CHAPTER 4

Becoming an Ubuntu Power User

- Administering System and User Settings
- Understanding How Linux Stores and Organizes Files
- Learning Unity Keyboard Shortcuts
- Using the Terminal
- Working with Windows Programs
- Installing Software from PPAs
- Compiling Software from Source
- Summary

UBUNTU IS RELATIVELY STRAIGHTFORWARD to set up and use for the common day-to-day tasks. With time, though, most users want to learn how to be more efficient performing their tasks on that computer. It is this efficiency, this elegance in the way tasks are performed, that forms the basis of the separation between beginners and power users. Those who are content to learn just enough to get a task done need not read this chapter. Those who want to perform these tasks more easily, more quickly, or more smoothly and with a deeper understanding of what they are doing and why are the target audience here. This chapter is not about customization; for that see Chapter 6. This chapter is about understanding and learning how to use what you already have to its fullest potential.

Administering System and User Settings

To begin, search for Settings in the Dash and open System Settings (Figure 4-1).

From here you can adjust many aspects of your system, as shown in Figure 4-2.

Click an item to select it and modify its settings. When you are done, you may close the window or click All Settings in the upper left to return to the menu. Some settings require you to unlock them by clicking the Unlock button at the upper right and entering a password before you can modify their settings (Figure 4-3). You must have administrative privileges to do this.

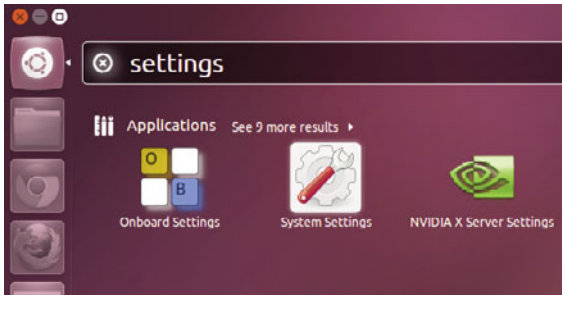

Figure 4-1 Click System Settings to begin.

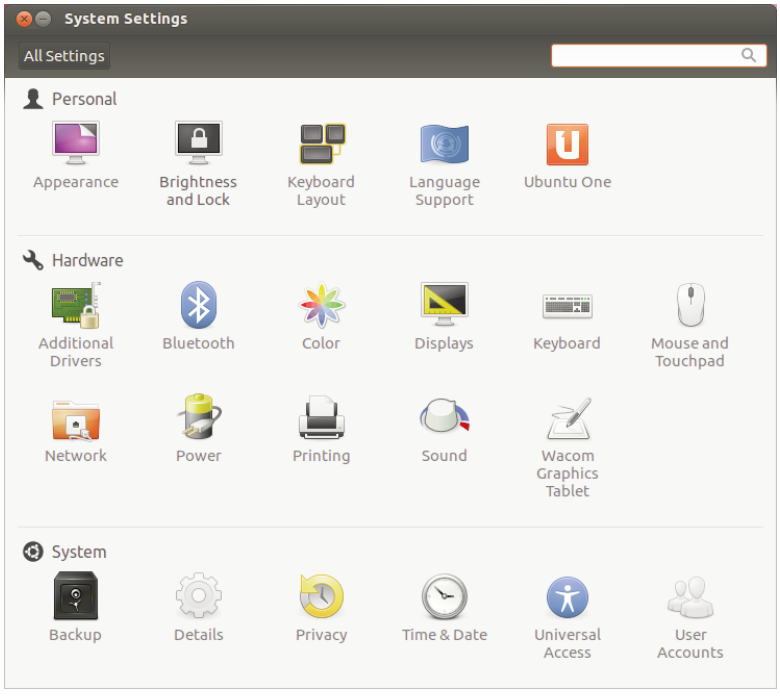

Figure 4-2 Choose from among many options in the System Settings window.

