Chapter 2
Computers and Networking

Adam Flanders

Contents

2.1 Introduction .............................................. 15
2.2 Computers 101 – Hardware ................................ 16
  2.2.1 Hardware Elements of Computers ...................... 16
2.3 Computers 101 – Software ................................. 19
  2.3.1 Computer Operating System ............................. 19
  2.3.2 Application Software .................................... 21
  2.3.3 Low-Level Programming Language ..................... 21
  2.3.4 High-Level Programming Language .................... 22
2.4 Computer Networking ...................................... 23
  2.4.1 Physical (Hardware) Networking Components ............ 23
  2.4.2 Network Switches ....................................... 24
  2.4.3 Network Protocols ....................................... 25
  2.4.4 Data Packets ............................................ 26
2.5 Client–Server Architecture .................................. 28
2.6 Database Applications ..................................... 29
Self-Assessment Questions ..................................... 30

2.1 Introduction

The core infrastructure of any modern Radiology department is made up of computers/computer workstations and the connectivity or networking capability between these devices. All transactions between modalities, PACS, scheduling, billing, dictation, and reporting systems are made possible through specialized computer programs or applications that are executed by computers. Computer systems are quite diverse and are often designed to augment a specific task, whether it is to support image reconstruction for a modality such as computed tomography (CT) or digital radiography (DR) or rapid image display as in PACS. Fundamentally, all computers are built around a similar base design with enhancements in specific areas to address certain needs such as rapid storage access and data transfer for file servers and improved video characteristics for PACS client display stations. The purpose of this chapter is to familiarize the reader with the fundamentals of computer architecture, networking, and computer applications.
2.2 Computers 101 – Hardware

2.2.1 Hardware Elements of Computers

- There are five core hardware components of the modern digital computer system: the central processing unit or CPU, memory, input devices, output devices, and a bus. While some components are given greater emphasis for a particular computer design (e.g., a faster CPU for computationally intensive tasks), virtually all types of computers have these five key components represented. Most of the hardware components in the modern digital computer are contained within small modular semiconductor packages (integrated circuits [ICs] or chip) that, in turn, contain millions of discrete components. Numerous ICs are interconnected on a large circuit board, frequently referred to as the motherboard. The motherboard is interfaced with other outside components (e.g., disk drives, power supply, keyboard, network, etc.) using specialized couplers that provide necessary power and connectivity to peripheral devices such as disk drives (storage), video displays, and keyboards.

- The central processing unit (CPU) or microprocessor is typically the largest integrated circuit on the motherboard and its role is to execute specific commands or instructions/machine code dictated by a computer program and orchestrate the movement of data and instructions through the entire computer system. Although the CPU is frequently personified as the “brain” of the computer, it has no innate “intelligence” or inherent ability to make decisions. The CPU’s strength is in its ability to process instructions and manipulate data at amazing speeds. In this regard, it is the perfect soldier; it follows all commands presented to it with blazing efficiency.

- The number of instructions that a CPU can perform per second is expressed as its clock speed. Typical personal computer CPUs can perform over 3 billion instructions per second or 3 gigahertz (3 GHz). Modern CPUs actually contain two to eight CPUs in one IC or chip (multi-core CPU). This provides unparalleled computational speed as each core shares the processing tasks formerly assigned to one CPU. While the strength of the CPU is in its ability to process instructions, it has limited capability to store data before or after execution. The CPU relies on physical memory to store this information and provides it to the CPU on demand.

Key Concept 2.1:
Core Computer Hardware Components
- CPU
- Memory
- Input devices
- Output devices
- Bus
• **Memory** is principally used to temporarily store data (and results) and applications or programs. In contrast to the CPU, a memory module has no capability to process instructions; instead memory is designed to reliably store large chunks of data and then release these data on command (often at the behest of the CPU). Physical memory can exist in **solid-state** form as an IC or as **physical media** (spinning disk, compact disk [CD], or digital versatile disk [DVD]). A solid-state memory module that can be erased and rewritten for unlimited number of times is generically referred to as **random access memory or RAM**.

• Memory that can only retain data with power applied is referred to as a **volatile memory** – most of the motherboard memory modules are of this type. These are rated by their **storage capacity** (given in megabytes or gigabytes), **access speed** (in nanoseconds), **data rate** (DDR2), and **configuration** (single or dual inline memory SIMM or DIMM).

