The OSI Reference Model—Review of the seven layers of tasks that make communications systems operate.

Types of Internetworking Devices—The main devices on an internetwork: bridges, switches, routers, and access servers.

An Internetwork Example—A specific internetwork topology that is used as an example throughout the book.

CHAPTER 1

Getting Started in Internetworking

This chapter helps you start learning about internetworking. Understanding this complex topic is the first step toward understanding the Cisco Internetwork Operating System (IOS). The IOS provides the intelligence that Cisco products require to perform their various internetworking tasks. The IOS is an operating system with a proprietary user interface, command set, configuration syntax, and so on. The IOS is to Cisco devices as Windows 2000 is to IBM-compatible personal computers. The IOS runs on all the Cisco products discussed in this text.

We encourage you to have a firm grasp of the internetworking principles surveyed in this chapter before you attempt to understand the complexities of the Cisco IOS. *Internetworking* is a term used to describe the collection of protocols and devices that interoperate on data networks. This chapter gives you the basic understanding of the subject; it is not meant to give you comprehensive coverage of the subject (which could take multiple books to cover completely). If you need a more extensive introduction to internetworking, a few good texts are cited in the "References" section at the end of this chapter.

When you finish this chapter, you should be comfortable with the OSI networking model and have a basic understanding of how bridges, switches, routers, and access servers work. Chapter 2, "The Basics of Device Configuration," introduces you to the basics of configuring a Cisco device.

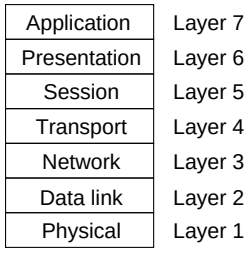
The OSI Reference Model

The Open System Interconnection (OSI) reference model is a principle of internetworking that you must understand to appreciate the way Cisco devices operate. The OSI reference model is a seven-layer architectural model developed by the International Organization for Standardization (ISO) and the International Telecommunications Union-Telecommunications (ITU-T). It is used universally to help individuals understand network functionality. The OSI reference model adds structure to the many complexities involved in the development of communications software. The development of communications software involves many tasks, including dealing with multiple types of applications, transmission strategies, and physical network properties. Without structure, communications software might be difficult to write, change, and support.

NOTE ISO is an international organization founded to promote cooperation in technological developments, particularly in the field of communications. ITU-T, on the other hand, is a global organization that drafts standards for all areas of international analog and digital communications. ITU-T deals with telecommunications standards.

The OSI reference model is divided into seven distinct layers. Each layer performs a specific, distinct task that helps communications systems operate. The layer operates according to a set of rules, which is called a *protocol*. In addition to following the rules of the protocol, each layer provides a set of services to the other layers in the model. The seven layers of the OSI reference model are the application, presentation, session, transport, network, data link, and physical layers, as shown in Figure 1-1. In the following sections, we briefly review each layer, starting with the application layer.

Figure 1-1 *The OSI Reference Model Contains Seven Layers*



The Application Layer

The application layer provides the interface to the communications system, which the user sees. Many common applications are used today in an internetwork environment, such as web browsers, File Transfer Protocol (FTP) clients, and electronic mail. An example of application layer communication is a web browser downloading a document from a web server. The web browser and server are peer applications on the application layer that communicate directly with each other for the retrieval of the document. They are unaware of the six lower layers of the OSI reference model, which are working to produce the necessary communications.

The Presentation Layer

The presentation layer deals with the syntax of data as it is being transferred between two communicating applications. The presentation layer provides a mechanism to convey the desired presentation of data between applications. Many people infer that the look and feel of the environment of a computer desktop, such as the way all the applications look and

interact uniformly on a computer by Apple Computer, Inc., is an example of a presentation layer. In fact, this is not a presentation layer, but a series of applications using a common programmer's interface. One common presentation layer in use today is Abstract Syntax Notation One (ASN.1), which is used by protocols such as the Simple Network Management Protocol (SNMP) to represent the structure of objects in network management databases.

