UBUNTU FOR NON-GEEKS

2ND EDITION

A Pain-Free, Project-Based, Get-Things-Done Guidebook

by Rickford Grant

Printed on recycled paper in the United States of America

BRIEF CONTENTS

CONTENTS IN DETAIL

3
A NEW PLACE TO CALL HOME
Getting to Know the Desktop   31

4
MORE THAN WEBBED FEET
The Internet, Linux Style   49

7
DRESSING UP THE BIRD
Customizing the Look and Feel of Your System　　101

8
SIMPLE KITTEN WAYS
Getting to Know the Linux Terminal and
Command Line　　125

9
AUTOMATIX AND NOT SO AUTOMATIX
More Ways to Install Programs   153

10
GUTENBIRD
Setting Up and Using Your Printer and Scanner   175

11
FONT FEATHERED FRENZY
Adding New Fonts to Your System   189

12
POLYGLOT PENGUINS
Linux Speaks Your Language
201

13
PENGUINS BACK AT WORK
Getting Down to Business in Linux
211

14
BRUSH-WIELDING PENGUINS
Linux Does Art
225

16
PLUGGIN' IN THE PENGUIN
Ubuntu and Your iPod

267

17
COUCH PENGUINS
Video and DVD Playback in Ubuntu

281

18
DEFENDING THE NEST
Security

291

A
UBUNTU DESKTOP CDS FOR AMD64 USERS

B
RESOURCES

INDEX

ACKNOWLEDGMENTS

There are a good number of people who deserve thanks for the help and support they provided either while or before this book was written, and there are still others whose help will come after the book is released. I would like to acknowledge all of them now.

Starting with my family, I'd like to thank Sumire, as always, for her support and encouragement throughout the project. The same thanks go to my mother, Dixie Angelina Burckel-Testa, who helped out with the initial proofreading of my earlier manuscripts and always lends a hand when she can. Thanks go out again to my auntie and uncle, Danica Lucia and David Zollars, for their continuous support and help (and for getting me into this book-writing business in the first place); and to round out the clan, thanks are due to my cousin and friend, Stephanie Garrabrant-Sierra, who tries to keep me on track whenever I am on the verge (or in the midst) of stumbling.

In the production of any book, editors are so very important, and I would like to thank my editor, David Brickner, who was, in addition to being full of good ideas for this book, a whole lot of fun to work with; Carthik Sharma, the technical editor, who kept everything I wrote by the book and pointed out a few easier ways of going about things (check out his blog, by the way, at http://ubuntu.wordpress.com); Elizabeth Campbell, the production editor, who kept all the stuff going into this second edition together (and helped me identify my alter ego—or at least another one); Megan Dunchak, who had the

INTRODUCTION

My computing life began long ago in the Commodore/Atari days. No doubt inspired by Alan Alda's television commercials at the time, I purchased my first machine, an Atari 600XL with a cassette drive for storage and 16KB of RAM—more than I thought I would ever need. Most of my time on that machine, I must admit, was spent playing cartridge-based games and transcribing pages and pages of machine code from the now-defunct magazine *Antic* to create even more games. Eventually, my interest in computers increased, especially after seeing my first (and actually *the* first) Macintosh at the UCLA bookstore. The very in-your-face nature of the Mac's operating system caused me to become an operating system maniac. To date, I have worked with a lot of different operating systems, including Mac OS up to and including OS X, every Windows version from 3.1 to XP, and even IBM's much forgotten OS/2.

Though tempted to join the Linux fray, I continued to steer away from it for a long time because I could not help but see it, as so many others do, as a system for never-seen-the-light-of-day-faced, late-night Dr. Pepper–drinking,

Domino's-pizza-eating compu-geeks. However, when I moved to Japan and was suddenly surrounded by machines loaded with Japanese versions of Windows, I encountered numerous problems, such as language constraints. Since everything, including Help files, was written in Japanese, I ended up using only a fraction of the full potential of most software. Then there were those annoying Windows-type problems, such as the constant freezes and restarts and the gradual system slowdowns, which were eventually only remedied by reinstalling the system. Getting the software I needed to do the things I wanted to do also took its toll on my wallet, and I began to rethink my initial resistance to Linux. With Linux's multilingual support, system stability, and extensive and free software packages, there were plenty of incentives for me to get over my preconceived notions about the typical Linux user.

After a few failed attempts at getting Linux to work on the oddball, Frankenstein-like collection of junk that was my computer, I finally succeeded with a CD-based Knoppix distribution, which worked well enough to hook me in a little further. I moved on to Mandrake (now known as Mandriva) next, since that was claimed to be the most newbie-friendly version, and then tried out SuSE as well, which I found to be rather quirky. Eventually, I tried out Red Hat Linux and stuck to that because it just didn't give me any grief; and I, like most others, do not want any more grief than necessary.

I started off with my three desktop machines at work and home set up as dual-boot systems running both Linux and Windows, but I gradually found myself using only Linux. Although I had expected to encounter numerous limitations in Linux that would force me to return to Windows often, I instead found that I had actually increased my productivity. Other than lack of native support for Windows streaming media, I was actually able to do more due to the extensive software base that was now installed on my machine. Without having to fork out money that I could ill afford to spend, I was able to manipulate my digital images, rip songs from CDs, create vector drawings, create PDF files, and do a variety of other things that I wasn't able to do under Windows. It was only a matter of time before my dual-boot setups became full Linux-only setups. I ceased to be a Windows user.

Since those early Linux days, I have gone on to try out a number of other distributions including JAMD, Xandros, Damn Small Linux, and most recently Ubuntu. I am happy to report that things have continued to get easier and better, and those early frustrations I suffered trying to get things to work with this machine or that piece of hardware are becoming more and more a thing of the past. Best of all, with the advent of live CDs, which allow you to try Linux out before you actually install it, you don't even have to take a leap of faith to get started.

Who Is This Book For?

If you are standing in the aisle of your local bookstore reading this right now, you may well be wondering who this book is for. If you also happen to see my previous books, *Linux for Non-Geeks* and *Linux Made Easy* on the

same shelf (or have at least heard about them), you might also be wondering what the differences among these books are. These questions are reasonable enough. To put it simply, there are two differences: the experience level of the target reader and the distribution covered in each book. *Linux for Non-Geeks*, based on Fedora Core 1, was my first book and was written with folks like my mother in mind—average computer users with some computer experience in the Windows or Mac worlds who had an interest in Linux but were afraid to give it a go. My second book, *Linux Made Easy*, was based on Xandros 3 and was written at a more basic level for those who just wanted a free and easy way out of the grasping tentacles of the Microsoft empire.

Ubuntu for Non-Geeks, as you might imagine, is based on Ubuntu Linux, and like the original *Linux for Non-Geeks*, it targets readers who are interested in Linux but feel the need for a jumping-off point of sorts. Although there are some similarities with its predecessor, *Ubuntu for Non-Geeks* is different in many ways. This is not only due to the inherent differences between the Debian-based Ubuntu and the RPM-based Fedora Core, but also due to the advances made by Linux as a whole. These advances have resulted in a system that is easier and more convenient to use than ever before.

If you are familiar with computers, but unfamiliar with Linux, or somewhat familiar with Linux but not with Ubuntu, you are essentially the reader for whom I have written this book. So to avoid any misunderstanding on your part (and at the risk of being redundant), I must re-emphasize that this is not a book for seasoned geeks or power users. It is instead an introductory guide that will provide new users with some hands-on experience in order to get them up, running, and comfortable with the Ubuntu distribution of Linux.

Version Compatibility

This book was prepared for use with Ubuntu 7.04 (Feisty Fawn) Desktop edition. Included with this book is a full working copy of that system on one CD, which functions not only as an install CD, but also as a live CD. This means that you can run Ubuntu directly off the CD without so much as touching your hard disk. You can thus give Ubuntu a try before making any hardware commitments. And since we're talking hardware, it is also worth pointing out that running a live CD session gives you the chance to see if Ubuntu works with the hardware you've got.

If you like what you see and all the hardware seems to work, you can go ahead and install the full shebang on your computer . . . and using the same disk, no less (instructions included in Chapter 2).

NOTE   *The world of computers is exceedingly dynamic, and as such, there may be changes in the software or the links to the files for projects in this book after the book is released. I'll post any such changes at www.edgy-penguins.org/UFNG.*

Concept and Approach

As a language teacher, I have always enjoyed programming books, mathematics books, and old-fashioned foreign language–learning texts because of their straightforward, skill-based orientation, one in which each chapter builds upon the skills acquired in the previous chapter. I have tried to organize this book in that manner so that you will never be called upon to do something that you have not already learned. I also like such books because they not only teach you how to do something, but they also provide you with the chance to put those morsels of knowledge into practice with exercises. I have therefore included several exercises, or projects, in this book so that you will have opportunities to apply your knowledge. This book will serve as a reference text and will also provide a dynamic learning experience so you can learn by doing, as they say.

The projects throughout this book have a secondary purpose as well: By working through them, you will properly configure and round out your Ubuntu system so that it can do anything you want it to do. By the time you finish with this book, your system will have all the bases covered. If that is still not enough to satisfy you, you will be happy to know that you will have access to even more—an unbelievably greater amount more—via the online Ubuntu repositories, which you will learn how to use in Chapter 5. If your interest is already piqued, take a look at these chapter descriptions:

Chapter 1: Becoming a Penguinista—Welcome to the World of Linux
What's Linux? What's Ubuntu? What's a distribution? Can I . . .? Will my . . .? Chapter 1 holds the answers to these and many other questions you might have as it introduces you to the world of Linux and what it takes to get it up and running on your machine.

Chapter 2: Wading and Diving—Running and (If You Like) Installing Ubuntu
The Ubuntu Desktop CD that comes with this book works as both a live and an install CD. Chapter 2 tells you how to run a live Ubuntu session off the CD and, assuming you catch the Linux bug after doing that, how to install the full Ubuntu system on your hard disk as your sole operating system or in a dual-boot setup with Windows.

Chapter 3: A New Place to Call Home—Getting to Know the Desktop
Regardless of whether you are an émigré from the Windows or Mac worlds, the Desktop is something you are already quite familiar with. Chapter 3 will point out the differences between Ubuntu's GNOME desktop and the one on your previous operating system, and it will teach you a number of cool tricks you can use to customize the look and feel of things. A couple of nifty GNOME Easter eggs are also introduced.

Chapter 4: More Than Webbed Feet—The Internet, Linux Style
"Have computer, will cyber-travel" could well be the mantra of the Internet age, and that being the case, Chapter 4 is an indispensable part of your Ubuntu experience. In this chapter you will learn how to

connect to the Internet and set up wireless connections, and you will meet the various software entities that allow you to interact with the Web.

Chapter 5: Rounding Out the Bird—Downloading, Installing, and Updating Programs the Easy Way
Ubuntu comes bundled with most of the software you need, but there is still much more available out there, free and waiting on the Web. Chapter 5 teaches you how to easily download and install applications using Advanced Package Tool (APT) and Synaptic. System and application updating is also covered.

Chapter 6: A Tidy Nest—File and Disk Handling in Ubuntu
From creating folders to copying files to browsing your system and network, all things file management are covered in Chapter 6. You will also learn how to work with USB storage devices, burn data CDs and DVDs, deal with CD-RW disks and multisession CDs, and create space-saving compressed archives of file folders.

Chapter 7: Dressing Up the Bird—Customizing the Look and Feel of Your System
Tired of looking at the same old desktop? Feeling nostalgic for the desktop in your previous operating system? Chapter 7 tells you how you can beat the déjà vu blues by changing the look and feel of just about every visual element of your system.

Chapter 8: Simple Kitten Ways—Getting to Know the Linux Terminal and Command Line
Many people still shy away from Linux because they perceive it as a system in which everything still needs to be done by typing commands. That perception is, as the saying goes, a load of squashed avocados. Still, there is a lot of cool stuff that can be done via the command Terminal, and Chapter 8 will tell you all about it as it tames your fears and piques your interest in commands. Really.

Chapter 9: Automatix and Not So Automatix—More Ways to Install Programs
Ever wanted to compile your own application? Or maybe you'd like to convert an RPM to a DEB package? Or maybe you'd just like to learn a really easy way to install all of the most essential missing bits of your system? In Chapter 9 you will learn how to do all these things, and you will even find out how you can run some Windows applications from within Linux.

Chapter 10: Gutenbird—Setting Up and Using Your Printer and Scanner
Just about everyone with a computer has or needs a printer, and Chapter 10 tells you exactly how to get yours working with your new Linux system. Scanner usage and support are also discussed.

Chapter 11: Font Feathered Frenzy—Adding New Fonts to Your System
Whether you want to use the same fonts that your Windows-using friends are plugging into their documents or you just want to add a bit of flair to your own, Chapter 11 will tell you how to do it in Ubuntu.

Chapter 12: Polyglot Penguins—Linux Speaks Your Language

Need to jot off a note in Urdu? Write a book in Korean? Send a letter in Chinese to your friend in Chengdu? All of the basics you need to know to read and write in just about any language in the world are provided in Chapter 12.

Chapter 13: Penguins Back at Work—Getting Down to Business in Linux

Work can be a drag, especially when there are so many other things you could be doing. Still, wearing the ol' fingers to the bone is a part of life for just about everyone outside of a Jane Austen novel, so you'll be glad to know that Linux is a very capable system in this regard. Chapter 13 introduces you to the various productivity applications bundled with or available for your system.

Chapter 14: Brush-Wielding Penguins—Linux Does Art

Those of you with an artistic bent will find Chapter 14 especially useful. Working with your digital camera, modifying images, and building web albums are just some of the topic areas covered.

Chapter 15: Tux Rocks—Music à la Linux

Chapter 15 is the music lover's treasure trove. You will learn how to rip CDs, encode MP3 or Ogg Vorbis audio files, and even find out how to create your own mix-and-match audio CDs from those files. A number of audio ripping and playback applications are also covered.

Chapter 16: Pluggin' In the Penguin—Ubuntu and Your iPod

Need I say more? Have an iPod? Want to use it in Ubuntu? Chapter 16 tells you how.

Chapter 17: Couch Penguins—Video and DVD Playback in Ubuntu

Sitting in your dorm room trying to figure out how to play your DVD copy of *The Baxter* on your Ubuntu-ized computer? Just finished filming a video of your sibling talking while asleep and want to do some creative editing of the evidence? Chapter 17 covers these and other video-related topics.

Chapter 18: Defending the Nest—Security

Although Linux is about as safe and secure a system as you are likely to come across, some folks feel a bit more secure . . . well, feeling a bit more secure. Chapter 18 tells you how to add a few lines of defense to your system.

Appendix A: Ubuntu Desktop CDs for AMD64 Users

The CD bundled with this book is designed to work with i386 processors. It will also work with AMD64 processors, although not in 64-bit mode. Want to run Ubuntu with an AMD64 processor in 64-bit mode? It's easy enough to do. Appendix A shows you how to download Ubuntu Desktop CD images and burn them to disk.

Appendix B: Resources

Are you crazy for Ubuntu and want to say so? Check out a forum. Do you have a hardware compatibility question? Some websites seem to have all the answers. Are you looking for free downloads or do you want to read up on the other Linux distributions? Appendix B is a great place to start.

How to Use This Book

It is possible, of course, to use this book as a mere reference text that you only consult when you have a problem to solve, but that would negate the basic concept behind its design. Instead, I recommend that you go through the entire book chapter by chapter, doing the projects along the way. This will give you a much broader understanding of how things are done (and of how you can get things done), and it will reduce the chance for anxiety, confusion, and worse yet, mistakes.

It is best to read this book and complete its projects when you are relaxed and have time to spare. Nothing makes things go wrong more than working in a rush. And keep in mind that Linux and the projects in this book are fun, not just work exercises. The whole point of the Linux world, in my opinion, is that it offers all kinds of fun. So go ahead and enjoy it.

About the Conventions Used in This Book

There are only a few minor points worth noting about the conventions I have used here. I have put in **bold** type the items within your system that you need to click or directly manipulate in any way, such as buttons, tabs, and menus. Where words or phrases are defined, they have been set in *italics*. Any text that I ask you to input will be indicated by `monospace` font. I have also opted to use the more graphically suggestive term *folder* instead of *directory*—no doubt the legacy of my many years as a Mac user.

About the Projects in This Book

The projects and other information in this book are primarily geared toward users who have installed Ubuntu using the CD that comes with this book. Most of the information also applies to Ubuntu live sessions run from the CD. You should note, however, that some projects and actions cannot be performed in live sessions, as they require write access to your hard disk, and this is not possible during live sessions.

1

BECOMING A PENGUINISTA

Welcome to the World of Linux

Now we begin our project to get you up and running in the world of Linux. If you have already made the commitment and have Ubuntu installed on your machine, you are essentially ready to go. Others of you might have made the commitment psychologically, but have yet to act on that commitment. And some of you are probably reading these words in the aisle of a bookstore, wondering about Linux and about whether you should spend your money on this book or on a latté every morning for the next couple of weeks. For those in this last group, I can only say, "Get this book." Save the wear and tear on your stomach and nerves.

In any case, the first thing we need to do is get you up to snuff on what this Ubuntu thing is all about, why you might want to install and use it, and what you will need in order to do so. I expect you will have lots of questions along the way, and if you are like most people, a few doubts. I hope that by the time you finish this book and have your Linux system up and running, your doubts will be gone and your questions, for the most part, will be answered. Anyway, until you are ready to make the commitment, you can

still follow along, because the CD that comes with this book contains, in addition to the Ubuntu installer, a live Ubuntu environment—meaning that you can have a taste of the Ubuntu Linux experience without having to even touch what you've got on your hard drive. You can kick back, put your worries in check, and go with the flow.

What Is Linux?

Your computer, despite being a collection of highly sophisticated parts, is really just . . . well, a collection of highly sophisticated parts. On its own, it can do nothing other than switch on and off and spin a disk or two. In order for it to do anything truly useful, it needs an operating system (OS) to guide it. The OS takes an essentially well-endowed but completely uneducated hunk of a machine and educates it, at least enough so that it will understand what you want it to do.

You already know of and have probably used at least one of the many operating systems that exist today. Windows, DOS, and the Mac OS are all such operating systems, and Linux is yet another. Linux is, however, different from these other operating systems, both in terms of its capabilities and its heritage. Linux was not created by a corporation or by some corporate wannabes out to make money. The Linux core, referred to as the *kernel*, was created by computer enthusiast Linus Torvalds, a Finn and a member of Finland's Swedish ethnic minority, who wanted to create a better Unix-like system that would work on home computers, particularly his.

Rather than keeping his creation to himself, Torvalds opened it up to the world, so to speak, and the Linux kernel, which communicates with the hardware, and makes it accessible to the other applications and support libraries created by compu-geeks around the globe who work to make Linux better and more powerful. It is this combination of applications built around the core of the Linux kernel that is the essence of all Linux distributions today.

Linux has acquired many fans and followers since its creation in 1991. Such devotees praise Linux for its many features, as well as for being robust, reliable, free, and open. Despite these positive characteristics, however, Linux is, on its own, just a text-based system. There is no pretty desktop, and there are no windows or charming little icons to make you feel safe and comfy once you are behind the keyboard. Powerful though it may be, Linux is still strictly a black-screen, command line–driven operating system. I guess you could think of it as DOS on steroids, though a Linux purist will surely cringe at the thought. Sorry.

Although you can use Linux by itself, accomplishing all your tasks by typing commands on a black screen (the most common way of doing things when Linux is used as a server), you don't have to. It is fair to say that with the advent of the Macintosh and its easy-to-use graphical user interface (GUI, pronounced *goo-ee*) in 1984, users of other operating systems began suffering something akin to GUI envy. They began clamoring for a GUI to call their own. The final result was Windows, which gave DOS a GUI and eased many command-wary users into the Microsoft world.

Similarly, many members of the Linux world felt the need and desire to go graphical. Various GUIs (called *window managers* and *desktop environments*)

and a subsystem with which to handle them (somewhat confusingly referred to as the *X Window System*) were developed by the community at large to bring about the change. The graphical desktop environment, GNOME, that is included in your Ubuntu distribution is one example of the fruit of that development.

About the Penguin

You may have been wondering about the penguin in the chapter title, so I might as well explain that now. The penguin was chosen by Linus Torvalds as the Linux mascot, and what has come to be thought of as *the* Linux penguin was designed by Larry Ewing and is named Tux (see Figure 1-1). This explains not only the ornithological references and graphics throughout the book, but also why there are so many penguin icons in Linux distributions and so many programs that include *penguin* or *Tux*, such as TuxRacer, XPenguins, and Pingus. This, combined with the fact that Linux is a revolutionary OS, helps to explain why Linux users are sometimes referred to as *Penguinistas*.
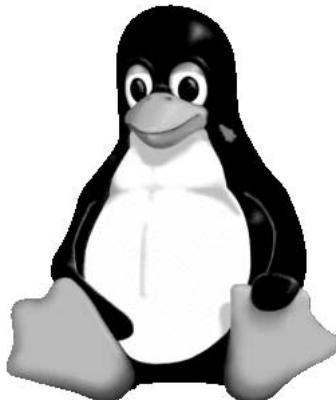
Figure 1-1: Tux, the Linux mascot

Why Should I Use Linux?

People use Linux for different reasons. For many it is a matter of power, stability, multilingual capabilities, or even personal philosophy. However, for others, crass as it may sound, it is a matter of money. Just think for a moment about what it usually costs to get started with an operating system. Go to wherever it is you go to buy software, and take a walk down the aisles. Make a list in your head of all the things you want to buy and how much each costs: an office suite; a game or two; maybe a graphics program with which to make yourself look better in your digital photos; and a collection of all those firewall, antispam, antivirus, and anti-adware programs that you really need to protect yourself in the Windows world. Now do the math.

After you pick yourself up off the floor, you will understand that we are talking big bucks here. On the other hand, for the price of this book you will have all of the things you wanted and more in the Linux world. Despite the

worries that many people have, making the move to Linux means not only savings for you, but also more computing versatility. You will not be hamstrung at some point along the way because you don't have this or that program when you need it most—you'll have it all from the get-go, or else be able to download it easily . . . and at no cost!

You might counter with the fact that there are a lot of freeware applications out there for other operating systems, but c'mon, let's face it—these are often rather limited in terms of their capabilities. The programs with a little more oomph are mostly shareware, and most shareware programs these days are limited in some way, or they only let you use them for a short time unless you are willing to pay for them. Sure, their costs are relatively low, but $25 here and $35 there eventually adds up to a considerable chunk of change. There is also the problem that some of these programs, unbeknownst to you, install backdoors, or keyloggers, or make your system a sudden garden of adware. Finally, at least in my experience, the majority of such programs are hardly worth the money asked. The only shareware programs I ever found worth buying were Lemke Software's GraphicConverter and Plasq's Comic Life, both for the Mac.

Is It All Just About Money?

While money is important to the average user, it is certainly not the only reason for taking the Linux plunge; there are a variety of other reasons as well. As I mentioned before, Linux is noted for its stability. Try running your present system for a month without restarting and see what happens. Linux has been known to run without a reboot for over a year without a hitch or decrease in performance. With its multilingual capabilities, Linux is also a perfect choice for language students or users in a multilingual environment.

In addition, Linux is infinitely customizable: You can get your system to look and act the way you want it to without being wizarded to death. And then there are the applications that come with most Linux distributions. In addition to their wide variety, most are well up to industry snuff, with some, such as Evolution and the GIMP, being sources of envy for those outside the Linux world.

Finally, with the advent of Microsoft's new Windows Vista system and its more demanding hardware requirements (especially if you want to take advantage of its most touted new features), you may find your present machine on the fast track to obsolescence. Turning it into a Linux machine will ensure it several more years of working life. Shame to put good hardware out to pasture so early, after all.

But Is Linux Really Ready for the Desktop?

Despite the advances Linux has made in recent years, this question still pops up quite often, and that's fair enough. But consider this: When you install a program on your present Windows system and get an error message saying that the program can't run because some DLL file is

missing, or when you connect a piece of hardware and can't get it to run, no one asks if that operating system is ready for the desktop.

In my own experience, I have found no reason to doubt that Linux is ready. Sure, Linux has its occasional quirks, but so does every other operating system. Linux is ready and able. If my mother, hardly a computer wiz, can do the work she needs to do and can keep herself amused till the middle of the night using her Linux system (without blowing the whole thing up), then I think it's pretty safe to say that you'll do all right too.

What Is a Distribution?

An operating system consists of a lot of files that perform a lot of different functions. And because there is no Linux corporation to package and distribute the files that make up Linux, the task of getting Linux onto your computer in working order, along with the applications that you are likely to want, has fallen to a varied group of entities—companies, universities, user groups, and even private individuals. These entities create Linux system and application collections called *distributions*, or *distros*. You could bypass such distros and try to collect everything you'd need to set up a system all on your own, but you would undoubtedly lose your mind in the process. Most people, even the geekiest, opt for the distros.

The majority of these distros, whatever their ultimate target audience, basically consist of the same main elements: the core operating system (better known as the *Linux kernel*), some sort of installer program to get all the system parts and applications properly installed on your machine, the X Window System to provide graphical interface support, one or more graphical desktop environments, and a series of applications, such as word processors, audio players, and games, as well as all the files needed to make these things work.

There are, of course, a large number of distros. Some are geared toward specific audiences, such as businesses, educators, gamers, students, program-mers, system administrators, and specific language users. What makes each distro different is the specific software that is bundled with the Linux kernel, as well as other convenience features like the package, or application, instal-lation mechanism, and the installer for the system itself. Some distros are especially appropriate for home users due to their ease of installation. Ubuntu, a relative newcomer to the Linux world, is one of these, joining other distros that have long been popular in the ease-of-use arena, such as Mandrake, SUSE, and Fedora Core. There are also many other new distros, like Xandros and Linspire, that are specifically geared toward making the transition for Windows users easier. While many of these entities charge for their distros, most also provide them free for download.

What Is Ubuntu?

Ubuntu is a completely free, easy-to-use, and extremely popular Linux distri-bution that is geared toward the desktop user. It is one of the hottest Linux distros in the marketplace today. It is also one of the few Linux distros with what could be described as a social agenda behind it.

Ubuntu was the brainchild of South African millionaire entrepreneur Mark Shuttleworth, who is probably better known for being one of the first space tourists—the first African in space, to be exact. Shuttleworth invested over $10 million starting up the Ubuntu Foundation based on his belief in free software and in order to fix what he describes as "bug #1"—Microsoft's dominance of the desktop PC marketplace. As Shuttleworth states in his blog (available at https://wiki.ubuntu.com/MarkShuttleworth):

> I believe that free software brings us into a new era of technology, and holds the promise of universal access to the tools of the digital era. I drive Ubuntu because I would like to see that promise delivered as reality.

As you can see, it's a vision thing.

Befitting the nationality and goals of the man who brought it into being, the word *ubuntu* comes from the Zulu and Xhosa languages. *Ubuntu*, according to Wikipedia, is a concept meaning something along the lines of *humanity toward others* or *I am because we are*. If you're interested, the 2005 film *In My Country*, starring Juliette Binoche and Samuel L. Jackson, although not one of the greatest films ever produced, is on many levels a 100-minute examination of the concept of ubuntu.

Why Ubuntu Then?

With so many distros out there, you may wonder why you should opt for Ubuntu. Well, as they say, numbers don't lie, and Ubuntu's popularity is not without good cause. These traits are especially crowd pleasing:

Easy to install
It's fair to say that most Linux distributions these days are pretty easy to install (and definitely easier and faster to install than Windows). Ubuntu is right in line with these improvements, and the fact that you can install it with only a few mouse clicks while running the live CD means it is pretty much ready to go whenever you are.

Easy to use
Ubuntu is easy to use in that it is very Windows-like in operation, and yet it's more Linux-like than other Windows user–oriented distributions.

DEB based
Ubuntu is based on the *Debian* distribution, which means that it utilizes Debian's very convenient DEB package system for application handling and installation. The two preconfigured, graphical package installers that come with Ubuntu make installing applications even easier. There are so many packages available for Debian systems like Ubuntu that you are likely to find more software out there than you'll ever know what to do with.

Up to date
Some distros are updated at a snail's pace, while others strive to be so cutting edge that they are often plagued with bugs. Ubuntu, with its reasonable six-month release cycle, tries to stay as up-to-date as possible, while at the same time making sure that things are not released before they are ready for prime time. In this way, you are ensured of having an up-to-date yet less buggy distro at your disposal.

Dependable and robust
> I know these terms come across as mere hype, but after you smack Ubuntu around a bit, you come to understand what they mean. Knock things down and around, and they bounce right back—this is very important for beginners who often have a knack for screwing things up. Nothing turns a new user off more than a twitchy system that has to be velvet gloved all the time.

Desktop user–oriented
> A lot of Linux distributions, although quite capable in the desktop arena, cater more to geeks and developers, taking up valuable disk space with a lot of junk you'll probably never use. Ubuntu's purpose is to grab desktop market share from the Redmond folks, so the needs of the common end user are always in mind. The result is that Ubuntu's GNOME desktop environment is a very comfy place for the average desktop user to be.

Hardware Compatibility

Well, enough of this background babble; it's time to get things rolling. If you haven't installed Linux on your machine yet and are wondering whether you can, it is relatively safe to say that Ubuntu will run on most machines out there today. Of course, this statement comes with a major caveat: You just never know until you get up and running. There are so many minor parts to your machine that it is difficult to say whether each part will cooperate with your installation. Ignoring minor parts for the time being, there are video cards, sound chips, LAN cards, monitors, and so on, and they all need to be considered.

Diving In

If you are going to buy a new machine on which to run Ubuntu, then it is reasonable enough to do a bit of worrying and check things out first, but if you are going to install it on the machine you have, I recommend just diving in. After all, you don't really have to install anything the first time out. You have a live CD right here in this book, after all, so you can just pop that CD in your drive, boot up your machine, and, *biff, bam, zowie*, you'll be up and running (or not) in a minute or two. If everything seems to be going as it should . . . well, your worries are over, and you can go ahead and install the system when you're ready and willing. That is one of the Ubuntu advantages—not only do all the essentials fit on a single CD (compared to four or more for other distros), but that CD is both a live operating environment and the installer! You can't get much more convenient than that.

When Research Is Required

If things don't work out for you with the live CD, you can search the Web to see if you can identify what part of your hardware puzzle is causing your problems. (Or if you are looking to buy a machine on which to install Ubuntu, you can search for hardware that is supported by Linux.) Of course, before you can do

this, you need to know what models of hardware you have. You should know at least what motherboard, central processing unit (CPU), monitor, and video card you have if you want to be able to find out anything of value. Identifying your CPU and monitor should be easy enough, but the motherboard and video card may require a bit more searching.

If you have no documentation that clearly states the make and model of these devices, you can find out most things you need to know from within Windows by going to the Windows Control Panel, double-clicking **System**, and then clicking the **Hardware** tab in that window. Once in the Hardware tab, click the **Device Manager** button, and see what you can find about your system components there. Sometimes the information there is rather limited, so you might instead want to try out a shareware application such as HWiNFO (www.hwinfo.com) or Sandra (www.sisoftware.net) to get more useful details, such as the specifications of your motherboard or the supported video modes for your present setup (see Sandra in Figure 1-2).

Figure 1-2: Finding out your hardware details from within Windows with Sandra

Both HWiNFO and Sandra should give you the information you need about your motherboard, but if they don't (or if you don't feel like bothering with them), you can always just open up the case of your computer and look at your board. You needn't worry about damaging anything because you don't need to touch anything—so don't. You may need a flashlight to find it, but the model name and number should be stamped on there somewhere,

either in the middle of the board or around the edges. Mine, for example, says quite clearly in the middle of the board, *AOpen MX46-533V.* You should be looking for similar information.

Once you have all your information, you can do a variety of things to check out your hardware's compatibility with Ubuntu. You can simply do a Yahoo! or Google search by entering your motherboard's make and model plus the word *Linux.* This works for other hardware devices too.

You can also post a question to the Ubuntu User Forums (at www .ubuntuforums.org) or one of the other various Linux forums or mailing lists on the Web. A listing of some of these is provided in Appendix C at the end of this book. Just write that you are a newbie and want to know if anyone has had any experience using Ubuntu with the board (or other hardware) in question. You will probably get quite a few responses. Linux users are usually rather evangelical in terms of trying to draw in new Penguinistas.

Hardware Requirements

All worries about compatibility aside, there are some minimum hardware requirements that you will need to meet:

- Any computer with an i386-based processor, or an Intel or AMD 64-bit processor
- About 2 gigabytes (GB) of hard disk space, though having at least 10GB would be a bit more comfy
- Sufficient memory (RAM)

NOTE *The CD that comes with this book is designed to work on machines with i386-based processors (basically, all the Pentium chips, including Celeron, Xeon, and the new Core Duo, as well as processors from AMD). Though this CD will install Ubuntu on a computer with an AMD or Intel 64-bit processor, it will only run in 32-bit mode. To make full use of your 64-bit processor, you need to download the 64-bit version of Ubuntu.*

As for RAM, the official specs tell you that you need a minimum of 128 megabytes (MB) to run Ubuntu. While you can no doubt get by with this, you'd get by much better with more. My basic rule of thumb, no matter what OS I am dealing with, is that you need the recommended (not the minimum) memory plus at least 128MB. Regardless of what the official specs say, put in more. You won't regret it.

Saying that the more memory you have, the better, may sound a bit simple, and perhaps even cavalier, but trust me on this one. When you have too little memory, no matter what system you are running, weird things happen: Applications seem to take years to open, or don't open at all; menus take forever to render their little icons; freezes and general system meltdowns just happen much more often.

To be realistic and exceedingly honest, I would say that 256MB is the absolute minimum you want to have. I personally would recommend that you have at least 384MB of RAM in order for things to move smoothly and comfortably. It is such a waste to have a pretty speedy CPU and not be able

to appreciate it because its hands are tied by a lack of memory. It is sort of like trying to do jumping jacks in a broom closet. Sure, you could do it, but you would be all contorted, and you'd be smashing your hands into the walls every 1.4 seconds.

Fortunately, it is pretty hard to find a machine with only 128MB of RAM these days, but if you do happen to have such a beastie, you can at least take solace in the fact that memory is relatively cheap, so go for it.

Good News for AMD64 Users

It is again important to mention that the CD that comes with this book is designed to work on machines with i386-based processors, which pretty much covers the vast majority of PCs out there. If your machine has an AMD64, you will be glad to know that it will also work, albeit not in 64-bit mode. Sorry, but no go.

Fortunately there is good news for those of you who were a bit disappointed by the content of that previous paragraph. Ubuntu is available in a native AMD64 version. Check Appendix A for information on how to get it. The information provided there will also be of use to i386 users who happen to lose or damage the disk that comes with this book.

Mixed News for Mac Users

Those of you with PowerPC Macs may be aware that previous Ubuntu releases have been available in PowerPC versions. Unfortunately, as of the Feisty Fawn release (the one I am writing about here), this is no longer the case. This means that if you are a PowerPC Mac user, you will have to use one of the older Ubuntu releases or wait for an unofficial, community-supported release of Feisty Fawn for PowerPCs (http://cdimage.ubuntu.com/ports).

Of course, if you are using one of the new Intel-based Macs, you're in luck because the disk that comes with this book will work on your Mac. After all, the *i* in *i386* stands for *Intel*.

Speaking Ubuntu

It's worth noting that there are a lot of weird phrases you are bound to come across when dealing with Ubuntu, especially when searching for information on the Net. In particular, I am referring to four seemingly incongruous phrases: *Warty Warthog*, *Hoary Hedgehog*, *Breezy Badger*, *Dapper Drake*, and *Edgy Eft*. These are the unlikely code names of each of the releases of Ubuntu since its first appearance in 2004. The important one for you to remember is that of the current release (the one on the CD that comes with this book): *Feisty Fawn*.

You are also likely to come across a few other variations of the Ubuntu theme. These are Kubuntu, a KDE-based version of Ubuntu; Edubuntu, a special version of Ubuntu designed for use in the classroom; and Xubuntu, which is a lightweight version of Ubuntu based on the XFCE desktop.

Where Do I Go from Here?

Now that you know more about the world of Linux and Ubuntu, and you've got your disk in hand, it's time to get down to it. If you have already installed Ubuntu on your machine, just flip ahead to Chapter 3. If your machine is still Linuxless, though, it's time to take it out for a spin and see if you like it. So for now, strap yourself down in front of that computer, clip on your spurs, and go straight to the next chapter. It's time to become a Penguinista!

2

WADING AND DIVING
Running and (If You Like) Installing Ubuntu

As I have already mentioned, one of the great things about Ubuntu is that it comes on a live CD, which means that you can try it out before you install it . . . or never install it at all, if that's what you prefer. Better yet is the fact that, unlike earlier editions of Ubuntu, you don't need an additional installation CD if you do choose to install it—the Ubuntu Desktop CD functions as both a live and an install CD. And while in the good-better-best swing of things, the best point of all is that installation from the live CD is actually much, much easier than any other installation process you've ever dealt with.

In this chapter, I will be covering the basics of starting up and running Ubuntu from the live CD, and then, assuming you've caught the Linux bug, the painless steps to installing the system on your hard disk. Let's put this book to use and get Ubuntu up and running. . . .

Going in for a Dip

To get a taste of what Ubuntu is all about (and to check out your hardware to see if it's all comfy-cozy with Ubuntu), there is probably no better way to go than to run Ubuntu directly from the live CD. To do this, just place the Ubuntu Desktop CD in your disk drive and restart your machine. When the machine starts up again, it should boot up from the CD, and, after a bit of white text scrolls up the screen, you should see the first Ubuntu startup screen (Figure 2-1). If the screen does not appear, and your machine instead boots up into your usual operating system, then it is very likely that your machine's BIOS needs to be changed so as to allow you to boot from CD.
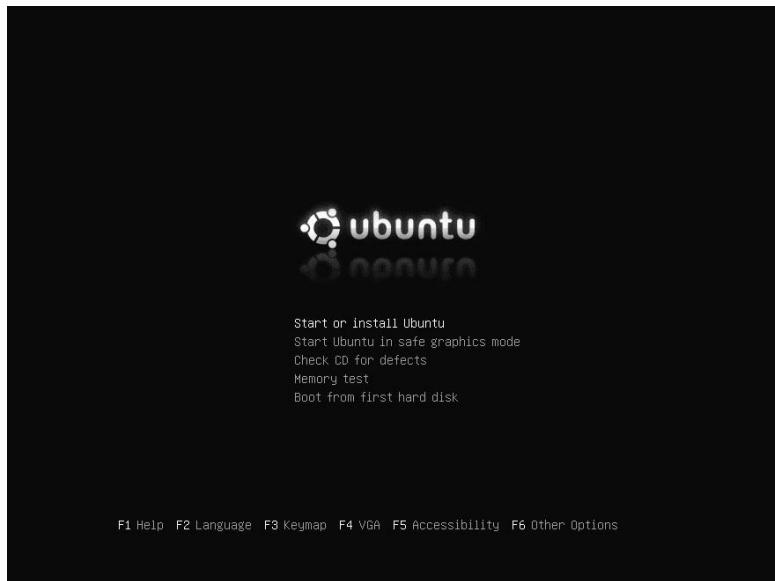
Figure 2-1: An Ubuntu live session startup screen

You can access your machine's BIOS by restarting and then pressing whatever key the onscreen startup instructions assign to accessing the BIOS setup. This is usually DELETE or F1, but not all machines are the same. If the onscreen information passes by so fast that you miss it, you can check your user's manual to see what the correct key is. Once you get into the BIOS setup, change the boot sequence so that your CD drive is first.

Once your machine does boot from the live CD, and you do see the Ubuntu live session startup screen shown in Figure 2-1, either press ENTER or just do nothing for about 30 seconds in order to continue. This will start up your Ubuntu in English with a US keyboard layout. Remember, your hard disk will go untouched, so rest easy—you're not going to touch, let alone hurt, anything.

If you would like your system to appear in a different language, press F2, select your language of choice using the cursor keys, and then press ENTER. You can change the default keyboard layout by pressing F3 and then following the same steps. Once you're done, press ENTER to start up the system.

From then on out, it is strictly autopilot time for you—all you have to do is wait. There will be some scrolling white text now and then, and an occasional period when things will go black for a few seconds, but eventually things will go totally graphical when the GNOME desktop environment begins its initialization process, and when that is done . . . well, you will be face to face with the Ubuntu desktop, meaning you're ready to roll—but I'll hold off on talking about that until Chapter 3.

Taking the Plunge—Installing Ubuntu

If you have already installed Ubuntu on your machine, are satisfied running it from the live CD, or still haven't made up your mind what to do, you can skip right over the rest of this chapter and go on to the next one to get started working with the Ubuntu desktop. If, however, you haven't installed Ubuntu yet and are ready and raring to do so, then you had better stay right where you are and read on.

Single- or Dual-Boot Setup?

If you don't already have Windows installed on your machine, you can skip right over this section. If, on the other hand, you do, then you are going to have to decide whether or not you want to keep it.

It is possible to have both Windows and Linux installed on the same machine and for them to happily coexist. This is known as a *dual-boot setup*. It has also become incredibly easy to set up such a system. I started out with a dual-boot setup; however, I eventually found that I used the Linux side of things exclusively. Having so much disk space being taken up by a Windows system I didn't use seemed a waste of prime real estate, so eventually I just dumped the whole thing and went for a straight Linux-only setup. My feeling is that unless there is some application that you really need that is not available on the Linux side (probably some game), then go for the Linux-only setup and just forget about Windows. Linux has most of what you will need anyway, and because the applications in the OpenOffice.org office suite can read and write Microsoft Office files, you'll still be able to collaborate with Windows users, if that is of concern to you.

You may be thinking that if you do as I suggest and dump your Windows system when you install Linux, you might have to reinstall Windows if you don't like Linux or if you can't get it installed properly. That would be a considerable waste of time and energy, to be sure. However, believe it or not, there are advantages to my suggestion even if your no-go scenario turns out to be the case.

You may have noticed that your Windows system, as you've used it over time, has gotten sort of gunked up—it is no longer the quick little kitten it used to be. Menus don't pop open as quickly as they used to, things take

longer to start up than they did before, and you find yourself asking, "What the Sam Habberdack is that?!" all the time as mysterious things happen with increasing frequency.

This is just the nature of the beast, and a very good way of getting things back to normal is to reinstall the whole thing. So even if you do decide to come back to Windows later, you'll be doing yourself a favor, because it should run better than before. It's a little more work up front, but in the long run, you'll be a happier camper.

If, on the other hand, you opt for a dual-boot setup, from which you can run both Windows and Linux, you will have the best of both worlds. Starting up in either system is easy. When you start up your machine, you will be greeted by the GRand Unified Bootloader (GRUB) screen, from which you can choose to continue booting up Linux or choose Windows in its stead. After that, bootup proceeds as normal for the system you selected. This setup works fine, so you needn't worry.

So as you see, either way you decide to go, you can't really go wrong. Just be sure to back up your important files before starting the installation; proceed with common sense, patience, and a positive attitude; and you'll be fine. In short: Don't worry.

Getting Ready for Action

There is less you need to do to prepare for an Ubuntu installation than for many other Linux distributions. Once you've decided whether or not you want to go the dual-boot route, all you really need to have on hand is your single Ubuntu Desktop disk and, for guidance and security, this book. The only mental energy you'll probably need to expend is to come up with a username and user password, just as you do for most other operating systems.

Usernames and User Passwords

Your username is something that you will be using quite a bit. You will input it every time you boot up your system, so be sure it is something you can live with, especially in terms of typing. It can be just your first name, or your initials, or whatever you want it to be. It must, however, begin with a lowercase letter, followed by numbers and/or other lowercase letters. Mine, for example, is simply *rg*, but you could use something like *hope4u2pal*, though that would get rather tiring to type at login day after day. Think of your fingers when deciding upon your username.

You also need to come up with a user password, which you will also need to type every time you log in. You will need to use it when you install new software or change certain system settings, as well. It should be a minimum of eight characters in length and consist of numbers and letters (upper- and lowercase) for improved security. You can, of course, get by with fewer characters and only letters if you prefer. The installer will advise you if the password you enter is unacceptable, so don't worry too much. Be sure to write it down and keep the paper you've written it on in a safe place so you don't end up locking yourself out of your system.

Dual-Booters Take Note

If you are going to be creating a dual-boot setup, it is a good idea to first defragment your present Windows disk before moving on to installing Ubuntu. This will make the repartitioning phase of the installation process must faster and safer.

You can do this while still running Windows by double-clicking **My Computer**, right-clicking the icon for your hard disk, and selecting **Properties**. You can then defragment your hard disk (in Windows XP) by clicking the **Tools** tab in the Properties window and clicking the **Defragment Now** button. In Windows NT, you can do this by going to the **Start** menu and selecting **Control Panel ▸ Administrative Tools ▸ Computer Management ▸ Disk Defragmenter**, while in Windows 98 and some other versions, you can do the same by selecting **Programs ▸ System Tools ▸ Disk Defragmenter**.

Doing the Deed

Well, now that we've covered all that prelim stuff, let's get down to the actual Linux installation. Set this book on your lap so you can follow along, and then get ready for action. It's time to do the deed!

Fortunately for you, the installation process is extremely easy, as there are very few steps in which you actually have to do anything. Most of what you will be doing is clicking buttons on your screen. Nothing hard about that, eh?

Of course, when you look at the directions and descriptions listed here, it may look like a long and cumbersome process. It is not. It will be over more quickly than you can imagine. As a beginner frequently referring to this text, you might take a bit longer, of course, but don't worry. All in all, the whole process is faster and easier than that for Windows XP or Mac OS X. And keep in mind that with XP and OS X, you are installing the operating system with just a few bundled applications. In an Ubuntu installation, on the other hand, you are installing not only the operating system itself, but also most of the applications you will ever want or need to use. You will thus be getting a lot done in one fell swoop.

One more thing before we start. Some people approach installing a system with a good deal of trepidation. The process makes them nervous, as if the house is going to go up in smoke if they click the wrong thing some-where along the line. Needless to say, there is no need for such concern. As long as you have backed up your data, you will be okay. If you screw up the installation the first time out, so what? Just start over again. No harm done, as you have nothing to harm. Just make sure that you give yourself more time

than you need for the process. Don't start installing one hour before you have to be at work or before you have to meet your friend downtown. Rushing makes people do weird things. Make things easy on yourself by giving yourself plenty of time and, as I mentioned before, by backing up any data you would mourn the loss of.

If you're ready, here are the steps:

1. **Start 'er up**   If you haven't already done so, boot up your machine from the Ubuntu Desktop CD.

2. **Start the installation**   Double-click the **Install** icon on the Ubuntu live desktop to start the installation. This will bring up the first page of the installation wizard (Figure 2-2).

Figure 2-2: The opening installation wizard screen

3. **Welcome (choose language)**   Choose the language you want to use during and after the installation process, and then click the **Forward** button. No matter what language you choose, you will always have the option of booting up in English once the system is installed. You can also add support for other languages later.

4. **Where are you? (choose location)**   The Ubuntu installer will select the default location for the installation language you have chosen. If your location is different, select the one appropriate for you by clicking directly on the wizard map (Figure 2-3). Once your selection is made, click **Forward**.

Figure 2-3: Choosing your geographic location in the Ubuntu installation wizard

5. **Keyboard layout**   The default keyboard layout for the installation language you have chosen will appear in the next wizard screen. If your keyboard layout is different, make the appropriate choice from the list. If you're not sure you've made the right choice, you can double-check by typing a few words in the text box at the bottom of the window. Once you've done this, click **Forward**. If need be, you can add other keyboard layouts later, after the system is installed.

6. **Prepare disk space**   What you do at this point depends on what you have on your machine. Assuming you have an operating system on the disk already, such as Windows, accept the uppermost option (selected by default). This will reduce the size of your Windows (or other OS) installation to about half the size of the disk, and then use the resultant free space on which to install Ubuntu. If you like, you can adjust the size of the two partitions by using the slider in the middle of the window (Figure 2-4).

    If you have no operating system on your hard disk, or if you intend to eliminate what you currently have and replace it with Ubuntu, you should instead choose the second option, **Guided – use entire disk**, and then, if given a choice, select the main drive onto which you want to install the system. Once your selection is made, click **Forward**.
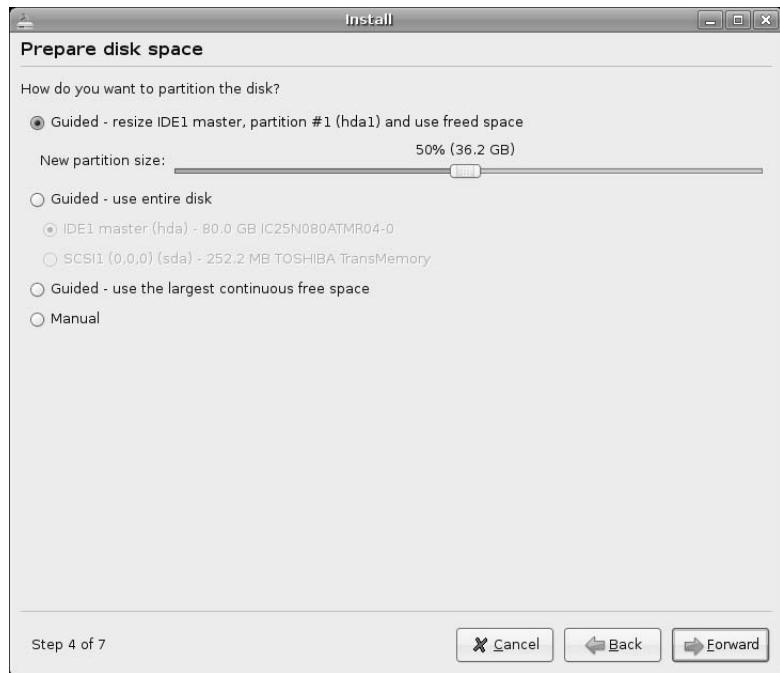
Figure 2-4: Partitioning your hard disk (or not)

7. **Migrate Documents and Settings**   If your machine already has another operating system on it, the installer will scan your hard disk to try to find settings, folders, and documents that it can transfer over to your new Ubuntu system. Just check the boxes next to the items you want to transfer (assuming Migration System finds something—it's still kind of buggy as of this writing), and then click **Forward**.

8. **Who are you?**   On the next page of the installation wizard, you are asked to provide your real name, your login name, and a password. The wizard will automatically generate a name for your computer based on your username (*rg-laptop* in my case), but you are free to change this to something else if you like (I changed mine to *UbuntuAcer-RG*). Once all of the fields are filled in, as mine are in Figure 2-5, click **Forward**.

9. **Ready to install**   The final page of the wizard (Figure 2-6) lists the details of your soon-to-be installed system and hard disk partition setup, along with a point-of-no-return warning. You've come this far, so you might as well go for it, even though there's no turning back. Click **Install**.

Figure 2-5: Providing your username, password, and computer name in the installation wizard
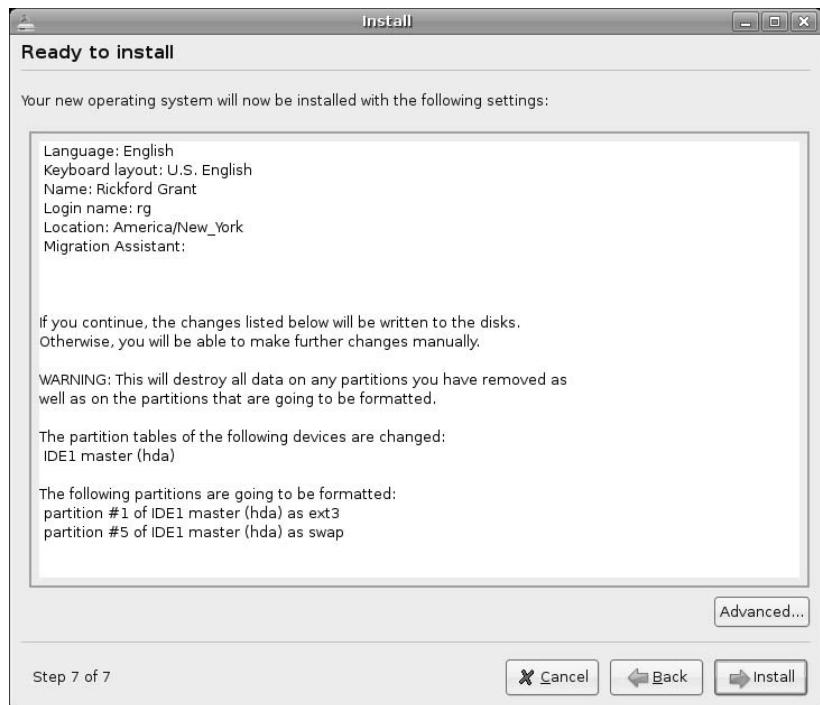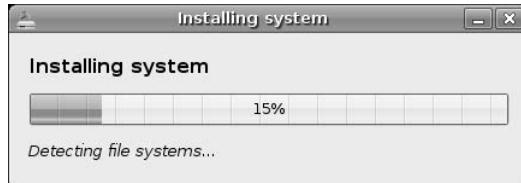
Figure 2-6: Ready to install—the final page of the installation wizard

The partitioner will then do whatever writing to disk it must in order to finish the partitioning process, after which the installation itself will seamlessly begin, without any additional input necessary from you or anyone else. The progress of the installation will be indicated in a window like that in Figure 2-7 so that you don't have to fret (and so you'll know how much more time you have left to veg in front of the TV).

Figure 2-7: Keeping track of the progress of the installation

10. **Installation complete**   As you can see in Figure 2-8, you've now come to the end of the first phase of the installation. You are given the option of either continuing to use the live CD or restarting the machine and running Ubuntu directly from your hard disk. Well, you didn't go through all of this just to keep using the live CD, so let's go for the second option by clicking the **Restart now** button, removing the live CD from your drive when it is automatically ejected, and pressing ENTER when prompted on screen to do so. Your machine will then restart.

Figure 2-8: A final warning before taking the plunge

After that . . . well, that's basically it. You now have Ubuntu installed on your machine. Congratulations. After a few moments, you will see the login screen, so to find out what to do then, go on to the next chapter. See you there. Aloha.

3

A NEW PLACE TO CALL HOME

Getting to Know the Desktop

Now Ubuntu is up and running, and you are ready and raring to go. If you are running Ubuntu from your hard disk, you will first see the login screen that will appear each and every time you boot up (Figure 3-1). There's no need to keep the login screen waiting, so type your username, and press ENTER. After that, you will be prompted for your user password in the same screen, so type it, and press ENTER again. Within moments, you will be face to face with your desktop in Ubuntu.

Figure 3-1: The Ubuntu login screen

Welcome to the GNOME Desktop

Ubuntu's implementation of the GNOME desktop is shown in Figure 3-2, and as you can see, it isn't all that different from what you might be used to in a Windows or Mac OS 9 environment, other than the fact that it has task-bars, or *panels*, at both the top and the bottom of the screen. There are also no desktop icons, except when running a live session from the Desktop CD, in which case you'll see a launcher to run the installation wizard (labeled Install) and a folder (labeled Examples), which contains a number of sample files. Among these is a video clip of Nelson Mandela discussing the meaning of ubuntu (the concept, not the distro). All in all, it is a very uncluttered place to be, and despite its superficial similarities to other OS desktop environments, things in the GNOME are different enough to be interesting.

The main elements of the GNOME desktop are the panels at the top and bottom of the screen and the icons that appear upon those panels. The desktop itself, although empty at startup, does see its share of action, but I'll come to that later. For now, I'll focus on the two panels.

The Top Panel

Of the two GNOME Panels on your desktop, the top panel is basically where all the action is. As you can see, there are three menus and two icons at the left end of the panel, and a few icons and a clock/calendar at the other end (Figure 3-3). So that you understand what each of the panel items does, I will now briefly describe each of them, moving from left to right, as seems to be the fashion these days.

Figure 3-2: The GNOME desktop in Ubuntu

The Menus

At the far left of the top panel, you will find a set of three menus. These provide access to most of what your system has to offer in terms of applications, locations, and utilities. These include:

Applications menu   The access point to the majority of your applications, a software manager, and some system tools.

Places menu   Your system navigator, from which you can hide all open windows so as to expose the desktop, access your home folder, browse your computer's filesystem and connected networks, and search for files on your hard disk.

System menu   The access point for your system preferences, software installer, and administration tools. This is also the place to go when you want to shut down or log out of your system.

Figure 3-3: The left and right sides of the top GNOME Panel

The Icons (Left)

Immediately to the right of the three menus are a set of two launchers. When these icons are clicked, they launch the following applications:

Firefox   Your web browser

Evolution Mail   The very popular Linux email program, scheduler, and task manager

Help   Access to the Ubuntu help files installed on your hard disk

The Icons (Right)

At the right side of the top panel are a series of icons that perform a variety of functions. Some of these are indicators, while some are applets that allow you to perform certain functions. These consist of:

Update Notification Tool   Tells you when there are system or application updates and allows you to download and install the updates. Only appears when updates are available.

Network Manager Applet   Lets you see your network status and configure your network devices. When used with a wireless network connection, indicates the wireless signal's strength and allows you to switch between wireless networks easily.

Battery   Shows your current battery level when charging or discharging the battery. Only appears on laptops.

Volume Control   A volume controller. Duh.

Calendar/Clock   Date and time, date and time, date and time. Just as you suspected.

Quit   Brings you to the logout screen, from which you can logout, shut down, restart, or switch users.

The Bottom Panel

The bottom panel, as you can see in Figure 3-4, is a much simpler affair, containing only the four items I will now briefly describe.
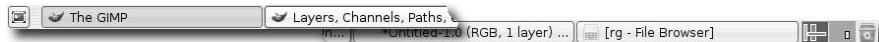
Figure 3-4: The bottom GNOME Panel

Show Desktop   A button that minimizes all open windows and allows you to see your desktop when it is obscured from view.

Window List   A list of windows or applications you have open, which is very similar to what happens in the Windows taskbar.

Workspace Switcher An application that allows you to switch between virtual desktops. (I'll talk about this more in "Virtual Desktops" on page 44.)

Trash There is nothing mysterious about Trash . . . other than its rather Mac OS X–ish location on the panel.

Project 3A: Customizing the GNOME Panel

The GNOME Panel is not a static thing. You can add *launchers* (respectively known as *program shortcuts* or *aliases* to Windows and Mac users), utilities, and even amusements to make it do almost anything you want it to—within limits, of course. In the various stages of this project, you will customize your panel to get some hands-on experience working with it and to make things more convenient for you as you make your way through the rest of this book. You are, of course, free to change any of the customizations I ask you to make (though you won't have a say in the matter if you're working in live session from the desktop CD, as you won't be able to save your settings).

Each of the following subprojects is very simple. Most are only three-step, point-and-click procedures that you should be able to handle without any difficulty.

3A-1: Adding Utility Buttons to the Panel

The GNOME Panel allows you to add a number of utility applets. Each of these has some specific function, such as tracking your stocks, telling you the weather, or performing some particular system-related function. To start out, let's add a clearly useful utility to the top panel: the Force Quit button. The Force Quit button lets you quickly and easily deal with non-responding windows.

Yes, it does happen on occasion: A window suddenly refuses to do anything. Regardless of what you want it to do or what it is supposed to be doing, it just sits there as if it is on strike (maybe it is). With just one click of the Force Quit button, your cursor becomes a powerful surgical instrument that will kill the window you click. You definitely don't want to be without this button, so here's how to add it to the panel:

1. Right-click any open space on the top panel.
2. From the popup menu, select **Add to Panel**, after which the Add to Panel window will appear.
3. In that window, click **Force Quit** once to highlight it, as I've done in Figure 3-5. Click the **Add** button, and then click **Close** to finish the job.

To reinforce what you've just learned how to do, let's add another utility to the panel: the Run Application panel applet. Once you start installing applications in Ubuntu, you will find that some of those applications do not automatically install program launchers in your Applications menu. This means that you have to open a Terminal window and type a command every time you want to run such programs, which can get old rather fast. The Run Application panel applet is one way around this problem.
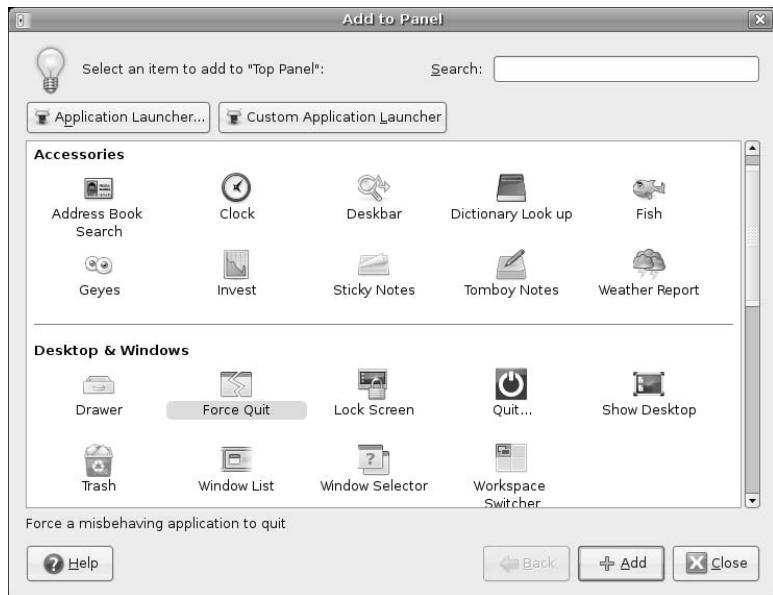
Figure 3-5: Adding launchers and utility applets to the GNOME Panel

To add the Run Application applet to the panel, just follow the same steps you used in adding the Force Quit button; but this time in step 3, highlight **Run Application** in the Add Launcher window instead of Force Quit.

NOTE *If you later decide not to keep the Run Application panel applet on the panel, or if you just prefer keyboard shortcuts to pointing and clicking, it is worth noting that you can also bring up the applet by pressing ALT-F2.*

3A-2: Adding Amusing Applets to the Panel

The GNOME Panel not only allows you to add very functional utilities, but it allows you to add quite seemingly useless amusements as well. In this part of the project, we will be adding two such amusements: Geyes and a little fish called Wanda.

At first glance, Wanda does little more than bat her tail around and spurt out a bubble or two. However, if you click on her, a window pops up in which Wanda will spew out quotes and offbeat one-liners.

To get a glimpse of Wanda in action, limited though that action may be, the steps are essentially the same as those in Project 3A-1 on page 35, but I'll run through them one more time:

1. Right-click any open space on the top panel.
2. From the popup menu, select **Add to Panel**, after which the Add to Panel window will appear.
3. In that window, click **Fish** once to highlight it, click the **Add** button, and then click **Close**.

Wanda will now appear on your panel, so go ahead and give her a click to see what she has to say.

Now you can add Geyes, which is a pair of eyes that follows your mouse cursor around as it moves about your desktop. Follow the same procedure, but click **Geyes** instead of Fish in step 3.

3A-3: Adding a Program Launcher to the Panel

Now let's move on to something a bit more practical—adding program launchers to the panel. While it is very easy to run an application by navigating through the Application menu, there are no doubt some applications that you will be using frequently enough to want easy access to them. OpenOffice.org Writer is probably one of those.

Method 1

There are a number of different ways to add a launcher to the panel, but let's start with the most conventional. To add a panel launcher for OpenOffice.org Writer, follow these steps:

1. Right-click any open space within the top panel.
2. Select **Add to Panel** in the popup menu to bring up the Add to Panel window.
3. In that window, click the **Application Launcher** button.
4. A new screen will then appear, showing the contents of the Application menu (Figure 3-6). Click the small arrow next to **Office** to expand that menu, and then scroll down and click **OpenOffice.org Word Processor** to highlight it.
5. Click the **Add** button, and then click **Close** to complete the process.
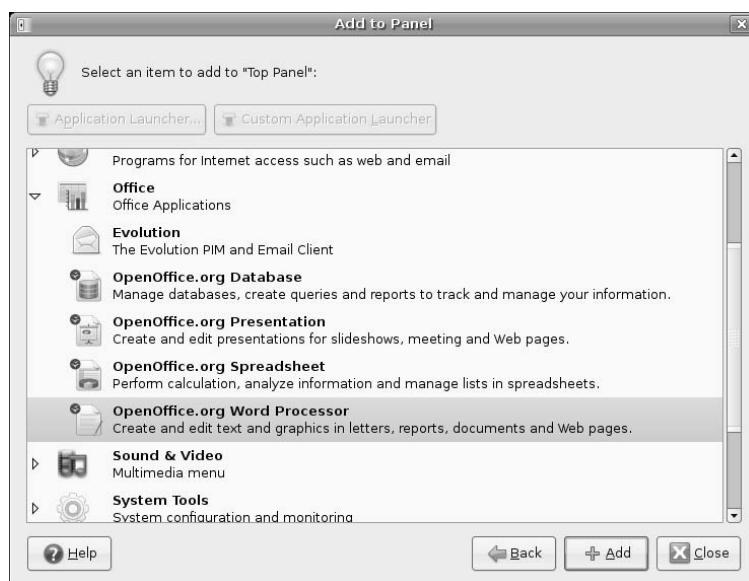
Figure 3-6: Adding an application launcher to the GNOME Panel

Method 2

There is another way to add program launchers to the panel, and it is actually a tad quicker. As an example, we'll add a launcher for the OpenOffice.org spreadsheet program, Calc. Here are the steps:

1. Go to the Applications menu, and navigate your way to (but do not click) **Office ▸ OpenOffice.org Spreadsheet**.
2. With your cursor over **OpenOffice.org Spreadsheet**, right-click.
3. In the popup menu that then appears, select (that's the usual ol' left-click this time) **Add this launcher to panel** (Figure 3-7). The Calc launcher will then appear in the panel.
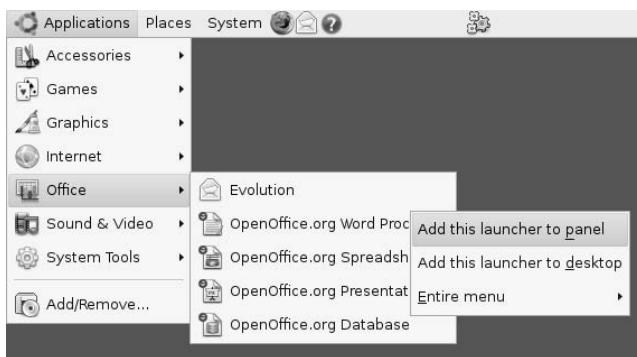
Figure 3-7: Another way to add application launchers to the panel

Method 3

Now that you've learned two ways to add application launchers to the panel, I might as well let you in on a third, even easier method. Just open a menu, select the item you want to add to the panel, and then drag it there. Well, it can't get much easier than that, eh?

3A-4: Changing Panel Launcher Icons

With your two new program launchers now added to the panel, you may come to feel that it is rather difficult to distinguish one from the other. Fortunately, you can change the icon for any launcher quite easily. To learn how to do it, let's address our immediate concerns with the two OpenOffice.org launchers. Here's what we need to do:

1. Right-click the first program launcher you added (Writer), and select **Properties** from the popup menu.
2. In the Launcher Properties window, click the **OpenOffice.org Word Processor** icon, which will bring up a Browse Icons window.
3. In that window, click the **Browse** button to bring up the Browse window.

4. At the top of that window, click the **icons** button. In the right pane, double-click the **hicolor** folder, click the **48x48** folder within the hicolor folder, and then click the **apps** folder within the 48x48 folder.

5. When you've finished all that folder clicking, click the **Open** button.

6. You will now be back at the Browse Icons window, which displays a different set of icons. Scroll down in that window until you find **openofficeorg22-writer.png**, and then click it once (Figure 3-8).
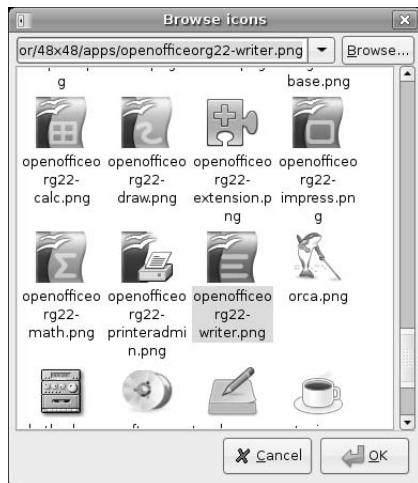
Figure 3-8: Selecting a new panel launcher icon

7. Click the **OK** button in that window, which will close it.

8. You will then be back at the Launcher Properties window, which should now look like that in Figure 3-9. If so, click **Close**.
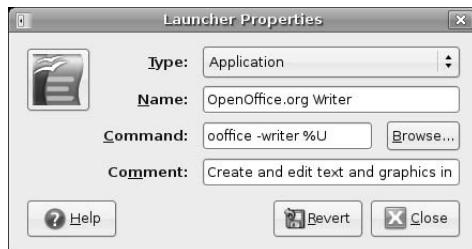
Figure 3-9: A Launcher Properties window

Once you have completed the transformation, follow essentially the same steps for the Calc launcher, but this time around you should select **openofficeorg22-calc.png** as the icon in step 3.

3A-5: Adding a Drawer to the Panel

One of the features I quite like about the GNOME Panel is the drawer. The *drawer* is a little drop-down panel that acts as the perfect location to place

launchers that you do not want to place in your GNOME Panel because of space considerations. This is also a handy location to place launchers for applications that you must normally run by typing a command in a Terminal window or via the Launch Application window, such as those you compile yourself from source code or that are run via scripts. You'll learn how to do this in Chapter 9. Of course, you can put anything you want there, including frequently used files.

Adding a drawer to your panel is very easy, and is basically the same procedure that you used to add the Force Kill button to the panel. Here is all you need to do:

1. Right-click any open space on the top panel.
2. From the popup menu, select **Add to Panel**, after which the Add to Panel window will appear.
3. In that window, click **Drawer** once to highlight it, and then click the **Add** button. Close the window by, quite logically, clicking **Close**.

3A-6: Adding Program Launchers to the Drawer

The drawer you've just added is, of course, empty at this stage, so let's put it to good use by adding launchers for three useful, yet less glamorous, system utilities. These are System Monitor, which allows you to view your computer's running applications and processes, memory and CPU usage, and storage device usage; Terminal, in which you can type and execute commands (slightly geeky, I admit, but very useful); and Synaptic Package Manager, which you can use to download and install applications.

Here's what you need to do:

1. Right-click the **drawer** applet in the panel, and select **Add to Drawer** in the popup menu.
2. In the Add to Drawer window that then appears (and looks and behaves exactly the same as the Add to Panel window), click the **Application Launcher** button.
3. In the next screen, click the small arrow next to Administration, scroll down and click **Synaptic Package Manager** to select it, and then click **Add**. The Synaptic Package Manager launcher will now be loaded into the drawer.
4. Add a launcher for the System Monitor by scrolling down a bit, clicking **System Monitor**, and then clicking **Add**.
5. Scroll back up to the Accessories category, and click the small arrow next to it.
6. Scroll down to **Terminal**, click it, and then click the **Add** button. You can now close the Add to Drawer window.

The three launchers should now be loaded in the drawer, so click the drawer to sneak a peek. Yours should look the same as mine in Figure 3-10.

Figure 3-10: Launchers in
a GNOME Panel drawer

3A-7: Adding the Entire Contents of a Menu to the Panel

If you find that you use the applications in a particular submenu of your Applications, Places, or System menus a lot, you can opt to add the entire menu to the panel as either a menu or as a drawer in a manner similar to the one you used in Project 3A-3's "Method 2" on page 38. To learn how to do this, let's add the Games submenu to the panel as a menu, and the Sound & Video submenu as a drawer. Here is what you need to do:

1. Add the Games menu to the panel by going to **Applications ▸ Games** and then right-clicking any of the launchers within that submenu.

2. In the popup menu that appears, select **Entire Menu ▸ Add this as menu to panel**.

3. Add the Sound & Video submenu to the panel as a drawer by going to **Applications ▸ Sound & Video** and then right-clicking any of the launchers you find there.

4. In the popup menu, select **Entire Menu ▸ Add this as drawer to panel**.

You should now have two new launchers on your panel with icons matching those found in the Applications menu next to the relevant items. Click each of these new panel entries to see how they work.

3A-8: Moving Things Around on the Panel

Well, now we've added all we are going to be adding to the panel. It may seem a little messy up there right now, so let's do a bit of housekeeping by moving things around. We will try to group things together somewhat thematically so as to make them easier to deal with.

Fortunately, you can move panel launchers quite easily by right-clicking the launcher in question, selecting Move from the popup menu, and then dragging the launcher to the spot you want to place it. Once the launcher is where you want it to be, click the launcher once, and it will stay there.

To get some practice with this moving business, let's move the launchers, menus, and drawers you added by placing them in the following order, from left to right: Applications, Places, System, Firefox, Mail, OpenOffice.org Writer, OpenOffice.org Calc, Sound & Video, Games, Drawer. Place the

remaining launchers at the right end of the panel, to the left of the update notification tool, in the following left-to-right order: Geyes, Wanda, Force Quit. Finally, place the Run Application panel applet by itself, midway between the two clusters of launchers.

