.4 安装 Linux

安装 Linux 包括几个过程,开始创建 Linux 分区,装入 Linux 软件,配置 X-Windows 界面,安装 Linux 引导器(LILO)。LILO 用来起动你的系统。

安装程序是一个一步接一步连续完成的程序,你可以用鼠标,光标移动键,空格键和回车键。按 ESC 键可以回到前一屏幕,你只要在安装过程中做出选择就可以了。一些屏幕,像显示器配置屏幕,提供你要做选择的选项列表,另外一些只要求你通过键入 y 或 n,及鼠标选择 YES 或 NO 来完成。在一些情况下,程序会询问你一些你准备好的信息,例如:在 X-Windows 安装过程中,显示卡屏幕会显示一些图形芯片的排列,要求你选择用在你的显示卡上的芯片,你应该准备的信息在本书前面部分已经讲过。

现在可以准备安装了,安装过程的每一步骤描述在下面部分,它将花不超过一个小时。

2.4.1 启动计算机,并创建 Linux 分区

如果你跟随本章第一部分的指导,在硬盘上开辟了自由空间,并创建你的启动盘,就可以创建分区了。首先,你须要用做好的安装盘启动计算机。启动计算机后,安装程序开始执行。

通过它你可以用 fdisk 工具创建 linux 分区。

1.把 Linux 启动盘插入软驱内并从新启动计算机。如果系统可从光区启动,可用光盘代替启动盘。最好用冷启动,完全关机后再开机。

安装程序开始执行后,屏幕先出现一些介绍,然后在底部显示下列提示符:

boot:

2.回车。有时需要键入一些参数,以防 CD-ROM 不被识别(见附录 A)大部分系统不需参数。

安装程序自动检测硬盘时,配置信息会显示在屏幕上。如果出现硬件问题,屏幕会列出。配置信息会充满整个屏幕。通过按 Shift-PageUp 键或 Shift-PageDown 键向后或向前翻页。注意,CD-ROM 的设备名可能是 hdc 或 sdc。如果硬件没有被正确探测到,你可能必须重新启动计算机,在启动提示符下键入正确设备名。例如,对于一个以太网卡,应键入 IRQ,端口,起始地址,结尾地址和设备。附录 A 列出了键入不同设备的语法。

3.如果你的系统有即插即用卡,键入 pnp,回车后继续。

```
>>>init: press <Enter> to continue <<<
```

```
pnp
```

4.一系列语言选项将出现,选择一种语言,回车。

5. 屏幕显示一系列键盘,选择你使用的键盘,回车。

过一会儿,屏幕显示一系列被探测到的 IDE 和 ATA/OTI 硬件。按回车继续。

6. 系统提示是否所有硬件都被正确识别。如果是,按 Y 选择 YES 继续安装,否则,选择 NO。然后,你要选择用 auto-probe 工具自动探测还是手动探测未被识别的硬件,如果选择手动探测,进入内核模块管理器,在这里,你能装入具体硬件模块来探测和配置你的硬件。

首先,选择需要的硬件类型,再选择需要的模块。必要时键入硬件参数(详见附录 A)。

7. 接下来,你应该创建或改变 Linux 分区。第一次安装,你必须创建 OpenLinux 系统被安装的分区。按 Y 选择 YES 创建 Linux 分区。如果你打算升级 Linux 系统,可使用原有分区。这个分区类型应当是 83,etx2 的 Linux 分区。在这种情况下,按 N 键继续格式化分区。跳到步骤 11,开始安装 OpenLinux 系统。

8. 屏幕显示系统硬盘列表,如果多于一个,选择你要分区的硬盘。

9. 系统询问是否转换分区表,按 Y 选择 YES。

现在,你以进入 Linux fdisk 程序,将使用前面为 Linux 操作系统创建的自由空间进行分区并把它格式化。如果使用全部硬盘运行 Linux,你将创建整个硬盘分区。

你需要创建两个 Linux 分区:主分区和交换分区。你需要确定分区有不同的类型。Linux fdisk 是一个行编辑程序,它有一套单字符命令,屏幕将提示你键入命令和确认。在运行 fdisk 过程中,如果出现任何问题,可按 P 键退到先前的屏幕,且不会有任何改动。按 W 键以前,硬盘不会有任何变化,它应是你的最后一个命令。它将真正改变硬盘并退出 fdisk 命令,使你回到安装程序。表 2-1 列出 fdisk 常用

命令。

表 2-1fdisk 命令列表

命令	动作
a	转换启动标记
l	列出未知分区类型
m	命令列表
n	新增一个分区
p	显示分区表
q	不保存所做修改而退出
t	改变分区类型
w	保存所做修改后退出

执行下面的步骤创建 Linux 分区,首先你要创建交换分区。

a.按 p 键显示你当前分区

b. 按 n 定义一个新分区,你会被询问新分区是否为主分区

c. 按 p 指出这是一个主分区

d. 键入你正在创建的分区号码

e. 键入分区的起始号码,这是在提示符末端括号内的第一位数字。这决定了分区的大小。换言之,你可以在数字前键入一个“+”号并在后面键入一个字母 m 来确定分区尺寸。+32M 就是确定了一个 32MB 的分区,在这种情况下,你不必决定最后一个确定分区尺寸的数字。

f. 键入交换分区的尺寸,在 16MB 和 32MB 之间。

g. 按 t 键指示出你设置的 Linux 分区的类型,键入分区号码,然后 HEX 码提示你键入 82。这是 Linux 交换分区的类型。

h. 按 n 键来定义另一个分区。并且按 P 键把它设置成为一个主分区。

i. 键入你正在创建的分区号码。

j. 键入分区的起始号码,这是在提示符末端括号内的第一个数字。

k. 然后在提示键入最后一个确定分区大小的数字,你也可以键入 +300M 来代表分区的大小是 300MB。记住,一个标准的安装最少使用 300MB。如果你有更多的空间,你可以使它变得更大,但你不能让它超出你的自由空间。全部文件的安装要花 664MB。Linux 分区的缺省类型是 83。你不必设置它。

l. 按 w 转换到硬盘上,然后按回车键继续。

下面是一个 fdisk 程序的运行例子,显示出用于创建的分区界面和你要使用的命令。在这个例子中,你可以用上述两种方法来确定一个 500MB 分区的大小。注意,当你被提示键入第一个分区起使数字的时候,你可以在提示符下发现它。在这个例子中,用的是 205,然后是 238。

Command (m for help): **p**

Disk /dev/hda: 32 heads, 63 sectors, 827 cylinders
Units = cylinders of 2016 * 512 bytes

Device	Boot	Begin	Start	End	Blocks	Id	System
/dev/hda1		1	1	204	205600+	6	DOS 16-bit >=32M
/dev/hda4		537	537	826	292320	5	Extended
/dev/hda6		543	543	826	286240+	6	DOS 16-bit >=32M

Command (m for help): **n**

Command action

- l logical (5 or over)
- p primary partition (1-4)

p

Partition number (1-4): **2**

First cylinder (205-827): **205**

Last cylinder or +size or +sizeM or +sizeK ([205]-536): **+32**

Command (m for help): **n**

Command action

- l logical (5 or over)
- p primary partition (1-4)

p

Partition number (1-4): **3**

First cylinder (238-827): **238**

Last cylinder or +size or +sizeM or +sizeK ([238]-536): **536**

Command (m for help): **p**

Disk /dev/hda: 32 heads, 63 sectors, 827 cylinders
Units = cylinders of 2016 * 512 bytes

Device	Boot	Begin	Start	End	Blocks	Id	System
/dev/hda1		1	1	204	205600+	6	DOS 16-bit >=32M
/dev/hda2		205	205	237	33264	83	Linux native

```

/dev/hda3          238          238          536      301392      83      Linux
  native
/dev/hda4          537          537          826      292320      5      Extended
/dev/hda6          543          543          826      286240+     6      DOS 16-bit
  >=32M

```

```

Command (m for help): t
Partition number (1-6): 2
Hex code (type L to list codes): 82
Changed system type of partition 2 to 82 (Linux swap)

```

```

Command (m for help): a
Partition number (1-6): 3

```

```

Command (m for help): p

```

```

Disk /dev/hda: 32 heads, 63 sectors, 827 cylinders
Units = cylinders of 2016 * 512 bytes

```

Device	Boot	Begin	Start	End	Blocks	Id	System
/dev/hda1			1	1	204	205600+	6 DOS 16-bit
			>=32M				
/dev/hda2		205	205	237	33264	82	Linux swap
/dev/hda3	*	238	238	536	301392	83	Linux
			native				
/dev/hda4		537	537	826	292320	5	Extended
/dev/hda6		543	543	826	286240+	6	DOS 16-bit
			>=32M				

```

Command (m for help): w
The partition table has been altered!
Calling ioctl() to re-read partition table
(reboot to insure the partition table has been updated)
Syncing disks.

```

屏幕将警告你,你的分区表已经被转换,按回车键,屏幕将显示出系统的硬盘,这个选项"1. No further disk changes"将变亮,如果不打算在你系统上建立另一个分区,就按回车键继续。

如果你已经有一个你不想要的老的 Linux 分区,你可以用 Linux 的 fdisk 把它删去,记住:只有 Linux 的 fdisk 能删除 Linux 分区,首先,按 p 键显示你要删除的分区号码。按 d 键提示你删除的分区号码,选择正确的号码,如果你不小心键入一个 DOS 分区,你就必须用 fdisk 删除它,如果发生了这种情况,你就要按 q 键退出 Fdisk 命令,而不要按 w 键,这样的话就不会删除任何东西。然后在硬盘的选择菜单上再选你的硬盘并重新开始。

10. 下一个屏幕提示你重新启动你的计算机,记住你的 Linux 启动盘仍然在你的软驱内,如果不在,就要插入它,按回车键启动。你将再次安装所有的内容,现在你的硬盘分区已经准备完毕,继续下面的部分来执行安装。

2.4.2 安装你的 Linux 系统:Open Linux Lite

现在你已将创建了 Linux 分区,可以安装 Linux 系统了。

在步骤 10 中,已经重新启动了你的计算机,前面显示了一引起介绍性信息,在屏幕的底部将显示出启动提示符。再次重复第二步骤到第六步骤来安装程序。但这次你要跳过分区,你将继续进行 Linux 系统的真正安装。

11. 重复前面步骤中的 2,3,4,5,6 当被询问是否转换分区表时,按 N 键或回车键。

12. 下一个屏幕显示你创建的 Linux 交换分区,通过选择它,你可以把它格式

化,要确定 Linux 交换分区已经被选中,它的类型是 82,按回车键来格式化你的交换空间。

13. 下一屏将出现 3 个选项:CD-ROM,硬盘或网络。选择 CD-ROM 来安装 OpenLinux Lite 系统。

14. 按回车键来确认你的 CD-ROM 驱动器。如果你确认你的 CD-ROM 有问题的话,你可以用硬盘来进行安装。你可以把 CD-ROM 复制到一个 DOS 分区,然后从那里开始安装,网络安装允许你通过一个像 NFS 的一个网络到达远距离的 OpenLinux 光盘,你也可以把光盘拷贝到硬盘的某个分区上,然后通过网络来安装。如果你选择硬盘来安装,你必须确定这个已经拷贝的硬盘分区和放置光驱的目录。`\dev\hda1` 表明了第一个分区,如果你选择了网络安装,你必须确定服务器的地址和放置 Linux 光驱目录的路径。

15. 再次显示硬盘上的分区目录。为 Linux 主分区创建的分区将会变亮,其类型为 83。确定主 Linux 分区变亮,然后按回车将其格式化。

16. 格式化的同时可以使 Linux 检测磁盘碎片。片刻后,将完成分区格式化。按下来,你要选择安装选项,根据空间大小将有不同的选择。共有七种选项,同时还有几种小型安装。下面列举四种常用选项。

- 标准的 OpenLinux Lite:标准的缺省安装;占 212MB
- 最小安装:无 Windows 支持的或无很多标准应用程序的完整的 Linux 系统,仅占 58MB。
- 小型标准:包括普通用途软件包;约占 91MB。
- 完全系统包括在 OpenLinux 光盘上的整套软件包和开发工具;约占 667MB。

17. 你应该选择 X-Windows 图形服务器。通过选择表中的服务器并按空格键,可以选择多个 X-Windows 图形服务器。完成后按回车键。为满足不同显示卡中特殊芯片集的要求,可以选择标准的 SVGA 服务器和一个专用的加速服务器。加速服务器的名字包含芯片集的名字(S3 系列)。VGA-16 服务器总是被安装(详见表 2-2)。一旦系统被安装,就会发现不同服务器的详细信息,以及该系统在 /use/X11R6/lib/X11/doc 目录和 /usr/doc/HOWTO/XFree86-HOWTO 文件所支持的芯片集信息。XFree86-HOWTO 列出了所有被支持的芯片集。

18. 阅读屏幕显示的注意事项,然后按回车键开始安装。

标准安装大约占 15 分钟。这段时间内屏幕将显示安装过程,指示已完成的百分率及已安装的软件包。Linux 屏幕保护程序将被激活,过一会显示器将变黑。要恢复显示器只需按 SHIFT,CTRL 或 ALT 键。

表 2-2 XFREE86 服务器

服务器	类型
XFree86_SVGA	彩色 SVGA 服务器
XFree86_VGA16	16 色 SVGA 及 VGA 非加速服务器
XFree86_MonoMonochrome	非加速服务器
XFree86_S3	S3 加速服务器
XFree86_S3V	S3 ViRGE 和 ViRGE/VX 加速服务器 (许多 3D 卡)
XFree86_1128	Number 9 Image 128 加速服务器
XFree86_8514	8514/A 加速服务器

续表

XFree86_Mach8	ATI Mach8 加速服务器
XFree86_Mach32	ATI Mach32 chipset 加速服务器
XFree86_P9000	P9000 加速服务器
XFree86_W32	ET4000/W32 加速服务器
XFree86_AGX	IIT AGX 加速服务器
XFree86_Mach64	ATI Mach64 加速服务器

2.4.3 网络配置

完成安装过程后,系统将提出一系列网络配置的问题。如果你正联接一个网络,你会需要前部分列举的信息,“网络配置信息”去配置 Linux 用于网络的界面。如果你正用一个调制解调器(MODEM)进入 Internet 网,你只需 DNS 信息或某些情况下的网关信息。OpenLinux 也提供了 NIS 网络支持。

- 计算机主机名
- 你的域名
- 分配给计算机的 IP 地址
- 网络 IP 地址
- 子网屏蔽
- 网络的广播 IP 地址
- 网关路由器的 IP 地址(如果存在网关)
- 网络使用的域名服务器的 IP 地址

- 如果网络使用了 NIS 服务器,要知道 NIS 域和 IP 地址

只有了解上述信息才能实行网络配置。以后,你可以轻松地使用 OpenLinux 工具键入配置信息。但是,有了这些信息后,你可以按下列步骤继续网络配置。屏幕将提示键入计算机主机名。如果无主机名,则应自己决定一个名字-通常 4 到 20 字节长(如果计算机已联网,则其应有一个主机名,如果不知其名则应与网络管理员确认)。

19. 键入系统主机名,然后回车。

20. 系统将询问你是否有网卡。如有,选 Yes,系统将继续提问。如果未联网,键入 n,然后继续"最终配置"步骤(如果你有一个单机,你将无法联网)。

- 你应该确认网络设备是 eth0。这是以太网设置的缺省设备。如果不是以太网,选择"NO"。然后会提示网络信息。

- 键入你的四部分 IP 地址

- 键入子网屏蔽。多数配置将是一个缺省值。

- 键入网络的广播地址。

21. 系统将询问网络是否有路由器或网关系统。如果有,则键入其 IP 地址。如果不是,按 n 选择 NO。

22. 指示你是否有一个 DNS 域名服务器。如有则键入其 IP 地址

23. 系统询问是否有 NIS 系统,选 Y 配置它,否则选 N。要配置 NIS 系统,需要键入 NIS 域和 NIS 服务器的 IP 地址,现在,网络配置已经完成。

2.4.4 最终配置

下面的两个屏幕提示选择当地时间和格林威治时间及正确的时区。对 PC 用户来说,普遍选当地时间,DOS 系统通常根据当地时间来操作,然而,对 Unix 系统,格林威治时间(GMT)是标准的,并且格林威治时间会给 Internet 网中其它 Unix 计算机提供更好的转换能力。

24. 选时间类型后,按回车,在下一个屏幕选一个时区,然后按回车。

25. 下一个屏幕将列出一系列鼠标类型。选择鼠标类型,按回车。

26. 下一屏幕提示选择鼠标所使用的一系列端口,选后按回车(通常是 COM1)。

27. 然后,提示设置打印机。首先,屏幕提供打印机各列表,然后用光标选择其中之一。

28. 选择打印机端口(通常是第一个并行端口)。

29. 选择缺省的打印结果(对多数打印机是 300*300)

30. 选择缺省纸张的尺寸(通常是信纸的尺寸),键入简单易记的超级用户指令。你必须以超级用户的身份来配置系统,执行类似于安装软件或配置应用程序等的管理操作。

31. 在口令提示符下,键入超级用户口令。

32. 再次键入超级用户口令,以确认。

33. 屏幕提示键入另一口令,该口令是称作 col 的普通用户帐号。创建并键入一个口令,重新键入以确认。可以用类似于 usecfg 的管理工具来增加新用户。

2.4.5 安装 LILO

接下来,屏幕将显示 LILO 的启动设置分析。LILO 是 Linux 的装载工具,启动时,LILO 会开启 Linux 系统。然而,如果硬盘上存在多个操作系统,LILO 会提供选择机会。如果你未选定任何操作系统,LILO 将以缺省形式指派其中一个。如果硬盘上有 DOS 和 Linux,LILO 将让你选择。

34. 按回车键继续。

35. 选择将要安装 LILO 启动管理器的分区。如果无法确定安装位置,则选缺省状态,该选项将会发亮。缺省是根据系统分析选定的(通常是 Linux 根区)。

36. 选择做为缺省状态开启的系统。通常是 Linux 或 DOS。Linux 将由 /vmlinux(Linux 核心映像)。DOS 或 Windows 95 系统将包含末端带 DOS 或 OS/2 的 DOS 术语。

37. 键入一系统标志。将会显示一缺省名词。dos 是 DOS 系统标志。Linux 是 Linux 系统标志。按回车键确认这些标志。如想变更按后退键后更改。

38. 如果选 Linux 系统,则特殊的启动参数将需确认,如无特殊参数,则该选项为空白,选 OK 继续。

39. 现在可另选其它的操作系统,屏幕会显示硬盘上所有的系统,包括 DOS 和 Linux。如果你既有 DOS 又有 Linux,你要选择其一。例如,选 DOS 为缺省系统,则必须选 Linux 为其它操作系统。

40. 选择 DOS 或 Linux 来做缺省的操作系统。

41. 屏幕提示键入另外一个操作系统的名字,如果选 Linux,屏幕会显示 Linux 然后选 OK,否则,键入其它的操作系统的标志。

42. 你可以用同样方式为许多操作系统设置标志。如果有第三个操作系统选择其分区,重复前面的步骤。

43. 一旦完成上述选项,选择 "No further entries to add to LILO"选项,按回车,屏幕会显示 "lilo.conf"文件,这是配置 LILO 的文件。然后会看见你想进入的操作系统选项。按回车确认 LILO 的安装。你将按配置安装 LILO。然后按 "y"或回车结束 LILO 安装后再按回车。然后标记带有 LILO 的分区,再按回车。

2.4.6 完成配置

现在你将完成配置的最后步骤

44. 当 Linux 启动时,系统自动提供服务器列表,包括 Apache Web Server。所有在 OpenLinux Lite 中所有可获得的都是已经被选好的服务器。你可以将光标移至该选项并按回车键来加选服务器。按回车键继续。

45. 应该选择 X-Windows 服务器。将光标移动到你想要的选项,然后按回车键继续。

46. 系统询问是否想要配置 X-Windows 服务器。对此最好的回答是 "NO"。然后完成安装过程。X-Windows 的配置会很复杂。如果不当,系统将会崩溃。这将损坏整个安装,你必须重新安装。按 "n"键或回车键来结束安装。你已经全部完成了你的安装。

47. 拿出软驱中的驱动软盘,然后按回车键,几分钟后系统将自动启动。当系统初始化完成后,注册提示符将出现。然后就可为任何用户设置一个注册名和口令。安装期间,你已经设置了一个超级用户和一个名叫 col 用户的名字和口令。

用它们其中的一个来注册。如果以超级用户的身份来注册,就可以执行象安装新软件和创建新用户的管理操作。键入 root 和回车键,实现超级用户的注册,然后在用户口令提示符下键入超级用户口令。然后回车,在某种情况下,你打算运行一个程序时,收到了丢失文件的错误时,用 Lisa 去核对是否安装了此软件包,如果没安装,就装上它,如何注册和推出系统详见第 3 章。

如果你正在升级一个 Linux 系统,并且打算使用已经备份的配置文件的,那么现在就可以恢复它了。安装存储这些文件的软盘,复制配置文件到你的新系统。你也可以用 tar xvMf/dev/fdo 命令来恢复 tar 文件及软件包。

完成以后用 logout 命令退出你的帐户。

```
$ logout
```

然后你需要关闭整个系统,无论什么时候你打算关闭系统,同时按一下 CTRL-ALT-DEL 键,对于关闭系统来说,总是用这三个键是十分重要的,如果用 DOS 系统就不用这种方法关机了。

在任何情况下,一旦你启动失败,你可以用执行紧急启动的安装盘,把安装盘放入软驱里,在启动提示符下,键入 "boot ro root="和启动 Linux 分区的设备名。例如:如果启动 Linux 分区时 /der/hda2,然后键入 boot ro root=/der/hda2。

```
boot>boot ro root=/der/hda2
```

2.5 安装和配置 X-Windows

首先你要用超级用户身份注册在 LILO 提示符下,键入 Linux 并按回车键。

- 1.在注册提示符下,键入 root,然后回车。
- 2.系统立刻提示你键入超级用户口令。
- 3.在口令提示符下,键入你为超级用户设置的口令,然后回车。

使用 X-Windows 以前,你必须配置鼠标键盘,显示卡和显示器,以支持 X-Windows 界面,用 XF86setup 程序配置 X-Windows。这是一个全屏幕的图形用户界面,在界面上单击按钮和菜单来选项。任何时候你都可以重新配置你的 X-Windows 界面。配置你的显示卡,可以从列表中选择。

配置 X-Windows 以前,找到显示器垂直和水平的刷新频率(然后参考显示器的硬件文件,这个频率是一个数字或者范围,例如一个 NEC3V 的显示器,水平频率是 31-50,垂直频率是 (5-90)。

4.执行 XF86setup 程序

在 Shell 提示符下,键入这个命令来执行 XF86setup 程序,这处程序安放在 /usr/X11R6/bin 目录下。它将创建用于系统的 X-Windows 配置文件.名叫 /etc/XF86Config。

```
$/usr/X11R6/bin/ XF86setup
```

如果你的 XF86setup 没有执行,原因可能是安装时没有完全安装 Xfree86 库文件。用第 3 章所述的 Lisa 工具,安装 Xfree86 软件和 Xfree86misc 软件。然后执行 XF86setup 文件。如果仍然不行,你就用 XF86Config。

5.你应该键入图形模式,按回车。

如果正在运行 XF86setup 程序,需要完成 5 个配置任务。每一个显示在 XF86setup 窗口顶部的一个按钮。这个按钮分别标有 Mouse,Keyboard,Card,Monitor 和 Other。每一个按钮可以产生一个配置该项

任务的屏幕。

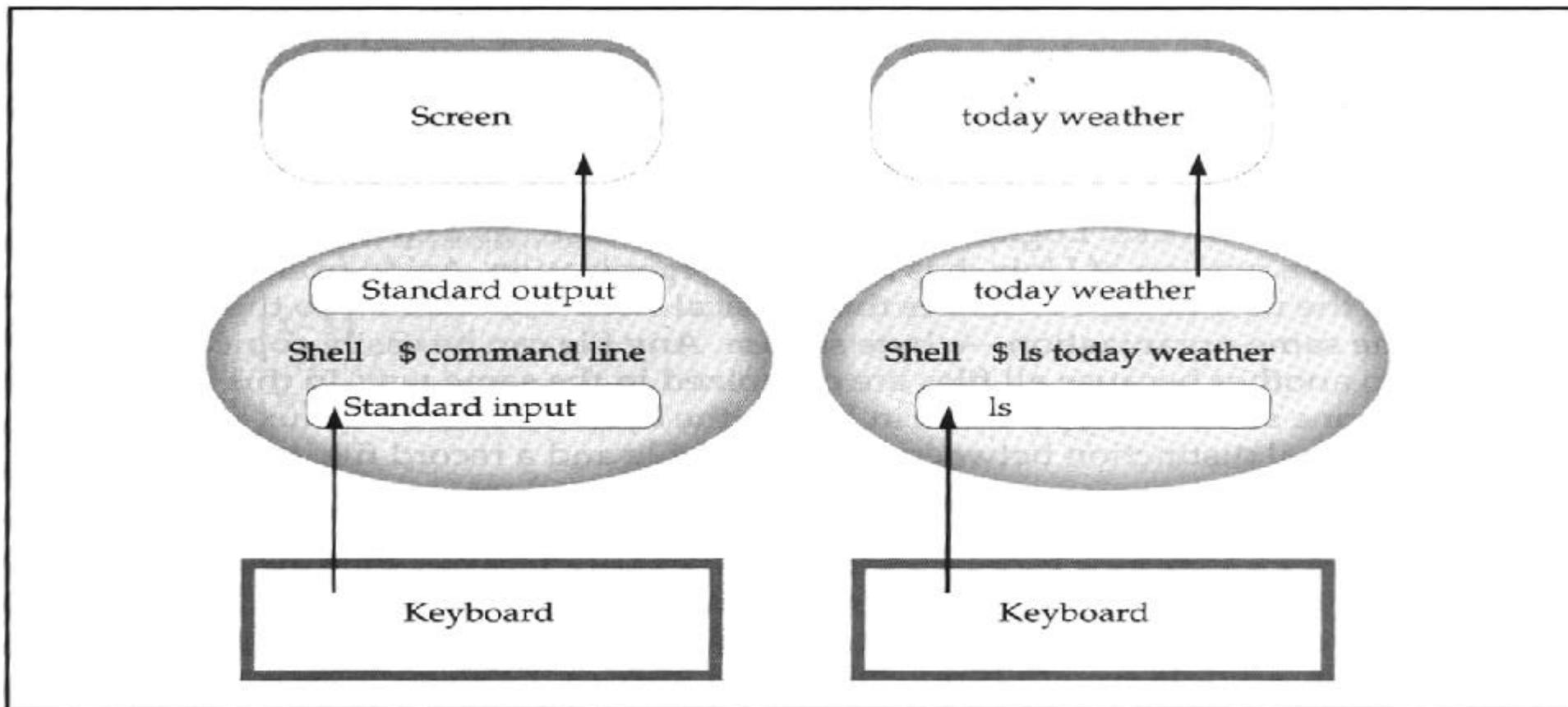


图 2-1 描述鼠标内容的 XF86setup 程序

屏幕的底部有三个按钮：Abort, Done, 和 Help. Abort 将取消 XF86setup 程序。如果你不能确定任何配置选项,你只要单击 Abort,就可以安全地结束程序。当你完成了所有的配置任务。单击 Done 按钮,存储你的配置,然后退出 XF86setup 程序。然后用 startx 命令初始化 X-Windows。

6.按回车或者 ALT-M 键 ,可以出现鼠标屏幕。鼠标屏幕第一次出现是列出

了许多键盘命令。鼠标配置以前,建议你不要使用鼠标。你可以用键盘命令来选择屏幕上的选项(看图 2-1)。

鼠标屏幕列出发一系列的鼠标品牌。在标有"选择鼠标协议"的标志下,按动屏幕顶部的按钮来选择要的鼠标品牌。Microsoft 是大部分鼠标都兼容的协议。大部分标准的鼠标是 Microsoft 系列。你可以根据鼠标的不同特点来设置按钮,缺省设置已经被键入。对于鼠标的路径来说,/der/ttys0 指定了第一个串行口,/der/ttys1。

指定了第二个串行口。系统已经选择了/der/ttys0,这是鼠标常用的端口。B 键设置了波特率。e 命令显示出中间按钮的配置内容,c 命令可以使用中间按钮的配置内容。完成上述命令后按 a 键选择右下角的 Apply 按钮。你就配置好了你所要用的鼠标,并且可以用它来配置其它内容。

7.单击键盘按钮显示出键盘屏幕,会出现一个选择键盘模式和语言的下拉菜单。你可以设置不同的控制键,或者使用缺省值。完成这些以后,选择键盘图像中的 Apply 按钮(看图 2-2)。

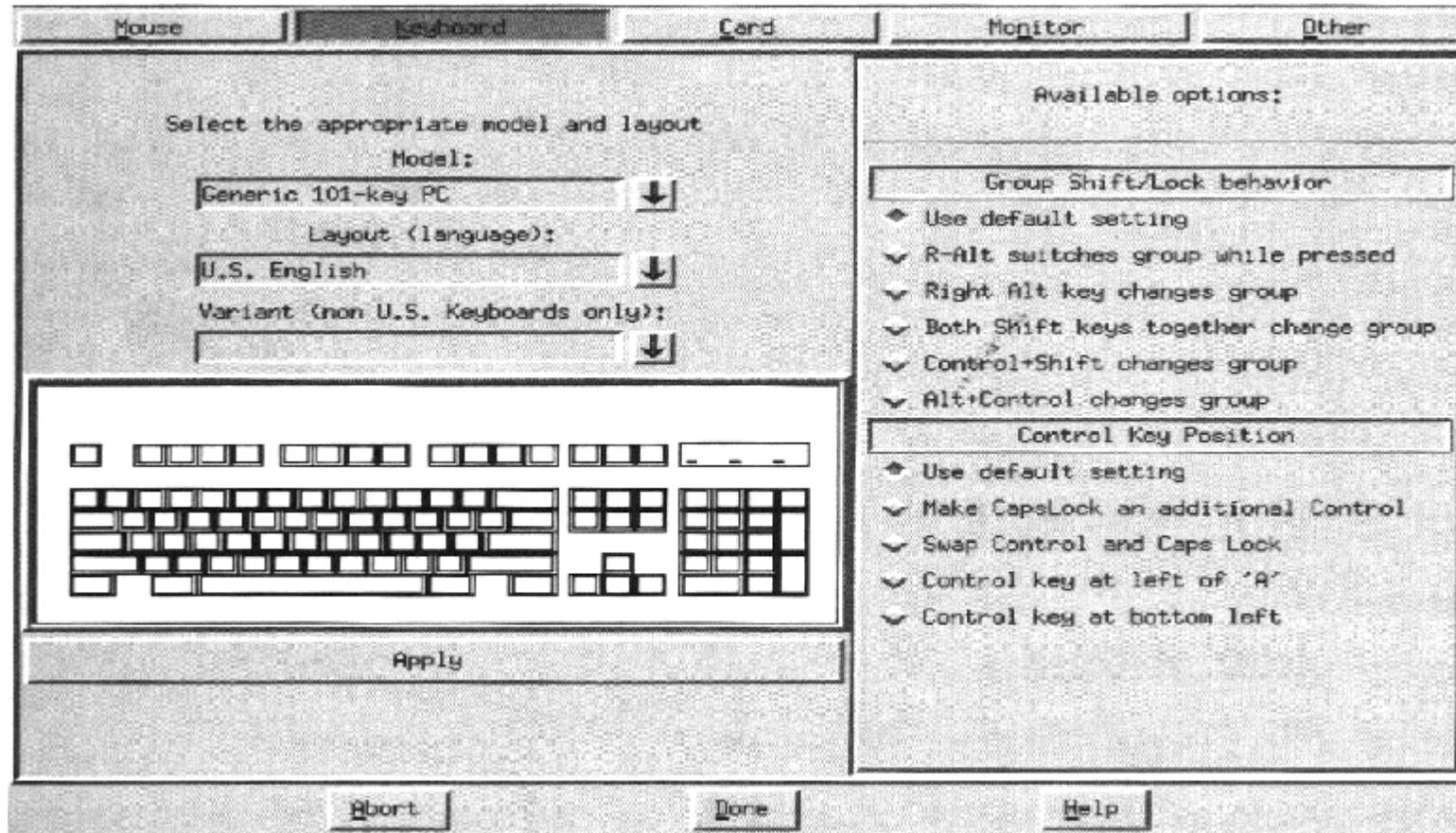


图 2-2XF86setup 的键盘屏幕

8.用于显示卡配置的两个屏幕:详细的设置和卡的列表屏。正常情况下,你必须使用卡的列表屏。单击 Card 按钮启动。如果你前面没有运行 XF86setup 程序,就立刻会出现卡的列表屏。如果你已经运行了 XF86setup 程序,首先会显示出详细设置屏幕。在屏幕的右下角是一个标有 Card List 的按钮。选择这个按钮来开启卡的列表屏。

卡的列表屏显示了当前大部分的显示卡。右侧是能移动列表的流动条。找到系统所使用的显示卡,单击它选择。然后单击 README 文件,查看卡的信息服务(见图 2-3)。然后继续配置显示器。

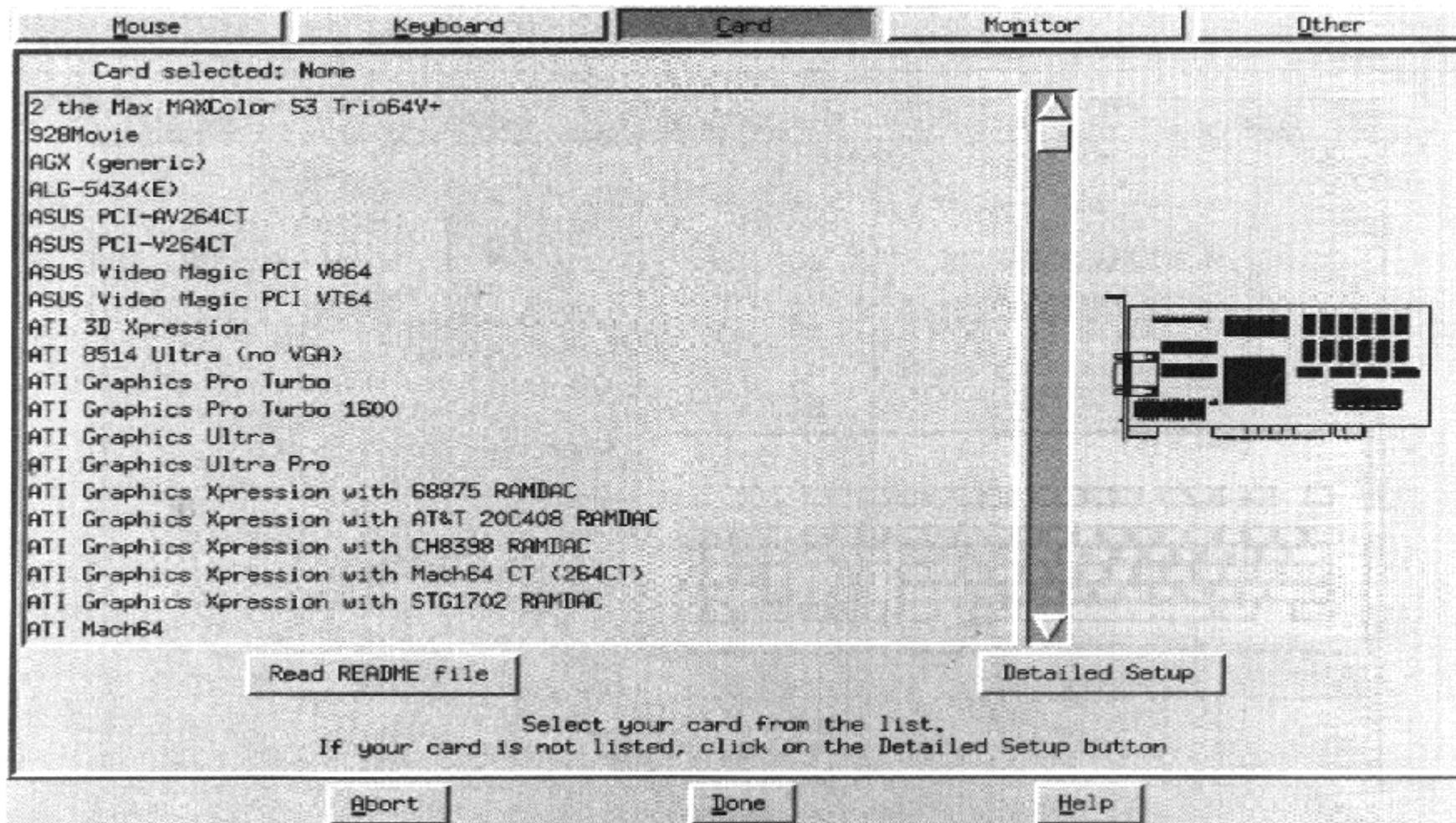


图 2-3 支持显示卡的 XF86setup 的卡的列表屏

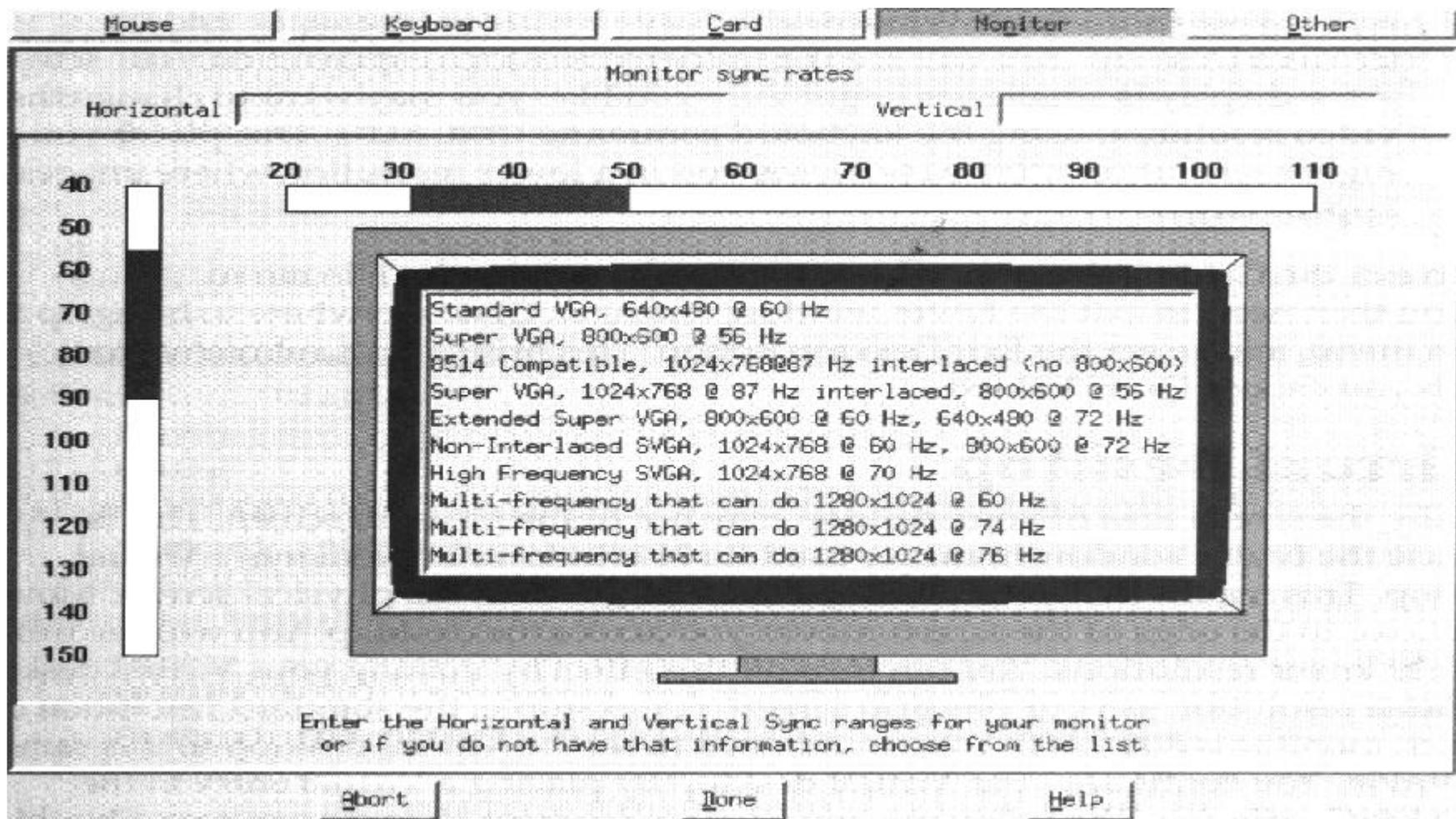


图 2-4XF86setup 的显示器配置屏

如果你的卡没有列在列表中,选择详细设置现详细设置窗口,你可以对你的卡进行详细设置。在详细设置窗口中,你将看到顶部有一排按钮指示不同的 X-Windows 服务器。它们的下面是一个选择芯片集的下拉菜单,也有选择键盘方式和语言的下拉菜单。

9.单击显示器按钮弹出后显示器屏幕,设置显示器的水平的垂直扫描频率的范围。但是,这个频率的设置是非常重要的。不正确的频率可以损坏显示器(见图2-4)。

注意选择显示器频率以前,核对你的硬件文件,如果你设置的频率高于显示器的要求,将严重损坏显示器。

屏幕的顶部标有 Horizontal 和 Vertical 的方框。在这里你可以键入水平和垂直的扫描频率,按回车键。你能看到这个频率被再次打印到有关的栏目中,如果你的显示器支持每一个选项范围,你可以分别地来键入它。例如一个老的多信号显示屏,它可能有 31-50 的水平扫描频率和一个 55-90 的垂直的扫描频率。垂直和水平的方框下面,有一个图显示出你键入的范围。在屏幕的中心,有一个方框,列出了普通的显示器类型,你能够单击其中一个,选择标准的频率设置。一定要核对显示文件设置的正确频率。

10.单击其他的按钮产生服务器设置,列出了五个选项,前 2 个选项是已经被选中的。一般来说你不用改变屏幕上的任何选项。第 1 个选项允许你用 CTRL-ALT-BACKSPACE 键退出 X-Windows。第 2 个选允许你用 CTRL-ALT 和任意一键来改变显示类型。不同结果之间的转换是容易的。第 3 项允许你退出服务器。第 4 项和第 5 项允许你从远处的工作站来改变显示器,鼠标和键盘的配置。

11.完成 X-Windows 配置后,选择屏幕下面的 Done 按钮, X-Windows 启动,然后出现 3 个按钮: Run xvidtune, save the configuration 和 exit,或者 Abort。如果 X-Windows 启动失败,原因可能是没有安装 XFree86 服务器。回到命令行,用 Lisa 命令安装你需要的服务器(详见第 2 章)。

12.选择 save the configuration 和 exit。

13.你能用 `startx` 命令启动 X-Windows。这个命令可以启动 X-Windows, `fvwm` 窗口管理器和 Caldera 工作平台,然后你就能看到这个工作平台显示在屏幕上。如果这个显示出的平台比你想要的大或是小,可以用 `CTRL-ATL` 和任意键命令来改变显示结果。这样你就会得到你想要的平台的尺寸。

从文件菜单中选择退出工作平台,就可以退出 Caldera 平台。然后退出 `fvwm` 窗口管理器。单击 `fvwm` 菜单的任意处,选择退出平台。这将出现退出平台的一个子菜单。

2.5.1 实际的工平台

最初,你工作平台的屏幕面积可能会不理想。这是因为 `fvwm` 窗口管理器实现了一个实际的工作平台,这意味着你的工作平台要比实际的屏幕大,移动光标到屏幕的边缘。你将看见不同结果的影响。你能设置实际的工作平台和的实际屏幕一样大。例如:一个 15 寸显示器的标准输出是 `800*600`,你需要设置相同结果的实际工作平台,通过选择 `/etc/XF86config` 文件中的 `Virtual` 选项,来配置你的工作平台。下面的选项就是为绝大多数 15 寸显示器设置的。如果你想要一个更大的实际工作平台,你可以用 `Virtual 1024 768` 来实现较大的尺寸。或者用 `Virtual 1152 864` 来实现更大的尺寸。

你的这个文件被分成几个控制 X-Windows 显示器的若干部分。在文件末端放置看屏幕选项的显示子目录下的实际设置。屏幕部分有 `Section,"Screen"` 标题。其显示部分有 `Subsection ,"Display"` 标题。系统中的不同驱动器有更多的屏幕。如果安装了一个加速卡,就用屏幕中的 `Driver "Accel"` 选项(详见 21 章)。

加速卡的屏幕中通常有几个显示子目录,包含了显示卡支持的各个部分,例如 8.16.24,缺省的深度是 8。在显示子目录部分中,增加 Virtual 选项。如果你要用另一个深度,就要用 Default Color Depth 选项。这将作为缺省状态选择其他显示子目录的方法。深度决定了显示屏上颜色的 16 位,为 1600 万种颜色的 24 倍。参考 XF86Setup 程序中关于你的显示卡信息,(通常是 8 位,16 位或 24 位)例如:如果你设置了 Default Color Depth 为 16 位,深度 16 的显示子程序就会被使用。确定你有一个想使用的 Virtual 选项,下面的例子表明发加速驱动器的屏幕程序和为 8 和 16 位的显示子目录。为 8 位的使用的工作平台是:1024*768,然而用 16 位深度的 800*600 的工作平台将适合 15 寸的显示器的尺寸。

```
Section "Screen"
```

```
Driver          "Accel"
```

```
Device          "Primary Card"
```

```
Monitor         "Primary Monitor"
```

```
DefaultColorDepth 16
```

```
SubSection "Display"
```

```
Depth          9
```

```
Modes          "1152X864""1024x768""800x600""640x480"
```

```
"640x400""480x300""400x300""320x240""320x200"
```

```
virtual 1024 768
```

```
EndSubSection
```

```
SubSection "Display"
```

```
Depth         16
```

```
Modes      "1154X864""1024X768""800X600""640X480""640X400"  
Virtual 800 600  
EndSubSection
```

你能用第 4 章中的 Crisplite 编辑器来编辑文件,用 `mcr/ect/XF86Config` 文件在命令行中编辑文。或者用 Vi 或 X(xEmacs)。首先要给这个文件作备份。要确认已经选择的显示屏上的显示目录。注意其它的部分不能有剩余。

2.6 安装声卡的驱动程序:声音

你必须安装声卡的驱动程序,它不会由 OpenLinux 安装来完成。声卡驱动程序放在 `Sound.o` 的内核模块中,并且在 `/lib/modules/2.0.29/misc` 目录下(2.0.29 是 OpenLinux 内核的当前版本)。Linux 内核支持 OpenLinux 版本的模块,是老版本不能支持了。这个模块包含了当前的标准声卡,用一个地址是 220 的 IRQ7 和 DMA1。

你安装了带有 `insmod` 命令的模块被显示在这(`insmod sound` 将卸载声音模块):

```
$ insmod sound
```

不管你何时启动这个系统,声音模块将自动装入,你必须在 `/etc/modules/2.0.29` 目录下的缺省文件里加入 `sound`。在你的这个系统中,文件的名称是以带有 `defaults` 扩展名的 Linux 系统的双版本。一个容易的方法是用 `*.default` 或 `uname-v .default` 做为文件名来编辑它。

用 `uname-v` 显示商业版本的名字。在 `/etc/modules/2.0.29` 目录中,下面的命令可以使缺省模块文件的末尾加上 "sound" 一词。 `uname-v` 两边的单引号和全名两边的双引号以及 `>>` 均必不可少.这个缺省文件的当前名字是 `#1 Tue Feb//20:36:48 MET 1997. default`。

```
$ echo "sound" "  uname-v  .default"
```

你能核对下面命令的声卡。如果你收到没有设备的信息,那么证明声卡没被安装。

```
$ cat /dev/sndstat
```

用 CD 播放器,或声卡文件中的 `cat` 测试你的声卡。如果听不到声音,那么卡就没有配置好。带有 `.au` 扩展名的声音文件被重新指到了 `/dev/audio` 设备上带有 `.wal` 或 `.voc` 扩展名的文件指到了 `/dev/dsp` 设备上。下面的例子播放了一个声音文件:

```
$ cat sample.au >/dev/audio
```

首先你必须产生一个用于你的光区设备名字的 `/dev/cdrom` 连接运行 CD 播放程序。如果你的声卡有不同的 `IRQ` 和 `DMA` 设置,或者是不被当前的配置所支持,你就必须重新编辑新配置的 `sound.o` 模块。这个声音模块的原代码是在 `/usr/src/linux/drivers/sound` 目录下。查看 `README` 文件,了解怎样配置和编辑模块,特别是 `README` 模块。从这个目录下,键入 `make config` 命令来配置声音模块,系统会询问你一些声卡的信息。配置完成后,用 `make` 命令,编辑新的卡模块,然后用 `make install` 命令,把新的 `sound.o` 拷贝到 `/lib/modules/2.0.29/misc` 目录下,重写 `sound.o` 文件,然后用 `insmod sound` 安装新的文件。

2.7 Lisa

Linux 安装和系统管理 (Lisa) 程序执行整个过程的安装。你能用相同的程序来改变全部的配置。用 Lisa 能修改你的系统。例如:增加新用户,安装新的分区或者安装新的打印机(见图 2-5)。你也可以增加和删除光盘上的软件改变网络配置。Lisa 是一个简单的下拉菜单的工具。

从命令行键入 Lisa 命令来启动。

```
#Lisa
```

通过光标移动键来选择菜单和按钮,当项目变时,说明已经被选中。实际上第 3 章中详述了用 Lisa 管理系统和增加新软件。第 7 章中描述发用 Lisa 管理你的文件系统。用 Lisa 来获得系统信息。浏览帮助文件及增加和删除特殊硬件中内核模块。

选择 Lisa 主菜单中,"Vervose system analysis"选项获得有关系统的信息(见图 2-6)在这个菜单中,可以找到你的系统是怎样配置的。例如:"Automatic Partition Analysis"列出了系统上所有分区。"Automatic Network Analysis"列出所有的网络信息。你可以用这些项目去核对安装过程中所提供的信息是否正确。如果不正确,就用 Lisa 改变它。Lisa 主菜单中的 "System Configuration" 选项让你进入增加或者改变系统配置特点的其他菜单。

如果你的计算机有特殊的部件使系统出现问题,你需要安装 Linux 驱动器,或者使用具体的硬件参数。Lisa 可以运行内核模块管理器安装这些驱动器或者设置硬件参数(详见附录 A-硬件参数列表)。从 Lisa 主菜单中选择系统配置。再从

下一个菜单中选择系统配置。选择"Kernel Module Administration"从这个菜单中你能看到已安装的内核模块列表及装入和删除的模块。

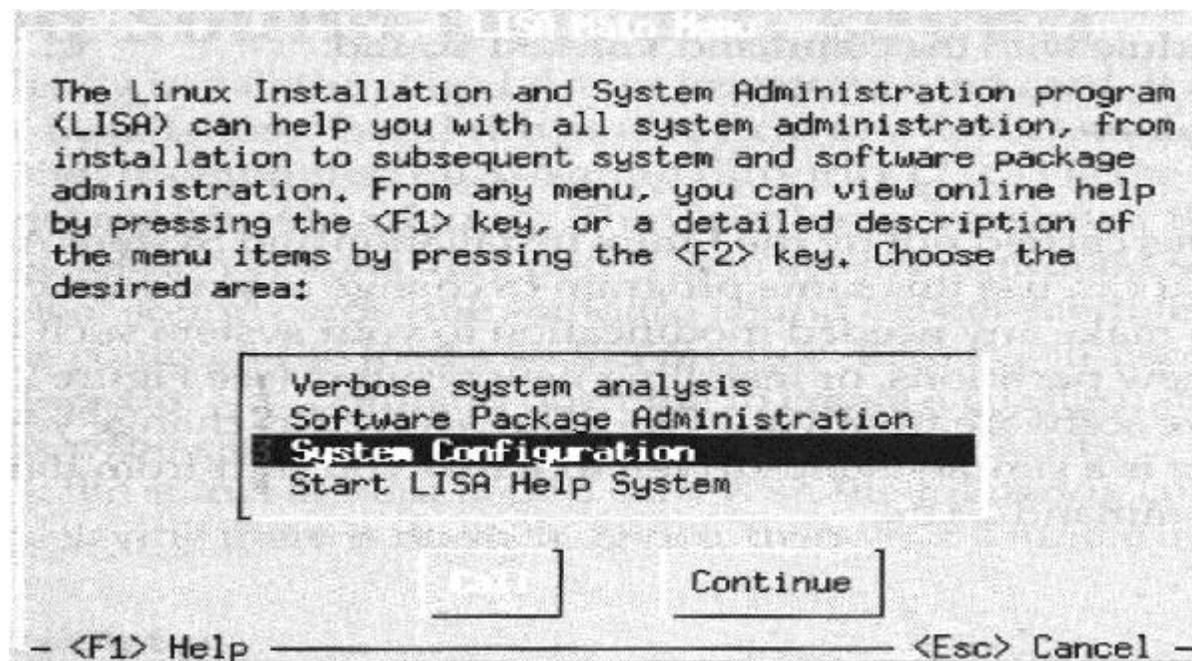


图 2-5 Lisa 主菜单

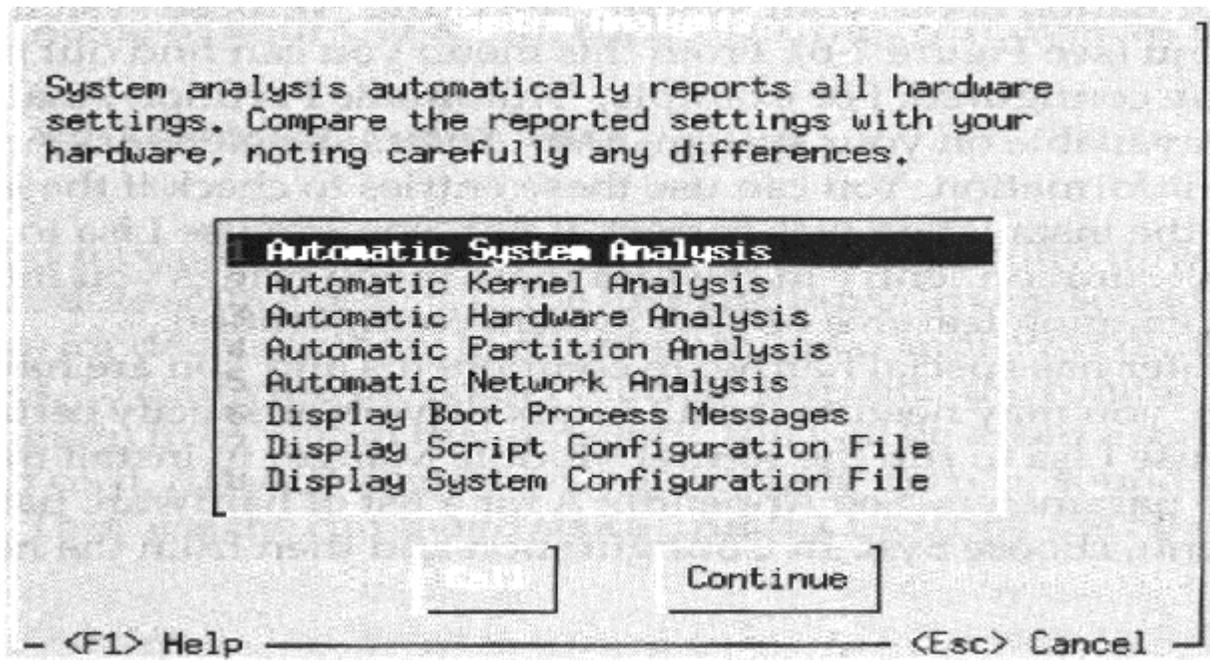


图 2-6 Lisa 系统分析

第 3 章 Linux 系统的启动和安装

要想用 Linux 启动,你需要知道如何访问 Linux 系统,一旦你已登录到 Linux 系统,要知道如何运行应用程序。一旦 Linux 在运行,你必须用预先设定的注册名和口令来注册系统。进入系统后,你可以执行命令和应用程序。用命令行界面和图形用户界面与你的 Linux 系统相连。Linux 系统使用带有窗口管理器的 X-windows 和带有图形用户界面的工作平台,这样你可以用窗口,菜单和与你的系统进行交互。

当注册进系统后,可以容易获得 Linux 系统的命令和使用工具的信息。Linux 几个网上工具提供信息和帮助。你能得到描述每个命令的网上在线说明和不同 Linux 特点的详细解释。系统中 Linux 资料提供了完整的说明,可以通过打印或者网络浏览器来获得这些信息。本章主要讨论了怎样登录你的 Linux 系统,包括用户帐号的注册和退出以及系统和启动和关闭。Caldera 工作平台的基本操作包含了 Linux 命令和使用工具。本章的末尾解释了基本的系统管理操作。例如产生新的用户帐号和安装软件包。

3.1 用户帐号

你永远不会直接访问 Linux,因为 Linux 设置了一个你和他交互的一个界面,Linux 系统能一次设置几个用户界面,同时满足几个用户的需要。事实上,许多用户可以工作在一台 Linux 系统的计算机上。对于不同的用户来说,只有他一个人在使用这个系统。实际的计算机是独立的管理用户和 Linux 系统交互的界面。这些用户界面经常被指定为一个帐号。作为 Unix 基础的 Unix,第一个首先被用在小型机和大型机上,能同时满足上万个用户的要求。用一些终端和主计算机相连,用户能够通过它们注册到 Unix 系统。所有的这些活动由系统管理员来管理。为了登录到系统上,你要有一个设置好的用户界面,这就是众所周知的一个开放帐号。一个系统管理员能在 Unix 系统上创建帐号,分配注册名和口令,然后就可以用此帐号登录和使用系统。一个注册名能识别一个帐号,并用一个口令来保护。当然,如果你知道注册名和密码,你可以用任何的帐号来登录。在你的 Linux 系统中,可创建几个帐号,其他的人想要登录你和系统就要使用你提供给他们们的注册名和帐号,其结果,他们会在自己的系统上拥有自己的帐号,在前一章 Linux 系统中,你已经创建了一个你自己的注册名和口令。当你创建注册名和口令时,就创建了一个新的帐号。

事实上,你能用特殊的系统管理工具去创建新的用户帐号,当你以超及用户的身份登录时,就可以使用这些工具。超级用户是一个特殊的用户帐号,它是系统保留的执行管理用户的帐号。例如:创建新用户和安装新软件。这一章讨论了基本的系统管理操作。在第 7,19,20 章中将再次详细讨论了这个内容。现在,人需要

普通用户的注册名和口令。

3.2 登录 Linux 系统

根据下面的启动和关闭程序来登录和使用 Linux 系统。你有能象 DOS 系统那样开机和关机。你要认为 Linux 操作系统正操作在三个不同的水平上。第一层是开机时系统的装入和运行。它控制你的计算机和外周设备。但是,你仍然不能和它交互。Linux 启动后,会出现一个注册启示符,等待用户的注册。

第二层是 Linux 的使用。现在你可以使用命令来执行任务。用一些编辑器,编译器或者游戏等的工具和使用程序。但是,开始注册时,你是用命令行界面和系统连接,通过输入命令和回车键来完成系统的活动。Linux 的 Caldera 分支 OpenLinux,支持许多窗口管理器和工作平台。包括了它自己的工作平台外形,使你能使用 X-Windows 图形用户界面。在 Linux 中, Startx 命令将启动 X-Windows 图形用户界面(GUI)。然后用窗口,菜单和图标来和系统交互。第三层是窗口管理器。X-Windows 窗口管理器运行在命令行界面上。在关机的过程中,第三层变得非常重要,如果你正在使用 X-Windows 界面并想关机的话,首先你必须退出 X-Windows 界面,回到命令行界面,然后退出你的 Shell,回到系统的注册提示符下,最后通过 shutdown 命令通知系统关机。同时按住 CTRL-ALT-DEL 键,系统将自动关闭并重新启动开始时,你才能关闭计算机。

可以用一些快捷键来进行层次转换。超级用户可以通过 shutdown 命令来退出和关机(详见第 19 章关机的过程)。当你注册时,还有一种自动启动 X-Windows

窗口器的方法。

3.2.1 初始化和关闭 Linux

当你开机时, Linux 引导器(LILO)将首先决定装入和运行的操作系统。你将看见下面的提示符,键入 Linux 启动 Linux 操作系统。

```
LILO: Linux
```

如果你不这样做, Linux 将装入缺省的操作系统。如果你要进入 DOS 系统, Linux 将给你几分钟提示你输入 DOS 系统的名字。如 dos。

当 Linux 装入时,就会出现一些信息,然后出现一些注册提示符,系统保持等待用户注册和使用。键入用户名和口令,系统和主机名将处理注册提示符。在这个例子中,主机名是

```
turtle.trek.com:
```

```
Caldera OpenLinux(TM)
```

```
Lite 1
```

```
Version 1.1
```

```
Copyright(c) 1996-1997 Caldera Inc.
```

```
turtle.trek.com login:
```

完成 Linux 系统后,首先要撤销帐号, Linux 系统就会出现一个注册提示符等待另一个用户注册。如果你要关机,首先使用 shutdown 命令,通过同时按 CTRL-ALT-DEL 键来关闭系统。当 Linux 系统关闭时,你会看到一些信息,然后 Linux 重新启动计算机,启动过程中,你可以关机。

下面描述了全部的启动和关机程序。

1.启动计算机

2.在 LILO 提示符直接键入 Linux,按回车(如果 Linux 系统是缺省的,只按回车)。

3.出现一些信息后,注册提示符出现,你可以注册到系统,开始使用它。

4.在注册提示符上,你可以关闭系统,当你注销帐户后,注册提示符会重现。

5.关机

3.2.2Linux 的注册和注销

命令行

一旦你注册,你可以执行系统命令。完成后,并在关机前退出系统。你也可以不必关机。在注册提示符下,注册另一个用户。

Linux 系统的注册

Linux 系统的注册包括两个步骤:键入用户名和口令。如果犯了错误,使用后退键就可以删除字符。在下面例子中,用户的名字是 richlp,提示你键入口令。

```
Caldera OpenLinux(TM)
```

```
Lite 1
```

```
Version 1.1
```

```
Copyright (c) 1996-1997 Caldera Inc.
```

```
turtle.trek.com login:richlp
```

```
Password:
```

当你键入口令后,它不会显示在屏幕上,这是对你口令的保护,如果注册名和密码有错误,系统会显示注册不正确的错误信息,要求你重新注册。

一旦你已经正确输入用户名和口令,就可以作用本系统了。系统的命令行和提示符将出现,等待你键入命令。注意:普通的命令行提示符是\$,而不是#。\$是普通用户的提示符,#是超级用户的提示符。

```
[turtle/home/richlp]$
```

改变你的口令

第一次用超户注册时,你创建了用户帐号的密码,创建后,你可以给你的个人帐号改变口令,只有你的个人帐号注册时,才能改变口令,你不必以超级用户的身份注册,每个用户都

可以改变他自己的口令。

注册后,通过 `Passwd` 命令来改变口令。首先要决定新口令是什么,那是容易记的,至少 8 位到 7 位。为了安全起见,口令=包括行键入 `Passwd` 命令,再按回车,命令提示符你键入当前的口令,然后提示你进入新的口令。系统会要求你键入两次口令,以确认真地改变口令。由于口令不会出现屏幕口,所以可能会按错键。

```
$ passwd
```

```
Old password:
```

```
New password:
```

```
Retype new password:
```

```
$
```

键入新口令时犯了错误,系统会显示出错信息,口令不会被改变。系统通过两个新口令的比较,如果他们不相匹配,错误就会被探测到。

Passwd 命令使你很快更改新口令,当你再次注册时,你就必须使用新口令。

注销

键入 logout 命令注销帐号,回到注册提示符下,系统等待另一个命令。

```
$ logout
```

3.2.3 启动和退出窗口管理器: X-Windows

一旦注册到系统后,选择使用 X-Windows 界面和你的 Linux 系统交互,通过在命令行键入 Startx 命令启动 X-Windows 界面。X-Windows 装入指定的窗口管理器。Linux 的 Caldera 分支也会装入一个工作平台,并提供文件和程序管理器。使用 Linux 的 Caldera 分支的界面被指定为 Caldera 工作平台,一旦被装入,用鼠标使用菜单打开窗口和启动程序(图 3-1 显示了 Caldera 工作平台的外表)。

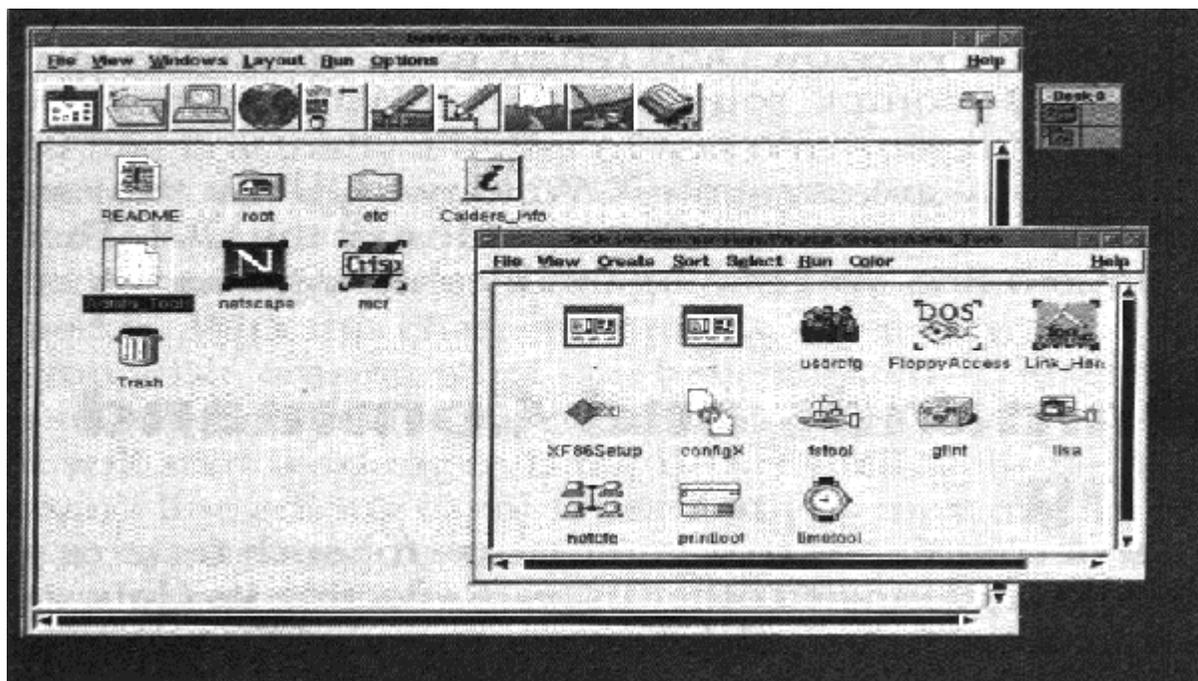


图 3-1 Caldera 工作平台

Caldera 工作平台的特点就像其它窗口环境一样。可用鼠标来选中不同的图标,菜单和窗口。窗口显示信息,列举文件或打开应用程序。像所有的系统一样,可同时打开多个窗口。然而,只有一个窗口是激活的,此窗口有紫色边框,其它未被激活的窗口具有灰白边框。鼠标指针可通过窗口间移动改变和激活窗口。不必像 Windows 或 Mac OS 系统那样需击鼠标来改变激活窗口。

随着窗口重叠会产生令人费解问题。因为激活的窗口不能自动跳到前端。一个激活窗口的一部分仍被其它窗口覆盖,需点该窗口的标题栏窗口才能跳到前端。除此以外,单击窗口其它部位无效。

选择退出工作平台选项来退出 Caldera 工作平台。屏幕顶部有几个选项菜

单。一个是文件菜单,当点中其时,即显示菜单。最末一项是退出工作平台。一旦选择退出,则回到蓝色背景的窗口管理器,然后需退出窗口管理器。击屏幕任何部位,下拉窗口管理器菜单,单击窗口管理器菜单以便退出工作平台。然后出现你要选择的 Quit 工作平台子菜单,这将关闭 X-Windows。Linux 将关闭 X-Windows 界面然后出现 \$ 提示符,通过键入 Startx 命令,在任何时候,重新启动 X-Windows。记住,使用 X-Windows 界面时,你要退出帐号时,必须先退出 X-Windows 界面。你要回到命令行界面后,才能使用 Logout 或 exit 命令。

3.3 Linux 命令和命令行编辑

Linux 有一大套可以与其他用户通讯和管理编辑文件的命令。Linux 命令用命令行界面来执行。即使用一个图形用户界面,你也经常需要使用命令行界面来执行命令。Linux 命令广泛地使用选项和参数。在你的命令行要小心使用选项和参数。Linux 命令包括选项和参数的格式被显示在下面:

\$命令名,选项,参数

一个选项是一个前面带有一个短横线的字母,它可以修改命令执行的类型。例如 ls 命令可以加上 -s 选项,显示一个目录下的文件列表。-s 选项增加了每个文件的大小。在命令行中键入命令和选项。例如:

\$ls -s

一个参数是命令需要执行它的数据,在许多情况下,将是一个文件名。例如:用 man 命令显示一个特殊命令的信息。特殊命令的命令名作为其参数。用 man 命

令来显示 ls 命令,作为一个例子,显示如下:

```
$ man ls
```

这个命令行实际上是一个可编辑的文件的缓冲区,按回车以前,编辑命令。这个编辑过程包括了修改敲错的命令及选项的能力。BACKSPACE 和 DEL 键,允许你删除命令行的字母。用这种方法来删除已经打印出来的整行命令。

工作在其中的 Shell 是 BASH Shell,它是你的缺省 Shell。此 Shell 有特殊的命令行编辑能力,你会发现在你学习 Linux 过程中,它是有帮助的,你可以方便的修改命令,在命令行的任意部位插入和删除字符。对复杂的命令来说,它有特殊的帮助。然后你可以用 CTRL-F 键或右移键向前移动一个字符。也可以用 CTRL-B 键或移键向后移动一个字符。CTRL-D 和 DEL 键能删除光标所在的字符。CTRL-H 键或 BACKSPACE 键可以删除光标前面的字符。增加文本就用光标移动键把光标移动到你想插入文本的地方。然后打印新的字符,在任何时候,你可以按回车键执行这个命令,例如,如果你在键入一个命令时犯了拼写错误,你可以用它来改正错误。你也能用键去重新显示先前执行过的命令。你能再执行这个命令或者编辑它。当你重复这些操作的时候,你会发现这个工具是有帮助的,当键入错误的命令时,它也是很有帮助的,在这种情况下,你被提示一个错误信息和新的命令行。按光标上移键可以显示先前的命令,修改它,然后再执行。

BASHSHELL 程序保持一个叫做历史记录列表。上边列出先前的列印命令。通过键入光标上移键来显示在命令行使用过的命令,光标下移键可以关闭这个列表。你可以修改和执行任何先前显示和执行命令。第 15 章中详述了历史记录的特点。

3.4 网络在线帮助

Linux 系统有编辑器和说明等使用工具。这些工具有它们自己的界面和一套命令，man 在线帮助允许显示任何 Linux 命令和程序的信息，在命令 man 后，打印你想要显示的命令名，在下面的例子中，用户激活了 ls 命令的在线帮助信息。

```
$man ls
```

在一个命令的 man 文件可以被组织或不同的层次，从其中的一个启动。第一个层次给出命令的基本信息。另一个层次提供了更多的信息。例如：用 8 个层次来显示 crontab 的帮助文件

```
$man 8 crontab
```

无论 man 什么层次产生的文件，其他层次的文件将列印在后面。大部分的文件只有一个用 man 命令产生的层次。

按回车键后，你就进入了 man 工具，显示了 ls 文件的第一页，man 有它自己的一套命令，通常由单个键组成，按空格键和 F 键将翻到下一页，按 B 键右翻到上一页。按 Q 键退出 man 工具，回到命令行。

一个说明选项有几部分组成。普通的五个部分是摘要，描述，选择，文件和对命令的参考。摘要表了命令的语法。打印它可能的选项和参。命令的描述确切地告诉你系统在做什么。接下来选项被显示并解释。文件部分显示了命令使用的系统文件。交叉的参考文件显示了其它相关的命令和帮助选项。下面上 ls 命令选项的一个缩短的版本。

NAME

ls, dir, vdir - list contents of directories

SYNOPSIS

```
ls [-abcdfgiklmnpqrstuxABCFGLNQRSUX1] [-w cols] [-T cols]
[-I pattern] [--all] [--directory] [--inode][--kilobytes]
[--no-group] [--hide-control-chars] [--reverse] [--size]
[--width=cols][--sort={none,time,size,extension}]
```

DESCRIPTION

This manual page documents the GNU version of ls. dir and vdir are versions of ls with different default output formats. These programs list each given file or directory name. Directory contents are sorted alphabetically. For ls, files are by default listed in columns, sorted vertically, if the standard output is a terminal; otherwise they are listed one per line. For dir, files are by default listed in columns, sorted vertically. For vdir, files are by default listed in long format.

