

第18学时 个人事务工具程序

在本学时教程中，将介绍一些可以用来帮助你处理个人事务的日历、命令和 X11 客户程序。不管这些程序是用来写一篇日记还是用来制作一份备忘录，你将学习在 OpenLinux 操作系统中如何使用它们这些工具程序来记录并安排日常事务。

在本学时教程中讨论的每一个程序或者技巧都可以使用这本书所附的 CD-ROM 光盘里找到的软件来实现。学习完这些内容之后，会掌握更多使用这些程序来完善日常事务处理所使用的工具和技巧的方法。

18.1 使用 at 命令安排个人日程备忘录和任务

如果需要记录那些重要的日程安排、设置备忘录或者在外出的时候自动运行某些程序，就可以使用 at 命令。可以在子目录 /usr/bin 中找到这个命令，它可以用来安排命令或者任务在你指定的某个时间去运行。举例来说，如果你正在进行某项工作但是需要提醒自己去搭载你的乘车伙伴，就可以从命令行输入一个简捷的 at 任务。



系统在缺省的情况下应该允许你使用 at 命令。如果不是这样，请阅读第24学教程“使用任务计划实现系统管理自动化”中关于如何在系统上激活 at 命令的内容。

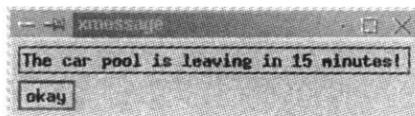
举例来说，如果想在适当的时间在屏幕上弹出一个备忘录，可以使用下面的命令：

```
# at 16:15
at > xmessage -display :0.0 " The car pool is leaving in 15 minutes . "
at > <EOT >
warning : commands will be executed using /bin/sh
job 12 at 1998 - 12 - 18 16 : 15
```

上面的内容告诉 at 命令在指定的 X11 显示窗口中运行 xmessage 程序来显示关于你的搭车伙伴消息的文本内容，如图 18-1 所示。上面内容中的文本结束符（EOT）表示你必须按下 Ctrl+D 组合键关闭命令并提交任务。如果在输入这个命令的时候出现了一个语法错误，那么在到了预定时间的时候就只能收到一个邮件消息。命令行参数 -display 告诉 xmessage 命令应该在哪个屏幕设备上显示你输入的消息，这个设备通常是 0.0，而你可以使用下面的命令找出它到底应该是哪一个屏幕：

```
# printenv | fgrep DISPLAY
```

图18-1 at命令可以把任何OpenLinux程序或者X11客户程序，比如xmessage，当作备忘录提示程序来运行



上面的命令行把 printenv 命令的执行结果经过管道送到 fgrep 命令，从所有的环境变量（在

第6学时教程“使用 shell”中介绍过)中查找出\$DISPLAY环境变量的值。

如果正在使用X11，也可以采用控制桌面颜色的办法使用 at命令提供一个直观形象的备忘录提示，如下所示：

```
# at 16 : 15
at > xsetroot -display : 0.0 -solid red
at > xmessage -display : 0.0 " The car pool is leaving in 15 minutes . "
at > <EOT >
warning : commands will be executed using /bin/sh
job 14 at 1998 - 12 - 18 16 : 15
```

上面的命令在预定时间会把桌面，也就是根显示区，显示成深红色，然后再显示提示信息。正如你所看到的，可以一次使用多个命令同时完成好几件事情。如果觉得这个方法很方便，也可以把这些命令输入到一个名为 carpool的文本文件中去，然后通过使用 -f参数，后面再跟上这个文件的文件名的方法来执行这些命令：

```
# at 16 : 15 -f carpool
warning : commands will be executed using /bin/sh
job 13 at 1998 - 12 - 18 10 : 19
```

at命令会从文本文件中读取这些命令并回显一个确认信息。如果想更方便一点儿，可以把这个命令行放入到用户子目录中的 .xinitrc命令脚本程序中去。这样在每天早晨启动 X11之后，这个任务就会被自动地安排好了。



请阅读第7学时教程“使用X窗口系统”中更多关于如何配置 .xinitrc文件的资料。

使用atq命令可以看到一个已经安排的任务清单，如下所示：

```
# atq
14      1998-12-18 12:00 a
15      1998-12-18 13:00 a
16      1998-12-18 14:00 a
17      1998-12-18 15:00 a
18      1998-12-25 16:00 a
```

