

The Complete PC Tech

In this chapter, you will learn how to

- Describe how computers work
- Explain the nuances of dealing with customers
- Implement a troubleshooting methodology

When a mission-critical computer goes down, regardless of the industry, people get upset. Workers can't work, so they feel guilty. Employers can't get product out on time, so they feel anxious. Supervisors blame employees for fouling things up, or at least the employees fear such blame, even if they did not break the machine.

Into this charged atmosphere comes the tech, ready to fix the computer and move on to the next challenge. Accomplishing this task, though, requires three things: First, a good tech must know the broken machine inside and out—how it's *supposed* to work when working properly. Second, the tech has to calm the workers and supervisors, and get answers to questions to gain relevant information about the problem. Third, the tech must troubleshoot the problem and fix the machine.

This chapter starts with an overview of how computers work and then dives into a section on dealing with customers and how to get them to tell you what you need to know and smile about it. The chapter wraps up with a proven troubleshooting methodology to help you figure out the source of problems and point you to the fix quickly.

Essentials

How Computers Work

You've spent a lot of time going through this book, reading about technologies and components in great detail. Each chapter contained troubleshooting information and methodologies for the components explained in that chapter. In Chapter 3, for example, you learned all about CPUs, from how they work to how to install them. You also learned about issues specific to CPUs, including the potentially difficult task of adding or removing the fan and heat sink assembly that all CPUs require. In Chapters 9 and 10, you dove into hard drives in gory detail. With each chapter, you added more and more information about the pieces that make up the personal computer today.

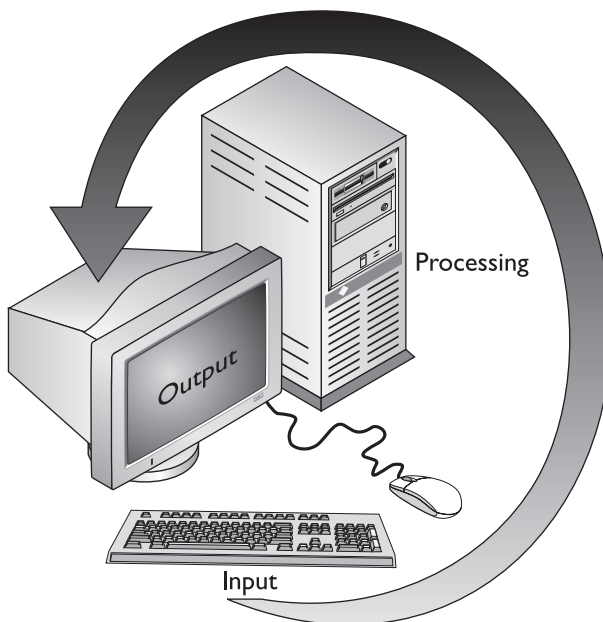
In this chapter, I want you to distill that knowledge, to think about the computer as a coherent machine. Each of the computer's components works together to enable people to produce some amazing things.

To master the art of troubleshooting as a PC tech, you need to approach a technical problem and answer one question: "What can it be? What can be causing this problem?" (Okay, that was two questions, but you get the idea.) Because every process involves multiple components, you must understand the interconnectedness of those components. If Jane can't print, for example, what could it be? Connectivity? Drivers? Paper jam? Slow network connection? Frozen application? Solar flares? Let's look at the process.

Computing Process

When you run a program, your computer goes through three of the four stages of the computer process: input, processing, and output (Figure 24-1). Input requires specific devices, such as the keyboard and mouse, that enable you to tell the computer to do something, such as open a program or type a word. The operating system (OS) provides an interface and tools so that the microprocessor and other chips can process your request. The image on the monitor or sound from the speakers effectively tell you that the computer has interpreted your command and spit out the result. The fourth stage, storage, comes into play when you want to save a document and when you first open programs and other files.

Figure 24-1
Input, processing,
and output



Making this process work, though, requires the complex interaction of many components, including multiple pieces of hardware and layers of software. As a tech, you need to understand all the components and how they work together so when something doesn't work right, you can track down the source and fix it. A look at a modern

program reveals that even a relatively simple-seeming action or change on the screen requires many things to happen within the computer.

Games such as Second Life (Figure 24-2) are huge, taking up multiple gigabytes of space on an Internet server. They simply won't fit into the RAM in most computers, so developers have figured out ways to minimize RAM usage.



Figure 24-2 Second Life

In Second Life, for example, you move through the online world in a series of more or less seamlessly connected areas. Crossing a bridge from one island to another triggers the game to update the information you're about to see on the new island quickly, so you won't be out of the action and the illusion of being in the game world remains intact. Here's what happens when you press the w key on your keyboard and your character steps across the invisible zone line.



NOTE Second Life is a massively multiplayer online role-playing game (MMORPG) that offers a unique twist on the genre. You can create just about anything you can imagine, as far as your time and talent can take you. Second Life has a functioning economy that spills out into the real world, meaning you can buy and sell things within the game and turn that into real US dollars, although the more common scenario is to spend real money to get virtual possessions.

