操作系统

实验指导

薛万奉　汇编

第一部分 进程管理

实验 1：UNIX 环境下复合命令的使用

UNIX 在发展过程中形成了一个丰富灵活的运行环境，这一环境为程序设计提供了广泛的资源，很多问题的求解可以不必按传统方法编程，而仅需把现有的资源组织在一起。UNIX 许多命令看起来简单而普通，但是当它们组织在一起时就表现出强有力的功能和用途。

1. 常用的 UNIX 命令

UNIX 命令格式：系统提示符 $ 指令名称 选择项 参数

注意，选择项用“-”开头。

(1) 列出文件目录命令 ls

格式：$ ls [-laF…] [name]

name：目录或文件名称。

-a：打印所有的文件，包括隐含的以“.”起始的文件。

-F：在目录名称后加上“/”标记，在执行文件后加上“*”标记。

-l：长列表显示文件和目录信息。

例：$ ls -Fl /bin

(2) 文件排序命令 sort

格式：$ sort [-nr…] [+position [-position]] [file…]

file：欲排序的文件。

-n：根据数值的大小排序。

-r：表示由大到小排序。

sort 可根据整个记录的大小排序，也可分别根据数据项进行排序，作为排序依据的数据项称为关键字段 key。在文件中若没有特别指定，sort 会以空白键或 tab 键作为数据项的间隔。当用数据项作为排序标准时，需要说明开始和结束的数据项号码。“+”号表示 key 开始的数据项，从该数据项开始往后的数据项都作为 sort 的依据。而“-”号表示从该数据项开始，往后的数据项不作为 key。数据项号码从 0 起编号。例如以第 3 个数据项作为排序文件 employs 的关键字的命令表示为：$ sort +2 -3 employs。

(3) 选取文件有关字段列命令 cut

格式：$ cut -clist [file1…]或 $ cut —flist [-dchar] [file1…]

file1：文件名。

-list：指定字段的范围，格式为整数-整数。

-c：以字符为单位。

-f：以数据项字段为单位，从 1 起编号。

char：字段分隔符。

在 c(或 f)与 list 之间不能有空格,同样 d 与 char 之间也不能有空格。

(4) 按合并行方式合并文件命令 paste

格式: $ paste [-dchar] file1 file2…

file1, file2:文件名。

char:字段分隔符。

cut 命令用于从文件中裁下一个垂直片段,而 paste 命令把若干垂直片段合并到一个文件中,这 2 个命令合作完成重排文件的列。cut 和 paste 的作用类似于文本编辑组合——剪切和粘贴。

(5) 文本文件搜索命令 grep

格式: $ grep [-y…] pattern [file…]

file:要搜索的文本文件。

pattern:搜索的字符串模型,最好将字符串括在单引号内。

-y:不区分大小写。

该命令执行后会将 file 文件中符合字符串模型的所有行输出显示。

(6) 输出重定向命令>和>>

UNIX 系统约定标准输入是键盘,标准输出和标准错误输出是屏幕。UNIX 可使用输出重定向命令>,它表明该命令的标准输出被定向到由命令下一个字表示的文件中。通常输出的重定向完全覆盖了那个输出文件。如使用输出重定向的另一个命令>>,则可把输出定向加到那个文件的末尾。

(7) 管道命令|

参见 2.4.3 节。

2. 实验

UNIX 大部分可执行的指令和程序都放在/bin、/usr/bin 子目录下,试用 UNIX 命令组合按字母排序列出指令的清单,每行包括文件名、存取权限、文件大小。

(1) 使用简单命令

$ ls -Fl /bin >file1	将/bin 子目录下文件和子目录存入 file1。
$ ls -Fl /usr/bin >>file1	将/use/bin 子目录下文件和子目录追加到 file1。
$ grep '*'file1 >file2	从 file1 中选取可执行文件的行,存入 file2。
$ cut -c54- file2 >file31	从文件 file2 中选取文件名垂直列存入 file31。
$ cut -c1-13 file2 >file32	从文件 file2 中选取文件类型和存取权限垂直列存入 file32。
$ cut -c16-22 file2 >file33	从文件 file2 中选取文件大小垂直列存入 file33。
$ paste file31 file32 file33 >file4	把 3 个文件按并行方式合并成 1 个文件 file4。
$ sort file4 >file	将文件 file4 按文件名由小到大排序,存入 file。

命令使用中生成了临时文件 file1、file2、file31、file32、file33、file34 和 file4。

(2) 使用组合命令

$ ls -Fl /bin /usr/bin | grep '*' | cut -c54-, 1-13, 16-22 | sort +54 >file

此时每一行各字段次序为文件类型和存取权限、文件大小、文件名。

实验 2：系统性能的监视

1. 系统监视器

(1) 概述

系统监视器作为 Windows 98 的一个工具，使用户可以通过测试硬件、软件服务和应用程序的性能来判别本地或远程计算机上出现的问题的原因。当用户对系统设置进行修改时，系统监视器显示用户的修改对整体系统性能的影响，用户可用系统监视器使硬件升级正常化。