When you've made all your changes, your panel should look pretty much like mine in Figure 3-11.

Figure 3-11: The GNOME Panel with the new launchers

More Panel Fun

In addition to the basic customization you did in Project 3A on page 35, you can do a lot more to change the look and feel of your panel. Of course, you can remove any of your launchers, drawers, or menus by right-clicking the item in question and then selecting **Remove From Panel** in the popup menu, but there are still more options. Most of these are available by right-clicking any open space in the panel and then selecting **Properties**, which will bring up the Panel Properties window.

From this window you can change the position of the panel, alter its size, change its color, or make it transparent—very cool. You can also set the panel so that it will automatically disappear when you are not using it and have it reappear when you bring your mouse cursor into the area where the panel normally resides. Don't feel afraid to play around and give things a try—that's half the fun!

Project 3B: Manipulating Menus

Now that you have learned about some of the cool and useful things you can do with your panel, let's now move on to the topic of menus. A very nice feature of GNOME is that it allows you to edit its menus. You can add launchers, remove items, move items, and even change the icons that appear within the menus. All in all, you have a lot of control over things, but for this project, we'll limit our work to two of these areas: changing icons and moving menu items.

3B-1: Changing Icons Within Menus

As you no doubt recall, one of the problems with the OpenOffice.org Writer and Calc launchers we added to the panel was that they shared rather similar icons. If you go to **Applications ▸ Office**, you will see that the icons for the various OpenOffice.org modules, although not the same, are a bit similar.

One might also argue that those icons are also rather frumpy, which may violate your personal sense of aesthetics as much as it does mine. To remedy this state of affairs, just follow these steps:

1. Right-click the **Applications** menu, and select **Edit Menus**.
2. In the menu editor window that then appears, click **Office** in the left pane. The contents of that menu will then appear in the right pane.
3. In the right pane of the window, right-click **OpenOffice.org Word Processor**, and then select **Properties** in the popup menu. A Launcher Properties window will then appear.
4. In that window, click the **Icon** button. Using the method described in Project 3A-4 on page 38, navigate to /usr/share/icons/hicolor/48x48/ apps in the Browse Icons window that then appears.
5. To make it easier on yourself when adding icons for the other Open-Office.org modules, drag the **apps** button at the top of the Browse Icons window to the left pane of the same window (Figure 3-12). From now on, you will be able to navigate directly to this folder by double-clicking that icon in the left pane.

Figure 3-12: Adding a folder to your list of places

6. Scroll down to and click **openoffice.org22-writer.png**, and then click **Open**.
7. The new icon should now appear in the Launcher Properties window. If it does, click **Close**, and you will be able to see the change in the menu editor window.

8. Repeat the process for each of the other OpenOffice.org icons, being sure to select the appropriate icons for each of the OpenOffice.org modules (openoffice.org22-math for the Formula module, for example). Once you're done, leave the menu editor window open to continue work on Project 3B-2.

NOTE *There is no alternative icon for* OpenOffice.org From Template*, so you can skip that particular item.*

3B-2: Changing the Order of Icons Within Menus

While we still have everything open to the Office menu, let's deal with what I consider to be another problem: the order of the items in the menu. While it is natural enough to have the email client Evolution at the top of the menu, it doesn't make sense to have what is arguably the most commonly used OpenOffice.org module, Writer, way down there at the bottom of the menu.

Remedying this situation is easy. Just click the **OpenOffice.org Word Processor** icon in the right pane of the menu editor window. Then click the up arrow on the right side of the window as many times as necessary until the Writer icon is right above the OpenOffice.org Base icon. Once you're done, your menu editor window should look something like mine in Figure 3-13.

Figure 3-13: Managing menus with the menu editor

Virtual Desktops

It is now time to discuss a rather unique and convenient feature of Linux: *virtual desktops.* Rather than listen to me babble on and on about this virtual desktop business, it is probably best to learn by just giving it a try.

In your GNOME Panel, click the **Wanda**, **OpenOffice.org Writer**, and **Firefox** launchers. You will then have three windows open in your present desktop, or workspace. Now look at the Workspace Switcher to the right of the bottom panel. There should be four boxes, with the one on the left, your present workspace, in brown. Click one of the other grayed-out boxes, and all your open windows will suddenly disappear.

Actually, nothing has really disappeared—you are just viewing a new desktop. All your other windows are still open and running in the previous desktop. In this second desktop you can open something else: Go to the **Applications** menu, and select **Games ▸ AisleRiot Solitaire**. The AisleRiot Solitaire card game will soon appear.

You now have windows open in two different desktops, and you can switch back and forth between them. To do so, just go to the Workspace Switcher in your panel and click the first grayed box, which will take you to your original desktop. Once you've done that, the box for the workspace you were just in will gray, and you can then click that one to go back to your game desktop.

As you can imagine, this feature has some potential benefits for you, in addition to helping you avoid clutter. Just imagine that you are at work typing some long document in OpenOffice.org Writer. Eventually, you get tired and decide to goof off a bit by playing a game, such as Mines, for a while. To do this, you switch to another desktop where you open and play the game. A bit later, when you notice your boss making the rounds of the office, you simply switch back to the first desktop so that you look busy when he walks by and asks, "Keeping yourself busy, Boaz?"

Phew!

By the way, you can also switch between virtual desktops by simultaneously pressing and holding CTRL-ALT and then pressing your left and right cursor keys to move to your targeted desktop.

Moving Running Applications Between Virtual Desktops

So what happens if, let's say, you are running OpenOffice.org Writer in one workspace and the GIMP in another, but suddenly think that it would be handy to have them both running in the same workspace? Do you quit the GIMP and start it up again in the other desktop? Fortunately, things are much simpler than that, and there are actually two ways to get the job done.

The first of these ways is to right-click on the title bar of the window you want to move, and then select Move to Workspace Left or Move to Workspace Right. I find it less confusing to select Move to Another Workspace, and then select the workspace I want to move the window to by number: Workspace 1, Workspace 2, and so on.

Another way to move a window from one workspace to another is via the Workspace Switcher on the bottom panel. Within the four boxes of the Workspace Switcher, each representing a different workspace, tiny icons will appear for each of the windows you have open in a given workspace. The icons, as you can see in Figure 3-14, will even resemble the shape of the actual windows themselves. You can use these icons to move windows from one workspace to another using a basic drag-and-drop maneuver.

Figure 3-14: The Workspace
Switcher shows icons of windows
you have open in each workspace.

For example, let's say you want to move Firefox from workspace 2 to workspace 1. To do this, you would simply click the tiny icon within the second box of the Workspace Switcher, and drag it to the first box, after which the icon would appear there because the Firefox window itself would have moved there.

Finally, if you like to keep your hands more on your keyboard than on your mouse, you can also move a window from workspace to workspace by using hotkeys. With the window you want to move active (on top of the pile, so to speak), press and hold SHIFT-CTRL-ALT, and then use the left and right cursor keys to move the window to the desired desktop.

Wanda Revisited—GNOME Easter Eggs

Well, now that we've finished with our work in this chapter, it's time to goof around a bit by revisiting our precocious piscean pal, Wanda. Knowing what you now do about Wanda the Fish, you might find it odd for me to start talking about her again, but Wanda has a few more tricks beneath her fins. In fact, she is a good means by which to introduce two of GNOME's most famous Easter eggs. *Easter eggs*, in case you don't know, are hidden snippets of code that programmers seem to love to sneak into their programs. They are usually pretty useless things, but they can be found in all operating systems, numerous applications, and even on DVDs (to find out more about those, go to www.dvdeastereggs.com).

A good example of an Easter egg is my first encounter with one on my first Mac, an ancient Mac SE with a whopping 2MB of RAM. On that machine, you could bring up an image (or was it a slideshow?) of the Mac SE development team by pressing the seldom used debug key on the side of the machine and then typing G 41D89A. Pretty cool, I guess, but I would never ever have stumbled upon it had I not read about it in some magazine.

As my example shows, accessing these Easter eggs usually requires some unusual maneuvers, ones that you would never perform in the normal course of things. To see a Wanda-related Easter egg in action, click the **Run Application** button you just added to the panel, type `gegls from outer space` in the Command box, and then click **Run**. You will then see an odd little game of the Space Invaders genre, shown in Figure 3-15, in which Wanda defends our beloved planet from . . . well, gegls, I guess.

To try out the other Wanda Easter egg, open the Run Application panel applet again, but this time type `free the fish`, and then click **Run**. Wanda will now appear swimming around your desktop. If you then click directly on her, she will swim away and out of the picture . . . but she'll be back.
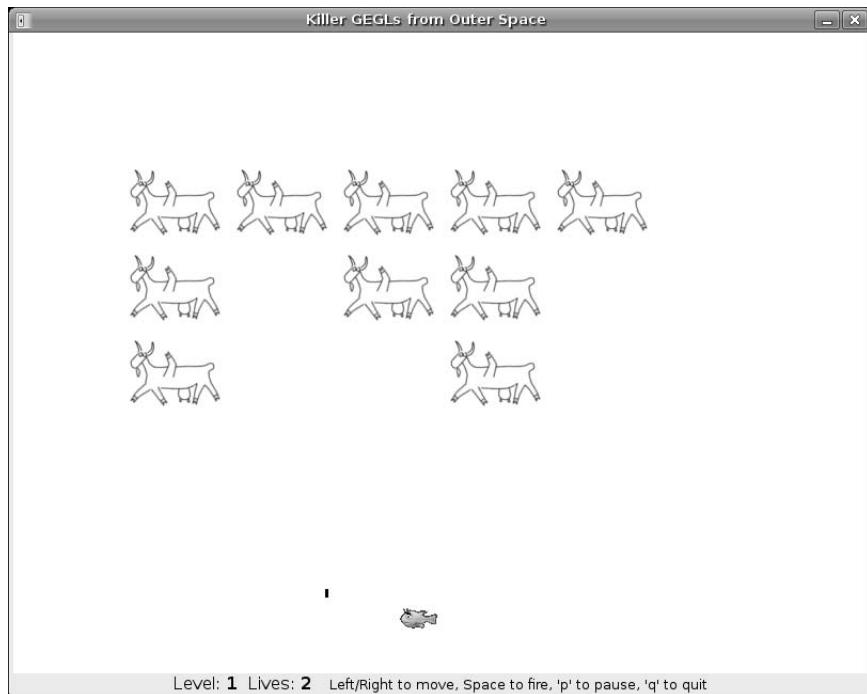
Figure 3-15: One of GNOME's Wanda-related Easter eggs

To put an end to Wanda's comings and goings, you will need to restart the GNOME Panel. There several rather inelegant ways of going about this, but for now we'll do it by opening the Run Application panel applet again, typing `killall gnome-panel`, and then clicking **Run**. Your panels will disappear for a second or two but will shortly reappear. Wanda, however, will be gone.

Shutting Down

Now that you know your desktop environment so well, you may feel like calling it a day and shutting down your machine. To do so, just go to the **System** menu and select **Log Out**. Your screen will darken, and then a small window (Figure 3-16) will appear with seven choices to choose from: Log Out, Lock Screen, Switch User, Suspend, Hibernate, Restart, and Shut Down (Hibernate is not an option when running a live session from the Desktop CD). Click **Shut Down**, and the shutdown process will begin. If, however, nothing seems to happen within a few seconds of clicking **OK**, press CTRL-ALT-BACKSPACE in unison, which will bring you to the login screen you saw at the beginning of the chapter (Figure 3-1). Once there, click **Shut Down** at the bottom of the screen. This will bring up a small window asking you if you are sure you want to shut down. Just click **OK**, and shutdown will commence.
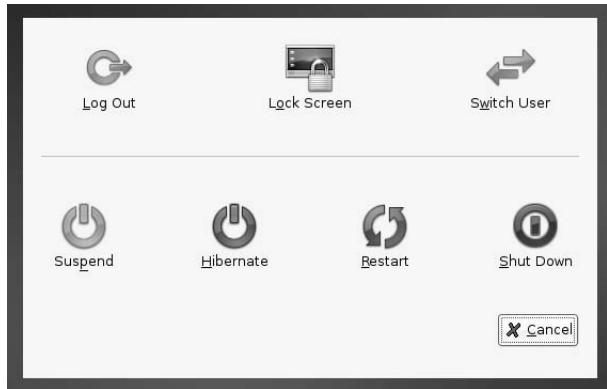
Figure 3-16: GNOME's logout window

The actual shutdown will take a few seconds as the system closes its various services. When it is all done, the system should power down your computer as well, in which case you are done. On a few machines, however, the system cannot power down your machine. You will know if this is so in your case because all screen activity will come to an end. If you get to that point and nothing else happens for 15 seconds or so, then just power down the machine manually by pressing the power button. It is completely safe to do so at that point.

4

MORE THAN WEBBED FEET

The Internet, Linux Style

These days, average home computer users spend more time surfing the Web and writing email messages than doing just about anything else. Even if you're not much of a surfer, there are still numerous other applications that aren't really Internet applications per se but that still make use of the Internet in some way, such as gathering song and album information when you rip audio CDs to create MP3 files. Having a computer that isn't hooked up to the Internet is like buying a new Maserati and then refusing to take it out of the garage.

Of course, how you connect to the Internet depends on your hardware and provider. There are a number of possibilities in this area, including high-speed local area networks (LANs), cable modems, and ADSL connections from phone companies. Most computers also have an internal 56Kb/s modem or can be connected to external dial-up modems for slower connections over regular phone lines. Depending on what you've got, setting things up on your system should prove a cinch in the case of LAN connections and any others that make use of your Ethernet port (such as cable modems),

possibly a bit more work in the case of wireless connections, and sometimes a bit of a challenge when it comes to the ol' dial-up connections. In this chapter, you will learn how to set up these connections and learn a bit about what Linux has to offer in terms of the most commonly used Internet applications—your web browser and email client.

Setting Up a High-Speed Connection

If you have a high-speed Internet connection from your cable television company, or if you are connected to the Internet by a LAN at your office, you are really in luck, because these setups are probably the easiest to deal with. Most likely, all you have to do is connect the Ethernet cable from the wall (if you are using a LAN) or from your cable modem to the port of the network card on your machine. If you're using a wireless router, then the Ethernet cable will connect to the wide area network (WAN) port of your wireless router. After that, once you start up your machine, you should be ready to go without any further settings to fool with. If you like, you can see if you're connected by opening Firefox (click your Firefox launcher in the top menu, right next to the System menu), and then once it starts up, trying to navigate to a common site, such as www.yahoo.com. If the site comes up, you know you're all set.

If you have a problem getting online, and you are trying to connect to via a LAN or cable modem, the problem could very likely be that your network card is not supported by Linux. This is relatively rare, but fortunately, easily remedied (by replacing it).

The problem could also be that your network or service provider does not automatically assign addresses via Dynamic Host Configuration Protocol (DHCP). *DHCP* is a means by which your Internet provider can automatically (dynamically) provide your system with the configuration information it needs in order to connect to the Internet. If your provider does not utilize DHCP, you will have to get the necessary information about settings from the network administrator or service provider and enter the settings yourself.

Setting Up a Cable or Ethernet Connection for Providers Not Utilizing DHCP

To input your cable or Ethernet settings yourself, first get the settings you need from your network administrator or Internet provider, and then perform the following steps:

1. From the **System** menu, select **Administration ▸ Network**.
2. You will be prompted for your password at this point, so type it (that would simply be your user password), and click **Continue**.
3. The Network Settings window (shown in Figure 4-1) will soon appear, and when it does, select **Wired connection** by clicking it once (make sure the box to the left is checked). Once you're done, click the **Properties** button.
4. In the Settings window that appears, change the Configuration selection from Automatic configuration (DHCP) to **Static IP address**.
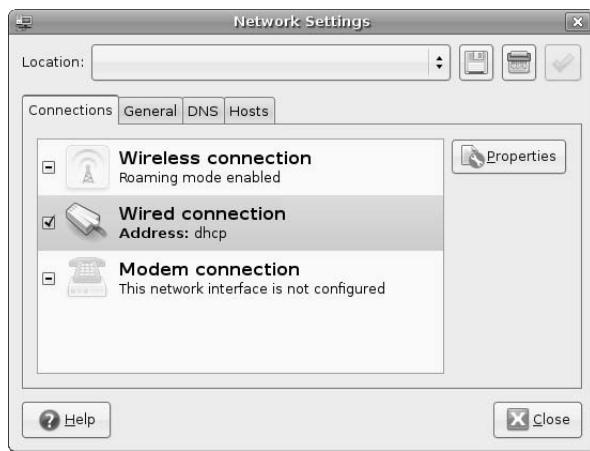
Figure 4-1: The Network Settings window

5. The three text boxes shown in Figure 4-2 will become active, thus allowing you to enter the information provided to you by your Internet provider or network administrator. Once you've entered the settings, click the **OK** button.
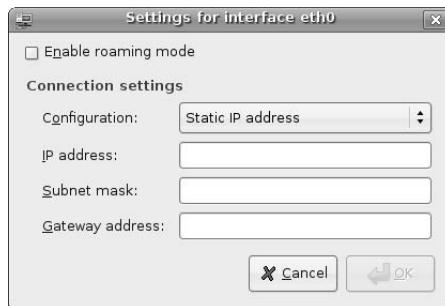
Figure 4-2: Manually inputting network
IP settings

6. Your system will then try to connect to the Internet using the settings you have provided and will show its progress in the Network Manager applet, which will look . . . well, blue and kind of spacey. Once the connection is made, a message will appear directly below the applet telling you so. You can then close the Network Settings window and surf to your heart's content.

NOTE *If you are wondering what Internet protocol (IP) and domain name service (DNS) are all about, you can simply think of them in this way: the DNS translates the easy-to-remember URLs that you have come to know, such as www.yahoo.com, into numerical, or IP, addresses that the Internet can understand. The address http://www.yahoo.com thus becomes http://216.109.118.68. You can type the numerical version into your browser later to see for yourself.*

Setting Up a Wireless Connection

Laptops have made the computer a more versatile tool. With the right wireless hardware, you can now surf the Web just about anywhere you can catch a wave, so to speak. Whether you happen to be at your breakfast table, on your backyard deck, in the library of your university, or at your local Starbucks, Port City Java, or Dunkin' Donuts, you can now go online without having to physically hook up your computer to anything. Fortunately, the process of setting up wireless networking in Ubuntu is quite easy—little different than what you just read in "Setting Up a High-Speed Connection" on page 50. In fact, as I just mentioned, chances are that everything will just work from the get-go.

Hardware

If you lead a solely wired existence or are just inexperienced in this particular area, there are a few things worth knowing. To get started, you need to have the right hardware. If you just want to go wireless outside of the home, then all you need is a wireless network interface card (NIC). These are often built in to modern laptops, while for others they are usually add-ons in the form of cards that pop into the PCMCIA slot on the side of your computer (as seen on the right of Figure 4-3).

Figure 4-3: All you need for a WiFi setup—an access point and a network interface card

There are, however, also some NICs that plug into one of your machine's USB ports or, in the case of desktop models, one of its PCI slots. While support for NICs of this type has improved through the last couple of years, there are still gaps, and some can be rather tricky to deal with. If you're looking for a sure thing, Centrino IPW-2100 and IPW-2200 cards are definitely supported by Ubuntu. As for NICs of the built-in or PC card variety, you will find support much better, but even then you have to make sure, or at least hope, that the card you are using is Linux compatible. No matter what NIC you are wondering about, the easiest way to find out whether or not it will work is to just

try it out. The Linux kernel now comes with several wireless drivers built in, and Ubuntu updates often provide new ones, so if you use a card that is compatible with one of those drivers, things will be smooth sailing. If things don't seem to work, check the Ubuntu forums to find a driver that is compatible or to see if anyone has experience with your particular card.

If you want to set up a wireless system in your home or office, then you will also need to get a wireless access point (WAP), shown on the left of Figure 4-3. Fortunately, Linux compatibility is not really much of an issue in this department, because the access point doesn't physically interface with your computer.

Unfortunately (for Linux users, anyway), there are a few WAPs that require you to use Internet Explorer to handle their setup chores. Because a Linux version of Internet Explorer does not exist, you will find yourself stuck unless you have a Mac or Windows machine somewhere else in the house to complete the job. Worse yet is the fact that some machines do not support browser-based settings at all. Instead, they require you to install special Windows software to do the job. When selecting hardware for your wireless setup, it is obviously best to stay clear of WAPs of this kind.
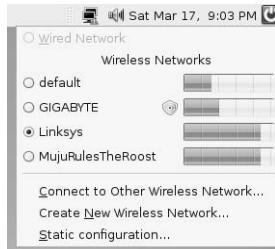
Activating Your Wireless Card

Setting up a Linux-compatible wireless PC card is actually relatively simple. First connect your WAP to your Internet source, and then turn on the WAP. Once it is up and running, plug your wireless NIC into the PCMCIA slot on your laptop, unless, of course, your NIC is built in, in which case you can forgo this step. Any LEDs on the external card will most likely light up at this time.

If your WAP was on when you booted up your computer, and your wireless card was in place during bootup, you probably don't need to do much else to access the Internet. Just follow these steps:

1. Click the **Network Manager** applet to reveal a drop-down menu showing the wireless signals that are present in your vicinity and the strength of each signal (Figure 4-4).

2. From that menu, select the signal for your WAP (or any other wave you are entitled to latch onto) by clicking it.

3. Your system will then try to connect to the Internet, and once a connection is made, you will be notified of it in a message below the Network Manager applet. The Network Manager applet icon will morph into a small signal-strength indicator. Pretty handy.

NOTE   *Some WAPs employ an added level of security in the form of Wired Equivalent Privacy (WEP) or, more recently (and more securely), a WEP key. If the WAP to which you are trying to connect requires such a key, a window will appear prompting you to enter it. If you don't happen to know the key, then chances are you are not supposed to be making the connection in the first place. Naughty, naughty.*

Figure 4-4: The Network Manager applet shows you available wireless networks and their signal strengths

If it turns out that you were not successful in making your wireless connection (the system will tell you if that's the case), there are a few possibilities beyond the annoying non-compatible-card scenario. First of all, double-check to make sure your card is actually physically turned on. I know that sounds dumb, but I have been guilty of this oversight myself on numerous occasions.

Another possibility is that your network does not utilize DCHP, in which case you will have to manually input your connection settings as provided by your network administrator or service provider. Armed with that information, simply follow the steps listed earlier in "Setting Up a Cable or Ethernet Connection for Providers Not Utilizing DHCP" on page 50. Just be sure to click **Wireless connection** instead of Wired connection in Step 3.

Releasing and Renewing Your Wireless Connection

There will be times when you want to turn your wireless card off, such as when you use your laptop on an airplane, or when you just want to flush the IP address in your network settings (*release*) and update the settings with the IP address of a new network (*renew*), as you might when moving your laptop from one wireless hotspot to another without rebooting your computer.

This last point might seem a bit mysterious to you, so I will explain things briefly. When you boot up your computer, your wireless card (NIC) performs a scan of available networks (WAPs) in order to see which one it can connect to. Once it finds a network, it gathers an IP address from that network via DHCP in order to allow you to access the Internet. Utilizing release and renew allows you to change WAPs without having to reboot your computer or when your machine, for whatever reason, just can't seem to make the change on its own.

To perform a quick and easy release/renew operation, click the **Network Manager** applet once to reveal the list of available wireless networks (i.e., signals). In that list, click the network to which you are connected. Within a second or two, you will be advised that you have been disconnected. A few seconds after that, you will be advised that a connection to that same network has been established. Wow.

Okay, so how do you turn off your wireless card, particularly if you don't have a physical way to do it? Just right-click the **Network Manager** applet, and then deselect **Enable Wireless**. All quite simple.

Oh, and if you want to turn that card back on again, just get back to that menu, and select **Enable Wireless** again.

Setting Up a Dial-Up Connection

Although much of the world is moving to high-speed Internet connections, many of you may still be using dial-up Internet connections, which means that you need to have a traditional dial-up modem to reach beyond your box to the outside world. In case you broadband surf-gods have forgotten, modems are those wonderful machines that whistle, chime, screech, and spit whenever you dial up your Internet provider. I suppose you could think of them as noisy telephones in need of a good burp.

The good news is that setting up your dial-up connection is a simple process that much resembles what I've covered thus far for other connection types. Now here comes the bad news—in the world of Linux, very few internal modems are supported. The main reason behind this compatibility problem is that most built-in modems are software dependent, and the software they depend on is part of, or designed for, Windows. Such modems are thus called *Winmodems.*

Of course, the Linux community has been working on ways to deal with these Winmodem beasts so that they will work with Linux systems. Though support for the wide variety of Winmodem models out there is still rather spotty, things are better than before, so you might just luck out. My advice is to hold off on the wondering and worrying and just give your modem a try to see if it works. If it does, then you're all set. If it doesn't, well, then you've got some options open to you, but at least you haven't done any damage to your system.

With all that intro-babble out of the way, let's get down to the steps for setting up your dial-up connection. First, get the setting information you need from your Internet provider. Most providers gear their operations to Windows and often Mac users, and very few offer Linux support, other than NeanderTech and a few others. Still, there is no technical reason for your Linux system not to work via their setup, so just nag and push them until they give you the information you want. After that, make sure your modem is connected to a live telephone connection—for example, the phone jack in your wall—and then do the following:

1. Go to the **System** menu, and select **Administration ▸ Network**. After typing your password when prompted to do so, the Network Settings window will appear.

2. In that window, click **Modem connection** to select it, and then click the **Properties** button.

3. When the Interface properties window appears, check the box next to the words *Enable this connection.*

4. In the Internet Service Provider Data section of that window, type the dial-up connection phone number as given by your provider, and the dial prefix, if necessary, to dial out of your current phone system (some offices require that you first dial 9, for example, to make a call out of the office).

5. Type your username and password, as given by your provider, in the boxes in the Account data section of the window.

6. Click the **Modem** tab, and then select either **Tones** or **Pulses** in the Dial Type section of the window, depending on the type of dialing scheme you have for your telephone service. Most are tone dialers these days, so you can choose that if you're not sure. You can generally figure it out yourself by paying attention to the sounds made when you dial a call. If you hear a series of different tones—almost a melody that you can recognize—then you have tone dialing.

7. Select **Off** or **On** in the Volume section of that tab, depending on whether or not you want to hear your modem doing its dialing. While testing things out, it is a good idea to select **On** so that you can hear what is (or is not) happening.

8. In the same tab, now click the **Autodetect** button. The system should then perform a scan of your ports as it looks for a live modem. Once it finds one, it will specify the correct port in the box labeled *Modem port.* If it doesn't find one, well . . . that's not a particularly good omen in terms of the usability of your present modem, but you can still give it a try by picking a port on your own. Start out with /dev/modem, and if that doesn't work after going through steps 9 and 10, try out the others one by one before finally opting for more dramatic measures.

9. Once your modem port has been found (or you've chosen one on your own), click the **Options** tab, and then check the box next to the words *Set modem as default route to internet.* Once you're done, click the **OK** button.

10. In the Network Settings window, be sure to check the box next to the words *Modem connection* before closing the window.

Once the steps are completed and assuming your modem and system are in loving compatibility, you should then hear your modem begin its dialing, spitting, and churning sequence as it makes the connection with your provider. Once it does, you can go on and start your browsing, emailing, or whatever else it is you do online. When you are ready to end your Internet session, click the **Deactivate** button to bring it all to a close.

What to Do if Your Modem Isn't Compatible

What I've just described is pretty much a best-case scenario. What happens, however, if your modem and Ubuntu do not see eye to eye? Well, there are a few options. My first, more radical suggestion, is to dump those extra movie channels from your cable service (nothing all that great on them anyway) and pay for cable Internet service instead. If that is not an option, then you have two ways to go. One is to try to geek around with your modem to see if

you can get it to work. This is a slightly more complicated process, but there are instructions at https://wiki.ubuntu.com/DialupModemHowto. If you are faint of heart or a novice user, you may find what the process entails to be a bit more than you're willing to deal with.

The simpler, although costlier, way to get your modem to work is to change your modem. In this case, changing to a true hardware modem. *Hardware modems*, in case you are not familiar with the term, are modems that are not software dependent and thus work with whatever system you happen to use with them. You can think of them as telephones without a handset.

Such modems come in two forms: internal and external. As for the internals, the USRobotics models 56K V.92 Performance Pro Modem (internal slot) and 56 PC Card Modem (PC card slot, for laptops) are true hardware modems that are easily available and are said to work. You can check out the USRobotics site (www.usr.com) for more information on these models, though checking the Ubuntu forums for suggestions is always a good idea.

External Dial-Up Modems

Perhaps the safest of all solutions is to buy an external serial modem. An external modem sits in a box outside your computer, and it connects to the serial port in the back of the computer, which is usually the only connector with little prongs in it (see Figure 4-5). Because the modem doesn't use your operating system to operate, it does not tie up system resources while it's busy, which may result in a possible pickup in computer speed.
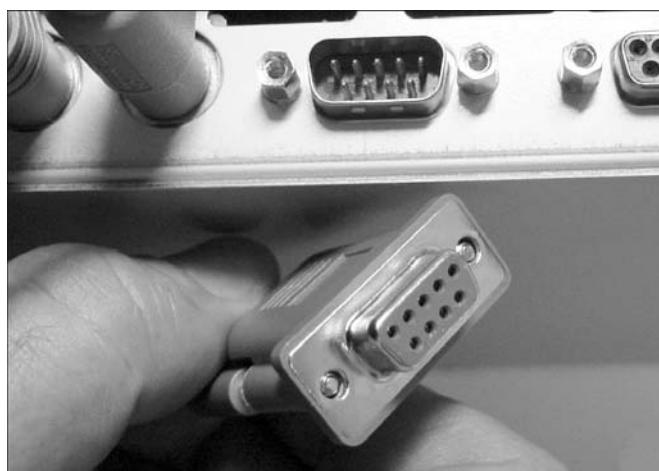
Figure 4-5: Serial port and connector

Most external serial modems should work with your system, or at least that is what most people will tell you. If you are worried and are looking for a sure thing, Zoom Telephonics (www.zoom.com) makes an external serial modem that is compatible with Linux, and they say so right on their website. The USRobotics 56K V.92 External Faxmodem is also said to work, though I haven't tried out this model myself.

If you find another model that you think will do the trick, before you commit to it by slapping down the cash, do a Yahoo! or Google search with that modem's make and model number, along with the word *linux*, and see what search results you get. Of course, you can also try out one of the Linux forums and ask about the modem make and model there. There are a lot of people in the same boat, so you are sure to get plenty of opinions and advice.

Firefox: Your Internet Browser

Now that you are connected to the Internet, you no doubt want to get down to some cyberspace discovery and exploration, and the most commonly used means of doing that is with a web browser. The default web browser in your Ubuntu system is Firefox, which is enjoying increasing popularity in not only the Linux world, but in the Windows and Mac worlds as well. Chances are you are already a Firefox user, but if you are not, then you needn't worry—things work more or less the same in all browsers. That being the case, you should be able to use Firefox's basic features without any instruction. Of course, there are some features that do distinguish Firefox from its competition, so I will mention those.

Controlling Browser Window Clutter with Tabs

Usually when you click a link on a web page, the new page opens in the same window. On some pages, links are coded so that the new page opens in a new, separate window, or maybe you occasionally opt for opening a link in a new window by right-clicking the link and then selecting the Open Link in New Window option. This can be very useful; however, once you have more than a few browser windows open, it gets sort of hard to find what you're looking for in all those open windows. It can also slow things down a bit.

This is where Firefox's tab feature comes in handy. To see how it works, try it out yourself right here and now. Open your Firefox browser by clicking the launcher on the top GNOME Panel (or going to **Applications ▸ Internet ▸ Firefox Web Browser**); then Google the word *nyckelharpa* using Firefox's handy search box, which is next to the word *Go* at the top-right corner of the browser window (see Figure 4-6). By default, Firefox will perform searches for keywords entered in the search box using Google. You can, if you like, select other search engines by clicking the G icon in the search box and then making your selection. Amazon.com, eBay, and Yahoo! are available, to name a few, and you can even add others. For now, however, let's stick to Google for our present search, by typing `nyckelharpa` in that search box. Once you've finished typing, press the ENTER key, after which a page of Google results should appear in the main pain of the Firefox window.

NOTE   *While tabbed browsing is no longer as unique as it once was (Safari has the feature built in, though you have to enable it yourself, and Internet Explorer now has the same functionality available as a downloadable add-on), it is implemented and enabled by default in Firefox.*
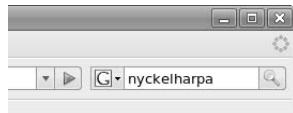
*Figure 4-6: Performing a Google
search from the Firefox search box*

The top result should be the American Nyckelharpa Association, and
you are now going to open that page in a new tab, rather than in the same or
a new window. To do that, right-click the link, and in the popup menu that
appears, select **Open Link in New Tab**. You can, if you prefer, make things
a tad easier and dispense with the popup menu selection step by simply
clicking the link with both mouse buttons simultaneously or by holding down
the CTRL key as you click the link. Either way, the new page will appear in a
new tab, while your original page of search results remains, ready and
waiting in the other tab (see Figure 4-7). I am pretty confident in saying
that, once you get used to this feature, you will wonder how you ever got
along without it.

Figure 4-7: A link opened in a new tab in Firefox

Other Firefox Features: Popup Manager

Firefox has a number of other useful features. One is its Popup Manager,
which suppresses those annoying popup windows that often appear when
you access a new web page. You can enable or disable this feature from the
Preferences window (**Edit ▸ Preferences**) by clicking the **Web Features** icon
in the left pane of that window and then checking or unchecking the box
next to the words *Block Popup Windows*. You can also permit certain sites to

provide popup windows (some popups are not only useful, but necessary for the correct functioning of a site) by clicking the **Allowed Sites** button next to that Block Popup Windows entry and inputting the URL for the site in question.

Project 4A: Installing Firefox Extensions

One of the coolest things about Firefox is that it allows you to further expand its functionality by adding various extensions. These extensions include all sort of things; many are quite functional, while others are just plain fun and goofy. They range from blog-writing tools to image viewers. For this project, however, we will be installing a cool weather station of sorts, called Forecastfox, that allows you to view not only the current weather conditions in your area (or any other area of your choosing), but also a two-day local forecast, Doppler radar maps, and more—all from AccuWeather.com. All of this is available at the click of a button from the Status or Menu bars, or the Bookmarks or Navigation toolbars—your choice (Figure 4-8).

Figure 4-8: The Forecastfox extension installed in Firefox

4A-1: Downloading and Installing the Forecastfox Extension

To get started with the process of installing any Firefox extension, you have to first, quite logically, find and download one. To do this, go to the Firefox **Tools** menu, and select **Add-ons**. The Add-ons window will then appear (Figure 4-9), showing you the extensions, themes, and language packs you already have installed, which at this point should be zero. To add something to this rather empty state of affairs, click the **Get Extensions** link at the bottom-right corner of the window, which will bring your browser to the Firefox Add-ons site.

At this point, you would normally browse for things that seem of interest to you, check to make sure they don't have any special requirements (such as Microsoft Windows—a few do), and then download and install the extension that has struck your fancy. In this case, we already know what we are going to install, so I'll just tell you where to find it.
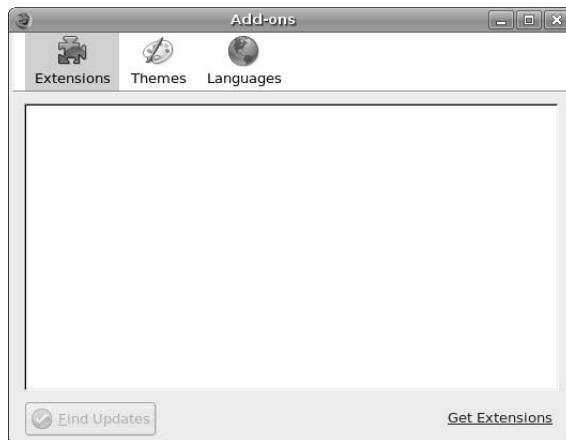
Figure 4-9: Seeing the extensions you have and getting more in Firefox

To find the Forecastfox extension, do a search for it by typing `forecastfox` in the search box on the Extensions page. Once you've found it, click the **Forecastfox** heading/link to go to that specific page, and once you're there, click the **Install Now** button. After that, a Software Installation window will appear. Click the **Install Now** button that appears in that window, and then . . . just wait. Sometimes the download will take a bit of time, sometimes it won't. Either way, just be patient. The extension will be in the process of downloading even if it seems as if nothing is happening, and when it is done, it will tell you so in the Add-ons window (Figure 4-10).
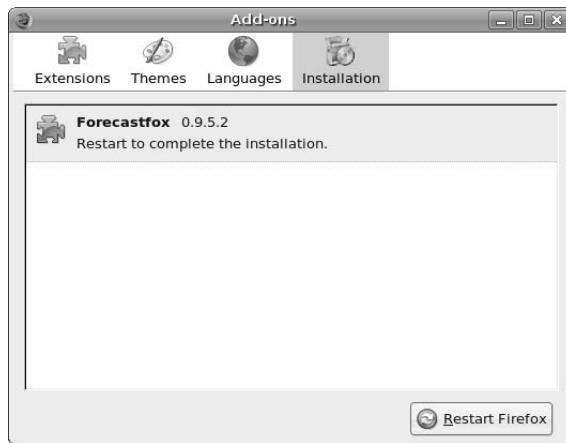
Figure 4-10: Firefox shows you the extensions you've added

A restart of Firefox is necessary in order to complete the installation, and that is easily accomplished by clicking the **Restart Firefox** button in the bottom-right corner of the Add-ons window.

4A-2: Setting Up the Forecastfox Extension

When Firefox first starts up after you've installed the Forecastfox extension, you will see the Forecastfox Options window (Figure 4-11). In that window, type your zip or postal code (or that of any other area for which you want weather information) in the text box next to the word *Code*. If you're not sure what the zip or postal code for your desired locale is, click the **Find Code** button to select that location by name.
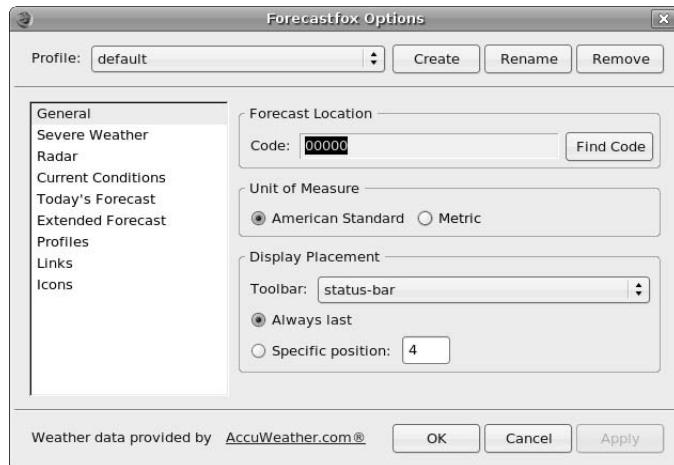
Figure 4-11: The Forecastfox Options window

In the Forecastfox Options window, you can also choose whether you want the temperatures shown in Fahrenheit or Celsius and where you want the information to appear in your browser window. The Status bar at the bottom of the browser window seems to be the least intrusive location, so that is what I've chosen. Of course, you are free to place it where you like. When you're done, click the **Apply** button and then the **OK** button, and you'll be on your way to intense meteorological entertainment.

Project 4B: Installing the Flash Plugin

While installing any of Firefox's available extensions can be cool, there are certain near-essentials that can be added in the form of *plugins,* although perhaps I am splitting hairs a bit in terms of the difference between the two. The most frequently used browser plugin for any system is Adobe's Flash Player. Flash provides websites with all sorts of exciting multimedia effects and capabilities, and as a result, it is utilized in an ever-increasing number of sites. In fact, chances are you've probably seen numerous Flash-enhanced pages without even being aware of it. For example, if you've ever watched a video on YouTube, you've enjoyed the wonders of Flash.

How do you get the Flash Player plugin? Well, the easiest way is to simply find a page that uses Flash, and install the plugin from there—the site will sense that you don't have the plugin and guide you through the installation. If you're not sure how to go about things, just follow these steps:

1. First of all, find a site that uses Flash. If you don't know of one off hand, try the site for northern Michigan's Finlandia University (www.finlandia .edu/flashpage.html), which is the site I will be using in these examples.

NOTE *Although I mentioned YouTube as a site that utilizes Flash, do not use that site here. The YouTube site, sensing that you do not have Flash already installed, will direct you to a different page for installing the Flash Player, and for Linux users anyway, this ends up being a much more cumbersome way of doing things.*

2. Once you have arrived at the Finlandia University homepage, you will notice a big blank area near the top with a green puzzle piece and a yellow message strip telling you that you're missing some plugins (Figure 4-12). To get started, click the **Install Missing Plugins** button.
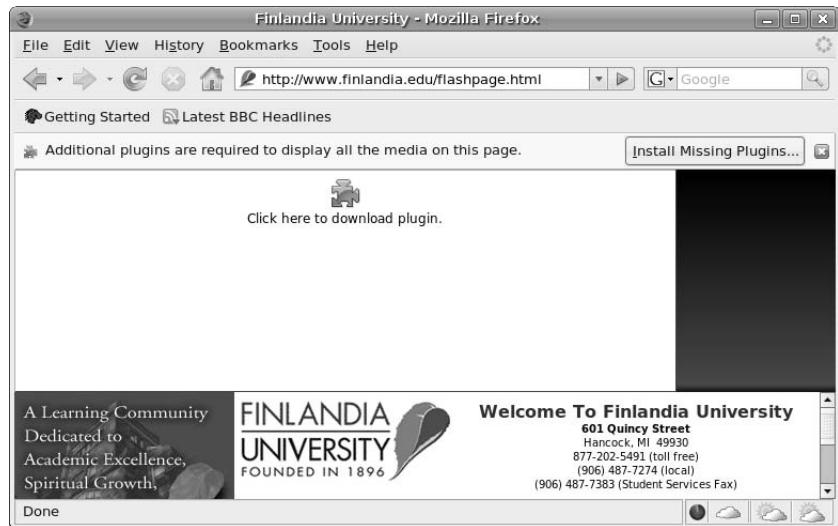
Figure 4-12: A site utilizing Flash on a system without the Flash Player plugin installed

3. A Plugin Finder Service window will now appear showing the plugin you are about to install: Adobe Flash Player. Click **Next**.

4. Now you will be shown the Adobe software license agreement. Give it a read, select **I agree**, and then click the **Next** button.

5. The plugin will then be downloaded and installed. When a message appears telling you that the installation is complete, click the **Finish** button, and Firefox will refresh the page you were viewing, which now displays Flash content where only moments before there had been none (Figure 4-13).
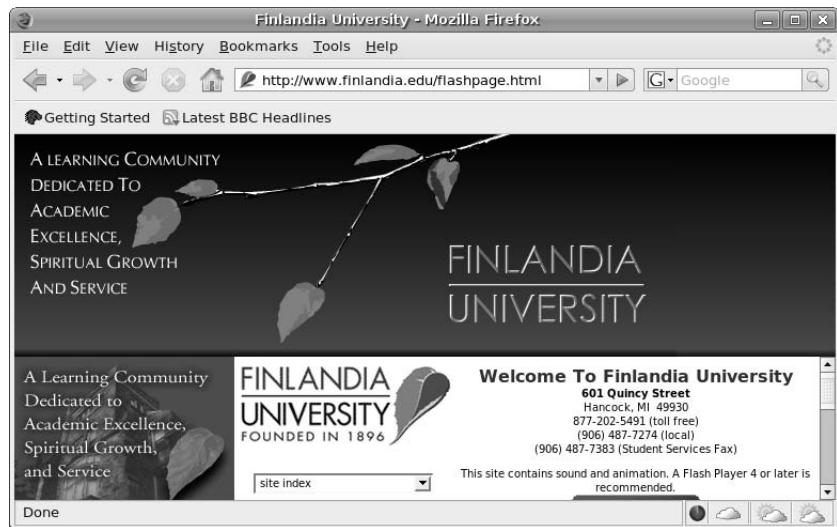
Figure 4-13: A Flash page after installation of the Flash Player plugin

Email with Evolution

Evolution is the default email program in Ubuntu, and it could probably best be described as a better-groomed, spunkier clone of Microsoft Outlook (see Figure 4-14). It allows you to send and receive mail, make appointments, and keep a list of tasks. It can also filter junk mail, which is a necessity these days, and even synchronize with your PalmPilot, if you still have one of those. Also, if such things are important to you, it is a much more handsome program to look at than Outlook.

Figure 4-14: Ubuntu's default email client—Evolution

To use Evolution, just click the email launcher on the top panel or go to the **Applications** menu and select **Internet ▸ Evolution Mail**. When you first run Evolution, you will be greeted by a setup wizard, so have the account details you received from your Internet service provider handy. These should consist of your POP host address for receiving mail, your SMTP host address for sending mail, and your mail password, which is very often different from your Internet logon password. Your mail password is not actually entered during the various wizard steps, so check the **Remember this password** checkbox when filling in the POP details. When you first connect to your mail server, you will be prompted for your mail password, so you can type it at that time, and you won't have to deal with it again.

An Email Alternative: Thunderbird

Evolution is, without a doubt, the most popular email software in the Linux world, but despite its obvious attractions and popularity, I have to say that I don't much like it, though I can't quite put a finger on the reason why. I just prefer the more straightforward Thunderbird for my email chores. In contrast to the multifunctional Evolution, Thunderbird (Figure 4-15) is a more mail-oriented program that is very straightforward to use, yet includes most of the most important email functions you've come to expect, such as junk mail filters. In fact, it is remarkably similar to Outlook Express in terms of appearance and handling. It also lacks the quirkiness that always seems to squirrel its way into Evolution in some form or another. The fact that Thunderbird is also available in both Mac and Windows versions means that you may already be familiar with it, or prepared to deal with it if you find it in use on another system you happen to be using.
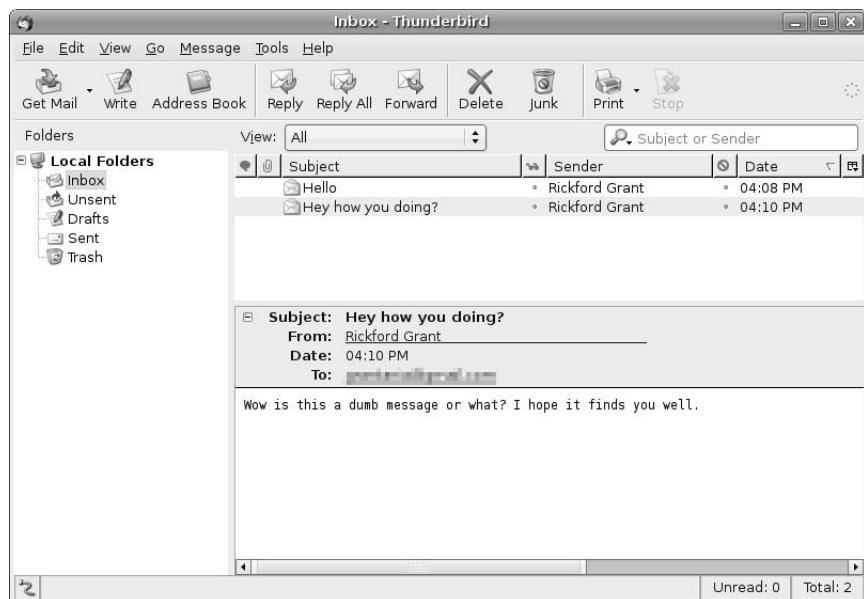
Figure 4-15: The Thunderbird email client

Thunderbird does not come bundled with Ubuntu, so if you would like to try it out, you will have to download it and install it yourself. Now that you have set up your machine to connect to the Internet, however, you can easily do this after going over the contents of Chapter 5 (okay, so I'm jumping the gun a bit again). Just do a Synaptic search for *thunderbird,* and then mark **mozilla-thunderbird** for installation. Once it is installed, you can then run it from the **Applications** menu by selecting **Internet ▸ Thunderbird Mail**.

As I mentioned, both Evolution and Thunderbird are equally capable and possess essentially the same features in terms of mail handling. The difference is primarily a look-and-feel matter. Why not try both Evolution and Thunderbird and see which you like better?

By the way, if you find that you prefer Thunderbird to Evolution, you can add a panel launcher for it so as to make things easier on yourself when you want to run the program. Just go to **Applications ▸ Internet ▸ Thunderbird Mail Client**, right-click that entry, and then in the popup menu that appears, select **Add this launcher to the panel**. You can then remove the Evolution launcher, if you are so inclined.

Other Internet Applications

What I've covered thus far in terms of Internet applications is just the tip of the iceberg (might as well use that worn-out phrase before there aren't any icebergs left, right?). Ubuntu also comes with a couple of other Internet applications that you might want to consider. These include the Instant Messenger client called Gaim (**Applications ▸ Internet ▸ Gaim Internet Messenger**), which allows you to do use any one of your MSN/Windows Instant Messenger, Yahoo! Messenger, AOL Instant Messenger (AIM), ICQ, Gadu-Gadu, Napster, GroupWise, IRC, or Jabber accounts . . . or all of them simultaneously. If you want to give Internet telephony a try, Ekiga Softphone (**Applications ▸ Internet ▸ Ekiga Softphone**) also comes bundled with your distribution, so you need not despair.
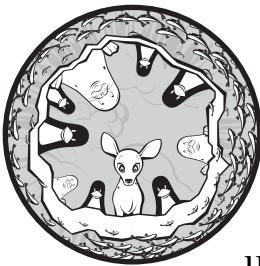
NOTE *Gaim Internet Messenger has recently undergone a name change. While the version that comes with Ubuntu 7.04 is still called Gaim, future versions will be known as Pidgin. You can go to the Pidgin project site (www.pidgin.im) to find out more.*

There are still more Internet applications that you might want to consider downloading and installing after you've completed Chapter 5, including Thunderbird, which I've already mentioned in this chapter. If this all sounds enticing, get those fingers of yours flipping—the mother lode awaits!

5

ROUNDING OUT THE BIRD

Downloading, Installing, and Updating Programs the Easy Way

One of the handiest things about Ubuntu is that it is equipped with a very simple-to-use application installation mechanism. The engine, using the term loosely, behind this is a mechanism called Advanced Package Tool (APT), which allows you to easily download, install, update, and remove software packaged in DEB archives, or *packages*.

APT is a rather foolproof way of installing programs; nothing will go missing, since it automatically downloads and installs any files that the main application you are installing requires to run. Tracking down such files, called *dependencies*, proves to be a significant headache for most Linux users. The painful quest of finding and then installing this file or that, as well as any dependencies that those files themselves might have, has led to the missing dependency problem being referred to as *dependency hell*. APT makes that pretty much a thing of the past.

The one thing about APT that some people, especially beginners, might consider a problem is that it is a command-driven application. This means that you control it via commands in a command Terminal. Fortunately, Ubuntu has two different graphical front ends for APT that allow you to bypass the command line and make everything about as easy as you could hope it would be.

Project 5A: Installing Applications via Synaptic

As I mentioned, Ubuntu comes with two graphical front ends for APT: the simpler, though arguably more attractive, GNOME App Install; and the main application handling workhorse, Synaptic. Because Synaptic is the more full-featured of the two applications, and because GNOME App Install also utilizes it for its settings chores, we'll start things off with Synaptic.

To get a look at Synaptic, go to the **System** menu, and select **Administration ▸ Synaptic Package Manager**. Because APT installs things in your system in areas that are write protected (more on that topic in Chapter 6), you will be prompted for your password in a separate window before Synaptic actually appears. The password is just your usual user password (users of other Linux distros take note), so that should be easy enough to remember. Type your password when prompted to do so, and click **Continue**. Synaptic, shown in Figure 5-1, will soon appear.

Figure 5-1: Synaptic Package Manager

Although the Synaptic interface may seem rather simple at first glance, it is a very powerful program. It not only allows you to search for, download, and install applications, but it also allows you to find and install the less glamorous, but equally important, libraries and support files that you might need

for use with other applications that come bundled with Ubuntu (such as MP3 support), or those you install by other means. You can also use Synaptic to find other things such as fonts, foreign language localization files, and less common utilities, and to update the software you have installed on your system when such updates are available.

Of course, if you don't know what you're looking for, you can browse through the packages available via Synaptic by clicking the **Sections** button at the bottom-left corner of the window, selecting a category of interest in the pane above that, and then clicking the name of a specific package in the top-right pane, after which a description of that package will appear in the pane below.

5A-1: Adding APT Repositories via Synaptic

It is important to know that the packages that APT (and thus both Synaptic and GNOME App Install) searches for, downloads, and installs are located in a set of specific online *repositories.* These repositories are basically online servers in which a great number of applications, support files, and more are stored for use with your particular system. All of the files that originally come bundled with your system, including the system (kernel) itself, and updates, when available, are stored there.

Only the main application and update repositories are activated by default in Ubuntu, but the real gold mine of applications is located in what are called the *universe* and *multiverse repositories.* These repositories contain software that is prepared and maintained by the Ubuntu community, but not officially supported by Ubuntu. In order to really make your Synaptic experience (and your GNOME App Install experience, for that matter) worthwhile, and to follow along with future projects in this book, you will, therefore, need to make sure that the universe and multiverse repositories are activated.

To do this, go to the Synaptic **Settings** menu, and select **Repositories**. When the Software Sources window appears, you will see all of the repositories that are currently activated (Figure 5-2). Check to see if the top four boxes are checked. If any of those boxes are not checked, well . . . check them. You can ignore the last box next to the words *Source code.* Once you've done all that, click the **Close** button.

A new window will then appear telling you that you will have to click the Reload button in order for your changes to take effect. Click the **Close** button in that window, and then click the **Reload** button in the main Synaptic window as you were just instructed to do.

Synaptic will then begin downloading package lists from each of the repositories you just added to its package database. A window like that in Figure 5-3 will show the progress of the download and will disappear when the process is complete. There will then be a brief period during which all of the panes in the main Synaptic window will be gray, but don't worry; this is only temporary.
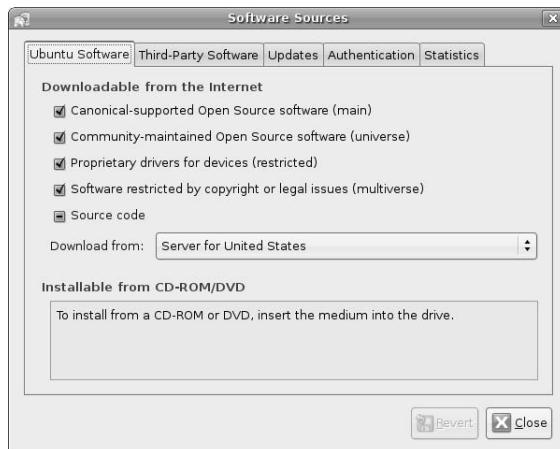
Figure 5-2: Synaptic's repository list

NOTE *Because the contents of the various repositories change quite often, it is advisable to reload the repository lists by clicking the Reload button every time you take Synaptic out for a spin.*

Figure 5-3: Download progress indicator in Synaptic

Adding New Repositories to Synaptic

You don't need to do it right now, you may never have to it in the future, and, in fact, I'm not sure I'd recommend you doing it at all anyway, that is unless you really know what you're doing and why you're doing it. Still, it is worth knowing that you can add new, outside repositories to Synaptic.

To add a repository, just follow these steps:

1. Go to the Synaptic **Settings** menu, and select **Repositories**.

2. In the Software Sources window that appears, click the **Third-Party Software** tab.

3. Once you're in the new tab, click the **Add** button.

4. Another window will appear. In the text box in that window, enter the APT line for the repository you want to add. The format for all repositories will basically follow that shown below:

```
deb http://wine.budgetdedicated.com/apt feisty main
```

5. Once you're done, click the **Add Source** button in that window. The window will disappear, and the new repository will appear checked in the Third-Party Software list.

The new repository, just like the universe and multiverse repositories, can be deactivated by simply unchecking the box next to its entry in that window.

5A-2: Installing Monkey Bubble

Let's start using Synaptic by installing a game called Monkey Bubble just to see how things work. What you will be doing is basically the standard method you would use to search for and install any package via Synaptic, so it is a process well worth remembering. Here are the steps:

1. In the main Synaptic window, click the **Search** button.
2. A Find window will appear. Be sure that **Description and Name** is selected in the Look In menu (to provide a more forgiving search), type `monkey` in the Search box, and then click the **Search** button.
3. After a short period of time, a list of results will appear in the upper-right pane. Look for **monkey-bubble**, and click it once to select it. A description of that file will then appear in the bottom-right pane.
4. Right-click that same entry, and select **Mark for Installation** in the popup menu that appears.
5. You are now ready to roll, so just click the **Apply** button in the main Synaptic window, after which yet another window (Summary) will appear, telling you what is going to be installed, what is going to be changed, and what is going to be left alone (Figure 5-4).

Figure 5-4: Synaptic lets you know what additional packages will be installed

6. Click **Apply** in that Summary window, after which Synaptic will begin the installation process, showing you its progress in a separate window (very much like the progress window you saw in Figure 5-3, in fact).

7. When the installation is complete, a new window like that in Figure 5-5 will appear to let you know. Click **Close** in that final window, and then wait until Synaptic snaps out of its temporary state of grayness, which signals the complete end of the installation process.
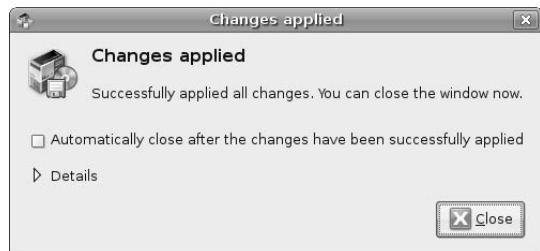
Figure 5-5: Synaptic lets you know when it's done doing its thing.

You can now run Monkey Bubble (Figure 5-6) by going to the **Applications** menu and selecting **Games ▶ Monkey Bubble**.

Figure 5-6: Monkey Bubble

Removing Applications via Synaptic

To remove applications installed from a DEB package via Synaptic, GNOME App Install, or any other means, search for the application you wish to remove either by clicking the **Search** button and typing the name of the

application you wish to remove, or by clicking the **Status** button in the bottom-left corner of the Synaptic window, clicking **Installed** in the pane above that, and then scrolling through the list of installed packages that then appears in the top-right pane. Once you find the application in that list, right-click on its name, and select **Mark for Complete Removal** to remove the application and any setting or configuration files for that application. When the window that tells you what is going to be axed appears, check through it carefully so as to make sure you aren't inadvertently uninstalling something you would mourn the loss of—like your whole desktop environment, for instance. Assuming everything looks okay, click the **Apply** button, after which the process will be the same as that for installing applications.

Upgrading Applications via Synaptic

As I mentioned, you can also use Synaptic to upgrade applications installed in your system. To do this, click the **Status** button in the bottom-right corner of the Synaptic window. In the pane above that button, then click the entry **Installed (Upgradeable)**, after which a list of the installed applications with available upgrades will appear in the upper-right pane. Right-click on any application listed there that you would like to upgrade, and select **Mark for Upgrade** in the popup menu that then appears. Repeat the right-click procedure for any other application you want to upgrade, then click the **Apply** button. The procedure from then on out will be the same as that for installing packages.

Project 5B: Installing Applications via GNOME App Install

While Synaptic is great for just about everything, when it comes to browsing to see what cool or handy applications are available for you, Ubuntu's other APT front end, GNOME App Install, is probably a more satisfying way of going about things, due to its graphical nature.

To get a feel for that application, make sure Synaptic is closed (you can't run Syanptic and GNOME App Install at the same time), and then open GNOME App Install by going to the **Applications** menu and selecting **Add/Remove**. The Add/Remove Applications window, shown in Figure 5-7, will then appear.

As you can see, items in GNOME App Install are categorized pretty much in the same manner as in the Ubuntu Applications menu itself. If you click any of the category icons in the left pane of the window, you will see a list of all the items available for that category in the top-right pane—installed items are checked, while those that are not installed are unchecked. Not surprisingly, you will find that the checked items within each category mirror those in the relevant submenu of the Applications menu.

Figure 5-7: GNOME App Install

5B-1: Selecting Applications for Installation

GNOME App Install, like Synaptic, can install multiple applications simultaneously, so to give it a whirl, we'll try installing a couple of applications that might be of interest to you. Before getting down to selecting the applications, however, be sure to select *All available applications* in the menu button next to the word *Show*. Doing this will enable you to view everything that is available to you in all the repositories you checked in Project 5A on page 68.

Now that all that there is to be seen can indeed be seen, let's start out by selecting the very cool and decidedly useful address book application called Rubrica (shown in Figure 5-8). To select Rubrica, just click the **Office** category in the left pane of the Add/Remove Applications window, scroll down the list of available applications in the top-right pane, and check the box next to the words *Rubrica Addressbook*. A small window will appear posing the question: *Install community maintained software?* Click **Install** to proceed, and then close that window.

Let's also select a handy audio application called SoundConverter (not to be confused with soundKonverter) that converts audio files from one format to another—MP3 to Ogg Vorbis, for example (you'll learn more about SoundConverter in Chapter 16). To select it for installation, click the **Sound & Video** category in the left pane of the window, scroll down in the top-right pane, and check the box next to the word *SoundConverter*.

NOTE   *Be careful not to uncheck the boxes next to any of the already checked applications listed in GNOME App Install. Doing so will result in their being removed once you go on to Project 5B-2 on page 75.*
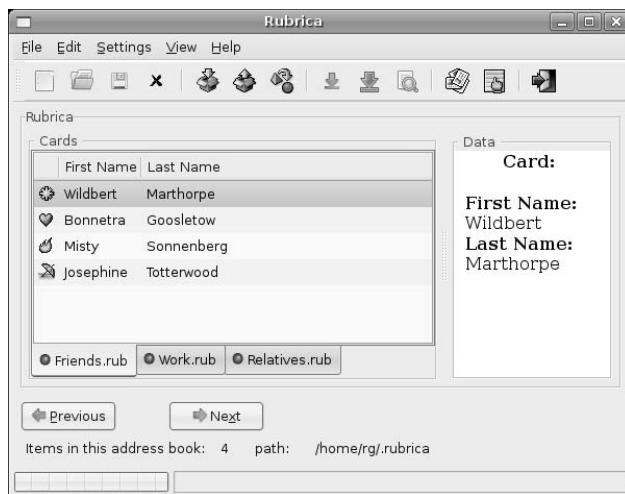
Figure 5-8: Rubrica—a very cool address book

5B-2: Downloading and Installing Selected Applications

Once you've made your selections, click the **OK** button at the bottom of the Add/Remove Applications window. A new window, like that in Figure 5-9, will appear showing what you are about to install. Click **Apply** in that window, after which you will be prompted for your password, just as you were for Synaptic (and for the same reasons), so type your user password, and click **OK**. GNOME App Install (APT, actually) will then begin the download and installation process, showing its progress via the same progress windows you saw when installing applications with Synaptic.

Figure 5-9: GNOME App Install tells you what it is about to install.

When the installation is done, a window like the one that you saw at the end of the Synaptic installation process (see Figure 5-5) will appear. Click **Close**; GNOME App Install will close itself down and you'll be ready to run your new applications. You can run the applications by going to the **Applications** menu and then looking in the submenu that matches the category in which each application was located within GNOME App Install. That would be **Office** for Rubrica, and **Sound & Video** for SoundConverter, which you'll learn more about in Chapter 16.

As you can see, installing applications via GNOME App Install is quite simple, and if you paid heed to my warning, you also know how easy it is to uninstall applications. Just check, and click Apply to install; uncheck, and click Apply to remove. Can't get much easier than that, you have to admit.

Performing System Upgrades via the System Update Panel Applet

At the beginning of this chapter I mentioned that there were two graphical front ends for APT, but I must now admit that I understated things a bit—there are actually a few more, though these are more specialized. The one covered here, for example, is strictly used for updating your system. Before updating individual packages via Synaptic, therefore, it is probably a good idea to save yourself some time by first checking for available updates via the System Update panel applet.

To perform the update, make sure that Synaptic and GNOME App Install are both closed (you just can't run two APT front ends at the same time, as I told you), click the **System Update** panel applet, and the Update Manager window (Figure 5-10) will appear.
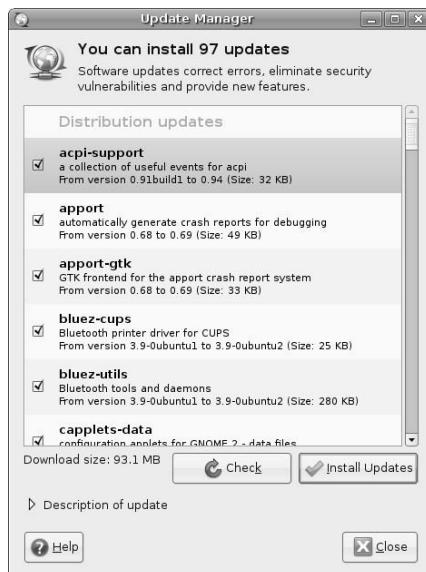
Figure 5-10: Upgrading packages en masse via the System Update panel applet

In that window, a list of all of the available updates will appear. You can scroll through that list and uncheck the box for any application you don't want to upgrade, but if you don't really know what you're doing, you had better just leave it as is. Either way, once you're ready, click the **Install Updates** button, after which you will be prompted for your password. Type your password, and then click **OK**. The updates will then be downloaded and installed, with the progress shown in the same types of windows you saw when using Synaptic and GNOME App Install. Once the installation is complete, you will be notified in a separate window. Click **Close** in that window, and then do the same in the Software Updates window. Depending on what you installed, another small window may appear telling you that you will have to restart your system in order for changes to take effect. If so, it is best to be obedient and reboot.

If an Update Ruins Your Day . . . or System

Now, I don't want to worry you too much, but one thing about massive system updates is that afterward, sometimes things just go all screwy, leaving you with all sorts of regrets and a mouthful of expletives just waiting to be uttered and, hopefully, deleted. This is a fact of life no matter what operating system you are using. It can happen in Windows, it can happen in Mac OS, and it can happen in Linux. Therefore, it is always a good idea to wait a bit before installing a seemingly major update and to check the user forums first to see if there are any disaster stories out there.

If you go ahead only to find that your system won't start up again, don't worry. All is not lost. In fact, nothing is lost . . . most likely. Just start up your machine again. If you have a dual boot system, you will arrive at the GRUB boot menu, just like always. If you're not a dual-booter, you'll have to get to this menu by pressing ESC as soon as you're prompted to do so just a few moments after starting the machine.

In the GRUB menu, you will see that you seem to have more than one Ubuntu system installed on your hard disk, which is true to some extent. Take advantage of this quasi-fact by traveling back in time to the core system (the Linux kernel) that you had in place before your ill-fated upgrade. Just use your arrow keys to select a kernel that ends in a number lower than the highest (2.6.20-11 rather than 2.6.20-13, for example), press ENTER, and *voilá*, you are back to the system you have come to know and love.

Hopefully, with all this disaster and recovery talk, I haven't scared you away from the idea of updating your system. After all, chances are great that nothing like this will ever happen to you. Still, it is good to know that if things do go awry, there is no need for panic. You've got backup.

6

A TIDY NEST

File and Disk Handling in Ubuntu

No matter which operating system you are using, you have to deal with files. Some people are very organized, placing every file in a logically named folder as soon as that file is saved for the very first time. Then there are people like me, who save everything to the desktop until it is so full of junk that they can no longer make out the wallpaper, and only at that point do they start organizing in earnest (if placing all of those files in a single folder called March17Cleanup can be called organizing). Ah.

Of course, files not only get stored on your hard disk, but they are also copied to and from CDs, DVDs, external hard disks, flash drives, and other storage media and devices. They are also often saved in archives, which are then compressed to reduce their spatial footprint, making them easier to send via email or to fit onto spatially challenged removable storage media.

With that intro, you may have already guessed that in this chapter I will be dealing with file handling in Ubuntu, particularly in relation to the Nautilus File Manager, which is at the heart of GNOME's file-handling capabilities.

Nautilus: Your File Manager

As I mentioned, the program that creates the file viewing and organizing interface in your system is called Nautilus, and it comes as part of the GNOME desktop environment. You may not have thought of an operating system's file manager as a program before, but in fact, that is what it is. (The Windows file manager is called Windows Explorer.) To have a look at Nautilus, just go to the **Places** menu, and select **Home Folder**.

When Nautilus opens up to your home folder, you should find nothing the first time out other than a folder titled Desktop (shown in Figure 6-1), which, if double-clicked, will show everything you have stored there (a lot in my case; most likely nothing in yours). There is also another folder, called Examples, which is the same as the one that appears when running Ubuntu off the live CD. It contains sample files that give you an idea of what Ubuntu has in store for you, along with that Nelson Mandela video I mentioned in Chapter 3.

You can create additional folders and, of course, files to your heart's content, so this rather spartan state of affairs is sure to change once you get down to really using your system. In fact, you will be creating a number of folders in Chapter 7, which will make everything look a bit more lived in.
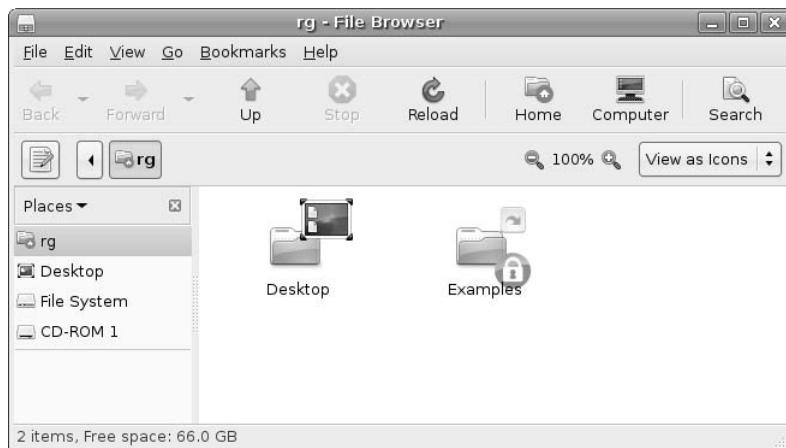
Figure 6-1: The contents of your home folder as viewed in a Nautilus window

The Side Pane

Nautilus has a lot of interesting features that deserve mention, and the most obvious of these is the side pane, which appears at the left side of the window. The side pane allows you to view a variety of information via selectable views. You can make your choices by clicking the drop-down menu at the top of the side pane (Figure 6-2).
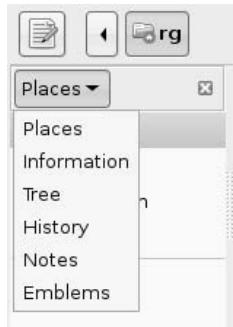
*Figure 6-2: Selecting views
for the Nautilus side pane*

The default view in Ubuntu is Places, which is a sort of quick navigation tool. In Places you will find icons representing various data-storage locations available to your system, such as your home folder, desktop, full filesystem, any network shares you are connected to (more on that in a moment), and any removal storage media or devices you have in or connected to your system. Double-clicking any of these icons will show the contents of that location in the right panel of the Nautilus window.

There are, of course, other views, such as Tree, which provides you with an expandable hierarchal view of your filesystem, and History, which shows you where in your filesystem you have been most recently, much in the way the history function works in a web browser. There are still other views for you to choose from, a couple of which you will work with in Chapter 7.

Now You See It; Now You Don't

The side pane is a rather handy feature, but there may well be times when you would prefer to have more space to view the contents of your window and thus want to get rid of the pane temporarily. You can do this quite easily by going to the **View** menu of a Nautilus window and then deselecting **Side Pane**. The check mark next to that entry will then disappear, as will the side pane. To get it back, just return to the **View** menu, and select **Side Pane** again. The check mark will then reappear, as will the side pane itself.

There is another way to hide the side pane that many people seem to stumble upon accidentally, usually resulting in a bit of unnecessary panic. If you look at the gray border at the right side of the side pane, you will notice that there is a small ribbed section in the center (see Figure 6-3). Clicking that ribbed section acts as a toggle to hide or show the side pane. The panic most users suffer is a result of the fact that when the side pane is hidden in this way, the Side Pane entry in the View menu is still checked, indicating the side pane is still in view, though your eyes tell you differently.

So there it is—the side pane's little secret. No, it's not a particularly interesting secret, but one that should not only provide you with an added layer of convenience, but also some peace of mind.
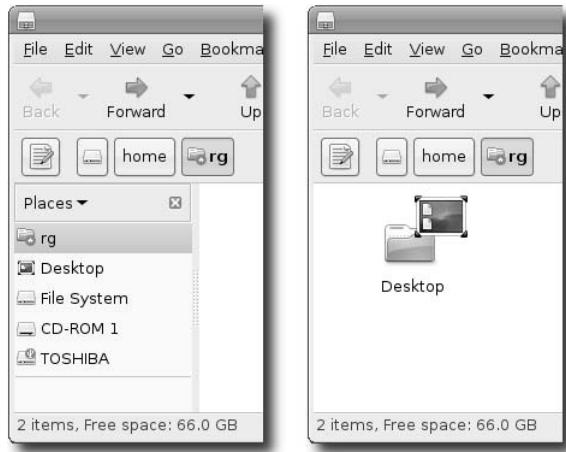
Figure 6-3: Showing and hiding the Nautilus side pane

File Handling in Nautilus

Since Nautilus is primarily a file manager, it only makes sense to get down to the business of using it at that level. Of course, most folks who use computers today are already familiar with the basics of drag-and-drop and a few other means of creating folders and copying, cutting, and moving files. But for those who are unfamiliar with one way or another of performing these essential procedures, I thought it best to spell it all out. If you find this all a bit redundant, please bear with me for the good of the masses.