• **Non-volatile memory** will retain data written to it until it is erased or overwritten. Examples include USB memory sticks and disk drives. Since the inherent speed of non-volatile memory is substantially slower than that of volatile memory, volatile RAM is typically employed on the motherboard to augment data processing.

• Some forms of memory are designed for specific tasks. **Video memory (VRAM)** is employed on video graphics cards to store graphical information to improve video display performance. A specialized form of high-performance memory is found on most CPUs to help efficiently buffer data that move in and out of the microprocessor core (**L2 cache memory**). 

• There are additional forms of computer memory that are classified simply as **storage**, principally because they are characterized by slower speed compared to solid-state memory and non-volatile characteristics (data persist indefinitely until erased/overwritten). These are made up of spinning media (disk drives, CDs, and DVDs) and linear media (tape).

• **On-line storage** refers to high-performance, non-removable media that requires no human or mechanical intervention to retrieve. Data on spinning hard disk arrays are an example of on-line storage. **Near-line storage** consists of removable media (e.g., tapes, CDs, or DVDs) that are made available through mechanical means such as a robotic tape or optical disk jukebox. The efficiency of data retrieval with a near-line system is dependant upon the mechanical speed of the robotic system and the queuing mechanism of the media. **Off-line storage** is removable media that requires human intervention to load and retrieve data. As a result, performance
is the lowest for off-line storage. While off-line storage is the least expensive storage strategy, it is otherwise quite inefficient and is therefore reserved for data that have a low probability for future use.

- **Input/output devices** are hardware extensions that allow humans (or other devices) to interact with a computer. Examples of input devices include the keyboard, touch screen, mouse, microphone, and camera. Typical output devices include the video display, printer, plotter, and speaker.

- Because the typical microprocessor can execute several billions of commands per second, it is highly dependant upon an efficient mechanism for delivering instructions and data to it. This requires that there is a well-orchestrated method for moving data between the motherboard components and the CPU. The **data bus** is the physical data chain built into the motherboard that allows for this efficient data transfer. This is supported by several ICs, known as the **chipset**, which coordinates uninterrupted data transfers through the bus. Multiple different designs have been developed; the most common in use today is peripheral component interconnect (PCI) and PCI-Express. The data bus is defined by a **data-width** (typically 32 or 64 bits), which specifies how much data are delivered across the bus per cycle and a **clock speed** (given in megahertz).

- Another key component to the typical computer motherboard is the **basic input/output system (BIOS)**. The BIOS is comprised of a non-erasable read only memory (ROM) chip that contains the minimal amount of software necessary to instruct the computer how to access the keyboard, mouse, display, disk drives, and communications ports.

- When the power is first applied to the computer, the motherboard relies on the BIOS to tell what additional components are available to the motherboard for input and output (e.g., disk drives, memory, keyboard, etc.). The motherboard “becomes aware” of what is available and how to access it, each and every time the computer is restarted.

- The BIOS also provides information to the motherboard on where to find the first piece of software to load during the startup process. The startup process is also known as the **boot process**. The first piece of software to load is usually a portion of the **operating system** that will coordinate the other software programs.
2.3 Computers 101 – Software

- Hardware can be seen and handled. Software, on the other hand, is a virtual concept. While we can handle the media that software is written on, we cannot actually “see” the software.
- The term “software” applies both to application programs and data.
- Software at its lowest level (the level at which it interacts with the CPU) consists of a long series of bits (ones and zeros). All data written to physical media, whether it is magnetic disk, USB stick, CD, DVD, or RAM memory is stored as an orderly series of bits. Eight-bit clusters of data form a byte of data.
- Software is divided into system software (or operating system) and application software – programs that help users to perform specific tasks and programming software (or development software) – programs that aid in the writing (i.e., coding) of other software.
- All software consists of individual procedures that command the computer to follow a precisely orchestrated series of instructions. The number of individual instructions specified in any one program varies depending upon the type and complexity of the software – from 10 to 100 million lines of code. (The Windows XP operating system, for example, contains approximately 40 million lines of code.)
- All computer software must be moved into storage (i.e., disk drive) or physical memory (RAM) before it can be executed by the microprocessor. The instructions are passed through a series of software layers where they ultimately reach the microprocessor. Each instruction causes the computer to perform one or more operations.