在进行较大的配置改变之前，用系统监视器来评估用户当前配置有助于用户判定某个专门系统或网络组件是否充当了性能瓶颈。

(2) 功能

系统监视器可跟踪以下类别的功能：核心、拨号适配器(Dial-Up Adapter)、磁盘高速缓存、文件系统、内存管理程序和 Windows 网络客户。

(3) 运行系统监视器

在“开始”菜单项上，先指向“程序”，后指向“附件”，再指向“系统工具”，然后单击“系统监视器”，出现系统监视器窗口，如图 2-18 所示。

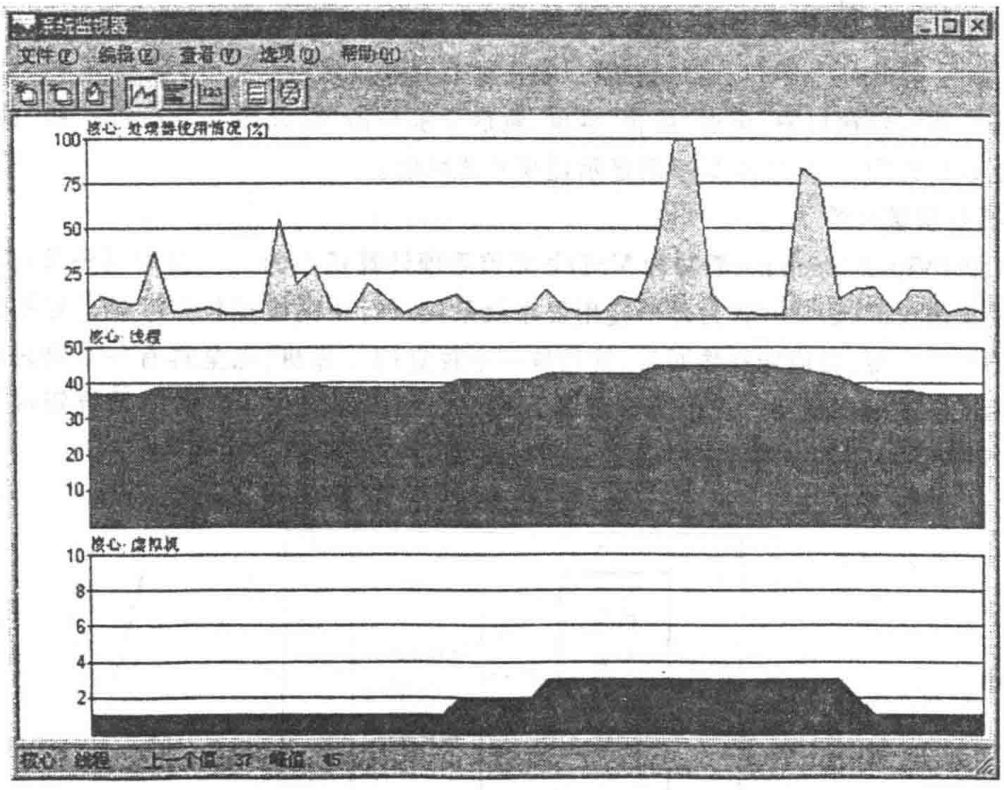

图 2-18 系统监视器窗口

(4) 系统监视器功能菜单

系统监视器窗口菜单条有“文件”、“编辑”、“查看”、“选项”和“帮助”5 个菜单。“文件”菜单有“连接”、“开始记录”、“停止记录”、“退出”等子菜单；“编辑”菜单有“添加项目”、“删除项目”、“编辑项目”、“清除窗口”等子菜单；“查看”菜单有“工具栏”、“状态栏”、“隐藏标题栏”、“前端显示”、“拆线图”、“条形图”、“数字图”等子菜单；“选项”菜单有“图表”子菜单；“帮助”菜单有“帮助主题”、“关于系统监视器”2 个子菜单。

(5) 系统监视器窗口显示图表项目的编辑

为了调整系统监视器窗口显示图表项目，可使用“编辑”菜单。单击“编辑”菜单的“添加项目”子菜单，出现添加项目对话框，如图 2-19 所示。对话框左边列出跟踪性能的类别，框右边列出显示选中类别的项目。图中选中的类别为核心项目的处理机使用情况(%)、线程和虚拟机。

(6) 改变系统监视器显示图表外观

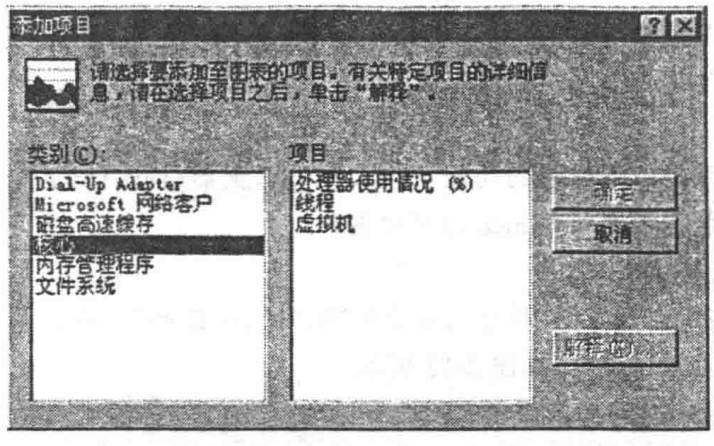

图 2-19 添加项目对话框

在系统监视器窗口中，单击“查看”菜单，然后分别单击“拆线图”、“条形图”和“数字图”可以分别以拆线图、条形图和数字图显示监视系统性能。

2. 虚拟机管理器

虚拟机(Vitual Machine，缩写为 VM)管理器管理计算机中的每个应用程序及每个系统进程所需要的资源，创建并保持每个应用程序及系统运行中的虚拟机器环境。虚拟机是在内存中的一个环境，对应用程序而言，就像是一个独立的计算机，完全具有一个物理计算机运行应用程序的所有资源。虚拟机管理器为每个应用程序提供了所需的系统资源。Windows 98 基本组件结构中的虚拟机管理器的框图如图 2-20 所示。

图 2-20 虚拟机管理器框图

Windows 98 只有一个称作"系统虚拟机"的虚拟机,在这里运行所有的系统进程。每个在系统虚拟机上运行的 32 位的基于 Windows 的程序都有各自的地址空间,而在系统虚拟机上运行的 16 位的基于 Windows 的程序共享相同地址空间和资源,以便与 Windows 3.x 向下兼容。每个 MS-DOS 程序也有自己的虚拟机器。而 Windows NT 有能力让每个 16 位的基于 Windows 的程序在其独立的虚拟机器中运行。虚拟机管理器负责提供 3 个关键领域的服务。

(1) 管理进程及多任务

Windows 98 使用 2 种方式来管理同时发生的进程——合作方式的多任务以及优先级的多任务。

Windows 3.x 通过被称为合作方式的多任务来同时运行应用程序,在这种方法中,应用程序周期性地检查消息队列,为别的运行着的程序放弃它的系统资源。为了向下兼容,Windows 98 支持 16 位的 Windows 程序通过合作方式实现多任务。