Creating, Naming, and Renaming Folders

Creating a folder is a simple enough task, and there are two ways of going about it. The easiest (in my opinion) is to right-click any empty space within a Nautilus window, and select **Create Folder** in the popup menu that appears. If you prefer using menus to right-clicking empty space, you can instead start things rolling by going to the Nautilus **File** menu, and selecting **Create Folder**.

Regardless of where you made your Create Folder selection, a new folder with the name *untitled folder* will appear in the Nautilus window. The name box of the folder will be highlighted and surrounded with a black box, which means that you can immediately give that folder a name by simply typing one—nothing to click or do other than that. Press ENTER or click any open space in the Nautilus window to complete the job.

If you later decide that the name you gave your folder needs some tweaking or even a complete revision, you can rename it by right-clicking it and selecting **Rename** in the popup menu. Alternatively, you can click the folder once to highlight it, and then select **Edit ▸ Rename**. After that, you can type the new name for the folder and then press ENTER or click any open space in the Nautilus window to seal the deal.

Moving Files and Folders

Perhaps the easiest of all file manipulations you can perform in Nautilus is moving a file by means of drag-and-drop. I am pretty sure that anybody who has wielded a mouse is familiar with that particular move. There is another way of moving files and folders, however: cut and paste.

The easiest way of doing this is to right-click the file (or folder) you want to move, and then select **Cut** from the popup menu (**Edit ▸ Cut** will also do the trick). At this point, it will seem as if nothing has happened, as the file will still be there, so don't worry. After that, right-click any open space in the folder to which you wish to move the file, and then select **Paste** in the popup menu. The file will then disappear from its original location and appear in its new one.

Can you use key combinations to do this, you ask? Sure. Simply follow the directions I just gave, but use CTRL-X to cut and CTRL-V to paste.

Copying Files and Folders

Based on the instructions I just gave, you can pretty well imagine the methods for copying files and folders, as they are essentially a variation on the same theme. Just right-click the file you want to copy, select **Copy** from the popup menu, right-click any open space within the target location, and then select **Paste**. Keystroke-wise, that would be CTRL-C to copy and, as before, CTRL-V to paste.

It is also possible to copy folders and files via the wonders of drag-and-drop, though this involves more hands than required for a simple drag-and-drop move; fortunately, the two you have will do nicely. Just press and hold the CTRL key while you drag the file or folder you want to copy to the target location. Be sure to release the mouse button and then the CTRL key (releasing in the opposite order will not work), and you will find a copy of the file in its new location.

Navigating in Nautilus

Navigating through your various folders and subfolders in Nautilus is quite straightforward. In fact, all is conceptually pretty much the same as what you are accustomed to in Windows and Mac OS. You can simply move into and out of folders through a combination of double-clicking folders and clicking the Back, Forward, and Up buttons.

Tabbed Browsing in Nautilus

In addition to the hierarchal view option provided in the side pane, there is another handy feature that can make your navigation chores even easier: Nautilus's tabbed browsing feature. These tabs (they look like buttons, actually) appear in the navigation bar for each folder you opened on the way to the one you are currently viewing.

Say, for example, that you have a folder called gooseberries inside a folder called Dalarna inside a folder called SwedeStuff inside a folder called NordicStuff, which itself is in your home folder. As you click your way to that gooseberries folder, starting with a double-click on the NordicStuff folder, Nautilus will display a tab for that folder . . . and any folder opened before it. Take a look at Figure 6-4 to see what I mean.

Figure 6-4: Tabbed navigation in Nautilus

As you can see, there is a tab button for each of the folders within the path from your home folder (mine, in this case) to your target: gooseberries. So what, right? Well, say you want to go back to the NordicStuff folder to open a file in which there's some text that you want to copy and then paste into a doc within the gooseberries folder. Sounds like a minor pain, right?

Well, rather than goof around with the Back button, you can instead simply click the NordicStuff tab button, and the contents of that folder will be there before you. Need to go back to gooseberries? Just click the gooseberries tab button. Back to SwedeStuff, you say? Just click the SwedeStuff tab button. All quite *fantastisk*!

Spelling It Out—Typing File Paths in Nautilus

If you prefer typing to clicking, you will be happy to know that you can navigate to a folder by typing its path. Just go to the **Go** menu, and select **Location**, or keyboard shortcut lovers can just press CTRL-L. A search box will then appear in the location bar (Figure 6-5), in which you can type the path to your target folder, and then press ENTER. In the case of my berried example, that path would be: /home/rg/NordicStuff/SwedeStuff/Dalarna/gooseberries.

Figure 6-5: Typing the path to your target folder

Bookmarks Within Nautilus

With all this clicking away to deeply buried subfolders, it is worth mentioning another very handy feature of Nautilus: bookmarks. Yes, Nautilus lets you bookmark folders to which you have navigated. While you are no doubt familiar with creating bookmarks for web pages that you frequent, you may be wondering why on earth you would want to create bookmarks within your filesystem.

Well, imagine that you have a folder that you need to use often, but it is even more buried away than my gooseberries folder in the previous section. Getting there would take an excessive number of mouse clicks, and all that clicking is bound to eventually give you a bad case of carpal tunnel syndrome. While that is great for your HMO, it is most decidedly not good for you. Instead of maiming yourself, you could click your way to that folder once, and then, in the **Bookmarks** menu of the Nautilus window, select **Add Bookmark**. After that, whenever you want to get back to that buried folder, you can just click the **Bookmarks** menu, and the folder will be right there waiting for you in the drop-down list.

Another handy thing about Nautilus bookmarks is that they also appear in Save As dialog boxes, such as when you save an OpenOffice.org document or download a file via Firefox. In any Save As dialog box, just click the **Save in folder** button, and you will find your bookmarks.
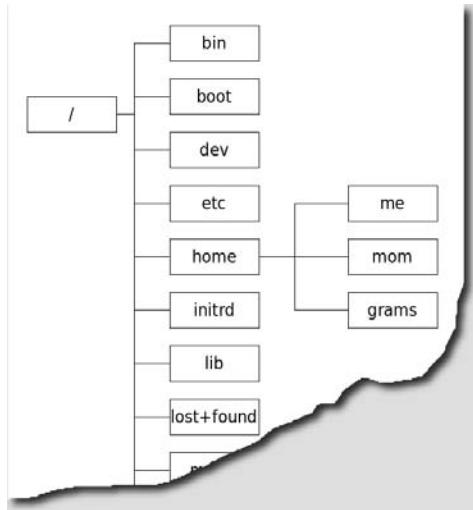
Understanding the Linux Filesystem

With all that path typing, navigating, and bookmarking you've just been learning, you should pretty much be able to figure out how to get from here to there in your home folder. Still, it is probably a good idea to know where your home folder actually is in the scheme of things, just in case you manage to get yourself really lost someday. To understand this, you should have a basic understanding of the Linux filesystem which, as you will find, is a bit different from what you were accustomed to in your previous operating system.

Unlike Windows, the Linux filesystem all stems from a single point called *root* and is represented by a solitary forward slash (/). Your own user account folder is located within a subfolder of root called *home*. This is represented as /home, or to put it into words, the home folder within root. If your user account were to have the same name as mine, rg, the path to that folder would be /home/rg, or the rg folder within home within root. Whatever the name of your user account or the names of any other accounts you have on your machine happen to be, just remember that when you are lost, your folder, and those for all the other folks with user accounts on your machine, are located within the home folder. If you're a more graphically oriented person, the map in Figure 6-6 should help you out.

What's in All Those Other Folders?

Needless to say, there is more to the Linux filesystem than the root, home, and user account folders. There are several other directories at the same level as home, though for the most part, you really shouldn't be directly mucking around with them unless you know what you're doing. Fortunately, most of these folders are write protected, so you should be fairly well protected from yourself. Still, it is natural to be curious about what those other folders are there for, since . . . well, they're there. So, to satisfy your curiosity a bit, I'll do a little explaining.

Figure 6-6: The location of your users' folders
within the Linux filesystem

In Ubuntu there are four folders that contain most of the applications on your system. The essential elements of your system are located in /sbin. Other elements that need to be there, though they may not be used, such as commands and the like, are located in /bin. Most of the applications that you actually think of as applications and use in a hands-on way are located in /usr. Finally, there are some add-on applications, such as RealPlayer, that install themselves into /opt.

Three other top-level folders that might be of interest to you are /etc, /lib, and /media. The first of these, /etc, is the location of all the configuration files on your system. The second, /lib, is the home of all of the libraries that are required by your system or applications installed on it. These libraries are the Linux equivalent of a Windows .dll file. Finally, /media is where the contents of your various attached external media (such as USB drives and CDs) appear when present.

There are other top-level folders, but their purposes are a bit less straightforward for the average Linux newbie and, to be honest, probably not all that interesting to you at this stage of the game. Anyway, as I mentioned before, there is no reason for you to be mucking around with any of these folders for the time being, as the items located within them can be accessed in different, and much safer, ways.

Using Nautilus as a Network Browser

Another handy Nautilus feature is its ability to function as a network browser. You can, for example, see what networks and shares are available to you on your home or office network by going to the **Places** menu in the GNOME Panel and selecting **Network Servers**. You can do the same from within a Nautilus window by going to its **Go** menu and selecting **Network**. Icons for any

networks or computers on that network would then appear in the Nautilus window, as you can see in the case of my own home setup in Figure 6-7. From that window, you can then double-click your way to a share that you have permission to access, such as the Shared Documents folder of a Windows machine or the public folders of a Mac.
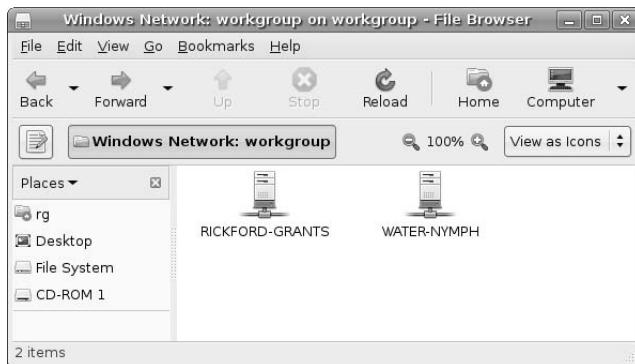

Figure 6-7: Network browsing in Nautilus

By the way, in case you are wondering what a share is, I'll clear that up for you. Basically, a *network share* is a location on a computer, such as a folder, where other users on a network can access and save files. The Shared Documents folder on a Windows system is a good example. Other users on a network can copy files from and (usually) write files to the Shared Documents folder, whereas they cannot access any other part of the filesystem on that host machine.

If the share you are trying to open requires a username and password, you will be asked for those in a new window (Figure 6-8). Note that in this case, the username and password you need to enter are those for the machine to which you are trying to connect—not the ones you're using in Ubuntu (unless the usernames and/or passwords happen to be the same, of course). You can then copy files to and from that share as if it were a folder on your own hard disk.

Figure 6-8: Entering network usernames and passwords for the machine to which you're trying to connect

In some cases, especially when trying to access shares on a Mac running OS X, the double-click method will not work. You will not be able to access any share on a particular machine, even though an icon for that machine appears in the Nautilus window. In such cases, double-click the icon for the machine you are tying to access, and then once open (to an empty window), go to the **Go** menu, and select **Location**. A text box will appear in the location bar (just like in Figure 6-5) with the location of the machine you are trying to access already listed.

To that location, add a forward slash (/) followed by the username used on the target machine. For example, if the target machine is called cowboycats, and the username is mewtoyou, the location would be `smb://cowboycats/mewtoyou`. You can also narrow things down to a particular folder on the machine, as long as you know the path to that folder, by adding to the path you've already typed, `smb://cowboycats/mewtoyou/Documents`, for example.

NOTE   *In case you are wondering, the* `smb` *at the head of that path stands for server message block, but to make things easier (and perhaps more useful in terms of your memory) you can just think of it as being short for Samba, which is software used by Unix-based systems (such as Linux and Mac OS X) in order to interact with Windows networks.*

Once you've made your way to the folder you wish to browse on the networked machine you've connected to, it might be a good idea to use the bookmark function in Nautilus to bookmark that open share window. You can then easily access that share in the future by choosing the share's name in the Nautilus Bookmarks menu. Pretty cool, if I do say so myself.

Using Nautilus as an FTP Client

Nautilus not only allows you to browse and mount shares on local networks, but it can also act as an FTP client, say for instance when you want to change the files for your website on a remote server. To do this, go to the Connect to Server window (**File ▸ Connect to Server**), and then select **FTP (with login)** for Service Type. Then type the information provided by your website host, and click the **Connect** button. An FTP Network Share icon will then appear on the desktop and in the side pane. Double-click that icon, type your password (for that account—not the one for your Ubuntu system, unless it happens to be the same) when prompted to do so, and then you will be able to view and add to the files you have there.

File and Folder Permissions Within Nautilus

As you make your way in the world of Ubuntu, you will find that occasionally you'll come across files or folders which are in some way locked, either in terms of your being able to read them, being able to alter them, or both. These readability and alterability states of being are referred to as *permissions*.

Now you may be (legitimately) wondering what the point of this permissions business is, so in order to help you understand, I will give you some examples of how it can be useful. Let's say that you have some files that

you don't want your spouse or kids to see—some bad poetry or a Christmas shopping list, for example. By denying read permission to those files, or to a folder containing those files, no one would be able to sneak a peek unless they were savvy at changing permissions and had the permissions necessary to change permissions for those files or folders.

As another example, imagine you have a file that you have worked many hours on and have finally completed. To alleviate fears that you might accidentally trash it in some way, you could deny write permissions. By doing this, you wouldn't be able to do anything to that file without changing its permissions. In fact, if you tried to move that file to the Trash, or even to the desktop or another folder, you would be greeted with a no-can-do message, such as the one shown in Figure 6-9.
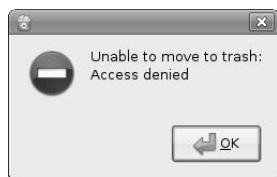

Figure 6-9: Nautilus tells you when permissions restrict your freedom of movement.

Changing File and Folder Permissions in Nautilus

To change file or folder permissions in Nautilus, just right-click the file or folder in question, and then select **Properties** from the popup menu. Once the Properties window opens, click the **Permissions** tab, and you will see who the owner of the file or folder is and what you are allowed or not allowed to do with it. As you can see in Figure 6-10, the options for files and folders are slightly different.

You might find this permissions business a bit confusing, but it is really quite simple to understand. Permissions can be granted or denied to the *owner* of the file or folder (you), to a specified *group*, or to *others* (everybody else). These permissions are:

Read   Permission to view the contents of a file or folder

Write   Permission to alter the contents of a file or folder

Execute Permission to run a program or script

In general, you needn't worry all that much about setting permissions for your own files, as you are really the only one who has access to your user account. One of the main exceptions to this is when you transfer files from CD to your hard disk. In this case, the files will be write protected, meaning that you cannot alter the files until you change the permissions for them. You can change the permissions of such files in order to allow yourself to alter them by clicking the checkbox in the Owner row next to the word *Write*. Once you are done, click the **Close** button, and you'll be on your way.
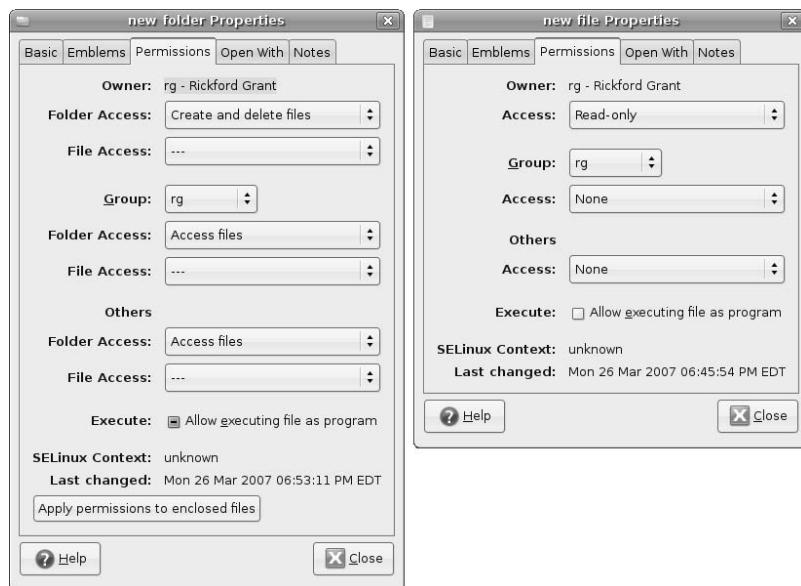
Figure 6-10: Changing permissions in a Nautilus Properties window

Keeping Your Home Folder Private

Another exception to my you-don't-need-to-worry-about-permissions claim, and a potentially important one at that, is the state of permissions for your home folder, particularly when other people have user accounts on your machine. In Ubuntu, when someone logs in to their own account on your computer, they can click their way to your user folder and view its contents.

To remedy this situation—and thus protect the sanctity of your home folder, the privacy of its contents, and the peace of mind of its owner (you)—you can change the permissions of your home folder. Here's what you need to do:

1. Open up a Nautilus window, and then double-click **File System** in the side pane. The contents of your entire hard disk will then appear in the right pane of the Nautilus window.

2. Look for and then double-click the folder named **home**. When the contents of the home folder you just clicked appear in the right pane, there should only be one folder there—your own folder, which will have the same name as your own username. For example, mine, as I mentioned earlier, is named rg.

3. Right-click your folder, and then select **Properties** in the popup menu.

4. In the *username* Properties window (mine says *rg Properties*), click the **Permissions** tab.

5. In the Permissions tab, go down to the Group and Others sections, and select **None** in the drop-down menus next to the words *Folder Access*. Be sure to do this in both the Groups and Others sections. When you're done, your window should look like mine in Figure 6-11. If so, click **Close**.
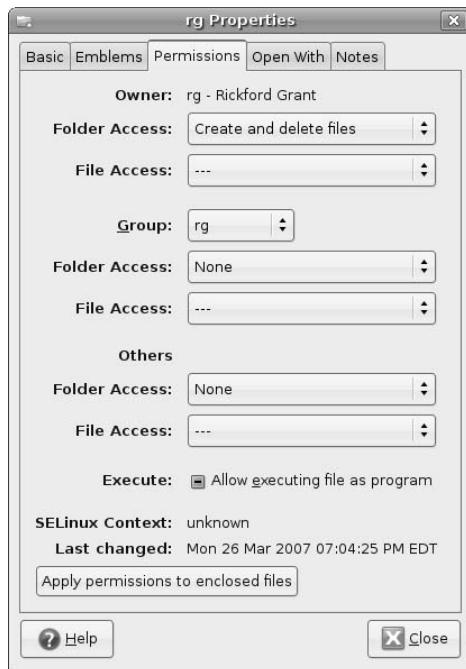
*Figure 6-11: Changing the permissions of a
home folder for privacy*

Reading Data CDs and DVDs

Dealing with data CDs and DVDs in Ubuntu is quite simple, as everything is
automatic. To read a CD or DVD with data on it, rather than music or video,
place the disk in your drive, and a CD or DVD icon (they look the same) will
automatically appear on the desktop. A Nautilus window displaying the disk's
contents will also appear, after which you can copy files from the CD or
DVD to your hard disk using standard drag-and-drop procedures.

When you want to remove the CD or DVD, just close its Nautilus window,
right-click the desktop icon for that disk, and in the popup menu, select **Eject**.
The disk will then be ejected automatically.

Burning Data CDs and DVDs

Burning data CDs and DVDs in GNOME is extremely easy. All you have to
do is place a blank CD-R (CD-Recordable) or DVD (DVD-RW, DVD-R, and
DVD-RW are all supported by Ubuntu) in your drive, making sure to select a
media format supported by your drive, and a window will appear asking you
what you want to do with the disk. Click the **Burn Data CD** button, or in the
case of a blank DVD, the **Make DVD** button. Nautilus's CD/DVD Creator
window will then appear, which as you no doubt notice looks pretty similar to
other Nautilus windows, save for the brown band below the location bar and
the Write to Disc button (Figure 6-12).
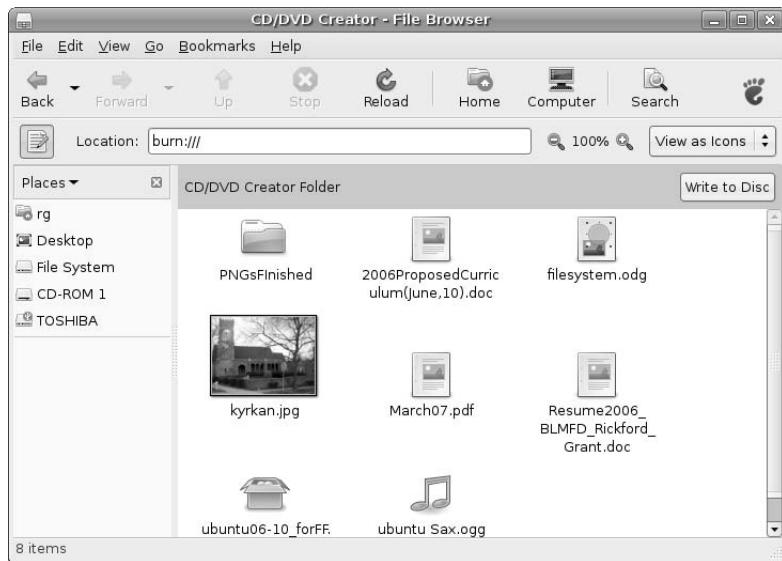
Figure 6-12: A Nautilus CD/DVD Creator window with files ready to be burned to disk

Once the CD/DVD Creator window is open, copying the files you want to burn to disk is pretty much a simple drag-and-drop maneuver. Just open a new Nautilus window, and drag the files you want to burn to disk from that window over to the CD/DVD Creator window. If you prefer to do things in a decidedly Windows-esque fashion, you can select the files you want to transfer to disk by clicking each file once, holding down the CTRL key while doing so, for multiple selections. If you want to select multiple consecutive files, you can click the first file in the group, press and hold SHIFT, and then click the last file in the group, automatically selecting all the files in between. Once you've made your selections, release the CTRL or SHIFT key, right-click any of the highlighted files, and select **Copy** in the popup menu. After that, go back to the CD/DVD Creator window, right-click any open space, and then select **Paste** in the popup menu.

It is probably worth mentioning that the files you copy to the CD/DVD Creator window are not actually copied. Instead, what you see in the Creator window are essentially aliases pointing to the original files in their original locations. Thus, if you move one of the files from its original location before burning the contents of the CD/DVD Creator window to disk, the file will automatically vanish from the window, and it will not be burned to disk when you finally get around to that step. This is not problematic, but it is something worth being aware of.

Once you have copied all of the files you want to burn to disk, click the **Write to Disc** button, after which a window (shown in Figure 6-13) will appear, telling you, among other things, how many megabytes of files you have selected to write to disk. In this window, you can give your disk a title and adjust the speed at which your disk will be burned (slower speed = fewer chances for errors), though you can just as well accept the defaults.
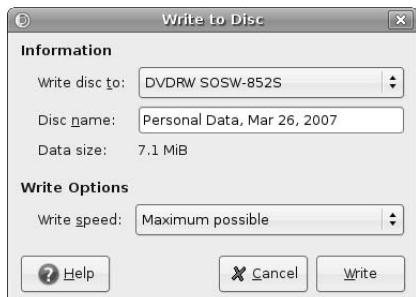
*Figure 6-13: Setting options before burning
a CD or DVD*

Once you are ready to burn the disk, just click the **Write** button, and the CD/DVD Creator will do its work. In some cases a warning window will appear telling you that some of your files may not be suitably named for Windows compatibility. Just click **OK** if you come face to face with such a window, unless you do intend to transfer the files to a Windows system, in which case, you had better go back and rename the files according to Windows naming conventions before you get down to the actual burning. In particular, avoid special characters and diacritics (such as umlauts and accents), and do not use the following characters, which are reserved for Windows system functions: / : ? * '' < > |.

Once the burning gets under way, its progress will be shown in a new window, and then when the job is done, you will be asked what you would like to do next. Assuming you are done with your disk burning for the day, click **Eject**, and then click **Close**.

In case you are wondering, the disks you burn in Linux *will* be readable in other operating systems.

Dealing with CD-RW Disks

CD-RW disks are pretty much like CD-R disks except that they can be erased and then written to again. They are also quite a bit more expensive than CD-R disks and, generally, cannot handle faster burning speeds.

Using CD-RW disks is much like working with CD-R disks. If the disk is blank, there is no difference in the process at all, which makes things quite simple. And, if the CD-RW disk already has data on it that you wish to replace with something else, the process is only slightly different.

One of these differences is that Nautilus will treat your CD-RW disk as a regular data disk rather than a blank one. This means that when you pop your disk into the drive, a regular Nautilus window will automatically open, rather than a CD/DVD Creator window.

To write to the disk, you will need to manually switch from the Nautilus window to a CD Creator window, which is easily done by selecting **CD/DVD Creator** in the **Go** menu of the Nautilus window opened for that disk. Once

you've done this, the window will become a CD/DVD Creator window. Now drag the files you want to burn to CD to that window, and, once you are ready to burn, click the **Write to Disc** button.

As is the case with regular CD-R disks or DVDs, a Write to Disc window will appear. When you click the **Write** button in that window with a used CD-RW disk in the drive, however, a slight difference occurs. At this point, a new window like that in Figure 6-14 will appear telling you that the disk seems to have files already written on it. Click the **Erase Disc** button in that window, and the CD/DVD Creator will erase the files already on the CD-RW and replace them with the new ones that you dragged to the CD/DVD Creator window. Not bad at all, eh?

Figure 6-14: Erasing a CD-RW disk with Nautilus

Burning ISO Images to Disk

When you download Fedora Core or other Linux distributions from the Internet, you usually download them in the form of one or more disk images, which are commonly referred to as ISOs because such files end in the .iso extension. An *ISO* is an image of a CD's file contents, which means that it is the CD minus the media itself. To put it another way, if CDs had souls, the ISO would be the soul of a CD; take away the CD's metal and plastic, and the remaining data would be an ISO.

As it is impossible to download a physical CD over the Internet, the bodiless ISOs are the next best thing. For example, to get a working copy of Ubuntu from the Web, you need to download an ISO, which you then burn onto a blank CD in order to give the images their bodies back, so to speak. In the process you create the working installation disk that you need to install Ubuntu.

Fortunately, burning an ISO to disk is a pretty simple chore. Just open a Nautilus window, and locate the icon for the ISO file you want to burn to disk. Right-click the ISO file, and in the popup menu that appears, select **Write to Disc**. Once you do this, the Write to Disc window will appear; just click the **Write** button, and you'll be on your way.

Duplicating Data CDs

Duplicating CDs and DVDs is also easily accomplished in Ubuntu. Just right-click the desktop icon for the disk you want to copy, and then select **Copy Disc** in the popup menu. The Write to Disc window will then appear. Click the **Write** button in that window, and the Nautilus CD/DVD Creator will begin copying the contents of the disk as a disk image to your hard disk. This may

take a bit of time, so don't worry. When it is done, you will be prompted to insert a blank disk into your drive. Once you've done that, the Creator will begin copying the disk image to the blank disk.

If you want to know how to duplicate audio CDs, hang on until you get to Chapter 15.

Burning Multisession CDs

If you are coming from the Windows environment, you are no doubt familiar with multisession CDs. These are CDs on which data is added one session at a time. For example, you burn a few files to disk today, add a few more to the disk tomorrow and a few more files the day after that. Each time you burn additional files to the same disk, you are adding a session, which explains the name *multisession*. If that explanation seems a bit obtuse, you can basically think of them working like floppy disks (albeit with considerably more storage capacity). While Nautilus can read and display the contents of such multi-session disks, it cannot (yet) write multisession disks. To do that, you will need a different disk-burning application. I recommend GnomeBaker.

You can download and install GnomeBaker via Synaptic (search for *gnomebaker*). Once installed, you can run it from the **Applications** menu by selecting **Sound & Video ▸ CD/DVD Writer GnomeBaker**.

To burn the first session to CD, click the **Data CD** button in the bottom half of the GnomeBaker window. After that, drag the files that you want to burn during the session to the bottom-right pane of the GnomeBaker window, under the Data CD tab (Figure 6-15). You can drag files from the file browser in the top two panes of the GnomeBaker window, from a Nautilus window, or from the desktop.

When you have all the files in place that you want to burn, click the **Burn** button at the bottom-right corner of the screen. A new window, Create Data CD, will appear. In that window, select **tao** in the drop-down menu next to the word *Mode*. Once you've done that, insert a blank disk in the drive, and click **Start**. The burning process will begin. GnomeBaker will eject the disk when the burn is complete.

Burning Subsequent Sessions

To add a new session to a multisession disk, make sure that the bottom-right pane of the GnomeBaker window is empty (click the **Clear** button if it isn't), and then insert the multisession disk into your drive. After your drive has finished its spinning and you have pushed the Navigator window for that disk out of the way, go to the GnomeBaker **Project** menu, and select **Import Session**. A small window will appear asking you to select the drive where the multisession disk is located. You most likely only have one drive, so the correct device should already be displayed. If so, click **OK**.

The files burned in previous sessions will now appear in the bottom pane. To add additional files to burn to the disk (thus creating a new session), just follow the procedures you used for burning the initial session.
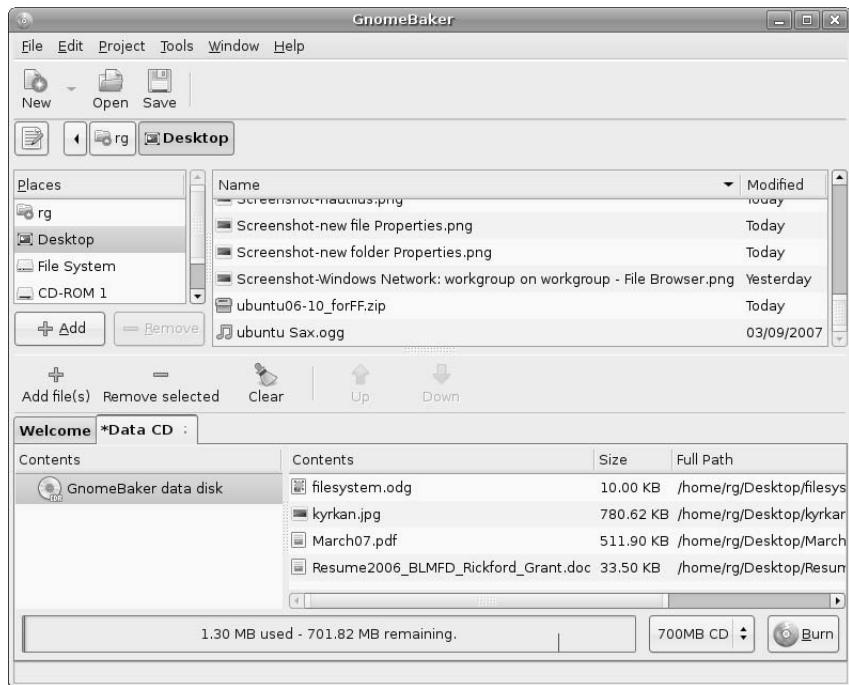
Figure 6-15: Using GnomeBaker to create multisession disks

USB Storage Devices

No chapter dealing with file handling and storage would be complete without touching upon the topic of USB storage devices, so that is exactly where I will now turn. Unless you have been under a digital rock for the past few years, you are no doubt well familiar with USB devices. Your printer is very likely a USB device, as is your scanner. And although your digital camera is not a USB device in the traditional sense, chances are that every time you connect it to your computer in order to transfer photos, you are doing so via a USB connector.

Among the most popular USB devices out there are those for file storage. These include external hard disks, flash memory card readers, and the tiny, finger-sized devices known as *flash drives* (Figure 6-16). Flash drives are especially popular today, and deservedly so: they are quite inexpensive; extremely handy when you need to transfer fairly large, but not gigantic, amounts of data from computer to computer (from work to home, for example); and pretty safe in terms of cross-platform (including Linux) compatibility.

Figure 6-16: USB storage devices

Putting USB Storage Devices to Work

Let's start with some good news here by pointing out that working with USB storage devices is really easy. Just plug the device into one of the USB ports on your computer. The LED on the device will do a bit of blinking as the system reads what's on it, and after that, a disk icon for that device will appear on your desktop. A few moments later, a Nautilus window will open, revealing the contents of the device. You can then copy files to and from the device using the drag-and-drop or copy-and-paste procedures I mentioned earlier in this chapter.

Once you are done and wish to remove the device, right-click its desktop icon, and select **Unmount** in the popup menu. If there is any data that needs to be written to the device, the system will start writing and tell you so in a message at the bottom-right corner of the screen (shown at left in Figure 6-17). Another message will appear in the same location when the writing process is complete, telling you that it is safe to remove the drive from the USB port (Figure 6-17, right).


Figure 6-17: Ubuntu lets you know what is happening with your USB drives

Project 6: Creating and Extracting Compressed Files

Since I have been talking about file storage, it seems only fitting to wrap things up in this chapter by teaching you how to create and extract compressed files. In the Windows world, these are generally referred to as Zip files, while in the Linux world, *tarball* is the operative name. The Linux name, in case you're wondering, comes from the application that is used to create the archive for such files, Tar.

Anyway, to get some of the hands-on stuff down, we'll be creating a Windows/Linux/Mac–friendly Zip file, and then extracting it. We can get down to business by opening a Nautilus window and creating a couple of dummy files to work with. You can do this by going to the Nautilus **File** menu and selecting **Create Document ▸ Empty File**. A new file will appear in the Nautilus window, with its name highlighted. You can now just type a name for the file, such as the one I'm using: *dogwood.* Now repeat the process to create a second file. I'll be calling that one *violet.* Use something equally evocative for yours.

Now that we have two files to work with, let's start creating the compressed archive by following these steps:

1. Select the two files either by clicking your mouse to the side of the files, and then dragging the cursor (with the mouse button still pressed) over both files until they are highlighted, or by holding down the CTRL key and clicking on each file individually.

2. Once both files are highlighted, right-click either one, and select **Create Archive** from the popup menu.

3. In the Create Archive window that then appears, type `blossoms` in the Archive text box, and then select **.zip** from the drop-down menu button to the right of that. Once everything looks like what I've set up in Figure 6-18, click the **Create** button, after which a compressed archive of your two files (blossoms.zip) will appear in your home folder.
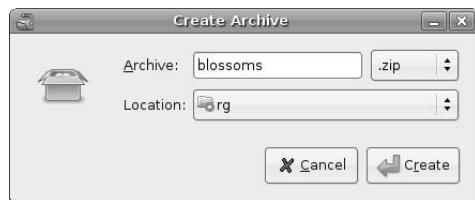

Figure 6-18: Creating a compressed archive

Now that you know how to put things together, let's get back to work and learn the equally simple task of ripping it all apart—well, okay, *extracting* it.

1. Drag the original dogwood and violet files to the Trash to get them out of the way.
2. Double-click the **blossoms.zip** file you've just created. A window showing the contents of the file will then appear (Figure 6-19).
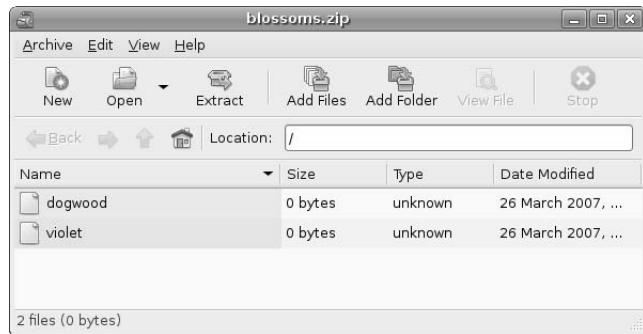

Figure 6-19: Extracting a compressed archive

3. In that window, click the **Extract** button, after which another window, Extract, will appear. Click the **Extract** button in that window, and within a second (two at the most), you will find two new copies of dogwood and violet in your home folder.

Now you've created and extracted a compressed archive, which is in this case a Zip file. You can also create a compressed tarball in the future by following the same procedure (hopefully with real rather than dummy files), but when it's time to select an archive type, select **.tar.gz** instead of .zip. Other than that single step, it is the same creation and extraction process.

7

DRESSING UP THE BIRD

Customizing the Look and Feel of Your System

Before entering the world of Linux, I had used just about every desktop operating system around. Despite the differences among them all, however, one thing that I eventually suffered from in each case was a kind of visual boredom. I suppose you might call it GUI fatigue. It wasn't that I was tired of using a graphical interface; it was just that I couldn't help but get sick of looking at the same old icons, window borders, and color schemes. Of course, there were some changes that could be made, but it just wasn't possible to get around the basic look and feel without add-ons that demanded a price in terms of performance.

One of the features of Linux that pleased me to no end, and continues to do so, is that users can drastically change the look of things. I don't mean just the icons and backgrounds, but everything, including the actual window borders and controls. Add to that the variety of graphical desktop environments and window managers available for Linux, and you have a totally customizable system. Is it any wonder that there are so many more Linux

desktop screenshots out there on the Web than for any other system? If you don't believe me, just have a look at a site dedicated to Linux screenshots, www.lynucs.org, and click the **Screenshots** link.

You may not be as fickle as I am in terms of the look and feel of your system, but you can learn to use and enjoy all the graphical customization power that Linux offers you as you work through this chapter.

Project 7A: Creating a New User Account

If you are reluctant to alter the look of your present setup, you can create a new user account and experiment with making the changes in this chapter when logged in to the new account. If you opt to go this route, your regular home environment will remain untouched because look-and-feel customizations that are performed in one user account do not affect other user accounts. When you are all done with the project, you can then simply delete the new user account. Either way, it's up to you.

To set up a new user account, follow these steps:

1. Go to the **System** menu, and select **Administration ▸ Users and Groups**.

2. When prompted for your password, type it, and then click **OK**. The Users settings window will open.

3. In the Users settings window, click the **Add User** button, which will bring up a New User Account window.

4. In that window, type a new username: `graphika`. In the Real Name field, you can type whatever you like; I used *Graphics Lover* in the example. Then move on down to the Password section of the window, skipping over Contact Information, and type an easy-to-remember user password in the two password boxes; in this case, the one you're using for your present account will do just fine. Once you've done all this, your window should look more or less like that in Figure 7-1.

5. If everything looks fine and dandy, click **OK** to close the window and get back to the Users settings window, which will now list your new user account right below your current one (see Figure 7-2).

Before using the graphika account, you need to give yourself user privileges to install software while logged in to that account. Normally, this privilege is disabled by default on new user accounts, as you probably don't want your kids, workmates, or anyone else with their own user account on your computer installing all sorts of weird stuff and screwing up your system settings.
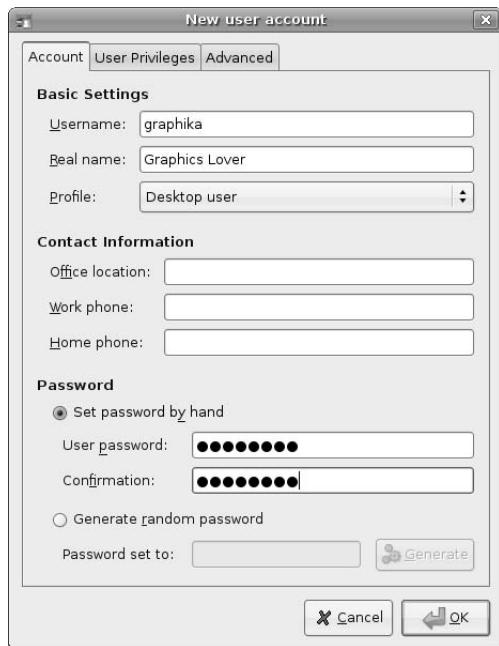
Figure 7-1: Creating a new user account
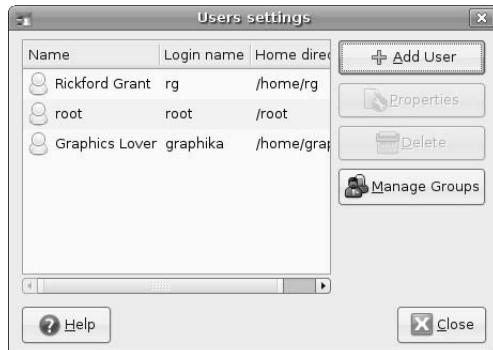
*Figure 7-2: The new user account displayed in the
Users settings window*

To change the privileges for the graphika account, click the **graphika**
entry within the Users settings list to highlight it. Once you've done that,
click the **Properties** button. A new window, Account 'graphika' Properties,
will then appear. In that window, click the **User privileges** tab, and then check
the unchecked box next to the words *Administer the system* (as shown in Fig-
ure 7-3). Once you're done, click **OK**, and then exit the Users settings window
by clicking **Close**.
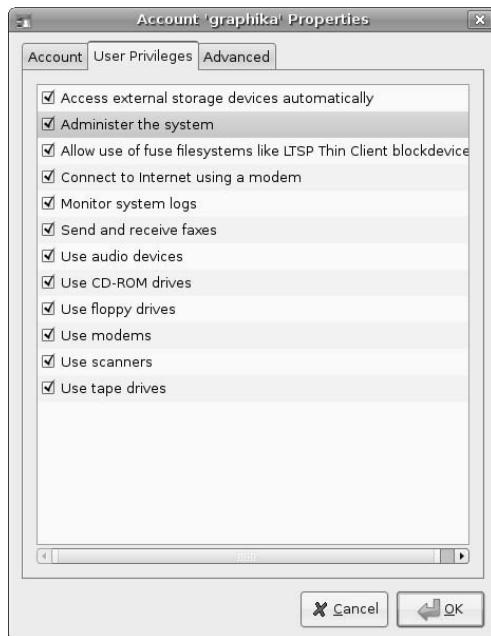
Figure 7-3: Changing user account privileges

Logging In to Your New Account

To use this new account, go to the **System** menu and select **Log Out** (or just click the **Logout** button at the far-right corner of the top panel). When the logout window appears, click **Log Out**. After a few seconds, you will be back at the login screen. Type your new username in that window, and then press ENTER. After that type the new password for the account, and press ENTER again. You will soon be at the desktop for your new user account.

Switching Users

It's worth knowing that there is another way to switch users other than simply logging out of one account and into another. This alternative approach is logically referred to as *switching users.* Switching users differs from the logout/login approach, in that you remain logged in to your original account while you log in to your other account (or while someone else with an account on your computer logs in to theirs). Going this route keeps all of the windows or applications you have open. These windows will not appear in the account you are switching to, but they will be there, conveniently waiting for you, when you switch back to the account from whence you came.

This is a good way to proceed if you plan to be switching back and forth between your two accounts. It is also a good approach when, say, your child needs to log in to his or her account for a moment to do a quick email check, burn a CD to play on the way to the beach, or print a file for school. When your child is done, you can quickly get back to what you were doing before without having to reopen files, web pages, or whatever else you happened to be dealing with at the time of the switch.

You can switch users in much the same way you would using the logout/login approach. After clicking the **Quit** button in the top panel, you can click **Switch User** instead of Log Out. You are then delivered to the login screen where you type the username and user password for the other account (each step followed by a press of the ENTER key), just as you would normally log in.

To get back to your original user account after going the switch-user route, just click the **Quit** button at the right corner of the top panel, click **Switch User** (if you plan to return shortly to the new account) or **Log Out** (if you don't plan to return soon), and then once back at the login screen, type your original username, press ENTER, type your password, and press ENTER again. In the small window that then appears, click the **Return to previous login** button.

Another window will appear, in which you must type the user password of the account you are returning to. Type your password, click **Unlock**, and you will be back at your original desktop, with everything as it was when you last saw it, open windows and all. Pretty cool.

Logging In to Another Account in a Separate Window

If switching back and forth between accounts seems like too much of a hassle to you, there is yet another, almost surreal, alternative—logging in to another account in a separate window while still in your regular user account.

This being-two-places-at-the-same-time approach requires a little bit of work up front, but it provides a lot of convenience in the long run, not only when going through Project 7B on page 106, but also whenever one of those "Hey 'rents, can I check my email for a second?" moments arises. To set things up so that you can do this, you need to install, set up, and use an application called Xnest. Here are the steps:

1. Run Synaptic, do a search for *xnest*, and install it.
2. Once Xnest is installed, close Synaptic, right-click the **Applications** menu, and select **Edit Menus** in the pop-up menu.
3. When the menu editor window appears, click **System Tools** in the left pane, and then check the box next to *New Login* in the right pane.
4. Right-click **New Login**, and in the Launcher Properties window that appears, change the command to `gdmflexiserver -n` (making sure there is a space before `-n`).
5. Click **Close** in both windows to finish.

After performing these steps, you can log in to your new account by going to the **Applications** menu and selecting **System Tools ▸ New Login**. An Xnest window will appear, and after a few seconds, the Ubuntu login screen will appear within that window. In that screen, type your new username (*graphika*, or whatever name you chose) and password as you would during any other login. Your new desktop will soon appear in the Xnest window (Figure 7-4).

When you are done with your Xnest session, click the **Logout** button at the far right of the top panel within the Xnest window, and then click **Logout** in the set of choices that appears. The Xnest window will automatically close after you have been logged out of that session.
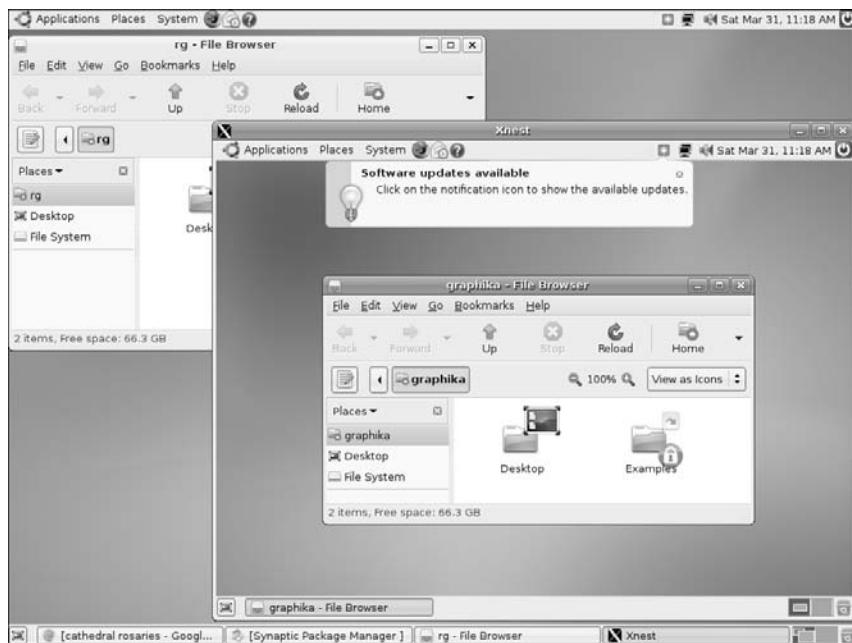
Figure 7-4: Using Xnest to log in to another account while still in your own

Project 7B: Customizing Your Desktop Environment

Whichever user account you've decided to play with, you are now ready for action. We are going to start off easy by just creating folders, but by the time we get to the end of the process, you will have created a much wilder, and, depending on how you look at things, gaudier desktop environment than you've ever seen before. All of this is in good fun, of course, and when you are done, you should be able to completely and confidently customize things the way you want on your own. So let's go.

7B-1: Creating Folders

To get started, open your home folder. Unless you've been working on your own without me, there should be nothing there at this point except for the Examples folder, which I mentioned in Chapter 3, and a folder called Desktop, which basically contains any files, folders, and shortcuts to programs that you have put on your desktop. If you are anything like me, you will want to put an end to this rather empty state of affairs by creating some folders in which you can organize your files in the future.

The first folder you will create will be for your documents, which you'll call Documentia, though you can, of course, change the name later if you like. You can create a folder quite easily, much like you do in Windows, either

by clicking the Nautilus **File** menu and then selecting **Create Folder** or by right-clicking in an empty space within the window and then selecting **Create Folder** from the popup menu. An untitled folder will appear.

Having a folder called untitled is, of course, a tad goofy, so you will want to change that. The text in the box below the folder will already be highlighted, so just type `Documentia`, and press ENTER. Once you have done that, the folder will show its new name.

Now you can repeat the process and create four more folders: Photos (for your photos, of course), Downloads (where you can dump any used files you will download along the way in this book), PDFs (for PDF files), and Music (for music files you will later learn to rip from music CDs with the program Sound Juicer). Once you have done all this, your home folder window should look like that in Figure 7-5.

Figure 7-5: Adding folders to your home folder

7B-2: Adding Emblems to Folders

You must admit, things do indeed look a bit better than before, but this is just the beginning. To graphically remind yourself what each folder is for, you can add little folder-top icons called *emblems*. These can be added to any folder or file. For now, let's add one to the **Documentia** folder by right-clicking it and then, in the popup menu, selecting **Properties**. When the Properties window appears, click the **Emblems** tab and then scroll down until you see the emblem called Documents (Figure 7-6). Click the checkbox next to **documents**, and then click the **Close** button. The emblem should now appear on your folder.

Now, for additional practice, try adding the **sound** emblem to your Music folder. Just use the same steps as before, and substitute the appropriate items and entries.
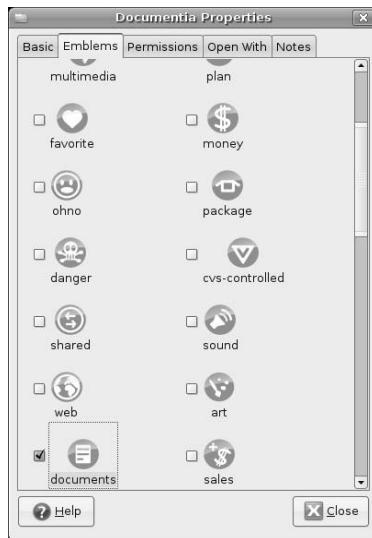
Figure 7-6: Choosing emblems for your
folders

7B-3: Setting Window Backgrounds (and Emblems Again)

Once you've added those two emblems, your folders should look a bit
spunkier (and we'll make those emblems look spunkier still further along
in the chapter). Nevertheless, the background of the Nautilus window is
still white. You need not stand for that if you don't want to; you can change
it as well. To do so, just go to the menu bar of your home window, click the
Edit menu, and select **Backgrounds and Emblems**. The Backgrounds and
Emblems window will then appear (see Figure 7-7).
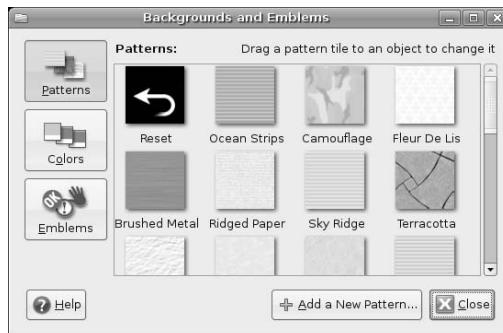
Figure 7-7: Choosing a background for your Nautilus
window

From this window, you can drag any pattern into your home window, or
into any other Nautilus window for that matter, and the pattern will then
become the background for all your Nautilus windows. So, for experience's
sake, scroll down to find the pattern swatch called **Manila Paper**, and then
drag it to the white space in the main pane of your home window. Once

you've done that, the previously white window area will look like the wallpaper in a lawyer's office. Very nice, if you like that sort of thing. You can change it to a different background in the same way, of course, or you can go back to the default white by dragging the **Reset** swatch into the window.

NOTE *If you prefer to use an image of your own for the Nautilus window background, you can also do so quite easily. Just locate the image in a new Nautilus window, click it with both the left and right mouse buttons (or just the middle mouse button, if you have a three-button mouse), and then drag the image to any open space within the target window. When you release the buttons, select **Set as Background** in the popup menu that then appears.*

In addition to the buttons for pattern and color swatches, there is a third button in the Backgrounds and Emblems window called Emblems. Clicking the Emblems button reveals all of the emblems you saw in Project 7B-2 on page 107, thus providing you with another way to add emblems to your folders. This method is far handier when adding emblems to several folders or files in the same go.

To see how this works, click the **Emblems** button. Then drag the **Camera** emblem onto your Photos folder, **Package** onto your Downloads folder, and **Special**, for lack of a better choice, onto your PDFs folder. The selected emblems will then immediately appear on those folders.

7B-4: Dolling Up the Side Pane (and Emblems Yet Again)

Now let's change the look of the Nautilus side pane. Keeping the Backgrounds and Emblems window open (if you already closed it, open it again), click the **Places** menu button in the Nautilus side pane, and select **Information**.

You can add a different background pattern to the side pane now as well, but for practice let's add a color instead. To do this, click the **Colors** button in the Backgrounds and Emblems window. The window will now be filled with swatches of color. Drag the **Grapefruit** swatch to your side pane, and it will turn from gray to, of all things, grapefruit (albeit a very dark and unusually colored grapefruit). You can also create a two-color gradation effect by adding yet another color. Drag the **Mango** swatch to the very bottom of the side pane (but still within the pane), and you should have a grapefruit-to-mango, top-to-bottom gradation within the pane. Of course, if you are not pleased with this tropical color set, you can get back to your original default gray panel by dragging the **Reset** swatch onto the area. When you're done, you can close the Backgrounds and Emblems window.

The side pane of your Nautilus window provides yet a third way to work with emblems. But before I let you in on this third, and last, way, you will need to add two more folders to your home folder. Create one folder and name it Finances, which you can use to store files dealing with your relative worth in the modern scheme of things, and then create another and call it MyFaves, where you can place . . . well, your favorite files.

After you've created the new folders, go to the side pane, click the **Information** drop-down menu, and select **Emblems**. A list of emblems will appear within the side pane. Select the **Money** emblem, and drag it onto

your Finances folder. Next, select the **Favorite** emblem, and drag it onto your MyFaves folder. Your window should now look like that in Figure 7-8. Once you are done, go back to the drop-down menu and select **Information** to get everything back to relative normalcy again.


Figure 7-8: Selecting emblems from the Nautilus side pane

Even if it's not your cup of tea, you have to admit that your Nautilus window is definitely more colorful now. You can, of course, change it to look however you want it to, but I'll ask you to hold off on that a little while longer, because you are going to be doing a some more playing around with it shortly.

7B-5: Changing the Desktop Background

Now that your home folder window is all gussied up (or gaudied up, depending on your aesthetic sense of things), you may feel that your desktop looks rather drab in comparison.

Changing the desktop background (often called *wallpaper*) is easily achieved by right-clicking any open space on the desktop and selecting **Change Desktop Background** in the popup menu. This will bring up the Desktop Background Preferences window (see Figure 7-9).

Installing Additional Wallpapers

As you can see, the default wallpaper in Ubuntu is called Simple Ubuntu, but other than that there really isn't much for you to choose from. To remedy the situation, you need to provide some images of your own. These could be photos from a digital camera, works of art you created on your computer, or just about anything you want to put there. In this case, however, you are going to venture out onto the Web to get and then install some wallpaper. A number of sites provide free desktop wallpaper, such as those for automobile manufacturers, singers, television shows, and so on.
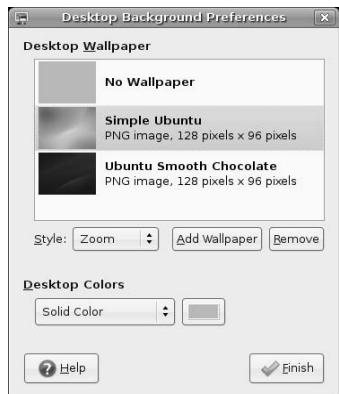
Figure 7-9: Changing your desktop background

Two sites specifically geared toward Linux users are www.kde-look.org and http://art.gnome.org, but you can get wallpaper from wherever you like. If you want to follow along using the same bigger-than-life-Tux wallpaper that I use here, go directly to the wallpaper image by pointing your web browser to www.taiabati.com/linux/OLDindex.php, scrolling down the page a tad to the second TUX section, and then clicking the download button next to the image size that best matches your screen. When the picture appears in the browser window, right-click it, and then select **Save Image As**. In the Save Image window, give it an easy-to-remember name (*4wall_TUX-2_1024x768.jpg* is exact, but a bit much to deal with) or use the one I gave it, *mightyTux.jpg*, and click **Save**. If you prefer, you can download any wallpaper you like from wherever you like, as long as it is in a supported format, such as BMP, PNG, or JPEG. It's all up to you.

Once you've downloaded your wallpaper, place it in your Downloads folder. After that, you can install it by going to the Desktop Background Preferences window and clicking the **Add Wallpaper** button. In the Add Wallpaper window that then appears, navigate to your new wallpaper, click it once to highlight it, and then click **Open**. The wallpaper will then appear highlighted in the Desktop Background Preferences window and will soon thereafter appear on the desktop itself (Figure 7-10). Once it does, click **Finish** to complete the process.

Wallpaper from Net to Desktop—Quick and Easy

It is also possible to almost automatically set an image from the Web as your desktop wallpaper by right-clicking that image within your web browser and then selecting **Set as Desktop Background**. A small window will then appear, in which you can preview what the download will look like on screen. You can also adjust the position (tiled or centered, for example) and background color for your desktop in this window. Once you're done making adjustments, click the **Set Desktop Background** button. The image will then appear on your Desktop, while the image file itself will be saved to your download location (your home folder, by default) with the title *Firefox_wallpaper.png*.

Hiding the Bottom Panel

By the way, you may have noticed that you can barely see the panel at the bottom of the screen in Figure 7-10. This is because the panel obscured the bottom of the new wallpaper, which irritated me. I went to the Panel Properties window by right-clicking some empty space in the bottom panel and selecting **Properties** in the popup menu. In the Panel Properties window, I clicked the checkbox next to the word **Autohide** and then clicked **Close**. The autohide function works just like it does in Windows or Mac OS X—the panel stays out of view until you move your mouse into the general vicinity of where it should be. You can make the same change if you like, but that is an aesthetic matter that I will leave up to you. Ah, the sweet taste of artistic freedom.
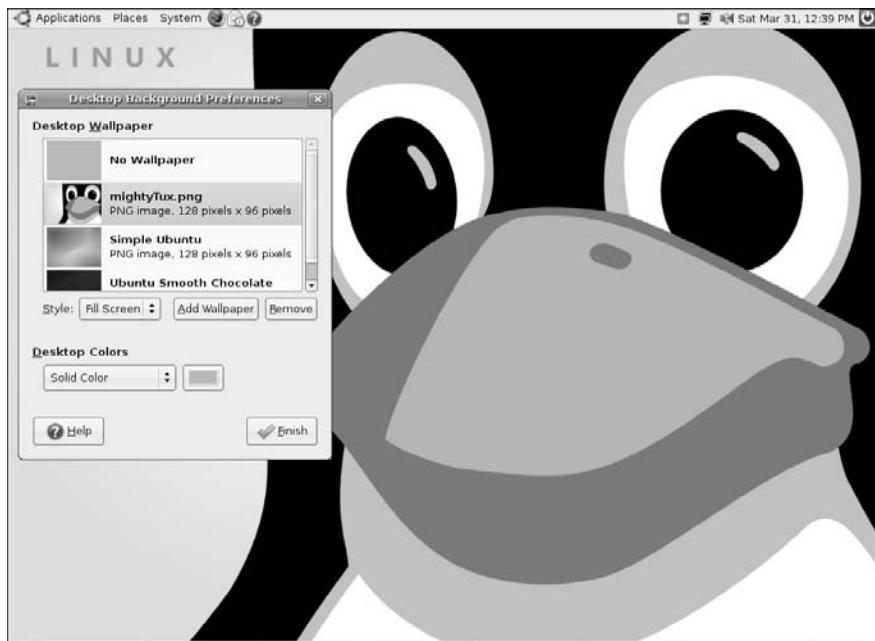
Figure 7-10: The newly wallpapered desktop

7B-6: Downloading and Installing the Art Manager (GNOME Art)

Searching the Internet for wallpaper to install can in itself be a rather fun adventure, but sometimes it can also feel like quite a chore. Fortunately for you, me, and all involved in such things, there is an even easier way: the Art Manager. The Art Manager, also known as GNOME Art, is a handy application that searches the art.gnome.org site, and downloads a list, with thumbnails, of all the wallpapers that are available there. It can also do this for the various window border, controls, and icon theme sets that you can use in the following parts of this project. Using the thumbnailed lists, you can easily download and install whatever you want—all without ever placing a cursor in your web browser. Needless to say, the Art Manager is decidedly cool!

Unfortunately, the Art Manager is not installed by default; however, after having gone through Chapter 5, you know how easy it is to download and install applications like the Art Manager. All you have to do is run Synaptic, do a search for *gnome-art*, and then install it.

You can then run Art Manager by selecting **System ▸ Preferences ▸ Art Manager**. The GNOME Art window will then appear with absolutely nothing in it. To put it to use, and relieve that emptiness, go to the **Art** menu, and select **Backgrounds ▸ GNOME**. (You can select **All** instead of GNOME if you like, but it will take longer to download the list of available wallpapers.)

Once your selection is made, the Art Manager will begin downloading a list of all that is available for you at http://art.gnome.org. It may seem like nothing is happening for a minute or so, but that is normal; just hang in there. When it's done, you will see a list of thumbnails for you to choose from (Figure 7-11).
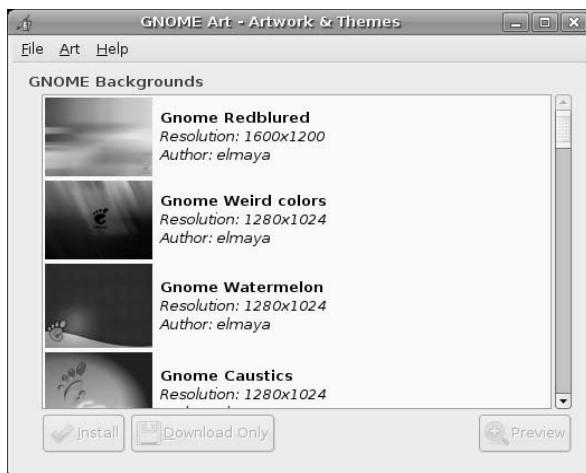
Figure 7-11: Installing desktop wallpapers using the Art Manager

You can now install a wallpaper by scrolling down until you find one that suits your fancy, clicking it once to highlight it, and then clicking the **Install** button. Art Manager will then download it, install it, and then automatically open the Desktop and Background Preferences window so that you can immediately apply it to your desktop if you like. As I said before, it's a very handy tool to have, especially since you'll be using it more soon within this project.

7B-7: Changing Window Borders, Controls, and Icon Sets

Now we get to my favorite part of this journey through the world of digital cosmetic surgery—changing the way window borders and controls look in GNOME. Let's set about doing just that.

The procedure is really quite easy. Go to the **System** menu, and select **Preferences ▸ Theme**. The Theme Preferences window will open and show you a list of the themes that are installed on your system (see Figure 7-12). The default theme in Ubuntu is called Human, but, as you can see, there are several others.

To get the hang of things, have a look at each of the themes listed by clicking them one by one. The changes will take effect immediately. Just clicking on a theme will change your window borders, controls, and even, if you take a peek in your home folder, the icons. This is especially noticeable when you click Crux or Glider.

Each theme consists of a window border, a set of controls, and a collection of icons. This being the case, it is possible to mix and match these elements on your own. For example, let's say that you like the look and color of the quasi-industrial controls in Crux, but you prefer the window icons in Mist and the borders in Human. Well, you needn't despair, because you can create a custom theme consisting of these three different elements.

Figure 7-12: Selecting a theme in GNOME

To create your own mix-and-match theme, just click the **Customize** button in the Theme Preferences window. A new window will open, and you will find three tabs: Controls, Window Border, and Icons. From within each of these tabs you can select the components you prefer. First click the **Controls** tab and select **Crux**. Then click the **Window Border** tab and select **Human**. Finally, click the **Icons** tab and select **Mist**.

Now keep the Theme Details window open, but open your home folder and take a look at what you've done. Hmm . . . not bad. But, perhaps you don't really like the look of those Human window borders all that much. To find something that suits you better, click the **Window Border** tab again, go down the list, and click each entry until you see something you do like (Metabox seems to do the trick for me), and select that. Better? Now that you are satisfied, you can click the **Close** button.

You will now be back at the Theme Preferences window, where you will notice that at the top of the list, there is a new theme entry called Custom Theme. If you want to save this new combination for later use, click the **Save Theme** button. Doing so will open a dialog box in which you can name your

theme and write a brief comment about it. So, name your theme, write a comment if you like, and then click **Save**. Your new theme will now appear in alphabetical order within the theme list under the name you chose.

Once that's all done, your home folder window should look like that in Figure 7-13 (and take a look at your panel and Applications menu while you're at it). Ah, très cool!

Figure 7-13: Changing the look of the home folder window

7B-8: Installing Additional Window Borders, Controls, and Icons

If you are excited about this customization thing but you're not satisfied with the theme choices included with the system, you can download and install still other window borders, controls, and icons. To show you how to do this, I will walk you through creating a faux Mac theme, which will look fairly similar to the standard Aqua theme of Mac OS X, as you can see in Figure 7-14.

Figure 7-14: An Aqua-fied Ubuntu desktop

Getting and Installing the Files You'll Need

To get the files you'll need to do this, take the Art Manager for another ride. Once it is up and running, go to the **Art** menu, and select **Backgrounds ▸ Other**. Once the list of available wallpapers appears in the Art Manager window, scroll down until you find one called **Real shoot**, install it by clicking the **Install** button, and then select it and apply it in the Desktop Background Preferences window, which will automatically open.

NOTE   *If Art Manager does not automatically start downloading a list of available files when you make a selection from the Art menu, just restart Art Manager and try again.*

Now to get an appropriate window border for your new theme, go back to the **Art** menu, but this time select **Desktop Themes ▸ Window Border**. Look for one called **Hacked**, and install it by clicking the **Install** button. When the Theme Preferences window automatically appears, click the **Customize** button, and click the **Window Border** tab. In the list that appears in that tab, click **Hacked**. Close the Theme Details and Desktop Background Preferences windows to complete the task.

Next get a set of matching application control widgets by going back to the Art Manager **Art** menu and selecting **Desktop Themes ▸ Application**. When the list is downloaded, look for a file called **Yattacier 3**, and install it. In the Desktop Background Preferences window that then appears, click **Customize** as you did before—but this time around click the **Controls** tab, and select **Yattacier3** in the list within that tab.

To round things up, let's add some new icons to the mix by going back to the Art Manager, heading to the **Art** menu, and selecting **Desktop Themes ▸ Icon**. Once the list is downloaded, look for and install **Snow-Apple**. After that it's basically a repeat of the previous step, but this time around, click the **Icons** tab in the Theme Details window, and then select **Snow Apple**.

Finishing Touches

Well, things are certainly sort of Mac-ish now, but there is even more we can do to emphasize the effect. First, open a Nautilus window, click the **Computer** button, and then drag the **Filesystem** icon to your desktop. This will create an alias of your hard disk there. Next, go to the bottom panel, and remove everything except Trash. After that, open the Preferences window for your bottom panel by right-clicking the area to the far right of Trash and selecting **Properties** in the popup menu. In the **General** tab of that window, uncheck **Expand**, and then increase the size of the panel to around 54 pixels. When you're done, click **Close**, and then start adding launchers for the applications you use most.

Finally, go to the top menu, and remove the three icons next to the System menu. After that, add a Window Selector applet, so that you have some way to navigate through your open windows. You might also want to change the background in your home folder, since the warm tones presently there no longer match your new cooler configuration.

The transformation is now complete, and if you followed along correctly, your desktop should look something like mine back in Figure 7-14. You can stick with your new OS X–ish theme, or switch to something else.

For consistency's sake, I will switch back to *Human* now. By the way, if you do decide to keep the faux-Aqua theme, remember to click the **Save Theme** button in the Theme Preferences window and give the theme a name.

Project 7C: Placing Home and Trash Icons on the Desktop

As you are already aware, unlike Windows, Mac OS X, or other Linux distributions, Ubuntu has a completely empty desktop upon installation. A lot of people advocate this approach because it discourages the permanent use of the desktop as a location to store files and program launchers. After all, as the argument goes, you don't place your trash can or file cabinet on the desktop in your office, do you?

All such logic aside, there are still many people who prefer to have their trash can, hard disk, and home folder on their desktops, thank you very much. If you are one of them, as I am, here's what you need to do:

1. Press ALT-F2 to bring up the Run Application window. This keyboard shortcut is the equivalent of clicking the Run Application panel applet that you placed on the panel in your original user account.

2. Run the GNOME Configuration Editor by typing `gconf-editor` in that window and then pressing ENTER.

3. When the Configuration Editor window appears, click the small arrow next to apps, scroll down to nautilus, and click the small arrow next to that.

4. Click **desktop** in that expanded nautilus section, after which the options for that item will appear in the right pane of the window (Figure 7-15).
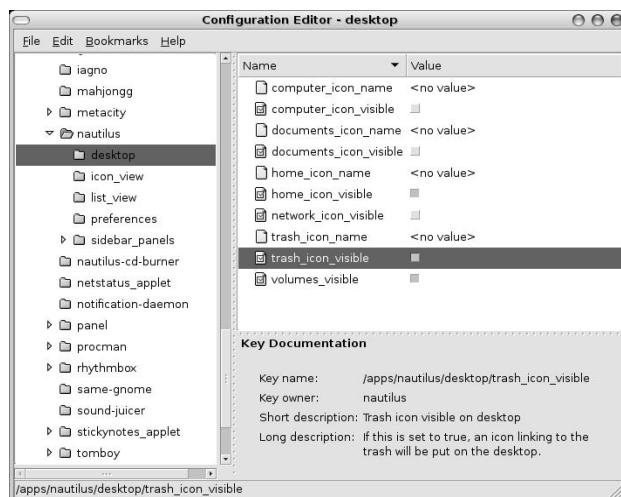
Figure 7-15: The Configuration Editor

5. Check the boxes next to the items you would like to appear on the desktop. You have four unchecked choices to choose from: computer_icon_visible (like My Computer in Windows), documents_icon_visible

(to create a link to your Documents folder, if you have one), home_icon_visible (for quick access to your home folder), and trash_icon_visible (for you-know-what).

6. When you're done, close the Configuration Editor.

NOTE *Changes made in the way I just described will affect all user accounts. If you choose, for example, to show the trash can on the desktop for one user account, it will appear there for all others.*

Project 7D: Changing Your Login Screen

You may have noticed while working with the Art Manager that there is a menu selection (**Art ▸ Other Themes**) for something called Login Manager. The Login Manager is your login screen, also known as a *greeter*—the screen where you type your username and user password when you first log in to your system. The Login Manager is another thing that you can customize, but be forewarned that any changes you make will be system-wide, not just for you; whatever Login Manager theme you install and choose will be the Login Manager theme that everyone else with user accounts will see when they use the machine. Of course, if you are the only one using your machine, this point is moot.

7D-1: Downloading a Login Manager Theme

In order to try customizing the Login Manager, run the Art Manger, and select **Art ▸ Other Themes ▸ Login Manager**. Once you've done this, browse through the various themes in the list, and choose one or two that you want; I give you free rein on this one, but I'll be choosing the theme called GDM-ManzanaTux to complete the faux-Mac theme, in case you want to follow along exactly.

Unlike your other experiences with the Art Manager up to now, Login Manager themes can only be downloaded, leaving you to do the installation yourself. That being the case, once you've made your selection, click the **Download Only** button. Once the download is complete, you will find the file in archive form, with a tar.gz ending, in your home folder, or any other folder you decided to save it to. You do not need to extract, or *untar*, the archived file.

NOTE *If you have any trouble installing greeters via Art Manager, go to http://art.gnome.org, click the Login Manager link on that page, and then download a greeter directly.*

7D-2: Installing Your New Login Manager Theme

Once you've downloaded a theme or two of your liking, you need to open the Login Screen Setup window. To do this, go to the **System** menu, and select **Administration ▸ Login Window**. A dialog box asking for your password will then open. Type the password for the account you are currently using, and click **OK**. The Login Window Preferences window will soon appear (see Figure 7-16), showing a thumbnailed list of the greeters available.

You can add the greeters you just downloaded to this list by dragging the files directly to the list. A small window will then appear, asking if you're sure that you want to install the file you've just dragged to the list, and since you do want to install the file, click **Install**.

Figure 7-16: Customizing the Login Manager

To select the greeter you wish to use, just click the round button next to its name in the list in the Login Window Preferences window, and then click the **Close** button. Of course, to see your greeter in action, you will have to log out first so you can log back in, but you needn't restart or shut down the machine. You can then see your new greeter when the login screen appears (Figure 7-17), though yours may well be different.

