Introduction to Operating Systems

Learning Outcomes

In this lesson you will provide students with an overview of operating systems. At the end of the lesson students should be able to:

LO 1.1 Describe the purpose and functions of operating systems.

LO 1.2 Describe major events in the evolution of operating systems.

LO 1.3 List and compare the common desktop operating systems in use today.

LO 1.4 List the most common mobile OSs, the devices associated with them, and the features found in most of these devices.

Estimated time for lesson: 3 to 4 hours

Preparing for Class

Be sure to review the chapter in advance, making note of the operating systems mentioned in the text. As an overview to the course, this chapter does not contain *Step-by-Step* tutorials, but it does contain 5 *Try This* exercises with a few simple steps to reinforce concepts and to help break up the content. Review each of these before class.

* The *Try This* on page five in the first Learning Objective has students research the current state of tablet computers using any Internet-connected device they have available to them.
* Later in the first Learning Objective, under the topic Task Management, a *Try This* instructs students on how to see the active tasks on a Windows computer and on an Apple Mac.
* On page 11 in the second Learning Objective a *Try This* asks students to research the history of UNIX, providing a link to opengroup.org/unix.
* Also in the second Learning Objective, a *Try This* points students to a link where they can view commercials about Lotus 1-2-3 and the IBM PC. These are both entertaining and a bit educational if you point out that the earliest PCs did not have hard drives, and the OS, the program and the data had to reside on a diskette. Those diskettes they are passing out in the Lotus commercial, would have each held the OS and a separate program (spreadsheet, database, or graphic). So, before the hapless user could use the spreadsheet program, he/she had to put the diskette in the drive on a PC, boot up the computer to the OS installed on that diskette, and then run the app. Then they would have to save the spreadsheet on that same diskette or another diskette. The user would have either saved to the Lotus 1-2-3 program diskette or removed the Lotus 1-2-3 diskette (possible, because the program ran in RAM) and then inserted a data diskette on which they saved their work. This was easier if the user was lucky enough to have a PC with 2 diskette drives. Hard drives were not available on the earliest PCs.
* The final *Try This* is also in the second Learning Objective gives the steps for determining if your computer is running a 32-bit or 64-bit version of Windows. Virtually all Windows 10 is 64-bit.

In addition, although this chapter only lightly touches on hardware, it would also be a good idea to have an already opened desktop system and/or samples of individual components and peripherals for students to examine up close. Invite the students to examine the samples of each hardware component as you discuss what’s inside a microcomputer. Similarly, be prepared to show them a variety of mobile devices—perhaps by simply asking what devices students have with them.

Prerequisites for Class

Ensure that the students are:

* Comfortable with basic computer skills (mouse, keyboard, and touch screen use).
* Able to access a running lab computer, either individually or in small groups.
* Able to browse the Internet and capable of using a Web browser from the available devices.

Class Preparation Notes

The authors have worked to pare down the historical detail of this chapter, while attempting to give students a sense of computing history—the many years and incremental advancement of technologies that brought those ever-present mobile devices into our hands. And there is more to come in their lifetimes.

Emphasize the need for students to pay careful attention to terms and definitions, listed (with page numbers) at the end of each chapter. Point out the Glossary of all the key terms in the back of the book and, due to the large number of key terms in this chapter, consider distributing a chapter glossary (taken from this document) to help bring all students up to speed on some basic concepts prior to beginning later chapters.

General Teaching Tips

The authors begin by justifying the necessity of understanding microcomputer operating systems in the chapter introduction. Students will still wonder why they should care about the many different hardware platforms and operating systems this chapter discusses. Instructors should try and provide rationales for studying these subjects that speak directly to the student’s own interests whenever possible.

This chapter assumes little about the student’s prior knowledge of computers and operating systems. As such, it is ideal for those just beginning their studies. For more experienced students some of the material may be very elementary.

Key Terms

**accelerometer —** A device that detects the physical tilt and acceleration of the device.

**application (app) —** Software that allows a computer user to perform useful functions such as writing a report or calculating a budget. Commonly called an “app.”

**central processing unit (CPU) —** An integrated circuit (chip) that performs the calculations, or processing, for a computer. Also called a microprocessor.

**client —** A software component that accesses services of a network server.

**command-line interface (CLI) —** A user interface that consists of a character-based command line that provides only sparse amounts of information.

**computer —** Adevice that performs calculations.

**Chrome OS –** An operating system based on a Linux kernel with a GUI based on the Chrome web browser.

**cursor —** In a command-line interface (CLI), the cursor is merely a marker for the current position where what you type on the keyboard will appear. In a GUI, a graphical pointer that can have a variety of shapes you can move around by manipulating a pointing device sometimes replaces the cursor.

**device driver —** Software that is added to an OS to control a physical component (device). Each unique hardware device connected to a computer needs a component-specific device driver.

**device management —** An OS function that controls hardware devices using device drivers.

**directory —** A special file that can contain files as well as other directories. This term is most often used with non-GUI operating systems. When describing a directory in a GUI, the commonly used term is *folder*.

**distribution —** A bundling of the Linux kernel and software—both enhancements to the OS and applications. Applications may include word processors, spreadsheets, media players, and more.

**embedded OS —** An operating system stored in firmware, as in a mobile device.

**file management —** An operating system function that allows the operating system to read, write, and modify data and programs organized into files.