The Session Layer

The session layer allows two applications to synchronize their communications and exchange data. This layer breaks the communication between two systems into dialogue units and provides major and minor synchronization points during that communication. For example, a large distributed database transaction between multiple systems might use session layer protocols to ensure that the transaction is progressing at the same rate on each system.

The Transport Layer

The transport layer, Layer 4, is responsible for the transfer of data between two session layer entities. Multiple classes of transport layer protocols exist, from those that provide basic transfer mechanisms (such as unreliable services) to those that ensure that the sequence of data arriving at the destination is in the proper order, that multiplex multiple streams of data, that provide a flow control mechanism, and that ensure reliability.

As you will see in the next section, some network layer protocols, called connectionless protocols, do not guarantee that the data arrives at the destination in the order in which it was sent by the source. Some transport layers handle this by sequencing the data properly before handing it to the session layer. *Multiplexing* of data means that the transport layer can simultaneously handle multiple streams of data (which could be from different applications) between two systems. *Flow control* is a mechanism that the transport layer can use to regulate the amount of data sent from the source to the destination. Transport layer protocols often add reliability to a session by having the destination system send acknowledgments back to the source system as it receives data.

In this text, we discuss the three commonly used transport protocols: the Transmission Control Protocol (TCP) that is used on the Internet, Novell's Streams Packet Exchange (SPX), and Apple's AppleTalk Transport Protocol (ATP).

The Network Layer

The network layer, which routes data from one system to another, provides addressing for use on the internetwork. The Internet Protocol (IP) defines the global addressing for the Internet; Novell defines proprietary addressing for the Internetwork Packet Exchange

(IPX), its client/server architecture; and Apple's AppleTalk uses the Datagram Delivery Protocol (DDP) and proprietary addressing for communicating between its machines on the network layer. In later chapters, we explore the specifics of each of these types of network layer addresses.

Network layer protocols route data from the source to the destination and fall into one of two classes, connection-oriented or connectionless. Connection-oriented network layers route data in a manner similar to using a telephone. They begin communicating by placing a call or establishing a route from the source to the destination. They send data down the given route sequentially and then end the call or close the communication. Connectionless network protocols, which send data that has complete addressing information in each packet, operate like the postal system. Each letter, or packet, has a source and a destination address. Each intermediate post office, or network device, reads this addressing and makes a separate decision on how to route the data. The letter, or data, continues from one intermediate device to another until it reaches the destination. Connectionless network protocols do not guarantee that packets arrive at the destination in the same order in which they were sent. Transport protocols are responsible for the sequencing of the data into the proper order for connectionless network protocols.

The Data Link Layer

Layer 2, the data link layer, provides the connection from the physical network to the network layer, thereby enabling the reliable flow of data across the network. Ethernet, Fast Ethernet, Token Ring, Frame Relay, and Asynchronous Transfer Mode (ATM) are all Layer 2 protocols that are commonly used today. As you will see throughout this text, data link layer addressing is different from network layer addressing. Data link layer addresses are unique to each data link logical segment, while network layer addressing is used throughout the internetwork.

The Physical Layer

The first layer of the OSI reference model is the physical layer. The physical layer is concerned with the physical, electrical, and mechanical interfaces between two systems. The physical layer defines the properties of the network medium, such as fiber, twisted-pair copper, coaxial copper, satellite, and so on. Standard network interface types found on the physical layer include V.35, RS-232C, RJ-11, RJ-45, AUI, and BNC connectors.

NOTE Many people add an eighth layer to the top of the OSI reference model, the political layer. Although used in jest, the term *political layer* is often accurate because all lower layers of the OSI reference model are encapsulated within the politics involved in the organizations that design a data network.

The Data Exchange Process

These seven layers all work together to provide a communications system. The communication occurs when a protocol on one system, which is located at a given layer of the model, communicates directly with its corresponding layer on another system. The application layer of a source system logically communicates with the application layer of the destination system. The presentation layer of the source system passes data to the presentation layer of the destination system. This communication occurs at each of the seven layers of the model.

