Boson®

IBNT Curriculum

Introduction to Basic Networking Technologies

Introduction to Basic Networking Technologies

IBNT Curriculum

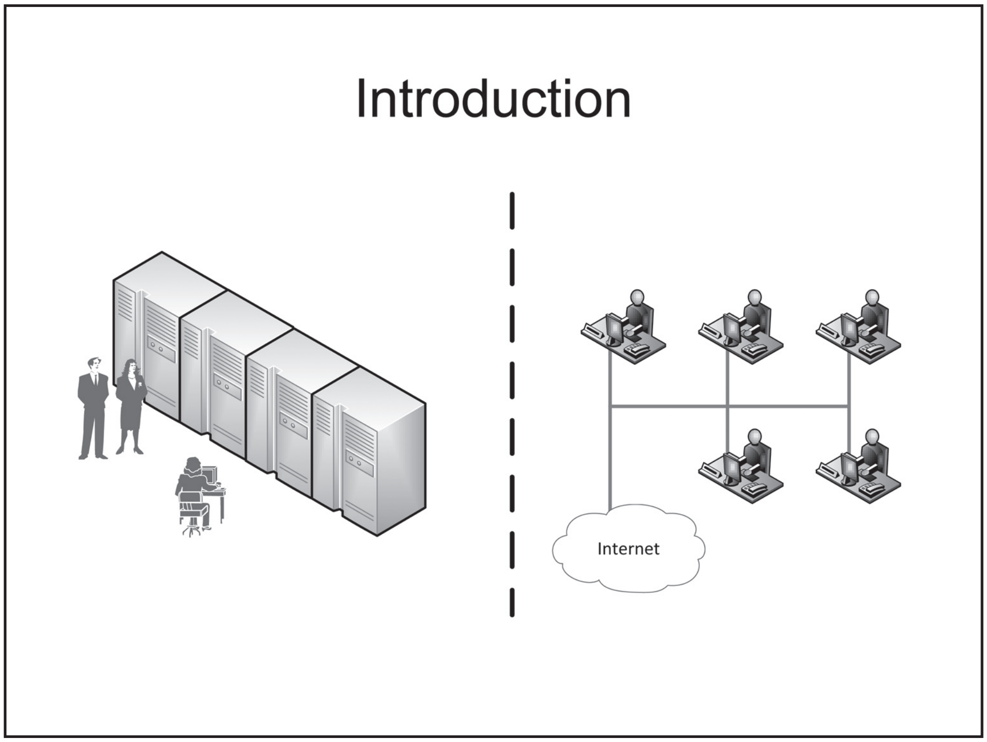
LM20160404/BV1.0

Boson®

25 Century Blvd., Ste. 500, Nashville, TN 37214 | Boson.com

The labs referenced in this book have been printed in the Boson IBNT Lab Guide, which is included with the purchase of the curriculum.

Introduction



Introduction

Introduction

A mere few decades ago, computers were bulky, slow, and so prohibitively expensive that only larger businesses could afford them. They were used mostly for scientific research and calculations. In those times, the most advanced piece of equipment in a typical office was a fax machine. Over time, computers became smaller, faster, and more affordable. As a result, their main purpose shifted from crunching numbers to processing, storing, and manipulating data—primarily business data, but personal data as well. Nowadays, it is hard to imagine an office without a computer on every desk and a smartphone in everyone's pocket. There are so many types of computing equipment now that the word computer can no longer cover them all; it has become customary to use a generic, all-inclusive word device instead. Desktops, laptops, netbooks, tablets, smartphones as well as other related types of equipment are all referred to as devices now.

Society today produces and consumes astronomic amounts of information. Information does not stay at the point of its origin; it flows from device to device in emails, text messages, telephone conversations, streams of videos, and webpages. This brings us to the importance of communications and the technologies that make them possible.

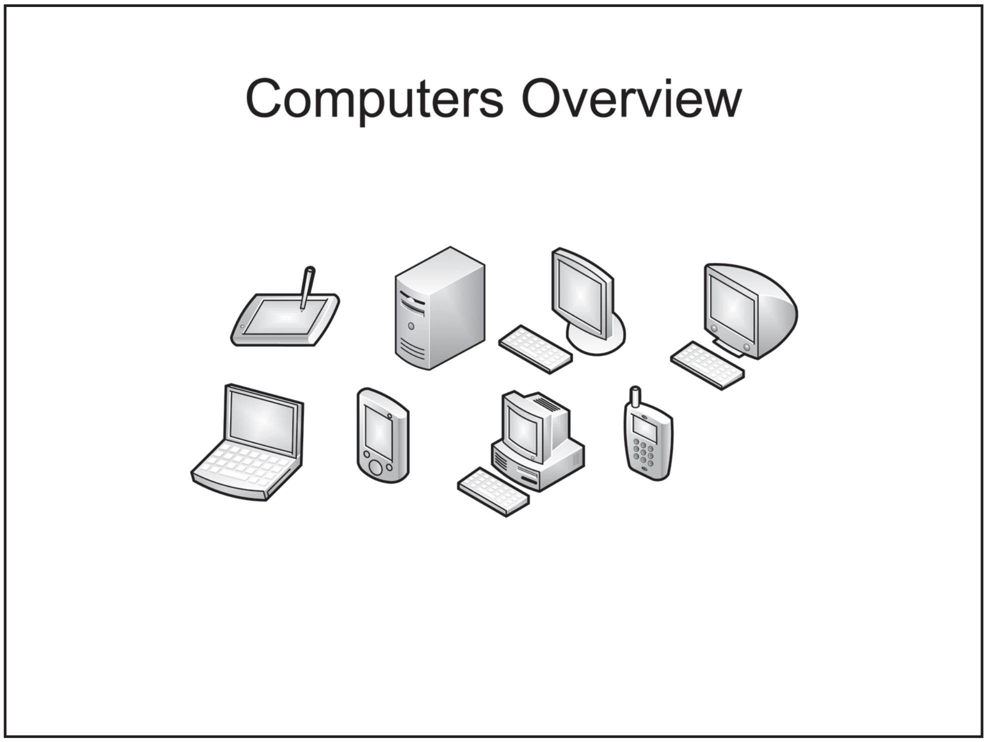
In the olden days, each computer was usually an island to itself. Operators loaded programs, input data prepared on punched cards, and ran the programs. Computers swallowed the punched cards, crunched some numbers, and produced printouts for analysts to ponder on. Any long-term data accumulated during such sessions was stored on disks or tapes, and the tapes were occasionally moved to an archive as data on them was no longer needed. That was about all that computers did: input data, process that data, and print out the results. Each computer performed its own designated functions, and very few of those functions involved computers' talking to one another. Telephone and telegraph networks existed long before computers, but using that infrastructure for computer communications was not very common. Those networks were not designed to transmit data, and the first two or three generations of computers were not designed with communications as a priority in mind.

Nowadays, the situation is almost completely reversed: very few computers are used as merely local productivity devices, and nearly all computers are connected to one type of network or another and have access to the Internet, the network of all networks in the world. Now most of what computers do involves communications with other computers. For example, whenever you browse the Internet, your computer talks to different web servers located anywhere in the world. When you open a webpage in a web browser, your computer downloads that page from a web server that hosts the website you are visiting. When you send or receive an email message on your home computer, your computer talks to a mail server hosted by your Internet service provider (ISP), which in turn talks to other mail servers hosted by other ISPs, and those mail servers talk to the computers of the people who receive your emails or send their emails to you. Whenever you upload your photos or videos to social media websites, such as Facebook, Twitter, or YouTube, or when you look at photos or watch videos posted by others, your computer talks to different servers that can be physically located in a building across the street or in another country halfway around the globe.

Thus computer communications are of great importance in this day and age. For communications to be possible, merely plugging a network cable into your computer is not enough. The other end of that cable must be connected to certain equipment, that equipment must be connected to some other equipment, and that equipment must be connected to yet some other equipment, and so on. If you want to become a network specialist, you need to learn how networks are designed, how they are built, and how they work. This tutorial is a good place for you to start that journey.