**file system —** The logical structure used on a storage device for storing files, as well as the code within an operating system that allows the OS to store and manage files on a storage device.

**firmware —** Software resident in integrated circuits.

**folder —** A term describing a type of file that can contain other folders as well as files—most often used with GUI operating systems. When describing a folder in a non-GUI operating system, people most often use the term *directory*.

**formatting —** The process that writes the logical structure of a file system on a storage device.

**graphical user interface (GUI) —** A user interface that takes advantage of a computer’s graphics capabilities to make it easier to use with graphical elements that one can manipulate to perform tasks, such as system and file management and the running of applications.

**input/output (I/O) —** Anything sent into a computer (input); anything coming out of a computer (output). Every keystroke you enter, all files read in, and even voice commands are input. Output can include a printed page, what you see on the screen, and even sounds.

**integrated** **circuit (IC) —** A small electronic component made up of transistors (tiny switches) and other miniaturized parts.

**Internet of Things (IoT) —** Devices not normally associated with computer that contain microcomputers and are connected to networks. These devices include, but are not limited to: kitchen appliances, thermostats, utility meters, components in automobiles, light bulbs, and industrial control devices.

**job management —** An operating system function that controls the order and time in which programs are run. For example, an operating system's print program can manage and prioritize multiple print jobs.

**kernel —** The main component of an operating system, which always remains in memory while a computer is running.

**killer app –** An application that many people need enough to purchase a computer that will run that app.

**Mac —** The product name for Apple’s computers.

**memory —** The physical chips that store programs and data. There are two basic types: random-access memory (RAM) and read-only memory (ROM).

**memory management —** An operating system function that manages and tracks the placement of programs and data in memory. Advanced operating systems, such as Windows, Linux, and macOS use memory management to make optimal use of memory.

**microcomputer —** A computer built around a microprocessor.

**microprocessor —** An integrated circuit (chip) that performs the calculations, or processing, for a computer. Also called a processor or central processing unit (CPU).

**mobile device —**A hand-help device that uses wireless technologies.

**motherboard —** The central circuit board of a computer to which all other devices connect.

**multitasking —** Two or more programs (tasks) running simultaneously on a computer.

**New Technology File System (NTFS) —** An expandable file system used by the Windows OS. It uses transaction processing to track changes to files and includes many other features. The most important of these features are file and folder security and file encryption.

**operating system (OS) —** A collection of programs that provides a computer with critical functionality, such as the user interface, management of hardware and software, and ways of creating, managing, and using files.

**partition —** An area of a storage device where the logical structure of a single file system may reside.

**personal computer (PC) —** A computer running Windows or Linux.

**plug-and-play (PNP) —**

**portable operating system —** An operating system that you can use on a variety of computer system platforms, with only minor alterations required to be compatible with the underlying architecture.

**processes —** Components of a program that are active in memory.

**random-access memory (RAM) —** Memory that acts as the main memory for holding active programs.

**screen rotation —** A feature of mobile operating systems that takes advantage of the built-in hardware accelerometer in a device by rotating the image on the screen to accommodate the device’s position and allow you to read the screen.

**security —** An operating system function that provides password-protected authentication of the user before allowing access to the local computer. Security features may restrict what a user can do on a computer.

**server —** A computer that plays one or more of several important roles in a network. In all these roles, it provides services to other computers (clients).

**smartphone —** A device that works as a cell phone but also lets you connect to the Internet, view your email, and install a variety of apps for entertainment, education, and work.

**solid-state drive (SSD) —** A storage device that uses integrated circuits, which can be written to and read from much faster than conventional hard disk drives and optical drives.

**system firmware —** Program code that informs the processor of the devices present and how to communicate with them.

**system on a chip (SoC) —** The single microchip containing all or most of the electronic circuity required for a device.

**tablet —** A mobile device that has a touch screen, no integrated keyboard (usually), is larger than a smartphone, and is much more portable than a laptop computer.

**task management —** An operating system function in multitasking OSs that controls the focus. The user can switch between tasks by bringing an application to the foreground, which gives the focus to that application.

**user interface (UI) —** The software layer, sometimes called the *shell*, through which the user communicates with the OS, which, in turn, communicates with the computer hardware.

**virtual keyboard —** A screen image of a **keyboard** with labeled keys that you can tap to enter characters when required.

Lecture Outline

# LO 1.1 Overview of Microcomputer Operating Systems

Teaching Tip:

Essay question 1 at the end of this chapter asks the students to describe the interactions they have had with computers in a typical day. Consider turning this into a classroom discussion and have one student list on the board each unique interaction as it is mentioned.

**Teaching Tip:**

Mention that theoretical designs for computers existed as early as the 1820’s.

## Operating Systems Defined

An **operating system** is a collection of programs that controls all interactions among computer components.

An **application** is software that allows a user to perform useful functions.

## Microcomputers Defined

### Introduction to Section:

#### 1977 First Consumer microcomputers

##### Apple II

##### TRS-80

##### Commodore PET

#### Today’s microcomputer come in many forms

### Hardware Components

#### A **computer** is a device that performs calculations.

### A computer has a **central processing unit (CPU)** that performs the calculations, or processing of the computer.

#### A **microcomputer** is small enough to dedicate to the use of one person.

#### The CPU in a microcomputer is a **microprocessor.**

#### An important invention that led to the miniaturization of computers was the **integrated circuit (IC)**, often called a “chip.”

