Chapter 1: Introducing Fedora Core

In This Chapter

- **Explaining what Fedora Core is**
- **Going over what Fedora Core includes**
- **Discovering what Fedora Core helps you manage**
- **Getting started**

I bet you've heard about Linux and you probably know about Red Hat Linux as well, but chances are Fedora Core is new to you. If you're wondering what exactly Fedora Core is and what it can help you do, this chapter is all about answering those questions. Here I provide a broad-brushstroke picture of Fedora Core and tell you how you can start using it right away.

By the way, this book covers Fedora Core for Intel 80x86 and Pentium processors (basically any PC that can run any flavor of Windows).

What Is Fedora Core?

Fedora Core is what used to be Red Hat Linux — the Linux distribution from Red Hat that used to be available free of charge. If you never knew Red Hat Linux, think of Fedora Core as another Linux distribution. That leaves you with the question: What is a Linux distribution anyway?

Trying to describe a Linux distribution is a bit like that story of six blind men trying to describe an elephant. You know the one — one blind man touches the elephant's side and says the elephant is like a wall, another checks out the tusk and concludes that an elephant is like a spear, and so on. Along those lines, a Linux distribution appears to be many different things, depending on what you experience. You can think of it as the graphical user interface or just a PC to run your e-mail program, but, at its heart, it's an operating system. The following sections explain what I mean by this statement.

Operating systems and Linux

You know that your PC is a bunch of *hardware* — things you can touch, like the system box, monitor, keyboard, and mouse. The system box contains the most important hardware of all — the *central processing unit* (CPU), the microchip that runs the *software* (any program that tells the computer how

to do your bidding) — which you actually *can't* touch. In a typical Pentium 4 PC, the Pentium 4 microprocessor is the CPU. Other important hardware in the system box includes the memory (RAM chips) and the hard drive — and one program has to run all this stuff and get it to play nice: the operating system.

The *operating system* is software that manages all the hardware and runs other software at your command. You, the user, provide those commands by clicking menus and icons or by typing some cryptic text. Linux is an operating system — as are UNIX, Windows 98, Windows 2000, and Windows XP. The Linux operating system is modeled after UNIX; in its most basic, no-frills form, it also goes by the name *Linux kernel*.

The operating system is what gives a computer — any computer — its personality. For example, you can run Windows 98 or Windows XP on a PC — and on that same PC, you can *also* install and run Linux. That means, depending on which operating system is installed and running, *the same PC* can be a Windows 98, Windows XP, or Linux system.

The primary job of an operating system is to load software (computer programs) from the hard disk (or other permanent storage) into the memory and get the CPU to run those programs. Everything you do with your computer is possible because of the operating system — so if the operating system somehow messes up, the whole system freezes up. You know how infuriating it is when your favorite operating system — maybe even the one that came with your PC — suddenly calls it quits just as you are about to click the Send button after composing that long e-mail to your friend. You try the three-finger salute (pressing Ctrl+Alt+Del), but nothing happens. Then it's time for the Reset button (provided your computer's builders were wise enough to include one). Luckily, that sort of thing almost never happens with Linux — it has a reputation for being a very reliable operating system.

Does Linux really run on any computer?

Linux runs on many different types of computer systems — and it does seem able to run on nearly any type of computer. Linus Torvalds and other programmers originally developed Linux for the Intel 80x86 (and compatible) line of processors. Nowadays, Linux is also available for systems based on other processors — such as those with AMD's 64-bit AMD64 processors, the Motorola 68000 family; Alpha AXPs; Sun SPARCs and UltraSPARCs; Hewlett-Packard's HP PA-RISC; the PowerPC and PowerPC64 processors; and the MIPS R4x00 and R5x00. More recently, IBM has released its own version of Linux for its S/390 mainframe. This book covers Linux for Intel 80x86 and Pentium processors (these have in common a basic physical structure known as *IA-32 architecture*).

In technical mumbo jumbo, Linux is a *multiuser, multitasking operating system*. All this means is that Linux enables multiple users to log in, and Linux can run more than one program at the same time. Nearly all operating systems are multiuser and multitasking these days, but when Linux first started in 1994, *multiuser* and *multitasking* were big selling points.

Linux distributions

Fedora Core is a specific *Linux distribution* — essentially a package of features. Fedora Core consists of the Linux *kernel* (the operating system) and a collection of applications, together with an easy-to-use GUI installation program, called Anaconda.

You find many Linux distributions, and each includes the standard Linux operating system:

- **The X.Org X Window System:** The graphical user interface.
- **One or more graphical desktops:** Among the most popular are GNOME and KDE.
- **A selection of applications:** Linux programs come in the form of ready-to-run software, but the *source code* (the commands we humans use to tell the computer what to do, but you don't need the source code to run the operating system or any applications) is included, as is its documentation.

  Current Linux distributions include a huge selection of software — so much that it requires multiple CD-ROMs or a single DVD-ROM (which this book includes).

Many Linux distributions are commercial products that you can buy in computer stores and bookstores. If you have heard about Open Source and the GNU (*GNU's Not UNIX*) license, you may assume that no one can sell Linux for profit. Luckily for companies that sell Linux distributions, the GNU license — also called the GNU General Public License (GPL) — does allow commercial, for-profit distribution, but requires that the software be distributed in source-code form, and stipulates that anyone may copy and distribute the software in source-code form to anyone else. Fedora Core is available free of charge under the GPL, which means that my publisher may include the Fedora Core DVD-ROM with this book and you may make as many copies of the DVD as you like.