Module 1

Computers



Computers Overview

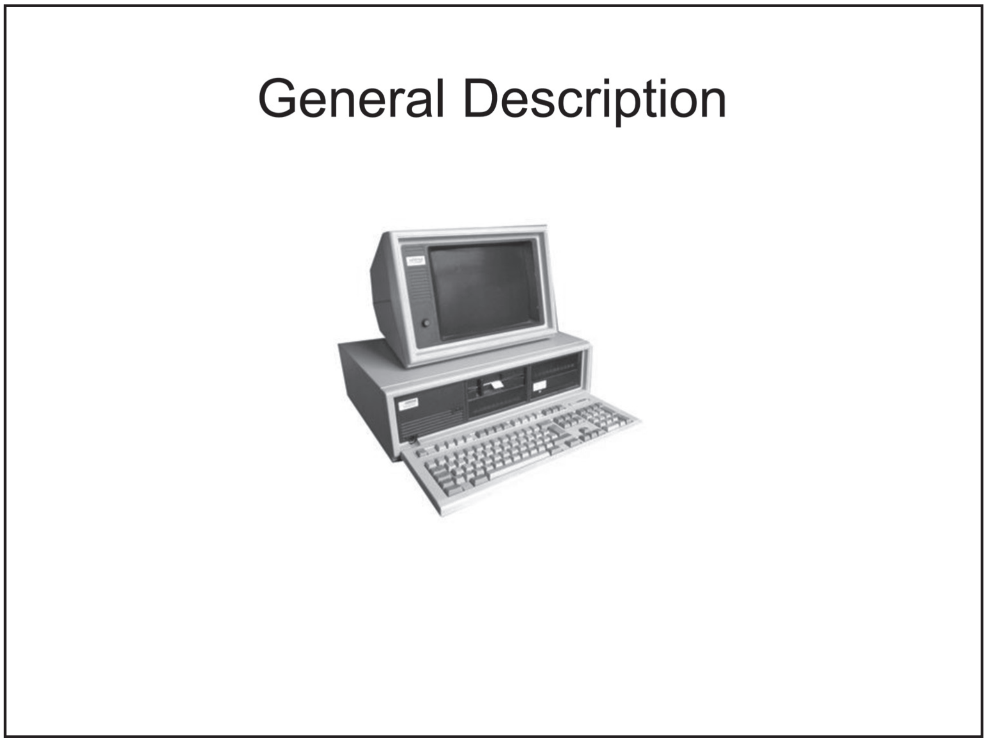
Overview

This tutorial is an introduction to Cisco networking devices. It is intended for the people who are new to information technologies (IT) in general and to telecommunications in particular. The main purpose of this tutorial is to provide a low-threshold entry point into the world of computers, telecommunications, and Cisco devices. Cisco is a major manufacturer of networking equipment. Networking equipment is the foundation of networking. But networks exist not for the sake of networking equipment; their main purpose is to connect computers so that computers can exchange information with one another. To be able to understand how networks work, you first need to understand computers: what they are made of, what they do, and how they do it.

Objectives

After completing this module, you should have basic knowledge of computers and should be able to perform the following tasks:

- Explain what computers can do.
- Name main computer components.
- Explain the difference between computer software and firmware.
- Explain the difference between operating systems (OSs) and applications.
- Describe what happens when a computer is turned on.
- Describe how OSs are installed.
- Describe how applications are installed and run.
- Explain why computers communicate with one another.



General Description

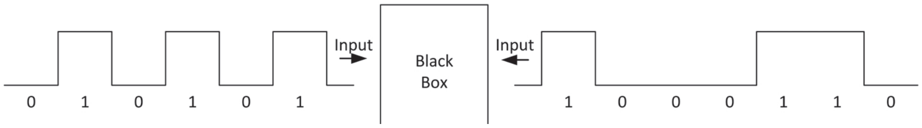
General Description

A computer is an electronic device that can receive information from external sources, manipulate that information, and provide the resulting information to human users, other computers, or other types of devices. There are many types of computers, but they are all designed based on the same main principles. This module primarily focuses on what is commonly known as a personal computer (PC) or a desktop computer and provides a simplified description of its main components as well as their purposes and functionalities.

The term *desktop* in relation to a computer originates from the fact that the first PCs were one-unit devices that were supposed to sit on users' desktops. A user could input information by using a keyboard and could view the output on a display, also known as a monitor or simply a screen. A separate device, called a printer, could be plugged into the computer box and could produce a hard copy of the output information. Later, the display became a separate device too, connected to the main box by a cable. When that happened, there were no longer any compelling reasons for keeping the PC box on the desktop. It was often more convenient to stand the PC box vertically on its side and keep it *under* the desk. But the term *desktop* is still widely used to refer to a nonportable PC intended for office or home use.

A modern desktop PC consists of a box—the technical term for it is a *case*—and several external devices, or peripherals, connected to it by cables or through wireless links. The most common peripherals are a keyboard, a mouse, a monitor, a printer, and speakers.

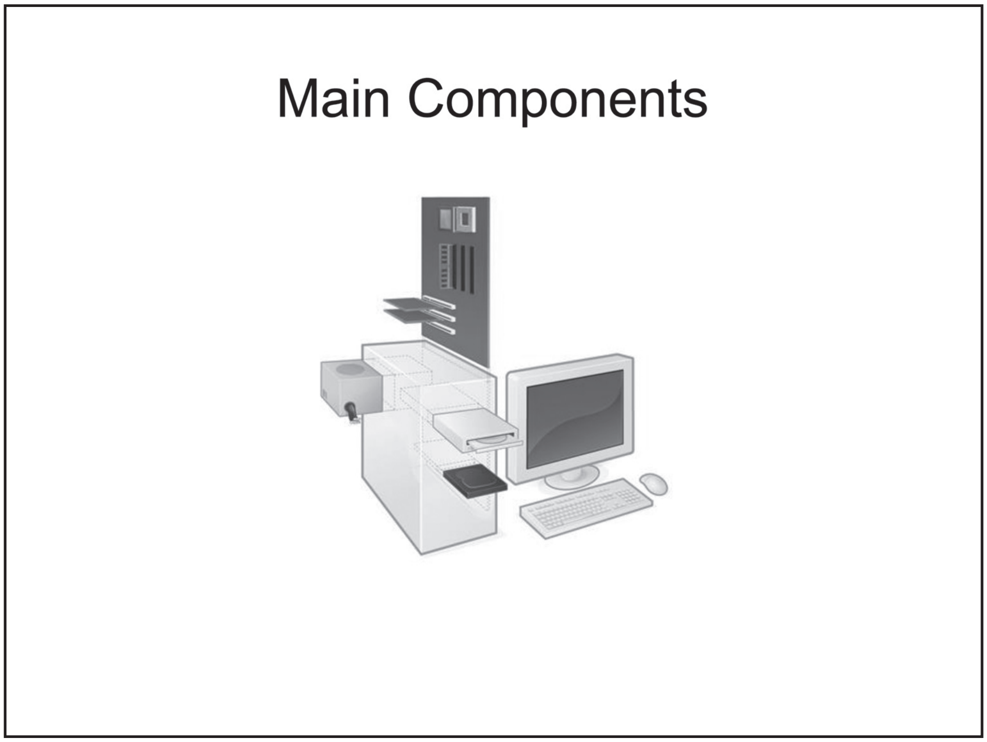
A computer is not a live object; it is merely a collection of electronic equipment without any inherent intelligence of its own. All a computer can do is run programs created and input into it by humans. The main job of a PC is to process information. A computer does not understand the concept of information; the only thing it understands is electrical signals. Imagine a black box—an abstract concept representing a device with known functionalities and an unknown internal structure. Suppose there is one electrical connection going into the black box and one coming out of it. If you apply a certain level of voltage to the input connection for a specific time interval, the box treats it as a numerical value of 1. If there is no voltage for the duration of the same time interval, the box treats it as a numerical value of 0. In this manner, you can input a string of 1s and 0s into the device. Based on its internal logic, the device will react by producing a different string of 1s and 0s on the output connection.



By using only two digits, 1 and 0, you can encode other numbers as well as alphabetical and special characters. An eight-digit string of 1s and 0s can produce 256 combinations, which is sufficient to represent all alphanumeric characters in almost any language. The following exhibit presents a few examples of the encoding known as ASCII:

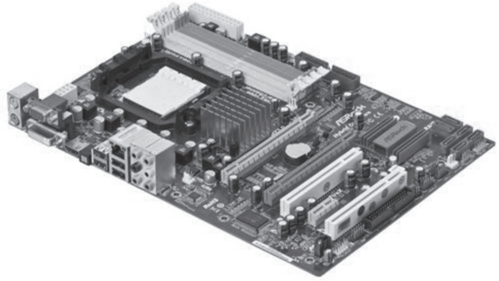
A	01000001
B	01000010
a	01100001
b	01100010
0	00110000
1	00110001
2	00110010

The use of 1s and 0s to encode other numbers is known as binary notation and will be covered in more detail in Module 3: IP Addressing in this tutorial. Computers accept information in the form of binary code, perform various manipulations on the code, and produce the resulting information also as binary code. Input devices such as keyboards digitize human-readable information (that is, convert letters and numbers into 1s and 0s), and output devices perform the opposite transformation. For example, a monitor converts binary code into letters and numbers and presents them in human-readable form on the screen.



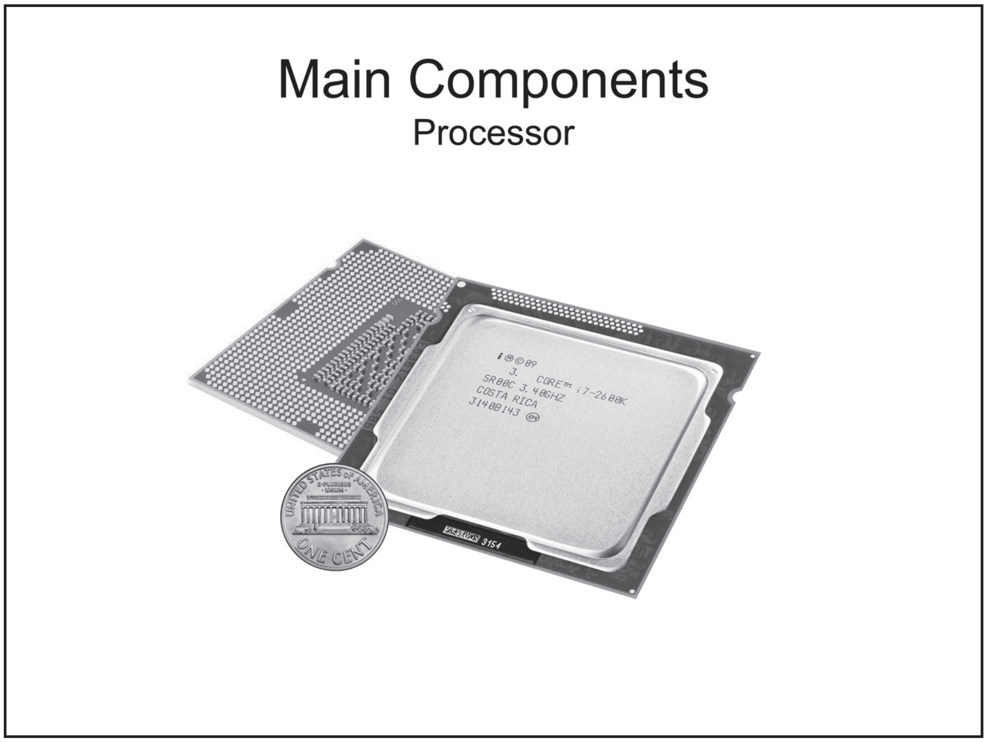
Main Components

Main Components



Motherboard

The main structural component that links all the other components of a PC to one another is a motherboard. It is a circuit board that contains one or more integrated microchips and various slots, sockets, and connectors for additional equipment. The integrated microchips are commonly referred to as chipsets. They contain supporting circuitry for other devices. Put simply, a motherboard is a PC component into which all the other components are plugged.