上面显示在12月18号已经安排了四件事情，还有一件事情安排在了12月25日。使用at命令安排任务的时候，会在 /var/spool/at子目录中生成一个包含了每一条命令的 shell命令脚本程序。atq命令会在这个子目录中查找你安排的任务，然后把它们显示到显示屏幕上。

可以使用at命令提前几分钟、几小时、几天、几星期甚至几年来安排某个任务。如果打算在三个小时之后运行你的搭车伙伴备忘录提示程序，可以使用 at命令的加号(+)命令行参数，如下所示：

```
# at +3 hours -f carpool
```

这个命令会从当前系统时间开始计时，在三个小时之后运行显示你的事务安排。还可以使用时间和日期(按照mm/dd/yy的格式)来安排任务，如下所示：

```
# at 16 : 15 12/25/00
at > xmessage -display : 0.0 " The car pool is leaving in 15 minutes . "
at > <EOT >
warning : commands will be executed using /bin/sh
job 21 at 2000 - 12 - 31 16 : 15
```

正如所看到的，这个 at 命令已经解决了 2000 年千年虫问题。使用 atrm 命令可以控制管理你的 at 任务。你可以有选择地使用任务号来删除指定的任务。举例来说，我们还是使用前面例子中的任务队列，你可以输入下面的命令：

```
# atrm 16 18
```

这个命令将删除安排在第 16 号和第 18 号的任务，但是会保留另外的两个。使用 at 命令是为某个特定的时刻编写一次性备忘录提示程序的好方法。在下一小节中，将介绍如何按照一定的时间间隔安排其他的任务反复执行。



也可以使用由 Rob Nation 编写的 X11 客户程序 rclock 来安排备忘录提示或者在预定的日期或时间运行程序。如果想建立备忘录提示，需要在用户子目录中建立一个名为 rclock 的文件，然后再输入备忘录提示命令行。比如说：

```
11:30 mtwtf Time for lunch!
```

这个命令行将在整个星期内的每天显示一个提醒午餐时间的提示信息。rclock 客户程序通常是与 X11 的 rxvt 终端程序一起发行的，但是并没有包括在 OpenLinux 操作系统中。可以从下面的站点获得它的最新的版本：

```
http://babyaga.math.fu-berlin.de/~rxvt/.
```

或者也可以在下面的站点从一个 rpm 文件中获得 rxvt 和 rclock 这两个程序：

```
http://metalab.unc.edu/pub/Linux/X11/terms/.
```

如果想获得一份用于 OpenLinux 操作系统和 X11 中的各式各样不同的时钟程序（许多都带有报时功能）的清单，请浏览站点：

```
http://metalab.unc.edu/pub/Linux/X11/clocks.
```

18.2 使用 crontab 命令安排周期性的备忘录提示

虽然 at 命令在安排一次性任务的时候很方便，但是需要以固定的周期完成固定的任务的时候，就可以使用由 Paul Vixie 编写的 crontab 命令来帮助你。crontab 命令（可以在子目录 /usr/bin 中找到它）是用来在一个私人文件中输入设定的预定时间和命令的。

crontab 命令的工作原理是在子目录 /var/spool/cron 中按照用户名来查找 crontab 命令的日程安排。Linux 操作系统中使用的 crontab 文件就叫做 crontab，保存在子目录 /etc 中。而用来管理运行整个系统和各个用户的 cron 日程安排的程序就是 cron 程序包，它在开机启动 Linux 操作系统的时候就被启动了，然后每分钟被唤醒一次并对整个系统和各个用户的文件进行检查。

如果想建立你自己的 crontab 文件，必须使用这个命令的 -e 参数，如下所示：

```
# crontab -e
```



要确定已经为你的系统激活了 crontab 功能。请阅读第 24 学时教程“任务计划”中关于如何进行这个操作的详细资料。建立或编辑你的 crontab 文件的时候，也许还想把最喜欢的文本编辑器程序定义为缺省的 \$EDITOR 环境变量。请阅读第 6 学时教程“使用 shell”中关于如何设置环境变量的说明，以及第 14 学时教程“文字处理程序”中关于使用各种文本编辑器程序的说明。

这个命令将调入在 shell的\$EDITOR环境变量中定义的缺省的文本编辑器程序进入运行，这样就可以建立或者编辑个人专用的 crontab文件。但是在学习 crontab数据项的格式之前，还可以考虑对缺省的编辑器程序进行一些配置。