The keyboard controller reads the grid of your keyboard and, on discovering your input, sends the information to the CPU through the wires of the motherboard (Figure 24-3). The CPU understands the keyboard controller because of a small program it loaded into RAM from the system BIOS stored on the system ROM chip on the motherboard when you first booted the computer.

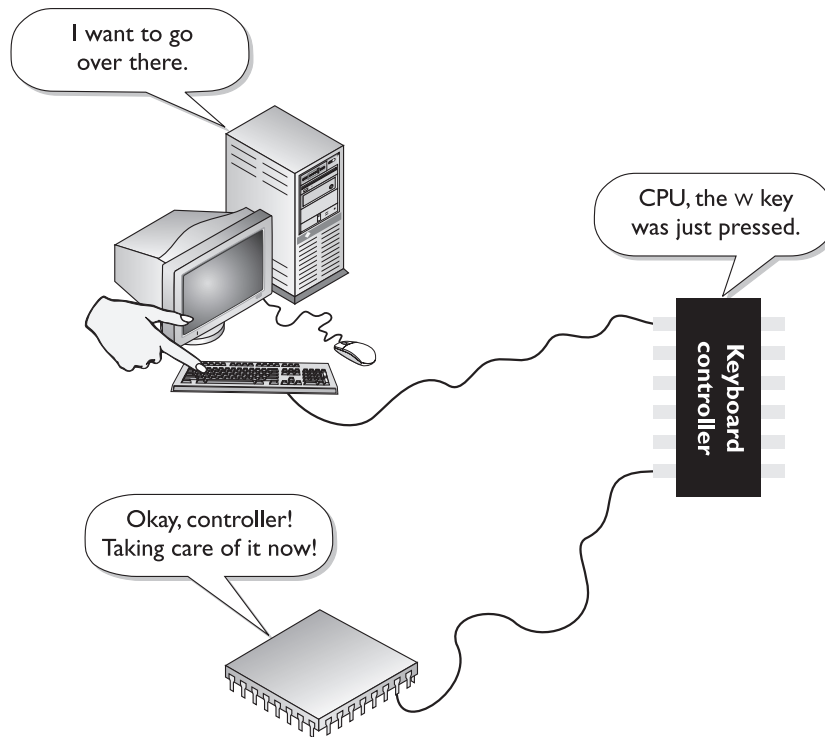


Figure 24-3 Keyboard to CPU

The CPU and the application determine what should happen in the game, and on discovering that your character is about to cross the zone line, they trigger a whole series of actions. The application sends the signal to the operating system (OS) that it needs a specific area loaded into RAM. The OS sends a signal to the CPU that it needs data stored on the hard drive plus information stored on the Second Life servers. The CPU then sends the commands to the hard drive controller for it to grab the proper

stored data and send it to RAM, while at the same time sending a command to the NIC to download the updated information (Figure 24-4).

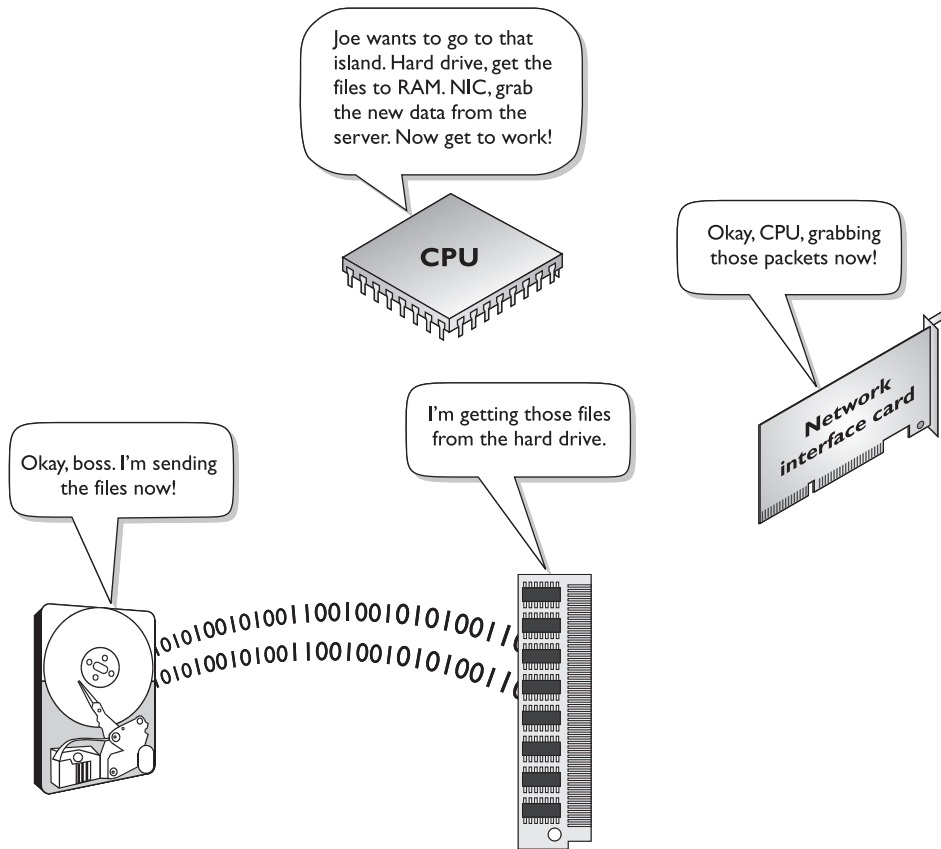


Figure 24-4 CPU to hard drive and NIC

The hard drive controller tells the hard drive to cough up the data—megabytes worth—and then sends that data through the motherboard to the memory controller, which puts it into RAM and communicates with the CPU when it's finished. The network card and network operating system communicate with the Second Life servers and download the necessary updated information. The CPU then uses the application and OS to process the new data, sending video data to the video card and sound data to the sound card, again through the wires on the motherboard (Figure 24-5).