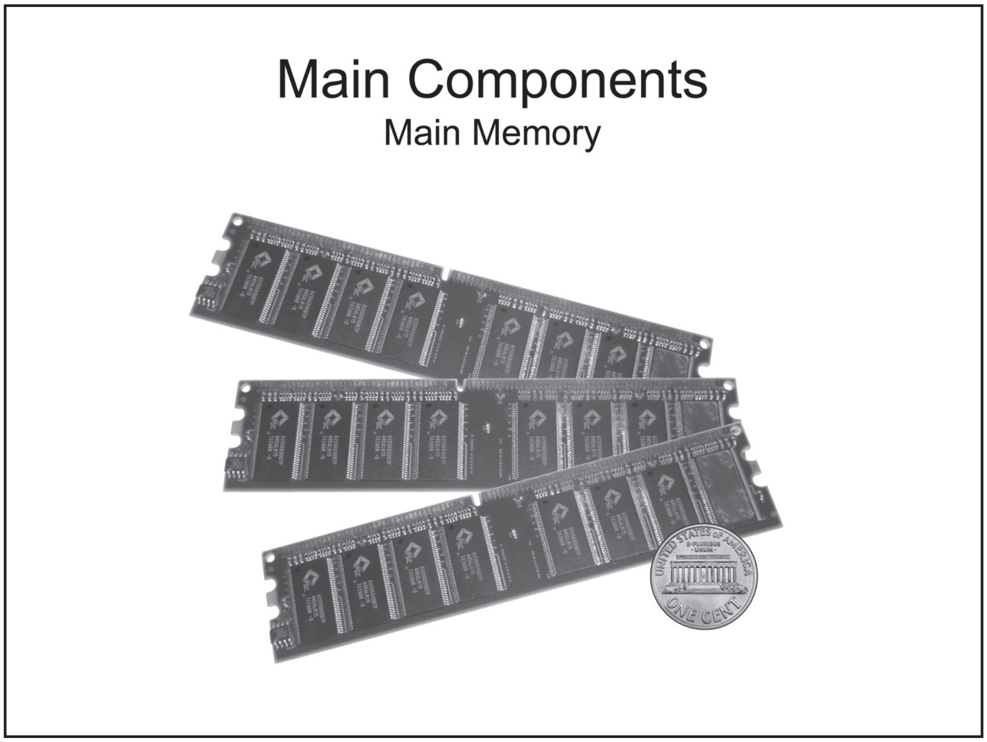
Main Components

Processor

Processor

The most important component of a PC is a processor. The proper technical term for it is a *central processing unit* (CPU). In the past, computers were big; a single computer consisted of several cabinets filled with dozens or even hundreds of circuit boards. Such a computer would often take a large portion of a room, and a CPU would occupy one or more of those cabinets. Presently, a PC is a small case that easily fits under your desk and a CPU is a microchip, a 1- to 2-inch (3- to 5-cm) square flat piece of ceramic with all of its circuitry hidden inside it. This microchip is commonly referred to as a physical CPU and can contain two, four, six, or even more CPU cores, which are referred to as logical CPUs. They share their common workload and in nontechnical discussions can be treated as a single CPU.

The main job of a CPU is to execute commands. A CPU receives a binary string that consists of a specific number of 1s and 0s, interprets it as a command, and executes it. For example, a command can add, subtract, multiply, or divide numbers or can check the result of the previous command and then call a different command, depending on whether the result is positive or negative. Similarly to how any alphanumeric character can be represented by a combination of 1s and 0s, complex calculations and sophisticated logic can be implemented by using simple commands.



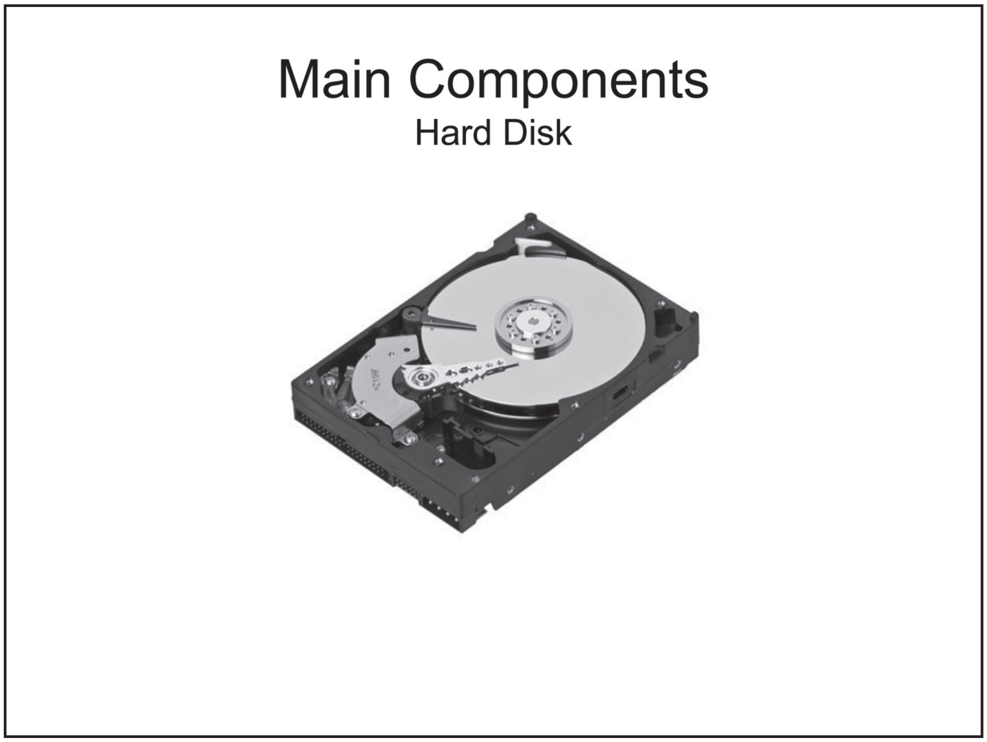
Main Components

Main Memory

Main Memory

Memory is a device that can store information. Conceptually, it is similar to a rack of shelves in a storage room: you put something on a shelf, and it stays there until you need to use it. A PC contains several types of memory. The main memory is what people usually have in mind when they talk about computer memory. The technical term for the main memory is *random-access memory* (RAM). Physically, RAM consists of modules, plastic strips of about 4 to 5 inches (10 to 12 cm) long and about 1 inch (2 to 3 cm) wide with several microchips mounted on each strip. RAM is also often called volatile or dynamic memory because it stores information only while the PC is powered up. If you turn off power on your computer or press the Reset button, the contents of RAM are gone.

The term *RAM* indicates that information stored in the main memory can be accessed in an arbitrary order. Typically, most modern PCs have a 64-bit architecture, which means that they store, transfer, and process information in blocks of 64 binary digits. Bit is the smallest unit of measure in regards to the quantity of information; one binary digit, 1 or 0, constitutes one bit of information. Each 64-bit block, or cell, in RAM has its own address, and a CPU can read from or write to any specific cell without having to access any preceding cells. RAM is the place where a CPU reads commands from and where it writes the results to.



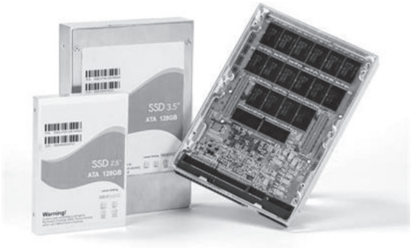
Main Components

Hard Disk

Hard Disk

A hard disk, or simply a disk, is a nonvolatile-storage device. Information can be written to or read from a disk only while the disk is powered up. However, once written to a disk, information does not disappear when the disk is turned off. The information stays on the disk for as long as the disk remains functional or until the information is erased or overwritten. The term *disk* is commonly used as shorthand; the proper term for this type of device is a *hard disk drive* (HDD). An HDD can include one or more metallic platters with a common spindle. The platters are covered with a magnetic coating. HDDs use the same physical principles as tape recorders, except that access to a tape is sequential, whereas access to a disk is almost direct. An HDD has an arm similar to the one in turntables that are used for playing vinyl records. The arm is fitted with read/write heads, one head for each surface of each platter. The disks spin, and the heads move radially. Information is written on each surface in concentric circles, or tracks. All the disks in the same drive spin only together, and all the heads move radially also only together. While the disks are making one revolution, information can be written to or read from the same track on each surface. A set of the tracks with the same number is called a cylinder. Access to any specific cylinder as well as to any track in the cylinder is direct, but within each track, access is sequential.