2.3.1 Computer Operating System

- The operating system (OS) is the underlying software that integrates the hardware with software applications. It is distinguished from the essential hardware components in that it consists entirely of software – millions of lines of machine commands that are understood and obeyed by the microprocessor. The OS actually consists of hundreds or thousands of individual programs that bundled together. Many of these individual programs are
designed to work cooperatively with each other. The OS is automatically executed each time the computer is started and it is the most important software component running on any computer. A modern computer cannot operate without an OS.

- Although the CPU is frequently personified as the “brain” of the computer, it is really the OS software and the CPU acting together that provides the underlying “intelligence” of system. The OS and the CPU are inexorably linked; therefore, the distinction between the software and the hardware is sometimes blurred.
- The OS is designed to automatically manage nearly every task (or process) including maintenance of the files on the disk, tracking input from peripheral devices like keyboards or network cards, displaying output on printers and video displays and control of memory allocation. Memory allocation is crucial for maintaining stability of the system because if two programs try to use the same area of memory, both programs will usually fail. Two of the most critical jobs of the OS are ensuring that programs do not unintentionally interfere with each other and maintaining security.
- A function paramount to the modern OS is the support of the graphical user interface (GUI). A GUI replaces typed computer commands with a graphical representation of the task (e.g., moving a file). This is accomplished by creating a visual representation of the computer file system (the desktop), icons, and windows and linking them to the movements of a pointing device such as a mouse or trackball.
- The OS also provides a foundation or software platform for all other software (application programs). Therefore, the choice of OS, to a large extent, determines which application software can be used on a particular system.
- There are a number of operating systems in use today. The most popular is the Windows OS (Microsoft, Redmond Washington) that runs on the majority of computers worldwide. Other choices include UNIX, Linux, DOS, and the Mac OS (Macintosh).
- A multiprocessing OS supports use of more than one CPU. A multitasking OS allows more than one program to run simultaneously. A multithreading OS allows different parts of a program to run concurrently and a multi-user OS supports two or more individuals to run programs concurrently on the same computer system.
- An OS may consist of hundreds (or even thousands) of small programs called drivers. Drivers enable software to interact with the ubiquitous hardware devices attached to the motherboard and between

**Key Concept 2.6: Drivers**

Drivers are small programs that enable the operating system and application programs to interact with each other and with peripheral hardware devices. They require periodic upgrades, especially when the OS changes.
components on the motherboard itself. In other instances, drivers allow one software component to safely interact with another piece of software.

- From the user perspective, the OS provides the framework that application software runs inside of. All application software runs *on top of* the OS that, in turn, is directly integrated to the hardware. In general, application software cannot interact directly with the hardware; it must work through the OS. The modern OS is intentionally designed to sustain itself automatically with minimal user interaction. The software that is designed to perform real work for users is the application software.

### 2.3.2 Application Software

- OS software is designed to run autonomously with little interaction from the individual user. The OS monitors all internal functions of the computer, maintains stability of the hardware components, and regulates the processing of data in the microprocessor. Application software is a program designed to do *real work* for a user. Application software does not supplant the base OS software. Instead, application software runs *on top of* the OS such that an application is written (or coded) to work with a specific OS. Examples of application software include a word processor or spreadsheet.

### 2.3.3 Low-Level Programming Language

- Low-level programming language is the software language that is directly understood by a microprocessor, and is termed *machine code* or *machine language*. Every CPU model has its own native machine code or *instruction set*. The instruction set consists of a limited number of relatively primitive tasks such as adding or subtracting data in specialized memory placeholders called registers, or moving data from one register to the next.

- Despite their enormous processing speed, the intrinsic mathematical capabilities of a microprocessor are quite limited; a CPU cannot perform simple multiplication or division on its own – it has to be *taught how to do it*. By stringing a series of machine codes together, more complex processing (e.g., multiplication) is possible.

