

第4章 桌面环境

本章将带领读者熟悉一下 Linux 的桌面环境，这里仍然以 Ubuntu 8.04 为例。使用其他发行版本的用户可能会发现具体操作完全不同，但是没有关系，读者需要做的无非是在“另一个地方”找到这些工具，或者是这些工具的等效替代品。Linux 桌面环境如今变得越来越华丽，越来越人性化，即便是第一次使用 Linux 的用户也可以像模像样地做些事情了。

4.1 快速熟悉你的工作环境

本节介绍第一次使用 Linux 必须要知道的事情。如何运行应用程序？如何浏览硬盘？如何建立一个文本文件？读者可能早就知道了这些事情，那么尽管跳过这一节。这些原本是 Windows 教程比较关心的内容。

4.1.1 运行应用程序

在 Ubuntu 中运行应用程序和读者想象得一样简单（或许更简单？），所有的应用程序都被安放桌面左上角的“应用程序”下拉菜单中。在这个下拉菜单中划分了多个类别，图形化应用程序在安装时一般都会遵循这个分类把自己放在相应的目录中。例如读者可以依次选择“应用程序”|“办公”|“OpenOffice.org 文字处理”命令打开这个文字处理软件，如图 4.1 所示。这一套办公软件将在 19.1 节详细讨论。

4.1.2 浏览文件系统

可以使用一个类似于 Windows 中“资源管理器”的工具浏览整个硬盘。依次单击“位置”|“计算机”命令能够看到当前计算机中所有的存储设备及分区，如图 4.2 所示。双击相应的图标可以进入该目录。用户也可以在窗口上部的“位置”一栏中输入具体的路径名来访问，注意在 Linux 中路径的分隔符是正斜杠“/”而不是反斜杠“\”。用户自己的主目录存放在/home 下以用户名命名的目录中。

Linux 中的文件系统结构和 Windows 非常不同，读者暂时可以不必理会，在 6.1 节会详细介绍。

4.1.3 创建一个文本文件

有多种方法可以创建一个文本文件。依次选择“应用程序”|“附件”|“文本编辑器”

命令可以打开一个文本编辑工具，如图 4.3 所示。Ubuntu 附带的这个编辑器叫做 gedit，看上去这个工具和 Windows 下的记事本非常相似，但事实上它的功能要强大得多。gedit 支持拼写检查、文本加密以及编程需要的语法加亮、自动缩进、行号显示等常用功能。图 4.4 是打开一个 C 语言程序显示的效果。