如果缺省文本编辑器程序是 vi，但是却想使用另外一个不同的编辑器程序，可以使用 shell临时修改一下\$EDITOR变量。举例来说，如果正在使用名为 bash的shell，并且打算使用 pico文本编辑器程序，可以输入下面的内容：

```
# EDITOR = /usr/bin/pico ; export EDITOR
```

这个命令将把缺省的编辑器程序设置为 pico编辑器程序。可以再查看一下你的环境变量来确认这一点，如下所示：

```
# printenv | fgrep EDITOR
EDITOR=/usr/bin/pico
If you've never created a crontab file, you're initially presented an
empty file where you can enter crontab settings, regardless of which
editor you use. Before you can enter your own schedule, you need to know
how to format a crontab request.
```

crontab数据项的格式看起来与表 18-1中的差不多。在不同的数据域中都可以使用周期性数值，比如*/3可以表示每三分钟或者每三个小时进行一次操作。也可以使用时间间隔范围，比如7~10表示在早晨7、8、9、和10点钟的时候执行操作。请注意数字（0或者7用来代表星期日）或者单词（开头的三个字母）可以用来代表月份和星期数据域中的某一个日子）。

表18-1 crontab数据项的格式

分钟	小时	日期	月份	一星期中的某一天
0 ~ 59	0 ~ 23	0 ~ 31	0 ~ 12 (或者名称)	0 ~ 7 (或者名称)

可以在/usr/man/man5子目录中的crontab命令的使用手册页中查到 crontab请求的格式和一些使用的例子。可以用下面的命令来阅读这个使用手册页：

```
# man 5 crontab
```

在crontab文件中可以包含有对那些想定期或者甚至不那么定期地执行的程序的设置，如下所示：

```
***** somecommand
0, 15, 30, 45, ***** somecommand
0 ***** somecommand
```

第一个例子显示想每分钟执行一次某个程序；第二个 crontab数据项每15分钟执行一次某个程序；第三个例子每小时执行一次某个程序，在整点的时候执行。

如果想每天一次在固定的时刻运行某个程序，可以使用下面的命令：

```
30 7 *** somecommand
30 0 *** somecommand
15 16 *** somecommand
```

第一个例子表示在早晨7：30分的时候执行；第二个例子在午夜过了30分钟之后执行；第三个例子在每天的下午4：15都会执行一次。还可以在一个月或者在一个星期中指定的某一天执行某个程序，如下所示：

```
30 16 1 ** somecommand
30 15 ** mon somecommand
```

第一个例子在每个月的第一天的下午 4 : 30 运行；第二个在每个星期一的下午 3 : 30 分运行。总结以上的例子，还可以具体指定是哪一个月，如下所示：

```
30 7 25 12 * somecommand
```

这个例子在每年的12月25号这一天的上午7 : 30分都会执行指定的命令。指定的命令可以是系统工具程序，但是也可以是你自己的 shell 的命令脚本程序。可以使用 crontab 命令的 -l (列清单) 参数来显示 cron 设置，如下所示：

```
# crontab -l
# DO NOT EDIT THIS FILE - edit the master and reinstall.
# (/tmp/crontab.XXXa02274 installed on Fri Dec 18 13:20:49 1998)
# (Cron version -- $Id: crontab.c,v 2.13 1994/01/17 03:20:37
↳vixie Exp $)
0,30 * * * * /usr/local/bin/saytime
0 8 * * * * /usr/local/bin/ppp on
3 8 * * * * /usr/bin/fetchmail -u bball -p mypasswd staffnet.com
```

在上面显示的 crontab 文件中，可以看到有这样几个设置：每半个小时报一次时间、每天早晨8点启动 PPP 连接以及在 PPP 连接启动开始三分钟之后下载当天的邮件。可以使用 crontab 命令的 -r (删除) 参数删掉 crontab 文件，如下所示：

```
# crontab -r
```

使用 crontab 命令是建立、运行和管理周期性任务的简便方法。能够建立自己的备忘录提示之后，也许会想使用一个日历来进行短期或者长期的计划。

18.3 使用 X11 的 ical 客户程序建立约会备忘录提示

可以使用由 Sanjay Ghemawat 编写的 ical 日历程序把约会和备忘录提示保存到一个私人日历中。ical 客户程序 (可以在子目录 /usr/X11R6/bin 中找到这个程序) 与 cal 或者 gcal 日历打印程序相比，有许多独特的功能和改进：