Figure 7-17: The new login screen

Project 7E: Changing Your Splash Screen

Well, now you've changed just about all there is to change system-wise, but there is actually one more item that you can tinker with—your splash screen. In case you're not familiar with the term *splash screen*, it's the screen that appears when GNOME is starting up right after you log in but before you get to your desktop. The default screen, shown in Figure 7-18, is pleasant enough, but you can choose something a bit wilder or at least more colorful if you are so inclined.

Figure 7-18: The default Ubuntu splash screen

7E-1: Enabling Automatic Login

While the Login Window Preferences window is open, it's as good a time as any to mention an option that may be of interest to you. If you find it a bit of a drag to type your username and password every time you start up your machine, you'll be happy to know that you can bypass the whole login process. If you share your machine with other users, of course, this isn't something you'd want to do because anybody with access to your machine would have access to your user account. I also wouldn't recommend doing this on a laptop, since they are more easily lost or stolen, thus leaving your data at risk to absolute strangers.

To enable automatic login, click the **Security** tab in the Login Window Preferences window, and then click the checkbox at the top of the page next to the words *Enable Automatic Login*. After that, click the arrow on the right side of the drop-down menu next to the word *User*, and select your username from the list. Once you're done, click the **Close** button. The next time you start up your machine, you will bypass the login screen and be delivered directly to the desktop.

7E-2: Installing New Splash Screens

Ubuntu does not come with any alternative splash screens for you to play with, so in order to make changes, you first need to download and install some splash screens with which to work. This can be done quite easily by opening Art Manager (**System ▸ Preferences ▸ Art Manager**) and following these steps:

1. Go to the **Art** menu, and select **Other Themes ▸ Splash Screen**. Art Manager will then download the available splash screens.
2. Once the download is complete, scroll through the offerings, and select one that suits your fancy by clicking it once.

3. After your selection has been made, click the **Install** button, after which Art Manager will download and install the file. When it's done, the GNOME Splash Screen Preferences window will appear, showing the newly installed screen.

4. Go back to the Art Manager, and repeat steps 2 and 3 to add a few more screen choices to your repertoire.

5. Once you have installed three or four splash screens, close Art Manager.

7E-3: Selecting and Activating Splash Screens

When you have multiple splash screens installed on your system, you can select and then activate them via the GNOME Splash Screen Preferences window (Figure 7-19), which should already be open. When it isn't, you can bring it up by going to the **System** menu and selecting **Preferences ▸ Splash Screen**.

Figure 7-19: Choosing a new login screen

When the window first opens, all of the installed screens will be grayed out, and thus cannot be selected. In order to select and activate a particular splash screen, first check the box next to the words *Show splash screen on startup.* The thumbnails for the installed screens should now be selectable, so make your selection by clicking on the splash screen of your choice once, and then click the **Activate** button. You will then see your splash screen in action when you next log in to your system.

Choosing a Screensaver

Screensavers used to be a must-have (and must-use) item for computer users who wanted to prevent damage (burn-in) to their monitors. Video display technology, however, has now advanced to the point where screensavers are no longer completely necessary. Nevertheless, screensavers are cool to look at, and one very nice thing about Ubuntu is that it comes with an unusually extensive collection of screensaver modules—nearly 200 of them! There are so many that you are sure to find at least a few you like. The screensaver settings are preconfigured to switch between modules

randomly, changing the current module every few minutes. You can change these settings by going to your **System** menu and selecting **Preferences ▸ Screensaver**.

The Screensaver Preferences window, shown in Figure 7-20, allows you to do a variety of things, such as set the length of time between module changes and the length of idle time before the screensaver starts up. You can also opt to use only one screensaver module or no screensaver at all. In random mode, you can also omit the modules that you don't like by unchecking the boxes next to the undesired modules' names. This can be quite useful, especially if certain screensavers seem to tax the graphic capabilities of your machine.
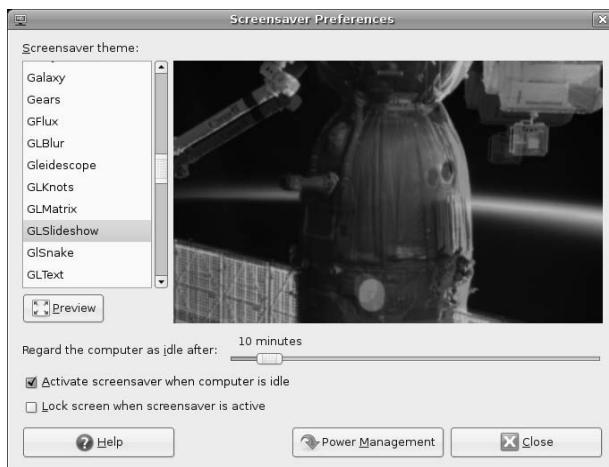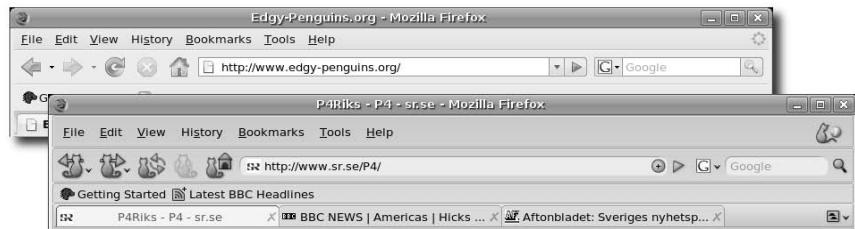
Figure 7-20: Setting screensaver preferences

Project 7F: Wrapping Things Up—Installing and Applying Firefox Themes

It's time to bring this rather lengthy exercise in customization to a close. Now that you know how to do most of the things you might want to do in this area, you can return to your own user account and use what you've learned to make changes that suit your own tastes and desires.

If you've really gotten hooked on this customization thing, you'll be happy to know that there are always more things you can bend to your will. In addition to being able to customize your system sounds, the splash screen that appears after you log in, and the system fonts that are displayed in your menus and folders, you can also play around with the many applications that allow customizations through the use of skins designed specifically for those apps.

Perhaps the most notable of these customizable applications is Firefox, which you first met in Chapter 4. Firefox allows you to change the look of its buttons, application controls, and even background colors through the use of *themes*, some examples of which you can see in Figure 7-21. You will be glad

Figure 7-21: Firefox themes

to learn that downloading and installing these themes is a fairly straight-forward process, so to finally wrap things up, I will end this chapter by providing you with the steps you need to follow:

1. In Firefox, go to the **Tools** menu and select **Add-ons**. The Firefox Add-ons window will then appear.

2. At the top of that window, click **Themes**.

3. In the bottom-right corner of that window, now click the **Get More Themes** link.

4. A new Firefox window listing available theme categories will then appear. Click any of the links to see previews of the themes.

5. When you find a theme you like, click the link for that theme, and then click **Install Now** in the window that opens.

6. A confirmation window will then appear, asking if you're sure that you want to install the theme in question. Click **Install Now**.

7. Once the theme is downloaded and successfully installed, it will appear in the list of available themes in the Themes and Installation pages of the Add-ons window (Figure 7-22). On the Themes page, select the theme you want to use by clicking it once, and then click the **Use Theme** button when it appears.

8. To see your new theme in action, restart Firefox.

Figure 7-22: Selecting Firefox themes

8

SIMPLE KITTEN WAYS
Getting to Know the Linux Terminal and Command Line

Many people shy away from Linux because they envision it as a system for compu-geeks, an environment in which you do everything the hard way—by command line. In this era of graphical interfaces, the idea of typing commands to get things done seems a dreadful throwback to the days of DOS, and that puts many people off—especially those who remember what it was like in the "old days."

This reaction is fair enough, but it is not really an accurate reflection of the reality of the Linux world. After all, most Linux users today utilize some sort of graphical interface. They can, and often do, achieve all that they hope to achieve through drop-down menus and mouse clicks alone. Many are able to survive quite happily without ever once opening their Terminal. The same could be true of you.

Be that as it may, there is still much to be said for the power and convenience of the command line. The fact that the command line can now be utilized within a graphical environment also makes it much less forbidding. The Terminal is just a tiny text-based island in a sea of graphical bodies (see Figure 8-1). Using the command line can be as pain-free as anything else you do on your system, and it can actually provide you with a little fun if you are willing to give it a try.
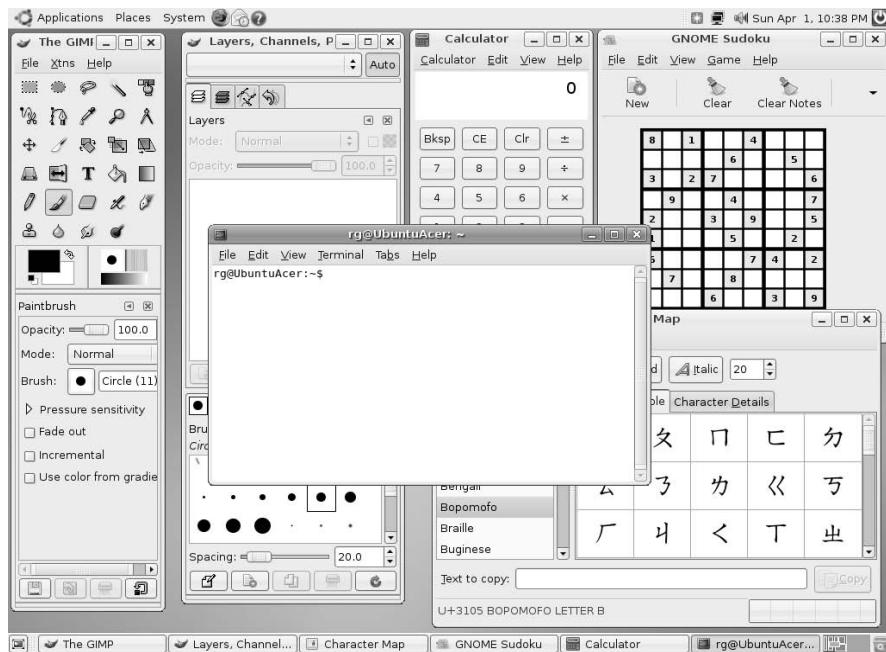
Figure 8-1: Putting the Terminal in perspective

Unfortunately, many guides to using the command line are written by hard-core command-line junkies, whose enthusiasm for what they see as a really good thing inadvertently makes what they write seem even more off-putting to the recent Linux immigrant or wannabe.

For your sake, I will try to curb my own enthusiasm so as not to scare you right back to Chapter 5 and the more comfortable world of Synaptic. I will also try to help you keep things in perspective by teaching you, whenever possible, to use the command line as a complement to the various graphical tools that you have at your disposal, rather than presenting it as the sole way of going about things. Of course, I am not going to cover every possible angle in this regard—just enough to give you some exposure and experience and, hopefully, make you feel at least a little more at ease with the command line. Who knows; could you actually come to think of using the command line as . . . fun? Well, I won't get too carried away.

Meet the Terminal

The Linux Command Terminal application in your Ubuntu system can be run by going to the **Applications** menu and selecting **Accessories ▶ Terminal**. When the Terminal opens, it will, in all its simplicity, look much like Figure 8-2.
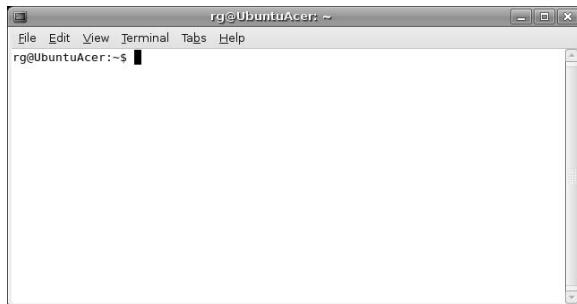
Figure 8-2: The Terminal application

As you can see, all it says is `rg@UbuntuAcer:~$`. In this case, the `rg` is my username, `UbuntuAcer` is the name I gave my computer during installation, and the tilde (`~`) signifies that I am in my home folder. If it were to say `~/Music`, for example, it would mean that I am currently in the Music folder within my home folder. Of course, all this will be different in your case, as your username and computer name will be different. If your username is *frog* and your computer's name is *wetrock*, for example, the command line will say `frog@wetrock:~$`. If all this is sounding rather obtuse to you, just think of it this way: *username@computer_name*`:~$` in the Terminal is the equivalent of your home folder in Nautilus.

Typing in the Terminal is straightforward enough; you just type as you usually do. You can also delete and insert letters or phrases by using the DELETE and BACKSPACE keys and the cursor keys. For practice, try the following:

1. Type **I like strawberries so very much**.
2. Change `strawberries` to `cherries` (because cherries are, in fact, so much better). Just use your left cursor key to move the cursor in front of the first *s* in `strawberries`.
3. Tap your DELETE key as many times as necessary to erase the word `strawberries` (uh, that would be 12 times, methinks).
4. Just type **cherries**, and then use your right cursor key to move the cursor back to the end of this meaningful sentence.

Now that you've completed this fascinating bit of typing practice, press the ENTER key. As you will almost immediately see, the Terminal's response to your efforts thus far is merely a dismissive `bash: I: command not found`. Although you've typed a string of text that has meaning to you, it means absolutely nothing to your system. In fact, the system was so shortsighted

that it could see nothing other than the first word you typed in the Terminal (I); and because I is not a valid command, the system had no idea what do to with it.

Shells

You may be wondering what this bash business is all about and why it is talking to you. Well, Bash (Bourne Again Shell) is one of the many shells that are used in Linux systems, and it's the one that happens to come with your Ubuntu distro (and most others, for that matter). A *shell* is a program that interprets the commands you type into the Terminal and delivers them, so to speak, to your system so that it can act upon them. I like to think of it as a command-handling subsystem, for which the Terminal acts as a graphical front end. Some scripting languages, as you will find out in Project 8C on page 146, also have their own shells; but other than those few exceptions, you generally need not be unduly concerned with shells other than to know what they are and what people are talking about when referring to them.

Some Goofy, Yet Useful, Fun with the Command Terminal

A rather cool thing about typing in the command Terminal is that it has what you might call short-term memory. Try it out by typing the word **cherry** and then pressing ENTER. Ignoring the command-not-found message, go on and type **vanilla**, and press ENTER. Now type **icecream**, and press ENTER. So far, so dumb, right? Well, not really. Let's type everything we've typed thus far again, but this time let's do it with only one key.

Huh?

Yes, just press the up cursor key once, and what do you see? That's right—the last command you typed appears, which in this case would be icecream. Press the up cursor key again, and the command that you typed before that will appear—vanilla. One more time? Yes, cherry. And one more time for the grand finale . . . I like cherries so very much.

Considering what we have thus far, this may all seem a bit silly, but imagine that you're not typing goofy little words and instead have to deal with considerably longer strings, such as a simple copy command (which you'll learn about later in this chapter) like

```
cp Photos/mypics/stpierre/coastal/onthebeach1_27.jpg /home/frog/
photos_for_mom/stpierre
```

By typing that string, you are copying an image called onthebeach1_27.jpg from the coastal folder to another folder called stpierre. If you wanted to copy another photo in the coastal folder, onthebeach1_16.jpg, for instance, you could simply press the up arrow key once, use the left cursor key and DELETE key to move over to and delete the 27, replace it with **16**, and then use the right arrow key to get back to the end of the command. All in all, it would be much simpler and much faster. It would also help you avoid mistakes in typing. Not so dumb anymore, eh?

Nontoxic Commands

As you now know, all of this typing is easy enough, but in order to actually do something useful with your Terminal, you need to type commands—and there are more of them than you could ever hope or need to know. To get you started, we will begin with some commands that are easy to understand, nontoxic, and completely kitten-friendly.

$ whoami

There is no command as easy, safe, or even as seemingly useless, as `whoami`. Rather than help those with multiple-personality disorders discover who they are at any given moment, the `whoami` command simply tells you which user is currently logged in. Try it out by typing **whoami** after the $ and then pressing the ENTER key. Remember that commands are case sensitive, so type accordingly.

The Terminal will now tell you the username of the person currently logged in. If you are logged in as frog, you should get `frog` as the answer to your command.

$ finger

If you enjoyed discovering who you are with the `whoami` command, then you might enjoy finding out even more about yourself using the `finger` command. The `finger` command can be used in a number of ways, but a very simple one is finding out about a particular user. Try this out on yourself by typing **finger** and then your username. In my case, that would be **finger rg**. Once you've typed the command, press ENTER and see what you get. You can see my results in Figure 8-3.
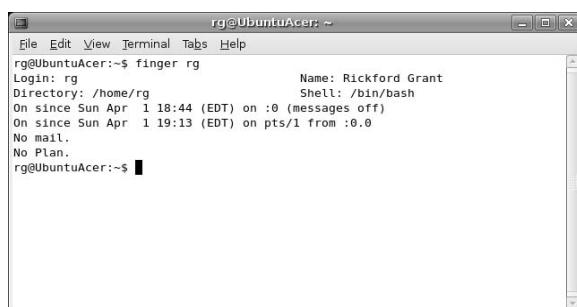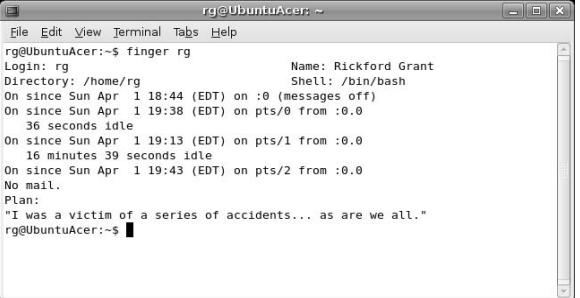
```
rg@UbuntuAcer: ~                                    _  □  ✕
File  Edit  View  Terminal  Tabs  Help
rg@UbuntuAcer:~$ finger rg
Login: rg                              Name: Rickford Grant
Directory: /home/rg                    Shell: /bin/bash
On since Sun Apr  1 18:44 (EDT) on :0 (messages off)
On since Sun Apr  1 19:13 (EDT) on pts/1 from :0.0
No mail.
No Plan.
rg@UbuntuAcer:~$ █
```

Figure 8-3: Output from the `finger` command

As you can see, my login name is rg, my real name is Rickford Grant, my home directory is /home/rg, I am using the bash shell for typing my commands, I have been logged on since Sun Apr 1 at 18:44 EDT, and I have no mail or plan. It doesn't tell you my social security number or my mother's maiden name, but it is pretty cool, don't you think?

I mentioned that the results said I had no plan, and you may well be wondering what that is all about, so I'll fill you in. A .plan file is a small file kept in your home folder that other users see when they use the `finger`

command on you. Traditionally, a .plan file contained information about where you were going to be or what you were working on. These days, however, most people use them to leave odd little messages, quotations, or whatever, much as they do in email signatures. Take a look at Figure 8-4 to see what happens after I add a .plan file to my home folder.

Figure 8-4: Output from the `finger` command with a plan

You can now see my plan, which is a quotation from Kurt Vonnegut's *The Sirens of Titan* (or Al Stewart's song by the same name, for that matter), though you can put anything you want in your own. You will get the chance to create your own plan file in Project 8A on page 140, so if this all seems fun to you, just hang in there.

Before moving on, I should mention that you can also use the `finger` command to do a little domestic espionage of sorts. Let's say your child, Chris, has a user account on your machine. Chris, who wants your permission to go to the movies, claims to have been hard at work on the computer all day writing up a report for school. Having your doubts, you could type `finger chris` to see what the facts actually are. It may be a bit underhanded and rotten, but it works. It also works both ways, so others can check up on you as well. You can give it a try right now by seeing when the last time you logged in to your graphika account was. Just type **finger graphika**, and then press ENTER.

You can even use the `finger` command to find out facts about people on other systems, providing their network's finger service is active and you know their email address. Typing something like `finger username@hostname.com` would do the trick. Kind of cool, but also kind of spooky, I suppose.

$ pwd
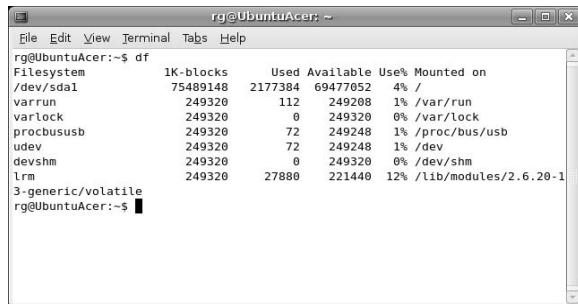
If you know who you are but aren't exactly sure where you are, `pwd` (print working directory) should come in handy. The `pwd` command tells you exactly where the Terminal is in your directory tree.

Let's say, for example, that my Terminal is in my personal home directory (which is actually called rg) in the system's home directory (which is actually called home, and which is where all the user account directories are located) when I use the `pwd` command; I would, after pressing the ENTER key, get `/home/rg` printed to my Terminal. You should get similar results if you try it out.

NOTE   *The word* print, *in this case, has nothing to do with your printer; it merely means that the response will be printed to, or displayed in, the Terminal.*

$ df

Another safe and easy, but much more useful, command is df (disk file-system). The df command tells you how much disk space you have used, as well as how much space you still have available, on each of the partitions on your various mounted disks. Try it out by typing **df** and then pressing ENTER. Your output should look something like that shown in Figure 8-5 (depending, of course, on the size of your mounted disks and how they are set up).
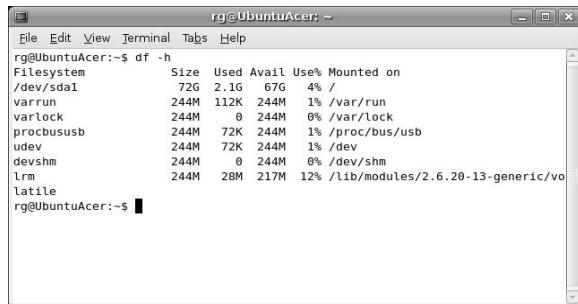
Figure 8-5: Output from the df command

As you will notice, the sizes are given in kilobytes (KB) rather than the gigabytes (GB) and megabytes (MB) you are probably more used to, but there is a way around this. Many commands accept a *flag*, or *option*, to further fine-tune how the command performs. These flags are written directly after the main command and are preceded by a space and a hyphen.

In this case, you can try using the -h (human readable) flag to have your figures come out in the way you are most familiar with. Try this out by typing **df -h** on the command line and pressing ENTER. The output should now appear in a more familiar format (see Figure 8-6).

Figure 8-6: Output from the df command with the -h flag

$ ls

Another harmless but handy command is ls (list directory contents). The ls command shows you what is in your current directory. This is the non-graphical equivalent of double-clicking a folder in Nautilus to see what is inside. Try it out by typing **ls** and then pressing the ENTER key.

If you've been following *my* commands so far, your results should list all of the folders in your home directory. You can also use the -R flag to show not only the list of files in the folder, but also what is within the subfolders. Of course, you should have no subfolders in any of the folders you created in Chapter 6, so you can hold off experimenting with this for a while. Instead, try typing **ls -a** to see your invisible, or *hidden*, files.

$ sudo

When you ran Synaptic back in Chapter 5, you were first asked to input your password before you could run the program. The reason for this, as I mentioned then, is that Synaptic installs the files it downloads in various folders throughout your system, almost all of which are write protected. By supplying your password, you are telling your system that you, as holder of the password, have the right to allow Synaptic to do that.

The command version of that same password-giving process is the sudo command. To perform an operation in a folder that is write protected, you would first type sudo and then the command you want to perform. For example, if you wanted to copy an icon image, let's call it myicon.png, to the globally located and write-protected pixmaps folder (/usr/share/pixmaps), you would type sudo cp myicon.png /usr/share/pixmaps.

After typing a command preceded by the sudo command and pressing ENTER, you will be prompted for your password. Once you type your password and press ENTER again, the command will be executed. I should mention that once you input your password, it will stay in memory for about 15 minutes. This means that you will not be prompted for your password when using the sudo command again within that time frame.

$ locate

In contrast to the seemingly lightweight commands you have learned so far, the locate command is really quite useful. In fact, you might well find it a much easier, faster, and more effective method of finding files than the graphical search tool in the Places menu. Using the command is quite easy: you simply type the command followed by a space and the name of the file you are searching for.

Before you can use this command, though, you will need to create a database of filenames for locate to use. This is quite easily done by using the sudo command, which you just learned about, and then typing **updatedb** (Whoa, Nellie! Yet another command!), followed by a tap on the ENTER key (that would be sudo updatedb). After you type your password when asked to do so, it will seem that nothing is happening for a while, but don't worry. As long as the cursor in your Terminal is blinking, progress is being made, and when your user prompt returns, you will have successfully created the database file. After that, you can go on and use the locate command. Oh, and in the future if you think that the process seems to be taking longer and longer, don't worry—it is. The more files and applications you add to your system, the longer it will take your system to catalog them all.

To take this new command out for a test drive, let's look for the openofficeorg22-writer.png file that we worked with in Chapter 3. Just type the following, and press ENTER:

```
locate openofficeorg22-writer.png
```

Your results should look like those in Figure 8-7.
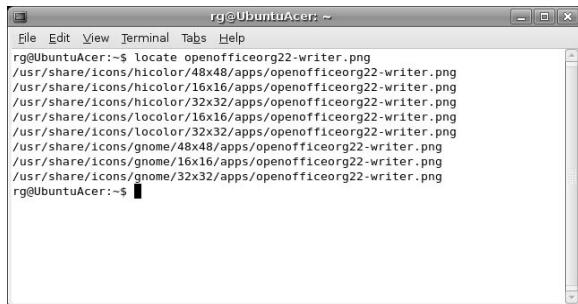
Figure 8-7: The results of a `locate` search

$ calendar

I'll let you experiment with this one on your own. Just type **calendar**, and press ENTER to see the somewhat interesting results.

$ exit

The exit command is a simple one that allows you to exit the Terminal. Just type **exit**, and press ENTER. The Terminal window will close.

Commands with Some Teeth

The simple commands you have tried so far are all of the safe-and-sane, fire marshal–approved variety; they merely print information to your Terminal. Now you are going to try to get some real tangible results from the commands you use. These commands are also essentially safe and sane if you follow my instructions.

$ mkdir

You have already learned how to create folders by means of menus and your mouse, but you can also do this using the command line. The command is mkdir (make directory), and it is easy as pie to use (though I've never been quite sure how pie is easy).

To see how this command works, and to work with the commands that follow, use the mkdir command now to create a folder called command_exp (for command experiments). All you have to do is type **mkdir command_exp** in a new Terminal window, and press ENTER. The new folder should appear in your home folder, so go ahead and check to see if it is there by clicking the home icon on your desktop.

Okay, good, *bra, bueno*! Now let's create another new folder within that new folder—a *subfolder*, if you will. We'll call this one sub. So, just type **mkdir command_exp/sub**, and press ENTER. You can now go take a peek and see if the subfolder appears within the command_exp folder, if you like.

$ mv

The next command is the mv (move) command, but before we experiment with it, we need to create a dummy file—we need something to move, after all. We can do this by using another command—touch. To make the file, and let's call it expfile.txt, go to the Terminal, type **touch expfile.txt**, and press ENTER. The new file will now appear in your home folder.

To move the file that you've just created, you will use the mv command, of course. Just type **mv expfile.txt command_exp/sub** (this tells the system which file to move and where to move it to), and press ENTER. The file will now be in your sub folder.

$ cd

Until now, you have been using the command line from your home folder. With the cd command, you can change your Terminal's location to another folder. This is a very handy command that you will be using quite a lot when doing the other projects in this book. To take it out for a spin, let's get inside the command_exp folder by typing **cd command_exp** and pressing ENTER. If you've done this correctly, the prompt in your Terminal should now read *username*@*computer_name*:~/command_exp$. If so, you can pat yourself on the back.

While you are there, you might as well try out the ls command with the -R (recursive) flag to see how that works. Just type **ls -R**, and press ENTER. Your Terminal should now show that you have a subfolder there called sub and a file inside that subfolder called expfile.txt.

That is all you really want to do in there for now, so to get back to your home directory, just type **cd**, and press ENTER, which will take you back home, so to speak.

For future reference, it is worth noting a couple of other cd command shortcuts. If you are within a subfolder of a subfolder and want to move back a step, so to speak (from /home/rg/peas/pudding to /home/rg/peas, for example), you can do so by typing **cd ..** (with a space between cd and ..) and pressing ENTER. You can also type **cd -** (with a space between cd and -) in order to get back to a directory where you were previously (from /home/rg to /home/rg/peas/pudding, for example).

$ cp

Being fickle, as humans are by nature, you might decide that you not only want your expfile.txt file in the sub folder, but that you also want a copy in your home directory, where it was in the first place. To copy expfile.txt, you can use the cp (copy) command.

To do this, the command needs to know where the file you want to copy is, what it is called, and where you want to copy it, which in this case is to your home folder. Normally you would type cp command_exp/sub/expfile .txt /home/*username* to do this, but if you recall my mention of it near the

beginning of this chapter, you can abbreviate the /home/*username* portion of the command string to ~/, which means the same thing, and is an important tip to remember, as the tilde is frequently used in online instructions. As reducing wear and tear on the fingers is always a desirable goal, type the following command, and then press ENTER:

```
cp command_exp/sub/expfile.txt ~/
```

Be sure to put a space between the file you are copying and its destination (in this case, between the expfile.txt and ~/).

Once you've done this, you should have two copies of expfile.txt, one in your home folder and one in your sub folder. Go have a look to see the fruit of your endeavors.

$ rm

When you were a kid, you may well have experienced the joy of building a castle out of LEGO bricks and then the even greater joy of tearing the whole thing down (preferably by hurling D cell batteries at it). We will now embark on a similar move. The first tool in this nostalgic endeavor is the rm (remove) command, with which we can trash files.

The rm command, albeit very useful and easy to use, should be used with caution. Once you remove a file with this command, there is no going back—the file will not be placed in the Trash; it is gone for good.

To play it safe, let's try out the rm command by getting rid of that new copy of expfile.txt that we just created in the home folder. The basic rm command structure consists of the command itself, rm, followed by the name of the file that you wish to remove. In this case, you want to remove the file called expfile.txt located in your home folder. Assuming your Terminal shows you to be home, remove the file by typing **rm expfile.txt** followed by a tap on the ol' ENTER key. The file will then be gone, and gone for good.

Now, double your pleasure by getting rid of the version of expfile.txt that is located in the subfolder sub. In this case, you need to specify where the file is because it isn't in the folder that the Terminal is in. Just type **rm command_exp/sub/expfile.txt**, and then press ENTER. Oooh, very cool. Brings ya back, doesn't it?

$ rmdir

You will now continue the fun with the rmdir (remove directory) command, which is a bigger and more powerful version of the rm command.

The rmdir command, like the rm command, should be used with caution. There are no do-overs with rmdir. Once you remove a directory or folder with this command, it is gone for good.

To try this command, you can get rid of that sub folder you created. Type **rmdir command_exp/sub**, and press ENTER. The sub folder should now be gone. Finally, to round out the fun, use the rmdir command once more to get rid of the command_exp folder that we created earlier. You've probably got it down by now, but just in case you haven't, type **rmdir command_exp**, and then press ENTER.

$ chmod

In Chapter 6, you learned how to change file permissions via the Nautilus interface. This is without a doubt the easiest way to go about such things, but when you have a folder full of files, perhaps copied to your hard disk from CD, that are write protected, it can be quite tiring to change the permissions of such files one by one. In this case, the command-line approach proves to be much easier to deal with.

The command for changing file permissions is chmod (change mode). To use it, just type the command followed by the permissions you want to extend to a file, and then the location of the file itself. For example, let's say that you copied a JPEG file, mybirthday.jpg, from a CD to the personal subfolder within the Photos folder on your hard disk, and the file is write protected. To change the file so that you have write permissions (meaning that you can alter the file), you would type the following and then press ENTER:

```
chmod 744 ~/photos/personal/mybirthday.jpg
```

To change the permissions of all the files and subfolders (and all the files within those subfolders) in one fell swoop, you can add the -R (recursive) flag to the chmod command. The command would thus be as follows:
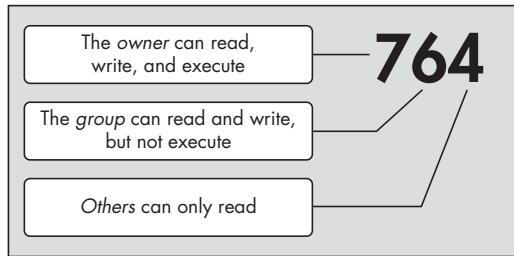
```
chmod -R 744 ~/photos/personal
```

The number *744*, by the way, extends read, write, and execute (run) permissions to you, the owner, but gives read-only rights to everyone else— a pretty safe choice when in doubt. If you want to figure out permission numbers for yourself, it is pretty easy. You are basically dealing with three number positions, each of which has eight numerical possibilities (0–7). The left slot represents permissions for the owner; the center slot represents permissions for the group; and the third slot represents permissions for others. The meanings of the numbers themselves are as follows:

- **7** Read, write, and execute permissions
- **6** Read and write permissions
- **5** Read and execute permissions
- **4** Read-only permissions
- **3** Write and execute permissions
- **2** Write-only permissions
- **1** Execute-only permissions
- **0** No permissions

Figure 8-8 points out the meaning of each of these numbers and what each number slot represents. In fact, if you don't mind a bit of simple addition, things are even easier to understand. To start out with, remember that 1 = execute, 2 = write, and 4 = read. Add any of those numbers together, and you get the other permissions combos. For example, 1 (execute) + 4 (read) = 5 (read and execute). As you can see, permissions aren't all that complicated.

Figure 8-8: The meaning of permissions numbers

Now if you're more of a letters than numbers sort of person, you'll be happy to know that there is another way to change permissions that is probably even easier. In this approach, you only have to deal with two groups of letters and the symbols + and −.

The first group consists of the following:

u   User (owner of the file)

g   Group (specified group of users)

o   Others (anyone who is not the user or a member of the group)

a   All (all of the above)

The second group consists of:

r   Read

w   Write

x   Execute

You might already be able to figure out how this is all going to work, but I'll spell it out just in case your intuition is worn out for the day. Let's say you want to change the permissions of a file (butterhaters.txt, for example) so that all users on your machine can read and write to it. After opening a Terminal window, you can make the change by typing `chmod a+rw butterhaters.txt` and pressing ENTER.

Oops! Just remembered that you don't want anyone changing the content of the file, eh? Well, to take back the write permissions for that file, you just need to type `chmod a-w butterhaters.txt` and then press ENTER. As you can see, the + gives permissions, while the - taketh away.

Much simpler, you've got to admit.

$ apt-get

To wrap up this section, let's finish with a command that might seem a bit familiar to you: `apt-get`. Yes, this command is indeed the key to controlling the powerful package download and installation tool, APT, which I covered in Chapter 5. Although it might not be as pleasing to use APT via the command line as it is via Synaptic or Add/Remove Programs, it can come in handy. I'll cover the basics for you here.

Before using the `apt-get` command, be sure to close any APT front ends you might have open, such as Synaptic. As I mentioned before, you can only run one APT tool at a time. Once the coast is clear, you should always start out any operations involving `apt-get` with an update of the APT database so that you will be downloading the newest stuff. To do this, just type the command `sudo apt-get update`.

If you want to install a single package without heading over to Synaptic, you can do so by typing `sudo apt-get` *package-name*. For example, if you want to download and install the Shufflepuck game clone, Tuxpuck, you would type `sudo apt-get tuxpuck`. If you eventually get annoyed with Tuxpuck after having lost one too many times, you can uninstall it by typing `sudo apt-get remove tuxpuck`.

Finally, bearing in mind all the warnings offered in Chapter 5, if you want to upgrade your entire system via the command line, you can do so by typing `sudo apt-get dist-upgrade` (but only after doing a `sudo apt-get update` first).

A Couple of Other Biters You'll Be Using Soon

This is as good a place as any to introduce two more commands that you will be called upon to use in this chapter and elsewhere in the book: `ln` and `tar`. You needn't practice with these yet, as you will be using them very soon, but you might as well know what they are all about.

$ ln

The `ln` (link) command is used to create a link file that launches or activates another file located in a separate folder. This is very useful when trying to activate a file that is buried deep in the subfolder of a subfolder of a subfolder somewhere on your hard disk. The command is very often used with the `-s` (symbolic) flag, which provides essentially the same thing as the shortcut you've come to know in Windows, or the alias on the Mac.

The easiest way to use the `ln` command is to first use the `cd` command to change the Terminal's location to the folder where you want to place the link. Then you can type the `ln` command on the command line, followed by the path of the file you wish to link to. For example, let's say that you want to put a link in your home folder for an OpenOffice.org Writer file of your autobiography called myLife.sxw. The file is pretty well buried in a nest of subfolders deep within your home folder: /home/*username*/Documentia/personal/self/autobiography/myLife.sxw. To create the link, you would open a new Terminal window, type the following command string, and then press ENTER:

```
ln -s Documentia/personal/self/autobiography/myLife.sxw
```

Once you are finished, the link will appear in your home folder as an icon matching the original file in appearance, albeit sporting an arrow to signify that it is a link.

$ tar

In Chapter 6 you learned to create and extract archives, or tarballs, but did you know that you can also create and extract tarballs using the command line? The tar command is your key to doing this.

To create an archive, you would simply type tar -cvf, followed by the name the final tarball will be, and then the name of the folder or file you are trying to archive. For example, let's say that you want to create an archive of your photos folder, and you want to call it pics4pals. In this case, you would type the following command, and then press ENTER:

```
tar -cvf pics4pals.tar photos
```

As you no doubt noticed, there are some flags after the tar command in that string. The *c* tells the tar program to *create* a new archive. The *v* tells the program to be *verbose*, or, in other words, to tell you what it is doing in the Terminal as it is doing it. Finally, the *f* tells the program that what follows is the *file information*.

If, after creating the archive, you suddenly remember that there is one more file you want to add to the mix, you can use the -r flag to append the archive. For example, to add a file called cranky.png to the archive, you would type the following and then press ENTER:

```
tar -rvf pics4pals.tar cranky.png
```

Of course, chances are that you will be doing more tarball extracting than creating, so you no doubt want to know how to do that. Fortunately, the process is pretty similar to what you use when creating the tarball. The main difference is in the first flag. Rather than using the tar command with the -c flag, you would instead use it with the -x flag, which tells the tar program to *extract* the specified archive. So if you want to extract a tarball called spicyfood.tar, type the following command, and press ENTER:

```
tar -xvf spicyfood.tar
```

What you have been doing thus far is creating and extracting archives, which are basically just collections of files. They are not, however, compressed. In fact, most tarballs you find will be compressed, and you can tell by the ending *tar.gz*. That *gz* means that the archive was compressed using the gzip program. Extracting a compressed tarball is just as easy as extracting a straight tar archive; all you have to do is add the -z flag, which tells your system to use the gzip program to decompress the archive. For example, if you want to extract a compressed tarball called goosedown.tar.gz, type the following command, and press ENTER:

```
tar -xzvf goosedown.tar.gz
```

Well, now that you know how to decompress and extract a gzipped tarball, you probably want to know how to create one. This is, again, little different than creating the tar archive itself; you would just add the -z tag to tell the program to use gzip to compress the folder. For example, to create a compressed version of your Documentia folder, which we'll call tightdocs.tar.gz, for example, you would type the following, and press ENTER:

```
tar -czvf tightdocs.tar.gz Documentia
```

It's worth mentioning at this point that you may also come across some files compressed with the bzip program. Such files are recognizable by some variation on the .bz file extension. Dealing with these files should pose no problem, as the commands are almost identical to those for gzip. Just substitute -j for -z in the command string.

Compressing and Extracting Compressed Single Files

If you want to compress or decompress a single file, there is no real need to use the tar program at all, since its purpose is to create archives consisting of several files. You can instead use the gzip and gunzip commands directly. For example, to compress a file called matilda.png, you would type gzip matilda.png, and press ENTER. The matilda.png file would then become matilda.jpg.gz. To decompress the file, you would type gunzip matilda.jpg.gz, and press ENTER, after which the matilda.jpg file would be back to normal.

So can you compress an archive you've already created with the tar command? Sure. For example to compress the spicyfood.tar archive we mentioned before, you would type gzip spicyfood.tar, and voilà—you've got yourself a compressed spicyfood.tar.gz archive. Pretty cool, don't you think?

Project 8A: Creating a Plan

Well, now that you have a bit of command experience, it's time to get some practice and put all those commands to good use. In this project we start off easy by creating a .plan file, like I mentioned in the section "$ finger" on page 129. The actual .plan file is a hidden file (as you can see by the period before its name), which contains the plan or message that you add to that file. That message will appear in the output of someone's Terminal when they use the finger command to find out more about you. You may not need such a .plan file, but it is an easy enough way to get started working a bit more with commands and the Terminal itself, so let's give it a try.

To start out, we are going to open the Terminal-based Pico editor to create the .plan file. To do this, open a Terminal window, type **pico .plan** (being sure to put a space between pico and .plan), and press ENTER. Your Terminal should now look a bit different, as you can see in Figure 8-9.
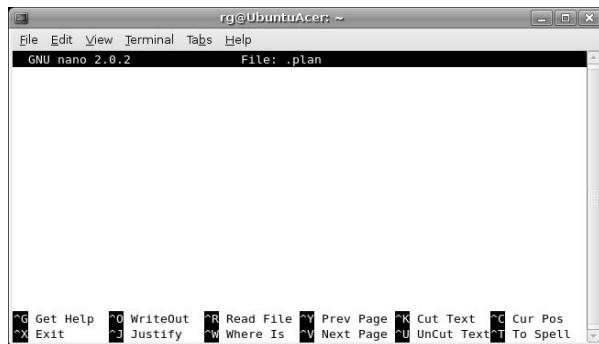
Figure 8-9: The Pico editor

You will now be looking at your new, and totally empty, .plan file within the Pico editor. All you have to do now is type your plan or message. Once you've done that, press CTRL-X to exit the Pico editor, and it will ask you if you want to save your work. You do, so type **Y**, after which you will be presented with a set of save options. You have already named the file .plan, as you can see near the bottom of the screen, so all you have to do is press ENTER. You will be back at your now-familiar user prompt in the Terminal window.

To wrap things up, you want to change the permission of the new .plan file by typing **chmod 644 .plan** in the Terminal window and then pressing ENTER. The .plan file should now be in your home folder and visible by all, so go on and test your work by typing **finger *username*** and pressing ENTER. The message you entered in your .plan file should now appear in the results in place of the no Plan you found there earlier. If you want to change the contents of your .plan file later on, just follow the same steps, and change the text when the .plan appears in the Pico editor.

NOTE   *The name of the .plan file is preceded by a period, which means that it is a hidden file. Thus, if you take a look in your home folder, you will not be able to see the file unless you have checked the box next to the words* Show hidden and backup files *in the Nautilus Preferences window.*

Project 8B: More Command Practice with pyWings

Now let's get some more experience behind the Terminal by installing a simple, and admittedly kind of silly, oracle program called pyWings (see Figure 8-10). pyWings will give you cryptic guidance in response to whatever questions you may ask it.

Figure 8-10: Seeking wisdom from pyWings

To use pyWings, you type whatever your confusion is in the input box, click one of the concern icons on the left (self, another, world), one of the realm icons on the right (love, work, truth), and hit the big button that looks like half an eye. The oracle will then tell you what it has to say. As an example, I asked the oracle why I haven't become rich yet, and I picked *self* as my concern and *truth* as the realm. Figure 8-11 shows the wisdom that was bestowed upon me.
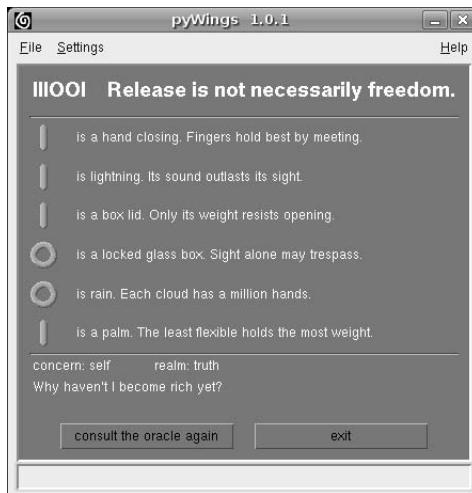
Figure 8-11: PyWings bestows its wisdom

As you can see, the oracle told me, "Release is not necessarily freedom," which I will interpret as one of those "dreams come true, not free" sort of pieces of advice.

Great.

pyWings was written in a programming language called Python, which actually creates scripts rather than true conventional programs. You will learn a little more about this distinction in Chapter 9, but one of the differences I can

mention right off the bat is that you don't actually have to install pyWings; you are simply going to put it on your hard drive in your home folder and run it from there, more or less as is.

8B-1: Getting pyWings

You are just about ready to begin "installing" pyWings, but before you do, you've got to get it from the project's home page at http://pywings .sourceforge.net/download.html. Just scroll down the page to the section below the words *Unix/Linux/X11*, and click the **HTTP** link, which will get you the file pywings-1.0.1.tar.gz (*.tar.gz* tells you that this is a *tarball*—the Linux world's answer to Zip files). Be sure to place the file in your home folder so that you can follow along easily.

8B-2: Creating a LocalApps Folder for pyWings

As I mentioned before, you will be installing the pyWings program locally in your home folder. Installing a program *locally* means that you are installing the program and all its support and data files in your home folder. This makes things a bit easier, but it also means that the program will not be available to other users. It also means that, if you're not careful, you might inadvertently delete it.

To make things a bit easier and safer for you, you are going to create a folder in your home folder in which to place pyWings and all other applications that you install locally on your machine in the future. You will, logically enough, call the folder LocalApps.

Let's make the folder by command, in order to get some more practice. Go to the Terminal, make sure you are in your home folder, type the following command, and then press ENTER:

```
mkdir LocalApps
```

8B-3: Extracting the pyWings Tarball

Now it is time to extract the tarball. You can do this by the double-click method you learned in Chapter 6, but since we're working with the command line, let's use that instead.

To start out, we're going to place the tarball in the same folder into which we extract its files. Usually this isn't necessary because the contents of most tarballs are already packaged in a folder of their own. By double-clicking the tarball to open it up in File Roller, you can see whether things are packed in a folder or simply as a group of files. In the case of pyWings, it is the latter, so follow these steps:

1. Create a pywings folder by typing `mkdir pywings` and pressing ENTER.
2. Move the pyWings tarball into that folder by typing `mv pywings*.gz pywings` and pressing ENTER.

You can see that in this step we used an asterisk (*) to save some wear and tear on our fingers. The asterisk is a wildcard character that in this case told your system to move any file beginning in *pywings* and ending in *.gz*. Fortunately we had only one item matching those criteria.

3. Move over to the new pywings folder by typing **cd pywings** and pressing ENTER.

4. Now we get down to the process of extracting the tarball itself using the tar command. To do this, type **tar -xzvf pywings*.gz**, and press ENTER.

   Again, notice that we used the asterisk to save ourselves some keystrokes, though we could just as well have typed **-1.0.1.tar** in its stead.

5. Finally, type **cd**, and press ENTER to bring the Terminal back to your home folder.

8B-4: Moving the pyWings Folder to Your LocalApps Folder

The extraction process is now complete. Before going on to running pyWings, however, let's move it to the new LocalApps folder you created in Project 8B-2 on page 143. To do this, type the following command, and then press ENTER:

```
mv pywings LocalApps
```

8B-5: Running pyWings

Now that you have pyWings in place and ready for action, let's start up the great oracle right now so that you can get a better perspective on how to deal with the aspects of life that trouble you.

In the Terminal, make sure you are in your home directory, type **LocalApps/pywings/pywings.py**, and press ENTER.

If all went according to plan, pyWings should be up and running and will soon be making you a wiser person.

8B-6: Creating a Launchable Link for pyWings

The method of running pyWings that you've just used works well enough, but it is a pain in the posterior to open your Terminal and type that somewhat lengthy string every time you want to find out what fate has in store for you. Let's find a way to make things easier in the future.

To run an application from the Terminal, you generally type the name of that application or, to put it more precisely, the name of that program's executable file; the application's name thus acts as a sort of command. In order for your system to recognize that command, however, the command (the executable file, or a link to it) must be in a location where the system can find it. Whenever you run a command of any sort, your system checks a series of locations (most of which are *bin folders*, where executable files are located) to find that command.

You can easily find out where these locations are by typing `echo $PATH` in a new Terminal window and then pressing ENTER. As you will see, on your Ubuntu system, these locations are:

- `/usr/local/bin`
- `/usr/local/sbin`
- `/sbin`
- `/usr/sbin`
- `/bin`
- `/usr/bin`
- `/usr/bin/X11`
- `/usr/games`

If the command you typed is in one of those locations, the program, or script, will run. As you no doubt know, however, pyWings is not in any of those locations. It is in /home/*username*/LocalApps/pyWings and is thus, in a sense, out of your system's sight.

To remedy this situation, you could add the path of your pyWings script to the list of paths that the system checks for run commands, so as to make the system aware of your new application's existence. However, let's try another method that I think is easier. What you will do is create a link to pyWings, a sort of launchable alias, in one of the locations your system does check for commands.

To create this link, you will be using three commands: `cd` (to change directories), `sudo` (to give yourself write access to the destination folder), and `ln -s` (to create the link).

1. In the Terminal, type **cd /usr/games**, and press ENTER. This puts you in one of the folders your system searches when you enter commands.
2. Type **sudo ln -s /home/*username*/LocalApps/pyWings/pywings.py pywings**, and press ENTER. (Note that there is a space between the words `pywings.py` and `pywings` at the end of that command string.)

    The `pywings` at the end of that command string is the name that you are giving the link; the name of the link thus becomes the command you will use to run the application. If you type nothing, the link will be called pywings.py, which would mean three more keystrokes for you every time you wanted to start the program.
3. Type your password when prompted to do so, and then press ENTER.
4. Type **cd**, and press ENTER to return the Terminal to your home folder.

8B-7: Running pyWings Again

Now that you have created the link, you should be able to run the pyWings program much more easily. To try it out, quit pyWings (if it is still running), type **pywings** in the Terminal window, and press ENTER. Your personal pyWings oracle should appear again.

You've managed to cut down on the number of keystrokes required to run pyWings from the Terminal. However, if you are really into this pyWings thing and want to use it often, it will probably be handiest to add a launcher to your panel, a drawer, or the Applications menu.

To add a pyWings panel launcher, for example, right-click any open space in the panel, and select **Add to Panel**. When the Add to Panel window appears, click the **Custom Application Launcher** button. In the Create Launcher window that appears, type `pyWings` in the Name section, anything you want in the Comment section (`Your Obtuse Guru`, for example), and, assuming you created a launchable link in Project 8B-6 on page 144, type `pywings` in the Comment section. For an icon, click the **No Icon** button, and then look around until you find an icon that suits your fancy. I like gnome-eog.png myself. Once you've made your selection, click **OK** in the Browse Icons window, and then click **OK** in the Create Launcher window.

8B-8: Adding Emblems to Your LocalApps Folder

Now that pyWings is successfully installed and working, it is probably a good idea to add an emblem to your new LocalApps folder so that you don't inadvertently dump it in the Trash someday. You have already learned how to do this in Chapter 7, so I won't give you the step-by-step instructions.

Project 8C: Command Practice Review with Briscola

If you would like to reinforce the skills you've put to use in the previous project, why not go a bit Continental, and try out Briscola—a simple, yet very traditional, Italian card game (see Figure 8-12). Unlike pyWings, which is a Python script, Briscola is a script of a different flavor, written in a scripting language called Tcl, which uses something called Tk to create its graphical interface.

Figure 8-12: Briscola

8C-1: Getting Briscola

You are just about ready to begin "installing" Briscola, but before you do, you must get it. You can get Briscola by going to the project's home page at www.rigacci.org/comp/software and downloading in the traditional manner, but since we're working with commands, let's instead get Briscola by using a new command: `wget`.

To do this, just open a Terminal window, type the following command string, and then press ENTER:

```
wget http://www.rigacci.org/comp/software/briscola/briscola-4.1.tar.gz
```

In your Terminal window you will see `wget` in action as it connects to the site where Briscola is stored and then downloads the file. When it's done, you will find the Briscola tarball in your home folder.

8C-2: Extracting the Briscola Tarball and Renaming the Briscola Folder

Extracting the Briscola tarball is essentially the same process as that for pyWings; however, Briscola is already packaged within its own folder, so you won't have to create a special folder for it.

Although I am sure you now know the drill, I'll tell you again. Just open a Terminal window, type the following command, and press ENTER:

```
tar -xzvf briscola*.gz
```

A new folder, Briscola-4.1, will now appear in your home folder with all of the Briscola files in it. To make things easier to deal with in the future, let's shorten the name of the folder to simply *briscola*. We already know how to do this via the right-click method, but this time around let's to do it via the command line. To do this, you use, perhaps surprisingly, the `mv` command followed by the name of the file whose name you are going to change, followed by the new name of the file.

Give it a go by typing the following command and pressing ENTER:

```
mv briscola-4.1 briscola
```

8C-3: Preparing the Briscola Script

Most applications that come in tarball form include a README file, which includes information on what you need to do in order to install and use the application. If you double-click the **README** file in the briscola folder in Nautilus, you will see that the "HOW TO START" section tells you to adjust the first line of the briscola.tk script to point it to your Tk shell, and to adjust the second line of the script to point to the directory where the various Briscola files are located.

To perform the adjustments as instructed in the README file, just follow these steps:

1. Find the Tk shell, called Wish, by typing `locate wish` in the Terminal window and pressing ENTER, and then note the location given on a piece of paper. It should be `/usr/bin/wish`.
2. Direct the Terminal to the briscola folder by typing `cd briscola` and pressing ENTER.
3. Use the Pico editor, which we used in Project 8A on page 140, to edit the briscola.tk file by typing `pico briscola.tk` and pressing ENTER. The briscola.tk file will appear in the Pico editor in your Terminal window.
4. Change the very first line of the briscola.tk file from `#!/usr/local/bin/wish` to `#!/usr/bin/wish`.
5. In the second line, change /usr/local/games/briscola to `/usr/share/games/briscola`, which is where you will place Briscola in just a bit.
6. Press CTRL-X on your keyboard.
7. Type `Y`, and press ENTER to save your changes.
8. Type `cd`, and press ENTER to return the Terminal to your home folder.

8C-4: Moving the Briscola Folder to a Global Location

We could move the briscola folder to the LocalApps folder and play it from there, as we did with pyWings, but this time around, let's do things a bit differently by moving the whole thing to global territory. This not only keeps it safe from our obsessive housekeeping tendencies, but also allows all users on the same computer to play the game. We will need to use the sudo command to do this so that we can have write access in those protected folders.

To do this, just type the following command in the Terminal window, and press ENTER:

```
sudo mv briscola /usr/share/games
```

When you are prompted for your password, type it, and press ENTER.

8C-5: Creating a Launchable Link for Briscola

Even though we've moved Briscola to a global location, we still can't run it with a simple one-word command because the briscola.tk file itself is not in the system's command search path. Just as we did for pyWings, we will now create a launchable link for Briscola to solve that problem. Here are the steps:

1. In the Terminal, type `cd /usr/games`, and press ENTER.
2. Now create the link by typing `sudo ln -s /usr/share/games/briscola/briscola.tk briscola`, and press ENTER.
3. Type `cd`, and press ENTER to return the Terminal to your home folder.

You can now easily run Briscola by typing `briscola` in the Terminal and pressing ENTER.

Can I Do the Same Thing with pyWings?

Sure. If you want to move pyWings to a global location, just follow the same procedure for moving the pywings folder and creating the link as you did for Briscola, making the necessary substitutions, of course. You will have to remove the previously created pyWings link, though, by typing the following command and pressing ENTER:

```
sudo rm /usr/games/pywings
```

After that, move the pywings folder to global territory by typing **sudo mv ~/LocalApps/pywings /usr/share/games/pywings**, and pressing ENTER. When prompted for your password, type it, and press ENTER. You can then create the launchable link by typing **cd /usr/games**, pressing ENTER, typing **sudo ln -s /usr/share/games/pywings/pywings.py pywings**, and pressing ENTER once more.

Playing Briscola

As I already mentioned, Briscola is easy—about as easy a card game as there is. It is a trick-taking game, which means that you put out a card, then your opponent puts out a card, and the one who puts out the higher point-value card wins the hand, or *trick*. Points are awarded on the basis of the cards involved in that trick. The winner of the trick then goes on to *lead* the next trick, meaning that the winner puts out his or her card first the next time around. When all the cards are played, the points for each player are then tallied, and the player with the higher points wins. All much simpler to do than it is to describe.

The Cards and Their Rankings

Like many other Italian and Spanish games, Briscola is played with a 40-card deck, consisting of the following cards: K, Q, J, 7, 6, 5, 4, 3, 2, and the Ace. Traditionally, it is played with either French-suited cards (hearts, diamonds, clubs, and spades) or Italian-suited cards (swords, coins, batons, and chalices), usually dependent upon the region in which it is played. In the software version you have just installed, you will be playing with French-suited cards in the regional pattern of Tuscany.

Unlike most card games you are probably familiar with, the ranking and point values of the cards in Briscola is somewhat different, as you can see in the following chart.

Ranking of Cards	Point Value
Ace	11
3	10
King	4
Queen	3
Jack	2
7, 6, 5, 4, 2	0

While this ranking arrangement might seem odd, it is actually fairly common in card games from the southern and Catholic regions of Central Europe. With that bit of information in mind, it should all be pretty easy enough to fathom, taking a religious view, that God (Ace) and the Holy Trinity (3) rank higher than the quasi-mortal royals (K, Q, J) and their decidedly mortal subjects (7, 6, 5, 4, 2). The rankings are thus quite sensible, albeit slightly ironic, given that some religions frown upon playing cards, calling them "the devil's tool."

Game Play in Briscola

Once Briscola starts up, it will deal three cards to each player. It will then take the seventh card and place it face up under the downward-facing pile of undealt cards, known as the *stock* or *talon*. That seventh card is called the *Briscola* (from which the game gets its name), and it determines the *trump suit* for that particular game. This means that any card of the same suit as the Briscola will beat any card of any other suit, even one of a higher ranking. Of course, when you play a trump card against another trump card, the normal rankings of the cards come back into play.

It is important to note at this point that unlike many other trick-taking games, you are neither required to follow the suit of the card led in a trick in Briscola, nor are you required to beat it if you can. This means that if your opponent plays a club card, you can play a card of any suit you like, even if you have a card that can beat it, all depending on your own strategy for ultimately winning the game.

Before getting started, it is a good idea to first go to the **Options** menu and select **Show Score**. This will allow you to know how you're doing as you play. Once you've done that, you are ready for action, and as your computer opponent is always kind enough as to allow you to lead, you can begin by clicking the card you want to put into play. Your compu-opponent will then play its card.

Once you've assessed the situation, click on one of the blank spaces in your hand, and the points for that trick will be displayed in the box labeled *You* (if you won the trick) or the box labeled *Me* (if the computer won the trick). Of course, if the trick only involved the 7, 6, 5, 4, or 2 cards, no points will appear, as those cards have no point value.

Want to Know More?

If you would like a more detailed set of rules for playing Briscola (and just about any other card game in the world), check out www.pagat.com.

Customizing the Terminal

As you now well know, the Terminal is a very simple application in terms of looks. It doesn't have to be, however, because you can spice things up a bit if you're so inclined. Not only can you change the background and text colors in the Terminal, but you can even display one of your favorite photos as a background (as shown in Figure 8-13) or make the background transparent . . . well, kind of transparent anyway.
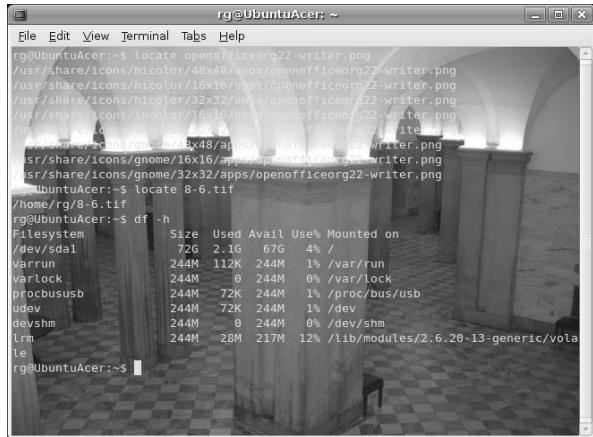
Figure 8-13: A Terminal window with customized background and font colors

To change the Terminal background, go to the Terminal **Edit** menu, and select **Current Profile**. When the Editing Profile window appears, click the **Effects** tab, select **Background image**, and then click the **Browse** button to navigate your way to the image you want to use (Figure 8-14). Depending on the image you use for your background, you may find it rather difficult to see the text once your image appears in the Terminal. If so, try moving the slider under the words *Shade transparent or image background* in the Editing Profile window. If that still doesn't do the trick, click the **Colors** tab, deselect **Use colors from system theme**, and then try some of the preset Foreground and Background combinations from the menu button next to the words *Built-in schemes* (I used White on Black in Figure 8-13).
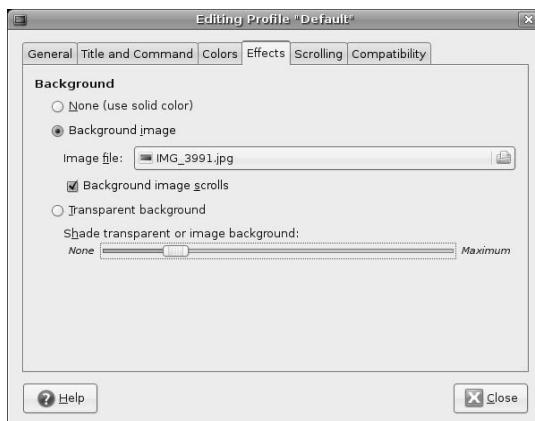
Figure 8-14: Customizing the Terminal window

If you just want to make the background transparent, select **Transparent background**, and drag the slider near the bottom of the Editing Profile window to the far left. You can also use the slider to adjust the shading of your background image if you choose to go that route.

The transparency effect used in the Terminal is a pseudo-transparency, as it is really just a mirroring of the desktop image. You will thus find, if you have any icons on your desktop, that when you pass the transparent terminal across such icon-laden areas, those icons will not appear in the seemingly transparent Terminal window—all you will see is the desktop wallpaper or background color.

Depending on the colors present in your background image or in your desktop wallpaper (if you've gone the transparent route), you may also want to change the font color for your Terminal to make things easier to see. To do this, click the **Colors** tab, deselect **Use colors from system theme**, and then make the appropriate font color selection.

Tabbed Shell Sessions in the Terminal

To wrap things up in this chapter, I thought I might mention one particularly convenient feature of the GNOME Terminal: tabs. Just as you can view multiple web pages in one Firefox web browser window through the use of tabs, tabs in the Terminal application allow you to have more than one shell session running at the same time without having more Terminal windows open (see Figure 8-15). This reduces the amount of desktop clutter and generally makes things easier to deal with. You can open a new tab within the Terminal by going to the Terminal **File** menu and selecting **Open Tab ▸ Default**.
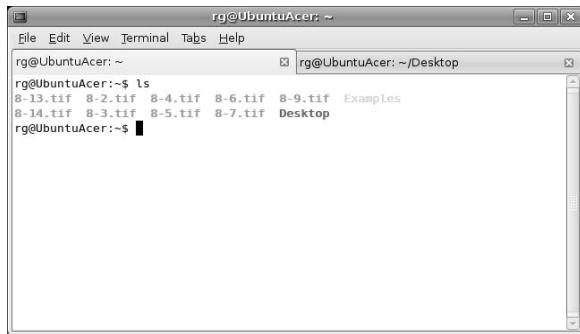
Figure 8-15: Running multiple shell sessions in tabs within the GNOME Terminal

9

AUTOMATIX AND NOT SO AUTOMATIX

More Ways to Install Programs

In the past few chapters, you learned how to install applications via Synaptic and run script-type applications from the command line, which gives you quite a variety of applications to choose from. It is now time, however, to expand your application-gathering repertoire even further.

In this chapter you will learn how you can add applications to your system in even more ways, a couple that involve working with APT, and a couple that don't. In fact, you will be dealing with one of the newest ways of making your Ubuntu system as functionally well rounded as any other system out there, and yet you will still be using one of the most manual and traditional ways to install an application, that is, compiling it from source. You will also learn how to convert Red Hat Package Manager (RPM) packages for use in your Debian-based system environment, running Java-based applications, and—say it ain't so— even running some Windows applications. When you are done, you will have more options at your disposal than you'll know what to do with . . . and that's not a bad position to be in.

As you may recall, the applications you install from the various Ubuntu repositories via Synaptic or Add/Remove Programs are in the form of DEB packages—and the repositories are filled with such packages. Unfortunately, although these repositories seem to contain just about every piece of software out there, for different reasons not every package finds its way into these repositories, even if it is available in DEB form. At the present time, the application Automatix is one of these.

Automatix is, interestingly enough, yet another front end for the powerful APT package-management software that is at the structural heart of all Debian-based systems. Like Synaptic, Automatix provides you with a graphical means to select, download, and install packages. What is different, however, is that Automatix provides a fairly automatic way for you to fill in all the holes left in your distro due to licensing and copyright concerns. In case you don't get what I'm talking about, I am referring to MP3 playback and encoding, unencrypting DVD playback, and allowing you to write to NTFS partitions on your hard drive. Sure, you can accomplish all this without Automatix, but using it makes the process a no-brainer.

9A-1: Getting and Installing Automatix
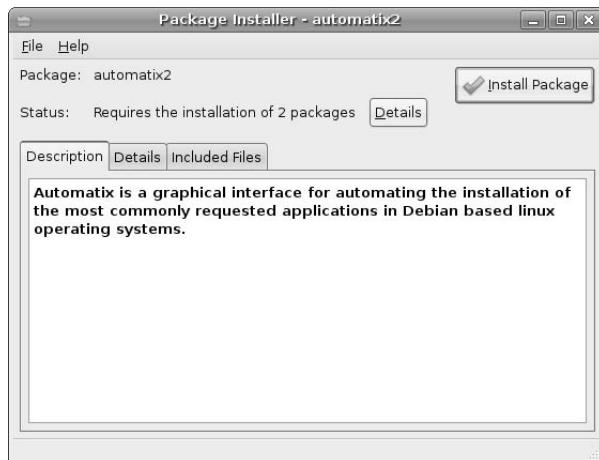
Getting and installing Automatix is a very simple process. To get started, go to www.getautomatix.com, and then click the **Installation** button. On the next page, click the **2.1 Ubuntu 7.04 (Feisty 386)** link near the top of the page, and then click the URL below either *Ubuntu 7.04 (Feisty 386)* or *Ubuntu 7.04 (Feisty amd64)*, depending on the type of processor you have in your machine.

At this point Firefox will bring up a window asking you what you want to do with the Automatix DEB package; either option will do. For the sake of experience, however, let's accept the default, *Open with GDebi Package Installer*, by just clicking **OK**. Within a moment or two, GDebi, yet another APT front end, will appear with information about the package you are about to install (Figure 9-1). In that window, click the **Install Package** button, after which you will be prompted for your password. Once the password formalities are complete, things should start looking rather Synaptic-esque, with the same orange progress bars zipping back and forth in progress windows until the process is complete. Then, just click the **Close** button in the progress window, and close the GDebi window. Mission accomplished.

9A-2: Running Automatix
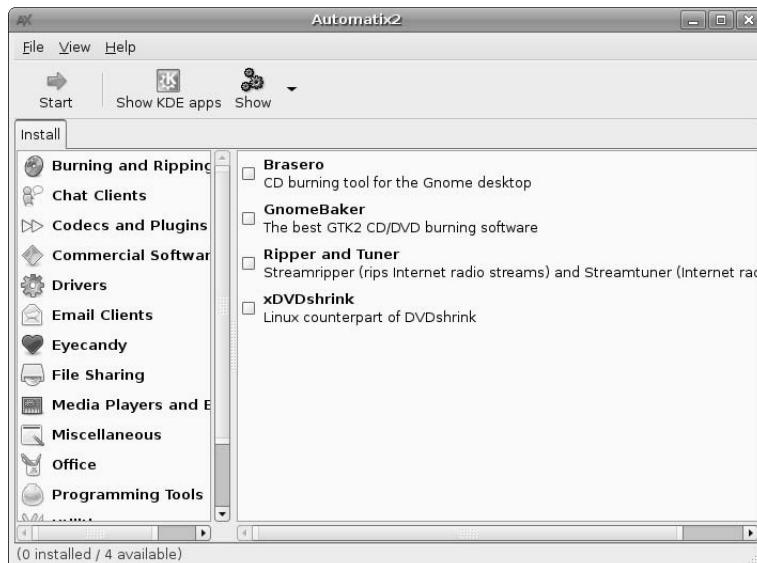
Once Automatix is installed, you can run it by going to the **Applications** menu and selecting **System Tools ▸ Automatix**. You will then be prompted for your password, so type it and click **OK**. A window will appear telling you that installing certain things without paying a fee is a legal no-no, depending

on where you live. Read the message, remember what it says, and then click **Yes**. A couple of windows will appear and disappear as Automatix checks the available package lists and adds a few repositories of its own. Once it's done, the main Automatix2 window, shown in Figure 9-2, will appear. Automatix is ready for action.

Figure 9-1: Installing individual DEB packages via GDebi

As you can see, Automatix looks very similar to Add/Remove Programs. The main difference is that its package collection is more specific and, as I mentioned before, geared towards plugging the holes in your system. In fact, most of the packages available from Automatix are also available via Synaptic and Add/Remove Programs; Automatix just groups the most commonly desired ones together, thus making things a bit more convenient.

Figure 9-2: Plugging up system holes using Automatix

9A-3: Installing Software with Automatix: Skype

Now that you know what Automatix does, let's
give it a spin by installing a package that is
not in the usual Ubuntu respositories but is in
the new repositories added by Automatix:
Skype. Skype, shown in Figure 9-3, is Voice
over IP (VoIP) software that allows you to
speak to other Skype users over the Internet
with the clarity of a regular telephone line.
And it doesn't cost anything—even if you
call users overseas. There are also for-fee
services, such as SkypeOut, which allows you
to call regular mobile and landline telephone
numbers from your computer at a fraction
of what it would normally cost from a regular
telephone.

> While it is true that Ubuntu comes
bundled with a VoIP package of its own,
called Ekiga Softphone, it is not compatible
with the much better known and more wide-
spread Skype. Since it is very likely that the

*Figure 9-3: Internet telephony
made easy with Skype*

majority of people you know who are using a VoIP software package are
using Skype, it only makes sense to go the Skype route so that you can easily
communicate with them. It's nice software anyway.

Installing and Running Skype

To install Skype via Automatix, click **Chat Clients** in the left pane of the
Automatix window, check the box next to the word *Skype*, then click the
Start button. The Automatix window will suddenly metamorphose into a
half-Terminal window with a bunch of white text streaming down a black
background in the bottom half of the window. That stream of text will
then stop at a configuration screen.

> Once you're in that screen, press the TAB key to highlight the OK button
(in red), as shown in Figure 9-4, and then press ENTER. In the next screen,
select Yes by pressing the TAB key once and then pressing ENTER. After another
second or two of rolling white text, you will be back to the regular Automatix
window, and Skype will have been installed. You can go ahead and quit
Automatix.

> Once all the pieces have been installed, you can run Skype by going to
the **Applications** menu and selecting **Internet ▸ Skype**. After setting up a Skype
user account, I would advise you to restart your computer. Skype seems to have
better sound quality after an initial restart. Also remember, while we're on
the topic of sound quality, that it is best to use Skype with a headset. Trying
to talk with a stick microphone could cause feedback or echoes because the
microphone will pick up and transmit sounds from the speakers.

Figure 9-4: Installing Skype via Automatix

9A-4: Installing Other Packages with Automatix: NTFS Partition Write Support

If you have Windows installed on another partition and want to play around a bit more with Automatix right now, then I would suggest installing the package *Automatix read/write NTFS and FAT32 Mounter*, which you can find by clicking **Miscellaneous** in the left pane of the Automatix window.

This package allows your system to automatically mount your Windows partitions at startup, and even more importantly, it allows you to write data to an NTFS partition, should you happen to have one. Pretty useful if you're a dual booter (and pretty much useless if you're not).

The procedure for installing the package is almost the same one you used when installing Skype. Just check the box next to the words *Automatix read/write NTFS and FAT32 Mounter*, agree to do whatever you are asked to do along the way, and then once the installation is complete, restart your system. Pretty cool.

Project 9B: Converting RPMs to DEB Packages Using Alien

While it is true that not all DEB packages make their way into the Ubuntu repositories, it is also true that some, albeit few, applications are not available as DEB packages at all. Some, however, are available as RPMs, which are the packages used in Red Hat, Fedora, Mandriva, and a few other Linux distributions. While this may not sound particularly promising to you as an Ubuntu user, there are a few rays of hope emanating from a handy utility by the name of Alien.