OPTIONS

- a, --all
List all files in directories, including all files that start with '.'.
- b, --escape
Quote nongraphic characters in file names using alphabetic and octal backslash sequences like those used in C.
- c, --time=ctime, --time=status
Sort directory contents according to the files'

```
status change time instead of the modification
time.  If the long listing format is being used,
print the status change time instead of the modifi-
cation time.
-d, --directory
    List directories like other files, rather than
    listing their contents.
-f
    Do not sort directory contents; list them in what-
    ever order they are stored on the disk.  The same
    as enabling -a and -U and disabling -l, -s, and -t.
--full-time
    List times in full, rather than using the standard
```

FSF

GNU File Utilities

1

man 工具还有几种有益的特征,例如:它有搜索功能,通过按/或者?来激活一个搜索。/代表向前搜索。?代表向后搜索,按下/后,在你的屏幕底下将会出现一条横线,然后你键入你所搜索的词。键入回车,激活搜索。搜索其实是一种模式搜索,所以你可以键入词的一部分或词的几个字符,通过按下 N 键来重复同样的搜索而不必重新键入模式。

3.4.1 Whatis 和 apropos

Whatis 和 apropos 将收录 man 题目的数据库,并根据每一个 Whatis 的短小描述显示结果。

Whatis 将整句话寻找 man 标题。例如:如果想寻找带有字母 X 的帮助选项,

应使用下列命令(为给也了处理 X-Windows 主题的帮助目录)

```
$ whatis X
X (3) - a portable, network-transparent window
system
X Consortium (3) - X Consortium information
X Standards (3) - X Consortium Standards
X security (3) - X display access control
X (3) - a portable, network-transparent window system
X Consortium (3) - X Consortium information
X Standards (3) - X Consortium Standards

X security (3) - X display access control
(END)
$
```

Whatis 和 apropos 使你进入一个 man 类型的界面。如果结果超过一页,可用 F 和 B 键进行前后移动。可以用 / 和 ? 键执行模式搜索,用 Q 键退出,回到命令行。

Apropos 命令执行了 Whatis 相同的任务,但其是通过模式来搜索而非整句搜索。例如: Apropos x 命令将产生几页长的结果,列出所以以 x 打头的 man 选项,例如 xwpe 和 Xloadimage。

下面例子中,用户列出了全部以 ls 打头的所有帮助选项。这包括了 ls 命令,以及其它的例如 lseek 或 lsearch 命令。

```

$ apropos ls
ls, dir, vdir (1)      - list contents of directories
lsattr (1)             - list file attributes on a Linux second
extended file system
lsearch (n)           - See if a list contains a particular element
lseek (2)             - reposition read/write file offset
lsort (n)             - Sort the elements of a list
lsattr (1)            - list file attributes on a Linux second
extended file system
lsearch (n)           - See if a list contains a particular element
lseek (2)             - reposition read/write file offset
lsort (n)             - Sort the elements of a list
(END)
$

```

你也能使用 Helptool 工具寻找 Linux 帮助文件中的关键字。Helptool 能搜索所有的信息页或者文本文件。只要键入 helptool 从显示出来的主题列表中选择就可以了。

3.4.2 Xman 和 tkman

如果你在工作平台上工作,就要使用 Xman 程序来搜索和显示帮助文件。Xman 是 X-Windows 为基础的程序,用鼠标单击扫纽和菜单寻找和显示帮助文件。单击 fvwm 工具栏中的 Xman 按钮,可以启动 Xman。屏幕出现带有二个按钮的小窗口,分别是 Help, Quit 和 Manual Page。单击 Manual Page 文件能够

打开 Xman 显示窗口,顶部出现两个菜单按钮。Options 和 Sections。在 Options 菜单中你可让选择寻找选项,开启一个搜索窗口,在这里键入想要显示的命令或者主题,你可以用 Sections 菜单来设置命令。这个菜单列出了用户命令,系统管理,设备和游戏等几项,选择一个用 man 项目显示的面板。例如 User 命令选项将列印所有的用户命令。然后,单击其中的一个和来显示其命令的 man 文件。你用左侧的滚动条浏览 man 文件(图 3-2 表明了 man 文件的窗口和最初的 Xman 窗口)。

Tkman 提供了 Xman 文件同样的特点,并且有一个容易使用的界面,在窗口的顶部有一个方块,你可以键入一个文件名,然后单击标有 man 的按钮。对于 apropos 寻找来说,在相同的方块内,键入要寻找的字符串,单击标有 apropos 的按钮,Volumes 菜单允许你轻松地选择主题。Sections 菜单允许你设置选项并提供帮助。一个滚动条允许你浏览 man 文件。通过选择在应用程序菜单中的 "Other Applications" 另的 Tkman 项目或者从一个终端窗口里使用 Tkman 命令来启动 Tkman。

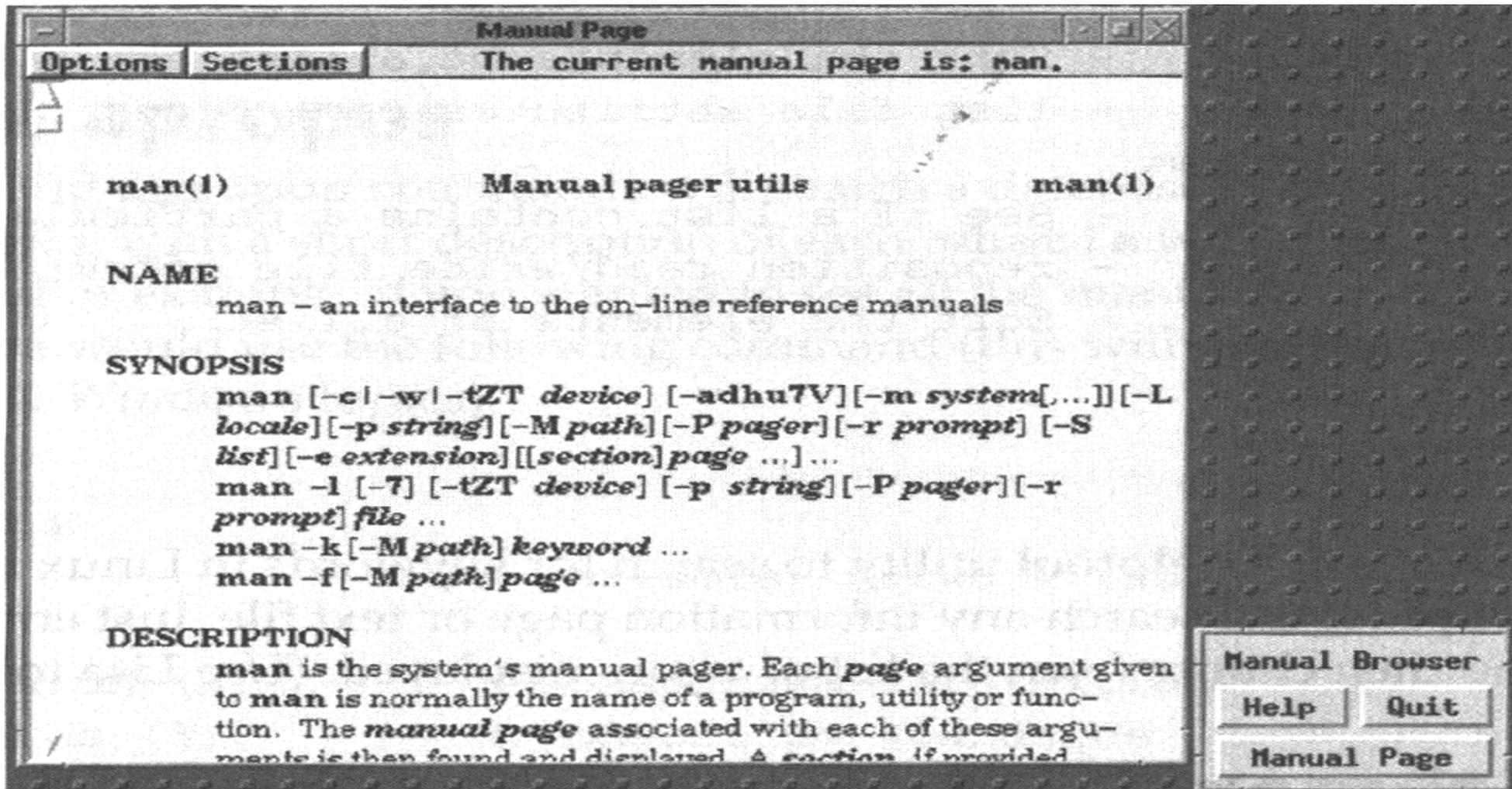


图 3-2 显示 man 文件的 Xman 窗口

3.5 网络在线文件

当启动你的浏览器时,一个缺省的网页连接着你的系统和 Caldera 和 Redhat 网址,要想使用这两个网址,首先你必须要与 Internet 相连。但是你的光盘和系统中已经包含了广泛的文件。这些文件是关于怎样使用工作平台来 Linux 系统的详细解释,包括 Vi 编辑器和 Shell 操作。这里列出了与这些文件的连接。在不同的 Linux 主题上,其他的文件提供了详细说明。

- 工作平台界面的用户指导。
- Linux 的安装和启动指导。
- Crisplite 编辑器的用户指导。
- 大众的监听器列表 Bugs。
- 其他的 linux 文件。

/usr/doc 目录包含了许多 Linux 应用程序的在线文件。一些带有安装 Linux 应用程序名字的子目录包含了像 README 文件。在 /usr/doc/HOWTO 目录下,你可以获得整套的 HOWTO 文本文件。HOWTO 文件包含了详细的硬件安装和网络配置的 Linux 的主题。另外,/doc/ HOWTO/HTML 目录包含了可用网络浏览器显示的网页格式文件(使用 Lisa 安装 HOWTO html 软件包)。

也有一些像 gcc 编译器和 Emacs 编译器等的 GNU 应用程序。你可以通过键入命令来获得这个文件,这将出现带有不同 GNU 应用程序的特殊屏幕。info 界面有它自己的整套命令,你可通过键入 info 来学到更多的东西。在屏幕的底部的一行内键入 m,它代表了某个应用程序的另一个字母。回车就可在应用程序

上得到 info 文件。

3.6 配置你的系统:用 Lisa 设置

如果你是 Lisa 系统管理员,就需要执行一些例如创建一些新帐户和新软件等的管理任务。用叫做 root 的系统管理帐号来执行。作为一名超级用户,你可以完全控制你的系统,并可以任意的改变他。你能够安装软件包,创建新用户,给你的系统增加新设备或者是磁盘,配置 X-Windows 或网络界面。

执行使用 Linux 安装和系统管理程序,可以执持管理任务, Lisa 提供了容易使用的能够管理任务的界面,可获得一套 REDHAT 管理工具。根据系统的管理任务,这些将被详述在第 7,19 和 20 章,但是用 Lisa 你能执行许多基本的任务。Lisa 有一个带有详细提示的全屏界面,帮助你完成象安装打印机或者安装新分区的任务。Lisa 有一个详细的在线描述,你可以通过 www.caldera.com 网址来获得它,单击在线文件,选择 OpenLinux 为基础或 OpenLinux 标准的选项,然后选择管理 OpenLinux 的章节。

要使用 Lisa ,你必须要以超级用户的身份注册。安装期间你已经确定了超级用户的口令。这个口令就是在超级用户注册时使用。如果其他的人正在用你的 Linux 系统。要注意保密你的超级口令。任何以超级用户注册的人,都可破坏你整个的 Linux 系统。在注册提示符下,键入 root,就可以以超级用户的身注册,然后键入超级用户口令。

Caldera OpenLinux(TM)

Lite 1

Version 1.1

Copyright(c) 1996-1997 Caldera Inc.

turtle.trek.com login: root

Password:

运行 Lisa 最容易的方式是在命令行键入 Lisa 命令。你也可以通过工作平台来运行它。但必须先用 startx 命令来启动工作平台。一旦注册后,系统将提示超级用户提示符#。注意它其与普通用户提示符的区别。

Lisa

Lisa 最初显示了带有 4 个选项的菜单,复杂地系统分析,系统配置软件包管理和帮助系统。要使用文件管理,设备配置或增加新用户的系统配置,软件包管理让你容易地增加和软件包。这个帮助系统将列出在的系统上可获得的 HOW-TO 文件。如果你运行时出现困难,你就可以参阅恰当地 HOW-TO 文件。复杂系统分析提供了当前系统如何配置的信息,第 2 章图 2-5 表明了最初的 Lisa 菜单。大部分的 Lisa 屏幕显示了底部的按钮。一个 Call 按钮将启动一个菜单和选择,Continue 按钮将离开这一屏转到前面的一屏,在主菜单的情况下,就会退出程序如果有,两个按钮,一个被选择了,另一个就不会被选择。然后按回车键 来选择已被选中的一个。

你用向右或向左的光标来选择另一个(你也可台用 TAB 键)。按右移键选择右侧的按钮,按左移键选择左侧的按钮,然后按回车键。Lisa 也提供了在线帮助,在任何情况下,按 F1 键会出现一个当前选项的解释。Continue 按钮或者 ESC 键使你退回前面的屏幕。连续地选择 Continue 键就会回到主菜单。

选择系统配置的选项进行执行管理任务 (见图 3-3)会出现不同面积的选择列表,然后你可以从硬件系统网络或者启动管理器配置中选择,网络选项允许你执行网络配置,硬件管理允许你增加象打印机一样的新设备。系统配置允许你执行例如增加新用户或者安装分区等任务。

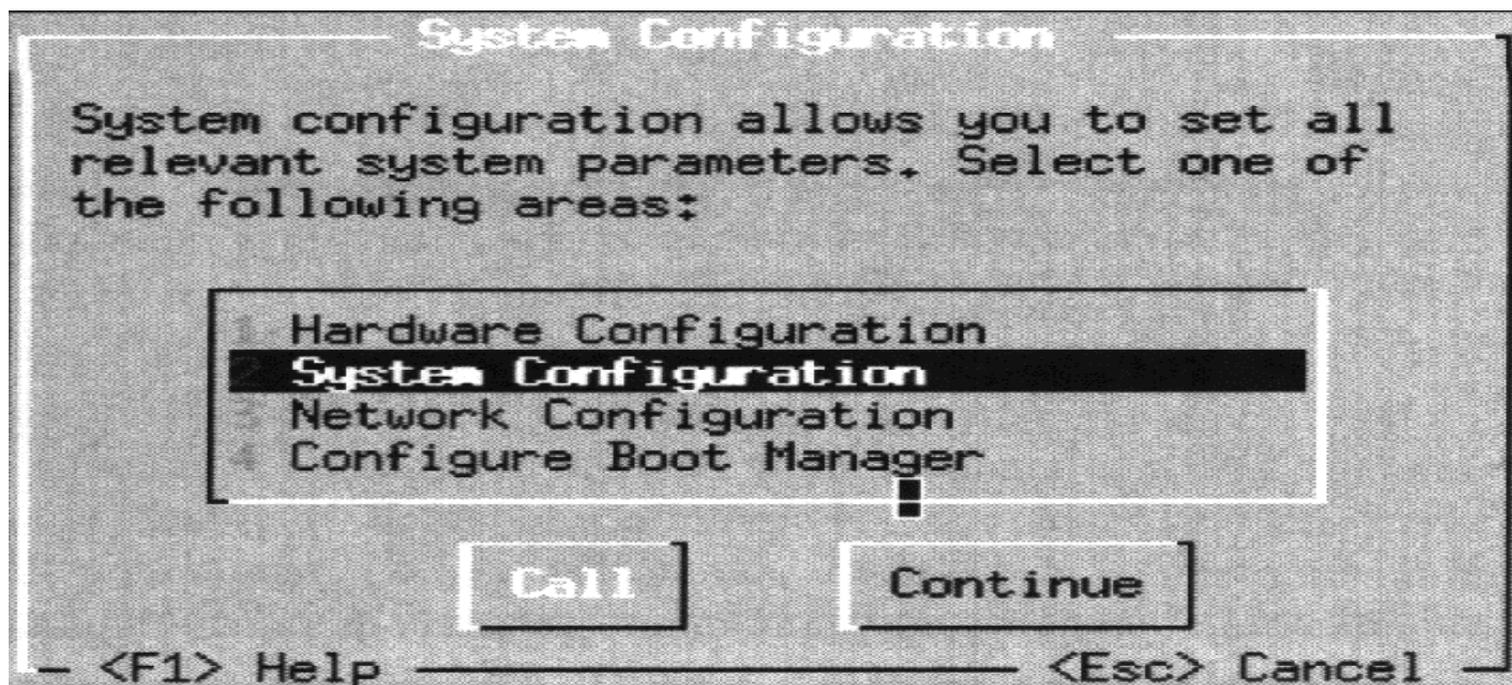


图 3-3 系统配置领域

系统管理选择菜单中的系统配置一项,一个带有选择项目的屏幕将会出现,当系统启动的时候配置安装表选项允许指定分区和设备。你的主要的 Linux 分区和交换分区就被键入。第 7 章详述了分区的解释和如何安装它们。

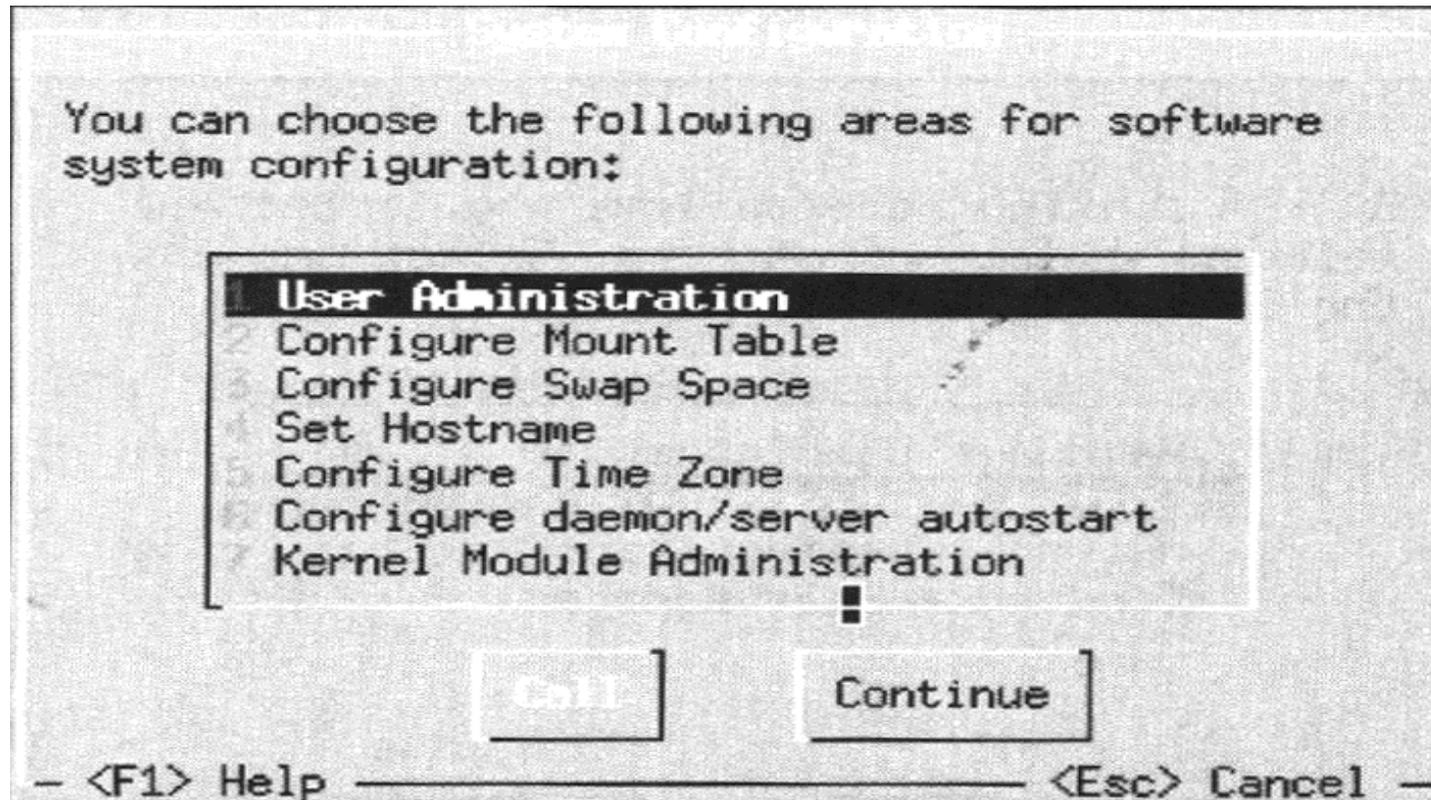


图 3-4 Lisa 系统配置

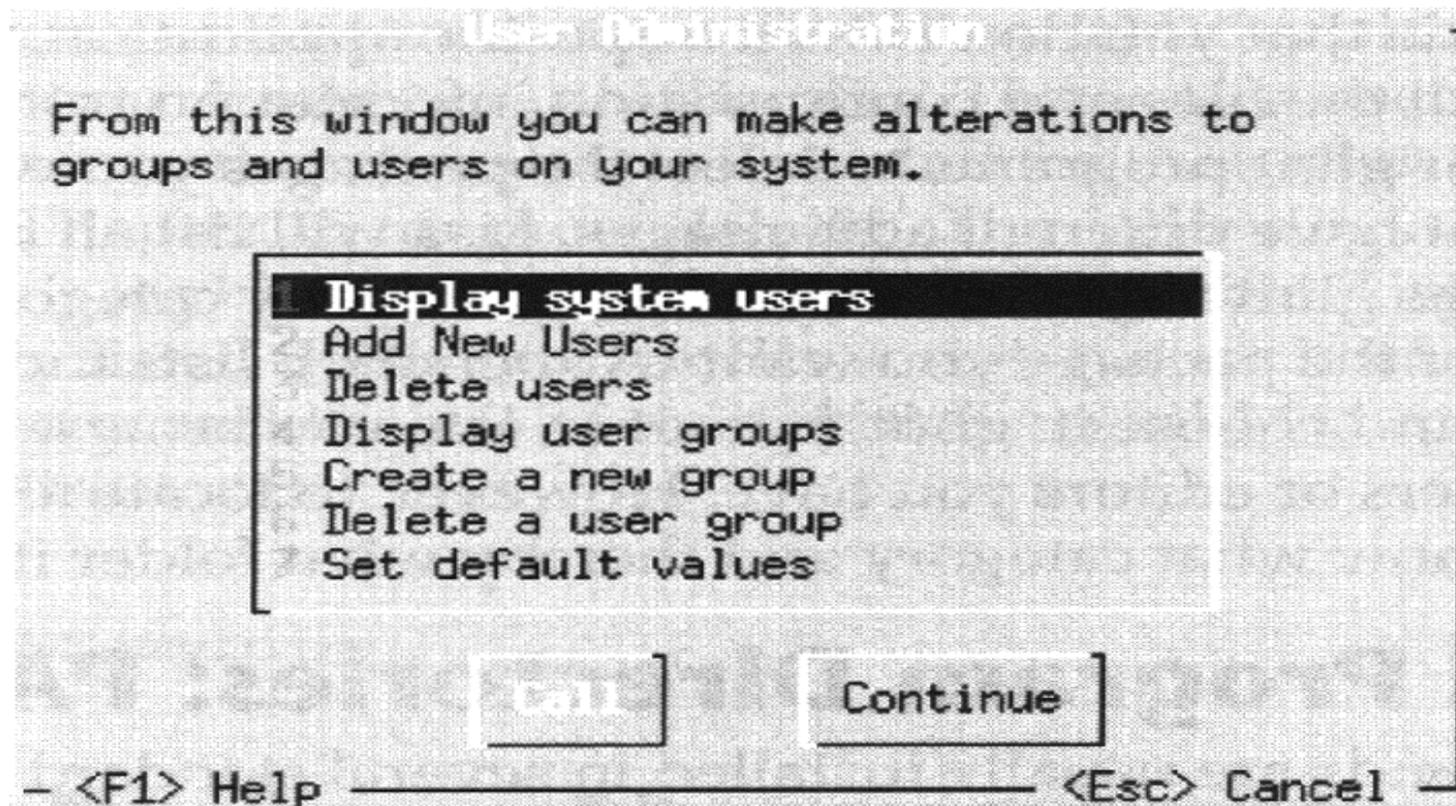


图 3-5 Lisa 用户管理系统

用户管理建立了一个带有管理用户和主群的选项的菜单,(见图 3-5)选择"添加新用户"选项来增加一个新的用户,一系列的提示让你键入注册名和 ID 号(100-64000 之间的任意数),一个主名,一个 home 目录,一个启动 shell(通常是 BASH 和口令)选择。"创建新主群"选项可以增加一个新主群。

硬件管理菜单列出一系列硬件,例如调制解调器,打印机,硬盘和鼠标以及 XFree86 服务器。选择你想要配置文件的选项系统将列出一系列可供选择的硬件型号。例如对于 CD-ROM,你将从不同的 CD-ROM 类型中做选择,XFree86 服

务器选项允许你选择不同的服务器。这样有利于你改变你的显示卡或服务器。

从工作台中有几种方式可以开启 Lisa 工具。首先通过键入 `startx` 命令来启动工作平台。在工作平台中,你可以击 `fvwm` 工具栏中的 `Admin` 按钮。这将产生一个具有 Lisa 按钮的小的按钮栏。你也可以用 `frwm` 程序菜单,选择在系统管理菜单下面的 Lisa,这是应用菜单的一部分。最后你可以在 Caldera 工作平台上选择 `Admin` 系统文件夹。

3.7 安装软件包

现在你知道如何启动 Linux 和 进入超级用户帐号,你就可以安装你想要的软件包。安装软件包是由超级用户执行的一种管理功能。在安装过程中,除非你选择完全安装,否则,在你的系统中,能安装部分的工具。附录 B 包含了关于 OpenLinux CD-ROM 的系列软件包。

有几种安装软件的方式。你可以使用工作平台中被称作 `glint` 的图形界面。也可以使用工作平台中或命令行中的 Lisa 工具。这些是安装几个软件包的最简便最有效的方法。可以用命令行中 `rpm` 命令直接激活 Redhat 软件包管理器 (RPM)。然而这是一个非常复杂的操作。最后,你可从网上下载软件,然后用 `rpm` 命令安装它们。这只是简单地操作描述,在第 7 章将有更详尽的描述。

Linux 的 Caldera 和 Redhat 两个系统都用 Redhat 软件包管理器 (RPM)。将你要安装或删除的软件装进软件包。一个 RPM 软件包象它自己的软件应用安装程序一样运行。一个 Linux 软件安装程序通常由安装在不同目录直的几个

文件组成。程序本身最可能放在 `/usr/bin` 的目录下,网络帮助文件将放在另一个目录。库文件在另外一个目录中,另外安装可能要求系统中配置文件的修改。RPM 软件包将为你执行所有任务。如果你决定删除一个具体的应用程序,你可以卸载软件包,删除系统上所有的文件和配置信息。使用 `Lisa` 或者 `glint` 程序选择你计划安装或者删除的软件包。`Lisa` 或 `glint` 有不同的优点,然后 `Lisa` 按字母列出所有的软件包,而 `glint` 把它们放在分类的文件夹中。如果你知道你需要安装软件包的名字,`Lisa` 是非常有帮助的(附录 B 列印了这样软件包)你只需浏览这个列表,并选择它,但是如果设置特殊的软件包,你必须知道它的和所在的文件。

3.7.1 命令和程序目录:路径

程序和命令通常安装在标准系统目录下的,例如 `/bin/usr/bin`,`/usr/X11R6/bin` 或 `/usr/local/bin`。但是,一些软件包将把它的软件包放置在标准目录下的子目录中,或者一个全部分开的目录,在某种情况下,你不能运行这些命令,因为你的系统不能在新的子目录上安放它们。你的系统包含了一系列的用于命令查询的目录。这套目录被放在一个叫作 `PATH` 系统中,它的名字可更改,启动系统时,它就会被创建。如果一个命令不在列表中的目录里,那么你的系统将不能设置和运行它,为了使用一些命令,你必须首先在 `PATH` 目录系统里增加新的目录。

在 `OpenLinux` 系统中,路径被目录系统分配到 `/etc/profile` 文件下,这个文件是一个文本文件。当你的系统启动和配置用户工作环境时运行它。在这个文件中,你将发现以 `PATH=` 为开头的一行,并且其后跟着一系列目录用冒号分开,许多目

录包含命令和程序。

用 GropLite.XEmacs.Emacs 或 Vi 等编辑器编辑 /etc/profile 文件,用来创建新目录。在一系目录末端,在双引号添加一个带有全部路径名的目录。确认新目录与最后一个用冒号隔开。在最后亦应有一冒号。例如:如果安装 MH 邮件工具,在 |usr|bin 目录下的 mh 子目录安装 MH 命令。该目录的全部路径名是 /usr/bin/mh。应该在一系列目录中加入表明/etc/profile 文本路径内的目录(软件包名命令 rpm-qp)将列出安装 RPM 软件包的全部命令)下面的例子表明随目录系列的变化路径亦是变化的,和目录 /usr/bin/mh 的添加(将在下栏中显示)。

```
PATH="/bin:/usr/bin:/opt/bin:/usr/X11R6/bin:/usr/openwin/bin:/usr/  
local/bin:/usr/bin/mh:"
```

/etc/profile 文件是为用户注册时使用的可执行文本。个别用户可通过用 :bashrc 或 .profile 文件设置 PATH。用这使方法,用户可以在自己的目录下创建和安装所要使用的命令和程序,并能顺利地使用它们(详见 15 章)在 .profile 文件里的选项中,有一个是在 PATH 里增加一个用户的 mybin 目录,用 \$PATH 将列出 /etc/profile 文件中的全部目录。不要忘记在新目录前放一个冒号,并用 \$HOME 来确定用户的 home 目录的录径名。

```
PATH=$PATH:$HOME/mybin:
```

3.7.2 Lisa:软件包管理

在 Lisa 主菜单下选择软件包管理。会出现四个选择:第一个是安装程序来源的选择。要从 Linux 光盘上增加软件,就必须先通知 Lisa 软件正放在光盘上。选

择这一选项,一系列设备会显示出来,其中 CD-ROM 设备选项是发亮的,按回车即可选择它,Lisa 将检测光盘并要求确认,按回车键转到软件包菜单,用光盘移动选择增加软件包选项,按回车,Lisa 会显示全部未安装的软件包,这可能花一点时间。

每个选项的左侧有一个空的括号,接下来是用黄色显示的软件包的文件,右侧是对软件的描述。用光标移动键选中一个软件包,然后按空格键确认选中。你会看到括号中出现 x。如果你又不想选中它,就再按一次空格键,x 会消失。如果你确定选中,按回车,你选择的全部软件包会被安装。图 3-6 显示了 Lisa 的软件包列表。也可以用 Tab 键来选中 Cancel 按钮再回车,以便取消安装。

选择“删除软件包”选项可以删除已被安装的软件包,选中后,屏幕显示所有被安装的软件包。同样的方法,可以选中并回车。你可以删除任何一个软件包。

3.7.3 Linux 安装管理器 glint

通过 glint 使工具的用户友好界面安装软件是很简单的,用鼠标就可以完成。glint 可以自动把软件或专用配置文件复制到正确的目录下,glint 能在增加和删除软件包过程中处理,全部的文件。

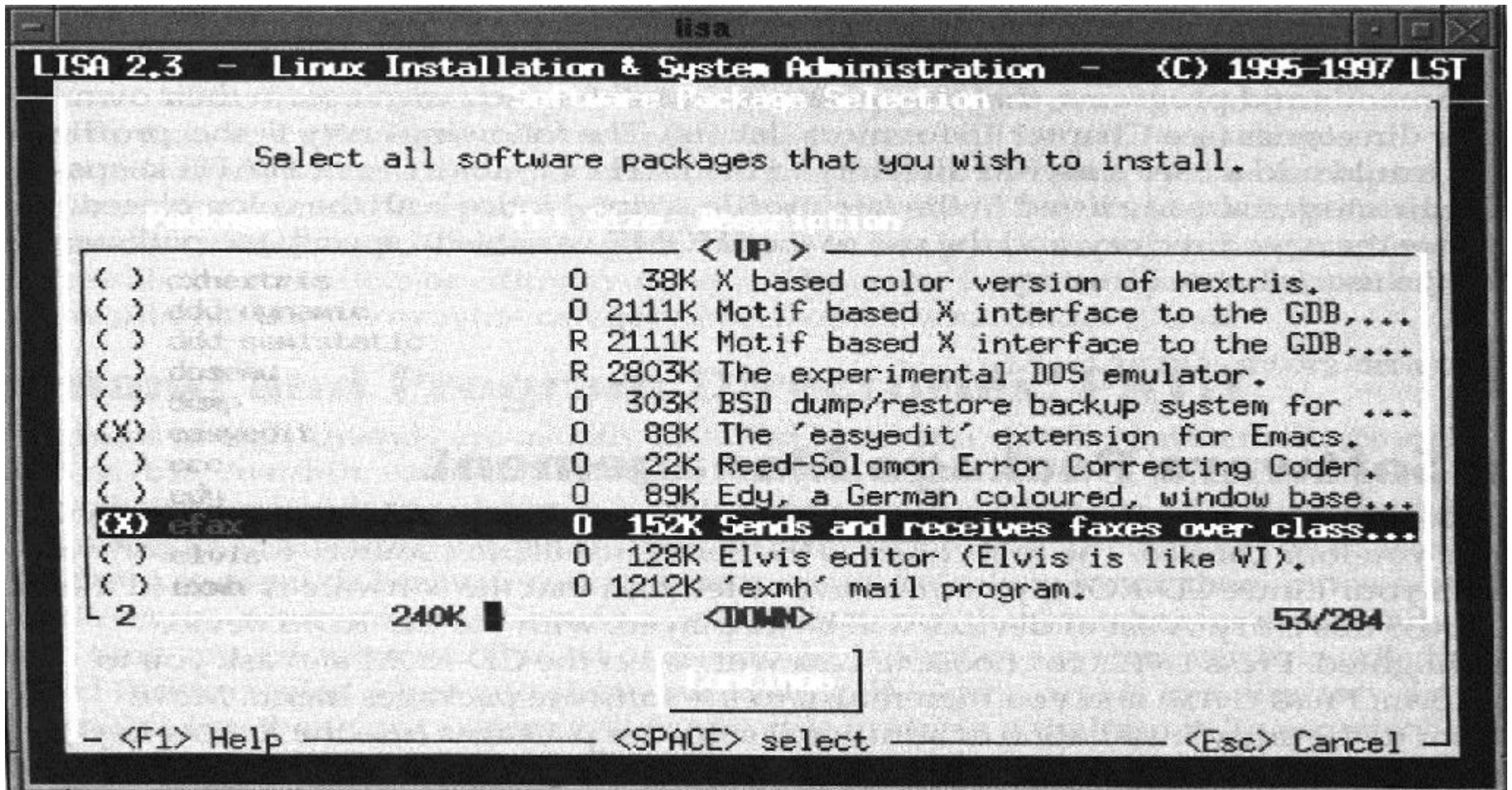


图 3-6 Lisa 安装列表

像使用 Lisa 一样,你首先必须用超级用户身份来注册,如果从光盘上安装软件,首先要使系统识别光驱。不像 Lisa,glint 不能自动完成这一点,你必须执行一

个被称作安装一个系统文件的命令,首先把光盘放入光驱,然后键入下面的命令,这使光驱上的一个盘区与 Linux 系统相通。安装文件系统将在第 7 章详细讨论。

```
$mount/mnt/cdrom
```

glint 是一个从工作台上运行的 X-Windows 程序。键入 Startx 来起动工作平台,用开启 Lisa 工具相同的方法来开启 glint,选择 fwm 工具栏中的 Admin 按钮,它在应用程序,菜单中,系统管理下的 fwm 程序菜单中。Admin-Tools 文件夹放在 Caldera 工作平台中,一旦你开启了 glint 图 3-17 中的 glint 窗口就会显示出来,glint 看起来就像一个文件管理器窗口,它显示出应用程序和文件夹图标,同类的应用程序放在相同的文件夹中,例如,Emacs 编辑器放在叫做编辑器的文件夹中。此文件夹放在应用程序文件夹中。

在你已经看到的文件夹和图标是已经被安装在系统中的(只图 3-7)你可以通过窗口来阅读这些软件包或删除它们。只要选中适合的文件夹中的图标,然后按右侧的拆卸按钮(Uninstall)。当你选中了一个软件包,它的图标就会出现红色边框。你可以通过单击它们同时选中几个图标。一次就可把它们全部删除。要想取消已经选中的图标,只要再单击它一次就可以了。你会看到红色的边框消失。

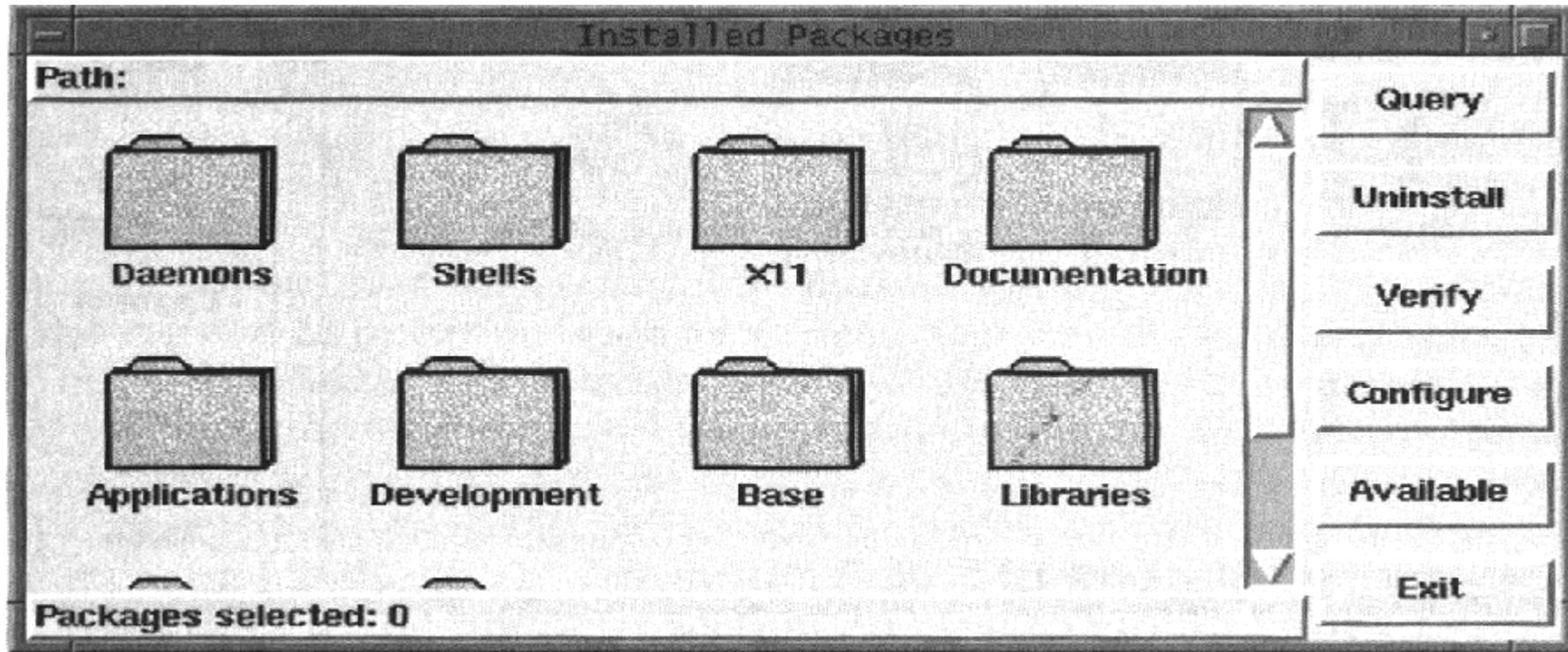


图 3-7glint 使用工具

当你打开并显示一个文件夹的内部时,会看到一个带有大黑箭头名字叫"Back"的图标,通过点中它,可以退回先前文件夹,也就是关闭当前文件夹。例如,双击应用程序文件夹,像游戏等的文件夹会出现,同时会出现"BACK"黑箭头图标,打开游戏文件夹。会看到一些游戏的图标和一个"BACK"大黑箭头图标。单击大黑箭头图标,可退回到前一个文件夹。直到退回应用程序文件夹。通过用鼠标击中文件夹和黑箭头图标,就可以自由的打开和关闭文件夹来使用其中的文件了。

要想安装新的软件包、首先选择 glint 窗口右侧的"Available"按钮。这将打

开一个标有"Available Packages"的一个 glint 新窗口(见图 3-8),Caldera 光盘中未安装的软件包将会以文件夹和图标的方式显示在这个窗口中。它们被分别放在适合的文件夹中。打开文件夹可以把它们显示出来。单击打算安装的软件包的图标,按下窗口右侧的"Install"按钮。软件包就会自动安装到系统上,可以同时选中个软件包的图标、把它们一次安装,当你要取消已经选中的图标时,只要用鼠标再单击一次就可以了。被选中的软件包的图标仍然会出现红色边框,按下"Install"按钮后,这个图标就会从"Available packages"窗口下消失。并出现在它被安装的窗口。

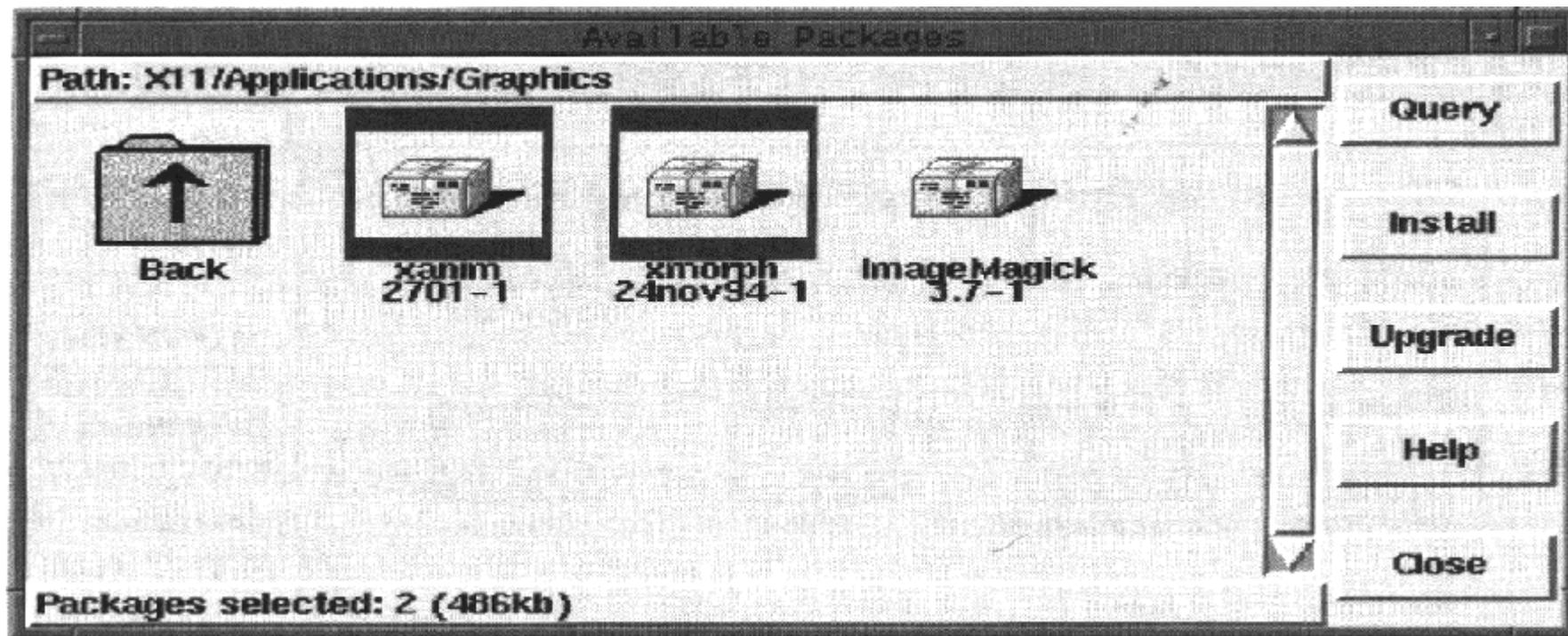


图 3-8 glint 的 Available Packages 窗口

在 glint 窗口中选择任何文件的图标并按下 "Info" 窗口, 就可以获得文件的简单描述, 它们试图打开 "Available Packages" 窗口时, 收到了没有可安装的 PRMS 文件 (并且不能选择完全安装选项) 时, 最大的可能是你没有安装 Open Linux 光盘驱动器。你需要打开终端窗口, 在其中键入 mount 命令, 单击 fwm 任务栏中的 Xterm 按钮。然后在终端窗口的提示符下键入 mount/mnt/cdom 命令。

3.7.4 命令行安装: Redhat Package Manager

如果你不使用工作平台, 可以在命令行用 rpm 命令来管理和安装软件包, rpm 是 Redhat 软件包管理的行命令, 它通常执行软件包的安装, 删除和确认。实际上 Lisa 和 glint 也是使用 rpm 命令来安装和删除软件包, 每个软件包通常是一个由包含软件文件和关于如何安装这些文件的信息文档组成的 RPM 软件包。每个文档驻留在一个带有 .rpm 文件名的文件中, 表明它是一个可被 Redhat 软件包管理器安装的软件包。

你能用 rpm 命令来安装和删除软件包。rpm 使用了一套决定完成各种功能的选项。表 3-1 列出了这套 rpm 选项。-i 选项将安装指定的软件包, -e 选项将删除软件包, -qi 选项将访问系统是否安装了某个软件包并显示此软件包的信息 (-qpi 选项将询问一个未安装的软件包文件)。--h 选项列出了全套的 rpm 选项。--force 选项是用来进行无条件的强制安装。下面是 rpm 命令的语法 (rpm-package-name 是你要安装的软件包的名子)。

```
rpm options rpm-package-name
```

软件包的名字通常是非常长, 包含了发行日期和版本, 所有的都是从 .rpm 结尾,

在光盘 Packages/RPMS 目录下是一套标准的 Linux 应用程序。在 OpenLinux 目录以及子目录下,驻留了像 Xapps1 和 develp1 等类型的应用程序。一些软件包通常被发现在名叫 RPMS 的子目录下。像 Java Development Kit 和 Xfm 的专用程序,放在 OpenLinux/Contrib 目录下。从 Caldera 光驱上安装软件包。先进入适当的 RPM 目录,再安装,你想要的软件包,如果你不能进入此目录,可能是因为你的系统可能没有安装光盘驱动器,如果是这样,参考前面部分如何安装你的光驱 (mount/mnt/cdrom),例如,要想出入包含 Java Development Kit(JDK)的子目录,键入如下命令。

```
$ cd/mnt/cdrom/openLinux/contrib/RPMS
```

Ls 命令是列出了全部的软件包,如果你知道一个软件包的开头一个或几个字母,可以用 Ls 命令和匹配等来显示它,文件包的列表通常很长超出一个屏幕,用通配符 * 和 Ls 命令显示软件包的详细名字是很有效果。下面的例子表明了如何列出 X-Windows 软件包。

```
$ ls x*
```

下面的例子中,用 rpm 命令安装 JDK 软件包,注意:因为要键入文件的全名,可以用 Ls 命令显示 JDK * 来查找全名。

```
$rpm -i JDK_static_1.0.2.p 12-3.i386.rpm
```

安装和载卸软件包之间并没有什么不同,当你想安装一个软件包时,必须知道它是否已被安装。可以用 rpm 命令的 -qi 选项来安装一个软件包。下面的例子中,用户可以来确认 JDK 是否已被安装在系统中,注意不用键入 JDK 文件名全名。如果已经被安装,文件的名称及安装的目录已经注册在系统中。

```
$ rpm -ql JDK_static
```

用 `-qpi` 选项来查看已被安装的软件包的目录及软件包里的所有文件。下面的例子是列出 JDK 文件包中的所有文件。

```
$ rpm -e JDK_static
```

要删除一个系统中的软件包,首先用 `rpm-qpi` 来确定此文件包的确已被安装在系统中,然后,用 `-e` 选项来删除它,用 `-qpi` 选项,不必知道文件的全名,只要知道应用程序的名字就可以了。在下面的例子,是用户删除系统中的 `JDK_static` 文件。

用 `-qpl` 选项来查看已被安装的软件包的目录及软件包里的所有文件。下面的例子是列出 JDK 文件包中的所有文件。

```
$ rpm -e JDK_static
```

联机帮助中说明了 `rpm` 命令的全部功能

```
$ man rpm
```

3.7.5 安装网络资源中的软件

从网上安装软件是很复杂的,你必须先登录到远端的资源位置。你需要知道软件安放的地点,然后用 `Netscape` 或 `ftp` 工具来下载软件包。如果正和一个 Internet 相连,首先必须使用 `pppd`(表 3-2 列出了 RPM 软件包的几个 `ftp` 网址)。有些软件包是被压缩的文件,并不是 RPM 格式,你可以用 `gunzip` 和 `tar` 命令来打开压缩文件再安装它。第 7 章详述了这一过程,所有的 RPM 软件包都是从 `.rpm` 结尾的,当下载这样的软件包时,必须用 `rpm` 命令来安装它。最好使用 RPM 格式的软件包。

通常,一个软件应用程序是一个 RPM 格式的软件包:或者对一些系统上没有

Redhat 软件包管理器的系统来说是一个 tar 文档,许多 RPM 软件包是放 Redhattp 网址:ftp.redhat.com 下/pub/contrib/i386 目录中。你可以轻松地从这个网址下载软件并且安装它,从其它的 Linux 相关信息网址如 Snnsite 和 tsx-11 中可获得一些压缩的文件和 RPM 软件包,在 ftp.caldera.com 网址中的 /pub/netscape/narigator 目录下,可以找到为 Linux 系统使用并且是 RPM 格式的 Netscape 浏览器版本。或在 Netscape 的 .ftp 网址(ftp 5.netscape.com)下找到这文件的压缩版本。

下面的例子是用 ftp 连接 Linux ftp 网址 ftp.caldera.com 来下载 Netscape 网络浏览器,用户键入一个 anonymous 作为一个注册 ID 和一个 Internet 地址作为一个口令,经常,一个映射网址列表将被显示。映射网址包含了那个 ftp 网址中的所有文件,当由于 ftp 网址太繁快以至于不允许进入或太慢时,可以试着进入一个映射网址。

```
# ftp ftp.caldera.com
Connected to rim.caldera.com.
220 rim.caldera.com FTP server (Version wu-2.4.2-academ[BETA-12](1) Wed Feb
19 03:54:33 MET 1997) ready.
Name (ftp.caldera.com:root): anonymous
331 Guest login ok, send your complete e-mail address as password.
Password: Enter-your-internet-address
230 Guest login ok, access restrictions apply.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd pub/netscape/navigator
250 CWD command successful.
ftp> cd 3.01
250-Please read the file README.license.txt
250- it was last modified on Tue Aug 19 19:47:00 1997 - 5 days ago
250 CWD command successful.
ftp> ls net*
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
total 2313
-r--r--r-- 1 root root 2336089 Aug 17 03:10
netscape-3.01-6.i386.rpm
226 Transfer complete.
ftp> get netscape-3.01-6.i386.rpm
local: netscape-3.01-6.i386.rpm remote: netscape-3.01-6.i386.rpm
200 PORT command successful.
150 Opening BINARY mode data connection for netscape-3.01-6.i386.rpm
(2336089 bytes).
226 Transfer complete.
2336089 bytes received in 711 secs (3.2 Kbytes/sec)
ftp> close
>221 Goodbye.
ftp> quit
#
```

进入 ftp 网址后,用 cd 命令,用户可以转到放置 Linux Netscape 浏览器的 /pub/netscape/navigator 目录下。然后用户进入 3.01 目录找到 Navigator 的 3.01 版本,键入 Ls 命令和带有通配符的,软件包开头几个字母,显示像需要的软件包的列表。用户首先键入 Ls nets*来显示此文件包的全名,要用 ftp 下载软件包,你必须使用文件的全名。像 ftp.caldera.com 指示的那样,用户第一次注册时,大部分网址会以二进制格式自动下载文件。如果不是自动下载,你必须用 binary 命令来指定二进制格式,使用 get 命令然后下载软件包,一旦完成你所需的软件包下载任务,你就能通过用 close.quit 命令离开这个 ftp 网址。

下载后,通过 rpm-i 命令安装 RPM 软件包,被下载的文件通常包括 README 文件和其它文件。不要忘了阅读它们。学生和老师可以免费注册使用 Netscape Navigator。而其它人不是免费注册的,注册后你能够被提供技术支持和升级服务,否则,可以免费使用非注册的 Netscape Navigator。

```
$ rpm -i package-name
```

在 Netscape 软件包中,你必须使用强行安装的选项,Open Linux 系统中已经存在 /usr/bin/netscape 文件(这个文件只是不含有 Netscape),你可以用 --force 选项使带有 Netscape 程序的文件覆盖原有文件(注意,--force 选项中有两个 -)。

```
$rpm -i --force netscape -3.01 -6.i386.rpm
```

另一方面,你能够用 Netscape 来寻找浏览和下载软件,而不必使用全部 ftp 命令。在 Netscape 下,用 ftp://代替 http://来处理一个 ftp 网址名,例如 Redhat,你应键入 ftp://ftp.redhat.com.Netscape 浏览器将列出此目录下的全部文件,你可以用滚动栏来浏览。双击一个目录的名,可以转到别的目录。每个目录名字前存一个黄色文件夹的图标,例如,你必须用鼠标双击 /pub 目录,然后屏幕显示此目录的

内容。双击 contrib 文件夹,再从 contrib 文件夹中的目录和文件中,选择 i386 目录,才可能到达 /pub/contrib/i386 目录,图 3-9 是用 Netscape 显示一个 ftp 目录。