#### **Input/output (I/O)**: all interaction with a computer

##### Input device: keyboard, pointing device, scanner, touchscreen

##### Output devices: display, printer, storage device, network adapter

Discussion Point:

Take a moment to discuss the commonality all computers share in terms of I/O (input and output) and the basic electronic components (transistors, circuit boards, and IC chips to provide a good basis for understanding later concepts.

Discussion Point

Talk about the various input and output devices. Ask students for examples of each type and ask them what devices may handle both input and output.

#### **Random** Access **Memory (RAM)** is the main **memory**.

#### The **motherboard** is the central computer circuit board where components connect.

### **Firmware** is software that is resident in integrated circuits.

#### **System firmware** contains program code that informs the processor of the devices present and how to communicate with them.

#### Internal and external devices have **firmware** for basic communications between the OS and the hardware.

#### System firmware performs a power on self-test (POST) when a computer is powered on. Older PCs display progress messages during the POST, but today’s computers only display error messages if a problem is detected during the POST.

Discussion Point

Depending on the technical level and interests of the students, discuss the old versus new system firmware: The original system firmware in PCs, called ROM BIOS (read-only basic input/output system), was phased out several years ago by firmware that complies with a newer standard, Unified Extensible Firmware Interface (UEFI**)**. The old ROM BIOS had many technical limits because it was designed to work with the original IBM PC. UEFI is faster and includes security features that protect the computer during that vulnerable time when an operating system is starting up and not entirely in control.

**Teaching Tip:**

Ensure that all students understand the basic function of memory and the measurements used for both memory and storage. Stress that, although we use the same units to measure both memory and storage, memory serves a very different purpose from that of storage.

### Today’s microcomputers

Teaching Tip:

Take a moment to ask students to identify the microcomputers they use.

#### Servers

##### A **server** is a computer that provides services (files, printers, messaging, etc.) to other computers.

##### A **client** computer connects to and uses the services on servers.

#### Desktops and Laptops

##### Personal computers (PCs) follow the Microsoft/Intel standard run Microsoft Windows or a distribution of Linux.

##### A laptop has a flat screen and a keyboard, each integrated into a panel with a hinge holding the two together.

##### Apple **Mac** computers are available in desktop (the all-in-one iMac) and laptops.

##### The common operating systems for desktops and laptops are Apple’s macOS for Apple brand computers, Microsoft Windows and Linux for other PCs and laptops, and the Chrome OS for Chromebooks.

##### In this text, a **personal computer (PC)** is a desktop computer running Windows or Linux, and the term **Mac** refers to the Apple iMac desktop computers as well as the MacBook laptop computers, which run Apple macOS.

Discussion Point and Tip:

Ask the students what devices they use the most for their school work. Make note of which type of device and operating system is the most prevalent and consider how to tailor the content to their needs.

#### Mobile Devices

##### A **mobile device** has all or most of its electronic circuity on a single microchip. This is called **system-on-a-chip (SoC)**.

##### Some are single-purpose (inventory scanners).

##### Many run proprietary OSs; other run scaled-down versions of desktop OSs.

##### A mobile device stores its OS in firmware, as an **embedded OS**

##### Popular mobile OSs include Apple iOS and Google Android.

##### A **smartphone** includes both cellular voice and Internet data communications.

###### Apple iPhone and various models by Motorola, Nokia, HTC, Samsung, LG, Google, and others.

##### A **tablet** has a touch screen, no integrated keyboard, and is usually larger than a smartphone. A tablet may or may not include cellular hardware, depending on accessing the Internet through Wi-Fi routers.

Teaching Tip:

Take a moment to allow students to do the Try This! Activity titled More About Tablets.

#### Internet of Things

##### The **Internet of Things (IoT)** describes the wide array of devices that previously may not have contained microcomputers, but now do and are also connected to networks.

##### Examples: kitchen appliances, thermostats, utility meters, components in automobiles, light bulbs, and industrial control devices.

## Functions of Microcomputer Operating System

**Teaching Tip:**

Explain to the students that this overview of operating systems will provide them with an understanding of the purpose of operating systems.

### The Big Picture

#### Power on (start up, boot up) a computer.

#### The **kernel** is the core component of the OS that remains in memory while the computer is turned on.

#### A developer writes an app that interacts with the OS making requests for hardware services.

#### OS interacts with the hardware on behalf of the app.

### **User Interface**

#### The software layer, or shell, through which the user communicates with the OS.

#### Includes the command processor.

#### Visual components of the OS. It may be command-line based (DOS) or it may be a graphical user interface (GUI) like the Apple Mac OS or Microsoft’s Windows OS.

#### A **command-line interface (CLI)** requires the user to type commands. A blinking **cursor** shows the insertion point for typing.

#### A **graphical user interface GUI** requires the user to learn the meaning of various graphical elements and how to use them.

### **Job Management**

##### An operating system function that controls the order and time in which programs run.

##### Scheduling programs.

##### Managing and prioritizing print jobs.

### **Task Management**

##### Most operating systems can **multitask** or run more than one program simultaneously.

##### Responds to the user’s control of the focus (where the system’s attention is at any given moment).

##### Task management allows the user to switch between different tasks.

##### Each program may include many small components called **processes**.

Teaching Tip:

Take a moment to allow students to do the Try This! Activity under Task Management.