Windows 98 为 32 位的 Windows 程序提供优先级的多任务,这表示操作系统可向运行中的任务提供控制,这与系统的要求有关。32 位的 Windows 程序能够享有多线程的优势,这是 Windows 提供的一种能够同时运行多个应用程序的机制。在系统中运行的 32 位 Windows 程序称作进程,每个进程至少包含一个运行着的线程。线程是一组能够从操作系统取得时间片并且能够与其他代码同时运行的、与进程相联系的代码。32 位的 Windows 应用程序能为一个给定的进程初始化多个线程,通过改进输入、响应及后台进程来提高应用程序的性能。

一个最好的多任务例子就是 Windows 98 本身。Windows 98 本身是 32 位的 Windows 进程,每一个打开的文件夹窗口是运行着的独立的线程。当用户进行一个在两个窗口间的拷贝操作时,操作在目标窗口的线程中实现。这样,用户可以不中断地使用其他的窗口,还可以在另一个窗口中开始别的拷贝操作。

(2) 内存分页

Windows 98 像 Windows NT 一样,使用要求分页的虚拟内存系统,该系统基于线性的内存空间,通过 32 位的地址来访问。

(3) 支持 MS-DOS 模式

有一些 MS-DOS 程序需要完全占有系统资源才能运行,虚拟内存管理器为这种程序创建排他的操作环境,称作 MS-DOS 模式。当一个 MS-DOS 程序在 MS-DOS 模式下运行时,另外的程序或进程不能同它争夺系统资源,所有的资源被该 MS-DOS 程序排他占有。

3. 系统核心性能的监视实验

(1) 选择菜单"开始"/"程序"/"附件"/"系统工具",然后单击"系统监视器",出现系统监视器的窗口,如图 2-18 所示。

(2) 选择系统监视器窗口"编辑"菜单的"添加项目"子菜单,出现"添加项目"对话框,在对话框左边"类别"列表框中选择"核心",在右边项目列表框中显示"处理器使用情况(%)"、"线程"和"虚拟机"3 个项目,如图 2-19 所示。选中以上这 3 项,按"确定"按钮后,图 2-18 系统监视器窗口中显示与这 3 个项目对应的 3 个图表。如原有图表中已有这 3 个项目中的某项,则应在右框内避免相应选择,否则会出现重复现象。

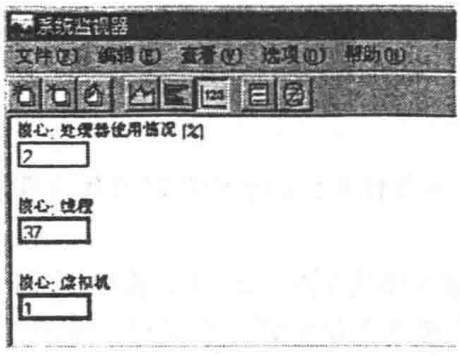

图 2-21　系统监视器的数字图

(3) 单击“123”按钮,或选择菜单“查看”/“数字图”,系统监视器窗口显示“处理器使用情况(%)”、“线程”和“虚拟机”3 个项目的数字图,如图 2-21 所示。图中显示“处理器使用情况(%)”为 2 个,“线程”为 37 个,“虚拟机”为 1 个。

(4) 启动一个 32 位程序(例如资源管理器),点击 Windows 子目录,再检查系统监视器窗口,“线程”增加到 39 个,但“虚拟机”仍为 1 个。

(5) 点击资源管理器目录 C:\Windows\Dosprmpt 文件。显示 MS-DOS 窗口,监视器窗口中“线程”增加到 41 个,“虚拟机”增加到 2 个。再点击此文件,又显示一个 MS-DOS 窗口,检查系统监视器窗口,“线程”增加到 43 个,“虚拟机”增加到 3 个。

(6) 再启动一个 32 位程序,例如 Microsoft Word,再检查系统监视器的窗口,“线程”增加到 45 个,但“虚拟机”仍为 3 个。

(7) 按序先后关闭 Word、MS-DOS、MS-DOS 和资源管理器,再检查系统监视器的窗口,“线程”恢复为 37 个,“虚拟机”恢复到 1 个。以上过程用拆线图显示“处理机使用(%)”、“线程”和“虚拟机”3 个项目随操作时间变化如图 2-18 所示。从图中可看出:在执行打开和关闭文件时,“处理器使用情况(%)”大大提高,即 CPU 工作负担增加。

第二部分 存储管理

实验 1:在 Windows 95/98 下观察虚拟内存管理程序的性能

Windows 95/98 大量使用虚拟内存。应用程序通过页面调度程序(换页器)使用虚拟内存。换页器跟踪计算机中内存的使用情况,当需要更多的内存空间时,换页器会将一部分内存页面换到硬盘上,以此来模拟更多的内存。应用程序不会意识到自己是在真正的物理内存上,还是在硬盘的虚拟内存中。

我们可以用"系统监视器"来观察虚拟内存管理程序的性能。例如:如果虚拟内存管理程序的页故障值很高,那么可能是正在运行的应用程序所需要的内存数量超过了计算机的容量。

实验步骤:(例如我们先来观察内存的换页情况)

(1) 启动"附件"/"系统工具"/"系统监视器"。

(2) 选择"编辑"/"删除项目",在随之出现的对话框中将以前加入的项目都选中(可按[Ctrl]或[Shift]),单击"确定"按钮。

(3) 选择"编辑"/"添加项目",在"类别"列表中选择"内存管理程序",在"项目"列表中选择"进页"、"出页"、"页故障"和"已分配的内存",单击"确定"按钮。

(4) 要模拟页面出错,必须要打开较多的程序。

(5) 选择"折线图"查看方法,结果如图 3-41 所示。

从实验中可以看出系统换页的工作,当打开大量程序时,进页和页故障数就大大增加。可以利用"系统监视器"观察虚拟内存管理程序性能的项目有:

