Chapter 4

A Graphical Introduction to the Structural Elements of Cyberspace

Elihu Zimet and Edward Skoudis

THE MAJORITY of Internet users do not have (or need) much understanding of what is going on "under the hood" of cyberspace, beyond use of their keyboard, display, and favorite applications. To further the understanding of other chapters of this book by readers without a technical background, this chapter explains various elements of cyberspace, particularly the Internet, with a graphical presentation.

The purpose here is to shed light on elements of cyberspace by example, so there is little attempt to be comprehensive. The focus is on Internet technology, particularly networks using the Internet protocol (IP). However, because legacy telephone networks, cellular networks, and cable networks, as well as private enterprise, government, and military networks, increasingly use Internet technology, we discuss them briefly as well.

The structure of cyberspace will change as new technology is developed and employed in a variety of new applications and social structures. Thus, the representation here reflects a snapshot of the present only. Figure 4–1 presents the domains of cyberspace addressed in this chapter.

Figure 4-1 Domains of Cyberspace

The *systems domain* comprises the technical foundation, infrastructure, and architecture of cyberspace. It includes hardware and software, as well as the infrastructure items supporting them, such as the electrical power grid.

The *content and application domain* contains both the information base that resides in cyberspace and the mechanisms for accessing and processing this information.

Communications among people and interactions between people and information occur in the *people and social domain*. Businesses, consumers, advocacy groups, political campaigns, and social movements are in this domain.

The *governance domain* overlays all of the aspects of cyberspace, including the technological specifications for the systems domain, the conventions for data formatting and exchange in the content and application domain, and the legal frameworks of various countries associated with the people and social domain. This chapter focuses on the first three domains defined above (the governance domain is described in chapter 21 of this book, "Internet

Governance”).

The Systems Domain

The systems domain of cyberspace is the infrastructure that carries, stores, and manipulates information. Hundreds of millions of computer and network systems interact in cyberspace, carrying information generated by over a billion people. A major portion of the modern economy is associated with manufacturing the components and systems of cyberspace, including computer chips, desktop computers, routers, servers, and operating systems. Another major component of the economy is associated with operating this infrastructure, including Internet service providers (ISPs), telecommunications firms, electrical power companies, and other organizations.

The global open communications backbone—including telecommunications, cable, cellular, Internet, and other public networks—is the principal infrastructure for both civil and military communications today. In addition, many militaries, governments, and commercial enterprises also employ closed private networks. Many of these private networks have interconnection points with public communications networks, either integrated as part of their design or created on an accidental, ad hoc basis.

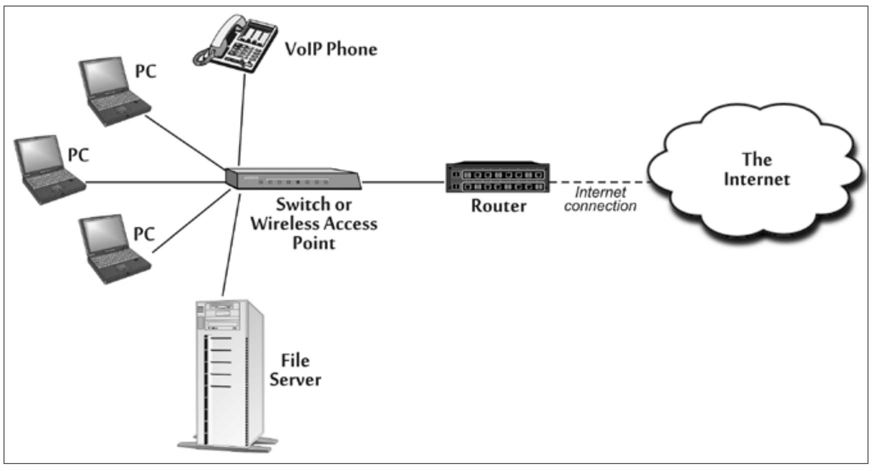
Network Building Blocks

Most networks are made up of smaller subnetworks. For example, among the subnetworks of the telephone network are smaller telephone networks operated either by telephone companies or independent enterprises, all of which are interconnected at various points. The original telephone network employed switchboards on which telephone operators established a stable circuit between two or more parties. The circuit remained dedicated to those particular users as long as the call was maintained. In such *circuit-switched* networks, the network allocates resources—originally a physical connection using switched wires, later a timeslot in a protocol—for every communicating session, established in real time, to move information across the network. With circuit switching, then, the network is “aware” of individual communicating sessions, tracking them constantly and allocating resources for each of them.

Instead of the circuit switching of the traditional telephony network, the Internet relies on the alternative technology of *packet switching*. Instead of the network dedicating resources between the end systems for each communications stream, the data that make up that communication are broken up into separate packets or chunks of data and each is delivered independently by the network to the desired destination. “Header” information is attached to each packet to help the network move the packets to the right place. The network itself is generally ignorant of the relationships between the packets but instead focuses on getting individual packets to their destination as quickly as possible.

The fundamental packet-switching network building block of the Internet is a local area network (LAN), an extremely focused form of subnet in a relatively localized geographical area. Most enterprise networks for corporations, government agencies, and military organizations are groups of LANs. Increasingly, LAN technology is being deployed in homes so that disparate devices such as personal computers, voice over IP (VoIP) telephones, and servers can communicate with each other. Components on a single LAN are typically connected using a switch or wireless access point, perhaps using wired Ethernet or WiFi wireless technology; these simple devices send packets directly between interconnected systems over short distances ranging from one to a few hundred meters. Figure 4–2 shows a common LAN setup.

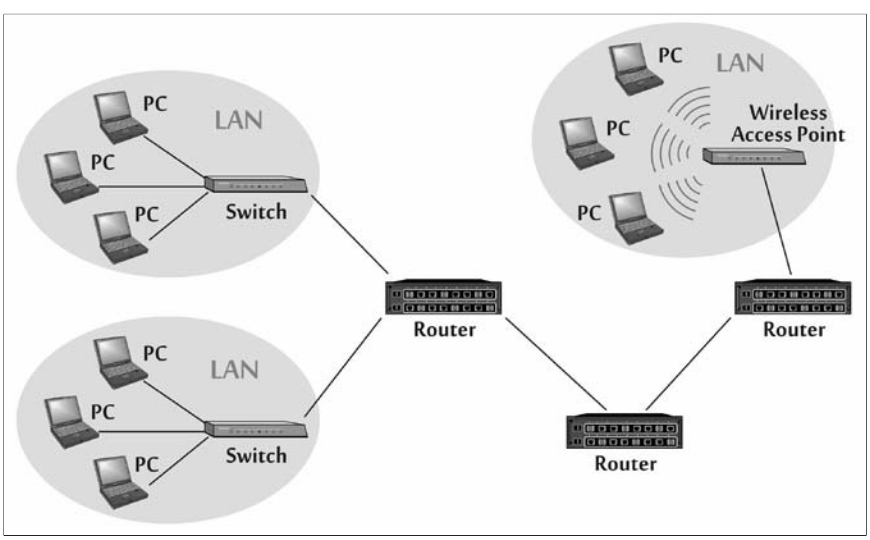
Figure 4-2 A Local Area Network



LANs are connected by routers, devices that can direct packets to their destination subnets based on the addresses in the packets themselves. Routers can also connect LANs to point-to-point links to create even larger networks. Point-to-point link technologies include T1, Digital Signal 3 (DS3, also known as T3), cable modem, fiber optic service (FIOS), and digital subscriber line (DSL), a relatively higher speed network connection over telephone network wires. The result is a network of networks, or internetwork, of a multitude of interconnected LANs, routers, and point-to-point links.[1]