If you haven't guessed already, what Alien does is convert RPM packages to DEB packages that you can install and run in your DEB-based system. It doesn't work perfectly 100 percent of the time, and it isn't as desirable an alternative as a properly compiled DEB-from-source package, but if there's an application you really need, it might just get the job done. To give Alien a try, we'll convert and install an application called TuxCards (shown in Figure 9-5) that calls itself a *hierarchical notebook.*
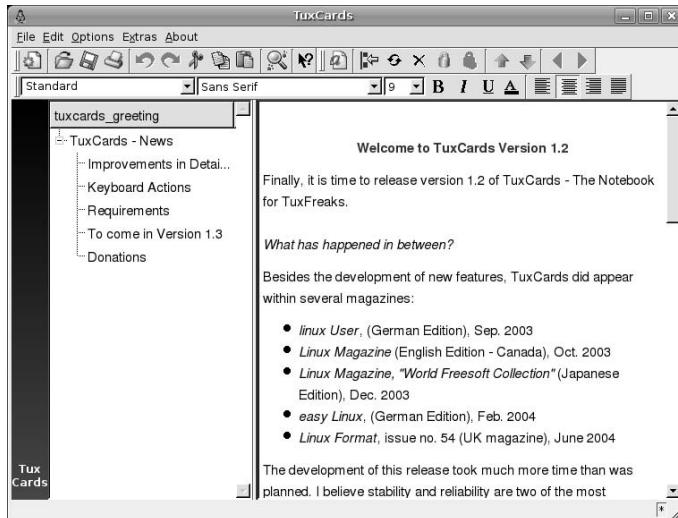
Figure 9-5: TuxCards—a hierachical notebook

9B-1: Installing Alien

Before downloading, converting, and installing the TuxCards RPM, you have to first install Alien. Fortunately, Alien is available via Synaptic, so all you have to do is perform a search for *alien*, and then follow the standard installation procedures. If you prefer, you can download and install it by using the `apt-get` command you learned about in the previous chapter. If you go this route, first type `sudo apt-get update`, and press ENTER. After the repository database update is complete, go ahead and install Alien by typing `sudo apt-get install alien` and pressing ENTER.

9B-2: Getting the TuxCards RPM

Once Alien is installed, you need to download the TuxCards RPM. You can get this from www.tuxcards.de/requirements.html. On that page, right-click **tuxcards-1.2.1mdk.i586.rpm**, and then select **Save Link As** in the popup menu. There are other versions on the page that might work, but for the sake of this project, stick with the one I just mentioned.

Once the download is complete, make sure to place the RPM in your home folder, if it isn't already there.

9B-3: Converting the TuxCards RPM

Now that the TuxCards RPM is in your home folder, you can use Alien to convert it to an Ubuntu-friendly DEB package by typing the following command and pressing ENTER:

```
sudo alien --scripts tuxcards*.rpm
```

After you type your password, nothing much will seem to happen. Once Alien is done doing its thing, however, you will get the message `tuxcards_1.2.2_i386.deb generated` (or something like that) in the Terminal window. If you look in your home folder, you will find the newly generated DEB package along with the original RPM.

9B-4: Installing the Newly Generated TuxCards DEB Package

You can now install the TuxCards DEB package by using the `dpkg` (Debian package) command with the `-i` (install) flag and then pressing ENTER. The command should be:

```
sudo dpkg -i tuxcards*.deb
```

If I haven't won you over to the world of commands (not that I'm trying to), and you still prefer going about things graphically whenever possible, you can install the TuxCards DEB package the same way you installed Skype in Project 9A—by double-clicking the DEB package and using the Package Installer window. It all works out the same in the end, so it's strictly up to you.

9B-5: Running TuxCards

Once TuxCards has been installed, you can run it via the Run Application applet. Just type `tuxcards` in the command box, and then click the **Run** button.

Compiling Programs from Source

For the beginner, just the mention of compiling a program from source seems off-putting enough. The words *compile* and *source* seem to instill a sense of foreboding in the heart of the new user. That certainly was the case for me, anyway.

However, this method of installing programs is a lot easier than it sounds. In fact, now that you have some experience using the Terminal and command line, it is just plain easy—a sort of one, two, three, and you're done process.

Of course, you can live long and prosper without ever bothering to compile anything on your system. You can move along quite happily with your system as is, or you can just install programs by means of the very convenient Synaptic application, which you learned about in Chapter 5.

So why would you want to bother compiling programs from source? Well, although there are a lot of DEB packages that Synaptic can ease into your system, not every bit of software in Linuxdom is available in that file format. Or perhaps you want the newest version of the software you have your eyes on, but the version available via Synaptic is a slightly older version. Of course, there is that good old human nature factor at play as well—there may well come a time when curiosity gets the better of you, and you will want to move just one step beyond the way you've grown accustomed to doing things. After working through this short project, you will be able to do just that, and I am sure you will find that your initial worries will have been for naught.

What Is Source?

In order to understand what *source* is, you should understand a bit about how a program actually gets from its primitive state on the programmer's computer to an up-and-running application on your machine. First the programmer writes a program in a programming language. You have probably heard of programming languages such as BASIC or C, and there are many others. What the programmer actually writes with such a language is a set of instructions called the *source code*, or *source*. Your computer, however, cannot actually understand any of that source on its own. It is as if the computer speaks ancient Greek, and the source code is all written in French. The computer therefore needs some sort of interpreter to help it out.

The various languages that programmers use are called *high-level languages*—they are relatively easy for programmers to read. The computer, on the other hand, only understands *low-level languages*, which are quite difficult for most mere mortal programmers to deal with. To convert the high-level language instructions to a low-level language, the computer needs some other program to translate.

This can be done while a program is running, in which case the translator program is called an *interpreter*. Applications that run using an interpreter are usually *scripts*. The pyWings and Briscola applications in Chapter 8 are examples of such script applications.

The problem with such scripts is that they can be slower than most of the applications you're familiar with because the computer must run an interpreter, interpret the source code, and run the actual application all at the same time. This is like having a French book translated into Greek by a live interpreter—very slow indeed.

As an alternative, most programs use a compiler instead of an interpreter. A *compiler* translates the high-level source code into low-level *machine code*, or *object code*, that the computer can understand before the application is actually run. Once this translation is done, the computer never has to bother with the high-level instructions again; it can merely read the translated version each and every time it runs the program. This is like having a translated version of a foreign book that you can read any time you want to. Because computers can run compiled programs without simultaneously using an interpreter, compiled programs run faster than scripts. Most applications for all operating systems are, therefore, compiled.

Tarballs: The Containers of Source

Almost all source packages come in the form of tarballs (tarballs, DEBs, and even the RPMs, such as those used in Red Hat–based systems, are all referred to as *packages,* which is why the icon for such files looks like a little parcel-post box). As you learned in Chapters 6 and 8, tarballs consist of a group of files that have been archived into a single file, which is most often compressed to save disk space, much like Zip files on Windows systems or SIT (StuffIt) files on Macs. In Linux, the most common method of creating such archives is through the *tar program,* from which the tarball gets its name, while the compression of that archive is usually accomplished by means of the gzip program. Compressed tar files, or *tarballs,* can thus be recognized by their file endings, which are *.tar.gz,* or when compressed with the bzip program, *tar.bz2.*

As you learned in Chapter 8, the files archived in tarballs can be extracted by using the command line, but to keep things easy in this chapter, you can just use the simple double-click method that you used in Chapter 6, if you prefer.

The Basics

As I mentioned before, the process of compiling an application from source and then installing it is actually simple. Basically, after untarring the source tarball, you would use the following commands to accomplish the task:

./configure   To configure a *makefile,* which provides instructions for the make command

make   To translate the source code into object code that the computer can understand

sudo make install   To give yourself write privileges in protected folders and then install the application

make clean   To clean up the leftovers once the process is complete (to clean up the mess)

I know that sounds like a lot of commands, but as I always say, it is easier to actually do than it looks like on the page, so fear not.

Installing the Tools You Need

Because Ubuntu is designed with the average computer user in mind, it does not come with the various applications and libraries you need to compile applications from source. Fortunately, however, just about everything you need to get the job done is available via Synaptic. To get ready for the work at hand in this chapter (and many other jobs you are likely to do on your own in the future) perform searches for and install the following packages:

- build-essential
- checkinstall
- libgtk1.2-dev

I should mention that although there are only three packages you need to mark for installation, Synaptic will download and install quite a few more necessary packages in addition to these. Nothing to worry about though; just a heads-up so that you don't go into a mid-1960s freakout.

Project 9C: Compiling and Installing Xmahjongg

To get some hands-on experience with compiling a program from source, you will be working with a game called Xmahjongg, which you can see in Figure 9-6. If you've tried out the version of Mahjongg that comes with your Ubuntu distribution, you will notice that this one is much easier on your eyes and is a bit more colorful (check out the project site at www.lcdf.org/xmahjongg to catch a glimpse of it in its full four-color glory).
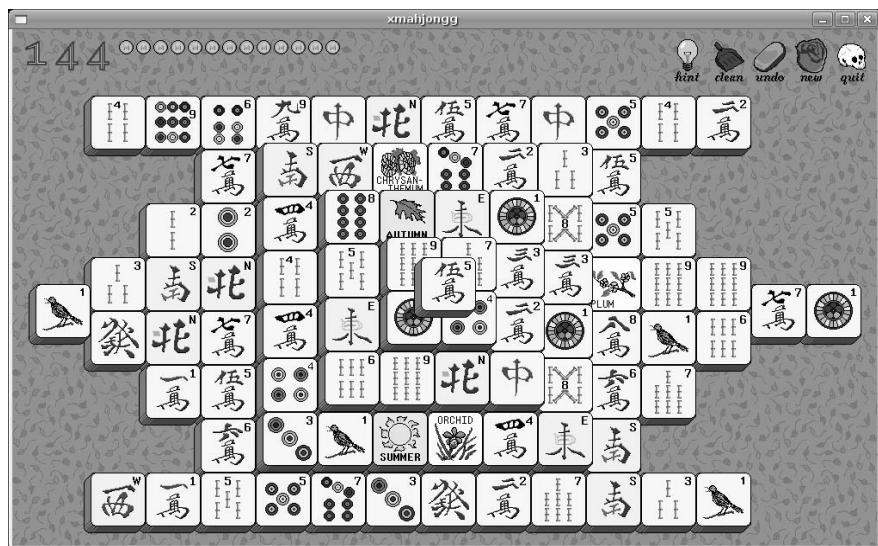
Figure 9-6: The Xmahjongg game

Xmahjongg, the version of Mahjongg that we'll be working with, is available via Synaptic, so it is not absolutely necessary to install it in this manner, but Xmahjongg provides a perfect opportunity to learn how to compile a program from source. The amount of source code isn't all that great, so it won't take too much time to do, and it requires no tinkering.

In case you are not already familiar with this genre of Mahjongg game, the idea is simple enough. Each tile has an image and there are four of each tile in the pile. You must match pairs of like tiles that are open on at least one side. When you click the two matching open tiles, they will disappear. The object of the game is, thus, to remove all the tiles from the board. A very simple solitaire game.

9C-1: Downloading and Extracting the Xmahjongg Files

To get started, you will first have to download the Xmahjongg source code. You can get this from the Xmahjongg project page at www.lcdf.org/xmahjongg. Download the tarball xmahjongg-3.7.tar.gz or a newer version if there is one. Do not download any of the other file types available on that page.

If you prefer, you can instead download the Xmahjongg tarball by using the `wget` command that you learned in Chapter 8. Just open a Terminal window, type the following command, and then press ENTER.

```
wget http://www.lcdf.org/xmahjongg/xmahjongg-3.7.tar.gz
```

Once you have the file on your hard disk, untar the xmahjongg-3.7.tar.gz file either via the command line, as you learned to do in Chapter 8, or via the simple double-click method. To make it easier for you to follow along with the directions I'll be giving you, be sure to place the untarred Xmahjongg folder in your home folder. Then you will be ready to roll.

Normally at this point, you would look through the folder to find some instructions for dealing with the package, just as you did in Chapter 8 for Briscola. In most source code packages, this information is included in an INSTALL file like that in the xmahjongg-3.7 folder (Figure 9-7). To read the INSTALL file, just double-click it, and it will open in Gedit.
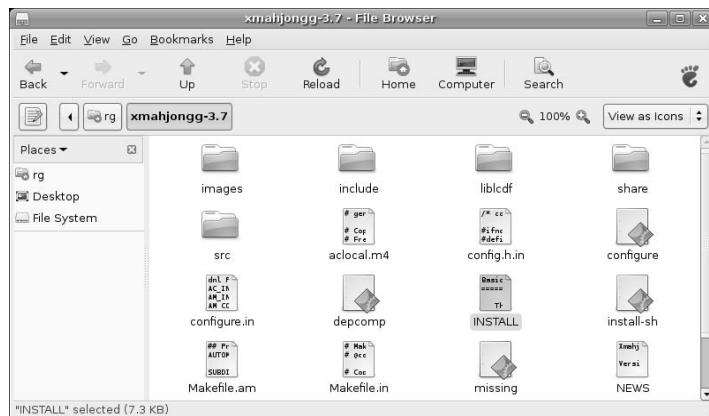
Figure 9-7: Identifying a package's INSTALL file

In this case, you can simply close the INSTALL file, as it prescribes the same steps as I've listed below. However, in the future, when you install other programs from source, you will need to follow the instructions in the INSTALL files that come with the source files for those programs. With most INSTALL files, the instructions will match those that follow, though I would suggest using the alternative method I'll be presenting as a substitute for the `make install` step.

Of course, it may well occur that you take a look at the contents of the INSTALL file and start wondering what alien tongue it is written in. In such cases when you have no idea what the INSTALL file is going on about, which sometimes happens, just look for a configure file in the package folder. If you find one, then just try doing things the way you will learn in this project.

9C-2: Running configure and make for Xmahjongg

Now that you have downloaded and untarred the Xmahjongg tarball, installation is pretty standard, though we will be finishing up in a slightly different way. Here's what you need to do:

1. Open a Terminal window, and then move into the new folder by typing `cd xmahjongg*` and pressing ENTER.

   The next step is sort of a setup phase that runs the configure script in the xmahjongg-3.7 folder. The configure script checks what files, compilers, and other things it needs, and then it searches your computer to see if those things are there, and if so, where. Based on this information, it writes a file called a makefile, which is a set of instructions that will tell the `make` command in the subsequent step how to set things up specifically for your system configuration.

2. Configure the program by typing `./configure` and pressing ENTER.

   While you are running `configure`, you will see lots of odd and mysterious things flowing through your Terminal window; this is essentially a running account of what is going on, each step of the way. Depending on the program you are dealing with, this could take a bit of time—a few seconds or a few minutes. Either way, you needn't worry. As long as the mysterious text keeps flowing and you don't get an error message at the very end of the whole process, all will be well.

   When `configure` has done its thing, you will see your prompt again, and you can go on to the next step, which is the translation, or *compilation*, step. The `make` command reads the makefile, created by `configure`, to see how things need to be set up on your machine. Then it proceeds to call on the compiler to translate the high-level source code into low-level, machine-readable files that can be installed in the subsequent step.

3. To perform this translation, type `make`, and press ENTER.

   Again, you will be treated to more mysterious text flowing through the window and a short wait, usually a tad longer than for the configure process. Once `make` has done its job and you see your prompt again, you are ready to install the program.

   Up to this point, you have not changed your system in any way. All the changes thus far have taken place in the xmahjongg-3.7 folder only. Your system is still as pure as the day you started. Of course, all that is going to end right now when you perform the final installation step.

9C-3: Installing Xmahjongg

Normally at this point of the process, you would type `sudo make install` and press ENTER, after which the files that were compiled in the `make` step would be installed in various locations throughout your system. You would then follow this up with the `make clean` command to tidy things up in the Xmahjongg folder by getting rid of any unnecessary files.

This traditional approach works fine enough, but things can get a bit messy if you find yourself wanting to upgrade or remove the application later on, as you have to keep track of where everything is and remove each item piece by piece using a series of `sudo rm` commands. Sometimes an application package will provide a `make uninstall` routine, but such routines don't always work perfectly, aren't available for every package, and require you to keep the project folder on your hard disk in order for you to perform the procedure.

If you are a traditionalist, you can go ahead and install Xmahjongg in this traditional manner. In the same Terminal window that you performed the `./configure` and `make` steps, just type **`sudo make install`**, and then press ENTER. You will be prompted for your password, so type it, and then press ENTER again. Xmahjongg will then be installed and ready for use.

The Alternative Installation Method

The traditional installation step is fine enough, but because of the limitations I mentioned in terms of removing the application, I recommend finishing things off in a slightly different way. What you are going to do, assuming you are willing to go this route, is to skip the `make install` step and instead convert the compiled, but uninstalled, application into a DEB package, and then install that. By doing this, you can remove the package quite easily using Synaptic whenever the desire strikes. Having a DEB package for your application, you don't need to keep the project folder, and you can save on hard disk clutter.

To create and install the DEB package, here's what you need to do:

1. Make sure that Synaptic, Add/Remove Programs, or any other APT installation tool is closed. As I mentioned before, you cannot have two APT tools running at the same time, and after all, `checkinstall` will be installing a DEB package.
2. Assuming you are using the still-open Terminal window in which you entered the make command, type **`sudo checkinstall`**, and then press ENTER.
3. A message will soon appear in the Terminal telling you that the ./doc-pak directory does not exist, and asking you if the program should create it. You want it to, so type **Y**, and press ENTER.
4. The `checkinstall` program will then begin performing a mock install, after which it will create the DEB package. Before completing the process it will ask you for a description of the package. Type something like **`An easy-to-see Mahjongg game`**, press ENTER, and then press ENTER again.

5. You will then be presented with a screen, such as that in Figure 9-8, giving you the chance to change the package information. You don't need to change anything, so just press ENTER.
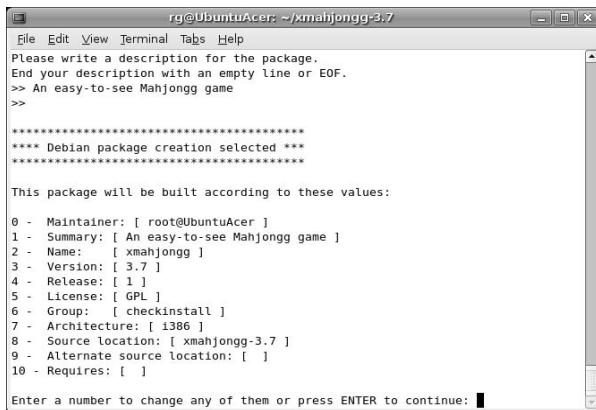
Figure 9-8: Accepting checkinstall's *default package information*

6. The checkinstall program will then finish up the packaging of the DEB package, install it, and then let you know when it is done.

And that is that. The Xmahjongg game is now installed on your system, and you will find an Xmahjongg DEB package within the Xmahjongg folder. You should move that DEB package to a safe place, such as a Tarballs or Software folder (or whatever else you choose to call it) so that you can use it again should you someday need to reinstall your system. Once you've moved the DEB package, dump the Xmahjongg folder, assuming it is in your home folder, by typing **cd ; sudo rm -r xmahjongg*** and pressing ENTER.

NOTE   *The semicolon, surrounded by spaces on either side, is used to separate commands written on a single line. In following the directions just given, you are really performing two operations in a single line:* cd *and* sudo rm –r xmahjongg*.

9C-4: Running Xmahjongg

As a general rule, programs compiled from source do not automatically install a launcher in your Applications menu; you must instead run them by command. Although you can run a program for the first time by typing a command in the Run Application panel applet, it is better to run the program for the first time by typing the command in your Terminal window. If anything has gone amiss during installation, the Terminal will tell you what the problem is, whereas the Run Program method would just leave you wondering what's going on.

To run Xmahjongg, just type **xmahjongg** in a Terminal window, and then press ENTER. If everything goes as it should, you can create a program launcher for Xmahjongg in your Applications menu, on the GNOME Panel, or in a panel drawer.

Project 9D: Running Java Apps: Risk

In Chapter 8 you learned to run a couple of applications based on scripts, such as the Python-based pyWings and the Tcl/Tk–based Briscola, but it just wouldn't be right to finish up our discussion of applications without mentioning what might be well considered the granddaddy of all scripting languages—Java. In this project, you will learn how to install the Java Runtime Environment and how to install and run the Java-based version of the classic board game, Risk.

9D-1: Installing the Java Runtime Environment

In order to run Java-based applications, or scripts, you need to first install the Java Runtime Environment. This is an excellent chance to use Automatix again. Just click **Codecs and Plugins** in the left pane of the Automatix window, and check the box next to the words *SUN JAVA 1.6 JRE* in the right pane. After that, follow the process you learned earlier for installing packages via Automatix. Once the installation gets going, another Terminal-like window will appear showing what's going on. Don't panic; you won't have to do anything with that window. The whole Java package is pretty hefty in download weight, so don't get freaked out if it seems to take longer than you're used to.

Eventually, a graphical Configuration window, such as that shown in Figure 9-9, will appear. In that window, read and then agree to the licensing terms (by checking the box), and then click the **Forward** button. More white text will roll by in the other two windows until the process is complete, after which the Automatix window will be back to its usual graphical self. You can then quit Automatix.
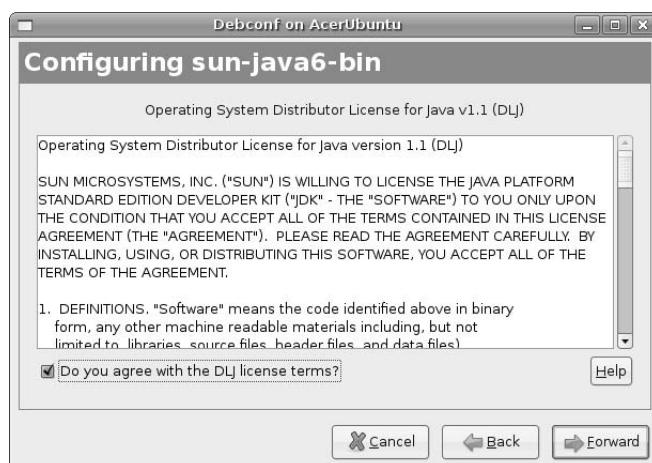
Figure 9-9: Configuring Java support for your system

9D-2: Getting the Risk File

Risk, as well as many other Java-based applications, is available from
www.sourceforge.net, but to make things easier, just point your browser
directly to the Risk project home page at http://jrisk.sourceforge.net
(and don't forget the *j* at the beginning of that URL). On the main page,
click the **Download** link. Then, on the Downloads page, click the **Risk jar
and source** link.

A window will appear asking you what to with the Risk Zip file, and your
choices are to save it to disk or to open it with Archive Manager. Since you're
going to have to unzip the package once you download it anyway, you might
as well choose the top option, *Open with Archive Manager*, and then click **OK**.
The Archive Manager window will appear. All you have to do now is click
Extract, and when the Extract window appears, select your home folder as
the destination by double-clicking its icon, and then click **Extract** again.
Once the file is extracted you can close the Archive Manager window.

You will now have a new Risk folder within your home folder. If you take
a look in that folder, you will notice the file Risk.jar. This is the game that you
will be running via the command line. In the future, should you choose to
download other Java-based applications, the file with the .jar extension will
be the one you will be trying to run.

9D-3: Running Risk

Now that you've installed the Java Runtime Environment and downloaded
and extracted the Risk Zip file, you are ready to run Risk without any further
work. To get going, open a new Terminal window, and do the following:

1. Move into the Risk folder by typing `cd Risk` and pressing ENTER.
2. Type `java -jar Risk.jar` (be sure to place a space between java and -jar),
    and press ENTER. Risk (shown in Figure 9-10) will soon appear, after which
    you can start playing.

In case you're wondering, here's what you did in that last line: The first
part of the command string, java, calls the Java Runtime Environment into
action; the -jar flag after that tells Java that you are going to be running a
JAR file; and the last part is the actual file you are going to run, `Risk.jar`. In
the future, if you choose to run other Java-based applications, just follow the
same pattern: java -jar *application_name*.jar.

Figure 9-10: Risk

Running Windows Applications with Wine

Now that you've learned just about every way there is to add applications to your system, I might as well throw in one more—running Windows applications. Despite the fact that there is a Linux equivalent to most of the Windows programs that people use or need, there may be one or two programs that you will come to miss. For me it has long been the freeware version of the Austrian card game Schnapsen. Fortunately, it is possible to run some Windows applications from within Linux with the help of a program called Wine.

The folks at Wine seem keen on pointing out that Wine is not a Windows emulator, preferring to call it a *Windows compatibility layer.* In fact, even the name itself drives home the point, as Wine is a recursive acronym for *Wine Is Not an Emulator.* Whatever way you choose to look at it, its function is to allow you to run Windows apps without having Windows installed on your machine.

It is only fair to point out that Wine continues to be a work in progress. It works fairly well with some programs and not at all with others. Things have improved, however, and Wine now seems to work better with more applications. Until the latest release, for example, I had been unable to get my beloved Schnapsen game to run. Now it does, and pretty well at that (Figure 9-11).
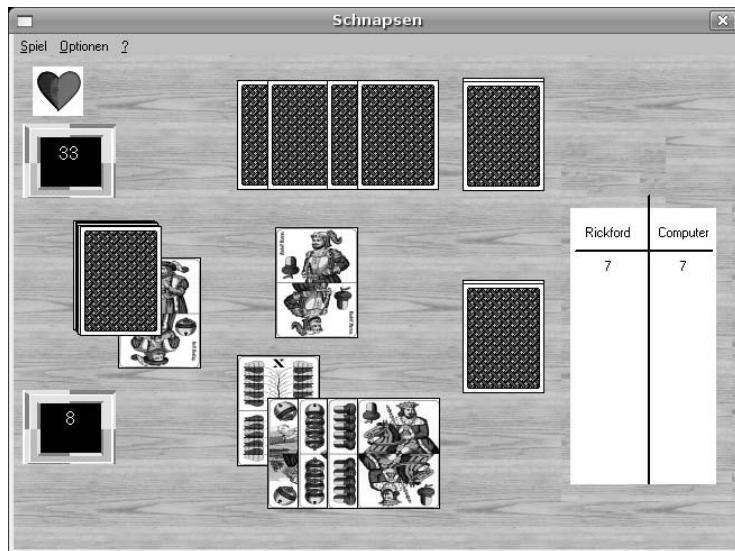
Figure 9-11: A Windows application (Schnapsen) running under Linux

If you are curious as to which apps are known to run under Wine, and to what degree of success, check out the Wine home page at www.winehq.org, and click the **AppDB** link. And if you are curious about Schnapsen and aren't afraid of wading through a bit of German, get the program at www.members .a1.net/zillinger. You can also learn the rules of Schnapsen (and any other card game in the world) at www.pagat.com.

Installing and Checking Out Wine

You can get Wine via Automatix by clicking **Virtualization** in the left pane of the window and the checking the box next to the word *Wine* in the right pane. After that, click **Start**, and you'll be on your way. Once the installation is complete, quit Automatix.

Before the installation is complete, the Wine installation script will automatically try to create a fake Windows C: drive within your user folder by running Wine Configuration. Automatix will warn you of this, and once the Wine configuration window appears, all you need to do is click **OK**, the Wine configuration window will close, and Wine itself will be ready for action.

If the installation completes without the Wine configuration window ever appearing, you can achieve the same thing on your own by running *Wine File*, which is a Windows Explorer–like file browser (Figure 9-12). You can run Wine File by going to **Applications ▶ Accessories ▶ Wine File** or by opening the Run Application window, and typing `winefile`. Once you've opened Wine File, there isn't anything else you need to do, since the work will have already been done in the background. You can immediately close Wine File.

Installing a Windows Application in Wine

To get a feel for how you can go about installing a Windows application to use under Wine, I will point you to one that will definitely work—a pretty cool tabbed text editor called NoteTab Light (Figure 9-13).
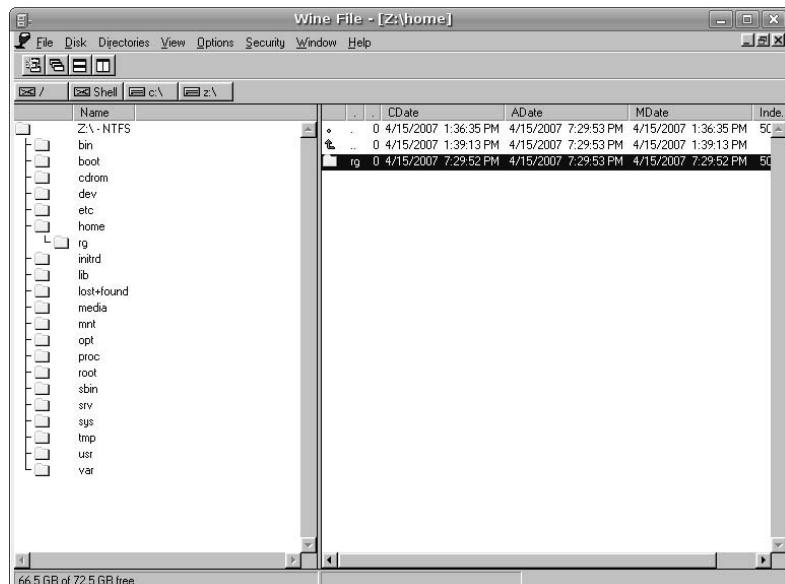
Figure 9-12: Windows Explorer, Wine style—Wine File

To get NoteTab Light, go to www.notetab.com, right-click the **EXE Package** button in the NoteTab Light 4.95 section, and then select **Save Link As** in the popup menu. When the download is complete, be sure that the NoteTab_Setup.exe file is in your home folder. If it's not, put it there so as to make it easier for you to follow along.
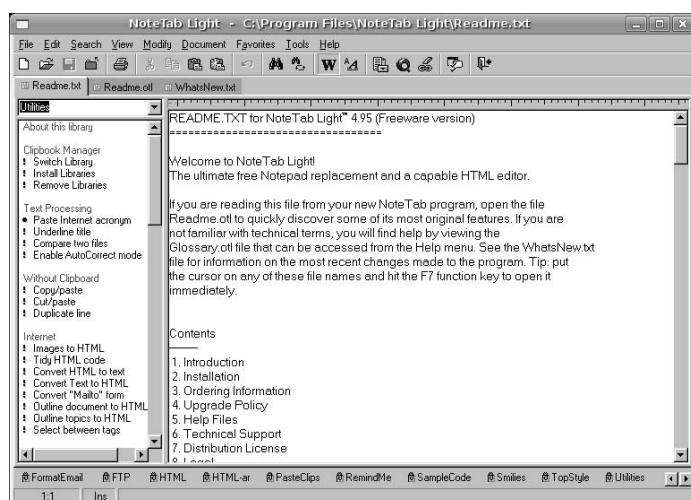
Figure 9-13: NoteTab Light

Now, open a Terminal window, type `wine NoteTab_Setup.exe`, and press ENTER. A few moments after that, the same setup wizard that you would see if you were installing NoteTab Light in Windows will appear (Figure 9-14). Go through the wizard, accept the license agreement, and then accept all the defaults along the way until the installation is complete.
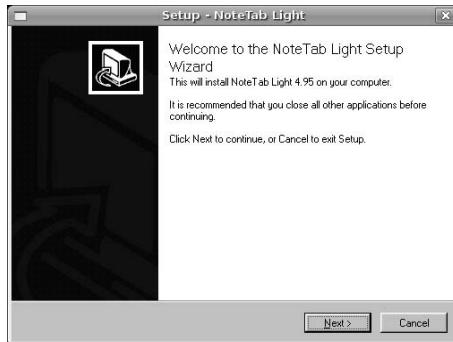
Figure 9-14: A Windows installation wizard running under Wine

Running a Windows Application in Wine

Running NoteTab Light is quite easy because it provides you with a desktop launcher. Just double-click that launcher, and NoteTab Light will soon appear, just like a regular Linux app would. You can also run NoteTab Light and other Windows applications under Wine by going to the **Applications** menu, selecting **Wine ▸ Programs**, and then selecting the program you want to run.

Installing Microsoft Internet Explorer

Although more and more people are trying to get away from Microsoft Internet Explorer, there are times when you just have to have it. There are, after all, sites that won't let you use anything else. For Linux users, this used to be a problem, because getting IE to work via Wine could be quite a geeky chore. Not so anymore, thanks to IEs 4 Linux, which is a simple utility that helps you easily download, install, and set up Internet Explorer for use with Wine on your Linux system.

Before using IEs 4 Linux, you need to have two things installed: Wine, which you may have installed while following along with the previous section, and cabextract, which you can easily download via Synaptic. Once both packages are installed, you are ready to proceed, and you don't even need to go to the project home page to do so (www.tanaka.com.br/ies4linux, in case you're wondering). Here are the steps:

1. Download the IEs 4 Linux package by opening a Terminal window and typing the following:

    ```
    wget http://www.tatanka.com.br/ies4linux/downloads/ies4linux-latest.tar.gz
    ```

2. When the download is complete, extract the compressed tarball by typing `tar zxvf ies4linux-latest.tar.gz` in the Terminal window and pressing ENTER.

3. Move yourself into the ies4linux folder via the Terminal by typing `cd ies4Linux-*` and pressing ENTER.

4. Run the script that will download and set up Internet Explorer for you by typing `./ies4linux` and pressing ENTER.

5. IEs 4 Linux will download and install Internet Explorer 6, but it will first ask you if you also want to download Internet Explorer 5.5 and Internet Explorer 5.01. Assuming you don't want these, just press ENTER each time when asked. (Type **y** and press ENTER if you do want them.)

6. When asked what language version of Internet Explorer you want, press ENTER for English, or type the two-letter code of the language you want, and then press ENTER.

7. You will be asked if you want IEs 4 Linux to install the Flash 9 plugin and create a Desktop shortcut for you. You should want this, so just press ENTER. (Type **n** and press ENTER if you don't want it.)

After that, IEs 4 Linux will begin downloading, installing, and setting up Internet Explorer. When it is done, you will find an Internet Explorer-like launcher on your desktop, which you can then double-click to start up your own Wine-run version of Internet Explorer 6 (shown in Figure 9-15).
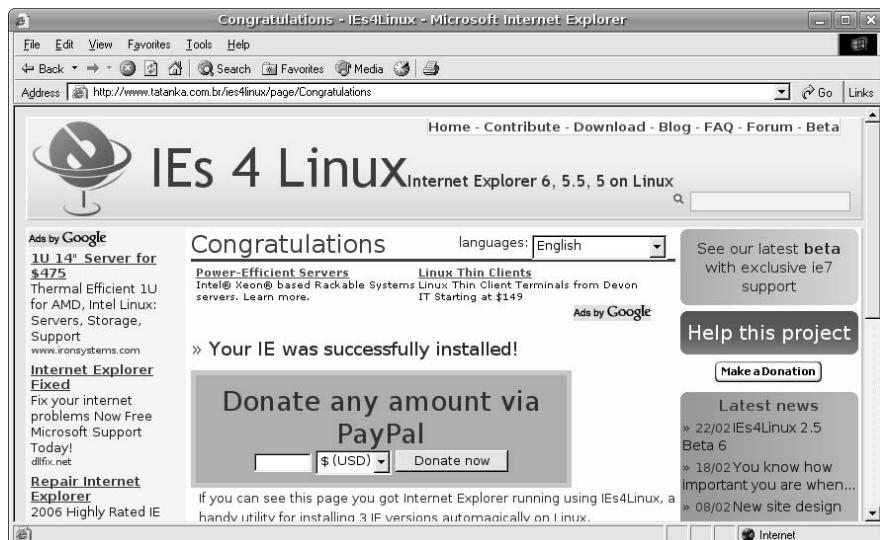
Figure 9-15: Internet Explorer 6 running under Wine

10

GUTENBIRD

Setting Up and Using Your Printer and Scanner

Two of the most common computer peripherals are printers and scanners. This only makes sense, as it is those two tools that turn a web-surfing, game-playing, music-churning, number-crunching box of chips into a meaningful production tool—a virtual publishing house, if you will. These two tools help your computer convert digital information into hard copy (in the case of printers) and hard copy into digital information (in the case of scanners). It is not surprising, therefore, that these tools often come together these days in the form of multifunction printers.

In this chapter, you will learn how to connect these useful devices to your computer, how to set them up, and how to use them. If you're more into working with your digital camera than with a scanner, you might want to sneak a peek at Chapter 14; otherwise, put on your printer's smock, and follow on. . . .

Printers

Unless the only thing you use your computer for is playing games, listening to MP3s, or stopping doors on hot, breezy days, you will no doubt want to hook up your machine to a printer. Despite the paperless office era that the personal computer was supposedly going to usher in, it seems that the computer's strength as a desktop-publishing and general work tool has made producing high-quality printed documents an even more attractive proposition than ever before.

Confirming That Your Printer Is Supported

Setting up a printer to work with your new system is a pretty easy task, but you do have to make sure that your printer is supported by Linux. Fortunately, it seems that printer support in the Linux world is getting much, much better than it once was. In general, support for Epson, Brother, Samsung, and Hewlett-Packard printers is pretty good, while support for other makers and other printer types is a bit spottier, though improving.

If you really want to make sure your printer is supported (and why wouldn't you?), the best thing to do is go to www.linuxprinting.org. On that site, you can check out the online database to see if your printer is currently supported and, if so, to what degree. Listings for supported printers also include information on what drivers are best for your purposes. If you're thinking of buying a printer, there is also a page of suggested makes and models. Read this before making your purchase decision. You might also want to consult Ubuntu's list at https://wiki.ubuntu.com/HardwareSupport-ComponentsPrinters.

Setting Up Your Printer

If you have a USB printer, get started by connecting it to your computer and then powering up the printer. If your printer connects to your computer's parallel port, you will first have to turn your computer off, connect your printer to the computer, turn on the printer, and then start your computer. Of course, if your printer was connected and on during installation, you can skip this step. Either way, your printer should be detected, though you won't get one of those immediate new-hardware-found alerts that you do in Windows.

Once your system has restarted, here is what you need to do:

1. Go to the **System** menu, and select **Administration ▸ Printing**.
2. A Printers window will appear, showing a printer icon with the name *New Printer*. Double-click that icon, after which a setup wizard will appear (Figure 10-1).

    The name of your printer should appear in the pane below the words *Use a detected printer*, due to the wonders of plug and play (which means that your printers, scanners, and other devices can tell your computer what they are). Unfortunately, just because your system knows which

printer you have hooked up to it hardware-wise doesn't actually mean it can communicate with that printer in terms of printing (ditto for Windows and Mac OS X). That's what drivers are for.

Figure 10-1: The printer setup wizard

3. Continue by clicking the **Forward** button, which will take you to the second page of the wizard, shown in Figure 10-2.
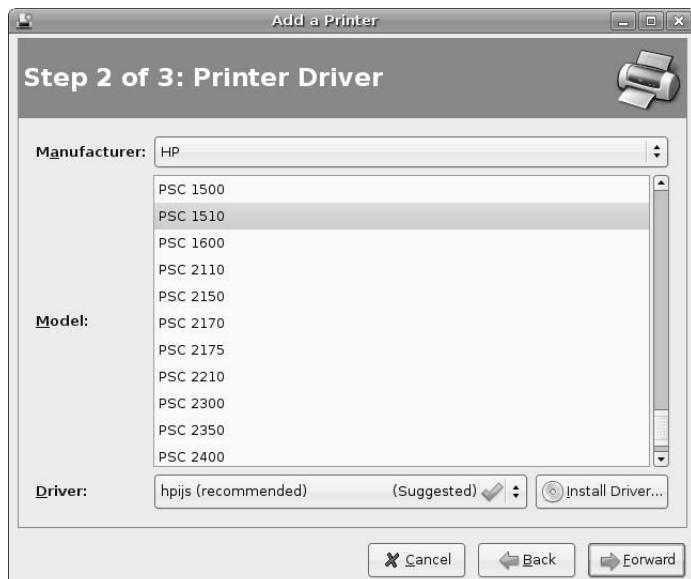
Figure 10-2: Selecting the make and model of the printer in the printer setup wizard

This second page of the wizard should automatically open with the make and model of your printer selected, as well as the recommended driver for that printer. If the model chosen is not correct, scroll down the list until you find the correct model. If your model isn't listed, try the one that was selected by default.

4. Once your printer is selected, click the **Forward** button, which will bring you to the last page of the wizard.

5. On this page, type a name for your printer, and, if you want, give a description and its location. When you're finished, your window should look something like mine in Figure 10-3. You can then click **Apply**.
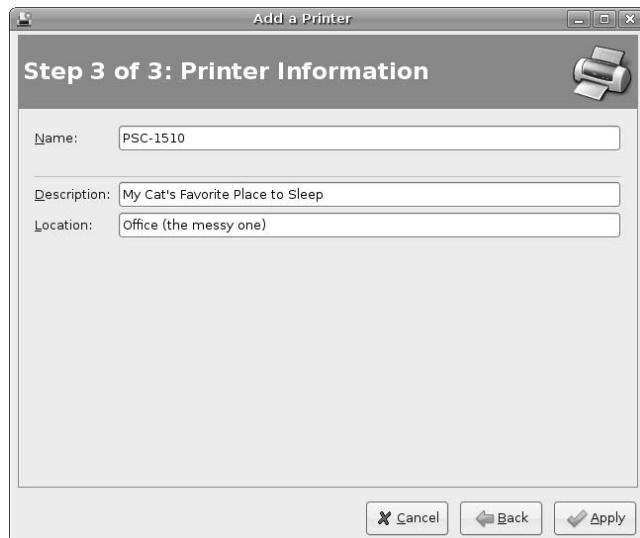
Figure 10-3: The final page of the printer setup wizard

The wizard should close, leaving you at the Printers window where you started this quest. An icon for your printer should now appear in that window alongside the New Printer icon that you originally found there. If you have set up more than one printer, you can choose which one you would like to be the default by right-clicking the icon for that printer and selecting **Make Default** in the menu that appears. The icon for that printer should then be emblazoned with a blue checkbox (Figure 10-4), indicating that it is the default printer for your system.

Figure 10-4: The New Printer icon in the Printers window after completion of the printer setup wizard

Your printer is now set up and ready for use. However, just to make sure that everything is hunky-dory, it is always a good idea to print a test page.

1. Right-click the icon for your printer in the Printers window, and select **Properties** in the popup menu.

2. In the Properties window that appears, click the **Print a Test Page** button.

Your system will send a test document to your printer and let you know it's done so in a small window, which you can close. The printer should print out the test document shortly.

NOTE *If your test page didn't come out the way it should, or if your printer wasn't configured automatically or correctly by the system, check out www.linuxprinting.org, and see if there are any special requirements or caveats for your model.*

For the Driverless Among You

As I mentioned, there are some printers for which Linux does not yet have built-in support. For those of you who find yourselves with such printers, there are a few routes you can take in order to get things to work.

Checking the Connections

You'd be surprised how many times I have triumphantly solved someone's printer problems by simply wiggling or replugging their USB or parallel connectors. Loose connections are often the culprit when your printer's model name fails to appear in the first page of the printer setup wizard.

If that approach fails, browse to www.linuxprinting.org, as it is sometimes the case that Linux can only support certain printers if they are connected via the parallel port, even if they work via USB in other systems.

Third-Party Drivers

Recently, more and more Linux printing drivers are becoming available. If you don't find your printer on that second page of the printer setup wizard, just trying Googling the make and model of your printer plus the word *linux*. In my case, having a Samsung SCX4100 laser printer that did not appear at www.linuxprinting.org, I searched for *samsung+SCX4100+linux.*

This led me to www.driverstock.com, which provides free printer drivers for most operating systems, including Linux. On that site, I found not only the driver for my printer, but also the driver for its built-in scanner.

You might also want to check the website for the manufacturer of your printer, as many now provide Linux drivers for a number of their printers. Brother, Lexmark, Hewlett-Packard, and Samsung do, to name a few. Canon seems to be the main holdout as far as the big boys go. There is also a German company (www.turboprint.info) that provides Linux drivers, albeit for a fee, for machines that are really hard to deal with and for high-quality graphic solutions.

If you happen to find and download a driver for your printer that does not come with its own installer (the driver file should end in *.ppd*), go back to the printer setup wizard, click the **Install Driver** button on the second page, and then navigate to the driver file you downloaded.

Don't forget to give the Ubuntu forums (www.ubuntuforums.org) a try to see if anyone there has any experience getting the printer in question to work on their system.

Trial and Error

Finally, there is always the old trial-and-error approach, which works on occasion. When setting up your printer via the wizard window, try selecting **Generic ▶ PostScript Printer**, which works with some machines. If that fails, try choosing from one of the other models and/or drivers available for printers from your printer's manufacturer.

After you are done with the wizard, open the Preferences window for the printer you added, click the **Print a Test Page** button, and see what happens. If nothing happens, click the **Driver** tab, select a different printer model, and then press the **Print a Test Page** button again. Repeat that process until something works. With any luck, something will . . . might . . . well, just give it a try if you're desperate.

Printing Details

Now that your printer is set up, you will no doubt want to start printing. This is an easy task and not much different from how it works in the Windows and Mac worlds, so you will need little explanation in this regard. There are a few points of difference, however, so I will touch upon these.

Printing to PDF

One of the nice features of Linux is that you can save most documents as PDF files. In some cases, such as in OpenOffice.org, you do this by exporting the document to PDF. In most other applications, however, you do it via the Print dialog box, in which case you are said to be "printing to PDF." Whether you are *saving* to PDF, *exporting* to PDF, or *printing* to PDF, you are essentially doing the same thing: creating a PDF file of your document.

This is very handy, as it allows you to create documents that cannot be altered by others and yet can easily be read regardless of what word processor program or computer platform the reader is using. Best of all, this feature, which you would have to pay a pretty penny for in the Windows world, costs you nothing, as it is built in to your system.

In most GNOME applications, you can print to PDF by going to the **File** menu of the application in question and selecting **Print**. When the Print window (Figure 10-5) appears, select **Create a PDF document** in the Printer pane, and then click the **Print** button.
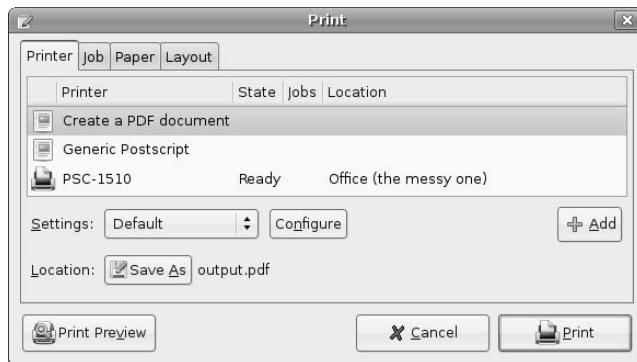
Figure 10-5: Printing a file to PDF in most GNOME applications

Printing Web Pages to PDF

Printing to PDF is also a very handy way of saving a web page that you would like to keep on hand in its graphical entirety for future reference. At this time, Firefox does not support this feature, though it does allow you to print a page to PostScript. As long as you are strictly using the file on your own system or giving it to someone else with a Linux system, you most likely won't notice any difference between PS and PDF files, as they look and act pretty much the same.

To print a web page to PostScript, go to the Firefox **File** menu, and select **Print**. When the Print dialog window appears, select **Print to file**, and then give your file a name, being sure to keep the *.ps* file ending. Click **Print**, and within seconds, the file will appear on your hard disk.

This is all fine and dandy, but if you want to distribute the document to a broader audience, the PostScript alternative just won't do, as most Windows users won't have any way to view the file. So what can you do? Well, one way is to create the PS document and then convert it to PDF using the `ps2pdf` command. This is very easy to do, though it doesn't always work so well. You may find, for example, that the text within your PS document vanishes in the output PDF file, or that the text appears, while the graphics vanish. Giving it a try can't hurt, though, so here's how you do it. Just type `ps2pdf` *`filename.ps filename.pdf`* in your Terminal, and press ENTER. Yup, that's all there is to it.

Project 10: Creating a Virtual PDF Printer

If you aren't satisfied with the results of a `ps2pdf` conversion, but you still want to be able to print web pages to PDF, there is a fairly simple solution—create a virtual PDF printer. By doing so, you will be able to print to PDF in any application that allows printing (sorry gamers).

10-1: Getting and Setting Up the Files You Need

To get started, run Synaptic, perform a search for *cups-pdf*, and install it. Once it is installed, open a Terminal window, type `sudo chmod +s /usr/lib/cups/backend/cups-pdf`, and press ENTER. When prompted for your password, type it, and press ENTER.

10-2: Setting Up Your Virtual PDF Printer

After the cups-PDF installation and preparation is complete, you can go on to setting up your printer. To do this, go to the **System** menu, and select **Administration ▸ Printing**. Provide your password when prompted, and then, when the Printers window appears, double-click the **New Printer** icon. In the Add a Printer wizard that appears, select **Local Printer** and **Use a detected printer**, and then click once on **PDF Printer** to select it. Once you're done, click **Forward**.

On the second page of the wizard, select **Generic** in the Manufacturer menu, and then select **postscript color printer rev3b** in the Model section. Your window should then look like that in Figure 10-6. If it does, click **Forward**.
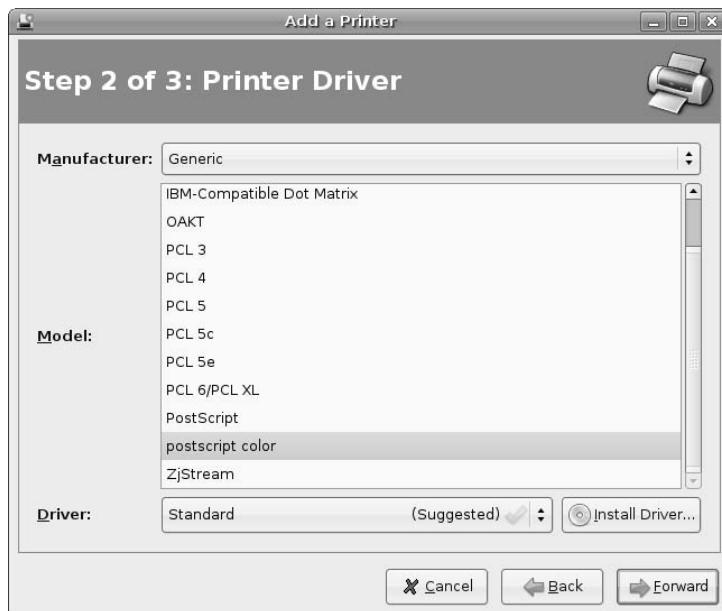
Figure 10-6: Creating a virtual PDF printer

Finally, on the last page of the wizard, fill in the text boxes as you like and click **Apply** to seal the deal. You might want to shorten the name of the printer to something more manageable, such as *My PDF Printer*.

10-3: Using Your Virtual PDF Printer
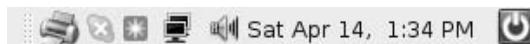
Once you've set up your virtual PDF printer, you can use it quite easily. In fact, because your system views it as a real printer, all you have to do is go through the usual steps for printing any document, but just be sure to select your virtual PDF printer as the printer for the job in the Print dialog window when it appears.

Once you've printed a document using the virtual PDF printer, you can find your documents in the PDF folder that cups-PDF will create for you within your home folder the first time you use the virtual PDF printer.

Canceling a Print Job
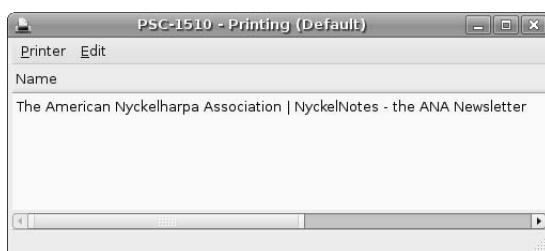
It happens to all of us. You wanted to print just 1 page of a 57-page document, but you accidentally started printing the whole thing. What can you do to save your ink and 56 sheets of paper? Fortunately, the solution is simple.

Once you've clicked the **Print** button and the print job is sent to your printer, a small printer icon will appear somewhere at the right end of the top GNOME Panel (usually to the left of the other items there), as you can see in Figure 10-7.

Figure 10-7: A printer icon appears in the GNOME Panel while printing.

Just click that icon once, and a window showing your current and queued print jobs will appear (as shown in Figure 10-8). Your errant print job will be listed in that window, so click the name of the job to select it, and then, in the **Edit** menu, select **Cancel Jobs**. If you just want to stop things temporarily, you can select **Pause Jobs** instead.

Figure 10-8: Canceling a print job via the print queue window

After you do this, the print job listed in the queue window will disappear, and your printer will stop printing. You can then close the print queue window. This is a very easy process that you may well find is more effective than what you've experienced in other operating systems.

In some cases, things will be even easier, particularly if you installed your printer driver yourself by means of the installer that came packaged with the driver. In many such cases, the driver will provide its own progress

window that will appear whenever you print a document. If so, you can simply click the **Cancel** button (or equivalent) in that window to cancel the printing job.

In some cases you may have to clear your printer after canceling a print job. You can do this by turning your printer off, waiting a few seconds, and then turning it on again.

Scanners

Scanners are extremely useful and about as cheap a peripheral device as you can get. They allow you to take images or pages of text and input them, in digital form, into your computer—in much the same way as you would duplicate a document on a copy machine. However, even as digital cameras are rapidly overtaking traditional film cameras as the photographic device of choice for the masses, the number of people using scanners to transfer their non-digital images into digital form is slowly decreasing.

Despite this trend, scanners are not in immediate danger of extinction because there are more images around than those you photograph yourself. In some ways, you could even argue that scanners are becoming a bit more common due to the fact that they now often come as part of the increasingly popular multifunction printers.

Even though scanners have been around for a relatively long time, support for them in Linux is still a bit spotty. Fortunately, this is changing for the better with every new Linux release. The *back end*, the essentially hidden part of your system that handles scanner recognition and support in Linux, is called Sane, while the graphical interface for Sane is called XSane (shown in Figure 10-9). If you are wondering whether Linux will be able to recognize your scanner, or if you are trying to figure out what type of scanner to buy, you will probably want to go to the Sane website, www.sane-project.org/sane-mfgs.html.
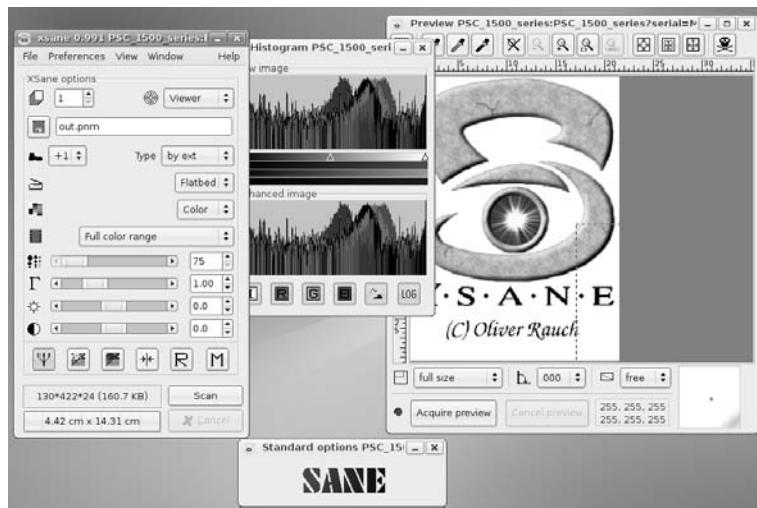
Figure 10-9: Scanning with XSane

There you will be able to see whether your scanner is supported or get tips about what scanner to buy. As I have mentioned before, you can also try out the Ubuntu forums (www.ubuntuforums.org) and ask for Ubuntu-specific recommendations there.

Scanning with XSane

To run XSane, go to the **Applications** menu, and select **Graphics ▸ XSane Image scanning program**. XSane will then perform a search for an attached scanner. If it finds one, it will start up. If it doesn't, it will pop up a tiny window that says, "No devices available." You can do little else at that point other than click the **Close** button.

If you do run up against this problem, you can try running XSane as root by going to the **Applications** menu and selecting **System Tools ▸ Run as different user**. When the Run program window appears, type `xsane` in the Run box, and then press the **OK** button. If your stars are in alignment, XSane should detect your scanner, pop up a message that it is dangerous to run XSane as root, and then open up in its full multi-windowed glory. If your scanner still isn't detected, a trip to www.sane-project.org or www.ubuntuforums.org might be in order to see if there are any known workarounds for your particular scanner model.

To actually scan something, place the photo, document, or whatever it is you are planning to scan on the scanner bed, and then click the **Acquire preview** button at the bottom-left corner of the XSane Preview window (which usually opens up at the right end of your screen). Once the preview appears, use the selection tools in the same window to define the exact area you want to scan, and then choose your resolution and color depth settings in the main XSane window (which usually opens up at the top left of your screen). When everything is ready, click the **Scan** button, after which your scanned image will appear in a new Viewer window (as shown in Figure 10-10).

In that window, you can perform some minor tweaks of the scanned image using the buttons and menu items provided, and then save the image by going to the **File** menu and selecting **Save**. If you scanned a document that you want to convert to text, click the second button from the left (the one that says **ABCDEF**), and you will also be able to save the file as a text document.

NOTE   *If you happened to download and install a driver for your scanner or multifunction printer from the device manufacturer's site (or elsewhere), you may find yourself with another scanning application provided within the driver package. If you prefer, you can use that application instead of XSane to perform your scanning chores.*

Size 443 x 667 pixel, 1 bit/color, 3 colors, 150 dpi x 150 dpi, 865.7 KB
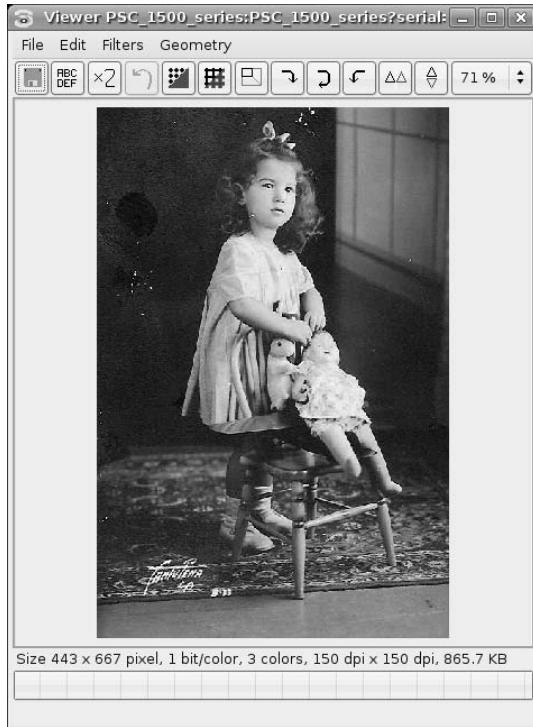
Figure 10-10: The results of your scan are displayed in a separate Viewer window.

Why Are My Scanned Images So Big?

To wrap up this section on scanning (and, for that matter, this chapter), let me address a question that seems to confuse a lot of people. One of the first areas of confusion is that there is a general blurring of how the terms *ppi (pixels per inch)* and *dpi (dots per inch)* are used. Most applications use these terms interchangeably, and yet they aren't really the same thing. To make things simple, when you are talking about images on your screen, you are talking about *pixels* (the little squares that make up your screen image) per inch, and when you are talking about printer resolution, you are talking about *dots* (of printer ink) per inch.

Your computer screen generally has a resolution of 76ppi, while most modern inkjet and laser printers have a resolution range of 300 to 1200dpi, or sometimes even more. This means that a photo scanned at 76ppi, which looks just fine on your screen, ends up looking pretty lame when you print it out. On the other hand, when you scan a picture at 300ppi, the image will look much better in your printout but will seem gigantic on your screen. This makes sense, as the resolution of your image is more than three times that of your computer screen's resolution. The result is that your computer can accommodate the higher resolution of the image only by displaying that image at three times its original size.

As an example, have a look at Figure 10-11, where you can see an identical image scanned at three different resolutions: 76ppi, 150ppi, and 300ppi. As you can see, the 76ppi image at the far left (measuring 5 × 6 inches—about the size of the hard copy itself) is the smallest, while the other two images are proportionally bigger (about 10 × 12 inches for the 150ppi image and about 22 × 25 inches for the 300ppi image).

Figure 10-11: The same image scanned at three different resolutions

What Resolution Should I Use When Scanning?

What resolution you use when scanning really depends on a variety of factors, the most important of which is what you plan to do with the image when you're finished. When I look at Figure 10-11 on my computer screen, the smallest image looks best, the middle image looks okay, and the largest looks a bit odd, not as sharp as the other two. Basically, when scanning images for display on a computer—on web pages, for instance—it is probably best to stick with a ppi similar to typical screen resolutions or slightly larger: 76 to 150ppi.

When it comes to printing, a whole new set of considerations comes into play. First of all, there are the limitations of your scanner, since different models have different maximum resolutions. The resolution limits of your printer itself are also, naturally enough, a major consideration. For example, laser printers and inkjet printers have different characteristics; laser printers will produce better-quality images than inkjet printers, while inkjet output will be more greatly affected by the type of paper used than a laser printer will be. Of course, your printed output is not going to suffer if you scan your images at higher resolutions than those at which you plan to print them out, but you will end up with a lot of files taking up too much disk space. Remember: *The higher the resolution of a scanned image, the greater the file size in terms of disk space.* If this is of concern to you, you can simply resize the images after you're

done printing using an application such as the GIMP (more on that in Chapter 14), but if you would prefer not being so cavalier with your use of disk space from the get-go, you can follow these very general guidelines:

- If you are using a laser printer, scan at the same resolution at which you are going to print.

- If you are going to use an inkjet printer with photo-quality paper, scan at about 80 percent of your target printout resolution—about 240ppi for a 300dpi printout.

- If you are using an inkjet printer with regular paper, scan at about 65 percent of your target printout resolution—about 195ppi for a 300dpi printout.

Needless to say, these are just suggestions to get you started. What works best for you and your particular scanner/printer setup may be slightly different. Nothing works better than a bit of experimentation and trial and error. In this case, you can't really go wrong. Just give yourself some time, don't get frustrated, and, most importantly, don't wait until you desperately need to scan something before trying things out—stay ahead of the game.

11

FONT FEATHERED FRENZY

Adding New Fonts to Your System

There are basically two kinds of fonts: bitmap and outline. The difference between these two is essentially the same as that between bitmap and vector graphics, which I will talk more about in Chapter 14. *Bitmap* characters are stored as a map of dots—the bitmap. The main limitation of bitmaps is that they only look good at the size and resolution they were designed for. Just like bitmap images, the more you enlarge a bitmap character, the worse it looks. This is particularly noticeable in rounded characters, such as *O* and *P*, where "the jaggies" becomes an issue.

Outline fonts, on the other hand, are similar in concept and design to vector graphics. Each character is stored as a mathematical formula, and just like vector graphics, outline characters keep their clean shape no matter how much you enlarge them. The main outline font formats are Type 1, or PostScript, which was developed by Adobe, and TrueType, which was developed by Apple. As free TrueType fonts are so readily available on the Internet and so easily handled in Ubuntu, I will focus on TrueType fonts in this chapter.

Your Ubuntu system comes with a wide variety of very usable and, at least to my eyes, rather handsome TrueType fonts. However, these tend to be a bit on the conservative side of the aesthetic spectrum, and many users will want to add a few more distinctive fonts to the system repertoire. In my own case, I had this really cool idea of writing messages to my friend in old Scandinavian runes. (Of course, my friend wet-blanketed the idea, so it all came to naught. . . .)

You probably won't be interested in sending cryptic, runic messages to your friends, but you may want to print out an award for an event using some sort of Gothic font, or you might be preparing a newsletter for the local chapter of your snail-breeders society and want to use a font that is round, bubbly, and slimy. Whatever your penchant, purpose, or desire, you will probably come to the point when you want to install some other TrueType fonts on your system.

Before you install anything, of course, you have to find some fonts. The Internet is always good source, and there are many sites that have a variety of freeware, shareware, and for-sale TrueType fonts available for download. When choosing and downloading fonts, it is best to select those designed for Windows rather than those designed for Mac. Fonts designed for the Windows world will most likely be in the form of Zip files, which will pose no problem for you, because you can extract them with File Roller. Simply double-click them, just as you have with the other archived files you have used thus far, and click the **Extract** button in the File Roller window when it appears. The font file will appear on your desktop (or wherever your Zip file happened to be) as an icon showing an upper- and lowercase sample of the first letter in that font (Figure 11-1).

Figure 11-1: Font icons display the first letter in the font.

In addition to getting a glimpse of what the fonts look like through these icons, you can also see all, or at least almost all, of the characters in a given font by double-clicking the font icon. A window, as shown in Figure 11-2, will open, showing you most of the characters in A-to-Z format and then in the traditional "The quick brown fox jumps over the lazy dog" format that you may well remember from your junior or senior high school typing classes.

Figure 11-2: Previewing a font by double-clicking the font icon

Project 11A: Installing TrueType Fonts Locally

How you install fonts depends on who is going to use them. If you have only one user account on your machine, the easiest way is to install the fonts locally. Locally installed fonts are ones that only you or someone logged in to your user account will be able to use. On the other hand, if you have more than one user account and want the fonts to be available to all of the users on your machine, use the method described in Project 11B on page 193.

11A-1: Getting the Font Files

The Internet is awash in free fonts. For this project, I will point you to the www.fontfreak.com site, which has a very nice collection of fonts. Once you get to the FontFreak splash page, click **ENTER SITE HERE**, which will lead you to the main page. Then click the button on the left that says **Fonts**, which will bring you to the first page of fonts. You can browse through the various pages until you find some fonts to your liking. Which fonts you download is completely up to you, but be sure to choose the PC version of the font you want, not the Mac version. Just click the Windows icon to the right of the font sample.

Since you will also need fonts for Project 11B, you might as well download them now too. All in all, for the two projects, you will need at least four fonts. When you've finished downloading, drag the font files to your home folder so it's easy to follow along with my instructions. Also be sure to unzip your font files before going on to the installation steps.

11A-2: Installing the Fonts

For this project, let's use one of the fonts that you downloaded. After you've decided which font to use, follow these steps:

1. Open a new Nautilus window by going to the **Places** menu in the GNOME Panel and selecting **Home Folder**. Once you've done that, open another Nautilus window by doing the same thing.

2. In one of the Nautilus windows, go to the **Go** menu, and select **Location**. A Location input box will then appear below the button bar in that window. You can also achieve the same result by pressing CTRL-L.

3. In that Location box, type `fonts:///`, and press ENTER. Once you've done this, the window should be filled with the many icons representing the fonts you already have installed locally on your system. While you're there, it wouldn't be a bad idea to bookmark this window for future use.

4. From the other open Nautilus window (your home folder window), select the unzipped font you want to install, and drag it to the fonts:/// folder window.

You will notice that when you dragged your new font from your home folder window to the fonts:/// window, the icon for that new font file did not immediately appear. In fact, it will seem as if nothing at all happened and your installation attempt was a bust. Well, despite this lack of system acknowledgment, things were happening. The first time you install a font in this way, your system automatically creates a hidden .fonts folder in your home folder and places the font there. If you don't like this lack of immediate feedback, you can check your .fonts folder by typing `~/.fonts` in the Location box of one of the Nautilus windows. This will show you what fonts you have installed locally—at this point there should be only one.

Now that you have installed your font, you can go ahead and give it a try in one of your applications, such as OpenOffice.org Writer. (Any running applications need to be restarted before the new font will appear in that application's font menu.)

11A-3: An Alternative Approach to Installing Fonts Locally

If you prefer truly immediate feedback on your installation progress, you can try the following variation of the installation process, which yields the same results as in Project 11A-2 on page 191. However, because this installation method uses the .fonts folder that is created in step 4 of Project 11A-2, you must either go through the process in Project 11A-2 with at least one font file prior to trying this installation method, or first create the folder yourself by opening a Terminal window (**Applications ▸ Accessories ▸ Terminal**), typing `mkdir .fonts` in that window, and pressing ENTER. Now that you've got your own .fonts folder, let's use two more of the fonts you downloaded to try out the steps:

1. Assuming that you have already closed the two windows from Project 11A-2, open two new Nautilus windows.

2. In one of the open Nautilus folders, type `~/.fonts`, and press ENTER. The main pane of that Nautilus window should show only one font or, if you bypassed Project 11A-2 and created the .fonts folder yourself, nothing at all. It would definitely be useful to bookmark this location while you're there.

3. In the other open Nautilus window, which is your home folder window, highlight both of the fonts you want to install by holding down the CTRL key and then clicking the icon for each font. Then release the CTRL key, right-click either of the highlighted fonts, and select **Copy Files** in the popup menu.

4. In the .fonts window, right-click on any open space in the window, and select **Paste Files** from the popup menu. The fonts will then appear in the Nautilus window.

Of course, you can just drag the fonts from your home folder to the .fonts folder instead of going through the procedure in steps 3 and 4. It's strictly up to you. Either way, the procedure outlined in Project 11A-3 yields exactly the same results as that in Project 11A-2. The only difference is that you will immediately see that your fonts have been copied to the .fonts folder. Which method you choose in the future is, therefore, strictly a matter of personal preference.

11A-4: Uninstalling Locally Installed Fonts

Whether you used the installation method outlined in Project 11A-2 or Project 11A-3, you can uninstall any fonts installed locally by opening the .fonts folder in your home folder and then dragging the fonts you want to remove to the Trash.

Project 11B: Installing TrueType Fonts Globally

As I mentioned already, the fonts you have installed thus far can only be used by you when you log in under your usual username. If, however, you want to install fonts that can be used by you and anyone else who has an account on your computer, the process is slightly different. We'll use only one of the fonts you downloaded for this part of the project.

11B-1: Installing Individual Fonts Globally

The font folder for globally installed fonts is in root territory, so you will need to put the `sudo` command to use in order to install fonts in this way. Here are the steps:

1. Open a Terminal window, and then create a new folder for your fonts within the global location (let's call it MyFonts) by typing `sudo mkdir /usr/share/fonts/truetype/MyFonts` and pressing ENTER.

2. Type your password when prompted to do so, and press ENTER.

3. Copy the font you want to install globally to your newly created global font folder by typing `sudo cp fontname.ttf /usr/share/fonts/truetype/MyFonts` and pressing ENTER. Be sure to include spaces on both sides of `cp` and after `.ttf`.

Be sure to include spaces on both sides of *fontname*.`ttf`. Also, be sure to use the name of your font in place of the word *fontname*. For example, if your font is called arachnid, you would type `sudo cp arachnid.ttf /usr/share/fonts/truetype/MyFonts`.

You shouldn't have to enter your password after step 3 since you already provided it in step 2, so the process is now complete. That being the case, go ahead and test things out by opening OpenOffice.org and looking for the font in the font menu. Remember that you will need to restart OpenOffice.org if it was already open when you installed the font.

11B-2: Installing Multiple Fonts Globally

If you want to install several fonts globally at the same time, you can do so quite easily. The process is essentially the same as in Project 11B-1 on page 193, with a slight variation. Here's what you've got to do:

1. Create a new subfolder in your home folder window. You can call it anything you like, but I'll be using the name *fonts2go* in this project.
2. Unzip the fonts and then place the ones you want to install via this method into the new fonts2go folder.
3. Open a Terminal window, and then copy all of the fonts in your fonts2go folder to your new global font folder by typing **sudo cp fonts2go/*.ttf /usr/share/fonts/truetype/MyFonts** and pressing ENTER. Note the spaces surrounding cp and after *.ttf.
4. Type your password if prompted to do so, and press ENTER.

NOTE *In step 3, you did not need to type the names of the fonts, as the wildcard character (*) was essentially telling your system to copy all files ending in .ttf within the fonts2go folder.*

Now that you are done, check your results in OpenOffice.org. You should also back up the font files in the fonts2go folder to CD or floppy (just in case you need to install them on another system) and then trash the files. Keep the folder, however, so that you can use it again in the future.

11B-3: Uninstalling Globally Installed Fonts

If you want to remove any fonts that you installed using either of the global installation methods introduced in this project, you can do so in the following manner:

1. Open a Terminal window, type **sudo rm /usr/share/fonts/truetype/MyFonts/*fontname*.ttf**, and press ENTER. Be sure to replace *fontname* with the name of the font you are installing.
2. Type your password, and press ENTER when prompted to do so.

Project 11C: Installing Microsoft Windows Core Fonts via Synaptic

Like it or not, the computing world is still pretty much a Microsoft world, and that means that the vast majority of users, even Mac users, are using Microsoft core fonts. That being the case, it is inevitable that you will have to deal with documents using fonts such as Georgia, Verdana, Times New Roman, and Courier, to name a few. Of course, your system can substitute the fonts it has for those used in the document, but in order for you to see things as they were intended and to allow others to see your documents the way you intended (web pages are good examples), it will probably behoove you to install those Microsoft core fonts on your own system.