This logical communication between corresponding layers of the protocol stack does not involve many different physical connections between the two communications systems. The information each protocol wants to send is encapsulated in the layer of protocol information beneath it. The encapsulation process produces a set of data called a *packet*.

NOTE Data encapsulation is the process in which the information in one protocol is wrapped, or contained, in the data section of another protocol. In the OSI reference model, each layer encapsulates the layer immediately above it as the data flows down the protocol stack.

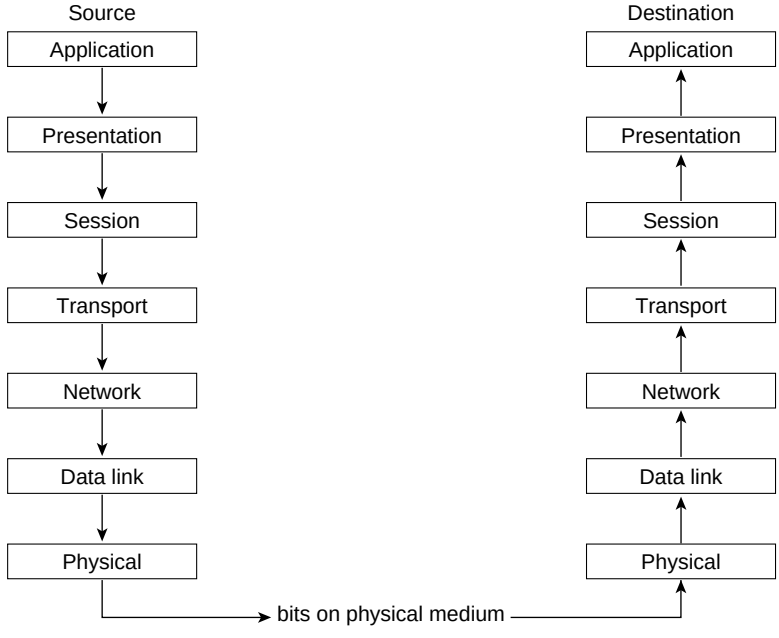
Starting at the source, as shown in Figure 1-2, the application-specific data is encapsulated in the presentation layer information. To the presentation layer, the application data is generic data being presented. The presentation layer hands its data to the session layer, which attempts to keep the session synchronized. The session layer passes data to the transport layer, which transports the data from the source system to the destination system. The network layer adds routing and addressing information to the packet and passes it to the data link layer. The data link layer provides framing for the packet and the connection to the physical layer.

At Layer 1, as shown in the figure, the physical layer sends the data as bits across a medium, such as copper or fiber. The packet then traverses the destination network from Layer 1 to Layer 7. Each device along the way reads only the information necessary to get the data from the source to the destination. Each protocol de-encapsulates the packet data and reads the information sent by the corresponding layer on the source system.

As an example, consider what occurs when you open a Web page using a Web browser. Given a URL, such as www.telegis.net, your browser asks the TCP to open a reliable connection to the Web server that is located at www.telegis.net. (Many applications that use TCP skip the presentation and session layers, as we do in this example.) TCP then requests the network layer (IP) to route a packet from the source IP address to the destination IP address. The data link layer takes this IP packet and encapsulates it again for the particular type of data link leaving the source system, such as Ethernet. The physical layer carries the signal from the source system to the next system en route to the destination, such as a router.

The router de-encapsulates the data link layer; reads the network layer information; re-encapsulates the packet, if necessary, to place it on the next data link en route to the destination; and routes the packet appropriately.