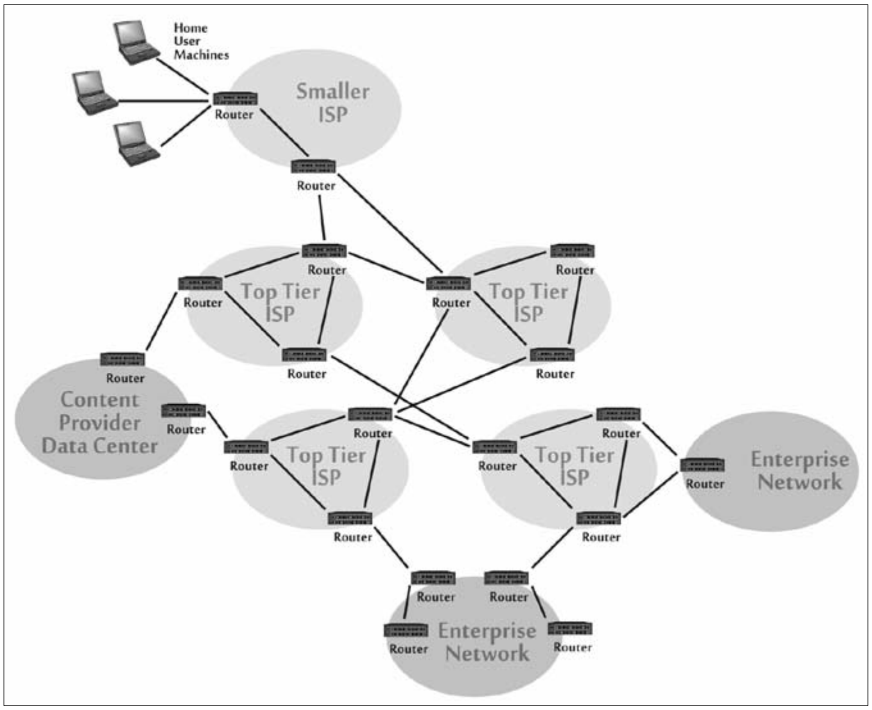
Figure 4–3 depicts an internetwork of various LANs, routers, and point-to-point links, such as might be found inside a small or medium-sized enterprise.

Figure 4-3 Internetworking Local Area Networks via Routers and Point-to-Point Links



The Internet is a publicly accessible internetwork with global reach tied together by means of common use of IP. Physically, the Internet is made up of a group of “backbone” routers operated by top-tier ISPs (see figure 4–4).

Figure 4-4 Internet Links to Backbone Routers Operated by Internet Service Providers



A few hundred high-end routers distributed around the world constitute this backbone, making up the major branches of the network architecture, moving packets across ISP networks and also between ISPs. The top-tier ISPs offer smaller ISPs access to the Internet backbone. Both the top-tier and smaller ISPs also give enterprise networks, individuals, and content providers access to the backbone infrastructure. In the United States and some other countries, numerous top-tier ISPs compete for this business. In other countries, a single dominant provider, often the legacy telephone company, acts as the only ISP.

From the introduction of the Internet in the early 1970s until today, the legacy communications network has provided connectivity for networked computers in addition to telephones. Other types of enterprises, including cable companies, wireless operators, and dedicated data communications carriers, have also begun to offer carriage of Internet data as part of their business. Bandwidth has been increased significantly through the use of better signal processing, more rapid transmitters, and fiber optic communications.

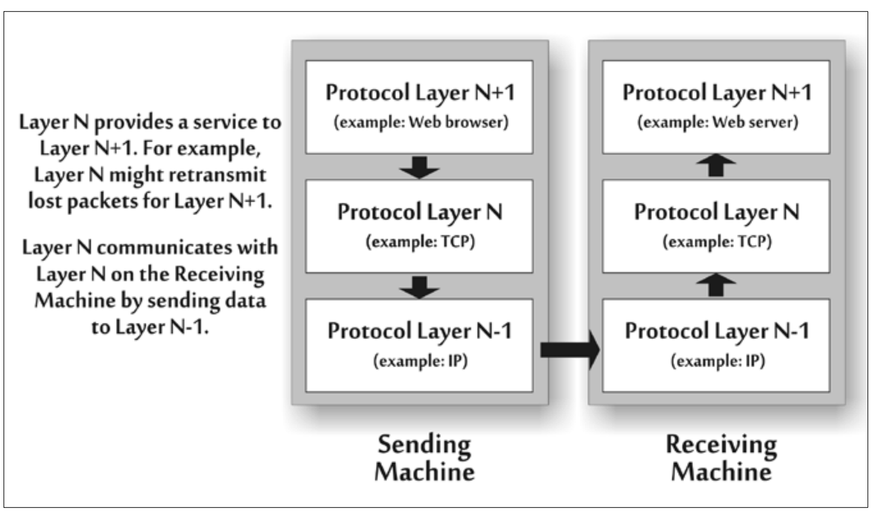
The communications networks require a significant support infrastructure, including the electrical power grid, required for operating all components of the networks. Supervisory control and data acquisition (SCADA) devices, which consist of simple computers directly connected to industrial equipment, are used to monitor and manage many types of utility networks and industrial operations, including electrical power grids, gas pipelines, and manufacturing lines. Increasingly, the SCADA devices themselves are managed remotely across a private network or even the public Internet. This approach introduces a significant dependency loop: the communications infrastructure relies on the electrical power grid, which is controlled with SCADA systems that rely on the communications infrastructure. There is thus is a risk of cascading failures. Another vital aspect of the support infrastructure is the manufacturing base, which creates the physical equipment that makes up the various systems of cyberspace. These, too, may rely on SCADA systems and their communications networks, again raising the risks of interdependency.