- 用户可以定制设置的带有菜单、对话框、滑动控制部件和按钮的图形化 X11 操作界面
- 可以对便条及约会进行复制粘贴与拖放操作
- 对即将发生的事件能够声音提示
- 多种日历显示方式
- 可以导入、导出以及自动保存日历文件
- 日历文件可以由同一组成员共享
- 能够打印不同的日历
- cron 程序风格的约会、笔记和备忘录安排方式
- 已安排工作检查清单表
- 内建的帮助信息

第一次运行 ical 程序的时候，这个程序会显示一个设置为当天日期的当前月份的日历，并在画面的右半部分显示一页笔记，也就是约会记录清单。启动 ical 客户程序的时候使用它的 -calendar 参数，后面再跟上想建立或者编辑的日历文件的文件名，如下所示：

```
# ical -calendar $HOME/.calendar &
```

这样就启动了 ical 客户程序，而它就会在用户子目录中建立一个名为 .calendar 的日历文件。



在每次启动ical的时候都需要指定要使用哪一个日历文件，除非定义了名为CALENDAR的环境变量指向某个 ical -calendar命令所使用的日历文件。如果这个程序无法打开某个日历文件，它就会反复报警并显示一条出错信息。如果想要定义自己的 CALENDAR环境变量，请使用一个编辑器程序打开 .bashrc文件，在其中加上下面的内容：

```
export CALENDAR = $HOME/.calendar
```

现在保存这个文件。如果想立刻使用这个设置值，请使用 bash shell 的source命令，如下所示：

```
source .bashrc
```

然后再按下回车键。

另外，ical客户程序还可以使用许多标准 X11的Toolkit工具参数，因此就可以把 ical客户程序作为一个图标来启动运行。如下所示：

```
# ical -iconic
```

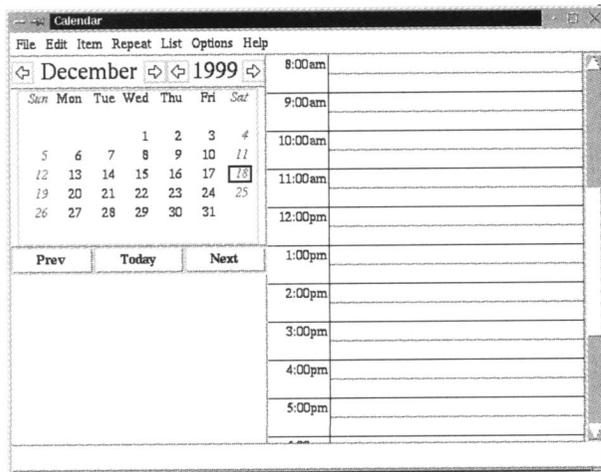
还可以改变窗口尺寸参数的设置值来设定日历最初开始时的大小，如下所示：

```
# ical -geometry 800 x 600
```

图18-2就是这个ical客户程序的画面。

图18-2 X11的ical客户程序

具有通告、约会、
已安排工作清单以
及备忘录提示报警
等功能



想建立通告的时候，需要先选定一个日子，然后在日历的下方单击鼠标并在其中输入文字内容。想记录某一天的约会的时候，需要先单击选定一个时间，然后再输入这个约会的名称。可以使用按住鼠标中键（当你使用的是双键鼠标的时候需要同时按下鼠标的左、右两键）的方法把这个约会拖放到这一天的任何位置上，这样就可以对日程重新进行安排。

设置好通告或者约会之后，还可以设置一个报时钟，让 ical客户程序在事件即将发生之前向你发出提示音。如果想设置事件的声音提示功能，先单击这个约会，然后再选择“Item”（项目）菜单中的“Properties”（属性）菜单项为这个约会设置一个报时钟。可以在某个约会上双击鼠标进入报时钟对话框。

可以设置ical在事件发生之前一到十五天提醒你，在某个约会到来之前的一个小时内最多

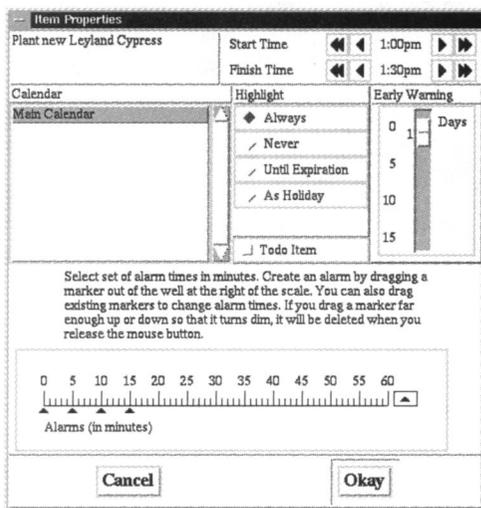
可以有60次报警提示。当最终到达预定时刻的时候，ical程序的提示通知窗口会以备忘录提醒的形式弹出来。