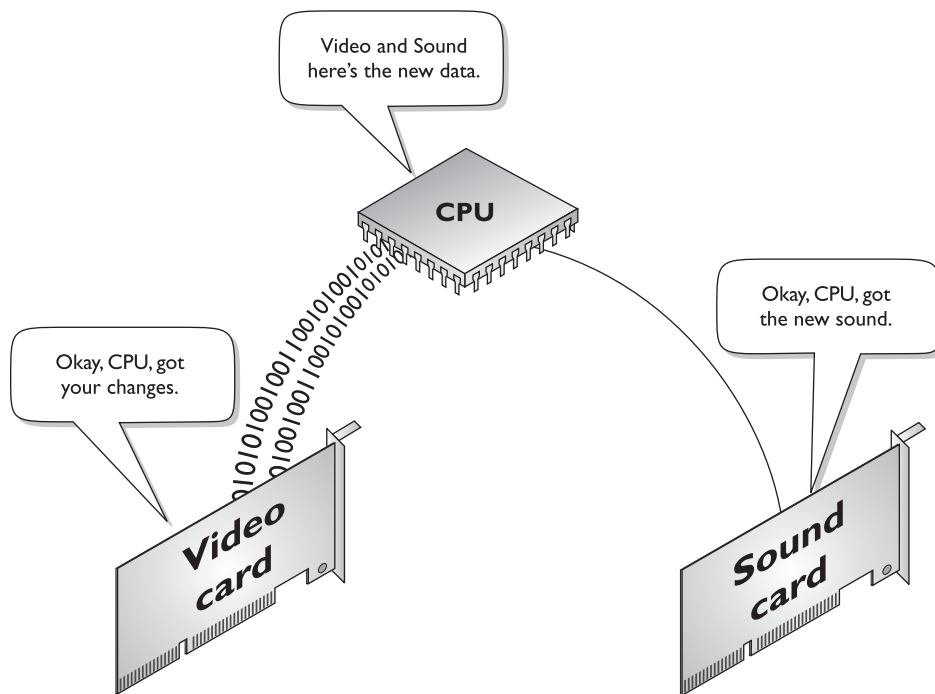


Figure 24-5 CPU to video card and sound card

The video card processor puts the incoming data into its RAM, processes the data, and then sends out commands to the monitor to update the screen. The sound card processor likewise processes the data and sends out commands to the speakers to play a new sound (Figure 24-6).