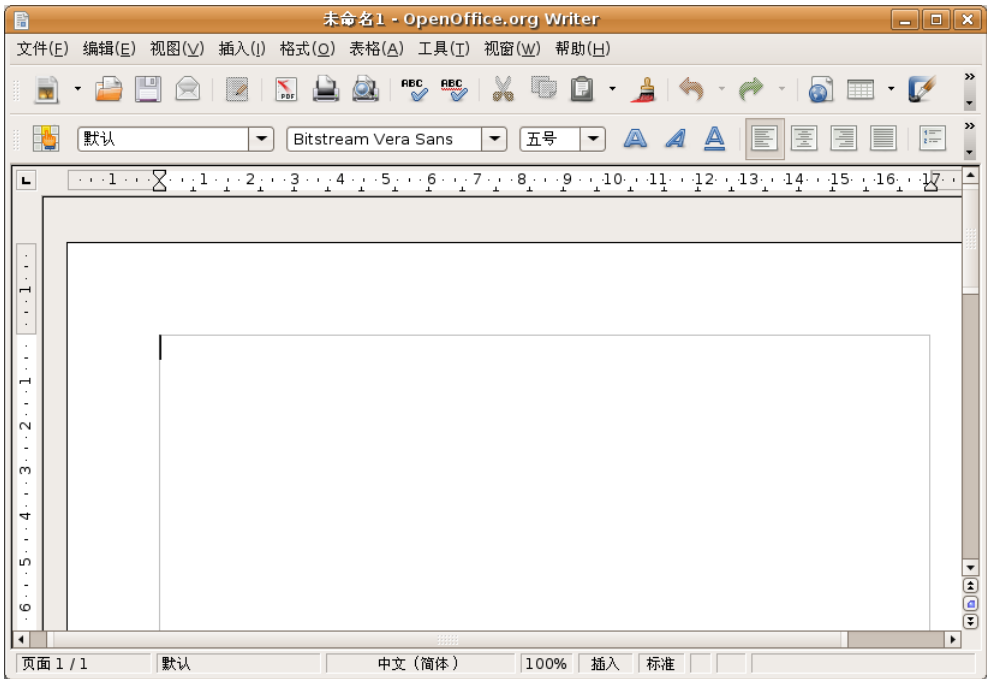


图 4.1 OpenOffice 的用户界面

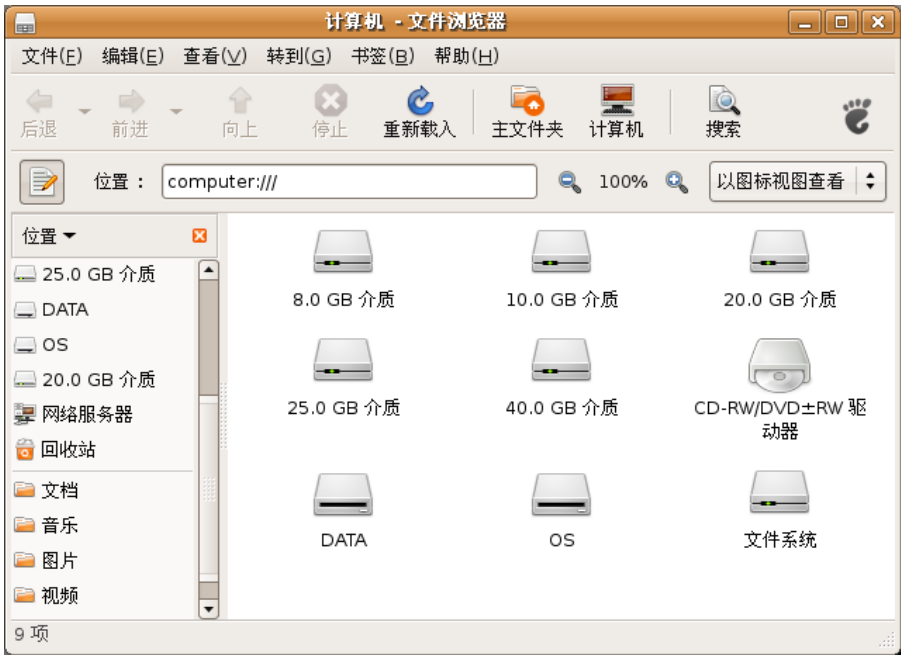


图 4.2 文件浏览器



图 4.3 gedit 的用户界面

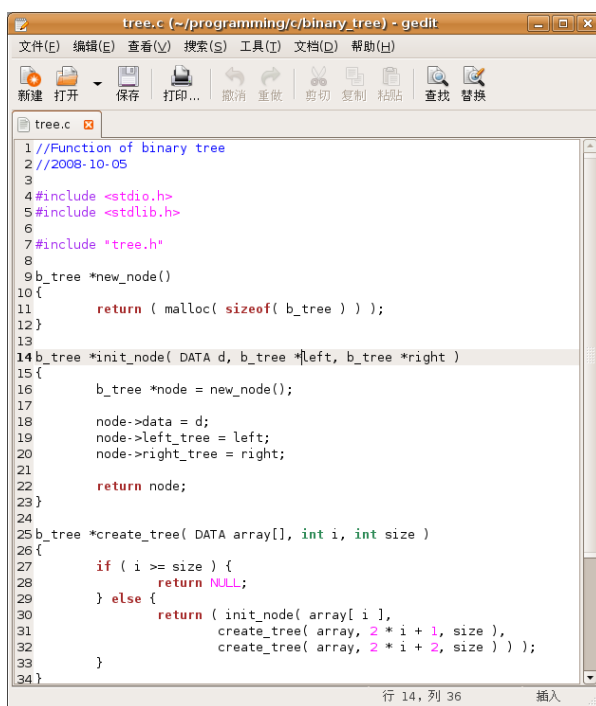


图 4.4 使用 gedit 打开一个 C 程序文件

完成一个文件的编辑后，依次选择“文件”|“保存”命令或者工具栏上的“保存”按钮可以打开“另存为”对话框，如图 4.5 所示。用户可在这里定位到相应的位置，在“名称”文本框中输入文件名，然后单击“保存”按钮。