Another View: Protocols and Packets

So far, this chapter has analyzed how internetworks and the Internet itself are built up of smaller building blocks such as LANs and point-to-point links. To further analyze the elements of the systems domain of cyberspace, a different view of networking can be helpful: the protocol and packet view. This view helps explain how different machines connected to the same Internet communicate with each other by sending packets according to standard protocols.

Protocol layering is a critical concept underlying this view of network communications. In 1980, the International Organization for Standardization released the Open Systems Interconnection (OSI) Reference Model, a generic description of how computer-to-computer communications could operate using protocol layering. In this model, a series of small software modules on each system perform a set of tasks that allow two computers to communicate with each other. For example, one module might focus on making sure that data are formatted appropriately, another takes care of retransmitting lost packets, and yet another transmits the packets from LAN to LAN across the network. Each of these modules, referred to as a *layer*, has a defined small job in communication (see figure 4–5).

Figure 4-5 Protocol Layering



The software of a given layer on the sending machine communicates with the same layer on the receiving machine. A layer is a collection of related functions that provides services to the layer above it and receives service from the layer below it. For example, one lower layer might send packets on behalf of the higher layer that is focused on retransmitting lost packets. This higher layer, in turn, serves an even higher layer that generates the data in the first place. In the example of figure 4–5, a layer of software inside a Web browser generates data to send to a Web server. This Web browser application passes the data to the transmission control protocol (TCP) layer software on the sending machine, which provides several services, including retransmitting lost packets. The TCP layer passes the software down to the IP layer, which provides the service of carrying the packet end to end through all the routers on the network. Although one layer relies on another to get things done, the layers are designed so the software of one layer can be replaced with other software, while all other layers remain the same. This modularity has proven especially useful in deploying new types of networks—for example, as IP version 4 (IPv4) networks are transitioned to IP version 6 (IPv6), the successor protocol for the Internet.

The communications modules taken together are called a *protocol stack* because they consist of several of these layers, one on top of the other. The OSI conceptual model defined by the International Organization for Standardization in 1980 includes seven such layers, each with a defined role in moving data across a network. At the "top," layer seven, the *application layer*, acts as a window to the communications channel for the applications themselves by interpreting data and turning it into meaningful information for applications. The application might be, for example, a Web browser or server, an email reader or server, a peer-to-peer file copy program, or an enterprise financial system.

Layer six, the *presentation layer*, deals with how data elements will be represented for transmission, such as the order of bits and bytes in numbers, the specific method for encoding textual information, and so on.

Layer five, the *session layer*, coordinates sessions between two communicating machines, helping to initiate and maintain them as well as to manage them if several different communications streams are going between them at the same time.

Layer four, the *transport layer*, supports the reliability of the communications stream between two systems by offering functions such as retransmitting lost packets, putting packets in the proper order, and providing error checking.

Layer three, the *network layer*, is responsible for moving data across the network from one system, possibly across a series of routers, to the destination machine. This layer is absolutely critical to making the network function end to end.

Layer two, the *data link layer*, moves data across one "hop" of the network, getting it from one system, perhaps to its destination on the same LAN, or to the nearest router, so it can be sent between LANs or to a point-to-point link.

At the "bottom" of the stack, layer one, the *physical layer*, actually transmits the bits across the physical link, which could be copper, fiber, wireless radio transmitters and receivers or another physical medium.

Today's Internet is loosely based on the OSI model, but it does not break out each layer exactly as the OSI model specifies. Most commonly, IP is paired with a transport protocol called the transmission control protocol (TCP)—hence the term TCP/IP to refer to the duo of protocols in most common use on the Internet today. TCP/IP is roughly equivalent to layer four (transport) and layer three (network) of the OSI Reference Model, plus a little interaction with layer two (data link). Everything above TCP/IP is left to the application with the application, presentation, and session layers (seven, six, and five) of the OSI Reference Model all folded into the application program itself. TCP/IP is mainly responsible for transmitting data for that application. It is important to note that the application layer is not TCP/IP itself: the application comprises the particular program trying to communicate across the network using TCP/IP. The application might be, for example, a Web browser and a Web server, or two mail servers, or a video player communicating with a streaming video server, or a file transfer protocol (FTP) client and server. Based on the OSI model, the application layer is often referred to as *layer seven*, even in TCP/IP networks.

The transport layer could be the transmission control protocol or its cousin, the user datagram protocol (UDP), a simpler but less reliable protocol. The transport layer ensures that packets are delivered to the proper place on the destination machine. For those applications requiring such functionality, TCP also delivers packets in the proper sequence or retransmits packets.

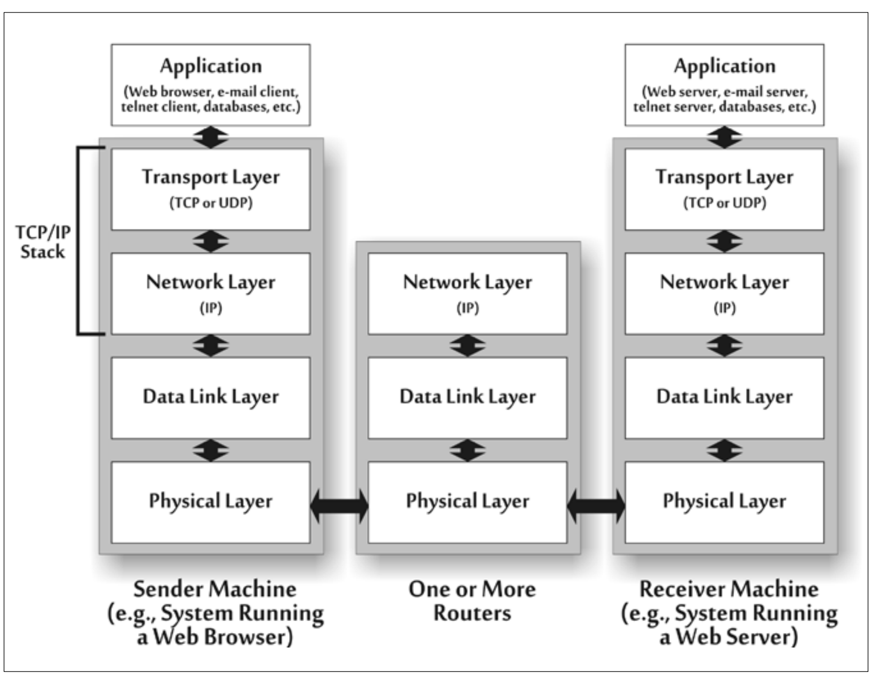
The network layer is based on the IP. Its purpose is to deliver packets end to end across the network, from a source computer to a given destination machine. Using terminology from the OSI Reference Model, the IP layer is sometimes referred to as *layer three*.