(1) 页故障:每秒钟产生页故障的次数,即当加载不在物理内存中的页时,所发生的页出错次数。

(2) 进页:每秒钟产生进页的次数,即每秒钟从虚拟内存交换到物理内存中的页数,其中包含从基于 Win32 的可执行文件或内存映像文件中调入的页。相应地,本值不一定表示缺少内存。

(3) 出页:每秒钟产生出页的次数,即每秒钟从物理内存交换到虚拟内存的页数。

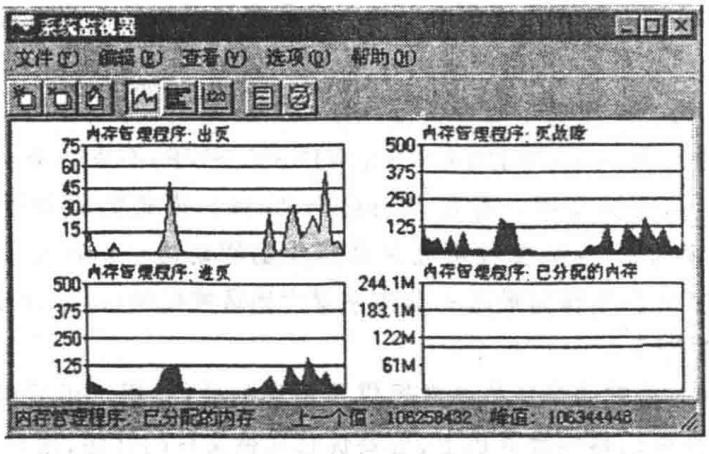

图 3-41 内存的换页情况

(4) 已分配的内存:以 B 为单位的其他内存和可交换内存的总数,如果计算机没有动作但其数值改变了,这表明磁盘高速缓存正在改变其自身大小。

(5) 废弃:每秒钟从内存中废弃的页数,这些页不交换到磁盘,因为磁盘上已经有了这些信息。

(6) 磁盘高速缓存大小:以 B 为单位的磁盘高速缓存的当前大小。

(7) 实例故障:每秒发生的实例故障数。

(8) 锁定的内存:已分配且锁定的内存数。

(9) 锁定的非缓存页:非高速缓存锁定的页数。

(10) 最大的磁盘高速缓存大小:磁盘高速缓存最大可能的大小,是一个系统启动时调入的定值。

(11) 中等磁盘高速缓存大小:磁盘高速缓存的中等大小,是一个系统启动时调入的定值。

(12) 最小的磁盘高速缓存大小:磁盘高速缓存最小可能的大小,是一个系统启动时调入的定值。

(13) 其他内存:已分配的但没有存储到交换文件的内存字节数。这种代码的例子有:Win32 动态链接库、可执行文件、内存映像文件、非分页的内存和磁盘缓存页。

(14) 从高速缓存映射的页面:用来监视 MapCache/WinAlign 的变化。在运行 WinAlign 工具以后,应该监视在该设置的同时使用的交换文件大小,以查看它们的变化情况。

(15) 交换文件损坏:交换文件中有关找到的交换介质物理损坏的字节数,因为交换文件被分配到 4 096 B 页框中,单个扇区被损坏将导致整个页框被标记为坏区。

(16) 正在使用的交换文件:在当前交换文件中使用的字节数。

(17) 交换文件大小:当前交换文件以 B 为单位的大小。

(18) 可交换的内存:从交换文件分配的字节数。按该规格来看,已被锁定的页仍然计为"可交换"。这种代码的例子包括 16 位应用程序以及动态链接库,但不包括 Win32 DLL 及可执行文件。

(19) 未使用的物理内存:当前没有使用的物理内存数值。

实验 2:设置 Windows 95/98 交换文件的大小

Windows 98 在用户的硬盘上使用一个特殊的文件,称为虚拟内存交换文件(或页交换文件)。Windows 98 虚拟内存管理程序可以把某些程序代码和其他信息暂时交换到虚拟内存中。当重新用到这些信息时,再取回到内存中,这样用户就可以使用小内存运行大程序,且这个过程是不可见的。

Windows 98 交换文件(C:\WINDOWS\WIN386.SWP)不是一个永久性的文件。但是也可以使用永久性的交换文件。若在 Windows 98 保护模式的驱动程序(DRVSPACE.VXD)控制下压缩的驱动器,交换文件可驻留在压缩的驱动器上。磁盘空间管理将交换文件标识为不可压缩的,并将其作为最后一个文件置于扇区堆栈中(减小分成碎片的危险),为交换文件留出增长的空间。

尽管交换文件的系统默认设置值通常提供了最好的性能,用户仍可调整那些设置。例如,在一台有并联硬盘驱动器的计算机上,若要优化交换文件的性能,用户可能想忽略 Win-

dows 98 交换文件的默认位置。一般交换文件被置于性能最快的硬盘上，除非该硬盘使用过度。若用户从一台有多个硬盘的计算机的某一个硬盘上调入所有的软件，那么最好把交换文件放到一个并不如此忙的硬盘上，可能会提高运行性能。注意：完全禁用虚拟内存可能会使计算机停止正常运行、或无法重新启动计算机、或系统运行性能降低。虚拟内存太小，会导致过度的页面出错，使系统不断地读盘，系统整体性能下降。