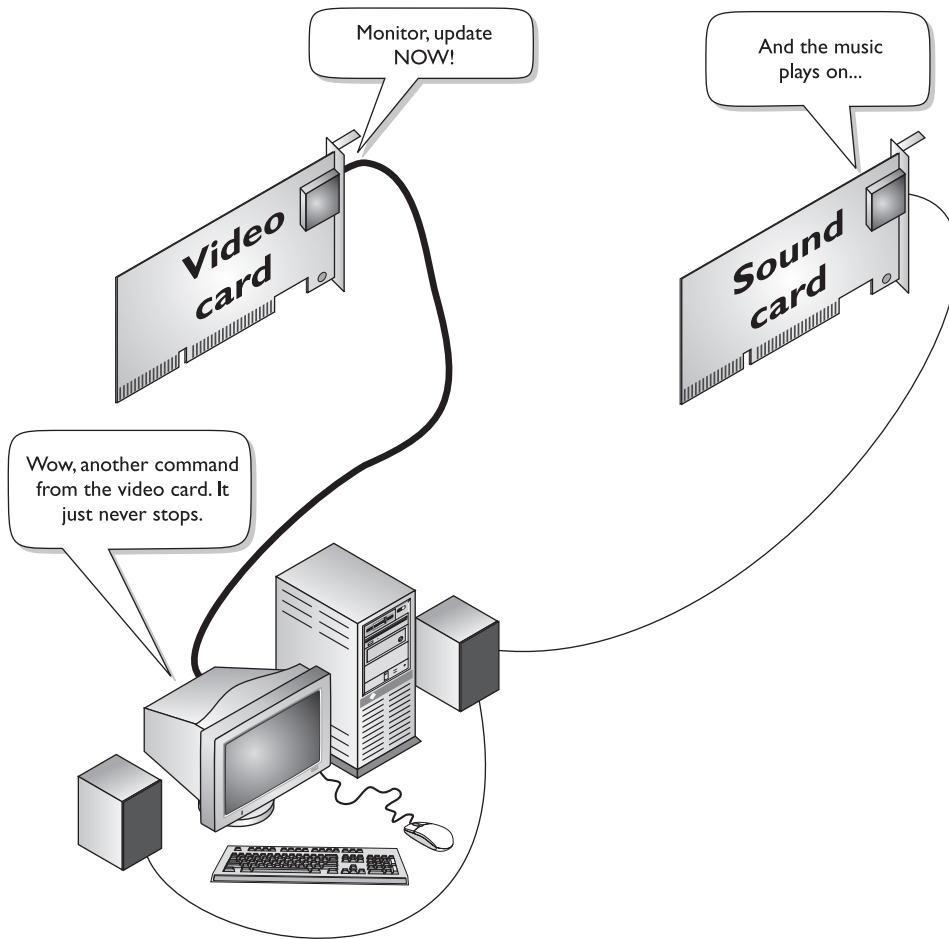


Figure 24-6 Updating the screen and speakers

For all of this to work, the PC has to have electricity, so the direct current (DC) provided by the power supply and the alternating current (AC) provided to the power supply must both be the proper voltage and amperage.

Finally, because Second Life is a network application, the OS has to send information through the NIC and onto the Internet to update everyone else's computer. That way, the other characters in the game world see you move forward a step (Figure 24-7).

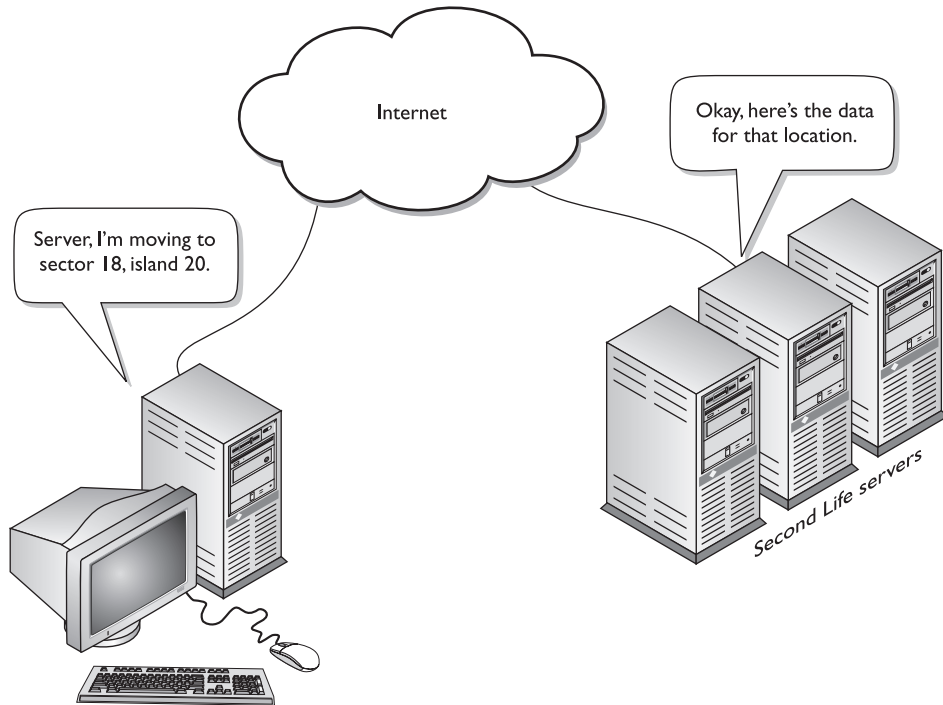


Figure 24-7 PC to Second Life servers

What do you see or hear with all these electrons zipping all over the place? Out of a seemingly blank vista (Figure 24-8), a castle begins to appear, building itself piece by piece as your computer processes the new information and updates the video screen.

You hear music begin to play from your speakers. Within a few seconds, with the data describing the new island fully downloaded and processed, the world on your monitor looks very different (Figure 24-9). That's when all goes well. Many megabytes of data have flowed from your hard drive and across the Internet, been processed by multiple processors, and sent to the monitor and the speakers.



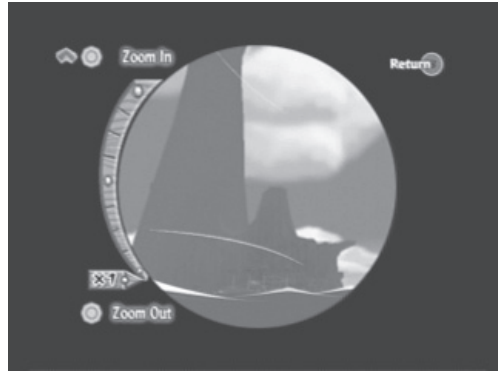
Figure 24-8 New area loading



Figure 24-9 Castle completed

To keep the action continuous and unbroken, Second Life, like many current online games, uses a process of continuous or *stream loading*: your computer constantly downloads updated information and data from the Second Life servers, so the world you see changes with every step you take. When done right, stream loading can do some amazing things. In the GameCube game *Zelda*, for example, the game anticipates where you will go next and loads that new area into RAM before you take the step. You can be in one area and use a telescope to zoom in on another fully developed area (Figure 24-10), making the experience amazingly seamless, just like real life.