图 4.5 gedit 的“另存为”对话框

用户还可以在下方的“字符编码”下拉列表框中选择合适的编码，gedit 默认使用 UTF-8 编码。编码选择在数据库（例如 MySQL）录入数据时显得尤为重要，并且已经成为了在 Web 开发中必须考虑的一个问题。对这方面内容的讨论已经超出了本书的范围，有兴趣的读者可以参考其他资料。

4.2 个性化设置

本节介绍 Ubuntu 桌面环境的个性化设置。大部分的设置均针对当前用户，因此不需要提供 root 口令。而涉及系统设置的部分，则要求拥有管理员权限。可以用一个简单的方

法判断是否需要 root 权限，即位于“首选项”菜单中的所有设置均不需要 root 口令，与之相反的是“系统管理”菜单中的大部分命令。

4.2.1 桌面背景和字体

右击桌面，在弹出的快捷菜单中选择“更改桌面背景”选项可以打开“外观首选项”对话框。选择“背景”标签进入“背景”选项卡。在其中可以看到当前能够使用的桌面背景图片，如图 4.6 所示。单击相应的图片可以更改桌面背景，注意这个对话框并没有提供“确定”按钮，所做的选择会即时反应在桌面上。



图 4.6 修改桌面背景

如果希望选择自己的壁纸图片，那么可以单击“添加”按钮打开“添加壁纸”对话框，如图 4.7 所示。定位到相应的图片文件，单击“打开”按钮把图片添加到“壁纸”列表框中。在选择图片的时候，效果图会即时显示在“添加壁纸”对话框的右侧。



图 4.7 添加壁纸

在“字体”选项卡中可以对各种屏幕元素的字体进行更改，如图 4.8 所示。单击相应元素后的按钮，打开“拾取字体”对话框，如图 4.9 所示。在这里面可以选择系统中已经

安装的字体、字体样式和大小。在下方的“预览”框中可以随时看到字体效果。单击“确定”按钮使改动生效。



图 4.8 修改外观字体



图 4.9 设置字体

4.2.2 屏幕保护程序

依次选择“系统”|“首选项”|“屏幕保护程序”命令打开“屏幕保护程序首选项”对话框，如图 4.10 所示。要开启屏幕保护功能，应该确保在“计算机空闲时激活屏幕保护程序”复选框中打钩。在左侧的“主题”列表框中选择相应的屏保主题，就可以在右侧的预览框中看到效果。Ubuntu 附带了很多屏幕保护主题，其中一些主题需要 3D 显卡支持。单击下方的“预览”按钮可以全屏预览。



图 4.10 设置屏幕保护程序

通过调整滑块位置，用户可以设定在多少时间后启动屏幕保护。如果读者希望在屏幕保护启动后自动锁定屏幕（通过输入用户口令解锁），那么还应该选中“屏幕保护程序激活时锁定屏幕”复选框。

4.2.3 显示器分辨率

依次选择“系统”|“首选项”|“屏幕分辨率”命令，打开“监视器分辨率设置”对话框，如图 4.11 所示。系统会根据显示器的实际情况列出可供选择的分辨率和刷新频率数值。建议读者不要随便更改分辨率设置，Ubuntu 的这个小工具有时候运行得不太稳定，并且在大部分情况下，修改屏幕分辨率并没有什么必要。



图 4.11 设置显示器分辨率

4.2.4 代理服务器

如果读者的计算机需要通过内网的代理服务器连接到互联网，那么就应该让系统知道这台服务器。依次选择“系统”|“首选项”|“网络代理”命令打开“网络代理首选项”对话框，如图 4.12 所示。依次填写各个文本框就可以了，当然用户应该在设置之前了解清楚代理服务器的地址和开放端口。如果代理服务器要求进行身份验证，那么可以单击“更多设置”按钮，在弹出的对话框中选中“需要身份验证”复选框，并依次填写用户名和密码，如图 4.13 所示。