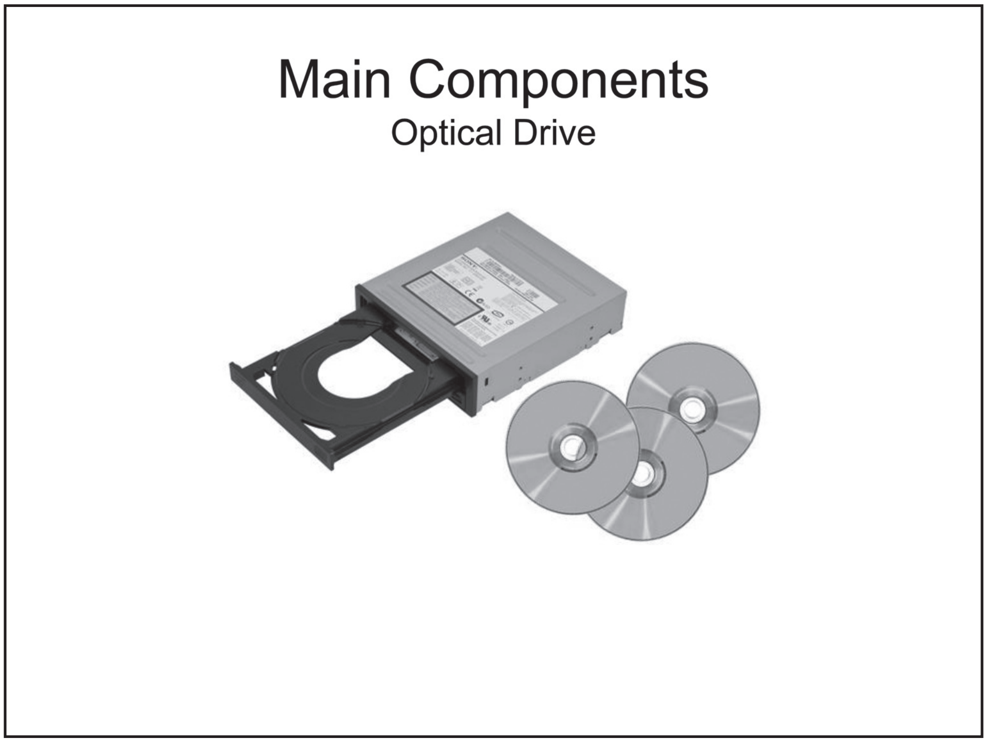
Another type of nonvolatile-storage device is known as a *solid-state drive* (SSD). An SSD is made of memory microchips and does not have any moving parts, but it interacts with a PC similarly to an HDD. Therefore, SSDs are considered as a variety of disk.



SSDs are faster than HDDs but are more expensive and have lower storage capacity. A widely known class of devices that are very similar to SSDs is flash thumb drives. *Flash* refers to the fact that this type of memory is nonvolatile (flash once, or write once, and the data stays for the duration of the device's life span or until it is erased or overwritten), and *thumb* refers to the size of an individual device.

Although both SSDs and flash thumb drives use the same technology, there are two main differences between them. SSDs are more like HDDs; both are connected through the same type of interface, and both are usually installed inside a PC case. Flash thumb drives are external devices; they are connected to a PC through an interface known as a *universal serial bus* (USB). Hence their abbreviated name, UFD, which stands for *USB flash drive*. The second difference comes from the fact that flash memory can sustain only a limited number of write cycles throughout its life span. SSDs are intended as an alternative to HDDs; therefore, they are designed to support intensive write operations that are characteristic for HDDs. UFDs, on the other hand, are intended for only occasional use; consequently, they do not have to be as reliable and as expensive as SSDs.

To keep things relatively simple, this tutorial will refer to the three aforementioned types of nonvolatile-storage devices—HDDs, SSDs, and UFDs—as disks. Disks are used for two main purposes: as long-term storage for applications and data and as an extension of RAM. When RAM starts running out of free space, some data or code that has not been recently used is temporarily copied to a disk and is loaded back into RAM when it is needed. This combination of RAM and temporary disk storage is called virtual memory. The use of disks as long-term storage is described in the How Software Is Installed and Run subsection.



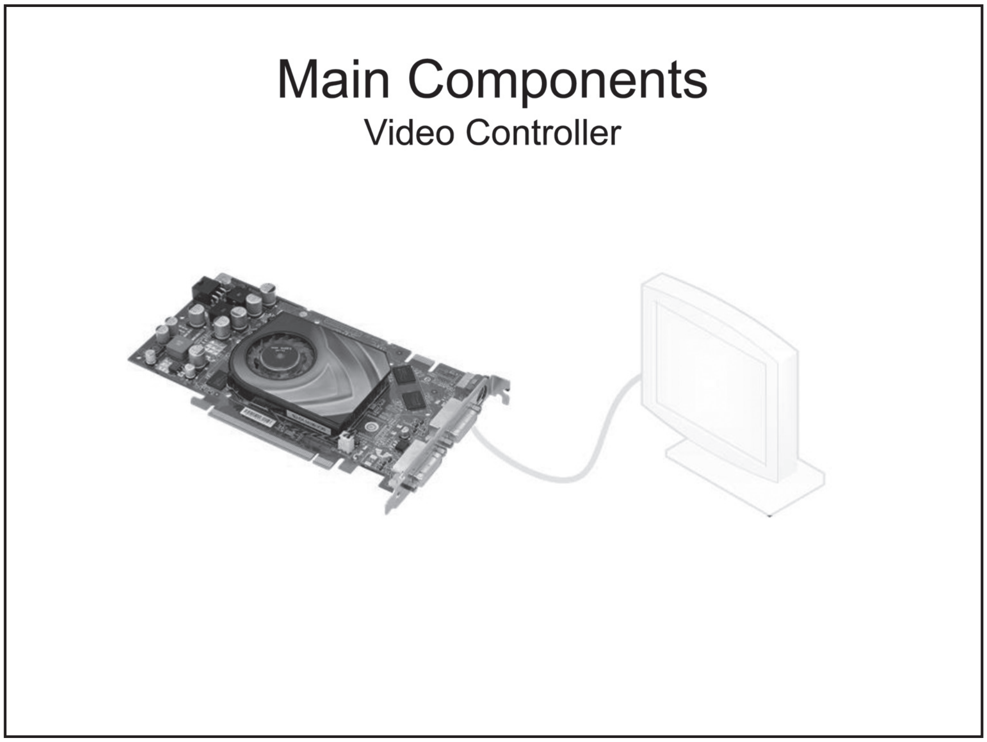
Main Components

Optical Drive

Optical Drive

An optical drive is a device that can read data from and, possibly, write data to removable optical media, which are collectively referred to as optical discs. Note the spelling, *disc*, as opposed to the word *disk*, which is used to refer to HDDs, SSDs, and UFDs. Older discs were called compact discs (CDs), a newer generation is called digital video discs or digital versatile discs (DVDs), and the most recent generation is known as Blu-ray discs (BDs or BRDs). Typically, optical discs are used to permanently store information. The information is written to a disc only once and cannot be changed. Although there are also erasable, or rewritable, optical discs, which can be written to multiple times, they are not an adequate substitute for HDDs or other disks. Rewritable discs are much slower, less reliable, and have lower storage capacity than disks.

Typical uses for optical media are software distribution, data archival, and entertainment. For example, when you purchase an OS, such as Windows, or a program suite, such as Microsoft Office, it is often provided to you on a DVD. If you want to free up some space on your PC's hard disk, you can burn your photos or videos to a DVD, put it on a shelf, and delete the photos and videos from the disk. DVDs are also widely used to store movies, and CDs are still used to store music or other audio recordings. Movies and music can be played on computers or on special appliances, known as DVD players and CD players, respectively.