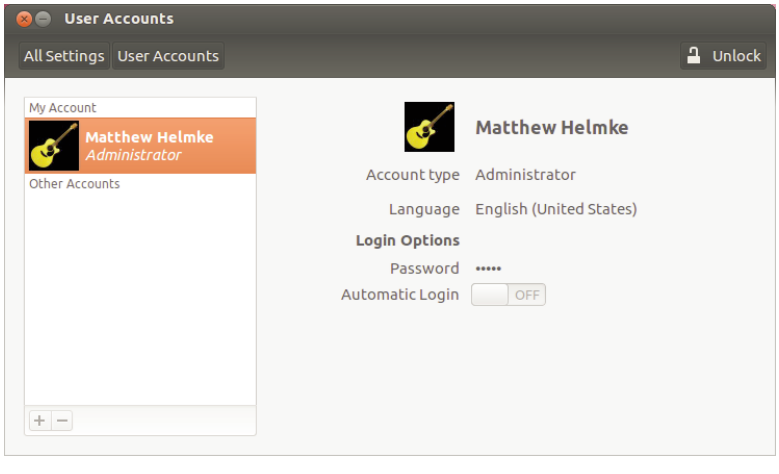

Figure 4-3 Click All Settings to return to the main menu or Unlock to continue.

User Settings

Let's begin with a look at available user settings. Once you unlock the window shown in Figure 4-3, you can click the + symbol at the bottom left to add a new user, click the – symbol to delete a user, or click on a user account to modify its settings. Click on any item shown at the right to modify its settings, as in Figure 4-4.

Privacy Settings

The privacy menu is new and allows you to carefully manage what information is recorded about you when you use Ubuntu (Figure 4-5). For example, when you open programs, Ubuntu records what you use most often and will list it first in the Dash as you search. Most users appreciate this feature. However, in some instances, this could be insecure or undesirable. You can manage history, files, applications, and other privacy features here. You can even turn the entire feature off.

Ubuntu One

Ubuntu One is very similar to the commonly known and used Dropbox, which allows you to store data in the cloud for use on multiple computers. This feature is built in to Ubuntu and does not require any additional software to be installed, but you do need to register so that your information can be kept secure (Figure 4-6).

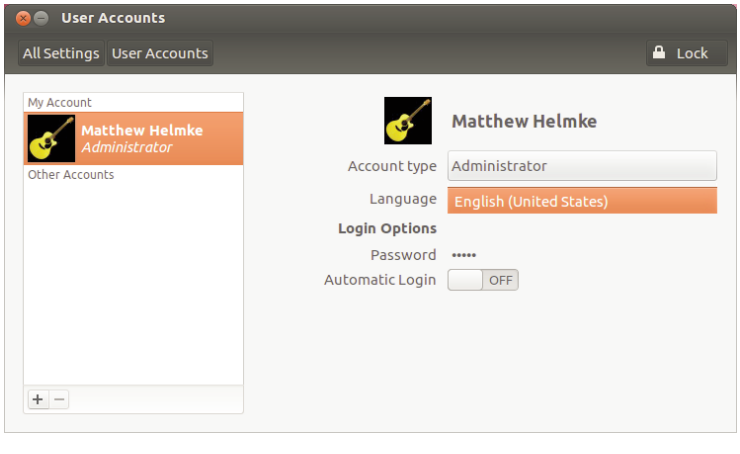

Figure 4-4 Items have either menus or their own pop-up windows.

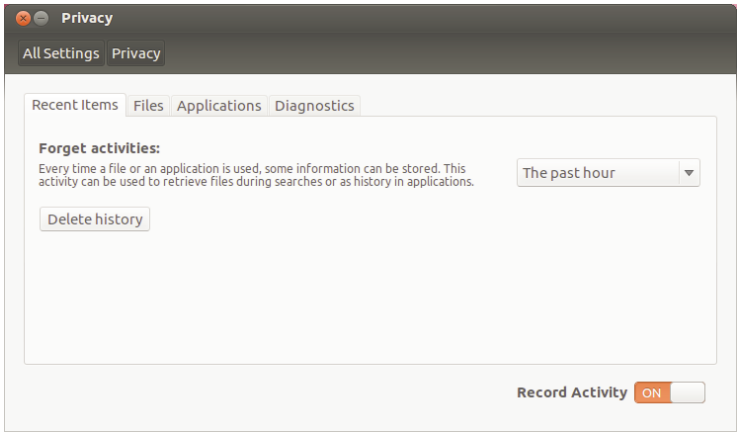

Figure 4-5 Want to delete information recorded only in the last hour? Easy.