在“高级配置”选项卡中可以设置哪些地址不需要通过代理服务器访问，如图 4.14 所示。通常来说，本地主机（localhost）和同一个网络中的主机应该被设置。当然，用户也可以选择让所有的数据包都通过代理服务器，但这并不是一个好主意，因为这会无谓地加重代理服务器的负担，并且一旦代理服务器出现故障，那么用户连内网都访问不了了。如

果要添加其他主机，在“忽略主机列表”文本框中输入主机地址，并单击“添加”按钮即可。如果要删除一台主机，在列表框中选定后单击“删除”按钮即可。



图 4.12 设置网络代理



图 4.13 设置代理身份验证



图 4.14 设置忽略主机

4.2.5 鼠标和键盘

Ubuntu 可以对鼠标和键盘的按键进行设置。依次选择“系统”|“首选项”|“键盘”命令可以打开“键盘首选项”对话框，如图 4.15 所示。读者可以通过尝试各个选项卡，把自己的键盘调整到最舒服的状态。如果需要使用“鼠标键”功能，那么用户的键盘应该要包含数字小键盘。用户可以在下方的文本框中随时测试调整效果。

相应地，依次选择“系统”|“首选项”|“鼠标”命令可以打开“鼠标首选项”对话框，如图 4.16 所示。对于“左撇子”而言，将鼠标方向设置为左手是必要的。同键盘一样，这里也使用滑块来改变灵敏度。通过右下方的电灯泡图片可以测试鼠标设置。



图 4.15 设置键盘首选项



图 4.16 设置鼠标首选项

如果读者正在使用笔记本的话，在“触摸板”选项卡中可以选择是否开启触摸板。有些时候，这块板会因为太过灵敏而影响到敲击键盘。

4.2.6 键盘快捷键

使用键盘快捷键往往可以提高工作效率。依次选择“系统”|“首选项”|“键盘快捷键”命令打开“键盘快捷键”对话框，如图 4.17 所示。在这里可以看到所有能够被设置快捷键的命令，其中的一些（例如注销）已经做过设置了。



图 4.17 设置键盘快捷键

用户也可以自己添加、更改和删除某个快捷键。在相应的行上单击，系统会提示用户输入快捷键，此时直接使用键盘上相应的按键即可完成修改（例如要设置快捷键

Ctrl+Alt+R，那么应该依次按住键盘上的 Ctrl 键、Alt 键和 R 键，然后同时松手）。要删除某个快捷键，那么只要在提示的时候按下 Backspace 键就可以了。

4.2.7 登录界面

依次选择“系统”|“系统管理”|“登录窗口”命令可以打开“登录窗口首选项”对话框（为此需要提供 root 口令），如图 4.18 所示。用户可以在这里选择不同的登录主题；也可以“辅助功能”选项卡中为不同的登录阶段选择音乐。用户也可以选择自动登录系统，甚至可以选择让某个用户在登录前暂停一段时间。其中涉及的一些概念，如远程登录将在后续章节中介绍。



图 4.18 设置登录窗口主题

4.3 进阶 1：究竟什么是“桌面”

Linux 中“桌面”的概念在初学者看来只能用“乱七八糟”来形容，好在那些试图解释清楚这件事情的人们也有同样的感受。这一节的内容有一点枯燥，更令人沮丧的是，读者可能在很长一段时间内都不会用到这些概念。

4.3.1 可以卸载的图形环境

这句话在 Windows 专家们看来简直是不可思议的。“那我们还怎么工作？”，他们会

这样问。Linux 不是一种基于图形环境的操作系统，40 年前的 UNIX 用户可以在命令行下完成所有的工作，现在仍然可以。在内核眼里，图形环境只是一个普通的应用程序，和其他服务器程序（如 Apache、NFS 等）没有什么不同。