Transitioning to the Fedora Project

In late September 2003, Red Hat announced the Fedora Project — an open-source project sponsored by Red Hat where the developer community can participate and continue to evolve what used to be the Red Hat Linux product. The new product goes by the name *Fedora Core* and the project is expected to have Fedora Core releases every four to six months. Red Hat

continues to participate in the Fedora Project and help prepare the Fedora Core releases, but everything is done with involvement of the open source community under a public release schedule. As you may expect, Fedora Core is available freely, just as Red Hat Linux used to be, and you can expect books such as this one to include Fedora Core on DVD or CDs.

Red Hat continues to sell its commercial Linux distribution — called Red Hat Enterprise Linux. Red Hat anticipates that new technologies and enhancements that first appear in Fedora Core will eventually find their way into Red Hat Enterprise Linux. In this way, the Fedora Project serves as an incubator and testing ground for future Linux development.

To find out more about the Fedora Project, visit `fedora.redhat.com`.

Making sense of version numbers

Both the Linux kernel and Fedora Core have their own version numbers, not to mention the many other software programs (such as GNOME and KDE) that come with Fedora Core. The version numbers for the Linux kernel and Fedora Core are unrelated, but each has particular significance.

Linux-kernel version numbers

After Linux kernel version 1.0 was released on March 14, 1994, the loosely knit Linux development community adopted a version-numbering scheme. Version numbers such as 1.*X.Y* and 2.*X.Y,* where *X* is an even number, are considered the stable versions. The last number, *Y,* is the patch level, which is incremented as problems are fixed. For example, 2.6.5 is a typical, stable version of the Linux kernel. Notice that these version numbers are in the form of three integers separated by periods — *Major.Minor.Patch* — where *Major* and *Minor* are numbers denoting the major and minor version numbers, and *Patch* is another number representing the patch level.

Version numbers of the form 2.*X.Y* with an odd *X* number are beta releases for developers only; they may be unstable, so you should not adopt such versions for day-to-day use. For example, when you look at version 2.5.75 of the Linux kernel, notice the *5* — that tells you it's a beta release. Developers add new features to these odd-numbered versions of Linux.

You can find out about the latest version of the Linux kernel online at `www.kernel.org`.

Fedora Core version numbers

The Fedora Core development community, led by Red Hat, assigns the Fedora Core version numbers, such as 1 or 2. They are of the form *X.Y,* where *X* is the major version and *Y* the minor version. Nowadays if the minor version number is zero, it's simply dropped — as in Fedora Core 1 and Fedora Core 2.

Unlike with the Linux-kernel version numbers, no special meaning is associated with odd and even minor versions. Each version of Fedora Core includes specific versions of the Linux kernel and other major components, such as GNOME, KDE, and various applications.

The Fedora Project releases new versions of Fedora Core on a regular basis — every six months or so. For example, Fedora Core 1 came out in November 2003 and Fedora Core 2 in May 2004. Typically, each new major version of Fedora Core provides significant new features.

Under the hood in Linux kernel 2.6

Linux kernel 2.6 includes many new features and improvements when compared to its predecessor — the 2.4 kernel. I highlight some of these improvements in this section. You may not notice many of these improvements because they work behind the scenes. All you see is a Linux system that simply works great!

Support for wider range of computer hardware

For starters, the 2.6 kernel has been redesigned to support computers spanning a wider range of hardware than before — from bare-bones embedded microcontrollers to larger-scale servers with multiple processors.

To support distinct hardware architectures of the same processor family (such as x86), Linux 2.6 uses the concept of a *subarchitecture,* which refers to the processor and the associated bus and other hardware that defines a unique type of computer. For example, most of today's PCs are based on what is called the PC/AT subarchitecture because these PCs are based on the original IBM PC/AT. The 2.6 kernel supports PC/AT machines as well as other x86 subarchitectures, such as the NEC Voyager and the PC-9800 machines. The bottom line is that the 2.6 kernel can run on many variations of the x86-based machines.

Linux 2.6 also supports advanced features of processors such as hyperthreading, which enables a single processor to act as multiple virtual processors at the hardware level.

Better scalability

The 2.6 kernel provides better scalability for Intel x86 hardware by supporting advanced features such as Intel's Physical Address Extension (PAE), which enables many newer 32-bit x86 systems to access up to 64GB of memory. Linux 2.6 also provides better handling of interrupts for multiprocessor systems through improved support for Intel's Input/Output (I/O) Advanced Programmable Interrupt Controller (APIC).

Internally, the 2.6 kernel raises many internal limits from number of users to the maximum number of open files. For example, the number of unique users and groups has been increased from 65,536 to over 4 billion. The maximum number of open files can now grow as needed. File systems can be as large as 16TB (that's about 16,000 gigabytes!).

Linux 2.6 also increases the limits on the major and minor device numbers, which used to be a maximum of 255 in earlier kernels. These device numbers translate to 255 device types and 255 devices of a single type. In kernel version 2.6, the major device numbers can be up to 4,095 and minor device numbers can be more than a million. The upshot is that Linux 2.6 can support many more device types and many devices of a single type.