实验步骤：

(1) 利用“资源管理器”查找 Windows 98 的交换文件(C: \ WINDOWS \ WIN386.SWP)。

(2) 启动“控制面板”/“系统”，单击“性能”选项卡，单击“虚拟内存”按钮。打开“虚拟内存”对话框(如图 3-42 所示)。

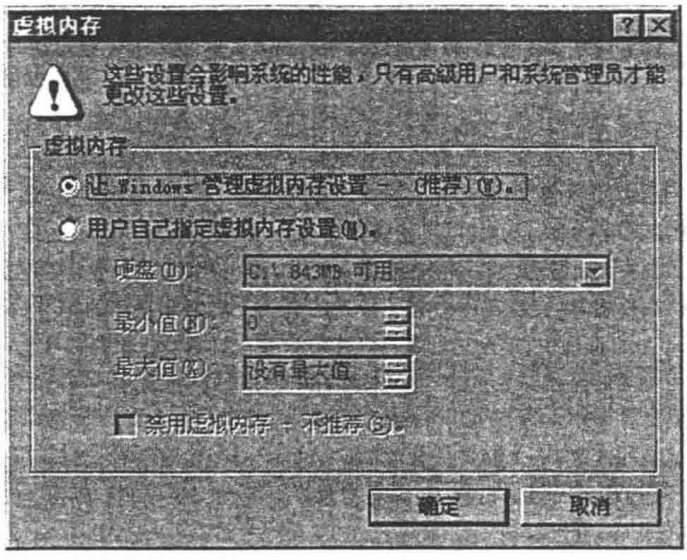

图 3-42 “虚拟内存”对话框

(3) 单击“用户自己指定虚拟内存设置”就可以设置关于虚拟内存的 4 个项目，在“硬盘”列表中可选择用哪一个硬盘存放交换文件(虚拟内存)，盘符后的字节数是驱动器的最大的剩余空间。“最大值”和“最小值”是设置虚拟的内存大小，“最大值”的缺省值是交换文件所在硬盘的最大剩余空间。选取“禁用虚拟内存”则可取消虚拟内存。以上设置都要重新启动系统后才能生效。

实验 3：观察交换文件大小的变化

Windows 98 的交换文件会根据当前的内存的负荷和硬盘的大小自动改变大小。通过此实验来观察交换文件大小的自动变化情况。

实验步骤：

(1) 启动“附件”/“系统工具”/“系统监视器”。

(2) 选择“编辑”/“删除项目”，在随之出现的对话框中将以前加入的项目都选中，单击“确定”按钮。

(3) 选择“编辑”/“添加项目”，在“类别”列表中选择“内存管理程序”，在“项目”列表中选择“交换文件大小”，单击“确定”按钮。

(4) 连续启动多个应用程序，选择“折线图”查看方法，可看到如图 3-43 所示，每启动一个程序，交换文件就扩大一点。

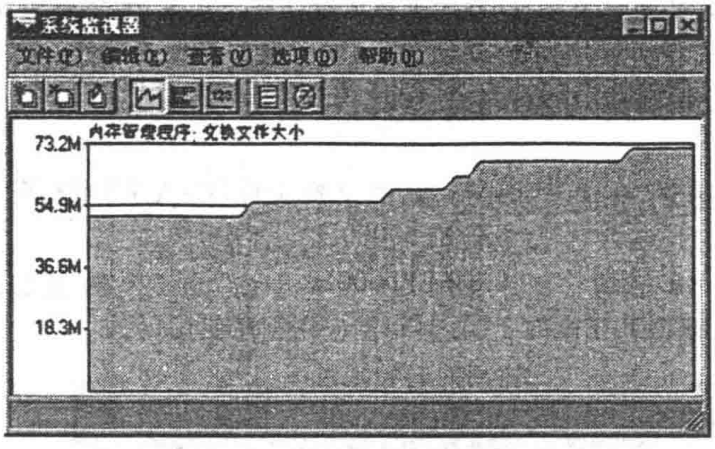

图 3-43 交换文件正在增大

(5) 然后再一个个地关闭应用程序，可以看到每关闭一个应用程序，交换文件会减小一点，但是减少带有滞后性。

实验 4：观察 Windows 95/98 中 32 位模块程序的加载情况

Windows 95/98/NT 使用 32 位寻址的平滑(flat)方式的内存地址空间，将有限的物理内存映射到“虚拟内存空间”，每个应用程序都如同拥有 4 GB 的内存。那么，如何来观察虚拟内存空间的分配情况呢？Windows 98 提供了“系统信息”工具可以用来观察 32 位模块程序的加载情况。

实验步骤：

(1) 启动“程序”/“附件”/“系统工具”/“系统信息”(如图 3-44 所示)。

(2) 切换到“已加载 32 位模块”，观察虚拟内存空间的分配情况。

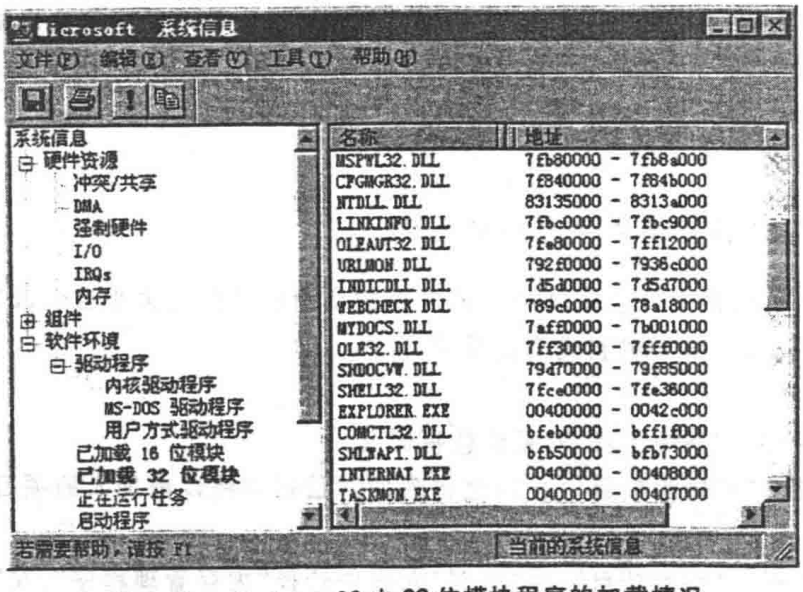

图 3-44 Windows 98 中 32 位模块程序的加载情况

在前面我们讲过从 0X80000000H～0XFFFFFFFFH 的 2 GB 的空间供操作系统、驱动程序与动态程序库等使用，例如：

加载的文件	占用的起始地址
KERNEL32.DLL	0XBFF70000H
USER32.DLL	0XBFF50000H
COMCTL32.DLL	0XBFEB0000H