Figure 24-10
Zelda zoomed



Troubleshooting

Good techs understand the components involved in inputting, processing, and outputting data, including the devices that store data, such as hard drives. That's because if something doesn't work properly, you can start answering the ultimate troubleshooting question—what can it be?—accurately. If your screen freezes or the sound goes wonky, where in the process is the problem located?

As you go into any troubleshooting scenario, always keep the computing process in mind. This helps you sort through possibilities quickly and accurately. If you know all the stages, you won't miss a simple step, such as figuring out that a user can't print because the cleaning service accidentally turned off the print server the night before, or waste time reinstalling printer drivers when the real issue is a stalled print job in the print queue.

Dealing with Customers

When you deal with users, managers, and owners who are frustrated and upset because a computer or network is down and they can't work, your job requires you to take on the roles of detective and psychologist. It takes skill to talk with frazzled and confused people and get answers to questions about how the PC got into the state it's in. It's important to be able to communicate clearly and effectively. Plus, you need to follow the rules of tech-person decorum, acting with personal integrity and respect for the customer. Finally, use assertive communication to empathize with and educate the user. Great techs spend the time needed to develop these essential skills.



EXAM TIP This section has no IT Technician section because the entire chapter applies to Essentials, IT Technician, and Help Desk Technician exams. Expect 1 out of every 5 questions on the 220-603 exam to be on Communication and Professionalism, so you have to know this stuff!

Eliciting Answers

Your job as a tech is to get the computer fixed, and the best way to start that process is to determine what the computer is doing or not doing. You must start by talking to the customer. Allow the customer to explain the problem fully while you record the information. Once the person has described the situation, you must then ask questions.

Although each person is different, most users with a malfunctioning computer or peripheral will be afraid and often defensive about the problem. To overcome this initial attitude, you need to ask the right questions *and* listen to the customer's answers. Then ask the proper follow-up questions.

Always avoid accusatory questions because they won't help you in the least. "What did you do?" generally gets a confused or defensive "Nothing" in reply, which doesn't get you closer to solving the problem. First, ask questions that help clarify the situation. Repeat what you think is the problem after you've listened all the way through the user's story.

Follow up with fact-seeking questions, such as "When did it last work?"; "Has it ever worked in this way?"; "Has any software changed recently?"; or "Any new hardware?"

By keeping your questions friendly and factual, you show the user that you won't accuse them or judge their actions. You also show them that you're there to help them. After the initial tension drops away, you'll often get more information, for instance, a recitation of something the user might have tried or changed. These clues can help lead to a quick resolution of the problem.

It's important to remember that you may know all about computer technology, but the user probably does not. This means they will often use vague and/or incorrect terms to describe a particular computer component or function. That's just the way it works, so don't bother to correct the user. Wherever possible, avoid using jargon, acronyms, or abbreviations specific to computers. They simply confuse the already upset user and can make it sound like you're talking down to them. Just ask direct, factual questions in a friendly tone using simple, non-jargon language to zero in on what the user was trying to accomplish and what happened when things went wrong. Point at the machine or go to a working PC to have the user show what went wrong or what he or she did or tried to do.

Although you don't want to overwhelm them, people do usually want to get a handle on what you are doing—although in a simplified way. Don't be afraid to use simple analogies or concepts to give them an idea what is happening. If you have the time (and the skills), use drawings, equipment, and other visual aids to make technical concepts more clear. If a customer is a "closet tech" and is really digging for answers—to the point it's affecting your ability to do your job—compliment their initiative and then direct them to outside training opportunities. Better yet, tell them where they can get a copy of this book!

Integrity

A computer tech must bring integrity to his or her job, just like any other service professional. Treat anything said to you as a personal confidence, not to be repeated to co-workers or bosses. Respect the privacy and property of the user.

You have a lot of power when you sit in front of someone's computer. You can readily read private e-mail, discover Web sites surfed, and more. With a click of the Start button, you can know the last five programs the user ran, including Word and Solitaire, and the last few documents he or she worked on. Don't do this. You really don't want to know! Plus, if you get caught violating a customer's privacy, you'll not only lose credibility and respect, you could lose your job.

Passwords are a big issue for techs. We have to reboot computers and access shares and other jobs that require passwords. The rule here is to avoid learning other folks' passwords at all costs. If you only need a password once, let the user type it in for you. If you anticipate accessing something multiple times (the more usual situation), ask the user to change his or her password temporarily.

It's funny, but people assume ownership of things they use at work. John in accounting doesn't call the computer he uses anything but "my PC." The phone on Susie's desk isn't the company phone, it's "Susie's phone." Regardless of the logic or illogic involved with this sense of ownership, a tech needs to respect that feeling. You'll never go wrong if you follow the Golden Rule: Treat people's things as you would have other people treat yours. Don't use or touch anything—keyboard, printer, laptop, monitor, mouse, phone, pen, paper, or cube toy—without first asking permission. Follow this rule at all times, even when the customer isn't looking!