Fortunately for you, there are two ways to get these fonts. One is to download and install them via Synaptic, while the other, for those of you with a dual-boot setup, is to simply copy them from your Windows partition. In this project, I will explain how to perform the first of these procedures, so if you're a Windows-less Ubuntu user (or a dual-booter who thinks the Synaptic approach is easier), here's what you need to do:

1. Run the Synaptic Package Manager (**System ▸ Administration ▸ Synaptic Package Manager**), and provide your username when prompted.
2. Click the **Search** button. In the Search window that appears, type `msttcorefonts`, and then click the **Search** button in that window.
3. The msttcorefonts package should now appear in the list pane of the main Synaptic window, so click on the package name, select **Mark for installation** in the popup menu, and then follow the standard procedures you learned in Chapter 5 for installing a package via Synaptic.

When the process is complete, your new Microsoft fonts will have been successfully installed and ready for immediate use by every user account on your machine.
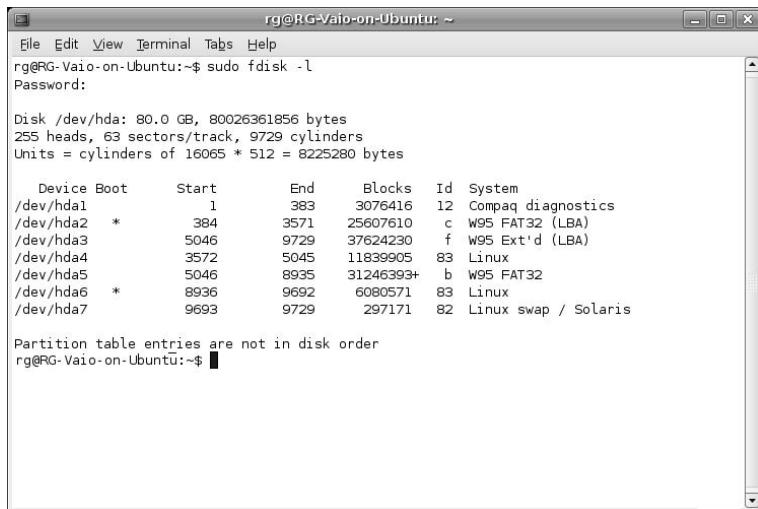
Project 11D: Installing Microsoft Core Fonts from Your Windows Partition (for Dual-Booters)

If you're a dual-booter with Windows installed on another partition of your hard disk, you can take advantage of the fonts you already have on your Windows partition. This can be as easy as what you did in Projects 11A and 11B or a bit more difficult, all depending on whether or not you installed the *Automatix read/write NTSF and FAT32 Mounter* in Chapter 9. If you did install it, your Windows partition should already be mounted and you can just scoot right down to Project 11D-3.

If you did not install the Automatix Mounter, you can either go back and do so now, or you can mount your Windows partition manually. If you are going to go this more cumbersome manual route, continue on to 11D-1.

11D-1: Finding Your Windows Partition

Before you can actually mount your Windows partition, you need to find out where it is, and in what filesystem it is formatted, either File Allocation Table (FAT) or New Technology File System (NTFS). As is so common in Linux, there are a couple of ways to do this. If you are a command lover, you can check things out by opening a Terminal window, typing **sudo fdisk -l** (that's the lowercase *L* there, not a numeral *1*), and then pressing ENTER. You will be prompted for your password, so type it, and press ENTER. A list of the partitions on your hard disk will appear (Figure 11-3).

```
                     rg@RG-Vaio-on-Ubuntu: ~
File  Edit  View  Terminal  Tabs  Help
rg@RG-Vaio-on-Ubuntu:~$ sudo fdisk -l
Password:

Disk /dev/hda: 80.0 GB, 80026361856 bytes
255 heads, 63 sectors/track, 9729 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/hda1               1         383     3076416   12  Compaq diagnostics
/dev/hda2   *         384        3571    25607610    c  W95 FAT32 (LBA)
/dev/hda3            5046        9729    37624230    f  W95 Ext'd (LBA)
/dev/hda4            3572        5045    11839905   83  Linux
/dev/hda5            5046        8935    31246393+   b  W95 FAT32
/dev/hda6   *        8936        9692     6080571   83  Linux
/dev/hda7            9693        9729      297171   82  Linux swap / Solaris

Partition table entries are not in disk order
rg@RG-Vaio-on-Ubuntu:~$ ▮
```

Figure 11-3: Finding your Windows partition in the fdisk *table*

Take a look at this seemingly incomprehensible list, and then find the location and format of the Windows partition on your hard disk by scanning the column on the right for the first entry with the words FAT32 or NTFS. In most cases, this will be the first partition, /dev/hda1, but in the list for my hard disk, shown in Figure 11-3, the first partition is taken up by some diagnostic utility that the manufacturer's Windows install disk had on it. If you're using a computer from a brand-name manufacturer, you might find yourself in a similar situation.

Once you've found the first FAT32 or NTFS partition, jot down the file-system type, and then do the same for its location at the far left of the list. In my case, that would be /dev/hda2 and FAT32.

11D-2: Mounting Your Windows Partition

Armed with the information you just gathered, you can now mount your Windows partition by doing the following:

1. Create a mount point for your Windows partition by opening a Terminal window, typing **sudo mkdir /media/windows**, and pressing ENTER. You only need to perform this step the first time you try to mount your partition. After that, you need only perform steps 2 and 3.

2. What you do in this step depends on whether your Windows partition is FAT32 or NTFS formatted. Choose the appropriate step, being sure to substitute /hda2 for /hda1 if that is what your Windows partition happens to be.

If your Windows partition is in FAT32 format, type

```
sudo /dev/hda1 /media/windows/ -t vfat -o iocharset=utf8,umask=000
```

If it is in NTFS format, type

```
sudo /dev/hda1 /media/windows/ -t ntfs -o nls=utf8,umask=0222
```

3. Once you've typed the appropriate entry, press ENTER. If asked for your password, type it, and press ENTER. The Windows partition will then be mounted. You can double-check by opening a Nautilus window and clicking the **Computer** button. You should then find a hard disk icon with the word *windows* beneath it. You can even double-click that icon to access the files you have there, but you cannot save files to the partition if it is in NTFS format (a good reason for installing the Automatix Mounter is that it allows you to do so).

11D-3: Installing Fonts from Your Windows Partition

Now that your Windows partition is mounted, you can copy the fonts you have there and install them for use within Linux. To get started, go back to the Terminal window, type **cd**, press the spacebar, and then type the path to your Windows font folder. Most likely, the path to your font folder will be /mnt/windows/WINDOWS/Fonts, but it might be slightly different (/mnt/windows/ windows/Fonts, for example) depending on the version of Windows you are using. Once you've typed what you need to type, press ENTER.

Depending on whether you want to install your Windows fonts locally or globally, the next steps will be different.

Installing Windows Fonts Locally

To install your Windows fonts locally, just copy the font that you want to install to your .fonts folder by typing **cp** *fontname***.ttf ~/.fonts/** in the still open Terminal window and pressing ENTER. Be sure to replace *fontname* with the name of the font you want to install.

If you want to go wild and copy all of the fonts in your Windows font directory (that's a lot of fonts, mind you!), you can type ***.ttf** instead of *fontname*.ttf. Be sure that you have a space between cp and *fontname*.ttf and between *fontname*.ttf and ~/.fonts/. Once you're done, you're done.

Installing Windows Fonts Globally

To install your Windows fonts so that all users on your machine can use them, just return to the still open Terminal window. Then copy the font you want to the personal system font folder you created in step 1 of Project 11B-1 on page 193 (and if you didn't perform that step then, do so now) by typing

sudo cp *fontname***.ttf /usr/share/fonts/truetype/MyFonts**. If you want to copy and install all of the fonts in your Windows font directory, you can do so by typing **sudo cp *.ttf /usr/share/fonts/truetype/MyFonts** instead. After typing either of those strings, press ENTER, after which you will probably be prompted for your password, so type it, and press ENTER again. The job will then be done.

Unmounting Your Windows Partition

If you mounted your Windows partition manually, you can unmount it immediately after finishing your work with it, or you can just wait until you shut down your system, whereupon it will be automatically unmounted. Before you can unmount it yourself, you must first be certain you aren't accessing the directory, either from the Terminal window or from Nautilus. So, before issuing the unmount command, close any Nautilus windows open to that partition or, if you are using the Terminal, type **cd** to change back to your home directory. Type **sudo umount /media/windows** (that's *u-mount*, not *un-mount*), and press ENTER.

Customizing Your System Fonts

Now that you know how to get and install fonts, you might be itching to use some of them to further customize your system. To get started doing this, just select **Preferences ▸ Font** in the **System** menu, which will bring up the Font Preferences window, shown in Figure 11-4.
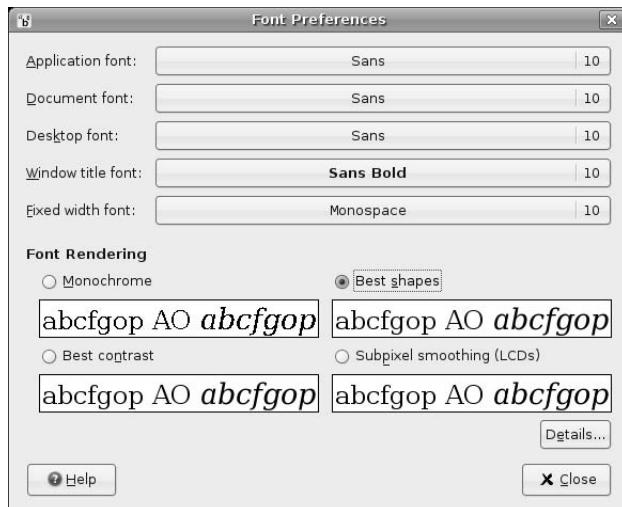
Figure 11-4: Setting system font preferences

As you can see, you can specify font preferences in five categories: applications, documents, desktop, window titles, and the Terminal (fixed width font).

The choices you make take effect immediately, so you will soon know whether or not you can live with them. Unlike the other aspects of customization, those choices could drive you stark raving mad. Sure, it is easy and fun to live with the gaudiest desktop imaginable, the wildest and most mismatched color scheme on the planet, and the goofiest icons ever to be seen by post-Neanderthal man, but if your font selections get too out of hand, watch out! You do have to be able to read the results, after all.

Making Things Look Better

The fonts you see on your screen generally look quite smooth and clean. If you are using an LCD monitor, however, you may find that fonts will look even better if you select **Subpixel smoothing (LCDs)** in the Font Preferences window. If you're not sure whether you need to do this, just give it a try, and see if you notice any difference. GNOME applies changes immediately upon selection, so if you keep a window with text in it open behind the preferences window, you can easily see the effect of each of your selections as you make them.

Creating Your Own Fonts with FontForge

After all this font talk, it seems only appropriate that I end this chapter by giving you the means to create your own fonts (or at least modify someone else's). FontForge (previously known as PfaEdit) allows you to create or modify TrueType, PostScript, and bitmap fonts (see Figure 11-5). The interface itself seems a bit dated, but don't let that fool you; FontForge is quite capable and easy to use.
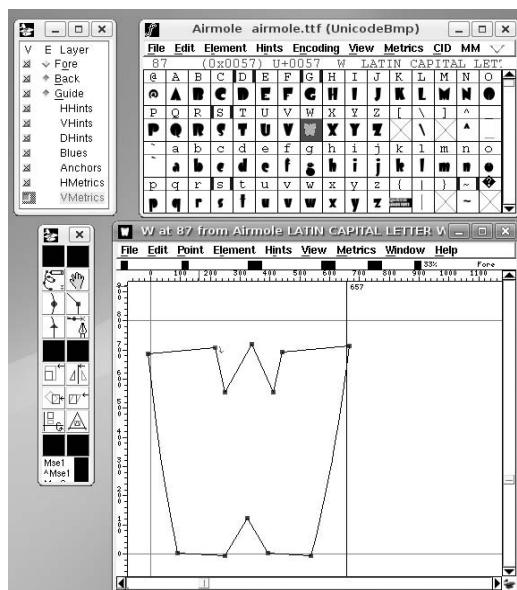
Figure 11-5: Using FontForge to create and modify fonts

Downloading, Installing, and Running FontForge
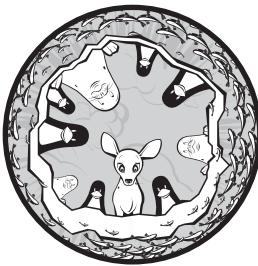
You can easily download and install FontForge via Synaptic. Just do a search for *fontforge*, mark the **fontforge** and **fontforge-doc** packages for installation when they appear in the list of search results, and then follow the usual procedure for installing packages via Synaptic. You can run FontForge by going to the **Applications** menu and selecting **Graphics ▶ FontForge**. An Open Font window will appear, in which you can select the font you want to work with. Make sure that it is not an installed font, but rather a font located within the home folder itself, or within another visible folder.

After that, I think I had better leave you to your own resources (and those of others). Font creation and modification is, after all, too complex a topic for this book, but those of you already into this stuff probably just needed a pointer to the right tool and not my meager instructions. If you do need more than a pointer, the FontForge help files are a good place to start, as is the FontForge project page at http://fontforge.sourceforge.net.

12

POLYGLOT PENGUINS

Linux Speaks Your Language

These days, almost all operating systems are multilingual, or at least capable of becoming so. This is true of Linux as well. Just open your web browser, and, without performing any special installations, you can read pages in any European language, including those with Cyrillic alphabets, such as Russian. You can even view pages in Chinese, Japanese, Thai, Arabic, and Hebrew, to name but a few.

But the multilingual capabilities of Linux are much greater than this, and the way that GNOME-based distros, such as Ubuntu, handle multilingual matters makes it quite easy to take advantage of these capabilities. As you will soon see, you can even set up your system to give you a totally foreign language environment, allowing you to function completely in the language of your choice. Add to this the ever-expanding number of free programs available for language study, and you have a truly meaningful language-learning tool.

Read-Only Language Support

If all you want is to be able to read web pages or documents written in a foreign language, you don't need to install any additional language support except in some rare cases. From the get-go, you will be able to view documents in just about any language you happen to throw at your system—doesn't matter if it's Swedish, Italian, Chinese, Japanese, Arabic, Hebrew, Russian, Vietnamese, Armenian, or Thai. You will be able to read whatever you are linguistically capable of reading (see Figure 12-1 for an example).

Figure 12-1: A Japanese web page displayed in Firefox

Changing the Character Encoding in Firefox

Firefox usually automatically recognizes the language in which a web page is written and thus displays the page correctly. Sometimes, however, the author of the page may neglect to include the character coding for that page in the HTML, in which case Firefox, not knowing that the page is prepared in another language, will open it in the default language of your system. The result is a page in which you see nothing but odd combinations of symbols and letters that have no meaning. In this case, try changing the character coding in Firefox to the language encoding you believe the page to be in. Some languages employ more than one encoding scheme, so if you're not sure, give each one a try. You can make your choices by going to the Firefox **View** menu and selecting **Character Encoding**. From the submenu there, you can select the appropriate coding for the language of that page.

Typing Nonstandard Characters

Typing characters that are not standard in English, such as é, ç, ß, ø, æ, and å, can be done quite easily in Linux without any modifications. In most situations, you can do this by using the Character Map utility included in your

system, which can be found at **Applications ▶ Accessories ▶ Character Map**. Upon running Character Map, a window like the one shown in Figure 12-2 will appear.
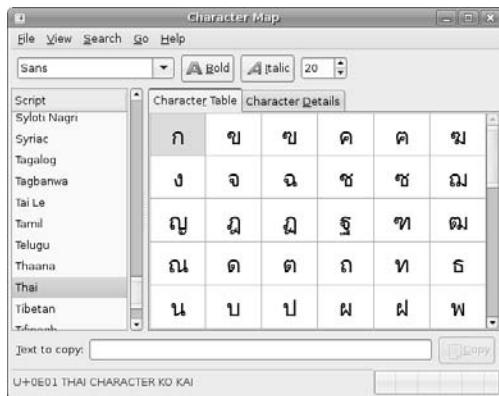
Figure 12-2: Inputting characters with the Character Map utility

To input the character you want, just select the language or character set in the left pane of the window, and then, in the right pane, click on the character you want to input. The character will appear in the little input box next to the words *Text to copy* at the bottom of the window. Just click the **Copy** button, and then paste the character wherever you want to place it.

In certain applications, such as the various OpenOffice.org modules, this method will not work. In these cases, the program usually offers a method of its own. In OpenOffice.org, for example, you can click **Insert** in the menu bar and then select **Special Character**. A selection window will open, and you can select the character you want there. Once you've done that, click the **OK** button, and the character will appear in your document, after which the selection window will close by itself.

Using the Compose Key Option

If you only need to type an accent or umlaut once in a while, and don't feel particularly keen on opening an application or going to a special menu to do so, using the *compose key option* for your keyboard is a good way to go. Basically what this means is that you use one of the lesser-used keys on your keyboard in conjunction with six symbols (` , ' ~ " ^) to help in the creation, so to speak, of accented characters. By default, the compose key is the right ALT key.

Let's say, for example, that you wanted to type an umlauted u (ü). While pressing the right ALT key, you would press ". Then you would release the right ALT key, press u, and . . . voilà, you'd have yourself an ü. Here are some more examples:

á	right ALT + ' then a	**ñ**	right ALT + ~ then n
ç	right ALT + , then c	**ô**	right ALT + ^ then o
è	right ALT + ` then e		

Using the Keyboard Indicator

If you often type in a particular foreign language, it might be more convenient for you to use the Keyboard Indicator GNOME Panel utility. This utility lets you switch quickly among various keyboard layouts. For example, if you often type in Swedish, and thus use the characters å, ä, and ö regularly, using the appropriate keyboard layout would be easier than repeatedly using the Character Map. Of course, you will have to familiarize yourself with the keyboard layout, or *keymap*, for each language you choose, but this is a relatively easy task.

The Keyboard Indicator is already included in your system, so there is no need to install it. To access it, simply right-click somewhere on the GNOME Panel where you would like to place a launcher for it. Then, from the popup menu select **Add to Panel**. When the Add to Panel window appears, scroll down and click **Keyboard Indicator**, and then click the **Add** button. The letters *USA* or *GBr* will then appear on the panel.

Now, this alone will give you nothing except your default keymap, so you must configure Keyboard Indicator if you want to be able to use other keymaps. To do so, just right-click the icon, and from the popup menu select **Open Keyboard Preferences**, which will open the Keyboard Preferences window.

To add a keymap, click the **Layouts** tab in that window, and then once in the new tab, click **Add**. After you do this, the Choose a Layout window will appear with a list of the keymaps available to you (Figure 12-3). In the left pane of that window, scroll down to the keymap for the country of the language you want to add, and then click it to select the standard keyboard layout for that locale (click the arrow next to the country for a list of optional layouts if you prefer). Once you've made your selection, click the **OK** button, which will bring you back to the Keyboard Preferences window, where you will find your new keymap listed along with the original. You can then click the **Close** button to complete the setup.
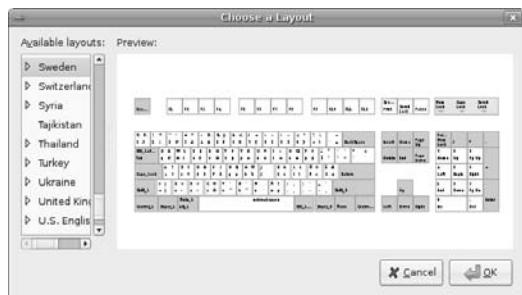
Figure 12-3: Adding keyboard layouts to the Keyboard Indicator

After closing the Keyboard Preferences window, the keyboard layouts can be changed by clicking the icon in the panel. You can also change between keyboard layouts by going to the **Layout Options** tab of the Keyboard Preferences window and assigning a keyboard shortcut. If you are not sure where

the keys you want to type are located in your new keymap, switch to the key-map in question by clicking the panel Keyboard Indicator until it appears and then right-clicking it. In the popup menu that appears, select **Layout View**, after which a map of the new layout will appear in a separate window. The default window size is quite small, but it can be resized in the usual drag-the-corner manner.

If you just want to choose a single keyboard layout to replace your present one (such as British English instead of American English or German instead of Spanish), you can do so from the Keyboard Preferences window by adding the keyboard layout you want to use, checking the box next to the word *Default*, and then removing the original layout by clicking it once to select it and clicking the **Remove** button.

Viewing Your System in Another Language

One of the many things that originally attracted me to the Linux world was being able to install language support for languages other than English. On one of my machines, I have installed support for Chinese, Japanese, Swedish, and my default, English. With just a simple logout and a few more clicks, I can log back in with an interface in a totally different language. I can have a Chinese, Japanese, or Swedish system whenever I want.

This is very useful if you are going to be doing a lot of work in a foreign language, or if you are studying a foreign language and want to give yourself as much exposure as possible to it. It is also very handy when you have users with different native languages using the same machine. At my former university, for example, where my Japanese and Chinese students sometimes used my computer, the additional language support allowed them to log in using their own language. All in all, it is a very useful feature.

Taking advantage of this feature in the GNOME environment is very easy and, depending on the language you wish to use, requires little in terms of special installation measures. Basically all you have to do is install a group of support files for each language you want to add to your system. You can do this quite easily by going to the **System** menu and selecting **Administration ▸ Language Support**.

You will be prompted for your password at this point, so provide what is required, and the Language Support window (Figure 12-4) will appear. In that window, check the boxes next to the languages for which you want to install support. Once you have made your selections, click the **Apply** button, and the Language Selector will begin downloading and then installing the support packages you specified.

Figure 12-4: Installing additional language support

Language Support is another graphical front end for APT, and that being the case, be sure to close all other APT-related tools before using it.

Multilingual Login

Once you have installed support for any additional languages you want, you can log out of your current session and log in to a new one in a different language environment. The actual switch is made from your login screen.

Just below the text box where you would normally type your username, click the link that says **Language**, after which a window with a list of the languages you have installed will appear. Select the language you want to use in the next session by clicking once on its name and then clicking the **OK** button. The language list window will then close automatically, and you can proceed as usual by typing your login name and password (pressing the ENTER key after each, of course). A little window will then pop up asking you if you want to use the language you've chosen as your default language forever and ever. You can either click **Just for This Session** or **Make Default** depending on your personal preferences. Whatever you do end up choosing, fear not; it's no big deal to change back at a later point in time by logging out of your system, logging back in to your system under your previous default language, and clicking **Make Default**.

Your startup process will then continue, and everything will progress as it usually does. Depending on what language you've chosen, once your desktop appears you will be in another linguistic world. Your menus, applications, and even the little Tips windows that pop up when you run your mouse over a panel icon will all be in the newly selected language. (Figure 12-5 shows the System menu in a variety of languages.)

Figure 12-5: The Applications and Sound & Video menus shown in Japanese

Chinese, Japanese, and Korean Input

For most European languages (and many other alphabet-based non-European languages), pressing a letter on the keyboard simply prints that letter to the screen. However, Chinese, Japanese, and Korean require a kind of conversion process that is handled by a special application (actually

a set of applications) called an Input Method Editor (IME). Of course, this is a Windows-world term, but I will use it here for convenience's sake. In any case, each of these languages has its own IME, and each is quite different due to the basic differences in the three writing systems.

Chinese

While most people (at least those in the linguistic know) would think that Chinese would be the most complicated system, because the writing system consists of thousands of characters, it is in fact the simplest. The Chinese IME simply takes the romanized keyboard input, known as *pinyin*, and converts it into Chinese characters, or *Hanzi*. For the IME, it is essentially a simple dictionary lookup task—big dictionary, simple IME. In the event that there is more than one character for the pinyin input, a list of possible candidates will appear, and the user can then simply select the appropriate character from that list.

Japanese

The Japanese IME has a considerably more complicated task to perform, as it has three writing systems to deal with: *Kanji* (ideographic characters borrowed long ago from China), *hiragana* (the phono-alphabetic system used mainly for tense and case endings), and *katakana* (used mainly for words borrowed from other languages). Still, the standard input method for Japanese is primarily via the standard Roman keyboard layout, plus a few extra special-function keys. Thus, typing in Japanese is a two-step process whereby the IME first converts the romanized text into hiragana as it is typed and then converts it to appropriate Kanji, katakana, or hiragana elements after the spacebar is pressed.

You can see an example of these steps in Figure 12-6. In the first line, the IME has already converted the romanized input on the fly. It has converted `rinakkusdenihongonyuuryokumodekimasu` (which means *You can also input Japanese in Linux*) to hiragana. The fact that line is underlined means that it has not yet been converted beyond that. In the second line, however, the user has subsequently pressed the spacebar, which caused the IME to convert the hiragana string into the appropriate Kanji, hiragana, and katakana elements. The first word, *Linux*, has been converted to katakana text, as it is a borrowed word, while *Japanese input* has been converted to Kanji; the rest stays in hiragana.

りなっくすでにほんごにゅうりょくもできます。

リナックスで日本語入力もできます。

Figure 12-6: IME conversions while typing in Japanese

Korean

The job of the Korean IME is again quite different from that of the Chinese and Japanese IMEs, as the language itself is written in a very different way. Korean is written either entirely in alphabetic letters, called *Hangul*, or in a combination of Hangul and ideographic characters borrowed from Chinese called *Hanja*. While the Hanja characters are essentially the same as their Chinese and Japanese counterparts, Hanzi and Kanji, the Korean phonetic alphabet, Hangul, has it own unique appearance, as you can see in the Korean word for Korea, *Hangug(k)*, in Figure 12-7.

Figure 12-7: Korea (Hangug) written horizontally in Hangul

This seems simple; however, the representation is not quite correct, as Korean is very unique in the way that its alphabetic characters are put to the page. Unlike the usual side-by-side positioning of hiragana, katakana, and most other languages written with an alphabet, Hangul letters are grouped in pairs, triplets, or even quadruplets, which are written, as a general rule, clockwise. The IME, therefore, must take the input (usually based on a Korean alphabetical keyboard layout) while it is being typed, and it must adjust the size, spacing, and positioning of each of the letters as it puts them into appropriate clusters (see Figure 12-8).

Figure 12-8: An example of the clustering process in the Korean IME

Project 12: Installing Asian Language Input Support for SCIM

So what do you do if, for example, you want to be able to type Chinese, Japanese, Korean, Hindi, Thai, or Nepali while still in your usual English environment? What if you want to be able to type all of those languages in the same document? Can you do it?

You bet.

There are actually several ways of going about this, but the one of the most straightforward to use is Smart Common Input Method (SCIM), the base for which comes preinstalled in Ubuntu. SCIM supports most Asian languages, including Chinese, Japanese and Korean, and it provides a number of input methods for many of these.

12-1: Downloading and Installing SCIM Input Method Modules

Despite coming preinstalled on Ubuntu, SCIM will not be able to do anything for you right out of the box. In order to enable SCIM so you can input CJK and other similar types of characters, you will need to install a package called im-switch. Then you will need to install the appropriate input modules for the language you wish to type in.

The easiest and least problematic way of getting these packages is via the Language Support utility. Just go to the **System** menu, select **Administration ▸ Language Support**, and then install support for the languages you want.

After the installation of the language support modules is complete, there is still one simple step left. Just log out, and then once you're back at the login screen, log back in. A SCIM applet will appear at the right side of the top panel, letting you know that SCIM is ready for action.

12-2: Typing in Asian Languages with SCIM

To get a feel for how SCIM works, try using it with Gedit (**Applications ▸ Accessories ▸ Text Editor**).

Once Gedit appears, click the SCIM panel applet. A list of the available languages that can be handled via SCIM will then appear in a drop-down menu (Figure 12-9). Remember, however, that the number of languages appearing in that menu will be dependent on the number of language packages you installed.

Figure 12-9: A list of the languages
that can be handled by SCIM

From that list, select the language you want to use, and then, for those languages where you are given a choice, select the input method you prefer. The icon for the SCIM panel applet will reflect the change by showing the icon for the language you have chosen. You can then start typing.

At this point, a small input palette like the one in Figure 12-10 will appear at the bottom-right corner of the screen. This palette, in addition to displaying the language and input method currently in use, also allows you to easily switch between English and the current language of input, or to switch from the language in which you are currently typing to another. By clicking the red button at the far-right end of the palette, you can also find out what the hotkeys are for the input method you are currently using.

Figure 12-10: The SCIM input palette

Regardless of the input method you are dealing with, probably the most important hotkey combination you will want to use is SHIFT-spacebar, which toggles you back and forth between English and the language you currently have selected in SCIM.

NOTE *You must be working with a document or application that accepts text input, or no menu will appear when you click the SCIM applet.*

12-3: Installing Additional Input Methods

You may find that the input methods supplied for the languages you have chosen may include some that you are not familiar with. For example, the default Korean set is not based on the standard keyboard (the only one I'm familiar with), and it does not have Hangul-to-Hanja conversion capabilities like the hiragana-to-Kanji ones included by default in the Japanese module. Fortunately, this is easily remedied by using Synaptic to install additional input modules. Which package you download depends, of course, on the language in question. To add additional modules for Korean, install scim-tables-ko. For additional Japanese input methods, install scim-tables-ja. Finally, for additional Chinese modules, install scim-tables-zh.

You can also use SCIM to conveniently type in other Asian languages that do not require special conversion routines, such as Thai, Hindi, Telugu, Bengali, and Punjabi. To get input support for these languages, install scim-tables-additional.

13

PENGUINS BACK AT WORK

Getting Down to Business in Linux

I have to admit it: When I think about the joys of computing, I tend to think of the more hedonistic, self-indulgent areas like gaming, music, and graphics. Still, as is the case for most computer users, what I usually end up doing on my computer is work, and writing this book falls into that category.

Fortunately, Linux can get down to business and do it as well as the next OS. I think it's safe to say that you are missing nothing and are probably gaining quite a bit in terms of home and office productivity programs in the world of Linux. In this chapter, I'll walk you through the Linux offerings in this department.

OpenOffice.org

Whether they should be or not, people are quite obsessed with office suites, even though most people seldom need more than a word processor. The de facto standard among office suites is Microsoft Office, which is available in both the Windows and Macintosh worlds. Of course, as it is a Microsoft product, you can be quite sure that there is no Linux version available.

Fortunately, Linux does have an exceedingly capable office suite in the form of OpenOffice.org, which is, incidentally, also freely available (as in *free*) in Windows and Mac OS X versions. OpenOffice.org is not some lightweight sour-grapes substitute for the Microsoft Office–less Linux world; it is a full-featured contender, and in some cases, OpenOffice.org is a clear winner.

OpenOffice.org Applications

The entire OpenOffice.org office suite consists of six applications: a word processor (Writer), a spreadsheet (Calc), a presentation creator and player (Impress), a vector drawing program (Draw), a database (Base), and a mathematical formula editor (Math). Most of these can be accessed at **Applications ▸ Office**.

Since giving full and detailed instructions on how to use each of these applications would take up an entire book (and there are entire books on the subject), I will simply introduce each module to you. Being a bit of a mathematics dunce, however, I won't be venturing too deeply into what you can do with Math. That said, here goes.

Writer

As I mentioned earlier, the word processor is the office application that the majority of users turn to most often. Fortunately, OpenOffice.org Writer is a good one (see Figure 13-1). It is chock-full of features and can read and save Microsoft Word files. Like Word, it will even let you save your documents as HTML files so that you can easily change your documents into web pages.
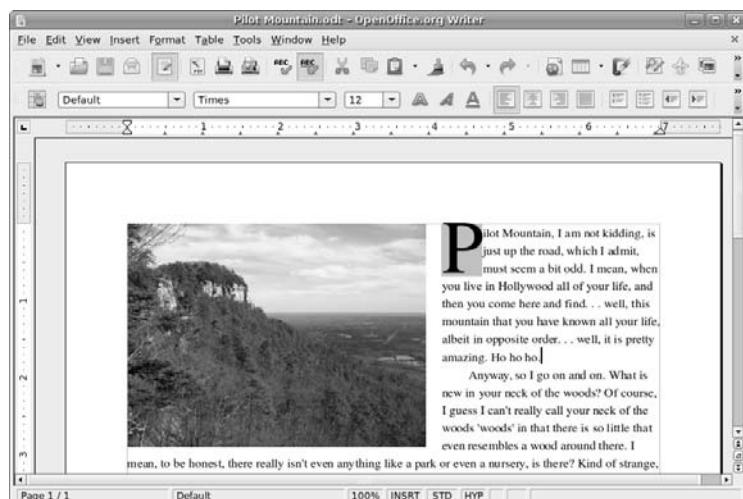
Figure 13-1: OpenOffice.org Writer

As I said, Writer is a very straightforward word processor, so I won't go on about it, but if you would like an introduction to using Writer, check out the First Steps tutorial at the OpenOffice.org website (www.openoffice.org/writerfirststeps/writerfirststeps.html).

Calc

Calc is the OpenOffice.org spreadsheet application, and it is similar to Excel in terms of capabilities and general layout (see Figure 13-2). It can also, quite importantly, read and save Microsoft Excel files.
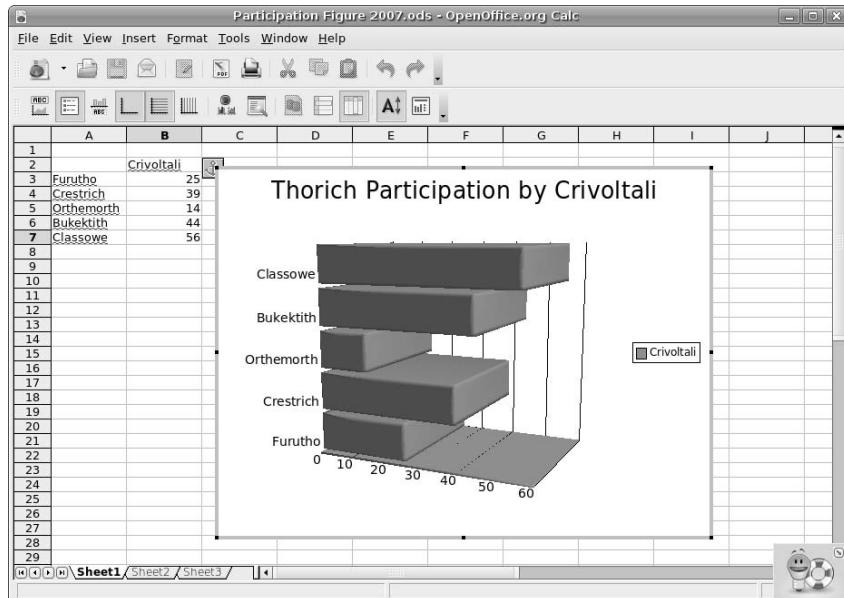
Figure 13-2: A graph created in OpenOffice.org Calc

Since most people who use spreadsheets generally understand what they are all about and, after a bit of poking around, can figure out how to use them, I won't go into any sort of primer about using Calc. However, as there are many others who don't see any need to even try using spreadsheets, I will mention a few of the simpler things that can be done with Calc, in the hope of enticing some of you into trying it.

Most people who don't use spreadsheets think of them as a sort of giant calculators used for computing uncomfortably large sets of numbers, like payrolls (which was the original purpose of such applications). That's right, of course, but spreadsheets can be used for everyday tasks too, such as projecting household budgets, calculating grade point averages (by teachers or students), figuring out how long it will take you to save up for your trip to Hungary, or even for something as weird as comparing the seat heights for the four or five motorcycles you are trying to choose among. And when doing any of these minor mathematical tasks, you can easily create graphs in order to make all the abstract numbers speak to you visually.

If numbers are just not your thing, you can still use Calc for creating lists of information, such as birthday lists, class rosters, shopping lists, address lists . . . whatever. You can even have Calc put the lists into alphabetical order, or sort them by date of birth, and so on. Everyone eventually seems to find some use for Calc, so don't ignore it entirely.

Impress

Impress is OpenOffice.org's answer to Microsoft's PowerPoint, with which it is compatible. It allows you to create graphically attractive slides for use in presentations and also allows you to create notes or handouts to accompany them. While these features make Impress quite handy in business and education settings, you may not find as much value in it as a home user.

Draw

More useful to the home user is OpenOffice.org Draw. Although Draw isn't all that great a program for creating true graphics in the artistic sense, it is very useful for creating flowcharts, organizational diagrams (like seating arrangements for wedding receptions or conferences), or any other document in which you want a bit more control over the placement of text and graphics (especially when the two are combined), such as for fliers, awards, diagrams, and newsletters. In this sense, Draw can be used quite effectively as a simple page layout program, as you can see in Figure 13-3.
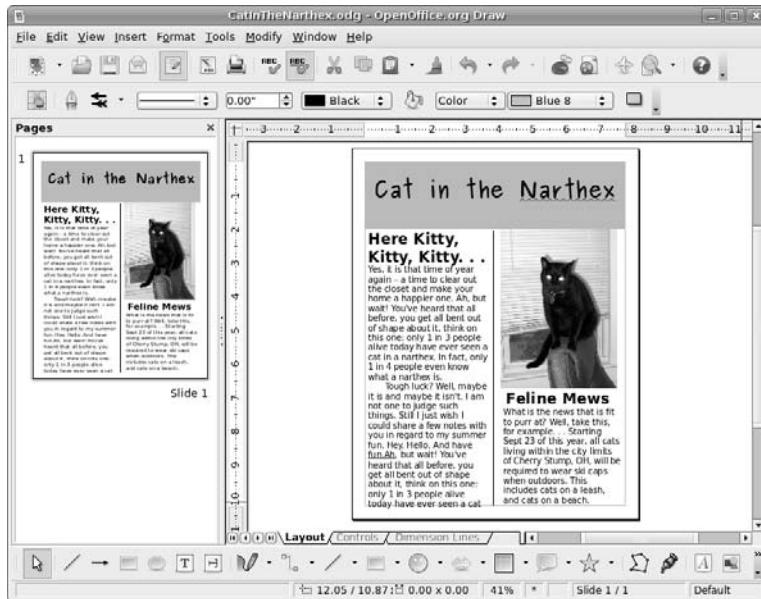
Figure 13-3: A newsletter created (and displayed) in OpenOffice.org Draw

Although Draw comes bundled with Ubuntu, it is no longer listed in the Applications menu, probably because its functions can be accessed through the other OpenOffice.org modules. Still, it is a handy application to use on its own, so if you would like to use it in its stand-alone form, you can run it

either via the Run Application panel applet by typing `ooffice -draw` or by adding the launcher yourself to the Applications menu.

To add Draw to the Applications menu, open the Menu Editor by right-clicking **Applications** and then selecting **Edit Menus**. Once the editor window appears, click **Graphics** or **Office** (or both) in the left pane, depending on where you want the menu launcher to appear. In the right pane of the window, check the box next to the words *OpenOffice.org Drawing*. Once you're done, click the **Close** button, and the Draw launcher will then appear in the menu.

Base

New to OpenOffice.org is Base, which is a graphical front end to any databases you have on your machine, including Base's own database, called HSQL. Unfortunately, learning how to use Base is no cakewalk if you don't have any experience working with database applications, so if the database module in AppleWorks is the extent of your contact with such applications, you might want to try out a relatively easy-to-follow tutorial (with a not-so-easy-to-follow URL) at http://searchopensource.techtarget.com/tip/ 1,289483,sid39_gci1148271,00.html.

Math

Finally, we come to OpenOffice.org Math. As I am a bit out of it in terms of math, I can't really tell you much more about Math other than that it is a mathematics formula editor that allows you to produce complex mathematical formulae and then paste them as embedded objects into other OpenOffice.org programs. It is strictly an editor and does not calculate. I think I am safe in saying that most common folk have no need for it in their everyday lives, unless, of course, you write math textbooks for a living.

Like Draw, OpenOffice.org Math is not listed in the Applications menu, but you can access it via the Run Application panel applet by typing `ooffice -math`. Oh, and if you do happen to write math textbooks for a living, you can also add a launcher for Math to your Applications menu, just as you did for Draw, by means of the Menu Editor. In the open editor window, just click **Office**, and then click the **New Item** button. In the Create Launcher window that appears, type `OpenOffice.org Math` in the Name box, and `ooffice -math` in the Command box. Once you're done, click **OK**. The new item will then appear in the Menu Editor window, selected and ready to go.

Microsoft Office and OpenOffice.org File Compatibility

Although I mention the point throughout this section, it is worthwhile to re-emphasize that OpenOffice.org can read and write Microsoft Office files. This compatibility is quite good, though tables sometimes prove slightly problematic.

In order to read Microsoft Office files, all you need to do is double-click the file in question, and it will open in the appropriate OpenOffice.org module. When saving files within OpenOffice.org to use within Microsoft Office, however, you must save them into the appropriate format, as

OpenOffice.org will otherwise save files into its native format (.odt for Writer documents, .ods for Calc documents, and .odp for Impress documents) by default.

To do this when saving a file, click the small arrow next to the words *File type* in the Save window, and select the appropriate Microsoft Office format from the list of available file formats listed in the pane that then appears—**Microsoft Word 97/2000/XP** for a Writer document, for example.

OpenOffice.org Features

The three main applications in OpenOffice.org (Writer, Calc, and Impress) are, respectively, quite similar to their equivalents in Microsoft Office (Word, Excel, and PowerPoint), so switching over to the OpenOffice.org applications should be relatively easy.

If you don't have any experience with Microsoft Office, you should still find it all pretty straightforward, as the basic layout is pretty intuitive. And if you are lacking in the intuition department, the built-in Help files are pretty good, too. To further help you along, the Tips system works just like tooltips in the Windows and Mac worlds. In case you aren't sure what I'm talking about, tooltips are those little yellow boxes that pop up to tell you what a button or menu item does when you place your mouse over that button or menu item.

Despite all the straightforwardness I am speaking of, there are a few interface items that will most likely be unfamiliar to you. That being the case, I will briefly discuss those items. I'll be using the word processor, Writer, as I describe these things, so if you want to run OpenOffice.org while following along, Writer might be a convenient starting point for you too.

Getting to Know the Buttons

Although you should pretty much be able to figure out what all of the buttons on the OpenOffice.org toolbars do, there are a few buttons common to all OpenOffice.org modules that most likely require, or at least deserve, a bit more explanation.

Export to PDF

This first button is situated to the left of the two printer buttons (Print and Print Preview). You can use this button to *export*, or save, your document as a PDF file, in a manner that is similar to (albeit easier than) the one in Chapter 10.

Hyperlink

Clicking the Hyperlink button, the button to the left of the Table button, brings up the Hyperlink window, from which you can assign links to specified documents—not only to web pages, but also to documents on an individual computer, and even targets within that document. While a hyperlink on a web page is something we have all come to take for granted, the idea of hyperlinking between text documents sounds like a pretty radical concept. It is, in fact, a rather old one that has been around since before you or I even heard of the Internet.

Navigator

The fourth to the last button in the top row (the one that looks like a starburst) is the Navigator button. Clicking this button brings up the Navigator window (see Figure 13-4), which is a pretty cool navigational feature that comes in handy when working with lengthy or otherwise complex documents.

Figure 13-4: The Navigator window

The Navigator allows you to easily bounce back and forth between pages in a document or even between elements therein, such as sections, links, and so on. Let's say that you have a document with lots of illustrations in it (like this chapter), and you want to jump directly from graphic to graphic. In this case, you would double-click the word Graphics in the main pane of the Navigator window and then click the jump buttons (the odd little buttons to the left of the page number selector) to begin jumping.

If you are dealing with a document containing various heading levels, like all of the chapters in this book, you can also use Navigator to switch among those levels. Say you've decided to add a new main heading at the last minute to a document you've been writing. All of the headings you had before thus need to be dropped down a notch; the former main heading becoming a subheading, and so on. By double-clicking the word Headings, the text of all the headings you have listed in the document would then appear. You could then select a heading in that list, and then click the Demote Level button (that's the one at the far right of the second row of buttons) to move it down a notch.

Gallery

To the right of the Navigator button is the Gallery button. By clicking this button, the Gallery, a library of graphical elements for use in your documents or web pages, will appear in a separate pane at the top of your document window (see Figure 13-5). The elements within the Gallery range from various types of lines to buttons to colored three-dimensional doughnuts, and you can even add items of your own.

Inserting a graphic into your document is a simple enough task even when not using the Gallery. All you need to do is go to the **Insert** menu, select **Picture ▶ From File**, and then locate the image file you want to insert. It can be handier to use the Gallery, however, when you intend to use

certain graphics frequently. Once in the Gallery, your graphics are always only a click or two away and can be conveniently viewed in the Gallery browser window.
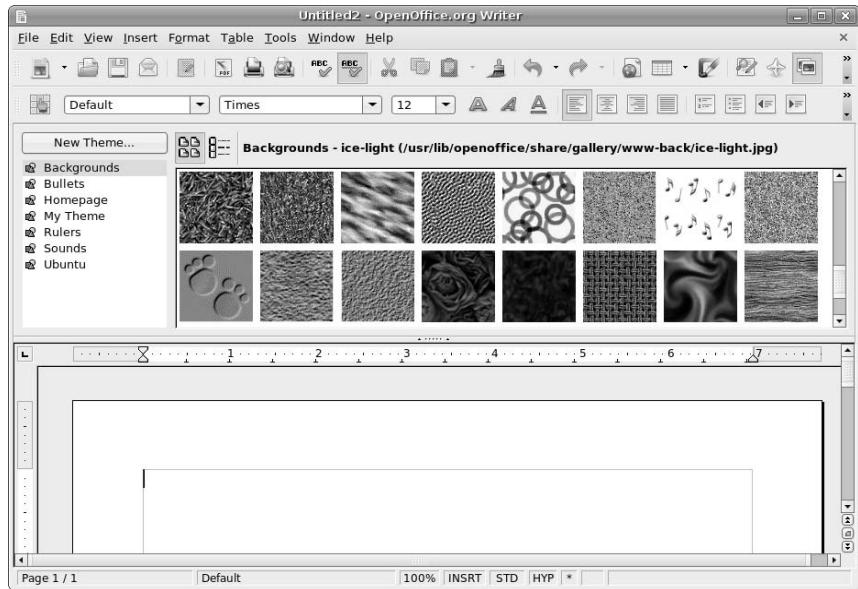
Figure 13-5: The Gallery

The collection of artwork that comes with OpenOffice.org is mostly geared toward building web pages, but there is other clipart available, and Automatix provides an easy way to get and install it. Just run Automatix, click **Office** in the left pane, check the box next to *OpenOffice Clipart* in the right pane, and then click the **Start** button. Once the rather lengthy download and installation process is complete (there's a lot of clipart in the collection), you will find it all, nicely organized and waiting for you, in the Gallery, including my favorite "stickmen" collection (Figure 13-6).
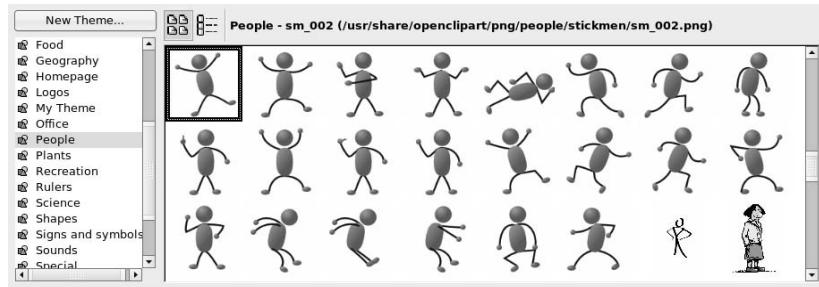
Figure 13-6: Adding clipart (including some cool stickmen) to the Gallery

Adding your own graphics to the Gallery is also relatively easy to do. First you have to create a new category (called a *theme*) for each group of images you wish to add. To create a Gallery theme of your own, just click the **New Theme** button in the Gallery window. This will open the New Theme Properties

window, where you should first click the **General** tab and give your theme a name. Once you've done that, click the **Files** tab, and then the **Find Files** button, which will bring up a Select Path window. From there you can navigate to the folder in which you are storing your clip art, photos, or other graphics. Once you have found the folder, click the **Select** button, after which a list of all the files in that folder will appear in New Theme Properties window.

You can easily add images to your new Gallery theme by clicking on the name of each image you wish to add (you might want to make sure that the box next to the word *Preview* is checked to make things a bit easier), and then clicking the **Add** button. Once you have done that, a copy of the image will immediately appear in the Gallery browser, where it will remain for future use. To use one of the images in the Gallery, just right-click the image you wish to insert into your document, and then select **Copy** or **Link** in the popup menu.

Data Source

The second to the last button in the top row is the Data Source button. Clicking this button opens a Data Source browser within a new pane in the top half of your document window. From that pane you can then access and edit records stored within any databases you have registered in OpenOffice.org. From a database containing personal contact information (such as Evolution's Addressbook), for example, you could simply do a search for an individual entry, and then drag the name, address, phone number, and/or email address for that entry into your document, rather than searching for the data elsewhere and then typing it in all over again.

Styles and Formatting

The final stop on our tour through OpenOffice's unfamiliar buttons is the Styles and Formatting button, which is located at the far left side of the second row of buttons. This button acts as a toggle for the Styles and Formatting window (Figure 13-7), from which you can select and then apply styles to any of the various elements within your document. Oh, and if you just can't take your hands off the keyboard for a moment to fiddle with the mouse, you can also open the window by pressing the F11 key.
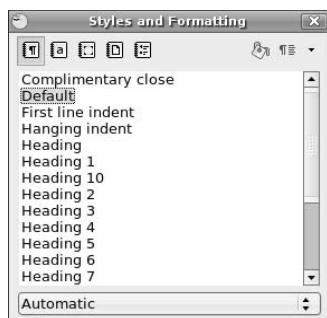
Figure 13-7: The Styles and Formatting window

To give you an example of how convenient using styles can be, imagine that you are typing a bibliography page for some document you've prepared. You typed each entry as you might any paragraph, as in:

```
    Smythe, W. (2004). Reconsidering the need for speech
between non-human interlocutors beyond the age of seven. The
Journal of the Society of Elves, Faeries, and Garden Gnomes,
20 (2), 125-147.
```

Like most paragraphs you type, the entry is formatted as a first-line indent, which is fine and dandy except for the fact that you want a hanging indent, which is the norm for bibliography entries. Rather than messing around with tabs or margins to get things the way you want, all you have to do is click your mouse anywhere within the paragraph, and then double-click the **Hanging Indent** entry in the Styles and Formatting window. After that, as if by magic (though you know better), the transformation is made:

```
Smythe, W. (2004). Reconsidering the need for speech
    between non-human interlocutors beyond the age of
    seven. The Journal of the Society of Elves, Faeries,
    and Garden Gnomes, 20 (2), 125-147.
```

You could follow the same procedure for each of your other entries, or, with Hanging Indent selected, click the **paint can** button in the Styles and Formatting window, after which your mouse cursor, when placed over the document, will appear as a paint can. Place that paint-can cursor in any other paragraph in your bibliography, and that entry, too, will be formatted in the new style. The process is essentially the same when applying different styles to any other document elements.

Word Processing Done Lightly with AbiWord

If OpenOffice.org's Writer is a bit more powerful than what you need for your everyday word processing chores, and you would prefer something that pops up as soon as you click the launcher, then you might want to consider giving another word processor, AbiWord, a try (Figure 13-8).

AbiWord has a very straightforward and easy-to-use interface, which you should be able to figure out without much, if any, help. It also has a couple of rather interesting features, such as its auto-resize function, which magnifies the onscreen document size (fonts, images, and everything) or shrinks it as you increase or decrease the size of the program window. And in case you're wondering, AbiWord can save and read Microsoft Word DOC files and save documents as PDF files.

You can easily install AbiWord via Synaptic by performing a search for *abiword*, and then . . . well, you know the drill by now. Once you have AbiWord installed, you can run it by going to the **Applications** menu and selecting **Office ▸ AbiWord Word Processor**.
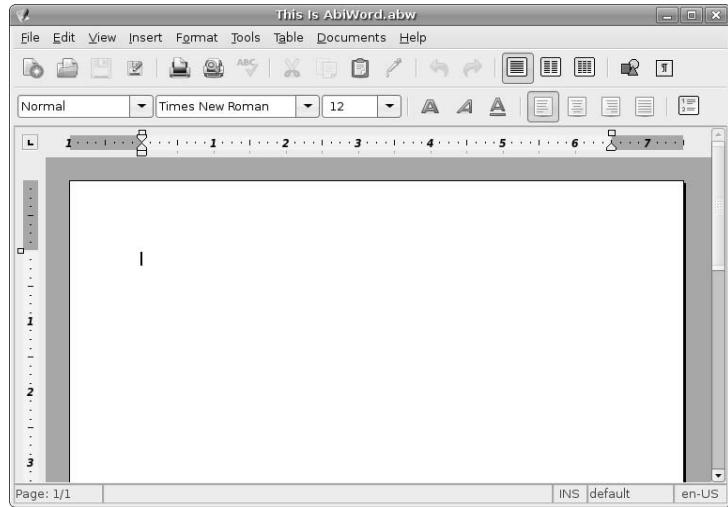
Figure 13-8: The other Linux word processor—AbiWord

Some Other Cool Productivity Apps

In addition to the more traditional office applications, there are a number of other applications either included with or available for your system that can be grouped together under the "productivity" label. I will introduce a few of those to you here.

Sticky Notes

Mac users will be well familiar with the digital version of the now ubiquitous little yellow Post-It–like notes called Sticky Notes (Figure 13-9) that come as part of the GNOME desktop. GNOME's Sticky Notes is a panel applet that you can add to your own panel by right-clicking on any open panel space and then selecting **Add to Panel** in the popup menu that then appears. When the Add to Panel window appears, go to the **Accessories** section, click **Sticky Notes**, and then click the **Add** button. You will then be ready for note-taking action.
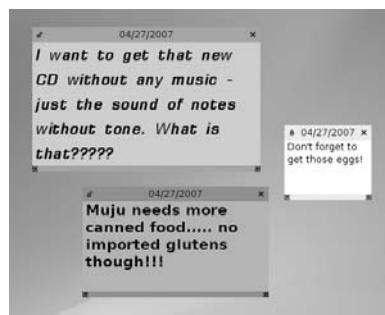
Figure 13-9: GNOME's Sticky Notes

I should mention that there is one slightly annoying "feature" of Sticky Notes that makes it a bit different from what you might be used to. When you click on the desktop, all of your notes will disappear. This is as it is supposed to be, though I personally can't see the point (and wouldn't like it much even if I could). Fortunately, however, you can bring all of your notes back to view by simply clicking the Sticky Notes panel applet. In fact, if you decide to remove the panel applet and then bring it back at some later point in time, you'll be happy to know that all of the notes you created before will be back too.

Tomboy

If Sticky Notes just doesn't cut it for you and your more dramatic note-taking needs, then perhaps you will find yourself better served by another application that comes with your system by the name of *Tomboy* (Figure 13-10). Like Sticky Notes, Tomboy also works as a panel applet, but it is a bit more full featured, albeit without making any claims to stickiness. Instead, the various notes you create can be viewed by selecting them from the menu that appears when you click the Tomboy panel applet itself. All in all, a very handy approach.
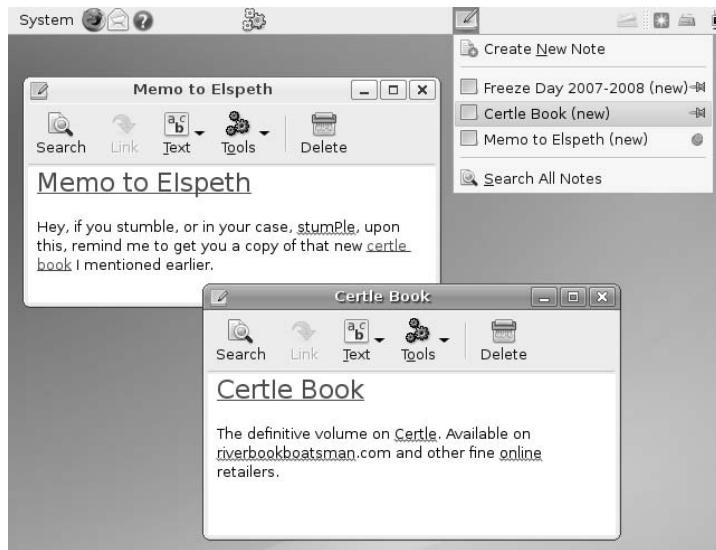
Figure 13-10: Notes taken seriously—Tomboy

What really gives Tomboy its bragging rights, however, is its search and hyperlink functions. These allow you to search for entries within your entire Tomboy note library, and create hyperlinks that connect text in one note to another linked note. In fact, Tomboy will automatically create a hyperlink whenever you type a word that matches one of your existing note headings.

To make matters even more exciting (or at least more useful), Tomboy, by means of its plugins feature, allows you to export notes to HTML or print them out, either as hard copy or as PDF docs.

If you'd like to give Tomboy a try, start it up by going to **Applications ▸ Accessories ▸ Tomboy Notes**, after which Tomboy will appear in the top GNOME Panel.

GnuCash

If you are familiar with the personal financial management software Quicken, then you might be interested in GnuCash (shown in Figure 13-11), which is the Linux world's best-known offering in the personal finance arena. It reads Quicken and Intuit QIF files, which makes things even nicer should you be making the transition from another operating system. Unfortunately, unless you live in Germany, you cannot use GnuCash for online banking; but as the GnuCash folks themselves say, don't blame them, blame your bank.

To install GnuCash, just do a Synaptic search for *gnucash*, and then mark both **gnucash** and **gnucash-docs** for installation. Once these are installed, you can run the application by going to **Applications ▸ Office ▸ GnuCash Finance Management**.

Figure 13-11: GnuCash

Scribus

To round things out, we come to the open source desktop publishing application, Scribus (shown in Figure 13-12), for those times when OpenOffice.org Writer and Draw just don't cut it. Scribus is designed to produce commercial-grade output, with support for professional publishing features, such as CMYK colors, PostScript handling, and creation of color separations, to name but a few.
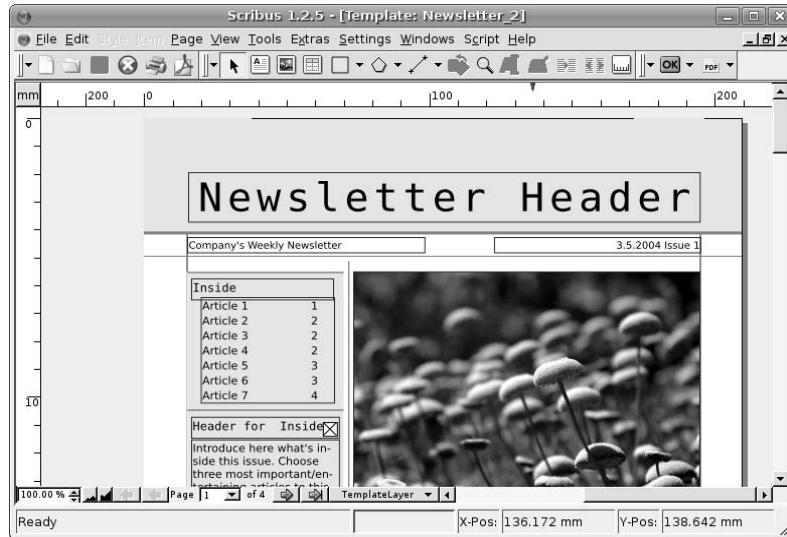
Figure 13-12: Scribus

You can download and install Scribus by doing a Synaptic search for *scribus*. Once the installation is complete, you can run the application from **Applications ▸ Office ▸ Scribus**.

14

BRUSH-WIELDING PENGUINS

Linux Does Art

Now that you know that you can get down to business in Linux, it is time to don that beret of yours and address the artistic side of things. Yes, Linux does art, and as you will soon find out, there are a good number of programs on your system that allow you to create and manipulate graphic files. These days, however, there is perhaps nothing as important to most users' graphical repertoires as their digital cameras, so that is where we'll begin.

Project 14A: Digital Cameras

While scanner support for Linux is still a bit spotty, support for digital cameras is significantly better. Linux supports over 800 cameras through the gPhoto2 digital camera software package, which comes bundled with your system. What gPhoto2 essentially consists of is a collection of drivers that works in the background to tell your computer how to communicate with your camera. To see if your camera is supported, go to the gPhoto2 website (www.gphoto.org), scroll down, and click **800 cameras** (the number may

actually be higher when you check). On that page you will find a complete list of all the cameras supported by gPhoto2. If your camera isn't on the list, it most likely means (as the page points out) that your camera is so old that there is little demand for support for it or that it is so new that there hasn't been enough time to develop support for it. Of course, gPhoto2 is constantly being updated, so if your camera isn't on the list now, it could be in the near future. Updating gPhoto2 with Synaptic now and again should keep you as up to date as possible.

If your camera is not supported by gPhoto2, all is not lost. First of all, there are some cameras that utilize what is called the USB Mass Storage protocol. These cameras function, without the help of gPhoto2, as USB storage devices, just like a thumb drive or USB external hard disk. You can access the photos on such cameras just as you would data on any other USB storage device—just connect it to your computer, and a Nautilus window appears displaying its contents. You can then transfer files via conventional drag-and-drop procedures.

Even if your camera doesn't seem to communicate with your computer by either of these means, you can still get your images to your hard disk by removing the memory card from your camera, inserting it into a USB flash memory card reader, and plugging that reader into one of your computer's USB ports. Once you've done that, your system will mount the card reader as if it were an external drive (which is pretty much what it is), thus allowing you to use simple drag-and-drop procedures to get the images to your hard disk. Of course, you can use this method even if your camera is supported by gPhoto2 or compliant with the USB Mass Storage protocol—sometimes it is the easiest way to deal with things anyway.

14A-1: Connecting Your Camera and Starting the Import Process

If your camera is supported by gPhoto2 and you prefer to access your photos directly from your camera rather than fiddle with flash storage cards (or if you're just not sure whether your camera is supported or not), connect the USB cord supplied with your camera to the camera itself and then to one of the USB ports on your computer.

Once you've done that, turn the camera on, and set it to communicate with your computer. Switching the camera to Play mode, rather than Camera mode, usually seems to do the trick, but you should check your owner's manual just to be sure. If your camera is supported by gPhoto2, and the chances of that are quite good, a small window will appear, as shown in Figure 14-1. To view the images on your camera, click the **Import Photos** button.

Figure 14-1: Your system seeks your approval before importing photos from your digital camera.

14A-2: Setting Up and Cleaning Up Before Importing the Images

After you have clicked the **Import Photos** button in that first window, another window (Figure 14-2) will appear displaying the photos currently in your camera. In this window, you can do a little housecleaning and setup before actually saving photos to your hard disk.
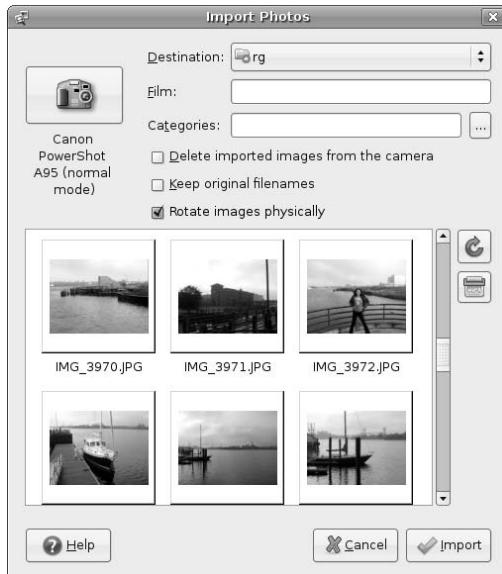
Figure 14-2: Importing digital photos from a camera

If you would like to delete all of the photos on your camera after importing them, check the box next to the words *Delete imported images from the camera.* If you would like to delete only a few images before getting around to the business of importing, click on the image or images you want to delete (holding down the CTRL key if you are selecting more than one), and then click the Delete icon.

As you can see, there is also an option that allows you to keep the original filenames assigned by your camera. You no doubt understand what this means, so I'll tell you what not selecting this option means instead. When you import images from your digital camera, they will be saved in a folder and named in date-and-time format, such as 2006-02-19-19.08.13, while the images within that folder will be saved in numeric format, such as 0001.jpg, 0002.jpg, and so on. If you prefer to keep the filenames assigned by your camera (and as they appear in the Import Photos window), check the box next to *Keep original filenames.*

Finally, there is the question of where you save these date-and-time named folders. The default location is in your home folder, which is convenient enough; however, if you take a lot of photos, you will find that after some time your home folder contains so many folders full of imported images that it is difficult to find anything. Therefore, you may want to have your photos downloaded to a more specific location, such as a Photos folder.

To do this, click the drop-down menu button next to the word *Destination*, and select **Other**. This will bring up a Choose Destination Folder window. In that window, navigate to your Photos folder (or create one by clicking the **Create Folder** button in that window), and then select that folder by clicking it once. Once your selection is made, click the **Open** button, after which you will be returned to the Import Photos window with your new destination folder now listed in the Destination box at the top of the window.

14A-3: Saving the Images to Your Hard Disk

Now that everything is set up, you can save all of the photos in the Import Photos window by clicking the **Import** button. If you are in a pick-and-choose sort of mood, you can save only the images you want by holding down the CTRL key and then selecting just those images by clicking on each of them before clicking the **Import** button.

Once you have selected your photos (if you choose) and clicked the Import button, the images will be saved to your hard disk with the progress of the process indicated at the bottom of the Import Photos window. When the process is complete, the images you have just saved to your hard disk will appear in a new window, as you can see in Figure 14-3.
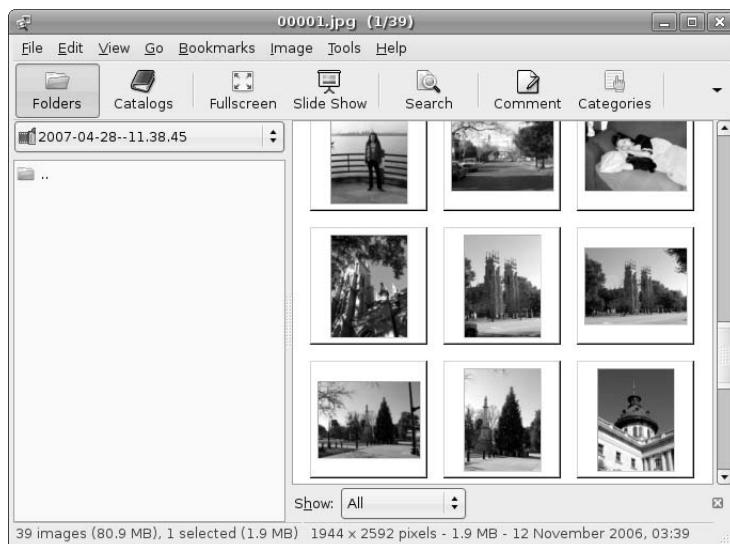
Figure 14-3: Imported images displayed in the gThumb image viewer

Viewing Your Images with gThumb

The images that appeared in Figure 14-3 are being displayed by an application called gThumb. As you can see, gThumb is an image viewer, but it can actually do a lot more. Although not as elegant in appearance as Apple's iPhoto, gThumb can actually perform many of the same functions, and it can even perform some functions that iPhoto can't. If you don't happen to

be importing any images from your digital camera right now and would like to give gThumb a look, you can do so by going to the **Applications** menu and selecting **Graphics ▸ gThumb Image Viewer**.

Getting to Know Some of gThumb's Features

Now you know how to view images in gThumb, but so what, right? Well, as I said, gThumb does a lot more—so much, in fact, that it is a really useful tool for the digital photographer. With gThumb you can view your images as a slideshow by clicking the **Slide Show** button; view them at full-screen size (**View ▸ Fullscreen**, or just click the **Fullscreen** button); rotate and mirror them (**Tools ▸ Rotate Images**); and convert their file formats, from JPEG to PNG, for example (**Tools ▸ Convert Format**).

If you want to adjust the actual look of things, you have still more options. To access most of these editing options, first select the image you want to tackle, and then go to the **Image** menu and select **Enhance**. Once you've done that, gThumb will have a go at making your image look its Sunday best (if it supports your image format).

If your aesthetic sense is a bit different than gThumb's, you can go the manual route, vetoing the changes gThumb made by clicking once on the image and then clicking the **Do Not Save** button. You can then go to the **Image** menu, and change the brightness, contrast, color balance, and hue saturation of your image. You can also resize and crop your images from the same menu.

If you want to print out one of your images, doing so via gThumb is extremely easy. Just click the image you want to print, and then select **Print** in the **File** menu.

Saving Photos to CD in gThumb

One of gThumb's very iPhoto-ish features is its ability to burn images to CD. To use this feature, select the images you want to save to CD (in Folder view— just click the **Folders** button to get there), and then select **Write to CD** in the **File** menu. After that, you will be asked in a separate window whether you want to save the whole shebang to disk or just those items you selected. Make your choice, and then click **OK**. A Nautilus CD/DVD Creator window will then appear showing the items you've selected. After that, it is all the same process as described in Chapter 6, so it is pretty much smooth sailing.

Project 14B: Creating Web Albums with gThumb

The gThumb features I've mentioned so far are pretty much the same as those found in Apple's iPhoto. Now, however, we come to one feature that isn't: gThumb's ability to create web albums of your photo collection. These albums can be used on your computer itself, distributed on CDs, or uploaded to your website for the whole world to see . . . and they look pretty nice.

14B-1: Selecting Images

To create a web album of your own, select the images you would like to add to your album (while holding down the CTRL key), go to the **Tools** menu, and select **Create Web Album**. A new window, Web Album (shown in Figure 14-4), will appear.
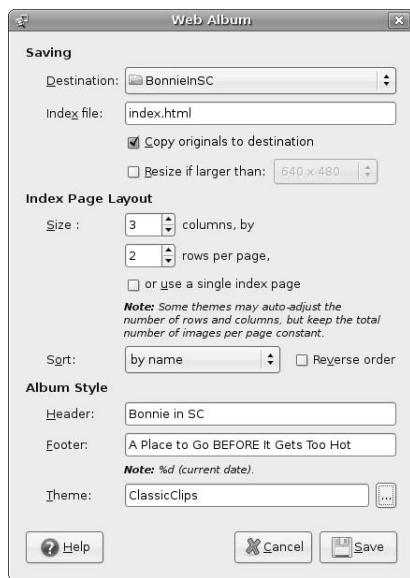
Figure 14-4: Creating a web album in gPhoto

14B-2: Selecting a Destination Folder

In the Web Album window, select a folder other than your home folder in which to place your web album (you'll end up with a mess in your home folder if you don't). After that, select the new folder you've created as the destination folder by clicking the drop-down menu button next to the word *Destination* and then selecting **Other**. In the Choose destination folder window that appears, navigate to the new target folder, select it by clicking its name once, and then click **Open**.

14B-3: Copying Images and Page Layout Options

Once you've set the destination folder for your web album, you can decide whether or not to copy the images in your album to the destination folder. In the same window, check the box next to *Copy originals to destination* if you plan to upload the web page to your website or distribute it as an archive or on CD. If you just plan to view the album on your own computer, leave the box blank.

Once you've done that, decide how many images you want to appear on each page of your album by adjusting the numbers in the Index Layout section of the window. For most screen resolutions, 2 or 3 rows by 4 columns works very nicely.

14B-4: Adding Headers and Footers

While still in the same window, give your album a title by typing that title in the Header text box. You can also type a string of text to appear at the bottom of the page by typing text in the Footer text box.

14B-5: Choosing a Theme

Finally, choose a theme for your album by clicking the **…** button to the right of the Theme text box. As you can see in Figure 14-5, there are six themes available. Click on each of the themes once to see a preview of that theme in the right half of the window. Once you have made your selection, click the **Select** button. For my part, I will be using the Classic Clips theme.

Figure 14-5: Selecting a theme for your web album

14B-6: Creating and Viewing a Web Album

Once your selections are made, you can create your web album by clicking the **Save** button. If you chose to create a web folder, a small window will appear, telling you that there is no such folder and asking you if you want to create it. Of course you do, so click **Create** in that window. After a few seconds, your web album will ready for viewing.

   To see your new creation, open a Nautilus window, navigate to the target folder you created for your web album, double-click it, and then look for the file index.html within that folder. Once you've found it, double-click it. Firefox will then open to your new album. You can see mine in Figure 14-6.

Figure 14-6: Viewing thumbnail images in a web album

As you can see, the web album presents your photos as a set of thumbnails in a photo or slide motif (depending on the theme you choose), with buttons there to connect you to whatever other pages your album happens to have. Click any one of the thumbnails in your web album, and that image will appear in enlarged form within the same window, as you can see in Figure 14-7.

Figure 14-7: Viewing enlarged images in a web album

Project 14C: Emblems Again! (Creating Your Own)

One of the first things my mother asked me after she got Linux up and running was how to create her own emblems. You already learned how to deal with emblems in Chapter 7, but now that I have introduced many of the graphics tools at your disposal, it might be a good time to address my mother's question as well.

14C-1: Using an Existing Icon as an Emblem

You can use any image as the basis of an emblem, but the easiest way to get started creating emblems is to use one of the icons on your system as the base. The only problem with that idea is that most of your system icons are 48 by 48 pixels in size, while emblems are 36 by 36 pixels. If you don't want your new emblem to look weirdly oversized and out of place, it is best to resize it.