- Both machine code and its symbolic representation (*assembly language*) are considered as low-level languages because they are the closest command analog to the actual functional details of the microprocessor. Low-level does not imply diminished quality or efficiency; in fact, programs written directly in machine code or assembly language are very efficient.
2.3.4 High-Level Programming Language

- Although low-level programming instructions produce efficient programs, programming in machine code or assembler is difficult, tedious, and very time consuming.
- High-level programming language is really an abstraction of machine code programming because it uses natural language elements instead of arcane numbers and abbreviations. This makes the process of programming simpler, intuitive, and more understandable to the human programmer.
- High-level programming is the foundation of most software development projects. There are many high-level languages in common use today. Some of the languages currently include C, C#, C++, BASIC, Pascal, Java, FORTRAN, COBOL, and others.
- Using high-level programming languages, programmers (or “coders”) type out individual lines of the source code for an application, using a development software program. The lines of the source code need to be translated into machine code before the program can be understood and tested on the microprocessor. This conversion process is known as compiling a program and the software that converts the source code to machine code is known as a compiler.
- Most development software platforms include one or more compilers. The compiler turns the source code into an executable program that is customized for the specific OS/microprocessor combination that the program was developed for.
- The compiler saves the programmer a substantial amount of time and effort by constructing the sequence of machine codes that accurately represents each source code command.
- Programmers must follow a tedious sequence of compiling, testing, identifying errors, correcting errors, re-coding, and re-compiling a program in a process known as debugging the program. The majority of time devoted to programming is spent on debugging the code.
- Scripting languages differ from compiled languages in that the source code is interpreted and converted into machine code at the time of execution – obviating the compiling process. The development process with scripted languages is typically more rapid than with compiled code; however, because scripting languages are interpreted at the time of execution, they are typically slower to execute. Therefore, scripted language is often reserved for smaller programs that are not computationally intensive. Scripting languages include Apple Script, Visual Basic (or VB) script, shell script, and JavaScript.

Key Concept 2.7: Low-Level Versus High-Level Programming Languages

A single print statement in high-level source code programming language might produce a thousand individual machine code commands once it is compiled.
2.4 Computer Networking

- A computer network is a group of two or more interconnected computers that are capable of sharing data and resources. Networking allows multiple independent users to share the same resources (i.e., applications and data) and work with these data simultaneously. Fast, reliable networks form the backbone of digital Radiology department and allow large quantities of imaging data to be efficiently transported between modalities, archives, and viewing stations.

- Computer networks can be classified on the basis of scale (i.e., size, complexity), scope, topology, architecture, and connection method. The most common network is the local area network (LAN). A LAN is characterized by serving computers in a small geographic area such as a home or an office.

- A network that is comprised of two or more LANs is termed a wide area network (WAN). Although the term is somewhat ambiguous, it is more commonly used to describe networks with a broad geographic coverage—metropolitan, regional, or national. The largest WAN is the public Internet, which is a global system of interconnected computer networks.

- A typical Radiology department network would consist of at least one LAN that may be interconnected to a larger WAN (e.g., hospital network).

- Connection of two or more networks (i.e., Internetworking) changes the scope of network resources to any computer on the network. An intranet is one or more networks that are under control of a single administrative authority. Access to any external or unregulated networks is either not provided or is limited to authorized users.

- An extranet is an internally managed network (intranet) that maintains limited connectivity to networks that are neither managed, owned, nor controlled by the same entity. An extranet is typically isolated from the public Internet with security measures such as firewalls that regulate connectivity to outside or unmanaged networks. Most hospitals and business organizations configure their internal network in this way.

- Many home networks (wireless or wired) are extranets that consist of a LAN with access provided to the public Internet (WAN) via an Internet service provider (ISP) (e.g., Comcast, AT&T, Verizon, etc.).

Further Reading 2.8: Networking


2.4.1 Physical (Hardware) Networking Components

- Basic physical components of a computer network include the network card, cabling, and a point of connection (e.g., hub, repeater, bridge, router, or network switch).
The network interface card (NIC) is the piece of computer hardware that provides the capability for a computer to communicate over a network. Every NIC possesses a unique number, its media access control (MAC) address. This number can be used to help route data to and from other computers.

The physical connection of the computer to the network is usually accomplished through specialized cabling that contains four pairs of simple copper wires (twisted pair) in a configuration known as category 5 or Cat5, or its enhanced version Cat5e. Cat5 cabling frequently terminates in special rectangle plastic connectors that resemble oversized telephone handset connectors.

Other forms of physical connection used less often include fiber optic cables (optical fiber) and wireless (802.11x). Fiber optic provides greater transmission capacity (bandwidth) than Cat5 and wireless affords greater access where physical connections are not readily available.

The term Ethernet describes the wiring and signaling schema for the NIC and the cabling between devices on the network.

2.4.2 Network Switches

The cornerstones of the computer network are switches, the devices that connect other devices together on the network. Switches vary in the degree of functionality by which they manage the data traffic that passes through them. The term switch is an imprecise term that refers to many types of network devices.