Beyond basic manners, never assume that just because you are comfortable with friendly or casual behavior, the customer will be, too. Even an apparently casual user will still expect you to behave with professional decorum. On the flip side, don't allow a user to put you in an awkward or even potentially dangerous or illegal situation. Never socialize with customers while on the clock. Never do work outside the scope of your assigned duties without the prior approval of your supervisor (when possible in such cases, try to direct users to someone who *can* help them). You are not a babysitter—never volunteer to "watch the kids" while the customer leaves the job site, or tolerate a potentially unsafe situation if a customer isn't properly supervising a child. Concentrate on doing your job safely and efficiently, and maintain professional integrity.

Respect

The final key in communicating with the user revolves around respect. You don't do his or her job, but should respect that job and person as an essential cog in your organization. Communicate with users the way you would like them to communicate with you were the roles reversed.

Generally, IT folks are there to support the people doing a company's main business. You are there to serve their needs, and all things being equal, to do so at their convenience, not yours.

Don't assume the world stops the moment you walk in the door and that you may immediately interrupt their work to do yours. Although most customers are thrilled

and motivated to help you the moment you arrive, this may not always be the case. Ask the magic question, “May I start working on the problem now?” Give your customer a chance to wrap up, shut down, or do anything else necessary to finish his or her business and make it safe for you to do yours.

Engage the user with the standard rules of civil conversation. Take the time to listen. Don’t interrupt a story, but rather let it play out. You might hear something that leads to resolving the problem. Use an even, non-accusatory tone, and although it’s okay to try to explain a problem if the user asks, never condescend, and never argue. Remain positive in the face of adversity. Don’t get defensive if you can’t figure something out quickly and the user starts ragging on you. Remember that an angry customer isn’t really angry with you—he’s just frustrated—so don’t take his anger personally. Take it in stride; smile and assure him that computer troubleshooting sometimes takes awhile!

Avoid letting outside interruptions take your focus away from the user and his or her computer problem. Things that break your concentration slow down the troubleshooting process immensely. Plus, customers will feel insulted if you start chatting on your cell phone with your significant other about a movie date later that night when you’re supposed to be fixing their computer! You’re not being paid to socialize, so turn those cell phones and pagers to vibrate. That’s why the technogods created voicemail. Never take any call except one that is potentially urgent. If a call is potentially urgent, explain the urgency to the customer, step away, and deal with the call as quickly as possible.

If you discover that the user caused the problem, either through ignorance or by accident, don’t minimize the importance of the problem, but don’t be judgmental or insulting about the cause. We all screw up sometimes, and these kinds of mistakes are your job security! *You get paid because people make mistakes and machines break.* Chances are you’ll be back at that workstation six months or a year later, fixing something else. By becoming the user’s advocate and go-to person, you create a better work environment. If it’s a mistaken action that caused the problem, explain in a positive and supportive way how to do the task correctly and then have the user go through the process while you are there to reinforce what you said.

Assertiveness

In many cases, a PC problem is due to user error or neglect. As a technician, you must show users the error of their ways without creating anger or conflict. You do this by using assertive communication. *Assertive communication* is a technique that isn’t pushy or bossy, but it’s also not the language of a pushover. Assertive communication first requires you show the other person that you understand and appreciate the importance of his or her feelings. Use statements such as “I know how frustrating it feels to lose data” or “I understand how infuriating it is when the network goes out and you can’t get your job done.” Statements like these cool off the situation and let the customer know you are on his or her side.

The second part of assertive communication is making sure the problem is clearly stated—without accusing the user directly: “Not keeping up with defragmenting your

hard drive slows it down,” or “Help me understand how the network cable keeps getting unplugged during your lunch hour.” Lastly, tell the user what you need from them to prevent this error in the future. “Please call me whenever you hear that buzzing sound,” or “Please check the company’s approved software list before installing anything.” Always use “I” and “me,” and never make judgments. “I can’t promise the keyboard will work well if it’s always getting dirty” is much better than “Stop eating cookies over the keyboard, you slob!”

Troubleshooting Methodology

Following a sound troubleshooting methodology helps you figure out and fix problems quickly. But because troubleshooting is as much art as science, I can’t give you a step-by-step list of things to try in a particular order. You’ve got to be flexible.

First, make sure you have the proper tools for the job. Second, back up everything important before doing repair work. And third, analyze the problem, test your solution, and complete your troubleshooting.

Tech Toolkit

Way back in Chapter 2, “The Visible PC,” you learned the basic tools in every well-prepared tech’s toolkit (see Figure 24-11): a Phillips-head and straight-slot screwdriver along with a few other useful tools, such as a Torx wrench and a pair of tweezers. You also should carry some computer components.

Figure 24-11
Typical technician
toolkit



Always carry several *field replaceable units (FRUs)*—a fancy way to say *spare parts*—when going to a job site or workstation. Having several known good components on hand enables you to swap out a potentially bad piece of hardware to see if that’s the problem. Different technicians will have different FRUs. A printer specialist might carry a number of different fusers, for example. Your employer will also have a big effect on what is an FRU and what is not. I generally carry a couple of RAM sticks (DDR and DDR2), a PCI video card, a NIC, and a 300-watt power supply.