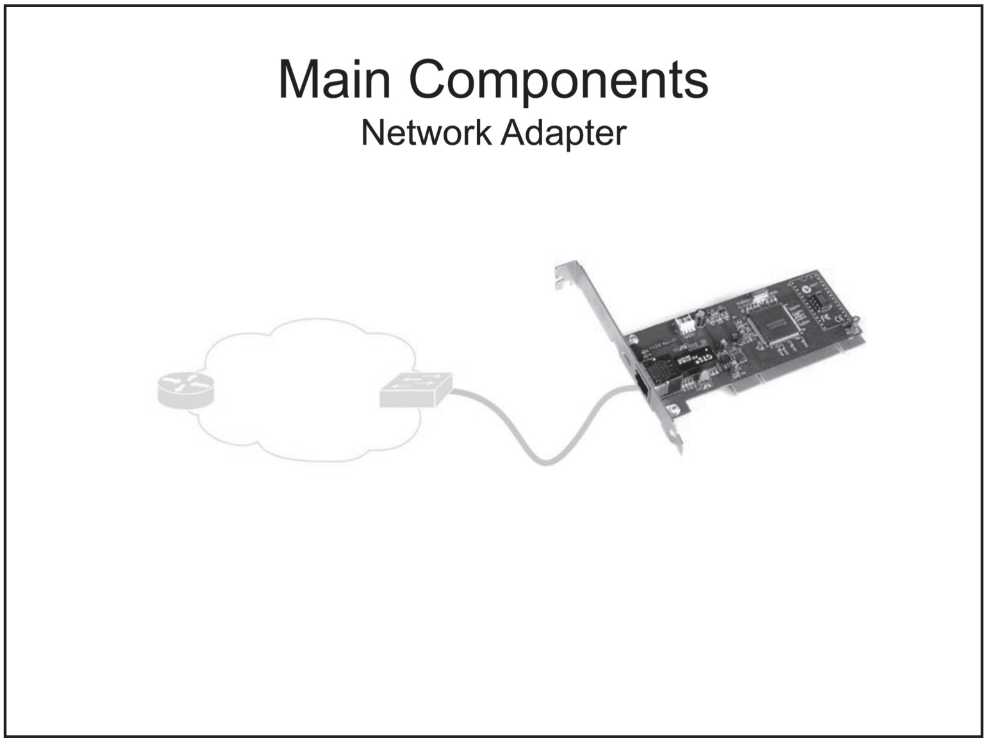
Main Components

Video Controller

Video Controller

A display, also known as a monitor, is a device similar to a television (TV) set. It accepts electrical signals from a PC and converts them into graphic images on its screen. Although modern displays are quite sophisticated devices, conceptually they are nothing more than simply monitors. All they can do is show a picture on the screen. A video controller, also known as a video adapter or a display adapter, is a device that generates the signals that the display renders into graphic images and presents on the screen. A video controller can be implemented as a microchip soldered onto the motherboard or as a separate circuit board that is inserted into one of the special expansion slots on the motherboard.

A picture on the screen does not happen magically out of thin air; it is generated by the signals sent by the video controller through a cable to the display. The video controller produces those signals in response to the commands executed by the CPU. In other words, every single dot on the screen is controlled by programs running on the PC. The technical term for those dots is *pixels*. A modern display's screen consists of several million pixels. Each pixel is controlled separately and can be turned on or off. When a pixel is off, it emits no light and appears as a black dot. When a pixel is on, its color and brightness depend on the binary code that is part of the signal coming from the video controller.

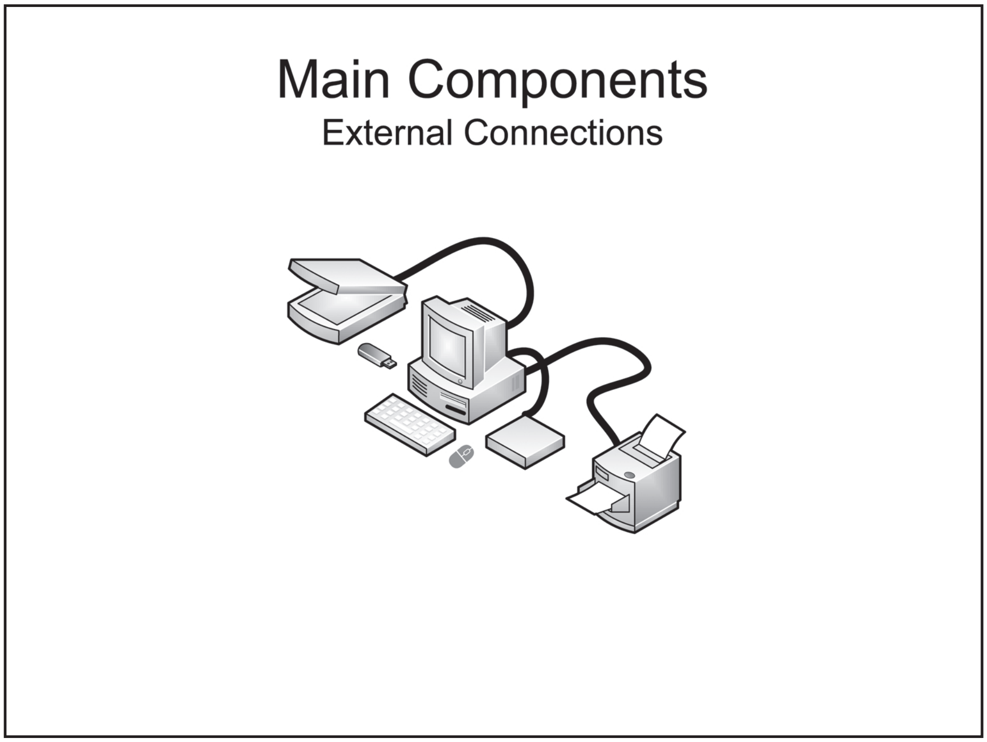


Main Components

Network Adapter

Network Adapter

A network adapter, also known as a network interface card (NIC), is a device that connects a PC to a network. A NIC can be integrated into a motherboard, or it can be implemented as a separate circuit board that is inserted into one of the special expansion slots on the motherboard. In response to commands executed by the CPU, a NIC can generate signals and send them out to the network. When the NIC on another computer receives those signals, it transforms them into a form suitable for internal use by the PC and writes the resulting data into RAM, where it is processed by a program running on the PC and for which the data is intended. The processing can involve changing the data, storing it on a disk, or sending it elsewhere.

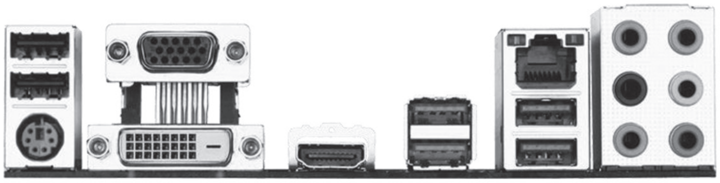


Main Components

External Connections

External Connections

A PC’s core equipment includes a motherboard, CPU, disk, video controller, and NIC, which are usually located inside a PC case and therefore are characterized as internal devices. In addition, a PC can include external devices, also known as peripherals, which are located outside the PC case and which are usually attached to the case with cables. Alternatively, some external devices can be connected to a PC by using wireless technologies. Some devices can be implemented as either internal or external. A device being internal or external is often just a matter of convenience rather than an inherent characteristic of specific types of devices. A motherboard can include several types of interfaces for communicating with external locally attached devices. Each interface includes a controller, one or more physical connectors, and possibly other supporting circuitry. The physical connectors are also often referred to as ports.



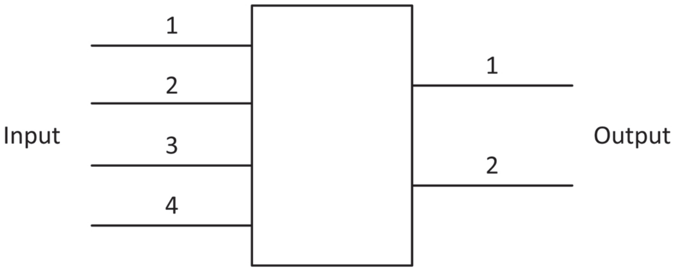
The following list presents some of the interfaces that a PC can include for connecting local peripherals:

- **USB** is the most popular and widely used interface. It was mentioned in the Hard Disk subsection. Almost any modern PC peripherals can be connected through USB ports. Typical examples of external devices that can be connected by using USB are a keyboard, mouse, printer, scanner, camera, UFD, MP3 player, and smartphone.
- **IEEE 1394**, also known as FireWire, is somewhat similar to, but is not as widely used as, USB.
- **Parallel** is a legacy interface. In the past, it was mostly used for connecting a printer or a scanner.

- **Serial** is also a legacy interface. In the past, it was mostly used for connecting a mouse, an external modem, or sometimes a printer. Modems are discussed in Module 2: Networks.
- **PS/2** is yet another legacy interface. In the past, it was used for connecting a keyboard or a mouse. Most modern PCs use USB for that purpose but can still include one PS/2 port for an older keyboard or mouse.
- **Audio** is an interface for connecting audio devices, such as speakers, headphones, microphones, and audio recorders or players.
- **Infrared** is a short-range wireless line-of-sight interface that can be used to connect the same types of devices as the ones that can be connected through USB, such as printers, smartphones, and cameras.
- **Bluetooth** is another short-range wireless interface. Typical examples of devices that can be connected by using this technology are keyboards, mice, smartphones, and cameras.