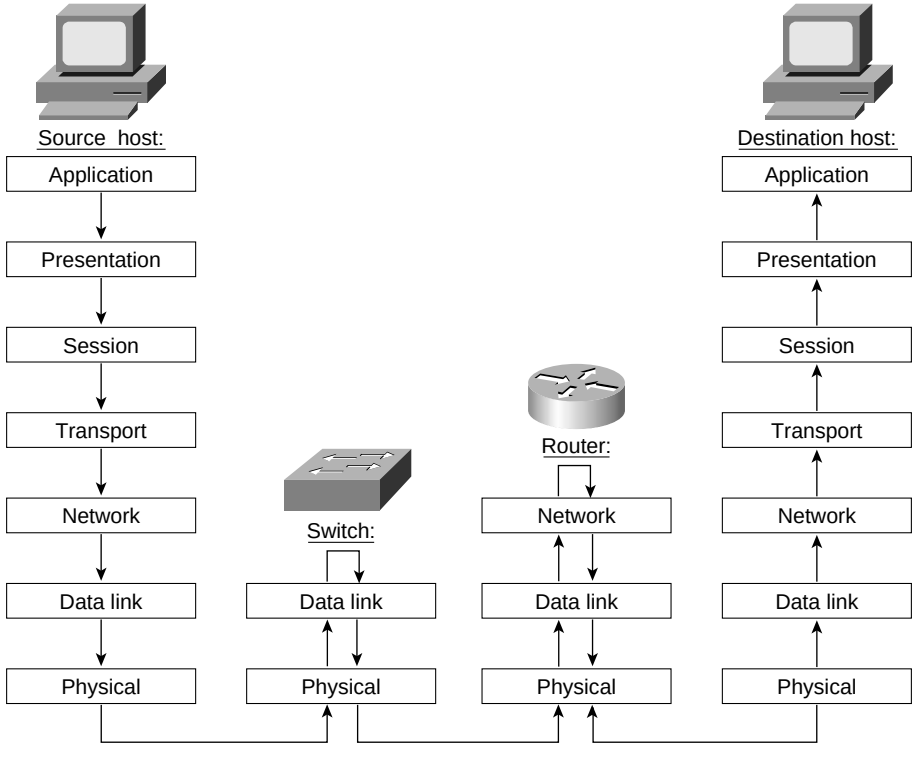
Figure 1-2 *Data Flow from a Source Application to a Destination Application Through the Seven Layers of the OSI Reference Model*



This process continues until the packet reaches the destination IP address. At the destination IP address, the data link layer de-encapsulates the packet, sees that the destination IP address is the local system, and passes the data in the IP packet to the transport layer. The transport layer ensures the reliability of the connection and passes the data from your Web browser to the www.telegis.net Web server. The Web server then responds to your Web browser request and sends a Web page of data back to your browser (using the same process, but with the source and destination IP addresses reversed).

Cisco devices covered in this book operate at the physical, data link, and network layers of the OSI reference model and read information in these layers to carry data from one location to another. Throughout this book, we reference these layers and explain how the Cisco IOS uses the protocol information at each layer. Some Cisco devices, such as bridges and switches, operate at the data link layer. Other Cisco devices, such as routers, operate at the network layer, as shown in Figure 1-3. We describe the various types of internetworking devices in the next section.

Figure 1-3 *An OSI Reference Model Depiction of Data That Travels from a Source Host, Through a Cisco Switch, Through a Cisco Router, and Then to a Destination Host*



Types of Internetworking Devices

Cisco devices fall into three main categories: bridges and switches, routers, and access servers. We discuss bridges and switches first.

Bridges and Switches

A *bridge* is a network device that operates at the data link layer. A bridge connects multiple data link layer network segments into a single logical network segment. There are many different types of bridges:

- Transparent or learning
- Encapsulation
- Translational

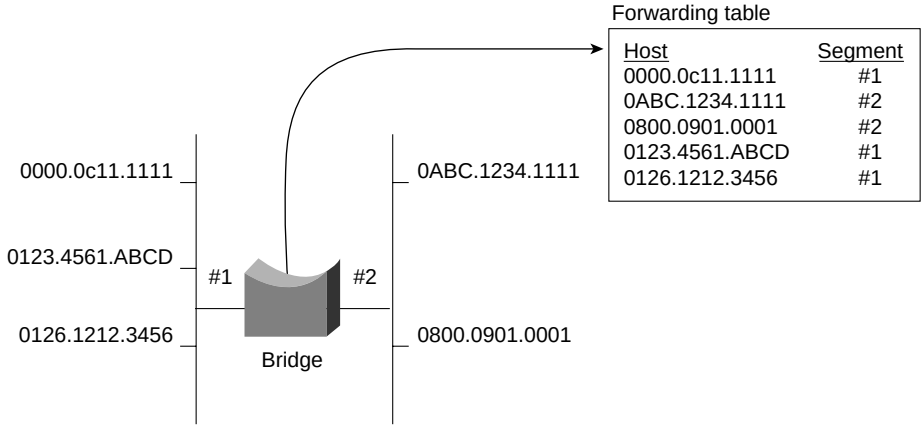
- Source-route
- Source-route translational

Although the Cisco IOS implements each of these types of bridging, we discuss only the first three types of bridging in this book. Source-route and source-route translational bridging are used in Token Ring environments.

Bridging allows for physical and logical separation of traffic when necessary to reduce traffic loads on a network segment. The main advantage of bridging is to ensure network reliability, availability, scalability, and manageability by segmenting a logical network into multiple physical pieces. We examine bridging as it relates to routing throughout this text.

A bridge performs its function by examining the data link layer information in each packet and forwarding the packet to other physical segments only if necessary. The information concerning which packets to forward to which network segments is learned by the bridge and kept in a forwarding table. The forwarding table includes a list of known data link layer addresses and the associated network segment where these devices are believed to exist, as shown in Figure 1-4.

Figure 1-4 *The Forwarding Table Maps Data Link Addresses to Physical Network Segments*



Bridges communicate with one another to determine the best method of forwarding packets to a given data link layer destination using a Spanning Tree Protocol. This protocol allows bridges to build a loop-free topology over which to forward packets. A *loop-free topology*, a topology that guarantees that a packet reaches every segment of a network exactly once, is needed in a bridging environment to avoid broadcast storms and to avoid multiple parallel bridges forwarding a packet multiple times to a given segment. A *broadcast storm* is a network segment event in which a *broadcast packet*—that is, a packet meant for every station on the segment—is sent in a continual loop until the segment is overloaded with traffic.

The simplest form of a bridge, a *transparent bridge*, can handle the connection of only like data link layer protocols. *Encapsulation* and *translational bridges* can be considered transparent bridges, with the additional functionality of enabling different data link layer protocols to interoperate.

An encapsulation bridge encapsulates an entire data link layer frame in another data link layer, which allows transparent bridging between like data link layers to occur when they are physically separated by a second, different data link layer. For example, two encapsulation bridges, each with one Ethernet port and one serial port, can bridge Ethernet network segments when they are connected by a serial link. The serial link is a different Layer 2 medium than is Ethernet. Encapsulation bridging allows the entire Ethernet frame to be bridged from one segment to another when separated by the serial link because the bridge encapsulates the Ethernet frame in the serial link data link protocol. The result is that the devices on the two Ethernet segments that are joined by the encapsulation bridges believe that all the devices are attached to a single, logical Ethernet segment.

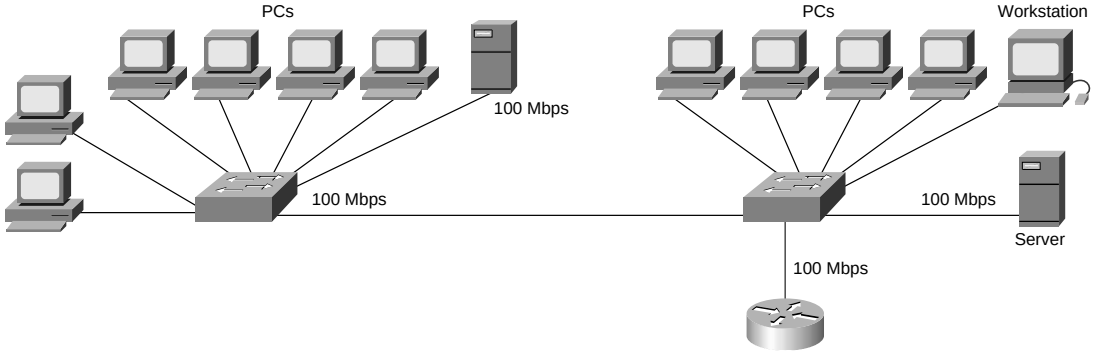
Another type of bridge is a translational bridge. A translational bridge performs the function of a transparent bridge between different types of data link layer protocols. For example, a translational bridge may translate Ethernet frames into Token Ring frames on the data link layer. If two devices are on different mediums connected by a translational bridge, they appear to be on one logical network segment. The transparent interconnection of two different mediums can provide the necessary connectivity for two devices that need to communicate solely at the data link layer.