Backup

In many troubleshooting situations, it's important to back up critical files before making changes to a system. Partly this is a matter of proper ongoing maintenance. If you're in charge of a set of machines for your company, for example, make sure they're set to back up critical files automatically on a regular basis.



EXAM TIP The CompTIA A+ certification exams assume that all techs should back up systems *every time* before working on them, even though that's not how it works in the real world.

If you run into a partially functional system, where you might have to reinstall the OS but can access the hard drive, then you should definitely back up essential data, such as e-mail, browser favorites, important documents, and any data not stored on a regularly backed-up server. Because you can always boot to a copy of Windows and go to the recovery console, you should never lose essential data, barring full-blown hard drive death.



NOTE Dead hard drives retain their data, so you can recover it—if you're willing to pay a lot of money. Having a good backup in place makes a lot more economic sense!

Steps

Troubleshooting a computer problem can create a great day for a computer tech—if he or she goes about solving it systematically and logically. Too many techs get into a rut, thinking that when symptom A occurs, the problem must be caused by problem A and require solution A. They might even be right nine times out of ten, but if you think this way, you're in trouble when the problem is really Z and requires a completely different solution. What do you do when the fix doesn't work and you're sure it's got to be problem A? Your customer won't be happy, you'll end up frustrated or embarrassed, and you still won't have the problem fixed. Follow the right methodology: analyze, test, and complete your task.

Analyze

Everything starts with analyzing the problem. Because you know how the process should work, when you run into a computer problem, the first question you should ask is, "What can it be?" Don't limit yourself in the initial analysis, but examine all possibilities.

Make sure you understand the nature of the problem first. A big portion of the initial fact-finding involves talking to the user with your newly-honed customer communication skills. Practice asking friendly, factual questions to get the results you want.

When you run into big problems, such as a completely dead PC, or some weird issue that involves networking as well as a local machine, break the problem down into smaller parts. For a dead PC, for example, organize your inquiry into categories, like this:

- **Power** Check the AC connections and power switches on the power supply and UPS or surge suppressor.
- **Connectivity** Check inside the box to make sure nothing is unplugged.
- **CMOS** If you've got power (you see an LED lit up, for example), then check CMOS to see if the hard drive shows up.
- **Operating system** If you have some life in the PC, but get no boot option, then try booting to the CD and running the recovery console. Check to see if you can access the hard drive. If you have a boot option, then try booting to Safe Mode or Last Known Good Configuration.

By breaking the problem into discrete chunks, you make a big problem more manageable. You can determine whether the problem is caused by a hardware failure, connection problem, or perhaps an issue with the operating system or other software.

Test

Once you determine what might have caused the problem, test your theories. The testing procedure follows a simple set of rules:

First, check the easy stuff. Is the failed device plugged in? Is it turned on? Does the printer have paper?

Second, use a process of elimination to home in on the problem. Use a notepad to write down what you've tried and what effect that effort had. Do one thing at a time. If you go into CMOS and make a whole series of changes to various settings, for example, and then computer works, how do you know what was wrong? You can't know when you make multiple changes. Worse yet, you risk breaking something else while fixing the first problem!

Third, if hardware seems to be the problem, swap out the suspect parts with known good parts from your stash of FRUs. With a dead or dying piece of hardware, you can generally get almost immediate results. A fried stick of RAM, for example, can create a dead PC. Pop in a good stick, and you'll have a functional machine.



NOTE The power supply provides the only exception to the instant gratification rule of part swapping. If you have a wonky computer that has intermittent problems, swap out the power supply and leave your FRU in the computer for a half day or longer. You need the user to road test the PC and report if he or she has the same problems.

Complete

Once you finish the testing phase, you can complete the troubleshooting process with four more steps. First, evaluate the results of your actions. Run the system through its paces; don't just get it working and walk away. Make certain the user can accomplish his or her primary tasks before you consider a job complete.

If you can't get the computer or peripheral working in a fairly short period of time, take the second step and call for help. Because you've taken notes on the symptoms and each troubleshooting step you've tried, you can very quickly get a more senior tech up to speed on the problem and get suggestions for where you might go next.

Third, clean up the work environment. If you installed a drive, for example, do the right thing and tie off the ribbon cables. If you see a mess of cables coming out of the back of the PC or running across the floor, take a moment to tie them off. Good cable management does more than leave your clients with a nice looking computer and workstation area, it also helps prevent accidents.

Finally, document your results. Many companies have specific forms for you to use to describe the problem and its resolution. If not, then make some for yourself and other techs in your workplace. Documenting problems helps you track the troubleshooting history of a machine over time, enabling you to make longer-term determinations about retiring it or changing out more parts. If you and fellow techs fix a specific problem with Mary's machine several times, for example, you might decide to swap out her whole system rather than fix it a fourth time.

Documenting helps fellow techs if they have to follow up on a task you didn't finish or troubleshoot a machine that you've worked on previously. The reverse is also true. If you get a call about Frank's computer, for example, and check the records to find other service calls on his computer, you might find that the fix for a particular problem is already documented. This is especially true for user-generated problems. Having documentation of what you did also means you don't have to rely on your memory when your co-worker asks what you did to fix the weird problem with Jane's computer a year ago!