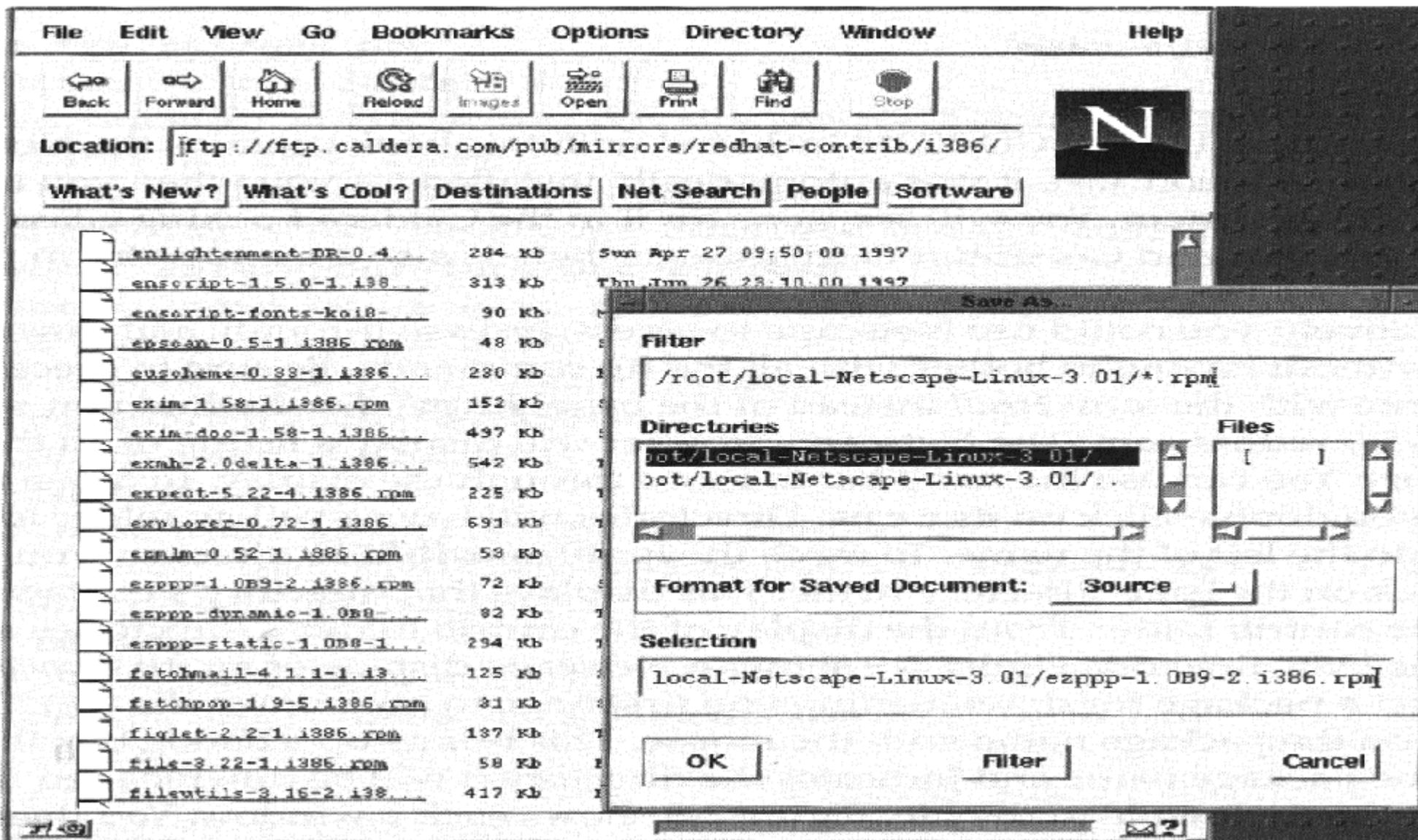


图 3-9 Netscape ftp 显示

选择一个要下载的软件包,首先必须按住 SHIFT 键,然后用鼠标来单击软件

包名字,这会出现一个对话框,显示出包名字并指出要下载到的目录名。单击 OK 按钮来下载,图 3-9 表明了这样一个窗口,然后你能用 rpm-i 命令来安装它。

3.8 远 程 通 讯

用 Linux 直接和远程系统相连是非常容易的。简单地配置调制解调器(modem)后,你就可以使用几个通讯程序来拨号并注册入网。最流行的两个流行远程通讯程序是 minicom 和 seyon 提供了容易使用的界面和像自动注册等的高级特点。

3.8.1 调制解调器的设置

如果你的 PC 有一个调制解调器,它可被连接四个通讯端口中的一个,这些端口的名字分别是:COM1,COM2,COM3 和 COM4,这些端口也被用于像鼠标等的串行口设备。通常,鼠标使用 COM1,COM2,COM3 和 COM4,这些端口也被用于像鼠标等的串行口设备。通常,鼠标使用 COM1,modem 使用 COM2,或者 COM4,要配置调制解调器,你需要知道你的鼠标和 modem 使用的端口。COM1 与 COM3 是计算机上的同一端口,所以,千万不要让你的鼠标使用 COM1 端口,而让 modem 使用 COM3 端口。否则当你使用 modem 时会发现鼠标被切断。如果你的鼠标与 COM1 相连,那么你的 modem 应用 COM2 或 COM4 端口。

在 Linux 系统中,四个端口有不同的名字,调制解调器的端口是以 /dev/cua0

为起始的(注意,开始的数字是 0,而不是 1)。`/dev/cua0` 是第一个端口 `COM1`,`/dev/cua1`,`/dev/cua2`,`/dev/cua3` 分别是二、三、四端口,许多 Linux 通讯程序里,你需要知道 modem 所连的端口。

一些通讯程序允许 `/dev/modem` 来访问 modem 的端口,通常这个名字不存在你的系统中,它是 modem 所用端口的别名,你一旦知道 modem 所用端口可以用 `ln -s` 命令来创建别名。下面的例是为 `.com2`,端口 `/dev/cua1` 创建别名 `/dev/modem`。首先你必须是超级用户的身份注册。

```
#ln -s/dev/cua1/dev/modem
```

鼠标所使用端口的另名通常设置为 `/dev/mouse`,这个端口通常是 `/COM1`,`/dev/cua0`。

```
#ln -s/dev/cua0/dev/mouse
```

3.8.2 远程通讯程序: `minicom` 和 `seyon`

`minicom` 是一个可以使你注册到其它系统的终端仿真器,它不是一个 X-`Windows` 程序。任何用户帐户口都可以通过命令行来运行它。键入 `minicom` 命令就可以启动它,可以用 `CTRL-A` 加另外一个字符来使用命令,例如 `CTRL-A-D` 键将出现一个拨号目录,按 `ESC` 键可以退出任何窗口。`CTRL-A-Z` 键显示列出不同命令的帮助屏幕。在这个屏幕中,你能够按命令相对应的字符来执行命令,按 `CTRL-A-P` 键会出现一个屏幕,要求你键入你的 modem 参数。按 `CTRL-A-O` 键出现一个屏幕,要求你键入你的 modem 参数。按 `CTRL-A-O` 键出现一个配置菜单,端口设置选项会产生一个选择 modem 使用端口的屏幕。例如 `/dev/cua1`。调

制解调器选项产生一个键入 modem 信息的屏幕。文件名和路径选项让你选择上或和下载文件的路径。

按下 CTRL-D 键,出现一个拨号目录,选择你要与之连接的系统号码,设置一个远程连接。拨号目录允许你用屏幕底部的命令增加和编辑选项。用左右光标移动键选择选项,用上下光标键选择准备拨打的号码然后选择拨号命令。在屏幕上可看到注册和口令提示符,你也可以创建一个自动执行拨号和注册操作的文本。

minicom 支持 zmodem,ymodem,xmodem 和 Kermit 转换协议,用 CTRL-A-S 键可以发送文件用 CTRL-A-R 键接收它们,minicom 允许你增加可获得的新协议 CTRL-A-H 键挂机,CTRL-A-X 键退出 minicom。

Seyon 是一个运行在工作平台上的一个远程通讯程序,它提供了一个终端仿真和支持用于下载,上载文件的传输协议。它包括一个拨号目录和一个自动注册和下载的脚本语言,Seyon 界面由分布式窗口,按钮和菜单组成,每个部件支持各种特点。拨号目录允许你编辑注册文本,来选择像自动重播等优点,终端仿真使用 Xterm 窗口,使它具有像滚动缓冲器,剪切和粘贴的功能。当 Seyon 启动时,它显示了一个带有不同按钮的命令中心窗口,每个按钮执行不同的任务。Transfer 按钮是用来下载文件,用 Dial 按钮,可以拨号和注册到一个远程系统。

3.9 使用调制解调器和 Internet 连接,pppd 和 ezppp

如果使用 modem 通过电话线与 Internet 相连,你就必须设置 PPP 连接,点对点(PPP)协议是大部分使用 modem 与 Internet 相连的协议(你的计算机是通过网卡直接与一个 Intranet 相连。你的网络连接应该是它配置好的。如果没被配置好,可以用 Lisa 来配置它)。

在 Linux 系统中,pppd 命令完成了 PPP 连接,20 章将会详述,你可以用下面部分描述的方法直接执行 pppd 命令,也可以通过 Ezppp Internet 连接工具来实现。Ezppp 并没有放在你的光盘上,你必须从 Redhat Contrib 目录下下载它。要热这样做,首先要设置一个 ppp 连接,下面部分简要叙述了如何用 pppd 命令设置的 ppp 连接,20 章将再次详述。

3.9.1 pppd

大部分的 Internet 服务器在它们与 Internet 的连接中使用动态的本地和远程地址。这里描述了一个简单的动态连接,20 章详细讨论了 pppd 和静态地址。pppd 使用两个文件来建立连接。chat 文件规定了如何用注册名和口令与服务器相连。Option 文件包括了一套 pppd 用来配置连接的标准选择。

你首先应该设置 pppd 选项,pppd 有很多选项。例如,noipdefanlt 选项,指示了 pppd 从 ISP 远程系统中探测和使用一个动态 IP 地址。你可以在 PPP 命令行的 Speed 后面列出各种选项,但这可能使命令行变得十分长和复杂。另一种

方式,pppd 允许你键入/etc/ppp/Options 文件中的选项,pppd 将自动阅读和使用这个文件中的选项,这里列出了你需要使用的标准选项,用 Crisplite Vi或 XEmaes 编辑器来编辑,/etc/ppp/Option 选项并键入这些命令。

```
/etc/ppp/Options
```

```
Crtscts
```

```
defaultroute
```

```
modem
```

```
asynmap o
```

```
noipdefault
```

现在准备设置连接信息,Chat 命令会完成拨号连接,Chat 必须明确全部连接信息,电话号码,注册提示符,用户 ID 号,口令提示符和口令,及一些连接用的字符串,Chat 操作由一对期待与回答的字符串形成的一些选项组成,第一个字符串是期望接收的内容,第二个字符串是收到时发送回答的内容,如果你不想收到任何内容,使用一个无效的空串""。下面的例子中,首先是期望不收到正文回答准备拨号。

```
"" ATDT 5556666
```

远程系统通常发送一个注册提示符,经常是一个带冒号的"login",你只需要最后几个字符 Ogin,不要忘了冒号,你键入用户的 ID 号,回答依据不同的 ISP,你可能需要在用户 ID 号后加上 n 如 mylogin\n。

```
ogin:mylogin.
```

注册进入系统后,你能期望收到口令提示符,仍仅需要最后几个字符 Word:回答你的口令。

Word :mypass

用 Chat 命令把期望回答对合并在对话操作中,通常使用 -v 选项,以便 Chat 报告其行动。

```
' Chat -v -f " "ATDT 5556666 Ogin:mylogin Word:mypass'
```

你必须把对话操作放进 pppd 命令可调用的执行中,整个对话操作用单引号引起来,并键入 pppd 命令的同一行中,Chat 程序用 Chat 文件中的信息初使化调制解调器,向远程主计算机拨号入网,然后用用户 ID 和口令注册。

然后,你需要键入带有 Chat 调用 modem 的设备名,及速度的 pppd 命令。标准的语法显示如下,如果你对 ISP 有一个静态连接,在速度后还要加上 ISP 静态 Internet 地址。

```
pppd Options Serial-device-name Speed
```

Serial-device-name 是调制解调器(modem)的设备名,通常是 /dev/cua 加上从 0 到 3 的一个数字。速度是波特率,V44 modem 的波特率是 14400。V.28 modem 的波特率是 38400 或 57600 在 modem 文件中找到它能支持的最高速度。

Connect 选项指示 pppd 完成一个连接,在 pppd 后键入 Connect 选项和 Chat 命令。整个 Chat 操作用单引号引起来。下面的例子中,用户用 Chat 操作激活 pppd。调制解调器连在端口 2./dev/Cna1 速度是 57600 波特,单引号中包括了期望回答对。

```
# pppd connect 'Chat-v'" ATDT556666 Ogin:mylogin word:  
mypass' plev/cua1 57600
```

现在,可以试着用 pppd 与远程系统相连了。一些号码可能是错误的,也可能

你没有使用正确的字符串或 modem 初始化出错,pppd 会把每一步的操作级在 /var/log/messages 文件中,你可以 more,tail 或 cat 来查看,对于 pppd 操作的描述,如果连接成功,它会列出 IP 地址。下列的命令将显示 ppp 操作,完成后,用 CTRL-C 键回钮提示符下。

```
$tail -f/var/log/messages
```

下面是一个简单的 ppp 操作,显示了 pppd 和 chat 注册到远程系统时 /var/log/messages 文件中的内容,当看到本地和远程系统的 IP 地址时,连接已经建立。键入 CTRL-C 结束报告,也可以使用几个不带 -f 选项的 tail/var/log/essages 命令,可以使不使用使用 CTRL-C。

```
# pppd connect 'chat -v "" ATDT5556666  ogin: mylogin  word:  mypass'  
/dev/cua1 57600  
# tail -f /var/log/messages  
Aug 24 17:55:49 turtle chat[294]: send (ATDT8659004^M)  
Aug 24 17:55:49 turtle chat[294]: expect (ogin:)  
Aug 24 17:56:05 turtle chat[294]: CONNECT 57600^M  
Aug 24 17:56:07 turtle chat[294]: ^M  
Aug 24 17:56:07 turtle chat[294]: login: -- got it  
Aug 24 17:56:07 turtle chat[294]: send (mylogin^M)  
Aug 24 17:56:08 turtle chat[294]: expect (word:)  
Aug 24 17:56:08 turtle chat[294]: Password: -- got it  
Aug 24 17:56:08 turtle chat[294]: send (mypass^M)  
Aug 24 17:56:08 turtle pppd[293]: Serial connection established.  
Aug 24 17:56:09 turtle pppd[293]: Using interface ppp0  
Aug 24 17:56:09 turtle pppd[293]: Connect: ppp0 <--> /dev/cua1  
Aug 24 17:56:12 turtle pppd[293]: local IP address 204.32.168.173  
Aug 24 17:56:12 turtle pppd[293]: remote IP address 163.179.4.32  
Ctrl-c  
#
```

用 disconnect 选项击活 pppd,断开 ppp 连接,必须用 Chat 指示 modem 挂机,你可能必须键入 HO 等 modem 命令一定要包括 modem 设备行或 /dev/modem。

```
# pppddisconnect 'Chat" " +++ATHO OK' dev/cua1
```

有几种方法自动完成这一过程,你可以创建一个含有期望回答对的 Chat 文件,也可以把整个 pppd 命令放在一个 Shell 脚本里,单独使用 Shell 脚本名来执行全部命令,20 章详述了其它的特点。

3.9.2 Ezppp

使用 Ezppp 程序,可以不必亲自配置连接。Jaypaintor 编写的 Ezppp 使 PPP 配置过程与在 Windows 95 中,配置 Internet Setup Wizard 一样简单,屏幕出现要求键入 Internet 和 modem 信息的对话框,一旦配置完成,只要单击"CONNECT"。按钮连接就被完成了。Ezppp 是在超级用户下运行的,但需要系统上其它用户可以使用这些专门文件的允许设置,见 /usr/doc/exppp* 目录下的文件。它也包含了你可用网络浏览器浏览的详细信息。

Ezpppp 并没有存放在你的光盘中。但可能过 rpm-i 命令来下载它的 RPM 软件包并安装它们。这些软件包放在 /pub/contrib/i386 目录中 ftp.redhat.com 下的 Redhat Contrib 目录下。目前,这个程序是二级发行,但仍很稳定,Ezppp 网页放在 www.sev.net/~cameron/exppp 网址下(你可以通过第 1 章表 1-3 所列的 Linux 网页资源来连接它)。在这里,可以发现更多的信息和 Ezppp How-To 文件,并下载软件)。这部分描述了要求使用 QT 共享库第一版本——二级发行 9 的安装和使用。QT 共享库也在 Redhat Contrib 目录下由 rpm 配置并自动安装。现在可获得 Ezppp 的最新版本,包括静态和非静态的版本,如果下载非静态版本,不必使用 QT 库。

下面的例子是使用 ftp 从 ftp.redhat.com 网址上下载 Ezppp 用 cd 命令、用

户可转到放置 Linux 软件的 /pub/Contrib/i386 目录下。这个目录包含大量软件包。

键入 Ls 和带通配符的文件包开头几个字符,可以列出准备获得的软件包在这个例子中用 Ls ezppp*显示 Ezppp 软件的全名。用 get 命令下载 Ezppp RPM 文件然后用另一个 get 命令下载 QT 库。hash 命令用于下载大的文件。它为每个接收到的数据包设置一个 hash 标志使你了解 ftp 正在工作。

```
# ftp ftp.redhat.com
Connected to spoodier.redhat.com.
Name (ftp.redhat.com:root): anonymous
331 Guest login ok, send your complete e-mail address as password.
Password:

230 Guest login ok, access restrictions apply.
Remote system type is UNIX.
sUsing binary mode to transfer files.
ftp> cd pub/contrib/i386
>250 CWD command successful.
ftp> ls ezppp*
>200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
-rw-r--r-- 1 root root 74297 May 25 00:55 ezppp-1.0B9-2.i386.rpm
-rw-r--r-- 1 root root 84817 Apr 22 03:38
ezppp-dynamic-1.0B8-1.i386.rpm
-rw-r--r-- 1 root root 301329 Apr 22 03:40 ezppp-static-1.0B8-1.i386.rpm
226 Transfer complete.
ftp> binary
200 Type set to I.
ftp> get ezppp-1.0B9-2.i386.rpm
>local: ezppp-1.0B9-2.i386.rpm remote: ezppp-1.0B9-2.i386.rpm
200 PORT command successful.
150 Opening BINARY mode data connection for ezppp-1.0B9-2.i386.rpm (74297 bytes).
226 Transfer complete.
ftp> hash
>Hash mark printing on (1024 bytes/hash mark).
ftp> ls qt-1*
>200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
-rw-r--r-- 1 root root 419166 Apr 7 20:00 qt-1.31-1.i386.rpm
226 Transfer complete.
ftp> get qt-1.2-1.i386.rpm
>local: qt-1.31-1.i386.rpm remote: qt-1.31-1.i386.rpm
200 PORT command successful.
150 Opening BINARY mode data connection for qt-1.2-1.i386.rpm (419166 bytes).
#####
#####
226 Transfer complete.
```

```
419166 bytes received in 132 secs (3.1 Kbytes/sec)
ftp> close
>221 Goodbye.
ftp> quit
#
```

一旦下载后,你能用 rpm-i 命令来安装 RPM 软件包。

```
$ rpm-i ezpppp-1.0139-z.i386.rpm
```

```
$ rpm-i qt-1.31-1.i386.rpm
```

你安装了 Ezppp 后,可能 打开 Xterm 窗口键入 ezppp 命令来运行这个程序。你也能通过 fvwm 程序菜单来运行它。此菜单放在应用程序菜单里的其它应用菜单中。图 3-10 显示了主屏幕单击"configure"按钮来设置一个连接。屏幕会出现带有一个面板的配置窗口,就好像 Window 95 的配置工具一样,一个面板在最前端。其它面板的标签显示在上部,在配置窗口里有三个面板。一个帐号,另一个调制解调器的设置,还有一个是一般的特点。帐号包括了你的 Internet 信息。根据你开设不同 Internet 服务商的个数。你可以有多个帐户(大部分的用户只有一个帐号)。单击"New"按钮创建帐号。这会出现带有一套 Internet 信息面板的窗口。拨号面板为了键入是像用于连接 Internet 电话号码等的连接信息的地方。确定键入一个用于连接的名子,它可以是任何名。"Argument"按钮将产生一个可键入 PPP 要点的另一个窗口,20 章详述了 PPP 保护选项。通常只需要已经被键入的 -detach 和 defaulttrout 选项。

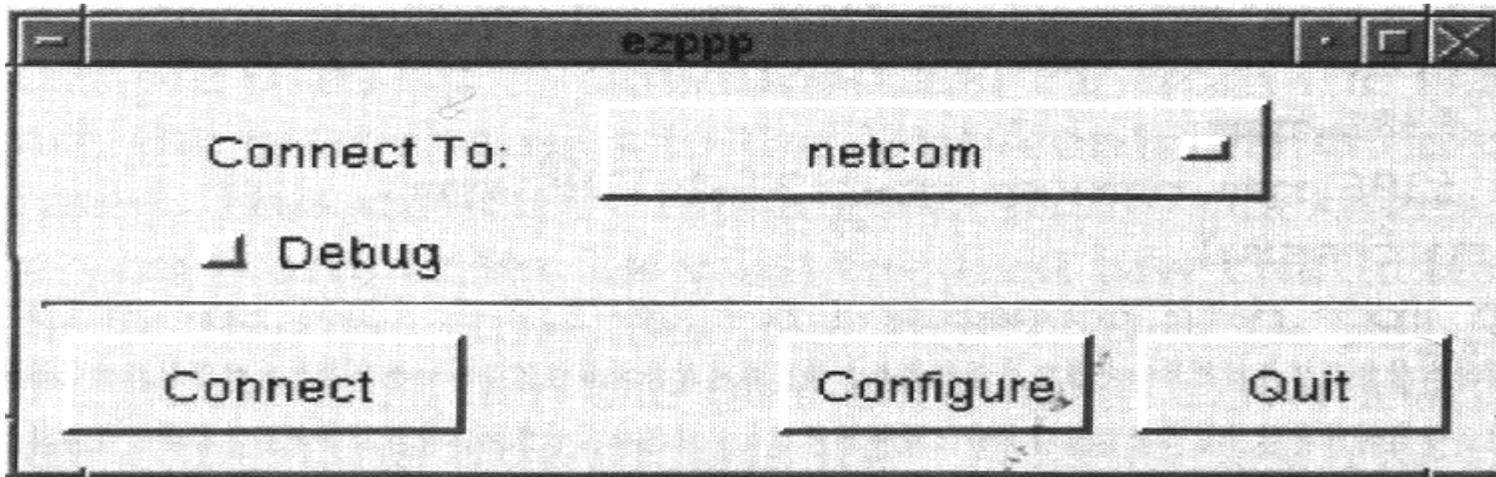


图 3-10EzPPP main window

大部分的准标选项做为缺省值,图 3-11 显示了 Configuration,New Account,和 Arguments 窗口在 New Account(新帐号)窗口中 IP 面板用于键入任何本地和远处的 IP 地址,和子网屏蔽。每个选项有一个动态(Dynamic)和静态(static)的确认按钮。并且动态的按自己被缺省设置。如果你的 Internet 服务商提供你一个动态的 IP 地址和子网屏蔽号。你可以离开这个面板,否则,如果你有一个静态的本地或远程地址、你就必须用这个面板键入它们。

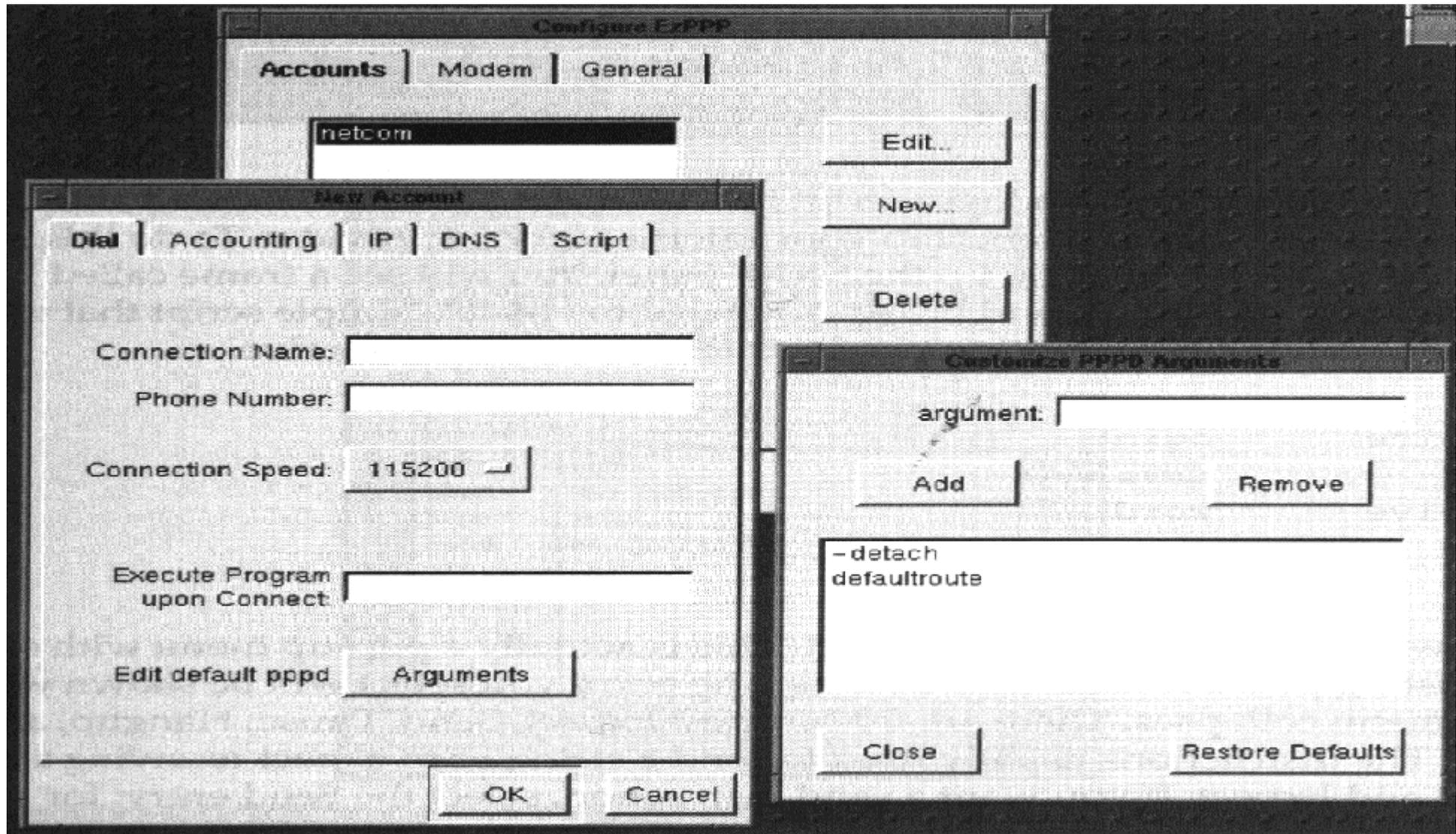


图 3-11 配置 Ezppp,新帐号、和 Customizepppd Argument 窗口
单击标有 DNS 的标题栏,屏幕出现了域名服务器面板。在这里,键入 Internet

服务商的域名服务器的 IP 地址。单击 IP 地址的方框。键入地址。再单击“Add”按钮增加它到域名服务器的列表中。图 3-12 表明了带有 IP 地址例子的面板。

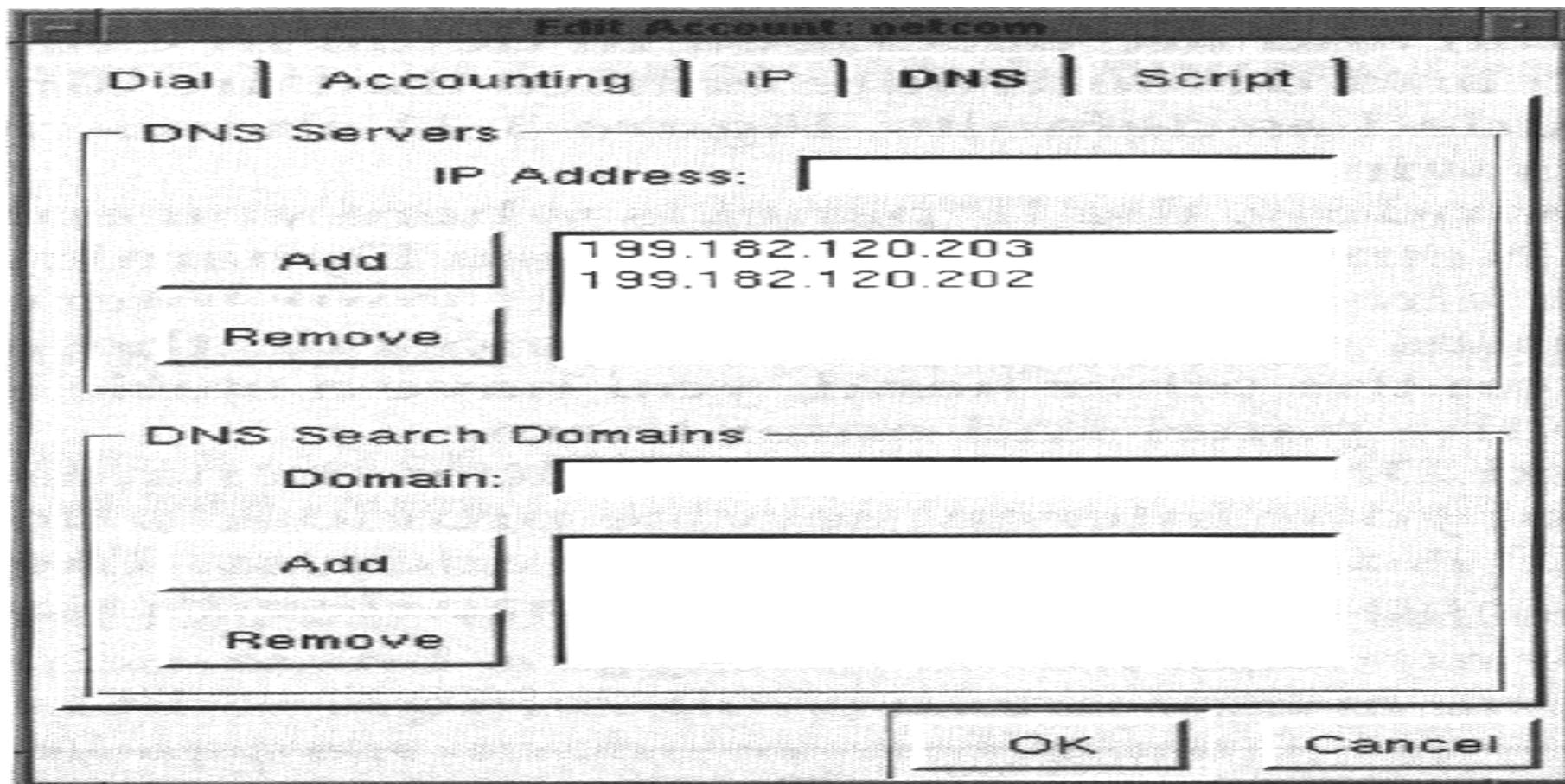


图 3-12 Ezpppp 域名服务器面板

然后,你必须创建一个可以输入用户名和口令的注册文件,以便能和 Internet 服务器相连。为了做到这一点,你可以单击 Script 标题栏产生 Script 面板。你会看到一个叫 Login Script 的面板,其中有几个按钮和方框,可以用 Send 和 Receive

两个指令创建一个执行如下任务的简单文本。

```
Receive      ogin:  
Send         username  
Receive      word  
Send         password
```

你将看到一个 Expect 按钮,这通常是一个带有几个可能列出注册指令选项的一个弹出式菜单,当选中跳出式菜单,选项会出现,包括:Expect,Send,Pause,Hangup 和 Answer。Send 和 Expert 指令通常发送和期望接收一个字符串。如果你想发送一个指令,选择 Send 选项,期望收到指令,选择 Expect 选项。按钮右侧的方框是键入你想操作的指令的地方。例如,如果期望收到"ogin"指令,你应选择 Expect 选项并在方框里键入 ogin。单击 Add 按钮,把它放在文件中,在 Add 按钮下的框中你会看到"expect"项目中对应"ogin"。如果你单击此项,按 Remave 按钮,可将其删除。如果打算插入一个指令用 Insert 按钮。

大部分注册文本只需包含两对发送和期望收到指令,一个是为了注册名,另一个是为了口令,第一个口令是期望收到"ogin"第二个口令是发送用户名,第三个口令期望收到文本"word",第四个发送口令。根据 Internet 供应商的不同,这可能会有一点变化。图 3-13 表明了一个注册文本的例子。"my-login-name"做为用户名"my-password"做为口令。

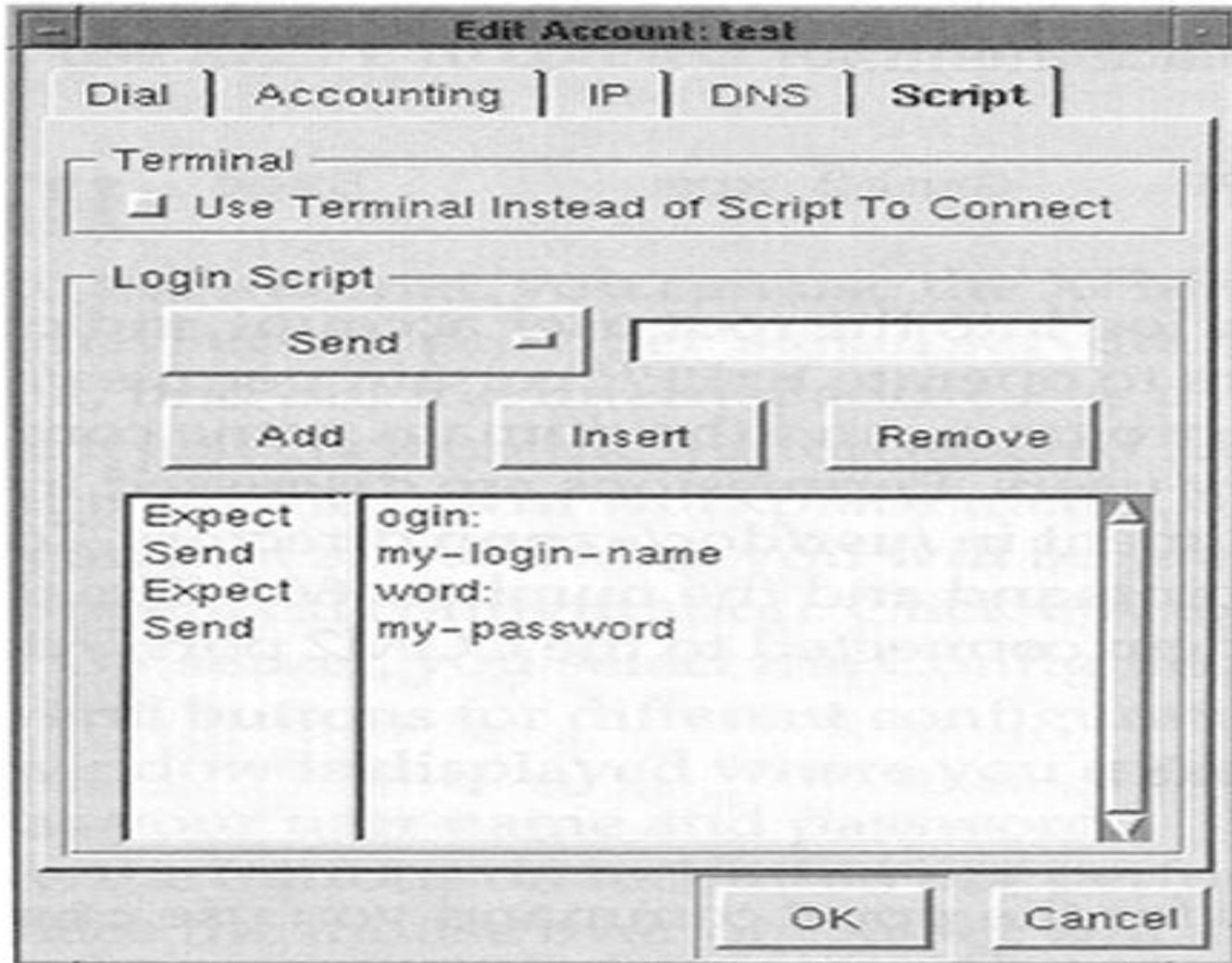


图 3-13 Ezppp 注册文本面板

你已经完成了键入 Internet 信息,单击 OK 按钮关闭 Script 和新帐户窗口,在 Configure 窗口中单击调制解调器标题打开调制解调器面板,确定调制解调器的端口是正确的,以 /dev/cua 开头附加从 0 到 3 的数字,在微机中,有四个端口供选

择通常一个调制解调器与第 2 个或第 4 个端口相连。名字分别是 /dev/cua1 和 /dev/cua3(计数从 0 开始,所以端口 1 是 cua0,端口 2 是 cua1)。

可以设置 /dev/modem 做为端口的别名。然后,你需要核对拨号前缀和一些特殊设置。单击 Edit 按钮产生一个窗口,显示一系列的配置 modem 的方框,通常,一些缺省设置已被键入再分别核对 Connect 选项和 Hangup 选项。缺省连接 CONNECT 对于大多数 ISP 都是有效的。Hangup 字符串是大部分 modem 使用的挂执字符串。

完成后,单击 OK 按钮,准备使用 EzPPP。单击主 Ezppp 窗口中的 Connect 按钮,如果出现问题,单击调试按钮,出现一个显示连接过程和错误的窗口,当 EzPPP 完成一个连接,它会显示接通的时间,按 Disconnect 按钮,可退出 EzPPP 程序。

你也可以在 fvwm 工具栏中设置一个按钮,单击它时启动 EzPPP。/USR/lib/X11/fvwm 目录下的 System.fvwmrc.ggod stuff 文件末尾键入下面一行(在 fvwm 工作地点菜单里的工作平台菜单下选择 Good Stuff 来显示工具栏)。

```
*GoodStuff EzPPP Game2.xpm Exec "" Xlanrch ezppp
```

EzPPP 是运行在超级用户帐号下。但是,对于 modem 设备设置允许权限,可让普通用户操作 EzPPP PPPd 程序和域名配置文件可以允许普通用户使用它们,第 7 章详述了允许权限,看 /usr/doc/ezppp 目录下的 E2PPP How To 文件。用 chmod 命令和数字 666 来获得使用调制解调器,下面的例子设置了连接 COM2 端口 /dev/cua1 的调制解调器允许权限。

```
# chmod 666/dv/cua1
```

用 chomd 命令 +s 选项可以设置 pppd 程序的使用权限。

```
# chmod +s/usr/sbin/pppd
```

下面的例子是确定 pppd 命令只被超级用使用。

```
# chown root.root pppd
```

可以用 ls-l 命令查看 pppd 命令的使用权限

```
# ls -l/usr/sbin/pppd
```

```
-rwsr-sr-x | root voot 8460+Avg 14 1996
```

```
/usr/sbin/pppd
```

要使用户可以增加域名服务器,必须允许普通用户可以使用 /etc/resolv.conf 文件。这是一个选择。超级用户可以容易地增加域名服务器。

```
#chmod 666/etc/ resolv.conf
```

每个用户可以用前面的几个步骤创建他们的 EzPPP 帐号,用户 home 目录下的 .ezpppre 文件包含了这些信息,事实上,不同用户可以 EzPPP 与不同的 Internet 服务器相连。

3.10 电子邮件:XFMail

一旦你与 Internet 相连,就可以用 XFMail 程序从 Internet 服务商的 Mail 服务器中收到邮件,大部分的 ISP 用 POP 服务器接收邮件,首先,找到服务器的地址,从 fvwm 工作地点菜单中选择 fvwm Good Stuff 工具栏,单击 XFMail 按钮开始 XFMail 程序。或者在一个终端窗口内键入 xfmmail。当启动 XFMail 后,会出现一系列用于一般配置的配置窗口。单击 OK 键接收缺省值。选择 MISC 菜单中的

Config 选项,配置与 POP 服务器的连接。这会打开一个带有几个完成不同配置任务按钮的窗口。单击 POP 标题,POP 窗口被激活,在这里你可以键入 POP 服务器的 Internet 地址,用户名和口令。

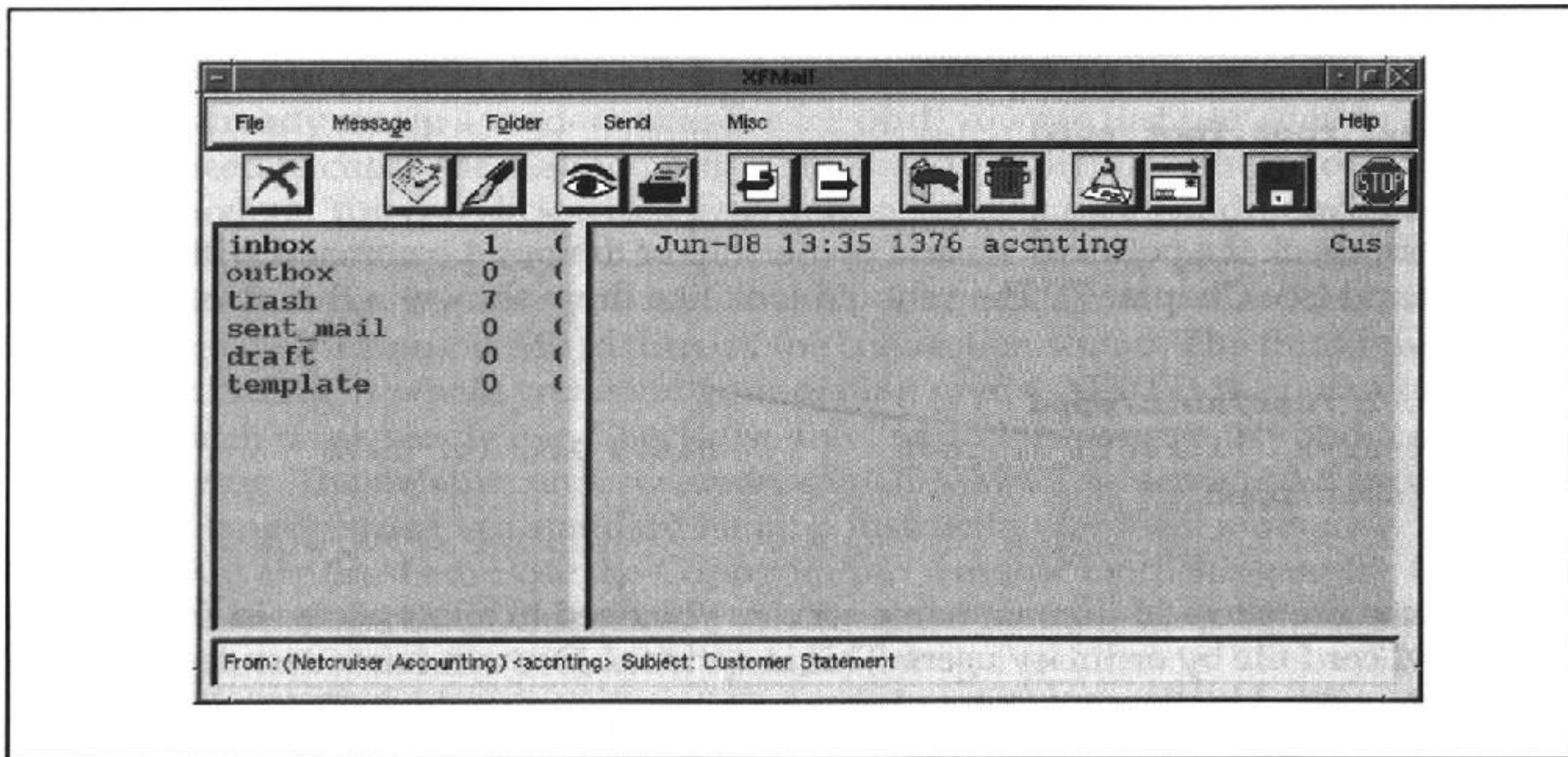


图 3-14 XFMail

在 XFMail 窗口中,按钮执行了各种邮件任务(见图 3-14)。如果在按钮上放置

鼠标箭头,一个解释按钮功能的标鉴将显现,单击正确按钮,可选则查看信件,信件的标题选被显示出来,双击一个标题,屏幕会显示邮件的全部信息。

3.11 X-Windows 和网络配置

用网络配置工具,可以增加和改变你的网络配置。如果你正在与一个网络相连,且安装时没有配置它,可以用这个工具执行对网络的配置,在超级用户的工作台上,Admin-Tools 文件夹中的 netcfg 图标代表网络配置工具,双击图标,配置窗口就被激活,20章详细讨论了 netcfg 工具的使用。

如果你不能启动 X-Windows,可能是因为系统没有安装正确的显示驱动程序,尽管 XF86 Setup 程序提供了大多会显示卡的驱动程序,但有几个是它不支持的,详见 21章和附录 C。也可使用 Xconfigurator 或 X86Config 等其它 X-Windows 配置程序,它们是用行来配置的工具。

3.12 总结:Linux 系统的启动和退出

通过在注册提示符下键入注册名在口令提示符下键入口令,来使用 Linux 系统。一旦注册后,屏幕显示可以键入命令的命令行界面。命令带有选项和要点,在命令行键入命令再回车来执行命令,当完成后,用 logout 命令来退出系统,注册提

示符重新出现如果要关闭计算机,必须用 CTRL-ALT-DEL 键来关闭 Linux 系统。

操作 Linux 系统时,一定要记住必要的启动和关闭步骤,采取下述 2 个步骤可进入 Linux 工作平台。先启动 Linux 系统,注册、再用 startx 命令启动 Caldera 工作平台的 X-Windows,退出工作平台,注锁帐号,然后才能用 CTRL-ALT-DEL 键关闭系统。

你也可以用一个窗口管理器与你们的系统交互,Startx 命令启动 Caldera 工作平台,它使用 fvwm 窗口管理器和它自己的文件和程序管理器,从 File 菜单中选择 Exit Desktop 来退出工作平台,下面列出了操作 Linux 系统的所有步骤,记住注册和图形用户界面的启动过程和图形用户界面的退出、注锁帐号和关机过程。

1.启动计算机

2.在 LILO 提示符下键入 Linux,回车

LILO:Linux

3.在注册和口令提示符下分别键入用户名和口令。

Login: richlp

password:

4.一旦注册后,出现 shell 命令行提示符\$。

\$

5.在 Shell 提示符下键入 Startx 命令、启动 X-Windows 图形用户界面。

\$ startx

这个命令安装了 X-Windows 图形用户界面,包括 X-Windows,管理器和 Caldera 工作平台外形。他们的集合叫做 Caldera 工作平台。

6.选择 File 菜单中的 Exit DeskTop 选项退出工作平台,即退出 X-Windows

图形用户界面、回到窗口管理器。

7.单击屏幕上任一键,跳出窗口管理器菜单,选择 Exit Desktop 退出窗口管理器。这时出现一个子菜单、选择 Quit Desktop,回到命令行提示符。

\$

8.在命令行键入 exit 或 logout 再回车来注销帐号。

\$ logout

9.然后可以再以其它用户名注册另一个帐号,或用超级用户注册帐号。

login:

10.现在可以用 CTRL-ALT-DEL 键来关闭系统。

11.当屏幕出现空白,计算机会重新启动,这时,可以关闭计算机。

你可能需要用超级用户身份执行一些如创建新用户和安装软件的系统管理任务,以超级用户身份注册,然后启用工作平台,使用系统管理工具完成这些任务。例如,用 Usercfg 允许你创建和删除用户,netcfg 允许你配置你的网络连接。

Linux 的 Caldera 分支部门把软件组织易用 glint 工具和 rpm 命令安装和拆卸的软件包。glint 工具放在工作平台上。在命令行,可用 rpm 命令执行相同的操作,表 3-1 列出了 rpm 命令的选项。

表 3-1 Redhat Package Manager(RPM)

操作方式

效果

rpm -i <i>options package-file</i>	Installs a package; the complete name of the package file is required
rpm -u <i>options package-name</i>	Uninstalls a package; you only need the name of the package, often one word
rpm -q <i>options</i>	Queries a package; an option can be a package name or a further option and package name, or an option applied to all packages
rpm -b <i>Options package-specifications</i>	Builds your own RPM package
rpm -verify <i>options</i>	Verifies that a package is correctly installed; uses same options as query; you can use -V or -y in place of -verify

Install Option (to be used with **-i**)

-U	Upgrade; same as install, but any previous version is removed
-p	Displays percentage of package during installation
-t	Tests installation; does not install, just checks for conflicts
-f	Forces installation despite conflicts
-root <i>directory-path</i>	Installs at directory

Uninstall Option (to be used with **-u**)

-t	Tests uninstall; does not remove, just checks for what will be removed; use with -v
-----------	--

Query Option (to be used with **-q**)

<i>package-name</i>	Queries package
-a	Queries all packages
-f <i>filename</i>	Queries package that owns <i>filename</i>
-F	Queries package that owns <i>filename</i> ; <i>filename</i> is read from standard input
-p <i>package-name</i>	Queries an uninstalled package
-i	Displays all package information
-l	Lists files in package
-d	Lists only documentation files in package
-c	Lists only configuration files in package
-t	Tests uninstall; does not remove, just checks for what will be removed; use with -v

General Option (to be used with any option)

-v	Verbose; displays descriptions of all actions taken
-quit	Displays only error messages
-version	Displays rpm version number
-root <i>directory</i>	Uses <i>directory</i> as top-level directory for all operations (instead of root)

Other Sources of Information

RPM-HOWTO on www.redhat.com

More detailed information, particularly on how to build your own RPM packages

man rpm

Detailed list of options

表 3-2 Linux Resources Web Pages(Links) and RPM Software ftp Sites
Linux RPM 软件 目录 描述
ftp 站点

<code>ftp://ftp.redhat.com</code>	<code>/pub/contrib/i386</code> <code>/pub/linux/redhat</code>	RPM packages for applications
<code>ftp://ftp.caldera.com</code>	<code>/pub/mirrors</code> <code>/redhat-contrib</code>	RPM packages for applications
<code>ftp://ftp.caldera.com</code>	<code>/pub/netscape</code> <code>/navigator</code>	Netscape navigator RPM package
<code>ftp://sunsite.unc.edu</code>	<code>/pub/linux</code>	RPM and compressed archives
<code>ftp://tsx-11.mit.edu</code>	<code>/pub/linux</code>	RPM and compressed archives
<code>ftp://ftp.x.org</code>	<code>/contrib</code>	X-Windows compressed archives
Web Sites	Page Location	Web Page Description
<code>www.ssc.com</code>	<code>/linux/resources</code> <code>/web.html</code>	Linux on the World Wide Web
<code>www.caldera.com</code>	<code>/tech-ref/linux_info.html</code>	Caldera Linux Resources Page
<code>www.redhat.com</code>	<code>/linux-info</code>	Redhat Linux Resources Page
<code>www.cs.washington.edu</code>	<code>/homes/tlau/linux-games</code>	Linux Game Tome
<code>www.xnet.com</code> <code>www.redhat.com</code>	<code>-blatura/linapps.shtml</code> <code>/linux-info/linux-app-list/linapps.html</code>	Linux Applications and Utilites Page
<code>www.uk.linux.org</code>	<code>/LxCommercial.html</code>	Linux Commercial Vendors Index
<code>www.m-tech.ab.ca</code>	<code>/linux-biz</code>	Linux Business Applications
<code>www.fokus.gmd.de</code>	<code>/linux/linux-prog.html</code>	Woven Goods list of Linux programs

第 4 章 窗口管理器和桌面 :X-Windows

不用命令行的界面,你可以用基于 X-Windows 的图形用户界面来和 Linux 系统进行交互。采用这种界面你可以使用图标,窗口,和菜单来发布命令和运行程序。和普通的基于个人计算机的图形用户界面譬如像 Windows 或者是 Mac OS 不同的是,Linux 和 Unix 系统把图形用户界面分成三个分离的部分:X-Windows,窗口管理器,和程序/文件管理器。X-Windows 是底层的标准图形工具,它提供所有基本的图形操作,譬如像打开窗口或者显示图象。一个窗口管理器负责处理窗口的操作,像改变大小或者移动窗口之类。窗口管理器在窗口被显示的方法上有所不同,使用不同的边框和窗口菜单。但是所有的这一切都是采用了同样的底层 X-Windows 图形工具。文件管理器负责处理文件的操作,它使用图标和菜单,和一个程序管理器来运行程序,通常允许你从一个工具条上选取常用的操作。不像窗口管理器,文件和程序管理器可以在它们的功能上差别很大。Caldera 桌面是一个商业级别的文件和程序管理器。而 Caldera 桌面和 Xfm 文件管理器都可以运行在 fvwm 窗口管理器上。

所有 Linux 和 Unix 系统都使用同样标准的底层 X-Windows 图形工具。这意味着,在大多数情况下,基于 X-Windows 的程序可以运行在任何一个窗口管理器和桌面的系统上。基于 X-Windows 的软件常常可以在 Linux 和 Unix 的 ftp 站点上可以看到,它们通常标注为 X11。你可以下载这些包然后在你的 Caldera 桌

面上或者是在 fvwm 下直接运行它们。有些可能已经是 Linux 的二进制形式,你可以下载,安装,然后直接运行。另一些将会是源码的形式。用一些简单的命令它们很容易被设置,编译,然后安装在你的系统上。一些应用程序,像 Motif 应用程序,可能还需要一些特别的库。

4.1 窗口、文件和程序管理器

采用一个窗口管理器你可以用你的鼠标来完成象打开,关闭,改变大小,和移动窗口这样的窗口操作。有几种窗口管理器可以用在 Linux 系统上(如表 4-1)。一些更通用的窗口管理器象自由虚拟窗口管理器 (Free Virtual Windows Manager(fvwm)),fvwm95,LessTif(mwm),Motif(mwm),Xview(olwm), 和 AfterStep(afterstep)。fvwm 是 Robert Nation 写的用于绝大多数 Linux 系统上的主窗口管理器。它使用简单,功能强大,而且灵活.fvwm95 是一个有着 Windows 95 的用户界面的 fvwm.它使用和 Windows 95 操作系统使用的相类似的窗口,任务栏,和菜单。将来的版本还将包括一个允许使用图标来显示和操作文件的文件管理器。LessTif 是 Motif 的一个免费的克隆。所有 Motif 的应用程序都可以在 LessTif 下运行。Motif 是一个和它所使用的 Unix 一样的商业产品。Xview 是一个 Linux 版本的 Sun 系统的 OpenLook 界面(不是 Solaris)。AfterStep 是用于 NeXT 操作系统上的 NeXTSTEP 界面的一个克隆。

窗口管理器在底层的 X-Windows 图形工具之上运行。X-Windows 实际上提供了让你打开,移动和关闭窗口以及显示菜单和选择图标这样的基本操作。

fvwm 和 Xview 用各自的方式管理这些操作,提供不同的界面以供选择。所有的窗口管理器,无论它们外观上有多么大的差别,都是使用 X-Windows 工具。从这个意义上说,Linux 不是只能绑定于一种图形用户界面。在同一个 Linux 系统上,一个用户可以使用 fvwm 窗口管理器,而另一个用户可能使用 Xview,还有可能使用 Motif。

窗口管理器一般仅仅提供非常基本的窗口管理操作,像打开,移动,和对窗口改变大小。而文件管理器和程序管理器让您用图标和菜单来管理和运行程序。使用文件管理器,你可以在不同的目录窗口中拷贝,移动,或者删除文件。使用程序管理器,你可以用任务栏和程序图标来执行命令和运行程序。一个桌面程序利用文件管理器和程序管理器的功能来提供一个有菜单和图标的虚拟桌面,用以访问文件和管理程序。Caldera 桌面就是这样一个桌面程序。表 4-1 列出了几种其他的桌面,许多都可以从 Web 站点上免费的获得。

有些窗口管理器的功能增强包括了桌面的一些特点。在许多 Linux 系统上使用的 fvwm 就在窗口管理的功能之外还加上了程序管理的功能。它有一个任务栏和一个工作区菜单让你可以用来访问你所有的 X-Windows 程序。使用菜单或者是任务栏,你可以在 fvwm 中直接运行任何 X-Windows 程序。其他的窗口管理器,像 fvwm95,提供了一个和 Windows 95 一样的任务栏和初始菜单的桌面界面。它基于 fvwm2,但是使用 Windows 95 的界面。除了程序管理,它还计划最终支持一个名叫探索者(explorer)的文件管理器。使用 fvwm95,你可以像你在 Windows 95 中那样显示文件和目录的图标。

4.2 启动和退出 X-Windows 窗口管理器和文件桌面

像在前一章中指出的那样,你在命令行方式下输入 `starx` 启动 X-Windows。你的 X-Windows 服务器将在窗口管理器之后加载。你的 OpenLinux 系统现在被设置为在 Caldera 桌面中自动加载窗口管理器。桌面运行在窗口管理器之上。窗口管理器的背景颜色是淡蓝色。你可以在 Caldera 加载的时候看到这些。Caldera 桌面是一个棕色或者是灰色的窗口,上方是任务栏,在窗口中是程序的图标。你可以用你的鼠标来访问菜单,打开窗口,以及运行程序。

你通过从文件菜单里选择退出桌面选项来退出 Caldera 桌面。象许多窗口应用程序一样,它的桌面在屏幕的上方有一个菜单栏。单击“文件”就可以下拉菜单。退出桌面是最后一个选项。

当你选择退出桌面的时候,Linux 仅仅关闭 Caldera 桌面。它并不关闭 `fvwm` 窗口管理器。`Fvwm` 窗口管理器在你的屏幕上显示为深蓝色的背景。你必须退出窗口管理器才能关闭 X-Windows 退回到 Linux 命令行方式界面。

要关闭 `fvwm` 窗口管理器,首先弹出 `fvwm` 工作区菜单。通过在深蓝色的背景的任何一点上单击就可以看到这个菜单。你必须点住它然后往下拉一点才能让这个菜单停留在屏幕上。选择这个菜单的最后一项,离开桌面。这个操作打开一个子菜单,它其中的一个选项就是退出桌面。选择这个就会退出窗口管理器和 X-Windows。你接着就可以回到给你一个 `$` 提示符的命令行方式界面。在命令行方式下,你可以在任何时候通过输入 `starx` 命令来启动 `fvwm`。

你还可以在 `fvwm` 中启动其他的窗口管理器或者桌面,以及显示 `fvwm` 的任

务栏。在工作区菜单中选择桌面选项。它将打开一个窗口管理器的子菜单。这个菜单中的“开启任务”选项将显示 fvwm 的任务栏,其中包括一些 X-Windows 经常访问的程序的图标,譬如象你的万维网浏览器。“Start Looking Glass”选项开启 Caldera 桌面(Looking Glass 是 Caldera 桌面的另一个名字)。窗口管理器的选项将带出一个子菜单列出你的系统上已经安装的其他窗口管理器。通过在这个菜单里选中其它启动的窗口管理器来启动它。fvwm 已经被设置成在它的菜单中加入了几种常用的窗口管理器。但是,olwm Xview 窗口管理器还不是完全的 Xview 窗口管理器。你必须首先从网上下载然后安装它们。

4.2.1 系统设置的工具

为在 X-Windows 下运行设计的一系列容易使用的设置工具,使你可以完成象安装软件,设置打印机,或者添加新用户之类的任务。XF85Setup,就是用来设置 X-Windows 的这样一个工具,就像 glint 用来安装软件一样。Lisa 是一个菜单驱动的管理工具,它包括了系统管理的各个方面。在 Redhat Linux 的发布中还有一些基于窗口的工具提供了对打印机,用户,和文件系统很方便的设置。这些工具将在第 3,7,9,和 12 章加以讨论。这些工具可以从 Caldera 桌面或者是 fvwm 菜单以及 fvwm 任务栏中启动。为访问这些工具,首先你必须用 root 登陆。然后用 starx 命令启动 X-Windows。从 Caldera 桌面你可以看到一个名为管理工具的图标。双击它打开这个窗口。每个工具的图标都显示在里面。从 fvwm 窗口管理器中选择应用程序,然后是系统管理。从子菜单中选择你想要的工具。fvwm 的任务栏有一个名为管理的按钮将带出一个系统设置的按钮条。对应于同样的

工具也有对应的按钮,象 gline 和 XF86Steup。还有写名为 RH 工具的按钮。它们会带出 Redhad 工具的按钮条,象用来管理你的文件系统的 fstool,安装打印机的 printtool,和设置你的网络的 netcfg。fstool 没有设计成和自动 mount 的守护进程 amd 一起工作,usrcfg 也没有设计被用于 NIS 网络。

4.3 窗口和图标

你在窗口中运行程序,显示信息,和对文件列表。一个窗口由几个基本的部分构成。外面的边界包括改变大小的控制。还有不同的按钮可以让你控制窗口的大小或者关闭窗口。在边界的内部是窗口的主要部分:题目栏用来显示窗口的名字;菜单用来让你发布命令;窗口面板用来显示窗口的内容。

fvwm 窗口,如图 4-1 所示,可以让你用按钮和改变大小区域来改变它的大小和形状。改变

大小区域是窗口的边界线。点住改变大小区域后移动鼠标来让窗口在高度和宽度上变大和变小。

通过单击右上方的最大化按钮(一个正方形套着一个正方形的那个按钮)你可以使窗口充满整个屏幕。要把窗口恢复到原来的大小,只需要再次单击最大化按钮。

如果你想把窗口变成一个图标,可以单击最小化按钮。它是最大化按钮边上那个小正方形中间有一点的按钮。当你把一个窗口变成图标以后,你可以双击这个图标来打开它。

你可以通过选定它的题目栏或者边界(不是角落)在桌面上移动任何窗口。把你的鼠标移动到窗口的题目栏上,点住并在你移动鼠标的时候按住它。你就会看到窗口被移动了。当你到达你想要的位置的时候,松开鼠标的按钮。仅仅单击题目栏将把该窗口从层叠窗口中移动到最前。把鼠标移动到窗口的边界直到你看见鼠标指针变成一条短的直线。然后点击住边界并拖动鼠标指针。你会看到整个窗口被移动。

为 X-Windows 设计的应用程序有它们自己的菜单,按钮,甚至在它们窗口中的图标。在这样的程序中你通过菜单和图标来发布命令。如果你在运行一个象编辑器这样的应用程序,窗口的内容就是菜单操作的数据。如果你在使用文件管理器,内容就会是代表文件或者目录的图标。一些窗口,象终端窗口,可以没有菜单。

你可以同时打开几个窗口。但是,同一时刻,只能有一个窗口处于活动状态。活动窗口的边界是紫色的,非活动窗口边界是灰白色的。把你的鼠标指针移动到一个非活动窗口上可以使它成为一个活动窗口,而使其他窗口成为非活动窗口。你不需要单击鼠标,就象你在其他的图形用户界面,象微软的 Windows 或者 Mac 的 OS 中所做的那样。

层叠的窗口有时候会导致混乱。使窗口活动并不是自动的使它移动到最前。一个活动的窗口有可能被其他窗口部分的遮挡。要把一个窗口移动到最前,你需要单击该窗口的题目栏。单击该窗口的其他任何一个地方只会使它变成一个活动窗口,而不会使它移动到最前。

图标代表你可以运行的程序或者那些程序的数据文件。他们出现在你的桌面上,在文件管理器的窗口里面,下方是该文件或者应用程序的名字。要运行这个

应用程序,只需要双击它的图标。

4.3.1 终端窗口:Xterm

在 Linux 的窗口管理器中你可以打开一个叫做终端的特殊窗口,它给你提供一个标准的命令行界面。你可以在此窗口的命令行上的提示符后面输入带有参数和选项的命令。这个终端窗口是一个叫做 Xterm 的程序创建的。你可以从 fvwm 的任务栏,程序菜单,或者象 Caldera 这样的桌面中开启一个终端。fvwm 任务栏上有一个标注为 Xterm 的按钮。只要单击它即可以打开一个 Xterm 窗口。相应的,从 fvwm 工作区菜单中选中 Shells,再从子菜单中选中终端,再从那里选中 Xterm 也可以打开一个 Xterm 窗口。要从 Caldera 桌面上打开一个终端窗口,你只要双击你的桌面左上角图标栏里的终端的图标即可。

一旦打开,终端窗口就显示一个 Shell 提示符,一般是"\$",你可以象在命令行方式下

一样输入命令。你会看到你输入的 Linux 命令的结果在终端的窗口上显示出来,跟在后面的是一个 shell 提示符表明命令行的开始。必须注意的是终端窗口必须是你的活动窗口你才能使用它的命令行。如果它不是活动窗口,你需要移动鼠标指针使它变为活动窗口。

Xterm 窗口支持几个文字处理特性。在窗口的左边是一个滚动条,你可以象前滚动来查看显示过的文本。当文本移出屏幕的上方,你可以用回滚来看它。如果你在查看有很多文件无法在一个屏幕里显示的目录,这对你就很有帮助。Xterm 还让你拷贝文本再粘贴到命令行上。你用鼠标的左键来拷贝文本再用第二个键

