

# 1

## Why Learn About Linux? What's in It for Me?

**“I**’ve already got Windows 2000 and XP running and mostly stable—and *they* were no picnic—why should I care about Linux?”

There are several reasons. For one, Linux can do many things that your network needs and, furthermore, it may be able to do at least a few of them better than any Microsoft OS can. Second, even if it’s just a toss-up between NT and Linux in functionality, you can get as many copies of Linux as you want, free.

And don’t forget that there’s a decent chance that at some point the boss is going to come ask you about “this Linux thing”—a couple of his golf buddies claim to be using it to good effect, so he’s giving you the hairy eyeball and asking why *we’re* not using this modern software miracle? Knowing about Linux may either equip you to tell him why it’s not the answer for you, or enable you to explain that yes, Linux would be useful, here’s how, and here’s what you’ll need in the way of resources to make it happen. And if he takes you up on the request for more resources, then it’d be nice to know how to make NT and Linux interoperate!

Also, Linux is a cheap way to learn many skills that are not extremely operating system-specific but that are marketable: for less than 20 bucks, you have a tool that will let you learn C, Perl, or Python programming, webmaster skills, or basic e-mail server administration, to name a few things. Those skills might be of dollars-and-cents value to you

personally, as demand for Linux administrators is growing rapidly. And, finally, whether it succeeds or not, Linux offers an alternative model for software development and marketing, and that different way of thinking cannot help but affect how more traditional software firms like Microsoft sell their wares.

## Linux Can Be a Powerful Server Alternative (or Complement)

As you'll read in the next chapter, people who use Linux tend to use it for a particular set of tasks, a set that's a bit different from what people use NT or 2000 for. By knowing what Linux is good for (and what it's not so good for), you can intelligently plan where to use it and where to use NT instead.

For example, if you need a file server that Windows boxes can talk to and you've already got NT running, then you're probably best off leaving that unchanged. But if you're thinking of setting up a DNS server, then you might want to slap Linux on a box and run DNS on the Linux box instead of an NT box. First, every copy of Linux includes, free of charge, a piece of DNS server software called BIND, which is the most well-known, tested, robust piece of DNS server software that you can get your hands on. Second, as you'll read in the next section (and as you probably know), Linux is basically free software. Third, you can get DNS running well on PC hardware as modest as a 100MHz Pentium with 32MB of RAM, or perhaps less. Fourth, you might just want to run Linux as a DNS server *even if* you're running an NT or 2000 DNS server, because you might need a second DNS server (as you do if you're hosting your own DNS server on an Internet domain) and don't have an extra NT license for that second one.

## Linux Software Licenses Are Very Cheap

Notice I *didn't* say "free."

Yes, if you have the bandwidth and/or the patience, you can surely download an entire Linux distribution and not pay anyone a dime for it. When Craig Hunt and I worked on the first version of this book, we thought it'd be interesting to try to download Red Hat Linux from Red Hat's site with my 56K modem. Windows 2000 Professional started downloading the Red Hat files, reporting that the estimated remaining time was "13 days, 6 hours, and 13 minutes." Needless to say, we terminated the download and decided that it'd be quicker to run up to Best Buy and get the CD-ROM.

But is Linux *really* free? Absolutely not. Suppose I gave you, free of charge, all of the NT Server licenses that you could ever want. Would NT then be free? Well, the software might be, but NT doesn't run itself. You need administrators and consultants to keep NT running, and those administrators and consultants aren't cheap. (And speaking as a consultant, let me say, "Thank *heavens* that we're not cheap!")

It's the same story with Linux. Sure, Linux is powerful, but it's not simple. Despite the efforts of large Linux vendors to make GUI front-ends for Linux almost mandatory, the fact remains that any Linux administrator worth his or her salt is comfortable working with cryptic-looking configuration files. One doesn't learn to be a Linux guru overnight any more than one can become an NT, 2000, XP, or .NET guru overnight. So let me be clear about the fact that if you choose to incorporate Linux in your network, then you should expect either to incur some training costs or be prepared to pay for someone who's already taken the time to develop a Linux expertise.

On the other hand, I should add a positive note here: while Linux techies may not be plentiful, Unix techies *are* a bit more common, and Linux's similarity to Unix means that it's moderately simple for someone who already knows Unix to "cross over" to Linux expertise.