The simplest and most inexpensive of network switches is the network hub. The hub provides a simple and passive method for all computers connected to it to transmit and receive data to each other. Each computer network cable has an individual connection (port) to the hub. The hub creates a shared medium where only one computer can successfully transmit at a time and each computer (host) is responsible for the entire communication process.
The hub is a passive device. The hub merely replicates all messages to all hosts connected to it and does not have any capability to route messages to a specific destination. A network hub is the most basic and inefficient means of connectivity. For this reason, simple hubs are rarely used today.

The network bridge improves upon the design of the basic network hub by providing a level of active management of the communication between attached hosts. The bridge is capable of learning the MAC addresses of the connected host computers and will only send data destined for a specific host through the port associated with a unique MAC address. By routing the data stream to the intended recipient, switching creates a more efficient method for network transmission.

Since the bridge needs to examine all data sent through it, it creates some processing overhead that slows the data transmission rate. Bridges typically support data transmission rates of 10, 100, and 1000 megabits per second (Mb/s).

The network router offers yet another level of technical sophistication over the network bridge. Like the network bridge, a router is capable of examining the contents of the data passing through it and is able to discern the identity of the sender and the recipient. However, instead of relying on the value of the hardware NIC MAC address (which is fixed and not configurable), the router is capable of discerning data based upon a software configurable identifier known as the Internet protocol address (IP address).

The IP address is a configurable 32-bit numeric value (e.g., 192.123.456.789) that is used to uniquely identify devices and the networks to which they belong. Using this schema, a host that is accessible globally must have a unique IP address; however, a host that is hidden within a private network need not have a globally unique address (it only needs to be unique on the local network). This scheme allows for conservation of unique IP addresses.

The typical broadband network router used in home networking has additional features such as dynamic host control protocol (DHCP), network address translation (NAT), and a network firewall. These additional features provide a secure connection between the home LAN and the ISP WAN. The router using NAT serves as a proxy that allows multiple computers to share a single public Internet IP address. The broadband network router assigns each computer in the home network its own IP address that is only unique within the home network.

2.4.3 Network Protocols

In order for them to communicate effectively, each device must adhere to a specific set of rules for communication called network protocols. Networks are usually comprised of a heterogeneous group of devices of different make,
model, vintage, and performance. The most ubiquitous network protocol over Ethernet is the Internet protocol suite (IPS) or transmission control protocol/Internet protocol (TCP/IP).

- TCP/IP is a software abstraction of protocols and services necessary for the establishment of communication between two computers on a network. This network abstraction was set down by the International Organization for Standardization (OSI) and is referred to as the OSI network model. The model describes five to seven information layers that link computer software application to the hardware that must perform the actual transmission and receipt of data.

- The layers in the network OSI model rely upon protocols to regulate how information is passed up through and down the OSI stack.

- The Internet protocol suite defines a number of rules for establishment of communication between computers. In most instances, the connection is a one-to-one relationship. Two computers go through a negotiation process prior to making a connection. The negotiations include request and acceptance of an initial connection, the type of connection, the rate of transmission, data packet size, data acknowledgement as well as when and how to transmit missing data.

2.4.4 Data Packets

- Data transmitted over a network is broken up into multiple small discrete chunks or packets before being sent over the network by the NIC. Packet size is variable and is part of the “negotiations” when establishing a network connection with another computer.

- Since a network segment can only be used by a single computer at any one instant and the physical parts of the network (i.e., cabling and switches) are shared by many computers, splitting data streams up into smaller parcels in a shared network model improves network efficiency dramatically.

- Switching and assigning resources on a shared network is a complex process – one which needs to occur in the order of microseconds to maintain efficient communication between thousands of devices that are potentially competing for these resources. Despite the refined sophistication of the system, there are instances where two or more computers attempt to send data along the same segment simultaneously. This phenomenon is termed a collision. Optimum network design mandates minimizing collisions and maximizing collision detection to maintain fidelity of data transmission.

- Additional metadata is automatically married to each data packet based upon protocols specified in by IPS and contains information such as the data type, packet number, total number of packets as well as the IP address of the...
sender and receiver. This is analogous to placing a letter (packet) in an envelope with delivery information (sender and return address). Data packets with this additional data wrapper are referred to as data frames.