如果 Linux 发行版本的安装程序允许用户自己定制安装软件的话，那么从一开始就可以选择不要图形环境（参见 2.2.3 节），这样 Linux 启动后会把用户带至命令行。Linux 的命令将在后面的章节陆续介绍。

4.3.2 X 窗口系统的基本组成

X 窗口系统（X Window System）是 Linux 图形用户环境的基础。这个系统最初诞生于 MIT（麻省理工大学）的 Athena 项目，时间是 20 世纪 80 年代。X 的发展经历了一段复杂曲折的过程，如今绝大多数 Linux 使用的是由 X.org 基金会维护的 X.Org（曾经被广泛使用的 XFree86 因为许可证的转变正逐渐退出 Linux 市场）。

X 系统基于一种独特的服务器/客户机架构。作为起步，本节首先解释几个基本概念。这些概念现在看起来可能有点抽象，这样安排的用意是，如果读者被后面的内容弄糊涂了，那么还可以回到这里寻求帮助。

1. X 服务器

X 服务器用于实际控制输入设备（例如鼠标和键盘）和位图式输出设备（例如显示器）。准确地说，X 服务器定义了给 X 客户机使用这些设备的抽象接口。和大部分人的想法不同，X 服务器没有定义高级实体的编程接口，这意味着它不能理解“画一个按钮”这样的语句，而必须告诉它：“嗯……画一个方块，这个方块周围要有阴影，当用户按下鼠标左键的时候，这些阴影应该消失……对了，这个方块上还应该写一些字……”

这种设计的意义在于，X 服务器能够做到最大程度上的与平台无关。用户可以自由选择窗口管理器和 widget 库来定制自己的桌面，而不需要改变窗口系统的底层配置。

2. X 客户端程序

需要向 X 服务器请求服务的程序就是 X 客户端程序。具体来说，OpenOffice、gedit 这些应用程序都是 X 客户端程序，它们运行时需要把自己的“长相”描述给 X 服务器，然后由 X 服务器负责在显示器上绘制这些应用程序的界面。


3. 窗口管理器（Window Manager）

窗口管理器负责控制应用程序窗口的各种行为，例如移动、缩放、最大化和最小化窗口，在多个窗口间切换等。从本质上来说，窗口管理器是一种特殊的 X 客户端程序，因为这些功能也都是通过向 X 服务器发送指令实现的。Window Maker、FVWM、IceWM、Sawfish 等是目前比较常见的窗口管理器。

4. 显示管理器（Display Manager）

显示管理器提供了一个登录界面，其任务就是验证用户的身份，让用户登录到系统。可以说，图形界面的一切（除了它自己）都是由这个显示管理器启动的，包括 X 服务器。

用户也可以选择关闭显示管理器，这样就必须通过命令行运行 `startx` 命令（或者使用 `.login` 脚本）来启动 X 服务器。

 **提示：**这里所说的“脚本”是指 Shell 脚本，它是一段能够被 Linux 理解的程序。这部分内容将在第 21 章详细讨论。

5. widget 库

widget 库定义了一套图形用户界面的编程接口。应用程序开发人员通过调用 widget 库来实现具体的用户界面，如按钮、菜单、滚动条、文本框等。程序员不需要理解 X 服务器的语言，widget 库会把“画一个按钮”这句话翻译成 X 服务器能够理解的表述方式。

6. 桌面环境

现在终于到了问题的关键，究竟什么是桌面环境？以 KDE 和 Gnome 为代表的 Linux 桌面环境是把各种与 X 有关的东西（除了 X 服务器）整合在一起的大杂烩，这些程序包括像 gedit 这样的普通应用软件、窗口管理器、显示管理器和 widget 库。但无论桌面环境如何复杂，最后处理图形输出的仍然是 X 服务器。这一点在后面的讨论中还会用到，千万不要搞反了。