0X00000000～0X7FFFFFFFH 的 2 GB 空间供应用程序使用，例如：

加载的文件	占用的起始地址
SHELL32.DLL	0X7FCE0000H
EXPLORER.EXE	0X00400000H
INTERNAT.EXE	0X00400000H

当我们观察 Win32 应用程序所载入的地址时，会发现每个应用程序的起始地址(基地址)都是 0X00400000H，这是因为每个 Win32 应用程序都具有独立的地址空间，即认为所有的 4 GB 的地址空间归它所有。

实验 5：调整物理内存

Windows 98 除了可以调整虚拟内存之外，还可以调整物理内存，即可以人为地放弃使用一部分内存。当系统发生问题时，此功能可以用来检测物理内存。例如，物理内存有 64 MB，可以设置为 32 MB，则系统只用 32 MB 的物理内存，忽略剩余的 32 MB。要注意的是：当把物理内存容量设置成小于 10 MB 时，Windows 98 会无法以正常方式启动，而只能进入安全模式！

实验步骤：

(1) 启动“程序”/“附件”/“系统工具”/“系统信息”打开“Microsoft 系统信息”窗口。

(2) 执行“工具”/“系统配置实用程序”命令(如图 3-45 所示)。

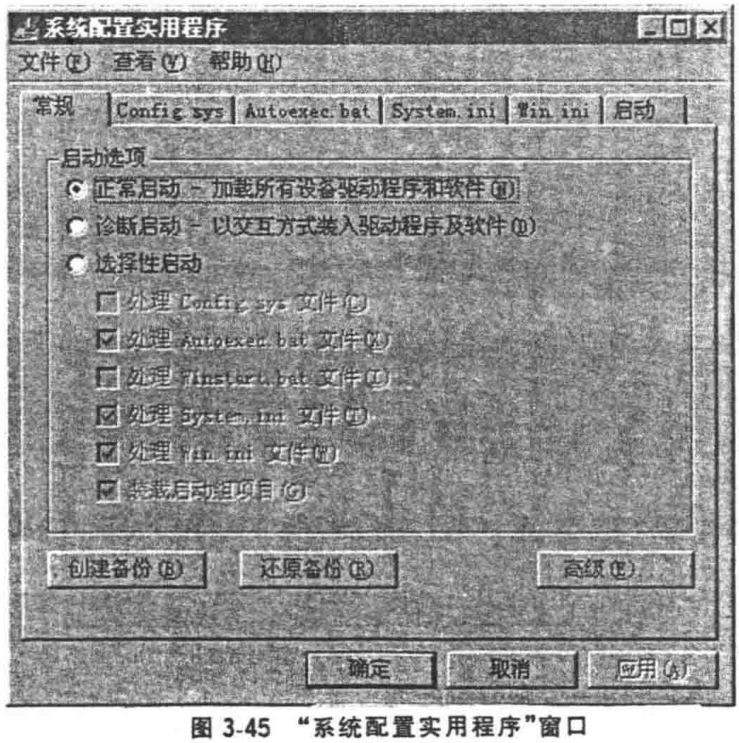

图 3-45 “系统配置实用程序”窗口

(3) 按“高级”按钮，出现“高级疑难解答设置”对话框(如图 3-46 所示)。

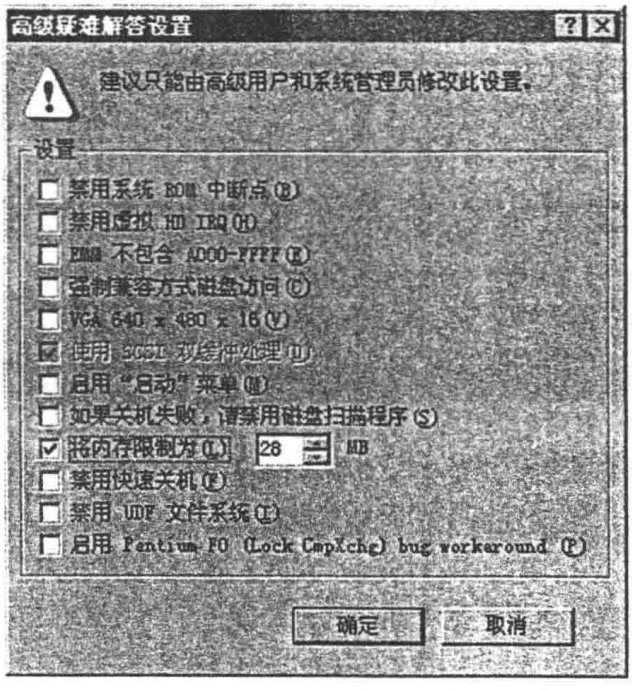

图 3-46 “高级疑难解答设置”对话框

第三部分 设备管理

实验 1：注册表编辑器的使用

注册表编辑器提供了一种相当直接的方式来编辑注册表，注册表编辑器有 4 个主菜单“注册表”、“编辑”、“查看”和“帮助”。“注册表”菜单有“引入注册表文件”、“连接网络注册表”、“断开网络注册表”、“打印”和“退出”几个子菜单。这里介绍注册表编辑器的有关操作。但作者还是建议用户使用控制面板等工具来修改注册表，因为这样更安全更方便。

1. 备份注册表

注册表是 Windows 98 中极重要的数据，一旦误删或改错即无法恢复，尽管在子系统启动时注册表都自动备份到 DA0 文件中，但注册表编辑器还是提供了随时备份和恢复的功能。在对注册表数据备份时，用户可以选择全部备份或是只备份部分的注册表数据。

选择菜单“注册表”/“导出注册表文件”，出现图 4-16 的“导出注册表文件”对话框，在对话框下部的“导出范围”中，使“全部”有效则可对整个注册表备份，如使“选择的分支”有效，

并在编辑下框中输入子主键名，则可对注册表部分主键备份，由于主键名难记，在作“选择的分支”备份时，可先打开注册表编辑器，并在左边树状目录上选定要备份的子主键，然后再选择菜单“注册表”/“导出注册文件”，这时在对话框最下边的编辑框中会显示选中的子主键名。在打开的对话框中指定路径和文件名(缺省扩展名为 REG)，文件类型选择注册表文件。备份的 REG 文件是一般的纯文本文件，可用 Word 打开查看内容。