来粘贴文本。你可以拷贝任何先前显示的文本,譬如象先前的命令或者那些命令的输出结果。要拷贝文本,按住鼠标的左键拖动到你想要拷贝的地方,到顶端的时候松开。双击将选择一个单词,三击将选择一行句子。如果你想延伸选择的文本,用鼠标的第三个按钮。当你选择好文本,单击鼠标的第二个按钮。这就自动把文本粘贴到命令行的末尾。重复单击就会重复粘贴。这种拷贝粘贴的操作在从先前的命令中创建一个复杂的命令的时候十分有用。你可以在不同的终端窗口间拷贝和粘贴。

(术语"第二个"和"第三个鼠标按钮"可能容易混淆。在一个双键鼠标上,第二个按钮就是指右键,而第三个键就是指左右键同时按下。在一个三键鼠标上,第二个键就是指中键,而第三个按钮就是指最右边的按钮。这三个按钮是用来访问 Xterm 的菜单。一个 Xterm 的窗口有四个菜单:主菜单,一个虚拟终端选项菜单,一个虚拟终端字体菜单,和一个 Tektronix 窗口选项菜单。要弹出主菜单,按住 Ctrl 键并单击鼠标左边的按钮。要弹出虚拟终端选项菜单,按住 Ctrl 键并单击鼠标的第二个(右边或者中间)按钮。要弹出字体菜单,用 Ctrl 键和鼠标的第三个按钮。你就可以设置字体和字符显示的大小)。

一个 Xterm 窗口可以仿真 DEC VT 102 终端和 Tektronix 4014 终端。有模拟这两种的窗口,尽管开始显示的只是一个虚拟终端窗口。要显示一个 Tektronix 窗口,在虚拟终端选项菜单上(Ctrl 键和第二个按钮)选择这个选项。这将弹出一个可以显示图形的 Tektronix 窗口。在 Tektronix 窗口中你可以按住 Ctrl 键并单击鼠标的第二个按钮弹出一个 Tektronix 选项菜单。终端窗口有一种特别的能力。它可以在它的命令行里运行任何 X-Windows 程序。这个终端窗口在 X-Windows 环境中运行。要运行任何一个 X-Windows 程序,只需要打开一个 Xterm

窗口输入命令,再敲击一个回车键。这个 X-Windows 程序就会在它自己的窗口里运行。例如,要运行一个 Netscape 你可以打开一个终端窗口然后输入命令 `netscape`。一个新的窗口就会打开运行 Netscape。你可以打开任意多的终端窗口,可以在它们中的任何一个中启动一个 X-Windows 程序。但是,关闭一个终端窗口同时也会关闭从这个终端窗口中启动的程序。

你会注意到在你输入 X-Windows 命令的终端窗口把自己挂起。当你输入回车键运行程序以后它并没有跟随的命令提示符。那是因为终端窗口正忙于运行你刚刚执行的 X-Windows 程序。你可以通过带有一个和符号(&)来激活该程序使终端窗口释放自己,在那个程序在运行的同时来执行其他的命令。从技术上来说,就此终端窗口而言这是把该程序放在后台运行(参阅第 5 章)。但是你可以随意移动到那个 X-Windows 程序在那里运行它。下面的例子将运行 Netscape,释放终端窗口来执行其他的命令。注意提示符:

```
$ netscape &  
$
```

当你使用完毕终端窗口,在命令行输入一个 `exit` 命令来结束它。每个终端窗口都是它自己的 shell,`exit` 就是结束这个 shell 的命令(在第 5 章里将详细讨论 shell)。图 4-1 显示了终端窗。

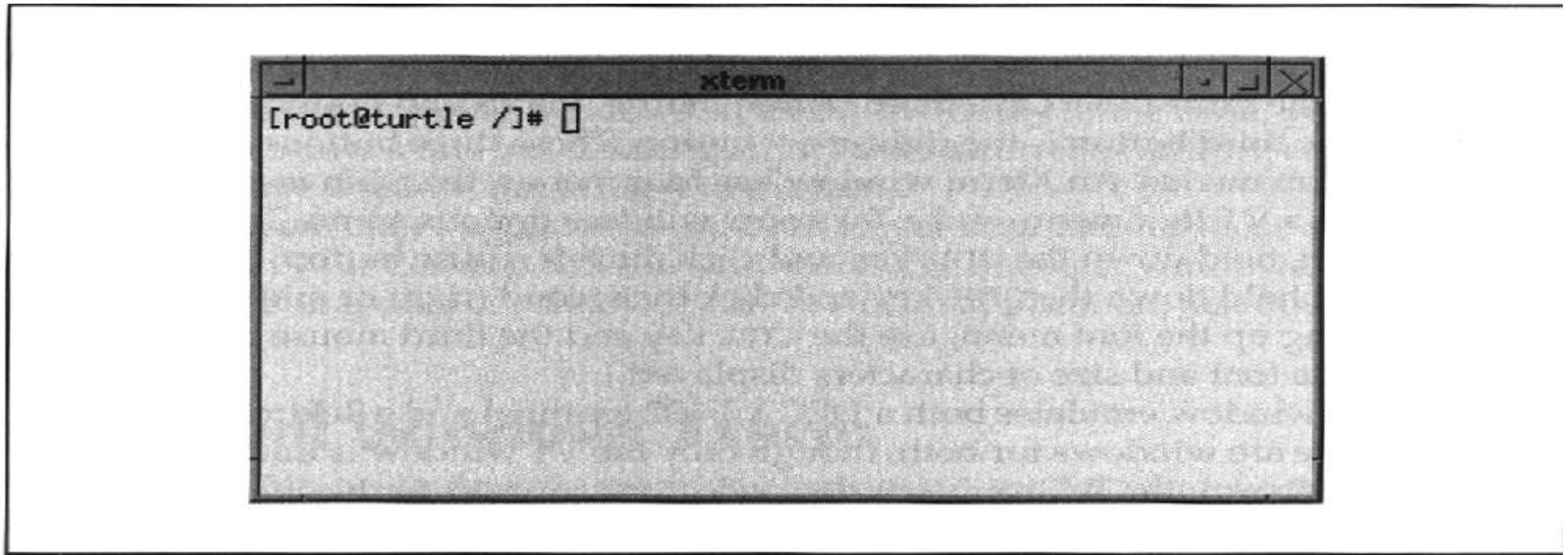


图 4-1 fvwm 终端窗口

窗口用户输入了几个命令,输出在窗口上显示。当你到达窗口的底部的时候,显示的文字将一行一行的上卷,就像一个正常的终端屏幕那样。

当然,你可以使用窗口控制来使终端窗口变大或者变小。你甚至可以把它最小化到一个图标在晚些时候再打开它。

4.3.2 X-Windows 多任务

你的 Linux X-Windows 界面最有用的特性之一就是它可以同时进行几个操

作的能 力,每一个都有自己的窗口。注意在命令行界面中你每次只可以工作在一个任务上(这里有一个例外就是将在第 5 章要讨论的后台进程)。你发布一个命令在它执行完以后,你可以执行另一个。在 X-Windows 你可以同时运行几个不同的应用。把你的鼠标指针从一个窗口移动到另一个窗口有效的使你从一个应用切换到另一个应用。X-Windows 的这个特性表明了 Linux 最有用的特性之一:并行性,可以同时运行好几个进程的能力。在你的 X-Windows 界面中,你可以有好几个应用同时运行,每个都有各自的窗口。

这个特性可以很容易的用终端窗口来演示。你可以在同一时刻有好几个终端窗口,每个都有各自的命令行。要从 Caldera 桌面上打开一个终端窗口,双击图标栏的终端图标;在 fvwm 中,使用工作区菜单或者是 fvwm 图标栏上的 Xterm 按钮。要打开第二个终端窗口,再次双击该图标。还要打开另一个终端窗口,只要再次双击这个图标,然后继续。每个终端窗口都有各自的命令行,把你的鼠标指针从一个窗口移动到另一个窗口使你从一个命令行切换到另一个。你可以在一个活动窗口中输入命令并执行它,你可以然后切换到另一个终端窗口再输入另一个命令。每个终端窗口都相互独立的运行。如果你发布一个命令需要一段时间来运行,你移动到另一个窗口,你会注意到你刚刚离开的那个窗口中的命令仍然在继续执行。

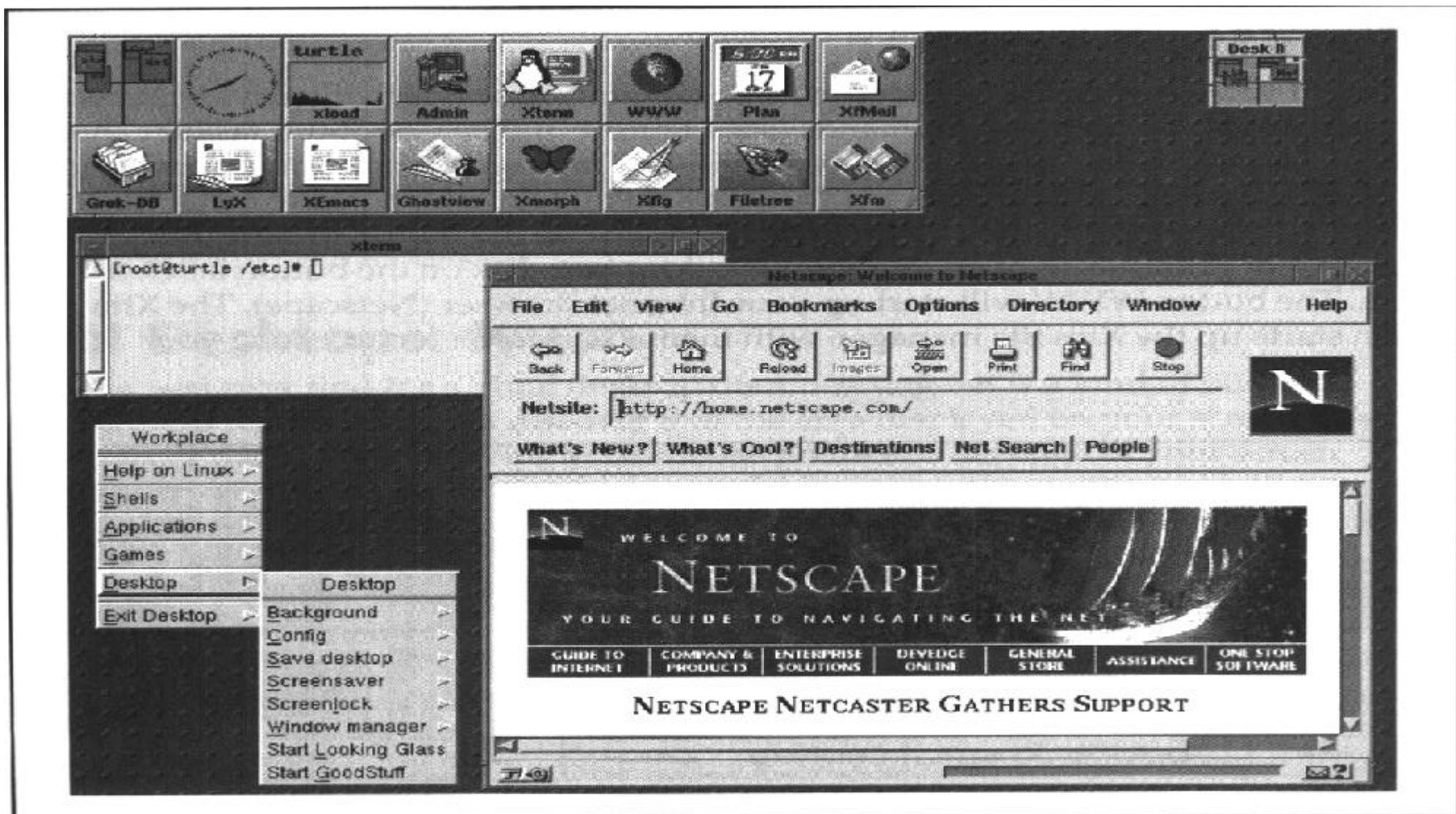


图 4-2 包括任务栏和工作区菜单的 fvwm 桌面运行有终端窗口和 Netscape

4.4 fvwm 窗口管理器和桌面

尽管 Linux 上有很多窗口管理器可供使用,这一章只涉及 fvwm 的一些基本特性。fvwm 窗口管理器不只提供了基本的窗口,任务栏,和菜单工具,而且象虚拟桌面几个高级特性。实际上,用 fvwm 你可以有好几个桌面,每个都有不同的窗口和在其中打开的程序。图 4-2 显示了 fvwm 的工作区菜单和任务栏,以及在 fvwm 中直接运行的 Xterm 终端窗口和 Netscape 浏览器。

4.4.1 fvwm 工作区菜单

要弹出工作区菜单,用你的鼠标单击然后稍微向下拖动。工作区菜单就会弹出来。这个主菜单的选项引出子菜单。选择一个选项也将带出一个子菜单。这些子菜单中的一些本身也列出其他子菜单。例如,应用将带出一个子菜单列出你所有的 X-Windows 程序类别。如果你选择 Xpaint,Xpaint 程序将会开始运行。

当你安装或者删除一个 X-Windows 程序的时候,fvwm 将会自动更新工作区菜单。经常使用的 X-Windows 程序项目已经被设置。现在,fvwm 被设置成检查在 /usr/X11r6/bin 和 /usr/openwin/bin 目录下任何新的程序。这些将被加到工作区菜单中。

要退出 fvwm,从这个菜单中选择退出桌面项目。这带出一个包括有停止桌面选项的子菜单。当你选择这个,fvwm 将会和 X-Windows 一同退出。

4.4.2 GoodStuff 任务栏

最近的 fvwm 版本包括 GoodStuff 任务栏可以用按钮来显示经常使用的 X-
Windows 命令(看图 4-3)。

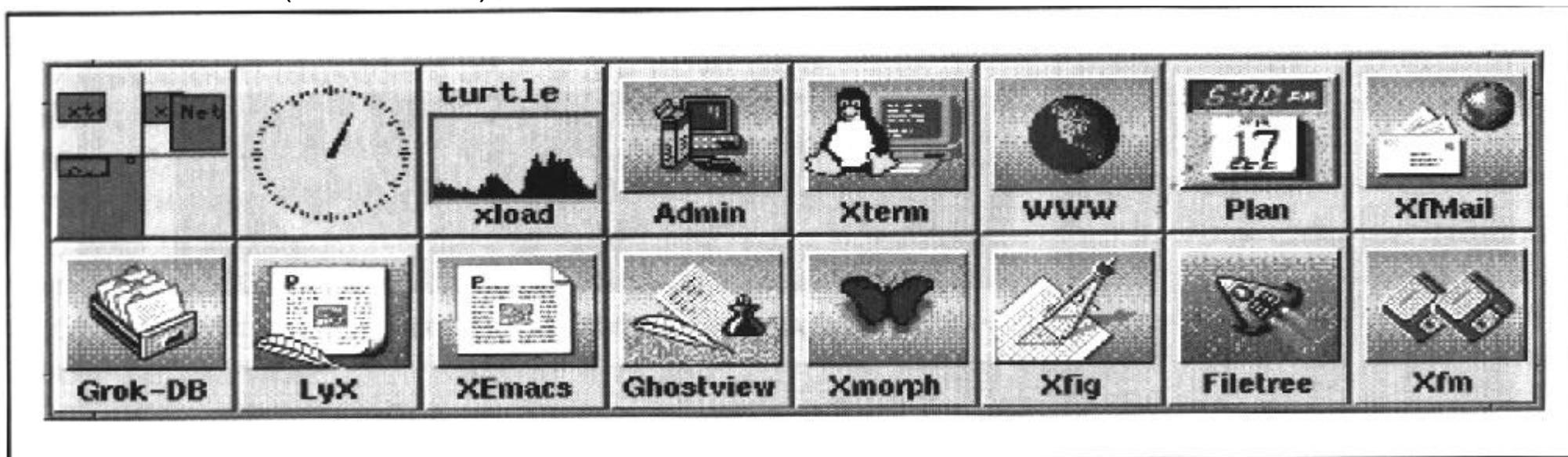


图 4-3 fvwm GoodStuff taskbar

GoodStuff 任务栏将在你启动 fvwm 窗口管理器的时候被显示；如果没有，你可以通过在工作区菜单的桌面菜单中选择启动 GoodStuff 带出它。每个按钮都将显示程序的名字。只要单击该按钮就可以启动那个程序。例如，要打开一个 Xterm 窗口，只需要单击那个标注为 Xterm 的按钮。

WWW 的按钮将启动你的因特网浏览器(Netscape 如果你有，否则就是 Arena)。Xfm 按钮启动 Xfm 文件管理器，用菜单和图标来访问你的目录和文件。如果你选择一个标准的安装，你可能没有安装所有的这些程序。如果没有，你将被

问到你是否愿意安装它们。

要使 GoodStuff 任务栏永久的显示在你的 fvwm 桌面上,在 fvwm 工作区菜单的桌面上的设置菜单的 GoodStuff 菜单中选择位置选项。你可以选择把它放在你的屏幕的顶部,中间,左边,或者右边。你可以随意的移动或者重置任务栏的大小。要移动任务栏,用鼠标右边的按钮点住它的边缘,然后拖动到你想要移动到的地方。要重置大小,用鼠标左键单击再拖动它的边缘或者一角。

如果你不想再显示任务栏,在位于 fvwm 工作区菜单的桌面上的设置菜单中的 GoodStuff 菜单中选择"无效"选项。

任务栏的配置文件叫作 system.fvwmrc.goodstuff。它位于/usr/lib/X11/fvwm目录下。文件的底部有任务栏的按钮的条目。你可以通过在一行的前面加上一个#来删除一个按钮。你还可以给新按钮添加其他行。一个按钮项目用关键字*GoodStuff 开始,跟随其后的是你想显示的按钮和你想来在按钮上显示的图片。这行结束的部分是要运行的命令,通常用 Exec 命令开始,跟着一个空字符串和 xlaunch 命令然后是程序名。下面的例子给 Crisplite 编辑器添加一个按钮。激活 Crisplite 编辑器的命令是 mcr。

```
*GoodStuff Crisplitedtp.xpmExec "" xlaunch mcr
```

.xpm 文件就是用来显示的图片文件。你可以在/usr/share/data/pixmaps 和 /usr/openwin/lib/pixmaps 目录下找到很多这样的文件。

4.4.3 虚拟桌面: fvwm

开始的时候,你可能会觉得虚拟桌面令人不安——它是一种内置放大特性。

你将发现在你的屏幕上显示的区域可能只是桌面的一部分。移动你的鼠标指针到你的屏幕边缘移动屏幕到桌面隐藏的隐藏的部分。你还会注意到在你的桌面显示的右上角有一个小方块。这叫做 pager,你用它来查看你的虚拟桌面的不同区域。这个 pager 被分成四个小正方形。你可以把每个正方形想象成一个独立的桌面扩展。它就像你有有一个很大的桌子,只有部分被显示在屏幕上。这个桌子活动的部分就是加亮的正方形,通常是白色的。这就是现在显示在你的屏幕上的那部分桌面。

通过移动这个加亮的正方形你可以移动到桌子的不同地方。你用鼠标的右键来单击和拖动这个加亮的正方形。你也可以单击 pager 四个部分中的一个来移动到桌子的那个部分。你可以把不同的窗口放在你的桌子的不同部分然后当你需要它们的时候再移动到那个部分去。用这种办法,你想放在你的桌面的所有东西就不一定要一起显示在你的屏幕上,使你的屏幕显得凌乱不堪。

如果你在桌面上工作,突然所有的东西都不见了,这可能是你碰巧单击了其中的一个正方形。只要再次单击右上方的正方形就回到你原来的显示。你就可以看到你原来的窗口重新出现了。

某些图标始终显示在你的屏幕上,不论你的虚拟桌面显示在那个部分。这些被称为“粘性”(sticky)图标。fvwm pager 就是这样一个图标,和 fvwm 任务栏一样。例如,fvwm pager 总是显示在你的屏幕的右上角上,而不管你在查看你的虚拟桌面的哪一部分。而另一方面,窗口不是粘性的。它们停留它们原来在虚拟桌面的位置上。

Caldera 桌面既可以是粘性的停留在屏幕显示上也可以是一个窗口,保留在虚拟桌面的一个窗口中。

Calder 任务栏最左边的图标使其在粘性和窗口之间切换。如果它是一个窗口,你将看到桌面有一个标准的 fvwm 窗口边界,你可以随意移动或者重置大小。作为一个粘性图象,窗口没有边界。不论屏幕移动到虚拟桌面的哪里,Caldera 桌面在屏幕上的显示都带有任务栏,菜单和图标。

你放置在你的桌面上的图标每次都将显现在相同位置。这样,你就可以访问桌面上的一个图标。

开始的时候,你的虚拟桌面的屏幕区域可能不到你的 pager 的一个部分。你可以在 /etc/XF86Config 文件中通过把虚拟选项设置成你的显示器的分辨率,就像第 2 章中所写的那样。在大多数 15 英寸的显示器中,这就是 800*600。

大多数窗口管理器,包括 fvwm,也支持多种桌面。这个桌面,在这个意义上,是一个屏幕中包含了图标,菜单,和窗口。象 fvwm 这样的窗口管理器让你可以使用很多种不同的桌面.fvwm 可以支持四十亿个桌面。

4.5 Caldera 桌面

Caldera 桌面是一个集成的程序和文件管理器,给你提供菜单和图标使你可以管理你的文件,运行程序和设置你的系统。Caldera 桌面提供了对很多 Linux 可用的软件程序以及因特网工具很方便的访问。它是和 Caldera 商业包一起提供的一个商业产品。Looking Glass 桌面是和 Caldera Lite 一起提供的一个九十天的试用版本(在 fvwm 工作区菜单中,Caldera 桌面就是指 Looking Glass)。你可以利用 Caldera 桌面的全部特性,象工具栏,设置工具,文件管理窗口和自动历

史列表。

当你启动 Caldera 桌面的时候,你会注意到在顶部显示有一个图标栏。开始的时候它包括代表了常用的 Linux 操作的图标,象打开一个终端窗口或者查找一个文件。这是一个可设置的图标栏。你可以添加你自己的特定操作的图标。这个桌面还有很多种文件管理器。对于每个目录,你可以打开一个窗口查看该目录中所有的文件,显示为图标。你就可以用双击它们的图标来运行应用程序,或者你可以把文件从文件管理器的窗口移动到桌面上以便很方便的存取。

桌面是一个你可以放置图标的窗口。右边和底部的滚动条可以让你在有很多的图标不能装在一个屏幕里的时候,在这些图标之间移动(见图 4-4)。你会注意到在你的桌面窗口上已经有几个图标。例如,一个表示 Crisplite 编辑器的图标和另一个表示 Netscape 浏览器的图标。你可以随意的删除或者添加这些图标。

桌面顶部的任务栏包括了 Linux 常用的操作,象文件管理器和终端窗口。你可以在任务栏中添加你自己的图标,也可以删除那里已经有的图标。图标栏的右边是你的时钟和一个用来获得帮助的图标。桌面的帮助特性将在你指向的桌面的任何一个部分显示气球状线圈围成的帮助文字。

在图标栏之上是一个标准的菜单栏。不同的选项包括从设置桌面到发布 Linux 命令之类的操作。

你已经在文件菜单中看过退出桌面选项。请注意每个菜单名字中加下划线的字母。这表示一个快捷键,你可以用它来执行一个命令。例如,要打开一个文件菜单,你可以用按 F 来代替把你的鼠标移到菜单栏上来按下文件项目。

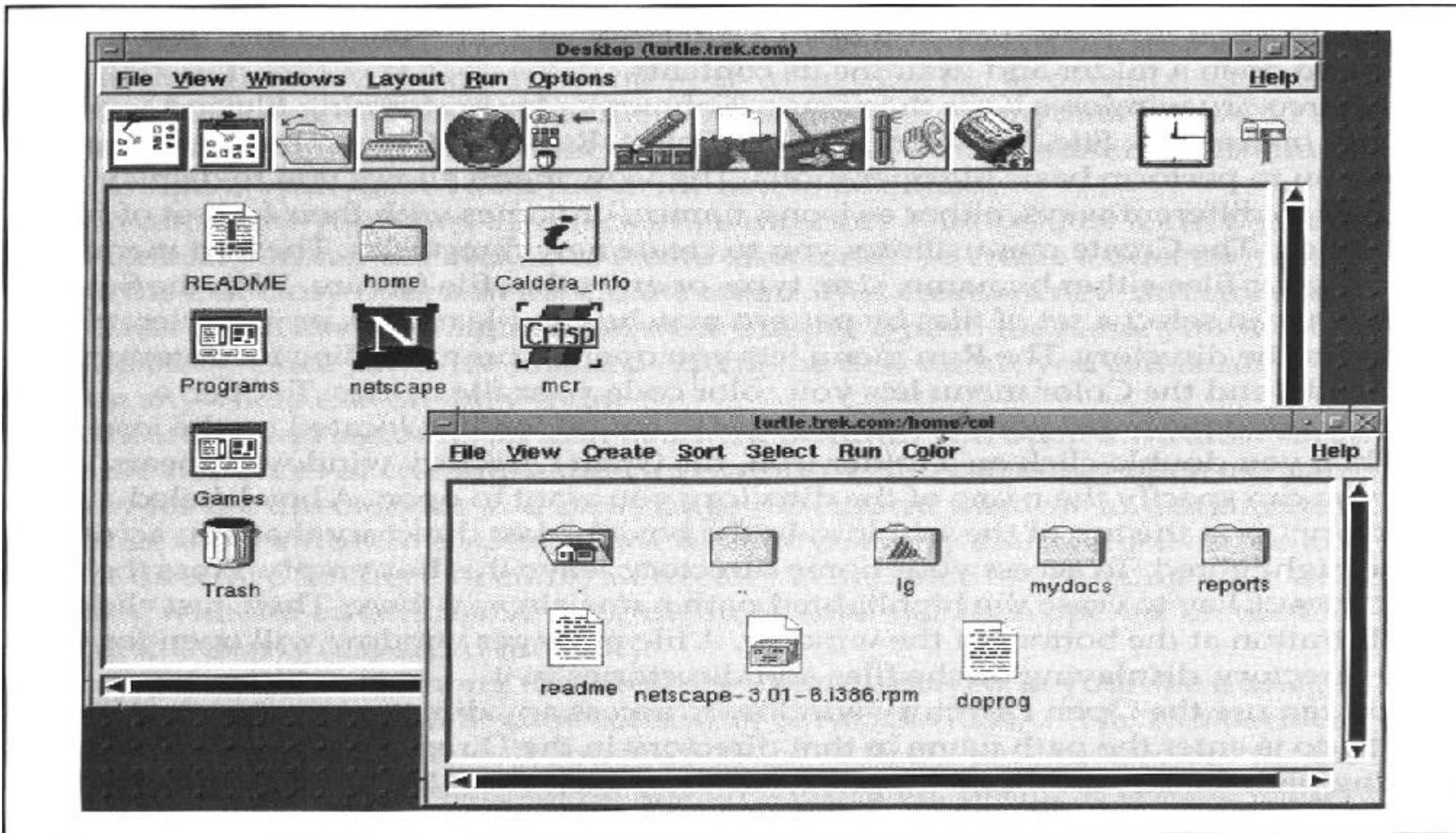


图 4-4 The Caldera Desktop

通过用鼠标在你的屏幕上选中不同的窗口,菜单,和图标,你可以运行和那个图

标相关的应用程序,为它打开一个新的窗口。

鼠标的右键用来打开一个文件下拉菜单。不用把鼠标移动到菜单栏上,你只要单击鼠标右键就在你的屏幕上的那个位置打开一个文件菜单。然后你就可以选择一个选项,譬如退出桌面。

4.5.1 目录窗口

除了让你输入命令和文件名字参数的命令行,Caldera 桌面上的目录窗口让你可以使用菜单,图标,和窗口来管理文件和目录。在桌面中,一个目录就表示为一个窗口。不用再用 ls 命令来列出你的目录中的文件,文件已经被显示成图标。目录被表示成象文件夹一样的图标。这个想法是因为目录象文件夹一样存放文档——那就是,文件。你打开一个目录查看其中的文件,就像你打开一个文件夹查看其中的内容。

所有的目录窗口都有相同的菜单集合用来完成文件和目录操作:文件,查看,创建,排序,选择,运行,和设置颜色。文件菜单让你完成基本的文件操作。查看菜单让你可以用不同的方式显示你的文件,可以是图标,名字,或者是带有全部文件信息的名字。创建菜单让你创建新的目录。排序菜单让你用名字,大小,种类,或者任何其他文件特性给文件排序。用选择菜单,你可以用文件名字的模式匹配来选定一系列文件,或者选中该目录下所有文件。

运行菜单让你打开一个可以输入命令的命令行。而设置颜色菜单让你对你的文件名字着色编码。要打开一个目录窗口,把你的鼠标指针移动到图标栏上文件夹的图标。

当你双击一个文件夹的图标,打开目录窗口就出现;你可以在那儿指明你要打开的目录。一个标注为"目录:"的盒子出现在窗口的上方。在这个盒子中,你最近存取的目录被加亮显示.清空这个盒子,就可以访问你的工作目录。按下 DEL 或者 BACKSPACE 键来删除那里已经有的加亮显示的路径名。然后,单击窗口底部的 OK 按钮。一个你的工作目录的文件管理器的窗口就会打开,并显示其中所有的文件和目录。

你可以用打开目录窗口来访问你希望的任何一个目录。你所要做的只是在目录框中输入那个目录的路径名。打开目录窗口包括一个历史框列出了你访问过的所有目录的路径名。如果你看见其中有你想访问的目录,只需要单击它,它的路径名就出现在标有"目录:"的框里。

通过在查看菜单里取消选定可定位的选项,你可以把目录窗口分成上下两个部分。表示目录的文件夹图标出现在上方的面板中,在下面的面板中文件的图标上由它们用于的应用程序而显示不同的图象。在每个面板的右边是一个滚动条,你可以用它滚动来显示在你的特定目录下的目录或者文件。你还可以改变每个面板的大小。在两个面板之间,和他们的滚动条相邻的地方是一个小条,你可以用它使一个面板变大而另一个变小。把这个小条向上移动使目录面板变小而使文件面板变大。把它向下移动则正好相反。

图标一旦被选中,你可以在它上面执行一种操作。例如,要拷贝一个文件,选择那个文件的图标,然后在文件菜单中选择拷贝选项。一个特别的窗口就会打开,提示你输入要拷贝的文件名称。要删除一个文件,选择文件的图标然后在文件菜单中选择删除选项。这个文件就会被删除,它的图标也会从目录窗口中删除。

在一个图标上基本的操作之一就是打开操作。你可以通过把鼠标移动到一

个图标上双击或者单击这个图标再从文件菜单中选择打开选项来打开一个图标。不同的操作将被执行,取决于这个图标是一个文件,目录,还是应用程序。如果你打开一个应用程序,那个应用程序的窗口就会打开,应用程序开始执行。另一方面,如果你打开一个目录,另一个目录窗口将被打开,显示该目录下的图标。如果你打开的是一个文件图标,和那个文件相关联的应用程序将使用那个文件并执行。例如,如果 mydata 文件是一个 Vi 编辑器数据文件,那么当你双击该文件的图标的时候,Vi 编辑器应用窗口就会打开并且 mydata 文件就是要编辑的文字。

你可以在目录窗口中创建你自己的目录,就像在命令行界面下,你用 mkdir 来创建目录一样。要创建一个目录,首先在创建菜单中选中目录选项。这就创建了一个新的目录图标,它的下面是加亮显示的缺省名 Dir1。只要输入你希望的新的目录名就可以给它更名。一旦你完成了输入新的名称,你可以通过在窗口上其他的任何地方单击来取消选择它。

目录作为一个图标存在于窗口中直到你为它打开一个窗口。有两种方法可以打开一个目录窗口。你可以给这个目录打开一个单独的窗口,你可以可以把当前的窗口改变到那个目录下。要改变这个窗口来显示一个新的目录,只要双击那个目录的图标。在效果上,打开一个目录和用 cd 命令来改变你的工作目录是相同的。你的菜单选项就可以在新显示的目录中的文件和图标进行操作。

你会注意到在你的文件管理器窗口中显示的目录图标里有一个名字有两点构成的目录,..。两点表示父目录。如果你想移动到当前目录的父目录,只要双击这个目录的图标。这样做和用 cd ..命令是等效的。

你用文件菜单中的移动选项来移动一个文件或者给文件改名。要给文件改名,选中文件的图标在选择移动选项。这会打开一个在顶部标有"移动到:"的框的

对话框。你只要输入那个文件的新名称。你还可以有其他的选项,只用你的鼠标,你可以使用单击再拖动的方法把文件从一个目录移动到另外一个目录。首先,为我想把文件移动到的目录单独打开一个窗口。然后把鼠标移动到你想移动的那个文件的图标上。按下你的鼠标并在你把图标拖动到你想把文件移动到的那个目录去的时候不松开鼠标。然后松开鼠标按钮。那个文件的图标就会出现在新的目录中。

至于移动操作,有两种方法可以拷贝一个文件:或者是用鼠标选中再拖动,或者是用文件菜单中的拷贝选项。使用文件菜单,首先选择该文件的图标,然后选择拷贝选项。这就打开把东西拷贝到目录的对话框窗口。在窗口上部标有"拷贝到:"的框里输入拷贝的名字。然后单击对话框窗口下方的 OK 按钮。

使用鼠标和 CTRL 键,你可以用单击再拖拉的方法把文件从一个目录拷贝到另一个目录。用来拷贝的单击再拖拉的方法和用来移动的不大一样。因为你必须在整个过程中按住 CTRL 键。在移动文件中,你首先为我想把文件拷贝到的目录打开一个窗口。然后把鼠标移动到你想拷贝的文件上选择它。按住 CTRL 键,并在你把图标拖动到新窗口的时候按住并保持你的鼠标左键。松开鼠标按键和 CTRL 键,该文件的图标就出现在另一个目录里。

你可以删除文件。或者是用把文件的图标拖动到垃圾箱的图标上,或者是在文件菜单中选中删除选项。如果你把你一个文件拖动到垃圾箱,它还会存放在那里直到你清空垃圾箱。你还可以删除整个目录,如果你愿意的话,通过把他们的目录的图标拖动到垃圾箱里。

你可以通过在文件菜单中选择信息选项来设置一个文件或者目录的权限。这将打开一个有三个选项的窗口:访问,属主和时间。单击访问的按钮会打开一个

有访问权限列表的窗口。在这里,你可以单击按钮来指明属主,同组人和其他用户的读,写,和执行权限。(权限将在第 7 章详细解释)。属主的选项允许你改变一个文件的主人。

4.5.2 管理 Caldera 桌面

桌面是一个你可以为你的需要定制的一个界面。你可以在你的桌面上放上文件,程序,和目录的图标来使访问更加容易。你可以创建你的桌面的布局,保存一套你的桌面上特定的打开的窗口和图标的快照。然后你就可以通过在你的图标栏上指明项目或者设置文件怎样被创建,显示,和删除的首选项来设置你的桌面。

当你在桌面上工作时,你可能需要为不同的任务打开不同的图标,窗口,或者程序。例如,如果你在和一个数据库打交道,你可能打开一个数据库的窗口和一个用来存放数据库文件的目录窗口。要做图形工作,你可能在你的桌面上有图形程序的图标和存放图形文件的目录。不必要在你从不同任务之间切换的时候打开和关闭各个窗口,你可以把你整个的桌面保存为一个定制。对于数据库工作,一个用户可能为它创建一个名字象"数据工作"这样的定制然后在他或者她需要在数据库上工作的时候再打开这个定制。

已经提供的有好几种定制。root 和管理员定制对当你作为一个 root 用户登录来做一些管理的操作的时候是很有用的。它们包括一个用来访问设置工具的管理工具图标。第 3 章的图 3-1 展示一个 root 的定制。用户缺省的定制包括一个时钟和图标栏上几个常用的按钮。最左边的按钮将把桌面扩展成整个屏幕并

把它变成粘性,在虚拟桌面上随着你移动。再次单击使它分解成一个窗口。图标栏上的下一个按钮是用来给出一个小一些的桌面窗口或者使它成为粘性图象。图 4-4 展示了带有用户缺省定制的 Caldera 桌面。

要创建一个定制,把你的桌面设置成你希望定制的那样,在桌面上打开你想打开的窗口再放上你想放置的图标。然后在定制菜单中选择定制选项。这就打开一个定制窗口。在标有“定制”的框里选择当前的选项(第一个选项)。在定制框的下面是一个名字区域。单击它在输入你想给你的定制取的名字。然后,在窗口底部的按钮中单击保存按钮。单击关闭按钮来关闭定制窗口。新的定制就会在定制菜单中列出来。要恢复它,只要从此菜单中选中这个选项。

4.6 安装窗口管理器和文件管理器

你的 OpenLinux Lite 系统提及 Caldera 桌面的时候,只是指 fvwm 窗口管理器和一个所有权文件和叫做 Looking Glass 的程序管理器。Caldera 桌面程序的名字叫做 lg。用 OpenLinux Lite,你被授予三个月的 Caldera 桌面免费使用期,然后它就不可使用了。你可以继续使用 fvwm 窗口管理器和它自己的程序管理器和任何一种公开许可的的文件管理器。还有 GNU 的文件管理器可供使用。用这样的文件管理器,你可以使用图标和菜单,而不是只用 fvwm 提供的窗口。

在你的 OpenLinux Lite 的 CD-ROM 中还有 Xfm 的文件管理器。你可以用 glint 或者 Lisa 来安装它。

在网上有许多不同的文件管理器你可以下载和安装。你可以从 Linux 的 ftp

镜像站点来下载文件管理器,通常是在/pub/Linux/X11 的目录下。Linux 应用和实用程序页列出了几个文件管理器的站点,你可以连接到和下载软件(看第 3 章的表 3-2)。如果你知道你需要的文件管理器的包的名字,最好是从 Redhat 的 contrib 目录下,ftp.redhat.com/pub/contrib/i386 下载它的 RPM 版本。

作为 fvwm 窗口管理器一个替代品,你可以用其他窗口管理器象 Xview, fvwm32, fvwm95, LessTif, AfterStep, and Motif. 这些当中也带有文件管理器给你一个完整的桌面界面。你可以从 Redhat 的 ftp 站点的 contrib 目录下下载这写中的任何一个窗口管理器的 RPM 版本。

如果你使用 ftp,用带有那个窗口管理器的名字的开头几个字母和一个星号的 ls 命令来列出它的 RPM 文件。当然,用象 Netscape 之类的浏览器来下载要容易的多。绝大多数这些窗口管理器也自己带有它们的 Web 站点。你可以从那里下载原始的代码。

4.6.1 启动窗口管理器

一般,你应该可以先启动 fvwm,然后在 fvwm 的工作区菜单中的桌面选项中的窗口管理器的菜单中选择那个窗口管理器的名字来启动绝大多数窗口管理器。那儿有 fvwm95, olvwm, 和其他(但是没有 AfterStep)的选项。如果你想直接启动窗口管理器,你必须把一个选项放在 .xinitrc 文件里。第 21 章详细讨论了怎么设置一个 .xinitrc 文件和在其中放置一个窗口管理器的调用。

你的 home 目录开始没有一个 .xinitrc 文件。当你启动 X-Windows, .xinitrc 文件将被用来设置你的 X-Windows 过程,而不用系统的 .xinitrc 文件。系统

的 .xinitrc 文件在 /usr/X11R6/lib/X11/xinit/xinitrc。下面的命令生成它的一个备份。要确定你是在你的 home 目录下。注意 .xinitrc 文件有一个前置的点号,而系统的 xinitrc 文件则没有。

```
$ cp/usr/X11R6/lib/X11/xinit/xinitrc .xinitrc
```

你必须启动你想要的窗口管理器你只需在你的 .xinitrc 文件末尾加上命令。

在其它的窗口管理器的命令所在的命令行前，确认加上一个 #号来使其无效。文件的别的地方不要动。下面的例子演示了取消命令 fvwm,并取而代之的运行 fvwm95 的指令。

```
# start the window-manager and redirect output to xconsole  
# some applications are started by the window manager itself  
# exec fvwm >/dev/console 2>&1  
exec fvwm95
```

要回到 fvwm，只需将它的命令中的那个符号 #去掉，并在 fvwm95 所在的那一行前加上一个。

如果你还想自动运行其它的程序，例如，一个文件管理器，你可以在窗口管理器的命令行前加上这些程序的命令。在命令行末尾加上一个 (&)。第 21 章更详细的讨论了这个。下面的例子在 AfterStep 窗口管理器启动时运行了 Xfm 文件管理器。

```
xfm &  
exec afterstep
```

4.6.2 fvwm 和 fvwm2

当有新版本的 fvwm 时你可以轻松的下载并安装。本书随书配套的 OpenLinuxLite 光盘上有 fvwm 的 1.24r 版本。这个当前版本支持诸如任务条, 工作区域菜单和模块这些特征。模块是一些小程序, 它们可以动态的加载和运行, 和 Mac 里的 extension 很类似。

现在正在开发 fvwm 的 2.0 版本, 称作 fvwm2。版本 2.0.46 正在以 beta 版的形式发放。fvwm2 扩展了 fvwm 的功能, 以便提供更好的配置文件, 允许定制各个窗口, 及提供更好的模块支持。例如, 面版模块在 fvwm2 下配置了一个 CDE 一样的任务条 (CDE 代表 Common Desktop Environment, 公共桌面环境, UNIX 图形接口的新标准。) fvwm2 是现在 fvwm 的发展项目, 将会代替 fvwm 1.24。当 fvwm2 的新版本出来时你可以从 fvwm 的网页上得到它们, 现在的地址是 ww.hpc.uh.edu/fvwm.html。新版本的 fvwm 的 RPM 包装也可以在 Redhat 的 ftp 站点下的 Redhat contrib 路径下找到: fvwm2-2.0.46-4.8386.rpm。fvwm2 的第一个完整的发放版本就要完成了。

4.6.3 Xview: olwm 和 olvwm

fvwm 窗口管理器的一个替代选择是 Xview。Xview 是 Sun 系统的 OpenLook 界面的实现。熟悉 OpenLook 的人将发现 Linux 版本运行几乎一样。你可以从 sunsite.unc.edu 这个 ftp 站点下载 Xview, 在 /pub/Linux/libs/X/xview 路径下, xvview-3.2p1.4.bin.tar.gz 文件。这个 Xview 包既包括了 olwm 也包括了 olvwm

窗口管理器，还有几个工具，例如一个时钟和一个文本编辑器。Xview 的 olvwm 版本支持虚拟桌面。

你的 OpenLinux 光盘上的 Xview 包里有一套共享的库,提供了 OpenLook 菜单,按钮,和其它的窗口小部件。这些在许多 Linux 应用程序里用得到。虽然使用这些 Xview 窗口小部件的应用程序不用使用 Xview 窗口管理器,可是它确实使用了 Xveiw 的库。因此你可以在 fvwm 窗口管理器下运行象进行文本编辑的 Xview 编辑器这样的程序。文本编辑器使用的按钮和菜单是 Xview 的窗口小部件,虽然他可以在一个 fvwm 窗口下运行。你的 OpenLinux 的 Xview 包不是完整的 Xview 窗口管理器。它是库的一个子集,包括一些常用的应用程序。

要安装完整的 Xview 窗口管理器,首先从 sunsite.unc.edu 下载它的 tar 文件。然后你得去掉 OpenLinux 的 Xview 包的安装(使用 Lisa)。检查一下以确认目录 /usr/openwin 已经完全被删除了。如果不是,到 /usr 路径下(使用 cd/usr)用 rm -r openwin 命令将它删除。然后使用 tar xvf 命令解开你下载的 Xview 包。这在你的当前路径下创建了一个叫 usr 的子目录。在这个 usr 目录下有一个叫做 openwin 的子目录,把它移到 /usr 路径下(注意斜杠)。

```
# mv usr/openwin/usr
```

然后运行 /usr/openwin/lib 目录下的 ldconfig 命令。这将安装 Xview 库。

```
# ldconfig/usr/openwin/lib
```

确认一下 /usr/openwin/lib 项在 /etc/ld.so.config 文件里。因为 Xview 文件不是一个 RPM 包,你必须自己完成安装任务,例如将文件移到正确的路径下和运行 ldconfig。别的窗口管理器可以得到 RPM 格式的,那可以自动安装和为你设置它们的文件。

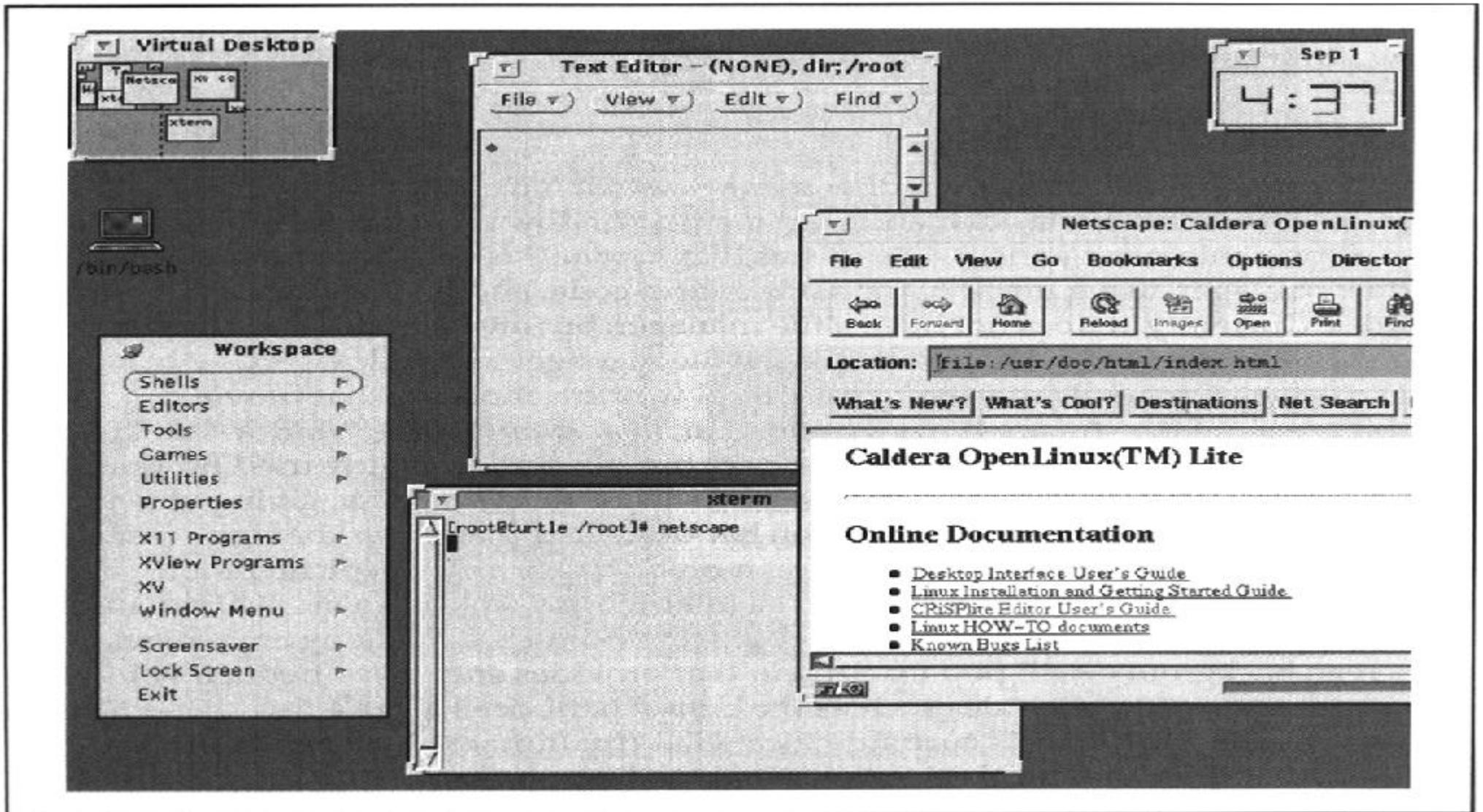


图 4-5 Xview Openlook 窗口管理器

从 fvwm 开始 olwm, 你可以通过在 fvwm 工作区域菜单的"桌面"项的窗口管

理器菜单中选择“切换到 olvwm”，也可以使用一个修改过的 .xinitrc 文件来调用 olvm 或 olvwm 来代替 fvwm。最好通过一个 .xinitrc 文件开始来使用 olvwm。olvwm 提供了一个虚拟桌面，象 fvwm 一样。当你第一次启动 olvwm 窗口管理器时，你会看到一个空白的屏幕，有一个带六个矩形区域的呼叫器显示在左上角。双击任何一个矩形将移到那个屏幕。要显示出 Xview 工作区域菜单，用右键在屏幕的任意位置单击。工作区域菜单是一个可以别住的菜单。你可以在顶部看到一个图钉形状的图案。如果你单击它，工作区域菜单将保持显示在屏幕的那个地方。再单击一下图钉就使得可以移动工作区域菜单。工作区域菜单上的 'X11 Program' 项自动列出已安装了的 X11 程序，而 'Xview Progs' 项列出了所有 Xview 程序(安装在 /usr/openwin/bin 里的那些)。

窗口组件比较 fvwm 起来略有不同，虽然它们提供的功能大致相同。菜单用右键显示。在工作区域菜单里，你用右键单击菜单项右侧的小三角就可以显示出子菜单。要移动窗口和别住的菜单你用左键拖动窗口的某条边(不是象在 fvwm 中那样用右键)。见图 4-5。

Xview 组件安装在 /usr/openwin 目录下。这里你将发现一个子目录，例如是 /usr/openwin/bin，里面有 Xview 的程序；和 /usr/openwin/lib，里面有 Xview 的库。当 X-Windows 启动时，它必须有一个叫做 OPENWINHOME 的外壳变量，设置成 Xview 的路径，/usr/openwin。如果你检查一下前一节讨论过的你的 .xinitrc 文件，你将看到做出这样的赋值的一项。

/usr/openwin/lib 目录下有菜单文件，这是用来显示 Xview 工作区域菜单和子菜单的。你可以象这些菜单中添加项，如果你愿意。一项包括要显示的标签和将采取的行动。例如，要在工作区域菜单里为 Netscape 加一项你得把下面这

项放到 openwin-menu 文件里。xview 的 man 页面里有 Xview 菜单和你能定制的项的详细解释。

```
"Netscape"          exec/usr/bin/netscape.
```

你的光盘里有一个非常好的 Xview 文件管理器，叫做 treebrowser，它可以用命令 tb 来激活。也有一个 Linux 版本的 XVfilemanager 可用，你可以从 XVfilemanager 网站(见表 4-1)下载它。按照 INSTALL 文件的指令来安装它。装上了任何一个文件管理器之后，你就有了一个完整的桌面。你也可以在诸如 fvwm 或 AfterStep 的别的窗口管理器下使用 XVfilemanager 或者 treebrowser，只要你保留 Xview 的库。

4.6.4 fvwm95 和 qvwm

fvwm95 和 qvwm 是 Linux 的窗口管理器，有一个 Windows 95 的用户界面，连带有任务条和启动菜单。窗口组件是一样的。你甚至可以将一个窗口最小化到任务条上。fvwm95 是基于 fvwm2 窗口管理器的，使用了很多相同的源代码，而且象 fvwm2 一样，它支持模块。现在，它没有文件管理器，不过有一个正在开发中。它叫做 explorer 而且将操作起来与 Windows 95 里的文件管理器一样。

fvwm95 窗口管理器有几个窗口模块提供和 fvwm 类似的功能。在 Windows 95 的任务条之外，它有一个按钮条，操作起来象 fvwm 的任务条一样。象 Xterm 这样频繁使用的程序有自己的按钮。它还有一个呼叫器，使用起来和 fvwm 的呼叫器一样，给你提供六区域的虚拟桌面。一个小按钮条将带出一个带有很少的应用程序的任务栏。一个呼叫器模块显示出一个大一些的呼叫器，它描绘四个标

有象因特网和开发这样的不同任务的桌面。你可以在桌面之间自由移动，就像在区域之间一样。在屏幕背景的任何地方单击都会弹出一个菜单，提供常用功能项，就像按钮条这样的模块一样。见图 4-6。

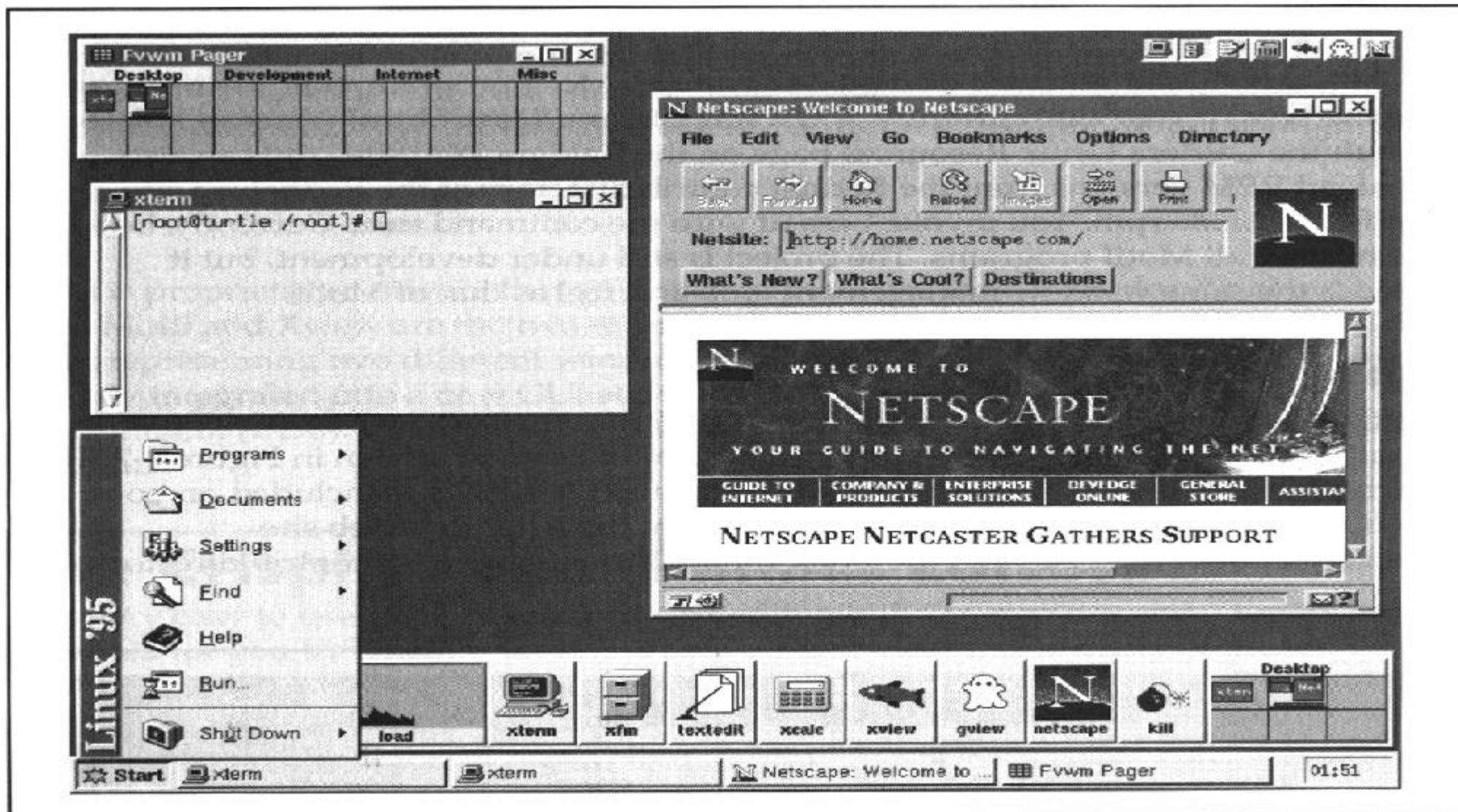


图 4-6 带有启动菜单,按钮条,pager,Xterm窗口和运行着 Netscape 的 fvwm95 窗口管理器

fvwm95 网页现在在 <ftp://mitac11.uia.ac.be/html-test/fvwm95.html>, 有一个官方的镜像站点在 www.terraware.net/ftp/pub/Mirrors/FVWM95/fvwm95.html。(你可以从 fvwm 网页连到它, 而从 www.caldera.com 网址的 Linux 资源页可以连到前者)。对你的 OpenLinux 系统, 最好从 Redhat ftp 网址, [ftp.redhat.com/put/contrib/i386](ftp://redhat.com/put/contrib/i386) 下载软件的 RPM 版本, 虽然你也可以从 fvwm95 网址下载它。还有几个图表包你应该安装。另外, 如果你希望进行 OpenLinux Lite 的标准安装, 你就得先从你的光盘上安装 X-Windows 开发库。现在的 fvwm95 包列表如下和更新的版本核对一下。用 rpm -i 来安装它们。

`fvwm95-2.0.43a-2.i386.rpm`

`fvwm95-icons-2.0.43a-2.i386.rpm`

`fvwm95-mini-icons-2.0.42-1.i386.rpm`

`fvwm95-modules-2.0.42a-1.i386.rpm`

`fvwm95-utils-2.0.42a-1.i386.rpm`

qvwm 是一个很稳定的 Windows 95 样式的窗口管理器, 它是用源代码开发的。它是一个功能齐全的虚拟窗口管理器, 带有 9 个区域的呼叫器。快捷方式可以移到桌面上的任何位置。

但是, 它现在还是处于 beta 测试阶段的产品。一个带 qvwm Linux 可执行的 RPM 包可以在 Redhat 的 contrib 目录下找到, 你可以在 qv.myth.co.uk 得到从 qvwm 镜像站点下载的大约最新版本的源代码, 或者从表 4-1 列出的主页。使用简单的 `xmkmf` 和 `make` 命令你可以轻松的编译和安装 qvwm。qvwm 要求 `libg++` 和 `xpm` 库, 所以在试图运行这个窗口管理器之前确认一下他们已经安

装。

qvwmm 的配置文件是 `qvwmm.system`，在 `/usr/X11R6/lib/X11/qvwmm` 目录下。用户也可以创建他们自己的配置文件，叫做 `.qvwmmrc`。qvwmm 的 man 页里面详细说明了如何做“启动”菜单项和快捷方式。“启动”菜单项在配置文件里有一个项，图标，命令，都用双引号引起来。对于快捷方式项，输入快捷方式名字，你想快捷方式使用的图标，和将执行的命令。图标可以是任何像素图象。记得把像素图象放到 qvwmm 的 `pixmap` 目录下。下面是一个 netscape 的快捷方式。

```
"Netscape"           "netscape.xpm"           "netscape"
```

4.6.5 LessTif: mwm

LessTif 是一个 OSF/Motif 克隆，它被设计来运行任何 Motif 程序。它可以从 LessTif 网址上免费得到，现在在 www.lesstif.org/lesstif.html。你可以从 Caldera 网址 [Caldera Linux 黄页](#) 上连过去。你也可以从 Redhat ftp 网址的 `Redhat contrib` 目录下下载 RPM 版本：`lesstif-0.87-2.i386.rpm`。你用命令 `mwm` 激活 LessTif。现在，LessTif 不能运行任何 Motif 程序。项目仍然在建设中，但它提供了看起来感觉象 Motif 那样的窗口管理器。

4.6.6 AfterStep

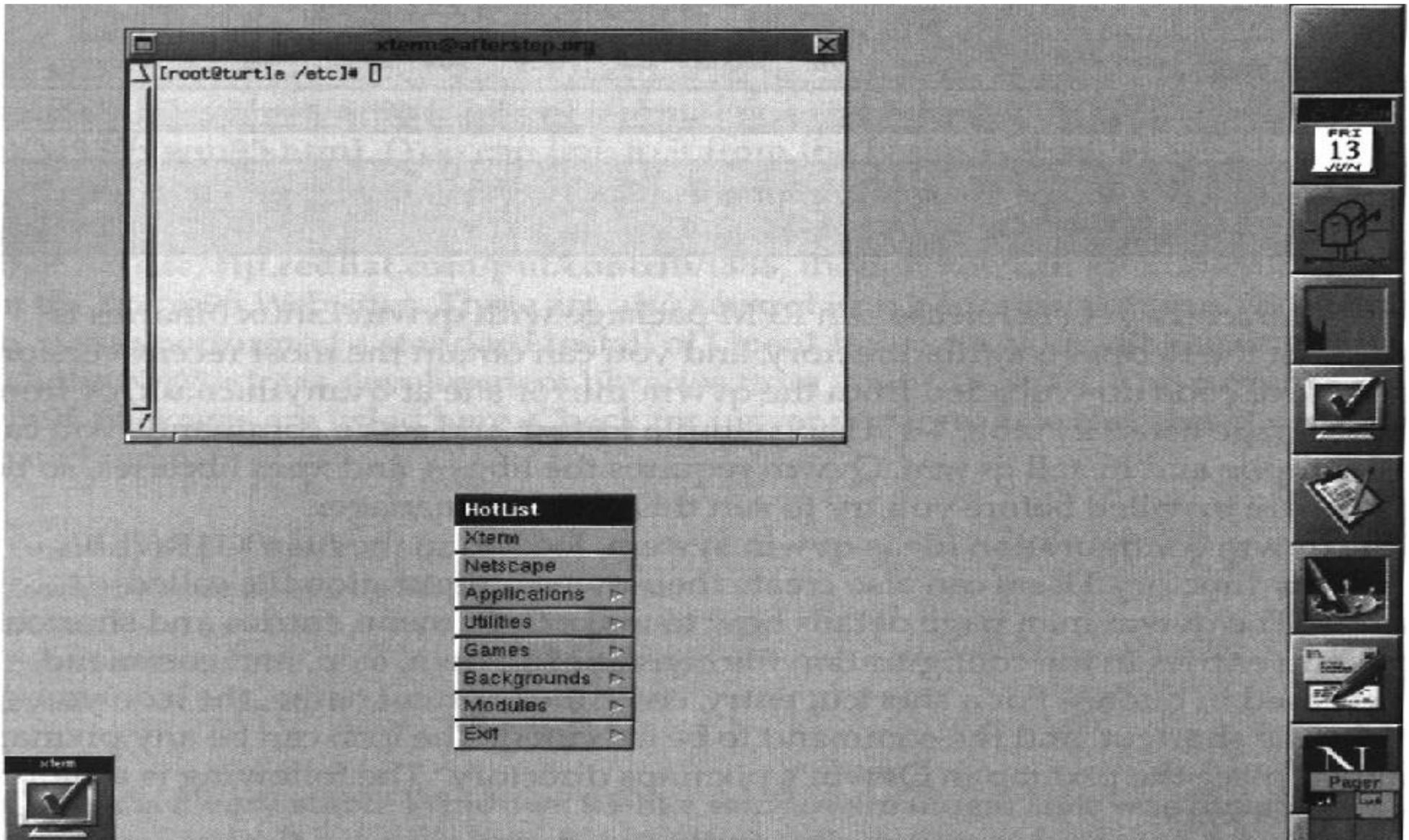


图 4-7 The AfterStep window manager

AfterStep 是作为在 NeXT 操作系统上使用的 NeXTSTEP 界面的克隆兴起

的，但是从那之后增加了自己的功能，如图 4-7 所示。源代码是基于 fvwm 窗口管理器的。你可以从 AfterStep 网址(afterstep.edoc.com)下载 AfterStep，或者从 Redhat contrib 目录下得到 RPM 版本，AfterStep-1.0-3.i386.rpm。AfterStep 有一个很有特色的应用程序装载器，叫做 Wharf，它有其他应用程序的包。还有弹出式菜单和 NeXTSTEP 风格的图标。AfterStep 网址有很多屏幕快照，说明界面怎样不同的设置。还提供有例子 .steprc 配置文件。

4.6.7 Motif

Motif 是私有的软件：你必须向开发者支付大概 150 美元来购买它。

Motif 和 Xview 是两个主要的相互竞争的 Unix 窗口界面，代表着两种不同的窗口标准。这两种标准近年来已经综合成一个新的 Unix 的 GUI 标准，称作公共桌面环境(Common Desktop Environment CDE)。现在已经有了 for Linux 的带 CDE 接口的商用桌面。