Firmware

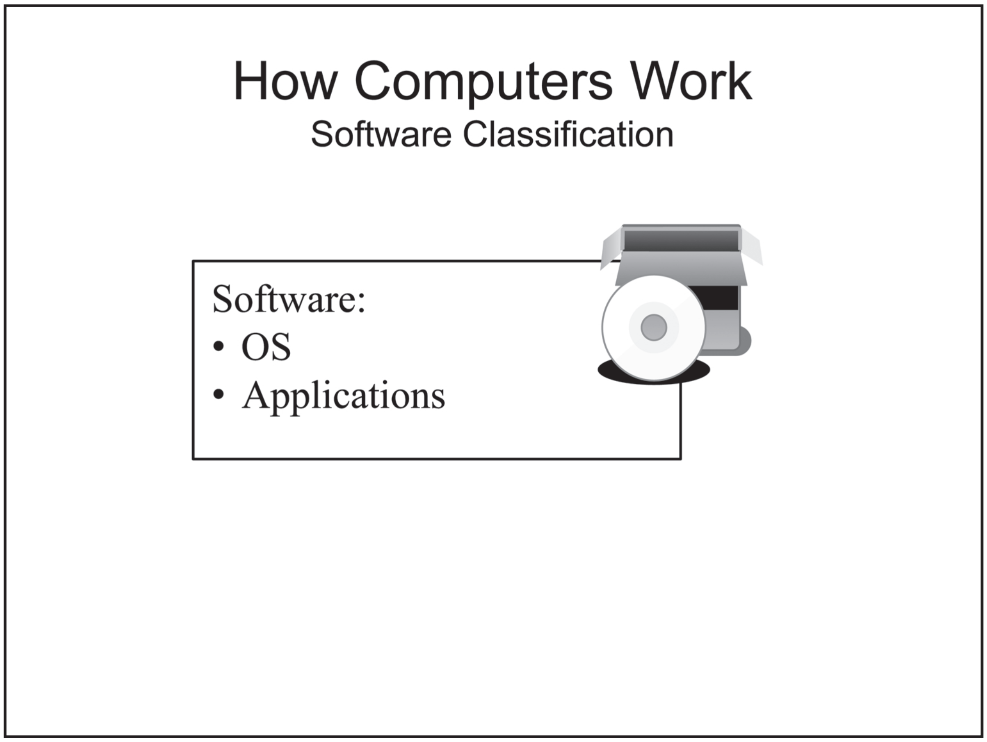
The functionalities supported by a PC and a PC's behavior in specific situations are implemented through two major mechanisms: equipment and programs. The former is commonly referred to as hardware, and the latter is commonly referred to as software. Ultimately, any functionality in a PC boils down to generating specific electrical signals in response to other electrical signals. For example, consider a device whose function is to convert decimal code into binary. Suppose the device has four input lines and two output lines as presented in the following exhibit:



Suppose the purpose of this device is to convert decimal values 0, 1, 2, and 3 into binary values 00, 01, 10, and 11, respectively. When the device receives a signal on the first input line, it generates a signal that represents the binary value of 0 on each output line. When the device receives a signal on the second input line, the device generates 0 on the first output line and generates 1 on the second output line. A signal on the third input line results in 1 on the first output line and 0 on the second output line, and a signal on the fourth input line produces 1 on each output line. If this functionality is implemented through hardware, the device contains electronic circuitry that transforms specific input signals into specific output signals. If this functionality is implemented through software, the device contains some sort of a processor along with the necessary supporting electronic circuitry that can execute commands and contains memory in which those commands are stored. The device runs a program that analyzes input signals and generates the appropriate output signals. The functionality described in this example—converting four decimal numbers into binary numbers—is so trivial that it is much easier to implement it directly through hardware than to design and build all the circuitry necessary for executing commands. However, a programmable device can implement significantly more complex logic on the same hardware, whereas more complex logic implemented through hardware requires more complex circuitry. To modify the behavior of a programmable device, all you need to do is write new code and load it into the device in place of the existing code. However, to modify the behavior of a hardware device, you usually need to create a new device.

Software is a general term that refers to any executable code, but commonly this term is used to refer to only the code that is executed by a computer's CPU. The CPU reads commands from RAM, executes them, and writes any results back into RAM. The commands that the CPU executes are dynamically loaded into RAM as a result of user actions. Firmware is a special type of software that runs in programmable devices. Firmware code for a programmable device is semipermanently stored in the device's local non-volatile random access memory (NVRAM), which is often referred to as *flash memory*. Examples of programmable devices used in PCs are video controllers and NICs. Each of these devices has its own NVRAM, processor, and firmware code.

In addition, a PC's motherboard also has its own firmware that controls the PC's most important basic functionalities. Historically, PC firmware was called Basic Input/Output System (BIOS). About a decade ago, a new generation of PC firmware was developed; it was called Extensible Firmware Interface (EFI). Its newer version is known as Unified EFI (UEFI); it supports all the new EFI features and can also emulate the legacy BIOS environment. Although BIOS and UEFI are significantly different, it is acceptable to refer to both types of firmware as **BIOS** in order to simplify the language in documents that do not focus on the differences between the two types of firmware.



How Computers Work

Software Classification

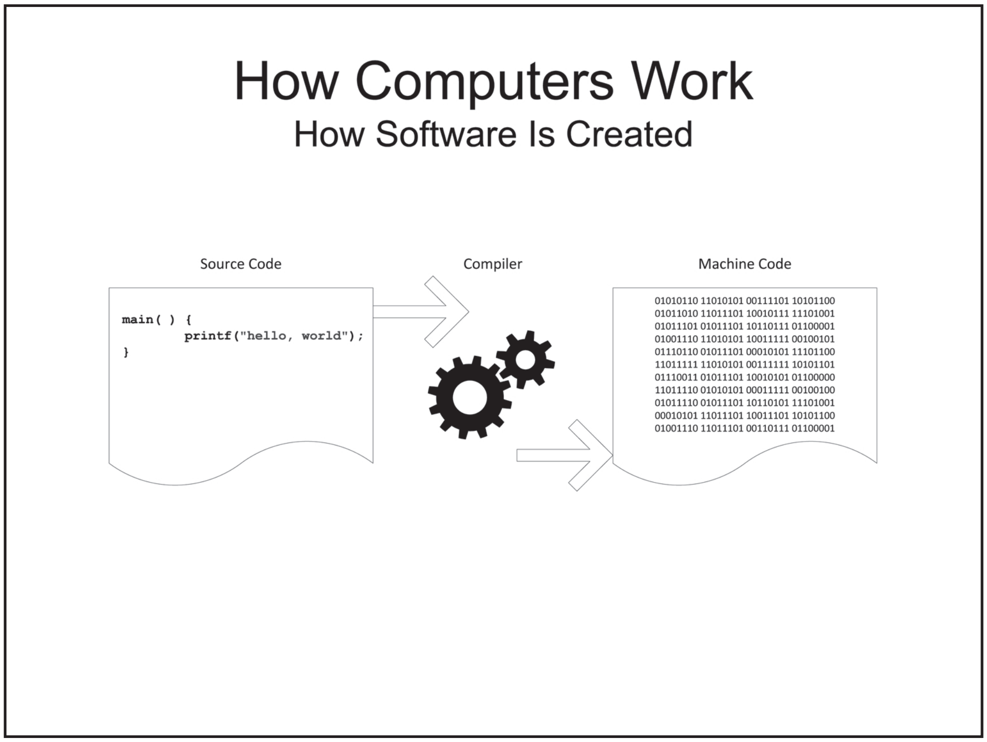
Software:

- OS
- Applications

How Computers Work

Software Classification

A PC is a fairly complex system that performs a multitude of tasks. Many of those tasks are repetitive, such as accepting user input from a keyboard or mouse, writing data to a disk from RAM, reading data from a disk into RAM, reading commands from RAM and executing them, and displaying information on the screen. Nearly every program requires all or some of those actions. However, including all the requisite low-level functionalities in every program would be extremely inefficient because all programs would have to include the same or similar code intended for controlling the entire computer. Such programs would be excessively large and would have to be developed by large teams of programmers, or the code for controlling a computer would have to be copied from another program and adjusted so that it would work with the new program. Any changes to that “borrowed” code would necessitate changes to all the existing programs. A more efficient approach is to separate the software that controls a computer from the software that processes user information or performs other practical tasks for a computer’s user. The software that controls a computer is known as an OS, and programs that perform tasks for a user are often referred to as applications. An OS is automatically loaded each time a computer is started, and applications are usually launched by a user on an as-needed basis. Examples of OSs are Windows, Linux, and UNIX, and examples of applications are Microsoft Word, various email clients, and web browsers.



How Computers Work

How Software Is Created

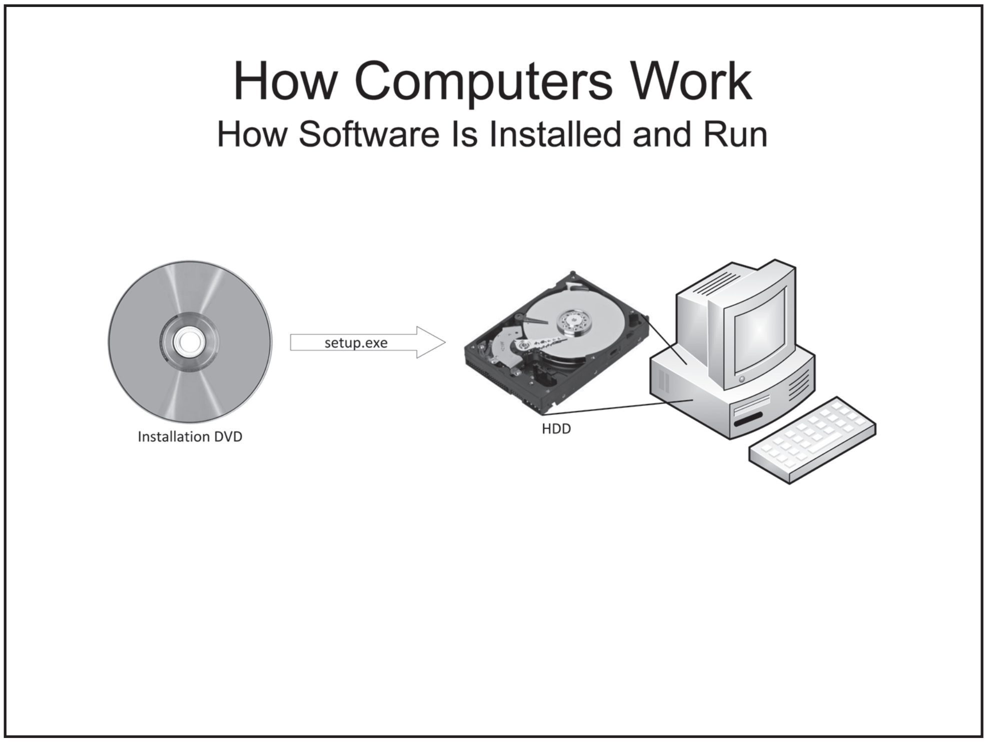
```
main( ) {
        printf("hello, world");
}
```

```
01010110 11010101 00111101 10101100
01011010 11011101 10010111 11101001
01011101 01011101 10110111 01100001
01001110 11010101 10011111 00100101
01110110 01011101 00010101 11101100
11011111 11010101 00111111 10101101
01110011 01011101 10010101 01100000
11011110 01010101 00011111 00100100
01011110 01011101 10110101 11101001
00010101 11011101 10011101 10101100
01001110 11011101 00110111 01100001
```