Improved device handling

Linux 2.6 has a number of new features for handling devices — especially hot plug devices such as the ones that connect to USB and Firewire interfaces common in today's PCs. First, the kernel uses a new virtual file system called *sysfs* that is meant to hold information about the devices on the system. The sysfs file system mounts on `/sys` and it presents a hierarchical view of all the devices organized by device type, bus, and so on. Through sysfs, the 2.6 kernel makes available to other applications a lot of information about devices, including the name of a device, resources such as interrupts and I/O ports used by the device, the power status of the device, and so on.

By using the sysfs capabilities available in the 2.6 kernel, a separate device-handling program called `udev` can now dynamically add device files when a device is added to a system. The `udev` program is invoked by the `/sbin/hotplug` shell script that runs when any hot plug device such as a USB device is plugged into the computer. `udev` gives each device a name that stays the same every time that device connects to the system. In Fedora Core, `udev` is not configured by default to manage the device files in the `/dev` directory. Instead `udev` manages device files in the `/udev` directory. However, you can easily configure `udev` (through the `/etc/udev/udev.conf` file) to manage the device files in `/dev` directory.

Other device-handling improvements in Linux 2.6 include ensuring that device driver modules are not unloaded while still in use and standardizing the way in which device drivers make available information about devices they support. All device driver module filenames now use the `.ko` extension — for *kernel object* — instead of the generic `.o` extension commonly used for object files.

Linux 2.6 also has improved support for many devices such as USB 2.0 and wireless devices. As for storage devices, the Integrated Drive Electronics (IDE) — also called AT Attachment (ATA) — and Small Computer System Interface (SCSI) support was updated in Linux 2.6. For example, IDE

CD-recorders are now accessed through the IDE driver instead of a special SCSI-emulation driver that was used in earlier versions of the kernel. The 2.6 kernel also supports the new Serial ATA (SATA) interface that can support data transfer rates of 150MB per second.

Mandatory access control with Security Enhanced Linux (SELinux)

Linux kernel 2.6 includes the mandatory access control framework provided by Security Enhanced Linux (SELinux), which was developed by the National Security Agency (NSA), a U.S. government agency. SELinux is implemented as a Linux Security Module (LSM) — an extension of the Linux kernel that allows security mechanisms to be easily added to the kernel. You can find more about SELinux at the NSA's Web site, `www.nsa.gov/selinux/`.

Without SELinux, access control in Linux is based on the user and group ID that owns a process or a file. In this discretionary access control approach, the superuser (`root`) has absolute discretion to access and do anything on the system. In contrast to this approach, SELinux views the system in terms of *subjects* (users or processes) and *objects* (files, devices, any system resources). Subjects can take on different *roles* such as normal user or system administrator. Each subject also has a *domain* and each object has a *type*. SELinux provides fine-grained control over who can access what in a Linux system by defining what domains can access what types and how one domain can transition into another when programs execute.

The mandatory access control rules are defined in the SELinux *security policy*. To support the fine-grained access control, all files need additional attributes called *contexts* that are stored in labels added to the files. Think of the contexts as information on which roles can access and do what with the file. When SELinux is enabled, all files in the file system have to be labeled with the security contexts. Only then can SELinux manage the fine-grained access control.

When you install Fedora Core, you can type **selinux** at the installer's `boot:` prompt to install SELinux and select the level of access control you want SELinux to enforce. This option appears in the screen where you configure the firewall.

Note: SELinux can be very helpful in securing your organization's external Web and e-mail servers that are exposed to the Internet and, therefore, subject to attacks. With a well-designed security policy, SELinux can make such Internet-facing servers resistant to damage from attacks, even if an attacker manages to gain superuser privileges. However, the additional effort involved in setting up and running SELinux may not be worthwhile for internal servers not directly connected to the Internet.

What Fedora Core Includes

Fedora Core comes with the Linux kernel and a whole lot more software. These software packages include everything from the graphical desktops to Internet servers and programming tools to create new software. In this section, I briefly describe some major software packages that come bundled with Fedora Core. Without this bundled software, Fedora Core wouldn't be as popular as it is today.

GNU software

I start with a collection of software that came from the GNU Project. You get to know these GNU utilities only if you use your Fedora Core system through a text terminal (or a graphical window that mimics one) — a basic *command-line interface* that puts nothing much on-screen but a prompt at which you type in your commands. The GNU software is one of the basic parts of Fedora Core.

As a Fedora Core user, you may not realize the extent to which Fedora Core (and, for that matter, all Linux distributions) relies on GNU software. Nearly all tasks you perform in a Fedora Core system involve one or more GNU software packages. For example, the GNOME graphical user interface (GUI) and the command interpreter (that is, the Bash shell) are both GNU software programs. By the way, the *shell* is the command-interpreter application that accepts the commands you type and runs programs in response to those commands. If you rebuild the kernel or develop software, you do so with the GNU C and C++ compiler (which is part of the GNU software that accompanies Fedora Core). If you edit text files with the `ed` or emacs editor, you're again using a GNU software package. The list goes on and on.