The data link layer transmits each packet across each hop of the network. For example, this layer moves data from a home computer to a router that connects the LAN to the Internet. Then, the router uses its data link layer software to move the data to another router. In the OSI Reference Model vernacular, the data link layer is often referred to as *layer two*.

The physical layer, called *layer one*, is the physical media, such as the wire, fiber optic cable, or radio frequencies, across which the information is actually transmitted.

To illustrate how these layers on IP networks typically communicate, figure 4–6 shows an example in which two computers, the sender machine and the receiver machine, communicate.

Figure 4-6 Protocol Layering to Transmit Packets on an Internet Protocol Network



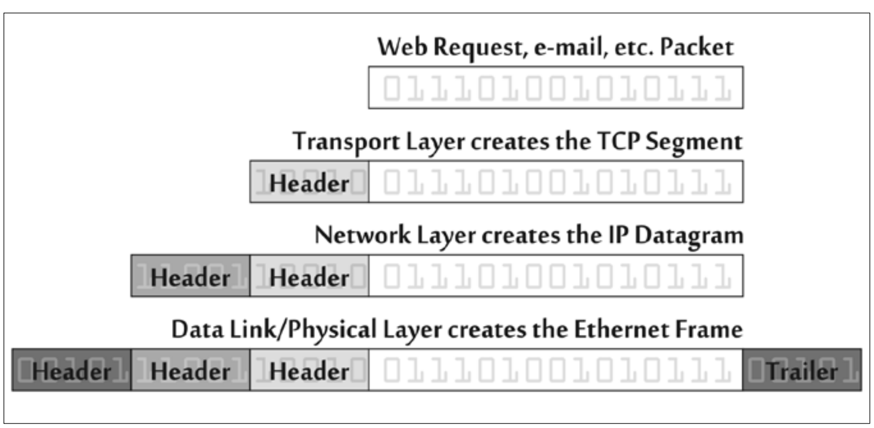
Suppose a user on the sender machine wants to surf the Internet with a Web browser application such as Internet Explorer or Firefox to access a Web site. The browser on the sender needs to communicate with the Web server on the receiver, so it generates a packet and passes it to the TCP/IP stack software running on the sender machine. The data, which consists of a Web request, travel down the communications layers on the sender to the physical layer and get transmitted across the network (which usually consists of a series of routers). The packet is sent through one or more routers this way, until it reaches the receiver machine. It then travels up the receiver's communications stack.

To start this process, the sender's transport layer (that is, TCP software running on the sender machine) takes the packet from the browser application and formats it so it can be sent reliably to the transport layer on the receiver. This TCP software also engages in a packet exchange (called the TCP Three-Way Handshake) to make sure all of the sender's packets for this connection arrive in sequence. (Other types of transport layer protocols, such as UDP, do not care about sequence, so they have no such packet exchange for ordering packets.)

Just as the two applications, here the Web browser and the Web server, communicate with each other, so do the two transport layers. On the sender, the transport layer passes the packet down to the network layer, which delivers it across the network on behalf of the transport layer. The network layer adds the source and destination address in the packets, so they can be transmitted across the network to the receiver's network layer. Finally, the data are passed to the sender's data link and physical layers, where it is transmitted to the closest router on the way to the destination. Routers move the packet across the network, from subnet to subnet. The routers include the network, data link, and physical layer functions required to move the packet across the network. (Because these routers are focused on moving packets, not receiving them, they do not require the transport or application layers to deliver the packet to the receiver.) On the receiver side of the communication, the message is received and passed up the protocol stack, going from the physical layer to the data link layer to the network layer to the transport layer to the ultimate destination, the application.

This passing of data between the layers is illustrated in figure 4–7.

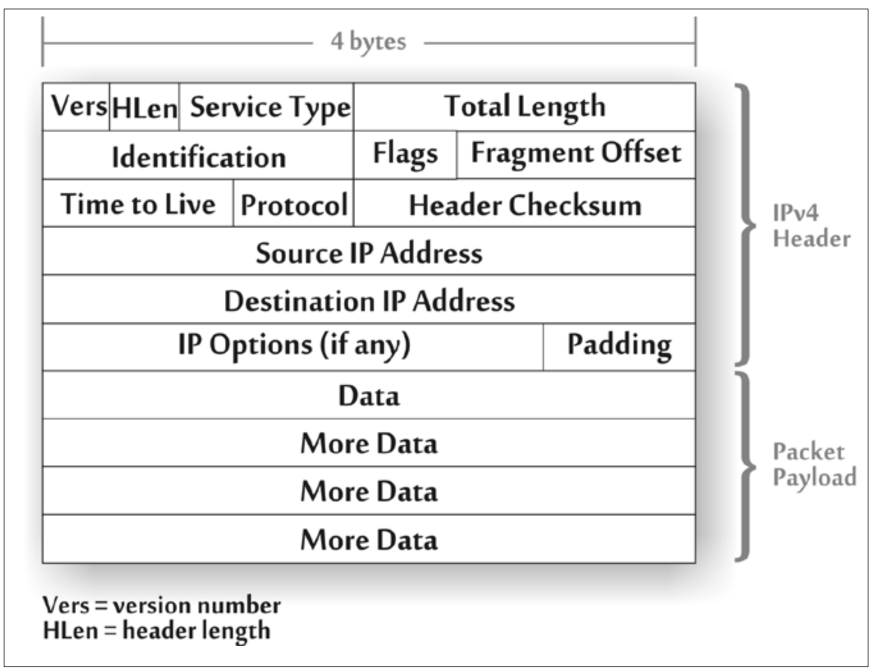
Figure 4-7 Protocol Layering Applying Various Headers and a Trailer



Each layer attaches some information in front of (and in some cases, behind) the data it gets from the layer above it. The information added in front of the data is called a *header*, and it includes critical information for the layer to get its job done. As figure 4–7 shows, the application generates a packet. This packet might be part of a Web request, for example, or a piece of an email message. The transport layer adds a header to this data, which is likely to include information about where on the destination machine the packet should go. When TCP is used, the resulting header and data element is called a TCP *segment*. The TCP segment gets passed to the network layer, which adds another header with information about the source and destination address in the IP header. This header is analogous to an envelope with a postal address for the data. The resulting packet is called an IP *datagram*. This package is sent to the data link and physical layers, where a header (and a trailer) are added to create a *frame* that makes it possible for the data to be transmitted across the link.

The packets sent between machines pass through different layers of this stack and have various headers in front of them. The one layer that all systems and routers communicating with each other on the network must conform to is the network layer; for the Internet, the network layer is the Internet Protocol. Today's Internet relies on IPv4 from one end to the other. Certain subnets and routers also support the successor protocol, IPv6. Every packet sent across the Internet today using IPv4 has the structure shown in figure 4–8. It includes a source IP address (a 32-bit number indicating the network address of the system that sent the packet) and a destination IP address, which routers use to determine where to send the packet.[2] Other fields of the IPv4 header are associated with controlling the number of hops a packet can take as it traverses the network (the time-to-live field), fragmentation (which breaks larger packets into smaller ones), and other network functions.