How Software Is Created

As mentioned earlier, a CPU understands and can work with only 1s and 0s. The raw presentation of software in binary notation is called machine commands or the machine language. The machine language is not well suited for human perception. Writing any, even a very simple, program in machine commands by hand would be a daunting and absolutely impractical task. This leads to the problem of a chicken and an egg. If writing computer programs can be done efficiently only on a computer and a computer can do anything useful only if it runs programs, then what came first: programs or computers? At some point in history, the creators of the very first computer must have had to write binary code and load it by hand. But once that computer became functional, it probably could be used to make writing new programs easier.

Presently, software developers use existing software on existing computers to create new software. They write programs in special human-readable high-level programming languages. Commands in those languages look like normal words, such as Add, Read, Write, Call, or Go To. A program written in a high-level language is referred to as source code. When source code is ready, it is fed to a special program known as a compiler. The compiler translates the source code into the machine language and creates an executable module that contains binary code and can be run on a computer.



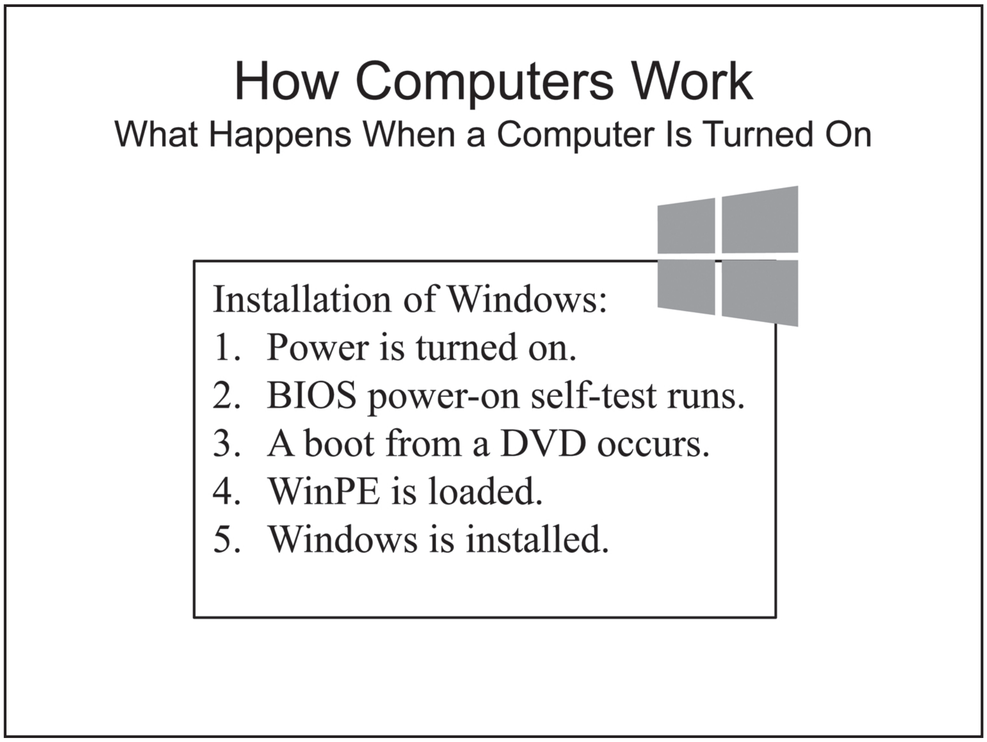
How Computers Work

How Software Is Installed and Run

How Software Is Installed and Run

In addition to the main executable module, most programs also include multiple other modules that contain ancillary executable code or the data necessary to run those programs. All those modules are packaged into a format suitable for distribution. For example, when you buy a program, it is often provided on a DVD, but you usually cannot run the program directly from the DVD. Typically, you need to run a special auxiliary program from that DVD and then that program, which is known as a setup program, copies the necessary files from the DVD to your PC's disk and performs other necessary steps to prepare the program for use. In computer terminology, a *file* is a named set of data that is treated as a unit. For example, you cannot directly copy a portion of a file from a DVD to a disk; you can copy only the entire file. The process of copying a program from its original medium, such as a DVD, to your PC's disk and preparing the program for use is referred to as installation. Correspondingly, the DVD is referred to as an installation DVD and the files on that DVD are referred to as installation files.

Once you have installed a program on your PC's disk, you can run it by calling the program's main executable file. This leads to another chicken-and-egg problem. If a program must be installed on a computer before it can be run and if a computer must be running a program in order to be able to perform any useful work, then how do you install the very first program on a bare-metal computer?



How Computers Work

What Happens When a Computer Is Turned On

Installation of Windows:

1. Power is turned on.
2. BIOS power-on self-test runs.
3. A boot from a DVD occurs.
4. WinPE is loaded.
5. Windows is installed.

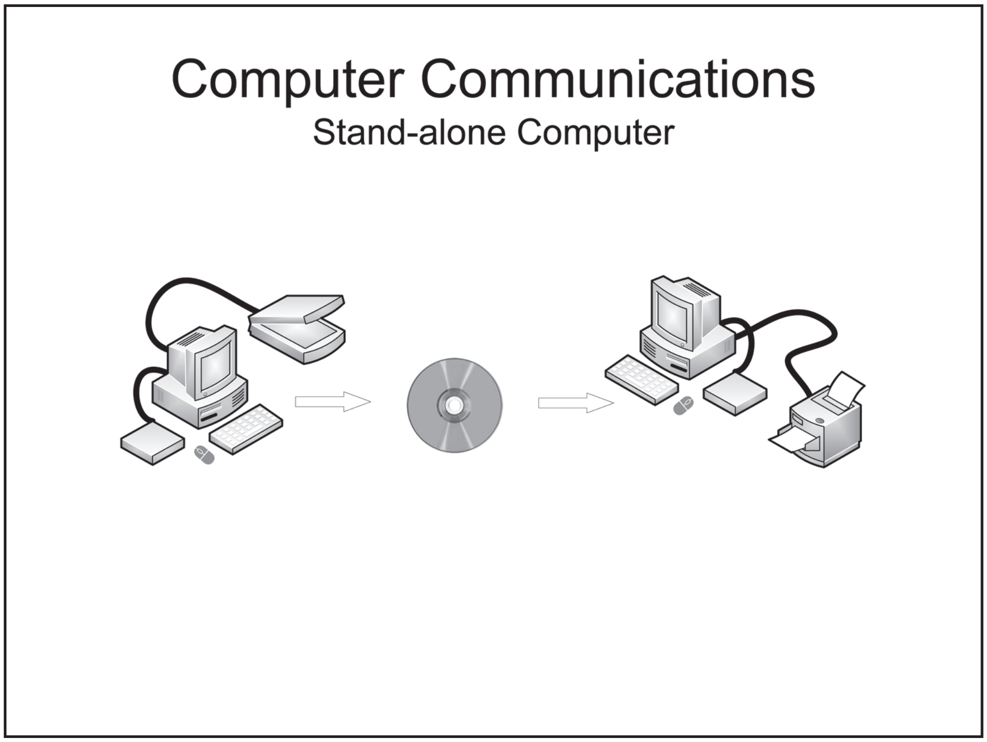
What Happens When a Computer Is Turned On

Typically, when you receive a new PC, it is already fully functional and all you need to do to start using it is just turn it on. But how did that PC become functional? What happens when a PC is turned on for the first time? The exact steps are somewhat different, depending on whether a PC is produced at a factory or is assembled from components at home, but the general idea is still the same. First, you hook up the necessary minimum of peripherals—a display, a keyboard, and a mouse—connect a power cord, and turn on the power. At this point, the PC is blank, or bare-metal; that is, there is no software installed on it yet. But there is BIOS, firmware in the motherboard's NVRAM; it has been preloaded by the motherboard's manufacturer by using special equipment controlled by already functional computers. BIOS automatically runs a program that checks whether the PC is properly configured and ready for work. It also displays a message on the screen that advises you to press a certain key or a key combination on the keyboard if you want to change the default settings. If you do, BIOS runs a setup program, which presents a collection of screens where you can adjust the PC's settings. For example, you can choose which device will be used to load the very first software program. Typically, you would specify an optical drive first and specify a disk second.