What is the GNU Project?

GNU is a recursive acronym that stands for *GNU's Not UNIX*. The GNU Project was launched in 1984 by Richard Stallman to develop a complete UNIX-like operating system. The GNU Project developed nearly everything needed for a complete operating system except for the operating system kernel. All GNU software was distributed under the GNU General Public License (GPL). GPL essentially requires that the software is distributed in source-code form and stipulates that any user may copy, modify, and distribute the software to anyone else in source-code form. Users may, however, have to pay for their individual copies of GNU software.

The Free Software Foundation (FSF) is a tax-exempt charity that raises funds for work on the GNU Project. To find out more about the GNU Project, visit its home page at `www.gnu.org`. You can find information about how to contact the Free Software Foundation and how to help the GNU Project.

Table 1-1 lists some of the well-known GNU software packages that come with Fedora Core. I show this table only to give you a feel for all the different kinds of things you can do with GNU software. Depending on your interests, you may never need to use many of these packages, but knowing they are there in case you ever need them is good.

Table 1-1 Well-Known GNU Software Packages

Software Package	*Description*
Autoconf	Generates shell scripts that automatically configure source-code packages
Automake	Generates `Makefile.in` files for use with Autoconf
Bash	The default shell — command interpreter — in Fedora Core
Bc	An interactive calculator with arbitrary precision numbers
Binutils	A package that includes several utilities for working with binary files: `ar`, `as`, `gasp`, `gprof`, `ld`, `nm`, `objcopy`, `objdump`, `ranlib`, `readelf`, `size`, `strings`, and `strip`
Coreutils	A package that combines three individual packages called Fileutils, Shellutils, and Textutils and implements utilities such as `chgrp`, `chmod`, `chown`, `cp`, `dd`, `df`, `dir`, `dircolors`, `du`, `install`, `ln`, `ls`, `mkdir`, `mkfifo`, `mknod`, `mv`, `rm`, `rmdir`, `sync`, `touch`, `vdir`, `basename`, `chroot`, `date`, `dirname`, `echo`, `env`, `expr`, `factor`, `false`, `groups`, `hostname`, `id`, `logname`, `nice`, `nohup`, `pathchk`, `printenv`, `printf`, `pwd`, `seq`, `sleep`, `stty`, `su`, `tee`, `test`, `true`, `tty`, `uname`, `uptime`, `users`, `who`, `whoami`, `yes`, `cut`, `join`, `nl`, `split`, `tail`, `wc`, and so on
Gnuchess	A chess-playing program
GNU C Library	For use with all Linux programs
Cpio	Copies file archives to and from a disk or to another part of the file system
Diff	Compares files, showing line-by-line changes in several different formats
`ed`	A line-oriented text editor
emacs	An extensible, customizable full-screen text editor and computing environment
Findutils	A package that includes the `find`, `locate`, and `xargs` utilities
Finger	A utility program designed to enable users on the Internet to get information about one another
Gawk	The GNU Project's implementation of the AWK programming language
GCC	Compilers for C, C++, Objective C, and other languages

(continued)

Table 1-1 ***(continued)***

Software Package	*Description*
Gdb	Source-level debugger for C, C++, and Fortran
Gdbm	A replacement for the traditional `dbm` and `ndbm` database libraries
Gettext	A set of utilities that enables software maintainers to internationalize (that means make the software work with different languages such as English, French, Spanish, and so on) a software package's user messages
Ghostscript	An interpreter for the Postscript and Portable Document Format (PDF) languages
Ghostview	An X Window System application that makes Ghostscript accessible from the GUI, enabling users to view Postscript or PDF files in a window
The GIMP	The GNU Image Manipulation Program is an Adobe Photoshop-like image-processing program
GNOME	Provides a graphical user interface (GUI) for a wide variety of tasks that a Linux user may perform
Gnumeric	A graphical spreadsheet (similar to Microsoft Excel) that works in GNOME
`grep` package	Includes the `grep`, `egrep`, and `fgrep` commands that are used to find lines that match a specified text pattern
Groff	A document-formatting system similar to `troff`
GTK+	A GUI toolkit for the X Window System (used to develop GNOME applications)
Gzip	A GNU utility for compressing and decompressing files
Indent	Formats C source code by indenting it in one of several different styles
Less	A page-by-page display program similar to `more`, but with additional capabilities
Libpng	A library for image files in the Portable Network Graphics (PNG) format
m4	An implementation of the traditional UNIX macro processor
Make	A utility that determines which files of a large software package need to be recompiled, and issues the commands to recompile them
Mtools	A set of programs that enables users to read, write, and manipulate files on a DOS file system (typically a floppy disk)
Ncurses	A package for displaying and updating text on text-only terminals
Patch	A GNU version of Larry Wall's program to take the output of `diff` and apply those differences to an original file to generate the modified version

Software Package	Description
RCS	The Revision Control System is used for version control and management of source files in software projects
Sed	A stream-oriented version of the `ed` text editor
Sharutils	A package that includes `shar` (used to make shell archives out of many files) and `unshar` (to unpack these shell archives)
Tar	A tape archiving program that includes multivolume support; the capability to archive sparse files, handle compression and decompression, and create remote archives; and other special features for incremental and full backups
Texinfo	A set of utilities that generates printed manuals, plain ASCII text, and online hypertext documentation (called Info), and enables users to view and read online Info documents
Time	A utility that reports the user, system, and actual time that a process uses

GUIs and applications

Face it — typing cryptic Linux commands on a terminal is boring. For average users like us, using the system through a *graphical user interface* (*GUI,* pronounced "gooey") that gives us pictures to click and windows (small *w*) to open is much easier. This is where the X Window System, or X, comes to the rescue.