A Cisco *switch* is essentially a multiport bridge that runs the IOS. A switch, which functions at the data link layer, performs the same basic functions as a bridge. The essential difference between a bridge and a switch is not technical, but packaging.

A switch may have more ports than a bridge, cost less per port than a bridge, and possess embedded management functions that a bridge does not have. Yet, when you examine the functionality of bridges and switches within the context of the OSI reference model, they do not differ. Many switches have multiple ports supporting a single data link layer protocol, such as Ethernet, and a smaller number of high-speed data link layer ports used to connect to faster mediums, such as ATM or Fast Ethernet. If a switch has two or more different interfaces to two or more data link layer protocols, it can be considered a translational bridge. Many switches today have interfaces that operate at multiple speeds, such as Ethernet, Fast Ethernet, and Gigabit Ethernet.

Figure 1-5 shows a small switched internetwork.

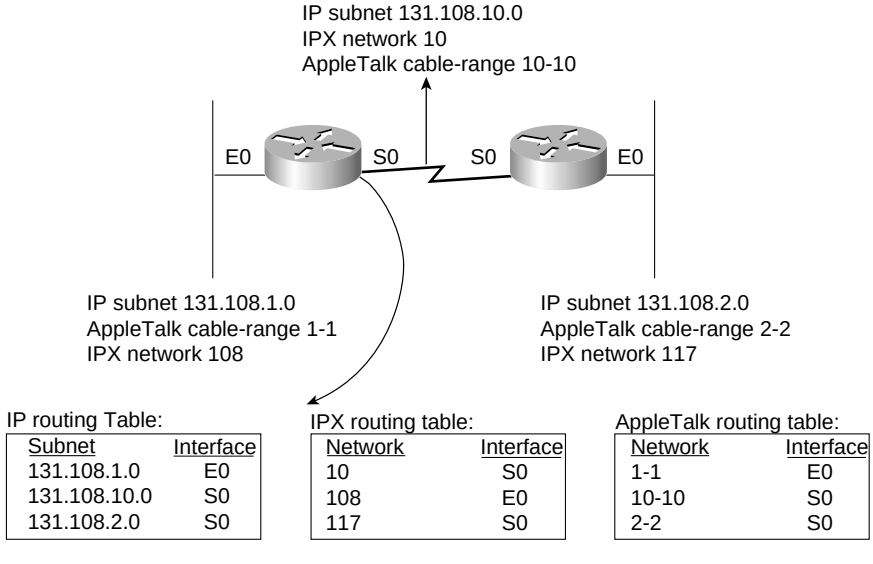
Figure 1-5 *A Small Switched Internetwork*



Routers

A *router* is a device that directs packets through the network based on network layer information. We focus on three network layer protocols in this book: IP, IPX, and AppleTalk. A router understands the network layer addressing in a packet and it has algorithms, called routing protocols, that build tables to determine the route that a packet should take to reach its final destination. For a multiprotocol router—one that understands multiple network layer addressing formats and routing protocols, such as a Cisco router—the router keeps a separate routing table for each network layer protocol that is being routed, as shown in Figure 1-6.