Figure 4-8 An Internet Protocol Version 4 Packet



Although the packet in figure 4–8 looks two-dimensional, that formulation is done purely for diagram purposes. Each field is transmitted on the wire (or wireless radio) one byte after another, linearly. The version number (such as IPv4) goes out first, followed by the header length (the size of the overall header in front of the data), field by field as shown, through total length, the identification field, which is associated with fragmentation, and so on. Bit after bit, the packet leaves the sender source. When it reaches a router, the router absorbs the packet on one network interface, inspects its various fields, and moves it to the appropriate network interface for transmission to its destination, or one router hop closer to its destination. In this way, a packet is sent and routed across the Internet.

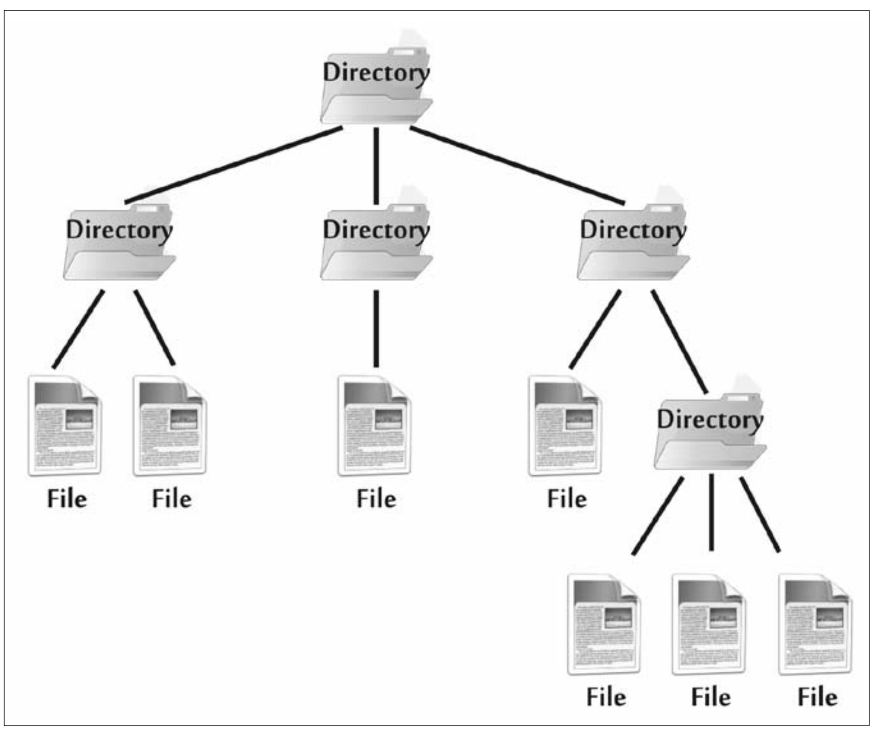
The Content and Applications Domain

The systems domain of cyberspace provides the technical underpinnings of the network, but it is merely an infrastructure on which to store, transmit, and manipulate content or information using various software applications. The content and applications domain rides on top of the systems domain and provides the usable applications and the information they handle. This section describes some of the common methods for content storage in cyberspace, provides an overview of various application architectures, and outlines some of the most common application types used today.

Content Storage

Although content is stored on computer systems in cyberspace in many ways, two methods of information storage dominate: hierarchical file systems and relational databases. Nearly all cyberspace applications rely on at least one of these concepts; most applications use both. In a hierarchical file system, as shown in figure 4–9, one or more computers stores information in individual files, which themselves are made up of small sections of the hard drive or chunks of memory, depending on the computer system.

Figure 4-9 A Hierarchical Files Structure

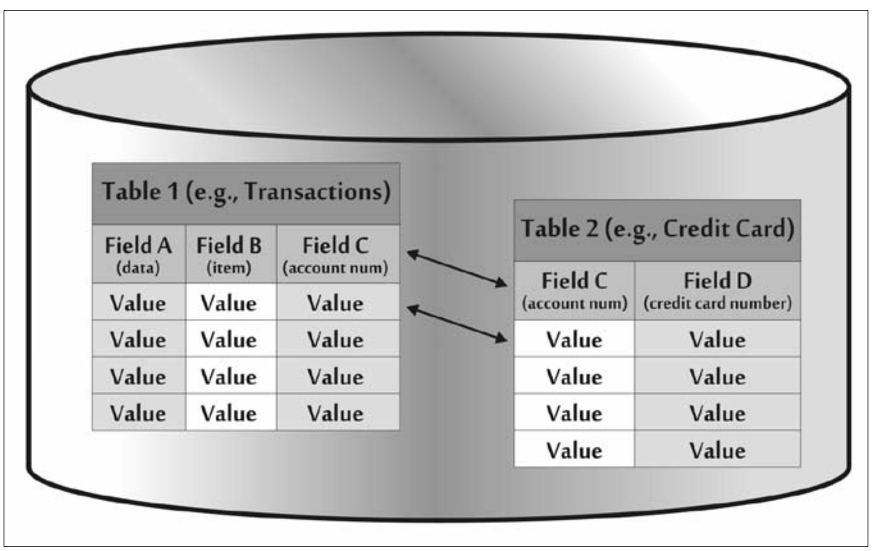


Files on a typical computer system include the software of the operating system itself (such as Windows, Linux, or Apple's Mac OSX), executable programs that make up the applications of the computer (such as Adobe Acrobat, a Web browser, or a word processing program), configuration files that hold settings for the programs on the machine, and the data stored on the system by the application and its users, such as document files. Files are located inside of directories (also called folders on some operating systems). The directories themselves may contain subdirectories. This results in a hierarchical structure.

Today, many computer users take this structure for granted, and may assume that this organization of content is the way that computers necessarily work. But early computers did not have such a structure. Development of the hierarchical file structure revolutionized human interactions with machines. A hierarchical file system is useful because it provides an unambiguous way to refer to files, indication that there is a relationship between files—for example, files in the same directory are likely to have some common purpose—and a method for navigation between files. This paradigm is so useful for information storage that it is found in all of the major general-purpose operating systems available today— including Windows, Linux, UNIX, Mac OSX, and others—across various kinds of computing equipment, including laptop computers, desktops, and servers as well as many cell phones, music players, and video storage devices. Even systems without hard drives are increasingly likely to use a file system implemented in memory.[3]

Another common method for storing content involves a relational database. As illustrated in figure 4–10, a relational database stores information in a series of tables. The tables specify a number of fields under which specific types of data, called values, are stored. In the value section of each field, all of the elements in a given row of the table relate to each other as a single record in the database. Two or more tables may have the exact same field in them to forge a relationship between them. Such a relationship is illustrated by field C in the figure.