X is kind of like Microsoft Windows, but the underlying details of how X works is completely different from Windows. Unlike Windows, X provides the basic features of displaying windows on-screen, but it does not come with any specific look or feel for graphical applications. That look and feel comes from GUIs, such as GNOME and KDE, which make use of the X Window System.

Fedora Core comes with the X Window System in the form of X.Org X11 — an implementation of X Window System for 80x86 systems. X.Org X11 works with a wide variety of video cards available for today's PCs.

Until recently XFree86 from the XFree86 Project (`www.xfree86.org`) was the commonly used X Window System implementation for x86 systems. However, around version 4.4, some changes to the XFree86 licensing terms caused concerns to many Linux and UNIX vendors — they felt that the licensing terms were no longer compatible with the GNU General Public License (GPL). In January 2004, several vendors formed the X.Org Foundation (`www.x.org`) to promote continued development of an open source X Window System and graphical desktop. The first release of X.Org X11 uses the same code as that used by XFree86 4.4, up until the time when the XFree86 license changes precipitated the creation of X.Org Foundation.

As for the GUI, Fedora Core includes two powerful GUI desktops: KDE (K Desktop Environment) and GNOME (GNU Object Model Environment). If both GNOME and KDE are installed on a PC, you can choose which desktop you want as the default — or switch between the two. KDE and GNOME provide desktops similar to those of Microsoft Windows and the Macintosh OS. GNOME also comes with the Nautilus graphical shell that makes finding files, running applications, and configuring your Fedora Core system easy. With GNOME or KDE, you can begin using your Fedora Core workstation without having to know cryptic Linux commands. However, if you ever need to use those commands directly, all you have to do is open a terminal window and type them at the prompt.

Fedora Core also comes with many graphical applications. The most noteworthy program is the GIMP (GNU Image Manipulation Program), a program for working with photos and other images. The GIMP's capabilities are on a par with Adobe Photoshop.

Providing common productivity software — such as word-processing, spreadsheet, and database applications — is an area in which Linux used to be lacking. This situation has changed, though. Fedora Core comes with the OpenOffice.org office-productivity applications. In addition, you may want to check out these prominent, commercially available office-productivity applications for Linux that are not included on the companion DVD-ROMs:

- **Applixware Office:** Now called Anyware Desktop for Linux, this office package is a good example of productivity software for Linux. You can find it at `www.vistasource.com`.
- **StarOffice:** From Sun Microsystems (`www.sun.com/staroffice`), StarOffice is another well-known productivity software package.
- **CrossOver Office:** From CodeWeavers (`www.codeweavers.com/products/office/`), you can install your Microsoft Office applications (Office 97, Office 2000, and Office XP) in Linux. Note that only the Office 2000 versions of two Office applications — Microsoft Access and Outlook — are supported.

As you can see, there's no shortage of Microsoft Office-compatible office applications for Linux.

Networks

Fedora Core comes with everything needed to use the system in networks so that the system can exchange data with other systems. On networks, computers that exchange data have to follow well-defined rules or protocols. A *network protocol* is a method that the sender and receiver agree upon for exchanging data across a network. Such a protocol is similar to the rules you might follow when you're having a polite conversation with someone at a party. You typically start by saying hello, exchanging names, and then

taking turns talking. That's about the same way network protocols work. The two computers use the protocol to send bits and bytes back and forth across the network.

One of the most well-known and popular network protocols is Transmission Control Protocol/Internet Protocol (*TCP/IP*). TCP/IP is the protocol of choice on the Internet — the "network of networks" that now spans the globe. Fedora Core supports the TCP/IP protocol and any network applications that make use of TCP/IP.

Internet servers

Some popular network applications are specifically designed to deliver information from one system to another. When you send electronic mail (e-mail) or visit Web sites using a Web browser, you use these network applications (also called Internet services). Here are some common Internet services:

- Electronic mail (e-mail) that you use to send messages to any other person on the Internet using addresses like `joe@someplace.com`.
- World Wide Web (or simply, Web) that you browse using a Web browser.
- News services, where you can read newsgroups and post news items to newsgroups with names such as `comp.os.linux.networking` or `comp.os.linux.setup`.
- File-transfer utilities that you can use to download files.
- Remote login that you can use to connect to and work with another computer (the remote computer) on the Internet — assuming you have the required username and password to access that remote computer.

Any Fedora Core PC can offer these Internet services. To do so, the PC must be connected to the Internet and it must run special server software called *Internet servers*. Each of the servers uses a specific protocol for transferring information. For example, here are some common Internet servers that you find in Fedora Core:

- `sendmail` is the mail server for exchanging e-mail messages between systems using SMTP *(Simple Mail Transfer Protocol)*.
- Apache `httpd` is the Web server for sending documents from one system to another using HTTP *(Hypertext Transfer Protocol)*.
- `vsftpd` is the server for transferring files between computers on the Internet using FTP *(File Transfer Protocol)*.
- `innd` is the news server for distribution of news articles in a store-and-forward fashion across the Internet using NNTP *(Network News Transfer Protocol)*.