只有在 ical 客户程序运行的时候才能收到声音提示。因此可以把 ical 命令和它的 -popup 命令行参数放在一个 crontab 数据项中。如果使用了 -popup 参数，ical 会在一个窗口中列出所有当天的约会直到按下“OK”按钮之后才退出。

图18-3显示了ical客户程序的提醒对话框。

图18-3 ical客户程序的提醒对话框提供了多个阶梯状排列带有可拖放控件的报警钟



如果选定了某个约会或者通知，可以通过选择相应的“Repeat”（重复）菜单项把它设置为能够在每天、每星期、每月甚至每年反复循环地执行。把某个记录项设置为重复执行的时候，它将会自动复制到日程序中的各个对应位置。

如果想把某个约会设置为必须确认完成的操作，需要单击这个约会，再从ical程序的“Item”（项目）菜单选择“Todo Item”（确认完成）菜单项。在这个约会的提示内容的前面会出现一个小框。只有当你使用鼠标左键单击这个小框并在小框里放上一个表示确认完成的选中记号之后，这个项目才算完成；否则这个需要确认完成的约会记录项就会出现第二天的约会清单中。

可以使用ical程序的“List”（列清单）菜单列出约会和通告。如果想得到日历的一份硬拷贝，可以选择按照六种不同的内建日历格式进行打印，还可以选择打印某个日期范围。开始打印之前，还可以对日历进行预览。

ical客户程序非常便于组织个人或者集体任务。可以查看一下保存在子目录/usr/lib/ical/contrib中的一些与ical程序一起发行的shell命令脚本和程序，到那里去寻找介绍怎样对ical程序进行定制设置才能满足自己工作需要的窍门与提示。



可以在下面的站点找到 ical 程序的最新版本 (v2.2)：

http://www.research.digital.com/SRC/personal/Sanjay_Ghemawat/ical/home.html

18.4 使用emacs查看日历和管理约会

我们在第14学时教程“文字处理程序”中已经比较全面地介绍过 emacs文本编辑器程序，它具有许多能够帮助处理个人事务甚至提供娱乐的功能。可以查看当前日历、查看节假日清单（还有更多）还可以使用约会备忘录写日记。

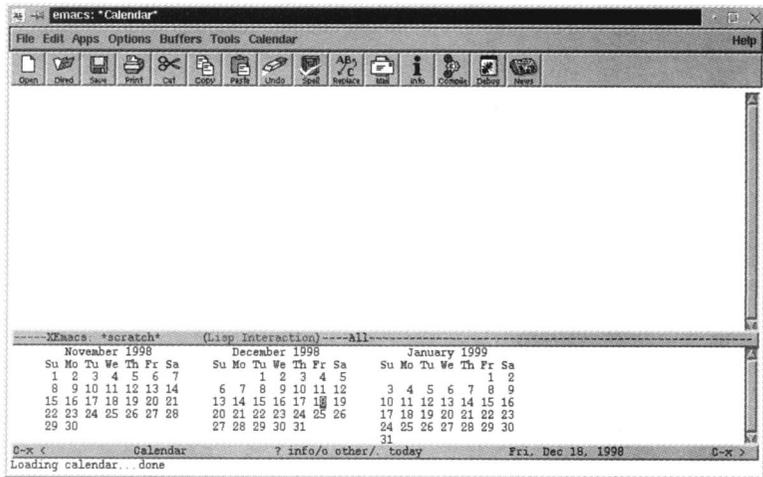
如果正在使用 X11，那么除非在命令行特意使用 `-nw` 命令行参数来指定必须在终端窗口中运行 emacs，否则就会自动运行 emacs 程序的 X11 版本。如下所示：

```
# emacs -nw
```