Then you should insert a DVD with the installation files for the OS that you want to install. All OS installation DVDs are bootable; this means that they are designed in such a way that BIOS can automatically load some small amount of executable code from an installation DVD into RAM, pass control of the PC to the CPU, and tell the CPU where in RAM the executable code is located. The CPU will run that code, which will load more executable code from the DVD into RAM. In this manner, enough executable code is loaded into RAM to give you some degree of control over the computer and thus make it possible for you to call and execute a setup program from the DVD.

Suppose you want to install a recent version of Windows, such as Windows 8. BIOS will check the optical drive, will discover the Windows installation DVD in it, and will load the initial boot code. Boot in computer speak means *start* or *load*; for example, booting a computer from a DVD refers to starting a computer and loading an initial program from a DVD. The initial boot code will load a small starter OS named Windows Preinstallation Environment (WinPE). WinPE accepts your input from the keyboard and mouse and gives you control of the optical drive and the disk. You are presented with a screen where you can click the **Install now** button in order to start installing Windows. The Windows Setup program prompts you for some basic information and then runs the installation.

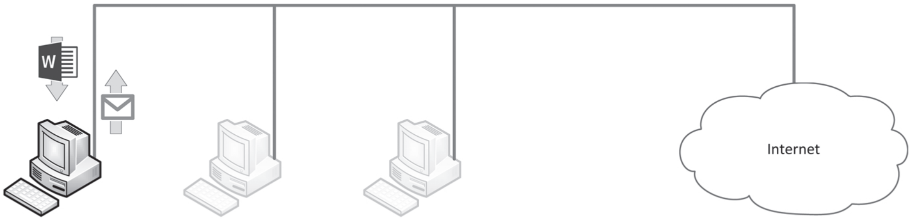
When the installation is completed, the computer automatically reboots (that is, restarts). At this point, you can remove the Windows installation DVD. Because the disk now contains a fully functional OS, the appropriate part of that OS is loaded into RAM and the CPU starts executing the loaded code. When the OS is fully loaded, you have full control of your PC and can install any applications that you want to use, as described in the How Software Is Installed and Run subsection.



Computer Communications
Stand-alone Computer

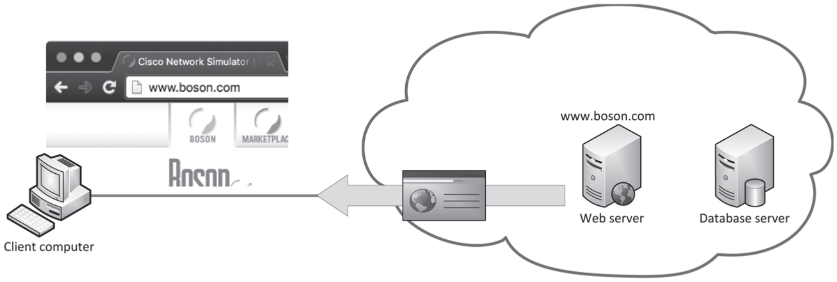
Computer Communications

A stand-alone PC is a computer that is not connected to any other computers or remote devices. A device is deemed remote if it is not attached to your PC directly through a cable or a local wireless link, such as the ones described in the External Connections subsection. On a stand-alone PC, you can install programs from DVDs, UFDs, or other removable media and you can run those programs. The types of work those programs will be able to do depend on the devices attached to your PC. For example, if you have a printer, you can print documents or pictures; if you have a scanner, you can create electronic copies of paper documents; and if you have an audio controller and speakers, you can play music and, possibly, watch movies. The only way for you to acquire new content or to share content available on your PC with other people is through removable media, such as DVDs or UFDs, physically delivered to or from you, respectively.



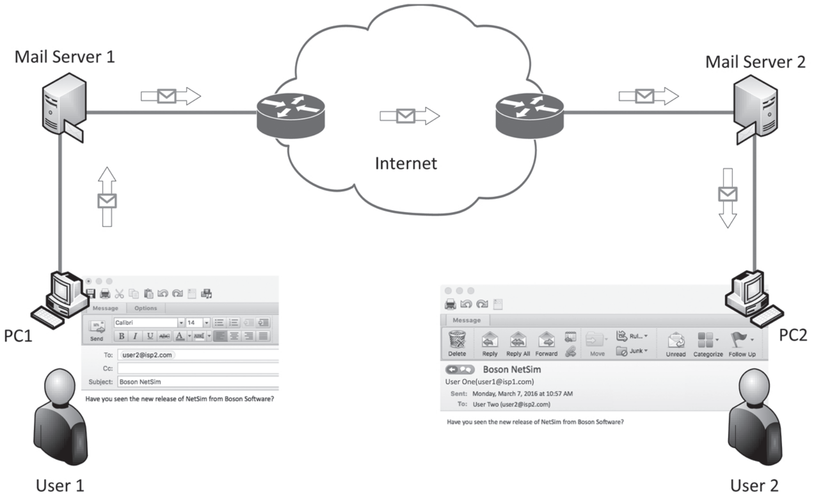
Networked computer

The opposite of a stand-alone PC is a PC connected to a network. Over the network, you can communicate with other computers and their users. For example, you can send and receive emails, download programs, access data on other computers, and allow others to access data on your PC. If your home or office network—whichever network your PC is connected to—is connected to the Internet, you can browse the web.



Computer browsing the web

Your PC as well as the PCs of your friends or colleagues are classified as client computers. There is also another type of computer; it is called a server. Servers run special programs that provide various services to other computers or their users. For example, suppose you open Internet Explorer or another web browser on your PC and attempt to connect to a website. A website is a collection of data that is exposed through special software that runs on a computer known as a web server. The web browser that you are using to access the website acts as a client in relation to the web server software, which serves you the content that you requested. If that website is intended for selling merchandise or services online, the catalog of the available items is probably maintained in a database that is located on another server.



To: user2@isp2.com
Cc:
Subject: Boson NetSim

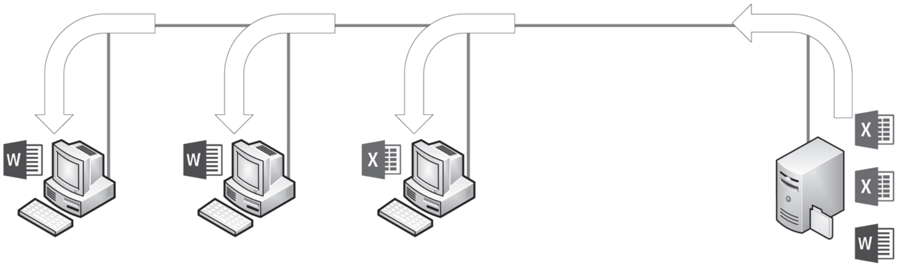
Have you seen the new release of NetSim from Boson Software?

Boson NetSim
User One(user1@isp1.com)
Sent: Monday, March 7, 2016 at 10:57 AM
To: User Two (user2@isp2.com)

Have you seen the new release of NetSim from Boson Software?

Sending an email

Another example is email services. When you want to send an email to another person, you first prepare a message in an email client application, such as Microsoft Outlook, on your PC. The client sends the message to a mail server that the client is configured to use. That mail server forwards your message to another mail server, the one that the email client on the recipient's PC is configured to use. The recipient's email client periodically polls its mail server and downloads any available messages from that server to the local disk on the recipient's PC. The recipient can then open and view those messages in the email client.



File sharing

In a workplace, employees often need access to the same documents. Rather than a separate copy of those documents being maintained on each employee's PC, a single copy of the documents can be stored on a computer known as a file server and can be shared among all the employees. Doing so eliminates the need for ensuring that all the employees' individual copies of the same documents are identical. If only one instance of the documents exists, the file server automatically locks a document while someone is making changes to it so that all the other users can only read that document. Alternatively, documents can be stored on a web server in a special website that is specifically designed for collaboration at the workplace. The web server can be located on the company's private network or can be maintained by a public service provider somewhere on the Internet.

These examples illustrate the importance of computer communications. In the past, a PC was mostly an individual productivity tool. You could write, read, edit, print, or copy documents; make calculations; or keep some records, such as an inventory, a list of contacts, or a business calendar. Without networks and communications with other computers, you could not do much else on your PC. That paradigm completely changed with the advent of the Internet and revolutionary advances in telecommunications technologies over the last two or three decades. Nowadays, almost everyone and everything is connected to the Internet, and the modern society is all about communications and exchange of information.

Index

Index

Certification Candidates

Boson Software’s ExSim-Max practice exams are designed to simulate the complete exam experience. These practice exams have been written by in-house authors who have over 30 years combined experience writing practice exams. ExSim-Max is designed to simulate the live exam, including topics covered, question types, question difficulty, and time allowed, so you know what to expect. To learn more about ExSim-Max practice exams, please visit www.boson.com/exsim-max-practice-exams or contact Boson Software.

Organizational and Volume Customers

Boson Software’s outstanding IT training tools serve the skill development needs of organizations such as colleges, technical training educators, corporations, and governmental agencies. If your organization would like to inquire about volume opportunities and discounts, please contact Boson Software at orgsales@boson.com.

Contact Information

E-Mail:	support@boson.com
Phone:	877-333-EXAM (3926) 615-889-0121
Fax:	615-889-0122
Address:	25 Century Blvd., Ste. 500 Nashville, TN 37214

Boson.com

877.333.3926 support@boson.com