Discussion Point

For more detail on task management, explain that a computer cannot simultaneously run more tasks than the number of processors that exist within the computer. Not too long ago, most microcomputers had only a single processor, so they accomplished multitasking through a scheme that made order out of chaos by determining which program responds to the key­strokes and mouse movements. New processors can have multiple CPUs within a single chip, but not usually enough for the many processes, so they have true multitasking coexisting with task switching.

### **File Management**

##### Also called data management.

##### Allows OS to read, write, and modify data.

##### Data (and programs) are organized into files.

##### A **file system** is the logical structure used on a storage device for storing files. It also includes the code within an operating system that supports file management.

##### **Formatting** writes the logical structure to a storage device.

##### A file system uses storage, such as a conventional hard disk drive or **solid-state drive (SSD).**

Discussion Point:

Not to put too fine a point on the file management discussion, briefly mention that the operating system maps the logical organization of the file system to physical locations on the storage device, keeping track so that all pieces of a file can be retrieved.

**Teaching Tip:**

More detail on solid-state drives (SSDs): They use rewritable integrated circuits, which the computer can write to and read from much faster than conventional hard disk drives and optical drives.

##### A **partition** is an area of a storage device where the logical structure of a single file system resides.

##### File management allows users to organize their files in special files called **folders** or **directories**.

### **Device Management**

##### Controls hardware using device drivers.

##### A **device driver** is a special program installed into an operating system for controlling a device.

##### Device drivers are unique to a device (brand and model) that allows the device to respond to requests from the operating system.

##### OSs are **plug-and-play (PNP**), detecting an installed device and automatically installing the device driver.

**Teaching Tip:**

Beginning on page 7, time line of events relating to microcomputer operating systems runs along the bottom of several pages of this chapter.

### **Memory Management**

##### Manages the placement of programs and data in memory.

##### Makes optimal use of memory, allowing more code and data than physical RAM can hold. Can cause computer to slow down.

##### Adding more memory (up to limits of hardware and OS) will allow apps to run faster.

Discussion Point:

Ask students if they have heard of a computer being slow because it does not have enough RAM. Do they know why this happens? Explain that if a computer does not have enough physical RAM for the active programs and their data, the OS memory manager will move code and data, as necessary, to a portion of the disk defined as virtual memory. The OS does this transfer of code and data that are part of a program that currently does not have the user’s attention. Then point out that a computer may become noticeably slower when this occurs.

### **Security**

##### Provides password-protected authentication of the user before allowing access.

##### An operating system may provide other security features beyond authentication, such as antivirus protection and file and folder permissions.

Discussion Point:

If the operating system and its file system support it, an administrator can grant or withhold special permissions from accounts to allow for a customized level of security for each user. More on this in Chapter 2.

# LO 1.2 Yesterday’s Operating Systems

**Teaching Tip:**

Many events that appear on the timeline are not discussed in the content. Review this time line and be prepared to discuss selected events with the students. Consider recent events you would add to it.

## Introduction to Learning Objective 1.2

### Miniaturization of computer components led to evolution of desktop OS available to consumer.

### Over several decades.

### Technical advances

### Evolutionary changes in how people use computers.

### Computers as multipurpose devices.

### Early user government agency, research institute, or large business.

### Each user defined the single purpose of computer.

## UNIX — The Operating System for All Platforms

### Started with an OS for a Digital Equipment Corporation (DEC) computer.

### Further work done at Bell Labs

#### 1975: UNIX emerged from Bell Labs Computer Science Research Center (Bell Labs) as UNIX version 6 for minicomputers and mainframe computers distributed with government, commercial, and academic licenses.

#### A **portable operating system** works on a variety of platforms with minor alterations.

### University of California at Berkeley

#### Licensed UNIX from Bell Labs, modified it, and distributed it as Berkeley Software Distribution (BSD) version 4.2.

#### Students and others improved on and added to UNIX, sharing code.

### UNIX Going Forward

#### Free BSD, Net BSD, Open BSD, and Open Solaris.

#### Commercial versions: AIX, OpenServer (derived from SCO UNIX), and HP/UX.

#### UNIX versions run on PCs and very large computer systems including many Internet servers.

#### Some versions use traditional CLI shells, such as the Bourne shell or the C shell.

#### GUI shells are available, such as GNOME or KDE.

**Teaching Tip:**

Point out the Try This! Titled “Research the History of UNIX” and either assign it as outside work or allow time in class to complete this activity. Take class time to discuss what students learned about recent use of UNIX.

**Teaching Tip:**

There are many open source UNIX versions complying with the standard of the Open Source Initiative (OSI). See www.opensource.org.

## The Evolution of Desktop Operating Systems

### Small Steps

#### Microcomputers arose in the 1970’s with the introduction of computers like the TRS-80 and the Apple II.

#### Designers built the very earliest machines as single-function devices, and, as such, they had no real need of a user interface or obvious operating system.

#### Operating systems evolved from the need for computers to be more multipurpose, rather than just being single-purpose machines.