- Since each frame of transmitted data contains information about where it originated and where it is supposed to go, routers can then examine each packet and forward it through the relevant port that corresponds to the recipient. Moreover, since each packet is self-contained and auto-routable, packets from a single message can travel over completely different routes to arrive at the same destination. Routers instantaneously analyze and balance network traffic and will route packets over segments that are currently under a lighter load.

- At the receiving end, the OSI model also details how to reassemble the individual packets back into the original file. Each packet bears both an identifier and sequential number that specify what part of the original file each packet contains. The destination computer uses this information to re-create the original file. If packets are lost during the transmission process, TCP/IP also has methods for requesting re-transmission of missing or corrupt packets.

- Network bandwidth is defined as the rate at which information can be transmitted per second (bits/s). This can vary tremendously depending upon the type of physical connection, switches, and medium (i.e., cabling versus fiber versus wireless). Theoretical bandwidth of Ethernet, for example, varies from 10 to 1000 Mb/s. Another technology, known as asynchronous transfer mode (ATM) can support bandwidths ranging from 155 Mb/s (OC3), 622 Mb/s (OC12) to 2488 Mb/s (OC48).

Key Concept 2.12: Data Packets
Since each packet is self-contained and auto-routable, different packets from a single message can travel over completely different routes to arrive at the same destination.

Checklist 2.13: Theoretical Bandwidths
- Ethernet 10 Mbits/s
- Ethernet 100 Mbits/s
- ATM 155 Mbits/s (OC3)
- ATM 622 Mbits/s (OC12)
- Ethernet 1000 Mbits/s
- ATM 2488 Mbits/s (OC48)

Key Concept 2.14: Theoretical Bandwidth
In general, actual bandwidth is approximately one-half of theoretical values.
It is important to recognize that there can be a substantial difference between the values of a theoretical bandwidth and actual bandwidth. While packets of data move at the speed of light, other factors such as quality of cabling and efficiency of network switches contribute to network overhead that can impede actual performance.

2.5 Client–Server Architecture

- The client–server computing model is one of interdependency between two or more computers where one computer provides data or services to the other.
- Early networks were used primarily to backup data to a central location during off-hours. As technology has continued to evolve, there has been a growing convergence of desktop computing and network computing. In the past, maximizing computing efficiency required application software and data to reside on the client computer. A fat client (thick or rich client) is a host application that performs the bulk of data processing operations for the user with minimal to no reliance on network resources.
- By leveraging the power of faster network services, real-time transfer of data and application resources to the client desktop computer is afforded. The client makes requests from a dedicated, powerful networked host computer (a server) that stands ready to provide application services or data to the client over the network.
- While any computer can be configured to act as a server, most servers have additional hardware capacity to support the increased demands of multiple simultaneous users (i.e., faster multi-core CPUs, large memory stores, and large hard drives).
- This close interrelationship of multiple clients and a server is known as client–server architecture. Almost the entire structure of the Internet is based upon the client–server model. This infrastructure supports delivery of web pages over the World Wide Web and e-mail. The most basic client application is the web browser, which interacts directly with the server to render web pages.

**Definition 2.15: Server–Client**
A server is a computer that provides application services or data. A client is a computer or software application that receives those services and data.

**Key Concept 2.16: Client–Server Architecture**
In its purest form, client–server architecture concentrates on maximizing virtually all of the computational power on the server while minimizing the computational requirements of the client stations. This affords great economies of scale without loss of functionality.
data, images, or advanced visualizations. Any application that is accessed via a web browser over a network that is coded in a browser-supported language (i.e., JavaScript, Active Server Pages – ASP, Java, HTML, etc.) are called web applications or webapps.

- A **thin client** (lean or slim client) is an application that relies primarily on the server for processing and focuses principally on conveying input and output between the user and the server.
- The term thin-client application is often misused by industry to refer to any function or application that runs within a web browser – however, this is an incomplete definition. Even if the application is used inside of a web browser, if additional software or browser plug-ins are required or local data processing occurs, the term **hybrid-client** is more appropriate. Most PACS client viewing software that runs within a web browser is classified as hybrid-client.
- Modern PACS systems are designed to leverage this configuration where the majority of the image management is controlled by a powerful central server that responds to multiple simultaneous requests for image data from relatively inexpensive, less-powerful client viewing stations.
- Software applications that are designed to operate principally over a network in a client–server configuration are grouped collectively into something known as **web services**. There are established profiles and specifications that define how these services are supposed to interoperate with service providers and service requesters. Web services differ from web applications in that web services need not run inside a browser or be constructed with web elements.