Figure 1-6 *A Multiprotocol Router Keeps a Routing Table for Each of Its Network Layer Protocols*



A bridge or switch connects two or more physical networks into a single logical network, while a router connects two or more logical networks and routes between them using information that is built by routing protocols and kept in routing tables. The advantages of a router (as compared to using any type of bridge) are that it physically and logically breaks a network into multiple manageable pieces, allows for control of routed packets, and routes many different network layer protocols at the same time. In this book, we discuss many router configuration options in the Cisco IOS.

Access Servers

An *access server*, also called a *communications server*, is a device that connects asynchronous devices to a network. A common application of an access server is to connect a computer communicating over a modem to the Internet. The access server combines the functions of a router with the functions of an asynchronous protocol.

If a machine connects to an access server via an asynchronous interface, the access server provides the software that allows the machine to appear to be on the network. For example, an access server may have 16 asynchronous ports and a single Ethernet port. Any device that connects to an asynchronous port appears to be on the Ethernet where the access server resides, which allows people running IP, IPX, or AppleTalk to work from a remote machine, just as they would if they were on the local network. We discuss the configuration and functions of access servers throughout this book.

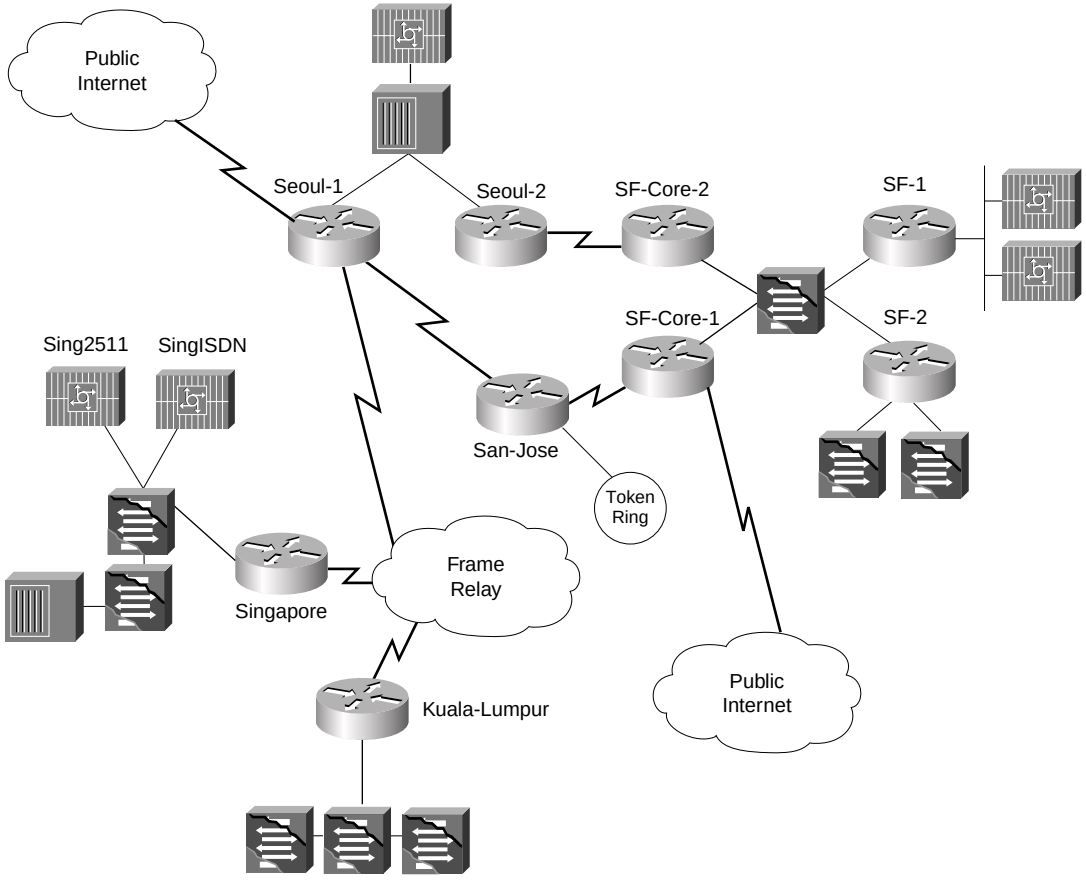
An Internetwork Example

Figure 1-7 shows the network that we use as an example throughout this book.