### Early Apple Computers and Their OSs

#### 1976: Steve Jobs and Stephen Wozniak founded Apple Computer based on the Apple 1.

#### 1977: They introduced the Apple II at the West Coast Computer Faire in San Francisco.

#### 1978: They added disk drives to the Apple II.

**Addition Info:**

In 1982 Apple introduced the Lisa, the first commercially available computer with a purely graphical operating system—and a mouse. However, this computer lacked something very important for consumers—applications. It was unsuccessful, and Apple’s own Macintosh computer, released in 12984, overshadowed the Lisa and marked the beginning of consumer excitement and the near-cult following of the Apple computer products. The Macintosh came with the Mac OS System 1, a GUI operating system that used a mouse. Apple improved the Mac OS over the years to include many easy to use feature.

The final release of the classic Mac OS family was Mac OS 9, introduced in 1999. With its roots in the original 1984 OS, Apple revised and improved the operating system to support multiple uses, but it was weak in memory management full multitasking. In 2001 it was replaced by a completely new operating system—Mac OS X, based on UNIX. There is a brief overview of OS X later in this chapter and more detail on this OS in Chapter 7.

#### The Killer App for the Apple II

##### An app that solves a problem, inspiring many to purchase a computer that will run that app.

##### In 1979 the electronic spreadsheet program VisiCalc for the Apple OS on the Apple II filled a need.

##### Apple, however, was slow in updating the Apple II, opening the door for IBM to introduce their IBM-PC.

#### The Lisa Computer—a pretty face but no apps.

#### 1982 Apple Lisa first commercially available computer with a GUI and a mouse.

#### Not successful due to lack of apps, let alone a killer app.

#### Apple Macintosh and the Mac OS

##### 1984 Macintosh computer with Mac OS System 1

##### GUI OS that used a mouse.

##### Improved and added apps.

##### Mac OS replaced in 2001 by Mac OS X, based on UNIX.

##### 2016 Apple renamed Mac OS X to macOS.

**Discussion Point:**

Early in Apple’s history, the company strategically targeted schools and universities as places to sell its products, which, over the years, has resulted in large numbers of people who learned computing on a Mac; this has contributed to Apple’s loyal following.

### The IBM PC

#### Introduced in 1981.

#### Exceeded IBM’s sales forecast of a quarter of a million units during its predicted 5-year product lifetime.

#### IBM took orders for half a million computers in the first few days after the introduction.

#### Sold well despite $5,000 price tag for a typical configuration.

#### IBM logo implied it was a serious business computer.

#### IBM PC DOS

#### IBM initially offered the IBM-PC with either Microsoft’s DOS or Digital Research’s CP/M operating system.

#### IBM PC included Microsoft’s BASIC interpreter in firmware.

#### The Killer App for the IBM-PC

#### Lotus 1-2-3

#### It was more capable and sophisticated spreadsheet program than VisiCalc.

#### PC DOS versus MS-DOS

#### PC DOS for IBM computers.

#### MS-DOS Microsoft’s product for non-IBM PCs.

#### DOS = disk operating system

#### Single-tasking

#### No support for virtual memory

#### No native GUI

#### No built-in security

#### Limited memory support

#### Text mode CLI

#### PC DOS 1.0 supported single-sided 5.25-inch floppy disks.

#### Each major version of DOS (PC or MS) supported new disk types and/or capacities.

#### MS-DOS 6.22 last widely used version.

#### Some forms of DOS still available from 3-rd party source.

#### Demand for DOS dwindled.

**Teaching Tip:**

There are many stories (some true, many mythical) that surround the birth of MS-DOS and its relationship to CP/M. Students will often volunteer outlandish stories they have heard. The few sentences on this topic in the text derive from a well-documented version. If possible, avoid the false techno-theology discussions that assume that Microsoft is the Evil Empire. As time goes by, these stories have less weight with younger students and the whole thing becomes a non-issue.

Ask the students what they would consider as being today’s “killer app.” Possible candidates might be an app that would compel millions of non-gamer consumers to purchase VR headsets for personal use. What about more AR apps for smartphones? Or perhaps the availability of a vast number of iPhone and Android apps counts collectively as the killer app for each mobile OS.

**Discussion Point:**

There were several notable West Coast Computer Faires. Chuck Peddle also introduced his Commodore PET computer at the first West Coast Computer Faire in 1977. Reports state that more than 12,000 people attended, and among the 180 exhibiters were Intel, MITS, and Digital Research. At the fourth West Coast Computer Faire, in 1979, Dan Bricklin demonstrated VisiCalc (the first “killer app” discussed earlier). At the sixth, Adam Osborne introduced the Osborne 1 portable computer (see 1981 on the timeline on page 12).

**Discussion Point:**

DOS provided support for interaction, or input and output (I/O), between the memory and disk drives. It does not have the more advanced capabilities of today’s OSs. It is single-tasking OS, has no support for virtual memory, no native GUI, and no built-in security function. While DOS could use much more memory than the OSs that preceded it, it had very limited memory support compared to Windows and Apple macOS.

PC DOS 1.0 supported single-sided 5¼-inch floppies; PC DOS 1.1 added support for double-sided 5¼-inch floppies; and PC DOS 2.0, released with the IBM PC-XT, included support for the XT’s 10MB hard drives. PC DOS 3.0 was released with the IBM PC-AT and included support for the larger AT hard drives. Support for 3½-inch floppies and the larger hard drives of the IBM PS-2 computers were added in PC DOS 4.0.