当启动 emacs 程序的 X11 版本运行之后，从主菜单开始单击“Apps”（应用程序），再单击“Calendar”（日历）然后再单击选择“3 Month Calendar”（三个月日历），就可以看到包括前一个月、本月和下一个月三个月份的日历。emacs 编辑器程序还可以提供日记、节假日、月亮盈亏周期以及日出/日落时间。也可以使用非 X11 版本的 emacs 的 `Esc-x-calendar` 命令。按住键盘上的 Esc 键，输入一个字母 x，然后再输入单词“calendar”（日历），最后按下回车键。emacs 程序将显示一个如图 18-4 所示的三个月份的日历。

图18-4 emacs编辑器程序的

特色包括有一个内建的日历功能，提供一整套包括一个日记和约会备忘录提示功能的日历工具



在 emacs 程序的日历窗口中单击鼠标之后，再输入一个问号（？）就可以获得使用 emacs 日历工具程序的帮助信息。屏幕上会出现一个关于日历数据项的帮助菜单。在 emacs 中启动运行了日历状态之后，在 emacs 程序的菜单条上就会出现一个“Calendar”（日历）菜单项。

使用“Calendar”（日历）菜单项中的“Go to”（去往）子菜单可以在日历中前进或着后退，还可以进入将近二十种不同类型的日历，比如雅利安人的日历等等，甚至还有玛雅人的日历！“Holidays”（节假日）子菜单可以根据屏幕底部显示的日历给出某个时期内、某些天或某几个月内的所有节假日；也可以在日历列表中标出节假日的具体日期。“Moon”（月相）子菜单列出月亮盈亏周期的各个阶段。



在 emacs 程序中有超过一百个不同的日历命令（与那些明显地出现在菜单中的菜单项不一样）。如果想收到约会备忘录提示，在那个时候就必须运行着 emacs 程序。如果想了解更多的资料，可以先看看它的日历功能本身的帮助信息，然后再从 emacs 程序的“Help”（帮助）菜单中使用 `apropos` 命令获取更详细的信息。

还可以通过 Neil Mager 为 emacs 程序编写的约会函数集让 emacs 程序提醒那些重要的事件。如果想设置约会备忘录提示，首先需要在自己用户子目录中建立一个包含下列内容的 .emacs 文件：

```
(require 'appt)
(display - time)
(appt - initialize)
```

使用“File”(文件)菜单保存这个文件。退出，然后再重新运行 emacs 程序。如果想建立一个 emacs 日记，先进入 emacs 程序的日历状态。单击“Apps”(应用程序)、再单击“Calendar”(日历)菜单项、最后单击选择“3 Month Calendar”(三个月日历)菜单项。当激活 emacs 的日历状态之后，在日历窗口中的某个特定日子上单击鼠标。然后转到“Calendar”(日历)菜单，单击“Diary”(日记)子菜单，再选择“Insert Diary”(插入日记)。就把输入缓冲区转到了“日记”区，并且会发现跟在文本内容后面的光标中包含着选定的日期。输入一些日记内容之后，保存这个缓冲区(通过“File”(文件)菜单)，然后退出并重新启动 emacs 程序。



emacs 程序的约会函数集是用 LISP 编程语言 (emacs 程序的大部分和它的许多功能函数使用的都是 LISP 语言) 编写的。通过阅读 emacs 约会函数集的源代码，可以学习到更多关于设置约会的参数用法。方法如下所示：

```
less /usr/lib/xemacs-19.16/lisp/calendar/appt.el
```

输入约会

重新启动 emacs 程序之后，可以用 emacs appt-add 命令在日记中输入约会备忘录提示。首先，进入到日历状态，在某个特定日子上单击鼠标左键。然后，按下再放开键盘上的 Esc 键，敲入一个 x，然后再输入 appt-add，再按下回车键。在 emacs 窗口底部的命令行上会提示输入约会的时间是几点几分，如下所示：

```
Time (hh : mm [ am / pm ]):
```

按照一个 24 小时时钟 (比如 15 : 00) 或者一个 12 小时时钟 (比如 3 : 00 pm) 的形式输入时间。按下回车键之后，会提示输入提示消息，如下所示：

```
Message :
```

输入一个备忘录提示消息 (比如 “Wake up ! (起床)”)，然后再按下回车键。

到了预定约会时间之前五分钟的时候，emacs 程序会发出蜂鸣声并显示一个新的状态行，这样就把这个约会通知给你了。两分钟之后，上述过程将会再次重复出现。还可以把备忘录提示手动添加到日记当中，如下所示：

```
8 : 00 am Jogging with dog
12 : 00 am Lunch with Cathy
15:00 pm Check with car repair shop
```