- `in.telnetd` allows a user on one system to log in to another system on the Internet using the TELNET protocol.
- `sshd` allows a user on one system to securely log in to another system on the Internet using the SSH (Secure Shell) protocol.

Software development

Fedora Core is particularly well suited to software development. Straight out of the box, it's chock-full of software-development tools such as the compiler and libraries of code needed to build programs. If you happen to know UNIX and the C programming language, you will feel right at home programming in Fedora Core.

As far as the development environment goes, Fedora Core has the same basic tools (such as an editor, a compiler, and a debugger) that you might use on other UNIX workstations, such as those from IBM, Sun Microsystems, and Hewlett-Packard (HP). What this means is that if you work by day on one of these UNIX workstations, you can use a Fedora Core PC in the evening at home to duplicate that development environment at a fraction of the cost. Then you can either complete work projects at home or devote your time to software you write for fun and to share on the Internet.

Just to give you a sense of Linux's software-development support, here's a list of various features that make Linux a productive software-development environment:

- GNU C compiler, `gcc`, can compile ANSI-standard C programs.
- GNU C++ compiler (`g++`) supports ANSI-standard C++ features.
- The GNU Java compiler (`gcj`) can compile programs written in the Java programming language.
- The GNU `make` utility enables you to compile and link large programs.
- The GNU debugger, `gdb`, enables you to step through your program to find problems and to determine where and how a program failed. (The failed program's memory image is saved in a file named core; `gdb` can examine this file.)
- The GNU profiling utility, `gprof`, enables you to determine the degree to which a piece of software uses your computer's processor time.
- Subversion, Concurrent Versions System (CVS), and Revision Control System (RCS) maintain version information and control access to the source files so two programmers don't modify the same source file inadvertently.

- The GNU emacs editor prepares source files and even launches a compile-link process to build the program.
- Perl is a scripting language you can use to write scripts to accomplish a specific task, tying together many smaller programs with Linux commands.
- The Tool Command Language and its graphical toolkit (`Tcl/Tk`) enable you to build graphical applications rapidly.
- Python is an interpreted programming language comparable to Perl and Tcl (the Fedora Core installation program, called `anaconda`, is written in Python).
- Dynamically linked shared libraries allow your actual program files to be much smaller because all the library code that several programs may need is shared — with only one copy loaded in the system's memory.

Online documentation

As you become more adept at using Fedora Core, you may want to look up information quickly — without having to turn the pages of (ahem) this great book, for example. Luckily, Fedora Core comes with enough online information to jog your memory in those situations when you vaguely recall a command's name, but can't remember the exact syntax of what you're supposed to type.

If you use Linux commands, you can view the manual page — commonly referred to as the *man page* — for a command by using the `man` command. (You do have to remember that command in order to access online help.)

You can also get help from the GUI desktops. Both GNOME and KDE desktops come with help viewers to view online help information. In GNOME, select Main Menu➪Help. You then see two broad categories of information:

- **GNOME - Desktop** is information on how to use the GNOME desktop and some GNOME applications.
- **Additional documents** include online documentation for the GNU software (primarily for the applications and the software development tools).

You can then browse the information by clicking the links on the initial help window. Figure 1-1 shows a typical help file in GNOME.

In KDE desktop, you can start KDE Help by choosing Main Menu➪Help. The KDE Help application looks similar to the GNOME help browser (see Figure 1-2).

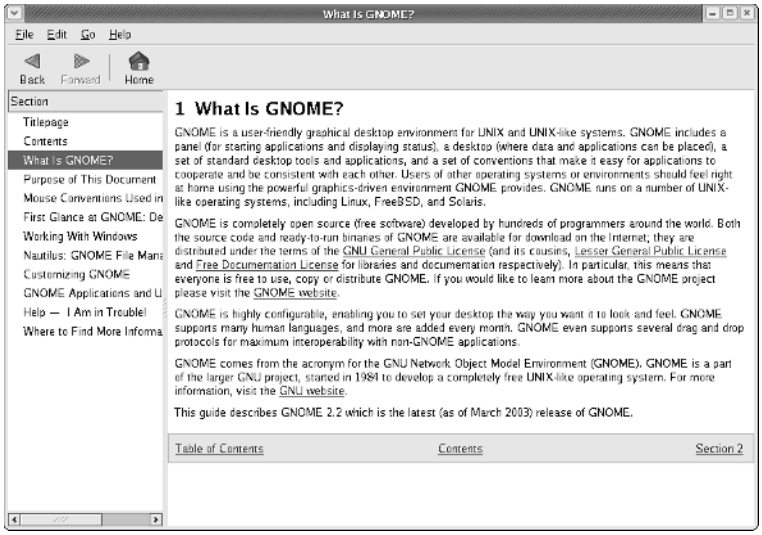

Figure 1-1: Here's a typical sample of online help in the GNOME desktop.