**Discussion Point:**

In the past, computer professionals often found DOS handy as a very small OS that fit on a floppy disk, to which they added various utilities for troubleshooting computers. This practice has disap­peared today, as have floppy disks and floppy disk drives. Those same techs moved on to using optical discs (CDs or DVDs) or a flash drive loaded with specialized software for their work.

### OS/2

#### 1987: Microsoft and IBM developed OS/2 to replace DOS by offering features like multitasking.

#### Underpowered and required more expensive computer than DOS.

#### Limited support for DOS apps, which people wanted to continue to use.

#### 1994: IBM retreated from the desktop when they released OS/2 Warp, with a GUI, primarily sold with high-end servers.

#### 2003: IBM announced that it would not develop any future versions.

#### 2004: IBM sold its PC division (along with OS/2) to China-based Lenovo Group.

#### 2005: OS/2 support discontinued.

### Microsoft Windows

#### Introduced in 1985 as GUI shell on top of MS-DOS.

#### Slowly evolved into more advanced OS.

#### Dominated the desktop for many years.

#### Windows’ fate tied to PCs.

#### As more people move away from PCs to mobile devices, PC sales decline.

#### PCs sales compete with Apple iPads and Chromebooks in the K-12 education market.

#### Windows versions 1 through 3

#### 1985: Microsoft released the first version of Windows. It was essentially a DOS application that provided a GUI for DOS rather than an operating system in and of itself. It had little success.

#### 1990: Windows 3.0, the first successful version. It had the ability to run both DOS and Windows apps and some real added functionality.

**Discussion Point:**

Windows 3.0 still depended on DOS and was not a stand-alone OS. Windows 3.x worked in Intel 386 processor modes: real mode, standard mode, and 386 enhanced mode (Microsoft’s term for the Intel 386 processor modes).

#### 1992: Microsoft upgraded Windows to Windows 3.1 and found huge success, partially because of the availability of Microsoft Office for Windows and other applications written for Windows.

**Discussion Point:**

The authors attribute the success of Windows 3.1, in part, to the ability to run Microsoft’s legacy DOS apps (Word and Excel) from the 1980s. Discuss how the wide acceptance of Microsoft Office as the default productivity suite in large organizations continued to encourage the adoption of new versions of Windows.

#### Windows for Workgroups

#### DOS, Windows, and other OSs required a separate network operating system (NOS) if you wanted to network a personal computer.

#### 1992: Microsoft released Windows for Workgroups 3.1

#### Supported peer-to-peer networking with both client and server software within the OS.

#### Still dependent on DOS and not a stand-alone OS.

#### Windows NT

#### 1993: Microsoft introduced Windows NT.

##### The same user interface as Windows 3.1, so the first version was Windows NT 3.1.

##### Included the New Technology File System (NTFS).

#### True stand-alone operating system specifically designed for network servers.

#### The first Microsoft OS to take full advantage of an Intel processor’s protected mode, thus more stable secure.

#### 1994: Windows NT 3.5 was split into two editions.

##### Workstation, a desktop operating system for client machines.

##### Server, a network operating system for servers.

#### 1996: Windows NT 4.0 with a GUI similar to Windows 95 (1995).

#### Windows 95

#### 1995: Windows 95 introduced; predated Windows NT 4.0 Workstation.

#### A continuation of the Windows 3.X model: GUI on top of DOS.

#### Included both 16-bit and 32-bit code.

#### And a new GUI

#### Windows 98

#### 1998: an evolutionary upgrade from Windows 95.

#### More stable, and offered greater integration with Internet Explorer, more customization, and support for new devices like DVD drives.

#### Biggest drawback was lack of security, no local accounts database, and no support for NTFS.

#### Windows Me (Millennium Edition)

#### 2000: targeted the home market, home game user.

#### The last OS based on the original Windows 95 kernel.

#### Installed on computers intended for the home market.

#### Windows 2000

#### 2000: included the best features of Windows 98 and Windows NT.

#### Available in several editions from desktop to enterprise server.

#### Windows XP

#### 2001: strictly a desktop operating system.

#### Several editions. Most common:

##### Windows XP Home Edition.

##### Windows XP Professional.

##### Windows XP Media Edition.

##### All editions share the same GUI. Only the Professional version offers certain network- and security-related features.

#### Windows XP 64-bit Edition for the Intel Itanium processor platform that only supported 64-bit applications.

#### Last service pack for XP was SP3 for 32-bit editions only. MS ended support for XP with SP 2 in July 2010.

#### The last SP for 64-bit editions was SP 2, and support for this will end April 2014, along with support for XP SP 3 for 32-bit editions.

#### Windows Vista

#### 2007: perceived as an upgrade to Windows XP.

#### New GUI, many other improvements

#### Not widely adopted due to problems with speed, high hardware requirements, and annoyance caused by a new security feature.

#### Mainstream support ended April 2012.

Teaching Tip:

The terms 32-bit and 64-bit are mentioned several times in this chapter and lightly defined in a Note. Take time to discuss the differences between 32-bit and 64-bit OSs, comparing Memory Limits of 32-bit versus 64-bit Windows editions. Have students take time to do the Try This labeled Are You Running 32-bit or 64-bit Windows?