The fact that Linux software licenses are free, though, *can* save you money. Yes, Linux-savvy administrators cost money, but just one administrator can watch over several servers. Going from one Linux box to two, then, entails very little marginal cost. Even better, you can often put Linux on lower-power hardware than you'd normally choose for NT, so you can save money on the hardware as well. In the case of my network, I found that moving from Windows NT to Windows 2000 Server forced me to retire my old primary domain controller, a Pentium Pro 200MHz system; I replaced it with a Pentium III 600MHz server. That Pro 200 would have done a terrible job of running Windows 2000, but now that it's freed up, it's a terrific Linux server.

On the one hand, then, introducing Linux into your network can cost you in terms of personnel. But on the other hand, Linux can save you money on software licenses and on hardware.

## NT/2000/XP/.NET Administrators Need Linux Interoperability Facts

Microsoft never lets a year or two go by without a major change in NT, which typically means that just when you've got "Windows ZZ.DOT" mostly figured out, then you've got to dope out "Windows US.XXX," so it's not like you've got a pile of free time. (Don't worry, those aren't real versions of Windows, just names I made up. Remember you

heard them here first.) But, as I've already said, that doesn't stop the boss from asking you, "Hey, how come we're not using Linux like all my golf buddies ... errr, I mean, other CIOs in town?" So now you have to figure out Linux, and more important, you may have to figure out how to make Linux fit into your existing network, whether Linux is right for your organization or not.

Why? Well, of course, some of you need to know how to integrate Linux into your Microsoft-based network just because Linux makes *sense* in your organization. Some will just have to do it because Linux is the "flavor of the month," a new thing to try for the early accepters in the management suite. No matter what the reason, it's a fact that many Windows administrators will benefit from knowing at least enough Linux to know what they can do to integrate it with their system.

And there are a *lot* of things that you can do to make Linux and Windows networks work well together, as you'll see. They can easily share files and printers, work together in supplying e-mail, web, and other Internet infrastructure services.

As an NT/2000/XP/.NET administrator, your boss and/or clients are very likely to look to you for advice about Linux—should they include it in their existing Microsoft-based networks? Can it replace or complement some or all of their existing network infrastructure, or is it a waste of time to even consider that? Knowing at least a bit about Linux will help you answer those questions.

## Linux Is a Cheap Way to Learn Programming, Web Administration, and Unix

If you have an interest in learning the Perl, Python, or C languages, then Linux is a great and nearly free platform—free, that is, excluding the time to learn a new environment. In some ways, education was Linux's first task, as it started out as a learning tool for Linus Torvalds. To this day, students at hundreds of universities use Linux to learn not only programming, but also how operating systems work. If you want to learn programming using the Microsoft tools, you have to buy a lot of stuff—the operating system, the C compiler, the development environment. But every version of Linux that I've ever seen ships with all the tools you need to create and test programs of all kinds, whether they're graphical games, hardware device drivers, network services, or even basic internal operating system functions like memory managers. And all at that attractive Linux price!

But Linux isn't just for learning to program. It's also a good way to learn and practice many administration tasks. Experience with Linux's Apache web server, its Sendmail mail server, or its BIND DNS server would all translate to good knowledge of how these

work on a Unix system, which might pave the way for more job opportunities. Even *if* Linux doesn't succeed, it can't hurt to know how to run Unix networks, and Linux expertise takes you an awfully long way toward being a Unix expert.

## Linux Employs a Nontraditional Software Business Model

The Microsoft Way to make money in the software business is to build software and sell it, and not to let the buyer see the “source code”—the blueprints for the software. Many other firms, like Novell, Lotus, Symantec, Corel, and Intuit, have prospered using that same model of software manufacture and sales. But that's not the only way, as firms like MandrakeSoft and Red Hat Software have demonstrated; *their* model is to build it, then give it away. Clearly *someone's* making money doing software a different way. Here's an easy way to learn more about it.

---

**NOTE** You can learn more about this innovative approach to software marketing in Chapter 3.

## You Can Use Linux to Create Finely Tuned Operating Systems

While this isn't something *everyone* is going to do, many people have exploited the fact that Linux ships with its complete source code to do something pretty neat: they simply cull from the operating system anything that they'll never use.