4.3.3 X 系统的启动过程

X 系统的启动过程基本是由显示管理器（Display Manager）完成的。显示管理器启动后依次完成下面这些工作。

- ☐ 启动 X 服务器。
- ☐ 提供一个界面友好的屏幕，等待验证用户的身份。
- ☐ 执行用户的引导脚本，这个脚本用于建立用户的桌面环境。

简单提一下“引导脚本”——尽管到现在为止还没有正式接触“脚本”这个概念。桌面环境的引导脚本是一段用 Linux 命令组成的脚本程序，叫做 Xsession。Xsession 通过启动窗口管理器、任务栏，设定应用的默认值、安装标准键绑定等来启动整个桌面环境。KDE 和 Gnome 都有自己的启动脚本，这些通常不需要用户操心。

Xsession 会一直运行，直到用户退出（或者说，当 Xsession 运行结束后，用户就退出了）。窗口管理器（Window Manager）是 Xsession 启动的唯一的后台程序（其他程序都在后台执行），如果没有这个前台程序，那么用户会在登录后又立即退出系统。关于前、后台执行程序的区别，可参考 5.8 节的内容。

4.3.4 启动 X 应用程序

X 窗口的服务器/客户机架构意味着一台主机上的 X 应用程序可以在另一台主机的屏幕上显示出来。X 服务器接受来自多个应用程序的请求，然后在本地显示。而这些应用程序可能正运行在网络中的另一台主机上。

也就是说，为了运行一个 X 应用程序，必须指定在什么地方显示。环境变量 `DISPLAY`

定义了这些内容（环境变量用于在系统运行时保存一些同系统和用户相关的信息，详见 21.3.1 节）。下面给出了一个 `DISPLAY` 变量的典型设置：

```
DISPLAY=servername:3.2
```

当 X 应用程序启动时，它会查看这个环境变量。在上面这个例子中，X 应用程序把自己的图形输出到主机 `servername` 上的显示 3 和屏幕 2 上。

“显示 3 和屏幕 2”这个短语有点难懂。如果一台主机只运行一个 X 服务器，那么这个 X 服务器就工作在端口 6000，对应的显示号是 0；如果再安装一个 X 服务器程序，那么这个新的 X 服务器会工作在端口 6001，对应的显示号是 1……依次类推。至于“屏幕 2”，说的是在一台主机上连接有多台显示器的情况下，显示器也从 0 开始编号。第 1 台显示器标识为“屏幕 0”，因此“屏幕 2”就是这台主机所连接的第 3 台显示器。

由于大部分主机只运行一个 X 服务器、连接一台显示器，因此大部分情况下，环境变量 `DISPLAY` 的值会像下面这样：

```
servername:0.0
```

现在再反过来考虑最常见的情况——X 客户机（X 应用程序）向本地的 X 服务器传递图形输出，X 服务器在本地的显示器上显示图形。此时就不再需要指定服务器名了，环境变量 `DISPLAY` 的值相应地退化为下面这样：

```
:0.0
```

由于屏幕号也可以省略（默认屏幕号为 0），因此在最简单的情况下，`DISPLAY` 变量的值只是一个“:0”。

4.3.5 桌面环境：KDE 和 Gnome 谁更好

在开源世界，凡是涉及“什么和什么谁更好？”这一类的问题，总能引起一片硝烟弥漫，并且每一方的论据都很有说服力。这样一来，提问者最后总能学会如何“辩证”地看待问题。下面这些文字就是站在中立的立场上给出的建议。

KDE（K 桌面环境）是用 C++ 编写的，基于 Qt 库。这是刚从 Windows 或者 Mac 转过来的用户比较偏爱的桌面环境，因为它确实比较漂亮，在使用习惯上也同 Windows 比较接近。对于热衷于定制桌面的用户而言，KDE 可能是最好的选择。

为 KDE 编写的应用程序总是带着一个字母 K，如 Konqueror（文件浏览器）、Konsole（命令行终端）等。KDE 为程序员提供了一套功能完备的开发工具，包括一个集成开发环境（IDE），这使得程序员很容易在 KDE 上开发风格统一的应用程序

Gnome 是用 C 语言写成的，基于 GTK+widget 库。这个桌面环境最初就是为了对抗 KDE 而诞生的——这是另一个“自由”对抗“非自由”的故事。相对于 KDE 而言，Gnome 看上去不那么讨人喜欢，它有点严肃，好像总是板着一张脸。但 Gnome 的确更快速和简洁，因为在有些人看来，KDE 有点太啰嗦了。