Figure 4-10 A Simple Relational Database



For example, suppose the database in the figure is associated with electronic commerce. Table 1 might hold transaction information, perhaps indicating items various people have bought from an online store. Table 2 might hold customer credit card information. Within table 1, fields A, B, and C could be, respectively, the date, the item purchased, and the account number of the purchaser. Table 2 could also hold field C, the account number, and, in addition, field D, the credit card number for that customer. Data are presented in these tables discretely, for flexible search and update. A program or user could write a query and pull out or update information directly from table 1, analyzing transactions or adding new ones as purchases are made. Because of the relationship between the tables, which is embodied in field C (the account number), database query software can research, for example, the number of times a given credit card number (table 2 data) was used on a given date (table 1 data). This "join" operation offers significant flexibility in handling and manipulating content. Of course, figure 4–10 is a highly simplified diagram to illustrate the fundamental constructs of a relational database. Most real-world databases have dozens or hundreds of tables, each with 2 to 20 or more fields and thousands to millions of rows.[4]

Relational databases are the most popular form of database storage today in market share. However, other kinds of database storage options are available, including hierarchical databases, object-oriented databases, and flat files, each offering different performance and functions from relational databases.

Although relational databases and hierarchical file systems are fundamentally different paradigms for organizing content, the two have some connections. Relational databases are, for example, almost always implemented as software running on top of a hierarchical file system. That is, the database programs themselves consist of a group of hierarchically organized files. The database tables and their contents exist inside of files as well. Just as large-scale packet-switched networks are often built up of point-to-point links from circuit-switched networks, relational databases are an overlay on top of a file system, offering a more flexible way of organizing information for query and update. In addition, most applications in cyberspace today are a mixture of files interacting with a back-end database using a variety of application architectures that manage the interaction of the files and the database. Such systems are referred to as back-end infrastructure to differentiate them from the front-end systems such as browsers that users run to initiate transactions.

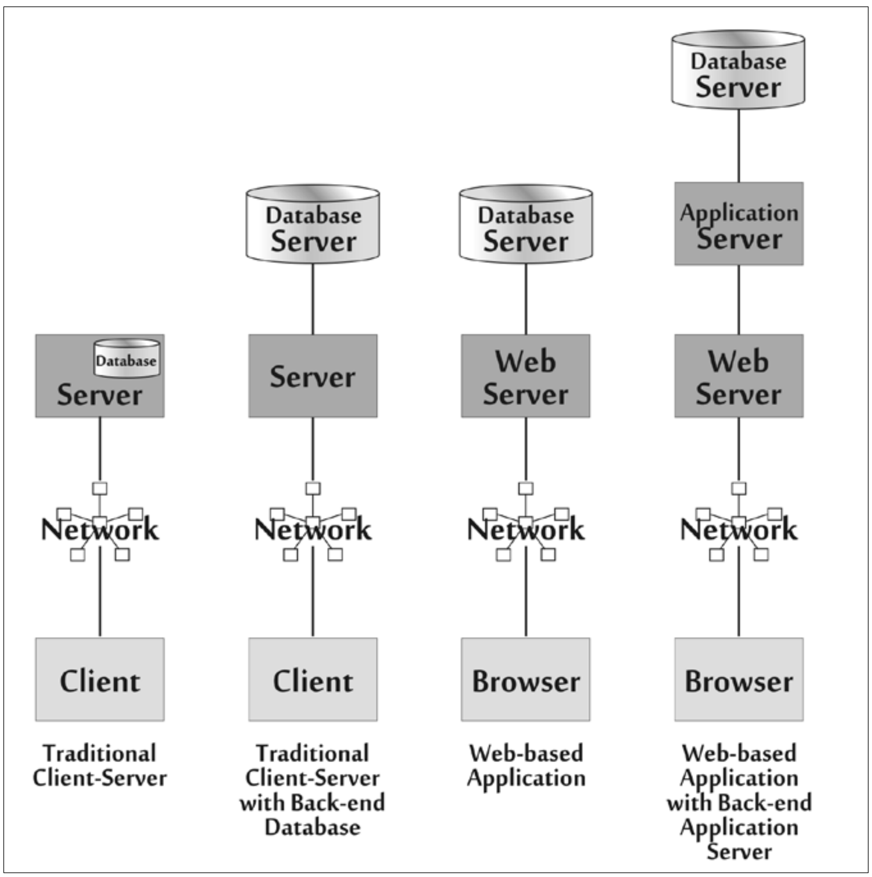
Application Architectures

From the 1950s through the 1970s, computer applications typically resided on mainframe computers. Users logged into the system from "dumb" terminals. These terminals displayed information stored and processed on the mainframe but did no processing themselves. They were connected to the mainframe using fairly limited network protocols. All aspects of the application were stored on the mainframe itself.

Starting in the 1980s, personal computers (PCs) with limited processing power began replacing dumb terminals. In the 1990s, many mainframes were replaced with lower cost and smaller server machines. Both enterprise and public networks connecting PCs and servers began to make significant use of IPv4. A variety of new, distributed application architectures arose. Figure 4–11 depicts some of the common consumer and business application architectures in use today on enterprise networks and on the Internet.

The common applications architecture depicted on the left of figure 4–11 is a traditional client-server application. The client consists of a custom program dedicated to that application running on a desktop or laptop computer. The client handles the user interface, some data processing, and communications with a server, accessed across the network. The server may include database software, although some client-server applications instead place the database on a separate machine, as shown in the second architecture of the figure. This separate database can then support multiple application servers.

Figure 4-11 Common Modern Application Architectures



One of the concerns of the architectures on the left half of figure 4–11 is the distribution and update of specialized client-side software. Each application has a custom program that controls access to the server. Thus, synchronization of updates to several different client applications is rather complex. In the mid- 1990s, various applications began to move to the third architecture of the figure, which involves using a Web browser as a generic client that formats, displays, and processes information for the user, relying on a single program—the browser— for multiple applications, thus simplifying client-side updates. This generic browser accesses a server across the network, which in turn accesses a back-end database server. This so-called three-tier architecture is the dominant enterprise and Internet application architecture today. Some applications introduce an additional element, an application server, shown in the fourth example of the figure. The application server handles some of the critical information processing capabilities that were previously handled by the Web server.