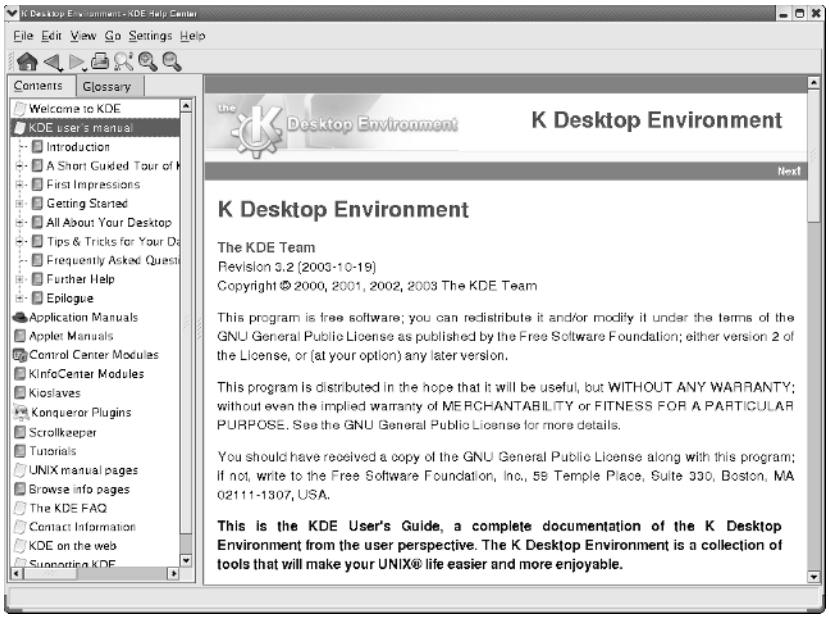

Figure 1-2: The KDE Help Center provides online help in the KDE desktop.

You can then click the links to view specific online help information. You can follow links to discover how to use KDE and obtain information about the KDE Project. Another set of links provides access to the man pages and GNU info pages, just as the GNOME help browser does.

What Fedora Core Helps You Manage

As an operating system, Fedora Core acts as the intermediary through which you, as the "lord of the system," manage all the hardware. The hardware includes the system box, the monitor, the keyboard, the mouse, and anything else connected to the system box. The catchall term *peripheral* refers to any equipment attached to the system. If you use a laptop computer, all your hardware is packaged into the laptop.

Inside that system box is the system's brain — the microprocessor (Intel Pentium 4, for example) or the central processing unit (CPU) — that performs the instructions contained in a computer program. When the microprocessor is running a computer program, that program's instructions are stored in the memory or RAM. RAM stands for *Random Access Memory* (that means any part of the memory can be accessed randomly — in arbitrary order).

The system box has another crucial component — the hard disk or hard drive, as it is sometimes called. The hard disk is the permanent storage space for computer programs and data. It's permanent in the sense that the contents don't disappear when you power off the PC. The hard disk is organized into files, which are in turn organized in a hierarchical fashion into directories and subdirectories (somewhat like organizing paper folders into the drawers in a file cabinet).

To keep a Fedora Core system running properly, you or someone else has to make sure the hardware is working properly and the files are backed up regularly. There is also the matter of security — making sure only legitimate people can access and use the system. These tasks are called *system administration*.

If you are using Fedora Core at a big facility with many computers, a full-time system administrator probably takes care of all system administration tasks. On the other hand, if you are running Fedora Core on a home PC, you are the system administrator. Don't let the thought frighten you. You don't have to know any magic incantations or prepare cryptic configuration files to be a system administrator. Fedora Core includes many graphical tools such as GNOME's Nautilus graphical shell that makes system administration a "point-and-click" job just like running any other application.

Disks, CD-ROMs, and DVD-ROMs

Fedora Core comes on a single DVD-ROM. After installation, the Linux kernel and all the applications are stored on your hard drive — which is where your PC looks first when you tell it to do something.

Typically, the hard drive is prepared to use Fedora Core during the installation process. After that, you usually leave the hard drive alone except to back up the data stored there or (occasionally) to install new applications.

Using CD-ROMs or DVD-ROMs in Fedora Core is easy. While you are logged in at the GNOME or KDE desktop, just pop in a CD or DVD in the drive and a DVD/CD-ROM icon appears on the desktop. You can then double-click that DVD/CD-ROM icon and the CD or DVD's contents appears in a window. This whole process of accessing the files on a CD or a DVD from Fedora Core is called *mounting the CD or the DVD*.

Besides the hard drive and DVD/CD-ROM drive, of course, your PC may have other drives, such as a floppy disk or Zip drive, and using those disks in Fedora Core is also simple: You insert a disk and double-click the icon that represents the disk drive on the GUI desktop. Doing so mounts the disk so you can begin using it.

Peripheral devices

Anything connected to your PC is a peripheral device, and so are some components like sound cards that are installed inside the system box. You can configure and manage these peripheral devices in Fedora Core.

One of the common peripherals is a printer, typically hooked up to the parallel port of your PC. (Fedora Core comes with a graphical Printer Configuration tool that you can use to configure the printer.)

Another peripheral device that needs configuration is the sound card. Unlike Windows, Fedora Core requires you to perform a configuration step before the sound works. Again, you can run a graphical sound-card detection utility to test the sound card (choose Main Menu➪System Settings➪Soundcard Detection from the graphical desktop).