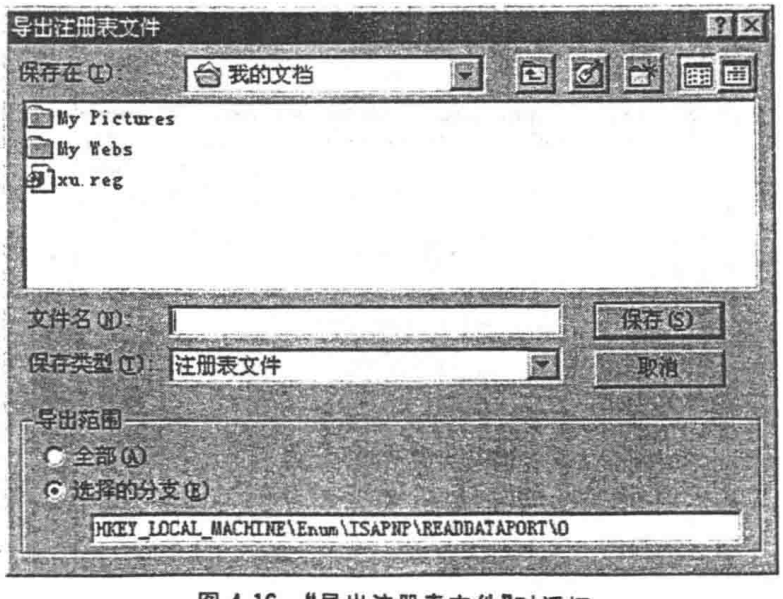

图 4-16 “导出注册表文件”对话框

2. 恢复注册表

(1) 直接点选

找到备份的注册表，直接在文件上双击鼠标左键，注册表会启动相关联的注册表编辑器程序，并且问用户是否要导入该注册表数据，这是最快恢复注册表的方法。

(2) 导入注册表文件

选择菜单“注册表”/“引入注册表文件”,在相应对话框中指定路径、文件名后打开,可恢复注册表的任何一个分支。

(3) 软盘启动命令导入

在 Windows 系统遭破坏情况下,就要用软盘启动,从命令行来恢复注册表,假定原导出注册表文件为 A:\FILE.REG,则使用命令 A:>REGEDIT FILE.REG(扩展名不能省略)将注册表文件引入到注册表(一个文件可能是注册表中的一个子集)。这时的引入不是替换,而是覆盖或叠加。如要替换整个注册表,则用命令 A:>REGEDIT /C FILE.REG。此命令先删除注册表的全部信息后再引入,所以必须保证引入注册表文件包含了注册表的完整信息。

(4) 恢复原系统拷贝文件

每次系统重启或重新登录成功后,都会对注册表文件进行自动备份,它们分别是 SYSTEM.DA0、USER.DA0,如果系统注册表被破坏,可将 SYSTEM.DA0 和 USER.DA0 拷贝到 SYSTEM.DAT 和 USER.DAT 中。

3. 查找注册表信息

要在注册表编辑器中快速查找某个主键或数值,可利用注册表编辑器提供的搜寻功能。在注册表编辑器窗口中选择菜单“编辑”/“查找”,出现图 4-17 的“查找”对话框。在对话框的“查找目标”编辑框中输入查找目标,例如 bmp,在查看框中使“主键”、“键值”和“数据”都有效,按下“查找下一个”按钮,在注册表编辑器窗口会找到第1个保存bmp字符串的注册表数据,如图 5-16 中的注册表 HKEY_CLASSES_ROOT\.Bmp 子键和键值显示。如再选择菜单“编辑”/“查找下一个”,可查找下一个有相同字符串的注册表数据。若直到注册表数据库结束为止都找不到,便会显示“完成对注册表的搜索”信息。

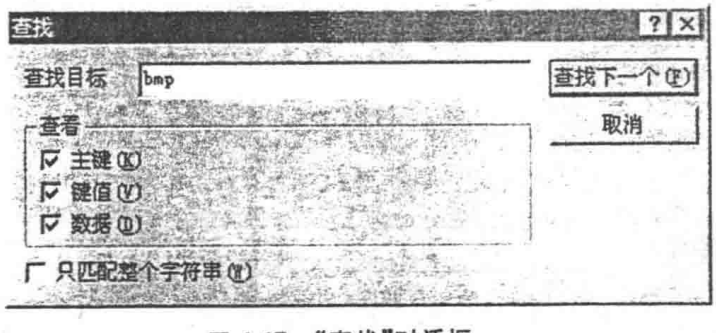

图 4-17 “查找”对话框

实验 2:声卡安装实例

1. 硬件和驱动程序的安装

在这里以 AD1816 声卡为例,看一下它的设备安装过程。

(1) 硬件的安装

硬件的安装非常简单,只需根据声卡插口的种类(ISA 或 PCI),将它插在主板上空闲的 ISA 或 PCI 槽内即可。注意一定要插到底,然后用螺丝固定。

(2) 驱动程序的安装

这块声卡完全支持 Windows PnP 的特性，在 Windows 启动过程中，会被 Windows 自动检测识别，并提示安装驱动程序。当然，也可以先跳过安装，以后再通过选择菜单“开始”/“设置”/“控制面板”/“添加新硬件”来进行安装。

如果选择了继续，添加新硬件向导会自动指引完成驱动程序的安装。如图 4-18 所示，点击“下一步”，出现图 4-19。

在图 4-19 对话框中“指定位置”处输入驱动程序所在的路径，在这里是“C:\unzipped\ad1816”。点击“下一步”，就能看到图 4-20 所示的画面。如果路径输入正确，点击“下一步”，出现图 4-21 所示的画面。点击“完成”，结束驱动程序的安装。