可以在日记文件中保存这些跟在每个日期后面的备忘录提示信息。如果想删除约会，可以在选择某个日期之后使用 appt-delete 命令。将通过交互式的问答从日记中删除各种约会。

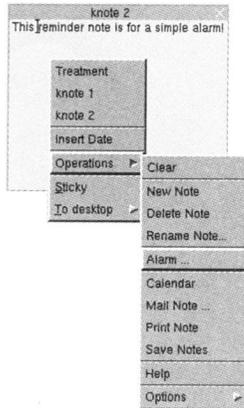
通过使用 emacs 的日历状态和约会函数集的方法，可以在私人日记中建立一个约会的历史记录。请练习使用不同的状态，并阅读 emacs 程序的 info 文件了解更多的信息。

18.5 使用knote客户程序设置报时钟

使用KDE桌面环境的时候，不需要为设置报时钟而运行大型的程序或者文本编辑器程序。如果想在X任务操作过程当中使用一个简单易用的工具程序设置备忘录或者报时钟的话，可以试试包括在KDE工具程序包发行版本中的knote客户程序。

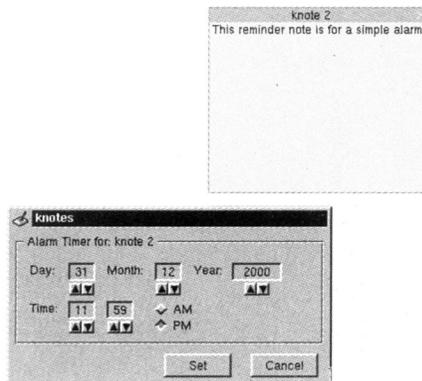
单击KDE桌面控制条上的“Utilities”(工具)菜单并选择knote。在桌面上会出现一个很像Post-It™(注：美国的一种明信片式样)的小窗口。输入一些文字，然后在这个knote窗口中单击鼠标右键。从弹出的菜单中找到并选择“Operation”(操作)菜单，然后再单击“Alarm”(报时钟)菜单项，如图18-5所示。

图18-5 knote客户程序除了简单易用的笔记本功能之外，还有报时钟和备忘录提示功能



屏幕上会出现一个报时钟对话框，如图18-6所示。单击年、月、日小框下面的上下箭头设置报时的日期。单击“AM”(上午)或者“PM”(下午)，然后再选择报时的小时和分钟。完成的时候，单击“Set”(设定)按钮就设置好报时钟了。

图18-6 使用报时钟对话框对knote程序的报时钟功能进行设置是一个简单的任务



18.6 课时小结

本学时教程介绍一些个人事务处理程序和X11客户程序，可以使用它们来安排时间、设置约会以及记录一天的工作。

18.7 专家答疑

问：我把光标放在那个三个月日历中的某个日期上，但是当我想插入一个日记数据的时候

候，emacs程序报告说“not on a date”(没有在日期上)。

答：试试在日历画面中双击那个日期。当某个日期被正确选择的时候，你会在“Diary”(日记)子菜单中看到一个完整的日记项目清单。

问：我想使用crontab和它的-e参数，但是屏幕画面全部消失了，然后沿着屏幕的左面显示了一溜上撇号(‘)，这是怎么一回事？

答：在缺省的情况下，crontab命令使用vi编辑器程序。请阅读vim的使用手册页了解更多关于使用这个编辑器程序的资料。另外，可以建立和使用自己的\$EDITOR环境变量指向pico编辑器程序(包括在pine电子邮件程序中)。

问：我定义了\$EDITOR环境变量，但是当我使用crontab和它的-e参数的时候，程序出了一个错并退出了。这是怎么一回事

答：请确认你已经从OpenLinux操作系统的CD-ROM光盘上安装了pine电子邮件程序，或者已经在\$EDITOR环境变量中包括了pine程序的完整的路径名。shell报告出错，当然crontab会因为这个错退出了。

18.8 练习题

1. 请看看下面的crontab数据项：

```
15 0 4 * * echo "What day is it?"
```

这个数据项会在什么时间以什么方式执行？

2. 请试试在emacs程序中设置几个约会。当某个约会临近的时候，emacs程序会做出什么样的反应？如果设置了一个约会之后立刻退出了emacs程序，会发生什么事情？

3. 试试使用crontab命令设置一个报时钟或者约会。然后再试试使用at命令设置一个报时钟或者约会。哪一个命令更好用一些？为什么？