同 KDE 类似，Gnome 应用程序大多带着一个字母 G，如 GIMP（图形处理软件）、gftp（FTP 工具）等。同样地，Gnome 也为开发人员提供了一套易于使用的开发工具。

究竟是使用 Gnome 还是 KDE，取决于在性能和外观之间的权衡。或者有些时候，这

仅仅是个人口味的差异。人们用舌头而不是逻辑来评判美食，家人、朋友、老板等，对食物和桌面环境都有不同的偏好，这很正常。

4.4 进阶 2：配置 X 服务器

X 服务器在默认情况下已经配置得很好了，如果用户的 X 系统工作得不错，那就没有必要手动修改这些设置。使用图形化的配置工具能够减少设置 X 服务器的风险，但有些时候，不正确的设置（例如使用了错误的显卡驱动）可能会让读者不得不在命令行下工作。如果碰到了这样的情况，那么本节所讨论的内容或许能帮上忙。

4.4.1 理解配置文件

Windows 用户从现在起就需要转变观念——Linux 中并没有类似于“注册表”这样的二进制表项，所有软件的配置都是通过文本文件实现的。这样的安排看起来有点“原始”，但读者很快会发现设计者的良苦用心。不同的软件会定义属于自己的语法规则，这些配置语法间的区别总是让人联想到人类语言的丰富，而不是复杂或是晦涩难懂。

X 服务器的配置文件是 `/etc/X11/xorg.conf`。读者可以使用自己喜欢的文本编辑器（如 `gedit` 和 `kate`）打开这个文件看一眼。

```
# xorg.conf (X.Org X Window System server configuration file)
#
# This file was generated by dexconf, the Debian X Configuration tool, using
# values from the debconf database.
...
Section "InputDevice"
    Identifier "Generic Keyboard"
    Driver      "kbd"
    Option      "XkbRules"    "xorg"
    Option      "XkbModel"    "pc105"
    Option      "XkbLayout"   "cn"
EndSection
...
Section "Module"
    Load       "glx"
EndSection
```

所有以“#”开头的行是注释行。注释是为了让用户更容易地阅读和修改配置文件而写的，应用软件会忽略配置文件中所有的注释行。正如读者看到的，`xorg.conf` 的内容被分成几段，每一段以关键字 `Section` 开头，以 `EndSection` 结尾。每一段负责完成一个特定的任务，表 4.1 给出了各段的说明。


表 4.1 xorg.conf 各配置段及其说明

配置段	说明
ServerFlags	X 服务器的全局参数设置
InputDevice	配置输入设备，如鼠标、键盘、触摸板等
Device	配置显卡和驱动程序

续表

配 置 段	说 明
Monitor	配置显示器的物理参数
Screen	把一台显示器和一块显卡联系起来
Module	配置动态加载的扩展模块，这些模块通常用于图形加速、特效显示等
ServerLayout	把输入设备和屏幕（Screen）联系起来

本节主要介绍和显示有关的配置段，这通常是用户最常用到的地方。修改 `xorg.conf` 需要 `root` 权限，用户可以在终端模拟器下运行命令 `sudo gedit /etc/X11/xorg.conf` 以 `root` 身份打开 X 服务器的配置文件。其中 `gedit` 可以替换为任何读者喜欢的文本编辑器。

 **提示：**怎样打开终端模拟器？Ubuntu 用户可以选择“应用程序”|“附件”|“终端”命令来打开；openSUSE 用户则可以依次选择“K 菜单”|“应用程序”|“系统”|“终端”|“终端程序”命令来打开。

4.4.2 配置显卡

Device 配置段用于描述一块特定的显卡，一个简单的 Device 段如下所示。

```
Section "Device"
    Identifier      "Configured Video Device"
    Driver          "fglrx"
EndSection
```

关键字 `Identifier` 用一个字符串来标识这块显卡，这里显卡的名字就叫做 `Configured Video Device`，这个名字还会在其他地方用到。关键字 `Driver` 指定了用于这块显卡的驱动程序。这个驱动程序只有在相应的 Section 段被用到的时候才会加载。