Teaching Tip:

**Numbering Systems:** Determine if you believe the students need a primer on numbering systems. If so, assign bonus Project 1, found near the end of this document. The Solution provided includes a comparison table of binary, decimal, and hexadecimal.

This may lead to a need to explain “bytes” as in kilobyte, megabyte, and so forth. These are ways to express large binary numbers related to computer memory and storage. The following may be helpful:

When we talk about storing things in memory or on disk, we use terms like megabyte and gigabyte to describe amounts of memory or disk space. To understand these terms, first consider the smallest unit of storage (disk or memory), which is a binary digit (abbreviated as bit). You can think of a single bit as being like a light switch: it is either on or off. When it is on, it represents 1; when it is off, it represents 0. Computers (or the folks who make computers) like binary notation because it can represent anything that has two states, like on or off. This is exactly how RAM and ROM work—with the equivalent of on and off switches. Floppy and hard disks have a metallic oxide coating that contains particles that an electrical charge can magnetize (polarize), or be left non-magnetized, and can thus represent on and off states.

Computers often use bits in groups of eight, which we call a byte. A single byte can represent a character, like the letter A in a word processing document, or a very simple command, like the command to move down one line. When you have 1,024 bytes, you have 1 kilobyte (2 to the 10th power—kilo means thousand); 1,048,576 bytes equal 1 megabyte (2 to the 20th power—mega means million); 1,073,741,824 bytes equal 1 gigabyte (2 to the 30th power—giga means billion); 1,099,511,627,776 bytes equal 1 terabyte (2 to the 40th power—tera means trillion). (Notice that the actual number of bytes is not a round number, so when you have a kilobyte of data, you have a little more than a thousand bytes—and it really adds up! A gigabyte is almost 74 million bytes larger than you would expect.)

# LO 1.3 Today’s Desktop Operating Systems

**Discussion Point:**

Point out Table 1-1 as a guide to the OSs discussed in this section. Try to hold off detailed discussion of specific OSs until the appropriate chapters.

### Windows 7, Windows 8, Windows 10, macOS, Chrome OS, and Linux.

### Summary in Table 1-1.

## Today’s Windows for the Desktop

### Windows 7 (Chapter 4)

#### Corrected problems of Windows Vista.

#### Redesigned desktop.

### Windows 8 and 8.1 (Not given a dedicated chapter in this book.)

#### 2012:

#### Faster than previous versions, better security, and improved wireless connectivity.

#### Supports newer hardware such as USB 3.0 ports and touch screens.

#### Includes two different GUIs.

##### Default Start screen GUI adopted from Windows Phone 7.5. Objects without shading and borders to run on a wide range of devices including PCs, laptops, and tablets.

##### Second GUI is a modified version of the Windows 7 desktop without the Start menu.

##### An unsuccessful version of Windows.

### Windows 10

### 2015:

### Was a free upgrade for one year after introduction

### A new Start menu.

### Windows Hello facial or fingerprint biometric authentication.

### Updated every month with security and fixes.

### Feature updates annually.

## Apple macOS

#### Previously Apple used proprietary hardware and software for better integration of the OS and hardware.

#### From mid-1990s – 2005: Apple computers used the Motorola PowerPC chip.

#### Since 2005: switched to Intel platform, but macOS is only licensed to run on an Apple computer.

### Today’s macOS

##### Based on NextStep OS with a UNIX kernel

##### GUI with optional CLI

## Linux

#### Development began 1991 and continues today.

#### Modeled on UNIX, and named for its original developer, Linus Benedict Torvalds, who began the Linux project. Linux was created with other programmers as an open-source OS for modern computers.

#### Developed in the C language using a free C compiler created through the GNU project, the GNU C Compiler (GCC).

#### Many open source (free) or inexpensive distributions are available.

#### Available in both 32-bit and 64-bit distributions.

#### Natively uses CLI, but excellent GUI shells come with popular distributions.

## Chrome OS

#### OS with Linux kernel plus a GUI

#### GUI based on Chrome web browser.

#### Chromebooks with Chrome OS offered by several manufacturers.

# Today’s Mobile Operating Systems

#### Mobile OSs have followed the trajectory of all computing, thanks to miniaturization of components and new technologies.

#### No single “killer app” for mobile devices, rather a large number of compelling apps.

#### PCs were productivity tools, but today’s mobile devices are personal for communicating, entertainment, and work and school tasks.

## Mobile Devices

#### Many manufacturers, but focus on those using the three OSs; Mac iOS, Android, or Windows.

#### They all support a variety of wireless technologies and the ability to customize them with mobile apps.

#### Mobile devices include these hardware features:

##### Network adapters for various types of wireless networks.

##### High-quality color touch screens that respond to several types of touch gestures.

##### One or two (front and back) digital cameras.

##### Built-in speakers or external speaker port.

##### Rechargeable batteries that can get through a day of use.

##### An **accelerometer** that detects physical tilt and acceleration of the device.

##### Solid-state **drives** (SSDs).

## Connectivity

#### Smart phones connect to cellular networks that provide voice and data connectivity.

#### Can also connect to Wi-Fi networks.

#### Can also connect using Bluetooth.

## Mobile Operating System Features

### Touch screen and virtual keyboard support

### Screen rotation.

### Updateable

### Availability of apps

### Security

### Data Synchronization

Bonus Project 1

Over the years, digital computers have used many different numbering systems. Three numbering systems: binary, decimal, and hexadecimal, are commonly used in today’s computers.