4.7 Crisplite 和 XEmacs 编辑器

OpenLinux Lite 提供给了你两种易用而功能全面的文本编辑器，它们是 Crisplites 和 XEmacs。这些编辑器合并了 Vi 和 Emacs 编辑器中体现的许多特征(见 17 和 18 章)。同时，你能享受到基于窗口的编辑器带来的使用方便。你

可以使用菜单和工具条选择命令，使用滚动条卷过文本，重新设置窗口大小，使用鼠标来选择和移动文本，或删除它(见图 4-8)。

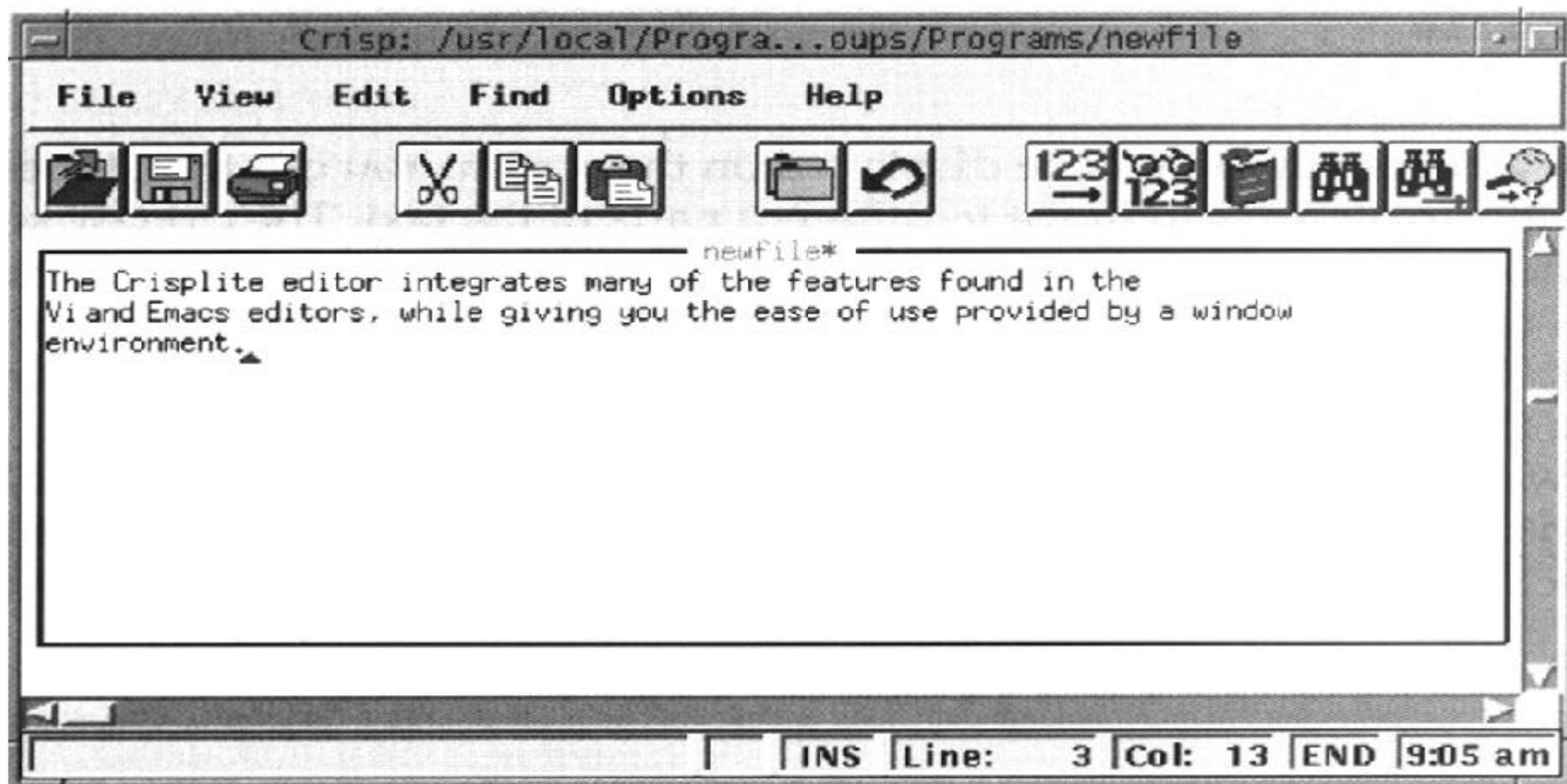


图 4-8Crisplite 编辑器

通过主窗口的 Help 菜单，你可以打开在线帮助，它提供了对 Crisplite 特征的详细解释。

Crisplite 编辑器已经安装在你的 OpenLinux 系统上了。在 Caldera 桌面，你

将看到一个 Crisplite 编辑器的图标。只要双击它就可以运行它了。现在，Crisplite 不在 fvwm 工作区域菜单里。一条简单的增加它的方法是从 /usr/bin/mcr 程序创建一个到 /usr/X11R6/bin/mcr 的连接。它可以用标题 mcr 显示其它的应用程序。要运行 XEmacs，你可以单击 fvwm GoodStuff 任务条上的它的按钮，或者在 Xterm 窗口里输入 xemacs 命令。

Crisplite 和 XEmacs 编辑器打开了一个有菜单和工具条的窗口。在窗口中，你可以输入文字。使用鼠标，你可以拖动来选择文字。工具条按钮提供了标准编辑功能的简捷方式，例如打开和存储文件，拷贝和粘贴文本。Crisplite 设计来让你同时在许多个不同的文件上工作的。你可以打开几个窗口，每个对一个单独的文件操作。使用鼠标，你可以从一个点中到另一个。你还可以打开缓冲区来编辑文本，以后可以将之存进文件。参看 18 章关于更详细的 XEmacs 的讨论。

你还可以用一个简单的基于屏幕的 Crisplite 版本，它使用你的命令行界面。你不用启动 X-Windows 和桌面。这种命令行版本的 Crisplite 只使用方向键来移动光标和控制，用功能键来发命令。但是它确实提供了一种全屏的视野，允许你在文本上逐屏的工作。你用 cr 命令启动 Crisplite 的命令行版本，通常后面跟着你要编辑的文件的名字。

```
$ cr myfile
```

你要编辑的文件将在屏幕上显示。你可以输入新的文字或者用方向键移到文本的别的位置。CTRL-W 键将你的文件存盘，CTRL-E 命令允许你打开另一个文件或一个新的。CTRL-X 键退出编辑器。参考 Crisplite 手册以得到编辑命令的列表。想使用 Crisplite 时用 Vi 命令的可以用 -mvi 选项激活编辑器，cr -mvi。如果你想在 fvwm 工具条上用一个按钮访问 Crisplite，你得在 /usr/lib/X11/fvwm

目录下的 `system.fvwmrc.goodstuff` 文件中为它做一项。前一节讨论 fvwm 工具栏的时候演示了这样的一项。

4.8 总结：X-Windows

使用 Linux X-Windows 窗口管理器和桌面，你可以和你的系统通过菜单，窗口，和图标交互，以及建立高效和简单的对 Linux 程序和 Internet 工具的访问。要开始 X-Windows，在命令行输入 `startx` 命令。X-Windows 支持很多种窗口管理器和桌面。有几种已经列在表 4-1 中了，连同你可以下载它们的网址。fvwm 窗口管理器支持虚拟和多桌面。它还有自己的工作区域菜单和任务栏，你可以从中开始 X-Windows 程序。一个虚拟桌面把桌面延伸以包含许多个屏幕视图。fvwm 虚拟桌面支持 4 个屏幕视图而 fvwm95 支持 6 个。有些项，如任务栏认为是粘性的，将在所有的虚拟桌面的视图上显示。

桌面程序，如 Looking Glass 集成了文件和程序管理的功能。通过目录窗口，你可以使用图标和菜单来管理文件和目录。目录窗口的菜单有你可以用来拷贝、移动、打印、删除或对文件采取的其它操作的命令。

你可以从 Redhat contrib 目录，[ftp.redhat.com/pub/contrib/i386](ftp://redhat.com/pub/contrib/i386) 下以 RPM 包形式下载大多数的窗口和文件管理器。当然，很难说清某一特定的软件包是什么。Linux 资源网络黄页会组织这些软件，分类并描述看，例如，www.chariott.com/linapps.html 下的 Linux 应用程序和工具页面。

表 4-1 窗口和文件管理器 (对大多数软件包,用位于 Redhat Contrib 目录下的 RPN1 安装,ftp://ftp.redhat.com/pub/contrib/i386)

窗口管理器	命令	描述	网址
TWM	Twm	Tom 的窗口管理器	
FVWM	Fvwm	免费的虚拟窗口管理器 (v.1.24)	www.hpc.uh.edu/fvwm
FVWM2	fvwm2f	vwm 2.0 版本 (现在是 Beta 版)	www.hpc.uh.edu/fvwm
FVWM95	fvwm95	在 fvwm2 上建立的 Windows95 界面	www.terraware.net/ftp/pub/Mirrors/FVWM95/fvwm95.html
Q 虚拟窗口管理器	Qvwm	使用直接编程写的 Windows95 界面	www-masuda.is.s.u-tokyo.ac.jp/~kourai/qvwm/qvwm-e.html
Xview	Olwm	Xview 窗口管理器 (OpenLook)	sunsite.unc.edu/pub/
	olvwm	olvwm 支持一个虚拟桌面	Linux/libs/X/xview

续表

AfterStep	Afterstep	基于 NeXTSTEP 的界面	afterstep.edoc.com
LessTif	mwm	Motif 的一个克隆	www.lesstif.org
Motif	Mwm	Motif 窗口管理器和桌面(商业产品)	
类似 Macintosh	mlvwm	Macintosh 界面的一个克隆, /linux 的虚拟窗口带有菜单的窗口管理器	www.iaseste.or.at
amiwn	amiwn	Amiga 窗口管理器界面	www.lysator.liu.se/~marcus/amiwn
通用窗口管理器	gwm	GNU 的窗口管理器	ftp.x.org/contrib/windowmanagers/gwm
WindowMaker	Wmaker	NeXSTEP 的一个克隆	www.myth.co.uk/~ximenes
Enlightment		一个原始的窗口管理器	www.rasterman.com

续表

K 桌面	Kwm	K 桌面环境窗口， 和 —	www.kde.org
	kde	些工具文件管理 器，	
Freedom Lite		Freedom 界面的 窗口管理器	fsw.com
文件管理器			
Caldera 桌面 (Looking Glass)	lg	Caldera 桌棉的商 业产品	www.caldera.c om
Xfm	xfm	文件管理器	col/contrib/RPM S(打 开 Linux CD-ROM)
treebrowser	tb	用 Xview 工具 的文件管理器	
XV 文件管理器	xvfilemgr	用 Xview 工具 和图标的文件管理 器	www.info.polymt l.ca/ada1/coyote /www

续表

TKDesk	tkdesk	一个基于 TK 的文件管理器	www.people.mainz.netsurf.de/~boliktkdesk
Midnight Commander		Norton Commander 的一个克隆	mc.blackdown.org/mc
		Commander	
Filerunner		文件管理器	www-c.informatik.uni-hanover.de/~kaiser/dfm
DFM	dfm	文件管理器	www.cd.chalmers.se/~hch/filerunner.html

第二部分 Linux 操作

第 5 章 Shell 操作

Shell 是一种命令解释器,它提供了一个面向行(line-Oriented)的用户和操作系统间的交互式界面,你在命令行中输入命令,它们被成指令送往操作系统。Shell 这一能力提供了许多高级的特征。例如,Shell 有一套特殊的字符集可以产生文件名,它可能重定向输入和输出,它也可以在后台运行操作,使你用来完成其它任务。

人们为 Linux 开发了三种不同的 Shell 它们分别是:the Bourne Again shell(BASH),公众域 Korn Shell (PDKSH)和 TCSH Shell。你可以利用所有这三种 Shell,尽管 BASH 是使用较少的,本书中都使用了 BASH Shell。你只需要一种 SHELL 即可完成你的工作。这一章讨论了所有的 Shell 的共同特征,而 15 和 16 章讨论了 BASH,PDKSH 和 TCSH 的不同之处,你可以看到如何设置 SHELL 来使用别名(alias)历史事件和系统变量。这些 SHELL 甚至有自己编程语言。

这一章主要讲述命令执行的共同特征；尤其是,命令行编辑,特殊字符、重定向、管道、变量、scripts 和作业控制。

5.1 命令行

当你登录到 linux 时,你可以看到一个命令界面。它是一条线,在这里你可以输入带着操作和参数的任何命令,一个 Shell 提示符.如下所示,标识了命令行的开始。

```
$
```

Linux 安装了 Bourne Again Shell 通常叫作 BASH Shell。 BASH shell 有一个美元提示符 ;但 Linux 有其它几种 Shell,每一种 Shell 都有自己的提示符。不同的几种 Shell 将在第 15 章讨论。

当系统提示符出现时,你已登寻到系统中,提示符标识了命令行的开始,你现在可以准备输入命令和它的参数了。

在下一个例子中,用户输入了 date 命令,它可以显示日期。用户在第一行输入了命令,然后按 Enter 执行命令。

```
$ date
```

```
Sun July 7 10:30:21 PST 1996
```

当你登录时,你实际进入了 Shell,它将你输入的命令加以翻译并传递给系统,Shell 遵循一定的语法对命令进行翻译,命令行中输入的第一个字必行是一个命令的名字,第二个字是命令的选项和参数,命令行中的每个字必须由空格或 tab

来 隔 开

```
$ Command Options Arguments.
```

当你输入一个 Linux 命令时,Shell 首先读入这一命令的名字,然后核实是否确实存在此名字的命令。如果没有这一命令,Shell 会显示一条出错的信息。

5.1.1 选项和参数

选项是一个只有一个字母的代码,它前面有一横线,选项可用于改变命令执行的动作的类型。有选项的命令的例子是 ls 命令。没有选项的 ls 命令列出了你当前目录中所有文件的列表。它只列出各个文件的名称。而不显示其它更多的信息。

加入 -l 项,ls 命令将改变它的任务,它会为每个文件列出一行信息,诸如数据大小和数据最后被修改的时间。在下一个例子中,用户使用了带 -L 选项的 ls 命令,-l 项之前的横线是必要的,Linux 用它来区别选项和参数。

```
$ ls -l
```

另一个选项 -a,可以列出目录中所有的文件,包括一些被隐藏的文件,隐藏文件通常是配置文件,文件名以点号开头。正因为这个原因,这些原因这些文件被称作点文件(dotfiles)。在大部分情况下,你可以使用组合选项。你应在选项之前加一起始横线。然后列出你想要的选基本项。例如:选项 -al,会列出你目录中所有文件的信息,包括任何的隐藏文件。

```
$ ls -al
```

ls 命令的另一选项是 -F。利用这一选项,ls 命令可以用于前面带斜杠的形式

显示目录名,这样你便可以轻松地加以识别。

大多数命令都被设计为可以接纳参数。参数是你在命令行中的选项之后键入的一个词。许多文件管理命令把文件名作为参数。例如,如果你只想显示某一特定文件的信息,你可以把此文件的名称加在-L选项之后。

```
$ ls -l mydata
```

你可以输入参数,也可以不用输入,取决于不同命令。一些命令,例如ls,不需要任何参数。另一些命令可能需要一些最小数目的参数。例如,cp命令至少需要两个参数,如果参数的数目与命令所要求的不符,Shell将给出出错信息。

记住在命令行中选项先于参数输入。以cp命令为例,用于检查覆盖状况(overwrite condition)的选项-i先于文件名参数键入。以下是一些输入命令的例子:

\$ ls	无选项命令
\$ ls -F	带选项命令
\$ cp -i mydata newdata	带选项和参数的命令

5.1.2 命令行特征

命令行实际上是你可以在编辑的一个文本缓冲区。在按回车之前,你可以在输入时的文本上进行编辑。编辑的功能是有限的,但它的确提供了一个改正错误的方法。BACK SPACE和DEL键允许你删除刚刚敲入的字符。利用这种字符删除的功能,你可以整行删除,并让你所输入的东西,只要你愿意。CTRL-D组合键可以整行删除,并让你从提示符重新开始,下一个例子中,用户键入了datl而不是

date。利用 BACKSPACE,用户删除了 1,然后输入了 e。

```
$ datl  
$ dat  
$ date
```

你将在其中进行工作的 Shell 是 BASH Shell,你缺省的 Shell,这一 Shell 具有特别的命令行编辑能力,这一功能在你学习 Linux 时会非常有用。你可以在执行之前对命令任意修改,也可以在命令行上任意移动并插入或删除字符。这时复杂命令尤其必要。你可以用 CTRL-D 或 DEL 删除光标所在位置的字符。如要加入文本,只需将光标移动到需要加入字符的地方,然后键入字符即可。你在任何时候都可以按 ENTER 键来执行命令。例如,如果在输入命令时,你犯了一个拼写错误,无须重新输入整个命令,你只要利用编辑操作即可发改正这一错误。

你也可以利用上箭头来重新显示你刚执行的命令。然后你可以重新执行这一命令,也可以对之修改,执行修改后的命令。当你进行一些重复性的工作,例如编辑同一个文件时,你会发现这一个功能是非常有用的。当你执行了一个你键入不正确的命令时这一功能也非常有用的。在这种情况下,会显示一条了出错信息和一个空的命令行,按上箭头后便可重新显示刚才执行的命令,将之改正并重新执行。这样,你就无需再重新敲入整个命令了。BASH shell 保存着你以前敲过的命令的列表,这一列表被称作历史列表(history list)。按动上箭头,你便可以在命令行上逐次显示各条命令。下箭头使你在列表中向下移动。当你将以前的这些命令显示在命令行上时,你就可以修改并执行这些命令。这一特征将在第 15 章进行详细的讨论。

一些命令可能很复杂,需要比较长的时间来执行。当你执行了错误的命令时,

你可以中断并中止这些命令的执行,只要利用中断键-CTRL_C 或 DEL。

你可以在一个命令行中置入多个命令,或者你也可以用几行来输入一个命令。如果要在一行中输入多个命令,用分号将命令隔开。下一个例子说明了如何在一个命令行中输入 ls 和 cp 命令。

```
$ ls-F ; cp -T mydata newdata
```

在按回车键之前加入一个反斜杠,你便可以在几个命令行中输入一个命令。反斜杠可以有效地将一个命令行持续到另一行。在下一个例子中,cp 命令是在三行中输入的。开始的两行以反斜杠告终,有效地使三行保持一个命令行。

```
$ cp -i \  
my data \  
newdata
```

5.2 特殊字符和文件名参数:* ,?,[]

文件名是参数命令中最常见的参数。你通常只知道文件名的一部分,或者你想参看几个有相同扩展名的不同文件,或者以相同字符开头的文件。Shell 提供了一套特殊字符来查找,匹配并产生文件名的列表。这些特殊的字符是星号,问号和中括号(*,?,[])。给出一个部分文件名,Shell 用找到的匹配操作来查找文件并产生文件的一个列表。Shell 用找到的匹配文件列不来代替部分文件名变量。这些文件名更可以成为那些操作文件的命令,如 ls。表 5-1(在本章最后)列出了 shell 的特殊字符。

你可以利用星号,*来查找以特定字符集开头或结尾的文件。你可将星号置于字符集的开头或结尾以形成进行文件名查找的格式。如果星号置于字符集前,shell 会查找以字符集结尾的文件。如果星号置后,则会查找以字符集开头的文件。任一匹配的文件名都会被拷贝入此操作产生的一个文件列表中。在下一个例子中,所有以"doc"开头的文件都会被查找并产生一个列表。然后所有以"day"结尾的文件也会被查找并产生一个列表。

```
$ ls
doc1 doc2 document docs mydoc monday tuesday
$ ls *day
monday tuesday
$
```

文件名常常包括一个扩展名,它由句号标识,同一个单字符构成。扩展名没有特殊的地位,它只是构成文件名的字符的一部分。使用星号使选择带扩展的文件变得很容易。在下一个例子中,星号用来显示那些具有.C 扩展的文件。.C 前的星号构成了 ls 的参数。

```
$ ls *.C
calc.c main.c
```

你也可以使用特殊字符,*与 rm 命令一起同时删除几个文件。星号首先选择具有特定扩展的一个文件列表,或者以给定字符集开头或结尾的文件列表,然后将此文件列表提交 rm 命令来进行删除。在下一例子中,rm 命令删除了所有以"doc"开头的文件。

```
$ rm doc*
```

星号本身会匹配所有的文件。如果你只用一个星号作为 `rm` 命令的参数,那么所有的文件都会删除。下一例子中,`ls*`命令列出了所有文件,`rm*`命令则删除了所有文件。

```
$ ls *
doc1 doc2 document docs mydoc , myletter your letter
$ rm *
$ ls
$
```

你应小心使用星号,尤其使用 `rm` 命令时,这一组合可能是非常危险的。在没有一选项的 `rm` 命令中错误放置的星号可能会删除的你所有文件。下一例子中第一个命令只删除了那些具有 `.C` 扩展的文件。而第二命令删除了所有的文件。注意第二个命令中星号与句号之间的空格。在命令行中空格的的作用是一个定界符,用于分隔参数。星号被认为是一个参数与 `rm` 命令一起使用时,会指示 `rm` 删除所有的文件。

```
$ rm *.C
$ rm * .C
```

问号 `?` 只匹配文件名中一个单一的不完整字符。假设你想匹配 `doc1` 和 `docA`,而没有 `document`。星号会匹配所有长度的文件名,而问号会将匹配限制于一个字符。下一例子匹配了以“`doc`”开头,后面只有一个不同字符的文件。

```
$ ls
doc1 docA document
$ ls doc?
```

```
doc1 docA
```

如果你愿意,你可以使用多个问号,你也可以把问号放在文件搜索样式的任一地方,下一例子寻找一个可能有三个不同字符的文件样式。

```
$ ls ?y?oc?
```

```
mydocs mylock sydoc1
```

你可以将?和其它的特殊字符组合来创建一些非常强大的匹配操作,假设你想查找只有一个字符扩展的所有文件。你可以使用星号来做文件名的正确匹配,使用?来做单字符扩展的匹配,:*、?。下一例子中,用户显示了所有只有一个扩展字符的所有文件。

```
$ ls *.*?
```

```
clac.c lib.a
```

星号和问号标识的是文件名中不完整的部分,而中括号[],允许你查找一套有效的字符。放在中括号中的任何字符都要与文件名相匹配。假设你想列出以"doc"开头,而以"L"或"A"结尾的所有文件,你对以"l"、"B"其它字符结尾的文件不感兴趣。下面即是如何来完成这一要求的方法:

```
$ ls
```

```
doc1 doc2 doc3 docA docB docD document
```

```
$ ls doc [lA]
```

```
doc1 docA
```

你也可以列出一个字符集的范围,而不是将之逐一列出。在上限和下限字符之间的横线将选择此范围内的所有字符。这一范围通常由所使用的字符集来决定。在ASCII码字符集中,a-g这一范围将选择包括a,g在内的a-g间的所有字

符。下一例子中,以"doc"开头并以"1-3"之间的字符结束的文件将被选取。然后以 B-E 间字符结束的将匹配。

```
$ ls doc[1-3]
doc1 doc2 doc3
$ ls doc[B-E]
docB docD
```

你可将中括号与其它特殊字符组合在一起产生非常灵活的匹配操作符。假设你只想列出以 .C 或 O 作扩展的文件。

```
$ ls *.[co]
main.c main.o calc.c
```

可能有时候特殊字符是文件名的一部分。在这种情况下,你需要在前面加一反斜杠来引用这一特殊字符来进行文件查找。在下一例子中,用户要查找一人以?结尾的文件,answer?。然而?是一人特殊字符它会匹配所有以"answer"开头并另有一人或多个字符的文件。在这种情况下,用户在"? "号前加一反斜杠来 quote 该字符以查找文件。

```
$ ls answers\?
answers?
```

你可以将一个被引用的字符与特殊字符在文件名中相组合。在下一个例子中,用户列出所有以"answer?"开头并有扩展的文件。

```
$ ls answers\?.*
answers?.quiz answers?.mid answers?.final
```

5.3 标准输入/输入和重定向

当设计 UNIX 系统时,文件的逻辑结构和物理结构是有区别的。物理上,文件以随机的块存放。逻辑上,文件是持续的 byte 流。作为 UNIX 的一个版本,Linux 也具有相同的组织形式。除了一些特别的系统调用,用户无须参照,文件的物理结构。对用户而言,所有文件具有单一的形式-byte 流。任何一个文件都可以轻易地拷贝或添加入另一个文件,因为所有文件都具有相同的组织形式。在这种意义上讲,Linux 中只有一种文件的标准,即 byte 流文件。Linux 对字符文件或记录文件,文本文件或二进制文件不作区别。

这一逻辑文件组织形式延伸到输入和输出操作。输入和输出操作中的数据像文件一样进行组织。键盘输入的数据被置入为持续 byte 集的数据流中。命令或程序的输出也被置入数据流中,并被设为持续的 byte 集。这种输入数据流在 Linux 中被称为标准输入,而输出数据流被称作标准输出。

因为标准输入和标准输出与文件具有相同的组织形式,它们可以很容易地与文件相互作用。Linux 有一种重定向能力,利用它你可轻易地将数据移入和移出文件。你可以将输出重定向,使之不在屏幕上显示,而是存到一个文件中。你也可以将标准输入重定向,使之从键盘移至一个文件,那么输入就会对文件进行读取而不是对键盘。

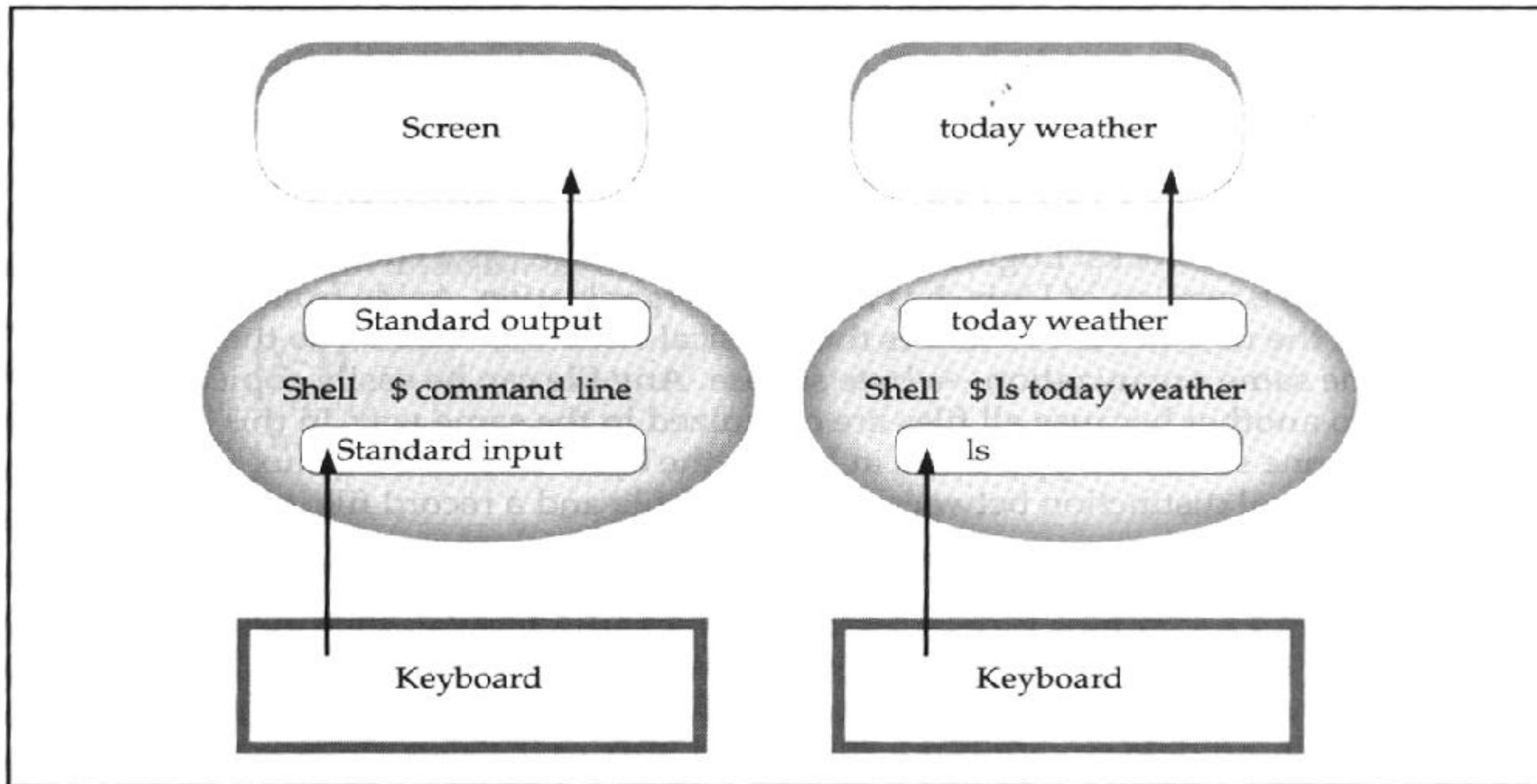


图 5-1 标准输出,标准输入和 shell

当 Linux 命令被执行并产生输出时,这一输出被置于标准输出数据流中。标准输出数据流的缺省目的地是一个设备,在这种情况下,是屏幕设备,诸如键盘和屏幕,被当作文件对待,它们的组织形式接收、发送 byte 流,正如同那些 byte 流文

件。屏幕是一种显示持续 byte 流的设备,缺省情况下,标准输出将把数据送到屏幕设备,屏幕将数据显示出来。例如,ls 命令产生所有文件名的一个列表,并将之输出到标准输出。在标准输出中的这一 byte 流被导向屏幕设备。然后文件名列表被显示在屏幕上。cat 命令也将输出结果送往标准输出。文件的内容被拷贝到标准输出中,而一标准输出缺省的目的地是屏幕。然后文件的内容便在屏幕上显示出来。图 5-1 说明了标准输入与键盘,标准输出和屏幕设备的相互关系。

5.3.1 重定向标准输出 :>和 >>

假如你不想在屏幕上显示一个文件列表,而想把列表存入一个文件。换言之,你想将标准输出指向一个文件而不是显示屏幕。要做到这一点,你应在命令行中 Linux 命令之后加入一个输出重定向符>(大于号)和一个文件的名称。表 5-2(在本章最后提到)列出了重定向符的不同使用方法。在下一个例子中,cat 命令的输出被由显示设备重定向到了一个文件。正如图 5-2 所示,mylette 文件的内容并未在屏幕上显示,它被重定向到文件 mewletter 当中去了。

```
$ cat myletter >newlette2
```

重定向操作创建了新的目的文件。如果这一文件已经存在,它将被标准输出中的数据所覆盖。你可以对 noclobber 特征进行设置,以防止在重定向操作中对已有文件进行覆盖。在这种情况下,对已有文件的重定向操作将会失败。你可以在重定向操作符后加一个惊叹号来强行忽略 noclobber 属性。下一例子设置了 BASH shell 的 noclobbe 属性,然后对 oldletter 文件强行写入,假如这一文件已经存在的话(图 5-2)。

```
$ set -o noclobber
$ cat myletter>! obllete
```

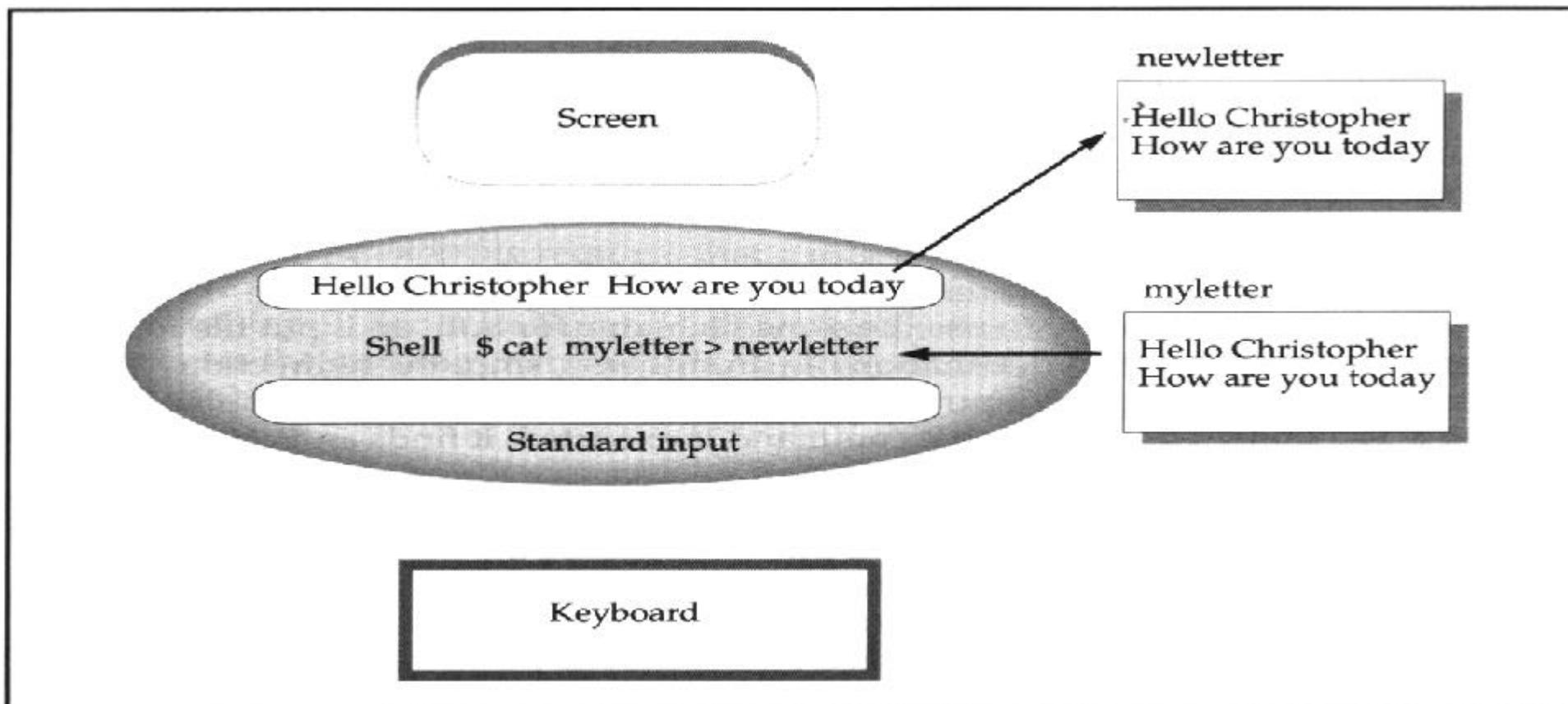


图 5-2 标准输出重定向到一个文件

重定向文件的创建

尽管重定向操作符和文件名都被放在命令之后,重定向并不是在命令之后执行。实际上,它是在命令执行之前执行,重定向操作创建了该文件并在它从标准输出接收数据之前设置重定向。如果这一文件已经存在,它将会被破坏并不被同名

的文件所代替。实际上,产生输出结果的这一命令只有在重定向文件创建之后才被执行。

在下一个例子中,ls 命令的输出被由显示设备重定向到一个文件中。首先,ls 命令作文件列表,在第二个命令中,ls 命令将文件列表重定向到 listf 文件中。然后 cat 命令显示了存在 listf 文件中的文件列表。注意 listf 中的文件列表包括 listf 的名字。由 ls 命令产生的文件列表将会包括由重定向操作创建的文件的文件名,在这一例子中,即 listf。listf 文件首先由重定向操作所创建,然后 ls 命令将与其它文件一起进行列表。然后 ls 的文件列表输出被重定向到 listf 文件中,而不是在显示设备上输出。

```
$ ls
mydate intro preface
$ ls> listf
$ cat listf
mydata intro listf preface
```

当你试图将同一个文件作为命令的输入文件和重定向的目的文件时,会有错误发生。在这种情况下,因为重定向操作是先执行的,因为输入文件是已存在的,它会存坏并被同名的文件所代替。当命令执行时,它就会发现输入文件是空的。

在下面介绍的 cat 命令中,文件 myletter 既是重定向输出的目的文件名,又是 cat 命令的输入文件名。正如下一个例子所示,重定向操作首先执行,将 myletter 文件破坏,然后用一个同名的 myletter 文件替换它。然后再 cat 命令并试图读入 myletter 文件中所有的数据。然而,现在在 myletter 文件中已经什么都没有的。

```
$ cat mylettaer>myletter
```

添加标准输出:>>

你也可以利用重定向操作符将标准输出添加到一个现存的文件中去。标准输出中的数据被添加到文件的末尾,而并不会覆盖这一文件。在下一个例子中,文件 myletter 和 oldletter 天加到 alletters 中去。alletters 文件将会包含 mylteler 和 oldletter 文件的内容。

```
$ cat myletter >>alletters
```

```
$ cat oldletter >>alletters
```

5.3.2 标准输入

许多 linux 命令可以标准输入中接收数据。标准输入本身从一个设备或文件中接收数据。标准输入的缺省设备是键盘。键盘上敲入的字符置于标准输入中,然后再被导向 Linux 命令。

cat 命令如果没有文件参数,它将会从标准输入中读入数据。当你在键盘上敲入数据时,每一个字符都会被置在标准输入中,然后被导向 cat 命令。cat 命令会将这些字符送往标准输出——显示设备——它会将字符在屏幕上显示出来。

当你做上述工作时,你会发现当你敲入一行,这一行会迅速在屏幕上显示出来。这都归功于在许多 Linux 系统中采用的行缓冲方法。行缓冲(Line buffering)要求用户键入整个行,然后输入才会被送往标准输入。cat 命令每次从标准输入获得一整行,然后它会将之迅速在屏幕上显示。在下一个例子中,用户执行了 cat 命令而未加任何参数。当用户输入一整行,它被送往标准输入,cat 命令将之从标准输入中读出,并送到标准输出。

```
$ cat
This is a new line.
This is a new line
for the cat
for the cat
command
command
^D
$
```

cat 命令将会持续执行下去,直到一个 CTRL-D 字符在一行中输入。CTRL-D 是任何 Linux 文件的结束字符。在某种意义上,用户是在键盘上创建一个文件,并用文件结束字符来结束文件。记住,标准输入,同标准输出一样,有着与 Linux 文件相同的格式。如果你将 cat 命令与重定向结合起来,你就有一种简易的为存储你敲入一个文件的数据。正如下一个例子所示,cat 操作的输出被重定向到 mydat 文件中。mydat 文件中将存在所有从键盘敲入的数据。cat 命令在这种情况下仍然没有文件参数。它将从标准输入-键盘设备——中接收数据。重定向操作符将 cat 命令的输出重定向到 mydat 文件,cat 命令与任何文件之间并没有直接的关系。它只是从标准输入中接收数据,然后将输出送往标准输出。

```
$ cat >mydat
This is a new line
for the cat
command
```

```
^D
$
```

5.3.3 标准输入的重定向:<

如同标准输出一样,你也可以对标准输入重定向。标准输入可从一个文件获得而不是键盘。对标准输入重定向的操作符是小于号<。在下一个例子中,标准输入被重定向为从文件 myletter 而不是从键盘中获得输入。myletter 文件中的内容被重定向操作读入标准输入。然后 cat 命令读入标准输入并在屏幕上显示 myletter 的内容。

```
$ cat < myletter
hello Christopher
How are you today
$
```

你可将对标准输入和标准输出的重定向结合起来,在一个例子中,cat 命令没有文件名参数。没有文件名参数的 cat 命令将会从标准输入中接收输入并将输出送往标准输出。然而,标准输入被重定向为从一个文件中接受数据,而标准输出重定向为将数据置于一个文件中。

```
$ cat <mylttev >neolettev
```

5.4 管道：|

你有时会发现处于这样一种情形,你需要将数据由一个命令传至另一个命令。换句话说,你想将一个命令的标准输出送向另一个命令,而不是一个目的文件。假设你想将一个文件列表送往打印机。你需要执行两个命令来完成这一工作:ls 命令来产生文件列表和 lpr 命令来将列表送往打印机。实际上,你需要获得 ls 命令的输出并将之作为 lpr 命令的输入。你可以想象为数据由一个命令流向另一个命令。要在 linux 中形成这样一种联系,你应使用称为管道的功能。管道操作符|(竖线条字符)放在两个命令之间二者之间形成了联系。一个命令的标准输出成了另一个命令的标准输入。管道操作接收了置于管道之前的命令的输出,并将数据作为置于管道之后命令的输入。如下例所示,你可以用一个管道符将 ls 命令和 lpr 命令连接起来,ls 命令产生的文件名列表被通过管道送往 lpr 命令。

```
$ ls | lpr
```

你可以将管道操作同诸如特殊字符等 shell 的其它特生结合起来,并完成一些特殊的操作。下一个例子中只打印了以 .c 为扩展的文件。ls 命令利用星号和 .c 来产生一个只具有 .c 扩展的文件列表,然后这一列表被通过管道送往 lpr 命令。

```
$ ls * .c | lpr
```

重定向只是简单地将输出放入一个文件,管道则将输出送往另一个 linux 命令。你也许想知道为什么这一功能不能由重定向来完成。人在头脑中保持一个文件和一个命令之间区别的概念:一个文件是一个包含数据的存贮媒体。你可以对之存贮或读。一个命令是执行指令的程序。一个命令可能会对文件进行数据

的读写,但命令本身绝不是文件。重定向可将数据由一个程序送往一个文件,但它不能将数据从一个程序送往另一个程序。只有文件才能成为重定向操作的目的地,程序则不能。

然而你可以通过一系列的重定向操作来模拟管道的过程。你可以将一个命令的输出送入一个文件。然后,在第二行,你可以将此文件作为要执行命令的重定向的输入。下一个例子用了两个不同的命令的重定向来打印一个文件名列表。同样的任务在上一个例子中已由管道操作完成。管道操作字面上获得一个命令的标准输出并将之作为另一个命令的标准输入。

```
$ ls * .c > tenpfile
```

```
$ lpr < tenpfile
```

到现在为止,我们一直利用文件名来作输入,但是管道是作用于一个命令的标准输出的,而不管这一输出是什么。注意到这一点是非常重要的。一个文件甚至几个文件的内容都可能通过管道由一人命令送往另一个。在下一个例子中,cat命令读入 mydata 文件的内容,并将内容输出,然后通过管道送给 lpr 命令。

```
$ cat mydata | lpr
```

假定你想将敲入的字符打印,而不是一个文件的数据。记住没有文件参数的 cat 命令将从标准输入中读入数据。在下一个例子中,cat 从键盘,而不是从文件中读入输入,并将输出通过管道送往 lpr 命令,cat 命令在 lpr 命令之前执行,因此你可以先在键盘上敲入你的数据,并以文件结束符 CTRL-D 结尾,一个管通的 byte 流可能来自于任何来源。

```
$ cat | lpr
```

```
This text will
```

```
be printed
```

```
^D
```

```
$
```

Linux 为 `cat` 提供了一个 `-n` 选项以在输出文件内容时,加入行的标量。你想打印有行标量的文件,然后将此输入送往 `lpr` 命令进行打印。例如:

```
$ cat -n mydata | lpr
```

为显示一个具有行标量的文件,你做的的大都是同样的事情,在这种情况下,有标号的输出通常会由管道送到 `more` 命令以进行一屏幕一屏幕的检查。你甚至可同时指定几个文件,把它们的输出由管道送到 `more` 命令,来检查所有这些文件。在下一个例子中,`mydata` 和 `preface` 文件都被标量并由管道送到 `more` 命令以进行屏幕检查。

```
$ cat -n mydata preface | more
```

Linux 有许多命令可产生修改过的输出;有 `-n` 选项的 `cat` 命令即是其中之一。另一个是 `sort` 命令。`sort` 命令读入文件内容,并且产生一个以首字母顺序来安排每行的版本。这一命令叫作过滤器。过滤器将在 14 章详细讨论它们通常同管道一起工作。在下一个例子中,产生了 `mylist` 的一个被 `sort` 处理过的版本并将之由管道-送往 `more` 命令以便进行屏幕的显示。注意原有的文件,`mylist` 并没有改变本身也没有被 `sort` 处理。只有 `sort` 在标准输出中的输出才会被分类。

```
$ sort mylist | more
```

你当然可以将几个命令组合起来,并将每对用管道符连接起来。一个命令的输出可由管道送往另一命令,然后,另一命令也可将它的输出送往第三个命令。假定你有一个清单列表的文件,你想把它用字母序和行号把它打印出来。要打印这

种字母序有行号的文件,你可以先用 `sort` 命令产生一个处理过的版本,然后用管道送给 `cat` 命令。有 `-n` 选项的 `cat` 命令把 `sort` 处理过的列表作为自己的输入,然后产生一人首字母序,有行号的列表输出,并将之由管道送往 `lpr` 命令进行打印。下一个例子中展示了这一命令,图 5-3 说明了 `mylist` 的一个首字母序,有行号的版本如何进行打印。

```
$ sort mylist | cat -n | lpr
```

你也可能用一种比较繁琐的办法来完成这一任务,利用重定向和若干单独的命令。在这种情况下,需要两个新的临时文件(`sfile` 和 `nfile`)来存贮每一操作的输出

```
$ sort mylist > sfile
```

```
$ cat -n < sfile >nfile
```

```
$ lpr < nfile
```

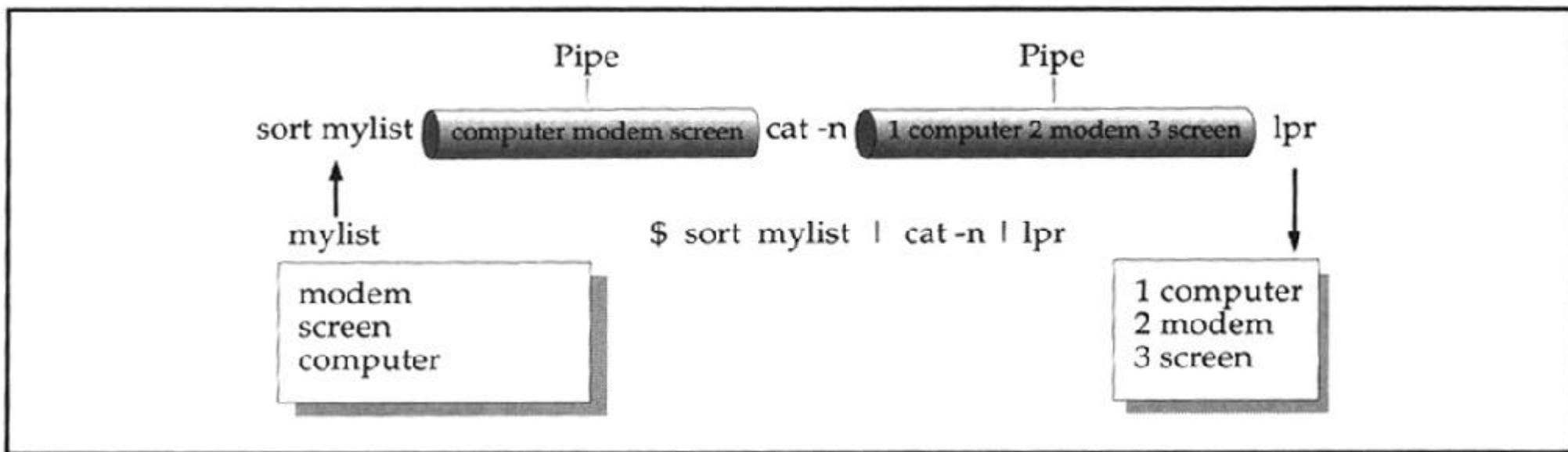


图 5-3 连接 x 多个命令的管道

利用标准输出参数 `-`, 可以细致地控制管道送往一个命令的标准输入。当你用横线作为一个命令的参数时, 它代表着标准输入。假定你想打印一个有顶部有其目录的文件形式, `pwd` 命令输出一个目录名, `cat` 命令输出一个文件的内容。在这种情况下, `cat` 命令需要将文件和由管道传送的 `pwd` 命令的标准输入同进作为自己的输入。 `cat` 命令将有两个参数: 由横线代表的标准输入和要打印的文件名。

下一个例子中, `pwd` 命令产生了目录名并由管道送往 `cat` 命令。对 `cat` 命令而言, 由管道送入的标准输入现在包含着目录名。如横线所代表的, 标准输入现在是 `cat` 命令的第一个参数。 `cat` 命令将目录名和文件的内容拷贝到标准输出, 并由管道送往 `lpr` 命令以进行打印。如果你想将目录名印在文件内容之后, 只要把横线作为后一个参数, 文件名作第一个参数即可, 正如 `cat mylist -`。

```
$ pwd | cat -mylist | lpr
```

5.4.1 管道和重定向:tee

假定你想将标准输入重定向到一个文件并且同时,在屏幕上显示输出的内容,这样你便可以确认你在存贮的东西。用 tee 命令你便可以做到这一点。

tee 命令将标准输出拷贝到一个文件,标准输出被拷贝到的新文件是 tee 命令的参数。标准输出继续前进,但当它经过 tee 命令时,它的内容被拷贝到一个文件。这就好像标准输出被分作两个拷贝,一个被重定向入一个文件,而另一个叫继续前进,通常是去往显示设备。下一个例子将文件 mylist 的内容拷贝到文件 newlist 中,并将内容显示在屏幕上。

```
$ cat mylist | tee newlist
```

当你正在更改输出,并且你想将输出存入一个文件,同时要观察到更改的内容时,tee 命令是非常方便的。在下一个例子中,文件 mylist 又被 sort 处理并且处理的内容由管道送至 tee 命令。tee 命令既将 sort 处理的输出存入一个叫作 sfile 的文件,又将其屏幕上显示了出来。

```
$ sort mylist | tee sfile
```

```
compwter
```

```
ncodem
```

```
screen
```

```
$
```

你也可以利用重定向将输出存一个文件,然后再它打印出来。

```
$ sort mylist >sfile
```

```
$ cat sfile
```

```
computer
screen
$
```

假如,你需要将你的输出存入一个文件并且把它打印出来。这种情况下,标准输出需要传送另一个命令。你需要利用 `tee` 命令来把输出拷贝到一个文件中,同时让标准输出经管道关入下一个命令。在下一个例子中,`sort` 命令的输入首先由管道送到 `tee` 命令,`tee` 命令将输出拷贝入文件 `sfile`。输出本身再由管道送到 `lpr` 命令以进行打印。

```
$ sort mylist | tee sfile | lpr
```

当你使用管道和重定向时要当心。标准输出的重定向确定了标准输出的目的地。标准输出会被存贮在那一目的文件中。一旦被存贮,就没有什么输出由管道送往下一个命令了。重定向不能在管道内部发生,尽管它可以被置于一系列管道操作的末尾。下一个例子是管道和重定向的可行的使用方法。`sort` 命令的输出经由管道送往带 `-n` 选项的 `cat` 命令来加入行标量,最后的结果被存贮在文件 `nfile` 中。

```
$ sort mylist | cat -n >nfile
```

如果你想要将文件存入 `nfile` 并把它打印出来应怎样?你不能做如下类似的操作:

```
sort mylist | cat -n > nfile | lpr ERROR
```

存贮文件并将之进行打印的唯一方法是使用 `tee` 命令,

```
$ sort mylist | cat -n | tee nfile | lpr
```

你可以在管道序列的任何地方使用 `tee` 命令。下一个例子中存贮了列表的

一个 sort 处理的版本并打印了一加行标号的另一个版本。

```
$ sort mylist | tee sfile | cat -n | lpr
```

5.4.2 重定向和管道处理标准错误 (standard Error): >&, 2>

当你执行命令时,有可能会有错误发生。你可能给出了数目不正确的参数,也有可能发生某种系统错误。当发生错误时,系统会产生一个错误信息。通常情况下,这一错误信息会同标准输出一起显示在屏幕上。然而,Linux 系统对标准输出和错误信息是有区别的。错误信息被放入另一个被称作“标准错误”的标准 byte 流中。在下一个例子中,cat 命令中的参数是一个不存在的文件名,myintro。这种情况下,cat 命令将会产生一个错误:

```
$ cat myintro  
cat :myintro not found.
```

```
$
```

因为错误信息是在除标准输出外的另一单独的数据流中,即使你已将标准输出重定向到一个文件,错误信息仍然会在屏幕上显示。在下一个例子中,cat 命令的标准输出被重定向到文件 mydate 中,然而,包含着错误信息的标准错误仍然被定向于屏幕。

```
$ cat myintro > mydata  
cat : myintro not found
```

```
$
```

如同标准输出一样,你也可以对标准错误重定向。这意味着你可以将错误信

息存入一个文件,以备将来参考。如果你需要一个错误记录的话,这一功能是非常有用的。如同标准输出一要,标准错误的缺少目的地也是显示设备,但你可以利用特殊的重定向操作符将标准错误定向到文件或你选择的任何设备。在这种情况下,错误信息将不会在屏幕上显示。对标准错误的重定向依赖于 shell 重定向的一个特殊的特征。你可以在重定向操作中利用数字访问所有的标准 byte 流。数字 0,1 和 2 分别代表着标准输入,标准输出和标准错误。缺省情况下,输出重定向 > 作用于标准输出。然而,利用在输出重操作符前加入 2,你也可将输出重定向列改为对标准错误进行操作。在下一个例子中,cat 命令又会产生一个错误。错误信息被重定向到以 2 为标量的标准 byte 流,标准错误中,

```
$ cat nodata 2> myerrors
$ cat merrors
cat : nodata not found
$
```

你也可以将错误信息添加入一个文件,只要利用数字 2 和重定向添加操作符 >> 即可。

在下一个例子中,用户将标准错误添加入文件 myerrors 中,这样 myerrors 便成为一个错误的记录。

```
$ cat nodata 2 >> myerrors
$ cat compls 2>> myerrors
$ cat myerrors
cat : nodata not found
cat : compls not found
```

\$

如果要同时对标准输出和标准错误重定向,你需要为二者加入各自的重定向操作符和文件。在下一个例子中,标准输出被重定向到文件 mydata,而标准错误被重定向到文件 myerrors 中。如果 nodata 确实存在,文件 mydata 将包含它内容的一个拷贝。

```
$ cat nodata 1 > mydata > myerrors
```

```
$ cat myerrors
```

```
cat: nodata not found
```

\$

然而,如果你想把错误信息,同重定向的标准输出存入同一个文件中,你需要将标准错误重定向到标准输出。在 BASH shell 中,你可以在数据流的数字前面加入一个 & 号来进行对标准数据流的访问,&1 指向标准输出。你可以在重定向操作中利用这种访问方法,将一个标准 byte 流作为一个目的文件。重定向操作 2>&1 将标准错误重定向到标准输出。

结果,标准输出成为了标准错误的一个目的文件。与之相反,重定向操作 1>&2 将标准输出重定向到标准错误。

在下一个例子中,cat 命令使用了一个不存在的文件 -nodata- 来作它的参数,结果产生的错误信息的重定向到文件 mydata。标准错误和标准输出的内容都将被存入同一个文件中 mydata。错误信息不再继续在屏幕上显示而是存入一个文件之中。如果想看错误,信息只须显示文件 mydata 的内容即可。如果 nodata 文件确实存在,mydata 将包含 nodata 文件的内容而不是错误信息。

```
$ cat nodata 1> mydata 2>&1
```

```
$ cat mydata
```

```
cat :no data not found
```

>&操作的缺省输出流是标准输出,缺省输入流是标准错误。如果这一操作前后没有数字,正如下一个例子所示,任何错误信息都会被重定向到标准输出,并同标准输出一同存入其目的文件。

```
$ cat no data > & mydata
```

5.5 shell 变量

你在 shell 当中定义一个变量,这一变量被称作-逻辑上-shell 变量。现在有许多不同的 shell。一些应用,例如 mailx 应用,有自己的 shell 程序和 shell 变量。你也可以利用所谓的 shell scripts 来创建自己的 shell。你有一个一旦登录之后便被激活的用户 shell。这通常被称作登录 shell。特殊的系统变量便是在这个登录 shell 中定义的。shell 变量在你的 shell 的激活状态中一直存在,意即直到你出 shell 为止。例如,注销即会退出现存的登录 shell。当你再一次登录时,你在登录(login) shell 所需的任何变量都必须重新进行定义。

5.5.1 变量的定义和 Evaluation: =,\$, set, unset

当你首次用变量名时,你便在 shell 中定义了一个变量。一个变量的名字可以

是字母表中任何字母的字符集,也包括下划线。变量名中也可以包含数字,但数字不能是名字的第一个字符。名字当中不能有其它类型的字符,例如冒号、&号,甚至不能有空格。这些符号由 shell 保留自用。并且,名字不能超过一个词。Shell 利用空格来区分命令行中命令的不同要素,诸如选项,参数和命令的名字。

你可用赋值操作符 = 来给给变量赋值。你应敲入变量名,赋值操作符,然后是要赋的值。不要在赋值操作符前后留任何空格。诸如 poet = Virgil 的赋值操作会失败。(c-shell 有一个稍微不同的赋值操作,在本章稍后的 c-shell 变量中将会谈及)你可以将任何字符集赋给一个变量。下一个例子中,变量 poet 被赋值为字符串 Vivgil

```
$ poet=Vivgil
```

一旦你为一个变量进行了赋值,你便可以利用变量名来引用所赋值。通常你可以把变量的值作为一个命令的参数。利用在变量名前加 \$ 操作符,你便可以引用变量值了。美元符是一种利用变量名来引用变量值的特殊字符。Evaluation 可获得变量的值,通常是一个字符集合。这个字符集将代替命令行中的变量名。只要变量名前加入了 \$ 符,这个变量名就会被变量值所代替。在下一个例子中,shell 变量 poec 被取值,它的内容,Virgil,被用作 echo 命令的参数。echo 命令只是简单地将字符集显示在屏幕上。

```
$ echo $ poet
```

```
Virgil
```

你必须仔细区分对变量取值和变量名本身。如果变量名之前你没有加入 \$ 符,那么你就只有变量名本身,在下一个例子中,变量名之前没有 \$ 操作符。在这种情况下,echo 命令将 "poet" 作为自己的参数,因此显示了 "poet"。

```
$ echo poet
poet
```

变量的内容常被用作命令的参数。一种通常的命令参数是一个目录路径名。要重复的键入相同的目录路径是很乏味的。如果你把一个目录路径名赋值给一个变量,你不对变量取值即可。当对变量用 \$ 操作符取值时,你便得到赋给这个变量的目录路径。下一个例子中将目录路径名赋给一个变量,并在 copy 命令中使用了变量的值。 ldir 的取值(即 \$ldir)是路径名 /home/chris/letters. copy 命令等价于 cp my/etter/home/chris/letters.

```
$ ldir =/home/chris/lettevs
$ cp myletter $ ldir
```

你可以利用 set 命令获得已定义变量的列表。下一个例子利用 set 命令显示了所有已定义变量和它们的值。

```
$ set
poet virgil
ldir/home/chris/letters/old
$
```

5.5.2 TCSH shell 变量

TCSH shell 中的变量与 BASH shell 只有一点极小的差别。在 TCSH shell 中,你要用 set 命令和赋值操作符对一个变量赋值。要给一个变量赋值,首先键入 set,然后是变量名,赋值操作符,和要赋的值,赋值操作符必须前后都有或都没有空

格.例如——`set poet=Virgil` 是一个错误,在下面,变量 `poet` 被赋值为 `Virgil`:

```
set poet=Vigil
```

同 `BASH` shell 类似,`TCSH` `SHELL` 利用美元符进行取值,取值操作获得变量的值,它通常是一套字符集。然后字符将代替命令中的变量名。在下一个例子中,shell 变是 `poet` 被取值,其值 `Virgil` 被作为 `echo` 命令的参数。

```
>echo $poet
```

```
Vigril
```

同 `BASH` shell 一样,双引号,单引号,和横线会取制特殊字符的取值.下一个例子中,双引号限制了特殊字符?。

```
>set hotice="Is the meeting romorrow?"
```

```
>echo $hotice
```

```
Is the meeting tomorrow?
```

```
>
```

5.6 shell 脚本:用户定义的命令

你可以把 shell 命令放入一个文件,然后让 shell 从文件中读并执行这些命令。在这种意义上,文件成为 shell 一个程序,它象程序中的语句一样执行 shell 的命令.包含着 shell 命令的一个文件被称作 shell 脚本。

你可以利用一个标准的文本编值器,如 `Vi`,来将 shell 命令键入一个 `script` 文件。`sh` 或。命令再加上文件名,将会读 `script` 文件并执行其中的命令,下一个例

子中,文本文件 lsc 中包含着 ls 命令,它将只列出带 .c 为后缀的文件。

```
lsc
ls *c
$ sh lsc
main.c calc.c
$ .lsc
maim.c calc.c
```

你可以利用对 script 文件的执行许可的设置来避免 sh 或 .命令的使用。当 script 文件最早由文本编辑器创建时,它只有读和写的许可。chmod 命令加入 +x 选项将给予 script 文件执行许可,(第 7 章中将讨论许可)一旦它是可执行的,在 shell 提示符后键入文件名并加回车,文件和其中的 shell 命令将被执行,在效果上,script 文件名变成了一个新命令名。这样,你便可以利用 script 文件来设计并创建自己的 linux 命令,你只须设置一次许可,在下一例子中,lsc 文件所有者的执行许可被设置为开。然后 lsc 这一 shell script 如同其它 linux 命令一样得到执行。

```
$ chwod u+x lsc
$ lsc
main.c calc.c
```

如同其它 linux 命令一样,shell script 也可以接受参数,命令行上的参数被依次自上来引用.利用 \$ 操作符和位置的标识可以来引用参数,第一个参数可由 \$1 引用,第二个参数 \$2,以此类推,在下一个例子中,lsext script 打印有特殊扩展的文件。第一个参数是扩展,该 script 以参数 C 执行,(当然,可执行许可已被设置过了)。

```
lsext
```

```
ls * $!
$ ls ext c
main.c calc.c
```

在下一个例子中,打印具有行标号文件的命令已被放入一个可执行文件 /lphum 中,/lphum 以一个文件各作为参数。打印行标号的命令在后台执行。

```
lphum
pr -t -n$1/lp&
$ lphum mnydata
```

你可能会同时利用不止一个变量,变量数目可能会变化。利用 lphum 你可能一次想打印三个文件,也可能是五个文件。\$操作符再加上星号,\$*,引用命令行上所有的参数。利用\$*你便可以接收数目可变的参数。在下一个例子中,lphum 使用了\$*,这样它便可以接受可变数目的参数了。

```
lphum
pr--t--n$*/lp&
$lphum mydata prefae
```

使用 shell script 是为命令创建别名的另外一种方法。在下一个例子中,比如:shell script 包含着一个有一定选项的 rm 命令。rm 在删除一个文件之前会征求用户的认可

```
rmi
rm-i $*
$rmi mydata doc1
```

5.7 作业:后台,杀除(kills)和中断

在 Linux 中,你不仅可以控制命令的输入、输出,也可以控制它的执行。你可以在执行命令的同时在后台运行一个作业。你也可以在命令执行完之前取消一个命令。你甚至可以中断一个命令,然后再从断的地方继续执行,后台操作对于大作业非常有利的,无须在终端前等待一个命令执行结束,你可以把它放入后台。这样你便可以执行其它的 Linux 命令。例如,你可以在其它一些文件打印时编辑另一个文件。

取消一个后台的命令常常会为你节省许多不必要的开支。假如说,你执行了一个命令来打印所有的文件,你又意识到有一些非常大的文件,你并不想打印,你可以引用打印命令的执行并将之取消,对命令的中断是不经常使用的,有时,它是无意中进行的,只要你想,你便可以中断一个编辑过程去发送 mail,然后再回到编辑会话中,从离开的地方开始。后台命令和取消及中断作业的命令见表 5-2。

在 Linux 中,命令被视为一个进程—要执行的任务,Linux 系统同时可以执行多个进程,正如同 Linux 系统可以同时处理几个用户一样。有一些命令可以用来检查和控制进程,而这些命令通常是系统监察操作而保留的。实际上进程不仅包括了用户执行的命令,还包括了系统执行的以维持 Linux 运行的许多任务。