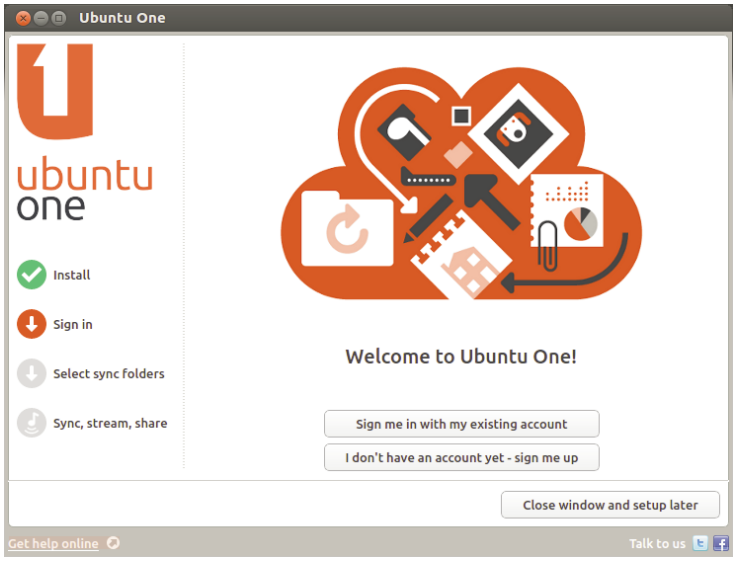

Figure 4-6 You need to log in or create an account to use Ubuntu One.

Once you have logged in, you can use Ubuntu One to store data, settings, and other information, as in Figure 4-7. It can automatically sync these settings across multiple computers, if all of them are running Ubuntu, and you can control or revoke access by any of these devices using this menu.

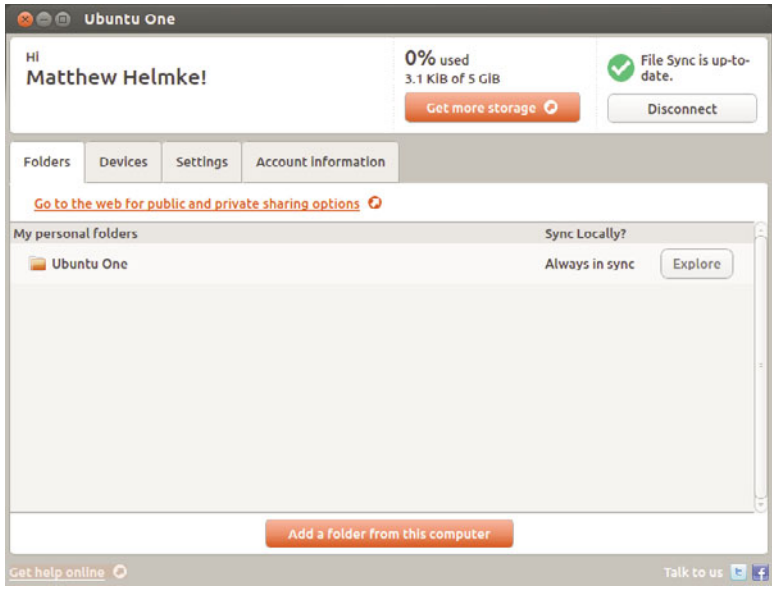

Figure 4-7 Ubuntu One has many menus to make setup and use easy.

In addition, any music you purchase from the Ubuntu Music Store using the music player will also be stored in Ubuntu One and accessible from a Web interface through your username and password using any computer and Web browser as well as by any device you permit to access, including Windows computers and Android phones. Access for more operating systems is in process and may be available by the time you read this. See https://one.ubuntu.com for more details.

Default Settings

Default settings generally make sense for most users. However, what if you want a different program to open when you try to play a certain media file? Click the Details icon in the System Settings menu to change from the default settings, as in Figure 4-8. Select different items from the menu at the left for other options.

TIP There are lots of settings available. Feel free to change any of them to suit your needs or preferences. You may want to take note of the current settings before you make changes, just in case you change your mind and want to go back to what you had.