Suppose you knew that you were going to roll out 200 machines that would be e-commerce web servers, and that your firm had decided to use Linux and its included web server, Apache. Two hundred machines are a *lot* of machines; buying them represents a fair amount of money that your firm is laying out. How, you are asked, can we get the maximum performance from these systems?

Well, if these were NT or 2000 servers, you'd study up to find the right Registry entries that would tweak more throughput out of your machines—that's about the most that you could do. But these are *Linux* boxes. So you go hire a C-and-Linux expert. She then modifies the Linux code, throwing out the pieces that you will never use: the servers will never have FAT or NTFS partitions, throw that code out ... no need for Samba, or NFS... power management's unnecessary...and in no time she has pared the base Linux operating system down to half its size.

**NOTE** Don't believe me? There are several groups that have boiled down Linux to *one single floppy disk* so that you can use an old machine without a hard disk as an IP router! One example is at [www.linuxrouter.org](http://www.linuxrouter.org). And Dan tells me, "The list of distributions at Linux Weekly News now lists 19 distributions that can run off a floppy! See the list at <http://lwn.net/Distributions/>. Floppy 'distros' start down around 148 with Brutalware." And if that's not enough, I've seen business-card-sized CD-ROMs with bootable copies of Linux.

But she doesn't stop there. All of that free space lets her expand some of Linux's memory buffers, allowing Linux to keep more of its essential data in RAM rather than having to constantly run out to the far-slower disk drive for that data. Result: a special-purpose version of Linux that runs like the wind on fairly modest hardware.

So perhaps you only need *one* hundred of those machines. Hey, who knows, you might even get a bonus for saving the company a couple of bucks! (No, huh? Yeah, I think I used to work for that company, too.)

## You Can Find Uses for Older Hardware

Part of the Linux Legend is that you can get fantastic performance out of ancient hardware—that all of those moldy old 486s can run Linux and outperform NT or 2000 on Pentium IIIs. My personal experience leads me to say that much of that is merely hype, as a Red Hat 6.1 system running the GNOME graphical interface needs (in my opinion) 64MB of RAM, as well as at least a 1024 × 768 graphic mode and a Pentium II to offer decent performance.

*But* if you can live without that graphical user interface stuff, and if you have a distribution of Linux that doesn't *require* you to load graphics, then yes, some of the legends of still-useful 486s aren't completely absurd—although for fairness' sake, you might consider putting Linux on a senescent 100MHz Pentium rather than the *extremely* senescent 486.

What would you expect to get done with a 100MHz Pentium with 32MB of RAM and a 2GB hard disk? Very little, with NT Server or 2000. But such a system could work admirably as a DNS server or a web server, provided again that you don't throw away most of the CPU's power on an unnecessary GUI. Or, you might well decide that you'd *like* to throw away some of the CPU's power on a GUI, but only occasionally—that's the beauty of Linux. Depending on the version of Linux that you use, you'll probably be able to set the system to come up in text mode with only a text shell and no graphics, but then to start up the GUI when you need to do administration, if you prefer to use graphical

administration tools. (You needn't—every administration task that I've come across can be done from the text shell, and I will try very hard throughout this book to show you how to do everything from the command shell. Furthermore, I live that reality—as my e-mail server is a Linux box running on an Alpha chip, I can't get the GUI to work at all, so I *have* to do all of my administration from the command line.)

## Demand for Linux Administrators Is Growing

What better reason is there ever for learning things than filthy lucre? For many NT administrators, NT administration is their first IT-related job, and they made the choice to go get some NT training specifically so that they could move from a non-IT job to one that paid more money. NT jobs pay well because of the demand for NT administrators, a demand that grew out of companies adopting NT in great numbers. While it's still true that more and more firms are adopting NT and 2000, and so that demand will continue for the near future, it's every bit as true that many of those same firms and others besides are incorporating Linux in their networks. So it's reasonable to guess that there will be a demand for Linux administrators as well.