这时，“音量”的图标会显示在 Windows 任务栏的右边，这块声卡就可以正常工作了。

2. 设备参数的检测

设备驱动程序安装完成后，到底给系统带来了哪些变化呢？这可以从 Windows 附件中提供的系统信息实用程序来检查。打开菜单“开始”/“程序”/“附件”/“系统工具”/“系统信息”，出现图 4-22 所示的画面。展开图中左框“组件”/“多媒体”，选择图中右框“多媒体”的“基本信息”栏目，可以看到关于这块声卡的基本配置信息(如图 4-22 所示)。

选择图中右框“多媒体”的“高级信息”栏目，可以看到关于这个设备的更多信息，如图 4-23 所示。包括它的注册表项、强制资源、启动资源、已筛选的资源、基本的资源以及驱动器信息。其中，启动资源即设备当前的资源配置，就是在“基本信息”中的内容。

图 4-18 添加新硬件向导之一

选择图中右框“多媒体”的“历史记录”栏目，可以看到该设备的原始资源配置和最近的配置改变信息，如图 4-24 所示。

这样，我们便在 Windows 提供的“系统信息”实用程序中，得到了这块声卡的详细配置信息。类似的信息也可以在 Windows 设备管理器中得到。打开“控制面板”/“系统”，点击“设备管理器”选项卡，出现图 4-25 所示的画面。选中图中“声音、视频和游戏控制器”下的声卡条目，点击“属性”，选择“资源”选项卡，出现图 4-26 所示的画面。在“资源”选项卡中

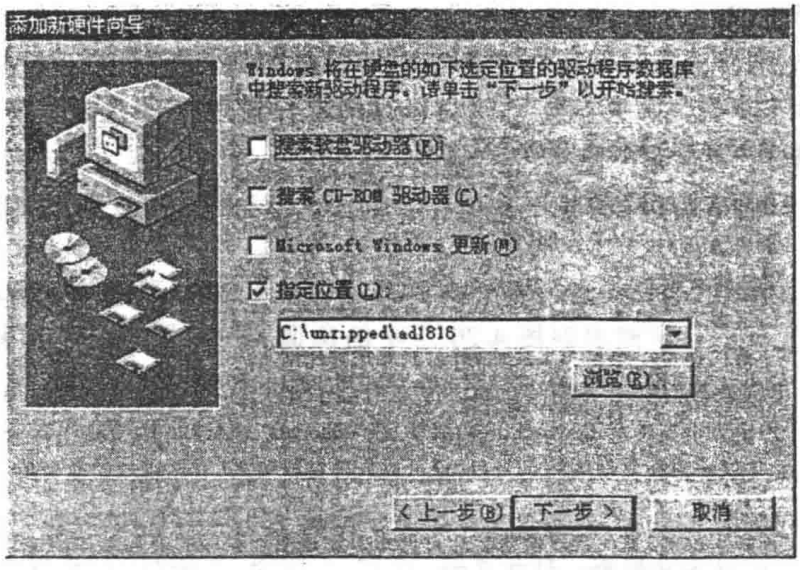

图 4-19　添加新硬件向导之二

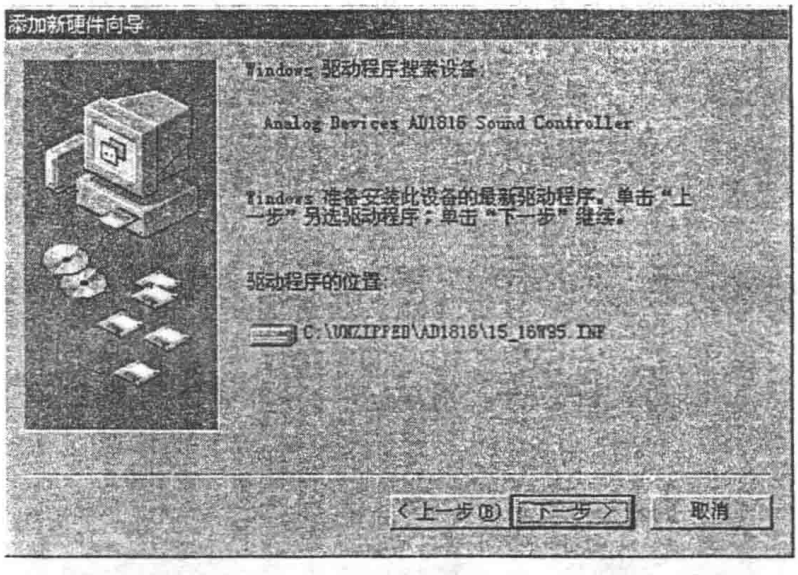

图 4-20　添加新硬件向导之三

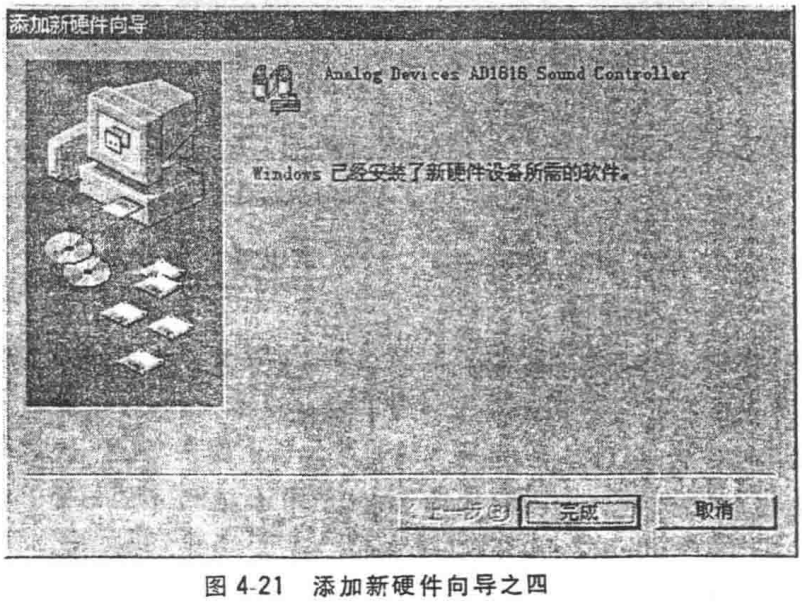

图 4-21　添加新硬件向导之四