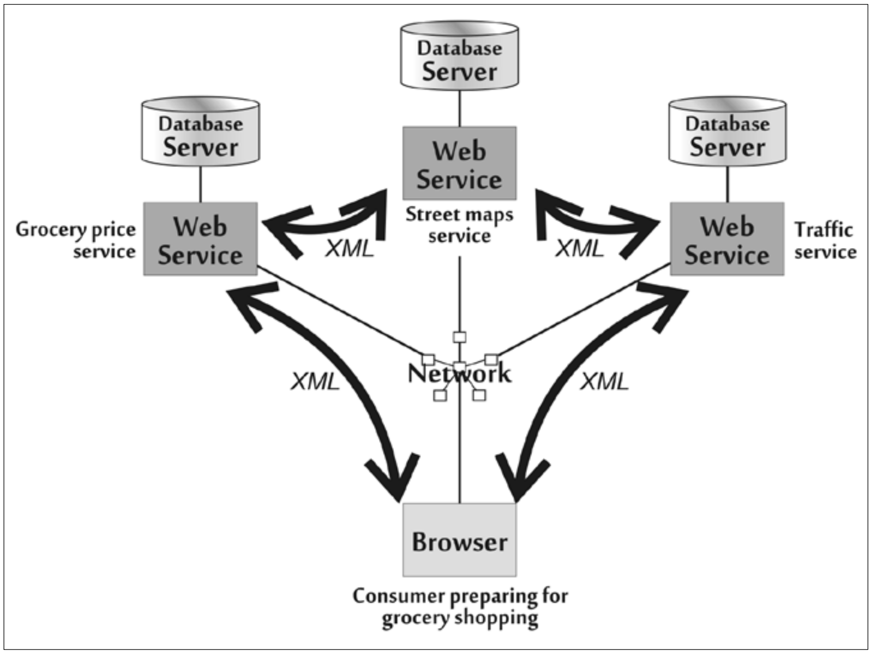
While the three-tier architectures shown on the right side of figure 4–11 will continue to be dominant for several years, a newer architecture paradigm started to emerge in the mid-2000s: the so-called service oriented architecture (SOA) model. Figure 4–12 illustrates this approach. In this model, client-browser software still accesses Web servers; often, this interaction relies on the Extensible Markup Language (XML), a flexible format for defining data structures to be exchanged on the network. Web servers may, similarly, interact with other Web servers using XML to pull together information for the client, processing information in a distributed, coordinated fashion on the network. This is sometimes called computing “in the cloud.” Using this SOA model, different functions, such as geographical mapping, product search, or price calculation and comparison, can be distributed across different servers. Applications can be stitched together by information flows between different Web-enabled services. An example might offer a user a way to create a consumer grocery shopping list that minimizes cost and travel time. A user enters a desired grocery shopping list and starting location into a browser. The browser sends XML requests to a grocery pricing Web service that responds with prices at various local grocery stores. This Web service might pass on a request to a separate street maps Web service, asking for directions between the consumer and different stores. The service might then forward a request to a traffic service that provides information about clogged highways. All of the information is reassembled at the user’s browser, indicating the most efficient options for buying the desired groceries.

Common Application Types

The architectures shown in figures 4–11 and 4–12 are used to create many application types on internal networks and the Internet. New applications are constantly being devised and becoming popular. Some of the most common applications on today’s Internet include email, instant messaging, search engines, and others described below.

Email is one of the oldest and most widely used applications of the Internet. Many companies are heavily dependent on their email infrastructure for communication between employees and also as the technical underpinnings of vital business processes. Email servers exchange email messages, which are accessed by users running either specialized email reading software or a general-purpose Web browser. In addition to user-to-user email, hundreds of thousands of mailing lists are used by communities with shared interests to exchange information to all subscribers. Email is also used to spread malicious code and online scams.

Figure 4-12 A Service-Oriented Architecture



Email is store-and-forward technology: messages wait for users to access them. By contrast, instant messaging focuses on real-time exchange of messages between users. Many such chats occur one-on-one, as two users send information to each other. Chat rooms allow many users to exchange information simultaneously. America OnLine, Yahoo! Messenger, and Microsoft Messenger, as well as independent, open standards for messaging such as Jabber and Internet Relay Chat, are widely used on the Internet. An alternative technology used for cell phones, called Short Messaging Service, has much the same functionality but different underlying communication protocols.

Search engines are another important set of applications on the Internet. Search engines employ their own specialized browsing machines, known as crawlers, to fetch billions of pages from the Internet by following links from one page to the next, discovering new pages as they are posted and linked to other pages. The search engine company's software then assembles a searchable index of the pages fetched by the crawlers, offering users a Web-based front- end service by which users can search the massive index, which is stored in a distributed database maintained by the search engine company.

E-commerce applications have burgeoned since the late 1990s, as retailers have set up Web sites to sell products and services to users. Consumer e-commerce companies such as Amazon.com and Apple's iTunes Store are among the most familiar. Numerous business-to-business e-commerce applications are also in use. Business management systems have moved online to automate business purchasing, payroll, supply-chain management, and other vital aspects of enterprise operations. Many organizations have deployed enterprise resource planning and enterprise resource management systems such as those offered by SAP, PeopleSoft, Oracle, and Microsoft.

Wikis are Internet-based applications that use specialized Web server software to inventory, categorize, and store information about a given topic or group of topics and allow it to be created and updated quickly. The most widespread and widely used wiki is Wikipedia, a comprehensive encyclopedia of information that can be updated by anyone who wishes to contribute to the information.

Blogs (a shorted form of “web logs”) are another important kind of Internet application, made up of online diaries that are frequently updated. Most blogs are devoted to a single topic, such as politics, news, sports, a business or industry, or hobbies. Hundreds of thousands of blogs are available today, and some acquire readership numbering in the hundreds of thousands of users per week.

Social networking sites, another fast-growing form of Internet application, allow users to store information online in a personal profile and then link that profile to those of friends or business associates. By following these links, people with related interests can reach out to each other, keeping personal relationships up to date via socially oriented services such as MySpace or Friendster, or making business connections using professional services such as LinkedIn and Orkut.

In other applications, users are increasingly turning to the Internet for audio and video news, information, and entertainment. Such services are pack- aged as podcasts, audio or video shows that are automatically downloaded to user machines periodically when new content is published by the podcast author. Radio shows, television programs, and content from other mass media are published in tens of thousands of free and commercial-subscription podcasts on a daily basis. Downloadable and streaming video services with short-form videos are also becoming quite popular; offerings include those of YouTube and Google Video. Increasingly, entire television programs and movies are available on the Internet through related services, such as Amazon’s Unbox and the television studios themselves.

Peer-to-peer file-sharing services allow large number of users to share files on the Internet, copying music, movie, and other files automatically between user systems. User machines that join the peer-to-peer network are used to store files that are redistributed to other users without any direct user interaction. Some of the most popular peer-to-peer networks are BitTorrent, Gnutella, and eDonkey. Many of these networks are used to exchange song and movie files in violation of copyright, but they are also increasingly used by legitimate publishers to distribute movies and television shows on a pay-per-view basis.