As I mentioned in the previous section, the gThumb application is very handy for resizing images. For this project, we will use the gnome-tigert icon, so in the left pane of the gThumb window, choose **File System** in the drop-down menu button above the left pane, and then navigate to /**usr**/**share**/ **pixmaps**. Once there, choose **View ▸ View As List**, then look for the **gnome-tigert.png** file (it looks like an orange cartoon character), double-click it, and then go to the **Image** menu and select **Resize**. This will bring up the Scale Image window (see Figure 14-8), where you can change the width of the icon to 36 pixels (the height will change automatically, as long as the box next to the words *Keep aspect ratio* is checked). Once you are done, click **Scale**.

Now you need to save the newly sized icon to your home folder, while keeping the original as it was. To do this, go to the **File** menu and select **Save As**, which will bring up the Save Image window. Save the file as gnome-tigert_emblem.png in your home folder. You can do this by clicking the drop-down menu button next to the words *Save in folder* and navigating your way home. Once that's done, click **Save**.

14C-2: Adding the Newly Sized Image to the Emblem Collection

Now that you have an image of the appropriate size, it is time to add it to your system's collection of emblems. To do this, open a Nautilus window, go to the **Edit** menu, and then select **Backgrounds and Emblems**. Once the Backgrounds and Emblems window opens, click the **Emblems** button on the left side of the window, and then click the **Add a New Emblem** button at the bottom of that window. This will bring up a small Create a New Emblem window (see Figure 14-9).

In that window, start out by giving your emblem a name. As the image is that of a tiger, I called mine *Kitten Stuff*, but you can call it something else if you'd like. Just type the name in the Keyword box.
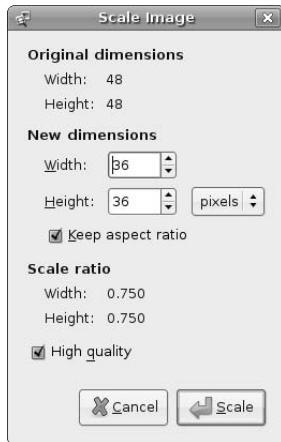
Figure 14-8: Scaling an image
in gThumb

Figure 14-9: Adding a new
emblem to your collection

After you have done that, it is time to work on adding your new emblem to your system's emblem collection. To do this, just click the big button next to the word *Image*. This will open a window from which you can select the image. In the text box at the top of that window, type **~/gnome-tigert_emblem.png**, and click **OK**. Your new Kitten Stuff emblem should now appear in the Create a New Emblem window. If so, click **OK**.

That window should now disappear, and your new Kitten Stuff emblem (or whatever you called it) will appear in the Emblems window (see Figure 14-10). Now you can use your new emblem as you would any other.
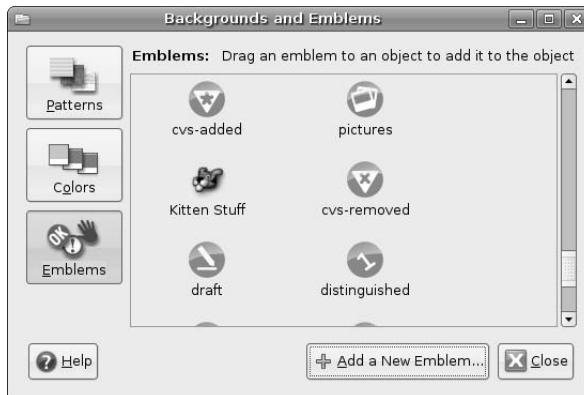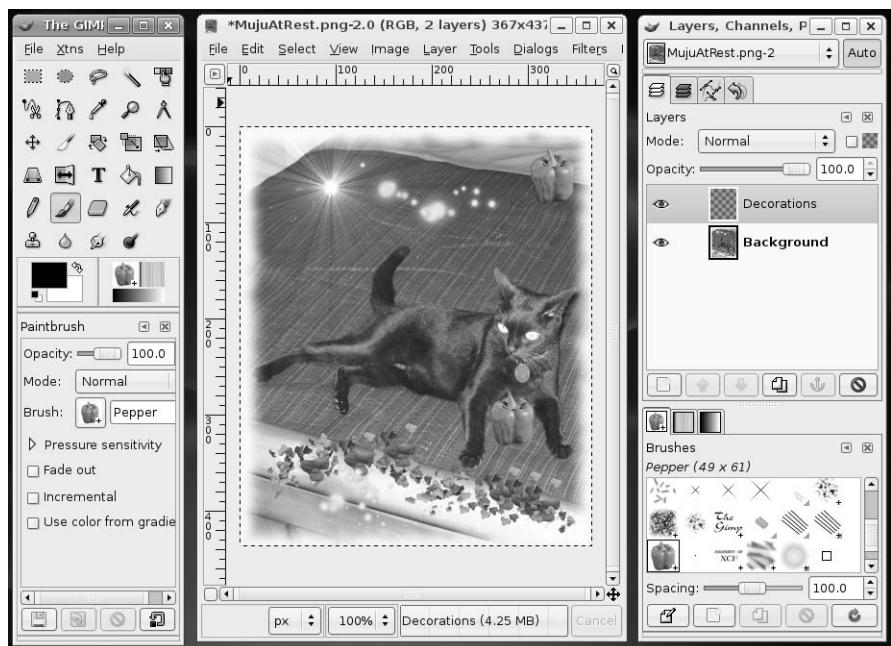
Figure 14-10: A new emblem (Kitten Stuff) in the Emblems window

Getting Arty with the GIMP

The Windows and Mac worlds may have Photoshop, but the Linux world has the GIMP. While arguably not as powerful as Photoshop, the GIMP is a capable contender, which may explain why it has been ported over to both

Mac and Windows. The GIMP allows you to create bitmap graphics and, quite importantly, retouch or completely doctor image files. With the GIMP you can get rid of red-eye in your digital photos, airbrush out unwanted shadows (or even facial blemishes), give your image a canvas texture, change a photo into an oil painting, and even add a bell pepper here and there—and with drop shadows no less (see Figure 14-11). To run the GIMP, go to the **Applications** menu, and select **Graphics ▸ GIMP Image Editor**.

Figure 14-11: Manipulating a digital image in the GIMP

Using the GIMP to Resize Images and Convert File Formats

Like gThumb, the GIMP is also a very handy tool for resizing images. This can be done by simply right-clicking an image opened in the GIMP and then selecting **Image ▸ Scale Image** in the popup menu. This will bring up the Scale Image window, where you can set the new size of the image.

The GIMP is also an excellent tool for converting images from one file format to another. You can, for example, open a bitmap (.bmp) file and save it as a PNG (.png) file, or save a JPEG (.jpg) file as a GIF (.gif) file, and so on. While this can also be done with gThumb, the GIMP supports an extremely wide variety of file formats, and it even lets you save an image file as a compressed tarball, which makes it a true file-conversion king.

To perform a file conversion, just right-click an image opened within the GIMP and then select **File ▸ Save As** in the popup menu. You can make the same selection from the **File** menu if you prefer. Either way, the Save Image window will then appear. In that window you can specify the new file format

by replacing the original file extension in the Name box at the top of that window with the extension for the format you want to convert the image to. If you're not sure what formats are available to you, click the small arrow to the left of the words *Select File Type (By Extension)*, and then choose from the options in the pane that appears below. To save a work in progress, use the GIMP-native XCF format, so that you can continue working on the image later.

Dialogs

The GIMP interface, as you might have noticed, is based on a series of dialogs, two of which appear when you first run the application. You can, however, open still others, and, more importantly, you can dock those together to form larger single dialogs. While this is quite easy to do, figuring out how to do it is something that leaves many GIMP newbies at a loss.

The various dialogs available to you in the GIMP can be found by going to the **File** menu in the main GIMP window (that's the one that usually appears at the left of your screen, as in Figure 14-11), and selecting **Dialogs**. From the submenu that appears, make your choice from the list of dialogs shown.

After you've opened a couple of dialogs, you can dock them together by clicking and then holding the name of the dialog, located just below the title bar, and dragging that name to the bottom of the dialog you wish to dock it to. The name of the dialog will appear as a floating icon/name tab as you do your dragging (Figure 14-12).
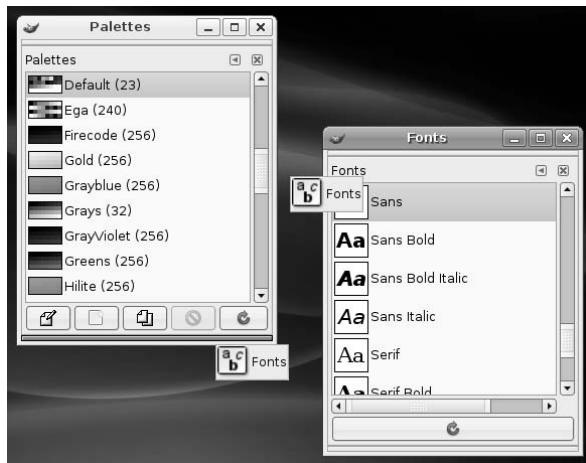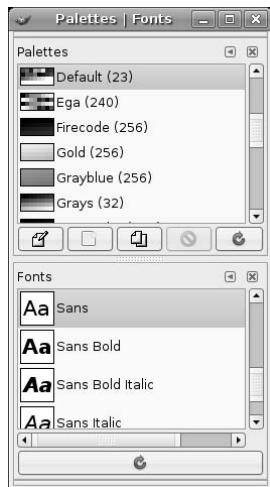
Figure 14-12: Docking dialogs in the GIMP

When the thin, gray band near the bottom of the target dialog turns brown (compare the bottom of the two dialogs in Figure 14-12 to see the difference), release the mouse button, and the dialogs will be joined as one, as you can see in Figure 14-13.

Figure 14-13: Docked dialogs
in the GIMP

Learning More

It is lots of fun to learn to use the GIMP by just playing around with it for a while. Most of the fun stuff, to get you started, is located in the Filters and Script-Fu menus of any image window. Of course, you should make a backup copy of any file you are planning to experiment with before altering it.

If you prefer working through manuals and tutorials to just finding things out by goofing around, you can download and install the GIMP User Manual via Synaptic by searching for *gimp* and then marking **gimp-help-en** for installation. Once it is installed, you can access the manual from within the GIMP from the Help menu. If you prefer, you can just view the manual online at http://docs.gimp.org/en. You can also find a series of skill-level-based tutorials at www.gimp.org/tutorials.

XPaint

If the GIMP comes across as a bit overwhelming for you, or if it seems to be overkill for your simpler tasks, you might want to try out an application known as XPaint, which you can download and install via Synaptic (search for *xpaint*). XPaint is similar to Windows Paint and MacPaint.

Even if you are a satisfied GIMP user, there is still good reason to download and install XPaint, because in addition to its simple paint tools, it can take screenshots of regions (Figure 14-14) rather than just of a particular window or of the full screen. If you have ever used the Mac OS X application Grab, you may well be familiar with this feature.

By the way, once installation is complete, there will be no launcher for XPaint in the Applications menu, so you will have to run from the Run Application panel applet or from a Terminal window by typing `xpaint`.
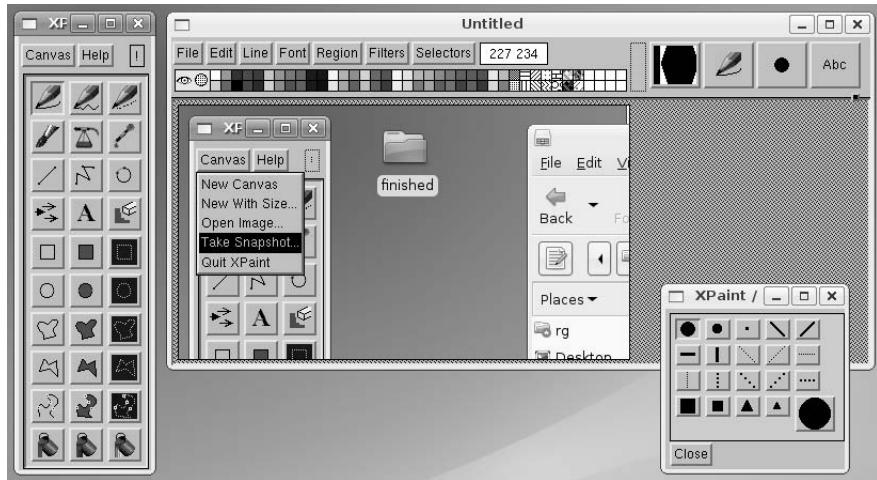
Figure 14-14: Taking screenshots of screen regions with XPaint

Sodipodi

The GIMP, like other so-called *paint programs*, creates bitmap images in various file formats. These are images in which the location and color of every single pixel is recorded. The image is essentially a collection of dots, or *bits*. The file you create is a rather hefty map of these bits (hence its name), and this map tells your system where everything in your image is supposed to go when it is displayed or printed.

Drawing programs, on the other hand, create vector images, or drawings. The vector image file is a collection of mathematical formulae representing the various shapes in your image. This may sound rather unimportant to you, but such drawings have advantages in certain cases. One of these advantages is that vector image files take up less space on your hard disk than bitmaps. Another, and perhaps the most important, advantage is that shapes in vector images retain their smooth edges when the images are enlarged. A smooth circle created as a bitmap, for example, would begin to show jagged edges ("the jaggies") when enlarged to any extent, while the same circle in a vector image would remain smooth and round no matter how much you increased its size.

If you're interested in giving a drawing program a go, then my pick for creating true graphics is Sodipodi—the name is Estonian for *mishmash* (see Figure 14-15).

To download and install Sodipodi, do a Synaptic search for *sodipodi*, and then mark the file for installation. Once it is installed, you can run it from the **Applications** menu by selecting **Graphics ▶ Sodipodi**.

If you would like to learn how to use Sodipodi, go to the Sodipodi home page at www.sodipodi.com. Be sure to click the **Galleries** link at the top of that page to see examples of what you can do with the program.
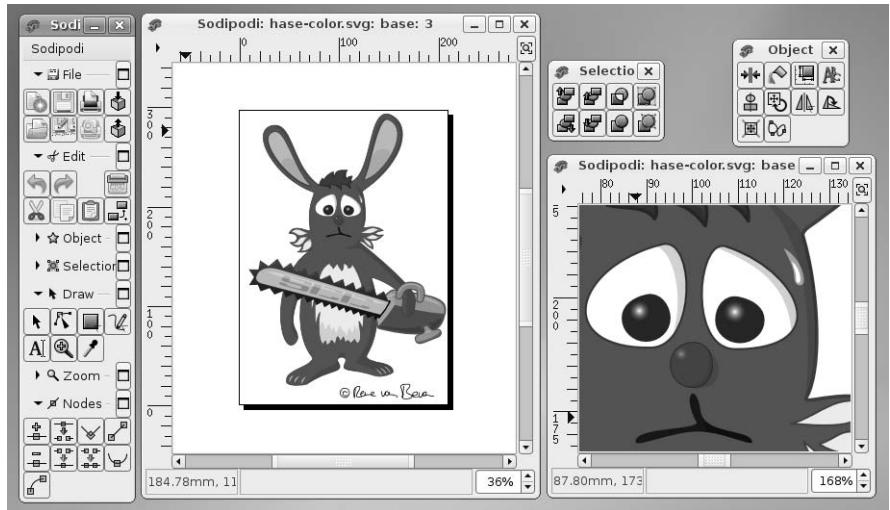
Figure 14-15: Sodipodi

Project 14D: Installing Picasa via Automatix

It wouldn't be right to dedicate a whole chapter to Linux's graphical capabilities without mentioning one of the newest entries in the Linux application arena—Picasa. Those of you coming from the Windows world are no doubt familiar with this very popular image viewing, organizing, and editing application from Google, and you will no doubt be pleased to discover that it is now available for Linux. Picasa is not an open source application, which may cause some Linux diehards to turn away, but it is free, feature rich, nice to look at, and decidedly cool (Figure 14-16). It also gives you easy access to various online photo blogs, photo finishers, and product providers, such as PhotoStamps, Shutterfly, Kodak, and even Walgreens.

14D-1: Downloading and Installing the Picasa Package

The easiest way to get Picasa is via Automatix, which you installed and learned to use in Chapter 9. Open Automatix (**Applications ▸ System Tools ▸ Automatix**), and once it is up and running, click **Office** in the left pane. In the right pane of the Automatix window, check the box next to *Google Picasa*, and then click the **Start** button at the top of the window. The download and installation process will then begin.

A small window will appear, telling you that you need administrative rights in order to proceed. Click **Grant**, and you will be prompted for your password. Type that, and click **OK** to begin the installation. Automatix will show the progress of the download and installation and let you know when it is done.

Figure 14-16: Picasa

14D-2: Running and Setting Up Picasa

Once Picasa is installed, you can run it by going to the **Applications** menu and selecting **Graphics ▸ Picasa**. The first time you do this, you will be greeted by a Picasa License Agreement window. Agree to what you're asked by clicking **I Agree**.

You will then be asked in another window, and again only the first time out, whether you want it to scan your entire computer for images (the default selection) or just your desktop. Do not, I repeat, do not accept the default (*Completely scan my computer and all network directories for pictures*).

The reason for this warning is that even though you may have yet to add a single image to your computer, your system is already chock-full of them. If you were to accept the default, Picasa would be filled with all sorts of stuff that you really wouldn't want there. It is supposed to be an organizer for *your* photos, after all. With that in mind, select **Only scan the Desktop**, and then click the **Continue** button. Picasa will then scan your desktop for images and add any images it finds to its library.

Working with Picasa should be very straightforward, but if you want to find out more, check out http://picasa.google.com/linux. On that page, you will find a basic overview, links to more Linux-specific information (including FAQ and forum pages), and a Picasa tour.

A Few Other Graphics Apps to Consider

In addition to the graphics applications I have covered in this chapter, there are still more available. One comes preinstalled with your system, while the others can be grabbed via Synaptic (or for some easy browsing, via the Add Applications tool). While you can experiment with what's available, I will point out a few others worth noting. If nothing else, these applications will give you an idea of the breadth of stuff out there waiting for you.

F-Spot Photo Manager

In addition to gThumb Image Viewer, your system comes with yet another application for your image viewing and managing needs, F-Spot Photo Manager (Figure 14-17).
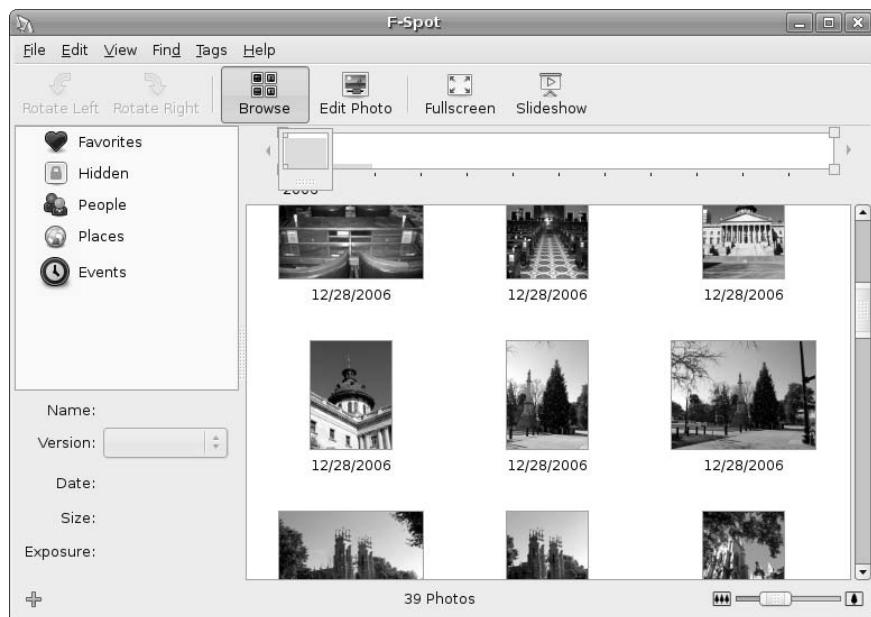
Figure 14-17: F-Spot Photo Manager

When it comes down to it, the features of F-Spot are quite similar to gThumb, so it really boils down to your personal preferences. Have a go at both (and Picasa too, for that matter), and see which you like. You can run F-Spot from **Applications ▸ Graphics ▸ F-Spot Photo Manager**.

Blender

Perhaps one of the most impressive open source applications available today is Blender. Blender (Figure 14-18) is a professional-level 3D modeling, animation, and rendering program. It is rather complex, but that is the source of its power and popularity (it comes in versions for just about every operating system out there). If you would like to find out a bit more about Blender before taking the time (and disk space) to install it, go to www.blender3d.org. Find it via Synaptic by searching for *blender*; once the program is installed, you can run it from **Applications ▸ Graphics ▸ Blender 3D modeller**.

Figure 14-18: Blender

QCad

Another open source application that has found its way into almost all operating systems is QCad (Figure 14-19). QCad is a 2D computer-aided design (CAD) program with which you can create technical drawings such as room interiors, machine parts, or even musical instruments (I've seen a plan for a Nyckelharpa done on QCad!). To find out more, go to the project home page at www.ribbonsoft.com/qcad.html. Search Synaptic for *qcad*, and install the application. Once it is installed, you can run it from **Applications ▸ Graphics ▸ QCad**.
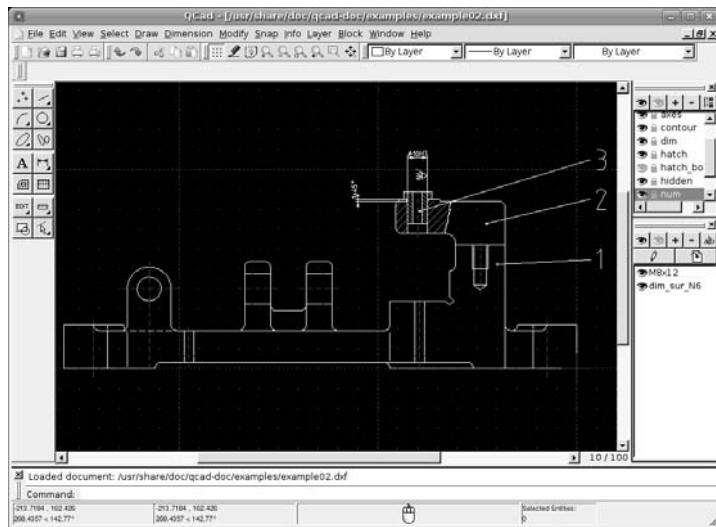
Figure 14-19: QCad

Tux Paint

To wrap things up, let's turn to an application for the kids (or the kids within us) and have a look at Tux Paint. With its big colorful buttons and fun and funky tools, Tux Paint, shown in Figure 14-20, is an app that your children can handle and enjoy. The best of Tux Paint's features (at least in my opinion) are its stamps, of which there is a good variety—everything from apples to seahorses, and euro coins to boot! Search Synaptic for *tuxpaint*; launch it from **Applications ▸ Graphics ▸ Tux Paint**.
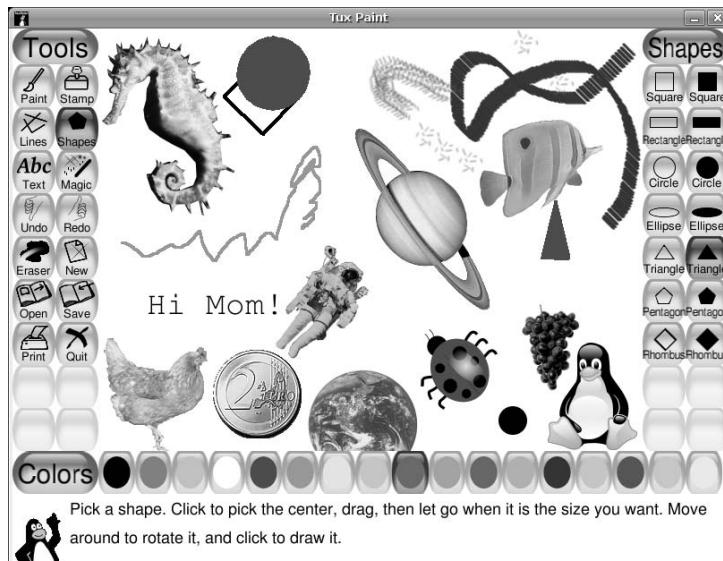
Figure 14-20: Tux Paint

15

TUX ROCKS
Music à la Linux

It's now time to move on to the audio side of things. Yes, Linux does indeed rock, and in this chapter you will find out about those musical talents that your system possesses. The size of this chapter alone should be some indication of how much there is to offer. You will learn how to rip CDs, create MP3 and Ogg Vorbis files (files which you can recognize by their .mp3 and .ogg filename extensions), add album cover art, change file tags, play music files, and burn files onto audio CDs that you can play in any CD player. You'll also learn how to play a variety of audio streams. If you're interested in learning how to work with your iPod in Linux, how to deal with podcasts, and how to convert audio files from one format to another . . . well, you'll have to wait until Chapter 16.

Audio File Formats

Before we go any further, it is probably best to discuss the various formats in which audio data can be stored on your computer. For the longest time, the de facto standards have been WAV (created by Microsoft/IBM and using the

.wav extension), AU (from Sun/Unix and using the .au extension), and AIFF (from Apple and using the .aiff extension), all of which are uncompressed formats. Files saved in these formats are, therefore, exceedingly large, with an average WAV file of CD-quality music weighing in at about 10MB per minute. To put that in perspective, back in 1988 my first Macintosh had a 40MB hard disk—more space than I thought I would ever need, but not enough space to store a WAV file of Nirvana's "Come as You Are."

As computers underwent their evolution into the multimedia machines they are today, it became clear that something was going to have to be done about those disk-space-devouring audio files. Audio compression formats were thus developed. These compression formats worked, to oversimplify things a bit, by cutting out the portions of a sound signal that the human ear cannot hear—sort of a dog-whistle approach. The most widely known and embraced of these audio compression formats is MP3. Audio files encoded in MP3 format can end up being as little as one-twelfth the size of the original WAV file without any noticeable loss in quality.

Another audio compression format that was developed was Ogg Vorbis. Ogg Vorbis was a product of the open source community, so, unlike MP3, which has always been used under the shadow of yet-to-be-exercised patent rights, it was free of patent and licensing worries from the get-go. Because of that, and the fact that it was the equal of MP3 in terms of quality and performance (if not, as many claim, better), Ogg Vorbis became the darling of the Linux community.

As you work with the audio rippers mentioned in this chapter, you are sure to notice yet another encoding option—FLAC. *Free Lossless Audio Codec (FLAC)* is an encoding format that, unlike MP3 or Ogg Vorbis formats, does not remove any audio information from the audio file during the encoding process. The downside of this is that FLAC only provides space savings of 30 to 50 percent, which is much less than the 80 percent neighborhood achieved by MP3 or Ogg Vorbis formats. The upside, of course, is that the FLAC files should be equivalent to CDs in terms of quality.

Given that retention of audio quality, FLAC becomes an ideal choice if you are not satisfied with the audio quality provided by Ogg Vorbis or MP3 files. It is also a good choice if you might want to create both Ogg Vorbis and MP3 files . . . or if you just don't know which one you want to work with yet. In such cases, you can just rip the file and encode it in FLAC format. You can convert the FLAC file later, when you know what you want or need. You'll learn how to easily convert audio files from one format to another in Chapter 16.

Project 15A: Installing MP3 Support for Audio Apps

All that talk about Ogg Vorbis and FLAC aside, there are still many people who like or need to deal with MP3 files. You may, for example, already have numerous MP3 files ripped from your music collection, or you may enjoy listening to one of the many Internet radio streams that are broadcast in

MP3 format. Unfortunately, MP3 playback and encoding support is not included in Ubuntu due to licensing concerns. Even if you plan on using Ogg Vorbis format in your future ripping and encoding endeavors, installing MP3 support is a good idea so as to cover all your audio bases. It's also a very easy process.

The easiest way to install playback and encoding support for just about any format, and for most of the applications I will be discussing in this chapter and throughout the book, is via Automatix (**Applications ▸ System Tools ▸ Automatix**). Once Automatix is open, click **Codecs and Plugins** in the left pane of the window, and then check the boxes next to *AUD-DVD Codecs* and *Multimedia Codecs* in the right pane. Once you've done that, click the **Start** button at the top of the window, and the download and installation processes will begin. Be forewarned that this is a somewhat hefty download, so have a little patience at hand if your Internet connection is a slow one.

Audio Rippers and Encoders

The application you use to rip audio files from CD and encode them into space-saving MP3 or Ogg Vorbis formats is commonly referred to as a *ripper*. For some time now, the most commonly used non–command-line ripper was Grip, which still has its dedicated following. Other simpler-to-use rippers, however, have surfaced more recently, such as RipperX, Goobox, and the one that comes bundled with Ubuntu: Sound Juicer.

Sound Juicer

Sound Juicer (Figure 15-1) is a relatively new program that is pretty straight-forward to use and quite capable in terms of what it does. It isn't perfect, though, and it still can be a bit quirky. In addition, unlike many of its ripping cousins, Sound Juicer does not automatically create a playlist for the songs you rip and encode, and it lacks a simple means by which to adjust the encoding bitrate.

Figure 15-1: Ubuntu's default ripping and encoding application—Sound Juicer

Despite these limitations, there are still many people who prefer Sound Juicer to the competition, so you might as well give it a try to see how you like it. To get started, just place the CD you want to rip in your drive; Sound Juicer will start up automatically and display the title of your CD, the artist's name, and titles of all the tracks in the application window.

NOTE   *It is important to mention at this point that if you are not connected to the Internet, these bits of album information will not appear because album and track data are not embedded in the CD itself. What happens instead is that the audio ripper or player on your machine sends the digital ID of the CD you're playing to an online database, such as freeDB.org or CD DataBase (CDDB), which is now officially known as Gracenote. In turn, the online database sends the album information for that CD back to the player or ripper.*

You can go about things another way by going to **Applications ▸ Sound & Video ▸ Sound Juicer CD Extractor** and then placing your CD in the drive. In this case, however, you might have to go to the Sound Juicer **Disc** menu and select **Re-read Disc** before your album and track information will appear.

Setting the Default Folder for Ripped Files in Sound Juicer

By default, Sound Juicer is set up to rip your CDs and encode audio tracks in Ogg Vorbis format, and the default location in which Sound Juicer saves these files is your home folder. Because Sound Juicer, like all other rippers, will create an artist folder for each CD you rip, you will end up with a lot of folders in your home folder if you rip albums from a large number of artists. It is best to create a *Music* folder within your home folder, as you did for your graphika account, and then make that folder your default location for ripped music.

To set the default from within Sound Juicer, go to the **Edit** menu and select **Preferences**. This will open the Preferences window where you can change the output path by clicking the menu button next to the word *Folder* (the button itself should say *Home* at this point) and then selecting **Other** in the menu that appears. After that, browse to your **Music** folder, click it once in the list to select it, and then click the **Open** button.

Ripping and Encoding Sound Files in Sound Juicer

To start ripping the audio tracks from the CD you have in your drive, you first need to select the format in which you wish to encode the tracks. To do this, go to **Edit ▸ Preferences**, and then in the **Output Format** menu select the encoding format of your liking. The most of common of these is **MP3 (MP3 audio)** or, in the Linux world, **CD Quality, Lossy (Ogg multimedia)**. Once you've made your choice, click **Close**.

After that, all you need to do is click the **Extract** button at the bottom of the Sound Juicer window to rip and encode all of the tracks on the CD. If there are certain tracks you do not care to rip and encode, just uncheck the checkboxes next to the names of those songs before you click Extract. If you

only want to rip and encode a few of the songs in the list, it might be better to first go to the Edit menu, select Deselect All, and then check the checkboxes next to the songs you do want to rip before clicking Extract.

While the songs are being ripped and encoded, Sound Juicer will show you its progress in the lower-left corner of the window, and when it's all done, it will tell you so in a small window. Once you get that message, click **Close**. You can eject the CD by going to the **File** menu and selecting **Eject**. If you want to rip and encode another CD, pop it into the drive; just as with the first CD, the album, artist, and song titles will all appear in the program window, and you can rip away yet again.

Once you're done with your ripping chores, you *could* check out the results of your efforts using one of the players discussed in the following sections, of course. But the quickest and perhaps the most interesting way to play back your newly ripped files is to open a Nautilus window, and then navigate to the new tracks within your Music folder. Once you're there, place your cursor over any one of tracks and a little eighth note in one of those comic bubbles will appear (Figure 15-2), and the track, without so much as a single mouse click, will mysteriously start playing. Kind of cool, don't you think?

Figure 15-2: Playing an audio file from within Nautilus

Audio Players

Now that you know how to rip your audio CDs and encode them to space-saving MP3 and Ogg Vorbis files, you need to use another application to play them. Fortunately, this is an area where Linux shines, as there is quite a selection of audio players available, two of which I will discuss here.

Rhythmbox

The default audio player in Ubuntu is called Rhythmbox (see Figure 15-3). Rhythmbox, using Sound Juicer as its ripper/encoder, seems to function pretty much like a simplified version of Apple's iTunes application, though, for better or worse, iTunes it is not. It is, however, a relatively easy-to-use audio player which, despite some quirkiness in its early stages, has developed quite a following in the Linux world.
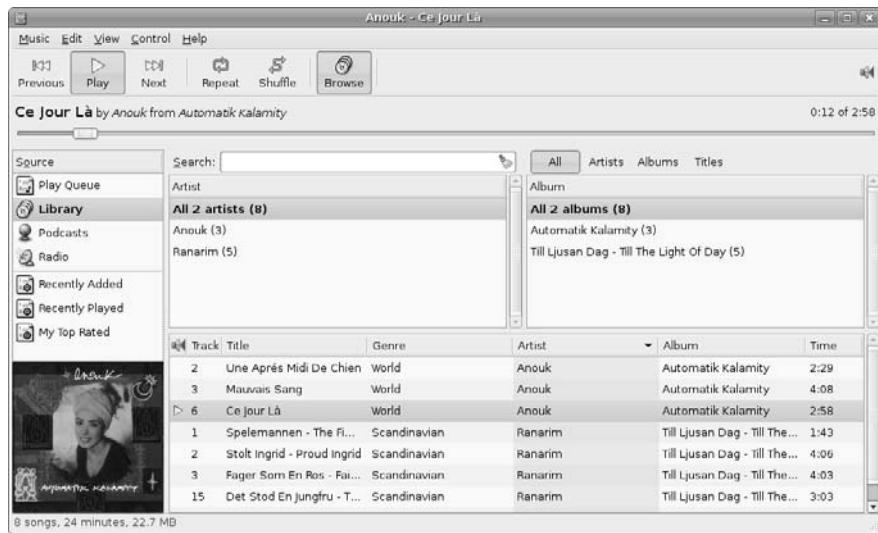
Figure 15-3: The Rhythmbox audio player

Running and Setting Up Rhythmbox

To run Rhythmbox, just go to the **Applications** menu and select **Sound & Video ▸ Rhythmbox Music Player**. The first time you run Rhythmbox, you are greeted with a three-step setup wizard. To get through the first introduction screen, click **Forward**. In the second screen, you should tell Rhythmbox where it is you keep (or plan to keep) your music files. Use the Music folder you created in "Setting the Default Folder for Ripped Files in Sound Juicer" on page 248. Just click the **Browse** button to locate the folder graphically, and then click **Forward** in the wizard window. In the third and final screen, click **Apply**, and Rhythmbox will scan your Music folder and add any songs it finds there to its Library.

Adding Songs and Albums to the Rhythmbox Library

If you are familiar with Apple's iTunes, then you should understand the Library in Rhythmbox too, as it is essentially the same concept, though it does not physically move your files as iTunes does.

To add new albums to your Rhythmbox Library, go to the **Music** menu, select **Import Folder**, and then navigate to the folder for the new album you want to add. If you want to add a number of albums by the same artist, just navigate to and select the folder for that artist instead. If you just want to add a single audio file, you can also do that—just select **Music ▸ Import File**, and then navigate to the song in question.

If you are not the navigating type, you can also add files and folders by other means. The simplest way is to drag the folder or song you want to add to the Library into the right pane of the Rhythmbox window. You can also add a song to the Library directly by right-clicking the file and selecting **Open with ▸ Open with Rhythmbox Music Player** in the popup menu that appears.

Rhythmbox, in conjunction with Sound Juicer, also allows you to add songs to the Library directly from a CD, albeit in a somewhat indirect way. To do this, select **Music ▸ Import Audio CD**, and Sound Juicer will appear and display the contents of the CD you currently have in your drive. After that, you can rip and encode the songs as you normally would in Sound Juicer. Once you're done, the songs will automatically appear in the Rhythmbox Library.

Browsing the Rhythmbox Library

The Rhythmbox Library is a collection of all of the music you add to it, which can prove to be a bit unwieldy as your collection grows. Fortunately Rhythmbox has a nice browser function, like the one in iTunes, that allows you to see lists of the artists and albums in the Library. If you click a specific artist in the Artist pane, a list of albums by that artist will appear in the right pane. You can then double-click one of the albums in that right pane to play it. If you want to play all of the albums you have by that artist, just double-click the artist's name in the left pane. All in all, a very handy feature.

If you would like to add an additional layer of categorization to the browser, you can do so by adding a Genre pane. To do this, just go to the **Edit** menu, select **Preferences**, select the **General** tab of the window that appears, and then select **Genres, artists, and albums**. After clicking the **Close** button in the Preferences window, you will have a three-pane browser in Rhythmbox (Figure 15-4).
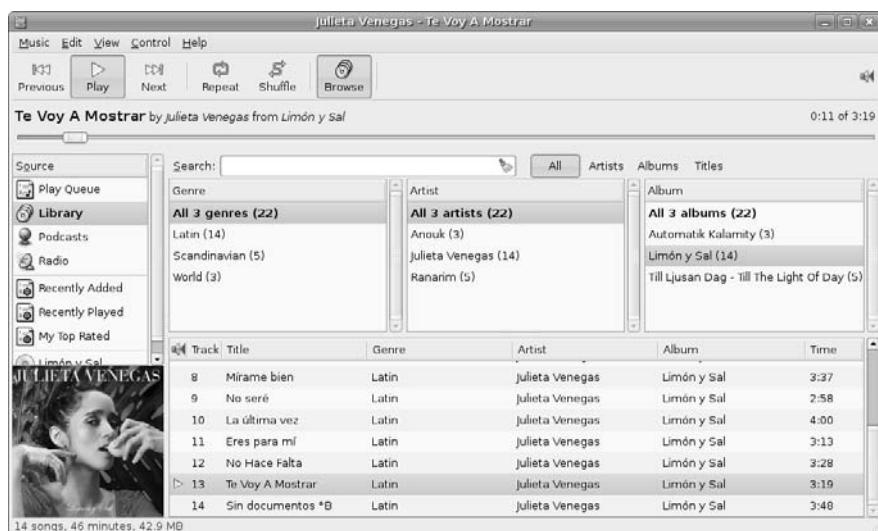
Figure 15-4: The Rhythmbox browser

Creating Playlists in Rhythmbox

Of course, you can tailor things even further so as to match every situation and your every mood. There are days, after all, when you're feeling a bit too ethereal for Kelly Clarkson (and thank goodness for that). To prepare

for such moments, you can create *playlists*, which are lists of songs to be played in a predetermined order. Just think of each playlist as an all-request radio station . . . where all the requests are your own.

To create a playlist in Rhythmbox, go to the **Music** menu and select **Playlist ▸ New Playlist**. When the Playlist icon appears in the left pane of the Rhythmbox window, type a name for the list, and then add the songs you want by simply dragging them from the Library pane to the Playlist icon. The songs themselves will remain in the Library, so you aren't really moving anything—just creating aliases.

You can also create *automatic playlists*, which are lists that automatically scan the Library for songs that match your creation criteria. You could choose to create a list for the all the songs in your Library by a particular artist or of a specific genre. To do this, just select **Music ▸ Playlist ▸ New Automatic Playlist**. A window will appear in which you can specify what the list is to contain. You can even specify how many songs you want in the list.

Once you've created your lists, you can play one by clicking the list once and then clicking the Play button near the top of the window, or by just double-clicking the list.

Streaming Media with Rhythmbox

In addition to allowing you to play the music you have on your hard disk, Rhythmbox also allows you to play Internet radio streams in either MP3 or Ogg Vorbis format. To add a stream, click the **Radio** icon in the left pane of the Rhythmbox window, and you will come face to face with . . . well not all that much the first time around. Still, you can try out what there is to try. Make sure you're online and double-click one of the stations listed, and your computer should start to play music downloaded from that Internet radio station.

Once you've had a taste of what Rhythmbox does with radio, you will no doubt be hungry for more. To add a bit more choice to your listening diet, you can add new streams by clicking the **New Internet Radio Station** button. When the New Internet Radio Station window appears, type the URL for the stream, and then click **Add**. The stream will appear in URL form in the right pane of the Rhythmbox window. If you would prefer the stream to appear in the list as a name rather than a URL, just right-click the entry for the stream, select **Properties** in the pop-up menu, and then make the appropriate changes in the Properties window.

Ah, but where do I find these radio streams, you ask? For the largest collection of MP3 streams, just go to www.shoutcast.com, find a stream that seems interesting to you, right-click the **Tune In!** button for that stream, and then select **Copy Link Location** in the popup menu. After that, paste the URL in the URL box of Rhythmbox's New Internet Radio Station window, and click **Add**. You can then double-click the stream in the right pane of the Rhythmbox window to play it.

Of course, many broadcasters also offer live Internet feeds that do not appear on SHOUTcast lists. An example of this would be Radio France, which offers a number of streams. The stream I recommend you try out is Fip, which is an exceedingly cool collection of music of all genres; you will

be constantly surprised by what they play. If you are worried about not under-standing the language, don't be; there is very little talk, and roughly 80 percent of the music played is in English. I don't speak any French other than *je suis fatigué*, but I have no problems at all.

If you want to give streaming media in general (and Radio France in particular) a try, type `http://viphttp.yacast.net/V4/radiofrance/fip_hd.m3u` in the URL box of Rhythmbox's New Internet Radio Station window, and then click **Add**. Assuming you are already connected to the Internet, double-click the new addition to your list, and Fip should start playing after a few seconds of pre-buffering.

Other Cool Features in Rhythmbox

Rhythmbox includes a couple of cool features worth noting. One is pretty obvious once you start playing your ripped tracks: album cover art. As you can see in both Figures 15-3 and 15-4, whenever you play a track in Rhythm-box, the album cover art for that track is automatically downloaded and displayed in the lower-left corner of the window (assuming that Rhythmbox can find it online).

Another cool and somewhat similar feature has to do with lyrics. To utilize this feature you must first activate it by going to the **Edit** menu and selecting **Plugins**. In the **Configure Plugins** window that appears (Figure 15-5), check the box next to *Song Lyrics* and then click the **Close** button. After that, go to the **View** menu and select **Song Lyrics**. Rhythmbox will then search the Net for the lyrics to the track you are currently playing and display them in a separate window (Figure 15-6).

Figure 15-5: Activating Rhythmbox's lyrics function

Finally, if you are an iTunes user and attached to a bit of visual stimulation while playing your tunes, Rhythmbox can now satisfy you with a visualizer of its own. To activate it, go to the **Control** menu and select **Visualization** (you can turn it off in the exact same way). The right half of the Rhythmbox window will then be filled with the wild, swirling shapes and colors of psyche-delia that you've come to know and love.

Figure 15-6: Rhythmbox displays the lyrics to
the track currently playing.

Exaile—An amaroK-like Audio Player for GNOME

Now that Rhythmbox has, for the most part, matured into a usable and multi-featured application, you may not want to bother with any other players. Still, there are other interesting players out there.

One of the newest is Exaile (Figure 15-7). *Exaile* is essentially a clone of the Linux application amaroK, which I covered in the previous edition of this book. While amaroK was (and is) a great player, it is a KDE application, so when it's running in Ubuntu's GNOME environment, it can be a bit quirky at times. As a KDE app, it also fails to adopt the look-and-feel customizations that you may have made to your Ubuntu environment.

Figure 15-7: Exaile!

Exaile, on the other hand, is a GNOME app through and through. It looks like the rest of your system, reflects any customizations you may have made to the rest of your system, and runs as smoothly and quickly as any GNOME native application.

Installing and Running Exaile

The easiest way to install a good working copy of Exaile is via Automatix. Just go to the **Applications** menu, and select **System Tools** ▸ **Automatix** to get started. Once the main Automatix window appears, click **Media Players and Editors** in the left pane. Then in the right pane, check the box next to *Exaile* and click the **Start** button and you'll be on your way.

Once everything has been downloaded and installed, you can quit Automatix and run Exaile from the **Applications** menu by selecting **Sound & Video** ▸ **Exaile!**

Streaming Media with Exaile

Like Rhythmbox, Exaile can also act as an Internet radio tuner. In fact, in many ways it does a better, or at least more convenient, job of it because it automatically gathers a list of the feeds available from SHOUTcast, making it a bit more iTunes-like. To access these station lists, just click the **Radio** tab at the left side of the Exaile window, and then navigate to the genre and sub-genre of your liking.

If you would like to play a stream not listed in the SHOUTcast offerings, you can do so by going to the **File** menu, selecting **Open URL**, typing or pasting the URL for the stream in the **Enter the address** window that then appears, and then clicking the **OK** button in that window. A new Stream tab will appear in the right half of the Exaile window, showing your stream. To save the stream for future play, right-click the stream on the **Stream** tab, and select **Add to Saved Stations** ▸ **New Station**. In the small **Enter the name of the station** window that appears, enter the name of the station, and then click **OK** (Figure 15-8). After that, you can find your newly saved station in the Saved Station folder in the Radio tab on the left pane.

Other Cool Features in Exaile

Exaile has a lot of other cool features worth noting. One that you can't help but notice is that it provides an onscreen display of the track or stream that is playing, showing not only the title and artist of the track, but the album cover art as well (Figure 15-9). This is especially handy when using Exaile in hidden mode, which you can accomplish by using the Exaile panel applet as a toggle.
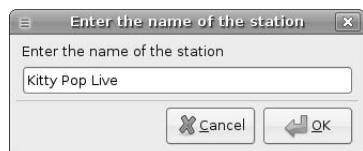
Figure 15-8: Naming a stream you've added to the Exaile Radio list

Figure 15-9: Exaile's onscreen display

If you don't like where this onscreen display appears (or how it looks), you can change it by going to the **Tools** menu and selecting **Preferences**. When the Preferences window appears, click **OSD** in the left pane, and a list of options available to you will appear in the right half of the window. To change the location of the display, just drag it to the spot you like while the Preferences window is open. When you're done making your changes, click **Apply** to make things take effect immediately, and then click **OK** to close the Preferences window and seal the deal.

As I've already mentioned, Exaile also downloads and displays the album covers for the tracks you are playing, much in the same way Rhythmbox does. Like Rhythmbox, it can also download and display the lyrics for the track you are currently playing. To see the lyrics, just go to the **View** menu and select **Information ▸ Lyrics** (you can, if you prefer, just right-click the track you are playing and select **Information** in the pop-up menu). A set of Information tabs will appear, with the lyrics being found, logically enough, in the Lyrics tab. By clicking the Artist tab, you can also display the results of a Wikipedia search for information on the artist currently playing (Figure 15-10), including the URL for the Wikipedia page being displayed. Of course, this is not fool-proof. When I tried it for the British band Sing-Sing, for example, I got a Wikipedia page on Sing Sing prison in New York. Interesting enough, I suppose. . . .

Figure 15-10: Artist information from Wikipedia in Exaile

Creating Audio CDs

All this talk about encoding and listening to MP3 and Ogg Vorbis files on your computer is fine and dandy, but there are no doubt times when you would like to have your songs on a plain audio CD that you can play while you slog your way to work on the New Jersey Turnpike or the Ventura Freeway. Luckily, this is easy enough to do, and there a couple of ways to go about it.

Burning Audio CDs with Rhythmbox

For Rhythmbox users, the easiest way of going about things is to do it all from within Rhythmbox. To do this, first create a playlist with the songs you want to burn to CD, and then click the **Create Audio CD** button. A confirmation window will appear, asking you if you want to create the CD you've just set out to create. You no doubt do, so click the **Create** button, and copies of your tracks will be created in WAV format. When that process is complete, you will be prompted for a blank CD, so pop one into your drive, click **OK**, and the burn process will begin. When it's done, the progress window will vanish, and your new audio CD should pop out of your drive. Simple.

Burning Audio CDs with Serpentine

Another way to create play-anywhere audio CDs from your ripped MP3 and Ogg Vorbis files is with *Serpentine Audio CD Creator,* an application that comes bundled with your system. To go this route, go to the **Applications** menu and select **Sound & Video ▸ Serpentine Audio CD Creator**. When Serpentine appears, add songs to the main pane by clicking the **Add** button and navigating to the songs you want to add. You can add any mix of songs, in any mix of audio formats. You can also add songs you have saved to a playlist by going to the **File** menu and selecting **Open Playlist** or **Open Rhythmbox Playlist**, depending on the type of playlist in question. If you are a drag-and-drop kind of person, you can also drag the icons for your audio files directly from one or more Nautilus windows, or from playlists within Rhythmbox, to the main pane of the Serpentine window.

As you add songs to the Serpentine window, the CD image in the bottom of the window will show you how much more space you have available on the disk (Figure 15-11). The default disk size is 74 minutes, which would be the norm for a 650MB blank CD, so if you are using a 700MB CD, change the setting to 80 minutes by clicking the drop-down menu button next to the words *Disc Capacity* and making the appropriate selection.

When you have added all the songs you want to burn to the CD, click the **Write to Disc** button. Serpentine will then begin preparing the tracks to burn to disk. When it is ready, it will prompt you in a new window to insert a blank CD into your drive, so do that, and once the LED on your drive stops blinking, click the **OK** button. Serpentine will then begin burning the disk, and

when it is done, it will automatically eject your disk and let you know in yet another window that the job is done. You can then take the disk, plop it in your car stereo or wherever else you want to play it, and enjoy the results.

Figure 15-11: Preparing to burn an audio CD with Serpentine

Project 15B: Listening to RealMedia Streams with RealPlayer

Now that you know how to play Ogg Vorbis and MP3 files and Internet broadcast streams, it is time to help your system go a bit more mainstream by installing RealPlayer 10 (Figure 15-12). RealMedia streams are widely available on the Internet and are provided by many mainstream broadcasters, both local and international. You can also use RealPlayer to play RealVideo streams, when they're available.

Figure 15-12: RealPlayer 10

Linux users familiar with earlier versions of RealPlayer will be happy to know that RealPlayer 10 is much improved over previous Linux offerings. One of the big advantages is that RealPlayer 10 now has a working Firefox-compatible plugin, which means that you can now listen to RealMedia streams that were previously unavailable because they could only be accessed through browser-embedded players. If you don't know what I'm talking about, take a look at the example from the Radio Sweden site in Figure 15-13.

The Radio Sweden site (www.sr.se), like many others, uses an embedded player for the broadcast streams it provides; without installing the proper plugin, the player cannot be used (Figure 15-13, left). But once the RealPlayer plugin is installed, its controls automatically appear (Figure 15-13, right), allowing you to use the embedded player just as you would the stand-alone version (shown in Figure 15-12).
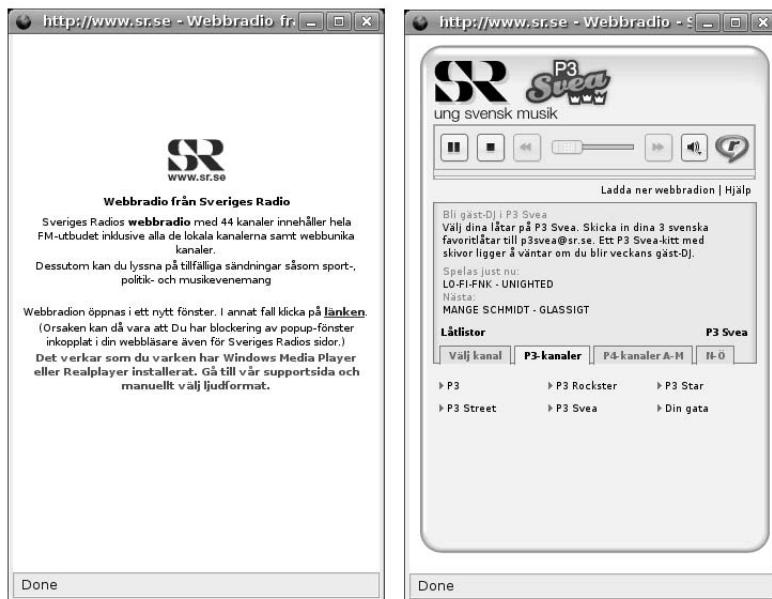
Figure 15-13: Embedded media players in Firefox before and after installing the RealPlayer plugin

While on the topic of changes, I should also mention that RealPlayer 10 is now theme compatible. This means that if you change your system theme from Human to Crux, for example, those effects take place in RealPlayer 10 too. You are no longer stuck with the look that the Real folks provide. Yes, things just keep getting better.

Now that you know what is store for you with RealPlayer 10, it's time to get down to installing it. Before we begin, however, I should mention that there are actually three ways to install RealPlayer. In this project, I am going to focus on the easiest. It is also the standard and, without question, the most legal way of doing it.

Installing RealPlayer

Installing RealPlayer is easily accomplished via Automatix. Just click **Media Players and Editors** in the left pane of the Automatix window, and then check the box next to *RealPlayer* in the right pane. After that, just click the **Start** button, and you'll be rolling.

Setting Up RealPlayer and Testing Your Installation

Now that Real Player 10 is installed on your machine, it is time to go through the final setup steps and then test it all out. You can start by going to the **Applications** menu and selecting **Sound & Video ▸ RealPlayer 10**. A simple setup wizard will appear, which you will have no trouble with on your own, as all you really have to do is click the button in the bottom-right corner of the window four times. Once you have completed the wizard, the RealPlayer window will appear. To test the installation, close the RealPlayer window and open Firefox.

The next step is to check whether the RealPlayer plugin is doing what it should. You can determine this by going to www.npr.org and clicking the **24-hour Program Stream** link at the top of the page. Because you installed the two audio codec packages via Automatix earlier in the chapter, an embedded Windows Media Player window will appear. To switch over to RealPlayer, click the **Set Player Preference** button at the bottom of the window, click the **RealPlayer** icon in the screen that then appears, and then click **Set Player Preference** again at the bottom of the window. The embedded player near the top of the page should now be RealPlayer (as you can tell by the RealPlayer icon at the right side of the player).

Playing MP3 and Ogg Vorbis Streams with RealPlayer

An interesting and not unwelcome result of installing RealPlayer with the Firefox plugin is that you can now play MP3 and Ogg Vorbis Internet broadcast streams with a single click of your mouse. Go to www.shoutcast.com and click the **Tune In!** button next to one of the streams to see what I mean. All very handy, you've got to admit.

Going to Town with RealPlayer

Now that you are all set up, you probably want some more streams to try out. There are plenty of sites with RealMedia streams, but let me steer you to some of my favorites to get you started (you may already know some of these):

- Världen (world music from Radio Sweden)
  www.sr.se/srvarlden
- Car Talk (car talk)
  www.cartalk.com/radio/show
- Michael Feldman's Whad'Ya Know? (comedy/game)
  www.notmuch.com
- A Prairie Home Companion (Garrison Keillor, et al.)
  www.prairiehome.org
- Radio Netherlands (pop music)
  www.rnw.nl
- Sounds Eclectic (alternative/world music)
  www.soundseclectic.com

Other Cool Audio Apps

The applications I have covered so far are only a taste of what Ubuntu has in store for you in its repositories. You can start out by browsing though the offerings listed in the Add Applications tool or Automatix. You can also perform searches in Synaptic for other applications you hear about on the Internet or that I mention in this section. Try them out, keep them if you like them, or remove them if you don't. After all, it doesn't cost you anything in Linux.

To give you a starting point, I will mention a few applications that I think might be worth investigating.

Other Audio Players

If the music players that I've covered in this chapter have whetted your appetite for more, you are in luck—there are plenty, a few of which I will briefly introduce now.

Listen

The very appropriately named *Listen* is a nice variation on the Exaile theme, though many find it more intuitive to use. Listen provides album cover art and lyrics fetching and display; onscreen display of currently playing track info; Wikipedia searches; Internet radio streams, including Last.fm similar artists ratings (Figure 15-14); and a whole lot more. Check it out by downloading and installing it via Automatix (click **Media Players and Editors** in the left pane, and then click **Listen Media Manager** in the right pane). Once you have installed Listen, start it up from **Applications ▸ Sound & Video ▸ Listen Music Player**.

Figure 15-14: Last.fm similar artists suggestions in Listen

Songbird

Another, somewhat more recent entry in the Linux music player sweeps is Songbird (Figure 15-15), which is very similar to and yet very different from everything out there. It is essentially what the Songbird folks themselves describe as "a desktop Web player, a digital jukebox and Web browser mash-up." Hard to resist giving any sort of "mash-up" a try, isn't it?

If you find yourself losing the resistance game, give Songbird a try by installing it via Automatix (**Media Players and Editors ▸ Songbird**) and starting it up from **Applications ▸ Sound & Video ▸ Songbird** once it's installed.
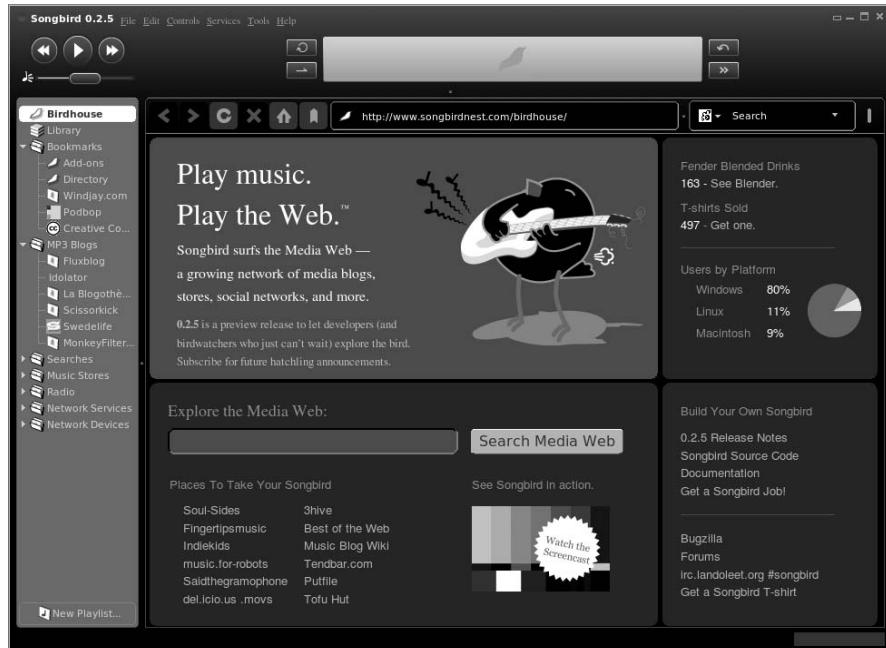
Figure 15-15: The well-connected browser, jukebox, and player mash-up—Songbird

Beep

With all the big, multi-featured music players out there, it is easy to overlook the original, or in this case, a clone of the original—Beep. Beep is essentially a more GNOME-friendly version of the classic X MultiMedia System (XMMS) player, which was the original king of Linux audio players. Beep, like XMMS, is set up to work like a component stereo (with movable components!). The fact that it is skinnable means that you can make it look like a real component stereo, as well (Figure 15-16). You can download and install Beep via either Automatix or Synaptic. To take it for a spin, go to **Applications ▸ Sound & Video ▸ Beep Media Player**.

Figure 15-16: Different faces of the versatile, skinnable audio player Beep

If you are an Internet radio junkie, there is probably no application as useful to you as Streamtuner (Figure 15-17). Streamtuner is, as its name implies, an online radio stream tuner. It works by downloading lists of available streams from a variety of sources, which you can then easily browse. When you find something you like, just select the stream, click the **Play** button, and the stream will open in the player of your choice (though it works best with RealPlayer or XMMS). If you will be using a different player for playback, go to Streamtuner's **Edit** menu, select **Preferences**, and then in the right pane of the Preferences window, change the default applications for playback (most will be set as XMMS). For example, if you want your playback to be from Beep, you would click **Listen to a .m3u file**, scroll to the right, and change xmms %q to `beep-media-player %q`.

If you're installing Streamtuner, you might as well install Streamripper, which is a sort of Internet audio stream recorder. It works in conjunction with Streamtuner, allowing you to rip the streams you are listening to (and even those you aren't listening to) and save them as MP3 files. To record a stream from within Streamtuner, just right-click the target stream, and then select **Record** in the popup menu.
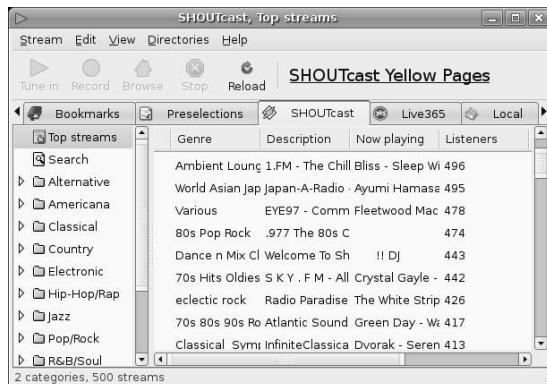
Figure 15-17: Browsing Internet radio streams with Streamtuner

EasyTAG and Audio Tag Tool

For the true audio geek, these two applications allow you to alter the tags of your MP3 and Ogg Vorbis music files. EasyTAG is the more full-featured of the two, but Audio Tag Tool (Figure 15-18) has a friendlier user interface. Give 'em both a try and see what you think. Search Synaptic for *easytag* or *tagtool*, install one or both, and then locate either under **Applications ▸ Sound & Video**.
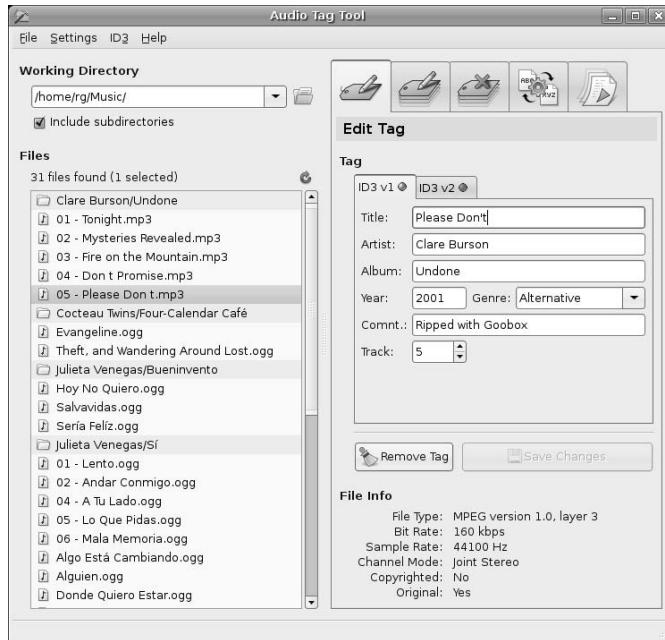
Figure 15-18: Editing MP3 and Ogg Vorbis file tags in Audio Tag Tool

LMMS

Linux MultiMedia Studio (LMMS) is a combination tracker/sequencer/synthesizer/sampler that is easy to use—at least that's what the LMMS home page suggests when it promises "easy music-production for everyone." I can't swear to that fact myself, as I don't know much about apps in this genre, but there is a lot of hoopla about it out there in the Linux world, and it sure looks cool, as you can see in Figure 15-19. It sounds really cool too (try out the demos once you install it). Search Synaptic for *lmms*, install the program, and run the command `lmms`.

Audacity

If you're interested in podcasting, one application you will find repeatedly mentioned is Audacity. *Audacity* (Figure 15-20) is a multi-platform audio recording and editing application that is frequently used not only in the

Linux world, but in the Mac and Windows worlds as well. It allows you to cut and paste bits of sound, raise pitch, increase speed, add echo and other effects, and . . . well, all sorts of other neat stuff. To get it, run Automatix, click **Media Players and Editors** in the left pane, and then check the box next to *Audacity* in the right pane. After that, click the **Start** button. Audacity will then be available from **Applications ▸ Sound & Video**.
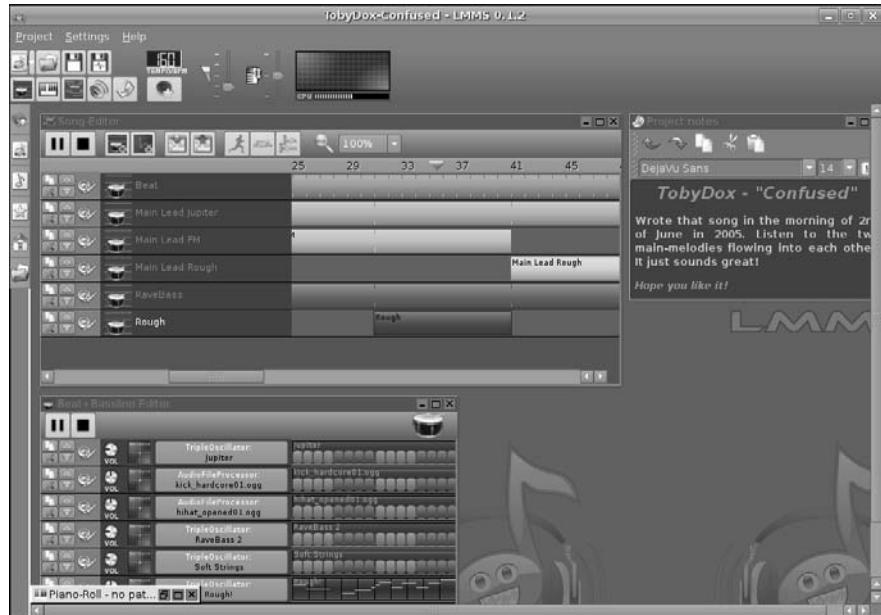
Figure 15-19: Creating music with LMMS
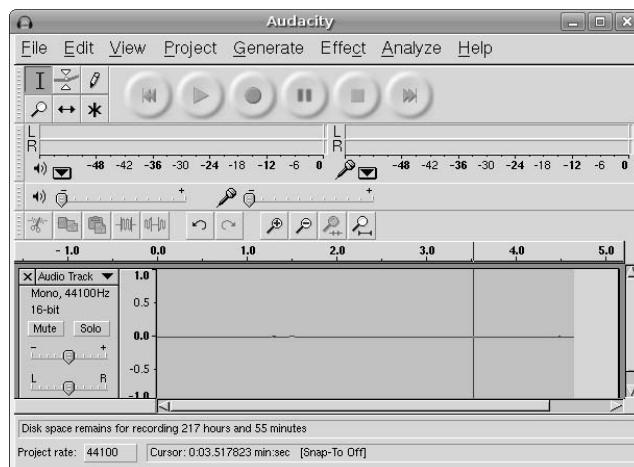
Figure 15-20: Audio recording and editing with Audacity

16

PLUGGIN' IN THE PENGUIN

Ubuntu and Your iPod

With all the talk in the previous chapter about ripping, encoding, and playing back audio files, you may be wondering whether or not you'll be able to transfer any of those files to your iPod using Linux.

Well, you will be happy to know that Ubuntu does iPods, even Nanos. You will also be happy to know that using your iPod on your Ubuntu system is quite easy. All you have to do is plug your iPod into one of your computer's USB ports, after which Ubuntu will automatically mount the iPod and place an iPod icon on your desktop (Figure 16-1). No longer do you have to mess around with mount and unmount commands or editing system tables. Just plug in your pod, and Ubuntu will do the rest.

Figure 16-1: A desktop icon for a mounted iPod

Knowing Your Limits

Although you can use your iPod in Ubuntu, you should remember that support for such devices is still rather new. Because of that, there is likely to be the occasional odd moment while working with your iPod. I have been using mine without problems, but a friend did lose all his data when he managed to freeze his system while doing something bizarre. The same thing happened to me on a Mac, so I guess that's just the life of the Pod. Anyway, if you do happen to lose all the data on your iPod or somehow corrupt its system, you can just use Apple's iPod Updater while in Windows to bring it back to normal and repopulate its song library from your hard disk.

Assuming you downloaded and installed the two codec packages via Automatix in Project 15A on page 246, you shouldn't encounter any problems if your iPod already has songs on it in either MP3 or iTunes' default AAC format that you ripped using iTunes. As for the protected AAC files you might have bought from the iTunes store—well, forget about playing those back in Linux unless you're willing to do a bit of geeking around (https://help .ubuntu.com/community/RestrictedFormats/iTunesMusicStore). Never-theless, these playback support limits shouldn't prove to be a problem— you can still add protected AAC files to playlists, copy them to your hard disk, and delete them from your iPod in Linux.

For a Linux diehard, however, there is one big problem with regard to encoded audio formats (and it isn't a limitation only in Linux): iPods do not support Ogg Vorbis files. There is talk that Apple might include support in future iPods, but I wouldn't hold my breath. In the meantime, you can quite simply convert your Ogg Vorbis files to MP3 format for use on your iPod, though there will be a bit of a loss in quality. (This is understandable, consider-ing that both formats have their own, incompatible methods of compression, which require the throwing out of actual music data.) Of course, if you origi-nally encoded your audio files in FLAC format, this quality issue will be moot. You'll learn more about this in "Converting Audio File Formats" on page 279.

iPod Filesystem Formats

All iPods have a formatted filesystem, just like your hard disk. In fact, with the exception of the Shuffle and the Nano, they actually have hard disks inside them. The filesystem format that is in place on your iPod depends on which system you originally used it on. If you first used it on a Mac, it will be in Apple's HFS+ file format. If you first used it on your Windows machine, it will be in Microsoft's FAT32 format.

Actually, in the short term, it doesn't matter which filesystem formatted your iPod; Ubuntu will usually mount either one, allowing you to browse through all the files on your little white beastie. If your iPod is HFS+ format-ted, however, browsing and exporting tracks is just about all you will be able to do dependably, assuming your system does mount it. It is important that your iPod be FAT32 formatted if you want to really use it as you would in Windows or, ironically, Mac OS.

Determining Your iPod's Format

How do you know whether your iPod is HFS+ or FAT32 formatted? Well, as I said, it is basically a matter of knowing which system you've been using your iPod with up until now. When you first plugged your fresh, out-of-the-box iPod into your computer, it really couldn't do anything yet. At that time, your Windows or Mac system popped up some wizard asking you to run the iPod Updater tool. That tool is primarily a formatter, which formats your device.

If you've been a two-OS sort of person up until now and have been happily using your iPod on both a Mac and a Winbox, then you can be sure that your iPod was formatted using FAT32, because Windows spews out chalk spittle when it tries to deal with anything that Microsoft itself didn't create. In other words, Windows can't read drives formatted by HFS+, while Mac OS can read both HFS+ and FAT32 drives. If you are using an iPod Shuffle, you can also be sure that it is FAT32 formatted, because all iPod Shuffles are—period.

Of course, if you're a prove-it-to-me kind of person, you can seek truth from facts by just plugging your iPod into your iTunes-equipped Mac or Winbox, and seeing what it says in the right pane of the iTunes window next to the word *Format*. If it says *Windows*, you're ready to go.

If you don't have a Mac or Winbox handy, you can check from within Ubuntu by going to the **Applications** menu and selecting **Accessories ▸ Text Editor**. Choose **Open** and **File System** in the left pane, double-click **etc** in the right pane, select the file **mtab**, and then click the **Open** button to reveal the format of your iPod. Just look for a line that says something like `/dev/sda2 /media/ipod` or `/dev/sdb2 /media/ipod` and see what is listed to the right of that. If it says `vfat`, then you know your device is FAT32 formatted. If it reads `hfsplus`, well . . . you're just going to have to change it.

Reformatting Your iPod

And how do you change your iPod from HFS+ to FAT32 format? Well, the easiest way is to first find a machine running Windows 2000 or newer and a recent edition of iTunes. But, before you go any further, make sure that you have everything on that iPod backed up, because you are going to be reformatting it, which means wiping it clean.

Once you are ready to roll, make sure the Windows machine is connected to the Internet, start up iTunes, and then plug your iPod into one of the Windows machine's USB ports. A window should appear telling you that you have a Mac-formatted iPod that must be restored before you can use it in Windows. Click **OK**.

To bring your iPod to a comfy state of Linux usability, click the **Restore** button in the right pane of the iTunes window. A warning window will appear telling you that everything on your iPod is about to be removed (hence my warning about backing up everything first). Click the **Restore and Update** button, and then follow along with the brief wizard that follows. Once the wizard is complete, sit back and wait for your iPod to be restored (and Windows-ized). This will take a couple of minutes, so enjoy the brief respite.

When the process is complete, a new window will appear asking you to name your iPod, and asking you whether you want iTunes to automatically synchronize your iPod with your photo and music collections (Figure 16-2). What you name your iPod is up to you (short names show up better on the GNOME desktop), but I strongly recommend unchecking the two auto-sync checkboxes; doing otherwise could lead to unwanted weirdness down the line. Once you're done, click **OK**.

Figure 16-2: Completing your iPod's HFS+ to FAT32 conversion process

Once the entire process is complete, you will have a properly configured, FAT32-formatted, and all-but-empty iPod that is ready for use in Linux, Windows, and Mac OS. You can even use your iPod now on all three systems interchangeably, though I would only do so if the auto-update function is disabled.