Documenting also comes into play when you or a user has an accident on site. If your colleague Joe dropped a monitor on his foot and broke both the monitor and his foot, for example, you need to fill out an incident report, just like you would with any kind of accident—electrical, chemical, or physical. An incident report should detail what happened and where it happened. This helps your supervisors take the appropriate actions quickly and efficiently.

Chapter Review Questions

1. While troubleshooting a fairly routine printing problem, the customer explains in great detail precisely what he was trying to do, what happened when he tried to print, and what he attempted as a fix for the problem. At what point should you interrupt him?

- A. After he describes the first problem
 - B. As soon as you understand the problem
 - C. As soon as you have a solution
 - D. Never
2. While manning the help desk, you get a call from a distraught user who says she has a blank screen. What would be a useful follow-up question? (Select two.)
- A. Is the computer turned on?
 - B. Is the monitor turned on?
 - C. Did you reboot?
 - D. What did you do?
3. While manning the help desk, you get a call from Sharon in accounting. She's lost a file that she knows she saved to her hard drive. Which of the following statements would direct Sharon to open her My Documents folder in the most efficient and professional manner?
- A. Sharon, check My Documents.
 - B. Sharon, a lot of programs save files to a default folder, often to a folder called My Documents. Let's look there first. Click the Start button and move the mouse until the cursor hovers over My Documents. Then click the left mouse button and tell me what you see when My Documents opens.
 - C. Probably just defaulted to My Docs. Why don't you open Excel or whatever program you used to make the file, and then open a document and point it to My Documents.
 - D. Look Sharon, I know you're clueless when it comes to computers, but how could somebody lose a file? Just open up My Documents, and look there for the file.
4. What tool should be in every technician's toolkit?
- A. Pliers
 - B. Hammer
 - C. Straight-slot screwdriver
 - D. Phillips-head screwdriver
5. When is it appropriate to yell at a user?
- A. When he screws up the second time
 - B. When he interrupts your troubleshooting

- C. When he screws up the fifth time
 - D. Never
6. Al in marketing calls for tech support, complaining that he has a dead PC. What is a good first question or questions to begin troubleshooting the problem?
- A. Did the computer ever work?
 - B. When did the computer last work?
 - C. When you say “dead,” what do you mean? What happens when you press the power button?
 - D. What did you do?
7. While manning the help desk, you get a call from Bryce in Sales complaining that he can’t print and every time he clicks on the network shared drive, his computer freezes. He says he thinks it’s his hard driver. What would be a good follow-up question or statement?
- A. Bryce, you’re an idiot. Don’t touch anything. I’ll be there in five minutes.
 - B. Okay, let’s take this one step at a time. You seem to have two problems, one with printing and the second with the network shared drive, right?
 - C. First, it’s not a hard driver, but a hard drive. It doesn’t have anything to do with the network share or printing, so that’s just not right.
 - D. When could you last print?
8. When troubleshooting a software problem on Phoebe’s computer and listening to her describe the problem, your beeper goes off. It’s your boss. Which of the following is the most appropriate action for you to take?
- A. Excuse yourself, walk out of the cube, and use a cell phone to call your boss.
 - B. Pick up Phoebe’s phone and dial your boss’ number.
 - C. Wait until Phoebe finishes her description and then ask to use her phone to call your boss.
 - D. Wait until Phoebe finishes her description and run through any simple fixes; then explain that you need to call your boss on your cell phone.
9. You’ve just installed new printer drivers into Roland’s computer for the big networked laser printer. What should you do to complete the assignment?
- A. Document that you installed new printer drivers.
 - B. Tell Roland to print a test page.
 - C. Print a test page and go to the printer to verify the results.
 - D. Print a test page and go to the printer to verify the results. Document that you installed new printer drivers successfully.

10. What's an FRU?
- A. Foreign replacement unit—a cheaper part to replace an expensive American-made part
 - B. Field replacement unit—a known good computer part to swap with a suspect part as part of the troubleshooting process
 - C. Free repair unit—a gimmick used by some companies to provide free tech support for the first time customer
 - D. Short for FRU Linux, a Linux distribution

Answers

- 1. D. Don't interrupt customers. Let them tell you what's happening and then ask questions.
- 2. A, B. Go for the simple answer first. When faced with a blank screen, check to see if the computer and the monitor are on.
- 3. B. Walking customers through the path to a fix, using simple, nontechnical words, is the best way to accomplish tasks over the phone.
- 4. D. Every tech's toolkit should have a Phillips-head screwdriver, at the very least.
- 5. D. Don't get angry or yell at clients!
- 6. C. Asking for clarification is a good first step. This is a tough call because A and B are obviously useful questions that will probably be the second and third questions you ask!
- 7. B. Similar to question 6, asking for clarification and stating the problem as you heard it are good follow-up options.
- 8. D. Focus on the customer and don't use her things!
- 9. D. Always verify the results once you finish troubleshooting. Then document what you accomplished.
- 10. B. A field replacement unit is a known good computer part.