用户执行的进程通常被称为作业,以此区别系统进程。当用户执行一个命令时,它便成为等待系统执行的一个作业,shell 提供了一套作业控制操作以使用户控制作业的执行,你可以把作业在后台,取消或中断一个作业。

5.7.1 后台和前台:& fg bg

在一个命令后面加入一个"&"符号,你就可以在后台执行这一命令。当你这么做的时候,屏幕会显示一个用户的作业号和系统的进程号。放在中括号中的用户数,是用户可以对它进行引用的号码。系统进程号是系统对这一个任务的标识号。在下一例子中,打印文件 mydata 的命令被置于后台执行。

```
$ lpr mydata &  
[1] 534  
$
```

你可以把不止一个命令放在后台执行。每一个命令都被标识为一个任务并且有一个名字和一个任务号。jobs 命令会列出所有在后台执行的任务。表中的每一行都包括一个中括号内的任务号,以及任务名,而不管该任务已终止或还在运行。"+"号表明该任务当前正在被处理,"-"号标明了下一个要被执行的任务。在下一个例子中,两个命令被放入后台执行。jobs 命令列出了这些任务,并显示哪个任务正被执行。

```
$ lpr intro &  
[1] 547  
$ cat *,C > mnyprogs &  
[2] 548  
$ jobs  
[1] + Running lpr intro  
[2] - Running cat * . c >nnyprogs
```

```
$
```

如果你愿意,可以在命令行上以"&"号为间隔敲入几个命令,然后将这些命令同时放入后台执行。在这种情况下,"&"号不仅在命令行上将命令进行了隔离,还将这些命令置于后台执行。在下一个例子中,第一个命令将所有具有.l扩展的文件 sort 处理并且重定向,这个命令被置于后台。在同一命令行上的第二命令,打印有.c扩展的文件,也被放到后台执行。注意这两个命令都是以"&"号结尾的。jobs 命令然后将 sort 和 lpr 命令作为单独的操作列出。

```
$ sort *.l >/ldocs & /lpr *.c &
[1] 534
[2] 567
$ jobs
[1]+  Running      sort *.l > ldocs
[2]-  Running      lpr
$
```

在你执行完 linux 中的任何命令时,如果你有运行于后台的作业,系统会通知你这些后台作业是否完成,系统不会中断任何操作,例如编辑,来通知你一个工作已经完成。如果你希望某个任务完成时立即通知你,而不管你正在系统中用何种操作,你可以利用 notify 命令来指示系统通知你,notify 命令的参数是作业号。当这一任务完成时,不管你在做什么,系统都会打断你并通知该任务已完成,下一例子中告诉系统当 2 号任务完成时即通知用户。

```
$ notify%2
```

你可以利用前台命令,fg,当一个任务自后台取出,如果后台只有一个任务,单

独的 fg 命令就可将其带到前台,如果后台有多个任务,使用此命令时,你必须加入任务量。你应将任务量置于 fg 命令之后,并在任务量前加一个百分量,下一个例子中,第二个任务被置入前台。你可能不会立即发现提示符,因为第二个命令正在前台执行,当这一个命令执行完时,提示符会重新出现,这时候你就可以执行另一个命令了。

```
$ fg %2
sat *.c nnyprogs
$
```

也有一个 bg 命令可以将任务置于后台,这一命令通常用于中断的任务。它将会”中断,CTRL-Z”中进行简短的讨论。

5.7.2 取消任务:Kill

如果你想停止一个正在后台运行的任务,你可以使用 Kill 命令使其强行结束。kill 作业号或系统进程号作为其参数。用户的作业号前必须加%量。利用 jobs 命令你可以查到作业号。在下一个例子中,jobs 命令列出了后台的任务,然后任务 2 被取消。

```
$ jobs
[1]+running /pr intro
[2]-running cat *.c > nnyprogs
$ Kill %2
$
```

你也可以利用系统进程号来取消一个任务。利用 ps 命令可得到系统进程号。ps 命令比 jobs 命令显示了更多信息。在系统管理中将会对之详细讨论,下一个例子列出了用户正在运行的进程。PID 是系统进程号,也被称作进程的 ID。TTY 是终端识别量。时间是进程被运行的时间长度。Command 是进程的名字。

```
$ ps
PID      TTY      TIME    COMMAND
523      tty24    0:05    SH
567      tty24    0:01    lpr
570      tty24    0:00    ps
```

然后你会可以在 Kill 命令中引用系统进程号。在此进程无须加 % 号。下一个例子取消了进程 567

```
$ Kill    567
```

5.7.3 中断: CTRL-Z

你可以用 CTRL-Z 命令中断并停止一个任务。这一命令将任务置于一边直到任务被重新启动。任务本身并没有结束,它只是被暂停了,等待你希望它继续进行的命令。当你准备好后,使用 fg 或 bg 命令你可以继续进行这一任务。fg 命令将在前台继续一个中断的任务。bg 命令则将这一任务放入后台执行。

有时你需要将一个正在前台运行的任务移至后台。但是,你不能下接将一个正在执行的例子中,列表并重定同 .C 文件的命令首先被用 CTRL-Z 中断。然后这一任务被置于后台。

```
$ cat *.c nnyprogs
^ z
$ bg
```

5.7.4 延迟执行:at

利用 at 命令,你可以在某一特定的时间执行命令。你可以指定一个你希望命令执行的时间,而不是立刻将一个任务放入后台。然后你可以退出,由系统的监视执行什么命令及何时执行。

at 命令以你希望命令执行的时间作为参数。时间是一个标识小时的数字以其后的关键词 a.m 或 p.m。你也可以加入一个日期。如果未指定日期,系统即认为当前的日期。然后 at 命令会从标准输入中输入 linux 命令。你可以在键盘敲入上述命令,并以 CTRL-D 来结束标准输入,你也可以把它敲入一个文件,并将文件由标准输入重定向到 at 命令。在下一个例子中,用户决定在 4:00 a.m 执行一个命令。

```
$ at 4am
lpr iwtr0
^ D
$
```

在在下一个例子中,用户将几个命令放入一个 latexcmds 文件中,然后将文件内容重定向为 at 命令的输入。at 命令将在 6:00p.w 执行这些命令。

```
latexcmds
```

```
lpr    intro
cat *.c > nnyprogs
$ at 6pm < letecnnds
```

你可以有许多方法来标定时间和日期。at 命令假定了一个 24 小时的时间序列,除非被关键字 a.m 或 p.m 修改。你可以在小时和分钟之间加入“:”号来标定小时中的某一分钟,例如:6:30。at 命令也能识别一些标定某些日期和时间的关键词。关键词 noon 标定的是 12p.m,你可以使用关键词 midnight 来代替 12a.m。在下一个例子中,用户执行了一个分钟标定的命令和一个关键词 noon 标定的命令

```
$ at 8:15pm < latecods
$ at noon < latechds
```

日期可以是一月中的某一天,也可以是一周中的某一天。一月中某日包含着天数于一个代表月份的关键词。月份可以三个字母由子缩略词表示。例如,一月(January)被写作 jan,天数跟在月份名之后。如果没有月份名,叫被认为是当前月。Feb14 指明二月十四号;21 本身指的是本月的 21 号。在下一个例子中,用户首先指定在本月 15 号执行命令,而后又指定在 10 月 29 日执行。

```
$ at 8:15pm 15 < lsteconds
$ at hoon Oct 29 < lsteconds
```

如果你只想在本星期内运行任务,那么,你只需指定星期几。而无须指定月和日。星期的每天都由其名字表示,这样,敲入 Tuesday 即意味着将在星期二运行你的任务。你也可以利用 today 或 tomorrow 这样的关键词来指定日期。在下一个例子中,用户在周五和明天运行任务。

```
$ at 8;15 am    Friday < latecods
```

```
$ at noon tomorrow <latechods
```

利用时间或日期,你也可以指定一个增量,例如,你可以指定一个任务在距今天一周后执行或周五后两个月执行。你可以使用+号跟着一个指定时间段长短的关键词来指定增量,时间段关键字有:minutes, hours, days, weeks, months, 或者 years 复数符 s 可去掉以表示单一的时间间隔。

例如,在 19 号后的一个月运行命令,你可以输入 19+ month。明天后的一星期是 tomorrow+ week。今天之后的两周是 today+2weeks。在下一个例子中,用户指定在周一后的 6 周和今后之后的 3 个月时执行命令。

```
$ at 8:15pm    monday + 6weeks <latecnols
```

```
$ at  noon    today  +3months <latecmds
```

at 有一些选项可以让你列出所有等待执行的 at 任务,也可以取消任 -at 任务。每次你执行 at 命令,你指定以后执行的 Linux 命令都被放入队列并作为一个 at 作业列出。at 命加 l 选项可以列出所有的 at 任务,每一个任务都有一个你可以引用的编号。

```
$ at -l
```

```
732893802 ,a Fri  sept 27 20:15:00,1996
```

```
732893803,a Fwe  sept 24 12:00:00  1996
```

用 -r 选项你就可以取消 at 任务,要取消一个特定的任务,你需要在 -r 选项后敲入任务编号,在下一个例子中,用户取消 at 任务 732893802,a

```
$ at -r 732893802  ,a
```

```
$ at -l
```

```
732893802, a Tue sept 24 12:00:00 1996
```

通常,at 命令不会通知你任务已经执行完毕,然而,利用 -m 选项,你可以要求当任务执行完后,用 mail 通知你,你可以指定接受 mail 的一个特定的任务编号。在下一个例子中,当任务 732893803,a 执行完毕后,将后有 mail 通知你

```
$ at -m 732893803,a
```

5.8 总结: shell 操作

shell 是用户和操作系统之间提供接口的一个命令解释器。你在命令行上输入命令,这此命令被 shell 解释,并作为指令送往操作系统,shell 有非常复杂的特征,诸如特殊字符、重定向、管道、scripts 和任务控制。

shell 有三个特殊字符,*,?, [], 它们可以使你产生一系列的文件名作为命令行上的参数,*号会匹配所有可能的字符序列,?是会匹配任意一个字符,[],匹配一个字符集。你甚至可以利用特殊字符来组成复杂的匹配。

在 linux 中,文件和设备,以及和命令的输入和输出,都有相同的结构— byte 流,命令的所有输入数据流中,因为标准输入和标准输出同文件有着相同的结构,它们可以很容易地与文件交互,利用重定向操作符,你可以将标准输入和标准输出指向一个文件。利用 "<"重定向符,你可以将标准输出由一个命令重定向到一个文件。利用 "<"重定向符,你可将标准输入重定向为从一个文件读入,你可以也利用重定向添加操作符 ">>"将标准输出添加到一个已存的文件中去。

既然命令的标准输入和标准输出有着相同的标准格式,你可以很容易地将一

个命令的输出作为另一个命令的输入,管道允许你接收一个命令的标准输出,并由管道将这作为另一个命令的标准输入,在同一个命令行上,你可以串连几个命令,每一个的输入都另一是个命令的输出。

使用编辑器,你可发创建一些包含 shell 命令和变量定义的文件。这些文件被称作 shell scripts,。shell scripts 甚至有参数变量,它们将从命令行上接收参数,通过设置 shell script 的执行许可,你可以把 shell script 的各字当作 shell 命令对待。

你在 shell 中也可以定义变量变对其赋值。在变量名前加\$,你可以就引用一个变量。你可以把变量作为命令参数使用。它们可以包含目录络经名或者要执行的命令。

当你执行一个命令时,它被 Linux 当作一个要执行的任务。你可以指示 Linux 在后台执行任务,以使你执行其它命令,将后台操作符放于命令行末尾会指示 Linux 在后台执行这一命令。用 jobs 命令可列出你在后台的任务。用 fg 命令,你可将一个任务由后台移至前台。你可以用 kill 命令来终止后台任务,或用 CTRL-C 命令来中断任务。

表 5-1 shell 符号

标准错误/重定向符号	执行
ENTER	执行一个命令行
;	在同一命令行中分隔命令
'命令'	执行一条命令
*	匹配文件名中的任何符集
?	匹配文件名中的单个字符
[]	匹配文件名中的可能的字符集
\	引用紧跟的字符,用于引用特殊字符
>	将标准输出重定向回到文件或设备。 如文件不存后,则创建新文件;如文件已存在,覆盖该文件假设文件存在,则强制进行覆盖,掩盖 noclobber 选项
<	将标准输入重定向为由文件或设备强程序
>>	将标准输出重定向到一个文件或设备,将输出添加到文件尾
	将一个命令的标准输出由管道作为另一命令的输入
&	在后台执行命令
!	历史命令
2>	将标准错误重定向到一个文件或设备

2>>	将标准错误重定向并添加一个文件或设备
2>&1	将标准错误重定向标准输出
1&	将标准错误由管道作为另一命令的输入
>&	把标准错误重定向到一个文件或设备

表 5-2 shell 操作
执行

命令	执行
ENTER	执行了一个命令行
;	在同一命令行上分隔命令
command \opts args	在回车之前键入后斜杠以复在另一命令行继续输入
命令	
'commerel'	执行一个命令
BACKSPACE	删除前一个字符
CTRL-H	
CTRL-U	删除命令行并重新开始
CTRL-C	中断并停止一个命令的执行
文件名 Gerererion	执行
的特殊字符	
*	匹配任何字符集
?	匹配任何单个字符

[]	匹配可能的字符集
\	引用跟的字符。用于引用特殊字符
重定向	执行
command.>filnemme	将标准输出重定向到一个文件或设备
commane <filnemme	将标准输入重定向到一个文件或设备
commane>>filerane	将标准输出添加到一个文件末尾
commane>!filerane	在 C-Shell 和 K-Shell 中,强制写入一个已存在的文件
commane2>filerane	B-Shell 中,将标准错误重定向到某文件
commane2>>filerane	B-Shell 中,将标准错误添加到文件末尾
commane2>&1filerane	B-Shell 中,将标准错误重定向到标准输出
commane>&filerane	C-Shell 中,将标准错误重定向到文件
管道	执行
管道 管道	把一个命令的输出当作另一个命令输入
管道 &管道	执行 TCSH-Shell 中,把标准错误作为另一个命令输入
后台任务	执行
&	后台执行任务
fg % jobnum	将后台的命令放入前台或继续被中断的程序
bg	将前台的命令放入后台
CTRL-Z	中断并停止以后正运行的程序,程序在将之继续之前保持信止和等待状态

notify % jobnum
kill % jobnum
kill processnum
jobs

将任务结束时通知用户
取消并结束在后台运行的任务

ps
at time date

列出所有后台任务,除非使用 jsh shell, 否则在 Bourne shell 中不存在
列出包括的后台任务在内的所有在运行的进程
在某一特定的时间和日期执行命令;time 可用分钟和小时输入,并由 am 和 pm 加以限定
hour: minutes am pm
日期可用一日或一周中某天确定
mpnth day
month 可由三字因缩略词表示
3 am, Feb etc
day 是由名字表示
monday Tuesday wednesday ,etc
可利用关键词来标识日期和时间
am, pm., how, noon, midnight, today,
tomorrow
你可利用+操作符由某一天或时间增加一个时间段。+操作符之后数字标明时间段的个数
date + num time - segment
time - segment 可以是

hours minutes days weeks months
years

关键是 keyword 从现在时间增加一个时间段

wext time -- segnert

next week

选项

l jobnum (列出现有的 at 任务)

r jobnum (取消一个任务)

m jobnum (任务结束时用 mail 进行通知)

第 6 章 Linux 文件结构

在 Linux 中,文件被组织在目录中,反过来,文件又在统一的文件结构下相互联系。文件引用不仅仅是依照它的名字,此外还依照它在文件结构中的位置。你想创建并向文件结构增加多少目录都可以。Linux 文件命令可以执行复杂的操作,诸如移动文件,带同子目录的整个目录的拷贝。使用文件操作,诸如 find, cp, mv 和 ln,你可以对文件定位,拷贝或将他们由一个目录连接到另一个目录。

这些特性一齐构成了 Linux 的文件系统。本章首先介绍不同的文件类型和类别(classes),然后,本章介绍了 Linux 文件总体结构,以及如何利用路径名和工作目录即可实现对目录和文件的引用。本章的最后部分讨论了诸如拷贝、移动、连接文件等不同文件操作。

6.1 Linux 文件

你可以用任何字母表中字母,下划线和数字来命名一个文件。你也可以使用句号和逗号。但是,文件名不能以数字开头。此外,除非在特殊的情况中。你都不应该以点号作为文件名开头。其它的字符,如横线、问号或逗号,它们都是系统保

留的特殊字符,不能用于文件名。文件名可长达 256 个字符。

你可以把扩展作为文件名的一部分。句号用于区别文件名和扩展。扩展对于将你的文件分类是非常有用的。你可能对某些大众已接纳的标准扩展比较熟悉。例如,C 的源代码文件总是有 C 的扩展。包含编辑过的目标代码文件有 O 扩展。当然,你可以加入自己的文件扩展。下一例子都是有效的 Linux 文件名。

```
preface
chapter 2
Calc.c
intro.bk 1
```

有一些特殊的初始化文件用于包含 shell 的设置命令。这些文件是隐藏的或者是点号文件,如第 5 章所述,以点号开头的文件。点号文件有预先已定好的名字。回想一下当使用 ls 命令来显示文件名时,点号文件不会被显示。要包括点号文件,你需要在 ls 命令中加入 -a 选项。点号文件将在 14 章详细讨论。

6.1.1 文件类型

正如在第 5 章中所知道的,linux 中所有文件只有一种物理格式-byte 流。byte 流只是一个 byte 的序列。这一点使 linux 将文件概念应用于系统中所有的数据元素。目录和设备都被分类成文件。将所有东西都当作文件对待,可使 linux 很方便地组织并交换数据。文件中的数据可以被直接送往设备,例如屏幕因为设备与常规文件一样,与系统有着相同的 byte 流文件格式接口。

这种相同的文件格式被用于其它操作系统元素的执行。设备,诸如屏幕或键

盘的接口也被设定为一个文件。其它元素,如目录,本身也是 byte 流文件,但它们有着不同的内部组织形式。一个目录文件包含着目录信息,并以一种特殊的目录格式组织。既然这些不同的元素都被作为文件对待,他们可被视为构成不同的文件类型。字符设备是一种文件类型。目录又是另一种文件类型。这些文件类型的数目因你所用 Linux 类型不同而变化。然而,有四种常见的文件类型:普通文件、目录文件,字符设备文件和块设备文件。尽管你可能很少会引用文件的类型,但搜索目录或设备时它还是非常有用的。在本章的后面,你将会看到在 find 的命令的搜索标准中,如何利用文件类型的确定搜索的目录和设备。

6.1.2 文件分类(classification):文件和 od 命令

尽管所有的普通文件都有 byte 流的格式,但它们有不同方面的应用。最显著的区别是二进制和文本文件之间。编译过的程序是二进制文件。然而,即便是文本文件也可根据不同的用途加以分类。你可以创建包含 C 编程源代码或 shell 命令的文件,甚至可以有空的文件。文件也可以是可执行程序或目录文件。Linux 中的 file 命令可帮你判定一个文件的用途。它对文件的头几行进行检查并试图对文件进行分类。file 命令在文件头几行中查找特殊关键词或特殊数字,但这一命令并不总是正确的。在下一个例子中,file 命令检查了两个文件的内容并对它们进行了分类。

```
$ file monday reports
monday:  text
reports: directory
```

为了描述文件分类的不同,在下一个例子中,file 命令检查了一个 C 源代码文件,一个可执行文件和一个空文件。

```
$ file calc.c proj newdata
calc.c: C program text
proj:  executable
newdata: empty
```

file 命令也接受 -f 选项使你可以从文件读入文件名,而无须在命令行上敲入。在下一个例子中,文件名从文件 myfile 中读出。

```
$ cat myfiles
calc.c proj newdata
$ file -f myfiles
calc.c: c program text
proj: executable
naodata: empty
```

如果你要逐字节地检查文件,可以使用 od 命令完成。od 命令执行对文件的 (dump)。缺省情况下,它将以八进制形式打印文件每一个字节。当然,你也可以指定字符,十进制或十六进制的表示形式。od 命令在你需要查找文件中的特殊字符,或显示二进制文件时是非常有帮助的。如果你进行字符的清查,某些特定无法打印的字符将以字符定义的形式表示。例如,回车键将用 \n 表示。file 和 od 命令的选项都列在本章后的表 6-1 中。

6.2 文件结构

Linux 将文件组织在一个分层相连的目录集中。每一个目录中可包含文件或其它的目录。在这种意义上,目录有两个重要的功能。一个目录包含文件,正如同文件放入文件柜中,一个目录又同其它目录相关联,正如同树枝与其它的树枝相关联。对文件而言,目录就如同文件柜,每个抽屉之中都放着几个文件。要访问文件,你只要打开抽屉即可。然而,同文件抽屉不同,目录不只可以包含文件,还可以包含其它目录。这样,一个目录便可与另一个目录关联。

因为这一结构与树的相似性,在计算机术语中它通常被称作树形结构 (tree structure)。然而,把它想像成一株倒置的灌木更为贴切。没有主干,树是倒置的,头上根下。由根部向下扩展。每一个分支都只来自另一个分支,但它可以有許多低层的分支。在这种观点上,可以认为它有一种父子结构 (parent child structure)。同样,每一个目录也是另一个目录的子目录,每一个目录中都可以包含许多子目录,但它只是某一个父目录的子目录。

图 6-1 描述了文件的分层结构。由顶部的根目录开始,其它的目录不断分支。每个目录都包含其它几个目录或文件,但每个目录只能有一个父目录。例如,目录 chris 有两个子目录,reports 和 pregrams,然而 Chris 只跟一个父目录相连,这一目录叫作 home。

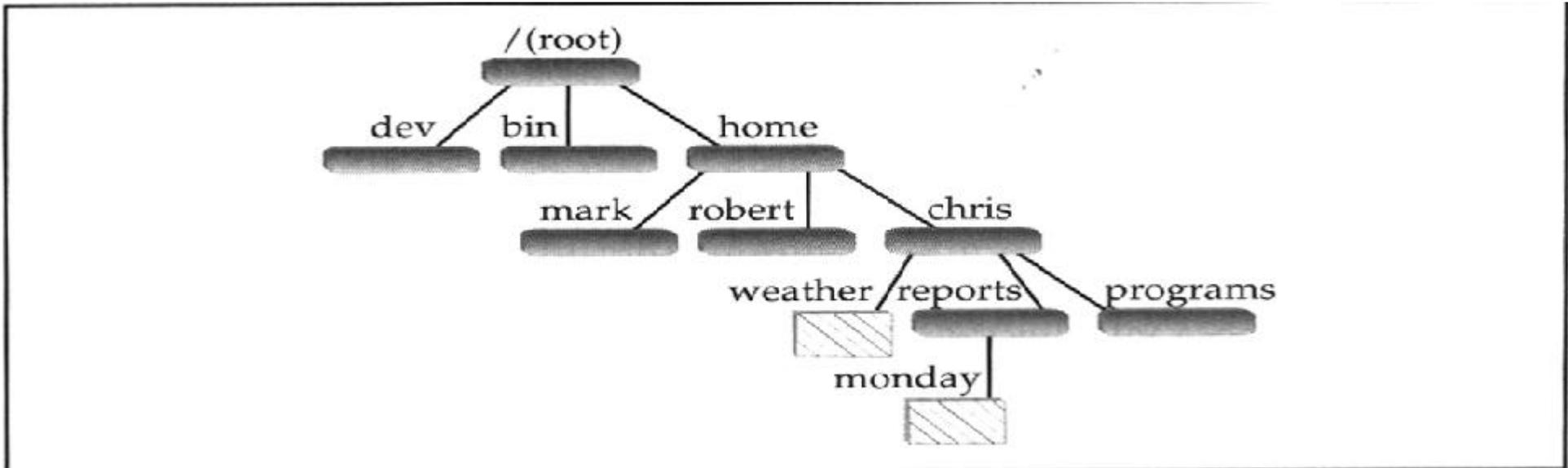


图 6-1Linux 树结构

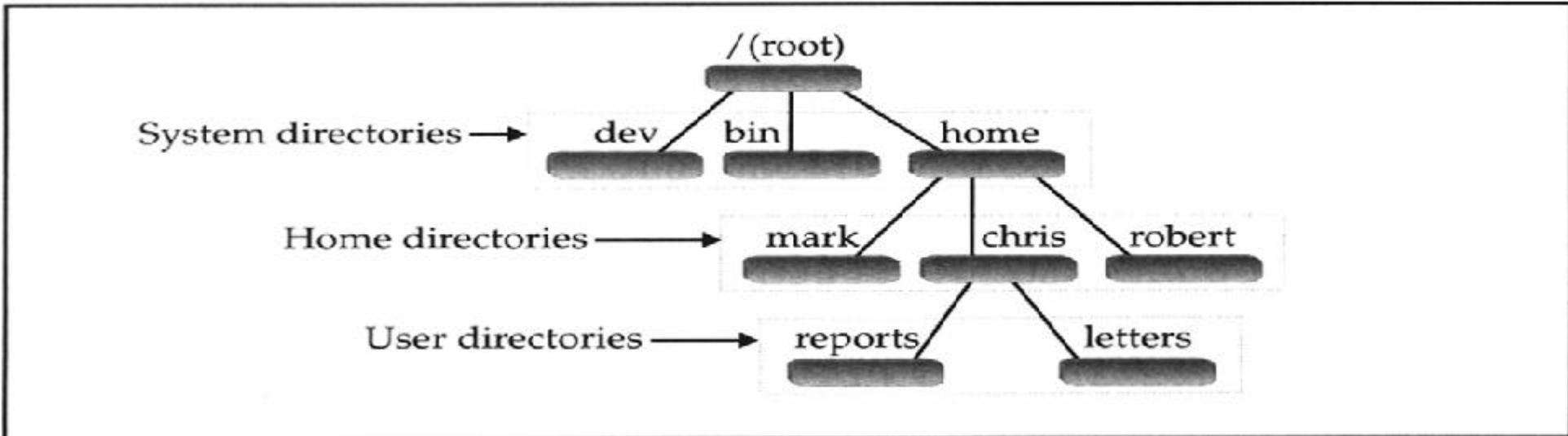


图 6-2Linux 文件结构，从根目录开始，包括系统、home 和用户目录

Linux 文件结构以根目录开始,分支成几个目录。在根目录中有几个系统目录,它们包括着描述 Linux 系统特征的文件和程序。根目录也包含着一个用于容纳系统中所有用户的 home 目录。每个用户的 home 目录中都包含着用户自行创建的子目录。这其中的目录都可以再包含其它目录。这一网状目录将从用户的 home 目录进行分支。

6.2.1 Home 目录

当你登录入系统,你便处于你的 home 目录中。系统起这个目录起的名字同你登录的名字相同。你第一次登录后创建的所有文件都会被置于你的 home 目录中。然而,在你的 home 目录中,你也可以创建其它新的目录。你可以转向这些目录并把文件存于其中。这于系统中其它用户而言也是相同的。每一个用户都有自己的以正确的登录名标识的 home 目录。它们也可以用于创建其它自己的目录。

通过目录名或将目录定为缺省目录,你便可以访问一个目录。每个目录在创建时都有一个名字。在文件操作中,你可利用这个名字进行目录中的文件访问。你也可比将目录设为你的缺少目录。如果在文件操作中你不使用任何目录名,那么将会对缺省目录进行访问。缺省目录被叫作工作目录(working directory)。在这种意义上,工作目录是你当前正在进行工作的目录。

当你登录时,工作目录便是你的 home 目录,它通常与你的登录名有相同的名字。你也可以更换工作目录,用 cd 命令来指定另一个工作目录。工作目录更换之后,你便可由一个目录移往另一个目录。另一种考虑目录的方法是把它认为一

条走廊。在走廊上,每个门上都有自己的名字。一些门通向房间,另一些门通向其它的走廊。通向房间的门就是目录中的文件。通向其它走廊的门就是其它的目录。从一个走廊去往另一个走廊就是更换工作目录。穿越几条走廊即是更换几个目录。

6.2.2 路径名

在创建文件或目录时,你所给的名字并不是它们的全名。一个目录的全名是它的路径名。目录之间分层的网状关系构成了路径,这些路径可无歧义的标识及引用目录和文件。在图 6-3 中,有一条路径从根目录"/"开始,通过 home 目录到达 robert 目录。有另一条路径,从根目录通过 home 和 Chris 目录到达 report 目录。尽管路径的某些部分开始是一样的,但在某一点它们不同了。目录 robert 和 report 有两个目录是共享的:根和 home。然后它们便不同了。在 home 目录中,robert 以 robert 告终,而目录 Chris 还引向 reports.这样,文件结构中的每个目录都可以说有着单独的路径。系统来标识一个目录的真实名字总是一个以根目录开头及包含目录上方所有网状目录的名字。

在 Linux 中,在路径中列出各个目录名并用斜杠加以分隔就可以写出路径名。路中第一个目录前的斜杠代表根目录。目录 robert 的路径名是/home/robert. 目录 reports 的路径名是/home/chris/reports.

路径名也适用于文件。当在目录中创建一个文件时,你起文件起了一个名字。然而,系统标识文件的真正名字是文件名和从根目录到文件所在目录的目录路径。在图 6-4 中,文件 weather 的路径包括根目录,home 和 chris 目录以及文件

名 weather。weather 的路径名是 /home/chris/weathes.(根目录由第一个斜杠代表)。

路径名可以是绝对的或相对的。绝对路径名是文件或目录由根目录开始的完全路径名。相对路径名(relative path name)是工作目录开始的,它是与你的工作目录相对的文件的路径。

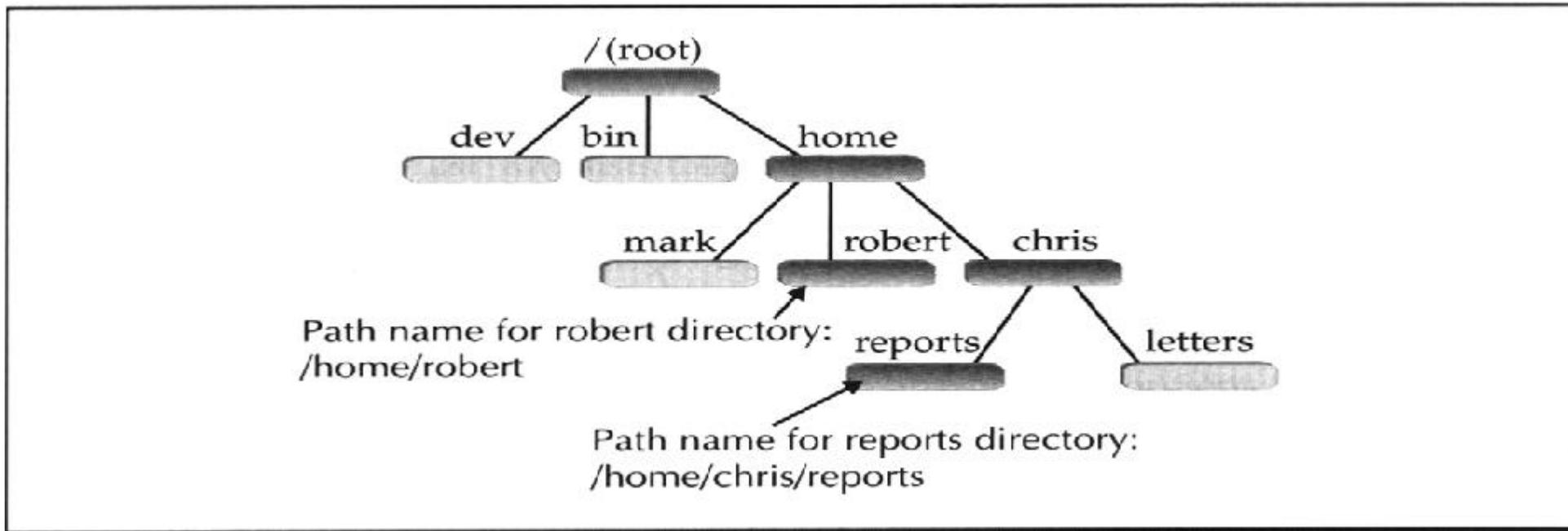


图 6-3 目录路径名

利用图 6-4 所示的目录结构。如果 chris 是你的工作目录。文件 monday 的相对路径是 /reports/monday monday 的绝对路径名是 /home/chris/reports/monday.

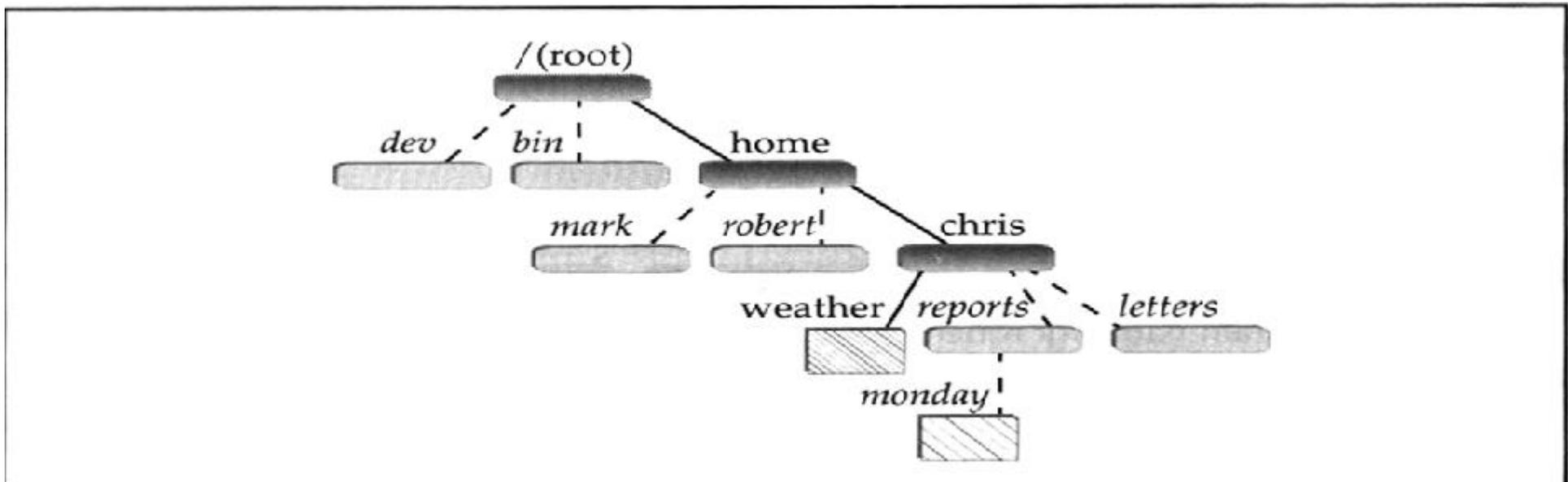


图 6-4 weather 的路径名 : /home/chris/weather

6.2.3 系统目录

开始 Linux 文件结构的根目录包含几个系统目录。系统目录中包含着系统运行和维护的必不可少的文件和程序。许多还包含有子目录,子目录中有许多系统设备的接口,如打印机或终端。表 6-2 列出了基本的系统目录,图 6-5 表明这些目录在树形结构中是如何组织的。

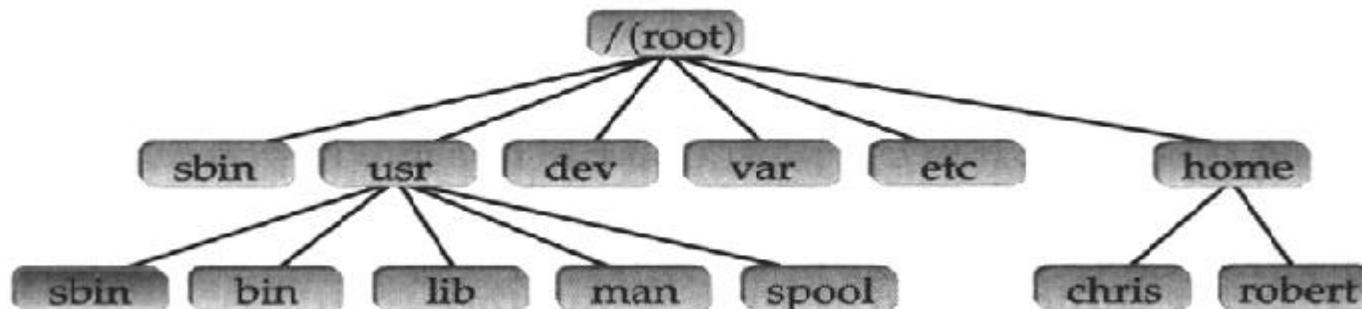


图 6-5Linux 中系统目录

6.3 列表、显示及打印文件：ls, cat, more 和 lpr

操作系统的主要功能之一是文件管理。你可能会对你的文件执行某些基本的输出操作,诸如把它们显示在屏幕上或把它们打印出来。linux 系统提供了一套命令来完成诸如列表、显示、文件打印及拷贝、复命令、文件、删除等基本文件管理操作。这些命令通常是由简写的词构成。例如,ls 命令是“list”的化简形式,它将对目录中的文件进行列表。lpr 命令是“line print”的……,它将打印文件。cat 和 more 命令将文件的内容显示在屏幕上。表 6-3 列出了这些命令和它们的不同选项。

当登录到你的 linux 系统中后,你可能会对你目录中的文件进行列表。ls 命令会输出一个你的文件和目录名的列表,它是非常有用的。ls 命令有许多选项用于

显示有不同特征的文件的列表。这些在第 5 章中有详细的讨论。

6.3.1 显示文件:cat 和 more

你可能需要查看一个文件的内容。cat 和 more 命令将文件的内容显示在屏幕上。cat 代表“concatenate”(连锁)。正如第 7 章所述,它实际上是一个非常复杂、功能强大的命令。这里只是被有限的利用,用于将文件的文本显示在屏幕上。

```
$ cat sncydata
computers
```

cat 命令将一个文件的全部文本一次全部送往显示设备。将文件较大时,就带来一个问题。因为文本在屏幕上迅速地闪过,more 命令即是设计用来克服这一限制的,它一次可只显示一屏的文本。你便可以随心所欲地在文本上上下下移动。你可键入 more 命令,然后键入你想查看的文件名来使用 more 命令。

```
$ more mydata
```

当 more 命令调用一个文件时,显示第一屏的文本。按 f 键或空格可进入下一屏。要想在文本中前移,你可按 b 键。按 q 键即可退出。

6.3.2 文件打印:lpr,lpq 和 lprm

当你希望打印文件时,可使用 lpr 命令将文件送往与系统相连的打印机。在下一个例子中,用户打印了 mydata 文件。

```
$ lpr mydata
```

如果你一次想打印多个文件,那么在命令行的 `lpr` 命令之后你可以指定不止一个文件。在下一个例子中,用户打印文件 `mydata` 和 `preface`。

```
$ lpr mydata preface
```

打印作业被放入队列中,并在后台依次打印。当进行文件打印时,你可以继续从事其它的工作。用 `lpq` 命令你可以看到某一时刻一个打印任务的位置 `lpq` 给出打印任务的 用户(发送作业用户的登录名),打印任务的 ID,以 `bye` 标识的大小,以及它所贮存的临时文件。在下一个例子中,打印任务的 用户是 `Chris` 打印功能是 `000/5`。

```
$ lpq
```

Owner	ID	Chars	Filename
Chris	000/5	360	/usr/lpd/cfa 000/5

如果您想取消一个不需要的打印任务,你可以利用 `lprm` 来完成。`lprm` 以打印作业的 ID 号或者其用户名作为参数。`lprm` 会将打印任务从打印队列中移出。在这项工作中,`lpq` 是非常有用的,它可以给你提供打印任务 ID 号和用户名以便 `lprm` 使用。在下一个例子中,15 号打印任务被取消。

```
$ lprm 000/5
```

你的 Linux 系统可以连接若干台打印机,其中一台会被指定为缺省打印机,除非指定另一台打印机,否则 `lpr` 命令将会在此打印机上打印。用 `lpr` 命令你可以指定想要进行打印的特定的打印机。系统中的每一台打印机都有自己的名字。利用 `-p` 选项及打印机的名字,你可以指定使用哪一台打印机。在下一个例子中,文件 `mydata` 在打印机 `enansi` 上进行打印。

```
$ lpr -Pevansl mydata
```

6.4 目录管理 :mkdir, rmdir, ls, cd 和 pwd

如第 4 章所述,你可以创建并移动你自己的目录,也可以改变自己的工作目录,利用 mkdir,rmdir 和 cd 命令。每个命令都可将目录的路径名作为自己的参数。pwd 命令会显示你工作目录的绝对路径名。除这些命令之外,特殊字符如点号、双点号和 ~ 符号都可被用来引用工作目录,工作目录的父目录和 home 目录。所有这些命令加在一起,你便可以对你的目录进行管理。你可以创建网状的目录,从一个目录移动到另一个目录,或利用路径名来引用你其它的目录。用于管好目录的这些通常的命令在表 6-4 中列出。

6.4.1 创建和删除目录 :mkdir 和 rmdir

用 mkdir 和 rmdir 命令你便可以创建和删除目录。两个情况下,你都可以使用目录的路径名。在下一个例子中,用户创建了目录 reports.然后用户用路径名创建了 letters 目录。在 rmdir 命令后加入目录名可以删除目录。在下一个例子中,用户用 rmdir 命令删除了目录 reports,然后用路径名删除了目录 letters.

```
$ rmdir neports
```

```
$ rmdir/home/chris/letters
```

6.4.2 目录列表 :ls

你已看到如何使用 ls 命令列出工作目录中的文件和目录。然而,为了区分文件名和目录名,你需使 ls 命令的 -F 选项。然后列表中每个目录名后会加入一条斜线

```
$ ls
loathes neports letters
$ ls -F
weather reports/  letters/
```

ls 命令也可以目录名或目录路径名作参数。这样你无需先改换向该目录,便可列出此目录中的文件。在下一个例子中,ls 命令的参数是一个目录名,reports。然后 ls 命令又被执行了一次,不过这次使用了 reports 的绝对路径名。

```
$ ls reports
monday tuesday
$ ls/home/chris/reports
monday tuesday
$
```

6.4.3 路径名 :pwd 命令

在每个目录中,你可以再创建另外的目录,即在实际上成为网状目录。利用 cd 命令,你可以一个目录改换入另一个目录。然而,没有指示符来告知你目前处于哪

一个目录中。要找出你改换到了哪个目录,用 `pwd` 命令可以显示你目前的工作目录。`pwd` 命令不仅仅显示目录名,它显示了整个路径名,如下例所示,这里显示的路径名包括 `home` 目录,`dylam` 它是 `home` 目录的一部分。每一个目录名都用斜线隔开。根目录以开头的斜线表示:

```
$ pwd
/home/dylan
```

6.4.4 更换目录: `cd` 命令

如何所知的,用 `cd` 命令可以更换目录。更换后的目录当成为工作目录,它是你的缺省目录。文件名诸如 `ls` 和 `cp`,都将对你工作目录中的文件除当操作当特别的指定。

当你登录入系统,你的工作目录是你的 `home` 目录。当一个用户帐号创建时,系统也为该用户创建了一个 `home` 目录。当你登录时,你便总被置于 `home` 目录中。`cd` 命令允许你将另一个目录作为工作目录。在某种意义上,你可从你的 `home` 目录移到另一个目录中。这一目录便成为任何命令和新文件创建的缺省目录。例如,`ls` 命令将列出新的工作目录中的文件。`cd` 命令将你想转移的目录名作的参数。

```
$ cd directory-name
```

在下一个例子中,用户从 `home` 目录更换到 `props` 目录中。用户执行了 `pwd` 命令来显示工作目录。

```
$ pwd
```

```
/home/dylan
$ pwd
/home/dylan/props
$
```

注意,当你创建一个新目录时,你已经处于一个工作目录当中了。你所创建的任何目录都在工作目录之中。工作目录和你在其中创建的新目录之间有父子关系。工作目录是新建目录的父目录。如果用户在 home 目录中,创建了一个 props 目录,那么 home 目录便是 props 目录的父目录,props 是 home 目录的子目录。

你可用两点“.”来代表一个目录的父目录。在字面上,它表示着父目录的路径名。你可用 cd 命令和双点号移回到父目录。将父目录作为当前目录。在下一个例子中,用户进入 props 目录,然后又移回 home 目录。

```
$ cd props
$ pwd
/home/dylan/props
$ cd..
$ /home/dylan
```

如果你想改换回 home 目录,你只需键入 cd 命令本身,而无须加任何文件名参数。你将移回 home 目录,将主再次作为工作目录。在下一个例子中,用户改换回 home 目录。

```
$ pwd
/home/chris/letless
$ cd
```

```
$ pwd
/home/chris
```

你会发现你会频繁地用 home 目录进入其它目录,然后再返回。在下一个例子中,用户由自己的 home 目录,dylan,进入 props 目录。然后用户用单独的 cd 命令,又返回了自己的 home 目录。在每次改换之前,用户使用 pwd 命令来显示工作目录。

```
$ pwd
/home/dylan
$ cd props
$ pwd
/home/dylan/props
$ cd
$ pwd
/home/dylan
```

6.4.5 网状目录 (nested Directories)

让我们来看一下 cd 命令如何在一系列网状目录间穿梭的。在下一个例子中,cd 命令进入 letters 目录。mkdir 命令建了一个叫作 thankyou 的 letters 的子目录。再次使用 cd 命令,用户进入 thankyou 目录。在此目录中,又创建了一个目录 birthday。用户又进入该目录。每次,pwd 命令都显示了路名。最后,没有参数的 cd 命令回到了 home 目录中。ls 命令加 -R 选项将打印出,工作目录下的所

有网状子目录。

```
$ pwd
```

```
/home/chris
```

```
$ cd letters
```

```
$ pwd
```

```
/home/chris/letters
```

```
$ mkdir  thankyou
```

```
$ cd thankyou
```

```
$ pwd
```

```
/home/chris/tetters/thankyou  letters:
```

```
$ mkdir birthday
```

```
$ cd birthday
```

```
$ pwd
```

```
/home/chris/letters/thankyou/birn
```

```
$ cd
```

```
$ pwd
```

```
/home/chris
```

```
$ ls -R
```

```
thankyou
```

```
letters/thankyou
```

```
birthday
```

```
letters/thanbyan/birthday
```

\$

6.4.6 引用工作目录和父目录：.和..

一个目录总是有父目录的(当然,根目录除外)。例如,在上一列表中,thankyou目录的父目录是 letters 目录。将一个目录被创建后,便有两个入口。一个以.表示,另一个以..表示。点是表示目录的路径名,双点号表示目录的父目录的路径名。双点是可以命令中作为参数,代表的父目录的引用。单点代表目录本身。在下一个例子中,用户进入 letters 目录。ls 命令与.参数一起列出了 letters 目录中的文件。ls 命令和..参数一起列出了 letters 的父目录,Chris 中的文件名。

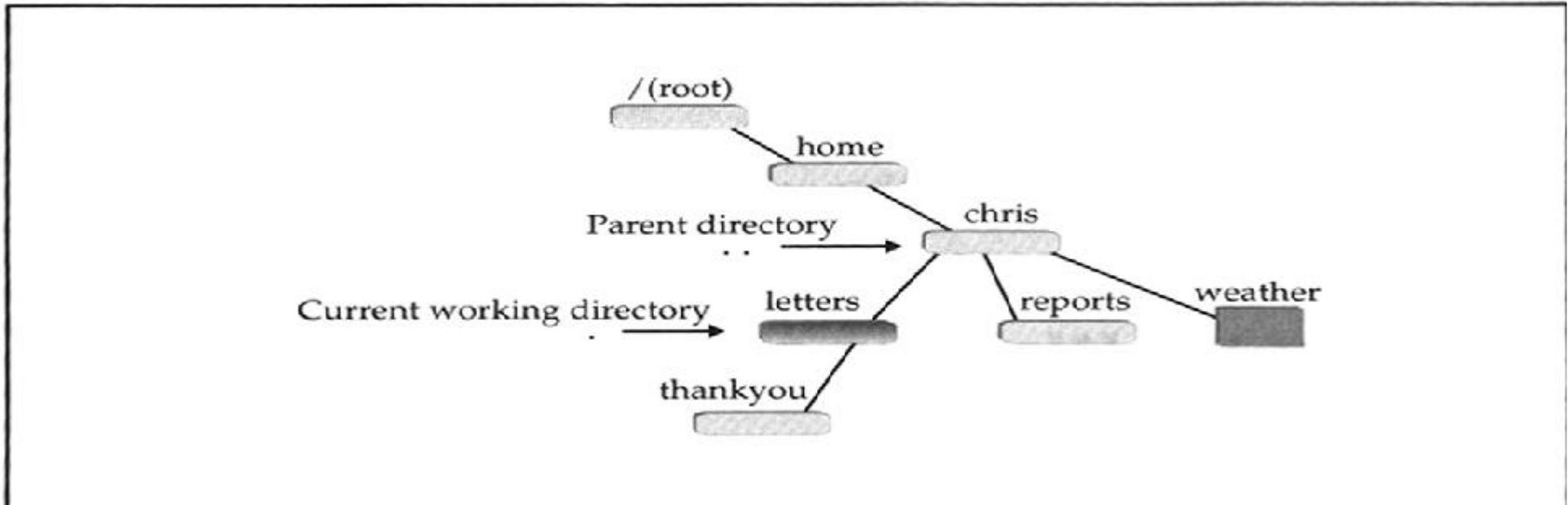


图 6-6 工作目录(用点表示)是 letters,它的父目录, chris 用两个点表示,是 home 目录

```
$ cd letters
$ ls.
thankyou
$ ls..
weathen letters
$
```

图 6-6 描述了点号和双点号代表工作目录和父目录的使用。正如上例所示,父目录是 home 目录:chris。工作目录是 letters。

你可以用单点号来引用工作目录,而不用它的路径名。例如,将一个文件以原名拷贝入工作目录,点号可用来代替工作目录的路径名。在这种意义上,点号是工作目录的另一个名字。在下一个例子中,用户将文件 weather 由目录 chris 拷贝入目录 reports 中。reports 目录是工作目录,因此可用点号代替。

```
$ cd reports
$ cp/home/chris/weather
```

双点是常用于引用父目录中的文件。下列中,cat 命令显示了父目录中的 weather 文件,文件的目录路径名是 ..是加斜杠和文件名。

```
$ cat ../weather
raining and warm
```

利用 cd 命令和 ..是你可以从一个低的目录间到连续的相应父目录中。在下一个例子中,user 在目录 thankyou 中。然后不断用户回到 chris 中,使用 cd 命令。

```
$ pwd
```

```
/home/chris/letters/thankyou
$ cd..
$ pwd
/home/chris/letters
$ cd..
$ pwd
/home/chris
```

很多时候你会同时使用 `..` 和 `.` 作为命令参数。例如,以 `letters` 作为工作目录,`weather` 可被拷贝到 `letters` 之中,只需用 `..` 引用 `chris` 目录及用 `.` 引用 `letters` 目录。

```
$ cp ./weather.
```

6.4.7 使用绝对和相对路径名:~

如前所述,你可用绝对或相对路径名来引用文件和目录。然而,每一个都有其限制。尽管绝对路径名可引用任何文件或路径,它通常以较长并且复杂,使之很难使用。相对路径名通常较短且易于使用,但它在引用文件的数目上受到限制。通常,你会尽可能地使用相对路径名,只有在必要时,只有在必要时,不使用绝对路径名。一些 Shell 提供了缩略路径名的方法。相对路径名由工作目录开始。在下一个例子中,`ls` 命令利用了文件 `thankgan` 的相对路径和绝对路径。工作目录是用户 `home` 目录,Chris `thakyan` 的相对路径名是 `letters/ thakyan`。

```
$ls letters/ thakyan
```

```
larisa
$ls/home/Chain/letters/thankyan
larisa
$.
```

相对路径名只能引用工作目录的子目录中的文件。这一子目录可以在任何深度,但它们的必须从工作目录开始分支。假如你要引用某一在目录树上更高的目录或在另一分子上的目录。例如说,reports 目录。在这种情况下,你需要利用 reports 的绝对路径名。在下一个例子中,用户用绝对路径名引用了 reports 目录中的 monday 文件。

```
$cat/home/Chris/reports/monday
```

将引用目录树中更高层目录时也需要绝对路径名。假定 thandyman 是工作目录,一个用户想显示你的 home 目录:/home/Chris 中的文件。Chris 目录不是 thankyan 的一个子目录,不能用相对路径名加以引用。在这种情况下,用户为了引用 home 目录中的文件就必须使用全路径名。在下一个例子中,用户想在 thankyan 目录中显示 home 目录:/home/chris 中的一个文件 weather。用户需要使用 weather 的绝对路径名来完成这一工作。

```
$Pwd
/Home/Chris/letters/thankyan
$cat/home/chris/weather
raining and waim
```

由根目录到你的 home 目录的绝对路径名可能非常复杂,并有可能因系统管理员的原因而改变。为了让它更易使用,你可以使用特殊字符,波浪线 ~,它代表

你的 home 目录的绝对路径名。在下一个例子中,用户在 weather 之前加入了波浪线和斜杠来完成对 home 目录中 weather 文件的引用。

```
$Pwd  
/home/Chris/letters/thankyou  
$cat /weather  
raining and warm  
$
```

你必须指定由 home 目录之后的其它路径。在下一个例子中,用户引用了 reports 目录中的 monday 文件中。波浪线代表了用户的 home 目录的路径:/home/Chris,然后又指定了到 monday 文件的剩余路径。

```
$cat /reports/monday.
```

6.5 文件和目录操作:find.cp.mv.rm 和 ln

随着你创建文件数目的增加,你可能希望将之备份,改变它们的名字,删除部分文件,或者给它们增加名字。Linux 为你提供了几个文件命令来查找文件、拷贝文件、重命名或删除文件。如果你有大量的文件,你可以对不查找并确定某一特殊文件。这些命令是整词的缩略形式,由两个字符组成。CP 命令代表"copy",可以完成文件拷贝命令,mv 代表"more",可对文件重命名或移动文件,rm 代表"remove",可以删除一个文件,ln 代表"link",可对文件添加一个名字。这一规则的例外是 find 命令,它对文件名进行搜索来找到一个文件。表 6-5 列出了包括它

们的选项在内的不同的操作。

6.5.1 查找目录:find.

一旦在许多不同目录中有大量的文件,你便需要对某一文件或某一类型的文件进行查找。利用 find 命令可以进行此种查找。find 命令的参数是目录名和几个指定查找类型和查找标准的可能的选项。然后 find 将在指定的目录及其子目录中查找符合标准的文件。find 命令可按文件名、文件类型,用户名甚至最新更改的日期来查找文件。

```
$find directory -fst option criteria
```

-name 选项确定了查找标准并指导 find 命令查找与此标准匹配的文件名。要按文件名进行查找,可使用 find 命令,加带目录名,其后再加上 -name 选项和文件名。

```
$find directory -list -name filename
```

find 命令也有些只是执行操作的选项,如将查找结果输出的操作。如果你希望 find 命令显示找到的文件,只须在命令行上将 -print 选项和其它任何选项加入即可。-print 选项指示 find 命令将所查找到的所到文件名送入标准输出。在下一个例子中,用户搜索了 reports 目录中所有以 monday 为名的文件。一旦找到,这一文件和它的相对路径名将会被打印出来。

```
$find reports -name monday -print  
reports/monday
```

find 命令将用目录列表中指定的目录名来显示一个文件名。如果你指定了

一个绝对路径名,找到的目录的绝对路径将被显示。如果指定了相对路径名,那么只会显示目录的相对路径。上例中,用户在目录列表中拽定了一个相对路径名,reports.查找到的文件名也以这个相对路径名为开始。在下一个例子中,用户指定了目录列表中一个绝对路径名,查找的文件名则会以绝对路径名输出。

```
$find /home/chais -name monday -print  
/home/chris/reports/monday.
```

如果你想搜索你的工作目录,可以使用目录路径名中的点来代表工作目录。双点是代表父目录。下一个例子用点来代表工作目录,搜索了工作目录中的所有文件和子目录。如果你位于 home 目录中,这是一种搜索你所有目录的方便的方法。注意找到的文件是以点开头而输出的。

```
$find -name weather -print  
./weather
```

你可以将 shell 特殊字符作为搜索标准的一部分。然而,特殊字符必须加以引用以免 shell 对之取值。在下一个例子中,将对 programs 目录中所有以 .c 为扩展的文件加以搜索。

```
$find programs -name '*.c' -print
```

搜索标准

你也可利用 find 命令来查找其它目录。在 Linux 中,一个目录被正式划分为一种特殊类型的文件。尽管所有的文件都有 byte 流的格式,但是一些文件,诸如目录,有特殊的用法。在这个意义上,一个文件也说是有一种文件类型。find 命令有一个 -type 选项,可用来查找一个文件或一种文件类型。-type 选项有一个字符代表文件类型。这一代表目录的字符是 d,在下一个例子中,利用了目录名和目录

文件类型来查找称作 thankyon 的目录。

```
$find /home/chris -name thankyon -type d -print  
/home/ chris/letters /thankyon
```

如上文所述,文件不同类型并不像操作系统的其它元素——诸如设备的文件格式那样是截然不同的文件。在这种意义上,一个设备被作为一种类型的文件看待,你可以用 find 命令来查找设备、目录,和普通的文件。表 6-6 列出了 find 命令的 -type 选项可使用的不同文件类型。

find 命令包含许多不同的搜索标准。你可以按文件大小、文件最后被修改的时间,它所具有的连接,它所属的组群,及其它一些标准来查找文件。表 6-6 中列出了这些不同的查找标准。其中两个比较常用的是 -size 和 -mtime 选项。每一个都以一个数字作参数。-size 选项缺省以块作度量单位,如果在数字后加入 c,它将以字符 cbyte 为度量单位。你可用 "+" "-" 来修改一个数字以查找比指定是大小要大或小的文件。例如 +100C 会选择超过 100 个字符的文件。-mtime 选项以它们多少天之前被修改过为查找标准。例如,利用这一选项,你可以查找你两天前使用过的文件。在下一个例子中,find 命令用 -size 选项来查找所有大于 10 个字节的文件。第二个例子中,find 命令用 -mtime 选项来查找三天前修改过的文件。

```
$find .-size +10c -print
```

```
.weather
```

```
./reports/monday
```

```
$find . -mtime +3 -print
```

```
./weather_
```

复杂的搜索

当选项在命令行上列出时,它们之间是潜在的与操作。只有满足所有要求的文件名才会被列出。然而利用引用的括号和逻辑上的 OR 和 NOT 操作符,你便可以创建复杂的搜索查询。find 命令的 NOT 操作将是一个惊叹号"!";放在任何搜索标准前的!号将对比标准取反。如果这一标准的匹配结果为假,那么文件便是有效匹配。便如,下述命令列出所以无.C 扩展的文件名。

```
$/s
main.c/ib.o  today
$find . !-name "*.c"-print
/ib.o  today
```

find 命令的逻辑或操作符是 -o,放在两个搜索标准之间的 -o 选项表明二者之间是或的关系。如果文件满足其一或两者的要求,那文件就是一个有效的匹配。你需要把 -o 每项和它的搜索标准操作数放入括号内。括号之前加左斜杠并在前后加入空格将会实现对括号的引用,下一个命令查找有 weather 名字的文件和目录。注意在引用的括号前后的空格。

```
$ find . \(-name weathee -o -type d\) -print
./weather
./reports
./cettters
./cettters/ thankyou
```

用引用的括号将搜索标准组合起来,你便可以实现复杂的逻辑操作。下列中 对一个名为 reports 的目录和所有尺寸大于 10byte 的文件进行查找。引用的括号放在 -name 和 -type 的与操作和与 -size 的或操作之外。这一命令的逻辑操作

可被表达为：

```
((name=reports) AND (file=directory type)) OR (size>10)
$find.\(\(-name reports -type d\)-o -size+10\)
```

-print
./reports
./weather

6.5.2 文件拷贝

要进行文件拷贝,只须将两个文件名指定为 cp 命令的参数。第一个文件名是要被进行拷贝的文件——已存在的那个文件。这通常被称作源文件。第二个文件名是拷贝的文件。这一操作会产生一个包含着源文件全部数据的新的文件。第二个参数通常被称作目标文件。cp 命令的语句如下:

```
$cp source-file destination-file
```

下列中,用户将一个叫作 proposac 的文件拷贝到叫做 oldprop 的新文件中。

```
$cp proposal oldprop
```

当用户对目录中文件列表时,新的拷贝也在列表中。

```
$ ls
```

```
proposat oldprop
```

你在不经意间用 cp 命令破坏了另一文件是有可能的。cp 命令进行拷贝时,首先创建一个文件,然后将数据拷贝入其中。如果另一文件与目标文件同为,那么这一文件将被破坏,而另一同名的新文件会被创建。在某种定义上,原有文件被新

的拷贝覆盖了。在下一个例子中,proposal 文件被 newprop 文件覆盖了。因为 proposal 文件已经存在了。

```
$cp newprop proposal
```

在 15 章,你将学习如何对系统进行设置来察觉这种覆盖。那时,用 cp 命令和 -i 选项会更安全。利用这一选项,cp 首先会察看这一文件是否已存在。如果已存在,便会问你是否愿意覆盖已存在的文件。如果你输入 y,这一文件将被破坏并创建一个新文件作为拷贝文件。如果你输入其它任何东西,它便认为是否定的回答,cp 命令将会被中断,保留原有的文件。

```
$cp -i newprop proposal
```

```
Orewrite proposal? n
```

```
$
```

将文件拷入目录

要将文件由工作目录拷贝到另一目录,只须将目录名作为命令的第二个参数即可。新的拷贝文件名与原来相同,但拷贝文件在一个不同的目录中。

不同目录中的文件可以有相同的名字。因为它们在不同目录中,它们被当作不同的文件登记。

```
$cp filename directly-name
```

要将一个文件由 home 目录拷贝入一个目录,你只须指定子目录名即可。下列中,文件 newprop 被由工作目录拷贝入 props 目录。

```
$cp newprop props
```

CP 命令可将几个文件名列表作为自己的参数,因此你可以同时将几个文件拷贝入一个目录。只要在命令行上列出文件名,并将目录名作为最后一个参数,然

后所有文件便被拷贝入目标在下一个例子中,用户将文件 `preface` 和 `doc1` 拷贝到 `props` 目录中,注意 `props` 是最后一个参数。

```
$cp preface doc props
```

你可以利用第 5 章介绍的特殊字符来创建文件名列表供 `cp` 或 `mv` 使用。例如,你想把所有的 C 源代码文件拷贝到指定目录。无须在命令行上将每个文件名逐一列出,只要用特殊符 `*` 和 `.c` 扩展即可匹配并列出的所有 C 源代码文件(所有此类文件都有 `.c` 扩展名)。在下一个例子中,用户将 `propes` 目录中所有文件拷贝到 `oldprop` 目录中。注意在 `.*` 之前 `props` 路径名的使用。在这种语境下,`props` 是一个加在 `.*` 产生的文件列表上的路径名。

```
$cp props/ .* oldprop
```

你当然可以使用其它的特殊字符,如 `."`、`"?"` 或 `"[]"`。在下一个例子中,用户将源代码和目标代码文件(`.c` 和 `.o`)拷贝到 `projbk` 目录。

```
$cp *.*[oc] projbk
```

当你拷贝文件时,你可能想给拷贝文件起一个新的名字。将新文件名对目录名之后,并以斜杠隔开即可完成这一任务。

```
$cp filename directory-name/new-filename
```

在下一个例子中,文件 `new prop` 被拷贝到 `props` 目录中,拷贝文件名字为 `version1`。用户然后进入 `props` 目录并对文件列表。其中只有一个叫作 `Version1` 的文件。

```
$cp newprop props/version/
```

```
$cd props
```

```
$ls
```

version1.