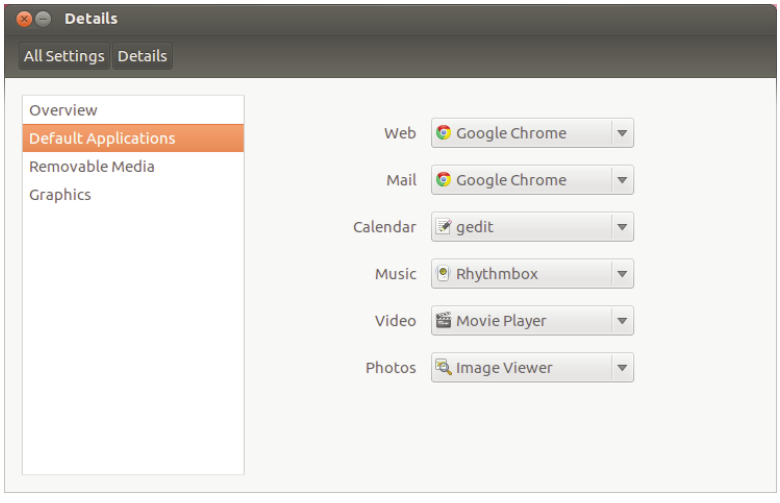

Figure 4-8 Changing your default applications is easy: just make sure you install what you want to use before you try to tell Ubuntu to use it.

Understanding How Linux Stores and Organizes Files

If you have not used Linux before, the way that Linux stores and organizes files is likely to be new to you because the layout is quite different from Windows and Mac OS X.

TIP **Folders and Directories**
When reading about file management, don't get confused by the terms *folders* and *directories—both* words describe the same thing.

In the Windows world, each disk drive is labeled with an identifying letter such as C: for your hard disk and D: for the CD/DVD drive. In the Linux world, however, everything is part of the same filesystem organization. As such, if you have two or three hard disks, a CD drive, and a USB stick all plugged in, they will all be part of the same folder structure.

The diagram shown in Figure 4-9 should give you an idea of how everything hangs together.

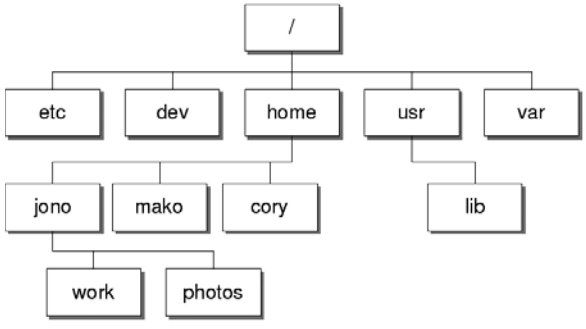

Figure 4-9 Linux filesystem organization

Right at the top of the tree is the root folder, referred to as /. Inside this folder are a number of special system folders, each with a specific use. As an example, the /home folder contains a number of home directories for each user on the system. As such, the user account has the home folder set to /home/matthew.

Which Folder Does What? The folder structure in a modern Linux distribution such as Ubuntu was largely inspired by the original UNIX foundations that were created by men with large beards. Although you don't really need to know what these folders do, since Ubuntu looks after the housekeeping for you, some of you may be interested in the more important folders. To help, we present the Linux folder hit list in Table 4-1.

Configuration Files In Table 4-1, /etc is described as storing systemwide configuration files for your computer. Aside from these files that affect everyone, there are also configuration files for each specific user. Earlier, when you customized Ubuntu's look and feel, the settings were applied only to your current user account. So where are those settings stored?

Inside your home directory are a number of folders that begin with a dot (.), such as .config and .libreoffice. These folders contain the configuration settings for user-specific applications. By default, these dot folders are hidden in the graphical file viewer because you rarely need to access them. You can view these hidden files and folders when you are viewing your home folder by clicking View > Show Hidden Files in the menu or by pressing Ctrl-H.

Table 4-1 Linux Folders

Folder	Use
/boot	This folder contains important files to boot the computer, including the bootloader configuration and the kernel.
/dev	Each device on your system (such as sound cards, Webcams, etc.) has an entry in this folder. Each application accesses the device by using the relevant items inside /dev.
/etc	Systemwide configuration files for the software installed on your system are stored here.
/home	Each user account on the system has a home directory that is stored here.
/lib	Important system software libraries are stored here. You should never need to delve into this world of the unknown.
/media	Media devices such as CD drives and USB sticks are referenced here when they are plugged in. More on this later.
/mnt	Other devices can be mounted too. Again, more on this later.
/opt	Optional software can be installed here. This folder is usually used when you want to build your own software. If you don't build your own software, you ignore this folder.
/proc/sys	Information about the current running status of the system is stored here.
/root	This is the home directory for the main superuser.
/bin	Software that is vital for the system to be able to boot is stored here.
/sbin	Software that should be run only by the superuser is stored here.
/usr	General software is installed here.
/var	This folder contains log files about the software on your computer.