### 2.6 Database Applications

- Many useful web services and web applications provide direct access to databases.
- There are a number of **database models**; however, the **relational model** is used most often. In the relational model, data are abstracted into **tables** with rows and columns. Each row is an individual **record** and each column is a separate **attribute or field** for each record. One or more tables are linked logically by a common

---

**Definition 2.17:** Thin Client

A software application that does not depend upon any additional software components and does not perform any processing on the local host.

**Definition 2.18:** Database

A structured collection of data. Data that are housed in a database are more amenable to analysis and organization. Databases are ubiquitous and are the essential component of nearly every computer application that manages information.
attribute (e.g., an order number, serial number, accession number, etc.).

- Databases also support an **indexing** mechanism that confers greater speed to the system when accessing or updating data. Indexing comes at some cost since it adds some processing overhead to the system.

- The most common programmatic operations on a relational database include reading or selecting records for analysis, adding records, updating records, or deleting records.

- Structured query language (**SQL**) is a database-specific computer language designed to retrieve and manage data in relational database management systems (**RDMS**). SQL provides a programmatic interface to databases from virtually any development platform.

- Databases are integral to the infrastructure of most business systems including information systems in Radiology. Virtually, every aspect of Radiology services is tied to relational database functions from patient scheduling to transcription.

---

**Pearls 2.19**

- Although the microprocessor is frequently personified as the “brain” of the computer, it has no innate “intelligence” or inherent ability to make decisions. The microprocessor’s strength is in its ability to process instructions and manipulate data at amazing speeds.

- All application software runs on top of the OS that, in turn, is directly integrated to the hardware. In general, application software cannot interact directly with the hardware; all interactions are brokered by the OS.

- A computer that is accessible globally must have a unique IP address; however, a computer that is hidden within a private network need not have a globally unique address (it only needs to be unique on the local network). This scheme allows for conservation of unique IP addresses.

- A thin client (lean or slim client) is an application that relies primarily on the server for processing and focuses principally on conveying input and output between the user and the server.

- Software applications that are designed to operate principally over a network in a client–server configuration are grouped collectively into something known as web services.

---

**Self-Assessment Questions**

1. The core hardware components of a digital computer include everything except
   
   a. Microprocessor
   b. Memory
   c. Bus
   d. Keyboard
   e. Operating system
2. Volatile memory is distinguished from non-volatile memory by
   a. Poorer performance of volatile memory
   b. Flammability of volatile memory
   c. Inability of volatile memory to retain data with power loss
   d. Greater expense of volatile memory
   e. None of the above

3. Which is not true about storage?
   a. On-line storage is readily available
   b. Near-line storage requires human intervention
   c. Off-line storage is not accessible by robotic devices
   d. Data are stored on media such as tape, compact disk or DVD
   e. None of the above

4. Which is the best statement regarding the motherboard data bus?
   a. It connects to the keyboard
   b. It connects to the power supply
   c. It interconnects the components on the motherboard
   d. It connects to the disk drive
   e. None of the above

5. What is the fundamental distinction between software and hardware?
   a. Price
   b. Hardware is a physical entity
   c. Packaging
   d. Complexity
   e. None of the above

6. The purpose of the operating system (OS) is
   a. To manage memory allocations
   b. To copy files to disk
   c. To manage the user interface
   d. To manage computer resources
   e. All of the above

7. Computer drivers are
   a. Names for a specific type of golf club
   b. Large programs that take control of the OS
   c. Small programs that provide a bridge or interface between hardware and software
   d. Similar to computer viruses
   e. None of the above

8. Low-level programming languages are (best answer possible)
a. Fairly simple to learn and use
b. Are primarily used by human computer programmers to create applications
c. Are not as costly as high-level programming languages
d. Are used primarily by the CPU
e. All of the above

9. The most complex network switch is the
   a. Network hub
   b. Network router
   c. Network bridge
   d. They are all similar in complexity
   e. Not listed

10. Which is true of thin-client applications?
    a. They require a web browser to run
    b. They do not need additional software
    c. They require a networked server
    d. They require an internal database
    e. All of the above
Practical Imaging Informatics
Foundations and Applications for PACS Professionals (Ed.)
2010, XVIII, 456 p. 27 illus. in color., Softcover
ISBN: 978-1-4419-0483-6