Fedora Core configures other peripheral devices such as the mouse and keyboard at the time of installation. You can pretty much leave them alone after installation.

Nowadays PCs come with the USB (Universal Serial Bus) interface and many devices, such as printers and scanners, plug into a PC's USB port. One nice feature of USB devices is that you can plug them into the USB port and unplug them at any time — the device does not have to be connected when you power up the system. These devices are called *hot plug* because you can plug in a device when the system is hot, meaning while it's running. Fedora Core supports many hot plug USB devices. When you plug in a device into the USB port, Fedora Core loads the correct driver and makes the device available to applications.

File systems and sharing

The whole organization of directories and files is called the *file system*. You can, of course, manage the file system using Fedora Core. When you browse the files from the GNOME or KDE graphical desktop, you work with the familiar folder icons.

A key task in caring for a file system is to back up important files. In Fedora Core you can use the `tar` program to archive one or more directories on a floppy or a Zip drive. You can even back up files on a tape (if you have a tape drive). If you have a CD burner, you can also burn a CD with the files you want to back up or save for posterity.

Fedora Core can also share parts of the file system with other systems on a network. For example, you can use the Network File System (NFS) to share files with other systems on the network. To a user on the system, the remote system's files appear to be in a directory on the local system.

Fedora Core also comes with the Samba package, which supports file sharing with Microsoft Windows systems. Samba makes a Fedora Core system work just like a Windows file or print server.

Network

Now that most PCs are either in a local area network or connected to the Internet, you need to manage the network as well. Fedora Core comes with a Network Configuration tool to set up the local area network. For connecting to the Internet using a modem, you use the Internet Configuration Wizard (choose Main Menu➪System Tools➪Internet Configuration Wizard).

If you connect to the Internet using DSL (that's the fast Internet connection from the phone company) or cable modem, you need a PC with an Ethernet card that connects to the cable or DSL. It also means you have to set up a local area network and configure the Ethernet card. But fortunately, these steps are typically a part of Fedora Core installation. If you want to do the configurations later, you can — by using the Network Configuration tool (choose Main Menu➪System Settings➪Network).

Fedora Core also includes tools for configuring a *firewall,* a protective buffer that helps keep your system relatively secure from anyone trying to snoop over your Internet connection. You can configure the firewall by running the Firewall Configuration tool (choose Main Menu➪System Settings➪ Security Level from the graphical desktop).

How Do I Get Started?

Based on my personal experience in learning new subjects, I prescribe a four-step process to get started with Fedora Core:

1. Install Fedora Core on your PC.
2. Configure Fedora Core so that everything works to your liking.
3. Explore the GUI desktops and the applications.
4. Learn the details of specific subjects such as Internet servers.

In the following sections, I explain this prescription a bit more.

Install

Microsoft Windows comes installed on your new PC, but Fedora Core usually doesn't. So your first hurdle is to get Fedora Core onto your PC.

After you overcome that initial human fear of the unknown, I'll bet you find Fedora Core fairly easy to install — but where do you *get* it in the first place? Well, the good news is that it's free — available just for the downloading. For example, you can visit the Linux Online Web site at `www.linux.org` and click the Download button.

Because the complete distribution is HUGE — it takes up several CDs or a single DVD — your best bet is to buy a book (such as this one) that comes with Fedora Core on a DVD-ROM. You can then do the installation by following the instructions in the book.

Just to pique your curiosity, installation involves creating space on the hard disk for both Windows and Linux. Then a step creates the Linux partitions and installs Fedora Core from the DVD. Along the way, you configure many items from the Ethernet card (if any) to the X Window System.

Configure

When you finish installing Fedora Core, the next step is to configure individual system components (for example, the sound card and the printer) and tweak any needed settings that aren't configured during installation.

If you aren't getting a graphical login screen, for example, Fedora Core comes with tools that help you troubleshoot that problem (typically by configuring the X Window System).

You also want to configure your GUI desktop of choice — GNOME or KDE. Each has configuration tools. You can use these tools to adjust the look and feel of the desktop (background, title fonts, even the entire color scheme).

After you're through with the configuration step, all the hardware on your system and the applications run to your liking.

Explore

With a properly configured Fedora Core PC at your disposal, you are ready to explore Fedora Core itself. You can begin the exploration from the GUI desktop that you get after logging in.

Explore the GUI desktops — GNOME and KDE — and the folders and files that make up the Linux file system. You can also try out the applications from the desktop. You find office and multimedia applications and databases to explore.

Also try out the `shell` — open up a terminal window and type some Linux commands in that window. You can also explore the text editors that work in text mode. Knowing how to edit text files without the GUI just in case the GUI is not available is good. At least you won't be helpless.

Learn

After you explore the Fedora Core landscape and know what is what, you can then dig in deeper and learn specific subject areas. For example, you may be interested in setting up Internet servers. You can then learn the details of setting up individual servers such as sendmail for e-mail, Apache for a Web server, and the INN server for news.

You can choose to find out about many more areas, such as security, programming, and system administration.

Of course, you can expect this step to go on and on, even after you have your system running the way you want it — for now. After all, learning is a lifelong journey.

Bon voyage!