Using Windows Files on Another Partition

For those of you who spend a considerable amount of your life on Windows partitions, you may want to be able to access these partitions from Ubuntu. This is no problem, although you will need to edit a special configuration file to do this. Luckily, you need to edit this file only once, and then everything will be set up.

Ubuntu should automatically recognize any Windows partitions you have on your computer and set them up for you; however, you may need to

modify them or add your own. Search for Disk Utility in the Dash and write down the partition numbers and filesystem for your Windows partitions. The partition number will look something like /dev/hdb1 or /dev/sdb1, and the filesystem will be either FAT, VFAT, or NTFS.

The next step is to create some mount points. When your Windows partitions are enabled, they are accessed via a particular folder in Ubuntu. This is called a mount point. As an example, if you have a mount point as /media/win1 and on your Windows partition you want to access your Work folder, you would access it from Ubuntu as /media/win1/work.

Mount points usually live in the /media folder. Create a different mount point for each Windows partition. As an example, if you have three Windows partitions, run the following commands:

```
foo@bar:~$ sudo mkdir /media/win1
foo@bar:~$ sudo mkdir /media/win2
foo@bar:~$ sudo mkdir /media/win3
```

Now open up the following configuration file:

```
foo@bar:~$ sudo gedit /etc/fstab
```

The /etc/fstab file maps partition numbers to mount points. At the bottom of the file, add a line like this for each mount point:

```
/dev/hdb1 media/win1 vfat users,rw,owner,umask=000 0 0
```

You will need to change the partition number (the first column), mount point (second column), and filesystem (third column) for your relevant partitions.

Now reload /etc/fstab to enable the partitions:

```
foo@bar:~$ sudo mount -a
```

Some hard disk icons for the new partitions now appear.

TIP You can find more about fstab at https://help.ubuntu.com/community/Fstab.

Learning Unity Keyboard Shortcuts

Using a mouse is intuitive because of the direct interaction between your hand, your eyes, and the cursor, but for repeated actions, keyboard shortcuts enable you to work (and play) faster. No need to take the time to pick up your hand, move it to a different spot on the desk, use the mouse, click something, move your hand back to the keyboard to type a few letters, go back to the mouse to click something else, and so on. The more you can do from one place, the more efficient your actions will be. Here is a list of things you can do with Unity without moving your hands away from the keyboard. Of course, you don't have to do things this way, but power users love to save time and energy, even if it means spending a little more time up front learning how; in the long term, the savings add up.

TIP These keyboard shortcuts are new. Some are very different from what you may have used with previous operating systems or even earlier desktops with Ubuntu. Also, they are actively being developed and tested with end users for maximum usability and memorability. This means some may change from the time this was written, although we hope not.

The Super key is used for several of the shortcuts. It is sometimes known as the Windows key. In the following tables, it is simply called Super.

Table 4-2 lists keyboard shortcuts related to using the Launcher.

Dash

Table 4-3 lists keyboard shortcuts related to using the Dash.

Table 4-2 Launcher Shortcuts

Key Combination	Action
Super (long press)	Open Launcher, display shortcuts
Super-Tab	Switch applications via the Launcher
Super-# (1–9)	Same as clicking on a Launcher icon
Super-Shift-# (1–9)	Open a new window for the same application
Super-T	Open Trash

Table 4-3 Dash Shortcuts

Key Combination	Action
Super (quick tap)	Open the Dash home
Super-A	Open the Dash App Lens
Super-F	Open the Dash Files Lens
Super-M	Open the Dash Music Lens
Ctrl-Tab	Switch between Lenses
Cursor keys	Move the focus
Enter & Return	Open the item that currently has focus

Switching

Table 4-4 lists keyboard shortcuts related to switching between applications.

Windows

Table 4-5 lists keyboard shortcuts related to windows and managing them.

Workspaces

Table 4-6 lists keyboard shortcuts related to using workspaces.

Other

Table 4-7 lists keyboard shortcuts that don't fit in any of the previous categories.