当你想把一个文件由子目录拷贝到父目录中时,你需要指定子目录的名字。`cp`命令的第一个参数是需要拷贝的文件名。文件名前必须加由斜杠分隔开的子目录的名字。第二个参数是将拷贝在父目录中的文件名。

```
$cp Child-directory-name/filename new-filename
```

下列中,文件 `version1` 被由 `props` 目录拷贝到 `home` 目录。

```
$cp props/version/ version1
```

假定,你已将工作目录更换到了一个子目录,你想将一个文件中子目录拷贝入父目录。你需要通过一些方法来引用父目录。你可以用双点号来完成,它代表着父目录的路径名。

```
$cp filename ..
```

```
$cp filename../new-filename
```

例如,如果 `props` 目录是你现在的工作目录,你想将文件 `Version` 的拷贝到它的父目录中(本例中,是用户的 `home` 目录),你需要把双点号作为 `cp` 命令的第二个参数。

```
$cp version1..
```

如果你想给以 `version/` 的拷贝起一个新名字,将名字加在第二个参数之后,名字之前加斜杠。

```
$cp version 1../newversion
```

6.5.3 移动文件

你可以使用 `mv` 命令来为文件改名或将文件由一个目录移入另一个目录。当利用 `mv` 来对文件改名时,只需将新文件名作为其第二个参数。第一个参数是你正在使用的文件名。

```
$mv original-filename new-filename
```

```
$mv proposal version1
```

与 `cp` 相同,`mv` 命令也很有可能意外地删除一个文件。将对文件更名时,你可能不小心选择了另一个已存在的文件的名称。`mv` 命令的一个选项也可以事先查看有相同文件名的文件是否存在。如果存在,会向你询问是否将之覆盖。下列中,一个以 `version1` 为名的文件已经存在。

覆盖状况被察觉到,你会被询问是否将那个文件覆盖。

```
$ls
```

```
proposal version1
```

```
$mv-i version1 proposal
```

```
over write proposal? n
```

```
$
```

你可将文件由一个目录移动到另一目录,只要将文件名作为 `mv` 命令的第二个参数。这种情况下,你可以认为 `mv` 命令是简单地将文件由一个目录移到另一个,而不是将文件更名。在将文件移动之后,文件将与在原目录中具有相同的文件名,除非你有特殊的指定。

```
$mv filename directory-name
```

在下一个例子中,文件 newprop 由 home 目录移动到 props 目录。

```
$mv newprop props
```

如果你想在移动文件时对文件更名,可在目录名后指定新的文件名。目录名和新的文件名之间由斜杠分隔。在下一个例子中,文件 new prop 移动到 props 目录并被更名为 verion1.

```
$mv newprops props/version1
```

```
$cd props
```

```
$ls
```

通过在文件名前指定子目录名,一个文件也可以被很容易地由一个子目录移动到父目录。

```
$mv props/version1 version1
```

假定,你已将你的工作目录移到工作目录的子目录中,你想将一个由子目录移到父目录中。如同 CP 命令一样,你也可以用双点是来引用父目录。

```
$mv filename ..
```

```
$mv filename ../new-filename
```

如果 props 是你现在的工作目录,你想将文件由 props 目录移到它的父目录中,-home 目录中,那你应把双点是作为 mv 命令的第二个参数:

```
$mv version1..
```

如果你想在 X 目录中给文件 revsion1 更换一个新名字,只要将新名字加在 mv 命令第二个参数之后,名字之前加斜杠。

```
$mv version1 ../ddprop
```

一个文件的实际名字是文件名之前加它的目录路径。当文件 tuesday 移到

reports 目录后,实际上它的路径名已经变了。文件 tuesday 的全名已由 /home/chris/tuesday 变成 /home/chris/reports/tuesday,它的路径名现在包括目录 reports。在这种意义上,将文件更名更像是将文件进行了移动。

利用绝对路径名也同样方便。在下一个例子中,文件 today 移到目录 reports 中,并更名为 tuesday,注意在 mv 和 ls 命令中都使用文件的绝对路径来作文件名参数。

```
$mv today/home/chris/reports/tuesday
$ls/home/chris/reports
monday tuesday
$
```

同 cp 命令一样,mv 命令也可以同时将几个文件由一个目录移到另一目录。你只需在命令行上列出文件名。你最后输入应该总是目标目录。在下一个例子中,用户将文件 wednesday 和 friday 移到了 lastleek 目录中。

```
$cp wednesday friday lastleek
```

你也可以利用第 5 章描述的任一个特殊字符产生文件名的列表以供 mv 命令使用。在下一个例子中,用户将现有目录中的所有源代码文件移到 newproj 目录中。

```
1209 $mv *.c newproj
```

如果你想将一个给定目录中所有的文件移到另一个目录,你可以用 *.* 来作匹配以产生这些文件的列表。在下一个例子中,用户将 reports 目录中所有的文件都移到 lepbks 目录。

```
$mv leports/*. * lepbks
```

6.5.4 目录的移动和拷贝

你也可以对目录进行移动和拷贝。cp 和 mv 都可以把目录名作为第一个参数,使你可以把子目录由一个目录拷贝或移动到另外一个目录中去。第一个参数是要被拷贝成移动的目录名,第二个参数是将要把子目录置于其中的目录名。文件适用的路径名结构同样适用于目录的移动或拷贝。

将子目录由一个目录拷贝到另一个目录也同样容易。要进行目录拷贝,cp 命令要求你使用一个选项。一个选项代表了“回归的”。它指示 cp 命令拷贝一个目录和它所包含的子目录。换句话说,整个的目录树,由那个目录起,都会被拷贝。在下一个例子中,目录 thankyou 被拷贝到目录 oldletters 中。现在就有两个 thankyou 子目录了。一个在 letters 中,另一个 oldletters 中。

```
$cp-r letters/thankyou oldletters
```

```
$ls-F letters
```

```
/thankyou
```

```
$ls -F oldletters
```

```
/thankyou
```

假定,你只想拷贝一个目录中的文件,而不想拷贝一个目录,使之成为另一个目录的子目录。要将一个目录中的全部文件拷贝到另一个目录中,你必须指定它们的文件名。特殊字符星号可以匹配一个目录中的全部文件和子目录名。要将 letters 目录中的全部文件拷贝到 oldletters 目录中,你应将星号作为你的第一个

参数以便产生 letters 目录中所有文件名的列表。如果你要为第一参数指定一个路径名,你可以把星号放在末尾。在下一个例子中,letters 目录中所有的文件都被拷贝到 oldletters 目录。为 letters 指定了一个路径名,路径名末尾的星号匹配了 letters 目录中所有的文件。

```
$cp-letters/* oldletters
```

为了将子目录包含在拷贝操作中,你需要使用 cp 的一个选项。

```
$cp-r letters/* oldletters
```

6.5.5 特殊字符 ~

你已经看过你可以如何利用波浪线来代表 home 目录的绝对路径名。例如,为了将一个文件由低层的目录拷贝到 home 目录中,你可以用波浪线来代表 home 目录的绝对路径名。在下一个例子中,用户进入 reports 目录,然后将文件 monday 由 reports 目录拷贝到 home 目录中。

```
$cd reports
```

```
$cd monday ~
```

为了在将文件拷贝到 home 目录时给文件更换一个新的名字,你可将名字放在 ~ /之后。在下一个例子中,文件 monday 被拷贝回 home 目录中,拷贝被更名为 today。

```
$cp monday ~/today
```

在 mv 命令中也可以用同样的方法使用波浪线。在下一个例子中,文件 monday 被从 reports 目录移动到 home 目录。

```
$mv monday ~
```

如果将文件由低层目录移到 home 目录时你想将之更名,新名字前应加波浪线和斜杠:~/。在下一个例子中,用户进入了 reports 目录,文件 monday 被移动回 home 目录并被更名为

```
today。
```

```
$cd reports
```

```
$mv monday ~/today
```

当你将利用 home 目录的路径名时你就可以使用波浪线。在下一个例子中,以前介绍过的一个 mv 和 ls 命令与波浪线一齐执行。

```
$mv weather ~/reports/monday
```

```
$ls ~/reports
```

```
monday
```

```
$
```

6.5.6 删除文件:rm 命令

当你使用 linux 时,你会发现你所使用文件的数目剧增。在 linux 中创建文件是很容易的。诸如象编辑器的应用程序,象 cp 命令,都很容易地可以创建文件。最终,这其中的许多文件会变得过时且毫无用处。你可以用 rm 命令将之删除。在下一个例子中,用户删除了文件 oldprop。

```
$ rm oldprop
```

rm 命令可接受任何数目的参数,使你可以列出几个文件的名字并一次把它们

同时删除。你只要在输入 `rm` 之后把它们列在命令行上即可。

```
$rm proposal version1 version2
```

使用 `rm` 命令时要小心。它是不可恢复的。一旦一个文件被删除,它是不能被修复的。例如,假如你在输 `cp,rm` 或其它命令时,不小心误输入了 `rm` 命令。当你按了回车键并认识到自己的错误时,已经太晚了。文件已经没有了。为了防止此种情况的发生,你可以使用 `rm` 命令的一个选项来确保你想删除一个文件。利用一个选项,你会被每个文件揭示是否确定将之删除。如果你输入 `y`,文件将被删除。如果你输入任何其它东西,文件将被保留。在下一个例子中,`rm` 命令被指示去删除文件 `proposal` 和 `oldprop`。然后它会询问对每个文件的确认。用户决定删除 `oldprop`,保留 `proposal`。

```
#rm -i proposal oldprop  
Remove proposal? n  
Remove oldprop? y  
$
```

6.5.7 连接 :ln 命令

用 `ln` 命令有两个参数:文件的原名和文件新的添加名。`ls` 操作会列出两个文件名,但是物理文件只有一个。

```
$ ln original-file-name added-file-name
```

在下一个例子中,文件 `today` 有一个另外的名字 `weather`。它是 `today` 文件的另一个名字。

```
$ls
today
$ln today weather
$ls
today weather
```

将 ln 命令对同一文件使用多次,你可以给同一文件几个不同的名字。在下一个例子中,文件 today 有了 weather 和 weekend 名字。

```
$ln today weather
$ln today weekend
$ls
today weather weekend
```

你可以使用 ls 命令的一个选项来察看一个文件有几个连接。有一个选项的 ls 命令列出了若干条信息,诸如许可(将在下章中介绍),和一个文件拥有的连接数,文件大小和最后修改的日期。在这行信息中,在用户登录名前的第一个数字,标定了文件的连接数。在日期之前的数字是文件的大小。日期是文件最后被修改的时间。在下一个例子中,用户列出了文件 today 和 weather 的全部信息。注意二者的连接数都为 2。而且,文件大小和修改日期相同。这意味着这两个文件确实是一个文件的不同名字。

```
$ls -l today weather
-rw-rw-r-- 2  Chris group 563 Feb 14 10:30 today
-rw-rw-r-- 2  Chris group 563 Feb 14 10:30 weather
```

这些仍然没有告诉你哪些文件名是确切相连的。当文件有相同连接数、大

小和修改日期时,如上面 today 文件和 weather 文件的情况,你在某种程度上可以确认二者是同一文件。然而,要想完全确认,你应使用 ls 命令的一个选项。利用一个选项,ls 命令列出了文件名和它的 i 节点(inode)数。I 节点数是系统来标识文件的一个特殊数字。如果两个文件有相同的 I 节点数,它们代表了相同的文件。它们是同一文件的不同名字。在下一个例子中,用户列出了文件 today,weather 和 larisa。注意 today 和 weather 有相同的 I 结点数。

```
$ls -i today weather larisa  
1234 today 1234 weather 3976 larisa
```

添加名,或连接,它们由 ln 命令创建,并通常用于在不同目录中对同一文件的引用。在一个目录中的文件可以被链接到另一个目录并在另一个目录中进行访问。假定你想在另一个目录中引用 home 目录中的文件。你可在那个目录中建立一个 home 目录中文件的连接。这一连接实际是文件的另一个名字,因为连接在另一个目录中,它可与原文件有相同的名字。要将 home 目录中的一个文件连接到另一个目录,将另一个目录名作为 ln 命令的第二个参数即可。

```
$ln filename directory-name
```

在下一个例子中,Chris 目录中的文件 today 被链接到目录 reports 中。ls 命令会列出 Chris 目录中的文件 today 被链接到目录 reports 中。ls 命令会列出 Chris 目录和 reports 目录中的 today 文件,实际上,只存在 today 文件的一份拷贝,即 home 目录中的原文件。

```
$ln today reports  
$ls  
today reports
```

```
$ls reports
today
$
```

如同 cp 和 mv 命令一样,你也可以给连接一个新名字。只要将新名字放在目录名之后,并加斜杠分隔即可。在下一个例子中,文件 today 被用 wednesday 的名字连接到目录 reports 中。这时也只有有一个真实的文件,即 Chris 目录中叫作 today 的原文件。然而,现在 today is wednesday 的名字连接到了目录 reports 中,在这种意义上,today 有了另外一个名字。在目录 reports 中,today 以 wednesday 的名字存在。

```
$ln today reports/wednesday
$ls
today reports
$ls reports
wednesday
$
```

你可以引用文件的路径名很轻易地把一个目录的一个文件连接到另一个目录的另一个文件中。在下一个例子中,reports 目录中的文件 monday 被连接到目录 Chris 中。注意第二个参数是绝对路径名。

```
$ln monday/home/Chris
```

要删除文件,你需要删除它所有的连接。一个文件名实际上被认为是到文件的一个连接。命令 rm 将到文件的连接删除。如果你的文件有几个连接而你只删除了其中一个,其它的原地未动,那么你便可以通过其它的文件名来引用文件。

即使你将初始的连接——文件的初始文件名删除,情况仍然如此。任何增加的连接都会一样地起作用。在下一个例子中,today 文件被 rm 命令删除。然而,同一个文件还有一个叫 weather 的连接,文件可由 weather 的名字加以引用(见图 6-7)。

```
$ln today weather
```

```
$rm today
```

```
$cat weather
```

```
The storm broke today and the sun came out.
```

```
$
```

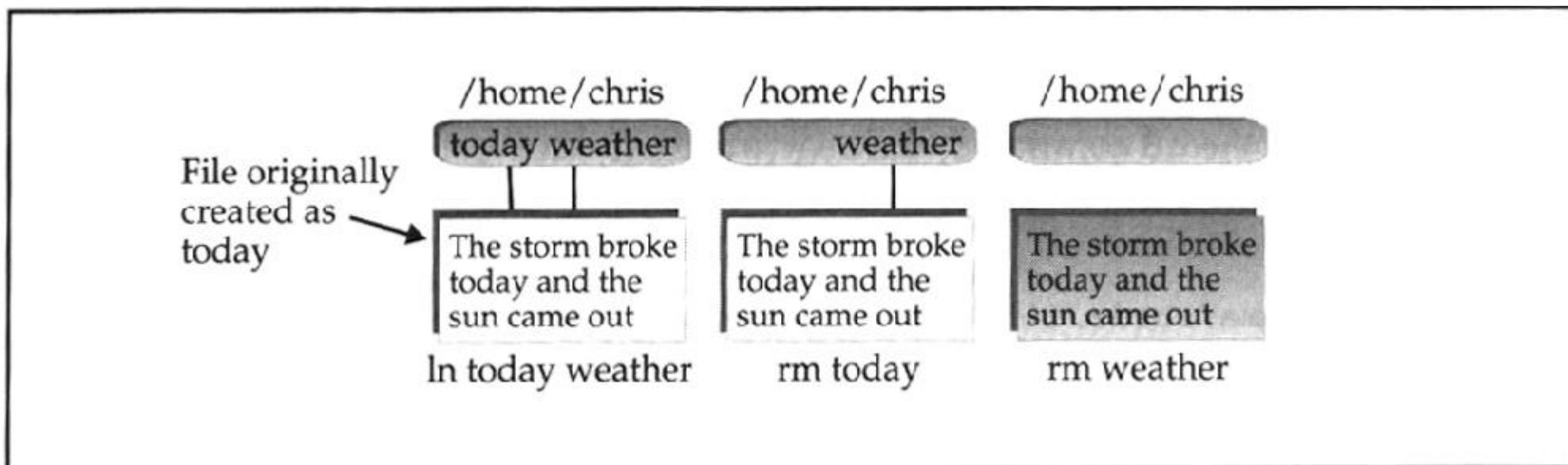


图 6-7 要删除一个文件,你需要删除它的所有链接

符号连接和硬连接

Linux 支持所谓的符号连接,就我们所介绍的而言,被称作硬连接。尽管,硬连接可以满足你大部分的需要,但它仍然有很大的局限性。当你试图连接其它用户的目录中的文件时,硬连接可能会失败。这是因为 Linux 文件结构可被物理上划分为称作文件系统的部分。一个文件系统可由许多物理存储设备或设备组成,从一个软盘到一系列的硬盘。尽管所有文件系统中的文件都被连接到同一个总的目录树中,每一个文件系统在物理上将管理自己的文件和目录。这意味着一个文件系统中的文件不能由硬连接到另一个文件系统的文件上。如果你试图连接另一个文件系统中另一用户目录中的文件,硬连接将会失败。

使用符号连接使可以克服这一限制。符号连接中包含着它所连接到的文件的路径名。它不是一个直接的硬连接,而是如何定位一个特定文件的信息。不像

硬连接一样为同一个文件再登记一个名字,符号连接只是被认为是代表路径名的另一种符号。它是文件路径名的另一种写法。

使用 `ln` 命令的 `-s` 选项你便可以创建一个符号连接。在下一个例子中,用户为文件 `/home/george/veglist` 创建了一个叫作 `lunch` 的连接。

```
$ ln -s lunch/home/george/veglist
```

如果你列出了一个符号连接和它的文件的信息,你会发现显示的信息是不同的。在下一个例子中,用户用 `ls` 命令和一个选项列出了 `lunch` 和 `/home/george/veglist` 的全部信息。命令行中第一个字符标识了文件的类型。符号连接有它们自己的文件类型——用 `L` 表示。`lunch` 的文件类型是 `L`,表明它是一个符号连接,而不是一个通常的文件。在术语“group”之后的是文件的尺寸。注意到二者的大小是不同的。文件 `lunch` 仅有 4byte 大。这是因为 `Lunch` 只是一个符号连接——包含着另一文件的路径的文件——路径名只会占用几个字节。它不是文件 `veglist` 的一个直接的硬连接。

```
$ ls lunch/home/george/veglist
```

```
Lrw-rw-r--1 Chris group 4 Feb 14 10:30 lunch
```

```
-rw-rw-r--1 george group 793 Feb14 10:30 veglist
```

要删除一个文件,你只需删除它的硬连接。如果还有一些符号连接,它们不会完成对文件的访问。在这种情况下,符号连接中将包含一个已经不存在的文件的路径名。与硬连接不同,使用符号连接你可以创建目录的连接。在效果上,你可以创建另一个名字来实现对目标目录的引用。然而,如果你用符号连接来代表一个目录名,你要记住 `pwd` 命令总是会显示实际的目录名,而不是符号名。在下一个例子中,用户将目录 `thankyou` 连接到符号连接 `gifts` 上。当用户使用 `cd` 命令和 `gifts`

时,用户实际上将进入 `thankyou` 目录。`pwd` 会显示 `thankyou` 目录的路径名。

```
$ln -s/home/chris/letters/thankyou gifts
```

```
$cd gifts
```

```
$pwd
```

```
/home/chris/letters/thankyou
```

```
$
```

如果你想显示符号连接的名字,你可用 `cwd` 变量对之访问。`cwd` 是一个特殊的系统变量,它容纳着一个目录的符号递接的名字,如果存在这一符号连接的话。`cwd` 变量将在 15 章加以讨论。你可 `echo $ cwd` 命令显示了 `cwd` 的内容。

```
$pwd
```

```
/home/Chris/letters/thankyou
```

```
$echo $cwd
```

```
/home/Chris/gifts
```

6.6 总结:Linux 文件系统

在 Linux 中,文件被组织在目录中。目录本身相互间分层连接,构成一种树形结构。每一个目录中都包含了文件和其它目录。在分层结构的顶端是根目录,由它分支出系统目录和用户的 `home` 目录。系统目录中包含着维持 Linux 系统运行的应用程序,`home` 目录是用户的登录目录。

目录的网状结构构成了一个由高层目录到底层目录的路径,反之亦然。由根

目录到一个特定目录之间的目录集就是目录的路径名。实际上,每一个文件都由一个路径名,由根目录开始到文件所在目录。路径名和文件名一同构成了文件的绝对路径名。

你可以轻松地管理目录——创建新目录,删除旧目录,由一个目录更换到另一个目录。在一个目录中你还可以创建任何深层的子目录。当然录入系统后,你总是处于一个缺省的目录中。这一缺省目录叫作你的工作目录,你创建的任何文件都会被放入这一目录中,除非另加指定。

你可以用 `cd` 命令来改变缺省目录。在此意义上,你可以从一个目录移到另一个目录。当你首次登录入系统时,你缺省的目录是你的 `home` 目录。

你可在目录之间或目录本身上进行文件操作。你可将文件由一个目录拷贝,移动到另一个目录。你甚至可以移动,拷贝整个目录,你也可以给文件创建连接。连接是文件的另一个名字。你可以在一个目录中有一个可引用另一目录中文件的连接。

表 6-1 file 和 od 命令

命令

file

-f filename

od

-c

-d

执行

检查一个文件的头几行以确定其分类

将要进行检查的文件名由一个文件中读出

将文件的内容以八进制、字符、十进制或十六

进制的形式逐一打击;八进制是缺省格式

将 byte 值以字符形式输出;非打印字符有相应的字符表示形式

将 byte 值以十进制形式输出

- x 将 byte 值以十六进制形式输出
- o 将 byte 值以八进制形式输出

表 6-2Linux 中标准系统目录

目录	功能
/	文件系统结构的开始—叫作根目录
/home	包含着用户的 home 目录
/bin	包含着所有标准的命令和应用程序
/usr	包含着系统使用的文件和命令；这一目录被分为若干子目录
/usr/bin	包含着面向用户的命令和应用程序
/usr/sbin	包含系统管理命令
/usr/lib	包含编程语言的序
/usr/doc	包含 Linux 的文档
/usr/man	包含在线的手工 man 文件
/usr/spooc	包含缓冲文件,如一些打印作业和网络传输
/sbin	包含系统重启动的系统管理命令
/var	包含一些如信箱文件之类的可变文件
/dev	包含与诸如终端和打印机的文件接口
/etc	包含系统设置和其它系统文件

表 6-3 文件列表、显示和打印

命令选项	执行
ls	列出文件和目录名 \$ls
cat	这一过滤器被用于文件显示,它可将文件名作为参数。它将文件内容直接送入标准输出,后者在缺省情况下指向屏幕
more	这一应用将文件逐屏显示。它可将文件名作为参数。它将文件的内容输出到屏幕,一次只显示一屏内容 \$more filenames
+num	从文件的第 num 页开始显示
numf	跳过 num 数的屏幕
numb	返回 num 数的屏幕
d	显示半屏
h	列出所有 more 命令
q	退出 more 命令
lpr	将文件送往在线打印机打印;参数可以是一个文件的列表
-p printername	选择一个特定的打印机
lpq	列出打印任务的队列
lprm	将一个打印任务自打印队列中删除

表 6-4 目录命令

命令	执行
mkdir	创建一个目录 \$ mkdir reports
rmdir	删除一个目录 \$ rmdir letters
ls -F	用斜杠列出目录名 \$ls -F today reports/letters/
ls -R	列出工作目录和子目录
cd directory name	进入特定的目录,并将之作为工作目录; 没有指定目录名的 cd 命令将回到 home 目录中 \$cd reports \$cd
pwd	列出一个工作目录的路径名
\$pwd	
directory name/filename	/home/chris/reports 在路径名中的斜杠用于分隔每个目录名。对文件的路径名而言,斜杠将文件名同前面的目录名分开 \$ cd/home/Chris/reports \$ cat/home/Chris/reports/mydata

引用工作目录。你可将之作为参数或路径名的一部分

```
$ ls.
```

```
$ mv ../aleina.
```

~ /pathname

波浪线是一个代表 home 目录路径名的特殊字符。当使用文件或目录的绝对路径名时它是很有用的

```
$ cp monday ~ / today
```

```
$ mv tuesday ~ / weather
```

表 6-5 文件操作

命令

执行

cp filename filename

拷贝文件。CP 有两个参数:原文件名和新的拷贝文件名。你可以使用文件的路径名以进行跨目录的拷贝

```
$cp today reports/monday
```

cp -r dirname dirname

将一个子目录由一个目录拷贝到另一个目录中。被拷贝的子目录将包含它所有的子目录

```
$cp -r letters/thankyou oldletters
```

mv filename filename

移动或将文件更名。mv 有两个参数:第一个是要移动的文件第二个是新文件名或目录的路径名。如果它是一个目录名,这一文件在字面上被移到那

mv dirname dirname	一目录中,并且将文件的路径名加以改变 \$mv today/home/Chris/reports
ln filename filename	移动一个目录,在这种情况下,两个参数都是目录 \$mv letters/thankyou oldletters 为文件创建一个名为连接的增加名。可在一个 目录中创建另一个目录中文件的连接 \$ln today reports/monday
rm filenames	删除文件。可以有任意数目的文件作参数。字 面上删除了文件的连接。如果一个文件有多个 连接,你应当连接全部删除以最终删除一个文件 \$rm today weather weekend

表 6-6 find 命令
执行

命令或选项	
find	根据一个标准搜索目录以查找文件。这一命 令有许多参数确定搜索标准和要采取的行为
-have pattern	查找名字中具有 pattern 的文件
-group name	查找属于那一组名的文件
-size name	查找具有 num 块大小的文件。如果在 num 之后 加入 C,那将以 byte 为单位进行查找
-mtime num	查找 num 天之前修改过的文件

-newer pattern

-print

-type filetype

b

c

d

f

p

l

查找与 pattern 匹配文件之后的修改过的文件
将查找结果输出到标准输出。结果通常是文
件名的列表,包括它们的全路径名

查找特定类型的文件

块设备文件

字符设备文件

目录文件

常规文件

指定的管道 (fifo)

符号连接

第 7 章 文件管理操作

Linux 在文件管理和目录管理上有几个特征,你可以找到文件的详细信息,如它们最近何时被修改过和它们拥有的连接数。你可以控制对文件的访问。Linux 中的每一个文件都有许可,它确定谁有权对之访问,以及何种访问。你可以允许其它用户访问指定的文件或者限制为只由你自己访问。

文件存在于诸如 CD-ROMS 或软盘等物理设备上。每一设备上的文件都被组织成为一个文件系统。要访问一个设备上的文件,你便将它的文件系统连到一个特别的目录中。这被称作装卸文件系统。例如,要访问软盘上的文件,如何可以访问 CD-ROMS,软盘和硬盘分区。你甚至可以访问 MS-DOS 的硬盘分区和软盘,以及一个远程服务器上的文件系统。

你也可以将你的文件备份到一个档案中,以备以后读取,你也可以把它们组合成为一个档案文件已通过网络传到另一系统中。你也可以将文件压缩以进行更有效的传送或只是为了占用更少的空间。档案和压缩是在线软件包经常使用的方法。你可以下载一个压缩并存档的软件包,然后将之解压并将档案扩展开。然后你便可以把新的软件包安装到你的系统上了。这是一种取得 Linux 软件的常用方法。

7.1 显示文件信息:ls -l

如图 7-1 所示,ls -l 命令显示文件的详细信息。首先显示的是许可,其次是连接数,文件的所有者,用户所属的组名,以 byte 度量的文件大小,文件最后被修改的日期和时间,以及文件的名字。组名指的是有组许可的组群。在图 7-1 中,文件 mydata 的类型是一个普通文件。只有一个连接,即文件没有其它的名字。所有者的名字是 Chris,与登录名相同,组名是 Weather 可能还有些属于 Weather 组的其它的用户。文件的大小是 207bytes,它最后一个被修改是在 February 20,在上午 11:55 分。文件名是 mydata。

```
File type  Permissions  Number of links  Owner name  Group name  Size of file in bytes  Date and time last modified  File name
  ↓          ↓          ↓          ↓          ↓          ↓          ↓          ↓
-rw-r--r--  1  chris  weather  207  Feb  20  11:55  mydata

$ ls -l mydata
-rw-r--r--  1  chris  weather  207  Feb  20  11:55  mydata
```

图 7-1 用 ls 和 -l 选项显示的文件信息

如果你想显示一个目录中有文件的详细信息,只要用 ls-l 命令即可,无须加任

何参数。

```
$ls-l
```

```
-rw-r--r--1Chrisweather207Feb 2011:55mydata  
-rw-r--r--1Chrisweather568Feb 1410:30otday  
-rw-r--r--1Chrisweather308Feb 1712:40monday
```

7.2 文件和目录许可:Chmod

Linux 系统中的每个文件和目录都有一许可来确定谁可以通过何种方式对文件和系统加以访问,你可以设置许可将访问限制在三种方法之内:你可以只限由自己进行访问;你可以允许一个预先指定的组中的用户访问权,或者你可以允许系统上的任何一个人访问,你也可以控制一个文件和目录如何被访问。一个文件和目录可能有读,写和执行许可,当文件被创建时,文件所有者自动拥有读和写权限,以例将文件显示和修改。你可把许可改为你想要的任何组合。一个文件可以有只读许可,禁止任何的更改操作。它可以有执行许可,允许它作为一个程序执行。

有三种不同类型的用户可对目录和文件加以访问:文件所有者,组、或其它人。所有者是文件的创建者。你创建了任何文件,你便是它的拥有者。你也可以允许你的组群有权访问文件的权限。用户被分为组。例如,某一班或一个工程的所有用户被系统管理员分为组。一个用户有可能将文件的高访问权限赋予组中的其它用户。最后,你还可以将文件的访问权赋予系统中所有其它的用户。在这种情况下,系统中每一位用户可能都能访问你的某一文件或目录。在这种意义上,系

统中的任何其它用户都构成了“其它”这一类。

每一类用户都有自己的的一套读、写和执行许可。第一套许可控制用户对自己文件的访问——所有者的访问。第二套控制组对一个用户文件的访问。第三个控制着所有其它用户对用户文件的访问。这三类——所有者,组和其它的三种读、写、执行许可可能成了九种类型的许可。

正如你在上节中所知,ls 命令和 -l 选项列出了文件的详细信息,包括许可。在下一个例子中,最右边的一套字符就是为文件 mydata 设定的许可。

```
$ls-l mydata
```

```
-rw-r--r--l Chris weather 207 Feb 20 11:55 mydata
```

横线代表的是空许可。r 代表的是读许可,w 代表写,执行许可由 x 代表。注意这里共有 10 个位置。第一个字符指定了文件类型。在通常意义上,一个目录可被认为是一个文件。如果第一个字符是横线,那么列出的是一个文件。如果是 d,那么列出的是一个目录。

```

                                Permissions
                                Owner Group Other users
                                rwx  rwx  rwx
                                rw-  rw-  r--
File type Permissions
↓
-rw-r--r-- 1 chris weather 207 Feb 20 11:55 mydata

$ ls -l mydata
-rw-r--r-- 1 chris weather 207 Feb 20 11:55 mydata

```

图 7-2 属主,组和其它文件许可:r 表示读许可,w 表示写,x 表示执行,横线表示无权限后面的九个字符是按不同的用户类型安排的。第一套三个字符是用户自己的文件许可。第二组三个字符是组的文件许可。最后三个字符是其它用户的文件许可。在图 7-2 中,文件 mydata 的所有者有读和写许可,组类只有读许可,其它用户类也只有读许可。这意味着,尽管组中的其它用户或系统中任意其它用户都可以进行读操作,只有所有者可对文件修改。

你可以使用 Chmod 命令来创建不同的许可设置。Chmod 的参数是两个列表:许可变更和文件名。你可用两种不同方法指定许可列表。一种方法是使用许可符号,它被叫作符号方法(symbol method)。另一种方法利用被称作二进制屏蔽,它被叫作绝对或相对方法。两种方法中,符号法是更简单,因此会首先介绍它。在本章最后的表 7-1。列出了 Chmod 命令的选项。

7.2.1 设置许可:许可符号

正如你所预料的,设置许可的符号方法使用字符 `r`、`w` 和 `x` 来各自代表读,写和执行。每一项许可都可以进行添加或删除。加入一个许可的符号是加号,十、删除许可的符号是减号,一。在下一个例子中,`Chmod` 命令加入文件 `mydata` 的执行许可并删除了写许可。读许可没有变动。

```
$ chmod +x-w mydata
```

还有用于标识每种用户类型的许可符号。所有者、组和其它用户分别由字符 `u`、`g` 和 `o` 代表。注意所有者这一类是由 `u` 代表,可认为它是代表用户。类别字符是放在写、读和执行许可前面的。如果未使用类别字符,那就认为是所有的类,设定的许可适用于用户所有者,组和其它用户。下例及图 7-3 中,第一个 `Chmod` 命令将组许可设为读和写。第二个 `Chmod` 命令将其它用户许可设为读。注意在许可和类别之间没有空格、许可列表也是一个长短语,中间没有空格。

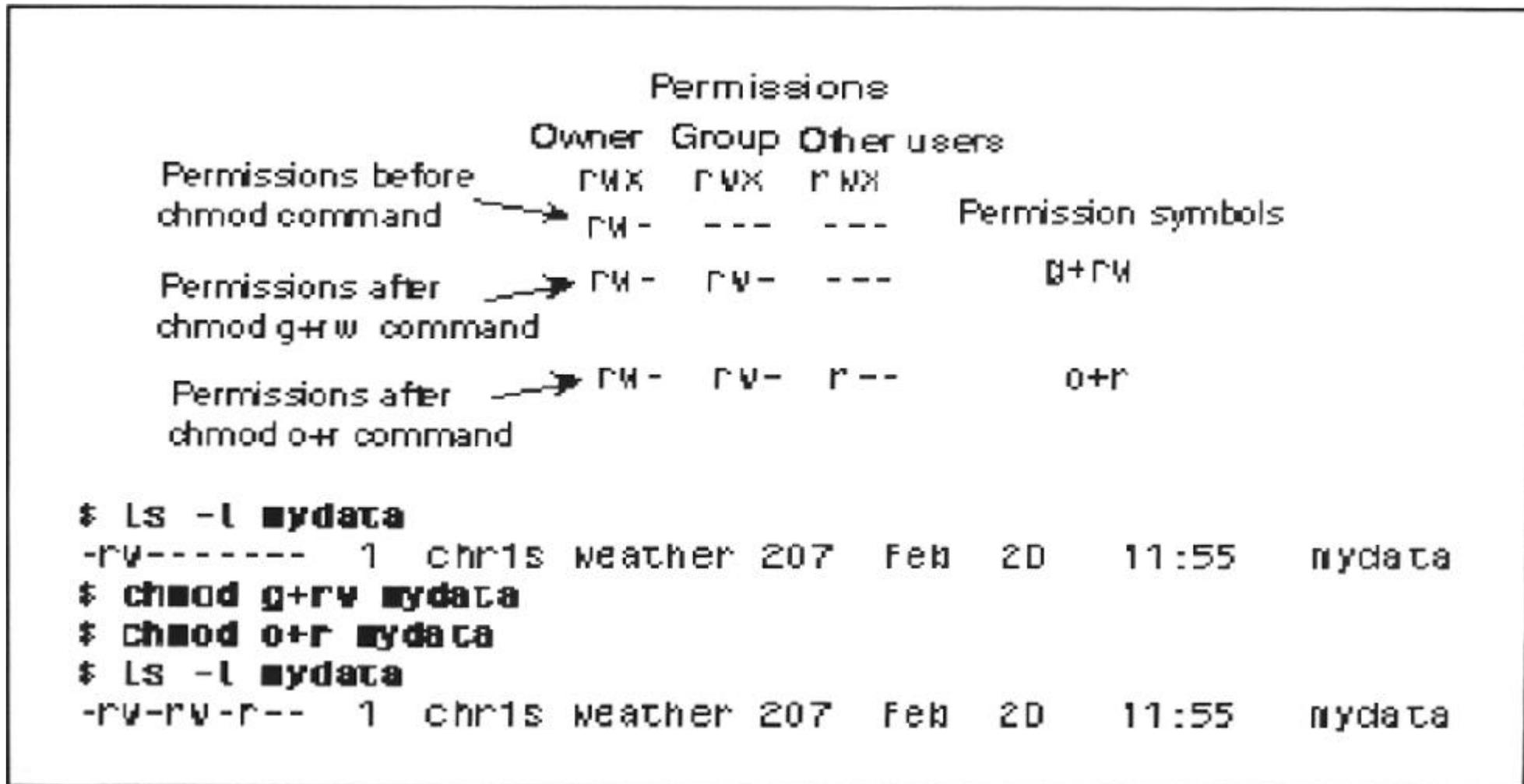


图 7-3 用许可符号来设置权限

```
$ chmod g+rw mydata
```

```
$ chmod o+r mydata
```

用户可以同添加许可一样删除许可。在下一个例子中,其它用户被设置了读

许可,而写必执行许可被删除了。

```
$ chmod o+r-wx mydata
```

还有一个类别符号,a,它代表所有的类别。符号 a 是缺省的,在下一个例子中,两个命令是等价的。显式地利用了符号 a 来指定所有类型的用户:其它,组和用户,并为它们设置了读许可。

```
$ chmod a+r mydata
```

```
$ chmod +r mydata
```

一个最常用的许可操作是设定一个文件的执行许可。这通常是在 Shell 程序文件的情况下执行的,这将在第 8 和 16 章讨论。执行许可表明一个文件中包含着可执行的指令并可直接由系统运行。在下一个例子中,文件 lsc 被设定了可执行许可后被执行。

```
$ Chmod u+x lsc
```

```
$ lsc
```

```
main,c lib,c
```

```
$
```

除读/写/执行许可外,你也可以设置可执行文件所有者属性。正常情况下,运行程序的用户在程序运行期间拥有该程序,尽管程序文件本身可能是另一用户所拥有的。用户 ID 许可允许一个程序的所有者总是拥的这一程序,即使其它用户正在运行这一程序。例如,在系统中的大部分软件都是由根用户的所拥有而被普通用户所运行。在这种情况下,根用户应在普通用户运行程序时仍然对程序有所有权,以便程序有修改根用户所有文件修改的许可。组 ID 许可是为组设定执行上述功能的许可。一个组所拥有的程序在另一个组的用户运行程序时仍然有所

有权。这样程序便可修改组的文件。

用 s 选项可以加入用户 ID 和组 ID 对文件许可。下一例子将用户 ID 许可赋予了 pppd 程序,它是由根用户所拥有的。当普通用户运行 pppd 时,根用户仍保留所有仅,使 pppd 程序可修改根用户文件。

```
# Chmod +s /usr/sbin/pppd
```

用户 ID 和组 ID 许可在所有者和组段的执行位置上以 S 代表。用户 ID 和组 ID 是执行许可,x 的一个变形。读、写和用户 ID 许可是 rws 而不是 rwx。

```
# ls -l /usr/sbin/pppd
```

```
-rwsr-sr-x      1      root      root  84604          Aag   14      1996
```

```
/usr/sbin/pppd
```

另一个特殊许可允许程序的有效利用。粘着置位指示系统在一个程序执行完后仍将之留在内存中。这对那些用户频繁使用的小程序是很有用的。粘着置位许可是 t。粘着置位在其它用户许可的执行部分以 t 表示。一个有读、执行和粘着置位的程序的许可将显示为 r-t。

```
# Chmod +t /usr/X11 R6/bin/xtetris
```

```
# ls -l /usr/X11 R6/bin/xtetris
```

```
-rwxr-xr-t     1     root     oot   27482   Nov19   1996
```

```
/usr/X11 R6/bin/xtetris
```

7.2.2 绝对许可:二进制屏蔽

许多用户发现利用绝对方法更方便,而不是使用许可符号。绝对方法一次可

变更,所有的许可,而不是一项加以指定。它使用了一种二进制屏蔽来引用每类中的所用许可。三类,每类有三种许可,构成了总数为八的二进制数,每进制数都以八为基础结构的,将变成二进制数时,每一个八以内的数都成为三位二进制数。二进制数是 0 或 1 的集合。一个数字中的三个八进制数可变为三套二进制数,加起来总共是九,即文件许可的确切数目。

你可以利用八进制数作屏蔽来设置不同的文件许可。每一个八进制数适用于用户类型的一种。你可以认为这些数与从左到右排列的许可类型相匹配,最左端是所有者类型。第一个八进制数适用于所有者类型,第二个适用于组,第三个适用于其它类型。

你所选择的实际的八进制数将确定每类的读、写和执行属性。这时,你需要知道八进制数字如何转换成为相应的二进制数字。下面的表中列出了八进制数 0~7 如何转换成为有三个数字的相应二进制数字。你可以认为八进制数首先转换成为二进制数,然后二进制数中的三个数字再分别设置读、写和执行属性。每一个二进制的数字都与一个相应的属性相匹配,也是从左到右。如果二进制数字是 0,那么许可被禁止。如果二进制数为 1,那么许可被允许。第一个数字设置读的许可为允许或禁止,第二个设置写的许可,第三个设置执行许可,例如,八进制数 6 转换为二进制数 110,这将把读、写许可设为允许,而将执行许可标止。

Octal	Binary
0	000
1	001
2	010
3	011

```

4          100
5          101
6          110
7          111

```

```

                                Permissions
                                Owner  Group  Other users
Octal/Binary Translation
Octal digits  5    4    4
Binary digits 101  100  100
                                rwx  rwx  rwx
                                101  100  100
                                r-x  r--  r--

$ chmod 544 mydata

$ ls -l mydata
-r-xr--r-- 1 chris weather 207 Feb 20 11:55 mydata

```

图 7-4 用八进制来设置权限

当使用二进制屏蔽时,你需要为三个种类的三种许可确定三个数字。这使得它比符号许可显得要死板。要设置所有者对文件 mydata 的执行许可为允许,及写许可禁止,并且保留读许可,你要使用二进制数 5(101),同时还要指定组和其它用户记问的数字。如果这几类要保留读许可,你需要为每类指定 4(100),这样你有三个八个八进制数,544,转换为二进制数为,101 100 100。在图 7-4 中,为文件 mydata 设置了这些许可。

```
$ Chmod 544 mydata
```

无须十分费力地完成由八进制到二进制的转换以确定你应使用哪个数字,有一种十分简单的方法。读、写和执行许可分别与数字 4,2,1 相联系。(见下表)。要确定为某类应使用哪个数字,只需将 4(读)和 2(写)加起来即为 6。这就是你所需的八进制数。要给一类读和执行许可,只要将 4(读)和 1(执行)相加得 5。要想设置所有的许可,你要将 4(读)2(写)和 1(执行)相加得 7,它等于二进制为 111。下一个例子用这种方法来计算上例中所使用的许可。所有者有读和执行许可,而组和其它用户只有读许可。

Owner	rx		4 + 1 = 5	
group	r	4	4	
others	r	4	4	
			544	101 100 100
\$	Chmod	544	mydata	

例如,要给你组中的成员读和写的许可,而系统中其它用户设有许可,你需要给组数字 6(110),给其它用户 0,给所有者 6 保存了所有者的读写许可。你可以通过把 4(读)和 2(写)相加得 6 来计算读,写许可。即然,你不想,任何其它用户拥有许可,你可以给它们 0,你现在得到数字 660,等于二进制数 110 110 000,注意八进制数将利用二进制的 000 来把所有的许可设为禁止。

\$	Chmod	660	mydata
	许可数字	二进制数	
r	4	100	
w	2	010	
x	1	001	

二进制屏蔽的最通常的用法是用来设置执行许可。如第 8 章中将要介绍的,你可以创建包含 Linux 命令的文件。这些文件都称为 Shell scripts,要执行 Shell scripts 中的命令,你必须首先指明这一文件是可执行的——它包含着系统可以执行的命令。可以有几种方法完成上述工作,其中之一是设置 Shell script 文件的可执行许可。假定你刚完成了一个 Shell scripts 文件并给予它执行许可以例将之运行,同时你想保留读和写许可,包括执行许可(你也可以将 4 读,2-写,和 1-执行相加得 7)。给组和其它用记许可赋 0 将禁止他们的记问。你得到数字 700,与之相等的二进制数是 111 000 000,在下一个例子中,文件 myprog 的所有者的许可被设置的包含执行许可。

```
$ Chmod 700 myprog
```

如果你希望其它的用户可以执行并读文件,但不能修改,你可以用数字 5(101)来设置读和执行许可为允许,并禁止写许可。在这种情况下,你要使用八进制数 755,即二进制的 11 101 101。

```
$ Chmod 755 myprog
```

对于所有权和粘着置位许可,你可在八进制数的前面再加一个八进制数。用户 ID 许可的八进制许可是 4(100); 组 ID 为 2(010); 对于粘着置位是 1(001)。在下一个例子中设置了程序 pppd 的用户 ID 许可,同时设定了所有者、组和其它用户的读、写许可。

```
# Chmod 4555 /usr/sbin/pppd
```

下面的例子设置了程序 xtetris 的粘着置位。

```
# Chmod 1755 /usr/X11R6/bin/xtetris
```

下一个例子将设置程序 XMan 的粘着置位和用户 ID 许可。许可 5755

的二进制数为 101

```
11 101 101。
```

```
# chmod 5755 /usr/X11R/bin/xman
```

```
# ls -l /usr/X11R6/bin/xman
```

```
-rwsr-xr-t 1 root root 44364 Mar 26 04:28
```

```
/usr/X11R6/bin/xman
```

7.2.3 目录许可

你也可以设置目录许可。一个目录的读许可可以使用户显示目录中的文件。执行许可使用户可进入该目录。写许可允许用户在该目录中创建或删除自己的文件。如果你允许其它用户对一个目录有写许可,用户便可在目录中添加自己的文件。当你创建一个目录时,它主动给所有者读、写和执行许可。你可例出目录中的文件,进入该目录,可在其中创建文件。

同文件相同,目录也有所有者,组和其它用户的一套许可。通常,你可能希望用户可以进入你的一个目录并对目录中文件进行列表,但不希望他们在目录中添加文件。这种情况下,你应设置目录的读和执行许可,而禁止写许可。这会使用户可以进入目录并对文件列表,但不能的创建新文件或将自己的文件拷贝入目录。下一个例子设置了目录 `thankyou` 的组的读和执行许可,但没有写许可。组中的成员可以进入 `thankyou` 目录并对文件列表,但不能创建新文件。

```
# Chmod g+rx-w letters/thankyou
```

如同文件一样,你也可利用八进制数来设置目录许可。要设置与上例中相同

的许可,你要使用八进制数 750,即二进制的 11 101 000。

```
$ Chmod 750 letters/thankyou
```

正如你所知的,ls 命令加上 -l 选项可例出一个目录中所有的文件。要只列出目录的消息,可加入 d 修改符。在下一个例子中,命令 ls -ld 列出了目录 thankyou 的信息。注意在许可列表中的第一个字符是 d,代表它是一个目录。

```
$ ls -cd thankyou
```

```
drwxr-x--- 2 chris 512 Feb10 04:30 thankyou
```

如果你有一个希望其它用户访问的文件,你不仅要为文件设置许可,而且还要保证对文件所在的目录。这一点也适用于一个目录的父目录。尽管一个目录允许其它用户访问,但其父目录未设定访问许可,那么这一目录仍是不可访问的。在这一点上,你必须对目录树多加注意。要提供一个目录的访问权,目录树中在此目录上面的所有父目录也必须是可访问的。

在图 7-5 中,用户树中在此目录上面的所有父目录也必须是可访问的。需要进入 Chris 和 Letters 目录才能进行访问。将(每个目录)其它用户类的执行许可设为允许即可做到这一点。注意只须设定执行许可,而设有读和写许可。其它用户不允许在目录 Chris 和 letters 中进行文件列表和增加文件。他们只能进入这一目录并引用其子目录。

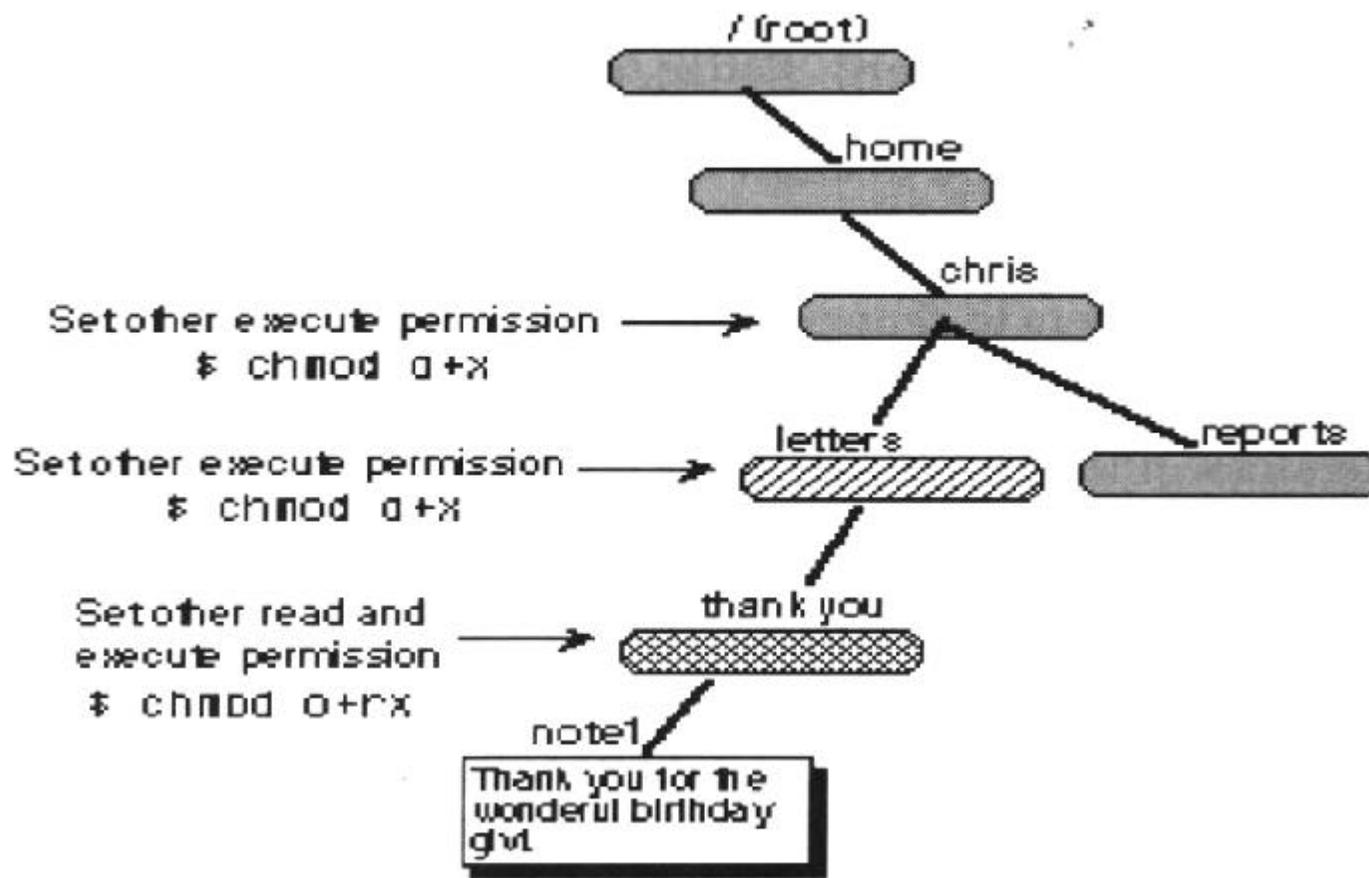


图 7-5 要打开一个其它用户可以访问的目录,你必须也让该目录的所有父目录可被访问

7.2.4 改变一个文件的所有者或组:Chown 和 Chgrp

尽管其它用户有可能对文件访问,但只有文件所有者才能修改许可。如果,你

想赋予其它某个用户对你某个文件许可的控制,你需要将文件的所有者由你改为那个用户。Chown 命令将一个文件的控制权移交给另一个用户。这一命令以另一个用户名作为第一个参数。在用户名后,你可列出想移交的文件名。在下一个例子中,用户将文件 mydata 的控制权移交给 Robert.

```
$ Chown robert mydata
```

```
$ ls -l mydata
```

```
-rw-r--r-- 1 robert weather 207 Feb 20 11:15 mydata
```

只要你愿意,你也可以用 Chgrp 命令来改变一个文件的组。Chgrp 把文件新的组名作为第一个参数。在新组名后,你可以列出想变更到那一组去的文件名。在下一个例子中,用户将文件 today 和 welkend 的组改到 forecast 组中。ls -l 命令反映出了组的改变。

```
$ Chgrp forecast today welkend
```

```
$ ls -l
```

```
-rw-r--r-- 1 Chris weather 207 Feb 20 11:55 mydata
```

```
-rw-r--r-- 1 Chris forecast 568 Feb 14 10:30 today
```

```
-rw-r--r-- 1 Chris forecast 308 Feb 17 12:40 welkend
```

7.3 文件系统:安装和卸载

尽管你的 Linux 系统中所有的文件都被连接入一个总的目录树中,文件本身位于一些诸如硬盘和 CD-ROM 之类的存贮设备中。在某一存贮设备上的文件被

组织成为一种文件系统。你的 Linux 目录树可能包含几个文件系统,每一个在不同的存贮设备上。文件本身被组织成为根目录开始的无缝隙的目录树。尽管你的根目录可能位于硬盘分区的文件系统中,但在 CD-ROM 上的文件仍然有一条直接的路径名。

在一个文件系统中的文件与你的目录树是相分离的,除非你专门将文件联入目录树。一个文件系统将文件组织成自己的目录树。你可以把它想象成为必须被连入主目录树的一颗子目录树。例如,一个有 Linux 文件的软盘有自己的目录树。你需要把这棵子树与你硬盘分区上的主目录树相连。在它们相连之前,你是无法访问软盘上的文件的。

将一种存贮设备上的文件系统连接到主目录树上被称为安装设备。这是由 mount 命令来完成的。要访问 CD-ROM 上的文件,你首先要安装 CD-ROM。安装操作将把存贮设备的目录树连接到你指定的目录中去。它是一种系统管理的任务,不能由普通的用户来完成。要安装一个文件系统,一定要作为根用户登录。表 7-2 列出了 mount 命令的不同选项。

mount 命令有两个参数:Linux 系统要访问文件系统的存贮设备,新的文件系统将要被连入的文件结构中的目录。mount 点是你希望存贮设备上文件连入系统的主目录中的目录。device 是将你的系统与硬件设备相连的特殊的设备文件。mount 命令的语法如下:

```
# mount device mountpoint
```

设备文件目录 /dev 中,它们通常有一些以设备数结尾的缩略名。例如,fd0 可能指联入你的系统的第一个软驱。在 PC 操作 Linux 系统中,硬盘分区有一个 hd 的前缀,后面是一个字母表中字符以标明硬盘,最后是分区号。例如,hda2 指第一

硬盘中的第二个分区。在多数情况下,你可以利用有前缀的 man 命令来获取与设备有关的更多的详细信息。例如,mansd 显示了 SCSI 设备的 man 页数。

一个文件系统要是可访问的,它首先要被安装。甚至你的硬盘分区上的文件系统也要由 mount 命令加以安装。然而,当你装 Linux 系统并在硬盘上开辟 Linux 分区时,你的系统一旦开始便被设置为自动安装你的主文件系统。软驱和光驱则需要显式地安装。下例将第一个软盘驱动设备的软盘(fd0)安装到目录/mydir 中。

```
# mount /dev/fd0/mydir
```

在你关闭系统之前,所有你安装的文件系统都必须被卸载。你的主文件系统是主动卸载的。然而,如果你想用一个文件系统来代替一个已安装的文件系统,你必须首先将已安装的文件系统卸载。假定你已安装了一个软盘,现在你想把它取出再放入一个新的,你必须在放入新盘之前首先将原有的卸载,然后再安装新的软盘。用 unmount 命令可以卸载一个文件系统。umount 可将设备名或设备安装的目录作为参数。语法如下:

```
# unmount device-or-mountpoint
```

下一个例子卸载了安装在目录/mydir 下的软盘:

```
# unmount /dev/fdo
```

利用将设备安装在目录/mydir 中的例子,你也可以利用这一目录来卸载文件系统。

```
# unmount /mydir
```

umount 命令有一个重要的限制。你永远不能卸载一个你正在其中工作的文件系统。如果你进入了一个文件系统中的目录,然后你想卸载这一文件系统,你会接到一个文件系统忙的错误信息。例如,你将 Open Linux 的 CD-ROM

装在目录 /mnt/cdrow 下,然后你进入了 /mntcdrom 目录。

如果你想更换 CD。这将会失败,因为你现在正位于它所安装的目录中。在你卸载 CD-ROM 之前你必须离开这一目录。

```
# mount /dev/hdc /mnt/cdrom
# cd /mnt/cdrom
# unmount /mnt/cdrom
unmount:/dev/hdd:device is busy
# cd /root
# unmount /mnt/cdrom (239)
```

在每个存贮设备上的文件系统都被格式化过以占据一定数量空间。例如,你可将你的硬盘分区格式化为占 300m 空间。在这一文件系统中安装或创建的文件将占据部分空间,剩余空间可留给新的文件和目录。要查出在一个文件系统中还有多少剩余空间,你可以使用 df 命令。它按设备名会例出你所有的文件系统,它们的存贮空间大小,已经利用的空间的百分比,以及它们所安装的位置。df 命令,而不是 fdisk 命令,是一种列出分区表的安全的方法。

\$ df

File sytem	1024-blocks	used	available
capacity	Mounted on	/dev/hda3	297635
169499	112764	60%	/
/dev/hda1	205380	182320	23060
89%	/mnt/dos	/dev/hdc	637986
637986	0	100%	/mnt/cdrom

利用 df 命令,你也可知某一目属于哪个文件系统,只需输入 df 和目录名,

或 df,来代表当前目录。

```
$ df .
```

```
File system 1024-blocks used avai/able capacity Mounted on  
/dev/hda3 297635 169499 112764 60% 1
```

要确认某一文件系统运行正常,你可以使用 fsck 命令进行检查。输入 fsck 和代表文件系统的设备名。表 7-3 列出了 fsck 的选项。下面的例子检查了软驱中的磁盘和主硬盘:

```
# fsck /dev/fdo  
# fsck /dev/hda1
```

7.3.1 安装并格式化软盘

要访问一张软盘上的文件,你首先需要把软盘安装到你的 Linux 系统中。你的软驱设备名是 fd0,目录 /dev./dev/fd0 代表软驱。注意 fd 之后的 0,如果你有多个软驱,它们会分别用 fd1,fd2 等口来代表。你可以把它安装到任何目录。但 Open Linux 安装已经为软盘创建了一个范例的目录,/mnt/floppy。下一个例子把软驱中的软盘安装到 /mnt/floppy 目录:

```
# mount /dev/fdo /mnt/floppy
```

记住你正在安装一张软盘,而不是软驱。你不能只是将一张软盘取走,再放入一张新的。mount 命令已将这些文件联入你的主目录树中,你的系统认为在软驱的软盘中可以找到这些文件。如果你将软盘取出后又放入另外一张,那在对之访问时会出现错误。

要更换磁盘,你首先必须将软驱中的磁盘卸载,在放新盘之后,你必须明确地安装新磁盘。可用 `umount` 命令来完成这一任务。注意在 `umount` 命令中并没有 `n`。

```
# umount /dev/fdo
```

在卸载操作中,你可以指定其安装的目录或 `/dev/fdo` 设备:

```
# umount /mnt/floppy
```

你现在可以取走软盘,放入一张新盘,然后进行安装了。

```
# mount /mat/floppy
```

当你关闭系统时,你所安装的磁盘将自动被卸载,无须你将之明确卸载。要将软盘格式化,使用 `mkfs` 命令。它在盘上创建了一个 Linux 文件系统。一定要确保用 `-t ext2` 选项来指定 `ext2` 类型的文件系统。一旦格式化,你命名可以安装这一文件系统。`mkfs` 将设备名和盘上存贮块的数目作为参数。一个块中有 1000byte,1400 格式化了一张 1.44MB 的软盘。你不用先安装空盘,只要批发它放在软驱中然后输入 `mkfs` 命令和它的参数就可以了。下一个例子格式化了一张 1.44MB 的软盘。

```
# mkfs -t ext2 /dev/fdo 1400
```

7.3.2 安装 CD-ROM

你也可以把 CD-ROM 盘片 `mount` 到你的 Linux 系统。在 Open Linux 系统中,目录 `/mnt/cdrom` 是专为 CD-ROM 文件系统保留的。在文件 `/etc/fstab` 中你会看到它的一个入口。要安装 CD-ROM,你只要输入 `mount` 命令和目录名

/mnt/cdrom 即可。你无须指定设备名。一旦安装,你可以通过/mnt/cdrom 目录对 CD-ROM 进行访问。

```
# mount /mnt/cdrom
```

如同软盘一样,要记住你安装的是一张光盘,而不是光驱。你不能仅仅将光盘移走再放入另外一张。mount 命令已将这些文件联入主目录树,你的系统认为在 CD-ROM 驱动器的光盘中会找到这些文件。

要更换盘片,你需要先用 umount 命令卸载在 CD-ROM 驱动器中的盘片。在你的命令实施之前,你的 CD-ROM 不会打开。在放入新的光盘之后,你可以显式地安装新的盘片。

```
# umount /mnt/cdrom
```

你现在可以取走光盘并放入另外一张。然后执行 mount 命令以进行安装。

```
# mount /mnt/cdrom
```

如果你想把 CD-ROM 安装到另一个目录中,你必须在 mount 命令中加入设备名。下一个例子把光驱中的光盘安装到/mydir 目录中。本例中 CD-ROM 的设备名是/dev/hdc。

```
# mount /dev/hdc /mydir
```

光驱的设备名是随着光驱类型而变的。IDE CD-ROM 的设备名有与 IDE 硬盘分区相同的前缀 hd。其后的一个字符标识了光驱并将它与其它的 IDE 设备区分开来。例如,一个接到第二个 IDE 口的 CD-ROM 的名字是 hdc。一个作为 slave 接到第二个 IDE 口的 IDE CD-ROM 的名字可能是 hdd。实际的名字将在 CD-ROM 安装的时候确定,正如同安装 Linux 系统时一样。SCSI 的光驱有自己的单独的命名法。它以 sd 开头代表 SCSI 驱动器,其后跟一个标识的字符。例如,一

个 SCSI CD-ROM 的名字可能是 sdb 或 sda。当安装系统时,CD-ROM 的名字便已确定。你可以查看 /etc/fstab 文件或使用根目录用户桌面中的 fstool 来进行确认。

要更换光盘,你应先卸载光驱中的 CD-ROM。然后,放入新盘片,再对它进行明确地安装。

```
# umount /mydir
```

你现在可以取走 CD-ROM 并放入另一张光盘。然后执行 mount 命令进行安装。

```
# mount /dev/hdc /mydir
```

7.3.3 安装硬盘分区:Linux 和 MS-DOS

你可以用 mount 命令安装 Linux 或 MS-DOS 硬盘分区。然而,更加实用的是利用文件 /etc/fstab 进行自动安装,正始下节第述。你在安装时所创建的 Linux 硬盘分区已经自动装好。要安装一个 Linux 硬盘分区,只要输入 mount 命令和分区的设备名及你想安装的目录名。IDE 硬盘使用 hd 前缀,SCSI 硬盘使用 sd 前缀。一个硬盘驱动器的前缀后面跟着一个标识硬盘的第二个分区,sdb3 代表第二个 SCSI 硬盘上的第三个分区。要查找设备名,你可以用 df 来显示你的硬盘分区或检查 /etc/fstab 文件。下一个例子将 /dev/hda4 上的 Linux 硬盘分区安装到目录 /mnt/mydata。

```
# mount -t ext2 /dev/hda4 /mnt/mydata
```

你也可以安装一个 MS-DOS 分区并直接访问其中的文件。同 Linux 分区一

样,你也要使用 `mount` 命令,但是你要把文件系统的类型定 MS-DOS。需要使用 `-t` 选项和 `msdos` 类型来完成上述任务。在下一个例子中,用户将 `/dev/hda1` 的 MS-DOS 硬盘分区安装到 Linux 文件结构的目录 `/mnt/dos` 中。`/mnt/dos` 是 MS-DOS 文件系统的—个普通的命名,你可以把分区安装到任何一个目录。但要保证你已创建了该目录。

```
# mount -t msdos /dev/hda1 /mnt/dos
```