Internet telephony applications such as Gizmo and Skype offer free or very low-cost telephone calls from a user’s computer across the Internet. Gizmo and Skype carry calls from end-user machine to end-user machine without traversing any intermediary servers. Vonage, on the other hand, carries calls via VoIP servers, which provide value-added services, such as voice mail. Because they often undercut the cost of long distance and international calls, both kinds of Internet telephony services are starting to supplant more expensive traditional telephone calls.

Another important kind of Internet application bridges cyberspace and the real world:

mapping applications. These tools, often offered for free on the Inter- net, provide comprehensive street maps and driving directions on the Internet; MapQuest and Google Maps are dominant.

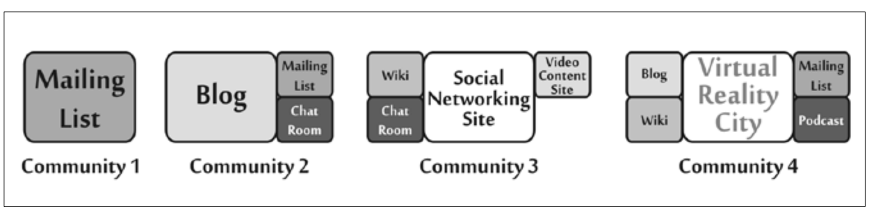
With virtual reality sites, users are presented with a well-developed “virtual world” that they can explore using an avatar, a digital icon representing their online persona. People meet in virtual reality sites for a variety of reasons, including social interactions, business relationships, and online gaming. Second Life is one of the most popular virtual reality sites, in which users interact in social and business settings. Many online games are also a form of virtual reality simulation, including the popular World of Warcraft game. Within some virtual reality simulations, different “cities” are founded with different focuses, such as a given hobby or business pursuit.

While these are among the most popular applications on the Internet today, the content and applications domain of the Internet is constantly changing and growing. New applications rise to prominence frequently, as they empower people to interact with their information and with each other on a more flexible basis.

The People and Social Domain

Cyberspace is a human-made artifact, created to gain access to information and share it between people and machines. With the increasing presence of cyberspace in modern life, the underlying systems domain and content and applications domain have given rise to a new people and social domain as individuals build communities in cyberspace. Numerous types of online communities exist in cyberspace, visited by users from around the world who share personal, business, political, or other interests. These communities are sometimes built using a single application: a community may be entirely based on an email list of hobbyists. More commonly, however, online communities consist of people who use multiple applications for different aspects of their community; the hub might consist of a particular blog, augmented with mailing lists, sections in video-sharing sites, chat rooms, or cities within a virtual reality simulation, as shown in figure 4–13.

Figure 4-13 Online Communities Built of One or More Applications



Such cyberspace communities have been founded for numerous purposes. New communities form on a regular basis and older communities dry up. Flourishing communities today include those built around social interactions, such as dating sites, teenage hangouts, and corporate “virtual water coolers.” Nearly every hobby imaginable - model rocketry, chess, recipes, and more—has a Web site, and many have full-blown online communities.

Online news communities include the Web sites of major news services such as CNN and the *New York Times*, local newspapers, local television news, professional bloggers who work as independent journalists supported by online ads, and amateur bloggers who fact-check other news sources and provide commentary. Over 50 million Americans read news online, according to a 2006 survey by the Pew Internet and American Life Project, and the number is trending upward rapidly.

Numerous online communities are devoted to discussions of a given type of technology, helping users understand and relate to the technology more effectively, such as the Web sites Slashdot, devoted to open-source software and related technologies, Ars Technica, focused on detailed aspects of PC hardware and software, and Gizmodo, focused on technical gadgets and consumer electronics.[5]

Many health care–oriented online communities help doctors and patients share information to better understand and cope with various medical conditions. Online support communities offer help for people suffering from particular diseases.

Adherents of many religions, major and minor, have created online communities for proselytizing, fundraising, and placing their messages and beliefs in front of a wider audience.

Political communities on the Internet are used for debate, analysis, and fundraising activities. Several popular blogs from nearly every aspect of the political spectrum, and their associated mailing lists and online video distribution sites, are increasingly helping to shape political messages and candidacies.

Both business-to-consumer and business-to-business communities have flourished in cyberspace, helping to make business processes more efficient. Sites such as eBay and Amazon.com offer storefronts through which numerous other organizations and individuals sell to consumers. Some industries have established their own online communities to improve efficiencies in bidding and service delivery. For example, American and some international automotive companies created the Automotive Network Exchange.

Not all communities in cyberspace have beneficial impacts. Terrorists rely on cyberspace to recruit, plan, and spread propaganda. International criminals likewise use cyberspace to commit crime and track their business ventures, as described in more detail in chapter 18, “Cyber Crime.”

The walls between these types of communities are permeable: a single individual may participate in numerous communities. The communities themselves may also interconnect: communities created for social interactions may flow into hobbyist communities or start to engage in political debate.

Conclusion

As cyberspace continues to expand, the diversity of elements in all of the domains discussed in this chapter is likely to increase. The underlying networks will become more complex, pulling in other kinds of technologies and other protocols. The application mix will be updated as people apply technology to new uses. Online communities will continue to evolve and merge as technology grows more immersive and people grow more accustomed to living various aspects of their lives in cyberspace.

This evolution of cyberspace also appears to be increasing in speed. Much of cyberspace got started in the 1970s as computers were interconnected and primitive email systems were established. In the 1990s, Web sites for information distribution became popular. In the early 2000s, search engines and e-commerce flourished. Many of the most popular applications and online communities are even newer, such as blogs, wikis, and virtual reality applications. With this pace of evolution continuing to accelerate, cyberspace is likely to have an increasing impact on the economy and society in coming years. For this reason, policy decisionmakers need to understand the underpinnings of the technology and its rapidly changing status to help inform their actions.

---

[1] While IP-based networks are packet-switched technologies, individual links, especially stable point-to-point links with constant connectivity, are usually carried over circuit- switched technology provided by legacy telecommunications, cable, or cellular networks.

[2] IPv6 uses 128-bit addresses.

[3] The hierarchical file system may or may not be visible to the user. Some operating systems and applications hide this organizational structure from users, trying to simplify human interactions with the machine.

[4] Queries for relational databases are usually formatted in the Structured Query Language (SQL), a standardized language for interacting with databases. Relational database software is available from a large number of commercial software companies, such as Oracle and Microsoft. An open-source relational database called MySQL is also popular.

[5] The Slashdot Web site is <http://slashdot.org>; Ars Technica is available at <www. arstechnica.com>; and the Gizmodo Web site is <http://gizmodo.com>.