Table 4-4 Switching Shortcuts

Key Combination	Action
Alt-Tab	Switch between applications
Alt-`	Hold Alt and press ` to switch between windows of current application
Cursor left or right	Move the focus within lists or images of applications when switching or in a menu

Table 4-5 Windows Management Shortcuts

Key Combination	Action
Super-W	Spread all windows in the current workspace
Super-D	Minimize all windows
Super-Cursor up	Maximize the current window
Super-Cursor down	Restore or minimize current window
Super-Cursor left or right	Semi-maximize current window
Alt-F4	Close current window
Alt-Space	Open window accessibility menu
Ctrl-Alt-Num	Place window in corresponding positions on the screen
Alt-Left mouse drag	Move window
Alt-Middle mouse drag	Resize window
Ctrl-Alt-Numpad 7	Place window in top left corner of screen; pressing a second time does nothing
Ctrl-Alt-Numpad 8	Place window in top half of screen; pressing a second time does nothing
Ctrl-Alt-Numpad 9	Place window in top right corner of screen; pressing a second time does nothing
Ctrl-Alt-Numpad 4	Place window on the left side of the screen in semi-maximized state (it is important that the window is actually semi-maximized, not just the same size and position as a semi-maximized window); pressing a second time does nothing
Ctrl-Alt-Numpad 5	Maximize window; if window is already maximized, pressing this key combo restores the window to the same size, shape, and position as before it was maximized
Ctrl-Alt-Numpad 6	Place window on the right side of the screen (it is important that the window is actually semi-maximized, not just the same size and position as a semi-maximized window); pressing a second time does nothing
Ctrl-Alt-Numpad 1	Place window in the bottom left corner of the screen; pressing a second time does nothing
Ctrl-Alt-Numpad 2	Place window in the bottom half of the screen; pressing a second time does nothing
Ctrl-Alt-Numpad 3	Place window in the bottom right corner of the screen; pressing a second time does nothing
Ctrl-Alt-Numpad 0	Minimize window; if window is already minimized, pressing this key combo restores the window to the same size, shape, and position as before it was minimized

Table 4-6 Workspace Shortcuts

Key Combination	Action
Super-S	Spread workspaces
Super-Ctrl-Cursor Keys	Switch workspaces
Super-Alt-Cursor Keys	Move focused window to different workspace

Table 4-7 Other Shortcuts

Key Combination	Action
Super-L	Lock screen
Ctrl-Super	Display an overlay image of the main keyboard shortcuts for Unity

Using the Terminal

Although Ubuntu is a desktop-driven OS, the system is running on a powerful and incredibly flexible command-line core. Inspired by more than 30 years of UNIX heritage, the command-line environment present on Linux systems enables you to perform some incredibly powerful tasks by stringing together different commands in different ways.

The philosophy behind UNIX is to create a large number of small tools, each of which is designed to do one task but do it incredibly well. As a quick example to whet your appetite, there is a command called `ls` that does nothing more than list files in a folder. Although listing files is its singular function in life, it has every option imaginable for listing files.

Now, `ls` is limited by itself, but it can be combined with other commands that have equal levels of flexibility to create impressively powerful combinations. To do this, a *pipeline* is created using the `|` symbol to connect these different commands. Pipelines can be constructed in any number of different ways, and once the user has even a basic knowledge of what a few different commands do, stringing together a pipeline of commands can solve virtually any task you can imagine in quick and powerful ways.

It should be made 100 percent clear that using the command line is *not* an essential skill required to use Ubuntu, but it is a skill that can increase the flexibility of your computer for more advanced, customized tasks. Rather than cover the use of the terminal here, we have included an excellent introduction in Chapter 7 and we strongly recommend that all who seek to become power users read it.

Working with Windows Programs

Although Linux offers an increasingly compelling platform for the desktop, some situations arise when there is just no alternative other than an application written for Windows. This is generally the case with specific business applications, some educational tools, and many games. Luckily, there is a way you can run many of these applications on your Ubuntu desktop.

For more than fifteen years, the Wine project team members have been working to create a free way to run Windows applications on Linux. While not every application works perfectly, and some don't work at all, the number of programs that do work in Wine has dramatically increased and continues to do so. However, it is recommended that you thoroughly test the applications you want to run in Wine before you use them for important work, and if you run into trouble, try consulting help resources, use virtualization to run Windows on top of Ubuntu, or search for a different application to use.