Research and contrast these three numbering systems.

Bonus Project 1 Solution

A numbering system is based on the number of digits available for use. Binary uses only two digits and uses just two numerals, 0 and 1, hence: binary. Decimal has a ten-digit base and uses the numerals 0 through 9, while hexadecimal has a 16-digit base and uses the numerals 0 through 9 and the letters A through F.

In very general terms, computers use decimal to communicate with people, hexadecimal is used to address memory, while binary is used to transfer data. You can add, subtract, multiply, and divide in each of these systems, and fortunately, there are calculator apps for every OS that can do that as well as translate numbers from one system to another. The following table of equivalents shows numbers in all three systems:

|  |  |  |
| --- | --- | --- |
| **Binary** | **Decimal** | **Hexadecimal** |
| 1 | 1 | 1 |
| 10 | 2 | 2 |
| 11 | 3 | 3 |
| 100 | 4 | 4 |
| 101 | 5 | 5 |
| 110 | 6 | 6 |
| 111 | 7 | 7 |
| 1000 | 8 | 8 |
| 1001 | 9 | 9 |
| 1010 | 10 | A |
| 1011 | 11 | B |
| 1100 | 12 | C |
| 1101 | 13 | D |
| 1110 | 14 | E |
| 1111 | 15 | F |
| 101 1101 0100 | 1492 | 5D4 |
| 111 1101 0100 | 2004 | 7D4 |

Bonus Project 2

Ask the students to open a browser and connect to the Computer Hope (computerhope.com) website. This site contains tips and information on desktop computing. At this time the home page contains a link titled “History,” under the heading Information links on the far right of the page. Click this link. On the Computer history page you will find links to many topics.

Direct the students to research the history links titled:

* Apple operating system history
* MS-DOS history
* Microsoft Windows history
* UNIX, Linux, and variant history

Ask the students to compare and contrast the nature and pace of development for these major operating systems. Ask them to describe the differences in the overall development tracks and theorize about some possible explanations for the differences they find. Have the students prepare a short (3- to 4-paragraph) paper on the subject.

Bonus Project 2 Solution

The students should notice much more activity in the development of Windows (more variants developed faster) than in the development of DOS. They may notice that development for Apple OSs was very steady until the 1990’s, and then it picked up beginning in 2001 with the introduction of macOS and its upgrades. They may also notice far more activity in the development of UNIX/Linux variants than in the other OSs. Possible explanations might include the profit motive that drives Microsoft and Apple as opposed to the open source nature of UNIX/Linux development. Students might also discuss the different customer bases served by the UNIX/Linux products as opposed to those served by the Apple and Microsoft OSs. Then they should consider the differences in customer bases between Apple and Microsoft OSs. Did they discover a convergence in recent years?

Assessment Quiz

This quiz will test the knowledge students have gained during the lesson.

Questions

1. The \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ is the software layer, sometimes called the shell, through which the user communicates with the OS, which, in turn, communicates with the computer.
2. A user interface that takes advantage of a computer’s graphics capabilities to make it easier to use is a \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.
3. The operating system function that controls the order and time in which programs run is known as \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_. An example would be a print program that manages and prioritizes multiple print jobs.
4. The central circuit board of a computer is the \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_. All other devices connect to it in one way or another.
5. Typing at a computer keyboard or manipulating a mouse, combined with the result you see on the screen are examples of \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.
6. The \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ is based on a Linux kernel with a GUI based on a web browser.
7. A \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ is a microcomputer that complies with the Microsoft / Intel set of standards and usually runs versions of Windows or (to a lesser degree) Linux.
8. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ means that the source code of a computer program is made available free of charge to the general public.
9. A/an \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ is also called a directory.
10. A/an \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ in a CLI is a blinking marker for the current position where what you type will appear, with in a GUI it is a graphical pointer that can be moved around by manipulating a pointing device.

Answers

1. The *u*s*er interface* is the software layer, sometimes called the shell, through which the user communicates with the OS, which, in turn, communicates with the computer.
2. A user interface that takes advantage of a computer’s graphics capabilities to make it easier to use is a *graphical user interface (GUI).*
3. The operating system function that controls the order and time in which programs run is known as *job management*. An example would be a print program that manages and prioritizes multiple print jobs.
4. The central circuit board of a computer is the *motherboard*. All other devices connect to it in one way or another.
5. Typing at a computer keyboard or manipulating a mouse, combined with the result you see on the screen are examples of *input/output (I/O)*.
6. The *Chrome OS* is based on a Linux kernel with a GUI based on a web browser.
7. A *personal computer (PC)* is a microcomputer that complies with the Microsoft/Intel set of standards and usually runs versions of Windows or (to a lesser degree) Linux.
8. *Open source* means that the source code of a computer program is made available free of charge to the general public.
9. A *folder* is also called a directory.
10. A *cursor* in a CLI is a blinking marker for the current position where what you type will appear, with in a GUI it is a graphical pointer that can be moved around by manipulating a pointing device.