如果你想从一个新的硬盘或一个现有硬盘上,来安装一个新的分区。你需要用 Linux 的 `fdisk` 或 `cdisk` 来创建这一分区,并用 `mkfs` 命令将之格式化。一旦将之创建并格式化,你就可以把它安装在你的文件系统中了。要开始 `cdisk`,需在命令行上 `fdisk`。它将启动一个交互式的程序,你可以用它创建你的 Linux 分区。使用 Linux 的 `fdisk` 要特别小心。如果你不小心,它可能会删除你的整个硬盘。

`fdisk` 的功用如同在第 2 章中讨论的安装过程中大致相同。命令 `n` 将创建一个新的分区,命令 `t` 允许你将它的类型设为 Linux 类型,83。表 7-5 列出了 `fdisk` 命令。

硬盘分区的命名是以 `hd`(IDE 驱动器)或 `sd`(SCSI 驱动器)开头,后面是一个标识硬盘的字母和一个代表硬盘上分区的数字。它们可以属于任何的操作系统,如 MS-DOS,OS/2,或 Windows NT。Linux。创建的第一个硬盘分区是 `hda2`,如果你添加一个新的 IDE 硬盘,它的第一个分区的名字是 `hdb1`。

一旦创建了分区,你需要将之进行格式化。用 `mkfs` 命令和硬盘分区的名字即可完成这一工作。一个硬盘分区是一个在 `/dev` 目录中有自己的设备名的设备。使用 `mkfs` 命令时,你要指定它的全路径名。例如,第一个硬盘上的第二个分区会有设备名 `/dev/hda5`。下一个例子将那个分区进行了格式化。

```
# mkfs -t ext2 /dev/hda5
```

你现在可以安装你新的硬盘分区,并将之联到文件结构中去。

7.3.4 自动安装文件系统:fstab 文件

当在你的 Linux 系统中加入一个新的硬盘分区时,你很可能希望在开始时对它进行自动安装,在结束时自动卸载。否则,你每次开启或关闭系统的时候,都要对硬盘分区明确地安装和卸载。要让 Linux 自动装卸你新的硬盘分区中的文件系统,你只须将它的名字加到 fstab 文件中,你可以直接仔细地编辑 fstab 文件输入一个新的入口来完成这一工作,你也可以用下一节中描述的 fstool 命令。fstab 文件在目录 /etc 中。它列出了用 mount 命令和 -a 选项安装的文件系统。这一 mount -a 命令在 /etc/rcd/rcboot 文件中。这一命令进行了系统初始化操作。每次你启动系统时它都被执行。当你关闭系统的时候,umount -a 命令被执行以卸载 fstab 文件中所列的全部文件系统。umount -a 命令可以在 /etc/rcd/init/balt 文件中找到,它当中包含着当你关闭系统时要执行的命令。这样,你在 /etc/fstab 文件中指定的文件系统将在系统启动时自动安装,当系统关闭时被自动卸载。其它如 Redhat 的系统可能会把 umount -a 命令放在名为 /etc/rcd/rcsysinit 的文件中。

在 fstab 文件中的一个入口包括几个域,每个域之间用空格或 tab 隔开。第一个域是要安装的文件系统的名字。它通常以 /dev 开头,如 /dev/hda3 代表第三个硬盘分区。第二个域是你希望将设备上的文件系统安装的文件结构中的目录。第三个域是要安装的文件系统的类型。表 7-6 列出了你所有可以安装的文件系统类型。一个标准的 Linux 硬盘分区的类型是 ext。下一个例子表示了一个

主 Linux 硬盘分区的入口。它被安装在根目录 / 中,文件类型是 ext2。

```
/dev/hda3    ext2    defaults    0    /
```

在文件系统类型之后的还列出了安装文件系统的不同选项。你可以输入 default 来指定一套缺省的选项。你可以一个个列出特定的选项,选项之间用逗号分隔(没有空格)。缺省选项指一设备为读/写,非同步、块、普通用户无法进行安装,程序可在上面执行。与之相反,一个 CD-ROM 只有有限的几个选项可供选择,ro 和 noauto,ro 指定它是只读的,noauto 指定它不是自动安装的。noauto 选项是同 CD-ROM 和软驱一同使用的,那么它们就不会被自动安装,既然你不知道在启动的时候在驱动器中是否有东西。同时,CD-ROM 和软驱的入口指定了当你决定进行安装时要安装的目录。表 7-7 列出了安装一个文件系统的选项。紧跟着是一个 CD-ROM 和软驱入口的例子。注意:CD-ROM 文件系统的类型同硬盘分区的文件系统类型 Lso9660 是不同的。软驱同硬盘分区一样有相同的缺省选项。

```
/dev/fdo    /mnt/floppy    ext2    defaults,noauto    0    0
```

```
/dev/hdc    /mnt/cdrom    Tso9660    ro,noauto    0    0
```

最后的两个域由整数值组成。前一个被 dump 命令用来判断一个文件系统是否需要被清除,将文件系统进行备份。最后一个被 fsck 用来判定是否需要对一个文件系统进行检查以及以何种次序进行。如果域的值为 1,它表明是一个 boot 分区。值 0 表明 fsck 无须对文件系统进行检查。

下面有文件 /etc/fstab 的一个拷贝。注意第一行是注释。所有的注释行都以 # 开头。/proc 文件系统的入口是一种特殊的入口,它被 Linux 操作系统用来管理它的进程。它并不是一个实际的设备。

```
/extc/fstab
```

#<device >	<mountpoi nt>	<fileyste m type>	<options>	<dump>	<fskorder >
/dev/hda3	/	ext2 defaults		0	1
/dev/hdc	/mnt/cdrom	iso9660	ro,noauto	0	1
/dev/fdo	/mnt/floppy	ext2	defaults,no auto	0	
/pdo	/proc	proc	defaults		
/dev/hda2	none	swap	sw		
/dev/hda1	/mnt/dos	msdos	defaults	0	0

为了要在 /etc/fstab 文件中加入一个入口,你或者直接编辑 /etc/fstab 文件,或者利用实用具 fstool 为你提供信息,并在 /etc/fstab 文件中加入正确的入口。

正如前面所描述的,你可以把 MS-DOS 操作系统使用的 MS-DOS 分区装到 Linux 的文件结构中,同你安装任何的 Linux 文件系统相同。你只要把文件的类型指定为 MS-DOS 即可。你可能觉得当你启动 Linux 系统时自动安装 MS-DOS 分区是很方便的。要做到这一点,你只要在你的 /etc/fstab 文件中为 MS-DOS 分区加一个入口即可。你可以为每一个想要安装的 MS-DOS 分区设置一个入口,并在你想安装的目录名之前批定分区的设备名。目录 /mnt/dos 是一个逻辑上的选择(要确保 /mnt 目录之中已经创建 3dos 目录)。至于文件系统的类型,输入 msdos。下一个例子列出了一个 etc/fstab 文件的标准 MS-DOS 分区的入口,注意在 etc/fstab 文件的最后一项是一个安装 MS-DOS 分区的项。

```
/dev/hda1 /mnt/dos msdos defaults 0 0
```

对于任何一个在文件 etc/fstab 中有条目的分区,你可以只用 fstab 中的项指

定的安装目录对之进行安装,你无需输入设备文件名。mount 命令将查询 fstab 文件中的项对之进行寻找,利用目录来标识项,这样,就可以找到设备名。例如,要将上一个例子中 /dev/hda1 DOS 分区卸载,mount 命令,只须知道它所安装的目录即可,在这个例子里是 /mnt/dos。

```
# mount /mnt/dos
```

如果你的 etc/fstab 文件被破坏了——比如说,有一行偶然被删除或修改了——那么你的系统将会启动并进入一种维护模式,允许你对分区进行只读访问。要想进行读/写访问以修复你的 etc/fstab 文件,你需要重新安装你的主分区。下面的命令进行了这样的操作:

```
# mount -n -O remount, ro/
```

你可以利用这里的 etc/fstab 文件的例子来看到你的条目应该是何种样子。/proc 和 swap 分区项是尤其重要的。

7.3.5 Linux 文件系统管理者: Lisa 和 fstool

你可以用 Lisa 工具来管理你的 fstab 文件,而不是直接对它进行编辑。Lisa 是同 amd daemon 装在一起进行工作的,因此它是首选的。在系统设置菜单中,你会看到一项是“安装表设置”,这一项会弹出,图 7-6 所示的菜单,列出了对你的 mount 表,即 etc/fstab 文件的项进行显示,添加和删除的选项。

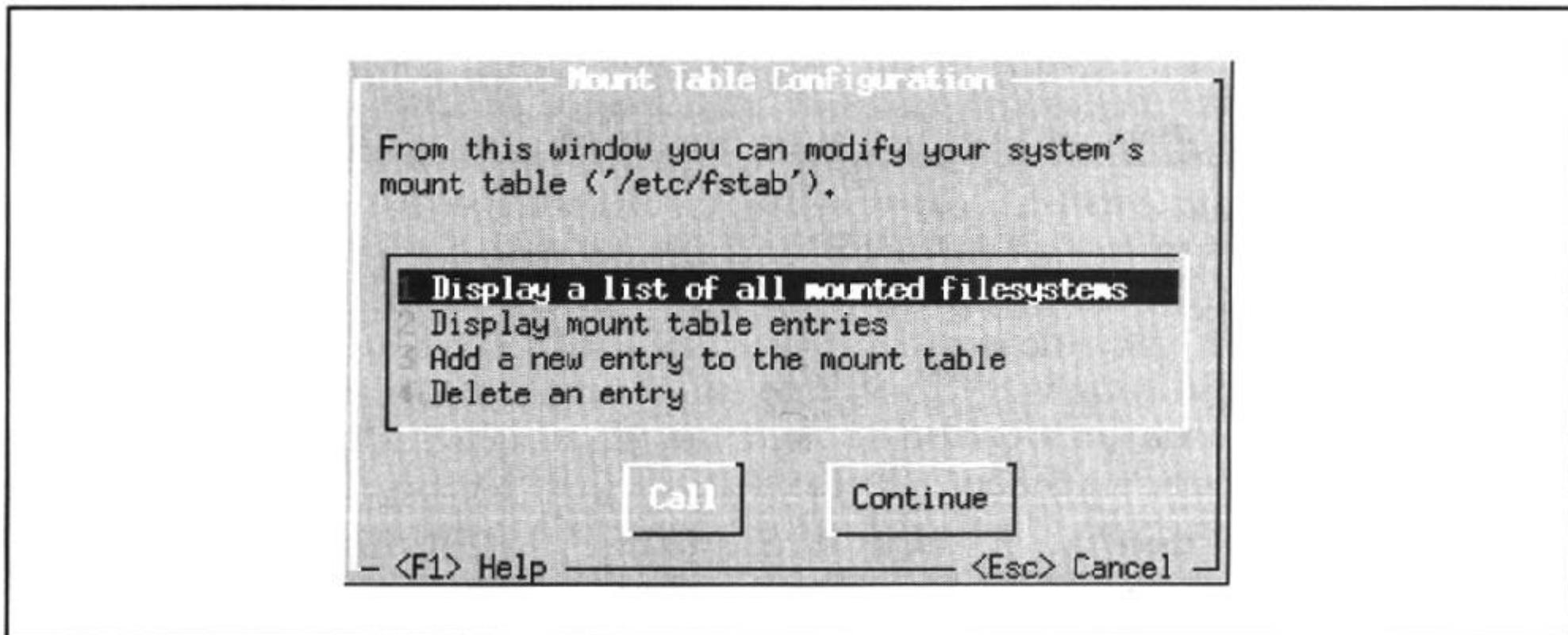


图 7-6 Lisa mount 表的选项

要加入一项,选择“在 mount 表中加入新的项目”。下一屏的显示要求你输入要安装的分区的名字或你想要安装的路径(见图 7-7)。它会使这一项生效,并确保它是一个实际的分区或是一个可访问的路径,然后你会被要求输入你想将分区中的文件系统安装到的目录的名字。再然后你会被要求确认文件系统是只读的,还是可以读写的。

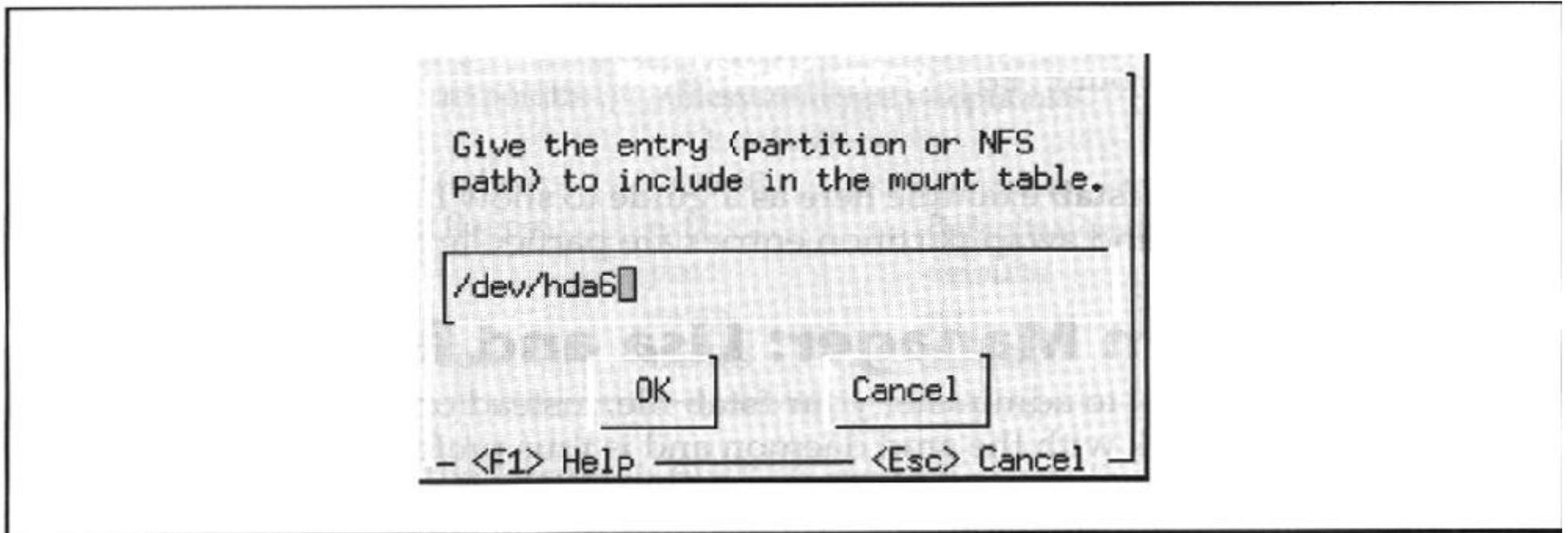


图 7-7 在 mount 表中加入一项

“Delete an entry”选项允许你交一项从文件系统中删除。要小心不要无意中删除根文件系统，即安装在 / 和 /proc 以及 swap 分区的文件系统。

如果不使用 Lisa，你也可以使用 Red hat 的 fstool 实用工具来管理你的 etc/fstab 文件。fstool 现在并不是设计为与 amd daemon 一同使用的。你可以从 fvwm 任务条或菜单来访问 fstool，也可从 Admin-Tools 文件夹的 Caldea 根桌面对这进行访问。当你启动 fstool 后，一个列着你的 etc/fstab 文件内容的窗口会出现。使用其中的按钮和菜单，你可以安装或卸载文件系统，在 etc/fstab 文件中加入新的文件系统，也可以更改你的文件系统的任何选项。你也可以轻松地加入 NFS 文件系统，图 7-8 显示了 fstool 的窗口。

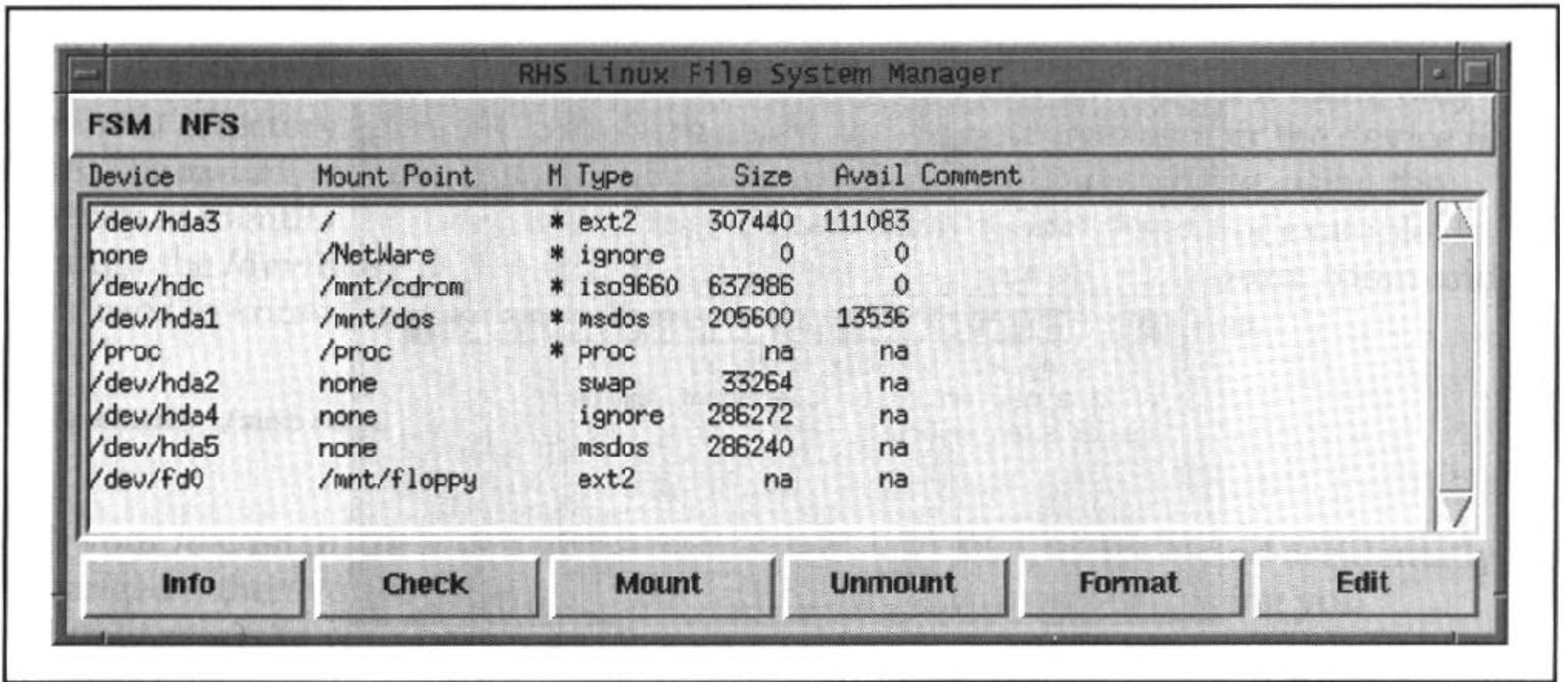


图 7-8fstool 窗口

在 fstab 文件中列出的文件系统的项被显示在窗口中央的栏内。最顶端的行列出了每个域的名字。下端依次排开的五个按钮描述了你可以对文件系统执行的不同操作。只要单击文件系统的条目,它就会处于高亮状态。然后单击按钮。单击 Info 按钮可以弹出一个显示文件系统详细信息的窗口,包括剩余空间的 byte 数。Check 按钮可以检查文件系统是否有错误。

Mount 和 unmount 按钮可对选定的文件系统安装或卸协。Format 按钮将一个文件系统进行格式化,Edit 按钮允许你改变一些域,例如文件系统的选项。使用 fstool 进要非常小心。你不会想将一个你想进行访问的文件系统卸载,你也不想无意中将一个文件系统格式化,以致丢失上面你所有的数据。

窗口的左上角有两个菜单,一个设置你的文件系统管理员(FSM),另一个设置网络文件系统(NIS),NFS 文件将在下章进行讲述。你可对文件系统管理员进行设置,以在窗口中显示更多的信息,例如你的文件系统中未占用的存贮空间所占的百分比。你也可以用 FSM 菜单在 /etc/fstab 文件中添加或删除文件系统,Add 条目允许你添加一个文件系统,而 Delete 条目允许你删除一个文件系统。例如,要将在一个硬盘分区上的 MS-DOS 文件系统加到 etc/fstab 文件中去,你应从 FSM 菜单中选择 Add,然后输入文件系统的设备名。

7.4 网络文件系统:NFS 和 etc/exports

网络文件系统:(NFS)可以允许你将一中远地计算机上的文件系统如同本地的文件系统一样进行安装。你便可以直接访问远地文件系统上的任何一个文件。这样做有一个优点:在网络上不同的文件系统可以访问同一上文件,而无须各自保留自己的拷贝。只会在远地文件系统中有一个其它的计算机可以直接进行访问的文件。

NFS 在 TCP/IP 的网络中工作,远地计算机使所包含的文件系统能够被其它网络上的计算机所访问。为了完成这项任务,它需要将文件系统输出,这便需要在

称作 `/etc/exports` 的 NFS 设置文件中加入一个条目,与此同时还要同时两个 daemon——`rpc.mountd` 和 `rpc.nfsd` 来支持其它文件系统的访问。`/etc/exports` 文件中的条目指定了要被输出的文件系统以及网络上可对之进行访问的计算机。对于文件系统,输入安装点,即它所要安装的目录。这之后是一个计算机的列表,列出了可以对这个文件系统访问的计算机。在每台计算机后可能跟着包含在括号中以逗号所分隔的安装选项。例出,你想给一台计算机读权限而给另外一台计算机读/写权限。如果只列出了选项,那么它们适用于在一台计算机。表 7-8 列出了安装选项的列表。在 `/etc/exports` 文件中条目的例子如下所示。无安全检查和只读访问权限给予所百的计算机,以对安装在 `/pub` 目录——一个公共访问的通用名字——中的文件系统访问。`ant.trek.com` 计算机被赋予了对安装在 `/home/foodstuff` 目录中文件系统访问的读/写访问的权限。下一项允许 `butterfly.trek.com` 对你的 CD-ROM 进行访问。最后一项禁止任何人对 `/home/richlp` 的访问。

```
/etc/exports
/pub(ro,Tnsecure,all-squash)
/home/foodstuffant.trek.com(rw)
/mnt/cdrombutterfly,trek,com(ro)
/home/richcp(noaccess)
```

一旦 NFS 的文件系统被设置为允许,网络上不同的计算机在使用文件系统之前应首先进行安装。你可以用 `etc/fstab` 文件的条目或用一个明确的 `mount` 命令来安装一个 NFS 文件系统。在 `etc/fstab` 文件文中 NFS 的条目的安装类型是 NFS。一个 NFS 文件系统名包括文件系统所在计算机的主机名,其后是它所安

装的目录名。二者之间由冒号分隔开。便如,rose,berkeley,edu:/nome/prject 指定了文件系统被安装在名为 rose.berkeley.edu 的计算机的/home/project 目录中。

NFS 也有自己的一些特殊选项可供你在 NFS 项中使用。你可以指定数据报的大小,以及你的计算机等待主机回应的的时间。你也以指定一个文件系统是进行硬安装还是进行软安装。对于一个硬安装的文件系统而言,如果因为某种原因远地的系统回应失败,你的计算机会不断地与远地计算机联系。对于软安装而言,在一个给定的时间间隔这后,你的计算机会停止与远端主机的联系并给出出错信息。硬安装是缺省的。表 7-9 和 mount 命令的页包含着这些 NFS 客户端选项的列表。它们与前面所说的 NFS 服务器端的选项是不同的。

下面是一个 NFS 条目的例子。远地系统是 ant.trek.com,文件系统被安装在 /nome/projects 目录中。文件系统将被安装在本地的/home/richlp 目录中。系统的类型是 NFS,timeo 选项指定本地系统将等待十分之一秒的二十倍,即两秒钟,来等候回应。

```
ant.trek.com:/home/Projects /home/richlp  nfs  time0=20
```

你也可以使用 mount 命令和 -t nfs 选项来明确地安装一个 NFS 文件系统。要明确地安装前一个条目,可以使用下述命令:

```
#mount -t nfs -o time0=20 ant.trek.com:/home/[rpkects/home/richlp
```

7.4.1 NIS

在支持网络的 NFS 中,许多资源和设备被同一个系统所共享。正常情况下,

每个系统都应有自己对每个设备和资源的设置文件。如果有变动,则需要对每个系统都加以改动。然而 NFS 提供了一种被称为网络信息服务(NIS)的特殊服务,它可为整个网络维护其设置文件。如果有变动,你只需改变 NIS 文件即可。NIS 对大部分管理信息所需的信息都是有效的,例如与用户有关的信息,网络或设备的访问。命名,你可以用 NIS 服务来维护密码信息,你只须修改 NIS 的密码文件即可。

NIS 最早是由 Sun Microsystem 开发出来的,它开始被称为 Sun 的黄皮书。NIS 文件保存在一个 NIS 服务器上(NIS 服务器有时也被称作 YP 服务器)。在网格上的单独的系统利用 NIS 客户来向 NIS 服务器发送请求。NIS 服务器把它的信息存在一个叫作 maps 的特殊的数据库文件中。现在有 NIS 客户端和服务器的 Linux 版本 Linux 的 NIS 客户很容易地就可以连接到使用 NIS 的网络上。

你的 CD-ROM 中有 Linux 的客户端和服务端,它们在 RPM 包中被作为缺省选项进行了安装。NIS 的客户端是作为初始的 Open Linux 安装的一部分来进行安装的,你在任何时候都可以用 Lisa 来设置你的 NIS 客户端。NIS 的客户端程序是 ypbind(NIS 客户端的 daemon)ypwhich,ypcat,yppoll,ypmatch,yppasswd 和 ypset。每一个都有列着自己的详细使用方法的 man 页,NIS 的服务端程序有 ypserv,ypinit,yppasswdd,yppush,ypxfr 和 netgroup——每一个都有自己的 man 页。在 /usr/doc/HOWTO 目录中可以找到一上详细的 NIS——HOWTO 文档。

7.5 档案文件和设备:tar

tar 实用工具可为文件和目录创建档案。利用 tar,你可以为某一特定文件创建档案,在档案中改变文件,或者向档案中加入新的文件,只要你愿意。你甚至可以为目录以及其中的文体和子目录创建档案,然后在必要的时候恢复。tar 最早是被开发用来在磁带上创建档案。名词"tar"代表的是磁带档案(tape archive),你可以在任何设备上创建档案,如软盘,你也可以创建另一个档案文件包含一个档案文件。对于备份文件或看将几个文件组合成为一个文件以便于网络传输而言,tar 是一个非常理想的实用工具。

在 Linux 中,tar 通常被用来在设备或文件上创建档案。利用 F 选项和设备或文件的名称,你可以指示 tar 命令将档案置于特定的设备或文件中。利用 tar 命令和 F 选项的语法如下例子如下例所示。设备或文件名通常被引用作档案文件中。当为 tar 的档案创建一个文件时,这个文件名的扩展通常是如果你指定了一个目录名,那么它所的子目录都会被包含在档案之中。

```
$ tar optionsf archive-name.tar directory-and-filenames
```

要创建一个档案,应使用 C 选项。与 F 选项一起,C 选项会在设备或文件中创建档案。你应在 F 选项的左边输入这一选项。注意在 tar 命令的选项前面没有横线。表 7-10 列出了 tar 的不同选项。在下一个例子中,目录 mydir 和它所有的子目录都被存入文件 myarch.tar 中。

```
$ tar cf myarch.tar mydir
```

用户然后可以利用 x 选项将目录由磁带中恢复。xf 选项将从一个档案文件

或设备进行恢复。tar 命令的恢复操作也会产生它所有的子目录,在下一个例子中,xf 选项指示 tar 命令从 tar 文件 myarch.tar 中恢复所有的文件和子目录。

```
$ tar xf myarch.tar
```

你可以用 r 选项向一个已经被创建的档案中添加文件。r 选项将文件添加到档案的末尾。在下一个例子中,用户将 Letters 目录中的文件添加到档案 mgarch.tar 中。

```
$ tar rf myarch.tar Letters
```

如果你想要改动那么已经存档的目录中的文件,你可以使用 u 选项使 tar 更新档案中的文件。tar 比较每一个档案文件与相应用户目录中文件最近一次被修改的时间,然后将上一次存档这后又被修改过的文件拷贝到档案文件中。在用户目录中新创建的文件也会被添加到档案文件中。在下一个例子中,用户用 mydir 目录中最近被修改或添加的文件来对 myarch.tar 文件进行更新。

```
tar uf myarch.tar mydir
```

如果你想查看在一个档案中存贮着哪些文件。你可以使用 tar 命令和 t 选项。下一个例子将列出所有存贮在 myarch.tar 档案中的文件。

```
tar tf myarch.tar
```

要将文件备份到一个特定的设备,只须把设备名作为档案。在下一个例子中,用户在 /dev/fdo 设备的软盘中创建了一个档案,并将 mydir 目录中所有的文件都拷贝到档案中。

```
$ tar cf/dev/fdo mydir
```

要恢复设备磁盘中的文件,可使用 xf 选项:

```
$ tar xf/dev/fdo
```

如果你所备份的文件大小超过设备可用的存贮器,如软盘,你可以创建一个使用多个标号的 tar 档案。M 选项指示 tar 命令提示你使用一个新的存贮设备,当玖百的设备已满的时候。当使用 M 选项向一个软驱进行存档时,tar 命令在一张软盘已满的时候会提醒你再放入一张新的软盘。这样你就可以把 tar 档案存入几张磁盘中。

```
$ tar cmf/dev/fdo mydir
```

要恢复几张盘中的档案,只要将第一张放入软驱,然后输入有 X 和 M 选项的 tar 命令。当必要时你会被提醒放入另外一张软盘。

```
$ tar xmf/dev/fdo
```

tar 操作不会对档案文件实行压缩操作。如果你想压缩档案文件,你可以指示 tar 激活 gzip 实用工具来在存档之前对文件进行压缩。利用 Z 选项,tar 在对文件存档之前首先要使用 gzip 来压缩文件。当恢复文件时,同样的 Z 选项将激活 gzip 对文件解压。

```
$ tar czf myarch.tar mydir
```

记住压缩档案中的单个文件和整体压缩档案是不同的。通常,档案文件是为了便于传输而把几个文件组合成为一个 tar 文件。为了缩短传输时间,档案应当越小越好。你可以将压缩可以对之压缩,减小它的大小,然后传送压缩后的版本。接收者可以对之解压缩,然后恢复 tar 文件。在 tar 文件上使用 gzip 将会产生有 .tar.gz 扩展的文件。扩展 .gz 被加到一个被压缩的 gzip 文件名之后。下面的例子创建了 myarch.tar 的一个压缩的版本,它又具有了 .gz 的扩展。

```
$ gzip myarch.tar
```

```
$ ls
```

```
$ myarch.tar.gz
```

如果你已经指定了一个特定的缺省设备,比如说磁带,你希望在上面创建一个档案,你只需要使用 `tar`,而无须使用 `f` 选项和设备或文件名。这对进行你文件的备份是有帮助的,缺省设备的名字的名字存在被称为 `/etc/default/tar` 的文件中。在下一个例子中说明了使用缺省设备的 `tar` 命令的语法。如果你指定了一个目录中,那么它所有的子目录都会被包含在档案中。

```
$ tar option directory-and-file-names
```

在下一个例子中,目录 `mydir` 和它所有的子目录都被存在缺省磁带设备的一个磁带上。

```
$ tar c mydir
```

在下一个例子中,`mydir` 目录和它所有的文件及子目录被从缺省的磁带设备中恢复出来并放入用户的工作目录中。

```
$ tar x mydir
```

7.5.1 XTar

`XTar` X-Windows 应用程序可让你使用 Windows 和菜单来选择 `tar` 档案文件并将之恢复。你可以在 `Xterm` 窗口中输入命令 `xtar` 或者从 `fvwm` 工作菜单中选项 `xtar`。`xtar` 有三个菜单:`XTar`,`Options` 和 `Attributes`。你首先应使用 `XTar` 菜单上的 `Open` 项来选择一个 `tar` 档案文件。这一操作将打开一个窗口。窗口中显示了一个目录中的文件,你可以从一个目录移到另一个目录。一旦你选择了你所希望的 `tar` 档案,组成 `tar` 档案的所有文件将在主窗口中列出。利用 `XTar` 你可

选择将整个 tar 档案恢复处处或者只是恢复其中的几个文件。要恢复整个档案,应选择 Xtar 菜单中 Unpack All item 选项。这一选项允许你将档案恢复到一个你指定的目录中。要恢复某一人单独的文件,你首先要选中这一文件(你可以使用 XTar 菜单中的 Search 功能来找到这一文件)然后选择 Option 菜单中 Extract item 项。选项菜单中也有一上 View 项来显示诸如 README 文件之类比较短的文本文件。图 7-9 显示一个显示 tar.gz 文件的 XTar 程序,XTar 也可工作于压缩的档案(你无须首先将之解压)。

7.6 文件压缩:gzip

减小文件的大小有若干个原因。最普遍的两个是减小存贮空间和通过网络传输文件时,减少传输的时间。gzip 实用工具是用来对文件压缩和解压的 GNU 实用压缩工具。要压缩一个文件,只要输入命令和文件名。这将使用文件的压缩版本来代替原文件,现在的文件有 3.gz 扩展。

```
$ gzip mydata
$ ls
mydata.gz
```

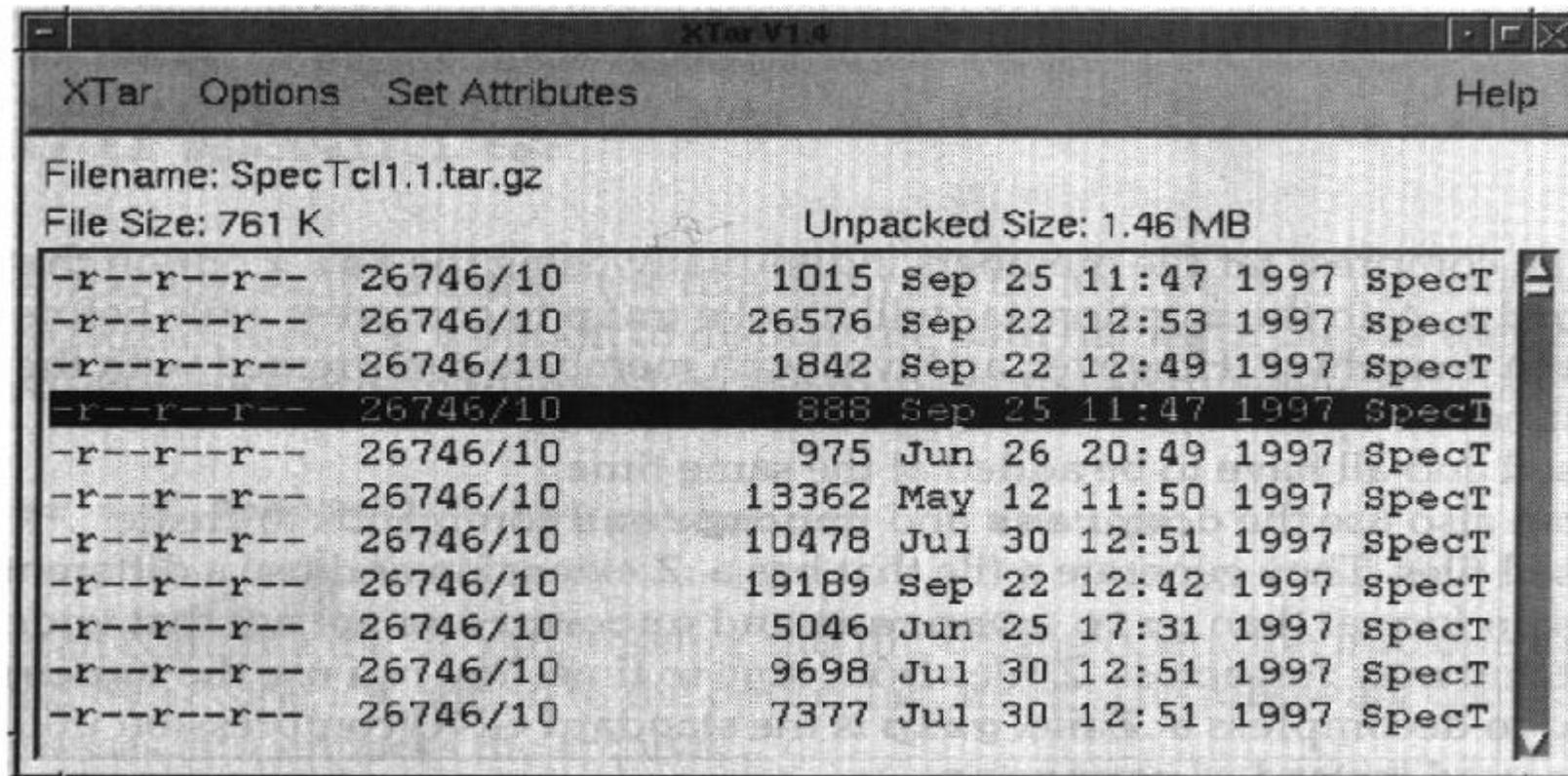


图 7-9 XTar 归档工具

要将一个 gzip 文件解压,可使用 gzip 命令和 -d 选项或 gunzip 命令。这些命令将把有 .gz 扩展的压缩文件解压,并用有相同根名的解压版本代替压缩文件,但是没有 .gz 扩展。当使用 gunzip 时,你甚至都有用输入 .gz 扩展。gunzip 和 gzip -d 将假定文件有 .gz 扩展。表 7-11 列出了 gzip 的不同选项。

```
$ gunzip mydata.gz
```

```
$ ls  
mydata
```

假定你希望显示或打印一个压缩文件的内容而不对文件先进解压。Zcat 命令将产生文件的一个解压版本并将之发向标准输出。你然后可以把这个输出定向到打印机或才显示上一实用工具,例如 more。原有的文件将保留它原有的压缩状态。

```
$ Zcat mydata.gz /more
```

你也可以压缩一个 tar 档案文件。这将产生有。tar.gz 扩展的文件,压缩的档案文件通常被用于通过网络传输大文件。

```
$ gzip myarch.tar
```

```
$ ls
```

```
myarch.tar.gz
```

你可以用 tar -Z 选项激活 gzip 来单独压缩 tar 文件。利用 -Z 选项,tar 将在把文件置入档案之前首先对文件进行压缩。然中,如果档案中的文件被用 -Z 选项压缩,档案不能被更新,也有能添加文件。所有有文件都必须被同时压缩并添加。

你也可以用 compress 和 uncompress 命令来创建压缩文件。它们产生具有 .Z 扩展的文件,使用不同的压缩格式。compress 和 uncompress 使用并不很普遍,但你不时会看到有 .Z 扩展的文件。你可用 uncompress 命令来将有 .Z 扩展的文件解压。gzip 是标准的 GNU 压缩实用工具,而 compress 不是。

7.7 从压缩的档案中安装软件 :.tar.gz

Linux 软件程序可以从互联网的不同网站上获得。你可以下载这些软件并将它们安装在自己的系统上。你可用在第 10 章中描述的 ftp 来下载文件。文件通常是以压缩的档案文件下载的。这是用 tar 命令存档并用 gzip 压缩的文件。要安装这样的文件,你首先要用 gunzip 实用工具将文件解压,然后用 tar 命令将组成软件包的文件和目录恢复。你也可以使用 gzip -d 来代替 gunzip。下一个例子将文件 Spec Te/l.l.tar.gz 解压,并用一个称作 SpecTe/l.l.tar 的解压版本代替它。

```
$ ls
specTe/l.l.tar.gz
$ gunzip spec Te/l.l.tar.gz
$ ls
Spec Te/l.l.tar
```

首先使用 tar 和 t 选项来检查档案的内容。如果第一项是一个目录,这一目录将被创建,然后将恢复的文件放入其中。如果第一项不是一个目录,你首先应该创建一个目录,然后将档案文件拷贝到其中去。然后在这个目录中恢复文件。如果第一项不是目录,那么文件将被恢复在现有的目录中。你需求自己创建一个目录来容纳这些文件。

```
$ tar tf SpcTell.l.tar
```

现在你已经准备好从 tar 档案中恢复文件。你可以使用 tar 和 X 选项来恢复文件,用 V 选项来在恢复文件时显示路径名,和 F 选项后面加档案文件的名字。

```
$ tar xvf SpcTell.l.tar
```

你可将解压和解包命令组合一个 tar 命令中,在选项表中加入 Z 选项,xzvf。下一个命令将对档案解压和解包。

```
$ tar xzvf SpcTell.l.tar.gz
```

软件的安装对每个包而言都是不同的。随着安装程序通常有相应的指示。

7.7.1 从网点下载压缩档案

许多的跨平台开发或设计的软件包可能不是 RPM 格式。相反,它们可能被存档并被压缩。这些文件的文件名可能是以 .tar.gz 或者 .tar.z 为扩展。实际上,大部分有 RPM 格式的软件都有相应的 .tar.gz 格式。在将文件下载之后,你首先应用 gunzip 将之解压,然后用 rar 命令将之解包。在将文件下载之后,你首先应用 gunzip 将之解压,然后用 tar 命令将之解包。许多 RPM 包中只包含着软件应用程序的二进制版本。如果你想得到应用程序的源代码,你应下载并解包那一软件的压缩档案文件。

在下一个例子中,用户用 ftp 连到 sunsite.unc.edu 的 Linux 网点。对于登录的身份,用户输入了匿名,对于密码,用户输了自己的网址。下载模式可用输入关键词 binary 设为二进制类型。利用 cd 命令,用户进入 pub/Linux/Libs/x/xview 目录,Xview 窗口管理软件就在这一目录中(见第 3 章)。然后 get 命令将包下载。Close 命令切断了连接,quit 离开 ftp 应用程序。

```
#ftp sunsite.unc.edu
```

```
Connected to sunsite.unc.edu
```

相应的,你也可以用 Netscape 或另一种网络浏览器来进行访问,直接浏览并下载软件而无须为 ftp 的命令发愁。确保要在一个 ftp 站点名之前加入术语 ftp //而不是通常的 http //。对于 Sunsite 你应输入 ftp //sunsite.unc.edu。一旦你选择了你想要的软件,按下 shift 键并单击文件以下载。一旦下载了文件,以 z 或 gZ 为结尾的文件是必须要被解压的压缩文件。你可以用 gunZip 命令和文件名来解压。

```
#gunZip Xview-3.2Pl.4.bin.tar.gZ
```

如果文件以 tar 结尾,它是一个必须用 tar 命令来解包的档案文件。在解压缩之前,你需要数的档案将被解包到一个它们创建的子目录中去。将组成软件包的文件或档案放于那个子目录中去。例如,xview-3.2p1.4.bin.tar 将被解包到称为 usr 的子目录中。要检查一个档案是否会被解包到一个目录中,利用 tar 和 t 选项可以列出它的内容并查看名字的前面是否有一个目录。

如果是那样,目录就将被创建,被恢复的文件就要被放在目录中。如果没有目录名,你需要先创建一个目录并将档案文件拷贝到目录中。然后在该目录中恢复档案。

```
#tar tf xview-3.2p1.4.bin.tar
```

现在你作好从 tar 档案中恢复文件的准备了。你可以用 tar 和 x 选项来恢复文件,用 v 选项在恢复文件时显示路径名,f 选项之后是档案文件名。

```
#tar xvf xview-3.2p1.r.bin.tar
```

每个软件包的安装可能都是不同的。每一个安装程序都有相应的指示。下载的文件中通常有 README 文件或其它的文档。确保要参照这些文档。

7.8 编译软件

许多软件可能是源代码的形式,在你安装之前你需要首先对它进行编译。这一点对于跨平台应用程序而言是事实。设计为在不同的 Unix 系统,如 sun 和 Linux 系统中运行的程序通常被分散为源代码以下载。并在不同的系统中加以编译。编译这些软件的任务最近由于设置脚本的使用最近已经大为简化,它将检测一个系统的设置并相应地编译程序。例如,在一个系统中的 C 编译器可能是 gcc 或 cc0 设置脚本将检测存在哪些东西并用它来编译程序。

在你编译软件之前,你首先应阅读它所带的 README 和 INSTALL 文件。它会给予编译和安装这一程序的详细的指示。如果软件使用了设置脚本,那么编译和安装通常只包括下面三个简单的命令:

```
#./configure  
#make  
#make install
```

· /configure 命令执行了设置检测的操作。make 命令执行实际的编译操作,利用前一个操作产生的编译文件脚本。make install 将把程序安装在你的系统上,将可执行文件存在如 /usr/local/bin 的目录上,并将任何的设置文件放在 /etc 目录中。它所创建的任何共享的目录都可能放在 /usr/lib 目录中。

如果你要编译一个基于 X-Window 的程序,要确保 X11 开发库已经被它装好。如果在安装 Open Linux 系统时你选择了标准安装,这些库不会被安装你需要用 Lisag 和 lint 对之进行安装。

要保证 xmkmf 程序也已被安装。查询 21 章关于 X-Window 程序更多的信息。许多 X-Window 应用程序可能需要特殊的共享库。例如,一些应用程序可能需要 xform 库或 gt 库。其中一些库你需要从网点上取得。例如,gt 库在 www.troll.no 上,尽管 gt 和 xforms 的 RPM 版本都在 Red hat 的 contrib 目录。

许多 X-Window 应用程序利用 xmkmf 代替一个设置角本来产生所需的。makefile。在这种情况下,你只需在 `./configure` 的位置上输入 `xmkmf` 命令即可。一定要参考软件的 README 或 INSTALL 文件。通常,你只需在包含软件源代码的目录中执行下述命令即可。

```
xmkmf
make
make install
```

确保要检查这些软件的文档来检查你是否要对 makefile 做任何的改动。可能会有几个改动,但更加详细的改动需要有对 C 语言的理解以及 make 是如何跟它一起工作的。如果你成功的设置了 makefile,你只需输入 `make` 和 `make install` 操作即可。例如,许多在 sunsite.unc.edu 网点的 `/pubLinux/games` 目录中的游戏可能需要对 makefile 作一些修改。一个可能的问题是定位 C 和 X-Windows 的开发库。X-Windows 库在 `/usr/X11R6/lib` 目录中。标准 C 库在 `/usr/lib` 目录中。

7.9 mtools 实用工具:msdos

你的 Linux 系统提供了一套被称为 mtools 的实用工具,它们可让你轻易地访

问格式化为 MS-DOS 的软盘。不再需要进行如安装等一些特殊的操作。利用 mtools,你不用安装一个 MS-DOS 便可对它进行访问。对一个 MS-DOS 的软盘,只要把软盘放在软驱中,你就可以用 mtool 命令来访问软盘上的文件。例如,要把一个文件从 MS-DOS 的软盘拷贝到你的 Linux 系统中,可以用 mcopy 命令。你可以用 a:代表 A 驱动器来指定 MS-DOS 盘。与通常的 DOS 路径名不同,mtool 命令使用的路径名用斜杠来代替反斜杠。在 A 驱动器中的 doc 目录将用路径为 a:/docs 来引用,而不是 a:\docs 下一个例子将文件 mydata 拷贝到 MS-DOS 盘中,并将 MS-DOS 盘中的 preface 文件拷贝到现在的 Linux 目录中。注意与 DOS 不同,mtools 用斜杠而不用反斜杠。

```
$mcopy mydata a:
```

```
$mcopy a:/preface
```

你可以用 mdir 命令来对 MS-DOS 命令都可以通过 mtool 操作获得。你可以用 mmd 命令创建 MS-DOS 你可以用 mdir 命令来对 MS-DOS 盘上的文件列表,你还可以用 mcd 命令在上面更换目录。下一个例子列出了你的软驱中 MSDOS 磁盘中的文件,然后又进入那个驱动器的 docs 目录。

```
$ mdir a:
```

```
$ mcd a:/docs
```

大部分的标准 MS-DOS 命令都可以通过 mtool 操作获得。你可以用 mmd 命令创建 MS-DOS 目录,用 mdel 删除 MS-DOS 文件。表 7-12 列出了 mtool 命令的一个列表。例如,要显示驱动器 b:,一个 MS-DOS 与 1/4 英寸软驱,中的一个文件,可使用 mtype 命令和 b:再加上文件名。

```
$ mtype b:/readme
```

对 MS-DOS 分区的访问是在 `/etc/mtools.conf` 文件中设置的。这一文件列出了几个不同的缺省 MS-DOS 分区和磁盘驱动器。每一个驱动器或分区用一个特殊的设备名来标识。你的软驱的项已经被输入,利用设备名 `/dev/fdo` 和 `/dev/fdl` 来代表第一个和第二个软驱。在 `/etc/mtools.conf` 文件中的项采用了下面的形式:首先是驱动器的标号,其次是名词 `file` 和等号,然后是你想用这一标号来标识的分区或驱动器的设备名。设备名初放在引号中。例如,第一个硬盘分区是一个 MS-DOS 分区,设备名为 `/dev/hda1`,下面的项将把它标识为 MS-DOS 系统上的 `c:` 驱动器。

```
drive c:file= /dev/hda1
```

你的分区有正确的设备名是很重要的。它们被列在 `/etc/fstab` 文件中,可以用根用户桌面上的 `fstool` 实用工具来对它浏览。如果你有一个 SCSI 硬盘分区,这一硬盘分区有这样的形式,首先是 `sd`,其后是一个标识硬盘的字符和代表分区的数字。`Sda1` 指的是 SCSI 硬盘上的第一个分区。IDE 硬盘有 `hd` 的形式,其后是一个字符和一个分区的字母。`hda1` 指的是 IDE 硬盘的第一个分区。

当你第一次安装 Open Linux 系统,你的 `/etc/mtools.conf` 文件将把 `c:` 驱设置为指向一个 SCSI 硬盘分区。对 IDE 的硬盘分区也有一个设置,不过它被前面所加的 `#` 号注释掉了。如果你有一个 IDE 硬盘(如大部分用户),你需要将 IDE 硬盘分区项前的 `#` 号去掉,而将 `#` 号加在 SCSI 硬盘分区项之前。另外,如果你的 IDE 硬盘上的 MS-DOS 分区不是第一个分区,你就需要更改设备名。例如,如果 MS-DOS 分区是第二个分区,设备名应是 `/dev/hadz`。如果你有几个 MS-DOS 分区,你需要为每一个分区加入条目,每个分区赋一个不同的标号。下一个例子将 `d:` 标号给了一个 IDE 驱动器的第四个硬盘分区。

```
drive d:file= /dev/hda4
```

```
/etc/mtools.conf
    drive a:file = /dev/fdo exclusive
    drive a:file = /dev/fdl exclusive
#First SCSI hard disk partitton
#device c:file= /dev/sdal
#First IDE hard disk partitton
    drive c:file= /dev/sdal
    drive c:file= /dev/sdal
#dosemu floppy image
drive m:file= /var/cib/dosemu/diskimage
# dosemu hdimage
drive n:file= /var/cib/dosemu/diskimage offset=3840
#Atali ramdisk image
drive o:file= /tmp/atari-rd offset=136
mtools-lower-case=1
```

一旦 DOS 硬盘分区被设置了,你可以用驱动器盘符来从 Linux 分区中拷文件或拷贝入系统。下一个命令将文件 mydoc.html 拷到了 C 分区的 webstuff 目录,并将之更名为 mydoc.htm.注意斜线的使用。

```
$ mcopy mypage.html c:/webstuff/mydoc.htm
```

因为 DOS 和 Linux 处理文本文件的新的行是不同的,当把一个 DOS 的文本文件拷贝到 Linux 分区中是你应使用 -t 选项。下一个命令将 c:/project 目录中的 mydoc.txt 文件拷贝到 /newdocs 目录中。

```
$ mcopy -t c:/project/mydoc.txt/newdocs
```

7.10 DOS 和 Window 仿真器:DOSEmu,Wine 和 Willow

Linux 中有仿真器来使你运行 DOS 和 Window 程序。这些技术仍在发展,它们的成功也只是部分的,作为仿真器,它们比 DOS 和 Window 运行程序的速度慢。DOS 的仿真器 DOSEmu。你可以从你的 Open Linux 的 CD-ROM 中对它进行安装。它包括一个在例子目录中叫作 config.dist 的示范设置文件。它包含了一长串的设置选项。所有选项前面都用 # 注释掉了。你要把文件拷贝到 dosemu.conf 文件中。编辑 dosemu.conf 文件,将适用于你系统中的项前的 # 号删掉。文件当中有每节的注释。

要创建一个 DOS 启动盘,用 mcopy 命令把 command.com,sys.com,emufs.sys,ems.sys,cdrom.sys 和 exitemu.com 文件拷贝到盘中。要运行 DOS 仿真器,只要在命令行中输入 dos。要退出仿真器,只要输入 exitemu.dos-? 可列出 dos 命令选项。

Wine 仿真器是设计用来运行 Window 程序的。它仍处于发展中,并不断有新版本发布。现在它仍被认为是试验性的。你可以从 /pub/Linux/ALPHA/Wine/development 目录下下载一个版本,网点是 sunsite.unc.edu.Window FAQ 提供了详细的信息。

Willow 工具仓为跨平台的开发提供了一个 Window API。它包括了 Willow Twin 库和工具。应用程序可用 Window API 来编写,并用 Willow Twin 库来在不

同的系统上执行。关于 Win32 和 Window 应用程序都有支持。Willow Twin 库在 GNU 的公共授权下可以获得。可查询 Willow 的网点来获取信息 (www.willow.com)

7.11 总结:文件管理

你可在文件或目录上设置许可来控制其它用户的访问。一个文件或目录可被设置为读写或执行。拥有者,同组用户和其它用户可对文件进行访问。chmod 命令可设置这些许可,ls 命令和 -l 选项将列出文件,目录和它们现在的许可。

不同的存储设备都有自己的文件系统,你可以将之安装并访问。你可以访问软盘,CDROM 和硬盘分区。你也可以安装 MS-DOS 硬盘分区并从 Linux 系统中直接加以访问。将一个文件系统的项放在 /etc/fstab 文件中,你可以在启动时自动安装文件系统。如果在网络上,你可以安装并访问远地服务器上的文件系统。

要传输或备份文件,你可以将之存档并对之压缩。tar 命令将文件存档,gZip 命令将文件压缩。大部分的软件包都可由网上获得,它们都被存档并压缩。它们通常有 tar.gz 的扩展。你可以下载软件,用 gunZip 解压,用 tar xf 命令来恢复档案。你可以用 rpm 命令来安装。

表 7-1 chmod 命令选项

命令或选项	执行
chmod	改变一个文件或目录的许可选项
+	加入一个许可
-	去掉一个许可
=	将整个许可集赋值
r	设置一个文件或目录的读许可。一个文件可被显示并打印 一个目录可以列出目录中的文件
w	设置一个文件或目录的写许可。一个文件可被编辑或删除,一个目录可以被删除
x	设置文件或目录的执行许可。如果文件是 shell 脚本,它可被作为程序执行。一个目录可以初改变或进入
u	为创建文件或目录的用户设置许可
g	为组群对文件或目录的访问设置许可
o	为系统上其它的用户对文件或目录的访问设置许可
a	为拥有者,组用户或其它用户的访问设置许可
s	设置用户 ID 来组 ID 许可
t	设置许可的粘着置位; 程序驻留内存
chgrp groupname	改变一个文件的组

续表

file names	
chown user name	改变文件的所有者
filenames	
ls-l filename	显示文件及它的许可
ls-l d directory	列出目录名和它的许可
ls-l	列出一个目录中的所有文件和它们的许可

表 7-2 mount 命令

mount 选项	描述
-f	伪装一个文件系统的安装。用于检查一个文件系统可否安装
-v	Verbose 模式。mount 命令显示它所采取动作的描述用 -f 选项检查安装文件系统的问题, -fv
-w	安装一个文件系统,并使之有读和写许可
-r	只按读许可安装文件系统
-n	安装一个文件系统而无须在 mstab 文件中加入条目
-t type	指定要安装的文件系统类型。表 7-6 列出了有效的文件系统类型
-a	将 /etc/fstab 文件列出了文件系统加以安装
-o option-list	用一个列表的选项安装文件系统。选项之前以逗号分隔,其后是 -o。表 7-7 有完整的选项列表和 mount 命令的 man 页

表 7-3 用于检查和修复文件系统的 fsck 选项

选项	描述
file-system	指定要检查的文件系统。利用文件系统的名称,如 /dev/hda3
-A	检查文件 /etc/fstab 中列出的所有文件系统
-V	Verbose 模式,列出 fsck 采取的行动
-t file-system-type	指定要检查的文件系统类型
-a	自动修复
-l	列出文件系统中所有的文件名
-r	在修复系统之前征求确认
-s	在检查文件系统之前列出 superblock

表 7-4mkfs 选项

选项	描述
blocks	文件系统中的块数目。1440 块代 1.44MB 的软盘空间
-t file-system-type	指定要格式化的文件系统类型。缺少类型为标准的 Linwx 类型,ext2
fs-options	指定文件系统的选项
-V	Verbose 模式。显示 mkfs 所采取的动作
-v	指示 mkvfs 启动的文件系统 builder 程序显示它的行动
-c	在格式化之前检查文件系统是否有坏块
-l file-name	读坏块的列表

表 7-5fdisk 命令

命令	描述
a	设置或取消一个分区的可启动标志
c DOS 兼容性标志
d	删除一个分区
l	列出分区类型
m	显示 fdisk 命令的列表
n	创建一个新的分区
p	打印分区表,列出盘上所有的分区
q	无需储存便退出。如果你犯了错误,用它来显示 fdisk 对话
t	选择一个分区的文件系统的类型
v	验证分区表
w	将分区表写入磁盘后退出。此时改变是不可恢复的
x	显示高级的 fdisk 命令。你可以用来设置标的数目,扇区和头,打印原始数据,改变数据在分区表中的位置

表 7-6 文件系统类型

类型	描述
minux	Minux 文件系统,文件名在 30 个字符之内
ext	Linwx 文件系统的早期版本,现已不在使用
ext2	支持长文件名和文件大小的标准 Linwx 文件系统类型
xiaf	xiaf 文件系统
msdos	MS-DOS 分区文件系统
hpfs	OS/2 高性能分区文件系统
proc	被操作系统用于进程
nfs	从远地系统中安装分区的 NFS 文件系统
umsdos	UMS-DOS 文件系统
swap	Linwx swap 分区或 swap 文件
sysv	Unixt 系统 V 文件系统
iso 9660	安装 CD-ROM 的文件系统

表 7-7 文件系统选项 :-O 和 /etc/fstab

选项	描述
async	文件系统的所有 I/O 都异步进行
auto	可以用 -a 选项进行安装
defaults	使用缺少选项:rw,suid,dev,exec,auto,nouser 和 async
dev	解释文件系统上的字有或块特殊设备
noauto	只能明确加以安装。-a 选项不会安装系统
exec	允许二进制文件的执行
nouser	禁止普通用户安装文件系统
remount	试图安装一个已经安装的文件系统。这通常被用于改变一个文件系统的安装标志,尤其是将一个只读文件系统变为可写
ro	对文件系统进行只读安装
rw 可写
suid	允许设置用户标识或设置组标识位有效
sync	所有文件系统的 I/O 同时进行
user	允许普通用户安装文件系统。普通用户可激活下列选项:noexec,nosuid 和 nodev
nodv	不中断文件系统上的字符或块特殊设备
nosuid	不允许设置用户标识或设置组君标识位有效

表 7-8/etc/exports 选项

通用选项	描述
secure	需要证实。这在缺少条件下为有效
ro	只允许只读访问
rw	允许读写操作,缺少选项
noaccess	使目录中所有东西对指定的用户为不可访问
link-absolute	使符号连接维持原状。缺少操作
link-relative	当绝对符号连接(link 内容以斜线开始)转换为相对连接用户 ID 映射
squash-uids	指定一系列受无名映射控制的 uids 和 gids
squash-gids	
all-squash	将 uids 和 gids 映射到匿名用户。对 NFS 输出的公共 ftp 目录 news 缓冲池目录等有用
no-all-squash	这是 all-squash 的对立选项。缺少设置
root-squash	将 uid/gido 的请求映射到匿名的 uid/gido
no-root-squash	将 root squashing 关闭。不映射 uid/gido 的请求。缺少设置
anongid	这些选择明确地设置匿名帐号的 uid 和 gid
anonuid	这些主要对 PC/NFS 用户有用,这时你希望所有的请求都似乎来自一个用户

续表

map-daemon	开启动态 uid/gid 映射。在 NFS 请求中的 uid 被转换为相应的服务器 uids.在 NFS 回答中的 uids 被用另外一种方法映射。缺少设置是映射身份,这不会变动 uids
------------	--

表 7-9 NFS 选项

选项	描述
rsize=n	当从 NFS 服务器读取文件时 NFS 所使用的字节数。缺少为 1024 字节
wsize=n	当向 NFS 服务器写文件时,NFS 所使用的字节数。缺少为 1024 字节
time=n	在时间到后进行第一次重新传输的十分之一秒数缺少值为十分之七秒
retry=n	在放弃之前尝试一个后台 NFS 安装操作的次数。缺少为 10 000 决
soft	用软安装来安装系统
hard 硬 ,缺少设置
intr	允许 NFS 打断文件操作并返回调用程序。缺少为不允许文件操作被打断

续表

bg	如果第一次安装的尝试时间到,继续在后台尝试安装。缺少是失败而不进行后台安装
tcp	用 TCP 协议而不用缺少的 UDP 协议安装 NFS 文件系统

表 7-10 文件备份:tar

命令	执行
tar options files	将文件备份到磁带,设备或档案文件
tar options f archive-name filelist	将文件备份到指定为 archive-name filelist 的文件或设备上。可以是文件或或目录名选项
c	创建一个新的档案
t	列出了一个档案中所有文件名
r	将文件添加到一个档案中
u	用新文件和变更的文件更新档案只添加那些在存档之后又被变更和档案中不存的文件
w	在将每个文件存档之前征求用户的确认;允许你有选择地更新档案
x	从档案中恢复文件
m	当从一个档案中恢复文件时,不增加新的时间标签

续表

M	创建多卷的档案,以便在几个磁盘中存放
f archive-name	将磁带档案存到文件档案名中,而不是缺少的磁带设备。当给定档案名时 f 选项将 tar 档案存到此名的文件中
f device-name	将一个 tar 档案存到诸如软盘或磁带的设备中。/dev/fdo 是你的软盘的设 备名;缺少设备在 /etc/default/tar-file 中
v	显示存档的每个文件名
z	用 gzip 来对文件进行压缩和解压

表 7-11 gzip 选项

选项	执行
-c	将文件的压缩版本送往标准输出;所列的每个文件都被单独进行压缩
-d	将压缩文件解压;同样,你也可以使用 gunzi
	<code>\$gzip -d myfiles.gz</code>
	<code>\$gunzip myfiles.gz</code>
-h	显示帮助列表
-l file-list	显示所列文件压缩或未压缩的文件大小

续表

-r directory-name	查找特定的目录并压缩其中所有的文件；查找由现在的目录开始；用同 gunzip 同使用时,在特定目录中被压缩的文件将被解压
-v file-list	对每一个压缩和解压的文件,显示文件名及压缩比
-num	决定压缩的速度和大小；范围是由-1到-9。一个小的数字有比较小,会产生一个压缩和解压速度快的比较大的文件。-1有最快的压缩速度,但文件尺寸最大；-9文件尺寸最小,但压缩和解压速度最慢。缺少是-6

表 7-12Mtools 访问命令

命令	执行
mcopy filename filename	在 MS-DOS 盘和 Linwx 系统之间进行文件拷贝。下面的命令将一个由 MS-DOS 盘拷贝到 Linwx 系统中:mcopy a:/filename directory-or-filename
mcd directory-name	在 MS-DOS 文件系统中更换目录。下面列出了你的软驱中 MS-DOS 盘中的文件
mdir	列出了你软驱中 MS-DOS 盘中的文件
mattrib	改变一个 MS-DOS 文件的属性
mdel filename	删除一个 MS-DOS 文件
mformat	将一个 MS-DOS 文件系统加入磁盘中

续表

mlabel	建一个卷标
mmd directory-name	建一个 MS-DOS 目录
mrd directory-name	删除一个 MS-DOS 目录
mread filename filename	低级读(拷贝)一个 MS-DOS 文件到 Unix
mren filename filename	将一个 MS-DOS 文件更名
mtype filename	显示一个 MS-DOS 文件的内容
mwrite filename filename	低级读(拷贝)一个 Unix 文件到 MS-DOS