(Not) Auto-Updating Your iPod

When you enable auto-update on your iPod via iTunes, the function is set up within your iPod itself. With a FAT32-formatted iPod, you can use your iPod on a Winbox, Mac, or Linuxbox—or all of them interchangeably. If you set up your iPod to auto-update songs and playlists, however, you are leaving yourself open for trouble unless you have exactly the same music collection on your Linux, Mac, and Windows machines.

The reason for this is simple. While iTunes allows you to add files to your iPod, it does not allow you to copy files from it. The auto-update feature is thus strictly a one-way street. This means that whenever you hook up your auto-update–enabled iPod to an iTunes-enabled computer, iTunes will automatically add the tracks in its library to your iPod, and, more frighteningly, it will remove any tracks from your iPod that are not present in its library. I learned this the hard way when I took my wife's loaded iPod to work and plugged it into my office Winbox with its completely empty iTunes library. When I brought the little Podster back home to her with nothing at all on it, what ensued wasn't pretty.

If you have an iPod that is already in FAT32 format, it is best for you to disable the auto-update function on your iPod while it is connected to your Winbox, before bringing your iPod into the Linux world. To do this in Windows, start by connecting your iPod to your machine. In the left pane of the iTunes window, click your iPod in the Devices section, and then click the **Music** tab in the right pane. In that tab, uncheck the box next to the words *Sync music* (Figure 16-3). Next, if you have a photo-capable iPod, repeat nearly the same process for your photos by clicking the **Photos** tab in the right pane and then unchecking the box next to the words *Sync photos from.*
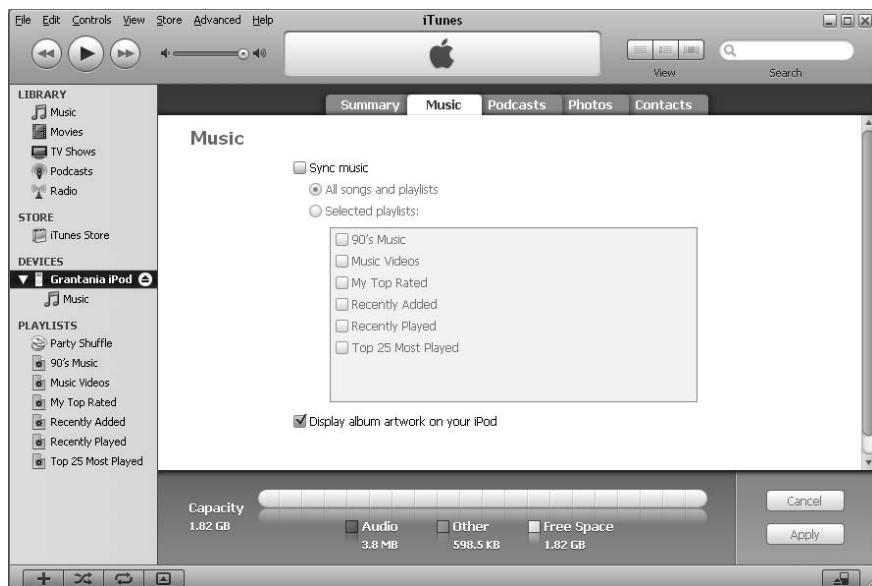
Figure 16-3: Disabling the auto-update function on your iPod via iTunes in Windows

Managing Your iPod in Ubuntu

Normally in Windows and Mac OS, you load files to your iPod via iTunes, but, of course, since Apple has not created a Linux version of that popular application (and I doubt it ever will), you will have to find some other way to go about things. Fortunately, there are a few Linux applications that can work with your iPod to various degrees, including one you learned about in Chapter 15—Rhythmbox.

Managing Your iPod's Audio Files with Rhythmbox

Rhythmbox is set as the default application for iPods in Ubuntu, and as such it will automatically appear once your iPod is connected to your machine and mounted by your system. All you should have to do then is click the icon for your iPod in the left pane of the Rhythmbox window to see its contents.

To copy a song from your library to your iPod, just click **Library** in the left pane, and then drag the artists, albums, or single tracks you want from the right pane to your iPod's icon in the left pane. The copy progress will be shown in an orange progress bar in the lower-right corner of the Rhythmbox window.

Copying Files from Your iPod to Your Hard Disk

One feature that isn't available in iTunes (at least not without the installation of a special freeware plugin) is the ability to copy songs from your iPod to your hard disk (or even to an external USB drive). Luckily, Rhythmbox, and in fact just about all Linux audio players with iPod support, allow you to do this quite easily.

To copy music from your iPod with Rhythmbox, select the tracks you want to export (hold down the CTRL key to make multiple selections or use the tabs to select whole artists or genres if you like), and then drag the selected items to your desktop, music folder, or wherever you want them.

NOTE   *This approach works well enough, but it doesn't work perfectly for AAC files (which end with a .m4a extension). While these files can be copied using the process just described, the copied files will not have their original filenames. The file* Over You AAC.m4a *on my iPod became* ZPAD.m4a *on my desktop. The files themselves are fine, though; in fact, if you add them to your Rhythmbox Library, they will appear with the original name and artist information. If you would like the files that you see in a Nautilus window to have a recognizable filename, however, just rename them as you would any file (right-click the file and select Rename—in case you forgot).*

Ejecting Your iPod

When you're done working with your iPod in Rhythmbox, you can eject it by either clicking its entry in the left pane of the Rhythmbox window and then clicking the **Eject** button at the top of the window, or by right-clicking it in the left pane and selecting **Eject** in the pop-up menu that appears.

Managing Your iPod's Audio Files in gtkpod

While the latest version of Rhythmbox should be able to meet your typical iPod needs, there are other apps that can do the job. For example, the audio players Songbird and amaroK (mentioned briefly in Chapter 15), have iPod support capabilities. There are also a few other applications that are iPod-dedicated, one of which is gtkpod. Shown in Figure 16-4, gtkpod is a pretty straightforward application with a fairly large user base, which means you should be able to get a lot of questions answered in Ubuntu and other Linux online forums should you have any. It also handles most of the functions that one uses when dealing with an iPod, including album cover art (but not photos). However, gtkpod is not an audio player; instead, it uses XMMS as its default playback program, though you can change the settings if you have another player in mind.

Figure 16-4: Managing your iPod with gtkpod

Downloading, Installing, and Running gtkpod

Of course, in order to use gtkpod, you first have to download and install it. You can do this via Synaptic by doing a search for *gtkpod* and then installing gtkpod and gtkpod-aac by following the installation steps in Project 5A on page 68. If you didn't install the multimedia codec packages in Project 15A on page 246, now would be a good time to do so. After that, plug your iPod into one of your computer's USB ports, if you haven't already. Once your iPod is mounted (once the desktop icon appears and Rhythmbox opens), go to the **Applications** menu and select **Sound & Video ▸ gtkpod**. You can close Rhythmbox if you like.

Using gtkpod

When the gtkpod window opens, you should see three entries in the left pane: Local, Podcasts, and iPod. To display the contents of your iPod, click the entry for your iPod in the left pane and then click the **Load iPod(s)** button. The contents of your iPod will appear in the right pane. If you have any playlists on your iPod, click the small arrow to the left of your iPod's name in the left pane of the window to reveal your lists.

To add new songs or folders full of songs, click the **Files** or **Dirs** buttons below the menu bar and then navigate to the items you want to add. You can also create playlists either by clicking **Playlist** to create a playlist of the contents of a particular folder, or by clicking the **New PL** button to build your own playlist.

When you click New PL, a new list will appear in the Playlists pane at the left side of the window. Drag the songs you want from the right pane to the new list in the left pane in order to build your list. (Just be sure to drag the files directly onto the playlist icon, as gtkpod can be a bit finicky in this department.) You can also add songs to existing lists in the same way.

While we're on the topic of dragging and dropping, it is worth noting that you can add songs to your iPod library by simply dragging them from your Music folder (or any other folder) and then dropping them in the right pane of the gtkpod window. To add a song to both your iPod's library and a specific playlist, drag a file or directory from your Music folder to the icon of the playlist in question. Pretty cool.

Setting Album Art and Editing Tags with gtkpod

Tag editing, including album cover images, is another area in which gtkpod can prove quite useful. To edit a tag, click any track in the playlist and select **Edit Track Details**; the Details window for that track will appear (Figure 16-5). In that window you can edit all of the tag entries (artist, album, track title, genre, and so on), and you can set the album cover art for the track. Once you have finished making your changes, click the **Apply** button, and then click **OK**.

Figure 16-5: Editing tags in gtkpod

Copying Files from Your iPod to Your Hard Disk with gtkpod

Like Rhythmbox, gtkpod can copy tracks from your iPod to your hard disk. To do this, select the tracks or playlists you want to export (hold down the CTRL key to make multiple selections, or use the tabs to select all of the tracks by a particular artist or in a particular genre). Then click the **File** menu, click **Export Tracks from Database,** and choose one of the destinations from the submenu: *Selected Playlist*, *Selected Tab Entry*, or *Selected Tracks.*

As with Rhythmbox, AAC files exported from gtkpod will not have the filenames they started out with. The remedy is the same as before: Just change the filenames manually. The correct track title and artist information will appear in any player window, whether or not you change the filename.

Playing Tracks on Your iPod with gtkpod

You can use gtkpod to play the tracks on your iPod, though gtkpod must use a helper application to do so, as it has no playback capabilities of its own. The default helper app for this purpose is XMMS, which isn't a bad choice, especially if you want to have headache-free AAC file support.

Of course, if you haven't installed XMMS, you will need to either install it or choose a different helper app, such as Rhythmbox or amaroK. To install XMMS, run Synaptic and search for *xmms* (and if you want AAC playback support, *xmms-mp4*). To change the audio helper application for gtkpod, open the gtkpod Preferences window by going to the **Edit** menu and selecting **Edit Preferences**. Once in the Preferences window, click the **Tools** tab, and then change the command in the uppermost box from `xmms` to `rhythmbox` or `exaile` or whatever app you prefer. Once your choice is made, click **Apply** and then click **OK** to record your changes and close the Preferences window.

Finishing Up the Job with gtkpod

Once you've done all you want to do with gtkpod, click the **Load iPod(s)** button to save any changes you've made to the files on your iPod. Then right-click your iPod's icon in the left pane of the gtkpod window and select **Eject iPod**. If the desktop icon for your iPod does not disappear after that, right-click it, and select **Eject**. When the desktop icon disappears, you can disconnect your iPod from your computer.

Floola and YamiPod

There is another pair of iPod management applications that might be of interest to you as alternatives to gtkpod: Floola and YamiPod. Both are relative newcomers to the iPod file management arena, but they are attractive and capable alternatives with significant followings. In addition to their clean interfaces, both allow audio-direct and helper-less playback (which gtkpod doesn't), and their playlist creation and handling is more refined.

Downloading, Installing, and Running Floola

As of this writing, neither Floola nor YamiPod are available via Automatix or Synaptic, so you'll probably need to download them from each project's website. To get Floola, go to www.floola.com and download the **linux (GTK)** version. When Firefox asks you what to do with the file, accept the default *Open with Archive Manager* by clicking **OK**, and then extract the Floola-linux folder to your home folder. You can run Floola by opening the Floola-linux folder and double-clicking the Floola binary file. Floola, as shown in Figure 16-6, should then appear.

Figure 16-6: An iPod management alternative to gtkpod—Floola

Downloading, Installing, and Running YamiPod

To get YamiPod, go to www.yamipod.com and download its Linux version. As with Floola, when Firefox asks you what to do with the file, accept the default *Open with Archive Manager* by clicking **OK**. Once the Archive Manager opens, extract YamiPod to your home folder.

Unlike Floola, you are not yet ready to use YamiPod. Before you can run YamiPod you will need to copy the file that begins with *libfmodex* within the yam-linux folder to the /usr/lib folder. Since /usr/lib is in protected territory, you will have to use the `sudo` command to perform the copy procedure.

Here are the steps:

1. Open a Terminal window, type `cd yam-linux`, and press ENTER.

2. In the same Terminal window, type the following command and press ENTER, making sure to leave a space on either side of `cp` and between `so.*` and /usr like so:

   ```
   sudo cp libfmodex.so.* /usr/lib
   ```

3. When prompted for your password, type it, and press ENTER. When your user prompt reappears, you can close the Terminal window.

Because YamiPod itself is an executable binary file, you don't have to install it to put it to work, and you are now ready to step into action. To get things under way, plug your iPod into one of your USB ports, and then double-click the YamiPod file within the yam-linux folder.

YamiPod will appear as a small window until it reads the contents of your iPod, after which the window should expand into something like the window in Figure 16-7.
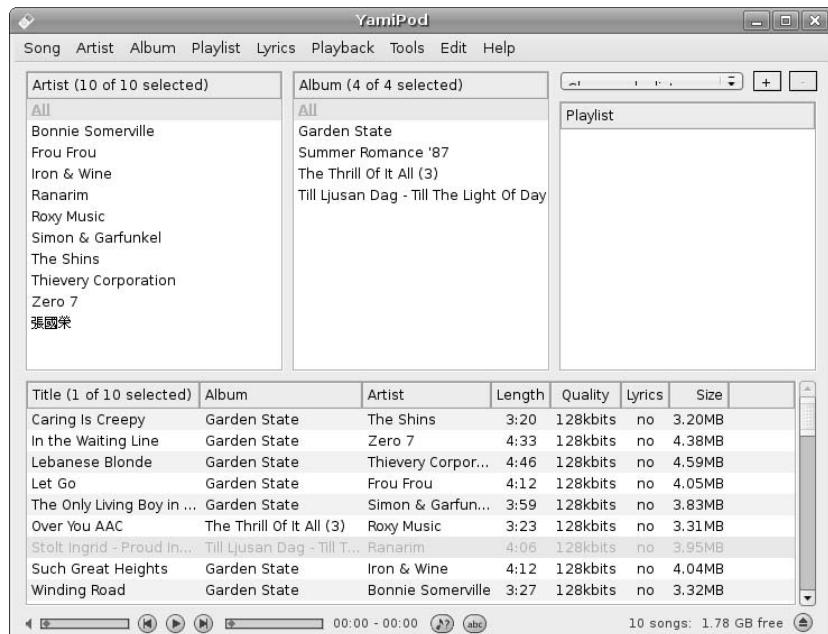
Figure 16-7: Managing your iPod with YamiPod

Setting Up Your System to Automatically Launch Floola, YamiPod, or gtkpod

You can set up your system so that YamiPod or gtkpod automatically launches when you plug your iPod into one of your USB ports. To do this, go to the **System** menu and select **Preferences ▸ Removable Drives and Media**. When the Preferences window appears, click the **Multimedia** tab, and then check the box next to the words *Play music files when connected* in the iPod section of that tab. In the Command box, type the name of the application that you prefer (`gtkpod`, `Floola`, or `YamiPod`), and then click the **Browse** button and navigate to the Floola or YamiPod binary file within your home folder. Once you're done, the Preferences window should look something like Figure 16-8. If it does, click **Close**.
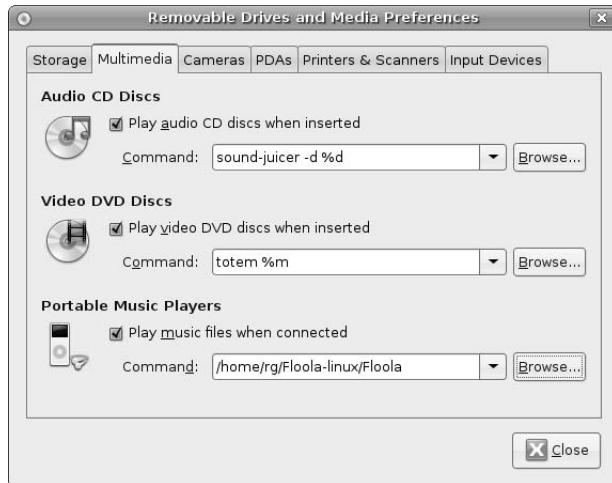
Figure 16-8: Setting up your system to launch YamiPod automatically when you plug in your iPod

Photo Transfer with GPixPod

Ever since the iPod Photo appeared in 2004, iPods have been able to display photos. Unfortunately, until recently there was no simple way to transfer photos from your computer to your iPod in Linux. Things have, quite fortunately, changed since the appearance of a handy application called GPixPod. GPixPod (Figure 16-9) is, admittedly, still in development.

You can get GPixPod via Synaptic by doing a search for *gpixpod*. Once it's installed, you can run it by selecting **Applications ▸ Graphics ▸ GPixPod**.

Figure 16-9: Adding photos to your iPod with GPixPod

You cannot add photos to the GPixPod interface without first connecting your iPod to your computer. You should also note that you cannot view images added to GPixPod's database in the right pane of the application window until you save your changes by clicking the Save button.

Converting Audio File Formats

SoundConverter (shown in Figure 16-10) makes it easy to convert MP3 files to Ogg Vorbis format or vice versa. Because SoundConverter does not come bundled with Ubuntu, you will have to install it if you haven't already done so. To do so, just run Synaptic, search for *soundconverter*, and install the application.

To use SoundConverter, just select **Applications ▸ Sound & Video ▸ SoundConverter**. Once SoundConverter starts up, add the songs you want to convert to the main pane by clicking either the **Add File** or **Add Folder** buttons. You can also drag audio files from your Music folder (or wherever else you store them).

Figure 16-10: Converting audio file formats with SoundConverter

Once you've chosen the files to convert, you need to choose which format to convert them to. You can do this by selecting **Edit ▸ Preferences** and then making your choice in the Preferences window (Figure 16-11). While you're there, it is also a good idea to tell SoundConverter to place your converted files in a location other than the folder where the original files are stored. Doing this prevents having to deal with duplicates in Linux audio playback applications that automatically scan your Music folder, such as Rhythmbox. (This is not an issue in either gtkpod or YamiPod, as they will not allow you to add Ogg Vorbis files to their library lists.) You might also want to check the box next to the words *Create subfolders* in order to keep things organized.

Once you have set things up and are ready to convert, click **Close** in the Preferences window, and then click **Convert** in the main window. Sound-Converter will then begin doing its stuff.
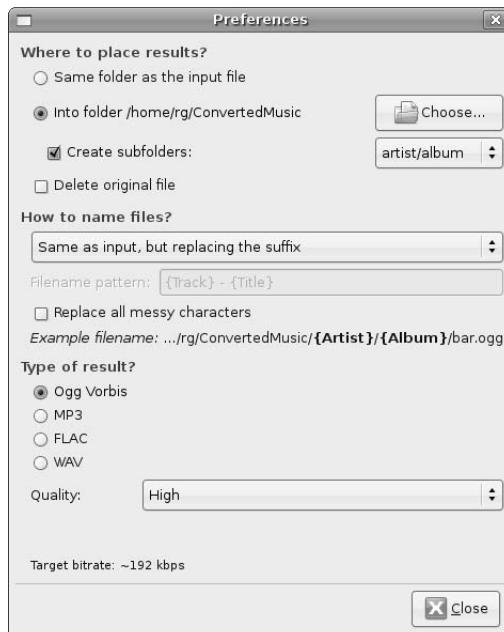
Figure 16-11: Setting conversion preferences in SoundConverter

Linux on Your iPod?

One thing you might notice if you do a Google search for information about using your iPod with Linux is that it is possible to actually replace your iPod's Apple-designed operating system with a form of Linux. Now, I am not advocating that you do this (and I most certainly have not done so myself), but some folks are interested in geeking around with whatever gadget they have in their hands. If you're such a person, or just curious, check out www.ipodlinux.org.

Other MP3 Players

Although Apple's iPod kind of dominates the portable audio player marketplace, it is not the only kid on the block. There are numerous other players out there, many of which can be used in Linux with varying degrees of simplicity; more information is available at http://tuxmobil.org/portable_players .html. If you have one of Creative Lab's Zen or Nomad players, the Linux application Gnomad makes things easy for you. Gnomad is available via Synaptic, and can be run, once installed, from **Applications ▸ Sound & Video ▸ Gnomad 2**.

17

COUCH PENGUINS

Video and DVD Playback in Ubuntu

Now that we've covered much of what Ubuntu can do in terms of audio, let's turn our attention to what is arguably the second most important of its talents in our CNN/MTV–era world: video. Ubuntu is quite capable in terms of video playback, allowing you to view video files you download from the Internet or from your digital movie camera, video CDs (VCDs), unencrypted DVDs (encrypted ones, too, if you installed the codec packages via Automatix in Chapter 15), and some Internet video streams. It even allows you to download movies from your digital video camera and then edit them.

Playing Video Streams with RealPlayer

In Chapter 15, we covered RealPlayer's role as an audio application that allows you to play Internet audio streams. Audio streams are not the only thing that RealPlayer can handle, however; you can use it to play video streams as well. If you would like to try out RealPlayer's capabilities for playing video streams, or if you just happen to be a fan of Katie Holmes, point your web

browser to www.katieholmespictures.com/movies.shtml, scroll down to the Videos section, and then try out one of the many clips available there. Once you have made your selection, RealPlayer will pop up and begin playing the stream in a slightly enlarged window.

If the Beatles are more your cup of tea, try "The Birth of Beatlemania" at NPR's All Songs Considered site (www.npr.org/programs/asc/archives/beatles40). Be sure to click the **Real Video** link on that page. Firefox will then pop up one of its what-should-I-do-with-this-thing windows with Movie Player listed as the default player. Movie Player doesn't do as good a job with RealMedia streams, so you definitely want to change things. In that window, click the drop-down menu button, which at this point says *Movie Player*, and select **Other**. When the Choose Helper Application window appears, double-click **File System** in the left pane of the window, navigate your way to /usr/bin in the right pane, and then scroll down until you find the file realplay. When you find it, click it once to select it, and then click **Open**. Once back in that original what-should-I-do window, check the box next to the words *Do this automatically for files like this from now on*. Your window should look like that in Figure 17-1. If so, click the **OK** button, after which RealPlayer will start up and play the stream.
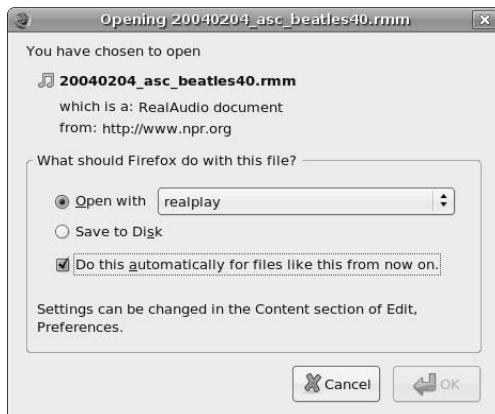
Figure 17-1: Telling Firefox what to do with the RealAudio document

If you would like to view the video at an enlarged size, you have two ways of going about it. The first and easiest way is to double-click the title bar of the RealPlayer window, which will maximize it. The second, equally easy way is to go to the **View** menu and select **Fullscreen** or **Zoom ▸ Double Size**.

DVDs

Your system also allows you to play DVDs; however, due to licensing concerns, playback is limited to unencrypted disks. Unfortunately, this rules out a vast majority of the DVD movies you buy or rent at your local video shop and leaves you with a rather limited choice of movies that you can play on your computer. Of course, DVDs created on personal computers (such as the DVD slideshows that people create these days with applications such as iDVD

on the Mac) pose no problem. Given the limited offerings in the unencrypted DVD world, you will no doubt want to enable your system to play back the encrypted variety, and, assuming you installed the two codec packages via Automatix back in Project 15A, you now can.

Can I Play Foreign DVDs?

Your computer can play DVDs of any broadcast standard (NTSC, PAL, or SECAM) and of any regional encoding. This is a better setup than that DVD player you have hooked up to your TV, because the vast majority of stand-alone DVD players in the United States (and I would venture to guess that the size of that majority is 99.9 percent) do not allow you to play anything other than Region 1 NTSC disks (NTSC being the broadcast standard in the United States and what its televisions are designed to display, while the DVD region is 1). This information is usually provided on the back of DVD packages (see examples in Figure 17-2), though the packaging for most disks produced for the US market does not include it.

Figure 17-2: Examples of regional encoding labels on DVD packages

Despite the wonderful everything-goes nature of your computer in terms of DVD playback, there is a serious caveat to bear in mind. Depending on the manufacturer of your DVD drive, you will only be able to switch back and forth between DVDs of differing regional encodings four or five times. After that, the drive will be locked into the regional encoding of the disk you were playing at that time . . . *forever.* This is unrelated to your operating system—it is strictly a hardware matter. The only exception to this region-lock rule are those DVDs labeled *Region Free* or *ALL* (sometimes inaccurately labeled as *Region 0*), which can be played on any DVD player in any region, and thus do not register as a regional encoding switch when you plop one of them in your computer's DVD drive.

If your drive does eventually lock into one regional encoding, especially one for which you have few DVDs, there is some good news. That news comes in the form of Videolan's libdvdcss2, the library that comes bundled with the two codec packages you installed in Project 15A, and which allows you to play back encrypted DVDs. In addition to that primary function, libdvdcss2 also, in theory, allows you to play back DVDs from multiple regions even if your DVD drive is already locked into one region. It does this by performing a *cryptic attack* (to use Videolan's term for it) on your drive until it can find the disk key for that drive. Of course, this process of cryptic bombardment can take several minutes, so it is not the optimal way of going about things. Better than being stuck, though. Whether or not this process of bypassing regional encodings is legal remains a subject of debate, so if you are concerned about such things, you should do a little research.

Totem Movie Player

The default video player in Ubuntu is Totem, which, as you can see in Figure 17-3, has a very simple interface that makes using it equally simple. You can run Totem by going to the **Applications** menu and selecting **Sound & Video ▸ Movie Player**. You can also run it by simply placing a DVD in your drive, because Ubuntu is set up to run Totem any time you do so. If you don't happen to have a DVD on hand but you want to see Totem in action, you can also bring it up by double-clicking that Ubuntu video that comes with your system in the Examples folder (Experience Ubuntu.ogg).

Switching Totems

Before going on any further, I should mention that if you installed the two codec packages back in Project 15A, you are not using the same Totem you were when you first installed your system. How can that be, you ask? Well, despite all that simplicity I talked about, Totem can be an absolute pain in the proverbial posterior at times. You may be one of the lucky ones, but for me and many others, Totem can be as cooperative as a cat on a leash (mine excluded, of course). The problem is not actually Totem itself, but rather the gstreamer back end that performs its video processing.

Figure 17-3: Totem Movie Player

Totem, you see, is merely a graphical *front end* that provides you with an easy way to use software that works in the background—a *back end*, which in this case is gstreamer. When you installed the multimedia and DVD codec packages with Automatix, that gstreamer version of Totem was replaced with

a version of Totem that utilizes the Xine video-processing back end instead. The reason Automatix automatically performed the switch is that its designers know that Xine is a much more cooperative beastie when it comes to video.

If you haven't made the switch and find yourself having problems with Totem as is, try installing the Xine version along with the other multimetia codec packages. The instructions for doing this are given in Project 15A, but to save your fingers some wear and tear, I'll repeat the procedure for you here. Just run Automatix (**Applications** ▸ **System Tools** ▸ **Automatix**), and, once the Automatix window opens, click **Codecs and Plugins** in the left pane of window, and click **AUD-DVD Codecs** and **Multimedia Codecs** in the right pane. Finish up by clicking the **Start** button.

Once you're done, you won't notice any changes in Totem up front. You will have the exact same Totem interface in the exact same menu location, but you'll most likely be a much happier camper because Totem should then work as it's supposed to. You will experience, among other things, superior sound (or at least better volume), more dependable DVD playback, and less quirky playback of Windows Media Video (WMV) files. If it's still giving you trouble after that, try one of the other playback applications mentioned at the end of this chapter.

Using Totem to Play DVDs, VCDs, and Other Multimedia Files

As I already mentioned, you can play a DVD in Totem by simply placing your DVD in the drive, after which Totem will open and begin playing your movie. If you've got a copy of *Red Detachment of Women* on VCD that you're aching to watch, you can do so in the same way. Just pop the VCD in your drive; Totem should start up and begin playing it.

Totem not only plays DVDs and VCDs, but it can also play MPEG files and, because you installed that big cocktail of packages I keep going on about, it can play WMV files too. You can play such files by either double-clicking them directly or going to the Totem **Movie** menu, selecting **Open**, and then navigating to the video file you want to view.

Making Things Look a Bit Better in Totem

One thing you may notice when using the Xine version of Totem is that videos seem to look a bit washed out. This can be easily fixed by going to the Totem **Edit** menu and selecting **Preferences**. In the Totem Preferences window, click the **Display** tab, and then lower the **Brightness** and **Contrast** sliders to points similar to those shown in Figure 17-4. You can, of course, change things to suit your own tastes. Once you're done, click **Close**.
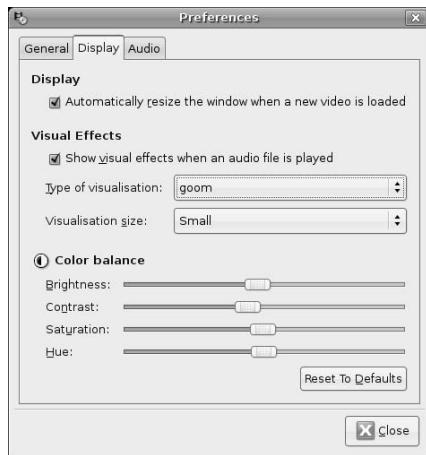
Figure 17-4: Adjusting brightness and contrast settings in Totem

Totem as an Audio Player?

You may have noticed while in the Display tab of the Preferences window that there was a Visual Effects section. Well, those visual effects aren't for the videos you play, but rather are visualizations to accompany your audio files when played via Totem (Figure 17-5). Yes, Totem not only does video, it does audio as well. In fact, it is, somewhat oddly, the default audio player for Ogg Vorbis and MP3 files in Ubuntu. Just double-click one of those files, and sure enough, Totem will be the application that pops up, blasting your ears out with your favorite melodies. Of course, you can also play such files from the Totem **Movie** menu, by selecting **Open** and then navigating to the songs you want to play. If you have a CD in your disk drive, you can even use Totem as a CD player by going to the **Movie** menu and selecting **Play Disc 'Audio Disc'**.

Figure 17-5: Totem as an audio player

A Couple of Other Cool Totem Features

There are a couple of other cool features in Totem that you might like to know about. One is its ability to perform screen captures of whatever video you happen to be viewing at the time. Just go to the **Edit** menu, select **Take Screenshot**, and you've got yourself a screen capture—a still image taken from a video file.

Another feature worth mentioning is Totem's Sidebar. As you no doubt noticed, at the bottom-right corner of the Totem window, there is a Sidebar button. If you click that button, a new pane will open at the right side of the Totem window. In that pane you can load, create, and save playlists. Such lists can consist of any combination of supported video or audio files, thus providing you with the whole multimedia banana. Any time you play a file in Totem, that file appears in the Playlist pane, but you can also add items to the list by simply dragging the files there from your desktop or any Nautilus window.

Using Your Digital Video Camera

Although they haven't achieved the ubiquitous status of digital still cameras, digital video cameras have become increasingly common in recent years. Likewise, while Linux support for still digital cameras is quite good, its support for DV cameras can be called . . . well, let's just say it's "progressing."

Don't be discouraged by my tone in that last sentence, though. You should have no trouble downloading video from your camera to your computer, editing those video files, and adding effects and even subtitles. To be honest, there are still some problems, especially in the area of file format conversions, but, as with all things Linux, it will only be a matter of time until the wrinkles are ironed out. There are also a couple of cool video editing apps that, while not quite ready for prime time, seem promising and are well worth keeping an eye on: PiTiVi and Diva.

For the time being, however, the application of choice for the digital video camera user is Kino. Kino, shown in Figure 17-6, is an iMovie-like application with which you can capture video from your camera and then edit it. To install Kino, run Automatix, click **Media Players and Editors** in the left pane, click **Kino** in the right pane, and then click the **Start** button. Once it's installed, you can run Kino by going to the **Applications** menu and selecting **Sound & Video ▸ Kino**.

Capturing and Editing Digital Video with Kino

Once you have gone through the preparatory steps I've just mentioned, you are ready to capture video from your camera. To do this, connect your camera to your computer by FireWire cable (if it isn't still connected), turn on your camera to Play mode, and then start up Kino. Once Kino is open, click the **Capture** tab to the right of the playback pane.

Figure 17-6: Using your digital video camera with Kino

To get started capturing video, you can use the playback controls located below the playback pane. These control buttons actually control the functions of your camera itself. Start out by clicking the rewind button until you get to the beginning of the video segment you want to capture. Once you get there, click the play button, after which the video on your camera will play back within the Kino window. When you reach the point at which you want to start capturing, click the **Capture** button just above the play back controls in the Kino window. Kino will then start capturing your video to disk (in your home folder by default). To keep things easy to manage, the captured video stream will be split into several files, the number of which depends on the length of the video played.

You will notice that after you click the Capture button, the video playback in the playback pane will stop moving. This being the case, you will need to view the action in your camera's viewer in order to know where you are in the capture process. When you get to the point where you'd like to stop capturing, click the **stop** button. You can then view the captured video by clicking the **Edit** button and then using the playback controls at the bottom of the playback pane. You can also view the video in Timeline view (as shown in Figure 17-7) in order to navigate between the various segments of the video; click the **Timeline** tab, click the segment you wish to view, and then use the playback controls below the playback pane.
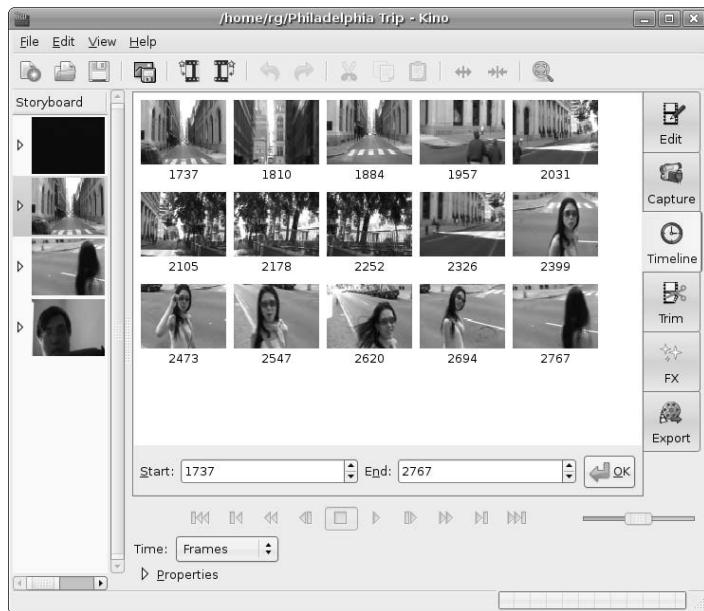
Figure 17-7: Kino's Timeline view

If you feel like getting a bit arty, you can also try out the effects available in Kino (some examples of which are shown in Figure 17-8) by clicking the **FX** tab and then playing around with the various effects in the drop-down menu below the words *Video Filter*. Make your choice, specify the segment you'd like to convert (or at least experiment with) by typing the beginning and ending frame numbers in the boxes below the word *Overwrite*, and then click the **Preview** button to see the results without saving the changes to disk. If you do want to convert the segment so as to keep the effect, click the **Render** button, and Kino will create a new file of just that segment. Those files, as well as the original captures, can all be viewed in Totem, which is a better application to use for video viewing, by simply double-clicking the files.

Figure 17-8: Examples of Kino's video effects, before and after

Other Video Apps

I've covered the main video applications in Ubuntu, but there are still others available that you might want to consider. A very popular alternative video/DVD player, for example, is MPlayer, which you can download and install by performing a Synaptic search for *mplayer*. Be sure to download not only mplayer itself, but also mplayer-fonts so as to avoid annoying error messages. Mplayer is also available via Automatix.

The cross-platform player, VLC Media Player, is another player with a significant following. It is noted for handling a number of video formats well, and for having a simple and easy-to-use interface. You can get it via Automatix (click **Media Players and Editors** in the left pane, click **VLC Media Player** in the right pane, and then click **Start**).

Another video/DVD player you might want to consider is gxine, which is similar in several ways to Totem, but it gives you a bit more control over things. Again, you can download and install gxine via Synaptic by performing a search for *gxine*.

If you are not only a video watcher but also a budding video artist, then you might want to consider a few handy video/DVD authoring and editing tools: the multiplatform video editor, Avidemux (Figure 17-9); QDVD-Author, a DVD creator application in the iDVD vein (albeit without the superficial polish of Apple's offering); and DVD::Rip, which, as you might imagine, allows you to rip the contents of a DVD to your hard disk. These applications are available via Synaptic by performing searches for *avidemux*, *qdvdauthor* and *dvdrip*. Avidemux and DVD::Rip are also both available via Automatix.
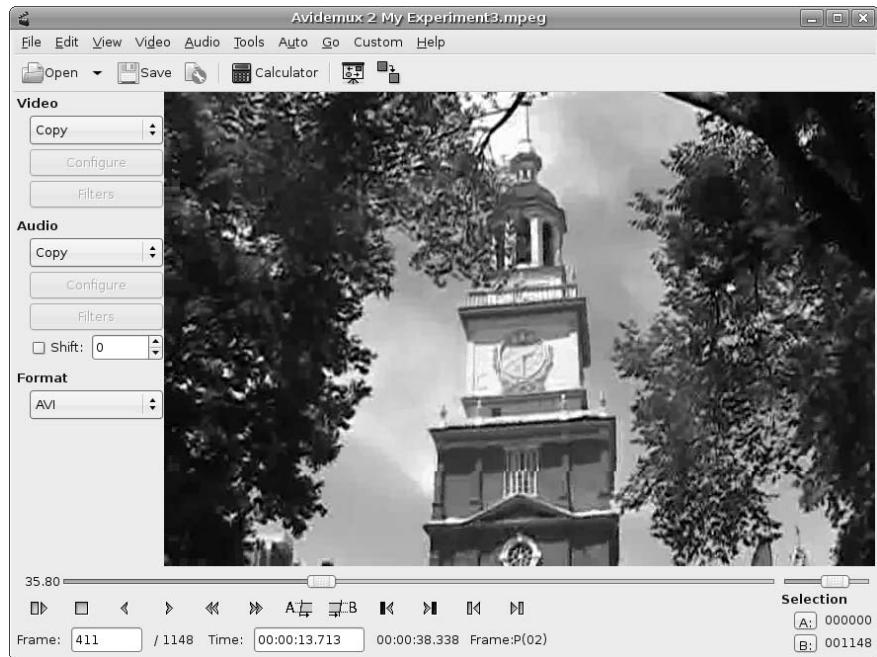
Figure 17-9: Cross-platform video editor, Avidemux

18

DEFENDING THE NEST
Security

Many a Windows user has entered the Linux fold after a host of bad experiences with *malware* in the Windows world—viruses, spyware, and all sorts of other malicious bits of software code, too numerous to imagine. Windows is also plagued by a seemingly endless array of security vulnerabilities, leaving the system easy prey to invaders with less than noble intentions. Every trip out into cyberspace thus becomes something like a run through the infectious diseases ward of a hospital. For a Windows user, it can sometimes seem that more time is spent ridding the system of viral pests and defending it from invaders than is actually spent getting things done.

Fortunately, Linux does not suffer greatly from such problems, leading to the much-touted claim that Linux is practically virus free and quite secure. There are numerous lines of reasoning proffered to explain Linux's malware- and exploit-resistant nature. One reason is simply popularity—or lack thereof. As Linux is not as widespread a system as Windows is, it is also a much less attractive target of digital evil-doers, who very often seem to be motivated by the challenge and headline-catching glory that comes with creating a truly global virus or finding a theretofore unknown back door.

Another reason is that Linux users, as a general rule, work on their computers in a non-privilege mode, one in which the user does not have the right to install software without a password. This is not the case, at least not by default, in Windows. A virus or other form of malware attached to an email or piggybacked upon another file or application cannot, therefore, install itself in your Linux system without that password . . . well, theoretically, at least. Of course, now that Windows uses its own privilege structure, this point is a bit less of an issue.

There is also the matter of structural design. Every system out there has security holes that can be exploited by digital and human foe alike. Windows might well be called the Swiss cheese of operating systems in this regard. Of course, Linux has its holes too, though far fewer of them than Windows does; and Linux plugs them up through downloadable updates faster, once they are found.

Finally, there is the defensive edge that Ubuntu's pre-configured security policy brings—there are no open incoming ports in Ubuntu desktop systems. This means that your Ubuntu Linux system is even less susceptible to unwanted intrusions.

Does My System Need Protection?

So with all this talk about Linux's great security, you may wonder whether you need to bother worrying about it at all. Well, if you take a look at the Ubuntu forums, you might find yourself a bit confused. When asked whether Linux users need to install antivirus software or firewalls, most users answer with an emphatic *no*. On the other hand, you'll find that there are an awful lot of people out there who have installed or are trying to install that software. Hmmm.

So what's a Linux user to do?

If you are on a network where you transfer a lot of files among a lot of Windows machines, you might want to think about installing some antivirus software, if for nothing more than the good of the Windows systems involved and the users of those systems—your unenlightened (i.e., Windows-using) email pals, for example. You might also want to give it a go if you are, by nature, on the cautious side of the spectrum. Basically, if it makes you feel safer to install some protection, go ahead. If it makes you feel safer to go whole hog and install the full line of defense mechanisms I cover in this chapter, go ahead. After all, either way you go, it isn't going to cost you anything, and it certainly isn't going to hurt you any.

The First Line of Defense

Regardless of the system you happen to be using (though I am assuming that you have become a Linux devotee by now), the first line of defense for any computer permanently hooked up to the Internet is a *router*—an electronic

device that allows a number of computers on a local network (such as in your home or at your office) to connect to and share a single connection from your Internet service provider (ISP). While the router is connected to your modem via cable, the connection from the router to the computers on your local network can be wired, wireless, or both.

What has a router got to do with the defense of your computer? Well, most routers include a *firewall*, which essentially functions to keep all of the bad stuff out there on the Web away from your computer, much in the way that the firewall in your car keeps the heat, fumes, and noise from your engine out of the passenger compartment. This built-in firewall is one reason that even people with only one computer, who could just as easily connect their computer directly to their cable or DSL modem, use a router. Of course, just how much security the firewall in your router provides depends on which filters you select in the firewall setup software. For example, a very common and useful filter (particularly for those with a wireless network) limits Internet access to those machines specified on the firewall's access list. This prevents your next-door neighbors from hitching a wireless ride via your ISP connection. They aren't paying the bill, after all.

The setup software for a router is built in to the router itself, so you don't have to worry about software installation and system compatibility. Using your router with Linux is no different than using it with Windows. You can access the software and modify your settings via a simple web browser, as you can see in Figure 18-1. Just type the IP address of the router (usually provided in the owner's manual) in the browser's location bar, press ENTER, and you'll be ready to go.
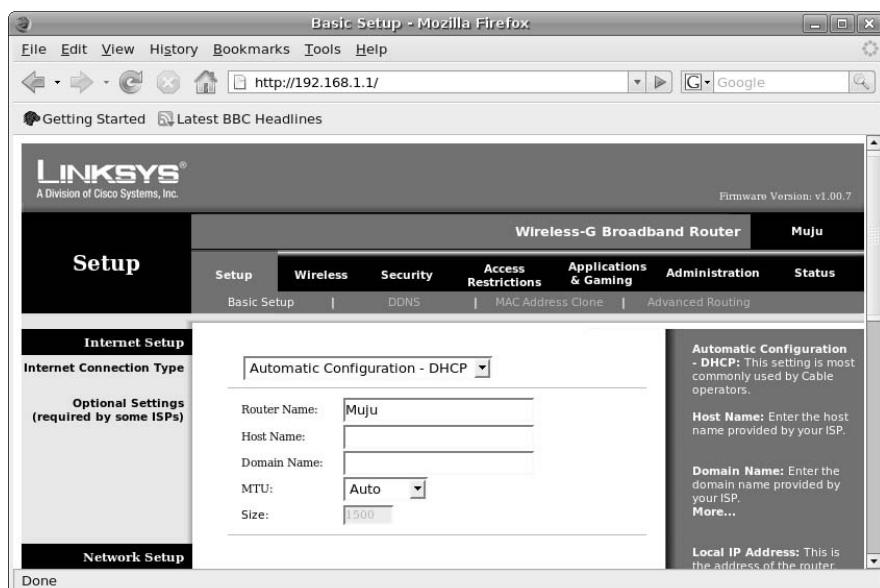
Figure 18-1: Setting up a router

Software Firewalls

If you don't have a router and don't plan on getting one, or if you have one, but you are bordering on paranoia, you might want to consider using a software firewall, in particular one of the most popular software firewalls available for Linux, Firestarter.

You can download and install Firestarter via Automatix by clicking **Utilities** in the left pane, selecting **Gnome Security Suite** in the right pane, and then clicking the **Start** button. Once it is installed, run Firestarter by choosing **System Administration ▸ Firestarter**. You will then be prompted for your password, so type that, and click the **OK** button.

Firestarter will open with a pretty self-explanatory setup wizard the first time out. If the wizard doesn't seem all that self-explanatory to you, and you're not sure what to do, just accept the default settings by clicking the **Forward** button in each of the wizard screens until you get to the last one (shown in Figure 18-2). In that screen, make sure that the box next to *Start firewall now* is checked, click the **Save** button, and then click **Quit**. When the wizard is finished, the main Firestarter window will appear and, assuming the correct network device was detected and selected, the firewall will be up and running.
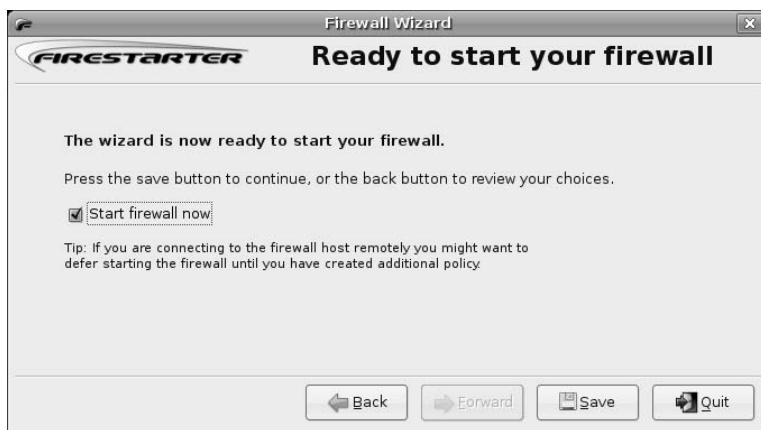
Figure 18-2: The last screen of the Firestarter setup wizard

If the wrong network device was selected, a warning window will appear telling you so. In that case, you can do a bit of trial-and-error manipulation by clicking the **Preferences** button in the main Firestarter window and then clicking **Network Settings** in the Preferences window that appears (Figure 18-3). In that window, select one of the other devices listed in the drop-down menu next to the words *Detected device(s)*, and then click the **Accept** button. Once back at the main Firestarter window, click the **Start Firewall** button, and see what happens. If you still can't start the firewall, repeat the process I've just described, this time selecting a different network device.
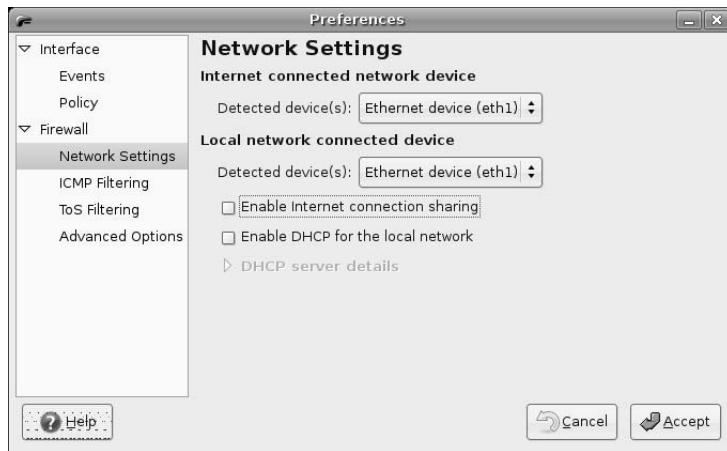
Figure 18-3: Selecting network devices in Firestarter

Once your firewall is up and running, there is nothing more that you really need to do. You can simply look at the Firestarter window (Figure 18-4) to see what is going on network-wise on your computer—what active connections you have, how much information has been coming and going, and if there have been any events in which, for example, the firewall has blocked an intruder. If you click the **Events** tab, you can then see the details of those events, such as what connection attempts were blocked, where they came from, and when they happened.

Figure 18-4: Firestarter in action

Taking Control of Firestarter

You can control how Firestarter deals with various network events by creating your own *policies*. The default policy set in Firestarter allows you to basically do whatever you normally do via the Internet, while it blocks new connections to your computer from the Internet or any other computer on your network.

To make things a bit more draconian, you can click the **Policy** tab, select **Outbound traffic policy** in the drop-down menu next to the word *Editing*, and then select **Restrictive by default, whitelist traffic**. If you just want to deny anyone working on your computer access to a specific website, for instance, simply click the **Deny connections to host** field to select it, and then click the **Add Rule** button. In the Add New Outbound Rule window, enter the URL for the targeted site, click **Add**, and then click the **Apply Policy** button in the main Firestarter window.

Confirming That Firestarter Runs Automatically

After you run Firestarter the first time, it will set itself to automatically start up whenever you start up your system. Don't be concerned when you don't see the graphical interface you saw when you first started it up; Firestarter will be running in the background, silently protecting your computer.

If you are the doubting type, you can check to see whether or not Firestarter actually is running in the background by opening a Terminal window, typing `sudo /etc/init.d/firestarter status`, and then pressing ENTER. If Firestarter is running, you will see the message `* Firestarter is running. . .` in the Terminal window. Worries over. If Firestarter isn't running, the response will read `* Firestarter is stopped.`

Finding Out More

If the world of firewalls is new to you, you can check out the Firestarter home page to learn a bit more. To check out the online manual, just to go to the Firestarter **Help** menu, and select **Online Users' Manual**, which will bring up the page in your web browser. If you prefer to check out the manual before installing Firestarter, point your browser to www.fs-security.com/docs. You will also find a pretty good quick tutorial there.

ClamAV: Antivirus Software, Linux Style

Despite the lack of viruses out there that can wreak havoc upon your Linux system, your computer could still act as a transmitter of Windows viruses. As a result, there are a number of free antivirus scanners out there for Linux users interested in helping to protect Windows users from viruses. These include Aegis (available via Synaptic), Panda Antivirus (www.pandasoftware.com/download/linux.htm), f-Prot (www.f-prot.com), and numerous others. For most Linux users, however, the virus scanner of choice is the open source contender—ClamAV.

Although it can be used on a number of operating systems, ClamAV is considered to be *the* Linux antivirus software package. It is open source, totally free, and you don't have to worry about licenses or suffer the bother of renewing them. Unfortunately, on its own, ClamAV is a command-driven application, which makes it a bit less user-friendly. Fortunately for all involved, there is also a graphical interface available, albeit a simple one, by the name of ClamTk. Both ClamAV and ClamTk are available via Automatix. In fact, they come bundled together with Firestarter in the Gnome Security Suite, so if you want to give them a try, just follow the directions for installing Firestarter in the previous section.

Using the ClamAV/ClamTk Duo

Once ClamAV is installed, you can perform a virus scan by going to **Applications ▸ Accessories ▸ Virus Scanner**. This will bring up the Clam Tk Virus Scanner window (Figure 18-5). After it appears, you should set up ClamAV so that it will automatically quarantine any viruses it finds. To do this, go to the **Actions** menu, and select **Quarantine Infected Files**. Then go to the **File** menu, select **Scan Directory**, and, in the left pane of that window, double-click the icon for the folder or disk you want to scan. Once you've done that, click the **OK** button, and ClamAV will start scanning your system. ClamTk will let you know what it (or, rather, ClamAV) is scanning at any given moment in the empty space just below the button bar. If it finds anything suspicious, it will list that item in the main pane of the window.
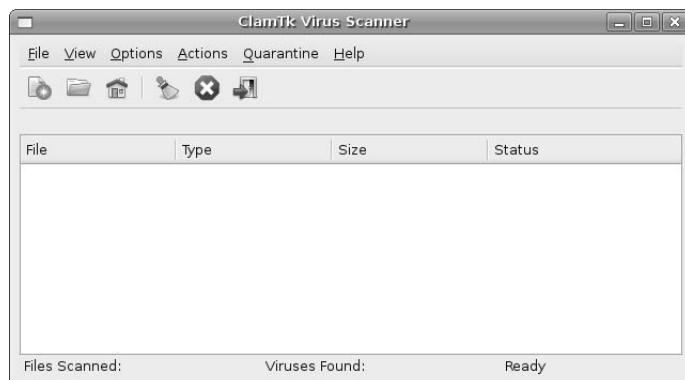
Figure 18-5: ClamTk—the graphical interface for ClamAV

As I mentioned before, the only viruses you are likely to encounter during a virus scan are those designed for Windows systems, and because those viruses cannot affect your Linux system, focusing your efforts where they are likely to do the most good seems a sound way of doing things. Although it won't hurt you any to scan your entire system for viruses, especially if doing so makes you feel better, it is probably better to focus your virus scanning activities on your Windows partition, if you have one, and on any files that you will be sending as email attachments, particularly to Windows users.

Project 18: Virus Scanning with avast!

While ClamAV may be the virus scanner of choice in the Linux world, my personal favorite is still avast!, which defended me during my Windows years. I think it is easier to use and, for what it's worth, nicer to look at. On the downside, it does require you to register every year so that you can get a license key, but that is hardly a monumental task, and it is still free—at least for home and noncommercial use, which, if my target audience for this book was correctly predicted, should include you. If nothing else, it is worth a try, so just go for it.

18-1: Downloading the avast! DEB Package and License Key

First you have to get the avast! file and license key. You can do this by opening your web browser and going to www.avast.com/eng/download-avast-for-linux-edition.html. Once you're there, click the **Download** button for the DEB package of the avast! Linux Home Edition. When the Firefox what-do-you-wanna-do-with-this-file window appears, accept the default (gdebi-gtk) by clicking **OK**. Once the download is complete, you can, of course, save the package to your hard disk and double-click it to open the Debian package installer.

When the Package Installer window appears, click the **Install Package** button, provide your password when prompted, and click **OK**. The download and installation will then begin. When these processes are complete, close the notification window and the Package Installer itself. Once you've downloaded and installed avast!, you'll need a license key in order to use it. Just go back to the avast! Download page in Firefox, click the **registration form** link, and then fill out and submit that form. You should receive your license key by email a few minutes later, even though the page says it could take up to 24 hours.

18-2: Running and Using avast!

Once avast! is installed, you can run it by clicking the Run Application panel applet (or pressing ALT-F2), typing **avastgui** in that window, and then clicking the **Run** button. The first time you run avast!, a small window (Figure 18-6) will appear asking you to input your license code. Assuming you registered your free copy of avast! at the beginning of the project as I instructed you to do, your registration key should already be waiting for you in your emailer's inbox. Go have a look, copy the key, and paste it into the text field in the Registration window (you'll have to use the CTRL-V key combination to do this). Once the license key is in place, click the **OK** button.

*Figure 18-6: Entering your avast!
license key*

The main avast! window (Figure 18-7) will now appear, but before you get started scanning away, it is a good idea to check for any updates to your virus database. New viruses are constantly appearing in cyberspace, so you should keep avast!'s virus database up to date so that it recognizes any viral newcomers. To perform the update, just click the **Update database** button, and let avast! do its thing.
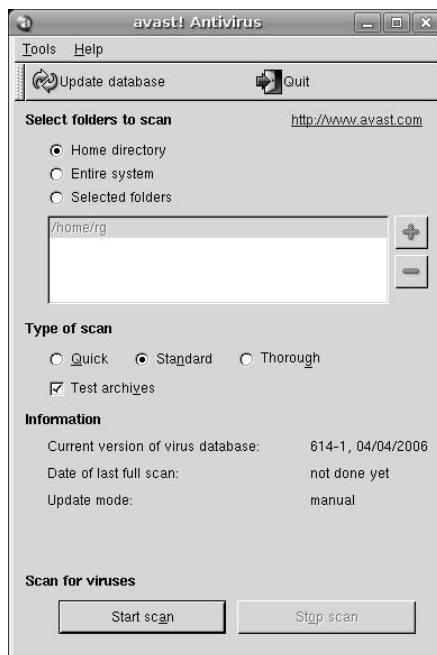
Figure 18-7: avast! virus scanner

When avast! is done updating the virus database, you can get on to the work of virus scanning by first deciding whether you want to scan just your Home directory, the entire system, or selected folders. You can also decide how thorough a scan you wish to perform via the three choices available in the middle of the window: Quick, Standard, or Thorough.

Which one of these sensitivity modes you choose depends on how thorough you want the scan to be. The Quick scan just scans files that end in certain extensions (.exe, .scr, .com, .doc, and so on), because these are the file types that are most often virus carriers. The Standard scan targets more files, ignoring extensions, but still limiting the scan to those file types that are usually associated with viruses. Finally, the Thorough scan scans everything and searches for every type of virus.

Once you have made your selections, click the **Start scan** button, and avast! will start doing exactly that—showing its progress within the same window, no less. When the scan is complete, a small window will pop up telling you, hopefully, that no viruses were found. (When was the last time you got a message like that on your Windows machine?)

A

UBUNTU DESKTOP CDS
FOR AMD64 USERS

As I mentioned in Chapter 1, the version of the Ubuntu Desktop CD that comes with this book is designed to work with i386 processors, either in PCs or Intel-based Macs. It will also work with AMD64 processors, albeit not in 64-bit mode. In order to use Ubuntu with AMD64 processors in 64-bit mode, you must get a different disk on your own. There are several ways of doing this: downloading an ISO (disk image) and then burning it to CD yourself, ordering the CD from Ubuntu (for free), or ordering it from an online Linux CD provider (for a nominal cost).

Downloading and Burning Ubuntu Desktop CD ISOs to CD

To download an ISO of the Ubuntu Desktop CD, go to the Ubuntu website at www.ubuntu.com, find the link to the download page, and then select and download the appropriate version for your machine. Remember that the ISO file you will be downloading is a heavyweight, weighing in at just over 700MB,

so the download will take a bit of time. Don't count on getting it all down-loaded and done before dinner . . . or, if you happen to be using a dial-up Internet connection, before dinner tomorrow. Yikes!

Burning the ISO to CD in Windows

Once the Desktop CD ISO has been downloaded, you need to burn it to CD before you can use it. Although Windows has built-in CD-writing capabilities, it does not have the ability to burn ISOs. To burn an ISO to CD in Windows, therefore, you must use a third-party commercial application, such as Nero. If you don't have a commercial disk-burning utility installed on your system, try the free and handy ISO Recorder.

To get ISO Recorder, visit to http://isorecorder.alexfeinman.com/v1.htm. Once the download is complete, double-click the **ISORecorderSetup.msi** file on your hard disk to install it.

After the installation is complete, burn your ISO to CD by double-clicking the Ubuntu ISO file on your disk and selecting **Copy image to CD** in the popup menu. A CD Recording Wizard window, like that in Figure A-1, should appear.
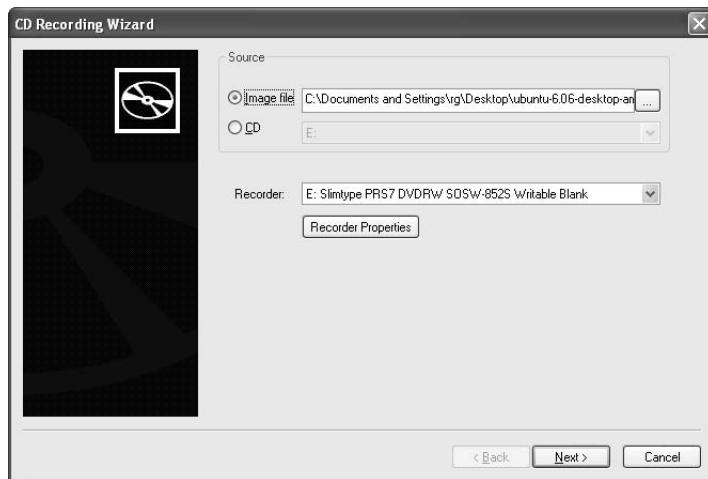
Figure A-1: Burning an ISO to CD in Windows using ISO Recorder

It is generally best if you burn installation or live CDs at a lower speed than the maximum speed allowed by your drive, in order to reduce the chance of error (with 2X to 4X speeds considered optimal). To do so, click the **Recorder Properties** button in the CD Recording Wizard window, and then drag the slider in the Properties window down to about 4X. Next, pop a blank CD into the drive and click the **Next** button. The CD burning process should begin. Once it's done, the CD should pop out of the drive, and if all goes well you'll have yourself an AMD64-compatible live CD. You can then use it by following the directions found at the beginning of Chapter 2 for using the live CD that comes with this book.

If your CD does not seem to work, there could be a problem with the ISO file you down-loaded. Find out by doing an integrity check as explained at https://help.ubuntu.com/community/HowToMD5SUM.

Burning the ISO to CD in OS X

Although Ubuntu no longer comes in PowerPC editions, the i386 editions can be installed and run on Intel-based Macs. You can also, of course, download ISOs for other architectures on your Mac, and then burn them to CD for use on other machines.

To burn an ISO file to CD in OS X, first check to make sure the ISO image is not mounted by opening a Finder window and checking the disk area at the top of the left pane. If it is mounted, a white drive icon will appear in that location. If the drive icon is there, click the arrow next to that entry to eject, or *unmount*, it.

After that, click **Applications** in the same Finder window, and then look for and open the **Utilities** folder. In that folder, find and then double-click **Disk Utility**. If the ISO is not listed in the left pane of the Disk Utility window when it opens, go back to the Finder window, locate the Ubuntu Live CD ISO you just downloaded, and then drag it to the left pane of the Disk Utility window, just below the listings for your current drives. Once the ISO file appears in that list, click it once to highlight it. Your Disk Utility window should then look something like that in Figure A-2.
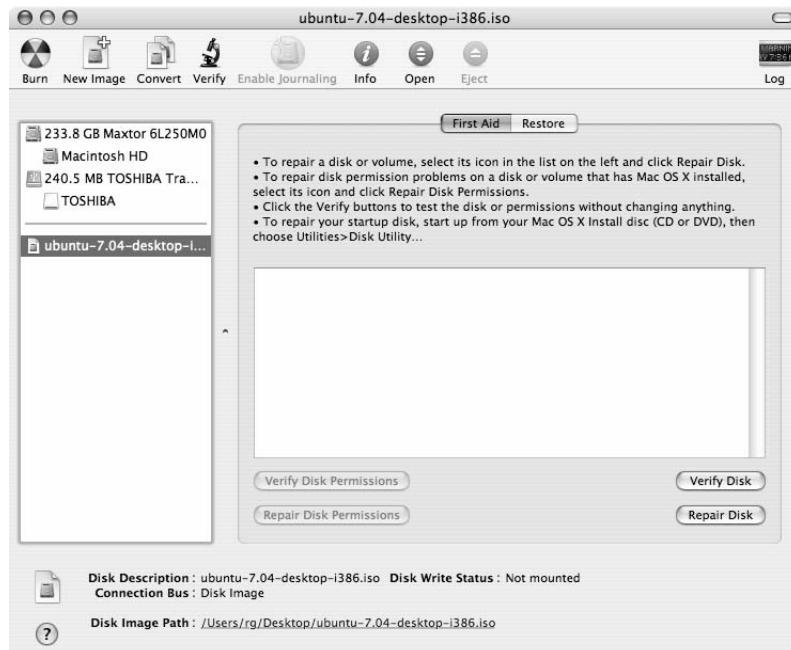
Figure A-2: Burning the Ubuntu Desktop CD ISO to CD in Mac OS X

To complete the process, click the **Burn** icon in the Disk Utility window's toolbar, and then insert a blank CD in your drive when prompted to do so. Once the blank disk is inserted and recognized, you will be able to adjust the burn speed from the drop-down menu next to the word *Speed*. Select as low a speed as your hardware will allow, which, depending on the age of your Mac, will probably be 4X to 8X. Finally, click the **Burn** button in that same window, and the burning process will begin.

Ordering an Install Disk from Ubuntu

The easiest and most foolproof way to get an Ubuntu Desktop CD is to simply order one (or more) for free from Ubuntu; you don't even have to pay shipping or handling. Ubuntu will not only send you one install CD for your particular machine architecture, but they will actually send you one for each of the architectures they support: i386 and AMD64. In fact, you don't even get to choose which you want; it's basically an all-or-nothing deal, which in this case isn't such a bad thing. Of course, the only downside to this approach is time. It can take four to six weeks for you to get the CDs in this manner, so if you're impatient, you might want to opt for one of the other methods.

To order your install CDs from Ubuntu, go to https://shipit.ubuntu.com and follow the directions there. It's easy.

Ordering an Install Disk from Other Online Sources

If you are in a hurry to get your install CD, you can also order a copy from one of the many online sources that specialize in copying and selling install CDs for a variety of Linux distributions at very low prices. For example, you can get an Ubuntu install CD from LinuxCD (www.linuxcd.org) for a quarter shy of two dollars. CheapBytes (www.cheapbytes.com) and UseLinux (www.uselinux.net) are two other well-known sources you might want to try.

B

RESOURCES

As Linux owes much of its growth and development to the Internet, it should come as no surprise that there is a wealth of information about Ubuntu available to you online. In addition to the usual news, how-to, and download sites, you will also find a variety of tutorials, forums, blogs, and other sources of useful information—all of which you can turn to as you use and learn more about your system.

Forums

When you are looking for advice, trying to solve a particular problem, or just looking for some general tips, online forums are the way to go. Fortunately, Ubuntu has a forum all its own, and since Ubuntu is primarily a desktop-oriented Linux distro, you are likely to find many fellow newbies and newbie-friendly posters there, rather than the hard-core geekiness you might find on

some other sites. There are, of course, other newbie-friendly forums, which, although not Ubuntu-specific, should also be able to provide you with lots of helpful information.

Regardless of which forum you are posting in, just be sure to mention that you are using Ubuntu, which version you have (Feisty Fawn, in case you forgot), and that you are new to Linux. And remember to always seek clarification when you get an answer you don't understand. The same poster will usually come back and clarify things for you. You should feel right at home at most of these sites, though you will probably come to like one or two more than the others.

http://www.ubuntuforums.org   The official Ubuntu community forum. Always a good place to start when you're in a fix, have a question, or just want to find out what's going on.

http://www.ubuntux.org/forum   Another slightly smaller Ubuntu-specific forum. A good place to turn if you find UbuntuForums.org a bit too much to wade through.

http://www.kubuntuforums.net   A forum dedicated to Kubuntu, an official Ubuntu edition based on the KDE desktop environment.

http://www.justlinux.com   If you can't find what you want in the previously mentioned forums, you can try this or the following two forums dealing with general Linux issues. You are sure to find many Ubuntu users on any of them.

http://www.linuxforum.com

http://www.linuxquestions.org

Linux Reference

These are sites, many of which are geared towards newbies, where you can learn more about using Ubuntu or Linux in general.

http://ubuntuguide.org/wiki/ubuntu:feisty   An unofficial Ubuntu startup guide.

https://help.ubuntu.com   Official documentation for the current release of Ubuntu.

http://www.tuxfiles.org   Lots of tutorials and information for Linux newbies.

http://www.linuxcommand.org   Learn to use commands in Linux.

http://www.linux.org   News, book reviews, downloads, and all sorts of other stuff—all about Linux!

Blogs

A lot of great information can also be found in blogs. In these, you can discover the findings of fellow users as they try new things, share tips, and offer solutions to problems.

http://ubuntu.wordpress.com   Ubuntu news with some tips thrown in.

http://www.ubuntux.org/blog   Mostly an Ubuntu-oriented news blog.

http://www.ubuntugeek.com   Despite having the word *geek* in the name, this is a pretty accessible set of mostly useful Ubuntu system and software tweaks.

Hardware Compatibility Issues

If you want to find out whether or not your hardware is compatible with Linux, or if you want to read up on other matters related to hardware support, take a look at the following sites:

https://wiki.ubuntu.com/HardwareSupport   Ubuntu-specific hardware compatibility information.

http://www.linuxcompatible.org/compatibility.html   Numerous compatibility lists, arranged by distribution.

http://www.linuxprinting.org   A great spot for reading up on printer compatibility issues.

http://www.linmodems.org   Find out if your modem is supported.

http://www.sane-project.org   Check to see if your scanner is Linux compatible, and look for fixes if it isn't.

http://www.linux-laptop.net   Have laptop, want Linux? Check it out here.

http://www.tuxmobil.org   Info for using Linux with anything that isn't stuck to your desk (laptops, PDAs, phones, etc.).

Wireless Connections

If you use a wireless card to connect to the Internet and have trouble getting your card to work, or if you just want to know where all the free wireless hotspots happen to be, the following sites should help.

http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux

http://www.ezgoal.com/hotspots/wireless

Free Downloads

If you find yourself looking for more goodies to play around with, you should be able to find plenty of free stuff to download at one of these sites.

Applications and Other Packages

http://sourceforge.net

http://www.freshmeat.net

http://www.gnomefiles.org

Free Fonts

http://www.fontfreak.com

http://www.fontparadise.com

http://fonts.tom7.com

News and Information

These sites are mainly informational, keeping you abreast of what's going on in the Linux world. DistroWatch focuses on the various distributions available out there, whereas Linux Today and LinuxPlanet fit better in the online magazine/newspaper genre.

http://www.distrowatch.com

http://www.linuxtoday.com

http://www.linuxplanet.com

Magazines

If you are more of a tactile type who enjoys the feel of paper pressed between your fingers, then you might like to turn to some of the Linux magazines available at most major newsstands. All have a good deal of online content, so even if you're not interested in the pleasures of holding a magazine in your hand, their sites are worth checking out.

http://www.tuxmagazine.com

http://www.linux-magazine.com (European)

http://www.linuxmagazine.com (US)

http://www.linuxjournal.com

There are two (unrelated) magazines sharing almost the same name: One of these is from the United States, one from Europe. The newbie who wants some pizzazz in his or her reading materials, plus some useful tips, and some things to play around with should go for the European version. The US version is targeted toward business users and power geeks, not newbies.

Books

Once you've finished working through this book, you should be able to do just about whatever you want in Ubuntu. Still, your interest may have been piqued enough that you would like to find out a bit more about Linux. Here are some books that might help in that quest.

How Linux Works by Brian Ward. (No Starch Press, 2004.)

The Debian System by Martin Krafft. (No Starch Press, 2005.)

The Linux Cookbook 2nd Ed. by Michael Stutz. (No Starch Press, 2004.)

Running Linux by Matthias Dalheimer and Matt Welsh. (O'Reilly Media, 2005.)

Linux Multimedia Hacks: Tips & Tools for Taming Images, Audio, and Video by Kyle Rankin. (O'Reilly Media, 2005.)

Linux Pocket Guide by Daniel J. Barrett. (O'Reilly Media, 2004.)

Ubuntu CDs

To order an AMD64 version of the Ubuntu Desktop CD, get a replacement for the i386 version that comes with this book, or get the next version of Ubuntu when it comes out (if you don't want to or can't download it), just place an order with any of the following sites. Remember that those from Ubuntu will be free, while those from other suppliers will cost you a little ($5 to $10 or so) but will be delivered much faster.

https://shipit.ubuntu.com

http://www.cheapbytes.com

http://www.linuxcd.org

INDEX

COLOPHON

UPDATES

Visit **http://www.nostarch.com/ubuntu_2e.htm** for updates, errata, and other information.

ABOUT THE CD

This software is released for free public use under several open source licenses. It is provided without any warranty, without even the implied warranty of merchantability or fitness for a particular purpose. See the license text included with each program for details. Source code for Ubuntu can be downloaded from http://archive.ubuntu.com or can be ordered from Canonical at the cost of the media and shipping. Ubuntu and Canonical are trademarks of Canonical Ltd. All other trademarks are the property of their respective owners.

Ubuntu's official documentation is online (https://help.ubuntu.com), and free technical support is available through web forums (www.ubuntuforums.org), mailing lists (www.ubuntu.com/community/lists), and IRC (#ubuntu). Commercial support is also available from Canonical (www.ubuntu.com/support/paid).

Ubuntu is sponsored by Canonical Ltd. For more information, visit www.ubuntu.com and www.canonical.com. To request free Ubuntu CDs, visit https://shipit.ubuntu.com.

The Intel x86 Edition packaged with this book will run on Intel x86–based systems (including Intel Pentium and AMD Athlon). See the section "Version Compatibility" on page 3 for more information. This and all other versions of Ubuntu are available for download at www.ubuntu.com/download.

No Starch Press does not provide support for Ubuntu. However, if the Ubuntu CD packaged with this book is defective, broken, or missing, you may email info@nostarch.com for a replacement.

You are legally entitled and encouraged to copy, share, and redistribute this CD. Share the spirit of Ubuntu!