4.4.3 配置显示器

Monitor 配置段描述一台显示器，下面是一段典型的显示器配置。

```
Section "Monitor"
    Identifier      "Configured Monitor"
    Option          "DPMS"
    HorizSync       30-65
    VertRefresh     50-120
EndSection
```

关键字 `Option` 用于开启特定属性，这里打开了 `DPMS`（显示器电源管理），让 X 服务器在一段时间没有接受到用户输入的时候关闭显示器电源。`HorizSync` 和 `VerRefresh` 设置了显示器的刷新频率。通常来说，驱动程序可以自动探测到这两个参数，但是在 Monitor 段指定可以避免驱动程序使用显示器实际不支持的刷新频率。

4.4.4 联系显卡和显示器

在完成显卡和显示器的配置后，还需要把这两者联系起来，这种联系是通过 `Screen` 段

实现的。下面是最简单的一个 Screen 段。

```
Section "Screen"
    Identifier      "Default Screen"
    Monitor        "Configured Monitor"
    Device         "Configured Video Device"
    Defaultdepth   24
EndSection
```

同样，Identifier 给这个“屏幕”取了一个名字（这里是 Default Screen）。随后的关键字 Monitor 和 Device 指定了这个“屏幕”使用的显示器和显卡。随后的关键字 Defaultdepth 设置了 X 服务器默认使用的颜色深度，X 服务器需要依此判断使用怎样的分辨率。

4.4.5 如果显示出了问题

如果因为错误的显示器配置或者错误的驱动程序而导致 X 服务器无法正常工作，那么最简单的办法就是关闭所有额外设置，把以上 3 段设置成下面这样：

```
Section "Device"
    Identifier      "Configured Video Device"
EndSection

Section "Monitor"
    Identifier      "Configured Monitor"
EndSection

Section "Screen"
    Identifier      "Default Screen"
    Monitor        "Configured Monitor"
    Device         "Configured Video Device"
    Defaultdepth   24
EndSection
```

这样 X 服务器将忽略已有的显卡驱动，以最“原始”的方式使用显示器。卸载出问题的驱动程序，并逐步恢复配置，重新启动 X 服务器，直到显示器正常工作。读者现在可能还不知道如何完成所有这些工作，这些内容分散在本书的各个章节中。阅读完本书后再回过头来看这部分内容，会对进一步理解有所帮助。

4.5 小 结

- ☐ Ubuntu 的图形化应用软件都存放在“应用程序”下拉菜单中。
- ☐ 依次选择“位置”|“计算机”命令可以打开文件系统浏览器。
- ☐ gedit 是 Gnome 桌面环境附带的文本编辑器，支持语法加亮等高级功能。
- ☐ 个性化设置针对当前用户，因此一般不需要提供 root 口令。
- ☐ 通常情况下应该避免设置显示器分辨率。
- ☐ 本地主机和同一网络的主机应该避免通过代理服务器访问。
- ☐ 恰当地设置键盘快捷键可以提高工作效率。
- ☐ Linux 的图形环境是可以卸载的。

- ❑ X 窗口系统 (X Window System) 是 Linux 图形用户环境的基础。
- ❑ 显示管理器负责启动 X 系统的绝大多数组件。
- ❑ X 窗口系统基于服务器 / 客户机架构, 图形化应用程序可以 “异地” 显示自己。
- ❑ X 应用程序通过查看环境变量 DISPLAY 决定在哪里显示自己。
- ❑ 选择 Gnome 还是 KDE 是性能和外观之间的权衡, 或者仅仅是个人口味的差异。
- ❑ Linux 中所有软件的配置都是通过文本文件实现的。
- ❑ X 服务器的配置文件是/etc/X11/xorg.conf, 可以使用任何文本编辑器打开和编辑, 但修改该文件需要 root 权限。
- ❑ 配置文件中所有以 “#” 开头的行是注释行。
- ❑ 如果显示出现问题, 可以通过修改 xorg.conf 禁用已有的显卡驱动程序。