But is that guess valid: is there really a great unfulfilled demand for Linux administrators? Believe it or not, that's a somewhat tough question to answer. I attended the 1999 Fall Comdex in Las Vegas because I was invited by the Comdex folks to give a couple of talks about the then-upcoming Windows 2000. While I was there, I noticed that Comdex offered talks on Linux as well—in fact, an entire other track. I sat in one panel discussion where the participants talked about whether or not there was a growing market for Linux administrators. Each of the panel members had one of two different points of view. The first point of view was, “Yes, it's hard to find Linux administrators. It's *always* been tough to find good Unix administrators, why would Linux be any different?” The other point of view was, “Look, almost anyone graduating with a computer science degree since 1995 has been using Linux at his or her college/university classes—just hire a youngster right out of school.”

In his Comdex talk, Craig—who was one of the invited Comdex speakers, although on Linux, not 2000—observed that a recent salary survey showed that Linux system administrators were being paid more than NT administrators. (There were no 2000 administrators at the time, recall.) He had an interesting perspective on what causes that. In Craig's view, NT administrators are suffering salary-wise “because your managers *don't respect you!*” He says that with tongue partly in cheek and goes on to explain that what he means is that NT *looks* like Windows, and managers use Windows, so they reason to themselves, “Hey, *I* can use Windows, so how hard could it be to run a network based on this Windows-like thing?” In contrast, Linux carries with it the complexity of Unix,

which “everyone knows” is an operating system that requires at least minor genius to keep working—so *of course* we have to pay those Linux guys a lot!

Craig got the salary figures from the *1999 SANS System, Network and Security Administration Salary Survey* from the SANS Institute. The majority of the administrators in the survey, 63.5 percent, are NT administrators, and the next largest group, 30.5 percent, are Unix/Linux administrators. Page 9 of the survey states that the “average NT salary was \$53,598/year” with the “average Unix salary at \$62,907/year.” A difference of almost \$10,000! Despite the fact that the demand for NT administrators is so high that more than twice as many respondents use NT than Unix, the value assigned to the Unix positions is \$10,000 greater than the NT positions. Clearly having some Unix/Linux skills can be good for your wallet. (The SANS Institute is an organization of system administrators focused on system and network security. You can find them at [www.sans.org](http://www.sans.org), and no, I've not been able to find a more up-to-date survey, but I still believe that Linux/Unix knowledge is a useful asset for an IT administrator.)

## It'll Help You Use Your Macintosh

Believe it or not, this is utterly true.

I bought a couple of Macs back in 1993 so that I could learn how to network them to NT for the first version of my NT book. (A IIci and a LCii if I recall right.) Playing with them was fun but ultimately caused me to write off the Mac as religion and hype.

Why do I say that? Because of the basic architectural weakness of its underlying operating system. First, the Mac had a basically nonexistent memory management system: you, the user, had to hand-allocate more or less memory to every application. Second, the multitasking system was a very weak “cooperative” multitasking system, meaning that once a given application got control of the computer, there was nothing that anyone could do to wrest control back from that application. Thus, every application was on the “honor system” not to hog the computer.

In short, the Mac OS was, in my opinion, as architecturally bankrupt as Windows 3.0. That didn't bother many users, as Macs benefit from the fact that 99 percent of the hardware in a Mac is hardware all designed by the same people who built the operating system, and because the smaller nature of the Mac market means that there are fewer applications and therefore the average bear ends up trying to make a smaller number of apps work together. As a result, purist geeks like me were right in turning up our noses at the Mac on theoretical grounds... but Mac lovers were right in a practical sense that their Macs did what they wanted them to do. Then came Mac OS/X.

Apple did an incredibly smart thing with OS/X: they blew up the Mac OS and started over. And, as you may know, here's what else they did: they started from a variant of Unix called BSD (which stands for Berkeley Software Distribution, not that that tells you anything), a rock-solid OS. Then they put a GUI atop it that was as beautiful as you'd expect from the Guys From Cupertino. The result is one nifty-looking OS that answers all of my objections: BSD does quite well in the memory management and multitasking areas, thank you very much.

Now inasmuch as BSD and Linux are both just variations on Unix, much of the stuff that you'll learn here can be immediately converted into useful Mac geek knowledge. (A side benefit, yes, but perhaps a useful one.)

Because it has clearly passed the "laugh test," and because it's an arguable good value for many businesses, Linux is an opportunity for the IT world in general. But that also means that it's an opportunity for your professional growth. *That's* probably the best reason to find out about Linux.