This network is used to examine the use of the Cisco IOS in the following environments:

- Various local-area network (LAN) technologies, such as Ethernet, Fast Ethernet, Gigabit Ethernet, Token Ring, and the Fiber Distributed Data Interface (FDDI). (See Chapter 3, “The Basics of Device Interfaces.”)
- Various synchronous and asynchronous wide-area network (WAN) technologies, such as HDLC, PPP, Frame Relay, ATM, and ISDN. (See Chapter 3.)
- IP routing. (See Chapter 4, “TCP/IP Basics.”)
- IPX routing. (See Chapter 6, “IPX Basics.”)
- AppleTalk routing. (See Chapter 5, “AppleTalk Basics.”)

Figure 1-7 *The Zoom Integrated Products Internetwork*



This network belongs to a fictitious company named Zoom Integrated Products (ZIP). ZIP, which has its corporate offices in San Francisco, California, makes components for the semiconductor industry. Its Asian sales headquarters are located in Seoul, Korea. Both the corporate offices and the Asian sales headquarters have connections to the public Internet. ZIP also has manufacturing facilities in Singapore and Kuala Lumpur, Malaysia.

The ZIP network uses Frame Relay to connect Singapore and Kuala Lumpur to Seoul. Seoul has ISDN BRI dialup facilities. At its corporate offices, the ZIP network has a

Gigabit Ethernet backbone and three Fast Ethernet network segments—two for high-speed connections to office suites and one for a LAN, where access servers reside for corporate dialup users. There are additional access servers for local dialup users in Seoul and Singapore. The corporate offices are connected to its sales headquarters via redundant HDLC links. A manufacturing assembly facility, which is located in San Jose, California, has dual HDLC links—one to the corporate offices and one to the sales headquarters in Seoul. The San Jose facility uses a Token Ring network on the assembly floor.

ZIP uses a variety of internetwork protocols on its network, including AppleTalk, IP, and IPX. Cisco switches are used for desktop connectivity, and routers interconnect each site and each location. (Each router is identified by name in Figure 1-7.) Most locations have at least one access server for remote dialup users.

The ZIP internetwork is representative of many internetworks throughout the world in that it uses multiple network layer protocols and wide-area network protocols, uses a combination of routing and switching, and has access servers to handle connections from asynchronous devices. Although it is only an example, this network and its complexities are typical of internetwork deployment today. As we progress through this book, we will use the ZIP network as an example and show you how to configure all the Cisco IOS devices necessary to make this fictitious network a reality.

Summary

Having completed this chapter, you should be comfortable with the OSI networking model and should have a basic understanding of how bridges and switches, routers, and access servers work. Next, Chapter 2 introduces you to the basics of configuring a Cisco device. Keep in mind the following central concepts from this chapter:

- The Cisco IOS is the operating system that runs Cisco devices.
- Cisco devices covered in this book operate at three layers of the OSI reference model: physical, data link, and network.
- The Cisco IOS uses protocol information at each layer of the OSI reference model.
- Bridges and switches operate at the data link layer and connect multiple data link layer network segments into a single logical network segment.
- Routers operate at the network layer and direct packets through the network based on network layer information.
- Access servers connect asynchronous devices to a network, allowing the device to appear to be on the network.

References

The following references explore the subjects in this chapter further:

Halsall, F. *Data Communications, Computer Networks, and Open Systems,* Fourth Edition. Reading, Massachusetts: Addison-Wesley Publishing Company, 1996.

Perlman, R. *Interconnections: Bridges, Routers, Switches and Internetworking Protocols,* Second Edition. Reading, Massachusetts: Addison-Wesley Publishing Company, 1999.

Peterson, L. and B.S. Davie. *Computer Networks: A Systems Approach,* Second Edition. San Francisco, California: Morgan Kaufmann Publishers, 1999.