TIP You can find some useful help resources for Wine at www.winehq.org/help/, and you can learn more about alternatives to Wine at www.winehq.org/docs/wineusr-guide/alternatives.

Install the Wine package from the Ubuntu Software Center or simply double-click an .exe file, and you will be prompted to install the package. You can configure Wine by searching for Wine Applications in the Applications Dash after clicking the Applications icon in the Launcher. Your C:\ drive will appear in your Places menu for easy access, and you will be able to uninstall Wine from the Ubuntu Software Center.

Running Applications

To run an application, simply double-click on the install .exe file. Once installed, the program should appear in your menu under Applications > Wine.

TIP You can find even more about Wine from the Ubuntu perspective at https://help.ubuntu.com/community/Wine.

Installing Software from PPAs

Sometimes software is not available in the Ubuntu repositories; perhaps it is too new, perhaps there is a new version available, or perhaps it is software that no one has had the opportunity to build and get approved for inclusion in the Ubuntu repositories. You can often find this software in a Personal Package Archive, or PPA. The main Launchpad page for PPAs says, "Personal Package Archives (PPA) allow you to upload Ubuntu source packages to be built and published as an apt repository by Launchpad" (see https://launchpad.net/ubuntu/+ppas).

In short, a PPA is a small repository that contains software that is not found in the main Ubuntu repositories, perhaps for one of the reasons listed earlier or for other reasons. If a software package or version you want to install is not available from the main Ubuntu repositories, use your Web browser to search the main Launchpad site for the software and see if a PPA exists.

TIP **Be Careful!**
Some PPAs are created for development or for beta or even alpha testing. They are not guaranteed to include stable or well-tested software. Use at your own risk.

For our example, we install a bleeding-edge testing version of LibreOffice. Unless you absolutely need something offered from a PPA, we suggest you stick with packages from the Ubuntu repositories.

To add the PPA to your system, make a note of its location from the specific PPA's Web page in Launchpad. In this case, the PPA is at ppa:libreoffice/ppa, as seen in Figure 4-10.

Figure 4-10 You can find technical details and installation instructions on all PPA pages.

The easiest way to add the PPA is to use the Terminal. Enter the following at the command line to add the LibreOffice PPA and to update your package manager's listing.

```
$ sudo add-apt-repository ppa:libreoffice/ppa
$ sudo apt-get update
```

Then, you can access the contents of the PPA from your preferred package manager: Ubuntu Software Center, Synaptic, the command line, and so on.

Compiling Software from Source

Some software cannot be found in any repository. The traditional way of installing software on Linux works just as well on Ubuntu. Download your software and put it in an otherwise empty directory. Sometimes, it will include installation instructions. If so, follow them. Most instructions will be the same as those listed here.

Before you can install a source code package, it must be compiled into a binary that your computer can run. For that, you need to install some packages that are not included by default. All you should need is included as a dependency of one package. Install it, and you should be all set.

```
$ sudo apt-get install build-essential
```

Next, browse to the directory where you placed the downloaded source code. It was most likely provided as a compressed archive, with a filename

extension like tar.gz or tar.bz2. Decompress the archive with the appropriate command, depending on the filename extension:

```
$ tar -xvf file.tar.gz
$ tar -xvf file.tar.bz2
```

Next, make the package. This step checks to ensure you have the required dependencies installed and will tell you if you don't—if not, you need to install them separately, hopefully using Synaptic or via the command line with apt. It will also build the package from source code into something installable.

```
$ make
```

Next, install the package. The traditional way is to use:

```
$ sudo make install
```

And although this works, there is a better method. Using `make install` will install the package, but it will not use your package manager. If you ever want to remove the package from your system, it can be quite a chore to figure out how. Instead, install the following:

```
$ sudo apt-get install checkinstall
```

and use it to install the software, which, like any software installed from an official repository, cannot be removed using your package manager.

```
$ sudo checkinstall
```

Summary

In this chapter we looked at a variety of advanced subjects related to running and managing your Ubuntu system. Installation, removal, and upgrade of software using the Ubuntu Software Center and other options were discussed. We also looked at the installation and use of several different types of hardware devices. We perused some of the methods of accessing remote files and mentioned the powerful Ubuntu terminal and the need for learning how to back up your data regularly. Finally, we looked at a possibility of running certain programs written for Microsoft Windows under Ubuntu and even compiling software from source code.